TI Designs: TIDEP-0096
# *Linux®による変電所自動化用の高可用性シームレス冗長化イーサネットのリファレンス・デザイン*

TEXAS INSTRUMENTS

概要

このイーサネットのリファレンス・デザインは、スマート・グリッド送配電網における変電所自動化機器用の、高信頼性、低レイテンシのネットワーク通信を実装します。このリファレンス・デザインは、IEC 62439規格のHSR（高可用性でシームレスな冗長性）仕様と、IEEE 1588のPTP（高精度タイム・プロトコル）仕様をサポートしています。これは、FPGA手法よりも安価な代替手法で、追加コンポーネントなしにIEC 61850サポートなどの機能を追加できる柔軟性と性能を持ち合わせています。このサポートは、AM335x、AM437x、AM572xを含む広範なデバイス・ファミリ全体に適用されます。このリファレンス・デザインは、Linux®高レベル・オペレーティング・システムをベースとしており、ファクトリ・オートメーション、輸送など、イーサネット・ネットワークで最小レイテンシの信頼性を必要とする他の市場にも適用可能です。

リソース

|	
TIDEP-0096	デザイン・フォルダ
AM572x 、AM437x、AM335X	プロダクト・フォルダ
AM572x IDK、AM437x IDK	ツール・フォルダ
AM335x ICE、プロセッサSDK	ツール・フォルダ

TI E2E™ Community

E2Eエキスパートに質問
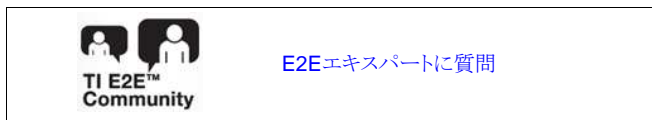
特長

- HSRイーサネット通信のIEC 62439-3 Clause 5仕様に準拠し、H、T、U、Nモードをサポート
- IEEE 1588の通常クロックおよびピアツーピア透過型クロック・プロファイルによるネットワーク同期
- IEEE 1588の境界クロックによるネットワーク同期（AM57xのみ）
- VLAN IDをベースとするトラフィック・フィルタリング、マルチキャストおよびブロードキャストのサポート、ポートごとに構成可能な内蔵のストーム防止機構
- VLAN PCPによるQoS（サービス品質）で8レベルをサポート
- ネットワーク障害時に復旧時間が不要
- デュアル・ポート、全二重の100Mbpsイーサネット
- AM335x、AM437x、AM57xデバイス・ファミリをサポート
- Linux高レベル・オペレーティング・システムをベースとする、完全にプログラム可能なソリューションで、追加アプリケーションを統合するプラットフォームとして機能

アプリケーション

- 変電所および配電オートメーション
- 保護リレー
- スマート・グリッド通信
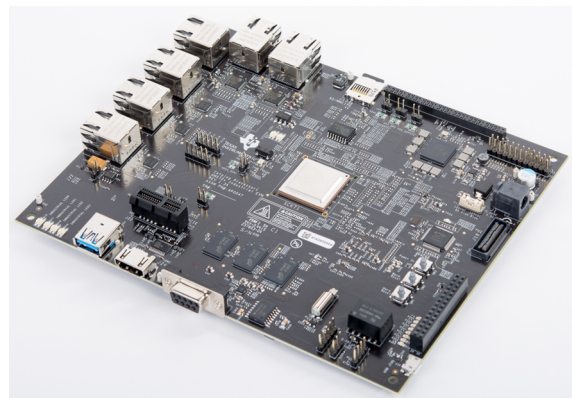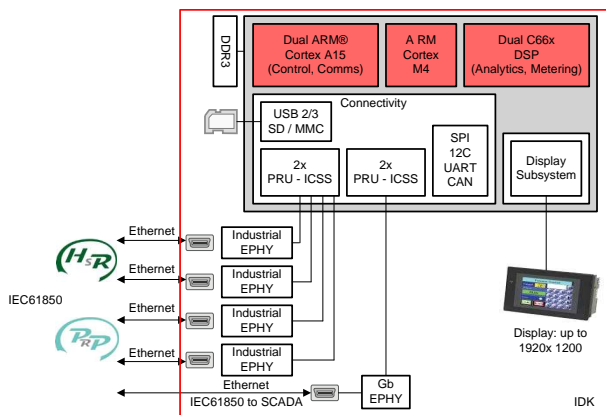
Copyright © 2017, Texas Instruments Incorporated

1 System Description

A substation is a key component of the electricity grid infrastructure, which is located everywhere from power generation facilities throughout the distribution network to the low-voltage feeders serving residences and businesses. Substations play a key role in transforming voltage levels for transmission and performing important functions, such as switching, monitoring, and protecting subsystems, to maintain grid efficiency and reliability. Traditional substation systems focused on fault monitoring, which can be manually fixed by switching to backup subsystems.

Consumers, regulators, and grid operators demand ever-greater reliability of electricity delivery. Automatic switching and protecting subsystems can require substations to start automating operations and communications such as monitoring grid conditions and communicating that information to grid operators reliably and rapidly.

Operators must continually monitor the health of their network and take action to maintain high-speed operation. This leads to the requirement for reliable and low-latency communications between the operator's control center and high-value nodes such as substations.

The International Electro-technical Commission (IEC) released specifications for industrial Ethernet communications under the IEC 62439 standard. The HSR specification is a static redundancy, Ethernet-based protocol, which supports critical real-time systems that require continuous monitoring.

The IEEE 1588 PTP is designed to provide high-accuracy network time synchronization between subsystems.

This design provides a reliable, high-speed, HSR communication solution for substation automation. This design implements HSR-compliant with IEC 62439-3 Clause 5.

This reference design is an affordable alternative to ASIC or FPGA-based Ethernet solutions while delivering equivalent performance. The programmable nature of the solution allows operating different redundancy Ethernet protocols without modifying hardware and adding applications, such as IEC 61850, without requiring extra system cost.

