

الحمادي للإلكترونيات
ALHAMMADI FOR ELECTRONICS

AA011

LCD1602 Blue Backlight

V1.0.23.10.20

Preface

OurCompany

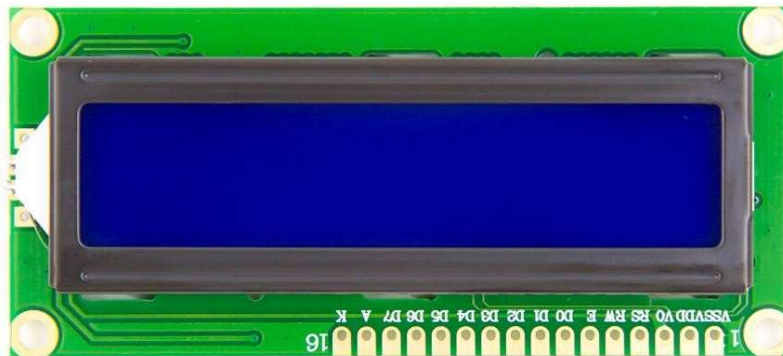
KUONGSHUN Electronic Company is a supplier and manufacturer of electronic components, it is committed to board and starter kit for Arduino, Raspberry PI, Smart Robot Car, 3D printer. It is also a collection of scientific research, design, production,

maintenance and sales of high-tech enterprises, in the field of automation with professional standards and mature technology, we rapid rise in the field of foreign trade. Relying on technology and development, continuing to provide users with high-tech products, is our constant pursuit. Fully introduction of foreign advanced technology to enhance the value of our products.

Company gains users' praise for supplying first-class quality product and superb technical services, has now become the first choice of domestic and international procurement company.

Official Website: <https://www.kuongshun-ks.com>

AA011 LCD1602 Blue Backlight



Product Description

This is a dot matrix LCD module that can be specialized for displaying letters, numeric characters, symbols, etc. It can display 2 lines, 16 characters per line, and the LCD screen displays white characters on a blue background. When used, it is divided into 4-bit and 8-bit two kinds of data transmission mode. The module provides internal power-on automatic reset circuit, when the external power supply voltage exceeds +4.5V, it will automatically initialize the module and set the module to the default display working state.

Product Parameters

Logic operating voltage (Vdd): DC +4.8 ~ +5.2V

LCD Driving Voltage (Vdd-Vo): DC +3.0 ~ +5.0V

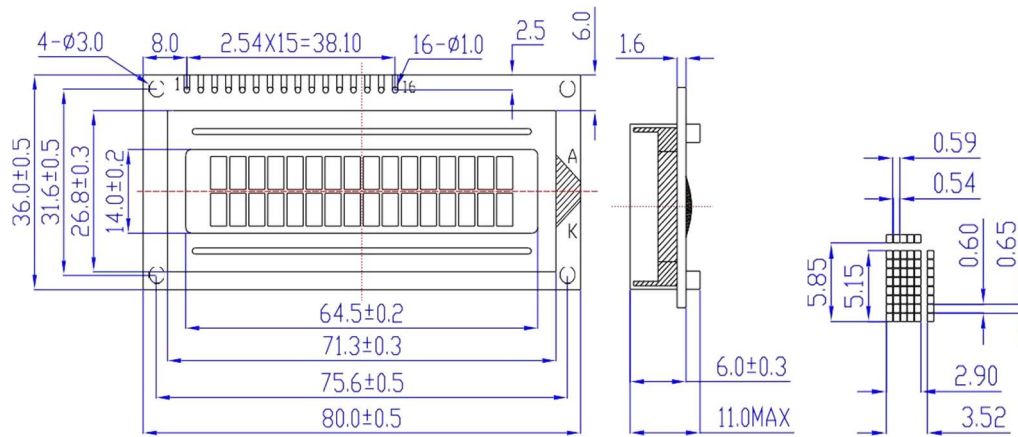
Operating Temperature: -20 ~ +70° C (Wide Temperature)

Storage Temperature: -30 ~ + 80° C (Wide Temperature)

Operating Current (except Backlight):1.7mA(max)

Operating current (except backlight):24.0mA(max)

