

# TI Designs

## SPI Master With Signal Path Delay Compensation on PRU-ICSS



### Design Overview

The Programmable Real-Time Unit and the Industrial Communication Subsystem (PRU-ICSS) enables customers to support real-time critical applications without using field-programmable gate arrays (FPGAs), complex programmable logic devices (CPLDs), or application-specific integrated circuits (ASICs). This TI design describes the implementation of the SPI master protocol with signal path delay compensation on PRU-ICSS. The design supports the 32-bit communication protocol of the ADS8688 device with an SPI clock frequency of up to 16.7 MHz.

### Design Resources

<a href="#">TIDEP0033</a>	Tool Folder Containing Design Files
<a href="#">TIDA-00164</a>	Tool Folder
<a href="#">TMDXIDK437X</a>	Tool Folder
<a href="#">AM4379</a>	Product Folder
<a href="#">ADS8688</a>	Product Folder
<a href="#">ISO7141CC</a>	Product Folder

### Design Features

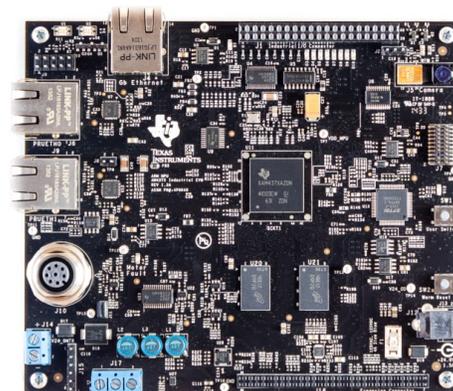
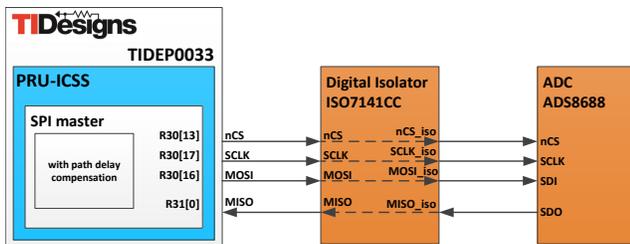
- Serial Peripheral Interface (SPI) Master Protocol With Adjustable Signal Path Delay Compensation (Does Not Require External Hardware for Signal Path Delay Compensation)
- Up to 16.7-MHz SPI Clock
- Supports ADS8688 SPI Communication Protocol
- Automatic Measurement of Signal Path Delay for Known Slave Response
- PRU-ICSS Firmware Validated With TIDA-00164 (ADS8688 and ISO7141CC) and Contains Firmware Source Code, Implementation Description, and Getting Started Instructions

### Featured Applications

- Industrial Drives
- Industrial Sensors and I/O Modules
- Industrial Communication Gateways
- Programmable Logic Control Systems (PLC)



[ASK Our E2E Experts](#)  
[WEBENCH® Calculator Tools](#)



An IMPORTANT NOTICE at the end of this TI reference design addresses authorized use, intellectual property matters and other important disclaimers and information.

## 1 Key System Specifications

FUNCTION	DESCRIPTION
SPI protocol	Master
Supported signals	MISO, MOSI, CLK, and nCS
SPI clock	1 MHz, 5 MHz, 10 MHz, and 16.7 MHz
Communication protocol	32-bit data protocol: 16-bit master to slave and 16-bit slave to master
Adjustable signal path delay compensation	Up to 117 ns of delay path
Additional features	Measurement of signal path delay for known slave answer

## 2 System Description

Many industrial automation solutions require isolating the communication paths between the application processor on the cold side and the I/O module on the hot side. Adding isolator components in the communication path to separate the cold side from the hot side is a common approach, but it always includes adding some delay in the signal path. Depending on the communication speed, this delay is sometimes negligible; but more often, especially with the high-speed SPI protocol, the delay in the signal path affects the requirements of the SPI master.

The SPI master must cope with the transmission path delay by shifting the sampling point for received data. In situations where this task requires the use of external logic, this TI design provides the use of that external logic, or even FPGA, without the cost of additional components.

This TI design implements the SPI master protocol with signal path delay compensation in the Programmable Real-time Unit and the Industrial Communication Subsystem (PRU-ICSS). The implementation has been validated with the TIDA-00164, which is a 16-bit, 8-channel, integrated analog input module for PLCs. For the validation, the design team used the TIDA-00164 device coupled with an ISO7141CC isolator and ADS8688 analog-to-digital converter (ADC).

### 2.1 Signal Path Delay from Communication Channel Isolators

Figure 1 shows the system block diagram of the application processor communicating through the SPI protocol with an I/O module over a digital isolator. The application processor contains the SPI master and the I/O module contains the SPI slave. The digital isolator provides galvanic isolation to the SPI communication channel, but the digital isolator also adds delay to all SPI signals. The digital isolator adds delay in both directions, which is from the master to the slave and from the slave to the master.

The master generates the signals  $\overline{\text{CS}}$ , SCLK, and MOSI. The slave receives these signals and uses them to generate the MISO signal, which the master receives. However, the master still uses its self-generated SCLK signal to sample the MISO signal that returns from the slave, as well.

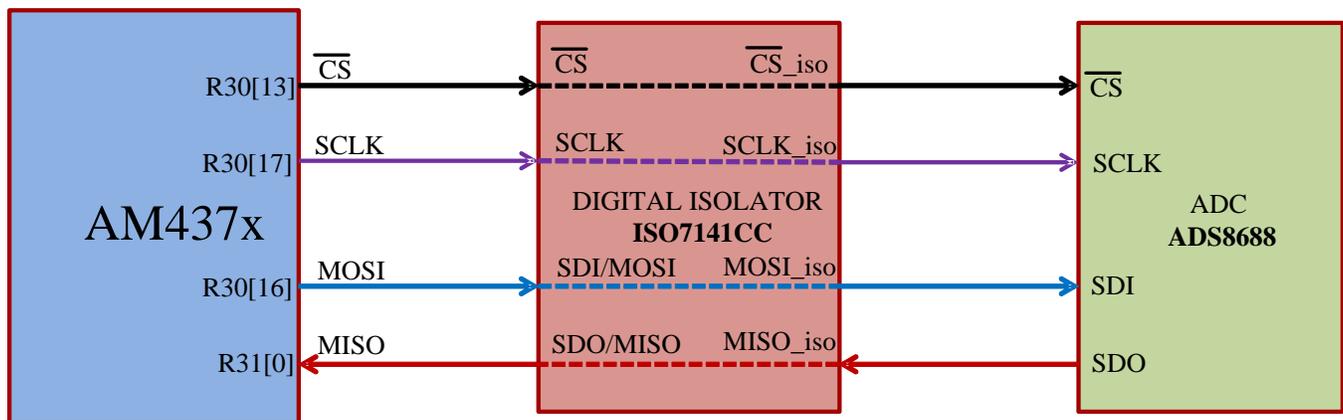


Figure 1. System Block Diagram With Processor, Digital Isolator, and I/O Module

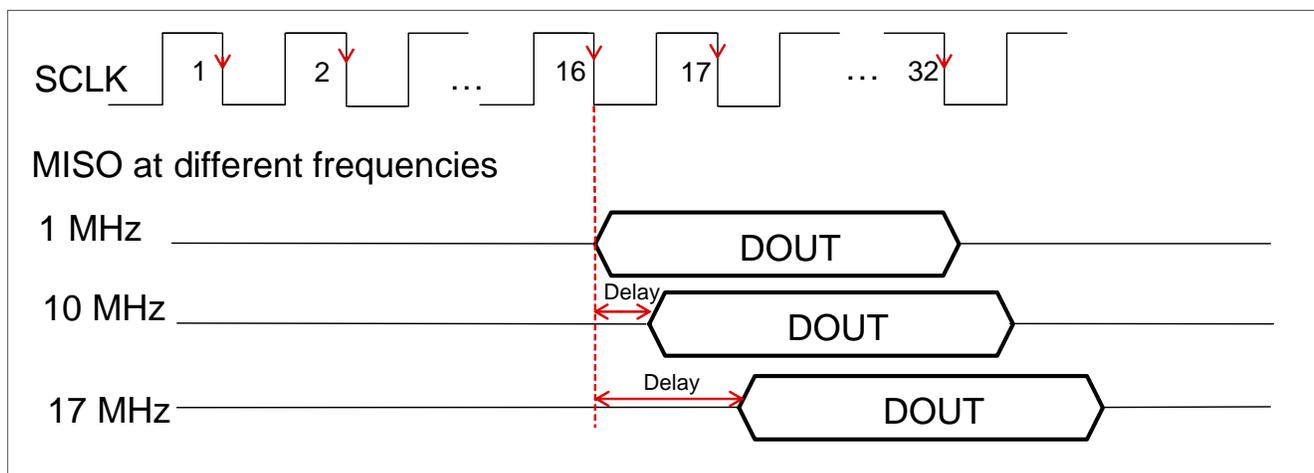
Figure 2 shows the impact of different SPI clock frequencies. The system is configured here to output new data on the falling edge and capture the current data with the rising edge.

At low SPI clock frequencies such as 1 MHz, the delay added by the digital isolator is negligible. The sample edge (rising edge) is well in the middle of the data bit, as output by the slave.

The corner case is at 10 MHz, where the MISO data bits are delayed such that they just output at the sample edge of the SPI clock. This is already an invalid state for sampling because the data may have been changed. Refer to Section 5.1.2 *Analog Input and Filter* and Section 5.1.2.1 *Input Filter Design* in the TIDA-00164 user's guide when measuring those boundaries ([TIDU365](#)).

At 17 MHz the start of the MISO data bit has been shifted by more than one clock cycle, so the MISO data bit is only valid with the next sampling edge. If keeping the same configuration, the master samples all of the data and the result would shift by one while the first MISO data bit would be sampled as invalid. The user must compensate for the sample edge with the generated clock to account for the path delay.

This TI design can account for the path delay time and compensate for the delay path, allowing up to a 16.7-MHz frequency to support the SPI.



**Figure 2. MISO Signal at Master for Different SPI Clock Frequencies**

## 2.2 PRU-ICSS Instances in AM437x Processor

The AM437x processor has two instances of PRU-ICSS, which are labeled as PRU-ICSS0 and PRU-ICSS1. See the block diagram in [Figure 3](#).

This TI design was developed by using the PRU1 core of the PRU-ICSS instance 0 (PRU-ICSS0). Note that in this TI design the PRU firmware (and, thus SPI master) are operable on any of the PRUs, either in ICSS0 or in ICSS1. However, if the designer uses a PRU with a function other than what has been validated in this design, the firmware source code may require changes. If using the PRU0 core or the PRU-ICSS1, some changes may be required to configure the peripherals correctly.

