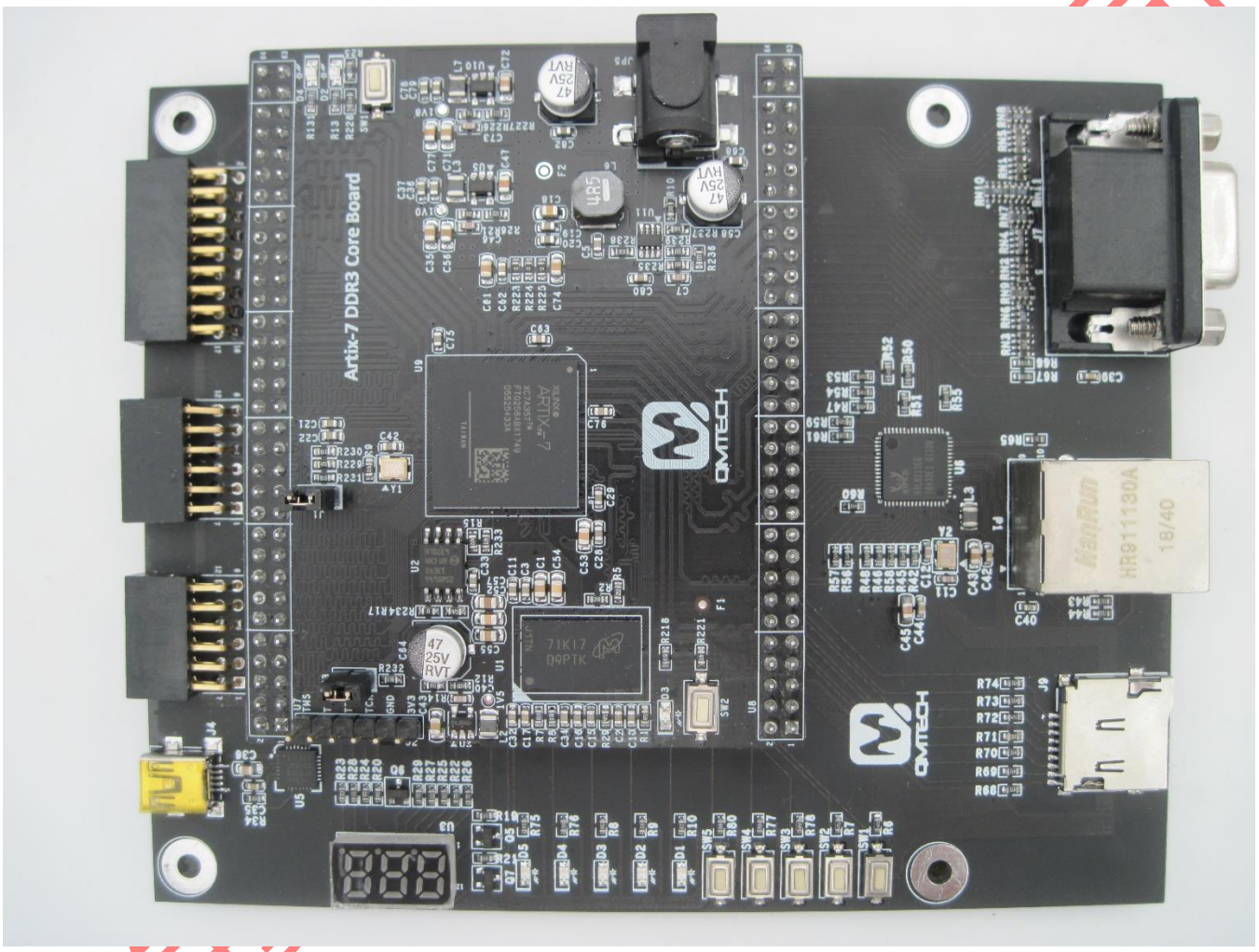
QM_ARTIX7_XC7A35T DB

USER MANUAL



Preface

The QMTech® XC7A35T DDR3 Development Kit uses Xilinx Artix®-7 devices to demonstrate the highest performance-per-watt fabric, transceiver line rates, DSP processing, and AMS integration in a cost-optimized FPGA. Featuring the MicroBlaze™ soft processor and 1,066Mb/s DDR3 support, the family is the best value for a variety of cost and power-sensitive applications including software-defined radio, machine vision cameras, and low-end wireless backhaul.

For more information, updates and useful links, please visit QMTECH Official Website:

http://www.chinaqmtech.com

Table of Contents

1. QM_Artix7_XC7A35T DB Introduction

1.1 DB Overview

QM_Artix7_XC7A35T provides several user interfaces to meet different customer needs. Below section lists the detailed info of these user interfaces:

- USB to UART Serial Port, by using Silicon Labs' CP2102-GMR chip.
- 16bit(RGB565) VGA display interface, by using resistor dividers;
- GMII ethernet interface, by using Realtek's RTL8211EG chip;
- CMOS/CCD camera interface, by using 18pin female header;
- Two Digilent PMOD standard compatible female headers;
- MicroSD card slot;

1.2 Daughter Board Top View

Below figure shows the daughter board of QM_Artix7_XC7A35T development kit. The daughter board's dimension is 108.71mm x 134.62mm. All the functional chips' power supply is injected from the 64P female connector, detailed connection refer to the hardware schematic.

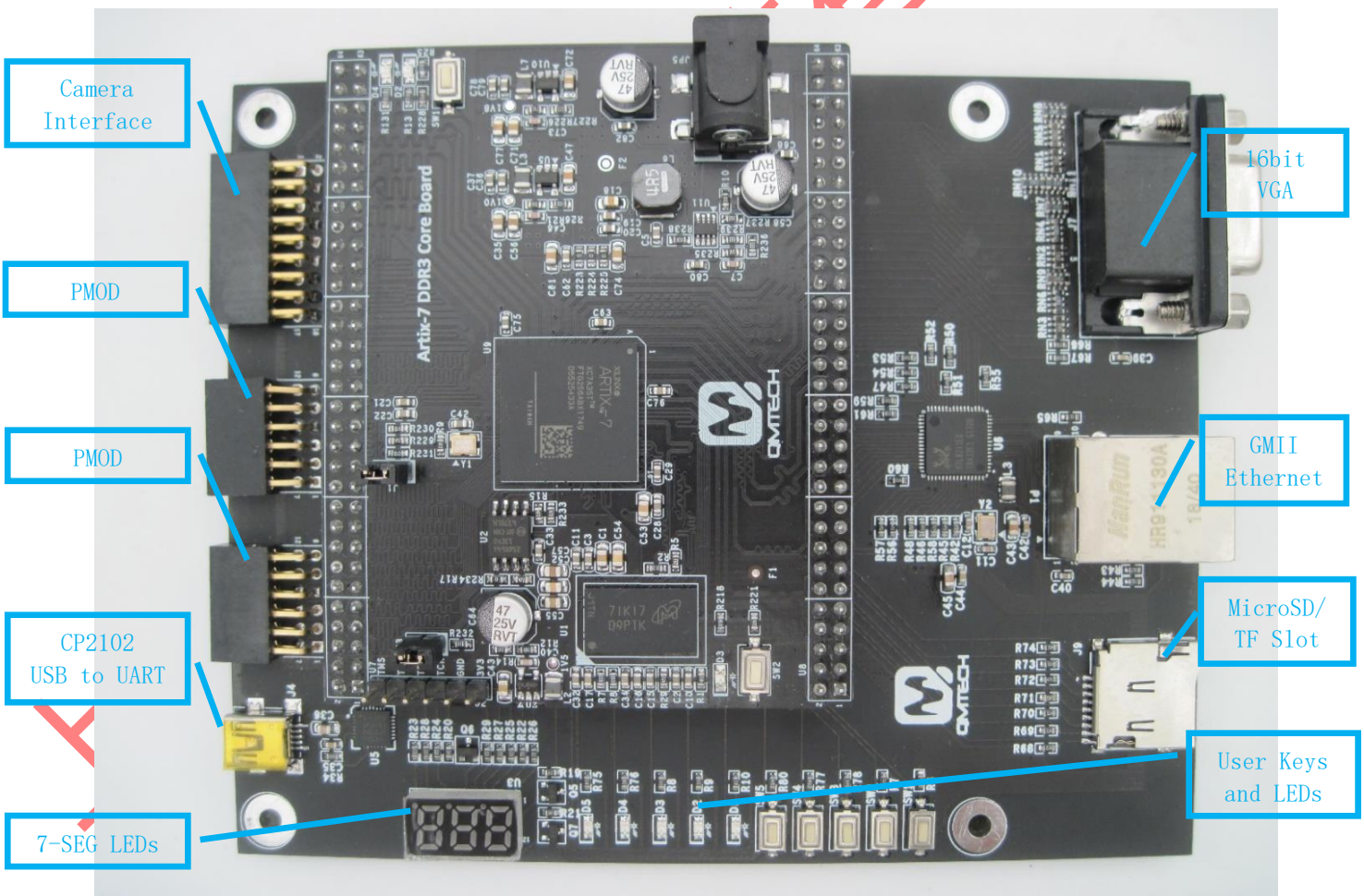


Figure 1-1. QM_Artix7_XC7A35T Daughter Board

2. Experiment (1): USB to Serial Port

The CP2102-GMR is a USB 2.0 to serial port bridge chip designed by Silicon Labs. The CP2102-GMR includes a USB 2.0 full-speed function controller, USB transceiver, oscillator, UART and eliminates the need for other external USB components are required for development. Below figure shows the hardware design of CP2102-GMR on the QM_Artix7_XC7A35T daughter board.

