Arduino uno-lite+LC-2000(PA), The wireless light control

silver

2016/2/24

Everything done in house

1.summary

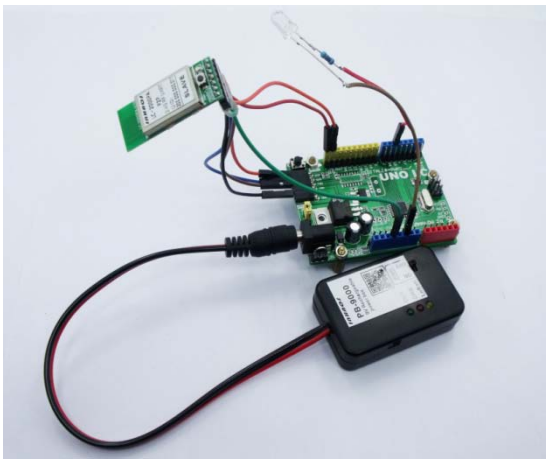
This document gives one example of how to use the ARDUINO-UNO-LITE, LC-2000(PA)-P2P to make a wireless light control.

2.material

- ARDUINO-UNO-LITE ;
- LC-2000(PA)-P2P-SLAVE;
  LC-2000(PA)-P2P-MASTER;
- CB-LC-2000;
- UC-2102;
- LED;
- LC-2000-DEBUGGER;
- THE SERIAL DEBUGGING ASSISTANT;

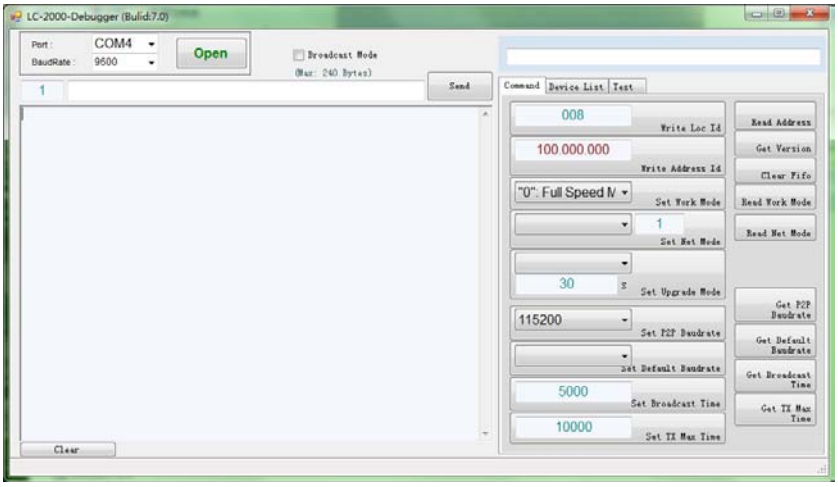
3.PCB board design

We need connected the LC-2000(PA)-SLAVE module and the ARDUINO UNO-LITE shield, The LC-2000(PA)-P2P shield and the CB-LC-2000 shield, and UC-2102 and CB-LC-2000. the Line connection order is vcc-vcc3.3v, RXD-RXD, TXD-TXD, DTR-DTR, GND-GND.

