TEXAS INSTRUMENTS

DRA790, DRA791
DRA793, DRA797

SPRS968F – AUGUST 2016 – REVISED NOVEMBER 2019

DRA79x Infotainment Applications Processor

1 Device Overview

1.1 Features

- Architecture designed for infotainment co-processor applications, hybrid radio and amplifier applications
- Arm® Cortex®-A15 microprocessor subsystem
- C66x floating-point VLIW DSP
  - Fully object-code compatible with C67x and C64x+
  - Up to thirty-two 16 × 16-bit fixed-point multiplies per cycle
- Up to 512KB of on-chip L3 RAM
- Level 3 (L3) and Level 4 (L4) interconnects
- DDR3/DDR3L Memory Interface (EMIF) module
  - Supports up to DDR-1333 (667 MHz)
  - Up to 2GB across single chip select
- Dual Arm® Cortex®-M4 Image Processing Units (IPU)
- Display subsystem
  - Display controller With DMA engine and up to three pipelines
  - HDMI™ encoder: HDMI 1.4a and DVI 1.0 compliant
- Video Processing Engine (VPE)
- One Video Input Port (VIP) module
  - Support for up to four multiplexed input ports
- General-Purpose Memory Controller (GPMC)
- Enhanced Direct Memory Access (EDMA) controller
- 2-port gigabit ethernet (GMAC)
  - Up to two external ports
- Sixteen 32-bit general-purpose timers
- 32-Bit MPU watchdog timer
- Six high-speed inter-integrated circuit (I^2C) ports
- HDQ™/ 1-Wire® Interface
- Ten configurable UART/IrDA/CIR modules
- Four Multichannel Serial Peripheral Interfaces (McSPI)
- Quad SPI Interface (QSPI)
- Media Local Bus Subsystem (MLBSS)
- Eight Multichannel Audio Serial Port (McASP) modules
- SuperSpeed USB 3.0 dual-role device
- High-speed USB 2.0 dual-role device
- High-speed USB 2.0 on-the-go
- Four MultiMedia Card/Secure Digital/Secure Digital Input Output Interfaces (MMC™/ SD®/SDIO)
- PCI Express® 3.0 subsystems with two 5-Gbps lanes
  - One 2-lane Gen2-compliant port
  - or two 1-lane Gen2-compliant ports
- Dual Controller Area Network (DCAN) modules
  - CAN 2.0B protocol
- MIPI® CSI-2 camera serial interface
- Up to 186 General-Purpose I/O (GPIO) pins
- Device security features
  - Hardware crypto accelerators and DMA
  - Firewalls
  - JTAG lock
  - Secure keys
  - Secure ROM and boot
  - Customer programmable keys
- Power, reset, and clock management
- On-chip debug with CTools technology
- 28-nm CMOS technology
- 17 mm × 17 mm, 0.65-mm pitch, 538-pin BGA (CBD)

1.2 Applications

- Digital and analog radio
- Hybrid radio
- DSP audio amplifier
- Vehicle connectivity co-processor

1.3 Description

The DRA79x processor is offered in a 538-ball, 17×17-mm, 0.65-mm ball pitch (0.8mm spacing rules can be used on signals) with Via Channel™ Array (VCA) technology, ball grid array (BGA) package.

The architecture is designed to deliver high-performance concurrencies for automotive co-processor, hybrid radio and amplifier applications in a cost-effective solution, providing full scalability from the DRA75x ("Jacinto 6 EP" and "Jacinto 6 Ex"), DRA74x "Jacinto 6", DRA72x "Jacinto 6 Eco", and DRA71x "Jacinto 6 Entry" family of infotainment processors.

Programmability is provided by a single-core Arm Cortex-A15 RISC CPU with Neon™ extensions and a TI C66x VLIW floating-point DSP core. The Arm processor lets developers keep control functions separate from other algorithms programmed on the DSP and coprocessors, thus reducing the complexity of the system software.

Additionally, TI provides a complete set of development tools for the Arm, and DSP, including C compilers and a debugging interface for visibility into source code execution.

Cryptographic acceleration is available in all devices. All other supported security features, including support for secure boot, debug security and support for trusted execution environment are available on High-Security (HS) devices. For more information about HS devices, contact your TI representative.

The DRA79x Jacinto 6 RSP (Radio Sound Processor) device family is qualified according to the AEC-Q100 standard.

The device features a simplified power supply rail mapping which enables lower cost PMIC solutions.

Device Information(1)

PART NUMBER	PACKAGE	BODY SIZE
DRA790CBD	FCBGA (538)	17.0 mm × 17.0 mm
DRA791CBD	FCBGA (538)	17.0 mm × 17.0 mm
DRA793CBD	FCBGA (538)	17.0 mm × 17.0 mm
DRA797CBD	FCBGA (538)	17.0 mm × 17.0 mm

(1) For more information, see Section 9, Mechanical, Packaging, and Orderable Information.

1.4 Functional Block Diagram

Figure 1-1 is functional block diagram for the device.

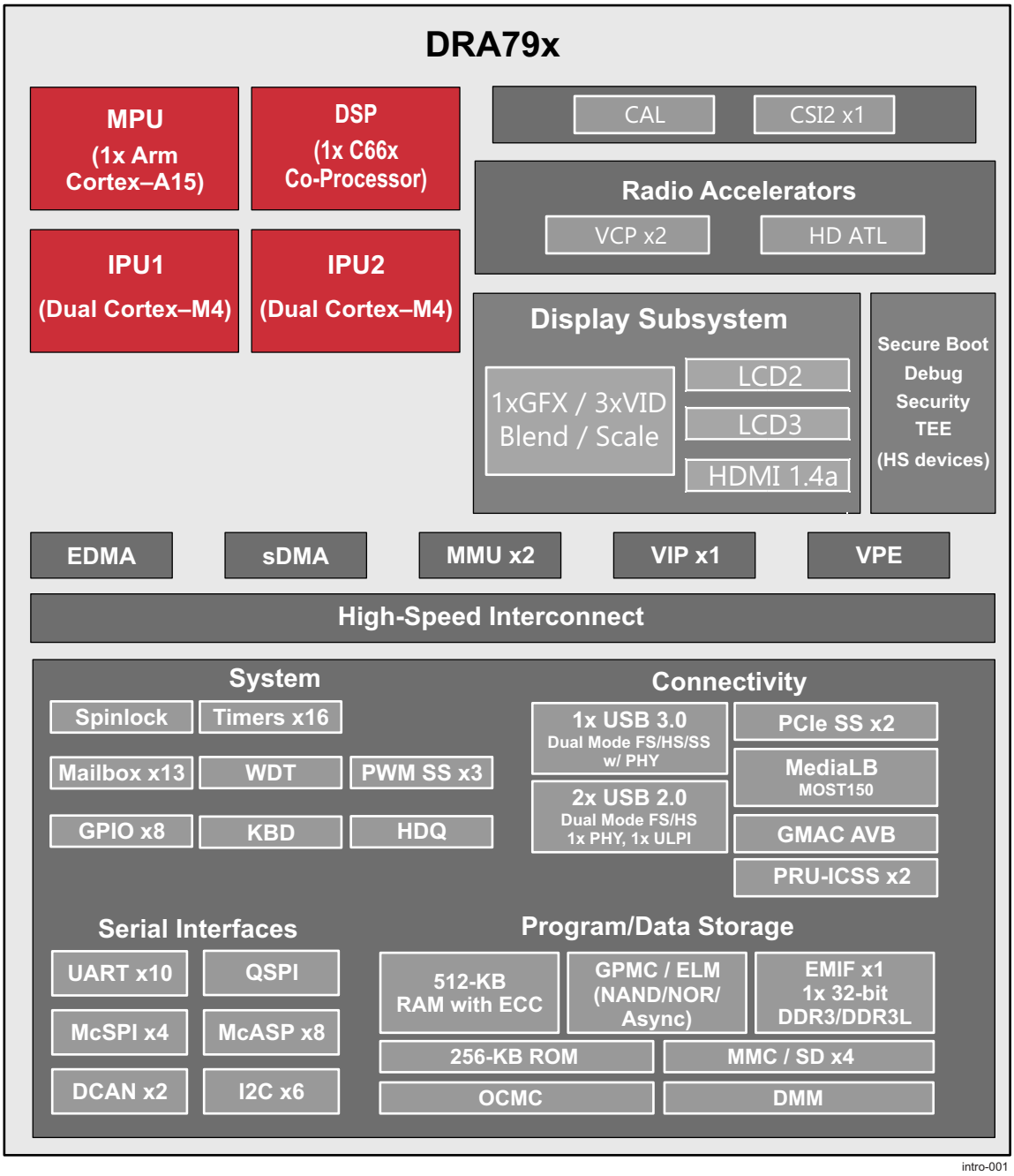


Figure 1-1. DRA79x Block Diagram

Table of Contents

2 Revision History

Changes from May 16, 2019 to November 15, 2019 (from E Revision (May 2019) to F Revision) **Page**

- Added reminders to disable unused pulls and RX pads in Section 4.2, *Pin Attributes* 10

3 Device Comparison

Table 3-1 shows a comparison between devices, highlighting the differences.

Table 3-1. Device Comparison[2]

FEATURES			DEVICE			
			DRA790	DRA791	DRA793	DRA797
Features						
CTRL_WKUP_STD_FUSE_DIE_ID_2 [31:24] Base PN register bitfield value[1][2]			128 (0x80)	129 (0x81)	131 (0x83)	135 (0x87)
Processors/Accelerators						
Speed Grades			A	B	D	H
Arm Single Cortex-A15 Microprocessor (MPU) Subsystem		MPU core 0	Yes			
C66x VLIW DSP		DSP1	Yes			
BitBLT 2D Hardware Acceleration Engine (BB2D)		BB2D	No			
Display Subsystem		VOUT1	No			
		VOUT2	Yes			
		VOUT3	Yes			
		HDMI	Yes			
Dual Arm Cortex-M4 Image Processing Unit (IPU)		IPU1	Yes			
		IPU2	Yes			
Image Video Accelarator (IVA)		IVA	No			
SGX544 Single-Core 3D Graphics Processing Unit (GPU)		GPU	No			
Video Input Port (VIP)	VIP1	vin1a	Yes			
		vin1b	Yes			
		vin2a	Yes			
		vin2b	Yes			
Video Processing Engine (VPE)		VPE	Yes			
Program/Data Storage						
On-Chip Shared Memory (RAM)		OCMC_RAM1	512KB			
General-Purpose Memory Controller (GPMC)		GPMC	Yes			
DDR3/DDR3L Memory Controller		EMIF1	up to 2GB			
Dynamic Memory Manager (DMM)		DMM	Yes			
Radio Support						
Audio Tracking Logic (ATL)		ATL	Yes			
Viterbi Coprocessor (VCP)		VCP1	Yes			
		VCP2	Yes			
Peripherals						
Controller Area Network (DCAN) Interface		DCAN1	Yes			
		DCAN2	Yes			
Enhanced DMA (EDMA)		EDMA	Yes			
System DMA (DMA_SYSTEM)		DMA_SYSTEM	Yes			
Ethernet Subsystem (Ethernet SS)		GMAC_SW[0]	MII, RMII, or RGMII			
		GMAC_SW[1]	MII, RMII, or RGMII			
General-Purpose I/O (GPIO)		GPIO	Up to 186			
Inter-Integrated Circuit Interface (I^2C)		I2C	6			
System Mailbox Module		MAILBOX	13			
Media Local Bus Subsystem (MLBSS)		MLB	Yes			

Table 3-1. Device Comparison[2] (continued)

FEATURES		DEVICE			
		DRA790	DRA791	DRA793	DRA797
Camera Adaptation Layer (CAL) Camera Serial Interface 2 (CSI2)	CSI2_0	1 CLK + 2 Data			
	CSI2_1	No			
Multichannel Audio Serial Port (McASP)	McASP1	16 serializers			
	McASP2	16 serializers			
	McASP3	4 serializers			
	McASP4	4 serializers			
	McASP5	4 serializers			
	McASP6	4 serializers			
	McASP7	4 serializers			
	McASP8	2 serializers			
MultiMedia Card/Secure Digital/Secure Digital Input Output Interface (MMC/SD/SDIO)	MMC1	1x UHSI 4b			
	MMC2	1x eMMC 8b			
	MMC3	1x SDIO 8b			
	MMC4	1x SDIO 4b			
PCI Express 3.0 Port with Integrated PHY	PCIe_SS1	Up to two lanes (second lane shared with PCIe_SS2 and USB1)			
	PCIe_SS2	Single lane (shared with PCIe_SS1 and USB1)			
Serial Advanced Technology Attachment (SATA)	SATA	No			
Real-Time Clock Subsystem (RTCSS)	RTCSS	No			
Programmable Real-Time Unit Subsystem and Industrial Communication Subsystem (PRU-ICSS)	PRU-ICSS1	No	No	No	Yes
	PRU-ICSS2	No	No	No	Yes
Multichannel Serial Peripheral Interface (McSPI)	McSPI	4			
HDQ1W	HDQ1W	Yes			
Quad SPI (QSPI)	QSPI	Yes			
Spinlock Module	SPINLOCK	Yes			
Keyboard Controller (KBD)	KBD	Yes			
Timers, General-Purpose	TIMERS GP	16			
Timer, Watchdog	WD TIMER	Yes			
Pulse-Width Modulation Subsystem (PWMSS)	PWMSS1	Yes			
	PWMSS2	Yes			
	PWMSS3	Yes			
Universal Asynchronous Receiver/Transmitter (UART)	UART	10			
Universal Serial Bus (USB3.0)	USB1 (Super-Speed, Dual-Role-Device [DRD])	Yes			
Universal Serial Bus (USB2.0)	USB2 (High-Speed, Dual-Role-Device [DRD], with embedded HS PHY)	Yes			
	USB3 (High-Speed, OTG2.0, with ULPI)	Yes			
	USB4 (High-Speed, OTG2.0, with ULPI)	No			

(1) For more details about the CTRL_WKUP_STD_FUSE_DIE_ID_2 register and Base PN bitfield, see the device TRM.

(2) X777 base part number with X speed grade indicator is the part number for the superset device. Software should constrain the features and speed used to match the intended production device. The Base PN register bitfield value is 0x4F.

3.1 Related Products

Automotive Processors

DRAx Infotainment SoCs The "Jacinto™ 6" family of infotainment processors (DRA7xx), paired with robust software and ecosystem offering bring unprecedented feature-rich, in-vehicle infotainment, instrument cluster and telematics features to the next generation automobiles.

4 Terminal Configuration and Functions

4.1 Pin Diagram

Figure 4-1 shows the ball locations for the 538 plastic ball grid array (PBGA) package and isused in conjunction with Table 4-1 through Table 4-31 to locate signal names and ball grid numbers.

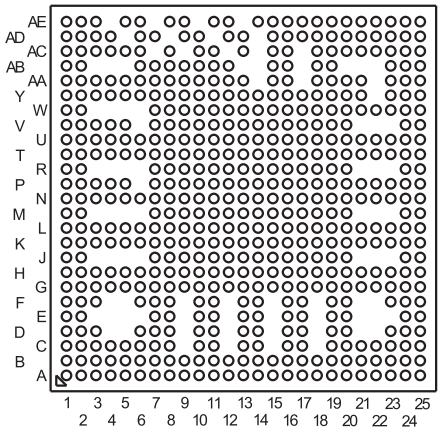


Figure 4-1. CBD S-PBGA-N538 Package (Bottom View)

NOTE

The following bottom balls are not pinned out: AE4 / AE7 / AE10 / AE13 / AD5 / AD8 / AD11 / AD14 / AC7 / AC9 / AC12 / AC14 / AC17 / AB3 / AB4 / AB5 / AB13 / AB14 / AB17 / AB20 / AB21 / AB22 / AA14 / AA17 / AA22 / Y22 / W3 / W4 / W5 / W6 / V6 / V21 / V22 / V23 / R3 / R4 / R5 / R6 / R21 / R22 / R23 / P6 / M3 / M4 / M5 / M6 / M21 / M22 / M23 / J3 / J4 / J5 / J6 / J21 / J22 / J23 / F4 / F5 / F9 / F12 / F15 / F18 / F21 / F22 / E3 / E4 / E5 / E6 / E9 / E12 / E15 / E18 / E21 / E22 / E23 / D4 / D5 / D9 / D12 / D15 / D18 / D21 / D22 / C9 / C12 / C15 / C18.

These balls do not exist on the package.

4.2 Pin Attributes

Table 4-1 describes the terminal characteristics and the signals multiplexed on each ball. The following list describes the table column headers:

1. **BALL NUMBER:**This column lists ball numbers on the bottom side associated with each signal on the bottom.
2. **BALL NAME:** This column lists mechanical name from package device (name is taken from muxmode 0).
3. **SIGNAL NAME:**This column lists names of signals multiplexed on each ball (also notice that the name of the ball is the signal name in muxmode 0).

NOTE

Table 4-1 does not take into account the subsystem multiplexing signals. Subsystem multiplexing signals are described in Section 4.3, *Signal Descriptions*.

NOTE

In driver off mode, the buffer is configured in high-impedance.

NOTE

In some cases Table 4-1 may present more than one signal name per muxmode for the same ball. First signal in the list is the dominant function as selected via CTRL_CORE_PAD_* register.

All other signals are virtual functions that present alternate multiplexing options. This virtual functions are controlled via CTRL_CORE_ALT_SELECT_MUX or CTRL_CORE_VIP_MUX_SELECT register. For more information on how to use this options, please refer to Device TRM, Chapter *Control Module*, Section *Pad Configuration Registers.*

4. **Support:** This column shows if the functionality is applicable for **DRA790 / DRA791 / DRA793** devices. Note that the Pin Attributes table presents the functionality of **DRA797** device. An empty box means "Yes".
5. **MUXMODE:** Multiplexing mode number:
   a. MUXMODE 0 is the primary mode; this means that when MUXMODE=0 is set, the function mapped on the pin corresponds to the name of the pin. The primary muxmode is not necessarily the default muxmode.

NOTE

The default mode is the mode at the release of the reset; also see the RESET REL. MUXMODE column.

   b. MUXMODE 1 through 15 are possible muxmodes for alternate functions. On each pin, some muxmodes are effectively used for alternate functions, while some muxmodes are not used. Only MUXMODE values which correspond to defined functions should be used.
   c. An empty box means Not Applicable.
6. **TYPE:** Signal type and direction:
   – I = Input
   – O = Output
   – IO = Input or Output
   – D = Open drain
   – DS = Differential Signaling
   – A = Analog
   – PWR = Power
   – GND = Ground
   – CAP = LDO Capacitor

NOTE

The RX buffer within the pad logic should be disabled on all pins that are not being used as an input. For more information, see the *Control Module / Control Module Functional Description / PAD Functional Multiplexing and Configuration* section in the device TRM.

7. **BALL RESET STATE:** The state of the terminal at power-on reset:
   – drive 0 (OFF): The buffer drives V_{OL} (pulldown or pullup resistor not activated)
   – drive 1 (OFF): The buffer drives V_{OH} (pulldown or pullup resistor not activated)
   – OFF: High-impedance
   – PD: High-impedance with an active pulldown resistor
   – PU: High-impedance with an active pullup resistor
   – An empty box means Not Applicable

NOTE

Designs that contain pullup or pulldown resistors, either on the board or in attached devices that oppose internal pullup or pulldown resistors, that are active while the device is held in reset, must not remain in reset for long periods of time.

8. **BALL RESET REL. STATE:** The state of the terminal at the deactivation of the rstoutn signal (also mapped to the PRCM SYS_WARM_OUT_RST signal)
   - drive 0 (OFF): The buffer drives V_{OL} (pulldown or pullup resistor not activated)
   - drive clk (OFF): The buffer drives a toggling clock (pulldown or pullup resistor not activated)
   - drive 1 (OFF): The buffer drives V_{OH} (pulldown or pullup resistor not activated)
   - OFF: High-impedance
   - PD: High-impedance with an active pulldown resistor
   - PU: High-impedance with an active pullup resistor
   - An empty box means Not Applicable

NOTE

For more information on the CORE_PWRON_RET_RST reset signal and its reset sources, see the Power, Reset, and Clock Management / PRCM Reset Management Functional Description section of the Device TRM.

9. **BALL RESET REL. MUXMODE:** This muxmode is automatically configured at the release of the rstoutn signal (also mapped to the PRCM SYS_WARM_OUT_RST signal).
   An empty box means Not Applicable.
10. **IO VOLTAGE VALUE**: This column describes the IO voltage value (the corresponding power supply).
    An empty box means Not Applicable.
11. **POWER:** The voltage supply that powers the terminal IO buffers.
    An empty box means Not Applicable.
12. **HYS:** Indicates if the input buffer is with hysteresis:
    - Yes: With hysteresis
    - No: Without hysteresis
    - An empty box: Not Applicable

NOTE

For more information, see the hysteresis values in Section 5.7, *Electrical Characteristics*.

13. **BUFFER TYPE:** Drive strength of the associated output buffer.
    An empty box means Not Applicable.

NOTE

For programmable buffer strength:
- The default value is given in Table 4-1.
- A note describes all possible values according to the selected muxmode.

14. **PULLUP / PULLDOWN TYPE:** Denotes the presence of an internal pullup or pulldown resistor. Pullup and pulldown resistors can be enabled or disabled via software.
    - PU: Internal pullup
    - PD: Internal pulldown
    - PU/PD: Internal pullup and pulldown
    - PUx/PDy: Programmable internal pullup and pulldown
    - PDy: Programmable internal pulldown
    - An empty box means No pull

NOTE

Internal pullup or pulldown resistors must be disabled when opposed by an external pullup or pulldown resistor on the board or within an attached device.

15. **DSIS:** The deselected input state (DSIS) indicates the state driven on the peripheral input (logic "0" or logic "1") when the peripheral pin function is not selected by any of the PINCNTLx registers.
    - 0: Logic 0 driven on the peripheral's input signal port.
    - 1: Logic 1 driven on the peripheral's input signal port.
    - blank: Pin state driven on the peripheral's input signal port.

NOTE

Configuring two pins to the same input signal is not supported as it can yield unexpected results. This can be easily prevented with the proper software configuration (Hi-Z mode is not an input signal).

NOTE

When a pad is set into a multiplexing mode which is not defined by pin multiplexing, that pad's behavior is undefined. This should be avoided.

NOTE

Some of the EMIF1 signals have an additional state change at the release of porz. The state that the signals change to at the release of porz is as follows:

drive 0 (OFF) for: ddr1_ck, ddr1_odt[0], ddr1_rst.

drive 1 (OFF) for: ddr1_casn, ddr1_rasn, ddr1_wen, ddr1_nck, ddr1_ba[2:0], ddr1_a[15:0], ddr1_csn[0], ddr1_cke, ddr1_dqm[3:0].

NOTE

Dual rank support is not available on this device, but signal names are retained for consistency with the DRA7xx family of devices.

Table 4-1. Pin Attributes[(1)]

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
F8	cap_vbbldo_dsp	cap_vbbldo_dsp			CAP									
T7	cap_vbbldo_gpu	cap_vbbldo_gpu			CAP									
G14	cap_vbbldo_iva	cap_vbbldo_iva			CAP									
F17	cap_vbbldo_mpu	cap_vbbldo_mpu			CAP									
U20	cap_vddram_core1	cap_vddram_core1			CAP									
K7	cap_vddram_core3	cap_vddram_core3			CAP									
G19	cap_vddram_core4	cap_vddram_core4			CAP									
L7	cap_vddram_dsp	cap_vddram_dsp			CAP									
V7	cap_vddram_gpu	cap_vddram_gpu			CAP									
G12	cap_vddram_iva	cap_vddram_iva			CAP									
G18	cap_vddram_mpu	cap_vddram_mpu			CAP									
AC1	csi2_0_dx0	csi2_0_dx0		0	I				1.8		Yes	LVCMOS CSI2	PU/PD	
AD1	csi2_0_dx1	csi2_0_dx1		0	I				1.8		Yes	LVCMOS CSI2	PU/PD	
AE2	csi2_0_dx2	csi2_0_dx2		0	I				1.8		Yes	LVCMOS CSI2	PU/PD	
AB2	csi2_0_dy0	csi2_0_dy0		0	I				1.8		Yes	LVCMOS CSI2	PU/PD	
AC2	csi2_0_dy1	csi2_0_dy1		0	I				1.8		Yes	LVCMOS CSI2	PU/PD	
AD2	csi2_0_dy2	csi2_0_dy2		0	I				1.8		Yes	LVCMOS CSI2	PU/PD	
H23	dcan1_rx	dcan1_rx		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart8_txd		2	O									
		mmc2_sdwp		3	I									0
		hdmi1_cec		6	IO									
		gpio1_15		14	IO									
		Driver off		15	I									
H22	dcan1_tx	dcan1_tx		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart8_rxd		2	I									1
		mmc2_sdcd		3	I									1
		hdmi1_hpd		6	IO									
		gpio1_14		14	IO									
		Driver off		15	I									
AC18	ddr1_a0	ddr1_a0		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AE19	ddr1_a1	ddr1_a1		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
AD19	ddr1_a2	ddr1_a2		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AB19	ddr1_a3	ddr1_a3		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AD20	ddr1_a4	ddr1_a4		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AE20	ddr1_a5	ddr1_a5		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AA18	ddr1_a6	ddr1_a6		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AA20	ddr1_a7	ddr1_a7		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
Y21	ddr1_a8	ddr1_a8		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AC20	ddr1_a9	ddr1_a9		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AA21	ddr1_a10	ddr1_a10		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AC21	ddr1_a11	ddr1_a11		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AC22	ddr1_a12	ddr1_a12		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AC15	ddr1_a13	ddr1_a13		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AB15	ddr1_a14	ddr1_a14		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AC16	ddr1_a15	ddr1_a15		0	O	PD	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AE16	ddr1_ba0	ddr1_ba0		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AA16	ddr1_ba1	ddr1_ba1		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AB16	ddr1_ba2	ddr1_ba2		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AD16	ddr1_casn	ddr1_casn		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AD21	ddr1_ck	ddr1_ck		0	O	PD	drive 0 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AB18	ddr1_cke	ddr1_cke		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AC19	ddr1_csn0	ddr1_csn0		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AA23	ddr1_d0	ddr1_d0		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AC24	ddr1_d1	ddr1_d1		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
AB24	ddr1_d2	ddr1_d2		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AD24	ddr1_d3	ddr1_d3		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AB23	ddr1_d4	ddr1_d4		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AC23	ddr1_d5	ddr1_d5		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AD23	ddr1_d6	ddr1_d6		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AE24	ddr1_d7	ddr1_d7		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AA24	ddr1_d8	ddr1_d8		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
W25	ddr1_d9	ddr1_d9		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
Y23	ddr1_d10	ddr1_d10		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AD25	ddr1_d11	ddr1_d11		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AC25	ddr1_d12	ddr1_d12		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AB25	ddr1_d13	ddr1_d13		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AA25	ddr1_d14	ddr1_d14		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
W24	ddr1_d15	ddr1_d15		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
W23	ddr1_d16	ddr1_d16		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
U25	ddr1_d17	ddr1_d17		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
U24	ddr1_d18	ddr1_d18		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
W21	ddr1_d19	ddr1_d19		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
T22	ddr1_d20	ddr1_d20		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
U22	ddr1_d21	ddr1_d21		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
U23	ddr1_d22	ddr1_d22		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
T21	ddr1_d23	ddr1_d23		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
T23	ddr1_d24	ddr1_d24		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
T25	ddr1_d25	ddr1_d25		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
T24	ddr1_d26	ddr1_d26		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
P21	ddr1_d27	ddr1_d27		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
N21	ddr1_d28	ddr1_d28		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
P22	ddr1_d29	ddr1_d29		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
P23	ddr1_d30	ddr1_d30		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
P24	ddr1_d31	ddr1_d31		0	IO	PD	PD		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AE23	ddr1_dqm0	ddr1_dqm0		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
W22	ddr1_dqm1	ddr1_dqm1		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
U21	ddr1_dqm2	ddr1_dqm2		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
P25	ddr1_dqm3	ddr1_dqm3		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AD22	ddr1_dqs0	ddr1_dqs0		0	IO	PD	PD		1.35/1.5	vdds_ddr1		LVCMOS DDR	PUx/PDy	
Y24	ddr1_dqs1	ddr1_dqs1		0	IO	PD	PD		1.35/1.5	vdds_ddr1		LVCMOS DDR	PUx/PDy	
V24	ddr1_dqs2	ddr1_dqs2		0	IO	PD	PD		1.35/1.5	vdds_ddr1		LVCMOS DDR	PUx/PDy	
R24	ddr1_dqs3	ddr1_dqs3		0	IO	PD	PD		1.35/1.5	vdds_ddr1		LVCMOS DDR	PUx/PDy	
AE22	ddr1_dqsn0	ddr1_dqsn0		0	IO	PU	PU		1.35/1.5	vdds_ddr1		LVCMOS DDR	PUx/PDy	
Y25	ddr1_dqsn1	ddr1_dqsn1		0	IO	PU	PU		1.35/1.5	vdds_ddr1		LVCMOS DDR	PUx/PDy	
V25	ddr1_dqsn2	ddr1_dqsn2		0	IO	PU	PU		1.35/1.5	vdds_ddr1		LVCMOS DDR	PUx/PDy	
R25	ddr1_dqsn3	ddr1_dqsn3		0	IO	PU	PU		1.35/1.5	vdds_ddr1		LVCMOS DDR	PUx/PDy	
AE21	ddr1_nck	ddr1_nck		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AD18	ddr1_odt0	ddr1_odt0		0	O	PD	drive 0 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AD17	ddr1_rasn	ddr1_rasn		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
AE17	ddr1_rst	ddr1_rst		0	O	PD	drive 0 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
Y20	ddr1_vref0	ddr1_vref0		0	PWR	OFF	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR		
AE18	ddr1_wen	ddr1_wen		0	O	PU	drive 1 (OFF)		1.35/1.5	vdds_ddr1	No	LVCMOS DDR	PUx/PDy	
C21	emu0	emu0		0	IO	PU	PU	0	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		gpio8_30		14	IO									
C22	emu1	emu1		0	IO	PU	PU	0	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		gpio8_31		14	IO									
E14	emu2	emu2		2	O	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
F14	emu3	emu3		2	O	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
F13	emu4	emu4		2	O	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
Y5	gpio6_10	gpio6_10		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	
		mdio_mclk		1	O									1
		i2c3_sda		2	IO									1
		usb3_ulpi_d7		3	IO									0
		vin2b_hsync1		4	I									
		vin1a_clk0		9	I									0
		ehrpwm2A		10	O									
		pr2_mii_mt1_clk	No	11	I									0
		pr2_pru0_gpi0	No	12	I									
		pr2_pru0_gpo0	No	13	O									
		gpio6_10		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
Y6	gpio6_11	gpio6_11		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	
		mdio_d		1	IO									1
		i2c3_scl		2	IO									1
		usb3_ulpi_d6		3	IO									0
		vin2b_vsync1		4	I									
		vin1a_de0		9	I									0
		ehrpwm2B		10	O									
		pr2_mii1_txen	No	11	O									
		pr2_pru0_gpi1	No	12	I									
		pr2_pru0_gpo1	No	13	O									
		gpio6_11		14	IO									
		Driver off		15	I									
H21	gpio6_14	gpio6_14		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		mcasp1_axr8		1	IO									0
		dcan2_tx		2	IO									1
		uart10_rxd		3	I									1
		i2c3_sda		9	IO									1
		timer1		10	IO									
		gpio6_14		14	IO									
		Driver off		15	I									
K22	gpio6_15	gpio6_15		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		mcasp1_axr9		1	IO									0
		dcan2_rx		2	IO									1
		uart10_txd		3	O									
		i2c3_scl		9	IO									1
		timer2		10	IO									
		gpio6_15		14	IO									
		Driver off		15	I									
K23	gpio6_16	gpio6_16		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		mcasp1_axr10		1	IO									0
		clkout1		9	O									
		timer3		10	IO									
		gpio6_16		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
M1	gpmc_a0	gpmc_a0		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_d16		2	I									0
		vout3_d16		3	O									
		vin1b_d0		6	I									0
		i2c4_scl		7	IO									1
		uart5_rxd		8	I									1
		gpio7_3 gpmc_a26 gpmc_a16		14	IO									
		Driver off		15	I									
M2	gpmc_a1	gpmc_a1		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_d17		2	I									0
		vout3_d17		3	O									
		vin1b_d1		6	I									0
		i2c4_sda		7	IO									1
		uart5_txd		8	O									
		gpio7_4		14	IO									
		Driver off		15	I									
L2	gpmc_a2	gpmc_a2		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_d18		2	I									0
		vout3_d18		3	O									
		vin1b_d2		6	I									0
		uart7_rxd		7	I									1
		uart5_ctsn		8	I									1
		gpio7_5		14	IO									
		Driver off		15	I									
L1	gpmc_a3	gpmc_a3		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_cs2		1	O									1
		vin1a_d19		2	I									0
		vout3_d19		3	O									
		vin1b_d3		6	I									0
		uart7_txd		7	O									
		uart5_rtsn		8	O									
		gpio7_6		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
K3	gpmc_a4	gpmc_a4		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_cs3		1	O									1
		vin1a_d20		2	I									0
		vout3_d20		3	O									
		vin1b_d4		6	I									0
		i2c5_scl		7	IO									1
		uart6_rxd		8	I									1
		gpio1_26		14	IO									
		Driver off		15	I									
K2	gpmc_a5	gpmc_a5		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_d21		2	I									0
		vout3_d21		3	O									
		vin1b_d5		6	I									0
		i2c5_sda		7	IO									1
		uart6_txd		8	O									
		gpio1_27		14	IO									
		Driver off		15	I									
J1	gpmc_a6	gpmc_a6		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_d22		2	I									0
		vout3_d22		3	O									
		vin1b_d6		6	I									0
		uart8_rxd		7	I									1
		uart6_ctsn		8	I									1
		gpio1_28		14	IO									
		Driver off		15	I									
K1	gpmc_a7	gpmc_a7		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_d23		2	I									0
		vout3_d23		3	O									
		vin1b_d7		6	I									0
		uart8_txd		7	O									
		uart6_rtsn		8	O									
		gpio1_29		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
K4	gpmc_a8	gpmc_a8		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_hsync0		2	I									0
		vout3_hsync		3	O									
		vin1b_hsync1		6	I									0
		timer12		7	IO									
		spi4_sclk		8	IO									0
		gpio1_30		14	IO									
		Driver off		15	I									
H1	gpmc_a9	gpmc_a9		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_vsync0		2	I									0
		vout3_vsync		3	O									
		vin1b_vsync1		6	I									0
		timer11		7	IO									
		spi4_d1		8	IO									0
		gpio1_31		14	IO									
		Driver off		15	I									
J2	gpmc_a10	gpmc_a10		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_de0		2	I									0
		vout3_de		3	O									
		vin1b_clk1		6	I									0
		timer10		7	IO									
		spi4_d0		8	IO									0
		gpio2_0		14	IO									
		Driver off		15	I									
L3	gpmc_a11	gpmc_a11		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		vin1a_fld0		2	I									0
		vout3_fld		3	O									
		vin1b_de1		6	I									0
		timer9		7	IO									
		spi4_cs0		8	IO									1
		gpio2_1		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
G1	gpmc_a12	gpmc_a12		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		gpmc_a0		5	O									
		vin1b_fld1		6	I									0
		timer8		7	IO									
		spi4_cs1		8	IO									1
		dma_evt1		9	I									0
		gpio2_2		14	IO									
		Driver off		15	I									
H3	gpmc_a13	gpmc_a13		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_rtclk		1	I									0
		timer7		7	IO									
		spi4_cs2		8	IO									1
		dma_evt2		9	I									0
		gpio2_3		14	IO									
		Driver off		15	I									
H4	gpmc_a14	gpmc_a14		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_d3		1	IO									0
		timer6		7	IO									
		spi4_cs3		8	IO									1
		gpio2_4		14	IO									
		Driver off		15	I									
K6	gpmc_a15	gpmc_a15		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_d2		1	IO									0
		timer5		7	IO									
		gpio2_5		14	IO									
		Driver off		15	I									
K5	gpmc_a16	gpmc_a16		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_d0		1	IO									0
		gpio2_6		14	IO									
		Driver off		15	I									
G2	gpmc_a17	gpmc_a17		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_d1		1	IO									0
		gpio2_7		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
F2	gpmc_a18	gpmc_a18		0	O	PD	PD	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_sclk		1	IO									
		gpio2_8		14	IO									
		Driver off		15	I									
A4[(9)]	gpmc_a19	gpmc_a19		0	O	PD	PD	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_dat4		1	IO									1
		gpmc_a13		2	O									
		vin2b_d0		6	I									0
		gpio2_9		14	IO									
		Driver off		15	I									
E7[(9)]	gpmc_a20	gpmc_a20		0	O	PD	PD	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_dat5		1	IO									1
		gpmc_a14		2	O									
		vin2b_d1		6	I									0
		gpio2_10		14	IO									
		Driver off		15	I									
D6[(9)]	gpmc_a21	gpmc_a21		0	O	PD	PD	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_dat6		1	IO									1
		gpmc_a15		2	O									
		vin2b_d2		6	I									0
		gpio2_11		14	IO									
		Driver off		15	I									
C5[(9)]	gpmc_a22	gpmc_a22		0	O	PD	PD	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_dat7		1	IO									1
		gpmc_a16		2	O									
		vin2b_d3		6	I									0
		gpio2_12		14	IO									
		Driver off		15	I									
B5	gpmc_a23	gpmc_a23		0	O	PD	PD	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_clk		1	IO									1
		gpmc_a17		2	O									
		vin2b_d4		6	I									0
		gpio2_13		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[1] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
D7[9]	gpmc_a24	gpmc_a24		0	O	PD	PD	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_dat0		1	IO									1
		gpmc_a18		2	O									
		vin2b_d5		6	I									0
		gpio2_14		14	IO									
		Driver off		15	I									
C6[9]	gpmc_a25	gpmc_a25		0	O	PD	PD	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_dat1		1	IO									1
		gpmc_a19		2	O									
		vin2b_d6		6	I									0
		gpio2_15		14	IO									
		Driver off		15	I									
A5[9]	gpmc_a26	gpmc_a26		0	O	PD	PD	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_dat2		1	IO									1
		gpmc_a20		2	O									
		vin2b_d7		6	I									0
		gpio2_16		14	IO									
		Driver off		15	I									
B6[9]	gpmc_a27	gpmc_a27		0	O	PD	PD	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_dat3		1	IO									1
		gpmc_a21		2	O									
		vin2b_hsync1		6	I									
		gpio2_17		14	IO									
		Driver off		15	I									
F1	gpmc_ad0	gpmc_ad0		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d0		2	I									0
		vout3_d0		3	O									
		gpio1_6		14	IO									
		sysboot0		15	I									
E2	gpmc_ad1	gpmc_ad1		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d1		2	I									0
		vout3_d1		3	O									
		gpio1_7		14	IO									
		sysboot1		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
E1	gpmc_ad2	gpmc_ad2		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d2		2	I									0
		vout3_d2		3	O									
		gpio1_8		14	IO									
		sysboot2		15	I									
C1	gpmc_ad3	gpmc_ad3		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d3		2	I									0
		vout3_d3		3	O									
		gpio1_9		14	IO									
		sysboot3		15	I									
D1	gpmc_ad4	gpmc_ad4		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d4		2	I									0
		vout3_d4		3	O									
		gpio1_10		14	IO									
		sysboot4		15	I									
D2	gpmc_ad5	gpmc_ad5		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d5		2	I									0
		vout3_d5		3	O									
		gpio1_11		14	IO									
		sysboot5		15	I									
B1	gpmc_ad6	gpmc_ad6		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d6		2	I									0
		vout3_d6		3	O									
		gpio1_12		14	IO									
		sysboot6		15	I									
B2	gpmc_ad7	gpmc_ad7		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d7		2	I									0
		vout3_d7		3	O									
		gpio1_13		14	IO									
		sysboot7		15	I									
C2	gpmc_ad8	gpmc_ad8		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d8		2	I									0
		vout3_d8		3	O									
		gpio7_18		14	IO									
		sysboot8		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
D3	gpmc_ad9	gpmc_ad9		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d9		2	I									0
		vout3_d9		3	O									
		gpio7_19		14	IO									
		sysboot9		15	I									
A2	gpmc_ad10	gpmc_ad10		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d10		2	I									0
		vout3_d10		3	O									
		gpio7_28		14	IO									
		sysboot10		15	I									
B3	gpmc_ad11	gpmc_ad11		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d11		2	I									0
		vout3_d11		3	O									
		gpio7_29		14	IO									
		sysboot11		15	I									
C3	gpmc_ad12	gpmc_ad12		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d12		2	I									0
		vout3_d12		3	O									
		gpio1_18		14	IO									
		sysboot12		15	I									
C4	gpmc_ad13	gpmc_ad13		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d13		2	I									0
		vout3_d13		3	O									
		gpio1_19		14	IO									
		sysboot13		15	I									
A3	gpmc_ad14	gpmc_ad14		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d14		2	I									0
		vout3_d14		3	O									
		gpio1_20		14	IO									
		sysboot14		15	I									
B4	gpmc_ad15	gpmc_ad15		0	IO	OFF	OFF	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d15		2	I									0
		vout3_d15		3	O									
		gpio1_21		14	IO									
		sysboot15		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
H5	gpmc_advn_ale	gpmc_advn_ale		0	O	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		gpmc_cs6		1	O									
		clkout2		2	O									
		gpmc_wait1		3	I									1
		gpmc_a2		5	O									
		gpmc_a23		6	O									
		timer3		7	IO									
		i2c3_sda		8	IO									1
		dma_evt2		9	I									0
		gpio2_23 gpmc_a19		14	IO									
		Driver off		15	I									
H2	gpmc_ben0	gpmc_ben0		0	O	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		gpmc_cs4		1	O									
		vin2b_de1		6	I									
		timer2		7	IO									
		dma_evt3		9	I									0
		gpio2_26 gpmc_a21		14	IO									
		Driver off		15	I									
H6	gpmc_ben1	gpmc_ben1		0	O	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		gpmc_cs5		1	O									
		vin2b_clk1		4	I									
		gpmc_a3		5	O									
		vin2b_fld1		6	I									
		timer1		7	IO									
		dma_evt4		9	I									0
		gpio2_27 gpmc_a22		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
L4	gpmc_clk	gpmc_clk		0	IO	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	0
		gpmc_cs7		1	O									
		clkout1		2	O									
		gpmc_wait1		3	I									1
		vin2b_clk1		6	I									
		timer4		7	IO									
		i2c3_scl		8	IO									1
		dma_evt1		9	I									0
		gpio2_22 gpmc_a20		14	IO									
		Driver off		15	I									
F3	gpmc_cs0	gpmc_cs0		0	O	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		gpio2_19		14	IO									
		Driver off		15	I									
A6	gpmc_cs1	gpmc_cs1		0	O	PU	PU	15	1.8/3.3	vddshv11	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc2_cmd		1	IO									1
		gpmc_a22		2	O									
		vin2b_vsync1		6	I									
		gpio2_18		14	IO									
		Driver off		15	I									
G4	gpmc_cs2	gpmc_cs2		0	O	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_cs0		1	IO									1
		gpio2_20 gpmc_a23 gpmc_a13		14	IO									
		Driver off		15	I									
G3	gpmc_cs3	gpmc_cs3		0	O	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		qspi1_cs1		1	O									1
		vin1a_clk0		2	I									0
		vout3_clk		3	O									
		gpmc_a1		5	O									
		gpio2_21 gpmc_a24 gpmc_a14		14	IO									
		Driver off		15	I									
G5	gpmc_oen_ren	gpmc_oen_ren		0	O	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		gpio2_24		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
F6	gpmc_wait0	gpmc_wait0		0	I	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	1
		gpio2_28 gpmc_a25 gpmc_a15		14	IO									
		Driver off		15	I									
G6	gpmc_wen	gpmc_wen		0	O	PU	PU	15	1.8/3.3	vddshv10	Yes	Dual Voltage LVCMOS	PU/PD	
		gpio2_25		14	IO									
		Driver off		15	I									
AE9	hdmi1_clockx	hdmi1_clockx		0	O				1.8	vdda_hdmi		HDMIPHY	Pdy	
AD10	hdmi1_clocky	hdmi1_clocky		0	O				1.8	vdda_hdmi		HDMIPHY	Pdy	
AE11	hdmi1_data0x	hdmi1_data0x		0	O				1.8	vdda_hdmi		HDMIPHY	Pdy	
AD12	hdmi1_data0y	hdmi1_data0y		0	O				1.8	vdda_hdmi		HDMIPHY	Pdy	
AE12	hdmi1_data1x	hdmi1_data1x		0	O				1.8	vdda_hdmi		HDMIPHY	Pdy	
AD13	hdmi1_data1y	hdmi1_data1y		0	O				1.8	vdda_hdmi		HDMIPHY	Pdy	
AE14	hdmi1_data2x	hdmi1_data2x		0	O				1.8	vdda_hdmi		HDMIPHY	Pdy	
AD15	hdmi1_data2y	hdmi1_data2y		0	O				1.8	vdda_hdmi		HDMIPHY	Pdy	
G22	i2c1_scl	i2c1_scl		0	IO				1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS I2C	PU/PD	
		Driver off		15	I									
G23	i2c1_sda	i2c1_sda		0	IO				1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS I2C	PU/PD	
		Driver off		15	I									
G21	i2c2_scl	i2c2_scl		0	IO			15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS I2C	PU/PD	1
		hdmi1_ddc_sda		1	IO									
		Driver off		15	I									
F23	i2c2_sda	i2c2_sda		0	IO			15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS I2C	PU/PD	1
		hdmi1_ddc_scl		1	IO									
		Driver off		15	I									
AB9	ljcb_clkn	ljcb_clkn		0	IO				1.8	vdda_pcie		LJCB		
AC8	ljcb_clkp	ljcb_clkp		0	IO				1.8	vdda_pcie		LJCB		
D16	mcasp1_aclkr	mcasp1_aclkr		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp7_axr2		1	IO									0
		i2c4_sda		10	IO									1
		gpio5_0		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
C16	mcasp1_aclkx	mcasp1_aclkx		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_fld0		7	I									0
		i2c3_sda		10	IO									1
		pr2_mdio_mdclk	No	11	O									
		pr2_pru1_gpi7	No	12	I									
		pr2_pru1_gpo7	No	13	O									
		gpio7_31		14	IO									
		Driver off		15	I									
D14	mcasp1_axr0	mcasp1_axr0		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		uart6_rxd		3	I									1
		vin1a_vsync0		7	I									0
		i2c5_sda		10	IO									1
		pr2_mii0_rxer	No	11	I									0
		pr2_pru1_gpi8	No	12	I									
		pr2_pru1_gpo8	No	13	O									
		gpio5_2		14	IO									
		Driver off		15	I									
B14	mcasp1_axr1	mcasp1_axr1		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		uart6_txd		3	O									
		vin1a_hsync0		7	I									0
		i2c5_scl		10	IO									1
		pr2_mii_mt0_clk	No	11	I									0
		pr2_pru1_gpi9	No	12	I									
		pr2_pru1_gpo9	No	13	O									
		gpio5_3		14	IO									
		Driver off		15	I									
C14	mcasp1_axr2	mcasp1_axr2		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp6_axr2		1	IO									0
		uart6_ctsn		3	I									1
		gpio5_4		14	IO									
		Driver off		15	I									
B15	mcasp1_axr3	mcasp1_axr3		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp6_axr3		1	IO									0
		uart6_rtsn		3	O									
		gpio5_5		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
A15	mcasp1_axr4	mcasp1_axr4		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp4_axr2		1	IO									0
		gpio5_6		14	IO									
		Driver off		15	I									
A14	mcasp1_axr5	mcasp1_axr5		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp4_axr3		1	IO									0
		gpio5_7		14	IO									
		Driver off		15	I									
A17	mcasp1_axr6	mcasp1_axr6		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp5_axr2		1	IO									0
		gpio5_8		14	IO									
		Driver off		15	I									
A16	mcasp1_axr7	mcasp1_axr7		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp5_axr3		1	IO									0
		timer4		10	IO									
		gpio5_9		14	IO									
		Driver off		15	I									
A18	mcasp1_axr8	mcasp1_axr8		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp6_axr0		1	IO									0
		spi3_sclk		3	IO									0
		vin1a_d15		7	I									0
		timer5		10	IO									
		pr2_mii0_txen	No	11	O									
		pr2_pru1_gpi10	No	12	I									
		pr2_pru1_gpo10	No	13	O									
		gpio5_10		14	IO									
		Driver off		15	I									
B17	mcasp1_axr9	mcasp1_axr9		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp6_axr1		1	IO									0
		spi3_d1		3	IO									0
		vin1a_d14		7	I									0
		timer6		10	IO									
		pr2_mii0_txd3	No	11	O									
		pr2_pru1_gpi11	No	12	I									
		pr2_pru1_gpo11	No	13	O									
		gpio5_11		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
B16	mcasp1_axr10	mcasp1_axr10		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp6_aclkx		1	IO									0
		mcasp6_aclkr		2	IO									
		spi3_d0		3	IO									0
		vin1a_d13		7	I									0
		timer7		10	IO									
		pr2_mii0_txd2	No	11	O									
		pr2_pru1_gpi12	No	12	I									
		pr2_pru1_gpo12	No	13	O									
		gpio5_12		14	IO									
		Driver off		15	I									
B18	mcasp1_axr11	mcasp1_axr11		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp6_fsx		1	IO									0
		mcasp6_fsr		2	IO									
		spi3_cs0		3	IO									1
		vin1a_d12		7	I									0
		timer8		10	IO									
		pr2_mii0_txd1	No	11	O									
		pr2_pru1_gpi13	No	12	I									
		pr2_pru1_gpo13	No	13	O									
		gpio4_17		14	IO									
		Driver off		15	I									
A19	mcasp1_axr12	mcasp1_axr12		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp7_axr0		1	IO									0
		spi3_cs1		3	IO									1
		vin1a_d11		7	I									0
		timer9		10	IO									
		pr2_mii0_txd0	No	11	O									
		pr2_pru1_gpi14	No	12	I									
		pr2_pru1_gpo14	No	13	O									
		gpio4_18		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
E17	mcasp1_axr13	mcasp1_axr13		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp7_axr1		1	IO									0
		vin1a_d10		7	I									0
		timer10		10	IO									
		pr2_mii_mr0_clk	No	11	I									0
		pr2_pru1_gpi15	No	12	I									
		pr2_pru1_gpo15	No	13	O									
		gpio6_4		14	IO									
		Driver off		15	I									
E16	mcasp1_axr14	mcasp1_axr14		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp7_aclkx		1	IO									0
		mcasp7_aclkr		2	IO									
		vin1a_d9		7	I									0
		timer11		10	IO									
		pr2_mii0_rxdv	No	11	I									0
		pr2_pru1_gpi16	No	12	I									
		pr2_pru1_gpo16	No	13	O									
		gpio6_5		14	IO									
		Driver off		15	I									
F16	mcasp1_axr15	mcasp1_axr15		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp7_fsx		1	IO									0
		mcasp7_fsr		2	IO									
		vin1a_d8		7	I									0
		timer12		10	IO									
		pr2_mii0_rxd3	No	11	I									0
		pr2_pru0_gpi20	No	12	I									
		pr2_pru0_gpo20	No	13	O									
		gpio6_6		14	IO									
		Driver off		15	I									
D17	mcasp1_fsr	mcasp1_fsr		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp7_axr3		1	IO									0
		i2c4_scl		10	IO									1
		gpio5_1		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
C17	mcasp1_fsx	mcasp1_fsx		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_de0		7	I									0
		i2c3_scl		10	IO									1
		pr2_mdio_data	No	11	IO									1
		gpio7_30		14	IO									
		Driver off		15	I									
E19	mcasp2_aclkx	mcasp2_aclkx		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d7		7	I									0
		pr2_mii0_rxd2	No	11	I									0
		pr2_pru0_gpi18	No	12	I									
		pr2_pru0_gpo18	No	13	O									
		Driver off		15	I									
A20	mcasp2_axr0	mcasp2_axr0		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		Driver off		15	I									
B19	mcasp2_axr1	mcasp2_axr1		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		Driver off		15	I									
A21	mcasp2_axr2	mcasp2_axr2		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp3_axr2		1	IO									0
		vin1a_d5		7	I									0
		pr2_mii0_rxd0	No	11	I									0
		pr2_pru0_gpi16	No	12	I									
		pr2_pru0_gpo16	No	13	O									
		gpio6_8		14	IO									
		Driver off		15	I									
B21	mcasp2_axr3	mcasp2_axr3		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp3_axr3		1	IO									0
		vin1a_d4		7	I									0
		pr2_mii0_rxlink	No	11	I									0
		pr2_pru0_gpi17	No	12	I									
		pr2_pru0_gpo17	No	13	O									
		gpio6_9		14	IO									
		Driver off		15	I									
B20	mcasp2_axr4	mcasp2_axr4		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp8_axr0		1	IO									0
		gpio1_4		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
C19	mcasp2_axr5	mcasp2_axr5		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp8_axr1		1	IO									0
		gpio6_7		14	IO									
		Driver off		15	I									
D20	mcasp2_axr6	mcasp2_axr6		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp8_aclkx		1	IO									0
		mcasp8_aclkr		2	IO									
		gpio2_29		14	IO									
		Driver off		15	I									
C20	mcasp2_axr7	mcasp2_axr7		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp8_fsx		1	IO									0
		mcasp8_fsr		2	IO									
		gpio1_5		14	IO									
		Driver off		15	I									
D19	mcasp2_fsx	mcasp2_fsx		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin1a_d6		7	I									0
		pr2_mii0_rxd1	No	11	I									0
		pr2_pru0_gpi19	No	12	I									
		pr2_pru0_gpo19	No	13	O									
		Driver off		15	I									
A22	mcasp3_aclkx	mcasp3_aclkx		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp3_aclkr		1	IO									
		mcasp2_axr12		2	IO									0
		uart7_rxd		3	I									1
		vin1a_d3		7	I									0
		pr2_mii0_crs	No	11	I									0
		pr2_pru0_gpi12	No	12	I									
		pr2_pru0_gpo12	No	13	O									
		gpio5_13		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
B22	mcasp3_axr0	mcasp3_axr0		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp2_axr14		2	IO									0
		uart7_ctsn		3	I									1
		uart5_rxd		4	I									1
		vin1a_d1		7	I									0
		pr2_mii1_rxer	No	11	I									0
		pr2_pru0_gpi14	No	12	I									
		pr2_pru0_gpo14	No	13	O									
		Driver off		15	I									
B23	mcasp3_axr1	mcasp3_axr1		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp2_axr15		2	IO									0
		uart7_rtsn		3	O									
		uart5_txd		4	O									
		vin1a_d0		7	I									0
		pr2_mii1_rxlink	No	11	I									0
		pr2_pru0_gpi15	No	12	I									
		pr2_pru0_gpo15	No	13	O									
		Driver off		15	I									
A23	mcasp3_fsx	mcasp3_fsx		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp3_fsr		1	IO									
		mcasp2_axr13		2	IO									0
		uart7_txd		3	O									
		vin1a_d2		7	I									0
		pr2_mii0_col	No	11	I									0
		pr2_pru0_gpi13	No	12	I									
		pr2_pru0_gpo13	No	13	O									
		gpio5_14		14	IO									
		Driver off		15	I									
C23	mcasp4_aclkx	mcasp4_aclkx		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp4_aclkr		1	IO									
		spi3_sclk		2	IO									0
		uart8_rxd		3	I									1
		i2c4_sda		4	IO									1
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
A24	mcasp4_axr0	mcasp4_axr0		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		spi3_d0		2	IO									0
		uart8_ctsn		3	I									1
		uart4_rxd		4	I									1
		i2c6_scl[(10)]		14	IO									
		Driver off		15	I									
D23	mcasp4_axr1	mcasp4_axr1		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		spi3_cs0		2	IO									1
		uart8_rtsn		3	O									
		uart4_txd		4	O									
		pr2_pru1_gpi0	No	12	I									
		pr2_pru1_gpo0	No	13	O									
		i2c6_sda[(10)]		14	IO									
		Driver off		15	I									
B25	mcasp4_fsx	mcasp4_fsx		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp4_fsr		1	IO									
		spi3_d1		2	IO									0
		uart8_txd		3	O									
		i2c4_scl		4	IO									1
		Driver off		15	I									
AC3	mcasp5_aclkx	mcasp5_aclkx		0	IO	PD	PD	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp5_aclkr		1	IO									
		spi4_sclk		2	IO									0
		uart9_rxd		3	I									1
		i2c5_sda		4	IO									1
		mlb_clk		5	I									1
		pr2_pru1_gpi1	No	12	I									
		pr2_pru1_gpo1	No	13	O									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
AA5	mcasp5_axr0	mcasp5_axr0		0	IO	PD	PD	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	0
		spi4_d0		2	IO									0
		uart9_ctsn		3	I									1
		uart3_rxd		4	I									1
		mlb_sig		5	IO									1
		pr2_mdio_mdclk	No	11	O									
		pr2_pru1_gpi3	No	12	I									
		pr2_pru1_gpo3	No	13	O									
		Driver off		15	I									
AC4	mcasp5_axr1	mcasp5_axr1		0	IO	PD	PD	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	0
		spi4_cs0		2	IO									1
		uart9_rtsn		3	O									
		uart3_txd		4	O									
		mlb_dat		5	IO									1
		pr2_mdio_data	No	11	IO									1
		pr2_pru1_gpi4	No	12	I									
		pr2_pru1_gpo4	No	13	O									
		Driver off		15	I									
U6	mcasp5_fsx	mcasp5_fsx		0	IO	PD	PD	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	0
		mcasp5_fsr		1	IO									
		spi4_d1		2	IO									0
		uart9_txd		3	O									
		i2c5_scl		4	IO									1
		pr2_pru1_gpi2	No	12	I									
		pr2_pru1_gpo2	No	13	O									
		Driver off		15	I									
L6	mdio_d	mdio_d		0	IO	PU	PU	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart3_ctsn		1	I									1
		mii0_txer		3	O									0
		vin2a_d0		4	I									0
		vin1b_d0		5	I									0
		pr1_mii0_rxlink	No	11	I									0
		pr2_pru1_gpi1	No	12	I									
		pr2_pru1_gpo1	No	13	O									
		gpio5_16		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
L5	mdio_mclk	mdio_mclk		0	O	PU	PU	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart3_rtsn		1	O									
		mii0_col		3	I									0
		vin2a_clk0		4	I									
		vin1b_clk1		5	I									0
		pr1_mii0_col	No	11	I									0
		pr2_pru1_gpi0	No	12	I									
		pr2_pru1_gpo0	No	13	O									
		gpio5_15		14	IO									
		Driver off		15	I									
U1	mlbp_clk_n	mlbp_clk_n		0	I					vdds_mlbp	No	BMLB18		
U2	mlbp_clk_p	mlbp_clk_p		0	I					vdds_mlbp	No	BMLB18		
T1	mlbp_dat_n	mlbp_dat_n		0	IO	OFF	OFF			vdds_mlbp	No	BMLB18		
T2	mlbp_dat_p	mlbp_dat_p		0	IO	OFF	OFF			vdds_mlbp	No	BMLB18		
U4	mlbp_sig_n	mlbp_sig_n		0	IO	OFF	OFF			vdds_mlbp	No	BMLB18		
T3	mlbp_sig_p	mlbp_sig_p		0	IO	OFF	OFF			vdds_mlbp	No	BMLB18		
U3	mmc1_clk	mmc1_clk		0	IO	PU	PU	15	1.8/3.3	vddshv8	Yes	SDIO2KV1833	Pux/PDy	1
		gpio6_21		14	IO									
		Driver off		15	I									
V4	mmc1_cmd	mmc1_cmd		0	IO	PU	PU	15	1.8/3.3	vddshv8	Yes	SDIO2KV1833	Pux/PDy	1
		gpio6_22		14	IO									
		Driver off		15	I									
V3	mmc1_dat0	mmc1_dat0		0	IO	PU	PU	15	1.8/3.3	vddshv8	Yes	SDIO2KV1833	Pux/PDy	1
		gpio6_23		14	IO									
		Driver off		15	I									
V2	mmc1_dat1	mmc1_dat1		0	IO	PU	PU	15	1.8/3.3	vddshv8	Yes	SDIO2KV1833	Pux/PDy	1
		gpio6_24		14	IO									
		Driver off		15	I									
W1	mmc1_dat2	mmc1_dat2		0	IO	PU	PU	15	1.8/3.3	vddshv8	Yes	SDIO2KV1833	Pux/PDy	1
		gpio6_25		14	IO									
		Driver off		15	I									
V1	mmc1_dat3	mmc1_dat3		0	IO	PU	PU	15	1.8/3.3	vddshv8	Yes	SDIO2KV1833	Pux/PDy	1
		gpio6_26		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
U5	mmc1_sdcd	mmc1_sdcd		0	I	PU	PU	15	1.8/3.3	vddshv8	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart6_rxd		3	I									1
		i2c4_sda		4	IO									1
		gpio6_27		14	IO									
		Driver off		15	I									
V5	mmc1_sdwp	mmc1_sdwp		0	I	PD	PD	15	1.8/3.3	vddshv8	Yes	Dual Voltage LVCMOS	PU/PD	0
		uart6_txd		3	O									
		i2c4_scl		4	IO									1
		gpio6_28		14	IO									
		Driver off		15	I									
Y2	mmc3_clk	mmc3_clk		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		usb3_ulpi_d5		3	IO									0
		vin2b_d7		4	I									0
		vin1a_d7		9	I									0
		ehrpwm2_tripzone_input		10	IO									0
		pr2_mii1_txd3	No	11	O									
		pr2_pru0_gpi2	No	12	I									
		pr2_pru0_gpo2	No	13	O									
		gpio6_29		14	IO									
		Driver off		15	I									
Y1	mmc3_cmd	mmc3_cmd		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi3_sclk		1	IO									0
		usb3_ulpi_d4		3	IO									0
		vin2b_d6		4	I									0
		vin1a_d6		9	I									0
		eCAP2_in_PWM2_out		10	IO									0
		pr2_mii1_txd2	No	11	O									
		pr2_pru0_gpi3	No	12	I									
		pr2_pru0_gpo3	No	13	O									
		gpio6_30		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
Y4	mmc3_dat0	mmc3_dat0		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi3_d1		1	IO									0
		uart5_rxd		2	I									1
		usb3_ulpi_d3		3	IO									0
		vin2b_d5		4	I									0
		vin1a_d5		9	I									0
		eQEP3A_in		10	I									0
		pr2_mii1_txd1	No	11	O									
		pr2_pru0_gpi4	No	12	I									
		pr2_pru0_gpo4	No	13	O									
		gpio6_31		14	IO									
		Driver off		15	I									
AA2	mmc3_dat1	mmc3_dat1		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi3_d0		1	IO									0
		uart5_txd		2	O									
		usb3_ulpi_d2		3	IO									0
		vin2b_d4		4	I									0
		vin1a_d4		9	I									0
		eQEP3B_in		10	I									0
		pr2_mii1_txd0	No	11	O									
		pr2_pru0_gpi5	No	12	I									
		pr2_pru0_gpo5	No	13	O									
		gpio7_0		14	IO									
		Driver off		15	I									
AA3	mmc3_dat2	mmc3_dat2		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi3_cs0		1	IO									1
		uart5_ctsn		2	I									1
		usb3_ulpi_d1		3	IO									0
		vin2b_d3		4	I									0
		vin1a_d3		9	I									0
		eQEP3_index		10	IO									0
		pr2_mii_mr1_clk	No	11	I									0
		pr2_pru0_gpi6	No	12	I									
		pr2_pru0_gpo6	No	13	O									
		gpio7_1		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
W2	mmc3_dat3	mmc3_dat3		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi3_cs1		1	IO									1
		uart5_rtsn		2	O									
		usb3_ulpi_d0		3	IO									0
		vin2b_d2		4	I									0
		vin1a_d2		9	I									0
		eQEP3_strobe		10	IO									0
		pr2_mii1_rxdv	No	11	I									0
		pr2_pru0_gpi7	No	12	I									
		pr2_pru0_gpo7	No	13	O									
		gpio7_2		14	IO									
		Driver off		15	I									
Y3	mmc3_dat4	mmc3_dat4		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi4_sclk		1	IO									0
		uart10_rxd		2	I									1
		usb3_ulpi_nxt		3	I									0
		vin2b_d1		4	I									0
		vin1a_d1		9	I									0
		ehrpwm3A		10	O									
		pr2_mii1_rxd3	No	11	I									0
		pr2_pru0_gpi8	No	12	I									
		pr2_pru0_gpo8	No	13	O									
		gpio1_22		14	IO									
		Driver off		15	I									
AA1	mmc3_dat5	mmc3_dat5		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi4_d1		1	IO									0
		uart10_txd		2	O									
		usb3_ulpi_dir		3	I									0
		vin2b_d0		4	I									0
		vin1a_d0		9	I									0
		ehrpwm3B		10	O									
		pr2_mii1_rxd2	No	11	I									0
		pr2_pru0_gpi9	No	12	I									
		pr2_pru0_gpo9	No	13	O									
		gpio1_23		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
AA4	mmc3_dat6	mmc3_dat6		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi4_d0		1	IO									0
		uart10_ctsn		2	I									1
		usb3_ulpi_stp		3	O									
		vin2b_de1		4	I									
		vin1a_hsync0		9	I									0
		ehrpwm3_tripzone_input		10	IO									0
		pr2_mii1_rxd1	No	11	I									0
		pr2_pru0_gpi10	No	12	I									
		pr2_pru0_gpo10	No	13	O									
		gpio1_24		14	IO									
		Driver off		15	I									
AB1	mmc3_dat7	mmc3_dat7		0	IO	PU	PU	15	1.8/3.3	vddshv7	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi4_cs0		1	IO									1
		uart10_rtsn		2	O									
		usb3_ulpi_clk		3	I									0
		vin2b_clk1		4	I									
		vin1a_vsync0		9	I									0
		eCAP3_in_PWM3_out		10	IO									0
		pr2_mii1_rxd0	No	11	I									0
		pr2_pru0_gpi11	No	12	I									
		pr2_pru0_gpo11	No	13	O									
		gpio1_25		14	IO									
		Driver off		15	I									
L24	nmin_dsp	nmin_dsp		0	I	PD	PD		1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
AE6	pcie_rxn0	pcie_rxn0		0	I	OFF	OFF		1.8	vdda_pcie		SERDES		
AD7	pcie_rxp0	pcie_rxp0		0	I	OFF	OFF		1.8	vdda_pcie		SERDES		
AE8	pcie_txn0	pcie_txn0		0	O				1.8	vdda_pcie		SERDES		
AD9	pcie_txp0	pcie_txp0		0	O				1.8	vdda_pcie		SERDES		
F19	porz	porz		0	I				1.8/3.3	vddshv3	Yes	IHHV1833	PU/PD	
K24	resetn	resetn		0	I	PU	PU		1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
N2	rgmii0_rxc	rgmii0_rxc		0	I	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	0
		rmii1_txen		2	O									
		mii0_txclk		3	I									0
		vin2a_d5		4	I									0
		vin1b_d5		5	I									0
		usb3_ulpi_d2		6	IO									0
		pr1_mii_mt0_clk	No	11	I									0
		pr2_pru1_gpi11	No	12	I									
		pr2_pru1_gpo11	No	13	O									
		gpio5_26		14	IO									
		Driver off		15	I									
P2	rgmii0_rxctl	rgmii0_rxctl		0	I	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	0
		rmii1_txd1		2	O									
		mii0_txd3		3	O									
		vin2a_d6		4	I									0
		vin1b_d6		5	I									0
		usb3_ulpi_d3		6	IO									0
		pr1_mii0_txd3	No	11	O									
		pr2_pru1_gpi12	No	12	I									
		pr2_pru1_gpo12	No	13	O									
		gpio5_27		14	IO									
		Driver off		15	I									
N4	rgmii0_rxd0	rgmii0_rxd0		0	I	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	0
		rmii0_txd0		1	O									
		mii0_txd0		3	O									
		vin2a_fld0		4	I									
		vin1b_fld1		5	I									0
		usb3_ulpi_d7		6	IO									0
		pr1_mii0_txd0	No	11	O									
		pr2_pru1_gpi16	No	12	I									
		pr2_pru1_gpo16	No	13	O									
		gpio5_31		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
N3	rgmii0_rxd1	rgmii0_rxd1		0	I	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	0
		rmii0_txd1		1	O									
		mii0_txd1		3	O									
		vin2a_d9		4	I									0
		usb3_ulpi_d6		6	IO									0
		pr1_mii0_txd1	No	11	O									
		pr2_pru1_gpi15	No	12	I									
		pr2_pru1_gpo15	No	13	O									
		gpio5_30		14	IO									
		Driver off		15	I									
P1	rgmii0_rxd2	rgmii0_rxd2		0	I	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	0
		rmii0_txen		1	O									
		mii0_txen		3	O									
		vin2a_d8		4	I									0
		usb3_ulpi_d5		6	IO									0
		pr1_mii0_txen	No	11	O									
		pr2_pru1_gpi14	No	12	I									
		pr2_pru1_gpo14	No	13	O									
		gpio5_29		14	IO									
		Driver off		15	I									
N1	rgmii0_rxd3	rgmii0_rxd3		0	I	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	0
		rmii1_txd0		2	O									
		mii0_txd2		3	O									
		vin2a_d7		4	I									0
		vin1b_d7		5	I									0
		usb3_ulpi_d4		6	IO									0
		pr1_mii0_txd2	No	11	O									
		pr2_pru1_gpi13	No	12	I									
		pr2_pru1_gpo13	No	13	O									
		gpio5_28		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
T4	rgmii0_txc	rgmii0_txc		0	O	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	
		uart3_ctsn		1	I									1
		rmii1_rxd1		2	I									0
		mii0_rxd3		3	I									0
		vin2a_d3		4	I									0
		vin1b_d3		5	I									0
		usb3_ulpi_clk		6	I									0
		spi3_d0		7	IO									0
		spi4_cs2		8	IO									1
		pr1_mii0_rxd3	No	11	I									0
		pr2_pru1_gpi5	No	12	I									
		pr2_pru1_gpo5	No	13	O									
		gpio5_20		14	IO									
		Driver off		15	I									
T5	rgmii0_txctl	rgmii0_txctl		0	O	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	
		uart3_rtsn		1	O									
		rmii1_rxd0		2	I									0
		mii0_rxd2		3	I									0
		vin2a_d4		4	I									0
		vin1b_d4		5	I									0
		usb3_ulpi_stp		6	O									
		spi3_cs0		7	IO									1
		spi4_cs3		8	IO									1
		pr1_mii0_rxd2	No	11	I									0
		pr2_pru1_gpi6	No	12	I									
		pr2_pru1_gpo6	No	13	O									
		gpio5_21		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
R1	rgmii0_txd0	rgmii0_txd0		0	O	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	
		rmii0_rxd0		1	I									0
		mii0_rxd0		3	I									0
		vin2a_d10		4	I									0
		usb3_ulpi_d1		6	IO									0
		spi4_cs0		7	IO									1
		uart4_rtsn		8	O									
		pr1_mii0_rxd0	No	11	I									0
		pr2_pru1_gpi10	No	12	I									
		pr2_pru1_gpo10	No	13	O									
		gpio5_25		14	IO									
		Driver off		15	I									
R2	rgmii0_txd1	rgmii0_txd1		0	O	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	
		rmii0_rxd1		1	I									0
		mii0_rxd1		3	I									0
		vin2a_vsync0		4	I									
		vin1b_vsync1		5	I									0
		usb3_ulpi_d0		6	IO									0
		spi4_d0		7	IO									0
		uart4_ctsn		8	IO									1
		pr1_mii0_rxd1	No	11	I									0
		pr2_pru1_gpi9	No	12	I									
		pr2_pru1_gpo9	No	13	O									
		gpio5_24		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
P3	rgmii0_txd2	rgmii0_txd2		0	O	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	
		rmii0_rxer		1	I									0
		mii0_rxer		3	I									0
		vin2a_hsync0		4	I									
		vin1b_hsync1		5	I									0
		usb3_ulpi_nxt		6	I									0
		spi4_d1		7	IO									0
		uart4_txd		8	O									
		pr1_mii0_rxer	No	11	I									0
		pr2_pru1_gpi8	No	12	I									
		pr2_pru1_gpo8	No	13	O									
		gpio5_23		14	IO									
		Driver off		15	I									
P4	rgmii0_txd3	rgmii0_txd3		0	O	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	
		rmii0_crs		1	I									0
		mii0_crs		3	I									0
		vin2a_de0		4	I									
		vin1b_de1		5	I									0
		usb3_ulpi_dir		6	I									0
		spi4_sclk		7	IO									0
		uart4_rxd		8	I									1
		pr1_mii0_crs	No	11	I									0
		pr2_pru1_gpi7	No	12	I									
		pr2_pru1_gpo7	No	13	O									
		gpio5_22		14	IO									
		Driver off		15	I									
P5	RMII_MHZ_50_CLK	RMII_MHZ_50_CLK		0	IO	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin2a_d11		4	I									0
		pr2_pru1_gpi2	No	12	I									
		pr2_pru1_gpo2	No	13	O									
		gpio5_17		14	IO									
		Driver off		15	I									
E20	rstoutn	rstoutn		0	O	PD	PD		1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
K25	rtck	rtck		0	O	PU	OFF	0	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		gpio8_29		14	IO									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
B24	spi1_cs0	spi1_cs0		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	1
		gpio7_10		14	IO									
		Driver off		15	I									
C25	spi1_cs1	spi1_cs1		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	1
		spi2_cs1		3	IO									1
		gpio7_11		14	IO									
		Driver off		15	I									
E24	spi1_cs2	spi1_cs2		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart4_rxd		1	I									1
		mmc3_sdcd		2	I									1
		spi2_cs2		3	IO									1
		dcan2_tx		4	IO									1
		mdio_mclk		5	O									1
		hdmi1_hpd		6	IO									
		gpio7_12		14	IO									
		Driver off		15	I									
E25	spi1_cs3	spi1_cs3		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart4_txd		1	O									
		mmc3_sdwp		2	I									0
		spi2_cs3		3	IO									1
		dcan2_rx		4	IO									1
		mdio_d		5	IO									1
		hdmi1_cec		6	IO									
		gpio7_13		14	IO									
		Driver off		15	I									
D25	spi1_d0	spi1_d0		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		gpio7_9		14	IO									
		Driver off		15	I									
D24	spi1_d1	spi1_d1		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		gpio7_8		14	IO									
		Driver off		15	I									
C24	spi1_sclk	spi1_sclk		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		gpio7_7		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
F24	spi2_cs0	spi2_cs0		0	IO	PU	PU	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart3_rtsn		1	O									
		uart5_txd		2	O									
		gpio7_17		14	IO									
		Driver off		15	I									
G24	spi2_d0	spi2_d0		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		uart3_ctsn		1	I									1
		uart5_rxd		2	I									1
		gpio7_16		14	IO									
		Driver off		15	I									
F25	spi2_d1	spi2_d1		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		uart3_txd		1	O									
		gpio7_15		14	IO									
		Driver off		15	I									
G25	spi2_sclk	spi2_sclk		0	IO	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	0
		uart3_rxd		1	I									1
		gpio7_14		14	IO									
		Driver off		15	I									
K21	tclk	tclk		0	I	PU	PU	0	1.8/3.3	vddshv3	Yes	IQ1833	PU/PD	
L23	tdi	tdi		0	I	PU	PU	0	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		gpio8_27		14	I									
J20	tdo	tdo		0	O	PU	PU	0	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		gpio8_28		14	IO									
L21	tms	tms		0	I	PU	PU	0	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
L22	trstn	trstn		0	I	PD	PD		1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
L20	uart1_ctsn	uart1_ctsn		0	I	PU	PU	15	1.8/3.3	vddshv4	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart9_rxd		2	I									1
		mmc4_clk		3	IO									1
		gpio7_24		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
M24	uart1_rtsn	uart1_rtsn		0	O	PU	PU	15	1.8/3.3	vddshv4	Yes	Dual Voltage LVCMOS	PU/PD	
		uart9_txd		2	O									
		mmc4_cmd		3	IO									1
		gpio7_25		14	IO									
		Driver off		15	I									
L25	uart1_rxd	uart1_rxd		0	I	PU	PU	15	1.8/3.3	vddshv4	Yes	Dual Voltage LVCMOS	PU/PD	1
		mmc4_sdcd		3	I									1
		gpio7_22		14	IO									
		Driver off		15	I									
M25	uart1_txd	uart1_txd		0	O	PU	PU	15	1.8/3.3	vddshv4	Yes	Dual Voltage LVCMOS	PU/PD	
		mmc4_sdwp		3	I									0
		gpio7_23		14	IO									
		Driver off		15	I									
N22	uart2_ctsn	uart2_ctsn		0	I	PU	PU	15	1.8/3.3	vddshv4	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart3_rxd		2	I									1
		mmc4_dat2		3	IO									1
		uart10_rxd		4	I									1
		uart1_dtrn		5	O									
		gpio1_16		14	IO									
		Driver off		15	I									
N24	uart2_rtsn	uart2_rtsn		0	O	PU	PU	15	1.8/3.3	vddshv4	Yes	Dual Voltage LVCMOS	PU/PD	
		uart3_txd		1	O									
		uart3_irtx		2	O									
		mmc4_dat3		3	IO									1
		uart10_txd		4	O									
		uart1_rin		5	I									1
		gpio1_17		14	IO									
		Driver off		15	I									
N23	uart2_rxd	uart2_rxd		0	I	PU	PU	15	1.8/3.3	vddshv4	Yes	Dual Voltage LVCMOS	PU/PD	1
		uart3_ctsn		1	I									1
		uart3_rctx		2	O									
		mmc4_dat0		3	IO									1
		uart2_rxd		4	I									1
		uart1_dcdn		5	I									1
		gpio7_26		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
N25	uart2_txd	uart2_txd		0	O	PU	PU	15	1.8/3.3	vddshv4	Yes	Dual Voltage LVCMOS	PU/PD	
		uart3_rtsn		1	O									
		uart3_sd		2	O									
		mmc4_dat1		3	IO									1
		uart2_txd		4	O									
		uart1_dsrn		5	I									0
		gpio7_27		14	IO									
		Driver off		15	I									
N5	uart3_rxd	uart3_rxd		0	I	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	1
		rmii1_crs		2	I									0
		mii0_rxdv		3	I									0
		vin2a_d1		4	I									0
		vin1b_d1		5	I									0
		spi3_sclk		7	IO									0
		pr1_mii0_rxdv	No	11	I									0
		pr2_pru1_gpi3	No	12	I									
		pr2_pru1_gpo3	No	13	O									
		gpio5_18		14	IO									
		Driver off		15	I									
N6	uart3_txd	uart3_txd		0	O	PD	PD	15	1.8/3.3	vddshv9	Yes	Dual Voltage LVCMOS	PU/PD	
		rmii1_rxer		2	I									0
		mii0_rxclk		3	I									0
		vin2a_d2		4	I									0
		vin1b_d2		5	I									0
		spi3_d1		7	IO									0
		spi4_cs1		8	IO									1
		pr1_mii_mr0_clk	No	11	I									0
		pr2_pru1_gpi4	No	12	I									
		pr2_pru1_gpo4	No	13	O									
		gpio5_19		14	IO									
		Driver off		15	I									
AB7	usb1_dm	usb1_dm		0	IO	OFF	OFF		3.3	vdda33v_usb1		USBPHY		
AC6	usb1_dp	usb1_dp		0	IO	OFF	OFF		3.3	vdda33v_usb1		USBPHY		

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
AD3	usb1_drvvbus	usb1_drvvbus		0	O	PD	PD	15	1.8/3.3	vdda33v_usb2	Yes	Dual Voltage LVCMOS	PU/PD	
		timer16		7	IO									
		gpio6_12		14	IO									
		Driver off		15	I									
AC5	usb2_dm	usb2_dm		0	IO				3.3	vdda33v_usb2	No	USBPHY		
AB6	usb2_dp	usb2_dp		0	IO				3.3	vdda33v_usb2	No	USBPHY		
AA6	usb2_drvvbus	usb2_drvvbus		0	O	PD	PD	15	1.8/3.3	vdda33v_usb2	Yes	Dual Voltage LVCMOS	PU/PD	
		timer15		7	IO									
		gpio6_13		14	IO									
		Driver off		15	I									
AE5	usb_rxn0	usb_rxn0		0	I	OFF	OFF		1.8	vdda_usb1		SERDES		
		pcie_rxn1		1	I									
AD6	usb_rxp0	usb_rxp0		0	I	OFF	OFF		1.8	vdda_usb1		SERDES		
		pcie_rxp1		1	I									
AE3	usb_txn0	usb_txn0		0	O				1.8	vdda_usb1		SERDES		
		pcie_txn1		1	O									
AD4	usb_txp0	usb_txp0		0	O				1.8	vdda_usb1		SERDES		
		pcie_txp1		1	O									
J15, J16, J18, K12, K18, L12, L17, M11, M13, M15, M17, N11, N13, N15, N18, P10, P12, P14, P16, P18, R10, R12, R14, R16, R17, T11, T13, T15, T17, T9, U11, U13, U15, U18, U9, V10, V12, V14, V16, V18, W10, W12, W14, W16	vdd	vdd			PWR									
F20	vpp	vpp[(11)]			PWR									
AA10	vdda33v_usb1	vdda33v_usb1			PWR									
Y10	vdda33v_usb2	vdda33v_usb2			PWR									
L9	vdda_core_gmac	vdda_core_gmac			PWR									
T6	vdda_csi	vdda_csi			PWR									
R20	vdda_ddr	vdda_ddr			PWR									
N10	vdda_debug	vdda_debug			PWR									
K10, L10	vdda_dsp_iva	vdda_dsp_iva			PWR									
N9	vdda_gpu	vdda_gpu			PWR									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
W15, Y15	vdda_hdmi	vdda_hdmi			PWR									
K16, L16	vdda_mpu_abe	vdda_mpu_abe			PWR									
W13, Y13	vdda_osc	vdda_osc			PWR									
W11, Y11	vdda_pcie	vdda_pcie			PWR									
M10	vdda_per	vdda_per			PWR									
W8	vdda_usb1	vdda_usb1			PWR									
Y8	vdda_usb2	vdda_usb2			PWR									
Y9	vdda_usb3	vdda_usb3			PWR									
K14, L14	vdda_video	vdda_video			PWR									
G11, H20, W7, Y18	vdds18v	vdds18v			PWR									
AA19, P20, Y19	vdds18v_ddr1	vdds18v_ddr1			PWR									
G10, G9	vddshv1	vddshv1			PWR									
G15, G17, H15, H17, J19, K19	vddshv3	vddshv3			PWR									
M19, N19	vddshv4	vddshv4			PWR									
U7, U8	vddshv7	vddshv7			PWR									
N8, P8	vddshv8	vddshv8			PWR									
M7, N7	vddshv9	vddshv9			PWR									
J7, J8, K8	vddshv10	vddshv10			PWR									
F7, G7, H7	vddshv11	vddshv11			PWR									
T19, T20, V20, W17, W18, W20	vdds_ddr1	vdds_ddr1			PWR									
P7, R7	vdds_mlbp	vdds_mlbp			PWR									
H11, H13, H9, J11, J13, J9	vdd_dsp	vdd_dsp			PWR									
D8	vin2a_clk0	vin2a_clk0		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	
		vout2_fld		4	O									
		emu5		5	O									
		kbd_row0		9	I									0
		eQEP1A_in		10	I									0
		pr1_edio_data_in0	No	12	I									0
		pr1_edio_data_out0	No	13	O									
		gpio3_28 gpmc_a27 gpmc_a17		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
C8	vin2a_d0	vin2a_d0		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d23		4	O									
		emu10		5	O									
		uart9_ctsn		7	I									1
		spi4_d0		8	IO									0
		kbd_row4		9	I									0
		ehrpwm1B		10	O									
		pr1_uart0_rxd	No	11	I									1
		pr1_edio_data_in5	No	12	I									0
		pr1_edio_data_out5	No	13	O									
		gpio4_1		14	IO									
		Driver off		15	I									
B9	vin2a_d1	vin2a_d1		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d22		4	O									
		emu11		5	O									
		uart9_rtsn		7	O									
		spi4_cs0		8	IO									1
		kbd_row5		9	I									0
		ehrpwm1_tripzone_input		10	IO									0
		pr1_uart0_txd	No	11	O									
		pr1_edio_data_in6	No	12	I									0
		pr1_edio_data_out6	No	13	O									
		gpio4_2		14	IO									
		Driver off		15	I									
A7	vin2a_d2	vin2a_d2		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d21		4	O									
		emu12		5	O									
		uart10_rxd		8	I									1
		kbd_row6		9	I									0
		eCAP1_in_PWM1_out		10	IO									0
		pr1_ecap0_ecap_capin_apwm_o	No	11	IO									0
		pr1_edio_data_in7	No	12	I									0
		pr1_edio_data_out7	No	13	O									
		gpio4_3		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
A9	vin2a_d3	vin2a_d3		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d20		4	O									
		emu13		5	O									
		uart10_txd		8	O									
		kbd_col0		9	O									
		ehrpwm1_synci		10	I									0
		pr1_edc_latch0_in	No	11	I									0
		pr1_pru1_gpi0	No	12	I									
		pr1_pru1_gpo0	No	13	O									
		gpio4_4		14	IO									
		Driver off		15	I									
A8	vin2a_d4	vin2a_d4		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d19		4	O									
		emu14		5	O									
		uart10_ctsn		8	I									1
		kbd_col1		9	O									
		ehrpwm1_synco		10	O									
		pr1_edc_sync0_out	No	11	O									
		pr1_pru1_gpi1	No	12	I									
		pr1_pru1_gpo1	No	13	O									
		gpio4_5		14	IO									
		Driver off		15	I									
A11	vin2a_d5	vin2a_d5		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d18		4	O									
		emu15		5	O									
		uart10_rtsn		8	O									
		kbd_col2		9	O									
		eQEP2A_in		10	I									0
		pr1_edio_sof	No	11	O									
		pr1_pru1_gpi2	No	12	I									
		pr1_pru1_gpo2	No	13	O									
		gpio4_6		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
F10	vin2a_d6	vin2a_d6		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d17		4	O									
		emu16		5	O									
		mii1_rxd1		8	I									0
		kbd_col3		9	O									
		eQEP2B_in		10	I									0
		pr1_mii_mt1_clk	No	11	I									0
		pr1_pru1_gpi3	No	12	I									
		pr1_pru1_gpo3	No	13	O									
		gpio4_7		14	IO									
		Driver off		15	I									
A10	vin2a_d7	vin2a_d7		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d16		4	O									
		emu17		5	O									
		mii1_rxd2		8	I									0
		kbd_col4		9	O									
		eQEP2_index		10	IO									0
		pr1_mii1_txen	No	11	O									
		pr1_pru1_gpi4	No	12	I									
		pr1_pru1_gpo4	No	13	O									
		gpio4_8		14	IO									
		Driver off		15	I									
B10	vin2a_d8	vin2a_d8		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d15		4	O									
		emu18		5	O									
		mii1_rxd3		8	I									0
		kbd_col5		9	O									
		eQEP2_strobe		10	IO									0
		pr1_mii1_txd3	No	11	O									
		pr1_pru1_gpi5	No	12	I									
		pr1_pru1_gpo5	No	13	O									
		gpio4_9 gpmc_a26		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
E10	vin2a_d9	vin2a_d9		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vout2_d14		4	O									
		emu19		5	O									
		mii1_rxd0		8	I									0
		kbd_col6		9	O									
		ehrpwm2A		10	O									
		pr1_mii1_txd2	No	11	O									
		pr1_pru1_gpi6	No	12	I									
		pr1_pru1_gpo6	No	13	O									
		gpio4_10 gpmc_a25		14	IO									
		Driver off		15	I									
D10	vin2a_d10	vin2a_d10		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		mdio_mclk		3	O									1
		vout2_d13		4	O									
		kbd_col7		9	O									
		ehrpwm2B		10	O									
		pr1_mdio_mdclk	No	11	O									
		pr1_pru1_gpi7	No	12	I									
		pr1_pru1_gpo7	No	13	O									
		gpio4_11 gpmc_a24		14	IO									
		Driver off		15	I									
C10	vin2a_d11	vin2a_d11		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		mdio_d		3	IO									1
		vout2_d12		4	O									
		kbd_row7		9	I									0
		ehrpwm2_tripzone_input		10	IO									0
		pr1_mdio_data	No	11	IO									1
		pr1_pru1_gpi8	No	12	I									
		pr1_pru1_gpo8	No	13	O									
		gpio4_12 gpmc_a23		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
B11	vin2a_d12	vin2a_d12		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		rgmii1_txc		3	O									
		vout2_d11		4	O									
		mii1_rxclk		8	I									0
		kbd_col8		9	O									
		eCAP2_in_PWM2_out		10	IO									0
		pr1_mii1_txd1	No	11	O									
		pr1_pru1_gpi9	No	12	I									
		pr1_pru1_gpo9	No	13	O									
		gpio4_13		14	IO									
		Driver off		15	I									
D11	vin2a_d13	vin2a_d13		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		rgmii1_txctl		3	O									
		vout2_d10		4	O									
		mii1_rxdv		8	I									0
		kbd_row8		9	I									0
		eQEP3A_in		10	I									0
		pr1_mii1_txd0	No	11	O									
		pr1_pru1_gpi10	No	12	I									
		pr1_pru1_gpo10	No	13	O									
		gpio4_14		14	IO									
		Driver off		15	I									
C11	vin2a_d14	vin2a_d14		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		rgmii1_txd3		3	O									
		vout2_d9		4	O									
		mii1_txclk		8	I									0
		eQEP3B_in		10	I									0
		pr1_mii_mr1_clk	No	11	I									0
		pr1_pru1_gpi11	No	12	I									
		pr1_pru1_gpo11	No	13	O									
		gpio4_15		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
B12	vin2a_d15	vin2a_d15		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		rgmii1_txd2		3	O									
		vout2_d8		4	O									
		mii1_txd0		8	O									
		eQEP3_index		10	IO									0
		pr1_mii1_rxdv	No	11	I									0
		pr1_pru1_gpi12	No	12	I									
		pr1_pru1_gpo12	No	13	O									
		gpio4_16		14	IO									
		Driver off		15	I									
A12	vin2a_d16	vin2a_d16		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin2b_d7		2	I									0
		rgmii1_txd1		3	O									
		vout2_d7		4	O									
		mii1_txd1		8	O									
		eQEP3_strobe		10	IO									0
		pr1_mii1_rxd3	No	11	I									0
		pr1_pru1_gpi13	No	12	I									
		pr1_pru1_gpo13	No	13	O									
		gpio4_24		14	IO									
		Driver off		15	I									
A13	vin2a_d17	vin2a_d17		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin2b_d6		2	I									0
		rgmii1_txd0		3	O									
		vout2_d6		4	O									
		mii1_txd2		8	O									
		ehrpwm3A		10	O									
		pr1_mii1_rxd2	No	11	I									0
		pr1_pru1_gpi14	No	12	I									
		pr1_pru1_gpo14	No	13	O									
		gpio4_25		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
E11	vin2a_d18	vin2a_d18		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin2b_d5		2	I									0
		rgmii1_rxc		3	I									0
		vout2_d5		4	O									
		mii1_txd3		8	O									
		ehrpwm3B		10	O									
		pr1_mii1_rxd1	No	11	I									0
		pr1_pru1_gpi15	No	12	I									
		pr1_pru1_gpo15	No	13	O									
		gpio4_26		14	IO									
		Driver off		15	I									
F11	vin2a_d19	vin2a_d19		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin2b_d4		2	I									0
		rgmii1_rxctl		3	I									0
		vout2_d4		4	O									
		mii1_txer		8	O									0
		ehrpwm3_tripzone_input		10	IO									0
		pr1_mii1_rxd0	No	11	I									0
		pr1_pru1_gpi16	No	12	I									
		pr1_pru1_gpo16	No	13	O									
		gpio4_27		14	IO									
		Driver off		15	I									
B13	vin2a_d20	vin2a_d20		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin2b_d3		2	I									0
		rgmii1_rxd3		3	I									0
		vout2_d3		4	O									
		mii1_rxer		8	I									0
		eCAP3_in_PWM3_out		10	IO									0
		pr1_mii1_rxer	No	11	I									0
		pr1_pru1_gpi17	No	12	I									
		pr1_pru1_gpo17	No	13	O									
		gpio4_28		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
E13	vin2a_d21	vin2a_d21		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin2b_d2		2	I									0
		rgmii1_rxd2		3	I									0
		vout2_d2		4	O									
		mii1_col		8	I									0
		pr1_mii1_rxlink	No	11	I									0
		pr1_pru1_gpi18	No	12	I									
		pr1_pru1_gpo18	No	13	O									
		gpio4_29		14	IO									
		Driver off		15	I									
C13	vin2a_d22	vin2a_d22		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin2b_d1		2	I									0
		rgmii1_rxd1		3	I									0
		vout2_d1		4	O									
		mii1_crs		8	I									0
		pr1_mii1_col	No	11	I									0
		pr1_pru1_gpi19	No	12	I									
		pr1_pru1_gpo19	No	13	O									
		gpio4_30		14	IO									
		Driver off		15	I									
D13	vin2a_d23	vin2a_d23		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	0
		vin2b_d0		2	I									0
		rgmii1_rxd0		3	I									0
		vout2_d0		4	O									
		mii1_txen		8	O									
		pr1_mii1_crs	No	11	I									0
		pr1_pru1_gpi20	No	12	I									
		pr1_pru1_gpo20	No	13	O									
		gpio4_31		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
B7	vin2a_de0	vin2a_de0		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	
		vin2a_fld0		1	I									
		vin2b_fld1		2	I									
		vin2b_de1		3	I									
		vout2_de		4	O									
		emu6		5	O									
		kbd_row1		9	I									0
		eQEP1B_in		10	I									0
		pr1_edio_data_in1	No	12	I									0
		pr1_edio_data_out1	No	13	O									
		gpio3_29		14	IO									
		Driver off		15	I									
C7	vin2a_fld0	vin2a_fld0		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	
		vin2b_clk1		2	I									
		vout2_clk		4	O									
		emu7		5	O									
		eQEP1_index		10	IO									0
		pr1_edio_data_in2	No	12	I									0
		pr1_edio_data_out2	No	13	O									
		gpio3_30 gpmc_a27 gpmc_a18		14	IO									
		Driver off		15	I									
E8	vin2a_hsync0	vin2a_hsync0		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	
		vin2b_hsync1		3	I									
		vout2_hsync		4	O									
		emu8		5	O									
		uart9_rxd		7	I									1
		spi4_sclk		8	IO									0
		kbd_row2		9	I									0
		eQEP1_strobe		10	IO									0
		pr1_uart0_cts_n	No	11	I									1
		pr1_edio_data_in3	No	12	I									0
		pr1_edio_data_out3	No	13	O									
		gpio3_31 gpmc_a27		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[(1)] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
B8	vin2a_vsync0	vin2a_vsync0		0	I	PD	PD	15	1.8/3.3	vddshv1	Yes	Dual Voltage LVCMOS	PU/PD	
		vin2b_vsync1		3	I									
		vout2_vsync		4	O									
		emu9		5	O									
		uart9_txd		7	O									
		spi4_d1		8	IO									0
		kbd_row3		9	I									0
		ehrpwm1A		10	O									
		pr1_uart0_rts_n	No	11	O									
		pr1_edio_data_in4	No	12	I									0
		pr1_edio_data_out4	No	13	O									
		gpio4_0		14	IO									
		Driver off		15	I									
A1, A25, AA13, AA15, AA7, AA8, AA9, AB8, AC13, AE1, AE15, AE25, G13, G16, G8, H10, H12, H14, H16, H18, H19, H8, J10, J12, J14, J17, K11, K13, K15, K17, K9, L11, L13, L15, L18, L8, M12, M14, M16, M18, M20, M8, M9, N12, N14, N16, N17, N20, P11, P13, P15, P17, P19, P9, R11, R13, R15, R18, R19, R8, R9, T10, T12, T14, T16, T18, T8, U10, U12, U14, U16, U17, U19, V11, V13, V15, V17, V19, V8, V9, W19, W9, Y14, Y16, Y17, Y7	vss	vss			GND									
AA12	vssa_osc0	vssa_osc0			GND									
AB11	vssa_osc1	vssa_osc1			GND									
AC10	Wakeup0	dcan1_rx		1	I			15	1.8/3.3	vdda33v_usb1	Yes	IHHV1833	PU/PD	1
		gpio1_0 sys_nirq2		14	I									
		Driver off		15	I									

Table 4-1. Pin Attributes(1) (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
AB10	Wakeup3	sys_nirq1		1	I			15	1.8/3.3	vdda33v_usb1	Yes	IHHV1833	PU/PD	
		gpio1_3 dcan2_rx		14	I									
		Driver off		15	I									
Y12	xi_osc0	xi_osc0		0	I				1.8	vdda_osc	No	LVCMOS Analog		
AC11	xi_osc1	xi_osc1		0	I				1.8	vdda_osc	No	LVCMOS Analog		
AB12	xo_osc0	xo_osc0		0	O				1.8	vdda_osc	No	LVCMOS Analog		
AA11	xo_osc1	xo_osc1		0	A				1.8	vdda_osc	No	LVCMOS Analog		
J25	xref_clk0	xref_clk0		0	I	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		mcasp2_axr8		1	IO									0
		mcasp1_axr4		2	IO									0
		mcasp1_ahclkx		3	O									
		mcasp5_ahclkx		4	O									
		atl_clk0		5	O									
		vin1a_d0		7	I									0
		hdq0		8	IO									1
		clkout2		9	O									
		timer13		10	IO									
		pr2_mii1_col	No	11	I									0
		pr2_pru1_gpi5	No	12	I									
		pr2_pru1_gpo5	No	13	O									
		gpio6_17		14	IO									
		Driver off		15	I									

Table 4-1. Pin Attributes[1] (continued)

BALL NUMBER [1]	BALL NAME [2]	SIGNAL NAME [3]	Support [4]	MUXMODE [5]	TYPE [6]	BALL RESET STATE [7]	BALL RESET REL. STATE [8]	BALL RESET REL. MUXMODE [9]	I/O VOLTAGE VALUE [10]	POWER [11]	HYS [12]	BUFFER TYPE [13]	PULL UP/DOWN TYPE [14]	DSIS [15]
J24	xref_clk1	xref_clk1		0	I	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		mcasp2_axr9		1	IO									0
		mcasp1_axr5		2	IO									0
		mcasp2_ahclkx		3	O									
		mcasp6_ahclkx		4	O									
		atl_clk1		5	O									
		vin1a_clk0		7	I									0
		timer14		10	IO									
		pr2_mii1_crs	No	11	I									0
		pr2_pru1_gpi6	No	12	I									
		pr2_pru1_gpo6	No	13	O									
		gpio6_18		14	IO									
		Driver off		15	I									
H24	xref_clk2	xref_clk2		0	I	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		mcasp2_axr10		1	IO									0
		mcasp1_axr6		2	IO									0
		mcasp3_ahclkx		3	O									
		mcasp7_ahclkx		4	O									
		atl_clk2		5	O									
		timer15		10	IO									
		gpio6_19		14	IO									
		Driver off		15	I									
H25	xref_clk3	xref_clk3		0	I	PD	PD	15	1.8/3.3	vddshv3	Yes	Dual Voltage LVCMOS	PU/PD	
		mcasp2_axr11		1	IO									0
		mcasp1_axr7		2	IO									0
		mcasp4_ahclkx		3	O									
		mcasp8_ahclkx		4	O									
		atl_clk3		5	O									
		hdq0		7	IO									1
		clkout3		9	O									
		timer16		10	IO									
		gpio6_20		14	IO									
		Driver off		15	I									

(1) NA in this table stands for Not Applicable.

(2) For more information on recommended operating conditions, see Section 5.4, *Recommended Operating Conditions*.

(3) The pullup or pulldown block strength is equal to: minimum = 50 μA, typical = 100 μA, maximum = 250 μA.

(4) The output impedance settings of this IO cell are programmable; by default, the value is DS[1:0] = 10, this means 40 Ω. For more information on DS[1:0] register configuration, see the Device TRM.

(5) IO drive strength for usb1_dp, usb1_dm, usb2_dp and usb2_dm: minimum 18.3 mA, maximum 89 mA (for a power supply vdda33v_usb1 and vdda33v_usb2 = 3.46 V).

(6) Minimum PU = 900 Ω, maximum PU = 3.090 kΩ and minimum PD = 14.25 kΩ, maximum PD = 24.8 kΩ.
For more information, see chapter 7 of the USB2.0 specification, in particular section Signaling / Device Speed Identification.

(7) This function will not be supported on some pin-compatible roadmap devices. Pin compatibility can be maintained in the future by not using these GPIO signals.

(8) In PUx / PDy, x and y = 60 to 200 μA.
The output impedance settings (or drive strengths) of this IO are programmable (34 Ω, 40 Ω, 48 Ω, 60 Ω, 80 Ω) depending on the values of the I[2:0] registers.

(9) The internal pull resistors for balls A4, E7, D6, C5, D7, C6, A5, B6 are permanently disabled when sysboot15 is set to 0 as described in the section Sysboot Configuration of the Device TRM. If internal pull-up/down resistors are desired on these balls then sysboot15 should be set to 1. If gpmc boot mode is used with SYSBOOT15=0 (not recommended) then external pull-downs should be implemented to keep the address bus at logic-1 value during boot since the gpmc ms-address bits are high-z during boot.

(10) I2C6 is not supported in TI standard software. I2C6 is not recommended for use to due to internal clock/reset dependencies on i2c1-5 and uart7.

(11) This signal is valid only for High-Security devices. For more details, see Section 5.8 *VPP Specification for One-Time Programmable (OTP) eFUSEs*. For General Purpose devices do not connect any signal, test point, or board trace to this signal.

4.3 Signal Descriptions

Many signals are available on multiple pins, according to the software configuration of the pin multiplexing options.

1. **SIGNAL NAME:** The name of the signal passing through the pin.

NOTE

The subsystem multiplexing signals are not described in Table 4-1 and Table 4-32.

2. **DESCRIPTION:** Description of the signal
3. **TYPE:** Signal direction and type:
   - I = Input
   - O = Output
   - IO = Input or output
   - D = Open Drain
   - DS = Differential
   - A = Analog
   - PWR = Power
   - GND = Ground
4. **BALL:** Associated ball(s) bottom

NOTE

For more information, see the Control Module / Control Module Register Manual section of the device TRM.

4.3.1 VIP

NOTE

For more information, see the Video Input Port (VIP) section of the device TRM.

Table 4-2. VIP Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
Video Input 1			
vin1a_clk0	Video Input 1 Port A Clock input. Input clock for 8-bit 16-bit or 24-bit Port A video capture. Input data is sampled on the CLK0 edge.	I	G3, J24, Y5
vin1a_d0	Video Input 1 Port A Data input	I	AA1, B23, F1, J25
vin1a_d1	Video Input 1 Port A Data input	I	B22, E2, Y3
vin1a_d2	Video Input 1 Port A Data input	I	A23, E1, W2

Table 4-2. VIP Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
vin1a_d3	Video Input 1 Port A Data input	I	A22, AA3, C1
vin1a_d4	Video Input 1 Port A Data input	I	AA2, B21, D1
vin1a_d5	Video Input 1 Port A Data input	I	A21, D2, Y4
vin1a_d6	Video Input 1 Port A Data input	I	B1, D19, Y1
vin1a_d7	Video Input 1 Port A Data input	I	B2, E19, Y2
vin1a_d8	Video Input 1 Port A Data input	I	C2, F16
vin1a_d9	Video Input 1 Port A Data input	I	D3, E16
vin1a_d10	Video Input 1 Port A Data input	I	A2, E17
vin1a_d11	Video Input 1 Port A Data input	I	A19, B3
vin1a_d12	Video Input 1 Port A Data input	I	B18, C3
vin1a_d13	Video Input 1 Port A Data input	I	B16, C4
vin1a_d14	Video Input 1 Port A Data input	I	A3, B17
vin1a_d15	Video Input 1 Port A Data input	I	A18, B4
vin1a_d16	Video Input 1 Port A Data input	I	M1
vin1a_d17	Video Input 1 Port A Data input	I	M2
vin1a_d18	Video Input 1 Port A Data input	I	L2
vin1a_d19	Video Input 1 Port A Data input	I	L1
vin1a_d20	Video Input 1 Port A Data input	I	K3
vin1a_d21	Video Input 1 Port A Data input	I	K2
vin1a_d22	Video Input 1 Port A Data input	I	J1
vin1a_d23	Video Input 1 Port A Data input	I	K1
vin1a_de0	Video Input 1 Port A Field ID input	I	C17, J2, Y6
vin1a_fld0	Video Input 1 Port A Field ID input	I	C16, L3
vin1a_hsync0	Video Input 1 Port A Horizontal Sync input	I	AA4, B14, K4
vin1a_vsync0	Video Input 1 Port A Vertical Sync input	I	AB1, D14, H1
vin1b_clk1	Video Input 1 Port B Clock input	I	J2, L5
vin1b_d0	Video Input 1 Port B Data input	I	L6, M1
vin1b_d1	Video Input 1 Port B Data input	I	M2, N5
vin1b_d2	Video Input 1 Port B Data input	I	L2, N6
vin1b_d3	Video Input 1 Port B Data input	I	L1, T4
vin1b_d4	Video Input 1 Port B Data input	I	K3, T5
vin1b_d5	Video Input 1 Port B Data input	I	K2, N2
vin1b_d6	Video Input 1 Port B Data input	I	J1, P2

Table 4-2. VIP Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
vin1b_d7	Video Input 1 Port B Data input	I	K1, N1
vin1b_de1	Video Input 1 Port B Field ID input	I	L3, P4
vin1b_fld1	Video Input 1 Port B Field ID input	I	G1, N4
vin1b_hsync1	Video Input 1 Port B Horizontal Sync input	I	K4, P3
vin1b_vsync1	Video Input 1 Port B Vertical Sync input	I	H1, R2
Video Input 2			
vin2a_clk0	Video Input 2 Port A Clock input.	I	D8, L5
vin2a_d0	Video Input 2 Port A Data input	I	C8, L6
vin2a_d1	Video Input 2 Port A Data input	I	B9, N5
vin2a_d2	Video Input 2 Port A Data input	I	A7, N6
vin2a_d3	Video Input 2 Port A Data input	I	A9, T4
vin2a_d4	Video Input 2 Port A Data input	I	A8, T5
vin2a_d5	Video Input 2 Port A Data input	I	A11, N2
vin2a_d6	Video Input 2 Port A Data input	I	F10, P2
vin2a_d7	Video Input 2 Port A Data input	I	A10, N1
vin2a_d8	Video Input 2 Port A Data input	I	B10, P1
vin2a_d9	Video Input 2 Port A Data input	I	E10, N3
vin2a_d10	Video Input 2 Port A Data input	I	D10, R1
vin2a_d11	Video Input 2 Port A Data input	I	C10, P5
vin2a_d12	Video Input 2 Port A Data input	I	B11
vin2a_d13	Video Input 2 Port A Data input	I	D11
vin2a_d14	Video Input 2 Port A Data input	I	C11
vin2a_d15	Video Input 2 Port A Data input	I	B12
vin2a_d16	Video Input 2 Port A Data input	I	A12
vin2a_d17	Video Input 2 Port A Data input	I	A13
vin2a_d18	Video Input 2 Port A Data input	I	E11
vin2a_d19	Video Input 2 Port A Data input	I	F11
vin2a_d20	Video Input 2 Port A Data input	I	B13
vin2a_d21	Video Input 2 Port A Data input	I	E13
vin2a_d22	Video Input 2 Port A Data input	I	C13
vin2a_d23	Video Input 2 Port A Data input	I	D13
vin2a_de0	Video Input 2 Port A Field ID input	I	B7, P4
vin2a_fld0	Video Input 2 Port A Field ID input	I	B7, C7, N4

Table 4-2. VIP Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
vin2a_hsync0	Video Input 2 Port A Horizontal Sync input	I	E8, P3
vin2a_vsync0	Video Input 2 Port A Vertical Sync input	I	B8, R2
vin2b_clk1	Video Input 2 Port B Clock input	I	AB1, C7, L4, H6
vin2b_d0	Video Input 2 Port B Data input	I	AA1, D13, A4
vin2b_d1	Video Input 2 Port B Data input	I	C13, Y3, E7
vin2b_d2	Video Input 2 Port B Data input	I	E13, W2, D6
vin2b_d3	Video Input 2 Port B Data input	I	AA3, B13, C5
vin2b_d4	Video Input 2 Port B Data input	I	AA2, F11, B5
vin2b_d5	Video Input 2 Port B Data input	I	E11, Y4, D7
vin2b_d6	Video Input 2 Port B Data input	I	A13, Y1, C6
vin2b_d7	Video Input 2 Port B Data input	I	A12, Y2, A5
vin2b_de1	Video Input 2 Port B Field ID input	I	AA4, B7, H2
vin2b_fld1	Video Input 2 Port B Field ID input	I	B7, H6
vin2b_hsync1	Video Input 2 Port B Horizontal Sync input	I	E8, Y5, B6
vin2b_vsync1	Video Input 2 Port B Vertical Sync input	I	B8, Y6, A6

4.3.2 DSS

Table 4-3. DSS Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
DPI Video Output 2			
vout2_clk	Video Output 2 Clock output	O	C7
vout2_d0	Video Output 2 Data output	O	D13
vout2_d1	Video Output 2 Data output	O	C13
vout2_d2	Video Output 2 Data output	O	E13
vout2_d3	Video Output 2 Data output	O	B13
vout2_d4	Video Output 2 Data output	O	F11
vout2_d5	Video Output 2 Data output	O	E11
vout2_d6	Video Output 2 Data output	O	A13
vout2_d7	Video Output 2 Data output	O	A12
vout2_d8	Video Output 2 Data output	O	B12
vout2_d9	Video Output 2 Data output	O	C11
vout2_d10	Video Output 2 Data output	O	D11
vout2_d11	Video Output 2 Data output	O	B11
vout2_d12	Video Output 2 Data output	O	C10
vout2_d13	Video Output 2 Data output	O	D10
vout2_d14	Video Output 2 Data output	O	E10
vout2_d15	Video Output 2 Data output	O	B10
vout2_d16	Video Output 2 Data output	O	A10
vout2_d17	Video Output 2 Data output	O	F10
vout2_d18	Video Output 2 Data output	O	A11
vout2_d19	Video Output 2 Data output	O	A8
vout2_d20	Video Output 2 Data output	O	A9
vout2_d21	Video Output 2 Data output	O	A7
vout2_d22	Video Output 2 Data output	O	B9
vout2_d23	Video Output 2 Data output	O	C8
vout2_de	Video Output 2 Data Enable output	O	B7
vout2_fld	Video Output 2 Field ID output. This signal is not used for embedded sync modes.	O	D8
vout2_hsync	Video Output 2 Horizontal Sync output. This signal is not used for embedded sync modes.	O	E8
vout2_vsync	Video Output 2 Vertical Sync output. This signal is not used for embedded sync modes.	O	B8
DPI Video Output 3			
vout3_clk	Video Output 3 Clock output	O	G3
vout3_d0	Video Output 3 Data output	O	F1
vout3_d1	Video Output 3 Data output	O	E2
vout3_d2	Video Output 3 Data output	O	E1
vout3_d3	Video Output 3 Data output	O	C1
vout3_d4	Video Output 3 Data output	O	D1
vout3_d5	Video Output 3 Data output	O	D2
vout3_d6	Video Output 3 Data output	O	B1
vout3_d7	Video Output 3 Data output	O	B2
vout3_d8	Video Output 3 Data output	O	C2
vout3_d9	Video Output 3 Data output	O	D3
vout3_d10	Video Output 3 Data output	O	A2
vout3_d11	Video Output 3 Data output	O	B3

Table 4-3. DSS Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
vout3_d12	Video Output 3 Data output	O	C3
vout3_d13	Video Output 3 Data output	O	C4
vout3_d14	Video Output 3 Data output	O	A3
vout3_d15	Video Output 3 Data output	O	B4
vout3_d16	Video Output 3 Data output	O	M1
vout3_d17	Video Output 3 Data output	O	M2
vout3_d18	Video Output 3 Data output	O	L2
vout3_d19	Video Output 3 Data output	O	L1
vout3_d20	Video Output 3 Data output	O	K3
vout3_d21	Video Output 3 Data output	O	K2
vout3_d22	Video Output 3 Data output	O	J1
vout3_d23	Video Output 3 Data output	O	K1
vout3_de	Video Output 3 Data Enable output	O	J2
vout3_fld	Video Output 3 Field ID output. This signal is not used for embedded sync modes.	O	L3
vout3_hsync	Video Output 3 Horizontal Sync output. This signal is not used for embedded sync modes.	O	K4
vout3_vsync	Video Output 3 Vertical Sync output. This signal is not used for embedded sync modes.	O	H1

4.3.3 HDMI

NOTE

For more information, see the Display Subsystem / Display Subsystem Overview of the device TRM.

Table 4-4. HDMI Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
hdmi1_cec	HDMI consumer electronic control	IOD	E25, H23
hdmi1_hpd	HDMI display hot plug detect	IO	E24, H22
hdmi1_ddc_scl	HDMI display data channel clock	IOD	F23
hdmi1_ddc_sda	HDMI display data channel data	IOD	G21
hdmi1_clockx	HDMI clock differential positive or negative	ODS	AE9
hdmi1_clocky	HDMI clock differential positive or negative	ODS	AD10
hdmi1_data2x	HDMI data 2 differential positive or negative	ODS	AE14
hdmi1_data2y	HDMI data 2 differential positive or negative	ODS	AD15
hdmi1_data1x	HDMI data 1 differential positive or negative	ODS	AE12
hdmi1_data1y	HDMI data 1 differential positive or negative	ODS	AD13
hdmi1_data0x	HDMI data 0 differential positive or negative	ODS	AE11
hdmi1_data0y	HDMI data 0 differential positive or negative	ODS	AD12

4.3.4 CSI2

NOTE

For more information, see the CAL Subsystem / CAL Subsystem Overview of the device TRM.

Table 4-5. CSI 2 Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
csi2_0_dx0	Serial data/clock input - line 0 (position 1)	I	AC1
csi2_0_dy0	Serial data/clock input - line 0 (position 1)	I	AB2
csi2_0_dx1	Serial data/clock input - line 1 (position 2)	I	AD1
csi2_0_dy1	Serial data/clock input - line 1 (position 2)	I	AC2
csi2_0_dx2	Serial data/clock input - line 2 (position 3)	I	AE2
csi2_0_dy2	Serial data/clock input - line 2 (position 3)	I	AD2

4.3.5 EMIF

NOTE

For more information, see the Memory Subsystem / EMIF Controller section of the device TRM.

NOTE

The index number 1 which is part of the EMIF1 signal prefixes (ddr1_*) listed in Table 4-6, EMIF Signal Descriptions, column "SIGNAL NAME" not to be confused with DDR1 type of SDRAM memories.

Table 4-6. EMIF Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
ddr1_csn0	EMIF1 Chip Select 0	O	AC19
ddr1_cke	EMIF1 Clock Enable	O	AB18
ddr1_ck	EMIF1 Clock	O	AD21
ddr1_nck	EMIF1 Negative Clock	O	AE21
ddr1_odt0	EMIF1 On-Die Termination for Chip Select 0	O	AD18
ddr1_casn	EMIF1 Column Address Strobe	O	AD16
ddr1_rasn	EMIF1 Row Address Strobe	O	AD17
ddr1_wen	EMIF1 Write Enable	O	AE18
ddr1_rst	EMIF1 Reset output (DDR3-SDRAM only)	O	AE17
ddr1_ba0	EMIF1 Bank Address	O	AE16
ddr1_ba1	EMIF1 Bank Address	O	AA16
ddr1_ba2	EMIF1 Bank Address	O	AB16
ddr1_a0	EMIF1 Address Bus	O	AC18
ddr1_a1	EMIF1 Address Bus	O	AE19
ddr1_a2	EMIF1 Address Bus	O	AD19
ddr1_a3	EMIF1 Address Bus	O	AB19
ddr1_a4	EMIF1 Address Bus	O	AD20
ddr1_a5	EMIF1 Address Bus	O	AE20
ddr1_a6	EMIF1 Address Bus	O	AA18
ddr1_a7	EMIF1 Address Bus	O	AA20
ddr1_a8	EMIF1 Address Bus	O	Y21
ddr1_a9	EMIF1 Address Bus	O	AC20
ddr1_a10	EMIF1 Address Bus	O	AA21
ddr1_a11	EMIF1 Address Bus	O	AC21
ddr1_a12	EMIF1 Address Bus	O	AC22
ddr1_a13	EMIF1 Address Bus	O	AC15

Table 4-6. EMIF Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
ddr1_a14	EMIF1 Address Bus	O	AB15
ddr1_a15	EMIF1 Address Bus	O	AC16
ddr1_d0	EMIF1 Data Bus	IO	AA23
ddr1_d1	EMIF1 Data Bus	IO	AC24
ddr1_d2	EMIF1 Data Bus	IO	AB24
ddr1_d3	EMIF1 Data Bus	IO	AD24
ddr1_d4	EMIF1 Data Bus	IO	AB23
ddr1_d5	EMIF1 Data Bus	IO	AC23
ddr1_d6	EMIF1 Data Bus	IO	AD23
ddr1_d7	EMIF1 Data Bus	IO	AE24
ddr1_d8	EMIF1 Data Bus	IO	AA24
ddr1_d9	EMIF1 Data Bus	IO	W25
ddr1_d10	EMIF1 Data Bus	IO	Y23
ddr1_d11	EMIF1 Data Bus	IO	AD25
ddr1_d12	EMIF1 Data Bus	IO	AC25
ddr1_d13	EMIF1 Data Bus	IO	AB25
ddr1_d14	EMIF1 Data Bus	IO	AA25
ddr1_d15	EMIF1 Data Bus	IO	W24
ddr1_d16	EMIF1 Data Bus	IO	W23
ddr1_d17	EMIF1 Data Bus	IO	U25
ddr1_d18	EMIF1 Data Bus	IO	U24
ddr1_d19	EMIF1 Data Bus	IO	W21
ddr1_d20	EMIF1 Data Bus	IO	T22
ddr1_d21	EMIF1 Data Bus	IO	U22
ddr1_d22	EMIF1 Data Bus	IO	U23
ddr1_d23	EMIF1 Data Bus	IO	T21
ddr1_d24	EMIF1 Data Bus	IO	T23
ddr1_d25	EMIF1 Data Bus	IO	T25
ddr1_d26	EMIF1 Data Bus	IO	T24
ddr1_d27	EMIF1 Data Bus	IO	P21
ddr1_d28	EMIF1 Data Bus	IO	N21
ddr1_d29	EMIF1 Data Bus	IO	P22
ddr1_d30	EMIF1 Data Bus	IO	P23
ddr1_d31	EMIF1 Data Bus	IO	P24
ddr1_dqm0	EMIF1 Data Mask	O	AE23
ddr1_dqm1	EMIF1 Data Mask	O	W22
ddr1_dqm2	EMIF1 Data Mask	O	U21
ddr1_dqm3	EMIF1 Data Mask	O	P25
ddr1_dqs0	Data strobe 0 input/output for byte 0 of the 32-bit data bus. This signal is output to the EMIF1 memory when writing and input when reading.	IO	AD22
ddr1_dqsn0	Data strobe 0 invert	IO	AE22
ddr1_dqs1	Data strobe 1 input/output for byte 1 of the 32-bit data bus. This signal is output to the EMIF1 memory when writing and input when reading.	IO	Y24
ddr1_dqsn1	Data strobe 1 invert	IO	Y25
ddr1_dqs2	Data strobe 2 input/output for byte 2 of the 32-bit data bus. This signal is output to the EMIF1 memory when writing and input when reading.	IO	V24
ddr1_dqsn2	Data strobe 2 invert	IO	V25

Table 4-6. EMIF Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
ddr1_dqs3	Data strobe 3 input/output for byte 3 of the 32-bit data bus. This signal is output to the EMIF1 memory when writing and input when reading.	IO	R24
ddr1_dqsn3	Data strobe 3 invert	IO	R25
ddr1_vref0	Reference Power Supply EMIF1	A	Y20

4.3.6 GPMC

NOTE

For more information, see the Memory Subsystem / General-Purpose Memory Controller section of the device TRM.

Table 4-7. GPMC Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
gpmc_ad0	GPMC Data 0 in A/D nonmultiplexed mode and additionally Address 1 in A/D multiplexed mode	IO	F1
gpmc_ad1	GPMC Data 1 in A/D nonmultiplexed mode and additionally Address 2 in A/D multiplexed mode	IO	E2
gpmc_ad2	GPMC Data 2 in A/D nonmultiplexed mode and additionally Address 3 in A/D multiplexed mode	IO	E1
gpmc_ad3	GPMC Data 3 in A/D nonmultiplexed mode and additionally Address 4 in A/D multiplexed mode	IO	C1
gpmc_ad4	GPMC Data 4 in A/D nonmultiplexed mode and additionally Address 5 in A/D multiplexed mode	IO	D1
gpmc_ad5	GPMC Data 5 in A/D nonmultiplexed mode and additionally Address 6 in A/D multiplexed mode	IO	D2
gpmc_ad6	GPMC Data 6 in A/D nonmultiplexed mode and additionally Address 7 in A/D multiplexed mode	IO	B1
gpmc_ad7	GPMC Data 7 in A/D nonmultiplexed mode and additionally Address 8 in A/D multiplexed mode	IO	B2
gpmc_ad8	GPMC Data 8 in A/D nonmultiplexed mode and additionally Address 9 in A/D multiplexed mode	IO	C2
gpmc_ad9	GPMC Data 9 in A/D nonmultiplexed mode and additionally Address 10 in A/D multiplexed mode	IO	D3
gpmc_ad10	GPMC Data 10 in A/D nonmultiplexed mode and additionally Address 11 in A/D multiplexed mode	IO	A2
gpmc_ad11	GPMC Data 11 in A/D nonmultiplexed mode and additionally Address 12 in A/D multiplexed mode	IO	B3
gpmc_ad12	GPMC Data 12 in A/D nonmultiplexed mode and additionally Address 13 in A/D multiplexed mode	IO	C3
gpmc_ad13	GPMC Data 13 in A/D nonmultiplexed mode and additionally Address 14 in A/D multiplexed mode	IO	C4
gpmc_ad14	GPMC Data 14 in A/D nonmultiplexed mode and additionally Address 15 in A/D multiplexed mode	IO	A3
gpmc_ad15	GPMC Data 15 in A/D nonmultiplexed mode and additionally Address 16 in A/D multiplexed mode	IO	B4
gpmc_a0	GPMC Address 0. Only used to effectively address 8-bit data nonmultiplexed memories	O	G1, M1
gpmc_a1	GPMC address 1 in A/D nonmultiplexed mode and Address 17 in A/D multiplexed mode	O	G3, M2
gpmc_a2	GPMC address 2 in A/D nonmultiplexed mode and Address 18 in A/D multiplexed mode	O	H5, L2
gpmc_a3	GPMC address 3 in A/D nonmultiplexed mode and Address 19 in A/D multiplexed mode	O	H6, L1

Table 4-7. GPMC Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
gpmc_a4	GPMC address 4 in A/D nonmultiplexed mode and Address 20 in A/D multiplexed mode	O	K3
gpmc_a5	GPMC address 5 in A/D nonmultiplexed mode and Address 21 in A/D multiplexed mode	O	K2
gpmc_a6	GPMC address 6 in A/D nonmultiplexed mode and Address 22 in A/D multiplexed mode	O	J1
gpmc_a7	GPMC address 7 in A/D nonmultiplexed mode and Address 23 in A/D multiplexed mode	O	K1
gpmc_a8	GPMC address 8 in A/D nonmultiplexed mode and Address 24 in A/D multiplexed mode	O	K4
gpmc_a9	GPMC address 9 in A/D nonmultiplexed mode and Address 25 in A/D multiplexed mode	O	H1
gpmc_a10	GPMC address 10 in A/D nonmultiplexed mode and Address 26 in A/D multiplexed mode	O	J2
gpmc_a11	GPMC address 11 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	L3
gpmc_a12	GPMC address 12 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	G1
gpmc_a13	GPMC address 13 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	A4, H3, G4
gpmc_a14	GPMC address 14 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	E7, H4, G3
gpmc_a15	GPMC address 15 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	D6, K6, F6
gpmc_a16	GPMC address 16 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	C5, K5, M1
gpmc_a17	GPMC address 17 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	B5, G2, D8
gpmc_a18	GPMC address 18 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	D7, F2, C7
gpmc_a19	GPMC address 19 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	A4(3), C6, H5
gpmc_a20	GPMC address 20 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	A5, E7(3), L4
gpmc_a21	GPMC address 21 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	B6, D6(3), H2
gpmc_a22	GPMC address 22 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	A6, C5(3), H6
gpmc_a23	GPMC address 23 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	B5, H5, C10, G4
gpmc_a24	GPMC address 24 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	D7(3), D10, G3
gpmc_a25	GPMC address 25 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	C6(3), F6, E10
gpmc_a26	GPMC address 26 in A/D nonmultiplexed mode and unused in A/D multiplexed mode	O	A5(3), M1, B10
gpmc_a27	GPMC address 27 in A/D nonmultiplexed mode and Address 27 in A/D multiplexed mode	O	B6(3), D8, C7, E8
gpmc_cs0	GPMC Chip Select 0 (active low)	O	F3
gpmc_cs1	GPMC Chip Select 1 (active low)	O	A6
gpmc_cs2	GPMC Chip Select 2 (active low)	O	G4
gpmc_cs3	GPMC Chip Select 3 (active low)	O	G3
gpmc_cs4	GPMC Chip Select 4 (active low)	O	H2
gpmc_cs5	GPMC Chip Select 5 (active low)	O	H6
gpmc_cs6	GPMC Chip Select 6 (active low)	O	H5

Table 4-7. GPMC Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
gpmc_cs7	GPMC Chip Select 7 (active low)	O	L4
gpmc_clk(1)(2)	GPMC Clock output	IO	L4
gpmc_advn_ale	GPMC address valid active low or address latch enable	O	H5
gpmc_oen_ren	GPMC output enable active low or read enable	O	G5
gpmc_wen	GPMC write enable active low	O	G6
gpmc_ben0	GPMC lower-byte enable active low	O	H2
gpmc_ben1	GPMC upper-byte enable active low	O	H6
gpmc_wait0	GPMC external indication of wait 0	I	F6
gpmc_wait1	GPMC external indication of wait 1	I	H5, L4

(1) This clock signal is implemented as 'pad loopback' inside the device - the output signal is looped back through the input buffer to serve as the internal reference signal. Series termination is recommended (as close to device pin as possible) to improve signal integrity of the clock input. Any nonmonotonicity in voltage that occurs at the pad loopback clock pin between V_{IH} and V_{IL} must be less than V_{HYS}.

(2) The gpio6_16.clkout1 signal can be used as an “always-on” alternative to gpmc_clk provided that the external device can support the associated timing. See Table 5-48 *GPMC/NOR Flash Interface Switching Characteristics - Synchronous Mode - Default* and Table 5-50 *GPMC/NOR Flash Interface Switching Characteristics - Synchronous Mode - Alternate* for timing information.

(3) The internal pull resistors for balls A4, E7, D6, C5, D7, C6, A5, B6 are permanently disabled when sysboot15 is set to 0 as described in the section Sysboot Configuration of the Device TRM. If internal pull-up/down resistors are desired on these balls then sysboot15 should be set to 1. If gpmc boot mode is used with SYSBOOT15=0 (not recommended) then external pull-downs should be implemented to keep the address bus at logic-1 value during boot since the gpmc ms-address bits are high-z during boot.

4.3.7 Timers

NOTE

For more information, see the Timers section of the device TRM.

Table 4-8. Timers Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
timer1	PWM output/event trigger input	IO	H21, H6
timer2	PWM output/event trigger input	IO	H2, K22
timer3	PWM output/event trigger input	IO	H5, K23
timer4	PWM output/event trigger input	IO	A16, L4
timer5	PWM output/event trigger input	IO	A18, K6
timer6	PWM output/event trigger input	IO	B17, H4
timer7	PWM output/event trigger input	IO	B16, H3
timer8	PWM output/event trigger input	IO	B18, G1
timer9	PWM output/event trigger input	IO	A19, L3
timer10	PWM output/event trigger input	IO	E17, J2
timer11	PWM output/event trigger input	IO	E16, H1
timer12	PWM output/event trigger input	IO	F16, K4
timer13	PWM output/event trigger input	IO	J25
timer14	PWM output/event trigger input	IO	J24
timer15	PWM output/event trigger input	IO	AA6, H24
timer16	PWM output/event trigger input	IO	AD3, H25

4.3.8 I²C

NOTE

For more information, see the Serial Communication Interface / Multimaster High-Speed I2C Controller / HS I2C Environment / HS I2C in I2C Mode section of the device TRM.

NOTE

I²C1 and I²C2 do NOT support HS-mode.

Table 4-9. I²C Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
Inter-Integrated Circuit Interface 1 (I2C1)			
i2c1_scl	I2C1 Clock	IOD	G22
i2c1_sda	I2C1 Data	IOD	G23
Inter-Integrated Circuit Interface 2 (I2C2)			
i2c2_scl	I2C2 Clock	IOD	G21
i2c2_sda	I2C2 Data	IOD	F23
Inter-Integrated Circuit Interface 3 (I2C3)			
i2c3_scl	I2C3 Clock	IOD	C17, K22, L4, Y6
i2c3_sda	I2C3 Data	IOD	C16, H21, H5, Y5
Inter-Integrated Circuit Interface 4 (I2C4)			
i2c4_scl	I2C4 Clock	IOD	B25, D17, M1, V5
i2c4_sda	I2C4 Data	IOD	C23, D16, M2, U5
Inter-Integrated Circuit Interface 5 (I2C5)			
i2c5_scl	I2C5 Clock	IOD	B14, K3, U6
i2c5_sda	I2C5 Data	IOD	AC3, D14, K2
Inter-Integrated Circuit Interface 6 (I2C6)			
i2c6_scl(1)	I2C6 Clock	IOD	A24
i2c6_sda(1)	I2C6 Data	IOD	D23

(1) I2C6 is not supported in TI standard software. I2C6 is not recommended for use to due to internal clock/reset dependencies on i2c1-5 and uart7.

4.3.9 HDQ1W

NOTE

For more information, see the Serial Communication Interface / HDQ/1-Wire section of the device TRM.

Table 4-10. HDQ / 1-Wire Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
hdq0	HDQ or 1-wire protocol single interface pin	IOD	H25, J25

4.3.10 UART

NOTE

For more information about UART booting, see the Initialization / Device Initialization by ROM Code / Perypheral Booting / Initialization Phase for UART Boot section of the device TRM.

Table 4-11. UART Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
Universal Asynchronous Receiver/Transmitter 1 (UART1)			
uart1_dcdn	UART1 Data Carrier Detect active low	I	N23
uart1_dsrn	UART1 Data Set Ready Active Low	I	N25
uart1_dtrn	UART1 Data Terminal Ready Active Low	O	N22
uart1_rin	UART1 Ring Indicator	I	N24
uart1_rxd	UART1 Receive Data	I	L25
uart1_txd	UART1 Transmit Data	O	M25
uart1_ctsn	UART1 clear to send active low	I	L20
uart1_rtsn	UART1 request to send active low	O	M24
Universal Asynchronous Receiver/Transmitter 2 (UART2)			
uart2_rxd	UART2 Receive Data	I	N23
uart2_txd	UART2 Transmit Data	O	N25
uart2_ctsn	UART2 clear to send active low	I	N22
uart2_rtsn	UART2 request to send active low	O	N24
Universal Asynchronous Receiver/Transmitter 3 (UART3)/IrDA			
uart3_rxd	UART3 Receive Data	I	AA5, G25, N22, N5
uart3_txd	UART3 Transmit Data	O	AC4, F25, N24, N6
uart3_ctsn	UART3 clear to send active low	I	G24, L6, N23, T4
uart3_rtsn	UART3 request to send active low	O	F24, L5, N25, T5
uart3_rctx	Remote control data	O	N23
uart3_sd	Infrared transceiver configure/shutdown	O	N25
uart3_irtx	Infrared data output	O	N24
Universal Asynchronous Receiver/Transmitter 4 (UART4)			
uart4_rxd	UART4 Receive Data	I	A24, E24, P4
uart4_txd	UART4 Transmit Data	O	D23, E25, P3
uart4_ctsn	UART4 clear to send active low	I	R2
uart4_rtsn	UART4 request to send active low	O	R1
Universal Asynchronous Receiver/Transmitter 5 (UART5)			
uart5_rxd	UART5 Receive Data	I	B22, G24, M1, Y4
uart5_txd	UART5 Transmit Data	O	AA2, B23, F24, M2
uart5_ctsn	UART5 clear to send active low	I	AA3, L2
uart5_rtsn	UART5 request to send active low	O	L1, W2
Universal Asynchronous Receiver/Transmitter 6 (UART6)			
uart6_rxd	UART6 Receive Data	I	D14, K3, U5
uart6_txd	UART6 Transmit Data	O	B14, K2, V5
uart6_ctsn	UART6 clear to send active low	I	C14, J1
uart6_rtsn	UART6 request to send active low	O	B15, K1
Universal Asynchronous Receiver/Transmitter 7 (UART7)			
uart7_rxd	UART7 Receive Data	I	A22, L2
uart7_txd	UART7 Transmit Data	O	A23, L1
uart7_ctsn	UART7 clear to send active low	I	B22
uart7_rtsn	UART7 request to send active low	O	B23
Universal Asynchronous Receiver/Transmitter 8 (UART8)			
uart8_rxd	UART8 Receive Data	I	C23, H22, J1
uart8_txd	UART8 Transmit Data	O	B25, H23, K1
uart8_ctsn	UART8 clear to send active low	I	A24
uart8_rtsn	UART8 request to send active low	O	D23

Table 4-11. UART Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
Universal Asynchronous Receiver/Transmitter 9 (UART9)			
uart9_rxd	UART9 Receive Data	I	AC3, E8, L20
uart9_txd	UART9 Transmit Data	O	B8, M24, U6
uart9_ctsn	UART9 clear to send active low	I	AA5, C8
uart9_rtsn	UART9 request to send active low	O	AC4, B9
Universal Asynchronous Receiver/Transmitter 10 (UART10)			
uart10_rxd	UART10 Receive Data	I	A7, H21, N22, Y3
uart10_txd	UART10 Transmit Data	O	A9, AA1, K22, N24
uart10_ctsn	UART10 clear to send active low	I	A8, AA4
uart10_rtsn	UART10 request to send active low	O	A11, AB1

4.3.11 McSPI

NOTE

For more information, see the Serial Communication Interface / Multichannel Serial Peripheral Interface (McSPI) section of the device TRM.

Table 4-12. SPI Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
Serial Peripheral Interface 1			
spi1_sclk(1)	SPI1 Clock	IO	C24
spi1_d1	SPI1 Data. Can be configured as either MISO or MOSI.	IO	D24
spi1_d0	SPI1 Data. Can be configured as either MISO or MOSI.	IO	D25
spi1_cs0	SPI1 Chip Select	IO	B24
spi1_cs1	SPI1 Chip Select	IO	C25
spi1_cs2	SPI1 Chip Select	IO	E24
spi1_cs3	SPI1 Chip Select	IO	E25
Serial Peripheral Interface 2			
spi2_sclk(1)	SPI2 Clock	IO	G25
spi2_d1	SPI2 Data. Can be configured as either MISO or MOSI.	IO	F25
spi2_d0	SPI2 Data. Can be configured as either MISO or MOSI.	IO	G24
spi2_cs0	SPI2 Chip Select	IO	F24
spi2_cs1	SPI2 Chip Select	IO	C25
spi2_cs2	SPI2 Chip Select	IO	E24
spi2_cs3	SPI2 Chip Select	IO	E25
Serial Peripheral Interface 3			
spi3_sclk(1)	SPI3 Clock	IO	A18, C23, N5, Y1
spi3_d1	SPI3 Data. Can be configured as either MISO or MOSI.	IO	B17, B25, N6, Y4
spi3_d0	SPI3 Data. Can be configured as either MISO or MOSI.	IO	A24, AA2, B16, T4
spi3_cs0	SPI3 Chip Select	IO	AA3, B18, D23, T5
spi3_cs1	SPI3 Chip Select	IO	A19, W2
Serial Peripheral Interface 4			
spi4_sclk(1)	SPI4 Clock	IO	AC3, E8, K4, P4, Y3
spi4_d1	SPI4 Data. Can be configured as either MISO or MOSI.	IO	AA1, B8, H1, P3, U6
spi4_d0	SPI4 Data. Can be configured as either MISO or MOSI.	IO	AA4, AA5, C8, J2, R2

Table 4-12. SPI Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
spi4_cs0	SPI4 Chip Select	IO	AB1, AC4, B9, L3, R1
spi4_cs1	SPI4 Chip Select	IO	G1, N6
spi4_cs2	SPI4 Chip Select	IO	H3, T4
spi4_cs3	SPI4 Chip Select	IO	H4, T5

(1) This clock signal is implemented as 'pad loopback' inside the device - the output signal is looped back through the input buffer to serve as the internal reference signal. Series termination is recommended (as close to device pin as possible) to improve signal integrity of the clock input. Any nonmonotonicity in voltage that occurs at the pad loopback clock pin between V_{IH} and V_{IL} must be less than V_{HYS}.

4.3.12 QSPI

NOTE

For more information about UART booting, see the Initialization / Device Initialization by ROM Code / Memory Booting / SPI/QSPI Flash Devices section of the device TRM.

Table 4-13. QSPI Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
qspi1_sclk	QSPI1 Serial Clock	IO	F2
qspi1_rtclk	QSPI1 Return Clock Input. Must be connected from QSPI1_SCLK on PCB. Refer to PCB Guidelines for QSPI1	I	H3
qspi1_d0	QSPI1 Data[0]. This pin is output data for all commands/writes and for dual read and quad read modes it becomes input data pin during read phase.	IO	K5
qspi1_d1	QSPI1 Data[1]. Input read data in all modes.	IO	G2
qspi1_d2	QSPI1 Data[2]. This pin is used only in quad read mode as input data pin during read phase	IO	K6
qspi1_d3	QSPI1 Data[3]. This pin is used only in quad read mode as input data pin during read phase	IO	H4
qspi1_cs0	QSPI1 Chip Select[0]. This pin is Used for QSPI1 boot modes.	IO	G4
qspi1_cs1	QSPI1 Chip Select[1]	O	G3
qspi1_cs2	QSPI1 Chip Select[2]	O	L1
qspi1_cs3	QSPI1 Chip Select[3]	O	K3

4.3.13 McASP

NOTE

For more information, see the Serial Communication Interface / Multichannel Audio Serial Port (McASP) section of the device TRM.

Table 4-14. McASP Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
Multichannel Audio Serial Port 1			
mcasp1_axr0	McASP1 Transmit/Receive Data	IO	D14
mcasp1_axr1	McASP1 Transmit/Receive Data	IO	B14
mcasp1_axr2	McASP1 Transmit/Receive Data	IO	C14
mcasp1_axr3	McASP1 Transmit/Receive Data	IO	B15
mcasp1_axr4	McASP1 Transmit/Receive Data	IO	A15, J25
mcasp1_axr5	McASP1 Transmit/Receive Data	IO	A14, J24
mcasp1_axr6	McASP1 Transmit/Receive Data	IO	A17, H24
mcasp1_axr7	McASP1 Transmit/Receive Data	IO	A16, H25

Table 4-14. McASP Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
mcasp1_axr8	McASP1 Transmit/Receive Data	IO	A18, H21
mcasp1_axr9	McASP1 Transmit/Receive Data	IO	B17, K22
mcasp1_axr10	McASP1 Transmit/Receive Data	IO	B16, K23
mcasp1_axr11	McASP1 Transmit/Receive Data	IO	B18
mcasp1_axr12	McASP1 Transmit/Receive Data	IO	A19
mcasp1_axr13	McASP1 Transmit/Receive Data	IO	E17
mcasp1_axr14	McASP1 Transmit/Receive Data	IO	E16
mcasp1_axr15	McASP1 Transmit/Receive Data	IO	F16
mcasp1_fsx	McASP1 Transmit Frame Sync	IO	C17
mcasp1_aclkr(1)	McASP1 Receive Bit Clock	IO	D16
mcasp1_fsr	McASP1 Receive Frame Sync	IO	D17
mcasp1_ahclkx	McASP1 Transmit High-Frequency Master Clock	O	J25
mcasp1_aclkx(1)	McASP1 Transmit Bit Clock	IO	C16
Multichannel Audio Serial Port 2			
mcasp2_axr0	McASP2 Transmit/Receive Data	IO	A20
mcasp2_axr1	McASP2 Transmit/Receive Data	IO	B19
mcasp2_axr2	McASP2 Transmit/Receive Data	IO	A21
mcasp2_axr3	McASP2 Transmit/Receive Data	IO	B21
mcasp2_axr4	McASP2 Transmit/Receive Data	IO	B20
mcasp2_axr5	McASP2 Transmit/Receive Data	IO	C19
mcasp2_axr6	McASP2 Transmit/Receive Data	IO	D20
mcasp2_axr7	McASP2 Transmit/Receive Data	IO	C20
mcasp2_axr8	McASP2 Transmit/Receive Data	IO	J25
mcasp2_axr9	McASP2 Transmit/Receive Data	IO	J24
mcasp2_axr10	McASP2 Transmit/Receive Data	IO	H24
mcasp2_axr11	McASP2 Transmit/Receive Data	IO	H25
mcasp2_axr12	McASP2 Transmit/Receive Data	IO	A22
mcasp2_axr13	McASP2 Transmit/Receive Data	IO	A23
mcasp2_axr14	McASP2 Transmit/Receive Data	IO	B22
mcasp2_axr15	McASP2 Transmit/Receive Data	IO	B23
mcasp2_fsx	McASP2 Transmit Frame Sync	IO	D19
mcasp2_ahclkx	McASP2 Transmit High-Frequency Master Clock	O	J24
mcasp2_aclkx(1)	McASP2 Transmit Bit Clock	IO	E19
Multichannel Audio Serial Port 3			
mcasp3_axr0	McASP3 Transmit/Receive Data	IO	B22
mcasp3_axr1	McASP3 Transmit/Receive Data	IO	B23
mcasp3_axr2	McASP3 Transmit/Receive Data	IO	A21
mcasp3_axr3	McASP3 Transmit/Receive Data	IO	B21
mcasp3_fsx	McASP3 Transmit Frame Sync	IO	A23
mcasp3_ahclkx	McASP3 Transmit High-Frequency Master Clock	O	H24
mcasp3_aclkx(1)	McASP3 Transmit Bit Clock	IO	A22
mcasp3_aclkr(1)	McASP3 Receive Bit Clock	IO	A22
mcasp3_fsr	McASP3 Receive Frame Sync	IO	A23
Multichannel Audio Serial Port 4			
mcasp4_axr0	McASP4 Transmit/Receive Data	IO	A24
mcasp4_axr1	McASP4 Transmit/Receive Data	IO	D23
mcasp4_axr2	McASP4 Transmit/Receive Data	IO	A15

Table 4-14. McASP Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
mcasp4_axr3	McASP4 Transmit/Receive Data	IO	A14
mcasp4_fsx	McASP4 Transmit Frame Sync	IO	B25
mcasp4_ahclkx	McASP4 Transmit High-Frequency Master Clock	O	H25
mcasp4_aclkx(1)	McASP4 Transmit Bit Clock	IO	C23
mcasp4_aclkr(1)	McASP4 Receive Bit Clock	IO	C23
mcasp4_fsr	McASP4 Receive Frame Sync	IO	B25
Multichannel Audio Serial Port 5			
mcasp5_axr0	McASP5 Transmit/Receive Data	IO	AA5
mcasp5_axr1	McASP5 Transmit/Receive Data	IO	AC4
mcasp5_axr2	McASP5 Transmit/Receive Data	IO	A17
mcasp5_axr3	McASP5 Transmit/Receive Data	IO	A16
mcasp5_fsx	McASP5 Transmit Frame Sync	IO	U6
mcasp5_ahclkx	McASP5 Transmit High-Frequency Master Clock	O	J25
mcasp5_aclkx(1)	McASP5 Transmit Bit Clock	IO	AC3
mcasp5_aclkr(1)	McASP5 Receive Bit Clock	IO	AC3
mcasp5_fsr	McASP5 Receive Frame Sync	IO	U6
Multichannel Audio Serial Port 6			
mcasp6_axr0	McASP6 Transmit/Receive Data	IO	A18
mcasp6_axr1	McASP6 Transmit/Receive Data	IO	B17
mcasp6_axr2	McASP6 Transmit/Receive Data	IO	C14
mcasp6_axr3	McASP6 Transmit/Receive Data	IO	B15
mcasp6_ahclkx	McASP6 Transmit High-Frequency Master Clock	O	J24
mcasp6_aclkx(1)	McASP6 Transmit Bit Clock	IO	B16
mcasp6_fsx	McASP6 Transmit Frame Sync	IO	B18
mcasp6_aclkr(1)	McASP6 Receive Bit Clock	IO	B16
mcasp6_fsr	McASP6 Receive Frame Sync	IO	B18
Multichannel Audio Serial Port 7			
mcasp7_aclkr(1)	McASP7 Receive Bit Clock I/O	IO	E16
mcasp7_aclkx(1)	McASP7 Transmit Bit Clock I/O	IO	E16
mcasp7_ahclkx	McASP7 Transmit High-Frequency Master Clock	O	H24
mcasp7_axr0	McASP7 Transmit/Receive Data I/O	IO	A19
mcasp7_axr1	McASP7 Transmit/Receive Data I/O	IO	E17
mcasp7_axr2	McASP7 Transmit/Receive Data I/O	IO	D16
mcasp7_axr3	McASP7 Transmit/Receive Data I/O	IO	D17
mcasp7_fsr	McASP7 Receive Frame Sync I/O	IO	F16
mcasp7_fsx	McASP7 Transmit Frame Sync I/O	IO	F16
Multichannel Audio Serial Port 8			
mcasp8_aclkr(1)	McASP8 Receive Bit Clock I/O	IO	D20
mcasp8_aclkx(1)	McASP8 Transmit Bit Clock I/O	IO	D20
mcasp8_ahclkx	McASP8 Transmit High-Frequency Master Clock I/O	O	H25
mcasp8_axr0	McASP8 Transmit/Receive Data I/O	IO	B20
mcasp8_axr1	McASP8 Transmit/Receive Data I/O	IO	C19
mcasp8_fsr	McASP8 Receive Frame Sync I/O	IO	C20
mcasp8_fsx	McASP8 Transmit Frame Sync I/O	IO	C20

(1) This clock signal is implemented as 'pad loopback' inside the device - the output signal is looped back through the input buffer to serve as the internal reference signal. Series termination is recommended (as close to device pin as possible) to improve signal integrity of the clock input. Any nonmonotonicity in voltage that occurs at the pad loopback clock pin between V_{IH} and V_{IL} must be less than V_{HYS}.

4.3.14 USB

NOTE

For more information, see: Serial Communication Interface / SuperSpeed USB DRD Subsystem section of the device TRM.

Table 4-15. Universal Serial Bus Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
Universal Serial Bus 1			
usb1_dm	USB1 USB2.0 differential signal pair (negative)	IODS	AB7
usb1_dp	USB1 USB2.0 differential signal pair (positive)	IODS	AC6
usb1_drvvbus	USB1 Drive VBUS signal	O	AD3
usb_rxn0(1)	USB1 USB3.0 receiver negative lane	IDS	AE5
usb_rxp0(1)	USB1 USB3.0 receiver positive lane	IDS	AD6
usb_txn0(1)	USB1 USB3.0 transmitter negative lane	ODS	AE3
usb_txp0(1)	USB1 USB3.0 transmitter positive lane	ODS	AD4
Universal Serial Bus 2			
usb2_dm	USB2 USB2.0 differential signal pair (negative)	IO	AC5
usb2_dp	USB2 USB2.0 differential signal pair (positive)	IO	AB6
usb2_drvvbus	USB2 Drive VBUS signal	O	AA6
Universal Serial Bus 3			
usb3_ulpi_d0	USB3 - ULPI 8-bit data bus	IODS	R2, W2
usb3_ulpi_d1	USB3 - ULPI 8-bit data bus	IODS	AA3, R1
usb3_ulpi_d2	USB3 - ULPI 8-bit data bus	IO	AA2, N2
usb3_ulpi_d3	USB3 - ULPI 8-bit data bus	IO	P2, Y4
usb3_ulpi_d4	USB3 - ULPI 8-bit data bus	IO	N1, Y1
usb3_ulpi_d5	USB3 - ULPI 8-bit data bus	IO	P1, Y2
usb3_ulpi_d6	USB3 - ULPI 8-bit data bus	IO	N3, Y6
usb3_ulpi_d7	USB3 - ULPI 8-bit data bus	IO	N4, Y5
usb3_ulpi_nxt	USB3 - ULPI next	I	P3, Y3
usb3_ulpi_dir	USB3 - ULPI bus direction	I	AA1, P4
usb3_ulpi_stp	USB3 - ULPI stop	O	AA4, T5
usb3_ulpi_clk	USB3 - ULPI functional clock	I	AB1, T4

(1) Signals are enabled by selecting the correct field in the PCIE_B1C0_MODE_SEL register. There are no CTRL_CORE_PAD* register involved.

4.3.15 PCIe

NOTE

For more information, see the *Serial Communication Interfaces* / *PCIe Controllers* and the *Shared PHY Component Subsystems* / *PCIe Shared PHY Susbsytem* sections of the device TRM.

Table 4-16. PCIe Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
pcie_rxn0	PCIe1_PHY_RX Receive Data Lane 0 (negative) - mapped to PCIe_SS1 only.	IDS	AE6

Table 4-16. PCIe Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
pcie_rxp0	PCIe1_PHY_RX Receive Data Lane 0 (positive) - mapped to PCIe_SS1 only.	IDS	AD7
pcie_txn0	PCIe1_PHY_TX Transmit Data Lane 0 (negative) - mapped to PCIe_SS1 only.	ODS	AE8
pcie_txp0	PCIe1_PHY_TX Transmit Data Lane 0 (positive) - mapped to PCIe_SS1 only.	ODS	AD9
pcie_rxn1	PCIe2_PHY_RX Receive Data Lane 1 (negative) - mapped to either PCIe_SS1 (dual lane- mode) or PCIe_SS2 (single lane- mode)	IDS	AE5
pcie_rxp1	PCIe2_PHY_RX Receive Data Lane 1 (positive) - mapped to either PCIe_SS1 (dual lane- mode) or PCIe_SS2 (single lane- mode)	IDS	AD6
pcie_txn1	PCIe2_PHY_TX Transmit Data Lane 1 (negative) - mapped to either PCIe_SS1 (dual lane- mode) or PCIe_SS2 (single lane- mode)	ODS	AE3
pcie_txp1	PCIe2_PHY_TX Transmit Data Lane 1 (positive) - mapped to either PCIe_SS1 (dual lane- mode) or PCIe_SS2 (single lane- mode)	ODS	AD4
ljcb_clkn	PCIe1_PHY / PCIe2_PHY shared Reference Clock Input / Output Differential Pair (negative)	IODS	AB9
ljcb_clkp	PCIe1_PHY / PCIe2_PHY shared Reference Clock Input / Output Differential Pair (positive)	IODS	AC8

4.3.16 DCAN

NOTE

For more information, see the Serial Communication Interface / DCAN section of the device TRM.

Table 4-17. DCAN Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
DCAN 1			
dcan1_rx	DCAN1 receive data pin	IO	H23, AC10
dcan1_tx	DCAN1 transmit data pin	IO	H22
DCAN 2			
dcan2_rx	DCAN2 receive data pin	IO	E25, K22, AB10
dcan2_tx	DCAN2 transmit data pin	IO	E24, H21

4.3.17 GMAC_SW

NOTE

For more information, see the Serial Communication Interfaces / Ethernet Controller section of the device TRM.

Table 4-18. GMAC Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
rgmii0_rxc	RGMII0 Receive Clock	I	N2
rgmii0_rxctl	RGMII0 Receive Control	I	P2
rgmii0_rxd0	RGMII0 Receive Data	I	N4
rgmii0_rxd1	RGMII0 Receive Data	I	N3
rgmii0_rxd2	RGMII0 Receive Data	I	P1
rgmii0_rxd3	RGMII0 Receive Data	I	N1
rgmii0_txc	RGMII0 Transmit Clock	O	T4
rgmii0_txctl	RGMII0 Transmit Enable	O	T5
rgmii0_txd0	RGMII0 Transmit Data	O	R1

Table 4-18. GMAC Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
rgmii0_txd1	RGMII0 Transmit Data	O	R2
rgmii0_txd2	RGMII0 Transmit Data	O	P3
rgmii0_txd3	RGMII0 Transmit Data	O	P4
rgmii1_rxc	RGMII1 Receive Clock	I	E11
rgmii1_rxctl	RGMII1 Receive Control	I	F11
rgmii1_rxd0	RGMII1 Receive Data	I	D13
rgmii1_rxd1	RGMII1 Receive Data	I	C13
rgmii1_rxd2	RGMII1 Receive Data	I	E13
rgmii1_rxd3	RGMII1 Receive Data	I	B13
rgmii1_txc	RGMII1 Transmit Clock	O	B11
rgmii1_txctl	RGMII1 Transmit Enable	O	D11
rgmii1_txd0	RGMII1 Transmit Data	O	A13
rgmii1_txd1	RGMII1 Transmit Data	O	A12
rgmii1_txd2	RGMII1 Transmit Data	O	B12
rgmii1_txd3	RGMII1 Transmit Data	O	C11
mii1_col	MII1 Collision Detect (Sense) input	I	E13
mii1_crs	MII1 Carrier Sense input	I	C13
mii1_rxclk	MII1 Receive Clock	I	B11
mii1_rxd0	MII1 Receive Data	I	E10
mii1_rxd1	MII1 Receive Data	I	F10
mii1_rxd2	MII1 Receive Data	I	A10
mii1_rxd3	MII1 Receive Data	I	B10
mii1_rxdv	MII1 Receive Data Valid input	I	D11
mii1_rxer	MII1 Receive Data Error input	I	B13
mii1_txclk	MII1 Transmit Clock	I	C11
mii1_txd0	MII1 Transmit Data	O	B12
mii1_txd1	MII1 Transmit Data	O	A12
mii1_txd2	MII1 Transmit Data	O	A13
mii1_txd3	MII1 Transmit Data	O	E11
mii1_txen	MII1 Transmit Data Enable Output	O	D13
mii1_txer	MII1 Transmit Error	O	F11
mii0_col	MII0 Collision Detect (Sense) input	I	L5
mii0_crs	MII0 Carrier Sense input	I	P4
mii0_rxclk	MII0 Receive Clock	I	N6
mii0_rxd0	MII0 Receive Data	I	R1
mii0_rxd1	MII0 Receive Data	I	R2
mii0_rxd2	MII0 Receive Data	I	T5
mii0_rxd3	MII0 Receive Data	I	T4
mii0_rxdv	MII0 Receive Data Valid input	I	N5
mii0_rxer	MII0 Receive Data Error input	I	P3
mii0_txclk	MII0 Transmit Clock	I	N2
mii0_txd0	MII0 Transmit Data	O	N4
mii0_txd1	MII0 Transmit Data	O	N3
mii0_txd2	MII0 Transmit Data	O	N1
mii0_txd3	MII0 Transmit Data	O	P2
mii0_txen	MII0 Transmit Data Enable Output	O	P1
mii0_txer	MII0 Transmit Error	O	L6

Table 4-18. GMAC Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
rmii1_crs	RMII1 Carrier Sense input	I	N5
rmii1_rxd0	RMII1 Receive Data	I	T5
rmii1_rxd1	RMII1 Receive Data	I	T4
rmii1_rxer	RMII1 Receive Data Error input	I	N6
rmii1_txd0	RMII1 Transmit Data	O	N1
rmii1_txd1	RMII1 Transmit Data	O	P2
rmii1_txen	RMII1 Transmit Data Enable output	O	N2
rmii0_crs	RMII0 Carrier Sense input	I	P4
rmii0_rxd0	RMII0 Receive Data	I	R1
rmii0_rxd1	RMII0 Receive Data	I	R2
rmii0_rxer	RMII0 Receive Data Error input	I	P3
rmii0_txd0	RMII0 Transmit Data	O	N4
rmii0_txd1	RMII0 Transmit Data	O	N3
rmii0_txen	RMII0 Transmit Data Enable output	O	P1
mdio_mclk	Management Data Serial Clock	O	D10, E24, L5, Y5
mdio_d	Management Data	IO	C10, E25, L6, Y6

4.3.18 MLB

NOTE

MLB in 6-pin mode may require pull ups/ downs on SIG and DAT bus signals.

For additional details, please consult the MLB bus interface specification.

NOTE

For more information, see the Serial Communication Interface / Media Local Bus (MLB) section of the device TRM.

Table 4-19. MLB Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
mlb_dat	Media Local Bus (MLB) Subsystem data input and output	IO	AC4
mlb_sig	Media Local Bus (MLB) Subsystem signal input and output	IO	AA5
mlb_clk	Media Local Bus (MLB) Subsystem clock	I	AC3
mlbp_clk_n	Media Local Bus (MLB) Subsystem clock differential pair (negative)	IDS	U1
mlbp_clk_p	Media Local Bus (MLB) Subsystem clock differential pair (positive)	IDS	U2
mlbp_dat_n	Media Local Bus (MLB) Subsystem data differential pair (negative)	IODS	T1
mlbp_dat_p	Media Local Bus (MLB) Subsystem data differential pair (positive)	IODS	T2
mlbp_sig_n	Media Local Bus (MLB) Subsystem signal differential pair (negative)	IODS	U4
mlbp_sig_p	Media Local Bus (MLB) Subsystem signal differential pair (positive)	IODS	T3

4.3.19 eMMC/SD/SDIO

NOTE

For more information, see the HS MMC/SDIO section of the device TRM.