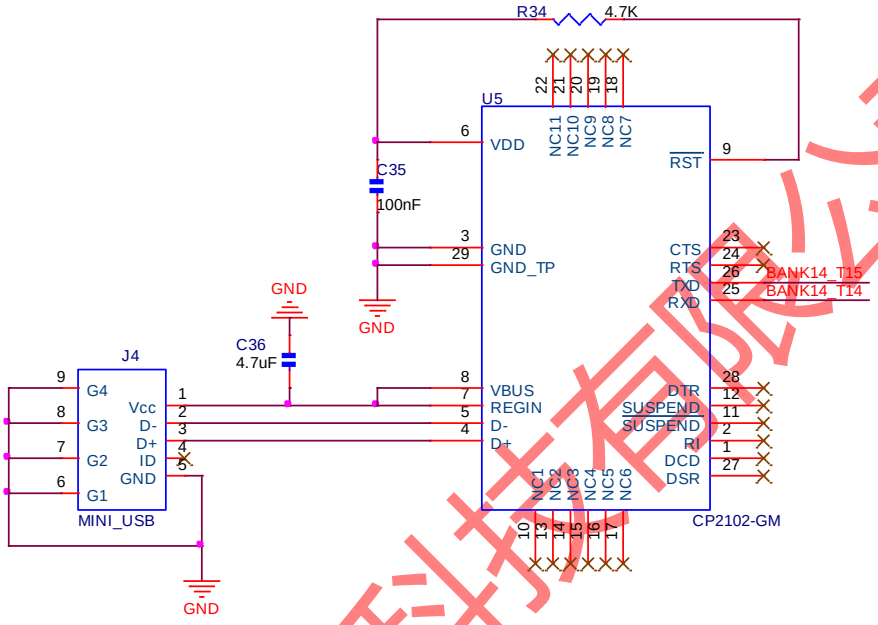
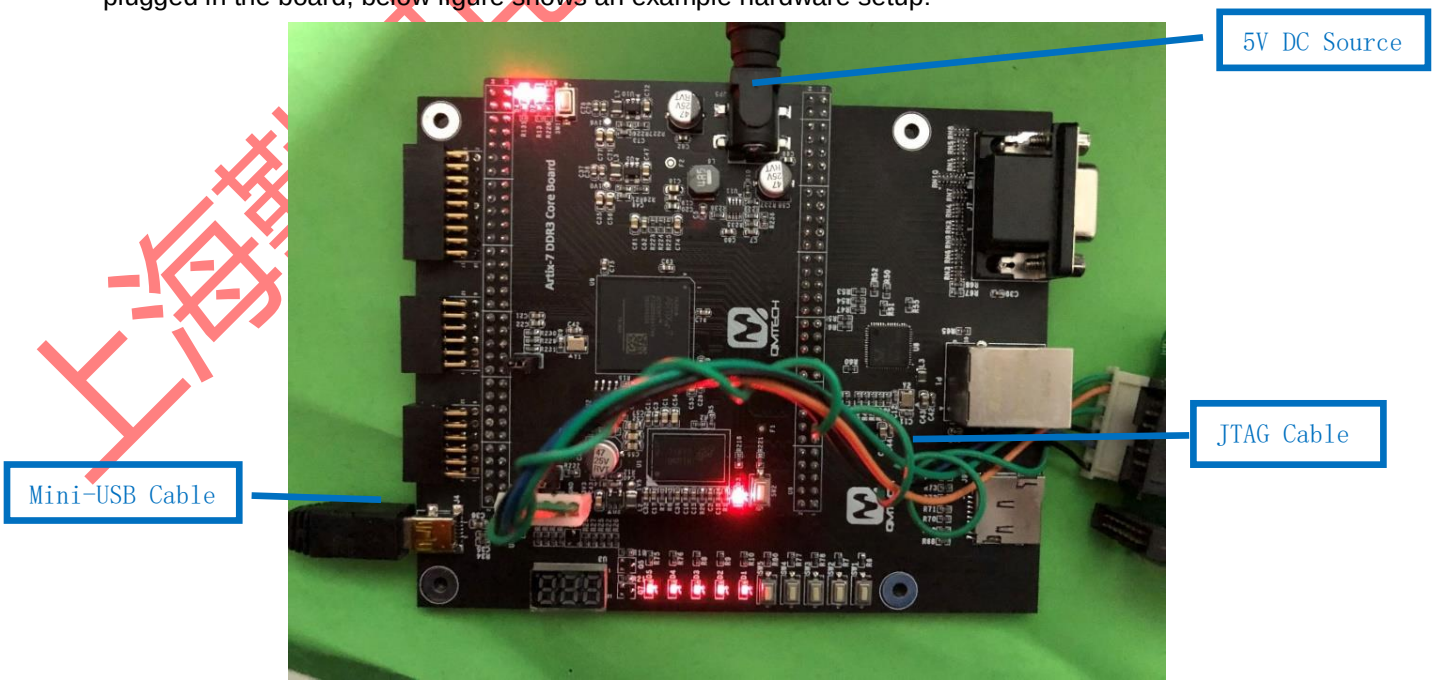


Figure 2-1. CP2102 Hardware Design

Before start to test the CP2102-GMR's USB to UART serial communication function, make sure all the hardware connections of the development kit are correctly connected. Xilinx USB platform cable's VREF, GND, TDI, TMS, TCK, TDO pins shall be connected to QM_XC7A35T_DDR3 core board's JTAG interface. Then power on the development kit with 5V DC power source. Xilinx USB platform cable's indicator LED's color will turn from brown into green. At the same time, the Mini-USB cable shall also be plugged in the board, below figure shows an example hardware setup:



All the test examples are developed in the Vivado2018.2 environment. Open the CP2102 test project located in this release folder: /Software/Test05_usb_uart_cp2102. Below figure shows the project of **uart_top**:

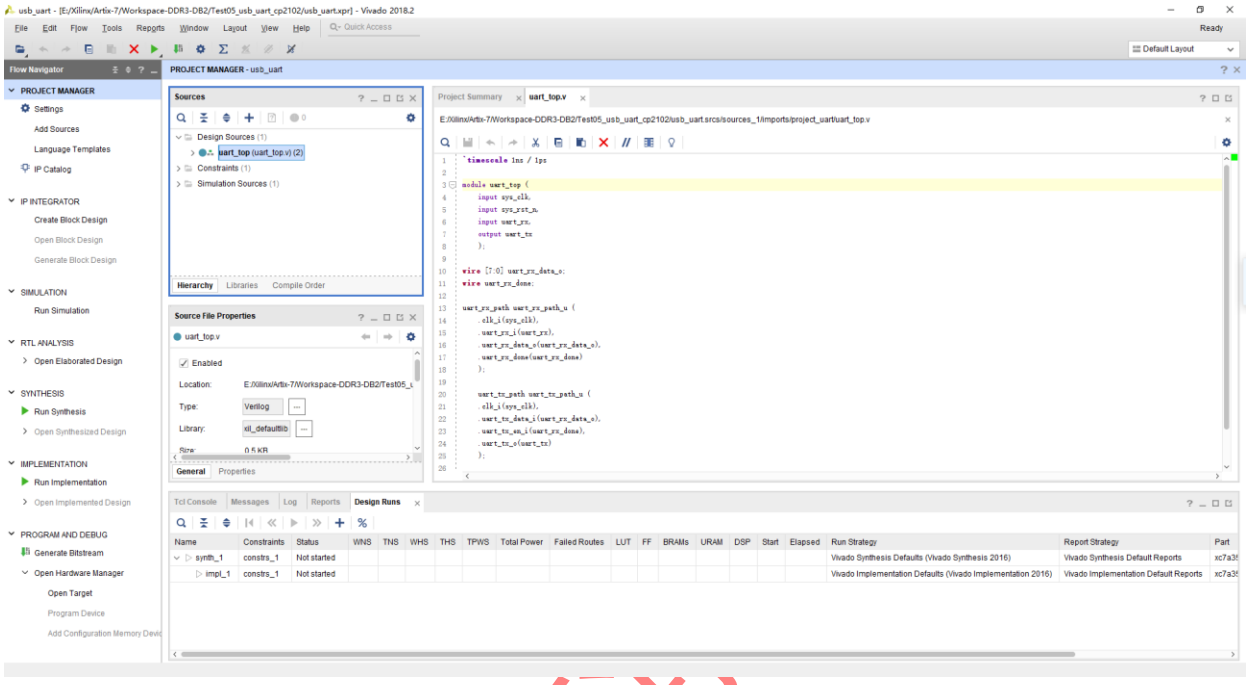


Figure 2-2. CP2102 UART Communication Test Example

In this example project, the default communication parameters are: 9600bps, 8 data bit, No Parity Check, 1 stop bit. If users want to test other communication parameters, change the source code accordingly.

uart_rx_path.v

```
`timescale 1ns / 1ps

module uart_rx_path(
    input clk_i,
    input uart_rx_i,

    output [7:0] uart_rx_data_o,
    output uart_rx_done,
    output baud_bps_tb          //for simulation
    );

parameter [12:0] BAUD_DIV     = 13'd5208;   //波特率时钟，9600bps，50Mhz/9600=5208
parameter [12:0] BAUD_DIV_CAP = 13'd2604;   //波特率时钟中间采样点，50Mhz/9600/2=2604

reg [12:0] baud_div=0;                      //波特率设置计数器
reg baud_bps=0;                             //数据采样点信号
reg bps_start=0;                            //波特率启动标志
always@(posedge clk_i)
begin
    if(baud_div==BAUD_DIV_CAP)              //当波特率计数器计数到采样点时，产生采样信号baud_bps
```

uart_tx_path.v

```
`timescale 1ns / 1ps

module uart_tx_path(
    input clk_i,

    input [7:0] uart_tx_data_i,         //待发送数据
    input uart_tx_en_i,                 //发送发送使能信号

    output uart_tx_o
);

parameter BAUD_DIV     = 13'd5208;      //波特率时钟，9600bps，50Mhz/9600=5208，波特率可调
parameter BAUD_DIV_CAP = 13'd2604;      //波特率时钟中间采样点，50Mhz/9600/2=2604，波特率可调

reg [12:0] baud_div=0;                  //波特率设置计数器
reg baud_bps=0;                         //数据发送点信号,高有效
(* MARKDEBUG = "TRUE" *)reg [9:0] send_data=10'b1111111111;     //待发送数据寄存器，1bit起始信号+8bit有效信号+1bit结束信号
(* MARKDEBUG = "TRUE" *)reg [3:0] bit_num=0;                    //发送数据个数计数器
reg uart_send_flag=0;                   //数据发送标志位
reg uart_tx_o_r=1;                      //发送数据寄存器，初始状态位高
```

After the CP2102 communication test project correctly synthesized, implemented and generated *.bit file, users could use Vivado 2018.2 program tool to program the generated *.bit file into FPGA. Below image shows the FPGA program status with program tool.