This TI design uses the following peripherals:

- PRU1 core in ICSS0
- PRU1 local data RAM (4KB) in ICSS0
- Register-mapped GPO
- 28-bit serial input module
- Control module configuration register (CFG) in PRU-ICSS0

### PRU-ICSS block diagram

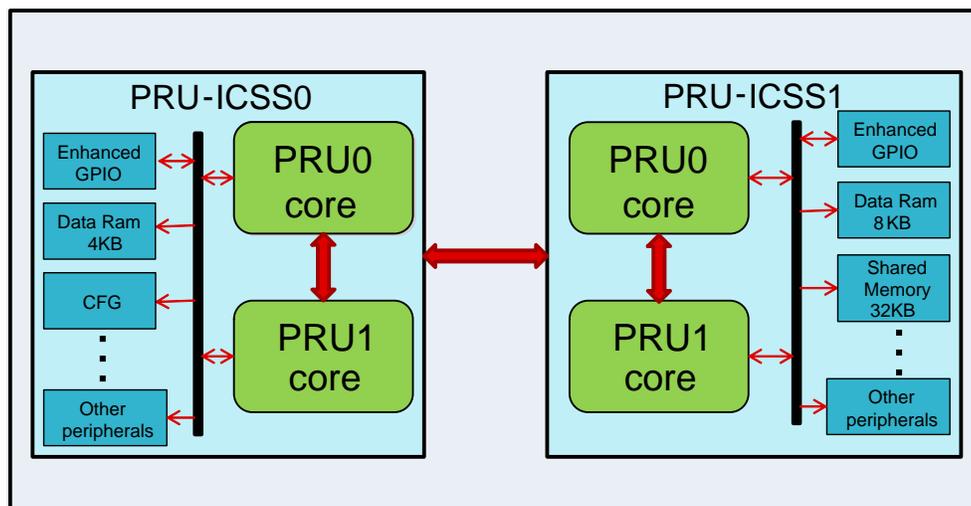


Figure 3. PRU-ICSS Instances on AM437x

### 3 Block Diagram

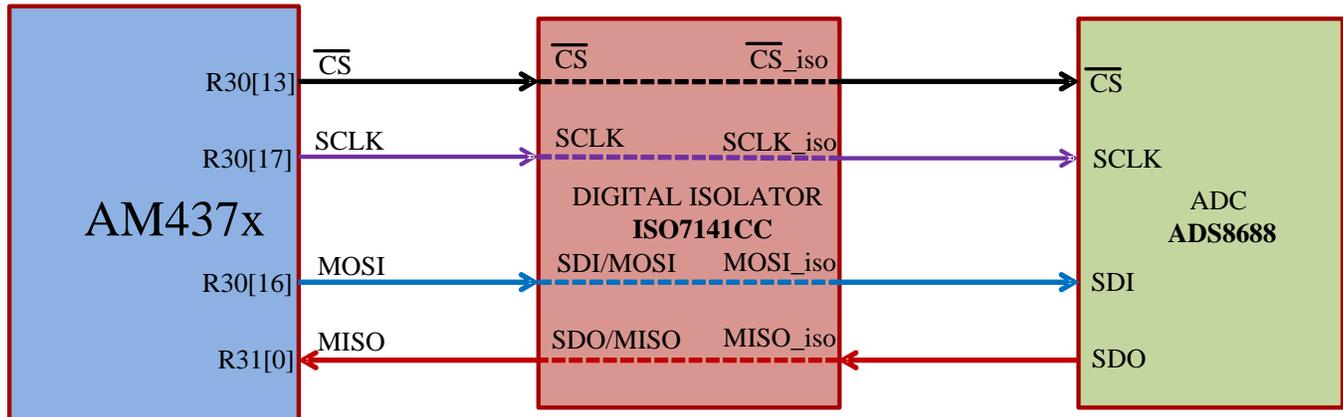


Figure 4. System Block Diagram

#### 3.1 Highlighted Products

##### 3.1.1 AM4379 Processor

Up to 1-GHz Sitara™ ARM® Cortex®-A9 32-bit RISC processor

- NEON™ SIMD coprocessor and ARM Vector Floating Point (VFPv3) coprocessor
- 32KB of L1 instruction and 32KB of data cache
- 256KB of L2 Cache or L3 RAM
- 256KB of on-chip boot ROM
- 64KB of dedicated RAM
- Emulation and debug - JTAG
- Interrupt controller

Programmable real-time unit subsystem and industrial communication subsystem (PRU-ICSS)

- Supports protocols such as EtherCAT®, PROFIBUS, PROFINET, EtherNet/IP™, EnDat 2.2, and more
- Two programmable real-time units (PRUs) subsystems with two PRU cores each
- 32-bit load and store RISC processor capable of running at 200 MHz
- 12KB (PRU-ICSS1), 4KB (PRU-ICSS0) of Instruction RAM with Single-Error Detection (Parity)
- 8KB (PRU-ICSS1), 4KB (PRU-ICSS0) 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 (PRU-ICSS1 only) of shared RAM with single-error detection (parity)
- Three 120-byte register banks accessible by each PRU
- Three 120-byte register banks accessible by each PRU
- Local interconnect bus for connecting internal and external masters to the resources inside the PRU-ICSS
- Peripherals inside the PRU-ICSS:
  - One UART port with flow control pins, supports up to 12 Mbps
  - One enhanced capture (eCAP) module
  - Two MII Ethernet ports that support industrial Ethernet, such as EtherCAT
  - One MDIO port

- On-chip memory (shared L3 RAM)
  - 256KB of general-purpose on-chip memory controller (OCMC) RAM
  - Accessible to all masters

#### External memory interfaces (EMIF)

- DDR controllers:
  - LPDDR2: 266-MHz clock (LPDDR2-533 data rate)
  - DDR3 and DDR3L: 400-MHz clock (DDR-800 data rate)
  - 32-bit data bus
  - 2GB of total addressable space
  - Supports one x32, two x16, or four 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, NOR, Muxed-NOR, SRAM)
  - Uses BCH code to support 4-, 8-, or 16-bit ECC
  - Uses hamming code to support 1-bit ECC

See the AM4379 data sheet for a complete list of features [ 1].

### 3.1.2 AM437X Industrial Development Kit (IDK) EVM Hardware Specifications

#### Features

- AM4379 ARM Cortex-A9
- 1GB DDR3, QSPI-NOR flash
- Discrete power solution
- EnDat connectivity for motor feedback control
- 24-V power supply
- USB cable for JTAG interface and serial console

#### Software and tools

- SYS/BIOS real-time operating system (OS)
- Starterware base port
- TI Code Composer Studio™ (CCS) integrated development environment (IDE)
- Application stack for industrial communication protocols
- Sample industrial applications

**Connectivity**

- PROFIBUS interface
- CANOpen
- EtherCAT
- EtherNet/IP
- PROFINET
- Sercos III
- IEC61850
- PWM
- Motor axis position feedback
- Up to 3-phase motor drive connector
- Sigma delta decimation filter
- Digital inputs and outputs (I/O)
- SPI
- UART
- JTAG

See the AM437X IDK tool folder for a complete list of features and design resources:

[www.ti.com/tool/tmdxidek437x](http://www.ti.com/tool/tmdxidek437x).

### 3.1.3 ADS8688

**Features**

- 16-bit ADCs with integrated analog front-end (AFE)
- 4-, 8-Channel MUX with auto and manual scan
- Channel-independent programmable input ranges:
  - Bipolar:  $\pm 10.24$  V,  $\pm 5.12$  V,  $\pm 2.56$  V
  - Unipolar: 10.24 V, 5.12 V
- 5-V analog supply: 1.65-V to 5-V I/O supply
- Constant resistive input impedance: 1 M $\Omega$
- Input overvoltage protection: Up to  $\pm 20$  V
- On-chip, 4.096-V reference with low drift
- Excellent Performance:
  - 500-kSPS aggregate throughput
  - DNL:  $\pm 0.5$  LSB; INL:  $\pm 0.75$  LSB
  - Low drift for gain error and offset
  - SNR: 92 dB; THD:  $-102$  dB
  - Low power: 65 mW
- AUX input  $\rightarrow$  direct connection to ADC inputs
- SPI<sup>TM</sup>-compatible interface with daisy-chain
- Industrial temperature range:  $-40^{\circ}\text{C}$  to  $125^{\circ}\text{C}$
- TSSOP-38 package (9.7 mm  $\times$  4.4 mm)

See the ADS8688 product website for a complete list of features, data sheet, and design resources:

[www.ti.com/product/ads8688](http://www.ti.com/product/ads8688).

### 3.1.4 ISO7141CC

#### Features

- Maximum signaling rate: 50 Mbps (with 5-V supplies)
- Robust design with integrated noise filter
- Default output low option (suffix F)
- Low power consumption, typical ICC per channel (with 3.3-V supplies):
  - ISO7131: 1.5 mA at 1 Mbps, 2.6 mA at 25 Mbps
  - ISO7140: 1 mA at 1 Mbps, 2.3 mA at 25 Mbps applications
  - ISO7141: 1.3 mA at 1 Mbps, 2.6 mA at 25 Mbps
- Low propagation delay: 23ns typical (3.3-V supplies)
- Wide temperature range: –40°C to 125°C
- 50-kV/ $\mu$ s transient immunity, typical
- Long life with SiO<sub>2</sub> isolation barrier
- Operates from 2.7-V, 3.3-V, and 5-V supply and logic levels
- Small QSOP-16 package

See the ISO7141CC product website for a complete list of features, data sheet, and design resources:  
[www.ti.com/product/ISO7141CC](http://www.ti.com/product/ISO7141CC).

## 4 System Design Theory

### 4.1 PRU-ICSS Introduction

The PRU-ICSS consists of two programmable RISC cores with dedicated RAM and ROM (see Figure 5). The RISC cores operate at 200 MHz with a single-cycle instruction execution (5 ns) that enables real-time deterministic programming. For more information on the PRU-ICSS, view the TI Wiki page: <http://processors.wiki.ti.com/index.php/PRU-ICSS>.

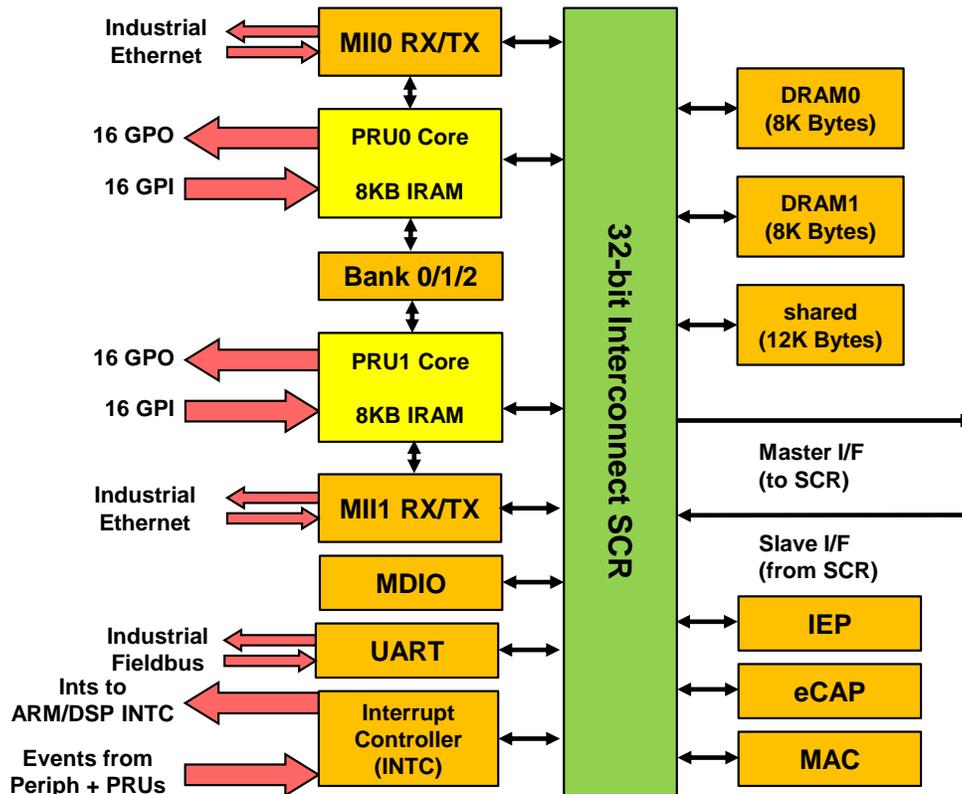


Figure 5. PRU-ICSS Block Diagram

This TI design uses PRU1 core inside PRU\_ICSS0, mainly due to pin availability on the AM437x Industrial Development Kit (IDK) evaluation module (EVM). The programmer has the option to change the PRU-ICSS firmware to effectively use the PRU0 core instead of the PRU1.