図 1 shows the overall system architecture. The HSR supports dual-port, full-duplex, Ethernet communication between network devices. The system includes Ethernet PHY as layer one and the Network Stack with HSR and PTP as the upper layers. HSR capability can be demonstrated simply with standard Linux applications like ping. RT Linux priorities can be used to meet the requirements of hard, real-time applications to minimize latencies. Standard TCP/IP traffic can use the well-recognized Linux network stack.
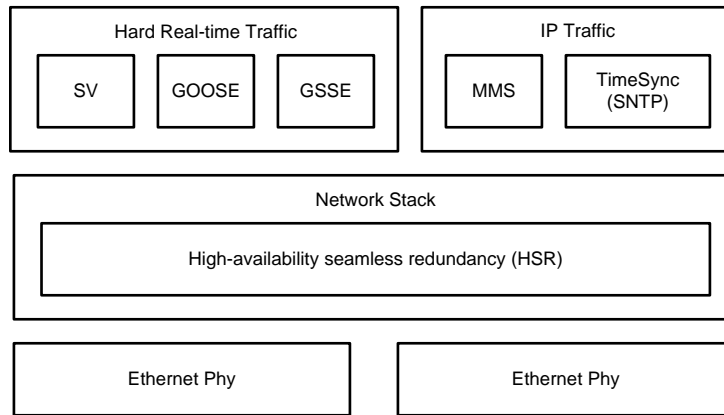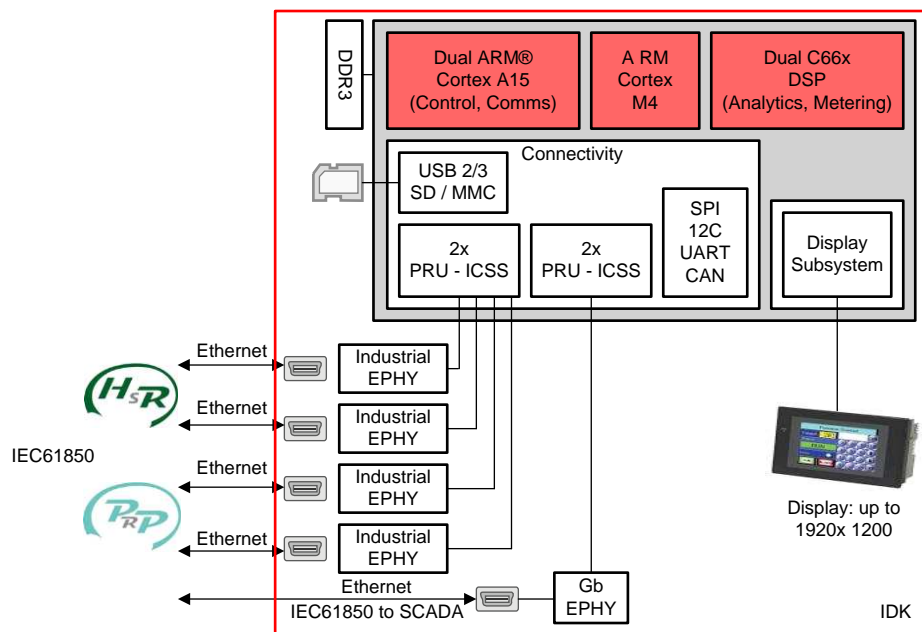
図 1. System Architecture

2 System Overview

2.1 *Block Diagram*

図 2 shows the block diagram. The primary device for this design is the Dual-core AM5728 ARM® Cortex®-A15 microprocessor (MPU) as the host processor to support HSR and user applications running under a real-time Linux environment. This support also scales across the AM335x and AM437x device families as well.

This design uses this highly-integrated device for these benefits:

- The PRU-ICSS subsystem allows independent operation for real-time communication stacks.

- The high-performance ARM cores support the real-time applications for substation automation.

- The programmable, flexible software design allows upgrades to different Ethernet-based redundancy protocols without hardware modification.

Copyright © 2017, Texas Instruments Incorporated

図 2. Block Diagram

Linux®による変電所自動化用の高可用性シームレス冗長化イーサネットのリファレンス・デザイン JAJU396A–October 2017–Revised December 2018

2.2 Design Considerations

2.2.1 HSR

HSR is a redundancy protocol for Ethernet networks, standardized as IEC 62439-3 Clause 5, and is selected as one of the redundancy protocols for substation automation in the IEC 61850 standard. HSR is application-protocol independent and can be used by most industrial Ethernet applications that require reliable high-speed communications.

The HSR supports ring topology. Compared to star topology where typical Ethernet is operating, the advantage of ring topology is that there is no requirement on the infrastructure (for example, router) to form networks, which saves installation cost. A disadvantage to ring topology is a possibility for delays to reach the destination if the packet goes through multiple hops.

In this design, the transmission delay over multiple hops was minimized by introducing cut-through mode. The cut-through mode is when a node receives packets that are partially decoded up to the destination address field, and, if the final destination is not the node, the packets are forwarded to the TX port. In addition, this design includes built-in HSR supervision, a storm prevention mechanism, and VLAN support. 図 3 shows HSR ring topology and how the packet reaches its destination. Once a packet is generated at Node 1, the packet is distributed in both directions until being used by the destination at Node 4. The redundancy provides zero recovery time in case the packet fails to be delivered in one direction.
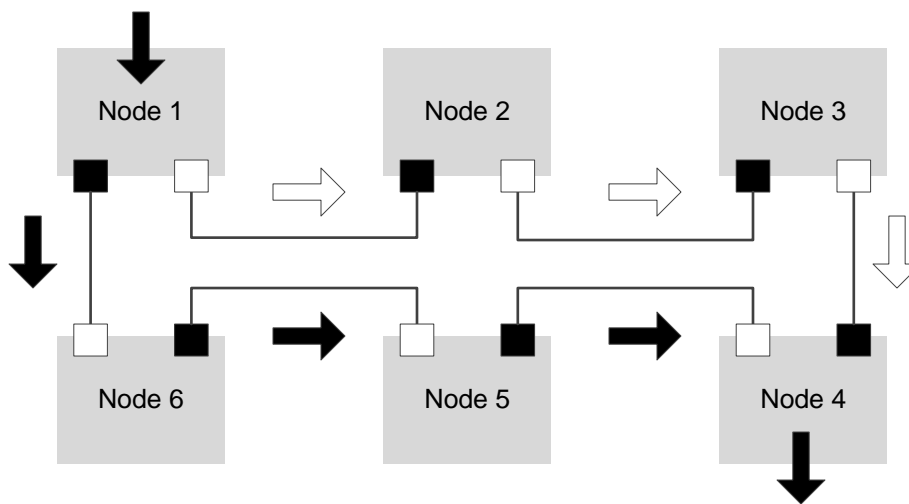

図 3. HSR Ring Topology

2.2.2 IEEE 1588 (PTP)

The IEEE 1588 is a protocol used to provide high-accuracy time synchronization over networks. Originally defined in the IEEE 1588 standard, this is designed to fill a niche not well served by NTP and GPS. In this design with the HSR protocol, IEEE 1588 v2 peer-to-peer transparent clock profile is supported at layer 2 to synchronize network time by measuring mean path delay using peer delay request and response mechanism. The PTP supports transmissions over IEEE 802.3,and only multicast PTP messages shall be used.