Table 4-20. eMMC/SD/SDIO Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
Multi Media Card 1			
mmc1_clk(1)	MMC1 clock	IO	U3
mmc1_cmd	MMC1 command	IO	V4
mmc1_dat0	MMC1 data bit 0	IO	V3
mmc1_dat1	MMC1 data bit 1	IO	V2
mmc1_dat2	MMC1 data bit 2	IO	W1
mmc1_dat3	MMC1 data bit 3	IO	V1
mmc1_sdcd	MMC1 Card Detect	I	U5
mmc1_sdwp	MMC1 Write Protect	I	V5
Multi Media Card 2			
mmc2_clk(1)	MMC2 clock	IO	B5
mmc2_cmd	MMC2 command	IO	A6
mmc2_dat0	MMC2 data bit 0	IO	D7
mmc2_dat1	MMC2 data bit 1	IO	C6
mmc2_dat2	MMC2 data bit 2	IO	A5
mmc2_dat3	MMC2 data bit 3	IO	B6
mmc2_dat4	MMC2 data bit 4	IO	A4
mmc2_dat5	MMC2 data bit 5	IO	E7
mmc2_dat6	MMC2 data bit 6	IO	D6
mmc2_dat7	MMC2 data bit 7	IO	C5
mmc2_sdcd	MMC2 Card Detect	I	H22
mmc2_sdwp	MMC2 Write Protect	I	H23
Multi Media Card 3			
mmc3_clk(1)	MMC3 clock	IO	Y2
mmc3_cmd	MMC3 command	IO	Y1
mmc3_dat0	MMC3 data bit 0	IO	Y4
mmc3_dat1	MMC3 data bit 1	IO	AA2
mmc3_dat2	MMC3 data bit 2	IO	AA3
mmc3_dat3	MMC3 data bit 3	IO	W2
mmc3_dat4	MMC3 data bit 4	IO	Y3
mmc3_dat5	MMC3 data bit 5	IO	AA1
mmc3_dat6	MMC3 data bit 6	IO	AA4
mmc3_dat7	MMC3 data bit 7	IO	AB1
mmc3_sdcd	MMC3 Card Detect	I	E24
mmc3_sdwp	MMC3 Write Protect	I	E25
Multi Media Card 4			
mmc4_clk(1)	MMC4 clock	IO	L20
mmc4_cmd	MMC4 command	IO	M24
mmc4_sdcd	MMC4 Card Detect	I	L25
mmc4_sdwp	MMC4 Write Protect	I	M25
mmc4_dat0	MMC4 data bit 0	IO	N23
mmc4_dat1	MMC4 data bit 1	IO	N25
mmc4_dat2	MMC4 data bit 2	IO	N22
mmc4_dat3	MMC4 data bit 3	IO	N24

(1) By default, this clock signal is implemented as 'pad loopback' inside the device - the output signal is looped back through the input buffer to serve as the internal reference signal. mmc1_clk and mmc2_clk have an optional software programmable setting to use an 'internal loopback clock' instead of the default 'pad loopback clock'. If the 'pad loopback clock' is used, series termination is recommended (as close to device pin as possible) to improve signal integrity of the clock input. Any nonmonotonicity in voltage that occurs at the pad loopback clock pin between V_{IH} and V_{IL} must be less than V_{HYS}.

4.3.20 GPIO

NOTE

For more information, see the General-Purpose Interface section of the device TRM.

Table 4-21. GPIOs Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
GPIO 1			
gpio1_0	General-Purpose Input	I	AC10
gpio1_3	General-Purpose Input	I	AB10
gpio1_4	General-Purpose Input/Output	IO	B20
gpio1_5	General-Purpose Input/Output	IO	C20
gpio1_6	General-Purpose Input/Output	IO	F1
gpio1_7	General-Purpose Input/Output	IO	E2
gpio1_8	General-Purpose Input/Output	IO	E1
gpio1_9	General-Purpose Input/Output	IO	C1
gpio1_10	General-Purpose Input/Output	IO	D1
gpio1_11	General-Purpose Input/Output	IO	D2
gpio1_12	General-Purpose Input/Output	IO	B1
gpio1_13	General-Purpose Input/Output	IO	B2
gpio1_14	General-Purpose Input/Output	IO	H22
gpio1_15	General-Purpose Input/Output	IO	H23
gpio1_16	General-Purpose Input/Output	IO	N22
gpio1_17	General-Purpose Input/Output	IO	N24
gpio1_18	General-Purpose Input/Output	IO	C3
gpio1_19	General-Purpose Input/Output	IO	C4
gpio1_20	General-Purpose Input/Output	IO	A3
gpio1_21	General-Purpose Input/Output	IO	B4
gpio1_22	General-Purpose Input/Output	IO	Y3
gpio1_23	General-Purpose Input/Output	IO	AA1
gpio1_24	General-Purpose Input/Output	IO	AA4
gpio1_25	General-Purpose Input/Output	IO	AB1
gpio1_26	General-Purpose Input/Output	IO	K3
gpio1_27	General-Purpose Input/Output	IO	K2
gpio1_28	General-Purpose Input/Output	IO	J1
gpio1_29	General-Purpose Input/Output	IO	K1
gpio1_30	General-Purpose Input/Output	IO	K4
gpio1_31	General-Purpose Input/Output	IO	H1
GPIO2			
gpio2_0	General-Purpose Input/Output	IO	J2
gpio2_1	General-Purpose Input/Output	IO	L3
gpio2_2	General-Purpose Input/Output	IO	G1
gpio2_3	General-Purpose Input/Output	IO	H3
gpio2_4	General-Purpose Input/Output	IO	H4

Table 4-21. GPIOs Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
gpio2_5	General-Purpose Input/Output	IO	K6
gpio2_6	General-Purpose Input/Output	IO	K5
gpio2_7	General-Purpose Input/Output	IO	G2
gpio2_8	General-Purpose Input/Output	IO	F2
gpio2_9	General-Purpose Input/Output	IO	A4
gpio2_10	General-Purpose Input/Output	IO	E7
gpio2_11	General-Purpose Input/Output	IO	D6
gpio2_12	General-Purpose Input/Output	IO	C5
gpio2_13	General-Purpose Input/Output	IO	B5
gpio2_14	General-Purpose Input/Output	IO	D7
gpio2_15	General-Purpose Input/Output	IO	C6
gpio2_16	General-Purpose Input/Output	IO	A5
gpio2_17	General-Purpose Input/Output	IO	B6
gpio2_18	General-Purpose Input/Output	IO	A6
gpio2_19	General-Purpose Input/Output	IO	F3
gpio2_20	General-Purpose Input/Output	IO	G4
gpio2_21	General-Purpose Input/Output	IO	G3
gpio2_22	General-Purpose Input/Output	IO	L4
gpio2_23	General-Purpose Input/Output	IO	H5
gpio2_24	General-Purpose Input/Output	IO	G5
gpio2_25	General-Purpose Input/Output	IO	G6
gpio2_26	General-Purpose Input/Output	IO	H2
gpio2_27	General-Purpose Input/Output	IO	H6
gpio2_28	General-Purpose Input/Output	IO	F6
gpio2_29	General-Purpose Input/Output	IO	D20
GPIO 3			
gpio3_28	General-Purpose Input/Output	IO	D8
gpio3_29	General-Purpose Input/Output	IO	B7
gpio3_30	General-Purpose Input/Output	IO	C7
gpio3_31	General-Purpose Input/Output	IO	E8
GPIO 4			
gpio4_0	General-Purpose Input/Output	IO	B8
gpio4_1	General-Purpose Input/Output	IO	C8
gpio4_2	General-Purpose Input/Output	IO	B9
gpio4_3	General-Purpose Input/Output	IO	A7
gpio4_4	General-Purpose Input/Output	IO	A9
gpio4_5	General-Purpose Input/Output	IO	A8
gpio4_6	General-Purpose Input/Output	IO	A11
gpio4_7	General-Purpose Input/Output	IO	F10
gpio4_8	General-Purpose Input/Output	IO	A10
gpio4_9	General-Purpose Input/Output	IO	B10
gpio4_10	General-Purpose Input/Output	IO	E10
gpio4_11	General-Purpose Input/Output	IO	D10
gpio4_12	General-Purpose Input/Output	IO	C10
gpio4_13	General-Purpose Input/Output	IO	B11
gpio4_14	General-Purpose Input/Output	IO	D11
gpio4_15	General-Purpose Input/Output	IO	C11

Table 4-21. GPIOs Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
gpio4_16	General-Purpose Input/Output	IO	B12
gpio4_17	General-Purpose Input/Output	IO	B18
gpio4_18	General-Purpose Input/Output	IO	A19
gpio4_24	General-Purpose Input/Output	IO	A12
gpio4_25	General-Purpose Input/Output	IO	A13
gpio4_26	General-Purpose Input/Output	IO	E11
gpio4_27	General-Purpose Input/Output	IO	F11
gpio4_28	General-Purpose Input/Output	IO	B13
gpio4_29	General-Purpose Input/Output	IO	E13
gpio4_30	General-Purpose Input/Output	IO	C13
gpio4_31	General-Purpose Input/Output	IO	D13
GPIO 5			
gpio5_0	General-Purpose Input/Output	IO	D16
gpio5_1	General-Purpose Input/Output	IO	D17
gpio5_2	General-Purpose Input/Output	IO	D14
gpio5_3	General-Purpose Input/Output	IO	B14
gpio5_4	General-Purpose Input/Output	IO	C14
gpio5_5	General-Purpose Input/Output	IO	B15
gpio5_6	General-Purpose Input/Output	IO	A15
gpio5_7	General-Purpose Input/Output	IO	A14
gpio5_8	General-Purpose Input/Output	IO	A17
gpio5_9	General-Purpose Input/Output	IO	A16
gpio5_10	General-Purpose Input/Output	IO	A18
gpio5_11	General-Purpose Input/Output	IO	B17
gpio5_12	General-Purpose Input/Output	IO	B16
gpio5_13	General-Purpose Input/Output	IO	A22
gpio5_14	General-Purpose Input/Output	IO	A23
gpio5_15	General-Purpose Input/Output	IO	L5
gpio5_16	General-Purpose Input/Output	IO	L6
gpio5_17	General-Purpose Input/Output	IO	P5
gpio5_18	General-Purpose Input/Output	IO	N5
gpio5_19	General-Purpose Input/Output	IO	N6
gpio5_20	General-Purpose Input/Output	IO	T4
gpio5_21	General-Purpose Input/Output	IO	T5
gpio5_22	General-Purpose Input/Output	IO	P4
gpio5_23	General-Purpose Input/Output	IO	P3
gpio5_24	General-Purpose Input/Output	IO	R2
gpio5_25	General-Purpose Input/Output	IO	R1
gpio5_26	General-Purpose Input/Output	IO	N2
gpio5_27	General-Purpose Input/Output	IO	P2
gpio5_28	General-Purpose Input/Output	IO	N1
gpio5_29	General-Purpose Input/Output	IO	P1
gpio5_30	General-Purpose Input/Output	IO	N3
gpio5_31	General-Purpose Input/Output	IO	N4
GPIO 6			
gpio6_4	General-Purpose Input/Output	IO	E17
gpio6_5	General-Purpose Input/Output	IO	E16

Table 4-21. GPIOs Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
gpio6_6	General-Purpose Input/Output	IO	F16
gpio6_7	General-Purpose Input/Output	IO	C19
gpio6_8	General-Purpose Input/Output	IO	A21
gpio6_9	General-Purpose Input/Output	IO	B21
gpio6_10	General-Purpose Input/Output	IO	Y5
gpio6_11	General-Purpose Input/Output	IO	Y6
gpio6_12	General-Purpose Input/Output	IO	AD3
gpio6_13	General-Purpose Input/Output	IO	AA6
gpio6_14	General-Purpose Input/Output	IO	H21
gpio6_15	General-Purpose Input/Output	IO	K22
gpio6_16	General-Purpose Input/Output	IO	K23
gpio6_17	General-Purpose Input/Output	IO	J25
gpio6_18	General-Purpose Input/Output	IO	J24
gpio6_19	General-Purpose Input/Output	IO	H24
gpio6_20	General-Purpose Input/Output	IO	H25
gpio6_21	General-Purpose Input/Output	IO	U3
gpio6_22	General-Purpose Input/Output	IO	V4
gpio6_23	General-Purpose Input/Output	IO	V3
gpio6_24	General-Purpose Input/Output	IO	V2
gpio6_25	General-Purpose Input/Output	IO	W1
gpio6_26	General-Purpose Input/Output	IO	V1
gpio6_27	General-Purpose Input/Output	IO	U5
gpio6_28	General-Purpose Input/Output	IO	V5
gpio6_29	General-Purpose Input/Output	IO	Y2
gpio6_30	General-Purpose Input/Output	IO	Y1
gpio6_31	General-Purpose Input/Output	IO	Y4
GPIO 7			
gpio7_0	General-Purpose Input/Output	IO	AA2
gpio7_1	General-Purpose Input/Output	IO	AA3
gpio7_2	General-Purpose Input/Output	IO	W2
gpio7_3	General-Purpose Input/Output	IO	M1
gpio7_4	General-Purpose Input/Output	IO	M2
gpio7_5	General-Purpose Input/Output	IO	L2
gpio7_6	General-Purpose Input/Output	IO	L1
gpio7_7	General-Purpose Input/Output	IO	C24
gpio7_8	General-Purpose Input/Output	IO	D24
gpio7_9	General-Purpose Input/Output	IO	D25
gpio7_10	General-Purpose Input/Output	IO	B24
gpio7_11	General-Purpose Input/Output	IO	C25
gpio7_12	General-Purpose Input/Output	IO	E24
gpio7_13	General-Purpose Input/Output	IO	E25
gpio7_14	General-Purpose Input/Output	IO	G25
gpio7_15	General-Purpose Input/Output	IO	F25
gpio7_16	General-Purpose Input/Output	IO	G24
gpio7_17	General-Purpose Input/Output	IO	F24
gpio7_18	General-Purpose Input/Output	IO	C2
gpio7_19	General-Purpose Input/Output	IO	D3

Table 4-21. GPIOs Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
gpio7_22	General-Purpose Input/Output	IO	L25
gpio7_23	General-Purpose Input/Output	IO	M25
gpio7_24	General-Purpose Input/Output	IO	L20
gpio7_25	General-Purpose Input/Output	IO	M24
gpio7_26	General-Purpose Input/Output	IO	N23
gpio7_27	General-Purpose Input/Output	IO	N25
gpio7_28	General-Purpose Input/Output	IO	A2
gpio7_29	General-Purpose Input/Output	IO	B3
gpio7_30	General-Purpose Input/Output	IO	C17
gpio7_31	General-Purpose Input/Output	IO	C16
GPIO 8			
gpio8_27	General-Purpose Input	I	L23
gpio8_28	General-Purpose Input/Output	IO	J20
gpio8_29	General-Purpose Input/Output	IO	K25
gpio8_30(1)	General-Purpose Input/Output	IO	C21
gpio8_31(1)	General-Purpose Input/Output	IO	C22

(1) gpio8_30 is multiplexed with EMU0 and gpio8_31 is multiplexed with EMU1. These pins will be sampled at reset release by the test and emulation logic. Therefore, if they are used as GPIO pins, they must return to the high state whenever the device enters reset. This can be controlled by logic driven from rstoutn. After the device exits reset (indicated by rstoutn rising), these can return to GPIO mode.

4.3.21 KBD

NOTE

For more information, see Keyboard Controller section of the device TRM.

Table 4-22. Keyboard Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
kbd_row0	Keypad row 0	I	D8
kbd_row1	Keypad row 1	I	B7
kbd_row2	Keypad row 2	I	E8
kbd_row3	Keypad row 3	I	B8
kbd_row4	Keypad row 4	I	C8
kbd_row5	Keypad row 5	I	B9
kbd_row6	Keypad row 6	I	A7
kbd_row7	Keypad row 7	I	C10
kbd_row8	Keypad row 8	I	D11
kbd_col0	Keypad column 0	O	A9
kbd_col1	Keypad column 1	O	A8
kbd_col2	Keypad column 2	O	A11
kbd_col3	Keypad column 3	O	F10
kbd_col4	Keypad column 4	O	A10
kbd_col5	Keypad column 5	O	B10
kbd_col6	Keypad column 6	O	E10
kbd_col7	Keypad column 7	O	D10
kbd_col8	Keypad column 8	O	B11

4.3.22 PWM

NOTE

For more information, see the Pulse-Width Modulation (PWM) SS section of the device TRM.

Table 4-23. PWM Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
PWMSS1			
eCAP1_in_PWM1_out	ECAP1 Capture Input / PWM Output	IO	A7
ehrpwm1_synci	EHRPWM1 Sync Input	I	A9
ehrpwm1_synco	EHRPWM1 Sync Output	O	A8
ehrpwm1_tripzone_input	EHRPWM1 Trip Zone Input	IO	B9
ehrpwm1A	EHRPWM1 Output A	O	B8
ehrpwm1B	EHRPWM1 Output B	O	C8
eQEP1_index	EQEP1 Index Input	IO	C7
eQEP1_strobe	EQEP1 Strobe Input	IO	E8
eQEP1A_in	EQEP1 Quadrature Input A	I	D8
eQEP1B_in	EQEP1 Quadrature Input B	I	B7
PWMSS2			
eCAP2_in_PWM2_out	ECAP2 Capture Input / PWM Output	IO	B11, Y1
ehrpwm2_tripzone_input	EHRPWM2 Trip Zone Input	IO	C10, Y2
ehrpwm2A	EHRPWM2 Output A	O	E10, Y5
ehrpwm2B	EHRPWM2 Output B	O	D10, Y6
eQEP2_index	EQEP2 Index Input	IO	A10
eQEP2_strobe	EQEP2 Strobe Input	IO	B10
eQEP2A_in	EQEP2 Quadrature Input A	I	A11
eQEP2B_in	EQEP2 Quadrature Input B	I	F10
PWMSS3			
eCAP3_in_PWM3_out	ECAP3 Capture Input / PWM Output	IO	AB1, B13
ehrpwm3_tripzone_input	EHRPWM3 Trip Zone Input	IO	AA4, F11
ehrpwm3A	EHRPWM3 Output A	O	A13, Y3
ehrpwm3B	EHRPWM3 Output B	O	AA1, E11
eQEP3_index	EQEP3 Index Input	IO	AA3, B12
eQEP3_strobe	EQEP3 Strobe Input	IO	A12, W2
eQEP3A_in	EQEP3 Quadrature Input A	I	D11, Y4
eQEP3B_in	EQEP3 Quadrature Input B	I	AA2, C11

4.3.23 PRU-ICSS

NOTE

For more information, see the *Serial Communication Interfaces* / *PCIe Controllers* and the *Shared PHY Component Subsystems* / *PCIe Shared PHY Susbsytem* sections of the device TRM.

Table 4-24. PRU-ICSS Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
PRU-ICSS1			
pr1_ecap0_ecap_capin_apwm_o	Capture Input / PWM output	IO	A7
pr1_edc_sync0_out	SYNC 0 Output	O	A8
pr1_edio_data_in0	Ethernet Digital Input	I	D8
pr1_edio_data_in1	Ethernet Digital Input	I	B7
pr1_edio_data_in2	Ethernet Digital Input	I	C7
pr1_edio_data_in3	Ethernet Digital Input	I	E8
pr1_edio_data_in4	Ethernet Digital Input	I	B8
pr1_edio_data_in5	Ethernet Digital Input	I	C8
pr1_edio_data_in6	Ethernet Digital Input	I	B9
pr1_edio_data_in7	Ethernet Digital Input	I	A7
pr1_edio_data_out0	Ethernet Digital Output	O	D8
pr1_edio_data_out1	Ethernet Digital Output	O	B7
pr1_edio_data_out2	Ethernet Digital Output	O	C7
pr1_edio_data_out3	Ethernet Digital Output	O	E8
pr1_edio_data_out4	Ethernet Digital Output	O	B8
pr1_edio_data_out5	Ethernet Digital Output	O	C8
pr1_edio_data_out6	Ethernet Digital Output	O	B9
pr1_edio_data_out7	Ethernet Digital Output	O	A7
pr1_edio_sof	Start Of Frame	O	A11
pr1_mdio_data	MDIO Data	IO	C10
pr1_mdio_mdclk	MDIO Clock	O	D10
pr1_edc_latch0_in	Latch Input 0	I	A9
pr1_mii0_col	MII0 Collision Detect	I	L5
pr1_mii0_crs	MII0 Carrier Sense	I	P4
pr1_mii0_rxd0	MII0 Receive Data	I	R1
pr1_mii0_rxd1	MII0 Receive Data	I	R2
pr1_mii0_rxd2	MII0 Receive Data	I	T5
pr1_mii0_rxd3	MII0 Receive Data	I	T4
pr1_mii0_rxdv	MII0 Data Valid	I	N5
pr1_mii0_rxer	MII0 Receive Error	I	P3
pr1_mii0_rxlink	MII0 Receive Link	I	L6
pr1_mii0_txd0	MII0 Transmit Data	O	N4
pr1_mii0_txd1	MII0 Transmit Data	O	N3
pr1_mii0_txd2	MII0 Transmit Data	O	N1
pr1_mii0_txd3	MII0 Transmit Data	O	P2
pr1_mii0_txen	MII0 Transmit Enable	O	P1
pr1_mii1_col	MII1 Collision Detect	I	C13
pr1_mii1_crs	MII1 Carrier Sense	I	D13
pr1_mii1_rxd0	MII1 Receive Data	I	F11
pr1_mii1_rxd1	MII1 Receive Data	I	E11
pr1_mii1_rxd2	MII1 Receive Data	I	A13
pr1_mii1_rxd3	MII1 Receive Data	I	A12
pr1_mii1_rxdv	MII1 Data Valid	I	B12
pr1_mii1_rxer	MII1 Receive Error	I	B13
pr1_mii1_rxlink	MII1 Receive Link	I	E13

Table 4-24. PRU-ICSS Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
pr1_mii1_txd0	MII1 Transmit Data	O	D11
pr1_mii1_txd1	MII1 Transmit Data	O	B11
pr1_mii1_txd2	MII1 Transmit Data	O	E10
pr1_mii1_txd3	MII1 Transmit Data	O	B10
pr1_mii1_txen	MII1 Transmit Enable	O	A10
pr1_mii_mr0_clk	MII0 Master Receive Clock	I	N6
pr1_mii_mr1_clk	MII1 Master Receive Clock	I	C11
pr1_mii_mt0_clk	MII0 Master Transmit Clock	I	N2
pr1_mii_mt1_clk	MII1 Master Transmit Clock	I	F10
pr1_pru1_gpi0	PRU1 General-Purpose Input	I	A9
pr1_pru1_gpi1	PRU1 General-Purpose Input	I	A8
pr1_pru1_gpi2	PRU1 General-Purpose Input	I	A11
pr1_pru1_gpi3	PRU1 General-Purpose Input	I	F10
pr1_pru1_gpi4	PRU1 General-Purpose Input	I	A10
pr1_pru1_gpi5	PRU1 General-Purpose Input	I	B10
pr1_pru1_gpi6	PRU1 General-Purpose Input	I	E10
pr1_pru1_gpi7	PRU1 General-Purpose Input	I	D10
pr1_pru1_gpi8	PRU1 General-Purpose Input	I	C10
pr1_pru1_gpi9	PRU1 General-Purpose Input	I	B11
pr1_pru1_gpi10	PRU1 General-Purpose Input	I	D11
pr1_pru1_gpi11	PRU1 General-Purpose Input	I	C11
pr1_pru1_gpi12	PRU1 General-Purpose Input	I	B12
pr1_pru1_gpi13	PRU1 General-Purpose Input	I	A12
pr1_pru1_gpi14	PRU1 General-Purpose Input	I	A13
pr1_pru1_gpi15	PRU1 General-Purpose Input	I	E11
pr1_pru1_gpi16	PRU1 General-Purpose Input	I	F11
pr1_pru1_gpi17	PRU1 General-Purpose Input	I	B13
pr1_pru1_gpi18	PRU1 General-Purpose Input	I	E13
pr1_pru1_gpi19	PRU1 General-Purpose Input	I	C13
pr1_pru1_gpi20	PRU1 General-Purpose Input	I	D13
pr1_pru1_gpo0	PRU1 General-Purpose Output	O	A9
pr1_pru1_gpo1	PRU1 General-Purpose Output	O	A8
pr1_pru1_gpo2	PRU1 General-Purpose Output	O	A11
pr1_pru1_gpo3	PRU1 General-Purpose Output	O	F10
pr1_pru1_gpo4	PRU1 General-Purpose Output	O	A10
pr1_pru1_gpo5	PRU1 General-Purpose Output	O	B10
pr1_pru1_gpo6	PRU1 General-Purpose Output	O	E10
pr1_pru1_gpo7	PRU1 General-Purpose Output	O	D10
pr1_pru1_gpo8	PRU1 General-Purpose Output	O	C10
pr1_pru1_gpo9	PRU1 General-Purpose Output	O	B11
pr1_pru1_gpo10	PRU1 General-Purpose Output	O	D11
pr1_pru1_gpo11	PRU1 General-Purpose Output	O	C11
pr1_pru1_gpo12	PRU1 General-Purpose Output	O	B12
pr1_pru1_gpo13	PRU1 General-Purpose Output	O	A12
pr1_pru1_gpo14	PRU1 General-Purpose Output	O	A13
pr1_pru1_gpo15	PRU1 General-Purpose Output	O	E11
pr1_pru1_gpo16	PRU1 General-Purpose Output	O	F11

Table 4-24. PRU-ICSS Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
pr1_pru1_gpo17	PRU1 General-Purpose Output	O	B13
pr1_pru1_gpo18	PRU1 General-Purpose Output	O	E13
pr1_pru1_gpo19	PRU1 General-Purpose Output	O	C13
pr1_pru1_gpo20	PRU1 General-Purpose Output	O	D13
pr1_uart0_cts_n	UART Clear-To-Send	I	E8
pr1_uart0_rts_n	UART Ready-To-Send	O	B8
pr1_uart0_rxd	UART Receive Data	I	C8
pr1_uart0_txd	UART Transmit Data	O	B9
PRU-ICSS 2			
pr2_mdio_data	MDIO Data	IO	AC4, C17
pr2_mdio_mdclk	MDIO Clock	O	AA5, C16
pr2_mii0_col	MII0 Collision Detect	I	A23
pr2_mii0_crs	MII0 Carrier Sense	I	A22
pr2_mii0_rxd0	MII0 Receive Data	I	A21
pr2_mii0_rxd1	MII0 Receive Data	I	D19
pr2_mii0_rxd2	MII0 Receive Data	I	E19
pr2_mii0_rxd3	MII0 Receive Data	I	F16
pr2_mii0_rxdv	MII0 Data Valid	I	E16
pr2_mii0_rxer	MII0 Receive Error	I	D14
pr2_mii0_rxlink	MII0 Receive Link	I	B21
pr2_mii0_txd0	MII0 Transmit Data	O	A19
pr2_mii0_txd1	MII0 Transmit Data	O	B18
pr2_mii0_txd2	MII0 Transmit Data	O	B16
pr2_mii0_txd3	MII0 Transmit Data	O	B17
pr2_mii0_txen	MII0 Transmit Enable	O	A18
pr2_mii1_col	MII1 Collision Detect	I	J25
pr2_mii1_crs	MII1 Carrier Sense	I	J24
pr2_mii1_rxd0	MII1 Receive Data	I	AB1
pr2_mii1_rxd1	MII1 Receive Data	I	AA4
pr2_mii1_rxd2	MII1 Receive Data	I	AA1
pr2_mii1_rxd3	MII1 Receive Data	I	Y3
pr2_mii1_rxdv	MII1 Data Valid	I	W2
pr2_mii1_rxer	MII1 Receive Error	I	B22
pr2_mii1_rxlink	MII1 Receive Link	I	B23
pr2_mii1_txd0	MII1 Transmit Data	O	AA2
pr2_mii1_txd1	MII1 Transmit Data	O	Y4
pr2_mii1_txd2	MII1 Transmit Data	O	Y1
pr2_mii1_txd3	MII1 Transmit Data	O	Y2
pr2_mii1_txen	MII1 Transmit Enable	O	Y6
pr2_mii_mr0_clk	MII0 Master Receive Clock	I	E17
pr2_mii_mr1_clk	MII1 Master Receive Clock	I	AA3
pr2_mii_mt0_clk	MII0 Master Transmit Clock	I	B14
pr2_mii_mt1_clk	MII1 Master Transmit Clock	I	Y5
pr2_pru0_gpi0	PRU0 General-Purpose Input	I	Y5
pr2_pru0_gpi1	PRU0 General-Purpose Input	I	Y6
pr2_pru0_gpi2	PRU0 General-Purpose Input	I	Y2
pr2_pru0_gpi3	PRU0 General-Purpose Input	I	Y1

Table 4-24. PRU-ICSS Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
pr2_pru0_gpi4	PRU0 General-Purpose Input	I	Y4
pr2_pru0_gpi5	PRU0 General-Purpose Input	I	AA2
pr2_pru0_gpi6	PRU0 General-Purpose Input	I	AA3
pr2_pru0_gpi7	PRU0 General-Purpose Input	I	W2
pr2_pru0_gpi8	PRU0 General-Purpose Input	I	Y3
pr2_pru0_gpi9	PRU0 General-Purpose Input	I	AA1
pr2_pru0_gpi10	PRU0 General-Purpose Input	I	AA4
pr2_pru0_gpi11	PRU0 General-Purpose Input	I	AB1
pr2_pru0_gpi12	PRU0 General-Purpose Input	I	A22
pr2_pru0_gpi13	PRU0 General-Purpose Input	I	A23
pr2_pru0_gpi14	PRU0 General-Purpose Input	I	B22
pr2_pru0_gpi15	PRU0 General-Purpose Input	I	B23
pr2_pru0_gpi16	PRU0 General-Purpose Input	I	A21
pr2_pru0_gpi17	PRU0 General-Purpose Input	I	B21
pr2_pru0_gpi18	PRU0 General-Purpose Input	I	E19
pr2_pru0_gpi19	PRU0 General-Purpose Input	I	D19
pr2_pru0_gpi20	PRU0 General-Purpose Input	I	F16
pr2_pru0_gpo0	PRU0 General-Purpose Output	O	Y5
pr2_pru0_gpo1	PRU0 General-Purpose Output	O	Y6
pr2_pru0_gpo2	PRU0 General-Purpose Output	O	Y2
pr2_pru0_gpo3	PRU0 General-Purpose Output	O	Y1
pr2_pru0_gpo4	PRU0 General-Purpose Output	O	Y4
pr2_pru0_gpo5	PRU0 General-Purpose Output	O	AA2
pr2_pru0_gpo6	PRU0 General-Purpose Output	O	AA3
pr2_pru0_gpo7	PRU0 General-Purpose Output	O	W2
pr2_pru0_gpo8	PRU0 General-Purpose Output	O	Y3
pr2_pru0_gpo9	PRU0 General-Purpose Output	O	AA1
pr2_pru0_gpo10	PRU0 General-Purpose Output	O	AA4
pr2_pru0_gpo11	PRU0 General-Purpose Output	O	AB1
pr2_pru0_gpo12	PRU0 General-Purpose Output	O	A22
pr2_pru0_gpo13	PRU0 General-Purpose Output	O	A23
pr2_pru0_gpo14	PRU0 General-Purpose Output	O	B22
pr2_pru0_gpo15	PRU0 General-Purpose Output	O	B23
pr2_pru0_gpo16	PRU0 General-Purpose Output	O	A21
pr2_pru0_gpo17	PRU0 General-Purpose Output	O	B21
pr2_pru0_gpo18	PRU0 General-Purpose Output	O	E19
pr2_pru0_gpo19	PRU0 General-Purpose Output	O	D19
pr2_pru0_gpo20	PRU0 General-Purpose Output	O	F16
pr2_pru1_gpi0	PRU1 General-Purpose Input	I	D23, L5
pr2_pru1_gpi1	PRU1 General-Purpose Input	I	AC3, L6
pr2_pru1_gpi2	PRU1 General-Purpose Input	I	U6, P5
pr2_pru1_gpi3	PRU1 General-Purpose Input	I	AA5, N5
pr2_pru1_gpi4	PRU1 General-Purpose Input	I	AC4, N6
pr2_pru1_gpi5	PRU1 General-Purpose Input	I	J25, T4
pr2_pru1_gpi6	PRU1 General-Purpose Input	I	J24, T5
pr2_pru1_gpi7	PRU1 General-Purpose Input	I	C16, P4
pr2_pru1_gpi8	PRU1 General-Purpose Input	I	D14, P3

Table 4-24. PRU-ICSS Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
pr2_pru1_gpi9	PRU1 General-Purpose Input	I	B14, R2
pr2_pru1_gpi10	PRU1 General-Purpose Input	I	A18, R1
pr2_pru1_gpi11	PRU1 General-Purpose Input	I	B17, N2
pr2_pru1_gpi12	PRU1 General-Purpose Input	I	B16, P2
pr2_pru1_gpi13	PRU1 General-Purpose Input	I	B18, N1
pr2_pru1_gpi14	PRU1 General-Purpose Input	I	A19, P1
pr2_pru1_gpi15	PRU1 General-Purpose Input	I	E17, N3
pr2_pru1_gpi16	PRU1 General-Purpose Input	I	E16, N4
pr2_pru1_gpo0	PRU1 General-Purpose Output	O	D23, L5
pr2_pru1_gpo1	PRU1 General-Purpose Output	O	AC3, L6
pr2_pru1_gpo2	PRU1 General-Purpose Output	O	U6, P5
pr2_pru1_gpo3	PRU1 General-Purpose Output	O	AA5, N5
pr2_pru1_gpo4	PRU1 General-Purpose Output	O	AC4, N6
pr2_pru1_gpo5	PRU1 General-Purpose Output	O	J25, T4
pr2_pru1_gpo6	PRU1 General-Purpose Output	O	J24, T5
pr2_pru1_gpo7	PRU1 General-Purpose Output	O	C16, P4
pr2_pru1_gpo8	PRU1 General-Purpose Output	O	D14, P3
pr2_pru1_gpo9	PRU1 General-Purpose Output	O	B14, R2
pr2_pru1_gpo10	PRU1 General-Purpose Output	O	A18, R1
pr2_pru1_gpo11	PRU1 General-Purpose Output	O	B17, N2
pr2_pru1_gpo12	PRU1 General-Purpose Output	O	B16, P2
pr2_pru1_gpo13	PRU1 General-Purpose Output	O	B18, N1
pr2_pru1_gpo14	PRU1 General-Purpose Output	O	A19, P1
pr2_pru1_gpo15	PRU1 General-Purpose Output	O	E17, N3
pr2_pru1_gpo16	PRU1 General-Purpose Output	O	E16, N4

4.3.24 ATL

NOTE

For more information, see the Audio Tracking Logic (ATL) section of the device TRM.

Table 4-25. ATL Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
atl_clk0	Audio Tracking Logic Clock 0	O	J25
atl_clk1	Audio Tracking Logic Clock 1	O	J24
atl_clk2	Audio Tracking Logic Clock 2	O	H24
atl_clk3	Audio Tracking Logic Clock 3	O	H25

4.3.25 Emulation and Debug Subsystem

NOTE

For more information, see the On-Chip Debug Support / Debug Ports section of the device TRM.

Table 4-26. Debug Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
tms	JTAG test port mode select. An external pullup resistor should be used on this ball.	IO	L21
tdi	JTAG test data	I	L23
tdo	JTAG test port data	O	J20
tclk	JTAG test clock	I	K21
trstn	JTAG test reset	I	L22
rtck	JTAG return clock	O	K25
emu0(1)	Emulator pin 0	IO	C21
emu1(1)	Emulator pin 1	IO	C22
emu2	Emulator pin 2	IO	E14
emu3	Emulator pin 3	IO	F14
emu4	Emulator pin 4	IO	F13
emu5	Emulator pin 5	O	D8
emu6	Emulator pin 6	O	B7
emu7	Emulator pin 7	O	C7
emu8	Emulator pin 8	O	E8
emu9	Emulator pin 9	O	B8
emu10	Emulator pin 10	O	C8
emu11	Emulator pin 11	O	B9
emu12	Emulator pin 12	O	A7
emu13	Emulator pin 13	O	A9
emu14	Emulator pin 14	O	A8
emu15	Emulator pin 15	O	A11
emu16	Emulator pin 16	O	F10
emu17	Emulator pin 17	O	A10
emu18	Emulator pin 18	O	B10
emu19	Emulator pin 19	O	E10

(1) EMU0 and EMU1 are multiplexed with GPIO. These pins will be sampled at reset release by the test and emulation logic. Therefore, if they are used as GPIO pins, they must return to the high state whenever the device enters reset. This can be controlled by logic driven from rstoutn. After the device exits reset (indicated by rstoutn rising), these can return to GPIO mode.

4.3.26 System and Miscellaneous

4.3.26.1 Sysboot

NOTE

For more information, see the Initialization (ROM Code) section of the device TRM.

Table 4-27. Sysboot Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
sysboot0	Boot Mode Configuration 0. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	F1
sysboot1	Boot Mode Configuration 1. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	E2
sysboot2	Boot Mode Configuration 2. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	E1
sysboot3	Boot Mode Configuration 3. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	C1

Table 4-27. Sysboot Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
sysboot4	Boot Mode Configuration 4. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	D1
sysboot5	Boot Mode Configuration 5. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	D2
sysboot6	Boot Mode Configuration 6. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	B1
sysboot7	Boot Mode Configuration 7. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	B2
sysboot8	Boot Mode Configuration 8. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	C2
sysboot9	Boot Mode Configuration 9. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	D3
sysboot10	Boot Mode Configuration 10. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	A2
sysboot11	Boot Mode Configuration 11. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	B3
sysboot12	Boot Mode Configuration 12. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	C3
sysboot13	Boot Mode Configuration 13. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	C4
sysboot14	Boot Mode Configuration 14. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	A3
sysboot15	Boot Mode Configuration 15. The value latched on this pin upon porz reset release will determine the boot mode configuration of the device.	I	B4

4.3.26.2 Power, Reset, and Clock Management (PRCM)

NOTE

For more information, see PRCM section of the device TRM.

Table 4-28. PRCM Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
clkout1	Device Clock output 1. Can be used externally for devices with non-critical timing requirements, or for debug, or as a reference clock on GPMC as described in Table 5-48 *GPMC/NOR Flash Interface Switching Characteristics - Synchronous Mode - Default* and Table 5-50 *GPMC/NOR Flash Interface Switching Characteristics - Synchronous Mode - Alternate*.	O	K23, L4
clkout2	Device Clock output 2. Can be used externally for devices with non-critical timing requirements, or for debug.	O	H5, J25
clkout3	Device Clock output 3. Can be used xternally for devices with non-critical timing requirements, or for debug.	O	H25
porz	Power on Reset (active low) input must be asserted low during a device power up sequence or cold reset state when all supplies are disabled. Typically, an external PMIC is the source and sets porz high after all supplies reach valid operating levels. Asserting porz low puts the entire device in a safe reset state.	I	F19
resetn	Reset (active low) input's falling edge can trigger a device warm reset state from an external component. This signal should be high prior to or simultaneous with, porz rising. If the signal is not used in the system, resetn should be pulled high with an external pull-up resistor to vddshv3.	I	K24

Table 4-28. PRCM Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
rstoutn	Reset out (Active low) output is asserted low whenever any global reset condition exists. After a brief delay, it will be set high upon removal of the internal global reset condition (i.e. porz, warm reset). It is only functional after its output buffer's reference voltage (vddshv3) is valid. If it is used as a reset for device peripheral components, then it should be AND gated with porz to avoid the possibility of reset signal glitches during a power up sequence.(2)	O	E20
xi_osc0	System Oscillator OSC0 Crystal input / LVCMOS clock input. Functions as the input connection to a crystal when the internal oscillator OSC0 is used. Functions as an LVCMOS-compatible input clock when an external oscillator is used.	I	Y12
xi_osc1	Auxiliary Oscillator OSC1 Crystal input / LVCMOS clock input. Functions as the input connection to a crystal when the internal oscillator OSC1 is used. Functions as an LVCMOS-compatible input clock when an external oscillator is used	I	AC11
xo_osc0	System Oscillator OSC0 Crystal output	O	AB12
xo_osc1	Auxiliary Oscillator OSC1 Crystal output	O	AA11
xref_clk0	External Reference Clock 0. For Audio and other Peripherals.	I	J25
xref_clk1	External Reference Clock 1. For Audio and other Peripherals.	I	J24
xref_clk2	External Reference Clock 2. For Audio and other Peripherals.	I	H24
xref_clk3	External Reference Clock 3. For Audio and other Peripherals.	I	H25
RMII_MHZ_50_CLK(1)	RMII Reference Clock (50MHz). This pin is an input when external reference is used or output when internal reference is used.	IO	P5

(1) This clock signal is implemented as 'pad loopback' inside the device - the output signal is looped back through the input buffer to serve as the internal reference signal. Series termination is recommended (as close to device pin as possible) to improve signal integrity of the clock input. Any nonmonotonicity in voltage that occurs at the pad loopback clock pin between V_{IH} and V_{IL} must be less than V_{HYS}.

(2) Note that rstoutn is only valid after vddshv3 is valid. If the rstoutn signal will be used as a reset into other devices attached to the SOC, it must be AND'ed with porz. This will prevent glitches occurring during supply ramping being propagated.

4.3.26.3 System Direct Memory Access (SDMA)

NOTE

For more information, see the DMA Controllers / System DMA section of the device TRM.

Table 4-29. SDMA Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
dma_evt1	System DMA Event Input 1	I	G1, L4
dma_evt2	System DMA Event Input 2	I	H3, H5
dma_evt3	System DMA Event Input 3	I	H2
dma_evt4	System DMA Event Input 4	I	H6

4.3.26.4 Interrupt Controllers (INTC)

NOTE

For more information, see the Interrupt Controllers section of the device TRM.

Table 4-30. INTC Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
nmin_dsp	Non maskable interrupt input, active-low. This pin can be optionally routed to the DSP NMI input or as generic input to the Arm cores. Note that by default this pin has an internal pulldown resistor enabled. This internal pulldown should be disabled or countered by a stronger external pullup resistor before routing to the DSP or Arm processors.	I	L24
sys_nirq2	External interrupt event to any device INTC	I	AC10
sys_nirq1	External interrupt event to any device INTC	I	AB10

4.3.27 Power Supplies

NOTE

For more information, see Power, Reset, and Clock Management / PRCM Subsystem Environment / External Voltage Inputs section of the device TRM.

Table 4-31. Power Supply Signal Descriptions

SIGNAL NAME	DESCRIPTION	TYPE	BALL
vdd	Core voltage domain supply	PWR	J15, J16, J18, K12, K18, L12, L17, M11, M13, M15, M17, N11, N13, N15, N18, P10, P12, P14, P16, P18, R10, R12, R14, R16, R17, T11, T13, T15, T17, T9, U11, U13, U15, U18, U9, V10, V12, V14, V16, V18, W10, W12, W14, W16
vpp(2)	eFuse power supply	PWR	F20
vss	Ground	GND	A1, A25, AA13, AA15, AA7, AA8, AA9, AB8, AC13, AE1, AE15, AE25, G13, G16, G8, H10, H12, H14, H16, H18, H19, H8, J10, J12, J14, J17, K11, K13, K15, K17, K9, L11, L13, L15, L18, L8, M12, M14, M16, M18, M20, M8, M9, N12, N14, N16, N17, N20, P11, P13, P15, P17, P19, P9, R11, R13, R15, R18, R19, R8, R9, T10, T12, T14, T16, T18, T8, U10, U12, U14, U16, U17, U19, V11, V13, V15, V17, V19, V8, V9, W19, W9, Y14, Y16, Y17, Y7
cap_vbbldo_gpu (1)	MM (SGX) Back bias supply	CAP	T7
cap_vbbldo_iva (1)	IVA Back bias supply	CAP	G14
cap_vbbldo_mpu (1)	MPU back bias supply	CAP	F17
cap_vbbldo_dsp (1)	External capacitor connection for the DSP vbb ldo output	CAP	F8
cap_vddram_core1(1)	SRAM array supply for core memories	CAP	U20
cap_vddram_core3 (1)	SRAM array supply for core memories	CAP	K7
cap_vddram_core4 (1)	SRAM array supply for core memories	CAP	G19
cap_vddram_gpu (1)	SRAM array supply for SGX (MM) memories	CAP	V7

Table 4-31. Power Supply Signal Descriptions (continued)

SIGNAL NAME	DESCRIPTION	TYPE	BALL
cap_vddram_iva (1)	SRAM array supply for IVA memories	CAP	G12
cap_vddram_dsp (1)	External capacitor connection for the DSP	CAP	L7
cap_vddram_mpu (1)	External capacitor connection for the MPU SRAM array ldo output	CAP	G18
vdda33v_usb1	HS USB1 3p3 supply	PWR	AA10
vdda33v_usb2	HS USB1 3p3 supply	PWR	Y10
vdda_core_gmac	DPLL_CORE and CORE HSDIVIDER analog power supply	PWR	L9
vdda_csi	CSI Interface 1.8v Supply	PWR	T6
vdda_dsp_iva	DSP PLL and IVA PLL analog power supply	PWR	K10, L10
vdda_mpu_abe	MPU_ABE PLL analog power supply	PWR	K16, L16
vdda_per	DPLL_ABE, DPLL_PER, and PER HSDIVIDER analog power supply	PWR	M10
vdda_usb2	HS USB2 1.8V analog power supply	PWR	Y8
vdds_mlbp	MLBP IO power supply	PWR	P7, R7
vdd_dsp	DSP voltage domain supply	PWR	H11, H13, H9, J11, J13, J9
vdda_ddr	DDR PLL and DDR HSDIVIDER analog power supply	PWR	R20
vdda_debug	Debug PLL inside IOSC PLL supply	PWR	N10
vdda_gpu	GPU (SGX) PLL analog power supply	PWR	N9
vdda_hdmi	HDMI PLL and HDMI analog power supply	PWR	W15, Y15
vdda_osc	HFOSC - 1.8v vdds supply	PWR	W13, Y13
vdda_pcie	PCIe PLL analog power supply	PWR	W11, Y11
vdda_usb1	USB2 PLL analog power supply	PWR	W8
vdda_usb3	USB3 PLL analog power supply	PWR	Y9
vdda_video	VIDEO1 and VIDEO2 PLL analog power supply	PWR	K14, L14
vdds18v	1.8V bump added for atestv esd supply	PWR	G11, H20, W7, Y18
vdds18v_ddr1	DDR2 - 1.8v bias supply	PWR	AA19, P20, Y19
vddshv1	VIN2 domain - 1.8/3.3 mode voltage Power cell - secondary power supply	PWR	G10, G9
vddshv3	GENERAL Domain - 1.8/3.3 mode voltage Power cell - secondary power supply	PWR	G15, G17, H15, H17, J19, K19
vddshv4	MMC4 Domain (UART4) - 1.8/3.3 mode voltage Power cell - secondary power supply	PWR	M19, N19
vddshv7	WIFI Power Group (MMC3/McASP5) - 1.8/3.3 mode voltage Power cell - secondary power supply	PWR	U7, U8
vddshv8	Dual Voltage (1.8V or 3.3V) power supply for the MMC1 Power Group pins	PWR	N8, P8
vddshv9	RGMII - 1.8/3.3 mode voltage Power cell - secondary power supply	PWR	M7, N7
vddshv10	GPMC - 1.8/3.3 mode voltage Power cell - secondary power supply	PWR	J7, J8, K8
vddshv11	MMC2 - 1.8/3.3 mode voltage Power cell - secondary power supply	PWR	F7, G7, H7
vdds_ddr1	DDR2 - vdds2 can be 1.8 (ddr2)/1.5(ddr3) - secondary power supply	PWR	T19, T20, V20, W17, W18, W20
vssa_osc0	OSC0 Analog ground	GND	AA12
vssa_osc1	OSC1 Analog ground	GND	AB11

(1) This pin must always be connected via a 1-µF capacitor to vss.

(2) This signal is valid only for High-Security devices. For more details, see Section 5.8 *VPP Specification for One-Time Programmable (OTP) eFUSEs*. For General Purpose devices do not connect any signal, test point, or board trace to this signal.

4.4 Pin Multiplexing

Table 4-32 describes the device pin multiplexing (no characteristics are provided in this table).

NOTE

Table 4-32, *Pin Multiplexing* doesn't take into account subsystem multiplexing signals. Subsystem multiplexing signals are described in Section 4.3, *Signal Descriptions*.

NOTE

For more information, see the Control Module / Control Module Functional Description / PAD Functional Multiplexing and Configuration section of the Device TRM.

NOTE

Configuring two pins to the same input signal is not supported as it can yield unexpected results. This can be easily prevented with the proper software configuration (Hi-Z mode is not an input signal).

NOTE

When a pad is set into a pin multiplexing mode which is not defined, that pad's behavior is undefined. This should be avoided.

NOTE

In some cases Table 4-32 may present more than one signal per muxmode for the same ball. First signal in the list is the dominant function as selected via CTRL_CORE_PAD_* register.

All other signals are virtual functions that present alternate multiplexing options. This virtual functions are controlled via CTRL_CORE_ALT_SELECT_MUX or CTRL_CORE_VIP_MUX_SELECT register. For more information on how to use this options, please refer to Device TRM, Chapter *Control Module*, Section *Pad Configuration Registers.*

Table 4-32. Pin Multiplexing

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
		P25	ddr1_dqm3															
		Y23	ddr1_d10															
		P21	ddr1_d27															
		T3	mlbp_sig_p															
		U25	ddr1_d17															

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
		AA20	ddr1_a7															
		V25	ddr1_dqsn2															
		AB16	ddr1_ba2															
		T25	ddr1_d25															
		N21	ddr1_d28															
		AB25	ddr1_d13															
		AE9	hdmi1_clockx															
		W23	ddr1_d16															
		AC24	ddr1_d1															
		AD16	ddr1_casn															
		AA23	ddr1_d0															
		AD18	ddr1_odt0															
		AE19	ddr1_a1															
		AC20	ddr1_a9															
		U21	ddr1_dqm2															
		AA24	ddr1_d8															
		U4	mlbp_sig_n															
		AC11	xi_osc1															
		AD1	csi2_0_dx1															
		AE3	usb_txn0	pcie_txn1														
		AC6	usb1_dp															
		AD6	usb_rxp0	pcie_rxp1														
		AA16	ddr1_ba1															
		Y12	xi_osc0															
		AB15	ddr1_a14															
		AC18	ddr1_a0															
		AE11	hdmi1_data0x															
		R25	ddr1_dqsn3															
		Y24	ddr1_dqs1															
		Y21	ddr1_a8															
		W21	ddr1_d19															
		AD20	ddr1_a4															
		AA25	ddr1_d14															
		AD13	hdmi1_data1y															
		AB9	ljcb_clkn															
		AC25	ddr1_d12															
		U22	ddr1_d21															

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
		AB23	ddr1_d4															
		AB24	ddr1_d2															
		AE16	ddr1_ba0															
		T22	ddr1_d20															
		T21	ddr1_d23															
		AB19	ddr1_a3															
		AE24	ddr1_d7															
		AC15	ddr1_a13															
		AC21	ddr1_a11															
		AD17	ddr1_rasn															
		AB12	xo_osc0															
		AD23	ddr1_d6															
		AD9	pcie_txp0															
		V24	ddr1_dqs2															
		U1	mlbp_clk_n															
		U23	ddr1_d22															
		T1	mlbp_dat_n															
		AC22	ddr1_a12															
		AD24	ddr1_d3															
		AC8	ljcb_clkp															
		AE21	ddr1_nck															
		Y20	ddr1_vref0															
		AD7	pcie_rxp0															
		T2	mlbp_dat_p															
		AE23	ddr1_dqm0															
		AD21	ddr1_ck															
		Y25	ddr1_dqsn1															
		AA11	xo_osc1															
		AE17	ddr1_rst															
		W22	ddr1_dqm1															
		AE12	hdmi1_data1x															
		AE14	hdmi1_data2x															
		AB2	csi2_0_dy0															
		AB18	ddr1_cke															
		AB6	usb2_dp															
		AC1	csi2_0_dx0															
		AE8	pcie_txn0															

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
		AC19	ddr1_csn0															
		AA21	ddr1_a10															
		AE6	pcie_rxn0															
		AB7	usb1_dm															
		F19	porz															
		W25	ddr1_d9															
		P24	ddr1_d31															
		AD22	ddr1_dqs0															
		P22	ddr1_d29															
		U24	ddr1_d18															
		AD2	csi2_0_dy2															
		AE18	ddr1_wen															
		AE20	ddr1_a5															
		W24	ddr1_d15															
		T24	ddr1_d26															
		R24	ddr1_dqs3															
		AD15	hdmi1_data2y															
		AE22	ddr1_dqsn0															
		AA18	ddr1_a6															
		U2	mlbp_clk_p															
		AC2	csi2_0_dy1															
		AD12	hdmi1_data0y															
		T23	ddr1_d24															
		AD10	hdmi1_clocky															
		AE5	usb_rxn0	pcie_rxn1														
		AE2	csi2_0_dx2															
		P23	ddr1_d30															
		AC5	usb2_dm															
		AC23	ddr1_d5															
		AD19	ddr1_a2															
		AC16	ddr1_a15															
		AD25	ddr1_d11															
		AD4	usb_txp0	pcie_txp1														
0x1400	CTRL_CORE_PAD_GPMC_AD0	F1	gpmc_ad0		vin1a_d0	vout3_d0											gpio1_6	sysboot0
0x1404	CTRL_CORE_PAD_GPMC_AD1	E2	gpmc_ad1		vin1a_d1	vout3_d1											gpio1_7	sysboot1

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x1408	CTRL_CORE_PAD_GPMC_AD2	E1	gpmc_ad2		vin1a_d2	vout3_d2											gpio1_8	sysboot2
0x140C	CTRL_CORE_PAD_GPMC_AD3	C1	gpmc_ad3		vin1a_d3	vout3_d3											gpio1_9	sysboot3
0x1410	CTRL_CORE_PAD_GPMC_AD4	D1	gpmc_ad4		vin1a_d4	vout3_d4											gpio1_10	sysboot4
0x1414	CTRL_CORE_PAD_GPMC_AD5	D2	gpmc_ad5		vin1a_d5	vout3_d5											gpio1_11	sysboot5
0x1418	CTRL_CORE_PAD_GPMC_AD6	B1	gpmc_ad6		vin1a_d6	vout3_d6											gpio1_12	sysboot6
0x141C	CTRL_CORE_PAD_GPMC_AD7	B2	gpmc_ad7		vin1a_d7	vout3_d7											gpio1_13	sysboot7
0x1420	CTRL_CORE_PAD_GPMC_AD8	C2	gpmc_ad8		vin1a_d8	vout3_d8											gpio7_18	sysboot8
0x1424	CTRL_CORE_PAD_GPMC_AD9	D3	gpmc_ad9		vin1a_d9	vout3_d9											gpio7_19	sysboot9
0x1428	CTRL_CORE_PAD_GPMC_AD10	A2	gpmc_ad10		vin1a_d10	vout3_d10											gpio7_28	sysboot10
0x142C	CTRL_CORE_PAD_GPMC_AD11	B3	gpmc_ad11		vin1a_d11	vout3_d11											gpio7_29	sysboot11
0x1430	CTRL_CORE_PAD_GPMC_AD12	C3	gpmc_ad12		vin1a_d12	vout3_d12											gpio1_18	sysboot12
0x1434	CTRL_CORE_PAD_GPMC_AD13	C4	gpmc_ad13		vin1a_d13	vout3_d13											gpio1_19	sysboot13
0x1438	CTRL_CORE_PAD_GPMC_AD14	A3	gpmc_ad14		vin1a_d14	vout3_d14											gpio1_20	sysboot14
0x143C	CTRL_CORE_PAD_GPMC_AD15	B4	gpmc_ad15		vin1a_d15	vout3_d15											gpio1_21	sysboot15
0x1440	CTRL_CORE_PAD_GPMC_A0	M1	gpmc_a0		vin1a_d16	vout3_d16			vin1b_d0	i2c4_scl	uart5_rxd						gpio7_3 gpmc_a26 gpmc_a16	Driver off
0x1444	CTRL_CORE_PAD_GPMC_A1	M2	gpmc_a1		vin1a_d17	vout3_d17			vin1b_d1	i2c4_sda	uart5_txd						gpio7_4	Driver off
0x1448	CTRL_CORE_PAD_GPMC_A2	L2	gpmc_a2		vin1a_d18	vout3_d18			vin1b_d2	uart7_rxd	uart5_ctsn						gpio7_5	Driver off
0x144C	CTRL_CORE_PAD_GPMC_A3	L1	gpmc_a3	qspi1_cs2	vin1a_d19	vout3_d19			vin1b_d3	uart7_txd	uart5_rtsn						gpio7_6	Driver off
0x1450	CTRL_CORE_PAD_GPMC_A4	K3	gpmc_a4	qspi1_cs3	vin1a_d20	vout3_d20			vin1b_d4	i2c5_scl	uart6_rxd						gpio1_26	Driver off
0x1454	CTRL_CORE_PAD_GPMC_A5	K2	gpmc_a5		vin1a_d21	vout3_d21			vin1b_d5	i2c5_sda	uart6_txd						gpio1_27	Driver off
0x1458	CTRL_CORE_PAD_GPMC_A6	J1	gpmc_a6		vin1a_d22	vout3_d22			vin1b_d6	uart8_rxd	uart6_ctsn						gpio1_28	Driver off
0x145C	CTRL_CORE_PAD_GPMC_A7	K1	gpmc_a7		vin1a_d23	vout3_d23			vin1b_d7	uart8_txd	uart6_rtsn						gpio1_29	Driver off
0x1460	CTRL_CORE_PAD_GPMC_A8	K4	gpmc_a8		vin1a_hsync0	vout3_hsync			vin1b_hsync1	timer12	spi4_sclk						gpio1_30	Driver off

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x1464	CTRL_CORE_PAD_GPMC_A9	H1	gpmc_a9		vin1a_vsync0	vout3_vsync			vin1b_vsync1	timer11	spi4_d1						gpio1_31	Driver off
0x1468	CTRL_CORE_PAD_GPMC_A10	J2	gpmc_a10		vin1a_de0	vout3_de			vin1b_clk1	timer10	spi4_d0						gpio2_0	Driver off
0x146C	CTRL_CORE_PAD_GPMC_A11	L3	gpmc_a11		vin1a_fld0	vout3_fld			vin1b_de1	timer9	spi4_cs0						gpio2_1	Driver off
0x1470	CTRL_CORE_PAD_GPMC_A12	G1	gpmc_a12					gpmc_a0	vin1b_fld1	timer8	spi4_cs1	dma_evt1					gpio2_2	Driver off
0x1474	CTRL_CORE_PAD_GPMC_A13	H3	gpmc_a13	qspi1_rtclk						timer7	spi4_cs2	dma_evt2					gpio2_3	Driver off
0x1478	CTRL_CORE_PAD_GPMC_A14	H4	gpmc_a14	qspi1_d3						timer6	spi4_cs3						gpio2_4	Driver off
0x147C	CTRL_CORE_PAD_GPMC_A15	K6	gpmc_a15	qspi1_d2						timer5							gpio2_5	Driver off
0x1480	CTRL_CORE_PAD_GPMC_A16	K5	gpmc_a16	qspi1_d0													gpio2_6	Driver off
0x1484	CTRL_CORE_PAD_GPMC_A17	G2	gpmc_a17	qspi1_d1													gpio2_7	Driver off
0x1488	CTRL_CORE_PAD_GPMC_A18	F2	gpmc_a18	qspi1_sclk													gpio2_8	Driver off
0x148C	CTRL_CORE_PAD_GPMC_A19	A4	gpmc_a19	mmc2_dat4	gpmc_a13				vin2b_d0								gpio2_9	Driver off
0x1490	CTRL_CORE_PAD_GPMC_A20	E7	gpmc_a20	mmc2_dat5	gpmc_a14				vin2b_d1								gpio2_10	Driver off
0x1494	CTRL_CORE_PAD_GPMC_A21	D6	gpmc_a21	mmc2_dat6	gpmc_a15				vin2b_d2								gpio2_11	Driver off
0x1498	CTRL_CORE_PAD_GPMC_A22	C5	gpmc_a22	mmc2_dat7	gpmc_a16				vin2b_d3								gpio2_12	Driver off
0x149C	CTRL_CORE_PAD_GPMC_A23	B5	gpmc_a23	mmc2_clk	gpmc_a17				vin2b_d4								gpio2_13	Driver off
0x14A0	CTRL_CORE_PAD_GPMC_A24	D7	gpmc_a24	mmc2_dat0	gpmc_a18				vin2b_d5								gpio2_14	Driver off
0x14A4	CTRL_CORE_PAD_GPMC_A25	C6	gpmc_a25	mmc2_dat1	gpmc_a19				vin2b_d6								gpio2_15	Driver off
0x14A8	CTRL_CORE_PAD_GPMC_A26	A5	gpmc_a26	mmc2_dat2	gpmc_a20				vin2b_d7								gpio2_16	Driver off
0x14AC	CTRL_CORE_PAD_GPMC_A27	B6	gpmc_a27	mmc2_dat3	gpmc_a21				vin2b_hsync1								gpio2_17	Driver off
0x14B0	CTRL_CORE_PAD_GPMC_CS1	A6	gpmc_cs1	mmc2_cmd	gpmc_a22				vin2b_vsync1								gpio2_18	Driver off
0x14B4	CTRL_CORE_PAD_GPMC_CS0	F3	gpmc_cs0														gpio2_19	Driver off
0x14B8	CTRL_CORE_PAD_GPMC_CS2	G4	gpmc_cs2	qspi1_cs0													gpio2_20 gpmc_a23 gpmc_a13	Driver off
0x14BC	CTRL_CORE_PAD_GPMC_CS3	G3	gpmc_cs3	qspi1_cs1	vin1a_clk0	vout3_clk		gpmc_a1									gpio2_21 gpmc_a24 gpmc_a14	Driver off

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x14C0	CTRL_CORE_PAD_GPMC_CLK	L4	gpmc_clk	gpmc_cs7	clkout1	gpmc_wait1			vin2b_clk1	timer4	i2c3_scl	dma_evt1					gpio2_22 gpmc_a20	Driver off
0x14C4	CTRL_CORE_PAD_GPMC_ADVN_ALE	H5	gpmc_advn_ale	gpmc_cs6	clkout2	gpmc_wait1		gpmc_a2	gpmc_a23	timer3	i2c3_sda	dma_evt2					gpio2_23 gpmc_a19	Driver off
0x14C8	CTRL_CORE_PAD_GPMC_OEN_REN	G5	gpmc_oen_ren														gpio2_24	Driver off
0x14CC	CTRL_CORE_PAD_GPMC_WEN	G6	gpmc_wen														gpio2_25	Driver off
0x14D0	CTRL_CORE_PAD_GPMC_BEN0	H2	gpmc_ben0	gpmc_cs4					vin2b_de1	timer2		dma_evt3					gpio2_26 gpmc_a21	Driver off
0x14D4	CTRL_CORE_PAD_GPMC_BEN1	H6	gpmc_ben1	gpmc_cs5			vin2b_clk1	gpmc_a3	vin2b_fld1	timer1		dma_evt4					gpio2_27 gpmc_a22	Driver off
0x14D8	CTRL_CORE_PAD_GPMC_WAIT0	F6	gpmc_wait0														gpio2_28 gpmc_a25 gpmc_a15	Driver off
0x1554	CTRL_CORE_PAD_VIN2A_CLK0	D8	vin2a_clk0				vout2_fld	emu5				kbd_row0	eQEP1A_in		pr1_edio_data_in0	pr1_edio_data_out0	gpio3_28 gpmc_a27 gpmc_a17	Driver off
0x1558	CTRL_CORE_PAD_VIN2A_DE0	B7	vin2a_de0	vin2a_fld0	vin2b_fld1	vin2b_de1	vout2_de	emu6				kbd_row1	eQEP1B_in		pr1_edio_data_in1	pr1_edio_data_out1	gpio3_29	Driver off
0x155C	CTRL_CORE_PAD_VIN2A_FLD0	C7	vin2a_fld0		vin2b_clk1		vout2_clk	emu7					eQEP1_index		pr1_edio_data_in2	pr1_edio_data_out2	gpio3_30 gpmc_a27 gpmc_a18	Driver off
0x1560	CTRL_CORE_PAD_VIN2A_HSYNC0	E8	vin2a_hsync0			vin2b_hsync1	vout2_hsync	emu8		uart9_rxd	spi4_sclk	kbd_row2	eQEP1_strobe	pr1_uart0_cts_n	pr1_edio_data_in3	pr1_edio_data_out3	gpio3_31 gpmc_a27	Driver off
0x1564	CTRL_CORE_PAD_VIN2A_VSYNC0	B8	vin2a_vsync0			vin2b_vsync1	vout2_vsync	emu9		uart9_txd	spi4_d1	kbd_row3	ehrpwm1A	pr1_uart0_rts_n	pr1_edio_data_in4	pr1_edio_data_out4	gpio4_0	Driver off
0x1568	CTRL_CORE_PAD_VIN2A_D0	C8	vin2a_d0				vout2_d23	emu10		uart9_ctsn	spi4_d0	kbd_row4	ehrpwm1B	pr1_uart0_rxd	pr1_edio_data_in5	pr1_edio_data_out5	gpio4_1	Driver off
0x156C	CTRL_CORE_PAD_VIN2A_D1	B9	vin2a_d1				vout2_d22	emu11		uart9_rtsn	spi4_cs0	kbd_row5	ehrpwm1_tripzone_input	pr1_uart0_txd	pr1_edio_data_in6	pr1_edio_data_out6	gpio4_2	Driver off
0x1570	CTRL_CORE_PAD_VIN2A_D2	A7	vin2a_d2				vout2_d21	emu12			uart10_rxd	kbd_row6	eCAP1_in_PWM1_out	pr1_ecap0_ecap_capin_apwm_o	pr1_edio_data_in7	pr1_edio_data_out7	gpio4_3	Driver off
0x1574	CTRL_CORE_PAD_VIN2A_D3	A9	vin2a_d3				vout2_d20	emu13			uart10_txd	kbd_col0	ehrpwm1_synci	pr1_edc_latch0_in	pr1_pru1_gpi0	pr1_pru1_gpo0	gpio4_4	Driver off
0x1578	CTRL_CORE_PAD_VIN2A_D4	A8	vin2a_d4				vout2_d19	emu14			uart10_ctsn	kbd_col1	ehrpwm1_synco	pr1_edc_sync0_out	pr1_pru1_gpi1	pr1_pru1_gpo1	gpio4_5	Driver off
0x157C	CTRL_CORE_PAD_VIN2A_D5	A11	vin2a_d5				vout2_d18	emu15			uart10_rtsn	kbd_col2	eQEP2A_in	pr1_edio_sof	pr1_pru1_gpi2	pr1_pru1_gpo2	gpio4_6	Driver off
0x1580	CTRL_CORE_PAD_VIN2A_D6	F10	vin2a_d6				vout2_d17	emu16			mii1_rxd1	kbd_col3	eQEP2B_in	pr1_mii_mt1_clk	pr1_pru1_gpi3	pr1_pru1_gpo3	gpio4_7	Driver off
0x1584	CTRL_CORE_PAD_VIN2A_D7	A10	vin2a_d7				vout2_d16	emu17			mii1_rxd2	kbd_col4	eQEP2_index	pr1_mii1_txen	pr1_pru1_gpi4	pr1_pru1_gpo4	gpio4_8	Driver off
0x1588	CTRL_CORE_PAD_VIN2A_D8	B10	vin2a_d8				vout2_d15	emu18			mii1_rxd3	kbd_col5	eQEP2_strobe	pr1_mii1_txd3	pr1_pru1_gpi5	pr1_pru1_gpo5	gpio4_9 gpmc_a26	Driver off

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x158C	CTRL_CORE_PAD_VIN2A_D9	E10	vin2a_d9				vout2_d14	emu19			mii1_rxd0	kbd_col6	ehrpwm2A	pr1_mii1_txd2	pr1_pru1_gpi6	pr1_pru1_gpo6	gpio4_10 gpmc_a25	Driver off
0x1590	CTRL_CORE_PAD_VIN2A_D10	D10	vin2a_d10			mdio_mclk	vout2_d13					kbd_col7	ehrpwm2B	pr1_mdio_mdclk	pr1_pru1_gpi7	pr1_pru1_gpo7	gpio4_11 gpmc_a24	Driver off
0x1594	CTRL_CORE_PAD_VIN2A_D11	C10	vin2a_d11			mdio_d	vout2_d12					kbd_row7	ehrpwm2_tripzone_input	pr1_mdio_data	pr1_pru1_gpi8	pr1_pru1_gpo8	gpio4_12 gpmc_a23	Driver off
0x1598	CTRL_CORE_PAD_VIN2A_D12	B11	vin2a_d12			rgmii1_txc	vout2_d11				mii1_rxclk	kbd_col8	eCAP2_in_PWM2_out	pr1_mii1_txd1	pr1_pru1_gpi9	pr1_pru1_gpo9	gpio4_13	Driver off
0x159C	CTRL_CORE_PAD_VIN2A_D13	D11	vin2a_d13			rgmii1_txctl	vout2_d10				mii1_rxdv	kbd_row8	eQEP3A_in	pr1_mii1_txd0	pr1_pru1_gpi10	pr1_pru1_gpo10	gpio4_14	Driver off
0x15A0	CTRL_CORE_PAD_VIN2A_D14	C11	vin2a_d14			rgmii1_txd3	vout2_d9				mii1_txclk		eQEP3B_in	pr1_mii_mr1_clk	pr1_pru1_gpi11	pr1_pru1_gpo11	gpio4_15	Driver off
0x15A4	CTRL_CORE_PAD_VIN2A_D15	B12	vin2a_d15			rgmii1_txd2	vout2_d8				mii1_txd0		eQEP3_index	pr1_mii1_rxdv	pr1_pru1_gpi12	pr1_pru1_gpo12	gpio4_16	Driver off
0x15A8	CTRL_CORE_PAD_VIN2A_D16	A12	vin2a_d16		vin2b_d7	rgmii1_txd1	vout2_d7				mii1_txd1		eQEP3_strobe	pr1_mii1_rxd3	pr1_pru1_gpi13	pr1_pru1_gpo13	gpio4_24	Driver off
0x15AC	CTRL_CORE_PAD_VIN2A_D17	A13	vin2a_d17		vin2b_d6	rgmii1_txd0	vout2_d6				mii1_txd2		ehrpwm3A	pr1_mii1_rxd2	pr1_pru1_gpi14	pr1_pru1_gpo14	gpio4_25	Driver off
0x15B0	CTRL_CORE_PAD_VIN2A_D18	E11	vin2a_d18		vin2b_d5	rgmii1_rxc	vout2_d5				mii1_txd3		ehrpwm3B	pr1_mii1_rxd1	pr1_pru1_gpi15	pr1_pru1_gpo15	gpio4_26	Driver off
0x15B4	CTRL_CORE_PAD_VIN2A_D19	F11	vin2a_d19		vin2b_d4	rgmii1_rxctl	vout2_d4				mii1_txer		ehrpwm3_tripzone_input	pr1_mii1_rxd0	pr1_pru1_gpi16	pr1_pru1_gpo16	gpio4_27	Driver off
0x15B8	CTRL_CORE_PAD_VIN2A_D20	B13	vin2a_d20		vin2b_d3	rgmii1_rxd3	vout2_d3				mii1_rxer		eCAP3_in_PWM3_out	pr1_mii1_rxer	pr1_pru1_gpi17	pr1_pru1_gpo17	gpio4_28	Driver off
0x15BC	CTRL_CORE_PAD_VIN2A_D21	E13	vin2a_d21		vin2b_d2	rgmii1_rxd2	vout2_d2				mii1_col			pr1_mii1_rxlink	pr1_pru1_gpi18	pr1_pru1_gpo18	gpio4_29	Driver off
0x15C0	CTRL_CORE_PAD_VIN2A_D22	C13	vin2a_d22		vin2b_d1	rgmii1_rxd1	vout2_d1				mii1_crs			pr1_mii1_col	pr1_pru1_gpi19	pr1_pru1_gpo19	gpio4_30	Driver off
0x15C4	CTRL_CORE_PAD_VIN2A_D23	D13	vin2a_d23		vin2b_d0	rgmii1_rxd0	vout2_d0				mii1_txen			pr1_mii1_crs	pr1_pru1_gpi20	pr1_pru1_gpo20	gpio4_31	Driver off
0x15E4	CTRL_CORE_PAD_VOUT1_D2	E14			emu2													
0x1604	CTRL_CORE_PAD_VOUT1_D10	F14			emu3													
0x1624	CTRL_CORE_PAD_VOUT1_D18	F13			emu4													
0x163C	CTRL_CORE_PAD_MDIO_MCLK	L5	mdio_mclk	uart3_rtsn		mii0_col	vin2a_clk0	vin1b_clk1						pr1_mii0_col	pr2_pru1_gpi0	pr2_pru1_gpo0	gpio5_15	Driver off
0x1640	CTRL_CORE_PAD_MDIO_D	L6	mdio_d	uart3_ctsn		mii0_txer	vin2a_d0	vin1b_d0						pr1_mii0_rxlink	pr2_pru1_gpi1	pr2_pru1_gpo1	gpio5_16	Driver off
0x1644	CTRL_CORE_PAD_RMII_MHZ_50_CLK	P5	RMII_MHZ_50_CLK				vin2a_d11								pr2_pru1_gpi2	pr2_pru1_gpo2	gpio5_17	Driver off
0x1648	CTRL_CORE_PAD_UART3_RXD	N5	uart3_rxd		rmii1_crs	mii0_rxdv	vin2a_d1	vin1b_d1		spi3_sclk				pr1_mii0_rxdv	pr2_pru1_gpi3	pr2_pru1_gpo3	gpio5_18	Driver off

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x164C	CTRL_CORE_PAD_UART3_TXD	N6	uart3_txd		rmii1_rxer	mii0_rxclk	vin2a_d2	vin1b_d2		spi3_d1	spi4_cs1			pr1_mii_mr0_clk	pr2_pru1_gpi4	pr2_pru1_gpo4	gpio5_19	Driver off
0x1650	CTRL_CORE_PAD_RGMII0_TXC	T4	rgmii0_txc	uart3_ctsn	rmii1_rxd1	mii0_rxd3	vin2a_d3	vin1b_d3	usb3_ulpi_clk	spi3_d0	spi4_cs2			pr1_mii0_rxd3	pr2_pru1_gpi5	pr2_pru1_gpo5	gpio5_20	Driver off
0x1654	CTRL_CORE_PAD_RGMII0_TXCTL	T5	rgmii0_txctl	uart3_rtsn	rmii1_rxd0	mii0_rxd2	vin2a_d4	vin1b_d4	usb3_ulpi_stp	spi3_cs0	spi4_cs3			pr1_mii0_rxd2	pr2_pru1_gpi6	pr2_pru1_gpo6	gpio5_21	Driver off
0x1658	CTRL_CORE_PAD_RGMII0_TXD3	P4	rgmii0_txd3	rmii0_crs		mii0_crs	vin2a_de0	vin1b_de1	usb3_ulpi_dir	spi4_sclk	uart4_rxd			pr1_mii0_crs	pr2_pru1_gpi7	pr2_pru1_gpo7	gpio5_22	Driver off
0x165C	CTRL_CORE_PAD_RGMII0_TXD2	P3	rgmii0_txd2	rmii0_rxer		mii0_rxer	vin2a_hsync0	vin1b_hsync1	usb3_ulpi_nxt	spi4_d1	uart4_txd			pr1_mii0_rxer	pr2_pru1_gpi8	pr2_pru1_gpo8	gpio5_23	Driver off
0x1660	CTRL_CORE_PAD_RGMII0_TXD1	R2	rgmii0_txd1	rmii0_rxd1		mii0_rxd1	vin2a_vsync0	vin1b_vsync1	usb3_ulpi_d0	spi4_d0	uart4_ctsn			pr1_mii0_rxd1	pr2_pru1_gpi9	pr2_pru1_gpo9	gpio5_24	Driver off
0x1664	CTRL_CORE_PAD_RGMII0_TXD0	R1	rgmii0_txd0	rmii0_rxd0		mii0_rxd0	vin2a_d10		usb3_ulpi_d1	spi4_cs0	uart4_rtsn			pr1_mii0_rxd0	pr2_pru1_gpi10	pr2_pru1_gpo10	gpio5_25	Driver off
0x1668	CTRL_CORE_PAD_RGMII0_RXC	N2	rgmii0_rxc		rmii1_txen	mii0_txclk	vin2a_d5	vin1b_d5	usb3_ulpi_d2					pr1_mii_mt0_clk	pr2_pru1_gpi11	pr2_pru1_gpo11	gpio5_26	Driver off
0x166C	CTRL_CORE_PAD_RGMII0_RXCTL	P2	rgmii0_rxctl		rmii1_txd1	mii0_txd3	vin2a_d6	vin1b_d6	usb3_ulpi_d3					pr1_mii0_txd3	pr2_pru1_gpi12	pr2_pru1_gpo12	gpio5_27	Driver off
0x1670	CTRL_CORE_PAD_RGMII0_RXD3	N1	rgmii0_rxd3		rmii1_txd0	mii0_txd2	vin2a_d7	vin1b_d7	usb3_ulpi_d4					pr1_mii0_txd2	pr2_pru1_gpi13	pr2_pru1_gpo13	gpio5_28	Driver off
0x1674	CTRL_CORE_PAD_RGMII0_RXD2	P1	rgmii0_rxd2	rmii0_txen		mii0_txen	vin2a_d8		usb3_ulpi_d5					pr1_mii0_txen	pr2_pru1_gpi14	pr2_pru1_gpo14	gpio5_29	Driver off
0x1678	CTRL_CORE_PAD_RGMII0_RXD1	N3	rgmii0_rxd1	rmii0_txd1		mii0_txd1	vin2a_d9		usb3_ulpi_d6					pr1_mii0_txd1	pr2_pru1_gpi15	pr2_pru1_gpo15	gpio5_30	Driver off
0x167C	CTRL_CORE_PAD_RGMII0_RXD0	N4	rgmii0_rxd0	rmii0_txd0		mii0_txd0	vin2a_fld0	vin1b_fld1	usb3_ulpi_d7					pr1_mii0_txd0	pr2_pru1_gpi16	pr2_pru1_gpo16	gpio5_31	Driver off
0x1680	CTRL_CORE_PAD_USB1_DRVVBUS	AD3	usb1_drvvbus							timer16							gpio6_12	Driver off
0x1684	CTRL_CORE_PAD_USB2_DRVVBUS	AA6	usb2_drvvbus							timer15							gpio6_13	Driver off
0x1688	CTRL_CORE_PAD_GPIO6_14	H21	gpio6_14	mcasp1_axr8	dcan2_tx	uart10_rxd						i2c3_sda	timer1				gpio6_14	Driver off
0x168C	CTRL_CORE_PAD_GPIO6_15	K22	gpio6_15	mcasp1_axr9	dcan2_rx	uart10_txd						i2c3_scl	timer2				gpio6_15	Driver off
0x1690	CTRL_CORE_PAD_GPIO6_16	K23	gpio6_16	mcasp1_axr10								clkout1	timer3				gpio6_16	Driver off
0x1694	CTRL_CORE_PAD_XREF_CLK0	J25	xref_clk0	mcasp2_axr8	mcasp1_axr4	mcasp1_ahclkx	mcasp5_ahclkx	atl_clk0		vin1a_d0	hdq0	clkout2	timer13	pr2_mii1_col	pr2_pru1_gpi5	pr2_pru1_gpo5	gpio6_17	Driver off
0x1698	CTRL_CORE_PAD_XREF_CLK1	J24	xref_clk1	mcasp2_axr9	mcasp1_axr5	mcasp2_ahclkx	mcasp6_ahclkx	atl_clk1		vin1a_clk0			timer14	pr2_mii1_crs	pr2_pru1_gpi6	pr2_pru1_gpo6	gpio6_18	Driver off
0x169C	CTRL_CORE_PAD_XREF_CLK2	H24	xref_clk2	mcasp2_axr10	mcasp1_axr6	mcasp3_ahclkx	mcasp7_ahclkx	atl_clk2					timer15				gpio6_19	Driver off
0x16A0	CTRL_CORE_PAD_XREF_CLK3	H25	xref_clk3	mcasp2_axr11	mcasp1_axr7	mcasp4_ahclkx	mcasp8_ahclkx	atl_clk3		hdq0		clkout3	timer16				gpio6_20	Driver off
0x16A4	CTRL_CORE_PAD_MCASP1_ACLKX	C16	mcasp1_aclkx							vin1a_fld0			i2c3_sda	pr2_mdio_mdclk	pr2_pru1_gpi7	pr2_pru1_gpo7	gpio7_31	Driver off
0x16A8	CTRL_CORE_PAD_MCASP1_FSX	C17	mcasp1_fsx							vin1a_de0			i2c3_scl	pr2_mdio_data			gpio7_30	Driver off

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x16AC	CTRL_CORE_PAD_MCASP1_ACLKR	D16	mcasp1_aclkr	mcasp7_axr2									i2c4_sda				gpio5_0	Driver off
0x16B0	CTRL_CORE_PAD_MCASP1_FSR	D17	mcasp1_fsr	mcasp7_axr3									i2c4_scl				gpio5_1	Driver off
0x16B4	CTRL_CORE_PAD_MCASP1_AXR0	D14	mcasp1_axr0			uart6_rxd				vin1a_vsync0			i2c5_sda	pr2_mii0_rxer	pr2_pru1_gpi8	pr2_pru1_gpo8	gpio5_2	Driver off
0x16B8	CTRL_CORE_PAD_MCASP1_AXR1	B14	mcasp1_axr1			uart6_txd				vin1a_hsync0			i2c5_scl	pr2_mii_mt0_clk	pr2_pru1_gpi9	pr2_pru1_gpo9	gpio5_3	Driver off
0x16BC	CTRL_CORE_PAD_MCASP1_AXR2	C14	mcasp1_axr2	mcasp6_axr2		uart6_ctsn											gpio5_4	Driver off
0x16C0	CTRL_CORE_PAD_MCASP1_AXR3	B15	mcasp1_axr3	mcasp6_axr3		uart6_rtsn											gpio5_5	Driver off
0x16C4	CTRL_CORE_PAD_MCASP1_AXR4	A15	mcasp1_axr4	mcasp4_axr2													gpio5_6	Driver off
0x16C8	CTRL_CORE_PAD_MCASP1_AXR5	A14	mcasp1_axr5	mcasp4_axr3													gpio5_7	Driver off
0x16CC	CTRL_CORE_PAD_MCASP1_AXR6	A17	mcasp1_axr6	mcasp5_axr2													gpio5_8	Driver off
0x16D0	CTRL_CORE_PAD_MCASP1_AXR7	A16	mcasp1_axr7	mcasp5_axr3									timer4				gpio5_9	Driver off
0x16D4	CTRL_CORE_PAD_MCASP1_AXR8	A18	mcasp1_axr8	mcasp6_axr0		spi3_sclk				vin1a_d15			timer5	pr2_mii0_txen	pr2_pru1_gpi10	pr2_pru1_gpo10	gpio5_10	Driver off
0x16D8	CTRL_CORE_PAD_MCASP1_AXR9	B17	mcasp1_axr9	mcasp6_axr1		spi3_d1				vin1a_d14			timer6	pr2_mii0_txd3	pr2_pru1_gpi11	pr2_pru1_gpo11	gpio5_11	Driver off
0x16DC	CTRL_CORE_PAD_MCASP1_AXR10	B16	mcasp1_axr10	mcasp6_aclkx	mcasp6_aclkr	spi3_d0				vin1a_d13			timer7	pr2_mii0_txd2	pr2_pru1_gpi12	pr2_pru1_gpo12	gpio5_12	Driver off
0x16E0	CTRL_CORE_PAD_MCASP1_AXR11	B18	mcasp1_axr11	mcasp6_fsx	mcasp6_fsr	spi3_cs0				vin1a_d12			timer8	pr2_mii0_txd1	pr2_pru1_gpi13	pr2_pru1_gpo13	gpio4_17	Driver off
0x16E4	CTRL_CORE_PAD_MCASP1_AXR12	A19	mcasp1_axr12	mcasp7_axr0		spi3_cs1				vin1a_d11			timer9	pr2_mii0_txd0	pr2_pru1_gpi14	pr2_pru1_gpo14	gpio4_18	Driver off
0x16E8	CTRL_CORE_PAD_MCASP1_AXR13	E17	mcasp1_axr13	mcasp7_axr1						vin1a_d10			timer10	pr2_mii_mr0_clk	pr2_pru1_gpi15	pr2_pru1_gpo15	gpio6_4	Driver off
0x16EC	CTRL_CORE_PAD_MCASP1_AXR14	E16	mcasp1_axr14	mcasp7_aclkx	mcasp7_aclkr					vin1a_d9			timer11	pr2_mii0_rxdv	pr2_pru1_gpi16	pr2_pru1_gpo16	gpio6_5	Driver off
0x16F0	CTRL_CORE_PAD_MCASP1_AXR15	F16	mcasp1_axr15	mcasp7_fsx	mcasp7_fsr					vin1a_d8			timer12	pr2_mii0_rxd3	pr2_pru0_gpi20	pr2_pru0_gpo20	gpio6_6	Driver off
0x16F4	CTRL_CORE_PAD_MCASP2_ACLKX	E19	mcasp2_aclkx							vin1a_d7				pr2_mii0_rxd2	pr2_pru0_gpi18	pr2_pru0_gpo18		Driver off
0x16F8	CTRL_CORE_PAD_MCASP2_FSX	D19	mcasp2_fsx							vin1a_d6				pr2_mii0_rxd1	pr2_pru0_gpi19	pr2_pru0_gpo19		Driver off
0x1704	CTRL_CORE_PAD_MCASP2_AXR0	A20	mcasp2_axr0															Driver off
0x1708	CTRL_CORE_PAD_MCASP2_AXR1	B19	mcasp2_axr1															Driver off
0x170C	CTRL_CORE_PAD_MCASP2_AXR2	A21	mcasp2_axr2	mcasp3_axr2						vin1a_d5				pr2_mii0_rxd0	pr2_pru0_gpi16	pr2_pru0_gpo16	gpio6_8	Driver off
0x1710	CTRL_CORE_PAD_MCASP2_AXR3	B21	mcasp2_axr3	mcasp3_axr3						vin1a_d4				pr2_mii0_rxlink	pr2_pru0_gpi17	pr2_pru0_gpo17	gpio6_9	Driver off

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x1714	CTRL_CORE_PAD_MCASP2_AXR4	B20	mcasp2_axr4	mcasp8_axr0													gpio1_4	Driver off
0x1718	CTRL_CORE_PAD_MCASP2_AXR5	C19	mcasp2_axr5	mcasp8_axr1													gpio6_7	Driver off
0x171C	CTRL_CORE_PAD_MCASP2_AXR6	D20	mcasp2_axr6	mcasp8_aclkx	mcasp8_aclkr												gpio2_29	Driver off
0x1720	CTRL_CORE_PAD_MCASP2_AXR7	C20	mcasp2_axr7	mcasp8_fsx	mcasp8_fsr												gpio1_5	Driver off
0x1724	CTRL_CORE_PAD_MCASP3_ACLKX	A22	mcasp3_aclkx	mcasp3_aclkr	mcasp2_axr12	uart7_rxd				vin1a_d3				pr2_mii0_crs	pr2_pru0_gpi12	pr2_pru0_gpo12	gpio5_13	Driver off
0x1728	CTRL_CORE_PAD_MCASP3_FSX	A23	mcasp3_fsx	mcasp3_fsr	mcasp2_axr13	uart7_txd				vin1a_d2				pr2_mii0_col	pr2_pru0_gpi13	pr2_pru0_gpo13	gpio5_14	Driver off
0x172C	CTRL_CORE_PAD_MCASP3_AXR0	B22	mcasp3_axr0		mcasp2_axr14	uart7_ctsn	uart5_rxd			vin1a_d1				pr2_mii1_rxer	pr2_pru0_gpi14	pr2_pru0_gpo14		Driver off
0x1730	CTRL_CORE_PAD_MCASP3_AXR1	B23	mcasp3_axr1		mcasp2_axr15	uart7_rtsn	uart5_txd			vin1a_d0				pr2_mii1_rxlink	pr2_pru0_gpi15	pr2_pru0_gpo15		Driver off
0x1734	CTRL_CORE_PAD_MCASP4_ACLKX	C23	mcasp4_aclkx	mcasp4_aclkr	spi3_sclk	uart8_rxd	i2c4_sda											Driver off
0x1738	CTRL_CORE_PAD_MCASP4_FSX	B25	mcasp4_fsx	mcasp4_fsr	spi3_d1	uart8_txd	i2c4_scl											Driver off
0x173C	CTRL_CORE_PAD_MCASP4_AXR0	A24	mcasp4_axr0		spi3_d0	uart8_ctsn	uart4_rxd										i2c6_scl	Driver off
0x1740	CTRL_CORE_PAD_MCASP4_AXR1	D23	mcasp4_axr1		spi3_cs0	uart8_rtsn	uart4_txd								pr2_pru1_gpi0	pr2_pru1_gpo0	i2c6_sda	Driver off
0x1744	CTRL_CORE_PAD_MCASP5_ACLKX	AC3	mcasp5_aclkx	mcasp5_aclkr	spi4_sclk	uart9_rxd	i2c5_sda	mlb_clk							pr2_pru1_gpi1	pr2_pru1_gpo1		Driver off
0x1748	CTRL_CORE_PAD_MCASP5_FSX	U6	mcasp5_fsx	mcasp5_fsr	spi4_d1	uart9_txd	i2c5_scl								pr2_pru1_gpi2	pr2_pru1_gpo2		Driver off
0x174C	CTRL_CORE_PAD_MCASP5_AXR0	AA5	mcasp5_axr0		spi4_d0	uart9_ctsn	uart3_rxd	mlb_sig						pr2_mdio_mdclk	pr2_pru1_gpi3	pr2_pru1_gpo3		Driver off
0x1750	CTRL_CORE_PAD_MCASP5_AXR1	AC4	mcasp5_axr1		spi4_cs0	uart9_rtsn	uart3_txd	mlb_dat						pr2_mdio_data	pr2_pru1_gpi4	pr2_pru1_gpo4		Driver off
0x1754	CTRL_CORE_PAD_MMC1_CLK	U3	mmc1_clk														gpio6_21	Driver off
0x1758	CTRL_CORE_PAD_MMC1_CMD	V4	mmc1_cmd														gpio6_22	Driver off
0x175C	CTRL_CORE_PAD_MMC1_DAT0	V3	mmc1_dat0														gpio6_23	Driver off
0x1760	CTRL_CORE_PAD_MMC1_DAT1	V2	mmc1_dat1														gpio6_24	Driver off
0x1764	CTRL_CORE_PAD_MMC1_DAT2	W1	mmc1_dat2														gpio6_25	Driver off
0x1768	CTRL_CORE_PAD_MMC1_DAT3	V1	mmc1_dat3														gpio6_26	Driver off
0x176C	CTRL_CORE_PAD_MMC1_SDCD	U5	mmc1_sdcd			uart6_rxd	i2c4_sda										gpio6_27	Driver off
0x1770	CTRL_CORE_PAD_MMC1_SDWP	V5	mmc1_sdwp			uart6_txd	i2c4_scl										gpio6_28	Driver off

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x1774	CTRL_CORE_PAD_GPIO6_10	Y5	gpio6_10	mdio_mclk	i2c3_sda	usb3_ulpi_d7	vin2b_hsync1					vin1a_clk0	ehrpwm2A	pr2_mii_mt1_clk	pr2_pru0_gpi0	pr2_pru0_gpo0	gpio6_10	Driver off
0x1778	CTRL_CORE_PAD_GPIO6_11	Y6	gpio6_11	mdio_d	i2c3_scl	usb3_ulpi_d6	vin2b_vsync1					vin1a_de0	ehrpwm2B	pr2_mii1_txen	pr2_pru0_gpi1	pr2_pru0_gpo1	gpio6_11	Driver off
0x177C	CTRL_CORE_PAD_MMC3_CLK	Y2	mmc3_clk			usb3_ulpi_d5	vin2b_d7					vin1a_d7	ehrpwm2_tripzone_input	pr2_mii1_txd3	pr2_pru0_gpi2	pr2_pru0_gpo2	gpio6_29	Driver off
0x1780	CTRL_CORE_PAD_MMC3_CMD	Y1	mmc3_cmd	spi3_sclk		usb3_ulpi_d4	vin2b_d6					vin1a_d6	eCAP2_in_PWM2_out	pr2_mii1_txd2	pr2_pru0_gpi3	pr2_pru0_gpo3	gpio6_30	Driver off
0x1784	CTRL_CORE_PAD_MMC3_DAT0	Y4	mmc3_dat0	spi3_d1	uart5_rxd	usb3_ulpi_d3	vin2b_d5					vin1a_d5	eQEP3A_in	pr2_mii1_txd1	pr2_pru0_gpi4	pr2_pru0_gpo4	gpio6_31	Driver off
0x1788	CTRL_CORE_PAD_MMC3_DAT1	AA2	mmc3_dat1	spi3_d0	uart5_txd	usb3_ulpi_d2	vin2b_d4					vin1a_d4	eQEP3B_in	pr2_mii1_txd0	pr2_pru0_gpi5	pr2_pru0_gpo5	gpio7_0	Driver off
0x178C	CTRL_CORE_PAD_MMC3_DAT2	AA3	mmc3_dat2	spi3_cs0	uart5_ctsn	usb3_ulpi_d1	vin2b_d3					vin1a_d3	eQEP3_index	pr2_mii_mr1_clk	pr2_pru0_gpi6	pr2_pru0_gpo6	gpio7_1	Driver off
0x1790	CTRL_CORE_PAD_MMC3_DAT3	W2	mmc3_dat3	spi3_cs1	uart5_rtsn	usb3_ulpi_d0	vin2b_d2					vin1a_d2	eQEP3_strobe	pr2_mii1_rxdv	pr2_pru0_gpi7	pr2_pru0_gpo7	gpio7_2	Driver off
0x1794	CTRL_CORE_PAD_MMC3_DAT4	Y3	mmc3_dat4	spi4_sclk	uart10_rxd	usb3_ulpi_nxt	vin2b_d1					vin1a_d1	ehrpwm3A	pr2_mii1_rxd3	pr2_pru0_gpi8	pr2_pru0_gpo8	gpio1_22	Driver off
0x1798	CTRL_CORE_PAD_MMC3_DAT5	AA1	mmc3_dat5	spi4_d1	uart10_txd	usb3_ulpi_dir	vin2b_d0					vin1a_d0	ehrpwm3B	pr2_mii1_rxd2	pr2_pru0_gpi9	pr2_pru0_gpo9	gpio1_23	Driver off
0x179C	CTRL_CORE_PAD_MMC3_DAT6	AA4	mmc3_dat6	spi4_d0	uart10_ctsn	usb3_ulpi_stp	vin2b_de1					vin1a_hsync0	ehrpwm3_tripzone_input	pr2_mii1_rxd1	pr2_pru0_gpi10	pr2_pru0_gpo10	gpio1_24	Driver off
0x17A0	CTRL_CORE_PAD_MMC3_DAT7	AB1	mmc3_dat7	spi4_cs0	uart10_rtsn	usb3_ulpi_clk	vin2b_clk1					vin1a_vsync0	eCAP3_in_PWM3_out	pr2_mii1_rxd0	pr2_pru0_gpi11	pr2_pru0_gpo11	gpio1_25	Driver off
0x17A4	CTRL_CORE_PAD_SPI1_SCLK	C24	spi1_sclk														gpio7_7	Driver off
0x17A8	CTRL_CORE_PAD_SPI1_D1	D24	spi1_d1														gpio7_8	Driver off
0x17AC	CTRL_CORE_PAD_SPI1_D0	D25	spi1_d0														gpio7_9	Driver off
0x17B0	CTRL_CORE_PAD_SPI1_CS0	B24	spi1_cs0														gpio7_10	Driver off
0x17B4	CTRL_CORE_PAD_SPI1_CS1	C25	spi1_cs1			spi2_cs1											gpio7_11	Driver off
0x17B8	CTRL_CORE_PAD_SPI1_CS2	E24	spi1_cs2	uart4_rxd	mmc3_sdcd	spi2_cs2	dcan2_tx	mdio_mclk	hdmi1_hpd								gpio7_12	Driver off
0x17BC	CTRL_CORE_PAD_SPI1_CS3	E25	spi1_cs3	uart4_txd	mmc3_sdwp	spi2_cs3	dcan2_rx	mdio_d	hdmi1_cec								gpio7_13	Driver off
0x17C0	CTRL_CORE_PAD_SPI2_SCLK	G25	spi2_sclk	uart3_rxd													gpio7_14	Driver off
0x17C4	CTRL_CORE_PAD_SPI2_D1	F25	spi2_d1	uart3_txd													gpio7_15	Driver off
0x17C8	CTRL_CORE_PAD_SPI2_D0	G24	spi2_d0	uart3_ctsn	uart5_rxd												gpio7_16	Driver off
0x17CC	CTRL_CORE_PAD_SPI2_CS0	F24	spi2_cs0	uart3_rtsn	uart5_txd												gpio7_17	Driver off

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x17D0	CTRL_CORE_PAD_DCAN1_TX	H22	dcan1_tx		uart8_rxd	mmc2_sdcd			hdmi1_hpd								gpio1_14	Driver off
0x17D4	CTRL_CORE_PAD_DCAN1_RX	H23	dcan1_rx		uart8_txd	mmc2_sdwp			hdmi1_cec								gpio1_15	Driver off
0x17E0	CTRL_CORE_PAD_UART1_RXD	L25	uart1_rxd			mmc4_sdcd											gpio7_22	Driver off
0x17E4	CTRL_CORE_PAD_UART1_TXD	M25	uart1_txd			mmc4_sdwp											gpio7_23	Driver off
0x17E8	CTRL_CORE_PAD_UART1_CTSN	L20	uart1_ctsn		uart9_rxd	mmc4_clk											gpio7_24	Driver off
0x17EC	CTRL_CORE_PAD_UART1_RTSN	M24	uart1_rtsn		uart9_txd	mmc4_cmd											gpio7_25	Driver off
0x17F0	CTRL_CORE_PAD_UART2_RXD	N23	uart2_rxd	uart3_ctsn	uart3_rctx	mmc4_dat0	uart2_rxd	uart1_dcdn									gpio7_26	Driver off
0x17F4	CTRL_CORE_PAD_UART2_TXD	N25	uart2_txd	uart3_rtsn	uart3_sd	mmc4_dat1	uart2_txd	uart1_dsrn									gpio7_27	Driver off
0x17F8	CTRL_CORE_PAD_UART2_CTSN	N22	uart2_ctsn		uart3_rxd	mmc4_dat2	uart10_rxd	uart1_dtrn									gpio1_16	Driver off
0x17FC	CTRL_CORE_PAD_UART2_RTSN	N24	uart2_rtsn	uart3_txd	uart3_irtx	mmc4_dat3	uart10_txd	uart1_rin									gpio1_17	Driver off
0x1800	CTRL_CORE_PAD_I2C1_SDA	G23	i2c1_sda															Driver off
0x1804	CTRL_CORE_PAD_I2C1_SCL	G22	i2c1_scl															Driver off
0x1808	CTRL_CORE_PAD_I2C2_SDA	F23	i2c2_sda	hdmi1_ddc_scl														Driver off
0x180C	CTRL_CORE_PAD_I2C2_SCL	G21	i2c2_scl	hdmi1_ddc_sda														Driver off
0x1818	CTRL_CORE_PAD_WAKEUP0	AC10		dcan1_rx													gpio1_0 sys_nirq2	Driver off
0x1824	CTRL_CORE_PAD_WAKEUP3	AB10		sys_nirq1													gpio1_3 dcan2_rx	Driver off
0x1830	CTRL_CORE_PAD_TMS	L21	tms															
0x1834	CTRL_CORE_PAD_TDI	L23	tdi														gpio8_27	
0x1838	CTRL_CORE_PAD_TDO	J20	tdo														gpio8_28	
0x183C	CTRL_CORE_PAD_TCLK	K21	tclk															
0x1840	CTRL_CORE_PAD_TRSTN	L22	trstn															
0x1844	CTRL_CORE_PAD_RTCK	K25	rtck														gpio8_29	
0x1848	CTRL_CORE_PAD_EMU0	C21	emu0														gpio8_30	
0x184C	CTRL_CORE_PAD_EMU1	C22	emu1														gpio8_31	

Table 4-32. Pin Multiplexing (continued)

ADDRESS	REGISTER NAME	BALL NUMBER	MUXMODE FIELD SETTINGS (CTRL_CORE_PAD_*[3:0])															
			0	1	2	3*	4*	5*	6*	7	8*	9	10	11	12	13	14*	15
0x185C	CTRL_CORE_PAD_RESETN	K24	resetn															
0x1860	CTRL_CORE_PAD_NMIN_DSP	L24	nmin_dsp															
0x1864	CTRL_CORE_PAD_RSTOUTN	E20	rstoutn															

1. NA in table stands for Not Applicable.

4.5 Connections for Unused Pins

This section describes the connection requirements of the unused and reserved balls.

NOTE

The following balls are reserved: K20, L19, G20

These balls must be left unconnected.

NOTE

All unused power supply balls must be supplied with the voltages specified in the Section 5.4, *Recommended Operating Conditions,* unless alternative tie-off options are included in Section 4.3, *Signal Descriptions*.

Table 4-33. Unused Balls Specific Connection Requirements

BALLS	CONNECTION REQUIREMENTS
Y12 / AC11 / L22 / AC10 / AB10 / AD22 / Y24 / V24 / R24	These balls must be connected to GND through an external pull resistor if unused.
K21 / L24 / K24 / G22 / G23 / L21 / G21 / F23 / AE22 / Y25 / V25 / R25	These balls must be connect to the corresponding power supply through an external pull resistor if unused.
F20 (vpp)	This ball must be left unconnected if unused

NOTE

All other unused signal balls with a Pad Configuration register can be left unconnected with their internal pullup or pulldown resistor enabled.

NOTE

All other unused signal balls without a Pad Configuration register can be left unconnected.

5 Specifications

NOTE

For more information, see Power, Reset, and Clock Management / PRCM Subsystem Environment / External Voltage Inputs or Initialization / Preinitialization / Power Requirements section of the Device TRM.

NOTE

The index number 1which is part of the EMIF1 signal prefixes (ddr1_*) listed in Table 4-6, *EMIF Signal Descriptions*, column "SIGNAL NAME" not to be confused with DDR1 type of SDRAM memories.

NOTE

Audio Back End (ABE) module is not supported for this family of devices, but “ABE” name is still present in some clock or DPLL names.

CAUTION

All IO Cells are NOT Fail-safe compliant and should not be externally driven in absence of their IO supply.

5.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted) (1)

PARAMETER(1)			MIN	MAX	UNIT
V_{SUPPLY} (Steady-State)	Supply Voltage Ranges (Steady-State)	Core (vdd, vdd_dsp)	-0.3	1.5	V
		Analog (vdda_usb1, vdda_usb2, vdda_per, vdda_ddr, vdda_debug, vdda_mpu_abe, vdda_usb3, vdda_csi, vdda_core_gmac, vdda_gpu, dda_hdmi, vdda_pcie, vdda_video, vdda_osc)	-0.3	2.0	V
		Analog 3.3V (vdda33v_usb1, vdda33v_usb2)	-0.3	3.8	V
		vdds18v, vdds18v_ddr1, vdds_mlbp, vdds_ddr1	-0.3	2.1	V
		vddshv1, vddshv3, vddshv4, vddshv7-11 (1.8V mode)	-0.3	2.1	V
		vddshv1, vddshv3, vddshv4, vddshv7, vddshv9-11 (3.3V mode)	-0.3	3.8	V
		vddshv8 (3.3V mode)	-0.3	3.6	V
V_{IO} (Steady-State)	Input and Output Voltage Ranges (Steady-State)	Core I/Os	-0.3	1.5	V
		Analog I/Os (except HDMI)	-0.3	2.0	V
		HDMI I/Os	-0.3	3.5	V
		I/O 1.35V	-0.3	1.65	V
		I/O 1.5V	-0.3	1.8	V
		1.8V I/Os	-0.3	2.1	V
		3.3V I/Os (except those powered by vddshv8)	-0.3	3.8	V
		3.3V I/Os (powered by vddshv8)	-0.3	3.6	V
SR	Maximum slew rate, all supplies			10^5	V/s
V_{IO} (Transient Overshoot / Undershoot)	Input and Output Voltage Ranges (Transient Overshoot/Undershoot) Note: valid for up to 20% of the signal period. See Figure 5-1, *IO transient voltage ranges*.			0.2 × VDD (4)	V
T_J	Operating junction temperature range	Automotive	-40	+125	°C
T_{STG}	Storage temperature range after soldered onto PC Board		-55	+150	°C
Latch-up I-Test	I-test(5), All I/Os (if different levels then one line per level)		-100	100	mA
Latch-up OV-Test	Over-voltage Test(6), All supplies (if different levels then one line per level)		N/A	1.5 × Vsupply max	V

(1) Stresses beyond those listed as absolute maximum ratings may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those listed under Section 5.4, *Recommended Operating Conditions*, is not implied. Exposure to absolute maximum rated conditions for extended periods may affect device reliability.

(2) All voltage values are with respect to VSS, unless otherwise noted.

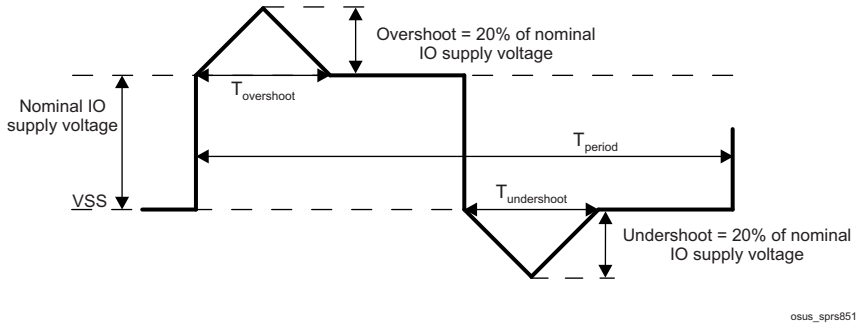
(3) See I/Os supplied by this power pin in Table 4-1 *Pin Attributes*

(4) VDD is the voltage on the corresponding power-supply pin(s) for the IO.

(5) Per JEDEC JESD78 at 125°C with specified I/O pin injection current and clamp voltage of 1.5 times maximum recommended I/O voltage and negative 0.5 times maximum recommended I/O voltage.

(6) Per JEDEC JESD78 at 125°C.

(7) The maximum valid input voltage on an IO pin cannot exceed 0.3 volts when the supply powering the IO is turned off. This requirement applies to all the IO pins which are not fail-safe and for all values of IO supply voltage. Special attention should be applied anytime peripheral devices are not powered from the same power sources used to power the respective IO supply. It is important the attached peripheral never sources a voltage outside the valid input voltage range, including power supply ramp-up and ramp-down sequences.



(1) $T_{overshoot} + T_{undershoot} < 20\%$ of T_{period}

Figure 5-1. IO transient voltage ranges

5.2 ESD Ratings

			VALUE	UNIT
V_{ESD} Electrostatic discharge	Human-Body model (HBM), per AEC Q100-002(1)		±1000	V
	Charged-device model (CDM), per AEC Q100-011	All pins	±250	
		Corner pins (A1, A25, AE1, AE25)	±750	

(1) AEC Q100-002 indicates that HBM stressing shall be in accordance with the ANSI/ESDA/JEDEC JS-001 specification.

5.3 Power on Hour (POH) Limits

IP	DUTY CYCLE	VOLTAGE DOMAIN	VOLTAGE (V) (MAX)	FREQUENCY (MHz) (MAX)	Tj(°C)	POH
All	100%	All	All Support OPPs		Automotive Profile(4)	20000

(1) The information in the section below is provided solely for your convenience and does not extend or modify the warranty provided under TI's standard terms and conditions for TI semiconductor products.

(2) POH is a functional of voltage, temperature and time. Usage at higher voltages and temperatures will result in a reduction in POH to achieve the same reliability performance. For assessment of alternate use cases, contact your local TI representative.

(3) Unless specified in the table above, all voltage domains and operating conditions are supported in the device at the noted temperatures.

(4) Automotive profile is defined as 20000 power on hours with junction temperature as follows: 5%@-40°C, 65%@70°C, 20%@110°C, 10%@125°C.

5.4 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

PARAMETER	DESCRIPTION	MIN (2)	NOM	MAX DC (3)	MAX (2)	UNIT
Input Power Supply Voltage Range						
vdd	Core voltage domain supply	See Section 5.5				V
vdd_dsp	DSP voltage domain supply	See Section 5.5				V
vdda_usb1	DPLL_USB and HS USB1 1.8V analog power supply	1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)		50			mV_{PPmax}
vdda_usb2	HS USB2 1.8V analog power supply	1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)		50			mV_{PPmax}
vdda33v_usb1(5)	HS USB1 3.3V analog power supply	3.135	3.3	3.366	3.465	V
	Maximum noise (peak-peak)		50			mV_{PPmax}
vdda33v_usb2(5)	HS USB2 3.3V analog power supply	3.135	3.3	3.366	3.465	V
	Maximum noise (peak-peak)		50			mV_{PPmax}
vdda_per	PER PLL and PER HSDIVIDER analog power supply	1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)		50			mV_{PPmax}

Recommended Operating Conditions *(continued)*

over operating free-air temperature range (unless otherwise noted)

PARAMETER	DESCRIPTION		MIN [2]	NOM	MAX DC [3]	MAX [2]	UNIT
vdda_ddr	DPLL_DDR and DDR HSDIVIDER analog power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_debug	DPLL_DEBUG analog power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_core_gmac	DPLL_CORE and CORE HSDIVIDER analog power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_gpu	DPLL_GPU analog power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_hdmi	PLL_HDMI and HDMI analog power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_pcie	DPLL_PCIe_REF and PCIe analog power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_usb3	DPLL_USB_OTG_SS and USB3.0 RX/TX analog power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_video	DPLL_VIDEO1 analog power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdds_mlbp	MLBP IO power supply		1.71	1.80		1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_mpu_abe	DPLL_MPU analog power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_osc	HFOSC analog power supply		1.71	1.80		1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdda_csi	CSI Interface 1.8v Supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdds18v	1.8V power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdds18v_ddr1	EMIF1 bias power supply		1.71	1.80	1.836	1.89	V
	Maximum noise (peak-peak)			50			mV_{PPmax}
vdds_ddr1	EMIF1 power supply (1.5V for DDR3 mode / 1.35V DDR3L mode)	1.35-V Mode	1.28	1.35	1.377	1.42	V
		1.5-V Mode	1.43	1.50	1.53	1.57	
	Maximum noise (peakpeak)	1.35-V Mode		50			mV_{PPmax}
		1.5-V Mode					
vddshv1	Dual Voltage (1.8V or 3.3V) power supply for the VIN2 Power Group pins	1.8-V Mode	1.71	1.80	1.836	1.89	V
		3.3-V Mode	3.135	3.30	3.366	3.465	
	Maximum noise (peak-peak)	1.8-V Mode		50			mV_{PPmax}
		3.3-V Mode					

Recommended Operating Conditions *(continued)*

over operating free-air temperature range (unless otherwise noted)

PARAMETER	DESCRIPTION		MIN [2]	NOM	MAX DC [3]	MAX [2]	UNIT
vddshv10	Dual Voltage (1.8V or 3.3V) power supply for the GPMC Power Group pins	1.8-V Mode	1.71	1.80	1.836	1.89	V
		3.3-V Mode	3.135	3.30	3.366	3.465	
	Maximum noise (peak-peak)	1.8-V Mode		50			mV_{PPmax}
		3.3-V Mode					
vddshv11	Dual Voltage (1.8V or 3.3V) power supply for the MMC2 Power Group pins	1.8-V Mode	1.71	1.80	1.836	1.89	V
		3.3-V Mode	3.135	3.30	3.366	3.465	
	Maximum noise (peak-peak)	1.8-V Mode		50			mV_{PPmax}
		3.3-V Mode					
vddshv3	Dual Voltage (1.8V or 3.3V) power supply for the GENERAL Power Group pins	1.8-V Mode	1.71	1.80	1.836	1.89	V
		3.3-V Mode	3.135	3.30	3.366	3.465	
	Maximum noise (peak-peak)	1.8-V Mode		50			mV_{PPmax}
		3.3-V Mode					
vddshv4	Dual Voltage (1.8V or 3.3V) power supply for the MMC4 Power Group pins	1.8-V Mode	1.71	1.80	1.836	1.89	V
		3.3-V Mode	3.135	3.30	3.366	3.465	
	Maximum noise (peak-peak)	1.8-V Mode		50			mV_{PPmax}
		3.3-V Mode					
vddshv7	Dual Voltage (1.8V or 3.3V) power supply for the WIFI Power Group pins	1.8-V Mode	1.71	1.80	1.836	1.89	V
		3.3-V Mode	3.135	3.30	3.366	3.465	
	Maximum noise (peak-peak)	1.8-V Mode		50			mV_{PPmax}
		3.3-V Mode					
vddshv8	Dual Voltage (1.8V or 3.3V) power supply for the MMC1 Power Group pins	1.8-V Mode	1.71	1.80	1.836	1.89	V
		3.3-V Mode	3.135	3.30	3.366	3.465	
	Maximum noise (peak-peak)	1.8-V Mode		50			mV_{PPmax}
		3.3-V Mode					
vddshv9	Dual Voltage (1.8V or 3.3V) power supply for the RGMII Power Group pins	1.8-V Mode	1.71	1.80	1.836	1.89	V
		3.3-V Mode	3.135	3.30	3.366	3.465	
	Maximum noise (peak-peak)	1.8-V Mode		50			mV_{PPmax}
		3.3-V Mode					
vss	Ground supply			0			V
vssa_osc0	OSC0 analog ground			0			V
vssa_osc1	OSC1 analog ground			0			V
T_J [1]	Operating junction temperature range	Automotive	-40			+125 [6]	°C
ddr1_vref0	Reference Power Supply EMIF1		0.5 × vdds_ddr1				V

(1) Refer to Power on Hours table for limitations.

(2) The voltage at the device ball should never be below the MIN voltage or above the MAX voltage for any amount of time. This requirement includes dynamic voltage events such as AC ripple, voltage transients, voltage dips, etc.

(3) The DC voltage at the device ball should never be above the MAX DC voltage to avoid impact on device reliability and lifetime POH (Power-On-Hours). The MAX DC voltage is defined as the highest allowed DC regulated voltage, without transients, seen at the ball.

(4) Logic functions and parameter values are not assured out of the range specified in the recommended operating conditions.

Recommended Operating Conditions *(continued)*

over operating free-air temperature range (unless otherwise noted)

(5) USB Analog supply also powers digital IO buffers. This supply cannot be tied to VSS if USB is unused since digital IO buffers must be powered during device operation.

(6) The TSHUT feature of the SoC resets the device by default when one of the on-die temp sensors reports 123 °C. This is intended to protect the device from exceeding 125 °C. Though not recommended, the TSHUT temperature threshold can be modified in software if other mechanisms are in place to avoid exceeding 125 °C. Refer to the device TRM for details on the TSHUT feature.

5.5 Operating Performance Points

This section describes the operating conditions of the device. This section also contains the description of each OPP (operating performance point) for processor clocks and device core clocks.

Table 5-1 describes the maximum supported frequency per speed grade for DRA79x devices.

Table 5-1. Speed Grade Maximum Frequency(1)

DEVICE SPEED	MAXIMUM FREQUENCY (MHz)				
	MPU	DSP	IPU	L3	DDR3 / DDR3L
DRA79xxA	300	500	106.4	266	532 (DDR-1066)
DRA79xxB	300	750	106.4	266	532 (DDR 1066)
DRA79xxD	500	750	212.8	266	532 (DDR 1066)
DRA79xxH	800	750	212.8	266	667 (DDR-1333)

(1) N/A stands for Not Applicable.

5.5.1 AVS and ABB Requirements

Adaptive Voltage Scaling (AVS) and Adaptive Body Biasing (ABB) are required on most of the vdd_* supplies as defined in Table 5-2.

Table 5-2. AVS and ABB Requirements per vdd_* Supply

SUPPLY	VOLTAGE DOMAIN	AVS REQUIRED?	ABB REQUIRED?
vdd	VD_CORE	Yes, for all OPPs	No
	VD_SGX	Yes, for all OPPs	Yes, for all OPPs
	VD_MPU	Yes, for all OPPs	Yes, for all OPPs
vdd_dsp	VD_DSP	Yes, for all OPPs	Yes, for all OPPs
	VD_IVA	Yes, for all OPPs	Yes, for all OPPs

5.5.2 Voltage And Core Clock Specifications

Table 5-3 shows the recommended OPP per voltage domain.

Table 5-3. Voltage Domains Operating Performance Points (1)

DOMAIN	CONDITION	OPP_NOM			OPP_HIGH			
		MIN (3)	NOM (2)	MAX (3)	MIN (3)	NOM (2)	MAX DC (4)	MAX (3)
VD_CORE (V) (8)	BOOT (Before AVS is enabled) (5)	1.11	1.15	1.2	Not Applicable			
	After AVS is enabled (5)	AVS Voltage (6) – 3.5%	AVS Voltage (6)	1.2	Not Applicable			
VD_DSP (V) (9)	BOOT (Before AVS is enabled) (5)	1.02	1.06	1.16	Not Applicable			
	After AVS is enabled (5)	AVS Voltage(6) – 3.5%	AVS Voltage (6)	1.2	AVS Voltage(6) – 3.5%	AVS Voltage (6)	AVS Voltage (6) +2%	AVS Voltage(6) + 5%

(1) The voltage ranges in this table are preliminary, and final voltage ranges may be different than shown. Systems should be designed with the ability to modify the voltage to comply with future recommendations.

(2) In a typical implementation, the power supply should target the NOM voltage.

(3) The voltage at the device ball should never be below the MIN voltage or above the MAX voltage for any amount of time. This requirement includes dynamic voltage events such as AC ripple, voltage transients, voltage dips, etc.

(4) The DC voltage at the device ball should never be above the MAX DC voltage to avoid impact on device reliability and lifetime POH (Power-On-Hours). The MAX DC voltage is defined as the highest allowed DC regulated voltage, without transients, seen at the ball.

(5) For all OPPs, AVS must be enabled to avoid impact on device reliability, lifetime POH (Power-On-Hours), and device power.

(6) The AVS voltages are device-dependent, voltage domain-dependent, and OPP-dependent. They must be read from the STD_FUSE_OPP Registers. For information about STD_FUSE_OPP Registers address, please refer to Control Module Section of the TRM. The power supply should be adjustable over the following ranges for each required OPP:
- OPP_NOM for DSP: 0.85 V – 1.15 V
- OPP_NOM for CORE: 0.85 V - 1.15 V
- OPP_HIGH: 1.01 V - 1.25 V

The AVS voltages will be within the above specified ranges.

(7) The power supply must be programmed with the AVS voltages for the CORE voltage domain, either just after the ROM boot or at the earliest possible time in the secondary boot loader before there is significant activity seen on these domains.

(8) The package routes VD_CORE (vdd) to the VD_MPU, VD_SGX, VD_CORE and VD_RTC domains on the die.

(9) The package routes VD_DSP (vdd_dsp) to the VD_DSPEVE and VD_IVA domains on the die.

Table 5-4 describes the standard processor clocks speed characteristics vs OPP of the device.

Table 5-4. Supported OPP vs Max Frequency (2)

DESCRIPTION	OPP_NOM	OPP_HIGH
	MAXIMUM FREQUENCY (MHz)	MAXIMUM FREQUENCY (MHz)
VD_CORE		
MPU_CLK	800	N/A
CORE_IPUx_CLK	212.8	N/A
L3_CLK	266	N/A
DDR3 / DDR3L	667 (DDR-1333)	N/A
VD_DSP		
DSP_CLK	600	750

(1) N/A stands for Not Applicable.

(2) Maximum supported frequency is limited according to the Device Speed Grade (see Table 5-1).

5.5.3 Maximum Supported Frequency

Device modules either receive their clock directly from an external clock input, directly from a PLL, or from a PRCM. Table 5-5 lists the clock source options for each module on this device, along with the maximum frequency that module can accept. To ensure proper module functionality, the device PLLs and dividers must be programmed not to exceed the maximum frequencies listed in this table.

Table 5-5. Maximum Supported Frequency

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
AES1	AES1_L3_CLK	Int	266	L4SEC_L3_GICLK	CORE_X2_CLK	DPLL_CORE
AES2	AES2_L3_CLK	Int	266	L4SEC_L3_GICLK	CORE_X2_CLK	DPLL_CORE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	**INPUT CLOCK NAME**	**CLOCK TYPE**	**MAX. CLOCK ALLOWED (MHz)**	**PRCM CLOCK NAME**	**PLL / OSC / SOURCE CLOCK NAME**	**PLL / OSC / SOURCE NAME**
ATL	ATL_ICLK_L3	Int	266	ATL_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	ATLPCLK	Func	266	ATL_GFCLK	CORE_X2_CLK	DPLL_CORE
					PER_ABE_X1_GFCLK	DPLL_ABE
					FUNC_32K_CLK	OSC0
					HDMI_CLK	DPLL_HDMI
					VIDEO1_CLK	DPLL_VIDEO1
COUNTER_32K	COUNTER_32K_FCLK	Func	0.032	FUNC_32K_CLK	SYS_CLK1/610	OSC0
	COUNTER_32K_ICLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
CTRL_MODULE_BANDGAP	L3INSTR_TS_GCLK	Int	4.8	L3INSTR_TS_GCLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
CTRL_MODULE_CORE	L4CFG_L4_GICLK	Int	133	L4CFG_L4_GICLK	CORE_X2_CLK	DPLL_CORE
CTRL_MODULE_WKUP	WKUPAON_GICLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
DCAN1	DCAN1_FCLK	Func	38.4	DCAN1_SYS_CLK	SYS_CLK1	OSC0
					SYS_CLK2	OSC1
	DCAN1_ICLK	Int	266	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
DCAN2	DCAN2_FCLK	Func	38.4	DCAN2_SYS_CLK	SYS_CLK1	OSC0
	DCAN2_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
DES3DES	DES_CLK_L3	Int	266	L4SEC_L3_GICLK	CORE_X2_CLK	DPLL_CORE
DLL	EMIF_DLL_FCLK	Func	EMIF_DLL_FCLK	EMIF_DLL_GCLK	EMIF_DLL_GCLK	DPLL_DDR
DLL_AGING	FCLK	Int	38.4	L3INSTR_DLL_AGING_GCLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
DMM	DMM_CLK	Int	266	EMIF_L3_GICLK	CORE_X2_CLK	DPLL_CORE
DPLL_DEBUG	SYSCLK	Int	38.4	EMU_SYS_CLK	SYS_CLK1	OSC0
DSP1	DSP1_FICLK	Int & Func	DSP_CLK	DSP1_GFCLK	DSP_GFCLK	DPLL_DSP

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
DSS	DSS_HDMI_CEC_CLK	Func	0.032	HDMI_CEC_GFCLK	SYS_CLK1/610	OSC0
	DSS_HDMI_PHY_CLK	Func	48	HDMI_PHY_GFCLK	FUNC_192M_CLK	DPLL_PER
	DSS_CLK	Func	192	DSS_GFCLK	DSS_CLK	DPLL_PER
	HDMI_CLKINP	Func	38.4	HDMI_DPLL_CLK	SYS_CLK1	OSC0
					SYS_CLK2	OSC1
	DSS_L3_ICLK	Int	266	DSS_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	VIDEO1_CLKINP	Func	38.4	VIDEO1_DPLL_CLK	SYS_CLK1	OSC0
					SYS_CLK2	OSC1
	VIDEO2_CLKINP	Func	38.4	VIDEO2_DPLL_CLK	SYS_CLK1	OSC0
					SYS_CLK2	OSC1
	DPLL_DSI1_A_CLK1	Func	209.3	N/A	HDMI_CLK	DPLL_HDMI
					VIDEO1_CLKOUT1	DPLL_VIDEO1
	DPLL_DSI1_B_CLK1	Func	209.3	N/A	VIDEO1_CLKOUT3	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
					DPLL_ABE_X2_CLK	DPLL_ABE
	DPLL_DSI1_C_CLK1	Func	209.3	N/A	HDMI_CLK	DPLL_HDMI
					VIDEO1_CLKOUT3	DPLL_VIDEO1
	DPLL_HDMI_CLK1	Func	185.6	N/A	HDMI_CLK	DPLL_HDMI
DSS DISPC	LCD1_CLK	Func	209.3	N/A	DPLL_DSI1_A_CLK1	See DSS data in the rows above
					DSS_CLK	
	LCD2_CLK	Func	209.3	N/A	DPLL_DSI1_B_CLK1	
					DSS_CLK	
	LCD3_CLK	Func	209.3	N/A	DPLL_DSI1_C_CLK1	
					DSS_CLK	
	F_CLK	Func	209.3	N/A	DPLL_DSI1_A_CLK1	
					DPLL_DSI1_B_CLK1	
					DPLL_DSI1_C_CLK1	
					DSS_CLK	
					DPLL_HDMI_CLK1	
EFUSE_CTRL_CUST	ocp_clk	Int	133	CUSTEFUSE_L4_GICLK	CORE_X2_CLK	DPLL_CORE
	sys_clk	Func	38.4	CUSTEFUSE_SYS_GFCLK	SYS_CLK1	OSC0
ELM	ELM_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
EMIF_OCP_FW	L3_CLK	Int	266	EMIF_L3_GICLK	CORE_X2_CLK	DPLL_CORE
EMIF_PHY1	EMIF_PHY1_FCLK	Func	DDR	EMIF_PHY_GCLK	EMIF_PHY_GCLK	DPLL_DDR
EMIF1	EMIF1_ICLK	Int	266	EMIF_L3_GICLK	CORE_X2_CLK	DPLL_CORE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
GMAC_SW	CPTS_RFT_CLK	Func	266	GMAC_RFT_CLK	PER_ABE_X1_GFCLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
					CORE_X2_CLK	DPLL_CORE
	MAIN_CLK	Int	125	GMAC_MAIN_CLK	GMAC_250M_CLK	DPLL_GMAC
	MHZ_250_CLK	Func	250	GMII_250MHZ_CLK	GMII_250MHZ_CLK	DPLL_GMAC
	MHZ_5_CLK	Func	5	RGMII_5MHZ_CLK	GMAC_RMII_HS_CLK	DPLL_GMAC
	MHZ_50_CLK	Func	50	RMII_50MHZ_CLK	GMAC_RMII_HS_CLK	DPLL_GMAC
	RMII1_MHZ_50_CLK	Func	50	RMII_50MHZ_CLK	GMAC_RMII_HS_CLK	DPLL_GMAC
	RMII2_MHZ_50_CLK	Func	50	RMII_50MHZ_CLK	GMAC_RMII_HS_CLK	DPLL_GMAC
GPIO1	GPIO1_ICLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
	GPIO1_DBCLK	Func	0.032	WKUPAON_SYS_GFCLK	WKUPAON_32K_GFCLK	OSC0
GPIO2	GPIO2_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	GPIO2_DBCLK	Func	0.032	GPIO_GFCLK	FUNC_32K_CLK	OSC0
GPIO3	GPIO3_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	GPIO3_DBCLK	Func	0.032	GPIO_GFCLK	FUNC_32K_CLK	OSC0
GPIO4	GPIO4_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	GPIO4_DBCLK	Func	0.032	GPIO_GFCLK	FUNC_32K_CLK	OSC0
	PIDBCLK	Func	0.032	GPIO_GFCLK		
GPIO5	GPIO5_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	GPIO5_DBCLK	Func	0.032	GPIO_GFCLK	FUNC_32K_CLK	OSC0
	PIDBCLK	Func	0.032	GPIO_GFCLK		
GPIO6	GPIO6_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	GPIO6_DBCLK	Func	0.032	GPIO_GFCLK	FUNC_32K_CLK	OSC0
	PIDBCLK	Func	0.032	GPIO_GFCLK		
GPIO7	GPIO7_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	GPIO7_DBCLK	Func	0.032	GPIO_GFCLK	FUNC_32K_CLK	OSC0
	PIDBCLK	Func	0.032	GPIO_GFCLK		
GPIO8	GPIO8_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	GPIO8_DBCLK	Func	0.032	GPIO_GFCLK	FUNC_32K_CLK	OSC0
	PIDBCLK	Func	0.032	GPIO_GFCLK		
GPMC	GPMC_FCLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
HDMI PHY	DSS_HDMI_PHY_CLK	Func	38.4	HDMI_PHY_GFCLK	FUNC_192M_CLK	DPLL_PER
HDQ1W	HDQ1W_ICLK	Int & Func	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	HDQ1W_FCLK	Func	12	PER_12M_GFCLK	FUNC_192M_CLK	DPLL_PER
I2C1	I2C1_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	I2C1_FCLK	Func	96	PER_96M_GFCLK	FUNC_192M_CLK	DPLL_PER

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
I2C2	I2C2_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	I2C2_FCLK	Func	96	PER_96M_GFCLK	FUNC_192M_CLK	DPLL_PER
I2C3	I2C3_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	I2C3_FCLK	Func	96	PER_96M_GFCLK	FUNC_192M_CLK	DPLL_PER
I2C4	I2C4_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	I2C4_FCLK	Func	96	PER_96M_GFCLK	FUNC_192M_CLK	DPLL_PER
I2C5	I2C5_ICLK	Int	266	IPU_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	I2C5_FCLK	Func	96	IPU_96M_GFCLK	FUNC_192M_CLK	DPLL_PER
I2C6	I2C6_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	I2C6_FCLK	Func	96	IPU_96M_GFCLK	FUNC_192M_CLK	DPLL_PER
IEEE1500_2_OCP	PI_L3CLK	Int & Func	266	L3INIT_L3_GICLK	CORE_X2_CLK	DPLL_CORE
IPU1	IPU1_GFCLK	Int & Func	425.6	IPU1_GFCLK	DPLL_ABE_X2_CLK	DPLL_ABE
					CORE_IPU_ISS_BOOST_CLK	DPLL_CORE
IPU2	IPU2_GFCLK	Int & Func	425.6	IPU2_GFCLK	CORE_IPU_ISS_BOOST_CLK	DPLL_CORE
KBD	KBD_FCLK	Func	0.032	WKUPAON_SYS_GFCLK	WKUPAON_32K_GFCLK	OSC0
	PICLKKBD	Func	0.032	WKUPAON_SYS_GFCLK		
	KBD_ICLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
	PICLKOCP	Int	38.4	WKUPAON_GICLK	DPLL_ABE_X2_CLK	DPLL_ABE
L3_INSTR	L3_CLK	Int	L3_CLK	L3INSTR_L3_GICLK	CORE_X2_CLK	DPLL_CORE
L3_MAIN	L3_CLK1	Int	L3_CLK	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	L3_CLK2	Int	L3_CLK	L3INSTR_L3_GICLK	CORE_X2_CLK	DPLL_CORE
L4_CFG	L4_CFG_CLK	Int	133	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
L4_PER1	L4_PER1_CLK	Int	133	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
L4_PER2	L4_PER2_CLK	Int	133	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
L4_PER3	L4_PER3_CLK	Int	133	L4PER3_L3_GICLK	CORE_X2_CLK	DPLL_CORE
L4_WKUP	L4_WKUP_CLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
MAILBOX1	MAILBOX1_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX2	MAILBOX2_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX3	MAILBOX3_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX4	MAILBOX4_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX5	MAILBOX5_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX6	MAILBOX6_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX7	MAILBOX7_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX8	MAILBOX8_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX9	MAILBOX9_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX10	MAILBOX10_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX11	MAILBOX11_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MAILBOX12	MAILBOX12_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
MAILBOX13	MAILBOX13_FLCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
McASP1	MCASP1_AHCLKR	Func	100	MCASP1_AHCLKR	ABE_24M_GFCLK	DPLL_ABE
					ABE_SYS_CLK	OSC0
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
	MCASP1_AHCLKX	Func	100	MCASP1_AHCLKX	ABE_24M_GFCLK	DPLL_ABE
					ABE_SYS_CLK	OSC0
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
	MCASP1_FCLK	Func	192	MCASP1_AUX_GFCLK	PER_ABE_X1_GFCLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
	MCASP1_ICLK	Int	266	IPU_L3_GICLK	CORE_X2_CLK	DPLL_CORE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
McASP2	MCASP2_AHCLKR	Func	100	MCASP2_AHCLKR	ABE_24M_GFCLK	DPLL_ABE
					ABE_SYS_CLK	OSC0
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
	MCASP2_AHCLKX	Func	100	MCASP2_AHCLKX	ABE_24M_GFCLK	DPLL_ABE
					ABE_SYS_CLK	OSC0
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
	MCASP2_FCLK	Func	192	MCASP2_AUX_GFCLK	PER_ABE_X1_GFCLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
	MCASP2_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
McASP3	MCASP3_AHCLKX	Func	100	MCASP3_AHCLKX	ABE_24M_GFCLK	DPLL_ABE
					ABE_SYS_CLK	OSC0
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
	MCASP3_FCLK	Func	192	MCASP3_AUX_GFCLK	PER_ABE_X1_GFCLK	DPLL_ABE
					VIDEO1_CLK	DPLL_ABE
					HDMI_CLK	DPLL_HDMI
	MCASP3_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
McASP4	MCASP4_AHCLKX	Func	100	MCASP4_AHCLKX	ABE_24M_GFCLK	DPLL_ABE
					ABE_SYS_CLK	OSC0
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
	MCASP4_FCLK	Func	192	MCASP4_AUX_GFCLK	PER_ABE_X1_GFCLK	DPLL_ABE
					VIDEO1_CLK	DPLL_ABE
					HDMI_CLK	DPLL_HDMI
	MCASP4_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
McASP5	MCASP5_AHCLKX	Func	100	MCASP5_AHCLKX	ABE_24M_GFCLK	DPLL_ABE
					ABE_SYS_CLK	OSC0
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
	MCASP5_FCLK	Func	192	MCASP5_AUX_GFCLK	PER_ABE_X1_GFCLK	DPLL_ABE
					VIDEO1_CLK	DPLL_ABE
					HDMI_CLK	DPLL_HDMI
	MCASP5_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
McASP6	MCASP6_AHCLKX	Func	100	MCASP6_AHCLKX	ABE_24M_GFCLK	DPLL_ABE
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
					ABE_SYS_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
	MCASP6_FCLK	Func	192	MCASP6_AUX_GFCLK	PER_ABE_X1_GFCLK	DPLL_ABE
					VIDEO1_CLK	DPLL_ABE
					HDMI_CLK	DPLL_HDMI
	MCASP6_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
McASP7	MCASP7_AHCLKX	Func	100	MCASP7_AHCLKX	ABE_24M_GFCLK	DPLL_ABE
					ABE_SYS_CLK	OSC0
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
	MCASP7_FCLK	Func	192	MCASP7_AUX_GFCLK	PER_ABE_X1_GFCLK	DPLL_ABE
					VIDEO1_CLK	DPLL_ABE
					HDMI_CLK	DPLL_HDMI
	MCASP7_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
McASP8	MCASP8_AHCLKX	Func	100	MCASP8_AHCLKX	ABE_24M_GFCLK	DPLL_ABE
					ABE_SYS_CLK	OSC0
					FUNC_24M_GFCLK	DPLL_PER
					ATL_CLK0	Module ATL
					ATL_CLK1	Module ATL
					ATL_CLK2	Module ATL
					ATL_CLK3	Module ATL
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					MLB_CLK	Module MLB
					MLBP_CLK	Module MLB
	MCASP8_FCLK	Func	192	MCASP8_AUX_GFCLK	PER_ABE_X1_GFCLK	DPLL_ABE
					VIDEO1_CLK	DPLL_ABE
					HDMI_CLK	DPLL_HDMI
	MCASP8_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
McSPI1	SPI1_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	SPI1_FCLK	Func	48	PER_48M_GFCLK	PER_48M_GFCLK	DPLL_PER
McSPI2	SPI2_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	SPI2_FCLK	Func	48	PER_48M_GFCLK	PER_48M_GFCLK	DPLL_PER
McSPI3	SPI3_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	SPI3_FCLK	Func	48	PER_48M_GFCLK	PER_48M_GFCLK	DPLL_PER

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
McSPI4	SPI4_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	SPI4_FCLK	Func	48	PER_48M_GFCLK	PER_48M_GFCLK	DPLL_PER
MLB_SS	MLB_L3_ICLK	Int	266	MLB_SHB_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	MLB_L4_ICLK	Int	133	MLB_SPB_L4_GICLK	CORE_X2_CLK	DPLL_CORE
	MLB_FCLK	Func	266	MLB_SYS_L3_GFCLK	CORE_X2_CLK	DPLL_CORE
CSI2_0	CTRLCLK	Int & Func	96	LVDSRX_96M_GFCLK	FUNC_192M_CLK	DPLL_PER
	CAL_FCLK	Int & Func	266	CAL_GICLK	CORE_ISS_MAIN_CLK	DPLL_CORE
					L3_ICLK	CM_CORE_AON
MMC1	MMC1_CLK_32K	Func	0.032	L3INIT_32K_GFCLK	FUNC_32K_CLK	OSC0
	MMC1_FCLK	Func	192	MMC1_GFCLK	FUNC_192M_CLK	DPLL_PER
			128		FUNC_256M_CLK	DPLL_PER
	MMC1_ICLK1	Int	266	L3INIT_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	MMC1_ICLK2	Int	133	L3INIT_L4_GICLK	CORE_X2_CLK	DPLL_CORE
MMC2	MMC2_CLK_32K	Func	0.032	L3INIT_32K_GFCLK	FUNC_32K_CLK	OSC0
	MMC2_FCLK	Func	192	MMC2_GFCLK	FUNC_192M_CLK	DPLL_PER
			128		FUNC_256M_CLK	DPLL_PER
	MMC2_ICLK1	Int	266	L3INIT_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	MMC2_ICLK2	Int	133	L3INIT_L4_GICLK	CORE_X2_CLK	DPLL_CORE
MMC3	MMC3_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	MMC3_CLK_32K	Func	0.032	L4PER_32K_GFCLK	FUNC_32K_CLK	OSC0
	MMC3_FCLK	Func	48	MMC3_GFCLK	FUNC_192M_CLK	DPLL_PER
			192			
MMC4	MMC4_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	MMC4_CLK_32K	Func	0.032	L4PER_32K_GFCLK	FUNC_32K_CLK	OSC0
	MMC4_FCLK	Func	48	MMC4_GFCLK	FUNC_192M_CLK	DPLL_PER
			192			
MMU_EDMA	MMU1_CLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MMU_PCIESS	MMU2_CLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
MPU	MPU_CLK	Int & Func	MPU_CLK	MPU_GCLK	MPU_GCLK	DPLL_MPU
MPU_EMU_DBG	FCLK	Int	38.4	EMU_SYS_CLK	SYS_CLK1	OSC0
					MPU_GCLK	DPLL_MPU
OCMC_RAM1	OCMC1_L3_CLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
OCMC_ROM	OCMC_L3_CLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
OCP_WP_NOC	PICLKOCPL3	Int	266	L3INSTR_L3_GICLK	CORE_X2_CLK	DPLL_CORE
OCP2SCP1	L4CFG1_ADAPTER_CLKIN	Int	133	L3INIT_L4_GICLK	CORE_X2_CLK	DPLL_CORE
OCP2SCP2	L4CFG2_ADAPTER_CLKIN	Int	133	L4CFG_L4_GICLK	CORE_X2_CLK	DPLL_CORE
OCP2SCP3	L4CFG3_ADAPTER_CLKIN	Int	133	L3INIT_L4_GICLK	CORE_X2_CLK	DPLL_CORE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
PCIESS1	PCIE1_PHY_WKUP_CLK	Func	0.032	PCIE_32K_GFCLK	FUNC_32K_CLK	DPLL_CORE
	PCIe_SS1_FICLK	Int	266	PCIE_L3_GICLK	CORE_X2_CLK	
	PCIEPHY_CLK	Func	2500	PCIE_PHY_GCLK	PCIE_PHY_GCLK	APLL_PCIE
	PCIEPHY_CLK_DIV	Func	1250	PCIE_PHY_DIV_GCLK	PCIE_PHY_DIV_GCLK	APLL_PCIE
	PCIE1_REF_CLKIN	Func	34.3	PCIE_REF_GFCLK	CORE_USB_OTG_SS_LFPS_TX_CLK	DPLL_CORE
	PCIE1_PWR_CLK	Func	38.4	PCIE_SYS_GFCLK	SYS_CLK1	OSC0
PCIESS2	PCIE2_PHY_WKUP_CLK	Func	0.032	PCIE_32K_GFCLK	FUNC_32K_CLK	
	PCIe_SS2_FICLK	Func	266	PCIE_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	PCIEPHY_CLK	Func	2500	PCIE_PHY_GCLK	PCIE_PHY_GCLK	APLL_PCIE
	PCIEPHY_CLK_DIV	Func	1250	PCIE_PHY_DIV_GCLK	PCIE_PHY_DIV_GCLK	APLL_PCIE
	PCIE2_REF_CLKIN	Func	34.3	PCIE_REF_GFCLK	CORE_USB_OTG_SS_LFPS_TX_CLK	DPLL_CORE
	PCIE2_PWR_CLK	Func	38.4	PCIE_SYS_GFCLK	SYS_CLK1	OSC0
PRCM_MPU	32K_CLK	Func	0.032	FUNC_32K_CLK	SYS_CLK1/610	OSC0
	SYS_CLK	Func	38.4	WKUPAON_ICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
PWMSS1	PWMSS1_GICLK	Int & Func	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
PWMSS2	PWMSS2_GICLK	Int & Func	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
PWMSS3	PWMSS3_GICLK	Int & Func	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
QSPI	QSPI_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	QSPI_FCLK	Func	128	QSPI_GFCLK	FUNC_256M_CLK	DPLL_PER
					PER_QSPI_CLK	DPLL_PER
RNG	RNG_ICLK	Int	266	L4SEC_L3_GICLK	CORE_X2_CLK	DPLL_CORE
SAR_ROM	PRCM_ROM_CLOCK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
SDMA	SDMA_FCLK	Int & Func	266	DMA_L3_GICLK	CORE_X2_CLK	DPLL_CORE
SHA2MD51	SHAM_1_CLK	Int	266	L4SEC_L3_GICLK	CORE_X2_CLK	DPLL_CORE
SHA2MD52	SHAM_2_CLK	Int	266	L4SEC_L3_GICLK	CORE_X2_CLK	DPLL_CORE
SL2	IVA_GCLK	Int	IVA_GCLK	IVA_GCLK	IVA_GFCLK	DPLL_IVA
SMARTREFLEX_CORE	MCLK	Int	133	COREAON_L4_GICLK	CORE_X2_CLK	DPLL_CORE
	SYSCLK	Func	38.4	WKUPAON_ICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
SMARTREFLEX_DSP	MCLK	Int	133	COREAON_L4_GICLK	CORE_X2_CLK	DPLL_CORE
	SYSCLK	Func	38.4	WKUPAON_ICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
SMARTREFLEX_IVAHD	MCLK	Int	133	COREAON_L4_GICLK	CORE_X2_CLK	DPLL_CORE
	SYSCLK	Func	38.4	WKUPAON_ICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
SMARTREFLEX_MPU	MCLK	Int	133	COREAON_L4_GICLK	CORE_X2_CLK	DPLL_CORE
	SYSCLK	Func	38.4	WKUPAON_ICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
SPINLOCK	SPINLOCK_ICLK	Int	266	L4CFG_L3_GICLK	CORE_X2_CLK	DPLL_CORE
TIMER1	TIMER1_ICLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
	TIMER1_FCLK	Func	100	TIMER1_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
TIMER2	TIMER2_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER2_FCLK	Func	100	TIMER2_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
TIMER3	TIMER3_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER3_FCLK	Func	100	TIMER3_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
TIMER4	TIMER4_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER4_FCLK	Func	100	TIMER4_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
TIMER5	TIMER5_ICLK	Int	266	IPU_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER5_FCLK	Func	100	TIMER5_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
					CLKOUTMUX[0]	CLKOUTMUX[0]
TIMER6	TIMER6_ICLK	Int	266	IPU_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER6_FCLK	Func	100	TIMER6_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
					CLKOUTMUX[0]	CLKOUTMUX[0]

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
TIMER7	TIMER7_ICLK	Int	266	IPU_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER7_FCLK	Func	100	TIMER7_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
					CLKOUTMUX[0]	CLKOUTMUX[0]
TIMER8	TIMER8_ICLK	Int	266	IPU_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER8_FCLK	Func	100	TIMER8_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
					CLKOUTMUX[0]	CLKOUTMUX[0]
TIMER9	TIMER9_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER9_FCLK	Func	100	TIMER9_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
TIMER10	TIMER10_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER10_FCLK	Func	100	TIMER10_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
TIMER11	TIMER11_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER11_FCLK	Func	100	TIMER11_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
TIMER12	TIMER12_ICLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
	TIMER12_FCLK	Func	0.032	OSC_32K_CLK	RC_CLK	RC oscillator
TIMER13	TIMER13_ICLK	Int	266	L4PER3_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER13_FCLK	Func	100	TIMER13_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
TIMER14	TIMER14_ICLK	Int	266	L4PER3_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER14_FCLK	Func	100	TIMER14_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
TIMER15	TIMER15_ICLK	Int	266	L4PER3_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER15_FCLK	Func	100	TIMER15_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
TIMER16	TIMER16_ICLK	Int	266	L4PER3_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	TIMER16_FCLK	Func	100	TIMER16_GFCLK	SYS_CLK1	OSC0
					FUNC_32K_CLK	OSC0
					SYS_CLK2	OSC1
					XREF_CLK0	XREF_CLK0
					XREF_CLK1	XREF_CLK1
					XREF_CLK2	XREF_CLK2
					XREF_CLK3	XREF_CLK3
					DPLL_ABE_X2_CLK	DPLL_ABE
					VIDEO1_CLK	DPLL_VIDEO1
					HDMI_CLK	DPLL_HDMI
TPCC	TPCC_GCLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
TPTC1	TPTC0_GCLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
TPTC2	TPTC1_GCLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
UART1	UART1_FCLK	Func	48	UART1_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART1_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
UART2	UART2_FCLK	Func	48	UART2_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART2_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
UART3	UART3_FCLK	Func	48	UART3_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART3_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
UART4	UART4_FCLK	Func	48	UART4_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART4_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
UART5	UART5_FCLK	Func	48	UART5_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART5_ICLK	Int	266	L4PER_L3_GICLK	CORE_X2_CLK	DPLL_CORE
UART6	UART6_FCLK	Func	48	UART6_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART6_ICLK	Int	266	IPU_L3_GICLK	CORE_X2_CLK	DPLL_CORE
UART7	UART7_FCLK	Func	48	UART7_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART7_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
UART8	UART8_FCLK	Func	48	UART8_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART8_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
UART9	UART9_FCLK	Func	48	UART9_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART9_ICLK	Int	266	L4PER2_L3_GICLK	CORE_X2_CLK	DPLL_CORE
UART10	UART10_FCLK	Func	48	UART10_GFCLK	FUNC_192M_CLK	DPLL_PER
	UART10_ICLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
USB1	USB1_MICLK	Int	266	L3INIT_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	USB3PHY_REF_CLK	Func	34.3	USB_LFPS_TX_GFCLK	CORE_USB_OTG_SS_LFPS_TX_CLK	DPLL_CORE
	USB2PHY1_TREF_CLK	Func	38.4	USB_OTG_SS_REF_CLK	SYS_CLK1	OSC0
	USB2PHY1_REF_CLK	Func	960	L3INIT_960M_GFCLK	L3INIT_960_GFCLK	DPLL_USB
USB2	USB2_MICLK	Int	266	L3INIT_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	USB2PHY2_TREF_CLK	Func	38.4	USB_OTG_SS_REF_CLK	SYS_CLK1	OSC0
	USB2PHY2_REF_CLK	Func	960	L3INIT_960M_GFCLK	L3INIT_960_GFCLK	DPLL_USB
USB3	USB3_MICLK	Int	266	L3INIT_L3_GICLK	CORE_X2_CLK	DPLL_CORE
	USB3PHY_PWRS_CLK	Func	38.4	USB_OTG_SS_REF_CLK	SYS_CLK1	OSC0
USB_PHY1_CORE	USB2PHY1_WKUP_CLK	Func	0.032	COREAON_32K_GFCLK	SYS_CLK1/610	OSC0
USB_PHY2_CORE	USB2PHY2_WKUP_CLK	Func	0.032	COREAON_32K_GFCLK	SYS_CLK1/610	OSC0
USB_PHY3_CORE	USB3PHY_WKUP_CLK	Func	0.032	COREAON_32K_GFCLK	SYS_CLK1/610	OSC0
VCP1	VCP1_CLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
VCP2	VCP2_CLK	Int	266	L3MAIN1_L3_GICLK	CORE_X2_CLK	DPLL_CORE
VIP1	L3_CLK_PROC_CLK	Int & Func	266	VIP1_GCLK	CORE_X2_CLK	DPLL_CORE
					CORE_ISS_MAIN_CLK	DPLL_CORE
VPE	L3_CLK_PROC_CLK	Int & Func	300	VPE_GCLK	CORE_ISS_MAIN_CLK	DPLL_CORE
					VIDEO1_CLKOUT4	DPLL_VIDEO1
WD_TIMER1	PIOCPCLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
	PITIMERCLK	Func	0.032	OSC_32K_CLK	RC_CLK	RC oscillator

Table 5-5. Maximum Supported Frequency (continued)

MODULE				CLOCK SOURCES		
INSTANCE NAME	INPUT CLOCK NAME	CLOCK TYPE	MAX. CLOCK ALLOWED (MHz)	PRCM CLOCK NAME	PLL / OSC / SOURCE CLOCK NAME	PLL / OSC / SOURCE NAME
WD_TIMER2	WD_TIMER2_ICLK	Int	38.4	WKUPAON_GICLK	SYS_CLK1	OSC0
					DPLL_ABE_X2_CLK	DPLL_ABE
	WD_TIMER2_FCLK	Func	0.032	WKUPAON_SYS_GFCLK	WKUPAON_32K_GFCLK	

5.6 Power Consumption Summary

NOTE

Maximum power consumption for this SoC depends on the specific use conditions for the end system. Contact your TI representative for assistance in estimating maximum power consumption for the end system use case.

5.7 Electrical Characteristics

NOTE

The data specified in Section 5.7 through Section 5.7.3 are subject to change.

NOTE

The interfaces or signals described in Section 5.7 through Section 5.7.3 correspond to the interfaces or signals available in multiplexing mode 0 (Function 1).

All interfaces or signals multiplexed on the balls described in these tables have the same DC electrical characteristics, unless multiplexing involves a PHY/GPIO combination in which case different DC electrical characteristics are specified for the different multiplexing modes (Functions).

Table 5-6. LVCMOS DDR DC Electrical Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
Signal Names in MUXMODE 0 (Single-Ended Signals): ddr1_d[31:0], ddr1_a[15:0], ddr1_dqm[3:0], ddr1_ba[2:0], ddr1_csn0, ddr1_cke, ddr1_odt0, ddr1_casn, ddr1_rasn, ddr1_wen, ddr1_rst					
Balls: AA23 / AC24 / AB24 / AD24 / AB23 / AC23 / AD23 / AE24 / AA24 / W25 / Y23 / AD25 / AC25 / AB25 / AA25 / W24 / W23 / U25 / U24 / W21 / T22 / U22 / U23 / T21 / T23 / T25 / T24 / P21 / N21 / P22 / P23 / P24 / AC18 / AE19 / AD19 / AB19 / AD20 / AE20 / AA18 / AA20 / Y21 / AC20 / AA21 / AC21 / AC22 / AC15 / AB15 / AC16 / AE23 / W22 / U21 / P25 / AE16 / AA16 / AB16 / AC19 / AB18 / AD18 / AD16 / AD17 / AE18 / AE17					
Driver Mode					
V_{OH}	High-level output threshold (I_{OH} = 0.1 mA)	0.9 × VDDS			V
V_{OL}	Low-level output threshold (I_{OL} = 0.1 mA)			0.1 × VDDS	V
C_{PAD}	Pad capacitance (including package capacitance)			3	pF

Table 5-6. LVCMOS DDR DC Electrical Characteristics (continued)

over operating free-air temperature range (unless otherwise noted)

PARAMETER			MIN	NOM	MAX	UNIT
Z_O	Output impedance (drive strength)	I[2:0] = 000 (Imp80)		80		Ω
		I[2:0] = 001 (Imp60)		60		
		I[2:0] = 010 (Imp48)		48		
		I[2:0] = 011 (Imp40)		40		
		I[2:0] = 100 (Imp34)		34		
Single-Ended Receiver Mode						
V_{IH}	High-level input threshold	DDR3/DDR3L	VREF+0.1		VDDS+0.2	V
V_{IL}	Low-level input threshold	DDR3/DDR3L	-0.2		VREF-0.1	V
V_{CM}	Input common-mode voltage		VREF -10%vdds		VREF+ 10%vdds	V
C_{PAD}	Pad capacitance (including package capacitance)				3	pF
Signal Names in MUXMODE 0 (Differential Signals): ddr1_ck, ddr1_nck, ddr1_dqs[3:0], ddr1_dqsn[3:0]						
Bottom Balls: AD21 / AE21 / AD22 / AE22 / Y24 / Y25 / V24 / V25 / R24 / R25						
Driver Mode						
V_{OH}	High-level output threshold (I_{OH} = 0.1 mA)		0.9 × VDDS			V
V_{OL}	Low-level output threshold (I_{OL} = 0.1 mA)				0.1 × VDDS	V
C_{PAD}	Pad capacitance (including package capacitance)				3	pF
Z_O	Output impedance (drive strength)	I[2:0] = 000 (Imp80)		80		Ω
		I[2:0] = 001 (Imp60)		60		
		I[2:0] = 010 (Imp48)		48		
		I[2:0] = 011 (Imp40)		40		
		I[2:0] = 100 (Imp34)		34		
Single-Ended Receiver Mode						
V_{IH}	High-level input threshold	DDR3/DDR3L	VREF+0.1		VDDS+0.2	V
V_{IL}	Low-level input threshold	DDR3/DDR3L	-0.2		VREF-0.1	V
V_{CM}	Input common-mode voltage		VREF -10%vdds		VREF+ 10%vdds	V
C_{PAD}	Pad capacitance (including package capacitance)				3	pF
Differential Receiver Mode						
V_{SWING}	Input voltage swing	DDR3/DDR3L	0.2		vdds+0.4	V
V_{CM}	Input common-mode voltage		VREF -10%vdds		VREF+ 10%vdds	V
C_{PAD}	Pad capacitance (including package capacitance)				3	pF

(1) VDDS stands for corresponding power supply (that is, vdds_ddr1). For more information on the power supply name and the corresponding ball, see Table 4-1, POWER [11] column.

(2) VREF in this table stands for corresponding Reference Power Supply (that is, ddr1_vref0). For more information on the power supply name and the corresponding ball, see Table 4-1, POWER [11] column.

(3) For more information on the I/O cell configurations (i[2:0], sr[1:0]), see the Chapter *Control Module* of the Device TRM.

Table 5-7. Dual Voltage LVCMOS I²C DC Electrical Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
Signal Names in MUXMODE 0: i2c1_scl; i2c1_sda; i2c2_scl; i2c_sda					
Balls: G22 / G23 / G21 / F23					
I²C Standard Mode – 1.8 V					
V_{IH}	Input high-level threshold	0.7 × VDDS			V
V_{IL}	Input low-level threshold			0.3 × VDDS	V
V_{hys}	Hysteresis	0.1 × VDDS			V
I_{IN}	Input current at each I/O pin with an input voltage between 0.1 × VDDS to 0.9 × VDDS			12	µA
I_{OZ}	I_{OZ}(I_{PAD} Current) for BIDI cell. This current is contributed by the tristated driver leakage + input current of the Rx + weak pullup/pulldown leakage. PAD is swept from 0 to VDDS and the Max($I_{(PAD)}$) is measured and is reported as I_{OZ}			12	µA
C_{IN}	Input capacitance			10	pF
V_{OL3}	Output low-level threshold open-drain at 3-mA sink current			0.2 × VDDS	V
I_{OLmin}	Low-level output current @V_{OL}=0.2 × VDDS	3			mA
t_{OF}	Output fall time from V_{IHmin} to V_{ILmax} with a bus capacitance CB from 5 pF to 400 pF			250	ns
I²C Fast Mode – 1.8 V					
V_{IH}	Input high-level threshold	0.7 × VDDS			V
V_{IL}	Input low-level threshold			0.3 × VDDS	V
V_{hys}	Hysteresis	0.1 × VDDS			V
I_{IN}	Input current at each I/O pin with an input voltage between 0.1 × VDDS to 0.9 × VDDS			12	µA
I_{OZ}	I_{OZ}(I_{PAD} Current) for BIDI cell. This current is contributed by the tristated driver leakage + input current of the Rx + weak pullup/pulldown leakage. PAD is swept from 0 to VDDS and the Max($I_{(PAD)}$) is measured and is reported as I_{OZ}			12	µA
C_{IN}	Input capacitance			10	pF
V_{OL3}	Output low-level threshold open-drain at 3-mA sink current			0.2 × VDDS	V
I_{OLmin}	Low-level output current @V_{OL}=0.2 × VDDS	3			mA
t_{OF}	Output fall time from V_{IHmin} to V_{ILmax} with a bus capacitance CB from 10 pF to 400 pF	20+0.1 × Cb		250	ns
I²C Standard Mode – 3.3 V					
V_{IH}	Input high-level threshold	0.7 × VDDS			V
V_{IL}	Input low-level threshold			0.3 × VDDS	V
V_{hys}	Hysteresis	0.05 × VDDS			V
I_{IN}	Input current at each I/O pin with an input voltage between 0.1 × VDDS to 0.9 × VDDS	31		80	µA
I_{OZ}	I_{OZ}(I_{PAD} Current) for BIDI cell. This current is contributed by the tristated driver leakage + input current of the Rx + weak pullup/pulldown leakage. PAD is swept from 0 to VDDS and the Max($I_{(PAD)}$) is measured and is reported as I_{OZ}	31		80	µA
C_{IN}	Input capacitance			10	pF
V_{OL3}	Output low-level threshold open-drain at 3-mA sink current			0.4	V
I_{OLmin}	Low-level output current @V_{OL}=0.4V	3			mA
I_{OLmin}	Low-level output current @V_{OL}=0.6V for full drive load (400pF/400KHz)	6			mA

Table 5-7. Dual Voltage LVCMOS I²C DC Electrical Characteristics (continued)

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
t_{OF}	Output fall time from V_{IHmin} to V_{ILmax} with a bus capacitance CB from 5 pF to 400 pF			250	ns
I²C Fast Mode – 3.3 V					
V_{IH}	Input high-level threshold	0.7 × VDDS			V
V_{IL}	Input low-level threshold			0.3 × VDDS	V
V_{hys}	Hysteresis	0.05 × VDDS			V
I_{IN}	Input current at each I/O pin with an input voltage between 0.1 × VDDS to 0.9 × VDDSS	31		80	µA
I_{OZ}	I_{OZ}(I_{PAD} Current) for BIDI cell. This current is contributed by the tristated driver leakage + input current of the Rx + weak pullup/pulldown leakage. PAD is swept from 0 to VDDS and the Max($I_{(PAD)}$) is measured and is reported as I_{OZ}	31		80	µA
C_{IN}	Input capacitance			10	pF
V_{OL3}	Output low-level threshold open-drain at 3-mA sink current			0.4	V
I_{OLmin}	Low-level output current @V_{OL}=0.4V	3			mA
I_{OLmin}	Low-level output current @V_{OL}=0.6V for full drive load (400pF/400KHz)	6			mA
t_{OF}	Output fall time from V_{IHmin} to V_{ILmax} with a bus capacitance CB from 10 pF to 200 pF (Proper External Resistor Value should be used as per I2C spec)	20+0.1 × Cb		250	ns
	Output fall time from V_{IHmin} to V_{ILmax} with a bus capacitance CB from 300 pF to 400 pF (Proper External Resistor Value should be used as per I2C spec)	40		290	

(1) VDDS stands for corresponding power supply (that is, vddshv3). For more information on the power supply name and the corresponding ball, see Table 4-1, POWER [11] column.