The PRU-ICSS firmware generates the SPI output signals using the register-mapped general purpose output (GPO) mode. The input signal MISO is oversampled with the 28-bit input shift register mode.

## 4.2 SPI Interface Between AM437x and ADS8688

Figure 6 shows the system block diagram and the SPI communication signals between the AM437x (SPI master) and the ADS8688 (SPI slave). The digital isolator ISO7141CC is in the SPI signal path. This digital isolator adds a signal path delay in both directions at approximately 23 ns per direction.

The PRU-ICSS supports the following SPI signals as SPI master:

- nCS (output) → SPI chip select
- SCLK (output) → SPI clock
- MOSI (output) → SPI master out slave in
- MISO (input) → SPI master in slave output

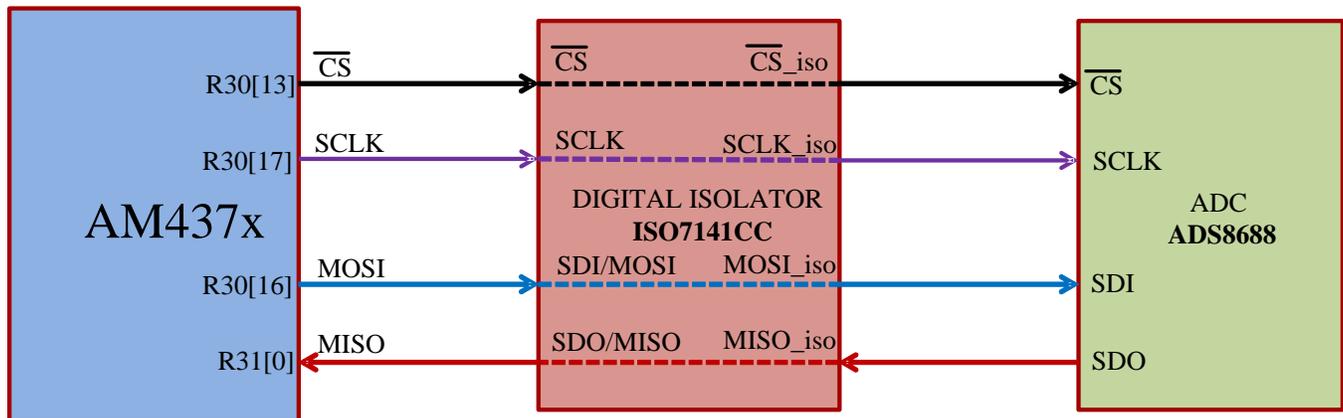


Figure 6. System Block Diagram

## 4.3 TIDA-00164 ADC—ADS8688

The TI design TIDA-00164, which is a 16-bit, 8-channel, integrated analog input module for PLCS, has been used to validate the SPI master communication as well as the delay path compensation function. The TIDA-00164 design uses the ISO7141CC to isolate the SPI communication path and uses the ADC8688 as an ADC.

The following lists the features of the TIDA-00164 TI design (see the board in Figure 7):

- Single-supply industrial control analog input module
- Eight channels; each channel can be configured as a current or voltage input
- All eight channels can be typically provided with the analog voltage and current ranges:
  - $\pm 2.5$  V,  $\pm 5$  V,  $\pm 10$  V, 0 to 5 V, 0 to 10 V
  - 0 to 20 mA, 4 to 20 mA
- 16-bit SAR ADC
- SPI interface, serial clock frequency (SCLK) up to 17 MHz
- Digital isolator ISO7141CC provides digital signal isolation between the host processor and the measurement side
- Applications
  - Process control end equipment
  - Programmable Logic Controllers (PLCs)
  - Distributed Control Systems (DCS)
  - Data Acquisition Systems (DAS)

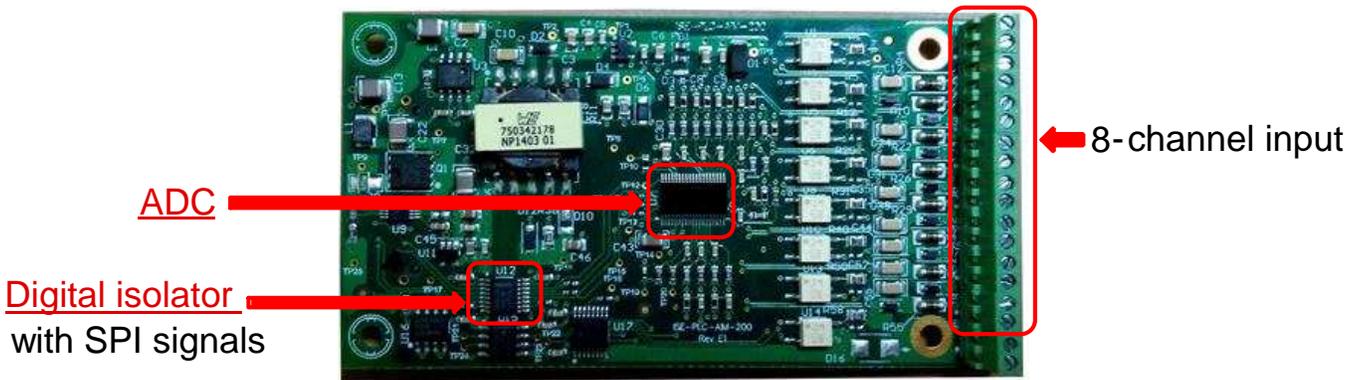


Figure 7. TIDA-00164 Board

Figure 8 shows the system block diagram. The ADS8688 is an 8-channel, integrated data acquisition system based on a 16-bit SAR ADC, operating at a throughput of 500 kSPS (sampling frequency), and a serial clock frequency of up to 17 MHz.

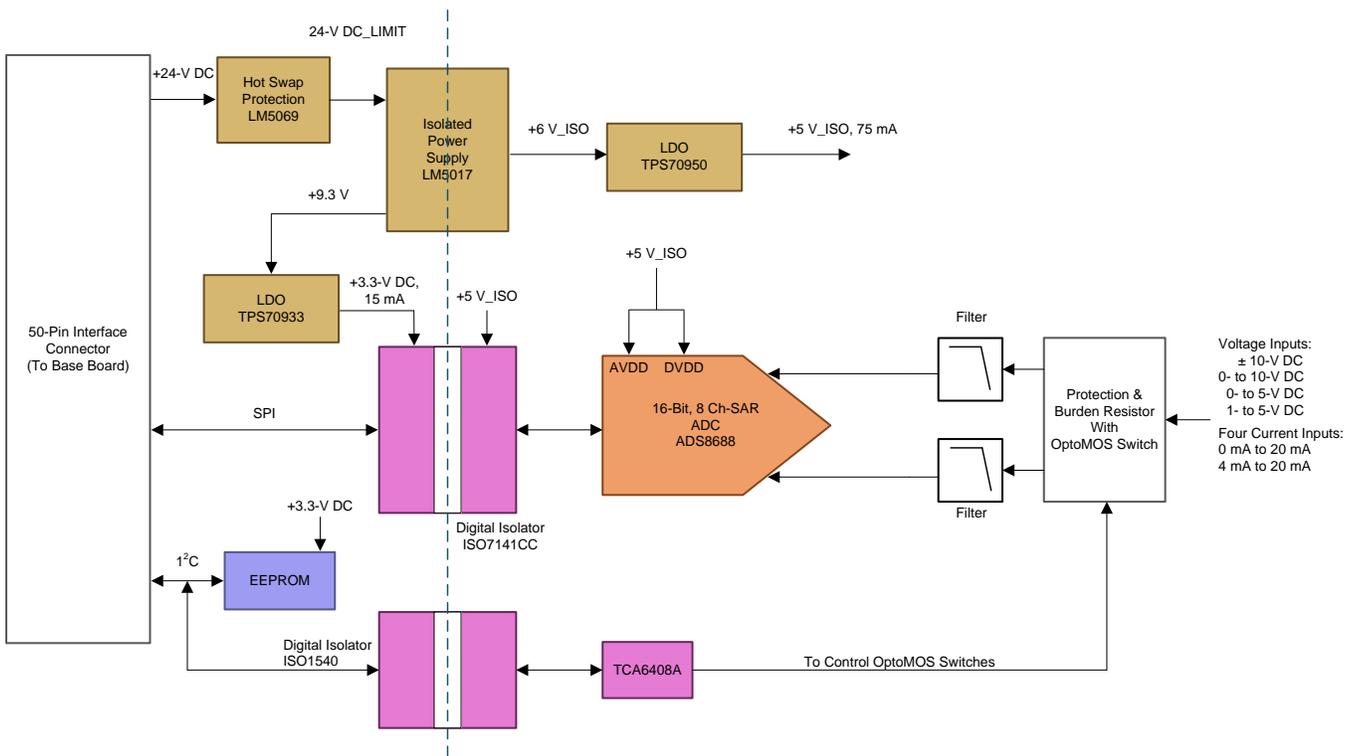


Figure 8. TIDA-00164 System Block Diagram

Figure 9 shows the SPI data frame format and the sample time diagram of the ADS8688 device. The master sends 16 bits of data on the SDI (MOSI) signal to the slave. The slave samples the data with the falling edge. At the 17th clock cycle, the slave sends the data on the SDO (MISO) signal to the master. The master must also sample the bits with the falling edge starting with the 17th clock cycle.

Due to the propagation path delay through the digital isolator, the SDO bits arrive at the master after twice the time of the propagation path delay of 23 ns, compared against the master's own SCLK. This total propagation path delay of 46 ns is negligible for SPI frequencies below 10 MHz, but with higher serial clock frequencies the delay must be taken into consideration by shifting the sample edge to the correct position. The next section addresses this approach.

