DCA1000EVM Data Capture Card

Contents

List of Figures

List of Tables

Trademarks

MicroSD is a trademark of SD-3C, LLC.

1 Introduction

The DCA1000EVM is a capture card for interfacing with Texas Instrument's 77GHz xWR1xxx EVM that enables users to stream the ADC data over Ethernet. This design is based on Lattice FPGA LFE5UM-85F-8BG381I with DDR3L. The signal interface between the capture card and the xWR1xxx EVM uses the 60-pin Samtec high density connector.

The following use cases can be accessed using the DCA1000EVM with TI's xWR1xxx EVM.

- LVDS over Ethernet streaming:
  - Raw mode: In this mode, all LVDS data is captured as it is and streamed over the Ethernet interface.
  - Data separated mode: In this mode, the user can add specific headers to different data types; FPGA separates out different data types based on the header and streams it over Ethernet interface.

1.1 Key Features

The key features of the DCA1000EVM are:

- The DCA1000EVM is based on LATTICE's High-Performance FPGA LFE5UM-85F-8BG381I
  - Package: 381 pins CABGA (17×17 mm).
  - 84K LUT's, 3744 Kbits Block RAM and 4 PLLs
- 2GByte of DDR3L device
- 16MByte SPI Flash for storing FPGA images
- MicroSD™ Card interface (presently not supported)
- JTAG interface through FTDI FT2232HL Chip
- USB interface to provide SPI, UART, and I2C interfaces through the onboard FTDI device. This is used for mmWave Studio.
- 9 user LEDs, 4 push-button switches, and 1×8 and 1×3 DIP switches for user configurations
- 1Gb/s RGMII Ethernet interface with RJ45 jack
- TI's xWR1xxx EVM provides the below mentioned interfaces on the 60-pin SAMTEC connector QTH-030--01-L-D-A. The connector QTH-030-01-L-D-A can be mated with TI's xWR1xxx EVM using board-to-board cable assembly (Samtec coax micro ribbon cable - Part Number# HQCD-030-02.00-SEU-TBR-1, is included in the kit).
  - SPI interface
  - UART interface
  - I2C interface
  - 4-Lane LVDS interface
  - GPIO signals
  - DMM interface
- The DCA1000EVM is powered up using 5-V power, either from an external DC adapter, or from TI's xWR1xxx EVM, by selecting the power selection switch (SW3).
- DC-DC regulators and LDO regulators are used to derive the required power required for FPGA, DDR3L, and other onboard peripherals.
- RoHS, Pb-free, and REACH-compliant design
- LVDS data rates support a maximum of 600 Mbps. It supports both 4 lanes (xWR1243/1443) and 2 lanes (xWR1642)
- Temperature range – industrial grade
- Maximum current rating – 700-mA maximum

1.2 Kit Contents

The following items are included with the DCA1000EVM kit. Unbox the board and identify various components and connectors as detailed in the user guide.

- DCA1000EVM Kit
- Ethernet cable
- Samtec coax micro ribbon cable
- 8GB microSD Card
- Spacers, screws, and washers

2 Hardware Specifications

2.1 Functional Block Diagram

The functional block diagram of the DCA1000EVM is shown in Figure 1.

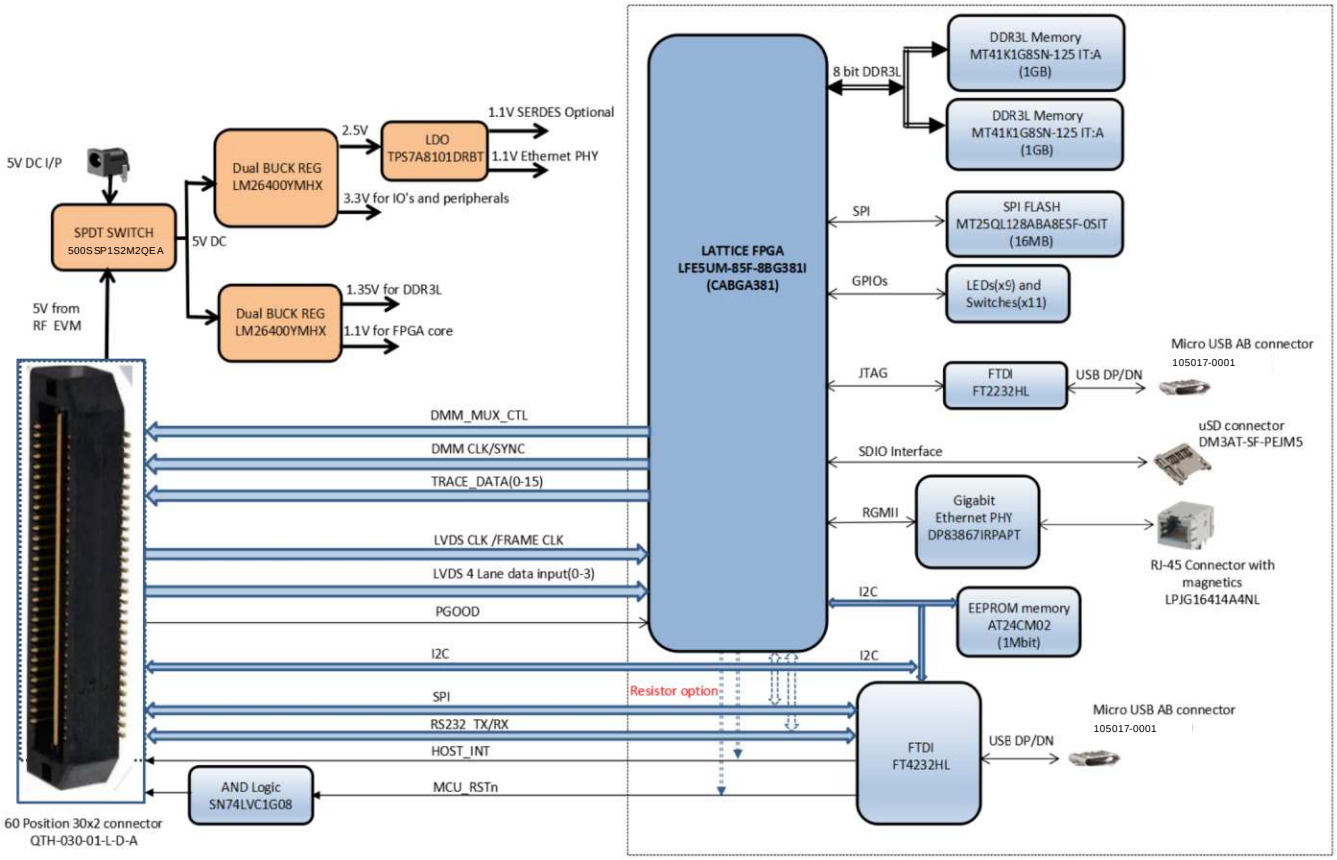


Figure 1. DCA1000EVM Functional Block Diagram

2.2 Board Dimensions

The DCA1000EVM dimensions are 90 mm × 102 mm × 1.6 mm. It is a 10-layer board fabricated with epoxy fiberglass FR4 grade material (UL94V-0 Certified).