2.3 Highlighted Products

2.3.1 TI Sitara™ AM572x Overview

The AM572x brings high-processing performance through the maximum flexibility of a fully-integrated, mixed processor solution. The devices also combine programmable video processing with a broad and highly-integrated peripheral set, which is well suitable for industrial applications. Programmability is provided by dual-core, ARM Cortex-A15 RISC CPUs with ARM NEON™ technology and two TI C66x VLIW, floating-point, digital signal processor (DSP) cores. The ARM Cortex lets developers keep separate control functions from other algorithms that are programmed on the DSPs and coprocessors. The separated control functions reduce the complexity of the system software. The ARM Cortex-A15 CPU supports multiple operating frequencies at a range of up to 1.5 GHz.

The AM572x processor is configured with two, dual-core programmable real-time unit and industrial communication subsystems (PRU-ICSS). The PRU-ICSS can be used for communication protocols, such as EtherCAT® master and slave, PROFINET®, Ethernet/IP™, SERCOS®, HSR, and so forth. The PRU-ICSS is separate from the ARM core, which allows for independent operation and clocking for greater efficiency and flexibility. The PRU-ICSS unit contains two PRUs, each of which includes 32-bit RISC processor capable of running at 200 MHz, that support real-time protocol for HSR and additional interfaces of media independent interface (MII) and reduced media independent interface (RMII) to connect to the Ethernet PHY devices directly.

Additionally, the programmable nature of the PRU-ICSS, along with its access to pins, events, and all system-on-chip (SoC) resources provides flexibility in implementing fast, real-time responses, and specialized data handling.
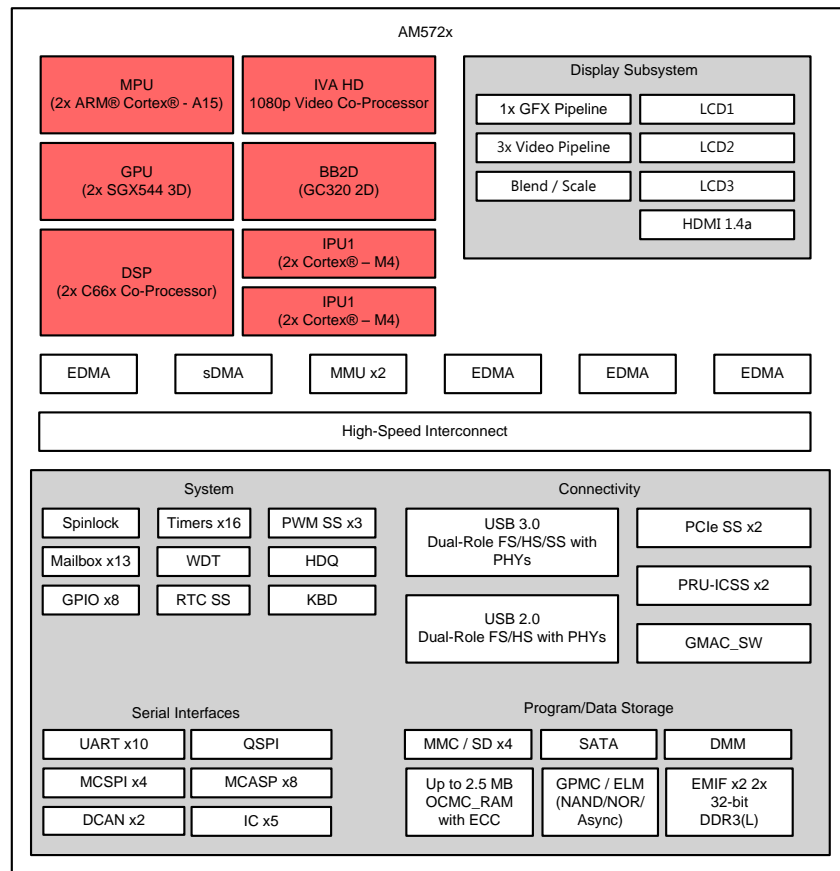
図 4. AM5728 Functional Block Diagram

2.3.2   TI Sitara AM437x Overview

The AM437x high-performance processors are based on the ARM Cortex-A9 core. The processors each provide a rich graphical user interface (GUI). The AM437x has PRU-ICSS co-processors for deterministic, real-time processing including industrial communication protocols, such as EtherCAT®, PROFIBUS®, HSR and others as well as industrial drive protocols such as EnDat, Tamagawa, Sigma Delta, and so forth. The devices support operating systems like Linux, Real-time Linux and TI RTOS. Other RTOSs are available from TI's Design Network and ecosystem partners.

These devices offer an upgrade to systems based on lower performance ARM cores and provide updated peripherals, including memory options such as QSPI-NOR.

High-performance interconnects provide high-bandwidth data transfers for multiple initiators to the internal and external memory controllers and to on-chip peripherals. The device also offers a comprehensive clock-management scheme.

One on-chip analog-to-digital converter (ADC0) can couple with the display subsystem to provide an integrated touch-screen solution. The other ADC (ADC1) can combine with the pulse width module to create a closed-loop motor control solution.

The RTC provides a clock reference on a separate power domain. The clock reference enables a battery-backed clock reference. The camera interface offers configuration for a single- or dual-camera parallel port. Cryptographic acceleration is available in every AM437x device. Secure boot is available only on AM437xHS devices for anticloning and illegal software update protection.