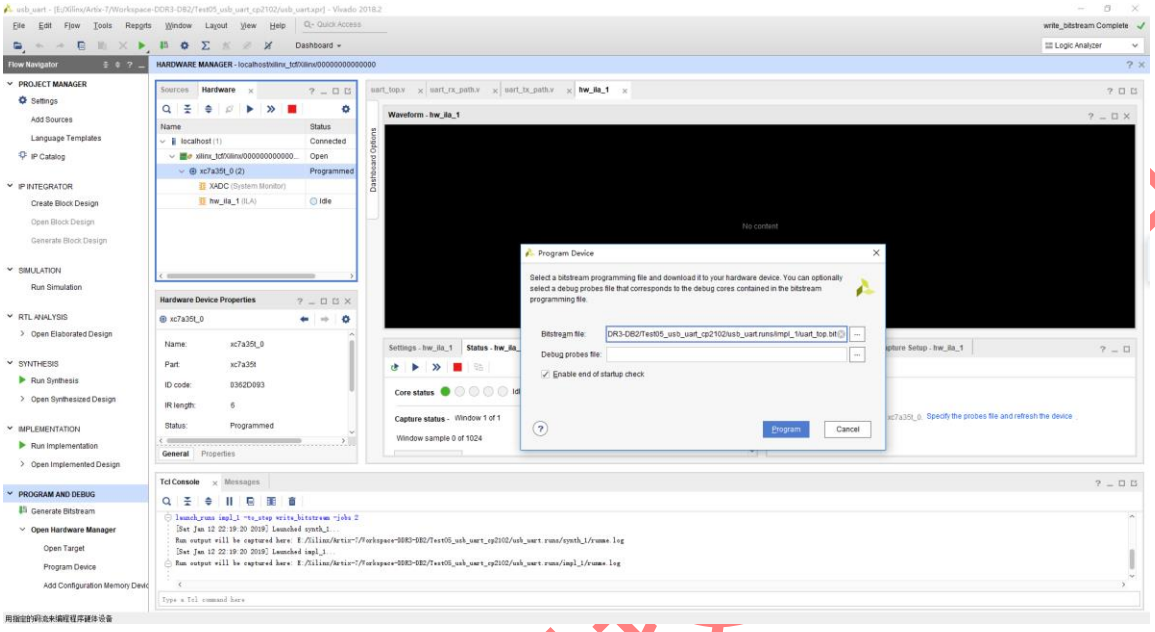


Figure 2-3. Program *.bit File

The CP2102 example test project's main functionality is performing an UART loopback communication. The FPGA program will send the received UART data back to the PC. Below figure shows user employees some PC based UART test tool to send data to FPGA: *http://www.cmsoft.cn QQ:10865600*. After a short while the PC UART test tool will receive the same data stream from FPGA, which means the CP2102 loopback test program is running correctly.

Figure 2-4. UART Loopback Test

3. Experiment (2): VGA Displays

The RGB signal accepted by the color monitor is an analog signal, one for each color, in the range 0V to 0.7V according to the VGA spec. So the digital color signal generated by the video controller should be converted to an analog signal. The daughter board uses resistor to form a voltage divider circuit in combination with the 75 ohm load resistance of VGA monitor. Below image shows the hardware design.

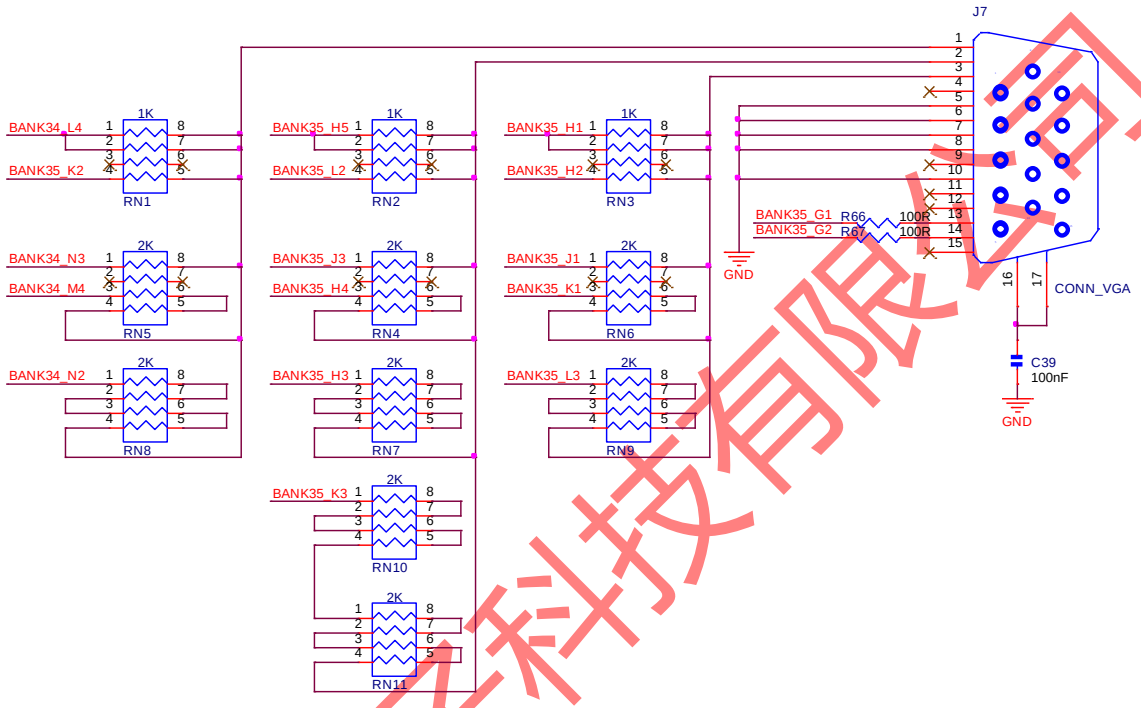
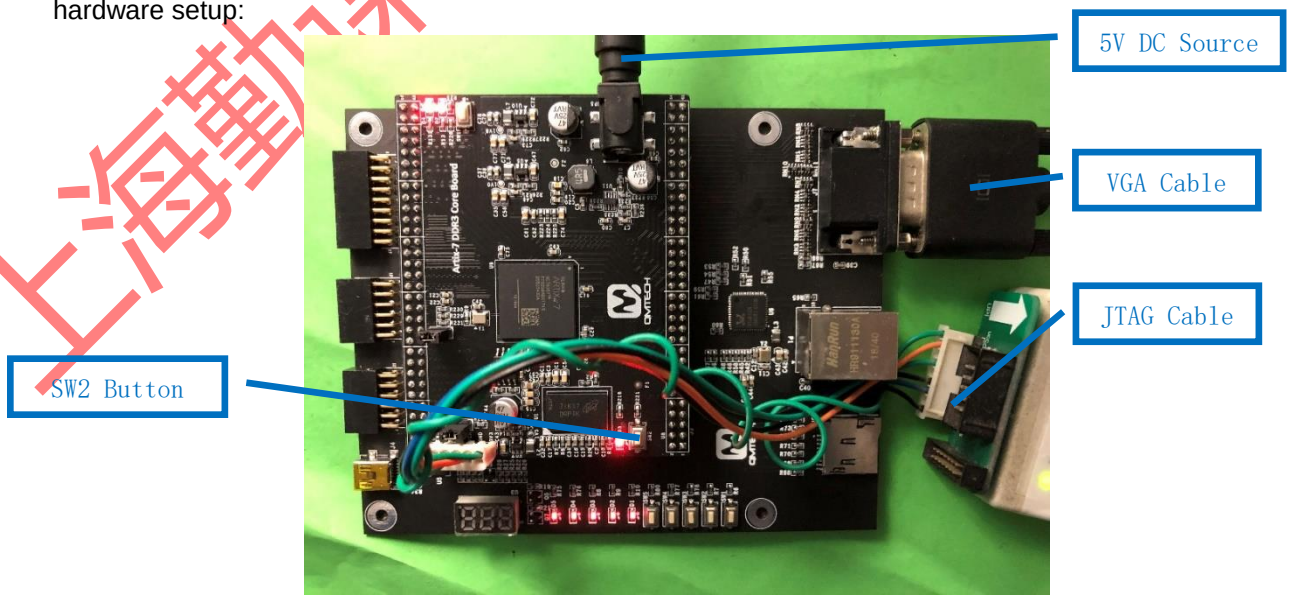


Figure 3-1. VGA Display Hardware Designs

Before start to test the VGA display function, make sure all the hardware connections of the development kit are correctly connected. Xilinx USB platform cable's VREF, GND, TDI, TMS, TCK, TDO pins shall be connected to QM_XC7A35T_DDR3 core board's JTAG interface. Then power on the development kit with 5V DC power source. Xilinx USB platform cable's indicator LED's color will turn from brown into green. At the same time, the VGA cable shall also be plugged in the board, below figure shows an example hardware setup:



Open the VGA test project located in this release folder: /Software/Test09_VGA_Test. Below figure shows the example project of **VGA_test**:

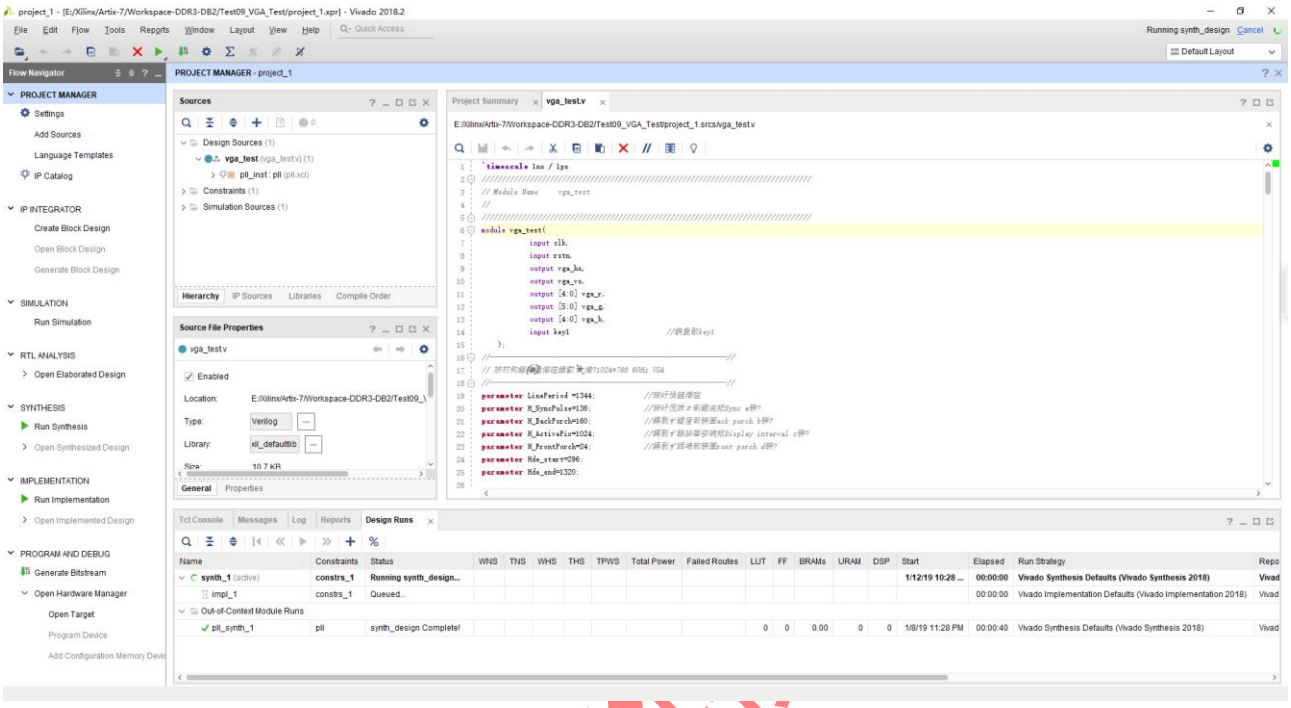


Figure 3-2. VGA Display Function Test

In this example project, the default VGA output resolution parameter is 1024x768@60Hz. If users want to test other display parameters, change the source code accordingly.

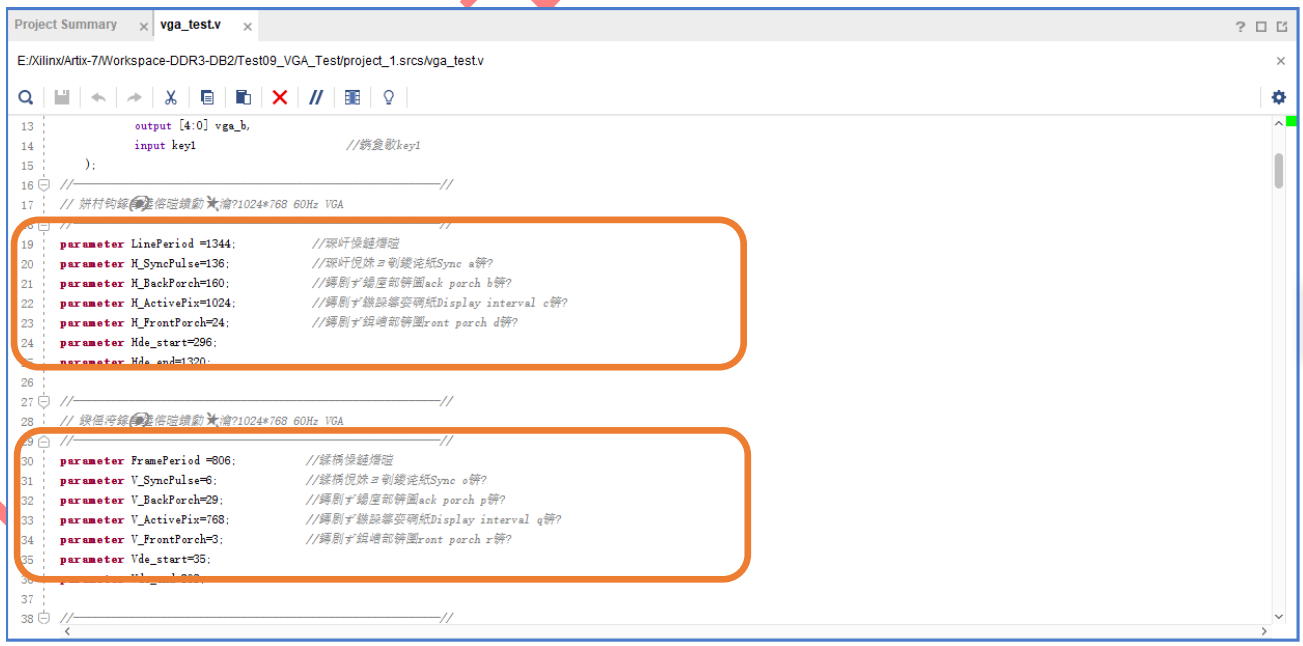


Figure 3-3. VGA Display Parameters

After the VGA display test project correctly synthesized, implemented and generated *.bit file, users could use Xilinx Vivado 2018.2 program tool to program the generated *.bit file into FPGA. Below image shows the FPGA program status with program tool.

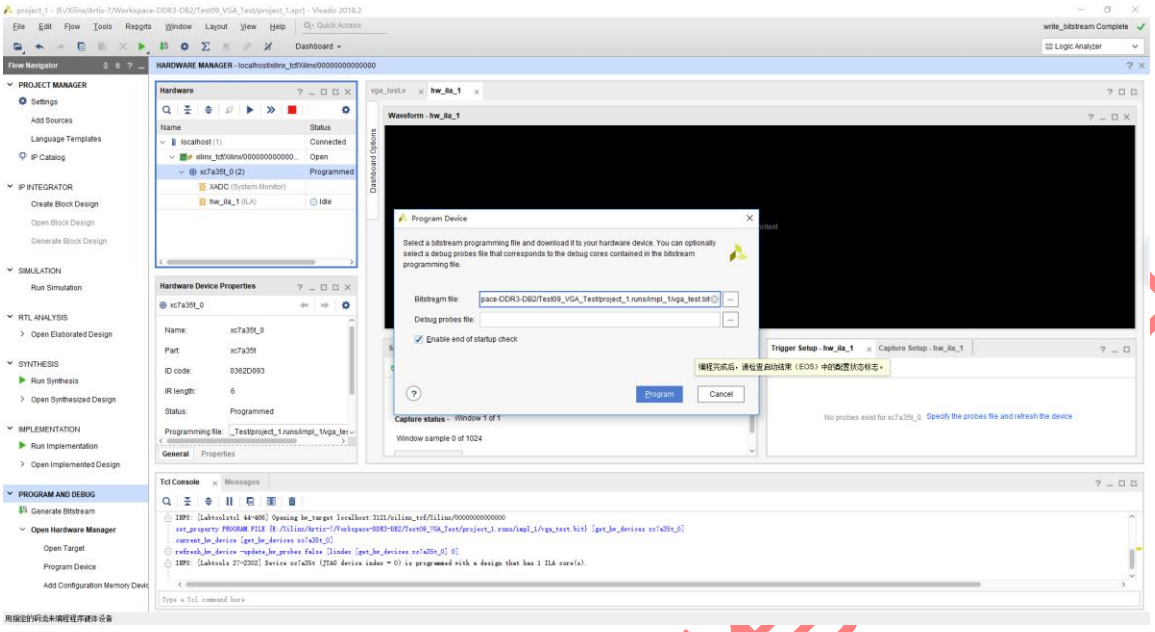


Figure 3-4. Program FPGA

After the FPGA correctly loaded the vga_test.bit file and users pressed the SW2 button on core board, the VGA monitor will display different color patterns. Below image shows the example color bar pattern.