(2) For more information on the I/O cell configurations, see the Control Module section of the Device TRM.

Table 5-8. IQ1833 Buffers DC Electrical Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
Signal Names in MUXMODE 0: tclk					
Balls: K21					
1.8-V Mode					
V_{IH}	Input high-level threshold (Does not meet JEDEC V_{IH})	0.75 × VDDS			V
V_{IL}	Input low-level threshold (Does not meet JEDEC V_{IL})			0.25 × VDDS	V
V_{HYS}	Input hysteresis voltage	100			mV
I_{IN}	Input current at each I/O pin	2		11	µA
C_{PAD}	Pad capacitance (including package capacitance)			1	pF
3.3-V Mode					
V_{IH}	Input high-level threshold (Does not meet JEDEC V_{IH})	2.0			V
V_{IL}	Input low-level threshold (Does not meet JEDEC V_{IL})			0.6	V
V_{HYS}	Input hysteresis voltage	400			mV
I_{IN}	Input current at each I/O pin	5		11	µA
C_{PAD}	Pad capacitance (including package capacitance)			1	pF

(1) VDDS stands for corresponding power supply (that is, vddshv3). For more information on the power supply name and the corresponding ball, see Table 4-1, POWER [11] column.

Table 5-9. IHHV1833 Buffers DC Electrical Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
Signal Names in MUXMODE 0: porz / wakeup3 / wakeup0					
Balls: AB10/ AC10/ F19					
1.8-V Mode					
V_{IH}	Input high-level threshold	1.2			V
V_{IL}	Input low-level threshold			0.4	V
V_{HYS}	Input hysteresis voltage	40			mV
I_{IN}	Input current at each I/O pin	0.02		1	µA
C_{PAD}	Pad capacitance (including package capacitance)			1	pF
3.3-V Mode					
V_{IH}	Input high-level threshold	1.2			V
V_{IL}	Input low-level threshold			0.4	V
V_{HYS}	Input hysteresis voltage	40			mV
I_{IN}	Input current at each I/O pin	5		8	µA
C_{PAD}	Pad capacitance (including package capacitance)			1	pF

Table 5-10. LVCMOS CSI2 DC Electrical Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
Signals MUXMODE 0 : csi2_0_dx[2:0]; csi2_0_dy[2:0]					
Bottom Balls: AC1 / AB2 / AD1 / AC2 / AE2 / AD2					
MIPI D-PHY Mode Low-Power Receiver (LP-RX)					
V_{IH}	Input high-level voltage	880		1350	mV
V_{IL}	Input low-level voltage			550	mV
V_{ITH}	Input high-level threshold(1)			880	mV
V_{ITL}	Input low-level threshold(2)	550			mV
V_{HYS}	Input hysteresis(3)	25			mV
MIPI D-PHY Mode Ultralow Power Receiver (ULP-RX)					
V_{IL}	Input low-level voltage			300	mV
V_{ITL}	Input low-level threshold(4)	300			mV
V_{HYS}	Input hysteresis(3)	25			mV
MIPI D-PHY Mode High-Speed Receiver (HS-RX)					
V_{IDTH}	Differential input high-level threshold	70			mV
V_{IDTL}	Differential input low-level threshold			–70	mV
V_{IDMAX}	Maximum differential input voltage(7)			270	mV
V_{IHHS}	Single-ended input high voltage(5)			460	mV
V_{ILHS}	Single-ended input low voltage(5)	–40			mV
V_{CMRXDC}	Differential input common-mode voltage(5)(6)	70		330	mV
Z_{ID}	Differential input impedance	80	100	125	Ω

(1) V_{ITH} is the voltage at which the receiver is required to detect a high state in the input signal.

(2) V_{ITL} is the voltage at which the receiver is required to detect a low state in the input signal. V_{ITL} is larger than the maximum single-ended line high voltage during HS transmission. Therefore, both low-power (LP) receivers will detect low during HS signaling.

(3) To reduce noise sensitivity on the received signal, the LP receiver is required to incorporate a hysteresis, V_{HYST}. V_{HYST} is the difference between the V_{ITH} threshold and the V_{ITL} threshold.

(4) V_{ITL} is the voltage at which the receiver is required to detect a low state in the input signal. Specification is relaxed for detecting 0 during

ultralow power (ULP) state. The LP receiver is not required to detect HS single-ended voltage as 0 in this state.

(5) Excluding possible additional RF interference of 200 mV_{PP} beyond 450 MHz.

(6) This value includes a ground difference of 50 mV between the transmitter and the receiver, the static common-mode level tolerance and variations below 450 MHz.

(7) This number corresponds to the VOD_{MAX} transmitter.

(8) Common mode is defined as the average voltage level of X and Y: $V_{CMRX} = (V_X + V_Y) / 2$.

(9) Common mode ripple may be due to t_R or t_F and transmission line impairments in the PCB.

(10) For more information regarding the pin name (or ball name) and corresponding signal name, see Table 4-5 *CSI 2 Signal Descriptions*.

Table 5-11. BMLB18 Buffers DC Electrical Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
Signal Names in MUXMODE 0: mlbp_dat_n / mlbp_dat_p / mlbp_sig_n / mlbp_sig_p / mlbp_clk_n / mlbp_clk_p					
Balls: T1 / T2 / U4 / T3 / U1 / U2					
1.8-V Mode					
V_{IH}/V_{IL}	Input high-level threshold	VCM ± 50mV			V
V_{HYS}	Input hysteresis voltage		NONE		mV
V_{OD}	Differential output voltage (measured with 50ohm resistor between PAD and PADN)	300		500	mV
V_{CM}	Common mode output voltage	1		1.5	V
C_{PAD}	Pad capacitance (including package capacitance)			4	pF

Table 5-12. Dual Voltage SDIO1833 DC Electrical Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
Signal Names in Mode 0: mmc1_clk, mmc1_cmd, mmc1_data[3:0]					
Bottom Balls: U3 / V4 / V3 / V2 / W1 / V1					
1.8-V Mode					
V_{IH}	Input high-level threshold	1.27			V
V_{IL}	Input low-level threshold			0.58	V
V_{HYS}	Input hysteresis voltage	50 (2)			mV
I_{IN}	Input current at each I/O pin			30	µA
I_{OZ}	I_{OZ}(I_{PAD} Current) for BIDI cell. This current is contributed by the tristated driver leakage + input current of the Rx + weak pullup/pulldown leakage. PAD is swept from 0 to VDDS and the Max(I(PAD)) is measured and is reported as I_{OZ}			30	µA
$I_{IN\ with\ pulldown\ enabled}$	Input current at each I/O pin with weak pulldown enabled measured when PAD = VDDS	50	120	210	µA
$I_{IN\ with\ pullup\ enabled}$	Input current at each I/O pin with weak pullup enabled measured when PAD = 0	60	120	200	µA
C_{PAD}	Pad capacitance (including package capacitance)			5	pF
V_{OH}	Output high-level threshold (I_{OH} = 2 mA)	1.4			V
V_{OL}	Output low-level threshold (I_{OL} = 2 mA)			0.45	V
3.3-V Mode					
V_{IH}	Input high-level threshold	0.625 × VDDS			V
V_{IL}	Input low-level threshold			0.25 × VDDS	V
V_{HYS}	Input hysteresis voltage	40 (2)			mV
I_{IN}	Input current at each I/O pin			110	µA

Table 5-12. Dual Voltage SDIO1833 DC Electrical Characteristics (continued)

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
I_{OZ}	I_{OZ}(I_{PAD} Current) for BIDI cell. This current is contributed by the tristated driver leakage + input current of the Rx + weak pullup/pulldown leakage. PAD is swept from 0 to VDDS and the Max(I(PAD)) is measured and is reported as I_{OZ}			110	µA
I_{IN} with pulldown enabled	Input current at each I/O pin with weak pulldown enabled measured when PAD = VDDS	40	100	290	µA
I_{IN} with pullup enabled	Input current at each I/O pin with weak pullup enabled measured when PAD = 0	10	100	290	µA
C_{PAD}	Pad capacitance (including package capacitance)			5	pF
V_{OH}	Output high-level threshold (I_{OH} = 2 mA)	0.75 × VDDS			V
V_{OL}	Output low-level threshold (I_{OL} = 2 mA)			0.125 × VDDS	V

(1) VDDS stands for corresponding power supply (that is, vddshv8). For more information on the power supply name and the corresponding ball, see Table 4-1, POWER [11] column.

(2) Hysteresis is enabled/disabled with CTRL_CORE_CONTROL_HYST_1.SDCARD_HYST register.

Table 5-13. Dual Voltage LVCMOS DC Electrical Characteristics

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
1.8-V Mode					
V_{IH}	Input high-level threshold	0.65 × VDDS			V
V_{IL}	Input low-level threshold			0.35 × VDDS	V
V_{HYS}	Input hysteresis voltage	100			mV
V_{OH}	Output high-level threshold (I_{OH} = 2 mA)	VDDS-0.45			V
V_{OL}	Output low-level threshold (I_{OL} = 2 mA)			0.45	V
I_{DRIVE}	Pin Drive strength at PAD Voltage = 0.45V or VDDS-0.45V	6			mA
I_{IN}	Input current at each I/O pin			16	µA
I_{OZ}	I_{OZ}(I_{PAD} Current) for BIDI cell. This current is contributed by the tristated driver leakage + input current of the Rx + weak pullup/pulldown leakage. PAD is swept from 0 to VDDS and the Max($I_{(PAD)}$) is measured and is reported as I_{OZ}			16	µA
I_{IN} with pulldown enabled	Input current at each I/O pin with weak pulldown enabled measured when PAD = VDDS	50	120	210	µA
I_{IN} with pullup enabled	Input current at each I/O pin with weak pullup enabled measured when PAD = 0	60	120	200	µA
C_{PAD}	Pad capacitance (including package capacitance)			4	pF
Z_O	Output impedance (drive strength)		40		Ω
3.3-V Mode					
V_{IH}	Input high-level threshold	2			V
V_{IL}	Input low-level threshold			0.8	V
V_{HYS}	Input hysteresis voltage	200			mV
V_{OH}	Output high-level threshold (I_{OH} = 100 µA)	VDDS-0.2			V
V_{OL}	Output low-level threshold (I_{OL} = 100 µA)			0.2	V
I_{DRIVE}	Pin Drive strength at PAD Voltage = 0.45V or VDDS-0.45V	6			mA
I_{IN}	Input current at each I/O pin			65	µA

Table 5-13. Dual Voltage LVCMOS DC Electrical Characteristics (continued)

over operating free-air temperature range (unless otherwise noted)

PARAMETER		MIN	NOM	MAX	UNIT
I_{OZ}	I_{OZ}(I_{PAD} Current) for BIDI cell. This current is contributed by the tristated driver leakage + input current of the Rx + weak pullup/pulldown leakage. PAD is swept from 0 to VDDS and the Max($I_{(PAD)}$) is measured and is reported as I_{OZ}			65	µA
$I_{IN\ with\ pulldown\ enabled}$	Input current at each I/O pin with weak pulldown enabled measured when PAD = VDDS	40	100	200	µA
$I_{IN\ with\ pullup\ enabled}$	Input current at each I/O pin with weak pullup enabled measured when PAD = 0	10	100	290	µA
C_{PAD}	Pad capacitance (including package capacitance)			4	pF
Z_O	Output impedance (drive strength)		40		Ω

(1) VDDS stands for corresponding power supply. For more information on the power supply name and the corresponding ball, see Table 4-1, POWER [11] column.

5.7.1 USBPHY DC Electrical Characteristics

NOTE

USB1 instance is compliant with the USB3.0 SuperSpeed Transmitter and Receiver Normative Electrical Parameters as defined in the USB3.0 Specification Rev 1.0 dated Jun 6, 2011.

NOTE

USB1 and USB2 Electrical Characteristics are compliant with USB2.0 Specification Rev 2.0 dated April 27, 2000 including ECNs and Errata as applicable.

5.7.2 HDMIPHY DC Electrical Characteristics

NOTE

The HDMIPHY DC Electrical Characteristics are compliant with the HDMI 1.4a specification and are not reproduced here.

5.7.3 PCIEPHY DC Electrical Characteristics

NOTE

The PCIe interfaces are compliant with the electrical parameters specified in PCI Express® Base Specification Revision 3.0.

5.8 VPP Specifications for One-Time Programmable (OTP) eFuses

This section specifies the operating conditions required for programming the OTP eFuses and is applicable only for High-Security Devices.

Table 5-14. Recommended Operating Conditions for OTP eFuse Programming

over operating free-air temperature range (unless otherwise noted)

PARAMETER	DESCRIPTION	MIN	NOM	MAX	UNIT
vdd	Supply voltage range for the core domain during OTP operation	1.11	1.15	1.2	V
vpp	Supply voltage range for the eFuse ROM domain during normal operation		NC		V
	Supply voltage range for the eFuse ROM domain during OTP programming(1)(2)		1.8		V
I(vpp)				100	mA
Tj	Temperature (junction)	0	25	85	ºC

(1) Supply voltage range includes DC errors and peak-to-peak noise. TI power management solutions TLV70018-Q1 from the TLV700xx family meet the supply voltage range needed for vpp.

(2) During normal operation, no voltage should be applied to vpp. This can be typically achieved by disabling the regulator attached to the vpp terminal. For more details, see TLV700xx-Q1 300-mA, Low-I_Q, Low-Dropout Regulator.

5.8.1 Hardware Requirements

The following hardware requirements must be met when programming keys in the OTP eFuses:

- The vpp power supply must be disabled when not programming OTP registers.
- The vpp power supply must be ramped up after the proper device power-up sequence (for more details, see Section 5.10.3).

5.8.2 Programming Sequence

Programming sequence for OTP eFuses:

1. Power on the board per the power-up sequencing. No voltage should be applied on the vpp terminal during power up and normal operation.
2. Load the OTP write software required to program the eFuse (contact your local TI representative for the OTP software package).
3. Apply the voltage on the vpp terminal according to the specification in Table 5-14.
4. Run the software that programs the OTP registers.
5. After validating the content of the OTP registers, remove the voltage from the vpp terminal.

5.8.3 Impact to Your Hardware Warranty

You accept that e-Fusing the TI Devices with security keys permanently alters them. You acknowledge that the e-Fuse can fail, for example, due to incorrect or aborted program sequence or if you omit a sequence step. Further the TI Device may fail to secure boot if the error code correction check fails for the Production Keys or if the image is not signed and optionally encrypted with the current active Production Keys. These types of situations will render the TI Device inoperable and TI will be unable to confirm whether the TI Devices conformed to their specifications prior to the attempted e-Fuse. CONSEQUENTLY, TI WILL HAVE NO LIABILITY (WARRANTY OR OTHERWISE) FOR ANY TI DEVICES THAT HAVE BEEN e-FUSED WITH SECURITY KEYS.

5.9 Thermal Resistance Characteristics for CBD Package

For reliability and operability concerns, the maximum junction temperature of the Device has to be at or below the T_J value identified in Section 5.4, *Recommended Operating Conditions*.

A BCI compact thermal model for this Device is available and recommended for use when modeling thermal performance in a system.

Therefore, it is recommended to perform thermal simulations at the system level with the worst case device power consumption.

5.9.1 Package Thermal Characteristics

Table 5-15 provides the thermal resistance characteristics for the package used on this device.

NOTE

Power dissipation of 3.0 W and an ambient temperature of 85ºC is assumed for CBD package.

Table 5-15. Thermal Resistance Characteristics

NO.	PARAMETER	DESCRIPTION	°C/W(1)	AIR FLOW (m/s)(2)
T1	$R\Theta_{JC}$	Junction-to-case	0.23	N/A
T2	$R\Theta_{JB}$	Junction-to-board	3.65	N/A
T3	$R\Theta_{JA}$	Junction-to-free air	12.8	0
T4		Junction-to-moving air	10.4	0.5
T5			9.6	1
T6			8.8	2
T7			8.3	3
T8	Ψ_{JT}	Junction-to-package top	0.1	0
T9			0.1	0.5
T10			0.1	1
T11			0.1	2
T12			0.1	3
T13	Ψ_{JB}	Junction-to-board	3.7	0
T14			3.7	0.5
T15			3.6	1
T16			3.6	2
T17			3.5	3

(1) These measurements were conducted in a JEDEC defined 2S2P system (with the exception of the Theta JC [$R\Theta_{JC}$] measurement, which was conducted in a JEDEC defined 1S0P system) and will change based on environment as well as application. For more information, see these EIA/JEDEC standards:

- JESD51-2, *Integrated Circuits Thermal Test Method Environment Conditions - Natural Convection (Still Air)*
- JESD51-3, *Low Effective Thermal Conductivity Test Board for Leaded Surface Mount Packages*
- JESD51-7, *High Effective Thermal Conductivity Test Board for Leaded Surface Mount Packages*
- JESD51-9, *Test Boards for Area Array Surface Mount Packages*

(2) m/s = meters per second

5.10 Timing Requirements and Switching Characteristics

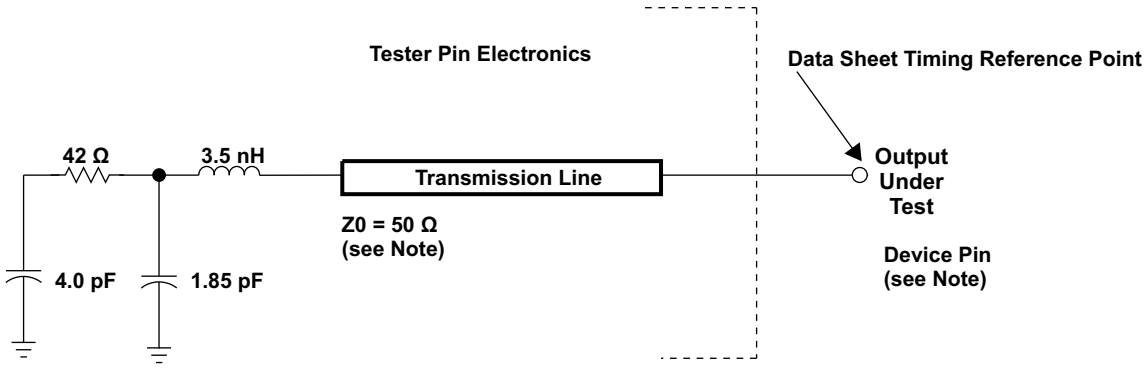
5.10.1 Timing Parameters and Information

The timing parameter symbols used in the timing requirement and switching characteristic tables are created in accordance with JEDEC Standard 100. To shorten the symbols, some of pin names and other related terminologies have been abbreviated as follows:

Table 5-16. Timing Parameters

SUBSCRIPTS	
SYMBOL	**PARAMETER**
c	Cycle time (period)
d	Delay time
dis	Disable time
en	Enable time
h	Hold time
su	Setup time
START	Start bit
t	Transition time
v	Valid time
w	Pulse duration (width)
X	Unknown, changing, or don't care level
F	Fall time
H	High
L	Low
R	Rise time
V	Valid
IV	Invalid
AE	Active Edge
FE	First Edge
LE	Last Edge
Z	High impedance

5.10.1.1 Parameter Information



NOTE: The data sheet provides timing at the device pin. For output timing analysis, the tester pin electronics and its transmission line effects must be taken into account. A transmission line with a delay of 2 ns can be used to produce the desired transmission line effect. The transmission line is intended as a load only. It is not necessary to add or subtract the transmission line delay (2 ns) from the data sheet timings.

Input requirements in this data sheet are tested with an input slew rate of < 4 Volts per nanosecond (4 V/ns) at the device pin.

Figure 5-2. Test Load Circuit for AC Timing Measurements

The load capacitance value stated is only for characterization and measurement of AC timing signals.

This load capacitance value does not indicate the maximum load the device is capable of driving.

5.10.1.1.1 1.8 V and 3.3 V Signal Transition Levels

All input and output timing parameters are referenced to V_{ref} for both "0" and "1" logic levels. V_{ref} = (VDD I/O)/2.

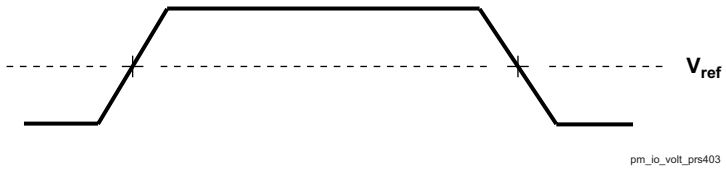


Figure 5-3. Input and Output Voltage Reference Levels for AC Timing Measurements

All rise and fall transition timing parameters are referenced to V_{IL} MAX and V_{IH} MIN for input clocks, V_{OL} MAX and V_{OH} MIN for output clocks.

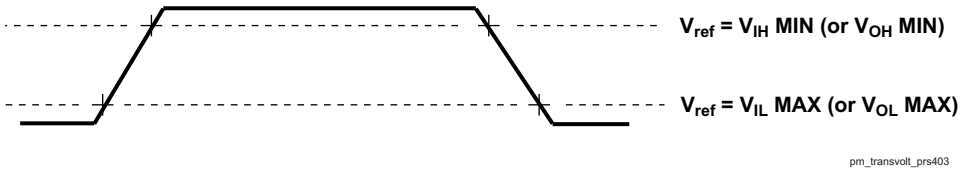


Figure 5-4. Rise and Fall Transition Time Voltage Reference Levels

5.10.1.1.2 1.8 V and 3.3 V Signal Transition Rates

The default SLEWCONTROL settings in each pad configuration register must be used to ensure timings, unless specific instructions otherwise are given in the individual timing subsections of the datasheet.

All timings are tested with an input edge rate of 4 volts per nanosecond (4 V/ns).

5.10.1.1.3 Timing Parameters and Board Routing Analysis

The timing parameter values specified in this data manual do not include delays by board routes. As a good board design practice, such delays must *always* be taken into account. Timing values may be adjusted by increasing/decreasing such delays. TI recommends using the available I/O buffer information specification (IBIS) models to analyze the timing characteristics correctly. To properly use IBIS models to attain accurate timing analysis for a given system, see the Using IBIS Models for timing Analysis application report. If needed, external logic hardware such as buffers may be used to compensate any timing differences.

5.10.2 Interface Clock Specifications

5.10.2.1 Interface Clock Terminology

The interface clock is used at the system level to sequence the data and/or to control transfers accordingly with the interface protocol.

5.10.2.2 Interface Clock Frequency

The two interface clock characteristics are:

- The maximum clock frequency
- The maximum operating frequency

The interface clock frequency documented in this document is the maximum clock frequency, which corresponds to the maximum frequency programmable on this output clock. This frequency defines the maximum limit supported by the Device IC and does not take into account any system consideration (PCB, peripherals).

The system designer will have to consider these system considerations and the Device IC timing characteristics as well to define properly the maximum operating frequency that corresponds to the maximum frequency supported to transfer the data on this interface.

5.10.3 Power Supply Sequences

This section describes the power-up and power-down sequence required to ensure proper device operation. The power supply names described in this section comprise a superset of a family of compatible devices. Some members of this family will not include a subset of these power supplies and their associated device modules. Refer to the Section 4.2, *Pin Attributes* of the Section 4, *Terminal Configuration and Functions* to determine which power supplies are applicable.

Figure 5-5 through Figure 5-9 and associated notes described the device Recommended Power Sequencing.

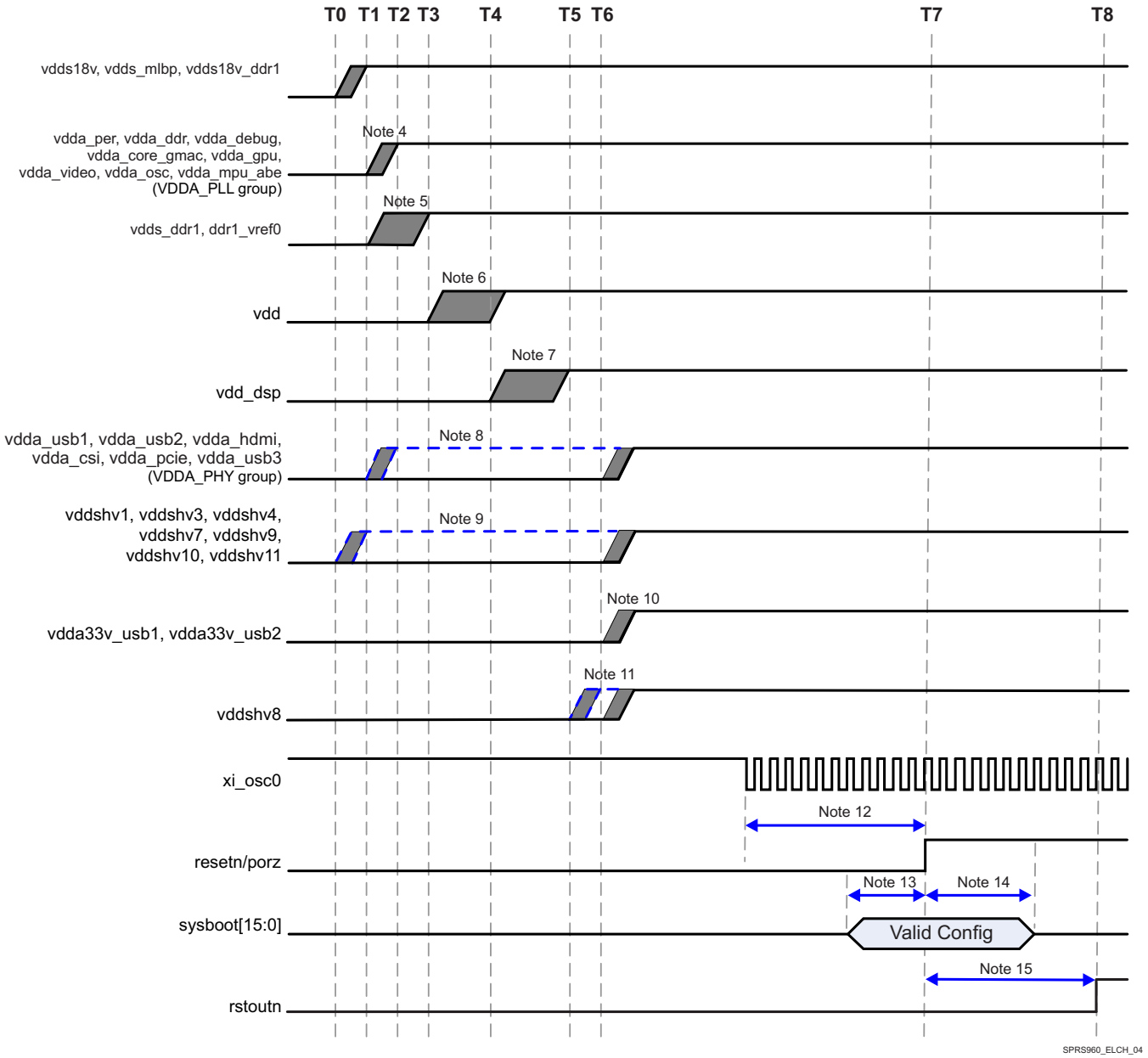


Figure 5-5. Recommended Power-Up Sequencing

(1) T0 = 0ms, T1 = 0.55ms, T2 = 1.1ms, T3 = 1.65ms, T4 = 2.2ms, T5 = 2.75ms, T6 = 3.3ms, T7 = 6.9ms, T8 ≈ 9ms. All “Tn” markers show total elapsed time from T0.

(2) Terminology:

- $V_{OPR\ MIN}$ = Minimum Operational Voltage level that ensures device functionality and specified performance per Section 5.4, *Recommended Operating Conditions*.
- Ramp Up = transition time from V_{OFF} to $V_{OPR\ MIN}$

(3) General timing diagram items:

- Grey shaded areas show valid transition times for supplies between $V_{OPR\ MIN}$ and V_{OFF}.
- Dashed horizontal lines are not valid ramp times but show alternate transition times based upon common sources and clarified in associated note.
- Dashed vertical lines show approximate elapse times based upon TI recommended PMIC power sequencer circuit performance.

(4) vdda_* rails should not be combined with vdds18v_* for best performance to avoid transient switching noise impacts on analog domains. vdda_* should not ramp-up before vdds18v_* but could ramp concurrently if design ensures final operational voltage will not be reached

until after vdds18v. The preferred sequence has vdda_* following vdds18v_* to ensure circuit components and PCB design do not cause an inadvertent violation.

(5) vdds_ddr1 should not ramp-up before vdds18v_*. The preferred sequence has vdds_ddr1 following vdds18v_* to ensure circuit components and PCB design do not cause an inadvertent violation. vdds_ddr1 can ramp-up before, concurrently or after vdda_*, there are no dependencies between vdds_ddr1 and vdda_* domains.
- For DDR2 mode of operation (1.8V), vdds_ddr1 supplies can be combined with all vdds18v_* supplies and ramped up together for simplified PDN and power sequencing.
- If vdds_ddr1 is combined with vdds18v_ddr1 but kept separate from vdds18v on board, then this combined 1.8V DDR supply can come up together or after the vdds18v supply. The 1.8V DDR supply should never ramp up before the vdds18v.

(6) vdd should not ramp-up before vdds18v_* or vdds_ddr1 domains have reached $V_{OPR\ MIN}$.

(7) vdd_dsp could ramp concurrently with vdd if design ensures:
- Final vdd_dsp operational voltage will not be reached until after vdd.
- vdd_dsp maintains a voltage level at least 150mV less than vdd during entire ramp time. The preferred sequence has vdd_dsp following vdd to ensure circuit components and PCB design do not cause an inadvertent violation.

(8) VDDA_PHY group:
- should ramp up concurrently or after vdda33v_usb[1-2] to avoid unintended current path between vdda_pcie to vdda33v_usb1 during power sequencing.
- could ramp up concurrently with VDDA_PLL group only if the vdda33v_usb1 power resource has an “off impedance” greater than 100Ω.

(9) vddshv[1, 3-4, 7, 9-11] domains:
- If 1.8V I/O signaling is needed, then 1.8V must be sourced from common vdds18v supply and ramp up concurrently with vdds18v.
- If any 3.3V I/O signaling is needed, then the desired 3.3V vddshv[1, 3-4, 7, 9-11] rails must ramp up after vdd_dsp.

(10) vdda33v_usb[1-2] domain should:
- ramp up before or concurrently with VDDA_PHY group if USB signaling is needed and to avoid unintended current path between vdda_pcie to vdda33v_usb[1-2] during power sequencing.
- connect to 3.3V vddshv[1, 3-4, 7, 9-11] common supply if USB signaling is not needed since USB analog power ball also supplies digital IO buffers that must be powered during operation.

(11) vddshv8 shows two ramp up options for 1.8V I/O or 3.3V I/O or SD Card operation:
- If 1.8V I/O signaling is needed, then vddshv8 must ramp up after vdd and before or concurrently with 3.3V vddshv* rails.
- If 3.3V I/O signaling is needed, then vddshv8 must be combined with other 3.3V vddshv* rails.
- If SD Card operation is needed, then vddshv8 must be sourced from a dual voltage (3.3/1.8V) power source per SDIO specifications and ramp up concurrently with 3.3V vddshv* rails.

(12) porz must remain asserted low until both of the following conditions are met:
- Minimum of 12 *P, where P = 1 / (SYS_CLK1/610), units in ns.
- All device supply rails reach stable operational levels.

(13) Setup time: sysboot[15:0] pins must be valid 2P(12) before porz is de-asserted high.

(14) Hold time: sysboot[15:0] pins must be valid 15P(12) after porz is de-asserted high.

(15) rstoutn will be set high after global reset, due to porz, is de-asserted following an internal 2ms delay. rstoutn is only valid after vddshv3 reaches an operational level. If used as a peripheral component reset, it should be AND gated with porz to avoid possible reset glitches during power up.

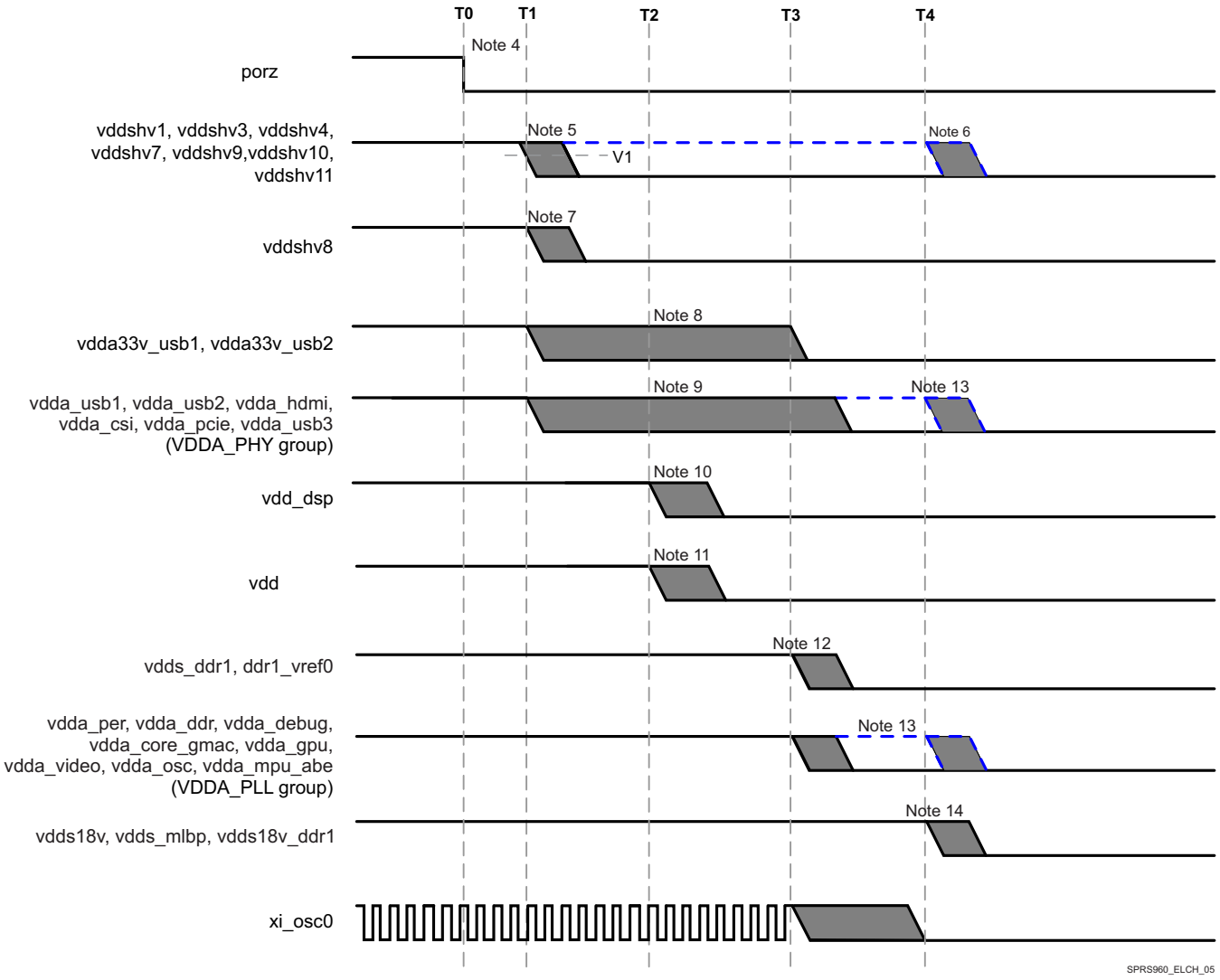


Figure 5-6. Recommended Power-Down Sequencing

(1) T1 ≥ 100 µs; T2 = 500 µs; T3 = 1.0 ms; T4 = 1.5ms; V1 = 2.7 V. All "Tn" markers are intended to show total elapsed time, not interval times.

(2) Terminology:
- $V_{OPR\ MIN}$ = Minimum Operational Voltage level that ensures device functionality and specified performance in Section 5.4, *Recommended Operating Conditions*.
- V_{OFF} = OFF Voltage level is defined to be less than 0.6 V where any current draw has no impact to POH.
- Ramp Down = transition time from $V_{OPR\ MIN}$ to V_{OFF} and is slew rate independent.

(3) General timing diagram items:
- Grey shaded areas show valid transition times for supplies between $V_{OPR\ MIN}$ and V_{OFF}.
- Blue dashed lines are not valid windows but show alternate ramp possibilities based on the associated note.
- Dashed vertical lines show approximate elapse times based upon TI recommended PMIC power-down sequencer circuit performance.

(4) PORz must be asserted low for 100 µs min to ensure SoC is set to a safe functional state before any voltage begins to ramp down.

(5) vddshv[1, 3-4, 7, 9-11] domains supplied by 3.3 V:
- must remain greater than 2.7 V to enable Dual Voltage GPIO selector circuit operation for 100 µs min after PORz is asserted low.
- must be in first group of supplies ramping down after PORz has been asserted low for 100 µs min.
- must not exceed vdds18v by more than 2 V during ramp down, see Figure 5-7, "vdds18v versus vddshv[1, 3-4, 7, 9-11] Discharge Relationship".

(6) vddshv[1, 3-4, 7, 9-11] domains supplied by 1.8 V must ramp down concurrently with vdds18v and be sourced from common vdds18v supply.

(7) vddshv8 supporting SD Card:

- must be sourced from independent power resource that can provide dual voltage (3.3 / 1.8 V) operation as required to be compliant to SDIO specification
- must be in first group of supplies to ramp down after PORz has been asserted low for 100 µs min.
- if SDIO operation is not needed, must be grouped and ramped down with other vddshv[1, 3-4, 7, 9-11] domains as noted above.

(8) vdda33v_usb[1-2] domains:
- can start ramping down 100 µs after low assertion of PORz
- can ramp down concurrently or before VDDA_PHY group

(9) VDDA_PHY domain group must ramp down concurrently or after vdda33v_usb[1-2].

(10) vdd_dsp domain can ramp down before or concurrently with vdd.

(11) vdd must ramp down after or concurrently with vdd_dsp.

(12) vdds_ddr1 domain:
- should ramp down after vdd begins ramping down.
- If DDR2 memory is used (requiring 1.8V supply),
  - then vdds_ddr1 can be combined with vdds18v and vdds18v_ddr1 domains and sourced from a common supply. Accordingly, all domains can ramp down concurrently with vdds18v.
  - if vdds_ddr1 and vdds18v_ddr1 are combined but kept separate from vdds18v, then the combined 1.8V DDR supply can ramp down before or concurrently with vdds18v.

(13) vdda_* domains:
- can ramp down before, concurrently or after vdds_ddr1, there is no dependency between these supplies.
- can ramp down before or concurrently with vdds18v.
- must satisfy the vdds18v versus vdda_* discharge relationship (see Figure 5-9) if any of the vdda_* disable point is later or discharge rate is slower than vdds18v.

(14) vdds18v domain:
- should maintain $V_{OPR\ MIN}$ (V_{NOM} -5% = 1.71 V) until all other supplies start to ramp down.
- must satisfy the vdds18v versus vddshv[1, 3-4, 7, 9-11] discharge relationship (see Figure 5-7) if any of the vddshv[1, 3-4, 7, 9-11] is operating at 3.3 V.
- must satisfy the vdds18v versus vdds_ddr1 discharge relationship (see Figure 5-8) if vdds_ddr1 discharge rate is slower than vdds18v.

Figure 5-7 describes vddshv[1, 3-4, 7, 9-11] Supplies Falling Before vdds18v Supplies Delta.

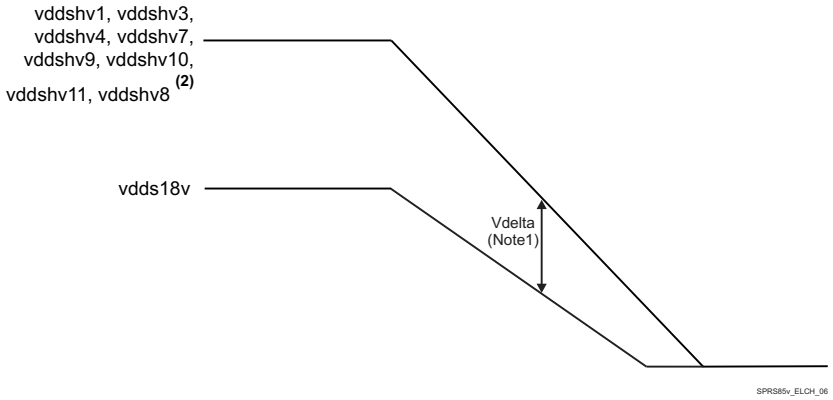


Figure 5-7. vdds18v versus vddshv[1, 3-4, 7, 9-11] Discharge Relationship

(1) $V_{delta\ MAX}$ = 2V.

(2) If vddshv8 is powered by the same supply source as the other vddshv[1, 3-4, 7, 9-11] rails.

If vdds18v and vdds_ddr1 are disabled at the same time due to a loss of input power event or if vdds_ddr1 discharges more slowly than vdds18v, analysis has shown no reliability impacts when the elapsed time period beginning with vdds18v dropping below 1.0 V and ending with vdds_ddr1 dropping below 0.6 V is less than 10 ms (Figure 5-8).

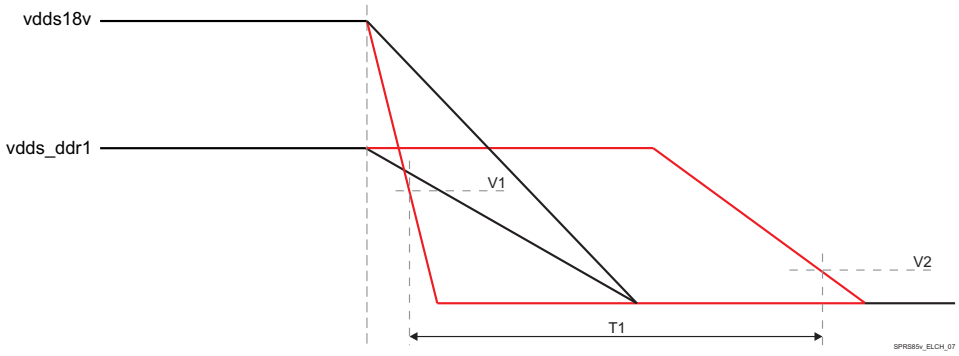


Figure 5-8. vdds18v and vdds_ddr1 Discharge Relationship(1)

(1) V1 > 1.0 V; V2 < 0.6 V; T1 < 10ms.

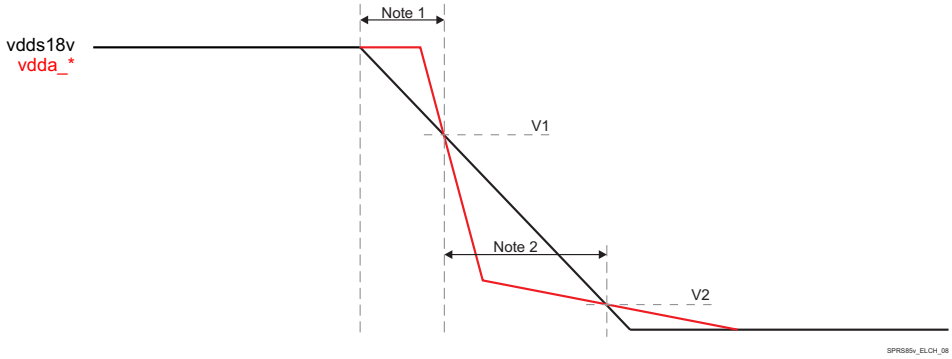


Figure 5-9. vdds18v and vdda_* Discharge Relationship(3)

(1) vdda_* can be ≥ vdds18v, until vdds18v drops below 1.62 V.

(2) vdds18v must be ≥ vdda_*, until vdds18v reaches 0.6 V.

(3) V1 = 1.62 V; V2 < 0.6 V.

Figure 5-7 through Figure 5-10 and associated notes described the device Abrupt Power Down Sequence.

A ”loss of input power event” occurs when the system’s input power is unexpectedly removed. Normally, the recommended power-down sequence should be followed and can be accomplished within 1.5-2 ms of elapsed time. This is the typical range of elapsed time available following a loss of power event, see Section 7.3.7 for design recommendations. If sufficient elapse time is not provided, then an “abrupt” power-down sequence can be supported without impacting POH reliability if all of the following conditions are met (Figure 5-10).

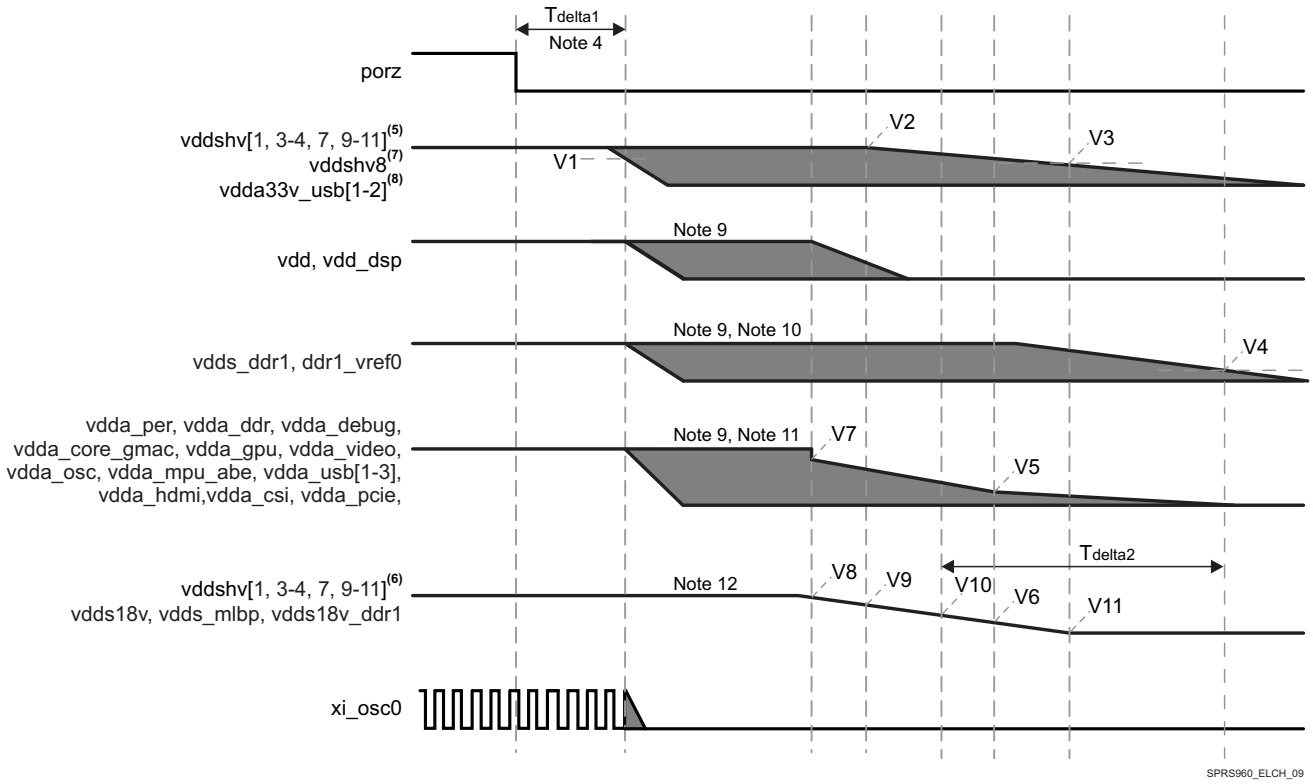


Figure 5-10. Abrupt Power-Down Sequencing(1)

(1) V1 = 2.7 V; V2 = 3.3 V; V3 = 2.0 V; V4 = V5 = V6 = 0.6 V; V7 = V8 = 1.62 V; V9 = 1.3 V; V10 = 1.0 V; V11 = 0.0 V; T_{delta1} > 100 µs; T_{delta2} < 10 ms.

(2) Terminology:
- $V_{OPR\ MIN}$ = Minimum Operational Voltage level that ensures device functionality and specified performance in Section 5.4, *Recommended Operating Conditions*.
- V_{OFF} = OFF Voltage level is defined to be less than 0.6 V, where any current draw has no impact to POH.
- Ramp Down = transition time from $V_{OPR\ MIN}$ to V_{OFF} and is slew rate independent.

(3) General timing diagram items:
- Grey shaded areas show valid transition times for supplies between $V_{OPR\ MIN}$ and V_{OFF}.
- Dashed vertical lines show approximate elapse times based upon TI recommended PMIC power-down sequencer circuit performance.

(4) PORz must be asserted low for 100 µs min to ensure SoC is set to a safe functional state before any voltage begins to ramp down.

(5) vddshv[1, 3-4, 7, 9-11] domains supplied by 3.3 V:
- must remain greater than 2.7 V to enable Dual Voltage GPIO selector circuit operation for 100µs min, after PORz is asserted low.
- must not exceed vdds18v voltage level by more than 2V during ramp down, until vdds18v drops below V_{OFF} (0.6 V).

(6) vddshv[1, 3-4, 7, 9-11] domains supplied by 1.8 V must ramp down concurrently with vdds18v and be sourced from common vdds18v supply.

(7) vddshv8 supporting SD Card:
- must be in first group of supplies to ramp down after PORz has been asserted low for 100 µs min.
- must be sourced from independent power resource that can provide dual voltage (3.3 / 1.8 V) operation as required to be compliant to SDIO specification.
- if SDIO operation is not needed, must be grouped with other vddshv[1, 3-4, 7, 9-11] domains.

(8) vdda33v_usb[1-2] domains must be in first group of supplies to ramp down after PORz has been asserted low for 100 µs min.

(9) vdd_dsp, vdd, vdds_ddr1, vdda_* domains can all start to ramp down in any order after 100 µs low assertion of PORz.

(10) vdds_ddr1 domain:

- can remain at $V_{OPR\ MIN}$ or a level greater than vdds18v during ramp down.
- elapsed time from vdds18v dropping below 1.0 V to vdds_ddr1 dropping below 0.6 V must not exceed 10 ms.

(11) vdda_* domains:

- can start to ramp down before or concurrently with vdds18v.
- must not exceed vdds18v voltage level after vdds18v drops below 1.62 V until vdds18v drops below V_{OFF} (0.6 V).

(12) vdds18v domain should maintain a minimum level of 1.62 V (V_{NOM} – 10%) until vdd_dsp and vdd start to ramp down.

5.10.4 Clock Specifications

NOTE

For more information, see Power Reset and Clock Management / PRCM Environment / External Clock Signal and Power Reset / PRCM Functional Description / PRCM Clock Manager Functional Description section of the Device TRM.

NOTE

Audio Back End (ABE) module is not supported for this family of devices, but “ABE” name is still present in some clock or DPLL names.

The device operation requires the following clocks:

- The system clocks, SYS_CLK1 (Mandatory) and SYS_CLK2 (Optional) are the main clock sources of the device. They supply the reference clock to the DPLLs as well as functional clock to several modules.

The Device also embeds an internal free-running 32-kHz oscillator that is always active as long as the the wake-up (WKUP) domain is supplied.

Figure 5-11 shows the external input clock sources and the output clocks to peripherals.

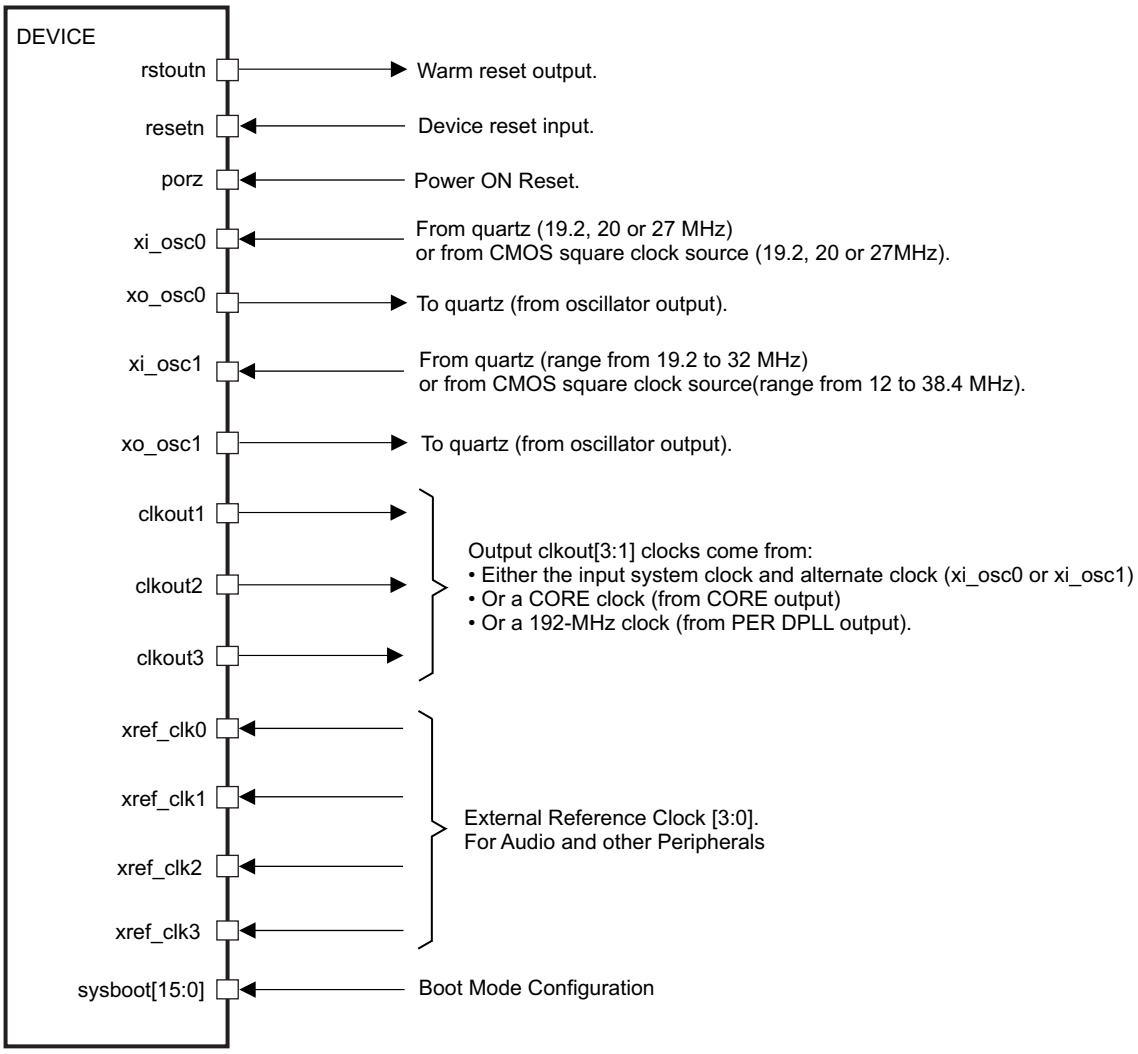


Figure 5-11. Clock Interface

5.10.4.1 Input Clocks / Oscillators

- The source of the internal system clock (SYS_CLK1) could be either:
  - A CMOS clock that enters on the xi_osc0 ball (with xo_osc0 left unconnected on the CMOS clock case).
  - A crystal oscillator clock managed by xi_osc0 and xo_osc0.
- The source of the internal system clock (SYS_CLK2) could be either:
  - A CMOS clock that enters on the xi_osc1 ball (with xo_osc1 left unconnected on the CMOS clock case).
  - A crystal oscillator clock managed by xi_osc1 and xo_osc1.

SYS_CLK1 is received directly from oscillator OSC0. For more information about SYS_CLK1 see Device TRM, Chapter: *Power, Reset, and Clock Management.*

5.10.4.1.1 OSC0 External Crystal

An external crystal is connected to the device pins. Figure 5-12 describes the crystal implementation.

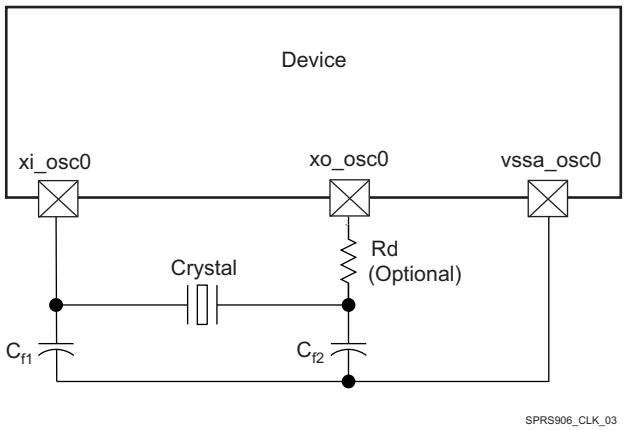


Figure 5-12. OSC0 Crystal Implementation

NOTE

The load capacitors, C_{f1} and C_{f2} in Figure 5-12, should be chosen such that the below equation is satisfied. C_L in the equation is the load specified by the crystal manufacturer. All discrete components used to implement the oscillator circuit should be placed as close as possible to the associated oscillator xi_osc0, xo_osc0, and vssa_osc0 pins.

$$C_L = \frac{C_{f1} C_{f2}}{(C_{f1} + C_{f2})}$$

Figure 5-13. Load Capacitance Equation

The crystal must be in the fundamental mode of operation and parallel resonant. Table 5-17 summarizes the required electrical constraints and Table 5-21

Table 5-17. OSC0 Crystal Electrical Characteristics

NAME	DESCRIPTION			MIN	TYP	MAX	UNIT
f_p	Parallel resonance crystal frequency				19.2, 20, 27		MHz
C_{f1}	C_{f1} load capacitance for crystal parallel resonance with $C_{f1} = C_{f2}$			12		24	pF
C_{f2}	C_{f2} load capacitance for crystal parallel resonance with $C_{f1} = C_{f2}$			12		24	pF
ESR(C_{f1},C_{f2}) (1)	Crystal ESR					100	Ω
C_O	Crystal shunt capacitance	ESR = 30 Ω ESR = 40 Ω	19.2 MHz, 20 MHz, 27 MHz			7	pF
		ESR = 50 Ω	19.2 MHz, 20 MHz			7	pF
			27 MHz			5	pF
		ESR = 60 Ω	19.2 MHz, 20 MHz			7	pF
			27 MHz		Not Supported		-
		ESR = 80 Ω	19.2 MHz, 20 MHz			5	pF
			27 MHz		Not Supported		-
		ESR = 100 Ω	19.2 MHz, 20 MHz			3	pF
			27 MHz		Not Supported		-
L_M	Crystal motional inductance for f_p = 20 MHz				10.16		mH
C_M	Crystal motional capacitance				3.42		fF

Table 5-17. OSC0 Crystal Electrical Characteristics (continued)

NAME	DESCRIPTION		MIN	TYP	MAX	UNIT
$t_{j(xiosc0)}$	Frequency accuracy(1), xi_osc0	Ethernet and MLB not used			±200	ppm
		Ethernet RGMII and RMII using derived clock			±50	
		Ethernet MII using derived clock			±100	
		MLB using derived clock			±50	

(1) Crystal characteristics should account for tolerance+stability+aging.

When selecting a crystal, the system design must consider the temperature and aging characteristics of a based on the worst case environment and expected life expectancy of the system.

Table 5-18 details the switching characteristics of the oscillator and the requirements of the input clock.

Table 5-18. Oscillator Switching Characteristics—Crystal Mode

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT
f_p	Oscillation frequency		19.2, 20, 27 MHz		MHz
t_{sx}	Start-up time			4	ms

5.10.4.1.2 OSC0 Input Clock

A 1.8-V LVCMOS-Compatible Clock Input can be used instead of the internal oscillator to provide the SYS_CLK1 clock input to the system. The external connections to support this are shown in Figure 5-14. The xi_osc0 pin is connected to the 1.8-V LVCMOS-Compatible clock source. The xi_osc0 pin is left unconnected. The vssa_osc0 pin is connected to board ground (VSS).

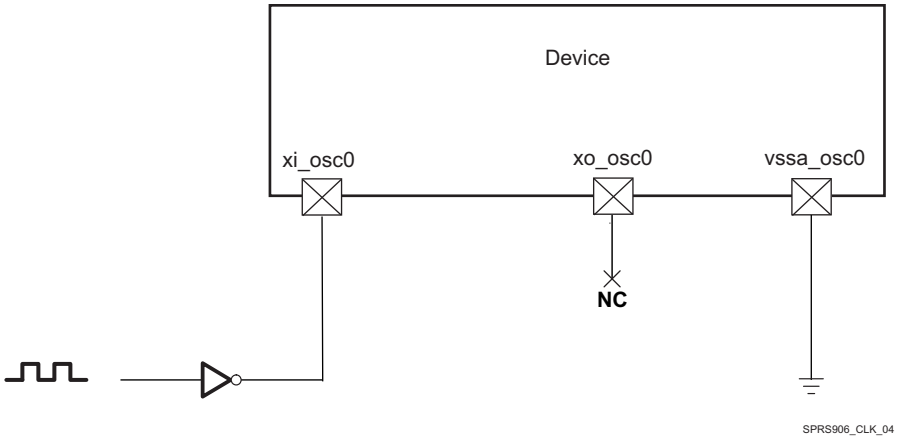


Figure 5-14. 1.8-V LVCMOS-Compatible Clock Input

Table 5-19 summarizes the OSC0 input clock electrical characteristics.

Table 5-19. OSC0 Input Clock Electrical Characteristics—Bypass Mode

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT
f	Frequency		19.2, 20, 27		MHz
C_{IN}	Input capacitance	2.184	2.384	2.584	pF
I_{IN}	Input current (3.3V mode)	4	6	10	µA

Table 5-20 details the OSC0 input clock timing requirements.

Table 5-20. OSC0 Input Clock Timing Requirements

NAME	DESCRIPTION		MIN	TYP	MAX	UNIT
CK0	$1 / t_{c(xiosc0)}$	Frequency, xi_osc0		19.2, 20, 27		MHz
CK1	$t_{w(xiosc0)}$	Pulse duration, xi_osc0 low or high	0.45 × $t_{c(xiosc0)}$		0.55 × $t_{c(xiosc0)}$	ns
	$t_{j(xiosc0)}$	Period jitter(1), xi_osc0			0.01 × $t_{c(xiosc0)}$	ns
	$t_{R(xiosc0)}$	Rise time, xi_osc0			5	ns
	$t_{F(xiosc0)}$	Fall time, xi_osc0			5	ns
	$t_{j(xiosc0)}$	Frequency accuracy(2), xi_osc0 — Ethernet and MLB not used			±200	ppm
		Frequency accuracy(2), xi_osc0 — Ethernet RGMII and RMII using derived clock			±50	ppm
		Frequency accuracy(2), xi_osc0 — Ethernet MII using derived clock			±100	ppm
		Frequency accuracy(2), xi_osc0 — MLB using derived clock			±50	ppm

(1) Period jitter is meant here as follows:

– The maximum value is the difference between the longest measured clock period and the expected clock period

– The minimum value is the difference between the shortest measured clock period and the expected clock period

(2) Crystal characteristics should account for tolerance+stability+aging.

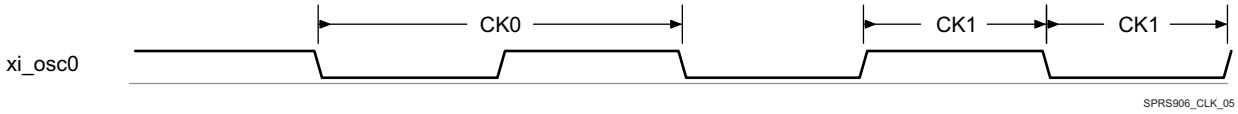


Figure 5-15. xi_osc0 Input Clock

5.10.4.1.3 Auxiliary Oscillator OSC1 Input Clock

SYS_CLK2 is received directly from oscillator OSC1. For more information about SYS_CLK2 see Device TRM, Chapter: *Power, Reset, and Clock Management.*

5.10.4.1.3.1 OSC1 External Crystal

An external crystal is connected to the device pins. Figure 5-16 describes the crystal implementation.

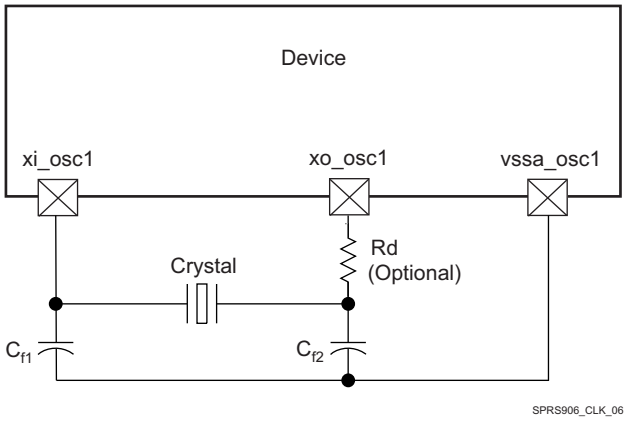


Figure 5-16. Crystal Implementation

NOTE

The load capacitors, C_{f1} and C_{f2} in Figure 5-16, should be chosen such that the below equation is satisfied. C_L in the equation is the load specified by the crystal manufacturer. All discrete components used to implement the oscillator circuit should be placed as close as possible to the associated oscillator xi_osc1, xo_osc1, and vssa_osc1 pins.

$$C_L = \frac{C_{f1} C_{f2}}{(C_{f1} + C_{f2})}$$

Figure 5-17. Load Capacitance Equation

The crystal must be in the fundamental mode of operation and parallel resonant. Table 5-21 summarizes the required electrical constraints.

Table 5-21. OSC1 Crystal Electrical Characteristics

NAME	DESCRIPTION			MIN	TYP	MAX	UNIT
f_p	Parallel resonance crystal frequency			Range from 19.2 to 32			MHz
C_{f1}	C_{f1} load capacitance for crystal parallel resonance with $C_{f1} = C_{f2}$			12		24	pF
C_{f2}	C_{f2} load capacitance for crystal parallel resonance with $C_{f1} = C_{f2}$			12		24	pF
ESR(C_{f1},C_{f2})	Crystal ESR					100	Ω
C_O	Crystal shunt capacitance	ESR = 30 Ω	19.2 MHz ≤ f_p ≤ 32 MHz			7	pF
		ESR = 40 Ω	19.2 MHz ≤ f_p ≤ 32 MHz			5	pF
		ESR = 50 Ω	19.2 MHz ≤ f_p ≤ 25 MHz			7	pF
			25 MHz < f_p ≤ 27 MHz			5	pF
			27 MHz < f_p ≤ 32 MHz		Not Supported		-
		ESR = 60 Ω	19.2 MHz ≤ f_p ≤ 23 MHz			7	pF
			23 MHz < f_p ≤ 25 MHz			5	pF
			25 MHz < f_p ≤ 32 MHz		Not Supported		-
		ESR = 80 Ω	19.2 MHz ≤ f_p ≤ 23 MHz			5	pF
			23 MHz ≤ f_p ≤ 25 MHz			3	pF
			25 MHz < f_p ≤ 32 MHz		Not Supported		-
		ESR = 100 Ω	19.2 MHz ≤ f_p ≤ 20 MHz			3	pF
			20 MHz < f_p ≤ 32 MHz		Not Supported		-
L_M	Crystal motional inductance for f_p = 20 MHz				10.16		mH
C_M	Crystal motional capacitance				3.42		fF
$t_{j(xiosc1)}$	Frequency accuracy(1), xi_osc1		Ethernet and MLB not used			±200	ppm
			Ethernet RGMII and RMII using derived clock			±50	
			Ethernet MII using derived clock			±100	
			MLB using derived clock			±50	

(1) Crystal characteristics should account for tolerance+stability+aging.

When selecting a crystal, the system design must consider the temperature and aging characteristics of a based on the worst case environment and expected life expectancy of the system.

Table 5-22 details the switching characteristics of the oscillator and the requirements of the input clock.

Table 5-22. Oscillator Switching Characteristics—Crystal Mode

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT
f_p	Oscillation frequency	Range from 19.2 to 32			MHz

Table 5-22. Oscillator Switching Characteristics—Crystal Mode (continued)

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT
t_{sX}	Start-up time			4	ms

5.10.4.1.3.2 OSC1 Input Clock

A 1.8-V LVCMOS-Compatible Clock Input can be used instead of the internal oscillator to provide the SYS_CLK2 clock input to the system. The external connections to support this are shown in, Figure 5-18. The xi_osc1 pin is connected to the 1.8-V LVCMOS-Compatible clock sources. The xo_osc1 pin is left unconnected. The vssa_osc1 pin is connected to board ground (vss).

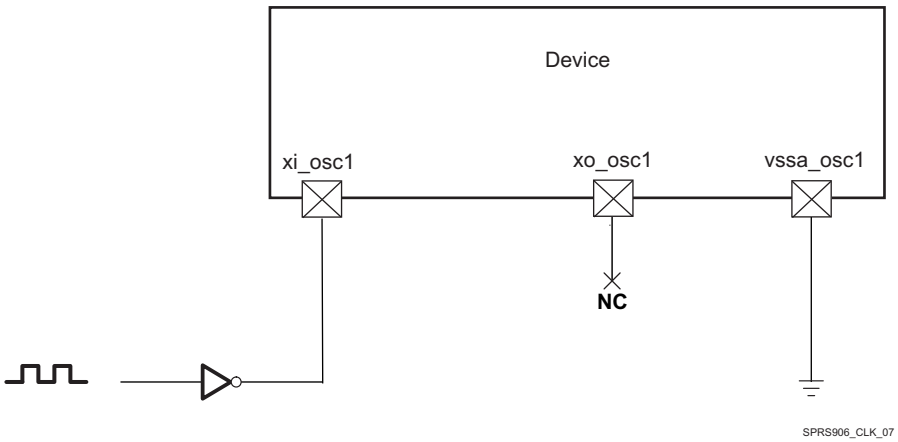


Figure 5-18. 1.8-V LVCMOS-Compatible Clock Input

Table 5-23 summarizes the OSC1 input clock electrical characteristics.

Table 5-23. OSC1 Input Clock Electrical Characteristics—Bypass Mode

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT
f	Frequency	Range from 12 to 38.4			MHz
C_{IN}	Input capacitance	2.819	3.019	3.219	pF
I_{IN}	Input current (3.3V mode)	4	6	10	µA
t_{sX}	Start-up time(1)	See(2)			ms

(1) To switch from bypass mode to crystal or from crystal mode to bypass mode, there is a waiting time about 100 µs; however, if the chip comes from bypass mode to crystal mode the crystal will start-up after time mentioned in Table 5-22, t_{sX} parameter.

(2) Before the processor boots up and the oscillator is set to bypass mode, there is a waiting time when the internal oscillator is in application mode and receives a wave. The switching time in this case is about 100 µs.

Table 5-24 details the OSC1 input clock timing requirements.

Table 5-24. OSC1 Input Clock Timing Requirements

NAME	DESCRIPTION			MIN	TYP	MAX	UNIT
CK0	$1 / t_{c(xiosc1)}$	Frequency, xi_osc1		Range from 12 to 38.4			MHz
CK1	$t_{w(xiosc1)}$	Pulse duration, xi_osc1 low or high		$0.45 \times t_{c(xiosc1)}$		$0.55 \times t_{c(xiosc1)}$	ns
	$t_{j(xiosc1)}$	Period jitter(1), xi_osc1				$0.01 \times t_{c(xiosc1)}$ (3)	ns
	$t_{R(xiosc1)}$	Rise time, xi_osc1				5	ns
	$t_{F(xiosc1)}$	Fall time, xi_osc1				5	ns

Table 5-24. OSC1 Input Clock Timing Requirements (continued)

NAME	DESCRIPTION			MIN	TYP	MAX	UNIT
	$t_{j(xiosc1)}$	Frequency accuracy(2), xi_osc1	Ethernet and MLB not used			±200	ppm
			Ethernet RGMII and RMII using derived clock			±50	
			Ethernet MII using derived clock			±100	
			MLB using derived clock			±50	

(1) Period jitter is meant here as follows:
- The maximum value is the difference between the longest measured clock period and the expected clock period
- The minimum value is the difference between the shortest measured clock period and the expected clock period

(2) Crystal characteristics should account for tolerance+stability+aging.

(3) The Period jitter requirement for osc1 can be relaxed to 0.02 × tc(xiosc1) under the following constraints:
a.The osc1/SYS_CLK2 clock bypasses all device PLLs
b.The osc1/SYS_CLK2 clock is only used to source the DSS pixel clock outputs

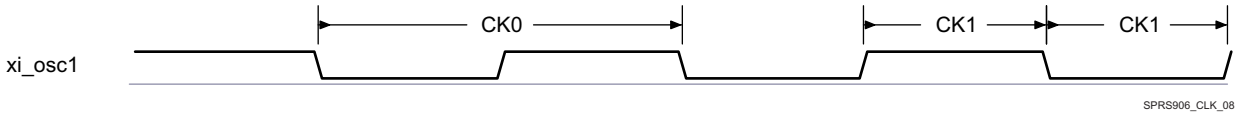


Figure 5-19. xi_osc1 Input Clock

5.10.4.1.4 RC On-die Oscillator Clock

NOTE

The OSC_32K_CLK clock, provided by the On-die 32K RC oscillator, inside of the SoC, is not accurate 32kHz clock.

The frequency may significantly vary with temperature and silicon characteristics.

For more information about OSC_32K_CLK see the Device TRM, Chapter: *Power, Reset, and Clock Management*.

5.10.4.2 Output Clocks

The device provides three output clocks. Summary of these output clocks are as follows:

- clkout1 - Device Clock output 1. Can be used as a system clock for other devices. The source of the clkout1 could be either:
  - The input system clock and alternate clock (xi_osc0 or xi_osc1)
  - CORE clock (from CORE output)
  - 192-MHz clock (from PER DPLL output)
- clkout2 - Device Clock output 2. Can be used as a system clock for other devices. The source of the clkout2 could be either:
  - The input system clock and alternate clock (xi_osc0 or xi_osc1)
  - CORE clock (from CORE output)
  - 192-MHz clock (from PER DPLL output)
- clkout3 - Device Clock output 3. Can be used as a system clock for other devices. The source of the clkout3 could be either:
  - The input system clock and alternate clock (xi_osc0 or xi_osc1)
  - CORE clock (from CORE output)
  - 192-MHz clock (from PER DPLL output)

For more information about Output Clocks see Device TRM, Chapter: *Power, Reset, and Clock Management*.

5.10.4.3 DPLLs, DLLs

NOTE

For more information, see:

- Power, Reset, and Clock Management / Clock Management Functional / Internal Clock Sources / Generators / Generic DPLL Overview Section

  and

- Display Subsystem / Display Subsystem Overview section of the Device TRM.

To generate high-frequency clocks, the device supports multiple on-chip DPLLs controlled directly by the PRCM module. They are of two types: type A and type B DPLLs.

- They have their own independent power domain (each one embeds its own switch and can be controlled as an independent functional power domain)
- They are fed with ALWAYS ON system clock, with independent control per DPLL.

The different DPLLs managed by the PRCM are listed below:

- DPLL_MPU: It supplies the MPU subsystem clocking internally.
- DPLL_IVA: It feeds the IVA subsystem clocking.
- DPLL_CORE: It supplies all interface clocks and also few module functional clocks.
- DPLL_PER: It supplies several clock sources: a 192-MHz clock for the display functional clock, a 96-MHz functional clock to subsystems and peripherals.
- DPLL_ABE: It provides clocks to various modules within the device.
- DPLL_USB: It provides 960M clock for USB modules (USB1/2/3/4).
- DPLL_GMAC: It supplies several clocks for the Gigabit Ethernet Switch (GMAC_SW).
- DPLL_DSP: It feeds the DSP Subsystem clocking.
- DPLL_GPU: It supplies clock for the GPU Subsystem.
- DPLL_DDR: It generates clocks for the two External Memory Interface (EMIF) controllers and their associated EMIF PHYs.
- DPLL_PCIE_REF: It provides reference clock for the APLL_PCIE in PCIE Subsystem.
- APLL_PCIE: It feeds clocks for the device Peripheral Component Interconnect Express (PCIe) controllers.

NOTE

The following DPLLs are controlled by the clock manager located in the always-on Core power domain (CM_CORE_AON):

- DPLL_MPU, DPLL_IVA, DPLL_CORE, DPLL_ABE, DPLL_DDR, DPLL_GMAC, DPLL_PCIE_REF, DPLL_PER, DPLL_USB, DPLL_DSP, DPLL_GPU, APLL_PCIE_REF.

For more information on CM_CORE_AON and CM_CORE or PRCM DPLLs, see the Power, Reset, and Clock Management (PRCM) chapter of the Device TRM.

The following DPLLs are not managed by the PRCM:

- DPLL_VIDEO1; (It is controlled from DSS)
- DPLL_HDMI; (It is controlled from DSS)
- DPLL_DEBUG; (It is controlled from DEBUGSS)
- DPLL_USB_OTG_SS; (It is controlled from OCP2SCP1)

NOTE

For more information for not controlled from PRCM DPLL's see the related chapters in TRM.

5.10.4.3.1 DPLL Characteristics

The DPLL has three relevant input clocks. One of them is the reference clock (CLKINP) used to generated the synthesized clock but can also be used as the bypass clock whenever the DPLL enters a bypass mode. It is therefore mandatory. The second one is a fast bypass clock (CLKINPULOW) used when selected as the bypass clock and is optional. The third clock (CLKINPHIF) is explained in the next paragraph.

The DPLL has three output clocks (namely CLKOUT, CLKOUTX2, and CLKOUTHIF). CLKOUT and CLKOUTX2 run at the bypass frequency whenever the DPLL enters a bypass mode. Both of them are generated from the lock frequency divided by a post-divider (namely M2 post-divider). The third clock, CLKOUTHIF, has no automatic bypass capability. It is an output of a post-divider (M3 post-divider) with the input clock selectable between the internal lock clock (Fdpll) and CLKINPHIF input of the PLL through an asynchronous multplexing.

For more information, see the Power Reset Controller Management chapter of the Device TRM.

Table 5-25 summarizes DPLL type described in Section 5.10.4.3, *DPLLs, DLLs Specifications* introduction.

Table 5-25. DPLL Control Type

DPLL NAME	TYPE	CONTROLLED BY PRCM
DPLL_ABE	Table 5-26 (Type A)	Yes(1)
DPLL_CORE	Table 5-26 (Type A)	Yes(1)
DPLL_DEBUGSS	Table 5-26 (Type A)	No(2)
DPLL_DSP	Table 5-26 (Type A)	Yes(1)
DPLL_GMAC	Table 5-26 (Type A)	Yes(1)
DPLL_HDMI	Table 5-27 (Type B)	No(2)
DPLL_IVA	Table 5-26 (Type A)	Yes(1)
DPLL_MPU	Table 5-26 (Type A)	Yes(1)
DPLL_PER	Table 5-26 (Type A)	Yes(1)
APLL_PCIE	Table 5-26 (Type A)	Yes(1)
DPLL_PCIE_REF	Table 5-27 (Type B)	Yes(1)
DPLL_USB	Table 5-27 (Type B)	Yes(1)
DPLL_USB_OTG_SS	Table 5-27 (Type B)	No(2)
DPLL_VIDEO1	Table 5-26 (Type A)	No(2)
DPLL_DDR	Table 5-26 (Type A)	Yes(1)
DPLL_GPU	Table 5-26 (Type A)	Yes(1)

(1) DPLL is in the always-on domain.

(2) DPLL is not controlled by the PRCM.

Table 5-26 and Table 5-27 summarize the DPLL characteristics and assume testing over recommended operating conditions.

Table 5-26. DPLL Type A Characteristics

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT	COMMENTS
f_{input}	CLKINP input frequency	0.032		52	MHz	F_{INP}
$f_{internal}$	Internal reference frequency	0.15		52	MHz	REFCLK
$f_{CLKINPHIF}$	CLKINPHIF input frequency	10		1400	MHz	F_{INPHIF}
$f_{CLKINPULOW}$	CLKINPULOW input frequency	0.001		600	MHz	Bypass mode: f_{CLKOUT} = $f_{CLKINPULOW}$ / (M1 + 1) if ulowclken = 1(6)
f_{CLKOUT}	CLKOUT output frequency	20(1)		1800(2)	MHz	[M / (N + 1)] × F_{INP} × [1 / M2] (in locked condition)

Table 5-26. DPLL Type A Characteristics (continued)

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT	COMMENTS
$f_{CLKOUTx2}$	CLKOUTx2 output frequency	40(1)		2200(2)	MHz	2 × [M / (N + 1)] × F_{INP} × [1 / M2] (in locked condition)
$f_{CLKOUTHIF}$	CLKOUTHIF output frequency	20(3)		1400(4)	MHz	F_{INPHIF} / M3 if clkinphifsel = 1
		40(3)		2200(4)	MHz	2 × [M / (N + 1)] × F_{INP} × [1 / M3] if clkinphifsel = 0
$f_{CLKDCOLDO}$	DCOCLKLDO output frequency	40		2800	MHz	2 × [M / (N + 1)] × F_{INP} (in locked condition)
t_{lock}	Frequency lock time			6 + 350 × REFCLK	µs	
p_{lock}	Phase lock time			6 + 500 × REFCLK	µs	
$t_{relock-L}$	Relock time—Frequency lock(5) (LP relock time from bypass)			6 + 70 × REFCLK	µs	DPLL in LP relock time: lowcurrstdby = 1
$p_{relock-L}$	Relock time—Phase lock(5) (LP relock time from bypass)			6 + 120 × REFCLK	µs	DPLL in LP relock time: lowcurrstdby = 1
$t_{relock-F}$	Relock time—Frequency lock(5) (fast relock time from bypass)			3.55 + 70 × REFCLK	µs	DPLL in fast relock time: lowcurrstdby = 0
$p_{relock-F}$	Relock time—Phase lock(5) (fast relock time from bypass)			3.55 + 120 × REFCLK	µs	DPLL in fast relock time: lowcurrstdby = 0

(1) The minimum frequencies on CLKOUT and CLKOUTX2 are assuming M2 = 1.

For M2 > 1, the minimum frequency on these clocks will further scale down by factor of M2.

(2) The maximum frequencies on CLKOUT and CLKOUTX2 are assuming M2 = 1.

(3) The minimum frequency on CLKOUTHIF is assuming M3 = 1. For M3 > 1, the minimum frequency on this clock will further scale down by factor of M3.

(4) The maximum frequency on CLKOUTHIF is assuming M3 = 1.

(5) Relock time assumes typical operating conditions, 10°C maximum temperature drift.

(6) Bypass mode: $f_{CLKOUT} = F_{INP}$ if ulowclken = 0. For more information, see the Device TRM.

Table 5-27. DPLL Type B Characteristics

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT	COMMENTS
f_{input}	CLKINP input clock frequency	0.62		60	MHz	F_{INP}
$f_{internal}$	REFCLK internal reference clock frequency	0.62		2.5	MHz	[1 / (N + 1)] × F_{INP}
$f_{CLKINPULOW}$	CLKINPULOW bypass input clock frequency	0.001		600	MHz	Bypass mode: f_{CLKOUT} = $f_{CLKINPULOW}$ / (M1 + 1) If ulowclken = 1(4)
$f_{CLKLDOOUT}$	CLKOUTLDO output clock frequency	20(1)(5)		2500(2)(5)	MHz	M / (N + 1)] × F_{INP} × [1 / M2] (in locked condition)
f_{CLKOUT}	CLKOUT output clock frequency	20(1)(5)		1450(2)(5)	MHz	[M / (N + 1)] × F_{INP} × [1 / M2] (in locked condition)
$f_{CLKDCOLDO}$	Internal oscillator (DCO) output clock frequency	750(5)		1500(5)	MHz	[M / (N + 1)] × F_{INP} (in locked condition)
		1250(5)		2500(5)	MHz	
t_J	CLKOUTLDO period jitter	–2.5%		2.5%		The period jitter at the output clocks is ± 2.5% peak to peak
	CLKOUT period jitter					
	CLKDCOLDO period jitter					
t_{lock}	Frequency lock time			350 × REFCLKs	µs	
p_{lock}	Phase lock time			500 × REFCLKs	µs	
$t_{relock-L}$	Relock time—Frequency lock(3) (LP relock time from bypass)			9 + 30 × REFCLKs	µs	

Table 5-27. DPLL Type B Characteristics (continued)

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT	COMMENTS
$p_{relock-L}$	Relock time—Phase lock(3) (LP relock time from bypass)			9 + 125 × REFCLKs	µs	

(1) The minimum frequency on CLKOUT is assuming M2 = 1.

For M2 > 1, the minimum frequency on this clock will further scale down by factor of M2.

(2) The maximum frequency on CLKOUT is assuming M2 = 1.

(3) Relock time assumes typical operating conditions, 10°C maximum temperature drift.

(4) Bypass mode: $f_{CLKOUT} = F_{INP}$ if ULOWCLKEN = 0. For more information, see the Device TRM.

(5) For output clocks, there are two frequency ranges according to the SELFREQDCO setting. For more information, see the Device TRM.

5.10.4.3.2 DLL Characteristics

Table 5-28 summarizes the DLL characteristics and assumes testing over recommended operating conditions.

Table 5-28. DLL Characteristics

NAME	DESCRIPTION	MIN	TYP	MAX	UNIT
f_{input}	Input clock frequency (EMIF_DLL_FCLK)			333	MHz
t_{lock}	Lock time			50k	cycles
t_{relock}	Relock time (a change of the DLL frequency implies that DLL must relock)			50k	cycles

5.10.4.3.3 DPLL and DLL Noise Isolation

NOTE

For more information on DPLL and DLL decoupling capacitor requirements, see the External Capacitors / Voltage Decoupling Capacitors / I/O and Analog Voltage Decoupling / VDDA Power Domain section.

5.10.5 Recommended Clock and Control Signal Transition Behavior

All clocks and control signals ***must*** transition between V_{IH} and V_{IL} (or between V_{IL} and V_{IH}) in a monotonic manner. Monotonic transitions are more easily ensured with faster switching signals. Slower input transitions are more susceptible to glitches due to noise and special care should be taken for slow input clocks.

5.10.6 Peripherals

5.10.6.1 Timing Test Conditions

All timing requirements and switching characteristics are valid over the recommended operating conditions unless otherwise specified.

5.10.6.2 Virtual and Manual I/O Timing Modes

Some of the timings described in the following sections require the use of Virtual or Manual I/O Timing Modes. Table 5-29 provides a summary of the Virtual and Manual I/O Timing Modes across all device interfaces. The individual interface timing sections found later in this document provide the full description of each applicable Virtual and Manual I/O Timing Mode. Refer to the "Pad Configuration" section of the TRM for the procedure on implementing the Virtual and Manual Timing Modes in a system.

Table 5-29. Modes Summary

Virtual or Manual IO Mode Name	Data Manual Timing Mode
DPI Video Output	

Table 5-29. Modes Summary (continued)

Virtual or Manual IO Mode Name	Data Manual Timing Mode
No Virtual or Manual IO Timing Mode Required	DPI3 Video Output Default Timings - Rising-edge Clock Reference
DSS_VIRTUAL1	DPI3 Video Output Default Timings - Falling-edge Clock Reference
VOUT2_IOSET1_MANUAL1	DPI2 Video Output IOSET1 Alternate Timings
VOUT2_IOSET1_MANUAL2	DPI2 Video Output IOSET1 Default Timings - Rising-edge Clock Reference
VOUT2_IOSET1_MANUAL3	DPI2 Video Output IOSET1 Default Timings - Falling-edge Clock Reference
VOUT2_IOSET1_MANUAL4	DPI2 Video Output IOSET1 MANUAL4 Timings
VOUT2_IOSET1_MANUAL5	DPI2 Video Output IOSET1 MANUAL5 Timings
VOUT3_MANUAL1	DPI3 Video Output Alternate Timings
VOUT3_MANUAL4	DPI3 Video Output MANUAL4 Timings
VOUT3_MANUAL5	DPI3 Video Output MANUAL5 Timings
GPMC	
No Virtual or Manual IO Timing Mode Required	GPMC Asynchronous Mode Timings and Synchronous Mode - Default Timings
GPMC_VIRTUAL1	GPMC Synchronous Mode - Alternate Timings
McASP	
No Virtual or Manual IO Timing Mode Required	McASP1 Asynchronous and Synchronous Transmit Timings
MCASP1_VIRTUAL1_SYNC_RX	See Table 5-77
MCASP1_VIRTUAL2_ASYNC_RX	See Table 5-77
No Virtual or Manual IO Timing Mode Required	McASP2 Asynchronous and Synchronous Transmit Timings
MCASP2_VIRTUAL1_SYNC_RX_80M	See Table 5-78
MCASP2_VIRTUAL2_ASYNC_RX	See Table 5-78
MCASP2_VIRTUAL3_SYNC_RX	See Table 5-78
MCASP2_VIRTUAL4_ASYNC_RX_80M	See Table 5-78
No Virtual or Manual IO Timing Mode Required	McASP3 Synchronous Transmit Timings
MCASP3_VIRTUAL2_SYNC_RX	See Table 5-79
No Virtual or Manual IO Timing Mode Required	McASP4 Synchronous Transmit Timings
MCASP4_VIRTUAL1_SYNC_RX	See Table 5-80
No Virtual or Manual IO Timing Mode Required	McASP5 Synchronous Transmit Timings
MCASP5_VIRTUAL1_SYNC_RX	See Table 5-81
No Virtual or Manual IO Timing Mode Required	McASP6 Synchronous Transmit Timings
MCASP6_VIRTUAL1_SYNC_RX	See Table 5-82
No Virtual or Manual IO Timing Mode Required	McASP7 Synchronous Transmit Timings
MCASP7_VIRTUAL2_SYNC_RX	See Table 5-83
No Virtual or Manual IO Timing Mode Required	McASP8 Synchronous Transmit Timings
MCASP8_VIRTUAL1_SYNC_RX	See Table 5-84
eMMC/SD/SDIO	
No Virtual or Manual IO Timing Mode Required	MMC1 DS (Pad Loopback), HS (Internal Loopback and Pad Loopback), SDR12 (Internal Loopback and Pad Loopback), and SDR25 Timings (Internal Loopback and Pad Loopback) Timings
MMC1_VIRTUAL1	MMC1 SDR50 (Pad Loopback) Timings
MMC1_VIRTUAL4	MMC1 DS (Internal Loopback) Timings
MMC1_VIRTUAL5	MMC1 SDR50 (Internal Loopback) Timings
MMC1_VIRTUAL6	MMC1 DDR50 (Internal Loopback) Timings
MMC1_MANUAL1	MMC1 DDR50 (Pad Loopback) Timings
MMC1_MANUAL2	MMC1 SDR104 Timings
No Virtual or Manual IO Timing Mode Required	MMC2 Standard (Pad Loopback), High Speed (Pad Loopback) Timings
MMC2_VIRTUAL2	MMC2 Standard (Internal Loopback), High Speed (Internal Loopback) Timings
MMC2_MANUAL1	MMC2 DDR (Pad Loopback) Timings
MMC2_MANUAL2	MMC2 DDR (Internal Loopback Manual) Timings

Table 5-29. Modes Summary (continued)

Virtual or Manual IO Mode Name	Data Manual Timing Mode
MMC2_MANUAL3	MMC2 HS200 Timings
No Virtual or Manual IO Timing Mode Required	MMC3 DS, SDR12, HS, SDR25 Timings
MMC3_MANUAL1	MMC3 SDR50 Timings
No Virtual or Manual IO Timing Mode Required	MMC4 DS, SDR12, HS, SDR25 Timings
QSPI	
No Virtual or Manual IO Timing Mode Required	QSPI Mode 3 Timings
QSPI1_MANUAL1	QSPI Mode 0 Timings
GMAC	
No Virtual or Manual IO Timing Mode Required	GMAC MII0/1 Timings
GMAC_RGMII0_MANUAL1	GMAC RGMII0 with Transmit Clock Internal Delay Enabled
GMAC_RGMII1_MANUAL1	GMAC RGMII1 with Transmit Clock Internal Delay Enabled
GMAC_RMII0_MANUAL1	GMAC RMII0 Timings
GMAC_RMII1_MANUAL1	GMAC RMII1 Timings
VIP	
VIP_MANUAL3	VIN2A (IOSET4/5/6) Rise-Edge Capture Mode Timings
VIP_MANUAL4	VIN2B (IOSET7/8/9) Rise-Edge Capture Mode Timings
VIP_MANUAL5	VIN2A (IOSET4/5/6) Fall-Edge Capture Mode Timings
VIP_MANUAL6	VIN2B (IOSET7/8/9) Fall-Edge Capture Mode Timings
VIP_MANUAL7	VIN1A (IOSET2) and VIN2B (IOSET1/10) Rise-Edge Capture Mode Timings
VIP_MANUAL9	VIN1B (IOSET6/7) Rise-Edge Capture Mode Timings
VIP_MANUAL10	VIN2B (IOSET2/11) Rise-Edge Capture Mode Timings
VIP_MANUAL11	VIN2B (IOSET2/11) Fall-Edge Capture Mode Timings
VIP_MANUAL12	VIN1A (IOSET2) and VIN2B (IOSET1/10) Fall-Edge Capture Mode Timings
VIP_MANUAL14	VIN1B (IOSET6/7) Fall-Edge Capture Mode Timings
VIP_MANUAL15	VIN1A (IOSET8/9/10) Rise-Edge Capture Mode Timings
VIP_MANUAL16	VIN1A (IOSET8/9/10) Fall-Edge Capture Mode Timings
PRU-ICSS	
No Virtual or Manual IO Timing Mode Required	All PRU_ICSS Modes not covered below
PR1_PRU1_DIR_IN_MANUAL	PRU-ICSS1 PRU1 Direct Input Mode Timings
PR1_PRU1_DIR_OUT_MANUAL	PRU-ICSS1 PRU1 Direct Output Mode Timings
PR1_PRU1_PAR_CAP_MANUAL	PRU-ICSS1 PRU1 Parallel Capture Mode Timings
PR2_PRU0_DIR_IN_MANUAL2	PRU-ICSS2 PRU0 IOSET2 Direct Input Mode Timings
PR2_PRU0_DIR_OUT_MANUAL2	PRU-ICSS2 PRU0 IOSET2 Direct Output Mode Timings
PR2_PRU1_DIR_IN_MANUAL1	PRU-ICSS2 PRU1 IOSET1 Direct Input Mode Timings
PR2_PRU1_DIR_IN_MANUAL2	PRU-ICSS2 PRU1 IOSET2 Direct Input Mode Timings
PR2_PRU1_DIR_OUT_MANUAL1	PRU-ICSS2 PRU1 IOSET1 Direct Output Mode Timings
PR2_PRU1_DIR_OUT_MANUAL2	PRU-ICSS2 PRU1 IOSET2 Direct Output Mode Timings
PR2_PRU0_PAR_CAP_MANUAL2	PRU-ICSS2 PRU0 IOSET2 Parallel Capture Mode Timings
PR2_PRU1_PAR_CAP_MANUAL1	PRU-ICSS2 PRU1 IOSET1 Parallel Capture Mode Timings
PR2_PRU1_PAR_CAP_MANUAL2	PRU-ICSS2 PRU1 IOSET2 Parallel Capture Mode Timings
HDMI, EMIF, Timers, I2C, HDQ/1-Wire, UART, McSPI, USB, PCIe, DCAN, GPIO, KBD, PWM, JTAG, TPIU, SDMA, INTC, MLB	
No Virtual or Manual IO Timing Mode Required	All Modes

5.10.6.3 VIP

The Device includes 1 Video Input Port (VIP).

Table 5-30, Figure 5-20 and Figure 5-21 present timings and switching characteristics of the VIP.

CAUTION

The I/O timings provided in this section are valid only for VIN1 and VIN2 if signals within a single IOSET are used. The IOSETs are defined in Table 5-31.

Table 5-30. Timing Requirements for VIP (3)(4)(5)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
V1	$t_{c(CLK)}$	Cycle time, vinx_clki (3) (5)	6.06 (2)		ns
V2	$t_{w(CLKH)}$	Pulse duration, vinx_clki high (3) (5)	0.45 × P (2)		ns
V3	$t_{w(CLKL)}$	Pulse duration, vinx_clki low (3) (5)	0.45 × P (2)		ns
V4	$t_{su(CTL/DATA-CLK)}$	Input setup time, Control (vinx_dei, vinx_vsynci, vinx_fldi, vinx_hsynci) and Data (vinx_dn) valid to vinx_clki transition (3) (4) (5)	3.11 (2)		ns
V6	$t_{h(CLK-CTL/DATA)}$	Input hold time, Control (vinx_dei, vinx_vsynci, vinx_fldi, vinx_hsynci) and Data (vinx_dn) valid from vinx_clki transition (3) (4) (5)	-0.05 (2)		ns

(1) For maximum frequency of 165 MHz.

(2) P = vinx_clki period.

(3) x in vinx = 1a, 1b, 2a, 2b.

(4) n in dn = 0 to 7 when x = 1b, 2b.
n = 0 to 23 when x = 1a, 2a.

(5) i in clki, dei, vsynci, hsynci and fldi = 0 or 1.

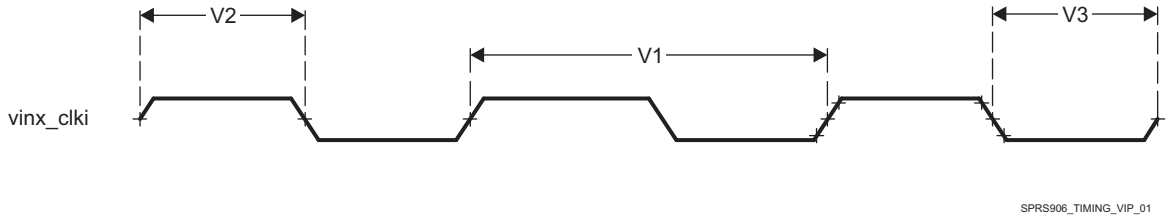


Figure 5-20. Video Input Ports Clock Signal

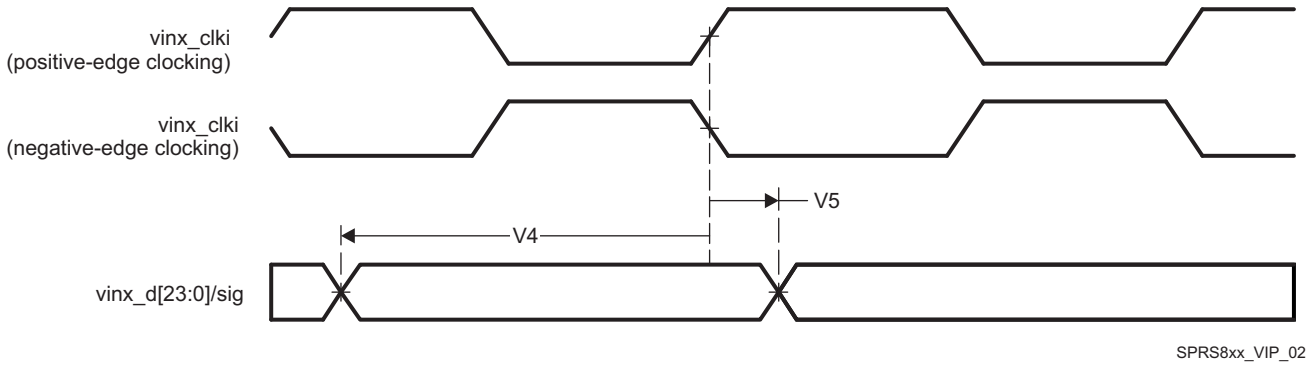


Figure 5-21. Video Input Ports Timings

In Table 5-31 and Table 5-32 are presented the specific groupings of signals (IOSET) for use with vin1 and vin2.

Table 5-31. VIN1 IOSETs

SIGNALS	IOSET2		IOSET6 (1)		IOSET7 (1)		IOSET8		IOSET9		IOSET10	
	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX
vin1a												
vin1a_clk0	G3	2					Y5	9	J24	7	J24	7
vin1a_hsync0	K4	2					AA4	9	B14	7	B14	7
vin1a_vsync0	H1	2					AB1	9	D14	7	D14	7
vin1a_fld0	L3	2							C16	7	C16	7
vin1a_de0	J2	2					Y6	9	C17	7	C17	7
vin1a_d0	F1	2					AA1	9	J25	7	B23	7
vin1a_d1	E2	2					Y3	9	B22	7	B22	7
vin1a_d2	E1	2					W2	9	A23	7	A23	7
vin1a_d3	C1	2					AA3	9	A22	7	A22	7
vin1a_d4	D1	2					AA2	9	B21	7	B21	7
vin1a_d5	D2	2					Y4	9	A21	7	A21	7
vin1a_d6	B1	2					Y1	9	D19	7	D19	7
vin1a_d7	B2	2					Y2	9	E19	7	E19	7
vin1a_d8	C2	2							F16	7	F16	7
vin1a_d9	D3	2							E16	7	E16	7
vin1a_d10	A2	2							E17	7	E17	7
vin1a_d11	B3	2							A19	7	A19	7
vin1a_d12	C3	2							B18	7	B18	7
vin1a_d13	C4	2							B16	7	B16	7
vin1a_d14	A3	2							B17	7	B17	7
vin1a_d15	B4	2							A18	7	A18	7
vin1a_d16	M1	2										
vin1a_d17	M2	2										
vin1a_d18	L2	2										
vin1a_d19	L1	2										
vin1a_d20	K3	2										
vin1a_d21	K2	2										
vin1a_d22	J1	2										
vin1a_d23	K1	2										

Table 5-31. VIN1 IOSETs (continued)

SIGNALS	IOSET2		IOSET6 (1)		IOSET7 (1)		IOSET8		IOSET9		IOSET10	
	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX
vin1b												
vin1b_clk1			L5	5	J2	6						
vin1b_hsync1			P3	5	K4	6						
vin1b_vsync1			R2	5	H1	6						
vin1b_fld1			N4	5	G1	6						
vin1b_de1			P4	5	L3	6						
vin1b_d0			L6	5	M1	6						
vin1b_d1			N5	5	M2	6						
vin1b_d2			N6	5	L2	6						
vin1b_d3			T4	5	L1	6						
vin1b_d4			T5	5	K3	6						
vin1b_d5			N2	5	K2	6						
vin1b_d6			P2	5	J1	6						
vin1b_d7			N1	5	K1	6						

(1) The IOSET under this column is only applicable for pins with alternate functionality which allows either VIN1 or VIN2 signals to be mapped to the pins. These alternate functions are controlled via CTRL_CORE_VIP_MUX_SELECT register. For more information on how to use these options, please refer to Device TRM, Chapter Control Module, Section Pad Configuration Registers.

Table 5-32. VIN2 IOSETs

SIGNALS	IOSET1		IOSET2		IOSET4		IOSET5		IOSET6		IOSET7 (1)		IOSET8 (1)		IOSET9 (1)	
	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX
vin2a																
vin2a_clk0					D8	0	D8	0	L5	4						
vin2a_hsync0					E8	0	E8	0	P3	4						
vin2a_vsync0					B8	0	B8	0	R2	4						
vin2a_fld0					C7	0	B7	1	N4	4						
vin2a_de0					B7	0			P4	4						
vin2a_d0					C8	0	C8	0	L6	4						
vin2a_d1					B9	0	B9	0	N5	4						
vin2a_d2					A7	0	A7	0	N6	4						
vin2a_d3					A9	0	A9	0	T4	4						
vin2a_d4					A8	0	A8	0	T5	4						

Table 5-32. VIN2 IOSETs (continued)

SIGNALS	IOSET1		IOSET2		IOSET4		IOSET5		IOSET6		IOSET7 (1)		IOSET8 (1)		IOSET9 (1)	
	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX
vin2a_d5					A11	0	A11	0	N2	4						
vin2a_d6					F10	0	F10	0	P2	4						
vin2a_d7					A10	0	A10	0	N1	4						
vin2a_d8					B10	0	B10	0	P1	4						
vin2a_d9					E10	0	E10	0	N3	4						
vin2a_d10					D10	0	D10	0	R1	4						
vin2a_d11					C10	0	C10	0	P5	4						
vin2a_d12					B11	0	B11	0								
vin2a_d13					D11	0	D11	0								
vin2a_d14					C11	0	C11	0								
vin2a_d15					B12	0	B12	0								
vin2a_d16					A12	0	A12	0								
vin2a_d17					A13	0	A13	0								
vin2a_d18					E11	0	E11	0								
vin2a_d19					F11	0	F11	0								
vin2a_d20					B13	0	B13	0								
vin2a_d21					E13	0	E13	0								
vin2a_d22					C13	0	C13	0								
vin2a_d23					D13	0	D13	0								
vin2b																
vin2b_clk1	L4	6	H6	4							C7	2	C7	2	AB1	4
vin2b_hsync1	B6	6	B6	6							E8	3	E8	3	Y5	4
vin2b_vsync1	A6	6	A6	6							B8	3	B8	3	Y6	4
vin2b_fld1	H6	6											B7	2		
vin2b_de1	H2	6	H2	6							B7	3			AA4	4
vin2b_d0	A4	6	A4	6							D13	2	D13	2	AA1	4
vin2b_d1	E7	6	E7	6							C13	2	C13	2	Y3	4
vin2b_d2	D6	6	D6	6							E13	2	E13	2	W2	4
vin2b_d3	C5	6	C5	6							B13	2	B13	2	AA3	4
vin2b_d4	B5	6	B5	6							F11	2	F11	2	AA2	4
vin2b_d5	D7	6	D7	6							E11	2	E11	2	Y4	4
vin2b_d6	C6	6	C6	6							A13	2	A13	2	Y1	4

Table 5-32. VIN2 IOSETs (continued)

SIGNALS	IOSET1		IOSET2		IOSET4		IOSET5		IOSET6		IOSET7 (1)		IOSET8 (1)		IOSET9 (1)	
	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX
vin2b_d7	A5	6	A5	6							A12	2	A12	2	Y2	4

(1) The IOSET under this column is only applicable for pins with alternate functionality which allows either VIN1 or VIN2 signals to be mapped to the pins. These alternate functions are controlled via CTRL_CORE_VIP_MUX_SELECT register. For more information on how to use these options, please refer to Device TRM, Chapter Control Module, Section Pad Configuration Registers.

NOTE

To configure the desired Manual IO Timing Mode the user must follow the steps described in section "*Manual IO Timing Modes*" of the Device TRM.

The associated registers to configure are listed in the **CFG REGISTER** column. For more information please see the *Control Module Chapter* in the Device TRM.

Manual IO Timings Modes must be used to ensure some IO timings for VIP1. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-33 *Manual Functions Mapping for VIN2A (IOSET4/5/6)* for a definition of the Manual modes.

Table 5-33 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-33. Manual Functions Mapping for VIN2A (IOSET4/5/6)

BALL	BALL NAME	VIP_MANUAL3		VIP_MANUAL5		CFG REGISTER	MUXMODE		
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		0	1	4
P5	RMII_MHZ_50_CLK	2616	1379	2798	1294	CFG_RMII_MHZ_50_CLK_IN	-	-	vin2a_d11
L6	mdio_d	2558	1105	2790	954	CFG_MDIO_D_IN	-	-	vin2a_d0
L5	mdio_mclk	998	463	1029	431	CFG_MDIO_MCLK_IN	-	-	vin2a_clk0
N2	rgmii0_rxc	2658	862	2896	651	CFG_RGMII0_RXC_IN	-	-	vin2a_d5
P2	rgmii0_rxctl	2658	1628	2844	1518	CFG_RGMII0_RXCTL_IN	-	-	vin2a_d6
N4	rgmii0_rxd0	2638	1123	2856	888	CFG_RGMII0_RXD0_IN	-	-	vin2a_fld0
N3	rgmii0_rxd1	2641	1737	2804	1702	CFG_RGMII0_RXD1_IN	-	-	vin2a_d9
P1	rgmii0_rxd2	2641	1676	2801	1652	CFG_RGMII0_RXD2_IN	-	-	vin2a_d8
N1	rgmii0_rxd3	2644	1828	2807	1790	CFG_RGMII0_RXD3_IN	-	-	vin2a_d7
T4	rgmii0_txc	2638	1454	2835	1396	CFG_RGMII0_TXC_IN	-	-	vin2a_d3
T5	rgmii0_txctl	2672	1663	2831	1640	CFG_RGMII0_TXCTL_IN	-	-	vin2a_d4
R1	rgmii0_txd0	2604	1442	2764	1417	CFG_RGMII0_TXD0_IN	-	-	vin2a_d10
R2	rgmii0_txd1	2683	1598	2843	1600	CFG_RGMII0_TXD1_IN	-	-	vin2a_vsync0
P3	rgmii0_txd2	2563	1483	2816	1344	CFG_RGMII0_TXD2_IN	-	-	vin2a_hsync0

Table 5-33. Manual Functions Mapping for VIN2A (IOSET4/5/6) (continued)

BALL	BALL NAME	VIP_MANUAL3		VIP_MANUAL5		CFG REGISTER	MUXMODE		
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		0	1	4
P4	rgmii0_txd3	2717	1461	2913	1310	CFG_RGMII0_TXD3_IN	-	-	vin2a_de0
N5	uart3_rxd	2445	1145	2743	923	CFG_UART3_RXD_IN	-	-	vin2a_d1
N6	uart3_txd	2650	1197	2842	1080	CFG_UART3_TXD_IN	-	-	vin2a_d2
D8	vin2a_clk0	0	0	0	0	CFG_VIN2A_CLK0_IN	vin2a_clk0	-	-
C8	vin2a_d0	1812	102	1936	0	CFG_VIN2A_D0_IN	vin2a_d0	-	-
B9	vin2a_d1	1701	439	2229	10	CFG_VIN2A_D1_IN	vin2a_d1	-	-
D10	vin2a_d10	1720	215	2031	0	CFG_VIN2A_D10_IN	vin2a_d10	-	-
C10	vin2a_d11	1622	0	1702	0	CFG_VIN2A_D11_IN	vin2a_d11	-	-
B11	vin2a_d12	1350	412	1819	0	CFG_VIN2A_D12_IN	vin2a_d12	-	-
D11	vin2a_d13	1613	147	1476	260	CFG_VIN2A_D13_IN	vin2a_d13	-	-
C11	vin2a_d14	1149	516	1701	0	CFG_VIN2A_D14_IN	vin2a_d14	-	-
B12	vin2a_d15	1530	450	2021	0	CFG_VIN2A_D15_IN	vin2a_d15	-	-
A12	vin2a_d16	1512	449	2044	11	CFG_VIN2A_D16_IN	vin2a_d16	-	-
A13	vin2a_d17	1293	488	1839	5	CFG_VIN2A_D17_IN	vin2a_d17	-	-
E11	vin2a_d18	2140	371	2494	0	CFG_VIN2A_D18_IN	vin2a_d18	-	-
F11	vin2a_d19	2041	275	1699	611	CFG_VIN2A_D19_IN	vin2a_d19	-	-
A7	vin2a_d2	1675	35	1736	0	CFG_VIN2A_D2_IN	vin2a_d2	-	-
B13	vin2a_d20	1972	441	2412	88	CFG_VIN2A_D20_IN	vin2a_d20	-	-
E13	vin2a_d21	1957	556	2391	161	CFG_VIN2A_D21_IN	vin2a_d21	-	-
C13	vin2a_d22	2011	433	2446	102	CFG_VIN2A_D22_IN	vin2a_d22	-	-
D13	vin2a_d23	1962	523	2395	145	CFG_VIN2A_D23_IN	vin2a_d23	-	-
A9	vin2a_d3	1457	361	1943	0	CFG_VIN2A_D3_IN	vin2a_d3	-	-
A8	vin2a_d4	1535	0	1601	0	CFG_VIN2A_D4_IN	vin2a_d4	-	-
A11	vin2a_d5	1676	271	2052	0	CFG_VIN2A_D5_IN	vin2a_d5	-	-
F10	vin2a_d6	1513	0	1571	0	CFG_VIN2A_D6_IN	vin2a_d6	-	-
A10	vin2a_d7	1616	141	1855	0	CFG_VIN2A_D7_IN	vin2a_d7	-	-
B10	vin2a_d8	1286	437	1224	618	CFG_VIN2A_D8_IN	vin2a_d8	-	-
E10	vin2a_d9	1544	265	1373	509	CFG_VIN2A_D9_IN	vin2a_d9	-	-
B7	vin2a_de0	1732	208	1949	0	CFG_VIN2A_DE0_IN	vin2a_de0	vin2a_fld0	-
C7	vin2a_fld0	1461	562	1983	151	CFG_VIN2A_FLD0_IN	vin2a_fld0	-	-
E8	vin2a_hsync0	1877	0	1943	0	CFG_VIN2A_HSYNC0_IN	vin2a_hsync0	-	-
B8	vin2a_vsync0	1566	0	1612	0	CFG_VIN2A_VSYNC0_IN	vin2a_vsync0	-	-

Manual IO Timings Modes must be used to ensure some IO timings for VIP1. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-34 *Manual Functions Mapping for VIN2B (IOSET7/8/9)* for a definition of the Manual modes.

Table 5-34 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-34. Manual Functions Mapping for VIN2B (IOSET7/8/9)

BALL	BALL NAME	VIP_MANUAL4		VIP_MANUAL6		CFG REGISTER	MUXMODE		
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		2	3	4
Y5	gpio6_10	2829	884	3009	892	CFG_GPIO6_10_IN	-	-	vin2b_hsync1
Y6	gpio6_11	2648	1033	2890	1096	CFG_GPIO6_11_IN	-	-	vin2b_vsync1
Y2	mmc3_clk	2794	1074	2997	1089	CFG_MMC3_CLK_IN	-	-	vin2b_d7
Y1	mmc3_cmd	2789	1162	2959	1210	CFG_MMC3_CMD_IN	-	-	vin2b_d6
Y4	mmc3_dat0	2689	1180	2897	1269	CFG_MMC3_DAT0_IN	-	-	vin2b_d5
AA2	mmc3_dat1	2605	1219	2891	1219	CFG_MMC3_DAT1_IN	-	-	vin2b_d4
AA3	mmc3_dat2	2616	703	2947	590	CFG_MMC3_DAT2_IN	-	-	vin2b_d3
W2	mmc3_dat3	2760	1235	2931	1342	CFG_MMC3_DAT3_IN	-	-	vin2b_d2
Y3	mmc3_dat4	2757	880	2979	891	CFG_MMC3_DAT4_IN	-	-	vin2b_d1
AA1	mmc3_dat5	2688	1177	2894	1262	CFG_MMC3_DAT5_IN	-	-	vin2b_d0
AA4	mmc3_dat6	2638	1165	2894	1187	CFG_MMC3_DAT6_IN	-	-	vin2b_de1
AB1	mmc3_dat7	995	182	1202	107	CFG_MMC3_DAT7_IN	-	-	vin2b_clk1
A12	vin2a_d16	1423	0	1739	0	CFG_VIN2A_D16_IN	vin2b_d7	-	-
A13	vin2a_d17	1253	0	1568	0	CFG_VIN2A_D17_IN	vin2b_d6	-	-
E11	vin2a_d18	2080	0	2217	0	CFG_VIN2A_D18_IN	vin2b_d5	-	-
F11	vin2a_d19	1849	0	2029	0	CFG_VIN2A_D19_IN	vin2b_d4	-	-
B13	vin2a_d20	1881	50	2202	0	CFG_VIN2A_D20_IN	vin2b_d3	-	-
E13	vin2a_d21	1917	167	2313	0	CFG_VIN2A_D21_IN	vin2b_d2	-	-
C13	vin2a_d22	1955	79	2334	0	CFG_VIN2A_D22_IN	vin2b_d1	-	-
D13	vin2a_d23	1899	145	2288	0	CFG_VIN2A_D23_IN	vin2b_d0	-	-
B7	vin2a_de0	1568	261	2048	0	CFG_VIN2A_DE0_IN	vin2b_fld1	vin2b_de1	-
C7	vin2a_fld0	0	0	0	0	CFG_VIN2A_FLD0_IN	vin2b_clk1	-	-
E8	vin2a_hsync0	1793	0	2011	0	CFG_VIN2A_HSYNC0_IN	-	vin2b_hsync1	-
B8	vin2a_vsync0	1382	0	1632	0	CFG_VIN2A_VSYNC0_IN	-	vin2b_vsync1	-

Manual IO Timings Modes must be used to ensure some IO timings for VIP1. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-35 *Manual Functions Mapping for VIN1A (IOSET2) and VIN2B (IOSET1/10)* for a definition of the Manual modes.

Table 5-35 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-35. Manual Functions Mapping for VIN1A (IOSET2) and VIN2B (IOSET1/10)

BALL	BALL NAME	VIP_MANUAL7		VIP_MANUAL12		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		2	6
M1	gpmc_a0	3080	1792	3376	1632	CFG_GPMC_A0_IN	vin1a_d16	-
M2	gpmc_a1	2958	1890	3249	1749	CFG_GPMC_A1_IN	vin1a_d17	-
J2	gpmc_a10	3073	1653	3388	1433	CFG_GPMC_A10_IN	vin1a_de0	-
L3	gpmc_a11	3014	1784	3290	1693	CFG_GPMC_A11_IN	vin1a_fld0	-
A4	gpmc_a19	1385	0	1246	0	CFG_GPMC_A19_IN	-	vin2b_d0
L2	gpmc_a2	3041	1960	3322	1850	CFG_GPMC_A2_IN	vin1a_d18	-
E7	gpmc_a20	859	0	720	0	CFG_GPMC_A20_IN	-	vin2b_d1
D6	gpmc_a21	1465	0	1334	0	CFG_GPMC_A21_IN	-	vin2b_d2
C5	gpmc_a22	1210	0	1064	0	CFG_GPMC_A22_IN	-	vin2b_d3
B5	gpmc_a23	1111	0	954	0	CFG_GPMC_A23_IN	-	vin2b_d4
D7	gpmc_a24	1137	0	1051	0	CFG_GPMC_A24_IN	-	vin2b_d5
C6	gpmc_a25	1402	0	1283	0	CFG_GPMC_A25_IN	-	vin2b_d6
A5	gpmc_a26	1298	0	1153	0	CFG_GPMC_A26_IN	-	vin2b_d7
B6	gpmc_a27	934	0	870	0	CFG_GPMC_A27_IN	-	vin2b_hsync1
L1	gpmc_a3	3019	2145	3296	2050	CFG_GPMC_A3_IN	vin1a_d19	-
K3	gpmc_a4	3063	1981	3357	1829	CFG_GPMC_A4_IN	vin1a_d20	-
K2	gpmc_a5	3021	1954	3304	1840	CFG_GPMC_A5_IN	vin1a_d21	-
J1	gpmc_a6	3062	1716	3348	1592	CFG_GPMC_A6_IN	vin1a_d22	-
K1	gpmc_a7	3260	1889	3583	1631	CFG_GPMC_A7_IN	vin1a_d23	-
K4	gpmc_a8	3033	1702	3328	1547	CFG_GPMC_A8_IN	vin1a_hsync0	-
H1	gpmc_a9	2991	1905	3281	1766	CFG_GPMC_A9_IN	vin1a_vsync0	-
F1	gpmc_ad0	2907	1342	3181	1255	CFG_GPMC_AD0_IN	vin1a_d0	-
E2	gpmc_ad1	2858	1321	3132	1234	CFG_GPMC_AD1_IN	vin1a_d1	-
A2	gpmc_ad10	2920	1384	3223	1204	CFG_GPMC_AD10_IN	vin1a_d10	-
B3	gpmc_ad11	2719	1310	3019	1198	CFG_GPMC_AD11_IN	vin1a_d11	-
C3	gpmc_ad12	2845	1135	3160	917	CFG_GPMC_AD12_IN	vin1a_d12	-
C4	gpmc_ad13	2765	1225	3045	1119	CFG_GPMC_AD13_IN	vin1a_d13	-

Table 5-35. Manual Functions Mapping for VIN1A (IOSET2) and VIN2B (IOSET1/10) (continued)

BALL	BALL NAME	VIP_MANUAL7		VIP_MANUAL12		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		2	6
A3	gpmc_ad14	2845	1150	3153	952	CFG_GPMC_AD14_IN	vin1a_d14	-
B4	gpmc_ad15	2766	1453	3044	1355	CFG_GPMC_AD15_IN	vin1a_d15	-
E1	gpmc_ad2	2951	1296	3226	1209	CFG_GPMC_AD2_IN	vin1a_d2	-
C1	gpmc_ad3	2825	1154	3121	997	CFG_GPMC_AD3_IN	vin1a_d3	-
D1	gpmc_ad4	2927	1245	3246	1014	CFG_GPMC_AD4_IN	vin1a_d4	-
D2	gpmc_ad5	2923	1251	3217	1098	CFG_GPMC_AD5_IN	vin1a_d5	-
B1	gpmc_ad6	2958	1342	3238	1239	CFG_GPMC_AD6_IN	vin1a_d6	-
B2	gpmc_ad7	2900	1244	3174	1157	CFG_GPMC_AD7_IN	vin1a_d7	-
C2	gpmc_ad8	2845	1585	3125	1482	CFG_GPMC_AD8_IN	vin1a_d8	-
D3	gpmc_ad9	2779	1343	3086	1223	CFG_GPMC_AD9_IN	vin1a_d9	-
H2	gpmc_ben0	1555	0	1425	0	CFG_GPMC_BEN0_IN	-	vin2b_de1
H6	gpmc_ben1	1501	0	1397	0	CFG_GPMC_BEN1_IN	-	vin2b_fld1
L4	gpmc_clk	0	0	0	0	CFG_GPMC_CLK_IN	-	vin2b_clk1
A6	gpmc_cs1	1192	0	1102	0	CFG_GPMC_CS1_IN	-	vin2b_vsync1
G3	gpmc_cs3	1324	374	1466	353	CFG_GPMC_CS3_IN	vin1a_clk0	-

Manual IO Timings Modes must be used to ensure some IO timings for VIP1. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-36 *Manual Functions Mapping for VIN1B (IOSET6/7)* for a definition of the Manual modes.

Table 5-36 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-36. Manual Functions Mapping for VIN1B (IOSET6/7)

BALL	BALL NAME	VIP_MANUAL9		VIP_MANUAL14		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		5	6
M1	gpmc_a0	1873	702	2202	441	CFG_GPMC_A0_IN	-	vin1b_d0
M2	gpmc_a1	1629	772	2057	413	CFG_GPMC_A1_IN	-	vin1b_d1
J2	gpmc_a10	0	0	0	0	CFG_GPMC_A10_IN	-	vin1b_clk1
L3	gpmc_a11	1851	1011	2126	856	CFG_GPMC_A11_IN	-	vin1b_de1
G1	gpmc_a12	2009	601	2289	327	CFG_GPMC_A12_IN	-	vin1b_fld1
L2	gpmc_a2	1734	898	2131	573	CFG_GPMC_A2_IN	-	vin1b_d2
L1	gpmc_a3	1757	1076	2106	812	CFG_GPMC_A3_IN	-	vin1b_d3
K3	gpmc_a4	1794	893	2164	559	CFG_GPMC_A4_IN	-	vin1b_d4
K2	gpmc_a5	1726	853	2120	523	CFG_GPMC_A5_IN	-	vin1b_d5

Table 5-36. Manual Functions Mapping for VIN1B (IOSET6/7) (continued)

BALL	BALL NAME	VIP_MANUAL9		VIP_MANUAL14		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		5	6
J1	gpmc_a6	1792	612	2153	338	CFG_GPMC_A6_IN	-	vin1b_d6
K1	gpmc_a7	2117	610	2389	304	CFG_GPMC_A7_IN	-	vin1b_d7
K4	gpmc_a8	1758	653	2140	308	CFG_GPMC_A8_IN	-	vin1b_hsync1
H1	gpmc_a9	1705	899	2067	646	CFG_GPMC_A9_IN	-	vin1b_vsync1
L6	mdio_d	1945	671	2265	414	CFG_MDIO_D_IN	vin1b_d0	-
L5	mdio_mclk	255	119	337	0	CFG_MDIO_MCLK_IN	vin1b_clk1	-
N2	rgmii0_rxc	2057	909	2341	646	CFG_RGMII0_RXC_IN	vin1b_d5	-
P2	rgmii0_rxctl	2121	1139	2323	988	CFG_RGMII0_RXCTL_IN	vin1b_d6	-
N4	rgmii0_rxd0	2070	655	2336	340	CFG_RGMII0_RXD0_IN	vin1b_fld1	-
N1	rgmii0_rxd3	2092	1357	2306	1216	CFG_RGMII0_RXD3_IN	vin1b_d7	-
T4	rgmii0_txc	2088	1205	2328	1079	CFG_RGMII0_TXC_IN	vin1b_d3	-
T5	rgmii0_txctl	2143	1383	2312	1311	CFG_RGMII0_TXCTL_IN	vin1b_d4	-
R2	rgmii0_txd1	2078	1189	2324	1065	CFG_RGMII0_TXD1_IN	vin1b_vsync1	-
P3	rgmii0_txd2	1928	1125	2306	763	CFG_RGMII0_TXD2_IN	vin1b_hsync1	-
P4	rgmii0_txd3	2255	971	2401	846	CFG_RGMII0_TXD3_IN	vin1b_de1	-
N5	uart3_rxd	1829	747	2220	400	CFG_UART3_RXD_IN	vin1b_d1	-
N6	uart3_txd	2030	837	2324	568	CFG_UART3_TXD_IN	vin1b_d2	-

Manual IO Timings Modes must be used to ensure some IO timings for VIP1. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-37 *Manual Functions Mapping for VIN2B (IOSET2/11)* for a definition of the Manual modes.

Table 5-37 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-37. Manual Functions Mapping for VIN2B (IOSET2/11)

BALL	BALL NAME	VIP_MANUAL10		VIP_MANUAL11		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		4	6
A4	gpmc_a19	1600	943	2023	477	CFG_GPMC_A19_IN	-	vin2b_d0
E7	gpmc_a20	1440	621	1875	136	CFG_GPMC_A20_IN	-	vin2b_d1
D6	gpmc_a21	1602	1066	2021	604	CFG_GPMC_A21_IN	-	vin2b_d2
C5	gpmc_a22	1395	983	1822	519	CFG_GPMC_A22_IN	-	vin2b_d3
B5	gpmc_a23	1571	716	2045	200	CFG_GPMC_A23_IN	-	vin2b_d4
D7	gpmc_a24	1463	832	1893	396	CFG_GPMC_A24_IN	-	vin2b_d5
C6	gpmc_a25	1426	1166	1842	732	CFG_GPMC_A25_IN	-	vin2b_d6

Table 5-37. Manual Functions Mapping for VIN2B (IOSET2/11) (continued)

BALL	BALL NAME	VIP_MANUAL10		VIP_MANUAL11		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		4	6
A5	gpmc_a26	1362	1094	1797	584	CFG_GPMC_A26_IN	-	vin2b_d7
B6	gpmc_a27	1283	809	1760	338	CFG_GPMC_A27_IN	-	vin2b_hsync1
H2	gpmc_ben0	1978	780	2327	389	CFG_GPMC_BEN0_IN	-	vin2b_de1
H6	gpmc_ben1	0	0	0	0	CFG_GPMC_BEN1_IN	vin2b_clk1	-
A6	gpmc_cs1	1411	982	1857	536	CFG_GPMC_CS1_IN	-	vin2b_vsync1

Manual IO Timings Modes must be used to ensure some IO timings for VIP1. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-38 *Manual Functions Mapping for VIN1A (IOSET8/9/10)* for a definition of the Manual modes.

Table 5-38 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-38. Manual Functions Mapping for VIN1A (IOSET8/9/10)

BALL	BALL NAME	VIP_MANUAL15		VIP_MANUAL16		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		7	9
Y5	gpio6_10	2131	2198	2170	2180	CFG_GPIO6_10_IN	-	vin1a_clk0
Y6	gpio6_11	3720	2732	4106	2448	CFG_GPIO6_11_IN	-	vin1a_de0
C16	mcasp1_aclkx	2447	0	3042	0	CFG_MCASP1_ACLKX_IN	vin1a_fld0	-
D14	mcasp1_axr0	3061	0	3380	292	CFG_MCASP1_AXR0_IN	vin1a_vsync0	-
B14	mcasp1_axr1	3113	0	3396	304	CFG_MCASP1_AXR1_IN	vin1a_hsync0	-
B16	mcasp1_axr10	2803	0	3362	0	CFG_MCASP1_AXR10_IN	vin1a_d13	-
B18	mcasp1_axr11	3292	0	3357	546	CFG_MCASP1_AXR11_IN	vin1a_d12	-
A19	mcasp1_axr12	2854	0	3145	320	CFG_MCASP1_AXR12_IN	vin1a_d11	-
E17	mcasp1_axr13	2813	0	3229	196	CFG_MCASP1_AXR13_IN	vin1a_d10	-
E16	mcasp1_axr14	2471	0	3053	0	CFG_MCASP1_AXR14_IN	vin1a_d9	-
F16	mcasp1_axr15	2815	0	3225	201	CFG_MCASP1_AXR15_IN	vin1a_d8	-
A18	mcasp1_axr8	2965	0	3427	83	CFG_MCASP1_AXR8_IN	vin1a_d15	-
B17	mcasp1_axr9	3082	0	3253	440	CFG_MCASP1_AXR9_IN	vin1a_d14	-
C17	mcasp1_fsx	2898	0	3368	139	CFG_MCASP1_FSX_IN	vin1a_de0	-
E19	mcasp2_aclkx	2413	0	2972	0	CFG_MCASP2_ACLKX_IN	vin1a_d7	-
A21	mcasp2_axr2	2478	0	3062	0	CFG_MCASP2_AXR2_IN	vin1a_d5	-
B21	mcasp2_axr3	2806	0	3175	242	CFG_MCASP2_AXR3_IN	vin1a_d4	-
D19	mcasp2_fsx	2861	78	2936	599	CFG_MCASP2_FSX_IN	vin1a_d6	-
A22	mcasp3_aclkx	1583	0	1878	0	CFG_MCASP3_ACLKX_IN	vin1a_d3	-

Table 5-38. Manual Functions Mapping for VIN1A (IOSET8/9/10) (continued)

BALL	BALL NAME	VIP_MANUAL15		VIP_MANUAL16		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		7	9
B22	mcasp3_axr0	2873	0	3109	375	CFG_MCASP3_AXR0_IN	vin1a_d1	-
B23	mcasp3_axr1	1625	1400	2072	1023	CFG_MCASP3_AXR1_IN	vin1a_d0	-
A23	mcasp3_fsx	2792	0	3146	257	CFG_MCASP3_FSX_IN	vin1a_d2	-
Y2	mmc3_clk	3907	2744	4260	2450	CFG_MMC3_CLK_IN	-	vin1a_d7
Y1	mmc3_cmd	3892	2768	4242	2470	CFG_MMC3_CMD_IN	-	vin1a_d6
Y4	mmc3_dat0	3786	2765	4156	2522	CFG_MMC3_DAT0_IN	-	vin1a_d5
AA2	mmc3_dat1	3673	2961	4053	2667	CFG_MMC3_DAT1_IN	-	vin1a_d4
AA3	mmc3_dat2	3818	2447	4209	2096	CFG_MMC3_DAT2_IN	-	vin1a_d3
W2	mmc3_dat3	3902	2903	4259	2672	CFG_MMC3_DAT3_IN	-	vin1a_d2
Y3	mmc3_dat4	3905	2622	4259	2342	CFG_MMC3_DAT4_IN	-	vin1a_d1
AA1	mmc3_dat5	3807	2824	4167	2595	CFG_MMC3_DAT5_IN	-	vin1a_d0
AA4	mmc3_dat6	3724	2818	4123	2491	CFG_MMC3_DAT6_IN	-	vin1a_hsync0
AB1	mmc3_dat7	3775	2481	4159	2161	CFG_MMC3_DAT7_IN	-	vin1a_vsync0
J25	xref_clk0	1971	0	2472	0	CFG_XREF_CLK0_IN	vin1a_d0	-
J24	xref_clk1	0	192	0	603	CFG_XREF_CLK1_IN	vin1a_clk0	-

5.10.6.4 DSS

Two Display Parallel Interfaces (DPI) channels are available in DSS named DPI Video Output 2 and DPI Video Output 3.

NOTE

The DPI Video Output i (i = 2, 3) interface is also referred to as VOUTi.

Every VOUT interface consists of:

- 24-bit data bus (data[23:0])
- Horizontal synchronization signal (HSYNC)
- Vertical synchronization signal (VSYNC)
- Data enable (DE)
- Field ID (FID)
- Pixel clock (CLK)

NOTE

For more information, see the Display Subsystem chapter of the Device TRM.

CAUTION

The I/O Timings provided in this section are valid only if signals within a single IOSET are used. The IOSETs are defined in Table 5-43.

CAUTION

The I/O Timings provided in this section are valid only for some DSS usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

CAUTION

All pads/balls configured as vouti_* signals must be programmed to use slow slew rate by setting the corresponding CTRL_CORE_PAD_*[SLEWCONTROL] register field to SLOW (0b1).

Table 5-39, Table 5-40 through Table 5-42 assume testing over the recommended operating conditions and electrical characteristic conditions.

Table 5-39. DPI Video Output i (i = 2, 3) Default Switching Characteristics(1)(2)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
D1	$t_{c(clk)}$	Cycle time, output pixel clock vouti_clk	DPI2/3	11.76		ns
D2	$t_{w(clkL)}$	Pulse duration, output pixel clock vouti_clk low		P × 0.5-1 (1)		ns
D3	$t_{w(clkH)}$	Pulse duration, output pixel clock vouti_clk high		P × 0.5-1 (1)		ns

Table 5-39. DPI Video Output i (i = 2, 3) Default Switching Characteristics[1][2] (continued)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI2 (vin2a_fld0 clock reference)	-2.5	2.5	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI2 (vin2a_fld0 clock reference)	-2.5	2.5	ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI3	-2.5	2.5	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI3	-2.5	2.5	ns

(1) P = output vouti_clk period in ns.

(2) All pads/balls configured as vouti_* signals must be programmed to use slow slew rate by setting the corresponding CTRL_CORE_PAD_*[SLEWCONTROL] register field to SLOW (0b1).

(3) SERDES transceivers may be sensitive to the jitter profile of vouti_clk. See Application Note Optimizing DRA7xx and TDA2xx Processors for Use With Video Display SerDes for additional guidance.

Table 5-40. DPI Video Output i (i = 2, 3) Alternate Switching Characteristics[2]

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
D1	$t_{c(clk)}$	Cycle time, output pixel clock vouti_clk	DPI2/3	6.06		ns
D2	$t_{w(clkL)}$	Pulse duration, output pixel clock vouti_clk low		P × 0.5-1 [1]		ns
D3	$t_{w(clkH)}$	Pulse duration, output pixel clock vouti_clk high		P × 0.5-1 [1]		ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI2 (vin2a_fld0 clock reference)	1.51	4.55	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI2 (vin2a_fld0 clock reference)	1.51	4.55	ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI3	1.51	4.55	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI3	1.51	4.55	ns

(1) P = output vouti_clk period in ns.

(2) All pads/balls configured as vouti_* signals must be programmed to use slow slew rate by setting the corresponding CTRL_CORE_PAD_*[SLEWCONTROL] register field to SLOW (0b1).

(3) SERDES transceivers may be sensitive to the jitter profile of vouti_clk. See Application Note Optimizing DRA7xx and TDA2xx Processors for Use With Video Display SerDes for additional guidance.

Table 5-41. DPI Video Output i (i = 2, 3) MANUAL4 Switching Characteristics [2]

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
D1	$t_{c(clk)}$	Cycle time, output pixel clock vouti_clk	DPI2/3	6.06 [3]		ns
D2	$t_{w(clkL)}$	Pulse duration, output pixel clock vouti_clk low		P*0.5-1 [1]		ns
D3	$t_{w(clkH)}$	Pulse duration, output pixel clock vouti_clk high		P*0.5-1 [1]		ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI1	2.85	5.56	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI1	2.85	5.56	ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI2 (vin2a_fld0 clock reference)	2.85	5.56	ns

Table 5-41. DPI Video Output i (i = 2, 3) MANUAL4 Switching Characteristics (2) (continued)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI2 (vin2a_fld0 clock reference)	2.85	5.56	ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI2 (xref_clk2 clock reference)	2.85	5.56	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI2 (xref_clk2 clock reference)	2.85	5.56	ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI3	2.85	5.56	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI3	2.85	5.56	ns

(1) P = output vouti_clk period in ns.

(2) All pads/balls configured as vouti_* signals must be programmed to use slow slew rate by setting the corresponding CTRL_CORE_PAD_*[SLEWCONTROL] register field to SLOW (0b1).

(3) SERDES transceivers may be sensitive to the jitter profile of vouti_clk. See Application Note Optimizing DRA7xx and TDA2xx Processors for Use With Video Display SerDes for additional guidance.

Table 5-42. DPI Video Output i (i = 2, 3) MANUAL5 Switching Characteristics (2)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
D1	$t_{c(clk)}$	Cycle time, output pixel clock vouti_clk	DPI2/3	6.06 (3)		ns
D2	$t_{w(clkL)}$	Pulse duration, output pixel clock vouti_clk low		P*0.5-1 (1)		ns
D3	$t_{w(clkH)}$	Pulse duration, output pixel clock vouti_clk high		P*0.5-1 (1)		ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI1	3.55	6.61	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI1	3.55	6.61	ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI2 (vin2a_fld0 clock reference)	3.55	6.61	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI2 (vin2a_fld0 clock reference)	3.55	6.61	ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI2 (xref_clk2 clock reference)	3.55	6.61	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI2 (xref_clk2 clock reference)	3.55	6.61	ns
D5	$t_{d(clk-ctlV)}$	Delay time, output pixel clock vouti_clk transition to output data vouti_d[23:0] valid	DPI3	3.55	6.61	ns
D6	$t_{d(clk-dV)}$	Delay time, output pixel clock vouti_clk transition to output control signals vouti_vsync, vouti_hsync, vouti_de, and vouti_fld valid	DPI3	3.55	6.61	ns

(1) P = output vouti_clk period in ns.

(2) All pads/balls configured as vouti_* signals must be programmed to use slow slew rate by setting the corresponding CTRL_CORE_PAD_*[SLEWCONTROL] register field to SLOW (0b1).

(3) SERDES transceivers may be sensitive to the jitter profile of vouti_clk. See Application Note Optimizing DRA7xx and TDA2xx Processors for Use With Video Display SerDes for additional guidance.

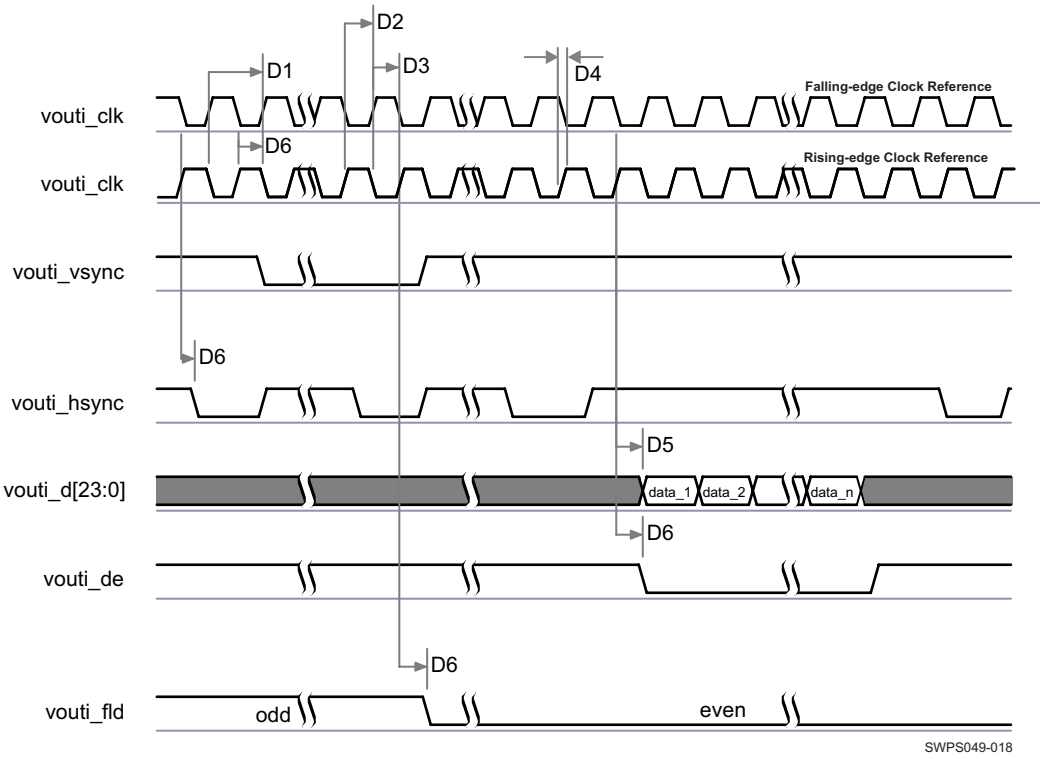


Figure 5-22. DPI Video Output(1)(2)(3)

(1) The configuration of assertion of the data can be programmed on the falling or rising edge of the pixel clock.

(2) The polarity and the pulse width of vouti_hsync and vouti_vsync are programmable, refer to the DSS section of the device TRM.

(3) The vouti_clk frequency can be configured, refer to the DSS section of the device TRM.

In Table 5-43 are presented the specific groupings of signals (IOSET) for use with VOUT2.

Table 5-43. VOUT2 IOSETs

SIGNALS	IOSET1	
	BALL	MUX
vout2_d23	C8	4
vout2_d22	B9	4
vout2_d21	A7	4
vout2_d20	A9	4
vout2_d19	A8	4
vout2_d18	A11	4
vout2_d17	F10	4
vout2_d16	A10	4
vout2_d15	B10	4
vout2_d14	E10	4
vout2_d13	D10	4
vout2_d12	C10	4
vout2_d11	B11	4

Table 5-43. VOUT2 IOSETs (continued)

SIGNALS	IOSET1	
	BALL	MUX
vout2_d10	D11	4
vout2_d9	C11	4
vout2_d8	B12	4
vout2_d7	A12	4
vout2_d6	A13	4
vout2_d5	E11	4
vout2_d4	F11	4
vout2_d3	B13	4
vout2_d2	E13	4
vout2_d1	C13	4
vout2_d0	D13	4
vout2_vsync	B8	4
vout2_hsync	E8	4
vout2_clk	C7	4
vout2_fld	D8	4
vout2_de	B7	4

NOTE

To configure the desired virtual mode the user must set MODESELECT bit and DELAYMODE bitfield for each corresponding pad control register.

The pad control registers are presented in Table 4-32 and described in Device TRM, *Control Module Chapter*.

Virtual IO Timings Modes must be used to ensure some IO timings for VOUT3. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Virtual IO Timings Modes. See Table 5-44 *Virtual Functions Mapping for VOUT3* for a definition of the Virtual modes.

Table 5-44 presents the values for DELAYMODE bitfield.

Table 5-44. Virtual Functions Mapping for DSS VOUT3

BALL	BALL NAME	Delay Mode Value	MUXMODE
		DSS_VIRTUAL1	3
B4	gpmc_ad15	14	vout3_d15
K4	gpmc_a8	15	vout3_hsync
D1	gpmc_ad4	14	vout3_d4
F1	gpmc_ad0	14	vout3_d0
C4	gpmc_ad13	14	vout3_d13
L2	gpmc_a2	15	vout3_d18
E2	gpmc_ad1	14	vout3_d1
K3	gpmc_a4	15	vout3_d20
J1	gpmc_a6	15	vout3_d22
A3	gpmc_ad14	14	vout3_d14
M2	gpmc_a1	15	vout3_d17
G3	gpmc_cs3	15	vout3_clk
H1	gpmc_a9	15	vout3_vsync
B3	gpmc_ad11	14	vout3_d11
B1	gpmc_ad6	14	vout3_d6

Table 5-44. Virtual Functions Mapping for DSS VOUT3 (continued)

BALL	BALL NAME	Delay Mode Value	MUXMODE
		DSS_VIRTUAL1	3
E1	gpmc_ad2	14	vout3_d2
C1	gpmc_ad3	14	vout3_d3
K1	gpmc_a7	15	vout3_d23
L1	gpmc_a3	15	vout3_d19
A2	gpmc_ad10	14	vout3_d10
B2	gpmc_ad7	14	vout3_d7
J2	gpmc_a10	15	vout3_de
K2	gpmc_a5	15	vout3_d21
C2	gpmc_ad8	14	vout3_d8
D2	gpmc_ad5	14	vout3_d5
M1	gpmc_a0	15	vout3_d16
C3	gpmc_ad12	14	vout3_d12
L3	gpmc_a11	15	vout3_fld
D3	gpmc_ad9	14	vout3_d9

NOTE

To configure the desired Manual IO Timing Mode the user must follow the steps described in section "*Manual IO Timing Modes*" of the Device TRM.

The associated registers to configure are listed in the **CFG REGISTER** column. For more information please see the *Control Module Chapter* in the Device TRM.

Manual IO Timings Modes must be used to ensure some IO timings for VOUT2. See Table 5-29, *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-45, *Manual Functions Mapping for DSS VOUT2 IOSET1* for a definition of the Manual modes.

Table 5-45 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-45. Manual Functions Mapping for DSS VOUT2 IOSET1

BALL	BALL NAME	VOUT2_IOSET1_MANUAL1		VOUT2_IOSET1_MANUAL2		VOUT2_IOSET1_MANUAL3		VOUT2_IOSET1_MANUAL4		VOUT2_IOSET1_MANUAL5		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		4
D8	vin2a_clk0	2571	0	1059	0	1025	0	4110	0	4980	0	CFG_VIN2A_CLK0_OUT	vout2_fld
C8	vin2a_d0	2124	0	589	0	577	0	3613	0	4483	0	CFG_VIN2A_D0_OUT	vout2_d23
B9	vin2a_d1	2103	0	568	0	557	0	3442	0	4312	0	CFG_VIN2A_D1_OUT	vout2_d22
D10	vin2a_d10	2091	0	557	0	545	0	3430	0	4200	0	CFG_VIN2A_D10_OUT	vout2_d13
C10	vin2a_d11	2142	0	608	0	596	0	3481	0	4251	0	CFG_VIN2A_D11_OUT	vout2_d12
B11	vin2a_d12	2920	385	1816	255	1783	276	3943	601	4713	601	CFG_VIN2A_D12_OUT	vout2_d11
D11	vin2a_d13	2776	322	1872	192	1838	213	3799	538	4669	538	CFG_VIN2A_D13_OUT	vout2_d10
C11	vin2a_d14	2904	0	1769	0	1757	0	3869	174	4739	174	CFG_VIN2A_D14_OUT	vout2_d9
B12	vin2a_d15	2670	257	1665	127	1632	148	3792	473	4662	473	CFG_VIN2A_D15_OUT	vout2_d8
A12	vin2a_d16	2814	155	1908	31	1878	43	3837	371	4707	371	CFG_VIN2A_D16_OUT	vout2_d7
A13	vin2a_d17	3002	199	1897	69	1865	89	4024	415	4894	415	CFG_VIN2A_D17_OUT	vout2_d6
E11	vin2a_d18	1893	0	358	0	347	0	3432	0	4302	0	CFG_VIN2A_D18_OUT	vout2_d5
F11	vin2a_d19	1698	0	163	0	151	0	3237	0	4007	0	CFG_VIN2A_D19_OUT	vout2_d4
A7	vin2a_d2	2193	0	658	0	646	0	3531	0	4401	0	CFG_VIN2A_D2_OUT	vout2_d21
B13	vin2a_d20	1736	0	202	0	190	0	3075	0	3945	0	CFG_VIN2A_D20_OUT	vout2_d3
E13	vin2a_d21	1636	0	101	0	89	0	3074	0	3944	0	CFG_VIN2A_D21_OUT	vout2_d2
C13	vin2a_d22	1628	0	93	0	81	0	3266	0	4036	0	CFG_VIN2A_D22_OUT	vout2_d1
D13	vin2a_d23	1538	0	0	0	0	0	2968	0	3838	0	CFG_VIN2A_D23_OUT	vout2_d0
A9	vin2a_d3	1997	0	462	0	450	0	3335	0	4205	0	CFG_VIN2A_D3_OUT	vout2_d20
A8	vin2a_d4	2528	0	993	0	982	0	3867	0	4537	0	CFG_VIN2A_D4_OUT	vout2_d19
A11	vin2a_d5	2038	0	503	0	492	0	3577	0	4347	0	CFG_VIN2A_D5_OUT	vout2_d18
F10	vin2a_d6	1746	0	211	0	200	0	3285	0	4055	0	CFG_VIN2A_D6_OUT	vout2_d17
A10	vin2a_d7	2213	0	678	0	666	0	3552	0	4272	0	CFG_VIN2A_D7_OUT	vout2_d16
B10	vin2a_d8	2268	0	733	0	721	0	3607	0	4277	0	CFG_VIN2A_D8_OUT	vout2_d15
E10	vin2a_d9	2170	0	635	0	623	0	3509	0	4379	0	CFG_VIN2A_D9_OUT	vout2_d14
B7	vin2a_de0	2102	0	568	0	556	0	3841	0	4611	0	CFG_VIN2A_DE0_OUT	vout2_de
C7	vin2a_fld0	0	983	1398	1185	1385	1202	0	994	0	994	CFG_VIN2A_FLD0_OUT	vout2_clk
E8	vin2a_hsync0	2482	0	974	0	936	0	4021	0	4891	0	CFG_VIN2A_HSYNC0_OUT	vout2_hsync
B8	vin2a_vsync0	2296	0	784	0	750	0	3935	0	4805	0	CFG_VIN2A_VSYNC0_OUT	vout2_vsync

Manual IO Timings Modes must be used to ensure some IO timings for VOUT3. See Table 5-29, *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-46, *Manual Functions Mapping for DSS VOUT3* for a definition of the Manual modes.

Table 5-46 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-46. Manual Functions Mapping for DSS VOUT3

BALL	BALL NAME	VOUT3_MANUAL1		VOUT3_MANUAL4		VOUT3_MANUAL5		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		3
M1	gpmc_a0	2395	0	3909	0	4779	0	CFG_GPMC_A0_OUT	vout3_d16
M2	gpmc_a1	2412	0	3957	0	4827	0	CFG_GPMC_A1_OUT	vout3_d17
J2	gpmc_a10	2473	0	3980	0	4850	0	CFG_GPMC_A10_OUT	vout3_de
L3	gpmc_a11	2906	0	4253	0	5123	0	CFG_GPMC_A11_OUT	vout3_fld
L2	gpmc_a2	2360	0	3873	0	4743	0	CFG_GPMC_A2_OUT	vout3_d18
L1	gpmc_a3	2391	0	4112	0	4982	0	CFG_GPMC_A3_OUT	vout3_d19
K3	gpmc_a4	2626	0	4336	0	5206	0	CFG_GPMC_A4_OUT	vout3_d20
K2	gpmc_a5	2338	0	3840	0	4710	0	CFG_GPMC_A5_OUT	vout3_d21
J1	gpmc_a6	2374	0	3913	0	4783	0	CFG_GPMC_A6_OUT	vout3_d22
K1	gpmc_a7	2432	0	3947	0	4817	0	CFG_GPMC_A7_OUT	vout3_d23
K4	gpmc_a8	3155	0	4309	105	5179	105	CFG_GPMC_A8_OUT	vout3_hsync
H1	gpmc_a9	2309	0	3842	0	4712	0	CFG_GPMC_A9_OUT	vout3_vsync
F1	gpmc_ad0	2360	0	3652	0	4522	0	CFG_GPMC_AD0_OUT	vout3_d0
E2	gpmc_ad1	2420	0	3762	0	4632	0	CFG_GPMC_AD1_OUT	vout3_d1
A2	gpmc_ad10	2235	0	3456	0	4326	0	CFG_GPMC_AD10_OUT	vout3_d10
B3	gpmc_ad11	2253	0	3584	0	4454	0	CFG_GPMC_AD11_OUT	vout3_d11
C3	gpmc_ad12	1949	427	3589	0	4459	0	CFG_GPMC_AD12_OUT	vout3_d12
C4	gpmc_ad13	2318	0	3547	0	4417	0	CFG_GPMC_AD13_OUT	vout3_d13
A3	gpmc_ad14	2123	0	3302	0	4172	0	CFG_GPMC_AD14_OUT	vout3_d14
B4	gpmc_ad15	2195	29	3532	0	4402	0	CFG_GPMC_AD15_OUT	vout3_d15
E1	gpmc_ad2	2617	0	3859	0	4729	0	CFG_GPMC_AD2_OUT	vout3_d2
C1	gpmc_ad3	2350	0	3590	0	4460	0	CFG_GPMC_AD3_OUT	vout3_d3
D1	gpmc_ad4	2324	0	3534	0	4404	0	CFG_GPMC_AD4_OUT	vout3_d4
D2	gpmc_ad5	2371	0	3609	0	4479	0	CFG_GPMC_AD5_OUT	vout3_d5
B1	gpmc_ad6	2231	0	3416	0	4286	0	CFG_GPMC_AD6_OUT	vout3_d6
B2	gpmc_ad7	2440	0	3661	0	4531	0	CFG_GPMC_AD7_OUT	vout3_d7
C2	gpmc_ad8	2479	0	3714	0	4584	0	CFG_GPMC_AD8_OUT	vout3_d8
D3	gpmc_ad9	2355	0	3593	0	4463	0	CFG_GPMC_AD9_OUT	vout3_d9
G3	gpmc_cs3	0	641	0	905	0	905	CFG_GPMC_CS3_OUT	vout3_clk

5.10.6.5 HDMI

The High-Definition Multimedia Interface is provided for transmitting digital television audiovisual signals from DVD players, set-top boxes and other audiovisual sources to television sets, projectors and other video displays. The HDMI interface is aligned with the HDMI TMDS single stream standard v1.4a (720p @60Hz to 1080p @24Hz) and the HDMI v1.3 (1080p @60Hz): 3 data channels, plus 1 clock channel is supported (differential).

In are presented the specific groupings of signals (IOSET) for use with HDMI.

NOTE

For more information, see the High-Definition Multimedia Interface chapter of the device TRM

5.10.6.6 CSI2

NOTE

For more information, see the Camera Serial Interface 2 CAL Bridge chapter of the device TRM

The camera adaptation layer (CAL) deals with the processing of the pixel data coming from an external image sensor, data from memory. The CAL is a key component for the following multimedia applications: camera viewfinder, video record, and still image capture. The CAL has two serial camera interfaces (primary and secondary):

- The primary serial interface (CSI2 Port A) is compliant with MIPI CSI-2 protocol with four data lanes.

5.10.6.6.1 CSI-2 MIPI D-PHY

The CSI-2 port A is compliant with the MIPI D-PHY RX specification v1.00.00 and the MIPI CSI-2 specification v1.00, with 2 data differential lanes plus 1 clock differential lane in synchronous mode, double data rate:

- 1.5 Gbps (750 MHz) @OPP_NOM for each lane.

5.10.6.7 EMIF

The device has a dedicated interface to DDR3 and DDR3L SDRAM. It supports JEDEC standard compliant DDR3 and DDR3L SDRAM devices with the following features:

- 16-bit or 32-bit data path to external SDRAM memory
- Memory device capacity: 128Mb, 256Mb, 512Mb, 1Gb, 2Gb, 4Gb and 8Gb devices
- One interface with associated DDR3/DDR3L PHYs

NOTE

For more information, see the EMIF Controller section of the Device TRM.

5.10.6.8 GPMC

The GPMC is the unified memory controller that interfaces external memory devices such as:

- Asynchronous SRAM-like memories and ASIC devices
- Asynchronous page mode and synchronous burst NOR flash
- NAND flash

NOTE

For more information, see the General-Purpose Memory Controller section of the Device TRM.

5.10.6.8.1 GPMC/NOR Flash Interface Synchronous Timing

CAUTION

The I/O Timings provided in this section are valid only for some GPMC usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-47 and Table 5-48 assume testing over the recommended operating conditions and electrical characteristic conditions below (see Figure 5-23, Figure 5-24, Figure 5-25, Figure 5-26, Figure 5-27 and Figure 5-28).

