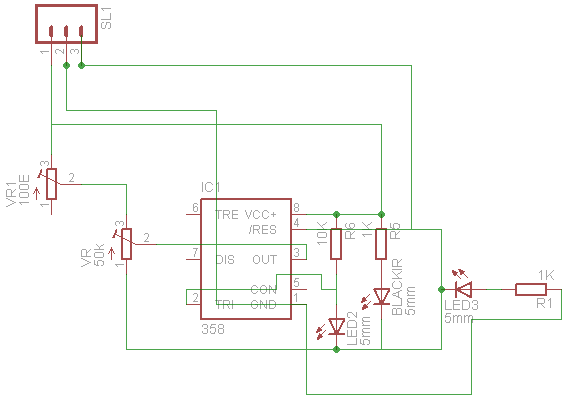
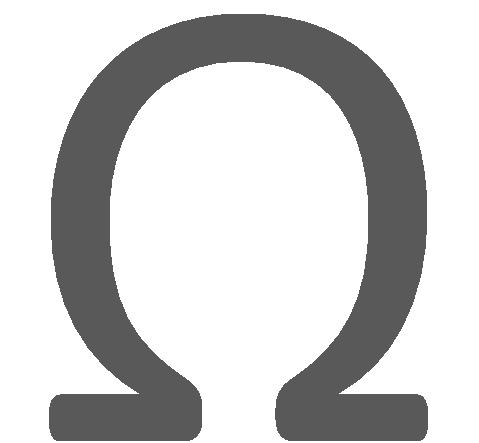
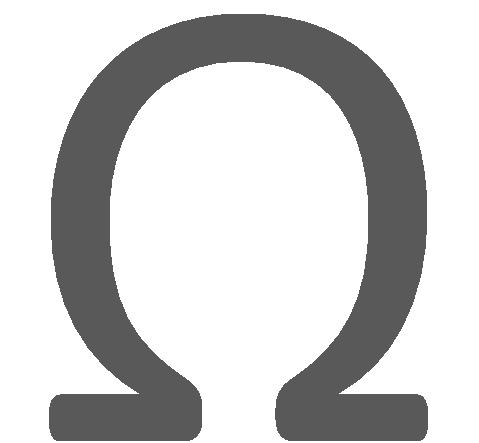
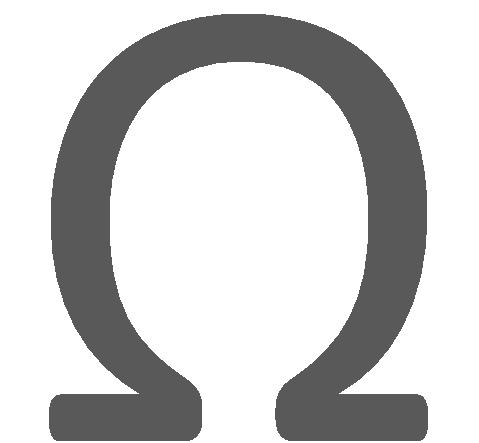
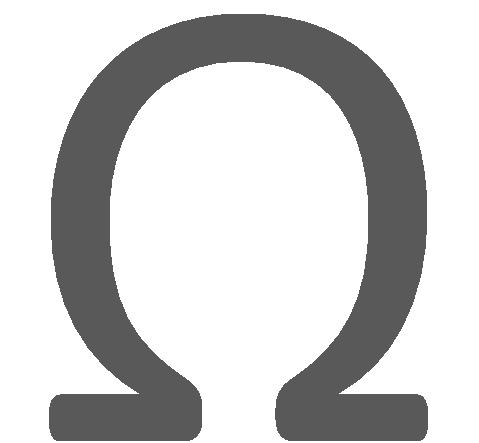
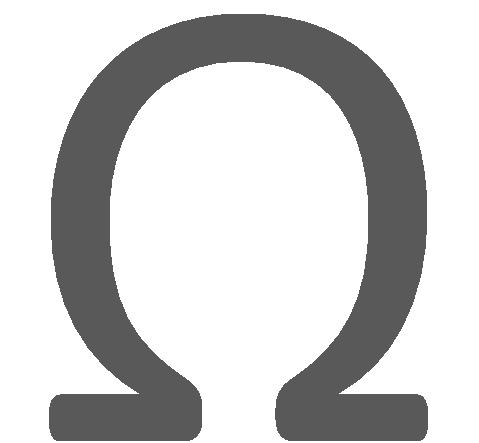


**IR MODULE**

The transmitter part of the sensor project is an [Infrared (IR) Led](http://engineersgarage.com/content/infrared-led) which transmits continuous IR rays to be received by an IR receiver. The output of the receiver varies depending upon its reception of IR rays. Since this variation cannot be analyzed as such, therefore this output can be fed to a comparator. Here operational amplifier (op-amp) of [LM 339](http://engineersgarage.com/content/ic-lm339) is used as comparator.