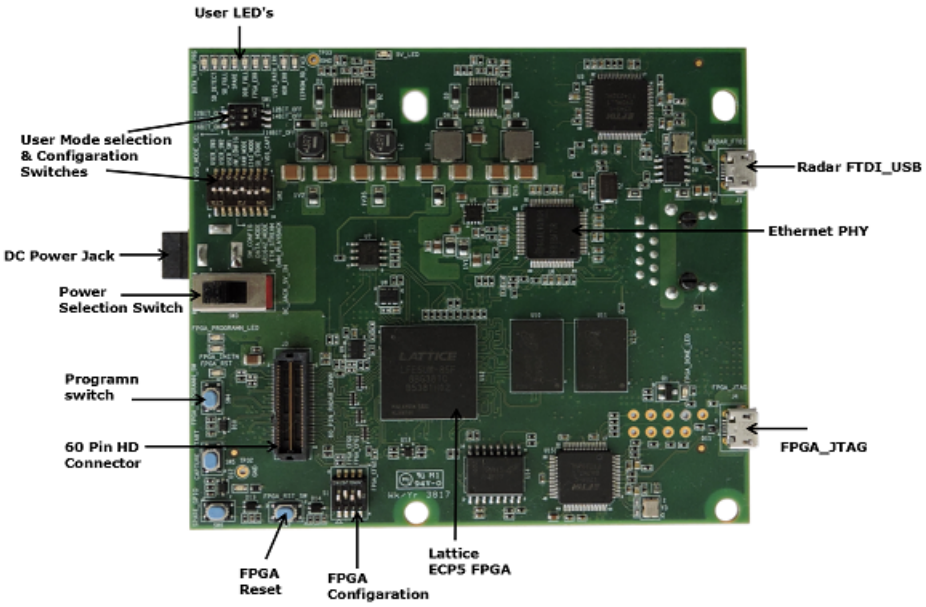


Figure 2. DCA1000EVM – Top View

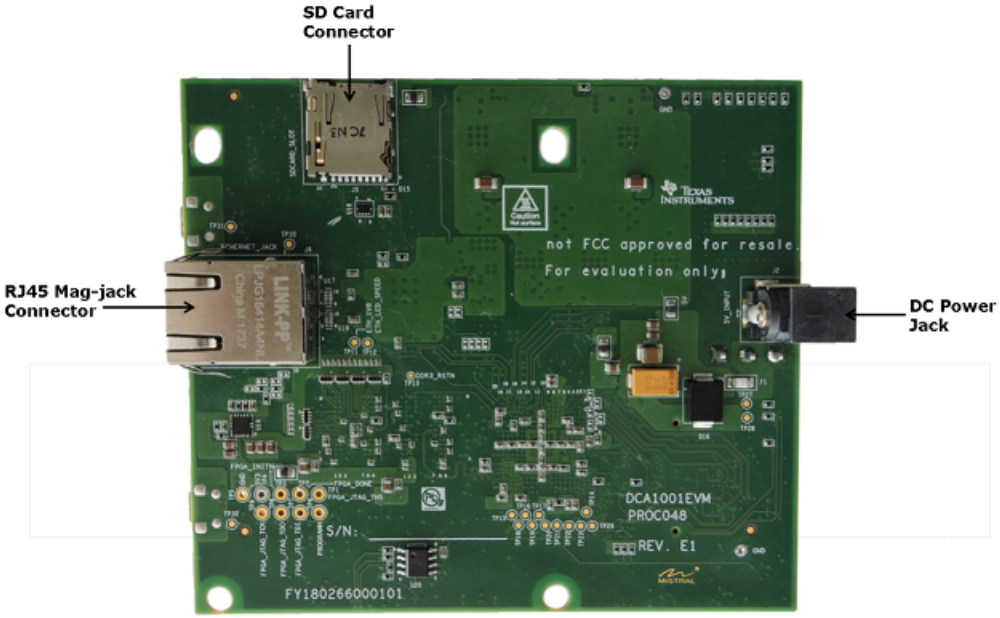


Figure 3. DCA1000EVM – Bottom View

2.3 DCA1000EVM and xWR1xxx EVM – Board to Board Connection

To enable easy measurements on the sensing objects on the horizontal plane, the xWR1xxx EVM (AWR1243BOOST/AWR1443BOOST/AWR1642BOOST) and DCA1000EVM can be mounted vertically by fixing the L-brackets. The following steps illustrate the assembly of the L-brackets, DCA1000EVM, and xWR1xxx EVM.

