**Project Report**

**Project 2 – BLE + Controlled Lighting**

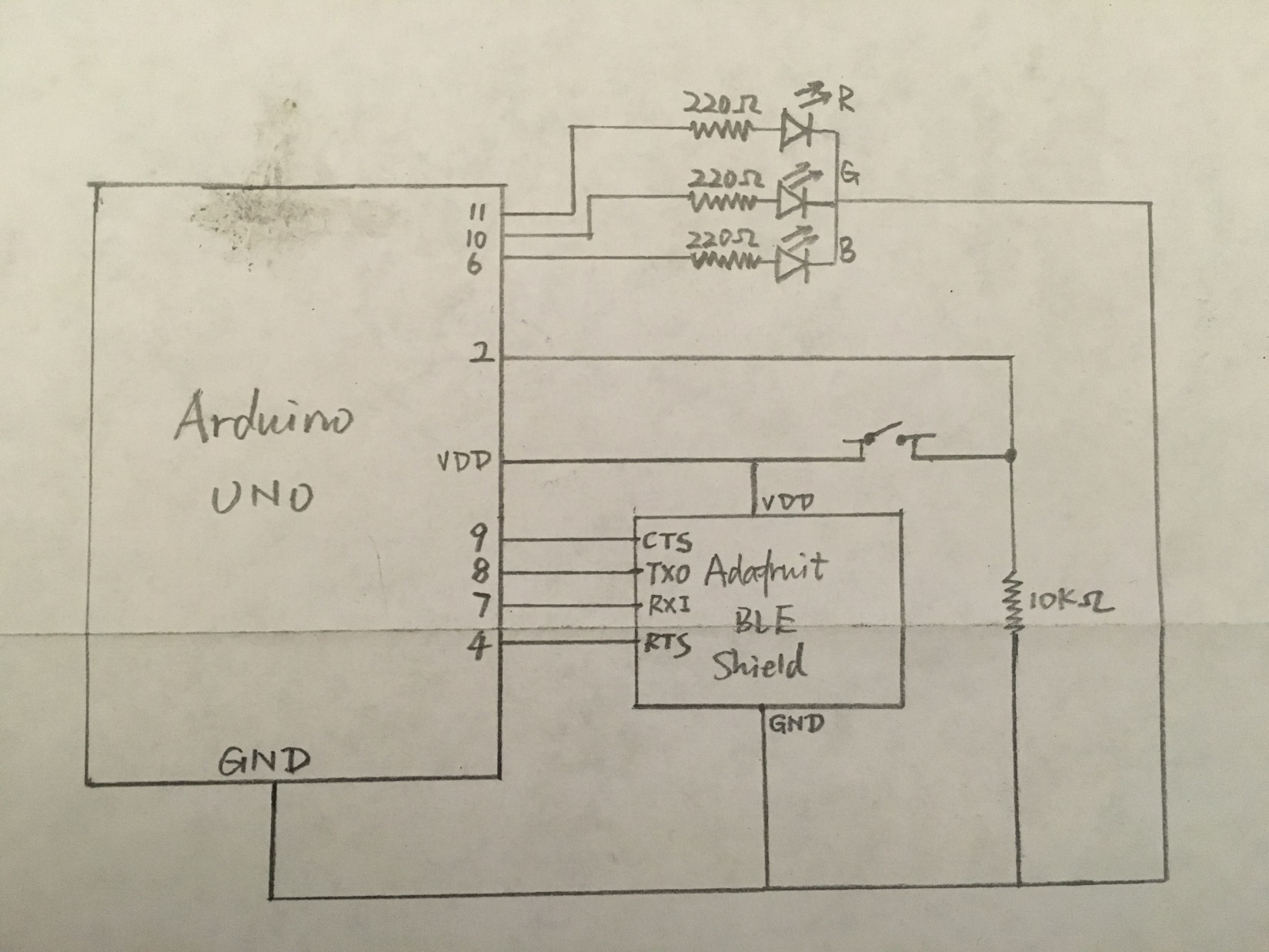
**Zhong Zheng**

**Design Objective**

This project is a RGB LED lighting system controlled by mobile phone BLE and a pushbutton on board.

**Hardware Design**

Here is the circuit diagram below



The components include:

1 Arduino UNO board

1 RGB LED

1 Pushbutton switch

1 Adafruit BLE shield

1 10k ohm resistors and 3 220 ohm resistors

PIN usage:

PWM Pin 11,10,6 are used as output for RGB anode.

Pin 2 are used as pushbutton control.

Pin 9,8,7,4 are used as Adafruit BLE shield connection.

**Software Design**

**Overall structure**

Use Adafruit BLE lib package to code this system, mainly including five parts.

