参考資料

TI Designs: TIDEP-0089
Sitara™ AM335x用のCC-Link IE Field Basicマスタおよびスレーブのリファレンス・デザイン

TEXAS INSTRUMENTS

概要

このCC-Link IE Field Basicリファレンス・デザインは、Sitara™AM335xプロセッサ上で、Processor SDK RTOSおよびProcessor SDK Linux®とともに動作します。RTOSの場合、このデザインはネットワーク開発キット(NDK)トランスポート・レイヤを使用し、NIMU (EMAC)およびNIMU_ICSS (PRU-ICSS Dual-emacファームウェア)レイヤの両方でRTOSをサポートする例が含まれています。Linuxの場合、このデザインはEMACまたはPRU-ICSSを基礎とするLinuxネットワーク・スタックを使用します。実装では、マスタ・ステーションまたはスレーブ・ステーション構成を使用できます。

リソース

TIDEP-0089	デザイン・フォルダ
AM3359	プロダクト・フォルダ
DP83822I	プロダクト・フォルダ
TMDSICE3359	プロダクト・フォルダ

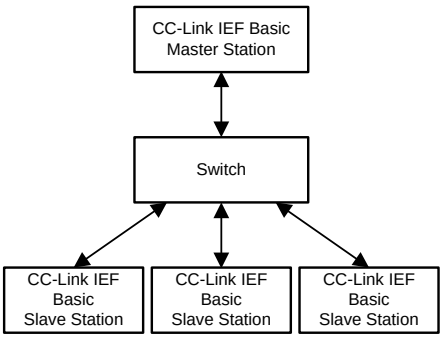
TI E2E™ Community　E2Eエキスパートに質問

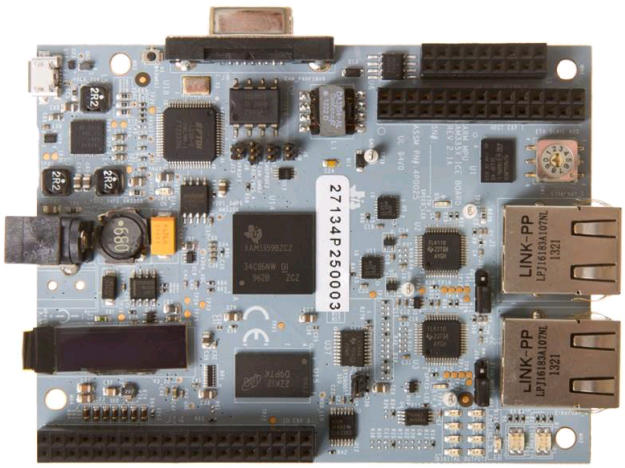
特長

- CC-Link産業用イーサネットField Basicマスタおよびスレーブ実装
- 100Mbpsに対応
- シームレス・メッセージ・プロトコル(SLMP)準拠 - スレーブ・ステーション
- 最大64のスレーブ・ステーションをサポート - マスタ・ステーション
- 占有ステーションの最大数はグループごとに16
- プロセッサSDKに付属のソースコードと完全互換
- プロセッサSDKを使用して、他のEVMもサポート可能

アプリケーション

- 産業用イーサネット
- サーボ・ドライブおよびモーション・コントロール
- プログラマブル・ロジック・コントローラ(PLC)
- 産業用通信モジュール
- 産業用入出力(IO)モジュール
- 産業用センサおよびアクチュエータ





1 System Description

Control and Communication Link (CC-Link) is an open network administered as a fully-open architecture by the CC-Link Partner Association (CLPA). CC-Link guarantees 10-Mbit/s performance across the fieldbus network, regardless of device type, which eliminates hidden bottle necks that are common with other open systems. For the Industrial Ethernet version of CC-Link IE Field, the speed is 1 Gbit. CC-Link offers the freedom to integrate a wide variety of automation components into a single, seamless automation system on the network. CC-Link is available in multiple formats: CC-Link, CC-Link Safety, CC-Link IE (Industrial Ethernet) Control, and CC-Link IE Field.

CC-Link IE Field Basic (or *IEF Basic*) is a new addition to the family of CC-Link IE open network technologies that enable device vendors to easily add CC-Link IE compatibility to any product with a 100-Mbit Ethernet port. IEF Basic is easily implemented on devices or master controllers by software alone, which enables added compatibility to existing products without any hardware modification. IEF Basic's stack is compatible with TCP/IP and UDP/IP; the stack blends seamlessly with other Ethernet technologies (including switches, cables, connectors, and wireless systems). Finally, a master controller for the network is also purely software based, so any industrial PC or other Ethernet equipped controller can be rapidly deployed to run an IEF Basic network without requiring any special interface cards, driver development, or other additional work. The devices all communicate using cyclic (synchronous) exchange of data, which means network updates are performed on a regular, deterministic schedule.

Seamless Message Protocol (SLMP) is a common protocol for achieving seamless communication between applications without awareness of network hierarchy or boundaries between the CC-Link family network and general-purpose Ethernet devices. SLMP is implemented on network hierarchies, such as TCP/IP, CC-Link IE, and CC-Link. SLMP implementation makes client and server-type and push-type communication possible between general-purpose Ethernet devices, CC-Link IE devices, and CC-Link devices.[21]

2 System Overview

This section gives a basic overview of CC-Link IEF Basic protocol. Most of this information here is from the IEF Basic User Guide. For additional details on IEF Basic, refer to the CC-Link IEF Basic User Guide available from CLPA.

2.1 Block Diagram

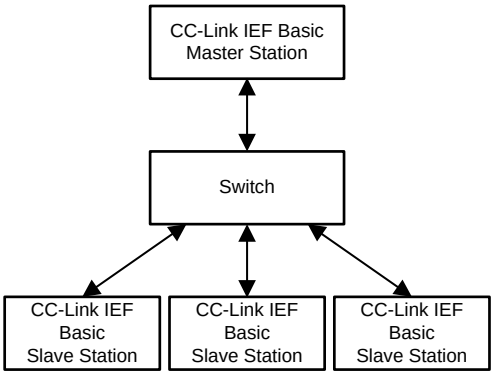


図 1. CC-Link Block Diagram

2.2 Highlighted Products

2.2.1 AM3359

- Up to 1-GHz Sitara ARM® Cortex™-A8 32-bit RISC processor
- NEON™ SIMD coprocessor
- 32KB of L1 Instruction and 32KB of data cache with single-error detection (parity)
- 256KB of L2 cache with error correcting code (ECC)
- 176KB of on-chip boot ROM
- 64KB of dedicated RAM
- Emulation and debug - JTAG
- Interrupt controller (up to 128 interrupt requests)
- PRU-ICSS:
  - Supports protocols such as EtherCAT®, PROFIBUS®, PROFINET®, EtherNet/IP™, and more
  - Two PRUs 32-bit load and store RISC processor capable of running at 200 MHz
  - 8KB of instruction RAM with single-error detection (parity)
  - 8KB of data RAM with single-error detection (parity)
  - Single-cycle, 32-bit multiplier with 64-bit accumulator
  - Enhanced GPIO module provides shift-in or shift-out support and parallel latch on external signal
  - 12KB of shared RAM with single-error detection (parity)
  - Three 120-byte register banks accessible by each PRU INTC for handling system input events
  - Local interconnect bus for connecting internal and external masters to the resources inside the PRU-ICSS
  - Peripherals inside the PRU-ICSS:
    - One universal asynchronous receiver and transmitter (UART) port with flow control pins that supports up to 12 Mbps
    - One enhanced capture (eCAP) module
    - Two MII Ethernet ports that support industrial Ethernet, such as EtherCAT
    - One management data input and output (MDIO) port
  - On-chip memory (shared L3 RAM):
    - 64KB of general-purpose on-chip memory controller (OCMC) RAM
    - Accessible to all masters
- External memory interfaces (EMIF):
  - mDDR(LPDDR), DDR2, DDR3, and DDR3L controller:
  - mDDR: 200-MHz clock (400-MHz data rate)
  - DDR2: 266-MHz clock (532-MHz data rate)
  - DDR3: 400-MHz clock (800-MHz data rate)
  - DDR3L: 400-MHz clock (800-MHz data rate)
  - 16-bit data bus
  - 1GB of total addressable space

  - Supports one x16 or two x8 memory device configurations
- General-purpose memory controller (GPMC)
- Flexible 8-bit and 16-bit asynchronous memory interface with up to seven chip selects (NAND, OR, Muxed-NOR, or SRAM)
- Uses BCH code to support 4-, 8-, or 16-bit ECC
- Uses hamming code to support 1-bit ECC

See the *AM335x Sitara Processors*[1] datasheet for a complete list of features.

