RFS2

User Manual

www.terasic.com

Content

Chapter 1

Introduction

The RFS2 (Radio Frequency and Sensor) daughter card is designed for the applications such as wireless control, environment monitor, and IoT (Internet of Things). The daughter card can be linked to the FPGA development kit via the 2x20 Pin GPIO connector.

The daughter card include Wi-Fi and Classic Bluetooth (BT) and Bluetooth Low Energy(BLE) wireless communication capabilities and includes a lot of sensors. Sensors include ambient light sensor, temperature sensor, humidity sensor, accelerometer, magnetometer, and gyroscope. An UART-to-USB interface also can provide communication channel with Host PC.

The RFS2 board is designed for wireless communication, environmental monitoring, networking, and IoT (The Internet of Things) applications. It can be used with the DE series FPGA motherboard to expand board functions. The RFS2 uses a 2x20 GPIO to interface with the FPGA board. It has Wi-Fi, Classic Bluetooth (BT) and Bluetooth Low Energy(BLE) wireless communication capabilities; it also has the ability to communicate through a PC host from UART-to-USB interface. In addition, there are all kinds of sensors, such as an ambient light sensor, temperature and humidity sensors, accelerators, gyroscopes, magnetometer ... and so on.

1.1 Package Contents

Figure 1-1 shows the RFS2 kit content. It includes

1. RFS2 Daughter Card
2. CD Download Guide
3. 40-pin IDC to Box Header Cable

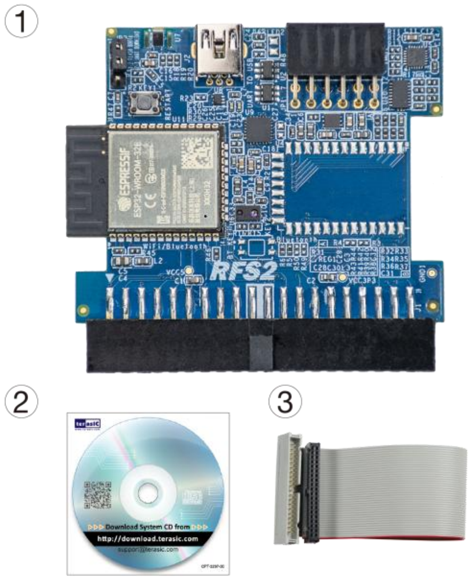


Figure 1-1 The RFS2 Kit content

1.2 RFS2 System CD

The RFS2 System CD contains all the documents and supporting materials associated with the daughter card, including the user manual, reference designs, and device datasheets. Users can download this system CD from the link: http://RFS2.terasic.com/cd.

1.3 Assemble the RFS2 Daughter Card

The RFS2 daughter card can be connected to any FPGA development kit that is equipped with a 2x20 Pin GPIO connector. It can directly connect to the FPGA mainboard or indirectly connect to the FPGA board via a 40-pin IDC Cable. The pictures below show how the RFS2 daughter card is connected to various Terasic FPGA Boards:

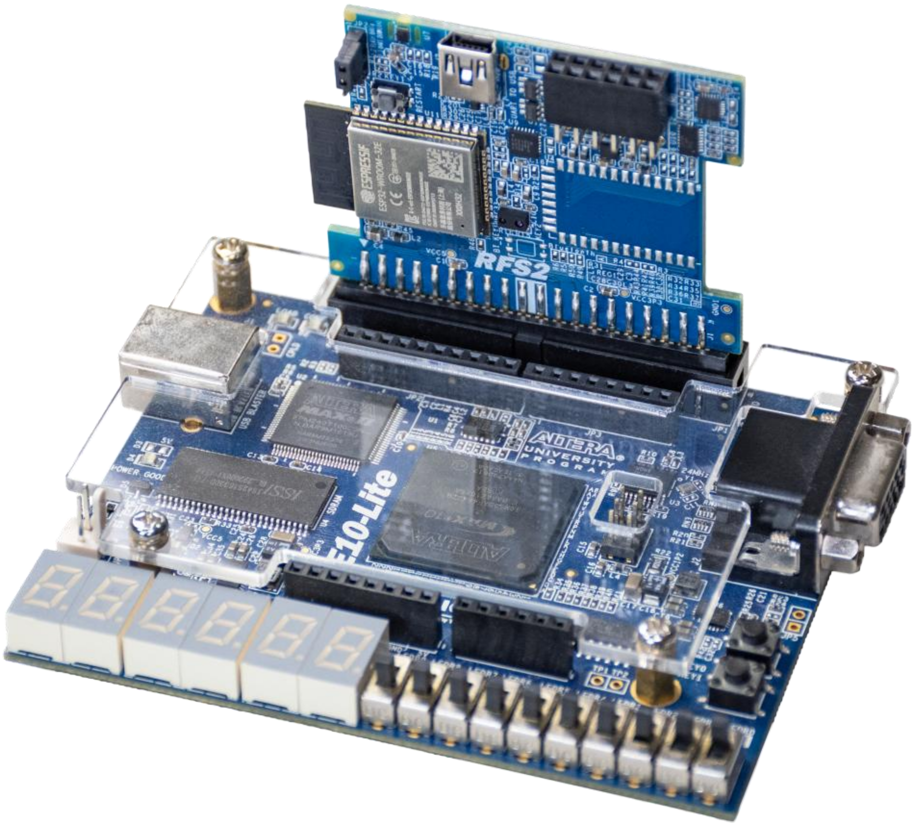


Figure 1-2 Connect the RFS2 to DE10-Lite directly

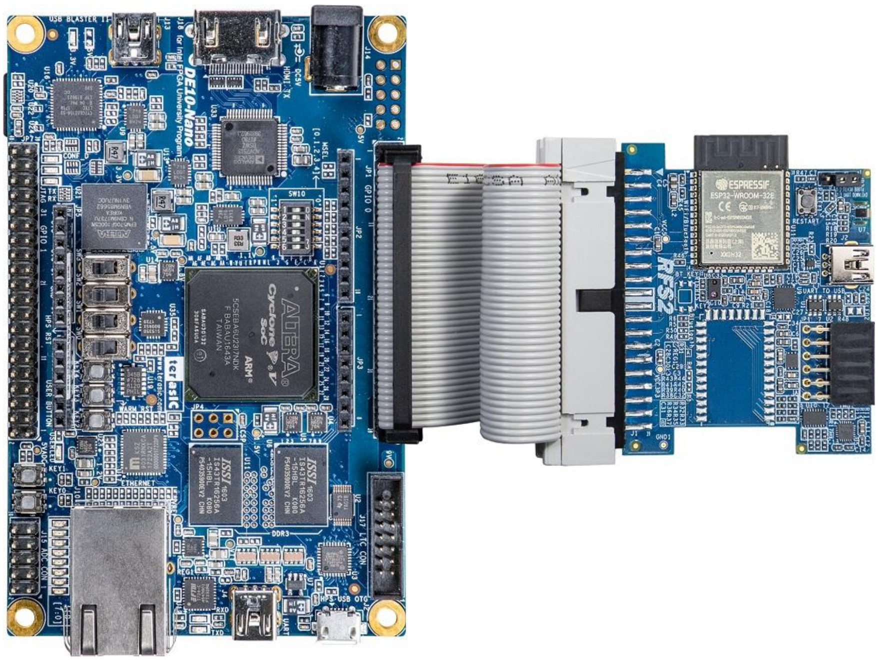


Figure 1-3 Connect the RFS2 to DE10-Nano

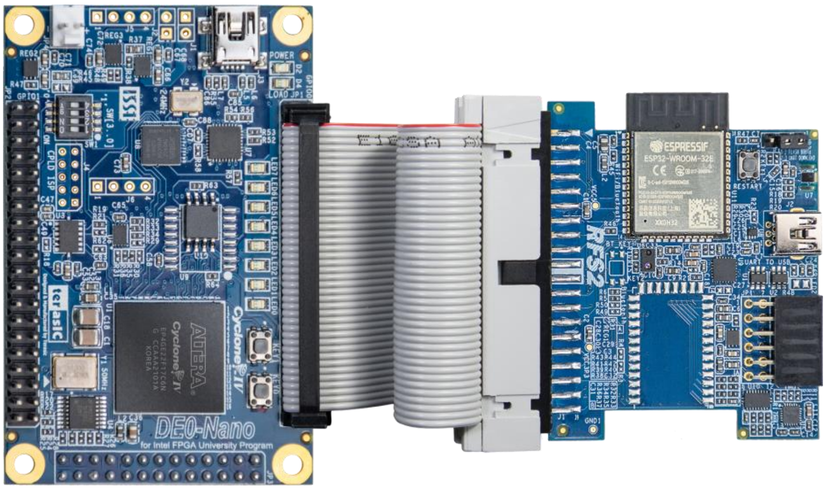


Figure 1-4 Connect the RFS2 to DE0-Nano

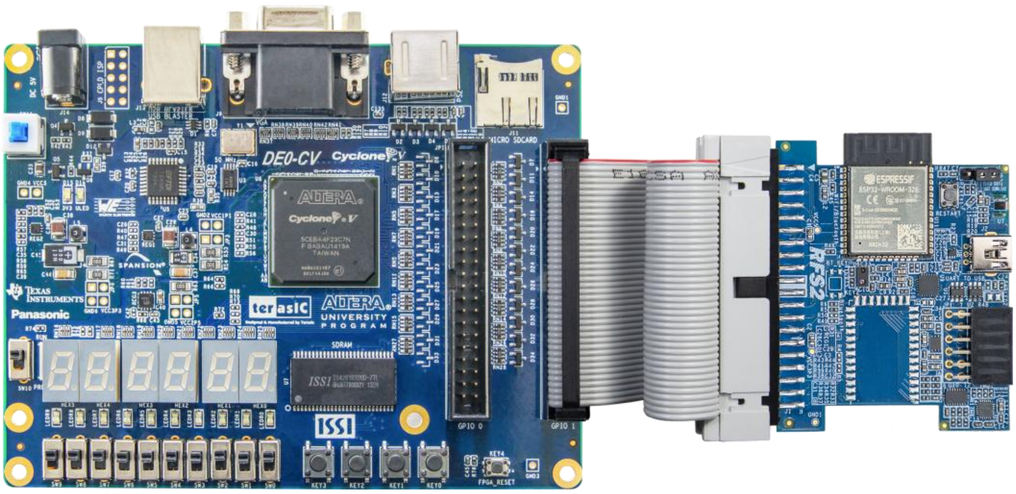


Figure 1-5 Connect the RFS2 to DE0-CV

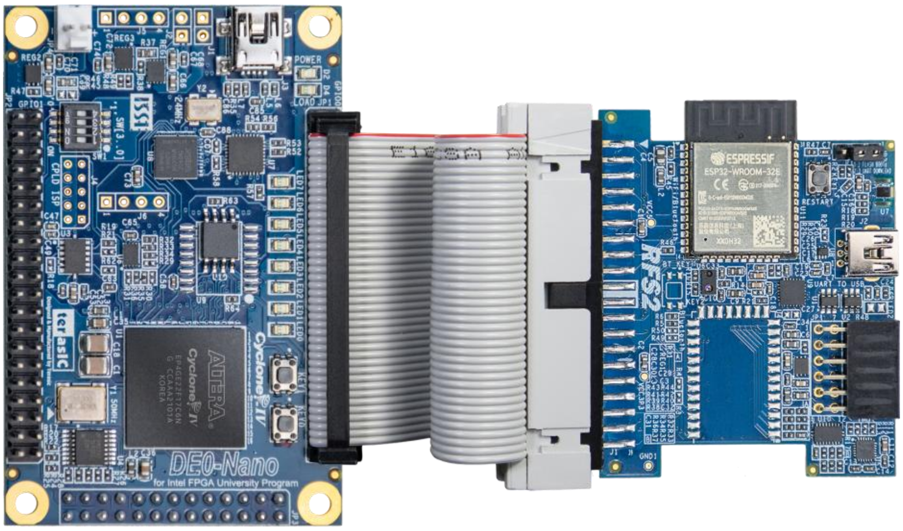


Figure 1-6 Connect the RFS2 to DE0-Nano

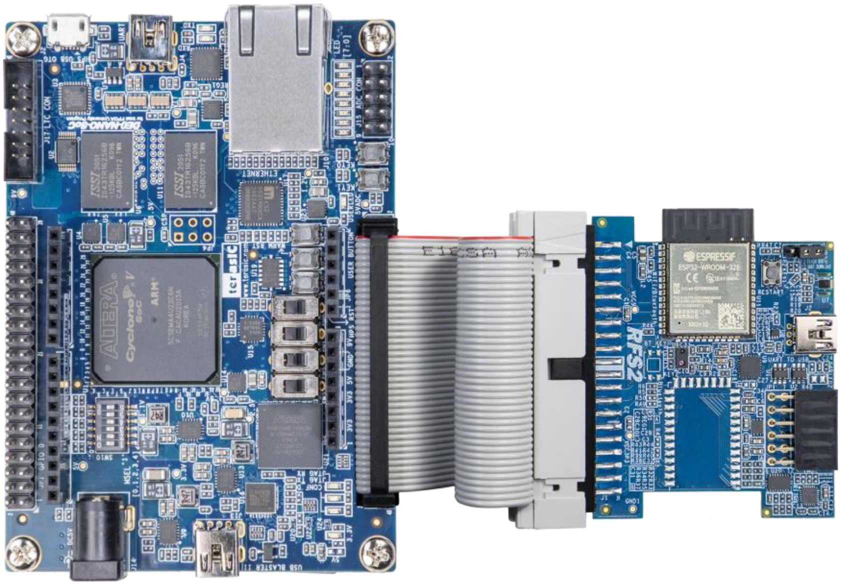


Figure 1-7 Connect the RFS2 to DE0-Nano-SoC

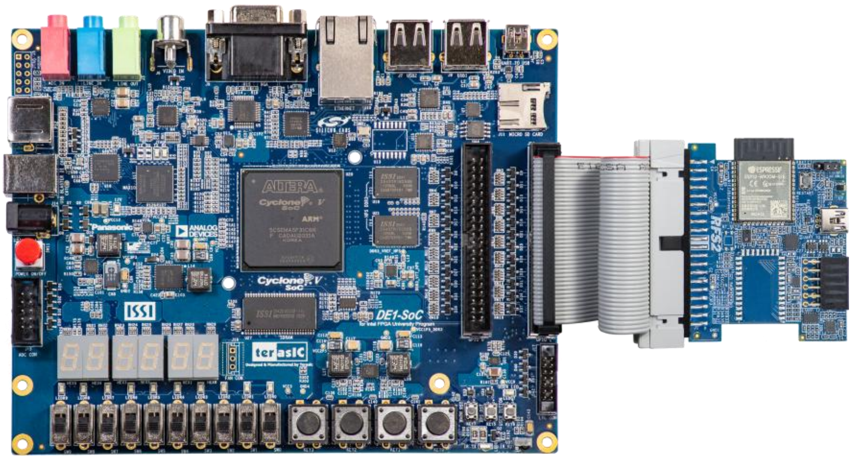


Figure 1-8 Connect the RFS2 to DE1-SoC

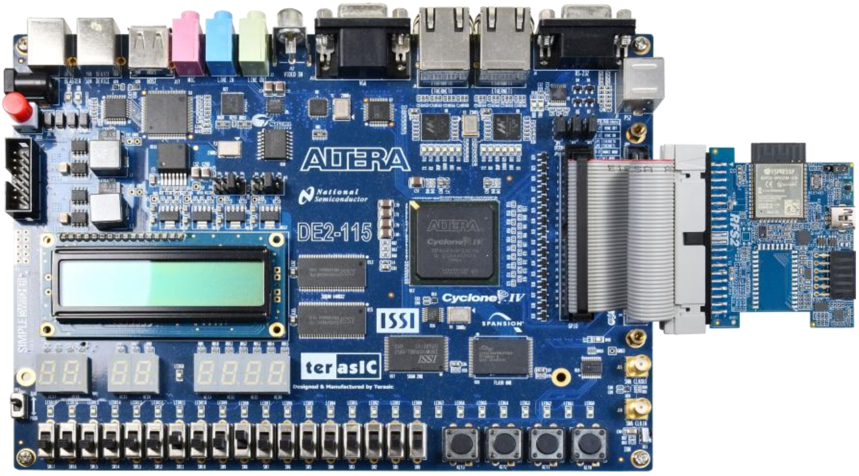


Figure 1-9 Connect the RFS2 to DE2-115

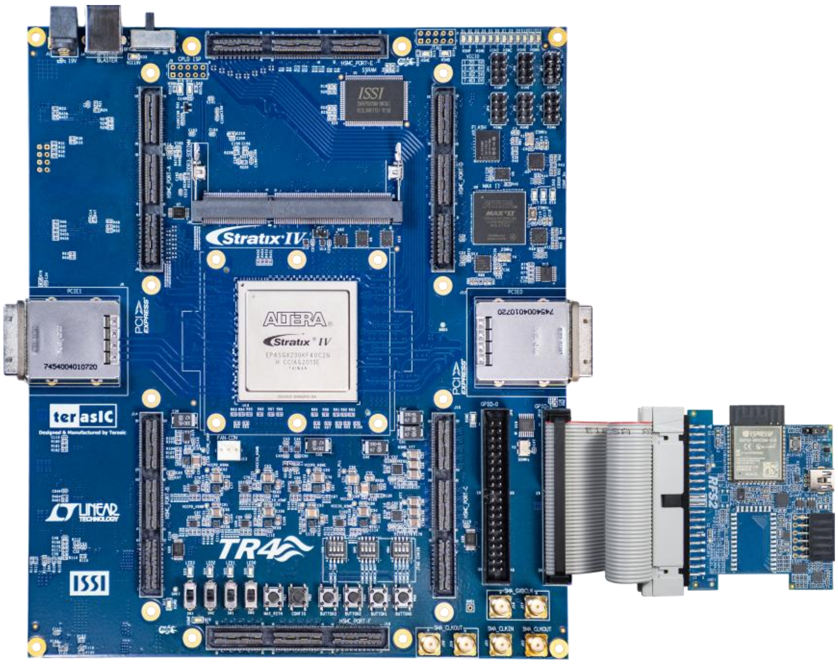


Figure 1-10 Connect the RFS2 to TR4

1.4 Getting Help

Here are the addresses where you can get help if you encounter any problems:

Terasic Technologies
9F., No.176, Sec.2, Gongdao 5th Rd, East Dist, Hsinchu City, 30070. Taiwan
Email: support@terasic.com
Tel.: +886-3-575-0880
Website: http://RFS2.terasic.com

This chapter will introduce the RFS2 daughter card included in the Kit. The daughter card is interfaced to FPGA mainboard by 2x20 GPIO interface. Except for the uart-to-usb chip, the card is powered from the FPGA mainboard. The 3.3V source in the 2x20 GPIO interface is sued to drive this daughter card. The uart-to-sub chip is powered from the USB power which came from host PC.

2.1 Features

Figure 2-1 shows a photograph of the RFS2 daughter card.

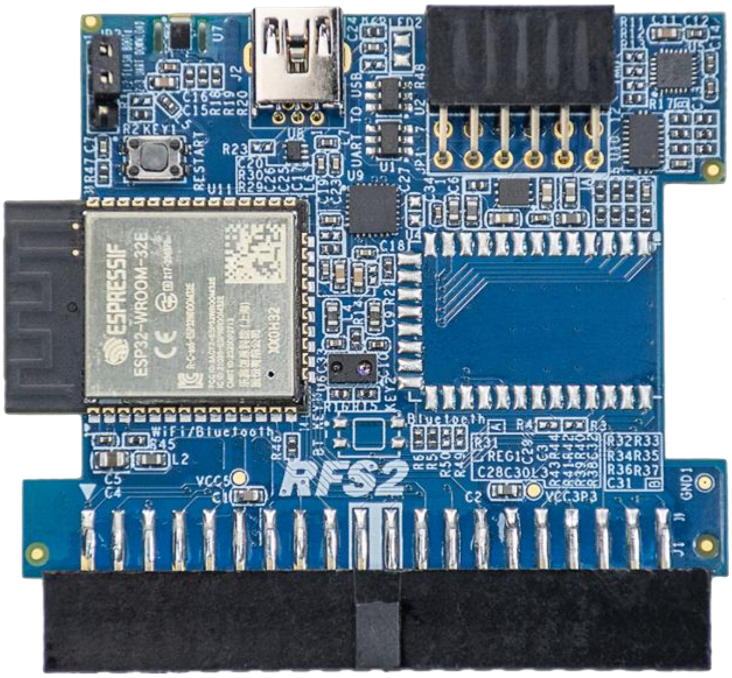


Figure 2-1 RFS2 Daughter Card

The features of the RFS2 card are:

- Wi-Fi, using ESP32-WROOM-32E module
- Classic Bluetooth (BT) and Bluetooth Low Energy(BLE), using ESP32-WROOM-32E module
- Ambient Light Sensor

- Temperature and humidity sensor
- 9-axis sensor – accelerometer, magnetometer, and gyroscope
- UART to USB
- 2x20 GPIO interface
- 2x6 TMD GPIO Header

2.2 Block Diagram of the RFS2 Daughter Card

Figure 2-2 shows the block diagram of the RFS2 daughter card. 2x20 GPIO is interface of this daughter card. The sensors can be communicated with I2C interface. Besides I2C, the 9-axis sensor also can be communicated with SPI interface if hi-speed is required. The UART-to-USB interface is 4-pin UART. The Bluetooth module interface is 2-pin UART. The Wi-Fi/ Bluetooth BLE module interface is 6-pin UART. In the 2x6 TMD header, signals are connected to the 2x20 GPIO header directly.

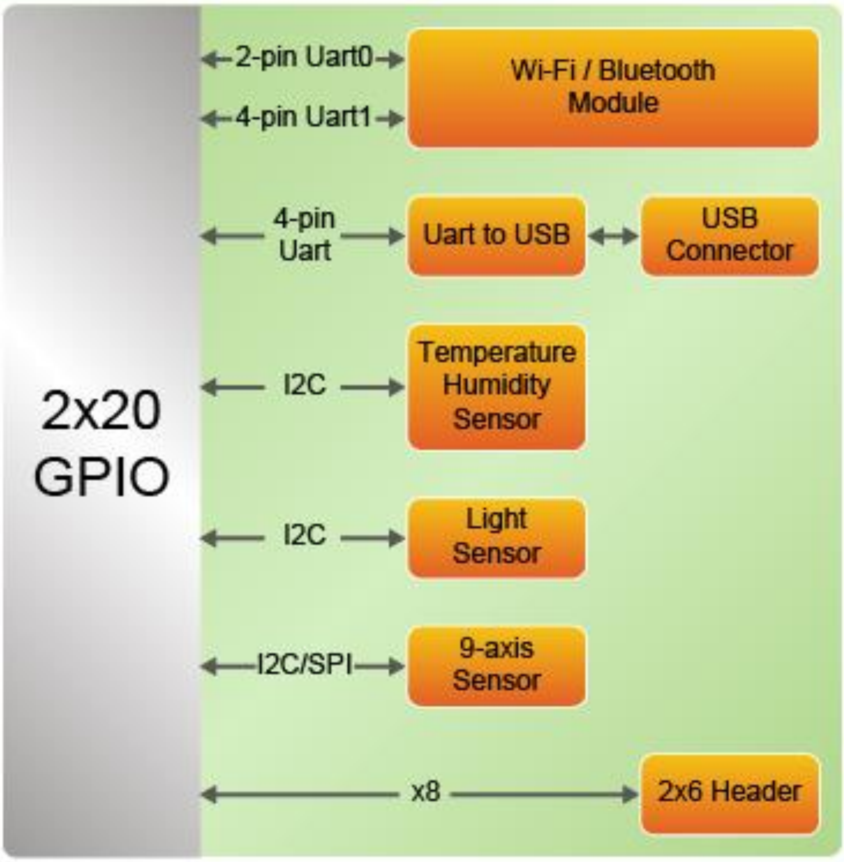


Figure 2-2 Block Diagram of RFS2 Card

2.3 Component Layout

Figure 2-3 shows the major component layout in the RFS2 daughter card.

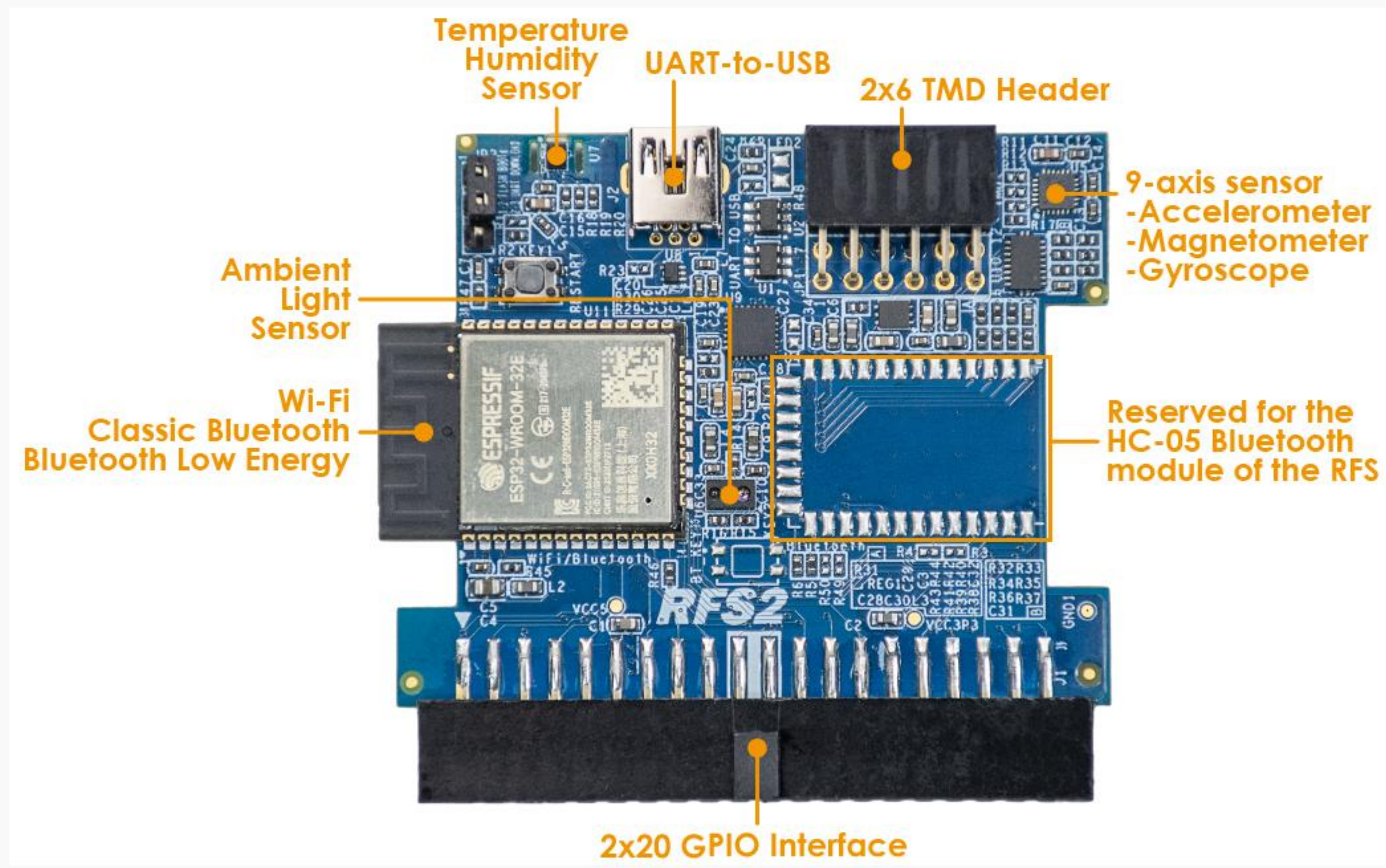


Figure 2-3 Major component layout on the RFS2 card

2.5 Interface for FPGA Host

The RFS2 card is connected to the host FPGA through the JP1 - a 2x20 GPIO header as shown in Figure 2-4. VCC3P3 power is used to drive this daughter card. BT_UART_CTS and BT_UART_RST is reserved pins. There are total 34 signal pins used.

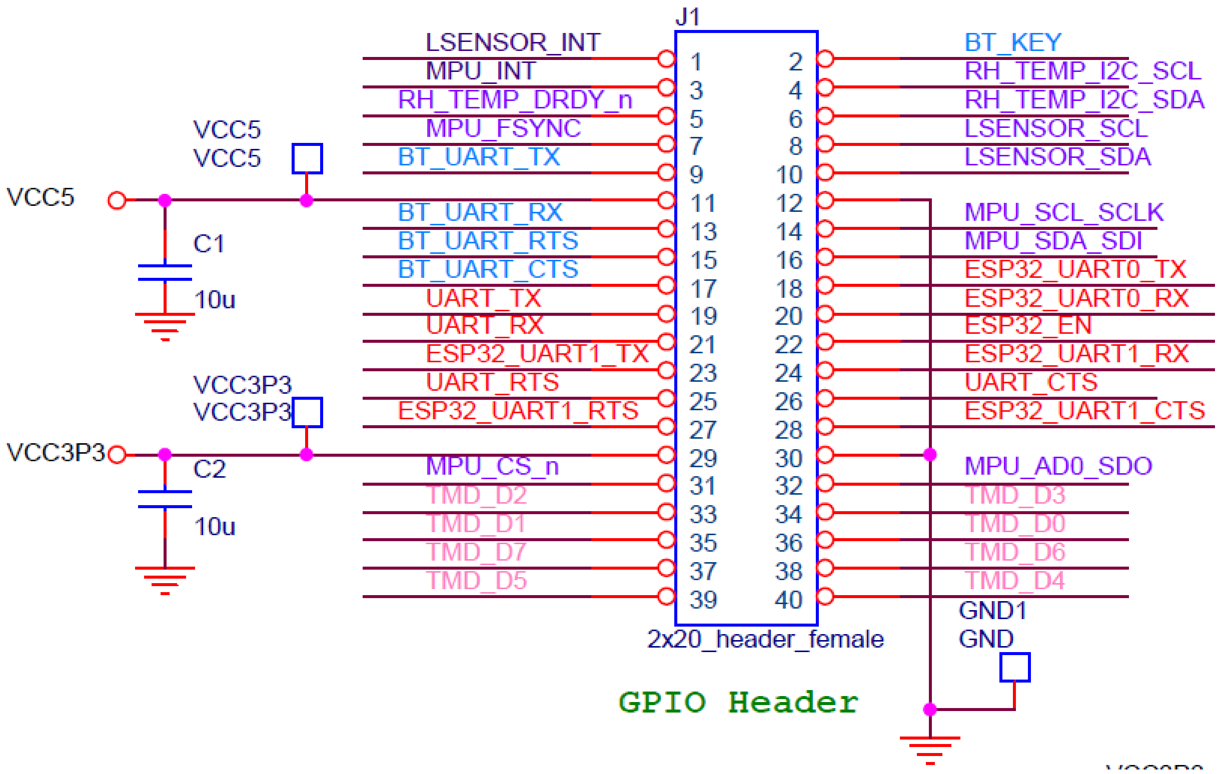


Figure 2-4 2x20 GPIO Pinout

Chapter 3

RTL Example Designs

This chapter will demonstrate how the FPGA to communicate with the RFS2 daughter card in RTL code.

3.1 Query Current Time through Wi-Fi

This section describes how to the Wi-Fi signal is transmitted via the ESP32-WROOM-32E Wi-Fi module on the RFS2 daughter card. The command is transmitted from the FPGA to the Wi-Fi module via UART. This demonstration uses the ESP32-WROOM-32E Wi-Fi module to query the current time. If the connection is successful, the current time will be display on the 7-segment of the board in hour:minute:second format.

■ Block Diagram

Figure 3-1 shows the function block diagram of querying current time through Wi-Fi demonstration.

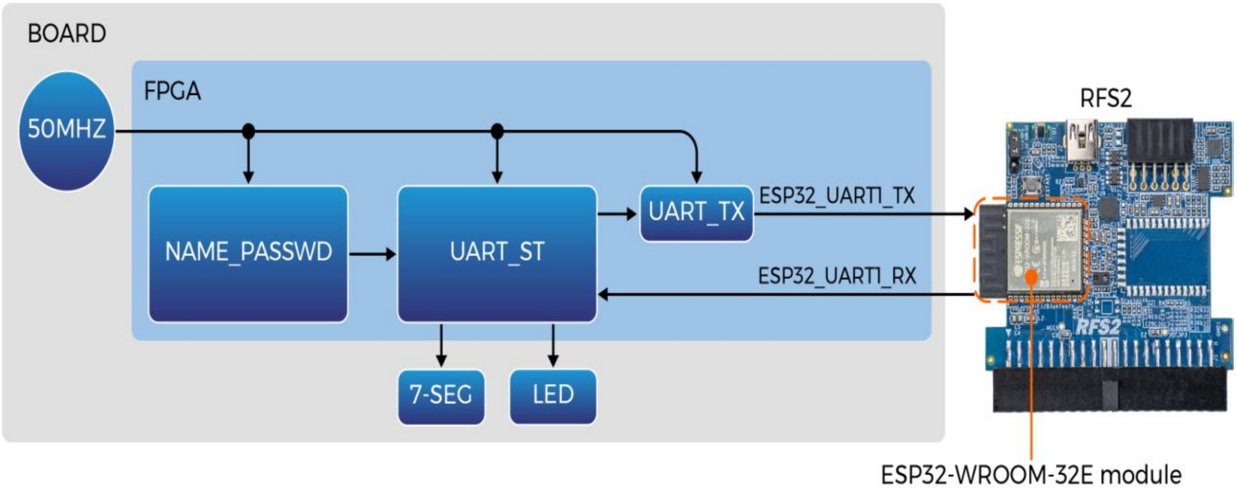


Figure 3-1 Function block diagram of querying current time

NAME_PASSWD**:** It takes NAME and PASSWORD for the Wi-Fi login. Users can enter up to 20 English, number, or symbol characters for each of them. The module will calculate the length of character in byte automatically.

UART_TX: It is the UART TX with baud-rate at 115200 bps.

UART_ST: It sends a series of command which queries the current time to the **ESP32-WROOM-32E** Wi-Fi module via the UART RX. The process can be divided to the 7 steps below. If the connection is successful, the current network time in hour:minute:second will be displayed on the 7-segment of the board.

Step 1: Reset the ESP32-WROOM-32E Wi-Fi module.
Step 2: Set the ESP32-WROOM-32E Wi-Fi module to the station mode.
Step 3: Enter the name and password.
Step 4: Establish the connection.
Step 5 : Ready to send out the request.
Step 6: Send out the request to retrieve the current time.
Step 7: Read the current time.

This demo can be realized on DE10-NANO and DE10-Lite. Please follow the steps below for running the demo on these two boards.

■ Quartus Project Information

For DE10-NANO Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Nano\DE10_NANO_WIFI_TIME_RTL
Demo Batch File	Demonstrations\DE10_Nano\DE10_NANO_WIFI_TIME_RTL\demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Lite\DE10_Nano\DE10_LITE_WIFI_TIME_RTL
Demo Batch File	Demonstrations\DE10_Lite\DE10_LITE_WIFI_TIME_RTL\demo_batch

■ Demonstration Setup for DE2-115

Please follow the procedures below to setup the demonstration, as shown in **Figure 3-2**.

1. Connect the RFS2 daughter card to the GPIO0 of DE10-NANO board.
2. Plug in 5 V DC to DE10-NANOConnect the host PC to the USB connector (J9) on DE2-115 via USB cable.
3. Connect the host PC to the USB connector (J13) on DE10-NANO via USB cable.
4. Please make sure Quartus II has been installed on the host PC.compilation is successful, copy the generated .sof file to **\demo_batch**.

5. Search for the key string assign NAME="" and assign PASSWORD="" in DE10_Nano_golden_top.v. Insert user name and password in "", respectively. After the compilation is successful, copy the generated .sof file to \demo_batch.
6. Execute the batch file " test.bat" under the demo_batch folder of DE10_NANO_WIFI_TIME_RTL project.
7. Open the SignalTapeII file " stp1.stp" under the demo_batch folder of DE10_NANO_WIFI_TIME_RTL project.
8. Press KEY0 and wait for 15 ~ 20 seconds. The current network time will be displayed in decimal hour:minute:second ,as shown in **Figure 3-3**.

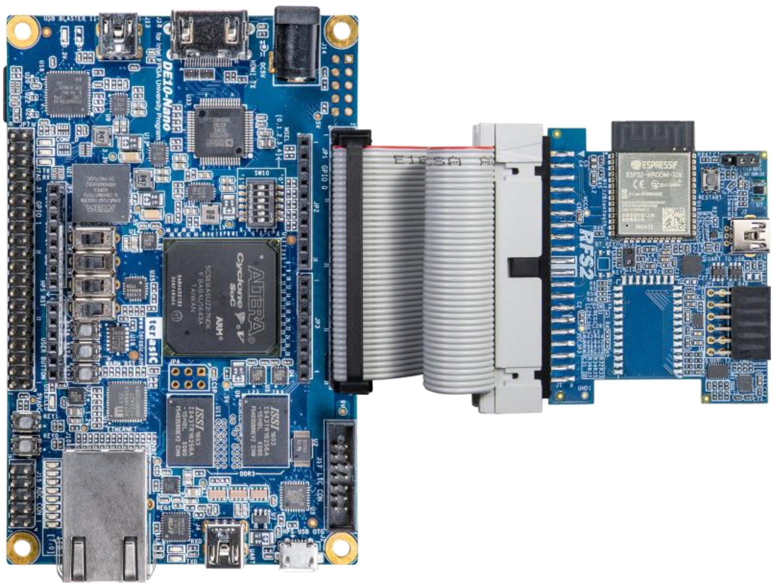


Figure 3-2 Wi-Fi Time setup with DE2-115

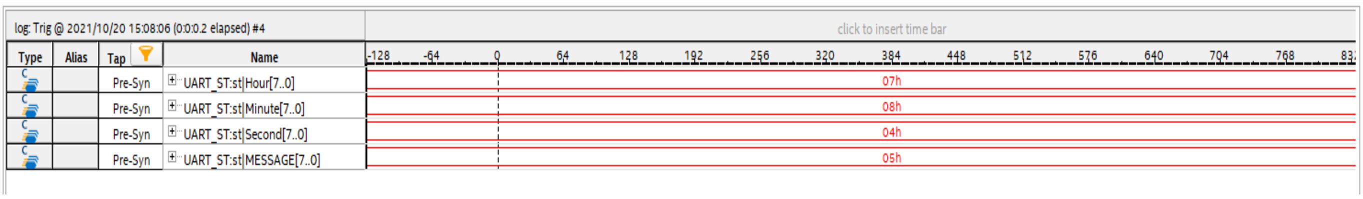


Figure 3-3 Using Signaltap II tool to read the current network time

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration, as shown in **Figure 3-4**

1. Connect the RFS2 daughter card to the GPIO of DE10-Lite board.
2. Connect the host PC to the USB connector (J9) on DE10-Lite via USB cable.

3. Please make sure Quartus II has been installed on the host PC.Search for the key string assign NAME="" and assign PASSWORD="" in DE10_LITE_WIFI_TIME_RTL.v. Insert user name and password in "", respectively. After the compilation is successful, copy the generated .sof file to **\demo_batch**.
4. Search for the key string assign NAME="" and assign PASSWORD="" in DE10_LITE_Golden_Top.v. Insert user name and password in "", respectively. After the compilation is successful, copy the generated .sof file to \demo_batch.
5. Execute the batch file " test.bat" under the demo_batch folder of DE10_LITE_WIFI_TIME_RTL project.
6. Press KEY0 and wait for 15 ~ 20 seconds. The current network time will be displayed on the 7-segment of DE10-Lite in decimal hour:minute:second.

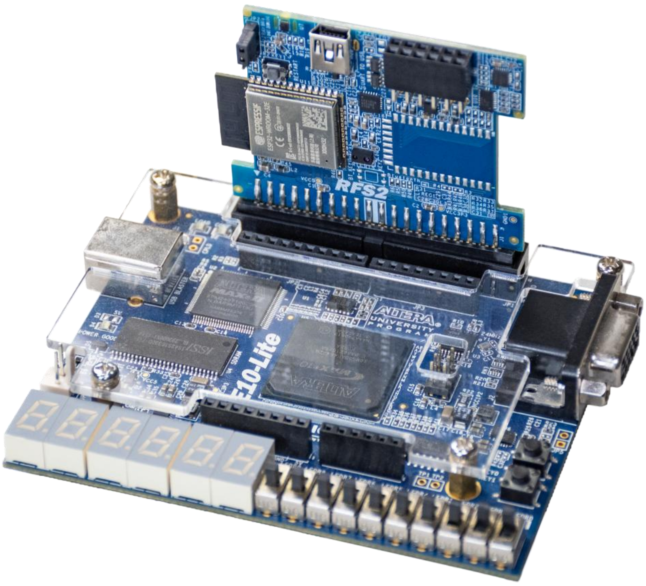


Figure 3-4 Wi-Fi Time Demo setup for DE10-Lite

3.2 Bluetooth SPP Slave

This demonstration uses the ESP32-WROOM-32E module to pair the mobile phone through Bluetooth. If the pairing is successful, the debugging APP on the mobile phone can be used to send the data to the FPGA via Bluetooth and then the 7-segment display or LED on the FPGA board will display the receiving data from the mobile phone. It can also upload the data which are pre-stored in the FPGA to the mobile phone and display on the Bluetooth debugging APP.

■ Block Diagram

Figure 3-5 shows the function block diagram Demonstration of reading and writing data via

Bluetooth.

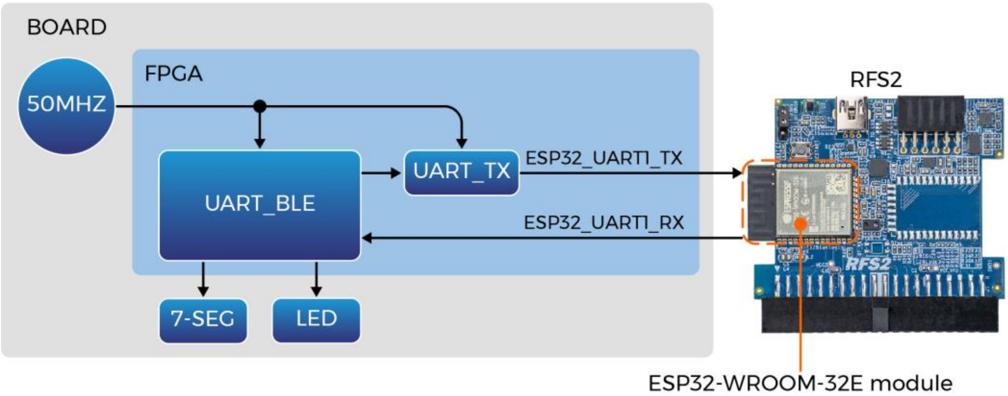


Figure 3-5 Function Block Diagram

UART_TX: It is the UART TX with baud-rate at 112500 bps.
UART_BLE: This module will send a series of AT commands to the ESP32-WROOM-32E module via UART. The process of this module can be divided to the 12 steps below. If the connection is successful, you can use the Bluetooth debugging APP of the mobile phone to transfer data to the FPGA board and display on the 7-segment display or LED. You can also transfer the data stored in the FPGA to the Bluetooth debugging APP of the mobile phone.

Step1:Reset the ESP32-WROOM-32E module
Step2:AT+BLEINIT=2\r\n
Step3:AT+BLEGATTSSRVCRE\r\n
Step4:AT+BLEGATTSSRVSTART\r\n
Step5:AT+BLENAME="Terasic_RFS2"\r\n
Step6:AT+BLEADVDATA="201060D09546572617369635f52465332"\r\n
Step7:AT+BLEADVSTART\r\n
Step8:AT+BLESPPCFG =1,1,6,1,5\r\n
Step9:AT+BLESPP\r\n
Step10:Terasic_ESP32E\r\n
Step11:+++
Step12:\r\n

This demo can be realized on DE10-NANO and DE10-Lite. Please follow the steps below for running the demo on these two boards.

■ Quartus Project Information

For DE10-NANO Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Nano\DE10_NANO_RFS2_BLE_SPP_Slave_RTL
Demo Batch File	Demonstrations\DE10_Nano\DE10_NANO_RFS2_BLE_SPP_Slave_RTL \demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Lite\DE10_LITE_RFS2_BLE_SPP_Slave_RTL
Demo Batch File	Demonstrations\DE10_Lite\DE10_LITE_RFS2_BLE_SPP_Slave_RTL \demo_batch

■ Demonstration Setup for DE10-Nano

Please follow the procedures below to setup the demonstration as shown in **Figure 3-6.**

1. Connect the RFS2 daughter card to the GPIO0 of DE10-NANO board.
2. Plug in 5V DC to DE10-NANO.
3. Connect the host PC to the USB connector (J13) on DE10-NANO via USB cable.
4. Please make sure Quartus II has been installed on the host PC.
5. Install the Bluetooth debugging app on the mobile phone. Below will introduce using the android phone to run this demo. Install the app named “LightBlue — Bluetooth Low Energy “ from the Google play (See **Figure 3-7**). Note, you can also find this app in the App Store for iPhone.

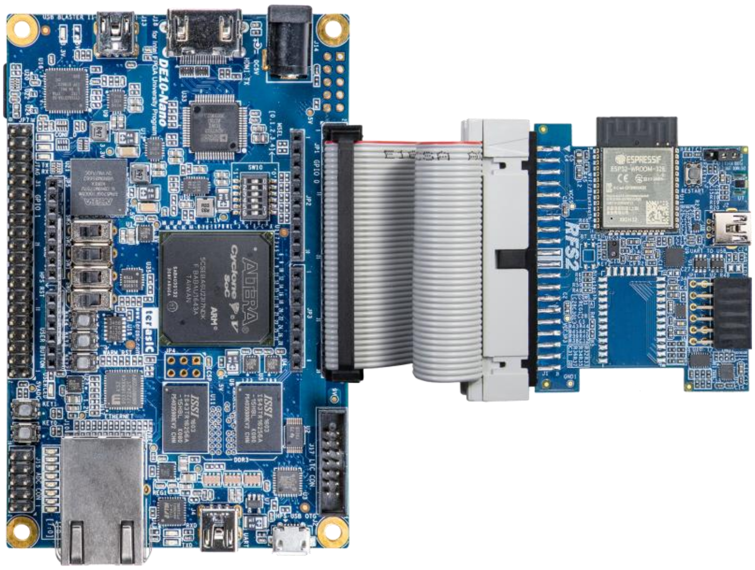


Figure 3-6 Demo setup on DE10-Nano

Figure 3-7 Bluetooth debugging app

6. Execute the batch file “test.bat” under the demo_batch folder of DE10_NANO_RFS2_BLE_SPP_Slave_RTL project.
7. Press KEY0 on the DE10-Nano and wait for 15 ~ 20 seconds. Note that when status of the LED[3] is light on and the other three LED (LED[2:0]) doesn’t light. It means the Bluetooth setup is ready for the RFS2 board and user can use the debugging APP of the mobile phone to connect to the Bluetooth of the RFS2.
8. Open the “LightBlue” — Bluetooth Low Energy app on your phone.

Figure 3-8 Open the Bluetooth debugging app

When you launch the app, you should be greeted with a list of nearby BLE devices by name. Find the “Terasic_RFS2” and press “CONNECT”.

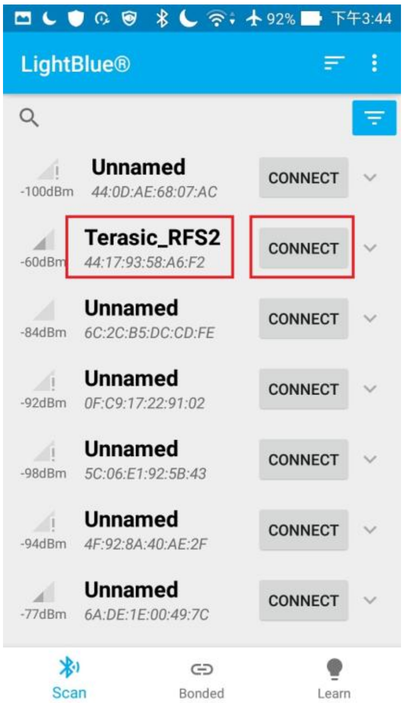


Figure 3-9 Open the Bluetooth debugging app

- Write data into the FPGA board:
  - When the app is connected to the RFS2 board through Bluetooth, find the “Writable” site and press the “➔” to enter the write function (See **Figure 3-10**).

- As shown in **Figure 3-11**, choose the Data format to “UTF-8 String” and enter one number such as “9”, then press “Write”.
- Observe the LED7~LED4, the 4 LED will show the hex number of the app just transmitted (See **Figure 3-12**).

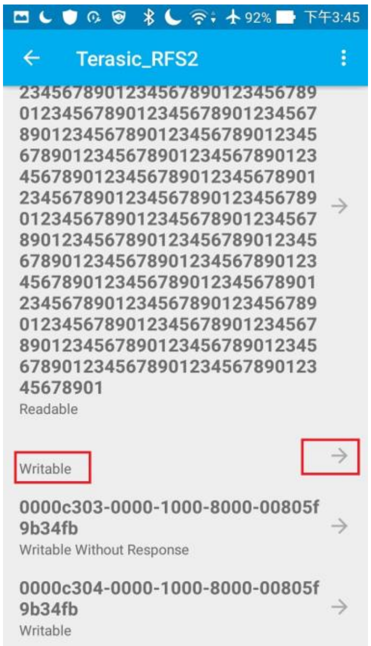


Figure 3-10 Open the Bluetooth debugging app

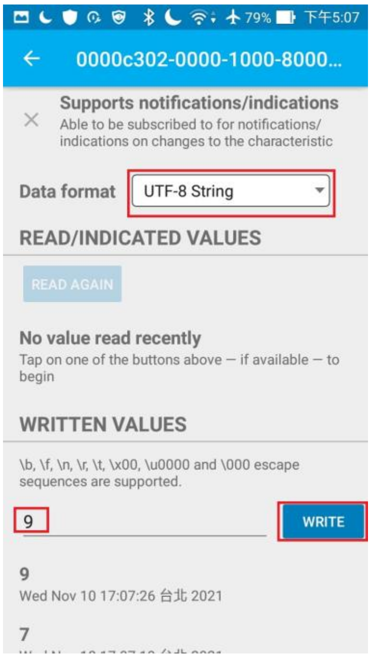


Figure 3-11 Write data into FPGA

Figure 3-12 Display the receiving data on LED

- Read string from the FPGA board
  - Find the "Notify" option and press the "➔" to enter the read function (See **Figure 3-13**).
  - As shown in **Figure 3-14**, choose the Data format to "UTF-8 String" and select the "SUBSCRIBE" in the READ/INDICATED VALES.
  - Press the KEY1 button on the DE10-Nano board, then a string "Terasic_ESP32E" will be send to the debug app from the FPGA board (See **Figure 3-15**).

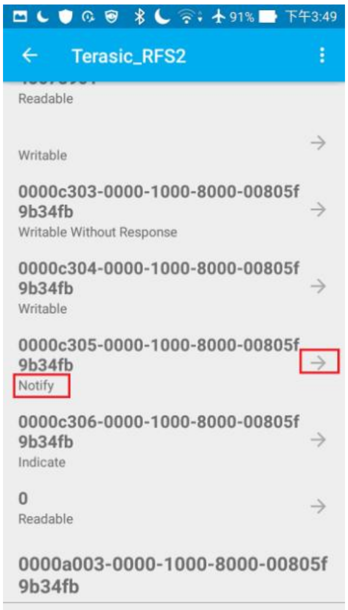


Figure 3-13 Read string from FPGA

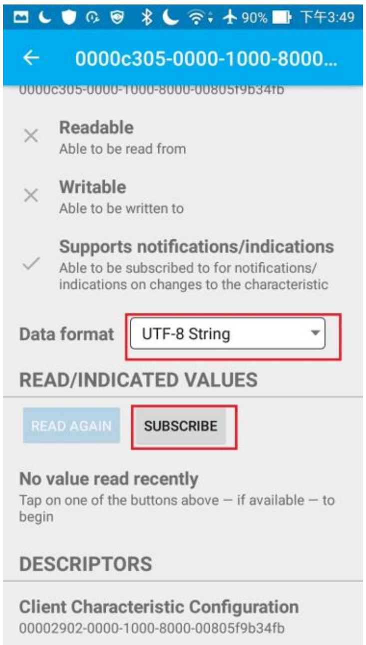


Figure 3-14 Read string from FPGA

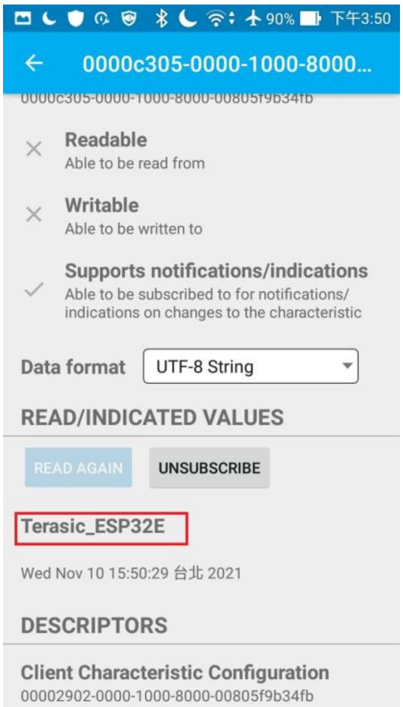


Figure 3-15 Read string from FPGA

■ Demonstration Setup for DE10-LITE

Please follow the procedures below to setup the demonstration as shown in **Figure 3-16**.

1. Power off the DE10-LITE board.
2. Connect the RFS2 daughter card to the GPIO connector of the DE10-Lite board.
3. Power on the DE10-LITE board.
4. Execute the batch file “ test.bat” under the demo_batch folder of DE10_LITE_BluetoothSPP_Slave project.
5. Follow the steps for the previous demo operation for DE10-Nano to download the debugging APP on your mobile phone and read write the string for the FPGA.
6. Note that for write data into the FPGA from the app, the DE10-Lite demo can write two numbers (See **Figure 3-17**)into the FPGA and display the number on the 7-segment display on the DE10-Lite (See **Figure 3-18**).

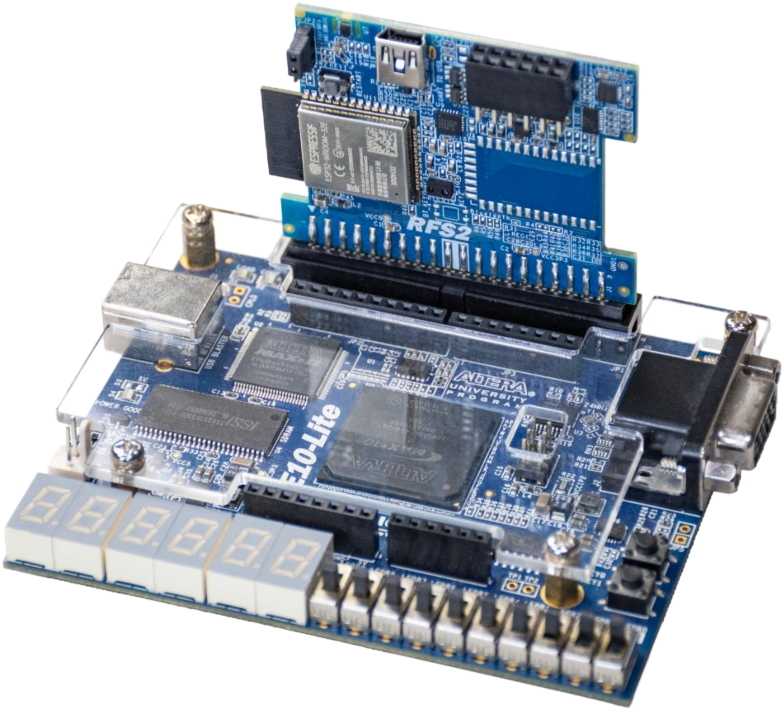


Figure 3-16 Demo setup on DE10-Lite

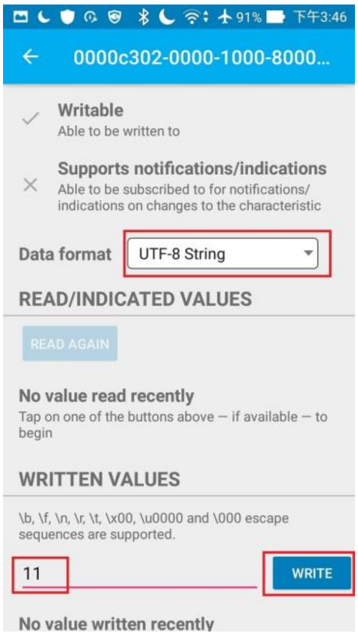


Figure 3-17 Write data into the DE10-Lite

Figure 3-18 Display data on DE10-Lite

3.3 Sensor Measurement

This part consists of three independent I2C controllers. Each of the I2C controllers is responsible for retrieving the measurement of corresponding sensor on RFS2 daughter card. The three sensors include humidity, temperature, light photo sensor, and 9-axis sensor which is the combination of 3-axis gyroscope, 3-axis accelerometer, and 3-axis magnetometer. The measurements of these three sensors can be displayed on the 7-segments onboard by the selection of switch settings or using the signaltab II tool in the Quartus.

■ Block Diagram

Figure 3-19 shows the function block diagram of Sensor Measurement demonstration.

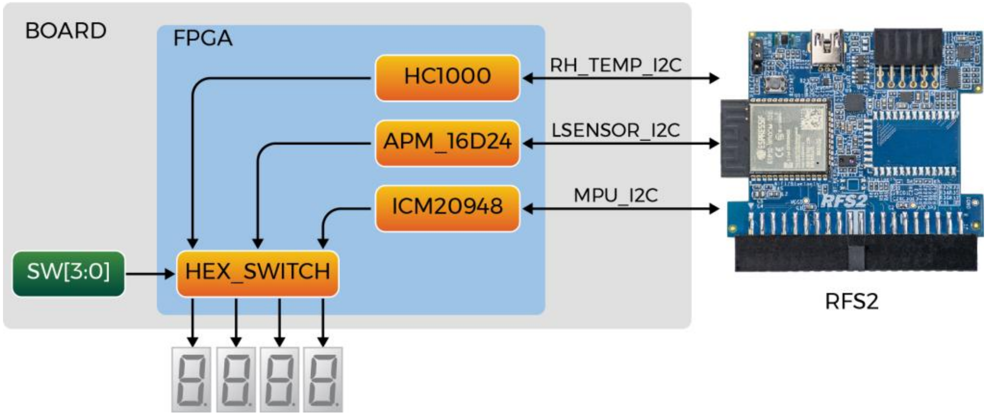


Figure 3-19 Function block diagram of sensor measurement

HC1000: It controls the HDC1000 IC on RFS2 daughter card via I2C bus. It can retrieve the measurement of humidity and temperature from the slave address 0x80. The values of humidity and temperature can be represented in decimal after conversion in percentage and degrees, respectively.

APM_16D24: It controls the APM-16D24 IC on RFS2 daughter card via I2C bus. It can retrieve the measurement of light photo sensor from ADC0 and ADC1, proximity sensor DATA ,IR sensor DATA , via slave address 0x70. The length of these four sets are 16-bit.

ICM-20948: It controls the ICM-20948 IC on RFS2 daughter card via I2C bus. There are two ICs integrated into a single ICM-20948 chip. One is the accelerometer, which can be accessed from slave address 0xD0 or DxD2 when the pin AD0_SDO = 1(board default setting)(The 7-bit of the slave address is 7'b1101001), and the other one is AK09916 as magnetometer, which can be accessed from slave address 0x18, provided the register 0x0F of ICM-20948 is set to 0x02 (I2C bypass mode). The usage is similar to several I2C slave devices in parallel for the three measurements of ICM-20948 (accelerometer, gyroscope, and magnetometer). Each one has x-y-z axis and each axis can be represented in 16-bit in 2's complement for positive and negative values. This module integrates the total of 9 sets of 16-bit data and outputs these values simultaneously.

HEX_SWITCH: This module takes total of 14 sets of data (one for humidity and temperature, another two for light photo sensor, and the last nine from 9-axis sensor) from the three independent modules above. The values can be represented on the 7-segments onboard(DE10-Lite) separately by the SW[3:0] onboard, as shown in Table 3-1.

Table 3-1 The Settings of SW[3:0] and Corresponding Values on four 7-segment

SW[3:0]=	Description	Format of HEX[3:0]
0	Humidity and temperature	HEX[3:2] for humidity (positive) in decimal HEX[1:0] for temperature (positive) in decimal
1	Light Sensor ADC0 (DATA0)	hexadecimal (positive)
2	Light Sensor ADC1 (DATA1)	hexadecimal (positive)
3	Proximity Sensor DATA	hexadecimal (positive)
4	IR sensor DATA	hexadecimal (positive)
5	Accelerometer X	hexadecimal (positive/negative)
6	Accelerometer Y	hexadecimal (positive/negative)
7	Accelerometer Z	hexadecimal (positive/negative)
8	Gyroscope X	hexadecimal (positive/negative)
9	Gyroscope Y	hexadecimal (positive/negative)
10	Gyroscope Z	hexadecimal (positive/negative)
11	Magnetometer X	hexadecimal (positive/negative)
12	Magnetometer Y	hexadecimal (positive/negative)
13	Magnetometer Z	hexadecimal (positive/negative)

This demo can be realized on DE10-NANO and DE10-Lite. Please follow the steps below for running the demo on these two boards.

■ Quartus Project Information

For DE10-NANO Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Nano\DE10_NANO_RFS2_SENSOR_RTL
Demo Batch File	Demonstrations\DE10_Nano\DE10_NANO_RFS2_SENSOR_RTL \demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V20.1.1

Project Directory	Demonstrations\DE10_Lite\DE10_LITE_RFS2_SENSOR_RTL
Demo Batch File	Demonstrations\DE10_Lite\DE10_LITE_RFS2_SENSOR_RTL \demo_batch

■ Demonstration Setup for DE10-Nano

Please follow the procedures below to setup the demonstration, as shown in the **Figure 3-22**.

1. Connect the RFS2 daughter card to the GPIO0 of DE10-NANO board.
2. Plug in 5 V DC to DE10-NANO.
3. Connect the host PC to the USB connector (J13) on DE10-NANO via USB cable.
4. Please make sure Quartus II has been installed on the host PC.
5. Execute the batch file " test.bat" under the demo_batch folder of DE10_NANO_RFS2_SENSOR_RTL project.
6. Open the SignaltapII file " all_sensor.stp" under the demo_batch folder of DE10_NANO_RFS2_SENSOR_RTL project .The measure value will be shown in the tool (see **Figure 3-21**)

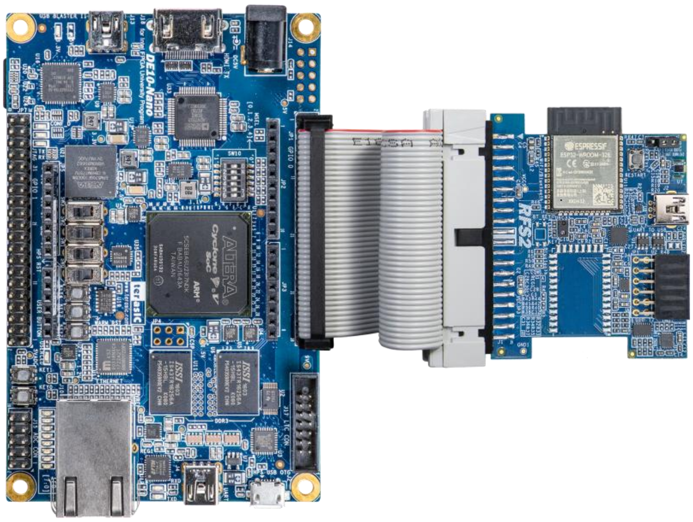


Figure 3-20 Sensor RTL Code Demo setup on DE10-Lite

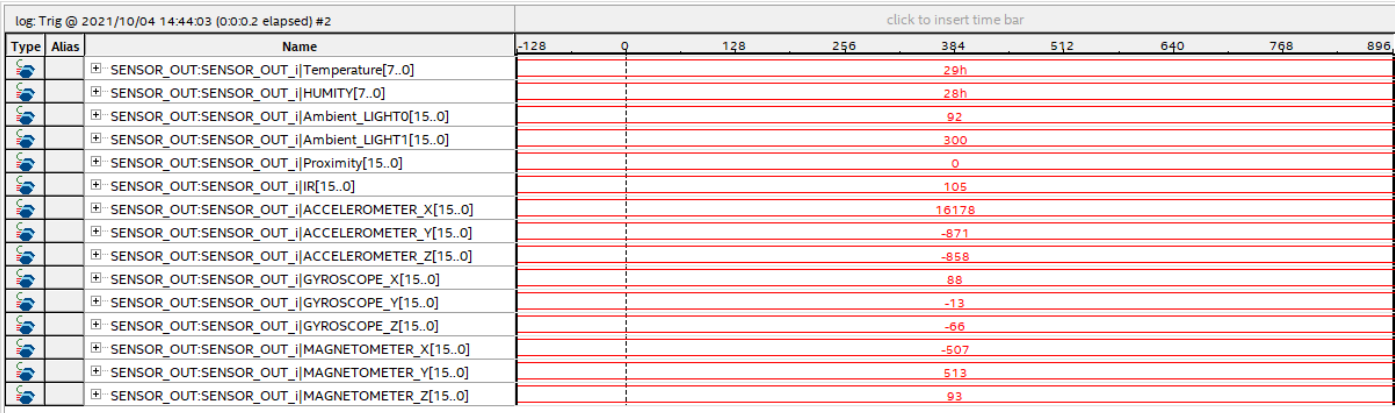


Figure 3-21 Sensor RTL Code Demo setup on DE10-Lite

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration, as shown in **Figure 3-22**.

1. Connect the RFS2 daughter card to the GPIO of DE10-Lite board.
2. Connect the host PC to the USB connector (J3) on DE10-Lite via USB cable.
3. Please make sure Quartus Prime has been installed on the host PC.
4. Execute the batch file " test.bat" under the demo_batch folder of DE10_LITE_RFS2_SENSOR_RTL project.
5. Set SW[3:0] to the corresponding positions, as shown in **Table 3-1**, for the measurement.

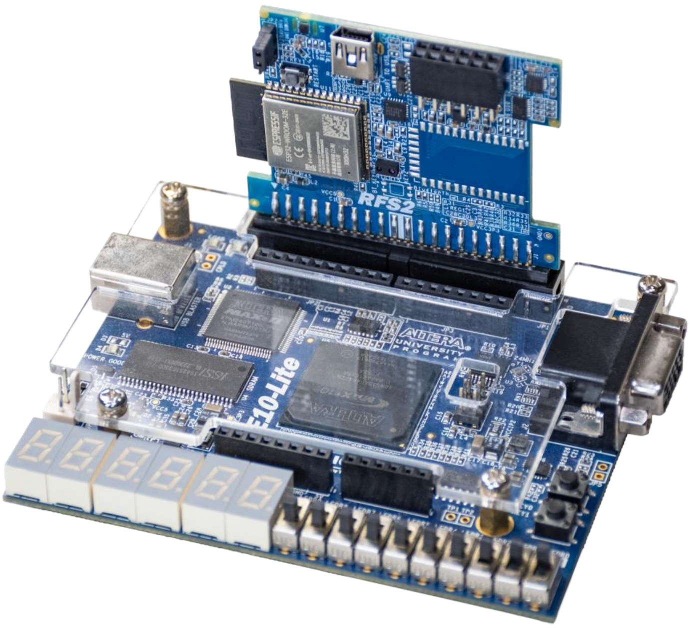


Figure 3-22 Sensor RTL Code Demo setup on DE10-Lite

3.4 UART

This demo will use the UART interface on the RFS2 board to perform a data loopback example with host PC.

User needs to connect the USB to UART port of RFS2 to the host with a USB cable first. When the UART interface on the RFS2 board receives the data transmitted from the host serial terminal program and it will be send to the FPGA, the received data will be processed in the FPGA then return the serial terminal program of the host through the UART interface of RFS2. This can show a simple example of UART transmission and reception.

This demo can be realized on DE10-NANO and DE10-Lite. Please follow the steps below for running the demo on these two boards.

■ Block Diagram

Figure 3-5 shows the function block diagram of UART demonstration

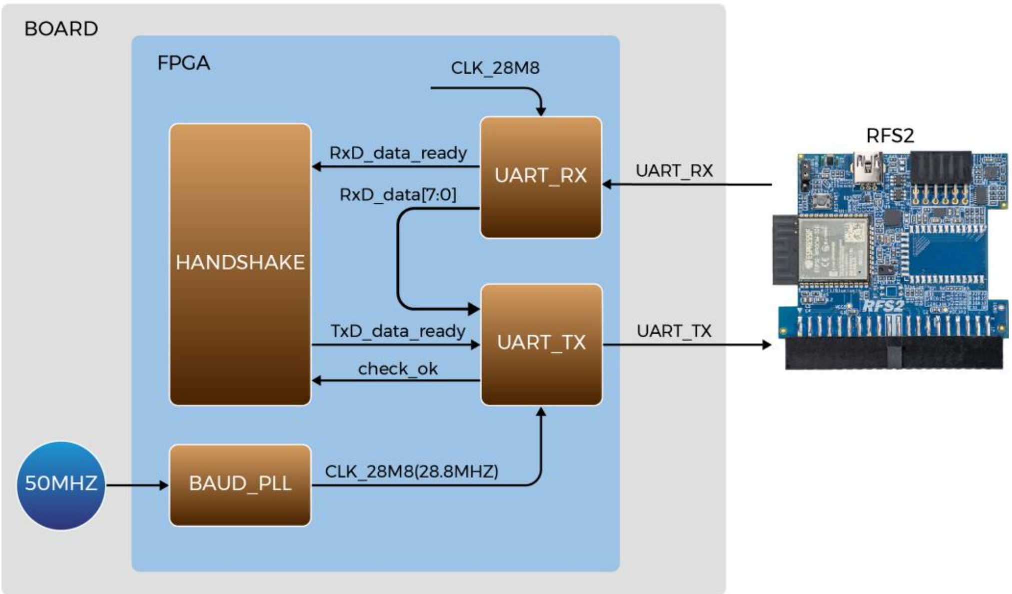


Figure 3-23 Function block diagram of UART

BAUD_PLL: This PLL module can divide the input 50MHz clock into 28.8MHz and output to UART_TX and UART_RX module.

UART_RX: This module will receive data from the TX pin of CP2102N-A02 (USB to UART) IC then convert the receiving 115200 bps serial data into 8-bit parallel data for output.

HANDSHAKE: When the **UART_RX** module successfully receives one byte data, **RxD_data_ready** will send a HI-pulse signal to this module. This module will immediately make **TxD_data_ready** output a HI signal to the **UART_TX** module. Until the check_ok of the UART_TX module becomes a logic high signal. The **TxD_data_ready** will be changed to Low until the **check_ok** pin output logic high level.

UART_TX: This module will convert 8 bit parallel to 115200 bps serial data then send it to the RX pin of the CP2102N-A02 (USB to UART) IC.

■ Quartus Project Inforamtioin

For DE10-NANO Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Nano\DE10_NANO_RFS2_UART_RTL
Demo Batch File	Demonstrations\DE10_Nano\DE10_NANO_RFS2_UART_RTL\demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Lite\DE10_LITE_RFS2_UART_RTL
Demo Batch File	Demonstrations\DE10_Lite\DE10_LITE_RFS2_UART_RTL\demo_batch

■ Demonstration Setup for DE10-NANO

Please follow the procedures below to setup the demonstration, as shown in **Figure 3-24**.

1. Connect the RFS2 daughter card to the GPIO0 of DE10-NANOboard.
2. Plug in 5V DC to DE10-NANO.
3. Connect the host PC to the USB connector (J13) on DE10-NANO via USB cable.
4. Connect the host PC to the USB connector (J2) on RFS2 via USB cable.
5. Please make sure Quartus II has been installed on the host PC.
6. Execute the batch file “ test.bat” under the demo_batch folder of DE10_NANO_RFS2_UART_RTL project.

7. User needs to setup the USB to UART (CP2102N) driver in your host PC, please refer to the this guide (The CP2102N Driver Installation Instructions) to install.
8. Open the serial terminal program and set the COM port number(check the Device Manager of your host PC) and baud rate to 115200 bps. Type any word to the serial terminal program and it will show on the serial terminal program.

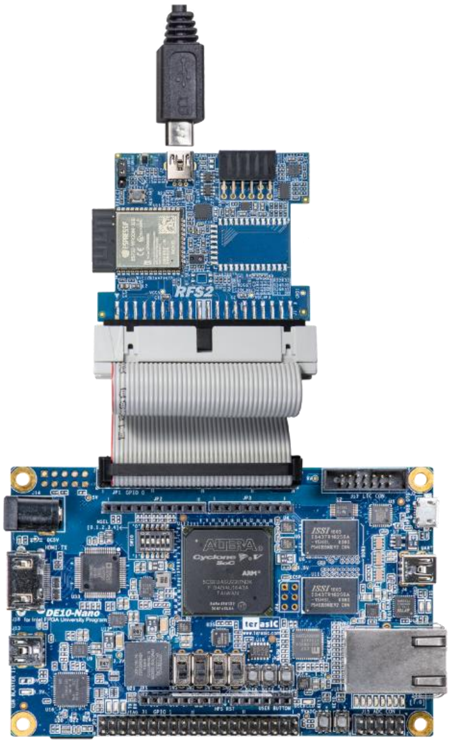


Figure 3-24 Demo setup with DE10-NANO

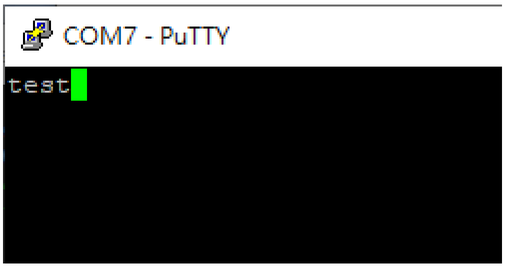


Figure 3-25 Demo setup with DE10-NANO

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration, as shown in **Figure 3-4**.

1. Connect the RFS2 daughter card to the GPIO of DE10-Lite board
2. Connect the host PC to the USB connector (J3) on DE10-Lite via USB cable.
3. Connect the host PC to the USB connector (J2) on RFS2 via USB cable.
4. Please make sure Quartus II has been installed on the host PC.
5. Execute the batch file " test.bat" under the demo_batch folder of DE10_LITE_RFS2_UART_RTL project.
6. User needs to setup the USB to UART (CP2102N) driver in your host PC, please refer to the this guide (The CP2102N Driver Installation Instructions) to install.
7. Open the serial terminal program and set the COM port number(check the Device Manager of your host PC) and baud rate to 115200 bps. Type any word to the serial terminal program and it will show on the serial terminal program.

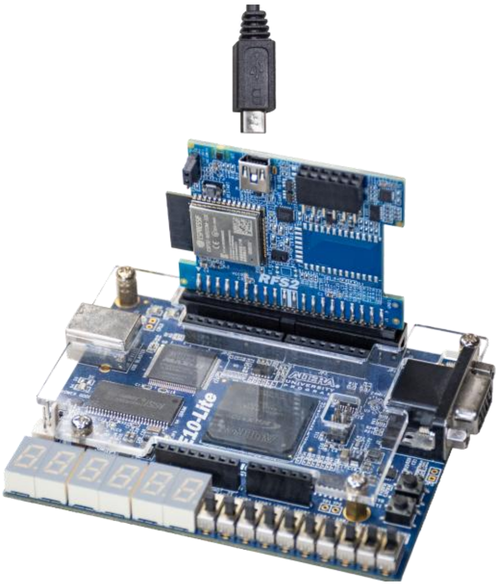


Figure 3-26 Demo setup for DE10-Lite

3.5 Restore ESP32-WROOM-32E Firmware

This section describes how to restore the Terasic Factory Firmware for ESP32-WROMM-32E on RFS2. The Terasic Factory Firmware is developed based on EPS32 firmware V2.2 and adding Classic Bluetooth AT command to support BT SPP Profile. User also can use the same way to update their customer firmware.

The firmware download interface of ESP32-WROOM-32E is the UART0 port. The FPGA is configured to connect the USB-to-UART signal (from mini USB connector (J2) of RFS2) to the UART0 port of ESP32, so user can download firmware from the mini USB connector (J2).

■ Block Diagram

Figure 3-27 shows the function block diagram of FPGA function. EPS32 UART0 TX/RX signaled are direct connector to the USB-to-UART TX/RX signals. LED[1] is used to indicate the ESP32 ENABLE Signal. LED[0] is turn on to indicated the FPGA is configured successfully.

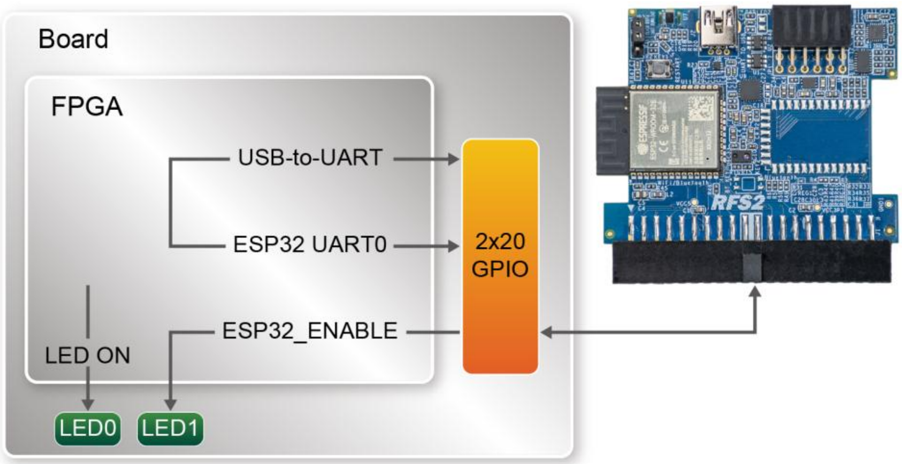


Figure 3-27 Function block diagram of sensor measurement

This demo can be realized on DE10-Lite and DE10-Nano. Please follow the steps below for running the demo on these two boards.

■ Quartus Project Information

For DE10-Nano Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Nano\DE10_Nano_ESP32_FW_Update
Demo Batch File	Demonstrations\DE10_Nano\DE10_Nano_ESP32_FW_ Update\demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Lite\DE10_LITE_ESP32_FW_ Update

Demo Batch File	Demonstrations\DE10_Lite\DE10_LITE_ESP32_FW_Update\demo_batch

■ Firmware and Download Tool

The Terasic Factory Firmware for ESP32-WROOM-32E module is included in the System CD. It locates in the demo_batch folder of this demo project with filename:

Terasic_factory_WROOM-32.bin

The EPS32 firmware download procedure required ESP32 DOWNLAOD TOOL. In this demonstration, Flash Download Tools V3.9.0 is used. The tool can be free download from the Web: https://www.espressif.com/en/support/download/other-tools

■ ESP32 Firmware Download Via DE10-Nano

Please follow the procedures below to perform ESP32 firmware download.

1. Install RFS2 on GPIO-0 2x20 Header of DE10-Nano, as show in **Figure 3-28.**
2. Set JP2 Jumper on RFS2 to position 2-3.
3. Connect the host PC to the Mini USB connector (J13) on DE10-Nano via an Mini USB cable.
4. Connect the host PC to the Mini USB connector (J2) on RFS2 via a Mini USB cable.
5. Execute the batch file " test.bat" under the demo_batch folder to configure FPGA. When configured is done, LED0 and LED1 of the FPGA mainboard are lighten.
6. Execute ESP32 DOWNLOAD TOOL flash_download_tool.exe
7. An ESP32 DOWNLOAD TOOL query dialog appears, as show in **Figure 3-29**. Select "ESP32" chipType, "develop" workMode, and then click "OK" button.
8. When ESP32 DOWNLOAD TOOL GUI appears, specify firmware location, offset address, comport, board rate, and then click "Start" button as shown in **Figure 3-30.**
9. If "SYNC" status appears on GUI, as show in **Figure 3-31**, please press and release the KEY1 button in RFS2.
10. When firmware download is progressing, users will see "DOWNLOAD" status on GUI, as shown in **Figure 3-32**.
11. When firmware download is finished, users will see "FINISHED" status on GUI, as shown in **Figure 3-33**.

12. In this moment, firmware is download successfully. Power off the FPGA mainboard and restore JP2 jumper of RFS2 to position 1-2.

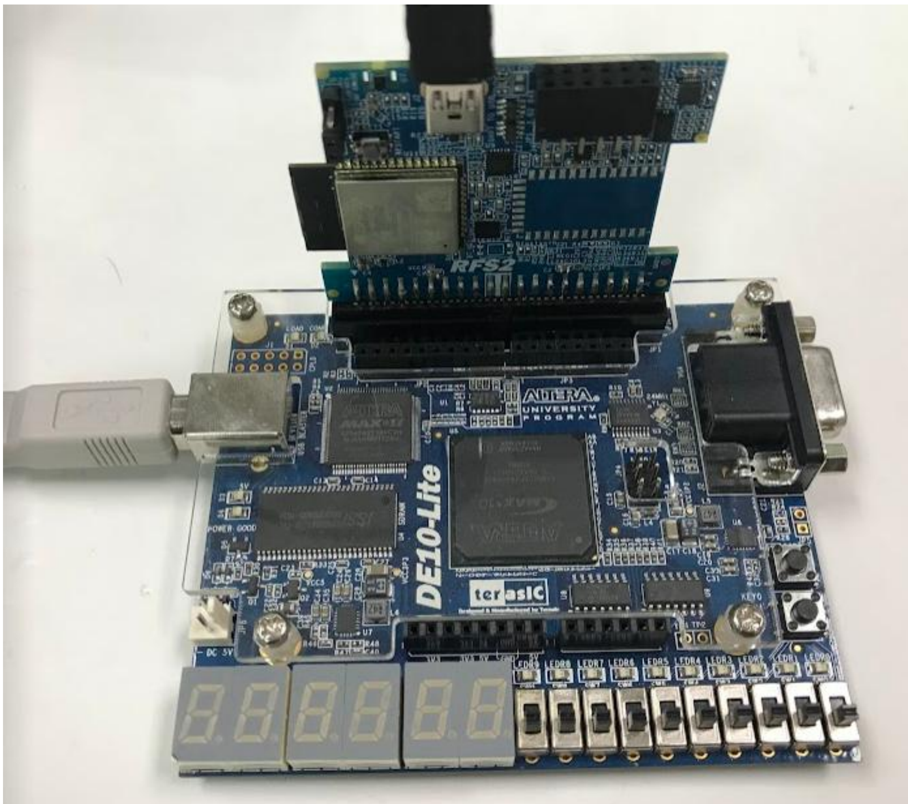


Figure 3-28 Setup RFS2 on DE10-Nano

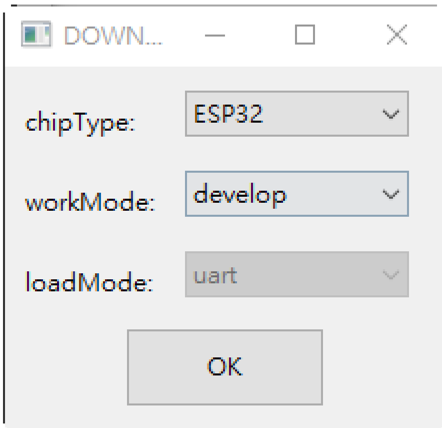


Figure 3-29 ESP32 DOWNLOAD TOOL Query Dialog

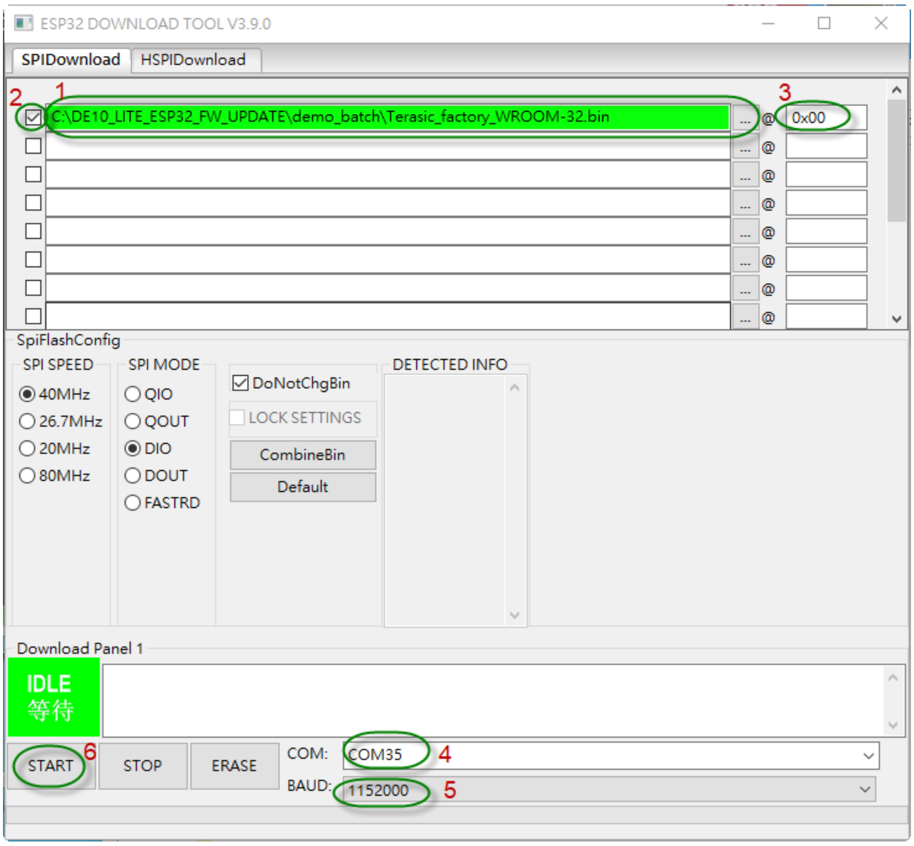


Figure 3-30 ESP32 Download Tool Configure

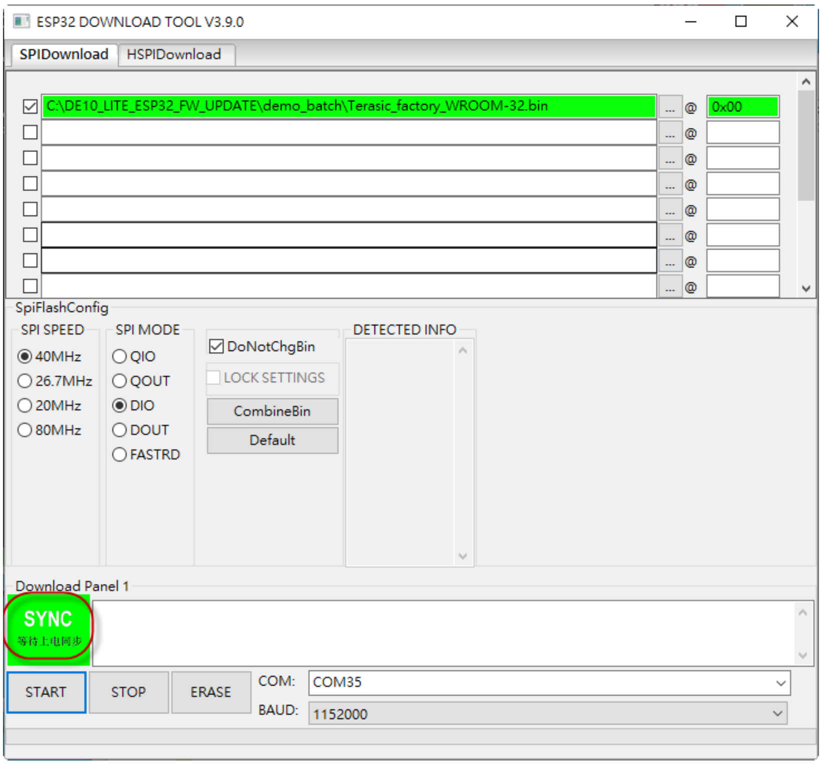


Figure 3-31 Firmware Download in SYNC status

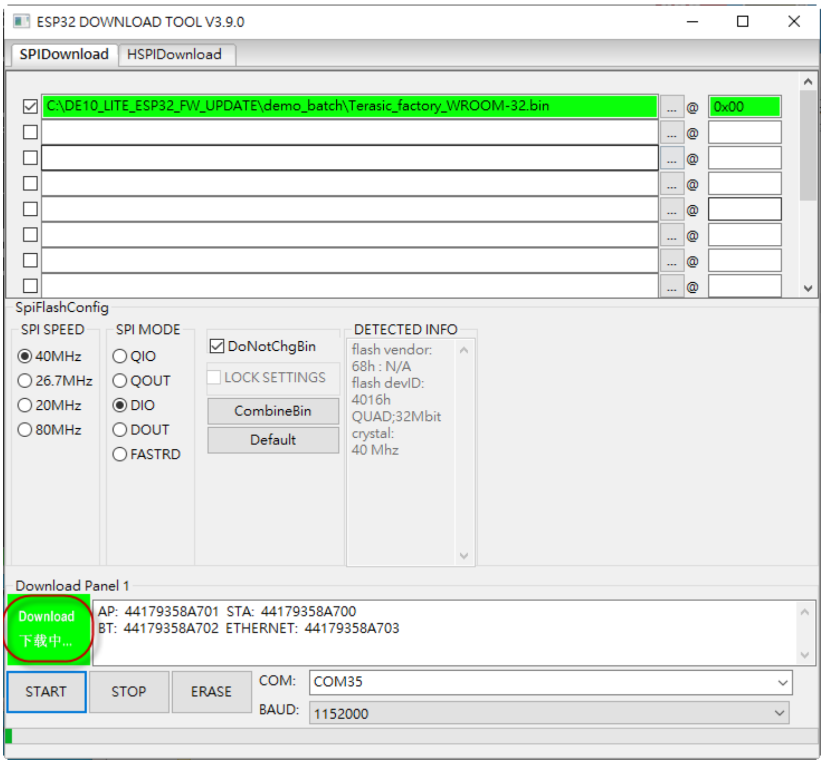


Figure 3-32 Firmware Download in Download Progress Status

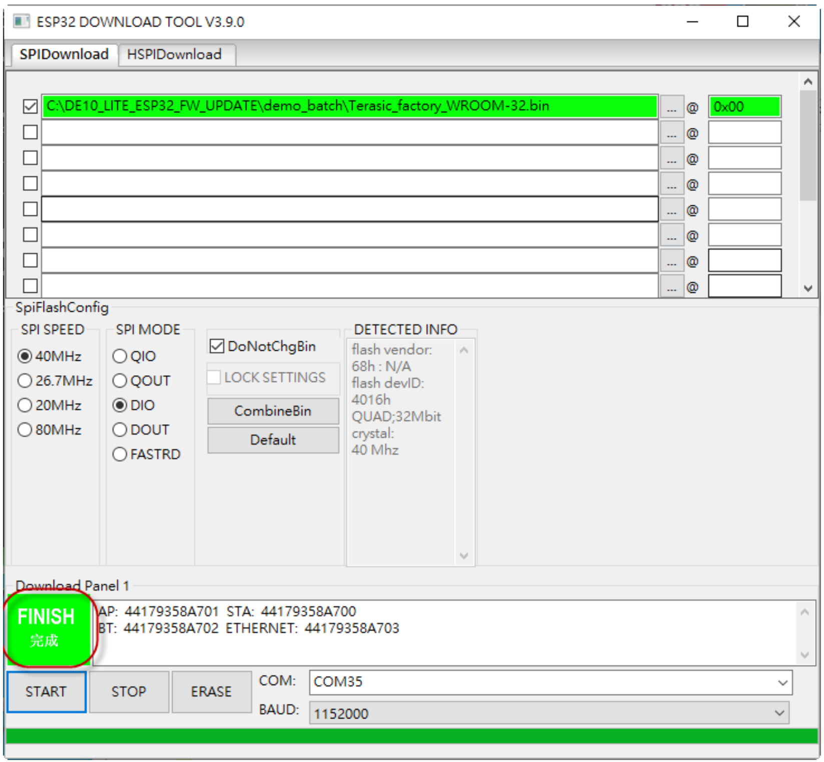


Figure 3-33 Firmware Download in FINISHED Status

■ ESP32 Firmware Download Via DE10-Lite

Please follow the procedures below to perform ESP32 firmware download.