Table 5-47. GPMC/NOR Flash Interface Timing Requirements - Synchronous Mode - Default

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
F12	$t_{su(dV-clkH)}$	Setup time, read gpmc_ad[15:0] valid before gpmc_clk high	3		ns
F13	$t_{h(clkH-dV)}$	Hold time, read gpmc_ad[15:0] valid after gpmc_clk high	1.1		ns
F21	$t_{su(waitV-clkH)}$	Setup time, gpmc_wait[1:0] valid before gpmc_clk high	2.5		ns
F22	$t_{h(clkH-waitV)}$	Hold Time, gpmc_wait[1:0] valid after gpmc_clk high	1.3		ns

NOTE

Wait monitoring support is limited to a WaitMonitoringTime value > 0. For a full description of wait monitoring feature, see the Device TRM.

Table 5-48. GPMC/NOR Flash Interface Switching Characteristics - Synchronous Mode - Default

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
F0	$t_{c(clk)}$	Cycle time, output clock gpmc_clk period	11.3		ns
F2	$t_{d(clkH-nCSV)}$	Delay time, gpmc_clk rising edge to gpmc_cs[7:0] transition	F-1.7 (7)	F+4.3 (7)	ns
F3	$t_{d(clkH-nCSIV)}$	Delay time, gpmc_clk rising edge to gpmc_cs[7:0] invalid	E-1.7 (6)	E+4.2 (6)	ns
F4	$t_{d(ADDV-clk)}$	Delay time, gpmc_a[27:0] address bus valid to gpmc_clk first edge	B-1.8 (3)	B+4.3 (3)	ns
F5	$t_{d(clkH-ADDIV)}$	Delay time, gpmc_clk rising edge to gpmc_a[27:0] gpmc address bus invalid	-1.8		ns
F6	$t_{d(nBEV-clk)}$	Delay time, gpmc_ben[1:0] valid to gpmc_clk rising edge	B-4.3(3)	B+1.5(3)	ns
F7	$t_{d(clkH-nBEIV)}$	Delay time, gpmc_clk rising edge to gpmc_ben[1:0] invalid	D-1.5(5)	D+4.3(5)	ns
F8	$t_{d(clkH-nADV)}$	Delay time, gpmc_clk rising edge to gpmc_advn_ale transition	G-1.3 (8)	G+4.2 (8)	ns
F9	$t_{d(clkH-nADVIV)}$	Delay time, gpmc_clk rising edge to gpmc_advn_ale invalid	D-1.3 (5)	G+4.2 (5)	ns
F10	$t_{d(clkH-nOE)}$	Delay time, gpmc_clk rising edge to gpmc_oen_ren transition	H-1.0 (9)	H+3.2 (9)	ns
F11	$t_{d(clkH-nOEIV)}$	Delay time, gpmc_clk rising edge to gpmc_oen_ren invalid	E-1.0 (6)	E+3.2 (6)	ns
F14	$t_{d(clkH-nWE)}$	Delay time, gpmc_clk rising edge to gpmc_wen transition	I-0.9 (10)	I+4.2 (10)	ns
F15	$t_{d(clkH-Data)}$	Delay time, gpmc_clk rising edge to gpmc_ad[15:0] data bus transition	J-2.1 (11)	J+4.6 (11)	ns
F17	$t_{d(clkH-nBE)}$	Delay time, gpmc_clk rising edge to gpmc_ben[1:0] transition	J-1.5 (11)	J+4.3 (11)	ns
F18	$t_{w(nCSV)}$	Pulse duration, gpmc_cs[7:0] low	A (2)		ns
F19	$t_{w(nBEV)}$	Pulse duration, gpmc_ben[1:0] low	C (4)		ns
F20	$t_{w(nADVV)}$	Pulse duration, gpmc_advn_ale low	K (12)		ns
F23	$t_{d(CLK-GPIO)}$	Delay time, gpmc_clk transition to gpio6_16 transition	0.5	7.5	ns

Table 5-49. GPMC/NOR Flash Interface Timing Requirements - Synchronous Mode - Alternate

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
F12	$t_{su(dV-clkH)}$	Setup time, read gpmc_ad[15:0] valid before gpmc_clk high	2.5		ns
F13	$t_{h(clkH-dV)}$	Hold time, read gpmc_ad[15:0] valid after gpmc_clk high	1.9		ns
F21	$t_{su(waitV-clkH)}$	Setup time, gpmc_wait[1:0] valid before gpmc_clk high	2.5		ns
F22	$t_{h(clkH-waitV)}$	Hold Time, gpmc_wait[1:0] valid after gpmc_clk high	1.9		ns

Table 5-50. GPMC/NOR Flash Interface Switching Characteristics - Synchronous Mode - Alternate

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
F0	$t_{c(clk)}$	Cycle time, output clock gpmc_clk period (13)	15.04		ns
F2	$t_{d(clkH-nCSV)}$	Delay time, gpmc_clk rising edge to gpmc_cs[7:0] transition	F+0.6 (7)	F+7.0 (7)	ns
F3	$t_{d(clkH-nCSIV)}$	Delay time, gpmc_clk rising edge to gpmc_cs[7:0] invalid	E+0.6 (6)	E+7.0 (6)	ns
F4	$t_{d(ADDV-clk)}$	Delay time, gpmc_a[27:0] address bus valid to gpmc_clk first edge	B-0.7 (3)	B+7.0 (3)	ns
F5	$t_{d(clkH-ADDIV)}$	Delay time, gpmc_clk rising edge to gpmc_a[27:0] gpmc address bus invalid	-0.7		ns
F6	$t_{d(nBEV-clk)}$	Delay time, gpmc_ben[1:0] valid to gpmc_clk rising edge	B-7.0	B+0.4	ns
F7	$t_{d(clkH-nBEIV)}$	Delay time, gpmc_clk rising edge to gpmc_ben[1:0] invalid	D-0.4	D+7.0	ns
F8	$t_{d(clkH-nADV)}$	Delay time, gpmc_clk rising edge to gpmc_advn_ale transition	G+0.7 (8)	G+6.1 (8)	ns
F9	$t_{d(clkH-nADVIV)}$	Delay time, gpmc_clk rising edge to gpmc_advn_ale invalid	D+0.7 (5)	D+6.1 (5)	ns
F10	$t_{d(clkH-nOE)}$	Delay time, gpmc_clk rising edge to gpmc_oen_ren transition	H+0.7 (9)	H+5.1 (9)	ns
F11	$t_{d(clkH-nOEIV)}$	Delay time, gpmc_clk rising edge to gpmc_oen_ren invalid	E+0.7 (6)	E+5.1 (6)	ns
F14	$t_{d(clkH-nWE)}$	Delay time, gpmc_clk rising edge to gpmc_wen transition	I+0.7 (10)	I+6.1 (10)	ns
F15	$t_{d(clkH-Data)}$	Delay time, gpmc_clk rising edge to gpmc_ad[15:0] data bus transition	J-0.4 (11)	J+4.9 (11)	ns
F17	$t_{d(clkH-nBE)}$	Delay time, gpmc_clk rising edge to gpmc_ben[1:0] transition	J-0.4 (11)	J+4.9 (11)	ns
F18	$t_{w(nCSV)}$	Pulse duration, gpmc_cs[7:0] low	A (2)		ns
F19	$t_{w(nBEV)}$	Pulse duration, gpmc_ben[1:0] low	C (4)		ns
F20	$t_{w(nADVV)}$	Pulse duration, gpmc_advn_ale low	K (12)		ns
F23	$t_{d(CLK-GPIO)}$	Delay time, gpmc_clk transition to gpio6_16.clkout1 transition (14)	0.5	7.5	ns

(1) Total GPMC load on any signal at 3.3V must not exceed 10pF.

(2) For single read: A = (CSRdOffTime - CSOnTime) × (TimeParaGranularity + 1) × GPMC_FCLK period
For burst read: A = (CSRdOffTime - CSOnTime + (n - 1) × PageBurstAccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK period
For burst write: A = (CSWrOffTime - CSOnTime + (n - 1) × PageBurstAccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK period
with n the page burst access number.

(3) B = ClkActivationTime × GPMC_FCLK

(4) For single read: C = RdCycleTime × (TimeParaGranularity + 1) × GPMC_FCLK
For burst read: C = (RdCycleTime + (n - 1) × PageBurstAccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For Burst write: C = (WrCycleTime + (n - 1) × PageBurstAccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK with n the page burst access number.

(5) For single read: D = (RdCycleTime - AccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For burst read: D = (RdCycleTime - AccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For burst write: D = (WrCycleTime - AccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK

(6) For single read: E = (CSRdOffTime - AccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For burst read: E = (CSRdOffTime - AccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For burst write: E = (CSWrOffTime - AccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK

(7) For nCS falling edge (CS activated):
Case GpmcFCLKDivider = 0 :
F = 0.5 × CSExtraDelay × GPMC_FCLK Case GpmcFCLKDivider = 1:
F = 0.5 × CSExtraDelay × GPMC_FCLK if (ClkActivationTime and CSOnTime are odd) or (ClkActivationTime and CSOnTime are even)
F = (1 + 0.5 × CSExtraDelay) × GPMC_FCLK otherwise
Case GpmcFCLKDivider = 2:
F = 0.5 × CSExtraDelay × GPMC_FCLK if ((CSOnTime - ClkActivationTime) is a multiple of 3)
F = (1 + 0.5 × CSExtraDelay) × GPMC_FCLK if ((CSOnTime - ClkActivationTime - 1) is a multiple of 3)
F = (2 + 0.5 × CSExtraDelay) × GPMC_FCLK if ((CSOnTime - ClkActivationTime - 2) is a multiple of 3)
Case GpmcFCLKDivider = 3:
F = 0.5 × CSExtraDelay × GPMC_FCLK if ((CSOnTime - ClkActivationTime) is a multiple of 4)

F = (1 + 0.5 × CSExtraDelay) × GPMC_FCLK if ((CSOnTime - ClkActivationTime - 1) is a multiple of 4)
F = (2 + 0.5 × CSExtraDelay) × GPMC_FCLK if ((CSOnTime - ClkActivationTime - 2) is a multiple of 4)
F = (3 + 0.5 × CSExtraDelay) × GPMC_FCLK if ((CSOnTime - ClkActivationTime - 3) is a multiple of 4)

(8) For ADV falling edge (ADV activated):
Case GpmcFCLKDivider = 0 :
G = 0.5 × ADVExtraDelay × GPMC_FCLK
Case GpmcFCLKDivider = 1:
G = 0.5 × ADVExtraDelay × GPMC_FCLK if (ClkActivationTime and ADVOnTime are odd) or (ClkActivationTime and ADVOnTime are even)
G = (1 + 0.5 × ADVExtraDelay) × GPMC_FCLK otherwise
Case GpmcFCLKDivider = 2:
G = 0.5 × ADVExtraDelay × GPMC_FCLK if ((ADVOnTime - ClkActivationTime) is a multiple of 3)
G = (1 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVOnTime - ClkActivationTime - 1) is a multiple of 3)
G = (2 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVOnTime - ClkActivationTime - 2) is a multiple of 3)
For ADV rising edge (ADV desactivated) in Reading mode:
Case GpmcFCLKDivider = 0:
G = 0.5 × ADVExtraDelay × GPMC_FCLK
Case GpmcFCLKDivider = 1:
G = 0.5 × ADVExtraDelay × GPMC_FCLK if (ClkActivationTime and ADVRdOffTime are odd) or (ClkActivationTime and ADVRdOffTime are even)
G = (1 + 0.5 × ADVExtraDelay) × GPMC_FCLK otherwise
Case GpmcFCLKDivider = 2:
G = 0.5 × ADVExtraDelay × GPMC_FCLK if ((ADVRdOffTime - ClkActivationTime) is a multiple of 3)
G = (1 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVRdOffTime - ClkActivationTime - 1) is a multiple of 3)
G = (2 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVRdOffTime - ClkActivationTime - 2) is a multiple of 3)
Case GpmcFCLKDivider = 3:
G = 0.5 × ADVExtraDelay × GPMC_FCLK if ((ADVRdOffTime - ClkActivationTime) is a multiple of 4)
G = (1 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVRdOffTime - ClkActivationTime - 1) is a multiple of 4)
G = (2 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVRdOffTime - ClkActivationTime - 2) is a multiple of 4)
G = (3 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVRdOffTime - ClkActivationTime - 3) is a multiple of 4)
For ADV rising edge (ADV desactivated) in Writing mode:
Case GpmcFCLKDivider = 0:
G = 0.5 × ADVExtraDelay × GPMC_FCLK
Case GpmcFCLKDivider = 1:
G = 0.5 × ADVExtraDelay × GPMC_FCLK if (ClkActivationTime and ADVWrOffTime are odd) or (ClkActivationTime and ADVWrOffTime are even)
G = (1 + 0.5 × ADVExtraDelay) × GPMC_FCLK otherwise
Case GpmcFCLKDivider = 2:
G = 0.5 × ADVExtraDelay × GPMC_FCLK if ((ADVWrOffTime - ClkActivationTime) is a multiple of 3)
G = (1 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVWrOffTime - ClkActivationTime - 1) is a multiple of 3)
G = (2 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVWrOffTime - ClkActivationTime - 2) is a multiple of 3)
Case GpmcFCLKDivider = 3:
G = 0.5 × ADVExtraDelay × GPMC_FCLK if ((ADVWrOffTime - ClkActivationTime) is a multiple of 4)
G = (1 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVWrOffTime - ClkActivationTime - 1) is a multiple of 4)
G = (2 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVWrOffTime - ClkActivationTime - 2) is a multiple of 4)
G = (3 + 0.5 × ADVExtraDelay) × GPMC_FCLK if ((ADVWrOffTime - ClkActivationTime - 3) is a multiple of 4)

(9) For OE falling edge (OE activated):
Case GpmcFCLKDivider = 0:
- H = 0.5 × OEExtraDelay × GPMC_FCLK
Case GpmcFCLKDivider = 1:
- H = 0.5 × OEExtraDelay × GPMC_FCLK if (ClkActivationTime and OEOnTime are odd) or (ClkActivationTime and OEOnTime are even)
- H = (1 + 0.5 × OEExtraDelay) × GPMC_FCLK otherwise
Case GpmcFCLKDivider = 2:
- H = 0.5 × OEExtraDelay × GPMC_FCLK if ((OEOnTime - ClkActivationTime) is a multiple of 3)
- H = (1 + 0.5 × OEExtraDelay) × GPMC_FCLK if ((OEOnTime - ClkActivationTime - 1) is a multiple of 3)
- H = (2 + 0.5 × OEExtraDelay) × GPMC_FCLK if ((OEOnTime - ClkActivationTime - 2) is a multiple of 3)
Case GpmcFCLKDivider = 3:
- H = 0.5 × OEExtraDelay × GPMC_FCLK if ((OEOnTime - ClkActivationTime) is a multiple of 4)
- H = (1 + 0.5 × OEExtraDelay) × GPMC_FCLK if ((OEOnTime - ClkActivationTime - 1) is a multiple of 4)
- H = (2 + 0.5 × OEExtraDelay) × GPMC_FCLK if ((OEOnTime - ClkActivationTime - 2) is a multiple of 4)
- H = (3 + 0.5 × OEExtraDelay)) × GPMC_FCLK if ((OEOnTime - ClkActivationTime - 3) is a multiple of 4)
For OE rising edge (OE desactivated):
Case GpmcFCLKDivider = 0:
- H = 0.5 × OEExtraDelay × GPMC_FCLK
Case GpmcFCLKDivider = 1:
- H = 0.5 × OEExtraDelay × GPMC_FCLK if (ClkActivationTime and OEOffTime are odd) or (ClkActivationTime and OEOffTime are even)
- H = (1 + 0.5 × OEExtraDelay) × GPMC_FCLK otherwise
Case GpmcFCLKDivider = 2:
- H = 0.5 × OEExtraDelay × GPMC_FCLK if ((OEOffTime - ClkActivationTime) is a multiple of 3)
- H = (1 + 0.5 × OEExtraDelay) × GPMC_FCLK if ((OEOffTime - ClkActivationTime - 1) is a multiple of 3)

- H = (2 + 0.5 × OEExtraDelay) × GPMC_FCLK if ((OEOffTime - ClkActivationTime - 2) is a multiple of 3)
Case GpmcFCLKDivider = 3:
- H = 0.5 × OEExtraDelay × GPMC_FCLK if ((OEOffTime - ClkActivationTime) is a multiple of 4)
- H = (1 + 0.5 × OEExtraDelay) × GPMC_FCLK if ((OEOffTime - ClkActivationTime - 1) is a multiple of 4)
- H = (2 + 0.5 × OEExtraDelay) × GPMC_FCLK if ((OEOffTime - ClkActivationTime - 2) is a multiple of 4)
- H = (3 + 0.5 × OEExtraDelay) × GPMC_FCLK if ((OEOffTime - ClkActivationTime - 3) is a multiple of 4)

(10) For WE falling edge (WE activated):
Case GpmcFCLKDivider = 0:
- I = 0.5 × WEExtraDelay × GPMC_FCLK
Case GpmcFCLKDivider = 1:
- I = 0.5 × WEExtraDelay × GPMC_FCLK if (ClkActivationTime and WEOnTime are odd) or (ClkActivationTime and WEOnTime are even)
- I = (1 + 0.5 × WEExtraDelay) × GPMC_FCLK otherwise
Case GpmcFCLKDivider = 2:
- I = 0.5 × WEExtraDelay × GPMC_FCLK if ((WEOnTime - ClkActivationTime) is a multiple of 3)
- I = (1 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOnTime - ClkActivationTime - 1) is a multiple of 3)
- I = (2 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOnTime - ClkActivationTime - 2) is a multiple of 3)
Case GpmcFCLKDivider = 3:
- I = 0.5 × WEExtraDelay × GPMC_FCLK if ((WEOnTime - ClkActivationTime) is a multiple of 4)
- I = (1 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOnTime - ClkActivationTime - 1) is a multiple of 4)
- I = (2 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOnTime - ClkActivationTime - 2) is a multiple of 4)
- I = (3 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOnTime - ClkActivationTime - 3) is a multiple of 4)
For WE rising edge (WE desactivated):
Case GpmcFCLKDivider = 0:
- I = 0.5 × WEExtraDelay × GPMC_FCLK
Case GpmcFCLKDivider = 1:
- I = 0.5 × WEExtraDelay × GPMC_FCLK if (ClkActivationTime and WEOffTime are odd) or (ClkActivationTime and WEOffTime are even)
- I = (1 + 0.5 × WEExtraDelay) × GPMC_FCLK otherwise
Case GpmcFCLKDivider = 2:
- I = 0.5 × WEExtraDelay × GPMC_FCLK if ((WEOffTime - ClkActivationTime) is a multiple of 3)
- I = (1 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOffTime - ClkActivationTime - 1) is a multiple of 3)
- I = (2 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOffTime - ClkActivationTime - 2) is a multiple of 3)
Case GpmcFCLKDivider = 3:
- I = 0.5 × WEExtraDelay × GPMC_FCLK if ((WEOffTime - ClkActivationTime) is a multiple of 4)
- I = (1 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOffTime - ClkActivationTime - 1) is a multiple of 4)
- I = (2 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOffTime - ClkActivationTime - 2) is a multiple of 4)
- I = (3 + 0.5 × WEExtraDelay) × GPMC_FCLK if ((WEOffTime - ClkActivationTime - 3) is a multiple of 4)

(11) J = GPMC_FCLK period, where GPMC_FCLK is the General Purpose Memory Controller internal functional clock

(12) For read:
K = (ADVRdOffTime - ADVOnTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For write: K = (ADVWrOffTime - ADVOnTime) × (TimeParaGranularity + 1) × GPMC_FCLK

(13) The gpmc_clk output clock maximum and minimum frequency is programmable in the I/F module by setting the GPMC_CONFIG1_CSx configuration register bit fields GpmcFCLKDivider

(14) gpio6_16 programmed to MUXMODE=9 (clkout1), CM_CLKSEL_CLKOUTMUX1 programmed to 7 (CORE_DPLL_OUT_DCLK), CM_CLKSEL_CORE_DPLL_OUT_CLK_CLKOUTMUX programmed to 1.

(15) CSEXTRADELAY = 0, ADVEXTRADELAY = 0, WEEXTRADELAY = 0, OEEXTRADELAY = 0. Extra half-GPMC_FCLK cycle delay mode is not timed.

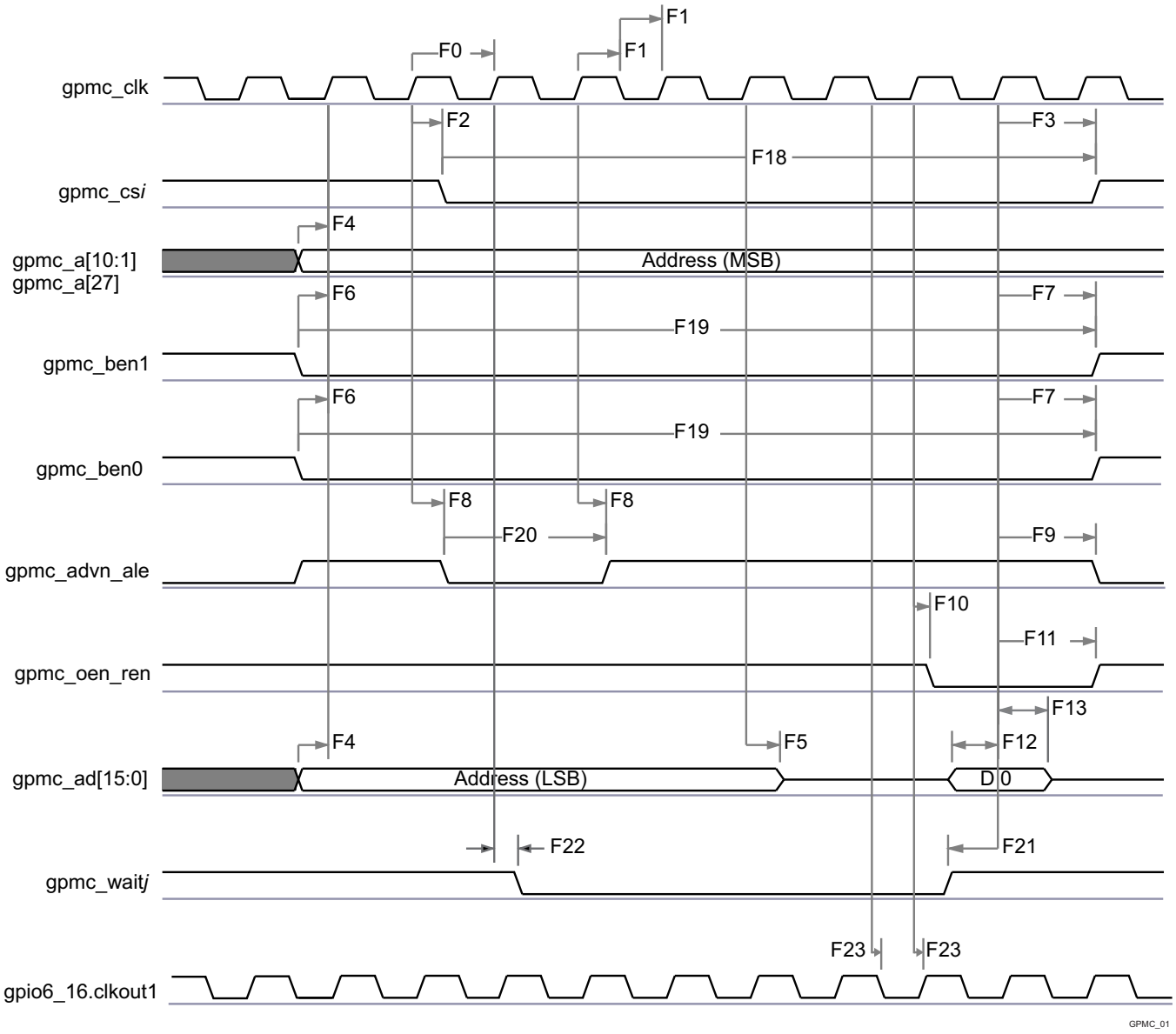


Figure 5-23. GPMC / Multiplexed 16bits NOR Flash - Synchronous Single Read - (GpmcFCLKDivider = 0)(1)(2)

(1) In gpmc_cs*i*, i = 0 to 7.

(2) In gpmc_waitj, j = 0 to 1.

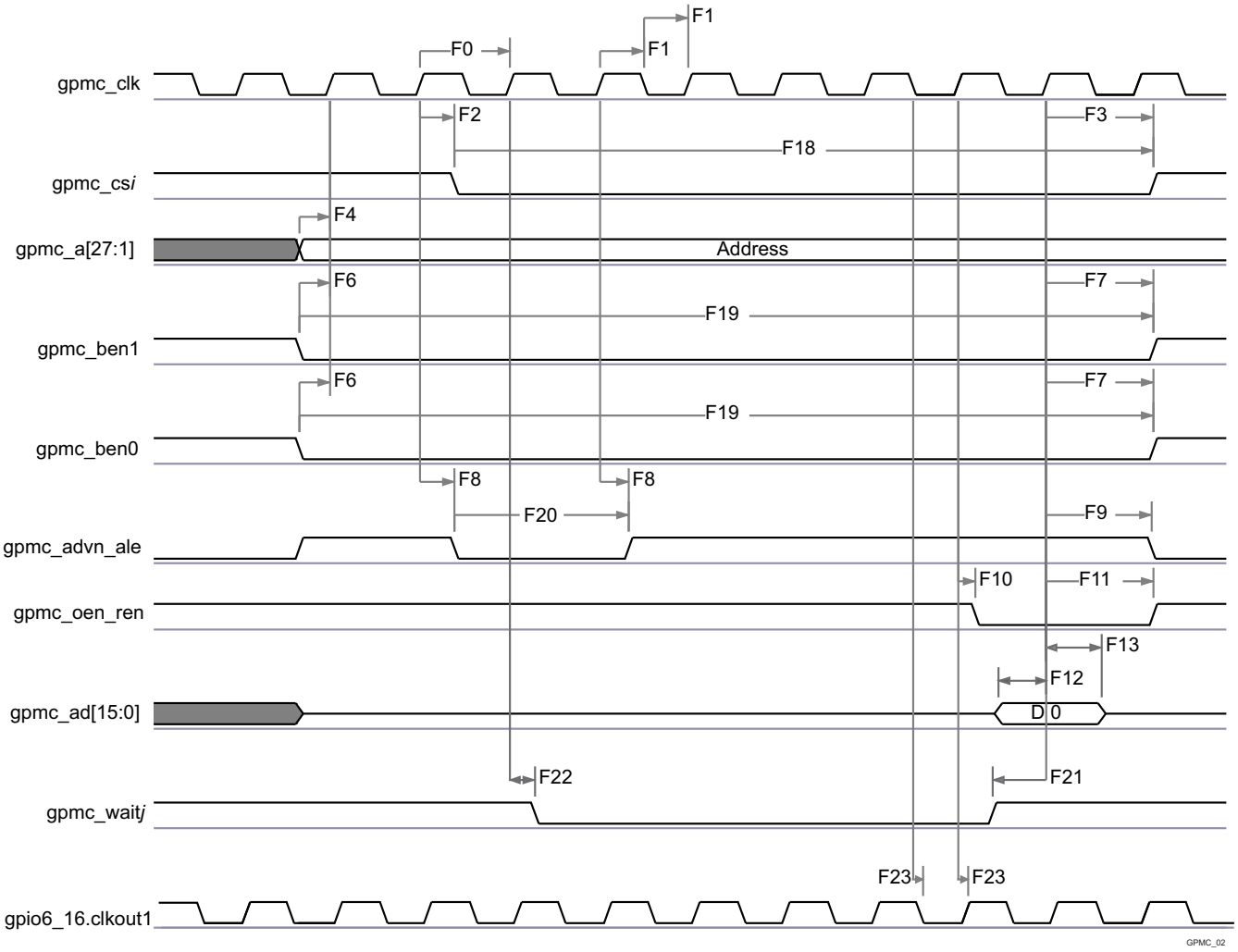


Figure 5-24. GPMC / Nonmultiplexed 16bits NOR Flash - Synchronous Single Read - (GpmcFCLKDivider = 0)[(1)(2)]

(1) In gpmc_cs*i*, i = 0 to 7.

(2) In gpmc_waitj, j = 0 to 1.

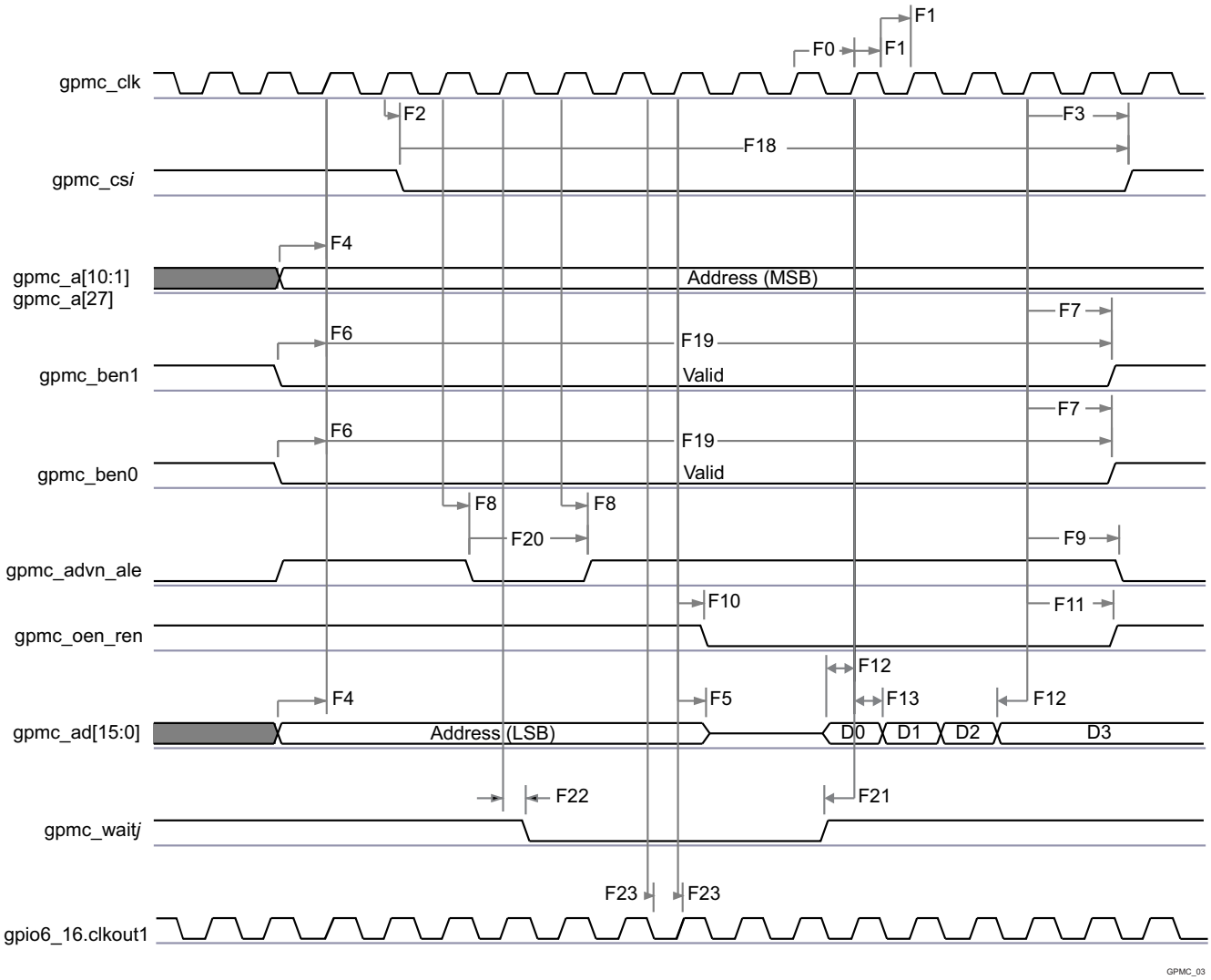


Figure 5-25. GPMC / Multiplexed 16bits NOR Flash - Synchronous Burst Read 4x16 bits - (GpmcFCLKDivider = 0)(1)(2)

(1) In gpmc_cs*i*, i= 0 to 7.

(2) In gpmc_waitj, j = 0 to 1.

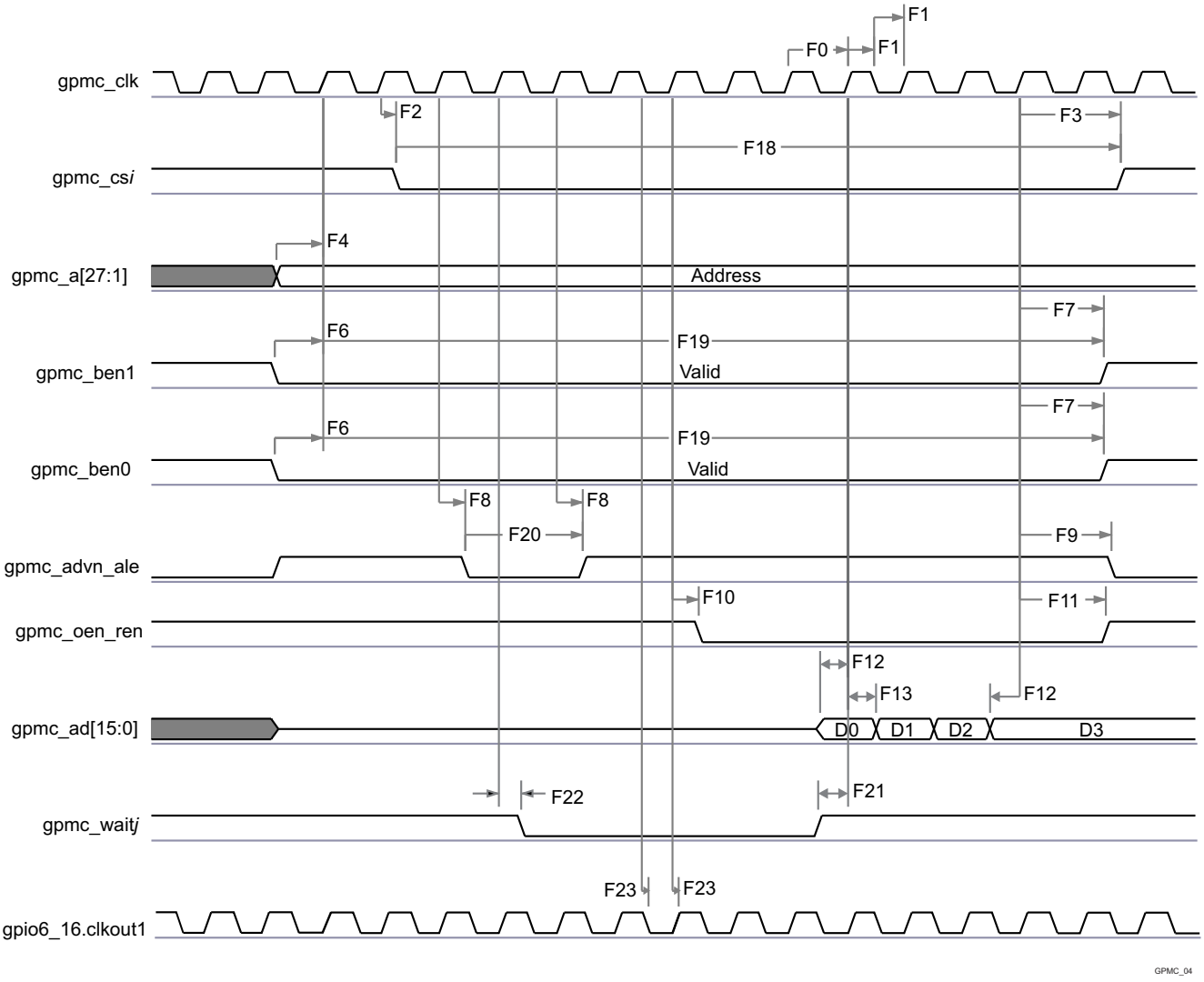


Figure 5-26. GPMC / Nonmultiplexed 16bits NOR Flash - Synchronous Burst Read 4x16 bits - (GpmcFCLKDivider = 0)[(1)][(2)]

(1) In gpmc_cs*i*, i = 0 to 7.

(2) In gpmc_waitj, j = 0 to 1.

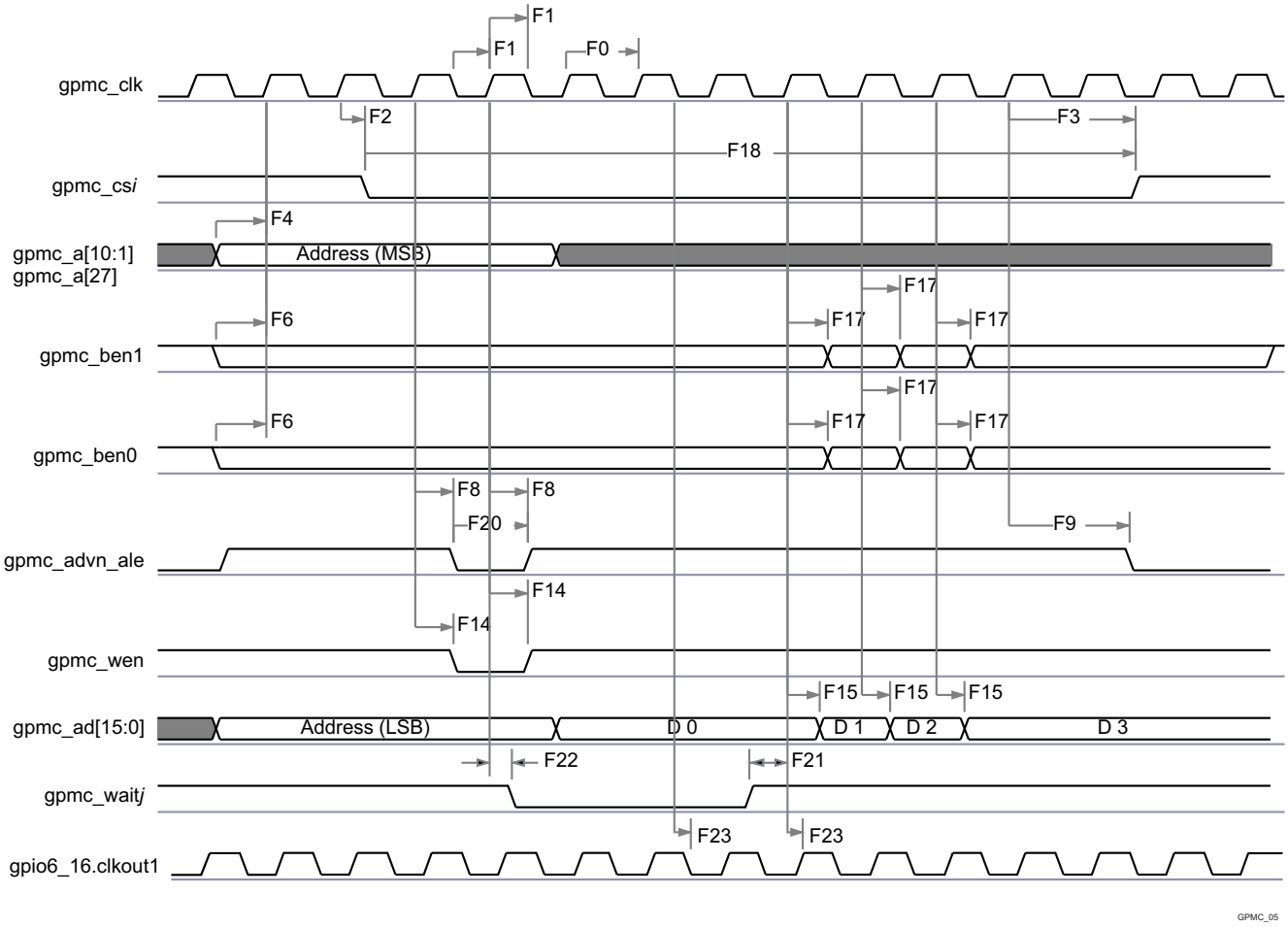


Figure 5-27. GPMC / Multiplexed 16bits NOR Flash - Synchronous Burst Write 4x16bits - (GpmcFCLKDivider = 0)(1)(2)

(1) In “gpmc_csi”, i = 0 to 7.

(2) In “gpmc_waitj”, j = 0 to 1.

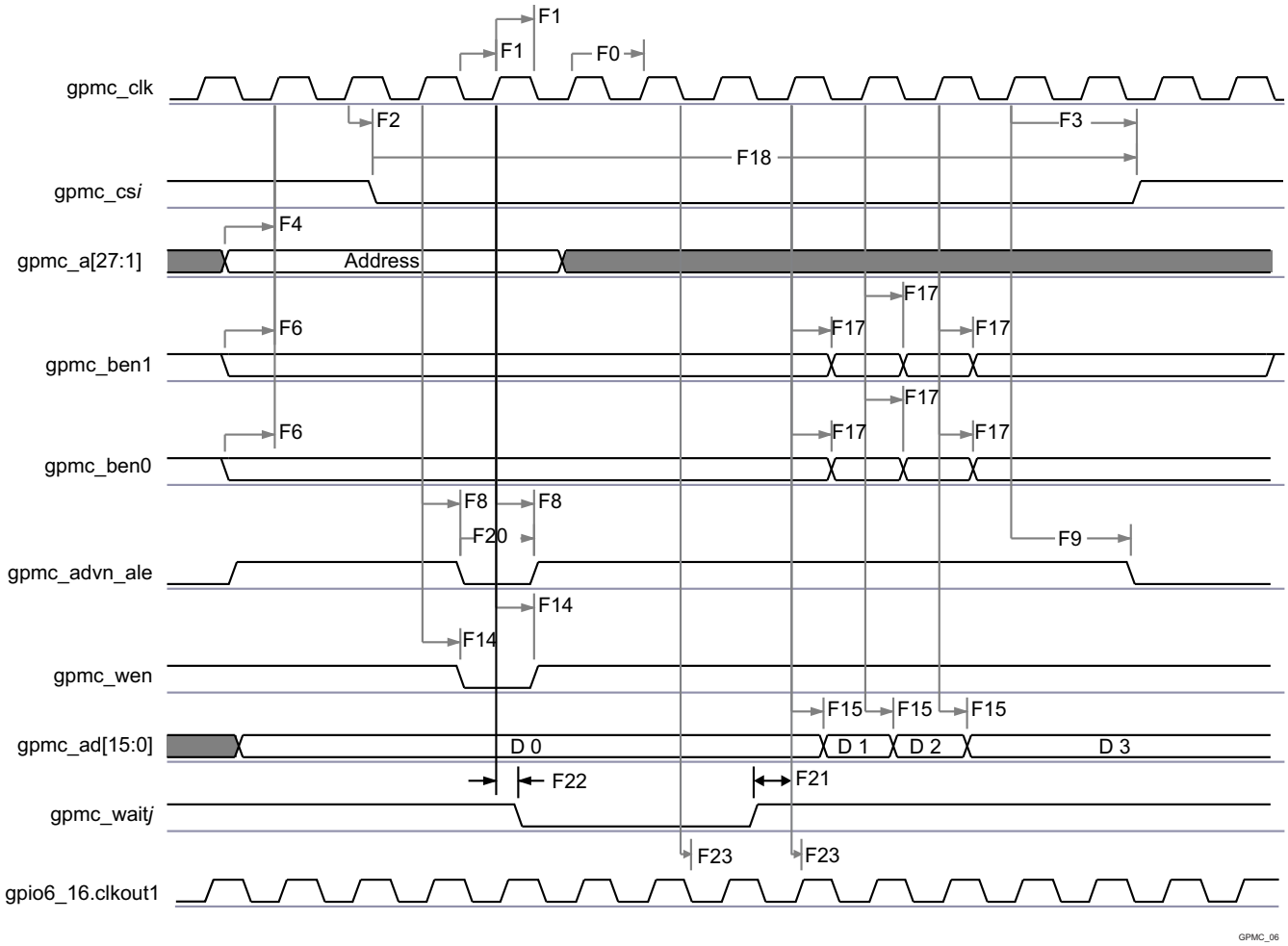


Figure 5-28. GPMC / Nonmultiplexed 16bits NOR Flash - Synchronous Burst Write 4x16bits - (GpmcFCLKDivider = 0)(1)(2)

(1) In "gpmc_csi", i = 1 to 7.

(2) In "gpmc_waitj", j = 0 to 1.

5.10.6.8.2 GPMC/NOR Flash Interface Asynchronous Timing

CAUTION

The I/O Timings provided in this section are valid only for some GPMC usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-51 and Table 5-52 assume testing over the recommended operating conditions and electrical characteristic conditions below (see Figure 5-29, Figure 5-30, Figure 5-31, Figure 5-32, Figure 5-33 and Figure 5-34).

Table 5-51. GPMC/NOR Flash Interface Timing Requirements - Asynchronous Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
FA5	$t_{acc(DAT)}$	Data Maximum Access Time (GPMC_FCLK cycles)		H (1)	cycles

Table 5-51. GPMC/NOR Flash Interface Timing Requirements - Asynchronous Mode (continued)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
FA20	$t_{acc1\text{-}pgmode(DAT)}$	Page Mode Successive Data Maximum Access Time (GPMC_FCLK cycles)		P (2)	cycles
FA21	$t_{acc2\text{-}pgmode(DAT)}$	Page Mode First Data Maximum Access Time (GPMC_FCLK cycles)		H (1)	cycles
-	$t_{su(DV\text{-}OEH)}$	Setup time, read gpmc_ad[15:0] valid before gpmc_oen_ren high	1.9		ns
-	$t_{h(OEH\text{-}DV)}$	Hold time, read gpmc_ad[15:0] valid after gpmc_oen_ren high	1		ns

(1) H = Access Time × (TimeParaGranularity + 1)

(2) P = PageBurstAccessTime × (TimeParaGranularity + 1)

Table 5-52. GPMC/NOR Flash Interface Switching Characteristics - Asynchronous Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
-	$t_{r(DO)}$	Rising time, gpmc_ad[15:0] output data	0.447	4.067	ns
-	$t_{f(DO)}$	Fallling time, gpmc_ad[15:0] output data	0.43	4.463	ns
FA0	$t_{w(nBEV)}$	Pulse duration, gpmc_ben[1:0] valid time		N (1)	ns
FA1	$t_{w(nCSV)}$	Pulse duration, gpmc_cs[7:0] low		A (2)	ns
FA3	$t_{d(nCSV\text{-}nADVIV)}$	Delay time, gpmc_cs[7:0] valid to gpmc_advn_ale invalid	B - 2 (3)	B + 4 (3)	ns
FA4	$t_{d(nCSV\text{-}nOEIV)}$	Delay time, gpmc_cs[7:0] valid to gpmc_oen_ren invalid (Single read)	C - 2 (4)	C + 4 (4)	ns
FA9	$t_{d(AV\text{-}nCSV)}$	Delay time, address bus valid to gpmc_cs[7:0] valid	J - 2 (5)	J + 4 (5)	ns
FA10	$t_{d(nBEV\text{-}nCSV)}$	Delay time, gpmc_ben[1:0] valid to gpmc_cs[7:0] valid	J - 2 (5)	J + 4 (5)	ns
FA12	$t_{d(nCSV\text{-}nADVV)}$	Delay time, gpmc_cs[7:0] valid to gpmc_advn_ale valid	K - 2 (6)	K + 4 (6)	ns
FA13	$t_{d(nCSV\text{-}nOEV)}$	Delay time, gpmc_cs[7:0] valid to gpmc_oen_ren valid	L - 2 (7)	L + 4 (7)	ns
FA16	$t_{w(AIV)}$	Pulse duration, address invalid between 2 successive R/W accesses	G (8)		ns
FA18	$t_{d(nCSV\text{-}nOEIV)}$	Delay time, gpmc_cs[7:0] valid to gpmc_oen_ren invalid (Burst read)	I - 2 (9)	I + 4 (9)	ns
FA20	$t_{w(AV)}$	Pulse duration, address valid : 2nd, 3rd and 4th accesses	D (10)		ns
FA25	$t_{d(nCSV\text{-}nWEV)}$	Delay time, gpmc_cs[7:0] valid to gpmc_wen valid	E - 2 (11)	E + 4 (11)	ns
FA27	$t_{d(nCSV\text{-}nWEIV)}$	Delay time, gpmc_cs[7:0] valid to gpmc_wen invalid	F - 2 (12)	F + 4 (12)	ns
FA28	$t_{d(nWEV\text{-}DV)}$	Delay time, gpmc_ wen valid to data bus valid		2	ns
FA29	$t_{d(DV\text{-}nCSV)}$	Delay time, data bus valid to gpmc_cs[7:0] valid	J - 2 (5)	J + 4 (5)	ns
FA37	$t_{d(nOEV\text{-}AIV)}$	Delay time, gpmc_oen_ren valid to gpmc_ad[15:0] multiplexed address bus phase end		2	ns

(1) For single read: N = RdCycleTime × (TimeParaGranularity + 1) × GPMC_FCLK
For single write: N = WrCycleTime × (TimeParaGranularity + 1) × GPMC_FCLK
For burst read: N = (RdCycleTime + (n - 1) × PageBurstAccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For burst write: N = (WrCycleTime + (n - 1) × PageBurstAccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK

(2) For single read: A = (CSRdOffTime - CSOnTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For single write: A = (CSWrOffTime - CSOnTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For burst read: A = (CSRdOffTime - CSOnTime + (n - 1) × PageBurstAccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK
For burst write: A = (CSWrOffTime - CSOnTime + (n - 1) × PageBurstAccessTime) × (TimeParaGranularity + 1) × GPMC_FCLK

(3) For reading: B = ((ADVRdOffTime - CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (ADVExtraDelay - CSExtraDelay)) × GPMC_FCLK
For writing: B = ((ADVWrOffTime - CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (ADVExtraDelay - CSExtraDelay)) × GPMC_FCLK

(4) C = ((OEOffTime - CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (OEExtraDelay - CSExtraDelay)) × GPMC_FCLK

(5) J = (CSOnTime × (TimeParaGranularity + 1) + 0.5 × CSExtraDelay) × GPMC_FCLK

(6) K = ((ADVOnTime - CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (ADVExtraDelay - CSExtraDelay)) × GPMC_FCLK

(7) L = ((OEOnTime - CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (OEExtraDelay - CSExtraDelay)) × GPMC_FCLK

(8) G = Cycle2CycleDelay × GPMC_FCLK × (TimeParaGranularity +1)

(9) I = ((OEOffTime + (n - 1) × PageBurstAccessTime - CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (OEExtraDelay - CSExtraDelay)) × GPMC_FCLK

(10) D = PageBurstAccessTime × (TimeParaGranularity + 1) × GPMC_FCLK

(11) E = ((WEOnTime - CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (WEExtraDelay - CSExtraDelay)) × GPMC_FCLK

(12) F = ((WEOffTime - CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (WEExtraDelay - CSExtraDelay)) × GPMC_FCLK

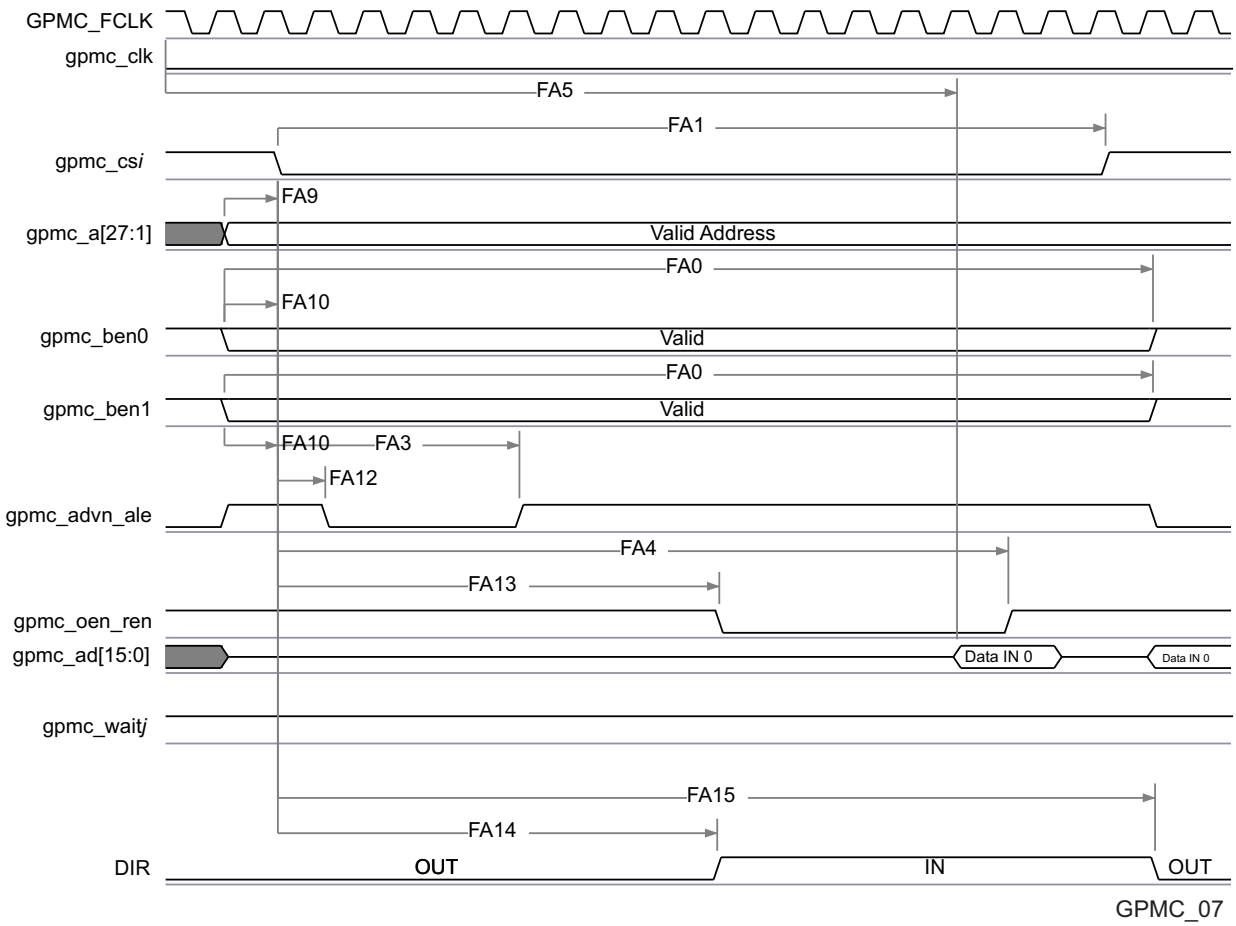


Figure 5-29. GPMC / NOR Flash - Asynchronous Read - Single Word Timing(1)(2)(3)

(1) In gpmc_cs*i*, i = 0 to 7. In gpmc_waitj, j = 0 to 1.

(2) FA5 parameter illustrates amount of time required to internally sample input data. It is expressed in number of GPMC functional clock cycles. From start of read cycle and after FA5 functional clock cycles, input Data will be internally sampled by active functional clock edge. FA5 value must be stored inside AccessTime register bits field.

(3) GPMC_FCLK is an internal clock (GPMC functional clock) not provided externally.

(4) The "DIR" (direction control) output signal is NOT pinned out on any of the device pads. It is an internal signal only representing a signal direction on the GPMC data bus.

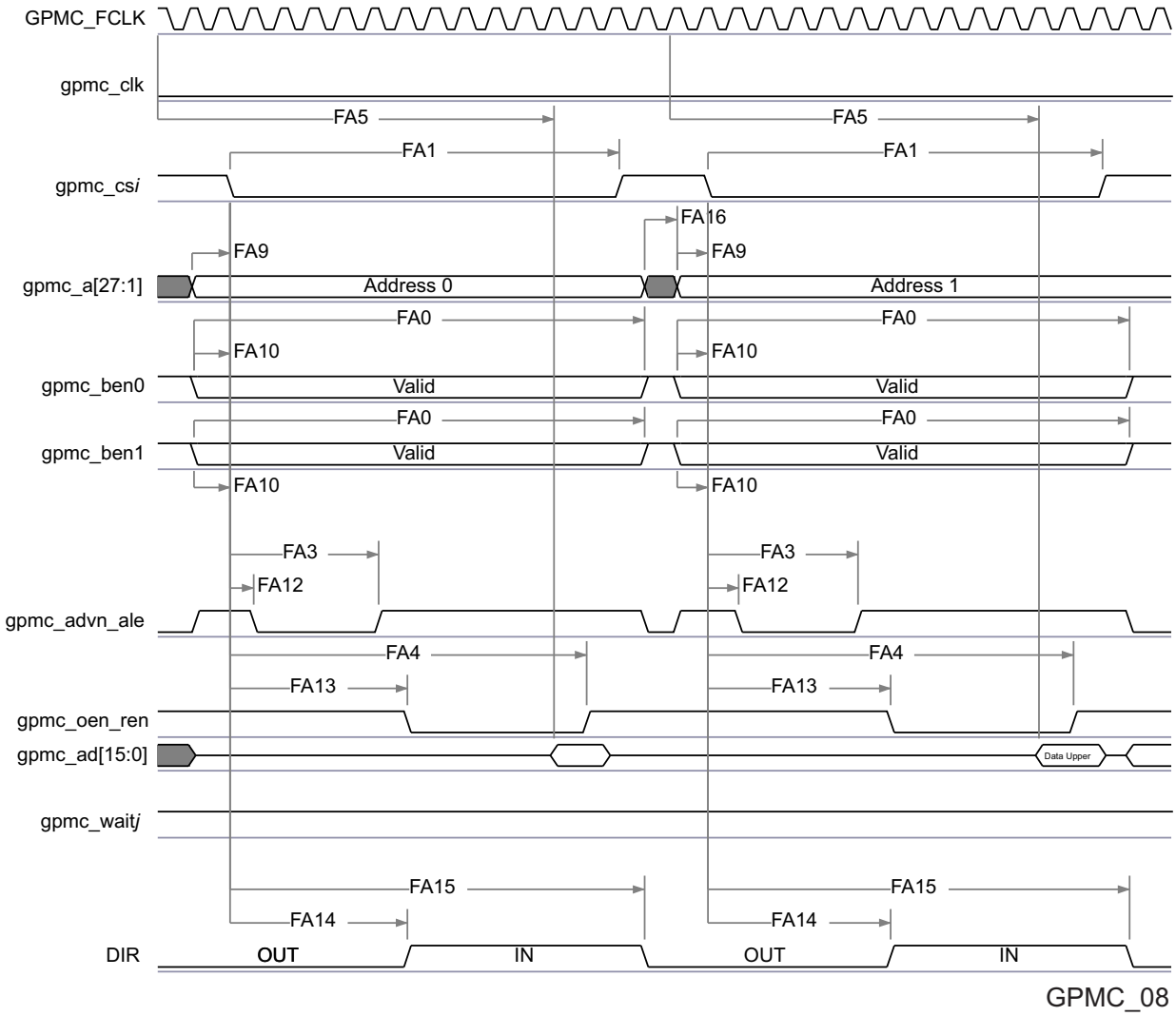


Figure 5-30. GPMC / NOR Flash - Asynchronous Read - 32-bit Timing(1)(2)(3)

(1) In “gpmc_csi”, i = 0 to 7. In “gpmc_waitj”, j = 0 to 1.

(2) FA5 parameter illustrates amount of time required to internally sample input Data. It is expressed in number of GPMC functional clock cycles. From start of read cycle and after FA5 functional clock cycles, input Data will be internally sampled by active functional clock edge. FA5 value should be stored inside AccessTime register bits field

(3) GPMC_FCLK is an internal clock (GPMC functional clock) not provided externally

(4) The "DIR" (direction control) output signal is NOT pinned out on any of the device pads. It is an internal signal only representing a signal direction on the GPMC data bus.

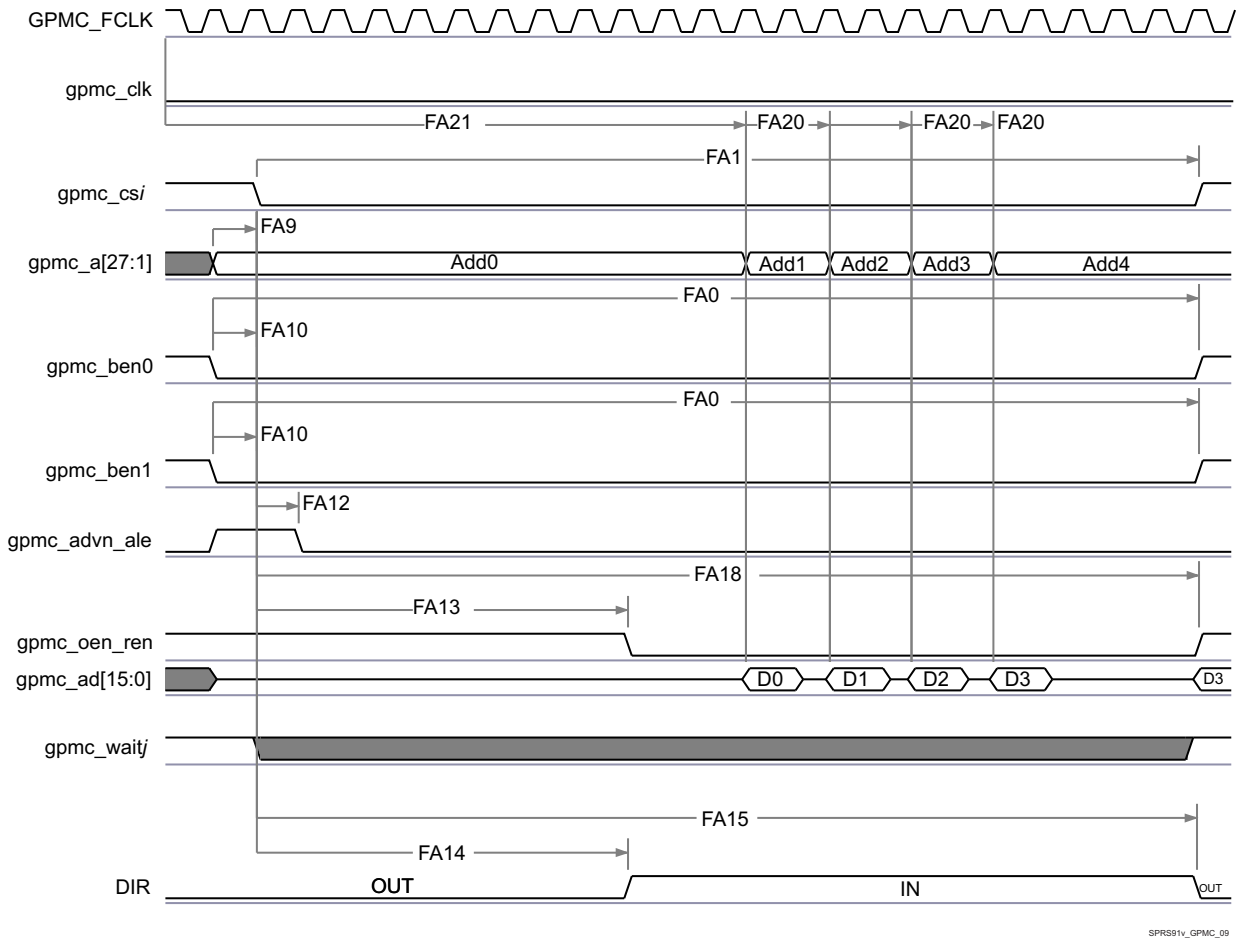


Figure 5-31. GPMC / NOR Flash - Asynchronous Read - Page Mode 4x16-bit Timing[(1)(2)(3)(4)]

(1) In “gpmc_csi”, i = 0 to 7. In “gpmc_waitj”, j = 0 to 1

(2) FA21 parameter illustrates amount of time required to internally sample first input Page Data. It is expressed in number of GPMC functional clock cycles. From start of read cycle and after FA21 functional clock cycles, First input Page Data will be internally sampled by active functional clock edge. FA21 calculation is detailled in a separated application note and should be stored inside AccessTime register bits field.

(3) FA20 parameter illustrates amount of time required to internally sample successive input Page Data. It is expressed in number of GPMC functional clock cycles. After each access to input Page Data, next input Page Data will be internally sampled by active functional clock edge after FA20 functional clock cycles. FA20 is also the duration of address phases for successive input Page Data (excluding first input Page Data). FA20 value should be stored in PageBurstAccessTime register bits field.

(4) GPMC_FCLK is an internal clock (GPMC functional clock) not provided externally

(5) The "DIR" (direction control) output signal is NOT pinned out on any of the device pads. It is an internal signal only representing a signal direction on the GPMC data bus.

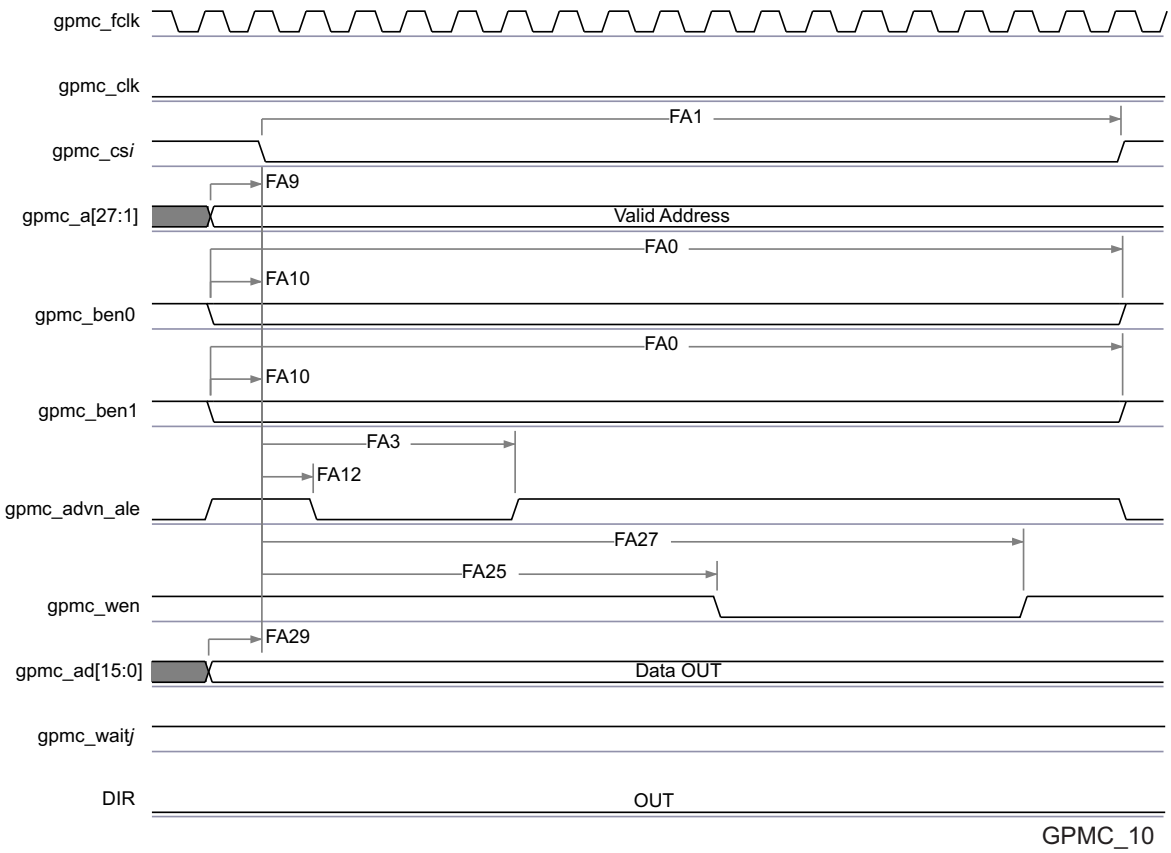


Figure 5-32. GPMC / NOR Flash - Asynchronous Write - Single Word Timing(1)

(1) In “gpmc_csi”, i = 0 to 7. In “gpmc_waitj”, j = 0 to 1.

(2) The "DIR" (direction control) output signal is NOT pinned out on any of the device pads. It is an internal signal only representing a signal direction on the GPMC data bus.

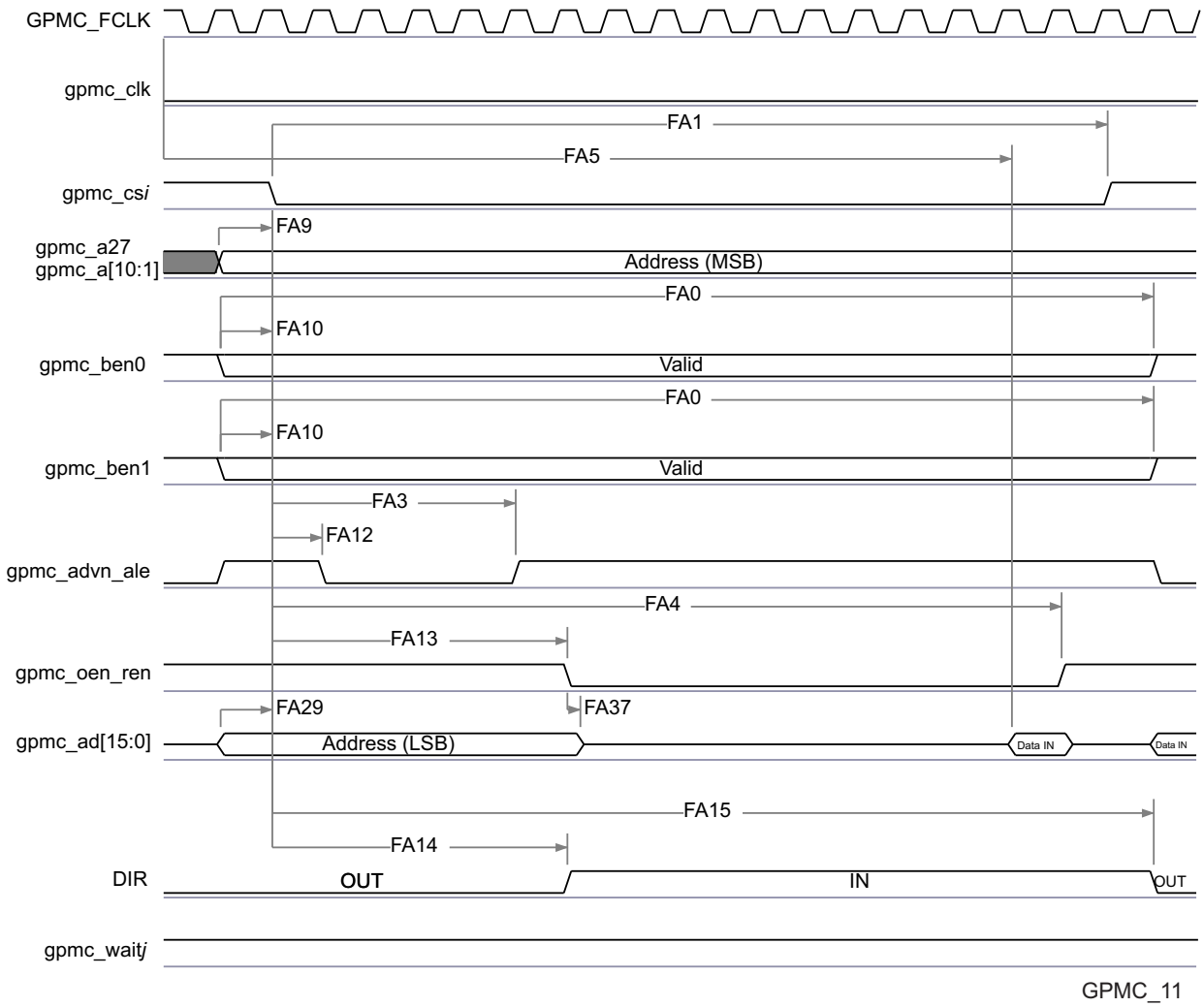


Figure 5-33. GPMC / Multiplexed NOR Flash - Asynchronous Read - Single Word Timing(1)(2)(3)

(1) In “gpmc_csi”, i = 0 to 7. In “gpmc_waitj”, j = 0 to 1

(2) FA5 parameter illustrates amount of time required to internally sample input Data. It is expressed in number of GPMC functional clock cycles. From start of read cycle and after FA5 functional clock cycles, input Data will be internally sampled by active functional clock edge. FA5 value should be stored inside AccessTime register bits field.

(3) GPMC_FCLK is an internal clock (GPMC functional clock) not provided externally

(4) The "DIR" (direction control) output signal is NOT pinned out on any of the device pads. It is an internal signal only representing a signal direction on the GPMC data bus.

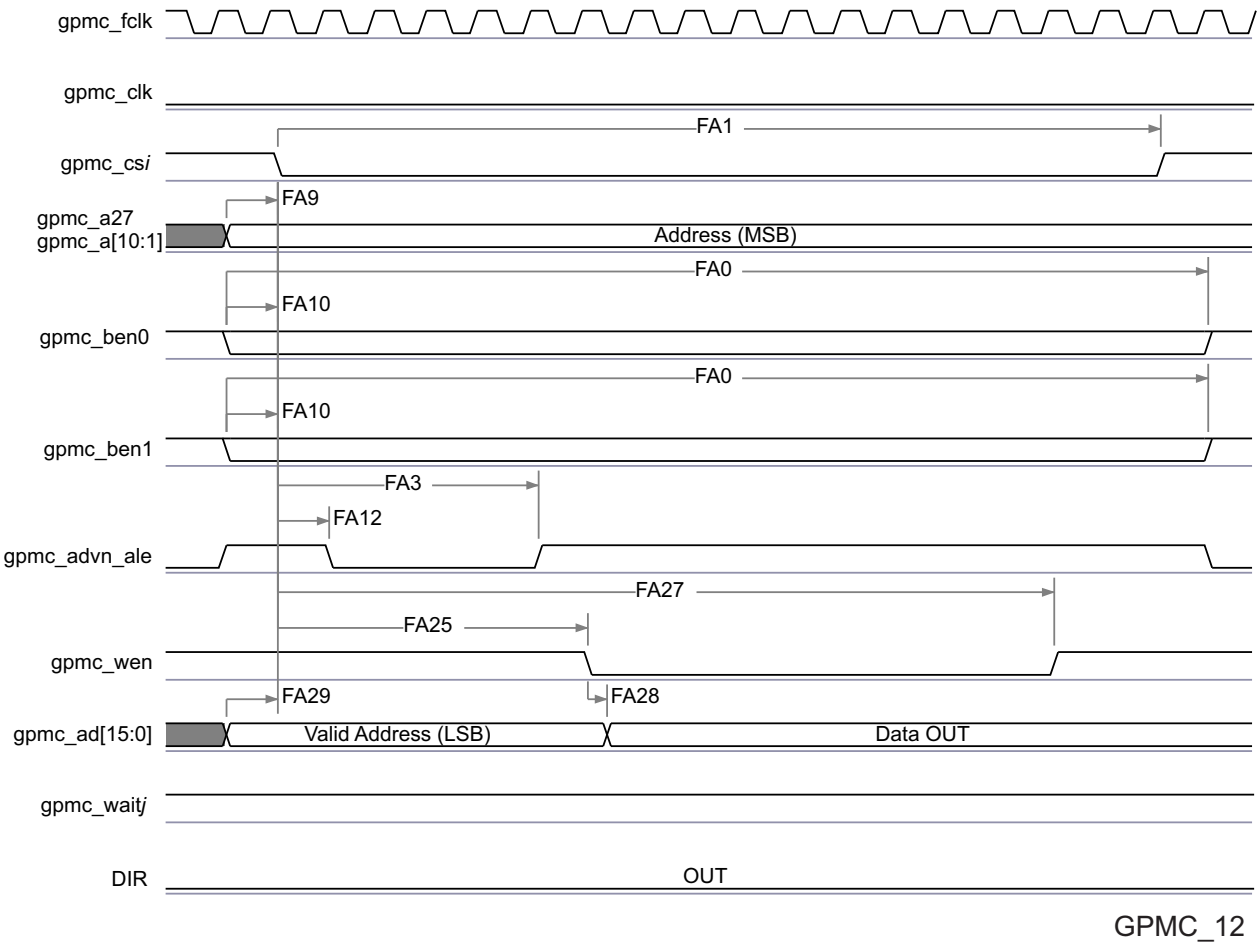


Figure 5-34. GPMC / Multiplexed NOR Flash - Asynchronous Write - Single Word Timing(1)

(1) In “gpmc_csi”, i = 0 to 7. In “gpmc_waitj”, j = 0 to 1.

(2) The "DIR" (direction control) output signal is NOT pinned out on any of the device pads. It is an internal signal only representing a signal direction on the GPMC data bus.

5.10.6.8.3 GPMC/NAND Flash Interface Asynchronous Timing

CAUTION

The I/O Timings provided in this section are valid only for some GPMC usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-53 and Table 5-54 assume testing over the recommended operating conditions and electrical characteristic conditions below (see Figure 5-35, Figure 5-36, Figure 5-37 and Figure 5-38).

Table 5-53. GPMC/NAND Flash Interface Timing Requirements

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
GNF12	$t_{acc(DAT)}$	Data maximum access time (GPMC_FCLK Cycles)		J (1)	cycles
-	$t_{su(DV-OEH)}$	Setup time, read gpmc_ad[15:0] valid before gpmc_oen_ren high	1.9		ns
-	$t_{h(OEH-DV)}$	Hold time, read gpmc_ad[15:0] valid after gpmc_oen_ren high	1		ns

(1) J = AccessTime × (TimeParaGranularity + 1)

Table 5-54. GPMC/NAND Flash Interface Switching Characteristics

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
-	$t_{r(DO)}$	Rising time, gpmc_ad[15:0] output data	0.447	4.067	ns
-	$t_{f(DO)}$	Fallling time, gpmc_ad[15:0] output data	0.43	4.463	ns
GNF0	$t_{w(nWEV)}$	Pulse duration, gpmc_wen valid time		A (1)	ns
GNF1	$t_{d(nCSV-nWEV)}$	Delay time, gpmc_cs[7:0] valid to gpmc_wen valid	B - 2 (2)	B + 4 (2)	ns
GNF2	$t_{d(CLEH-nWEV)}$	Delay time, gpmc_ben[1:0] high to gpmc_wen valid	C - 2 (3)	C + 4 (3)	ns
GNF3	$t_{d(nWEV-DV)}$	Delay time, gpmc_ad[15:0] valid to gpmc_wen valid	D - 2 (4)	D + 4 (4)	ns
GNF4	$t_{d(nWEIV-DIV)}$	Delay time, gpmc_wen invalid to gpmc_ad[15:0] invalid	E - 2 (5)	E + 4 (5)	ns
GNF5	$t_{d(nWEIV-CLEIV)}$	Delay time, gpmc_wen invalid to gpmc_ben[1:0] invalid	F - 2 (6)	F + 4 (6)	ns
GNF6	$t_{d(nWEIV-nCSIV)}$	Delay time, gpmc_wen invalid to gpmc_cs[7:0] invalid	G - 2 (7)	G + 4 (7)	ns
GNF7	$t_{d(ALEH-nWEV)}$	Delay time, gpmc_advn_ale high to gpmc_wen valid	C - 2 (3)	C + 4 (3)	ns
GNF8	$t_{d(nWEIV-ALEIV)}$	Delay time, gpmc_wen invalid to gpmc_advn_ale invalid	F - 2 (6)	F + 4 (6)	ns
GNF9	$t_{c(nWE)}$	Cycle time, write cycle time		H (8)	ns
GNF10	$t_{d(nCSV-nOEV)}$	Delay time, gpmc_cs[7:0] valid to gpmc_oen_ren valid	I - 2 (9)	I + 4 (9)	ns
GNF13	$t_{w(nOEV)}$	Pulse duration, gpmc_oen_ren valid time		K (10)	ns
GNF14	$t_{c(nOE)}$	Cycle time, read cycle time		L (11)	ns
GNF15	$t_{d(nOEIV-nCSIV)}$	Delay time, gpmc_oen_ren invalid to gpmc_cs[7:0] invalid	M - 2 (12)	M + 4 (12)	ns

(1) A = (WEOffTime – WEOnTime) × (TimeParaGranularity + 1) × GPMC_FCLK

(2) B = ((WEOnTime – CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (WEExtraDelay – CSExtraDelay)) × GPMC_FCLK

(3) C = ((WEOnTime – ADVOnTime) × (TimeParaGranularity + 1) + 0.5 × (WEExtraDelay – ADVExtraDelay)) × GPMC_FCLK

(4) D = (WEOnTime × (TimeParaGranularity + 1) + 0.5 × WEExtraDelay) × GPMC_FCLK

(5) E = (WrCycleTime – WEOffTime × (TimeParaGranularity + 1) – 0.5 × WEExtraDelay) × GPMC_FCLK

(6) F = (ADVWrOffTime – WEOffTime × (TimeParaGranularity + 1) + 0.5 × (ADVExtraDelay – WEExtraDelay) × GPMC_FCLK

(7) G = (CSWrOffTime – WEOffTime × (TimeParaGranularity + 1) + 0.5 × (CSExtraDelay – WEExtraDelay) × GPMC_FCLK

(8) H = WrCycleTime × (1 + TimeParaGranularity) × GPMC_FCLK

(9) I = ((OEOffTime + (n – 1) × PageBurstAccessTime – CSOnTime) × (TimeParaGranularity + 1) + 0.5 × (OEExtraDelay – CSExtraDelay)) × GPMC_FCLK

(10) K = (OEOffTime – OEOnTime) × (1 + TimeParaGranularity) × GPMC_FCLK

(11) L = RdCycleTime × (1 + TimeParaGranularity) × GPMC_FCLK

(12) M = (CSRdOffTime – OEOffTime × (TimeParaGranularity + 1) + 0.5 × (CSExtraDelay – OEExtraDelay) × GPMC_FCLK

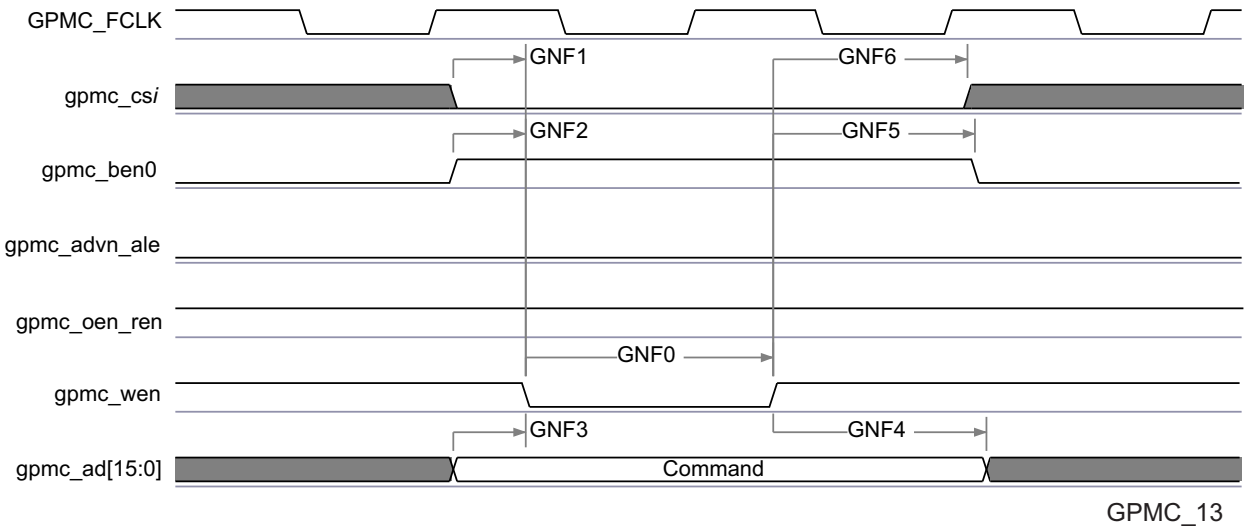


Figure 5-35. GPMC / NAND Flash - Command Latch Cycle Timing(1)

(1) In gpmc_cs*i*, i = 0 to 7.

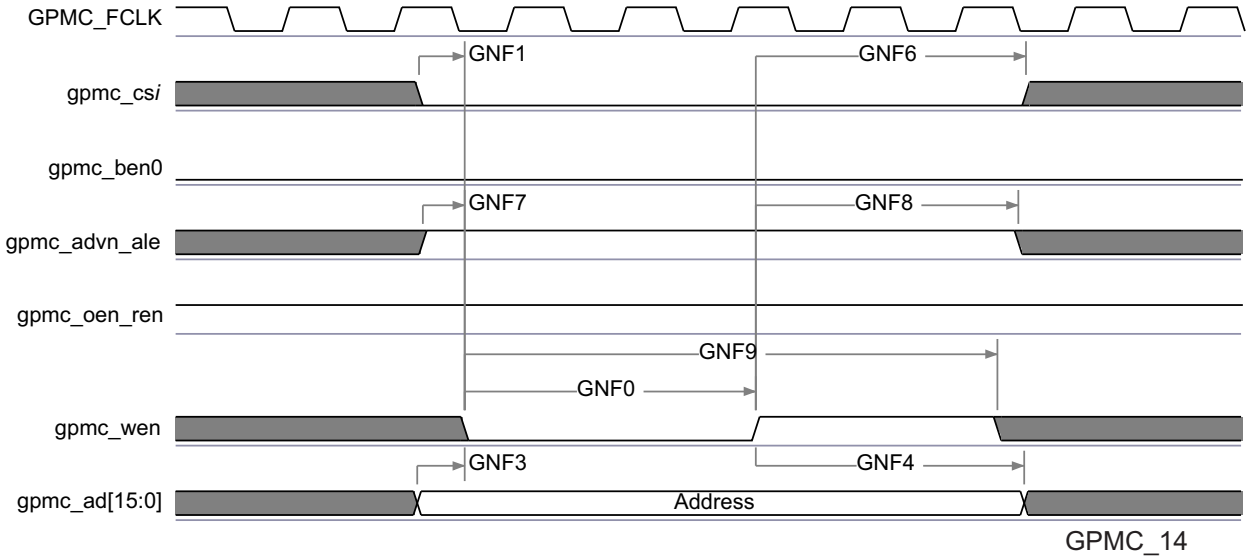


Figure 5-36. GPMC / NAND Flash - Address Latch Cycle Timing(1)

(1) In gpmc_cs*i*, i = 0 to 7.

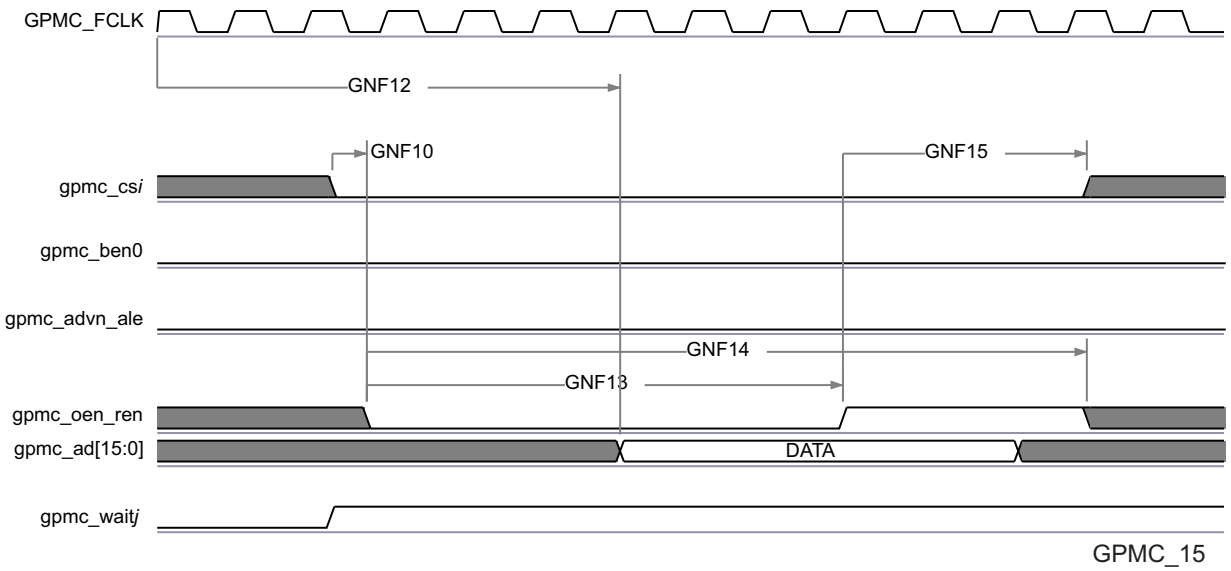


Figure 5-37. GPMC / NAND Flash - Data Read Cycle Timing(1)(2)(3)

(1) GNF12 parameter illustrates amount of time required to internally sample input Data. It is expressed in number of GPMC functional clock cycles. From start of read cycle and after GNF12 functional clock cycles, input data will be internally sampled by active functional clock edge. GNF12 value must be stored inside AccessTime register bits field.

(2) GPMC_FCLK is an internal clock (GPMC functional clock) not provided externally.

(3) In gpmc_cs*i*, i = 0 to 7. In gpmc_waitj, j = 0 to 1.

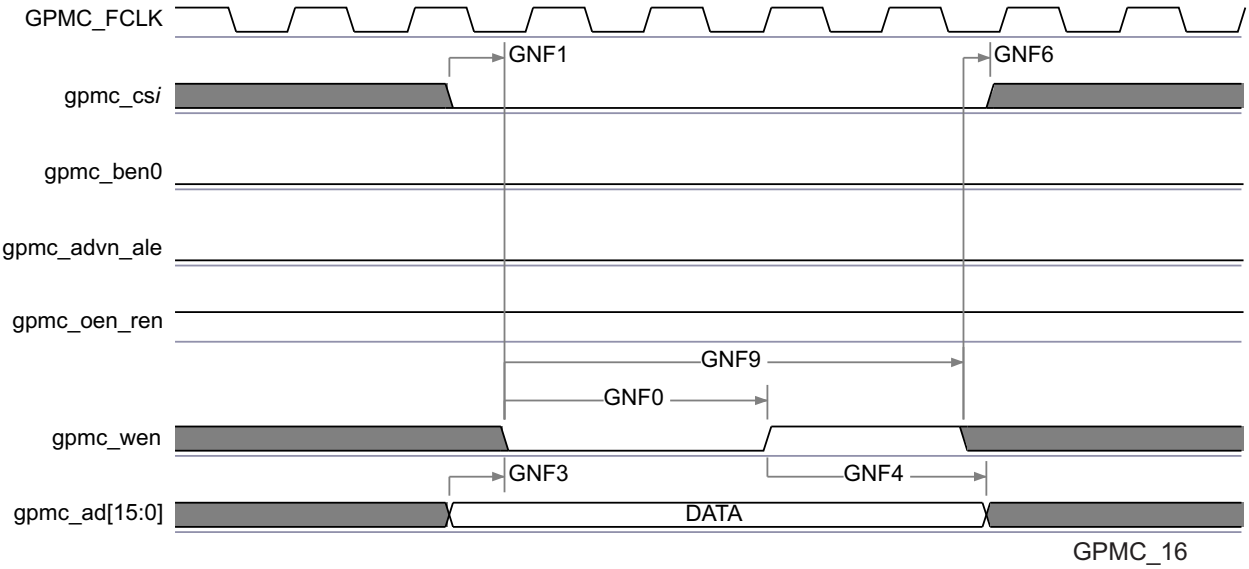


Figure 5-38. GPMC / NAND Flash - Data Write Cycle Timing(1)

(1) In gpmc_cs*i*, i = 0 to 7.

NOTE

To configure the desired virtual mode the user must set MODESELECT bit and DELAYMODE bitfield for each corresponding pad control register.

The pad control registers are presented in Table 4-32 and described in Device TRM, *Control Module Chapter*.

Virtual IO Timings Modes must be used to ensure some IO timings for GPMC. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Virtual IO Timings Modes. See Table 5-55 *Virtual Functions Mapping for GPMC* for a definition of the Virtual modes.

Table 5-55 presents the values for DELAYMODE bitfield.

Table 5-55. Virtual Functions Mapping for GPMC

BALL	BALL NAME	Delay Mode Value	MUXMODE							
		GPMC_VIRTUAL1	0	1	2	3	5	6	14[1]	14[1]
H5	gpmc_advn_ale	15	gpmc_advn_ale	gpmc_cs6		gpmc_wait1	gpmc_a2	gpmc_a23		
B4	gpmc_ad15	13	gpmc_ad15							
B1	gpmc_ad6	13	gpmc_ad6							
E1	gpmc_ad2	13	gpmc_ad2							
E10	vin2a_d9	9							gpmc_a25	
G6	gpmc_wen	15	gpmc_wen							
A3	gpmc_ad14	13	gpmc_ad14							
H3	gpmc_a13	15	gpmc_a13							
K4	gpmc_a8	14	gpmc_a8							
H4	gpmc_a14	15	gpmc_a14							
D1	gpmc_ad4	13	gpmc_ad4							
A5	gpmc_a26	15	gpmc_a26		gpmc_a20					
F1	gpmc_ad0	13	gpmc_ad0							
F6	gpmc_wait0	15	gpmc_wait0							
C10	vin2a_d11	9							gpmc_a23	
E2	gpmc_ad1	13	gpmc_ad1							
C4	gpmc_ad13	13	gpmc_ad13							
L2	gpmc_a2	14	gpmc_a2							
D2	gpmc_ad5	13	gpmc_ad5							
B10	vin2a_d8	9							gpmc_a26	
F3	gpmc_cs0	15	gpmc_cs0							
E8	vin2a_hsync0	9							gpmc_a27	
K3	gpmc_a4	14	gpmc_a4							
H2	gpmc_ben0	15	gpmc_ben0	gpmc_cs4						
J1	gpmc_a6	14	gpmc_a6							
K6	gpmc_a15	15	gpmc_a15							
B3	gpmc_ad11	13	gpmc_ad11							

Table 5-55. Virtual Functions Mapping for GPMC (continued)

BALL	BALL NAME	Delay Mode Value	MUXMODE							
		GPMC_VIRTUAL1	0	1	2	3	5	6	14[1]	14[1]
K5	gpmc_a16	15	gpmc_a16							
M2	gpmc_a1	14	gpmc_a1							
D7	gpmc_a24	15	gpmc_a24		gpmc_a18					
B5	gpmc_a23	15	gpmc_a23		gpmc_a17					
C2	gpmc_ad8	13	gpmc_ad8							
A2	gpmc_ad10	13	gpmc_ad10							
C3	gpmc_ad12	13	gpmc_ad12							
E7	gpmc_a20	15	gpmc_a20		gpmc_a14					
D10	vin2a_d10	9							gpmc_a24	
G3	gpmc_cs3	14	gpmc_cs3				gpmc_a1			
G5	gpmc_oen_ren	15	gpmc_oen_ren							
H1	gpmc_a9	14	gpmc_a9							
A6	gpmc_cs1	15	gpmc_cs1		gpmc_a22					
C1	gpmc_ad3	13	gpmc_ad3							
B2	gpmc_ad7	13	gpmc_ad7							
K1	gpmc_a7	14	gpmc_a7							
L1	gpmc_a3	14	gpmc_a3							
H6	gpmc_ben1	15	gpmc_ben1	gpmc_cs5			gpmc_a3			
L4	gpmc_clk	15	gpmc_clk	gpmc_cs7		gpmc_wait1				
C5	gpmc_a22	15	gpmc_a22		gpmc_a16					
G4	gpmc_cs2	15	gpmc_cs2							
C7	vin2a_fld0	11							gpmc_a27	gpmc_a18
J2	gpmc_a10	14	gpmc_a10							
G1	gpmc_a12	15	gpmc_a12				gpmc_a0			
G2	gpmc_a17	15	gpmc_a17							
K2	gpmc_a5	14	gpmc_a5							
D6	gpmc_a21	15	gpmc_a21		gpmc_a15					
B6	gpmc_a27	15	gpmc_a27		gpmc_a21					
D3	gpmc_ad9	13	gpmc_ad9							
A4	gpmc_a19	15	gpmc_a19		gpmc_a13					
C6	gpmc_a25	15	gpmc_a25		gpmc_a19					
M1	gpmc_a0	14	gpmc_a0							

Table 5-55. Virtual Functions Mapping for GPMC (continued)

BALL	BALL NAME	Delay Mode Value	MUXMODE							
		GPMC_VIRTUAL1	0	1	2	3	5	6	14[1]	14[1]
D8	vin2a_clk0	11							gpmc_a27	gpmc_a17
F2	gpmc_a18	15	gpmc_a18							
L3	gpmc_a11	14	gpmc_a11							

(1) Some signals listed are virtual functions that present alternate multiplexing options. These virtual functions are controlled via CTRL_CORE_ALT_SELECT_MUX or CTRL_CORE_VIP_MUX_SELECT registers. For more information on how to use these options, please refer to Device TRM, Chapter Control Module, Section Pad Configuration Registers.

5.10.6.9 Timers

The device has 16 general-purpose (GP) timers (TIMER1 - TIMER16), two watchdog timers, and a 32-kHz synchronized timer (COUNTER_32K) that have the following features:

- Dedicated input trigger for capture mode and dedicated output trigger/pulse width modulation (PWM) signal
- Interrupts generated on overflow, compare, and capture
- Free-running 32-bit upward counter
- Supported modes:
  - Compare and capture modes
  - Auto-reload mode
  - Start-stop mode
- On-the-fly read/write register (while counting)

The device has two system watchdog timer (WD_TIMER1 and WD_TIMER2) that have the following features:

- Free-running 32-bit upward counter
- On-the-fly read/write register (while counting)
- Reset upon occurrence of a timer overflow condition

The device includes one instance of the 32-bit watchdog timer: WD_TIMER2, also called the MPU watchdog timer.

The watchdog timer is used to provide a recovery mechanism for the device in the event of a fault condition, such as a non-exiting code loop.

In are presented the specific groupings of signals (IOSET) for use with TIMERS.

NOTE

For additional information on the Timer Module, see the Device TRM.

5.10.6.10 I2C

The device includes 6 inter-integrated circuit (I2C) modules which provide an interface to other devices compliant with Philips Semiconductors Inter-IC bus (I2C-bus™) specification version 2.1. External components attached to this 2-wire serial bus can transmit/receive 8-bit data to/from the device through the I2C module.

NOTE

Note that, on I2C1 and I2C2, due to characteristics of the open drain IO cells, HS mode is not supported.

NOTE

Inter-integrated circuit i (i=1 to 6) module is also referred to as I2Ci.

NOTE

For more information, see the Multimaster High-Speed I2C Controller section of the Device TRM.

Table 5-56, Table 5-57 and Figure 5-39 assume testing over the recommended operating conditions and electrical characteristic conditions below.

Table 5-56. Timing Requirements for I2C Input Timings(1)

NO.	PARAMETER	DESCRIPTION	STANDARD MODE		FAST MODE		UNIT
			MIN	MAX	MIN	MAX	
1	$t_{c(SCL)}$	Cycle time, SCL	10		2.5		µs
2	$t_{su(SCLH-SDAL)}$	Setup time, SCL high before SDA low (for a repeated START condition)	4.7		0.6		µs
3	$t_{h(SDAL-SCLL)}$	Hold time, SCL low after SDA low (for a START and a repeated START condition)	4		0.6		µs
4	$t_{w(SCLL)}$	Pulse duration, SCL low	4.7		1.3		µs
5	$t_{w(SCLH)}$	Pulse duration, SCL high	4		0.6		µs
6	$t_{su(SDAV-SCLH)}$	Setup time, SDA valid before SCL high	250		100(2)		ns
7	$t_{h(SCLL-SDAV)}$	Hold time, SDA valid after SCL low	0(3)	3.45(4)	0(3)	0.9(4)	µs
8	$t_{w(SDAH)}$	Pulse duration, SDA high between STOP and START conditions	4.7		1.3		µs
9	$t_{r(SDA)}$	Rise time, SDA		1000	$20 + 0.1C_b$(5)	300(3)	ns
10	$t_{r(SCL)}$	Rise time, SCL		1000	$20 + 0.1C_b$(5)	300(3)	ns
11	$t_{f(SDA)}$	Fall time, SDA		300	$20 + 0.1C_b$(5)	300(3)	ns
12	$t_{f(SCL)}$	Fall time, SCL		300	$20 + 0.1C_b$(5)	300(3)	ns
13	$t_{su(SCLH-SDAH)}$	Setup time, SCL high before SDA high (for STOP condition)	4		0.6		µs
14	$t_{w(SP)}$	Pulse duration, spike (must be suppressed)			0	50	ns
15	C_b (5)	Capacitive load for each bus line		400		400	pF

(1) The I2C pins SDA and SCL do not feature fail-safe I/O buffers. These pins could potentially draw current when the device is powered down.

(2) A Fast-mode I^2C-bus™ device can be used in a Standard-mode I^2C-bus system, but the requirement $t_{su(SDA-SCLH)} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line t_r max + $t_{su(SDA-SCLH)}$= 1000 + 250 = 1250 ns (according to the Standard-mode I^2C-Bus Specification) before the SCL line is released.

(3) A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the V_{IHmin} of the SCL signal) to bridge the undefined region of the falling edge of SCL.

(4) The maximum $t_{h(SDA-SCLL)}$ has only to be met if the device does not stretch the low period [$t_{w(SCLL)}$] of the SCL signal.

(5) C_b = total capacitance of one bus line in pF. If mixed with HS-mode devices, faster fall-times are allowed.

Table 5-57. Timing Requirements for I^2C HS-Mode (I^2C3/4/5/6 Only)(1)

NO.	PARAMETER	DESCRIPTION	C_b = 100 pF MAX		C_b = 400 pF (2)		UNIT
			MIN	MAX	MIN	MAX	
1	$t_{c(SCL)}$	Cycle time, SCL	0.294		0.588		µs
2	$t_{su(SCLH-SDAL)}$	Set-up time, SCL high before SDA low (for a repeated START condition)	160		160		ns
3	$t_{h(SDAL-SCLL)}$	Hold time, SCL low after SDA low (for a repeated START condition)	160		160		ns

Table 5-57. Timing Requirements for I²C HS-Mode (I²C3/4/5/6 Only)(1) (continued)

NO.	PARAMETER	DESCRIPTION	C_b = 100 pF MAX		C_b = 400 pF (2)		UNIT
			MIN	MAX	MIN	MAX	
4	$t_{w(SCLL)}$	LOW period of the SCLH clock	160		320		ns
5	$t_{w(SCLH)}$	HIGH period of the SCLH clock	60		120		ns
6	$t_{su(SDAV-SCLH)}$	Setup time, SDA valid vefore SCL high	10		10		ns
7	$t_{h(SCLL-SDAV)}$	Hold time, SDA valid after SCL low	0 (3)	70	0 (3)	150	ns
13	$t_{su(SCLH-SDAH)}$	Setup time, SCL high before SDA high (for a STOP condition)	160		160		ns
14	$t_{w(SP)}$	Pulse duration, spike (must be suppressed)	0	10	0	10	ns
15	C_b (2)	Capacitive load for SDAH and SCLH lines		100		400	pF
16	C_b	Capacitive load for SDAH + SDA line and SCLH + SCL line		400		400	pF

(1) I²C HS-Mode is only supported on I²C3/4/5/6. I2C HS-Mode is not supported on I²C1/2.

(2) For bus line loads C_b between 100 and 400 pF the timing parameters must be linearly interpolated.

(3) A device must internally provide a Data hold time to bridge the undefined part between V_{IH} and V_{IL} of the falling edge of the SCLH signal. An input circuit with a threshold as low as possible for the falling edge of the SCLH signal minimizes this hold time.

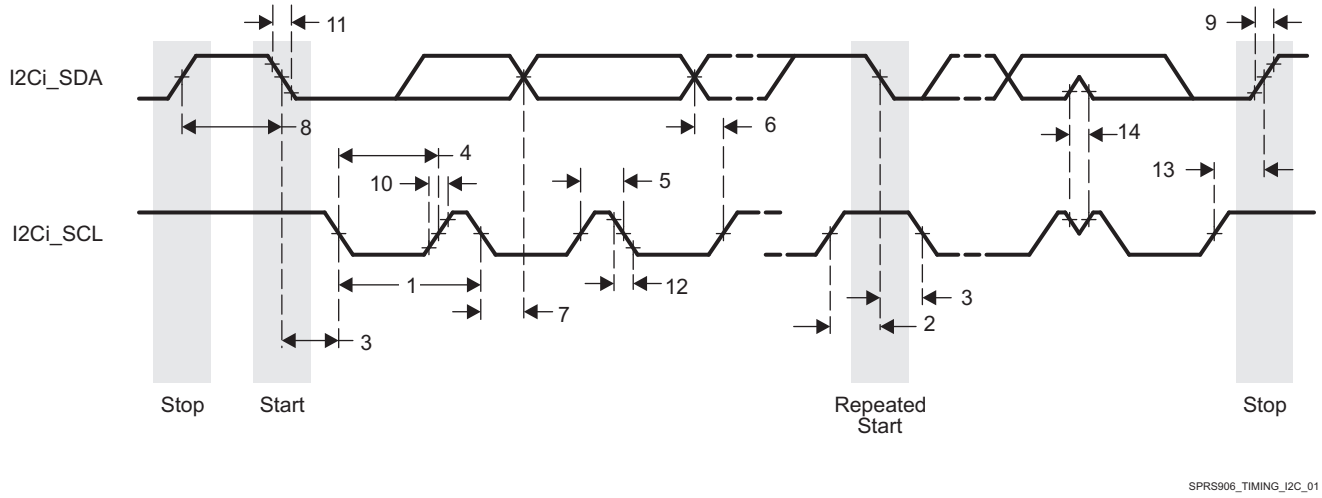


Figure 5-39. I2C Receive Timing

Table 5-58 and Figure 5-40 assume testing over the recommended operating conditions and electrical characteristic conditions below.

Table 5-58. Switching Characteristics Over Recommended Operating Conditions for I2C Output Timings(2)

NO.	PARAMETER	DESCRIPTION	STANDARD MODE		FAST MODE		UNIT
			MIN	MAX	MIN	MAX	
16	$t_{c(SCL)}$	Cycle time, SCL	10		2.5		µs
17	$t_{su(SCLH-SDAL)}$	Setup time, SCL high before SDA low (for a repeated START condition)	4.7		0.6		µs
18	$t_{h(SDAL-SCLL)}$	Hold time, SCL low after SDA low (for a START and a repeated START condition)	4		0.6		µs
19	$t_{w(SCLL)}$	Pulse duration, SCL low	4.7		1.3		µs
20	$t_{w(SCLH)}$	Pulse duration, SCL high	4		0.6		µs
21	$t_{su(SDAV-SCLH)}$	Setup time, SDA valid before SCL high	250		100		ns

Table 5-58. Switching Characteristics Over Recommended Operating Conditions for I2C Output Timings(2) (continued)

NO.	PARAMETER	DESCRIPTION	STANDARD MODE		FAST MODE		UNIT
			MIN	MAX	MIN	MAX	
22	$t_{h(SCLL-SDAV)}$	Hold time, SDA valid after SCL low (for I2C bus devices)	0	3.45	0	0.9	µs
23	$t_{w(SDAH)}$	Pulse duration, SDA high between STOP and START conditions	4.7		1.3		µs
24	$t_{r(SDA)}$	Rise time, SDA		1000	$20 + 0.1C_b$ (1) (3)	300(3)	ns
25	$t_{r(SCL)}$	Rise time, SCL		1000	$20 + 0.1C_b$ (1) (3)	300(3)	ns
26	$t_{f(SDA)}$	Fall time, SDA		300	$20 + 0.1C_b$ (1) (3)	300(3)	ns
27	$t_{f(SCL)}$	Fall time, SCL		300	$20 + 0.1C_b$ (1) (3)	300(3)	ns
28	$t_{su(SCLH-SDAH)}$	Setup time, SCL high before SDA high (for STOP condition)	4		0.6		µs
29	C_p	Capacitance for each I2C pin		10		10	pF

(1) C_b = total capacitance of one bus line in pF. If mixed with HS-mode devices, faster fall-times are allowed.

(2) Software must properly configure the I2C module registers to achieve the timings shown in this table. See the Device TRM for details.

(3) These timings apply only to I2C1 and I2C2. I2C3, I2C4, I2C5 and I2C6 use standard LVCMOS buffers to emulate open-drain buffers and their rise/fall times should be referenced in the device IBIS model.

NOTE

I2C emulation is achieved by configuring the LVCMOS buffers to output Hi-Z instead of driving high when transmitting logic-1.

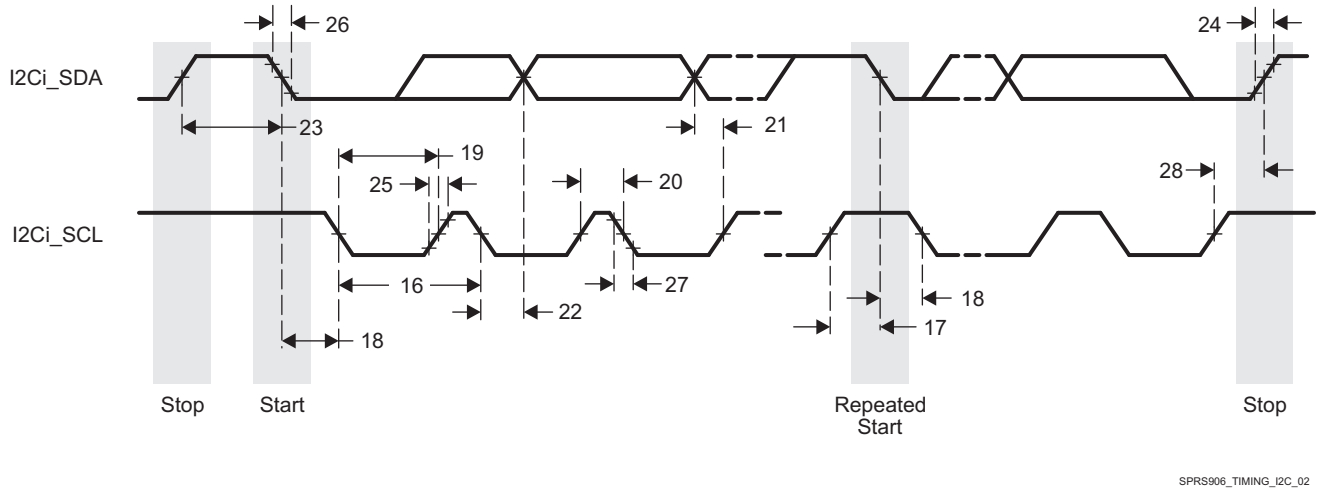


Figure 5-40. I2C Transmit Timing

In are presented the specific groupings of signals (IOSET) for use with I2C1/2/3/4/5.

5.10.6.11 HDQ1W

The module is intended to work with both HDQ and 1-Wire protocols. The protocols use a single wire to communicate between the master and the slave. The protocols employ an asynchronous return to one mechanism where, after any command, the line is pulled high.

NOTE

For more information, see the HDQ / 1-Wire section of the Device TRM.

5.10.6.11.1 HDQ / 1-Wire — HDQ Mode

Table 5-59 and Table 5-60 assume testing over the recommended operating conditions and electrical characteristic conditions below (see Figure 5-41, Figure 5-42, Figure 5-43 and Figure 5-44).

Table 5-59. HDQ/1-Wire Timing Requirements—HDQ Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	t_{CYCH}	Read bit window timing	190	250	µs
2	t_{HW1}	Read one data valid after HDQ low	32(2)	66(2)	µs
3	t_{HW0}	Read zero data hold after HDQ low	70(2)	145(2)	µs
4	t_{RSPS}	Response time from HDQ slave device(1)	190	320	µs

(1) Defined by software.

(2) If the HDQ slave device drives a logic-low state after tHW0 maximum, it can be interpreted as a break pulse. For more information see "HDQ / 1-Wire Switching Characteristics - HDQ Mode" and the HDQ/1-Wire chapter of the TRM.

Table 5-60. HDQ / 1-Wire Switching Characteristics - HDQ Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
5	t_B	Break timing	190		µs
6	t_{BR}	Break recovery time	40		µs
7	t_{CYCD}	Write bit windows timing	190		µs
8	t_{DW1}	Write one data valid after HDQ low	0.5	50	µs
9	t_{DW0}	Write zero data hold after HDQ low	86	145	µs

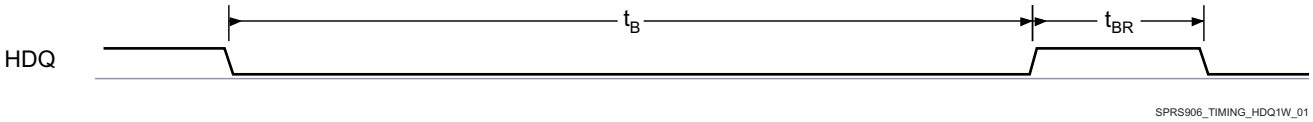


Figure 5-41. HDQ Break and Break Recovery Timing — HDQ Interface Writing to Slave

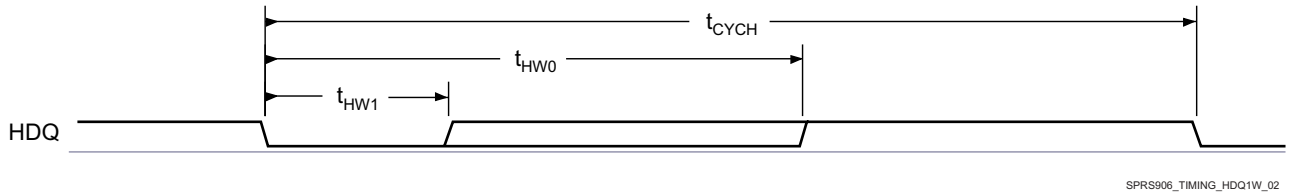


Figure 5-42. Device HDQ Interface Bit Read Timing (Data)

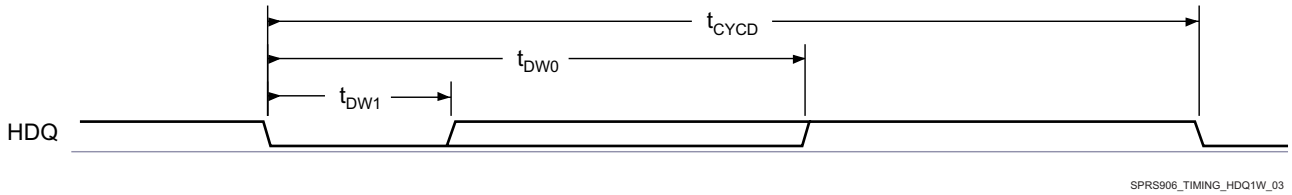


Figure 5-43. Device HDQ Interface Bit Write Timing (Command / Address or Data)

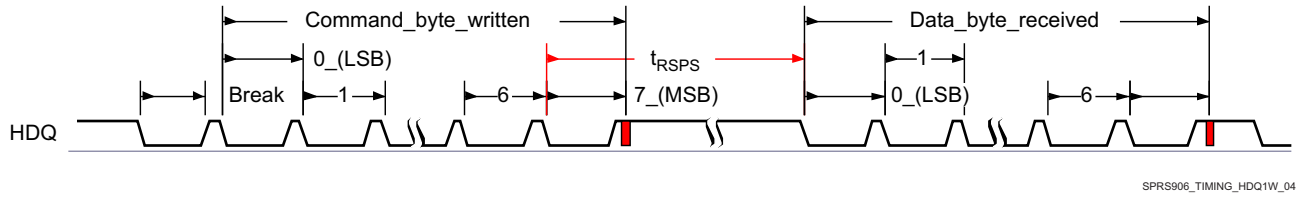


Figure 5-44. HDQ Communication Timing

5.10.6.11.2 HDQ/1-Wire—1-Wire Mode

Table 5-61 and Table 5-62 assume testing over the recommended operating conditions and electrical characteristic conditions below (see Figure 5-45, Figure 5-46 and Figure 5-47).

Table 5-61. HDQ / 1-Wire Timing Requirements - 1-Wire Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
10	t_{PDH}	Presence pulse delay high	15	60	µs
11	t_{PDL}	Presence pulse delay low	60	240	µs
12	t_{RDV}	Read data valid time	t_{LOWR}	15	µs
13	t_{REL}	Read data release time	0	45	µs

Table 5-62. HDQ / 1-Wire Switching Characteristics - 1-Wire Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
14	t_{RSTL}	Reset time low	480	960	µs
15	t_{RSTH}	Reset time high	480		µs
16	t_{SLOT}	Bit cycle time	60	120	µs
17	t_{LOW1}	Write bit-one time	1	15	µs
18	t_{LOW0}	Write bit-zero time(2)	60	120	µs
19	t_{REC}	Recovery time	1		µs
20	t_{LOWR}	Read bit strobe time(1)	1	15	µs

(1) t_{LOWR} (low pulse sent by the master) must be short as possible to maximize the master sampling window.

(2) t_{LOWR} must be less than t_{SLOT}.

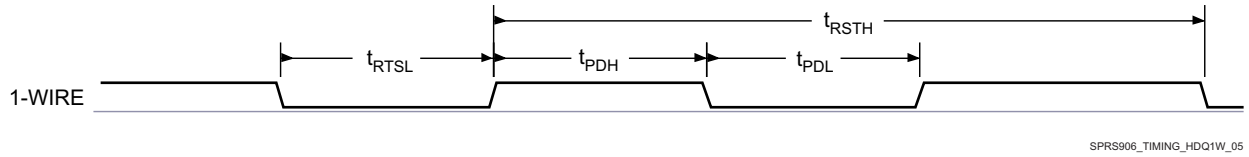


Figure 5-45. 1-Wire—Break (Reset)

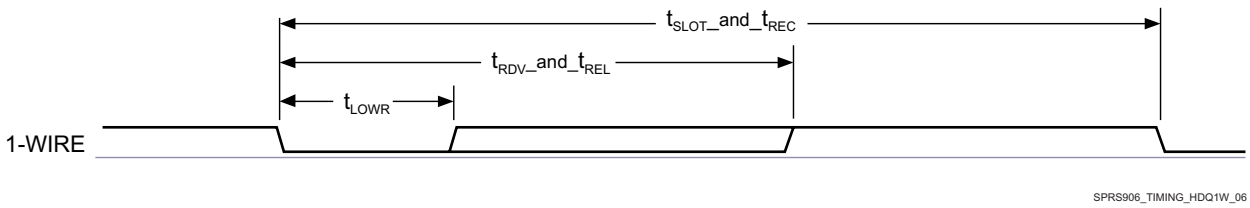


Figure 5-46. 1-Wire—Read Bit (Data)

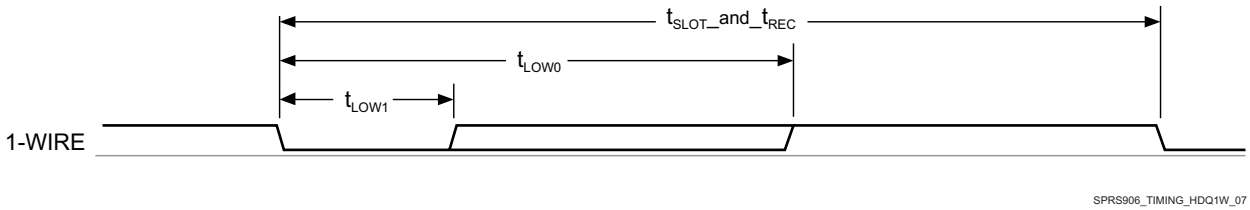


Figure 5-47. 1-Wire—Write Bit-One Timing (Command / Address or Data)

5.10.6.12 UART

The UART performs serial-to-parallel conversions on data received from a peripheral device and parallel-to-serial conversion on data received from the CPU. There are 10 UART modules in the device. Only one UART supports IrDA features. Each UART can be used for configuration and data exchange with a number of external peripheral devices or interprocessor communication between devices

The UARTi (where i = 1 to 10) include the following features:

- 16C750 compatibility
- 64-byte FIFO buffer for receiver and 64-byte FIFO for transmitter
- Baud generation based on programmable divisors N (where N = 1…16 384) operating from a fixed functional clock of 48 MHz or 192 MHz
- Break character detection and generation
- Configurable data format:
  - Data bit: 5, 6, 7, or 8 bits
  - Parity bit: Even, odd, none
  - Stop-bit: 1, 1.5, 2 bit(s)
- Flow control: Hardware (RTS/CTS) or software (XON/XOFF)
- Only UART1 module has extended modem control signals (CD, RI, DTR, DSR)
- Only UART3 supports IrDA

NOTE

For more information, see the UART section of the Device TRM.

Table 5-63, Table 5-64 and Figure 5-48 assume testing over the recommended operating conditions and electrical characteristic conditions below.

Table 5-63. Timing Requirements for UART

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
4	$t_{w(RX)}$	Pulse width, receive data bit, 15/30/100pF high or low	0.96U(1)	1.05U(1)	ns
5	$t_{w(CTS)}$	Pulse width, receive start bit, 15/30/100pF high or low	0.96U(1)	1.05U(1)	ns
	$t_{d(RTS-TX)}$	Delay time, transmit start bit to transmit data	P(2)		ns
	$t_{d(CTS-TX)}$	Delay time, receive start bit to transmit data	P(2)		ns

(1) U = UART baud time = 1/programmed baud rate

(2) P = Clock period of the reference clock (FCLK, usually 48 MHz or 192MHz).

Table 5-64. Switching Characteristics Over Recommended Operating Conditions for UART

NO.	PARAMETER	DESCRIPTION		MIN	MAX	UNIT
	$f_{(baud)}$	Maximum programmable baud rate	15 pF		12	MHz
			30 pF		0.23	
			100 pF		0.115	
2	$t_{w(TX)}$	Pulse width, transmit data bit, 15/30/100 pF high or low		U - 2(1)	U + 2(1)	ns

Table 5-64. Switching Characteristics Over Recommended Operating Conditions for UART (continued)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
3	$t_{w(RTS)}$	Pulse width, transmit start bit, 15/30/100 pF high or low	U - 2[(1)]	U + 2[(1)]	ns

(1) U = UART baud time = 1/programmed baud rate

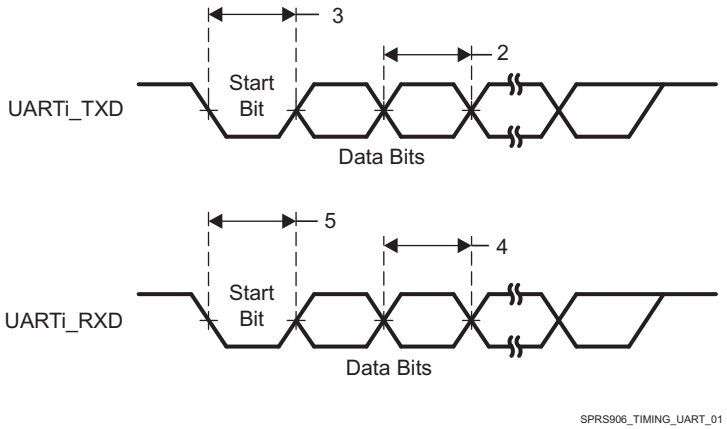


Figure 5-48. UART Timing

In are presented the specific groupings of signals (IOSET) for use with UART.

5.10.6.13 McSPI

The McSPI is a master/slave synchronous serial bus. There are four separate McSPI modules (SPI1, SPI2, SPI3, and SPI4) in the device. All these four modules support up to four external devices (four chip selects) and are able to work as both master and slave.

The McSPI modules include the following main features:

- Serial clock with programmable frequency, polarity, and phase for each channel
- Wide selection of SPI word lengths, ranging from 4 to 32 bits
- Up to four master channels, or single channel in slave mode
- Master multichannel mode:
  - Full duplex/half duplex
  - Transmit-only/receive-only/transmit-and-receive modes
  - Flexible input/output (I/O) port controls per channel
  - Programmable clock granularity
  - SPI configuration per channel. This means, clock definition, polarity enabling and word width
- Power management through wake-up capabilities
- Programmable timing control between chip select and external clock generation
- Built-in FIFO available for a single channel.
- Each SPI module supports multiple chip select pins spim_cs[i], whete i = 1 to 4.

NOTE

For more information, see the Serial Communication Interface section of the device TRM.

NOTE

The McSPIm module (m = 1 to 4) is also referred to as SPIm.

CAUTION

The I/O timings provided in this section are applicable for all combinations of signals for SPI1 and SPI2. However, the timings are valid only for SPI3 and SPI4 if signals within a single IOSET are used. The IOSETS are defined in Table 5-67.

Table 5-65, Figure 5-49 and Figure 5-50 present Timing Requirements for McSPI - Master Mode.

Table 5-65. Timing Requirements for SPI - Master Mode (1)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
SM1	$t_{c(SPICLK)}$	Cycle time, spi_sclk (1) (2)	SPI1/2/3/4	20.8 (3)		ns
SM2	$t_{w(SPICLKL)}$	Typical Pulse duration, spi_sclk low (1)		0.5 × P-1 (4)		ns
SM3	$t_{w(SPICLKH)}$	Typical Pulse duration, spi_sclk high (1)		0.5 × P-1 (4)		ns
SM4	$t_{su(MISO-SPICLK)}$	Setup time, spi_d[x] valid before spi_sclk active edge (1)		3.5		ns
SM5	$t_{h(SPICLK-MISO)}$	Hold time, spi_d[x] valid after spi_sclk active edge (1)		3.7		ns
SM6	$t_{d(SPICLK-SIMO)}$	Delay time, spi_sclk active edge to spi_d[x] transition (1)	SPI1	-3.57	4.1	ns
			SPI2	-3.9	3.6	ns
			SPI3	-4.9	4.7	ns
			SPI4	-4.3	4.5	ns
SM7	$t_{d(CS-SIMO)}$	Delay time, spi_cs[x] active edge to spi_d[x] transition			5	ns
SM8	$t_{d(CS-SPICLK)}$	Delay time, spi_cs[x] active to spi_sclk first edge (1)	MASTER_PHA0 (5)	B-4.2 (6)		ns
			MASTER_PHA1 (5)	A-4.2 (7)		ns
SM9	$t_{d(SPICLK-CS)}$	Delay time, spi_sclk last edge to spi_cs[x] inactive (1)	MASTER_PHA0 (5)	A-4.2 (7)		ns
			MASTER_PHA1 (5)	B-4.2 (6)		ns

(1) This timing applies to all configurations regardless of SPI_CLK polarity and which clock edges are used to drive output data and capture input data.

(2) Related to the SPI_CLK maximum frequency.

(3) 20.8ns cycle time = 48MHz

(4) P = SPICLK period.

(5) SPI_CLK phase is programmable with the PHA bit of the SPI_CH(i)CONF register.

(6) B = (TCS + 0.5) × TSPICLKREF × Fratio, where TCS is a bit field of the SPI_CH(i)CONF register and Fratio = Even ≥2.

(7) When P = 20.8 ns, A = (TCS + 1) × TSPICLKREF, where TCS is a bit field of the SPI_CH(i)CONF register. When P > 20.8 ns, A = (TCS + 0.5) × Fratio × TSPICLKREF, where TCS is a bit field of the SPI_CH(i)CONF register.

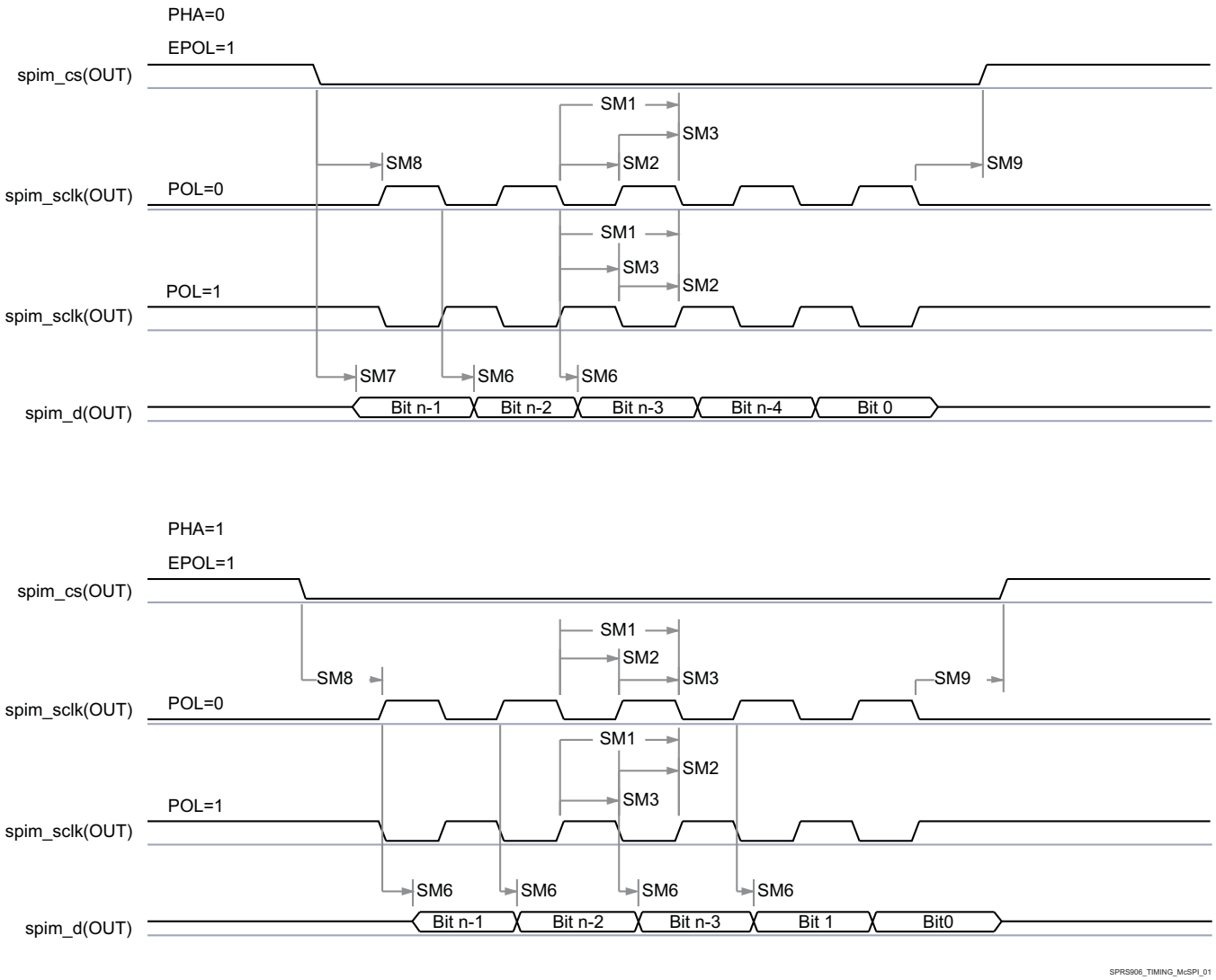


Figure 5-49. McSPI - Master Mode Transmit

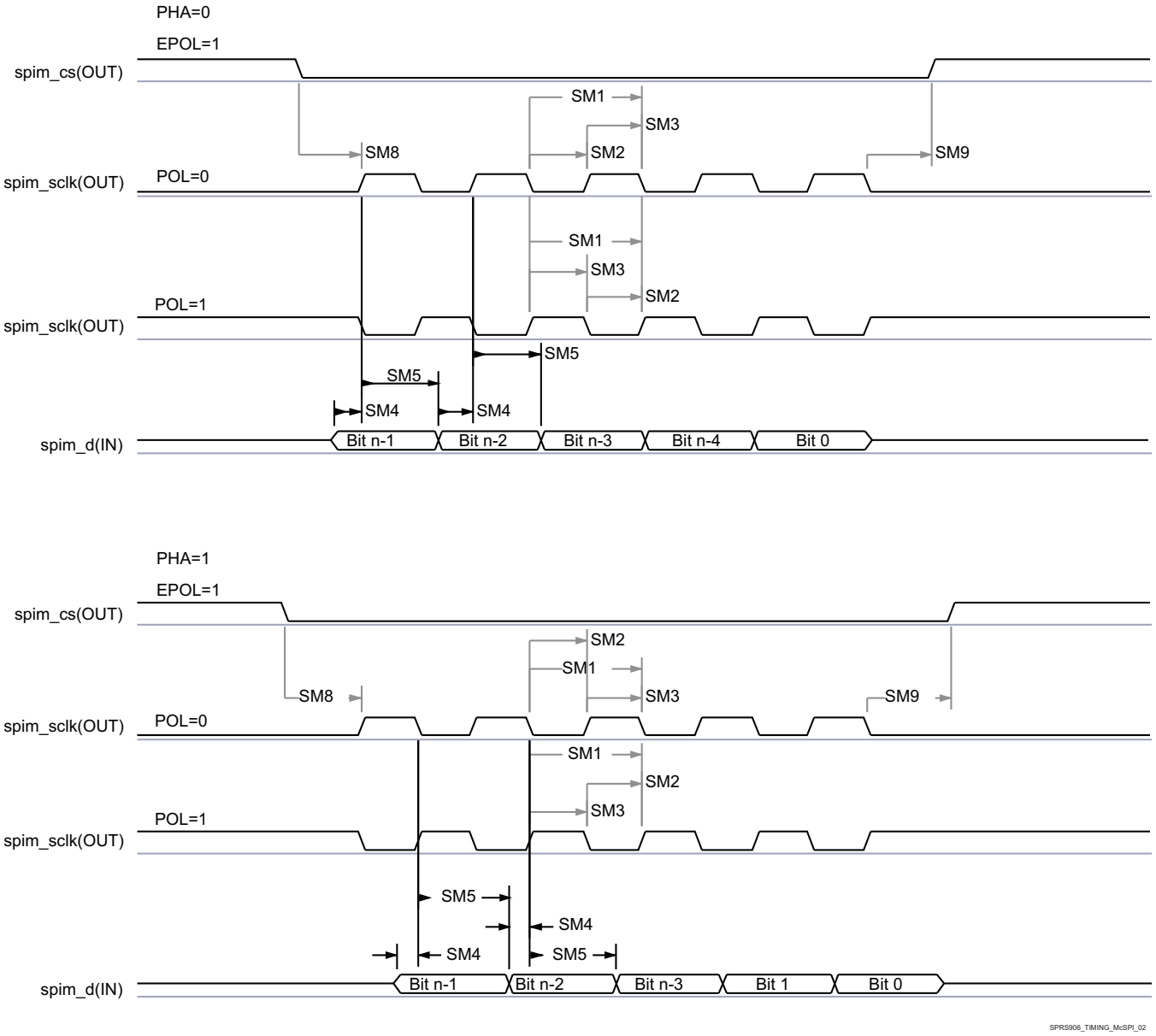


Figure 5-50. McSPI - Master Mode Receive

Table 5-66, Figure 5-51 and Figure 5-52 present Timing Requirements for McSPI - Slave Mode.

Table 5-66. Timing Requirements for SPI - Slave Mode

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
SS1 (1)	$t_{c(SPICLK)}$	Cycle time, spi_sclk		62.5 (2) (3)		ns
SS2 (1)	$t_{w(SPICLKL)}$	Typical Pulse duration, spi_sclk low		0.45 × P (4)		ns
SS3 (1)	$t_{w(SPICLKH)}$	Typical Pulse duration, spi_sclk high		0.45 × P (4)		ns
SS4 (1)	$t_{su(SIMO-SPICLK)}$	Setup time, spi_d[x] valid before spi_sclk active edge		5		ns
SS5 (1)	$t_{h(SPICLK-SIMO)}$	Hold time, spi_d[x] valid after spi_sclk active edge		5		ns
SS6 (1)	$t_{d(SPICLK-SOMI)}$	Delay time, spi_sclk active edge to mcspi_somi transition	SPI1/2/3	2	26.6	ns
			SPI4	2	20.1	ns

Table 5-66. Timing Requirements for SPI - Slave Mode (continued)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
SS7 (5)	$t_{d(CS-SOMI)}$	Delay time, spi_cs[x] active edge to mcspi_somi transition			20.95	ns
SS8 (1)	$t_{su(CS-SPICLK)}$	Setup time, spi_cs[x] valid before spi_sclk first edge		5		ns
SS9 (1)	$t_{h(SPICLK-CS)}$	Hold time, spi_cs[x] valid after spi_sclk last edge	SPI1/2	5		ns
			SPI3	7.5		ns
			SPI4	6		ns

(1) This timing applies to all configurations regardless of SPI_CLK polarity and which clock edges are used to drive output data and capture input data.

(2) When operating the SPI interface in RX-only mode, the minimum Cycle time is 26ns (38.4MHz)

(3) 62.5ns Cycle time = 16 MHz

(4) P = SPICLK period.

(5) PHA = 0; SPI_CLK phase is programmable with the PHA bit of the SPI_CH(i)CONF register.

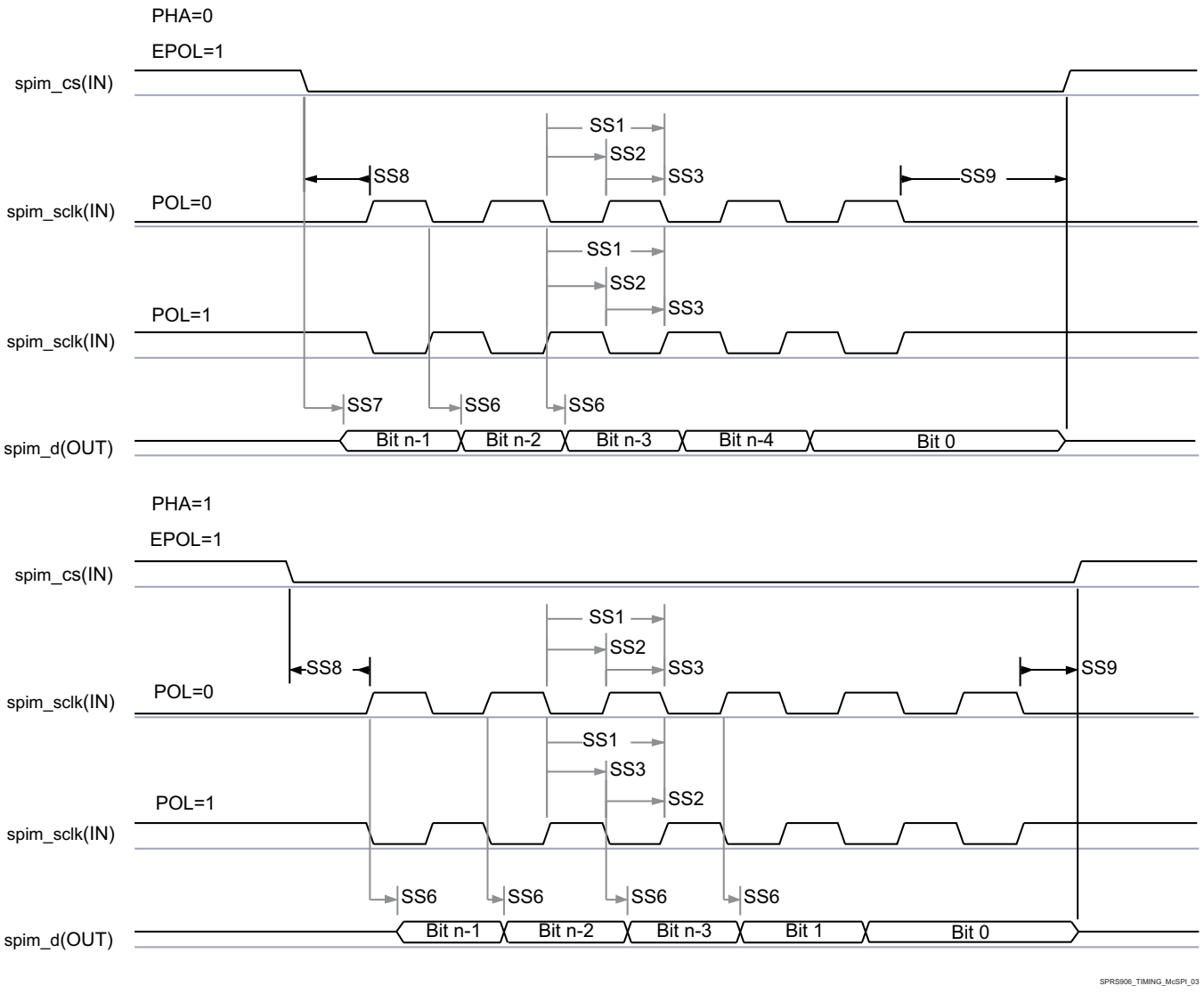


Figure 5-51. McSPI - Slave Mode Transmit

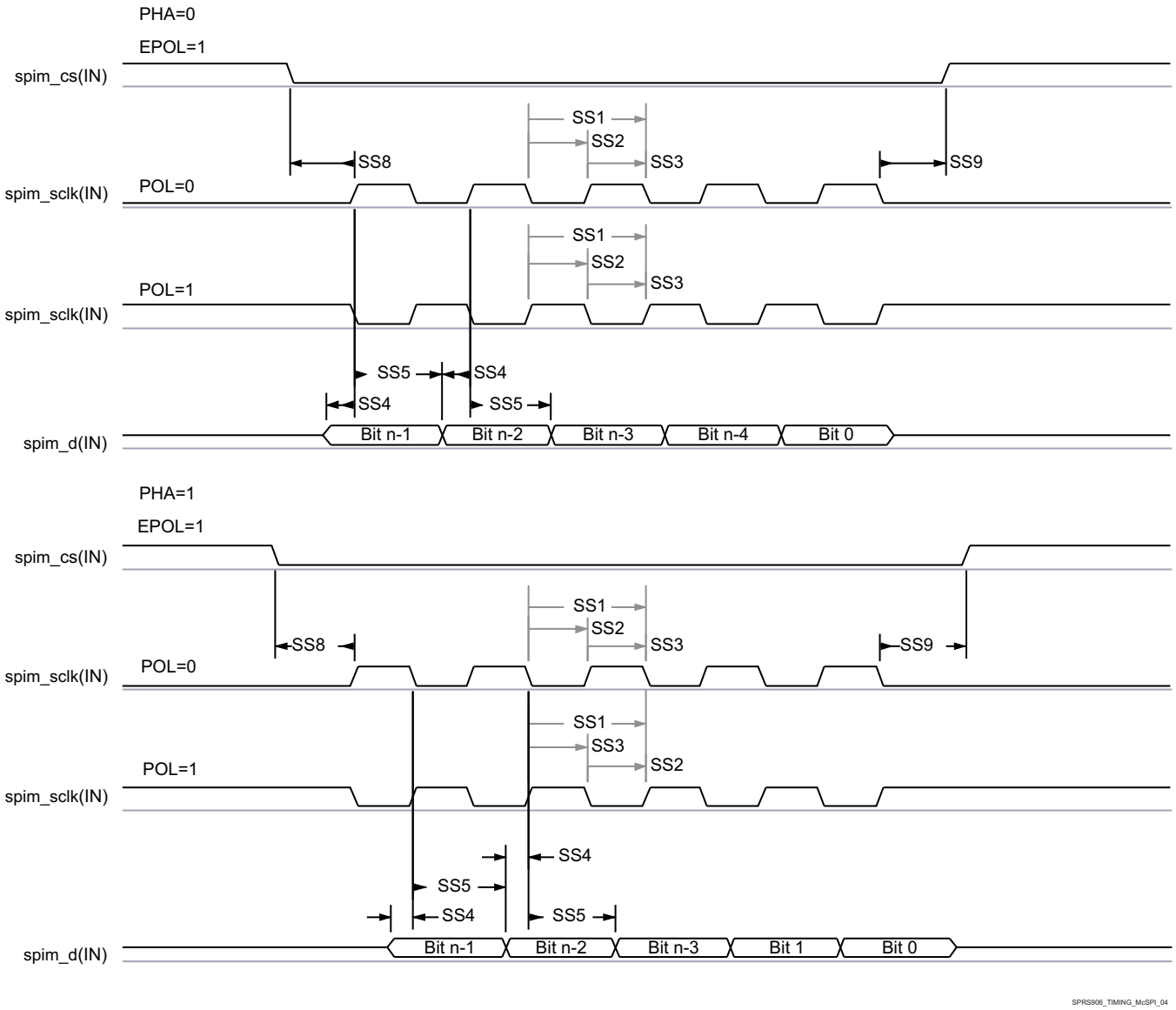


Figure 5-52. McSPI - Slave Mode Receive

In Table 5-67 are presented the specific groupings of signals (IOSET) for use with SPI3 and SPI4.

Table 5-67. McSPI3/4 IOSETs

SIGNALS	IOSET1		IOSET2		IOSET3		IOSET4		IOSET5	
	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX
McSPI3										
spi3_cs0			T5	7	B18	3	D23	2	AA3	1
spi3_cs1			W2	1	A19	3			W2	1
spi3_d0			T4	7	B16	3	A24	2	AA2	1
spi3_d1			N6	7	B17	3	B25	2	Y4	1
spi3_sclk			N5	7	A18	3	C23	2	Y1	1
McSPI4										
spi4_cs0	L3	8	B9	8	R1	7	AC4	2	AB1	1
spi4_cs1	G1	8	G1	8	N6	8	N6	8	N6	8
spi4_cs2	H3	8	H3	8	T4	8	T4	8	T4	8

Table 5-67. McSPI3/4 IOSETs (continued)

SIGNALS	IOSET1		IOSET2		IOSET3		IOSET4		IOSET5	
	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX
spi4_cs3	H4	8	H4	8	T5	8	T5	8	T5	8
spi4_d0	J2	8	C8	8	R2	7	AA5	2	AA4	1
spi4_d1	H1	8	B8	8	P3	7	U6	2	AA1	1
spi4_sclk	K4	8	E8	8	P4	7	AC3	2	Y3	1

5.10.6.14 QSPI

The Quad SPI (QSPI) module is a type of SPI module that allows single, dual or quad read access to external SPI devices. This module has a memory mapped register interface, which provides a direct interface for accessing data from external SPI devices and thus simplifying software requirements. It works as a master only. There is one QSPI module in the device and it is primary intended for fast booting from quad-SPI flash memories.

General SPI features:

- Programmable clock divider
- Six pin interface (DCLK, CS_N, DOUT, DIN, QDIN1, QDIN2)
- 4 external chip select signals
- Support for 3-, 4- or 6-pin SPI interface
- Programmable CS_N to DOUT delay from 0 to 3 DCLKs
- Programmable signal polarities
- Programmable active clock edge
- Software controllable interface allowing for any type of SPI transfer

NOTE

For more information, see the Quad Serial Peripheral Interface section of the Device TRM.

CAUTION

The I/O Timings provided in this section are only valid when all QSPI Chip Selects used in a system are configured to use the same Clock Mode (either Clock Mode 0 or Clock Mode 3).

CAUTION

The I/O Timings provided in this section are valid only for some QSPI usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-68 and Table 5-69 Present Timing and Switching Characteristics for Quad SPI Interface.

Table 5-68. Switching Characteristics for QSPI

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
Q1	$t_{c(SCLK)}$	Cycle time, sclk	Default Timing Mode, Clock Mode 0	11.71		ns
			Default Timing Mode, Clock Mode 3	20.8		ns

Table 5-68. Switching Characteristics for QSPI (continued)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
Q2	$t_{w(SCLKL)}$	Pulse duration, sclk low		Y × P-1 (1)		ns
Q3	$t_{w(SCLKH)}$	Pulse duration, sclk high		Y × P-1 (1)		ns
Q4	$t_{d(CS-SCLK)}$	Delay time, sclk falling edge to cs active edge, CS3:0	Default Timing Mode	-M × P-1.6 (2) (3)	-M × P+2.6 (2) (3)	ns
Q5	$t_{d(SCLK-CS)}$	Delay time, sclk falling edge to cs inactive edge, CS3:0	Default Timing Mode	N × P-1.6 (2) (3)	N × P+2.6 (2) (3)	ns
Q6	$t_{d(SCLK-D0)}$	Delay time, sclk falling edge to d[0] transition	Default Timing Mode	-1.6	2.6	ns
Q7	$t_{ena(CS-D0LZ)}$	Enable time, cs active edge to d[0] driven (lo-z)		-P-3.5	-P+2.5	ns
Q8	$t_{dis(CS-D0Z)}$	Disable time, cs active edge to d[0] tri-stated (hi-z)		-P-2.5	-P+2.0	ns
Q9	$t_{d(SCLK-D0)}$	Delay time, sclk first falling edge to first d[0] transition	PHA=0 Only, Default Timing Mode	-1.6-P(2)	2.6-P(2)	ns

(1) The Y parameter is defined as follows:
If DCLK_DIV is 0 or ODD then, Y equals 0.5.
If DCLK_DIV is EVEN then, Y equals (DCLK_DIV/2) / (DCLK_DIV+1).
For best performance, it is recommended to use a DCLK_DIV of 0 or ODD to minimize the duty cycle distortion. The HSDIVIDER on CLKOUTX2_H13 output of DPLL_PER can be used to achieve the desired clock divider ratio. All required details about clock division factor DCLK_DIV can be found in the device-specific Technical Reference Manual.

(2) P = SCLK period.

(3) M=QSPI_SPI_DC_REG.DDx + 1 when Clock Mode 0.
M=QSPI_SPI_DC_REG.DDx when Clock Mode 3.
N = 2 when Clock Mode 0.
N = 3 when Clock Mode 3.

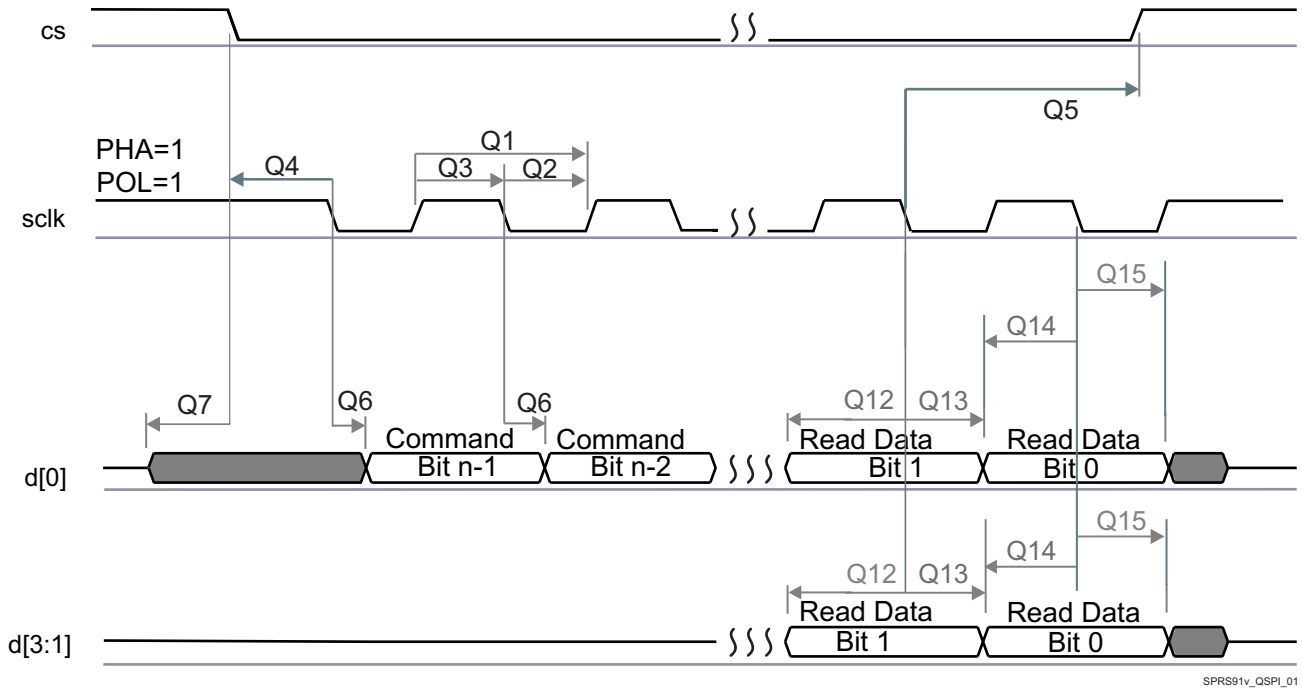


Figure 5-53. QSPI Read (Clock Mode 3)

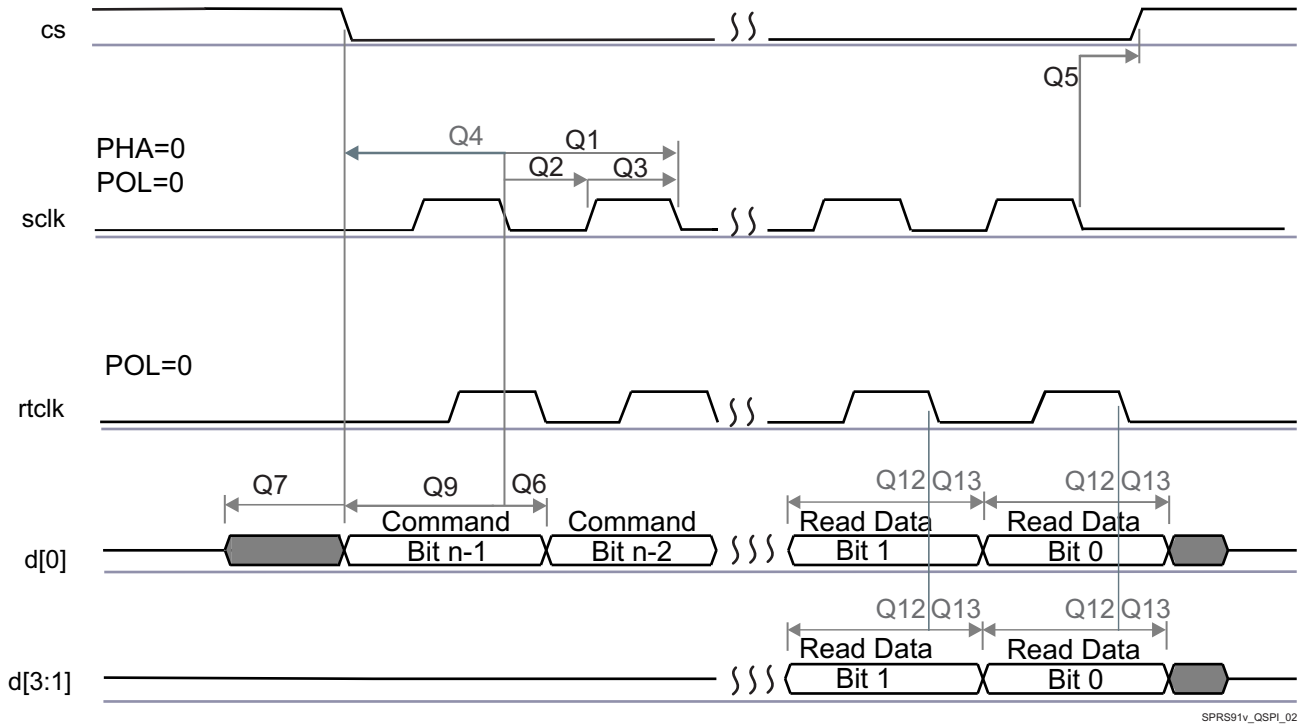


Figure 5-54. QSPI Read (Clock Mode 0)

CAUTION

The I/O Timings provided in this section are valid only for some QSPI usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-69. Timing Requirements for QSPI(3)(2)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
Q2	$t_{su(D-RTCLK)}$	Setup time, d[3:0] valid before falling rtclk edge	Default Timing Mode, Clock Mode 0	4.6		ns
	$t_{su(D-SCLK)}$	Setup time, d[3:0] valid before falling sclk edge	Default Timing Mode, Clock Mode 3	12.3		ns
Q13	$t_{h(RTCLK-D)}$	Hold time, d[3:0] valid after falling rtclk edge	Default Timing Mode, Clock Mode 0	-0.1		ns
	$t_{h(SCLK-D)}$	Hold time, d[3:0] valid after falling sclk edge	Default Timing Mode, Clock Mode 3	0.1		ns
Q14	$t_{su(D-SCLK)}$	Setup time, final d[3:0] bit valid before final falling sclk edge	Default Timing Mode, Clock Mode 3	12.3-P (1)		ns
Q15	$t_{h(SCLK-D)}$	Hold time, final d[3:0] bit valid after final falling sclk edge	Default Timing Mode, Clock Mode 3	0.1+P (1)		ns

(1) P = SCLK period.

(2) Clock Modes 1 and 2 are not supported.

(3) The Device captures data on the falling clock edge in Clock Mode 0 and 3, as opposed to the traditional rising clock edge. Although non-standard, the falling-edge-based setup and hold time timings have been designed to be compatible with standard SPI devices that launch data on the falling edge in Clock Modes 0 and 3.