1. Align the DCA1000EVM and xWR1xxx EVM boards, as shown in Figure 4.

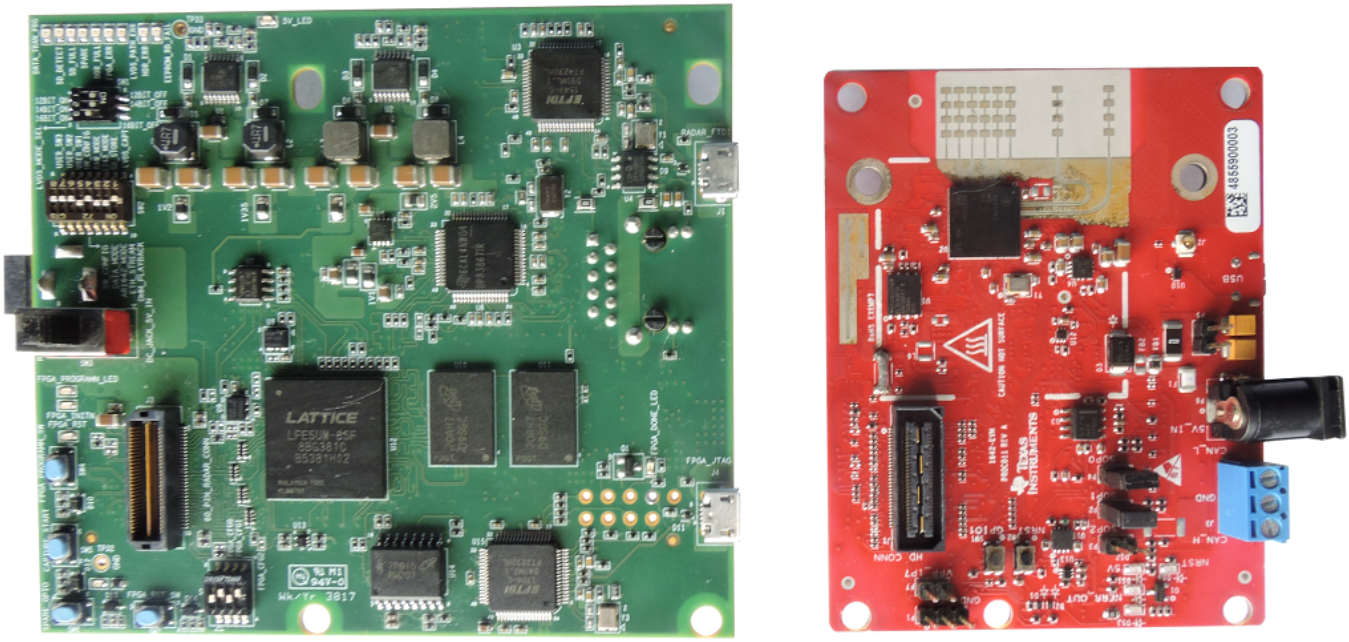


Figure 4. DCA1000EVM and xWR1xxx EVM Board Alignment

2. Connect the DCA1000EVM with screws, spacers, and stands, as shown in Figure 5.

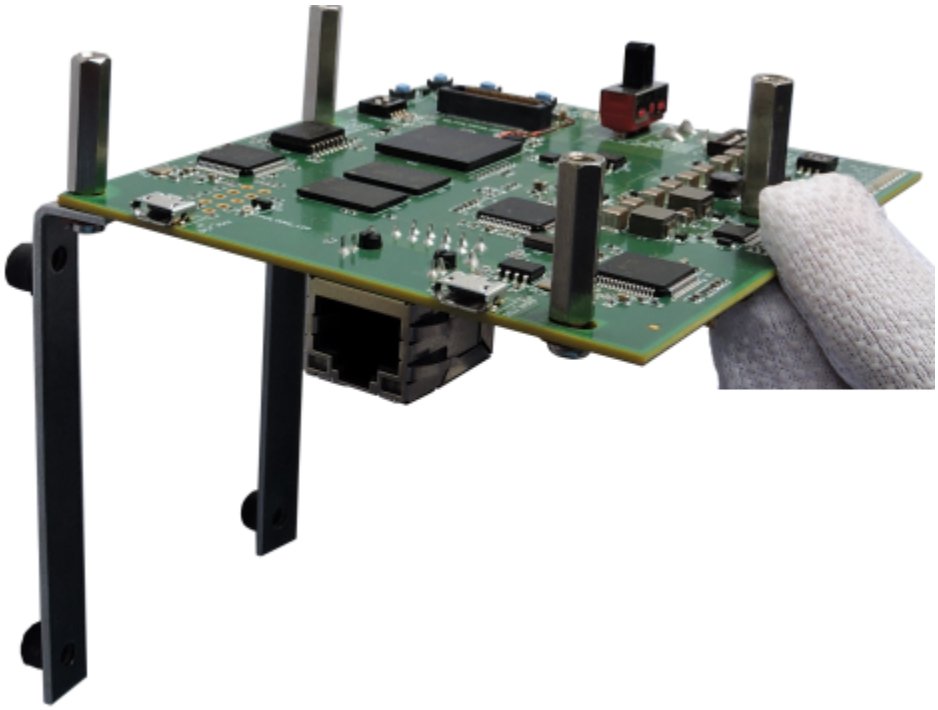


Figure 5. Connect Screws and Spacers

3. Connect the DCA1000EVM with the xWR1xxx EVM, as shown in Figure 6.

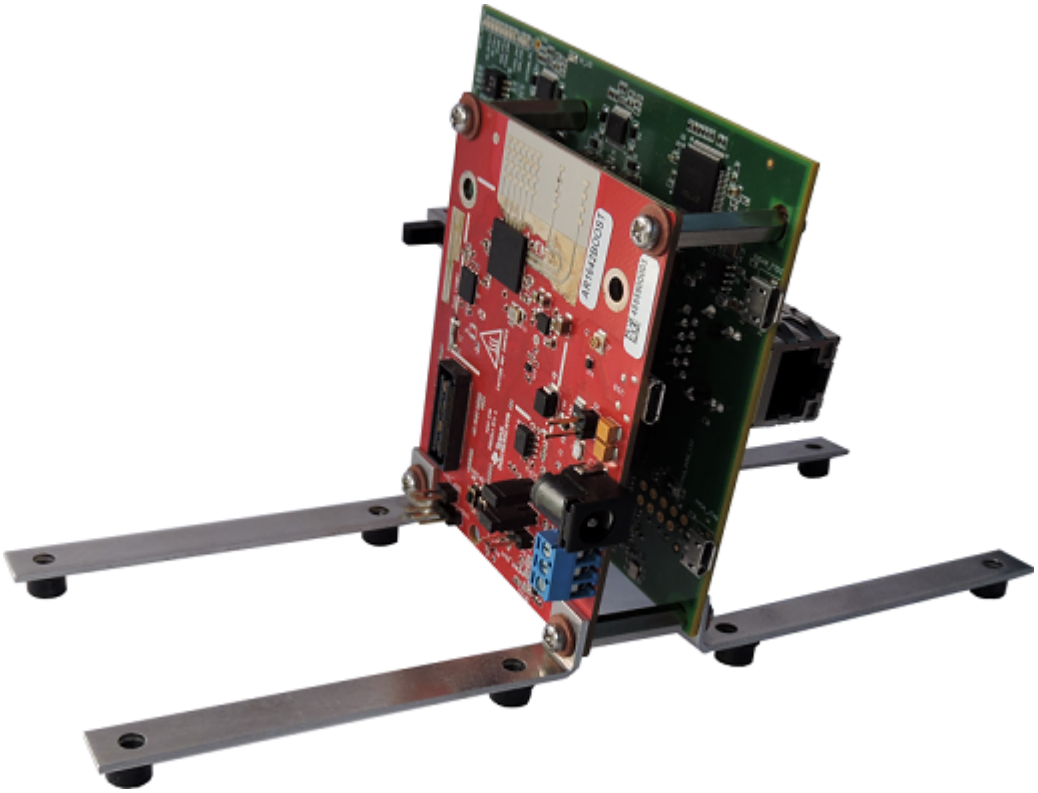


Figure 6. Connect Boards

4. Connect the Samtec ribbon cable, as shown in Figure 7 and Figure 8.

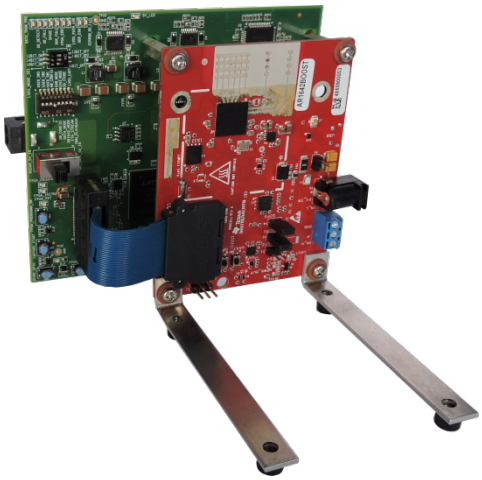


Figure 7. Connect Ribbon Cable

Figure 8. Connect Ribbon Cable (Side)

2.4 Power Connections

The DCA1000EVM is powered by 5-V power, either from a DC jack as shown in Figure 9, or from a 60-pin high density connector of TI's xWR1xxx EVM mated to J3 of DCA1000EVM, as shown in Figure 14. The power source is selected through the SPDT power switch (Figure 10) position, as shown in Table 1. When the power is provided, the 5-V LED (LD10) glows, indicating that the board is powered on.

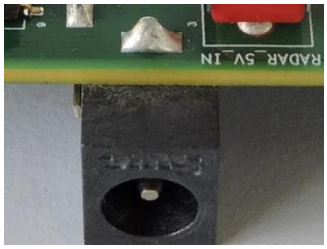


Figure 9. Power Connector

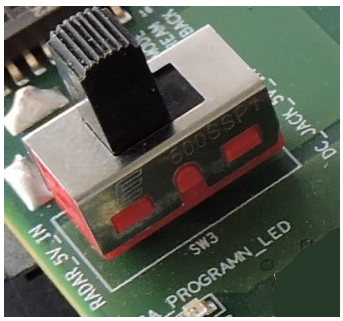


Figure 10. Power Selection Switch

Table 1. Power Selection Switch Information

Reference	Usage	Description
SW3.1	Power selection switch	When the switch is closed to position 1, then the external 5-V power is given through a DC jack.
SW3.3	Power selection switch	When the switch is closed to position 3, then the 5-V power is derived from the xWR1xxx EVM through the 60-pin connectors. Note: To get 5-V power from the xWR1xxx EVM, the R34 (0 Ω) resistor should be mounted in the xWR1xxx EVM.

2.5 Connectors

2.5.1 USB Connectors

The DCA1000EVM has two standard micro USB connectors. Connector J1 allows access to the xWR1xxx EVM through UART, SPI, or I2C through the FTDI chip, using the TI RADAR Studio application, and is as shown in Figure 11. The mmWave Studio PC application can be used to configure the radar device. For more details on the mmWave Studio, refer to the following link: http://www.ti.com/tool/MMWAVE-STUDIO. The second micro USB connector J4 allows programming the FPGA through the FTDI chip through the JTAG interface, and is shown in Figure 12.

Table 2 provides the J1 connector pin information. Table 3 provides the J4 connector pin information.

To detect the RADAR FTDI and emulate the SPI host bus, connect a micro USB cable to J1 on the DCA1000EVM.

Table 2. J1 USB Connector Information

Pin No	Signal Name	Description
1	TP	Test Point
2	FTDI_USB1_DM	USB Data Signal Negative
3	FTDI_USB1_DP	USB Data Signal Positive
4	GND	Ground
5	GND	Ground

Figure 11. J1 USB Connector

To configure the FPGA through JTAG, connect a micro USB cable to J4 on the DCA1000EVM.

Table 3. J4 USB Connector Information

Pin No	Signal Name	Description
1	TP	Test Point
2	FTDI_USB2_DM	USB Data Signal Negative
3	FTDI_USB2_DP	USB Data Signal Positive
4	GND	Ground
5	GND	Ground

Figure 12. J4 USB Connector

2.5.2 Ethernet Jack

The DCA1000EVM supports a Gigabit Ethernet port to provide the connection to the network. The Ethernet port is interfaced to the LATTICE FPGA through the Ethernet PHY DP83867IRPAPT, and is used to stream the captured data over the network to the host PC. Figure 13 shows the Ethernet RJ45 Mag-Jack connector, and Table 4 provides the connector pin details.

Table 4. J6 Ethernet Jack Information

Pin No	Signal Name	Description
1	GND	Ground
2	TP	Test point
3	ETH_D3P	Gigabit Ethernet differential pair 3 positive
4	ETH_D3N	Gigabit Ethernet differential pair 3 negative
5	ETH_D2P	Gigabit Ethernet differential pair 2 positive
6	ETH_D2N	Gigabit Ethernet differential pair 2 negative
7	ETH_D1P	Gigabit Ethernet differential pair 1 positive
8	ETH_D1N	Gigabit Ethernet differential pair 1 negative
9	ETH_D0P	Gigabit Ethernet differential pair 0 positive
10	ETH_D0N	Gigabit Ethernet differential pair 0 negative
11	3V3	Supply voltage for internal anode of activity LED
12	LED_ACTn	Activity LED
13	LED_LINKn	Link LED
14	3V3	Supply voltage for internal anode of link LED

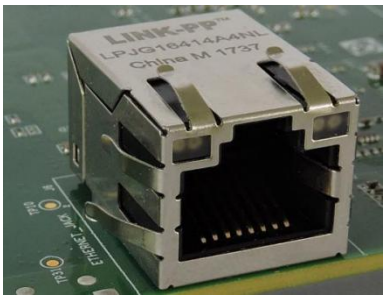


Figure 13. J6 Ethernet Jack

2.5.3 60-Pin High Density (HD) Connector

The DCA1000EVM supports a 60-pin HD connector to interface with the xWR1xxx EVM. This connector provides the high-speed LVDS data, control signals (UART, SPI, I2C, RESET), and DMM interface signals. The DCA1000EVM captures data from the xWR1xxx EVM over a 4-lane LVDS interface and playbacks the data from the host PC to the xWR1xxx EVM over a 16-bit DMM/trace interface through the 60-pin connector. Control signals on the 60-pin HD connector are used to access the xWR1xxx EVM through the FTDI chip. Figure 14 shows the HD connector, and Table 5 provides the connector pin details.