1. Setup environment in setup(), to setup all the initial value includes BLE shield mode, device mode etc.

2. GATT services include read, notify, write.

3. Phone proximity RSSI value, signal level detector and controller.

4. nRF toolbox UART button control and light system.

5. Interrupt push button light controller.

**Function list**

setup(void)

void loop(void)

void button\_Switch\_Pushed()

void lightUpRGB(int i,int s,int l,int b)

**Goal Achievement**

1. Connect a RGB LED as the lighting.

An RGB LED is connected to Arduino UNO to be controlled.

2. Connect a push button to control LED on/off .

This push button is coded in interrupt ISR using PIN 2 and Falling edge sensitive.

3. Use nRF toolbox app (UART) to control LED.

Configure four button for control. Write four different letters to achieve four function.

h: light on/off

c: change color (totally 6 level for brightness, color changing is smoothly and gradually)

u: brightness up (totally 10 level for brightness)

d: brightness down

Use command AT+BLEUARTRX to get those letter for conditional statement.

4. Automatically turn on LED when phone is in proximity. Turn off LED when phone is leaving proximity

Use command AT+BLEGETRSSI to get the RSSI value for conditional statement. There are two cut-off point for the distance. And there is also a simple filter to average the signal readings. ( currentRSSI = currentRSSI \* 0.8 + RSSI \* 0.2;).

There are 3 state for proximity.

Working: When RSSI >= -60, light is on. No warning message.

Low Signal: When RSSI <= -70, light is off. Warning message.

Unstable Signal: When -60 > RSSI > -70, light is still on but serial monitor gets a message about unstable signal warning.

5. Additional BLE requirements, setting your own device name.

Use ble.sendCommandCheckOK(F( "AT+GAPDEVNAME=Zhong RGB lighting" )) to set this device name as Zhong RGB lighting

6. Create a custom GATT service (Bonus)

Use provided website to create my own UUID 0x1220

*AT+GATTADDSERVICE=UUID128=1C-39-12-20-DC-1E-11-E5-84-FC-00-02-A5-D5-C5-1B*

Add two characteristics

LED status: read/notify properties; Value 0: off; Value 1: on

*AT+GATTADDCHAR=UUID=0x0002, PROPERTIES=0x12, MIN\_LEN=1, VALUE=1*

LED control: write property; Value 0: turn off; Value 1 : turn of

*AT+GATTADDCHAR=UUID=0x0003, PROPERTIES=0x08, MIN\_LEN=1, MAX\_LEN=3, VALUE=0*

Use Master Control Pane mobile app and turn on UUID 0x0002 notification to see the ledState value.

Use UUID 0x0003 to write 0x01 / 0x00 to control LED on/off

**Special Considerations**

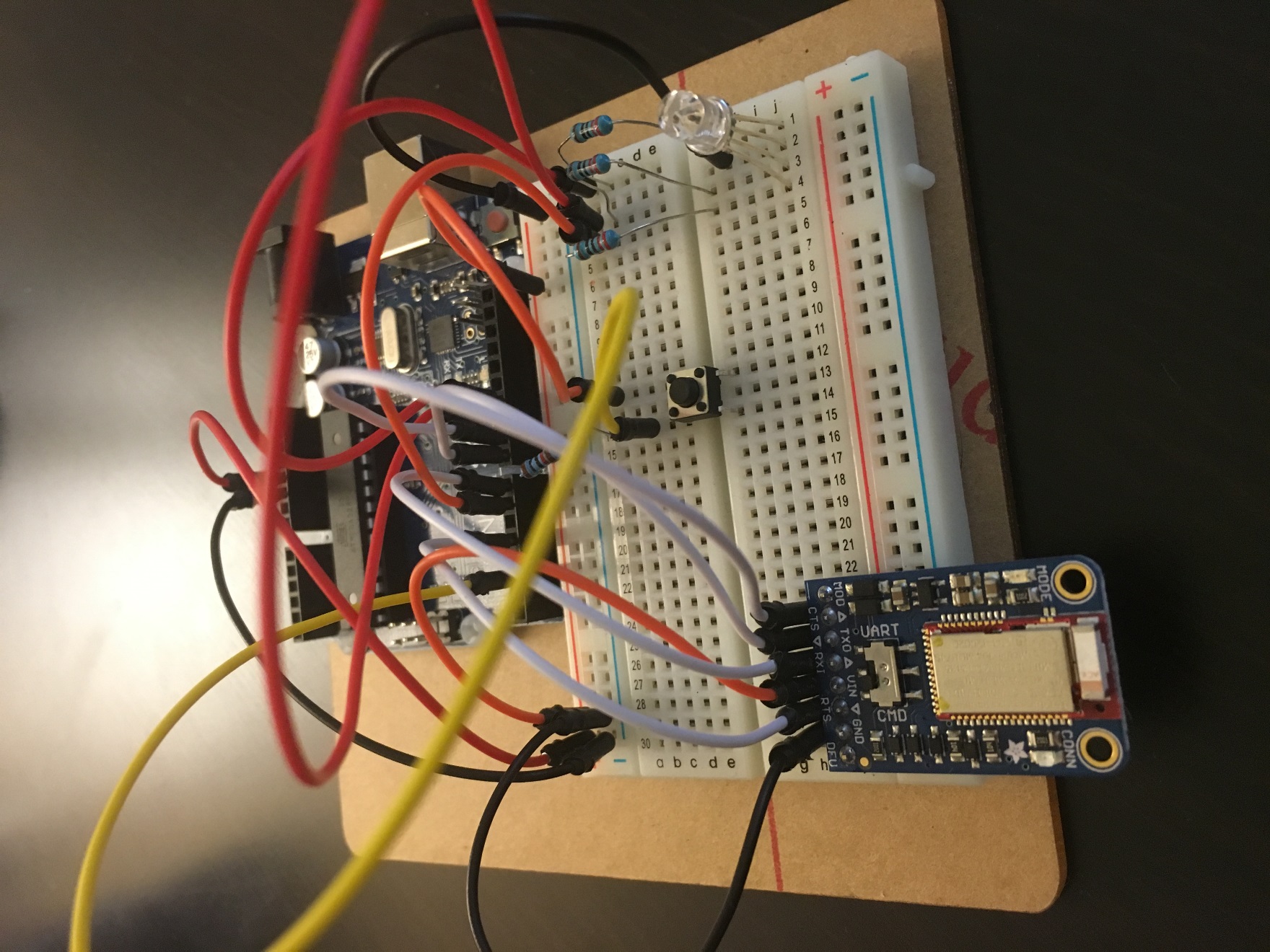
This project combines Arduino board and Adafruit BLE shield to control RGB LED light. The BLE shield is set to command mode. There are totally four sources to turn on/off the light (pushbutton, proximity, nRF Toolbox UART button, nRF MCP GATT service). They all drive one ledState to control the light, concurrently and no priority. It is the same with brightness variable.

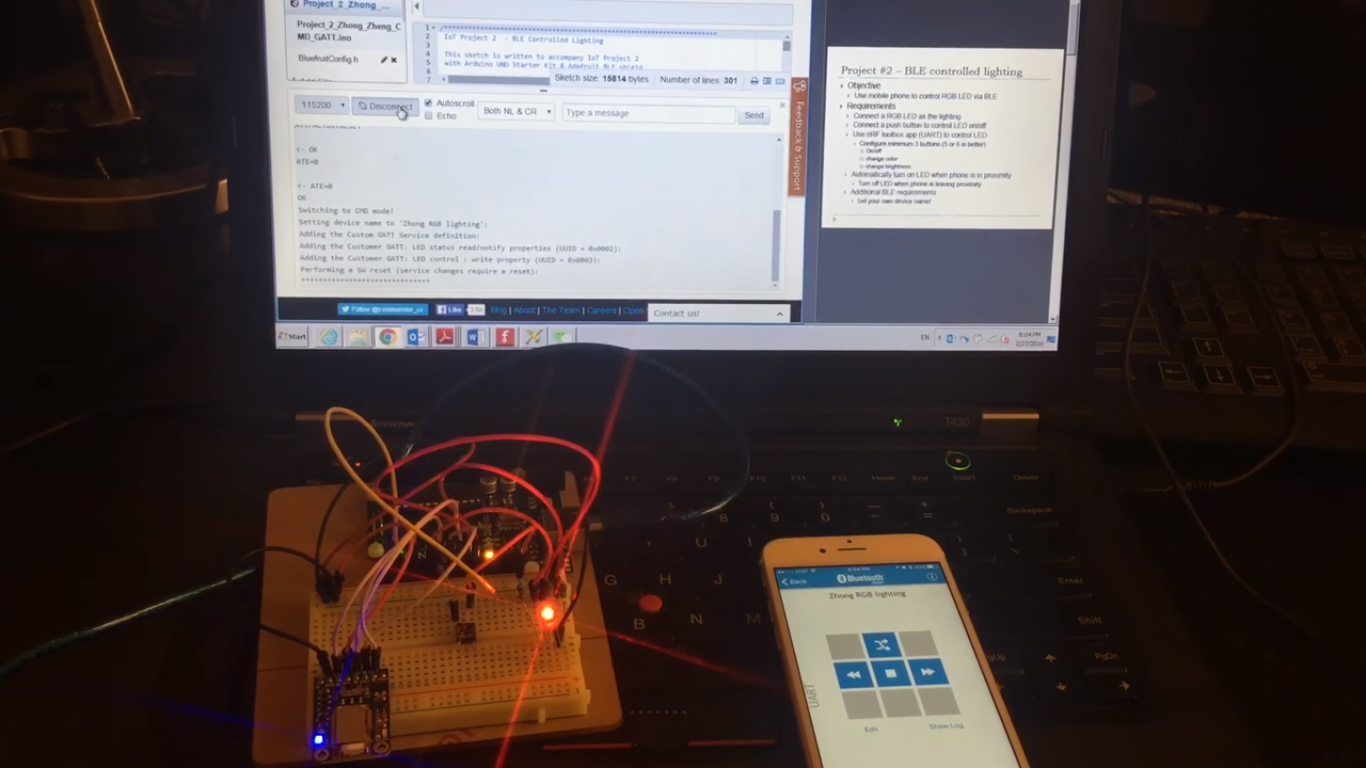
I am using the legacy code from project 1 to make the light color changing smoothly and gradually. There are totally 10 level of brightness during brightness control. We can see the GATT LIST in serial monitor using ble.println("AT+GATTLIST"), when ble.verbose(false) is commented.

**Appendix**

Sketch URL: <https://codebender.cc/sketch:251511>

Picture of hardware setup:





A complete session which demonstrate all the features

Link to video clip on YouTube: <http://youtu.be/a2b2RZ2xcyY>

Corresponding serial console output of one complete run

initial light ON

Zhong IoT Project 2 - BLE Controlled Lighting

-------------------------------------

Initializing the Bluefruit LE module: ATZ

<- OK

OK!

Performing a factory reset:

AT+FACTORYRESET

<- OK

ATE=0

<- ATE=0

OK

Switching to CMD mode!

Setting device name to 'Zhong RGB lighting':

Adding the Custom GATT Service definition:

Adding the Customer GATT: LED status read/notify properties (UUID = 0x0002):

Adding the Customer GATT: LED control : write property (UUID = 0x0003):

Performing a SW reset (service changes require a reset):

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

RGB on/off changed from

1 to 0

RGB on/off changed from

0 to 1

RGB on/off changed from

1 to 0

RGB on/off changed from

0 to 1

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -60.39

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -62.31

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -64.65

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -67.72

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -69.77

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -71.82

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -72.86

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -75.88

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -75.51

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -75.81

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -75.44

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -75.36

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -78.08

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -76.87

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -73.89

Low Signal( RSSI < -70 ). Please move your phone closer (RSSI>-60), current RSSI : -70.72

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -68.77

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -69.02

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -69.81

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -66.25

Signal unstable( -60 > RSSI > -70 ). Please move your phone closer (RSSI>-60), current RSSI : -61.40

[Button Recv] c

RGB change color,current color : 1

[Button Recv] c

RGB change color,current color : 2

[Button Recv] c

RGB change color,current color : 3

[Button Recv] c

RGB change color,current color : 4

[Button Recv] c

RGB change color,current color : 5

[Button Recv] c

RGB change color,current color : 0

[Button Recv] c

[Button Recv] d

RGB brightness down, current brightness : 9

[Button Recv] d

RGB brightness down, current brightness : 8

[Button Recv] d

RGB brightness down, current brightness : 7

[Button Recv] d

RGB brightness down, current brightness : 6

[Button Recv] d

RGB brightness down, current brightness : 5

[Button Recv] d

RGB brightness down, current brightness : 4

[Button Recv] d

RGB brightness down, current brightness : 3

[Button Recv] d

RGB brightness down, current brightness : 2

[Button Recv] d

RGB brightness down, current brightness : 1

[Button Recv] d

RGB brightness down, current brightness : 0

[Button Recv] u

RGB brightness up, current brightness : 1

[Button Recv] u

RGB brightness up, current brightness : 2

[Button Recv] u

RGB brightness up, current brightness : 3

[Button Recv] u

RGB brightness up, current brightness : 4

[Button Recv] u

RGB brightness up, current brightness : 5

[Button Recv] u

RGB brightness up, current brightness : 6

[Button Recv] u

RGB brightness up, current brightness : 7

[Button Recv] u

RGB brightness up, current brightness : 8

[Button Recv] u

RGB brightness up, current brightness : 9

[Button Recv] u

RGB brightness up, current brightness : 10

[Button Recv] h

RGB on/off state : 0

[Button Recv] h

RGB on/off state : 1

[Button Recv] h

RGB on/off state : 0

[GATT Recv] 1

[GATT Recv] 0

[GATT Recv] 1

[GATT Recv] 0

[GATT Recv] 1

Please email me if the sketch URL or YouTube link has any issue.

Thank you.

Zhong Zheng

[cathead@live.cn](mailto:cathead@live.cn)