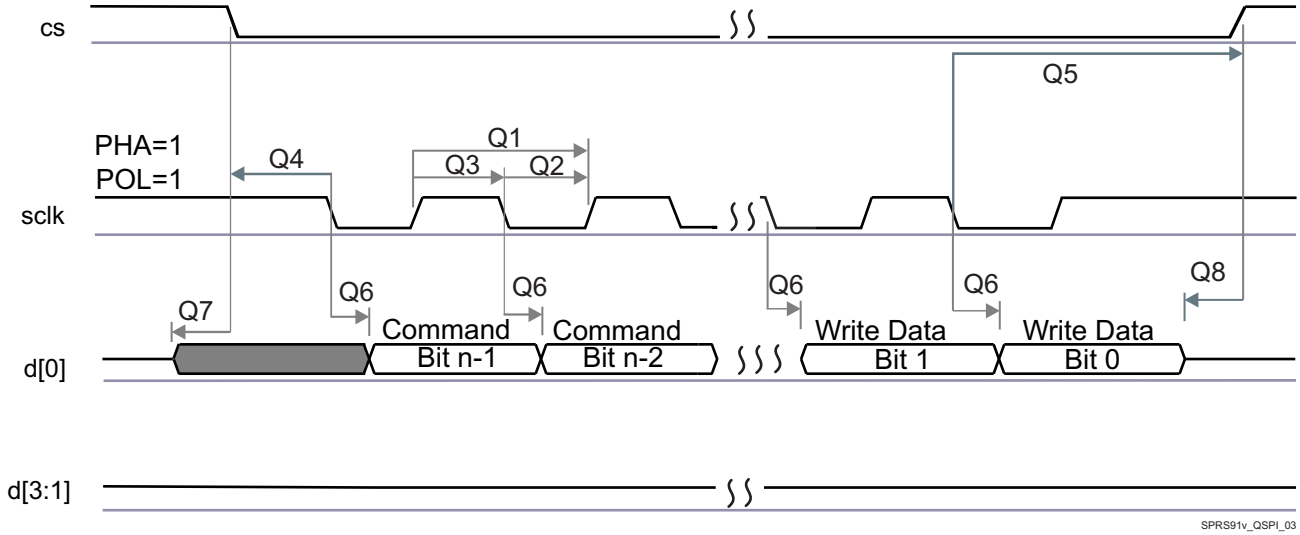


Figure 5-55. QSPI Write (Clock Mode 3)

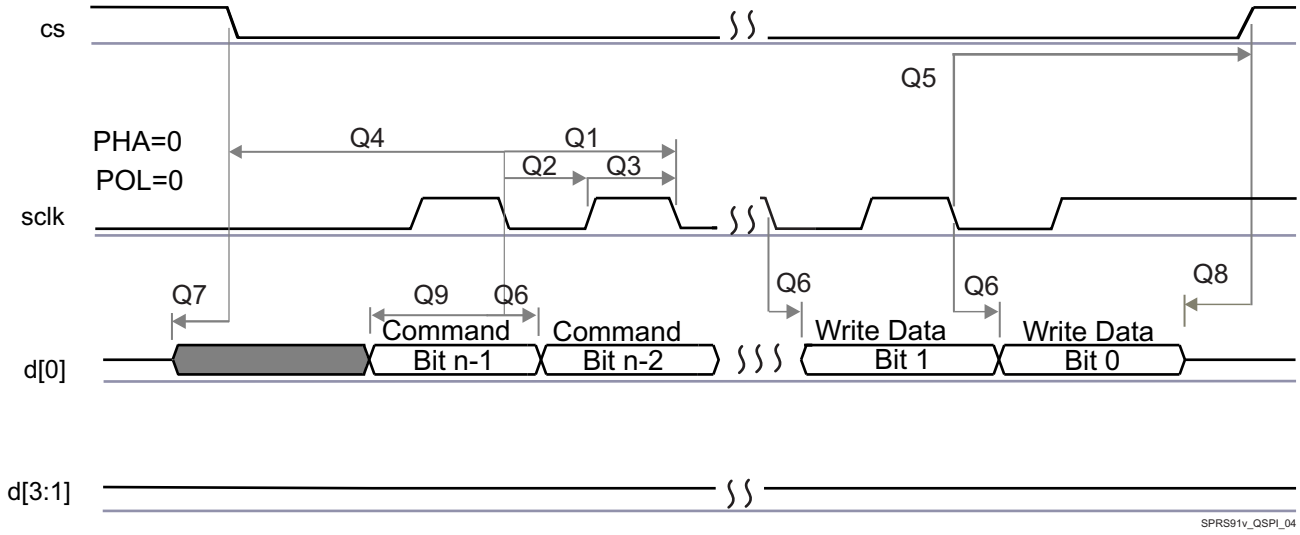


Figure 5-56. QSPI Write (Clock Mode 0)

CAUTION

The I/O Timings provided in this section are valid only for some QSPI usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

NOTE

To configure the desired Manual IO Timing Mode the user must follow the steps described in section Manual IO Timing Modes of the Device TRM.

The associated registers to configure are listed in the **CFG REGISTER** column. For more information see the Control Module chapter in the Device TRM.

Manual IO Timings Modes must be used to ensure some IO timings for QSPI. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-70 *Manual Functions Mapping for QSPI* for a definition of the Manual modes.

Table 5-70 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-70. Manual Functions Mapping for QSPI

BALL	BALL NAME	QSPI1_MANUAL1		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		1
L1	gpmc_a3	0	0	CFG_GPMC_A3_OUT	qspi1_cs2
K3	gpmc_a4	0	0	CFG_GPMC_A4_OUT	qspi1_cs3
H3	gpmc_a13	0	0	CFG_GPMC_A13_IN	qspi1_rtclk
H4	gpmc_a14	2247	1186	CFG_GPMC_A14_IN	qspi1_d3
K6	gpmc_a15	2176	1197	CFG_GPMC_A15_IN	qspi1_d2
K5	gpmc_a16	2229	1268	CFG_GPMC_A16_IN	qspi1_d0
K5	gpmc_a16	0	0	CFG_GPMC_A16_OUT	qspi1_d0
G2	gpmc_a17	2251	1217	CFG_GPMC_A17_IN	qspi1_d1
F2	gpmc_a18	0	0	CFG_GPMC_A18_OUT	qspi1_sclk
G4	gpmc_cs2	0	0	CFG_GPMC_CS2_OUT	qspi1_cs0
G3	gpmc_cs3	0	0	CFG_GPMC_CS3_OUT	qspi1_cs1

5.10.6.15 McASP

The multichannel audio serial port (McASP) functions as a general-purpose audio serial port optimized for the needs of multichannel audio applications. The McASP is useful for time-division multiplexed (TDM) stream, Inter-Integrated Sound (I2S) protocols, and intercomponent digital audio interface transmission (DIT).

The device have integrated 8 McASP modules (McASP1-McASP8) with:

- McASP1 and McASP2 modules supporting 16 channels with independent TX/RX clock/sync domain
- McASP3 through McASP7 modules supporting 4 channels with independent TX/RX clock/sync domain
- McASP8 module supporting 2 channels with independent TX/RX clock/sync domain

NOTE

For more information, see the Serial Communication Interface section of the Device TRM.

CAUTION

The I/O Timings provided in this section are valid only for some McASP usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-71, Table 5-72, Table 5-73 and Figure 5-57 present Timing Requirements for McASP1 to McASP8.

Table 5-71. Timing Requirements for McASP1(1)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
1	$t_{c(AHCLKX)}$	Cycle time, AHCLKX		20		ns
2	$t_{w(AHCLKX)}$	Pulse duration, AHCLKX high or low		0.35P (2)		ns
3	$t_{c(ACLKRX)}$	Cycle time, ACLKR/X		20		ns
4	$t_{w(ACLKRX)}$	Pulse duration, ACLKR/X high or low		0.5R - 3 (3)		ns
5	$t_{su(AFSRX-ACLK)}$	Setup time, AFSR/X input valid before ACLKR/X	ACLKR/X int	20.5		ns
			ACLKR/X ext in ACLKR/X ext out	4		ns
6	$t_{h(ACLK-AFSRX)}$	Hold time, AFSR/X input valid after ACLKR/X	ACLKR/X int	-1		ns
			ACLKR/X ext in ACLKR/X ext out	1.7		ns
7	$t_{su(AXR-ACLK)}$	Setup time, AXR input valid before ACLKR/X	ACLKR/X int	21.6		ns
			ACLKR/X ext in ACLKR/X ext out	11.5		ns
8	$t_{h(ACLK-AXR)}$	Hold time, AXR input valid after ACLKR/X	ACLKR/X int	-1		ns
			ACLKR/X ext in ACLKR/X ext out	1.8		ns

(1) ACLKR internal: ACLKRCTL.CLKRM=1, PDIR.ACLKR = 1
ACLKR external input: ACLKRCTL.CLKRM=0, PDIR.ACLKR=0
ACLKR external output: ACLKRCTL.CLKRM=0, PDIR.ACLKR=1
ACLKX internal: ACLKXCTL.CLKXM=1, PDIR.ACLKX = 1
ACLKX external input: ACLKXCTL.CLKXM=0, PDIR.ACLKX=0
ACLKX external output: ACLKXCTL.CLKXM=0, PDIR.ACLKX=1

(2) P = AHCLKX period in ns.

(3) R = ACLKR/X period in ns.

Table 5-72. Timing Requirements for McASP2(1)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
1	$t_{c(AHCLKX)}$	Cycle time, AHCLKX		20		ns
2	$t_{w(AHCLKX)}$	Pulse duration, AHCLKX high or low		0.35P (2)		ns
3	$t_{c(ACLKX)}$	Cycle time, ACLKX	Any Other Conditions	20		ns
			ACLKX/AFSX (In Sync Mode) and AXR are all inputs "80M" Virtual IO Timing Modes	12.5		ns
4	$t_{w(ACLKX)}$	Pulse duration, ACLKX high or low	Any Other Conditions	0.5R - 3 (3)		ns
			ACLKX/AFSX (In Sync Mode) and AXR are all inputs "80M" Virtual IO Timing Modes	0.38R (3)		ns
5	$t_{su(AFSX-ACLK)}$	Setup time, AFSX input valid before ACLKX	ACLKX int	20.3		ns
			ACLKX ext in ACLKX ext out	4.5		ns
			ACLKX ext in ACLKX ext out "80M" Virtual IO Timing Modes	3		ns

Table 5-72. Timing Requirements for McASP2[1] (continued)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
6	$t_{h(ACLK-AFSX)}$	Hold time, AFSX input valid after ACLKX	ACLKX int	-1		ns
			ACLKX ext in ACLKX ext out	1.8		ns
			ACLKX ext in ACLKX ext out "80M" Virtual IO Timing Modes	3		ns
7	$t_{su(AXR-ACLK)}$	Setup time, AXR input valid before ACLKX	ACLKX int	21.1		ns
			ACLKX ext in ACLKX ext out	4.5		ns
			ACLKX ext in ACLKX ext out "80M" Virtual IO Timing Modes	3		ns
8	$t_{h(ACLK-AXR)}$	Hold time, AXR input valid after ACLKX	ACLKX int	-1		ns
			ACLKX ext in ACLKX ext out	1.8		ns
			ACLKX ext in ACLKX ext out "80M" Virtual IO Timing Modes	3		ns

(1) ACLKX internal: ACLKXCTL.CLKXM=1, PDIR.ACLKX = 1
ACLKX external input: ACLKXCTL.CLKXM=0, PDIR.ACLKX=0
ACLKX external output: ACLKXCTL.CLKXM=0, PDIR.ACLKX=1

(2) P = AHCLKX period in ns.

(3) R = ACLKX period in ns.

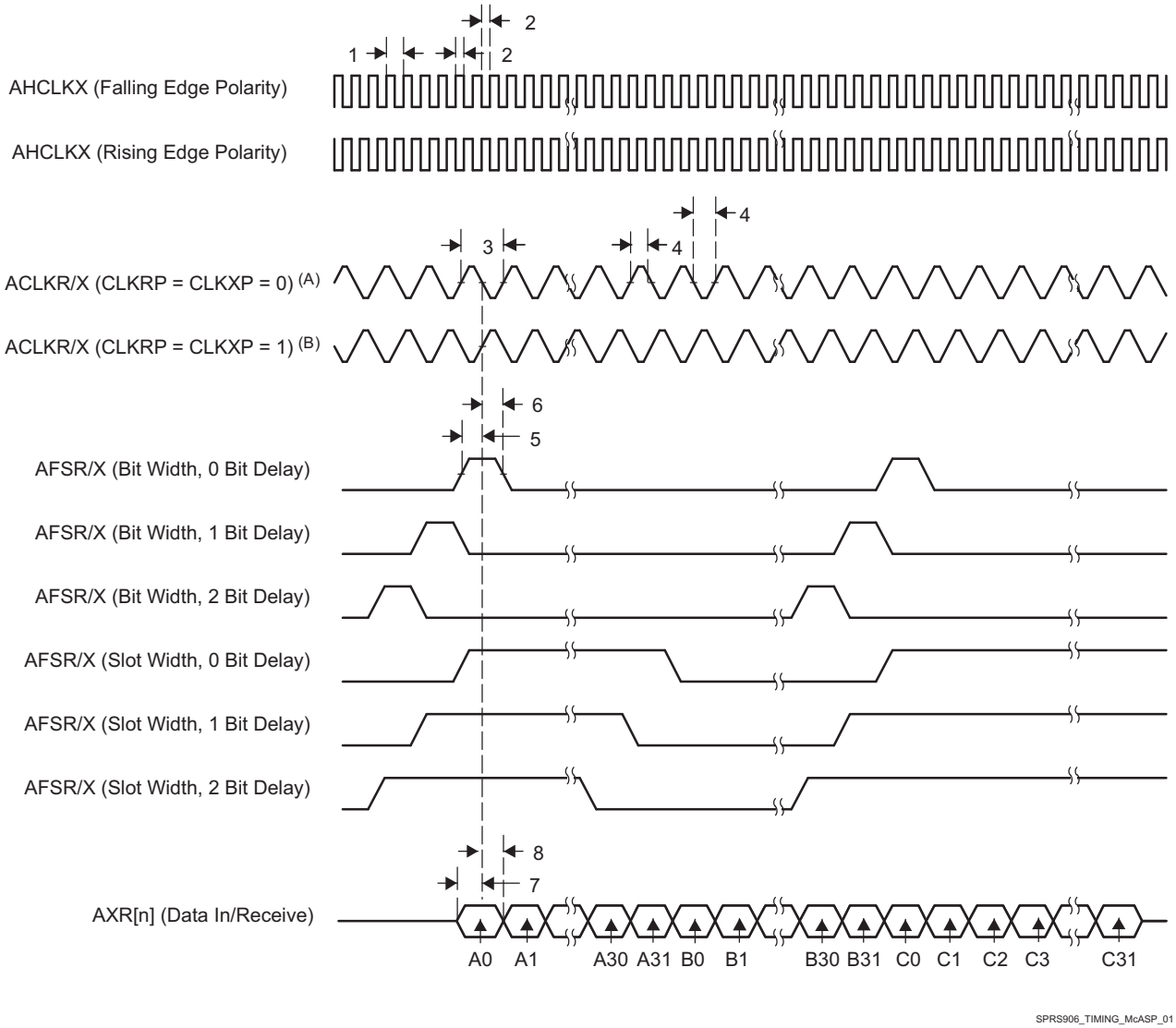
Table 5-73. Timing Requirements for McASP3/4/5/6/7/8[1]

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
1	$t_{c(AHCLKX)}$	Cycle time, AHCLKX		20		ns
2	$t_{w(AHCLKX)}$	Pulse duration, AHCLKX high or low		0.35P [2]		ns
3	$t_{c(ACLKRX)}$	Cycle time, ACLKR/X		20		ns
4	$t_{w(ACLKRX)}$	Pulse duration, ACLKR/X high or low		0.5R - 3 [3]		ns
5	$t_{su(AFSRX-ACLK)}$	Setup time, AFSR/X input valid before ACLKR/X	ACLKR/X int	19.7		ns
			ACLKR/X ext in ACLKR/X ext out	5.6		ns
6	$t_{h(ACLK-AFSRX)}$	Hold time, AFSR/X input valid after ACLKR/X	ACLKR/X int	-1.1		ns
			ACLKR/X ext in ACLKR/X ext out	2.5		ns
	$t_{su(AXR-ACLK)}$	Setup time, AXR input valid before ACLKX	ACLKX int (ASYNC=0)	20.3		ns
			ACLKR/X ext in ACLKR/X ext out	5.1		ns
8	$t_{h(ACLK-AXR)}$	Hold time, AXR input valid after ACLKX	ACLKX int (ASYNC=0)	-0.8		ns
			ACLKR/X ext in ACLKR/X ext out	2.5		ns

(1) ACLKR internal: ACLKRCTL.CLKRM=1, PDIR.ACLKR = 1 (NOT SUPPORTED)
ACLKR external input: ACLKRCTL.CLKRM=0, PDIR.ACLKR=0
ACLKR external output: ACLKRCTL.CLKRM=0, PDIR.ACLKR=1
ACLKX internal: ACLKXCTL.CLKXM=1, PDIR.ACLKX = 1
ACLKX external input: ACLKXCTL.CLKXM=0, PDIR.ACLKX=0
ACLKX external output: ACLKXCTL.CLKXM=0, PDIR.ACLKX=1

(2) P = AHCLKX period in ns.

(3) R = ACLKR/X period in ns.



A. For CLKRP = CLKXP = 0, the McASP transmitter is configured for rising edge (to shift data out) and the McASP receiver is configured for falling edge (to shift data in).

B. For CLKRP = CLKXP = 1, the McASP transmitter is configured for falling edge (to shift data out) and the McASP receiver is configured for rising edge (to shift data in).

Figure 5-57. McASP Input Timing

CAUTION

The I/O Timings provided in this section are valid only for some McASP usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-74, Table 5-75, Table 5-76 and Figure 5-58 present Switching Characteristics Over Recommended Operating Conditions for McASP1 to McASP8.

Table 5-74. Switching Characteristics Over Recommended Operating Conditions for McASP1[1]

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
9	$t_{c(AHCLKX)}$	Cycle time, AHCLKX		20		ns
10	$t_{w(AHCLKX)}$	Pulse duration, AHCLKX high or low		0.5P - 2.5 [2]		ns
11	$t_{c(ACLKRX)}$	Cycle time, ACLKR/X		20		ns
12	$t_{w(ACLKRX)}$	Pulse duration, ACLKR/X high or low		0.5P - 2.5 [3]		ns
13	$t_{d(ACLK-AFSXR)}$	Delay time, ACLKR/X transmit edge to AFSX/R output valid	ACLKR/X int	-0.9	6	ns
			ACLKR/X ext in ACLKR/X ext out	2	23.1	ns
14	$t_{d(ACLK-AXR)}$	Delay time, ACLKR/X transmit edge to AXR output valid	ACLKR/X int	-1.4	6	ns
			ACLKR/X ext in ACLKR/X ext out	2	24.2	ns

(1) ACLKR internal: ACLKRCTL.CLKRM=1, PDIR.ACLKR = 1
ACLKR external input: ACLKRCTL.CLKRM=0, PDIR.ACLKR=0
ACLKR external output: ACLKRCTL.CLKRM=0, PDIR.ACLKR=1
ACLKX internal: ACLKXCTL.CLKXM=1, PDIR.ACLKX = 1
ACLKX external input: ACLKXCTL.CLKXM=0, PDIR.ACLKX=0
ACLKX external output: ACLKXCTL.CLKXM=0, PDIR.ACLKX=1

(2) P = AHCLKX period in ns.

(3) R = ACLKR/X period in ns.

Table 5-75. Switching Characteristics Over Recommended Operating Conditions for McASP2 [1]

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
9	$t_{c(AHCLKX)}$	Cycle time, AHCLKX		20		ns
10	$t_{w(AHCLKX)}$	Pulse duration, AHCLKX high or low		0.5P - 2.5 [2]		ns
11	$t_{c(ACLKX)}$	Cycle time, ACLKX		20		ns
12	$t_{w(ACLKX)}$	Pulse duration, ACLKX high or low		0.5P - 2.5 [3]		ns
13	$t_{d(ACLK-AFSX)}$	Delay time, ACLKX transmit edge to AFSX output valid	ACLKX int	-1	6	ns
			ACLKX ext in ACLKX ext out	2	23.2	ns
14	$t_{d(ACLK-AXR)}$	Delay time, ACLKX transmit edge to AXR output valid	ACLKX int	-1.3	6	ns
			ACLKX ext in ACLKX ext out	2	23.7	ns

(1) ACLKX internal: ACLKXCTL.CLKXM=1, PDIR.ACLKX = 1
ACLKX external input: ACLKXCTL.CLKXM=0, PDIR.ACLKX=0
ACLKX external output: ACLKXCTL.CLKXM=0, PDIR.ACLKX=1

(2) P = AHCLKX period in ns.

(3) R = ACLKX period in ns.

Table 5-76. Switching Characteristics Over Recommended Operating Conditions for McASP3/4/5/6/7/8[1]

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
9	$t_{c(AHCLKX)}$	Cycle time, AHCLKX		20		ns
10	$t_{w(AHCLKX)}$	Pulse duration, AHCLKX high or low		0.5P - 2.5 [2]		ns
11	$t_{c(ACLKRX)}$	Cycle time, ACLKR/X		20		ns
12	$t_{w(ACLKRX)}$	Pulse duration, ACLKR/X high or low		0.5P - 2.5 [3]		ns

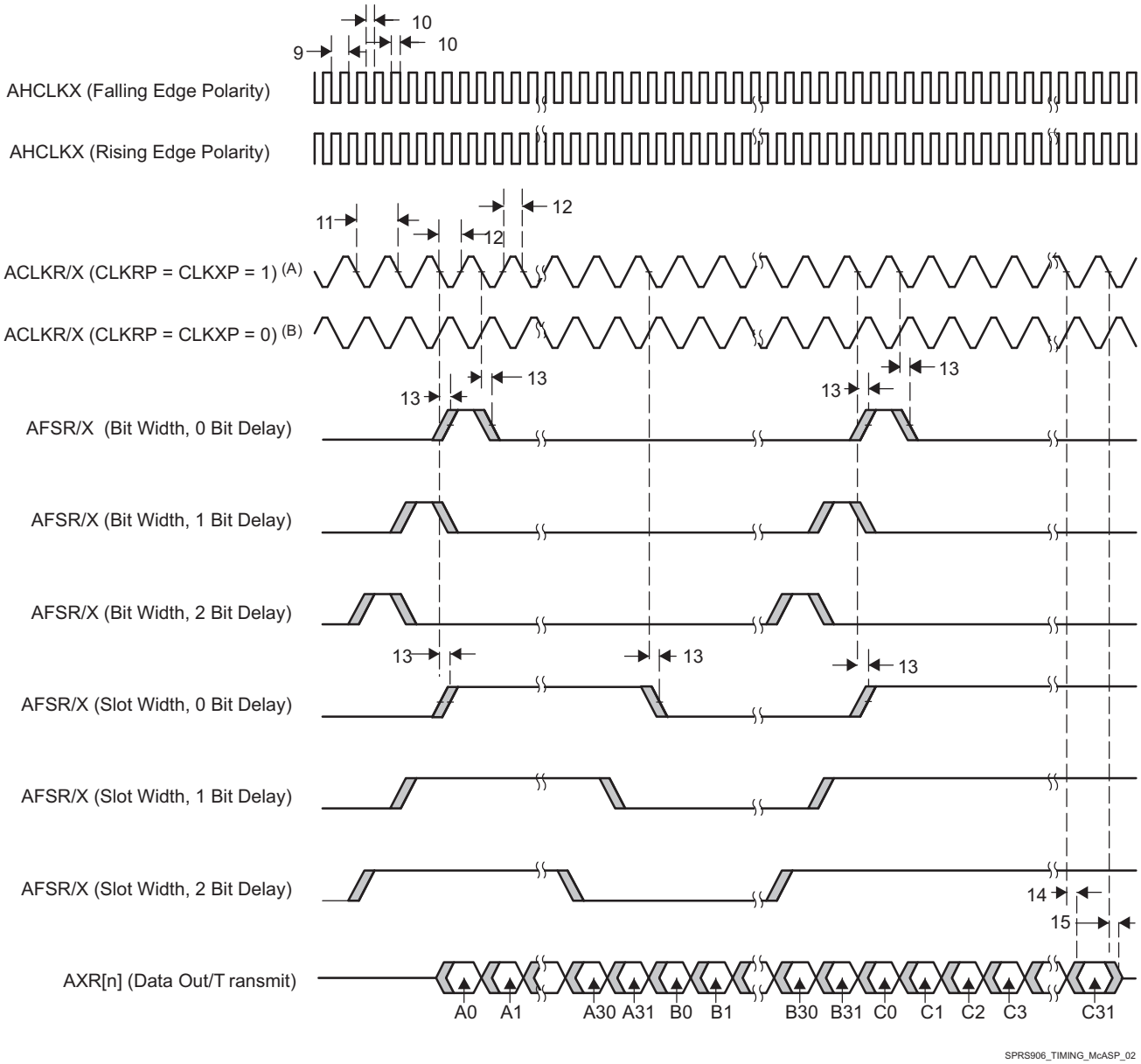
Table 5-76. Switching Characteristics Over Recommended Operating Conditions for McASP3/4/5/6/7/8[1] (continued)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
13	$t_{d(ACLK-AFSXR)}$	Delay time, ACLKR/X transmit edge to AFSX/R output valid	ACLKR/X int	-0.5	6	ns
			ACLKR/X ext in ACLKR/X ext out	1.9	24.5	ns
14	$t_{d(ACLK-AXR)}$	Delay time, ACLKR/X transmit edge to AXR output valid	ACLKR/X int	-1.4	7.1	ns
			ACLKR/X ext in ACLKR/X ext out	1.1	24.2	ns

(1) ACLKR internal: ACLKRCTL.CLKRM=1, PDIR.ACLKR = 1
ACLKR external input: ACLKRCTL.CLKRM=0, PDIR.ACLKR=0
ACLKR external output: ACLKRCTL.CLKRM=0, PDIR.ACLKR=1
ACLKX internal: ACLKXCTL.CLKXM=1, PDIR.ACLKX = 1
ACLKX external input: ACLKXCTL.CLKXM=0, PDIR.ACLKX=0
ACLKX external output: ACLKXCTL.CLKXM=0, PDIR.ACLKX=1

(2) P = AHCLKX period in ns.

(3) R = ACLKR/X period in ns.



A. For CLKRP = CLKXP = 1, the McASP transmitter is configured for falling edge (to shift data out) and the McASP receiver is configured for rising edge (to shift data in).

B. For CLKRP = CLKXP = 0, the McASP transmitter is configured for rising edge (to shift data out) and the McASP receiver is configured for falling edge (to shift data in).

Figure 5-58. McASP Output Timing

NOTE

To configure the desired virtual mode the user must set MODESELECT bit and DELAYMODE bitfield for each corresponding pad control register.

The pad control registers are presented in Table 4-32 and described in Device TRM, *Control Module Chapter*.

Table 5-77 through Table 5-84 explain all cases with Virtual Mode Details for McASP1/2/3/4/5/6/7/8 (see Figure 5-59 through Figure 5-66).

Table 5-77. Virtual Mode Case Details for McASP1

No.	CASE	CASE Description	Virtual Mode Settings		Notes
			Signals	Virtual Mode Value	
IP Mode : ASYNC					
1	COIFOI	CLKX / FSX: Output CLKR / FSR: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-59
			AXR(Inputs)/CLKR/FSR	MCASP1_VIRTUAL2_ASYNC_RX	
2	COIFIO	CLKX / FSR: Output CLKR / FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-60
			AXR(Inputs)/CLKR/FSR	MCASP1_VIRTUAL2_ASYNC_RX	
3	CIOFIO	CLKR / FSR: Output CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP1_VIRTUAL2_ASYNC_RX	See Figure 5-61
			AXR(Inputs)/CLKR/FSR	Default (No Virtual Mode)	
4	CIOFOI	CLKR / FSX: Output CLKX / FSR: Input	AXR(Outputs)/CLKX/FSX	MCASP1_VIRTUAL2_ASYNC_RX	See Figure 5-62
			AXR(Inputs)/CLKR/FSR	Default (No Virtual Mode)	
IP Mode : SYNC (CLKR / FSR internally generated from CLKX / FSX)					
5	CO-FO-	CLKX / FSX: Output	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-63
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	
6	CI-FO-	FSX: Output CLKX: Input	AXR(Outputs)/CLKX/FSX	MCASP1_VIRTUAL1_SYNC_RX	See Figure 5-64
			AXR(Inputs)/CLKX/FSX	MCASP1_VIRTUAL1_SYNC_RX	
7	CI-FI-	CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP1_VIRTUAL1_SYNC_RX	See Figure 5-65
			AXR(Inputs)/CLKX/FSX	MCASP1_VIRTUAL1_SYNC_RX	
8	CO-FI-	CLKX: Output FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-66
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	

Table 5-78. Virtual Mode Case Details for McASP2

No.	CASE	CASE Description	Virtual Mode Settings		Notes
			Signals	Virtual Mode Value	
IP Mode : SYNC (CLKR / FSR internally generated from CLKX / FSX)					
5	CO-FO-	CLKX / FSX: Output	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-63
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	
6	CI-FO-	FSX: Output CLKX: Input	AXR(Outputs)/CLKX/FSX	MCASP2_VIRTUAL3_SYNC_RX	See Figure 5-64
			AXR(Inputs)/CLKX/FSX	MCASP2_VIRTUAL3_SYNC_RX	
7	CI-FI-	CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP2_VIRTUAL3_SYNC_RX(1)	See Figure 5-65
			AXR(Inputs)/CLKX/FSX	MCASP2_VIRTUAL3_SYNC_RX(1)	
			AXR(Inputs)/CLKX/FSX	MCASP2_VIRTUAL1_SYNC_RX_80M(2)	
8	CO-FI-	CLKX: Output FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-66
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	

(1) Used up to 50MHz. Should also be used in a CI-FI- mixed case where AXR operate as both inputs and outputs (that is, AXR are bidirectional).

(2) Used in 80MHz input only mode when AXR, CLKX and FSX are all inputs.

Table 5-79. Virtual Mode Case Details for McASP3

No.	CASE	CASE Description	Virtual Mode Settings		Notes
			Signals	Virtual Mode Value	
IP Mode : ASYNC					
1	COIFOI	CLKX / FSX: Output CLKR / FSR: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-59
			AXR(Inputs)/CLKR/FSR	MCASP3_VIRTUAL2_SYNC_RX	

Table 5-79. Virtual Mode Case Details for McASP3 (continued)

No.	CASE	CASE Description	Virtual Mode Settings		Notes
			Signals	Virtual Mode Value	
2	COIFIO	CLKX / FSR: Output CLKR / FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-60
			AXR(Inputs)/CLKR/FSR	MCASP3_VIRTUAL2_SYNC_RX	
3	CIOFIO	CLKR / FSR: Output CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP3_VIRTUAL2_SYNC_RX	See Figure 5-61
			AXR(Inputs)/CLKR/FSR	MCASP3_VIRTUAL2_SYNC_RX	
4	CIOFOI	CLKR / FSX: Output CLKX / FSR: Input	AXR(Outputs)/CLKX/FSX	MCASP3_VIRTUAL2_SYNC_RX	See Figure 5-62
			AXR(Inputs)/CLKR/FSR	MCASP3_VIRTUAL2_SYNC_RX	
IP Mode : SYNC (CLKR / FSR internally generated from CLKX / FSX)					
5	CO-FO-	CLKX / FSX: Output	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-63
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	
6	CI-FO-	FSX: Output CLKX: Input	AXR(Outputs)/CLKX/FSX	MCASP3_VIRTUAL2_SYNC_RX	See Figure 5-64
			AXR(Inputs)/CLKX/FSX	MCASP3_VIRTUAL2_SYNC_RX	
7	CI-FI-	CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP3_VIRTUAL2_SYNC_RX	See Figure 5-65
			AXR(Inputs)/CLKX/FSX	MCASP3_VIRTUAL2_SYNC_RX	
8	CO-FI-	CLKX: Output FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-66
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	

Table 5-80. Virtual Mode Case Details for McASP4

No.	CASE	CASE Description	Virtual Mode Settings		Notes
			Signals	Virtual Mode Value	
IP Mode : ASYNC					
1	COIFOI	CLKX / FSX: Output CLKR / FSR: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-59
			AXR(Inputs)/CLKR/FSR	MCASP4_VIRTUAL1_SYNC_RX	
2	COIFIO	CLKX / FSR: Output CLKR / FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-60
			AXR(Inputs)/CLKR/FSR	MCASP4_VIRTUAL1_SYNC_RX	
3	CIOFIO	CLKR / FSR: Output CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP4_VIRTUAL1_SYNC_RX	See Figure 5-61
			AXR(Inputs)/CLKR/FSR	MCASP4_VIRTUAL1_SYNC_RX	
4	CIOFOI	CLKR / FSX: Output CLKX / FSR: Input	AXR(Outputs)/CLKX/FSX	MCASP4_VIRTUAL1_SYNC_RX	See Figure 5-62
			AXR(Inputs)/CLKR/FSR	MCASP4_VIRTUAL1_SYNC_RX	
IP Mode : SYNC (CLKR / FSR internally generated from CLKX / FSX)					
5	CO-FO-	CLKX / FSX: Output	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-63
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	
6	CI-FO-	FSX: Output CLKX: Input	AXR(Outputs)/CLKX/FSX	MCASP4_VIRTUAL1_SYNC_RX	See Figure 5-64
			AXR(Inputs)/CLKX/FSX	MCASP4_VIRTUAL1_SYNC_RX	
7	CI-FI-	CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP4_VIRTUAL1_SYNC_RX	See Figure 5-65
			AXR(Inputs)/CLKX/FSX	MCASP4_VIRTUAL1_SYNC_RX	
8	CO-FI-	CLKX: Output FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-66
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	

Table 5-81. Virtual Mode Case Details for McASP5

No.	CASE	CASE Description	Virtual Mode Settings		Notes
			Signals	Virtual Mode Value	
IP Mode : ASYNC					
1	COIFOI	CLKX / FSX: Output CLKR / FSR: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-59
			AXR(Inputs)/CLKR/FSR	MCASP5_VIRTUAL1_SYNC_RX	
2	COIFIO	CLKX / FSR: Output CLKR / FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-60
			AXR(Inputs)/CLKR/FSR	MCASP5_VIRTUAL1_SYNC_RX	
3	CIOFIO	CLKR / FSR: Output CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP5_VIRTUAL1_SYNC_RX	See Figure 5-61
			AXR(Inputs)/CLKR/FSR	MCASP5_VIRTUAL1_SYNC_RX	
4	CIOFOI	CLKR / FSX: Output CLKX / FSR: Input	AXR(Outputs)/CLKX/FSX	MCASP5_VIRTUAL1_SYNC_RX	See Figure 5-62
			AXR(Inputs)/CLKR/FSR	MCASP5_VIRTUAL1_SYNC_RX	
IP Mode : SYNC (CLKR / FSR internally generated from CLKX / FSX)					
5	CO-FO-	CLKX / FSX: Output	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-63
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	
6	CI-FO-	FSX: Output CLKX: Input	AXR(Outputs)/CLKX/FSX	MCASP5_VIRTUAL1_SYNC_RX	See Figure 5-64
			AXR(Inputs)/CLKX/FSX	MCASP5_VIRTUAL1_SYNC_RX	
7	CI-FI-	CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP5_VIRTUAL1_SYNC_RX	See Figure 5-65
			AXR(Inputs)/CLKX/FSX	MCASP5_VIRTUAL1_SYNC_RX	
8	CO-FI-	CLKX: Output FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-66
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	

Table 5-82. Virtual Mode Case Details for McASP6

No.	CASE	CASE Description	Virtual Mode Settings		Notes
			Signals	Virtual Mode Value	
IP Mode : ASYNC					
1	COIFOI	CLKX / FSX: Output CLKR / FSR: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-59
			AXR(Inputs)/CLKR/FSR	MCASP6_VIRTUAL1_SYNC_RX	
2	COIFIO	CLKX / FSR: Output CLKR / FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-60
			AXR(Inputs)/CLKR/FSR	MCASP6_VIRTUAL1_SYNC_RX	
3	CIOFIO	CLKR / FSR: Output CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP6_VIRTUAL1_SYNC_RX	See Figure 5-61
			AXR(Inputs)/CLKR/FSR	MCASP6_VIRTUAL1_SYNC_RX	
4	CIOFOI	CLKR / FSX: Output CLKX / FSR: Input	AXR(Outputs)/CLKX/FSX	MCASP6_VIRTUAL1_SYNC_RX	See Figure 5-62
			AXR(Inputs)/CLKR/FSR	MCASP6_VIRTUAL1_SYNC_RX	
IP Mode : SYNC (CLKR / FSR internally generated from CLKX / FSX)					
5	CO-FO-	CLKX / FSX: Output	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-63
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	
6	CI-FO-	FSX: Output CLKX: Input	AXR(Outputs)/CLKX/FSX	MCASP6_VIRTUAL1_SYNC_RX	See Figure 5-64
			AXR(Inputs)/CLKX/FSX	MCASP6_VIRTUAL1_SYNC_RX	
7	CI-FI-	CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP6_VIRTUAL1_SYNC_RX	See Figure 5-65
			AXR(Inputs)/CLKX/FSX	MCASP6_VIRTUAL1_SYNC_RX	
8	CO-FI-	CLKX: Output FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-66
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	

Table 5-83. Virtual Mode Case Details for McASP7

No.	CASE	CASE Description	Virtual Mode Settings		Notes
			Signals	Virtual Mode Value	
IP Mode : ASYNC					
1	COIFOI	CLKX / FSX: Output CLKR / FSR: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-59
			AXR(Inputs)/CLKR/FSR	MCASP7_VIRTUAL2_SYNC_RX	
2	COIFIO	CLKX / FSR: Output CLKR / FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-60
			AXR(Inputs)/CLKR/FSR	MCASP7_VIRTUAL2_SYNC_RX	
3	CIOFIO	CLKR / FSR: Output CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP7_VIRTUAL2_SYNC_RX	See Figure 5-61
			AXR(Inputs)/CLKR/FSR	MCASP7_VIRTUAL2_SYNC_RX	
4	CIOFOI	CLKR / FSX: Output CLKX / FSR: Input	AXR(Outputs)/CLKX/FSX	MCASP7_VIRTUAL2_SYNC_RX	See Figure 5-62
			AXR(Inputs)/CLKR/FSR	MCASP7_VIRTUAL2_SYNC_RX	
IP Mode : SYNC (CLKR / FSR internally generated from CLKX / FSX)					
5	CO-FO-	CLKX / FSX: Output	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-63
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	
6	CI-FO-	FSX: Output CLKX: Input	AXR(Outputs)/CLKX/FSX	MCASP7_VIRTUAL2_SYNC_RX	See Figure 5-64
			AXR(Inputs)/CLKX/FSX	MCASP7_VIRTUAL2_SYNC_RX	
7	CI-FI-	CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP7_VIRTUAL2_SYNC_RX	See Figure 5-65
			AXR(Inputs)/CLKX/FSX	MCASP7_VIRTUAL2_SYNC_RX	
8	CO-FI-	CLKX: Output FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-66
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	

Table 5-84. Virtual Mode Case Details for McASP8

No.	CASE	CASE Description	Virtual Mode Settings		Notes
			Signals	Virtual Mode Value	
IP Mode : ASYNC					
1	COIFOI	CLKX / FSX: Output CLKR / FSR: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-59
			AXR(Inputs)/CLKR/FSR	MCASP8_VIRTUAL1_SYNC_RX	
2	COIFIO	CLKX / FSR: Output CLKR / FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-60
			AXR(Inputs)/CLKR/FSR	MCASP8_VIRTUAL1_SYNC_RX	
3	CIOFIO	CLKR / FSR: Output CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP8_VIRTUAL1_SYNC_RX	See Figure 5-61
			AXR(Inputs)/CLKR/FSR	MCASP8_VIRTUAL1_SYNC_RX	
4	CIOFOI	CLKR / FSX: Output CLKX / FSR: Input	AXR(Outputs)/CLKX/FSX	MCASP8_VIRTUAL1_SYNC_RX	See Figure 5-62
			AXR(Inputs)/CLKR/FSR	MCASP8_VIRTUAL1_SYNC_RX	
IP Mode : SYNC (CLKR / FSR internally generated from CLKX / FSX)					
5	CO-FO-	CLKX / FSX: Output	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-63
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	
6	CI-FO-	FSX: Output CLKX: Input	AXR(Outputs)/CLKX/FSX	MCASP8_VIRTUAL1_SYNC_RX	See Figure 5-64
			AXR(Inputs)/CLKX/FSX	MCASP8_VIRTUAL1_SYNC_RX	
7	CI-FI-	CLKX / FSX: Input	AXR(Outputs)/CLKX/FSX	MCASP8_VIRTUAL1_SYNC_RX	See Figure 5-65
			AXR(Inputs)/CLKX/FSX	MCASP8_VIRTUAL1_SYNC_RX	
8	CO-FI-	CLKX: Output FSX: Input	AXR(Outputs)/CLKX/FSX	Default (No Virtual Mode)	See Figure 5-66
			AXR(Inputs)/CLKX/FSX	Default (No Virtual Mode)	

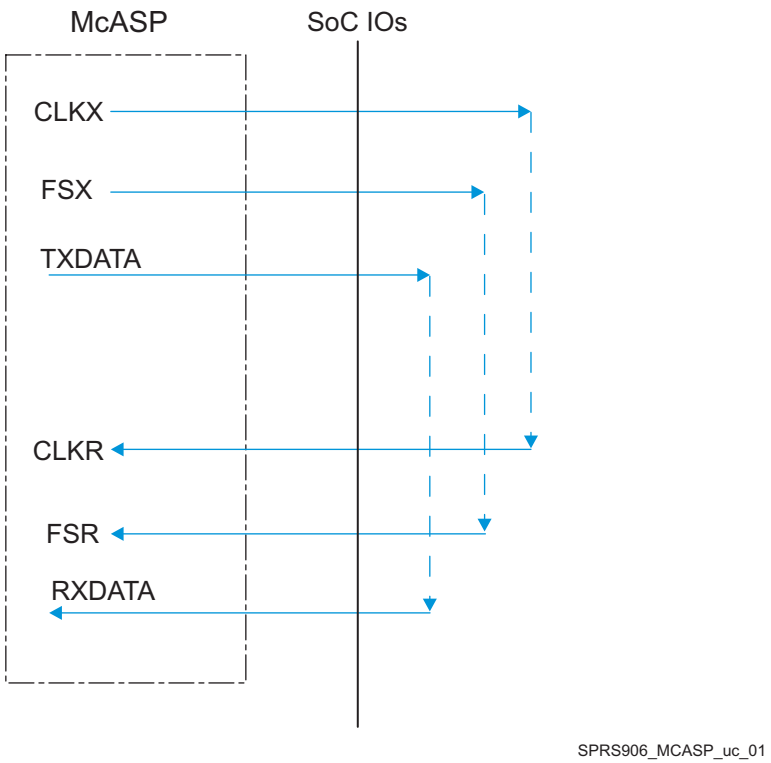


Figure 5-59. McASP1-8 COIFOI – ASYNC Mode

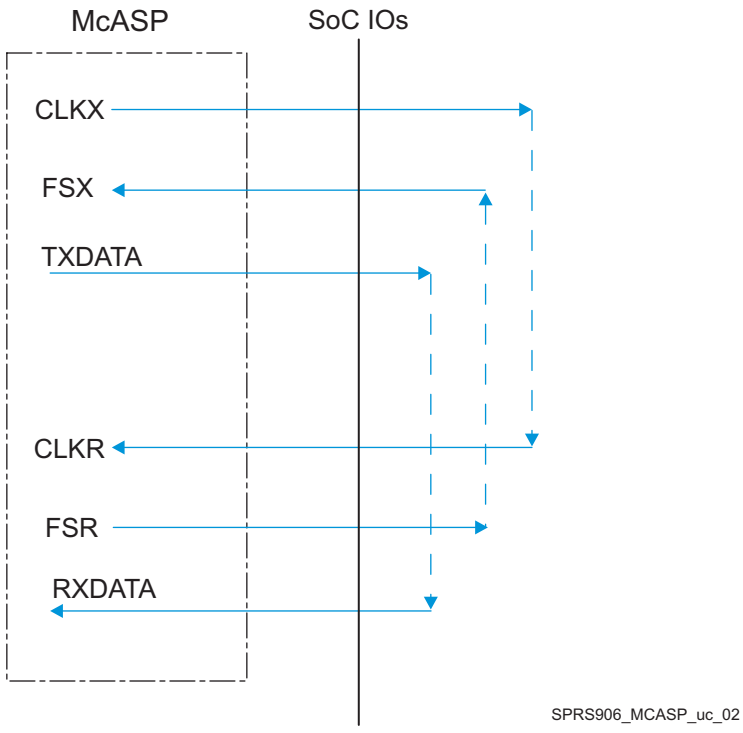


Figure 5-60. McASP1-8 COIFIO – ASYNC Mode

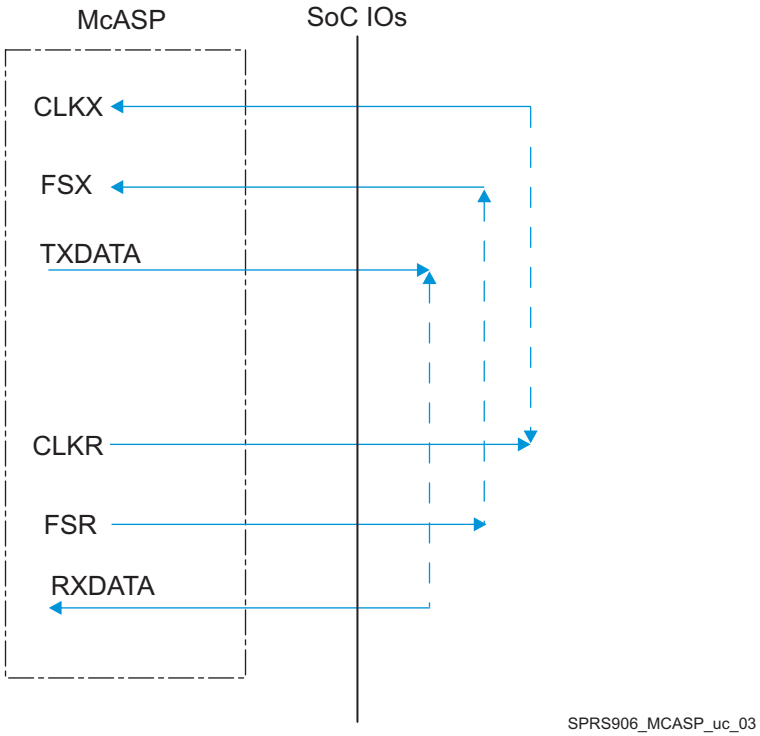


Figure 5-61. McASP1-8 CIOFIO – ASYNC Mode

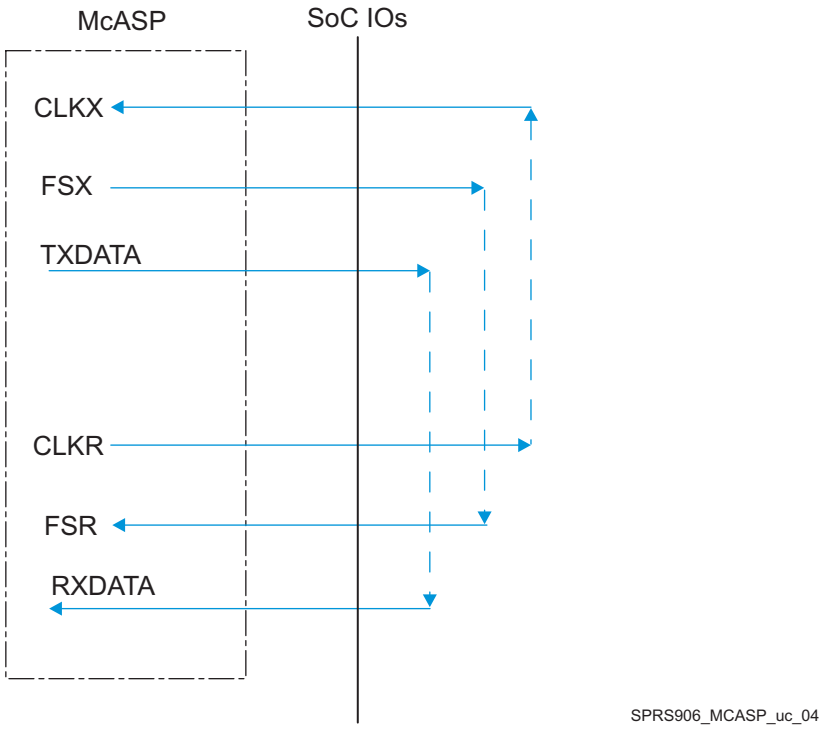


Figure 5-62. McASP1-8 CIOFOI – ASYNC Mode

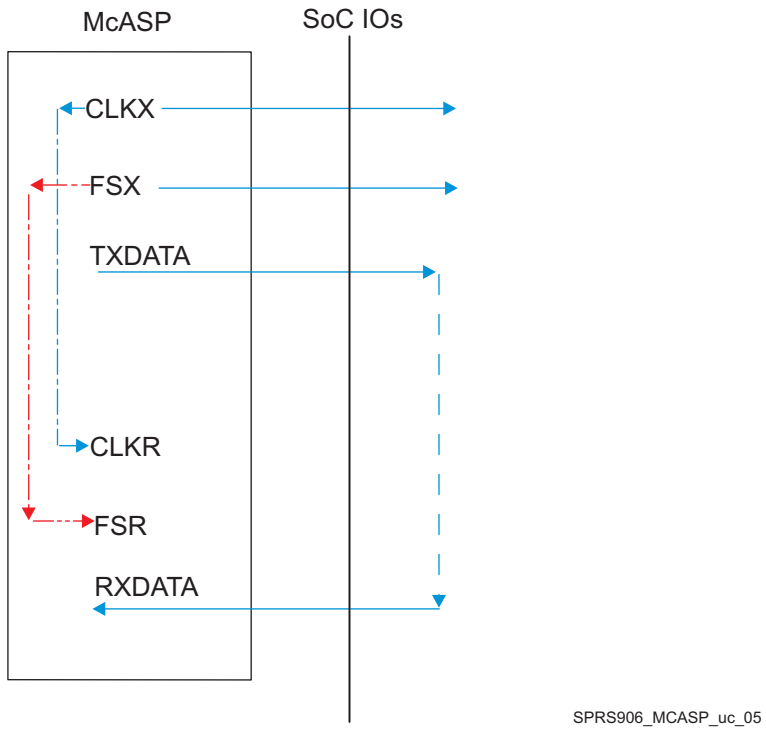


Figure 5-63. McASP1-8 CO-FO- – SYNC Mode

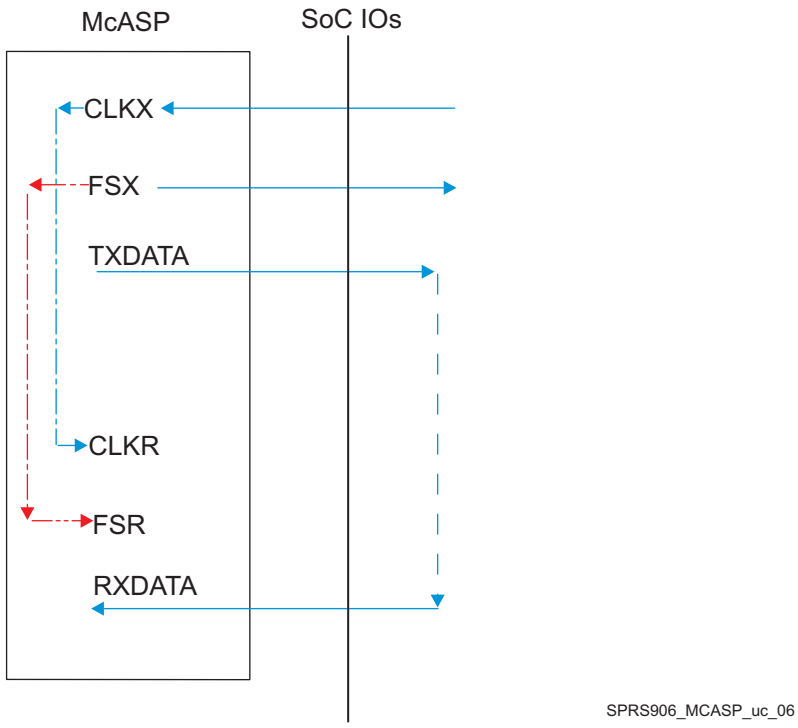


Figure 5-64. McASP1-8 CI-FO- – SYNC Mode

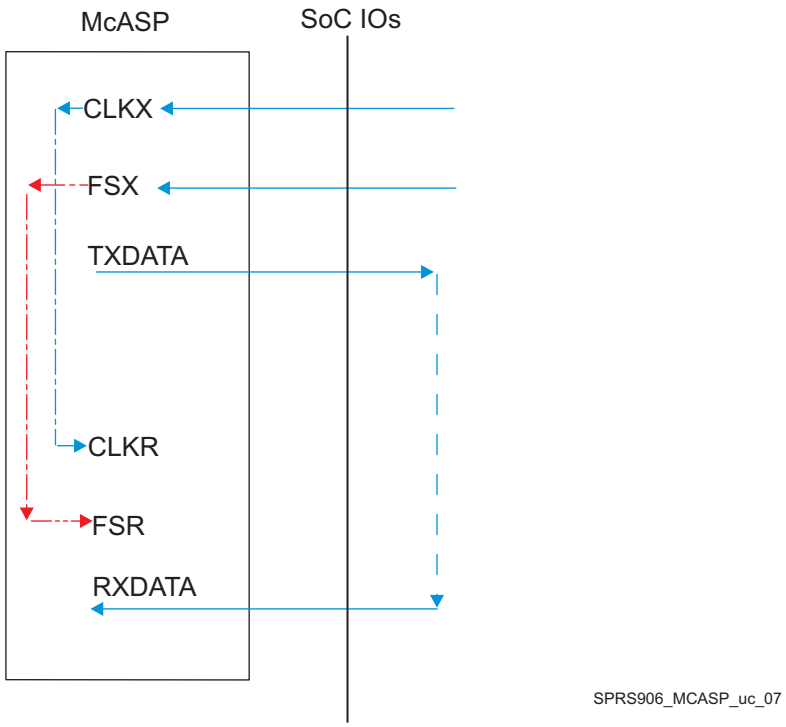


Figure 5-65. McASP1-8 CI-FI- – SYNC Mode

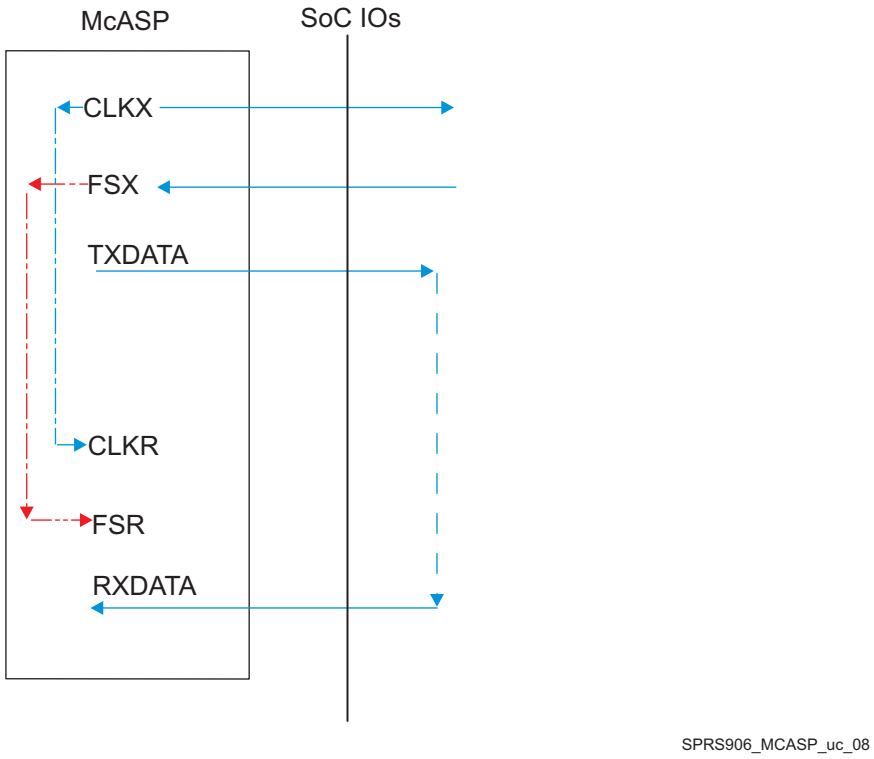


Figure 5-66. McASP1-8 CO-FI- – SYNC Mode

Virtual IO Timings Modes must be used to ensure some IO timings for McASP1. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Virtual IO Timings Modes. See Table 5-85 *Virtual Functions Mapping for McASP1* for a definition of the Virtual modes.

Table 5-85 presents the values for DELAYMODE bitfield.

Table 5-85. Virtual Functions Mapping for McASP1

BALL	BALL NAME	Delay Mode Value		MUXMODE		
		MCASP1_VIRTUAL1_SYNC_RX	MCASP1_VIRTUAL2_ASYNC_RX	0	1	2
C16	mcasp1_aclkx	15	14	mcasp1_aclkx		
H21	gpio6_14	14	13		mcasp1_axr8	
E17	mcasp1_axr13	15	14	mcasp1_axr13		
A15	mcasp1_axr4	14	13	mcasp1_axr4		
H24	xref_clk2	14	13			mcasp1_axr6
B17	mcasp1_axr9	15	14	mcasp1_axr9		
A16	mcasp1_axr7	14	13	mcasp1_axr7		
A19	mcasp1_axr12	15	14	mcasp1_axr12		
K23	gpio6_16	14	13		mcasp1_axr10	
K22	gpio6_15	14	13		mcasp1_axr9	
H25	xref_clk3	14	13			mcasp1_axr7
A17	mcasp1_axr6	14	13	mcasp1_axr6		
B16	mcasp1_axr10	15	14	mcasp1_axr10		
D17	mcasp1_fsr	N/A	14	mcasp1_fsr		
A18	mcasp1_axr8	15	14	mcasp1_axr8		
B18	mcasp1_axr11	15	14	mcasp1_axr11		
C14	mcasp1_axr2	14	13	mcasp1_axr2		
C17	mcasp1_fsx	15	14	mcasp1_fsx		
E16	mcasp1_axr14	15	14	mcasp1_axr14		
F16	mcasp1_axr15	15	14	mcasp1_axr15		
B14	mcasp1_axr1	15	14	mcasp1_axr1		
D16	mcasp1_aclkr	N/A	14	mcasp1_aclkr		
A14	mcasp1_axr5	14	13	mcasp1_axr5		
J24	xref_clk1	15	14			mcasp1_axr5
D14	mcasp1_axr0	15	14	mcasp1_axr0		
B15	mcasp1_axr3	14	13	mcasp1_axr3		
J25	xref_clk0	15	14			mcasp1_axr4

Virtual IO Timings Modes must be used to ensure some IO timings for McASP2. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Virtual IO Timings Modes. See Table 5-86 *Virtual Functions Mapping for McASP2* for a definition of the Virtual modes.

Table 5-86 presents the values for DELAYMODE bitfield.

Table 5-86. Virtual Functions Mapping for McASP2

BALL	BALL NAME	Delay Mode Value				MUXMODE		
		MCASP2_VIRTUAL1 _SYNC_RX_80M	MCASP2_VIRTUAL2 _ASYNC_RX	MCASP2_VIRTUAL3 _SYNC_RX	MCASP2_VIRTUAL4 _ASYNC_RX_80M	0	1	2
B22	mcasp3_axr0	15	14	10	9			mcasp2_axr14
D20	mcasp2_axr6	14	13	12	11	mcasp2_axr6		
C19	mcasp2_axr5	14	13	12	11	mcasp2_axr5		
D19	mcasp2_fsx	15	14	10	9	mcasp2_fsx		
H24	xref_clk2	12	11	10	9		mcasp2_axr10	
B21	mcasp2_axr3	15	14	10	9	mcasp2_axr3		
A22	mcasp3_aclkx	15	14	10	9			mcasp2_axr12
E19	mcasp2_aclkx	15	14	10	9	mcasp2_aclkx		
C20	mcasp2_axr7	14	13	12	11	mcasp2_axr7		
H25	xref_clk3	12	11	10	9		mcasp2_axr11	
B23	mcasp3_axr1	15	14	10	8			mcasp2_axr15
A23	mcasp3_fsx	15	14	10	9			mcasp2_axr13
A21	mcasp2_axr2	15	14	10	9	mcasp2_axr2		
B20	mcasp2_axr4	14	13	12	11	mcasp2_axr4		
J24	xref_clk1	10	9	8	6		mcasp2_axr9	
B19	mcasp2_axr1	14	13	12	11	mcasp2_axr1		
A20	mcasp2_axr0	14	13	12	11	mcasp2_axr0		
J25	xref_clk0	10	9	8	6		mcasp2_axr8	

Virtual IO Timings Modes must be used to ensure some IO timings for McASP3/4/5/6/7/8. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Virtual IO Timings Modes. See Table 5-87 *Virtual Functions Mapping for McASP3/4/5/6/7/8* for a definition of the Virtual modes.

Table 5-87 presents the values for DELAYMODE bitfield.

Table 5-87. Virtual Functions Mapping for McASP3/4/5/6/7/8

BALL	BALL NAME	Delay Mode Value	MUXMODE		
			0	1	2
MCASP3_VIRTUAL2_SYNC_RX					
B21	mcasp2_axr3	8		mcasp3_axr3	
A22	mcasp3_aclkx	8	mcasp3_aclkx	mcasp3_aclkr	
B22	mcasp3_axr0	8	mcasp3_axr0		
B23	mcasp3_axr1	6	mcasp3_axr1		
A23	mcasp3_fsx	8	mcasp3_fsx	mcasp3_fsr	
A21	mcasp2_axr2	8		mcasp3_axr2	
MCASP4_VIRTUAL1_SYNC_RX					
B25	mcasp4_fsx	14	mcasp4_fsx	mcasp4_fsr	
C23	mcasp4_aclkx	14	mcasp4_aclkx	mcasp4_aclkr	
A24	mcasp4_axr0	14	mcasp4_axr0		
D23	mcasp4_axr1	14	mcasp4_axr1		
A14	mcasp1_axr5	12		mcasp4_axr3	
A15	mcasp1_axr4	12		mcasp4_axr2	
MCASP5_VIRTUAL1_SYNC_RX					
AC3	mcasp5_aclkx	14	mcasp5_aclkx	mcasp5_aclkr	
U6	mcasp5_fsx	14	mcasp5_fsx	mcasp5_fsr	
AC4	mcasp5_axr1	14	mcasp5_axr1		
A17	mcasp1_axr6	12		mcasp5_axr2	
AA5	mcasp5_axr0	14	mcasp5_axr0		
A16	mcasp1_axr7	12		mcasp5_axr3	
MCASP6_VIRTUAL1_SYNC_RX					
C14	mcasp1_axr2	12		mcasp6_axr2	
B15	mcasp1_axr3	12		mcasp6_axr3	
B16	mcasp1_axr10	10		mcasp6_aclkx	mcasp6_aclkr
B17	mcasp1_axr9	10		mcasp6_axr1	
A18	mcasp1_axr8	10		mcasp6_axr0	
B18	mcasp1_axr11	10		mcasp6_fsx	mcasp6_fsr
MCASP7_VIRTUAL2_SYNC_RX					
A19	mcasp1_axr12	10		mcasp7_axr0	
F16	mcasp1_axr15	10		mcasp7_fsx	mcasp7_fsr
E16	mcasp1_axr14	10		mcasp7_aclkx	mcasp7_aclkr

Table 5-87. Virtual Functions Mapping for McASP3/4/5/6/7/8 (continued)

BALL	BALL NAME	Delay Mode Value	MUXMODE		
			0	1	2
E17	mcasp1_axr13	10		mcasp7_axr1	
D16	mcasp1_aclkr	13		mcasp7_axr2	
D17	mcasp1_fsr	13		mcasp7_axr3	
MCASP8_VIRTUAL1_SYNC_RX					
B20	mcasp2_axr4	10		mcasp8_axr0	
C20	mcasp2_axr7	10		mcasp8_fsx	mcasp8_fsr
D20	mcasp2_axr6	10		mcasp8_aclkx	mcasp8_aclkr
C19	mcasp2_axr5	10		mcasp8_axr1	

5.10.6.16 USB

SuperSpeed USB DRD Subsystem has four instances in the device providing the following functions:

- USB1: SuperSpeed (SS) USB 3.0 Dual-Role-Device (DRD) subsystem with integrated SS (USB3.0) PHY and HS/FS (USB2.0) PHY.
- USB2: High-Speed (HS) USB 2.0 Dual-Role-Device (DRD) subsystem with integrated HS/FS PHY.
- USB3: HS USB 2.0 Dual-Role-Device (DRD) subsystem with ULPI (SDR) interface to external HS/FS PHYs.

NOTE

For more information, see the SuperSpeed USB DRD section of the Device TRM.

5.10.6.16.1 USB1 DRD PHY

The USB1 DRD interface supports the following applications:

- USB2.0 High-Speed PHY port (1.8 V and 3.3 V): this asynchronous high-speed interface is compliant with the USB2.0 PHY standard with an internal transceiver (USB2.0 standard v2.0), for a maximum data rate of 480 Mbps.
- USB3.0 Super-Speed PHY port (1.8 V): this asynchronous differential super-speed interface is compliant with the USB3.0 RX/TX PHY standard (USB3.0 standard v1.0) for a maximum data bit rate of 5Gbps.

5.10.6.16.2 USB2 PHY

The USB2 interface supports the following applications:

- USB2.0 High-Speed PHY port (1.8 V and 3.3 V): this asynchronous high-speed interface is compliant with the USB2.0 PHY standard with an internal transceiver (USB2.0 standard v2.0), for a maximum data rate of 480 Mbps.

5.10.6.16.3 USB3 DRD ULPI—SDR—Slave Mode—12-pin Mode

TheUSB3 DRD interfaces support the following application:

- USB ULPI port: this synchronous interface is compliant with the USB2.0 ULPI SDR standard (UTMI+ v1.22), for alternative off-chip USB2.0 PHY interface; that is, with external transceiver with a maximum frequency of 60 MHz (synchronous slave mode, SDR, 12-pin, 8-data-bit).

NOTE

The Universal Serial Bus k ULPI modules are also refered as USBk where k = 3, 4.

Table 5-88, Table 5-89 and Figure 5-67 assume testing over the recommended operating conditions and electrical characteristic conditions.

Table 5-88. Timing Requirements for ULPI SDR Slave Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
US1	$t_{c(clk)}$	Cycle time, usb_ulpi_clk period	16.66		ns
US5	$t_{su(ctrlV-clkH)}$	Setup time, usb_ulpi_dir/usb_ulpi_nxt valid before usb_ulpi_clk rising edge	6.73		ns
US6	$t_{h(clkH-ctrlV)}$	Hold time, usb_ulpi_dir/usb_ulpi_nxt valid after usb_ulpi_clk rising edge	-0.41		ns
US7	$t_{su(dV-clkH)}$	Setup time, usb_ulpi_d[7:0] valid before usb_ulpi_clk rising edge	6.73		ns
US8	$t_{h(clkH-dV)}$	Hold time, usb_ulpi_d[7:0] valid after usb_ulpi_clk rising edge	-0.41		ns

Table 5-89. Switching Characteristics for ULPI SDR Slave Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
US4	$t_{d(clkH-stpV)}$	Delay time, usb_ulpi_clk rising edge high to output usb_ulpi_stp valid	0.44	8.35	ns
US9	$t_{d(clkL-doV)}$	Delay time, usb_ulpi_clk rising edge high to output usb_ulpi_d[7:0] valid	0.44	8.35	ns

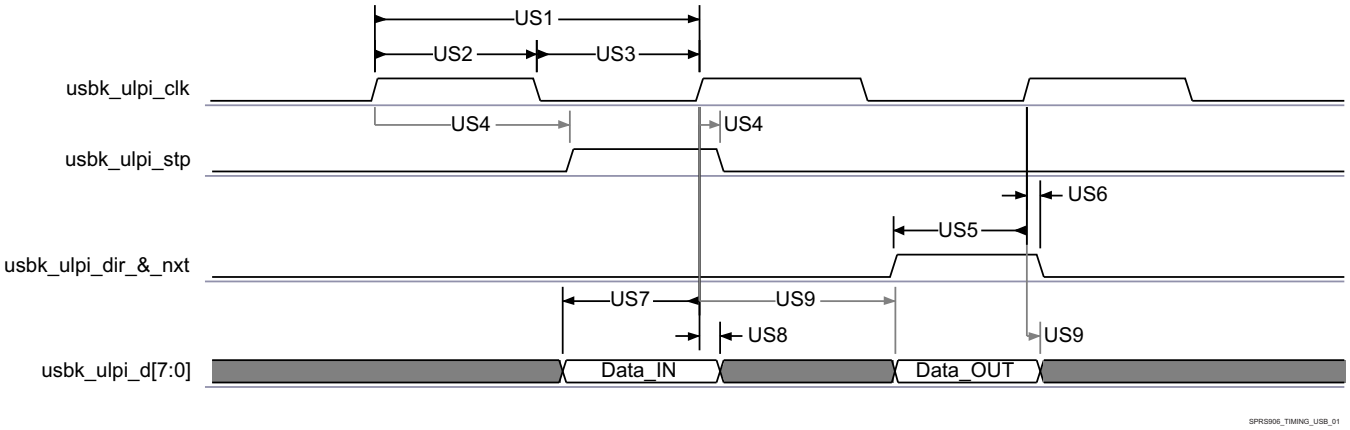


Figure 5-67. HS USB3 ULPI —SDR—Slave Mode—12-pin Mode

In Table 5-90 are presented the specific groupings of signals (IOSET) for use with USB3 signals.

Table 5-90. USB3 IOSETs

SIGNALS	IOSET2		IOSET3	
	BALL	MUX	BALL	MUX
usb3_ulpi_d7	Y5	3	N4	6
usb3_ulpi_d6	Y6	3	N3	6
usb3_ulpi_d5	Y2	3	P1	6
usb3_ulpi_d4	Y1	3	N1	6
usb3_ulpi_d3	Y4	3	P2	6
usb3_ulpi_d2	AA2	3	N2	6
usb3_ulpi_d1	AA3	3	R1	6
usb3_ulpi_d0	W2	3	R2	6
usb3_ulpi_nxt	Y3	3	P3	6
usb3_ulpi_dir	AA1	3	P4	6
usb3_ulpi_stp	AA4	3	T5	6
usb3_ulpi_clk	AB1	3	T4	6

5.10.6.17 PCIe

The device supports connections to PCIe-compliant devices via the integrated PCIe master/slave bus interface. The PCIe module is comprised of a dual-mode PCIe core and a SerDes PHY. Each PCIe subsystem controller has support for PCIe Gen-II mode (5.0 Gbps /lane) and Gen-I mode (2.5 Gbps/lane) (Single Lane and Flexible dual lane configuration).

The device PCIe supports the following features:

- 16-bit operation @250 MHz on PIPE interface (per 16-bit lane)
- Supports 2 ports x 1 lane or 1 port x 2 lanes configuration
- Single virtual channel (VC0), single traffic class (TC0)
- Single function in end-point mode

- Automatic width and speed negotiation
- Max payload: 128 byte outbound, 256 byte inbound
- Automatic credit management
- ECRC generation and checking
- Configurable BAR filtering
- Legacy interrupt reception (RC) and generation (EP)
- MSI generation and reception
- PCI Express Active State Power Management (ASPM) state L0s and L1 (with exceptions)
- All PCI Device Power Management D-states with the exception of $D3_{cold}$ / L2 state

The PCIe controller on this device conforms to the PCI Express Base 3.0 Specification, revision 1.0 and the PCI Local Bus Specification, revision 3.0.

NOTE

For more information, see the PCIe Controller section of the Device TRM.

5.10.6.18 DCAN

The device provides two DCAN interfaces for supporting distributed realtime control with a high level of security. The DCAN interfaces implement the following features:

- Supports CAN protocol version 2.0 part A, B
- Bit rates up to 1 MBit/s
- 64 message objects
- Individual identifier mask for each message object
- Programmable FIFO mode for message objects
- Programmable loop-back modes for self-test operation
- Suspend mode for debug support
- Software module reset
- Automatic bus on after Bus-Off state by a programmable 32-bit timer
- Direct access to Message RAM during test mode
- CAN Rx/Tx pins are configurable as general-purpose IO pins
- Two interrupt lines (plus additional parity-error interrupts line)
- RAM initialization
- DMA support

NOTE

For more information, see the DCAN section of the Device TRM.

NOTE

The Controller Area Network Interface x (x = 1 to 2) is also referred to as DCANx.

NOTE

Refer to the CAN Specification for calculations necessary to validate timing compliance. Jitter tolerance calculations must be performed to validate the implementation.

Table 5-91 and Table 5-92 present timing and switching characteristics for DCANx Interface.

Table 5-91. Timing Requirements for DCANx Receive

NO.	PARAMETER	DESCRIPTION	MIN	NOM	MAX	UNIT
-	$f_{(baud)}$	Maximum programmable baud rate			1	Mbps
-	$t_{d(DCANRX)}$	Delay time, DCANx_RX pin to receive shift register			15	ns

Table 5-92. Switching Characteristics Over Recommended Operating Conditions for DCANx Transmit

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
-	$f_{(baud)}$	Maximum programmable baud rate		1	Mbps
-	$t_{d(DCANTX)}$	Delay time, Transmit shift register to DCANx_TX pin(1)		23	ns

(1) These values do not include rise/fall times of the output buffer.

5.10.6.19 GMAC_SW

The three-port gigabit ethernet switch subsystem (GMAC_SW) provides ethernet packet communication and can be configured as an ethernet switch. It provides the Gigabit Media Independent Interface (G/MII) in MII mode, Reduced Gigabit Media Independent Interface (RGMII), Reduced Media Independent Interface (RMII), and the Management Data Input/Output (MDIO) for physical layer device (PHY) management.

NOTE

For more information, see the Ethernet Subsystem section of the Device TRM.

NOTE

The Gigabit, Reduced and Media Independent Interface n (n = 0 to 1) are also referred to as MIIn, RMIIn and RGMIIn.

CAUTION

The I/O Timings provided in this section are valid only if signals within a single IOSET are used. The IOSETs are defined in Table 5-97, Table 5-100, Table 5-105 and Table 5-112.

CAUTION

The I/O Timings provided in this section are valid only for some GMAC usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

5.10.6.19.1 GMAC MII Timings

Table 5-93 and Figure 5-68 present timing requirements for MIIn in receive operation.

Table 5-93. Timing Requirements for miin_rxclk - MII Operation

NO.	PARAMETER	DESCRIPTION	SPEED	MIN	MAX	UNIT
1	$t_{c(RX\_CLK)}$	Cycle time, miin_rxclk	10 Mbps	400		ns
			100 Mbps	40		ns
2	$t_{w(RX\_CLKH)}$	Pulse duration, miin_rxclk high	10 Mbps	140	260	ns
			100 Mbps	14	26	ns
3	$t_{w(RX\_CLKL)}$	Pulse duration, miin_rxclk low	10 Mbps	140	260	ns
			100 Mbps	14	26	ns

Table 5-93. Timing Requirements for miin_rxclk - MII Operation (continued)

NO.	PARAMETER	DESCRIPTION	SPEED	MIN	MAX	UNIT
4	$t_{t(RX\_CLK)}$	Transition time, miin_rxclk	10 Mbps		3	ns
			100 Mbps		3	ns

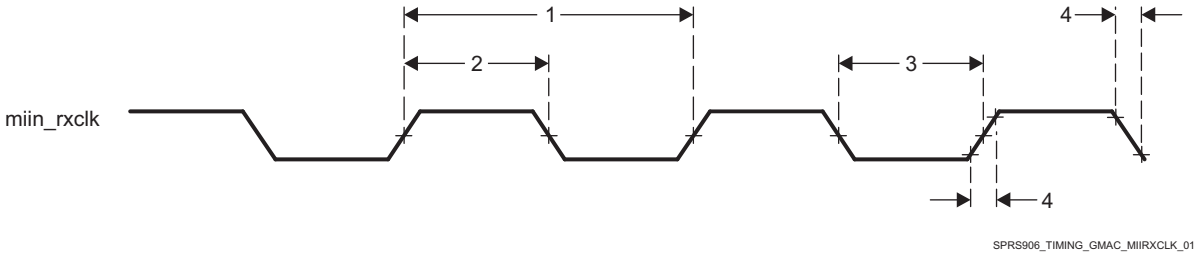


Figure 5-68. Clock Timing (GMAC Receive) - MIIn operation

Table 5-94 and Figure 5-69 present timing requirements for MIIn in transmit operation.

Table 5-94. Timing Requirements for miin_txclk - MII Operation

NO.	PARAMETER	DESCRIPTION	SPEED	MIN	MAX	UNIT
1	$t_{c(TX\_CLK)}$	Cycle time, miin_txclk	10 Mbps	400		ns
			100 Mbps	40		ns
2	$t_{w(TX\_CLKH)}$	Pulse duration, miin_txclk high	10 Mbps	140	260	ns
			100 Mbps	14	26	ns
3	$t_{w(TX\_CLKL)}$	Pulse duration, miin_txclk low	10 Mbps	140	260	ns
			100 Mbps	14	26	ns
4	$t_{t(TX\_CLK)}$	Transition time, miin_txclk	10 Mbps		3	ns
			100 Mbps		3	ns

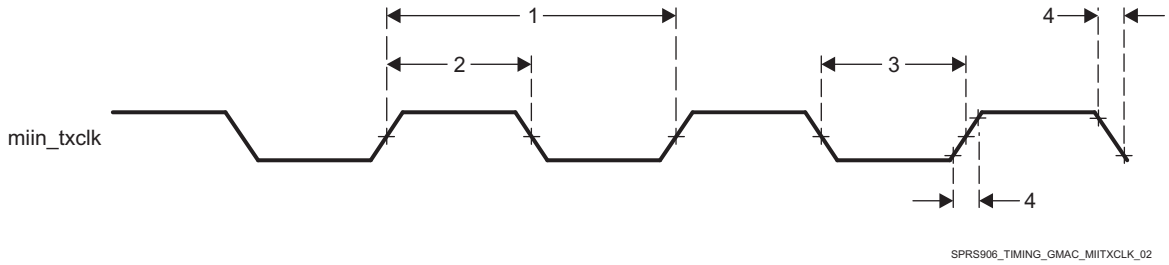


Figure 5-69. Clock Timing (GMAC Transmit) - MIIn operation

Table 5-95 and Figure 5-70 present timing requirements for GMAC MIIn Receive 10/100Mbit/s.

Table 5-95. Timing Requirements for GMAC MIIn Receive 10/100 Mbit/s

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{su(RXD-RX\_CLK)}$	Setup time, receive selected signals valid before miin_rxclk	8		ns
	$t_{su(RX\_DV-RX\_CLK)}$				
	$t_{su(RX\_ER-RX\_CLK)}$				
2	$t_{h(RX\_CLK-RXD)}$	Hold time, receive selected signals valid after miin_rxclk	8		ns
	$t_{h(RX\_CLK-RX\_DV)}$				
	$t_{h(RX\_CLK-RX\_ER)}$				

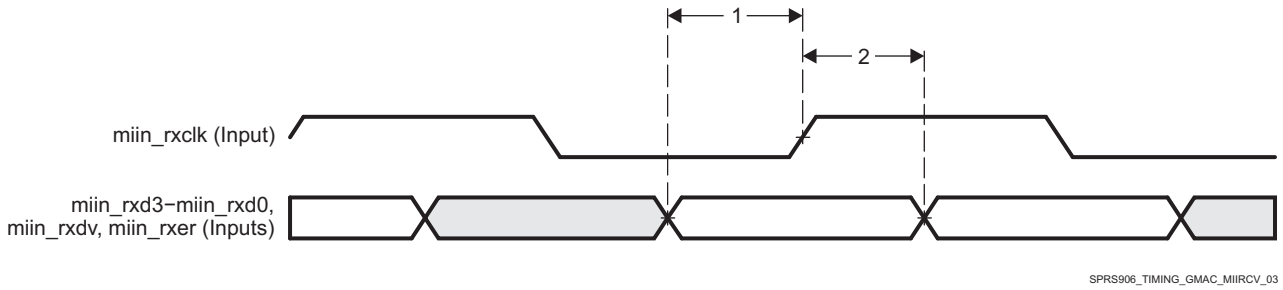


Figure 5-70. GMAC Receive Interface Timing MIIn operation

Table 5-96 and Figure 5-71 present timing requirements for GMAC MIIn Transmit 10/100Mbit/s.

Table 5-96. Switching Characteristics Over Recommended Operating Conditions for GMAC MIIn Transmit 10/100 Mbits/s

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{d(TX\_CLK-TXD)}$ $t_{d(TX\_CLK-TX\_EN)}$ $t_{d(TX\_CLK-TX\_ER)}$	Delay time, miin_txclk to transmit selected signals valid	0	25	ns

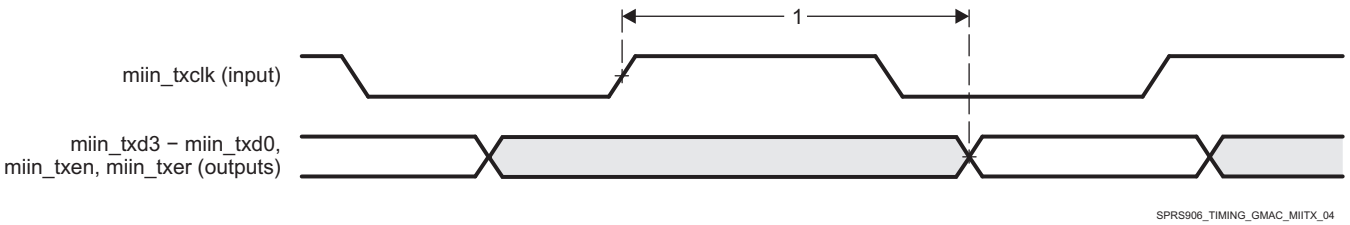


Figure 5-71. GMAC Transmit Interface Timing MIIn operation

In Table 5-97 are presented the specific groupings of signals (IOSET) for use with GMAC MII signals.

Table 5-97. GMAC MII IOSETs

SIGNALS	IOSET5		IOSET6	
	BALL	MUX	BALL	MUX
GMAC MII1				
mii1_txd3	E11	8		
mii1_txd2	A13	8		
mii1_txd1	A12	8		
mii1_txd0	B12	8		
mii1_rxd3	B10	8		
mii1_rxd2	A10	8		
mii1_rxd1	F10	8		
mii1_rxd0	E10	8		
mii1_col	E13	8		
mii1_rxer	B13	8		
mii1_txer	F11	8		
mii1_txen	D13	8		
mii1_crs	C13	8		
mii1_rxclk	B11	8		
mii1_txclk	C11	8		
mii1_rxdv	D11	8		

Table 5-97. GMAC MII IOSETs (continued)

SIGNALS	IOSET5		IOSET6	
	BALL	MUX	BALL	MUX
GMAC MII0				
mii0_txd3			P2	3
mii0_txd2			N1	3
mii0_txd1			N3	3
mii0_txd0			N4	3
mii0_rxd3			T4	3
mii0_rxd2			T5	3
mii0_rxd1			R2	3
mii0_rxd0			R1	3
mii0_txclk			N2	3
mii0_txer			L6	3
mii0_rxer			P3	3
mii0_rxdv			N5	3
mii0_crs			P4	3
mii0_col			L5	3
mii0_rxclk			N6	3
mii0_txen			P1	3

5.10.6.19.2 GMAC MDIO Interface Timings

CAUTION

The I/O Timings provided in this section are valid only for some GMAC usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-98, Table 5-98 and Figure 5-72 present timing requirements for MDIO.

Table 5-98. Timing Requirements for MDIO Input

No	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
MDIO1	$t_{c(MDC)}$	Cycle time, MDC	400		ns
MDIO2	$t_{w(MDCH)}$	Pulse Duration, MDC High	160		ns
MDIO3	$t_{w(MDCL)}$	Pulse Duration, MDC Low	160		ns
MDIO4	$t_{su(MDIO-MDC)}$	Setup time, MDIO valid before MDC High	90		ns
MDIO5	$t_{h(MDIO\_MDC)}$	Hold time, MDIO valid from MDC High	0		ns

Table 5-99. Switching Characteristics Over Recommended Operating Conditions for MDIO Output

NO	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
MDIO6	$t_{t(MDC)}$	Transition time, MDC		5	ns
MDIO7	$t_{d(MDC-MDIO)}$	Delay time, MDC low to MDIO valid	-150	150	ns

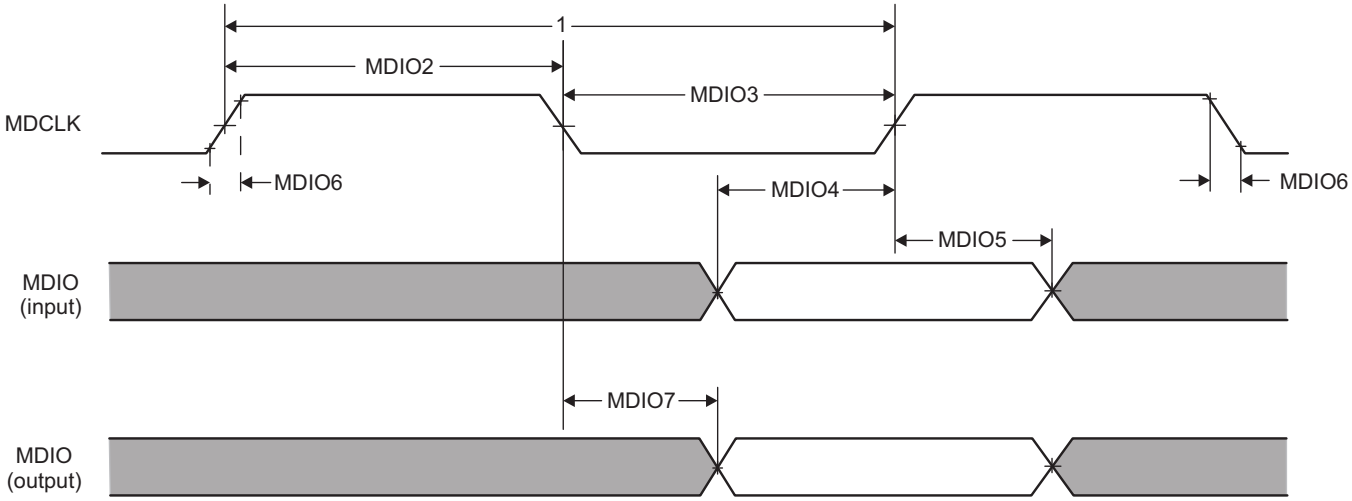


Figure 5-72. GMAC MDIO diagrams

In Table 5-100 are presented the specific groupings of signals (IOSET) for use with GMAC MDIO signals.

Table 5-100. GMAC MDIO IOSETs

SIGNALS	IOSET7		IOSET8		IOSET9		IOSET10	
	BALL	MUX	BALL	MUX	BALL	MUX	BALL	MUX
mdio_d	C10	3	L6	0	Y6	1	E25	5
mdio_mclk	D10	3	L5	0	Y5	1	E24	5

5.10.6.19.3 GMAC RMII Timings

The main reference clock REF_CLK (RMII_50MHZ_CLK) of RMII interface is internally supplied from PRCM. The source of this clock could be either externally sourced from the RMII_MHZ_50_CLK pin of the device or internally generated from DPLL_GMAC output clock GMAC_RMII_HS_CLK. Please see the PRCM chapter of the device TRM for full details about RMII reference clock.

CAUTION

The I/O Timings provided in this section are valid only for some GMAC usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-101, Table 5-102 and Figure 5-73 present timing requirements for GMAC RMIIn Receive.

Table 5-101. Timing Requirements for GMAC REF_CLK - RMII Operation

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
RMII1	$t_{c(REF\_CLK)}$	Cycle time, REF_CLK	20		ns
RMII2	$t_{w(REF\_CLKH)}$	Pulse duration, REF_CLK high	7	13	ns
RMII3	$t_{w(REF\_CLKL)}$	Pulse duration, REF_CLK low	7	13	ns
RMII4	$t_{tt(REF\_CLK)}$	Transistion time, REF_CLK		3	ns

Table 5-102. Timing Requirements for GMAC RMIIn Receive

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
RMII5	$t_{su(RXD-REF\_CLK)}$	Setup time, receive selected signals valid before REF_CLK	4		ns
	$t_{su(CRS\_DV-REF\_CLK)}$				
	$t_{su(RX\_ER-REF\_CLK)}$				
RMII6	$t_{h(REF\_CLK-RXD)}$	Hold time, receive selected signals valid after REF_CLK	2		ns
	$t_{h(REF\_CLK-CRS\_DV)}$				
	$t_{h(REF\_CLK-RX\_ER)}$				

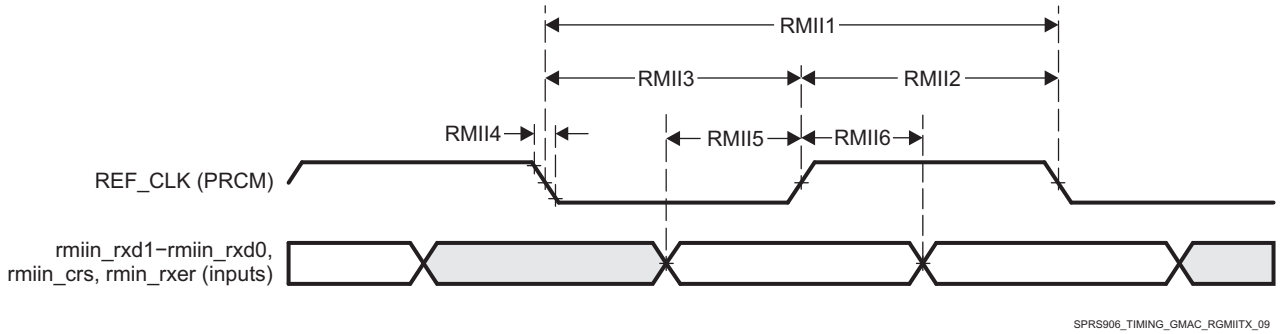


Figure 5-73. GMAC Receive Interface Timing RMIIn operation

Table 5-103, Table 5-103 and Figure 5-74 present switching characteristics for GMAC RMIIn Transmit 10/100Mbit/s.

Table 5-103. Switching Characteristics Over Recommended Operating Conditions for GMAC REF_CLK - RMII Operation

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
RMII7	$t_{c(REF\_CLK)}$	Cycle time, REF_CLK	20		ns
RMII8	$t_{w(REF\_CLKH)}$	Pulse duration, REF_CLK high	7	13	ns
RMII9	$t_{w(REF\_CLKL)}$	Pulse duration, REF_CLK low	7	13	ns
RMII10	$t_{t(REF\_CLK)}$	Transistion time, REF_CLK		3	ns

Table 5-104. Switching Characteristics Over Recommended Operating Conditions for GMAC RMIIn Transmit 10/100 Mbits/s

NO.	PARAMETER	DESCRIPTION	RMIIn	MIN	MAX	UNIT
RMII11	$t_{d(REF\_CLK-TXD)}$	Delay time, REF_CLK high to selected transmit signals valid	RMII0	2	13.5	ns
	$t_{dd(REF\_CLK-TXEN)}$					
	$t_{d(REF\_CLK-TXD)}$		RMII1	2	13.8	ns
	$t_{dd(REF\_CLK-TXEN)}$					

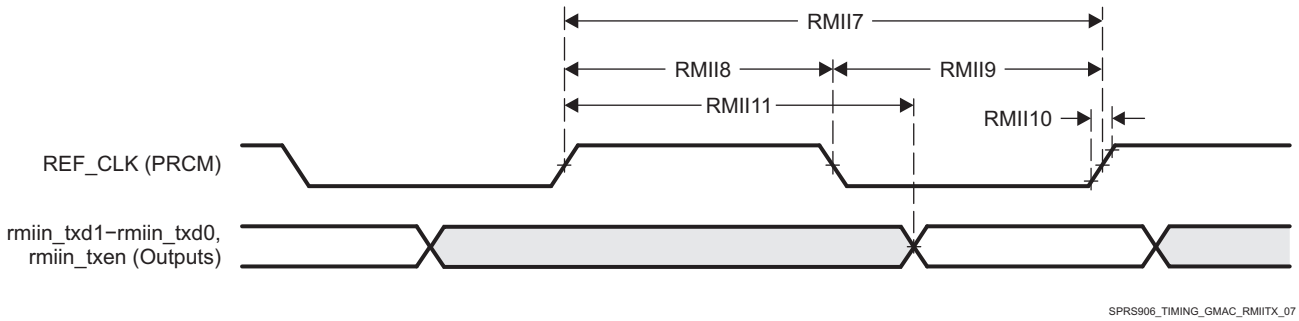


Figure 5-74. GMAC Transmit Interface Timing RMIIn Operation

In Table 5-105 are presented the specific groupings of signals (IOSET) for use with GMAC RMII signals.

Table 5-105. GMAC RMII IOSETs

SIGNALS	IOSET1		IOSET2	
	BALL	MUX	BALL	MUX
GMAC RMII1				
RMII_MHZ_50_CLK	P5	0		
rmii1_txd1	P2	2		
rmii1_txd0	N1	2		
rmii1_rxd1	T4	2		
rmii1_rxd0	T5	2		
rmii1_rxer	N6	2		
rmii1_txen	N2	2		
rmii1_crs	N5	2		
GMAC RMII0				
RMII_MHZ_50_CLK			P5	0
rmii0_txd1			N3	1
rmii0_txd0			N4	1
rmii0_rxd1			R2	1
rmii0_rxd0			R1	1
rmii0_txen			P1	1
rmii0_rxer			P3	1
rmii0_crs			P4	1

Manual IO Timings Modes must be used to ensure some IO timings for GMAC. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-106 *Manual Functions Mapping for GMAC RMII0* for a definition of the Manual modes.

Table 5-106 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-106. Manual Functions Mapping for GMAC RMII0

BALL	BALL NAME	GMAC_RMII0_MANUAL1		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)		0	1
P5	RMII_MHZ_50_CLK	0	0	CFG_RMII_MHZ_50_CLK_IN	RMII_MHZ_50_CLK	
R1	rgmii0_txd0	2444	804	CFG_RGMII0_TXD0_IN		rmii0_rxd0
R2	rgmii0_txd1	2453	981	CFG_RGMII0_TXD1_IN		rmii0_rxd1
P3	rgmii0_txd2	2356	847	CFG_RGMII0_TXD2_IN		rmii0_rxer
P4	rgmii0_txd3	2415	993	CFG_RGMII0_TXD3_IN		rmii0_crs

Manual IO Timings Modes must be used to ensure some IO timings for GMAC. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-107 *Manual Functions Mapping for GMAC RMII1* for a definition of the Manual modes.

Table 5-107 list the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-107. Manual Functions Mapping for GMAC RMII1

BALL	BALL NAME	GMAC_RMII1_MANUAL1		CFG REGISTER	MUXMODE	
		A_DELAY (ps)	G_DELAY (ps)		0	2
P5	RMII_MHZ_50_CLK	0	0	CFG_RMII_MHZ_50_CLK_IN	RMII_MHZ_50_CLK	
T5	rgmii0_txctl	2450	909	CFG_RGMII0_TXCTL_IN		rmii1_rxd0
T4	rgmii0_txc	2327	926	CFG_RGMII0_TXC_IN		rmii1_rxd1
N6	uart3_txd	2553	443	CFG_UART3_TXD_IN		rmii1_rxer
N5	uart3_rxd	1943	1110	CFG_UART3_RXD_IN		rmii1_crs

5.10.6.19.4 GMAC RGMII Timings

CAUTION

The I/O Timings provided in this section are valid only for some GMAC usage modes when the corresponding Virtual I/O Timings or Manual I/O Timings are configured as described in the tables found in this section.

Table 5-108, Table 5-109 and Figure 5-75 present timing requirements for receive RGMIIn operation.

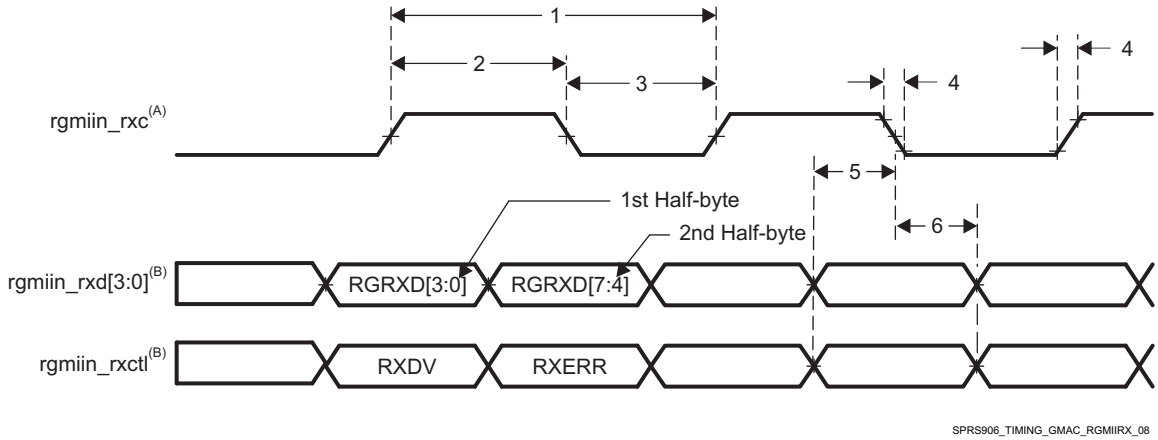
Table 5-108. Timing Requirements for rgmiin_rxc - RGMIIn Operation

NO.	PARAMETER	DESCRIPTION	SPEED	MIN	MAX	UNIT
1	$t_{c(RXC)}$	Cycle time, rgmiin_rxc	10 Mbps	360	440	ns
			100 Mbps	36	44	ns
			1000 Mbps	7.2	8.8	ns
2	$t_{w(RXCH)}$	Pulse duration, rgmiin_rxc high	10 Mbps	160	240	ns
			100 Mbps	16	24	ns
			1000 Mbps	3.6	4.4	ns
3	$t_{w(RXCL)}$	Pulse duration, rgmiin_rxc low	10 Mbps	160	240	ns
			100 Mbps	16	24	ns
			1000 Mbps	3.6	4.4	ns
4	$t_{t(RXC)}$	Transition time, rgmiin_rxc	10 Mbps		0.75	ns
			100 Mbps		0.75	ns
			1000 Mbps		0.75	ns

Table 5-109. Timing Requirements for GMAC RGMIIn Input Receive for 10/100/1000 Mbps (1)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
5	$t_{su(RXD-RXCH)}$	Setup time, receive selected signals valid before rgmiin_rxc high/low	RGMII0/1	1		ns
6	$t_{h(RXCH-RXD)}$	Hold time, receive selected signals valid after rgmiin_rxc high/low	RGMII0/1	1		ns

(1) For RGMII, receive selected signals include: rgmiin_rxd[3:0] and rgmiin_rxctl.



A. rgmiin_rxc must be externally delayed relative to the data and control pins.

B. Data and control information is received using both edges of the clocks. rgmiin_rxd[3:0] carries data bits 3-0 on the rising edge of rgmiin_rxc and data bits 7-4 on the falling edge ofrgmiin_rxc. Similarly, rgmiin_rxctl carries RXDV on rising edge of rgmiin_rxc and RXERR on falling edge of rgmiin_rxc.

Figure 5-75. GMAC Receive Interface Timing, RGMIIn operation

Table 5-110, Table 5-111 and Figure 5-76 present switching characteristics for transmit - RGMIIn for 10/100/1000Mbit/s.

Table 5-110. Switching Characteristics Over Recommended Operating Conditions for rgmiin_txctl - RGMIIn Operation for 10/100/1000 Mbit/s

NO.	PARAMETER	DESCRIPTION	SPEED	MIN	MAX	UNIT
1	$t_{c(TXC)}$	Cycle time, rgmiin_txc	10 Mbps	360	440	ns
			100 Mbps	36	44	ns
			1000 Mbps	7.2	8.8	ns
2	$t_{w(TXCH)}$	Pulse duration, rgmiin_txc high	10 Mbps	160	240	ns
			100 Mbps	16	24	ns
			1000 Mbps	3.6	4.4	ns
3	$t_{w(TXCL)}$	Pulse duration, rgmiin_txc low	10 Mbps	160	240	ns
			100 Mbps	16	24	ns
			1000 Mbps	3.6	4.4	ns
4	$t_{t(TXC)}$	Transition time, rgmiin_txc	10 Mbps		0.75	ns
			100 Mbps		0.75	ns
			1000 Mbps		0.75	ns

Table 5-111. Switching Characteristics for GMAC RGMIIn Output Transmit for 10/100/1000 Mbps (1)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
5	$t_{osu(TXD-TXC)}$	Output Setup time, transmit selected signals valid to rgmiin_txc high/low	RGMII0, Internal Delay Enabled, 1000 Mbps	1.05 (2)		ns
			RGMII0, Internal Delay Enabled, 10/100 Mbps	1.2		ns
			RGMII1, Internal Delay Enabled, 1000 Mbps	1.05 (3)		ns
			RGMII1, Internal Delay Enabled, 10/100 Mbps	1.2		ns

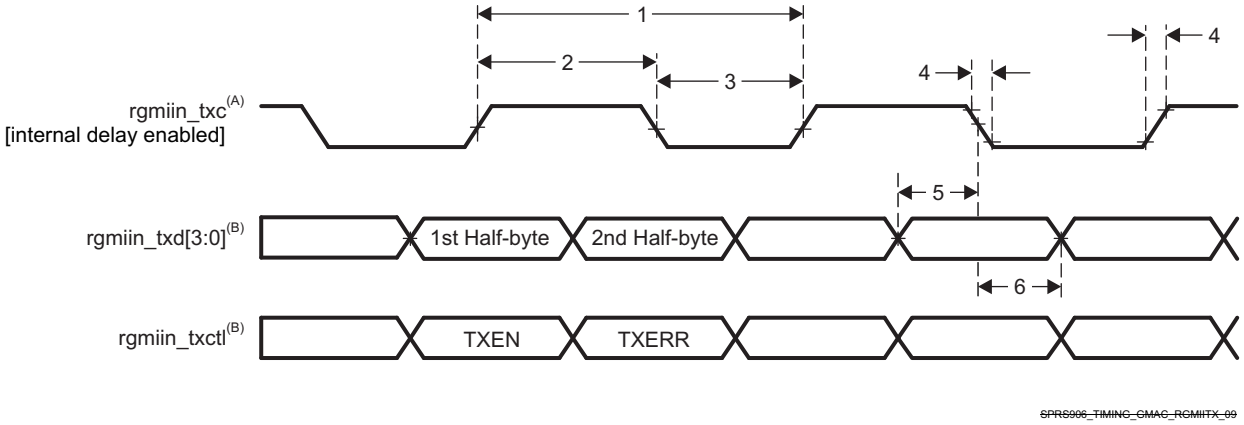
Table 5-111. Switching Characteristics for GMAC RGMIIn Output Transmit for 10/100/1000 Mbps (1) (continued)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
6	$t_{oh(TXC-TXD)}$	Output Hold time, transmit selected signals valid after rgmiin_txc high/low	RGMII0, Internal Delay Enabled, 1000 Mbps	1.05 (2)		ns
			RGMII0, Internal Delay Enabled, 10/100 Mbps	1.2		ns
			RGMII1, Internal Delay Enabled, 1000 Mbps	1.05 (3)		ns
			RGMII1, Internal Delay Enabled, 10/100 Mbps	1.2		ns

(1) For RGMII, transmit selected signals include: rgmiin_txd[3:0] and rgmiin_txctl.

(2) RGMII0 requires that the 4 data pins rgmii0_txd[3:0] and rgmii0_txctl have their board propagation delays matched within 50pS of rgmii0_txc.

(3) RGMII1 requires that the 4 data pins rgmii1_txd[3:0] and rgmii1_txctl have their board propagation delays matched within 50pS of rgmii1_txc.



A. TXC is delayed internally before being driven to the rgmiin_txc pin. This internal delay is always enabled.

B. Data and control information is transmitted using both edges of the clocks. rgmiin_txd[3:0] carries data bits 3-0 on the rising edge of rgmiin_txc and data bits 7-4 on the falling edge of rgmiin_txc. Similarly, rgmiin_txctl carries TXEN on rising edge of rgmiin_txc and TXERR of falling edge of rgmiin_txc.

Figure 5-76. GMAC Transmit Interface Timing RGMIIn operation

In Table 5-112 are presented the specific groupings of signals (IOSET) for use with GMAC RGMII signals.

Table 5-112. GMAC RGMII IOSETs

SIGNALS	IOSET3		IOSET4	
	BALL	MUX	BALL	MUX
GMAC RGMII1				
rgmii1_txd3	C11	3		
rgmii1_txd2	B12	3		
rgmii1_txd1	A12	3		
rgmii1_txd0	A13	3		
rgmii1_rxd3	B13	3		
rgmii1_rxd2	E13	3		
rgmii1_rxd1	C13	3		
rgmii1_rxd0	D13	3		
rgmii1_rxctl	F11	3		
rgmii1_txc	B11	3		
rgmii1_txctl	D11	3		

Table 5-112. GMAC RGMII IOSETs (continued)

SIGNALS	IOSET3		IOSET4	
	BALL	MUX	BALL	MUX
rgmii1_rxc	E11	3		
GMAC RGMII0				
rgmii0_txd3			P4	0
rgmii0_txd2			P3	0
rgmii0_txd1			R2	0
rgmii0_txd0			R1	0
rgmii0_rxd3			N1	0
rgmii0_rxd2			P1	0
rgmii0_rxd1			N3	0
rgmii0_rxd0			N4	0
rgmii0_txc			T4	0
rgmii0_rxctl			P2	0
rgmii0_rxc			N2	0
rgmii0_txctl			T5	0

NOTE

To configure the desired Manual IO Timing Mode the user must follow the steps described in section "*Manual IO Timing Modes*" of the Device TRM.

The associated registers to configure are listed in the **CFG REGISTER** column. For more information please see the *Control Module Chapter* in the Device TRM.

Manual IO Timings Modes must be used to ensure some IO timings for GMAC. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-113 *Manual Functions Mapping for GMAC RGMII0* for a definition of the Manual modes.

Table 5-113 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-113. Manual Functions Mapping for GMAC RGMII0

BALL	BALL NAME	GMAC_RGMII0_MANUAL1		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		0
N2	rgmii0_rxc	413	0	CFG_RGMII0_RXC_IN	rgmii0_rxc
P2	rgmii0_rxctl	27	2296	CFG_RGMII0_RXCTL_IN	rgmii0_rxctl
N4	rgmii0_rxd0	3	1721	CFG_RGMII0_RXD0_IN	rgmii0_rxd0
N3	rgmii0_rxd1	134	1786	CFG_RGMII0_RXD1_IN	rgmii0_rxd1
P1	rgmii0_rxd2	40	1966	CFG_RGMII0_RXD2_IN	rgmii0_rxd2
N1	rgmii0_rxd3	0	2057	CFG_RGMII0_RXD3_IN	rgmii0_rxd3
T4	rgmii0_txc	0	60	CFG_RGMII0_TXC_OUT	rgmii0_txc
T5	rgmii0_txctl	0	60	CFG_RGMII0_TXCTL_OUT	rgmii0_txctl
R1	rgmii0_txd0	0	60	CFG_RGMII0_TXD0_OUT	rgmii0_txd0
R2	rgmii0_txd1	0	0	CFG_RGMII0_TXD1_OUT	rgmii0_txd1
P3	rgmii0_txd2	0	60	CFG_RGMII0_TXD2_OUT	rgmii0_txd2
P4	rgmii0_txd3	0	120	CFG_RGMII0_TXD3_OUT	rgmii0_txd3

Manual IO Timings Modes must be used to ensure some IO timings for GMAC. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-114 *Manual Functions Mapping for GMAC RGMII1* for a definition of the Manual modes.

Table 5-114 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-114. Manual Functions Mapping for GMAC RGMII1

BALL	BALL NAME	GMAC_RGMII1_MANUAL1		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		3
E11	vin2a_d18	530	0	CFG_VIN2A_D18_IN	rgmii1_rxc
F11	vin2a_d19	71	1099	CFG_VIN2A_D19_IN	rgmii1_rxctl
B13	vin2a_d20	142	1337	CFG_VIN2A_D20_IN	rgmii1_rxd3
E13	vin2a_d21	114	1517	CFG_VIN2A_D21_IN	rgmii1_rxd2
C13	vin2a_d22	171	1331	CFG_VIN2A_D22_IN	rgmii1_rxd1
D13	vin2a_d23	0	1328	CFG_VIN2A_D23_IN	rgmii1_rxd0
B11	vin2a_d12	0	0	CFG_VIN2A_D12_OUT	rgmii1_txc
D11	vin2a_d13	170	0	CFG_VIN2A_D13_OUT	rgmii1_txctl
C11	vin2a_d14	150	0	CFG_VIN2A_D14_OUT	rgmii1_txd3
B12	vin2a_d15	0	0	CFG_VIN2A_D15_OUT	rgmii1_txd2
A12	vin2a_d16	60	0	CFG_VIN2A_D16_OUT	rgmii1_txd1
A13	vin2a_d17	60	0	CFG_VIN2A_D17_OUT	rgmii1_txd0

5.10.6.20 MLB

The MLBSS allows connection to a MOST (Media Oriented Systems Transport) network controller for transport of media and control data between multimedia nodes. The MLBSS supports the following features:

- 3 pin mode compliant to MediaLB Physical Layer Specification v4.0
- 6 pin mode (3 differential pairs) compliant to MediaLB Physical Layer Specification v4.0
- Supports 256/512/1024Fs in 3 pin mode and 2048Fs in 6 pin mode
- Supports all types of transfer (Sync, Isoc, Async/Packet, Control) over 64 logical channels
- 16KB buffering for synchronous /isochronous/control/packet data in the subsystem

NOTE

For more information, see the Media Local Bus (MLB) section of the Device TRM.

NOTE

MLB in 6-pin mode may require pull ups/ downs on SIG and DAT bus signals. For additional details, please consult the MLB bus interface specification.

Table 5-115 and Figure 5-77 present Timing Requirements for MLKCLK 3-Pin Option.

Table 5-115. Timing Requirements for MLBCLK 3-Pin Option (1)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
1	$t_{c(MLBCLK)}$	Cycle time, MLB_CLK	512FS	39		ns
			1024FS	19.5		ns
2	$t_{w(MLBCLK)}$	Pulse duration, MLB_CLK high	512FS	14		ns
			1024FS	9.3		ns
3	$t_{w(MLBCLK)}$	Pulse duration, MLB_CLK low	512FS	14		ns
			1024FS	6.1		ns

(1) The reference points for the rise and fall transitions are measured at V_{OL} MAX and V_{OH} MIN.

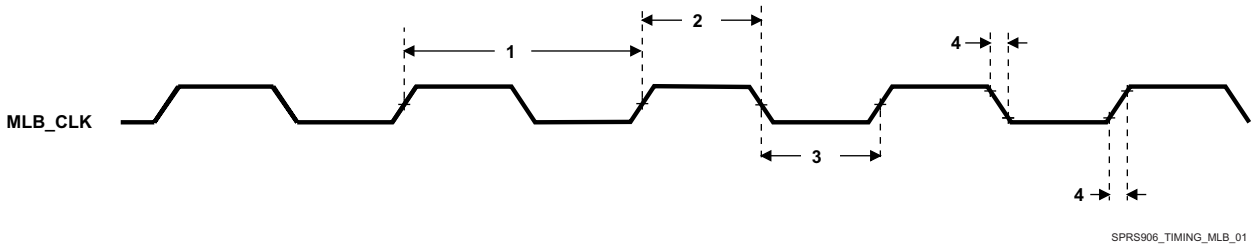


Figure 5-77. MLB_CLK Timing

Table 5-116 and Table 5-117 present Timing Requirements and Switching Characteristics for MLB 3-Pin Option.

Table 5-116. Timing Requirements for Receive Data for the MLB 3-Pin Option

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
5	$t_{su(MLBDAT-MLBCLKL)}$	Setup time, MLB_DAT/MLB_SIG input valid before MLB_CLK low	512FS	1		ns
			1024FS	1		ns
6	$t_{h(MLBCLKL-MLBDAT)}$	Hold time, MLB_DAT/MLB_SIG input valid after MLB_CLK low	512FS	4		ns
			1024FS	2		ns

Table 5-117. Switching Characteristics Over Recommended Operating Conditions for MLB 3-Pin Option

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
7	$t_{d(MLBCLKH-MLBDATV)}$	Delay time, MLBCLKH rising to MLB_DAT/MLB_SIG valid	512FS	0	10	ns
			1024FS	0	7	ns
8	$t_{dis(MLBCLKL-MLBDATZ)}$	Disable time, MLBCLKH falling to MLB_DAT/MLB_SIG Hi-Z	512FS	0	14	ns
			1024FS	0	6.1	ns

Table 5-118 and Figure 5-77 present Timing Requirements for MLKCLK 6-Pin Option.

Table 5-118. Timing Requirements for MLBCLK 6-Pin Option (1)

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
1	$t_{c(MLBCLKx)}$	Cycle time, MLB_CLKP/N	2048FS, 4096FS	10		ns
2	$t_{w(MLBCLKx)}$	Pulse duration, MLB_CLKP/N high	2048FS, 4096FS	4.5		ns
3	$t_{w(MLBCLKx)}$	Pulse duration, MLB_CLKP/N low	2048FS, 4096FS	4.5		ns

(1) The reference points for the rise and fall transitions are measured at 20%/80% of Vin+/-.

Table 5-119 and Table 5-120 present Timing Requirements and Switching Characteristics for MLB 6-Pin Option.

Table 5-119. Timing Requirements for Receive Data for the MLB 6-Pin Option

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
5	$t_{su(DATx-CLKxH)}$	Setup time, MLBP_DATx/MLBP_SIGx input valid before MLBP_CLKx rising	2048FS	1		ns
			4096FS	0.5 - n × P/2 (1)(2)		ns
6	$t_{h(CLKxH-DATx)}$	Hold time, MLBP_DATx/MLBP_SIGx input valid after MLBP_CLKx rising	2048FS	0.5		ns
			4096FS	0.6 + n × P/2 (1)(2)		ns

(1) P= tc(MLBCLKx) period.

(2) n=0 or 1, corresponding to two captures per clock cycle.

Table 5-120. Switching Characteristics Over Recommended Operating Conditions for MLB 6-Pin Option

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
7	$t_{d(CLKxH-DATxV)}$	Delay time, MLBPCLKxH rising to MLB_DATx/MLB_SIGx valid	2048FS	0.5	7	ns
			4096FS	0.6 + n × P/2 (1)(2)	2.5 + n × P/2	ns
8	$t_{dis(CLKPH-DATPZ)}$	Disable time, MLBPCLKxH rising to MLBP_DATx/MLBP_SIGx Hi-Z	2048FS	0.5	7	ns
			4096FS	0.6 + n × P/2 (1)(2)	3.5 + n × P/2	ns

(1) P= tc(MLBCLKx) period.

(2) n=0 or 1, corresponding to two captures per clock cycle.

In are presented the specific groupings of signals (IOSET) for use with MLB signals.

5.10.6.21 eMMC/SD/SDIO

The Device includes the following external memory interfaces 4 MultiMedia Card/Secure Digital/Secure Digital Input Output Interface (MMC/SD/SDIO)

NOTE

The eMMC/SD/SDIOi (i = 1 to 4) controller is also referred to as MMCi.

5.10.6.21.1 MMC1—SD Card Interface

MMC1 interface is compliant with the SD Standard v3.01 and it supports the following SD Card applications:

- Default speed, 4-bit data, SDR, half-cycle
- High speed, 4-bit data, SDR, half-cycle
- SDR12, 4-bit data, half-cycle
- SDR25, 4-bit data, half-cycle
- UHS-I SDR50, 4-bit data, half-cycle
- UHS-I SDR104, 4-bit data, half-cycle
- UHS-I DDR50, 4-bit data

NOTE

For more information, see the eMMC/SD/SDIO chapter of the Device TRM.

5.10.6.21.1.1 Default speed, 4-bit data, SDR, half-cycle

Table 5-121 and Table 5-122 present Timing requirements and Switching characteristics for MMC1 - Default Speed in receiver and transmitter mode (see Figure 5-78 and Figure 5-79).

Table 5-121. Timing Requirements for MMC1 - SD Card Default Speed Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DSSD5	$t_{su(cmdV-clkH)}$	Setup time, mmc1_cmd valid before mmc1_clk rising clock edge	5.11		ns
DSSD6	$t_{h(clkH-cmdV)}$	Hold time, mmc1_cmd valid after mmc1_clk rising clock edge	20.46		ns
DSSD7	$t_{su(dV-clkH)}$	Setup time, mmc1_dat[3:0] valid before mmc1_clk rising clock edge	5.11		ns
DSSD8	$t_{h(clkH-dV)}$	Hold time, mmc1_dat[3:0] valid after mmc1_clk rising clock edge	20.46		ns

Table 5-122. Switching Characteristics for MMC1 - SD Card Default Speed Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DSSD0	fop(clk)	Operating frequency, mmc1_clk		24	MHz
DSSD1	$t_{w(clkH)}$	Pulse duration, mmc1_clk high	0.5 × P-0.185 (1)		ns
DSSD2	$t_{w(clkL)}$	Pulse duration, mmc1_clk low	0.5 × P-0.185 (1)		ns
DSSD3	$t_{d(clkL-cmdV)}$	Delay time, mmc1_clk falling clock edge to mmc1_cmd transition	-14.93	14.93	ns
DSSD4	$t_{d(clkL-dV)}$	Delay time, mmc1_clk falling clock edge to mmc1_dat[3:0] transition	-14.93	14.93	ns

(1) P = output mmc1_clk period in ns

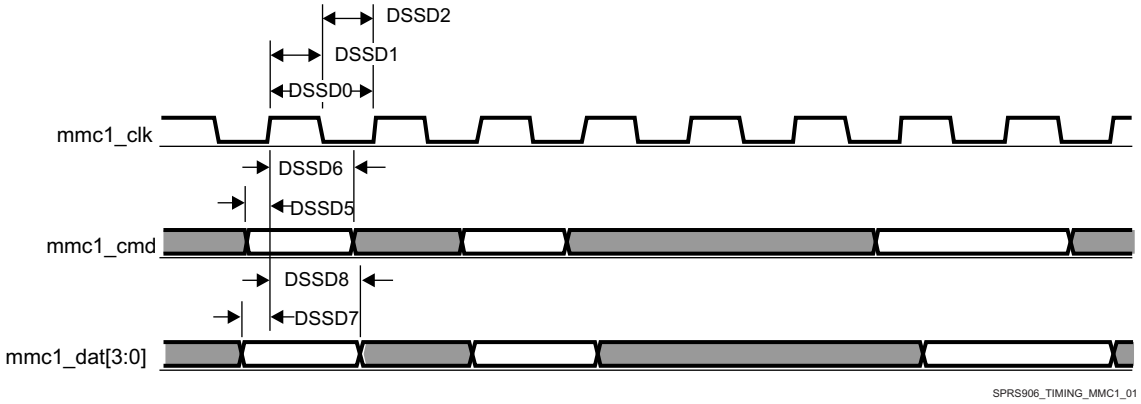


Figure 5-78. MMC/SD/SDIO in - Default Speed - Receiver Mode

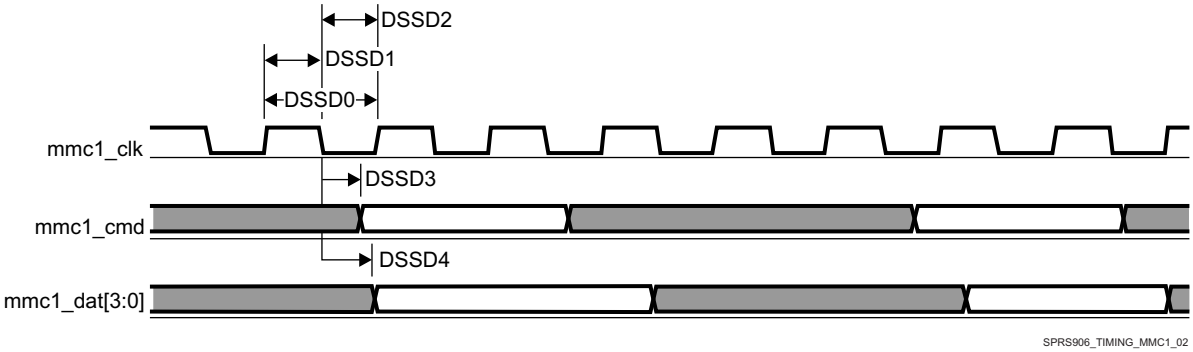


Figure 5-79. MMC/SD/SDIO in - Default Speed - Transmitter Mode

5.10.6.21.1.2 High speed, 4-bit data, SDR, half-cycle

Table 5-123 and Table 5-124 present Timing requirements and Switching characteristics for MMC1 - High Speed in receiver and transmitter mode (see Figure 5-80 and Figure 5-81).

Table 5-123. Timing Requirements for MMC1 - SD Card High Speed

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
HSSD3	$t_{su(cmdV-clkH)}$	Setup time, mmc1_cmd valid before mmc1_clk rising clock edge	5.3		ns
HSSD4	$t_{h(clkH-cmdV)}$	Hold time, mmc1_cmd valid after mmc1_clk rising clock edge	2.6		ns
HSSD7	$t_{su(dV-clkH)}$	Setup time, mmc1_dat[3:0] valid before mmc1_clk rising clock edge	5.3		ns
HSSD8	$t_{h(clkH-dV)}$	Hold time, mmc1_dat[3:0] valid after mmc1_clk rising clock edge	2.6		ns

Table 5-124. Switching Characteristics for MMC1 - SD Card High Speed

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
HSSD1	fop(clk)	Operating frequency, mmc1_clk		48	MHz
HSSD2H	$t_{w(clkH)}$	Pulse duration, mmc1_clk high	0.5 × P-0.185 (1)		ns
HSSD2L	$t_{w(clkL)}$	Pulse duration, mmc1_clk low	0.5 × P-0.185 (1)		ns
HSSD5	$t_{d(clkL-cmdV)}$	Delay time, mmc1_clk falling clock edge to mmc1_cmd transition	-7.6	3.6	ns
HSSD6	$t_{d(clkL-dV)}$	Delay time, mmc1_clk falling clock edge to mmc1_dat[3:0] transition	-7.6	3.6	ns

(1) P = output mmc1_clk period in ns

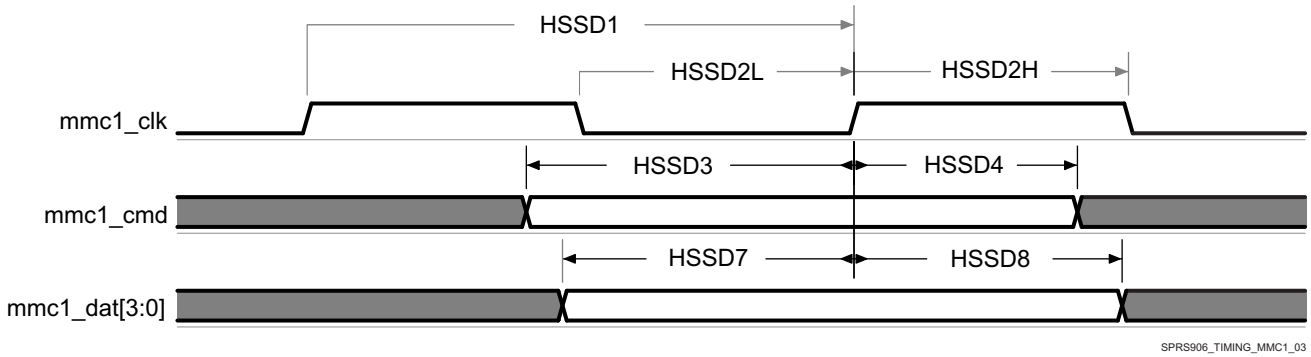


Figure 5-80. MMC/SD/SDIO in - High Speed - Receiver Mode

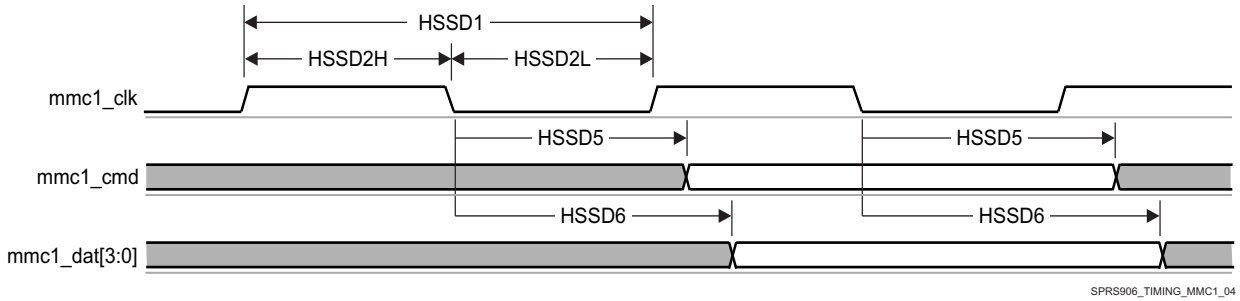


Figure 5-81. MMC/SD/SDIO in - High Speed - Transmitter Mode

5.10.6.21.1.3 SDR12, 4-bit data, half-cycle

Table 5-125 and Table 5-126 present Timing requirements and Switching characteristics for MMC1 - SDR12 in receiver and transmitter mode (see Figure 5-82 and Figure 5-83).

Table 5-125. Timing Requirements for MMC1 - SD Card SDR12 Mode

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
SDR125	$t_{su(cmdV-clkH)}$	Setup time, mmc1_cmd valid before mmc1_clk rising clock edge		25.99		ns
SDR126	$t_{h(clkH-cmdV)}$	Hold time, mmc1_cmd valid after mmc1_clk rising clock edge	Pad Loopback Clock	1.6		ns
			Internal Loopback Clock	1.6		ns
SDR127	$t_{su(dV-clkH)}$	Setup time, mmc1_dat[3:0] valid before mmc1_clk rising clock edge		25.99		ns
SDR128	$t_{h(clkH-dV)}$	Hold time, mmc1_dat[3:0] valid after mmc1_clk rising clock edge	Pad Loopback Clock	1.6		ns
			Internal Loopback Clock	1.6		ns

Table 5-126. Switching Characteristics for MMC1 - SD Card SDR12 Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR120	fop(clk)	Operating frequency, mmc1_clk		24	MHz
SDR121	$t_{w(clkH)}$	Pulse duration, mmc1_clk high	0.5 × P-0.185 (1)		ns
SDR122	$t_{w(clkL)}$	Pulse duration, mmc1_clk low	0.5 × P-0.185(1)		ns
SDR123	$t_{d(clkL-cmdV)}$	Delay time, mmc1_clk falling clock edge to mmc1_cmd transition	-19.13	16.93	ns
SDR124	$t_{d(clkL-dV)}$	Delay time, mmc1_clk falling clock edge to mmc1_dat[3:0] transition	-19.13	16.93	ns

(1) P = output mmc1_clk period in ns

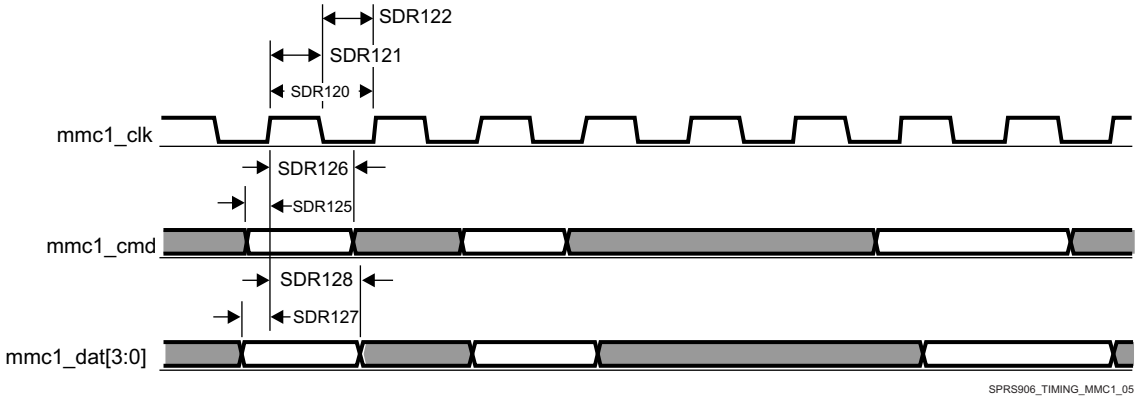


Figure 5-82. MMC/SD/SDIO in - High Speed SDR12 - Receiver Mode

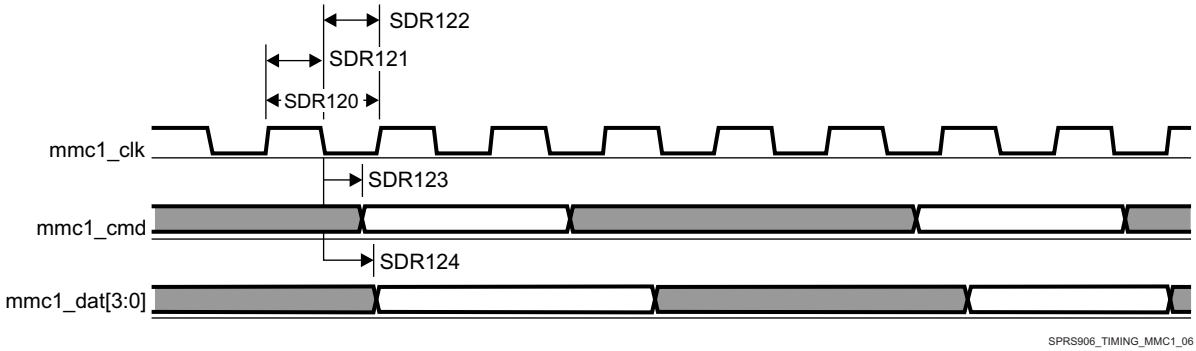


Figure 5-83. MMC/SD/SDIO in - High Speed SDR12 - Transmitter Mode

5.10.6.21.1.4 SDR25, 4-bit data, half-cycle

Table 5-127 and Table 5-128 present Timing requirements and Switching characteristics for MMC1 - SDR25 in receiver and transmitter mode (see Figure 5-84 and Figure 5-85).

Table 5-127. Timing Requirements for MMC1 - SD Card SDR25 Mode

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
SDR253	$t_{su(cmdV-clkH)}$	Setup time, mmc1_cmd valid before mmc1_clk rising clock edge		5.3		ns
SDR254	$t_{h(clkH-cmdV)}$	Hold time, mmc1_cmd valid after mmc1_clk rising clock edge		1.6		ns
SDR257	$t_{su(dV-clkH)}$	Setup time, mmc1_dat[3:0] valid before mmc1_clk rising clock edge		5.3		ns
SDR258	$t_{h(clkH-dV)}$	Hold time, mmc1_dat[3:0] valid after mmc1_clk rising clock edge	Pad Loopback Clock	1.6		ns
			Internal Loopback Clock	1.6		ns

Table 5-128. Switching Characteristics for MMC1 - SD Card SDR25 Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR251	fop(clk)	Operating frequency, mmc1_clk		48	MHz
SDR252H	$t_{w(clkH)}$	Pulse duration, mmc1_clk high	0.5 × P-0.185 (1)		ns
SDR252L	$t_{w(clkL)}$	Pulse duration, mmc1_clk low	0.5 × P-0.185 (1)		ns
SDR255	$t_{d(clkL-cmdV)}$	Delay time, mmc1_clk falling clock edge to mmc1_cmd transition	-8.8	6.6	ns

Table 5-128. Switching Characteristics for MMC1 - SD Card SDR25 Mode (continued)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR256	$t_{d(clkL-dV)}$	Delay time, mmc1_clk falling clock edge to mmc1_dat[3:0] transition	-8.8	6.6	ns

(1) P = output mmc1_clk period in ns

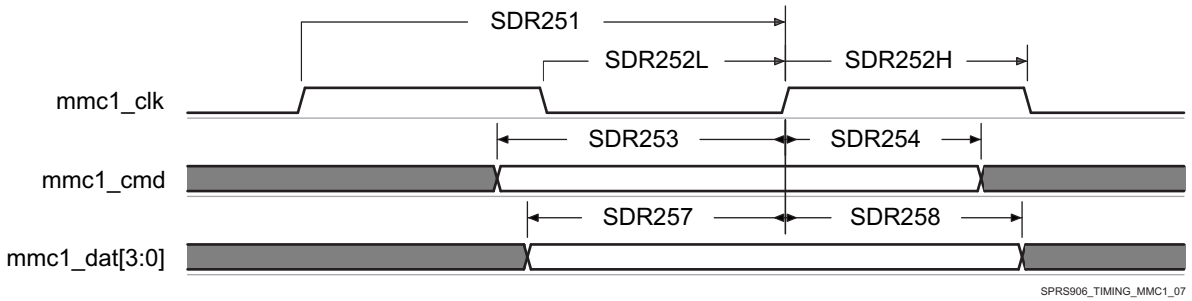


Figure 5-84. MMC/SD/SDIO in - High Speed SDR25 - Receiver Mode

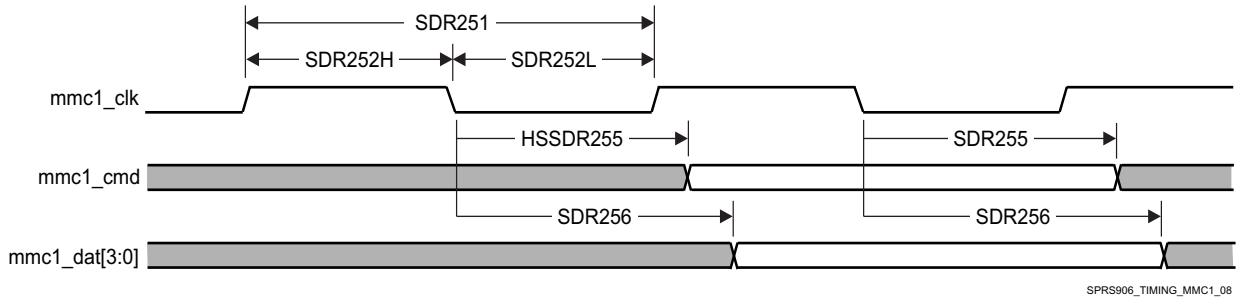


Figure 5-85. MMC/SD/SDIO in - High Speed SDR25 - Transmitter Mode

5.10.6.21.1.5 UHS-I SDR50, 4-bit data, half-cycle

Table 5-129 and Table 5-130 present Timing requirements and Switching characteristics for MMC1 - SDR50 in receiver and transmitter mode (see Figure 5-86 and Figure 5-87).

Table 5-129. Timing Requirements for MMC1 - SD Card SDR50 Mode

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
SDR503	$t_{su(cmdV-clkH)}$	Setup time, mmc1_cmd valid before mmc1_clk rising clock edge		1.48		ns
SDR504	$t_{h(clkH-cmdV)}$	Hold time, mmc1_cmd valid after mmc1_clk rising clock edge		1.7		ns
SDR507	$t_{su(dV-clkH)}$	Setup time, mmc1_dat[3:0] valid before mmc1_clk rising clock edge		1.48		ns
SDR508	$t_{h(clkH-dV)}$	Hold time, mmc1_dat[3:0] valid after mmc1_clk rising clock edge	Pad Loopback Clock	1.7		ns
			Internal Loopback Clock	1.6		ns

Table 5-130. Switching Characteristics for MMC1 - SD Card SDR50 Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR501	fop(clk)	Operating frequency, mmc1_clk		96	MHz
SDR502H	$t_{w(clkH)}$	Pulse duration, mmc1_clk high	0.5 × P-0.185 (1)		ns
SDR502L	$t_{w(clkL)}$	Pulse duration, mmc1_clk low	0.5 × P-0.185 (1)		ns
SDR505	$t_{d(clkL-cmdV)}$	Delay time, mmc1_clk falling clock edge to mmc1_cmd transition	-8.8	6.6	ns
SDR506	$t_{d(clkL-dV)}$	Delay time, mmc1_clk falling clock edge to mmc1_dat[3:0] transition	-3.66	1.46	ns

(1) P = output mmc1_clk period in ns

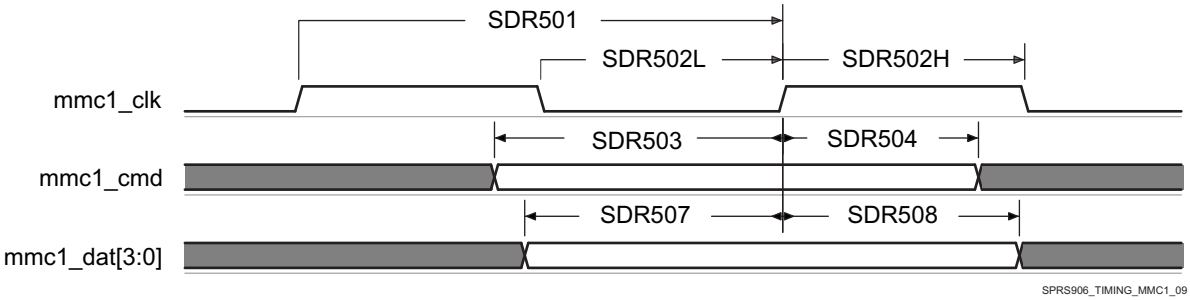


Figure 5-86. MMC/SD/SDIO in - High Speed SDR50 - Receiver Mode

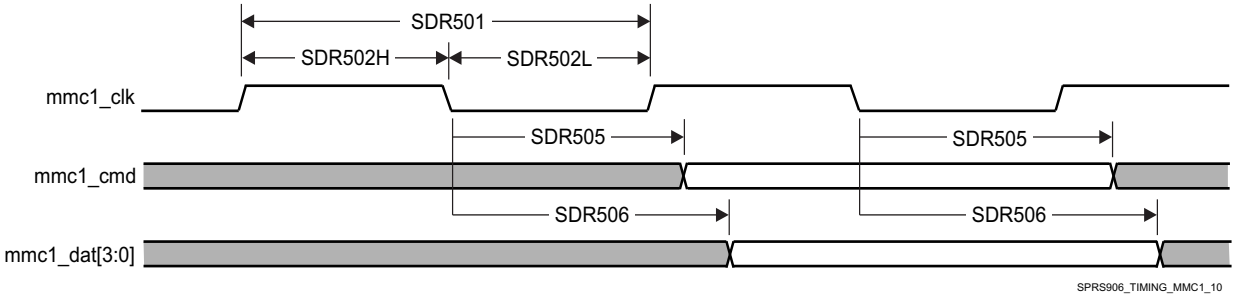


Figure 5-87. MMC/SD/SDIO in - High Speed SDR50 - Transmitter Mode

5.10.6.21.1.6 UHS-I SDR104, 4-bit data, half-cycle

Table 5-131 presents Timing requirements and Switching characteristics for MMC1 - SDR104 in receiver and transmitter mode (see Figure 5-88 and Figure 5-89).

Table 5-131. Switching Characteristics for MMC1 - SD Card SDR104 Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR1041	fop(clk)	Operating frequency, mmc1_clk		192	MHz
SDR1042H	$t_{w(clkH)}$	Pulse duration, mmc1_clk high	0.5 × P-0.185 (1)		ns
SDR1042L	$t_{w(clkL)}$	Pulse duration, mmc1_clk low	0.5 × P-0.185 (1)		ns
SDR1045	$t_{d(clkL-cmdV)}$	Delay time, mmc1_clk falling clock edge to mmc1_cmd transition	-1.09	0.49	ns
SDR1046	$t_{d(clkL-dV)}$	Delay time, mmc1_clk falling clock edge to mmc1_dat[3:0] transition	-1.09	0.49	ns

(1) P = output mmc1_clk period in ns

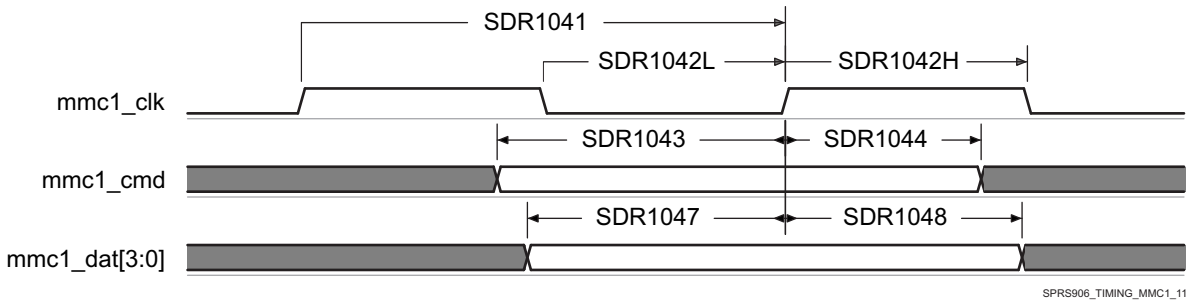


Figure 5-88. MMC/SD/SDIO in - High Speed SDR104 - Receiver Mode

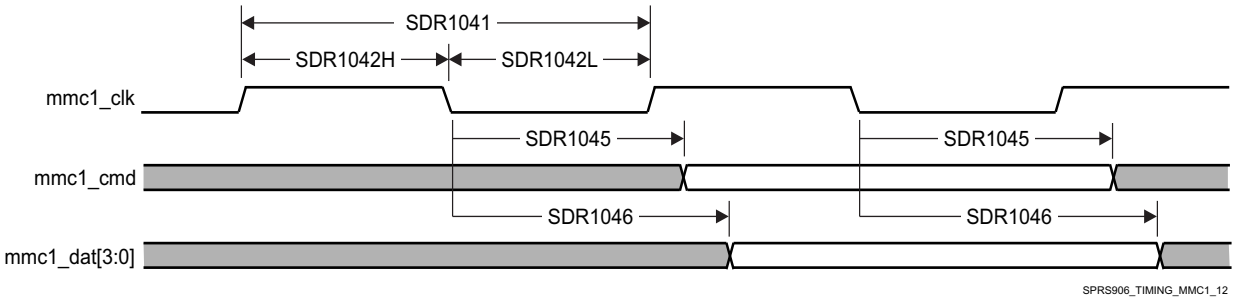


Figure 5-89. MMC/SD/SDIO in - High Speed SDR104 - Transmitter Mode

5.10.6.21.1.7 UHS-I DDR50, 4-bit data

Table 5-132 and Table 5-133 present Timing requirements and Switching characteristics for MMC1 - DDR50 in receiver and transmitter mode (see Figure 5-90 and Figure 5-91).

Table 5-132. Timing Requirements for MMC1 - SD Card DDR50 Mode

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
DDR505	$t_{su(cmdV-clk)}$	Setup time, mmc1_cmd valid before mmc1_clk transition		1.79		ns
DDR506	$t_{h(clk-cmdV)}$	Hold time, mmc1_cmd valid after mmc1_clk transition		2		ns
DDR507	$t_{su(dV-clk)}$	Setup time, mmc1_dat[3:0] valid before mmc1_clk transition	Pad Loopback	1.79		ns
			Internal Loopback	1.79		ns
DDR508	$t_{h(clk-dV)}$	Hold time, mmc1_dat[3:0] valid after mmc1_clk transition	Pad Loopback	2		ns
			Internal Loopback	1.6		ns

Table 5-133. Switching Characteristics for MMC1 - SD Card DDR50 Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DDR500	fop(clk)	Operating frequency, mmc1_clk		48	MHz
DDR501	$t_{w(clkH)}$	Pulse duration, mmc1_clk high	0.5 × P-0.185 (1)		ns
DDR502	$t_{w(clkL)}$	Pulse duration, mmc1_clk low	0.5 × P-0.185 (1)		ns
DDR503	$t_{d(clk-cmdV)}$	Delay time, mmc1_clk transition to mmc1_cmd transition	1.225	6.6	ns
DDR504	$t_{d(clk-dV)}$	Delay time, mmc1_clk transition to mmc1_dat[3:0] transition	1.225	6.6	ns

(1) P = output mmc1_clk period in ns

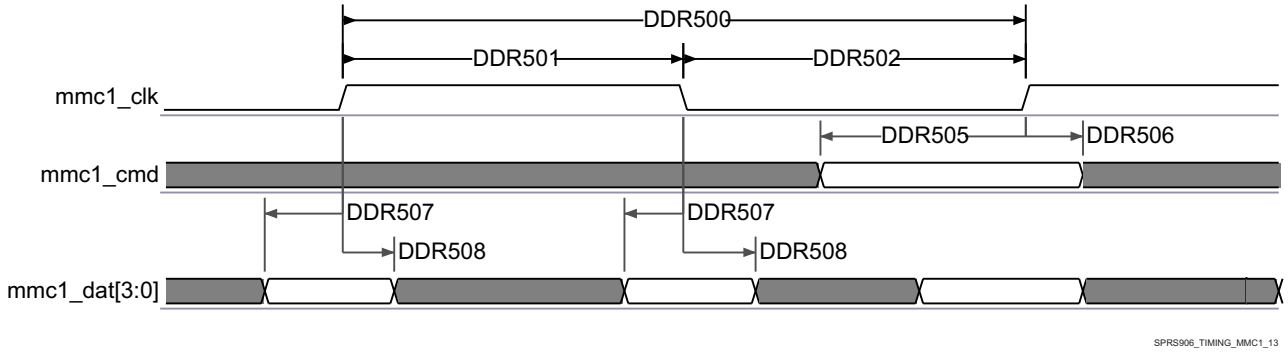


Figure 5-90. SDMMC - High Speed SD - DDR - Data/Command Receive

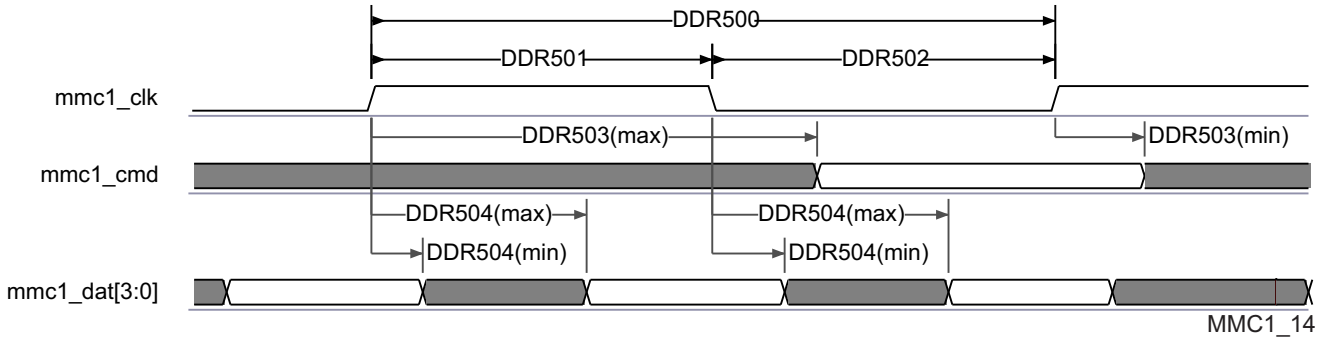


Figure 5-91. SDMMC - High Speed SD - DDR - Data/Command Transmit

NOTE

To configure the desired virtual mode the user must set MODESELECT bit and DELAYMODE bitfield for each corresponding pad control register.

The pad control registers are presented in Table 4-32 and described in Device TRM, *Control Module Chapter*.

Virtual IO Timings Modes must be used to ensure some IO timings for MMC1. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Virtual IO Timings Modes. See Table 5-134 *Virtual Functions Mapping for MMC1* for a definition of the Virtual modes.

Table 5-134 presents the values for DELAYMODE bitfield.

Table 5-134. Virtual Functions Mapping for MMC1

BALL	BALL NAME	Delay Mode Value				MUXMODE
		MMC1_ VIRTUAL1	MMC1_ VIRTUAL4	MMC1_ VIRTUAL5	MMC1_ VIRTUAL6	0
U3	mmc1_clk	15	12	11	10	mmc1_clk
V4	mmc1_cmd	15	12	11	10	mmc1_cmd
V3	mmc1_dat0	15	12	11	10	mmc1_dat0
V2	mmc1_dat1	15	12	11	10	mmc1_dat1
W1	mmc1_dat2	15	12	11	10	mmc1_dat2
V1	mmc1_dat3	15	12	11	10	mmc1_dat3

NOTE

To configure the desired Manual IO Timing Mode the user must follow the steps described in section Manual IO Timing Modes of the Device TRM.

The associated registers to configure are listed in the **CFG REGISTER** column. For more information see the Control Module chapter in the Device TRM.

Manual IO Timings Modes must be used to ensure some IO timings for MMC1. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-135 *Manual Functions Mapping for MMC1* for a definition of the Manual modes.

Table 5-135 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-135. Manual Functions Mapping for MMC1

BALL	BALL NAME	MMC1_MANUAL1		MMC1_MANUAL2		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		0
U3	mmc1_clk	588	0	-	-	CFG_MMC1_CLK_IN	mmc1_clk
V4	mmc1_cmd	1000	0	-	-	CFG_MMC1_CMD_IN	mmc1_cmd
V3	mmc1_dat0	1375	0	-	-	CFG_MMC1_DAT0_IN	mmc1_dat0
V2	mmc1_dat1	1000	0	-	-	CFG_MMC1_DAT1_IN	mmc1_dat1
W1	mmc1_dat2	1000	0	-	-	CFG_MMC1_DAT2_IN	mmc1_dat2
V1	mmc1_dat3	1000	0	-	-	CFG_MMC1_DAT3_IN	mmc1_dat3
U3	mmc1_clk	1230	0	520	320	CFG_MMC1_CLK_OUT	mmc1_clk
V4	mmc1_cmd	0	0	0	0	CFG_MMC1_CMD_OUT	mmc1_cmd
V3	mmc1_dat0	56	0	40	0	CFG_MMC1_DAT0_OUT	mmc1_dat0
V2	mmc1_dat1	76	0	83	0	CFG_MMC1_DAT1_OUT	mmc1_dat1
W1	mmc1_dat2	91	0	98	0	CFG_MMC1_DAT2_OUT	mmc1_dat2
V1	mmc1_dat3	99	0	106	0	CFG_MMC1_DAT3_OUT	mmc1_dat3
V4	mmc1_cmd	0	0	51	0	CFG_MMC1_CMD_OEN	mmc1_cmd
V3	mmc1_dat0	0	0	0	0	CFG_MMC1_DAT0_OEN	mmc1_dat0
V2	mmc1_dat1	0	0	363	0	CFG_MMC1_DAT1_OEN	mmc1_dat1
W1	mmc1_dat2	0	0	199	0	CFG_MMC1_DAT2_OEN	mmc1_dat2
V1	mmc1_dat3	0	0	273	0	CFG_MMC1_DAT3_OEN	mmc1_dat3

5.10.6.21.2 MMC2 — eMMC

MMC2 interface is compliant with the JC64 eMMC Standard v4.5 and it supports the following eMMC applications:

- Standard JC64 SDR, 8-bit data, half cycle
- High-speed JC64 SDR, 8-bit data, half cycle
- High-speed HS200 JEDS84, 8-bit data, half cycle
- High-speed JC64 DDR, 8-bit data

NOTE

For more information, see the eMMC/SD/SDIO chapter of the Device TRM.

5.10.6.21.2.1 Standard JC64 SDR, 8-bit data, half cycle

Table 5-136 and Table 5-137 present Timing requirements and Switching characteristics for MMC2 - Standart SDR in receiver and transmitter mode (see Figure 5-92 and Figure 5-93).

Table 5-136. Timing Requirements for MMC2 - JC64 Standard SDR Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SSDR5	$t_{su(cmdV-clkH)}$	Setup time, mmc2_cmd valid before mmc2_clk rising clock edge	13.19		ns
SSDR6	$t_{h(clkH-cmdV)}$	Hold time, mmc2_cmd valid after mmc2_clk rising clock edge	8.4		ns
SSDR7	$t_{su(dV-clkH)}$	Setup time, mmc2_dat[7:0] valid before mmc2_clk rising clock edge	13.19		ns
SSDR8	$t_{h(clkH-dV)}$	Hold time, mmc2_dat[7:0] valid after mmc2_clk rising clock edge	8.4		ns

Table 5-137. Switching Characteristics for MMC2 - JC64 Standard SDR Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SSDR1	fop(clk)	Operating frequency, mmc2_clk		24	MHz
SSDR2H	$t_{w(clkH)}$	Pulse duration, mmc2_clk high	0.5 × P-0.172 (1)		ns
SSDR2L	$t_{w(clkL)}$	Pulse duration, mmc2_clk low	0.5 × P-0.172 (1)		ns
SSDR3	$t_{d(clkL-cmdV)}$	Delay time, mmc2_clk falling clock edge to mmc2_cmd transition	-16.96	16.96	ns
SSDR4	$t_{d(clkL-dV)}$	Delay time, mmc2_clk falling clock edge to mmc2_dat[7:0] transition	-16.96	16.96	ns

(1) P = output mmc2_clk period in ns

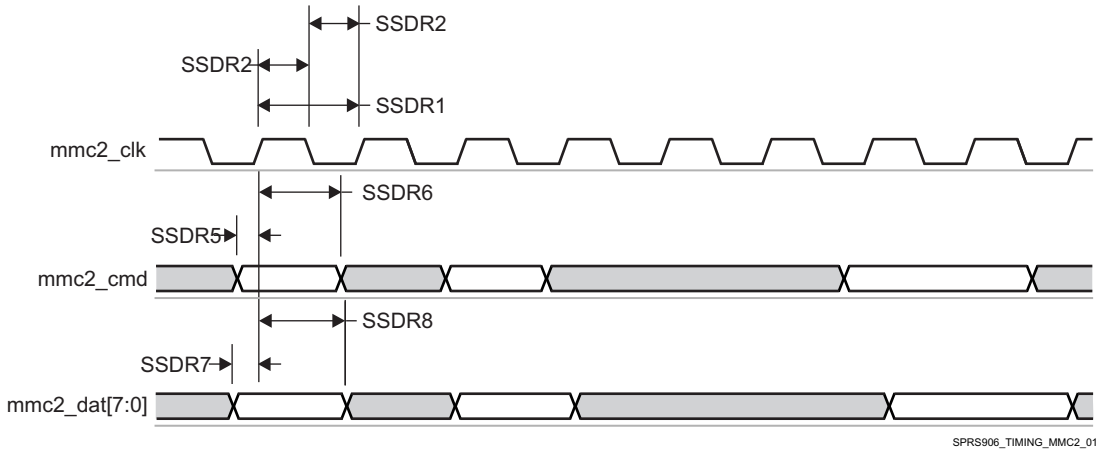


Figure 5-92. MMC/SD/SDIO in - Standard JC64 - Receiver Mode

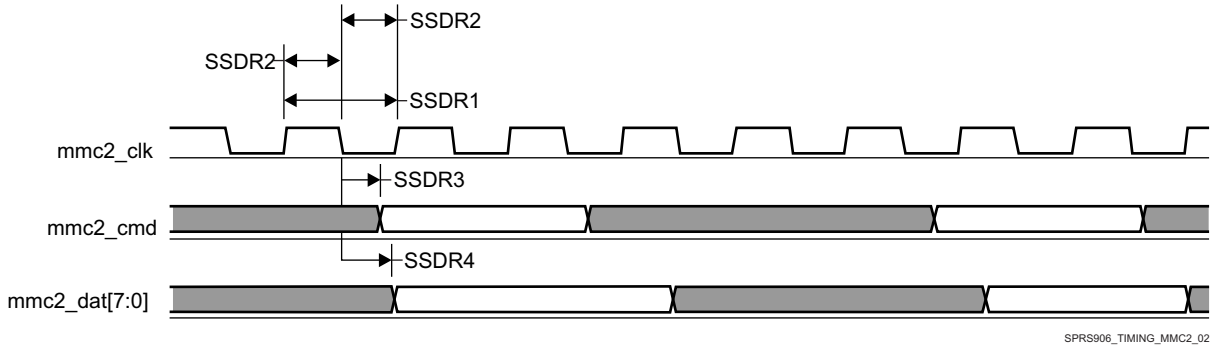


Figure 5-93. MMC/SD/SDIO in - Standard JC64 - Transmitter Mode

5.10.6.21.2.2 High-speed JC64 SDR, 8-bit data, half cycle

Table 5-138 and Table 5-139 present Timing requirements and Switching characteristics for MMC2 - High speed SDR in receiver and transmitter mode (see Figure 5-94 and Figure 5-95).

Table 5-138. Timing Requirements for MMC2 - JC64 High Speed SDR Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
JC643	$t_{su(cmdV-clkH)}$	Setup time, mmc2_cmd valid before mmc2_clk rising clock edge	5.6		ns
JC644	$t_{h(clkH-cmdV)}$	Hold time, mmc2_cmd valid after mmc2_clk rising clock edge	2.6		ns
JC647	$t_{su(dV-clkH)}$	Setup time, mmc2_dat[7:0] valid before mmc2_clk rising clock edge	5.6		ns
JC648	$t_{h(clkH-dV)}$	Hold time, mmc2_dat[7:0] valid after mmc2_clk rising clock edge	2.6		ns

Table 5-139. Switching Characteristics for MMC2 - JC64 High Speed SDR Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
JC641	fop(clk)	Operating frequency, mmc2_clk		48	MHz
JC642H	$t_{w(clkH)}$	Pulse duration, mmc2_clk high	0.5 × P-0.172 (1)		ns
JC642L	$t_{w(clkL)}$	Pulse duration, mmc2_clk low	0.5 × P-0.172 (1)		ns
JC645	$t_{d(clkL-cmdV)}$	Delay time, mmc2_clk falling clock edge to mmc2_cmd transition	-6.64	6.64	ns
JC646	$t_{d(clkL-dV)}$	Delay time, mmc2_clk falling clock edge to mmc2_dat[7:0] transition	-6.64	6.64	ns

(1) P = output mmc2_clk period in ns

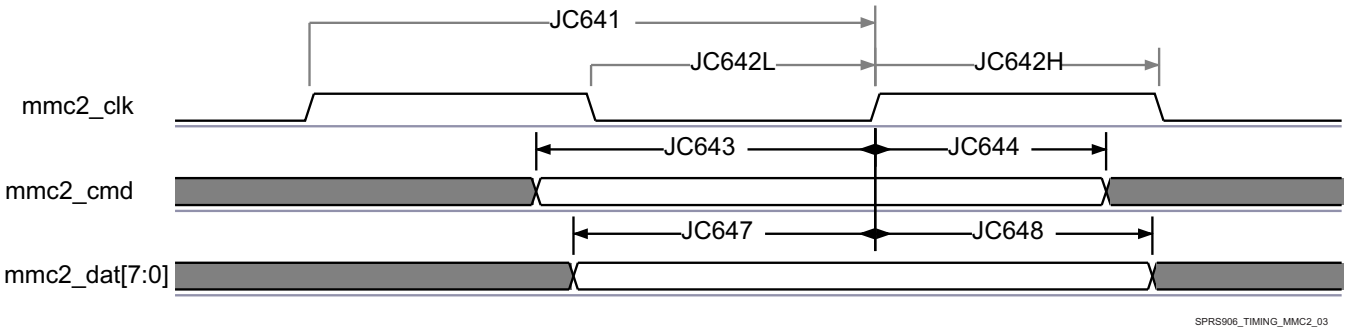


Figure 5-94. MMC/SD/SDIO in - High Speed JC64 - Receiver Mode

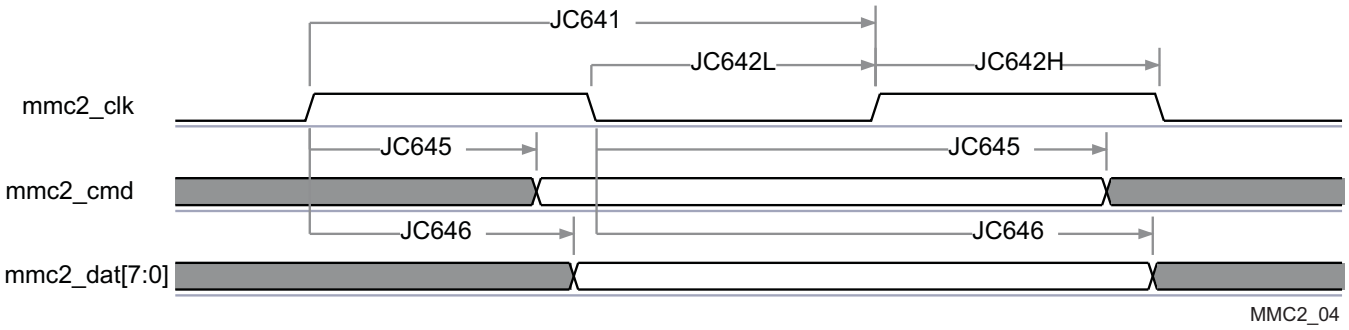


Figure 5-95. MMC/SD/SDIO in - High Speed JC64 - transmitter Mode

5.10.6.21.2.3 High-speed HS200 JEDS84 SDR, 8-bit data, half cycle

Table 5-140 presents Switching characteristics for MMC2 - HS200 in transmitter mode (see Figure 5-96).