When the IR receiver does not receive signal the potential at the inverting input goes higher than that that at non-inverting input of the comparator (LM 339). Thus the output of the comparator goes low and the [LED](http://engineersgarage.com/content/led) does not glow .When the IR receiver receives signal the potential at the inverting input goes low. Thus the output of the comparator (LM 339) goes high and the LED starts glowing. [Resistor](http://engineersgarage.com/content/resistor) R1 (100), R2 (10k) and R3 (330 ) are used to ensure that minimum 10 mA current passes through the IR LED, photodiode and normal LED, respectively. Resistor VR2 (preset=5k) is used to adjust the output. Resistor VR1 (preset=10k) is used to set the sensitivity of the circuit. Read more about [IR sensor](http://www.engineersgarage.com/articles/infrared-sensors) here.



**IC4033**

The functions of CD4033B IC pinouts:

 Pin#1: It’s the clock input pinout of the IC, which is assigned for accepting positive clock signals or the pulses which needs to be checked or counted.

 Pin#2: It’s the clock inhibit pinout of the IC, As the name refers to, this pinout could be used for inhibiting the IC from responding to the input pulses by configuring this pinout to the positive supply or the Vdd. Conversely in order to allow normal functioning of the IC this pinout should be grounded.

 Pin#3/#4: These are the Ripple blanking IN and Ripple blanking OUT pinouts of the IC, which provides the user with the option of either allowing the non-significant zeros to be displayed or to be left out from the connected digital displays.

 Pin#14: It’s the “lamp test” pinout of the IC. As the name signifies it is used for testing the connected digital displays in terms of illumination level. When this pinout is connected to a high level or the positive supply, the normal function of the IC is disabled and all the digits of the 7 segment display are applied with a high state so that the digits are allowed to get illuminated together. This allows us to test the intensity levels of the digits and if any of the display digits are not functioning optimally or are dim due to some malfunction.

 Pin#6,7,9,10,11,12,13: All these pinouts are the outputs of the IC which are configured with the discussed 7 segment digital display module.

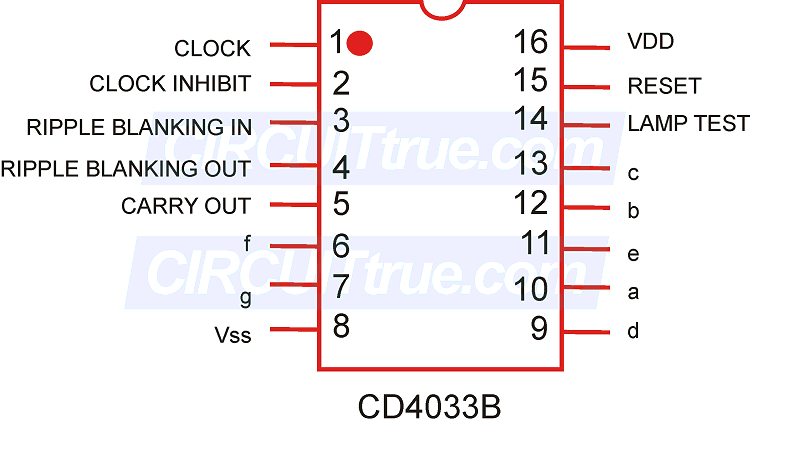
 Pin#15: It’s the reset input of the IC, a high logic or applying the supply voltage to this pin resets the IC completely, resulting in clearing all the data from the display and restoring it to zero.

 Pin#5: It’s the carryout pinout of the IC, it sends a high logic output after every 10 legit clocks at the clock pin#1 of the IC. Thus pin#5 is used as a clock output or a carry forward extension for the next corresponding IC 4033 when many of these are cascaded together in a multi-digit display counter systems.

 Pin#16 is the Vdd or the supply input of the IC.

 Pin#8 is the Vss, or the ground or the negative supply input pinout of the IC 4033.

 The IC works best with supply voltages between 5V and 20V.



#include<reg51.h>

#define on 0

#define off 1

#define reset 1

int scan=1;

//for first segment

int count\_1=0;

int runtime\_1 = 30;

sbit R1 = P2^0;

sbit Y1 = P2^1;

sbit G1 = P2^2;

sbit SS\_CLK\_1 = P2^3;

sbit SS\_RST\_1 = P2^4;

sbit module1\_1 = P0^0;

//for second segment

int count\_2=0;

int runtime\_2 = 30;

sbit R2 = P3^0;

sbit Y2 = P3^1;

sbit G2 = P3^2;

sbit SS\_CLK\_2 = P3^4;

sbit SS\_RST\_2 = P3^3;

sbit module2\_1 = P0^1;

// for third segment

int count\_3=0;

int runtime\_3 = 30;

sbit R3 = P2^5;

sbit Y3 = P2^6;

sbit G3 = P2^7;

sbit SS\_CLK\_3 = P3^7;

sbit SS\_RST\_3 = P3^6;

sbit module3\_1 = P0^2;