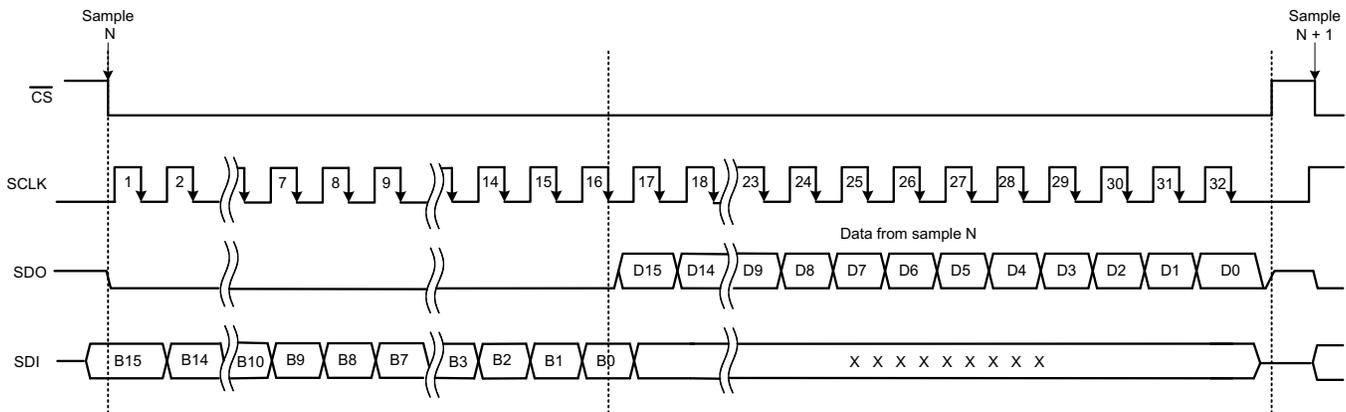


Figure 9. ADS8688 Sample Time Diagram and SPI Data Frame Format

Also note that the ADS supports two kind of access registers: command register and program register. With the command register, the processor receives the ADC samples; with the program register, the processor configures the mode of operation in the ADC. Refer to the ADS data sheet for further information.

#### 4.4 PRU Firmware Implementation

##### 4.4.1 PRU-ICSS Registers R30 and R31

The PRU has a total of 32 registers named R0 to R31, respectively. Registers R30 and R31 are not available for generic use but do support configuration specific functions. The functions SDO and the 28-bit serial input shift register are used with this TI Design.

The following Figure 10 shows the special function registers R30 and R31 and how they function in this TI design.

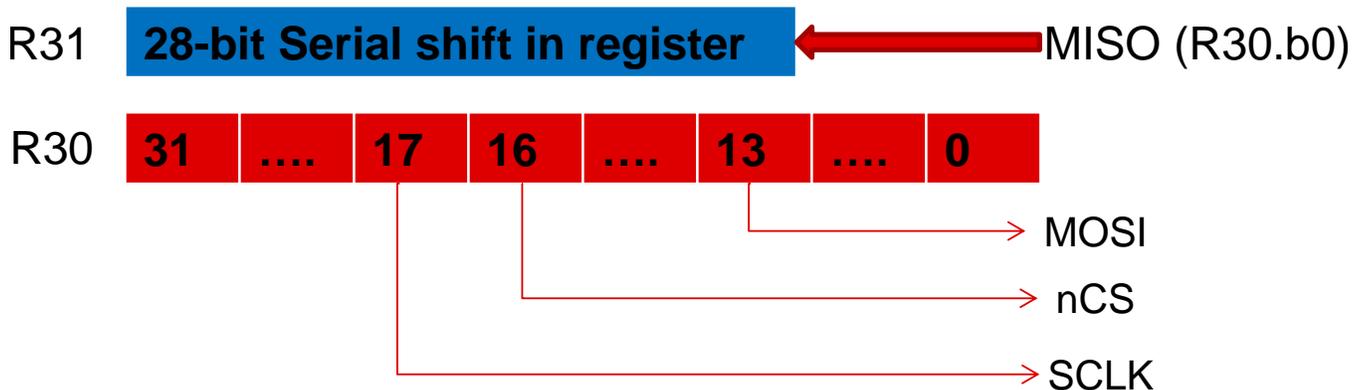


Figure 10. Register R30 and R31 Special Function Support

#### 4.4.2 GPO Mode

The register R30 is configured to register-mapped GPO pins to generate the nCS, SCLK, and MOSI signals. The PRU performs the bit-banging technique to generate these signals and to emulate the SPI communication transaction.

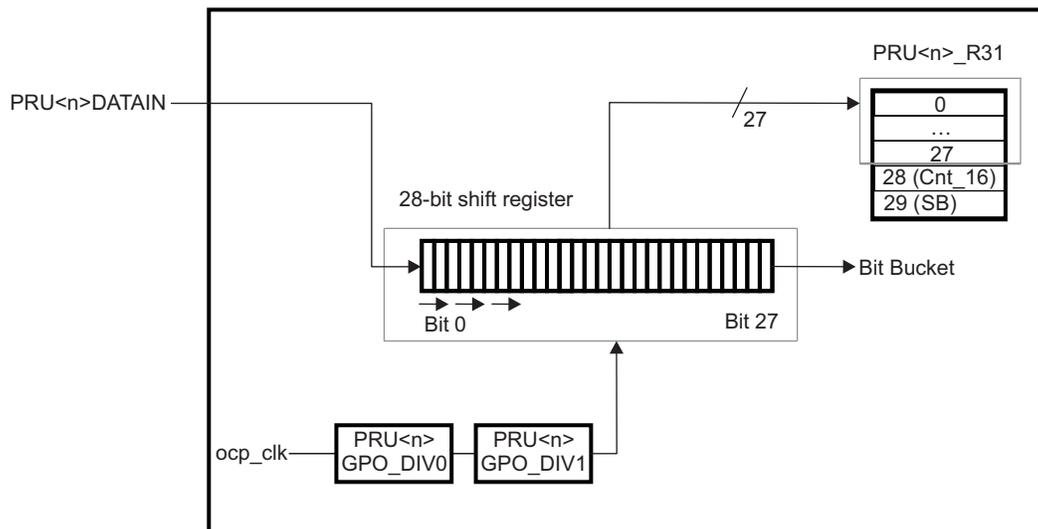
**Table 1. Table 2: Output Pin Assignment**

NAME	PRU PIN	COMMENT
nCS	pr0_pru1_gpo16	SPI chip select
SCLK	pr0_pru1_gpo17	SPI serial clock
MOSI	pr0_pru1_gpo13	SPI master-out slave-in

#### 4.4.3 28-Bit Serial Input Shift Register

The register R30 is configured as a 28-bit input shift register. The 28-bit shift input mode must be oversampled eight times with respect to the generated SCLK. The sampling clock of the 28-bit shift input is configured in the CFG\_CPCFGn register.

In the 28-bit shift input mode (see Figure 11), the general-purpose input pin PRU<n>\_DATAIN is sampled and shifted into a 28-bit shift register on an internal clock pulse, which feeds into a single one-stage flop synchronizer. This mode is enabled by configuring the CFG register of the PRU-ICSS. The register fills in least significant bit (LSB) order (from bit 0 to 27) and then overflows into a bit bucket. The 28-bit register is mapped to PRU<n>\_r31\_status [0:27] and can be cleared in the software through the PRU-ICSS CFG register space. The shift rate is controlled by the effective divisor of two cascaded dividers applied to the ocp clock signal (ocp\_clk).



**Figure 11. 28-Bit Shift Register Function Block**

#### 4.4.4 PRU-ICSS Firmware

The PRU-ICSS firmware is programmed using an RISC assembler for this TI design. This programming method improves the implementation feasibility for real-time critical applications. Note that TI's Code Composer Studio™ software version 6 (CCSv6) also allows the user to program the PRU in "C" language; however, the "C" language is not used for this TI design to meet the real-time constraints.

View the *PRU Assembly Instructions* manual on the TI Wiki:

[http://processors.wiki.ti.com/index.php/PRU\\_Assembly\\_Instructions](http://processors.wiki.ti.com/index.php/PRU_Assembly_Instructions).

The following Figure 12 explains the PRU-ICSS firmware program flow of this TI design. Please refer to the file *spi1.asm*, which contains the PRU-ICSS firmware source code.

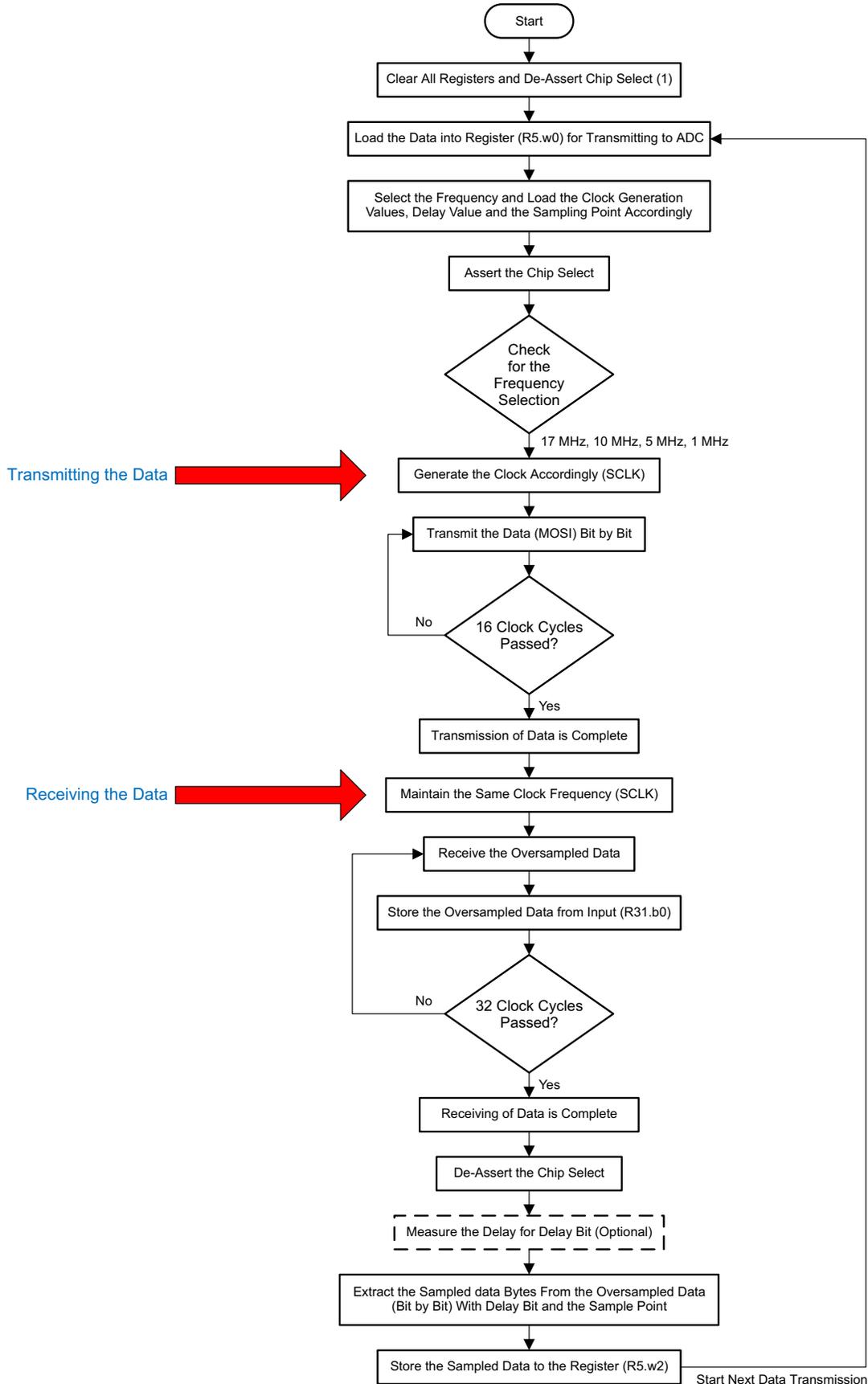


Figure 12. PRU-ICSS Firmware Program Flow

#### 4.4.5 Configuration Memory

The SPI configuration data for clock speed, signal path delay, and sample position are stored in the PRU1 data RAM. This configuration data is configured by the ARM code; whereas this TI design configures this data through a GEL script within CCSv6. [Table 2](#) shows the parameter assignment in the PRU1 data RAM. [Table 3](#) shows the SPI frequency.

