

Application Note

具有可编程实时单元 (PRU) 的串行和并行端口实施



Jason Ding, Nick Saulnier, and Thomas Yang

摘要

在医疗应用中，模拟或数字子板需要通过串行或并行端口与主板保持高吞吐量点对点通信带宽。根据应用场景的不同，此应用可能还需要自定义协议。

随着系统变得越来越复杂且需要更低的延迟，选择具有合适外设的处理器也变得非常困难。除了串行外设接口 (SPI)、OSPI 或 QSPI 等标准接口之外，许多微处理器或微控制器设计还需要高速串行或并行接口。TI 的许多 Sitara™ 处理器都能够使用 TI 独特的可编程实时单元 (PRU) 内核添加额外接口或自定义协议，从而很好地平衡成本和性能。

TI 的 PRU 内核具有非流水线型 CPU 架构和 1024 位数据总线，因此可实现低延迟且无抖动。TI 提供几种不同类型的 PRU 子系统：PRUSS (在 AM62x 上提供)、PRU-ICSS (在 AM335x、AM437x、AM57x、AM263x 上提供) 和 PRU_ICSSG (在 AM243x、AM64x 上提供)。

本应用手册讨论了如何使用 PRU_ICSSG 来实施串行和并行端口。

内容

1 引言.....	2
2 实施.....	3
2.1 PRU 实施串行端口.....	4
2.2 PRU 实施并行端口.....	9
3 验证.....	9
4 总结.....	11
5 参考文献.....	11

插图清单

图 2-1. PRU GPIO 接口.....	3
图 2-2. PRU 直接输入和输出模式方框图.....	4
图 2-3. 用于数据传输的宏.....	4
图 2-4. 对 BS RAM 进行编程.....	4
图 2-5. PRU 直接模式系统方框图.....	5
图 2-6. GPIO 直接模式工作流程.....	5
图 2-7. PRU R30 (GPO) 移出模式方框图.....	6
图 2-8. PRU R30 (GPO) 移出模式方框图.....	7
图 2-9. PRU 移出和移入编程工作流程.....	8
图 2-10. 并行端口输出的宏.....	9
图 3-1. 64 字节发送时间.....	9
图 3-2. 发送和接收数据比较.....	10
图 3-3. 移位模式.....	10
图 3-4. 发送/接收数据比较.....	10
图 3-5. 4 通道并行数据传输.....	11
图 3-6. 发送/接收数据比较.....	11

表格清单

表 2-1. PRU R30/R31 快速 GPIO 模式.....	3
------------------------------------	---

商标

Sitara™ is a trademark of Texas Instruments.

所有商标均为其各自所有者的财产。

1 引言

许多医疗设备设计需要低延迟通信或自定义通信协议。TI 具有可编程实时单元 (PRU) 内核的 Sitara 处理器旨在满足这些需求，而无需在设计中添加额外的现场可编程门阵列 (FPGA)、应用特定集成电路 (ASIC)、复杂可编程逻辑器件 (CPLD)。

一些潜在用例包括：

有多种类型监控参数的临床监控设备，这些参数包括心电图 (ECG)、肌电图 (EMG)、脉动氧血红蛋白饱和度 (SpO2)、温度和生物电阻抗。但是，这些参数通常需要实时刷新。为满足这些实时需求，可能需要在多参数模块和主板之间使用非标准接口或协议。

在成像系统中，使用连续波 (CW) 多普勒来测量人体内部的血流。解调的多普勒频率生成同相 (I) 和正交 (Q) 数据作为输出。需要使用模数转换器 (ADC) 来支持 3MHz 以上的同步采样率和 14-16 位精度。ADC 输出通常是并行 CMOS 或增强型 SPI。在 X 射线平板探测器 (FPD) 中通过直接成像将电荷转换为数字数据需要模拟前端读出电子器件。数字信号处理器 (DSP)、FPGA、ASIC 或这些器件的组合应用信号调节。这些处理器还管理通过高速接口与外部图像处理单元进行的高速串行或并行通信。

在体外聚合酶链反应 (PCR) 检测系统中，由多波段光源激发样本。需要在单个处理器上捕获来自荧光检测信号链的多个模拟信号，以便进一步处理。当需要将两个以上的 ADC 与处理器连接时，由于共享同一个 SPI DMA 数据路径，所以具有多个 SPI 接口的处理器同时访问两个以上 ADC 的能力受到限制。PRU 允许同时访问多个 ADC。有关更多信息，请参阅 [使用多个 ADC 的同步相干 DAQ 的灵活接口 \(PRU-ICSS\) 参考设计](#) 设计指南。

在这些应用场景中，与 FPGA 或 CPLD 相比，大多数流水线型 CPU 处理器都有一个根本缺点：流水线型处理器具有更高的延迟和更高的抖动。由于 FPGA 或 CPLD 具有灵活性和低延迟，所以非常适合用于这些场景，但添加 FPGA 会增加复杂性和系统成本。TI 具有 PRU 内核的 Sitara 处理器经过精心设计，无需在设计中添加 FPGA、CPLD 即可适合这些场景。本应用手册演示了 PRU 的超快速通用输入/输出 (GPIO)、板侧 RAM 和移出/移入外设如何通过串行或并行接口实现高速定制通信。

