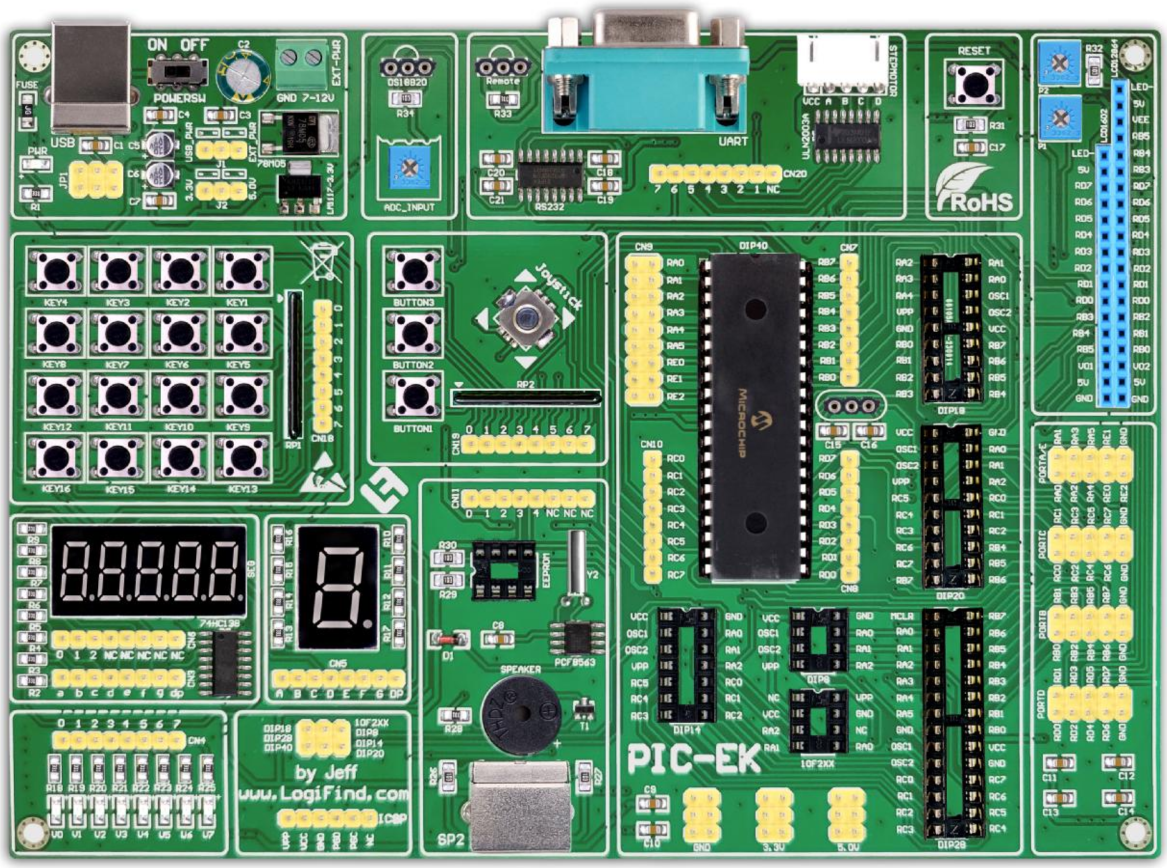


To our valued customers

I want to express my thanks to you for being interested in our products and for having confidence in www.MicroHello.com . The primary aim of our company is to design and produce high quality electronic products and to constantly improve the performance thereof in order to better suit your needs.

Please share your thoughts and feelings regarding our operation so that we can serve you better in the future. I thank you for your continued support and patronage.Your Dream is our Destination!

The **Microchip**, **TI**, **Freescale**, **NXP**, **ST**, **Atmel**, **Silicon** and **CYPRESS** name, logo and products names are trademarks of **Microchip**, **TI**, **Freescale**, **NXP**, **ST**, **Atmel**, **Silicon** and **CYPRESS** Inc. in the **U.S.A.** and other countries.

Sincerely,

Owner and General Manager
of www.MicroHello.com

This document is intended to explain how to perform hardware connections when the PIC-EK development board is equipped with PIC18F4550 during the testing of various modules, and briefly explain the experimental principle and results.

CONTENTS

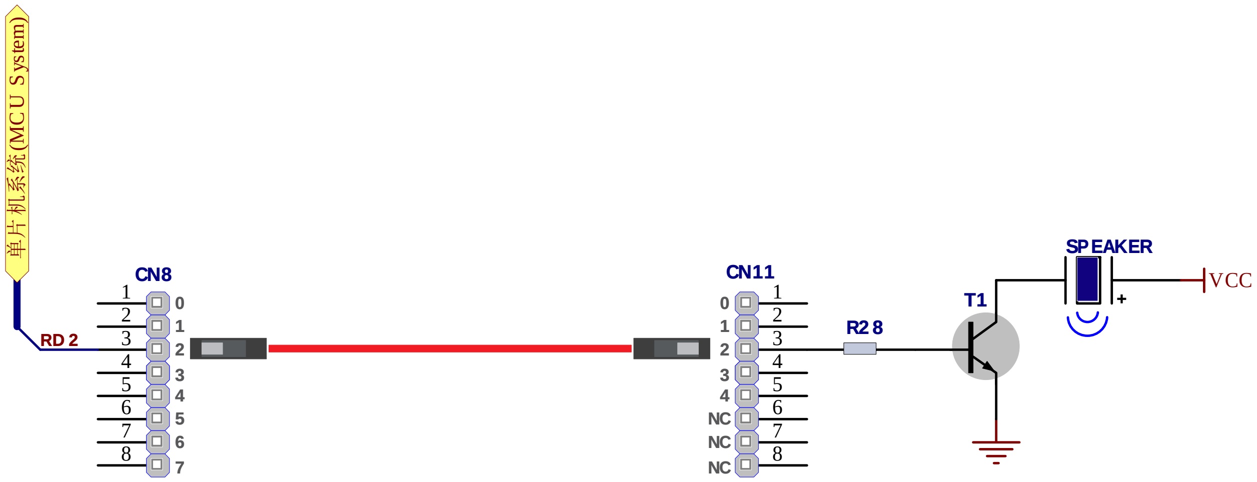
Hardware Connections..........5
Lab1.Beep_1..........5
Lab2. Beep_2..........6
Lab3.LED_V0_ON..........7
Lab4.LED_V0_Blinking..........8
Lab5.LEDs_Run..........9
Lab6.1SegLED_1..........10
Lab7.1SegLED_add..........11
Lab8.4SegLED_0..........12
Lab9.4SegLED_1..........13
Lab10.4SegLED_2..........14
Lab11.4SegLED_3..........15
Lab12.4SegLED_0123..........16
Lab13.4SegLED_Add..........19
Lab14.Jostick_Buttons_1SegLED..........20
Lab15.Jostick_Buttons_Beep..........21
Lab16. 4X4Keyboard_1SegLED..........22
Lab17. 4X4Keyboard_Beep..........24
Lab18. Stepmotor..........25
Lab19. LCD1602_Char..........26
Lab20. LCD1602_Num..........27
Lab21. LCD1602_Add..........28
Lab22. Timer0_V0..........29
Lab23. Timer0_1SegLED..........30
Lab24. Timer0_LCD1602..........31
Lab25. Timer1_V0..........32
Lab26. Timer1_1SegLED..........33
Lab27. Timer1_LCD1602..........34
Lab28. Timer2_V0..........35
Lab29. Timer2_1SegLED..........36
Lab30. Timer2_LCD1602..........37
Lab31. Interrupt(RB0)_Button1_LEDs..........38
Lab32. Interrupt(RB0)_Button1_1SegLED..........39
Lab33. Interrupt(RB0)_Button1_4SegLED..........40
Lab34. Interrupt(RB0)_Button1_LCD1602..........41
Lab35. INT_PS2_LCD1602..........42
Lab36. Watchdog_LCD1602..........43
Lab37. Internal_EEPROM_LCD1602..........44
Lab38. PWM_CCP1_V2..........45
Lab39. PWM_CCP2_V1..........46
Lab40. CCP2_IRD_LCD1602..........47
Lab41. CCP2_IRD_Beep..........49
Lab42. Interrupt(RB0)_IRD_LCD1602..........50

Lab43. AN0(8BIT)_LCD1602........52
Lab44. AN1(8BIT)_LCD1602........53
Lab45. AN2(8BIT)_LCD1602........54
Lab46. AN3(8BIT)_LCD1602........55
Lab47. AN4(8BIT)_LCD1602........56
Lab48. AN5(8BIT)_LCD1602........57
Lab49. AN6(8BIT)_LCD1602........58
Lab50. AN7(8BIT)_LCD1602........59
Lab51. AN0(10BIT)_LCD1602........60
Lab52. AN1(10BIT)_LCD1602........61
Lab53. AN2(10BIT)_LCD1602........62
Lab54. AN3(10BIT)_LCD1602........63
Lab55. AN4(10BIT)_LCD1602........64
Lab56. AN5(10BIT)_LCD1602........65
Lab57. AN6(10BIT)_LCD1602........66
Lab58. AN7(10BIT)_LCD1602........67
Lab59. AN0(8BIT)_Voltage_LCD1602........68
Lab60. AN0(10BIT)_Voltage_LCD1602........69
Lab61. AN0_CCP1_V2........70
Lab62. AN0_CCP2_V1........71
Lab63. DS18B20_LCD1602........72
Lab64. IIC_24CXX_LCD1602........73
Lab65. RTC_PCF8563_LCD1602........74
Lab66. UART_LCD1602........75
Lab67. LCD12864_ST7920........77
Contact Us........78
Other products on www.MicroHello.com........79
DISCLAIMER........82
HIGH RISK ACTIVITIES........82
TRADEMARKS........82

Hardware Connections

Lab1.Beep_1

a).Hardware Connection



b). OBJECTIVES

To understand the operation of an IO.

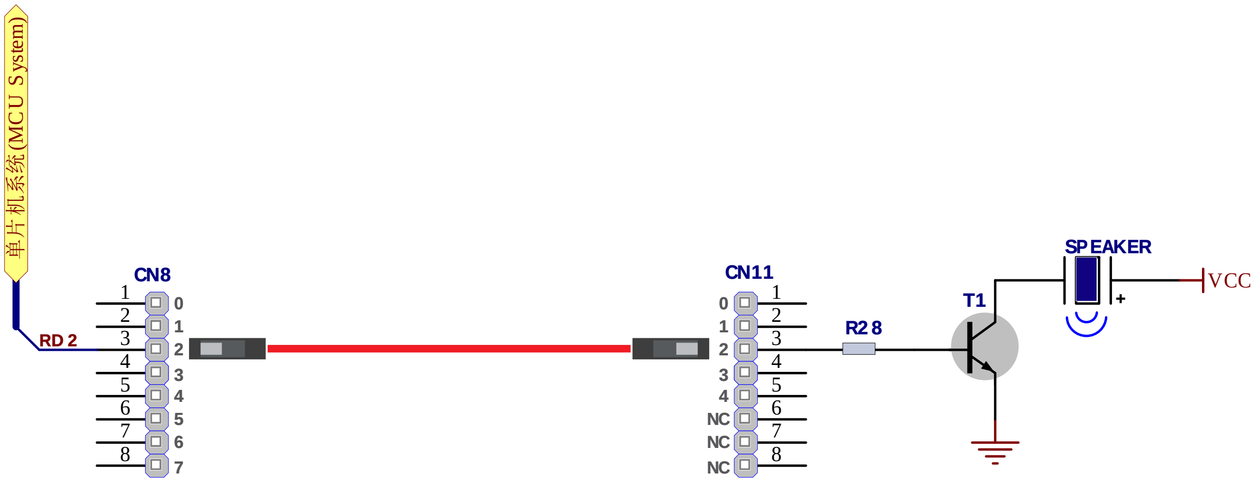
To interface and drive a Buzzer/Speaker.

c). Result

RD2=1,The transistor **T1**(S8050) is saturated and “**ON**”, and the Speaker will be continued activated.

Lab2. Beep_2

a).Hardware Connection



b). OBJECTIVES

To understand the operation of an IO.

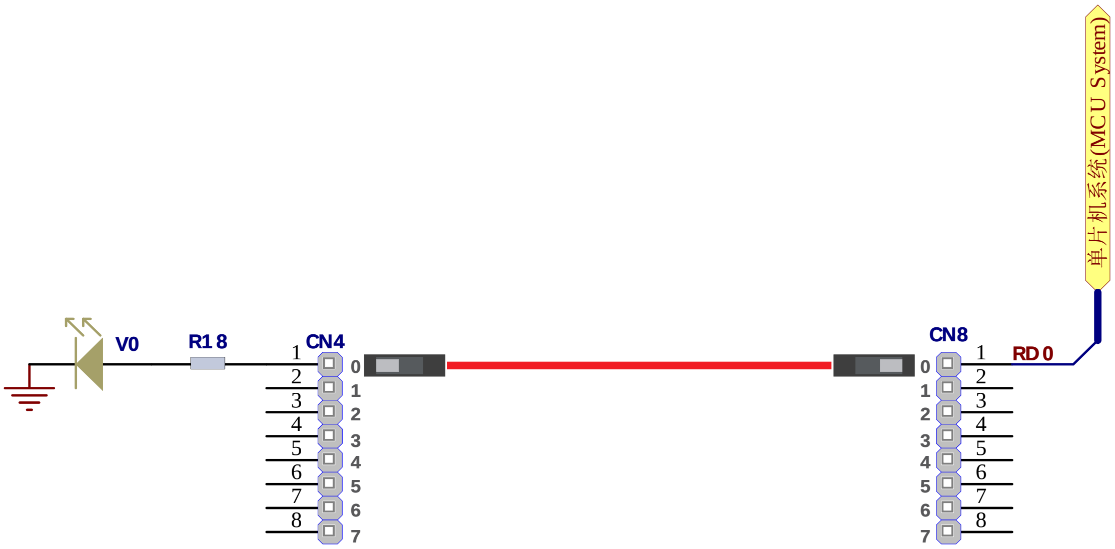
To interface and drive a Buzzer/Speaker.

c). Experimental Principle and Results

RD2=1,the transistor **T1**(S8050) is saturated and “**ON**”, and the Speaker will be activated; **RD2=0**,the transistor **T1**(S8050) is “**OFF**”, and the speaker will be stop chirping. When running this application, the buzzer sounds intermittently.

Lab3.LED_V0_ON

a).Hardware Connection



b). OBJECTIVES

To understand the operation of an IO.

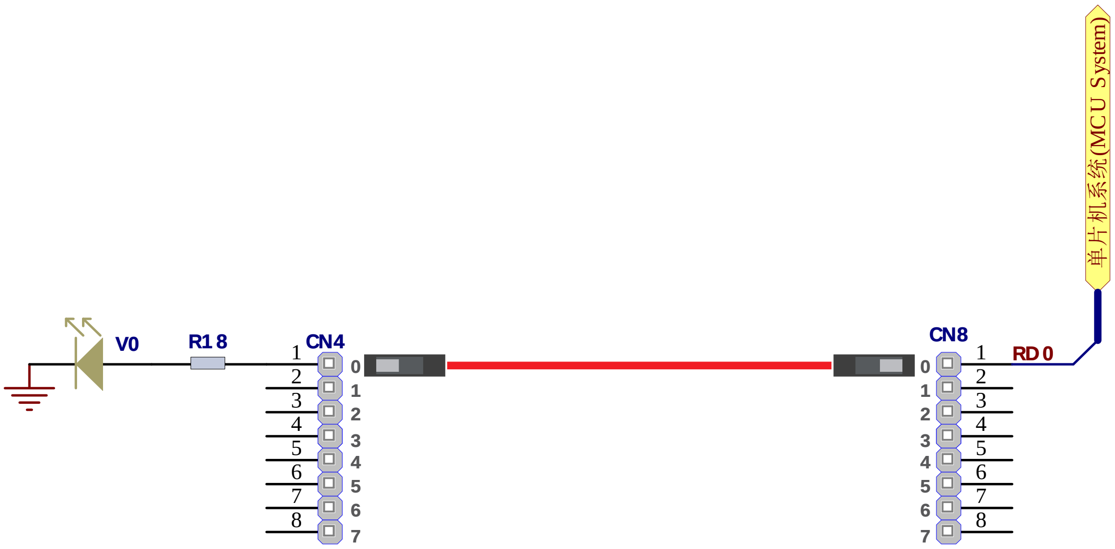
To interface and drive an LED.

c). Experimental Principle and Results

RD0=1, and the **V0** will be “**ON**”.

Lab4.LED_V0_Blinking

a).Hardware Connection



b). OBJECTIVES

To understand the operation of an IO.

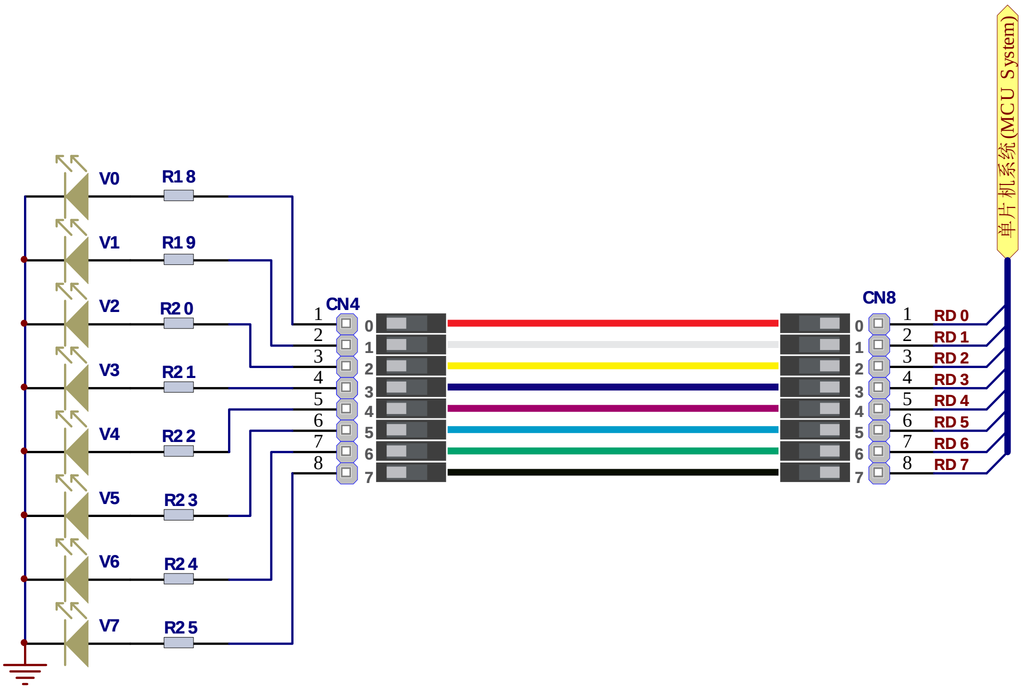
To interface and drive an LED.

c). Experimental Principle and Results

RD0=1, and the **V0** will be “**ON**”; **RD0**=0, and the **V0** will be “**OFF**”.We give it some frequency via delay,the **V0** will be blinking.

Lab5.LEDs_Run

a).Hardware Connection



b). OBJECTIVES

To understand the operation of an IO.

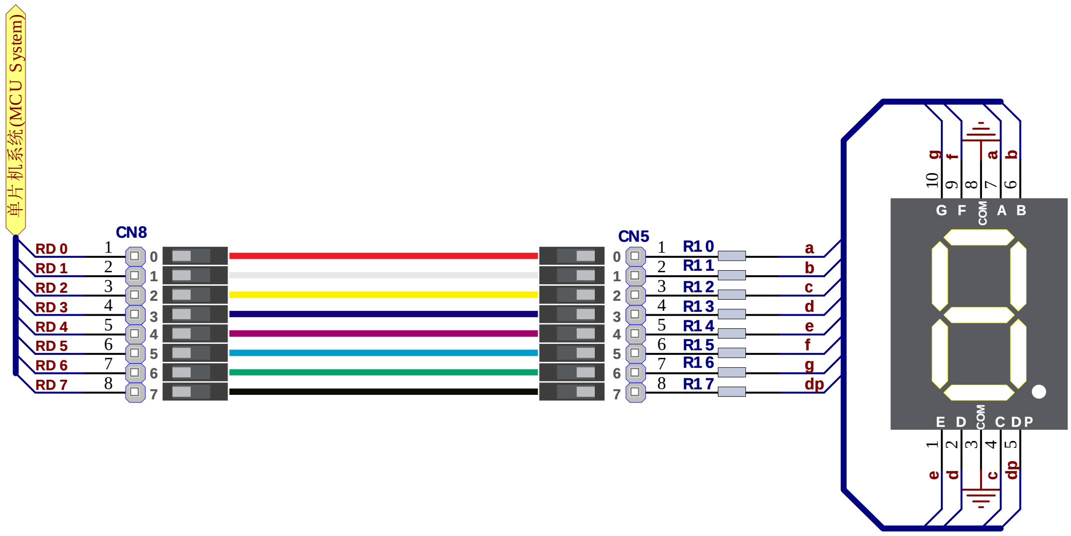
To interface and drive LEDs.

c). Experimental Principle and Results

Make a value table and send the value to the **RD** port to make the 8 LEDs running in flowing water mode , from left to right, reciprocating cycle.

Lab6.1SegLED_1

a).Hardware Connection



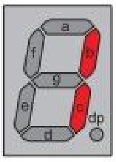
b). OBJECTIVES

To understand the operation of an IO.

To interface and drive a 1-digit segment LED display in static-state mode.

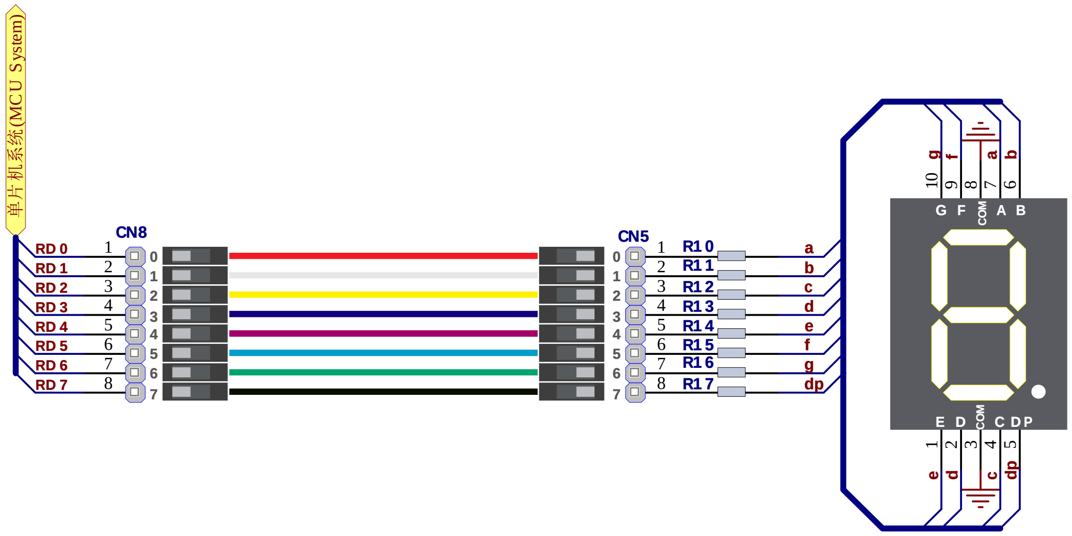
c). **Experimental Principle and Results**

segment “b” and “c”—“**ON**”,the others segments—“**OFF**”.



Lab7.1SegLED_add

a).Hardware Connection



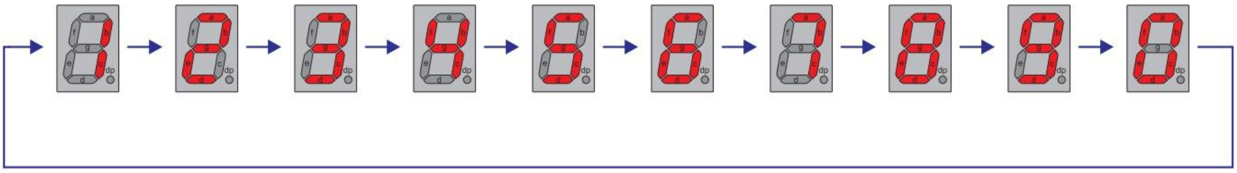
b). OBJECTIVES

To understand the operation of an IO.

To interface and drive a 1-digit segment LED display in static-state mode.

c). Experimental Principle and Results

Make a table to save value 0~9,and send them to **RD** port.

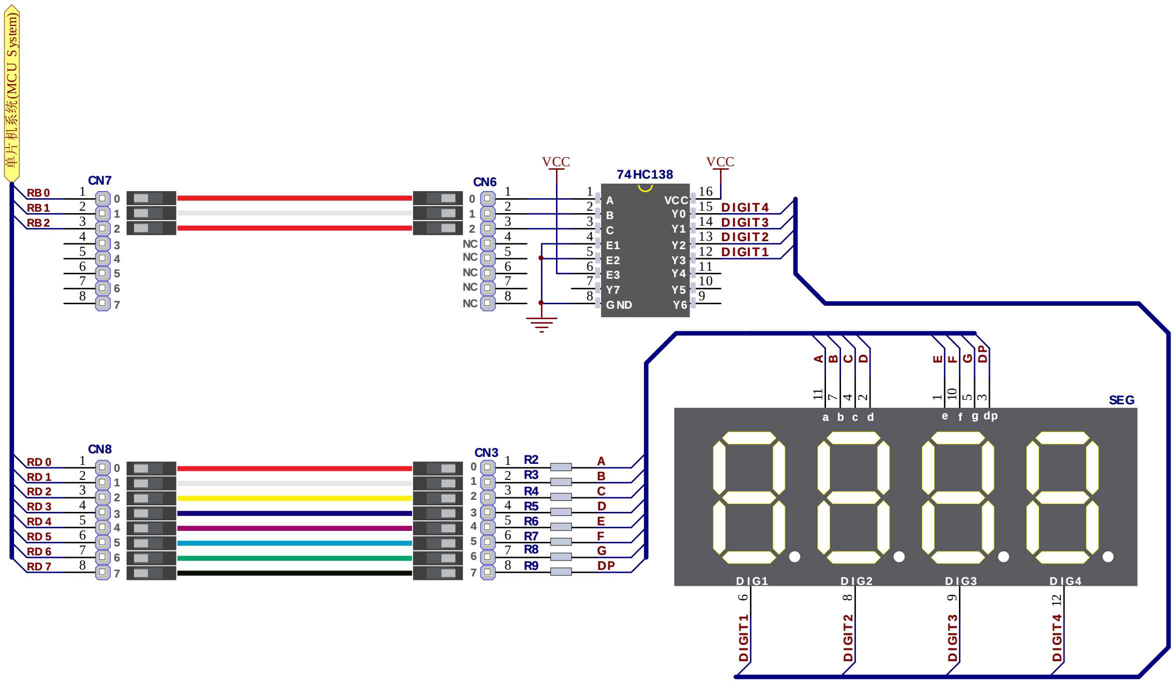


d).Table for value 0-9

	dp RD7	g RD6	f RD5	e RD4	d RD3	c RD2	b RD1	a RD0
0	0	0	1	1	1	1	1	1
1	0	0	0	0	0	1	1	0
2	0	1	0	1	1	0	1	1
3	0	1	0	0	1	1	1	1
4	0	1	1	0	0	1	1	0
5	0	1	1	0	1	1	0	1
6	0	1	1	1	1	1	0	1
7	0	0	0	0	0	1	1	1
8	0	1	1	1	1	1	1	1
9	0	1	1	0	1	1	1	1

Lab8.4SegLED_0

a).Hardware Connection



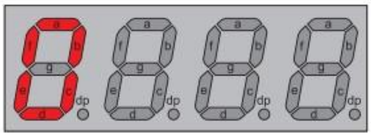
b). OBJECTIVES

To understand the operation of an IO.