2.2.2 DP83822I

- IEEE 802.3u compliant: 100BASE-FX, 100BASETX and 10BASE-Te
- MII, RMII, and RGMII MAC Interfaces
- Low-power single supply options:
  - 1.8-V average (AVD) < 120 mW
  - 3.3-V AVD < 220 mW
- ±16-kV HBM ESD Protection
- ±8-kV IEC 61000-4-2 ESD Protection
- Start of frame detect for IEEE 1588 time stamp
- Fast link-down timing
- Auto-crossover in force modes
- Operating temperature: –40°C to 125°C
- IO voltages: 3.3 V, 2.5 V, and 1.8 V
- Power savings features:
  - Energy efficient Ethernet (EEE) IEEE 802.3az
  - Wake-on-LAN (WoL) support with magic packet detection
  - Programmable energy savings modes
- Cable diagnostics
- BIST
- Management data clock (MDC) and MDIO interface

See the *DP83822 Robust, Low Power 10/100 Mbps Ethernet Physical Layer Transceiver*[2] datasheet for a complete list of features.

2.2.3 TMDSICE3359 ICE EVM

Hardware specifications:

- AM3359 ARM Cortex-A8
- DDR3, NOR flash, and SPI flash
- Organize light-emitting diode (OLED) display
- TPS65910 power management 24-V power supply
- USB cable for JTAG interface and serial console

PRU-ICSS subsystem for industrial communication, capable of supporting:

- CC-Link IEF Basic Master/Slave
- PROFIBUS interface
- CANOpen
- EtherNet/IP
- PROFINET
- Sercos III
- Digital IO
- SPI
- UART
- JTAG

2.3 System Design Theory

2.3.1 CC-Link

The following is an overview of the characteristics of IEF Basic:

1. Realization of cyclic transmission using IP packets
   - Using an Internet Protocol with an EtherType of Ethernet frame, IP packets allow the realization of cyclic transmission for periodically updating linked devices.
   - Protocols using other IP packets (including HTTP, FTP, SLMP, and so on) can transmit on the same IP network.
   - Data periodically communicates between the master station and slave stations using link devices.
2. Defining protocol at the application layer
   - Because the application layer defines the protocol, there is no required special hardware to realize IEF Basic, and implementing the software allows cyclic transmission realization.
3. Simple protocols
   - Protocol is request-response type with a simple status and status transition that is managed at the station. In addition, the small number of frame types allows simple implementation in machines.
4. Inheritance of CC-Link IE Field Network protocols
   - Because the primary components in CC-Link IE Field are inherited as much as necessary, configuration of the IEF Basic network is similar to that of the CC-Link IE Field Network.

2.3.1.1 Types of Communication

CC-Link IEF Basic performs transmission and reception of frames related to cyclic transmission. By storing the information related with cyclic transmission and station information within this frame type, a single frame can be used for cyclic transmission and network management functions.

2.3.1.2 Protocol Hierarchy

図 2 shows the protocol hierarchy of CC-Link IEF Basic.

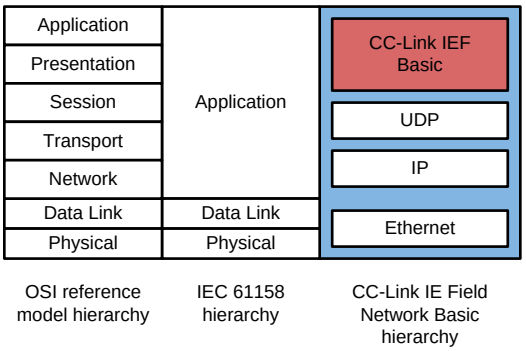


図 2. CC-Link IEF Basic Overview

2.3.2 SLMP

図 3 shows an overview of SLMP.

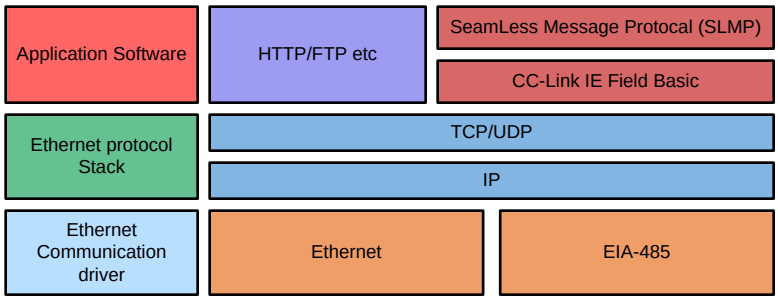


図 3. SLMP Overview

2.3.2.1 Features

SLMP offers the following features:

1. Access to network information
   - SLMP communication makes it possible to access (read and write) information (stored memory) within a server from a client. This stored memory may include internal memory, drive memory, expanded module memory, and so on as well as other information, such as device operation status information, production status information, and sequence program and parameter files.
2. Control from a remote location
   - SLMP-based communication makes it possible to perform server remote control from a client. The control operations include remote control (remote run, stop, pause, clear latch, reset), remote password setup and clearance, and error code initialization.
3. On-demand communication
   - SLMP-based communication makes it possible to transmit urgent data without request from the server to a client, which is called on-demand communication.
4. Efficient data collection
   - Using SLMP, the client can collect data within the server If the data to be collected is registered in the service in advance, the data distributes without a request by the client.
5. Access to device information
   - SLMP provides a meaning of directly accessing device information. For example, the connected device is automatically detected using the SLMP command and parameter setting. Monitoring and diagnosis can be performed for any device using the same procedure.
6. Integration of other open networks
   - For transient transmission in other open networks, access is enabled from CC-Link Family Network to other open networks from the conversion model using SLMP. For example, the connected devices in other open networks are automatically detected using SLMP, and parameter setting and diagnosis can be performed for any device using the same procedure.

2.3.3 Protocol Overview

The following sections show the sequencing of the communication between master and slave station in an IEF Basic network.[20]

2.3.3.1 Overall Processing Sequence

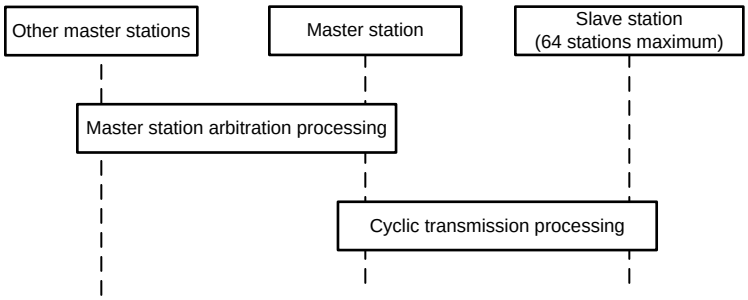


図 4. Basic Sequence of Processing

The processing is performed in the following sequence:

1. The master station performs master station arbitration processing.
2. If master station duplication was not detected in master station arbitration processing, the master station performs cyclic transmission processing.

2.3.3.2 Master Station Arbitration

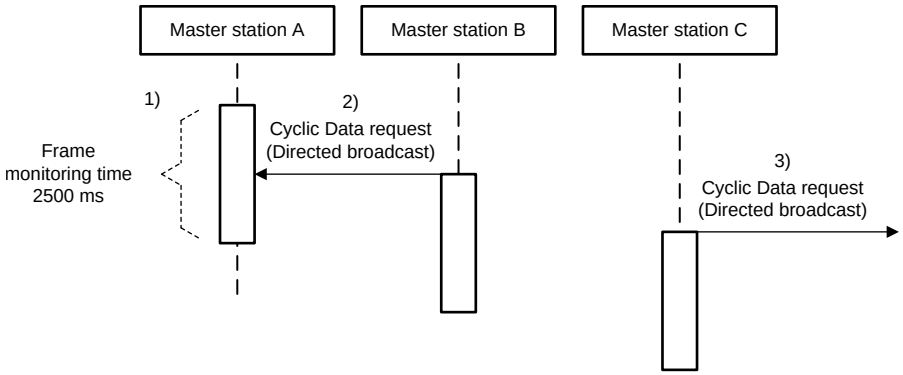


図 5. Master Station Arbitration

The processing is performed in the following sequence:

1. The master station monitors the frame for 2500 ms to check whether the station receives *Cyclic Data* command requests from other master stations, as master station arbitration, prior to performing cyclic transmission processing with the slave station.
2. If the master station receives a Cyclic Data command request, the station judges that there is master station duplication.
3. If the master station does not receive a Cyclic Data command request, the station judges that there is no master station duplication.
4. The master station performs cyclic transmission processing when master station duplication is not detected. If detected, the master station does not perform cyclic transmission processing.

