

Hardware Testing of Components

- Zain Rajani (c0752681)

Group # 8



Outline

- Testing Device
- Testing of LDR
- Testing of IR Sensor
- Testing of GSM Module
- Testing of BB-AI with connection to Wifi
- Conclusion
- References

Testing Device

- For any components before it is been put to use in the actual device or circuit one needs to check if the device is working suitably and providing suitable amount of voltage or current so as to make sure that it wouldn't damage the circuit or cause a huge damage at a later stage.
- Thus, before going to implement our actual circuit we decided to check if the devices that we require to complete the project needs to be tested to verify that they are working fine and also incase we encounter any connection issue
- If the device were not working we might have to order a new one after proper troubleshooting has been carried out.

Testing Device

- Thus in order to check if the circuit is functioning properly, we decided to use a testing device which would be Arduino and this device has some of the inbuilt examples already existing thus it will make our task simpler.
- Also Arduino is cheap and has an easy to use IDE. And since we just want to see the working and we are not likely to go in depth of the project we can use Arduino for this purpose.
- During the testing of each hardware components we shall see that how they are connected to the Arduino, what is their input and output voltages at various pins, are the components attached performing well and whether the LEDs are working as expected. Lastly, we will verify if they are providing necessary output or no which would be desired for the project.

Arduino Uno

- The Arduino Uno in all testing purpose is powered through the USB cable and to verify its working when powered up it will start its Power LED and it will blink the Tx and Rx LED along with the LED connected to the pin 13 of the board.
- When can also verify by simply running an in-built code to blink the LED to be confirm about the same.
- Since in our project we use LED separately thus during the testing procedure we shall attach an LED to the sensor to prove that LEDs we carry are working well. The voltage for the same shall also be measured if required
- To write the code in the Arduino one needs to write the code in the IDE aka Sketch, save it next compile it and then run it on the board.

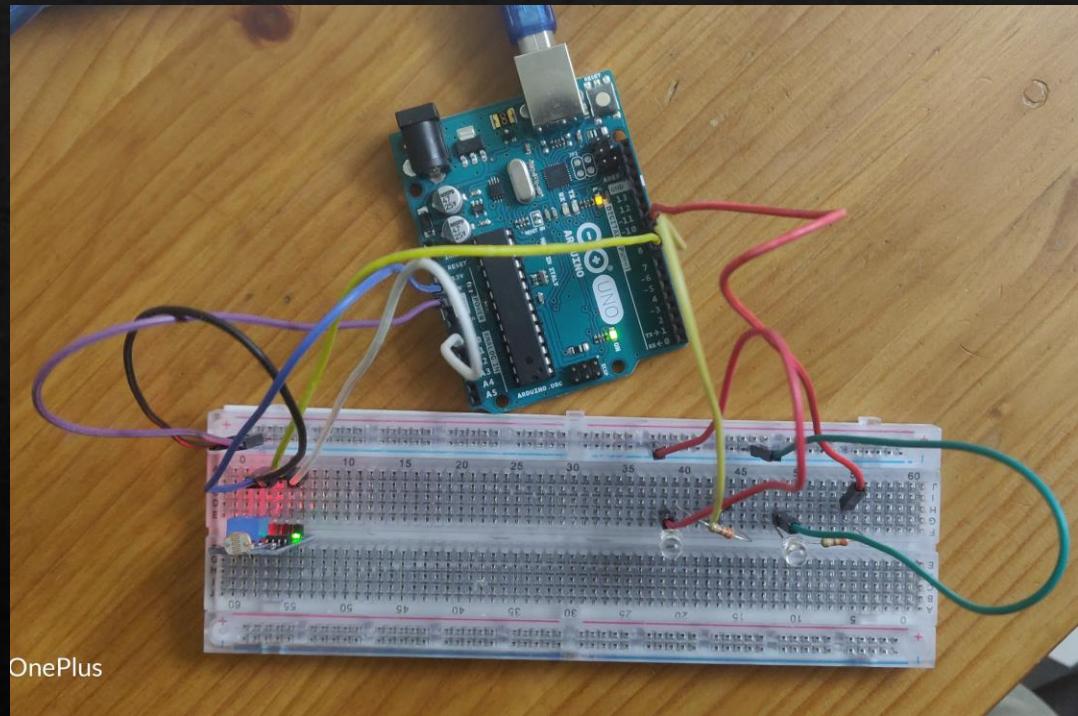
Testing the LDR Circuit

- LDR stands for a light dependent resistor and its primary function is to monitor the intensity of light and used as an indicator in many cases to check if it's a day time or night time, etc.
- Thus in the LDR module that we carry we have the following pins and they are connected to Arduino as follows:

LDR Module Pin	Arduino Pin Connected
VCC	5V
GND	GND
D0	8
A0	A0

Testing LDR Circuit

- We connected the circuit as shown in the figure and the connection were made as in the table on the previous slide.
- During the testing we expected that the LDR should be able to decide if its dark or bright depending upon the intensity falling on the sensitive material on the sensor. Also, we attach an LED to this circuit to get it on incase it detects night/ dark
- For testing purpose we had made sure that we tested the analog as well as digital output of the circuit.



Testing LDR Circuit – Sample Code and Output

The image shows the Arduino IDE interface with two windows. The left window is titled "sketch_oct17a | Arduino 1.8.13" and displays the following C++ code:

```
File Edit Sketch Tools Help
sketch_oct17a
void setup()
{
    Serial.begin(9600);
}

void loop()
{
    unsigned int AnalogValue;
    AnalogValue = analogRead(A0);
    Serial.println(AnalogValue);
}
```

The right window is titled "/dev/ttyACM0" and shows the serial output window with the following data:

Output
50,
350
356
391
356
350
388
365
345
382
374
344
373
383
346
363
388
351
356

At the bottom of the serial window, there are settings: "Autoscroll" (checked), "Show timestamp" (unchecked), "Both NL & CR" (selected), "9600 baud" (selected), and "Clear output".

* This testing was for the Analog pin available on the LDR Module

Testing of LDR Circuit Sample Code and Output

The screenshot shows the Arduino IDE interface. The left window displays the code for 'sketch_oct17b' in Arduino 1.8.13:

```
sketch_oct17b | Arduino 1.8.13
File Edit Sketch Tools Help
sketch_oct17b §
void setup() {
  pinMode(8, INPUT); //Set the pin mode
  pinMode(7, OUTPUT);
  Serial.begin(9600); //initialise serial monitor
}

void loop() {
  int temp=digitalRead(8); //assign value of LDR sensor to a temporary variable
  Serial.println("Intensity="); //print on serial monitor using ""
  Serial.println(temp); //display output on serial monitor
  delay(200); //Stop the program for 200 ms = 0.2 seconds
  if(temp==HIGH) //HIGH means,light got blocked
    digitalWrite(7,HIGH); //if light is not present,LED on
  else
    digitalWrite(7,LOW); //if light is present,LED off
}

Done uploading.

Sketch uses 2162 bytes (6%) of program storage space. Maximum is 32256 bytes.
Global variables use 198 bytes (9%) of dynamic memory, leaving 1850 bytes for local variables.
```

The right window shows the serial monitor output for port '/dev/ttyACM0' at 9600 baud:

```
/dev/ttyACM0
Intensity=
1
Intensity=
0
Intensity=
0
Intensity=
0
```

At the bottom of the serial monitor window, there are checkboxes for 'Autoscroll' and 'Show timestamp', and dropdown menus for 'Both NL & CR', '9600 baud', and 'Clear output'.

* This testing was done on the digital pin on the LDR Module

Testing LDR Module - Result

- Upon successful upload of the code to the Arduino we see that we are getting what we required but this is not the end. Further we need to check if the other parameters are working fine as well. Thus we create the following table:

Testing Parameter	Result
Power LED on the Module	ON
Input Supply Voltage	5.03 V
Analog Pin Voltage	0.76 - 5.01 V
Digital Pin Voltage	0.73 - 2.9 V
Output LED Working	ON & OFF depends on situation
Potentiometer Working As expected	Yes, when varied the intensity cut off value changes
Result	All parameters looks to be perfect and ideal and the voltage levels are good enough to supply to our main processing unit. No issues faced during the connections and interface etc.