To interface and drive a 4-digit segment LED display in static-state mode.

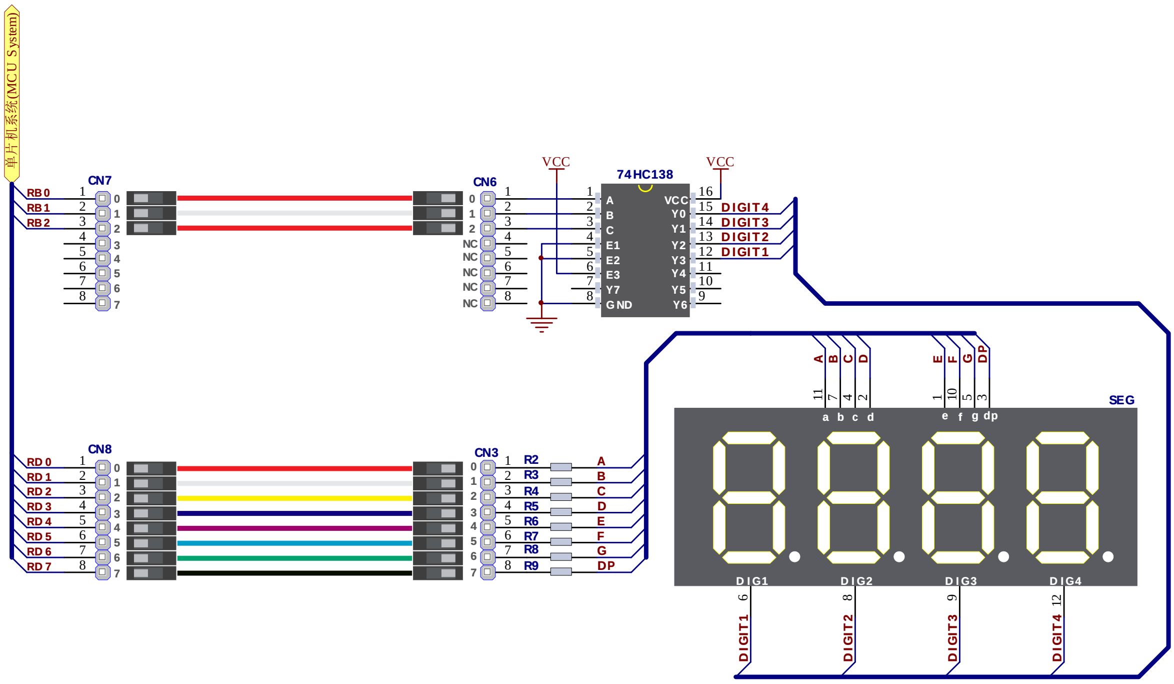
c). Experimental Principle and Results

In the code, set (A=0; B=0; C=0;) at the input of **74HC138**, and obtain (Y0=0, Y1=1, Y2=1, Y3=1) at the output of **74HC138**, then simultaneously send "0" (0x3f) to the RD port. Starting from the right, the fourth digit of the segmented LED lights up and displays "0".



Lab9.4SegLED_1

a).Hardware Connection



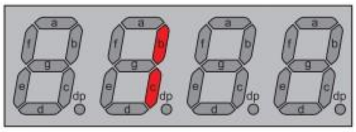
b). OBJECTIVES

To understand the operation of an IO.

To interface and drive a 4-digit segment LED display in static-state mode.

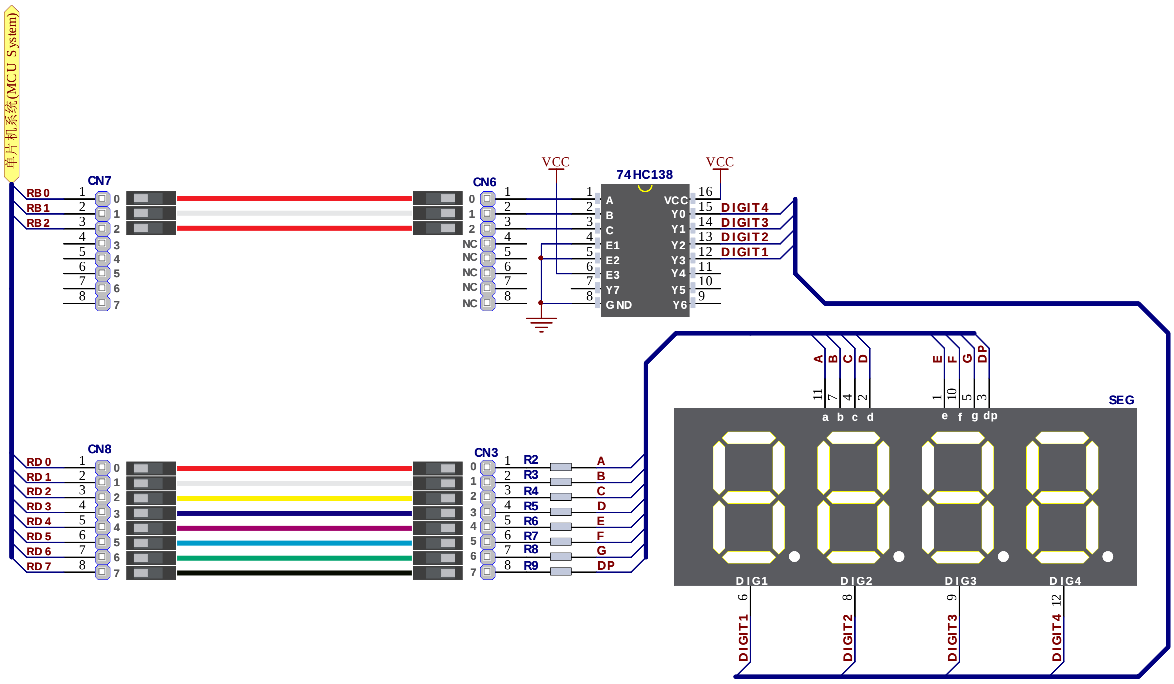
c). Experimental Principle and Results

In the code, set (A=1; B=0; C=0;) at the input of **74HC138**, and obtain (Y0=1, Y1=0, Y2=1, Y3=1) at the output of **74HC138**, then simultaneously send "1" (0x06) to the RD port. Starting from the right, the third digit of the segmented LED lights up and displays "1".



Lab10.4SegLED_2

a).Hardware Connection



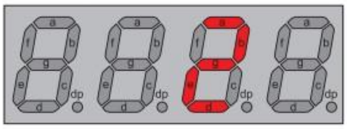
b). OBJECTIVES

To understand the operation of an IO.

To interface and drive a 4-digit segment LED display in static-state mode.

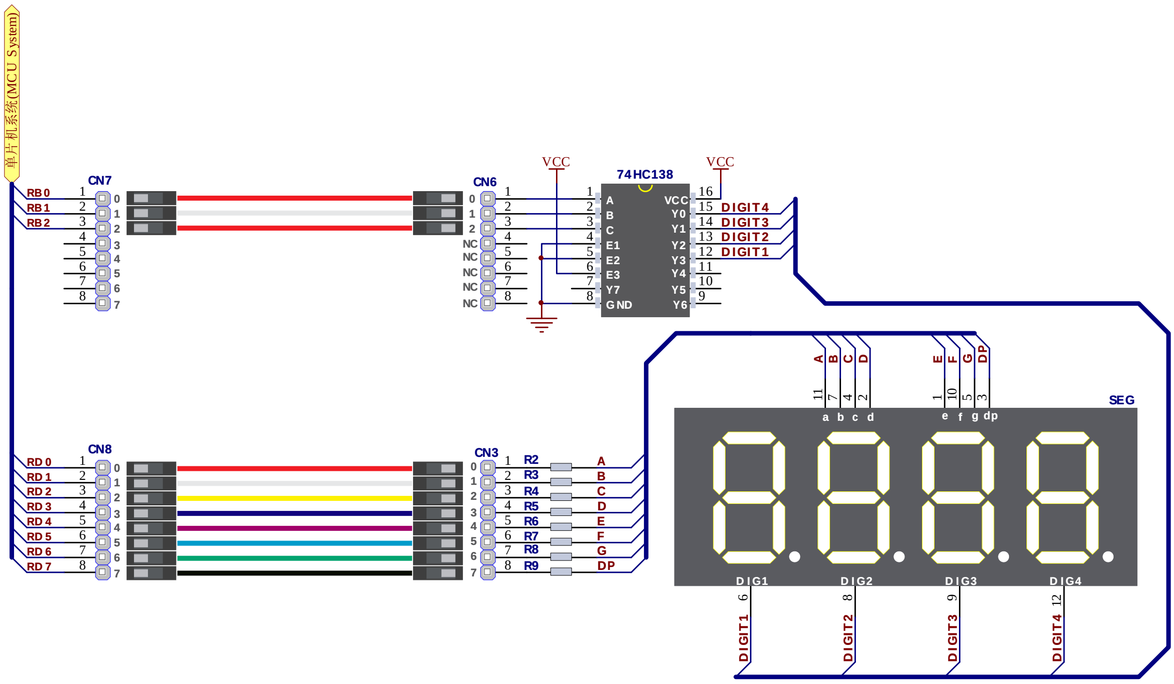
c). Experimental Principle and Results

In the code, set (A=0; B=1; C=0;) at the input of **74HC138**, and obtain (Y0=1, Y1=1, Y2=0, Y3=1) at the output of **74HC138**, then simultaneously send "2" (0x5B) to the RD port. Starting from the right, the second digit of the segmented LED lights up and displays "2".



Lab11.4SegLED_3

a).Hardware Connection



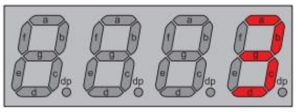
b). OBJECTIVES

To understand the operation of an IO.

To interface and drive a 4-digit segment LED display in static-state mode.

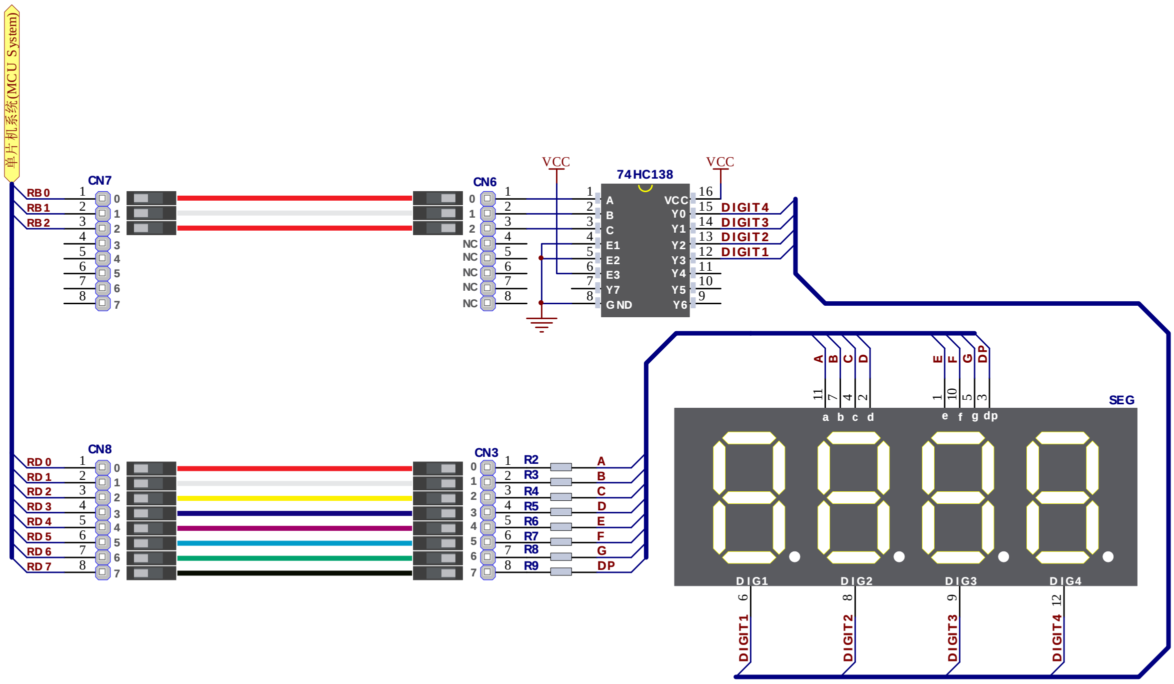
c). Experimental Principle and Results

In the code, set (A=1; B=1; C=0;) at the input of **74HC138**, and obtain (Y0=1, Y1=1, Y2=1, Y3=0) at the output of **74HC138**, then simultaneously send "3" (0x4F) to the RD port. Starting from the right, the first digit of the segmented LED lights up and displays "3".



Lab12.4SegLED_0123

a).Hardware Connection



b). OBJECTIVES

To understand the operation of an IO.

To interface and drive a 4-digit segment LED display in dynamic-state mode.

c). Experimental Principle and Results

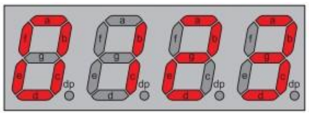
Set **Y0-Y3** to “0” one by one at some delay, and send “**0**”-“3”(0x3F, 0x06, 0x5B, 0x4F) to the **RD** port. The four-digit segment LED will display “0123”.

The Source Code:

```
#include <pic.h>                    //Head File
__CONFIG(1,HS) ;    //Bit Configuration
__CONFIG(2,WDTDIS) ;
__CONFIG(3,PBADDIS) ;
__CONFIG(4,LVPDIS&XINSTDIS) ;
//---------------------------------------
//Table for Segment code
const unsigned char LED[10]=
{
        0B00111111,                   //"0"，0x3F
        0B00000110,                   //"1"，0x06
        0B01011011,                   //"2"，0x5B
        0B01001111,                   //"3"，0x4F
        0B01100110,                   //"4"，0x66
        0B01101101,                   //"5"，0x6D
        0B01111101,                   //"6"，0x7D
        0B00000111,                   //"7"，0x07
```

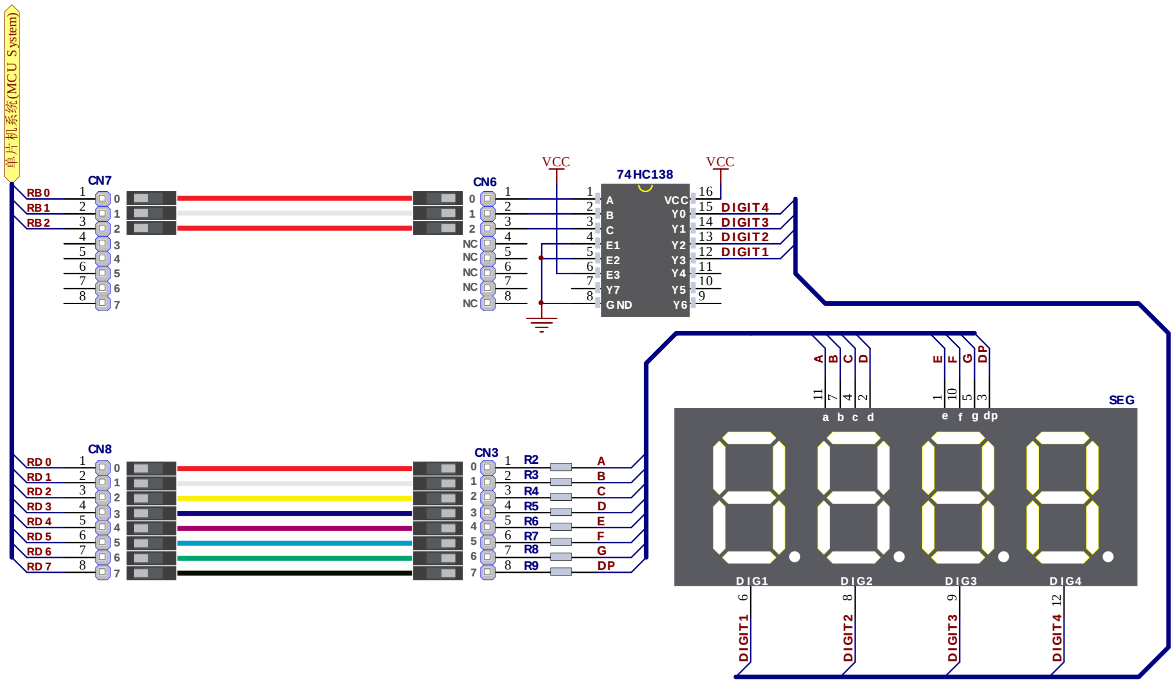
```
    0B01111111,                //"8"，0x7F
    0B01101111,                //"9"，0x6F
};

//---------------------------------------
#define U5A   RB0               // U5(74HC138)'s "A" Pin-------RB0
#define U5B   RB1               // U5(74HC138) 's "B" Pin-------RB1
#define U5C   RB2               // U5(74HC138) 's "C" Pin-------RB2
//---------------------------------------

void main(void)                //main
{

    unsigned char c=0;        // Define a char type variable for delay
    unsigned char d=0;        // Define a char type variable to control the display position

    TRISB=0B11111000;          // Initialize the input and output directions of RB7-RB0
    TRISD=0B00000000;          // Initialize the input and output direction of RD7-RD0
    PORTB=0B00000000;          // Initialize the value of RB7-RB0
    PORTD=0B00000000;          // Initialize the value of RD7-RD0

    while(1)                   // Dead loop, after the MCU is initialized, this dead loop will continue to run
    {

        for(c=0;c<250;c++);// Create a loop of 0-250 as a delay
        PORTD=0;               //Turn off the display
        if(++d>3) d=0;
        if(d==0)               // If d=0, the fourth digit of the segmented LED will be lighted up.
        {
            U5A=0;             //U5A=0,U5B=0,U5C=0,Select the fourth digit
            U5B=0;             //U5A=0,U5B=0,U5C=0
            U5C=0;             //U5A=0,U5B=0,U5C=0
            PORTD=LED[0];   //Display "0"
        }
        else if(d==1)          // If d=1, the third digit of the segmented LED will be lighted up.
        {
            U5A=1;             //U5A=1,U5B=0,U5C=0, Select the third digit
            U5B=0;             //U5A=1,U5B=0,U5C=0
            U5C=0;             //U5A=1,U5B=0,U5C=0
            PORTD=LED[1];   // Display "1"
        }
        else if(d==2)          // If d=2, the second digit of the segmented LED will be lighted up.
        {
            U5A=0;             //U5A=0,U5B=1,U5C=0, Select the second digit
            U5B=1;             //U5A=0,U5B=1,U5C=0
```

```
                U5C=0;              //U5A=0,U5B=1,U5C=0
                PORTD=LED[2];    // Display "2"
            }
            else if(d==3)           // If d=3, the first digit of the segmented LED will be lighted up.
            {
                U5A=1;              //U5A=1,U5B=1,U5C=0, Select the first digit
                U5B=1;              //U5A=1,U5B=1,U5C=0
                U5C=0;              //U5A=1,U5B=1,U5C=0
                PORTD=LED[3];    // Display "3"
            }
        }
}
```



Lab13.4SegLED_Add

a).Hardware Connection



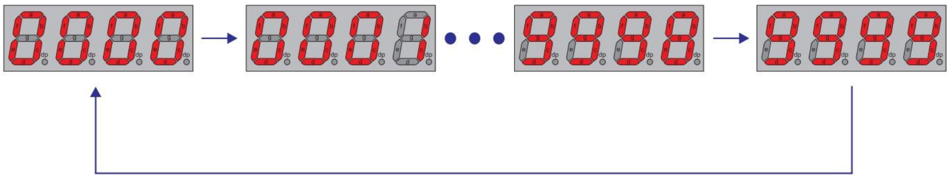
b). OBJECTIVES

To understand the operation of an IO.

To interface and drive a 4-digit segment LED display in dynamic-state mode.

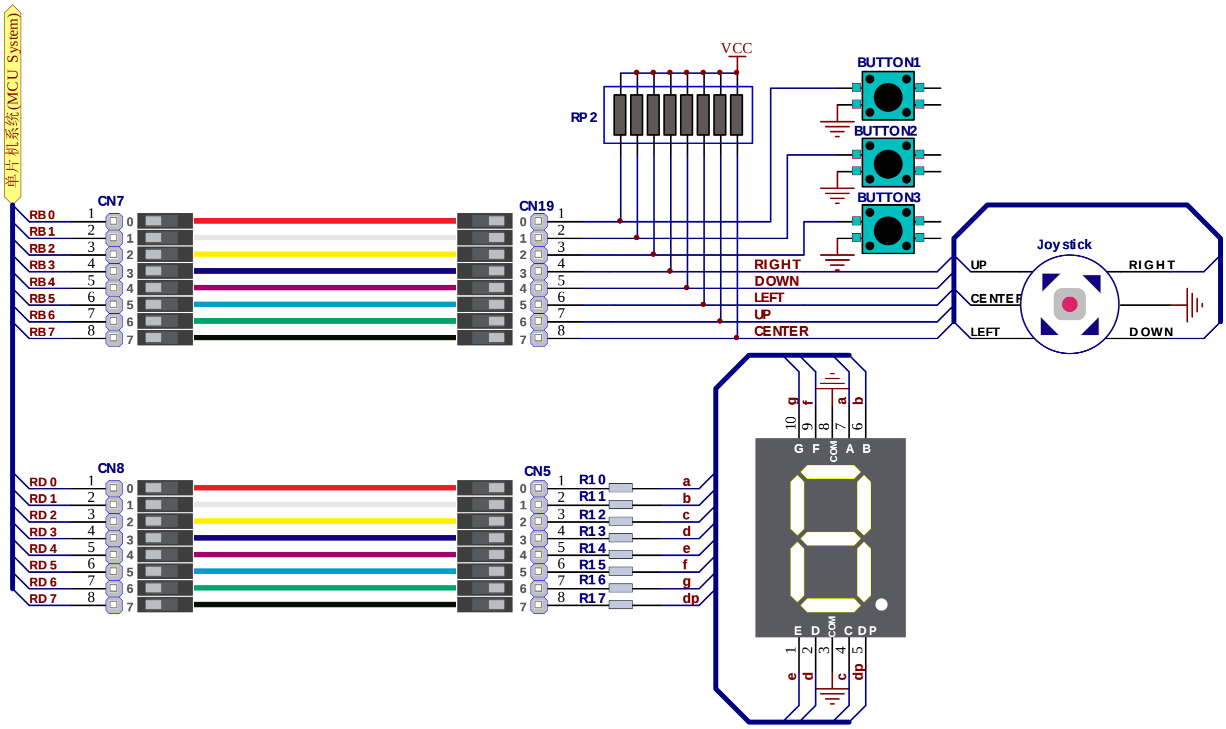
c). Experimental Principle and Results

Set **Y0-Y3** to “0” one by one, and send **value** to the **RD** port at the same time and then **value+1**.



Lab14.Jostick_Buttons_1SegLED

a).Hardware Connection



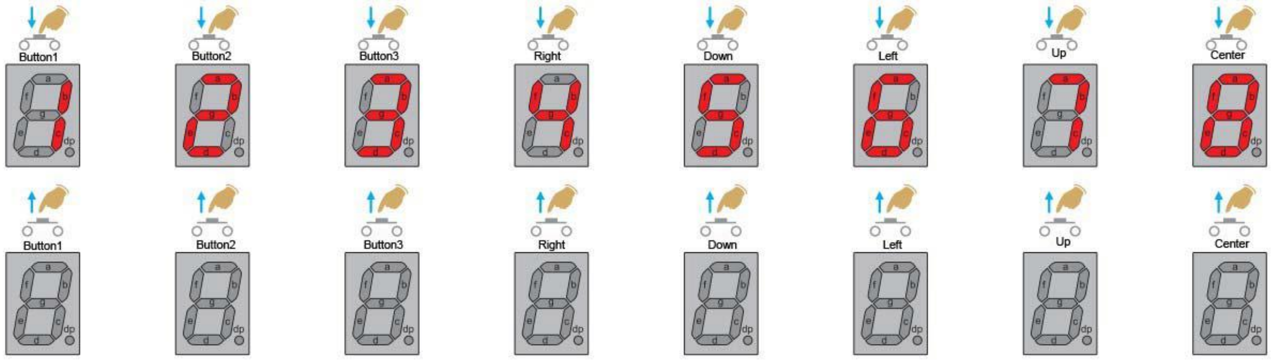
b). OBJECTIVES

To understand the operation of an IO.

To interface and drive a dynamic-state SEGLED.

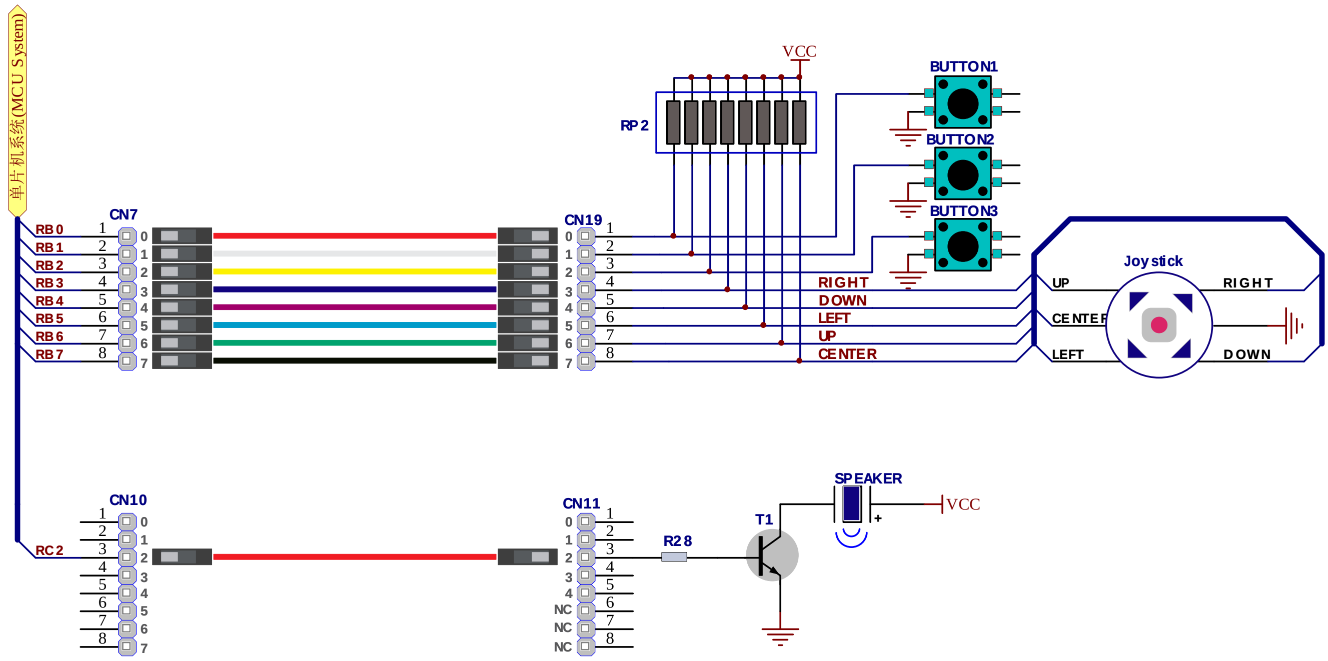
c). Experimental Principle and Results

The **Key** value display on the 4-digit segled when **Button1**-**Button3** or **Joystick** is pressed.



Lab15.Jostick_Buttons_Beep

a).Hardware Connection



b). OBJECTIVES

To understand the operation of an IO.

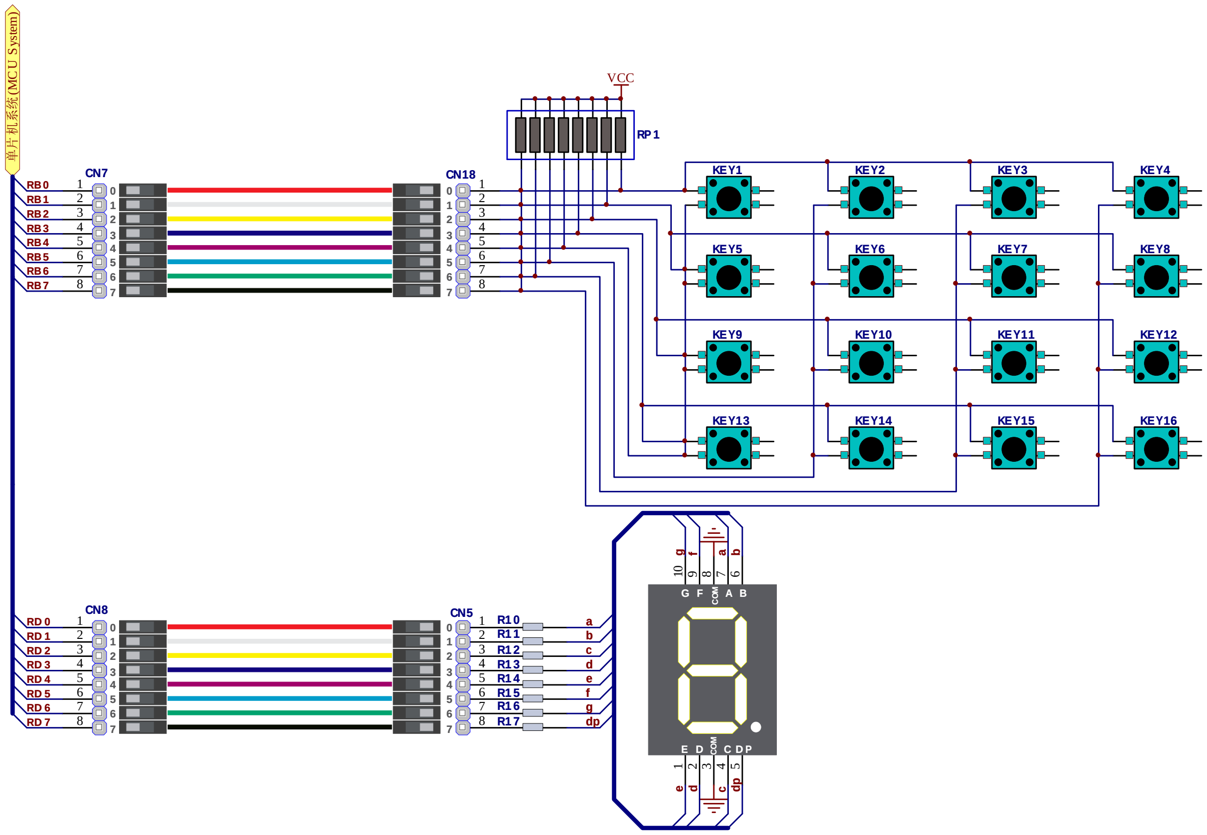
To interface and drive a Buzzer/speaker.

c). Experimental Principle and Results

The **Speaker** will be activated when **Button1**-**Button3** or **Joystick** is pressed.

Lab16. 4X4Keyboard_1SegLED

a).Hardware Connection



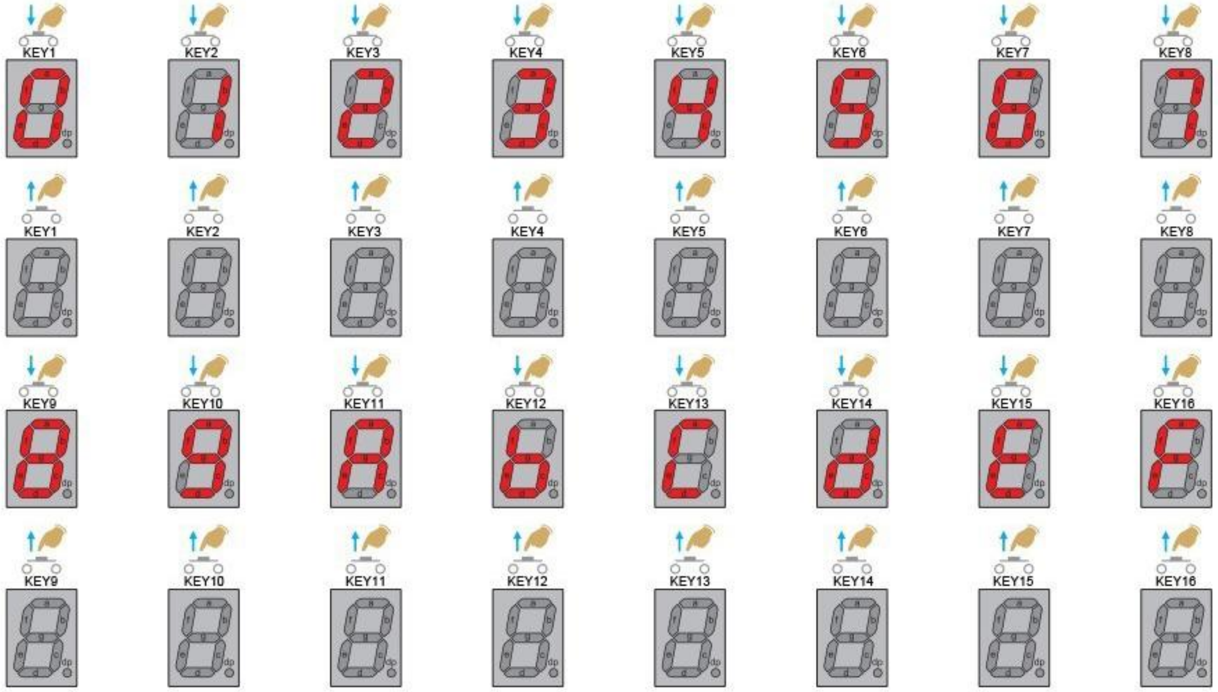
b). OBJECTIVES

To understand the operation of an IO.

To understand the keypad scan tech.

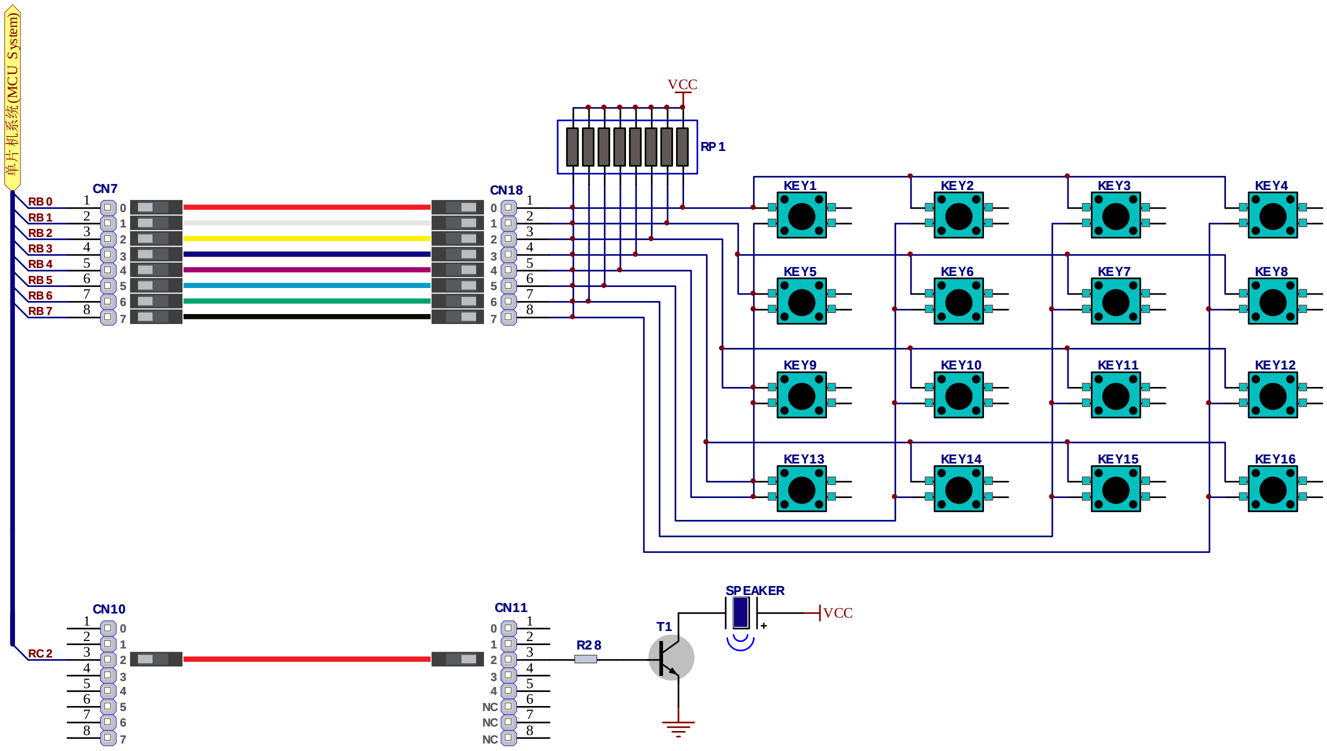
c). Experimental Principle and Results

The **Key** value display on the 1-digit segled when **KEY1-KEY16** is pressed.


KEY1
KEY2
KEY3
KEY4
KEY5
KEY6
KEY7
KEY8
KEY1
KEY2
KEY3
KEY4
KEY5
KEY6
KEY7
KEY8
KEY9
KEY10
KEY11
KEY12
KEY13
KEY14
KEY15
KEY16
KEY9
KEY10
KEY11
KEY12
KEY13
KEY14
KEY15
KEY16

Lab17. 4X4Keyboard_Beep

a).Hardware Connection



b). OBJECTIVES

To understand the operation of an IO.

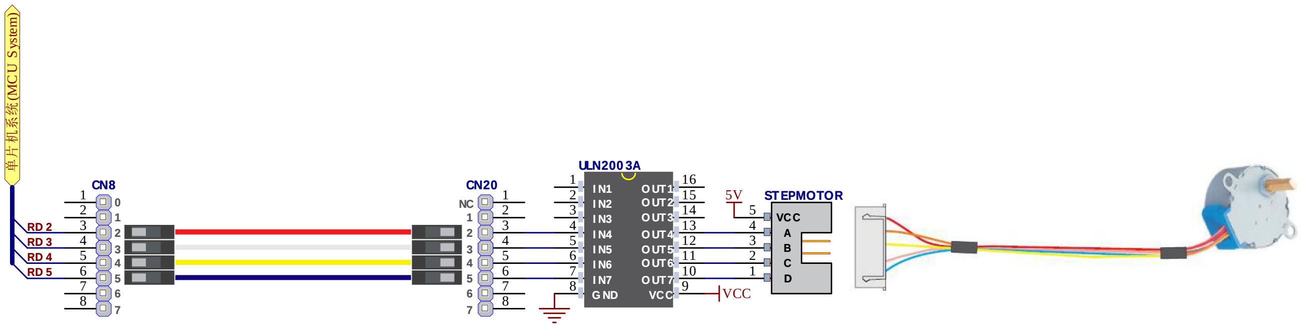
To understand the keypad scan tech.

c). Experimental Principle and Results

The **Speaker** will be activated when **KEY1**-**KEY16** is pressed.

Lab18. Stepmotor

a).Hardware Connection

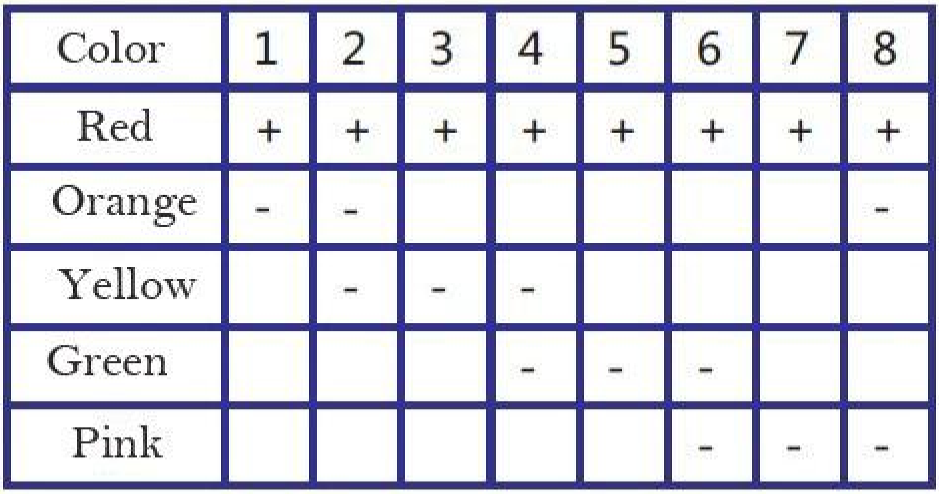


b). OBJECTIVES

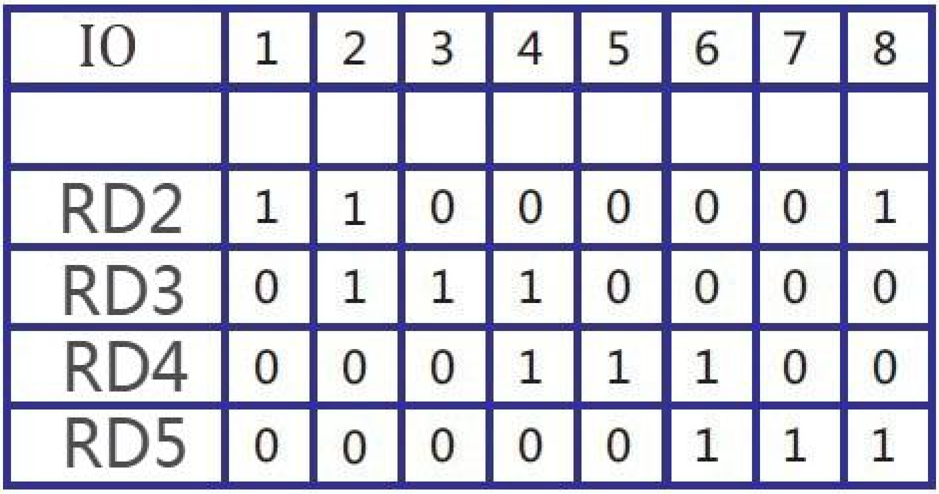
To understand how to control a stepmotor.

c). Experimental Principle and Results

We use a 4-phase step motor in this application. The step motor will rotate when we add different voltage on the 4 lines.



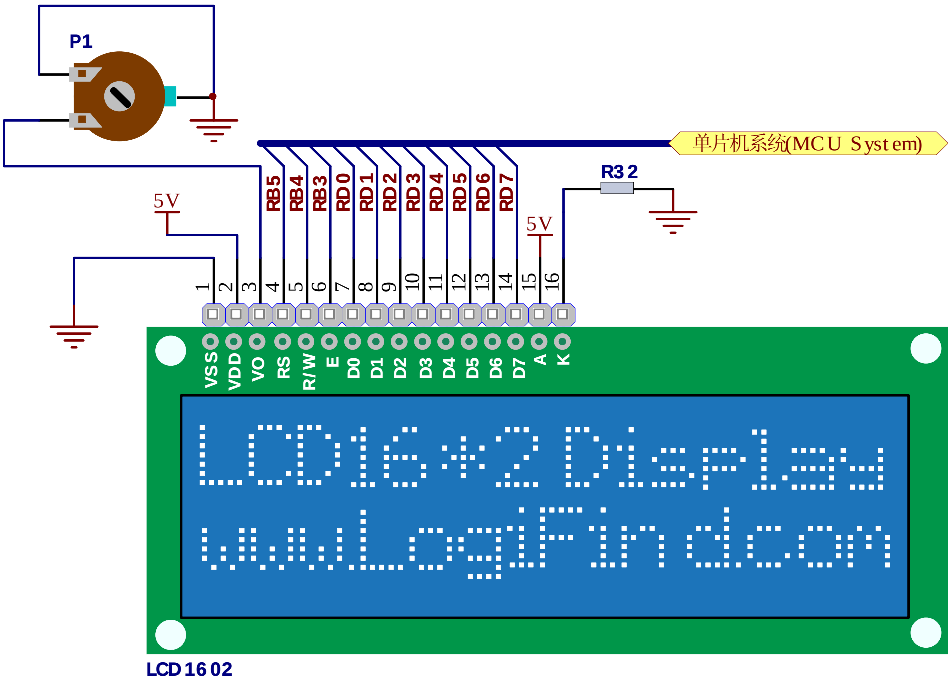
Color	1	2	3	4	5	6	7	8
Red	+	+	+	+	+	+	+	+
Orange	-	-						-
Yellow		-	-	-				
Green				-	-	-		
Pink						-	-	-



IO	1	2	3	4	5	6	7	8
RD2	1	1	0	0	0	0	0	1
RD3	0	1	1	1	0	0	0	0
RD4	0	0	0	1	1	1	0	0
RD5	0	0	0	0	0	1	1	1

Lab19. LCD1602_Char

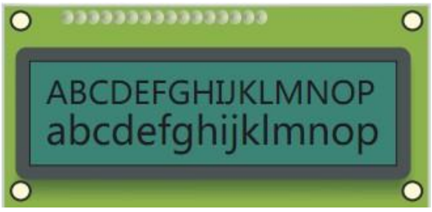
a).Hardware Connection



b). OBJECTIVES

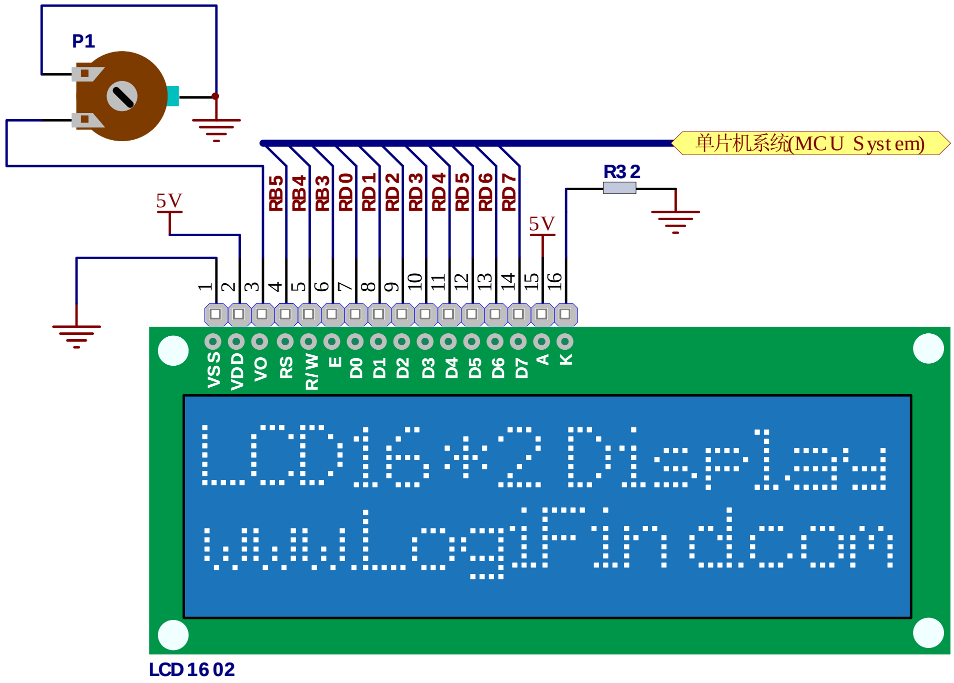
To understand the operation of a Char LCD.

c). Experimental Principle and Results



Lab20. LCD1602_Num

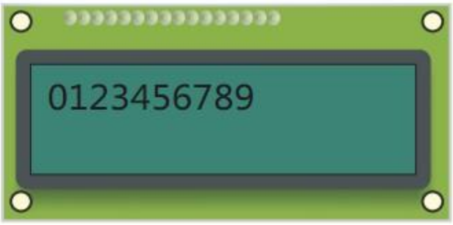
a).Hardware Connection



b). OBJECTIVES

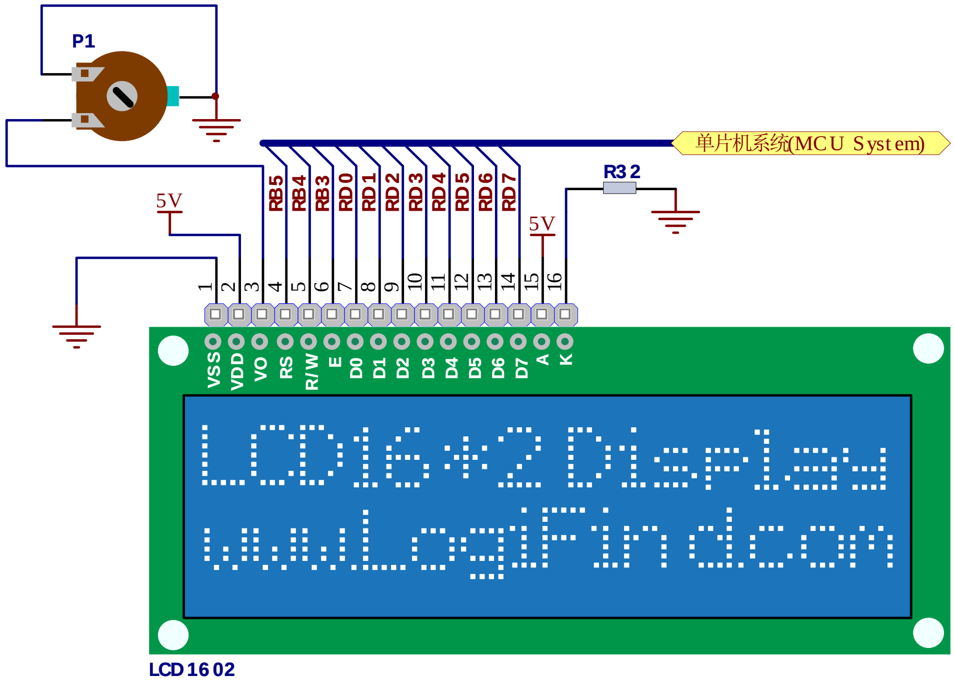
To understand the operation of a Char LCD.

c). Experimental Principle and Results



Lab21. LCD1602_Add

a).Hardware Connection



b). OBJECTIVES

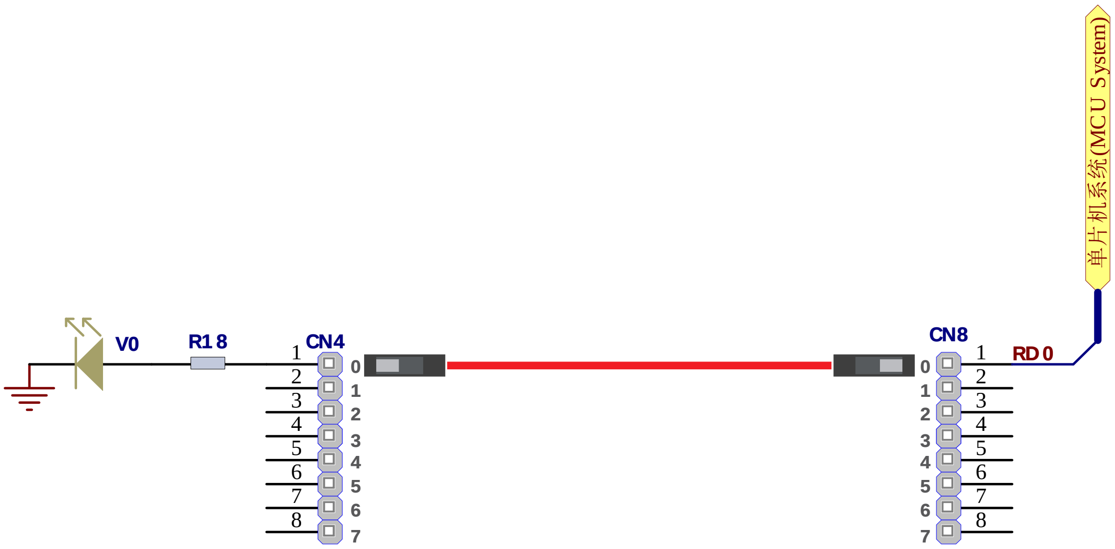
To understand the operation of a Char LCD.

c). Experimental Principle and Results

The numerical value on LCD1602 display continues to increase.

