参考資料

TI Designs: TIDEP-0101
AM437x上のTamagawaエンコーダ・インターフェイス・プロトコルのリファレンス・デザイン

概要

Tamagawa SmartAbsフルアブソリュート・エンコーダまたはSmart Incエンコーダと、ドライブ・フィードバックおよび制御システムとを接続するため、一般に受け入れられている既存のソリューションは、Smartceiver AU5561N1を使用することです。これらのレシーバを使用するには、追加のコンポーネントが必要で、システムが複雑になり、コストも高くなります。このリファレンス・デザインでは、Sitara™AM437xプロセッサ(ARM® Cortex®-A9コアとPRU-ICSSの複数のインスタンスを搭載)を使用して互換性のあるレシーバを実装する方法について、詳しく説明します。このレシーバ実装は、PRU-ICSSのインスタンスの1つがTamagawaエンコーダを制御し、ARMデバイスおよび他のPRU-ICSSリソースがドライブ制御、電流検出、産業用通信を処理するため、追加のコンポーネントが必要ありません。この実装により、コスト効率が優れた、シングル・チップの産業用ドライブおよび通信ソリューションを実現できます。

リソース

TIDEP-0101	デザイン・フォルダ
PRU-ICSS-INDUSTRIAL-DRIVES	ソフトウェア・フォルダ
プロセッサSDK AM437x	ソフトウェア・フォルダ
AM4379	プロダクト・フォルダ
SN65HVD75	プロダクト・フォルダ
AM437x IDK	プロダクト・フォルダ
RS485半二重EVM	プロダクト・フォルダ

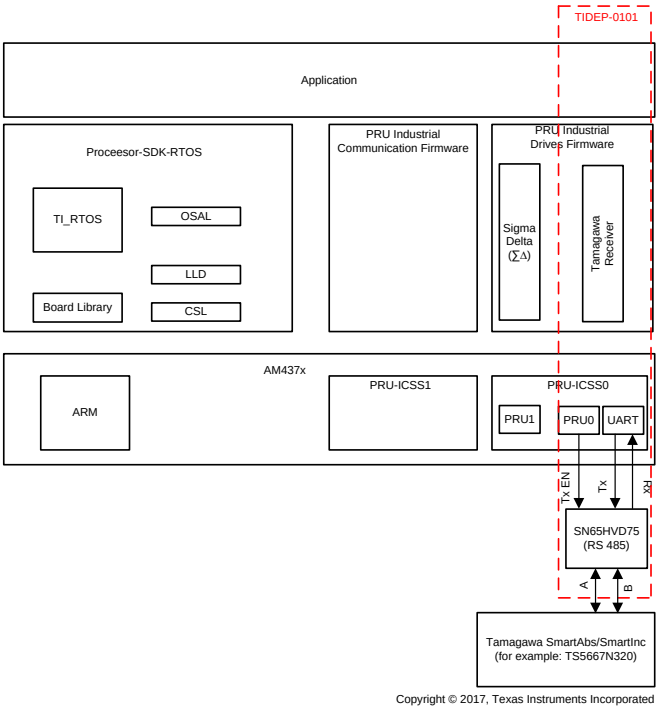
TI E2E™ Community　E2Eエキスパートに質問

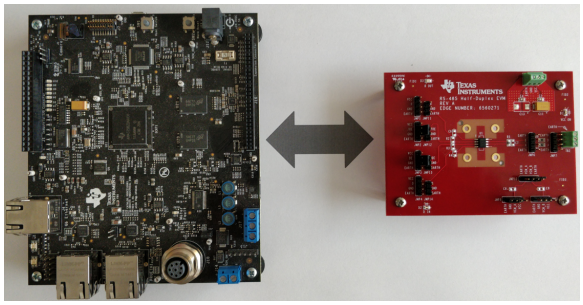
特長

- PRU-ICSSを使用するAM437x上のTamagawaレシーバ
- Tamagawa Smartceiver AU5561N1互換
- 2.5MbpsのSmartAbsフルアブソリュート・エンコーダおよびSmartIncエンコーダをサポート
- データ読み出しおよびリセット・フレームをサポート

アプリケーション

- サーボ・ドライブ位置フィードバック
- サーボ・ドライブ制御モジュール





1 System Description

This reference design implements the Tamagawa receiver on the Sitara AM437x processor using one instance of the two PRU-ICSS subsystems. The design is capable of communication with Tamagawa SmartAbs full-absolute and SmartInc encoders at 2.5 Mbps. The implementation is compatible with Tamagawa Smartceiver AU5561N1.

Industrial drives systems, especially for servo applications like machine tools, robotics, and so forth require position information with a high degree of precision. Position encoder technologies like EnDat, Tamagawa, Nikon, HIPERFACE DSL®, and BiSS C provide high-precision position information in digital format which is then provided to the servo control processor. Tamagawa is one of the most widely used of these position encoder technologies.

The commonly-accepted existing solution to interface Tamagawa SmartAbs full-absolute or Smart Inc encoders with a drive feedback and control system typically uses the Smartceiver AU5561N1. Use of these receivers requires additional components, complexity, and costs to the system. This reference design details a compatible receiver implementation using the Sitara™ AM437x processor, which has multiple instances of PRU-ICSS. This receiver implementation bypasses the requirement for additional components because it requires only one instance of the two PRU cores in one of the two PRU-ICSS subsystems on the AM437x, while the ARM device and remaining PRU-ICSS resources can handle the drive control, current sense, and industrial communications. This implementation allows for a cost-effective, single-chip, industrial drive and communication solution.

On a higher level, the AM437x processor is well suited for a single-chip drive design. The device has demodulators for sigma delta (ΣΔ) as well as the option of using an on-chip successive approximation register (SAR) analog-to-digital converter (ADC). The AM437x also has six pulse width modulation (PWM) instances to drive two three-phase motors. One of the PRU-ICSS subsystems can be used for industrial Ethernet connectivity such as EtherCAT. The powerful (1-GHz maximum) ARM® Cortex®-A9 series processor provides an excellent platform for the control algorithm, high-level industrial communication stack operations, and user applications.

1.1 Key System Specifications

表 1. Key System Specifications

PARAMETER	SPECIFICATION
Modules	AM437x IDK and RS485 half-duplex EVM (connected together through wires)

2 System Overview

The AM437x Tamagawa receiver implementation is a combination of hardware and software. The AM437x system-on-chip (SoC) and RS485 transceiver are the core hardware components and the PRU-ICSS-INDUSTRIAL-DRIVES package contains the necessary software. The AM437x Industrial Development Kit (IDK) and the RS485 half-duplex evaluation module, which contains the RS485 - SN65HVD75DR, are the boards used for this implementation. The PRU firmware, which is available in PRU-ICSS-INDUSTRIAL-DRIVES, is the core software component that exposes an interface for the application, which executes on the ARM Cortex-A9 processor. The PRU-ICSS-INDUSTRIAL-DRIVES package includes a diagnostic application to validate the Tamagawa receiver implementation.