Table 5. J3 High Density Interface Connector Information

Pin No	Signal Name	Description
1	VCC5V_CONN	5-V supply voltage from xWR1xxx EVM
2	VCC5V_CONN	5-V supply voltage from xWR1xxx EVM
3	VCC5V_CONN	5-V supply voltage from xWR1xxx EVM
4	NC	No connect
5	NC	No connect
6	NC	No connect
7	Conn_SPI_CSn	SPI chip select
8	NC	No connect
9	Conn_SPI_CLK	SPI clock
10	HOST_INT	Host interrupt from FTDI
11	Conn_SPI_MOSI	SPI master output slave input
12	Conn_SPI_MISO	SPI master input slave output
13	PGOOD	PGOOD input from xWR1xxx EVM
14	NC	No connect
15	DMM_CLK	DMM clock
16	DMM_MUX_CTL	DMM control
17	DMM_SYNC	DMM synchronization
18	GND	Ground
19	DP0	DMM Data0
20	LVDS_VALIDP	LVDS valid signal positive
21	DP1	DMM Data1
22	LVDS_VALIDM	LVDS valid signal negative
23	DP2	DMM Data2
24	GND	Ground
25	DP3	DMM Data3
26	LVDS_FRCLKP	LVDS frame clock signal positive
27	DP4	DMM Data4
28	LVDS_FRCLKN	LVDS frame clock signal negative

Table 5. J3 High Density Interface Connector Information (continued)

Pin No	Signal Name	Description
29	DP5	DMM Data5
30	GND	Ground
31	DP6	DMM Data6
32	LVDS_3P	LVDS data pair 3 positive
33	DP7	DMM Data7
34	LVDS_3M	LVDS data pair 3 negative
35	DP8	DMM Data8
36	GND	Ground
37	DP9	DMM Data9
38	LVDS_2P	LVDS data pair 2 positive
39	DP10	DMM Data10
40	LVDS_2M	LVDS data pair 2 negative
41	DP11	DMM Data11
42	GND	Ground
43	DP12	DMM Data12
44	LVDS_CLKP	LVDS clock pair positive
45	DP13	DMM Data13
46	LVDS_CLKM	LVDS clock pair negative
47	DP14	DMM Data14
48	GND	Ground
49	DP15	DMM Data15
50	LVDS_1P	LVDS data pair 1 positive
51	SDA	Serial Data
52	LVDS_1M	LVDS data pair 1 negative
53	SCL	Serial clock
54	GND	Ground
55	RS232RX	UART data receive
56	LVDS_0P	LVDS data pair 0 positive
57	RS232TX	UART data transmit
58	LVDS_0M	LVDS data pair 0 negative
59	MCU_RSTn	Reset input to xWR1xxx EVM
60	GND	Ground

Figure 14. J3 High Density Interface Connector

2.6 Switches, Push Buttons, and LEDs

2.6.1 Switches

The DCA1000EVM supports a 1×3 DIP switch (as explained in Table 7) and a 1×8 DIP switch (as explained in Table 8) for mode selection and other functions. In switch SW2, Pin5 indicates that the DCA1000EVM is configured either by a hardware switch-based configuration, or by a software-based configuration, as listed in Table 6.

Table 6. Switch2.5 Functionality Information

Reference	Usage	Description
SW2.5	CONFIG_VIA_HW	When positioned at 5 (Pin5), the DCA1000EVM is configured as per the settings of the hardware DIP switch SW2. In hardware configuration mode, software configurations are ignored
	CONFIG_VIA_SW	When positioned at the other side (Pin12), the DCA1000EVM is configured through the software configuration commands over Ethernet. In software configuration mode, hardware DIP settings are ignored.

When the DCA1000EVM is configured in hardware switch configuration mode, use the following switch settings for different modes of data capture.

Table 7. Switch1 Functionality Information

LVDS bit Setting Switch	SW1 (3 Position Switch)			Description
	1	2	3	
LVDS 12-bit mode	ON	OFF	OFF	LVDS 12 bit data is captured from the xWR1xxx EVM.
LVDS 14-bit mode	OFF	ON	OFF	LVDS 14 bit data is captured from the xWR1xxx EVM.
LVDS 16-bit mode	OFF	OFF	ON	LVDS 16 bit data is captured from the xWR1xxx EVM.

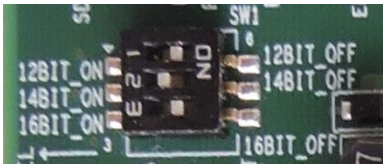


Figure 15. Mode Configuration_1

Table 8. Switch2 Functionality Information

Reference	Usage	Description
SW2.1	LVDS_CAPTURE	When positioned at 1(Pin1), the DCA1000EVM is in capture mode. In this mode, data is received over LVDS and streamed through Ethernet to the PC.
	DMM_PLAYBACK	When positioned at the other side (Pin16), the DCA1000EVM is in play back mode.
SW2.2	SD_STORE	When positioned at 2 (Pin2), the DCA1000EVM is in capture mode, and data is saved into the SD card.
	ETH_STREAM	When positioned at the other side (Pin15), the DCA1000EVM is in capture mode, and data is streamed over the network to the host PC.
SW2.3	1243_MODE	When positioned at 3 (Pin3), the DCA1000EVM captures 4-lane LVDS data.
	AR1642_MODE	When positioned at the other side (Pin14), the DCA1000EVM captures 2-lane LVDS data.
SW2.4	RAW_MODE	When positioned at 4 (Pin4), the DCA1000EVM is in raw mode. In this mode, all LVDS data is captured as it is.
	DATA_MODE	When positioned at the other side (Pin13), the DCA1000EVM is in data separated mode. In this mode, the user can add specific headers to different data types, and FPGA separates out different data types based on the header.

Table 8. Switch2 Functionality Information (continued)

Reference	Usage	Description
SW2.6	USER_SW1 (Using EEPROM configuration)	When positioned at 6 (Pin6), the Ethernet configuration data is read from EEPROM, and the same is used for Ethernet communication.
	GND (Using FPGA configuration)	When positioned at the other side (Pin11), the Ethernet configuration data hardcoded in FPGA is used for Ethernet communication.
SW2.7	USER_SW2	Future use
	GND	
SW2.8	USER_SW3	Future use
	GND	

Figure 16. Mode Configuration_2

2.6.2 Push Buttons

The DCA1000EVM supports four push buttons for FPGA reset, data capture start, and other basic functions listed in Table 9.

Table 9. Push Button Functionality Information

Reference	Usage	Description
SW4	PROGRAMN	Whenever SPI flash is re-flashed or reprogrammed, press the PROGAMN (SW4) button to program the FPGA from SPI flash. Note: A binary file should be present in the SPI flash.
SW5	Capture start	When the DCA1000EVM is in hardware configuration mode, press the capture start (SW5) button to initiate the data transfer. Press the button (SW5) again to stop the data transfer.
SW6	SPARE_GPIO_SW	Future use
SW7	FPGA_RST	Press the FPGA_RST button to reset the FPGA.

Figure 17. Push Button

2.6.3 LEDs

The DCA1000EVM supports LEDs for user indications, as listed in Table 10.

Table 10. LED Functionality Information

Reference	Usage	Description	Color
LD1	DATA_TRANS_PROG_LED1	This LED indicates the progress of data transfer, either through the network or saving into the SD card.	Green
LD2	SD_DETECT_LED5	When the SD card is mounted into the DCA1000EVM, this LED indicates whether the SD card is detected or not.	Green
LD3	SD_FULL_LED0	This LED indicates the full condition of the SD card.	Red
LD4	SPARE_LED8	Future use	Green
LD5	FPGA_ERR_LED2	When LVDS data is coming at high rate, the internal FPGA buffer gets full. In that case, this LED glows.	Red
LD6	DDR_FULL_LED7	This LED indicates the full condition of the 2GB DDR3 memory.	Red
LD7	LVDS_PATH_ERR_LED3	This LED indicates whether the xWR1xxx EVM is sending LVDS data or not.	Red
LD8	HEADER_ERR_LED4	This LED indicates the failure of the header when the DCA1000EVM is in data separated mode.	Red
LD9	EEPROM_RD_FAIL_LED6	This LED indicates an EEPROM read access failure.	Red
LD10	5V	This LED indicates the presence of the 5-V supply in the DCA1000EVM.	Green
LD11	PROGRAMN	This LED indicates assertion of the program to FPGA when the PROGRAMN button is pressed.	Red
LD12	INITN	This LED indicates that the FPGA is ready to be configured.	Red
LD13	FPGA_RST	This LED indicates the assertion of the FPGA reset.	Red
LD14	DONE	When FPGA is programmed from SPI flash, this LED glows.	Green
LD15	FTDI_JTAG_ACT	This LED indicates activity on the FPGA JTAG connectivity.	Green

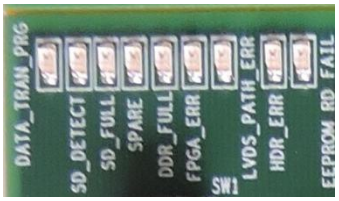


Figure 18. User LEDs

3 System Setup

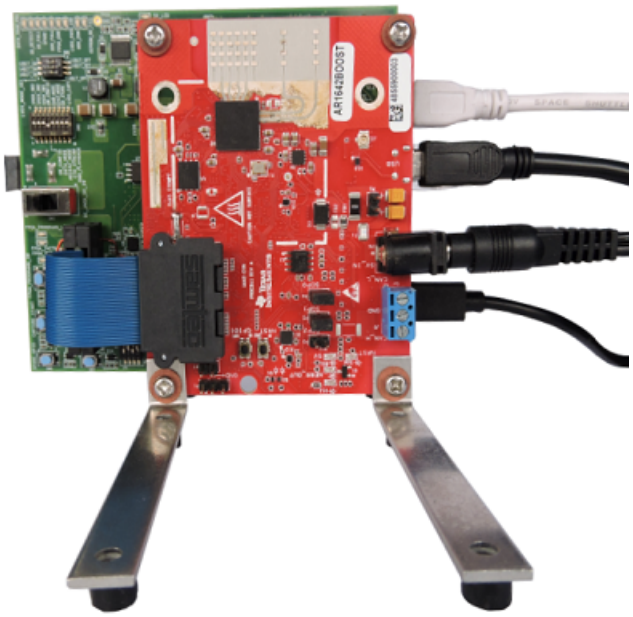


Figure 19. Board Setup

3.1 Prerequisites

- The DCA1000EVM should be connected to the host PC through an Ethernet cable for the data transfer process.
- The DCA1000EVM should be connected to a PC through a USB cable (J1-Radar FTDI) for configuring the xWR1xxx EVM if the mmWave Studio is used to configure the radar device. If an embedded application is used to configure the xWR1xxx EVM, then this is not required.
- The DCA1000EVM should be connected to TI's xWR1xxx EVM through a 60-pin HD connector by using a 60-pin Samtec ribbon cable.
- The DCA1000EVM power input should be connected from either a DC jack or TI's xWR1xxx EVM power output (from a 60-pin HD connector) by selecting the SW3 switch position.
- The xWR1xxx EVM should be connected to a PC through a USB cable (J8) for the RS232 connection in the Radar Studio interface.
- HOST PC configuration:
  - Intel Core i5 2.5 GHz or above
  - 8 GB RAM or above
  - Windows 10 Pro operating system
  - Static IP Address – 192.168.33.30
  - 1-Gbps Ethernet port
- Pre-installed mmWave Studio in the mmWave Studio user guide..
- The DCA1000EVM should be connected to a PC through a USB cable (J4-FPGA JTAG) for programming the FPGA (when the card is in programming mode).

4 DCA1000EVM Ethernet Configuration Data

The DCA1000EVM stores the default Ethernet configuration data (such as DCA1000EVM IP, DCA1000EVM MAC, System IP, Configuration port, and Data port) in FPGA internal registers.

The default Ethernet configuration data stored in the FPGA internal registers are listed in Table 11.

Table 11. Table 11 Default Value Stored in FPGA Registers

Parameter	Default Value (Decimal)
FPGA IP	192.168.33.180
FPGA MAC	12-34-56-78-90-12
System IP	192.168.33.30
Configuration port	4096
ADC/CP_ADC/ADC_CP/CP_ADC_CQ data	4098
CP data	4099
CQ data	4100
R4F debug data	4101
DSP debug data	4102

The DCA1000EVM has option of loading Ethernet configuration data from EEPROM or the FPGA internal register, based on SW2.6.

The Ethernet configuration data is loaded from the EEPROM when SW2 positioned at 6 (pin6); else, FPGA register values are loaded.

Loading the Ethernet configuration data from EEPROM:

- When SW2 is positioned at 6 (pin6), the DCA1000EVM loads the Ethernet configuration data from EEPROM.
- In general, the EEPROM is blank when the DCA1000EVM comes out of the factory. The EEPROM remains blank until the updated EEPROM command is given to the DCA1000EVM through the Ethernet port.
- When the updated EEPROM command is given to the DCA1000EVM, the Ethernet configuration data are stored into the EEPROM along with the header and footer.
- The DCA1000EVM requires a power cycle/hardware reset to load the updated Ethernet configuration data from the EEPROM.
- During the power ON/after hardware reset, the DCA1000EVM FPGA loads the Ethernet configuration data from the EEPROM and checks for the header and footer. If the header and footer are found, the data loaded from the EEPROM is used for Ethernet communication; else, the FPGA uses the default values stored in the internal registers.
- The Ethernet configuration data updated in the EEPROM should be used in the Ethernet packets which are going as commands to the DCA1000EVM, for further communication. Otherwise, the DCA1000EVM will not respond to the command.

Loading the Ethernet configuration data from the FPGA internal registers:

- When SW2 is positioned at (pin11), the DCA1000EVM loads the default Ethernet configuration data from the FPGA internal registers, and the same is used for Ethernet communication.

NOTE: For data separated mode, the DCA1000EVM FPGA receives only one data port number from the updated EEPROM command. This port is considered as an ADC data port, and other data type's port number is incremented by one.

NOTE: Whenever Switch 2.6 position is changed from one option to another option, the DCA1000EVM requires a power cycle and hardware reset.

5 DCA1000EVM Command and Data Format

The DCA1000EVM follows UDP protocol and supports 14 predefined commands. The configuration and status of the DCA1000EVM are communicated through the configuration port. The data port is used to transfer raw mode/data separated mode data.

5.1 DCA1000EVM Command Format

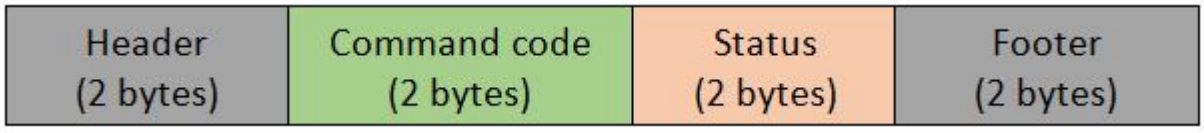
The DCA1000EVM receives a command from the host through the configuration port, and sends back the response in the same port.

The Command Request protocol consists of the following:

- Header
  a. Header
  b. Command code
  c. Data size
- Data
- Footer

The Command Response protocol consists of the following:

- Header
  a. Header
  b. Command code
- Status
- Footer



Commands supported in the DCA1000EVM are listed in Table 12.

Table 12. Supported Commands

Command	Command Code
RESET_FPGA_CMD_CODE	0x01
RESET_AR_DEV_CMD_CODE	0x02
CONFIG_FPGA_GEN_CMD_CODE	0x03
CONFIG_EEPROM_CMD_CODE	0x04
RECORD_START_CMD_CODE	0x05
RECORD_STOP_CMD_CODE	0x06
PLAYBACK_START_CMD_CODE	0x07
PLAYBACK_STOP_CMD_CODE	0x08
SYSTEM_CONNECT_CMD_CODE	0x09
SYSTEM_ERROR_CMD_CODE	0x0A
CONFIG_PACKET_DATA_CMD_CODE	0x0B
CONFIG_DATA_MODE_AR_DEV_CMD_CODE	0x0C

Table 12. Supported Commands (continued)

Command	Command Code
INIT_FPGA_PLAYBACK_CMD_CODE	0x0D
READ_FPGA_VERSION_CMD_CODE	0x0E

Table 13 and Table 14 show examples using RESET_FPGA_CMD_CODE.

Table 13. Reset FPGA Command Request From Host

Name	Data Type	Number of Bytes	Default Value	Min Value	Max Value	Description
Header	UINT16	2	0xA55A	-	-	0xA55A always. Start bits of packet.
Command code	UINT16	2	0x01	-	-	Command code
Size	UINT16	2	0	-	-	Data size
Footer	UINT16	2	0xEEAA	-	-	0xEEAA always. Stop bits of packet.

Table 14. Command Response From the DCA1000EVM

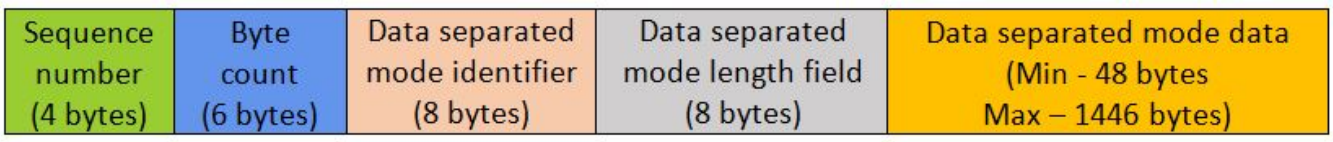
Name	Data Type	Number of Bytes	Default Value	Min Value	Max Value	Description
Header	UINT16	2	0xA55A	-	-	0xA55A always. Start bits of packet.
Command code	UINT16	2	0x01	-	-	Command code
Status	UINT16	2	0	0	1	0 – Success 1 – Failure
Footer	UINT16	2	0xEEAA	-	-	0xEEAA always. Stop bits of packet.

5.2 DCA1000EVM Data Format

The DCA1000EVM sends raw mode/data separated mode data to the host through the data port.

- Raw mode data format: the DCA1000EVM sends raw mode data in the following format.

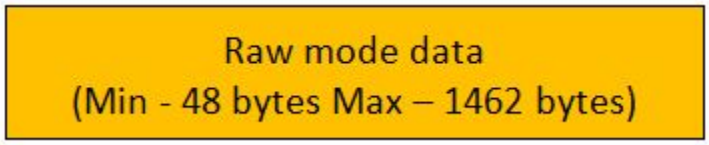
- Data separated mode data format: the DCA1000EVM sends data separated mode data in the following format.



5.3 Stored File Format

Raw mode data format: in raw mode, the file is stored in the following format:

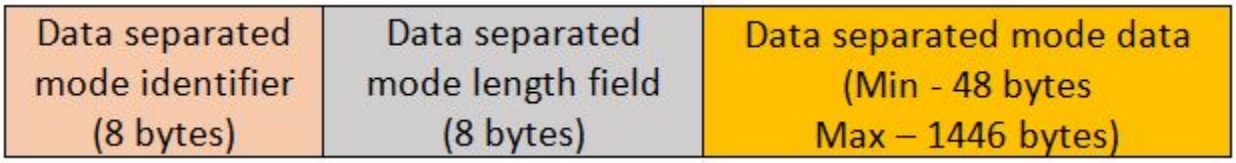
- Without sequence number:



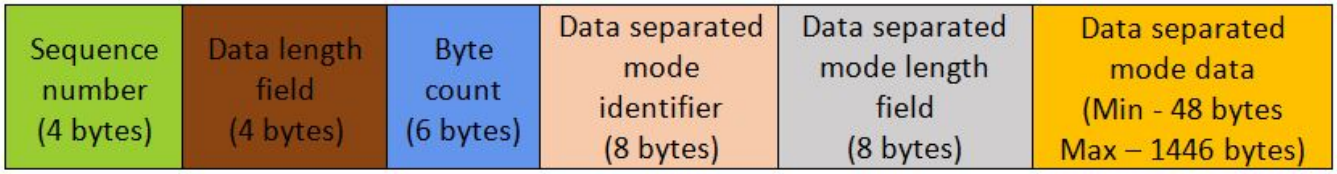
- With sequence number:

Data separated mode data format: in data separated mode, the file is stored in the following format:

- Without sequence number:



- With sequence number:



6 Use Case: LVDS Over Ethernet Streaming

LVDS over Ethernet streaming can be used in two modes of operations: raw mode and data separated mode. Both can be accessed either through the hardware switch configuration mode or software GUI configuration mode by selecting the switch (SW2.5).

6.1 Raw Mode

In Raw mode, the FPGA records all the data appearing on the LVDS lines as is. The data sent over the LVDS lanes is controlled by either the configuration from the mmwave studio, or the user application running on the radar device.

6.2 Data Separated Mode

Data separated mode features supports for receiving different data types with unique identifiers, along with length field information from the xWR1xxx EVM over the LVDS interface, and streamed over the Ethernet interface in different ports. This mode of capture supports specific identifiers, as listed in Table 15. Each data type has a 64-bit identifier and 64-bit length field. In the length field, the actual length information is only 24 bits (the first 12 bits are taken from the first sample and the next 12 bits from the second sample). The remaining 40 bits are reserved. The user application on the radar device must ensure that these headers are added on the LVDS data in the exact format the FPGA expects.

Table 15. Data Separated Mode Data, Identifier, Length Information

Data type	Identifier (64 bits)	Length Field (64 bits)	Actual Data
ADC Data	0x0ADC0CDA0ADC0CDA	Actual length of data is only 24 bits	Valid Data
CP Data	0x0CC909CC0CC909CC		
CQ Data	0x0CC808CC0CC808CC		
R4F Debug Data	0x084F0F48084F0F48		
DSP Debug Data	0x0C67076C0C67076C		
CP_ADC Data	0x0ADD0DDA0ADD0DDA		
ADC_CP Data	0x0ADE0EDA0ADE0EDA		
CP_ADC_CQ Data	0x0ADF0FDA0ADF0FDA		

Each data is streamed in different Ethernet ports through an Ethernet interface, as listed in Table 16.

Table 16. Data Type and Port for Data Separated Mode

Data type	Port
ADC Data/CP_ADC Data/ADC_CP Data/CP_ADC_CQ Data	4098
CP Data	4099
CQ Data	4100
R4F Debug Data	4101
DSP Debug Data	4102

7 Errors

Table 17. Errors

Error indication	Description	Troubleshooting
Out of sequence packet	The host PC might miss some Ethernet packets or receive jumbled packets from the DCA1000EVM.This may be due to either Ethernet fast streaming, UDP protocol, host PC configuration, or the network card (NIC).	Change the Ethernet packet delay options to avoid out of sequence. When increasing Ethernet packet delay, streaming throughput is decreased due to slower data capturing.
FPGA_ERR_LED	This LED indicates an FPGA internal LVDS buffer overflow error. This error occurs when the LVDS data rate is faster than the Ethernet data rate.	Change the periodicity in the radar studio sensorconfig settings to avoid LVDS buffer overflow conditions.
DDR_FULL_LED	This LED indicates a DDR3 full error. This condition occurs when the LVDS data rate is faster than the Ethernet data rate	Change the periodicity in the radar studio sensorconfig settings to avoid DDR3 full conditions.
LVDS_PATH_ERR_LED	This LED indicates NO LVDS DATA received from the radar EVM within a timeout period.	This occurs if the xWR1xxx EVM is not sending LVDS data to the DCA1000EVM. Check whether the radar EVM is configured and enabled to send data. Check that the cabling assembly is correct between the xWR1xxx EVM and the DCA1000EVM.
HEADER_ERR_LED	This LED glows when NO HEADER is found in the LVDS data.	This occurs if the xWR1xxx EVM is not sending LVDS data to the DCA1000EVM. Check whether the radar EVM is configured and enabled to send data separated mode data. Check that the cabling assembly is correct between the xWR1xxx EVM and the DCA1000EVM.
EEPROM_RD_FAIL_LED	This LED indicates EEPROM read failure status. This error can happen if any problem exists with EEPROM hardware connectivity or EEPROM addressing problem.	Power cycle the DCA1000EVM. Check EEPROM connectivity and address lines on hardware.

8 Troubleshooting

- To confirm Ethernet connectivity between the DCA1000EVM and host PC, use the Wireshark tool to check the Ethernet packet transfer.
- Ensure that UDP is enabled in the host PC for UDP data transmission and reception.
- When providing a hardware reset or power cycle on the board, before connecting the mmWave Studio with the DCA1000EVM, check whether the host PC Ethernet connectivity is established or not.
- Ensure that the Samtec cable is properly connected between the xWR1xxx EVM and DCA1000EVM to avoid signal discontinuity.
- The default Ethernet delay is pre-programmed as 25 µs in both the FPGA and software. The user can configure the delay from 5 µs to 500 µs, depending on the host PC configuration and capabilities, to avoid packet loss or packets out of sequence. Lower delay allows for higher bandwidth while increasing the probability of UDP packet drops at the PC end. Refer to Table 19 for the relationship between the packet delay and ethernet throughput.
- When the LVDS data rate is changed in Radar Studio, the DCA1000EVM requires a hardware/software reset to lock the PLL with the incoming clock in the FPGA.
- To update the FPGA image, follow the procedure explained in Section 9.

9 FPGA Programming

The DCA1000EVM supports four DIP switch (S1) options for the FPGA programming configuration, as listed in Table 18.

Table 18. FPGA Configuration Mode

Configuration Mode	S1[4:2]	S1.4	S1.3	S1.2
Master SPI	010	Pin5 (FPGA_CFG2)	Pin6 (FPGA_CFG1)	Pin7 (FPGA_CFG0)

Figure 20. FPGA Config Switch

For FPGA programming, install the Lattice Diamond Programmer standalone tool in the PC, as per the following steps.

1. Download the Lattice Diamond programmer tool, Windows version (Programmer Standalone 3.10 64-bit for Windows) from the following link: http://www.latticesemi.com/programmer.
2. A new folder named lscc will be created at the path of installation.

Post installation steps:

1. Connect the USB cable to the FPGA JTAG USB port (J4), and power on the board.
2. The FTDI cable driver is installed automatically. Otherwise, manually select the cable driver from the installation path.
3. The device manager will recognize two COM ports, as shown in Figure 21.

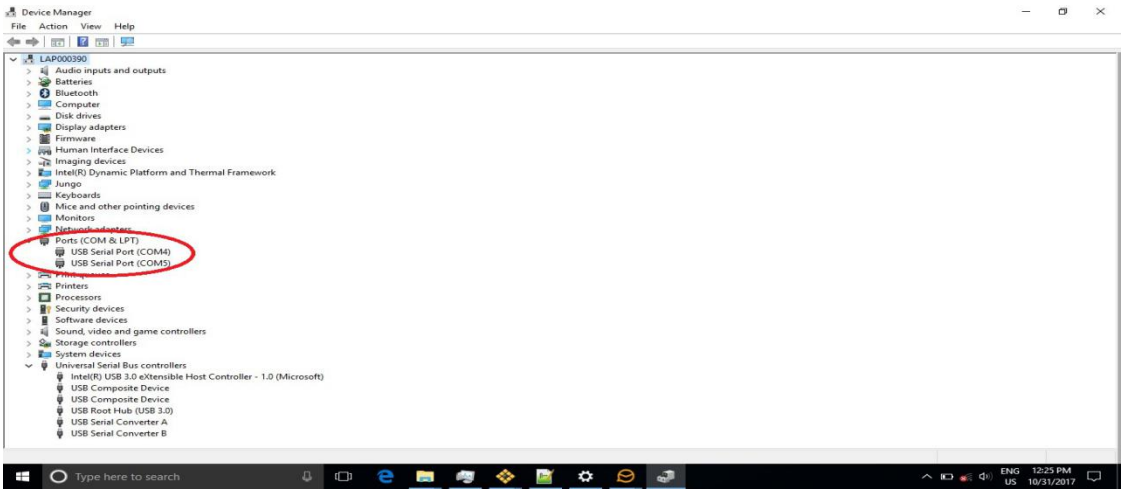


Figure 21. Device Manager

9.1 Hardware Setup

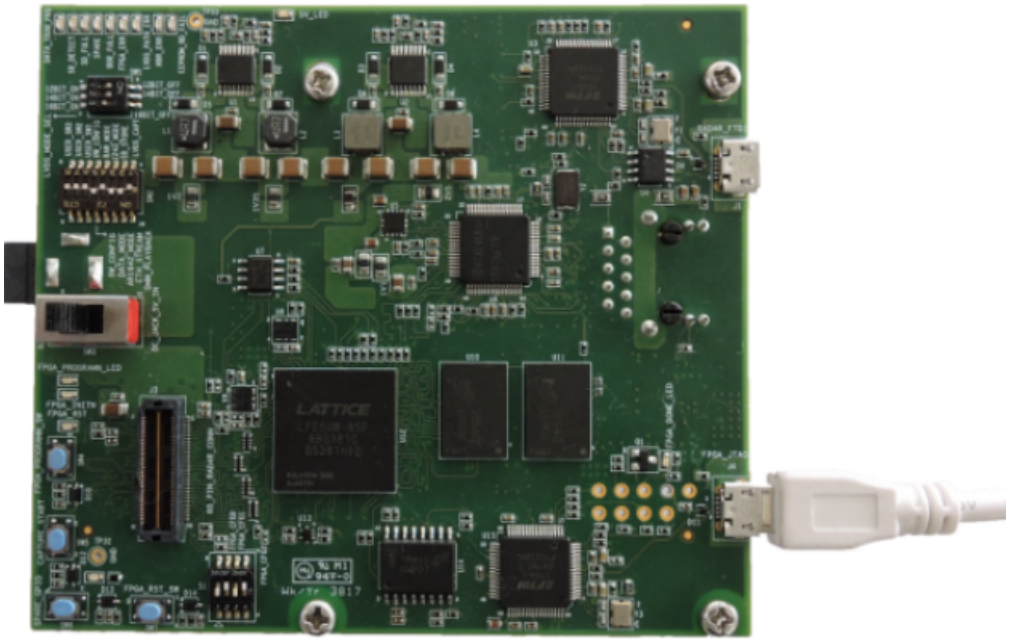


Figure 22. Hardware Setup

9.2 FPGA - SPI Flash Programming Mode

The FPGA image can be flashed in the onboard SPI flash once. This image is loaded to the FPGA each time the FPGA boots up. By default, the FPGA image for data capture would already be flashed in DCA1000EVM. The user must only perform this step if they need to update to a newer version of image, or re-flash the image due to some issues.

1. Openthe Lattice Diamond programmer tool from the Start menu and select Detect Cable. Click OK as shown in Figure 23.

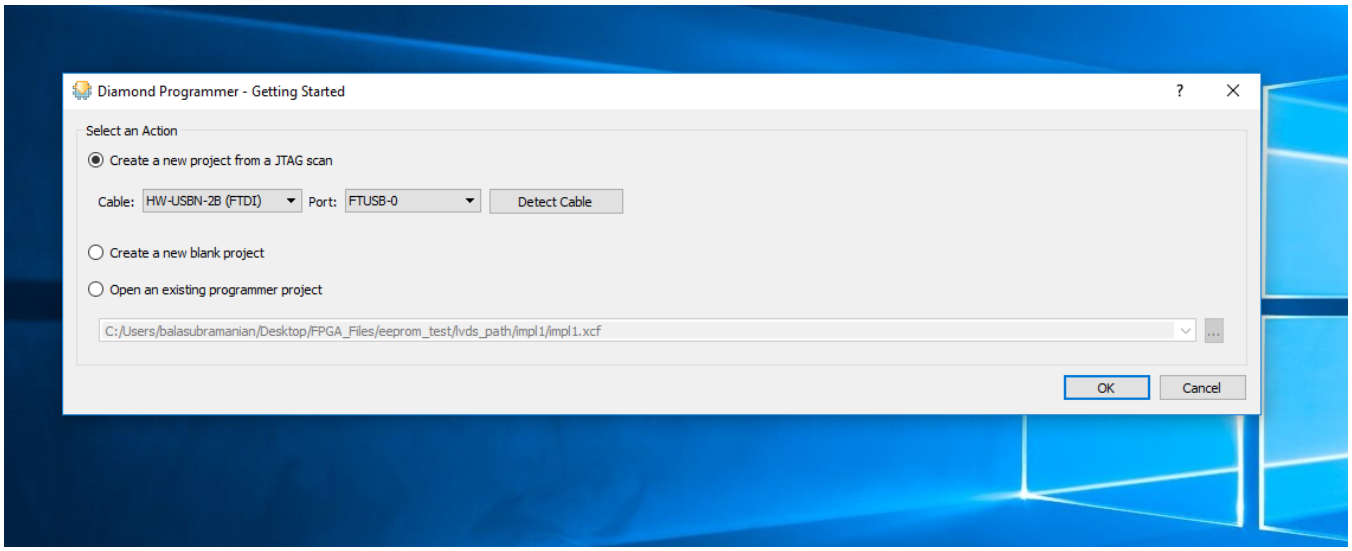


Figure 23. FTDI USB Cable Detection

2. The FPGA device (LFE5UM-85F) is detected in the programmer window, as shown in Figure 24.

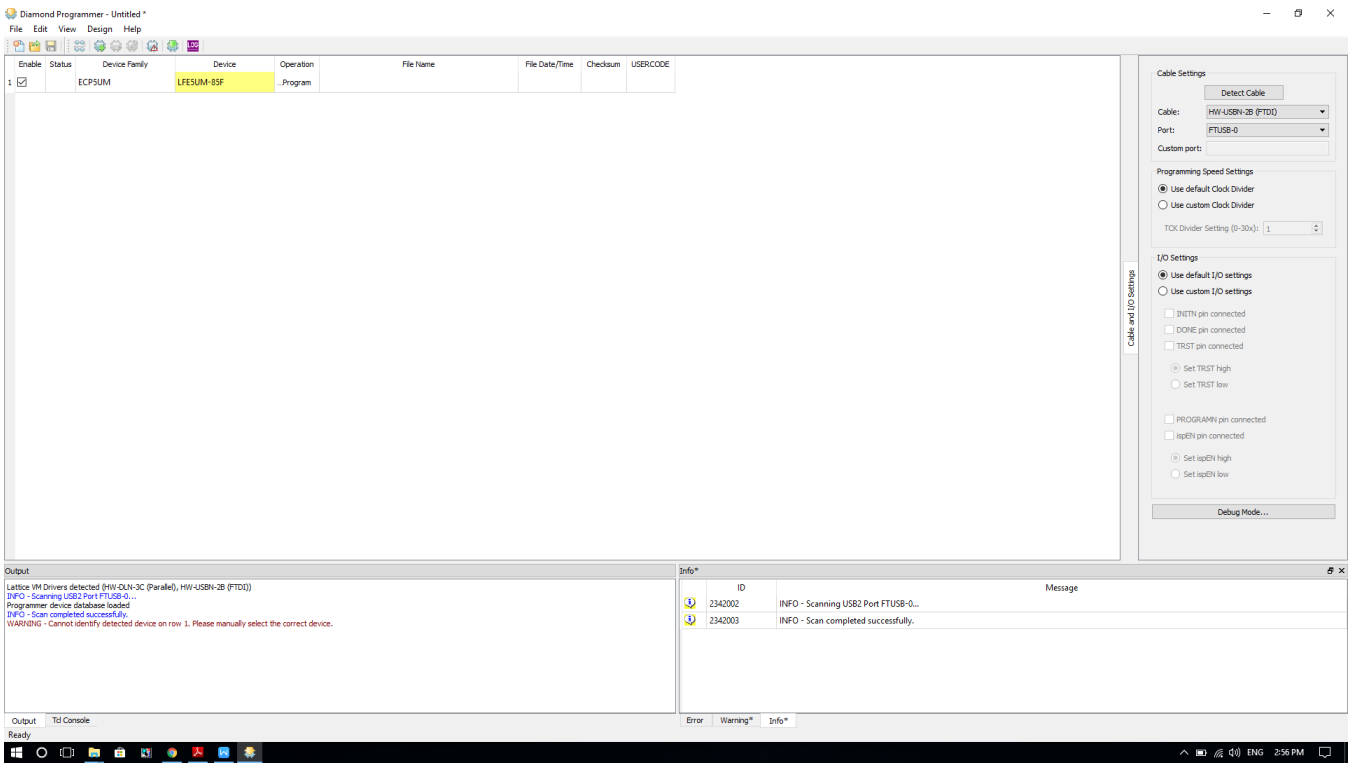


Figure 24. FPGA Device Scan

3. If the command line prompts to select the device manually, then click on the Device column and select the LFE5UM-85F device.
4. Click the Operation tab, and select the Access mode options as JTAG 1532 mode; and for Operation select “Erase Only” for the Erase bit file in the FPGA, as shown in Figure 25.

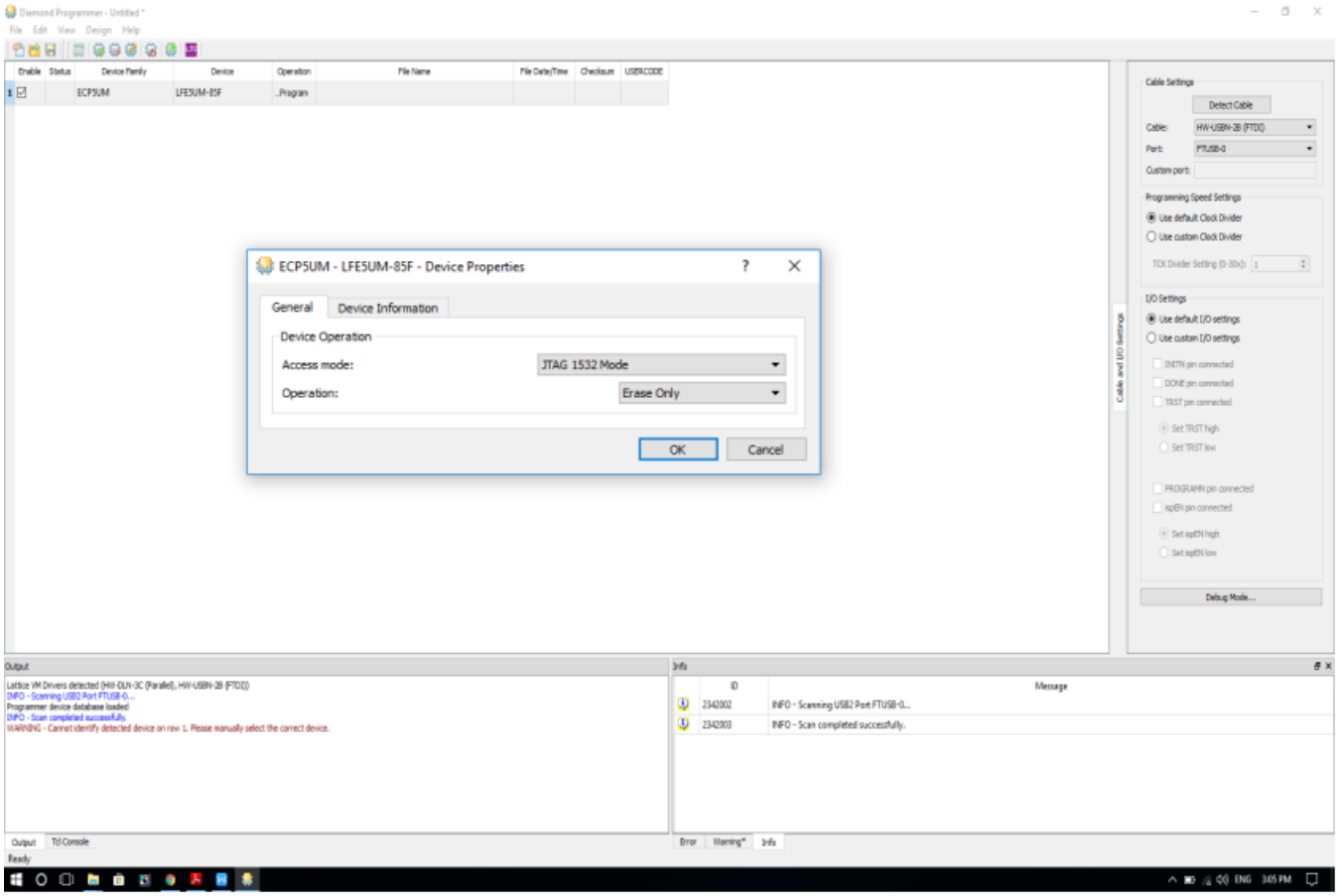


Figure 25. Select JTAG Option and Bit File

5. Click the Program option to complete the process.
6. Click the Operation tab to open the window, as shown in Figure 26.
   a. Set the Access mode option as SPI Flash Background Programming.
   b. Set the Operation as “SPI Flash Erase, Program, Verify” for bit file programming.
   c. Select the bit file from the following path with the mmWave studio package: \mmwave_studio_xx_yy\mmWaveStudio\DCA1000FPGA\ DCA1000_FPGA_RECORD_xx_yy.bit file in the programming file option. (xx -> Version No, yy -> Release Date).
   d. Select SPI Flash device as (Family: SPI Serial Flash, Vendor: Micron, Device: SPI-N25Q128A, Package: 16-pin SO16)

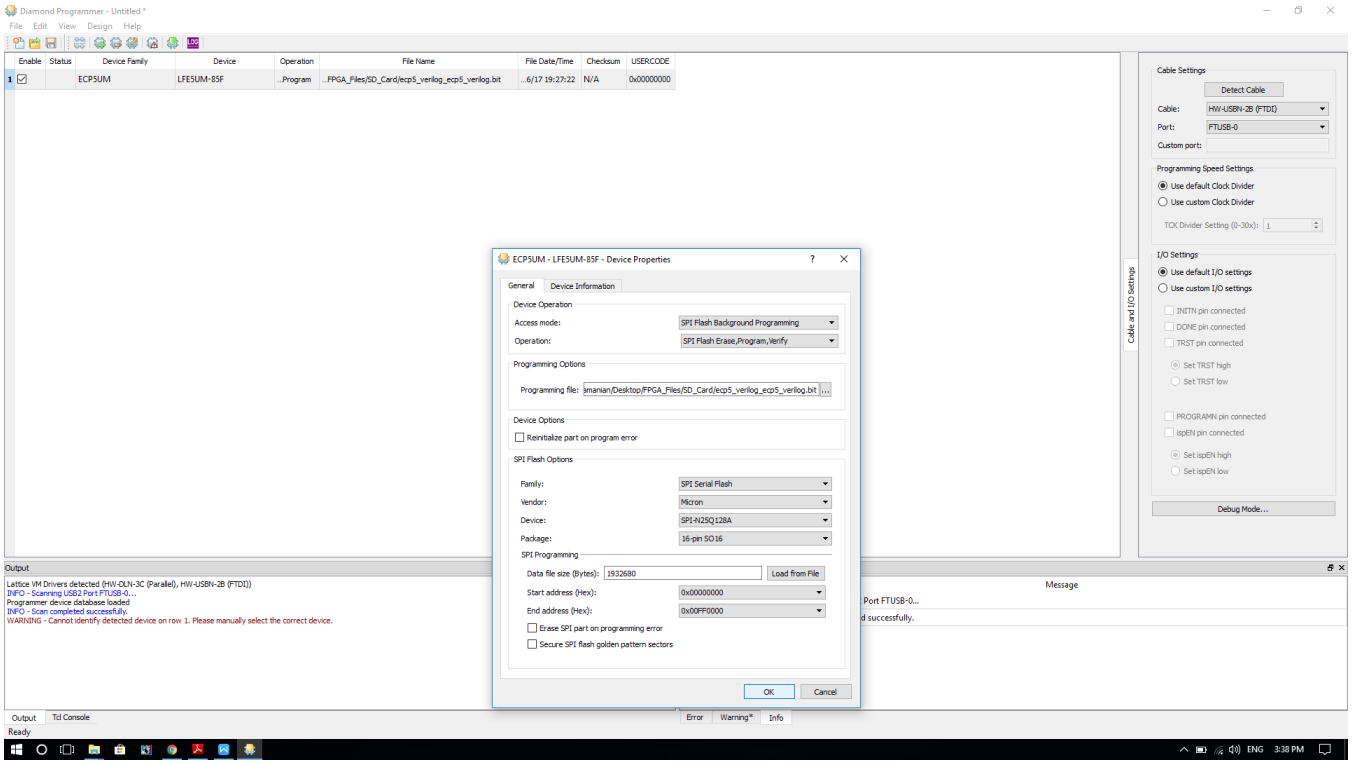


Figure 26. FPGA SPI Flash Programming

7. Click the Program option to complete the process.
8. When the programming is done, press the PROGRAMN(SW4) button or power cycle the board to load the FPGA bit file from SPI flash.
9. The FPGA DONE LED (LD14) glows to confirm the successful loading of the FPGA bit file from SPI flash.

10 Limitations

- This release supports LVDS to Ethernet streaming for raw mode and data segregated mode data capture.
- The SD card storage and playback feature is not supported in this release.
- The DCA1000EVM FPGA requires a minimum delay of 12 ms between the bit clock starts and the actual LVDS data start to lock the LVDS PLL IP.
- In default conditions, Ethernet throughput varies up to 325 Mbps speed in a 25-µs Ethernet packet delay. The user can change the Ethernet packet delay from 5 µs to 500 µs to achieve different throughputs, as explained in Table 19, to achieve different Ethernet throughputs.

Table 19. Theoretical Throughput

Ethernet Packet Delay	Theoretical Throughput (Mbps)
5 µs	~706
10 µs	~545
25 µs	~325
50 µs	~193

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Original (May 2018) to A Revision	Page
• Updated DCA1000EVM Functional Block Diagram.	4
• Updated FPGA - SPI Flash Programming Mode section.	25

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265