Testing of IR Sensor

- IR sensor consisting of LED and IR pair. This sensor is mostly used to sense any object or for line follower robots etc.
- The black LED is the IR LED which the transmitter and other is receiver which is a simple LED that transmits IR rays. The sensor when detects any IR rays it goes high indicating a presence or object or whatever it is programmed for.
- The sensor is attached with the potentiometer which can help vary the distance of sense and adjust the accuracy of the sensor. The sensor we had received is having the opposite principle in running when the object is present it indicates a low signal and when a object is not present it indicates a high signal.

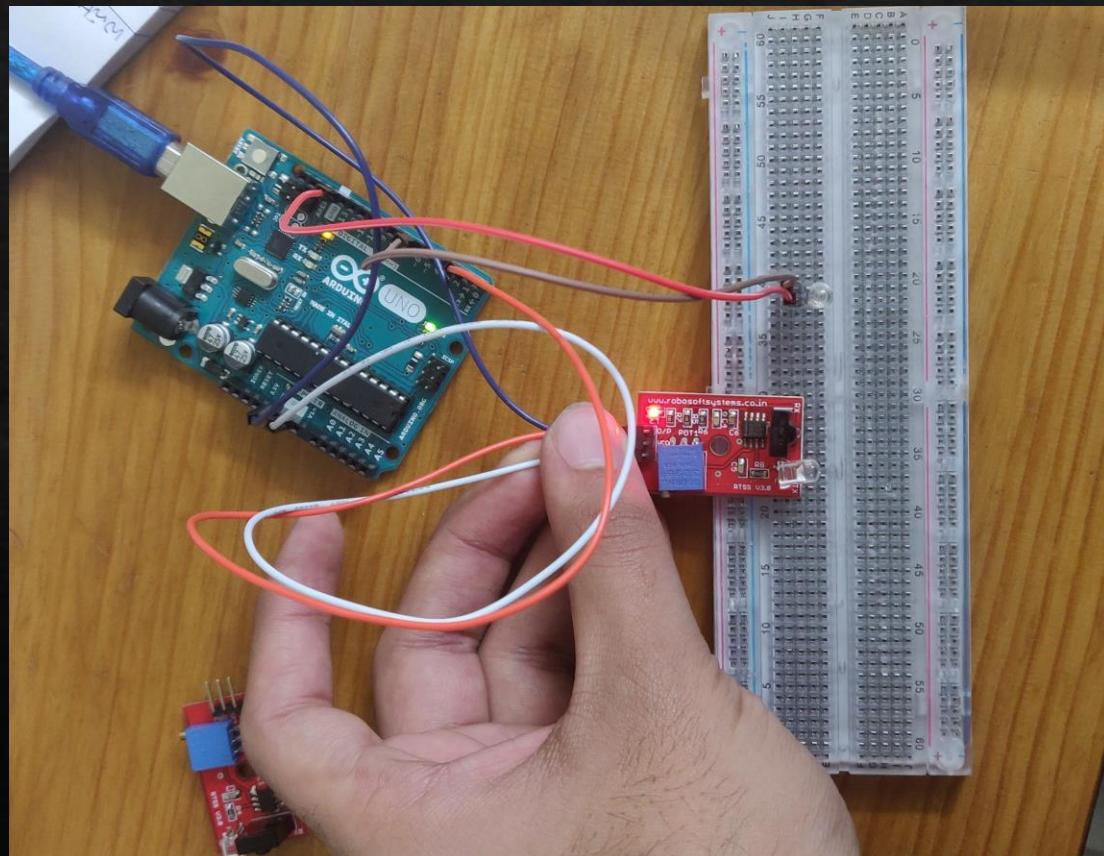
Testing IR Circuit

- The testing circuit is as shown in the figure. As one would notice we have three pins available for the connection namely the VCC, GND and Output pin.
- There can also be seen an LED attached to the circuit which is an indicator as to when the voltage goes high or low that is when the output is detected. The connection is made as follows:

Arduino Connection	IR Sensor Module Connection
5 V	VCC
GND	GND
2	OUT

Testing IR Circuit

- The connection were made as shown in the table. Also a simple LED was attached in order to avoid the confusion during the testing. That is when the obstacle was detected the external LED would go HIGH else would remain LOW.
- The distance initially was adjusted using a potentiometer to get the desired output and understand clear working of the sensor.