2 实施

没有灵活外部逻辑的微控制器可以使用软件，通过其通用输入/输出 (GPIO) 与外部器件进行通信，从而模拟协议或以特定波形驱动某些输出。CPU 内核和 GPIO 引脚之间以及管道之间存在像片上网络 (NOC) 这样的芯片基础设施。该模式从 CPU 的内部寄存器写入，通过 NOC 基础设施传输到达 GPIO。NOC 会对 GPIO 写入命令增加数十纳秒，甚至是数百纳秒的延迟和抖动。系统可能无法满足外设针对输出信号占空比的时序要求以及输出之间的关系。这些因素和其他因素结合起来，导致与 FPGA 相比，标准 MCU 在与 GPIO 交互时将具有更长的延迟和更低的确定性。

TI 的 PRU 内核是出色的定制 CPU，可以在 GPIO 读取和写入时实现极低的延迟和极高的确定性。PRU 内核内部有 32 个通用寄存器，两个寄存器直接连接到通用输出或输入，这意味着 PRU 内核可以在单个周期访问外部引脚。由于寄存器和引脚之间没有 NOC 电路，因此显著降低了从内核到处理器引脚的信号延迟。完全移除了由于信号路径引起的输出信号抖动。PRU 可提供出色的性能，同时不会丧失灵活性。

此应用仅关注 PRU_ICSSG。每个 PRU_ICSSG 实际上有六个内核，每个切片上有三个内核。因此，具有两个 PRU_ICSSG 子系统的 AM243x 或 AM64x 将总共具有 12 个内核。有关更多信息，请参阅 [AM64x/AM243x 技术参考手册](#) 的 PRU_ICSSG 部分。

如图 2-1 所示，通用寄存器 R30 和 R31 可直接布线到 PRU 的专用通用输入 (PRU GPI) 和通用输出 (PRU GPO)。可以通过读取 R31 寄存器来读取 PRU 的 GPI 信号和系统事件或中断，而 PRU 可以通过写入 R31 寄存器来发布系统事件。同样，对 R30 寄存器的写入操作会对专用 PRU GPO 引脚执行写入。

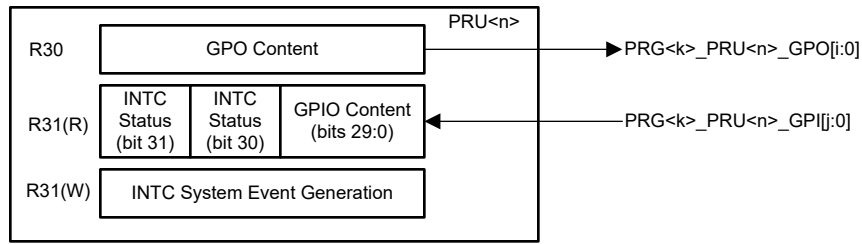


图 2-1. PRU GPIO 接口

PRU 实施了一个具有系统控制单元的增强型通用输入/输出 (GPIO) 模块，支持以下 GPI 和 GPO 模式：直接输入/输出、16 位并行采集、28 位串行移入/移出移位以及 MII_RT。表 2-1 详细说明了输入/输出模式。

表 2-1. PRU R30/R31 快速 GPIO 模式

模式	功能
直接输入/输出	GPI [19:0] 通过 R30 直接馈入 PRU R31 和 GPO [19:0] 输出
16 位并行采集	CLOCKIN 的正边沿或负边沿捕获 DATAIN [0:15]
28 位移入/移出	对 DATAIN 采样并移入 28 位移位寄存器。DATAOUT 通过 28 位移位寄存器移出

内部 PRU 寄存器文件的 PRUx_R31_status [0:19] 位映射到器件级通用输入引脚 (PRUx_GPI [0:19])。在 GPI 直接输入模式下，PRU0_GPI [0:19] 直接馈送到 PRUx_R31_status [0:19]。例如，指令 “ldi R10.b0, R31.b0” 可在单个 PRU 内核周期内将 PRUx_GPI [0:7] 通道上的数据移入 R10 的第一个字节。