Table 5-140. Switching Characteristics for MMC2 - JEDS84 HS200 Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
HS2001	fop(clk)	Operating frequency, mmc2_clk		192	MHz
HS2002H	$t_{w(clkH)}$	Pulse duration, mmc2_clk high	0.5 × P-0.172 (1)		ns
HS2002L	$t_{w(clkL)}$	Pulse duration, mmc2_clk low	0.5 × P-0.172 (1)		ns
HS2005	$t_{d(clkL-cmdV)}$	Delay time, mmc2_clk falling clock edge to mmc2_cmd transition	-1.136	0.536	ns
HS2006	$t_{d(clkL-dV)}$	Delay time, mmc2_clk falling clock edge to mmc2_dat[7:0] transition	-1.136	0.536	ns

(1) P = output mmc2_clk period in ns

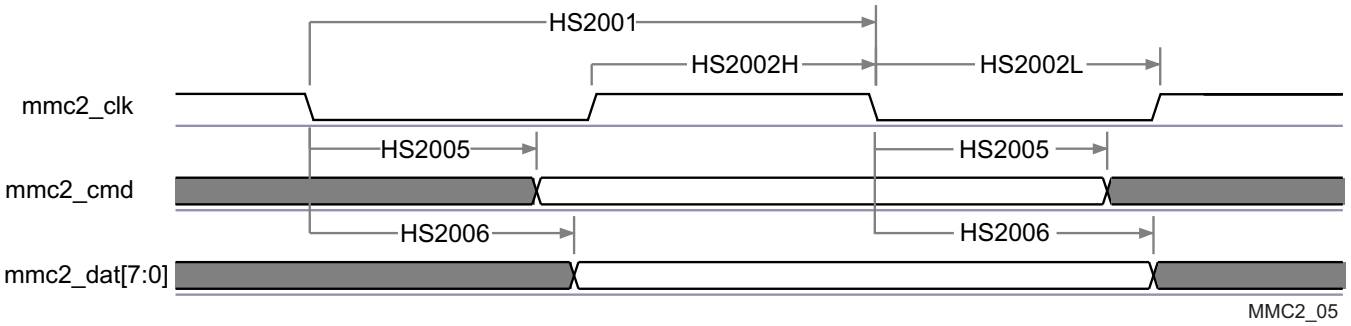


Figure 5-96. eMMC in - HS200 SDR - Transmitter Mode

5.10.6.21.2.4 High-speed JC64 DDR, 8-bit data

Table 5-141 and Table 5-142 present Timing requirements and Switching characteristics for MMC2 - High speed DDR in receiver and transmitter mode (see Figure 5-97 and Figure 5-98).

Table 5-141. Timing Requirements for MMC2 - JC64 High Speed DDR Mode

NO.	PARAMETER	DESCRIPTION	MODE	MIN	MAX	UNIT
DDR3	$t_{su(cmdV-clk)}$	Setup time, mmc2_cmd valid before mmc2_clk transition		1.8		ns
DDR4	$t_{h(clk-cmdV)}$	Hold time, mmc2_cmd valid after mmc2_clk transition		1.6		ns
DDR7	$t_{su(dV-clk)}$	Setup time, mmc2_dat[7:0] valid before mmc2_clk transition		1.8		ns
DDR8	$t_{h(clk-dV)}$	Hold time, mmc2_dat[7:0] valid after mmc2_clk transition	Pad Loopback (1.8V and 3.3V), Boot	1.6		ns
			Internal Loopback (1.8V with MMC2_VIRTUAL2)	1.86		ns
			Internal Loopback (3.3V with MMC2_VIRTUAL2)	1.95		ns
			Internal Loopback (1.8V with MMC2_MANUAL2)			ns
			Internal Loopback (3.3V with MMC2_MANUAL2)	1.6		ns

Table 5-142. Switching Characteristics for MMC2 - JC64 High Speed DDR Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DDR1	fop(clk)	Operating frequency, mmc2_clk		48	MHz
DDR2H	$t_{w(clkH)}$	Pulse duration, mmc2_clk high	0.5 × P-0.172 (1)		ns
DDR2L	$t_{w(clkL)}$	Pulse duration, mmc2_clk low	0.5 × P-0.172 (1)		ns

Table 5-142. Switching Characteristics for MMC2 - JC64 High Speed DDR Mode (continued)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DDR5	$t_{d(clk-cmdV)}$	Delay time, mmc2_clk transition to mmc2_cmd transition	2.9	7.14	ns
DDR6	$t_{d(clk-dV)}$	Delay time, mmc2_clk transition to mmc2_dat[7:0] transition	2.9	7.14	ns

(1) P = output mmc2_clk period in ns

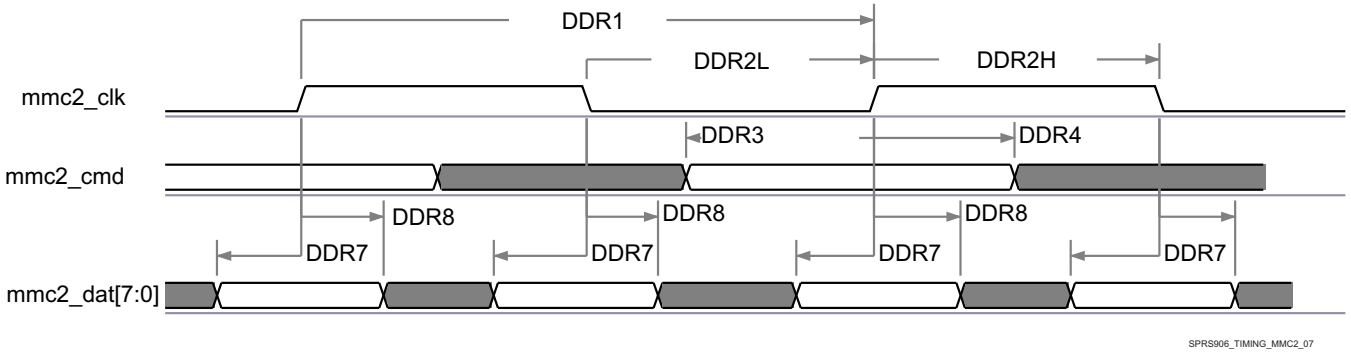


Figure 5-97. MMC/SD/SDIO in - High Speed DDR JC64 - Receiver Mode

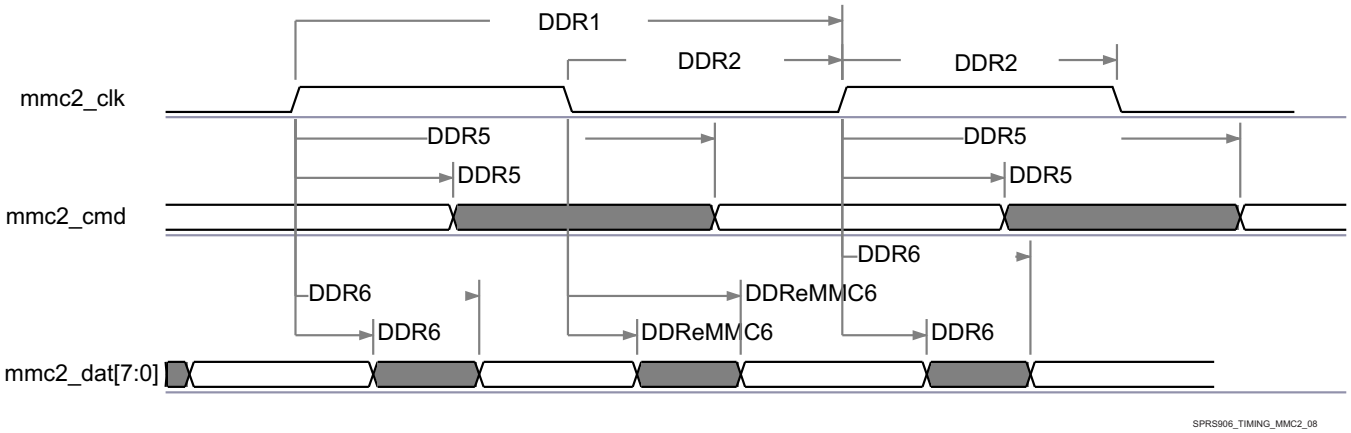


Figure 5-98. MMC/SD/SDIO in - High Speed DDR JC64 - Transmitter Mode

NOTE

To configure the desired virtual mode the user must set MODESELECT bit and DELAYMODE bitfield for each corresponding pad control register.

The pad control registers are presented in Table 4-32 and described in Device TRM, *Control Module Chapter*.

Virtual IO Timings Modes must be used to ensure some IO timings for MMC2. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Virtual IO Timings Modes. See Table 5-143 *Virtual Functions Mapping for MMC2* for a definition of the Virtual modes.

Table 5-143 presents the values for DELAYMODE bitfield.

Table 5-143. Virtual Functions Mapping for MMC2

BALL	BALL NAME	Delay Mode Value	MUXMODE
		MMC2_VIRTUAL2	1
A6	gpmc_cs1	13	mmc2_cmd
A4	gpmc_a19	13	mmc2_dat4

Table 5-143. Virtual Functions Mapping for MMC2 (continued)

BALL	BALL NAME	Delay Mode Value	MUXMODE
		MMC2_VIRTUAL2	1
E7	gpmc_a20	13	mmc2_dat5
D6	gpmc_a21	13	mmc2_dat6
C5	gpmc_a22	13	mmc2_dat7
B5	gpmc_a23	13	mmc2_clk
D7	gpmc_a24	13	mmc2_dat0
C6	gpmc_a25	13	mmc2_dat1
A5	gpmc_a26	13	mmc2_dat2
B6	gpmc_a27	13	mmc2_dat3

NOTE

To configure the desired Manual IO Timing Mode the user must follow the steps described in section Manual IO Timing Modes of the Device TRM.

The associated registers to configure are listed in the **CFG REGISTER** column. For more information see the Control Module chapter in the Device TRM.

Manual IO Timings Modes must be used to ensure some IO timings for MMC2. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-144 *Manual Functions Mapping for MMC2* for a definition of the Manual modes.

Table 5-144 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-144. Manual Functions Mapping for MMC2

BALL	BALL NAME	MMC2_MANUAL1		MMC2_MANUAL2		MMC2_MANUAL3		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		1
A4	gpmc_a19	0	0	0	14	-	-	CFG_GPMC_A19_IN	mmc2_dat4
E7	gpmc_a20	119	0	127	0	-	-	CFG_GPMC_A20_IN	mmc2_dat5
D6	gpmc_a21	0	0	22	0	-	-	CFG_GPMC_A21_IN	mmc2_dat6
C5	gpmc_a22	18	0	72	0	-	-	CFG_GPMC_A22_IN	mmc2_dat7
B5	gpmc_a23	894	0	410	4000	-	-	CFG_GPMC_A23_IN	mmc2_clk
D7	gpmc_a24	30	0	82	0	-	-	CFG_GPMC_A24_IN	mmc2_dat0
C6	gpmc_a25	0	0	0	0	-	-	CFG_GPMC_A25_IN	mmc2_dat1
A5	gpmc_a26	23	0	77	0	-	-	CFG_GPMC_A26_IN	mmc2_dat2
B6	gpmc_a27	0	0	0	0	-	-	CFG_GPMC_A27_IN	mmc2_dat3
A6	gpmc_cs1	0	0	0	0	-	-	CFG_GPMC_CS1_IN	mmc2_cmd
A4	gpmc_a19	152	0	152	0	285	0	CFG_GPMC_A19_OUT	mmc2_dat4
E7	gpmc_a20	206	0	206	0	189	0	CFG_GPMC_A20_OUT	mmc2_dat5
D6	gpmc_a21	78	0	78	0	0	120	CFG_GPMC_A21_OUT	mmc2_dat6
C5	gpmc_a22	2	0	2	0	0	70	CFG_GPMC_A22_OUT	mmc2_dat7
B5	gpmc_a23	266	0	266	0	730	360	CFG_GPMC_A23_OUT	mmc2_clk
D7	gpmc_a24	0	0	0	0	0	0	CFG_GPMC_A24_OUT	mmc2_dat0
C6	gpmc_a25	0	0	0	0	0	0	CFG_GPMC_A25_OUT	mmc2_dat1
A5	gpmc_a26	43	0	43	0	70	0	CFG_GPMC_A26_OUT	mmc2_dat2
B6	gpmc_a27	0	0	0	0	0	0	CFG_GPMC_A27_OUT	mmc2_dat3
A6	gpmc_cs1	0	0	0	0	0	120	CFG_GPMC_CS1_OUT	mmc2_cmd
A4	gpmc_a19	0	0	0	0	0	0	CFG_GPMC_A19_OEN	mmc2_dat4

Table 5-144. Manual Functions Mapping for MMC2 (continued)

BALL	BALL NAME	MMC2_MANUAL1		MMC2_MANUAL2		MMC2_MANUAL3		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)	A_DELAY (ps)	G_DELAY (ps)		1
E7	gpmc_a20	0	0	0	0	231	0	CFG_GPMC_A20_OEN	mmc2_dat5
D6	gpmc_a21	0	0	0	0	39	0	CFG_GPMC_A21_OEN	mmc2_dat6
C5	gpmc_a22	0	0	0	0	91	0	CFG_GPMC_A22_OEN	mmc2_dat7
D7	gpmc_a24	0	0	0	0	176	0	CFG_GPMC_A24_OEN	mmc2_dat0
C6	gpmc_a25	0	0	0	0	0	0	CFG_GPMC_A25_OEN	mmc2_dat1
A5	gpmc_a26	0	0	0	0	101	0	CFG_GPMC_A26_OEN	mmc2_dat2
B6	gpmc_a27	0	0	0	0	0	0	CFG_GPMC_A27_OEN	mmc2_dat3
A6	gpmc_cs1	0	0	0	0	360	0	CFG_GPMC_CS1_OEN	mmc2_cmd

5.10.6.21.3 MMC3 and MMC4—SDIO/SD

MMC3 and MMC4 interfaces are compliant with the SDIO3.0 standard v1.0, SD Part E1 and for generic SDIO devices, it supports the following applications:

- MMC3 8-bit data and MMC4 4-bit data, SD Default speed, SDR
- MMC3 8-bit data and MMC4 4-bit data, SD High speed, SDR
- MMC3 8-bit data and MMC4 4-bit data, UHS-1 SDR12 (SD Standard v3.01), 4-bit data, SDR, half cycle
- MMC3 8-bit data and MMC4 4-bit data, UHS-I SDR25 (SD Standard v3.01), 4-bit data, SDR, half cycle
- MMC3 8-bit data, UHS-I SDR50

NOTE

The eMMC/SD/SDIOj (j = 3 to 4) controller is also referred to as MMCj.

NOTE

For more information, see the MMC/SDIO chapter of the Device TRM.

5.10.6.21.3.1 MMC3 and MMC4, SD Default Speed

Figure 5-99, Figure 5-100, and Table 5-145 through Table 5-148 present Timing requirements and Switching characteristics for MMC3 and MMC4 - SD Default speed in receiver and transmitter mode.

Table 5-145. Timing Requirements for MMC3 - Default Speed Mode (1)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DS5	$t_{su(cmdV-clkH)}$	Setup time, mmc3_cmd valid before mmc3_clk rising clock edge	5.11		ns
DS6	$t_{h(clkH-cmdV)}$	Hold time, mmc3_cmd valid after mmc3_clk rising clock edge	20.46		ns
DS7	$t_{su(dV-clkH)}$	Setup time, mmc3_dat[i:0] valid before mmc3_clk rising clock edge	5.11		ns
DS8	$t_{h(clkH-dV)}$	Hold time, mmc3_dat[i:0] valid after mmc3_clk rising clock edge	20.46		ns

(1) i in [i:0] = 7

Table 5-146. Switching Characteristics for MMC3 - SD/SDIO Default Speed Mode (2)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DS0	fop(clk)	Operating frequency, mmc3_clk		24	MHz
DS1	$t_{w(clkH)}$	Pulse duration, mmc3_clk high	0.5 × P-0.270 (1)		ns

Table 5-146. Switching Characteristics for MMC3 - SD/SDIO Default Speed Mode (2) (continued)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DS2	$t_{w(clkL)}$	Pulse duration, mmc3_clk low	0.5 × P-0.270 (1)		ns
DS3	$t_{d(clkL-cmdV)}$	Delay time, mmc3_clk falling clock edge to mmc3_cmd transition	-14.93	14.93	ns
DS4	$t_{d(clkL-dV)}$	Delay time, mmc3_clk falling clock edge to mmc3_dat[i:0] transition	-14.93	14.93	ns

(1) P = output mmc3_clk period in ns

(2) i in [i:0] = 7

Table 5-147. Timing Requirements for MMC4 - Default Speed Mode (1)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DS5	$t_{su(cmdV-clkH)}$	Setup time, mmc4_cmd valid before mmc4_clk rising clock edge	5.11		ns
DS6	$t_{h(clkH-cmdV)}$	Hold time, mmc4_cmd valid after mmc4_clk rising clock edge	20.46		ns
DS7	$t_{su(dV-clkH)}$	Setup time, mmc4_dat[i:0] valid before mmc4_clk rising clock edge	5.11		ns
DS8	$t_{h(clkH-dV)}$	Hold time, mmc4_dat[i:0] valid after mmc4_clk rising clock edge	20.46		ns

(1) i in [i:0] = 3

Table 5-148. Switching Characteristics for MMC4 - Default Speed Mode (2)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
DS0	fop(clk)	Operating frequency, mmc4_clk		24	MHz
DS1	$t_{w(clkH)}$	Pulse duration, mmc4_clk high	0.5 × P-0.270 (1)		ns
DS2	$t_{w(clkL)}$	Pulse duration, mmc4_clk low	0.5 × P-0.270 (1)		ns
DS3	$t_{d(clkL-cmdV)}$	Delay time, mmc4_clk falling clock edge to mmc4_cmd transition	-14.93	14.93	ns
DS4	$t_{d(clkL-dV)}$	Delay time, mmc4_clk falling clock edge to mmc4_dat[i:0] transition	-14.93	14.93	ns

(1) P = output mmc4_clk period in ns

(2) i in [i:0] = 3

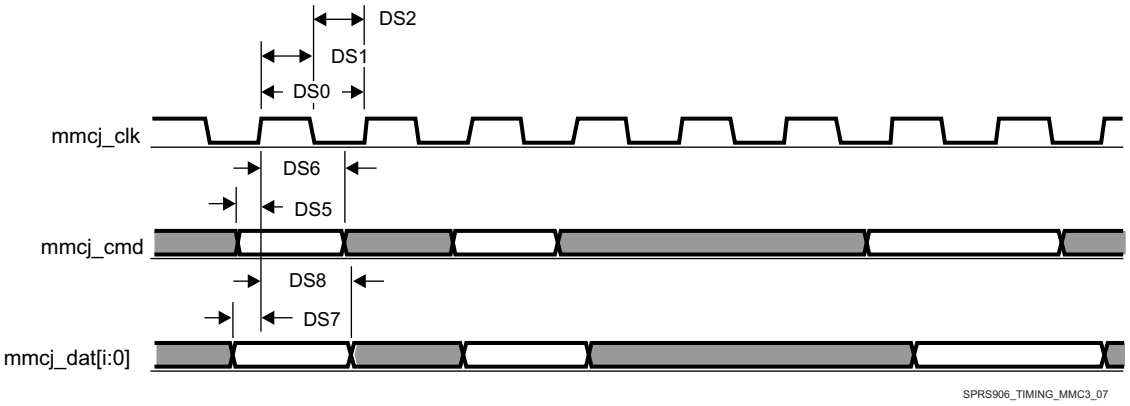


Figure 5-99. MMC/SD/SDIOj in - Default Speed - Receiver Mode

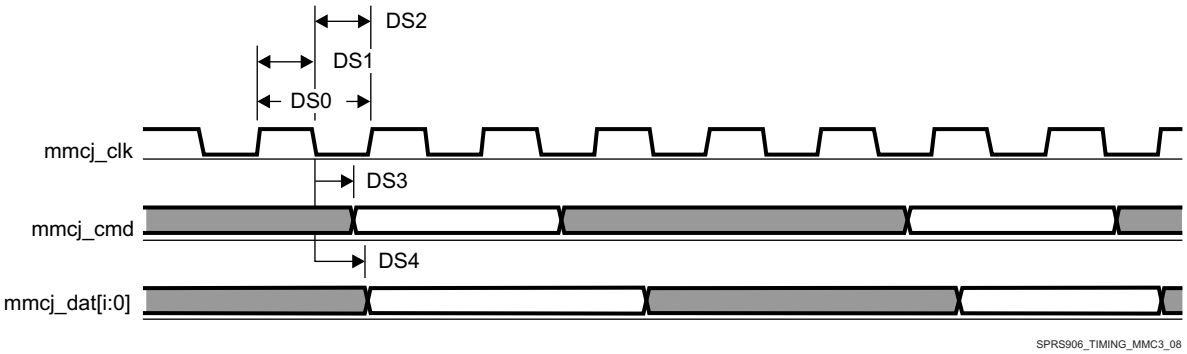


Figure 5-100. MMC/SD/SDIOj in - Default Speed - Transmitter Mode

5.10.6.21.3.2 MMC3 and MMC4, SD High Speed

Figure 5-101, Figure 5-102, and Table 5-149 through Table 5-152 present Timing requirements and Switching characteristics for MMC3 and MMC4 - SD and SDIO High speed in receiver and transmitter mode.

Table 5-149. Timing Requirements for MMC3 - SD/SDIO High Speed Mode (1)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
HS3	$t_{su(cmdV-clkH)}$	Setup time, mmc3_cmd valid before mmc3_clk rising clock edge	5.3		ns
HS4	$t_{h(clkH-cmdV)}$	Hold time, mmc3_cmd valid after mmc3_clk rising clock edge	2.6		ns
HS7	$t_{su(dV-clkH)}$	Setup time, mmc3_dat[i:0] valid before mmc3_clk rising clock edge	5.3		ns
HS8	$t_{h(clkH-dV)}$	Hold time, mmc3_dat[i:0] valid after mmc3_clk rising clock edge	2.6		ns

(1) i in [i:0] = 7

Table 5-150. Switching Characteristics for MMC3 - SD/SDIO High Speed Mode (2)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
HS1	fop(clk)	Operating frequency, mmc3_clk		48	MHz
HS2H	$t_{w(clkH)}$	Pulse duration, mmc3_clk high	0.5 × P-0.270 (1)		ns
HS2L	$t_{w(clkL)}$	Pulse duration, mmc3_clk low	0.5 × P-0.270 (1)		ns
HS5	$t_{d(clkL-cmdV)}$	Delay time, mmc3_clk falling clock edge to mmc3_cmd transition	-7.6	3.6	ns
HS6	$t_{d(clkL-dV)}$	Delay time, mmc3_clk falling clock edge to mmc3_dat[i:0] transition	-7.6	3.6	ns

(1) P = output mmc3_clk period in ns

(2) i in [i:0] = 7

Table 5-151. Timing Requirements for MMC4 - High Speed Mode (1)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
HS3	$t_{su(cmdV-clkH)}$	Setup time, mmc4_cmd valid before mmc4_clk rising clock edge	5.3		ns
HS4	$t_{h(clkH-cmdV)}$	Hold time, mmc4_cmd valid after mmc4_clk rising clock edge	1.6		ns
HS7	$t_{su(dV-clkH)}$	Setup time, mmc4_dat[i:0] valid before mmc4_clk rising clock edge	5.3		ns
HS8	$t_{h(clkH-dV)}$	Hold time, mmc4_dat[i:0] valid after mmc4_clk rising clock edge	1.6		ns

(1) i in [i:0] = 3

Table 5-152. Switching Characteristics for MMC4 - High Speed Mode (2)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
HS1	fop(clk)	Operating frequency, mmc4_clk		48	MHz
HS2H	$t_{w(clkH)}$	Pulse duration, mmc4_clk high	0.5 × P-0.270 (1)		ns

Table 5-152. Switching Characteristics for MMC4 - High Speed Mode (2) (continued)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
HS2L	$t_{w(clkL)}$	Pulse duration, mmc4_clk low	0.5 × P-0.270 (1)		ns
HS5	$t_{d(clkL-cmdV)}$	Delay time, mmc4_clk falling clock edge to mmc4_cmd transition	-8.8	6.6	ns
HS6	$t_{d(clkL-dV)}$	Delay time, mmc4_clk falling clock edge to mmc4_dat[i:0] transition	-8.8	6.6	ns

(1) P = output mmc4_clk period in ns

(2) i in [i:0] = 3

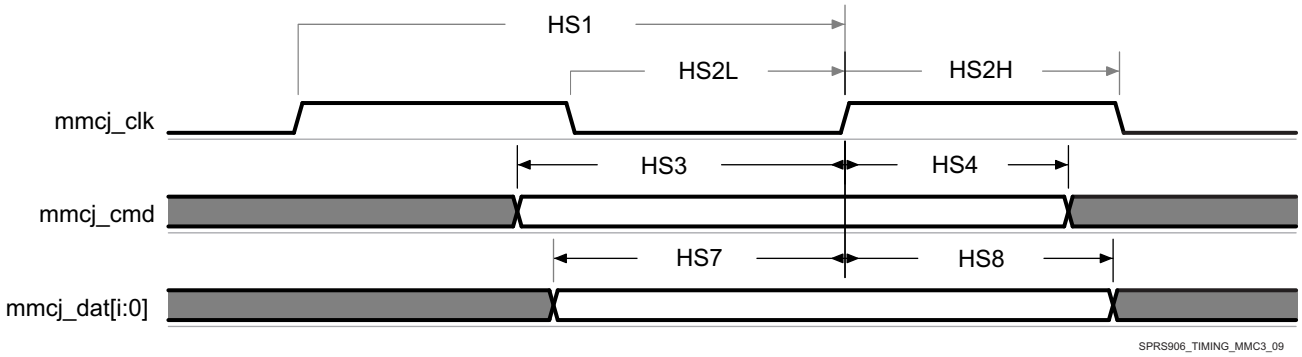


Figure 5-101. MMC/SD/SDIOj in - High Speed 3.3V Signaling - Receiver Mode

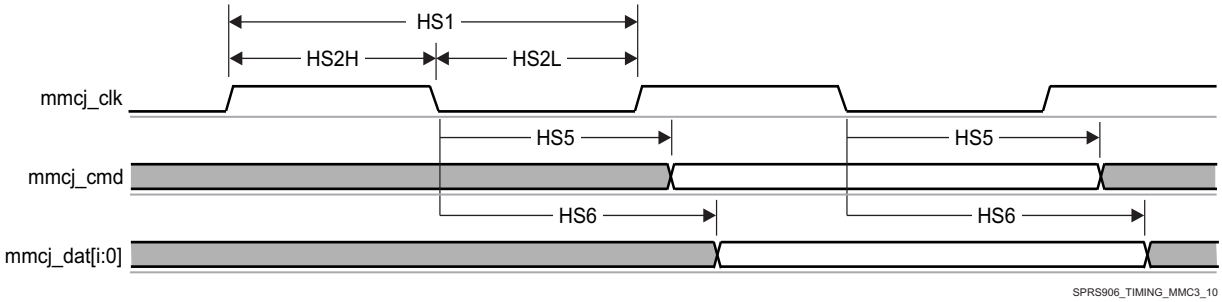


Figure 5-102. MMC/SD/SDIOj in - High Speed 3.3V Signaling - Transmitter Mode

5.10.6.21.3.3 MMC3 and MMC4, SD and SDIO SDR12 Mode

Figure 5-103, Figure 5-104, and Table 5-153, through Table 5-156 present Timing requirements and Switching characteristics for MMC3 and MMC4 - SD and SDIO SDR12 in receiver and transmitter mode.

Table 5-153. Timing Requirements for MMC3 - SDR12 Mode (1)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR125	$t_{su(cmdV-clkH)}$	Setup time, mmc3_cmd valid before mmc3_clk rising clock edge	25.99		ns
SDR126	$t_{h(clkH-cmdV)}$	Hold time, mmc3_cmd valid after mmc3_clk rising clock edge	1.6		ns
SDR127	$t_{su(dV-clkH)}$	Setup time, mmc3_dat[i:0] valid before mmc3_clk rising clock edge	25.99		ns
SDR128	$t_{h(clkH-dV)}$	Hold time, mmc3_dat[i:0] valid after mmc3_clk rising clock edge	1.6		ns

(1) i in [i:0] = 7

Table 5-154. Switching Characteristics for MMC3 - SDR12 Mode (2)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR120	fop(clk)	Operating frequency, mmc3_clk		24	MHz
SDR121	$t_{w(clkH)}$	Pulse duration, mmc3_clk high	0.5 × P-0.270 (1)		ns

Table 5-154. Switching Characteristics for MMC3 - SDR12 Mode (2) (continued)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR122	$t_{w(clkL)}$	Pulse duration, mmc3_clk low	0.5 × P-0.270 (1)		ns
SDR123	$t_{d(clkL-cmdV)}$	Delay time, mmc3_clk falling clock edge to mmc3_cmd transition	-19.13	16.93	ns
SDR124	$t_{d(clkL-dV)}$	Delay time, mmc3_clk falling clock edge to mmc3_dat[i:0] transition	-19.13	16.93	ns

(1) P = output mmc3_clk period in ns

(2) i in [i:0] = 7

Table 5-155. Timing Requirements for MMC4 - SDR12 Mode (1)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR125	$t_{su(cmdV-clkH)}$	Setup time, mmc4_cmd valid before mmc4_clk rising clock edge	25.99		ns
SDR126	$t_{h(clkH-cmdV)}$	Hold time, mmc4_cmd valid after mmc4_clk rising clock edge	1.6		ns
SDR127	$t_{su(dV-clkH)}$	Setup time, mmc4_dat[i:0] valid before mmc4_clk rising clock edge	25.99		ns
SDR128	$t_{h(clkH-dV)}$	Hold time, mmc4_dat[i:0] valid after mmc4_clk rising clock edge	1.6		ns

(1) j in [i:0] = 3

Table 5-156. Switching Characteristics for MMC4 - SDR12 Mode (2)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR120	fop(clk)	Operating frequency, mmc4_clk		24	MHz
SDR121	$t_{w(clkH)}$	Pulse duration, mmc4_clk high	0.5 × P-0.270 (1)		ns
SDR122	$t_{w(clkL)}$	Pulse duration, mmc4_clk low	0.5 × P-0.270 (1)		ns
SDR125	$t_{d(clkL-cmdV)}$	Delay time, mmc4_clk falling clock edge to mmc4_cmd transition	-19.13	16.93	ns
SDR126	$t_{d(clkL-dV)}$	Delay time, mmc4_clk falling clock edge to mmc4_dat[i:0] transition	-19.13	16.93	ns

(1) P = output mmc4_clk period in ns

(2) j in [i:0] = 3

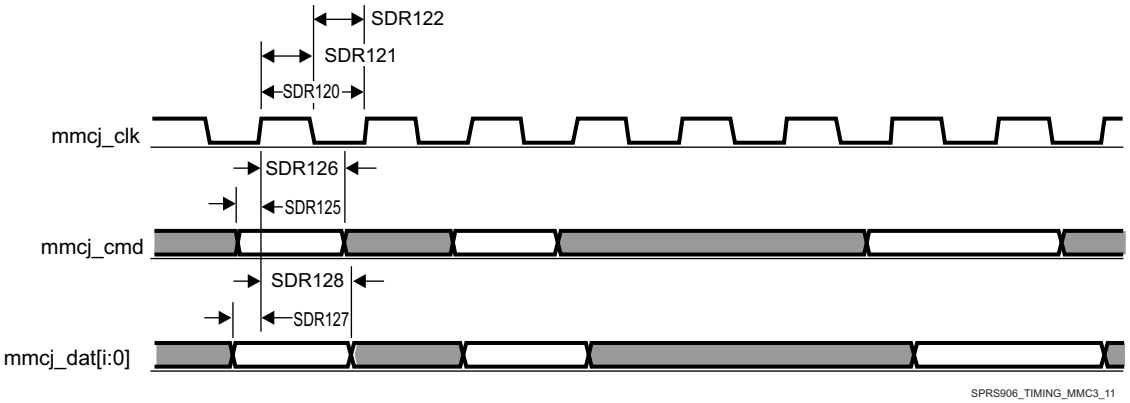


Figure 5-103. MMC/SD/SDIOj in - SDR12 - Receiver Mode

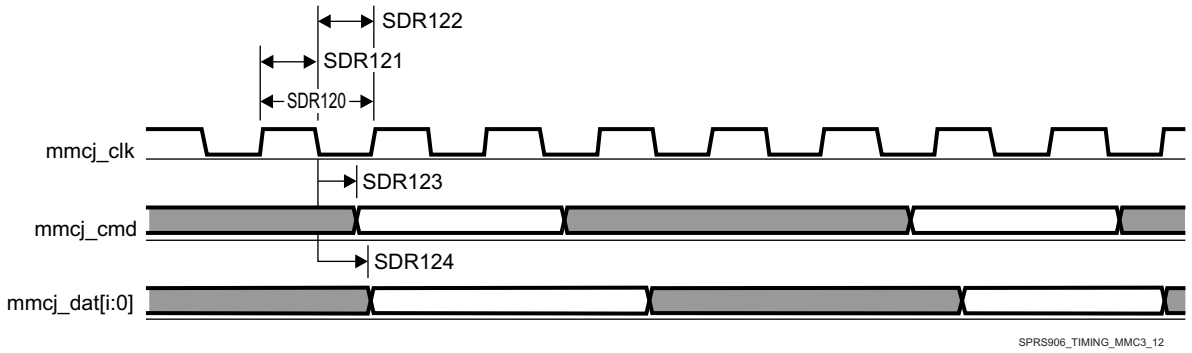


Figure 5-104. MMC/SD/SDIOj in - SDR12 - Transmitter Mode

5.10.6.21.3.4 MMC3 and MMC4, SD SDR25 Mode

Figure 5-105, Figure 5-106, and Table 5-157, through Table 5-160 present Timing requirements and Switching characteristics for MMC3 and MMC4 - SD and SDIO SDR25 in receiver and transmitter mode.

Table 5-157. Timing Requirements for MMC3 - SDR25 Mode (1)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR253	$t_{su(cmdV-clkH)}$	Setup time, mmc3_cmd valid before mmc3_clk rising clock edge	5.3		ns
SDR254	$t_{h(clkH-cmdV)}$	Hold time, mmc3_cmd valid after mmc3_clk rising clock edge	1.6		ns
SDR257	$t_{su(dV-clkH)}$	Setup time, mmc3_dat[i:0] valid before mmc3_clk rising clock edge	5.3		ns
SDR258	$t_{h(clkH-dV)}$	Hold time, mmc3_dat[i:0] valid after mmc3_clk rising clock edge	1.6		ns

(1) i in [i:0] = 7

Table 5-158. Switching Characteristics for MMC3 - SDR25 Mode (2)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR251	fop(clk)	Operating frequency, mmc3_clk		48	MHz
SDR252H	$t_{w(clkH)}$	Pulse duration, mmc3_clk high	0.5 × P-0.270 (1)		ns
SDR252L	$t_{w(clkL)}$	Pulse duration, mmc3_clk low	0.5 × P-0.270 (1)		ns
SDR255	$t_{d(clkL-cmdV)}$	Delay time, mmc3_clk falling clock edge to mmc3_cmd transition	-8.8	6.6	ns
SDR256	$t_{d(clkL-dV)}$	Delay time, mmc3_clk falling clock edge to mmc3_dat[i:0] transition	-8.8	6.6	ns

(1) P = output mmc3_clk period in ns

(2) i in [i:0] = 7

Table 5-159. Timing Requirements for MMC4 - SDR25 Mode (1)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR255	$t_{su(cmdV-clkH)}$	Setup time, mmc4_cmd valid before mmc4_clk rising clock edge	5.3		ns
SDR256	$t_{h(clkH-cmdV)}$	Hold time, mmc4_cmd valid after mmc4_clk rising clock edge	1.6		ns
SDR257	$t_{su(dV-clkH)}$	Setup time, mmc4_dat[i:0] valid before mmc4_clk rising clock edge	5.3		ns
SDR258	$t_{h(clkH-dV)}$	Hold time, mmc4_dat[i:0] valid after mmc4_clk rising clock edge	1.6		ns

(1) i in [i:0] = 3

Table 5-160. Switching Characteristics for MMC4 - SDR25 Mode (2)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR251	fop(clk)	Operating frequency, mmc4_clk		48	MHz
SDR252H	$t_{w(clkH)}$	Pulse duration, mmc4_clk high	0.5 × P-0.270 (1)		ns

Table 5-160. Switching Characteristics for MMC4 - SDR25 Mode (2) (continued)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR252L	$t_{w(clkL)}$	Pulse duration, mmc4_clk low	0.5 × P-0.270 (1)		ns
SDR255	$t_{d(clkL-cmdV)}$	Delay time, mmc4_clk falling clock edge to mmc4_cmd transition	-8.8	6.6	ns
SDR256	$t_{d(clkL-dV)}$	Delay time, mmc4_clk falling clock edge to mmc4_dat[i:0] transition	-8.8	6.6	ns

(1) P = output mmc4_clk period in ns

(2) i in [i:0] = 3

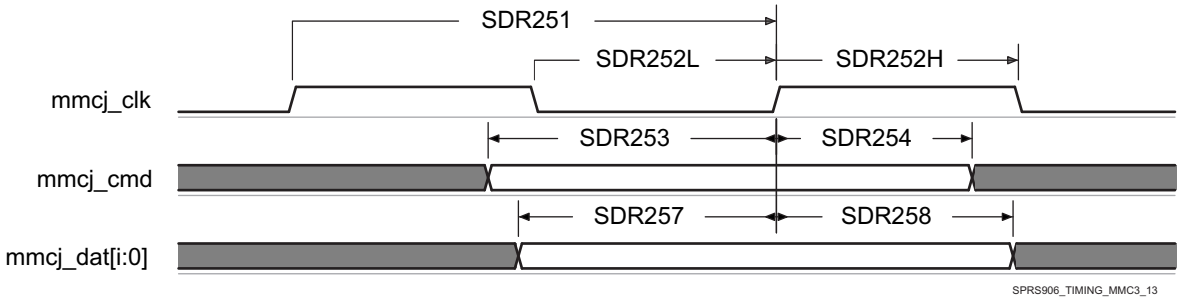


Figure 5-105. MMC/SD/SDIOj in - SDR25 - Receiver Mode

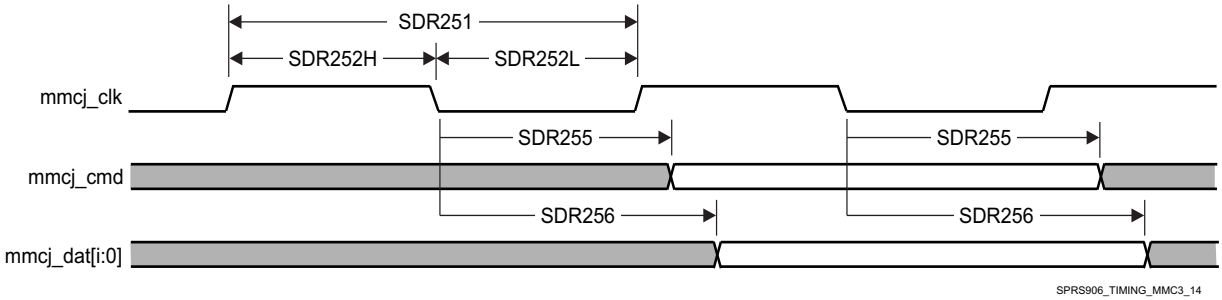


Figure 5-106. MMC/SD/SDIOj in - SDR25 - Transmitter Mode

5.10.6.21.3.5 MMC3 SDIO High-Speed UHS-I SDR50 Mode, Half Cycle

Figure 5-107, Figure 5-108, Table 5-161, and Table 5-162 present Timing requirements and Switching characteristics for MMC3 - SDIO High speed SDR50 in receiver and transmitter mode.

Table 5-161. Timing Requirements for MMC3 - SDR50 Mode (1)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR503	$t_{su(cmdV-clkH)}$	Setup time, mmc3_cmd valid before mmc3_clk rising clock edge	1.48		ns
SDR504	$t_{h(clkH-cmdV)}$	Hold time, mmc3_cmd valid after mmc3_clk rising clock edge	1.6		ns
SDR507	$t_{su(dV-clkH)}$	Setup time, mmc3_dat[i:0] valid before mmc3_clk rising clock edge	1.48		ns
SDR508	$t_{h(clkH-dV)}$	Hold time, mmc3_dat[i:0] valid after mmc3_clk rising clock edge	1.6		ns

(1) i in [i:0] = 7

Table 5-162. Switching Characteristics for MMC3 - SDR50 Mode (2)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR501	fop(clk)	Operating frequency, mmc3_clk		64	MHz
SDR502H	$t_{w(clkH)}$	Pulse duration, mmc3_clk high	0.5 × P-0.270 (1)		ns
SDR502L	$t_{w(clkL)}$	Pulse duration, mmc3_clk low	0.5 × P-0.270 (1)		ns

Table 5-162. Switching Characteristics for MMC3 - SDR50 Mode (2) (continued)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
SDR505	$t_{d(clkL-cmdV)}$	Delay time, mmc3_clk falling clock edge to mmc3_cmd transition	-3.66	1.46	ns
SDR506	$t_{d(clkL-dV)}$	Delay time, mmc3_clk falling clock edge to mmc3_dat[i:0] transition	-3.66	1.46	ns

(1) P = output mmc3_clk period in ns

(2) i in [i:0] = 7

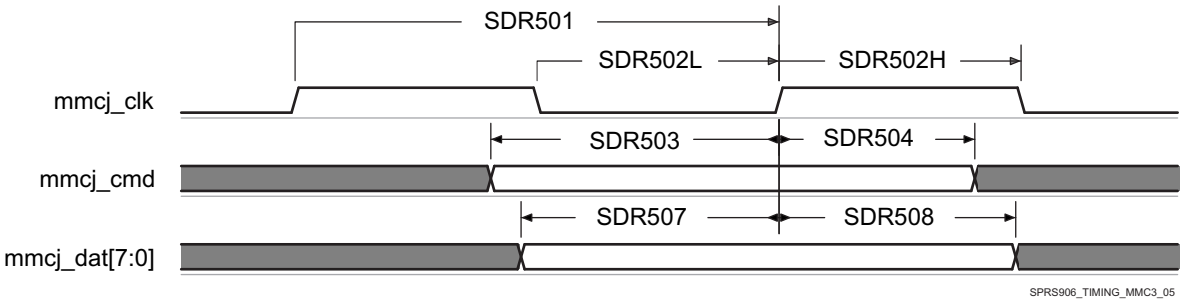


Figure 5-107. MMC/SD/SDIOj in - High Speed SDR50 - Receiver Mode

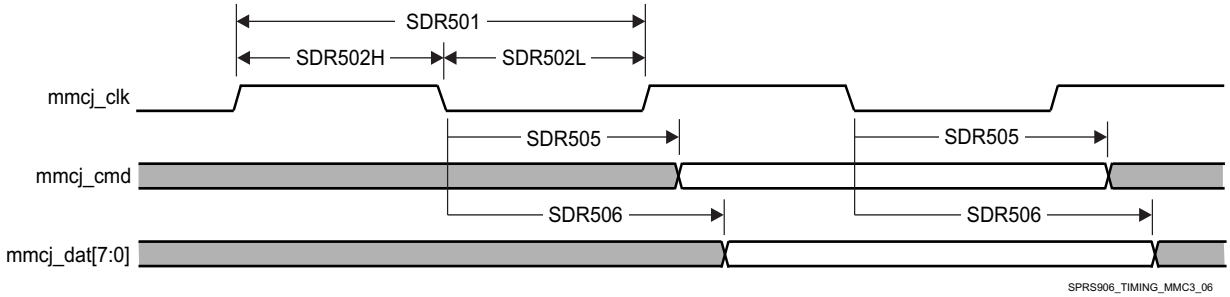


Figure 5-108. MMC/SD/SDIOj in - High Speed SDR50 - Transmitter Mode

NOTE

To configure the desired Manual IO Timing Mode the user must follow the steps described in section Manual IO Timing Modes of the Device TRM.

The associated registers to configure are listed in the **CFG REGISTER** column. For more information see the Control Module chapter in the Device TRM.

Manual IO Timings Modes must be used to ensure some IO timings for MMC3. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-163 *Manual Functions Mapping for MMC3* for a definition of the Manual modes.

Table 5-163 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-163. Manual Functions Mapping for MMC3

BALL	BALL NAME	MMC3_MANUAL1		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		0
Y2	mmc3_clk	1085	21	CFG_MMC3_CLK_IN	mmc3_clk
Y2	mmc3_clk	1269	0	CFG_MMC3_CLK_OUT	mmc3_clk
Y1	mmc3_cmd	0	0	CFG_MMC3_CMD_IN	mmc3_cmd
Y1	mmc3_cmd	128	0	CFG_MMC3_CMD_OEN	mmc3_cmd
Y1	mmc3_cmd	98	0	CFG_MMC3_CMD_OUT	mmc3_cmd
Y4	mmc3_dat0	0	0	CFG_MMC3_DAT0_IN	mmc3_dat0

Table 5-163. Manual Functions Mapping for MMC3 (continued)

BALL	BALL NAME	MMC3_MANUAL1		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		0
Y4	mmc3_dat0	362	0	CFG_MMC3_DAT0_OEN	mmc3_dat0
Y4	mmc3_dat0	0	0	CFG_MMC3_DAT0_OUT	mmc3_dat0
AA2	mmc3_dat1	7	0	CFG_MMC3_DAT1_IN	mmc3_dat1
AA2	mmc3_dat1	333	0	CFG_MMC3_DAT1_OEN	mmc3_dat1
AA2	mmc3_dat1	0	0	CFG_MMC3_DAT1_OUT	mmc3_dat1
AA3	mmc3_dat2	0	0	CFG_MMC3_DAT2_IN	mmc3_dat2
AA3	mmc3_dat2	402	0	CFG_MMC3_DAT2_OEN	mmc3_dat2
AA3	mmc3_dat2	0	0	CFG_MMC3_DAT2_OUT	mmc3_dat2
W2	mmc3_dat3	203	0	CFG_MMC3_DAT3_IN	mmc3_dat3
W2	mmc3_dat3	549	0	CFG_MMC3_DAT3_OEN	mmc3_dat3
W2	mmc3_dat3	1	0	CFG_MMC3_DAT3_OUT	mmc3_dat3
Y3	mmc3_dat4	121	0	CFG_MMC3_DAT4_IN	mmc3_dat4
Y3	mmc3_dat4	440	0	CFG_MMC3_DAT4_OEN	mmc3_dat4
Y3	mmc3_dat4	206	0	CFG_MMC3_DAT4_OUT	mmc3_dat4
AA1	mmc3_dat5	336	0	CFG_MMC3_DAT5_IN	mmc3_dat5
AA1	mmc3_dat5	283	0	CFG_MMC3_DAT5_OEN	mmc3_dat5
AA1	mmc3_dat5	174	0	CFG_MMC3_DAT5_OUT	mmc3_dat5
AA4	mmc3_dat6	320	0	CFG_MMC3_DAT6_IN	mmc3_dat6
AA4	mmc3_dat6	443	0	CFG_MMC3_DAT6_OEN	mmc3_dat6
AA4	mmc3_dat6	0	0	CFG_MMC3_DAT6_OUT	mmc3_dat6
AB1	mmc3_dat7	2	0	CFG_MMC3_DAT7_IN	mmc3_dat7
AB1	mmc3_dat7	344	0	CFG_MMC3_DAT7_OEN	mmc3_dat7
AB1	mmc3_dat7	0	0	CFG_MMC3_DAT7_OUT	mmc3_dat7

5.10.6.22 GPIO

The general-purpose interface combines eight general-purpose input/output (GPIO) banks. Each GPIO module provides up to 32 dedicated general-purpose pins with input and output capabilities; thus, the general-purpose interface supports up to 186 pins.

These pins can be configured for the following applications:

- Data input (capture)/output (drive)
- Keyboard interface with a debounce cell
- Interrupt generation in active mode upon the detection of external events. Detected events are processed by two parallel independent interrupt-generation submodules to support biprocessor operations
- Wake-up request generation in idle mode upon the detection of external events

NOTE

For more information, see the General-Purpose Interface chapter of the Device TRM.

NOTE

The general-purpose input/output i (i = 1 to 8) bank is also referred to as GPIOi.

5.10.6.23 PRU-ICSS

The device Programmable Real-Time Unit Subsystem and Industrial Communication Subsystem (PRU-ICSS) consists of dual 32-bit Load / Store RISC CPU cores - Programmable Real-Time Units (PRU0 and PRU1), shared, data, and instruction memories, internal peripheral modules, and an interrupt controller (PRU-ICSS_INTC). The programmable nature of the PRUs, along with their access to pins, events and all SoC resources, provides flexibility in implementing fast real-time responses, specialized data handling operations, customer peripheral interfaces, and in off-loading tasks from the other processor cores of the system-on-chip (SoC).

The each PRU-ICSS includes the following main features:

- 21x Enhanced GPIs (EGPIs) and 21x Enhanced GPOs (EGPOs) with asynchronous capture and serial support per each PRU CPU core
- One Ethernet MII_RT module (PRU-ICSS_MII_RT) with two MII ports and configurable connections to PRUs
- 1 MDIO Port (PRU-ICSS_MII_MDIO)
- One Industrial Ethernet Peripheral (IEP) to manage/generate Industrial Ethernet functions
- 1 x 16550-compatible UART with a dedicated 192 MHz clock to support 12Mbps Profibus
- 1 Industrial Ethernet timer with 7/9 capture and 8 compare events
- 1 Enhanced Capture Module (ECAP)
- 1 Interrupt Controller (PRU-ICSS_INTC)
- A flexible power management support
- Integrated switched central resource with programmable priority
- Parity control supported by all memories

CAUTION

The I/O timings provided in this section are valid only if signals within a single IOSET are used. The IOSETs are defined in the Table 5-186 and Table 5-187.

NOTE

For more information about PRU-ICSS subsystems interfaces, please see the device TRM.

NOTE

To configure the desired virtual mode the user must set MODESELECT bit and DELAYMODE bitfield for each corresponding pad control register.

The pad control registers are presented in Table 4-32 and described in Device TRM, *Control Module Chapter*.

5.10.6.23.1 Programmable Real-Time Unit (PRU-ICSS PRU)

5.10.6.23.1.1 PRU-ICSS PRU Direct Input/Output Mode Electrical Data and Timing

Table 5-164. PRU-ICSS PRU Timing Requirements - Direct Input Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{w(GPI)}$	Pulse width, GPI	2 × P (1)		ns
2	$t_{sk(GPI)}$	Skew between GPI[20:0] signals		4.5	ns

(1) ICSS_CLK clock period

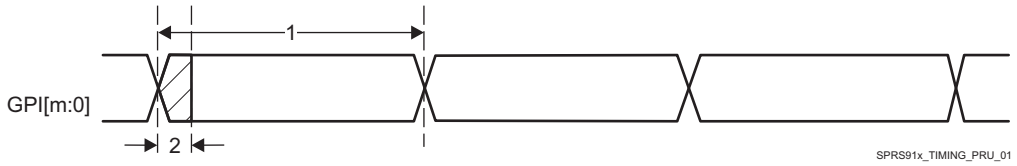


Figure 5-109. PRU-ICSS PRU Direct Input Timing

(1) m in GPI[m:0] = 20

Table 5-165. PRU-ICSS PRU Switching Requirements – Direct Output Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{w(GPO)}$	Pulse width, GPO	2 × P (1)		ns
2	$t_{sk(GPO)}$	Skew between GPO[20:0] signals		4.5	ns

(1) ICSS_CLK clock period

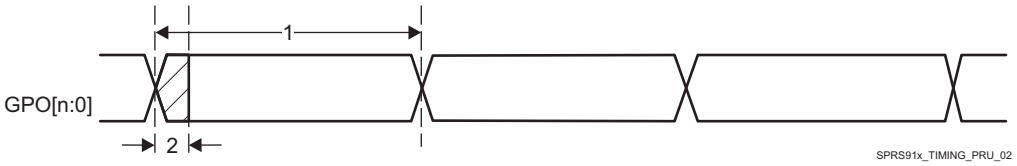


Figure 5-110. PRU-ICSS PRU Direct Output Timing

(1) n in GPO[n:0] = 20

5.10.6.23.1.2 PRU-ICSS PRU Parallel Capture Mode Electrical Data and Timing

Table 5-166. PRU-ICSS PRU Timing Requirements - Parallel Capture Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{w(CLOCKIN)}$	Cyle time, CLOCKIN	20		ns
2	$t_{w(CLOCKIN\_L)}$	Pulse duration, CLOCKIN low	9	11	ns
3	$t_{w(CLOCKIN\_H)}$	Pulse duration, CLOCKIN high	9	11	ns
4	$t_{su(DATAIN-CLOCKIN)}$	Setup time, DATAIN valid before CLOCKIN	4.5		ns
5	$t_{h(CLOCKIN-DATAIN)}$	Hold time, DATAIN valid after CLOCKIN	0		ns

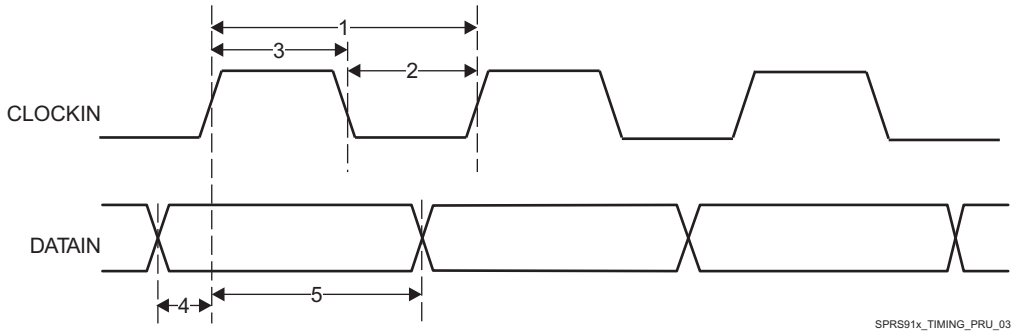


Figure 5-111. PRU-ICSS PRU Parallel Capture Timing - Rising Edge Mode

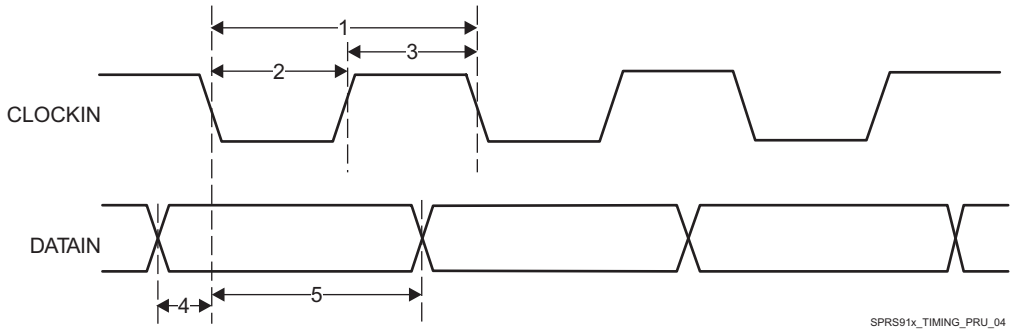


Figure 5-112. PRU-ICSS PRU Parallel Capture Timing - Falling Edge Mode

5.10.6.23.1.3 PRU-ICSS PRU Shift Mode Electrical Data and Timing

Table 5-167. PRU-ICSS PRU Timing Requirements – Shift In Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{c(DATAIN)}$	Cycle time, DATAIN	10.00		ns
2	$t_{w(DATAIN)}$	Pulse width, DATAIN	0.45 × P (1)		ns

(1) P = 10.00ns

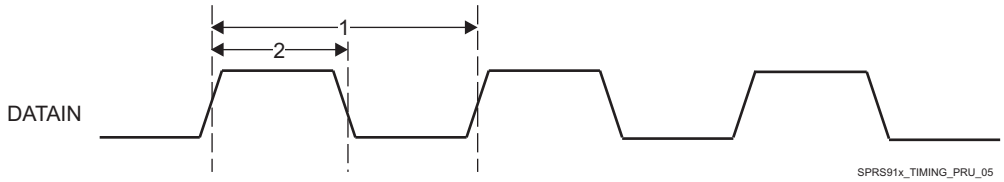


Figure 5-113. PRU-ICSS PRU Shift In Timing

Table 5-168. PRU-ICSS PRU Switching Requirements - Shift Out Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{c(CLOCKOUT)}$	Cycle time, CLOCKOUT	10.00		ns
2	$t_{w(CLOCKOUT)}$	Pulse width, CLOCKOUT	0.45 × P (1)		ns
3	$t_{d(CLOCKOUT-DATAOUT)}$	Delay time, CLOCKOUT to DATAOUT Valid	-3.00	3.60	ns

(1) P = 10.00ns

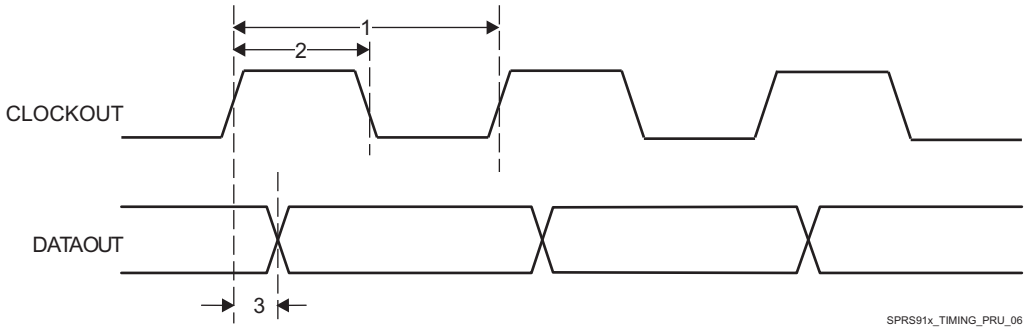


Figure 5-114. PRU-ICSS PRU Shift Out Timing

5.10.6.23.1.4 PRU-ICSS PRU Sigma Delta and EnDAT Modes

Table 5-169. PRU-ICSS PRU Timing Requirements - Sigma Delta Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	tw(SDx_CLK)	Pulse width, SDx_CLK	20		ns
2	tsu(SDx_D-SDx_CLK)	Setup time, SDx_D valid before SDx_CLK active edge	10		ns
3	th(SDx_CLK-SDx_D)	Hold time, SDx_D valid before SDx_CLK active edge	5		ns

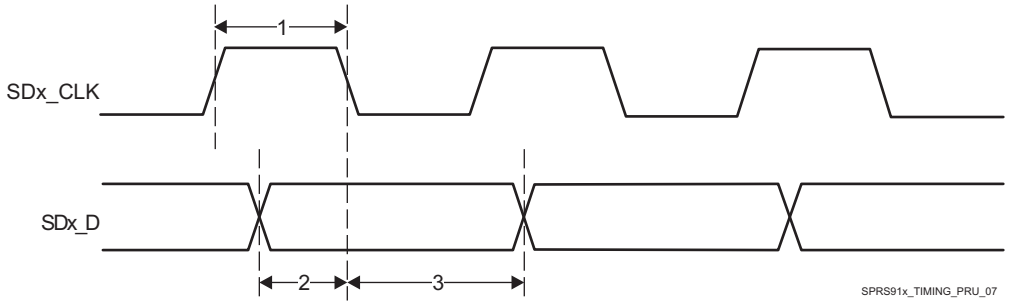


Figure 5-115. PRU-ICSS PRU SD_CLK Falling Active Edge

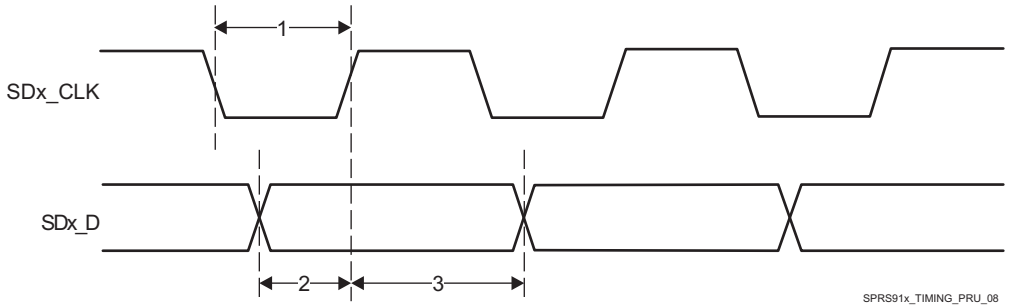


Figure 5-116. PRU-ICSS PRU SD_CLK Rising Active Edge

Table 5-170. PRU-ICSS PRU Timing Requirements - EnDAT Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	tw(ENDATx_IN)	Pulse width, ENDATx_IN	40		ns

Table 5-171. PRU-ICSS PRU Switching Requirements - EnDAT Mode

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
2	tw(ENDATx_CLK)	Pulse width, ENDATx_CLK	20		ns
3	td(ENDATx_OUT-ENDATx_CLK)	Delay time, ENDATx_CLK fall to ENDATx_OUT	-10	10	ns
4	td(ENDATx_OUT_EN-ENDATx_CLK)	Delay time, ENDATx_CLK Fall to ENDATx_OUT_EN	-10	10	ns

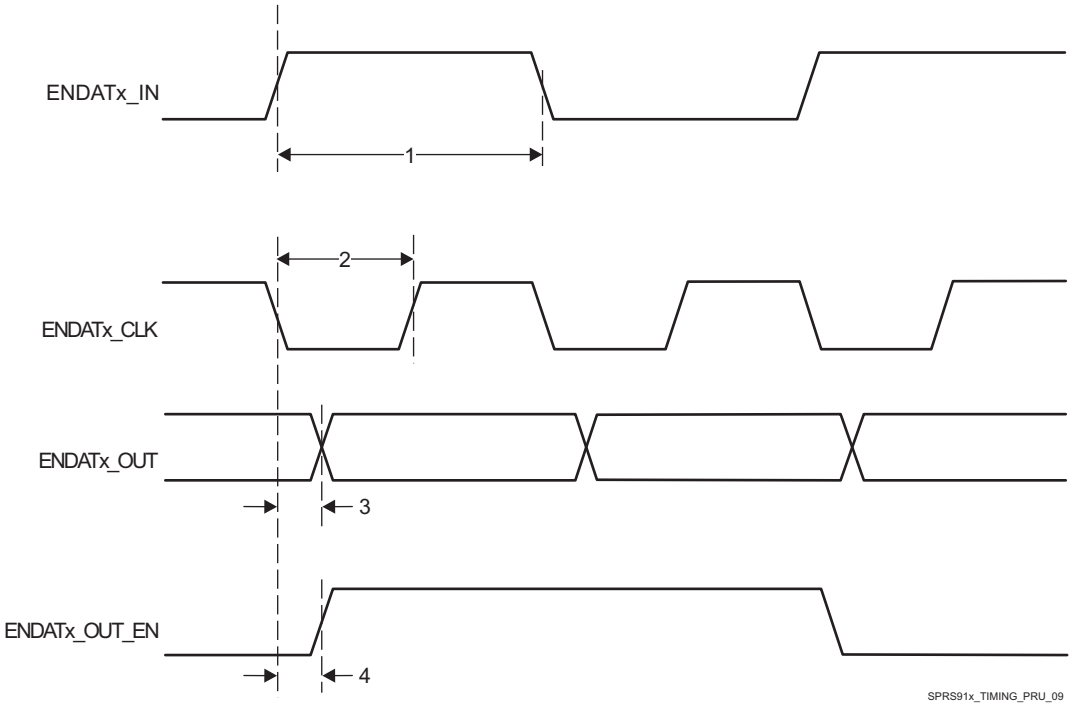


Figure 5-117. PRU-ICSS PRU EnDAT Timing

5.10.6.23.2 PRU-ICSS EtherCAT (PRU-ICSS ECAT)

5.10.6.23.2.1 PRU-ICSS ECAT Electrical Data and Timing

Table 5-172. PRU-ICSS ECAT Timing Requirements – Input Validated With LATCH_IN

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{w(EDIO\_LATCH\_IN)}$	Pulse width, EDIO_LATCH_IN	100.00		ns
2	$t_{su(EDIO\_DATA\_IN-EDIO\_LATCH\_IN)}$	Setup time, EDIO_DATA_IN valid before EDIO_LATCH_IN active edge	20.00		ns
3	$t_{h(EDIO\_LATCH\_IN-EDIO\_DATA\_IN)}$	Hold time, EDIO_DATA_IN valid after EDIO_LATCH_IN active edge	20.00		ns

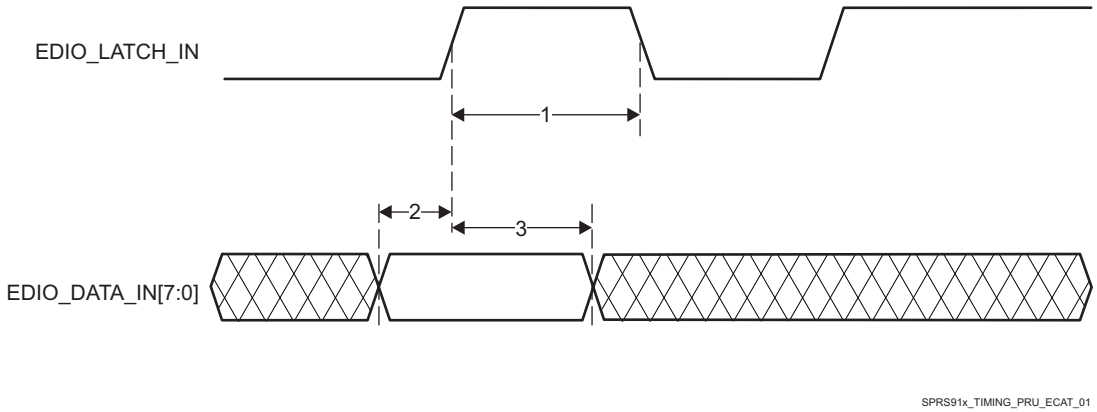


Figure 5-118. PRU-ICSS ECAT Input Validated with LATCH_IN Timing

Table 5-173. PRU-ICSS ECAT Timing Requirements – Input Validated With SYNCx

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{w(EDC\_SYNCx\_OUT)}$	Pulse width, EDC_SYNCx_OUT	100.00		ns
2	$t_{su(EDIO\_DATA\_IN-EDC\_SYNCx\_OUT)}$	Setup time, EDIO_DATA_IN valid before EDC_SYNCx_OUT active edge	20.00		ns
3	$t_{h(EDC\_SYNCx\_OUT-EDIO\_DATA\_IN)}$	Hold time, EDIO_DATA_IN valid after EDC_SYNCx_OUT active edge	20.00		ns

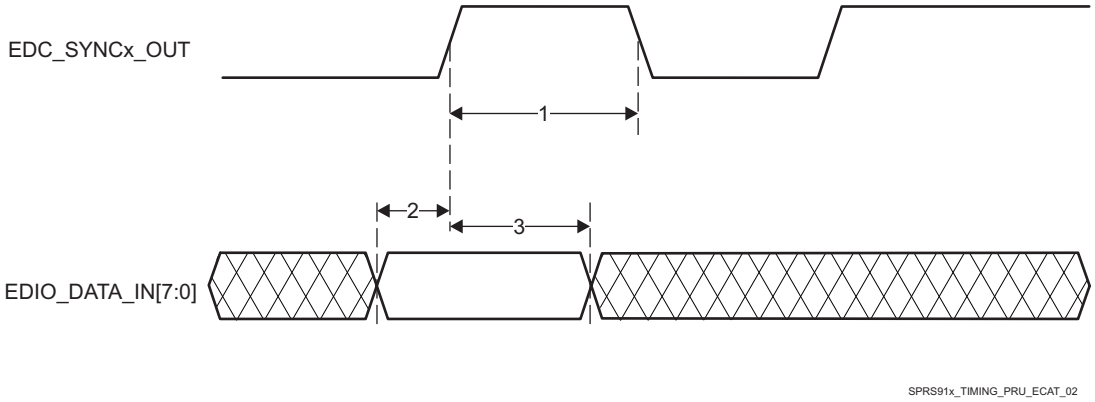


Figure 5-119. PRU-ICSS ECAT Input Validated With SYNCx Timing

Table 5-174. PRU-ICSS ECAT Timing Requirements – Input Validated With Start of Frame (SOF)

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{w(EDIO\_SOF)}$	Pulse duration, EDIO_SOF	4 × P (1)	5 × P (1)	ns
2	$t_{su(EDIO\_DATA\_IN-EDIO\_SOF)}$	Setup time, EDIO_DATA_IN valid before EDIO_SOF active edge	20.00		ns
3	$t_{h(EDIO\_SOF-EDIO\_DATA\_IN)}$	Hold time, EDIO_DATA_IN valid after EDIO_SOF active edge	20.00		ns

(1) ICSS_IEP_CLK clock period

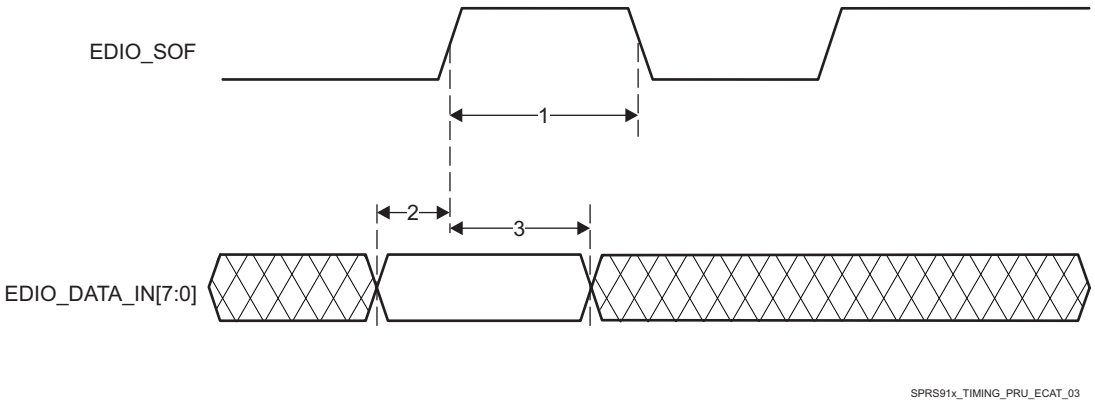


Figure 5-120. PRU-ICSS ECAT Input Validated With SOF

Table 5-175. PRU-ICSS ECAT Timing Requirements - LATCHx_IN

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{w(EDC\_LATCHx\_IN)}$	Pulse duration, EDC_LATCHx_IN	3 × P (1)		ns

(1) ICSS_IEP_CLK clock period

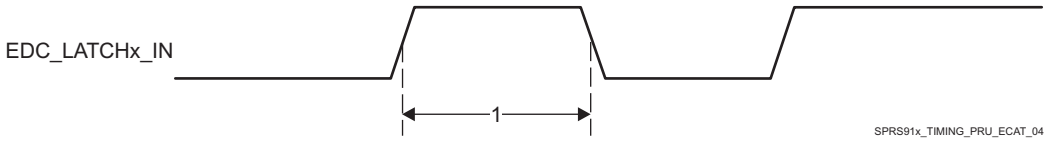


Figure 5-121. PRU-ICSS ECAT LATCHx_IN Timing

Table 5-176. PRU-ICSS ECAT Switching Requirements - Digital IOs

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{w(EDIO\_OUTVALID)}$	Pulse duration, EDIO_OUTVALID	14 × P (1)	32 × P (1)	ns
2	$t_{d(EDIO\_OUTVALID-EDIO\_DATA\_OUT)}$	Delay time, EDIO_OUTVALID to EDIO_DATA_OUT	0.00	18 × P (1)	ns
1	$t_{sk(EDIO\_DATA\_OUT)}$	EDIO_DATA_OUT skew		8	ns

(1) ICSS_IEP_CLK clock period

5.10.6.23.3 PRU-ICSS MII_RT and Switch

5.10.6.23.3.1 PRU-ICSS MDIO Electrical Data and Timing

Table 5-177. PRU-ICSS MDIO Timing Requirements – MDIO_DATA

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{su(MDIO-MDC)}$	Setup time, MDIO valid before MDC high	90		ns
2	$t_{h(MDIO-MDC)}$	Hold time, MDIO valid from MDC high	0		ns

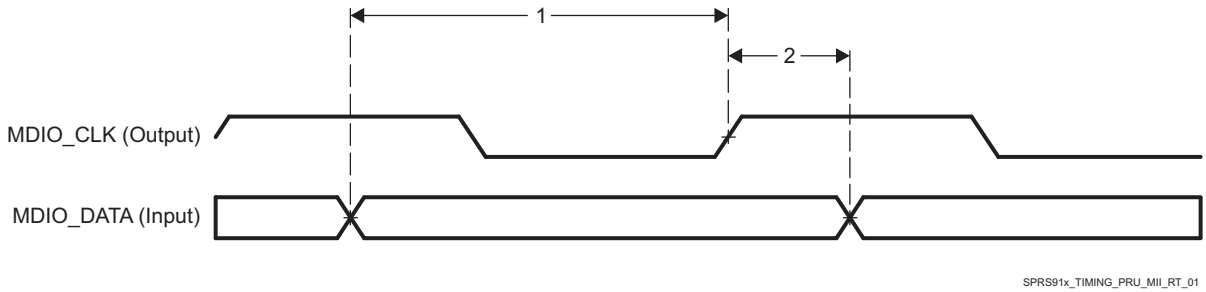


Figure 5-122. PRU-ICSS MDIO_DATA Timing - Input Mode

Table 5-178. PRU-ICSS MDIO Switching Characteristics - MDIO_CLK

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{c(MDC)}$	Cycle time, MDC	400		ns
2	$t_{w(MDCH)}$	Pulse duration, MDC high	160		ns
3	$t_{w(MDCL)}$	Pulse duration, MDC low	160		ns
4	$t_{t(MDC)}$	Transition time, MDC		5	ns

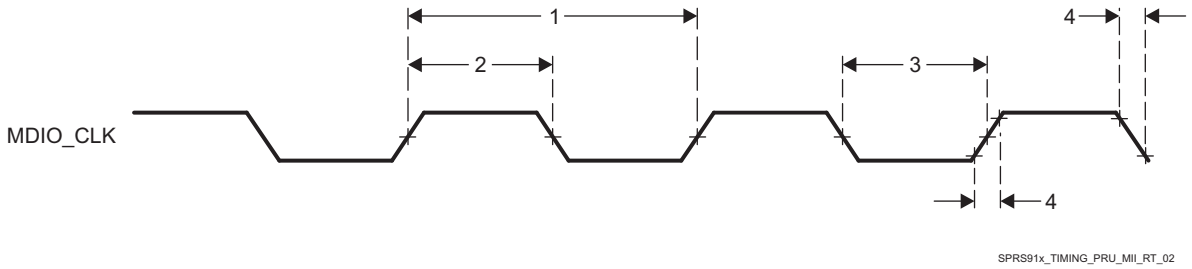


Figure 5-123. PRU-ICSS MDIO_CLK Timing

Table 5-179. PRU-ICSS MDIO Switching Characteristics – MDIO_DATA

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{d(MDC-MDIO)}$	Delay time, MDC high to MDIO valid	0	390	ns

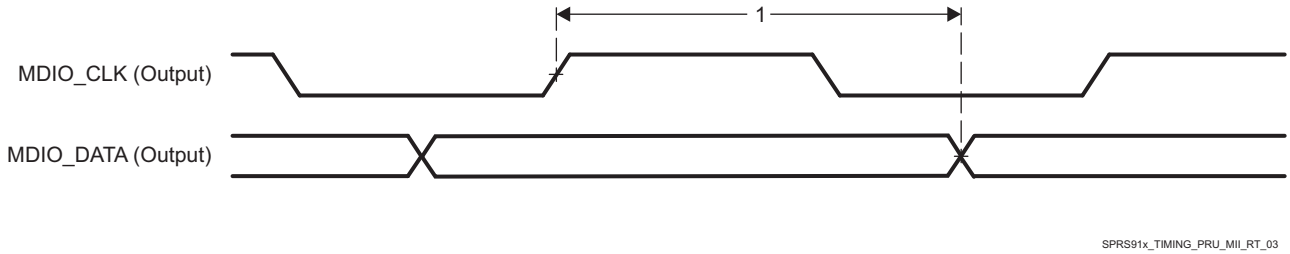


Figure 5-124. PRU-ICSS MDIO_DATA Timing – Output Mode

5.10.6.23.3.2 PRU-ICSS MII_RT Electrical Data and Timing

NOTE

In order to ensure the MII_RT IO timing values published in the device data manual, the ICSS_CLK clock must be configured for 200MHz (default value) and the TX_CLK_DELAY bitfield in the PRUSS_MII_RT_TXCFG0/1 register must be set to 6h (non-default value).

Table 5-180. PRU-ICSS MII_RT Timing Requirements – MII[x]_RXCLK

NO.	PARAMETER	DESCRIPTION	SPEED	MIN	MAX	UNIT
1	$t_{c(RX\_CLK)}$	Cycle time, RX_CLK	10 Mbps	399.96	400.04	ns
			100 Mbps	39.996	40.004	ns
2	$t_{w(RX\_CLKH)}$	Pulse duration, RX_CLK high	10 Mbps	140	260	ns
			100 Mbps	14	26	ns
3	$t_{w(RX\_CLKL)}$	Pulse duration, RX_CLK low	10 Mbps	140	260	ns
			100 Mbps	14	26	ns

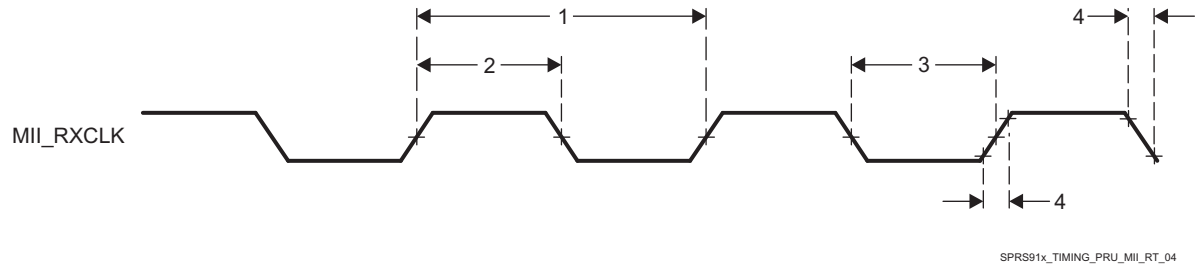


Figure 5-125. PRU-ICSS MII[x]_RXCLK Timing

Table 5-181. PRU-ICSS MII_RT Timing Requirements - MII[x]_TXCLK

NO.	PARAMETER	DESCRIPTION	SPEED	MIN	MAX	UNIT
1	$t_{c(TX\_CLK)}$	Cycle time, TX_CLK	10 Mbps	399.96	400.04	ns
			100 Mbps	39.996	40.004	ns
2	$t_{w(TX\_CLKH)}$	Pulse duration, TX_CLK high	10 Mbps	140	260	ns
			100 Mbps	14	26	ns
3	$t_{w(TX\_CLKL)}$	Pulse duration, TX_CLK low	10 Mbps	140	260	ns
			100 Mbps	14	26	ns
4	$t_{t(TX\_CLK)}$	Transition time, TX_CLK	10 Mbps		3	ns
			100 Mbps		3	ns

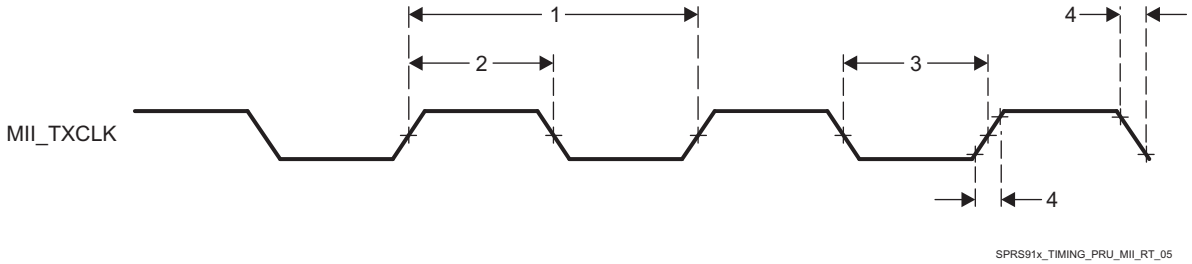


Figure 5-126. PRU-ICSS MII[x]_TXCLK Timing

Table 5-182. PRU-ICSS MII_RT Timing Requirements - MII_RXD[3:0], MII_RXDV, and MII_RXER

NO.	PARAMETER	DESCRIPTION	SPEED	MIN	MAX	UNIT
1	$t_{su(RXD-RX\_CLK)}$	Setup time, RXD[3:0] valid before RX_CLK	10 Mbps	8		ns
	$t_{su(RX\_DV-RX\_CLK)}$	Setup time, RX_DV valid before RX_CLK				
	$t_{su(RX\_ER-RX\_CLK)}$	Setup time, RX_ER valid before RX_CLK				
	$t_{su(RXD-RX\_CLK)}$	Setup time, RXD[3:0] valid before RX_CLK	100 Mbps	8		ns
	$t_{su(RX\_DV-RX\_CLK)}$	Setup time, RX_DV valid before RX_CLK				
	$t_{su(RX\_ER-RX\_CLK)}$	Setup time, RX_ER valid before RX_CLK				
2	$t_{h(RX\_CLK-RXD)}$	Hold time RXD[3:0] valid after RX_CLK	10 Mbps	8		ns
	$t_{h(RX\_CLK-RX\_DV)}$	Hold time RX_DV valid after RX_CLK				
	$t_{h(RX\_CLK-RX\_ER)}$	Hold time RX_ER valid after RX_CLK				
	$t_{h(RX\_CLK-RXD)}$	Hold time RXD[3:0] valid after RX_CLK	100 Mbps	8		ns
	$t_{h(RX\_CLK-RX\_DV)}$	Hold time RX_DV valid after RX_CLK				
	$t_{h(RX\_CLK-RX\_ER)}$	Hold time RX_ER valid after RX_CLK				

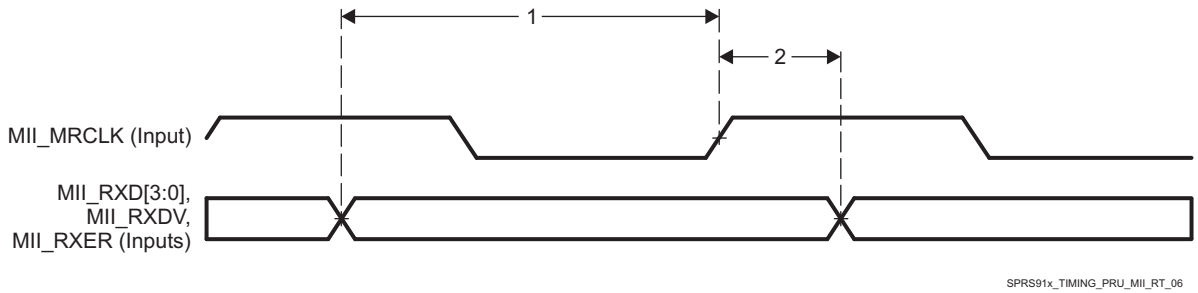


Figure 5-127. PRU-ICSS MII_RXD[3:0], MII_RXDV, and MII_RXER Timing

Table 5-183. PRU-ICSS MII_RT Switching Characteristics - MII_TXD[3:0] and MII_TXEN

NO.	PARAMETER	DESCRIPTION	SPEED	MIN	MAX	UNIT
1	$t_{d(TX\_CLK-TXD)}$	Delay time, TX_CLK high to TXD[3:0] valid	10 Mbps	5	25	ns
	$t_{d(TX\_CLK-TX\_EN)}$	Delay time, TX_CLK to TX_EN valid				
	$t_{d(TX\_CLK-TXD)}$	Delay time, TX_CLK high to TXD[3:0] valid	100 Mbps	5	25	ns
	$t_{d(TX\_CLK-TX\_EN)}$	Delay time, TX_CLK to TX_EN valid				

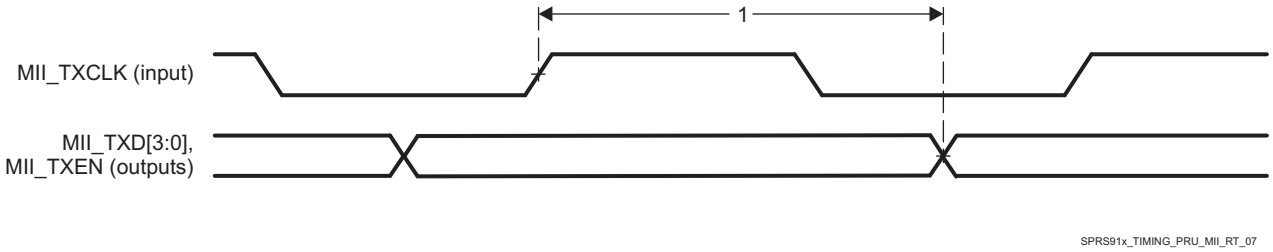


Figure 5-128. PRU-ICSS MII_TXD[3:0], MII_TXEN Timing

5.10.6.23.4 PRU-ICSS Universal Asynchronous Receiver Transmitter (PRU-ICSS UART)

Table 5-184. Timing Requirements for PRU-ICSS UART Receive

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
3	$t_{w(RX)}$	Pulse duration, receive start, stop, data bit	0.96U (1)	1.05U(1)	ns

(1) U = UART baud time = 1/programmed baud rate.

Table 5-185. Switching Characteristics Over Recommended Operating Conditions for PRU-ICSS UART Transmit

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$f$baud(baud)	Maximum programmable baud rate	0	12	MHz
2	$t_{w(TX)}$	Pulse duration, transmit start, stop, data bit	U - 2 (1)	U + 2(1)	ns

(1) U = UART baud time = 1/programmed baud rate.

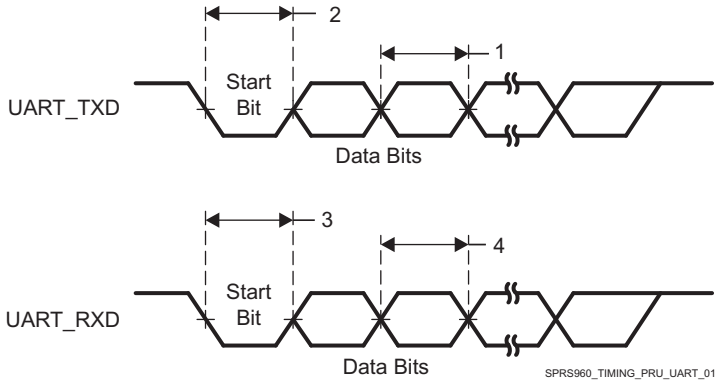


Figure 5-129. PRU-ICSS UART Timing

5.10.6.23.5 PRU-ICSS IOSETs

In Table 5-186 are presented the specific groupings of signals (IOSET) for use with PRU-ICSS1.

Table 5-186. PRU-ICSS1 IOSETs

SIGNALS	IOSET1		IOSET2	
	BALL	MUX	BALL	MUX
PRU-ICSS 1				
pr1_pru1_gpi20	D13	12		
pr1_pru1_gpi19	C13	12		
pr1_pru1_gpi18	E13	12		
pr1_pru1_gpi17	B13	12		
pr1_pru1_gpi16	F11	12		
pr1_pru1_gpi15	E11	12		
pr1_pru1_gpi14	A13	12		
pr1_pru1_gpi13	A12	12		
pr1_pru1_gpi12	B12	12		
pr1_pru1_gpi11	C11	12		
pr1_pru1_gpi10	D11	12		
pr1_pru1_gpo20	D13	13		
pr1_pru1_gpo19	C13	13		
pr1_pru1_gpo18	E13	13		
pr1_pru1_gpo17	B13	13		
pr1_pru1_gpo16	F11	13		
pr1_pru1_gpo15	E11	13		
pr1_pru1_gpo14	A13	13		
pr1_pru1_gpo13	A12	13		
pr1_pru1_gpo12	B12	13		
pr1_pru1_gpo11	C11	13		
pr1_pru1_gpo10	D11	13		
pr1_pru1_gpi9	B11	12		
pr1_pru1_gpi8	C10	12		
pr1_pru1_gpi7	D10	12		
pr1_pru1_gpi6	E10	12		
pr1_pru1_gpi5	B10	12		
pr1_pru1_gpi4	A10	12		
pr1_pru1_gpi3	F10	12		
pr1_pru1_gpi2	A11	12		
pr1_pru1_gpi1	A8	12		
pr1_pru1_gpi0	A9	12		
pr1_pru1_gpo9	B11	13		
pr1_pru1_gpo8	C10	13		
pr1_pru1_gpo7	D10	13		
pr1_pru1_gpo6	E10	13		
pr1_pru1_gpo5	B10	13		
pr1_pru1_gpo4	A10	13		
pr1_pru1_gpo3	F10	13		
pr1_pru1_gpo2	A11	13		
pr1_pru1_gpo1	A8	13		
pr1_pru1_gpo0	A9	13		
pr1_mii1_crs	D13	11		
pr1_mii1_rxlink	E13	11		
pr1_mii1_col	C13	11		

Table 5-186. PRU-ICSS1 IOSETs (continued)

SIGNALS	IOSET1		IOSET2	
	BALL	MUX	BALL	MUX
pr1_mii0_col	L5	11		
pr1_mii0_rxlink	L6	11		
pr1_mii0_crs	P4	11		
pr1_edio_data_out7			A7	13
pr1_edio_data_out6			B9	13
pr1_edio_data_out5			C8	13
pr1_edio_data_out4			B8	13
pr1_edio_data_out3			E8	13
pr1_edio_data_out2			C7	13
pr1_edio_data_out1			B7	13
pr1_edio_data_out0			D8	13
pr1_edio_data_in7			A7	12
pr1_edio_data_in6			B9	12
pr1_edio_data_in5			C8	12
pr1_edio_data_in4			B8	12
pr1_edio_data_in3			E8	12
pr1_edio_data_in2			C7	12
pr1_edio_data_in1			B7	12
pr1_edio_data_in0			D8	12
pr1_edio_sof			A11	11
pr1_edc_latch0_in			A9	11
pr1_edc_sync0_out			A8	11
pr1_uart0_cts_n	E8	11		
pr1_uart0_rts_n	B8	11		
pr1_uart0_txd	B9	11		
pr1_uart0_rxd	C8	11		
pr1_ecap0_ecap_capin_apwm_o	A7	11		
PRU-ICSS 1 MII				
pr1_mii1_txd3	B10	11		
pr1_mii1_txd2	E10	11		
pr1_mii1_txd1	B11	11		
pr1_mii1_txd0	D11	11		
pr1_mii1_rxd3	A12	11		
pr1_mii1_rxd2	A13	11		
pr1_mii1_rxd1	E11	11		
pr1_mii1_rxd0	F11	11		
pr1_mii1_rxdv	B12	11		
pr1_mii1_txen	A10	11		
pr1_mii1_rxer	B13	11		
pr1_mii_mr1_clk	C11	11		
pr1_mii_mt1_clk	F10	11		
pr1_mii0_txd3	P2	11		
pr1_mii0_txd2	N1	11		
pr1_mii0_txd1	N3	11		
pr1_mii0_txd0	N4	11		
pr1_mii0_rxd3	T4	11		

Table 5-186. PRU-ICSS1 IOSETs (continued)

SIGNALS	IOSET1		IOSET2	
	BALL	MUX	BALL	MUX
pr1_mii0_rxd2	T5	11		
pr1_mii0_rxd1	R2	11		
pr1_mii0_rxd0	R1	11		
pr1_mii0_rxdv	N5	11		
pr1_mii0_txen	P1	11		
pr1_mii0_rxer	P3	11		
pr1_mii_mt0_clk	N2	11		
pr1_mii_mr0_clk	N6	11		
pr1_mdio_mdclk	D10	11		
pr1_mdio_data	C10	11		

In Table 5-187, Table 5-188 and Table 5-189 are presented the specific groupings of signals (IOSET) for use with PRU-ICSS2.

Table 5-187. PRU-ICSS2 IOSETs

SIGNALS	IOSET1		IOSET2	
	BALL	MUX	BALL	MUX
PRU-ICSS 2				
pr2_pru1_gpi16	N4	12	E16	12
pr2_pru1_gpi15	N3	12	E17	12
pr2_pru1_gpi14	P1	12	A19	12
pr2_pru1_gpi13	N1	12	B18	12
pr2_pru1_gpi12	P2	12	B16	12
pr2_pru1_gpi11	N2	12	B17	12
pr2_pru1_gpi10	R1	12	A18	12
pr2_pru1_gpi9	R2	12	B14	12
pr2_pru1_gpi8	P3	12	D14	12
pr2_pru1_gpi7	P4	12	C16	12
pr2_pru1_gpi6	T5	12	J24	12
pr2_pru1_gpi5	T4	12	J25	12
pr2_pru1_gpi4	N6	12	AC4	12
pr2_pru1_gpi3	N5	12	AA5	12
pr2_pru1_gpi2	P5	12	U6	12
pr2_pru1_gpi1	L6	12	AC3	12
pr2_pru1_gpi0	L5	12	D23	12
pr2_pru1_gpo16	N4	13	E16	13
pr2_pru1_gpo15	N3	13	E17	13
pr2_pru1_gpo14	P1	13	A19	13
pr2_pru1_gpo13	N1	13	B18	13
pr2_pru1_gpo12	P2	13	B16	13
pr2_pru1_gpo11	N2	13	B17	13
pr2_pru1_gpo10	R1	13	A18	13
pr2_pru1_gpo9	R2	13	B14	13
pr2_pru1_gpo8	P3	13	D14	13
pr2_pru1_gpo7	P4	13	C16	13
pr2_pru1_gpo6	T5	13	J24	13

Table 5-187. PRU-ICSS2 IOSETs (continued)

SIGNALS	IOSET1		IOSET2	
	BALL	MUX	BALL	MUX
pr2_pru1_gpo5	T4	13	J25	13
pr2_pru1_gpo4	N6	13	AC4	13
pr2_pru1_gpo3	N5	13	AA5	13
pr2_pru1_gpo2	P5	13	U6	13
pr2_pru1_gpo1	L6	13	AC3	13
pr2_pru1_gpo0	L5	13	D23	13
pr2_pru0_gpi20			F16	12
pr2_pru0_gpi19			D19	12
pr2_pru0_gpi18			E19	12
pr2_pru0_gpi17			B21	12
pr2_pru0_gpi16			A21	12
pr2_pru0_gpi15			B23	12
pr2_pru0_gpi14			B22	12
pr2_pru0_gpi13			A23	12
pr2_pru0_gpi12			A22	12
pr2_pru0_gpi11			AB1	12
pr2_pru0_gpi10			AA4	12
pr2_pru0_gpi9			AA1	12
pr2_pru0_gpi8			Y3	12
pr2_pru0_gpi7			W2	12
pr2_pru0_gpi6			AA3	12
pr2_pru0_gpi5			AA2	12
pr2_pru0_gpi4			Y4	12
pr2_pru0_gpi3			Y1	12
pr2_pru0_gpi2			Y2	12
pr2_pru0_gpi1			Y6	12
pr2_pru0_gpi0			Y5	12
pr2_pru0_gpo20			F16	13
pr2_pru0_gpo19			D19	13
pr2_pru0_gpo18			E19	13
pr2_pru0_gpo17			B21	13
pr2_pru0_gpo16			A21	13
pr2_pru0_gpo15			B23	13
pr2_pru0_gpo14			B22	13
pr2_pru0_gpo13			A23	13
pr2_pru0_gpo12			A22	13
pr2_pru0_gpo11			AB1	13
pr2_pru0_gpo10			AA4	13
pr2_pru0_gpo9			AA1	13
pr2_pru0_gpo8			Y3	13
pr2_pru0_gpo7			W2	13
pr2_pru0_gpo6			AA3	13
pr2_pru0_gpo5			AA2	13
pr2_pru0_gpo4			Y4	13
pr2_pru0_gpo3			Y1	13
pr2_pru0_gpo2			Y2	13

Table 5-187. PRU-ICSS2 IOSETs (continued)

SIGNALS	IOSET1		IOSET2	
	BALL	MUX	BALL	MUX
pr2_pru0_gpo1			Y6	13
pr2_pru0_gpo0			Y5	13
pr2_mii1_crs	J24	11		
pr2_mii1_rxlink	B23	11		
pr2_mii0_crs	A22	11		
pr2_mii0_rxlink	B21	11		
pr2_mii0_col	A23	11		
pr2_mii1_col	J25	11		
PRU-ICSS 2 MII				
pr2_mii1_txd3	Y2	11		
pr2_mii1_txd2	Y1	11		
pr2_mii1_txd1	Y4	11		
pr2_mii1_txd0	AA2	11		
pr2_mii1_rxd3	Y3	11		
pr2_mii1_rxd2	AA1	11		
pr2_mii1_rxd1	AA4	11		
pr2_mii1_rxd0	AB1	11		
pr2_mii_mr1_clk	AA3	11		
pr2_mii1_rxer	B22	11		
pr2_mii_mt1_clk	Y5	11		
pr2_mii1_rxdv	W2	11		
pr2_mii1_txen	Y6	11		
pr2_mii0_txd3	B17	11		
pr2_mii0_txd2	B16	11		
pr2_mii0_txd1	B18	11		
pr2_mii0_txd0	A19	11		
pr2_mii0_rxd3	F16	11		
pr2_mii0_rxd2	E19	11		
pr2_mii0_rxd1	D19	11		
pr2_mii0_rxd0	A21	11		
pr2_mii_mr0_clk	E17	11		
pr2_mii0_rxer	D14	11		
pr2_mii_mt0_clk	B14	11		
pr2_mii0_rxdv	E16	11		
pr2_mii0_txen	A18	11		
pr2_mdio_mdclk	C16	11	AA5	11
pr2_mdio_data	C17	11	AC4	11

Table 5-188. PRU-ICSS2 IOSETs (EnDAT)(1)

SIGNALS	IOSET3		IOSET4	
	BALL	MUX	BALL	MUX
PRU-ICSS 2 EnDAT				
pr2_pru1_endat0_clk	L5	13	D23	13
pr2_pru1_endat0_out	L6	13	AC3	13
pr2_pru1_endat0_out_en	P5	13	U6	13
pr2_pru1_endat1_clk	N5	13	AA5	13

Table 5-188. PRU-ICSS2 IOSETs (EnDAT)(1) (continued)

SIGNALS	IOSET3		IOSET4	
	BALL	MUX	BALL	MUX
pr2_pru1_endat1_out	N6	13	AC4	13
pr2_pru1_endat1_out_en	T4	13	J25	13
pr2_pru1_endat2_clk	T5	13	J24	13
pr2_pru1_endat2_out	P4	13	C16	13
pr2_pru1_endat2_out_en	P3	13	D14	13
pr2_pru1_endat0_in	R2	12	B14	12
pr2_pru1_endat1_in	R1	12	A18	12
pr2_pru1_endat2_in	N2	12	B17	12

(1) These signals are internally muxed with the PRU GPI/GPO signals. Refer to the PRU chapter in the TRM for more details about the PRU-ICSS internal wrapper multiplexing.

Table 5-189. PRU-ICSS2 IOSETs (Sigma Delta)(1)

SIGNALS	IOSET4	
	BALL	MUX
PRU-ICSS 2 SD		
pr2_pru0_sd0_clk	Y5	12
pr2_pru0_sd0_d	Y6	12
pr2_pru0_sd1_clk	Y2	12
pr2_pru0_sd1_d	Y1	12
pr2_pru0_sd2_clk	Y4	12
pr2_pru0_sd2_d	AA2	12
pr2_pru0_sd3_clk	AA3	12
pr2_pru0_sd3_d	W2	12
pr2_pru0_sd4_clk	Y3	12
pr2_pru0_sd4_d	AA1	12
pr2_pru0_sd5_clk	AA4	12
pr2_pru0_sd5_d	AB1	12
pr2_pru0_sd6_clk	A22	12
pr2_pru0_sd6_d	A23	12
pr2_pru0_sd7_clk	B22	12
pr2_pru0_sd7_d	B23	12
pr2_pru0_sd8_clk	A21	12
pr2_pru0_sd8_d	B21	12

(1) These signals are internally muxed with the PRU GPI/GPO signals. Refer to the PRU chapter in the TRM for more details about the PRU-ICSS internal wrapper multiplexing.

5.10.6.23.6 PRU-ICSS Manual Functional Mapping

NOTE

To configure the desired Manual IO Timing Mode the user must follow the steps described in section "*Manual IO Timing Modes*" of the Device TRM.

The associated registers to configure are listed in the **CFG REGISTER** column. For more information see the *Control Module Chapter* in the Device TRM.

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS1 PRU1 Direct Input mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-190 *Manual Functions Mapping for PRU-ICSS1 PRU1 Direct Input mode* for a definition of the Manual modes.

Table 5-190 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-190. Manual Functions Mapping for PRU-ICSS1 PRU1 Direct Input mode

BALL	BALL NAME	PR1_PRU1_DIR_IN_MANUAL		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		12
D10	vin2a_d10	0	800	CFG_VIN2A_D10_IN	pr1_pru1_gpi7
C10	vin2a_d11	0	0	CFG_VIN2A_D11_IN	pr1_pru1_gpi8
B11	vin2a_d12	0	200	CFG_VIN2A_D12_IN	pr1_pru1_gpi9
D11	vin2a_d13	0	0	CFG_VIN2A_D13_IN	pr1_pru1_gpi10
C11	vin2a_d14	0	0	CFG_VIN2A_D14_IN	pr1_pru1_gpi11
B12	vin2a_d15	0	400	CFG_VIN2A_D15_IN	pr1_pru1_gpi12
A12	vin2a_d16	0	300	CFG_VIN2A_D16_IN	pr1_pru1_gpi13
A13	vin2a_d17	0	400	CFG_VIN2A_D17_IN	pr1_pru1_gpi14
E11	vin2a_d18	0	900	CFG_VIN2A_D18_IN	pr1_pru1_gpi15
F11	vin2a_d19	0	1500	CFG_VIN2A_D19_IN	pr1_pru1_gpi16
B13	vin2a_d20	0	100	CFG_VIN2A_D20_IN	pr1_pru1_gpi17
E13	vin2a_d21	0	500	CFG_VIN2A_D21_IN	pr1_pru1_gpi18
C13	vin2a_d22	0	500	CFG_VIN2A_D22_IN	pr1_pru1_gpi19
D13	vin2a_d23	0	600	CFG_VIN2A_D23_IN	pr1_pru1_gpi20
A9	vin2a_d3	0	900	CFG_VIN2A_D3_IN	pr1_pru1_gpi0
A8	vin2a_d4	0	100	CFG_VIN2A_D4_IN	pr1_pru1_gpi1
A11	vin2a_d5	0	600	CFG_VIN2A_D5_IN	pr1_pru1_gpi2
F10	vin2a_d6	0	200	CFG_VIN2A_D6_IN	pr1_pru1_gpi3
A10	vin2a_d7	0	400	CFG_VIN2A_D7_IN	pr1_pru1_gpi4
B10	vin2a_d8	0	500	CFG_VIN2A_D8_IN	pr1_pru1_gpi5
E10	vin2a_d9	0	600	CFG_VIN2A_D9_IN	pr1_pru1_gpi6

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS1 PRU1 Direct Output mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-191 *Manual Functions Mapping for PRU-ICSS1 PRU1 Direct Output mode* for a definition of the Manual modes.

Table 5-191 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-191. Manual Functions Mapping for PRU-ICSS1 PRU1 Direct Output mode

BALL	BALL NAME	PR1_PRU1_DIR_OUT_MANUAL		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		13
D10	vin2a_d10	0	1000	CFG_VIN2A_D10_OUT	pr1_pru1_gpo7
C10	vin2a_d11	0	1300	CFG_VIN2A_D11_OUT	pr1_pru1_gpo8
B11	vin2a_d12	0	2300	CFG_VIN2A_D12_OUT	pr1_pru1_gpo9
D11	vin2a_d13	0	2200	CFG_VIN2A_D13_OUT	pr1_pru1_gpo10
C11	vin2a_d14	0	1800	CFG_VIN2A_D14_OUT	pr1_pru1_gpo11
B12	vin2a_d15	0	1800	CFG_VIN2A_D15_OUT	pr1_pru1_gpo12
A12	vin2a_d16	0	1600	CFG_VIN2A_D16_OUT	pr1_pru1_gpo13
A13	vin2a_d17	0	2000	CFG_VIN2A_D17_OUT	pr1_pru1_gpo14
E11	vin2a_d18	0	700	CFG_VIN2A_D18_OUT	pr1_pru1_gpo15
F11	vin2a_d19	0	700	CFG_VIN2A_D19_OUT	pr1_pru1_gpo16
B13	vin2a_d20	0	500	CFG_VIN2A_D20_OUT	pr1_pru1_gpo17
E13	vin2a_d21	0	400	CFG_VIN2A_D21_OUT	pr1_pru1_gpo18

Table 5-191. Manual Functions Mapping for PRU-ICSS1 PRU1 Direct Output mode (continued)

BALL	BALL NAME	PR1_PRU1_DIR_OUT_MANUAL		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		13
C13	vin2a_d22	0	0	CFG_VIN2A_D22_OUT	pr1_pru1_gpo19
D13	vin2a_d23	0	400	CFG_VIN2A_D23_OUT	pr1_pru1_gpo20
A9	vin2a_d3	0	2200	CFG_VIN2A_D3_OUT	pr1_pru1_gpo0
A8	vin2a_d4	540	2800	CFG_VIN2A_D4_OUT	pr1_pru1_gpo1
A11	vin2a_d5	0	400	CFG_VIN2A_D5_OUT	pr1_pru1_gpo2
F10	vin2a_d6	0	1500	CFG_VIN2A_D6_OUT	pr1_pru1_gpo3
A10	vin2a_d7	0	2200	CFG_VIN2A_D7_OUT	pr1_pru1_gpo4
B10	vin2a_d8	0	2600	CFG_VIN2A_D8_OUT	pr1_pru1_gpo5
E10	vin2a_d9	0	2300	CFG_VIN2A_D9_OUT	pr1_pru1_gpo6

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS1 PRU1 Parallel Capture Mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-192 *Manual Functions Mapping for PRU-ICSS1 PRU1 Parallel Capture Mode* for a definition of the Manual modes.

Table 5-192 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-192. Manual Functions Mapping for PRU-ICSS1 PRU1 Parallel Capture Mode

BALL	BALL NAME	PR1_PRU1_PAR_CAP_MANUAL		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		12
D10	vin2a_d10	1535	0	CFG_VIN2A_D10_IN	pr1_pru1_gpi7
C10	vin2a_d11	1151	0	CFG_VIN2A_D11_IN	pr1_pru1_gpi8
B11	vin2a_d12	1173	0	CFG_VIN2A_D12_IN	pr1_pru1_gpi9
D11	vin2a_d13	970	0	CFG_VIN2A_D13_IN	pr1_pru1_gpi10
C11	vin2a_d14	1196	0	CFG_VIN2A_D14_IN	pr1_pru1_gpi11
B12	vin2a_d15	1286	0	CFG_VIN2A_D15_IN	pr1_pru1_gpi12
A12	vin2a_d16	1354	0	CFG_VIN2A_D16_IN	pr1_pru1_gpi13
A13	vin2a_d17	1331	0	CFG_VIN2A_D17_IN	pr1_pru1_gpi14
E11	vin2a_d18	2097	0	CFG_VIN2A_D18_IN	pr1_pru1_gpi15
F11	vin2a_d19	0	453	CFG_VIN2A_D19_IN	pr1_pru1_gpi16
A9	vin2a_d3	1566	0	CFG_VIN2A_D3_IN	pr1_pru1_gpi0
A8	vin2a_d4	1012	0	CFG_VIN2A_D4_IN	pr1_pru1_gpi1
A11	vin2a_d5	1337	0	CFG_VIN2A_D5_IN	pr1_pru1_gpi2
F10	vin2a_d6	1130	0	CFG_VIN2A_D6_IN	pr1_pru1_gpi3
A10	vin2a_d7	1202	0	CFG_VIN2A_D7_IN	pr1_pru1_gpi4
B10	vin2a_d8	1395	0	CFG_VIN2A_D8_IN	pr1_pru1_gpi5
E10	vin2a_d9	1338	0	CFG_VIN2A_D9_IN	pr1_pru1_gpi6

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS2 PRU0 IOSET2 Direct Input mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-193 *Manual Functions Mapping for PRU-ICSS2 PRU0 IOSET2 Direct Input mode* for a definition of the Manual modes.

Table 5-193 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-193. Manual Functions Mapping for PRU-ICSS2 PRU0 IOSET2 Direct Input mode

BALL	BALL NAME	PR2_PRU0_DIR_IN_MANUAL2		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		12
Y5	gpio6_10	1000	3300	CFG_GPIO6_10_IN	pr2_pru0_gpi0
Y6	gpio6_11	1000	3400	CFG_GPIO6_11_IN	pr2_pru0_gpi1
F16	mcasp1_axr15	0	1300	CFG_MCASP1_AXR15_IN	pr2_pru0_gpi20
E19	mcasp2_aclkx	0	800	CFG_MCASP2_ACLKX_IN	pr2_pru0_gpi18
A21	mcasp2_axr2	0	1900	CFG_MCASP2_AXR2_IN	pr2_pru0_gpi16
B21	mcasp2_axr3	0	1400	CFG_MCASP2_AXR3_IN	pr2_pru0_gpi17
D19	mcasp2_fsx	0	1400	CFG_MCASP2_FSX_IN	pr2_pru0_gpi19
B22	mcasp3_axr0	0	1400	CFG_MCASP3_AXR0_IN	pr2_pru0_gpi14
B23	mcasp3_axr1	0	1000	CFG_MCASP3_AXR1_IN	pr2_pru0_gpi15
A23	mcasp3_fsx	0	1300	CFG_MCASP3_FSX_IN	pr2_pru0_gpi13
Y2	mmc3_clk	1000	3700	CFG_MMC3_CLK_IN	pr2_pru0_gpi2
Y1	mmc3_cmd	1000	3500	CFG_MMC3_CMD_IN	pr2_pru0_gpi3
Y4	mmc3_dat0	1000	3500	CFG_MMC3_DAT0_IN	pr2_pru0_gpi4
AA2	mmc3_dat1	1000	4000	CFG_MMC3_DAT1_IN	pr2_pru0_gpi5
AA3	mmc3_dat2	1000	3300	CFG_MMC3_DAT2_IN	pr2_pru0_gpi6
W2	mmc3_dat3	1000	3900	CFG_MMC3_DAT3_IN	pr2_pru0_gpi7
Y3	mmc3_dat4	1000	3500	CFG_MMC3_DAT4_IN	pr2_pru0_gpi8
AA1	mmc3_dat5	1000	3600	CFG_MMC3_DAT5_IN	pr2_pru0_gpi9
AA4	mmc3_dat6	1000	3500	CFG_MMC3_DAT6_IN	pr2_pru0_gpi10
AB1	mmc3_dat7	1000	3100	CFG_MMC3_DAT7_IN	pr2_pru0_gpi11
A22	mcasp3_aclkx	0	0	CFG_MCASP3_ACLKX_IN	pr2_pru0_gpi12

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS2 PRU0 IOSET2 Direct Output mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-194 *Manual Functions Mapping for PRU-ICSS2 PRU0 IOSET2 Direct Output mode* for a definition of the Manual modes.

Table 5-194 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-194. Manual Functions Mapping for PRU-ICSS2 PRU0 IOSET2 Direct Output mode

BALL	BALL NAME	PR2_PRU0_DIR_OUT_MANUAL2		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		13
Y5	gpio6_10	1800	1900	CFG_GPIO6_10_OUT	pr2_pru0_gpo0
Y6	gpio6_11	2500	2100	CFG_GPIO6_11_OUT	pr2_pru0_gpo1
F16	mcasp1_axr15	0	400	CFG_MCASP1_AXR15_OUT	pr2_pru0_gpo20
E19	mcasp2_aclkx	0	400	CFG_MCASP2_ACLKX_OUT	pr2_pru0_gpo18
A21	mcasp2_axr2	0	500	CFG_MCASP2_AXR2_OUT	pr2_pru0_gpo16
B21	mcasp2_axr3	0	500	CFG_MCASP2_AXR3_OUT	pr2_pru0_gpo17
D19	mcasp2_fsx	0	0	CFG_MCASP2_FSX_OUT	pr2_pru0_gpo19
A22	mcasp3_aclkx	0	500	CFG_MCASP3_ACLKX_OUT	pr2_pru0_gpo12
B22	mcasp3_axr0	0	0	CFG_MCASP3_AXR0_OUT	pr2_pru0_gpo14
B23	mcasp3_axr1	0	200	CFG_MCASP3_AXR1_OUT	pr2_pru0_gpo15
A23	mcasp3_fsx	0	300	CFG_MCASP3_FSX_OUT	pr2_pru0_gpo13
Y2	mmc3_clk	2100	2200	CFG_MMC3_CLK_OUT	pr2_pru0_gpo2
Y1	mmc3_cmd	2300	2300	CFG_MMC3_CMD_OUT	pr2_pru0_gpo3
Y4	mmc3_dat0	2000	1600	CFG_MMC3_DAT0_OUT	pr2_pru0_gpo4

Table 5-194. Manual Functions Mapping for PRU-ICSS2 PRU0 IOSET2 Direct Output mode (continued)

BALL	BALL NAME	PR2_PRU0_DIR_OUT_MANUAL2		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		13
AA2	mmc3_dat1	2000	1700	CFG_MMC3_DAT1_OUT	pr2_pru0_gpo5
AA3	mmc3_dat2	2050	2200	CFG_MMC3_DAT2_OUT	pr2_pru0_gpo6
W2	mmc3_dat3	2000	2000	CFG_MMC3_DAT3_OUT	pr2_pru0_gpo7
Y3	mmc3_dat4	2150	2600	CFG_MMC3_DAT4_OUT	pr2_pru0_gpo8
AA1	mmc3_dat5	2400	2600	CFG_MMC3_DAT5_OUT	pr2_pru0_gpo9
AA4	mmc3_dat6	2200	2300	CFG_MMC3_DAT6_OUT	pr2_pru0_gpo10
AB1	mmc3_dat7	1800	2400	CFG_MMC3_DAT7_OUT	pr2_pru0_gpo11

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS2 PRU1 IOSET1 Direct Input mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-195 *Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET1 Direct Input mode* for a definition of the Manual modes.

Table 5-195 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-195. Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET1 Direct Input mode

BALL	BALL NAME	PR2_PRU1_DIR_IN_MANUAL1		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		12
P5	RMII_MHZ_50_CLK	1400	1200	CFG_RMII_MHZ_50_CLK_IN	pr2_pru1_gpi2
L6	mdio_d	1300	1600	CFG_MDIO_D_IN	pr2_pru1_gpi1
L5	mdio_mclk	1400	800	CFG_MDIO_MCLK_IN	pr2_pru1_gpi0
N2	rgmii0_rxc	1400	500	CFG_RGMII0_RXC_IN	pr2_pru1_gpi11
P2	rgmii0_rxctl	1400	1800	CFG_RGMII0_RXCTL_IN	pr2_pru1_gpi12
N4	rgmii0_rxd0	1400	1300	CFG_RGMII0_RXD0_IN	pr2_pru1_gpi16
N3	rgmii0_rxd1	1400	1650	CFG_RGMII0_RXD1_IN	pr2_pru1_gpi15
P1	rgmii0_rxd2	1400	1400	CFG_RGMII0_RXD2_IN	pr2_pru1_gpi14
N1	rgmii0_rxd3	1400	1650	CFG_RGMII0_RXD3_IN	pr2_pru1_gpi13
T4	rgmii0_txc	1400	900	CFG_RGMII0_TXC_IN	pr2_pru1_gpi5
T5	rgmii0_txctl	1400	1300	CFG_RGMII0_TXCTL_IN	pr2_pru1_gpi6
R1	rgmii0_txd0	1400	900	CFG_RGMII0_TXD0_IN	pr2_pru1_gpi10
R2	rgmii0_txd1	1300	1400	CFG_RGMII0_TXD1_IN	pr2_pru1_gpi9
P3	rgmii0_txd2	1300	1100	CFG_RGMII0_TXD2_IN	pr2_pru1_gpi8
P4	rgmii0_txd3	1300	1300	CFG_RGMII0_TXD3_IN	pr2_pru1_gpi7
N5	uart3_rxd	1300	1000	CFG_UART3_RXD_IN	pr2_pru1_gpi3
N6	uart3_txd	1300	800	CFG_UART3_TXD_IN	pr2_pru1_gpi4

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS2 PRU1 IOSET2 Direct Input mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-196 *Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET2 Direct Input mode* for a definition of the Manual modes.

Table 5-196 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-196. Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET2 Direct Input mode

BALL	BALL NAME	PR2_PRU1_DIR_IN_MANUAL2		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		12
C16	mcasp1_aclkx	400	0	CFG_MCASP1_ACLKX_IN	pr2_pru1_gpi7
D14	mcasp1_axr0	700	200	CFG_MCASP1_AXR0_IN	pr2_pru1_gpi8
B14	mcasp1_axr1	600	300	CFG_MCASP1_AXR1_IN	pr2_pru1_gpi9
B16	mcasp1_axr10	600	500	CFG_MCASP1_AXR10_IN	pr2_pru1_gpi12
B18	mcasp1_axr11	700	500	CFG_MCASP1_AXR11_IN	pr2_pru1_gpi13
A19	mcasp1_axr12	500	0	CFG_MCASP1_AXR12_IN	pr2_pru1_gpi14
E17	mcasp1_axr13	600	200	CFG_MCASP1_AXR13_IN	pr2_pru1_gpi15
E16	mcasp1_axr14	600	0	CFG_MCASP1_AXR14_IN	pr2_pru1_gpi16
A18	mcasp1_axr8	800	0	CFG_MCASP1_AXR8_IN	pr2_pru1_gpi10
B17	mcasp1_axr9	600	300	CFG_MCASP1_AXR9_IN	pr2_pru1_gpi11
D23	mcasp4_axr1	500	0	CFG_MCASP4_AXR1_IN	pr2_pru1_gpi0
AC3	mcasp5_aclkx	2100	1959	CFG_MCASP5_ACLKX_IN	pr2_pru1_gpi1
AA5	mcasp5_axr0	2300	2000	CFG_MCASP5_AXR0_IN	pr2_pru1_gpi3
AC4	mcasp5_axr1	2300	1800	CFG_MCASP5_AXR1_IN	pr2_pru1_gpi4
U6	mcasp5_fsx	2100	1780	CFG_MCASP5_FSX_IN	pr2_pru1_gpi2
J25	xref_clk0	0	0	CFG_XREF_CLK0_IN	pr2_pru1_gpi5
J24	xref_clk1	0	0	CFG_XREF_CLK1_IN	pr2_pru1_gpi6

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS2 PRU1 IOSET1 Direct Output mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-197 *Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET1 Direct Output mode* for a definition of the Manual modes.

Table 5-197 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-197. Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET1 Direct Output mode

BALL	BALL NAME	PR2_PRU1_DIR_OUT_MANUAL1		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		13
P5	RMII_MHZ_50_CLK	2306	100	CFG_RMII_MHZ_50_CLK_OUT	pr2_pru1_gpo2
L6	mdio_d	1900	2000	CFG_MDIO_D_OUT	pr2_pru1_gpo1
L5	mdio_mclk	2000	1100	CFG_MDIO_MCLK_OUT	pr2_pru1_gpo0
N2	rgmii0_rxc	2000	1200	CFG_RGMII0_RXC_OUT	pr2_pru1_gpo11
P2	rgmii0_rxctl	2000	1700	CFG_RGMII0_RXCTL_OUT	pr2_pru1_gpo12
N4	rgmii0_rxd0	2000	1000	CFG_RGMII0_RXD0_OUT	pr2_pru1_gpo16
N3	rgmii0_rxd1	2200	1000	CFG_RGMII0_RXD1_OUT	pr2_pru1_gpo15
P1	rgmii0_rxd2	2200	1300	CFG_RGMII0_RXD2_OUT	pr2_pru1_gpo14
N1	rgmii0_rxd3	2250	1100	CFG_RGMII0_RXD3_OUT	pr2_pru1_gpo13
T4	rgmii0_txc	2350	1000	CFG_RGMII0_TXC_OUT	pr2_pru1_gpo5
T5	rgmii0_txctl	2000	1200	CFG_RGMII0_TXCTL_OUT	pr2_pru1_gpo6
R1	rgmii0_txd0	2000	1500	CFG_RGMII0_TXD0_OUT	pr2_pru1_gpo10
R2	rgmii0_txd1	1850	1000	CFG_RGMII0_TXD1_OUT	pr2_pru1_gpo9
P3	rgmii0_txd2	2100	1100	CFG_RGMII0_TXD2_OUT	pr2_pru1_gpo8
P4	rgmii0_txd3	2200	1000	CFG_RGMII0_TXD3_OUT	pr2_pru1_gpo7
N5	uart3_rxd	2000	1600	CFG_UART3_RXD_OUT	pr2_pru1_gpo3
N6	uart3_txd	2000	1000	CFG_UART3_TXD_OUT	pr2_pru1_gpo4

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS2 PRU1 IOSET2 Direct Output mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-198 *Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET2 Direct Output mode* for a definition of the Manual modes.

Table 5-198 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-198. Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET2 Direct Output mode

BALL	BALL NAME	PR2_PRU1_DIR_OUT_MANUAL2		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		13
C16	mcasp1_aclkx	200	800	CFG_MCASP1_ACLKX_OUT	pr2_pru1_gpo7
D14	mcasp1_axr0	200	1000	CFG_MCASP1_AXR0_OUT	pr2_pru1_gpo8
B14	mcasp1_axr1	0	1110	CFG_MCASP1_AXR1_OUT	pr2_pru1_gpo9
B16	mcasp1_axr10	0	2500	CFG_MCASP1_AXR10_OUT	pr2_pru1_gpo12
B18	mcasp1_axr11	0	1900	CFG_MCASP1_AXR11_OUT	pr2_pru1_gpo13
A19	mcasp1_axr12	0	2300	CFG_MCASP1_AXR12_OUT	pr2_pru1_gpo14
E17	mcasp1_axr13	200	1200	CFG_MCASP1_AXR13_OUT	pr2_pru1_gpo15
E16	mcasp1_axr14	200	1100	CFG_MCASP1_AXR14_OUT	pr2_pru1_gpo16
A18	mcasp1_axr8	200	1600	CFG_MCASP1_AXR8_OUT	pr2_pru1_gpo10
B17	mcasp1_axr9	0	1900	CFG_MCASP1_AXR9_OUT	pr2_pru1_gpo11
D23	mcasp4_axr1	0	700	CFG_MCASP4_AXR1_OUT	pr2_pru1_gpo0
AC3	mcasp5_aclkx	1400	4000	CFG_MCASP5_ACLKX_OUT	pr2_pru1_gpo1
AA5	mcasp5_axr0	1500	3000	CFG_MCASP5_AXR0_OUT	pr2_pru1_gpo3
AC4	mcasp5_axr1	1500	1900	CFG_MCASP5_AXR1_OUT	pr2_pru1_gpo4
U6	mcasp5_fsx	1300	2700	CFG_MCASP5_FSX_OUT	pr2_pru1_gpo2
J25	xref_clk0	0	160	CFG_XREF_CLK0_OUT	pr2_pru1_gpo5
J24	xref_clk1	0	0	CFG_XREF_CLK1_OUT	pr2_pru1_gpo6

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS2 PRU0 IOSET2 Parallel Capture Mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-199 *Manual Functions Mapping for PRU-ICSS2 PRU0 IOSET2 Parallel Capture Mode* for a definition of the Manual modes.

Table 5-199 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-199. Manual Functions Mapping for PRU-ICSS2 PRU0 IOSET2 Parallel Capture Mode

BALL	BALL NAME	PR2_PRU0_PAR_CAP_MANUAL2		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		12
Y5	gpio6_10	4125	481	CFG_GPIO6_10_IN	pr2_pru0_gpi0
Y6	gpio6_11	3935	997	CFG_GPIO6_11_IN	pr2_pru0_gpi1
A21	mcasp2_axr2	0	0	CFG_MCASP2_AXR2_IN	pr2_pru0_gpi16
A22	mcasp3_aclkx	571	0	CFG_MCASP3_ACLKX_IN	pr2_pru0_gpi12
B22	mcasp3_axr0	1570	0	CFG_MCASP3_AXR0_IN	pr2_pru0_gpi14
B23	mcasp3_axr1	1405	0	CFG_MCASP3_AXR1_IN	pr2_pru0_gpi15
A23	mcasp3_fsx	1946	0	CFG_MCASP3_FSX_IN	pr2_pru0_gpi13
Y2	mmc3_clk	4093	1066	CFG_MMC3_CLK_IN	pr2_pru0_gpi2
Y1	mmc3_cmd	4043	921	CFG_MMC3_CMD_IN	pr2_pru0_gpi3
Y4	mmc3_dat0	4010	864	CFG_MMC3_DAT0_IN	pr2_pru0_gpi4
AA2	mmc3_dat1	3817	1643	CFG_MMC3_DAT1_IN	pr2_pru0_gpi5

Table 5-199. Manual Functions Mapping for PRU-ICSS2 PRU0 IOSET2 Parallel Capture Mode (continued)

BALL	BALL NAME	PR2_PRU0_PAR_CAP_MANUAL2		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		12
AA3	mmc3_dat2	4040	673	CFG_MMC3_DAT2_IN	pr2_pru0_gpi6
W2	mmc3_dat3	3923	1478	CFG_MMC3_DAT3_IN	pr2_pru0_gpi7
Y3	mmc3_dat4	4096	729	CFG_MMC3_DAT4_IN	pr2_pru0_gpi8
AA1	mmc3_dat5	3926	1271	CFG_MMC3_DAT5_IN	pr2_pru0_gpi9
AA4	mmc3_dat6	4004	929	CFG_MMC3_DAT6_IN	pr2_pru0_gpi10
AB1	mmc3_dat7	3963	666	CFG_MMC3_DAT7_IN	pr2_pru0_gpi11

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS2 PRU1 IOSET1 Parallel Capture Mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-200 *Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET1 Parallel Capture Mode* for a definition of the Manual modes.

Table 5-200 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-200. Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET1 Parallel Capture Mode

BALL	BALL NAME	PR2_PRU1_PAR_CAP_MANUAL1		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		12
P5	RMII_MHZ_50_CLK	1717	0	CFG_RMII_MHZ_50_CLK_IN	pr2_pru1_gpi2
L5	mdio_d	2088	0	CFG_MDIO_D_IN	pr2_pru1_gpi1
L6	mdio_mclk	1321	0	CFG_MDIO_MCLK_IN	pr2_pru1_gpi0
N2	rgmii0_rxc	1287	0	CFG_RGMII0_RXC_IN	pr2_pru1_gpi11
P2	rgmii0_rxctl	2456	0	CFG_RGMII0_RXCTL_IN	pr2_pru1_gpi12
N4	rgmii0_rxd0	0	0	CFG_RGMII0_RXD0_IN	pr2_pru1_gpi16
N3	rgmii0_rxd1	2157	0	CFG_RGMII0_RXD1_IN	pr2_pru1_gpi15
P1	rgmii0_rxd2	2008	0	CFG_RGMII0_RXD2_IN	pr2_pru1_gpi14
N1	rgmii0_rxd3	2271	0	CFG_RGMII0_RXD3_IN	pr2_pru1_gpi13
T4	rgmii0_txc	1851	0	CFG_RGMII0_TXC_IN	pr2_pru1_gpi5
T5	rgmii0_txctl	1875	0	CFG_RGMII0_TXCTL_IN	pr2_pru1_gpi6
R1	rgmii0_txd0	1685	0	CFG_RGMII0_TXD0_IN	pr2_pru1_gpi10
R2	rgmii0_txd1	2131	0	CFG_RGMII0_TXD1_IN	pr2_pru1_gpi9
P3	rgmii0_txd2	1734	0	CFG_RGMII0_TXD2_IN	pr2_pru1_gpi8
P4	rgmii0_txd3	1764	0	CFG_RGMII0_TXD3_IN	pr2_pru1_gpi7
N5	uart3_rxd	1654	0	CFG_UART3_RXD_IN	pr2_pru1_gpi3
N6	uart3_txd	1242	0	CFG_UART3_TXD_IN	pr2_pru1_gpi4

Manual IO Timings Modes must be used to ensure some IO timings for PRU-ICSS2 PRU1 IOSET2 Parallel Capture Mode. See Table 5-29 *Modes Summary* for a list of IO timings requiring the use of Manual IO Timings Modes. See Table 5-201 *Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET2 Parallel Capture Mode* for a definition of the Manual modes.

Table 5-201 lists the A_DELAY and G_DELAY values needed to calculate the correct values to be set in the CFG_x registers.

Table 5-201. Manual Functions Mapping for PRU-ICSS2 PRU1 IOSET2 Parallel Capture Mode

BALL	BALL NAME	PR2_PRU1_PAR_CAP_MANUAL2		CFG REGISTER	MUXMODE
		A_DELAY (ps)	G_DELAY (ps)		12
C16	mcasp1_aclkx	1928	0	CFG_MCASP1_ACLKX_IN	pr2_pru1_gpi7
D14	mcasp1_axr0	2413	0	CFG_MCASP1_AXR0_IN	pr2_pru1_gpi8
B14	mcasp1_axr1	2523	25	CFG_MCASP1_AXR1_IN	pr2_pru1_gpi9
B16	mcasp1_axr10	2607	0	CFG_MCASP1_AXR10_IN	pr2_pru1_gpi12
B18	mcasp1_axr11	2669	92	CFG_MCASP1_AXR11_IN	pr2_pru1_gpi13
A19	mcasp1_axr12	2225	0	CFG_MCASP1_AXR12_IN	pr2_pru1_gpi14
E17	mcasp1_axr13	2315	0	CFG_MCASP1_AXR13_IN	pr2_pru1_gpi15
E16	mcasp1_axr14	0	0	CFG_MCASP1_AXR14_IN	pr2_pru1_gpi16
A18	mcasp1_axr8	2201	0	CFG_MCASP1_AXR8_IN	pr2_pru1_gpi10
B17	mcasp1_axr9	2293	278	CFG_MCASP1_AXR9_IN	pr2_pru1_gpi11
D23	mcasp4_axr1	1759	0	CFG_MCASP4_AXR1_IN	pr2_pru1_gpi0
AC3	mcasp5_aclkx	3732	1810	CFG_MCASP5_ACLKX_IN	pr2_pru1_gpi1
AA5	mcasp5_axr0	3776	2255	CFG_MCASP5_AXR0_IN	pr2_pru1_gpi3
AC4	mcasp5_axr1	3886	1923	CFG_MCASP5_AXR1_IN	pr2_pru1_gpi4
U6	mcasp5_fsx	3800	1449	CFG_MCASP5_FSX_IN	pr2_pru1_gpi2
J25	xref_clk0	1375	21	CFG_XREF_CLK0_IN	pr2_pru1_gpi5
J24	xref_clk1	1320	0	CFG_XREF_CLK1_IN	pr2_pru1_gpi6

5.10.6.24 System and Miscellaneous interfaces

The Device includes the following System and Miscellaneous interfaces:

- Sysboot Interface
- System DMA Interface
- Interrupt Controllers (INTC) Interface

5.10.7 Emulation and Debug Subsystem

The Device includes the following Test interfaces:

- IEEE 1149.1 Standard-Test-Access Port (JTAG)
- Trace Port Interface Unit (TPIU)

5.10.7.1 IEEE 1149.1 Standard-Test-Access Port (JTAG)

The JTAG (IEEE Standard 1149.1-1990 Standard-Test-Access Port and Boundary Scan Architecture) interface is used for BSDL testing and emulation of the device. The trstn pin only needs to be released when it is necessary to use a JTAG controller to debug the device or exercise the device's boundary scan functionality. For maximum reliability, the device includes an internal Pulldown (IPD) on the trstn pin to ensure that trstn is always asserted upon power up and the device's internal emulation logic is always properly initialized. JTAG controllers from Texas Instruments actively drive trstn high. However, some third-party JTAG controllers may not drive trstn high but expect the use of a Pullup resistor on trstn. When using this type of JTAG controller, assert trstn to initialize the device after powerup and externally drive trstn high before attempting any emulation or boundary-scan operations.

The main JTAG features include:

- 32KB embedded trace buffer (ETB)
- 5-pin system trace interface for debug
- Supports Advanced Event Triggering (AET)
- All processors can be emulated via JTAG ports
- All functions on EMU pins of the device:
  - EMU[1:0] - cross-triggering, boot mode (WIR), STM trace
  - EMU[4:2] - STM trace only (single direction)

5.10.7.1.1 JTAG Electrical Data/Timing

Table 5-202, Table 5-203 and Figure 5-130 assume testing over the recommended operating conditions and electrical characteristic conditions below.

Table 5-202. Timing Requirements for IEEE 1149.1 JTAG

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{c(TCK)}$	Cycle time, TCK	62.29		ns
1a	$t_{w(TCKH)}$	Pulse duration, TCK high (40% of t_c)	24.92		ns
1b	$t_{w(TCKL)}$	Pulse duration, TCK low (40% of t_c)	24.92		ns
3	$t_{su(TDI-TCK)}$	Input setup time, TDI valid to TCK high	6.23		ns
	$t_{su(TMS-TCK)}$	Input setup time, TMS valid to TCK high	6.23		ns
4	$t_{h(TCK-TDI)}$	Input hold time, TDI valid from TCK high	31.15		ns
	$t_{h(TCK-TMS)}$	Input hold time, TMS valid from TCK high	31.15		ns

Table 5-203. Switching Characteristics Over Recommended Operating Conditions for IEEE 1149.1 JTAG

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
2	$t_{d(TCKL-TDOV)}$	Delay time, TCK low to TDO valid	0	30.5	ns

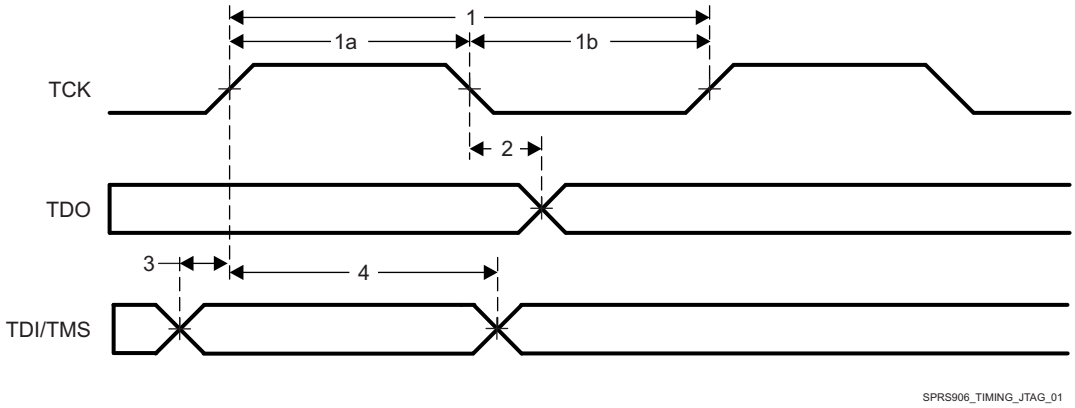


Figure 5-130. JTAG Timing

Table 5-204, Table 5-205 and Figure 5-131 assume testing over the recommended operating conditions and electrical characteristic conditions below.

Table 5-204. Timing Requirements for IEEE 1149.1 JTAG With RTCK

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
1	$t_{c(TCK)}$	Cycle time, TCK	62.29		ns
1a	$t_{w(TCKH)}$	Pulse duration, TCK high (40% of t_c)	24.92		ns
1b	$t_{w(TCKL)}$	Pulse duration, TCK low (40% of t_c)	24.92		ns
3	$t_{su(TDI-TCK)}$	Input setup time, TDI valid to TCK high	6.23		ns
	$t_{su(TMS-TCK)}$	Input setup time, TMS valid to TCK high	6.23		ns
4	$t_{h(TCK-TDI)}$	Input hold time, TDI valid from TCK high	31.15		ns
	$t_{h(TCK-TMS)}$	Input hold time, TMS valid from TCK high	31.15		ns

Table 5-205. Switching Characteristics Over Recommended Operating Conditions for IEEE 1149.1 JTAG With RTCK

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
5	$t_{d(TCK-RTCK)}$	Delay time, TCK to RTCK with no selected subpaths (i.e. ICEPick is the only tap selected - when the Arm is in the scan chain, the delay time is a function of the Arm functional clock).	0	27	ns
6	$t_{c(RTCK)}$	Cycle time, RTCK	62.29		ns
7	$t_{w(RTCKH)}$	Pulse duration, RTCK high (40% of t_c)	24.92		ns
8	$t_{w(RTCKL)}$	Pulse duration, RTCK low (40% of t_c)	24.92		ns

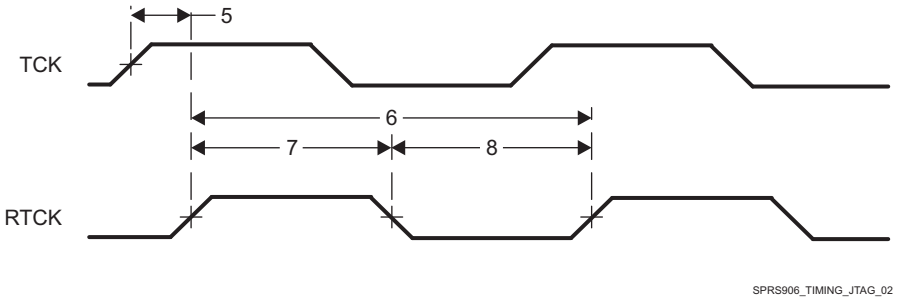


Figure 5-131. JTAG With RTCK Timing

5.10.7.2 Trace Port Interface Unit (TPIU)

CAUTION

The I/O timings provided in this section are valid only if signals within a single IOSET are used. The IOSETs are defined in Table 5-207.

5.10.7.2.1 TPIU PLL DDR Mode

Table 5-206 and Figure 5-132 assume testing over the recommended operating conditions and electrical characteristic conditions below.

Table 5-206. Switching Characteristics for TPIU

NO.	PARAMETER	DESCRIPTION	MIN	MAX	UNIT
TPIU1	$t_{c(clk)}$	Cycle time, TRACECLK period	5.56		ns
TPIU4	$t_{d(clk\text{-}ctlV)}$	Skew time, TRACECLK transition to TRACECTL transition	-1.61	1.98	ns
TPIU5	$t_{d(clk\text{-}dataV)}$	Skew time, TRACECLK transition to TRACEDATA[17:0]	-1.61	1.98	ns

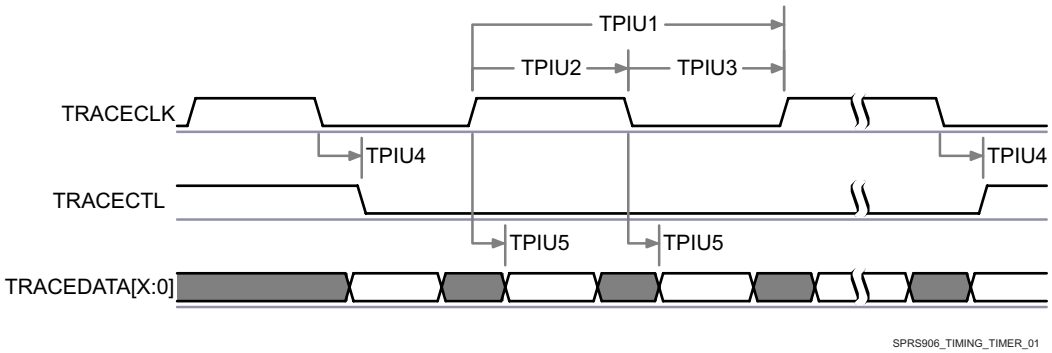


Figure 5-132. TPIU—PLL DDR Transmit Mode(1)

(1) In d[X:0], X is equal to 15 or 17.

In Table 5-207 are presented the specific groupings of signals (IOSET) for use with TPIU signals.

Table 5-207. TPIU IOSETs

SIGNALS	IOSET1		IOSET2	
	BALL	MUX	BALL	MUX
emu19	E10	5		
emu18	B10	5		
emu17	A10	5		
emu16	F10	5		
emu15	A11	5		
emu14	A8	5		
emu13	A9	5		
emu12	A7	5		
emu11	B9	5		
emu10	C8	5		
emu9	B8	5		
emu8	E8	5		
emu7	C7	5		
emu6	B7	5		

Table 5-207. TPIU IOSETs (continued)

SIGNALS	IOSET1		IOSET2	
	BALL	MUX	BALL	MUX
emu5	D8	5		
emu1	C22	0	C22	0
emu0	C21	0	C21	0

6 Detailed Description

6.1 Description

The DRA79x processor is offered in a 538-ball, 17×17-mm, 0.65-mm ball pitch (0.8mm spacing rules can be used on signals) with Via Channel™ Array (VCA) technology, ball grid array (BGA) package.

The architecture is designed to deliver high-performance concurrencies for automotive applications in a cost-effective solution, providing full scalability from the DRA75x ("Jacinto 6 EP" and "Jacinto 6 Ex"), DRA74x "Jacinto 6" and DRA72x "Jacinto 6 Eco" family of infotainment processors.

Programmability is provided by a single-core Arm Cortex-A15 RISC CPU with Neon extensions and a TI C66x VLIW floating-point DSP core. The Arm processor lets developers keep control functions separate from other algorithms programmed on the DSP and coprocessors, thus reducing the complexity of the system software.

Additionally, TI provides a complete set of development tools for the Arm, and DSP, including C compilers and a debugging interface for visibility into source code execution.

Cryptographic acceleration is available in all devices. All other supported security features, including support for secure boot, debug security and support for trusted execution environment are available on High-Security (HS) devices. For more information about HS devices, contact your TI representative.

The DRA79x Jacinto 6 Entry processor family is qualified according to the AEC-Q100 standard.

The device features are simplified power supply rail mapping which enables lower cost PMIC solutions.

6.2 Functional Block Diagram

Figure 6-1 is functional block diagram for the device.

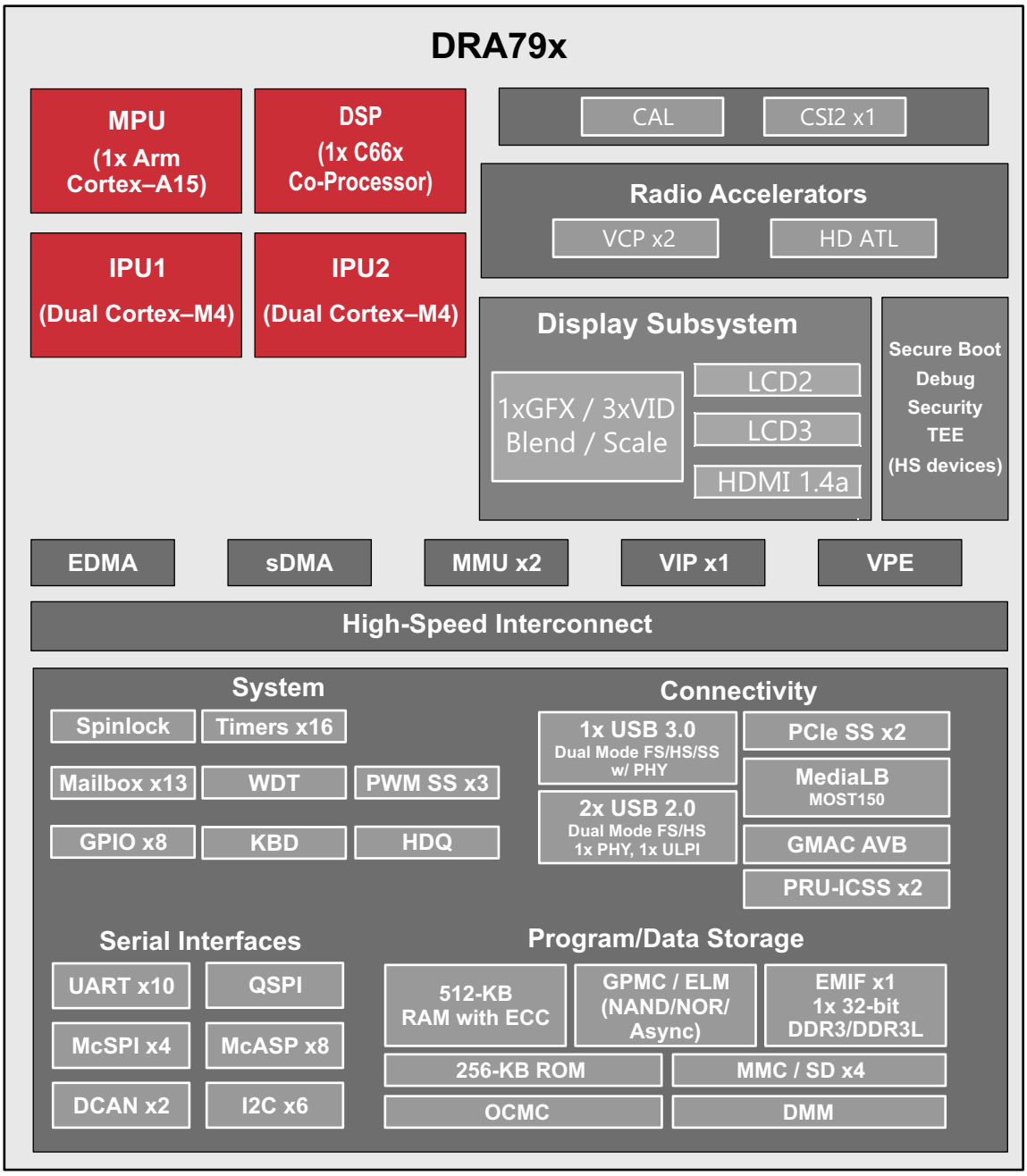


Figure 6-1. DRA79x Block Diagram

6.3 MPU

The Cortex®-A15 microprocessor unit (MPU) subsystem serves the applications processing role by running the high-level operating system (HLOS) and application code.

The MPU subsystem incorporates one Cortex-A15 MPU core (MPU_C0), individual level 1 (L1) caches, level 2 (L2) cache (MPU_L2CACHE) shared between them, and various other shared peripherals. To aid software development, the processor core can be kept cache-coherent with the L2 cache.

The MPU subsystem provides a high-performance computing platform with high peak-computing performance and low memory latency.

The Arm subsystem supports the following key features:

- Arm Cortex-A15 MP Core™ (MPU_CLUSTER)
  - One Cortex-A15 MPU core (revision r2p2) which has the following features:
    - Superscalar, dynamic multi-issue technology
      - Out-of-order (OoO) instruction dispatch and completion
      - Dynamic branch prediction with branch target buffer (BTB), global history buffer (GHB), and 48-entry return stack
      - Continuous fetch and decoding of three instructions per clock cycle
      - Dispatch of up to four instructions and completion of eight instructions per clock cycle
      - Provides optimal performance from binaries compiled for previous Arm processors
      - Five execution units handle simple instructions, branch instructions, Neon and floating point instructions, multiply instructions, and load and store instructions.
      - Simple instructions take two cycles from dispatch, while complex instructions take up to 11 cycles.
      - Can issue two simple instructions in a cycle
      - Can issue a load and a store instruction in the same cycle
    - Integrated Neon processing engine to include the Arm Neon Advanced SIMD (single instruction, multiple data) support for accelerated media and signal processing computation
    - Includes VFPv4-compatible hardware to support single- and double-precision add, subtract, divide, multiply and accumulate, and square root operations
    - Extensive support to accelerate virtualization using a hypervisor
    - 32-KiB L1 instruction (L1I) and 32-KiB L1 data (L1D) cache:
      - 64-byte line size
      - 2-way set associative
    - Memory management unit (MMU):
      - Two-level translation lookaside buffer (TLB) organization
        - First level is an 32-entry, fully associative micro-TLB implemented for each of instruction fetch, load, and store.
        - Second level is a unified, 4-way associative, 512-entry main TLB
      - Supports hardware TLB table-walk for backward-compatible and new 64-bit entry page table formats
      - New page table format can produce 40-bit physical addresses
      - Two-stage translation where first stage is HLOS-controlled and the second level may be controlled by a hypervisor. Second stage always uses the new page table format
  - Integrated L2 cache (MPU_L2CACHE) and snoop control unit (SCU):
    - 1-MiB of unified (instructions and data) cache organized as 16 ways of 1024 sets of 64-byte lines
    - Redundant L1 data (cache) tags to perform snoop filtering (L1 instruction cache tags are not duplicated)
    - Operates at Cortex-A15 MPU core clock rate
    - Integrated L2 cache controller (MPU_L2CACHE_CTRL):
      - Sixteen 64-byte line buffers that handle evictions, line fills and snoop transfers
      - One 128-bit AMBA4 Coherent Bus (AXI4-ACE) port
      - Auto-prefetch buffer for up to 16 streams and detecting forward and backward strides
  - Generalized interrupt controller (GIC, also referred to as MPU_INTC): An interrupt controller supplied by Arm. The single GIC in the MPU_CLUSTER routes interrupts to the MPU core. The GIC supports:
    - Number of shared peripheral interrupts (SPI): 160
    - Number of software generated interrupts (SGI): 16
    - Number of CPU interfaces: 1

        - Virtual CPU interface for virtualization support. This allows the majority of guest operating system (OS) interactions with the GIC to be handled in hardware, but with physical interrupts still requiring hypervisor intervention to assign them to the appropriate virtual machine.
    - Integrated timer counter and one timer block
    - Arm CoreSight™ debug and trace modules. For more information, see chapter *On-Chip Debug Support* of the Device TRM..
- MPU_AXI2OCP bridge (local interconnect):
    - Connected to Memory Adapter (MPU_MA), which routes the non-EMIF address space transactions to MPU_AXI2OCP
    - Single request multiple data (SRMD) protocol on L3_MAIN port
    - Multiple targets:
        - 64-bit port to the L3_MAIN interconnect. Interface frequency is 1/4 or 1/8 of core frequency
        - MPU_ROM
        - Internal MPU subsystem peripheral targets, including Memory Adapter LISA Section Manager (MA_LSM), wake-up generator (MPU_WUGEN), watchdog timer (MPU_WD_TIMER), and local PRCM module (MPU_PRCM) configuration
        - Internal AXI target, CoreSight System Trace Module (CS_STM)
- Memory adapter (MPU_MA): Helps decrease the latency of accesses between the MPU_L2CACHE and the external memory interface (EMIF1) by providing a direct path between the MPU subsystem and EMIF1:
    - Connected to 128-bit AMBA4 interface of MPU_CLUSTER
    - Direct 128-bit interface to EMIF1
    - Interface speed between MPU_CLUSTER and MPU_MA is at half-speed of the MPU core frequency
    - Quarter-speed interface to EMIF
    - Uses firewall logic to check access rights of incoming addresses
- Local PRCM (MPU_PRCM):
    - Handles MPU_C0 power domain
    - Supports SR3-APG (SmartReflex3 Automatic Power Gating) power management technology inside the MPU_CLUSTER
    - MPU subsystem has five power domains
- Wake-up generator (MPU_WUGEN)
    - Responsible for waking up the MPU core
- Standby controller: Handles the power transitions inside the MPU subsystem
- Realtime (master) counter (COUNTER_REALTIME): Produces the count used by the private timer peripheral in the MPU_CLUSTER
- Watchdog timer (MPU_WD_TIMER): Used to generate a chip-level watchdog reset request to global PRCM
- On-chip boot ROM (MPU_ROM): The MPU_ROM size is 48-KiB, and the address range is from 0x4003 8000 to 0x4004 3FFF. For more information about booting from this memory, see chapter *Initialization* of the Device TRM..

- Interfaces:
  - 128-bit interface to EMIF1
  - 64-bit master port to the L3_MAIN interconnect
  - 32-bit slave port from the L4_CFG_EMU interconnect (debug subsystem) for configuration of the MPU subsystem debug modules
  - 32-bit slave port from the L4_CFG interconnect for memory adapter firewall (MPU_MA_NTTP_FW) configuration
  - 32-bit ATB output for transmitting debug and trace data
  - 160 peripheral interrupt inputs

For more information, see section *Arm Cortex-A15 Subsystem* in chapter *Processors and Accelerators* of the device TRM.

6.4 DSP Subsystem

The device includes a single instance (DSP1) of a digital signal processor (DSP) subsystem, based on the TI's standard TMS320C66x DSP CorePac core.

The TMS320C66x DSP core enhances the TMS320C674x core, which merges the C674x™ floating point and the C64x+™ fixed-point instruction set architectures. The C66x DSP is object-code compatible with the C64x+/C674x DSPs.

For more information on the TMS320C66x core CPU, see the *TMS320C66x DSP CPU and Instruction Set Reference Guide*, (SPRUGH7).

The DSP subsystem integrated in the device includes the following components:

- A TMS320C66x CorePac DSP core that encompasses:
  - L1 program-dedicated (L1P) cacheable memory
  - L1 data-dedicated (L1D) cacheable memory
  - L2 (program and data) cacheable memory
  - Extended Memory Controller (XMC)
  - External Memory Controller (EMC)
  - DSP CorePac located interrupt controller (INTC)
  - DSP CorePac located power-down controller (PDC)
- Dedicated enhanced data memory access engine - EDMA, to transfer data from/to memories and peripherals external to the DSP subsystem and to local DSP memory (most commonly L2 SRAM). The external DMA requests are passed through DSP system level (SYS) wakeup logic, and collected from the DSP1 dedicated outputs of the device DMA Events Crossbar for the subsystem.
- A level 2 (L2) interconnect network (DSP NoC) to allow connectivity between different modules of the subsystem or the remainder of the device via the device L3_MAIN interconnect.
- Two memory management units (on EDMA L2 interconnect and DSP MDMA paths) for accessing the device L3_MAIN interconnect address space
- Dedicated system control logic (DSP_SYSTEM) responsible for power management, clock generation, and connection to the device power, reset, and clock management (PRCM) module

The TMS320C66x Instruction Set Architecture (ISA) is the latest for the C6000 family. As with its predecessors (C64x, C64x+ and C674x), the C66x is an advanced VLIW architecture with 8 functional units (two multiplier units and six arithmetic logic units) that operate in parallel. The C66x CPU has a total of 64 general-purpose 32-bit registers.

Some features of the DSP C6000 family devices are:

- Advanced VLIW CPU with eight functional units (two multipliers and six ALUs) which:
  - Executes up to eight instructions per cycle for up to ten times the performance of typical DSPs
  - Allows designers to develop highly effective RISC-like code for fast development time

- Instruction packing
  - Gives code size equivalence for eight instructions executed serially or in parallel
  - Reduces code size, program fetches, and power consumption
- Conditional execution of most instructions
  - Reduces costly branching
  - Increases parallelism for higher sustained performance
- Efficient code execution on independent functional units
  - Industry's most efficient C compiler on DSP benchmark suite
  - Industry's first assembly optimizer for fast development and improved parallelization
- 8-/16-/32-bit/64-bit data support, providing efficient memory support for a variety of applications
- 40-bit arithmetic options which add extra precision for vocoders and other computationally intensive applications
- Saturation and normalization to provide support for key arithmetic operations
- Field manipulation and instruction extract, set, clear, and bit counting support common operation found in control and data manipulation applications.

The C66x CPU has the following additional features:

- Each multiplier can perform two 16 × 16-bit or four 8 × 8 bit multiplies every clock cycle.
- Quad 8-bit and dual 16-bit instruction set extensions with data flow support
- Support for non-aligned 32-bit (word) and 64-bit (double word) memory accesses
- Special communication-specific instructions have been added to address common operations in error-correcting codes.
- Bit count and rotate hardware extends support for bit-level algorithms.
- Compact instructions: Common instructions (AND, ADD, LD, MPY) have 16-bit versions to reduce code size.
- Protected mode operation: A two-level system of privileged program execution to support higher-capability operating systems and system features such as memory protection.
- Exceptions support for error detection and program redirection to provide robust code execution
- Hardware support for modulo loop operation to reduce code size and allow interrupts during fully-pipelined code
- Each multiplier can perform 32 × 32 bit multiplies
- Additional instructions to support complex multiplies allowing up to eight 16-bit multiply/add/subtracts per clock cycle

The TMS320C66x has the following key improvements to the ISA:

- 4x Multiply Accumulate improvement for both fixed and floating point
- Improvement of the floating point arithmetic
- Enhancement of the vector processing capability for fixed and floating point
- Addition of domain-specific instructions for complex arithmetic and matrix operations

On the C66x ISA, the vector processing capability is improved by extending the width of the SIMD instructions. The C674x DSP supports 2-way SIMD operations for 16-bit data and 4-way SIMD operations for 8-bit data. C66x enhances this capabilities with the addition of SIMD instructions for 32-bit data allowing operation on 128-bit vectors. For example the QMPY32 instruction is able to perform the element to element multiplication between two vectors of four 32-bit data each.

