

## **Internet of Things Laboratory**

### **Assignment 3**

**Prajakta Deokule**

**C22019221332**

**4329**

**Problem Statement: Write an application using Beaglebone Black board to control the operation of hardware simulated traffic signal.**

#### **1.Explain the use of libraries with examples.**

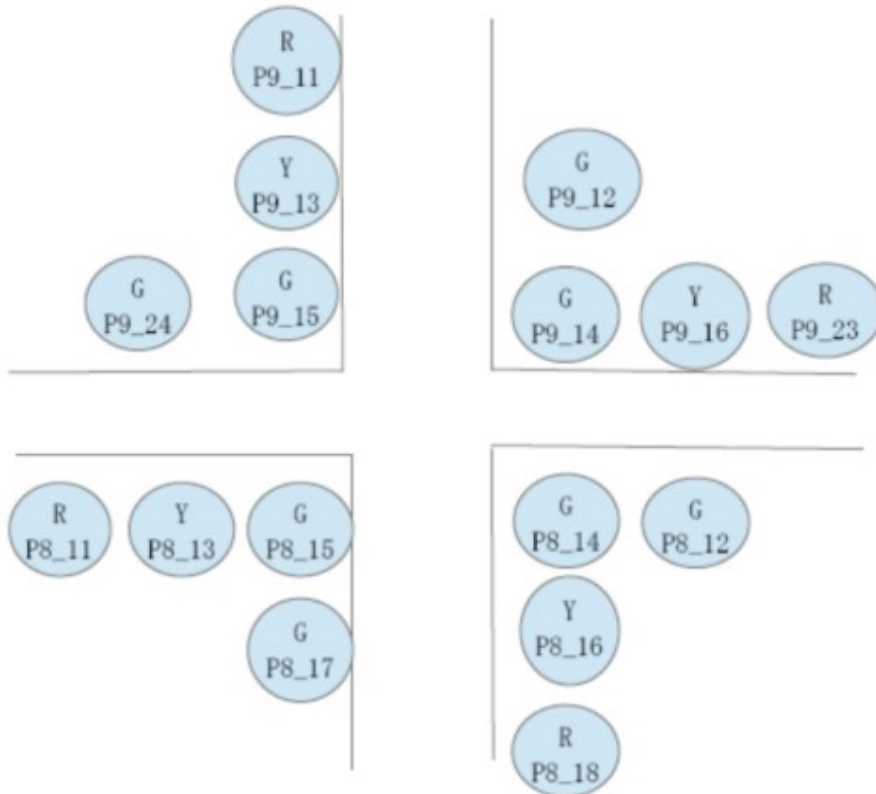
##### **a. Adafruit\_BBIO library in BBB**

Adafruit\_BBIO is an API to enable GPIO, PWM, ADC, UART, SPI and eQEP (Quadrature Encoder) hardware access from Python applications running on the Beaglebone.

##### **b. Time Library**

This library is used to create delays during the runtime of the python code/thread. The program stays idle during the specified time(in seconds)passed as parameter to time.sleep(t) function.

## 2. Draw the layout of the Traffic controller board.



## 3. List the pin numbers used for implementation.

P8\_11 => GPIO1\_13, P8\_12 => GPIO1\_12

P8\_13 => GPIO0\_23, P8\_14 => GPIO0\_26

P8\_15 => GPIO1\_15, P8\_16 => GPIO1\_14

P8\_17 => GPIO0\_27, P8\_18 => GPIO2\_1

P9\_11 => GPIO0\_30, P9\_12 => GPIO1\_28

P9\_13 => GPIO0\_31, P9\_14 => GPIO1\_18

P9\_15 => GPIO1\_16, P9\_16 => GPIO1\_19

P9\_23 => GPIO1\_17, P9\_24 => GPIO0\_15

## 4. Draw FRC pin diagram with pins on it.



## 5. Write sample code for Dancing LED.

```
import Adafruit_BBIO.GPIO as GPIO
```

```
import time
```

```
led=['P9_11','P9_13','P9_15','P9_24','P9_12','P9_14','P9_16','P9_23',  
'P8_11','P8_13','P8_15','P8_17','P8_14','P8_16','P8_18','P8_12']
```

```
GPIO.cleanup()
```

```
for i in range(len(led)):
```

```
    GPIO.setup(led[i],GPIO.OUT)
```

```
    GPIO.output(led[i],GPIO.LOW)
```

```
    time.sleep(10)
```

```
while True:
```

```
    GPIO.output("P9_11",GPIO.HIGH)
```

```
time.sleep(5)
GPIO.output("P9_11",GPIO.LOW)
GPIO.output("P8_11",GPIO.HIGH)
time.sleep(5)
GPIO.output("P8_11",GPIO.LOW)
GPIO.output("P8_18",GPIO.HIGH)
time.sleep(5)
GPIO.output("P8_18",GPIO.LOW)
GPIO.output("P9_23",GPIO.HIGH)
time.sleep(5)
```