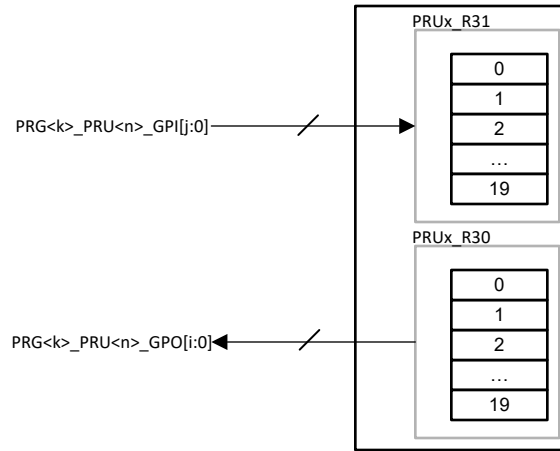


图 2-2. PRU 直接输入和输出模式方框图

以下各节介绍了如何通过 GPIO 直接模式自定义串行和并行端口。对于串行端口，通过 GPO 引脚模拟数据传输，例如，GPO 引脚状态取决于数据寄存器的相应位。

2.1 PRU 实施串行端口

图 2-3 中的宏展示了可以发送串行数据的其中一种方式。通过延迟补偿可调整不同的比特率。无延迟补偿的 single-bit 的发送时间应为 7 个 PRU 时钟周期。如果 PRU 内核在 333MHz 下运行，则周期时间为 21ns 或 47.6MHz。

```

m_send_packet_msb_gpo .macro dataReg, Temp_reg, bitId, TX_PIN, DELAY_COMPEN
    ldi    Temp_reg , bitId ;bits to temporary registers
SEND_BIT_LOOP?:
    qbbc      skip_data_high?, dataReg, Temp_reg ;determine whether bit is 0 or 1
    set      r30,r30,TX_PIN; set PRU<n> pin
    qba      skip_data_low?;jump to delay compensation
skip_data_high?:
    clr      r30,r30,TX_PIN;clr PRU<n> pin
    NOP
skip_data_low?:
    .loop   DELAY_COMPEN
    NOP;delay compensation
    .endloop
    sub     Temp_reg, Temp_reg, 1;move pointer to next bit
    qbne   SEND_BIT_LOOP?, Temp_reg, 0xFF;jump
    .endm
    
```

图 2-3. 用于数据传输的宏

需要考虑的一点是，当帧数据超过 PRU 内核通用寄存器可以保存的最大长度后，从外部存储器或共享存储器内的 PRU 提取数据产生的额外延迟会直接影响通信带宽。如果在整个过程中有较高的带宽，则必须减小带宽。与流水线型 CPU 相比，PRU_ICSSG 集成了宽边 RAM (BS RAM)，可通过指令 Xin 和 XOUT (10.6Gb/s) 来访问。此数据处理加速器让 PRU 可以在单个 PRU 时钟周期内从 BS RAM 读取或写入多达 32 字节的数据。这在大数据量或高通信带宽的情况下非常有用。在将数据移出到外部存储器或 PRU 内部的共享存储器之前，可以将数据暂时存放在 BS RAM 中，等待帧数据传输。PRU 固件执行图 2-4 中的步骤来写入到 BS RAM。

```

ldi r10.w0, 0xf000;BS-RAM write auto increment enable
xout 0x30 ,&r10, 2;Store RAM Address and AutoIndexEn to BS RAM using XOUT instruction
xout 0x30 ,&r2, 32;send r2-r9 data to BSRAM,If AutoIndexEn = 1h, repeat thisto write additional data
    
```

图 2-4. 对 BS RAM 进行编程

应该注意的是，自动索引启用会导致每个 PRU 或 RTU_PRU 内核对 BS RAM 进行写入或读取，进而导致 RAM 地址递增 1 位。RAM 地址的每个增量都相当于 32 个字节。无论读取或写入数据大小如何，下一个读取或写入地