Testing IR Circuit: Sample Code and Output

sketch_oct18b | Arduino 1.8.13

File Edit Sketch Tools Help

sketch_oct18b

```
const int ProxSensor=2;
int inputVal = 0;

void setup()
{
    pinMode(7, OUTPUT);          // Pin 13 has an LED connected on most Arduino boards:
    pinMode(ProxSensor, INPUT);   //Pin 2 is connected to the output of proximity sensor
    Serial.begin(9600);
}

void loop()
{
    if(digitalRead(ProxSensor)==HIGH)      //Check the sensor output
    {
        digitalWrite(7, HIGH);           // set the LED on
    }
    else
    {
        digitalWrite(7, LOW);          // set the LED off
    }
}
```

Done uploading.

Sketch uses 2330 bytes (7%) of program storage space. Maximum is 32256 bytes.
Global variables use 188 bytes (9%) of dynamic memory, leaving 1860 bytes for local variables.

The 1 and 0 indicate the presence and absence of the object.

Testing of IR Circuit: Other Parameters

- Unlike the LDR circuit though the program showed us that the working was well. But since we are doing testing we decide to check for some other parameters as well so that we know if everything else would be fine.
- Thus, we decided to check its range, voltage, current etc, and at this point we made a check of the voltage that the LED was getting to check what voltage the LED can handle. The table describes as follows:

Testing Parameter	Result
Range of Detection	12 cm (Accuracy may decrease)
Input Voltage to the Module	4.98 V
LED Indicator	ON
Digital Output Voltage	0.10 V - 4.96 V
External LED Voltage Receiving	3.56 V
Expected Results Achieved	The expected outcome is achieved but as the distance increased with the potentiometer the accuracy decreased but since we are planning to place the sensor near to the letter box opening, thus the testing is acceptable.

Testing GSM Module

- The GSM Module which is like a miniature of mobile phones. It is capable to send/receive text messages along with calling and receiving phone calls.
- The GSM module we are using is based on the SIM900. We are using the one which is like a shield for Arduino. This shield has an audio jack wherein one can connect earphones and talk like he/she would do using a normal phone.
- For the project since we are using the UART for connection thus we shall check the UART pins. Thus, when we received the module, we didn't get the UART pins soldered hence initial soldering was done by us.