図 5 shows the AM437x block diagram.
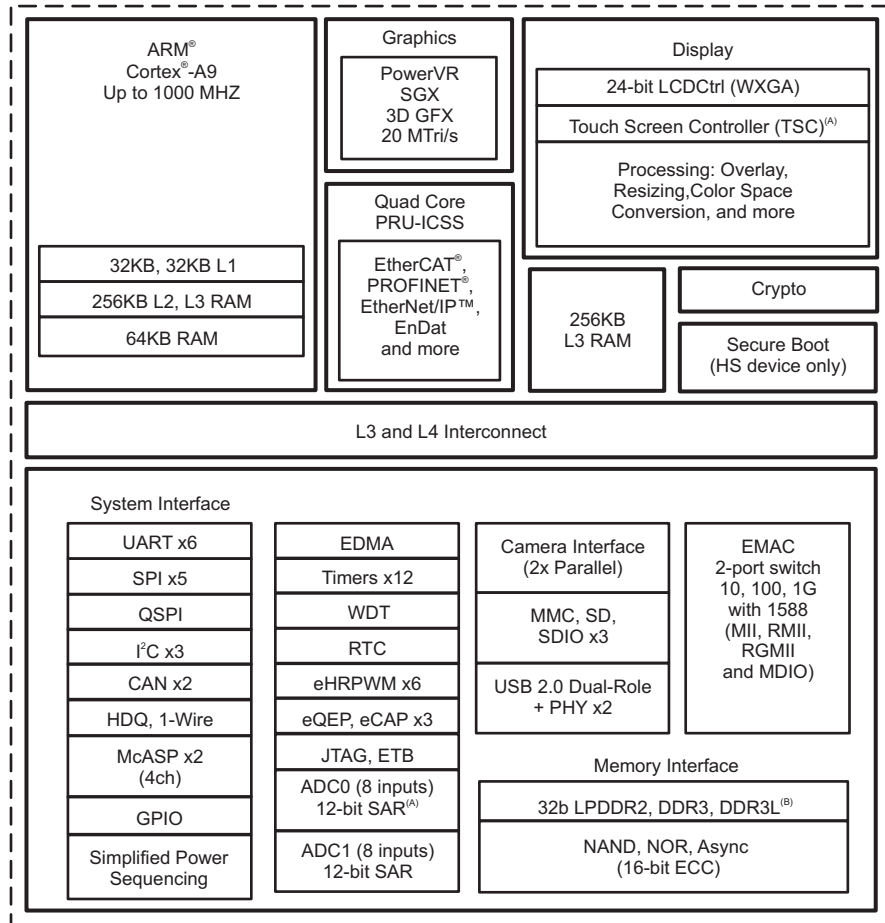
Copyright © 2017, Texas Instruments Incorporated

図 5. AM437x Block Diagram

2.3.3 TI Sitara AM3359 Overview

The AM3359 microprocessors (based on the ARM Cortex-A8 processor) are enhanced with image, graphics processing, peripherals, and an industrial-interface option for PRP and high-availability seamless redundancy (HSR). The PRU-ICSS is separate from the ARM core, allowing for independent operation and clocking for increased efficiency and flexibility. The PRU-ICSS unit contains two PRUs; each PRU includes a 32-bit RISC processor capable of running at 200 MHz that supports real-time protocol for PRP and HSR. The PRUs support additional media-independent interfaces (MIIs) and reduced media-independent interfaces (RMIIs) to connect to the Ethernet PHY devices directly.

The programmable nature of the PRU-ICSS, the access to pins, events, and all system-on-chip (SoC) resources provide flexibility when implementing fast, real-time, specialized-data handling (see 図 6).
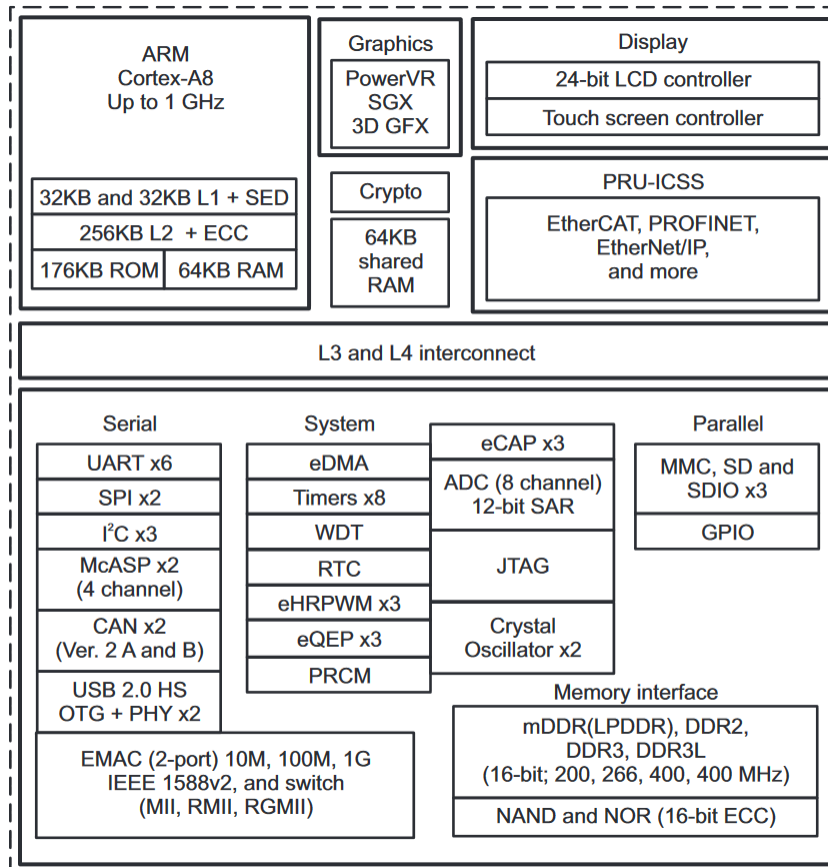

図 6. AM3359 Functional Block Diagram

2.3.4 PRU-ICSS

The Programmable Real-Time Unit and Industrial Communication Subsystem (PRU-ICSS) is separate from the ARM core and allows independent operation and clocking for greater efficiency and flexibility. The PRU-ICSS enables additional peripheral interfaces and real-time protocols such as EtherCAT, PROFINET IRT ®, EtherNet/IP™, PROFIBUS, Ethernet POWERLINK ™, Sercos III ™, HSR, PRP and others. The second PRU-ICSS subsystem of the AM437x enables EnDat 2.2, Tamagawa, Sigma Delta and another industrial communication protocol in parallel. Additionally, the programmable nature of the PRU-ICSS, along with their access to pins, events, and all SoC resources, provides flexibility in implementing fast real-time responses, specialized data-handling operations, custom peripheral interfaces, and off-loading tasks from the other processor cores of the SoC.