址始终在 32 个字节之后。在上述讨论的基础上，图 2-5 和图 2-6 展示了 PRU GPIO 根据直接模式和软件工作流程实施串行通信的系统方框图。

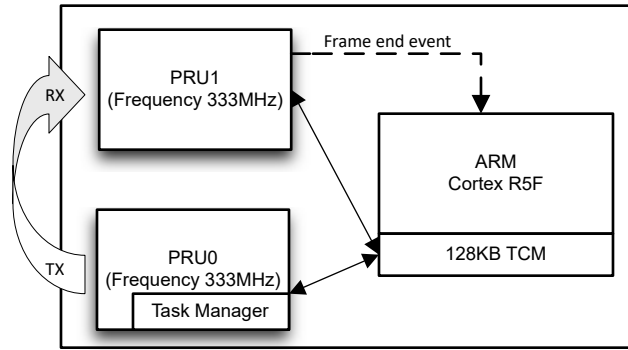


图 2-5. PRU 直接模式系统方框图

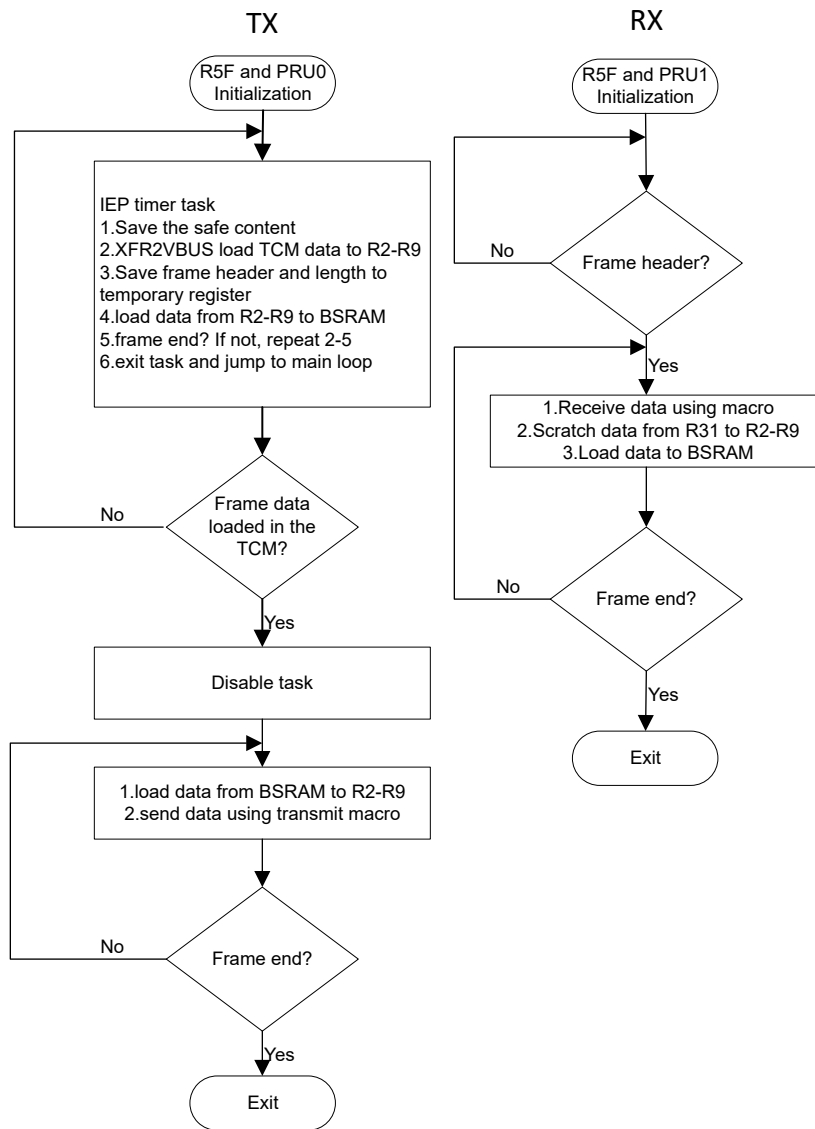
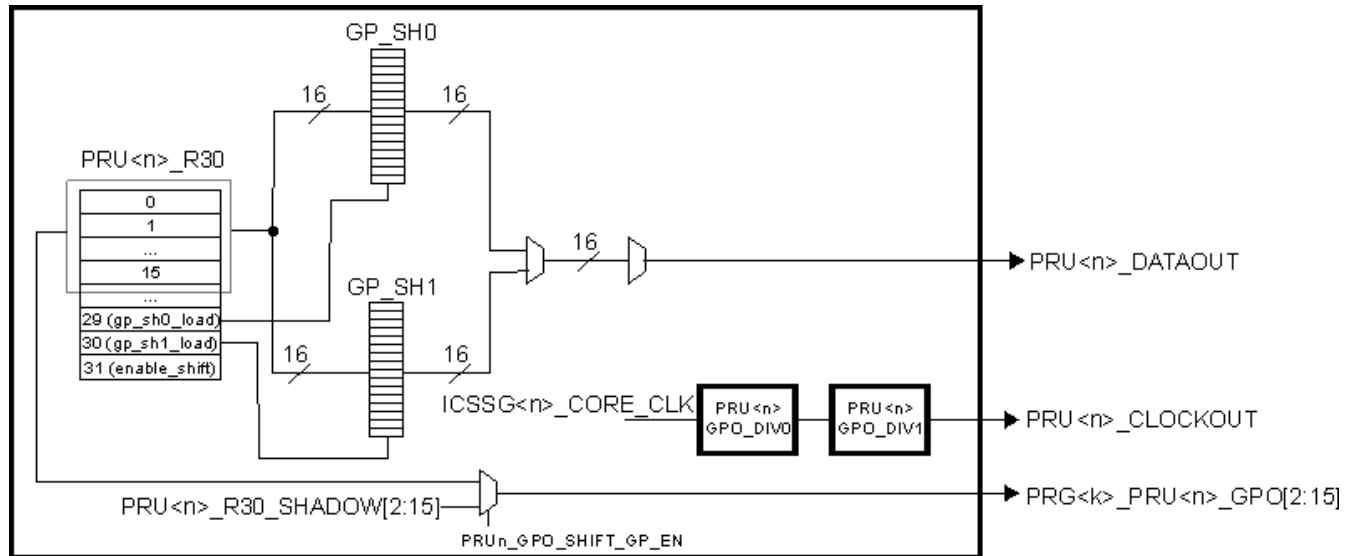


图 2-6. GPIO 直接模式工作流程

应用内核先初始化 PRU 内核，然后 PRU 才能正常启动，例如，PRU 初始化 GPIO 接口和工业以太网外设 (IEP)，这是工业以太网功能所需的硬件工作。IEP 模块采用具有 16 个比较事件的工业以太网计时器、工业以太网同步发生器和锁存捕获、工业以太网看门狗计时器和数字 I/O 端口 (DIGIO)。在这种情况下，可以通过 IEP 触发的事件定期查询紧耦合存储器 (TCM) 中的数据。IEP 计时器触发周期设置为 10kHz (100 μs)，以便在任务中轮询 TCM 中的有效数据。在任务中使用 Xin 和 Xout 指令来保护上下文，以避免与寄存器发生重叠，然后使用 XFR2VBUS DMA 小工具在紧耦合存储器 (TCM) 指定的地址加载数据，这一点很重要。XFR2VBUS 是一款简单的硬件加速器，用于实现多核存储器子系统 (MSMC) 的最低读取往返延迟，以及将 PRU 所感知到的延迟与其他部分隔离开来。当应用内核将帧数据存储到 TCM 时，PRU 会使用头数据来确定是否需要执行数据传输，并在执行数据传输之前将数据预存储到 BSRAM 中，从而消除存储器访问延迟的影响。