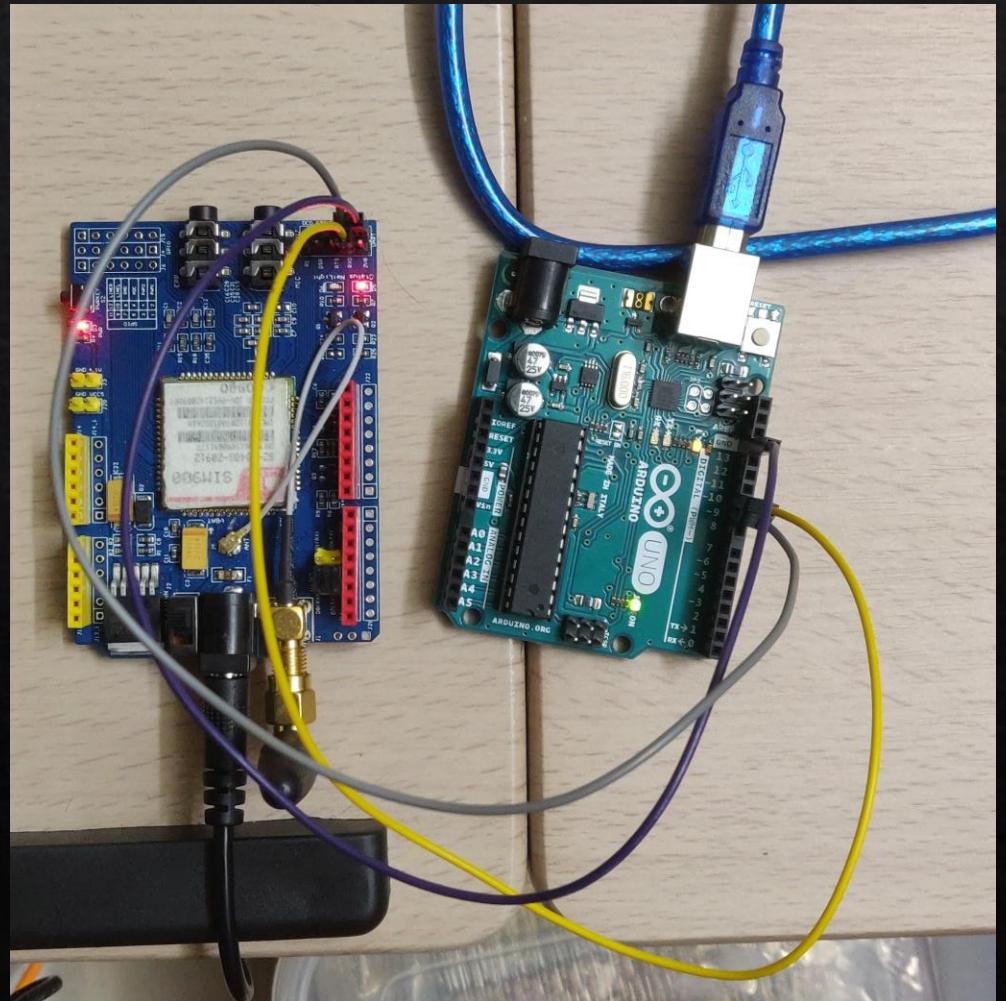
Testing the GSM Module

- The GSM module connection to the Arduino was quite a simple one. Also the only problem that we faced here was finding the appropriate library and using some internet sources that task was resolved.
- Since in the UART the transmitter and receiver pins are of importance thus they were tested. The GSM module was powered up using a 5 V wall adapter. The voltage of the adapter was checked before actually supplying. The voltage was around 5.17 V which is permissible as per the datasheet of the shield. The connection were made as follows:

GSM MODULE SIM900 UART SIDE	ARDUINO CONNECTION
Transmitter Pin (TX)	9
Receiver Pin (RX)	10
GND	GND

Testing GSM Module: Circuit

- The appropriate connections were made using the table shown on the previous slide (#17). The wall adapter was attached, and power was given to the GSM
- Then using the power key we made it connect to a suitable frequency range as it couldn't connect to the network initially.
- The circuit connection were as follows:



Testing GSM Module: Testing Code

The screenshot shows the Arduino IDE interface on a Linux desktop. The title bar indicates it's running on an Arduino Uno connected via /dev/ttyACM0 at 1.8.13. The code in the main editor is for a sketch named "sketch_oct18c". The code initializes a SoftwareSerial object for pins 9 and 10 at 9600 baud. It sets up the GSM module with AT+CMGF=1 and sends an SMS to a mobile number. It also receives messages from the mobile number. The code uses delay() functions for timing between commands and messages. The status bar at the bottom shows memory usage details.

```
Activities Arduino IDE
File Edit Sketch Tools Help
sketch_oct18c | Arduino 1.8.13
Sun 2:00 PM •
sketch_oct18c

#include <SoftwareSerial.h>
SoftwareSerial mySerial(9, 10);

void setup()
{
    mySerial.begin(9600); // Setting the baud rate of GSM Module
    Serial.begin(9600); // Setting the baud rate of Serial Monitor (Arduino)
    delay(100);
}

void loop()
{
    if (Serial.available()>0)
        switch(Serial.read())
    {
        case 's':
            SendMessage();
            break;
        case 'r':
            RecieveMessage();
            break;
    }

    if (mySerial.available()>0)
        Serial.write(mySerial.read());
}

void SendMessage()
{
    mySerial.println("AT+CMGF=1"); //Sets the GSM Module in Text Mode
    delay(1000); // Delay of 1000 milli seconds or 1 second
    mySerial.println("AT+CMGS=\\" +16477868334 "\\r"); // Replace x with mobile number
    delay(1000);
    mySerial.println("Welcome Zain! Looking for your letters?"); // The SMS text you want to send
    delay(100);
    mySerial.println((char)26); // ASCII code of CTRL+Z
    delay(1000);
}

void RecieveMessage()
{
    mySerial.println("AT+CNMI=2,2,0,0,0"); // AT Command to receive a live SMS
    delay(1000);
}

Done uploading.

Sketch uses 3648 bytes (11%) of program storage space. Maximum is 32256 bytes.
Global variables use 397 bytes (19%) of dynamic memory, leaving 1651 bytes for local variables. Maximum is 2048 bytes.
```