2.3.3.3 Cyclic Transmission

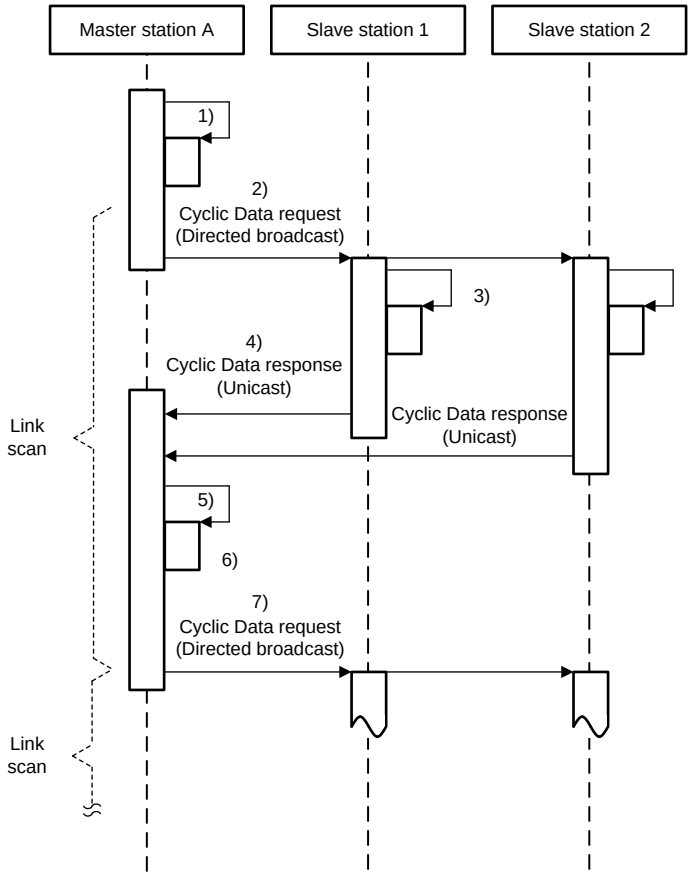


図 6. Cyclic Transmission

The processing is performed in the following sequence:

1. The master station creates cyclic data RY (Remote IO Request bits) and RWw (Remote Register words) before starting a link scan.
2. The master station sends the Cyclic Data command request using a directed broadcast.
3. After receiving a Cyclic Data command request, each slave station transfers the station's specific cyclic data from the request.
4. Each slave station creates its cyclic RX (Remote IO Response bits) and RWr (Remote Register word) data and sends the Cyclic Data command response through a unicast.
5. The master station receives the response from all slave stations with a cyclic transmission status bit turned on. When the constant link scan is used, the master station waits until the constant link scan time elapses.
6. The master station transfers the cyclic data RWr and RX from the Cyclic Data command response and creates the new cyclic data RY and RWw for the next cyclic transmission.
7. Steps 2 to 6 repeat. When multiple groups exist, after step 1 completes, steps 2 to 6 repeat independently for each group.

The upper limit of the link scan time is the total of the response waiting time and the processing time for completion of link scans, such as device transfers. The slave stations process the cyclic data of the Cyclic Data command requests as valid data when the cyclic transmission status bit of the own station is turned on (cyclic data is acquired).

The slave station does not return any command response if the Cyclic Data command request does not include the slave station ID of the own station. If the master station receives a command response from the slave station with a cyclic transmission status bit turned on, and the frame sequence number corresponds with the value of the request message, the master station processes the cyclic data as valid data (cyclic data is acquired).

2.3.4 State Transition

The general state transition of the master station and slave stations of IEF Basic is shown in the following subsections.

2.3.4.1 State Transition of Master Station

The status of the master station consists of a group status and an individual status of each slave station (sub status).

図 7 shows the state transition of a group status of a master station.

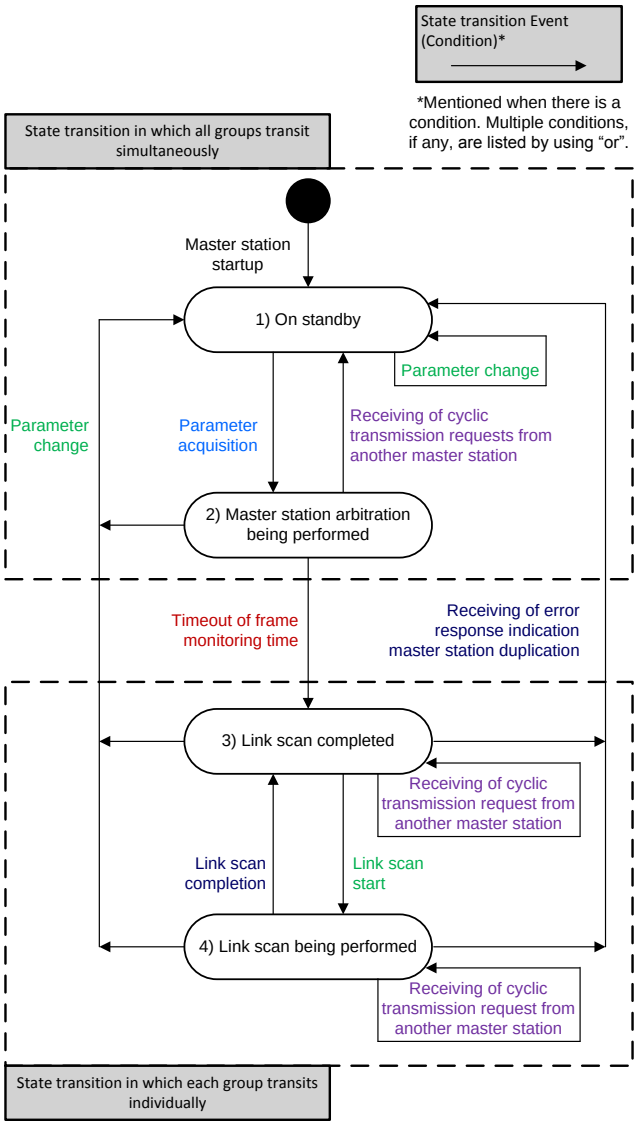


図 7. State Transition of Group Status of Master Station

The group status is the status of each group. When multiple groups exist, there are multiple group statuses and two types of state transition—transition where all groups transit simultaneously and where each group transits individually. Each group status connects with the individual status of each slave station belonging to each group.

図 8 shows the state transition for an individual status of each slave station.