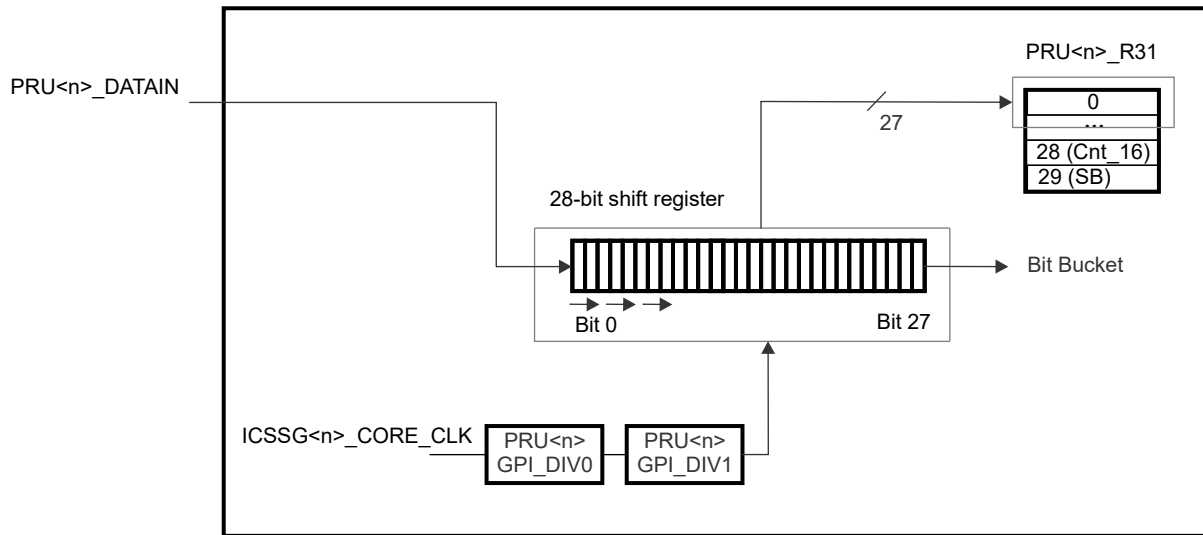
实施串行端口的另一种方法是 GPIO 移出和移入模式。在移出模式下，在 PRU_CLK 的每个上升沿从 PRU_DATAOUT 引脚移出数据。移位速率由应用于 PRU 内核时钟的两个级联分频器的有效除数控制。为了在串行通信中实现 100Mbit 数据速率，必须将 100MHz 时钟与除以三的 250MHz 内核时钟结合使用。为避免数据加载过程对串行通信产生影响，PRU 移位数据模式提供了两个影子寄存器 (GPO_SH0 和 GPO_SH1)，这两个寄存器可用于支持乒乓缓冲器。影子寄存器可通过 PRUx_R30[29:30] 单独编程，当设置 PRUx_R30[29:30] 时，将 PRUxR30[0:15]/[15:0] 中的数据加载到 GPO_SH0 和 GPO_SH1 中。PRU 移出模式可以设置为自由运行模式和固定时钟计数模式。对于自由运行时钟模式，移位操作将继续，直到清除 PRUx_ENABLE_SHIFT(PRUxR30[31])。清除 PRUx_ENABLE_SHIFT 后，移位操作将完成移出当前影子寄存器，停止，然后复位。对于固定时钟计数模式，要移出的数据位数由配置寄存器定义。图 2-7 展示了移出模式的系统架构。



icss-010

图 2-7. PRU R30 (GPO) 移出模式方框图

在 28 位移入模式下，对通用输入引脚 PRU_DATAIN 进行采样，并在内部时钟脉冲作用下移入 28 位移位寄存器。该寄存器填充最低有效位 (LSB) 顺序 (从位 0 到位 27)，然后溢出到一个位桶中。在移入检测到开始位 (1/0) 后，28 位寄存器映射到 PRU_R31[0:27]，每 16 个移位时钟设置一次 Cnt_16，映射到 R31 寄存器的串行数据可以复制到其他系统寄存器。与移出模式类似，选择具有 250MHz 内核时钟的 100MHz 采样时钟。图 2-8 展示了 PRU GPI 移入模式的方框图。在此应用中，传输在自由运行模式下运行。图 2-9 展示了编程工作流程。