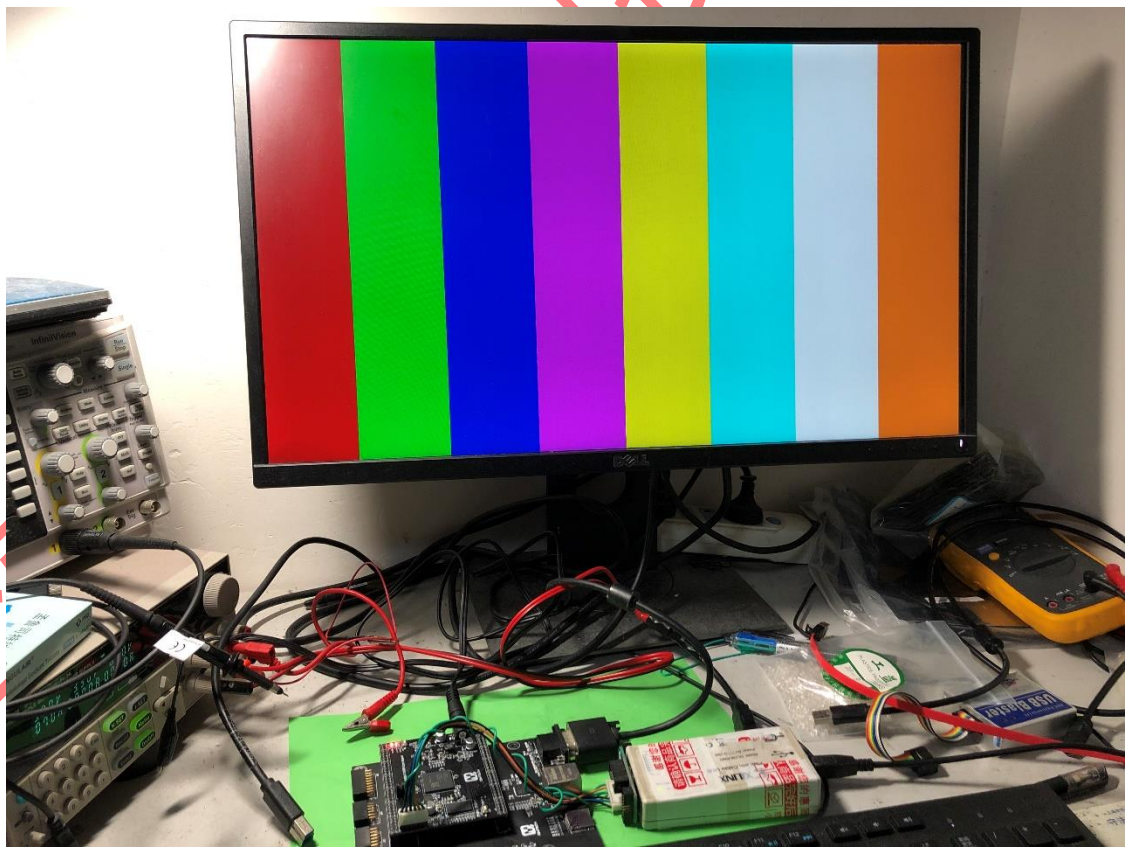


Figure 3-5. VGA Display Test

4. Experiment (3): GMII Ethernet Test

The daughter board uses RTL8211EG to implement the 10M/100M/1000M triple speed ethernet interface. It provides all the necessary physical layer functions to transmit and receive ethernet packets over the CAT.5 UTP cable. The data transfer between PHY and FPGA is via the Gigabit Media Independent Interface(GMII) for 1000Base-T. The RTL8211EG-VB chip supports 3.3V signaling for GMII interface. Below image shows the hardware design of RTL8211EG:

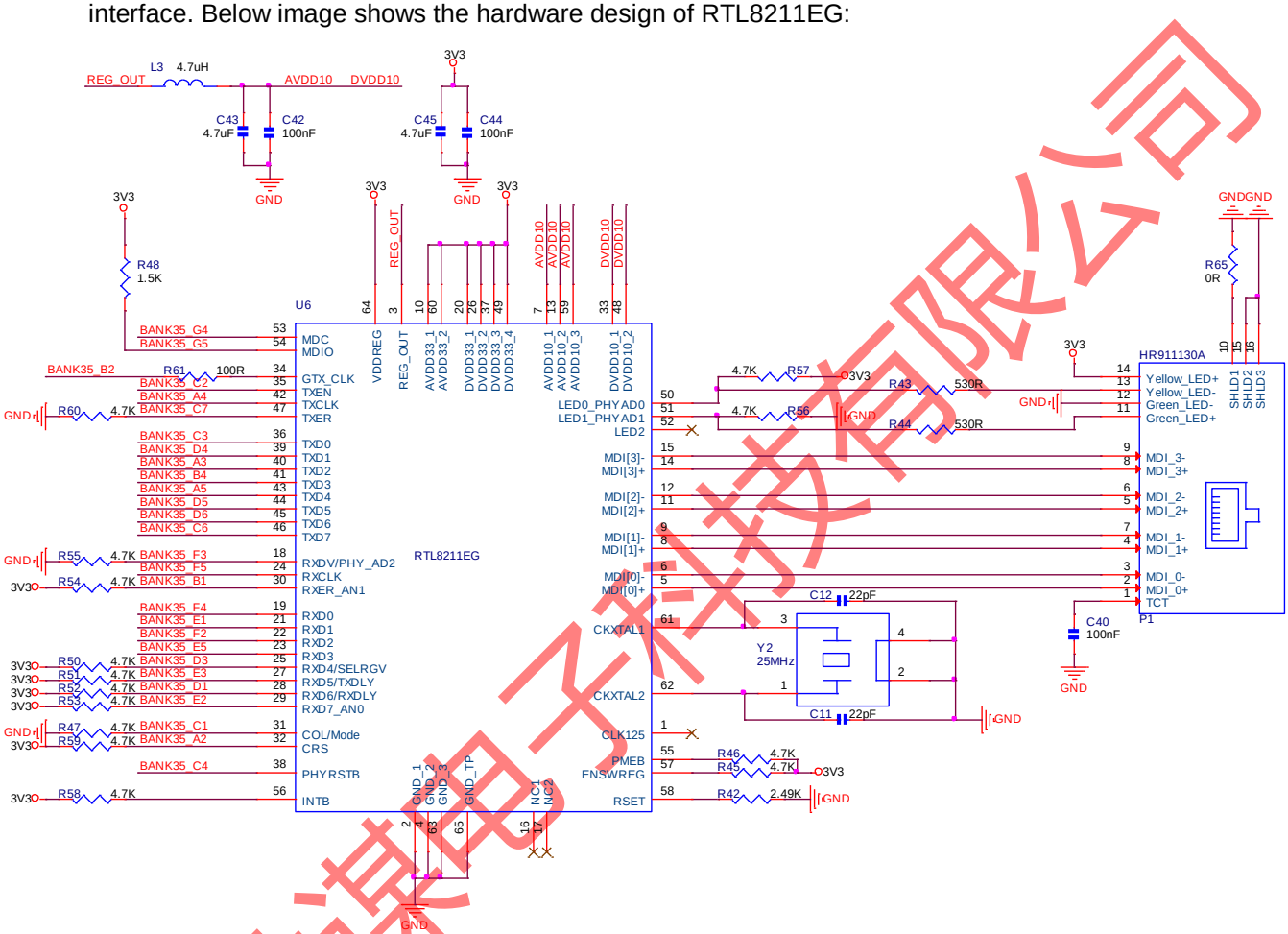


Figure 4-1. RTL8211 Hardware Design

Before start to test the GMII ethernet communication function, make sure all the hardware connections of the development kit are correctly connected. Xilinx USB Platform cable shall be connected to QM_Artix7_XC7A35T core board's JTAG interface. The ethernet cable shall be plugged in the board and the test computer simultaneously. Then power on the development kit with 5V DC power source. Below figure shows an example hardware setup:

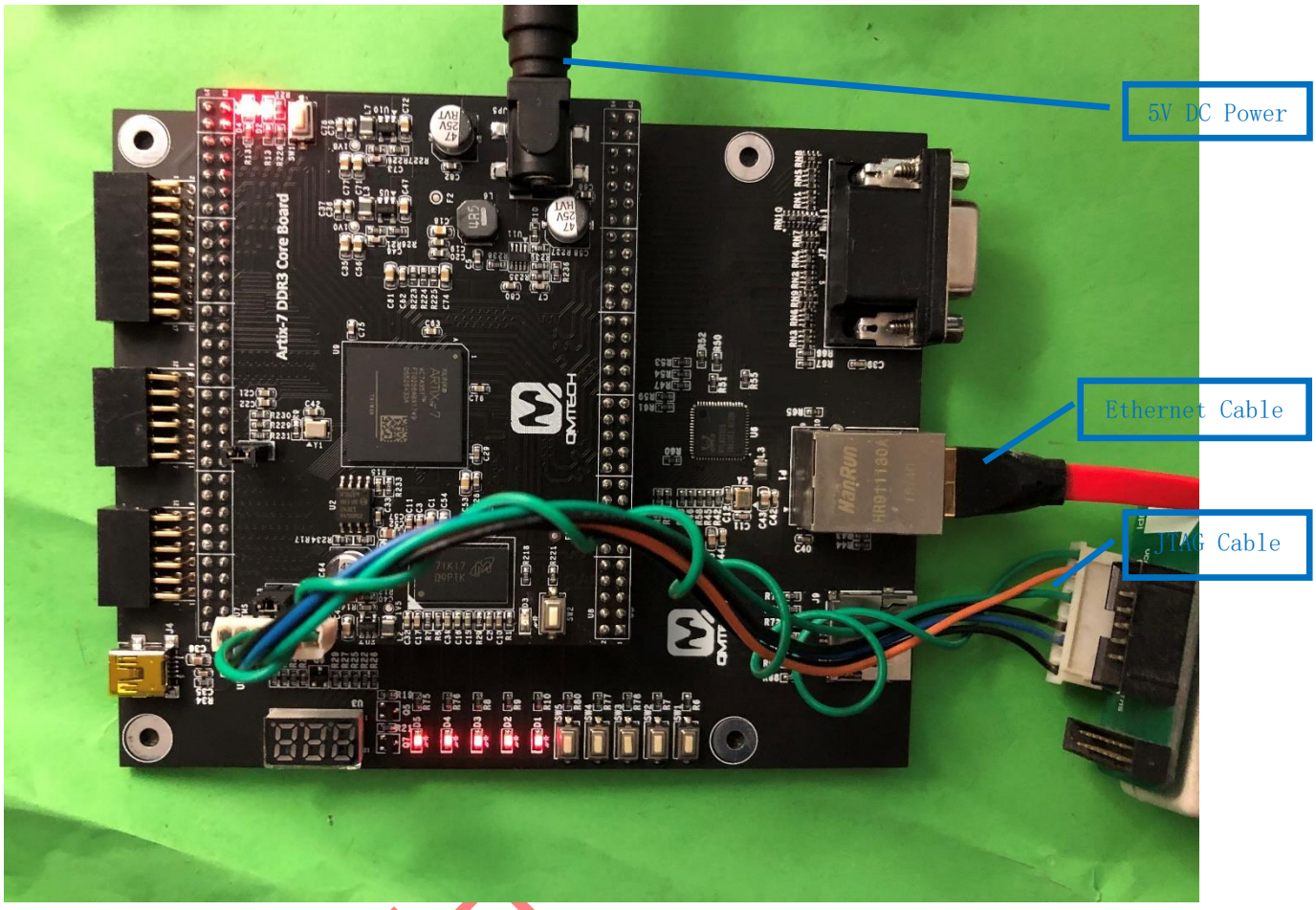
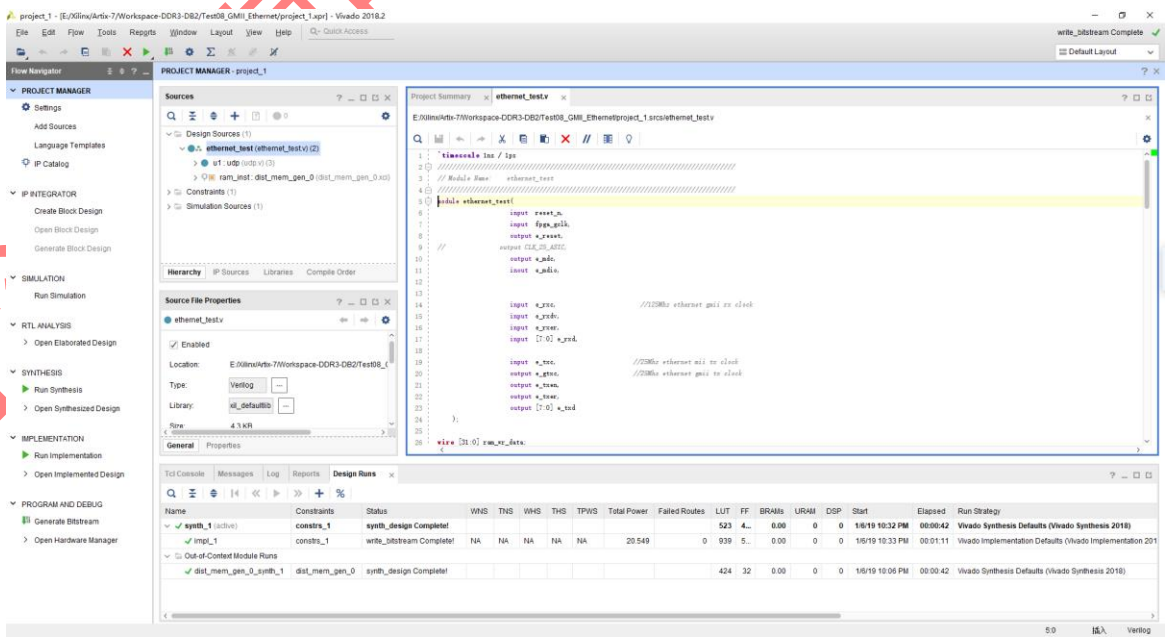