Lab22. Timer0_V0

a).Hardware Connection



b). OBJECTIVES

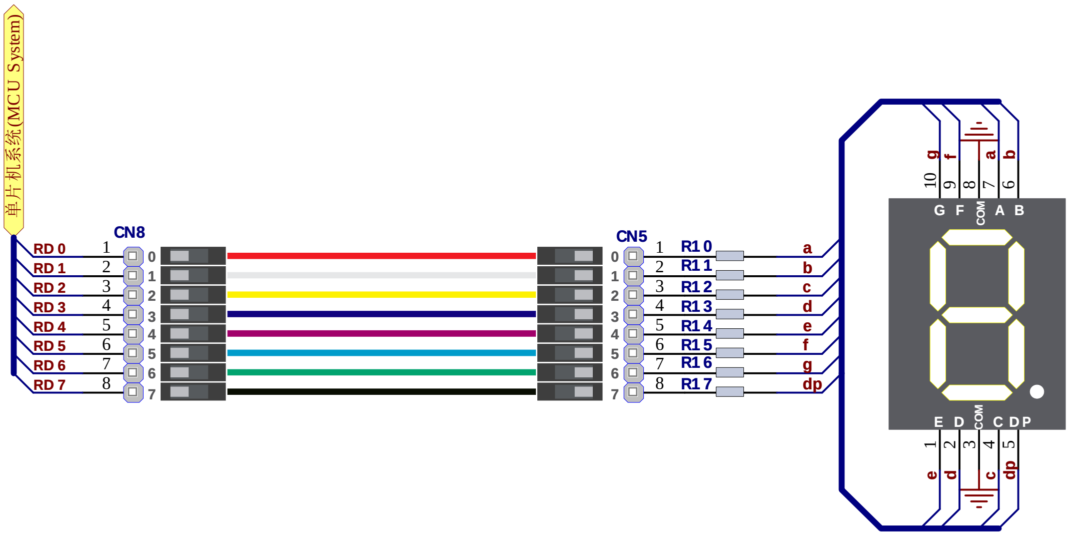
To understand the operation of timer.

c). Experimental Principle and Results

Set timer T0 250us*200=0.5s. Light Emitting Diode V0 will be blinking at 0.5S intervals.

Lab23. Timer0_1SegLED

a).Hardware Connection

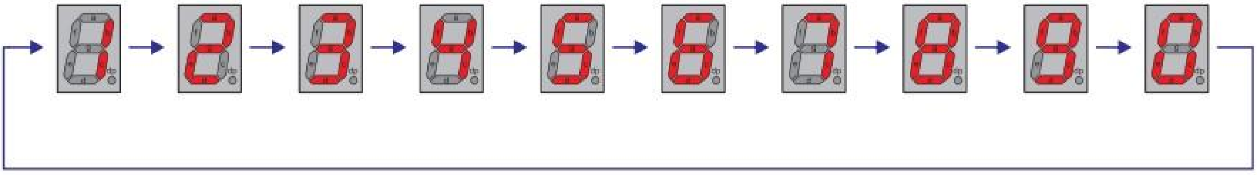


b). OBJECTIVES

To understand the operation of timer.

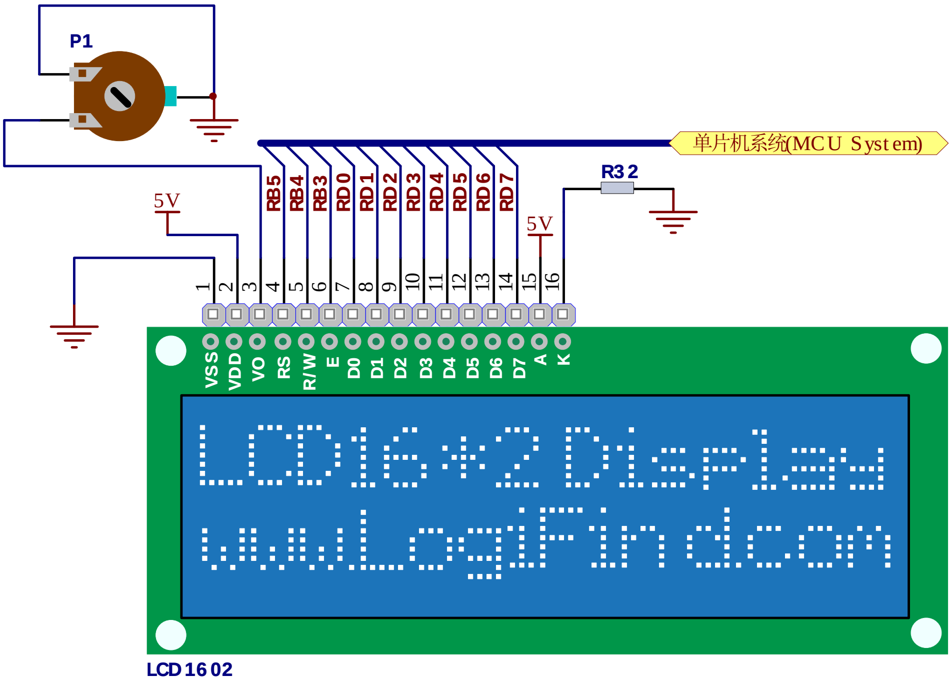
c). Experimental Principle and Results

Set timer T0 250us*400=1s,0,1,2,3,…9,0,1,2…..



Lab24. Timer0_LCD1602

a).Hardware Connection



b). OBJECTIVES

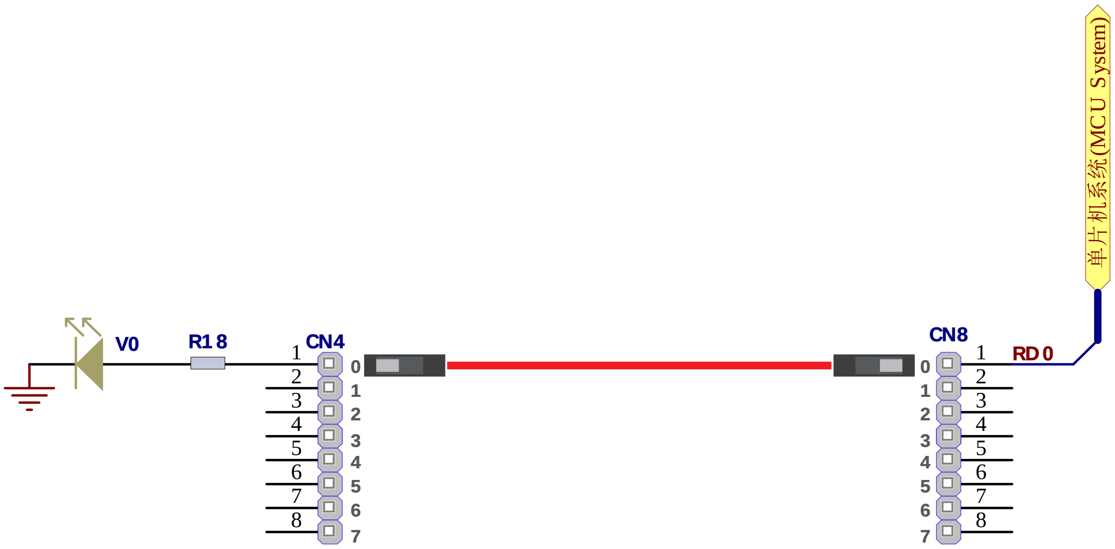
To understand the operation of timer.

c). Experimental Principle and Results

Set timer T0 250us*400=1s, 00000,00001,00002…….. 50000,00000,00001………

Lab25. Timer1_V0

a).Hardware Connection



b). OBJECTIVES

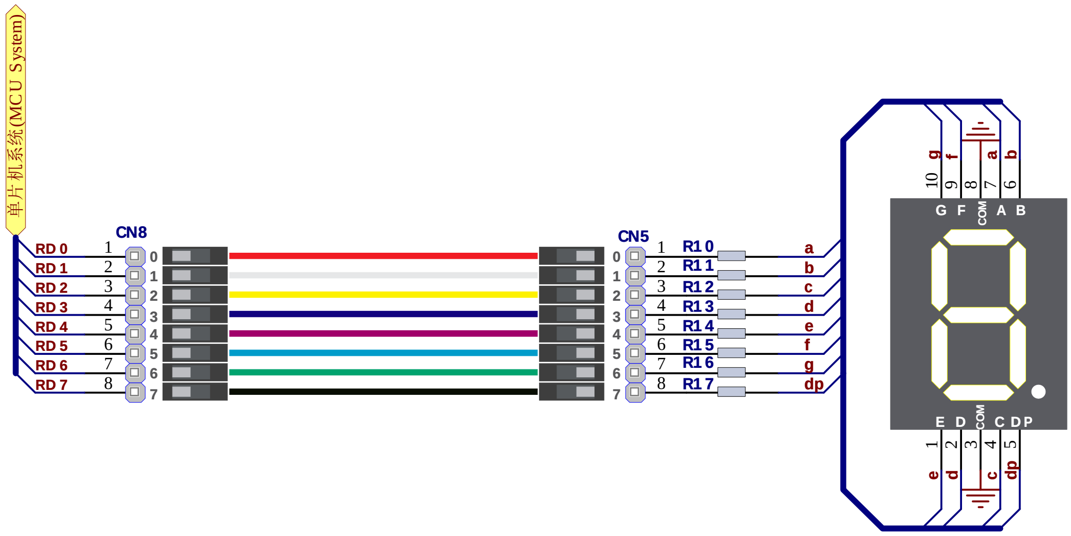
To understand the operation of timer.

c). Experimental Principle and Results

Set timer T1 250us*200=0.5s. Light Emitting Diode V0 will be blinking at 0.5S intervals.

Lab26. Timer1_1SegLED

a).Hardware Connection

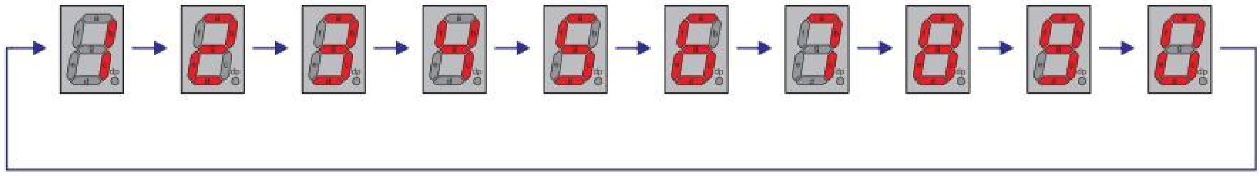


b). OBJECTIVES

To understand the operation of timer.

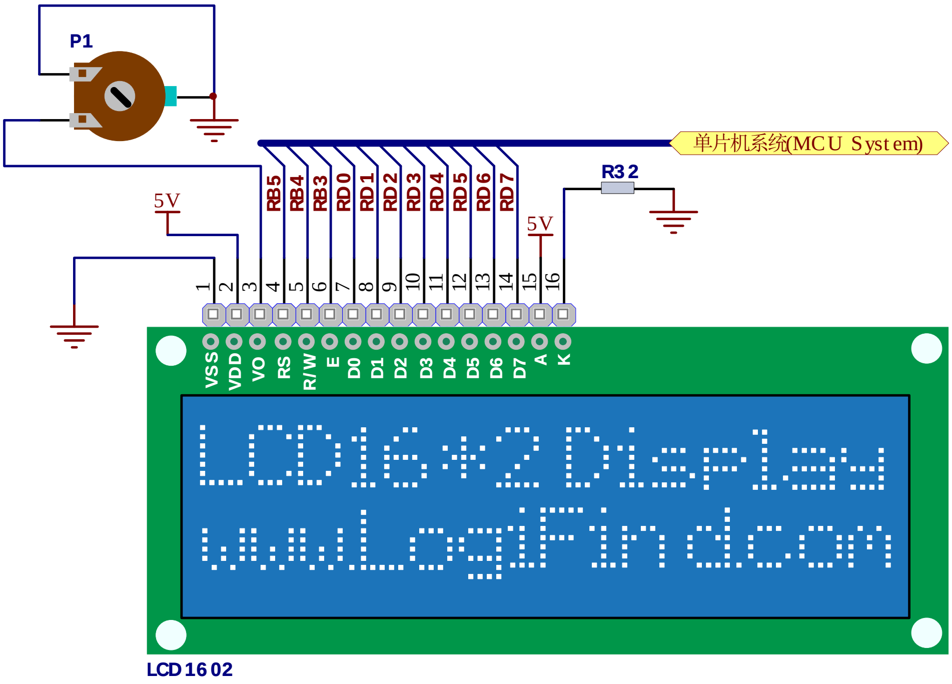
c). Experimental Principle and Results

Set timer T1 250us*400=1s,0,1,2,3,…9,0,1,2…..



Lab27. Timer1_LCD1602

a).Hardware Connection



b). OBJECTIVES

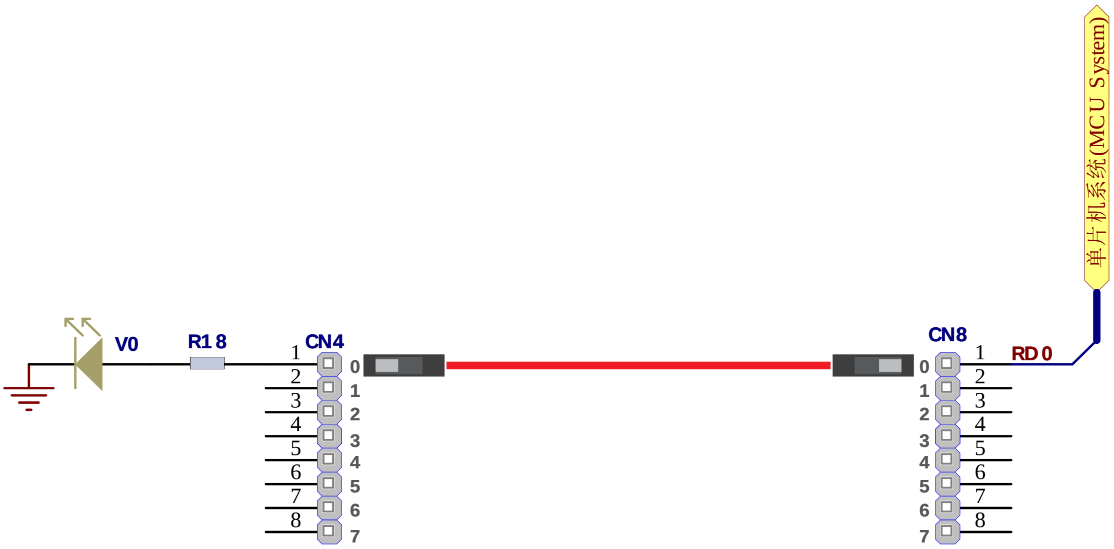
To understand the operation of timer.

c). Experimental Principle and Results

Set timer T1 250us*400=1s, 00000,00001,00002…….. 50000,00000,00001………

Lab28. Timer2_V0

a).Hardware Connection



b). OBJECTIVES

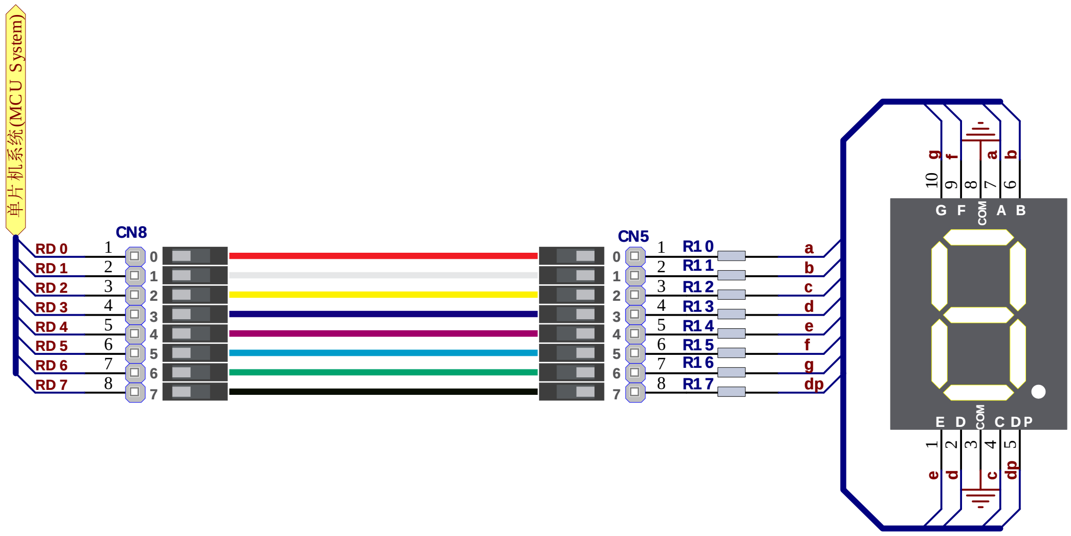
To understand the operation of timer.

c). Experimental Principle and Results

Set timer T2 250us*200=0.5s. Light Emitting Diode V0 will be blinking at 0.5S intervals.

Lab29. Timer2_1SegLED

a).Hardware Connection

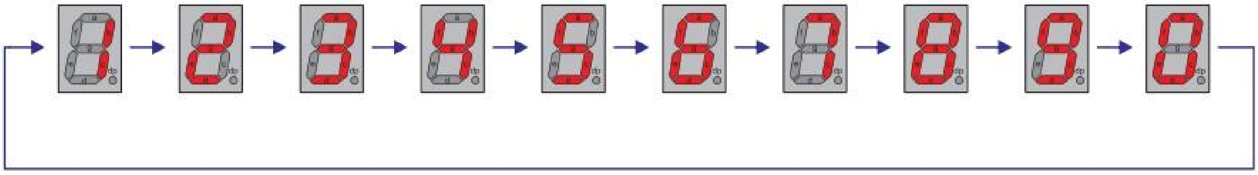


b). OBJECTIVES

To understand the operation of timer.

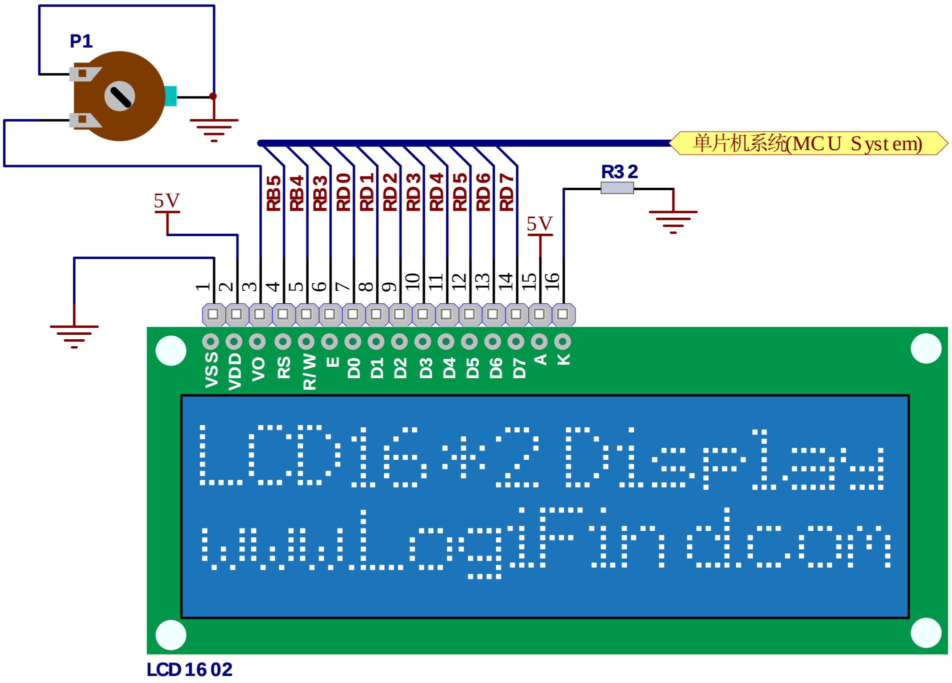
c). Experimental Principle and Results

Set timer T2 250us*400=1s,0,1,2,3,…9,0,1,2…..



Lab30. Timer2_LCD1602

a).Hardware Connection



b). OBJECTIVES

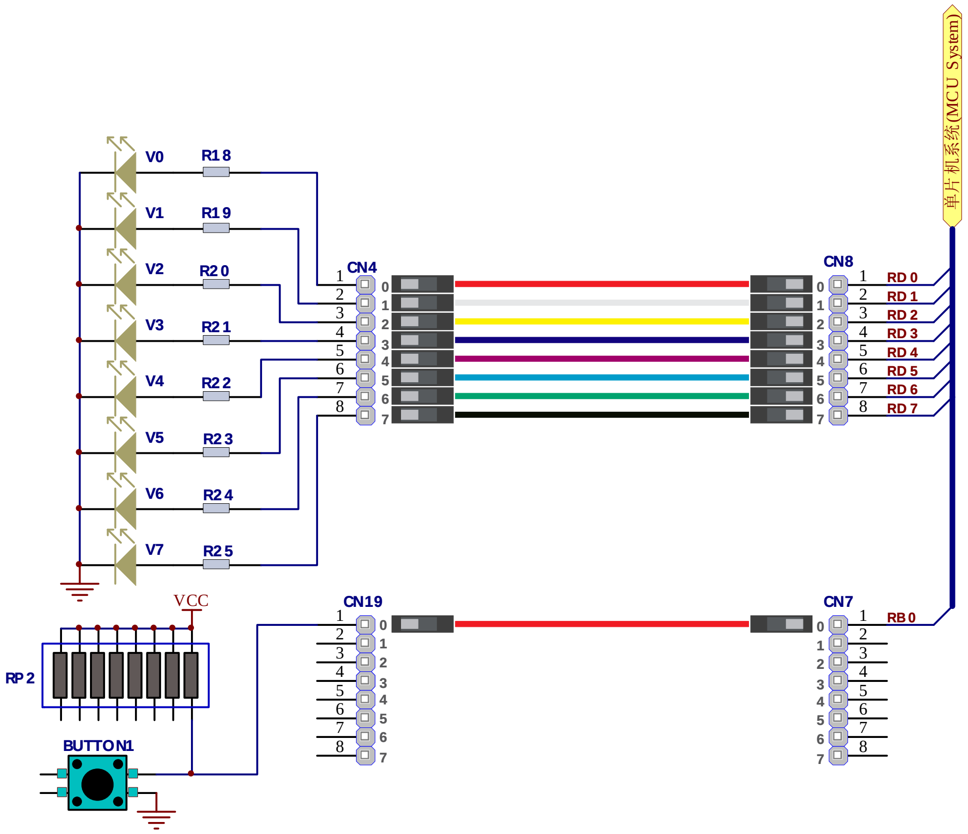
To understand the operation of timer.

c). Experimental Principle and Results

Set timer T2 250us*400=1s, 00000,00001,00002…….. 50000,00000,00001………

Lab31. Interrupt(RB0)_Button1_LEDs

a).Hardware Connection



b). OBJECTIVES

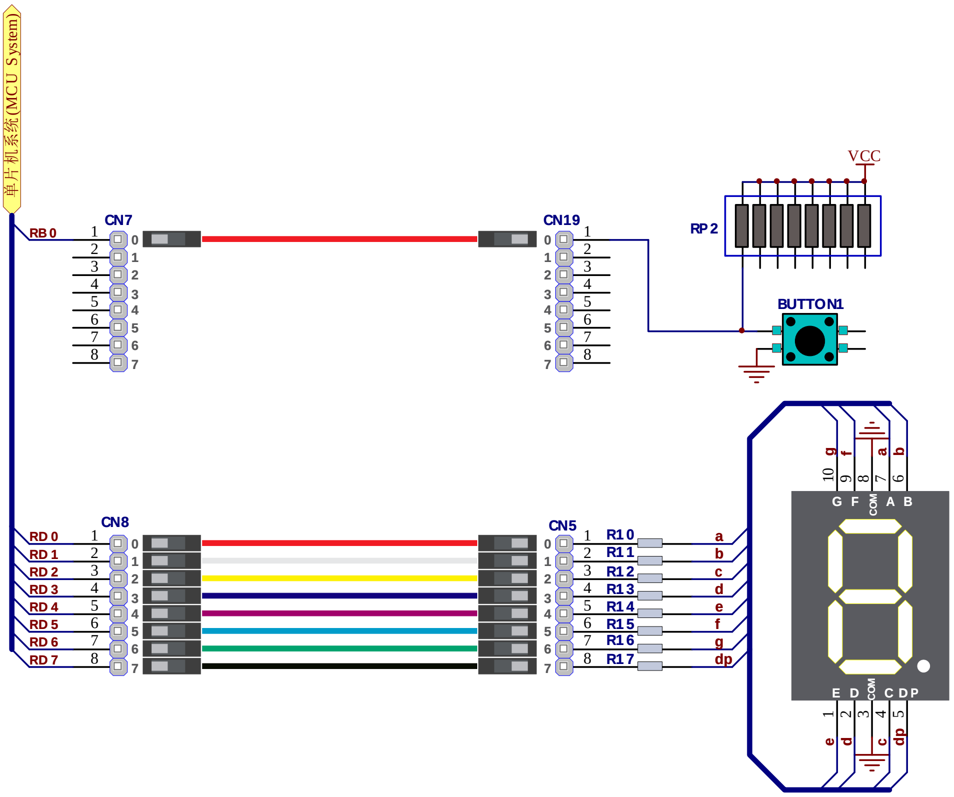
To understand the operation of Interrupt on RB0 pin.

c). Experimental Principle and Results

Press **BUTTON1**(**RB0**),then LEDs will restart.

Lab32. Interrupt(RB0)_Button1_1SegLED

a).Hardware Connection



b). OBJECTIVES

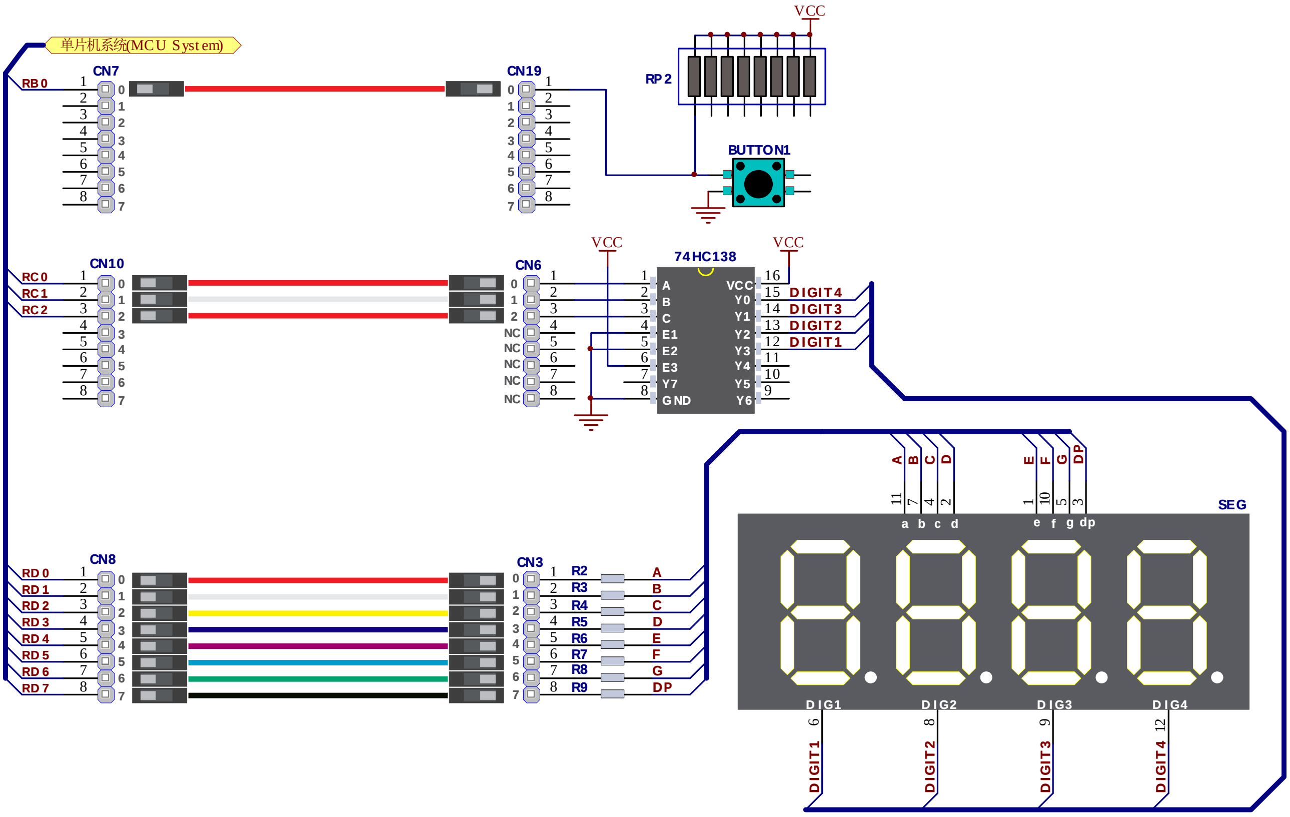
To understand the operation of Interrupt on RB0 pin.

c). Experimental Principle and Results

Press **BUTTON1**(**RB0**),then the 1-digit segled will restart to display .