The PRU-ICSS-INDUSTRIAL-DRIVES package is specified for use on top of the Processor-SDK-RTOS, which provides the TI-RTOS for the AM437x processor. Industrial applications, such as drive control, are typically built on top of a real-time operating system (RTOS) that runs on an ARM Cortex-A9 processor. The Processor-SDK-RTOS also provides drivers for various peripherals, such as PWM, I^2C, serial peripheral interface (SPI), and so forth, as well as an operating system abstraction layer (OSAL) that the application can use. The applications in the PRU-ICSS-INDUSTRIAL-DRIVES package use these features.

The Tamagawa receiver implementation makes use of the PRU-ICSS0 UART in the AM437x processor. The firmware, which is written in C, runs on the PRU0 of PRU-ICSS0. The UART transmit and receive interfaces are connected to the RS485 transceiver (SN65HVD75DR), which is located on a separate RS485 half-duplex evaluation module (EVM). The RS485 transmit transceiver enable is derived from the PRU0 GPO because the receive transceiver is always enabled.

The AM437x IDK 3.3-V supply feeds the power for the RS485 half-duplex EVM and also provides 5 V for the encoder. Both 5 V and 3.3 V are available on the AM437x IDK through the headers.

During start-up, the application that runs on the ARM Cortex-A9 processor initializes the module clocks and configures the pinmux. The PRU is initialized and the PRU firmware is loaded on PRU0. After the PRU0 execution starts, the Tamagawa interface is operational and the application can use it to communicate with the encoder. Use the Tamagawa diagnostic example in the PRU-ICSS-INDUSTRIAL-DRIVES package to learn more about initialization and communication with the Tamagawa interface.

2.1 Block Diagram

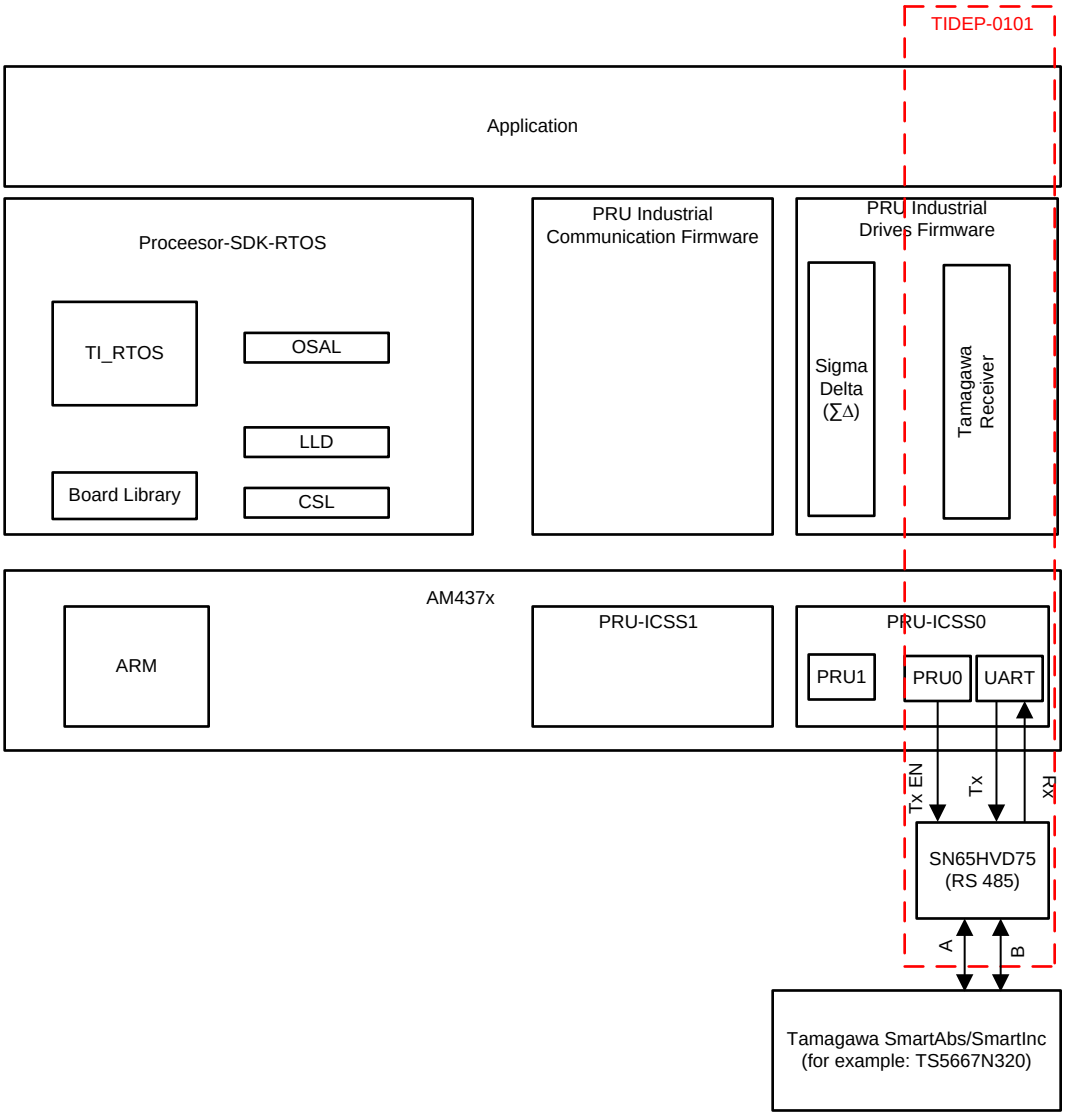


図 1. TIDEP-0101 Block Diagram

2.2 Tamagawa

Tamagawa is an encoder technology used for obtaining high-precision position information in machine tools, robotics, and so forth. Tamagawa rotary encoders consist broadly of two types: incremental or absolute. Incremental encoders provide a train of pulses, while the absolute-type provides digital values. The absolute encoder group contains the single-turn types that provide outputs which can be open collector or emitter follower. The absolute encoder types include the pure digital encoder types, which provide a digital word output through a line driver such as an RS485, or a semi-absolute encoder, which provides both digital word and pulse train outputs. Of the RS485 line-driver output absolute encoders that provide only digital output, another classification is the full absolute encoder. A full absolute encoder provides multi-turn digital data, which is known as SmartAbs, and is compatible with the Tamagawa Smartceiver AU5561N1. Another type of encoders, known as SmartInc, provide single-turn information in digital format with an RS485 line driver output. The AM437x Tamagawa receiver implementation is equivalent to the Smartceiver AU5561N1, which can communicate with Tamagawa SmartAbs as well as SmartInc encoders.