Figure 4-2. Test Setup

Use Vivado2018.2 to open the GMII ethernet test project located in this release folder: /Software/ Test08_GMII_Ethernet. Below figure shows the example project of **ethernet_test**:



After the ethernet test project correctly synthesized, implemented and generated *.bit file, users could use Vivado 2018.2 program tool to program the generated *.bit file into FPGA. Below image shows the FPGA program status with program tool.

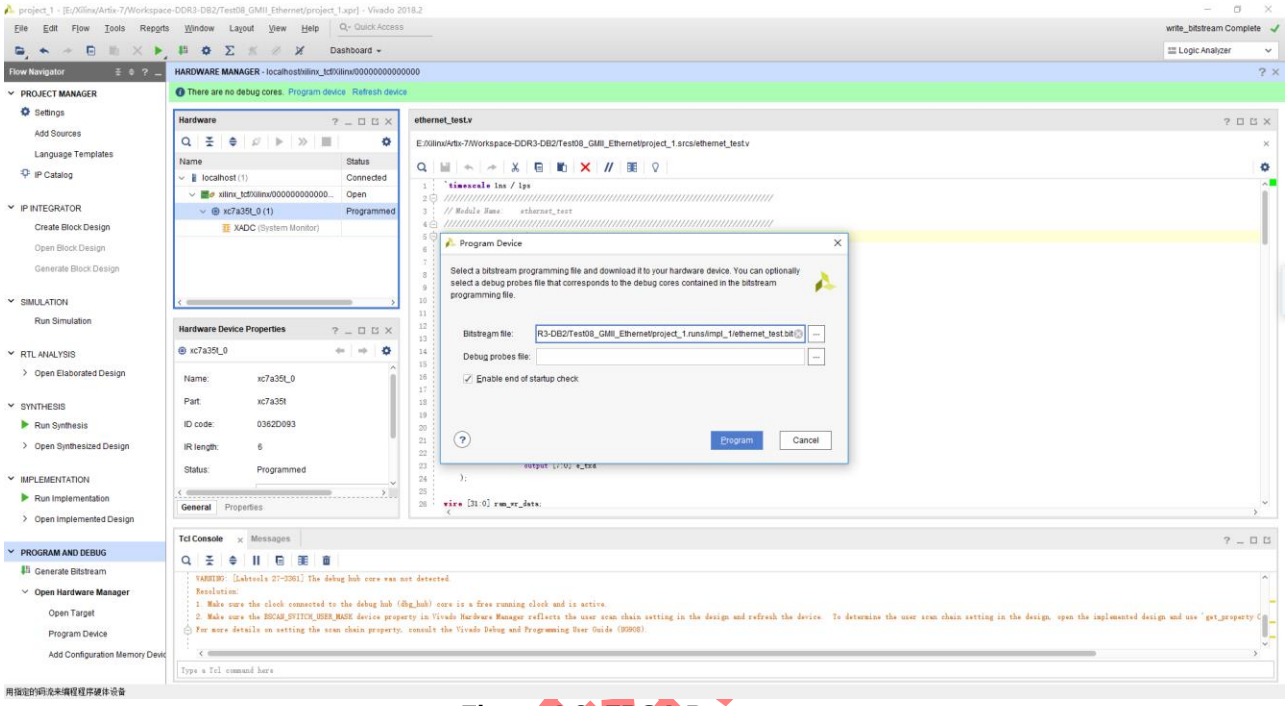


Figure 4-3. FPGA Program

Users could check the ethernet connection status in the Windows OS. Below images shows the ethernet communication speed between the FPGA development board and the test computer is 1Gbps based.

In order to finish this ethernet test, users need to set the Windows's Static IP into 192.168.0.3:

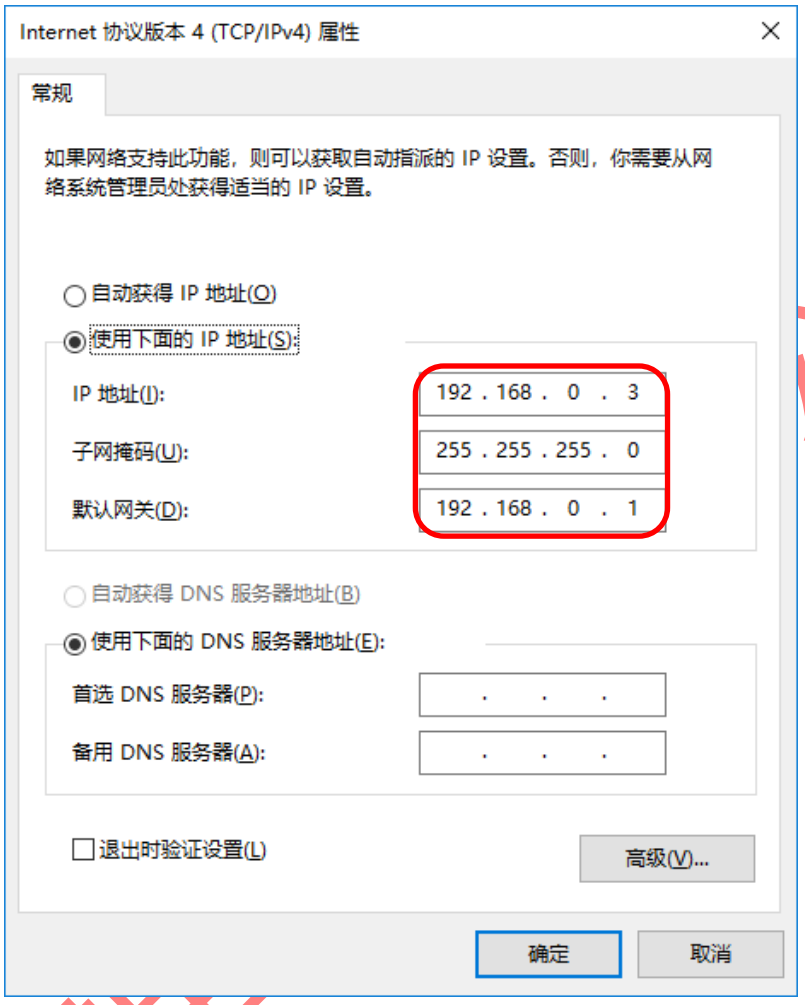


Figure 4-4. Configure Test PC's IP

Run Windows Command Console as administrator. In that DOS type command window bind the development board's IP address(192.168.0.2) and MAC address (00-0a-35-01-fe-c0) by typing command: ARP -s 192.168.0.2 00-0a-35-01-fe-c0:

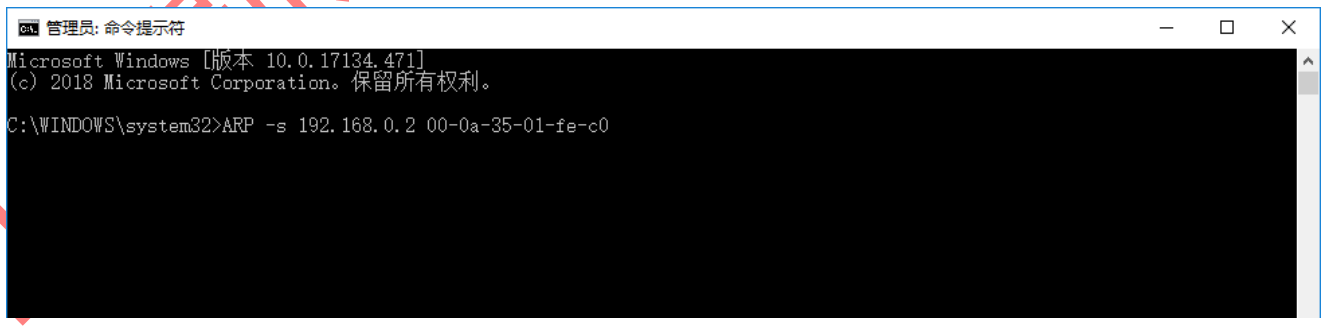


Figure 4-5. Binding IP and MAC

Open the NetAssist ethernet debug tool and set the communication parameters as shown in below figure. The NetAssist tool will receive the test data “HELLO QMTECH BOARD” sent from development board. Then press the【Send】button to send the test data http://www.cmsoft.cn QQ:10865600 to the FPGA development board. In response, the FPGA will send back test data to the test PC.

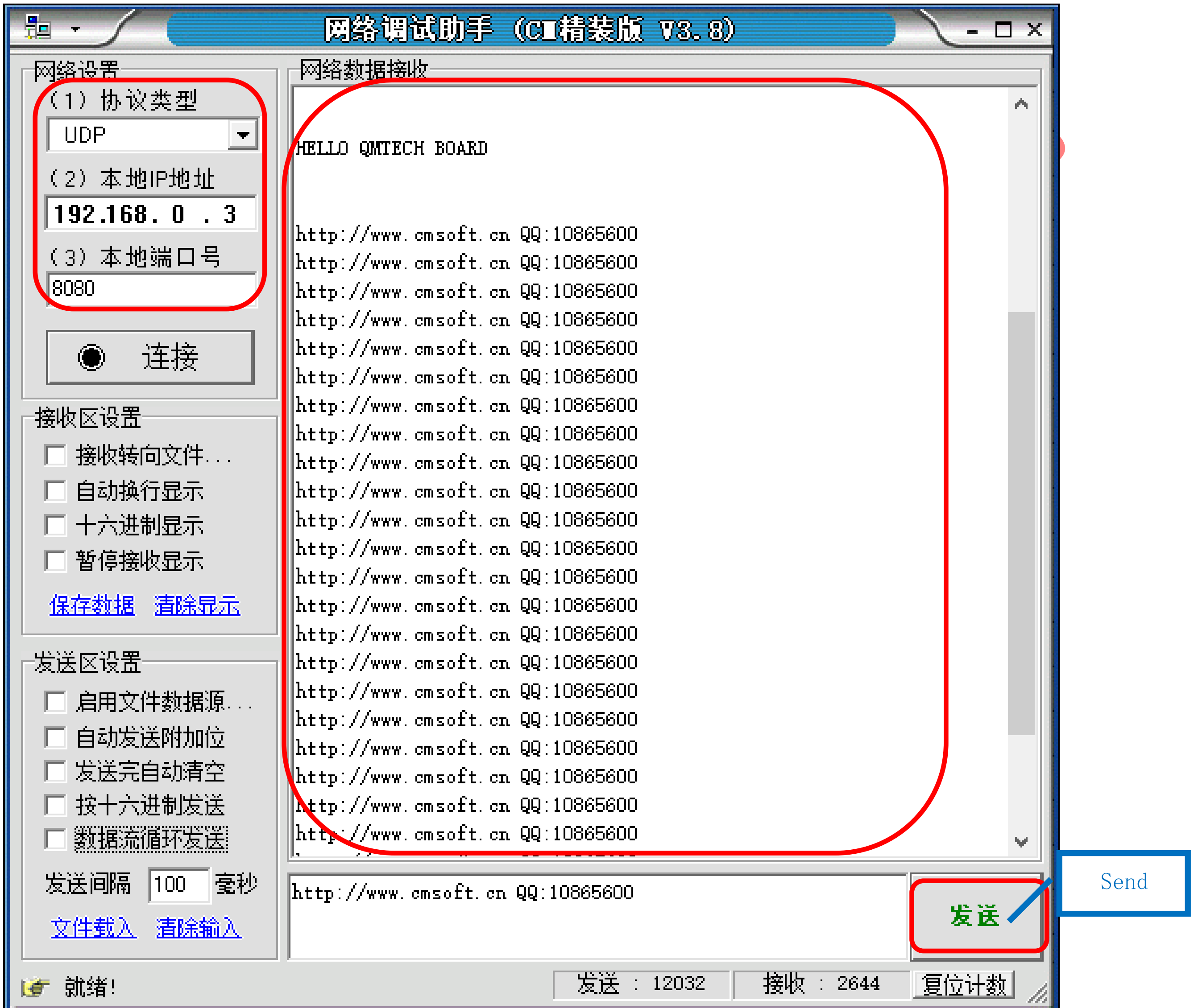


Figure 4-6. GMII Ethernet Test Result

5. Experiment (4): MicroSD Card Test

The daughter board provides a MicroSD slot to extend MicroSD or TF card. In this experiment, we uses 8GB Micro SDHC card provided by Kingston. The Micro SDHC card meets the specification of SD V2.0. Below image shows the hardware design of MicroSD slot:

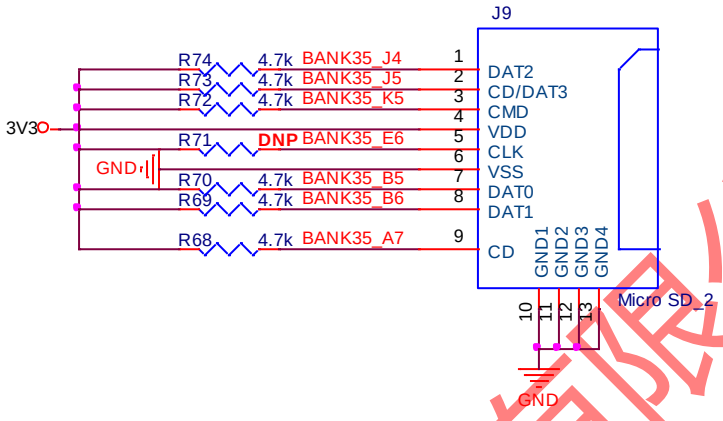


Figure 5-1. MicroSD Hardware Design

Before start to test the MicroSD read/write function, make sure all the hardware connections of the development kit are correctly connected. Xilinx Platform USB cable shall be connected to QM_Artix7_XC7A35T core board's JTAG interface. The Kinston 8 GB micro SD card shall be plugged in the board. Then power on the development kit with 5V DC power source. below figure shows an example hardware setup:

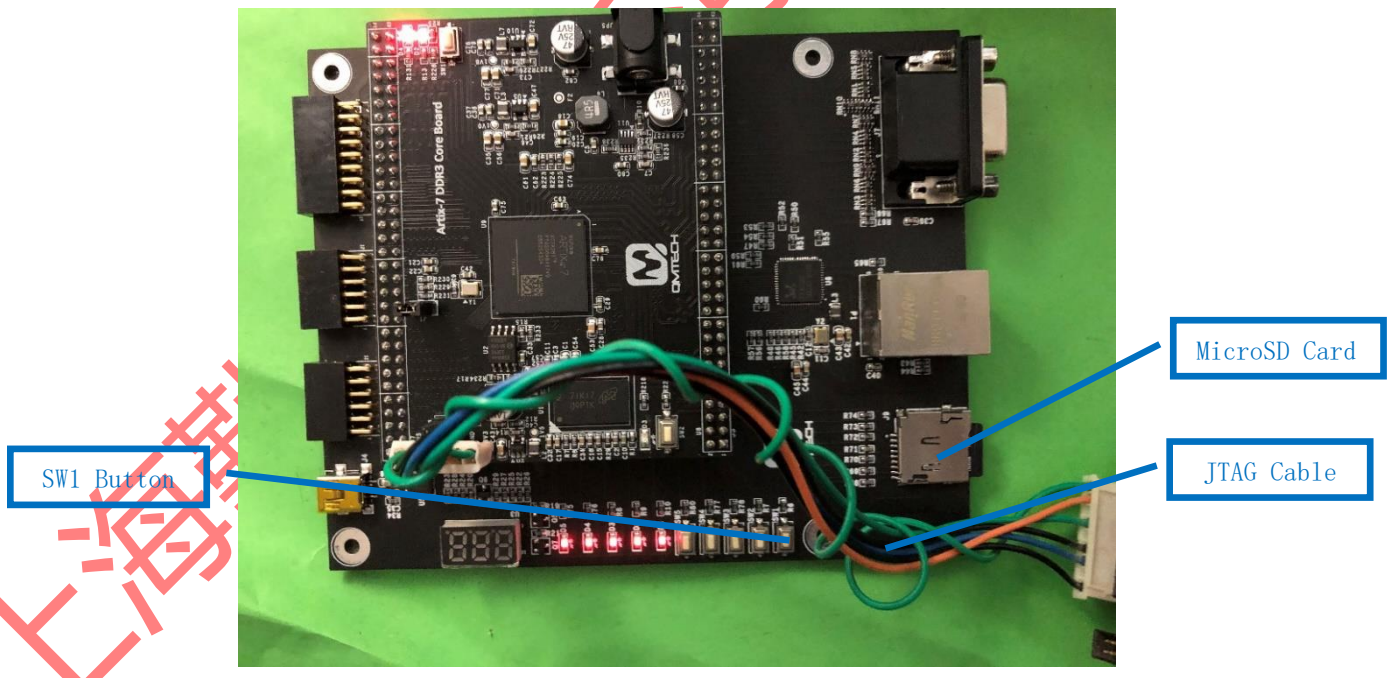


Figure 5-2. Test Environment Setup

In this test example, the MicroSD card is working under SPI mode which could be easily handled by FPGA. SPI interface only has four wires: CS, MOSI, MISO, CLK. The clock frequency for the SPI interface is 10MHz which is divided by the on board 50MHz crystal directly. After Power-On, the MicroSD card enters SD mode and users need to send command to make the MicroSD switch to SPI mode. Then users need to follow the sequence shown in below figure to initialize the MicroSD card. Users may refer to the SD v2.0 spec for more details regarding to the Read and Write protocol.

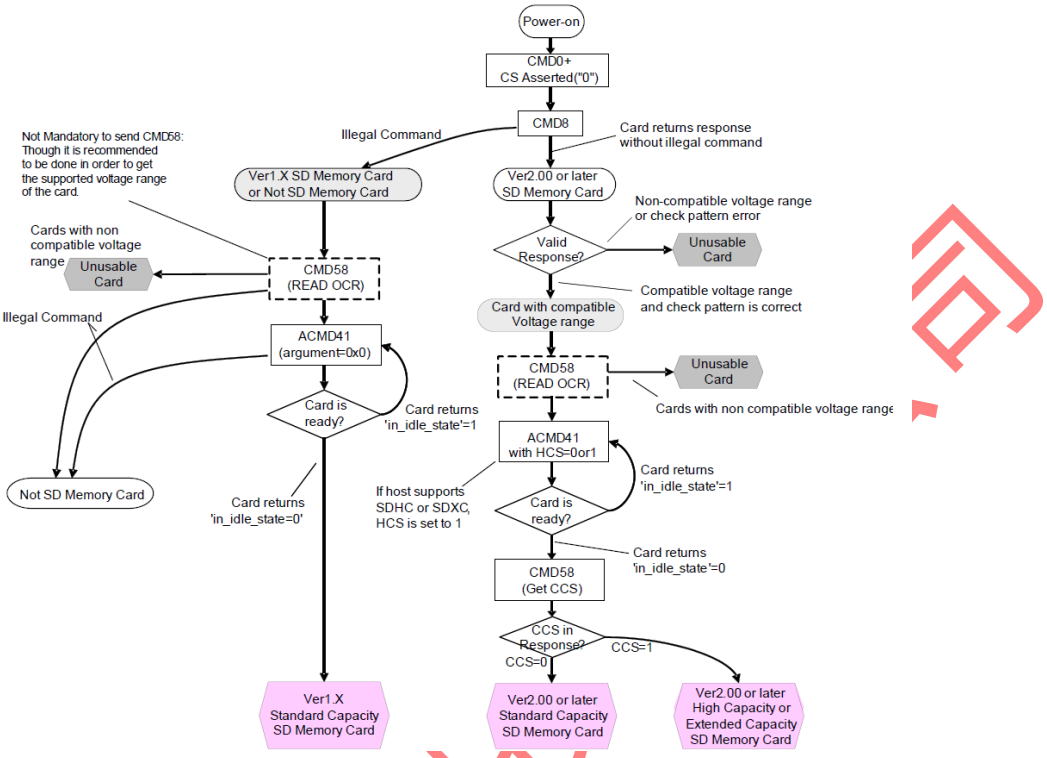
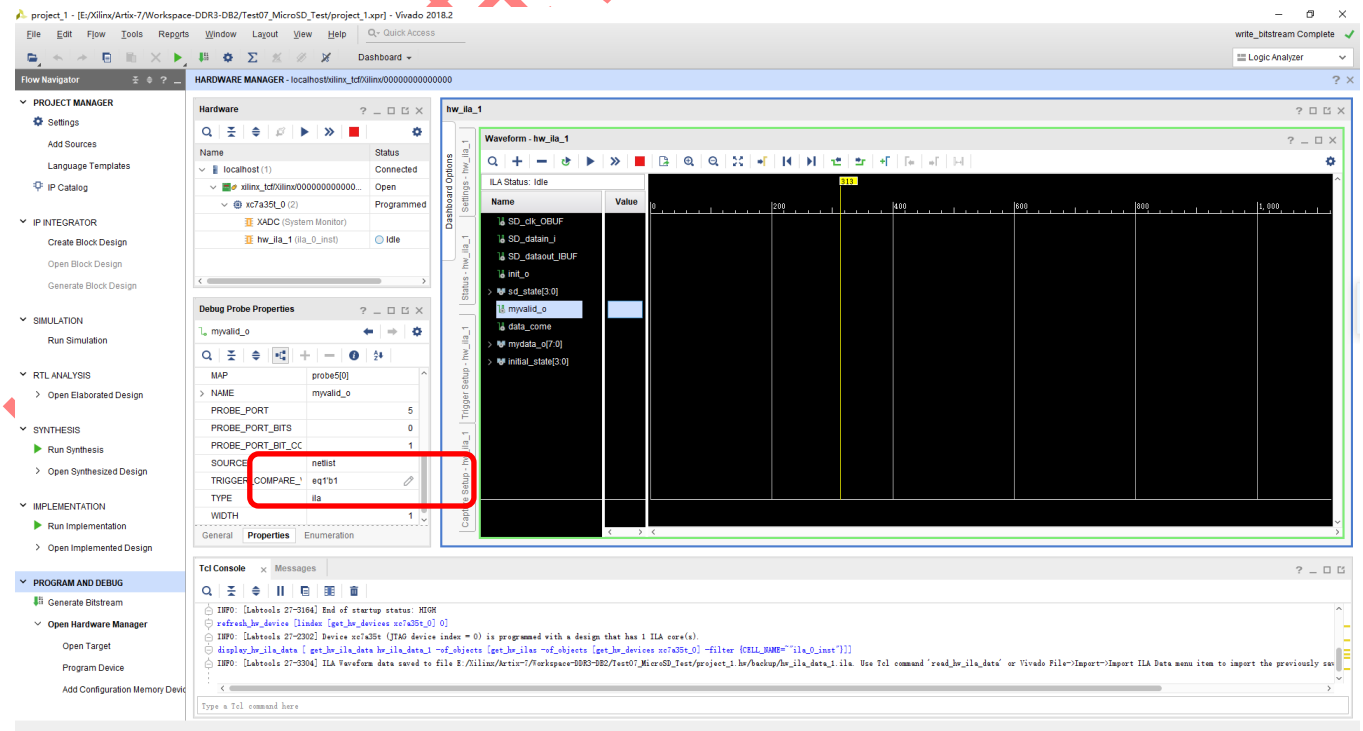


Figure 5-3. Initialize Sequence

After correctly initialized the MicroSD, the test program will write a batch of test data into the MicroSD card and then read back all these written value for further comparison. Here we use ILA to monitor these data transfer between the FPGA and the MicroSD card. Users may follow the ILA settings shown in below figure to observe the transactions. The sampling clock frequency is using on board 50MHz crystal and the trigger signal for sampling is myvalid_o. When myvalid_o goes high, there will be data comes from MicroSD card.



After the MicroSD test project correctly synthesized, implemented and generated *.bit file, users could use Xilinx program tool to program the generated *.bit file into FPGA. And then press the button SW1 on FPGA daughter board to trigger the test. After a short while, the ILA will stop capturing the data immediately after the myvalid_o signal goes high. Then we can see the init_o is already in high status which means the MicroSD has already been correctly initialized. And the data signal mydata_o displays the data read out from MicroSD card.

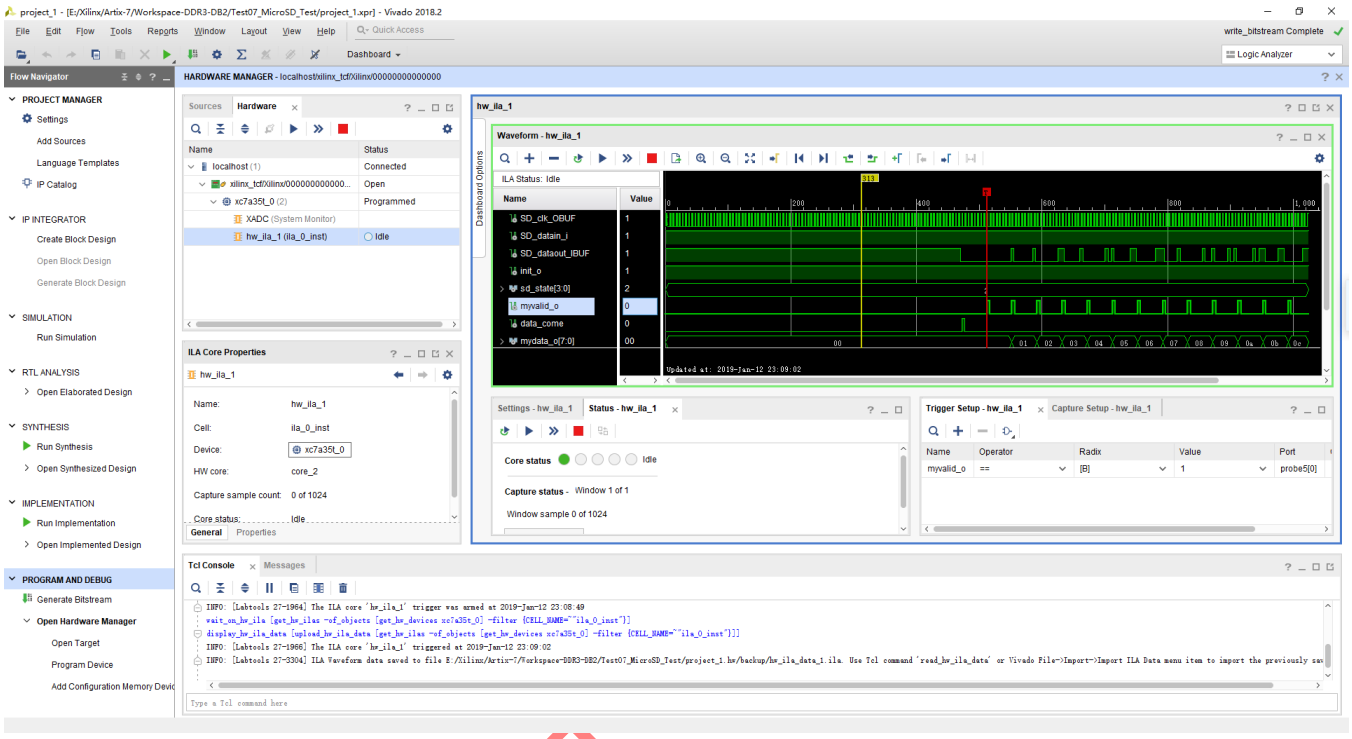


Figure 5-4. Data Write and Read Waveform

6. Reference

[1] ug470_7Series_Config.pdf
[2] ds181_Artix_7_Data_Sheet.pdf
[3] ug475_7Series_Pkg_Pinout.pdf
[4] n25q_64a_3v_65nm.pdf
[5] MT41J128M16JT-125K.pdf
[6] MP2359.pdf
[7] NCP1529-D.PDF

7. Revision

Doc. Rev.	Date	Comments
0.1	1/1/2018	Initial Version.
1.0	12/1/2018	V1.0 Formal Release.
3.0	12/01/2019	V2.0 Formal Release.