Lab33. Interrupt(RB0)_Button1_4SegLED

a).Hardware Connection



b). OBJECTIVES

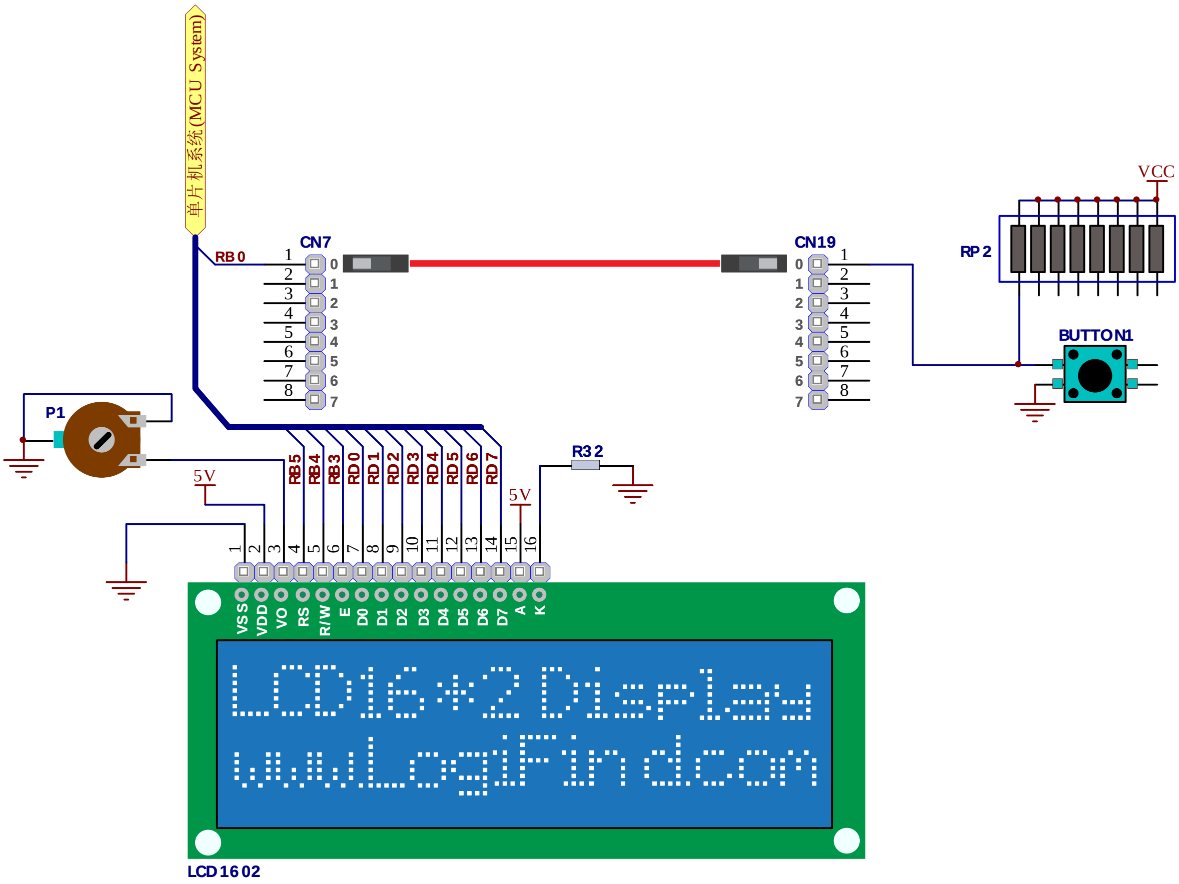
To understand the operation of Interrupt on RB0 pin.

c). Experimental Principle and Results

Press **BUTTON1**(**RB0**),the 4-digit segled will restart to display .

Lab34. Interrupt(RB0)_Button1_LCD1602

a).Hardware Connection


单片机系统(MCU System)
RB0
CN7
CN19
RP2
VCC
BUTTON1
P1
5V
R32
RB5 RB4 RB3 RD0 RD1 RD2 RD3 RD4 RD5 RD6 RD7
VSS VDD VO RS R/W E D0 D1 D2 D3 D4 D5 D6 D7 A K
LCD16*2 Display
www.LogiFind.com
LCD1602

b). OBJECTIVES

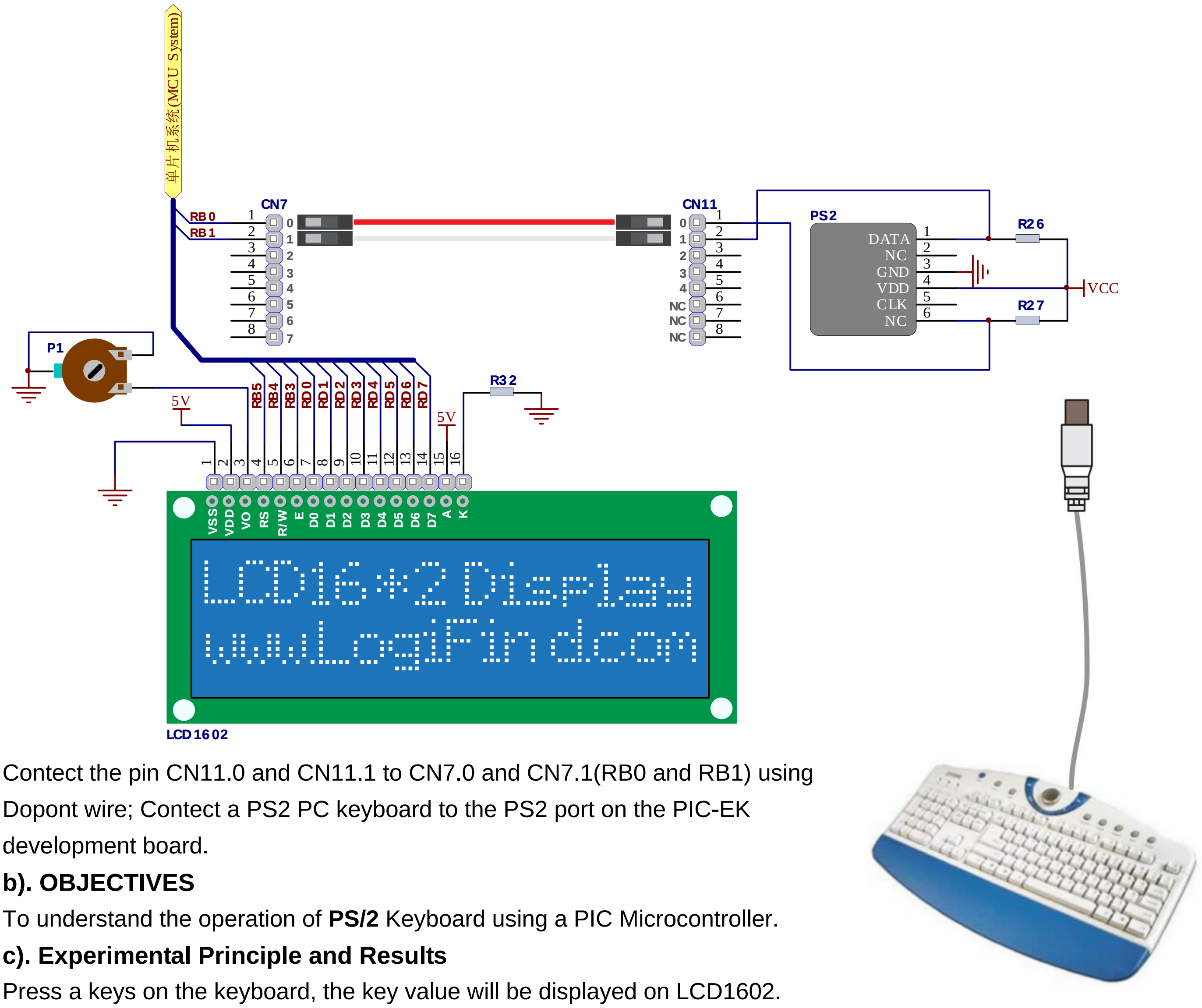
To understand the operation of Interrupt on RB0 pin.

c). Experimental Principle and Results

Press **BUTTON1**(**RB0**),the **LCD1602** will restart to display.

Lab35. INT_PS2_LCD1602

a).Hardware Connection



Contect the pin CN11.0 and CN11.1 to CN7.0 and CN7.1(RB0 and RB1) using Dopont wire; Contect a PS2 PC keyboard to the PS2 port on the PIC-EK development board.

b). OBJECTIVES

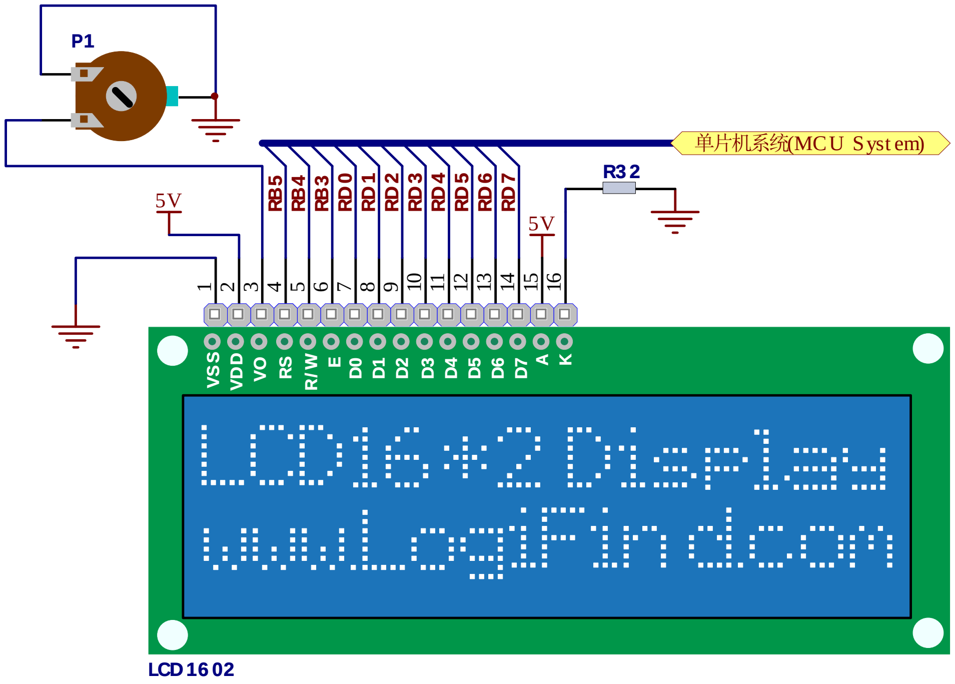
To understand the operation of **PS/2** Keyboard using a PIC Microcontroller.

c). Experimental Principle and Results

Press a keys on the keyboard, the key value will be displayed on LCD1602.

Lab36. Watchdog_LCD1602

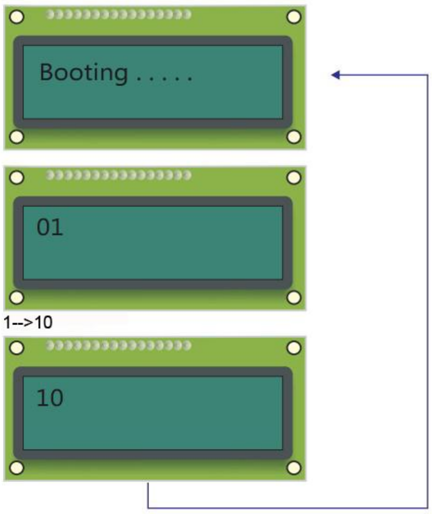
a).Hardware Connection



b. OBJECTIVES

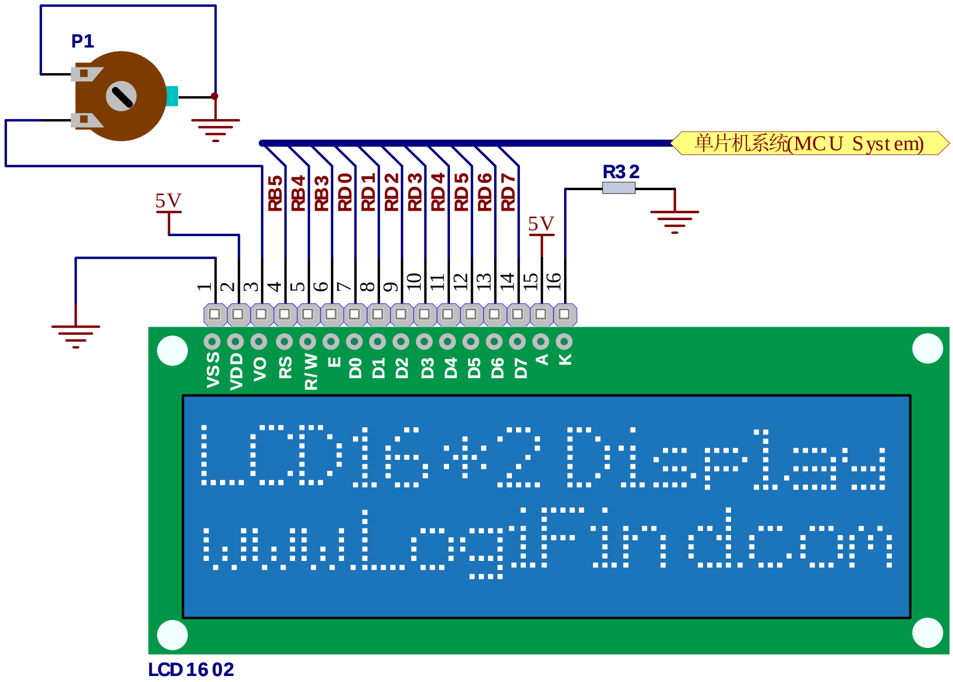
To understand the operation Watch Dog.

c). Experimental Principle and Results



Lab37. Internal_EEPROM_LCD1602

a).Hardware Connection

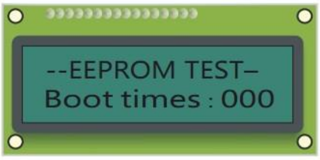


b. OBJECTIVES

To understand the operation of **internal eeprom**.

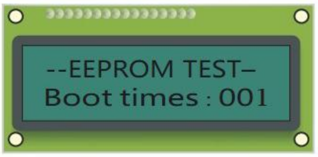
c). Experimental Principle and Results

Read the data X from address 0x10, X+1, write X+1 to 0x10,and display on LCD1602.



```
--EEPROM TEST–
Boot times : 000
```

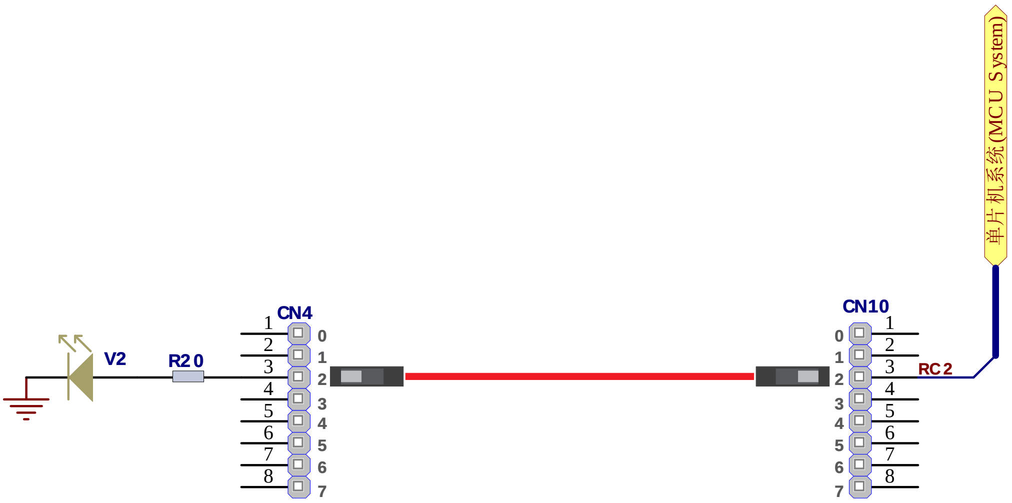
Reset or re-power the board.



```
--EEPROM TEST–
Boot times : 001
```

Lab38. PWM_CCP1_V2

a).Hardware Connection



b). OBJECTIVES

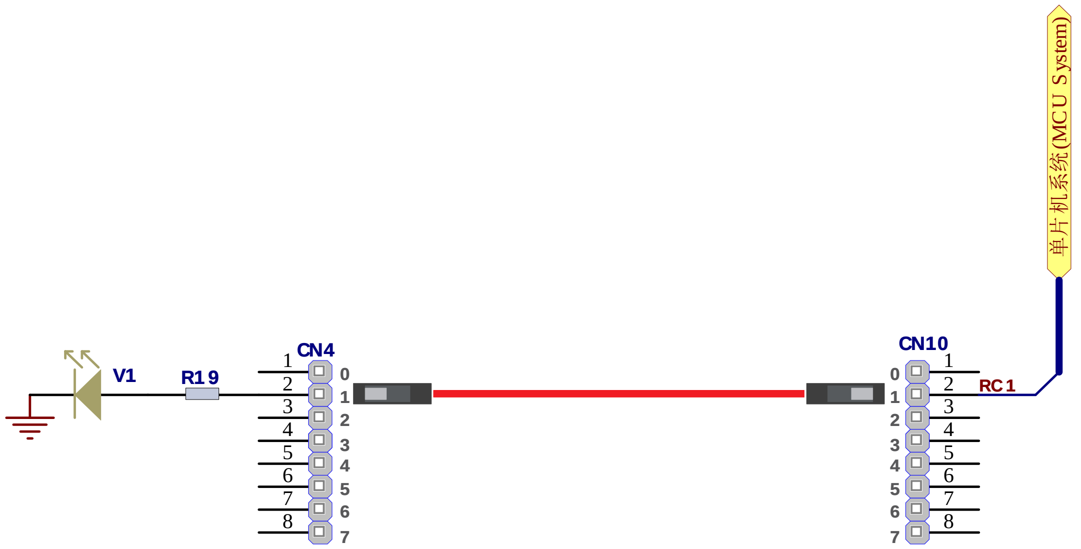
To understand the operation of **PWM.**

c). Experimental Principle and Results

Use different duty cycle to control the brightness of the LED.The **V2** will be activated and act from Dim to Bright…. Bright to Dim…..

Lab39. PWM_CCP2_V1

a).Hardware Connection



b). OBJECTIVES

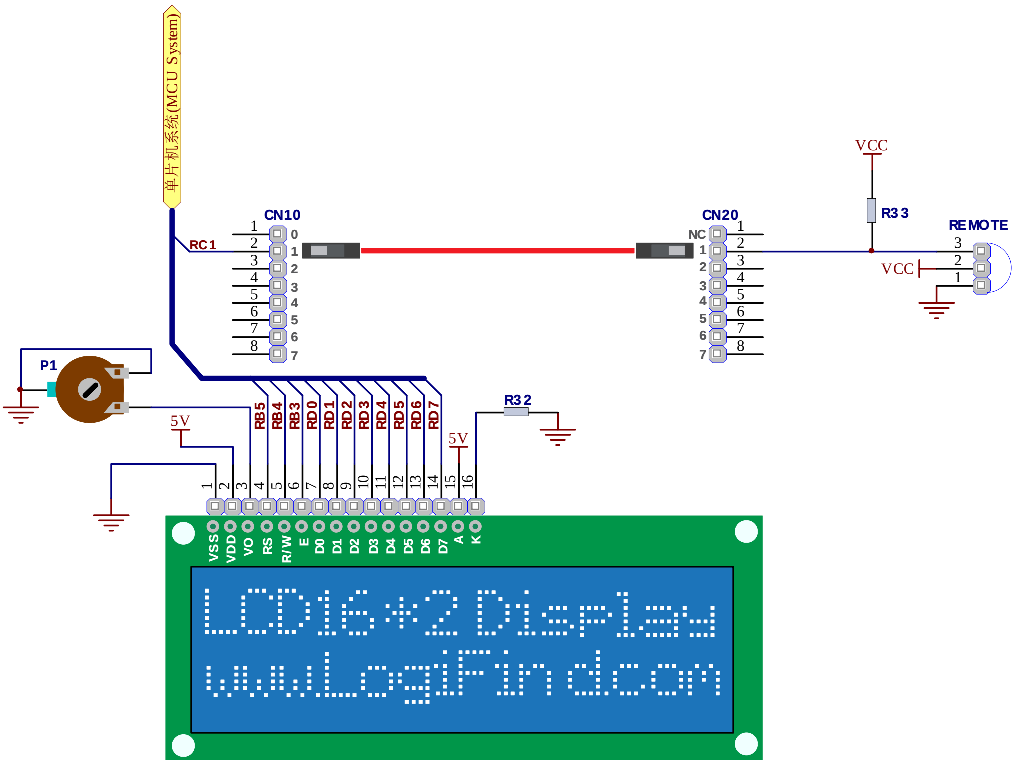
To understand the operation of **PWM**.

c). Experimental Principle and Results

Use different duty cycle to control the brightness of the LED.The **V1** will be activated and act from Dim to Bright…. Bright to Dim…..

Lab40. CCP2_IRD_LCD1602

a).Hardware Connection



LCD1602

b). OBJECTIVES

To understand the operation of CCP and infrared Remote control

c). Experimental Principle and Results

After pressed on the infrared remote controller, the code signal is received by the infrared receiver on **PIC-EK** board, trigger an interrupt, Decoding based on pulse width, send the decoding value to LCD1602.Press the key on the Remoter, the key value will be displayed on the LCD1602.

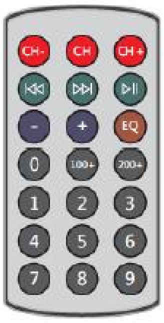
1) Power on the board:

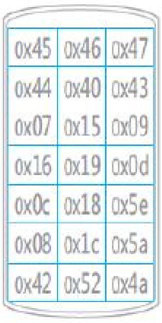
Press Remote con
trol

2) For example,Press“2”，“OX18”will be displayed:

CODE: 0X18

Table for Key value：

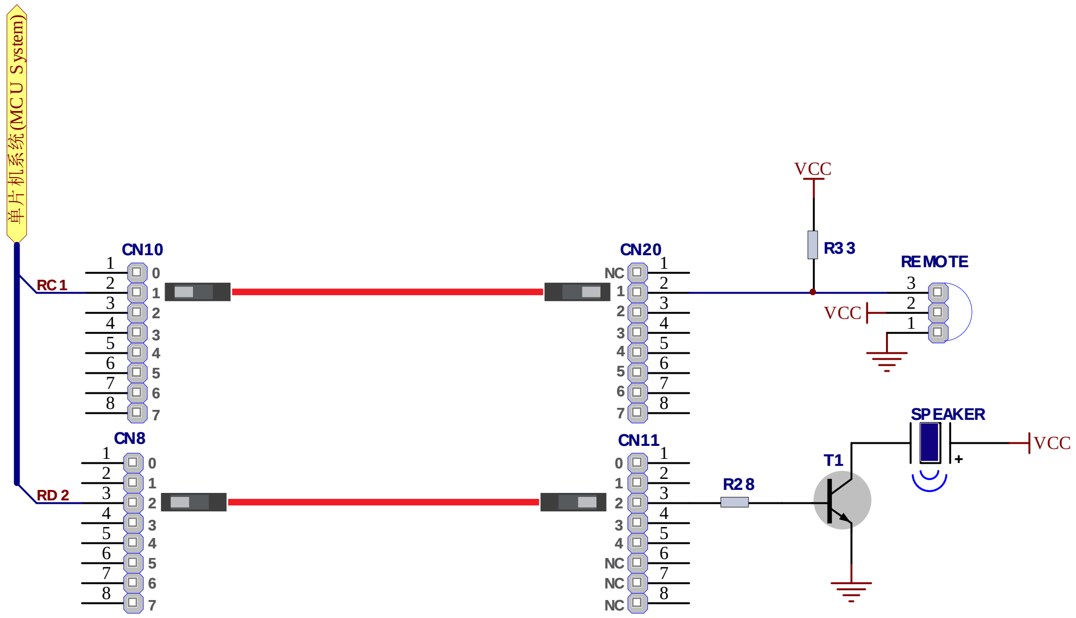




0x45	0x46	0x47
0x44	0x40	0x43
0x07	0x15	0x09
0x16	0x19	0x0d
0x0c	0x18	0x5e
0x08	0x1c	0x5a
0x42	0x52	0x4a

Lab41. CCP2_IRD_Beep

a).Hardware Connection



b. OBJECTIVES

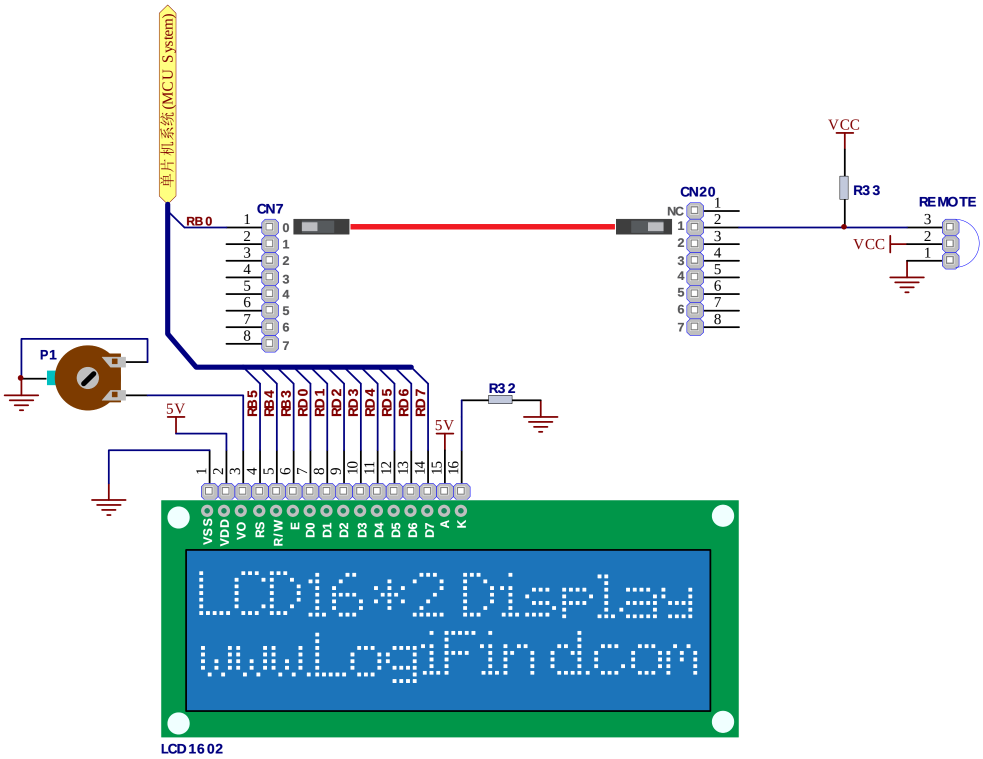
To understand the operation of infrared Remote control

c). Experimental Principle and Results

Press the any key on the Remoter, the speaker will be be activated.

Lab42. Interrupt(RB0)_IRD_LCD1602

a).Hardware Connection



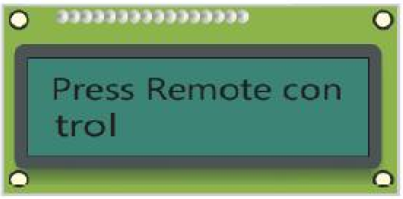
b). OBJECTIVES

To understand the operation of RB0 external Interrupt and infrared Remote control

c). Experimental Principle and Results

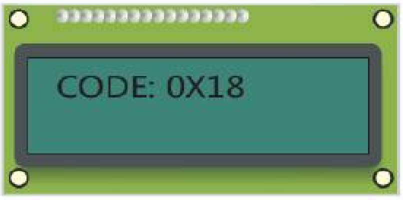
After pressed on the infrared remote controller, the code signal is received by the infrared receiver on **PIC-EK** board, trigger an interrupt, Decoding based on pulse width, send the decoding value to LCD1602.Press the key on the Remoter, the key value will be displayed on the LCD1602.

2) Power on the board:



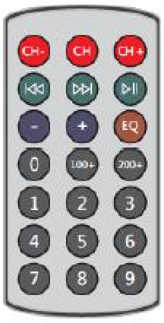
Press Remote con
trol

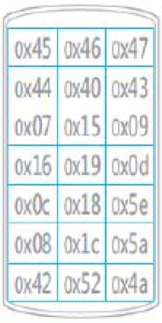
2) For example,Press“2”，“OX18”will be displayed:



CODE: 0X18

Table for Key value：

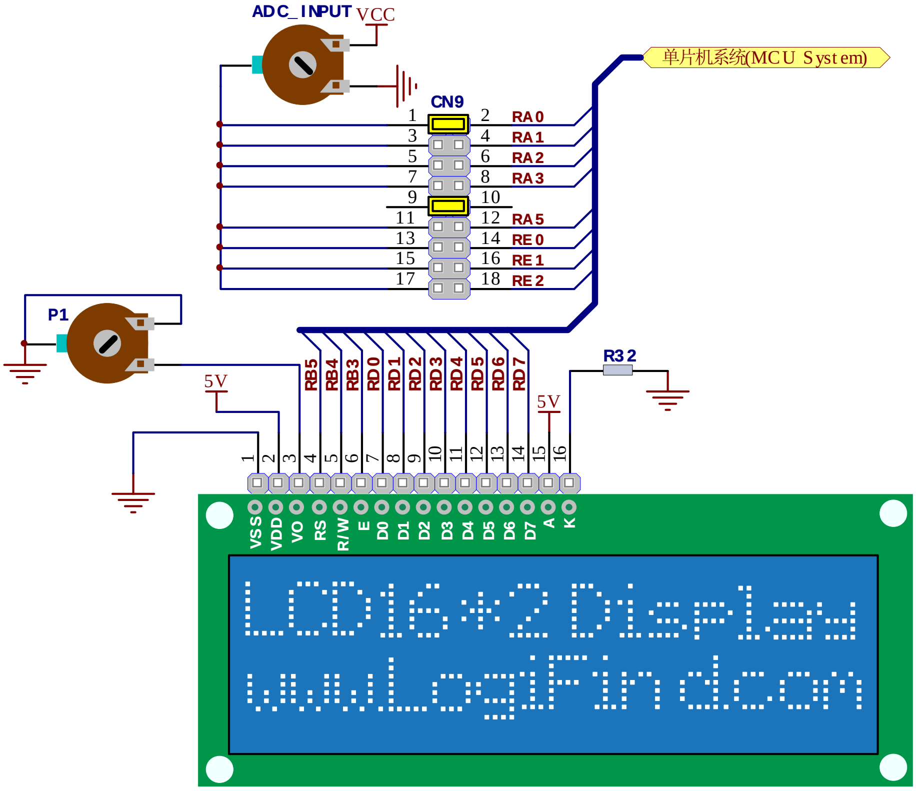




0x45	0x46	0x47
0x44	0x40	0x43
0x07	0x15	0x09
0x16	0x19	0x0d
0x0c	0x18	0x5e
0x08	0x1c	0x5a
0x42	0x52	0x4a

Lab43. AN0(8BIT)_LCD1602

a).Hardware Connection



LCD1602

b). OBJECTIVES

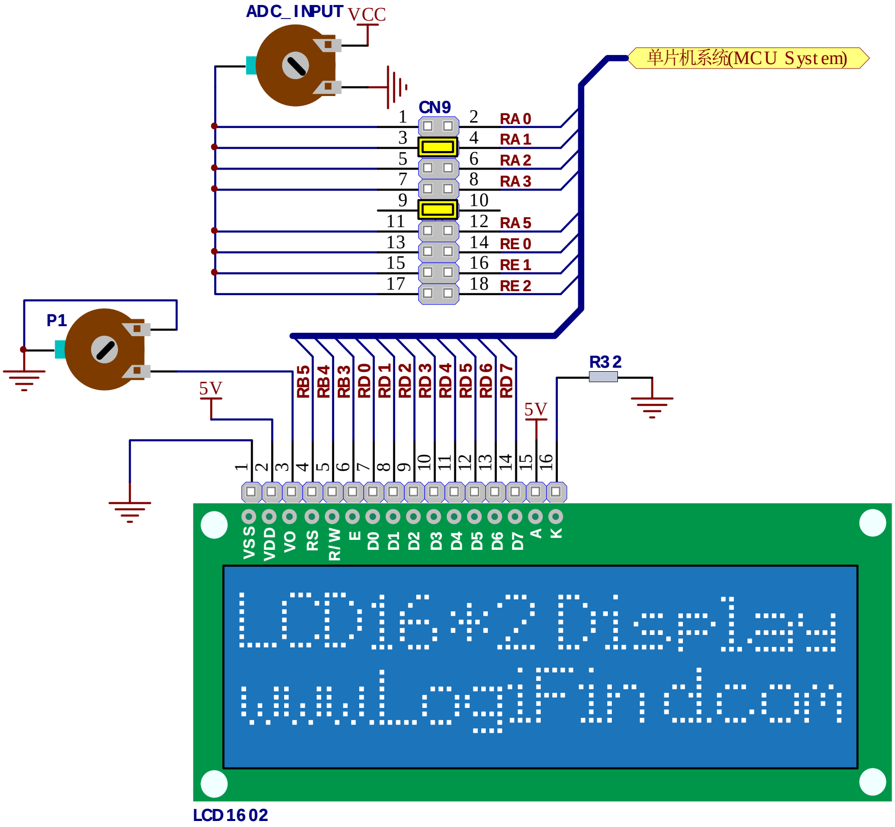
To understand the operation of 8-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 8-bit AD0 Sampling value will be displayed on the LCD1602,the range of 0-255.

Lab44. AN1(8BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

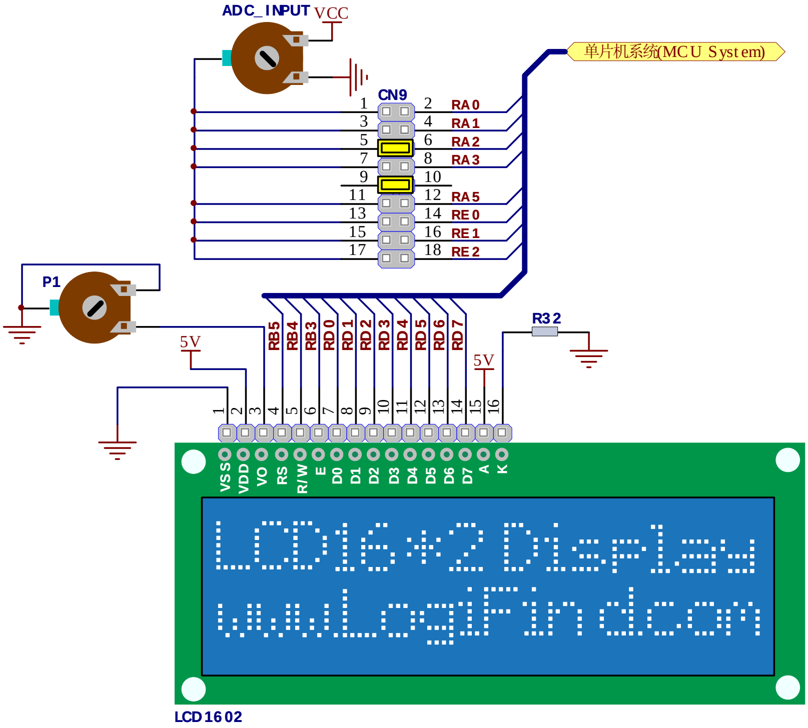
To understand the operation of 8-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 8-bit AD1 Sampling value will be displayed on the LCD1602,the range of 0-255.

Lab45. AN2(8BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

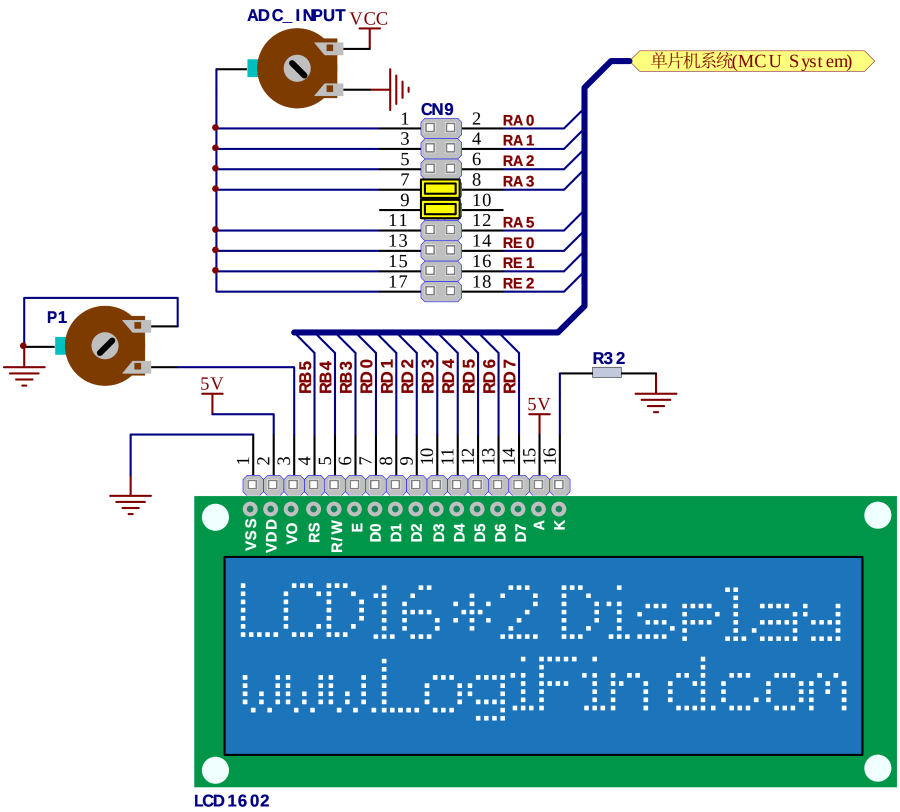
To understand the operation of 8-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 8-bit AD2 Sampling value will be displayed on the LCD1602,the range of 0-255.

Lab46. AN3(8BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

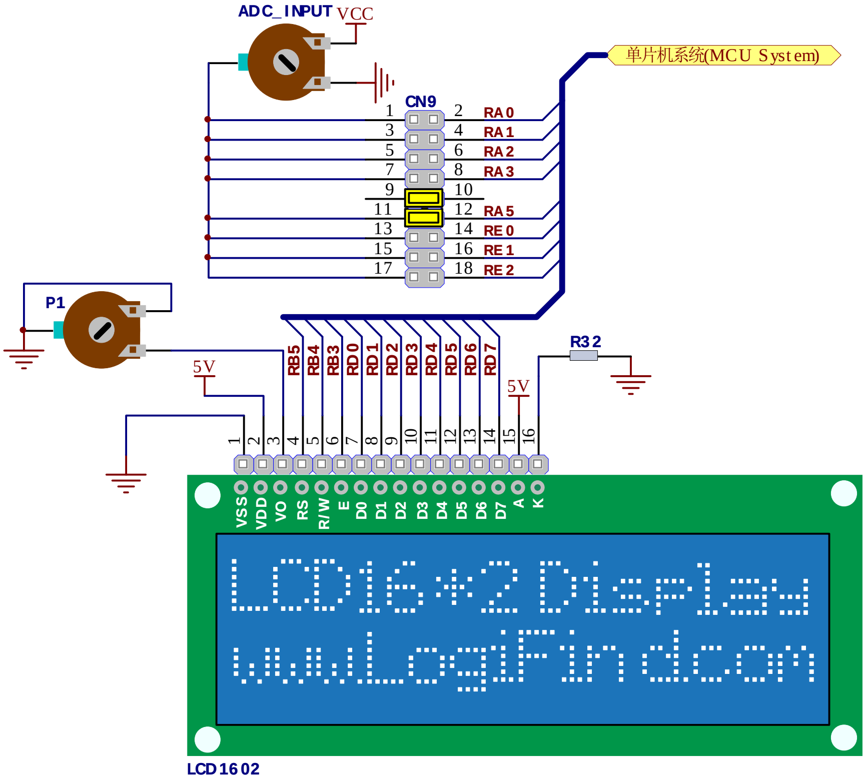
To understand the operation of 8-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 8-bit AD3 Sampling value will be displayed on the LCD1602,the range of 0-255.

Lab47. AN4(8BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

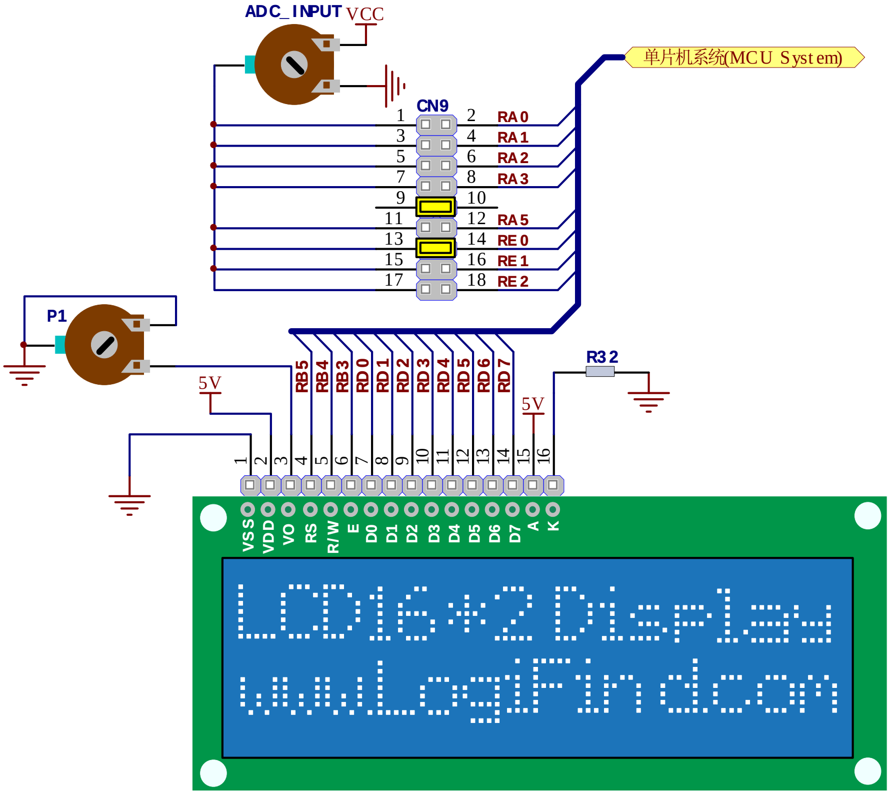
To understand the operation of 8-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 8-bit AD4 Sampling value will be displayed on the LCD1602,the range of 0-255.

Lab48. AN5(8BIT)_LCD1602

a).Hardware Connection



LCD1602

b). OBJECTIVES

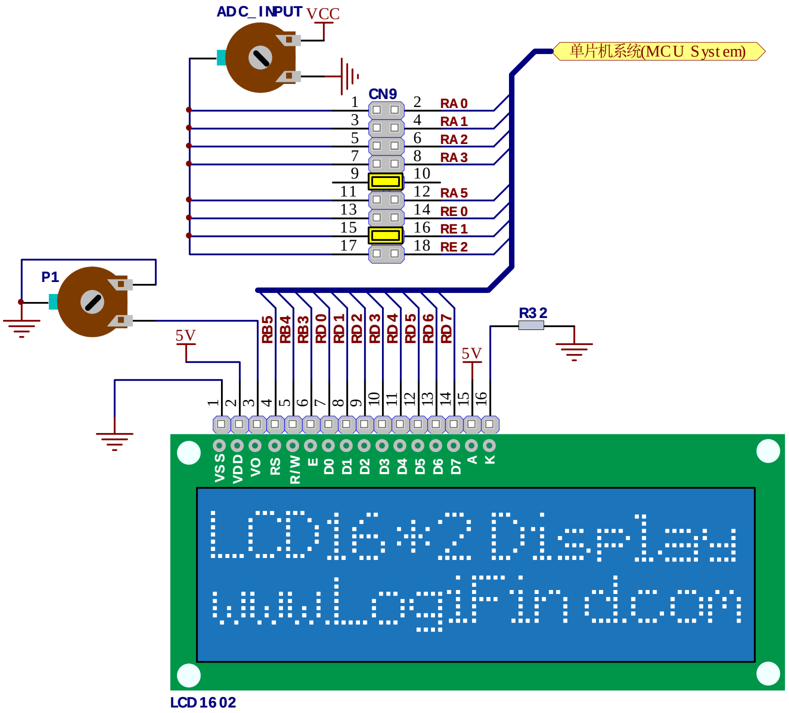
To understand the operation of 8-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 8-bit AD5 Sampling value will be displayed on the LCD1602,the range of 0-255.

Lab49. AN6(8BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

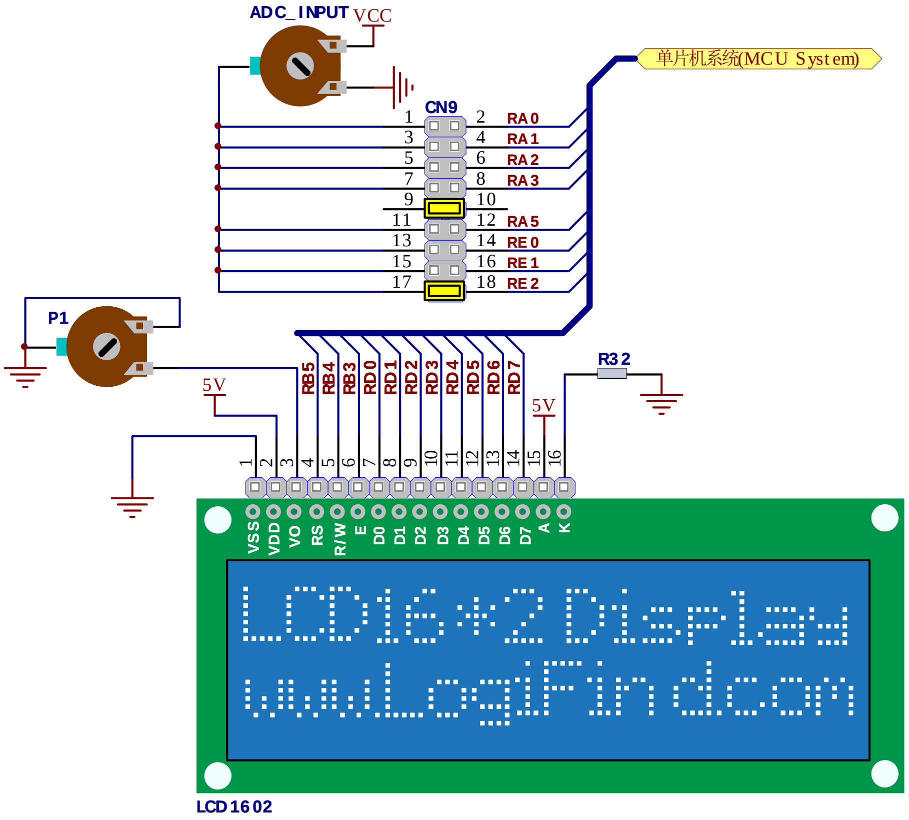
To understand the operation of 8-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 8-bit AD6 Sampling value will be displayed on the LCD1602,the range of 0-255.

Lab50. AN7(8BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

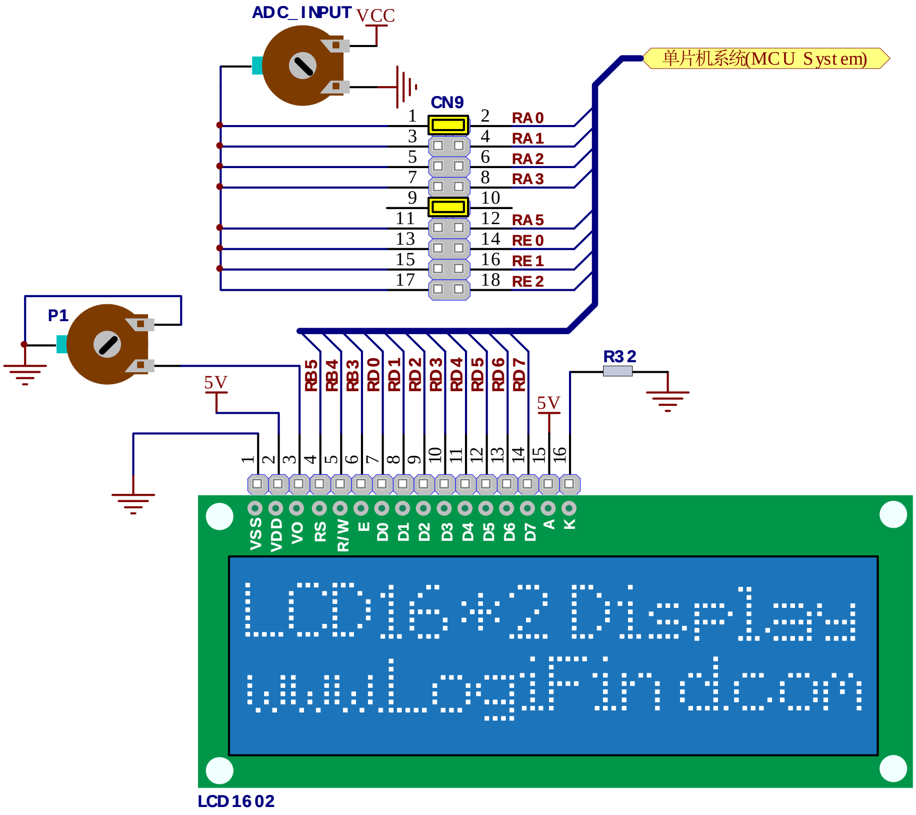
To understand the operation of 8-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 8-bit AD7 Sampling value will be displayed on the LCD1602,the range of 0-255.

Lab51. AN0(10BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

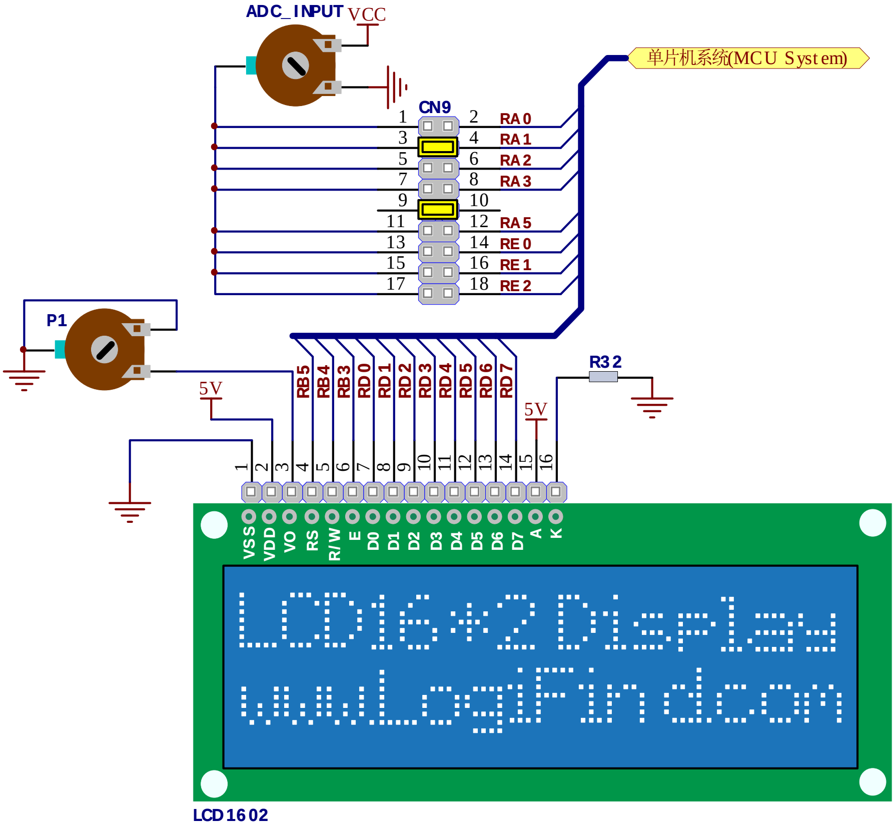
To understand the operation of 10-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 10-bit AD0 Sampling value will be displayed on the LCD1602,the range of 0-1023.