The AM437x Tamagawa receiver communicates with Tamagawa SmartAbs and SmartInc encoders and provides drive control with digital information to and from the encoder. Tamagawa communication is broadly classified into three types: data readout, reset, and EEPROM transactions. Four data readout transactions occur: absolute data in one revolution, multi-turn data, encoder ID, and a combination of all of these along with the encoder error status. The reset transaction always returns the absolute data in one revolution while performing different types of resets. Three types of reset are available: reset of absolute data in one revolution, reset of multi-turn data, and error reset. The EEPROM transaction allows the system to read and write to the EEPROM in the encoder.

Each transaction has a unique data ID and consists of different fields, namely control, status, data, cyclic redundancy check (CRC), EEPROM address, and EEPROM data depending on the type of transaction, that is, data ID.

Each field is 10-bits long, beginning with a start bit and ending with a delimiter. The 8 bits between these start bits and delimiters depends on the field type. The control field contains the data ID information. Data, status, and CRC fields similarly contain data, status, and CRC in those 8 bits.

The receiver initially sends the control field to start the communication. This action indicates the type of transaction to the encoder and the encoder returns this information based on the data ID, as the previous paragraph explains. The encoder always returns the control field back to the receiver.

In the case of data readout and reset transactions, the encoder returns the control field followed by the status, data, and ending with the CRC field at the end.

In the case of an EEPROM read or write, the receiver, in addition to the control field, sends the EEPROM address field (and EEPROM data field for write) followed by the CRC. The encoder returns the control field, followed by the EEPROM address, EEPROM data, and CRC fields.

The physical layer communication is RS422/RS485 based.

2.3 Highlighted Products

2.3.1 AM437x

The TI 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 coprocessors for deterministic, real-time processing including industrial communication protocols, such as EtherCAT®, PROFIBUS®, and others as well as industrial drive protocols such as EnDat, Tamagawa, Sigma Delta, and so forth. The devices support operating systems like 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 2 shows the AM437x block diagram.

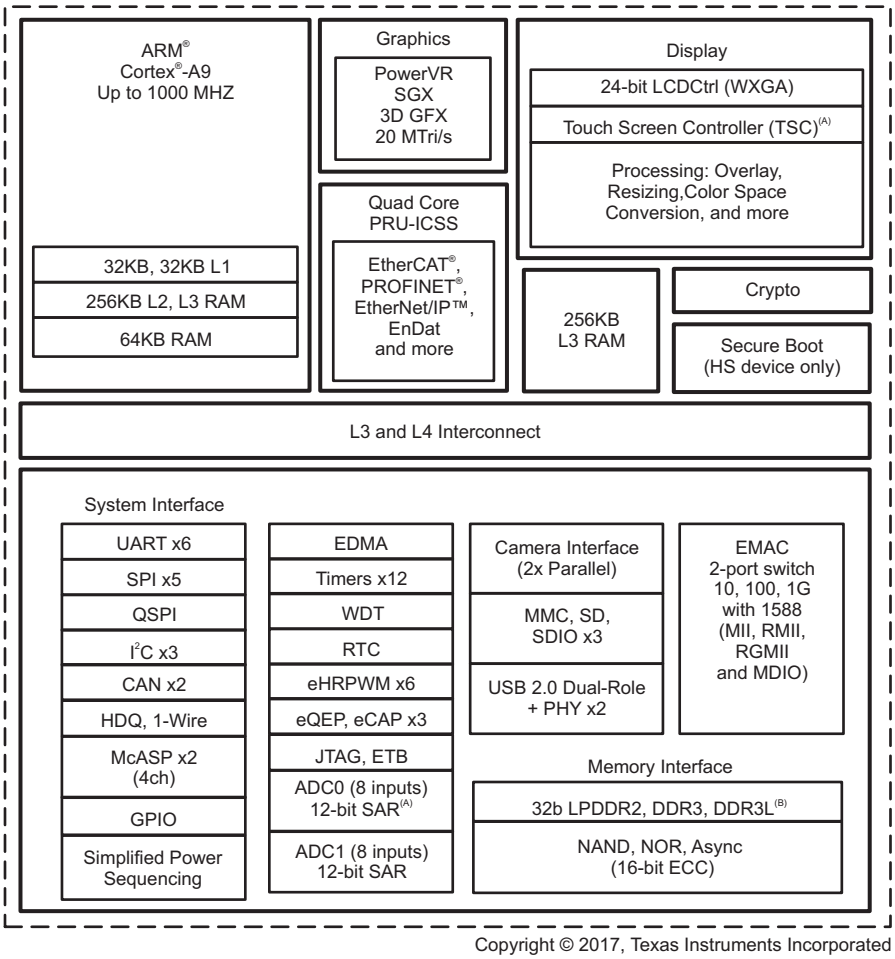


図 2. AM437x Block Diagram

2.3.1.1 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 ™, 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 3 shows the PRU-ICSS block diagram.