1. Install RFS2 on DE10-Lite, as show in **Figure 3-34.**
2. Set JP2 Jumper on RFS2 to position 2-3.
3. Connect the host PC to the USB connector (J3) on DE10-Lite via an USB cable.
4. Connect the host PC to the Mini USB connector (J2) on RFS2 via a Mini USB cable.
5. Execute the batch file " test.bat" under the demo_batch folder to configure FPGA. When configured is done, LED0 and LED1 of the FPGA mainboard are lighten.
6. Execute ESP32 DOWNLOAD TOOL flash_download_tool.exe
7. An ESP32 DOWNLOAD TOOL query dialog appears, as show in **Figure 3-29**. Select "ESP32" chipType, "develop" workMode, and then click "OK" button.
8. When ESP32 DOWNLOAD TOOL GUI appears, specify firmware location, offset address, comport, board rate, and then click "Start" button as shown in **Figure 3-30.**

9. If “SYNC” status appears on GUI, as show in **Figure 3-31**, please press and release the KEY1 button in RFS2.
10. When firmware download is progressing, users will see “DOWNLOAD” status on GUI, as shown in **Figure 3-32**.
11. When firmware download is finished, users will see “FINISHED” status on GUI, as shown in **Figure 3-33**.
12. In this moment, firmware is download successfully. Power off the FPGA mainboard and restore JP2 jumper of RFS2 to position 1-2.

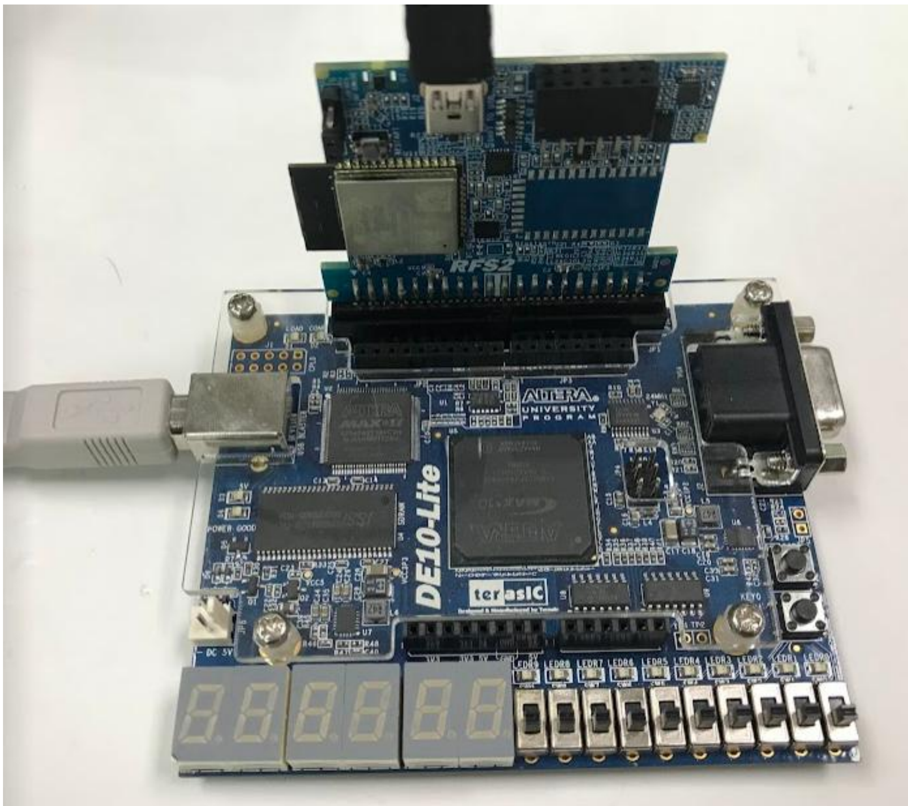


Figure 3-34 Setup RFS2 on DE10-Lite

Chapter 4

Nios II Based Example Designs

This chapter will demonstrate how the FPGA to communicate with the RFS2 daughter card based on Nios II processor.

4.1 Wi-Fi Client

This demonstration shows how clients use timing web server and query the internet time through the Wi-Fi function of the ESP32-WROOM-32E module on the RFS2 daughter card. The time information will be shown on nios2-terminal and the 7-segment display with the format HH:MM:SS if 7-segement displays are avaialbe.

In this demonstration, a Wi-Fi AP(Access Point) is required so the demo program can connect to the internet through the Wi-Fi module. Before accessing the Wi-Fi AP, users need to input the SSID and password for the Wi-Fi AP.

■ Block Diagram

Figure 4-1 shows the function block diagram of the Wi-Fi Client demonstration. The **UART Controller** is used to communicate with the 4-pin **UART1** port on the ESP32-WROOM-32 module on RFS2 daughter card. Through the UART1 interface, Nios II processor communicates with the ESP32-WROOM-32E module via AT command defined by ESP32-WROOM-32E. Besides, the 2-pin **UART0** port on ESP32-WROOM-32E module is connected to the RFS2 on-board UART-TO-USB port in RTL code, so users can see the firmware log if debug is required.

The **7-Seg Controller** is used to control the six 7-segment to display time information. These controllers are controlled by the Nios II processor through the Avalon memory-mapped bus. The Nios II program is running on on-chip memory.

First, the Nios II program sends “http get” request to the timing web server through the **UART Controller**. Then, the Nios II program receives responded data from the timing web server. Finally, the Nios II program parsing the responded data to extract the timing information, and display the information on the 7-segment display.

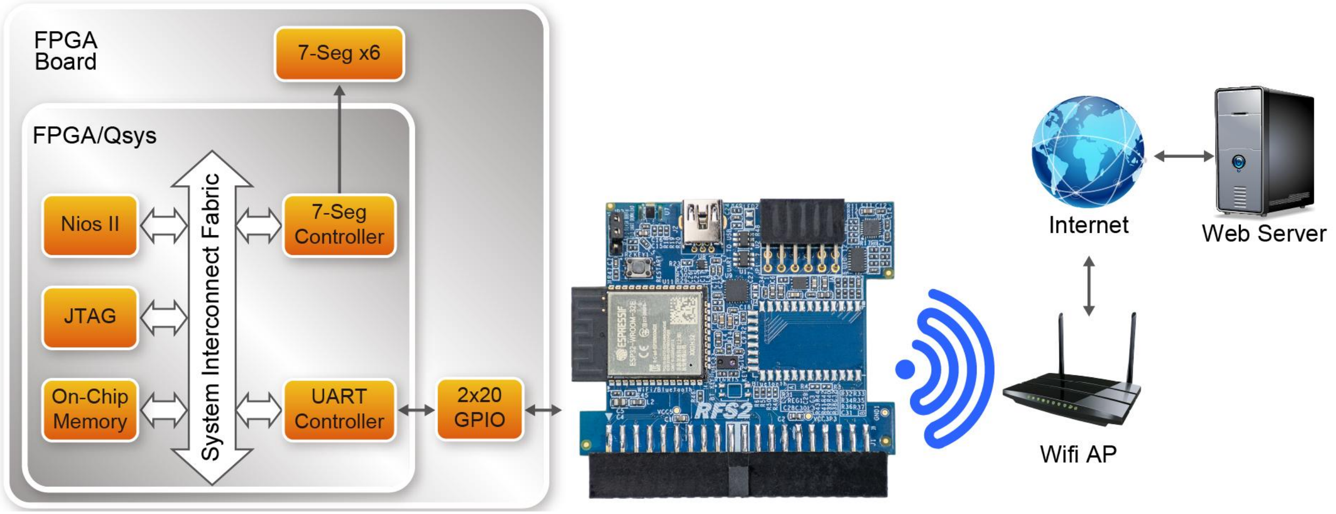


Figure 4-1 WiFi Client Demo Function Block Diagram

How to Query Internet Time

The internet time information is available on the web http://demo.terasic.com/. Sending URL http://demo.terasic.com/time/ to the web server, it will response current time in format HH:MM:SS.

Quartus Project Information

For DE10-Nano Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Nano\DE10_Nano_WiFi_Network_Time
Demo Batch File	Demonstrations\DE10_Nano\DE10_Nano_WiFi_Network_Time\demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Lite\DE10_LITE_WiFi_Network_Time
Demo Batch File	Demonstrations\DE10_Lite\DE10_LITE_WiFi_Network_Time\demo_batch

Demonstration Setup for DE10-Nano

Please follow the procedures below to setup the demonstration as shown in **Figure 4-2**.

1. Connect the RFS2 daughter card to the GPIO-0 connector of the DE10-Nano board.
2. Make sure the jumper is located at pin 1-2 on the JP2 of RFS2 daughter card.

3. Connect a USB cable between the host PC and the USB connector (J13) on the DE10-Nano.
4. Execute the batch file "test.bat" under the demo_batch folder of DE10_Nano_WiFi_Network_Time project. A Nios II terminal will appear and list nearby WiFi AP.
5. Input SSID of your WiFi AP according to the prompt in Nios II terminal. The terminal will not display the SSID.
6. Input password of your WiFi AP according the prompt in the Nios II terminal. The terminal will not display the password.
7. The UTC(Universal Time Coordinated) time will be display on the nios2-terminal as shown in **Figure 4-3**.
8. Optional, users can use a USB cable to connector the Mini USB Connector (J2) on RFS2 to your PC to see the ESP32-WROOM-32E firmware log.

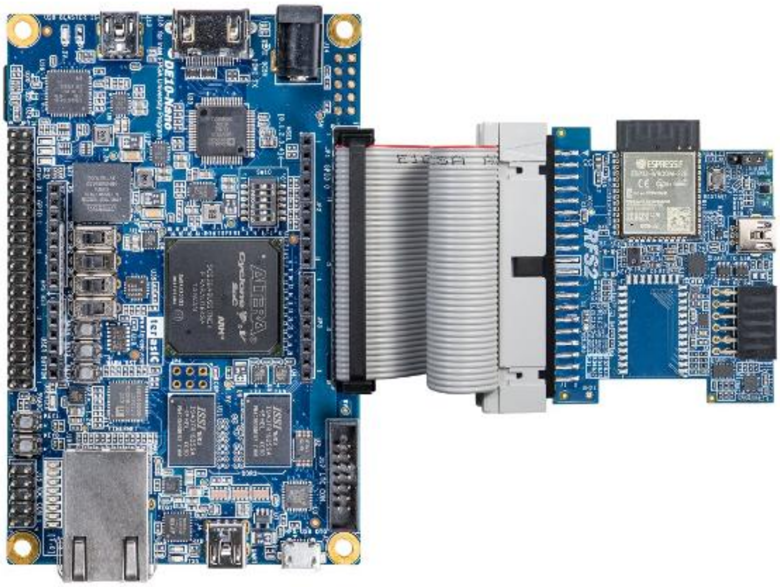


Figure 4-2 Setup WiFi Client Demo on DE10-Nano

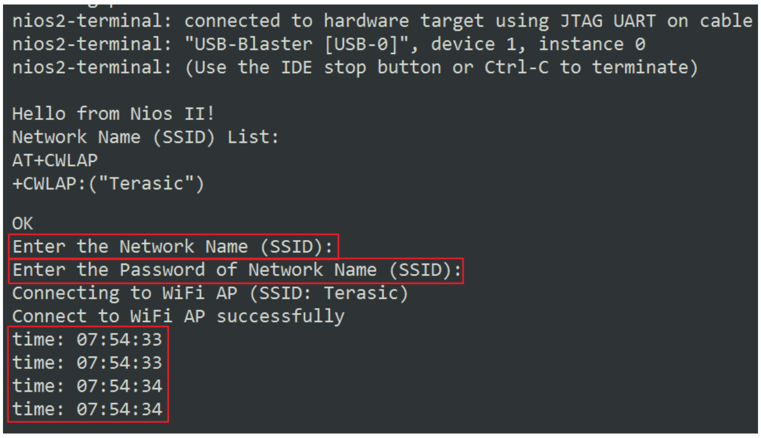


Figure 4-3 WiFi Client Demo Display UTC Time

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration as shown in **Figure 4-4**.

1. Connect the RFS2 daughter card to the GPIO connector of the DE10-Lite board.
2. Make sure the jumper is located at pin 1-2 on the JP2 of RFS2 daughter card.
3. Connect a USB cable between the host PC and the USB connector (J3) on the DE10-Lite.
4. Execute the batch file “test.bat” under the demo_batch folder of DE10_LITE_WiFi_Network_Time project. A Nios II terminal will appear and list nearby WiFi AP
5. Input SSID of your WiFi AP according to the prompt in Nios II terminal. The terminal will not display the SSID.
6. Input password of your WiFi AP according the prompt in the Nios II terminal. The terminal will not display the password.
7. The UTC(Universal Time Coordinated) time will be display on the six 7-segement as shown in **Figure 4-5**.
8. Optional, users can use a USB cable to connector the Mini USB Connector (J2) on RFS2 to your PC to see the ESP32-WROOM-32E firmware log.

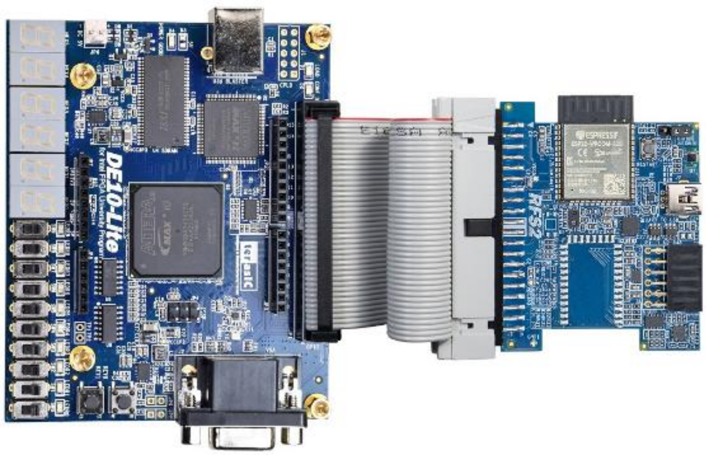


Figure 4-4 Setup WiFi Client Demo on DE10-Lite

Figure 4-5 UTC time display on 7-segment display

4.2 Wi-Fi Server

This demonstration shows how to implement a simple Wi-Fi web server in such a way that users can directly use a mobile device to visit the web server. In the mobile device, a web server is used to client the web server. The web server provides the LED control interface as shown in **Figure 4-6**. Users can click the blue led text to lighten/un-lighten the LEDs on the FPGA main board.

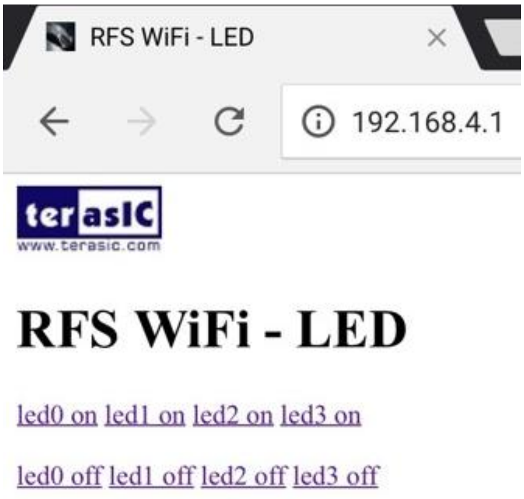


Figure 4-6 LED Control Web

■ Block Diagram

Figure 4-7 shows the Function block diagram of the Wi-Fi Server demonstration. The UART controller is used to communicate with communicate with the 4-pin UART1 port on the ESP32-WROOM-32 module on RFS2 daughter card. Through the UART1 interface, Nios II processor communicates with the ESP32-WROOM-32E module via AT command defined by ESP32-WROOM-32E. Besides, the 2-pin UART0 port on ESP32-WROOM-32E module is connected to the RFS2 on-board UART-TO-USB port in RTL code, so users can see the firmware log if debug is required.

The User Interface module is used to control the LED. When Web server is launched, it sends the AT command to the Wi-Fi module to configure it as a Soft AP (Access Point), and wait for connection request from the client. When a connection is established, the Web Server module is response for receiving the request command from the web browser, parsing the command and performing the associated LED control, then sends the responding result to the web browser.

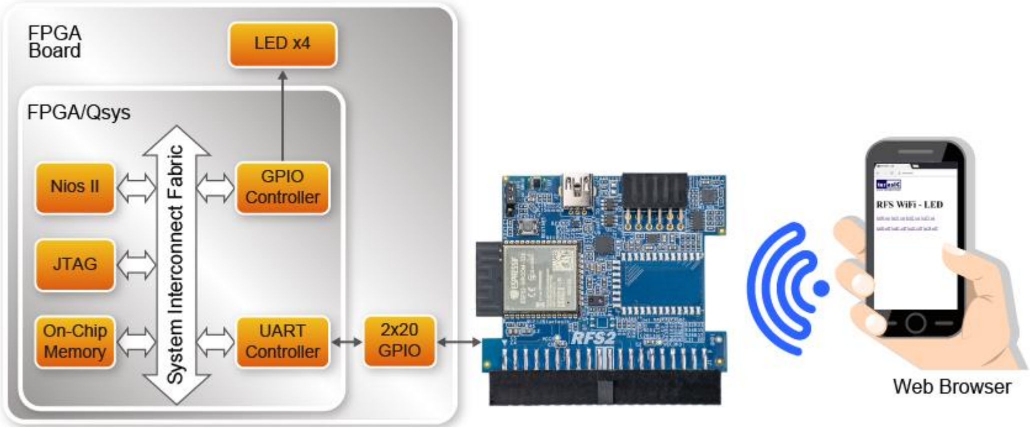


Figure 4-7 WiFi Server Demo Function Block Diagram

■ Control Flow and Message

HTTP is used in this web server demonstration. It is a request-response protocol in the client-server computing model. For details, please refer to:

https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol

Here is the control flow for the web server demo:

1. Waiting for a request message from the mobile device
2. Parsing the request message. In this simple demo, parsing the first line (a request line) in the message is enough.
3. Transmit the response message to the mobile device according to the parsing result. In this demonstration, there are five valid passing results.

Below shows an example request message that should be received from the web server. In this simple demonstration, parsing the first line string (a request line) in the message is enough. This means the Nios II program only needs to parse the "GET /led/on/3 HTTP/1.1" string in this request message.

```
GET /led/on/3 HTTP/1.1
Host: 192.168.4.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-tw
Connection: keep-alive
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/5.0 (iPad; CPU OS 9_3_2 like Mac OS X) AppleWebKit/601.1 (KHTML, like
Gecko) CriOS/55.0.2883.79 Mobile/13F69 Safari/601.1.46
```

In this demonstration, the request message is classified as five types according to the requested resource. The request message type can be determined by only parsing the first line string in the request message. For example, the first line string "GET /led/on/3 HTTP/1.1" is classified as "/led" resource type. **Table 4-1** shows the associated response message for each request type. The response message is consistent of three parts: status code, response header and body message. For the "/led" request, the response status code is "302 Found" which means re-directing to the resource location. The response header is "Location:/" it represents the new location for the request resource. The response message body is empty.

Table 4-1 FPGA Configuration Mode Switch (SW10)

Resource type	***Response***		
	Status Code	***Response Header***	***Body Message***
/	**200 OK**	**Content-Type: text/html**	**web_src/index.html**
/favicon.ico	**200 OK**	**Content-Type: text/plain**	**web_src/favicon.ico**
/Logo_Terasic.jpg	**200 OK**	**Content-Type: image/jpeg**	**web_src/Logo_Terasic.jpg**
/led	**302 Found**	**Location: /**	**---**
Other	**404 Not Found**	**Content-Type: text/html**	**web_src/404.html**

■ Quartus Project

For DE10-Nano Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Nano\DE10_Nano_WiFi_Web_Server
Demo Batch File	Demonstrations\DE10_Nano\DE10_Nano_WiFi_Web_Server\demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Lite\DE10_LITE_WiFi_Web_Server
Demo Batch File	Demonstrations\DE10_Lite\DE10_LITE_WiFi_Web_Server\demo_batch

■ Demonstration Setup for DE10-Nano

Please follow the procedures below to setup the demonstration as shown in **Figure 4-8.**

1. Connect the RFS2 daughter card to the GPIO-0 connector of the DE10-Nano board.
2. Make sure the jumper is located at pin 1-2 on the JP2 of RFS2 daughter card.
3. Connect a USB cable between the host PC and the USB connector (J13) on the DE10-Nano.
4. Execute the batch file "test.bat" under the demo_batch folder of DE10_Nano_WiFi_Web_Server project. The nios2-terminal will show 'Server has started to listen at port: 80' as shown in **Figure 4-9**.

5. Use a mobile device to search the nearby Wi-Fi device.
6. Connect the found SSID device “Terasic_RFS” with password: “1234567890”.
7. In the mobile device, launch a web browser to connect http://192.168.4.1.
8. Click led_on and led_off the control the LED on FPGA as shown in **Figure 4-6**.
9. Optional, users can use a USB cable to connector the Mini USB Connector (J2) on RFS2 to your PC to see the ESP32-WROOM-32E firmware log.

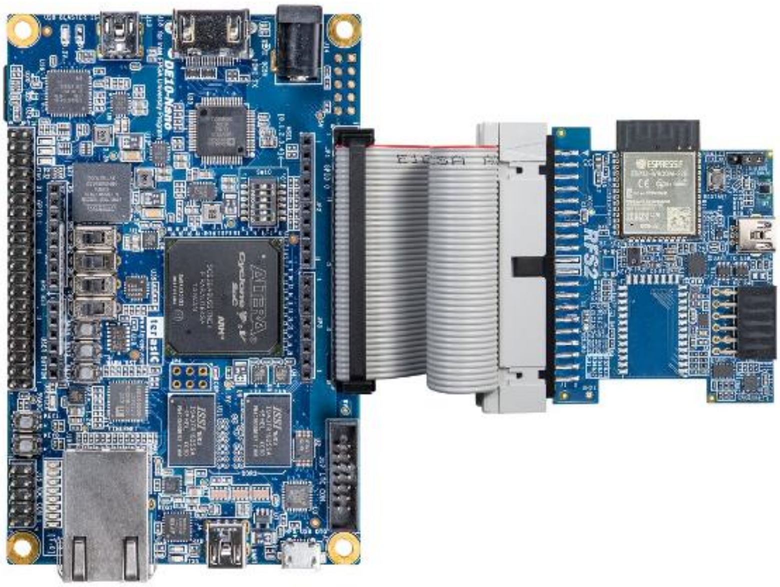


Figure 4-8 Setup WiFi Web Server Demo on DE10-Nano

```
Using cable "USB-Blaster [USB-0]", device 1, instance 0x00
Resetting and pausing target processor: OK
Initializing CPU cache (if present)
OK
Downloaded 129KB in 1.7s (75.8KB/s)
Verified OK
Waiting to allow other programs to start: done
Starting processor at address 0x00040238
nios2-terminal: connected to hardware target using JTAG UART on cable
nios2-terminal: "USB-Blaster [USB-0]", device 1, instance 0
nios2-terminal: (Use the IDE stop button or Ctrl-C to terminate)

Hello from Nios II!
Server has started to listen at port: 80.
```

Figure 4-9 WiFi Web Server listen at port 80

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration as shown in **Figure 4-10**.

1. Connect the RFS2 daughter card to the GPIO connector of the DE10-Lite board.
2. Make sure the jumper is located at pin 1-2 on the JP2 of RFS2 daughter card.
3. Connect a USB cable between the host PC and the USB connector (J3) on the DE10-Lite.
4. Execute the batch file "test.bat" under the demo_batch folder of DE10_LITE_WiFi_Web_Server project. The nios2-terminal will show 'Server has started to listen at port: 80' as shown in **Figure 4-9**.
5. Use a mobile device to search the nearby Wi-Fi device.
6. Connect the found SSID device "Terasic_RFS" with password: "1234567890".
7. In the mobile device, launch a web browser to connect http://192.168.4.1.
8. Click led_on and led_off the control the LED on FPGA as shown in **Figure 4-6**.
9. Optional, users can use a USB cable to connector the Mini USB Connector (J2) on RFS2 to your PC to see the ESP32-WROOM-32E firmware log.

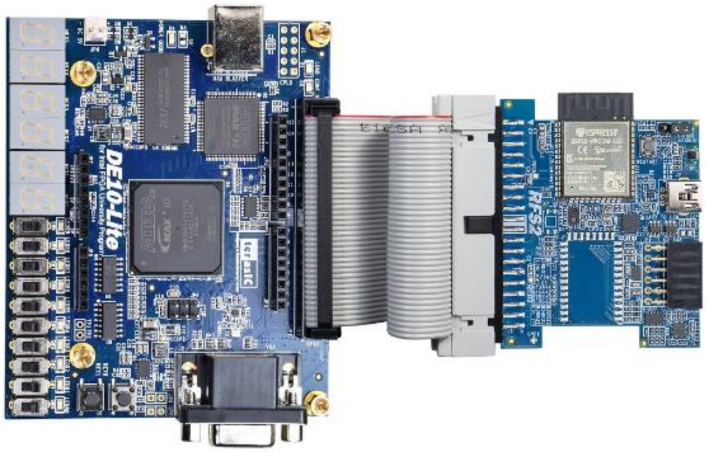


Figure 4-10 Setup WiFi Web Server Demo on DE10-Lite

4.3 Bluetooth Classic SPP

This demonstration will use two RFS2 cards and two FPGA motherboards for implementing Bluetooth master and slave. The NIOS CPU in the FPGA board will use AT command to set the ESP32 module on the RFS2 card to be SPP mater mode and SPP slave mode respectively. The SPP master board will first search for the SPP slave board and pair it. When the pairing is successful, the SPP master board will send a string to SPP slave board via Bluetooth protocol. After the slave board receiving the string, it will be sent back to the master board. The master board will process the received string (add number in the string for indicating the number of transmission times) and then send it to the slave board again. Such transmission will continue to verify that the transmission at both sides is normal.

■ Quartus Project

For DE10-Nano Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Nano\DE10_NANO_RFS2_Classic_SPP
Demo Batch File	Demonstrations\DE10_Nano\DE10_NANO_RFS2_Classic_SPP \demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Lite\LITE _RFS2_Classic_SPP
Demo Batch File	Demonstrations\DE10_Lite\DE10_LITE_RFS2_Classic_SPP \demo_batch

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration..

1. Prepare two DE10-Lite boards and two RFS2 cards and plug the RFS2 into the GPIO header on the DE10-Lite.
2. See **Figure 4-11**, using USB cable to connect both the DE10-Lite boards and host PC. **Note, due to the demo batch setting, please connect both the DE10-Lite boards to one host PC.**

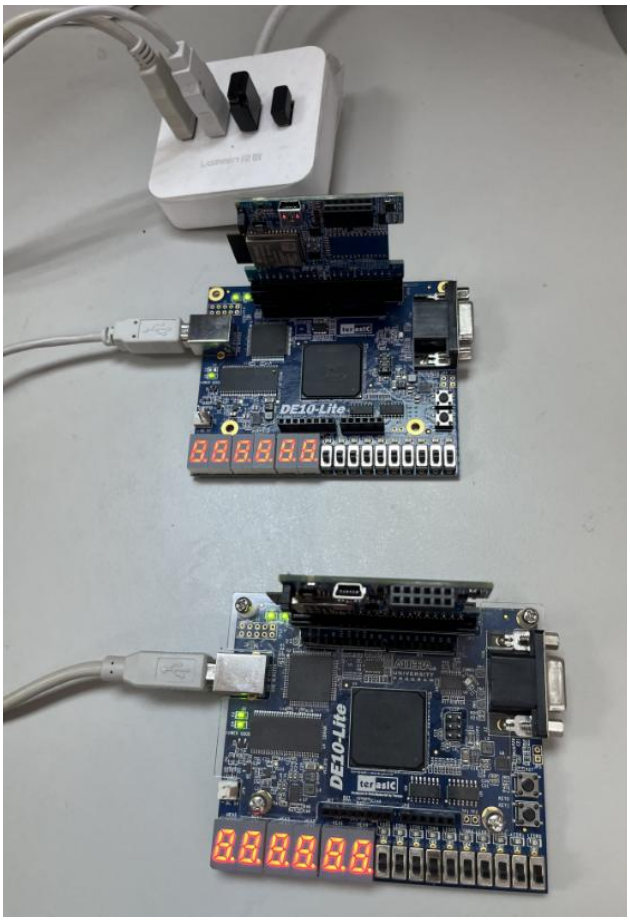


Figure 4-11 Setup the classic SPP Demo

3. Execute the batch file “ test.bat” under the **demo_batch\Classic_SPP_Slave** folder of DE10_LITE_RFS2_Classic_SPP project. This process can down load the SPP slave deign to one of the FPGA on the DE10-Lite board and configure the ESP32 module to run the SPP slave mode.
4. Make sure the SPP slave AT command had been send into the ESP32 module (See **Figure 4-12**).

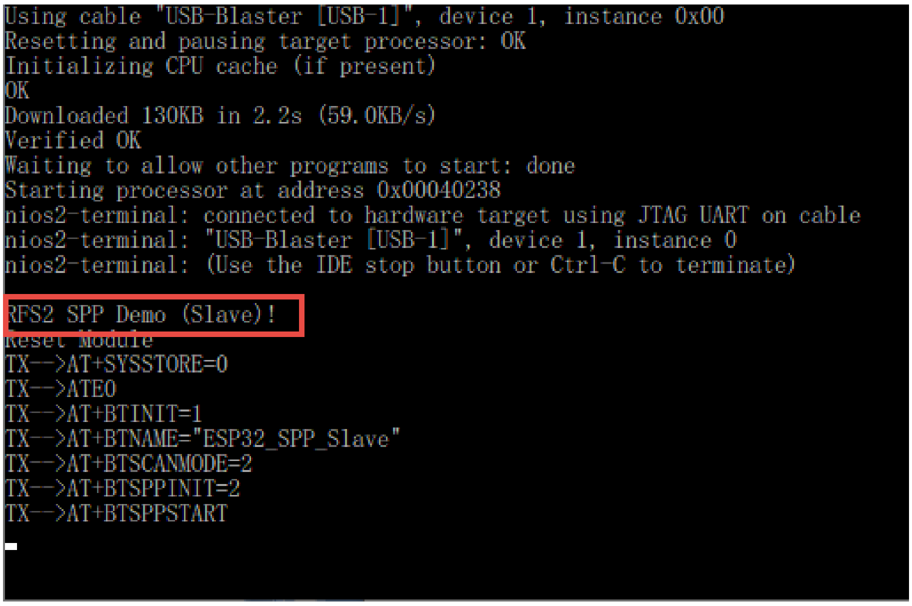


Figure 4-12 Set the slave SPP AT command

5. After the SPP slave board is setting done, execute the batch file " test.bat" under the **demo_batch\Classic_SPP_Master** folder of DE10_LITE_RFS2_Classic_SPP project. This process can down load the SPP mater deign to **the other FPGA of DE10-Lite board** and configure the ESP32 module to run the SPP master mode.
6. The SPP master AT command will be configure to the ESP32 module on the another DE10-Lite board (see **Figure 4-13**).

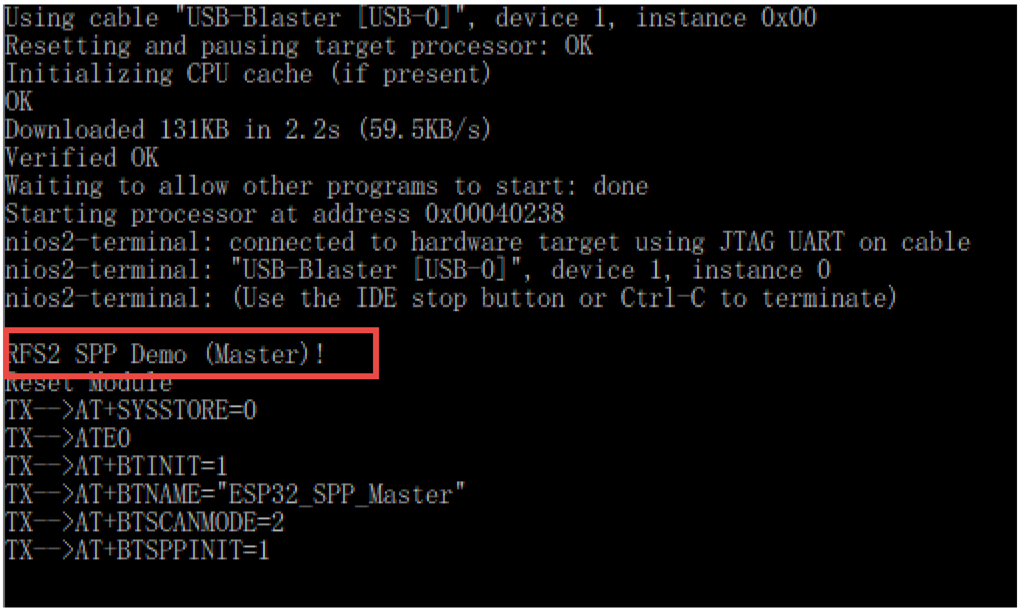


Figure 4-13 Set the master SPP AT command

7. The SPP master board will search the SPP slave board named “ESP_SPP_Slave” and pair the slave automatically. When the pairing is successful, the SPP master board will send a string : ***[0]Loopback*** to SPP slave board. When the slave board receives the string, it will send it back to the master board. User can observed data sending process from the both the master and slave board’s NIOS terminal as shown in **Figure 4-14**.

Figure 4-14 Transmit the string between master and slave

8. When the master board received the string ***[0]LoopBack*** send from the slave board, it will add “1” number for the string for example from “***[0] LoopBack***” to “***[1] LoopBack***”. And the modified string will be send to slave board. Then the boards on both sides will repeat step 7~8 to send data back and forth as shown in **Figure 4-15**.

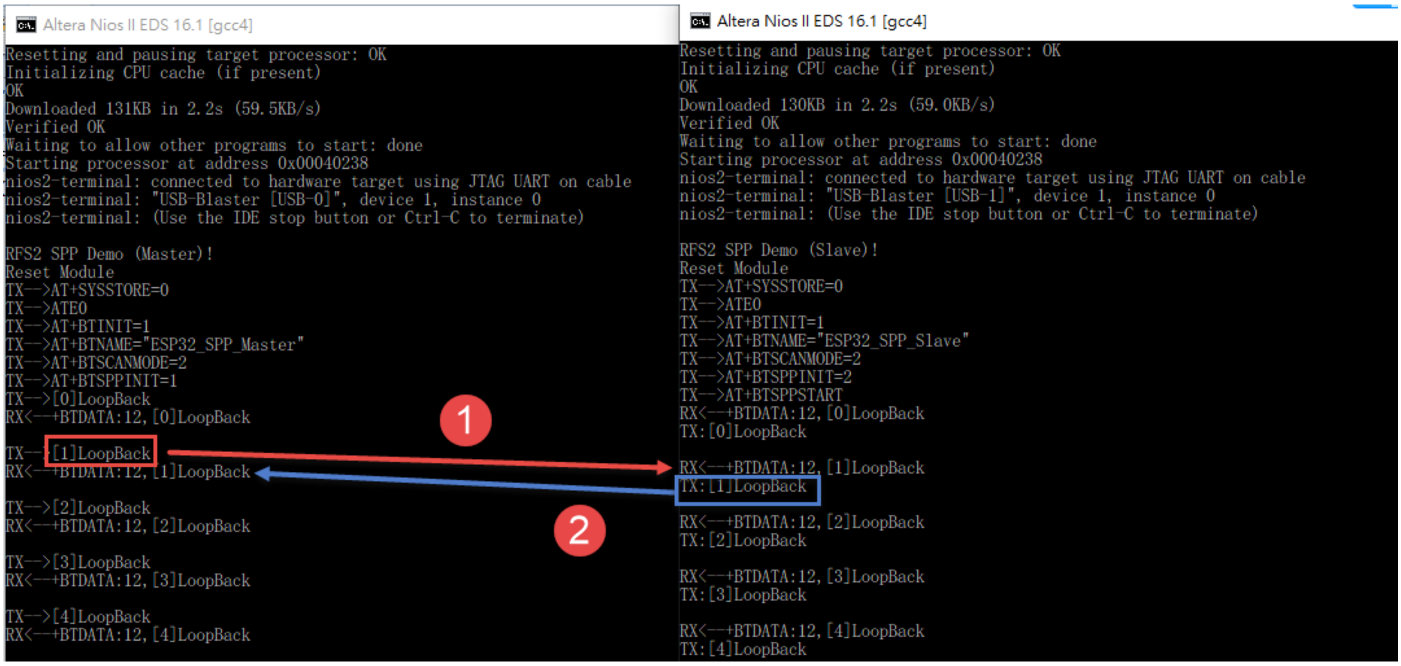


Figure 4-15 Transmit the string between master and slave

■ Demonstration Setup for DE10-Nano

Please follow the procedures below to setup the demonstration as shown in **Figure 4-16.**

1. Prepare two DE10-Nano boards and two RFS2 cards and plug the RFS2 into the GPIO-0 connector on the DE10-Nano board.
2. See **Figure 4-16**, using USB cable to connect both the DE10-Nano boards and host PC. **Note, due to the demo batch setting, please connect both the DE10-Nano boards to one host PC**.
3. Execute the batch file “ test.bat” under the demo_batch\Classic_SPP_Slave folder of DE10_NANO _RFS2_Classic_SPP project. This process can down load the SPP slave deign to one of the FPGA on the DE10-Lite board and configure the ESP32 module to run the SPP slave mode.
4. Make sure the SPP slave AT command had been send into the ESP32 module.
5. After the SPP slave board is setting done, execute the batch file “ test.bat” under the demo_batch\Classic_SPP_Master folder of DE10_NANO _RFS2_Classic_SPP project. This process can down load the SPP mater deign to the other FPGA of DE10-Nano board and configure the ESP32 module to run the SPP master mode.
6. The SPP master AT command will be configure to the ESP32 module on the another DE10-Nano board.