Lab52. AN1(10BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

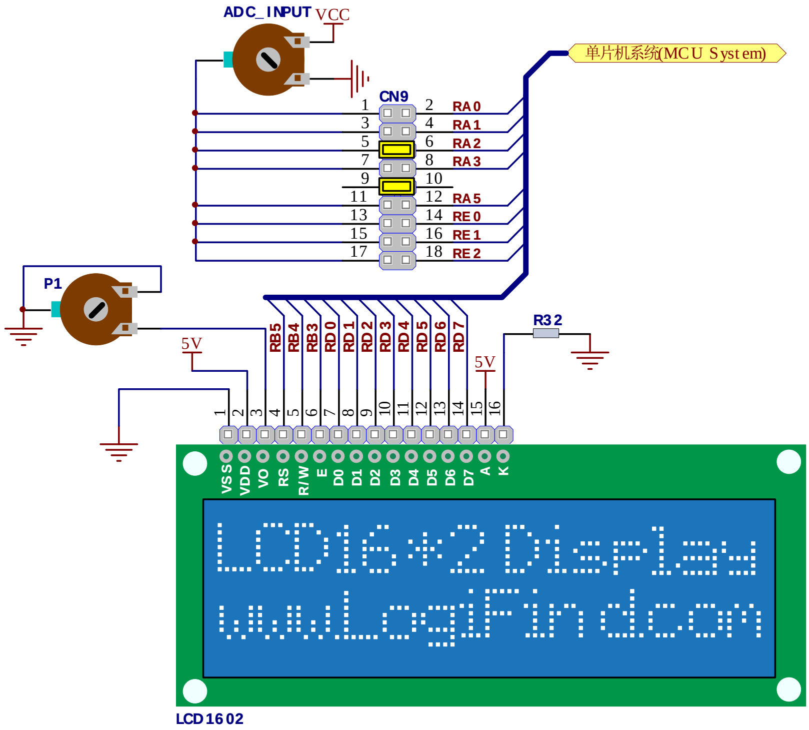
To understand the operation of 10-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 10-bit AD1 Sampling value will be displayed on the LCD1602,the range of 0-1023.

Lab53. AN2(10BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

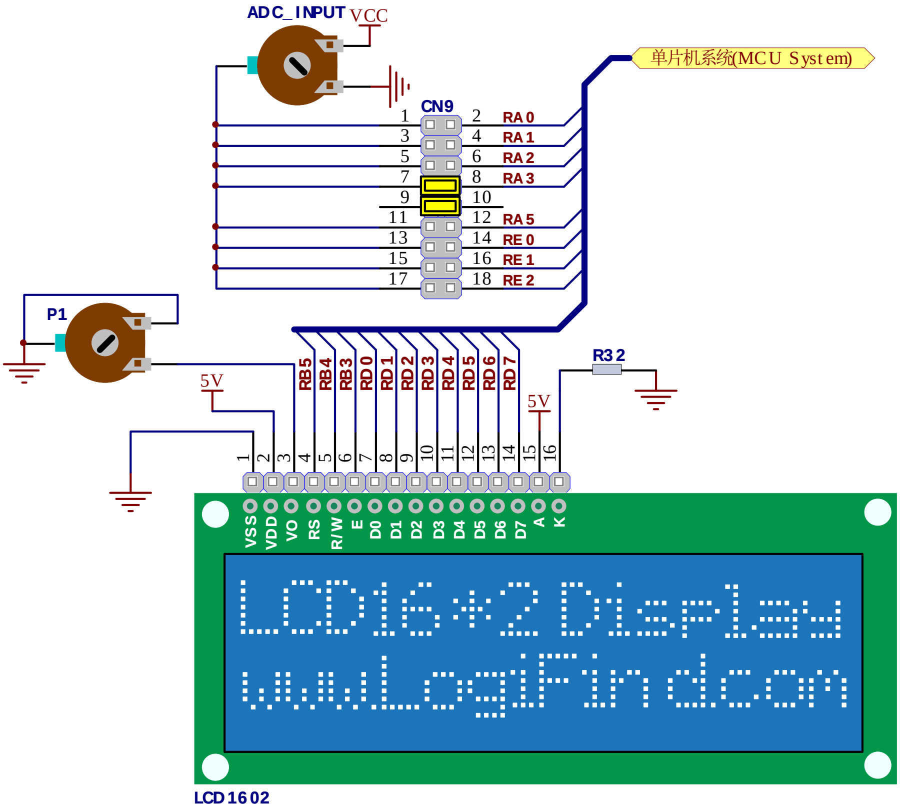
To understand the operation of 10-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 10-bit AD2 Sampling value will be displayed on the LCD1602,the range of 0-1023.

Lab54. AN3(10BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

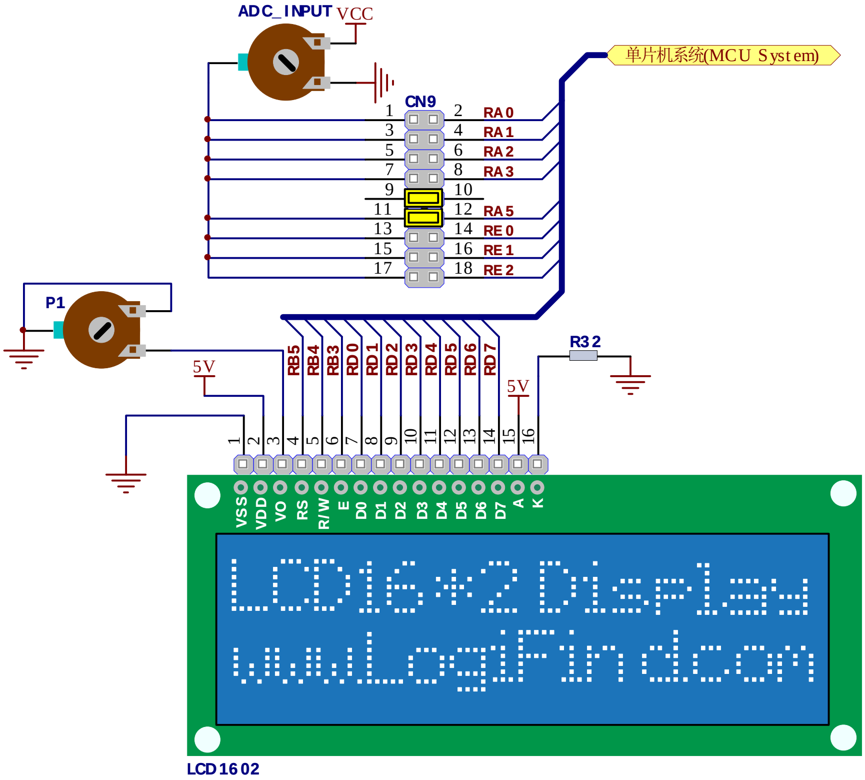
To understand the operation of 10-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 10-bit AD3 Sampling value will be displayed on the LCD1602,the range of 0-1023.

Lab55. AN4(10BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

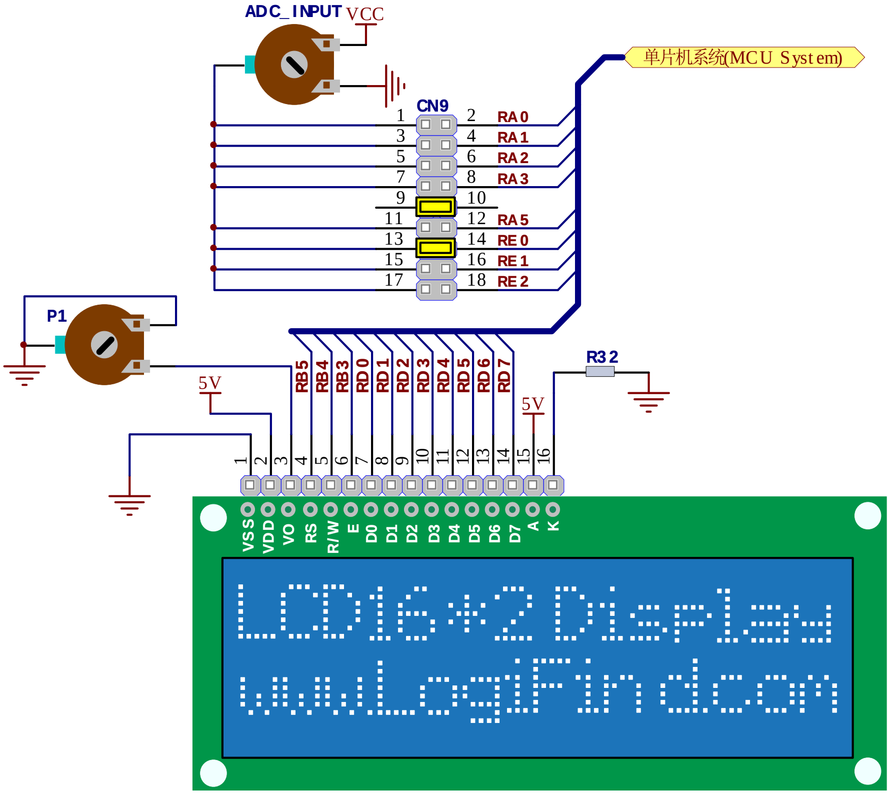
To understand the operation of 10-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 10-bit AD4 Sampling value will be displayed on the LCD1602,the range of 0-1023.

Lab56. AN5(10BIT)_LCD1602

a).Hardware Connection



LCD1602

b). OBJECTIVES

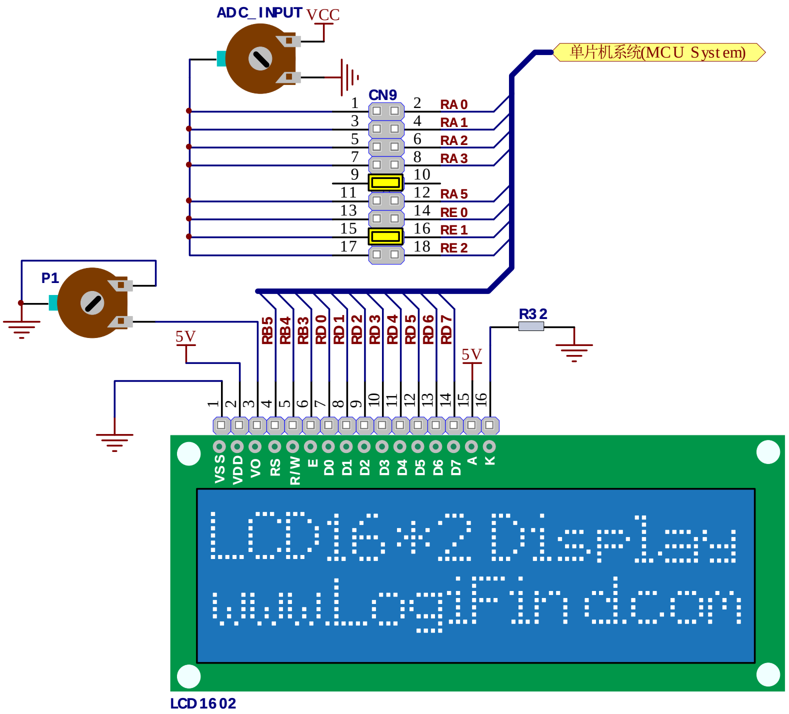
To understand the operation of 10-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 10-bit AD5 Sampling value will be displayed on the LCD1602,the range of 0-1023.

Lab57. AN6(10BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

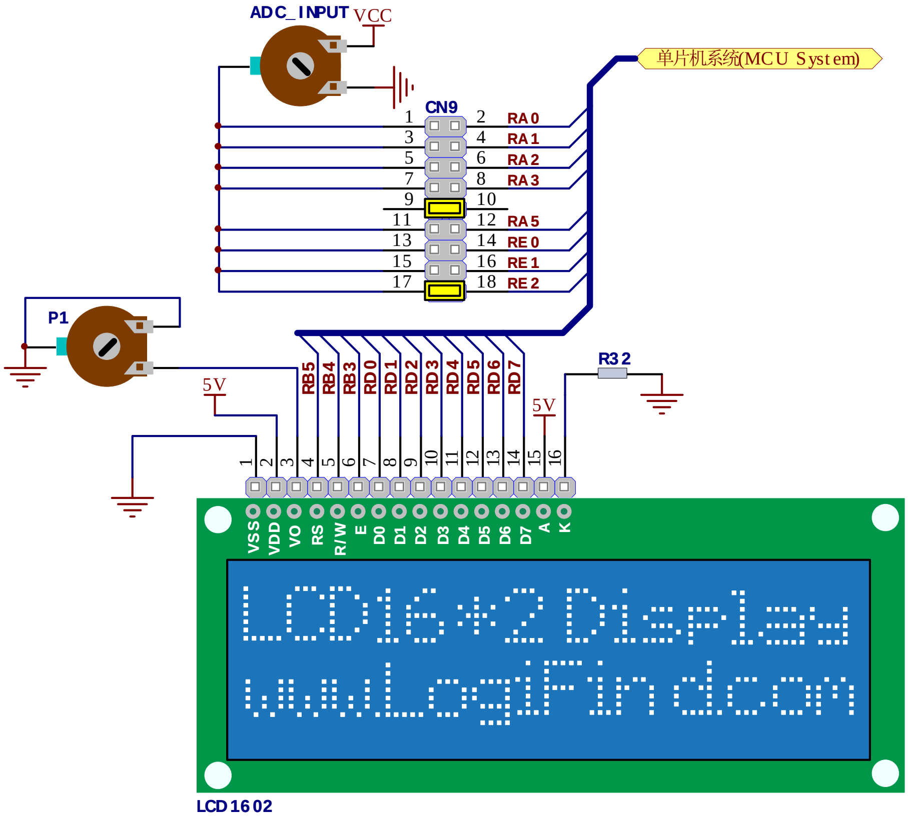
To understand the operation of 10-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 10-bit AD6 Sampling value will be displayed on the LCD1602,the range of 0-1023.

Lab58. AN7(10BIT)_LCD1602

a).Hardware Connection



b). OBJECTIVES

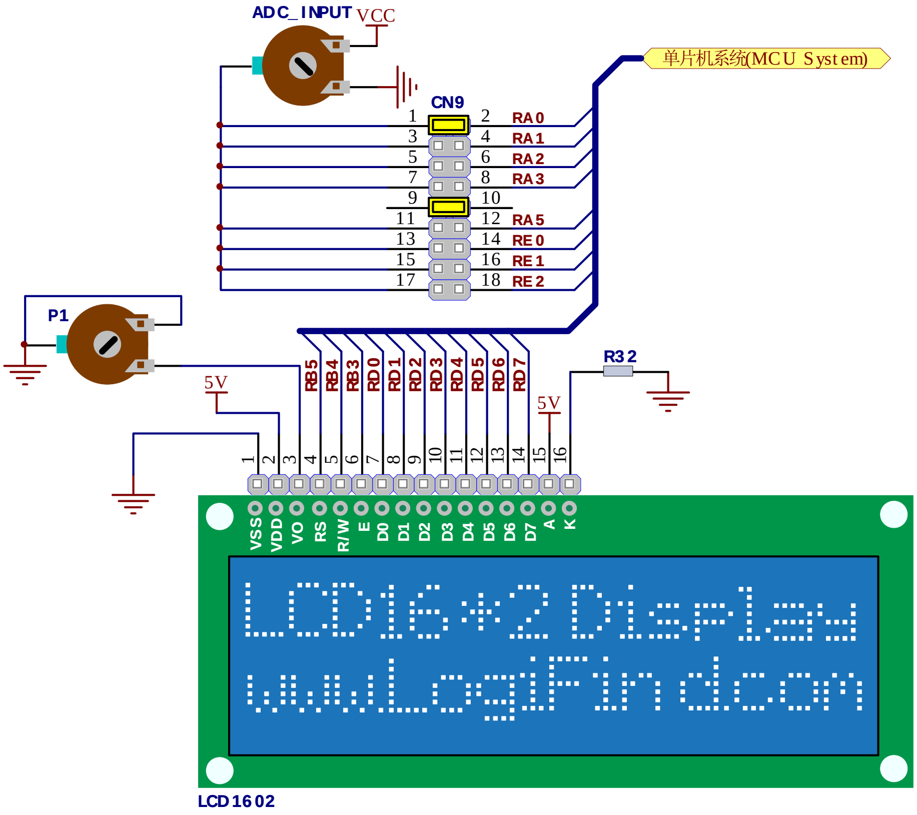
To understand the operation of 10-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 10-bit AD7 Sampling value will be displayed on the LCD1602,the range of 0-1023.

Lab59. AN0(8BIT)_Voltage_LCD1602

a).Hardware Connection



b). OBJECTIVES

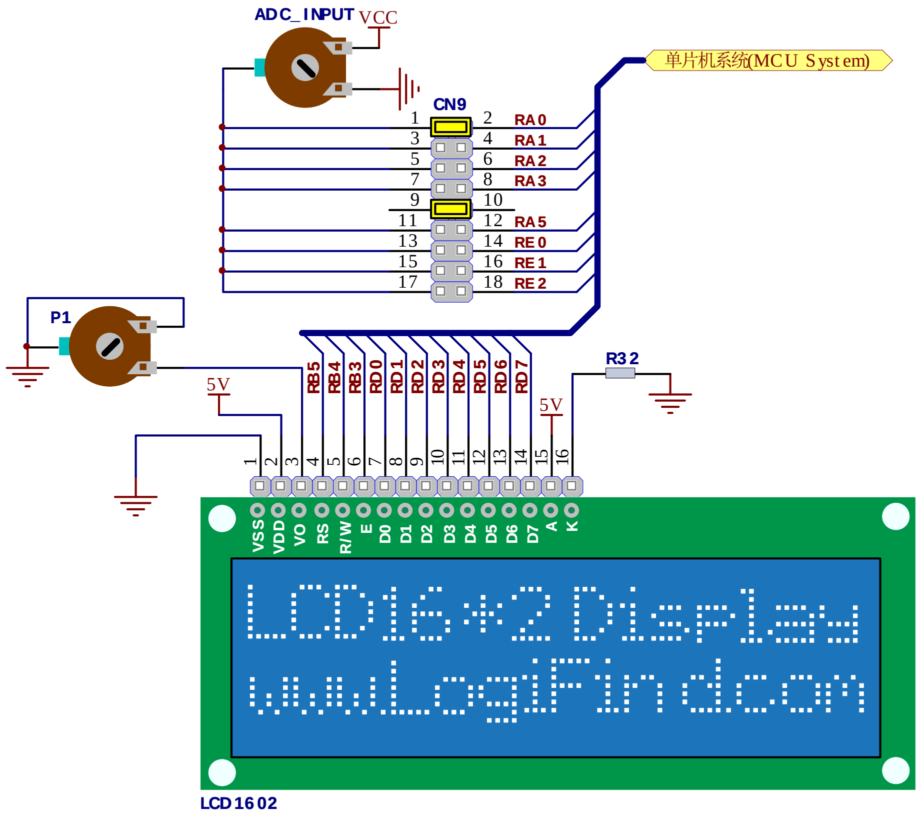
To understand the operation of 8-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 8-bit AD0 Sampling voltage will be displayed on the LCD1602.

Lab60. AN0(10BIT)_Voltage_LCD1602

a).Hardware Connection



b). OBJECTIVES

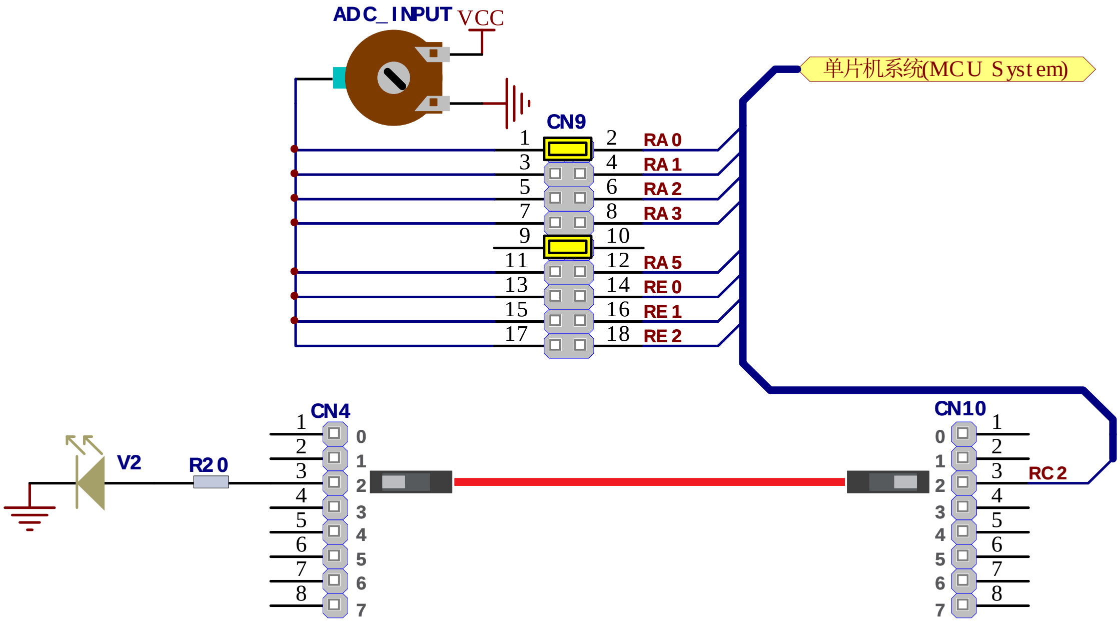
To understand the operation of 10-bit ADC.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, The 10-bit AD0 Sampling voltage will be displayed on the LCD1602.

Lab61. AN0_CCP1_V2

a).Hardware Connection



b). OBJECTIVES

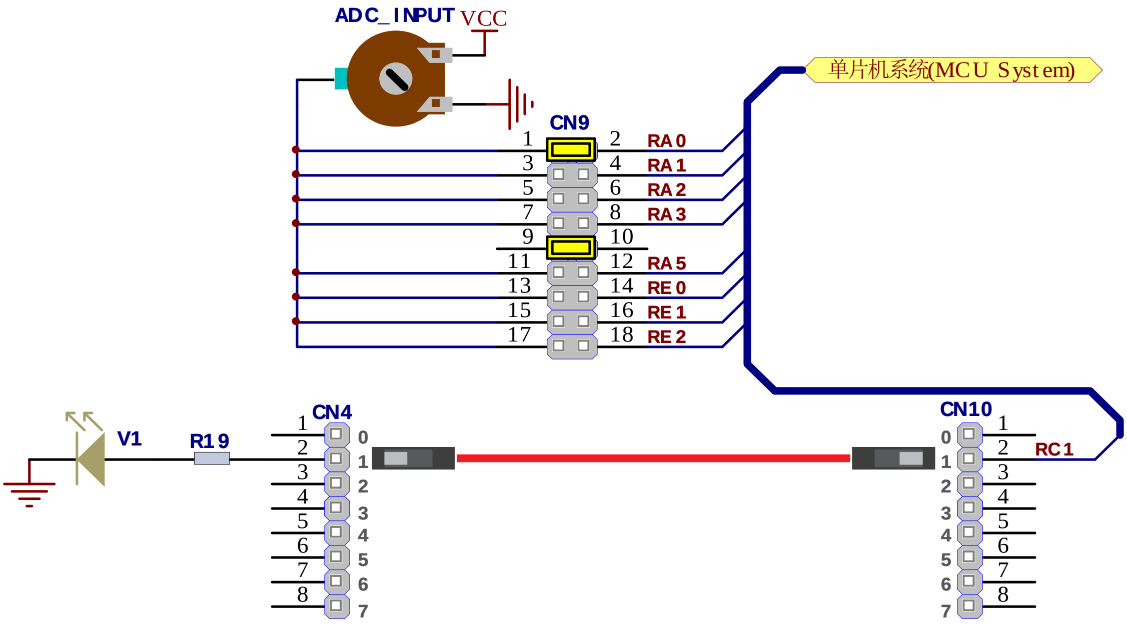
To understand the operation of CCP1 and AD0.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, the LED brightness V2 will changing.

Lab62. AN0_CCP2_V1

a).Hardware Connection



b). OBJECTIVES

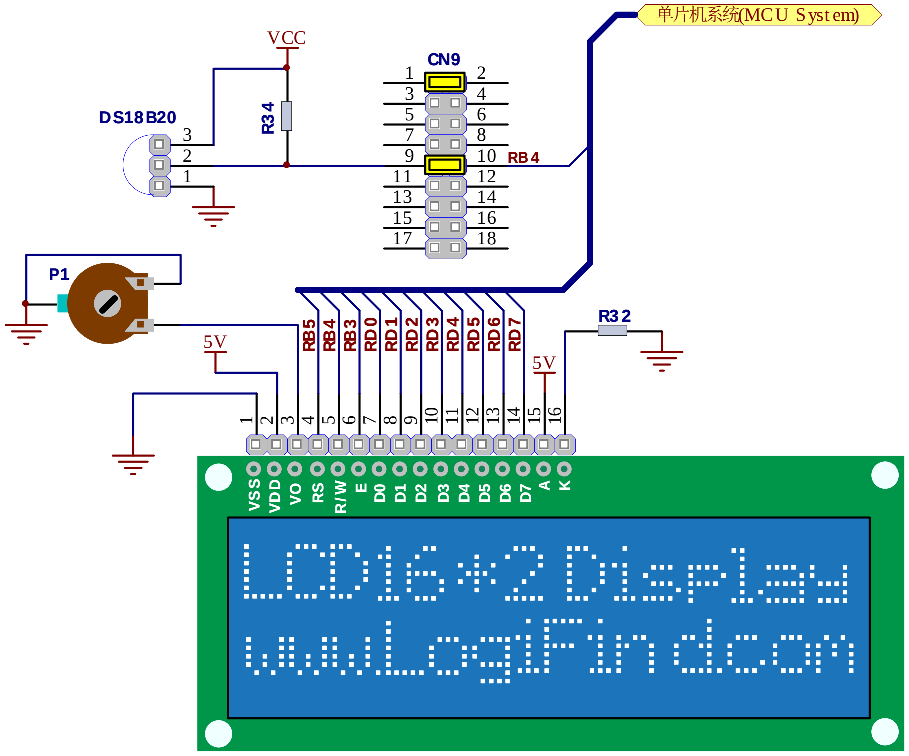
To understand the operation of CCP2 and AD0.

c). Experimental Principle and Results

Adjust the potentiometer **ADC_INPUT**, the LED brightness V1 will changing.