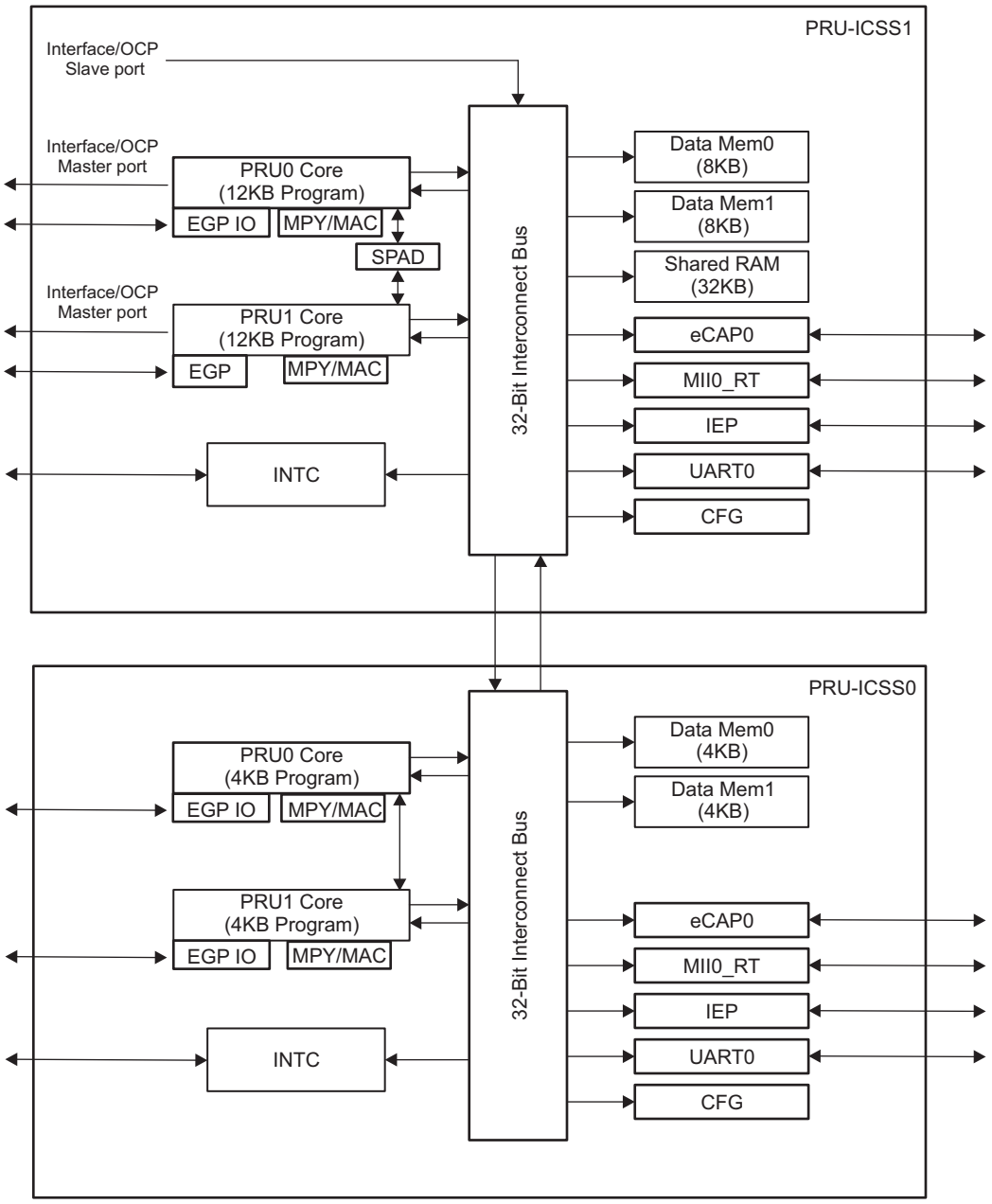


図 3. PRU-ICSS Block Diagram

2.3.2 SN65HVD75 (RS485)

These devices have robust 3.3-V drivers and receivers in a small package for demanding industrial applications. The bus pins are robust to electrostatic discharge (ESD) events with high levels of protection to human-body model (HBM) and IEC contact discharge specifications.

Each of these devices combines a differential driver and a differential receiver, which operate from a single 3.3-V power supply. The driver differential outputs and the receiver differential inputs are connected internally to form a bus port suitable for half-duplex (two-wire bus) communication. These devices feature a wide common-mode voltage range making the devices suitable for multi-point applications over long cable runs. These devices are characterized from –40°C to 125°C.

2.4 Software Description

At start-up, the application running on the ARM Cortex-A9 initializes the module clocks and configures the pinmux. The PRU is initialized and the PRU firmware is loaded on PRU0 of PRU-ICSS0. After the PRU0 starts executing, the Tamagawa interface is operational and the application can use it to communicate with an encoder. Use the Tamagawa diagnostic example in the PRU-ICSS-INDUSTRIAL-DRIVES package to learn more about initialization and communication with the Tamagawa interface.

This Tamagawa diagnostic example in the PRU-ICSS-INDUSTRIAL-DRIVES package (available at the path "examples/tamagawa_diagnostic" in the installed directory), also provides an easy way to validate the Tamagawa transactions. The diagnostic example provides menu options on the host PC in a serial terminal application (like TeraTerm), where the user can select the data ID code to be sent. Based on the data ID code, the application updates the Tamagwa interface with the data ID code and trigger transaction. The application then waits until it receives an indication of complete transaction by the firmware through the interface before displaying the result. See *Industrial Drives User Guide* for more details.

2.4.1 PRU Firmware Design

2.4.1.1 Overview

The PRU firmware is written in C. The firmware sources, as well as the binary, are available in the "interfaces/tamagawa_receiver/firmware" path in the installed directory of the PRU-ICSS-INDUSTRIAL-DRIVES package.

The firmware first initializes the PRU hardware, after which it waits until a command has been triggered through the interface. Upon triggering, the transmit data is set up based on the data ID code and the data is transmitted. The data ID code then waits until receiving all the data that depends on the data ID. The parsing over the received data then commences, which is again based on the data ID, and the interface is updated with the result. The CRC verification occurs next and the interface indicates command completion. The firmware then waits for the next command trigger from the interface.

図 4 shows the PRU firmware overview.

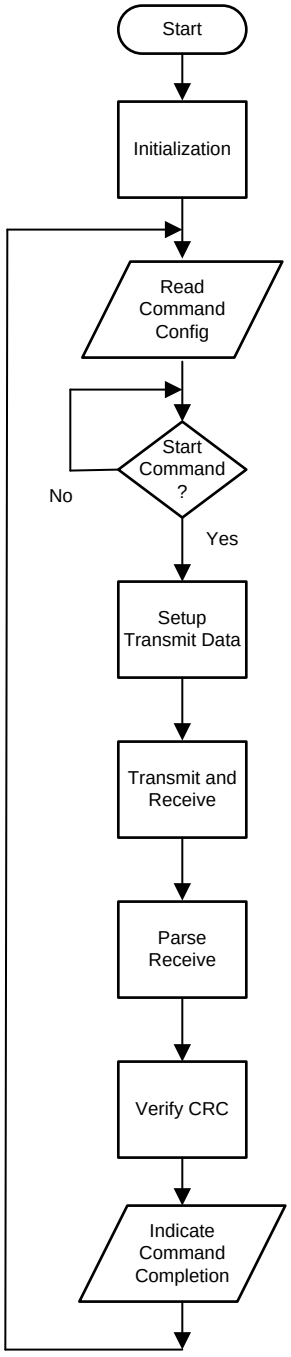


図 4. Overview Flow Chart

注: The firmware is an enhancement over the PRU hardware UART example, which is available as part of the PRU software support package, to manage Tamagawa-specific handling.

2.4.1.2 Initialization

The ARM application (Tamagawa diagnostic from PRU-ICSS-INDUSTRIAL-DRIVES package) performs SoC-specific initializations like pinmux, module clock enabling , and PRU-ICSS initialization before executing the PRU firmware. After enabling the open core protocol (OCP) master ports, the transmit is disabled (a GPO is used to enable and disable transmit). Set up the UART for a 2.5-Mbps baud rate by setting the divisors appropriately. First-in-first-out (FIFO) control is enabled and the transmit and receive FIFO are cleared as well. The next step is setting the protocol for 8-bit data, 1 stop bit, and no parity.

図 5 shows the initialization flow chart.