Testing GSM Module: Testing Code Output

```
/dev/ttyACM0
Send

AT+CMGF=1
OK
AT+CMGS="+16477868334"
> Welcome Zain! Loo
|+CMGS: 3
OK
```

```
/dev/ttyACM0
Send

AT+CMGF=1
OK
AT+CMGS="+16477868334"
> Welcome Zain! Loo
|+CMGS: 3
OK
AT+CNMI=2,2,0,0,0
OK
+CMT: "+16477868334","","","20/10/18,13:57:26-16"
Yes, can you please update me on the same?

Newline 9600 baud Clear output
Autoscroll Show timestamp
```



Testing GSM Module: Other Parameters

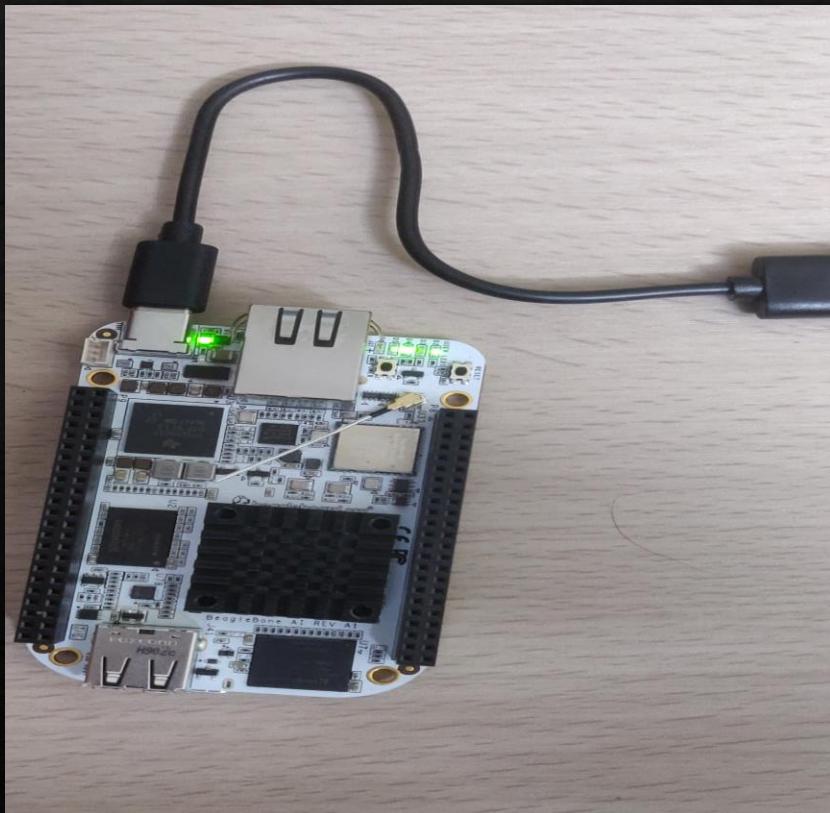
- Since the module accomplished the most important task successfully as required for the project thus, it has passed the testing stage.
- But we generate the following test table in order to make sure that module was working well.

Testing Parameter	Result
Input Voltage	5.17 V
GSM Module Power LED	ON
Status LED of GSM Module	ON
Network LED of GSM Module	ON initially then starts blink pattern
GSM Module capable to send Message	Yes
GSM Module able to receive message	Yes
Expected Output Achieved	Yes, performed all well even when the number was called the status LED went OFF indicating the SIM Card was busy.

Testing the Beaglebone Artificial Intelligence (BB-AI)

- Beaglebone Artificial Intelligence (BB-AI) which is the main processing unit in our project.
- BB-AI board requires a voltage of 5 V and 3 A current. This a high-speed device hence the processor may turn hot eventually when high computation or connection to wifi is done preferably.
- For the purpose of testing this device is connected to the host machine via a Type-C USB cable but for the purpose of the project it shall be connected using a wall adpater of the desired voltage.

Testing the BB-AI Connection and Login



```
zain@zain-xps-13-7390: ~
File Edit View Search Terminal Help
zain@zain-xps-13-7390:~$ ssh debian@192.168.7.2
Debian GNU/Linux 9

BeagleBoard.org Debian Image 2019-08-03

Support/FAQ: http://elinux.org/Beagleboard:BeagleBoneBlack_Debian
Default username:password is [debian:temppwd]

debian@192.168.7.2's password: □
```

BB-AI Testing: Set Up WiFi Connectivity

- Since the image present on the Beaglebone AI Seems to be quite new enough thus no need to upgrade it. Since the project requires a WiFi Connection thus we shall start to test the WiFi connection of the device using the connctl command as follows:

```
zain@zain-xps-13-7390: ~
File Edit View Search Terminal Help
default username:password is [debian:temppwd]
debian@192.168.7.2's password:
Permission denied, please try again.
debian@192.168.7.2's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Aug  4 02:41:40 2019 from 192.168.7.1
debian@beaglebone:~$ sudo connctl

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

[sudo] password for debian: 
```

```
zain@zain-xps-13-7390: ~
File Edit View Search Terminal Help
debian@beaglebone:~$ connctl
Error getting VPN connections: The name net.connman.vpn was not provided by any
connctl> exit
debian@beaglebone:~$ sudo connctl
Error getting VPN connections: The name net.connman.vpn was not provided by any
connctl> enable wifi
Error wifi: Already enabled
connctl> scan wifi
Scan completed for wifi
connctl> 
```

BB-AI Wifi Setup

```
zain@zain-xps-13-7390: ~
File Edit View Search Terminal Help
Error getting VPN connections: The name net.connman.vpn was not provided by any
connmanctl> exit
debian@beaglebone:~$ sudo connmanctl
Error getting VPN connections: The name net.connman.vpn was not provided by any
connmanctl> enable wifi
Error wifi: Already enabled
connmanctl> scan wifi
Scan completed for wifi
connmanctl> services
    $$VirusAttack$$      wifi_8091334a4b1f_2424566972757341747461636b2424_manage
d_psk
                               wifi_8091334a4b1f_hidden_managed_psk
                               wifi_8091334a4b1f_hidden_managed_ieee8021x
                               wifi_8091334a4b1f_6c6f70657a_managed_psk
                               wifi_8091334a4b1f_436f6f6c6b69647331_managed_psk
                               wifi_8091334a4b1f_49796164_managed_psk
lopez
BELL249
Coolkids1
Iyad
connmanctl> agent on
Agent registered
connmanctl> connect wifi_8091334a4b1f_2424566972757341747461636b2424_managed_psk
Agent RequestInput wifi_8091334a4b1f_2424566972757341747461636b2424_managed_psk
  Passphrase = [ Type=psk, Requirement=mandatory, Alternates=[ WPS ] ]
  WPS = [ Type=wpspin, Requirement=alternate ]
Passphrase? nopassword
connmanctl> 
```

```
zain@zain-xps-13-7390: ~
File Edit View Search Terminal Help
    $$VirusAttack$$      wifi_8091334a4b1f_2424566972757341747461636b2424_manage
d_psk
                               wifi_8091334a4b1f_hidden_managed_ieee8021x
                               wifi_8091334a4b1f_hidden_managed_psk
                               wifi_8091334a4b1f_6c6f70657a_managed_psk
                               wifi_8091334a4b1f_42454c4c323439_managed_psk
                               wifi_8091334a4b1f_436f6f6c6b69647331_managed_psk
                               wifi_8091334a4b1f_49796164_managed_psk
lopez
BELL249
Coolkids1
Iyad
connmanctl> services
*AR ONEPLUS_A6010_co_apcsix wifi_8091334a4b1f_4f4e45504c55535f41363031305f636f5f
617063736978_managed_psk
    $$VirusAttack$$      wifi_8091334a4b1f_2424566972757341747461636b2424_manage
d_psk
                               wifi_8091334a4b1f_hidden_managed_ieee8021x
                               wifi_8091334a4b1f_hidden_managed_psk
                               wifi_8091334a4b1f_6c6f70657a_managed_psk
                               wifi_8091334a4b1f_42454c4c323439_managed_psk
                               wifi_8091334a4b1f_436f6f6c6b69647331_managed_psk
                               wifi_8091334a4b1f_49796164_managed_psk
lopez
BELL249
Coolkids1
Iyad
connmanctl> connect wifi_8091334a4b1f_4f4e45504c55535f41363031305f636f5f61706373
6978_managed_psk
Error /net/connman/service/wifi_8091334a4b1f_4f4e45504c55535f41363031305f636f5f6
17063736978_managed_psk: Already connected
connmanctl> 
```

BB-AI Wifi Setup Confirm and Test Report

- The Wifi connection has been established just to be sure we can ping a particular website and check if we are connected to the Internet. This can be seen as in the image on right.
- Finally we make the report of testing as follows:

Testing Parameters	Result
BB-AI OS Image	Yes Updated to the nearest one released in November 2019
Power Supply	5.02 V
Wifi Connection	Established and working [Mobile Hotspot]
Device Ready	Yes, passed the testing stage can work for the project

```
zain@zain-xps-13-7390: ~
File Edit View Search Terminal Help
Iyad          wifi_8091334a4b1f_49796164_managed_psk
connmanctl> connect wifi_8091334a4b1f_4f4e45504c55535f41363031305f636f5f61706373
6978_managed_psk
Error /net/connman/service/wifi_8091334a4b1f_4f4e45504c55535f41363031305f636f5f6
17063736978_managed_psk: Already connected
connmanctl> quit
debian@beaglebone:~$ ping www.google.com
PING www.google.com(yyz12s05-in-x04.1e100.net (2607:f8b0:400b:801::2004)) 56 dat
a bytes
64 bytes from yyz12s05-in-x04.1e100.net (2607:f8b0:400b:801::2004): icmp_seq=1 t
tl=116 time=32.1 ms
64 bytes from yyz12s05-in-x04.1e100.net (2607:f8b0:400b:801::2004): icmp_seq=2 t
tl=116 time=51.4 ms
64 bytes from yyz12s05-in-x04.1e100.net (2607:f8b0:400b:801::2004): icmp_seq=3 t
tl=116 time=61.8 ms
64 bytes from yyz12s05-in-x04.1e100.net (2607:f8b0:400b:801::2004): icmp_seq=4 t
tl=116 time=59.9 ms
64 bytes from yyz12s05-in-x04.1e100.net (2607:f8b0:400b:801::2004): icmp_seq=5 t
tl=116 time=39.0 ms
64 bytes from yyz12s05-in-x04.1e100.net (2607:f8b0:400b:801::2004): icmp_seq=6 t
tl=116 time=37.5 ms
64 bytes from yyz12s05-in-x04.1e100.net (2607:f8b0:400b:801::2004): icmp_seq=7 t
tl=116 time=55.5 ms
```

Conclusion

- Looking at the testing result of each and every sensor that we received and require for the project one can say that they seem to be working fine.
- All the parameters like the voltage and current are in the acceptable range and thus we assume that no harm may be caused to the circuits we shall build in the future
- The sensors and the main processing unit (BB-AI) are all ready to be used for the project

References

- Sarwar, I. (2016, October 01). Advantages and Disadvantages of Using Arduino. Retrieved October 19, 2020, from <https://engineerexperiences.com/advantages-and-disadvantages.html>
- How to Configure the Wifi on a BeagleBone Black Wireless. (n.d.). Retrieved October 19, 2020, from <https://www.fis.gatech.edu/how-to-configure-bbw-wifi/>
- Interface GSM Module to Arduino - Send and Receive SMS. (2018, August 11). Retrieved October 19, 2020, from <https://www.circuitstoday.com/interface-gsm-module-with-arduino>