Display color: white letters on blue background

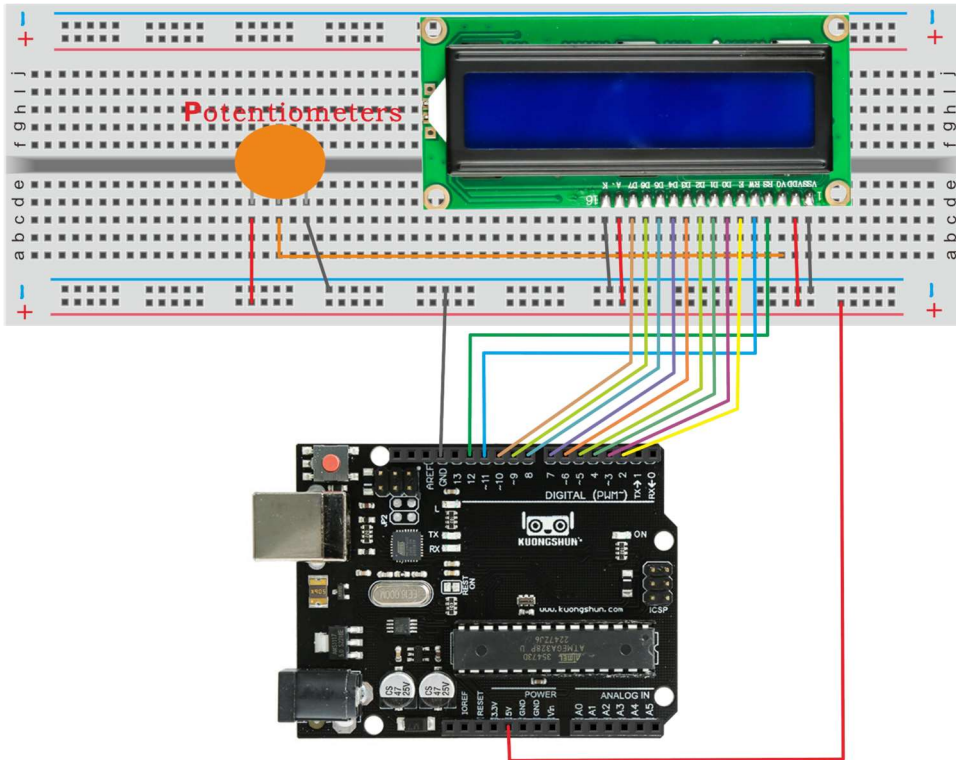


Pin Description

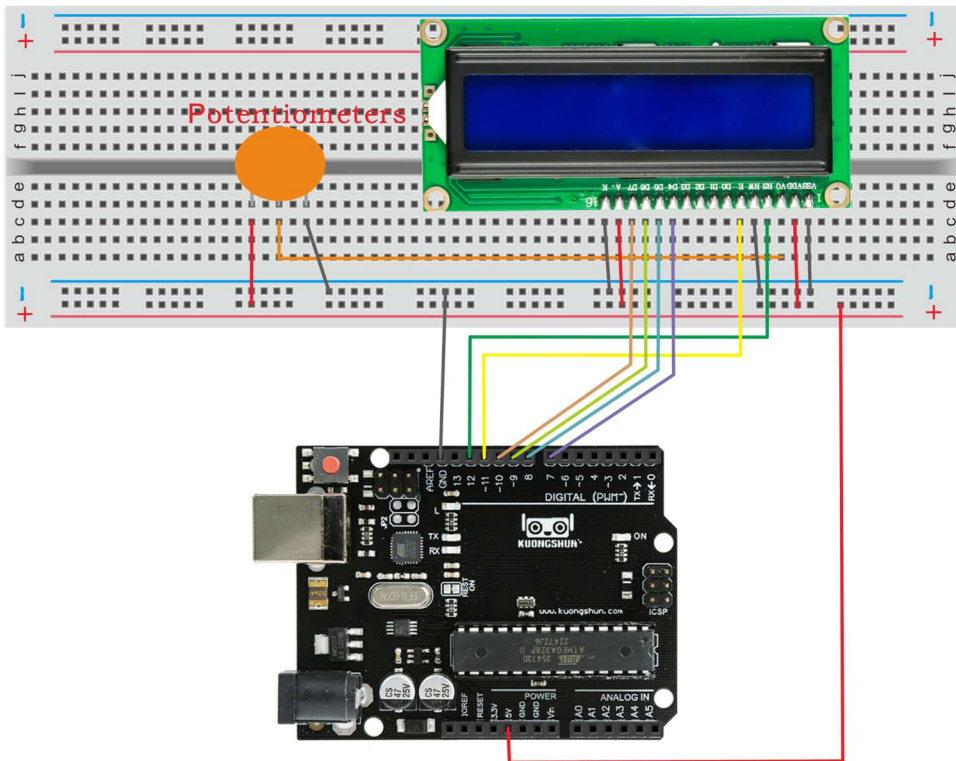
Pin NO.	Symbol	I/O	Description
1	GND	--	Power GND(0V)
2	VDD	--	Power positive(5V)
3	V0	--	LCD voltage bias signal
4	RS	I	Select data/command(V/L)
5	RW	I	Select read/write(H/L)
6	E	I	Enable signal
7	DB0	I/O	Date 0
8	DB1	I/O	Date 1
9	DB2	I/O	Date 2
10	DB3	I/O	Date 3
11	DB4	I/O	Date 4
12	DB5	I/O	Date 5
13	DB6	I/O	Date 6
14	DB7	I/O	Date 7
15	BL1	--	Back light power positive (5V)
16	BL2	--	Back light power negative(0V)