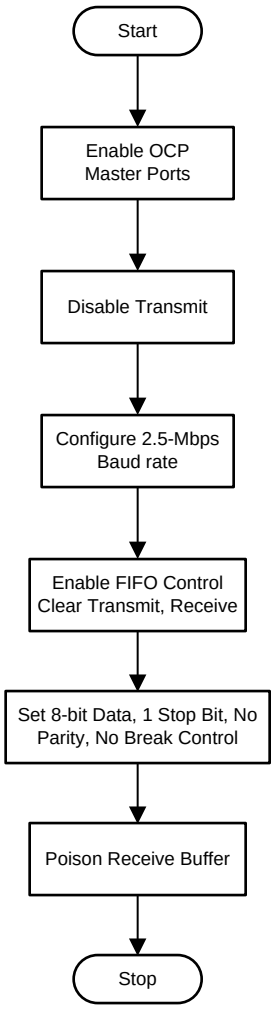


図 5. Initialization Flow Chart

2.4.1.3 Setup Transmit Data

The transmit and receive sizes are determined based on the data ID in the interface. Then copy the transmit data from the interface to the local buffer.

図 6 shows the flow chart for the setup transmit data.

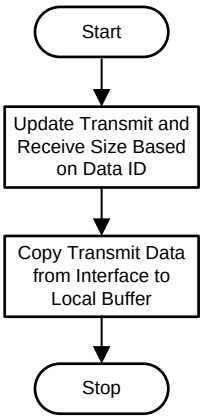


図 6. Setup Transmit Data Flow Chart

2.4.1.4 Transmit and Receive

The GPO initially enables the transmit. Write one byte at the beginning of the buffer from the local transmit buffer that has the data to be transmitted. The firmware then waits until the transmitted data returns (note that the receive is always enabled, so transmitted data always reflects back). The firmware then copies the reflected data to the receive buffer from the receive FIFO and continues until all of the data has been transmitted, after which it disables transmit.

At this point, the encoder starts sending the data and the firmware copies the receive FIFO contents onto the receive buffer, individually, until all the data has been received.

図 7 shows the transmit and receive flow chart.

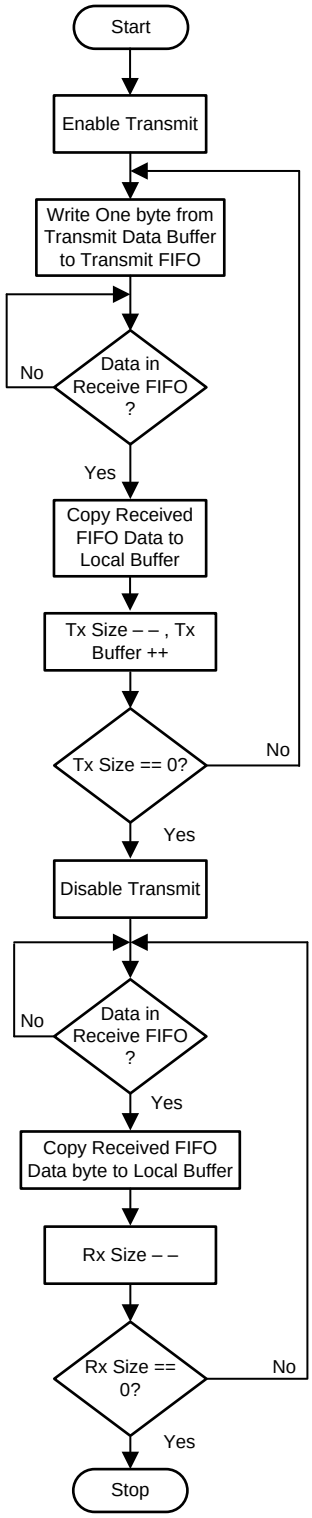


図 7. Transmit and Receive Flow Chart

2.4.1.5 Receive Data Parse

Depending on the data ID used for initiating the transfer, the firmware parses the received data and copies it onto relevant fields in the interface, accordingly.

図 8 shows the receive data parse flow chart

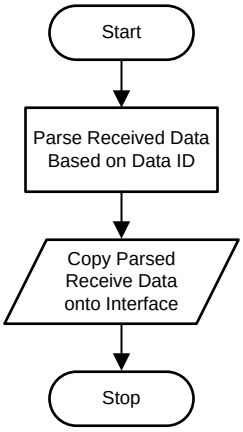


図 8. Receive Data Parse Flow Chart

2.4.1.6 Verify CRC

The CRC is the last byte of the received data. The firmware then calculates the CRC of the received data excluding the last byte, compares it with the received CRC value, and updates the CRC status in the interface.

図 9 shows the verify CRC flow chart.

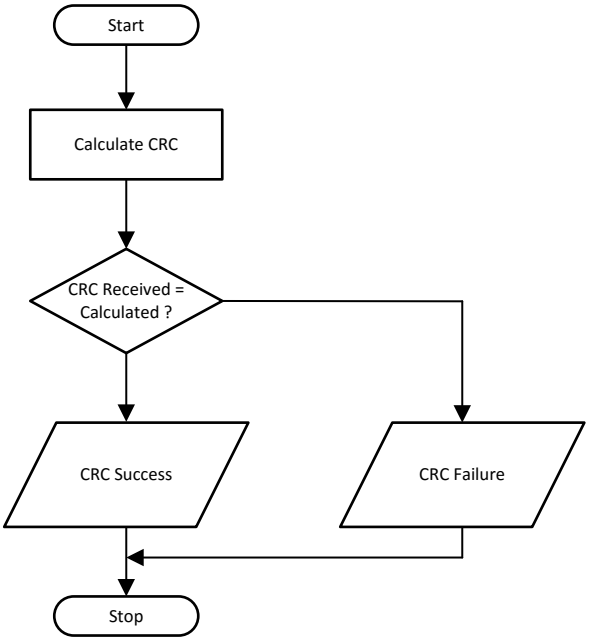


図 9. Verify CRC Flow Chart

2.4.2 PRU Firmware Interface

The firmware exposes an interface as follows. This firmware interface is defined in the file "tamagawa_interface.h" available at the path "interfaces/tamagawa_receiver/include" in the installed directory of the PRU-ICSS-INDUSTRIAL-DRIVES package. An application can use this firmware interface to communicate with the Tamagawa encoder. Note that the Tamagawa diagnostic uses this interface:

```
struct TamagawaInterface
{
    uint32_t version;  /**< Firmware version */
    uint8_t  config;   /**< Configuration */
    uint8_t  data_id;  /**< Data ID code */
    struct tx {        /**< Transmit data */
    uint8_t  adf;          /**< EEPROM address */
    uint8_t  edf;          /**< EEPROM data */
    } ;
    struct rx {        /**< Received data */
        uint32_t abs;      /**< Data in one revolution */
        uint32_t abm;      /**< Multi-turn Data */
        uint8_t  cf;       /**< Control Frame */
        uint8_t  sf;       /**< Status Frame */
        uint8_t  enid;     /**< Encoder ID */
        uint8_t  almc;     /**< Encoder error */
        uint8_t  adf;      /**< EEPROM address */
        uint8_t  edf;      /**< EEPROM data */
        uint8_t  crc;      /**< CRC */
    };
};
```

