RFS

UserManual

www.terasic.com

Content

Chapter 1 Introduction.. 1

1.1 Package Contents 1

1.2 RFS System CD 2

1.3 Assemble the RFS Daughter Card 2

1.4 Getting Help..5

Chapter 2 RFS Daughter Card 6

2.1 Features 6

2.2 Block Diagram of the RFS Daughter Card 7

2.3 Component Layout..7

2.5 Interface for FPGA Host 8

Chapter 3 RTL Example Designs 9

3.1 Query Current Time through Wi-Fi 9

3.2 Bluetooth SPP Slave 12

3.3 Sensor Measurement...17

Chapter 4 Nios II Based Example Designs 21

4.1 Wi-Fi Client 21

4.2 Wi-Fi Server...24

4.3 Bluetooth Configuration 28

4.4 Bluetooth SPP Master 33

4.5 Sensor Measurement...40

Chapter 5 Appendix A.. 45

5.1 Revision History 45

Chapter 1

Introduction

The RFS (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 Bluetooth wireless 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 be provided to communication with Host PC.

1.1 Package Contents

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

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

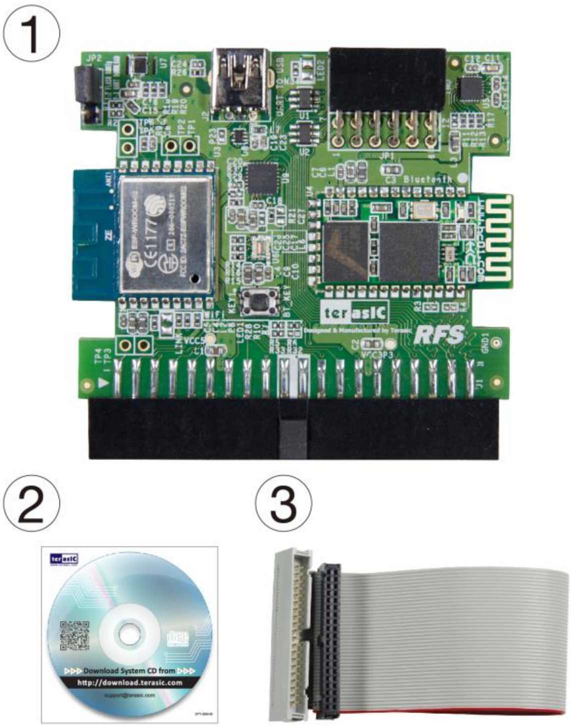


Figure 1-1 The RFS Kit content

1.2 RFS System CD

The RFS 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://rfs.terasic.com/cd.

1.3 Assemble the RFS Daughter Card

The RFS 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 RFS daughter card is connected to various Terasic FPGA Boards:

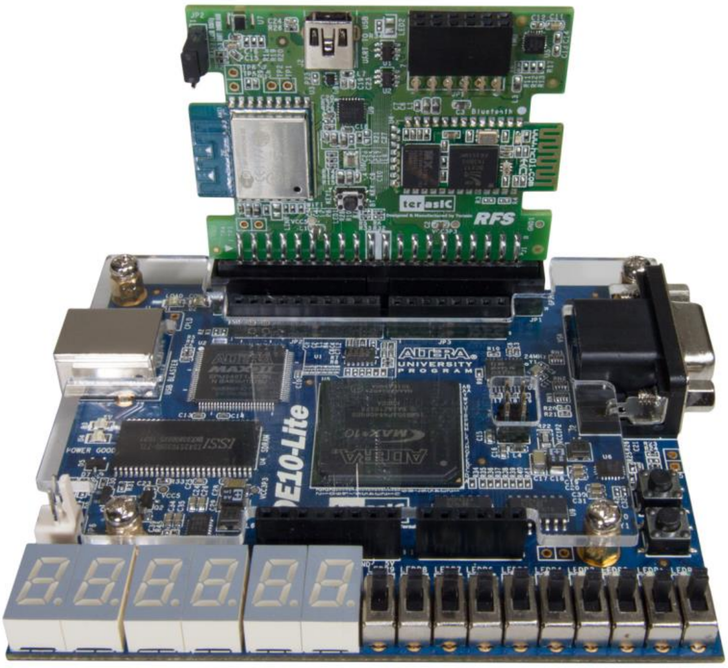


Figure 1-2 Connect the RFS to DE10-Lite directly

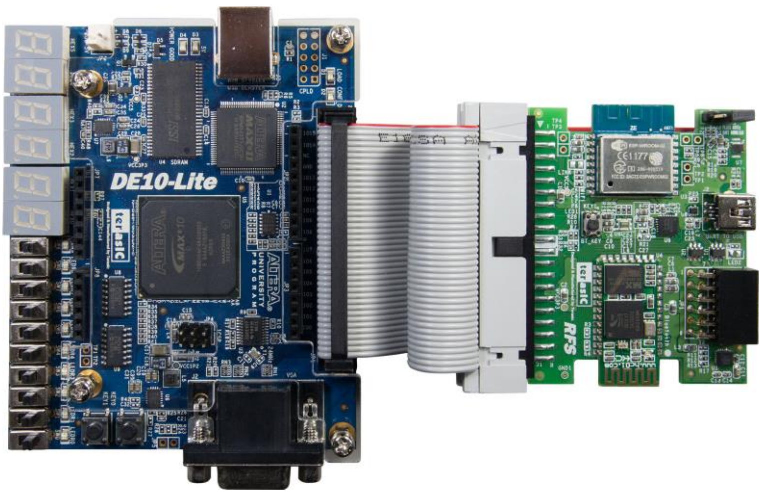


Figure 1-3 Connect the RFS to DE10-Lite

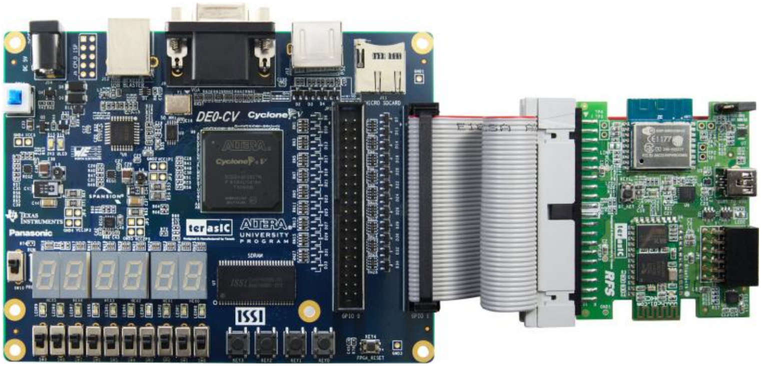


Figure 1-4 Connect the RFS to DE0-CV

Figure 1-5 Connect the RFS to DE0-Nano

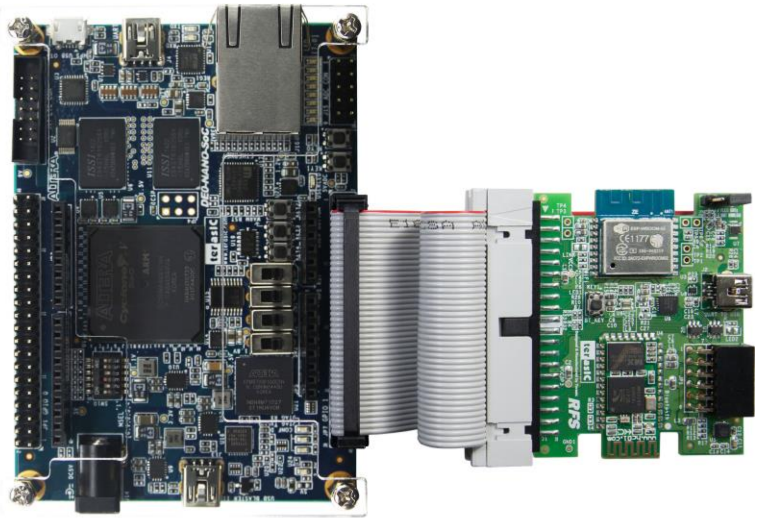


Figure 1-6 Connect the RFS to DE0-Nano-SoC

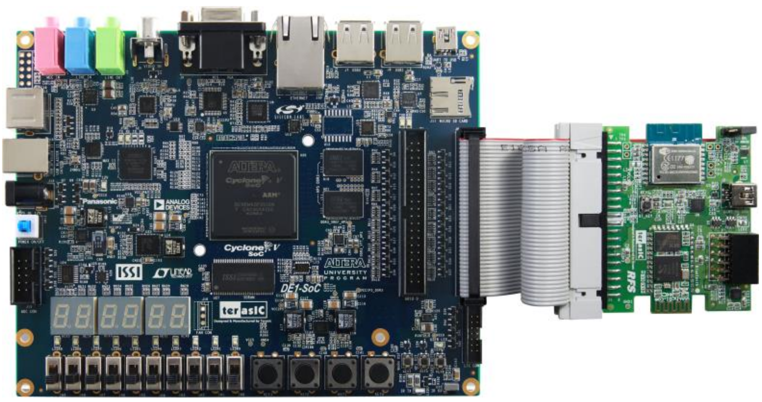


Figure 1-7 Connect the RFS to DE1-SoC

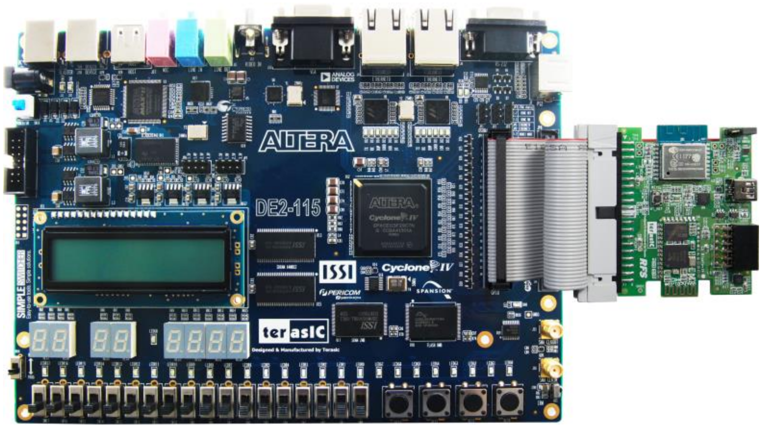


Figure 1-8 Connect the RFS to DE2-115

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://rfs.terasic.com

Chapter 2

RFS Daughter Card

This chapter will introduce the RFS 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 RFS daughter card.

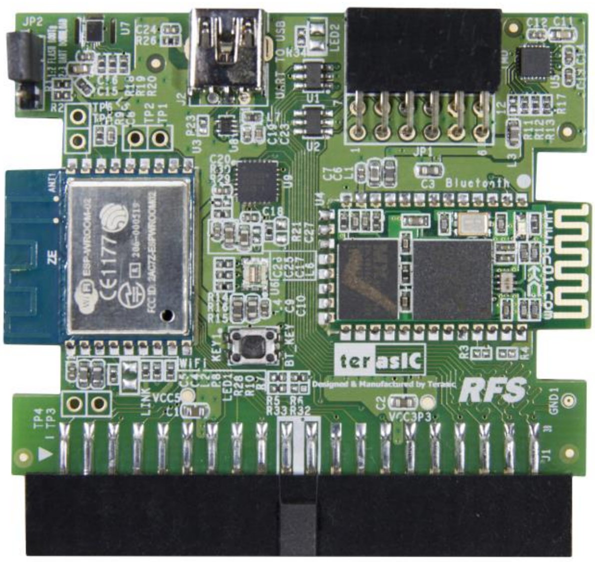


Figure 2-1 RFS Daughter Card

The features of the RFS card are:

- 2x20 GPIO interface
- Wi-Fi, using ESP-WRDDOM-02 module
- Bluetooth SPP, using HC-05 module
- Ambient Light Sensor
- Temperature and humidity sensor
- 9-axis sensor – accelerometer, magnetometer, and gyroscope
- UART to USB
- 2x6 TMD GPIO Header

2.2 Block Diagram of the RFS Daughter Card

Figure 2-2 shows the block diagram of the RFS 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 module interface is 4-pin UART. In the 2x6 TMD header, signals are connected to the 2x20 GPIO header directly.