図 7 shows an example PRU-ICSS block diagram.

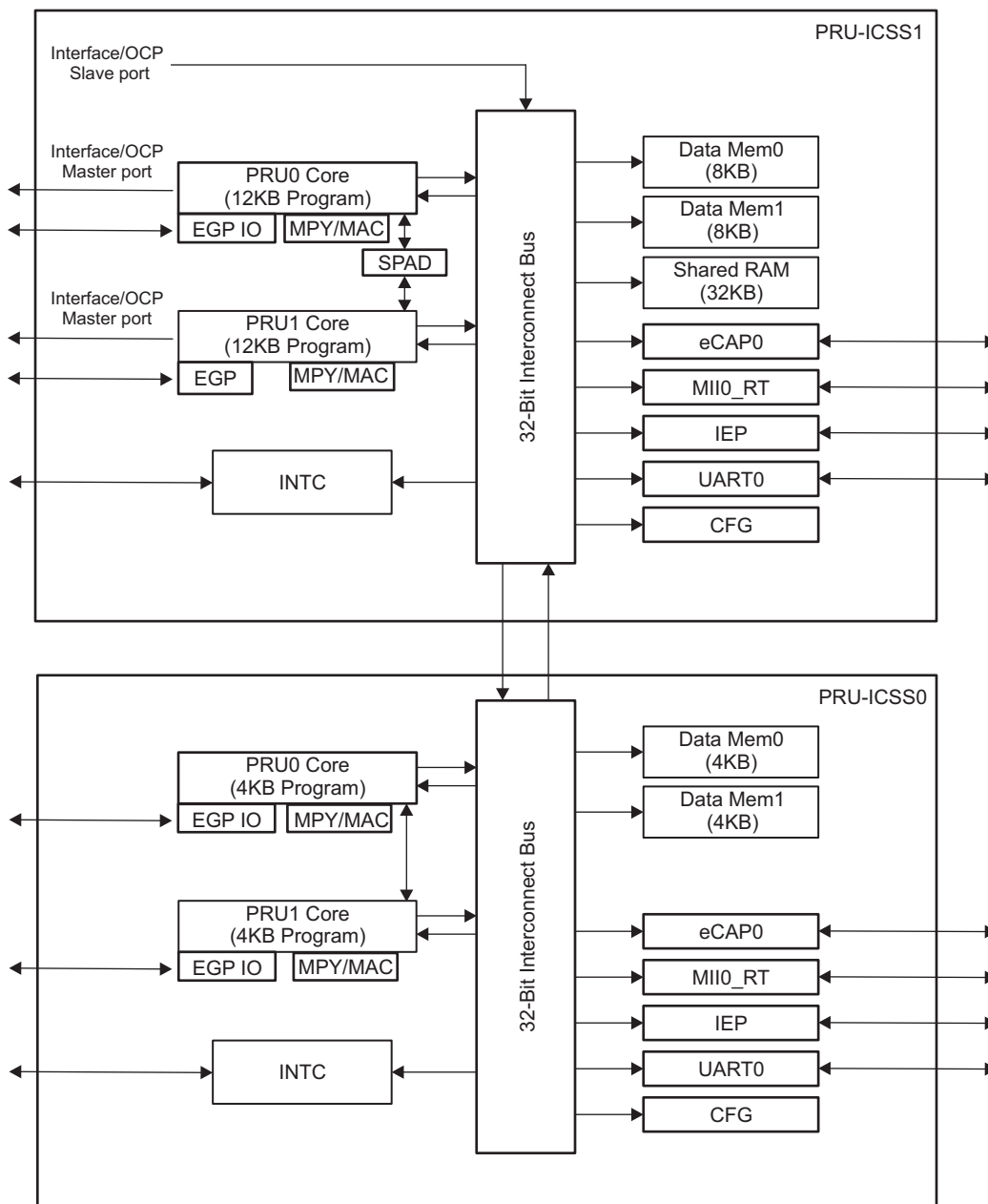**図 7. PRU-ICSS Block Diagram**

3 Hardware, Software, Testing Requirements

3.1 *Required Hardware and Software*

3.1.1 Hardware

図 8 shows the AM572x Industrial Development Kit (IDK) revision 1.3B. In addition to the AM572x processor, the EVM includes Ethernet PHYs, various storage devices, DDR memory, and power management support. The EVM is designed to support multiple communication standards by providing various interfaces such as Ethernet, CAN, and RS-485. The AM335x and AM437x families also have EVMs that are suitable for evaluating HSR/PRP. Links to these boards are provided in the リソース section.
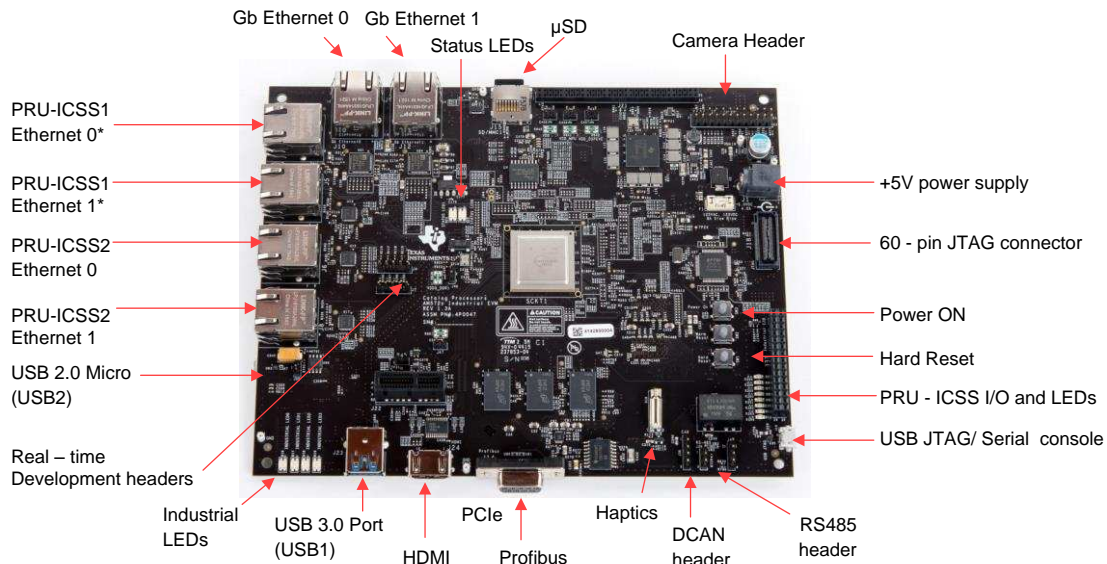
図 8. AM572x IDK Rev 1.3B