icss-008

图 2-8. PRU R30 (GPO) 移出模式方框图

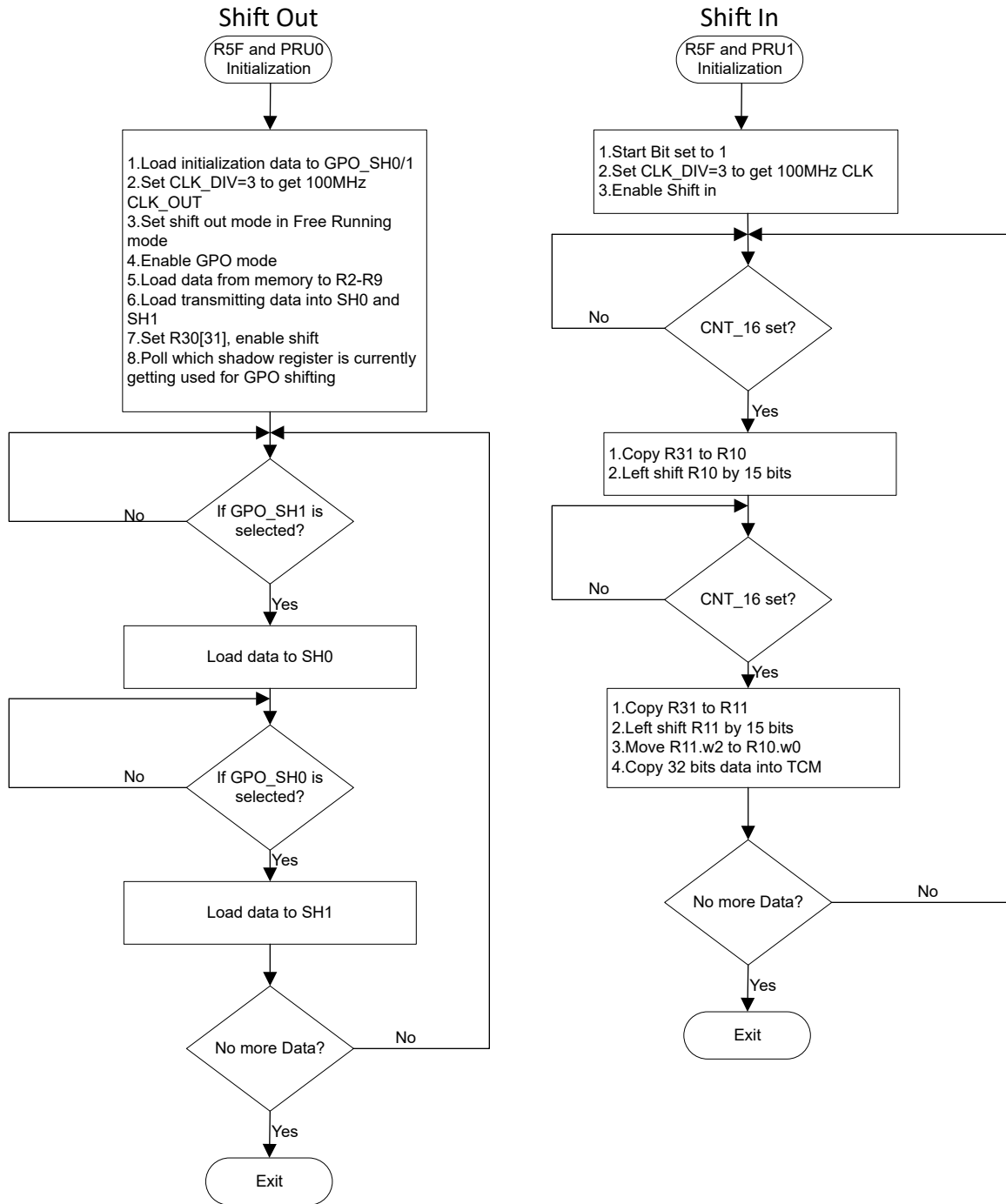


图 2-9. PRU 移出和移入编程工作流程

需要注意的是，SH0 和 SH1 的深度为 16 位，分频因子为 3 时，数据加载和处理时间必须小于 48 个 PRU 周期。在移入模式中，数据输入节点不知道此通信的总字节长度，因此可以在起始帧中包括总数据长度来定义移入模式的周期数。

有关更多用于串行通信的 PRU，请参阅 [使用 8b-10b 线路编码和可编程实时单元的驱动器内通信](#) 和 [适用于多轴伺服控制的 FSI 带宽优化](#)

2.2 PRU 实施并行端口

由于 PRU 具有出色的灵活性，因此 PRU 也适用于并行端口应用场景。本节还介绍了 PRU 的 GPIO 直接输入和输出模式。PRU 可以实施一个具有直接输入/输出模式或并行捕获模式的并行端口。并行捕获模式使用外部时钟在时钟的上升沿或下降沿锁存数据。直接输入/输出模式可用于为并行端口输出提供时钟。Set 和 clr 指令可用于切换时钟输出。例如，set r30, r30, 0 指令会将 PRUx_GPO0 引脚设置为高电压电平，而 clr r30, r30, 0 指令会将引脚设置为低电压电平。由于 PRU 指令具有确定性，因此它还可以补偿数据的稳定时间以及保持时间。

并行端口的一个简单宏如图 2-10 所示，其中时钟周期为 5 个 PRU 周期。

```
m_send_packet_msb_gpo_CLK_shfit .macro dataReg, clkPin

    lsr r30.b0,dataReg.b3,4
    set r30,r30,clkPin
    nop
    clr r30,r30,clkPin
    nop
    lsr r30.b0,dataReg.b3,0
    set r30,r30,clkPin
    nop
    clr r30,r30,clkPin
    nop
```

图 2-10. 并行端口输出的宏

3 验证

为了验证串行端口数据传输的精度和带宽，为 TX/RX 环回测试应用 GPIO 直接模式。R5F 将 64 字节的数据存储到 TCM 的地址 0x78000000。PRU0 在周期性任务中查询 TCM 和传输数据，传输后清除帧头数据，以避免重复。同时，PRU1 将接收到的数据存储到 TCM 的地址 0x78000040，从而通过比较已发送和已接收的数据来验证接收到的数据的准确性。示波器测量发送端的传输带宽。测试结果如图 3-1 和图 3-2 所示。