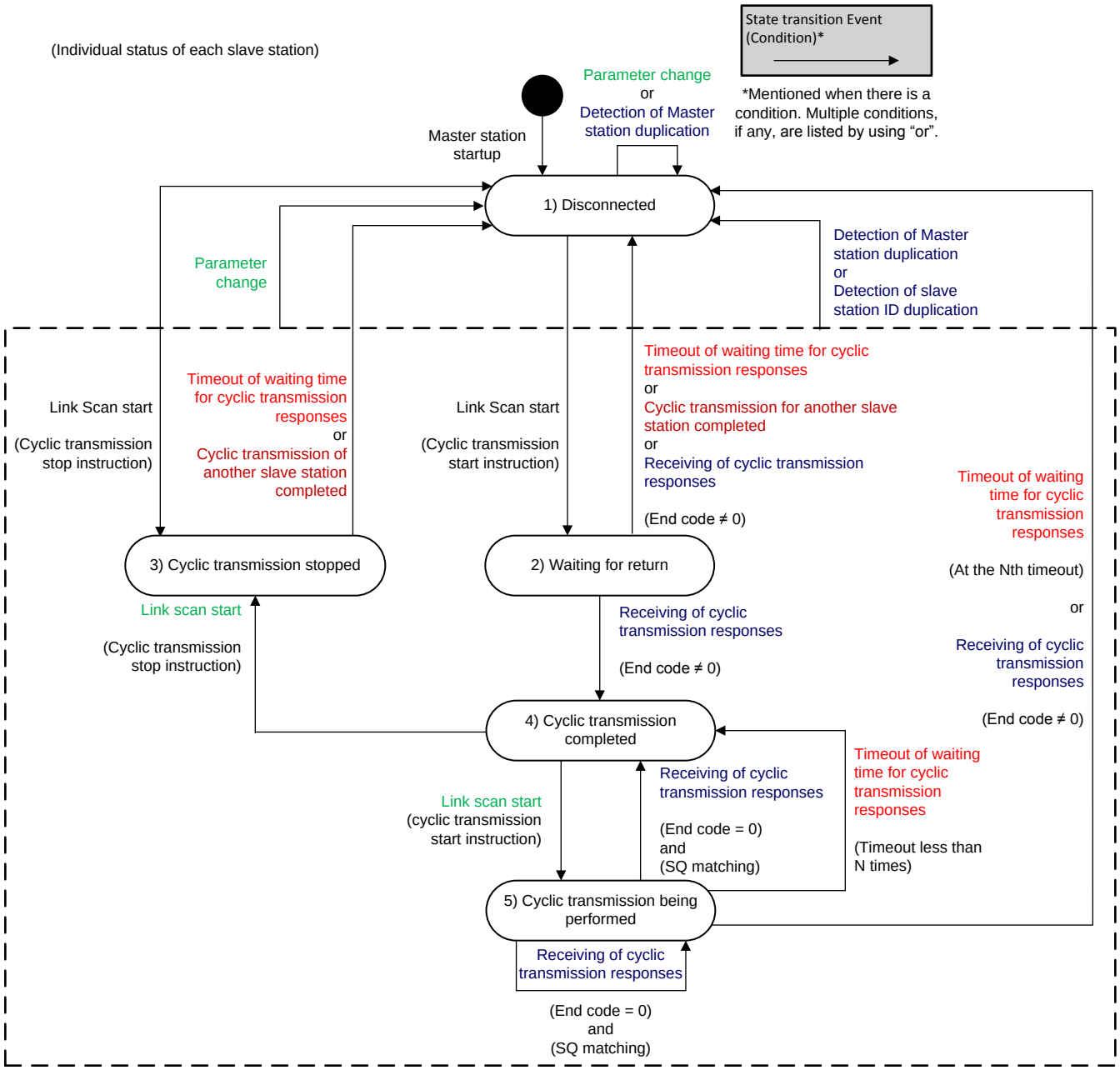


図 8. State Transition Diagram for Individual Status of Each Slave Station Possessed

The master station possesses the individual status of each slave station to control for the number of connected devices. Each status connects with the group status of the group that each slave station belongs.

2.3.4.2 State Transition of Slave Station

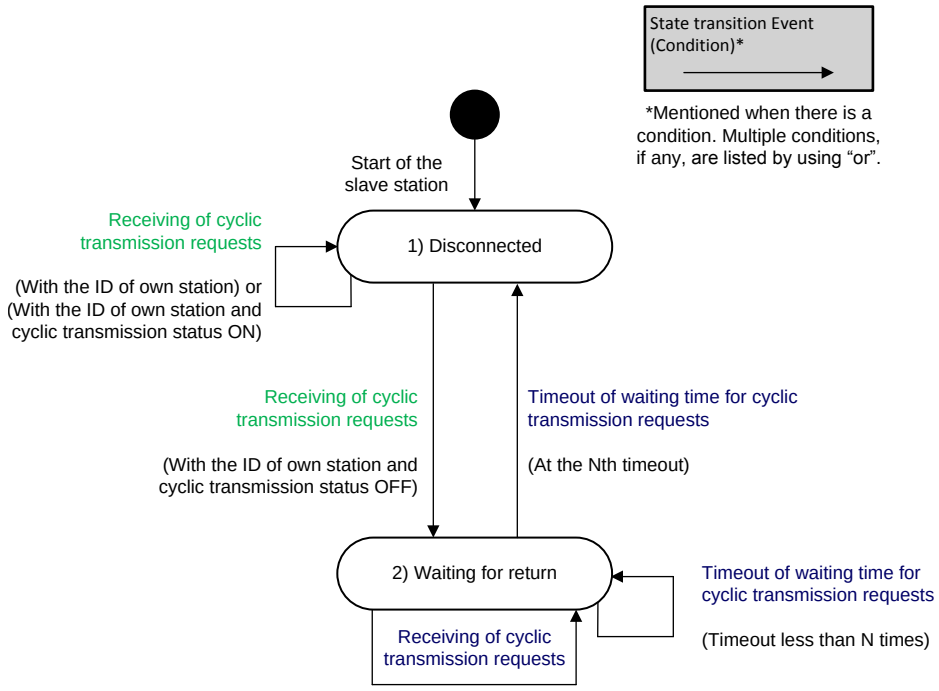


図 9. State Transition Diagram of Slave Station

3 Hardware, Software, Testing Requirements, and Test Results

3.1 Required Hardware and Software

3.1.1 Hardware

This reference design requires the following:

- AM3359 Sitara Processor (as shown in 図 10)

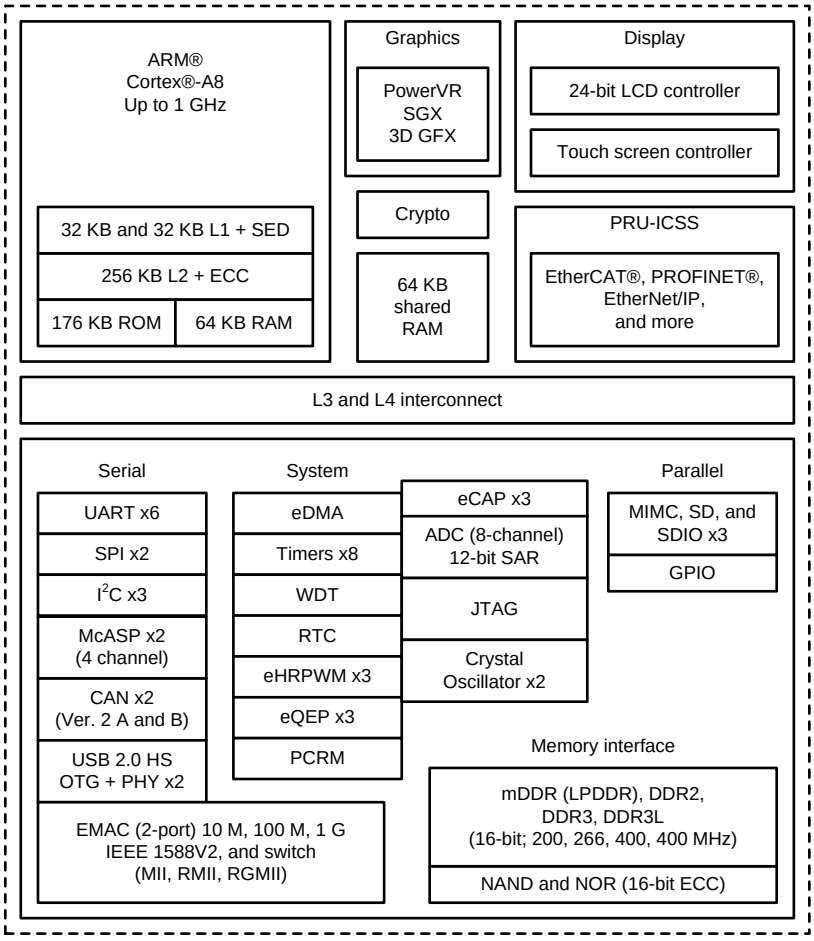


図 10. Functional Block Diagram of AM335x SOC

- DP83822I Transceiver

- TMDSICE3359 ICE EVM (as shown in 図 11)

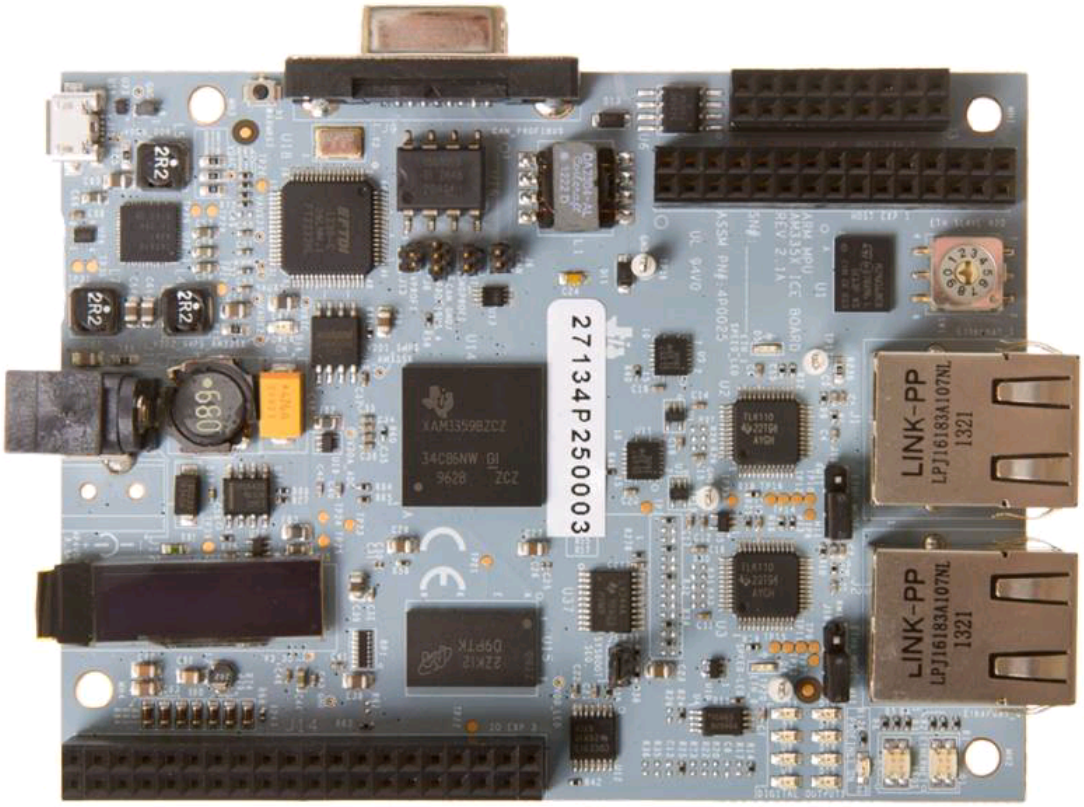


図 11. TMDSICE3359 ICE EVM

3.1.1.1 Additional EVMs Supported

In addition to icev2AM335x board, the IEF Basic master and slave station example also supports other EVMs. 表 1 details the additional supported EVMs.

表 1. Additional EVMs Supported

DEVICE	EVM	IEF Basic on EMAC		IEF Basic on PRU-ICSS	
		Linux	RTOS	Linux	RTOS
AM572x	AM572x EVMxx	X	X		
	AM572x IDK	X	X	X	X
AM571x	AM571x IDK	X	X	X	X
AM437x	AM437x EVM	X	X		
	AM437x SK	X	X		
	AM437x IDK	X	X	X	X
AM335x	AM335x EVM	X	X		
	AM335x BeagleBoneBlack	X	X		
	AM335x SK	X	X		
	AM335x ICE	X	X	X	X
K2G	K2G EVM	X	X		
	K2G ICE EVM	X	X	X	X

3.1.2 Software

3.1.2.1 CC-Link in Processor SDK RTOS

3.1.2.1.1 Software Stack

In *Processor SDK RTOS* the ARM application creates the OS (TI-RTOS) task for supporting various server end functionality.[20] The application creates the network stack for basic networking functionality using NDK. The application then initializes the PRU-ICSS subsystem for NIMU_ICSS and CPSW for NIMU.

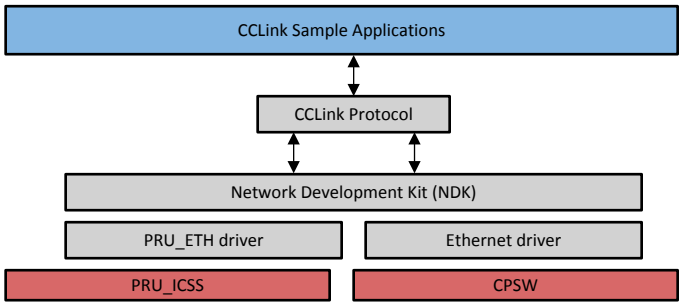


図 12. CC-Link IEF Basic Software Stack

3.1.2.1.1.1 RTOS Adaptation

The general available IEF Basic source code supports Windows® and Linux operating systems. Required modifications enable IEF Basic on RTOS. Most of the modifications are done the on *Hardware Abstraction* layer. The following is the list of changes made. RTOS has a default IP address as *192.168.3.10* for the slave station and *192.168.3.100* for the master station. In order to change this parameter, the RTOS application *.cfg files must be updated.

1. Socket
   - The network layer for RTOS is different than Windows and Linux. The network layer is provided by NDK layer for RTOS. TI NDK is compatible with standard BSD socket layer. All the network functionalities are supported using NDK stack.
2. RTC
   - In case of Windows and Linux, the timing information is extracted from RTC call in both OS. RTOS provides an abstraction layer for RTC call. RTOS configures the timers available in the SOC.
3. SYSBIOS
   - RTOS requires a top-level application to first configure the EVM parameters and set up the board. The application creates the NDK stack and the system configuration. The application then creates a task for IEF Basic application, which runs on top of the stack.
4. UART
   - RTOS provides the output to be printed on UART console.

3.1.2.1.2 Run CC-Link IEF Basic Sample RTOS Application

The following software is required:

- Code Composer Studio™ (CCS) v6 or higher
- PRU Compiler for CCSv6 (install through CCS add-on)
- PROCESSOR SDK RTOS AM335X

Software setup:

1. Install CCS development tool.
2. Install PROCESSOR SDK RTOS AM335X.
3. Create application projects depending on target application.
4. Import IEF Basic application project into CCS. Connect to one of the boards, and run the master application example. Connect to the other board, and run the slave application on it.
5. Output result will be printed on UART console.

注: Review 6 for wiki links with additional details.

3.1.2.2 CC-Link in Processor SDK Linux

3.1.2.2.1 Software Stack

In Processor SDK Linux, the IEF Basic application runs on top of the Linux networking, which can be based on either EMAC (CPSW) or PRU-ICSS.

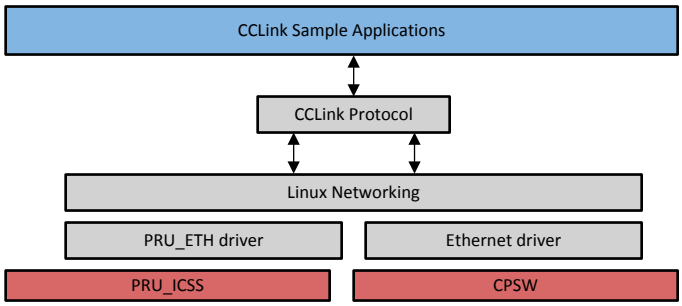


図 13. CC-Link IEF Basic Software Stack

3.1.2.2.2 Run CC-Link IEF Basic Sample Linux Application

The following software is required:

- PROCESSOR SDK Linux RT AM335X

Software setup:

1. Download PROCESSOR SDK Linux RT AM335X.
2. Follow the instructions on the wiki pages to create SD cards.
   - For a Windows machine, follow the instructions in *Processor SDK Linux Creating a SD Card with Windows*.
   - For a Linux machine, follow the instructions in *Processor SDK Linux create SD card script*.
3. Follow the instructions at *Processor SDK Linux CCLINK* to obtain the source code of the IEF Basic master and slave sample applications.

To run the IEF Basic sample application:

1. Boot the two icev2AM335x boards with the SD cards inserted.
2. On the master icev2AM335x board, modify *Slave1 IP address* in MasterParameter.csv to use the IP address of the slave icev2AM335x board.
3. Run *Master_sample* application on the master icev2AM335x board.
4. Run *Slave_sample* application on the slave icev2AM335x board.

3.2 Testing and Results

3.2.1 Test Setup

図 14 shows the test setup for the IEF basic master and slave application running on icev2AM335x board.