3.1.1.1   *Three-Node Setup Example*

The simplest HSR ring could be set up with just two EVMs. However, this would not allow evaluation of the latency across intermediate nodes in a ring, which is usually a critical factor in these architectures. Therefore, a minimum of a three-node ring is advantageous. Each node has two Ethernet ports, and there is no specific requirement on which ports must be connected between nodes. 図 9 shows an example of a three-node setup. For testing purposes, nodes can be connected to a serial terminal program with the baud rate of 115,200 bps by default.
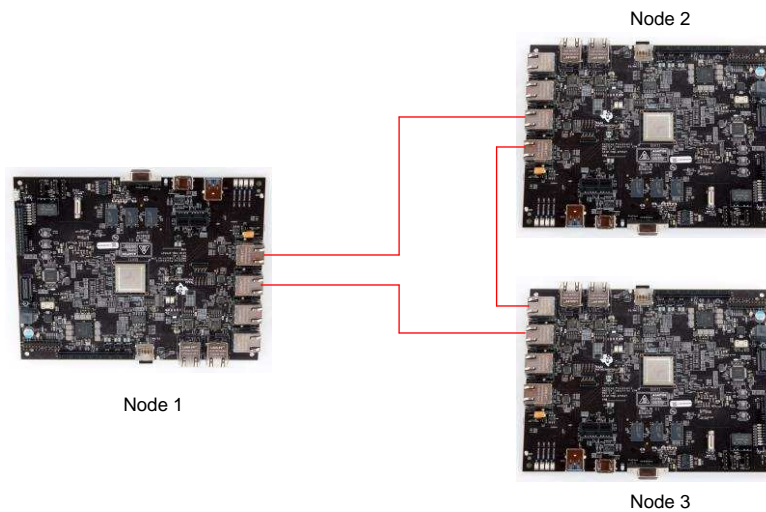

図 9. Three-Node Network Setup

3.1.2   Software

This section provides step-by-step procedure to develop an application using the HSR protocol based on the Processor SDK for real-time Linux for Sitara processors. All the software required for this setup is included in the SDK. The SDK can be downloaded here. The download page includes useful links to documentation to help users get started quickly.

3.1.2.1 *Overview*

図 10 provides a high-level diagram of the Linux HSR implementation. HSR is added to the Linux Kernel to provide redundant Ethernet interfaces. These interfaces are abstracted to the application layer, which requires no changes to take advantage of HSR. For example, a simple ping command can send data over both interfaces. If a network cable were to be compromised on one of the interfaces, the ping still succeeds on the other interface.
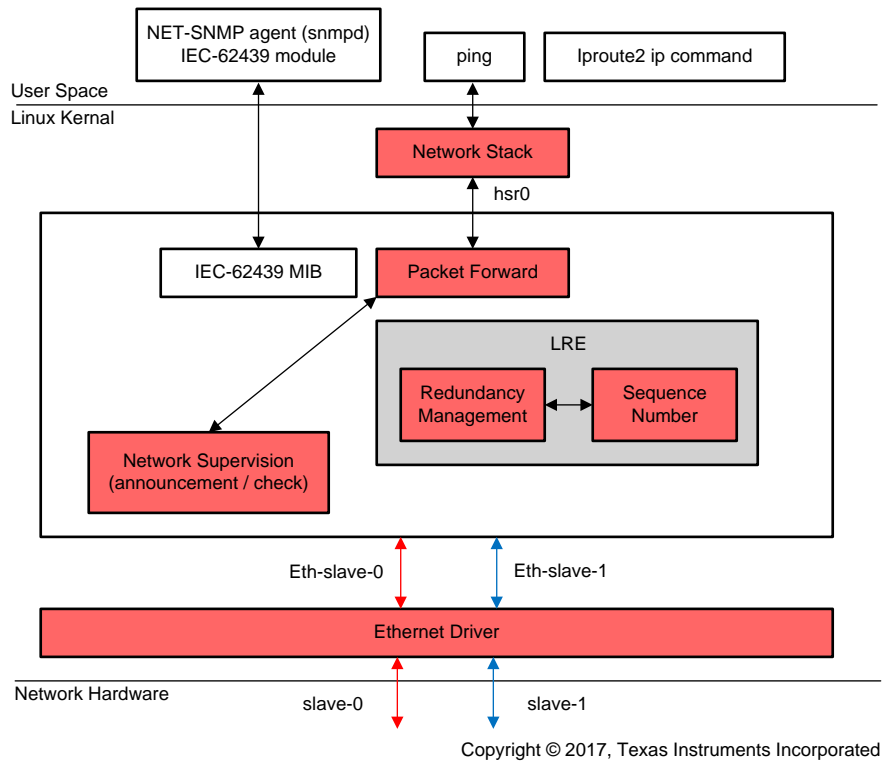
Copyright © 2017, Texas Instruments Incorporated

図 10. Linux® HSR Architecture

In this architecture, every packet must be processed by the HSR driver to see if it is meant for this particular node or if it needs to be forwarded to the next node in the ring. The time that this processing takes adds to the latency that a packet will experience traversing the ring. In order to minimize the trip delays experienced by each packet that is forwarded, this time is minimized.

図 11 shows a diagram of how the PRU-ICSS on the AM5728 device can be used to provide a fast, cut-through switch to minimize the delay added. The PRU-ICSS is a parallel core to the Cortex ARM-A15 that runs Linux. The PRU-ICSS is designed to minimize latency with deterministic processing. Instead of incurring the delay to pass every packet up to Linux, the PRU-ICSS can make a local decision as to whether or not to forward the packet. This dramatically lowers the delay experienced by each packet allowing it to traverse the ring with a minimized trip time. This capability may allow more flexible ring structures by adding more devices before the maximal delay is reached because of the size of the ring.

Copyright © 2017, Texas Instruments Incorporated
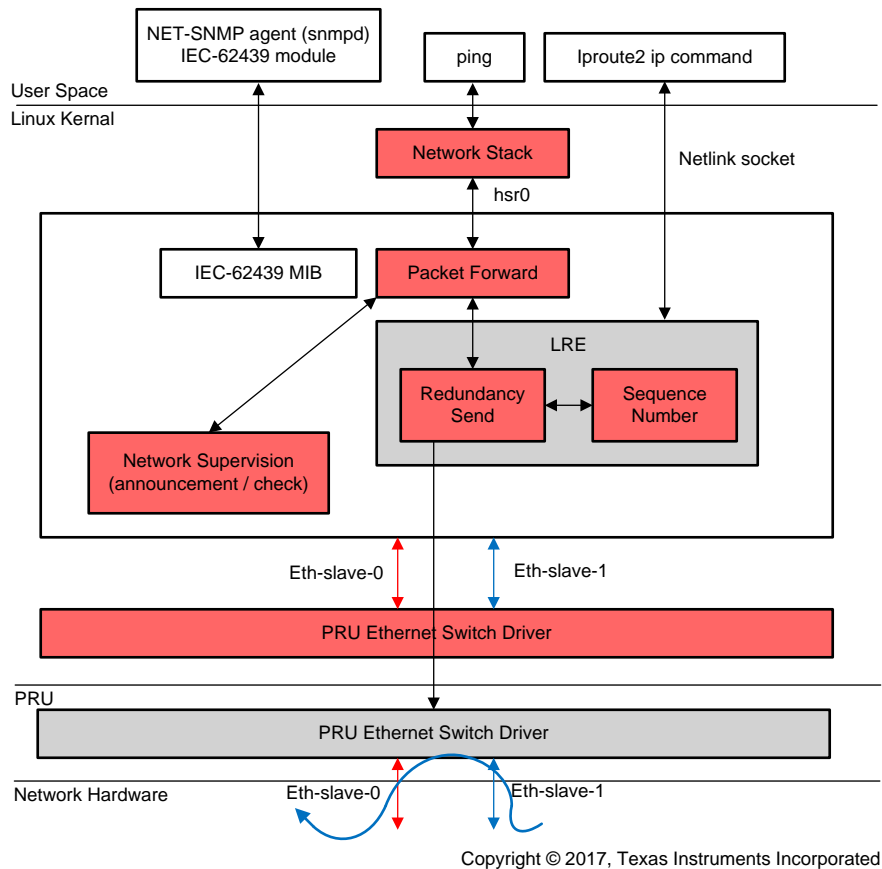
図 11. PRU-ICSS Enabled Fast Cut-Through Switch

3.1.2.2   Procedure

The Processor SDK includes everything required to test both the standard HSR implementation and the PRU-ICSS offload implementation. Both of these configurations abstract HSR to a standard Ethernet interface in Linux, so the upper-level applications do not require the details managed by the lower levels. Here is a list of the major decisions that must be made:

1. HSR must be built into the Linux kernel if it is not already. The SDK includes this step.
2. Choose whether to use standard HSR or PRU-ICSS Offload. This choice is made by passing parameters to the Linux kernel.
3. Configure the appropriate interfaces with the same Ethernet MAC addresses.
4. Bring both interfaces up.
5. Leverage the extended IP command to add a HSR interface using the two ports.
6. Assign the new HSR interface an IP address, and bring it up.
7. Use whatever network applications desire to use the HSR interface. A standard ping can easily test

connectivity and latency.

The details of this procedure are fully documented in the HSR documentation provided with the SDK.

3.2 Testing and Results

3.2.1 Test Setup

図 12 shows the test setup with three nodes. Each node has two Ethernet connections—each per adjacent node. For these experiments, the common Linux applications and commands were used to measure the performance of delivery ratio and latency. For the target TX and RX, a PC is attached to each node to configure test modes and parameters with a serial terminal program. In addition, the underlying firmware generates background traffic such as supervision frames to discover neighbors.
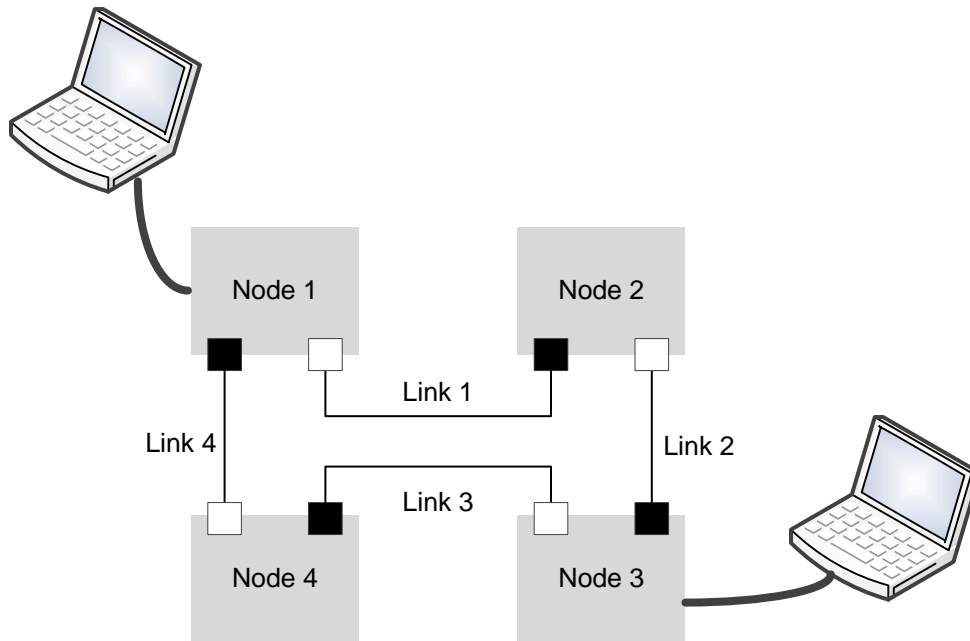
図 12. Test Setup (Three Nodes)

3.2.2 Test Results

The goal of these experiments is to evaluate that TI HSR/PTP solution meets the performance requirement for substation automation. 表 1 summarizes the performance requirement for substation automation. The required communication recovery time means the time duration in which a network recovers failure and the application recovery tolerated delay (or grace time) is the time duration in that the substation tolerates an outage of the automation system. The sampled values (SV) are sampled at a nominal value of 4 kHz. Therefore, the target application recovery tolerated delay for SV is 500 µs (= 2 × ¼ kHz).