Lab63. DS18B20_LCD1602

a).Hardware Connection



LCD1602

b). OBJECTIVES

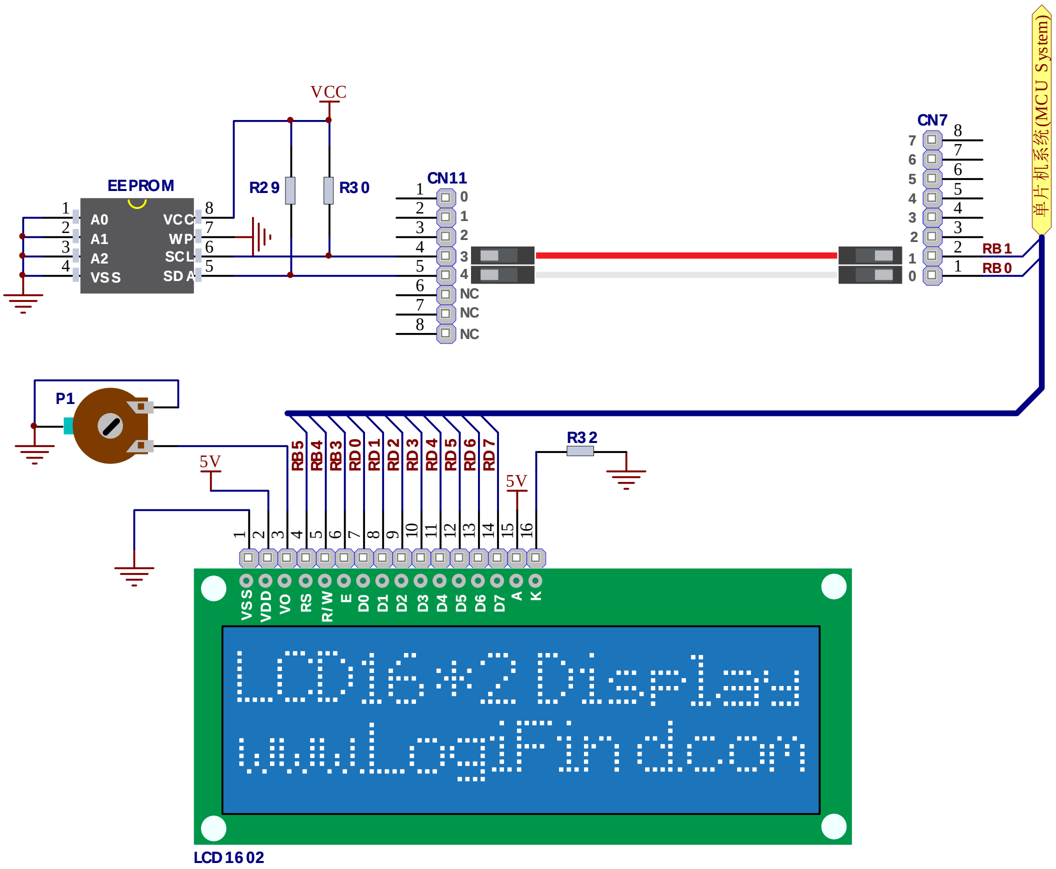
To understand the operation of 1-wire device DS18B20.

c). Experimental Principle and Results

The current environment’s temperature value will be displayed on the LCD1602.

Lab64. IIC_24CXX_LCD1602

a).Hardware Connection



b). OBJECTIVES

To understand the operation of external eeprom 24Cxx.

c). Experimental Principle and Results

Press the “reset” KEY, the external eeprom will record the number of times.

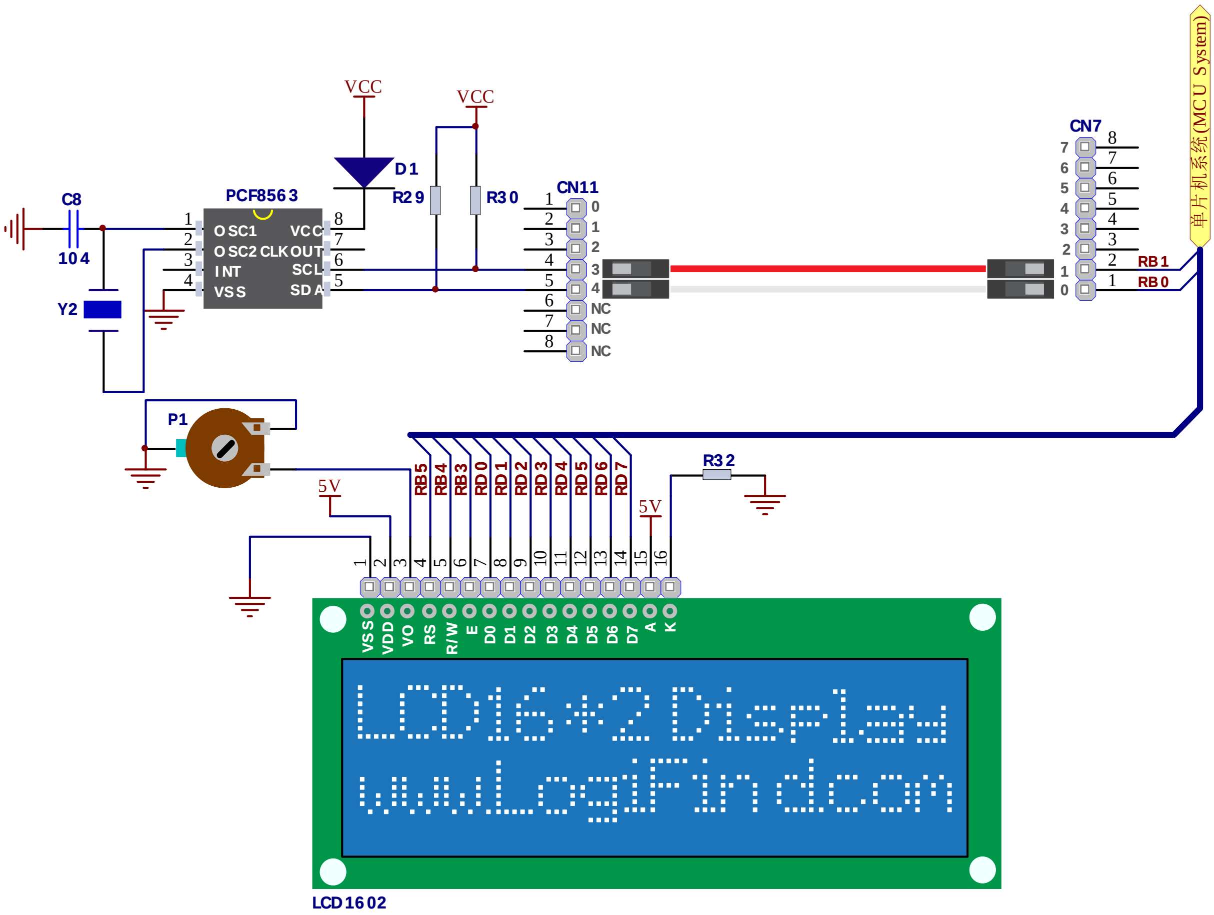
```
--EEPROM TEST–
Boot times : 000
```

Press Reset Button or re-power up the board…..

```
--EEPROM TEST–
Boot times : 001
```

Lab65. RTC_PCF8563_LCD1602

a).Hardware Connection

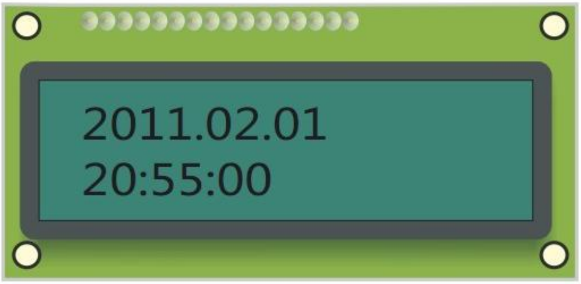


b). OBJECTIVES

To understand the operation of PCF8563 using IIC.

c). Experimental Principle and Results

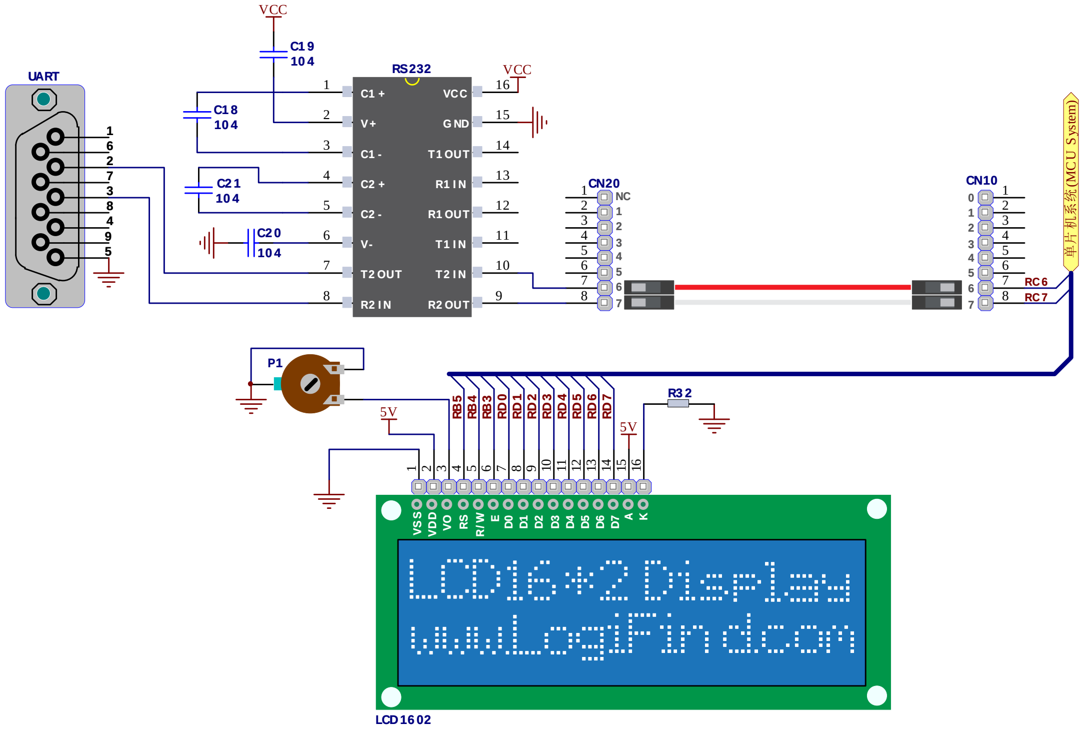
The RTC will will be displayed on the LCD1602.



2011.02.01
20:55:00

Lab66. UART_LCD1602

a).Hardware Connection



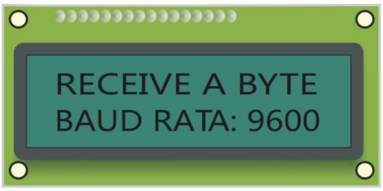
b). OBJECTIVES

To understand the operation of RS232.

c). Experimental Principle and Results

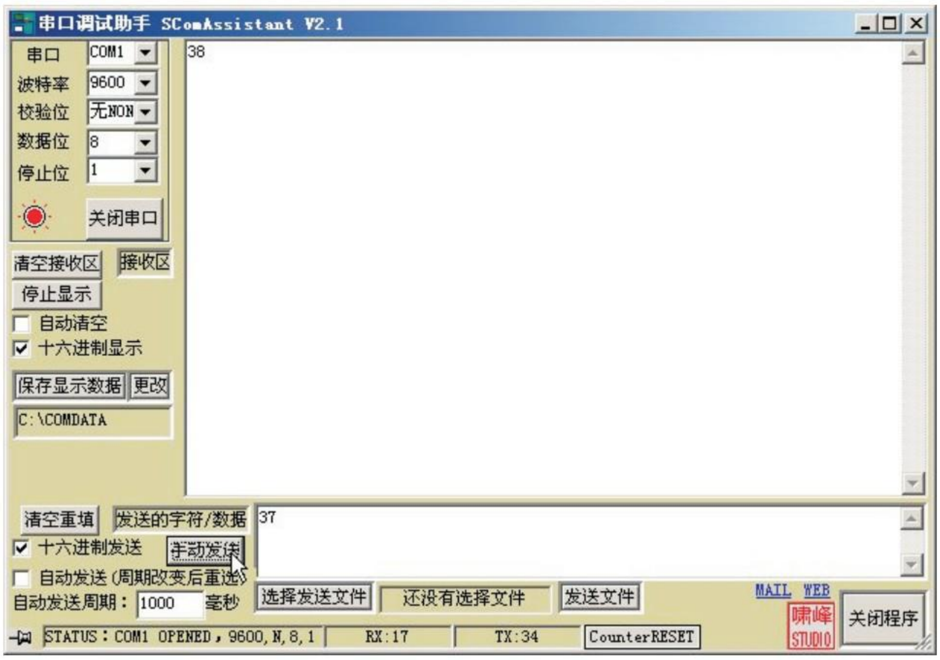
Send a data X from PC, the MCU will receive it and return X+1 to PC. All data will be displayed on the LCD1602.

1）Start to run, Baud Rate will be displayed.



RECEIVE A BYTE
BAUD RATA: 9600

2）On SComAssistant, “37” is data for sending, “38” is data for receiving.



串口 COM1
波特率 9600
校验位 无NON
数据位 8
停止位 1

38

□ 自动清空
☑ 十六进制显示

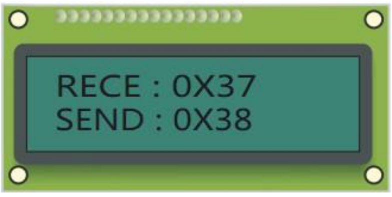
C:\COMDATA

37

☑ 十六进制发送
□ 自动发送（周期改变后重选）
自动发送周期：1000 毫秒

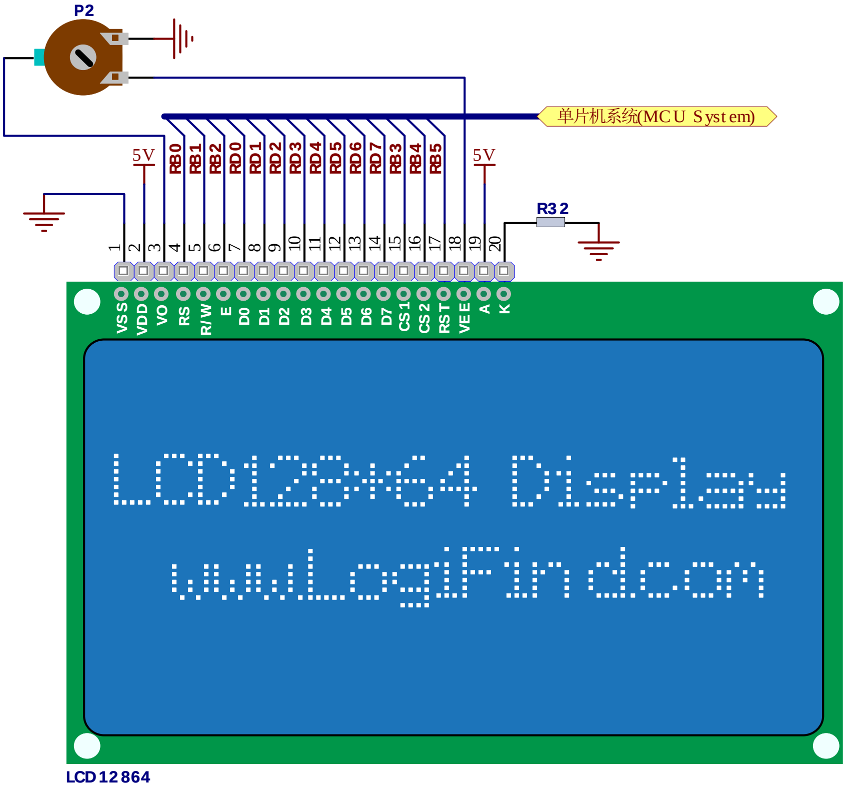
STATUS：COM1 OPENED，9600，N，8，1 RX:17 TX:34

3）Sent data “0x37” and received data “0x 38” will be displayed on LCD1602.



Lab67. LCD12864_ST7920

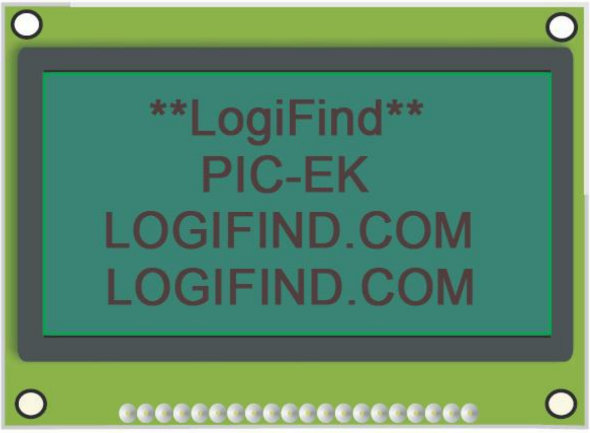
a).Hardware Connection



b). OBJECTIVES

To understand the operation of LCD12864.

c). Experimental Principle and Results



Contact Us

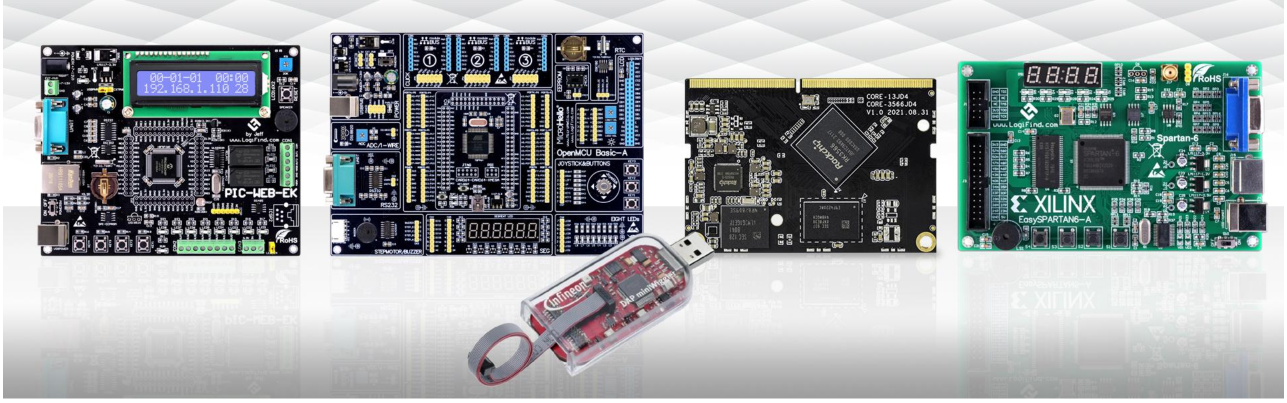
If you want to learn more about our products, please visit our website at:

www.MicroHello.com

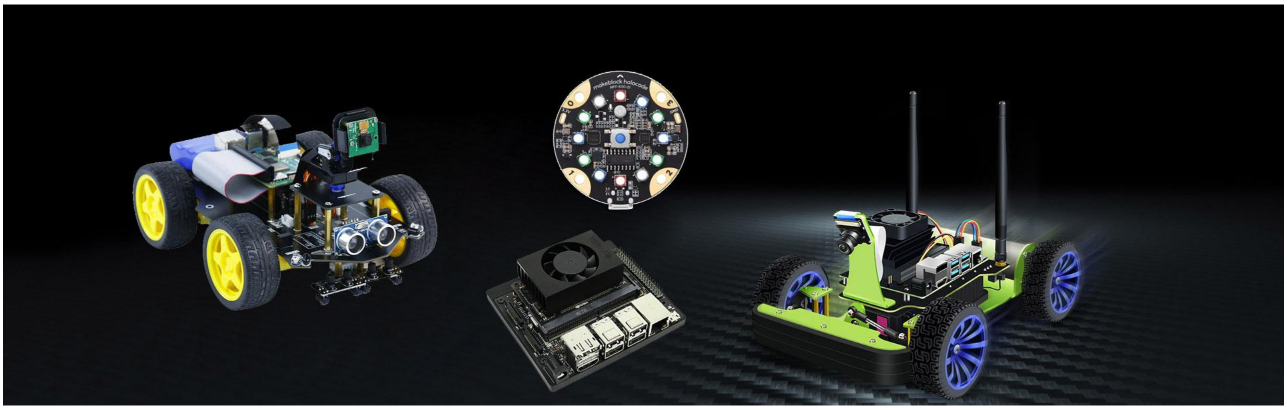
e-mail(Order)	microhello_sales@163.com
e-mail(Tech Support)	microhello_support@163.com
Skype(Order/Tech Support)	love100mhz
WhatsApp(Order/Tech Support)	(+86)15978193886
Wechat(微信) (Order/Tech Support)	(+86)15978193886
Mobile(Order/Tech Support)	(+86)15978193886

Other products on www.MicroHello.com

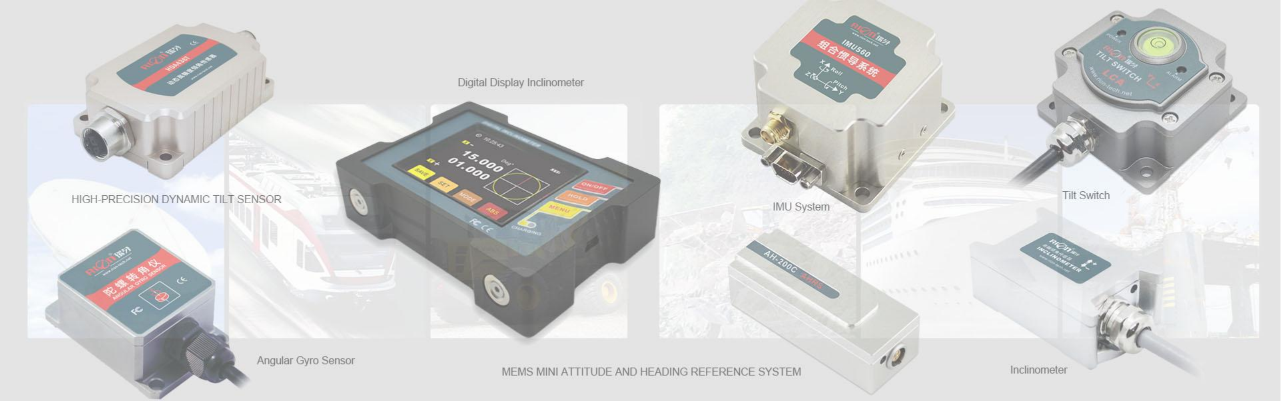
1. FPGA/CPLD/MCU Development Tools: Raspberry Pi; Arduino; MCU development boards and Programmers/debuggers of FPGA/CPLD, AVR, PIC, PIC32, PIC24, dsPIC, MSP430, NXP, C8051F, 51, STM32, STM8, LPC and ESP32.



2.AI and Robots: Jetson AI development board,AI camera,AI Robot,MCU Robot, Raspberry Robot, Arduino Robot, Robot/AI accessories.



3. Attitude Solutions: Inclinometer，Dynamic Inclinometer，Tilt Switch，Dynamic Tilt Switch，Digital Compass，Accelerometer，Digital Display Inclinometer，GYRO，AHRS，North Finder，Inertial Navigation System，Photovoltaic Tilt Sensor.


Digital Display Inclinometer
HIGH-PRECISION DYNAMIC TILT SENSOR
IMU System
Tilt Switch
Angular Gyro Sensor
MEMS MINI ATTITUDE AND HEADING REFERENCE SYSTEM
Inclinometer

4. CAN Bus Series: CAN Analyzer, USBCAN-OBD, Ethernet/Modbus TCP/Modbus RTU/WIFI to CAN, CANopen to UART, CANopen slave/master Converter, Programmable Smart Gateway Series, CANopen IO Coupler, Modbus IO Coupler.

5. Displays: Character LCD display, Graphic LCD display, Color LCD display(TFT), Segment LCD display, Segment LED display, Black & White ePaper, Multi-Color ePaper, Full Color ePaper.

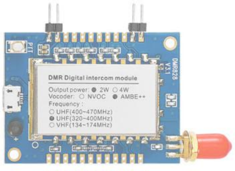
6. Sensors and Sensor Modules: Pressure/Force Sensor,Distance Measuring Sensor, Acceleration Sensor, Micro Flow/Air Flow Sensor, Dust Sensor, Temperature and Humidity Sensor, Transmissive Sensor, Hall-effect Rotary Position Sensor, Air Quality Sensor,Gas sensor(for CO2，CH4，O2，NH3，H2S，NO2，O3，SO2，CL2，HCL，HF，PH3，ETO，H2，HCHO，C2H5OH，C3H8，NO，CLO2，HCN or CH3SH，etc.)

7. Wireless Modules: RF Module，LoRa Module，FSK Front-End RF Module，Sensor Acquisition

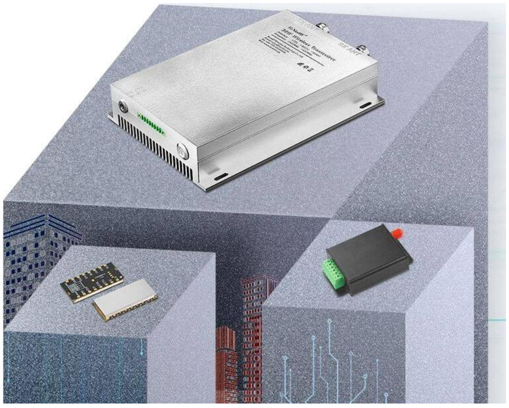
Gateway and Node，Data Transceiver Module，Walkie Talkie Module，Wireless Switch Module，Wireless Audio Module，GPS Module，SDR，LoRaWan Gateways and Nodes，ASK & Superheterodyne Module.

LoRa module

Wireless Switch Module


Digital Walkie Talkie Module

Wireless Data Transceiver Module



DISCLAIMER

All the products owned by www.MicroHello.com are protected by copyright law and international copyright treaty. Therefore, this manual is to be treated as any other copyright material. No part of this manual, including product and software described herein, must be reproduced, stored in a retrieval system, translated or transmitted in any form or by any means, without the prior written permission of www.MicroHello.com. The manual PDF edition can be printed for private or local use, but not for distribution. Any modification of this manual is prohibited.

www.MicroHello.com provides this manual 'as is' without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties or conditions of merchantability or fitness for a particular purpose.

www.MicroHello.com shall assume no responsibility or liability for any errors, omissions and inaccuracies that may appear in this manual. In no event shall www.MicroHello.com, its directors, officers, employees or distributors be liable for any indirect, specific, incidental or consequential damages (including damages for loss of business profits and business information, business interruption or any other pecuniary loss) arising out of the use of this manual or product, even if www.MicroHello.com has been advised of the possibility of such damages.

www.MicroHello.com reserves the right to change information contained in this manual at any time without prior notice, if necessary.

HIGH RISK ACTIVITIES

The products of www.MicroHello.com are not fault – tolerant nor designed, manufactured or intended for use or resale as on – line control equipment in hazardous environments requiring fail – safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of Software could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). www.MicroHello.com and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

TRADEMARKS

The www.MicroHello.com name and logo are trademarks of www.MicroHello.com. All other trademarks mentioned herein are property of their respective companies.All other product and corporate names appearing in this manual may or may not be registered trademarks or copyrights of their respective companies, and are only used for identification or explanation and to the owners' benefit, with no intent to infringe.