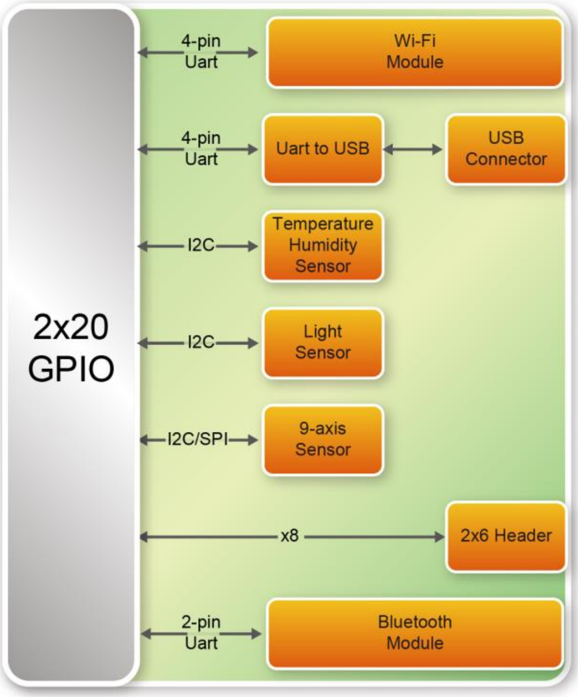


Figure 2-2 Block Diagram of RFS Card

2.3 Component Layout

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

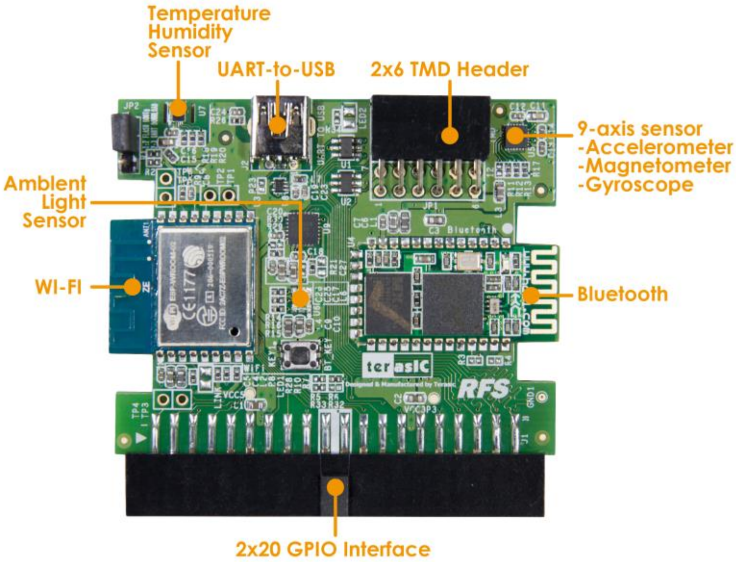


Figure 2-3 Major component layout on the RFS card

2.5 Interface for FPGA Host

The RFS 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.

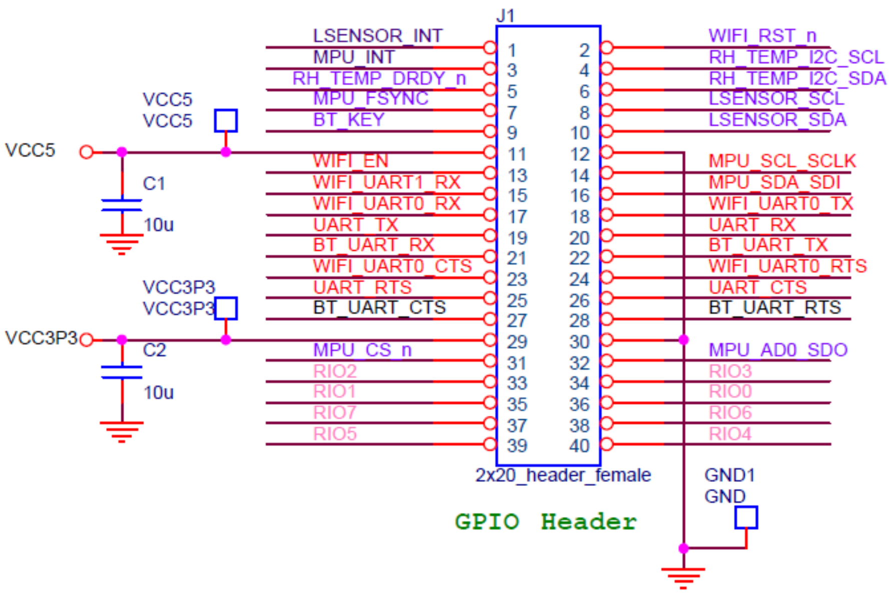


Figure 2-4 2x20 GPIO Pinout

Chapter 3

RTL Example Designs

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

3.1 Query Current Time through Wi-Fi

This section describe how to the Wi-Fi signal is transmitted via the ESP8266 Wi-Fi module on the RFS daughter card. The command is transmitted from the FPGA to the Wi-Fi module via UART. This demonstration uses the ESP8266 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.

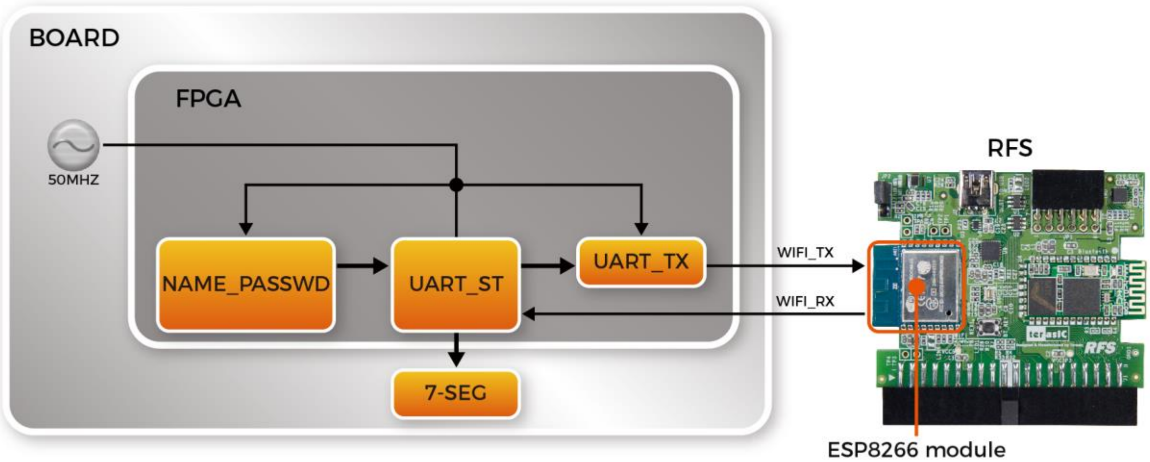


Figure 3-1 Function block diagram of querying current time

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 IP from Altera. It is set to Memery mapping with baud-rate at 112500 bps.
UART_ST: It sends a series of command which queries the current time to the ESP8266 Wi-Fi module via the UART IP. 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 ESP8266 Wi-Fi module
Step 2 Set the ESP8266 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 DE2-115 and DE10-Lite. Please follow the steps below for running the demo on these two boards.

Quartus Project Information

For DE2-115 Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE2_115_WIFI_TIME_RTL
Demo Batch File	Demonstrations\DE2_115_WIFI_TIME_RTL\demo_batch

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE10_LITE_WIFI_TIME_RTL
Demo Batch File	Demonstrations\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 RFS daughter card to the GPIO of DE2-115 board.
2. Plug in 12 V DC to DE2-115
3. Connect the host PC to the USB connector (J9) on DE2-115 via USB cable.
4. Please make sure Quartus Prime has been installed on the host PC.
5. Search for the key string assign NAME="" and assign PASSWORD="" in DE2_115_WIFI_TIME_RTL.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 DE2_115_WIFI_TIME_RTL project.
7. Press KEY0 and wait for 15 ~ 20 seconds. The current network time will be displayed on the 7-segment of DE2-115 in decimal hour:minute:second.

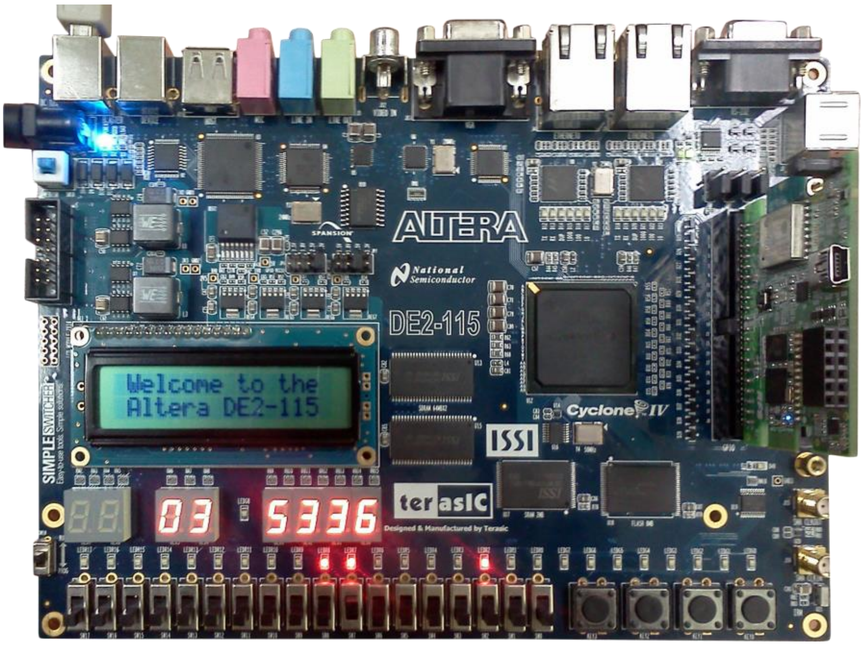


Figure 3-2 Wi-Fi Time setup with DE2-115

■ Demonstration Setup for DE10-Lite

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

1. Connect the RFS 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 Prime has been installed on the host PC.
4. 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**.
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.

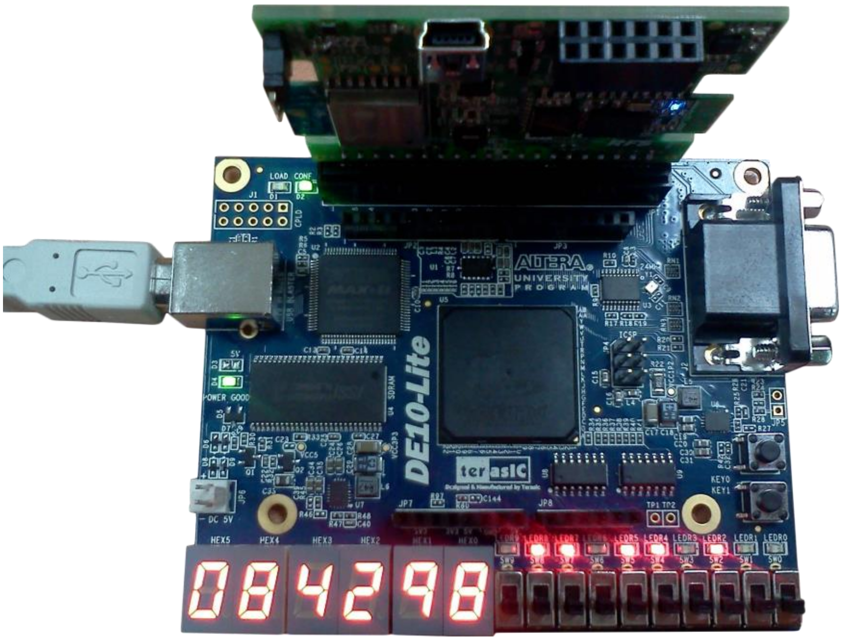


Figure 3-3 Wi-Fi Time Demo setup for DE10-Lite

3.2 Bluetooth SPP Slave

This demonstration shows how to make the HC-05 module in the RFS board to work as a SPP slave. When the SPP slave is launched, it is in standby mode for waiting a SPP connection request from a SPP master. When it accepts a SPP connection from an Android phone, it will be waiting for receiving command from Android phone and interpret the command to control the LED on the FPGA main board.

In this demonstration, Terasic Android TerasicRFS App is used to test the SPP Master. Users also can use other third party Bluetooth SPP APP which is able to send character '0' to '9' to test this demonstration.

■ Block Diagram

Figure 3-4 shows the System Block Diagram of the Bluetooth SPP slave demonstration. In the FPGA board, a **UART Controller** is used to communicate with the HC-05 Bluetooth module. The **Decoder** module is designed to parsing the received command string and control the LEDs on the FPGA main board. The Bluetooth App running on the Android Phone is designed by Terasic. It can discovery the HC-05 device, connect it, and send command string to it.

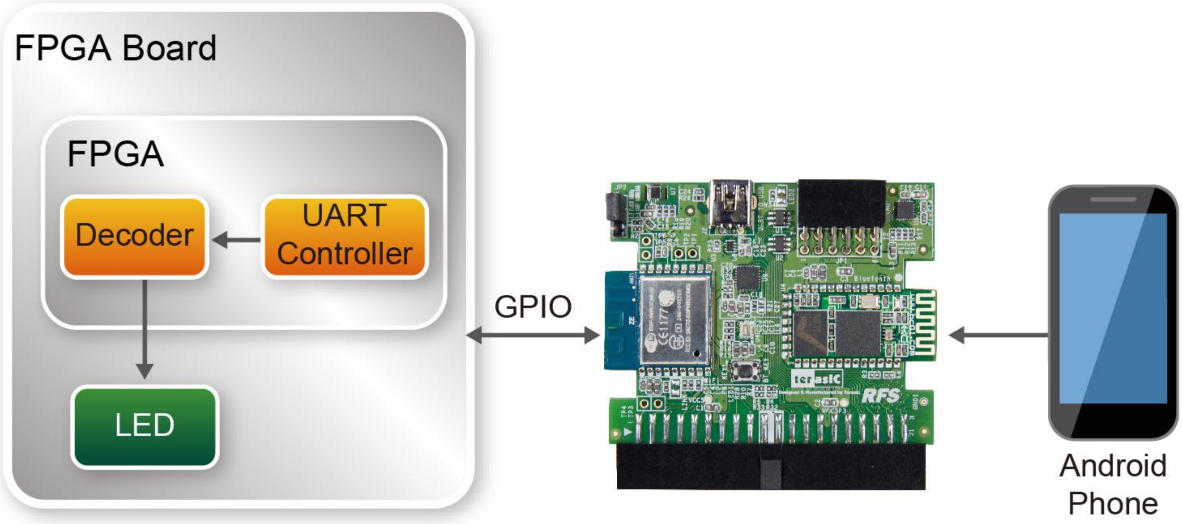


Figure 3-4 Function Block Diagram

■ Communication Protocol

A simple communication protocol is used in this demonstration. One way communication is used in the protocol. The **Decoder** only receive commands from the Android device. It does not response or send any message to the Android device. The command is fixed one-byte length. The valid value is from 0x30(ASCII code of character '0') to 0x39(ASCII code of character '9'). **Figure 3-5** shows the command parsing statement in **Decoder** module. For examples, when 0x30 is received, the LED0 will be turn on. When 0x34 is received, the LED0 will be turn off.

```
case(uart_data)
10'h30:LEDR <= LEDR | 8'd1;
10'h31:LEDR <= LEDR | 8'd2;
10'h32:LEDR <= LEDR | 8'd4;
10'h33:LEDR <= LEDR | 8'd8;
10'h34:LEDR <= LEDR & 8'he;
10'h35:LEDR <= LEDR & 8'hd;
10'h36:LEDR <= LEDR & 8'hb;
10'h37:LEDR <= LEDR & 8'h7;
10'h38:LEDR <= LEDR | 8'hf;
10'h39:LEDR <= LEDR & 8'h0;
default : LEDR <= LEDR;
endcase
```

Figure 3-5 UART data decoder

For SPP mater side, users also can use other third party Bluetooth SPP software to send the commands defined in this demonstration to control the LED on the FPGA main board.

If necessary, designers can expand the one-way communication to two-way communication. Designers can use the uart_tx port of the UART Controller to send command to the Android device.

■ Quartus Project Information

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE10_LITE_BluetoothSPP_Slave
Demo Batch File	Demonstrations\DE10_LITE_BluetoothSPP_Slave\demo_batch

For DE2-115 Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE2_115_BluetoothSPP_Slave
Demo Batch File	Demonstrations\DE2_115_BluetoothSPP_Slave\demo_batch

■ Android Project Information

Android project directory: Demonstrations\Android_Project\RFS.
Android APP Installer: http://www.terasic.com/downloads/demo/rfs/RFS.apk

■ Demonstration Setup for DE10-LITE

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

1. Power off the DE10-LITE board.
2. Connect the RFS 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. In your Android device, download the Android TerasicRFS App from the QR code shown **Figure 3-7**.
6. Install TerasicRFS on your Android device and Launch it.
7. For first time to connect HC-05 Bluetooth module, in the TerasicRFS App, click ZOOM icon to discover the **HC-05** and pair it with pin-code "1234" as shown in **Figure 3-8**.
8. Click the ZOOM icon and connect the paired **HC-05** device.
9. In the TerasicRFS App GUI, click the LED0/LED1/LED2/LED3 and ON/OFF icons to control the LED on DE10-Lite.

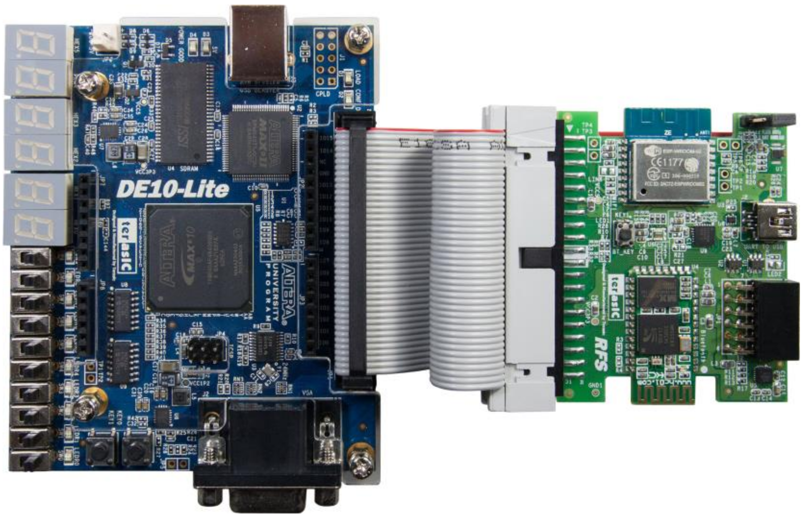


Figure 3-6 Demo setup to DE10-Lite

Note, the QR code link to:

http://www.terasic.com/downloads/demo/rfs/RFS.apk

Figure 3-7 Android QR Code

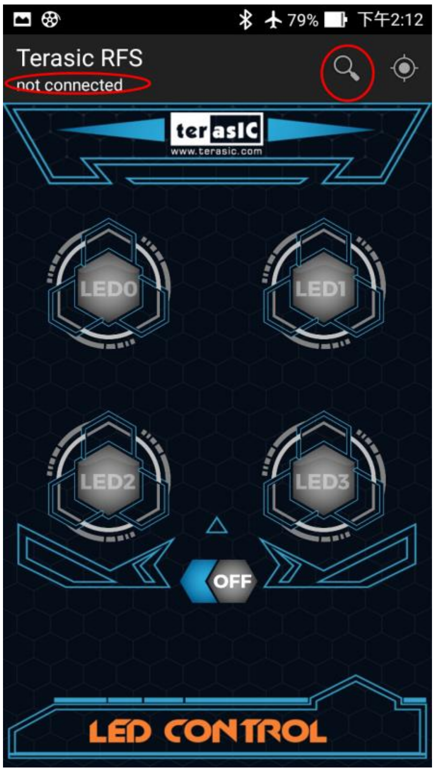


Figure 3-8 TerasicRFS Application UI

■ Demonstration Setup for DE2-115

Please follow the procedures below to setup the demonstration as shown in **Figure 3-9**

1. Power off the DE2-115 board.
2. Connect the GPIO0 output of the DE2-115 board to the RFS board.
3. Power on the DE2-115 board.
4. Connect a USB cable between the host PC and the USB BLASTER (J9) on the DE2-115.
5. Execute the batch file " test.bat" under the demo_batch folder of DE2-115_BluetoothSPP_Slave project.
6. In your Android device, download the Android TerasicRFS App from the QR code shown **Figure 3-7**.
7. Install TerasicRFS on your Android device and Launch it.
8. The first time connecting to the HC-05 Bluetooth module, in the TerasicRFS App, click ZOOM icon to discover the **HC-05** and pair it with pin-code "1234" as shown in **Figure 3-8**.
9. Click the ZOOM icon and connect the paired DE2-115 device.
10. In the TerasicRFS App GUI, click the LED0/LED1/LED2/LED3 and ON/OFF icons to control the on the DE2-115.

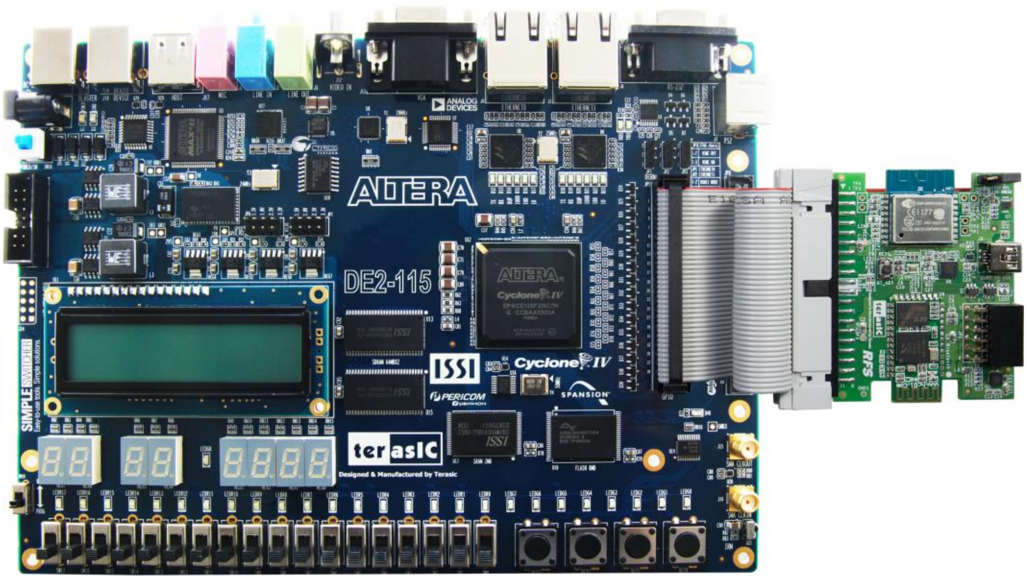


Figure 3-9 Demo Setup for DE2-115

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 RFS 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.

■ Block Diagram

Figure 3-10 shows the function block diagram of Sensor Measurement demonstration.

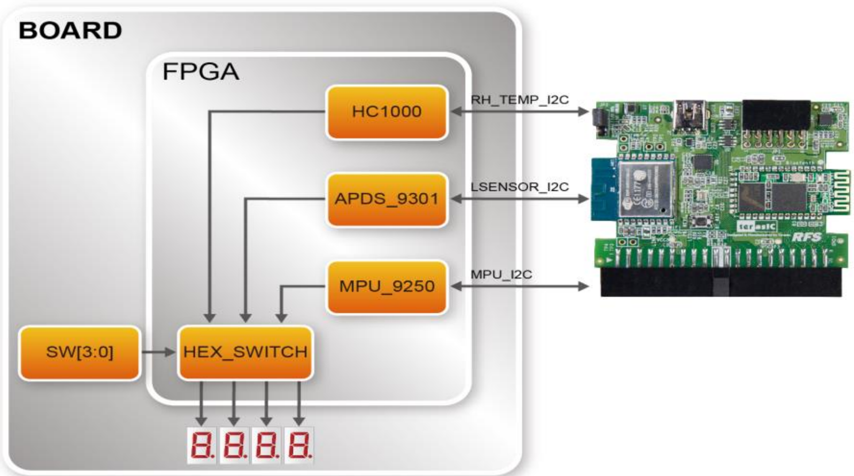


Figure 3-10 Function block diagram of sensor measurement

HC1000: It controls the HDC1000 IC on RFS 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.

APDS_9301: It controls the APDS9301 IC on RFS daughter card via I2C bus. It can retrieve the measurement of light photo sensor from ADC0 and ADC1 via slave address 0x52. The length of these two sets are 16-bit.

MPU_9250: It controls the MPU9250 IC on RFS daughter card via I2C bus. There are two ICs integrated into a single MPU9250 chip. One is the accelerometer, which can be accessed from slave address 0xD0 or DxD2 when the pin AD0_SDO = 1, and the other one is AK8963 as magnetometer, which can be accessed from slave address 0x18, provided the register 0x37 of MPU9250 is set to 0x02 (I2C bypass mode). The usage is similar to several I2C slave devices in parallel for the three measurements of MPU9250 (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 12 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 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

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	Accelerometer X	hexadecimal (positive/negative)
4	Accelerometer Y	hexadecimal (positive/negative)
5	Accelerometer Z	hexadecimal (positive/negative)
6	Gyroscope X	hexadecimal (positive/negative)
7	Gyroscope Y	hexadecimal (positive/negative)
8	Gyroscope Z	hexadecimal (positive/negative)
9	Magnetometer X	hexadecimal (positive/negative)
10	Magnetometer Y	hexadecimal (positive/negative)
11	Magnetometer Z	hexadecimal (positive/negative)

This demo can be realized on DE2-115 and DE10-Lite. Please follow the steps below for running the demo on these two boards.

■ Quartus Project Inforamtioin

Tool: Quartus Prime Standard Edition V16.1.1.
Project directory: Demonstrations\DE2_115_RFS_SENSOR_RTL
Demo batch file folder: Demonstrations\DE2_115_RFS_SENSOR_RTL\demo_batch

■ Demonstration Setup for DE2-115

Please follow the procedures below to setup the demonstration, as shown in **Figure 3-11** Demo setup with DE2-115

1. Connect the RFS daughter card to the GPIO of DE2-115 board.
2. Plug in 12 V DC to DE2-115
3. Connect the host PC to the USB connector (J9) on DE2-115 via USB cable.
4. Please make sure Quartus Prime has been installed on the host PC.
5. Execute the batch file " test.bat" under the demo_batch folder of DE2_115_RFS_SENSOR_RTL project.
6. Set SW[3:0] to the corresponding positions, as shown in **Table 3-1**, for the measurement. .

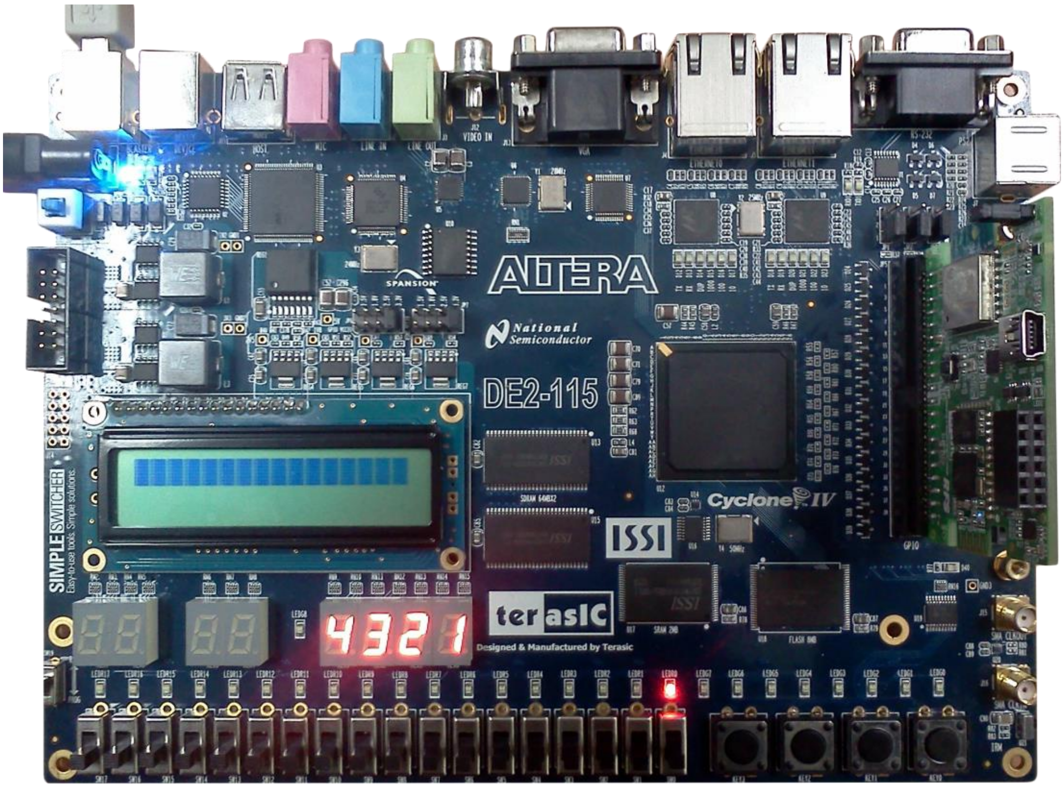


Figure 3-11 Demo setup with DE2-115

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration, as shown in **Figure 3-12**.

1. Connect the RFS 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_RFS_SENSOR_RTL project.
5. Set SW[3:0] to the corresponding positions, as shown in **Table 3-1**, for the measurement. .

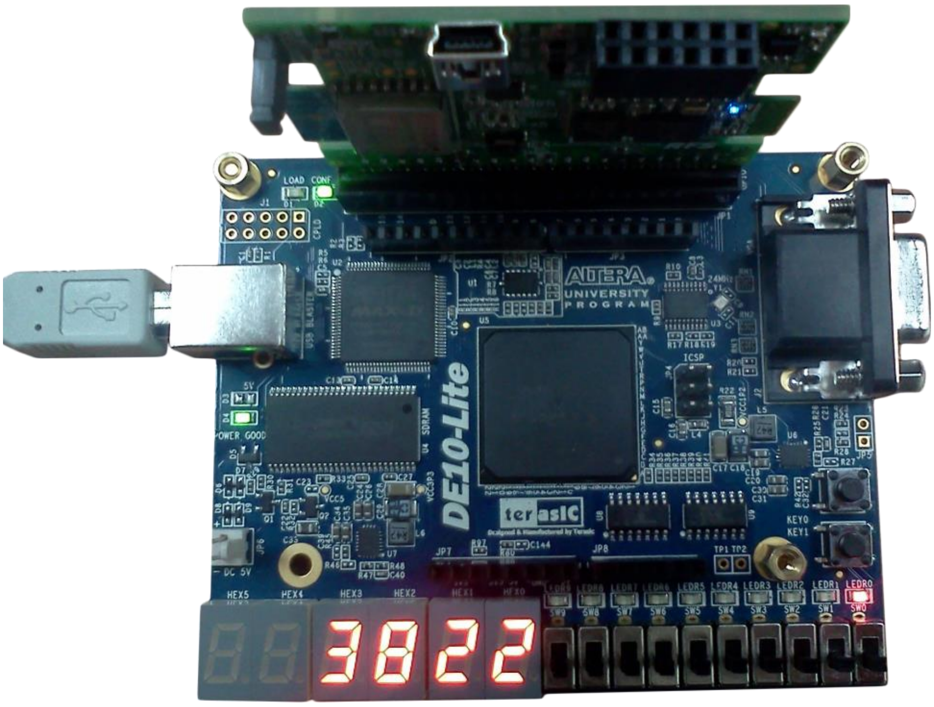


Figure 3-12 Demo setup for DE10-Lite

Chapter 4

Nios II Based Example Designs

This chapter will demonstrate how the FPGA to communicate with the RFS 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 module on the RFS daughter card. The time information will be shown on the 7-segment with the format HH:MM:SS.

In this demonstration, a Wi-Fi AP 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 Wi-Fi module on RFS daughter card. 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.

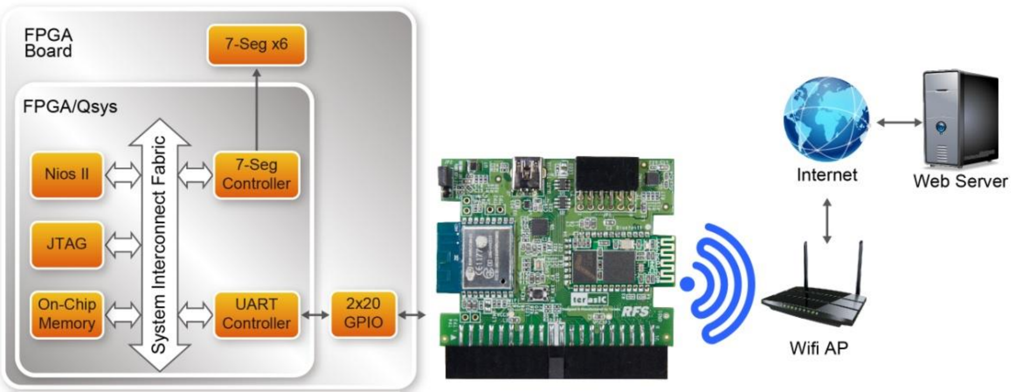


Figure 4-1 Function Block Diagram

How to Query Internet Time

The internet time information is available on the web http://www.timeapi.org. Sending URL http://www.timeapi.org/utc/now?\I:\M:\S to the web server, it will response current time in format HH:MM:SS.

Quartus Project Information

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE10_LITE_WiFi_Network_Time
Demo Batch File	Demonstrations\DE10_LITE_WiFi_Network_Time\demo_batch

For DE2-115 Mainboard

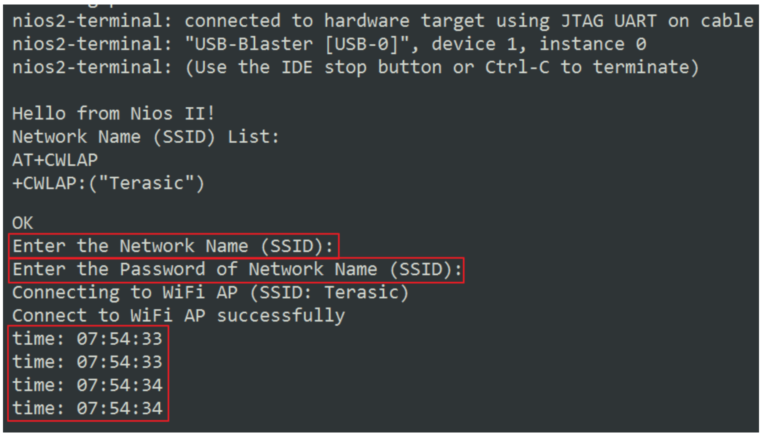
Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE2_115_WiFi_Network_Time
Demo Batch File	Demonstrations\DE2_115_WiFi_Network_Time\demo_batch

Demonstration Setup for DE10-Lite

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

1. Connect the RFS 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 RFS 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.
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 4-3**.

Figure 4-2 Setup ESP8266 Client Demo



```
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!
Network Name (SSID) List:
AT+CWLAP
+CWLAP:("Terasic")

OK
Enter the Network Name (SSID):
Enter the Password of Network Name (SSID):
Connecting to WiFi AP (SSID: Terasic)
Connect to WiFi AP successfully
time: 07:54:33
time: 07:54:33
time: 07:54:34
time: 07:54:34
```

Figure 4-3 WiFi_Network_Time Demo

■ Demonstration Setup for DE2-115

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

1. Connect the RFS daughter card to the GPIO connector of the DE2-115 board.
2. Make sure the jumper is located at pin 1-2 on the JP2 of RFS daughter card.
3. Connect a USB cable between the host PC and the USB connector (J9) on the DE2-115.
4. Execute the batch file “test.bat” under the demo_batch folder of DE2_115_WiFi_Network_Time project.
5. Input SSID of your WiFi AP according to the prompt in the Nios II terminal. The terminal will not display the SSID.
6. Input the password for your WiFi AP according to the prompt in the Nios II terminal. The terminal will not display the password.
7. The UTC(Universal Time Coordinated) time will be displayed on the six 7-segement as shown in **Figure 4-3**.

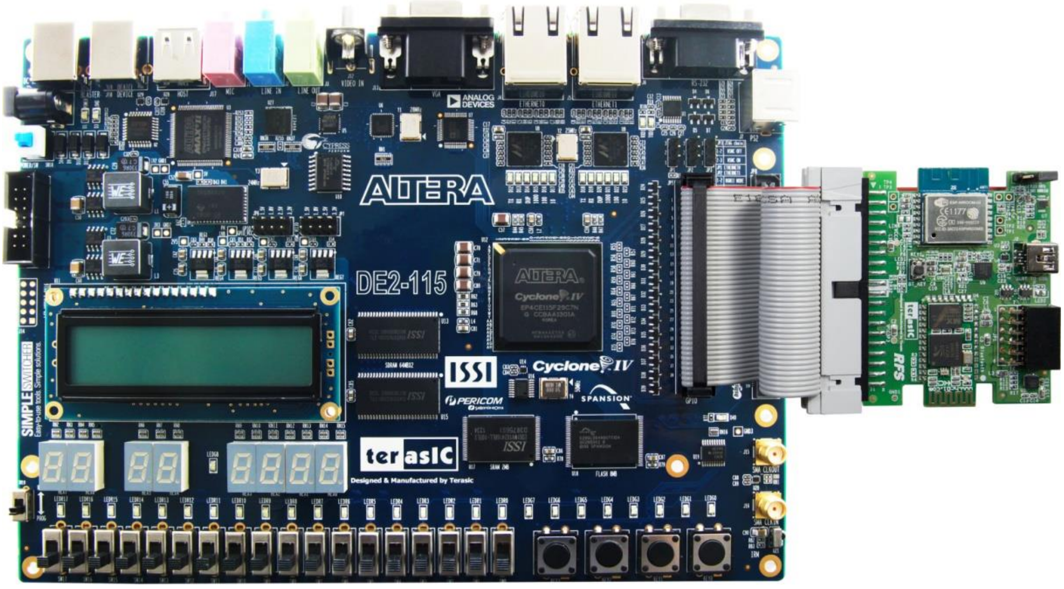


Figure 4-4 Setup ESP8266 Client Demo

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 4-5. Users can click the blue led text to lighten/un-lighten the LEDs on the FPGA main board.

Figure 4-5 LED Control Web

■ Block Diagram

Figure 4-6 shows the Function block diagram of the Wi-Fi Server demonstration. The UART controller is used to communicate with the Wi-Fi module. 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.

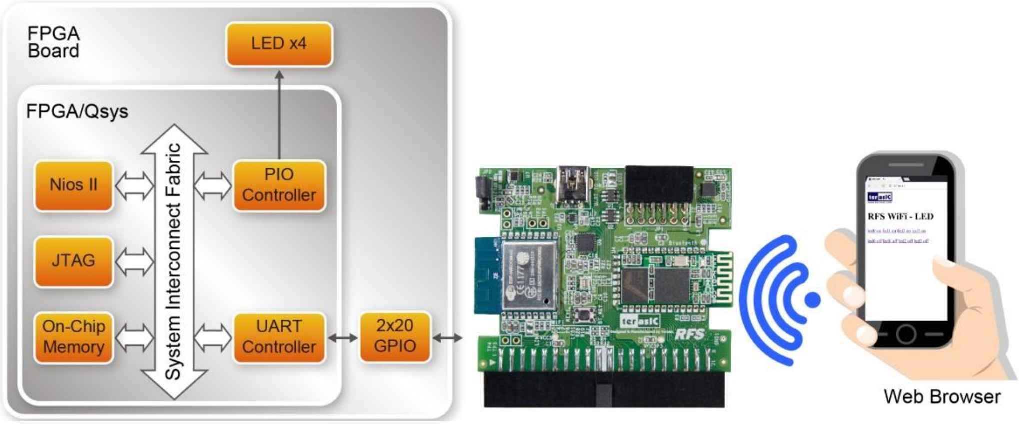


Figure 4-6 Function Block Diagram

■ 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-Lite Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE10_LITE_WiFi_Web_Server
Demo Batch File	Demonstrations\DE10_LITE_WiFi_Web_Server\demo_batch

For DE2-115 Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE2_115_WiFi_Web_Server
Demo Batch File	Demonstrations\DE2_115_WiFi_Web_Server\demo_batch

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration as shown in **Figure 4-7**.

1. Connect the RFS 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 RFS 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.
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 4-5**.

Figure 4-7 Setup ESP8266 Web Server Demo

■ Demonstration Setup for DE2-115

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

1. Connect the RFS daughter card to the GPIO connector of the DE2-115 board.
2. Make sure the jumper is located at pin 1-2 on the JP2 of RFS daughter card.
3. Connect a USB cable between the host PC and the USB connector (J9) on the DE2-115.
4. Execute the batch file “test.bat” under the demo_batch folder of DE2_115_WiFi_Network_Time project.
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 4-5**.

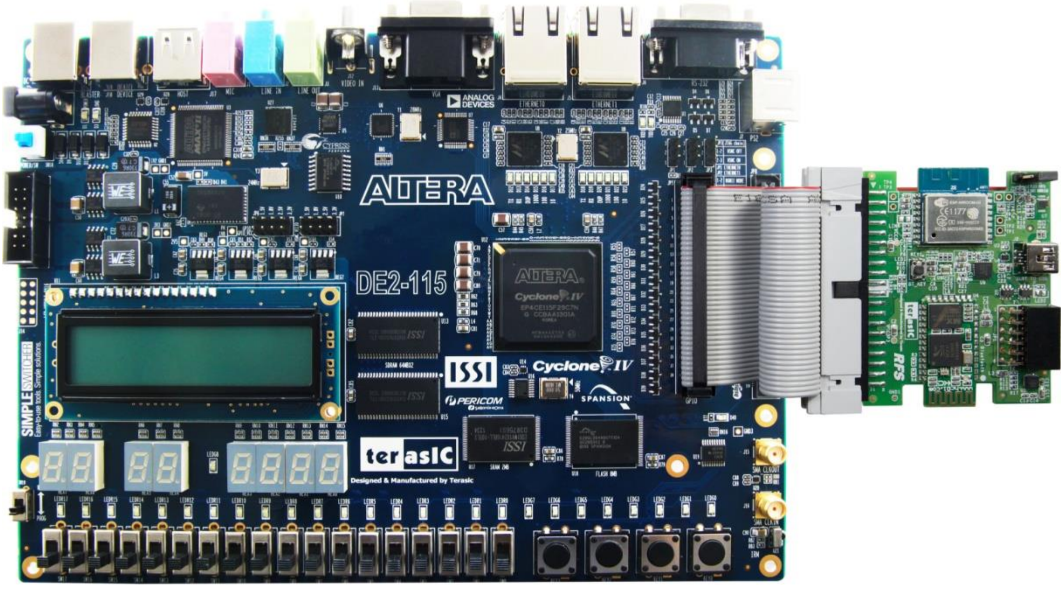


Figure 4-8 Setup ESP8266 Web Server Demo

4.3 Bluetooth Configuration

This section describe how to restore the HC-05 to default status and set the UART baud rate to 115200. The AT command set, defined by HC-05, is used to restore the setting. Besides restore UART baud rate, developers also can use the AT command set to configure the other parameters for the HC-05 firmware.

The HC-05 firmware has two work modes: **order-response** work mode (AT mode) and **automatic connection** work mode. In this demonstration, **order-response** work mode is used to configure the HC-05. Note, before send AT Command to configure the HC-05, HC-05 should enter **order-response** work mode first.

■ Block Diagram

Figure 4-9 shows the function block diagram of Bluetooth Configuration Demonstration. Nios II program is used to control the main flow. The Nios II program is running on the on-chip memory. The UART controller is used to communicate with the HC-05 Bluetooth module. The TX Port is used to send AT command and RX Port is used to receive response. The GPIO controller is used to monitor the status of KEY1 button on the FPGA main board. When the Nios II program is launched, it is continued polling the status of KEY1 button. When KEY1 is pressed, the Nios II program will start the configure process.

The baud rate of the UART controller is 38400. For HC-05 to work at baud rate 38400 in **order-response** work mode (AT mode), users should press BT_KEY button (pull high PIO11 pin in HC-05) while HC-05 power on.

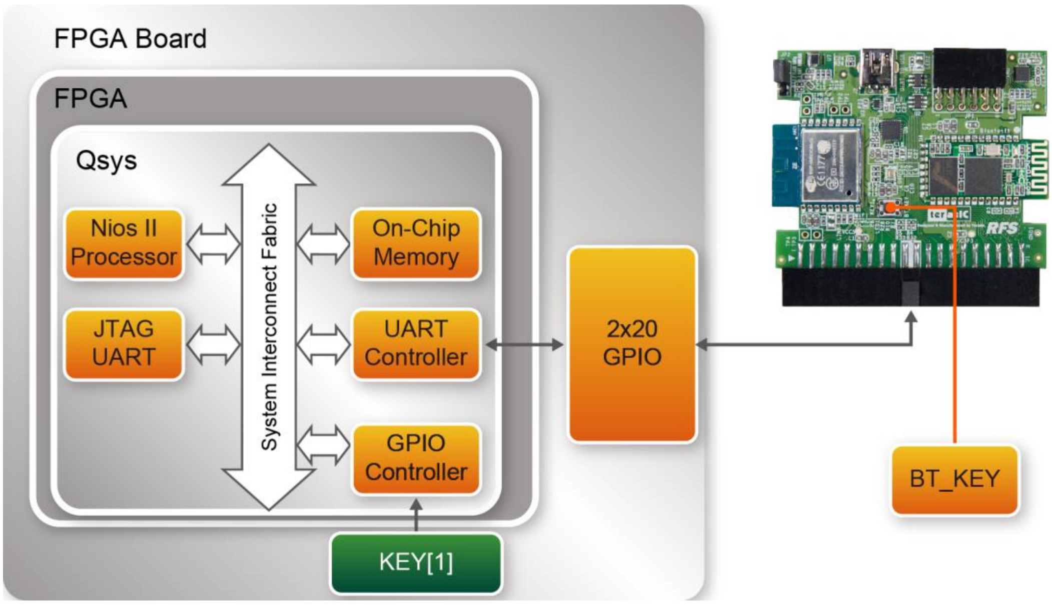


Figure 4-9 Block Diagram for Bluetooth Configuration

■ AT Command

The HC-05 firmware has two work modes: **order-response** work mode (AT mode) and **automatic connection** work mode. In the **order-response** work mode, AT Command is used to communicate with the HC-05 firmware. For details about AT command set defined by HC-05, please refer to the ***HC05_AT_Command_Ref.pdf*** in the folder Datasheet\HC-05 in the RFS System CD. In order to make HC-05 to work on **order-response** work mode with baud rate 38400, the “Way 2: how to get to the AT mode” described in the ***HC05_AT_Command_Ref.pdf*** is used in this demonstration. This means users need to press the BT_KEY button when HC-05 is power on.

In this demonstration, the AT Commend is used to configure the HC-05 firmware under the **order-response** work mode. When users press KEY1 button to start the configure process, the Nios II program sends “AT\r\n” command first. When HC-05 receives the “AT\r\n” command, it responses “OK\r\n”. Then, Nios II program sends “AT+ORGL\r\n” command to HC-05 for restore default status. HC-05 responses “OK\r\n” when it restores the status. Finally, the Nios II Program sends “AT+UART=115200,0,0\r\n” to the HC-05. When the baud-rate is changed to 115200, the HC-05 responses “OK\r\n”. **Figure 4-10** shows the handshake chart for the above three AT commands.

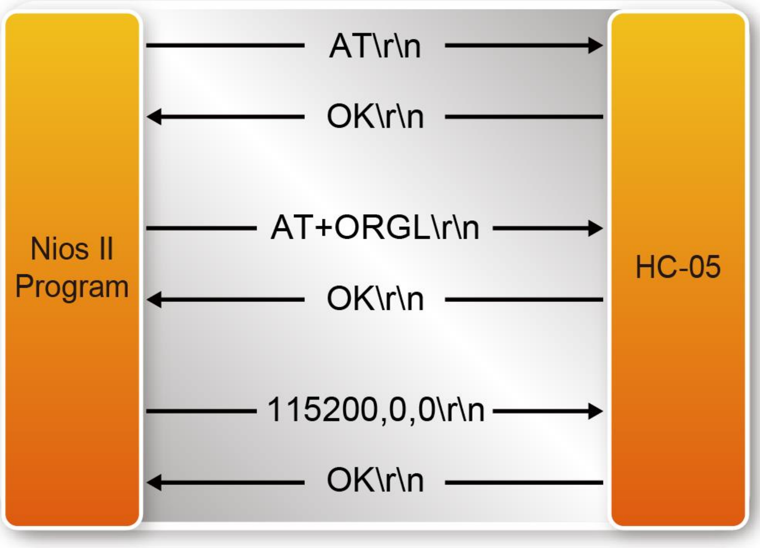


Figure 4-10 Handshake Chart

■ Quartus Project Information

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE10_LITE_Bluetooth_Config
Demo Batch File	Demonstrations\DE10_LITE_Bluetooth_Config\demo_batch

For DE2-115 Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE2_115_Bluetooth_Config
Demo Batch File	Demonstrations\DE2_115_Bluetooth_Config\demo_batch

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration as shown in **Figure 4-11**.

1. Connect the RFS daughter card to the GPIO connector of the DE10-Lite board.
2. Push down and hold the BT_KEY (pull high PIO11) as shown in **Figure 4-12**.
3. Power on the DE10-Lite board and release the BT_KEY.
4. Execute the batch file " test.bat" under the demo_batch folder of DE10_LITE_Bluetooth_Config project.
5. When your see "Press KEY1 to config HC-05" prompt in the Nios II terminal, press KEY [1] on DE10-lite to start the configure process.
6. **Figure 4-13** shows the screenshot when configure is completed.
7. You can re-power the mainboard to make the configure take effect.

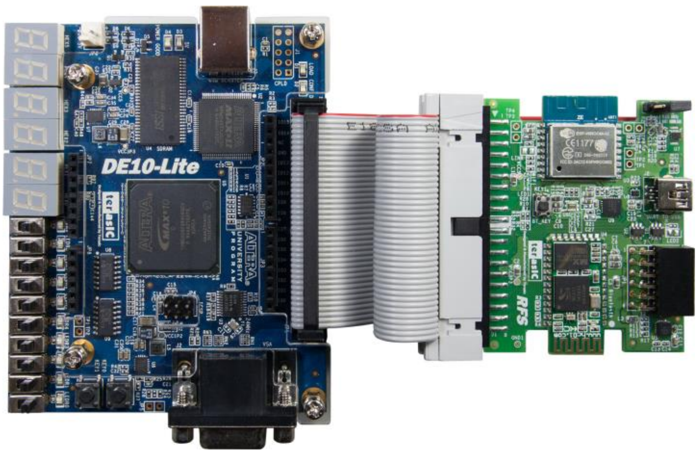


Figure 4-11 Setup HC-05 Configuration Demo

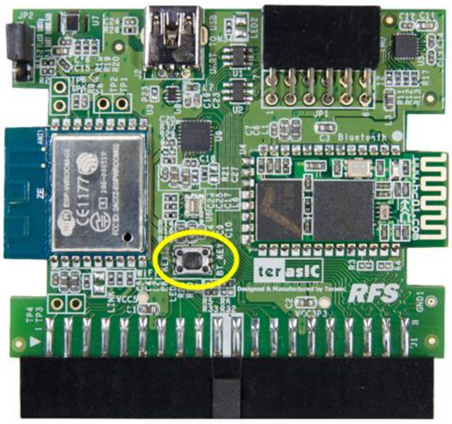


Figure 4-12 BT_KEY for enter order-response work mode (AT mode)

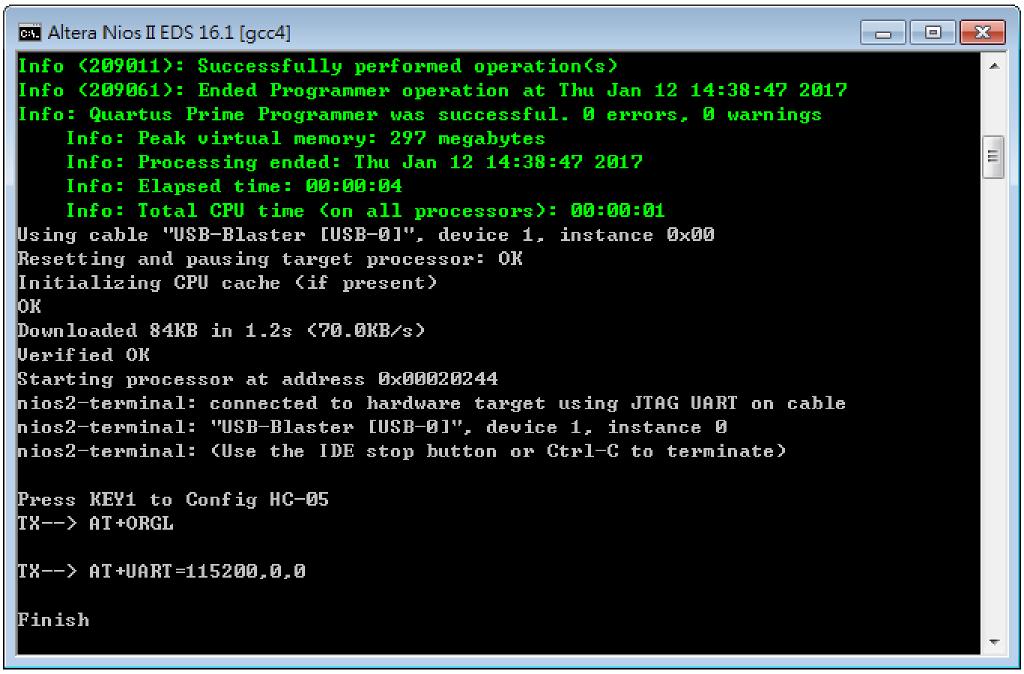


Figure 4-13 Screen shot of reconfig HC-05

■ Demonstration Setup for DE2-115

Please follow the procedures below to setup the demonstration as shown in **Figure 4-14**

1. Connect the GPIO0 output of the DE2-115 board to the RFS board.
2. Push down and hold the BT_KEY (pull high PIO11) as shown in **Figure 4-12**.
3. Power on the DE10-Lite board and release the BT_KEY.
4. Connect a USB cable between the host PC and the USB BLASTER (J9) on the DE2-115.
5. Execute the batch file “ test.bat” under the demo_batch folder of DE2_115_Bluetooth_Config project.
6. When you see “Press KEY1 to config HC-05” prompt in the Nios II terminal, press KEY [1] on DE2-115 to start the configure process.
7. **Figure 4-13** shows the screenshot when configure is completed.
8. You can re-power the mainboard to make the configure take effect.

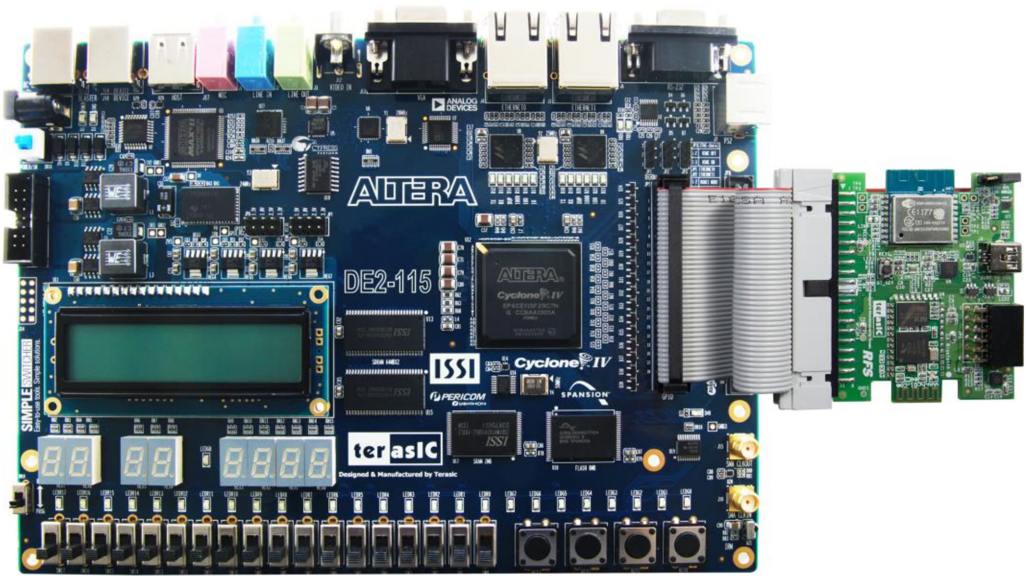


Figure 4-14 Setup HC-05 Configuration Demo

4.4 Bluetooth SPP Master

This demonstration shows how to use AT Command Set to make the HC-05 Bluetooth module in the RFS board to work as a SPP master. The SPP master is able to search nearby Bluetooth devices, and request to establish a SPP RFCOMM connection with a SPP slave device. Then, it can communicate with the connected the SPP slave device through the RFCOMM connection.

In this demonstration, Terasic Android TerasicRFS App is used to test the SPP Master. Users also can use other third party Bluetooth SPP APP to test this demonstration.

The HC-05 firmware has two work modes: **order-response** work mode (AT mode) and **automatic connection** work mode. In this demonstration, there two modes are used. Note, before send AT Command to configure the HC-05, HC-05 should enter **order-response** work mode first.

■ Block Diagram

Figure 4-15 shows the function block diagram of the Bluetooth SPP Master Demonstration. In order to let the HC-05 to work on **order-response** mode with baud rate 38400, the BT_KEY button should be hold down (keep PIO11 high) when module power on. The GPIO controller is used to control the PIO11 status. UART Controller is used to communicate with the HC-05 Bluetooth module with baud rate 38400

The Nios II processor is used for main control. The Nios II program is running on SDRAM memory. Before Nios II program sends AT Command to search nearby Bluetooth devices, it uses GPIO controller to pull high PIO11 to keep HC-05 in **order-response** mode. Nios II program will list all found Bluetooth devices, and ask users to select desired one. Then, it uses AT command to pair and connect the desired device. When connection is established, Nios II program uses GPIO controller to pull low PIO11 to keep HC-05 in **automatic connection** work mode, and start to transmission user data with the connected Bluetooth device.

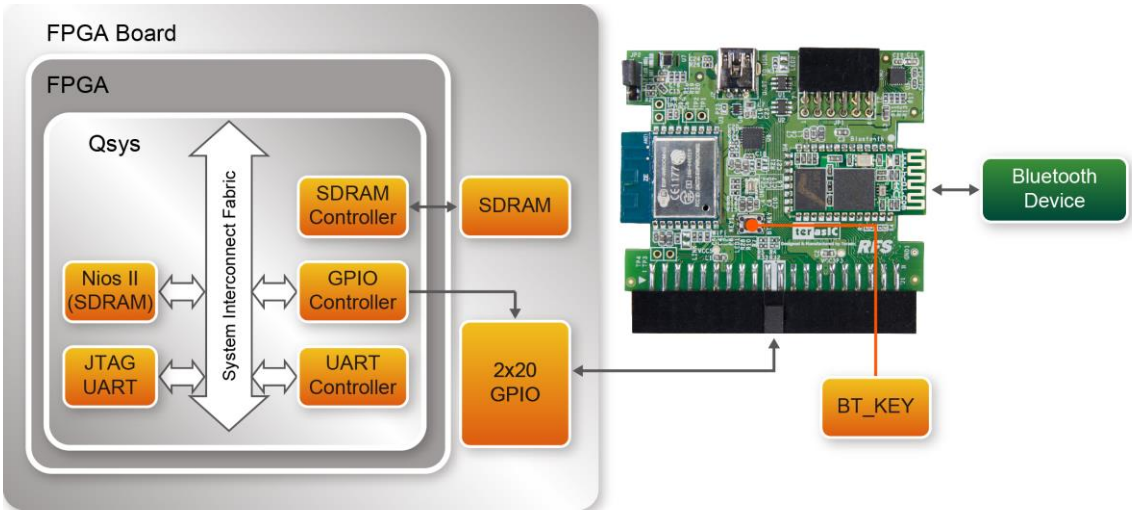


Figure 4-15 Function block diagram

■ AT Command

The HC-05 firmware has two work modes: **order-response** work mode (AT mode) and **automatic connection** work mode. In the **order-response** work mode, AT Command is used to communicate with the HC-05 firmware. For details about AT command set defined by HC-05, please refer to the HC05_AT_Command_Ref.pdf in the RFS CD folder CD\Datasheet\HC-05. In order to make HC-05 to work on **order-response** work mode with baud rate 38400, the "Way 2: how to get to the AT mode" described in the HC05_AT_Command_Ref.pdf is used in demonstration. This means users need to press BT_KEY button when HC-05 is power on.

When the Nios II program is launched, it pull PIO11 high to enter **order-response** work mode. Then, it send out the following AT commands to for initialization:

- AT\r\n
- AT+INIT\r\n
- AT+UART=115200,0,0\r\n
- AT+ROLE=1\r\n
- AT+INQ\r\n

Then, it send "AT+INQ\r\n" to search nearby Bluetooth devices. After users select the desired connected device, it send "AT+PAIR=xxx" and" AT+BIND=xxx" to pair and connect the device. When connection is establish, it pulls PIO11 low to enter **automatic connection** work mode, then send "Hello!" user data to the remote device.

■ Quartus Project

For DE10-Lite Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE10_LITE_BluetoothSPP_Master
Demo Batch File	Demonstrations\DE10_LITE_BluetoothSPP_Master\demo_batch

For DE2-115 Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE2_115_BluetoothSPP_Master
Demo Batch File	Demonstrations\DE2_115_BluetoothSPP_Master\demo_batch

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration as shown in **Figure 4-16** Figure 3-6

1. Connect the RFS daughter card to the GPIO connector of DE10-Lite board.
2. Hold down BT_KEY on the RFS card (Do not release the key) and power on the DE10-Lite.
3. Release the BT_KEY.
4. In your Android device, download the Android TerasicRFS App from the QR code shown **Figure 4-17**.
5. Install TerasicRFS on your Android device and Launch it.
6. In the Terasic RFS App GUI, click the Discoverable Icon as shown in **Figure 4-18**.
7. Execute the batch file " test.bat" under the demo_batch folder of DE10_LITE_BluetoothSPP_Master project.
8. The Nios II program will search the nearby Bluetooth devices and list the found devices as shown in **Figure 4-19**. Then, input a number to select your Android device.
9. For first time to connect, your Android Device will query a pin-code. In this case, please input the pin-code "1234".
10. If the cconnection is established successful, you will see a "Hello!" message on the TerasicRFS App UI as shown in **Figure 4-20**. This message is sent by the Nios II program running on DE10-Lite.
11. When users click the LED icon on the Android TerasicRFS App, the Nios II terminal will show the received characters as shown in **Figure 4-21**.

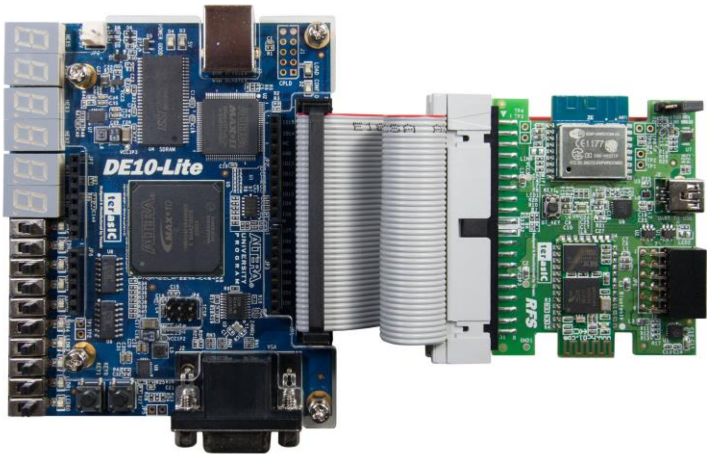


Figure 4-16 Setup Bluetooth Master to DE10-Lite

Figure 4-17 Android QR Code

Figure 4-18 TerasicRFS Application UI

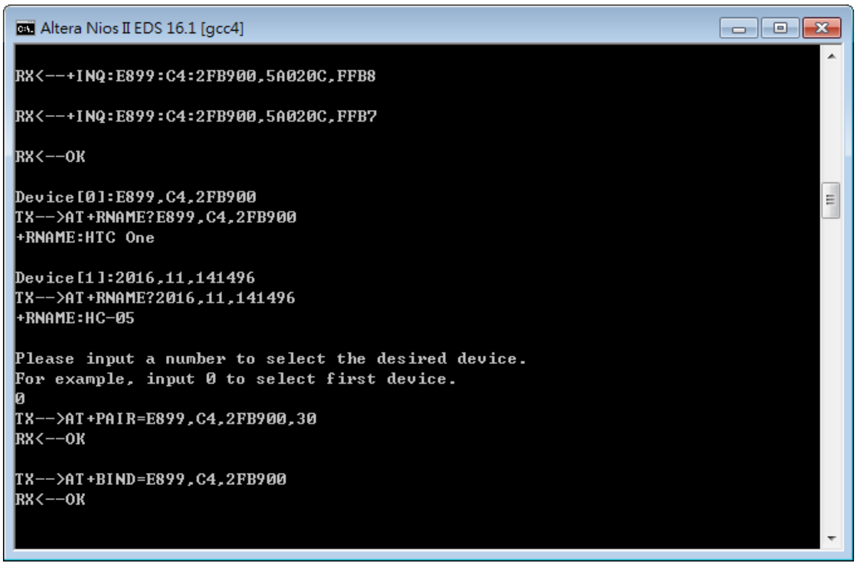


Figure 4-19 List the nearly Bluetooth devices

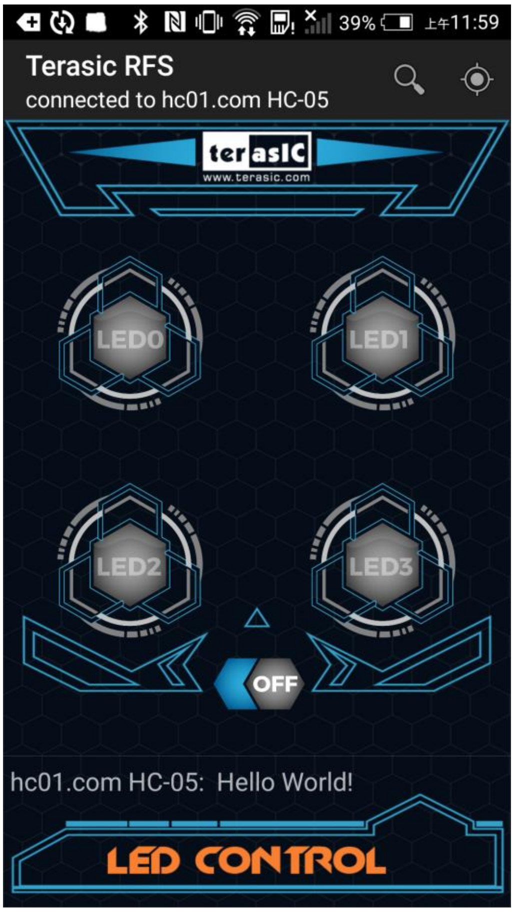


Figure 4-20 Received from the Motherboard Message Screen

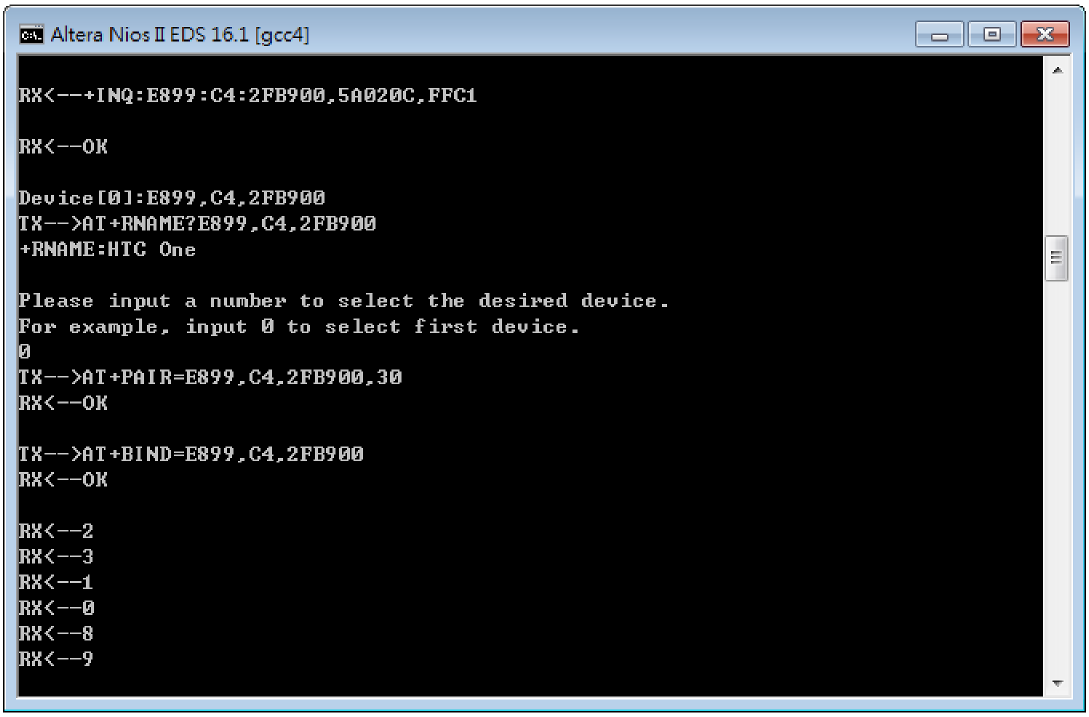


Figure 4-21 Show characters received from the Android device

■ Demonstration Setup for DE2-115

Please follow the procedures below to setup the demonstration as shown in **Figure 4-22.**