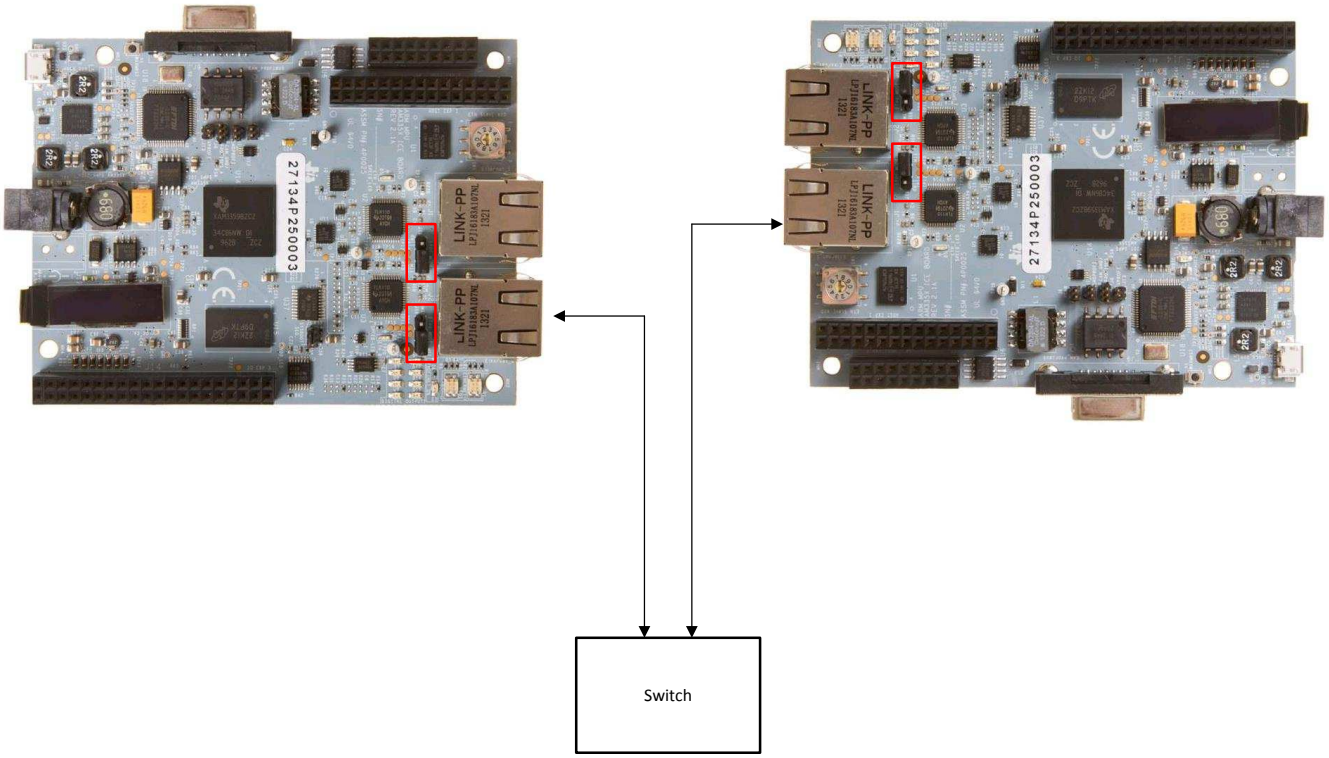


図 14. Hardware Test Setup

Connect port 0 of the design board with an Ethernet cable to a standard switch for both master and slave station. Make sure the jumper setting is correct and based on the type of application demonstrated.

- If running with EMAC, connect the jumper J18 and J19 for both boards into EMAC mode. Hence, connect pin1 and pin2.
- If running with PRU_ICSS, connect the jumper J18 and J19 for both boards into ICSS mode. Hence, connect pin2 and pin3.

In order to get best performance result, do not make any other connection with the switch or hub.

3.2.2 Test Results

3.2.2.1 RTOS

3.2.2.1.1 Sample Output

The following figures show the displays on the UART console when the link is up and communication takes place between slave and master. The default configuration of master and slave would be printed in their respective port.

図 15 shows the master UART console.

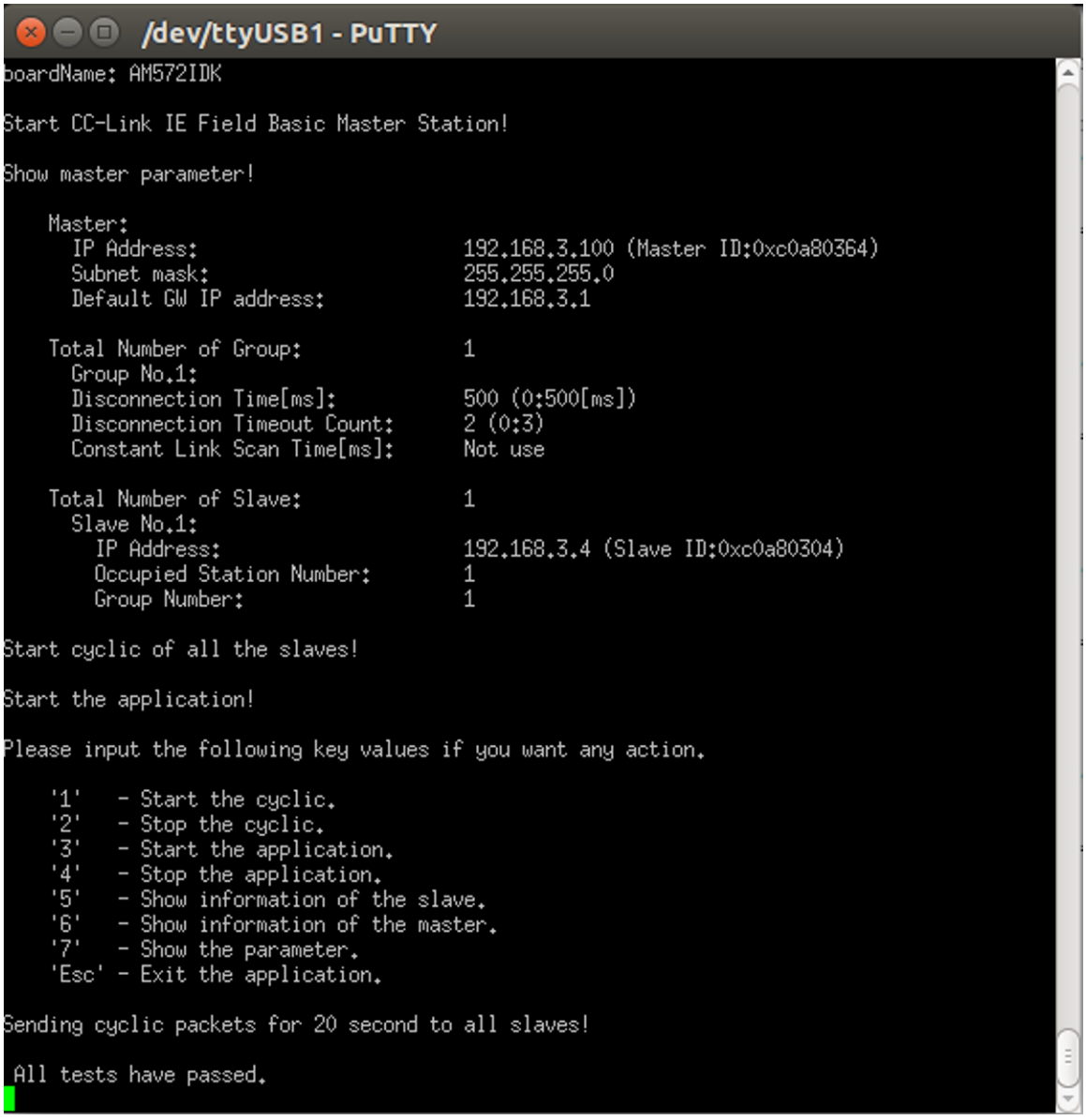


図 15. Master UART Console

図 16 shows the slave UART console.

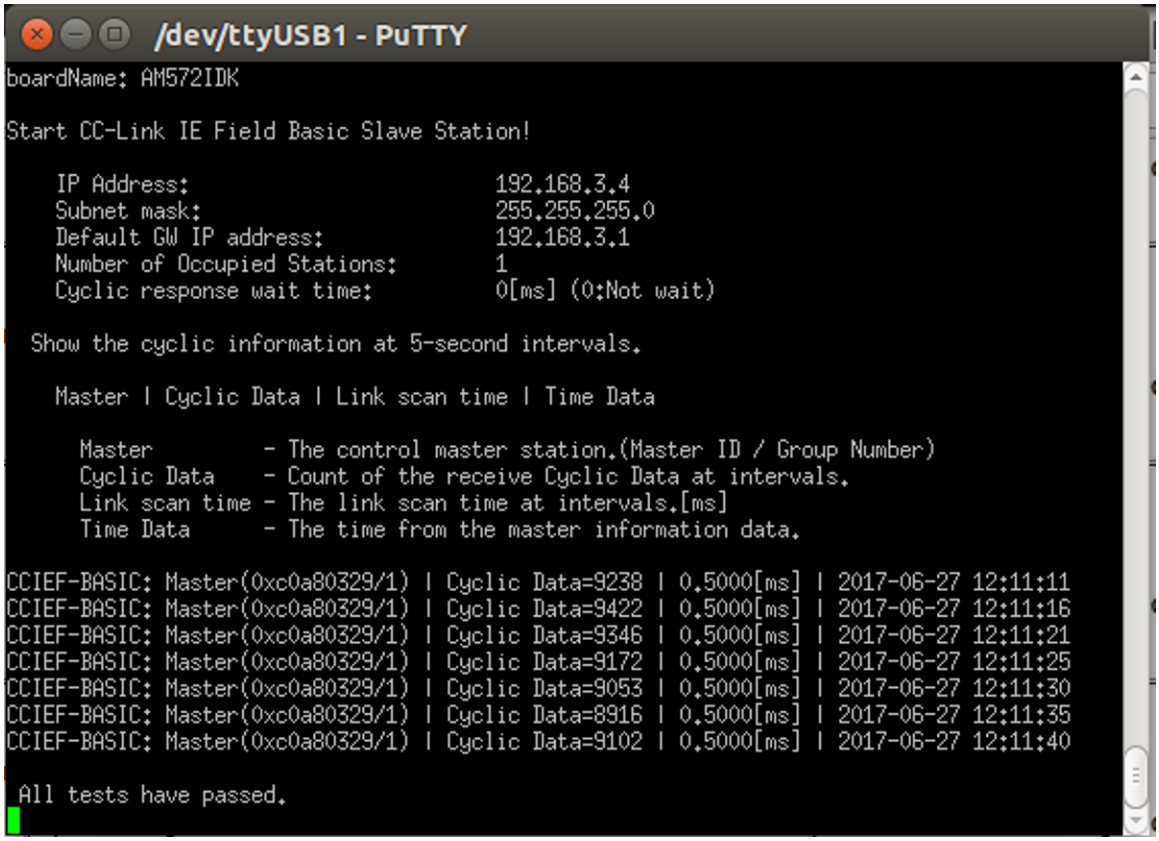


図 16. Slave UART Console

3.2.2.1.2 Compliance Testing

Every IEF Basic application when demonstrated on any platform has to pass the conformance testing for various functionalities of IEF Basic. The conformance test results are sent to CLPA for approval. Upon approval from CLPA, the platform is accepted as CC-Link IEF Basic complaint. See the conformance test results for TI EVMs at *Processor SDK RTOS CCLINK*.