C66x ISA includes a set of specific instructions to handle complex arithmetic and matrix operations.

- **TMS320C66x DSP CorePac memory components:**
  - A 32-KiB L1 program memory (L1P) configurable as cache and/or SRAM:
    - When configured as a cache, the L1P is a 1-way set-associative cache with a 32-byte cache line
    - The DSP CorePac L1P memory controller provides bandwidth management, memory protection, and power-down functions
    - The L1P is capable of cache block and global coherence operations
    - The L1P controller has an Error Detection (ED) mechanism, including necessary SRAM
    - The L1P memory can be fully configured as a cache or SRAM
    - Page size for L1P memory is 2KB
  - A 32-KiB L1 data memory (L1D) configurable as cache and / or SRAM:
    - When configured as a cache, the L1D is a 2-way set-associative cache with a 64-byte cache line
    - The DSP CorePac L1D memory controller provides bandwidth management, memory protection, and power-down functions
    - The L1D memory can be fully configured as a cache or SRAM
    - No support for error correction or detection
    - Page size for L1D memory is 2KB
  - A 288-KiB (program and data) L2 memory, only part of which is cacheable:
    - When configured as a cache, the L2 memory is a 4-way set associative cache with a 128-byte cache line
    - Only 256 KiB of L2 memory can be configured as cache or SRAM
    - 32 KiB of the L2 memory is always mapped as SRAM
    - The L2 memory controller has an Error Correction Code (ECC) and ED mechanism, including necessary SRAM
    - The L2 memory controller supports hardware prefetching and also provides bandwidth management, memory protection, and power-down functions.
    - Page size for L2 memory is 16KB
- The **External Memory Controller (EMC)** is a bridge from the C66x CorePac to the rest of the DSP subsystem and device. It has :
  - a 32-bit configuration port (CFG) providing access to local subsystem resources (like DSP_EDMA, DSP_SYSTEM, and so forth) or to L3_MAIN resources accessible via the CFG address range.
  - a 128-bit slave-DMA port (SDMA) which provides accesses of system masters outside the DSP subsystem to resources inside the DSP subsystem or C66x DSP CorePac memories, i.e. when the DSP subsystem is the slave in a transaction.
- The **Extended Memory Controller (XMC)** processes requests from the L2 Cache Controller (which are a result of CPU instruction fetches, load/store commands, cache operations) to device resources via the C66x DSP CorePac 128-bit master DMA (MDMA) port:
  - Memory protection for addresses outside C66x DSP CorePac generated over device L3_MAIN on the MDMA port
  - Prefetch, multi-in-flight requests
- A DSP local **Interrupt Controller (INTC)** in the DSP C66x CorePac, interfaces the system events to the DSP C66x core CPU interrupt and exceptions inputs. The DSP subsystem C66x CorePac interrupt controller supports up to 128 system events of which 64 interrupts are external to DSP subsystem, collected from the DSP1 dedicated outputs of the device Interrupt Crossbar.

- **Local Enhanced Direct Memory Access (EDMA) controller features:**
  - Channel controller (CC) : 64-channel, 128 PaRAM, 2 Queues
  - 2 x Third-party Transfer Controllers (TPTC0 and TPTC1):
    - Each TC has a 128-bit read port and a 128-bit write port
    - 2KiB FIFOs on each TPTC
  - 1-dimensional/2-dimensional (1D/2D) addressing
  - Chaining capability
- **DSP subsystem integrated MMUs**:
  - Two MMUs are integrated:
    - The MMU0 is located between DSP MDMA master port and the device L3_MAIN interconnect and can be optionally bypassed
    - The MMU1 is located between the EDMA master port and the device L3_MAIN interconnect
- A DSP local **Power-Down Controller (PDC)** is responsible to power-down various parts of the DSP C66x CorePac, or the entire DSP C66x CorePac.
- The DSP subsystem **System Control logic** provides:
  - Slave idle and master standby protocols with device PRCM for powerdown
  - OCP Disconnect handshake for init and target busses
  - Asynchronous reset
  - Power-down modes:
    - "Clockstop" mode featuring wake-up on interrupt event. The DMA event wake-up is managed in software.
- The device DSP subsystem is supplied by a PRCM DPLL, but DSP1 **has integrated its own PLL module** outside the C66x CorePac for clock gating and division.
- **The device DSP subsystem has following port instances** to connect to remaining part of the device. See also :
  - A 128-bit initiator (DSP MDMA master) port for MDMA/Cache requests
  - A 128-bit initiator (DSP EDMA master) port for EDMA requests
  - A 32-bit initiator (DSP CFG master) port for configuration requests
  - A 128-bit target (DSP slave) port for requests to DSP memories and various peripherals
- **C66x DSP subsystem (DSPSS) safety aspects**:
  - Above mentioned memory ECC/ED mechanisms
  - MMUs enable mapping of only the necessary application space to the processor
  - Memory Protection Units internal to the DSPSS (in L1P, L1D and L2 memory controllers) and external to DSPSS (firewalls) to help define legal accesses and raise exceptions on illegal accesses
  - Exceptions: Memory errors, various DSP errors, MMU errors and some system errors are detected and cause exceptions. The exceptions could be handled by the DSP or by a designated safety processor at the chip level. Note that it may not be possible for the safety processor to completely handle some exceptions

Unsupported features on the C66x DSP core for the device are:

- The Extended Memory Controller MPAX (memory protection and address extension) 36-bit addressing is NOT supported

Known DSP subsystem powermode restrictions for the device are:

- "Full logic / RAM retention" mode featuring wake-up on both interrupt or DMA event (logic in “always on” domain). Only OFF mode is supported by DSP subsystem, **requiring full boot.**

Fore more information about:

- C66x debug/trace support, see chapter *On-Chip Debug* of the device TRM.

6.5 IPU

The device instantiates two dual Cortex-M4 image processor unit (IPU) subsystems

NOTE

The two IPU subsystems are identical from functional point of view. Thus, a unified name **IPUx** shall be used throughout the chapter for simplification.

Each IPU subsystem contains two Arm Cortex-M4 processors (IPUx_C0 and IPUx_C1) that share a common level 1 (L1) cache (called unicache [IPUx_UNICACHE]). The two Cortex-M4 cores are completely homogeneous to one another. Any task possible using one Cortex-M4 core is also possible using the other Cortex-M4 core. It is software responsibility to distribute the various tasks between each Cortex-M4 core for optimal performance.

The integrated interrupt handling of the IPUx subsystem allows it to function as an efficient control unit.

Each IPU subsystem integrates the following:

- Two Arm Cortex-M4 microprocessors (IPUx_C0 and IPUx_C1):
  - Armv7-M and Thumb®-2 instruction set architecture (ISA)
  - Armv6 SIMD and digital signal processor (DSP) extensions
  - Single-cycle MAC
  - Integrated nested vector interrupt controller (NVIC) (also called IPUx_Cx_INTC, where x = 0, 1)
  - Integrated bus matrix
  - Registers:
    - Thirteen general-purpose 32-bit registers
    - Link register (LR)
    - Program counter (PC)
    - Program status register, xPSR
    - Two banked SP registers
  - Integrated power management
  - Extensive debug capabilities
- Unicache interface:
  - Instruction and data interface
  - Supports paralleled accesses
- Level 2 (L2) master interface (MIF) splitter for access to memory or configuration port
- Configuration port: Used for unicache maintenance and unicache memory management unit (IPUx_UNICACHE_MMU) configuration
- Unicache:
  - 32 KiB divided into 16 banks
  - 4-way
  - Cache configuration lock/freeze/preload
  - Internal MMU:
    - 16-entry region-based address translation
    - Read/write control and access type control
    - Execute Never (XN) MMU protection policy
    - Little-endian format
- Subsystem counter timer module (IPUx_UNICACHE_SCTM, or just SCTM)
- On-chip ROM (IPUx_ROM) and banked RAM (IPUx_RAM) memory

- Emulation/debug: Emulation feature embedded in Cortex-M4
- L2 MMU (IPUx_MMU): 32 entries with table walking logic
- Wake-up generator (IPUx_WUGEN): Generates wake-up request from external interrupts
- Power management:
  - Local power-management control: Configurable through the IPUx_WUGEN registers.
  - Three sleep modes supported, controlled by the local power-management module.
  - IPUx is clock-gated in all sleep modes.
  - IPUx_Cx_INTC interrupt interface stays awake.

For more information, see chapter *Dual Cortex-M4 IPU Subsystem* of the device TRM.

6.6 PRU-ICSS

The device Programmable Real-Time Unit and Industrial Communication Subsystem (PRU-ICSS) consists of dual 32-bit Load / Store RISC CPU cores - Programmable Real-Time Units (PRU0 and PRU1), shared, data, and instruction memories, internal peripheral modules, and an interrupt controller (PRU-ICSS_INTC). The programmable nature of the PRUs, along with their access to pins, events and all SoC resources, provides flexibility in implementing fast real-time responses, specialized data handling operations, customer peripheral interfaces, and in off-loading tasks from the other processor cores of the system-on-chip (SoC).

The each PRU-ICSS includes the following main features:

- 21x Enhanced GPIs (EGPIs) and 21x Enhanced GPOs (EGPOs) with asynchronous capture and serial support per each PRU CPU core
- One Ethernet MII_RT module (PRU-ICSS_MII_RT) with two MII ports and configurable connections to PRUs
- 1 MDIO Port (PRU-ICSS_MII_MDIO)
- One Industrial Ethernet Peripheral (IEP) to manage/generate Industrial Ethernet functions
- 1 x 16550-compatible UART with a dedicated 192 MHz clock to support 12Mbps Profibus
- 1 Industrial Ethernet timer with 7/9 capture and 8 compare events
- 1 Enhanced Capture Module (ECAP)
- 1 Interrupt Controller (PRU-ICSS_INTC)
- A flexible power management support
- Integrated switched central resource with programmable priority
- Parity control supported by all memories

For more information, see chapter *Programmable Real-Time Unit Subsystem and Industrial Communication Subsystem (PRU-ICSS)* of the device TRM.

6.7 Memory Subsystem

6.7.1 EMIF

The EMIF module provides connectivity between DDR memory types and manages data bus read/write accesses between external memory and device subsystems which have master access to the L3_MAIN interconnect and DMA capability.

The EMIF module has the following capabilities:

- Supports JEDEC standard-compliant DDR3/DDR3L-SDRAM memory types
- 2-GiB SDRAM address range over one chip-select. This range is configurable through the dynamic memory manager (DMM) module
- Supports SDRAM devices with one, two, four or eight internal banks
- Supports SDRAM devices with single or dual die packages
- Data bus widths:
  - 128-bit L3_MAIN (system) interconnect data bus width
  - 128-bit port for direct connection with MPU subsystem
  - 32-bit SDRAM data bus width
  - 16-bit SDRAM data bus width used in narrow mode
- Supported CAS latencies:
  - DDR3: 5, 6, 7, 8, 9, 10 and 11
- Supports 256-, 512-, 1024-, and 2048-word page sizes
- Supported burst length: 8
- Supports sequential burst type
- SDRAM auto initialization from reset or configuration change
- Supports self refresh and power-down modes for low power
- Partial array self-refresh mode for low power.
- Output impedance (ZQ) calibration for DDR3
- Supports on-die termination (ODT) DDR3
- Supports prioritized refresh
- Programmable SDRAM refresh rate and backlog counter
- Programmable SDRAM timing parameters
- Write and read leveling/calibration and data eye training for DDR3.

The EMIF module does not support:

- Burst chop for DDR3
- Interleave burst type
- Auto precharge because of better Bank Interleaving performance
- DLL disabling from EMIF side
- SDRAM devices with more than one die, or topologies which require more than one chip select on a single EMIF channel

For more information, see section *DDR External Memory Interface (EMIF)* in chapter *Memory Subsystem* of the device TRM.

6.7.2 GPMC

The General Purpose Memory Controller (GPMC) is an external memory controller of the device. Its data access engine provides a flexible programming model for communication with all standard memories.

The GPMC supports the following various access types:

- Asynchronous read/write access

- Asynchronous read page access (4, 8, and 16 Word16)
- Synchronous read/write access
- Synchronous read/write burst access without wrap capability (4, 8 and 16 Word16)
- Synchronous read/write burst access with wrap capability (4, 8 and 16 Word16)
- Address-data-multiplexed (AD) access
- Address-address-data (AAD) multiplexed access
- Little- and big-endian access

The GPMC can communicate with a wide range of external devices:

- External asynchronous or synchronous 8-bit wide memory or device (non burst device)
- External asynchronous or synchronous 16-bit wide memory or device
- External 16-bit non-multiplexed NOR flash device
- External 16-bit address and data multiplexed NOR Flash device
- External 8-bit and 16-bit NAND flash device
- External 16-bit pseudo-SRAM (pSRAM) device

The main features of the GPMC are:

- 8- or 16-bit-wide data path to external memory device
- Supports up to eight CS regions of programmable size and programmable base addresses in a total address space of 1 GiB
- Supports transactions controlled by a firewall
- On-the-fly error code detection using the Bose-Chaudhurl-Hocquenghem (BCH) (t = 4, 8, or 16) or Hamming code to improve the reliability of NAND with a minimum effect on software (NAND flash with 512-byte page size or greater)
- Fully pipelined operation for optimal memory bandwidth use
- The clock to the external memory is provided from GPMC functional clock divided by 1, 2, 3, or 4
- Supports programmable autoclock gating when no access is detected
- Independent and programmable control signal timing parameters for setup and hold time on a per-chip basis. Parameters are set according to the memory device timing parameters, with a timing granularity of one GPMC functional clock cycle.
- Flexible internal access time control (WAIT state) and flexible handshake mode using external WAIT pin monitoring
- Support bus keeping
- Support bus turnaround
- Prefetch and write posting engine associated with to achieve full performance from the NAND device with minimum effect on NOR/SRAM concurrent access

For more information, see section *General-Purpose Memory Controller (GPMC)* in chapter *Memory Subsystem* of the device TRM.

6.7.3 ELM

In the case of NAND modules with no internal correction capability, sometimes referred to as bare NAND, the correction process can be delegated to the error location module (ELM) used in conjunction with the GPMC.

The ELM supports the following features:

- 4, 8, and 16 bits per 512-byte block error location based on BCH algorithm
- Eight simultaneous processing contexts
- Page-based and continuous modes

- Interrupt generation when error location process completes:
  - When the full page has been processed in page mode
  - For each syndrome polynomial (checksum-like information) in continuous mode

For more information, see section *Error Location Module (ELM)* in chapter *Memory Subsystem* of the device TRM.

6.7.4 OCMC

There is one on-chip memory controller (OCMC) in the device.

The OCM Controller supports the following features:

- L3_MAIN data interface:
  - Used for maximum throughput performance
  - 128-bit data bus width
  - Burst supported
- L4 interface (OCMC_RAM only):
  - Used for access to configuration registers
  - 32-bit data bus width
  - Only single accesses supported
  - The L4 associated OCMC clock is two times lower than the L3 associated OCMC clock
- Error correction and detection:
  - Single error correction and dual error detection
  - 9-bit Hamming error correction code (ECC) calculated on 128-bit data word which is concatenated with memory address bits
  - Hamming distance of 4
  - Enable/Disable mode control through a dedicated register
  - Single bit error correction on a read transaction
  - Exclusion of repeated addresses from correctable error address trace history
  - ECC valid for all write transactions to an enabled region
  - Sub-128-bit writes supported via read modify write
- ECC Error Status Reporting:
  - Trace history buffer (FIFO) with depth of 4 for corrected error address
  - Trace history buffer with depth of 4 for non correctable error address and also including double error detection
  - Interrupt generation for correctable and uncorrectable detected errors
- ECC Diagnostics Configuration:
  - Counters for single error correction (SEC), double error detection (DED) and address error events (AEE)
  - Programmable threshold registers for exeptions associated with SEC, DED and AEE counters
  - Register control for enabling and disabling of diagnostics
  - Configuration registers and ECC status accessible through L4 interconnect
- Circular buffer for sliced based VIP frame transfers:
  - Up to 12 programmable circular buffers mapped with unique virtual frame addresses
  - On the fly (with no additional latency) address translation from virtual to OCMC circular buffer memory space
  - Virtual frame size up to 8 MiB and circular buffer size up to 1 MiB
  - Error handling and reporting of illegal CBUF addressing
  - Underflow and Overflow status reporting and error handling
  - Last access read/write address history

- Two Interrupt outputs configured independently to service either ECC or CBUF interrupt events

The OCM controller does not have a memory protection logic and does not support endianism conversion.

For more information, see section *On-Chip Memory (OCM)* in chapter *Memory Subsystem* of the device TRM.

6.8 Interprocessor Communication

6.8.1 MailBox

Communication between the on-chip processors of the device uses a queued mailbox-interrupt mechanism.

The queued mailbox-interrupt mechanism allows the software to establish a communication channel between two processors through a set of registers and associated interrupt signals by sending and receiving messages (mailboxes).

The device implements the following mailbox types:

- System mailbox:
  - Number of instances: 13
  - Used for communication between: MPU, DSP1, IPU1, and IPU2 subsystems
  - Reference name: MAILBOX(1..13)

Each mailbox module supports the following features:

- Parameters configurable at design time
  - Number of users
  - Number of mailbox message queues
  - Number of messages (FIFO depth) for each message queue
- 32-bit message width
- Message reception and queue-not-full notification using interrupts
- Support of 16-/32-bit addressing scheme
- Power management support

For more information, see chapter *MailBox* of the device TRM.

6.8.2 Spinlock

The Spinlock module provides hardware assistance for synchronizing the processes running on multiple processors in the device:

- Cortex®-A15 microprocessor unit (MPU) subsystem
- Digital signal processor (DSP) subsystem – DSP1
- Dual Cortex-M4 image processing unit (IPU) subsystems – IPU1 and IPU2

  The Spinlock module implements 256 spinlocks (or hardware semaphores), which provide an efficient way to perform a lock operation of a device resource using a single read-access, avoiding the need of a readmodify- write bus transfer that the programmable cores are not capable of.

For more information, see chapter *Spinlock Module* of the device TRM.

6.9 Interrupt Controller

The device has a large number of interrupts to service the needs of its many peripherals and subsystems. The MPU, DSP, and IPU (x2) subsystems are capable of servicing these interrupts via their integrated interrupt controllers. In addition, each processor's interrupt controller is preceded by an Interrupt Controller Crossbar (IRQ_CROSSBAR) that provides flexibility in mapping the device interrupts to processor interrupt inputs. For more information about IRQ crossbar, see chapter *Control Module* of the Device TRM.

Cortex®-A15 MPU Subsystem Interrupt Controller (MPU_INTC)

The MPU_INTC module (also called Generalized Interrupt Controller [GIC]) is a single functional unit that is integrated in the Arm® Cortex-A15 multiprocessor core (MPCore) alongside Cortex-A15 processor. It provides:

- 160 hardware interrupt inputs
- Generation of interrupts by software
- Prioritization of interrupts
- Masking of any interrupts
- Distribution of the interrupts to the target Cortex-A15 processor(s)
- Tracking the status of interrupts

The Cortex-A15 processor supports three main groups of interrupt sources, with each interrupt source having a unique ID:

- *Software Generated Interrupts (SGIs):* SGIs are generated by writing to the Cortex-A15 Software Generated Interrupt Register (GICD_SGIR). A maximum of 16 SGIs (ID0–ID15) can be generated for the CPU interface. An SGI has edge-triggered properties. The software triggering of the interrupt is equivalent to the edge transition of the interrupt signal on a peripheral input.
- *Private Peripheral Interrupts (PPIs):* A PPI is an interrupt generated by a peripheral that is specific to the processor. Although interrupts ID16–ID31 are dedicated to PPIs in general, only seven PPIs are actually used for the CPU interface (ID25–ID31). Interrupts ID16–ID24 are reserved (not used).
- *Shared Peripheral Interrupts (SPIs):* SPIs are triggered by events generated on associated interrupt input lines. In this device, the GIC is configured to support 160 SPIs corresponding to its external IRQS[159:0] signals.

For detailed information about this module and description of SGIs and PPIs, see the Arm *Cortex-A15 MPCore Technical Reference Manual* (available at infocenter.arm.com/help/index.jsp).

C66x DSP Subsystem Interrupt Controller (DSP1_INTC)

The DSP1 subsystem integrates an interrupt controller - DSP1_INTC, which interfaces the system events to the C66x core interrupt and exceptions inputs. It combines up to 128 interrupts into 12 prioritized interrupts presented to the C66x CPU.

For detailed information about this module, see chapter *DSP Subsystem* of the Device TRM.

Dual Cortex-M4 IPU Subsystem Interrupt Controller (IPUx_Cx_INTC, where x = 1, 2)

There are two Image Processing Unit (IPU) subsystems in the device - IPU1, and IPU2. Each IPU subsystem integrates two Arm Cortex-M4 cores.

A Nested Vectored Interrupt Controller (NVIC) is integrated within each Cortex-M4. The interrupt mapping is the same (per IPU) for the two cores to facilitate parallel processing. The NVIC supports:

- 64 external interrupts (in addition to 16 Cortex-M4 internal interrupts), which are dynamically prioritized with 16 levels of priority defined for each core
- Low-latency exception and interrupt handling
- Prioritization and handling of exceptions
- Control of the local power management
- Debug accesses to the processor core

For detailed information about this module, refer to Arm *Cortex-M4 Technical Reference Manual* (available at infocenter.arm.com/help/index.jsp).

6.10 EDMA

The primary purpose of the Enhanced Direct Memory Access (EDMA) controller is to service user-programmed data transfers between two memory-mapped slave endpoints on the device.

Typical usage of the EDMA controller includes:

- Servicing software-driven paging transfers (for example, data movement between external memory [such as SDRAM] and internal memory [such as DSP L2 SRAM])
- Servicing event-driven peripherals, such as a serial port
- Performing sorting or sub-frame extraction of various data structures
- Offloading data transfers from the main device CPUs, such as the C66x DSP CorePac or the Arm CorePac

The EDMA controller consists of two major principle blocks:

- EDMA Channel Controller
- EDMA Transfer Controller(s)

The EDMA Channel Controller (EDMACC) serves as the user interface for the EDMA controller. The EDMACC includes parameter RAM (PaRAM), channel control registers, and interrupt control registers. The EDMACC serves to prioritize incoming software requests or events from peripherals and submits transfer requests (TR) to the EDMA transfer controller.

The EDMA Transfer Controller (EDMATC) is responsible for data movement. The transfer request packets (TRP) submitted by the EDMACC contain the transfer context, based on which the transfer controller issues read/write commands to the source and destination addresses programmed for a given transfer.

There are two EDMA controllers present on this device:

- EDMA_0, integrating:
  - 1 Channel Controller, referenced as: EDMACC_0
  - 2 Transfer Controllers, referenced as: EDMACC_0_TC_0 (or EDMATC_0) and EDMACC_0_TC_1 (or EDMATC_1)
- EDMA_1, integrating:
  - 1 Channel Controller, referenced as: EDMACC_1
  - 2 Transfer Controllers, referenced as: EDMACC_1_TC_0 (or EDMATC_2) and EDMACC_1_TC_1 (or EDMATC_3)

The two EDMA channel controllers (EDMACC_0 and EDMACC_1) are functionally identical. For simplification, the unified name EDMACC shall be regularly used throughout this chapter when referring to EDMA Channel Controllers functionality and features.

The four EDMA transfer controllers (EDMACC_0_TC_0, EDMACC_0_TC_1, EDMACC_1_TC_0 and EDMACC_1_TC_1) are functionally identical. For simplification, the unified name EDMATC shall be regularly used throughout this chapter when referring to EDMA Transfer Controllers functionality and features.

Each EDMACC has the following features:

- Fully orthogonal transfer description
  - 3 transfer dimensions:
    - Array (multiple bytes)
    - Frame (multiple arrays)
    - Block (multiple frames)
  - Single event can trigger transfer of array, frame, or entire block
  - Independent indexes on source and destination
- Flexible transfer definition
  - Increment or constant addressing modes
  - Linking mechanism allows automatic PaRAM set update
  - Chaining allows multiple transfers to execute with one event

- 64 DMA channels
  - Channels triggered by either:
    - Event synchronization
    - Manual synchronization (CPU write to event set register)
    - Chain synchronization (completion of one transfer triggers another transfer)
  - Support for programmable DMA Channel to PaRAM mapping
- 8 Quick DMA (QDMA) channels
  - QDMA channels are triggered automatically upon writing to PaRAM set entry
  - Support for programmable QDMA channel to PaRAM mapping
- 512 PaRAM sets
  - Each PaRAM set can be used for a DMA channel, QDMA channel, or link set
- 2 transfer controllers/event queues
  - 16 event entries per event queue
- Interrupt generation based on:
  - Transfer completion
  - Error conditions
- Debug visibility
  - Queue water marking/threshold
  - Error and status recording to facilitate debug
- Memory protection support
  - Proxied memory protection for TR submission
  - Active memory protection for accesses to PaRAM and registers

Each EDMATC has the following features:

- Supports 2-dimensional (2D) transfers with independent indexes on source and destination (EDMACC manages the 3rd dimension)
- Up to 4 in-flight transfer requests (TR)
- Programmable priority levels
- Support for increment or constant addressing mode transfers
- Interrupt and error support
- Supports only little-endian operation in this device
- Memory mapped register (MMR) bit fields are fixed position in 32-bit MMR

For more information chapter *EDMA Controller* of the device TRM.

6.11 Peripherals

6.11.1 VIP

The VIP module provides video capture functions for the device. VIP incorporates a multi-channel raw video parser, various video processing blocks, and a flexible Video Port Direct Memory Access (VPDMA) engine to store incoming video in various formats. The device uses a single instantiation of the VIP module giving the ability of capturing up to two video streams.

A VIP module includes the following main features:

- Two independently configurable external video input capture slices (Slice 0 and Slice 1) each of which has two video input ports, Port A and Port B, where Port A can be configured as a 24/16/8-bit port, and Port B is a fixed 8-bit port.
- Each video Port A can be operated as a port with clock independent input channels (with interleaved or separated Y/C data input). Embedded sync and external sync modes are supported for all input configurations.

- Support for a single external asynchronous pixel clock, up to 165MHz per port.
- Pixel Clock Input Domain Port A supports up to one 24-bit input data bus, including BT.1120 style embedded sync for 16-bit and 24-bit data.
- Embedded Sync data interface mode supports single or multiplexed sources
- Discrete Sync data interface mode supports only single source input
- 24-bit data input plus discrete syncs can be configured to include:
  - 8-bit YUV422 (Y and U/V time interleaved)
  - 16-bit YUV422 (CbY and CrY time interleaved)
  - 24-bit YUV444
  - 16-bit RGB565
  - 24-bit RGB888
  - 12/16-bit RAW Capture
  - 24-bit RAW capture
- Discrete sync modes include:
  - VSYNC + HSYNC (FID determined by FID signal pin or HSYNC/VSYNC skew)
  - VSYNC + ACTVID + FID
  - VBLANK + ACTVID (ACTVID toggles in VBLANK) + FID
  - VBLANK + ACTVID (no ACTVID toggles in VBLANK) + FID
- Multichannel parser (embedded syncs only)
  - Embedded syncs only
  - Pixel (2x or 4x) or Line multiplexed modes supported
  - Performs demultiplexing and basic error checking
  - Supports maximum of 9 channels in Line Mux (8 normal + 1 split line)
- Ancillary data capture support
  - For 16-bit or 24-bit input, ancillary data may be extracted from any single channel
  - For 8-bit time interleaved input, ancillary data can be chosen from the Luma channel, the Chroma channel, or both channels
  - Horizontal blanking interval data capture only supported when using discrete syncs (VSYNC + HSYNC or VSYNC + HBLANK)
  - Ancillary data extraction supported on multichannel capture as well as single source streams
- Format conversion and scaling
  - Programmable color space conversion
  - YUV422 to YUV444 conversion
  - YUV444 to YUV422 conversion
  - YUV422 to YUV420 conversion
  - YUV444 Source: YUV444 to YUV444, YUV444 to RGB888, YUV444 to YUV422, YUV444 to YUV420
  - RGB888 Source: RGB888 to RGB888, RGB888 to YUV444, RGB888 to YUV422, RGB888 to YUV420
  - YUV422 Source: YUV422 to YUV422, YUV422 to YUV420, YUV422 to YUV444, YUV422 to RGB888
  - Supports RAW to RAW (no processing)
  - Scaling and format conversions do not work for multiplexed input
- Supports up to 2047 pixels wide input - when scaling is engaged
- Supports up to 3840 pixels wide input - when only chroma up/down sampling is engaged, without scaling
- Supports up to 4095 pixels wide input - without scaling and chroma up/down sampling

- The maximum supported input resolution is further limited by:
  - Pixel clock and feature-dependent constraints
  - For RGB24-bit format (RAW data), the maximum frame width is limited to 2730 pixels

For more information, see chapter *Video Input Port* of the device TRM

6.11.2 DSS

Display Port Interfaces (DPI) is available in DSS named DPI Video Output (VOUT).

VOUT interface consists of:

- 24-bit data bus (data[23:0])
- Horizontal synchronization signal (HSYNC)
- Vertical synchronization signal (VSYNC)
- Data enable (DE)
- Field ID (FID)
- Pixel clock (CLK)

For more information, see section *Display Subsystem (DSS)* of the device TRM.

6.11.3 Timers

The device includes several types of timers used by the system software, including 16 general-purpose (GP) timers, one watchdog timer, and a 32-kHz synchronized timer (COUNTER_32K).

6.11.3.1 General-Purpose Timers

The device has 16 GP timers: TIMER1 through TIMER16.

- TIMER1(1ms tick): has its event capture pin tied to 32KHz clock and can be used to gauge the system clock input and detects its frequency among 19.2, 20, or 27 MHz. It includes a specific functions to generate accurate tick interrupts to the operating system and it belongs to the PD_WKUPAON domain
- TIMER2 and TIMER10: (1ms tick timers): they include a specific functions to generate accurate tick interrupts to the operating system, TIMER2 and TIMER10 belong to the PD_L4PER domain
- TIMER3/4/9/11/13/14/15/16: they belongs to the PD_L4PER domain
- TIMER12 belongs to the PD_WKUPAON power domain
- TIMER5 trough TIMER8: belong to the PD_IPU module

Each timer (except TIMER12) can be clocked from the system clock (19.2, 20, or 27 MHz) or the 32-kHz clock. The selection of clock source is made at the power, reset, and clock management (PRCM) module level. TIMER12 can be clocked only from the internal oscillator (on-die oscillator)

The following are the main features of the GP timer controllers:

- Level 4 (L4) slave interface support:
  - 32-bit data bus width
  - 32-/16-bit access supported
  - 8-bit access not supported
  - 10-bit address bus width
  - Burst mode not supported
  - Write nonposted transaction mode supported
- Interrupts generated on overflow, compare, and capture
- Free-running 32-bit upward counter
- Compare and capture modes
- Autoreload mode
- Start/stop mode

- Programmable divider clock source (2^n, where n = [0:8])
- Dedicated input trigger for capture mode and dedicated output trigger/PWM signal
- Dedicated GP output signal for using the TIMERi_GPO_CFG signal
- On-the-fly read/write register (while counting)
- 1-ms tick with 32.768-Hz functional clock generated (only TIMER1, TIMER2, and TIMER10)

For more information, see section *Timers* of the device TRM.

6.11.3.2 32-kHz Synchronized Timer (COUNTER_32K)

The 32-kHz synchronized timer (COUNTER_32K) is a 32-bit counter clocked by the falling edge of the 32-kHz system clock.

The main features of the 32-kHz synchronized timer controller are:

- L4 slave interface (OCP) support:
  - 32-bit data bus width
  - 32-/16-bit access supported
  - 8-bit access not supported
  - 16-bit address bus width
  - Burst mode not supported
  - Write nonposted transaction mode not supported
- Only read operations are supported on the module registers; no write operation is supported (no error/no action on write).
- Free-running 32-bit upward counter
- Start and keep counting after power-on reset
- Automatic roll over to 0; highest value reached: 0xFFFF FFFF
- On-the-fly read (while counting)

For more information, see section *Timers* of the device TRM.

6.11.3.3 Watchdog Timer

The device includes one instance of the 32-bit watchdog timer: WD_TIMER2.

The watchdog timer is an upward counter capable of generating a pulse on the reset pin and an interrupt to the device system modules following an overflow condition. The WD_TIMER2 timer serves resets to the PRCM module (its interrupt outputs are unused).

WD_TIMER2 is located in the PD_WKUPAON power domain, and can run when the device is in lowest power state (all power domains are off except always-on (AON) and WKUP).

The watchdog timer can be accessed, loaded, and cleared by registers through the L4_WKUP interface. The watchdog timer has the 32-kHz clock for its timer clock input. WD_TIMER2 directly generates a warm reset condition on overflow.

WD_TIMER2 connects to a single target agent port on the L4_WKUP interconnect.

The main features of the watchdog timer controllers are:

- L4 slave interface support:
  - 32-bit data bus width
  - 32-/16-bit access supported
  - 8-bit access not supported
  - 11-bit address bus width
  - Burst mode not supported
  - Write nonposted mode supported
- Free-running 32-bit upward counter

- Programmable divider clock source (2^n where n = [0:7])
- On-the-fly read/write register (while counting)
- Subset programming model of the GP timer
- The watchdog timer is reset either on power on or after a warm reset before it starts counting.
- Reset or interrupt actions when a timer overflow condition occurs
- The watchdog timer generates a reset or an interrupt in its hardware integration.

For more information, see section *Timers* of the device TRM.

6.11.4 I2C

The device contains five multimaster high-speed (HS) inter-integrated circuit (I^2C) controllers (I2C*i* modules, where *i* = 1, 2 ,3, 4, 5, 6) each of which provides an interface between a local host (LH), such as a digital signal processor (DSP), and any I^2C-bus-compatible device that connects through the I^2C serial bus. External components attached to the I^2C bus can serially transmit and receive up to 8 bits of data to and from the LH device through the 2-wire I^2C interface.

Each multimaster HS I^2C controller can be configured to act like a slave or master I^2C-compatible device.

I^2C1 and I^2C2 controllers have dedicated I^2C compliant open drain buffers, and support Fast mode (up to 400Kbps). I^2C3, I^2C4, I^2C5 and I^2C6 controllers are multiplexed with standard LVCMOS IO and connected to emulate open drain. I^2C emulation is achieved by configuring the LVCMOS buffers to output Hi-Z instead of driving high when transmitting logic 1. These controllers support HS mode (up to 3.4Mbps).

For more information, see section *Multimaster High-Speed I2C Controller (I2C)* in chapter *Serial Communication Interfaces* of the device TRM.

6.11.5 UART

The UART is a simple L4 slave peripheral that utilizes the DMA_SYSTEM or EDMA for data transfer or IRQ polling via CPU. There are 10 UART modules in the device. Only one UART supports IrDA features. Each UART can be used for configuration and data exchange with a number of external peripheral devices or interprocessor communication between devices.

6.11.5.1 UART Features

The UARTi (where i = 1 to 10) include the following features:

- 16C750 compatibility
- 64-byte FIFO buffer for receiver and 64-byte FIFO for transmitter
- Programmable interrupt trigger levels for FIFOs
- Baud generation based on programmable divisors N (where N = 1…16,384) operating from a fixed functional clock of 48 MHz or 192 MHz

Oversampling is programmed by software as 16 or 13. Thus, the baud rate computation is one of two options:

- Baud rate = (functional clock / 16) / N
- Baud rate = (functional clock / 13) / N
- This software programming mode enables higher baud rates with the same error amount without changing the clock source
- Break character detection and generation
- Configurable data format:
  - Data bit: 5, 6, 7, or 8 bits
  - Parity bit: Even, odd, none
  - Stop-bit: 1, 1.5, 2 bit(s)
- Flow control: Hardware (RTS/CTS) or software (XON/XOFF)

- The 48 MHz functional clock option allows baud rates up to 3.6Mbps
- The 192 MHz functional clock option allows baud rates up to 12Mbps
- UART1 module has extended modem control signals (DCD, RI, DTR, DSR)
- UART3 supports IrDA

6.11.5.2 IrDA Features

UART3 supports the following IrDA key features:

- Support of IrDA 1.4 slow infrared (SIR), medium infrared (MIR), and fast infrared (FIR) communications:
  - Frame formatting: Addition of variable beginning-of-frame (xBOF) characters and end-of-frame (EOF) characters
  - Uplink/downlink cyclic redundancy check (CRC) generation/detection
  - Asynchronous transparency (automatic insertion of break character)
  - Eight-entry status FIFO (with selectable trigger levels) to monitor frame length and frame errors
  - Framing error, CRC error, illegal symbol (FIR), and abort pattern (SIR, MIR) detection

6.11.5.3 CIR Features

The CIR mode uses a variable pulse-width modulation (PWM) technique (based on multiples of a programmable *t* period) to encompass the various formats of infrared encoding for remote-control applications. The CIR logic transmits data packets based on a user-definable frame structure and packet content.

The CIR (UART3 only) includes the following features to provide CIR support for remote-control applications:

- Transmit mode only (receive mode is not supported)
- Free data format (supports any remote-control private standards)
- Selectable bit rate
- Configurable carrier frequency
- 1/2, 5/12, 1/3, or 1/4 carrier duty cycle

For more information, see section *Universal Asynchronous Receiver/Transmitter (UART)* in chapter *Serial Communication Interfaces* of the device TRM.

6.11.6 McSPI

The McSPI is a master/slave synchronous serial bus. There are four separate McSPI modules (McSPI1, McSPI2, McSPI3, and McSPI4) in the device. All these four modules support up to four external devices (four chip selects) and are able to work as both master and slave.

The McSPI modules include the following main features:

- Serial clock with programmable frequency, polarity, and phase for each channel
- Wide selection of McSPI word lengths, ranging from 4 to 32 bits
- Up to four master channels, or single channel in slave mode
- Master multichannel mode:
  - Full duplex/half duplex
  - Transmit-only/receive-only/transmit-and-receive modes
  - Flexible input/output (I/O) port controls per channel
  - Programmable clock granularity
  - McSPI configuration per channel. This means, clock definition, polarity enabling and word width
- Single interrupt line for multiple interrupt source events
- Power management through wake-up capabilities
- Enable the addition of a programmable start-bit for McSPI transfer per channel (start-bit mode)

- Supports start-bit write command
- Supports start-bit pause and break sequence
- Programmable timing control between chip select and external clock generation
- Built-in FIFO available for a single channel

For more information, see section *Serial Peripheral Interface (McSPI)* in chapter *Serial Communication Interfaces* of the device TRM.

6.11.7 QSPI

The quad serial peripheral interface (QSPI™) module is a kind of SPI module that allows single, dual, or quad read access to external SPI devices. This module has a memory mapped register interface, which provides a direct interface for accessing data from external SPI devices and thus simplifying software requirements. The QSPI works as a master only.

The QSPI supports the following features:

- General SPI features:
  - Programmable clock divider
  - Six pin interface
  - Programmable length (from 1 to 128 bits) of the words transferred
  - Programmable number (from 1 to 4096) of the words transferred
  - 4 external chip-select signals
  - Support for 3-, 4-, or 6-pin SPI interface
  - Optional interrupt generation on word or frame (number of words) completion
  - Programmable delay between chip select activation and output data from 0 to 3 QSPI clock cycles
  - Programmable signal polarities
  - Programmable active clock edge
  - Software-controllable interface allowing for any type of SPI transfer
  - Control through L3_MAIN configuration port
- Serial flash interface (SFI) features:
  - Serial flash read/write interface
  - Additional registers for defining read and write commands to the external serial flash device
  - 1 to 4 address bytes
  - Fast read support, where fast read requires dummy bytes after address bytes; 0 to 3 dummy bytes can be configured.
  - Dual read support
  - Quad read support
  - Little-endian support only
  - Linear increment addressing mode only

The QSPI supports only dual and quad reads. Dual or quad writes are not supported. In addition, there is no "pass through" mode supported where the data present on the QSPI input is sent to its output.

For more information, see section *Quad Serial Peripheral Interface (QSPI)* in chapter *Serial Communication Interfaces* of the device TRM.

6.11.8 McASP

The McASP functions as a general-purpose audio serial port optimized to the requirements of various audio applications. The McASP module can operate in both transmit and receive modes. The McASP is useful for time-division multiplexed (TDM) stream, Inter-IC Sound (I2S) protocols reception and transmission as well as for an intercomponent digital audio interface transmission (DIT). The McASP has the flexibility to gluelessly connect to a Sony/Philips digital interface (S/PDIF) transmit physical layer component.

Although intercomponent digital audio interface reception (DIR) mode (i.e. S/PDIF stream receiving) is not natively supported by the McASP module, a specific TDM mode implementation for the McASP receivers allows an easy connection to external DIR components (for example, S/PDIF to I2S format converters).

The device have integrated 8 McASP modules (McASP1-McASP8) with:

- McASP1 and McASP2 supporting 16 channels with independent TX/RX clock/sync domain
- McASP3 through McASP7 modules supporting 4 channels with independent TX/RX clock/sync domain

- McASP8 module supporting 2 channels with independent TX/RX clock/sync domain

For more information, see section *Multichannel Audio Serial Port (McASP)* in chapter *Serial Communication Interfaces* of the device TRM.

6.11.9 USB

SuperSpeed USB DRD Subsystem has three instances in the device providing the following functions:

- USB1: SuperSpeed (SS) USB 3.0 Dual-Role-Device (DRD) subsystem with integrated SS (USB3.0) PHY and HS/FS (USB2.0) PHY
- USB2: High-Speed (HS) USB 2.0 Dual-Role-Device (DRD) subsystem with integrated HS/FS PHY
- USB3: HS USB 2.0 Dual-Role-Device (DRD) subsystem with ULPI (SDR) interface to external HS/FS PHYs

SuperSpeed USB DRD Subsystem has the following features:

- Dual-role-device (DRD) capability:
  - Supports USB Peripheral (or Device) mode at speeds SS (5Gbps)(USB1 only), HS (480 Mbps), and FS (12 Mbps)
  - Supports USB Host mode at speeds SS (5Gbps)(USB1 only), HS (480 Mbps), FS (12 Mbps), and LS (1.5 Mbps)
  - USB static peripheral operation
  - USB static host operation
  - Flexible stream allocation
  - Stream priority
  - External Buffer Control
- Each instance contains single xHCI controller with the following features:
  - Internal DMA controller
  - Descriptor caching and data prefetching
  - Interrupt moderation and blocking
  - Power management USB3.0 states for U0, U1, U2, and U3
  - Dynamic FIFO memory allocation for all endpoints
  - Supports all modes of transfers (control, bulk, interrupt, and isochronous)
  - Supports high bandwidth ISO mode
- Connects to an external charge pump for VBUS 5 V generation
- USB-HS PHY (USB2PHY1 and USB2PHY2 for USB1 and USB2, respectively): contain the USB functions, drivers, receivers, and pads for correct D+/D– signalling
- USB3PHY. The USB3PHY is embedded in the USB1 subsystem and contains:
  - USB3RX_PHY deserializer to receive data at SuperSpeed mode
  - USB3TX_PHY serializer to transmit data at SuperSpeed mode
  - Power sequencer that contains a power control state machine, generating the sequences to power up/down the USB3RX_PHY/USB3TX_PHY
  - Dedicated DPLL (DPLL_USB_OTG_SS)

For more information, see section *SuperSpeed USB DRD (USB)* in chapter *Serial Communication Interfaces* of the device TRM.

6.11.10 PCIe

The Peripheral Component Interconnect Express (PCIe) module is a multi-lane I/O interconnect that provides low pin-count, high reliability, and high-speed data transfer at rates of up to 5.0 Gbps per lane, per direction, for serial links on backplanes and printed wiring boards. It is a 3-rd Generation I/O Interconnect technology succeeding PCI and ISA bus that is designed to be used as a general-purpose serial I/O interconnect. It is also used as a bridge to other interconnects like USB2/3.0, GbE MAC, and so forth.

The PCI Express standard predecessor - PCI, is a parallel bus architecture that is increasingly difficult to scale-up in bandwidth, which is usually performed by increasing the number of data signal lines. The PCIe architecture was developed to help minimize I/O bus bottlenecks within systems and to provide the necessary bandwidth for high-speed, chip-to-chip, and board-to-board communications within a system. It is designed to replace the PCI-based shared, parallel bus signaling technology that is approaching its practical performance limits while simplifying the interface design.

The device instantiates two PCIe subsystems (PCIe_SS1 and PCIe_SS2). The PCIe controller is capable to operate either in Root Complex (RC) or in End Point (EP) PCIe mode. The device PCIe_SS1 controller supports up to two 16-bit data lanes on its PIPE port. The device PCIe_SS2 controller supports only one 16-bit data lane on its PIPE port.

When the PCIe_SS1 controller PIPE port is configured to operate in a single-lane mode, it operates on a single pair of PCIe PHY serializer and deserializer - PCIe1_PHY_TX/PCIe1_PHY_RX. When PCIe_SS1 PIPE is configured to operate in dual-lane mode, it operates on two pairs of PCIe PHY serializer and deserializer - PCIe1_PHY_TX/PCIe1_PHY_RX and PCIe2_PHY_TX/PCIe2_PHY_RX, respectively. The single-lane PCIe_SS2 controller PIPE port (if enabled) can operate only on the PCIe2_PHY_TX/PCIe2_PHY_RX pair. Hereby, if PCIe_SS2 controller is used, the PCIe_SS1 can operate only in a single-lane mode on the PCIe1_PHY_TX/PCIe1_PHY_RX. In addition, PCIe PHY subsystem encompasses a PCIe PCS (physical coding sublayer), a PCIe power management logic, APLL, a DPLL reference clock generator and an APLL clock low-jitter buffer.

- The PCIe Controller implements the transport and link layers of the PCIe interface protocol.
- PCIe PCS (a physical coding sublayer component) converts a 8-bit portion of parallel data over a PCIe lane to a 10-bit parallel data to adapt the process of serialization and deserialization in the TX/RX PHYs to various requirements. At the same time it transforms the transmission rate to maintain the PCIe Gen2 bandwidth (5 Gbps) on both sides (PCIe controller and PHY).
- A multiplexer logic which adds flexibility to connect a PCIe controller hardware mapped PCS logic output to a single (for the single-lane PCIe_SS2 controller) or to a couple (for the 2-lane PCIe_SS1 controller) of PHY ports at a time
- Physical layer (PHY) serializer/deserializer components with associated power control logic, building the so called PMA (physical media attachment) part of the PCIe_PHY transceiver, as follows:
  - PCIe physical port 0 associated serializer (TX) - PCIe1_PHY_TX and deserializer (RX) - PCIe1_PHY_RX
  - PCIe physical port 1 associated serializer (TX) - PCIe2_PHY_TX and deserializer (RX) - PCIe2_PHY_RX
- DPLL_PCIe_REF is a DPLL clock source, controlled from the device PRCM, that provides a 100-MHz clock to the PCIe PHY serializer/deserializer components reference clock inputs.
- Both the PCIe_SS1 and PCIe_SS2 share the same APLL (APLLPCIe) which by default multiplies the DPLL_PCIe_REF (typically 100 MHz or 20 MHz) clock to 2.5 GHz.
- The APLLPCIe low-jitter buffer (ACSPCIE) and additional logic takes care to provide the PCIe APLL reference input clock.

PCIe module supports the following features:

- PCI Local Bus Specification revision 3.0
- PCI Express Base 3.0 Specification, revision 1.0.

At system level the device supports PCI express interface in the following configurations:

- Each PCIe subsystem controller has support for PCIe Gen2 mode (5.0 Gbps per lane) and Gen1 mode (2.5 Gbps per lane).
- One PCIe (PCIe_SS1) operates as Gen2 2-lanes supporting in either root-complex (RC) or end-point EP.
- Two PCIe (PCIe_SS1 and PCIe_SS2) operates Gen2 1-lane supporting either RC or EP with the possibility of one operating in Gen1 and one in Gen2.
- PCIe_SS1 can be configured to operate in either 2-Lane (dual lane) or 1-Lane (single lane) mode, as follows:
  - Single Lane - lane 0 mapped to the PCIe port 0 of the device
  - Flexible dual lane configuration - lanes 0 and 1 can be swapped on the two PCIe ports
- PCIe_SS2 can only operate in 1-Lane mode, as follows:
  - Single Lane - lane 0 mapped to the device PCIe port 1

  When PCIe_SS1 is configured to operate in dual-lane mode, PCIe_SS2 is in-operable as both PCIe1_PHY_RX/TX and PCIe2_PHY_RX/TX are assigned to PCIe_SS1, and thereby NOT available to PCIe_SS2.

The main features of a device PCIe controller are:

- 16-bit operation at 250 MHz on PIPE interface (per 16-bit lane)
- One master port on the L3_MAIN supporting 32-bit address and 64-bit data bus.
- PCIe_SS1 master port dedicated MMU (device MMU2) on L3_MAIN path, to which PCIe traffic can be optionally mapped.
- One slave port on the L3_MAIN supporting 29-bit address and 64-bit data bus.
- Maximum outbound payload size of 64 Bytes (the L3 Interconnect PCIe1/2 target ports split bursts of size >64 Bytes to the into multiple 64 Byte bursts)
- Maximum inbound payload size of 256 Bytes (internally converted to 128 Byte - bursts)
- No remote read request size limit: implicit support for 4 KiB-size and greater
- Support of EP legacy mode
- Support of inbound I/O accesses in EP legacy mode
- PIPE interface features fixed-width (16-bit data per lane) and dynamic frequency to switch between PCIe Gen1 and Gen2.
- Ultra-low transmit and receive latency
- Automatic Lane reversal as specified in the PCI Express Base 3.0 Specification, revision 1.0 (transmit and receive)
- Polarity inversion on receive
- Single Virtual Channel (VC0) and Single Traffic Class (TC0)
- Single Function in End point mode
- Automatic credit management
- ECRC generation and checking
- All PCI Device Power Management D-states with the exception of $D3_{cold}$/L2 state
- PCI Express Active State Power Management (ASPM) state L0s and L1 (with exceptions)
- PCI Express Link Power Management states except for L2 state
- PCI Express Advanced Error Reporting (AER)
- PCI Express messages for both transmit and receive
- Filtering for Posted, Non-Posted, and Completion traffic
- Configurable BAR filtering, I/O filtering, configuration filtering and completion lookup/timeout
- Access to configuration space registers and external application memory mapped registers through ECAM mechanism.

- Legacy PCI Interrupts reception (RC) and generation (EP)
- 2 x hardware interrupts per PCIe_SS1and PCIe_SS2 controller mapped via the device Interrupt Crossbar (IRQ_CROSSBAR) to multiple device host (MPU, DSP, and so forth) interrupt controllers in the device
- MSIs generation and reception
- PCIe_PHY Loopback in RC mode

For more information, see section *PCIe Controller* in chapter *Serial Communication Interfaces* of the device TRM.

6.11.11 DCAN

The Controller Area Network (CAN) is a serial communications protocol which efficiently supports distributed real-time applications. CAN has high immunity to electrical interference and the ability to self-diagnose and repair data errors. In a CAN network, many short messages are broadcast to the entire network, which provides for data consistency in every node of the system.

The device provides two DCAN interfaces for supporting distributed realtime control with a high level of security. The DCAN interfaces implement the following features:

- Supports CAN protocol version 2.0 part A, B
- Bit rates up to 1 MBit/s
- 64 message objects
- Individual identifier mask for each message object
- Programmable FIFO mode for message objects
- Programmable loop-back modes for self-test operation
- Suspend mode for debug support
- Software module reset
- Automatic bus on after Bus-Off state by a programmable 32-bit timer
- Direct access to Message RAM during test mode
- CAN Rx/Tx pins are configurable as general-purpose IO pins
- Two interrupt lines (plus additional parity-error interrupts line)
- RAM initialization
- DMA support

For more information, see section *DCAN* in chapter *Serial Communication Interfaces* of the device TRM.

6.11.12 GMAC_SW

The three-port gigabit ethernet switch subsystem (GMAC_SW) provides ethernet packet communication and can be configured as an ethernet switch. It provides the gigabit media independent interface (G/MII) in MII mode, reduced gigabit media independent interface (RGMII), reduced media independent interface (RMII), and the management data input output (MDIO) for physical layer device (PHY) management.

The GMAC_SW subsystem provides the following features:

- Two Ethernet ports (port 1 and port 2) with selectable RGMII, RMII, and G/MII (in MII mode only) interfaces plus internal Communications Port Programming Interface (CPPI 3.1) on port 0
- Synchronous 10/100/1000 Mbit operation
- Wire rate switching (802.1d)
- Non-blocking switch fabric
- Flexible logical FIFO-based packet buffer structure
- Four priority level Quality Of Service (QOS) support (802.1p)
- CPPI 3.1 compliant DMA controllers
- Support for Audio/Video Bridging (P802.1Qav/D6.0)

- Support for IEEE 1588 Clock Synchronization (2008 Annex D and Annex F)
  - Timing FIFO and time stamping logic embedded in the subsystem
- Device Level Ring (DLR) Support
- Energy Efficient Ethernet (EEE) support (802.3az)
- Flow Control Support (802.3x)
- Address Lookup Engine (ALE)
  - 1024 total address entries plus VLANs
  - Wire rate lookup
  - Host controlled time-based aging
  - Multiple spanning tree support (spanning tree per VLAN)
  - L2 address lock and L2 filtering support
  - MAC authentication (802.1x)
  - Receive-based or destination-based multicast and broadcast rate limits
  - MAC address blocking
  - Source port locking
  - OUI (Vendor ID) host accept/deny feature
  - Remapping of priority level of VLAN or ports
- VLAN support
  - 802.1Q compliant
    - Auto add port VLAN for untagged frames on ingress
    - Auto VLAN removal on egress and auto pad to minimum frame size
- Ethernet Statistics:
  - EtherStats and 802.3Stats Remote network Monitoring (RMON) statistics gathering (shared)
  - Programmable statistics interrupt mask when a statistic is above one half its 32-bit value
- Flow Control Support (802.3x)
- Digital loopback and FIFO loopback modes supported
- Maximum frame size 2016 bytes (2020 with VLAN)
- 8k (2048 × 32) internal CPPI buffer descriptor memory
- Management Data Input/Output (MDIO) module for PHY Management
- Programmable interrupt control with selected interrupt pacing
- Emulation support
- Programmable Transmit Inter Packet Gap (IPG)
- Reset isolation (switch function remains active even in case of all device resets except for POR pin reset and ICEPICK cold reset)
- Full duplex mode supported in 10/100/1000 Mbps. Half-duplex mode supported only in 10/100 Mbps.
- IEEE 802.3 gigabit Ethernet conformant

For more information, see section *Gigabit Ethernet Switch (GMAC_SW)* in chapter *Serial Communication Interfaces* of the device TRM.

6.11.13 eMMC/SD/SDIO

The eMMC/SD/SDIO host controller provides an interface between a local host (LH) such as a microprocessor unit (MPU) or digital signal processor (DSP) and either eMMC, SD® memory cards, or SDIO cards and handles eMMC/SD/SDIO transactions with minimal LH intervention.

Optionally, the controller is connected to the L3_MAIN interconnect to have a direct access to system memory. It also supports two direct memory access (DMA) slave channels or a DMA master access (in this case, slave DMA channels are deactivated) depending on its integration.

The eMMC/SD/SDIO host controller deals with eMMC/SD/SDIO protocol at transmission level, data packing, adding cyclic redundancy checks (CRCs), start/end bit, and checking for syntactical correctness.

The application interface can send every eMMC/SD/SDIO command and poll for the status of the adapter or wait for an interrupt request, which is sent back in case of exceptions or to warn of end of operation.

The application interface can read card responses or flag registers. It can also mask individual interrupt sources. All these operations can be performed by reading and writing control registers. The eMMC/SD/SDIO host controller also supports two DMA channels.

There are four eMMC/SD/SDIO host controllers inside the device. gives an overview of the eMMC/SD/SDIOi (i = 1 to 4) controllers.

Each controller has the following data width:

- eMMC/SD/SDIO1 - 4-bit wide data bus
- eMMC/SD/SDIO2 - 8-bit wide data bus
- eMMC/SD/SDIO3 - 4-bit wide data bus
- eMMC/SD/SDIO4 - 4-bit wide data bus

The eMMC/SD/SDIOi controller is also referred to as MMCi.

Compliance with standards:

- Full compliance with MMC/eMMC command/response sets as defined in the JC64 MMC/eMMC standard specification, v4.5.
- Full compliance with SD command/response sets as defined in the SD Physical Layer specification v3.01
- Full compliance with SDIO command/response sets and interrupt/read-wait suspend-resume operations as defined in the SD part E1 specification v3.00
- Full compliance with SD Host Controller Standard Specification sets as defined in the SD card specification Part A2 v3.00

Main features of the eMMC/SD/SDIO host controllers:

- Flexible architecture allowing support for new command structure
- 32-bit wide access bus to maximize bus throughput
- Designed for low power
- Programmable clock generation
- Dedicated DLL to support SDR104 mode (MMC1 only)
- Dedicated DLL to support HS200 mode (MMC2 only)
- Card insertion/removal detection and write protect detection
- L4 slave interface supports:
  - 32-bit data bus width
  - 8/16/32 bit access supported
  - 9-bit address bus width
  - Streaming burst supported only with burst length up to 7
  - WNP supported
- L3 initiator interface Supports:
  - 32-bit data bus width
  - 8/16/32 bit access supported
  - 32-bit address bus width
  - Burst supported
- Built-in 1024-byte buffer for read or write
- Two DMA channels, one interrupt line
- Support JC 64 v4.4.1 boot mode operations

- Support SDA 3.00 Part A2 programming model
- Support SDA 3.00 Part A2 DMA feature (ADMA2)
- Supported data transfer rates:
  - MMCi supports the following SD v3.0 data transfer rates:
    - DS mode (3.3V IOs): up to 12 MBps (24 MHz clock)
    - HS mode (3.3V IOs): up to 24 MBps (48 MHz clock)
    - SDR12 (1.8V IOs): up to 12 MBps (24 MHz clock)
    - SDR25 (1.8V IOs): up to 24 MBps (48 MHz clock)
    - SDR50 (1.8V IOs): up to 48 MBps (96 MHz clock) - MMC1 and MMC3 only
    - DDR50 (1.8V IOs): up to 48 MBps (48 MHz clock) - MMC1 only
    - SDR104 (1.8V IOs) cards can be supported up to 192 MHz clock (96 MBps max) - MMC1 only
  - MMCi supports the Default SD mode 1-bit data transfer up to 24Mbps (3MBps)
  - Only MMC2 supports also the following JC64 v4.5 data transfer rates:
    - Up to 192 MBps in eMMC mode, 8-bit SDR mode (192 MHz clock frequency)
    - Up to 96 MBps in eMMC mode, 8-bit DDR mode (48 MHz clock frequency)
- All eMMC/SD/SDIO controllers are connected to 1,8V/3.3V compatible I/Os to support 1,8V/3.3V signaling

NOTE

eMMC functionality is supported fully by MMC2 only. The other MMC modules are capable of eMMC functionality, but are not timing-optimized for eMMC.

The differences between the eMMC/SD/SDIO host controllers and a standard SD host controller defined by the *SD Card Specification, Part A2, SD Host Controller Standard Specification*, v3.00 are:

- The clock divider in the eMMC/SD/SDIO host controller supports a wider range of frequency than specified in the *SD Memory Card Specifications*, v3.0. The eMMC/SD/SDIO host controller supports odd and even clock ratio.
- The eMMC/SD/SDIO host controller supports configurable busy time-out.
- ADMA2 64-bit mode is not supported.
- There is no external LED control.

NOTE

Only even ratios are supported in DDR mode.

For more information, see chapter *eMMC/SD/SDIO* of the device TRM.

6.11.14 GPIO

The general-purpose interface combines eight general-purpose input/output (GPIO) banks.

Each GPIO module provides 32 dedicated general-purpose pins with input and output capabilities; thus, the general-purpose interface supports up to 186 pins.

These pins can be configured for the following applications:

- Data input (capture)/output (drive)
- Keyboard interface with a debounce cell
- Interrupt generation in active mode upon the detection of external events. Detected events are processed by two parallel independent interrupt-generation submodules to support biprocessor operations.
- Wake-up request generation in idle mode upon the detection of external events

For more information, see chapter *General-Purpose Interface (GPIO)* of the device TRM.

6.11.15 ePWM

An effective PWM peripheral must be able to generate complex pulse width waveforms with minimal CPU overhead or intervention. It needs to be highly programmable and very flexible while being easy to understand and use. The ePWM unit described here addresses these requirements by allocating all needed timing and control resources on a per PWM channel basis. Cross coupling or sharing of resources has been avoided; instead, the ePWM is built up from smaller single channel modules with separate resources and that can operate together as required to form a system. This modular approach results in an orthogonal architecture and provides a more transparent view of the peripheral structure, helping users to understand its operation quickly.

Each ePWM module supports the following features:

- **Dedicated 16-bit time-base counter with period and frequency control**
- **Two PWM outputs (EPWMxA and EPWMxB) that can be used in the following configurations:**
  - Two independent PWM outputs with single-edge operation
  - Two independent PWM outputs with dual-edge symmetric operation
  - One independent PWM output with dual-edge asymmetric operation
- **Asynchronous override control of PWM signals through software.**
- **Programmable phase-control support for lag or lead operation relative to other ePWM modules.**
- **Hardware-locked (synchronized) phase relationship on a cycle-by-cycle basis.**
- **Dead-band generation with independent rising and falling edge delay control.**
- **Programmable trip zone allocation of both cycle-by-cycle trip and one-shot trip on fault conditions.**
- **A trip condition can force either high, low, or high-impedance state logic levels at PWM outputs.**
- **Programmable event prescaling minimizes CPU overhead on interrupts.**
- **PWM chopping by high-frequency carrier signal, useful for pulse transformer gate drives.**

For more information, see section *Enhanced PWM (ePWM) Module* in chapter *Pulse-Width Modulation Subsystem* of the device TRM.

6.11.16 eCAP

Uses for eCAP include:

- Sample rate measurements of audio inputs
- Speed measurements of rotating machinery (for example, toothed sprockets sensed via Hall sensors)
- Elapsed time measurements between position sensor pulses
- 4 stage sequencer (Mod4 counter) which is synchronized to external events (ECAPx pin edges)

- Period and duty cycle measurements of pulse train signals
- Decoding current or voltage amplitude derived from duty cycle encoded current/voltage sensors

The eCAP module includes the following features:

- 32-bit time base counter
- 4-event time-stamp registers (each 32 bits)
- Edge polarity selection for up to four sequenced time-stamp capture events
- Interrupt on either of the four events
- Single shot capture of up to four event time-stamps
- Continuous mode capture of time-stamps in a four-deep circular buffer
- Absolute time-stamp capture
- Difference (Delta) mode time-stamp capture
- All above resources dedicated to a single input pin
- When not used in capture mode, the ECAP module can be configured as a single channel PWM output

For more information, see section *Enhanced Capture (eCAP) Module* in chapter *Pulse-Width Modulation Subsystem* of the device TRM.

6.11.17 eQEP

A single track of slots patterns the periphery of an incremental encoder disk, as shown in Figure 6-2. These slots create an alternating pattern of dark and light lines. The disk count is defined as the number of dark/light line pairs that occur per revolution (lines per revolution). As a rule, a second track is added to generate a signal that occurs once per revolution (index signal: QEPI), which can be used to indicate an absolute position. Encoder manufacturers identify the index pulse using different terms such as index, marker, home position, and zero reference.

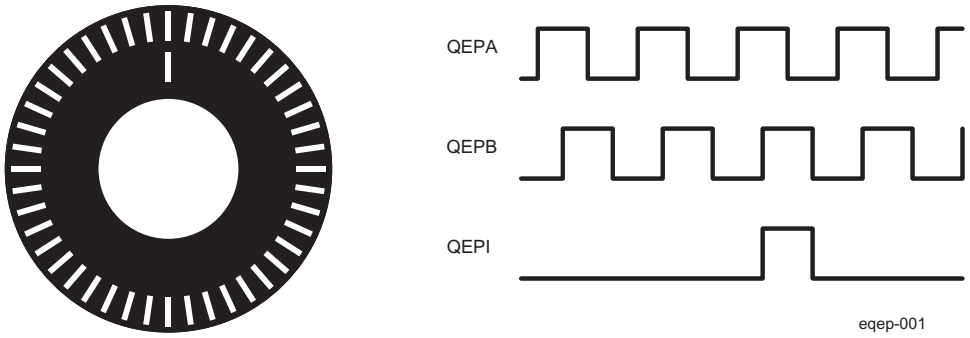


Figure 6-2. Optical Encoder Disk

To derive direction information, the lines on the disk are read out by two different photo-elements that "look" at the disk pattern with a mechanical shift of 1/4 the pitch of a line pair between them. This shift is realized with a reticle or mask that restricts the view of the photo-element to the desired part of the disk lines. As the disk rotates, the two photo-elements generate signals that are shifted 90 degrees out of phase from each other. These are commonly called the quadrature QEPA and QEPB signals. The clockwise direction for most encoders is defined as the QEPA channel going positive before the QEPB channel.

The encoder wheel typically makes one revolution for every revolution of the motor or the wheel may be at a geared rotation ratio with respect to the motor. Therefore, the frequency of the digital signal coming from the QEPA and QEPB outputs varies proportionally with the velocity of the motor. For example, a 2000-line encoder directly coupled to a motor running at 5000 revolutions per minute (rpm) results in a frequency of 166.6 KHz, so by measuring the frequency of either the QEPA or QEPB output, the processor can determine the velocity of the motor.

For more information, see section *Enhanced Quadrature Encoder Pulse (eQEP) Module* in chapter *Pulse-Width Modulation Subsystem* of the device TRM.

6.12 On-chip Debug

Debugging a system that contains an embedded processor involves an environment that connects high-level debugging software running on a host computer to a low-level debug interface supported by the target device. Between these levels, a debug and trace controller (DTC) facilitates communication between the host debugger and the debug support logic on the target chip.

The DTC is a combination of hardware and software that connects the host debugger to the target system. The DTC uses one or more hardware interfaces and/or protocols to convert actions dictated by the debugger user to JTAG® commands and scans that execute the core hardware.

The debug software and hardware components let the user control multiple central processing unit (CPU) cores embedded in the device in a global or local manner. This environment provides:

- Synchronized global starting and stopping of multiple processors
- Starting and stopping of an individual processor
- Each processor can generate triggers that can be used to alter the execution flow of other processors

System topics include but are not limited to:

- System clocking and power-down issues
- Interconnection of multiple devices
- Trigger channels

For more information, see chapter *On-chip Debug* of the device TRM.

The device deploys Texas Instrument's CTools debug technology for on-chip debug and trace support. It provides the following features:

- External debug interfaces:
  - Primary debug interface - IEEE1149.1 (JTAG) or IEEE1149.7 (complementary superset of JTAG)
    - Used for debugger connection
    - Default mode is IEEE1149.1 but debugger can switch to IEEE1149.7 via an IEEE1149.7 adapter module
    - Controls ICEPick (generic test access port [TAP] for dynamic TAP insertion) to allow the debugger to access several debug resources through its secondary (output) JTAG ports (for more information, see *ICEPick Secondary TAPs* section of the Device TRM).
  - Debug (trace) port
    - Can be used to export processor or system trace off-chip (to an external trace receiver)
    - Can be used for cross-triggering with an external device
    - Configured through debug resources manager (DRM) module instantiated in the debug subsystem
    - For more information about debug (trace) port, see *Debug (Trace) Port* and *Concurrent Debug Modes* sections of the Device TRM.
- JTAG based processor debug on:
  - Cortex-A15 in MPU
  - C66x in DSP1
  - Cortex-M4 (x2) in IPU1, IPU2
- Dynamic TAP insertion
  - Controlled by ICEPick
  - For more information, see , *Dynamic TAP Insertion.*

- Power and clock management
  - Debugger can get the status of the power domain associated to each TAP.
  - Debugger may prevent the application software switching off the power domain.
  - Application power management behavior can be preserved during debug across power transitions.
  - For more information, see *Power and Clock Management* section of the Device TRM.
- Reset management
  - Debugger can configure ICEPick to assert, block, or extend the reset of a given subsystem.
  - For more information, see *Reset Management* section of the Device TRM.
- Cross-triggering
  - Provides a way to propagate debug (trigger) events from one processor, subsystem, or module to another:
    - Subsystem A can be programmed to generate a debug event, which can then be exported as a global trigger across the device.
    - Subsystem B can be programmed to be sensitive to the trigger line input and to generate an action on trigger detection.
  - Two global trigger lines are implemented
  - Device-level cross-triggering is handled by the XTRIG (TI cross-trigger) module implemented in the debug subsystem
  - Various Arm® CoreSight cross-trigger modules implemented to provide support for CoreSight triggers distribution
    - CoreSight Cross-Trigger Interface (CS_CTI) modules
    - CoreSight Cross-Trigger Matrix (CS_CTM) modules
  - For more information about cross-triggering, see *Cross-Triggering* section of the Device TRM.
- Suspend
  - Provides a way to stop a closely coupled hardware process running on a peripheral module when the host processor enters debug state
  - For more information about suspend, see *Suspend* section of the Device TRM.
- MPU watchpoint
  - Embedded in MPU subsystem
  - Provides visibility on MPU to EMIF direct paths
  - For more information, see *MPU Memory Adaptor (MPU_MA) Watchpoint* section of the Device TRM.
- Processor trace
  - Cortex-A15 (MPU) and C66x (DSP) processor trace is supported
  - Program trace only for MPU (no data trace)
  - MPU trace supported by a CoreSight Program Trace Macrocell (CS_PTM) module
  - Three exclusive trace sinks:
    - CoreSight Trace Port Interface Unit (CS_TPIU) – trace export to an external trace receiver
    - CTools Trace Buffer Router (CT_TBR) in system bridge mode – trace export through USB
    - CT_TBR in buffer mode – trace history store into on-chip trace buffer
  - For more information, see *Processor Trace* section of the Device TRM.

- System instrumentation (trace)
  - Supported by a CTools System Trace Module (CT_STM), implementing MIPI System Trace Protocol (STP) (rev 2.0)
  - Real-time software trace
    - MPU software instrumentation through CoreSight STM (CS_STM) (STP2.0)
    - System-on-chip (SoC) software instrumentation through CT_STM (STP2.0)
  - OCP watchpoint (OCP_WP_NOC)
    - OCP target traffic monitoring: OCP_WP_NOC can be configured to generate a trigger upon watchpoint match (that is, when target transaction attributes match the user-defined attributes).
    - SoC events trace
    - DMA transfer profiling
  - Statistics collector (performance probes)
    - Computes traffic statistics within a user-defined window and periodically reports to the user through the CT_STM interface
    - Embedded in the L3_MAIN interconnect
    - 10 instances:
      - 1 instance dedicated to target (SDRAM) load monitoring
      - 9 instances dedicated to master latency monitoring
  - Power-management events profiling (PM instrumentation [PMI])
    - Monitoring major power-management events. The PM state changes are handled as generic events and encapsulated in STP messages.
  - Clock-management events profiling (CM instrumentation [CMI])
    - Monitoring major clock management events. The CM state changes are handled as generic events and encapsulated in STP messages.
    - Two instances, one per CM
      - CM1 Instrumentation (CMI1) module mapped in the PD_CORE_AON power domain
      - CM2 Instrumentation (CMI2) module mapped in the PD_CORE power domain
  - For more information, see *System Instrumentation* section of the Device TRM.
- Performance monitoring
  - Supported by subsystem counter timer module (SCTM) for IPU
  - Supported by performance monitoring unit (PMU) for MPU subsystem

For more information, see chapter *On-Chip Debug Support* of the device TRM.

7 Applications, Implementation, and Layout

NOTE

Information in the following Applications section is not part of the TI component specification, and TI does not warrant its accuracy or completeness. TI's customers are responsible for determining suitability of components for their purposes. Customers should validate and test their design implementation to confirm system functionality.

7.1 Introduction

This chapter is intended to communicate, guide and illustrate a PCB design strategy resulting in a PCB that can support TI's latest Application Processor. This Processor is a high-performance processor designed for automotive Infotainment based on enhanced OMAP™ architecture integrated on a 28-nm CMOS process technology.

These guidelines first focus on designing a robust Power Delivery Network (PDN) which is essential to achieve the desirable high performance processing available on Device. The general principles and step-by-step approach for implementing good power integrity (PI) with specific requirements will be described for the key Device power domains.

TI strongly believes that simulating a PCB's proposed PDN is required for first pass PCB design success. Key Device processor high-current power domains need to be evaluated for Power Rail IR Drop, Decoupling Capacitor Loop-Inductance and Power Rail Target Impedance. Only then can a PCB's PDN performance be truly accessed by comparing these model PI parameters vs. TI's recommended values. Ultimately for any high-volume product, TI recommends conducting a "Processor PDN Validation" test on prototype PCBs across processor "split lots" to verify PDN robustness meets desired performance goals for each customer's worst-case scenario. Please contact your TI representative to receive guidance on PDN PI modeling and validation testing.

Likewise, the methodology and requirements needed to route Device high-speed, differential interfaces (i.e. USB2.0, USB3.0, HDMI, PCI), single-ended interfaces (i.e. DDR3, QSPI) and general purpose interfaces using LVCMOS drivers that meet timing requirements while minimizing signal integrity (SI) distortions on the PCB's signaling traces. Signal trace lengths and flight times are aligned with FR-4 standard specification for PCBs.

Several different PCB layout stack-up examples have been presented to illustrate a typical number of layers, signal assignments and controlled impedance requirements. Different Device interface signals demand more or less complexity for routing and controlled impedance stack-ups. Optimizing the PCB's PDN stack-up needs with all of these different types of signal interfaces will ultimately determine the final layer count and layer assignments in each customer's PCB design.

This guideline must be used as a supplement in complement to TI's Application Processor, Power Management IC (PMIC) and Audio Companion components along with other TI component technical documentation (i.e. Technical Reference Manual, Data Manual, Data Sheets, Silicon Errata, Pin-Out Spreadsheet, Application Notes, etc.).

NOTE

Notwithstanding any provision to the contrary, TI makes no warranty expressed, implied, or statutory, including any implied warranty of merchantability of fitness for a specific purpose, for customer boards. The data described in this appendix are intended as guidelines only.

NOTE

These PCB guidelines are in a draft maturity and consequently, are subject to change depending on design verification testing conducted during IC development and validation.

7.1.1 Initial Requirements and Guidelines

Unless otherwise specified, the characteristic impedance for single-ended interfaces is recommended to be between 35 Ω and 65 Ω to minimize the overshoot or undershoot on far-end loads.

Characteristic impedance for differential interfaces must be routed as differential traces on the same layer. The trace width and spacing must be chosen to yield the recommended differential impedance. For more information see Section 7.5.1.

The PDN must be optimized for low trace resistance and low trace inductance for all high-current power nets from PMIC to the device.

An external interface using a connector must be protected following the IEC61000-4-2 level 4 system ESD.

7.2 Power Optimizations

This section describes the necessary steps for designing a robust Power Distribution Network (PDN):

- **Section 7.2.1, Step 1:** PCB Stack-up
- **Section 7.2.2, Step 2:** Physical Placement
- **Section 7.2.3, Step 3:** Static Analysis
- **Section 7.2.4, Step 4:** Frequency Analysis

7.2.1 Step 1: PCB Stack-up

The PCB stack-up (layer assignment) is an important factor in determining the optimal performance of the power distribution system. An optimized PCB stack-up for higher power integrity performance can be achieved by following these recommendations:

- Power and ground plane pairs must be closely coupled together. The capacitance formed between the planes can decouple the power supply at high frequencies. Whenever possible, the power and ground planes must be solid to provide continuous return path for return current.
- Use a thin dielectric between the power and ground plane pair. Capacitance is inversely proportional to the separation of the plane pair. Minimizing the separation distance (the dielectric thickness) maximizes the capacitance.
- Optimize the power and ground plane pair carrying high current supplies to key component power domains as close as possible to the same surface where these components are placed (see Figure 7-1). This will help to minimize “loop inductance” encountered between supply decoupling capacitors and component supply inputs and between power and ground plane pairs.

NOTE

1-2oz Cu weight for power / ground plane is preferred to enable better PCB heat spreading, helping to reduce Processor junction temperatures. In addition, it is preferable to have the power / ground planes be adjacent to the PCB surface on which the Processor is mounted.

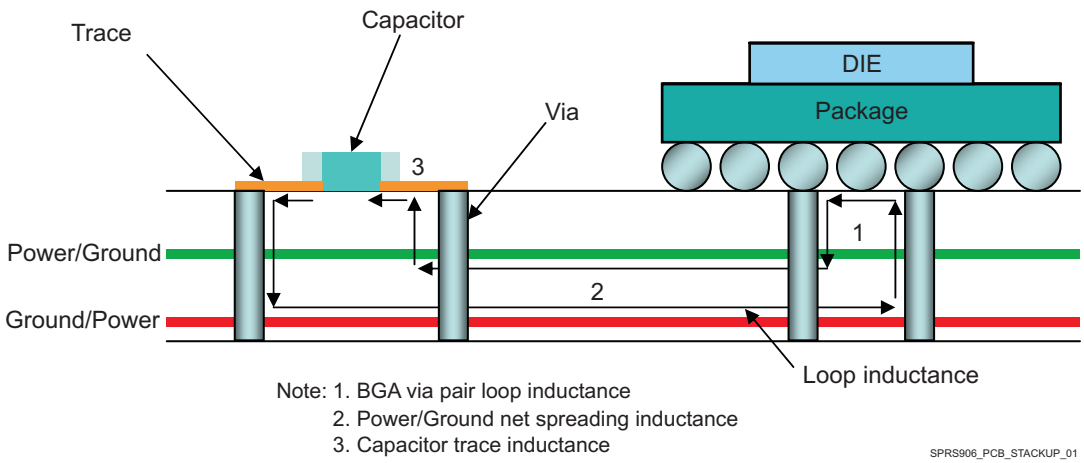


Figure 7-1. Minimize Loop Inductance With Proper Layer Assignment

The placement of power and ground planes in the PCB stackup (determined by layer assignment) has a significant impact on the parasitic inductances of power current path as shown in Figure 7-1. For this reason, it is recommended to consider layer order in the early stages of the PCB PDN design cycle, putting high-priority supplies in the top half of the stackup (assuming high load and priority components are mounted on the top-side of PCB) and low-priority supplies in the bottom half of the stackup as shown in the examples below (vias have parasitic inductances which impact the bottom layers more, so it is advised to put the sensitive and high-priority power supplies on the top/same layers).

7.2.2 Step 2: Physical Placement

A critical step in designing an optimized PDN is that proper care must be taken to making sure that the initial floor planning of the PCB layout is done with good power integrity design guidelines in mind. The following points are important for optimizing a PCB's PDN:

- Minimizing the physical distance between power sources and key high load components is the first step toward optimization. Placing source and load components on the same side of the PCB is desirable. This will minimize via inductance impact for high current loads and steps
- External trace routing between components must be as wide as possible. The wider the traces, the lower the DC resistance and consequently the lower the static IR drop.
- Whenever possible for the internal layers (routing and plane), wide traces and copper area fills are preferred for PDN layout. The routing of power nets in plane provide for more interplane capacitance and improved high frequency performance of the PDN.
- Whenever possible, use a via to component pin/pad ratio of 1:1 or better (i.e. especially decoupling capacitors, power inductors and current sensing resistors). Do not share vias among multiple capacitors for connecting power supply and ground planes.
- Placement of vias must be as close as possible or even within a component's solder pad if the PCB technology you are using provides this capability.
- To avoid any "ampacity" issue – maximum current-carrying capacity of each transitional via should be evaluated to determine the appropriate number of vias required to connect components.

  Adding vias to bring the "via-to-pad" ratio to 1:1 will improve PDN performance.
- For noise sensitive power supplies (i.e. Phase Lock-Loops, analog signals like audio and video), a Gnd shield can be used to isolate coplanar supplies that may have high step currents or high frequency switching transitions from coupling into low-noise supplies.

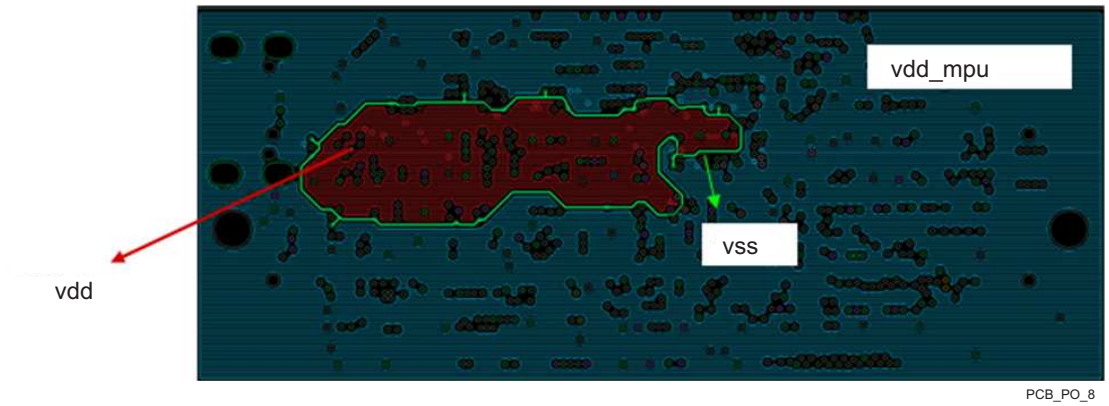


Figure 7-2. Coplanar Shielding of Power Net Using Ground Guard-band

7.2.3 Step 3: Static Analysis

Delivering reliable power to circuits is always of critical importance because voltage drops (also known as IR drops) can happen at every level within an electronic system, on-chip, within a package, and across the board. Robust system performance can only be ensured by understanding how the system elements will perform under typical stressful Use Cases. Therefore, it is a good practice to perform a Static or DC Analysis.

Static or DC analysis and design methodology results in a PDN design that minimizes voltage or IR drops across power and ground planes, traces and vias. This ensures the application processor's internal transistors will be operating within their specified voltage ranges for proper functionality. The amount of IR drop that will be encounter is based upon amount power drawn for a desired Use Case and PCB trace (widths, geometry and number of parallel traces) and via (size, type and number) characteristics.

Components that are distant from their power source are particularly susceptible to IR drop. Designs that rely on battery power must minimize voltage drops to avoid unacceptable power loss that can negatively impact system performance. Early assessments a PDN's static (DC) performance helps to determine basic power distribution parameters such as best system input power point, optimal PCB layer stackup, and copper area needed for load currents.

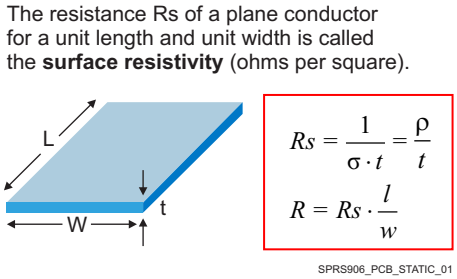


Figure 7-3. Depiction of Sheet Resistivity and Resistance

Ohm's Law (V = I × R) relates conduction current to voltage drop. At DC, the relation coefficient is a constant and represents the resistance of the conductor. Even current carrying conductors will dissipate power at high currents even though their resistance may be very small. Both voltage drop and power dissipation are proportional to the resistance of the conductor.

Figure 7-4 shows a PCB-level static IR drop budget defined between the power management device (PMIC) pins and the application processor's balls when the PMIC is supplying power.

- It is highly recommended to physically place the PMIC as close as possible to the processor and on the same side. The orientation of the PMIC vs. processor should be aligned to minimize distance for the highest current rail.

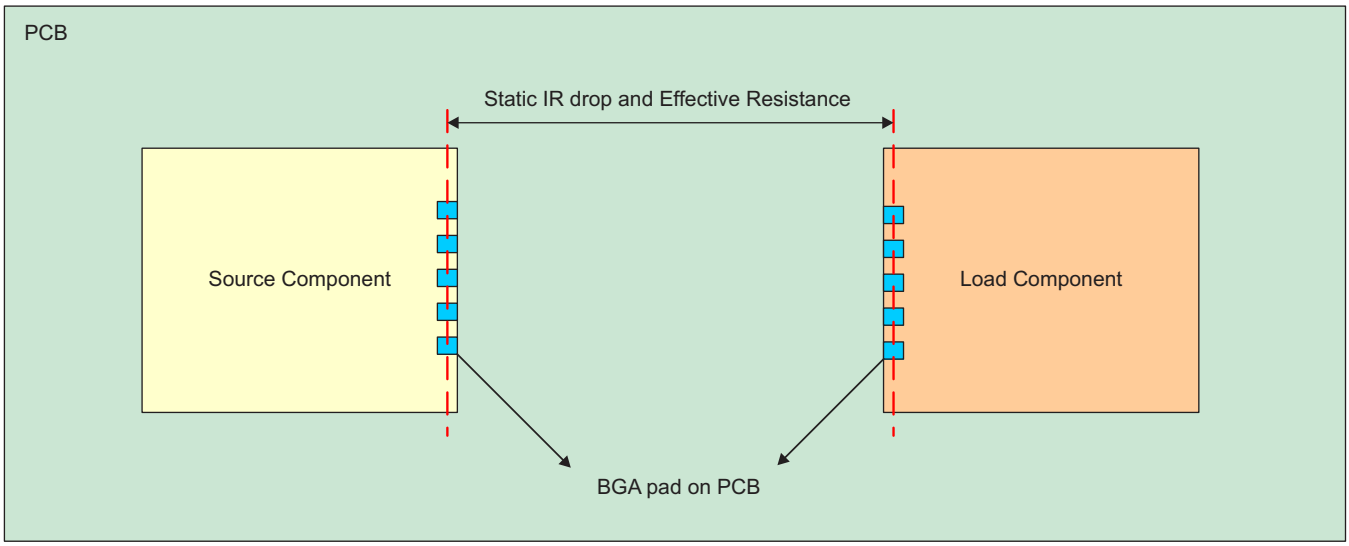


Figure 7-4. Static IR Drop Budget for PCB Only

The system-level IR drop budget is made up of three portions: on-chip, package, and PCB board. Static IR or DC analysis/design methodology consists of designing the PDN such that the voltage drop (under DC operating conditions) across power and ground pads of the transistors of the application processor device is within a specified value of the nominal voltage for proper functionality of the device.

A PCB system-level voltage drop budget for proper device functionality is typically 1.5% of nominal voltage. For a 1.35-V supply, this would be ≤20 mV.

To accurately analyze PCB static IR drop, the actual geometry of the PDN must be modeled properly and simulated to accurately characterize long distribution paths, copper weight impacts, electro-migration violations of current-carrying vias, and “Swiss-cheese” effects via placement has on power rails. It is recommended to perform the following analyses:

- Lumped resistance/IR drop analysis
- Distributed resistance/IR drop analysis

NOTE

The PMIC companion device supporting this processor has been designed with voltage sensing feedback loop capabilities that enable a remote sense of the SMPS output voltage at the point of use.

The NOTE above means the SMPS feedback signals and returns must be routed across PCB and connected to the Device input power ball for which a particular SMPS is supplying power. This feedback loop provides compensation for some of the voltage drop encountered across the PDN within limits. As such, the effective resistance of the PDN within this loop should be determined in order to optimize voltage compensation loop performance. The resistance of two PDN segments are of interest: one from the power inductor/bulk power filtering capacitor node to the Processor’s input power and second is the entire PDN route from SMPS output pin/ball to the Processor input power.

In the following sections each methodology is described in detail and an example has been provided of analysis flow that can be used by the PCB designer to validate compliance to the requirements on their PCB PDN design.

7.2.3.1 PDN Resistance and IR Drop

Lumped methodology consists of grouping all of the power pins on both the PMIC (voltage source) and processor (current sink) devices. Then the PMIC source is set to an expected Use Case voltage level and the processor load has its Use Case current sink value set as well. Now the lumped/effective resistance for the power rail trace/plane routes can be determine based upon the actual layout's power rail etch wide, shape, length, via count and placement Figure 7-5 illustrates the pin-grouping/lumped concept.

The lumped methodology consists of importing the PCB layout database (from Cadence Allegro tool or any other layout design tool) into the static IR drop modeling and simulation tool of preference for the PCB designer. This is followed by applying the correct PCB stack-up information (thickness, material properties) of the PCB dielectric and metallization layers. The material properties of dielectric consist of permittivity (Dk) and loss tangent (Df).

For the conductor layers, the correct conductivity needs to be programmed into the simulation tool. This is followed by pin-grouping of the power and ground nets, and applying appropriate voltage/current sources. The current and voltage information can be obtained from the power and voltage specifications of the device under different operating conditions / Use Cases.

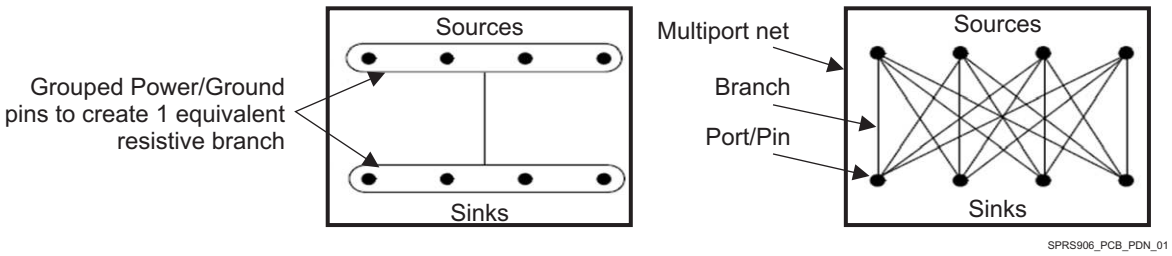


Figure 7-5. Pin-grouping concept: Lumped and Distributed Methodologies

7.2.4 Step 4: Frequency Analysis

Delivering low noise voltage sources are very important to allowing a system to operate at the lowest possible Operational Performance Point (OPP) for any one Use Case. An OPP is a combination of the supply voltage level and clocking rate for key internal processor domains. A SCH and PCB designed to provide low noise voltage supplies will then enable the processor to enter optimal OPPs for each Use Case that in turn will minimize power dissipation and junction temperatures on-die. Therefore, it is a good engineering practice to perform a Frequency Analysis over the key power domains.

Frequency analysis and design methodology results in a PDN design that minimizes transient noise voltages at the processor's input power balls. This allows the processor's internal transistors to operate near the minimum specified operating supply voltage levels. To accomplish this one must evaluate how a voltage supply will change due to impedance variations over frequency. This analysis will focus on the decoupling capacitor network (VDD_xxx and VSS/Gnd rails) at the load. Sufficient capacitance with a distribution of self-resonant points will provide for an overall lower impedance vs frequency response for each power domain.

Decoupling components that are distant from their load's input power are susceptible to encountering spreading loop inductance from the PCB design. Early analysis of each key power domain's frequency response helps to determine basic decoupling capacitor placement, optimal footprint, layer assignment, and types needed for minimizing supply voltage noise/fluctuations due to switching and load current transients.

NOTE

Evaluation of loop inductance values for decoupling capacitors placed ~300mils closer to the load's input power balls has shown an 18% reduction in loop inductance due to reduced distance.

- Decoupling capacitors must be carefully placed in order to minimize loop inductance impact on supply voltage transients. A real capacitor has characteristics not only of capacitance but also inductance and resistance.

  Figure 7-6 shows the parasitic model of a real capacitor. A real capacitor must be treated as an RLC circuit with effective series resistance (ESR) and effective series inductance (ESL).

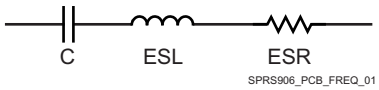


Figure 7-6. Characteristics of a Real Capacitor With ESL and ESR

The magnitude of the impedance of this series model is given as:

$$|Z| = \sqrt{ESR^2 + \left(\omega ESL - \frac{1}{\omega C}\right)^2}$$

$$\text{where :} \quad \omega = 2\pi f$$

SPRS906_PCB_FREQ_02

Figure 7-7. Series Model Impedance Equation

Figure 7-8 shows the resonant frequency response of a typical capacitor with a self-resonant frequency of 55 MHz. The impedance of the capacitor is a combination of its series resistance and reactive capacitance and inductance as shown in the equation above.

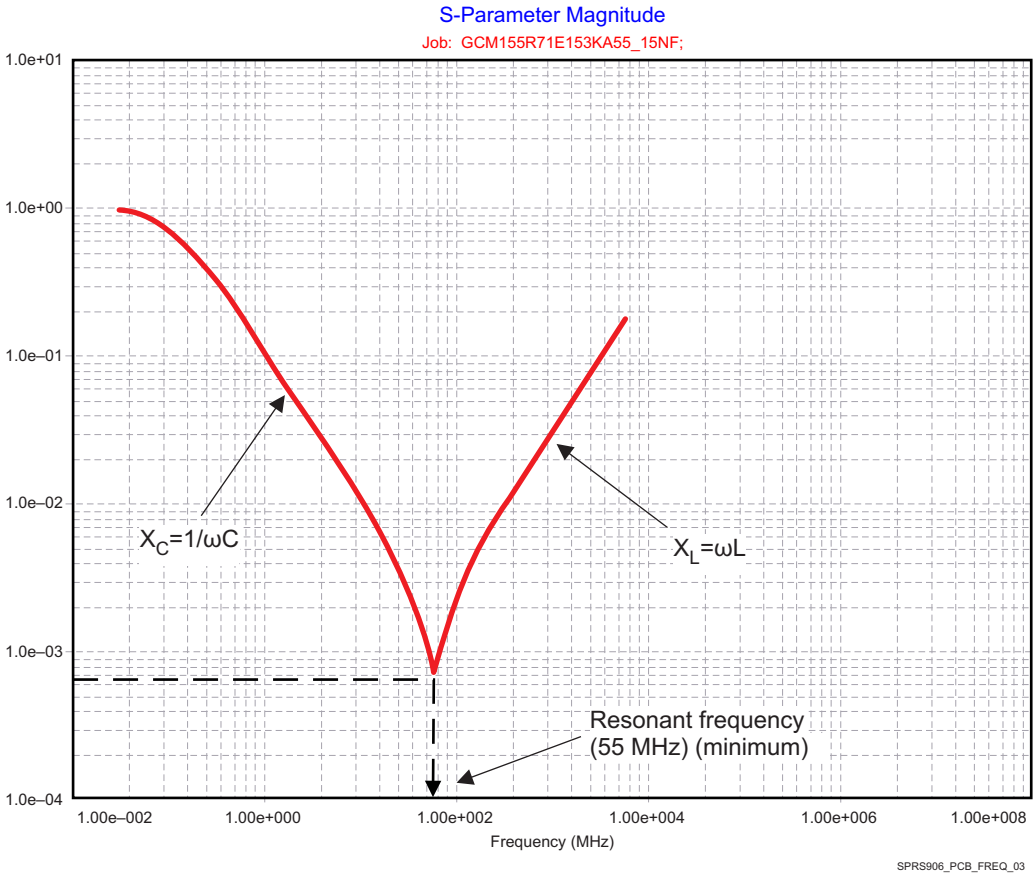


Figure 7-8. Typical Impedance Profile of a Capacitor

Because a capacitor has series inductance and resistance that impacts its effectiveness, it is important that the following recommendations are adopted in placing capacitors on the PDN.

Wherever possible, mount the capacitor with the geometry that minimizes the mounting inductance and resistance. This was shown earlier in Figure 7-1. The capacitor mounting inductance and resistance values include the inductance and resistance of the pads, trace, and vias. Whenever possible, use footprints that have the lowest inductance configuration as shown in Figure 7-9

The length of a trace used to connect a capacitor has a big impact on parasitic inductance and resistance of the mounting. This trace must be as short and as wide as possible. wherever possible, minimize distance to supply and Gnd vias by locating vias nearby or within the capacitor's solder pad landing. Further improvements can be made to the mounting by placing vias to the side of capacitor lands or doubling the number of vias as shown in Figure 7-9. If the PCB manufacturing processes allow it and if cost-effective, via-in-pad (VIP) geometries are strongly recommended.

In addition to mounting inductance and resistance associated with placing a capacitor on the PCB, the effectiveness of a decoupling capacitor also depends on the spreading inductance and resistance that the capacitor sees with respect to the load. The spreading inductance and resistance is strongly dependent on the layer assignment in the PCB stack-up. Therefore, try to minimize X, Y and Z dimensions where the Z is due to PCB thickness (as shown in Figure 7-9).

From left (highest inductance) to right (lowest inductance) the capacitor footprint types shown in Figure 7-9 are known as:

- 2-via, Skinny End Exit (2vSEE)
- 2-via, Wide End Exit (2vWEE)
- 2-via, Wide Side Exit (2vWSE)
- 4-via, Wide Side Exit (4vWSE)
- 2-via, In-Pad (2vIP)

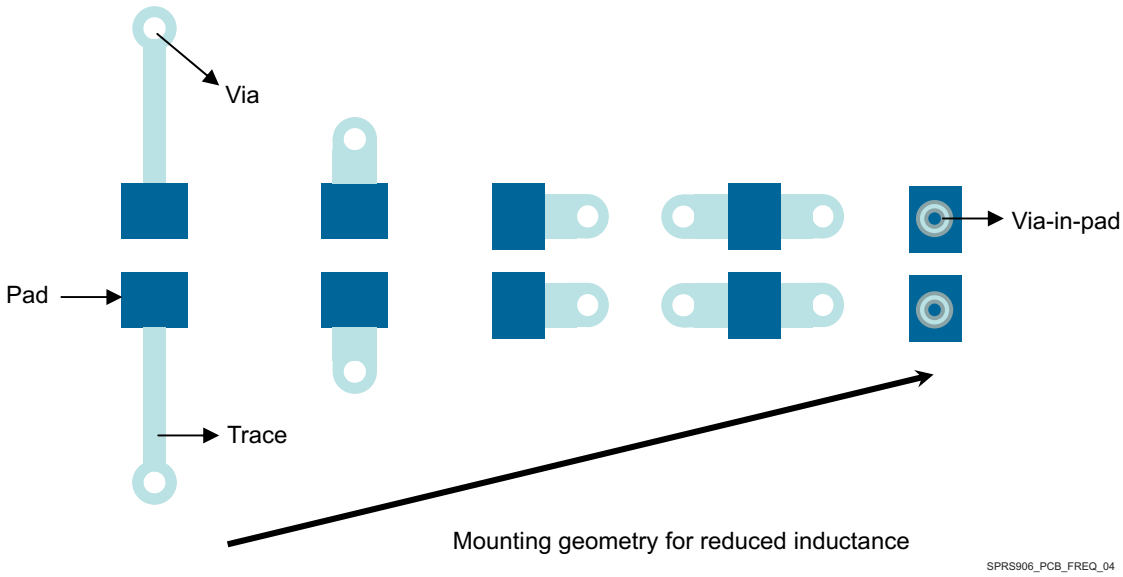


Figure 7-9. Capacitor Placement Geometry for Improved Mounting Inductance

NOTE

Evaluation of loop inductance values for decoupling capacitor footprints 2vSEE (worst case) vs 4vWSE (2nd best) has shown a 30% reduction in inductance when 4vWSE footprint was used in place of 2vSEE.

Decoupling Capacitor (Dcap) Strategy:

1. Use lowest inductance footprint and trace connection scheme possible for given PCB technology and layout area in order to minimize Dcap loop inductance to power pin as much as possible (see Figure 7-9).
2. Place Dcaps on “same-side” as component within their power plane outline to minimize “decoupling loop inductance”. Target distance to power pin should be less than ~500mils depending upon PCB layout characteristics (plane's layer assignment and solid nature). Use PI modeling CAD tool to verify minimum inductance for top vs bottom-side placement.
3. Place Dcaps on “opposite-side” as component within their power plane outline if “same-side” is not feasible or if distance to power pin is greater than ~500mils for top-side location. Use PI modeling CAD tool to verify minimum inductance for top vs bottom-side placement.
4. Use minimum 10mil trace width for all voltage and gnd planes connections (i.e. Dcap pads, component power pins, etc.).
5. Place all voltage and gnd plane vias “as close as possible” to point of use (i.e. Dcap pads, component power pins, etc.).
6. Use a “Power/Gnd pad/pin to via” ratio of 1:1 whenever possible. Do not exceed 2:1 ratio for small number of vias within restricted PCB areas (i.e. underneath BGA components).

Frequency analysis for the CORE power domain has yielded the vdd Impedance vs Frequency response shown in Section 7.3.8.2, vdd Example Analysis. As the example shows the overall CORE PDN R_{eff} meets the maximum recommended PDN resistance of 10mΩ.

7.2.5 System ESD Generic Guidelines

7.2.5.1 System ESD Generic PCB Guideline

Protection devices must be placed close to the ESD source which means close to the connector. This allows the device to subtract the energy associated with an ESD strike before it reaches the internal circuitry of the application board.

To help minimize the residual voltage pulse that will be built-up at the protection device due to its nonzero turn-on impedance, it is mandatory to route the ESD device with minimum stub length so that the low-resistive, low-inductive path from the signal to the ground is granted and not increasing the impedance between signal and ground.

For ESD protection array being railed to a power supply when no decoupling capacitor is available in close vicinity, consider using a decoupling capacitor (≥ 0.1 µF) tight to the VCC pin of the ESD protection. A positive strike will be partially diverted to this capacitance resulting in a lower residual voltage pulse.

Ensure that there is sufficient metallization for the supply of signals at the interconnect side (VCC and GND in Figure 7-10) from connector to external protection because the interconnect may see between 15-A to 30-A current in a short period of time during the ESD event.

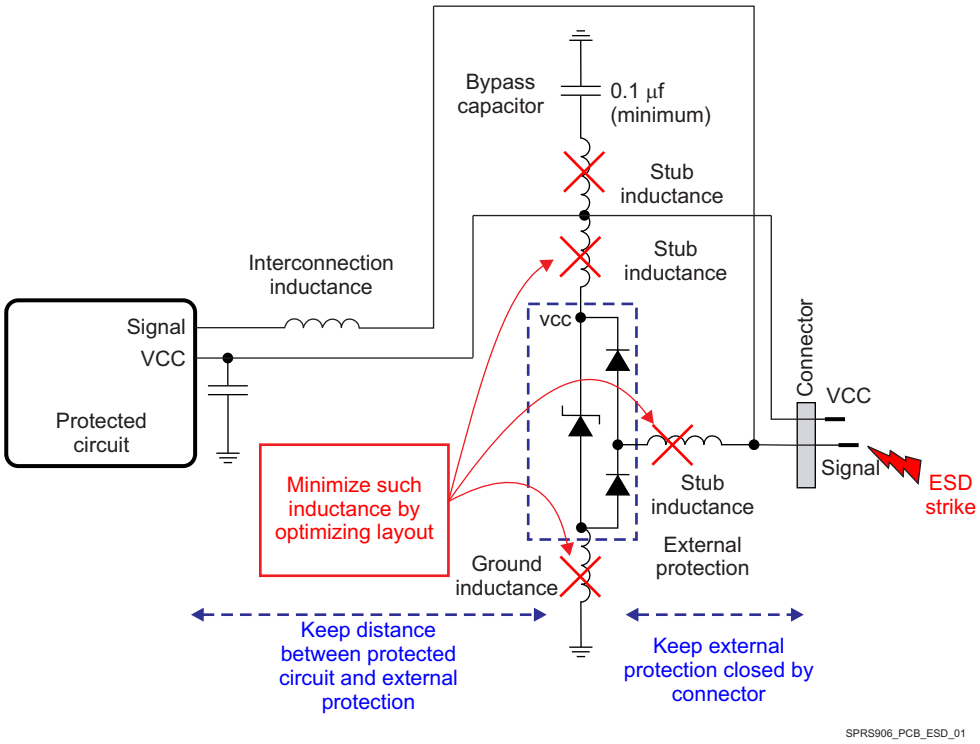


Figure 7-10. Placement Recommendation for an ESD External Protection

NOTE

To ensure normal behavior of the ESD protection (unwanted leakage), it is better to ground the ESD protection to the board ground rather than any local ground (example isolated shield or audio ground).

7.2.5.2 Miscellaneous EMC Guidelines to Mitigate ESD Immunity

- Avoid running critical signal traces (clocks, resets, interrupts, control signals, and so forth) near PCB edges.
- Add high frequency filtering: Decoupling capacitors close to the receivers rather than close to the drivers to minimize ESD coupling.
- Put a ground (guard) ring around the entire periphery of the PCB to act as a lightning rod.
- Connect the guard ring to the PCB ground plane to provide a low impedance path for ESD-coupled current on the ring.
- Fill unused portions of the PCB with ground plane.
- Minimize circuit loops between power and ground by using multilayer PCB with dedicated power and ground planes.
- Shield long line length (strip lines) to minimize radiated ESD.
- Avoid running traces over split ground planes. It is better to use a bridge connecting the two planes in one area.

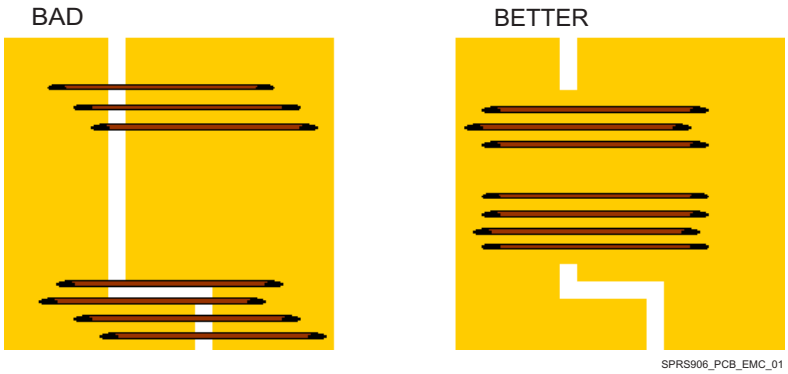


Figure 7-11. Trace Examples

- Always route signal traces and their associated ground returns as close to one another as possible to minimize the loop area enclosed by current flow:
  - At high frequencies current follows the path of least inductance.
  - At low frequencies current flows through the path of least resistance.

7.2.5.3 ESD Protection System Design Consideration

ESD protection system design consideration is covered in Section 7.5.2.2 of this document. The following are additional considerations for ESD protection in a system.

- Metallic shielding for both ESD and EMI
- Chassis GND isolation from the board GND
- Air gap designed on board to absorb ESD energy
- Clamping diodes to absorb ESD energy
- Capacitors to divert ESD energy
- The use of external ESD components on the DP/DM lines may affect signal quality and are not recommended.

7.2.6 EMI / EMC Issues Prevention

All high-speed digital integrated circuits can be sources of unwanted radiation, which can affect nearby sensitive circuitry and cause the final product to have radiated emissions levels above the limits allowed by the EMC regulations if some preventative steps are not taken.

Likewise, analog and digital circuits can be susceptible to interference from the outside world and picked up by the circuitry interconnections.

To minimize the potential for EMI/EMC issues, the following guidelines are recommended to be followed.

7.2.6.1 Signal Bandwidth

To evaluate the frequency of a digital signal, an estimated rule of thumb is to consider its bandwidth f_{BW} with respect to its rise time, t_R:

$$f_{BW} \approx 0.35 / t_R$$

This frequency actually corresponds to the break point in the signal spectrum, where the harmonics start to decay at 40 dB per decade instead of 20 dB per decade.

7.2.6.2 Signal Routing

7.2.6.2.1 Signal Routing—Sensitive Signals and Shielding

Keep radio frequency (RF) sensitive circuitry (like GPS receivers, GSM/WCDMA, Bluetooth/WLAN transceivers, frequency modulation (FM) radio) away from high-speed ICs (the device, power and audio manager, chargers, memories, and so forth) and ideally on the opposite side of the PCB. For improved protection it is recommended to place these emission sources in a shield can. If the shield can have a removable lid (two-piece shield), ensure there is low contact impedance between the fence and the lid. Leave some space between the lid and the components under it to limit the high-frequency currents induced in the lid. Limit the shield size to put any potential shield resonances above the frequencies of interest; see Figure 7-8, *Typical Impedance Profile of a Capacitor*.

7.2.6.2.2 Signal Routing—Outer Layer Routing

In case there is a need to use the outer layers for routing outside of shielded areas, it is recommended to route only static signals and ensure that these static signals do not carry any high-frequency components (due to parasitic coupling with other signals). In case of long traces, make provision for a bypass capacitor near the signal source.

Routing of high-frequency clock signals on outer layers, even for a short distance, is discouraged, because their emissions energy is concentrated at the discrete harmonics and can become significant even with poor radiators.

Coplanar shielding of traces on outer layers (placing ground near the sides of a track along its length) is effective only if the distance between the trace sides and the ground is smaller that the trace height above the ground reference plane. For modern multilayer PCBs this is often not possible, so coplanar shielding will not be effective. Do not route high-frequency traces near the periphery of the PCB, as the lack of a ground reference near the trace edges can increase EMI: see Section 7.2.6.3, *Ground Guidelines*.

7.2.6.3 Ground Guidelines

7.2.6.3.1 PCB Outer Layers

Ideally the areas on the top and bottom layers of the PCB that are not enclosed by a shield should be filled with ground after the routing is completed and connected with an adequate number of vias to the ground on the inner ground planes.

7.2.6.3.2 Metallic Frames

Ensure that all metallic parts are well connected to the PCB ground (like LCD screens metallic frames, antennas reference planes, connector cages, flex cables grounds, and so forth). If using flex PCB ribbon cables to bring high-frequency signals off the PCB, ensure they are adequately shielded (coaxial cables or flex ribbons with a solid reference ground).

7.2.6.3.3 Connectors

For high-frequency signals going to connectors choose a fully shielded connector, if possible (for example, SD card connectors). For signals going to external connectors or which are routed over long distances, it is recommended to reduce their bandwidth by using low-pass filters (resistor, capacitor (RC) combinations or lossy ferrite inductors). These filters will help to prevent emissions from the board and can also improve the immunity from external disturbances.

7.2.6.3.4 Guard Ring on PCB Edges

The major advantage of a multilayer PCB with ground-plane is the ground return path below each and every signal or power trace.

As shown in Figure 7-12 the field lines of the signal return to PCB ground as long as an infinite ground is available.

Traces near the PCB-edges do not have this infinite ground and therefore may radiate more than the others. Thus, signals (clocks) or power traces (core power) identified to be critical must not be routed in the vicinity of PCB edges, or, if not avoidable, must be accompanied by a guard ring on the PCB edge.

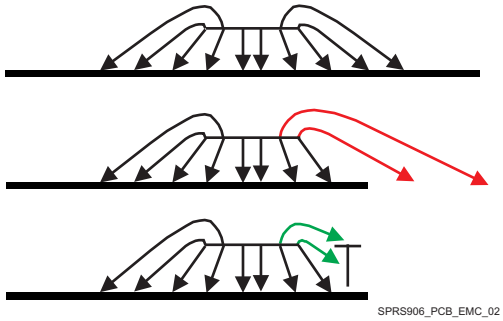


Figure 7-12. Field Lines of a Signal Above Ground

Figure 7-13. Guard Ring Routing

The intention of the guard ring is that HF-energy, that otherwise would have been emitted from the PCB edge, is reflected back into the board where it partially will be absorbed. For this purpose ground traces on the borders of all layers (including power layer) must be applied as shown in Figure 7-13.

As these traces must have the same (HF–) potential as the ground plane they must be connected to the ground plane at least every 10 mm.

7.2.6.3.5 Analog and Digital Ground

For the optimum solution, the AGND and the DGND planes must be connected together at the power supply source in a same point. This ensures that both planes are at the same potential, while the transfer of noise from the digital to the analog domain is minimized.

7.3 Core Power Domains

This section provides boundary conditions and theoretical background to be applied as a guide for optimizing a PCB design. The decoupling capacitor and PDN characteristics tables shown below give recommended capacitors and PCB parameters to be followed for schematic and PCB designs. Board designs that meet the static and dynamic PDN characteristics shown in tables below will be aligned to the expected PDN performance needed to optimize SoC performance.

7.3.1 General Constraints and Theory

- Max PCB static/DC voltage drop (IRd) budget of **1.5% of supply voltage** when using TI recommended PMICs **without remote sensing** as measured from PMIC's power inductor and filter capacitor node to Processor input including any ground return losses.
- Max PCB static/DC voltage drop (IRd) budget can be relaxed to **7.5% of supply voltage** when using PMICs **with remote sensing at the load** as measured from PMIC's power inductor and filter capacitor node to Device's supply input including any ground return losses.

- PMIC component DM and guidelines should be referenced for the following:
  - Routing remote feedback sensing to optimize per each SMPS's implementation
  - Selecting power filtering capacitor values and PCB placement.
- Max Effective Resistance (Reff) budget can range from **4 – 100mΩ** for key Device power rails not including ground returns depending upon maximum load currents and maximum DC voltage drop budget (as discussed above).
- Max Device supply input voltage difference budget of **5mV** under max current loading shall be maintained across all balls connected to a common power rail. This represents any voltage difference that may exist between a remote sense point to any power input.
- Max PCB Loop Inductance (LL) budget between Device's power inputs and local bulk and high frequency decoupling capacitors including ground returns should range from **0.4 – 2.5nH depending upon maximum transient load currents.**
- Max PCB dynamic/AC peak-to-peak transient noise voltage budgets between PMIC and Device including ground returns are as follows:
  - **+/-3% of nominal supply voltage** for frequencies below the PMIC bandwidth (typ Fpmic ~ 200kHz)
  - **+/-5% of nominal supply voltage** for frequencies between Fpmic to Fpcb (typ 20 – 100MHz)
- Max PCB Impedance (Z) vs Frequency (F) budget between Device's power inputs and PMIC's output power filter node including ground return is determined by applying the Frequency Domain Target Impedance Method to determine the PCB's maximum frequency of interest (Fpcb). Ideally a properly designed and decoupled PDN will exhibit smoothly increasing Z vs. F curve. There are 2 general regions of interest as can be seen in Figure 7-14.
  - 1st area is from DC (0Hz) up to Fpmic (typ a few 100 kHz) where a PMIC's transient response characteristic (i.e. Switching Freq, Compensation Loop BW) dominate. A PDN's Z is typically very low due to power filtering & bulk capacitor values when PDN has very low trace resistance (i.e. good Reff performance). The goal is to maintain a smoothly increasing Z that is less than Zt1 over this low frequency range. This will ensure that a max transient current event will not cause a voltage drop more than the PMIC's current step response can support (typ 3%).
  - 2nd area is from Fpmic up to Fpcb (typ 20-100MHz) where a PCB's inherent characteristics (i.e. parasitic capacitance, planar spreading inductances) dominate. A PDN's Z will naturally increase with frequency. At frequencies between Fpmic up to Fpcb, the goal is to maintain a smoothly increasing Z to be less than Zt2. This will ensue that the high frequency content of a max transient current event will not cause a voltage drop to be more than 5% of the min supply voltage.

$$Z_T = \frac{\text{Max Voltage Rail Drop}^{\text{Note1}}}{\text{Max Transient Current}^{\text{Note2}}}$$

$$Z_{T1} = \frac{\text{(Min Voltage) x (PMIC's Step Responce)}}{\text{(~50\% of Max DC Current)}} = \frac{\text{Vmin x 3\%(typ)}}{\text{Imax x ~50\%}}$$

$$Z_{T2} = \frac{\text{(Min Voltage) x (High-Freq Transient Noise)}}{\text{(~50\% of Max DC Current)}} = \frac{\text{Vmin x 5\%(typ)}}{\text{Imax x ~50\%}}$$

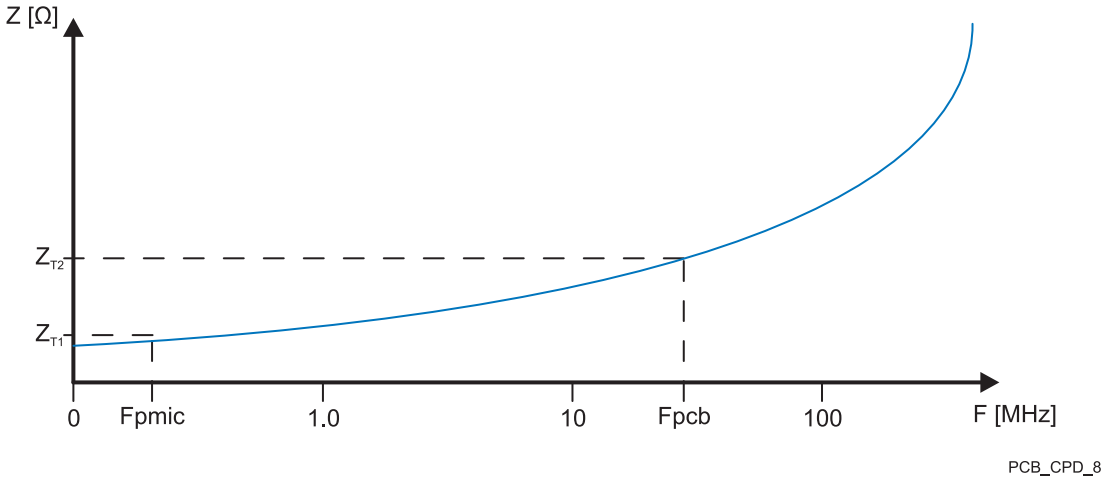


Figure 7-14. PDN's Target impedance

1.Voltage Rail Drop includes regulation accuracy, voltage distribution drops, and all dynamic events such as transient noise, AC ripple, voltage dips etc.

2.Typical max transient current is defined as 50% of max current draw possible.

7.3.2 Voltage Decoupling

Recommended power supply decoupling capacitors main characteristics for commercial products whose ambient temperature is not to exceed +85C are shown in table below:

Table 7-1. Commercial Applications Recommended Decoupling Capacitors Characteristics(1)(2)(3)

Value	Voltage [V]	Package	Stability	Dielectric	Capacitance Tolerance	Temp Range [°C]	Temp Sensitivity [%]	REFERENCE
22µF	6,3	0603	Class 2	X5R	- / + 20%	-55 to + 85	- / + 15	GRM188R60J226MEA0L
10µF	4,0	0402	Class 2	X5R	- / + 20%	-55 to + 85	- / + 15	GRM155R60G106ME44
4.7µF	6,3	0402	Class 2	X5R	- / + 20%	-55 to + 85	- / + 15	GRM155R60J475ME95
2.2µF	6,3	0402	Class 2	X5R	- / + 20%	-55 to + 85	- / + 15	GRM155R60J225ME95
1µF	6,3	0201	Class 2	X5R	- / + 20%	-55 to + 85	- / + 15	GRM033R60J105MEA2

Table 7-1. Commercial Applications Recommended Decoupling Capacitors Characteristics(1)(2)(3) (continued)

Value	Voltage [V]	Package	Stability	Dielectric	Capacitance Tolerance	Temp Range [°C]	Temp Sensitivity [%]	REFERENCE
470nF	6,3	0201	Class 2	X5R	- / + 20%	-55 to + 85	- / + 15	GRM033R60G474ME90
220nF	6,3	0201	Class 2	X5R	- / + 20%	-55 to + 85	- / + 15	GRM033R60J224ME90
100nF	6,3	0201	Class 2	X5R	- / + 20%	-55 to + 85	- / + 15	GRM033R60J104ME19

(1) Minimum value for each PCB capacitor: 100 nF.

(2) Among the different capacitors, 470 nF is recommended (not required) to filter at 5-MHz to 10-MHz frequency range.

(3) In comparison with the EIA Class 1 dielectrics, Class 2 dielectric capacitors tend to have severe temperature drift, high dependence of capacitance on applied voltage, high voltage coefficient of dissipation factor, high frequency coefficient of dissipation, and problems with aging due to gradual change of crystal structure. Aging causes gradual exponential loss of capacitance and decrease of dissipation factor.

Recommended power supply decoupling capacitors main characteristics for automotive products are shown in table below:

Table 7-2. Automotive Applications Recommended Decoupling Capacitors Characteristics (1)(2)

Value	Voltage [V]	Package	AEC-Q200	Dielectric	Capacitance Tolerance	Temp Range [°C]	Temp Sensitivity [%]	REFERENCE
22µF	10	1206	Yes	X7R	- / + 10%	-55 to + 125	- / + 15	GCM31CR71A226KE02
10µF	10	0805	Yes	X7R	- / + 10%	-55 to + 125	- / + 15	GCM21BR71A106KE22
4.7µF	10	0805	Yes	X7S	- / + 10%	-55 to + 125	- / + 22	GCM21BC71A475KA73
2.2µF	6,3	0603	Yes	X7R	- / + 10%	-55 to + 125	- / + 15	GCM188R70J225KE22
1µF	10	0402	Yes	X7S	- / + 10%	-55 to + 125	- / + 22	GCM155C71A105KE38
470nF	10	0402	Yes	X7S	- / + 10%	-55 to + 125	- / + 22	GCM155C71A474KE36
220nF	25	0603	Yes	X7R	- / + 10%	-55 to + 125	- / + 15	GCM155R71A104KA55
100nF	10	0402	Yes	X7R	- / + 10%	-55 to + 125	- / + 15	GCM155R71C104MA55
100nF	6.3	0201	Yes	X7S	- / + 10%	-55 to + 125	- / + 15	GCM033C70J104K
1.0µF	10	3T-0805(3)	Yes		- / + 20%	-55 to + 125		NFM21HC105R1C3
0.47µF	10	3T-0805(3)	Yes		- / + 20%	-55 to + 125		NFM21HC474R1C3
0.22µF	10	3T-0805(3)	Yes		- / + 20%	-55 to + 125		NFM21HC224R1C3
0.1µF	10	3T-0805(3)	Yes		- / + 20%	-55 to + 125		NFM21HC104R1C3

(1) Minimum value for each PCB capacitor: 100 nF.

(2) Among the different capacitors, 470 nF is recommended (not required) to filter at 5-MHz to 10-MHz frequency range.

(3) 3T designates this as a "3-terminal, low inductance type package".

7.3.3 Static PDN Analysis

One power net parameter derived from a PCB's PDN static analysis is the Effective Resistance (Reff). This is the total PCB power net routing resistance that is the sum of all the individual power net segments used to deliver a supply voltage to the point of load and includes any series resistive elements (i.e. current sensing resistor) that may be installed between the PMIC outputs and Processor inputs.

7.3.4 Dynamic PDN Analysis

Three power net parameters derived from a PCB's PDN dynamic analysis are the Loop Inductance (LL), Impedance (Z) and PCB Frequency of Interest (Fpcb).

- LL values shown are the recommended max PCB trace inductance between a decoupling capacitor's power supply and ground reference terminals when viewed from the decoupling capacitor with a "theoretical shorted" applied across the Processor's supply inputs to ground reference.

- Z values shown are the recommended max PCB trace impedances allowed between Fpmic up to Fpcb frequency range that limits transient noise drops to no more than 5% of min supply voltage during max transient current events.
- Fpcb (Frequency of Interest) is defined to be a power rail's max frequency after which adding a reasonable number of decoupling capacitors no longer significantly reduces the power rail impedance below the desired impedance target (Zt2). This is due to the dominance of the PCB's parasitic planar spreading and internal package inductances.

Table 7-3. Recommended PDN and Decoupling Characteristics (1)(2)(3)(4)

PDN Analysis:	Static	Dynamic			Number of Recommended Decoupling Capacitors per Supply							
Supply	Max R_{eff} (5) [mΩ]	Dec. Cap. Max LL (6) [nH]	Max Impedance [mΩ]	Frequency range of Interest [MHz]	100 nF	220 nF	470 nF	1µF	2.2 µF	4.7 µF	10 µF	22 µF
vdd_dsp	22	2.5	54	≤20	6	1	1	1	1	1		1
vdd	18	2	57	≤20	6	1	1	1	1		1	
vdds_ddr1	33	2.5	200	≤100	8	3		2		2		1
cap_vbbldo_dsp	N/A	6	N/A	N/A				1				
cap_vbbldo_gpu	N/A	6	N/A	N/A				1				
cap_vbbldo_iva	N/A	6	N/A	N/A				1				
cap_vbbldo_mpu	N/A	6	N/A	N/A				1				
cap_vddram_core1	N/A	6	N/A	N/A				1				
cap_vddram_core3	N/A	6	N/A	N/A				1				
cap_vddram_core4	N/A	6	N/A	N/A				1				
cap_vddram_dsp	N/A	6	N/A	N/A				1				
cap_vddram_gpu	N/A	6	N/A	N/A				1				
cap_vddram_iva	N/A	6	N/A	N/A				1				
cap_vddram_mpu	N/A	6	N/A	N/A				1				

(1) For more information on peak-to-peak noise values, see the Recommended Operating Conditions table of the Specifications chapter.

(2) ESL must be as low as possible and must not exceed 0.5 nH.

(3) The PDN (Power Delivery Network) impedance characteristics are defined versus the device activity (that runs at different frequency) based on the Recommended Operating Conditions table of the Specifications chapter.

(4) Maximum static voltage drop allowed drives the maximum acceptable power net resistance (R_{eff}) between the PMIC or the external SMPS and the processor power balls.

(5) Maximum R_{eff} (from SMPS to Processor) allows for max supply voltage drop when both remote voltage sensing very close to processor power balls and TI recommended PMICs are used.

(6) Maximum Loop Inductance to each high-frequency (30-70MHz) decoupling capacitor.

7.3.5 Power Supply Mapping

TPS65919 or LP8733 are the Power Management ICs (PMICs) that should be used for the Device designs. TI requires use of these PMICs for the following reasons:

- TI has validated their use with the Device
- Board level margins including transient response and output accuracy are analyzed and optimized for the entire system
- Support for power sequencing requirements (refer to Section 5.10.3 Power Supply Sequences)
- Support for Adaptive Voltage Scaling (AVS) Class 0 requirements, including TI provided software
- Remote sensing at point of load with output voltage compensation allows for the maximum IR drop budget

Whenever one SMPS supplies multiple SoC voltage domains from a common power rail, the most stringent PDN guideline across the voltage domains being combined should be applied to the common power rail.

It is possible that some voltage domains on the device are unused in some systems. In such cases, to ensure device reliability, it is still required that the supply pins for the specific voltage domains are connected to some core power supply output.

These unused supplies though can be combined with any of the core supplies that are used (active) in the system. e.g. if the DSP domain is not used, it can be combined with the CORE domain, thereby having a single power supply driving the combined CORE and DSP domains.

For the combined rail, the following relaxations do apply:

- The AVS voltage of active voltage domain in the combined rail needs to be used to set the power supply
- The decoupling capacitance should be set according to the active voltage domain in the combined rail
- The PDN guideline should be set according to the active voltage domain in the combined rail

Table 7-4 illustrates the approved and validated power supply connections to the Device for the SMPS outputs of the TPS656919 PMIC.

Table 7-4. TPS65919 Power Supply Connections(1)

SMPS	Valid Combination	TPS65919 Current Limitation(2) (3)
SMPS1	VD_CORE	3.5A
SMPS2	Free (DDR Memory)	3.5A
SMPS3	VD_DSP	3A
SMPS4	VDDS18V	1.5A

(1) Power consumption is highly application-specific. Separate analysis must be performed to ensure output current ratings (average and peak) is within the limits of the PMIC for all rails of the device.

(2) Refer to the PMIC data manual for the latest TPS65919 specifications.

(3) A product's maximum ambient temperature, thermal system design & heat spreading performance could limit the maximum power dissipation below the full PMIC capacity in order to not exceed recommended SoC max Tj.

Table 7-5 illustrates the approved and validated power supply connections to the Device for the SMPS outputs of the LP8733 PMIC.

Table 7-5. LP8733 Power Supply Connections

SMPS	Valid Combination	LP8733 Current Limitation(1) (2)
SMPS1	VD_CORE	3A
SMPS2	VD_DSP	3A

(1) Refer to the LP8733 Data Manual for exact current rating limitations, including assumed VIN and other parameters. Values provided in this table are for comparison purposes.

(2) Highly application-specific. Separate analysis must be performed to ensure average and peak power is within the limits of the PMIC.

7.3.6 DPLL Voltage Requirement

The voltage input to the DPLLs has a low noise requirement. Board designs should supply these voltage inputs with a low noise LDO to ensure they are isolated from any potential digital switching noise. The TPS65919 PMIC LDOLN output is specifically designed to meet this low noise requirement.

NOTE

For more information about Input Voltage Sources, see Section 5.10.4.3 *DPLLs, DLLs Specifications*.

Table 7-6 presents the voltage inputs that supply the DPLLs.

Table 7-6. Input Voltage Power Supplies for the DPLLs

POWER SUPPLY	DPLLs
vdda_per	DPLL_PER and PER HSDIVIDER analog power supply
vdda_ddr	DPLL_DDR and DDR HSDIVIDER analog power supply
vdda_debug	DPLL_DEBUG analog power supply
vdda_core_gmac	DPLL_CORE and HSDIVIDER analog power supply
vdda_gpu	DPLL_GPU analog power supply
vdda_video	DPLL_VIDEO1 analog power supply
vdda_mpu_abe	DPLL_MPU and DPLL_ABE analog power supply
vdda_osc	not DPLL input but is required to be supplied by low noise input voltage
vdda_dsp_iva	DSP PLL and IVA PLL analog power supply

7.3.7 Loss of Input Power Event

A few key PDN design items needed to enable a controlled and compliant SoC power down sequence for a “Loss of Input Power” event are:

- “Loss of Input Power” early warning
  - TI EVM and Reference Design Study SCHs and PDNs achieve this by using the First Stage Converter’s (i.e. LM536033-Q1) Power Good status output to enable and disable the Second Stage PMIC devices (i.e. TPS65917/919, LP8733, and LP8732). If a different First Stage Converter is used, care must be taken to ensure an adequate “PG_Status” or “Vbatt_Status” signal is provided that can disable Second Stage PMIC to begin a controlled and compliant SoC power down sequence. The total elapsed time from asserting “PG_Status” low until SoC’s PMIC input voltage reaches minimum level of 2.75 V should be minimum of 1.5ms and 2ms preferred.
- Maximize discharge time of First Stage Vout (VSYS_3V3 power rail = input voltage to SoC PMIC).
  - TI EVM and Reference Design Study SCHs and PDNs achieve this by opening an in-line load switch immediately upon “PG_Status” low assertion in order to remove the SoC’s 3.3V IO load current from VSYS_3V3. This will extend the VSYS_3V3 power rail’s discharge time in order to maximize elapsed time for allowing SoC PMIC to execute a controlled and compliant power down sequence. Care should be taken to either disable or isolate any additional peripheral components that may be loading the VSYS_3V3 rail as well.
- Sufficient bulk decoupling capacitance on the First Stage Vout (VSYS_3V3 per PDN) that allows for desired 1.5 – 2 ms elapsed time as described above.
  - TI EVM and Reference Design Study SCHs and PDNs achieve this by using 200 µF of total capacitance on VSYS_3V3. The First Stage Converter (i.e. LM536033-Q1) can typically drive a max of 400 µF to help extend VSYS_3V3 discharge time for a compliant SoC power down sequence.
- Optimizing the Second Stage SoC PMIC’s OTP settings that determines SoC power up and down sequences and total elapsed time needed for a controlled sequence.
  - TI EVM and Reference Design Study SCHs and PDNs achieve this by using optimized OTPs per the SCH and components used. The definition of these OTPs is captured in the detailed timing diagrams for both power up and down sequences. The PDN diagram typically shows a recommended PMIC OTP ID based upon the SoC and DDR memory types.

7.3.8 Example PCB Design

The following sections describe an example PCB design and its resulting PDN performance for the vdd processor power domain.

NOTE

Materials presented in this section are based on generic PDN analysis on PCB boards and are not specific to systems integrating the Device.

7.3.8.1 Example Stack-up

Layer Assignments:

- Layer Top: Signal and Segmented Power Plane
  - Processor and PMIC components placed on Top-side
- Layer 2: Gnd Plane1
- Layer 3: Signals
- Layer n: Power Plane1
- Layer n+1: Power Plane 2
- Layer n+2: Signal
- Layer n+3: Gnd Plane2
- Layer Bottom: Signal and Segmented Power Planes
  - Decoupling caps, etc.

Via Technology: Through-hole

Copper Weight:

- ½ oz for all signal layers.
- 1-2oz for all power plane for improved PCB heat spreading.

7.3.8.2 vdd Example Analysis

Maximum acceptable PCB resistance (R_{eff}) between the PMIC and Processor input power balls should not exceed 10mΩ.

Maximum decoupling capacitance loop inductance (LL) between Processor input power balls and decoupling capacitances should not exceed 2.0nH (ESL NOT included)

Impedance target for key frequency of interest between Processor input power balls and PMIC's SMPS output power balls should not exceed 57mΩ at 20MHz.

Table 7-7. Example PCB vdd PI Analysis Summary

Parameter	Recommendation	Example PCB
OPP	OPP_NOM	
Clocking Rate	266 MHz	
Voltage Level	1 V	1 V
Max Current Draw	1 A	1 A
Max Effective Resistance: Power Inductor Segment Total R_{eff}	10mΩ	9.7 mΩ
Max Loop Inductance	2.0nH	0.97 –1.75nH
Impedance Target	57mΩ F<20Mhz	57mΩ F<20Mhz

Figure 7-15 show a PCB layout example and the resulting PI analysis results.

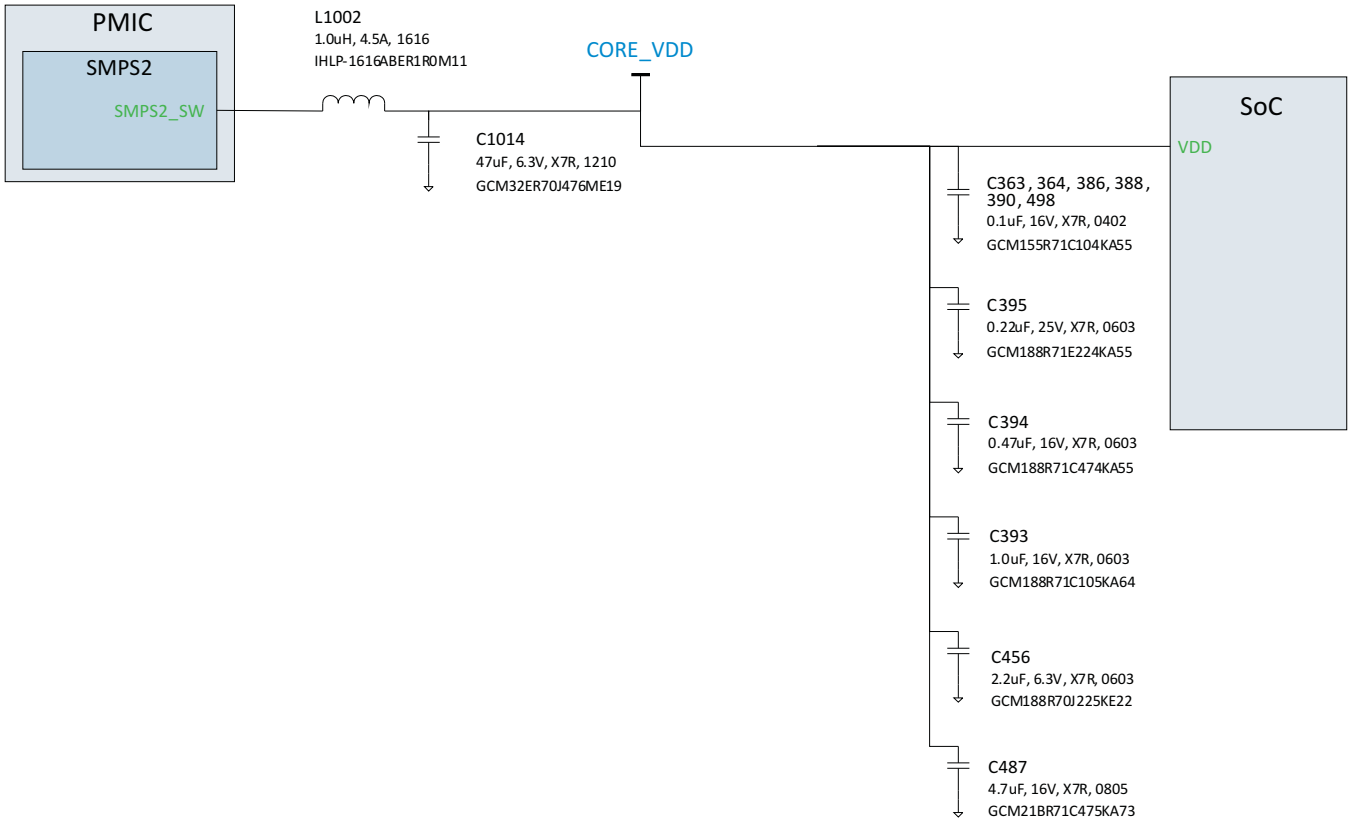


Figure 7-15. vdd Simplified SCH Diagram

NOTE

PCB Etch Resistance Breakdown, PDN Effective Resistance, and vdd routings are UNDER DEVELOPMENT!

IR Drop: vdd (PCB Rev Oct25, CAD sPSI v13.1.1)

- Source Conditions: 1V @ 1A
- Power Plane/Trace Effective Resistances
  - From PMIC SMPS to SoC load = 9.7mohm
  - From Power Inductor to SoC load = 6mohm
  - "Open-Loop" Voltage/IR Drop for 1A = 6mV

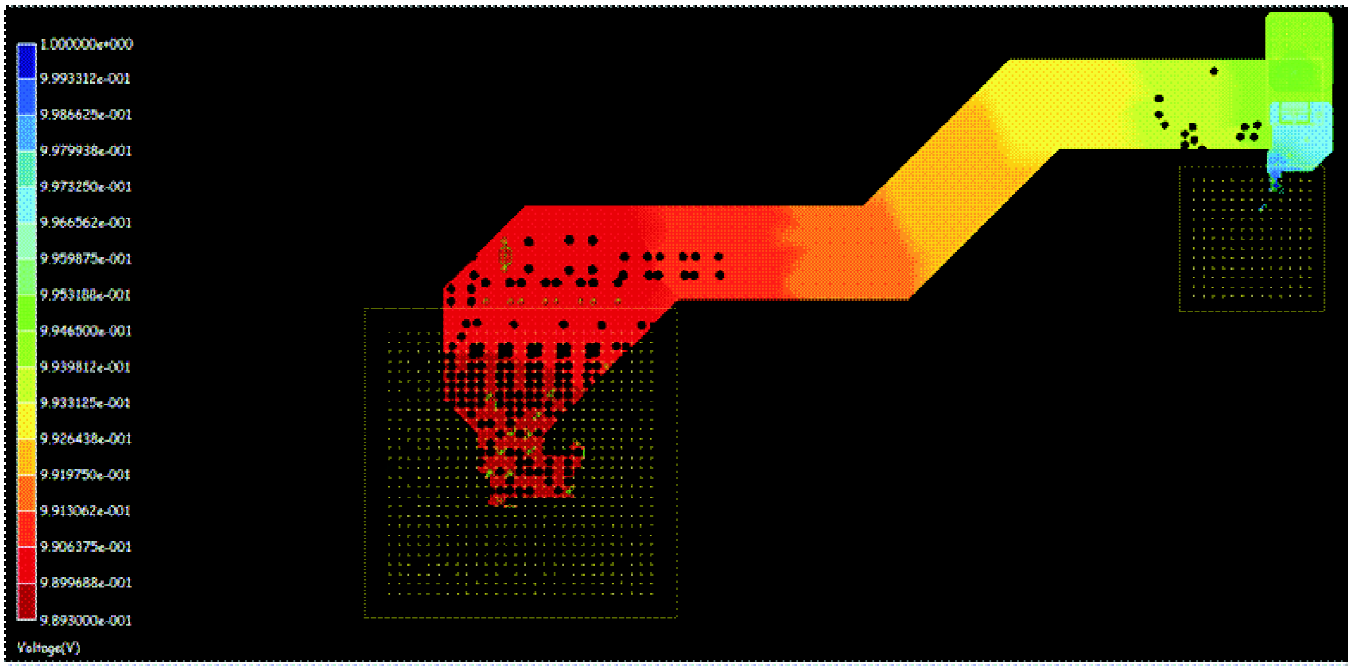


Figure 7-16. vdd Voltage/IR Drop [All Layers]

Dynamic analysis of this PCB design for the CORE power domain determined the vdd decoupling capacitor loop inductance and impedance vs frequency analysis shown below. As you can see, the loop inductance values ranged from 0.97 –1.75nH and were less than maximum 2.0nH recommended.

NOTE

Comparing loop inductances for capacitors at different distances from the SoC's input power balls shows an 18% reduction for caps placed closer. This was derived by averaging the inductances for the 3 caps with distances over 800mils (Avg LL = 1.33nH) vs the 3 caps with distances less than 600mils (Avg LL = 1.096nH).

Table 7-8. Rail - vdd

Cap Ref Des	Model Port #	Loop Inductacne [nH]	Footprint Types	PCB Side	Distance to Ball-Field [mils]	Value [µF]	Size
C487	10	0.97	4vWSE	Top	521	4.7	0805
C393	6	1.11	4vWSE	Bottom	358	1.0	0603
C394	7	1.12	4vWSE	Bottom	357	0.47	0603
C456	9	1.13	4vWSE	Bottom	403	2.2	0603
C386	3	1.16	2vWSE	Bottom	40	0.1	0402
C395	8	1.18	4vWSE	Bottom	460	0.22	0603
C363	1	1.46	2vWSE	Bottom	40	0.1	0402
C390	5	1.48	2vWSE	Bottom	40	0.1	0402
C364	2	1.74	2vWSE	Bottom	40	0.1	0402
C498	11	1.74	2vWSE	Bottom	40	0.1	0402
C388	4	1.75	2vWSE	Bottom	40	0.1	0402

Loop Inductance range: 0.97 –1.75nH

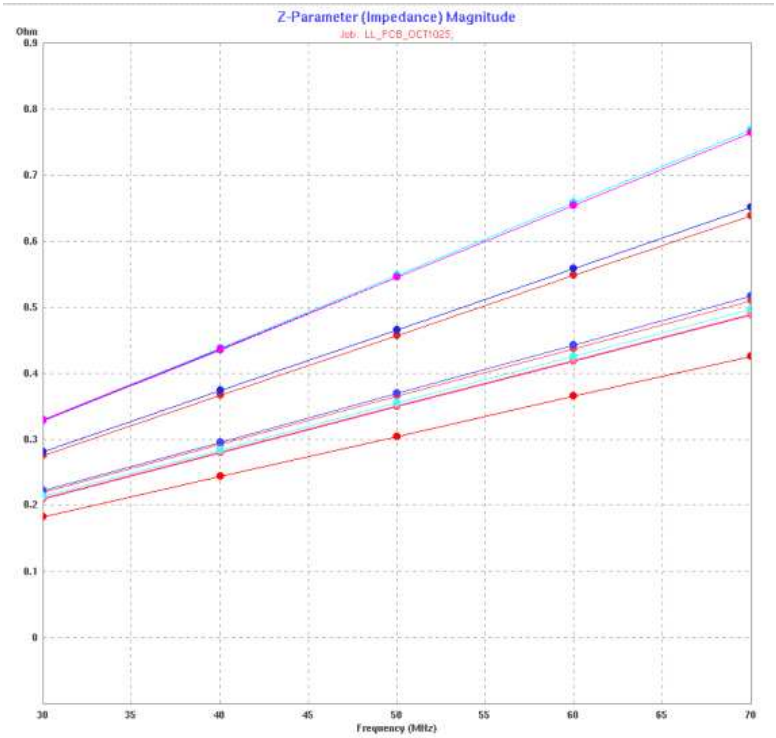


Figure 7-17. vdd Decoupling Cap Loop Inductances

Figure 7-18 shows vdd Impedance vs Frequency characteristics.

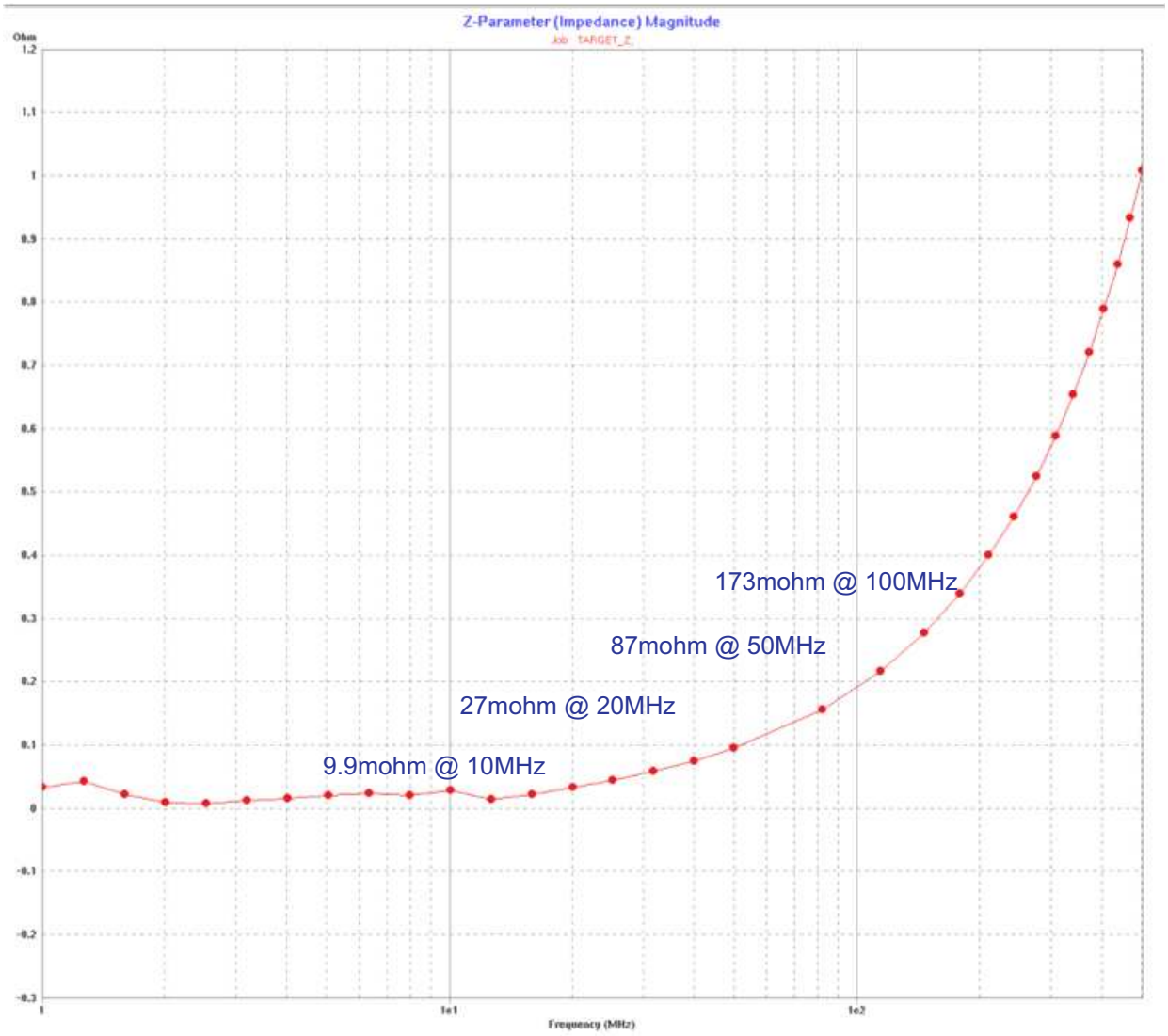


Figure 7-18. vdd Impedance vs Frequency

7.4 Single-Ended Interfaces

7.4.1 General Routing Guidelines

The following paragraphs detail the routing guidelines that must be observed when routing the various functional LVCMOS interfaces.

- Line spacing:
  - For a line width equal to W, the spacing between two lines must be 2W, at least. This minimizes the crosstalk between switching signals between the different lines. On the PCB, this is not achievable everywhere (for example, when breaking signals out from the device package), but it is recommended to follow this rule as much as possible. When violating this guideline, minimize the length of the traces running parallel to each other (see Figure 7-19).

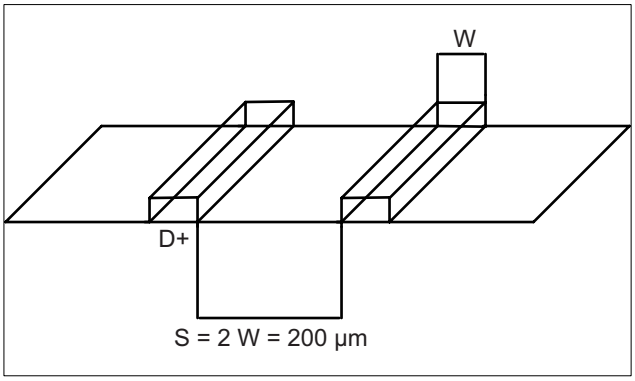


Figure 7-19. Ground Guard Illustration

- Length matching (unless otherwise specified):
  - For bus or traces at frequencies less than 10 MHz, the trace length matching (maximum length difference between the longest and the shortest lines) must be less than 25 mm.
  - For bus or traces at frequencies greater than 10 MHz, the trace length matching (maximum length difference between the longest and the shortest lines) must be less than 2.5 mm.
- Characteristic impedance
  - Unless otherwise specified, the characteristic impedance for single-ended interfaces is recommended to be between 35-Ω and 65-Ω.
- Multiple peripheral support
  - For interfaces where multiple peripherals have to be supported in the star topology, the length of each branch has to be balanced. Before closing the PCB design, it is highly recommended to verify signal integrity based on simulations including actual PCB extraction.

7.4.2 QSPI Board Design and Layout Guidelines

The following section details the routing guidelines that must be observed when routing the QSPI interfaces.

- The qspi1_sclk output signal must be looped back into the qspi1_rtclk input.
- The signal propagation delay from the qspi1_sclk ball to the QSPI device CLK input pin (A to C) must be approximately equal to the signal propagation delay from the QSPI device CLK pin to the qspi1_rtclk ball (C to D).
- The signal propagation delay from the QSPI device CLK pin to the qspi1_rtclk ball (C to D) must be approximately equal to the signal propagation delay of the control and data signals between the QSPI device and the SoC device (E to F, or F to E).
- The signal propagation delay from the qspi1_sclk signal to the series terminators (R2 = 10 Ω) near the QSPI device must be < 450pS (~7cm as stripline or ~8cm as microstrip)
- 50 Ω PCB routing is recommended along with series terminations, as shown in Figure 7-20.

- Propagation delays and matching:
  - A to C = C to D = E to F.
  - Matching skew: < 60pS
  - A to B < 450pS
  - B to C = as small as possible (<60pS)

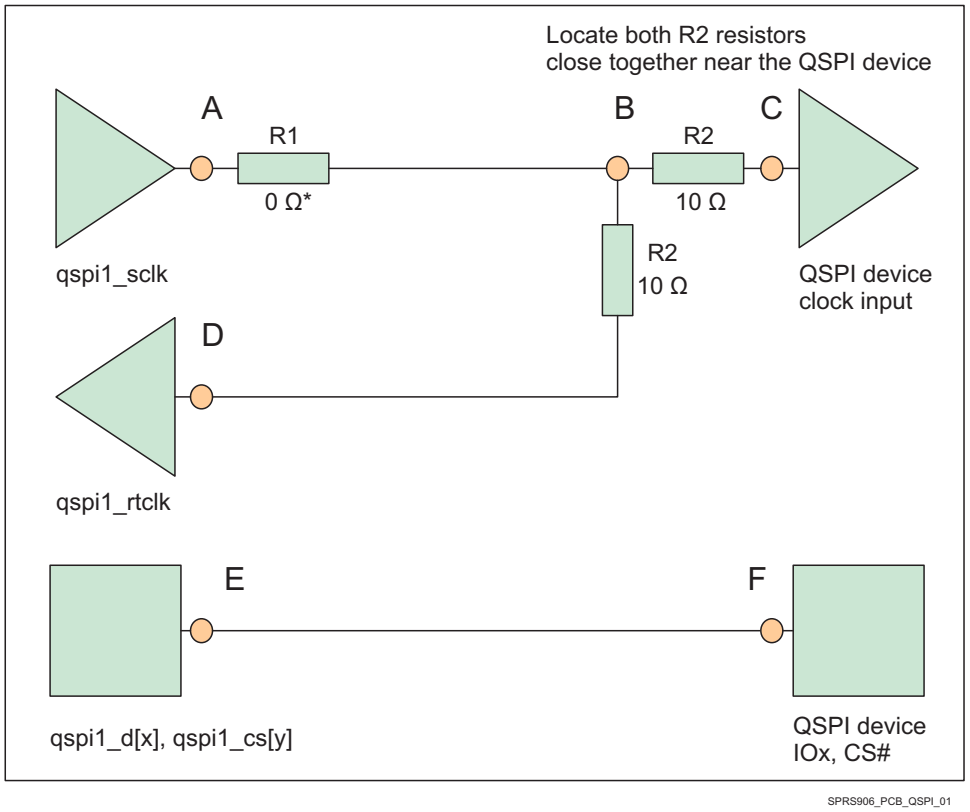


Figure 7-20. QSPI Interface High Level Schematic

NOTE

*0 Ω resistor (R1), located as close as possible to the qspi1_sclk pin, is placeholder for fine-tuning if needed.

7.5 Differential Interfaces

7.5.1 General Routing Guidelines

To maximize signal integrity, proper routing techniques for differential signals are important for high-speed designs. The following general routing guidelines describe the routing guidelines for differential lanes and differential signals.

- As much as possible, no other high-frequency signals must be routed in close proximity to the differential pair.
- Must be routed as differential traces on the same layer. The trace width and spacing must be chosen to yield the differential impedance value recommended.
- Minimize external components on differential lanes (like external ESD, probe points).
- Through-hole pins are not recommended.
- Differential lanes mustn't cross image planes (ground planes).
- No sharp bend on differential lanes.

- Number of vias on the differential pairs must be minimized, and identical on each line of the differential pair. In case of multiple differential lanes in the same interface, all lines should have the same number of vias.
- Shielded routing is to be promoted as much as possible (for instance, signals must be routed on internal layers that are inside power and/or ground planes).

7.5.2 USB 2.0 Board Design and Layout Guidelines

This section discusses schematic guidelines when designing a universal serial bus (USB) system.

7.5.2.1 Background

Clock frequencies generate the main source of energy in a USB design. The USB differential DP/DM pairs operate in high-speed mode at 480 Mbps. System clocks can operate at 12 MHz, 48 MHz, and 60 MHz. The USB cable can behave as a monopole antenna; take care to prevent RF currents from coupling onto the cable.

When designing a USB board, the signals of most interest are:

- Device interface signals: Clocks and other signal/data lines that run between devices on the PCB.
- Power going into and out of the cable: The USB connector socket pin 1 (VBUS) may be heavily filtered and need only pass low frequency signals of less than ~100 KHz. The USB socket pin 4 (analog ground) must be able to return the current during data transmission, and must be filtered sparingly.
- Differential twisted pair signals going out on cable, DP and DM: Depending upon the data transfer rate, these device terminals can have signals with fundamental frequencies of 240 MHz (high speed), 6 MHz (full speed), and 750 kHz (low speed).
- External crystal circuit (device terminals XI and X0): 12 MHz, 19.2 MHz, 24 MHz, and 48 MHz fundamental. When using an external crystal as a reference clock, a 24 MHz and higher crystal is highly recommended.

7.5.2.2 USB PHY Layout Guide

The following sections describe in detail the specific guidelines for USB PHY Layout.

7.5.2.2.1 General Routing and Placement

Use the following routing and placement guidelines when laying out a new design for the USB physical layer (PHY). These guidelines help minimize signal quality and electromagnetic interference (EMI) problems on a four-or-more layer evaluation module (EVM).

- Place the USB PHY and major components on the un-routed board first. For more details, see Section 7.5.2.2.2.3.
- Route the high-speed clock and high-speed USB differential signals with minimum trace lengths.
- Route the high-speed USB signals on the plane closest to the ground plane, whenever possible.
- Route the high-speed USB signals using a minimum of vias and corners. This reduces signal reflections and impedance changes.
- When it becomes necessary to turn 90°, use two 45° turns or an arc instead of making a single 90° turn. This reduces reflections on the signal traces by minimizing impedance discontinuities.
- Do not route USB traces under or near crystals, oscillators, clock signal generators, switching regulators, mounting holes, magnetic devices or IC's that use or duplicate clock signals.
- Avoid stubs on the high-speed USB signals because they cause signal reflections. If a stub is unavoidable, then the stub should be less than 200 mils.
- Route all high-speed USB signal traces over continuous planes (V_{CC} or GND), with no interruptions. Avoid crossing over anti-etch, commonly found with plane splits.

7.5.2.2.2 Specific Guidelines for USB PHY Layout

The following sections describe in detail the specific guidelines for USB PHY Layout.

7.5.2.2.2.1 Analog, PLL, and Digital Power Supply Filtering

To minimize EMI emissions, add decoupling capacitors with a ferrite bead at power supply terminals for the analog, phase-locked loop (PLL), and digital portions of the chip. Place this array as close to the chip as possible to minimize the inductance of the line and noise contributions to the system. An analog and digital supply example is shown in Figure 7-21. In case of multiple power supply pins with the same function, tie them up to a single low-impedance point in the board and then add the decoupling capacitors, in addition to the ferrite bead. This array of caps and ferrite bead improve EMI and jitter performance. Take both EMI and jitter into account before altering the configuration.