// for fourth segment

int count\_4=0;

int rntime\_4 = 30;

sbit R4 = P1^0;

sbit Y4 = P1^1;

sbit G4 = P1^2;

sbit SS\_CLK\_4 = P1^4;

sbit SS\_RST\_4 = P1^3;

sbit module4\_1 = P0^3;

void Delay(unsigned int x)

{

unsigned int i,j;

for(i=0;i<x;i++)

}

void \_reset(int d)

{

int delay = 30;

switch(d)

{

case 0: // reset all

SS\_RST\_1=1;

SS\_RST\_2=1;

SS\_RST\_3=1;

SS\_RST\_4=1;

Delay(delay);

SS\_RST\_1=0;

SS\_RST\_2=0;

SS\_RST\_3=0;

SS\_RST\_4=0;

Delay(delay);

count\_1=0;

count\_2=0;

count\_3=0;

count\_4=0;

break;

case 1:

SS\_RST\_1=1;

msDelay(delay);

SS\_RST\_1=0;

msDelay(delay);

break;

case 2:

SS\_RST\_2=1;

msDelay(delay);

SS\_RST\_2=0;

msDelay(delay);

break;

case 3:

SS\_RST\_3=1;

msDelay(delay);

SS\_RST\_3=0;

msDelay(delay);

break;

case 4:

SS\_RST\_4=1;

msDelay(delay);

SS\_RST\_4=0;

msDelay(delay);

break;

}

}

void clock(int d)

{

int delay = 20;

switch(d)

{

case 1:

count\_1++;

SS\_CLK\_1=1;

msDelay(delay);

SS\_CLK\_1=0;

msDelay(delay);

break;

case 2:

count\_2++;

SS\_CLK\_2=1;

msDelay(delay);

SS\_CLK\_2=0;

msDelay(delay);

break;

case 3:

count\_3++;

SS\_CLK\_3=1;

Delay(delay);

SS\_CLK\_3=0;

Delay(delay);

break;

case 4:

count\_4++;

SS\_CLK\_4=1;

Delay(delay);

SS\_CLK\_4=0;

Delay(delay);

break;

}

}

void main()

{

P2=0x00;

P1=0x00;

P0=0xFF;

R1=off;

Y1=off;

G1=off;

R2=off;

Y2=off;

G2=off;

R3=off;

Y3=off;

G3=off;

R4=off;

Y4=off;

G4=off;

R1=on;

R2=on;

R3=on;

R4=on;

ss\_reset(0);

while(1)

{

clock(1);

clock(2);

clock(3);

clock(4);

switch(scan)

{

case 1:

// scan first segment

if(count\_1 <= 3)

{

if( module1\_1 == 1)

{

runtime\_1=60;

}

else

{

runtime\_1=30;

}

}

R1=off;

G1=on;

if(count\_1 >= runtime\_1)

{

G1=off;

Y1=on;

msDelay(3000);

Y1=off;

ss\_reset(0);

scan=2;

R1=on;

}

break;

case 2:

// scan second segment

if(count\_2 <= 3)

{

if( module2\_1 == 1)

{

runtime\_2=60;

}

else

{

runtime\_2=30;

}

}

R2=off;

G2=on;

if(count\_2 >= runtime\_2)

{

G2=off;

Y2=on;

msDelay(3000);

Y2=off;

ss\_reset(0);

scan=3;

R2=on;

}

break;

case 3:

// scan third segment

if(count\_3 <= 3)

{

if( module3\_1 == 1)

{

runtime\_3=60;

}

else

{

runtime\_3=30;

}

}

R3=off;

G3=on;

if(count\_3 >= runtime\_3)

{

G3=off;

Y3=on;

msDelay(3000);

Y3=off;

ss\_reset(0);

scan=4;

R3=on;

}

break;

case 4:

// scan fourth segment

if(count\_4 <= 3)

{

if( module4\_1 == 1)

{

runtime\_4=60;

}

else

{

runtime\_4=30;

}

}

R4=off;

G4=on;

if(count\_4 >= runtime\_4)

{

G4=off;

Y4=on;

msDelay(3000);

Y4=off;

ss\_reset(0);

scan=1;

R4=on;

}

break;

}

}

}



Part Value Quantity

DIS1 COMMON CATHODE 8

IC2 4033 8

R1 33K 8

R2 47K 4

R3 470E 4

module IR MODULE 8

BRIDGE rectifier 1

C1 33pf 2

C2 470uf 1

C3 .01uf 2

CRYSTAL 11.0592 mhz 1

IC1 AT89S51 1

IC2 7805TV 1

JP1 TRANSFORMER 1

LED 5mm 1

R1 10k 1

R2 560E 1

RN10 10 k 1

Switch1 on/off 1

Switch2 push switch 1