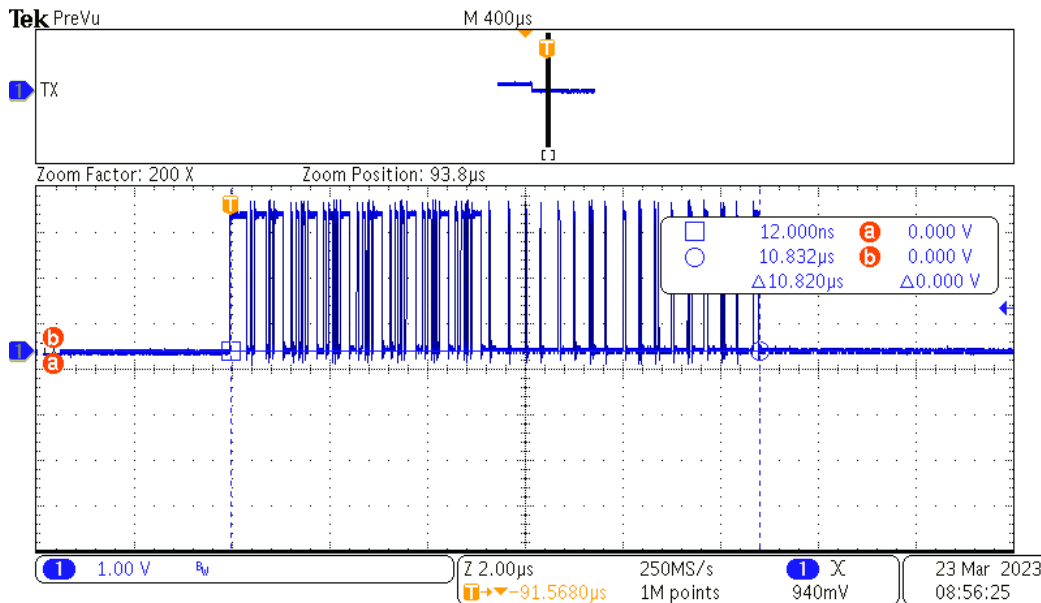


图 3-1. 64 字节发送时间

如图 3-1 所示，带宽=数据长度/时间=512 位/10.82 μ s=47.32Mbps。

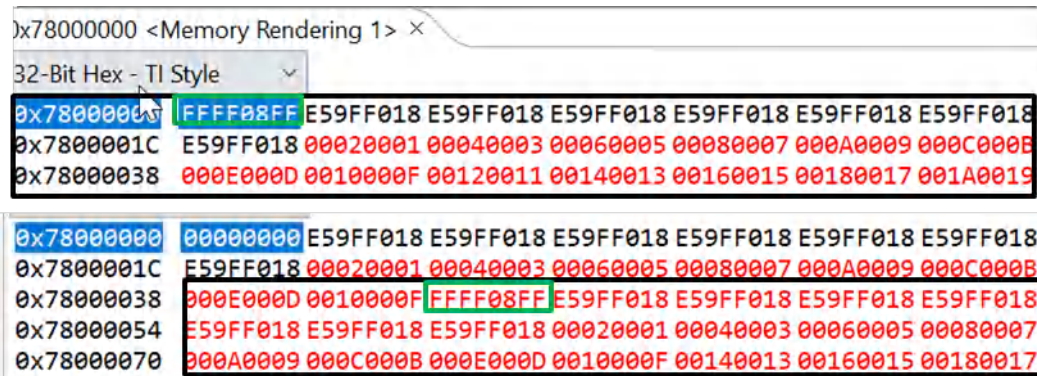


图 3-2. 发送和接收数据比较

PRU_GPIO 移位模式与直接模式类似，执行 64 字节的 TX/RX 环回测试，PRU0 处于移出模式，PRU1 处于移入模式，且发送数据存储于 R5F TCM 的地址 0x78000114。数据接收完成后，存储在 0x78000180。