表 1. Recovery Delay Demands as Shown in IEC 61850-5

COMMUNICATING PARTNERS	SERVICE	APPLICATION RECOVERY TOLERATED DELAY	REQUIRED COMMUNICATION RECOVERY TIME
SCADA to IED, client-server	IEC 61850-8-1	800 ms	400 ms
IED to IED interlocking	IEC 61850-8-1	12 ms (with T_{min} set to 4 ms)	4 ms
IED to IED, reverse blocking	IEC 61850-8-1	12 ms (with T_{min} set to 4 ms)	4 ms

表 **1. Recovery Delay Demands as Shown in IEC 61850-5 (continued)**

COMMUNICATING PARTNERS	SERVICE	APPLICATION RECOVERY TOLERATED DELAY	REQUIRED COMMUNICATION RECOVERY TIME
Protection trip excluding busbar protection	IEC 61850-8-1	8 ms	4 ms
Busbar protection	IEC 61850-9-2 on station bus	< 1 ms	Bumpless
Sampled values	IEC 61850-9-2 on process bus	Less than two consecutive samples	Bumpless

3.2.2.1   *Latency*

The goal of this experiment is to validate if the latency performance meets the requirement for substation automation applications. To measure the latency, one node sends a ping with packet sizes from 64 to 1500 bytes in increments of 200 bytes, and the target node will reply if it receives the request. The round-trip delay is measured at the originator of the packet by calculating the time gap between TX and RX. Then, the latency is calculated by a half of the round-trip delay. The latency measurement was performed 100 times, and the latency was averaged. The latency is considered as one-way delay based on the definition in the IEC/TR 61850-90-4.

To validate the impact of number of hops on the overall latency, the latency is measured over one-hop and two-hop network to compare the gap. In 図 13, to create two-hop network, Node 1 is configured as TX mode, and Node 3 is configured as echo-back RX mode. Similarly, to create one-hop network, Node 1 is configured as TX mode, and Node 2 is configured as echo-back RX mode. The latency is measured at Node 1.
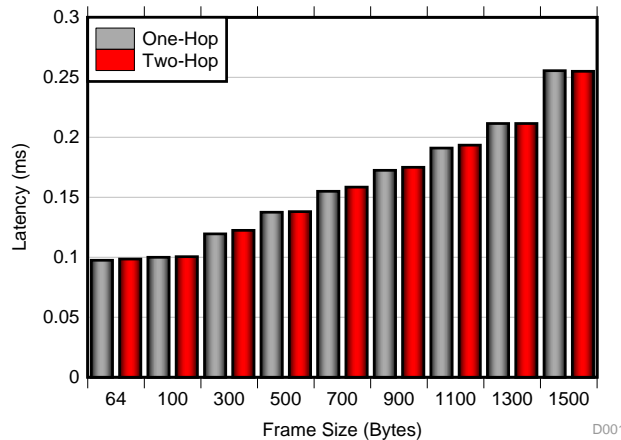

図 **13. Latency Performance**

図 13 shows latency performance as a function of frame size. The X-axis shows frame size in bytes, and the Y-axis shows latency in milliseconds. The result shows that latency performance with maximum frame size of 1500 bytes over two-hop meets the delay requirement (≤ 500 µs) for sampled values application. In addition, the result shows that the additional delay incurred by a single hop is very negligible due to the cut-through mechanism implemented in our HSR firmware. From this experiment, the worst case delay gap shows 3.5 µs.

3.2.2.2 Delivery Ratio

The goal of this experiment is to verify zero network recovery time that is one of requirements for substation automation. For this purpose, the delivery ratio performance was measured while emulating the link failures by disconnecting a link intentionally in the middle of data transmissions.

For this experiment, Node 1 is configured as TX mode with 10,000 packet transmissions, 1528-byte frame size, and 1-ms packet interval. The other nodes are configured as RX mode. During the experiment, Link 1 is disconnected to emulate link failure. To validate the impact of hops and packet types on delivery ratio performance, various experiments were performed with different hops and unicast or broadcast traffic. Each experiment emulated link failure by disconnecting Link 1.

Each experiment captured the number of TX packets at Node 1 and the number of RX packets at the other nodes. The delivery ratio is calculated by the number of TX packets divided by the number of RX packets.

表 2 shows delivery ratio performance over various scenarios. For all scenarios, the result shows a 100% delivery ratio even with link failure, which implies that link failure is recovered immediately. This is expected because redundant communication recovers the link failure immediately.

表 2. Delivery Ratio Performance

TEST SCENARIO	DELIVERY RATIO (%)
Unicast, 1-hop	100
Unicast, 2-hop	100
Broadcast, every node in the network	100

4 Design Files

4.1 Schematics

To download the schematics, see the design files at TIDEP-0096.

4.2 Bill of Materials

To download the bill of materials (BOM), see the design files at TIDEP-0096.

4.3 Layer Prints

To download the layer prints, see the design files at TIDEP-0096.

4.4 Assembly Drawings

To download the assembly drawings, see the design files at TIDEP-0096.

5 Software Files

To download the software files, see the design files at TIDEP-0096.

6 Related Documentation

1. Wikipedia, *High-availability Seamless Redundancy*
2. Texas Instruments, *AM572x Sitara™ Processors Silicon Revision 2.0*, AM5728 Datasheet (SPRS953)
3. Texas Instruments, AM437x Sitara Processors, AM437x Datasheet
4. Texas Instruments, AM335x Sitara Processors, AM335x Datasheet
5. Texas Instruments, *Processor SDK for Linux Getting Started Guide*, TI Software Documentation
6. Texas Instruments, *Processor SDK Linux HSR,PRP*, TI Software Documentation
7. University of Brescia, *IEC61850 One World, One Technology, One Standard* , PDF

7 About the Author

RON BIRKETT is an Applications Engineering Manager at Texas Instruments, where he is responsible for providing technical support and training on Linux running on Embedded Systems. He received his Master of Science in Computer Science from the University of Texas at Dallas, Dallas, TX.

改訂履歴

資料番号末尾の英字は改訂を表しています。その改訂履歴は英語版に準じています。