## **6. Write a program for traffic signal with output as HIGH, LOW after every 5 seconds.**

```
import Adafruit_BBIO.GPIO as GPIO
import time

GPIO.setup("P9_11",GPIO.OUT)
GPIO.setup("P9_13",GPIO.OUT)
GPIO.setup("P9_15",GPIO.OUT)
GPIO.setup("P9_24",GPIO.OUT)
GPIO.setup("P9_12",GPIO.OUT)
GPIO.setup("P9_14",GPIO.OUT)
GPIO.setup("P9_16",GPIO.OUT)
GPIO.setup("P9_23",GPIO.OUT)
GPIO.setup("P8_11",GPIO.OUT)
GPIO.setup("P8_13",GPIO.OUT)
GPIO.setup("P8_15",GPIO.OUT)
GPIO.setup("P8_17",GPIO.OUT)
GPIO.setup("P8_14",GPIO.OUT)
GPIO.setup("P8_16",GPIO.OUT)
```

```

GPIO.setup("P8_18",GPIO.OUT)
GPIO.setup("P8_12",GPIO.OUT)
while True:
    num=0

    GPIO.output("P8_16",GPIO.LOW) # Y low
    GPIO.output("P9_13",GPIO.LOW) #Y
    GPIO.output("P9_11",GPIO.LOW) # R
    GPIO.output("P8_18",GPIO.LOW) #R
    GPIO.output("P8_15",GPIO.LOW) # G
    GPIO.output("P8_17",GPIO.LOW) #G
    GPIO.output("P9_12",GPIO.LOW) #G
    GPIO.output("P9_14",GPIO.LOW) #G
    GPIO.output("P9_15",GPIO.HIGH) #G
    GPIO.output("P9_24",GPIO.HIGH) #G
    GPIO.output("P8_14",GPIO.HIGH) #G
    GPIO.output("P8_12",GPIO.HIGH) #G
    GPIO.output("P8_11",GPIO.HIGH) #R
    GPIO.output("P9_23",GPIO.HIGH) #R
    time.sleep(5)
    # Yellow Light Blinking (Q1 & Q3)
    while num<3:
        GPIO.output("P9_16",GPIO.LOW)
        GPIO.output("P8_13",GPIO.LOW)
        time.sleep(1)
        GPIO.output("P9_16",GPIO.HIGH)
        GPIO.output("P8_13",GPIO.HIGH)
        time.sleep(1)
        num=num+1
    # Q1 & Q3 Green
    num=0

```

```

GPIO.output("P8_13",GPIO.LOW) #Y
GPIO.output("P9_16",GPIO.LOW) #Y
GPIO.output("P9_11",GPIO.HIGH) #R
GPIO.output("P8_18",GPIO.HIGH) #R
GPIO.output("P9_15",GPIO.LOW) #G
GPIO.output("P9_24",GPIO.LOW) #G
GPIO.output("P8_14",GPIO.LOW) #G
GPIO.output("P8_12",GPIO.LOW) #G
GPIO.output("P8_11",GPIO.LOW) #R
GPIO.output("P9_23",GPIO.LOW) #R
GPIO.output("P9_14",GPIO.HIGH) #G
GPIO.output("P9_12",GPIO.HIGH) #G
GPIO.output("P8_15",GPIO.HIGH) #G
GPIO.output("P8_17",GPIO.HIGH) #G
time.sleep(5)

# Yellow Light Blinking (Q2 & Q4)

while num<3: # no of blinks for Yellow Light
    GPIO.output("P8_16",GPIO.LOW)
    GPIO.output("P9_13",GPIO.LOW)
    time.sleep(1)
    GPIO.output("P8_16",GPIO.HIGH)
    GPIO.output("P9_13",GPIO. HIGH)
    time.sleep(1)
    num=num+1

GPIO.cleanup()

```

## **7. Mention steps to copy file from PC to Beaglebone Board.**

1. Edit your IOTL program and save file as test.py
2. Open a new terminal
3. Type 'su' and input password
4. Now use sftp command connect to BBB as sftp <ip addr of BBB>

5. Now use put command to copy file from PC to BBB – put test.py
6. Now for BBB terminal compile file as – python test.py
7. Every time you modify test.py on PC perform step 4 to upload changes to BBB.