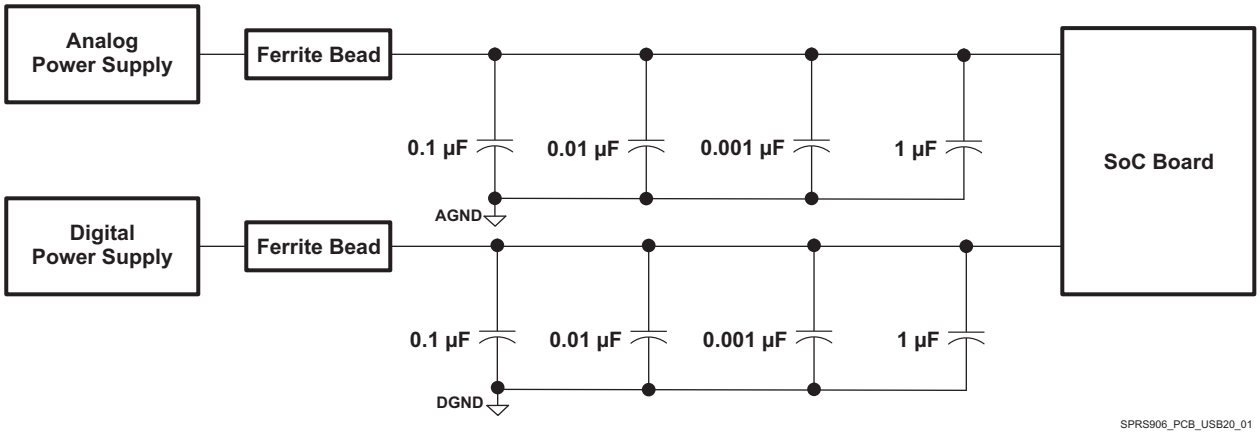


Figure 7-21. Suggested Array Capacitors and a Ferrite Bead to Minimize EMI

Consider the recommendations listed below to achieve proper ESD/EMI performance:

- Use a 0.01 µF cap on each cable power VBUS line to chassis GND close to the USB connector pin.
- Use a 0.01 µF cap on each cable ground line to chassis GND next to the USB connector pin.
- If voltage regulators are used, place a 0.01 µF cap on both input and output. This is to increase the immunity to ESD and reduce EMI. For other requirements, see the device-specific datasheet.

7.5.2.2.2.2 Analog, Digital, and PLL Partitioning

If separate power planes are used, they must be tied together at one point through a low-impedance bridge or preferably through a ferrite bead. Care must be taken to capacitively decouple each power rail close to the device. The analog ground, digital ground, and PLL ground must be tied together to the low-impedance circuit board ground plane.

7.5.2.2.2.3 Board Stackup

Because of the high frequencies associated with the USB, a printed circuit board with at least four layers is recommended; two signal layers separated by a ground and power layer as shown in Figure 7-22.

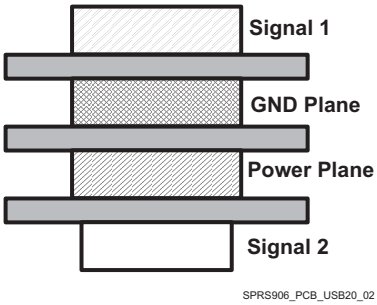


Figure 7-22. Four-Layer Board Stack-Up

The majority of signal traces should run on a single layer, preferably SIGNAL1. Immediately next to this layer should be the GND plane, which is solid with no cuts. Avoid running signal traces across a split in the ground or power plane. When running across split planes is unavoidable, sufficient decoupling must be used. Minimizing the number of signal vias reduces EMI by reducing inductance at high frequencies.

7.5.2.2.2.4 Cable Connector Socket

Short the cable connector sockets directly to a small chassis ground plane (GND *strap*) that exists immediately underneath the connector sockets. This shorts EMI (and ESD) directly to the chassis ground before it gets onto the USB cable. This etch plane should be as large as possible, but all the conductors coming off connector pins 1 through 6 must have the board signal GND plane run under. If needed, scoop out the chassis GND strap etch to allow for the signal ground to extend under the connector pins. Note that the etches coming from pins 1 and 4 (VBUS power and GND) should be wide and via-ed to their respective planes as soon as possible, respecting the filtering that may be in place between the connector pin and the plane. See Figure 7-23 for a schematic example.

Place a ferrite in series with the cable shield pins near the USB connector socket to keep EMI from getting onto the cable shield. The ferrite bead between the cable shield and ground may be valued between 10 Ω and 50 Ω at 100 MHz; it should be resistive to approximately 1 GHz. To keep EMI from getting onto the cable bus power wire (a very large antenna) a ferrite may be placed in series with cable bus power, VBUS, near the USB connector pin 1. The ferrite bead between connector pin 1 and bus power may be valued between 47 Ω and approximately 1000 Ω at 100 MHz. It should continue being resistive out to approximately 1 GHz, as shown in Figure 7-23.

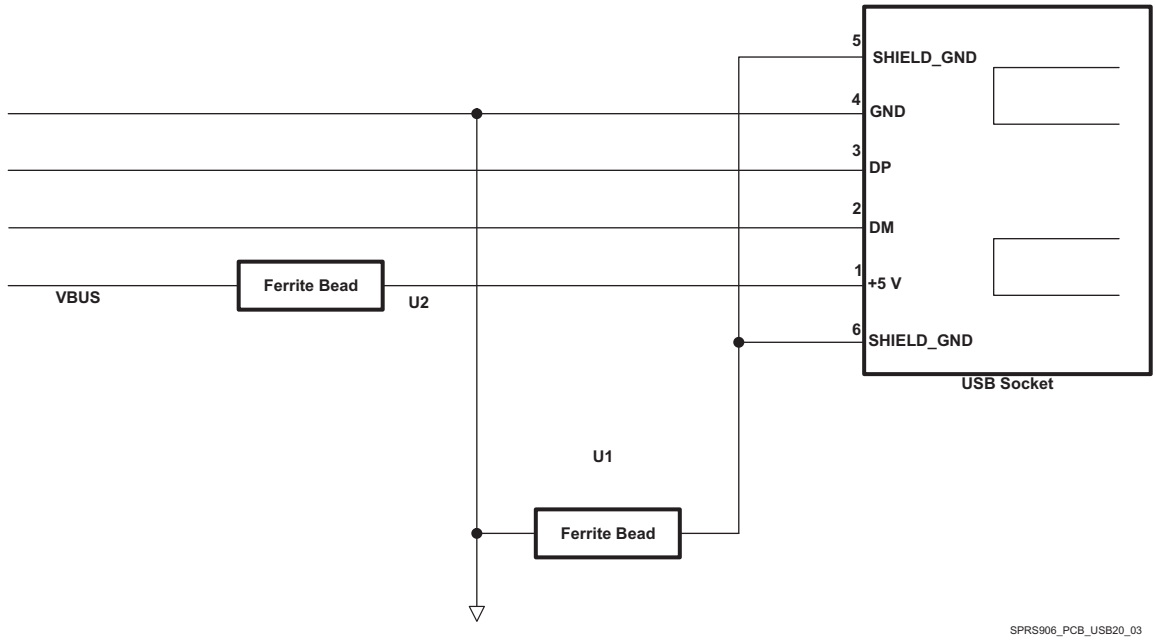


Figure 7-23. USB Connector

7.5.2.2.2.5 Clock Routings

To address the system clock emissions between devices, place a ~10 to 130 Ω resistor in series with the clock signal. Use a trial and error method of looking at the shape of the clock waveform on a high-speed oscilloscope and of tuning the value of the resistance to minimize waveform distortion. The value on this resistor should be as small as possible to get the desired effect. Place the resistor close to the device generating the clock signal. If an external crystal is used, follow the guidelines detailed in the *Selection and Specification of Crystals for Texas Instruments USB 2.0 Devices* (SLLA122).

When routing the clock traces from one device to another, try to use the 3W spacing rule. The distance from the center of the clock trace to the center of any adjacent signal trace should be at least three times the width of the clock trace. Many clocks, including slow frequency clocks, can have fast rise and fall times. Using the 3W rule cuts down on crosstalk between traces. In general, leave space between each of the traces running parallel between the devices. Avoid using right angles when routing traces to minimize the routing distance and impedance discontinuities. For further protection from crosstalk, run guard traces beside the clock signals (GND pin to GND pin), if possible. This lessens clock signal coupling, as shown in Figure 7-24.

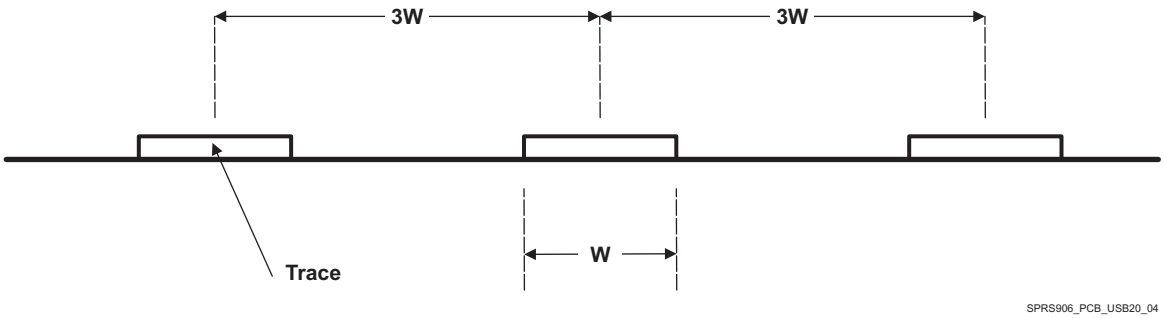


Figure 7-24. 3W Spacing Rule

7.5.2.2.2.6 Crystals/Oscillator

Keep the crystal and its load capacitors close to the USB PHY pins, XI and XO (see Figure 7-25). Note that frequencies from power sources or large capacitors can cause modulations within the clock and should not be placed near the crystal. In these instances, errors such as dropped packets occur. A placeholder for a resistor, in parallel with the crystal, can be incorporated in the design to assist oscillator startup.

Power is proportional to the current squared. The current is $I = C \times dv/dt$, because dv/dt is a function of the PHY, current is proportional to the capacitive load. Cutting the load to 1/2 decreases the current by 1/2 and the power to 1/4 of the original value. For more details on crystal selection, see the *Selection and Specification of Crystals for Texas Instruments USB 2.0 Devices* (SLLA122).

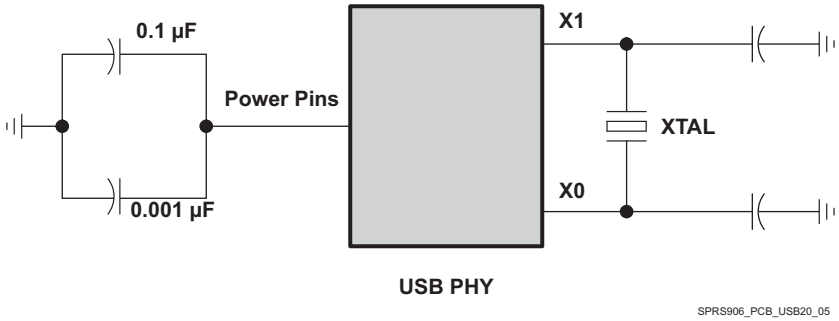


Figure 7-25. Power Supply and Clock Connection to the USB PHY

7.5.2.2.2.7 DP/DM Trace

Place the USB PHY as close as possible to the USB 2.0 connector. The signal swing during high-speed operation on the DP/DM lines is relatively small (400 mV ± 10%), so any differential noise picked up on the twisted pair can affect the received signal. When the DP/DM traces do not have any shielding, the traces tend to behave like an antenna and picks up noise generated by the surrounding components in the environment. To minimize the effect of this behavior:

- DP/DM traces should always be matched lengths and must be no more than 4 inches in length; otherwise, the eye opening may be degraded (see Figure 7-26).
- Route DP/DM traces close together for noise rejection on differential signals, parallel to each other and within two mils in length of each other. The measurement for trace length must be started from device's balls.
- A high-speed USB connection is made through a shielded, twisted pair cable with a differential characteristic impedance of 90 Ω ±15%. In layout, the impedance of DP and DM should each be 45 Ω ± 10%.
- DP/DM traces should not have any extra components to maintain signal integrity. For example, traces cannot be routed to two USB connectors.

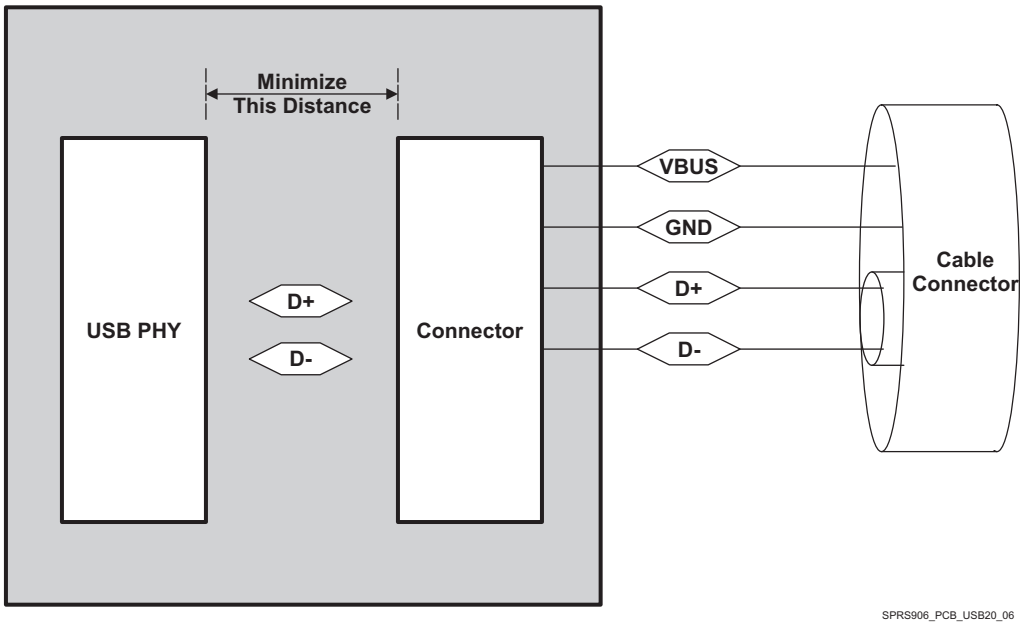


Figure 7-26. USB PHY Connector and Cable Connector

7.5.2.2.2.8 DP/DM Vias

When a via must be used, increase the clearance size around it to minimize its capacitance. Each via introduces discontinuities in the signal's transmission line and increases the chance of picking up interference from the other layers of the board. Be careful when designing test points on twisted pair lines; through-hole pins are not recommended.

7.5.2.2.2.9 Image Planes

An image plane is a layer of copper (voltage plane or ground plane), physically adjacent to a signal routing plane. Use of image planes provides a low impedance, shortest possible return path for RF currents. For a USB board, the best image plane is the ground plane because it can be used for both analog and digital circuits.

- Do not route traces so they cross from one plane to the other. This can cause a broken RF return path resulting in an EMI radiating loop as shown in Figure 7-27. This is important for higher frequency or repetitive signals. Therefore, on a multi-layer board, it is best to run all clock signals on the signal plane above a solid ground plane.
- Avoid crossing the image power or ground plane boundaries with high-speed clock signal traces immediately above or below the separated planes. This also holds true for the twisted pair signals (DP, DM). Any unused area of the top and bottom signal layers of the PCB can be filled with copper that is connected to the ground plane through vias.

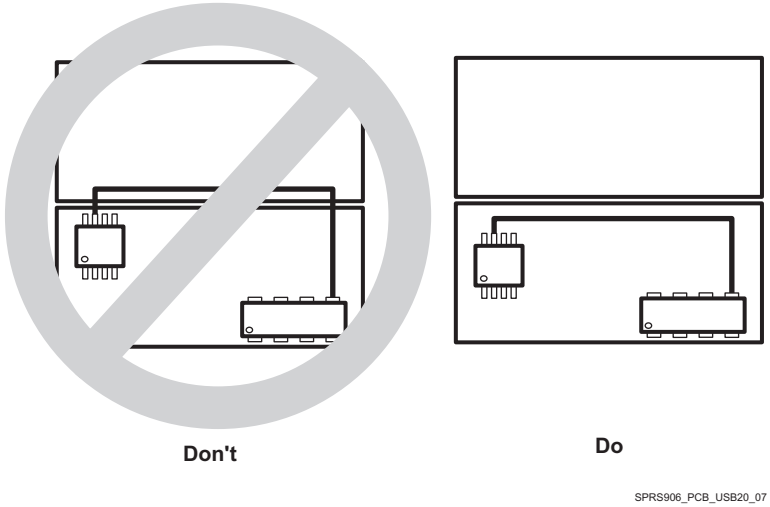


Figure 7-27. Do Not Cross Plane Boundaries

- Do not overlap planes that do not reference each other. For example, do not overlap a digital power plane with an analog power plane as this produces a capacitance between the overlapping areas that could pass RF emissions from one plane to the other, as shown in Figure 7-28.

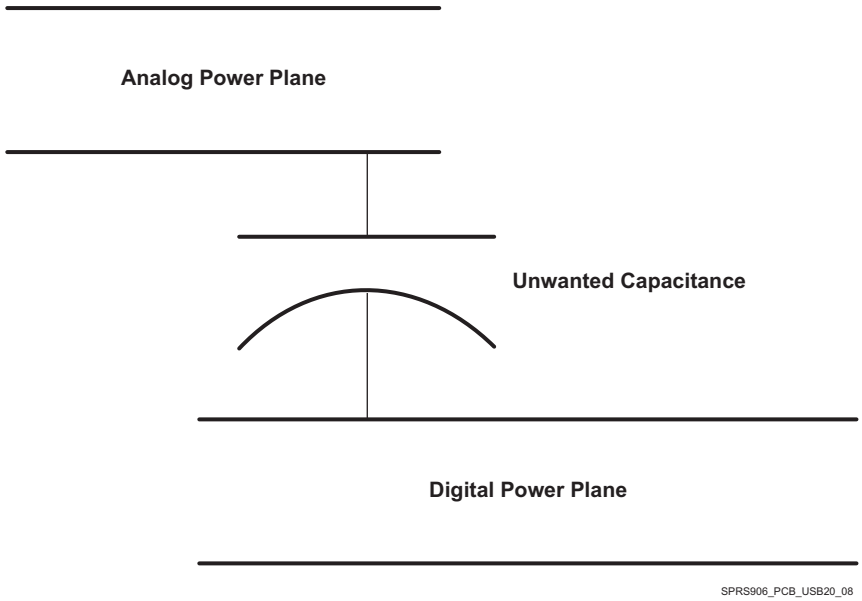


Figure 7-28. Do Not Overlap Planes

- Avoid image plane violations. Traces that route over a slot in an image plane results in a possible RF return loop, as shown in Figure 7-29.

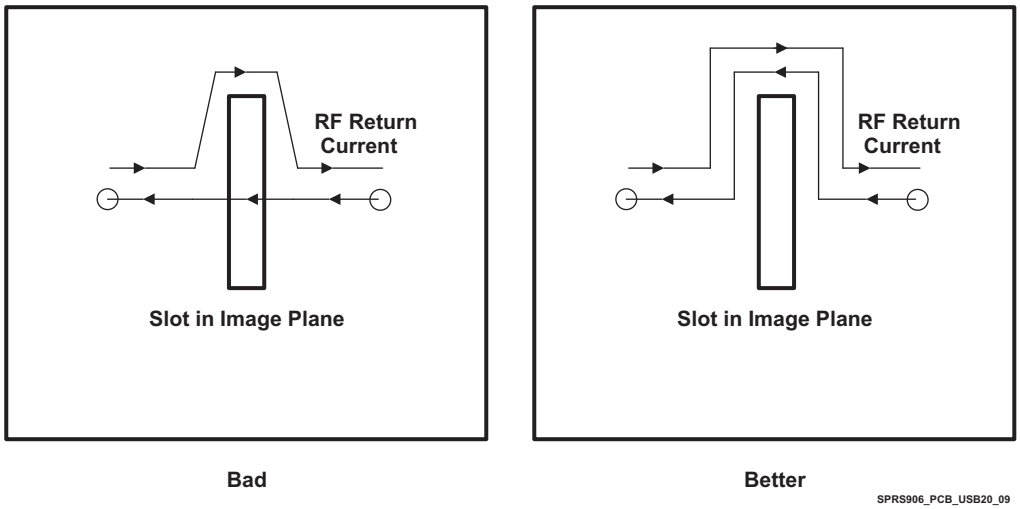


Figure 7-29. Do Not Violate Image Planes

7.5.2.2.2.10 Power Regulators

Switching power regulators are a source of noise and can cause noise coupling if placed close to sensitive areas on a circuit board. Therefore, the switching power regulator should be kept away from the DP/DM signals, the external clock crystal (or clock oscillator), and the USB PHY.

7.5.2.3 References

- *USB 2.0 Specification*, Intel, 2000, http://www.usb.org/developers/docs/
- *High Speed USB Platform Design Guidelines*, Intel, 2000, http://www.intel.com/technology/usb/download/usb2dg_R1_0.pdf
- *Selection and Specification of Crystals for Texas Instruments USB 2.0 Devices* (SLLA122)

7.5.3 USB 3.0 Board Design and Layout Guidelines

This section provides the timing specification for the USB3.0 (USB1 in the device) interface as a PCB design and manufacturing specification. The design rules constrain PCB trace length, PCB trace skew, signal integrity, cross-talk, and signal timing. TI has performed the simulation and system design work to ensure the USB3.0 interface requirements are met. The design rules stated within this document are targeted at DEVICE mode electrical compliance. HOST mode and/or systems that do not include the 3m USB cable and far-end 11-inch PCB trace required by DEVICE mode compliance testing may not need the complete list of optimizations shown in this document; however, applying these optimizations to HOST mode systems will lead to optimal DEVICE mode performance.

7.5.3.1 USB 3.0 interface introduction

The USB 3.0 has two unidirectional differential pairs: TXp/TXn pair and RXp/RXn pair. AC coupling caps are needed on the board for TX traces.

Figure 7-30 present high level schematic diagram for USB 3.0 interface.

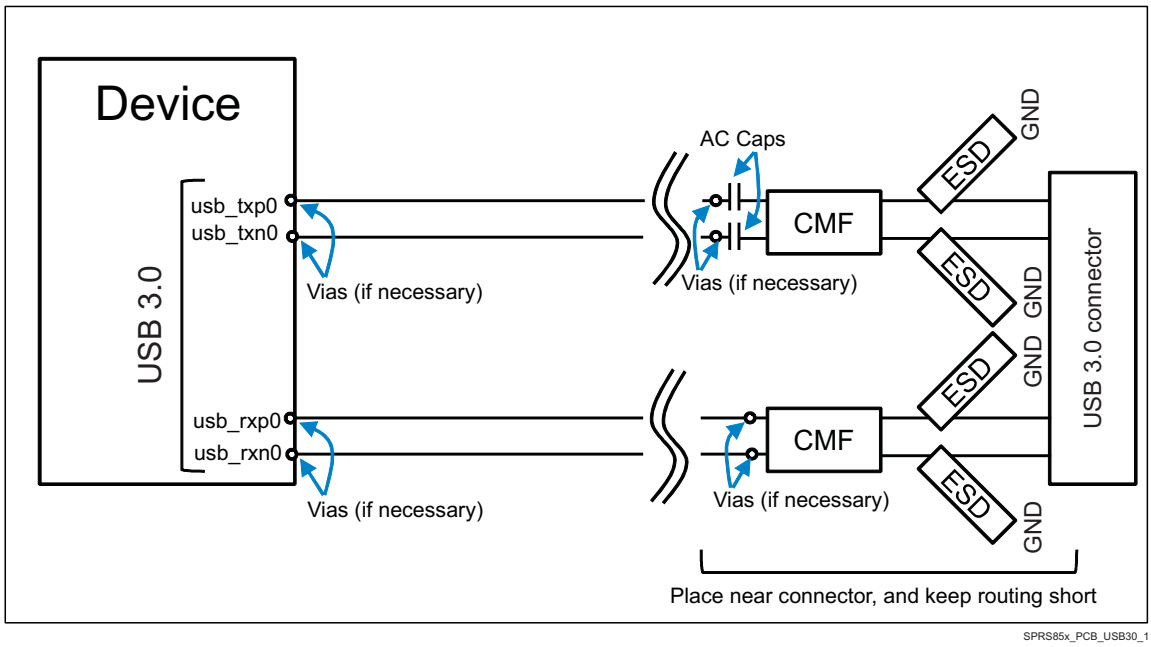


Figure 7-30. USB 3.0 Interface High Level Schematic

NOTE

ESD components should be on a PCB layer next to a system GND plane layer so the inductance of the via to GND will be minimal.

If vias are used, place the vias near the AC Caps or CMFs and under the SoC BGA, if necessary.

Figure 7-31 present placement diagram for USB 3.0 interface.

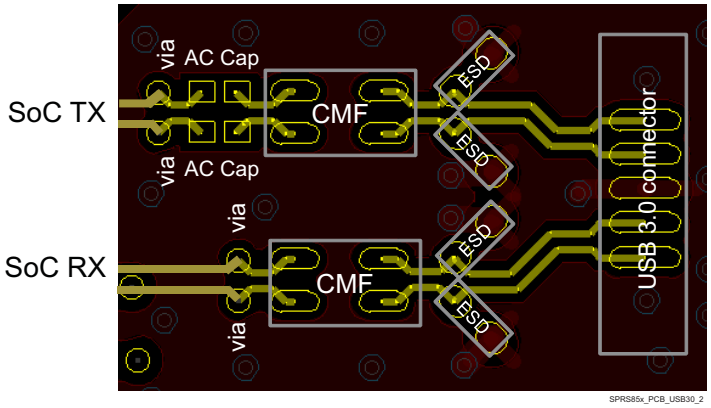


Figure 7-31. USB 3.0 placement diagram

Table 7-9. USB1 Component Reference

INTERFACE	COMPONENT	SUPPLIER	PART NUMBER
USB3 PHY	ESD	TI	TPD1E05U06
	CMF	Murata	DLW21SN900HQ2
	C	-	100nF (typical size: 0201)

7.5.3.2 USB 3.0 General routing rules

Some general routing guidelines regarding USB 3.0:

- Avoid crossing splits reference plane(s).
- Shorter trace length is preferred.
- Minimize the via usage and layer transition
- Keep large spacing between TX and RX pairs.
- Intra-lane delay mismatch between DP and DM less than 1ps. Same for RXp and RXn.
- Distance between common mode filter (CMF) and ESD protection device should be as short as possible
- Distance between ESD protection device and USB connector should be as short as possible.
- Distance between AC capacitors (TX only) and CMF should be as short as possible.
- USB 3.0 signals should always be routed over an adjacent ground plane.

Table 7-10 and Table 7-11 present routing specification and recommendations for USB1 in the device.

Table 7-10. USB1 Routing Specifications

PARAMETER	MIN	TYP	MAX	UNIT
Device balls to USB 3.0 connector trace length			3500	Mils
Skew within a differential pair		3	6	Mils
Number of stubs allowed on TX/RX traces			0	Stubs
TX/RX pair differential impedance	83	90	97	Ω
Number of vias on each TX/RX trace			2	Vias
Differential pair to any other trace spacing	2xDS	3xDS		
Number of ground plane cuts allowed within USB3 routing region (except for specific ground carving as explained in this document)			0	Cuts
Number of layers between USB3.0 routing region and reference ground plane			0	Layers
PCB trace width		6		Mils
PCB BGA escape via pad size		18		Mils
PCB BGA escape via hole size		10		Mils

1. Vias must be used in pairs and spaced equally along a signal path.
2. DS = differential spacing of the traces.
3. Exceptions may be necessary in the SoC package BGA area.
4. GND guard-bands on the same layer may be closer, but should not be allowed to affect the impedance of the differential pair routing. GND guard-bands to isolate USB3.0 differential pairs from all other signals are recommended.

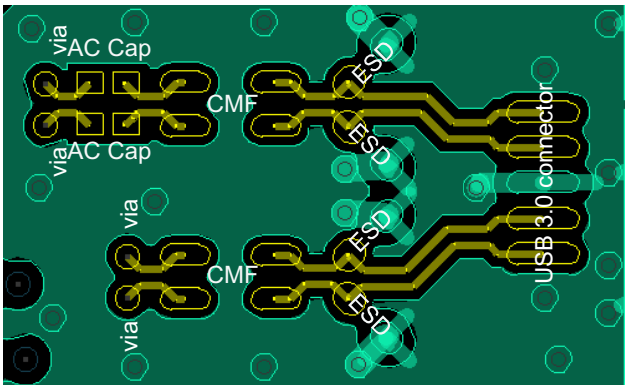
Table 7-11. USB1 Routing Recommendations

Item	Description	Reason
ESD location	Place ESD component on same layer as connector (no via or stub to ESD component)	Eliminate reflection loss from via & stub to ESD
ESD part number	TPD1E05U06	Minimize capacitance (0.42pF)
CMF part number	DLW21SN900HQ2	Manufacturer's recommended device
Connector	Use USB3.0 connector with supporting s-parameter model	Enable full signal chain simulation
Carve Ground	Carve GND underneath AC Caps, ESD, CMF, and connector	Minimize capacitance under ESD and CMF

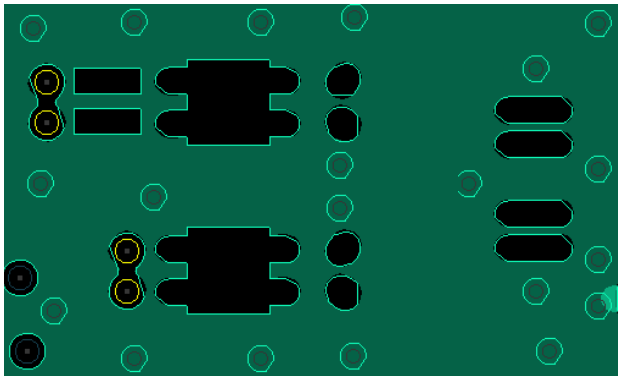
Table 7-11. USB1 Routing Recommendations (continued)

Item	Description	Reason
Round pads	Minimize pad size and round the corners of the pads for the ESD and CMF components	Minimize capacitance
Vias	Max 2 vias per signal trace. If vias are required, place vias close to the AC Caps and CMFs. Vias under the SoC grid array may be used if necessary to route signals away from BGA pattern.	Vias significantly degrade signal integrity at 2.5GHz

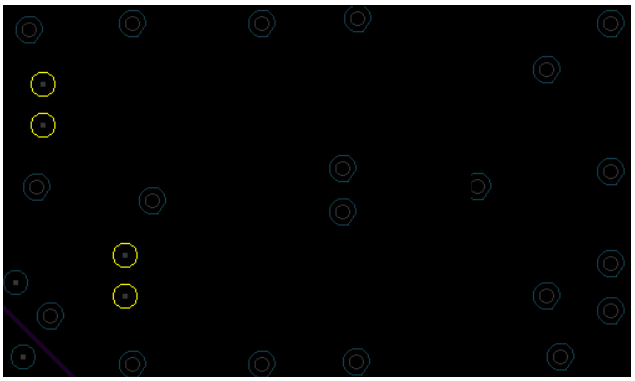
Figure 7-32 presents an example layout, demonstrating the “carve GND” concept.



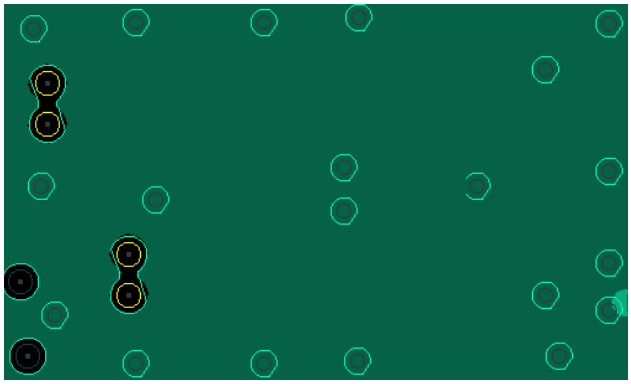
Top Layer: Routing from SoC through AC Caps, CMF, and ESD to connector.



Layer2, GND: Gaps carved in GND underneath AC Caps, CMF, ESD, and connector.



Layer3, Signal: Implement as keep-out zone underneath carved GND areas.



Layer4, GND Plane underneath AC Caps, CMF, ESD, and connector.

SPRS85x_PCB_USB30_3

Figure 7-32. USB 3.0 Example “carve GND” layout

7.5.4 HDMI Board Design and Layout Guidelines

This section provides the timing specification for the HDMI interface as a PCB design and manufacturing specification. The design rules constrain PCB trace length, PCB trace skew, signal integrity, cross-talk, and signal timing. TI has performed the simulation and system design work to ensure the HDMI interface requirements are met. The design rules stated within this document are targeted at resolutions less than or equal to 1080p60 with 8-bit color; deep color (10-bit) requires further signal integrity optimization.

7.5.4.1 HDMI Interface Schematic

The HDMI bus is separated into three main sections (HDMI Ethernet and the optional Audio Return Channel are not specifically supported by this Device):

1. Transition Minimized Differential Signaling (TMDS) high speed digital video interface
2. Display Data Channel (I2C bus for configuration and status exchange between two devices)
3. Consumer Electronics Control (optional) for remote control of connected devices.

The DDC and CEC are low speed interfaces, so nothing special is required for PCB layout of these signals.

The TMDS channels are high speed differential pairs and therefore require the most care in layout. Specifications for TMDS layout are below.

Figure 7-33 shows the HDMI interface schematic.

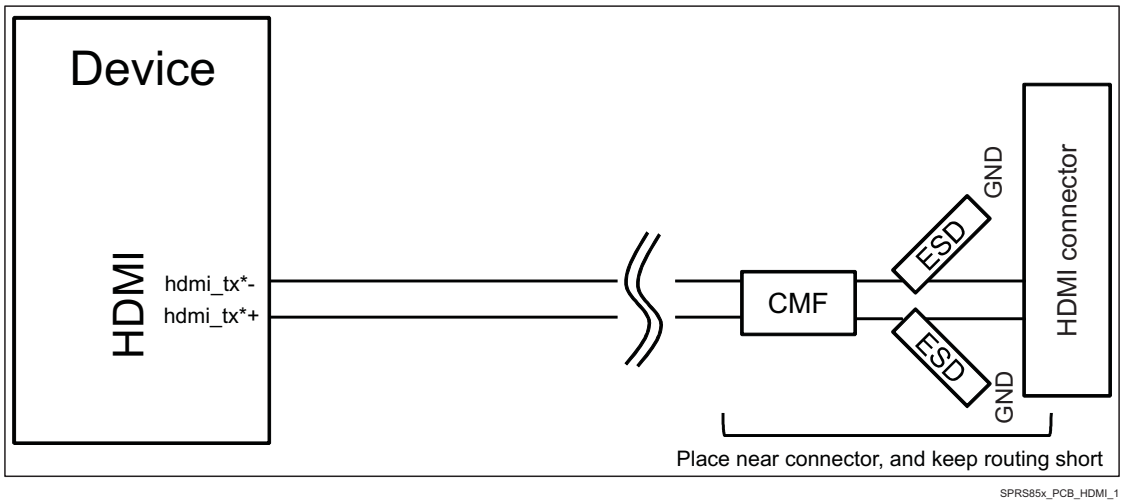


Figure 7-33. HDMI Interface High Level Schematic

Figure 7-34 presents placement diagram for HDMI interface.

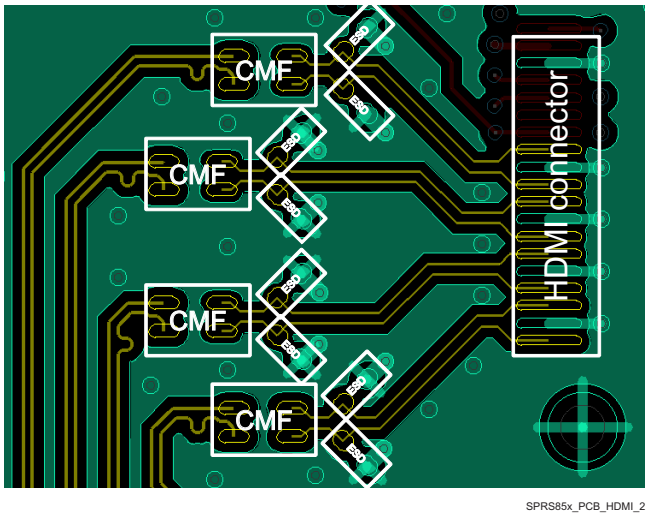


Figure 7-34. HDMI Placement Diagram

Table 7-12. HDMI Component Reference

INTERFACE	DEVICE	SUPPLIER	PART NUMBER
HDMI	ESD	TI	TPD1E05U06
	CMF	Murata	DLW21SN900HQ2

7.5.4.2 TMDS General Routing Guidelines

The TMDS signals are high speed differential pairs. Care must be taken in the PCB layout of these signals to ensure good signal integrity.

The TMDS differential signal traces must be routed to achieve 100 Ohms (+/- 10%) differential impedance and 60 ohms (+/-10%) single ended impedance. Single ended impedance control is required because differential signals can't be closely coupled on PCBs and therefore single ended impedance becomes important.

These impedances are impacted by trace width, trace spacing, distance to reference planes, and dielectric material. Verify with a PCB design tool that the trace geometry for both data signal pairs results in as close to 60 ohms impedance traces as possible. For best accuracy, work with your PCB fabricator to ensure this impedance is met.

In general, closely coupled differential signal traces are not an advantage on PCBs. When differential signals are closely coupled, tight spacing and width control is necessary. Very small width and spacing variations affect impedance dramatically, so tight impedance control can be more problematic to maintain in production.

Loosely coupled PCB differential signals make impedance control much easier. Wider traces and spacing make obstacle avoidance easier, and trace width variations don't affect impedance as much, therefore it's easier to maintain accurate impedance over the length of the signal. The wider traces also show reduced skin effect and therefore often result in better signal integrity.

Some general routing guidelines regarding TMDS:

- Avoid crossing splits reference plane(s).
- Shorter trace length is preferred.
- Distance between common mode filter (CMF) and ESD protection device should be as short as possible
- Distance between ESD protection device and HDMI connector should be as short as possible.

Table 7-13 shows the routing specifications for the TMDS signals.

Table 7-13. TMDS Routing Specifications

PARAMETER	MIN	TYP	MAX	UNIT
Device balls to HDMI header trace length			4000	Mils
Skew within a differential pair		3	5	Mils
Number of stubs allowed on TMDS traces			0	stubs
TMDS pair differential impedance	90	100	110	Ω
TMDS single-ended impedance	54	60	66	Ω
Number of vias on each TMDS trace			0	Vias
TMDS differential pair to any other trace spacing(1) (2) (3)	2×DS	3xDS		Mils
Number of ground plane cuts allowed within HDMI routing region (except for specific ground carving as explained in this document)			0	Cuts
Number of layers between HDMI routing region and reference ground plane			0	Layers
PCB trace width		4.4		Mils

(1) DS = differential spacing of the traces.

(2) Exceptions may be necessary in the SoC package BGA area.

(3) GND guard-bands may be closer, but should not be allowed to affect the impedance of the differential pair routing. GND guard-bands to isolate HDMI differential pairs from all other signals is recommended.

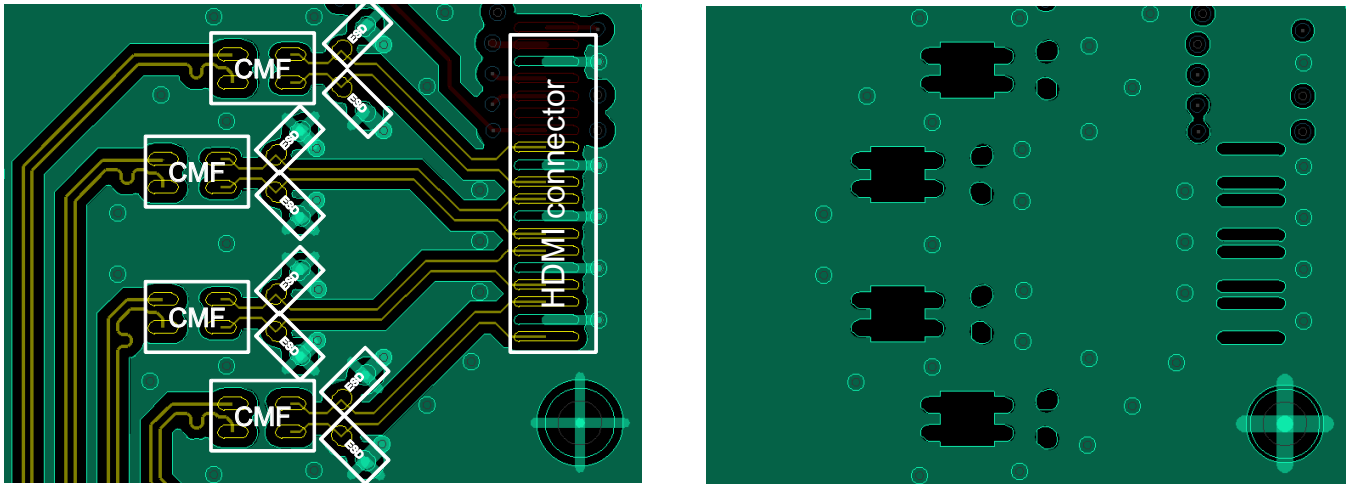
Table 7-14. TDMS Routing Recommendations

Item	Description	Reason
ESD part number	TPD1E05U06	Minimize capacitance (0.42pF)

Table 7-14. TDMS Routing Recommendations (continued)

Item	Description	Reason
Carve Ground	Carve GND underneath ESD and CMF	Minimize capacitance under ESD and CMF
Round pads	Reduce pad size and round the corners of the pads for the ESD and CMF components	Minimize capacitance
Routing layer	Route all signals only on the same layer as SoC	Minimize reflection loss

Figure 7-35presents an example layout, demonstrating the “carve GND” concept.



Top Layer: Routing from SoC through CMF, and ESD to connector.

Layer2, GND: Gaps carved in GND underneath, CMF, ESD, and connector.

SPRS85x_PCB_HDMI_3

Figure 7-35. HDMI Example “carve GND” layout

7.5.4.3 TPD5S115

The TPD5S115 is an integrated HDMI companion chip solution. The device provides a regulated 5 V output (5VOUT) for sourcing the HDMI power line. The TPD5S115 exceeds the IEC61000-4-2 (Level 4) ESD protection level.

7.5.4.4 HDMI ESD Protection Device (Required)

Interfaces that connect to a cable such as HDMI generally require more ESD protection than can be built into the processor’s outputs. Therefore this HDMI interface requires the use of an ESD protection chip to provide adequate ESD.

When selecting an ESD protection chip, choose the lowest capacitance ESD protection available to minimize signal degradation. In no case should be ESD protection circuit capacitance be more than 5pF.

TI manufactures these devices that provide ESD protection for HDMI signals such as the TPDxE05U06. For more information see the www.ti.com website.

7.5.4.5 PCB Stackup Specifications

Table 7-15 shows the stackup and feature sizes required for HDMI.

Table 7-15. HDMI PCB Stackup Specifications

PARAMETER	MIN	TYP	MAX	UNIT
PCB Routing/Plane Layers	4	6	-	Layers

Table 7-15. HDMI PCB Stackup Specifications (continued)

PARAMETER	MIN	TYP	MAX	UNIT
Signal Routing Layers	2	3	-	Layers
Number of ground plane cuts allowed within HDMI routing region	-	-	0	Cuts
Number of layers between HDMI routing region and reference ground plane	-	-	0	Layers
PCB Trace width		4		Mils

7.5.4.6 Grounding

Each TMDS channel has its own shield pin and they should be grounded to provide a return current path for the TMDS signal.

7.5.5 PCIe Board Design and Layout Guidelines

The PCIe interface on the device provides support for a 5.0 Gbps lane with polarity inversion.

7.5.5.1 PCIe Connections and Interface Compliance

The PCIe interface on the device is compliant with the PCIe revision 3.0 specification. Please refer to the PCIe specifications for all connections that are described in it. Those recommendations are more descriptive and exhaustive than what is possible here.

The use of PCIe compatible bridges and switches is allowed for interfacing with more than one other processor or PCIe device.

7.5.5.1.1 Coupling Capacitors

AC coupling capacitors are required on the transmit data pair. Table 7-16 shows the requirements for these capacitors.

Table 7-16. PCIe AC Coupling Capacitors Requirements

PARAMETER	MIN	TYP	MAX	UNIT
PCIe AC coupling capacitor value	90	100	110	nF
PCIe AC coupling capacitor package size		0402	0603	EIA(1)(2)

(1) EIA LxW units, i.e., a 0402 is a 40x20 mils surface mount capacitor.

(2) The physical size of the capacitor should be as small as practical. Use the same size on both lines in each pair placed side by side.

7.5.5.1.2 Polarity Inversion

The PCIe specification requires polarity inversion support. This means for layout purposes, polarity is unimportant because each signal can change its polarity on die inside the chip. This means polarity within a lane is unimportant for layout.

7.5.5.2 Non-standard PCIe connections

The following sections contain suggestions for any PCIe connection that is NOT described in the official PCIe specification, such as an on-board Device to Device or Device to other PCIe compliant processor connection.

7.5.5.2.1 PCB Stackup Specifications

Table 7-17 shows the stackup and feature sizes required for these types of PCIe connections.

Table 7-17. PCIe PCB Stackup Specifications

PARAMETER	MIN	TYP	MAX	UNIT
Number of ground plane cuts allowed within PCIe routing region	-	-	0	Cuts
Number of layers between PCIe routing area and reference plane (1)	-	-	0	Layers
PCB Routing clearance		4		Mils
PCB Trace width		4		Mils

(1) A reference plane may be a ground plane or the power plane referencing the PCIe signals.

7.5.5.2.2 Routing Specifications

7.5.5.2.2.1 Impedance

The PCIe data signal traces must be routed to achieve 100-Ω (±10%) differential impedance and 60-Ω (±10%) single-ended impedance. The single-ended impedance is required because differential signals are extremely difficult to closely couple on PCBs and, therefore, single-ended impedance becomes important. These requirements are the same as those recommended in the PCIe Motherboard Checklist 1.0 document, available from PCI-SIG (www.pcisig.com).

These impedances are impacted by trace width, trace spacing, distance between signals and referencing planes, and dielectric material. Verify with a PCB design tool that the trace geometry for both data signal pairs result in as close to 100-Ω differential impedance and 60-Ω single-ended impedance as possible. For best accuracy, work with your PCB fabricator to ensure this impedance is met. See Table 7-18 below.

7.5.5.2.2.2 Differential Coupling

In general, closely coupled differential signal traces are not an advantage on PCBs. When differential signals are closely coupled, tight spacing and width control is necessary. Very small width and spacing variations affect impedance dramatically, so tight impedance control can be more problematic to maintain in production. For PCBs with very tight space limitations (which are usually small) this can work, but for most PCBs, the loosely coupled option is probably best.

Loosely coupled PCB differential signals make impedance control much easier. Wider traces and spacing make obstacle avoidance easier (because each trace is not so fixed in position relative to the other), and trace width variations don't affect impedance as much, therefore it's easier to maintain an accurate impedance over the length of the signal. For longer routes, the wider traces also show reduced skin effect and therefore often result in better signal integrity with a larger eye diagram opening.

Table 7-18 shows the routing specifications for the PCIe data signals.

Table 7-18. PCI-E Routing Specifications

PARAMETER	MIN	TYP	MAX	UNIT
PCIe signal trace length (device balls to PCIe connector)			4700(1)	Mils
Differential pair trace matching			5(2)	Mils
Number of stubs allowed on PCIe traces(3)			0	stubs
TX/RX pair differential impedance	90	100	110	Ω
TX/RX single-ended impedance	54	60	66	Ω
Pad size of vias on PCIe trace			25(4)	Mils
Hole size of vias on PCIe trace			14	Mils
Number of vias on each PCIe trace			0	Vias
PCIe differential pair to any other trace spacing	2×DS(5)			

(1) Beyond this, signal integrity may suffer.
(2) For example, RXP0 within 5 Mils of RXN0.
(3) Inline pads may be used for probing.
(4) 35-Mil antipad maximum recommended.
(5) DS = differential spacing of the PCIe traces.

Table 7-19. PCI-E Routing Recommendations

Item	Description	Reason
ESD part number	None	ESD suppression generally not used on PCIe

7.5.5.2.2.3 Pair Length Matching

Each signal in the differential pair should be matched to within 5 mils of its matching differential signal. Length matching should be done as close to the mismatch as possible.

7.5.5.3 LJCB_REFN/P Connections

A Common Refclk Rx Architecture is required to be used for the device PCIe interface. Specifically, two modes of Common Refclk Rx Architecture are supported:

- **External REFCLK Mode**: An common external 100MHz clock source is distributed to both the Device and the link partner
- **Output REFCLK Mode**: A 100MHz HCSL clock source is output by the device and used by the link partner

In **External REFCLK Mode**, a high-quality, low-jitter, differential HCSL 100MHz clock source compliant to the PCIe REFCLK AC Specifications should be provided on the Device's ljcb_clkn / ljcb_clkp inputs. Alternatively, an LVDS clock source can be used with the following additional requirements:

- External AC coupling capacitors described in Table 7-20 should be populated at the ljcb_clkn / ljcb_clkp inputs.
- All termination requirements (ex. parallel 100ohm termination) from the clock source manufacturer should be followed.

In **Output REFCLK Mode**, the 100MHz clock from the Device's DPLL_PCIE_REF should be output on the Device's ljcb_clkn / ljcb_clkp pins and used as the HCSL REFCLK by the link partner. External near-side termination to ground described in Table 7-21 is required on both of the ljcb_clkn / ljcb_clkp outputs in this mode.

Table 7-20. LJCB_REFN/P Requirements in External LVDS REFCLK Mode

PARAMETER	MIN	TYP	MAX	UNIT
ljcb_clkn / ljcb_clkp AC coupling capacitor value		100		nF
ljcb_clkn / ljcb_clkp AC coupling capacitor package size		0402	0603	EIA(1)(2)

(1) EIA LxW units, i.e., a 0402 is a 40x20 mils surface mount capacitor.
(2) The physical size of the capacitor should be as small as practical. Use the same size on both lines in each pair placed side by side.

Table 7-21. LJCB_REFN/P Requirements in Output REFCLK Mode

PARAMETER	MIN	TYP	MAX	UNIT
ljcb_clkn / ljcb_clkp near-side termination to ground value	47.5	50	52.5	Ohms

7.5.6 CSI2 Board Design and Routing Guidelines

The MIPI D-PHY signals include the CSI2_0 camera serial interfaces to or from the Device.

For more information regarding the MIPI-PHY signals and corresponding balls, see Table 4-5, *CSI2 Signal Descriptions*.

For more information, you can also see the MIPI D-PHY specification v1-01-00_r0-03 (specifically the Interconnect and Lane Configuration and Annex B Interconnect Design Guidelines chapters).

In the next section, the PCB guidelines of the following differential interfaces are presented:

- CSI2_0 MIPI CSI-2 at 1.5 Gbps

Table 7-22 lists the MIPI D-PHY interface signals in the Device.

Table 7-22. MIPI D-PHY Interface Signals in the Device

SIGNAL NAME	BOTTOM BALL	SIGNAL NAME	BOTTOM BALL
csi2_0_dx0	AC1	csi2_0_dy0	AB2
csi2_0_dx1	AD1	csi2_0_dy1	AC2
csi2_0_dx2	AE2	csi2_0_dy2	AD2

7.5.6.1 CSI2_0 MIPI CSI-2 (1.5 Gbps)

7.5.6.1.1 General Guidelines

The general guidelines for the PCB differential lines are:

- Differential trace impedance Z_0 = 100 Ω (minimum = 85 Ω, maximum = 115 Ω)
- Total conductor length from the Device package pins to the peripheral device package pins is 25 to 30 cm with common FR4 PCB and flex materials.

NOTE

Longer interconnect length can be supported at the expense of detailed simulations of the complete link including driver and receiver models.

The general rule of thumb for the space S = 2 × W is not designated (see Figure 7-19, *Guard Illustration*). It is because although the S = 2 × W rule is a good rule of thumb, it is not always the best solution. The electrical performance will be checked with the frequency-domain specification. Even though the designer does not follow the S = 2 × W rule, the differential lines are ok if the lines satisfy the frequency-domain specification.

Because the MIPI signals are used for low-power, single-ended signaling in addition to their high-speed differential implementation, the pairs must be loosely coupled.

7.5.6.1.2 Length Mismatch Guidelines

7.5.6.1.2.1 CSI2_0 MIPI CSI-2 (1.5 Gbps)

The guidelines of the length mismatch for CSI-2 are presented in Table 7-23.

Table 7-23. Length Mismatch Guidelines for CSI-2 (1.5 Gbps)

PARAMETER	TYPICAL VALUE	UNIT
Operating speed	1500	Mbps
UI (bit time)	667	ps
Intralane skew	Have to satisfy mode-conversion S parameters(1)	
Interlane skew (UI / 50)	13.34	ps
PCB lane-to-lane skew (0.1 UI)	66.7	ps

(1) sdc12, scd21, scd12, sdc21, scd11, sdc11, scd22, and sdc22

7.5.6.1.3 Frequency-domain Specification Guidelines

After the PCB design is finished, the S-parameters of the PCB differential lines will be extracted with a 3D Maxwell Equation Solver such as the high-frequency structure simulator (HFSS) or equivalent, and compared to the frequency-domain specification as defined in the section 7 of the MIPI Alliance Specification for D-PHY Version v1-01-00_r0-03.

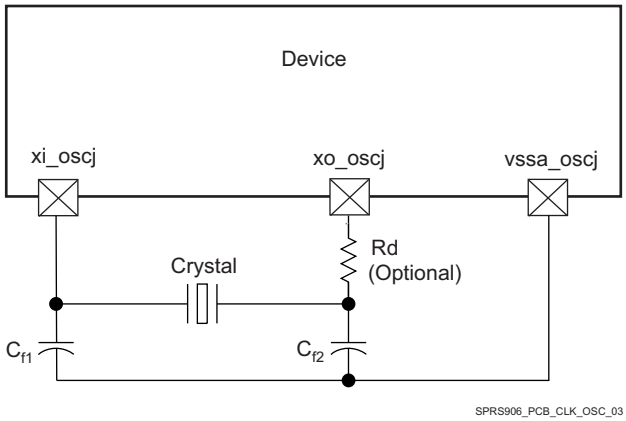
If the PCB lines satisfy the frequency-domain specification, the design is finished. Otherwise, the design needs to be improved.

7.6 Clock Routing Guidelines

7.6.1 Oscillator Ground Connection

Although the impedance of a ground plane is low it is, of course, not zero. Therefore, any noise current in the ground plane causes a voltage drop in the ground.

Figure 7-36 shows the grounding scheme for high-frequency clock.



(1) j in *_osc = 0 or 1

Figure 7-36. Grounding Scheme for High-Frequency Clock

7.7 DDR3 Board Design and Layout Guidelines

7.7.1 DDR3 General Board Layout Guidelines

To help ensure good signaling performance, consider the following board design guidelines:

- Avoid crossing splits in the power plane.
- Minimize Vref noise.
- Use the widest trace that is practical between decoupling capacitors and memory module.
- Maintain a single reference.
- Minimize ISI by keeping impedances matched.
- Minimize crosstalk by isolating sensitive bits, such as strobes, and avoiding return path discontinuities.
- Use proper low-pass filtering on the Vref pins.
- Keep the stub length as short as possible.
- Add additional spacing for on-clock and strobe nets to eliminate crosstalk.
- Maintain a common ground reference for all bypass and decoupling capacitors.
- Take into account the differences in propagation delays between microstrip and stripline nets when evaluating timing constraints.

7.7.2 DDR3 Board Design and Layout Guidelines

7.7.2.1 Board Designs

TI only supports board designs using DDR3 memory that follow the guidelines in this document. The switching characteristics and timing diagram for the DDR3 memory controller are shown in Table 7-24 and Figure 7-37.

Table 7-24. Switching Characteristics Over Recommended Operating Conditions for DDR3 Memory Controller

NO.	PARAMETER		MIN	MAX	UNIT
1	$t_{c(DDR\_CLK)}$	Cycle time, DDR_CLK	1.5	2.5(1)	ns

(1) This is the absolute maximum the clock period can be. Actual maximum clock period may be limited by DDR3 speed grade and operating frequency (see the DDR3 memory device data sheet).

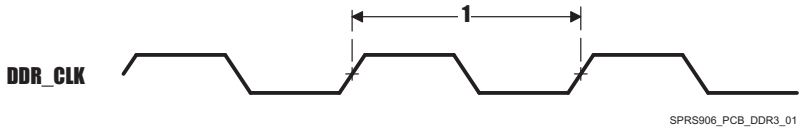


Figure 7-37. DDR3 Memory Controller Clock Timing

7.7.2.2 DDR3 EMIF

The processor contains one DDR3 EMIF with one chip select.

7.7.2.3 DDR3 Device Combinations

Because there are several possible combinations of device counts and single- or dual-side mounting, Table 7-25 summarizes the supported device configurations.

Table 7-25. Supported DDR3 Device Combinations

NUMBER OF DDR3 DEVICES	DDR3 DATA DEVICE WIDTH (BITS)	MIRRORED?	DDR3 EMIF WIDTH (BITS)
1	16	N	16
2	8	Y(1)	16
2	16	N	32
2	16	Y(1)	32
3	16	N(3)(4)	32
4	8	N	32
4	8	Y(2)	32
5	8	N (3)(4)	32

(1) Two DDR3 devices are mirrored when one device is placed on the top of the board and the second device is placed on the bottom of the board.

(2) This is two mirrored pairs of DDR3 devices.

(3) Three or five DDR3 device combination is not available on this device, but combination types are retained for consistency with the DRA7xx family of devices.

(4) The DDR memory connected to the DDR ECC bus does NOT need to be the same part number as the DDR memories connected to the DDR data bus. However, some constraints do apply. When selecting a memory for the DDR ECC bus, the following restrictions must be adhered to as compared to the DDR memories on the data bus:

- Match the same DDR3 speed grade
- Have an equal number of internal banks
- Have an equal number of columns
- Have a greater or equal number of rows

7.7.2.4 DDR3 Interface Schematic

7.7.2.4.1 32-Bit DDR3 Interface

The DDR3 interface schematic varies, depending upon the width of the DDR3 devices used and the width of the bus used (16 or 32 bits). General connectivity is straightforward and very similar. 16-bit DDR devices look like two 8-bit devices. Figure 7-38 and Figure 7-39 show the schematic connections for 32-bit interfaces using x16 devices.

7.7.2.4.2 16-Bit DDR3 Interface

Note that the 16-bit wide interface schematic is practically identical to the 32-bit interface (see Figure 7-38 and Figure 7-39); only the high-word DDR memories are removed and the unused DQS inputs are tied off.

When not using all or part of a DDR interface, the proper method of handling the unused pins is to tie off the ddrx_dqs*i* pins to ground via a 1k-Ω resistor and to tie off the ddrx_dqsn*i* pins to the corresponding vdds_ddr*x* supply via a 1k-Ω resistor. This needs to be done for each byte not used. Although these signals have internal pullups and pulldowns, external pullups and pulldowns provide additional protection against external electrical noise causing activity on the signals.

The vdds_ddr*x* and ddrx_vref0 power supply pins need to be connected to their respective power supplies even if ddrx is not being used. All other DDR interface pins can be left unconnected. Note that the supported modes for use of the DDR EMIF are 32-bits wide, 16-bits wide, or not used.

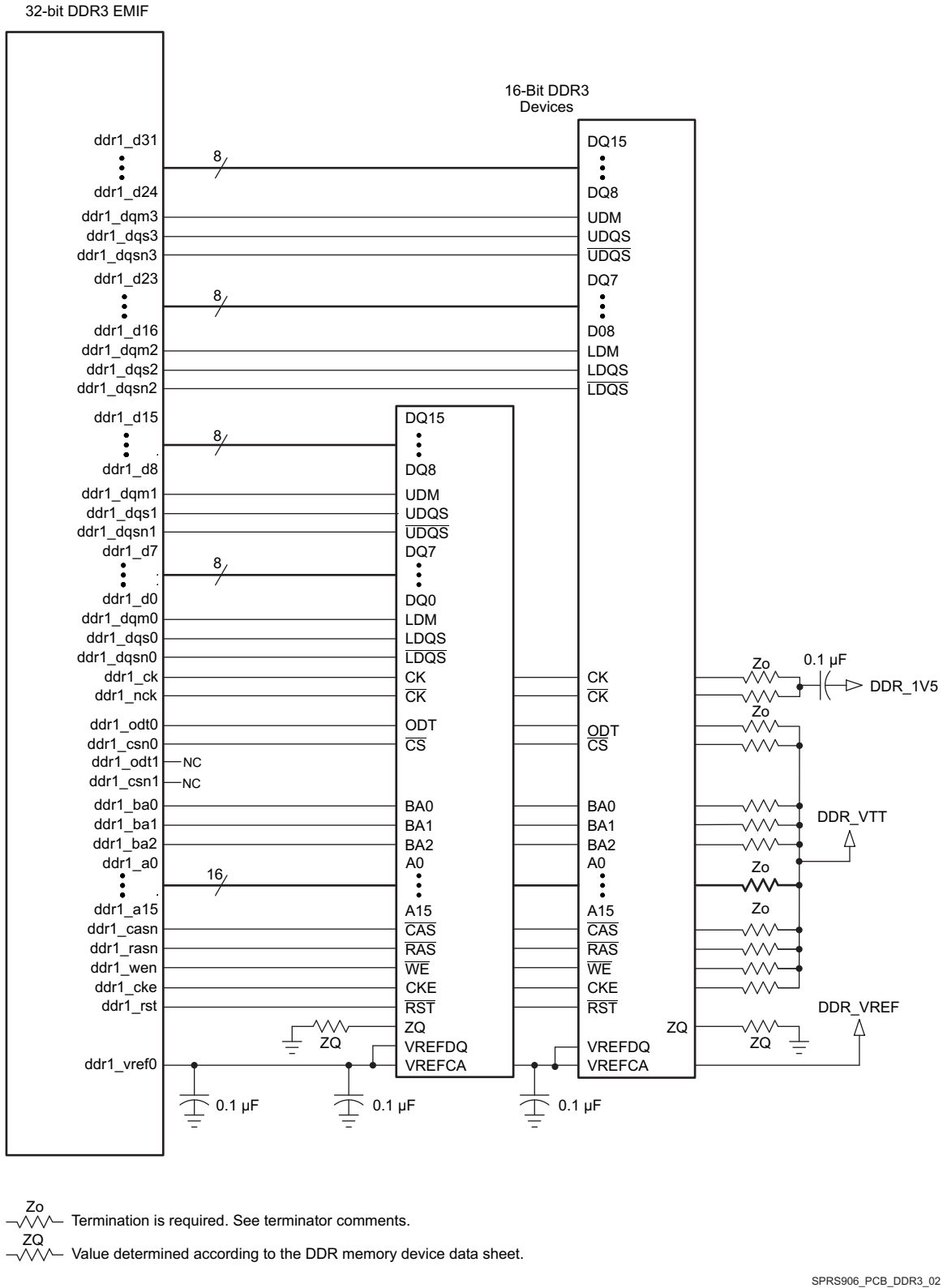


Figure 7-38. 32-Bit, One-Bank DDR3 Interface Schematic Using Two 16-Bit DDR3 Devices

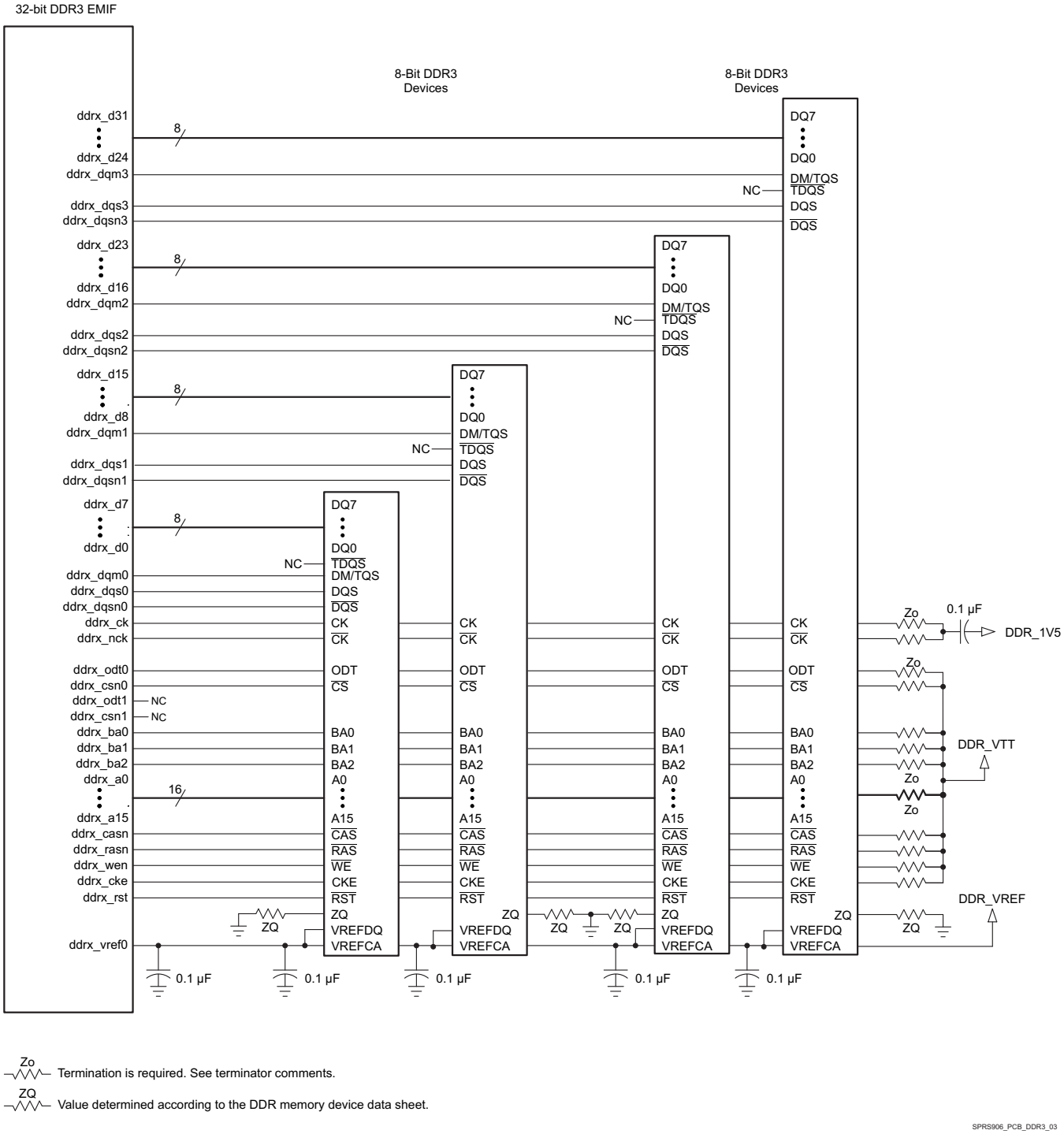


Figure 7-39. 32-Bit, One-Bank DDR3 Interface Schematic Using Four 8-Bit DDR3 Devices

7.7.2.5 Compatible JEDEC DDR3 Devices

Table 7-26 shows the parameters of the JEDEC DDR3 devices that are compatible with this interface. Generally, the DDR3 interface is compatible with DDR3-1333 devices in the x8 or x16 widths.

Table 7-26. Compatible JEDEC DDR3 Devices (Per Interface)

NO.	PARAMETER	CONDITION	MIN	MAX	UNIT
1	JEDEC DDR3 device speed grade(1)	DDR clock rate = 400MHz	DDR3-800	DDR3-1600	
		400MHz< DDR clock rate ≤ 533MHz	DDR3-1066	DDR3-1600	
		533MHz< DDR clock rate ≤ 667MHz	DDR3-1333	DDR3-1600	
2	JEDEC DDR3 device bit width		x8	x16	Bits
3	JEDEC DDR3 device count(2)		2	4	Devices

(1) Refer to Table 7-24 Switching Characteristics Over Recommended Operating Conditions for DDR3 Memory Controller for the range of supported DDR clock rates.

(2) For valid DDR3 device configurations and device counts, see Section 7.7.2.4, Figure 7-38, and Figure 7-39.

7.7.2.6 PCB Stackup

The minimum stackup for routing the DDR3 interface is a six-layer stack up as shown in Table 7-27. Additional layers may be added to the PCB stackup to accommodate other circuitry, enhance SI/EMI performance, or to reduce the size of the PCB footprint. Complete stackup specifications are provided in Table 7-28.

Table 7-27. Six-Layer PCB Stackup Suggestion

LAYER	TYPE	DESCRIPTION
1	Signal	Top routing mostly vertical
2	Plane	Ground
3	Plane	Split power plane
4	Plane	Split power plane or Internal routing
5	Plane	Ground
6	Signal	Bottom routing mostly horizontal

Table 7-28. PCB Stackup Specifications

NO.	PARAMETER	MIN	TYP	MAX	UNIT
PS1	PCB routing/plane layers	6			
PS2	Signal routing layers	3			
PS3	Full ground reference layers under DDR3 routing region(1)	1			
PS4	Full 1.5-V power reference layers under the DDR3 routing region(1)	1			
PS5	Number of reference plane cuts allowed within DDR routing region(2)			0	
PS6	Number of layers between DDR3 routing layer and reference plane(3)			0	
PS7	PCB routing feature size		4		Mils
PS8	PCB trace width, w		4		Mils
PS9	Single-ended impedance, Zo	50		75	Ω
PS10	Impedance control(5)	Z-5	Z	Z+5	Ω

(1) Ground reference layers are preferred over power reference layers. Be sure to include bypass caps to accommodate reference layer return current as the trace routes switch routing layers.

(2) No traces should cross reference plane cuts within the DDR routing region. High-speed signal traces crossing reference plane cuts create large return current paths which can lead to excessive crosstalk and EMI radiation.

(3) Reference planes are to be directly adjacent to the signal plane to minimize the size of the return current loop.

(4) An 18-mil pad assumes Via Channel is the most economical BGA escape. A 20-mil pad may be used if additional layers are available for power routing. An 18-mil pad is required for minimum layer count escape.

(5) Z is the nominal singled-ended impedance selected for the PCB specified by PS9.

7.7.2.7 Placement

Figure 7-40 shows the required placement for the processor as well as the DDR3 devices. The dimensions for this figure are defined in Table 7-29. The placement does not restrict the side of the PCB on which the devices are mounted. The ultimate purpose of the placement is to limit the maximum trace lengths and allow for proper routing space. For a 16-bit DDR memory system, the high-word DDR3 devices are omitted from the placement.

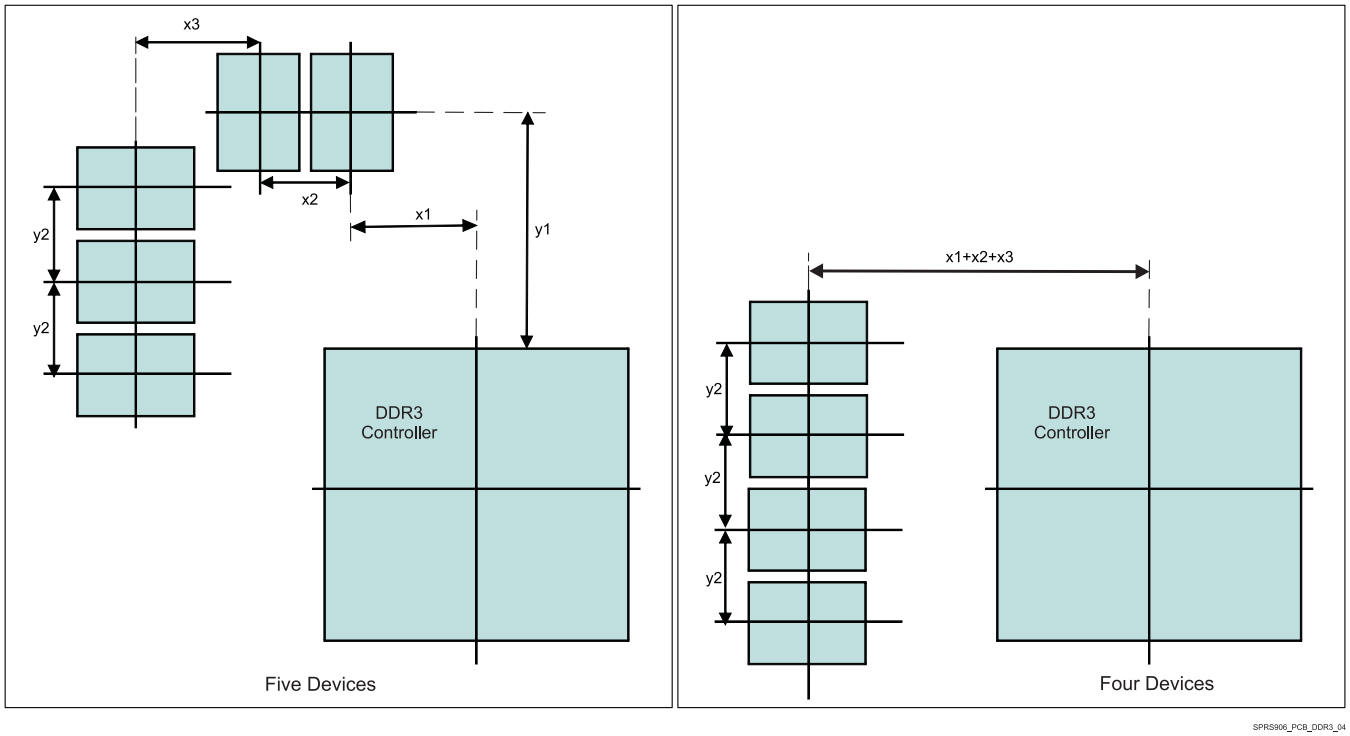


Figure 7-40. Placement Specifications

Table 7-29. Placement Specifications DDR3

NO.	PARAMETER	MIN	MAX	UNIT
KOD31	X1		500	Mils
KOD32	X2		600	Mils
KOD33	X3		600	Mils
KOD34	Y1		1800	Mils
KOD35	Y2		600	Mils
KOD36	DDR3 keepout region (1)			
KOD37	Clearance from non-DDR3 signal to DDR3 keepout region (2) (3)	4		W

(1) DDR3 keepout region to encompass entire DDR3 routing area.

(2) Non-DDR3 signals allowed within DDR3 keepout region provided they are separated from DDR3 routing layers by a ground plane.

(3) If a device has more than one DDR controller, the signals from the other controller(s) are considered non-DDR3 and should be separated by this specification.

7.7.2.8 DDR3 Keepout Region

The region of the PCB used for DDR3 circuitry must be isolated from other signals. The DDR3 keepout region is defined for this purpose and is shown in Figure 7-41. The size of this region varies with the placement and DDR routing. Additional clearances required for the keepout region are shown in Table 7-29. Non-DDR3 signals should not be routed on the DDR signal layers within the DDR3 keepout region. Non-DDR3 signals may be routed in the region, provided they are routed on layers separated from the DDR signal layers by a ground layer. No breaks should be allowed in the reference ground layers in this region. In addition, the 1.5-V DDR3 power plane should cover the entire keepout region. Also note that the two signals from the DDR3 controller should be separated from each other by the specification in Table 7-29 (see KOD37).

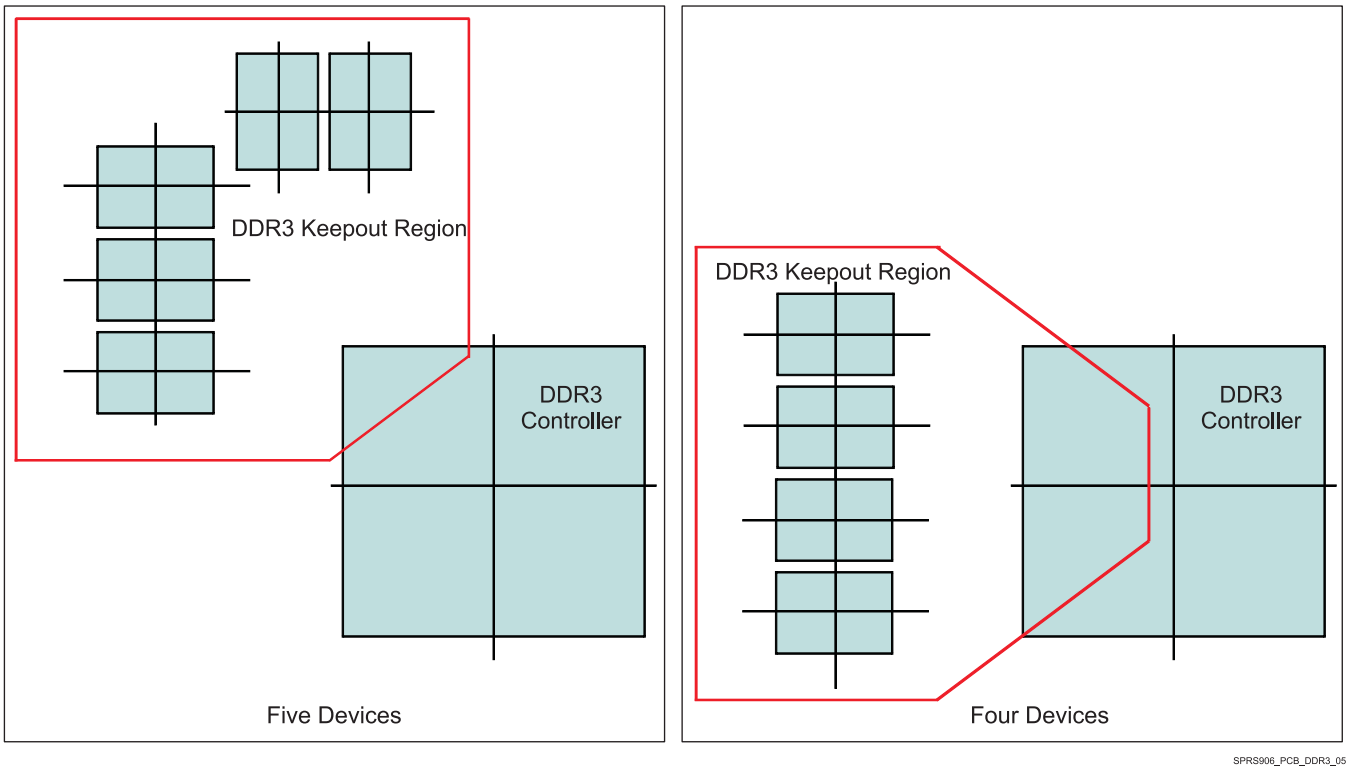


Figure 7-41. DDR3 Keepout Region

7.7.2.9 Bulk Bypass Capacitors

Bulk bypass capacitors are required for moderate speed bypassing of the DDR3 and other circuitry. Table 7-30 contains the minimum numbers and capacitance required for the bulk bypass capacitors. Note that this table only covers the bypass needs of the DDR3 controllers and DDR3 devices. Additional bulk bypass capacitance may be needed for other circuitry.

Table 7-30. Bulk Bypass Capacitors

NO.	PARAMETER	MIN	MAX	UNIT
1	vdds_ddr*x* bulk bypass capacitor count(1)	1		Devices
2	vdds_ddr*x* bulk bypass total capacitance	22		µF

(1) These devices should be placed near the devices they are bypassing, but preference should be given to the placement of the high-speed (HS) bypass capacitors and DDR3 signal routing.

7.7.2.10 High-Speed Bypass Capacitors

High-speed (HS) bypass capacitors are critcal for proper DDR3 interface operation. It is particularly important to minimize the parasitic series inductance of the HS bypass capacitors, processor/DDR power, and processor/DDR ground connections. Table 7-31 contains the specification for the HS bypass capacitors as well as for the power connections on the PCB. Generally speaking, it is good to:

1. Fit as many HS bypass capacitors as possible.
2. Minimize the distance from the bypass cap to the pins/balls being bypassed.
3. Use the smallest physical sized capacitors possible with the highest capacitance readily available.
4. Connect the bypass capacitor pads to their vias using the widest traces possible and using the largest hole size via possible.
5. Minimize via sharing. Note the limites on via sharing shown in Table 7-31.

Table 7-31. High-Speed Bypass Capacitors

NO.	PARAMETER	MIN	TYP	MAX	UNIT
1	HS bypass capacitor package size(1)		0201	0402	10 Mils
2	Distance, HS bypass capacitor to processor being bypassed(2)(3)(4)			400	Mils
3	Processor HS bypass capacitor count per vdds_ddrx rail(12)	See Table 7-3 and (11)			Devices
4	Processor HS bypass capacitor total capacitance per vdds_ddrx rail(12)	See Table 7-3 and (11)			µF
5	Number of connection vias for each device power/ground ball(5)				Vias
6	Trace length from device power/ground ball to connection via(2)		35	70	Mils
7	Distance, HS bypass capacitor to DDR device being bypassed(6)			150	Mils
8	DDR3 device HS bypass capacitor count(7)	12			Devices
9	DDR3 device HS bypass capacitor total capacitance(7)	0.85			µF
10	Number of connection vias for each HS capacitor(8)(9)	2			Vias
11	Trace length from bypass capacitor connect to connection via(2)(9)		35	100	Mils
12	Number of connection vias for each DDR3 device power/ground ball(10)	1			Vias
13	Trace length from DDR3 device power/ground ball to connection via(2)(8)		35	60	Mils

(1) LxW, 10-mil units, that is, a 0402 is a 40x20-mil surface-mount capacitor.

(2) Closer/shorter is better.

(3) Measured from the nearest processor power/ground ball to the center of the capacitor package.

(4) Three of these capacitors should be located underneath the processor, between the cluster of DDR_1V5 balls and ground balls, between the DDR interfaces on the package.

(5) See the Via Channel™ escape for the processor package.

(6) Measured from the DDR3 device power/ground ball to the center of the capacitor package.

(7) Per DDR3 device.

(8) An additional HS bypass capacitor can share the connection vias only if it is mounted on the opposite side of the board. No sharing of vias is permitted on the same side of the board.

(9) An HS bypass capacitor may share a via with a DDR device mounted on the same side of the PCB. A wide trace should be used for the connection and the length from the capacitor pad to the DDR device pad should be less than 150 mils.

(10) Up to a total of two pairs of DDR power/ground balls may share a via.

(11) The capacitor recommendations in this data manual reflect only the needs of this processor. Please see the memory vendor's guidelines for determining the appropriate decoupling capacitor arrangement for the memory device itself.

(12) For more information, see Section 7.3, *Core Power Domains*.

7.7.2.10.1 Return Current Bypass Capacitors

Use additional bypass capacitors if the return current reference plane changes due to DDR3 signals hopping from one signal layer to another. The bypass capacitor here provides a path for the return current to hop planes along with the signal. As many of these return current bypass capacitors should be used as possible. Because these are returns for signal current, the signal via size may be used for these capacitors.

7.7.2.11 Net Classes

Table 7-32 lists the clock net classes for the DDR3 interface. Table 7-33 lists the signal net classes, and associated clock net classes, for signals in the DDR3 interface. These net classes are used for the termination and routing rules that follow.

Table 7-32. Clock Net Class Definitions

CLOCK NET CLASS	processor PIN NAMES
CK	ddrx_ck/ddrx_nck
DQS0	ddrx_dqs0 / ddrx_dqsn0
DQS1	ddrx_dqs1 / ddrx_dqsn1
DQS2(1)	ddrx_dqs2 / ddrx_dqsn2
DQS3(1)	ddrx_dqs3 / ddrx_dqsn3

(1) Only used on 32-bit wide DDR3 memory systems.

Table 7-33. Signal Net Class Definitions

SIGNAL NET CLASS	ASSOCIATED CLOCK NET CLASS	processor PIN NAMES
ADDR_CTRL	CK	ddr*x*_ba[2:0], ddr*x*_a[14:0], ddr*x*_csn*j*, ddr*x*_casn, ddr*x*_rasn, ddr*x*_wen, ddr*x*_cke, ddr*x*_odt*i*
DQ0	DQS0	ddr*x*_d[7:0], ddr*x*_dqm0
DQ1	DQS1	ddr*x*_d[15:8], ddr*x*_dqm1
DQ2(1)	DQS2	ddr*x*_d[23:16], ddr*x*_dqm2
DQ3(1)	DQS3	ddr*x*_d[31:24], ddr*x*_dqm3

(1) Only used on 32-bit wide DDR3 memory systems.

7.7.2.12 DDR3 Signal Termination

Signal terminators are required for the CK and ADDR_CTRL net classes. The data lines are terminated by ODT and, thus, the PCB traces should be unterminated. Detailed termination specifications are covered in the routing rules in the following sections.

7.7.2.13 VREF_DDR Routing

ddrx_vref0 (VREF) is used as a reference by the input buffers of the DDR3 memories as well as the processor. VREF is intended to be half the DDR3 power supply voltage and is typically generated with the DDR3 VDDS and VTT power supply. It should be routed as a nominal 20-mil wide trace with 0.1 µF bypass capacitors near each device connection. Narrowing of VREF is allowed to accommodate routing congestion.

7.7.2.14 VTT

Like VREF, the nominal value of the VTT supply is half the DDR3 supply voltage. Unlike VREF, VTT is expected to source and sink current, specifically the termination current for the ADDR_CTRL net class Thevinen terminators. VTT is needed at the end of the address bus and it should be routed as a power sub-plane. VTT should be bypassed near the terminator resistors.

7.7.2.15 CK and ADDR_CTRL Topologies and Routing Definition

The CK and ADDR_CTRL net classes are routed in a fly-by topology. They are routed in a similar manner and are length matched to minimize skew between them. CK is a bit more complicated because it runs at a higher transition rate and is differential. The following subsections show the topology and routing for various DDR3 configurations for CK and ADDR_CTRL. The figures in the following subsections define the terms for the routing specification detailed in Table 7-34. Balanced-T routing is not recommended.

7.7.2.15.1 Four DDR3 Devices

Four DDR3 devices are supported on the DDR EMIF consisting of four x8 DDR3 devices arranged as one bank (CS). These four devices may be mounted on a single side of the PCB, or may be mirrored in two pairs to save board space at a cost of increased routing complexity and parts on the backside of the PCB.

7.7.2.15.1.1 CK and ADDR_CTRL Topologies, Four DDR3 Devices

Figure 7-42 shows the topology of the CK net classes and Figure 7-43 shows the topology for the corresponding ADDR_CTRL net classes.

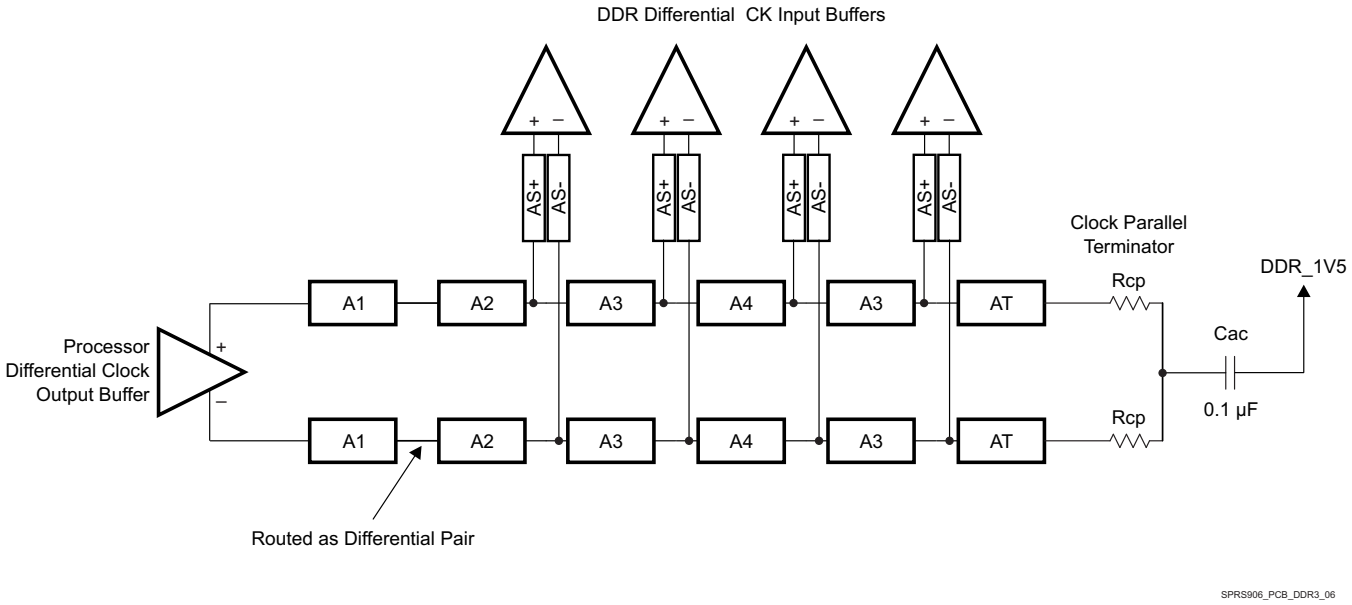


Figure 7-42. CK Topology for Four x8 DDR3 Devices

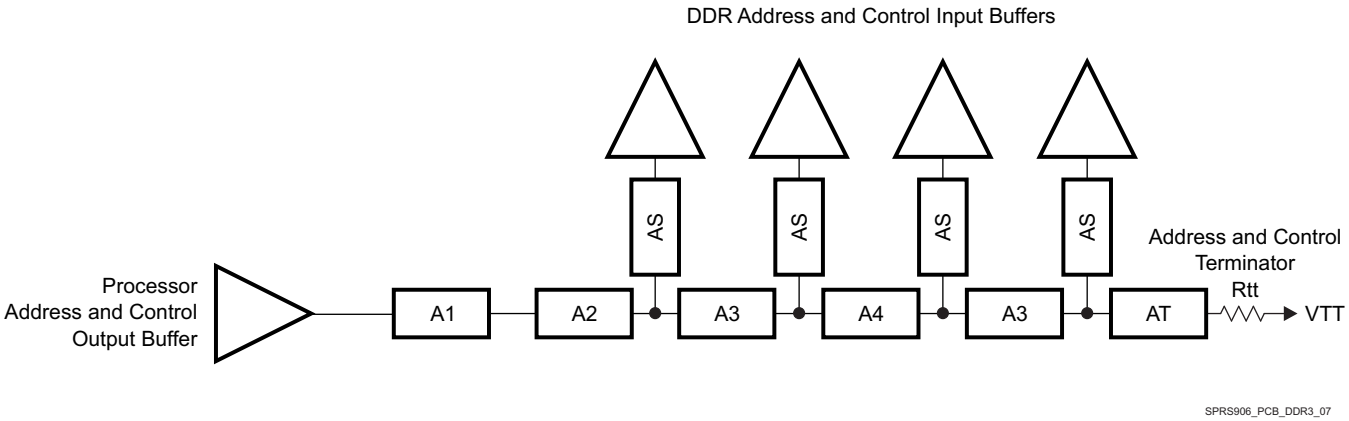


Figure 7-43. ADDR_CTRL Topology for Four x8 DDR3 Devices

7.7.2.15.1.2 CK and ADDR_CTRL Routing, Four DDR3 Devices

Figure 7-44 shows the CK routing for four DDR3 devices placed on the same side of the PCB. Figure 7-45 shows the corresponding ADDR_CTRL routing.

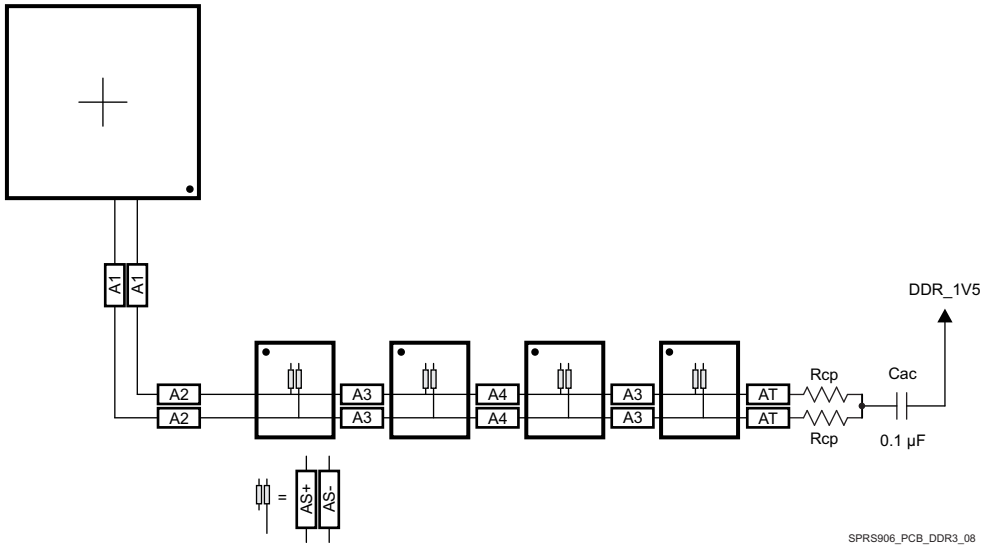


Figure 7-44. CK Routing for Four Single-Side DDR3 Devices

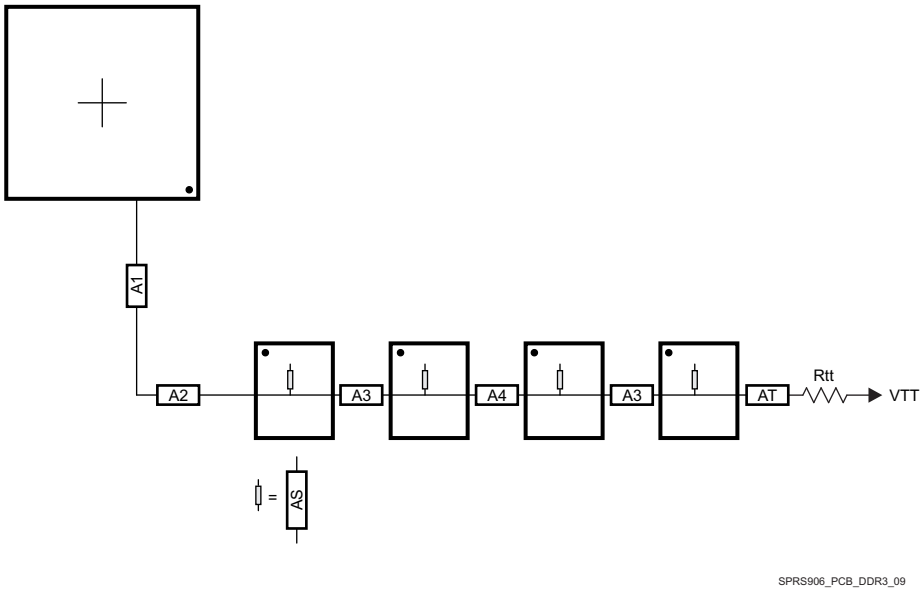


Figure 7-45. ADDR_CTRL Routing for Four Single-Side DDR3 Devices

To save PCB space, the four DDR3 memories may be mounted as two mirrored pairs at a cost of increased routing and assembly complexity. Figure 7-46 and Figure 7-47 show the routing for CK and ADDR_CTRL, respectively, for four DDR3 devices mirrored in a two-pair configuration.

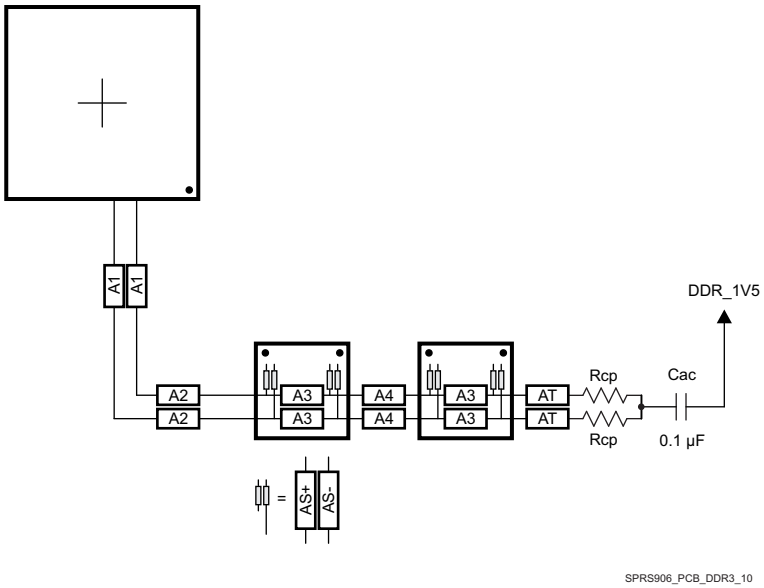


Figure 7-46. CK Routing for Four Mirrored DDR3 Devices

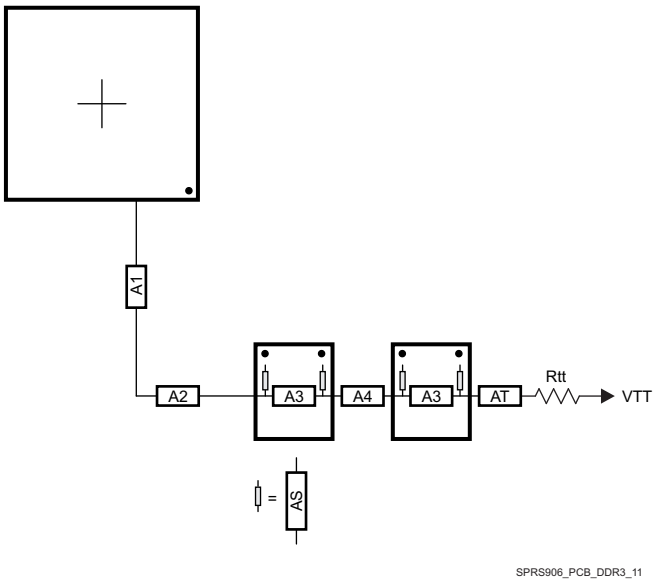


Figure 7-47. ADDR_CTRL Routing for Four Mirrored DDR3 Devices

7.7.2.15.2 Two DDR3 Devices

Two DDR3 devices are supported on the DDR EMIF consisting of two x8 DDR3 devices arranged as one bank (CS), 16 bits wide, or two x16 DDR3 devices arranged as one bank (CS), 32 bits wide. These two devices may be mounted on a single side of the PCB, or may be mirrored in a pair to save board space at a cost of increased routing complexity and parts on the backside of the PCB.

7.7.2.15.2.1 CK and ADDR_CTRL Topologies, Two DDR3 Devices

Figure 7-48 shows the topology of the CK net classes and Figure 7-49 shows the topology for the corresponding ADDR_CTRL net classes.

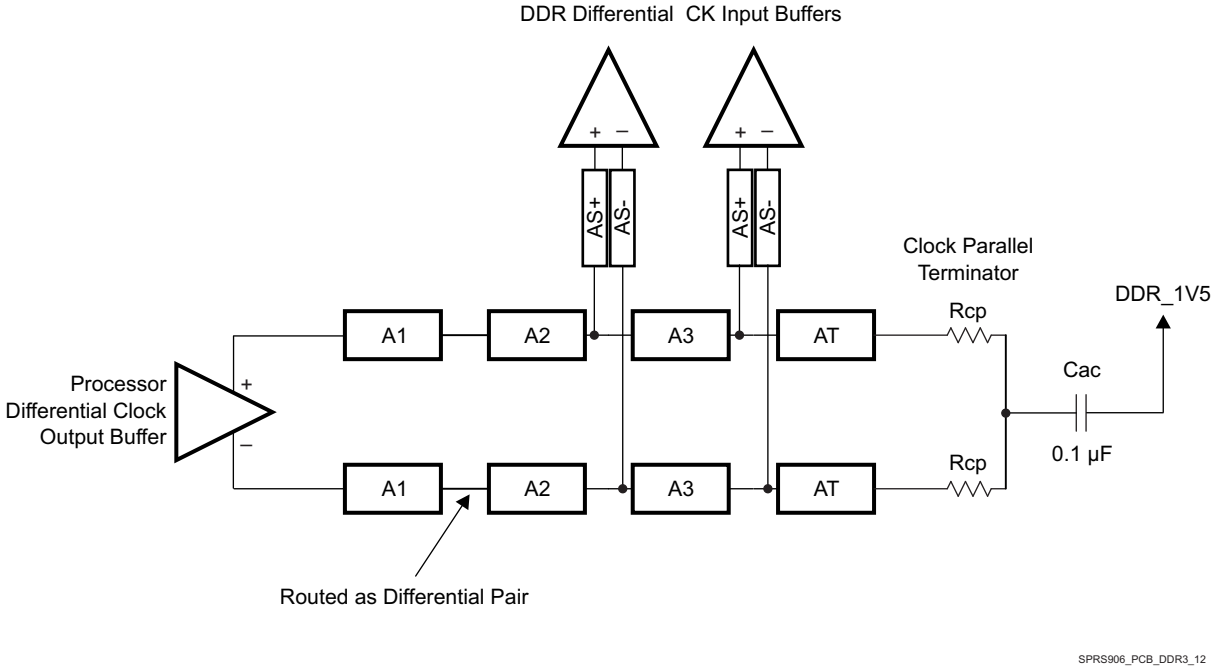


Figure 7-48. CK Topology for Two DDR3 Devices

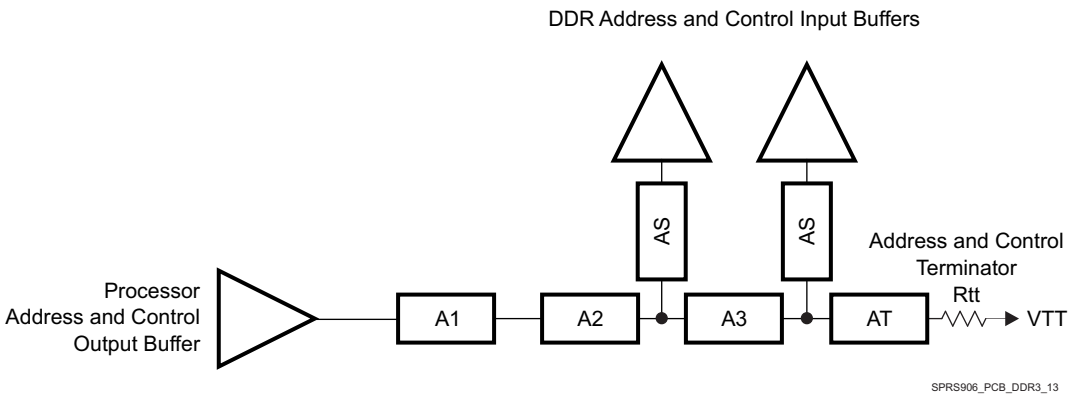


Figure 7-49. ADDR_CTRL Topology for Two DDR3 Devices

7.7.2.15.2.2 CK and ADDR_CTRL Routing, Two DDR3 Devices

Figure 7-50 shows the CK routing for two DDR3 devices placed on the same side of the PCB. Figure 7-51 shows the corresponding ADDR_CTRL routing.

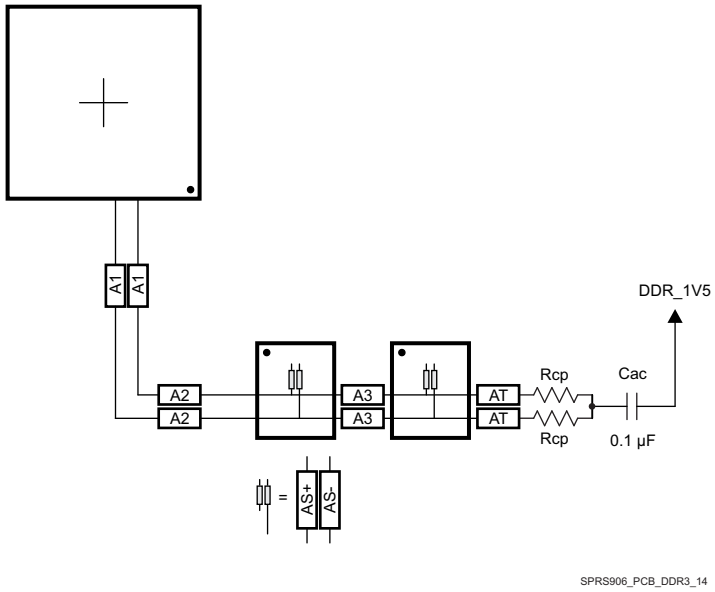


Figure 7-50. CK Routing for Two Single-Side DDR3 Devices

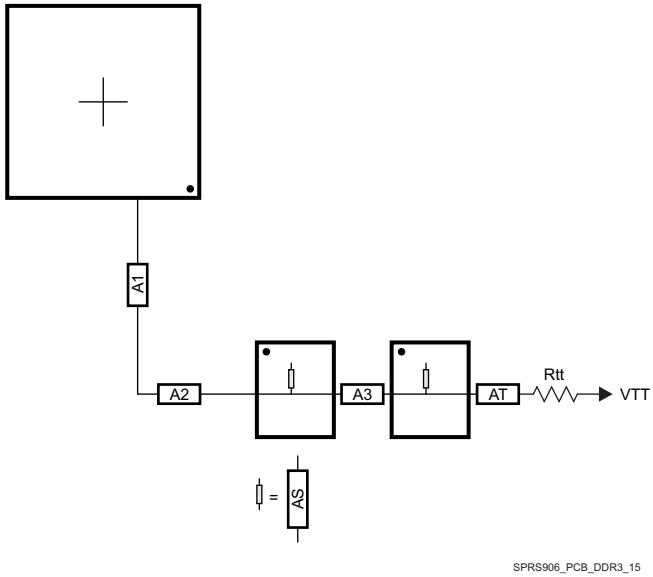


Figure 7-51. ADDR_CTRL Routing for Two Single-Side DDR3 Devices

To save PCB space, the two DDR3 memories may be mounted as a mirrored pair at a cost of increased routing and assembly complexity. Figure 7-52 and Figure 7-53 show the routing for CK and ADDR_CTRL, respectively, for two DDR3 devices mirrored in a single-pair configuration.

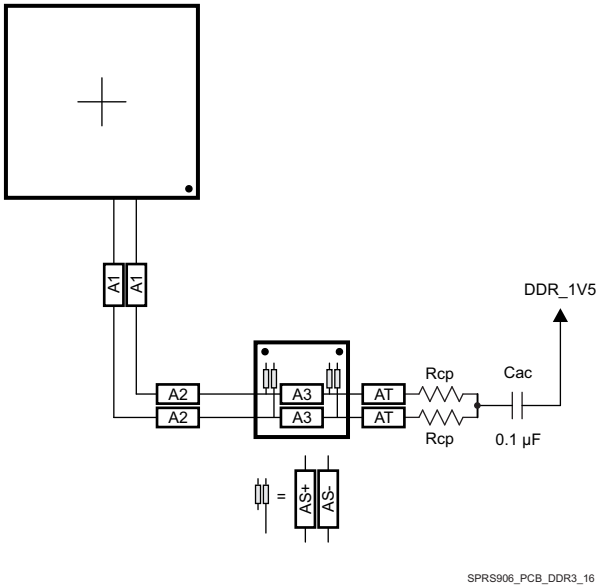


Figure 7-52. CK Routing for Two Mirrored DDR3 Devices

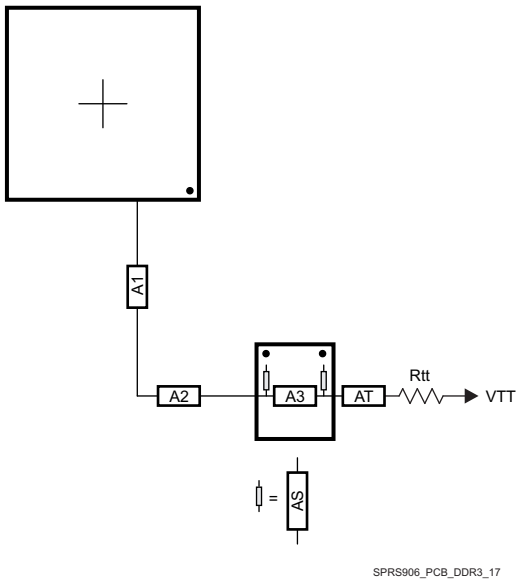


Figure 7-53. ADDR_CTRL Routing for Two Mirrored DDR3 Devices

7.7.2.15.3 One DDR3 Device

A single DDR3 device is supported on the DDR EMIF consisting of one x16 DDR3 device arranged as one bank (CS), 16 bits wide.

7.7.2.15.3.1 CK and ADDR_CTRL Topologies, One DDR3 Device

Figure 7-54 shows the topology of the CK net classes and Figure 7-55 shows the topology for the corresponding ADDR_CTRL net classes.

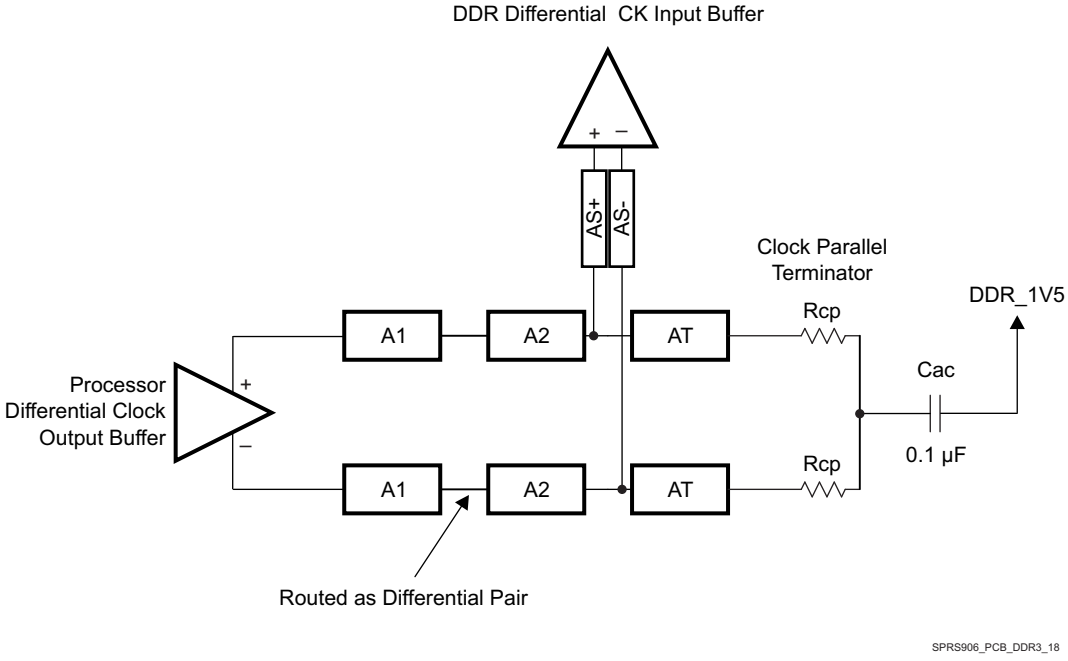


Figure 7-54. CK Topology for One DDR3 Device

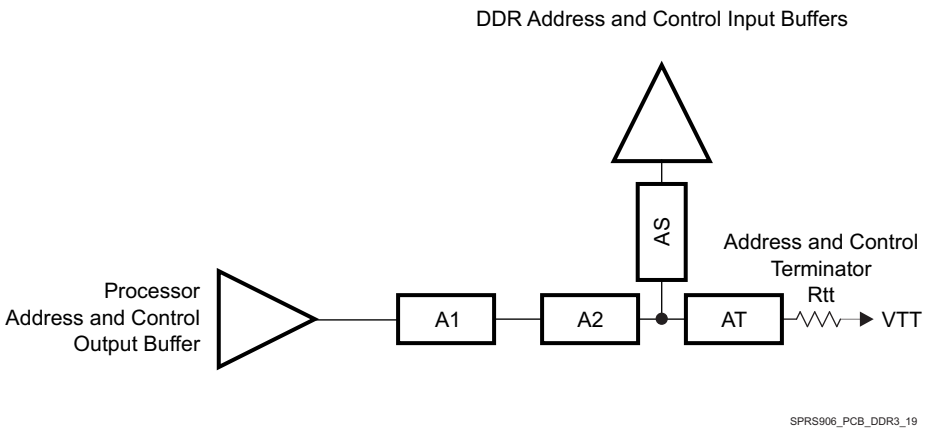


Figure 7-55. ADDR_CTRL Topology for One DDR3 Device

7.7.2.15.3.2 CK and ADDR/CTRL Routing, One DDR3 Device

Figure 7-56 shows the CK routing for one DDR3 device placed on the same side of the PCB. Figure 7-57 shows the corresponding ADDR_CTRL routing.

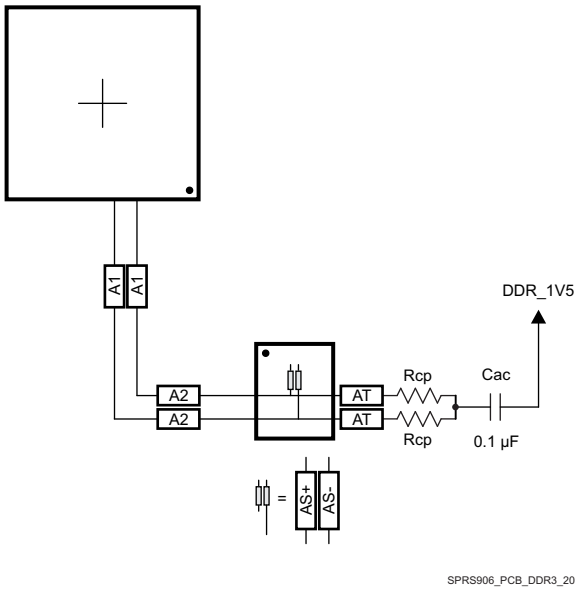


Figure 7-56. CK Routing for One DDR3 Device

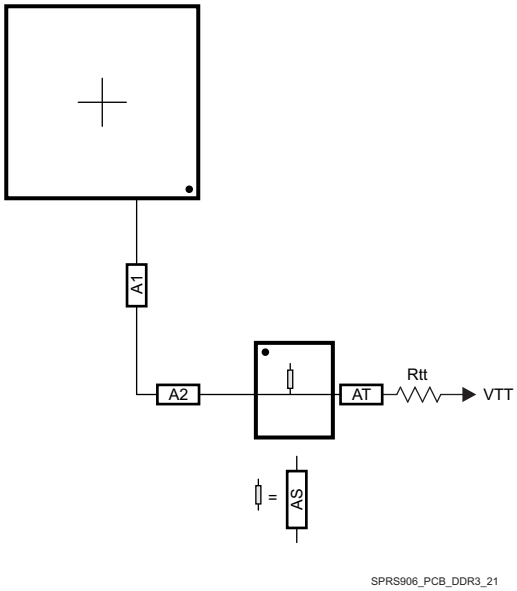


Figure 7-57. ADDR_CTRL Routing for One DDR3 Device

7.7.2.16 Data Topologies and Routing Definition

No matter the number of DDR3 devices used, the data line topology is always point to point, so its definition is simple.

Care should be taken to minimize layer transitions during routing. If a layer transition is necessary, it is better to transition to a layer using the same reference plane. If this cannot be accommodated, ensure there are nearby ground vias to allow the return currents to transition between reference planes if both reference planes are ground or vdds_ddr. Ensure there are nearby bypass capacitors to allow the return currents to transition between reference planes if one of the reference planes is ground. The goal is to minimize the size of the return current loops.

7.7.2.16.1 DQS and DQ/DM Topologies, Any Number of Allowed DDR3 Devices

DQS lines are point-to-point differential, and DQ/DM lines are point-to-point singled ended. Figure 7-58 and Figure 7-59 show these topologies.

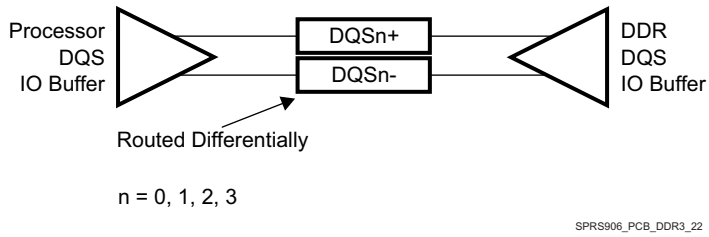


Figure 7-58. DQS Topology

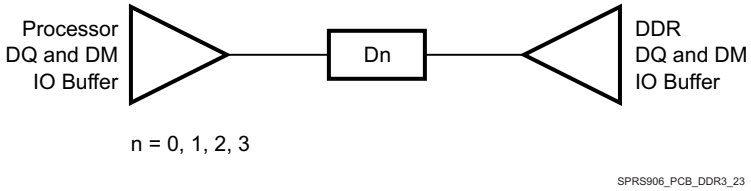


Figure 7-59. DQ/DM Topology

7.7.2.16.2 DQS and DQ/DM Routing, Any Number of Allowed DDR3 Devices

Figure 7-60 and Figure 7-61 show the DQS and DQ/DM routing.

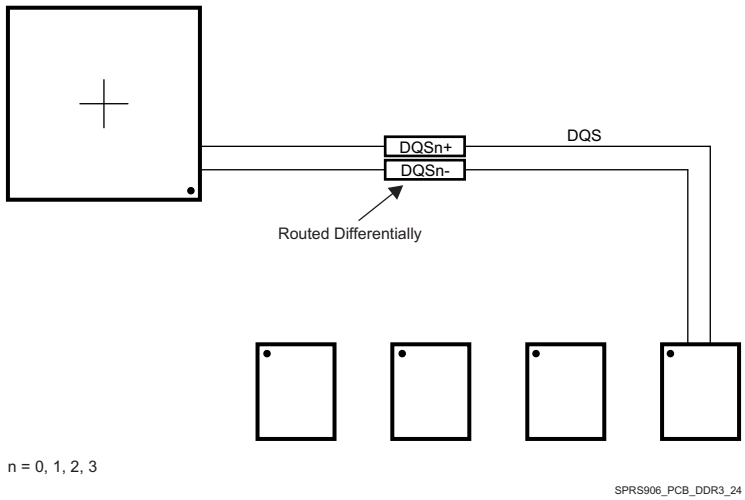


Figure 7-60. DQS Routing With Any Number of Allowed DDR3 Devices

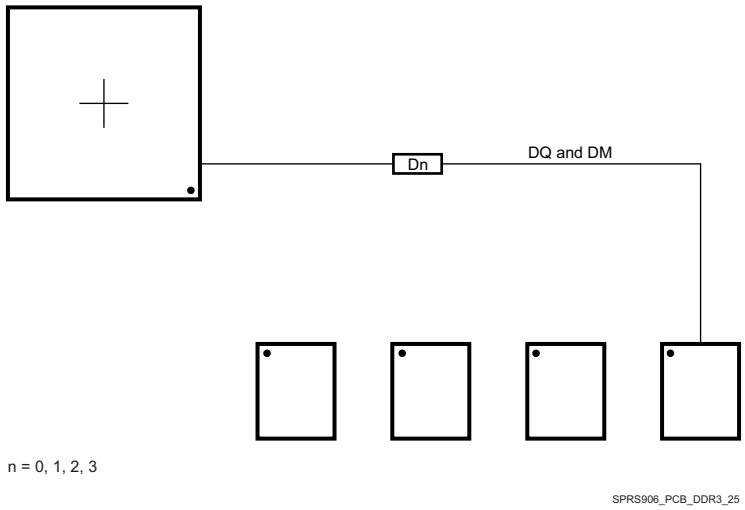


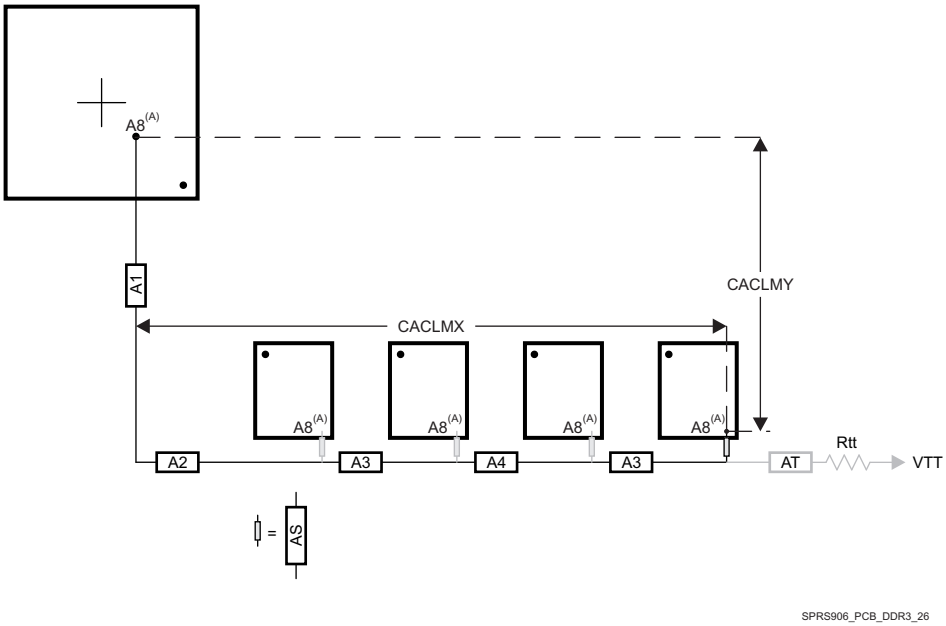
Figure 7-61. DQ/DM Routing With Any Number of Allowed DDR3 Devices

7.7.2.17 Routing Specification

7.7.2.17.1 CK and ADDR_CTRL Routing Specification

Skew within the CK and ADDR_CTRL net classes directly reduces setup and hold margin and, thus, this skew must be controlled. The only way to practically match lengths on a PCB is to lengthen the shorter traces up to the length of the longest net in the net class and its associated clock. A metric to establish this maximum length is Manhattan distance. The Manhattan distance between two points on a PCB is the length between the points when connecting them only with horizontal or vertical segments. A reasonable trace route length is to within a percentage of its Manhattan distance. CACLM is defined as Clock Address Control Longest Manhattan distance.

Given the clock and address pin locations on the processor and the DDR3 memories, the maximum possible Manhattan distance can be determined given the placement. Figure 7-62 and Figure 7-63 show this distance for four loads and two loads, respectively. It is from this distance that the specifications on the lengths of the transmission lines for the address bus are determined. CACLM is determined similarly for other address bus configurations; that is, it is based on the longest net of the CK/ADDR_CTRL net class. For CK and ADDR_CTRL routing, these specifications are contained in Table 7-34.

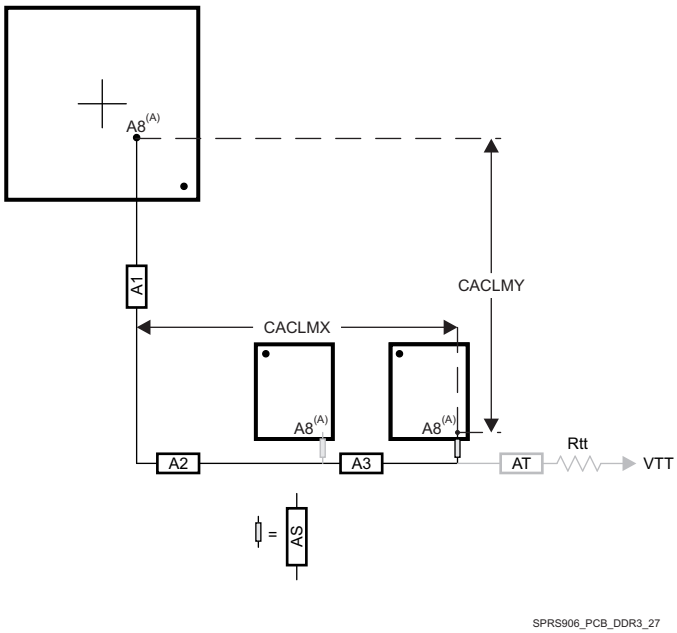


A. It is very likely that the longest CK/ADDR_CTRL Manhattan distance will be for Address Input 8 (A8) on the DDR3 memories. CACLM is based on the longest Manhattan distance due to the device placement. Verify the net class that satisfies this criteria and use as the baseline for CK/ADDR_CTRL skew matching and length control.

The length of shorter CK/ADDR_CTRL stubs as well as the length of the terminator stub are not included in this length calculation. Non-included lengths are grayed out in the figure.

Assuming A8 is the longest, CALM = CACLMY + CACLMX + 300 mils.
The extra 300 mils allows for routing down lower than the DDR3 memories and returning up to reach A8.

Figure 7-62. CACLM for Four Address Loads on One Side of PCB



A. It is very likely that the longest CK/ADDR_CTRL Manhattan distance will be for Address Input 8 (A8) on the DDR3 memories. CACLM is based on the longest Manhattan distance due to the device placement. Verify the net class that satisfies this criteria and use as the baseline for CK/ADDR_CTRL skew matching and length control.

The length of shorter CK/ADDR_CTRL stubs as well as the length of the terminator stub are not included in this length calculation. Non-included lengths are grayed out in the figure.

Assuming A8 is the longest, CALM = CACLMY + CACLMX + 300 mils.
The extra 300 mils allows for routing down lower than the DDR3 memories and returning up to reach A8.

Figure 7-63. CACLM for Two Address Loads on One Side of PCB

Table 7-34. CK and ADDR_CTRL Routing Specification(2)(3)

NO.	PARAMETER	MIN	TYP	MAX	UNIT
CARS31	A1+A2 length			500(1)	ps
CARS32	A1+A2 skew			29	ps
CARS33	A3 length			125	ps
CARS34	A3 skew(4)			6	ps
CARS35	A3 skew(5)			6	ps
CARS36	A4 length			125	ps
CARS37	A4 skew			6	ps
CARS38	AS length		5	17(1)	ps
CARS39	AS skew		1.3	14(1)	ps
CARS310	AS+/AS- length		5	12	ps
CARS311	AS+/AS- skew			1	ps
CARS312	AT length(6)		75		ps
CARS313	AT skew(7)		14		ps
CARS314	AT skew(8)			1	ps
CARS315	CK/ADDR_CTRL trace length			1020	ps
CARS316	Vias per trace			3(1)	vias
CARS317	Via count difference			1(15)	vias
CARS318	Center-to-center CK to other DDR3 trace spacing(9)	4w			
CARS319	Center-to-center ADDR_CTRL to other DDR3 trace spacing(9)(10)	4w			
CARS320	Center-to-center ADDR_CTRL to other ADDR_CTRL trace spacing(9)	3w			

Table 7-34. CK and ADDR_CTRL Routing Specification(2)(3) (continued)

NO.	PARAMETER	MIN	TYP	MAX	UNIT
CARS321	CK center-to-center spacing(11)(12)				
CARS322	CK spacing to other net(9)	4w			
CARS323	Rcp(13)	Zo-1	Zo	Zo+1	Ω
CARS324	Rtt(13)(14)	Zo-5	Zo	Zo+5	Ω

(1) Max value is based upon conservative signal integrity approach. This value could be extended only if detailed signal integrity analysis of rise time and fall time confirms desired operation.

(2) The use of vias should be minimized.

(3) Additional bypass capacitors are required when using the DDR_1V5 plane as the reference plane to allow the return current to jump between the DDR_1V5 plane and the ground plane when the net class switches layers at a via.

(4) Non-mirrored configuration (all DDR3 memories on same side of PCB).

(5) Mirrored configuration (one DDR3 device on top of the board and one DDR3 device on the bottom).

(6) While this length can be increased for convenience, its length should be minimized.

(7) ADDR_CTRL net class only (not CK net class). Minimizing this skew is recommended, but not required.

(8) CK net class only.

(9) Center-to-center spacing is allowed to fall to minimum 2w for up to 1250 mils of routed length.

(10) The ADDR_CTRL net class of the other DDR EMIF is considered *other DDR3 trace spacing*.

(11) CK spacing set to ensure proper differential impedance.

(12) The most important thing to do is control the impedance so inadvertent impedance mismatches are not created. Generally speaking, center-to-center spacing should be either 2w or slightly larger than 2w to achieve a differential impedance equal to twice the singleended impedance, Zo.

(13) Source termination (series resistor at driver) is specifically not allowed.

(14) Termination values should be uniform across the net class.

(15) Via count difference may increase by 1 only if accurate 3-D modeling of the signal flight times – including accurately modeled signal propagation through vias – has been applied to ensure all segment skew maximums are not exceeded.

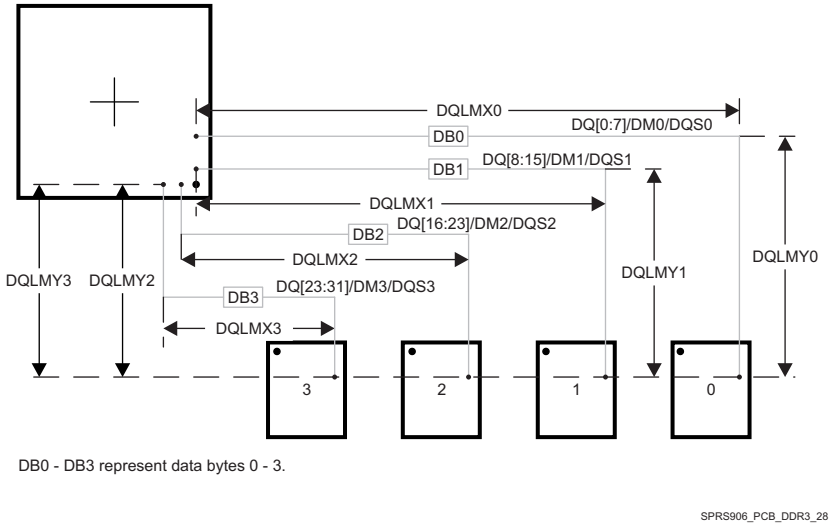
7.7.2.17.2 DQS and DQ Routing Specification

Skew within the DQS and DQ/DM net classes directly reduces setup and hold margin and thus this skew must be controlled. The only way to practically match lengths on a PCB is to lengthen the shorter traces up to the length of the longest net in the net class and its associated clock. As with CK and ADDR_CTRL, a reasonable trace route length is to within a percentage of its Manhattan distance. DQLMn is defined as DQ Longest Manhattan distance n, where n is the byte number. For a 32-bit interface, there are four DQLMs, DQLM0-DQLM3. Likewise, for a 16-bit interface, there are two DQLMs, DQLM0-DQLM1.

NOTE

It is not required, nor is it recommended, to match the lengths across all bytes. Length matching is only required within each byte.

Given the DQS and DQ/DM pin locations on the processor and the DDR3 memories, the maximum possible Manhattan distance can be determined given the placement. Figure 7-64 shows this distance for four loads. It is from this distance that the specifications on the lengths of the transmission lines for the data bus are determined. For DQS and DQ/DM routing, these specifications are contained in Table 7-35.



There are four DQLMs, one for each byte (32-bit interface). Each DQLM is the longest Manhattan distance of the byte; therefore:
DQLM0 = DQLMX0 + DQLMY0
DQLM1 = DQLMX1 + DQLMY1
DQLM2 = DQLMX2 + DQLMY2
DQLM3 = DQLMX3 + DQLMY3

Figure 7-64. DQLM for Any Number of Allowed DDR3 Devices

Table 7-35. Data Routing Specification(2)

NO.	PARAMETER	MIN	TYP	MAX	UNIT
DRS31	DB0 length			340	ps
DRS32	DB1 length			340	ps
DRS33	DB2 length			340	ps
DRS34	DB3 length			340	ps
DRS35	DBn skew(3)			5	ps
DRS36	DQSn+ to DQSn- skew			1	ps
DRS37	DQSn to DBn skew(3)(4)			5(10)	ps
DRS38	Vias per trace			2(1)	vias
DRS39	Via count difference			0(10)	vias
DRS310	Center-to-center DBn to other DDR3 trace spacing(6)	4			w(5)
DRS311	Center-to-center DBn to other DBn trace spacing(7)	3			w(5)
DRS312	DQSn center-to-center spacing(8)(9)				
DRS313	DQSn center-to-center spacing to other net	4			w(5)

(1) Max value is based upon conservative signal integrity approach. This value could be extended only if detailed signal integrity analysis of rise time and fall time confirms desired operation.

(2) External termination disallowed. Data termination should use built-in ODT functionality.

(3) Length matching is only done within a byte. Length matching across bytes is neither required nor recommended.

(4) Each DQS pair is length matched to its associated byte.

(5) Center-to-center spacing is allowed to fall to minimum 2w for up to 1250 mils of routed length.

(6) Other DDR3 trace spacing means other DDR3 net classes not within the byte.

(7) This applies to spacing within the net classes of a byte.

(8) DQS pair spacing is set to ensure proper differential impedance.

(9) The most important thing to do is control the impedance so inadvertent impedance mismatches are not created. Generally speaking, center-to-center spacing should be either 2w or slightly larger than 2w to achieve a differential impedance equal to twice the singleended impedance, Zo.

(10) Via count difference may increase by 1 only if accurate 3-D modeling of the signal flight times – including accurately modeled signal propagation through vias – has been applied to ensure DBn skew and DQSn to DBn skew maximums are not exceeded.

8 Device and Documentation Support

TI offers an extensive line of development tools, including methods to evaluate the performance of the processors, generate code, develop algorithm implementations, and fully integrate and debug software and hardware modules as listed below.

8.1 Device Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all microprocessors (MPUs) and support tools. Each device has one of three prefixes: X, P, or null (no prefix) (for example, DRA79x). Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMDX) through fully qualified production devices and tools (TMDS).

Device development evolutionary flow:

X Experimental device that is not necessarily representative of the final device's electrical specifications and may not use production assembly flow.

P Prototype device that is not necessarily the final silicon die and may not necessarily meet final electrical specifications.

null Production version of the silicon die that is fully qualified.

Support tool development evolutionary flow:

TMDX Development-support product that has not yet completed Texas Instruments internal qualification testing.

TMDS Fully-qualified development-support product.

X and P devices and TMDX development-support tools are shipped against the following disclaimer:

"Developmental product is intended for internal evaluation purposes."

Production devices and TMDS development-support tools have been characterized fully, and the quality and reliability of the device have been demonstrated fully. TI's standard warranty applies.

Predictions show that prototype devices (X or P) have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices are to be used.

For orderable part numbers of DRA79x devices in the CBD package type, see the Package Option Addendum of this document, the TI website (ti.com), or contact your TI sales representative.

For additional description of the device nomenclature markings on the die, see the DRA79x SoC for Automotive Infotainment Silicon Revision 2.1, 2.0.

8.1.1 Standard Package Symbolization

NOTE

Some devices may have a cosmetic circular marking visible on the top of the device package which results from the production test process. In addition, some devices may also show a color variation in the package substrate which results from the substrate manufacturer. These differences are cosmetic only with no reliability impact.

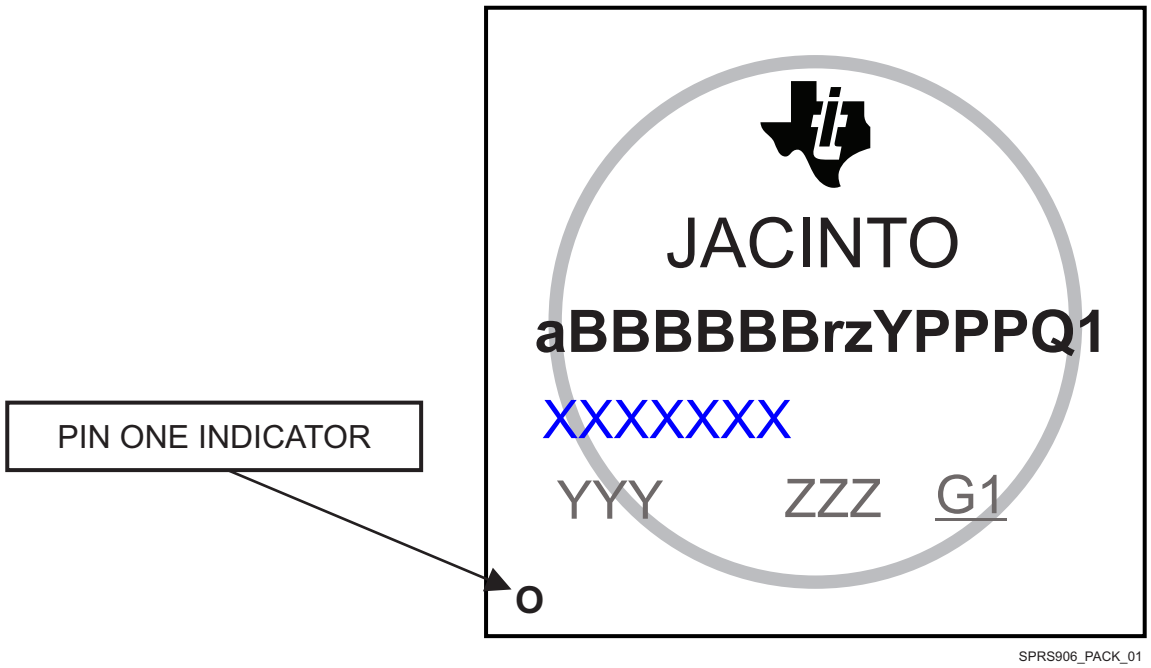


Figure 8-1. Printed Device Reference

8.1.2 Device Naming Convention

Table 8-1. Nomenclature Description

FIELD PARAMETER	FIELD DESCRIPTION	VALUE	DESCRIPTION
a	Device evolution stage(1)	X	Prototype
		P	Preproduction (production test flow, no reliability data)
		BLANK	Production
BBBBBB(3)	Base production part number	DRA790	J6RSP Low Tier
		DRA791	J6RSP Mid Tier
		DRA793	J6RSP High Tier
		DRA797	J6RSP Super High Tier
r	Device revision	BLANK	SR 1.0
		A	SR 2.0
		B	SR 2.1
z	Device Speed	A	Indicates the speed grade for each of the cores in the device. For more information see Table 3-1, *Device Comparison Table*.
		B	
		D	
		H	
		OTHER	
Y	Device type	G	General purpose (Prototype and Production)
		E	Emulation (E) devices
		S	High-Security device, Secure Boot Supported
		D	High security prototype devices with TI Development keys (D)
		Yn	Letter followed by number indicates HS device with customer key
PPP	Package designator	CBD	CBD S-PBGA-N538 (17mm × 17mm) Package
Q1	Automotive Designator	BLANK	Not meeting automotive qualification
		Q1	Meeting Q100 equal requirements, with exceptions as specified in DM.
XXXXXXX	Lot Trace Code		

Table 8-1. Nomenclature Description (continued)

FIELD PARAMETER	FIELD DESCRIPTION	VALUE	DESCRIPTION
YYY	Production Code, For TI use only		
ZZZ	Production Code, For TI use only		
O	Pin one designator		
G1	ECAT—Green package designator		

(1) To designate the stages in the product development cycle, TI assigns prefixes to the part numbers. These prefixes represent evolutionary stages of product development from engineering prototypes through fully qualified production devices.
Prototype devices are shipped against the following disclaimer:
"This product is still under development and is intended for internal evaluation purposes."
Notwithstanding any provision to the contrary, TI makes no warranty expressed, implied, or statutory, including any implied warranty of merchantability of fitness for a specific purpose, of this device.

(2) Applies to device max junction temperature.

(3) X777 base part number with X speed grade indicator is the part number for the superset device. Software should constrain the features and speed used to match the intended production device.

NOTE

BLANK in the symbol or part number is collapsed so there are no gaps between characters.

8.2 Tools and Software

The following products support development for DRA79x platforms:

Design Kits and Evaluation Modules

DRA79x Evaluation Module

The Jacinto™ DRA79x evaluation module designed to speed up development efforts and reduce time-to-market for applications such as infotainment, reconfigurable digital cluster, or integrated digital cockpit. To allow scalability and reuse across Jacinto DRA79x Infotainment SoCs, the EVM is based on the Jacinto DRA797 SoC, which incorporates a heterogeneous, scalable architecture that includes a mixture of the following:

- Arm® Cortex®-A15 core
- Two Arm Cortex-M4 processing subsystems
- One C66x Digital Signal Processors (DSPs)
- 2D- and 3D-graphics processing units including the POWERVR™ SGX544 from Imagination Technologies
- High-definition image and video accelerator

The EVM also integrates a host of peripherals including multicamera interfaces (both parallel and serial) for LVDS-based surround view systems, displays, CAN, and Gigabit Ethernet AVB. The main CPU board integrates these key peripherals such as Ethernet or HDMI, while the infotainment application daughter board (JAMR3) and LCD/TS daughter board will complement the CPU board to deliver complete system to jump start your evaluation and application development.

Development tools

Clock Tree Tool for Sitara, Automotive, Vision Analytics, & Digital Signal Processors

The Clock Tree Tool (CTT) for Sitara™ Arm®, Automotive, and Digital Signal Processors is an interactive clock tree configuration software that provides information about the clocks and modules in these TI devices. It allows the user to: Visualize the device clock tree. Interact with clock tree elements and view the effect on PRCM registers. Interact with the PRCM registers and view the effect on the device clock tree. View a trace of all the device registers affected by the user interaction with clock tree.

XDS110 JTAG Debug Probe

The Texas Instruments XDS110 is a new class of debug probe (emulator) for TI embedded processors. The XDS110 replaces the XDS100 family while supporting a wider variety of standards (IEEE1149.1, IEEE1149.7, SWD) in a single pod. Also, all XDS debug probes support Core and System Trace in all Arm and DSP processors that feature an Embedded Trace Buffer (ETB).

The Texas Instruments XDS110 connects to the target board via a TI 20-pin connector (with multiple adapters for TI 14-pin and, Arm 10-pin and Arm 20-pin) and to the host PC via USB2.0 High Speed (480Mbps). It also features two additional connections: the Auxiliary 14-pin port connector that enables EnergyTrace™, a full duplex UART port and four General-Purpose I/Os, and the Expansion 30-pin connector to connect the XDS110 EnergyTrace HDR add-on.

Models

DRA71x and DRA79x BSDL Model BSDL Model

DRA71x and DRA79x IBIS Model IBIS Model

DRA71x and DRA79x Thermal Model Thermal Model

For a complete listing of development-support tools for the processor platform, visit the Texas Instruments website at ti.com. For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

8.3 Documentation Support

To receive notification of documentation updates, navigate to the device product folder on ti.com. In the upper right corner, click on *Alert me* to register and receive a weekly digest of any product information that has changed. For change details, review the revision history included in any revised document.

The following documents describe the DRA79x devices.

Technical Reference Manual

DRA79x SoC for Automotive Infotainment Silicon Revision 2.1, 2.0

Details the integration, the environment, the functional description, and the programming models for each peripheral and subsystem in the TDA3 family of devices.

Errata

DRA79x SoC for Automotive Infotainment Silicon Revision 2.1, 2.0

Describes the known exceptions to the functional specifications for the device.

8.4 Related Links

The table below lists quick access links. Categories include technical documents, support and community resources, tools and software, and quick access to order now.

Table 8-2. Related Links

PARTS	PRODUCT FOLDER	ORDER NOW	TECHNICAL DOCUMENTS	TOOLS & SOFTWARE	SUPPORT & COMMUNITY
DRA790	Click here	Click here	Click here	Click here	Click here
DRA791	Click here	Click here	Click here	Click here	Click here
DRA793	Click here	Click here	Click here	Click here	Click here
DRA797	Click here	Click here	Click here	Click here	Click here

8.5 Support Resources

TI E2E™ support forums are an engineer's go-to source for fast, verified answers and design help — straight from the experts. Search existing answers or ask your own question to get the quick design help you need.

Linked content is provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's Terms of Use.

8.6 Trademarks

E2E is a trademark of Texas Instruments.
CoreSight is a trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

8.7 Electrostatic Discharge Caution

This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

8.8 Glossary

TI Glossary This glossary lists and explains terms, acronyms, and definitions.

9 Mechanical, Packaging, and Orderable Information

9.1 Packaging Information

The following pages include mechanical, packaging, and orderable information. This information is the most current data available for the designated devices. This data is subject to change without notice and revision of this document. For browser-based versions of this data sheet, refer to the left-hand navigation.

The device package has been specially engineered with a technology called Via Channel. The Via Channel Array technology allows larger than normal PCB via sizes, reduces the number of PCB signal layers required in a PCB design with this package, and will substantially reduce PCB costs compared to a full array 0.65mm pitch package.

PACKAGING INFORMATION

Orderable Device	Status (1)	Package Type	Package Drawing	Pins	Package Qty	Eco Plan (2)	Lead finish/ Ball material (6)	MSL Peak Temp (3)	Op Temp (°C)	Device Marking (4/5)	Samples
DRA790BAGCBDQ1	ACTIVE	FCCSP	CBD	538	84	RoHS & Green	Call TI	Level-3-260C-168 HR	-40 to 125	DRA790BAGCBDQ1 JACINTO 784 784 CBD G1	Samples
DRA790BAGCBDRQ1	ACTIVE	FCCSP	CBD	538	750	RoHS & Green	Call TI	Level-3-260C-168 HR	-40 to 125	DRA790BAGCBDQ1 JACINTO 784 784 CBD G1	Samples
DRA791BBGCBDQ1	ACTIVE	FCCSP	CBD	538	84	RoHS & Green	Call TI	Level-3-260C-168 HR	-40 to 125	DRA791BBGCBDQ1 JACINTO 784 784 CBD G1	Samples
DRA791BBGCBDRQ1	ACTIVE	FCCSP	CBD	538	750	RoHS & Green	Call TI	Level-3-260C-168 HR	-40 to 125	DRA791BBGCBDQ1 JACINTO 784 784 CBD G1	Samples
DRA793BDGCBDQ1	ACTIVE	FCCSP	CBD	538	84	RoHS & Green	Call TI	Level-3-260C-168 HR	-40 to 125	DRA793BDGCBDQ1 JACINTO 784 784 CBD G1	Samples
DRA793BDGCBDRQ1	ACTIVE	FCCSP	CBD	538	750	RoHS & Green	Call TI	Level-3-260C-168 HR	-40 to 125	DRA793BDGCBDQ1 JACINTO 784 784 CBD G1	Samples
DRA797BHGCBDQ1	ACTIVE	FCCSP	CBD	538	84	RoHS & Green	Call TI	Level-3-260C-168 HR	-40 to 125	DRA797BHGCBDQ1 JACINTO 784 784 CBD G1	Samples
DRA797BHGCBDRQ1	ACTIVE	FCCSP	CBD	538	750	RoHS & Green	Call TI	Level-3-260C-168 HR	-40 to 125	DRA797BHGCBDQ1 JACINTO 784 784 CBD G1	Samples

(1) The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.

PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

(2) **RoHS:** TI defines "RoHS" to mean semiconductor products that are compliant with the current EU RoHS requirements for all 10 RoHS substances, including the requirement that RoHS substance do not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, "RoHS" products are suitable for use in specified lead-free processes. TI may reference these types of products as "Pb-Free".
RoHS Exempt: TI defines "RoHS Exempt" to mean products that contain lead but are compliant with EU RoHS pursuant to a specific EU RoHS exemption.
Green: TI defines "Green" to mean the content of Chlorine (Cl) and Bromine (Br) based flame retardants meet JS709B low halogen requirements of <=1000ppm threshold. Antimony trioxide based flame retardants must also meet the <=1000ppm threshold requirement.

(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

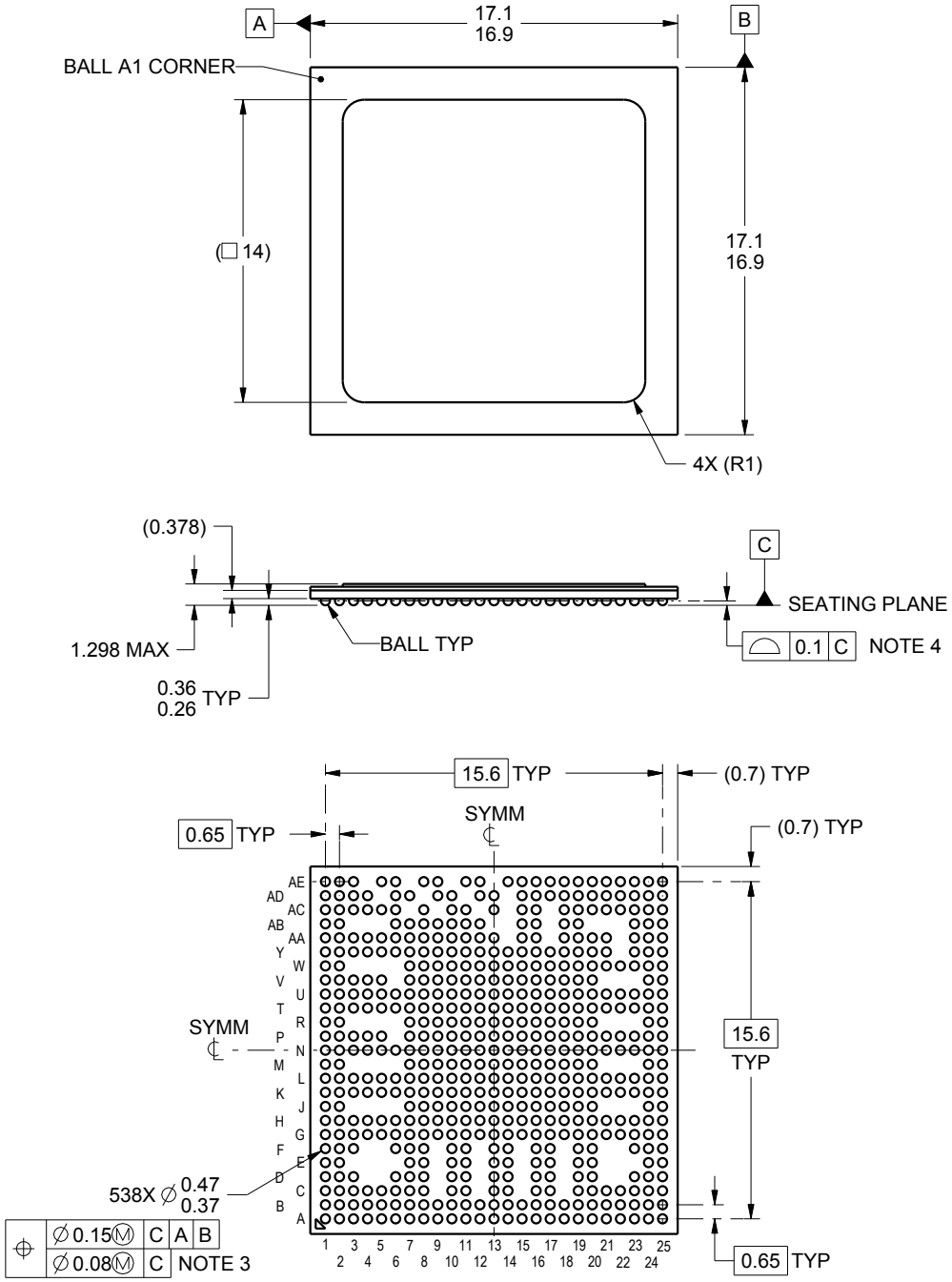
(6) Lead finish/Ball material - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

PACKAGE OUTLINE

CBD0538A

FCBGA - 1.298 mm max height

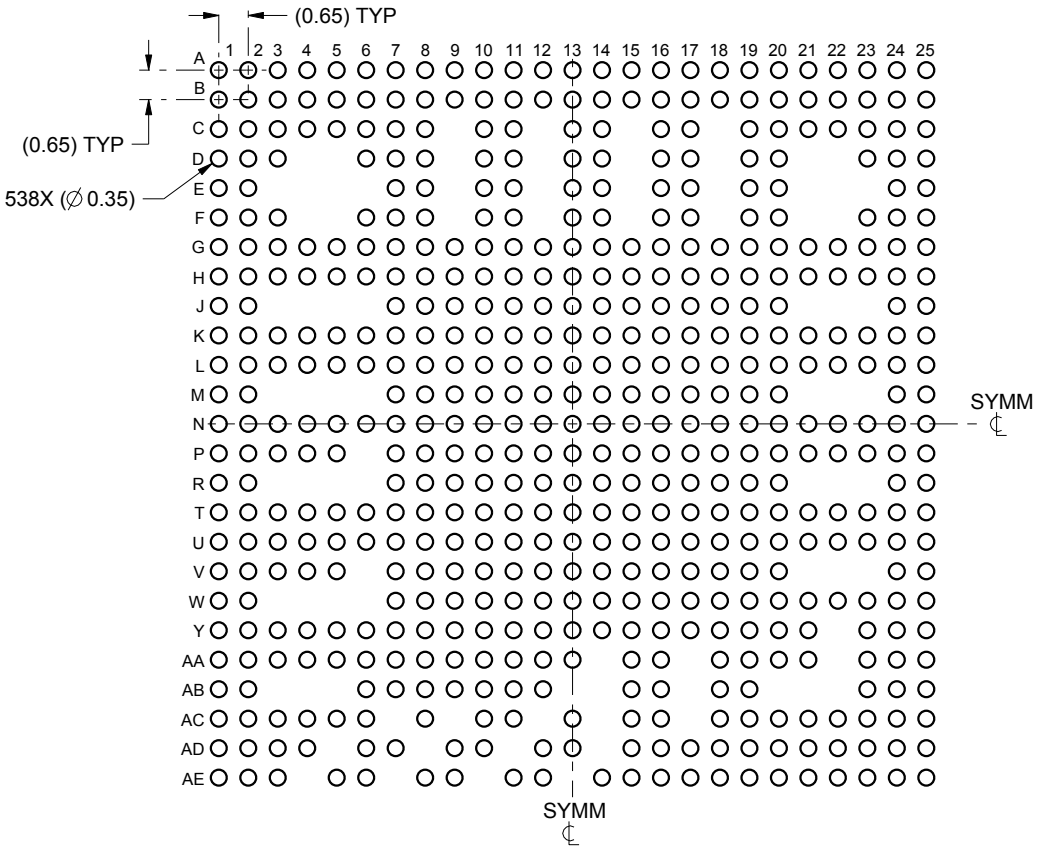
BALL GRID ARRAY



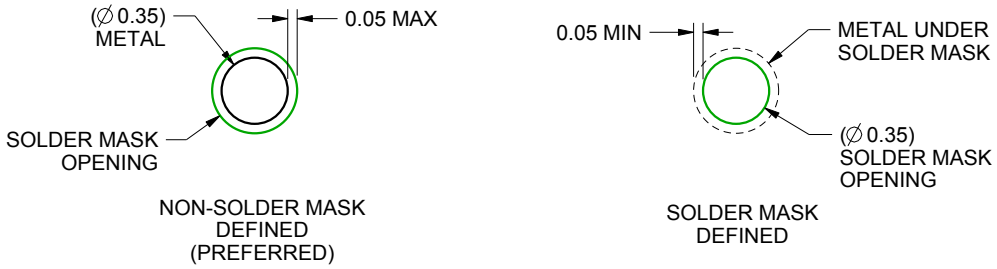
4222967/A 04/2016

NOTES:

1. All linear dimensions are in millimeters. Any dimensions in parenthesis are for reference only. Dimensioning and tolerancing per ASME Y14.5M.
2. This drawing is subject to change without notice.
3. Dimension is measured at the maximum solder ball diameter, parallel to primary datum C.
4. Primary datum C and seating plane are defined by the spherical crowns of the solder balls.



LAND PATTERN EXAMPLE
SCALE:6X

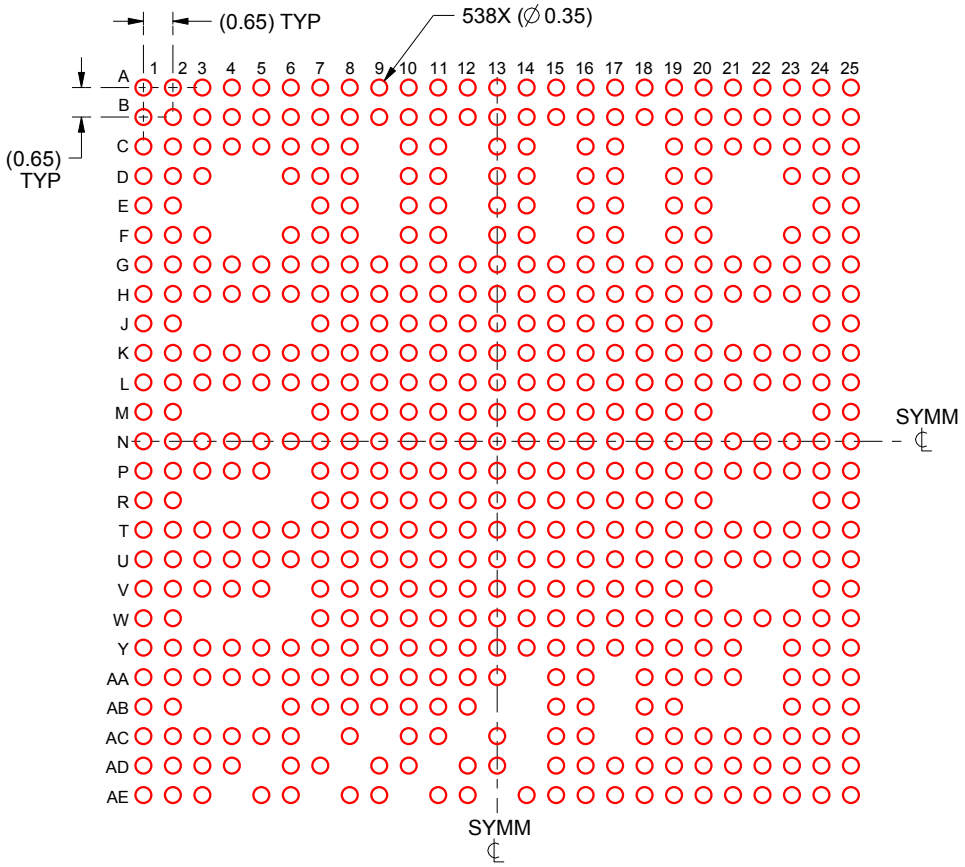


SOLDER MASK DETAILS
NOT TO SCALE

4222967/A 04/2016

NOTES: (continued)

5. Final dimensions may vary due to manufacturing tolerance considerations and also routing constraints.
For more information, see Texas Instruments literature number SPRU811 (www.ti.com/lit/spru811).



SOLDER PASTE EXAMPLE
BASED ON 0.15 mm THICK STENCIL
SCALE:6X

4222967/A 04/2016

NOTES: (continued)

6. Laser cutting apertures with trapezoidal walls and rounded corners may offer better paste release.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265