1. Connect the RFS daughter card to the GPIO connector of the DE2-115 board.
2. Hold down the BT_KEY on the RFS card (Do not release the key) and power on the DE2-115.
3. Release the BT_KEY.
4. Connect a USB cable between the host PC and the USB BLASTER (J9) on the DE2-115.
5. In your Android device, download the Android TerasicRFS App from the QR code shown **Figure 4-17**.
6. Install TerasicRFS on your Android device and Launch it.
7. In the Terasic RFS App GUI, click the Discoverable Icon as shown in **Figure 4-18**.
8. Execute the batch file " test.bat" under the demo_batch folder of DE2_115_BluetoothSPP_Master project.
9. The Nios II program will search the nearby Bluetooth devices and list the found devices as shown in **Figure 4-19**. Then, input a number to select your Android device.
10. On the first time connecting, your Android Device will query a pin-code. In this case, please input the pin-code "1234".
11. If the cconnection is established successful, you will see a "Hello!" message on the TerasicRFS App UI as shown in **Figure 4-20**. This message is sent by the Nios II program running on DE2-115.
12. When users click the LED icon on the Android TerasicRFS App, the Nios II terminal will show the received characters as shown in **Figure 4-21**

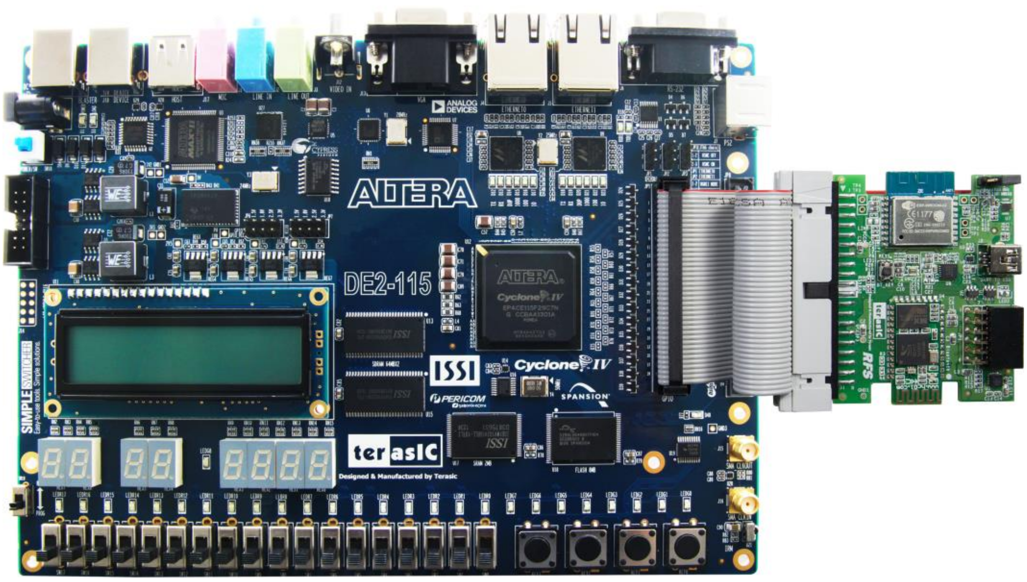


Figure 4-22 Setup Bluetooth Master Demo

4.5 Sensor Measurement

The RFS is equipped with three sensors; an ambient light sensor，a humidity and temperature sensor, and a 9-axis motion tracking sensor. The APDS-9301 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 MPU9250 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 MPU9250 provides complete 9-axis output through the I2C interface.

■ Block Diagram

Figure 4-1 shows the function block diagram of the RFS three-sensor demonstration. The system requires a 50 MHz clock input from the board. Three I2C_OPENCORES controllers are used to communicate with the APDS-9301, HDC1000 and MPU9250 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:

DE10_LITE_RFS_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:

DE10_LITE_RFS_SENSOR/software/RFS_SENSOR

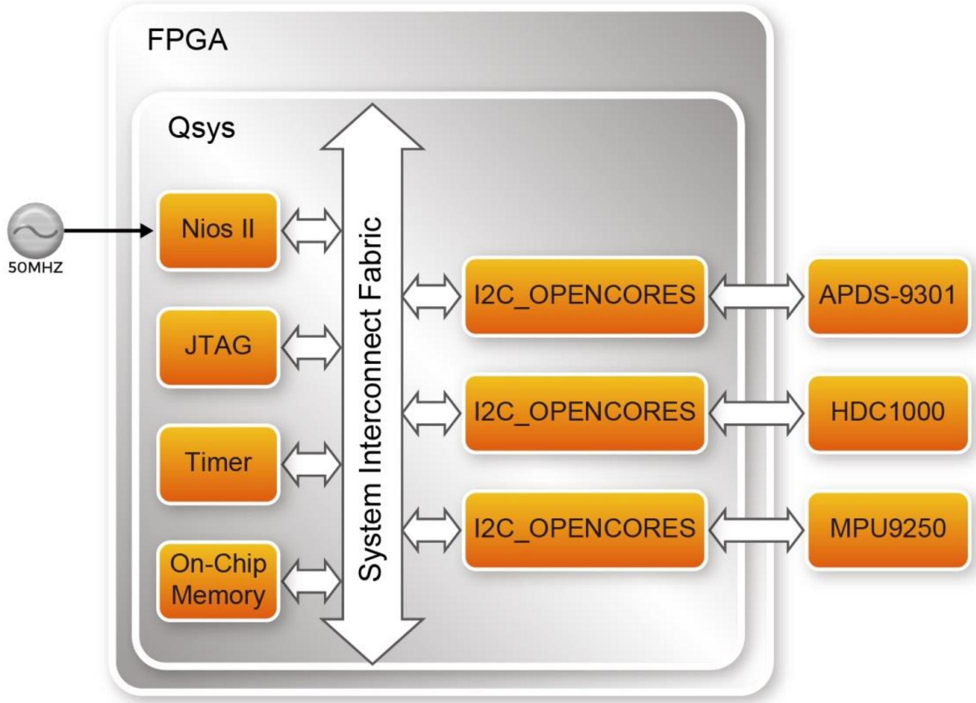


Figure 4-23 Block diagram of the DE10-Lite RFS Sensors demo

■ Sensor Explanation

The APDS-9301 contains two integrated analog-to-digital converters (ADC) 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 APDS-9301 is 0x52/0x53.

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 MPU9250 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 MPU9250 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-Lite Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE10_LITE_RFS_SENSOR
Demo Batch File	Demonstrations\DE10_LITE_ RFS_SENSOR \demo_batch

For DE2-115 Mainboard

Tool	Quartus Prime Standard Edition V16.1.1
Project Directory	Demonstrations\DE2_115_ RFS_SENSOR
Demo Batch File	Demonstrations\DE2_115_ RFS_SENSOR \demo_batch

■ Demonstration Setup for DE10-Lite

Please follow the procedures below to setup the demonstration as shown in **Figure 4-24**.

1. Make sure Quartus Prime v16.1 or later is installed on your PC.
2. Connect the RFS 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_RFS_SENSOR project.
5. The Nios II terminal will show the measured sensor value as shown in Figure 4-25.

Figure 4-24 Setup Sensor demo on DE10-Lite

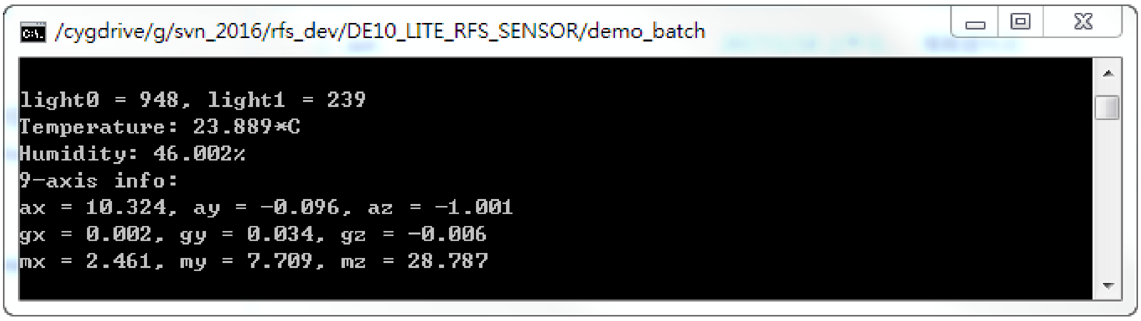


Figure 4-25 Report measured sensor values on DE10-Lite

■ Demonstration Setup for DE2-115

Please follow the procedures below to setup the demonstration as shown in **Figure 4-26**.

1. Make sure Quartus Prime v16.1 or later is installed on your PC.
2. Power off the DE2-115 board
3. Connect RFS daughter card to the 2x20 GPIO connector on the DE2-115 board.
4. Connect a USB cable between the host PC and the USB connector on the DE2-115.
5. Power on DE2-115 board.
6. Execute the batch file " test.bat" under the demo_batch folder of DE2_115 _RFS_SENSOR project.
7. The Nios II terminal will show the measured sensor value as shown in Figure 4-27.

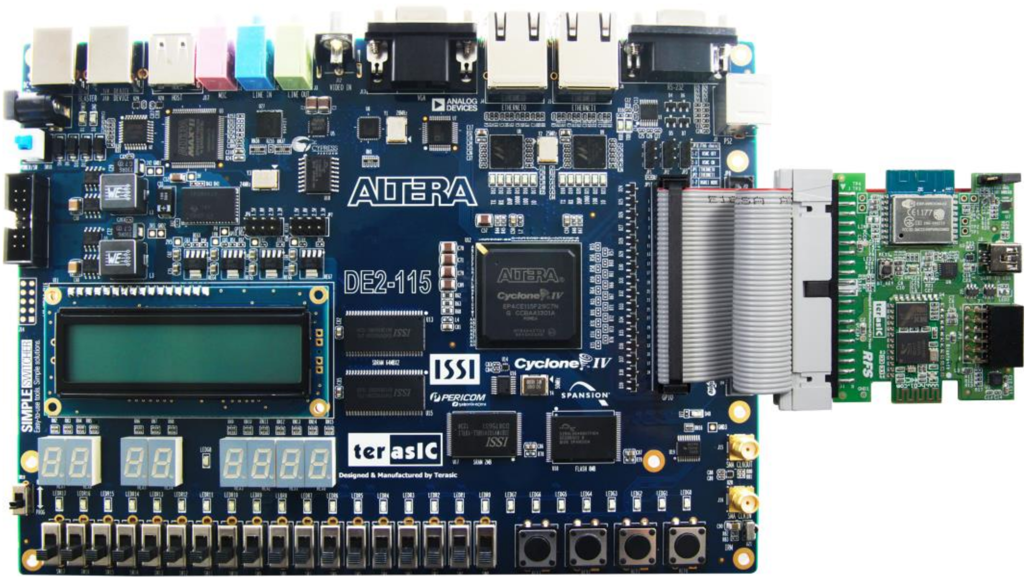


Figure 4-26 Setup Sensor demo on DE2-115

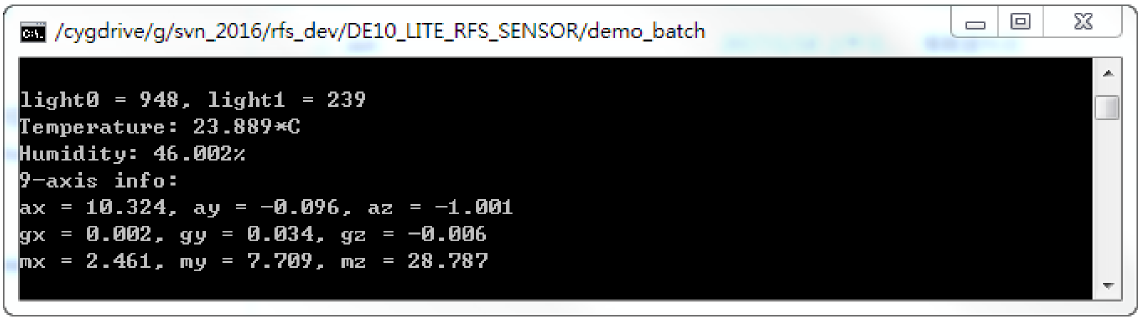


Figure 4-27 Report measured sensor values on DE2-115

Chapter 5

Appendix A

5.1 Revision History

Version	Change Log
V1.0	Initial Version
V1.1	Modify Chapter 3.1 content. Change uart controller.
V1.2	Modify Chapter 3.1, add step for handing AP name and password.