图 3-4 显示，PRU1 内核收到正确数据，没有错误。

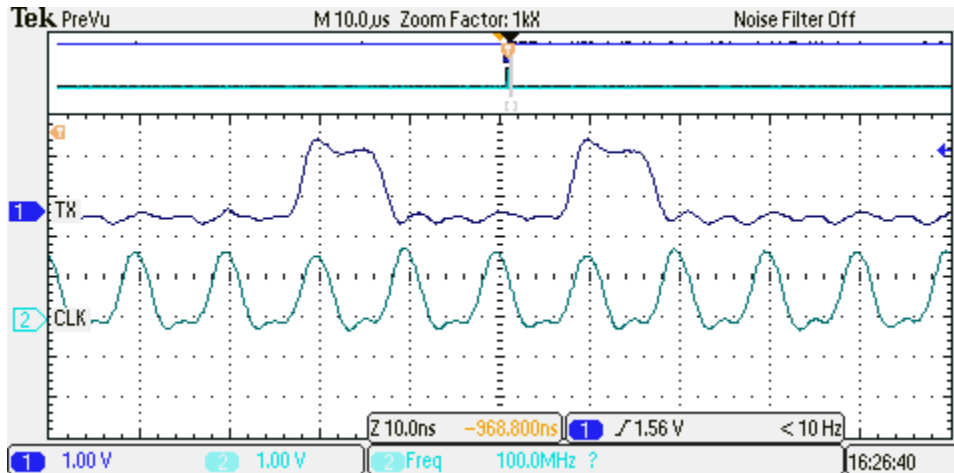


图 3-3. 移位模式

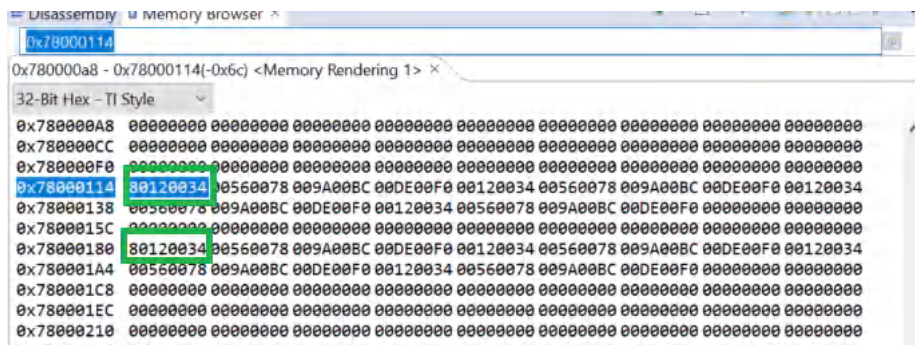


图 3-4. 发送/接收数据比较

目标是使用 GPIO 移出模式在串行通信中实现 100Mbit 数据速率。图 3-3 显示，CLK_OUT 时钟周期为 10MHz，并且发送和接收数据之间没有错误位，这证明了串行通信的有效性。并且为了提高串行通信的可靠性，CRC 硬件加速器集成在 PRU_ICSSG 内，以便在每个通信帧后添加 CRC 校验和。

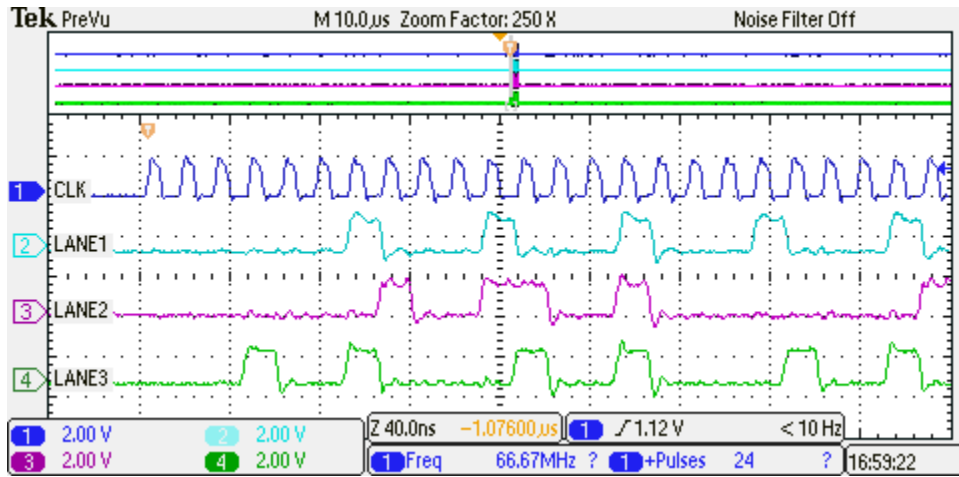


图 3-5. 4 通道并行数据传输

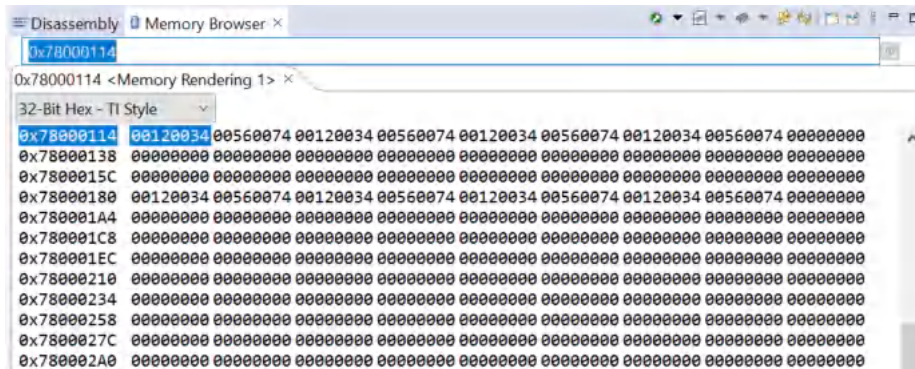


图 3-6. 发送/接收数据比较

为了显示并行端口性能，执行了 TX/RX 环回实验，发送数据存储在 R5F TCM 的 0x78000114。数据接收完成后，存储在 0x78000180。在这种情况下，时钟周期等于 5 个 PRU 周期，带宽 = 数据长度/时间 = 4 位/周期 / (3ns/时钟 x 5 个时钟/周期) = 266.4Mbps。随着使用更多 PRU GPI/GPO 信号，数据吞吐量会增加。例如，使用 1 个时钟信号和 15 条数据线可实现 1Gbps 的带宽：带宽 = 数据长度/时间 = 15 位/周期 / (3ns/时钟 x 5 个时钟/周期) = 1Gbps。

4 总结

此应用手册提供了使用 PRU 子系统实施串行和并行端口的方法和示例，该子系统为系统中的点对点数据交换提供了非常灵活的高带宽通信方式。

5 参考文献

- 德州仪器 (TI) : [AM243x Sitara™ 微控制器数据表](#)
- 德州仪器 (TI) : [AM64x/AM243x 处理器器件技术参考手册](#)
- 德州仪器 (TI) : [PRU 汇编指令用户指南](#)
- 德州仪器 (TI) : [使用 8b-10b 线路编码和可编程实时单元的驱动器内通信](#)
- 德州仪器 (TI) : [适用于多轴伺服控制的 FSI 带宽优化](#)

重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2024，德州仪器 (TI) 公司