Connection Diagram

Eight-bit Connection Method



Four-bit Connection Method



Sample Code

Eight-bit Connection Code

/*

<https://www.kuongshun-ks.com/> */

```

int DI = 12; int RW = 11; int DB[] = {3, 4, 5, 6, 7, 8, 9, 10}; // use
array to select pin for bus int Enable = 2;

void LcdCommandWrite(int value) {
// define all pins int i = 0; for (i=DB[0]; i <= DI;
i++) // assign value for bus
{
    digitalWrite(i,value & 01); // for 1602 LCD, it uses D7-D0( not D0-D7) for signal identification; here, it'
s used for signal inversion.
    value >>= 1;
}
digitalWrite(Enable,LOW);
delayMicroseconds(1);
digitalWrite(Enable,HIGH);
delayMicroseconds(1); // wait for 1ms
digitalWrite(Enable,LOW);
delayMicroseconds(1); // wait for 1ms
}

void LcdDataWrite(int value) {
// initialize all pins int i = 0;
digitalWrite(DI, HIGH);
digitalWrite(RW, LOW); for
(i=DB[0]; i <= DB[7]; i++)
{ digitalWrite(i,value & 01);
value >>= 1;
}
digitalWrite(Enable,LOW);
delayMicroseconds(1);
digitalWrite(Enable,HIGH);
delayMicroseconds(1);
digitalWrite(Enable,LOW);
delayMicroseconds(1); // wait for 1ms
}

void setup (void) { int i = 0;
for (i=Enable; i <= DI; i++)
{ pinMode(i,OUTPUT);
}
delay(100);
// initialize LCD after a brief pause
// for LCD control

```

```

LcdCommandWrite(0x38); // select as 8-bit interface, 2-line display, 5x7 character size
delay(64);

LcdCommandWrite(0x38); // select as 8-bit interface, 2-line display, 5x7 character size
delay(50);

LcdCommandWrite(0x38); // select as 8-bit interface, 2-line display, 5x7 character size
delay(20);

LcdCommandWrite(0x06); // set input mode
                        // auto-increment, no display of shifting

delay(20);

LcdCommandWrite(0x0E); // display setup
                        // turn on the monitor, cursor on, no flickering

delay(20);

LcdCommandWrite(0x01); // clear the scree, cursor position returns to 0
delay(100);

LcdCommandWrite(0x80); // display setup
                        // turn on the monitor, cursor on, no flickering

delay(20);
}

void loop (void) {
    LcdCommandWrite(0x01); // clear the scree, cursor position returns
    to 0 delay(10);
    LcdCommandWrite(0x80);
    delay(10);
    // write in welcome message
    LcdDataWrite('H');
    LcdDataWrite('e');
    LcdDataWrite('l');
    LcdDataWrite('l');
    LcdDataWrite('o');
    LcdDataWrite(',');
    LcdDataWrite(' ');
    LcdDataWrite('w');
    LcdDataWrite('o');
    LcdDataWrite('r');
    LcdDataWrite('l');
    LcdDataWrite('d');
    LcdDataWrite('!');
    delay(10);
    LcdCommandWrite(0xc0);
        // set cursor position
    at second line, second