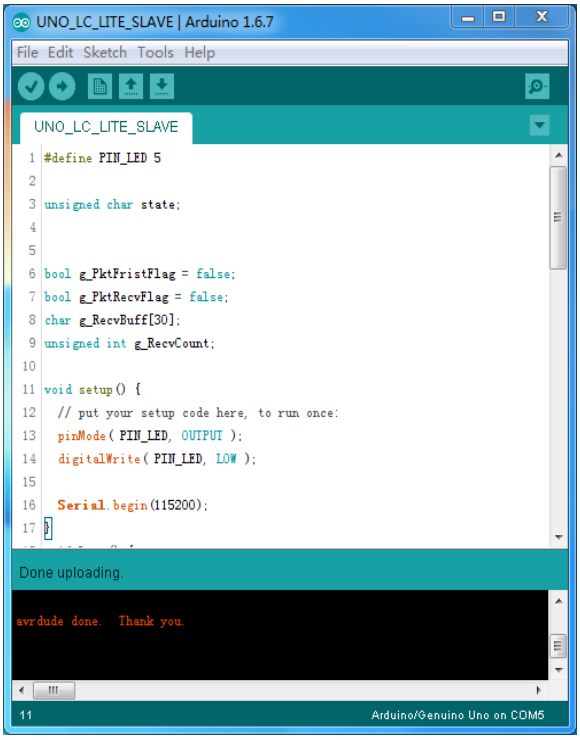


4. Detailed steps

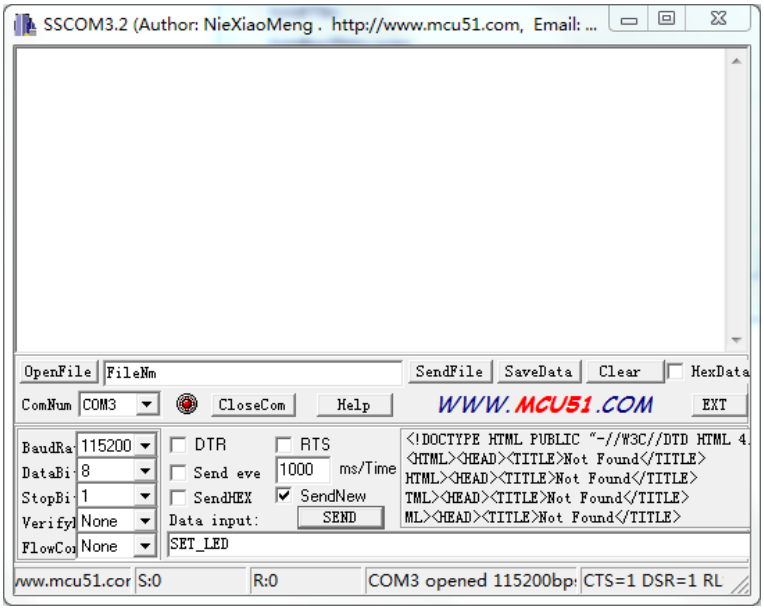
1. Connect the LC-2000(PA)-P2P-SLAVE module and the CB-LC-2000 shield, then open the LC-2000-DEBUGGER, Set the address of the module, Work mode is set to full speed, Baud rate is set to 115200 bps. Remove the module and connects it to the ARDUINO UNO-LITE;
2. Connect the LC-2000(PA)-P2P-MASTER module and the CB-LC-2000 shield, then open the LC-2000-DEBUGGER, Set the address of the module、Working mode、and Baud rate just same like the slave module;



3. Pairing the two RF module by Pushing the CFG long until pairing LED flashing;
4. Open the ARDUINO IDE, choose the port of UC-2102, copy the program to IDE, and download the program, The program can be wirelessly uploaded to the ARDUINO UNO-LITE；



5. Open the serial debugging assistant, choose the port of UC-2102, set the Baud rate to 115200 bps, choose “send new”;
6. Send data ”SET_LED”;



7. Now you can control the LED light;

5.The program

```
#define PIN_LED 5

unsigned char state;
```

```
bool g_PktFristFlag = false;
bool g_PktRecvFlag = false;
char g_RecvBuff[30];
unsigned int g_RecvCount;

void setup() {
  // put your setup code here, to run once:
  pinMode( PIN_LED, OUTPUT );
  digitalWrite( PIN_LED, LOW );

  Serial.begin(115200);
}
void loop() {
  // put your main code here, to run repeatedly:
  if ( g_PktRecvFlag ) {
    g_PktRecvFlag = false;
    g_PktFristFlag = false;
    ReadCommand();
    g_RecvCount = 0;
  }
}

void serialEvent() {
  while ( Serial.available() ) {
    char inChar = Serial.read();
    if ( inChar == '\r' ) {
      g_PktFristFlag = true;
    }
    else if ( inChar == '\n' ) {
      if ( g_PktFristFlag ) {
        g_RecvBuff[g_RecvCount] = '\0';
        g_PktRecvFlag = true;
      }
    }
    else {
      g_RecvBuff[g_RecvCount++] = inChar;
    }
  }
}

void ReadCommand() {
  char *pString = (char *)&g_RecvBuff[0];
  if ( 0 == strncmp( pString, "SET_LED", 7 ) ) {
    state = ~state;
    if ( state ) {
      digitalWrite( PIN_LED, HIGH);
    }
    else {
      digitalWrite( PIN_LED, LOW);
    }
  }

}
```