**Table 2. PRU1 Data RAM Usage**

PRU1 DATA RAM ADDRESS	PARAMETER	PARAMETER SIZE (BYTES)
0x0000	SPI frequency value	4
0x0004	Loop delay value	4
0x0008	Sample bit	4

**Table 3. SPI Frequency**

SPI FREQUENCY VALUE	SPI SCLK FREQUENCY
0	17 MHz
1	10 MHz
2	5 MHz
3	1 MHz

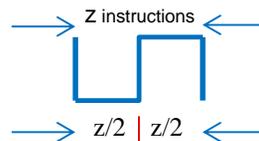
#### SPI clock generation

The SPI serial clock (SCLK) is generated by the PRU using the bit-banging technique, which means that the PRU sets and clears the SCLK bit to emulate the SPI clock. There is a delay time in between the clock bit-banging and the delay time is different for each clock frequency.

This TI design has been tested with SPI clock frequencies of 1 MHz, 5 MHz, 10 MHz, and 17 MHz. Generating additional frequencies to this programming mode by changing the appropriate SPI configuration parameters in the PRU1 data RAM is possible.

To calculate the SPI clock frequencies between 1 MHz to 12 MHz, the following equations are valid.

- Let X MHz be the selected frequency, then  $1 / X \text{ MHz} = Y \text{ nsec}$
- To execute one instruction, the PRU takes 5 nsec, so for Y nsec the PRU should execute  $Y \text{ nsec} / 5 \text{ nsec} = Z$  (round to the nearest even number) instructions



**Figure 13. SPI Clock Frequency Generation**

- Setting the clock requires  $Z / 2$  number of instructions; and to clear the clock requires  $Z / 2$  number of instructions (see [Figure 13](#))
- To maintain a 50% duty cycle for the clock for both transmission and reception of the data, use the following delay value byte positions:
  - On transmission of data b0 on set and b1 on clear
  - On reception of data b2 on set and b3 on clear
- Each  $b_n$  is 1 byte in size, or 4 bytes total. Each  $b_n$  entry corresponds to the loop delay value parameter located in the PRU1 data RAM.

Each  $b_n$  can be calculated as the following Equation 1 shows:

$$\begin{aligned}
 b_0 &= (Z / 2) - 2 \\
 b_1 &= b_0 - 6 \\
 b_2 &= b_0 \\
 b_3 &= b_0 - 3
 \end{aligned} \tag{1}$$

The following Equation 2 shows an example for 10 MHz:

$$1 / 10 \text{ MHz} = 100 \text{ nsec}$$

$$100 \text{ nsec} / 5 \text{ nsec} = 20 \text{ number of instructions}$$

where

- $b_0 = 8$
  - $b_1 = 2$
  - $b_2 = 8$
  - $b_3 = 5$
- (2)

For 17 MHz, the following loop delay values must be used:

where

- $b_0 = 3$
  - $b_1 = 0$
  - $b_2 = 4$
  - $b_3 = 1$
- (3)

### Oversampling clock configuration

This TI design requires the use of an 8-bit oversampling frequency. To use this oversampling frequency, the user must configure two cascaded dividers in the PRU-ICSS CFG register:

- The sample and shift clock is controlled by a programmable divider that can divide the 200-MHz clock
- These two values for the cascaded dividers can be calculated as seen in Figure 14:
  - $\text{PRU}\langle n \rangle\text{GPO\_DIV0} = D_0$
  - $\text{PRU}\langle n \rangle\text{GPO\_DIV1} = D_1$

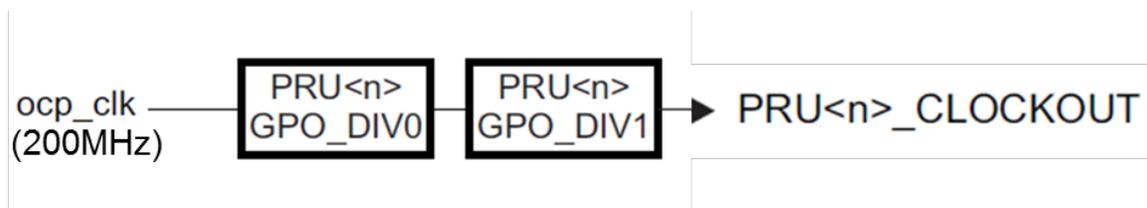


Figure 14. 28-Bit Shift Register Clock Generation

- Let (x MHz) be the SPI clock frequency, then for 8-bit oversampling frequency the  $\text{PRU}\langle n \rangle\text{_CLOCKOUT}$  must be configured to  $8 \times (X \text{ MHz})$  frequency
  - The divider values can be derived from the following Equation 4:

$$8 \times (X \text{ MHz}) = (200 \text{ MHz} / (D_0 \times D_1))$$

$$D_0 \times D_1 = 200 \text{ MHz} / 8 \times (X \text{ MHz})$$

where

- $D_0$  and  $D_1$  are represented as integer values from 1.0, 1.5, to 16.0 in 0.5 increments (4)

These divider values are represented in the divider register as: 0h for div 1.0, 1h for div 1.5, 2h for div 2.0, and so on until 1Eh for div 16.0.

By using the preceding [Equation 4](#), the following example frequencies can be calculated.

**For a 5-MHz frequency:**

$$5 \text{ MHz} \times 8 = (200 \text{ MHz} / (D0 \times D1))$$

$$D0 \times D1 = 200 \text{ MHz} / 40 \text{ MHz}$$

$$D0 \times D1 = 5$$

where

- D0 = 8h
  - D1 = 0h
- (5)

**For a 10-MHz frequency:**

$$10 \text{ MHz} \times 8 = (200 \text{ MHz} / (D0 \times D1))$$

$$D0 \times D1 = 200 \text{ MHz} / 80 \text{ MHz}$$

$$D0 \times D1 = 2.5$$

where

- D0 = 3h
  - D1 = 0h
- (6)

**For a 17-MHz frequency:**

$$10 \text{ MHz} \times 8 = (200 \text{ MHz} / (D0 \times D1))$$

$$D0 \times D1 = 200 \text{ MHz} / 80 \text{ MHz}$$

$$D0 \times D1 = 1.47$$

where

- The divider cannot be configured to 1.47, so it must be rounded to 1.5
  - D0 = 1h
  - D1 = 0h
- (7)

If the divider is set to 1h, the oversampling clock is 16.667 MHz. Note that the PRU clock (SCLK) generation of 17 MHz does result in:

- $1 / 17 \text{ MHz} = 58.823 \text{ ns}$ , rounded to 60 ns

So if a 60-ns time period is used for the SPI clock frequency, the 28-bit shift clock is the same as the SPI frequency clock, and there is no drift in the sampled data.

**For an 11-MHz frequency:**

$$11 \text{ MHz} \times 8 = (200 \text{ MHz} / (D0 \times D1))$$

$$D0 \times D1 = 200 \text{ MHz} / 88 \text{ MHz}$$

$$D0 \times D1 = 2.27$$
(8)

The divider cannot be configured as 2.27, so the divider must be chosen as 2.25, with 1.5 for D0 and 1.5 for D1.

In situations when the PRU clock (SCLK) generation is configured to 11 MHz:

- $1 / 11 \text{ MHz} = 90.909 \text{ nsec}$ , rounded to 90 nsec

If the PRU clock generation uses a 90-nsec time period, it generates a clock frequency of 11.11 MHz, which is the same frequency as the oversampled data rate.

### Delay measurement and sample bit

To take the appropriate bit sample of the oversampled data, the user must determine the sample point within the bit stream of the 28-bit input shift register. To calculate the sample point, the user must either calculate or determine the signal path delay by measuring. This sample point is the starting bit of the received data.

According to [Figure 9](#), the ADS does output the first bit with the 16th falling clock cycle. Due to the delay of the digital isolator, there is some delay during the output of the first bit, which is double the amount of delay in the delay path. The cause of this delay ratio is the clock and the MISO response traveling through the digital isolator. For the ISO7141CC device, the delay is 46 ns.

To calculate this delay, the user is to assume that the starting bit is “1” and that the PRU-ICSS firmware detects the left-most bit position. Under these assumptions, the user can utilize the PRU firmware to calculate the delay (position of the starting bit).

From this delay value, the user can calculate the sample bit by adding 4 to the delay value.

### GEL script example

The project GEL file contains an example configuration memory for each of the supported frequencies. As soon as the GEL function is executed, it sets the parameters in the configuration memory for the selected frequency.

## 4.5 Summary

The PRU-ICSS firmware integrates the support for the SPI communication link between the AM437x and the ADS8688. The PRU-ICSS firmware supports various frequencies from 1 MHz up to 16.7 MHz. Note that the maximum serial clock frequency for the ADS8688 device is 17 MHz. The established communication link works with a maximum signal path delay of 2 clock cycles at 17 MHz, which is 120 ns. The user can calculate the delay manually or determine the delay by using the PRU-ICSS firmware (in situations where the response from the slave is known). Note that the signal path delay depends on the digital isolator.

## 5 Getting Started Hardware

The following hardware and software is required to demonstrate and evaluate the multi-protocol industrial Ethernet detection with PRU-ICSS.

### 5.1 Hardware Requirements

- TMDXIDK437X EVM
- TIDA-00164 EVM

The TIDA-00164 EVM connector does not fit directly into the expansion connector of the TMDXIDK437X device. For testing purposes, the user can either build an adapter board or initially connect the SPI bus of both boards through blue wires.

### 5.2 Software Requirements

- Code Composer Studio v6.0 (CCSv6) for Windows ([http://processors.wiki.ti.com/index.php/Download\\_CCS](http://processors.wiki.ti.com/index.php/Download_CCS))
- PRU C-Compiler v2.1.1
- AM437X SYSBIOS Industrial SDK v02.00.00.02

The TIDEP0033 TI design consists of a PRU project for CCSv6. The PRU project generates a CCS debug file (.out) during the build process. Note that if the user modifies the PRU-ICSS firmware, the PRU project must be rebuilt for the firmware changes to take effect.

### 5.3 CCS Import, Build, and Download

First import both projects into CCS through the *Import CCS Eclipse Projects* window (see [Figure 15](#) and [Figure 16](#)).

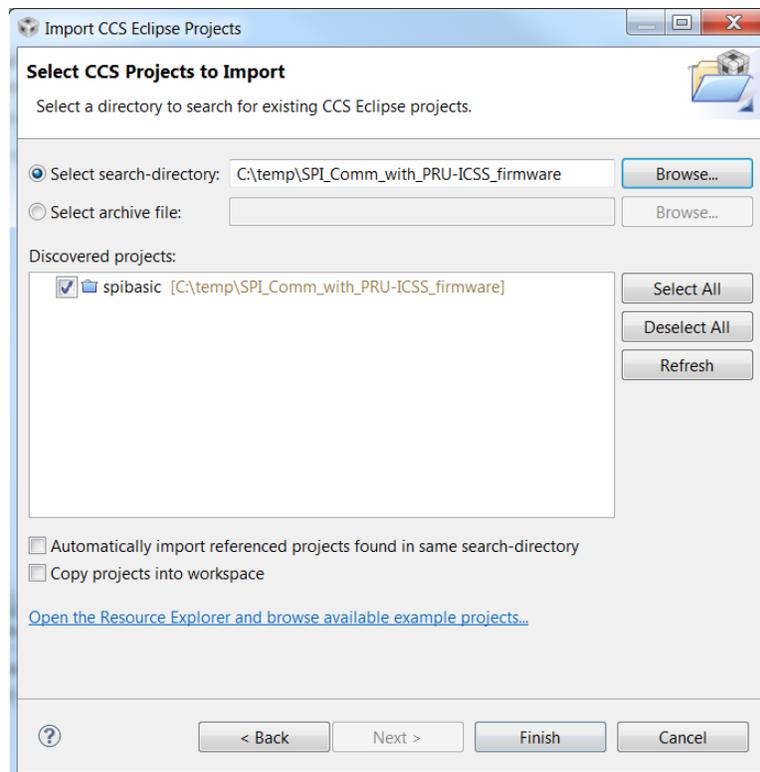


Figure 15. CCS Import Dialog for ARM Project

After importing the projects, the project explorer shows the PRU project *spibasic* (see Figure 16).

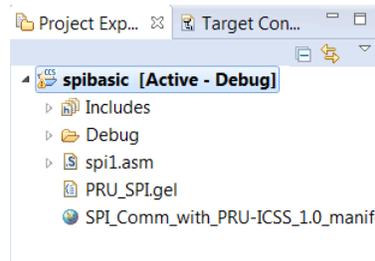


Figure 16. Project View

The next step is to rebuild the PRU project:

- In the *Project Explorer* task pane, select the *spibasic* folder with the cursor
- From the menu bar, select: *Project* → *Build Project*
- This process builds the PRU-ICSS project and generates the file *spibasic.out*

The next step is to generate a target configuration file (if it does not exist already) (see Figure 17). Generating this file allows the CCS software to connect through the JTAG with the TMDXIDK437X EVM.

- From the menu bar, select: *File* → *New* → *Target Configuration*
- In the basic panel:
  - Under the *All Connections* panel, select: Texas Instruments XDS100v2 USB Emulator
  - In *Board or Device*, select: IDK\_AM437X
- In the cascaded advanced panel:
  - Select *CortexA9*
  - Select the GEL file *AM437x\_EVMs.gel* from the SDK in the folder *sdk\tools\gel\AM437X*
  - Click the *Save* button

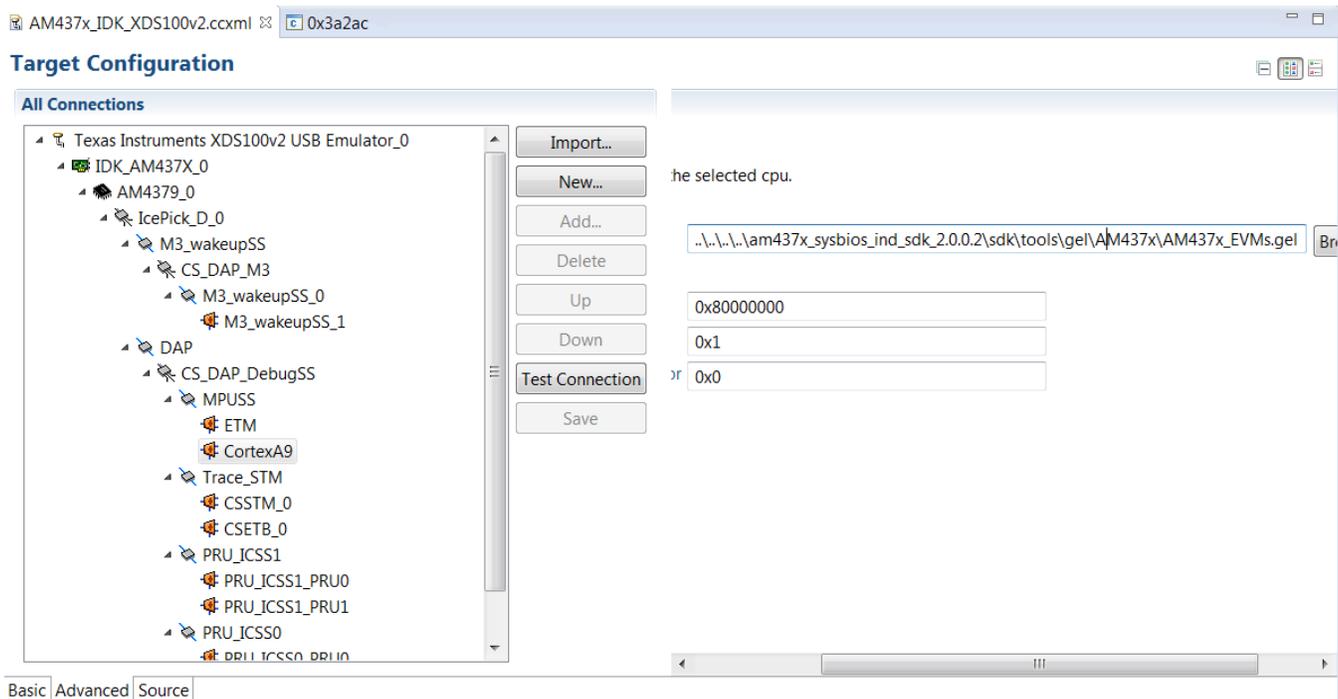


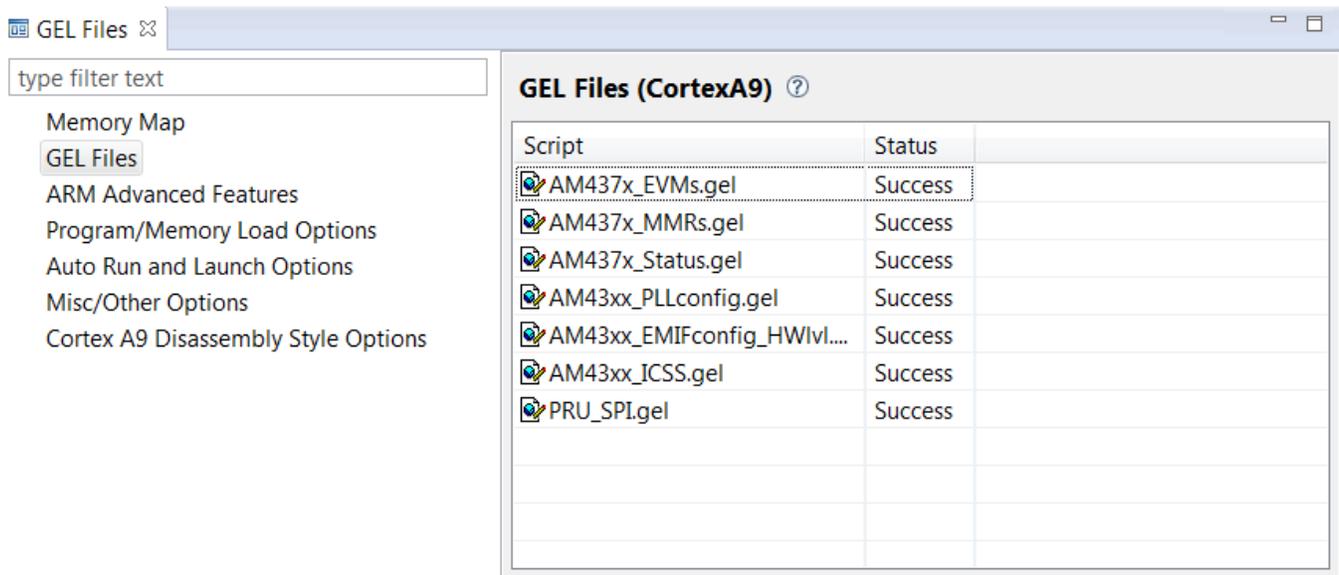
Figure 17. Target Configuration File

Connect the TMDXIDK437X EVM to a 24-V power supply; connect a USB cable between the IDK board and the PC. Next, launch the target configuration file to set up the JTAG debug connection between CCSv6 and the ICE EVM.

- In the *Debug* window, select the *CortexA9* entry
- From the menu bar, select: *Run* → *Connect Target*
- Select: *Scripts* → *AM43xx System Initialization* → *AM43xx\_IDK\_EVM\_Initialization*
- Select: *Scripts* → *ICSS* → *ICSSPRMC\_Enable*

The next step is to load the *PRU\_SPI.gel* file into CCS.

- From the menu bar, select: *Tools* → *GEL Files*
- Right-click in the script section of the *GEL Files* window and select *Load GEL*
- Browse to the *SPI\_Comm* project and select the *PRU\_SPI.gel*



**Figure 18. Load PRU\_SPI.gel**

The next step is to configure the SPI communication parameter RAM and load the PRU-ICSS firmware.

- In the *Debug* window, select the *CortexA9* entry
- From the menu bar, select: *Scripts* → *PRU\_SPI* → *PRU\_SPI\_Init*
- This configures the SPI parameter RAM
- In the *Debug* window, select the *PRU\_ICSS0\_PRU1* entry
- Select: *Run* → *Connect Target*
- Select: *Run* → *Load* → *Load Program*
- Select the file *spbibasic.out* using the *Browse* button and press the *OK* button two times when prompted
- The program is now downloaded into the PRU processor

Depending on the CCS version, the PRU application executes immediately. To restart the PRU firmware and to stop at main, run the following steps.

- From the menu bar, select: *Run* → *Suspend*
- Place a breakpoint at the second set of instructions (see [Figure 19](#))
- Select: *Run* → *Restart*

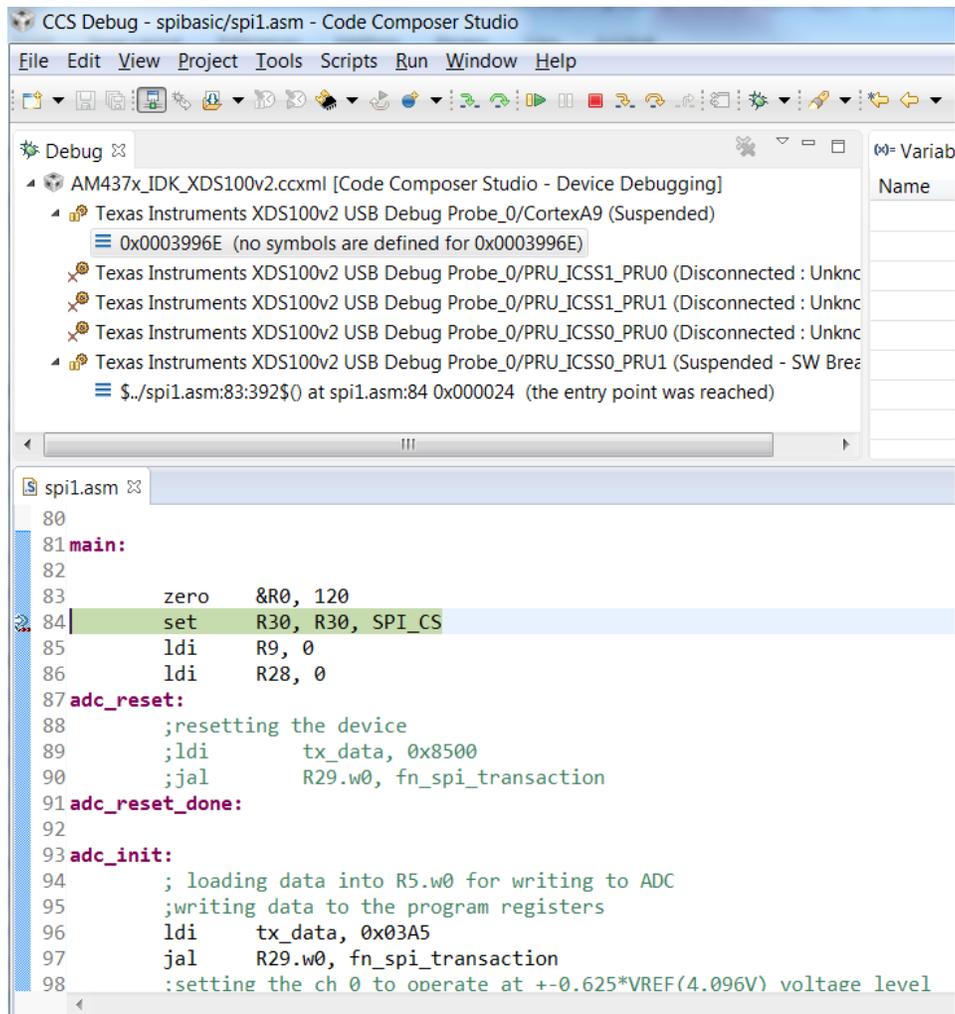


Figure 19. CCS Debug View After Program Download

## 6 Test Setup

Figure 20 shows the test setup. A function generator is used as the signal source for the ADS8688 device. The ADS8688 is triggered by SPI transactions from the AM437x to read out the ADC samples.

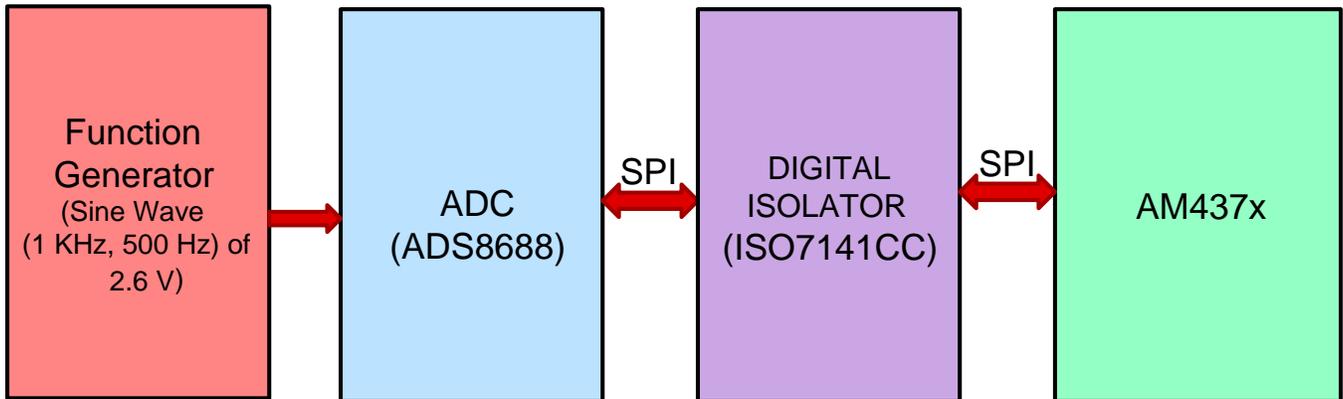


Figure 20. Test Setup

## 7 Test Data

### 7.1 Validation of Communication Link

The validation of the SPI communication was performed through the command register for different frequencies. Using this method ensures the validation of all 16-bits on the MISO line (see Figure 21).

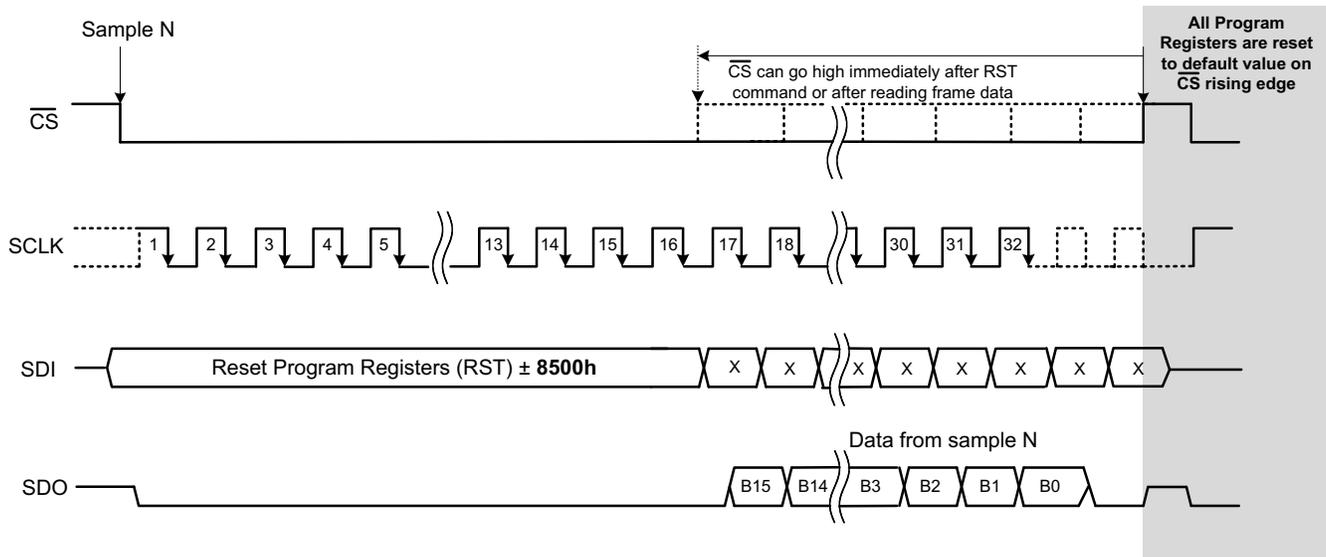


Figure 21. Command Register Frame Format

By using the command register, the user can transmit the data to the 8-channel input on SDI line and can receive the data from samples N on the SDO line. To implement these data transmissions, the testing users generated a sine wave of 1 KHz and 500 Hz with 2.6 V using a function generator and given to the ADC and tested the SPI communication. To successfully validate the 16-bit data, the user must configure the ecap timer with 100  $\mu$ s for every SPI transaction.

The SPI data was accumulated in PRU memory and plotted using the CCS graph function.

### 7.1.1 1-KHz Sine Wave on ADC Input

Figure 22 shows the SPI communication and the sine wave (ADC Input) of 1 KHz at 2.6 V.

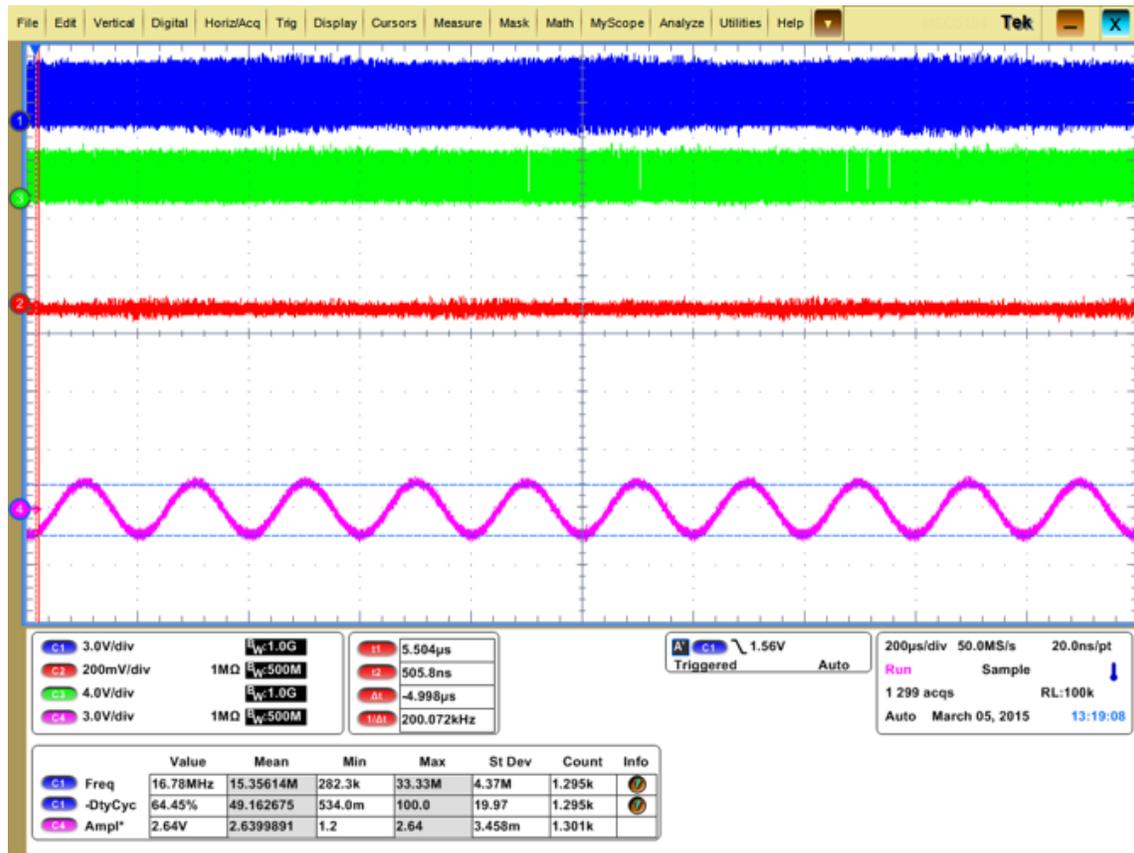


Figure 22. Oscilloscope Capture of ADC Input (1 KHz) and SPI Communication

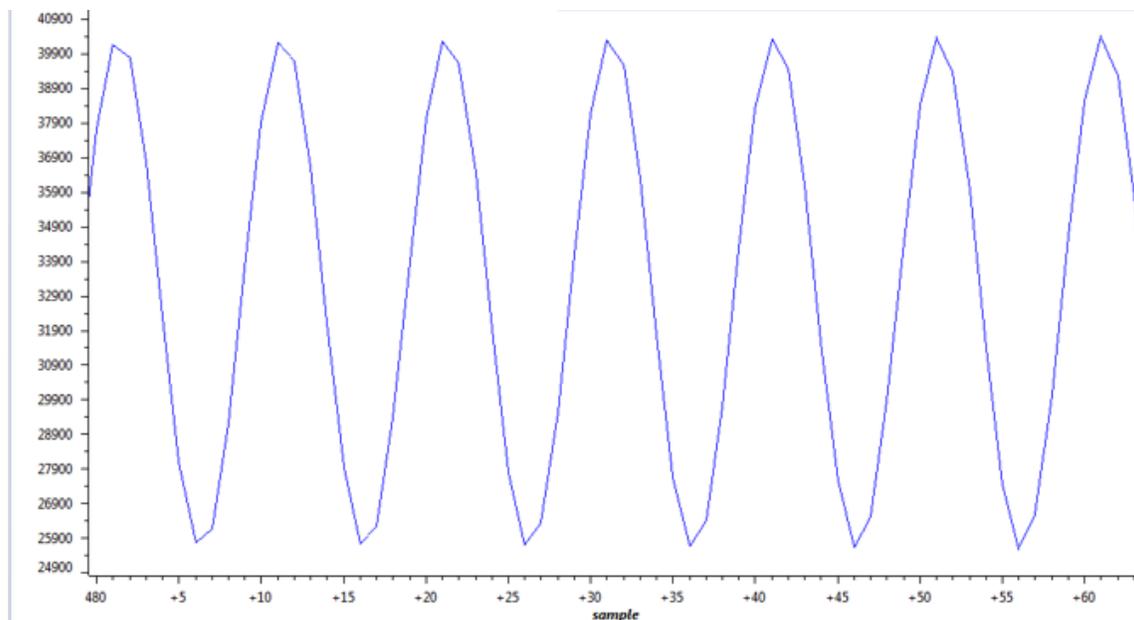


Figure 23. ADC Output (1 KHz) of Sampled SPI Data

### 7.1.2 500-Hz Sine Wave on ADC Input

Figure 22 shows the SPI communication and the sine wave (ADC Input) of 500 KHz at 2.6 V.