3.2.2.2 Linux

3.2.2.2.1 Sample Output

The following figures show the displays on the console when communication takes place between the slave and master boards. The default configuration of master and slave would be printed in their respective console.

図 17 shows the master console for the master board.

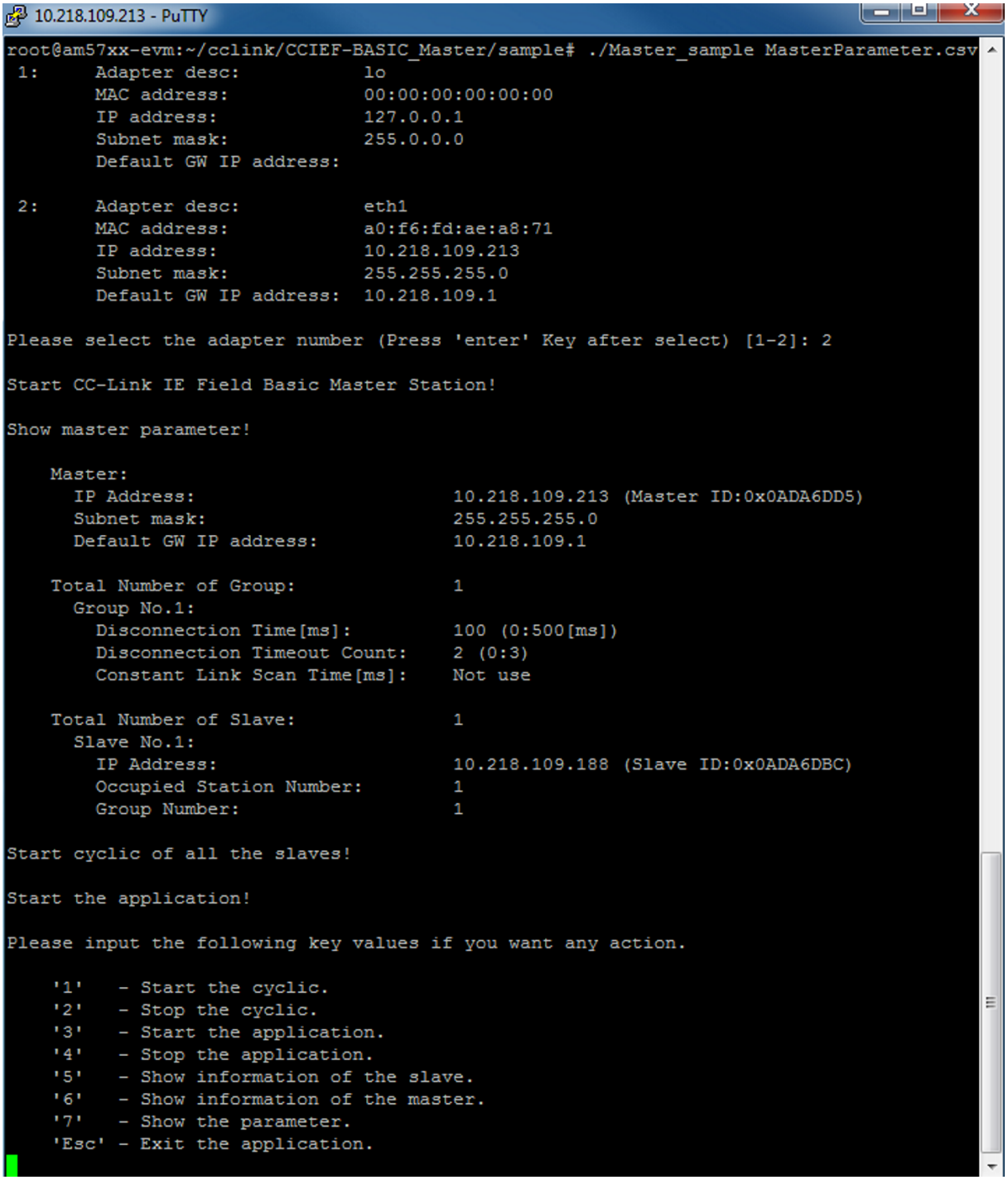


図 17. Master Console

図 18 shows the slave console for the slave board.

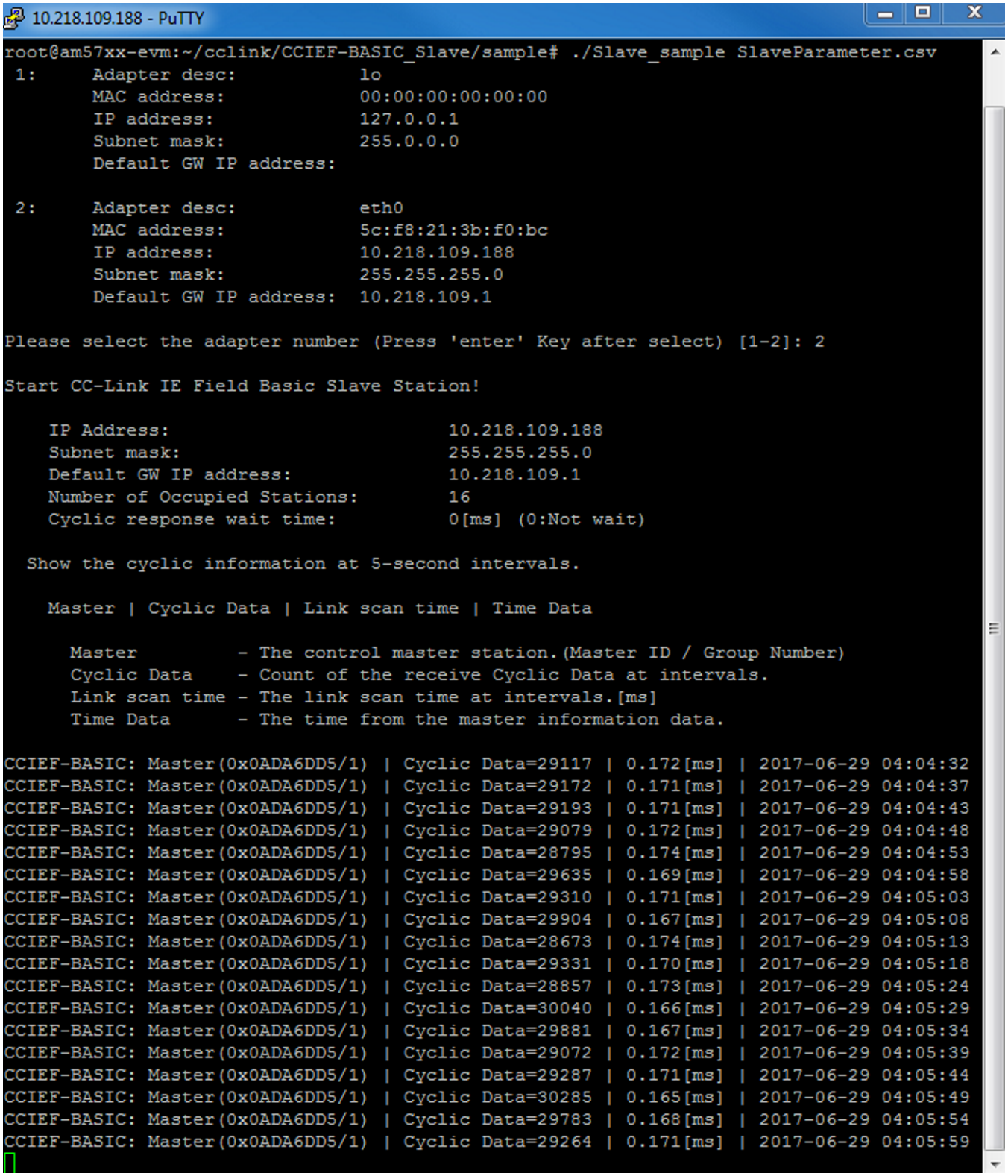


図 18. Slave Console

3.2.2.2.2 Compliance Testing

See the conformance testing results for icev2AM335x with Processor SDK Linux at *Processor SDK Linux CCLINK*.