The possible data ID codes in the "data_id" field of the interface structure is as follows:

```
enum data_id
{
    DATA_ID_0,  /**< Data readout data in one revolution */
    DATA_ID_1,  /**< Data readout multi-turn data */
    DATA_ID_2,  /**< Data readout encoder ID */
    DATA_ID_3,  /**< Data readout data in one revolution, encoder ID, multi-turn, encoder error */
    DATA_ID_6,  /**< EEPROM write */
    DATA_ID_7,  /**< Reset error */
    DATA_ID_8,  /**< Reset one revolution data */
    DATA_ID_C,  /**< Reset multi-turn data */
    DATA_ID_D,  /**< EEPROM read */
    DATA_ID_NUM /**< Number of Data ID codes */
};
```

The configuration macros for the “config” field of the interface structure is as follows:

CONFIG_CMD_PROCESS: Dual purpose, start Tamagawa transaction as well as indicate transaction completion

CONFIG_CMD_STATUS: Tamagawa transaction status (valid after CONFIG_CMD_PROCESS indicates transaction completion)

2.4.2.1 Usage

Use of the Tamagawa interface from an application is as follows:

1. Enter the data ID code in the "data_id" field of the interface structure.
2. Enter the EEPROM address (for EEPROM read and write) and EEPROM data for the EEPROM write in the "struct tx" field of the interface structure.
3. Apply the logical OR of the CONFIG_CMD_PROCESS onto the configuration field.
4. Wait until the logical AND of the CONFIG_CMD_PROCESS on the configuration field is “0".
5. Read the received data from the "struct rx" field of the interface structure (see 表 2).

注: The firmware does not currently handle EEPROM commands, which TI plans to fix in a later release of the PRU-ICSS-INDUSTRIAL-DRIVES package.

表 2. Valid Received Data ("struct rx" Fields) for Data ID Codes

DATA ID NUMBER	struct rx FIELDS								
	abs[1]	abm[1]	cf	sf	enid	almc	adf	edf	crc
DATA_ID_0	X		X	X					X
DATA_ID_1		X	X	X					X
DATA_ID_2			X	X	X				X
DATA_ID_3	X	X	X	X	X	X			X
DATA_ID_6			X				X	X	X
DATA_ID_7	X		X	X					X
DATA_ID_8	X		X	X					X
DATA_ID_C	X		X	X					X
DATA_ID_D			X				X	X	X

(1) Though abs and abm are 32 bits, only the least significant 24 bits are valid, the most significant byte (MSB) is always zero. If the resolution of abs and abm for an encoder is less than 24, the valid bits will be less than 24; in that case, all leading bits would be zero.

3 Hardware, Software, Testing Requirements, and Test Results

3.1 Required Hardware and Software

3.1.1 Hardware

3.1.1.1 AM437x IDK

The AM437x Industrial Development Kit (IDK) is an application development platform for evaluating the industrial communication and control capabilities of Sitara™ AM4379 and AM4377 processors for industrial applications.

3.1.1.2 Half-Duplex RS485 EVM

The RS485 Half-Duplex EVM helps designers evaluate device performance, supporting fast development and analysis of data transmission systems using any of the SN65HVD1X, SN65HVD2X, SN65HVD7X, SN65HVD8X, and SN65HVD96 transceivers. The EVM board comes without the RS-485 device soldered to the board.

SN65HVD75 is RS-485 soldered to the board, this is available as sample too.

3.1.1.2.1 SN65HVD75 (RS485)

SN65HVD75 devices have robust 3.3-V drivers and receivers in a small package for demanding industrial applications. The bus pins are robust to ESD events with high levels of protection to human-body model and IEC contact discharge specifications.

3.1.1.3 SmartAbs TS5667N320

The TS5667N320 is the Tamagawa SmartAbs encoder used for validation. TS5667N320 is a full absolute encoder that has a total resolution of 33 bits with 17 bits per revolution resolution and a multi-turn counting of 16 bits.

The AM437x PRU-ICSS UART transmit and receive signals available on the AM437x IDK header is connected to the RS485 transceiver (SN65HVD75DR), which is located on the RS485 half-duplex EVM. The RS485 transmit transceiver enable is derived from the PRU0 GPO, which is also available on the AM437x IDK header. The receive transceiver is always enabled. The available 3.3 V on the AM437x IDK supplies the power for the RS485 half-duplex EVM. 表 3 further details the device connections.

表 3. Connections

SIGNAL	DEVICE		
	AM437x IDK	HALF-DUPLEX RS485 EVM(1)	TS5667N320
Receive (UART Rx)	J16-28	JMP1-R	—
Transmit (UART Tx)	J16-30	JMP4-D	—
Transmit enable (GPO-R30)	J16-32	JMP3-DE	—
3.3V	J16-1	TB1-VCC	—
GND	J16-59	TB1-GND	—
RS485 bus A	—	TB2-A	Serial data
RS485 bus B	—	TB2-B	Serial data inverted
5V	J16-2	—	VCC
GND	J16-60	—	GND

(1) On RS485 half duplex EVM, short JMP2-/RE & JMP2-GND as well as GND & EARTH on TB1.

注: The Tamagawa specification mentions usage of termination resistors, 220 Ω between bus terminals A and B, as well as 1 kΩ between bus terminals A, 5V and B, GND. These termination resistors were not used for this reference design because the cable length between the encoder and receiver was very short. However, TI recommends to use termination resistors as indicated in the specification for production use.

A battery supply of 3.6 V was not connected to the encoder; if multi-turn information must be retained over power interruption, then a battery supply must be provided.

3.1.2 Software

The PRU-ICSS-INDUSTRIAL-DRIVES page contains the necessary software. Pre-built binaries are also available. The Tamagawa diagnostic is the relevant application. Users can also build this application from sources; see *Industrial Drives User Guide* for more details.

3.2 Testing and Results

3.2.1 Test Setup

The previous 表 3 details the connections between the AM437x IDK and half-duplex RS485 EVM populated with an SN65HVD75 and SmartAbs TS5667N320.

図 10 shows a photo of the TIDEP-0101 test setup.

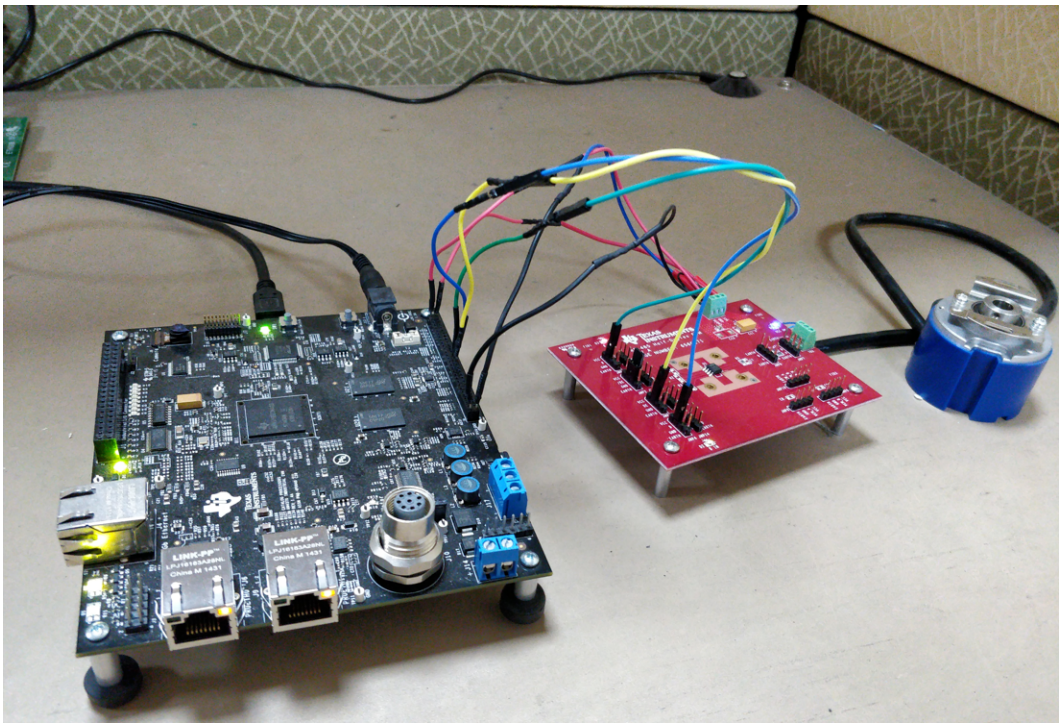


図 10. TIDEP-0101 Test Setup

The user must execute the Tamagawa diagnostic application referred to in 3.1.2 on the AM437x IDK. Use a pre-built application to get started quickly; to build the application from sources, see *Industrial Drives User Guide* . Copy the Tamagawa diagnostic application named "app" from the pre-built binaries to the uSD card. Copy the required bootloader to the uSD card, as well. This component is not part of the PRU-ICSS-INDUSTRIAL-DRIVES package but is located in Processor-SDK-RTOS, which can be found here for AM437x. After installing the Processor-SDK-RTOS package, navigate to the PDK component folder using the following path "packages\ti\starterware\binary\bootloader\bin\am43xx-evm\gcc". Locate the file named "bootloader_boot_mmcsd_a9host_release_ti.bin" and copy it over to the file named "MLO" in the uSD card. Now insert the uSD card into the AM437x IDK uSD card slot.

The application is controlled with a terminal interface using a serial-over-USB connection between the PC host and the EVM. Connect a USB cable between the PC and the EVM. Then use a serial terminal application (such as TeraTerm hyperterminal or Minicom) to run on the host (see 図 11). To configure, select the serial port corresponding to the port emulated over USB by the EVM. The host serial port should be configured to 115200 baud, no parity, 1 stop bit, and no flow control. The application starts executing upon power-on or reset of the AM437x IDK.

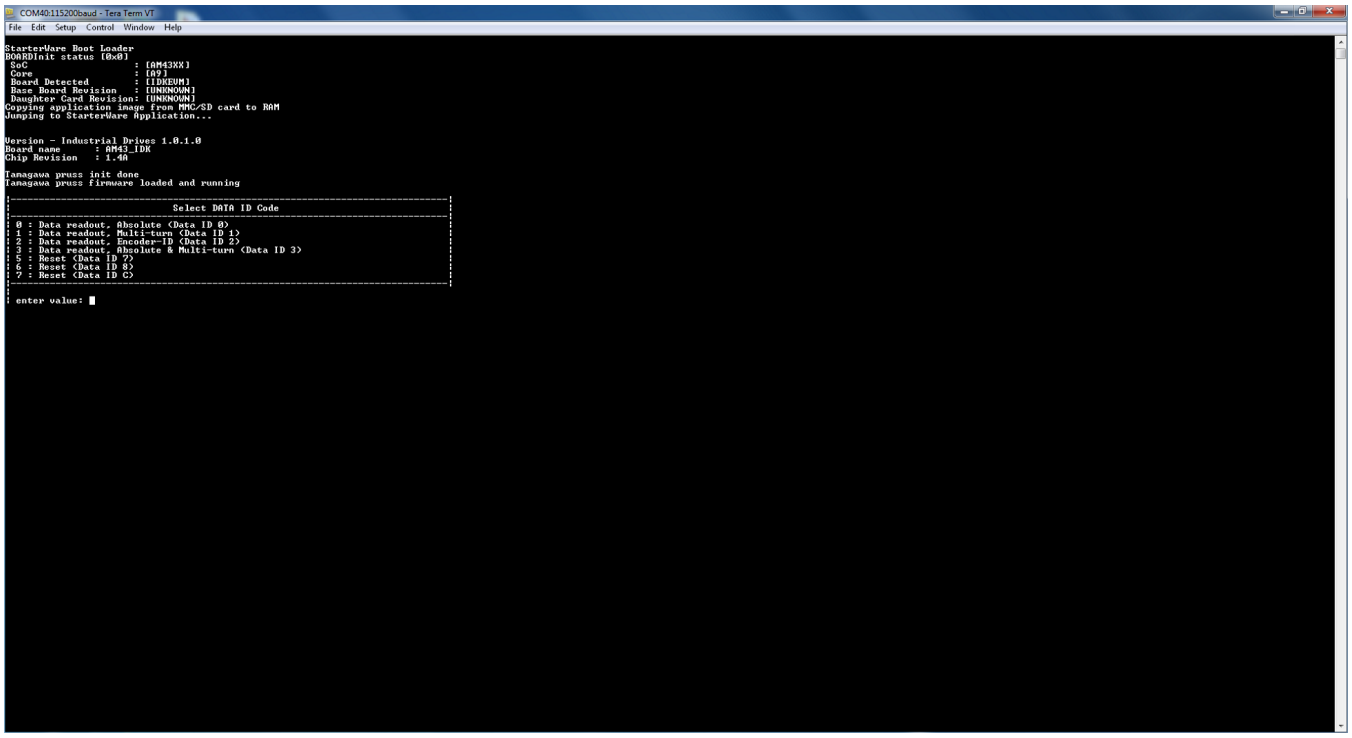


図 11. Start-Up Serial Terminal Display

The application provides a menu option on the serial terminal to select the data ID. Upon selecting the data ID, the Tamagawa transaction starts. After completion, the application terminals shows the transaction result depending on the selected data ID (see 図 12).

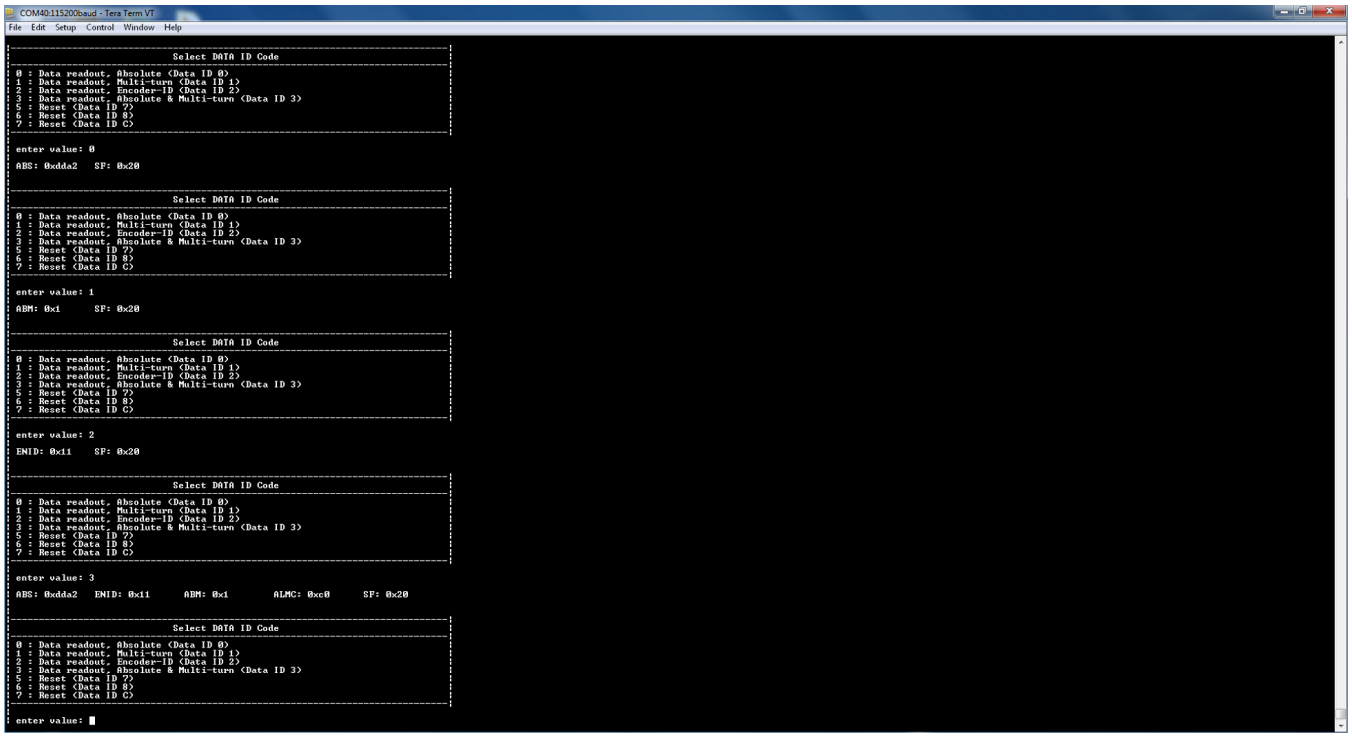


図 12. Data Readout Serial Terminal Display

3.2.2 Test Results

See 表 4 for the results and see 表 5 for the test details

表 4. Test Results

TEST CASE NO	DATA ID	DESCRIPTION	RESULT
1	Data ID 0	Data readout data in one revolution	PASS
2	Data ID 1	Data readout multi-turn data	PASS
3	Data ID 2	Data readout encoder ID	PASS
4	Data ID 3	Data readout data in one revolution, encoder ID, multi-turn, and encoder error	PASS
5	Data ID 8	Reset one revolution data	PASS
6	Data ID C	Reset multi-turn data	PASS

注: EEPROM transactions were not executed in testing as they do not work in this application.

表 5. Test Case Description

TEST CASE NO	TEST DESCRIPTION
1	After application starts execution, execute test case 5, rotate rotary part of the encoder 180°, select "DATA ID 0", verify that ABS reports the value 0x10000 ± 5. Also verify that serial terminal application does not report CRC failure.
2	After application starts execution, execute test case 6, followed by test case 5, rotate rotary part of the encoder slightly more than one rotation in either direction from the initial position. Select "DATA ID 1", verify that ABM reports the value 1 in one direction, while 0xfffe in other direction. Also verify that serial terminal application does not report CRC failure.
3	After application starts execution, select "DATA ID 2", verify that encoder ID is reported as 0x11. Also verify that serial terminal application does not report CRC failure.
4	After application starts execution, select "DATA ID 3", verify that encoder ID is reported as 0x11, ABS is reported as same value as with "DATA ID 0", ABM reported as same value as "DATA ID 1". Also verify that serial terminal application does not report CRC failure.
5	After application starts execution, select "DATA ID 0" in serial terminal application, ensure that ABS is non-zero, if it is zero, slightly rotate the rotary part of encoder to make sure it is non-zero. Keep rotary part of the encoder stationary, select "DATA ID 8" ten times through serial terminal menu one after the other, then select "DATA ID 0", verify that ABS is zero. Also verify that serial terminal application does not report CRC failure for any of the commands.
6	After application starts execution, select "DATA ID 1" in serial terminal application, ensure that ABM is non-zero, if it is zero, rotate the rotary part of encoder to make sure it reports non-zero. Keep rotary part of the encoder stationary, select "DATA ID C" ten times through serial terminal menu one after the other, then select "DATA ID 1", verify that ABM is zero. Also verify that serial terminal application does not report CRC failure for any of the commands.

4 Software Files

To download the software files, refer to the PRU-ICSS-INDUSTRIAL-DRIVES package.

5 Terminology

ADC— Analog-to-digital converter

CRC — Cyclic redundancy check

EEPROM— Electrically erasable programmable read-only memory

ESD— Electrostatic discharge

GUI— Graphical user interface

MCU— Microcontroller

OSAL— Operating system abstraction layer

PRU-ICSS— Programmable Real-Time Unit Subsystem and Industrial Communication SubSystem

PWM— Pulse width modulation

RTOS— Real-time operating system

SoC— System-on-chip

SPI— Serial peripheral interface

UART— Universal asynchronous receiver/transmitter

6 About the Author

AFZAL MOHAMMED is a system software engineer associated with Texas Instruments where he is responsible for Industrial Drive solutions in Catalog Processors BU.

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。