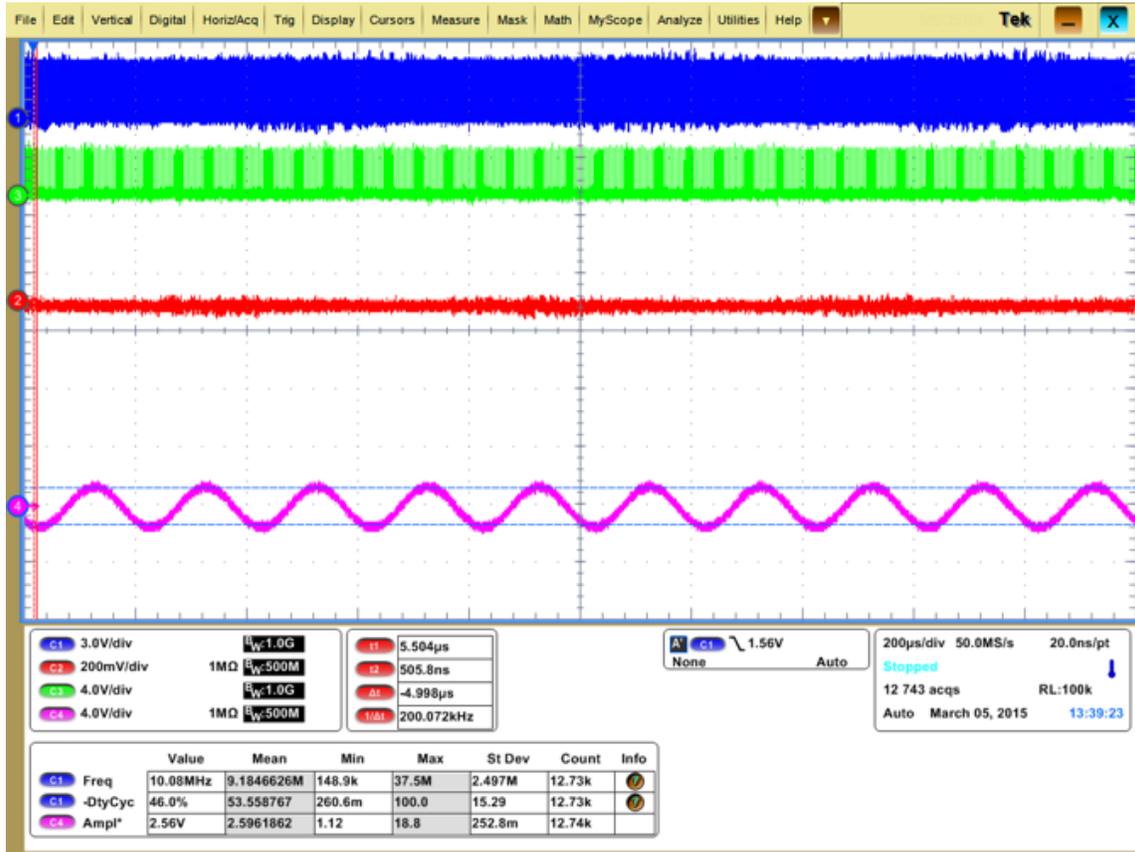


Figure 24. Oscilloscope Capture of ADC Input (500 Hz) and SPI Communication

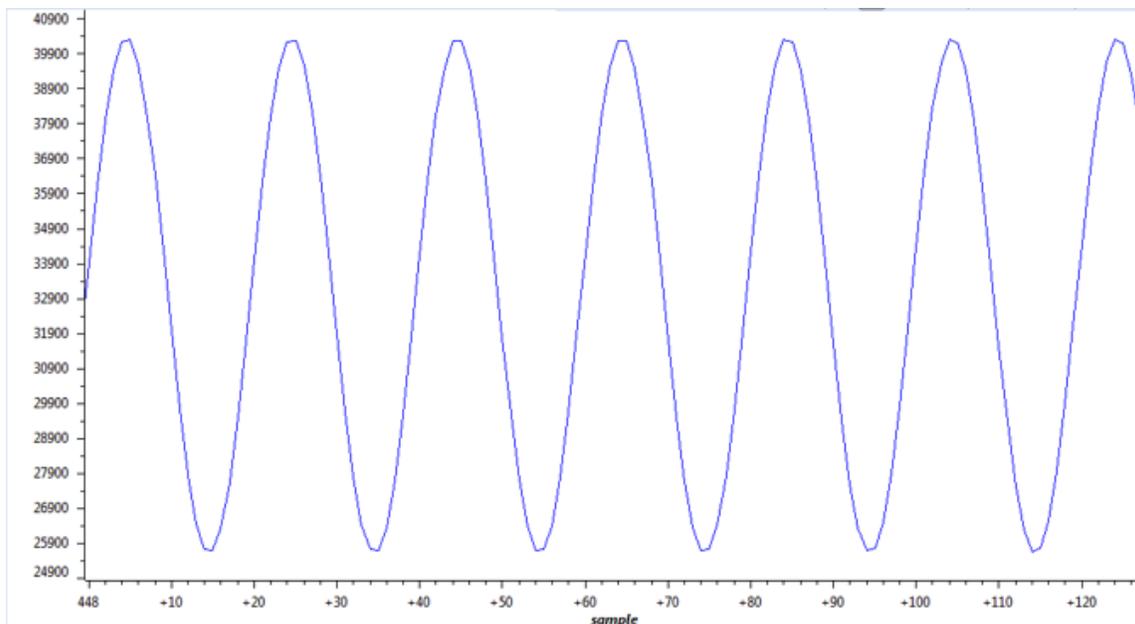


Figure 25. Oscilloscope Capture of ADC Input (500 Hz) and SPI Communication

## 8 Design Files

### 8.1 Schematics

To download the bill of materials (BOM), see the design files at [TMDXAM437X](#).

### 8.2 Bill of Materials

To download the bill of materials (BOM), see the design files at [TMDXIDK437X](#).

### 8.3 PCB Layout

To download the PCB layout, see the design files at [TMDXIDK437X](#).

### 8.4 Gerber Files

To download the Gerber files, see the design files at [TMDXIDK437X](#).

### 8.5 Software Files

To download the software files, see the design files at [TIDEP0033](#).

### 8.6 Physical AM437X IDK EVM

Purchase the AM437X IDK TMDXIDK437X EVM from the TI online shop. Please refer to: <http://www.ti.com/tool/tmdxidk437x>.

## 9 References

1. Texas Instruments, *AM437x Sitara™ Processors*, AM437x Data Sheet ([SPRS851](#))
2. Texas Instruments, *AM437x ARM® Cortex™-A9 Processors*, AM437x Technical Reference Manual ([SPRUHL7](#))
3. Texas Instruments, *16-Bit, 8-Channel, Software Configurable Analog Input Module for Programmable Logic Controllers (PLCs)*, TIDA-00614 User's Guide ([TIDU365](#))
4. Texas Instruments, *ADS868x 16-Bit, 500-kSPS, 4- and 8-Channel, Single-Supply, SAR ADCs with Bipolar Input Ranges*, ADS8688 Data Sheet ([SBAS582](#))

## 10 About the Author

**THOMAS MAUER** is a System Applications Engineer in the Factory Automation and Control Team at Texas Instruments Freising, where he is responsible for developing reference design solutions for the industrial segment. Thomas brings to this role his extensive experience in industrial communications like Industrial Ethernet and fieldbuses and industrial applications. Thomas earned his Electrical Engineering degree (Dipl. Ing. (FH)) at the University of Applied Sciences in Wiesbaden, Germany.

**KARTHIK BANDI** is a Master student at ESIGELEC, Rouen, France in Electronic Embedded Systems. As a part of his studies, he is a Master Intern in the Factory Automation and Control Team at Texas Instruments in Freising, Germany. He is developing solutions for reference design in the industrial area. Karthik Bandi did his Bachelor's degree in Electronics and Communication Engineering from JNTU-Hyderabad, India.

## IMPORTANT NOTICE FOR TI REFERENCE DESIGNS

Texas Instruments Incorporated ("TI") reference designs are solely intended to assist designers ("Buyers") who are developing systems that incorporate TI semiconductor products (also referred to herein as "components"). Buyer understands and agrees that Buyer remains responsible for using its independent analysis, evaluation and judgment in designing Buyer's systems and products.

TI reference designs have been created using standard laboratory conditions and engineering practices. **TI has not conducted any testing other than that specifically described in the published documentation for a particular reference design.** TI may make corrections, enhancements, improvements and other changes to its reference designs.

Buyers are authorized to use TI reference designs with the TI component(s) identified in each particular reference design and to modify the reference design in the development of their end products. HOWEVER, NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY THIRD PARTY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT, IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI REFERENCE DESIGNS ARE PROVIDED "AS IS". TI MAKES NO WARRANTIES OR REPRESENTATIONS WITH REGARD TO THE REFERENCE DESIGNS OR USE OF THE REFERENCE DESIGNS, EXPRESS, IMPLIED OR STATUTORY, INCLUDING ACCURACY OR COMPLETENESS. TI DISCLAIMS ANY WARRANTY OF TITLE AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, QUIET ENJOYMENT, QUIET POSSESSION, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS WITH REGARD TO TI REFERENCE DESIGNS OR USE THEREOF. TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY BUYERS AGAINST ANY THIRD PARTY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON A COMBINATION OF COMPONENTS PROVIDED IN A TI REFERENCE DESIGN. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, SPECIAL, INCIDENTAL, CONSEQUENTIAL OR INDIRECT DAMAGES, HOWEVER CAUSED, ON ANY THEORY OF LIABILITY AND WHETHER OR NOT TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, ARISING IN ANY WAY OUT OF TI REFERENCE DESIGNS OR BUYER'S USE OF TI REFERENCE DESIGNS.

TI reserves the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques for TI components are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

Reproduction of significant portions of TI information in TI data books, data sheets or reference designs is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards that anticipate dangerous failures, monitor failures and their consequences, lessen the likelihood of dangerous failures and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in Buyer's safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed an agreement specifically governing such use.

Only those TI components that TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components that have **not** been so designated is solely at Buyer's risk, and Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.