3.2.2.3 Additional EVMs

Follow the same procedure as mentioned in 3.2.2.1 and 3.2.2.2 for both RTOS and Linux.

3.2.2.3.1 Compliance Testing

3.2.2.3.1.1 RTOS

See the conformance test results for TI EVMs with Processor SDK RTOS at *Processor SDK RTOS CCLINK*.

3.2.2.3.1.2 Linux

See the conformance test results for TI EVMs with Processor SDK Linux at *Processor SDK Linux CCLINK*.

4 Design Files

4.1 Schematics

To download the schematics, see the design files at TIDEP-0089.

4.2 Bill of Materials

To download the bill of materials (BOM), see the design files at TIDEP-0089.

4.3 PCB Layout Recommendations

4.3.1 Layout Prints

To download the layer plots, see the design files at TIDEP-0089.

4.4 Altium Project

To download the Altium project files, see the design files at TIDEP-0089.

4.5 Gerber Files

To download the Gerber files, see the design files at TIDEP-0089.

4.6 Assembly Drawings

To download the assembly drawings, see the design files at TIDEP-0089.

5 Software Files

To download the software files, see the design files at TIDEP-0089.

6 Related Documentation

1. Texas Instruments, *AM335x Sitara Processors*, Datasheet (SPRS717)
2. Texas Instruments, *DP83822 Robust, Low Power 10/100 Mbps Ethernet Physical Layer Transceiver*, Datasheet (SNLS505)
3. Texas Instruments, *Processor SDK RTOS IEF Basic*, Wiki
4. Texas Instruments, *Download CCS*, Code Composer Studio TI Wiki
5. Texas Instruments, *CCS Getting Started Guide*, Wiki
6. Texas Instruments, *Processor SDK RTOS_CCS_Setup*, Wiki
7. Texas Instruments, *Creating and importing examples in Processor SDK RTOS*, Wiki
8. Texas Instruments, *Processor SDK RTOS AM335X*, Download Page
9. Texas Instruments, *Processor SDK RTOS AM437X*, Download Page
10. Texas Instruments, *Processor SDK RTOS K2G*, Download Page
11. Texas Instruments, *Processor SDK RTOS AM57X*, Download Page
12. Texas Instruments, *Processor SDK Linux IEF Basic Wiki Page*, Wiki
13. Texas Instruments, *SDK Linux create SD card script*, Wiki
14. Texas Instruments, *Processor SDK Linux Creating a SD Card with Windows*, Wiki
15. Texas Instruments, *Processor SDK Linux RT AM335X*, Download Page
16. Texas Instruments, *Processor SDK Linux RT AM437X*, Download Page
17. Texas Instruments, *Processor SDK Linux RT K2G*, Download Page
18. Texas Instruments, *Processor SDK Linux RT AM57X*, Download Page
19. Texas Instruments, *Processor SDK Software Page*, Product Page
20. Texas Instruments, *Category:SYSBIOS*, Wiki
21. CC-Link Partner Association, *CLPA Reference Material and Support*

6.1 商標

7 Terminology

- CCS - Code Composer Studio
- ICSS - Industrial communication system
- PLC - Programmable logic controller
- PRU - Programmable real-time unit

8 About the Author

SURAJ DAS is a Software Engineer at Texas Instruments, where he is responsible for developing PRU-ICSS based solution for the Catalog segment. Suraj brings to this role his extensive experience in Computer architecture & PRU cores, and has supported Catalog RTOS SDK release for various peripherals. Suraj earned his Master of Engineering degree in Computer Engineering from Virginia Tech in Blacksburg, VA.

TIの設計情報およびリソースに関する重要な注意事項

Texas Instruments Incorporated ("TI")の技術、アプリケーションその他設計に関する助言、サービスまたは情報は、TI製品を組み込んだアプリケーションを開発する設計者に役立つことを目的として提供するものです。これにはリファレンス設計や、評価モジュールに関係する資料が含まれますが、これらに限られません。以下、これらを総称して「TIリソース」と呼びます。いかなる方法であっても、TIリソースのいずれかをダウンロード、アクセス、または使用した場合、お客様(個人、または会社を代表している場合にはお客様の会社)は、これらのリソースをここに記載された目的にのみ使用し、この注意事項の条項に従うことに合意したものとします。

TIによるTIリソースの提供は、TI製品に対する該当の発行済み保証事項または免責事項を拡張またはいかなる形でも変更するものではなく、これらのTIリソースを提供することによって、TIにはいかなる追加義務も責任も発生しないものとします。TIは、自社のTIリソースに訂正、拡張、改良、およびその他の変更を加える権利を留保します。

お客様は、自らのアプリケーションの設計において、ご自身が独自に分析、評価、判断を行う責任がお客様にあり、お客様のアプリケーション(および、お客様のアプリケーションに使用されるすべてのTI製品)の安全性、および該当するすべての規制、法、その他適用される要件への遵守を保証するすべての責任をお客様のみが負うことを理解し、合意するものとします。お客様は、自身のアプリケーションに関して、(1) 故障による危険な結果を予測し、(2) 障害とその結果を監視し、および、(3) 損害を引き起こす障害の可能性を減らし、適切な対策を行う目的での、安全策を開発し実装するために必要な、すべての技術を保持していることを表明するものとします。お客様は、TI製品を含むアプリケーションを使用または配布する前に、それらのアプリケーション、およびアプリケーションに使用されているTI製品の機能性を完全にテストすることに合意するものとします。TIは、特定のTIリソース用に発行されたドキュメントで明示的に記載されているもの以外のテストを実行していません。

お客様は、個別のTIリソースにつき、当該TIリソースに記載されているTI製品を含むアプリケーションの開発に関連する目的でのみ、使用、コピー、変更することが許可されています。明示的または黙示的を問わず、禁反言の法理その他どのような理由でも、他のTIの知的所有権に対するその他のライセンスは付与されません。また、TIまたは他のいかなる第三者のテクノロジまたは知的所有権についても、いかなるライセンスも付与されるものではありません。付与されないものには、TI製品またはサービスが使用される組み合わせ、機械、プロセスに関連する特許権、著作権、回路配置利用権、その他の知的所有権が含まれますが、これらに限られません。第三者の製品やサービスに関する、またはそれらを参照する情報は、そのような製品またはサービスを利用するライセンスを構成するものではなく、それらに対する保証または推奨を意味するものでもありません。TIリソースを使用するため、第三者の特許または他の知的所有権に基づく第三者からのライセンス、あるいはTIの特許または他の知的所有権に基づくTIからのライセンスが必要な場合があります。

TIのリソースは、それに含まれるあらゆる欠陥も含めて、「現状のまま」提供されます。TIは、TIリソースまたはその仕様に関して、明示的か暗黙的かにかかわらず、他のいかなる保証または表明も行いません。これには、正確性または完全性、権原、続発性の障害に関する保証、および商品性、特定目的への適合性、第三者の知的所有権の非侵害に対する黙示の保証が含まれますが、これらに限られません。

TIは、いかなる苦情に対しても、お客様への弁護または補償を行う義務はなく、行わないものとします。これには、任意の製品の組み合わせに関連する、またはそれらに基づく侵害の請求も含まれますが、これらに限られず、またその事実についてTIリソースまたは他の場所に記載されているか否かを問わないものとします。いかなる場合も、TIリソースまたはその使用に関連して、またはそれらにより発生した、実際的、直接的、特別、付随的、間接的、懲罰的、偶発的、または、結果的な損害について、そのような損害の可能性についてTIが知らされていたかどうかにかかわらず、TIは責任を負わないものとします。

お客様は、この注意事項の条件および条項に従わなかったために発生した、いかなる損害、コスト、損失、責任からも、TIおよびその代表者を完全に免責するものとします。

この注意事項はTIリソースに適用されます。特定の種類の資料、TI製品、およびサービスの使用および購入については、追加条項が適用されます。これには、半導体製品(http://www.ti.com/sc/docs/stdterms.htm)、評価モジュール、およびサンプル(http://www.ti.com/sc/docs/sampterms.htm)についてのTIの標準条項が含まれますが、これらに限られません。