7. The SPP master board will search the SPP slave board named “ESP_SPP_Slave” and pair the slave automatically. When the pairing is successful, the SPP master board will send a string ***[0]Loopback*** to SPP slave board. When the slave board receives the string, it will send it back to the master board. User can observed data sending process from the both the master and slave board’s NIOS terminal.
8. When the master board received the string ***[0]Loopback*** send from the slave board, it will add “1” number for the string for example from “ ***[0]Loopback***” to “ ***[1]Loopback***”. And the modified string will be send to slave board. Then the boards on both sides will repeat step 7~8 to send data back and forth.

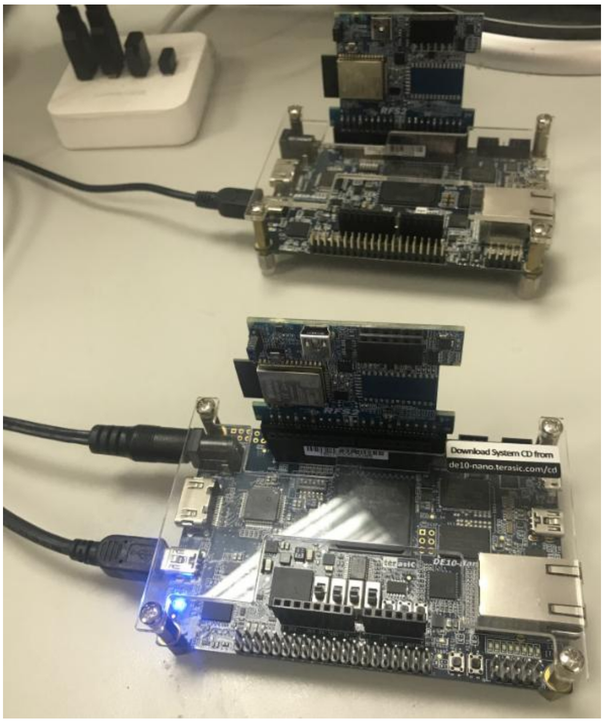


Figure 4-16 Setup Demo on DE10-Nano board

4.4 Sensor Measurement

The RFS2 is equipped with three sensors; an ambient light sensor, a humidity and temperature sensor, and a 9-axis motion tracking sensor. The APM-16D24 is a light-to-digital ambient light photo sensor whose intensity converts light into a digital signal output capable of I2C interface. The HDC1000 is a fully integrated humidity and temperature sensor, providing excellent measurement accuracy and long term stability, whose measurement results can be read out through the I2C interface. The ICM-20948 consists of two dies; one die houses the 3-axis gyroscope and 3-axis accelerometer, and the other die houses the 3-axis magnetometer. Similarly, the ICM-20948 provides complete 9-axis output through the I2C interface.

■ Block Diagram

Figure 4-17 shows the function block diagram of the RFS2 three-sensor demonstration. The system requires a 50 MHz clock input from the board. Three I2C_OPENCORES controllers are used to communicate with the APM-16D24, HDC1000 and ICM-20948 chips, respectively. The Nios II processor is used to configure the sensors, read the measured values, and show the measured values on the Nios II terminal. The Nios II processor communicates to the sensor through the I2C_OPENCORES I2C controllers. The NIOS II program is running on the on-chip memory.

The I2C_OPNECORES IP RTL source code is located in the folder:

*Boardname*_RFS2_SENSOR/ip/i2c_opencores

The driver for the I2C_OPNECORES IP is implemented in the IC2_core.c and I2C_core.h which located in the folder:

*Boardname*_RFS2_SENSOR/software/RFS2_SENSOR

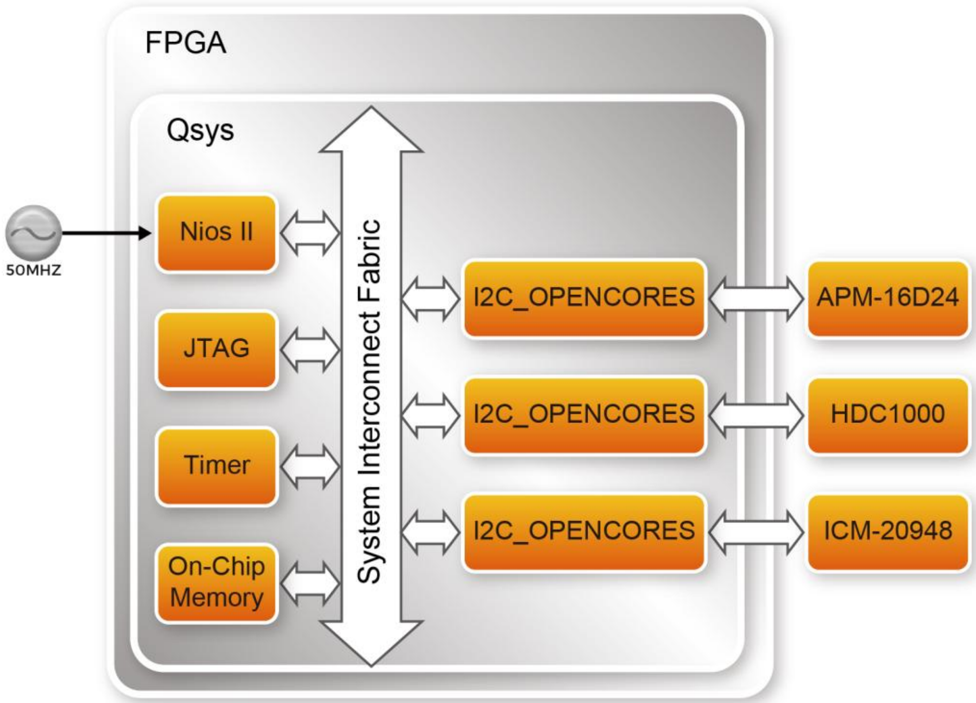


Figure 4-17 Block diagram of the DE10-Lite RFS2 Sensors demo

Sensor Explanation

The APM-16D24 contains two integrated analog-to-digital converters (human eye and clear) that integrate the currents from channel 0 and channel 1 photodiodes. Upon completion of the conversion cycle, the conversion result is transferred to channel 0 and channel 1 data register respectively. The device I2C address of APM-16D24 is 0x70/0x71.

The HDC1000 has three registers: temperature, humidity and configuration registers respectively. It can perform a measurement of both humidity and temperature, or humidity only or temperature only, which can be set in the configuration register. Please note the device I2C address of HDC1000 is 0x80/0x81.

The ICM-20948 chip features three 16-bit analog-to-digital converters (ADCs) for digitizing the gyroscope outputs, three 16-bit ADCs for digitizing the accelerometer outputs, and three 16-bit ADCs for digitizing the magnetometer outputs. The MPU9250 has a user-programmable gyroscope full-scale range of ±250, ±500, ±1000, or ±2000 degrees per second (dps), a user-programmable accelerometer full-scale range of ±2g, ±4g, ±8g, and ±16g, and a magnetometer full-scale range of ±4800μT. Please note the device I2C address of the ICM-20948 is decided by the Pin MPU_AD0_SDO, when the MPU_AD0_SDO is tied to low, the device I2C address is 0xD0/0xD1, and the MPU_AD0_SDO tied to high, the device I2C address is 0xD2/0xD3.

Project Information

For DE10-Nano Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Nano\DE10_NANO_RFS2_SENSOR
Demo Batch File	Demonstrations\DE10_Nano\DE10_L NANO_RFS2_SENSOR \demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V20.1.1
Project Directory	Demonstrations\DE10_Lite\DE10_LITE_RFS2_SENSOR
Demo Batch File	Demonstrations\DE10_Lite\DE10_LITE_RFS2_SENSOR \demo_batch

Demonstration Setup for DE10-Nano

Please follow the procedures below to setup the demonstration as shown in **Figure 4-18**.

1. Make sure Quartus Prime v16.1 or later is installed on your PC.
2. Connect the RFS2 daughter card to the 2x20 GPIO-0 connector on the DE10-Nano board.
3. Connect a USB cable between the host PC and the USB connector (J13) on the DE10-Nano.
4. Execute the batch file " test.bat" under the demo_batch folder of DE10_NANO_RFS2_SENSOR project.
5. The Nios II terminal will show the measured sensor value as shown in **Figure 4-19.**

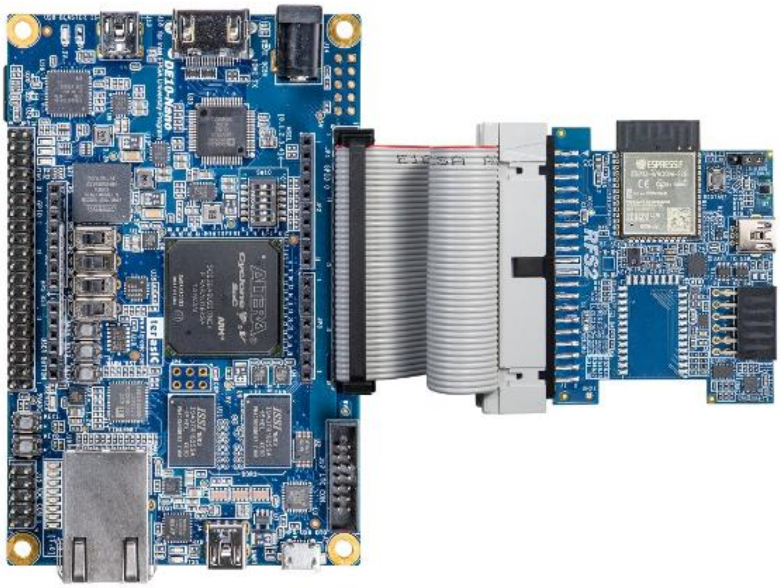


Figure 4-18 Setup Sensor demo on DE10-Nano

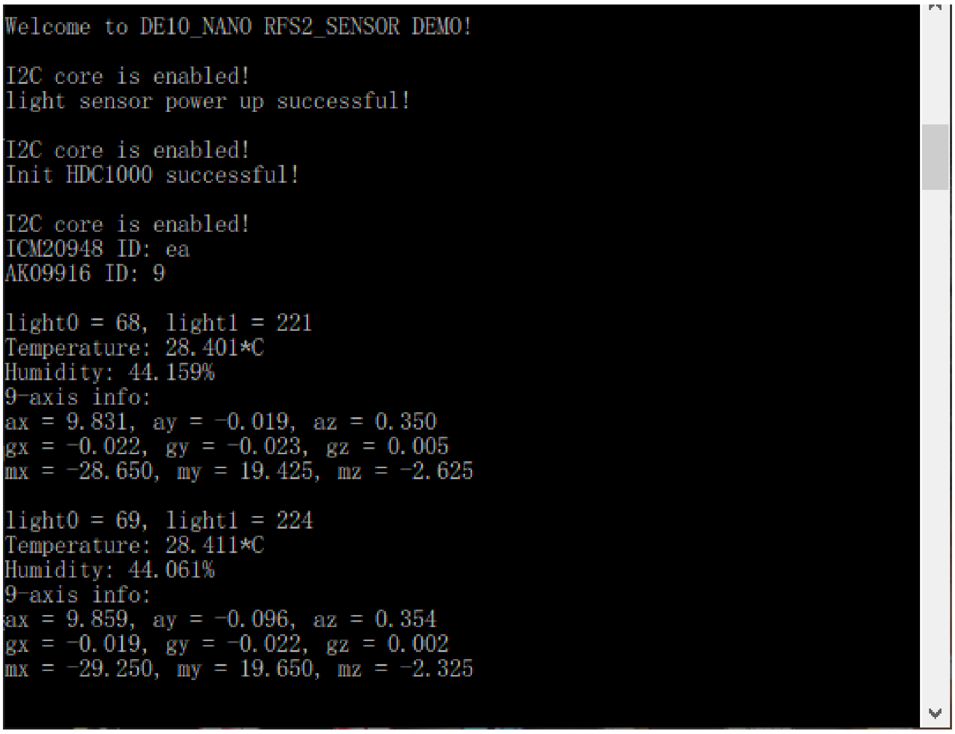


Figure 4-19 Report measured sensor values on DE10-Nano

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration as shown in **Figure 4-20**.

1. Make sure Quartus Prime v16.1 or later is installed on your PC.
2. Connect the RFS2 daughter card to the 2x20 GPIO connector on the DE10-Lite board.
3. Connect a USB cable between the host PC and the USB connector (J3) on the DE10-Lite.
4. Execute the batch file " test.bat" under the demo_batch folder of DE10_LITE_RFS2_SENSOR project.
5. The Nios II terminal will show the measured sensor value as shown in **Figure 4-21.**

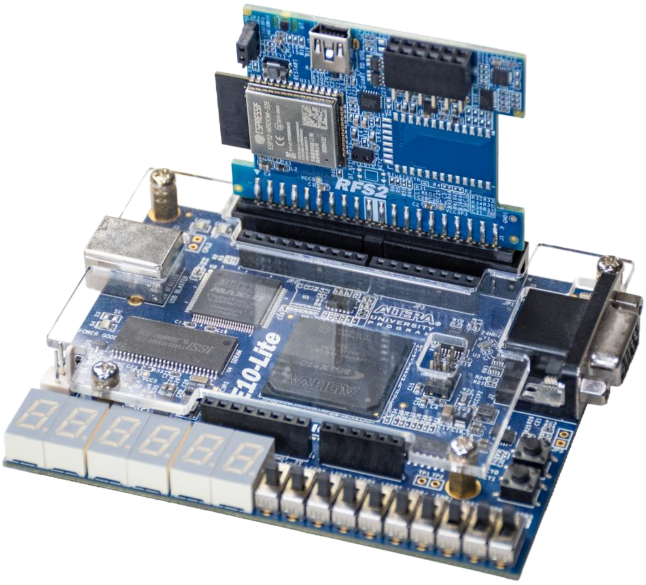
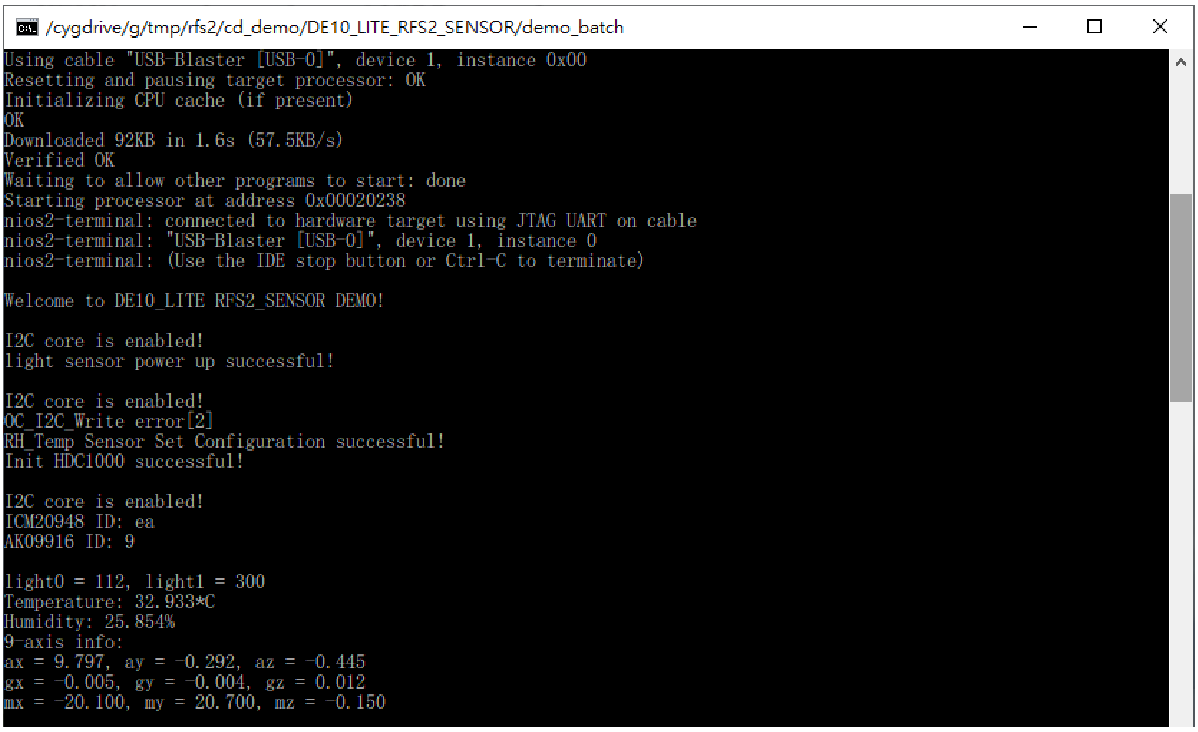


Figure 4-20 Setup Sensor demo on DE10-Lite



/cygdrive/g/tmp/rfs2/cd_demo/DE10_LITE_RFS2_SENSOR/demo_batch

```
Using cable "USB-Blaster [USB-0]", device 1, instance 0x00
Resetting and pausing target processor: OK
Initializing CPU cache (if present)
OK
Downloaded 92KB in 1.6s (57.5KB/s)
Verified OK
Waiting to allow other programs to start: done
Starting processor at address 0x00020238
nios2-terminal: connected to hardware target using JTAG UART on cable
nios2-terminal: "USB-Blaster [USB-0]", device 1, instance 0
nios2-terminal: (Use the IDE stop button or Ctrl-C to terminate)

Welcome to DE10_LITE RFS2_SENSOR DEMO!

I2C core is enabled!
light sensor power up successful!

I2C core is enabled!
OC_I2C_Write error[2]
RH_Temp Sensor Set Configuration successful!
Init HDC1000 successful!

I2C core is enabled!
ICM20948 ID: ea
AK09916 ID: 9

light0 = 112, light1 = 300
Temperature: 32.933*C
Humidity: 25.854%
9-axis info:
ax = 9.797, ay = -0.292, az = -0.445
gx = -0.005, gy = -0.004, gz = 0.012
mx = -20.100, my = 20.700, mz = -0.150
```

Figure 4-21 Report measured sensor values on DE10-Lite

Chapter 5

Appendix A

5.1 Revision History

Version	Change Log
V1.0	Initial Version