

Team: System Solutions

# Basic Functionality Tests

Test Number 1

Alexis Wilson  
2-4-2020

# TABLE OF CONTENTS

<b>TABLE OF CONTENTS .....</b>	<b>1</b>
<b>REVISION HISTORY .....</b>	<b>2</b>
<b>TEST PROCEDURE .....</b>	<b>3</b>
PURPOSE OF TEST .....	3
REQUIREMENTS .....	3
TEST IMPLEMENTATION .....	3
<i>Setup</i> .....	3
<i>Procedure</i> .....	5
<i>LED Functionality Test 1</i> .....	5
<i>LED Functionality Test 2</i> .....	5
<i>Function Generator Test</i> .....	6
<i>Oscilloscope Test</i> .....	6
<i>Results</i> .....	7
<i>LED Functionality Test 1</i> .....	7
<i>LED Functionality Test 2</i> .....	7
<i>Function Generator Test</i> .....	7
<i>Oscilloscope Test</i> .....	7
<b>CONCLUSION .....</b>	<b>7</b>

## Revision History

<b>Name</b>	<b>Date</b>	<b>Revision Made</b>	<b>Version Number</b>
Alexis Wilson	02/04/2020	Creation of Procedure	1.0

## **Purpose of Test:**

The overall purpose of this test is to verify that the STEMLab 125-14 Red Pitaya board functions as it should straight out of the box. This test will help with one's personal knowledge of the board as well as become familiar with some of the general capabilities the STEMLab 125-14 Red Pitaya can perform.

## **Requirements:**

To perform this test, one will need to have a STEMLab 125-14 Field Programmable Gate Array (FPGA) and a power cable for the FPGA. Then one will also need an ethernet cable, and a computer/laptop or two ethernet cables, a switch/hub, and a computer/laptop. No internet or special equipment will be required. No special skills will be required, but knowledge in Python will aid in understanding the basic test code that is programmed onto each board.

**CAUTION:** Before touching the STEMLab 125-14 board, make sure to ground oneself by touching metal or wearing an ESD strap so that no electric discharge can cause harm to the components on the board.

## **Test Implementation:**

### **Setup:**

1. Take the STEMLab 125-14 board out of the box and plug in the FPGA to an outlet.
  - a. The power plug in is on the left side of the board far left of the ethernet cable.
2. Take an ethernet cable and plug it back into ethernet slot on the FPGA.
3. Take the other end of the ethernet cable and plug it into a LAN slot of the network switch/hub.
  - a. If no network switch/hub, plug into the Ethernet port connected to the computer/laptop.
  - b. If FPGA Ethernet is plugged into switch/hub, take the second Ethernet cable and plug it into the Ethernet port of the computer/laptop. Shown in Figure 1 below:

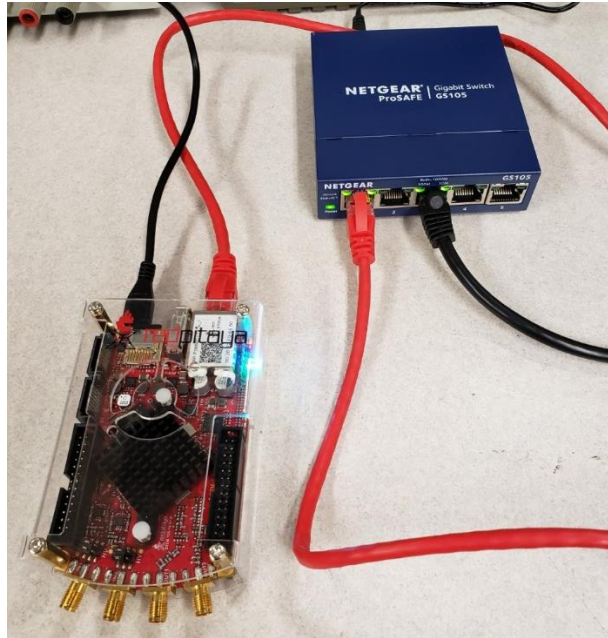


Figure 1: Set Up Between FPGA and Switch/Hub

4. Once the board is on and link light is coming from the switch for both ethernet connections, use an internet browser to connect to the board through Ethernet
  - a. Type `rp-xxxxxx.local/` and press enter
  - b. The resulting screen should pop up if connection is true shown in Figure 2 below:

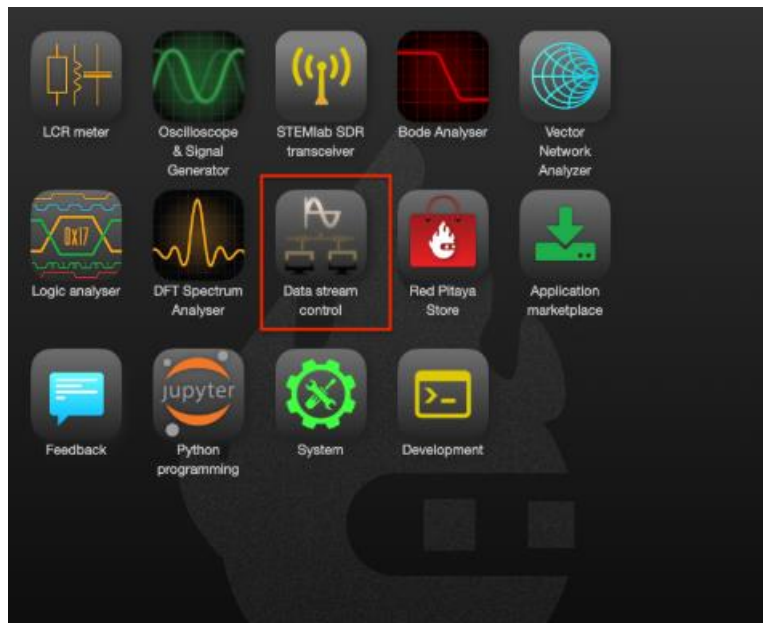


Figure 2: Red Pitaya Interface Page

5. Once that screen pops up, setup is complete.

## Procedure:

### LED Functionality Test 1

1. Click on the Python Programming section on the Red Pitaya main screen
2. Once the Interface opens the Jupyter notebook, select the Blink LED preset notebook
  - a. Figure 3 shows the Code that will be initially tested

#### Overlay

First thing we will do is to load the FPGA bitstream and device tree overlay.

We will do this by click on the following cell to select it and press the run button (▶) above or by pressing the key combination *Shift+Enter*.

```
In [ ]: from redpitaya.overlay.mercury import mercury as FPGA
overlay = FPGA()
```

The next line loads the *led* module that allows python to control LEDs on the STEMlab board.

```
In [ ]: LED = FPGA.led
```

There are 8 user controllable LEDs on board, so we need to tell python which LED we want to control. In the next line we declare an object **led1** that controls the 2<sup>nd</sup> LED and also sets the initial state of the LED, which in our case is off (0).

```
In [ ]: led1 = LED(1, 0)
```

To turn the LED *on* execute the following cell.

```
In [ ]: led1.write(1)
```

To turn the LED *off* execute the following cell.

```
In [ ]: led1.write(0)
```

*Figure 3: Blink LED On then Off*

3. Once the notebook is loaded, click on the first code cell and run it by using ‘Esc’ + ‘Enter’
4. Run the next two cells after that. You should notice that the LEDs on the board should all still be off.
5. Run the cell that says ‘led1.write(1)’
  - a. This should cause the LED to light up
6. Run the cell that says ‘led1.write(0)’
  - a. This should turn the LED off

### LED Functionality Test 2

1. Click on the Python Programming section on the Red Pitaya main screen
2. Once the Interface opens the Jupyter notebook, select the Blink LED preset notebook
  - a. Figure 4 shows the Code that will be tested

## Blink LED

To make the LED blink we will need to write a loop that will turn the LED on and off and we will also need to import the Time library so we can slow down the execution of the cell so we can see the LED turn on and off.

1. Import Time library.

```
In [ ]: import time
```

1. Turn the LED on and off 10 times every half of seconds.

```
In [ ]: for _ in range(10):  
        led1.write(1)  
        time.sleep(0.5)  
        led1.write(0)  
        time.sleep(0.5)
```

To close the led1 object execute the following cell.

```
In [ ]: led1.close()
```

*Figure 4: Recursively Blink LED*

3. Once the notebook is loaded, scroll down to the Blink LED section and run the 'import time' cell by using 'Esc' + 'Enter'
4. Next, run the 'for \_ in range(10)' cell.
5. Once ready to end the test, run the 'led1.close()' cell

## Built-In Function Generator Test

1. Click on the Oscilloscope & Signal Generator option on the Red Pitaya main screen
2. Click the 'Out1' located on the right-hand side of the screen
3. Click 'Show'
4. Choose an amplitude of 1 Volt and a frequency of 1000 Hz.

## Built-In Oscilloscope Test

1. Attach the SMA side of a BMA-SMA cable to the fast-analog input of the Red Pitaya
2. Attach the BMA side of the BMA-SMA cable to a Function Generator
3. Click on the Oscilloscope & Signal Generator option on the Red Pitaya main screen
4. Push a 1000 Hz signal from the Function Generator to the Red Pitaya
5. Press Run on the top right-hand side of the Oscilloscope screen
6. Click on the signal labeled In1
  - a. Change the operator to frequency and press done
  - b. Change the operator to P2P and press done
7. Change the voltage to a 10 Volt range instead of the defaulted 1 Volt range

## Results:

This section would explain the desired output from the test. It would explain why this result is desired and what it could mean if it is not accurate. It would include the necessary visual elements to show the outcome of the test.

### LED Functionality Test 1

1. At step 5 of this test, we should see the first LED turn on
2. At step 6 of this test, we should see the first LED turn off

### LED Functionality Test 2

1. At step 4 of this test, one should see LED1 turn on for 0.5 seconds and then turn off for 0.5 seconds
  - a. It will repeat this ten times

### Built-In Function Generator Test

1. At step 4, one should see a signal on the screen that has an amplitude of 1 Volt an

### Built-In Oscilloscope Test

1. After step 7, one should see a mirrored signal on the Oscilloscope screen as the 1000 Hz signal that is being pushed into the Red Pitaya

## Conclusion:

The purpose of these tests is to taste the basic built in functionality of the STEMLab 125-14 Red Pitaya FPGAs. These tests will give a specific output due to the scripts being premade and standardized. If these tests do not give a correct result, there is something wrong with the board and it may need to be replaced.