```

```

position delay(10);
LcdDataWrite('H');
LcdDataWrite('e');
LcdDataWrite('l');
LcdDataWrite('l');
LcdDataWrite('o');
LcdDataWrite(', ');
LcdDataWrite('k');
LcdDataWrite('u');
LcdDataWrite('o');
LcdDataWrite('n');
LcdDataWrite('g');
LcdDataWrite('s');
LcdDataWrite('h');
LcdDataWrite('u');
LcdDataWrite('n');
LcdDataWrite('!');
delay(5000);
LcdCommandWrite(0x01); // clear the screen, cursor
returns to 0 delay(10); LcdDataWrite('l'); LcdDataWrite(' ');
LcdDataWrite('a');
LcdDataWrite('m');
LcdDataWrite(' ');
LcdDataWrite('h');
LcdDataWrite('u');
LcdDataWrite('n');
LcdDataWrite('t');
LcdDataWrite('e');
LcdDataWrite('r');
LcdDataWrite('!');
delay(3000);
LcdCommandWrite(0x02); // set mode as new characters replay old ones, where there is no new ones
remain the same delay(10);
LcdCommandWrite(0x80+5); // set cursor position at first line, sixth position
delay(10); LcdDataWrite('t');
LcdDataWrite('h');
LcdDataWrite('e');
LcdDataWrite(' ');
LcdDataWrite('w');
LcdDataWrite('o');
LcdDataWrite('r');
LcdDataWrite('l');

```

```
LcdDataWrite('d');  
LcdDataWrite('!');  
delay(5000);  
}
```

Four-bit Connection Code

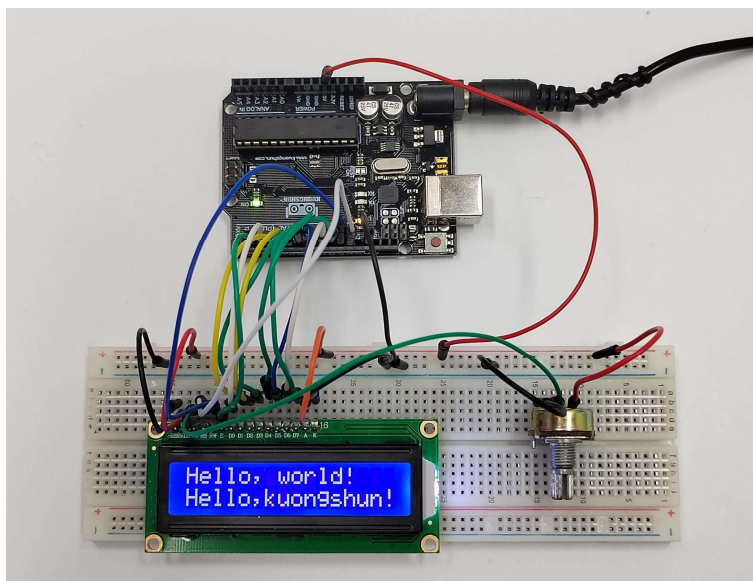
```
/*  
https://www.kuongshun-ks.com/  
*/  
#include <LiquidCrystal.h>  
// initialize the library with the numbers of the interface pins  
LiquidCrystal lcd(12, 11, 7, 8, 9, 10); void setup() {  
  // set up the LCD's number of columns and rows  

```

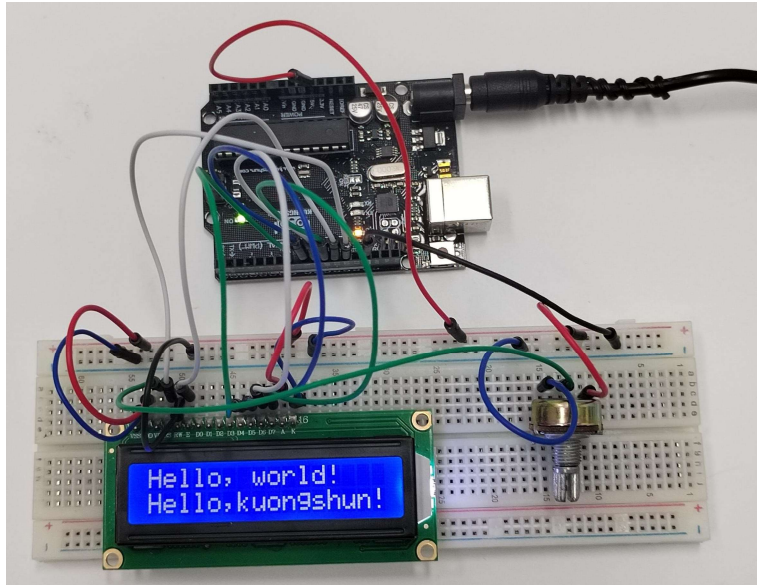
Test Result

Regardless of the four-bit connection or eight-bit connection, after the control board uploads the program successfully, the first line of the display shows "Hello, world!" characters, and the second line shows "Hello,kuongshun!" characters.

The eight-bit connection method is shown in the following figure:



The four-bit connection method is shown in the following figure:



Note: If you can't read the words clearly, you can turn the potentiometer on the breadboard to adjust the contrast. Resources

Related code and library files are linked below: