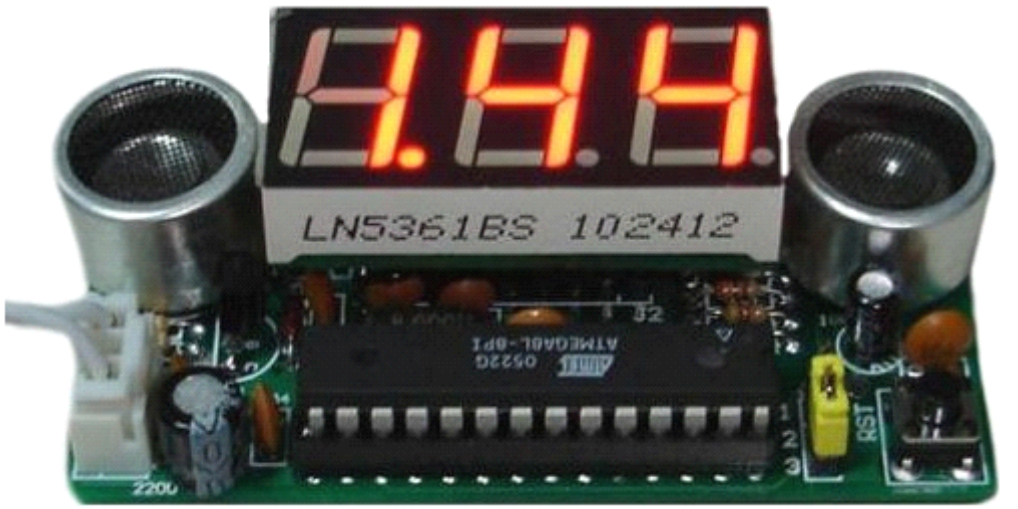
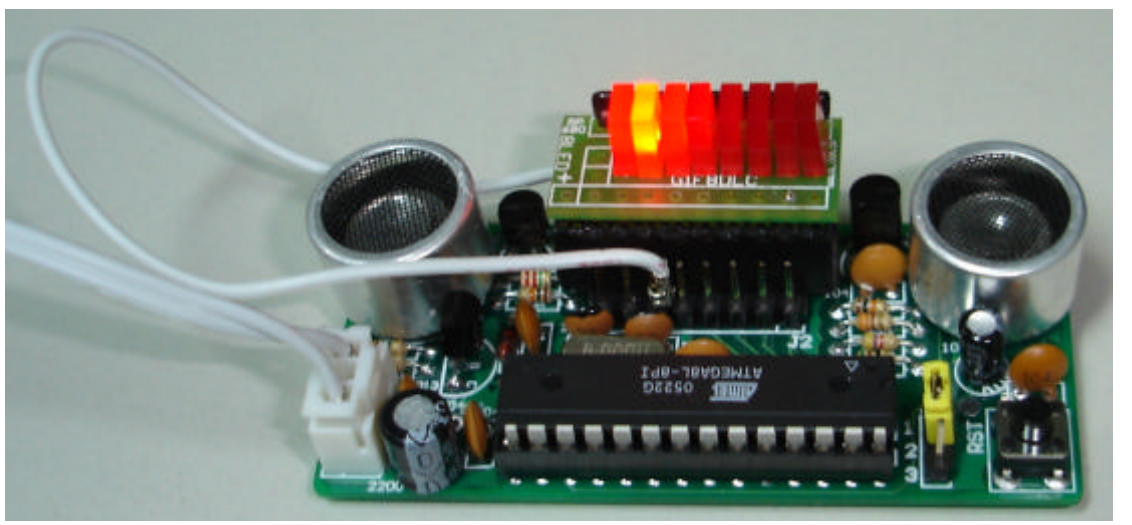
Robot Ultrasonic Ranging Shield

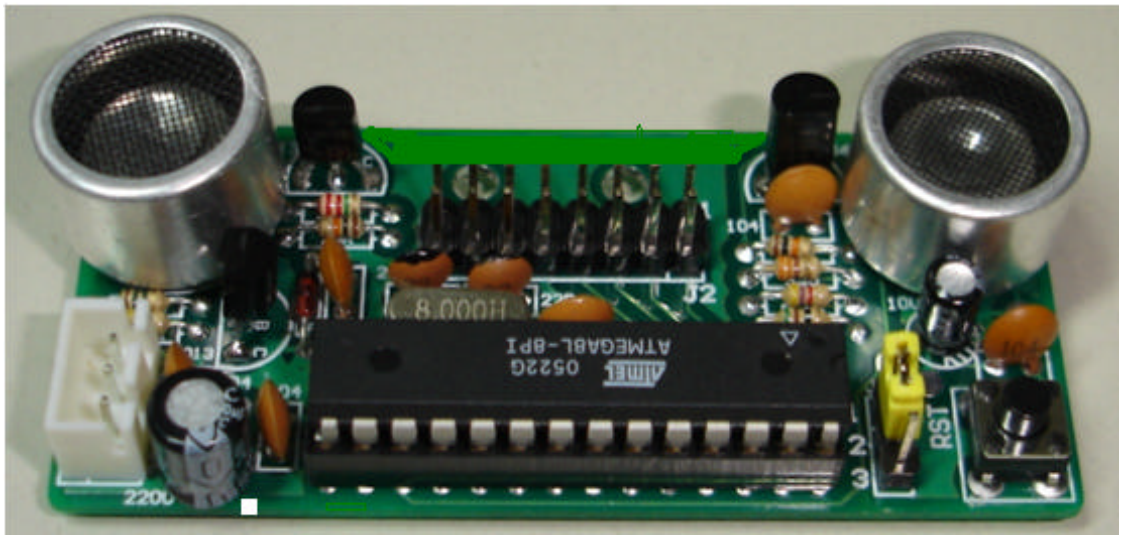
In order to make robot bypass obstacle automatically, it must be equipped with distance measuring system to make timely access to information of direction and distance from the obstacle. Ultrasonic ranging system can be composed of ultrasonic ranging modules in three direction (front, left, right), which can supply distance information from front, left and right of the robot.

Ultrasonic sensors can work separately while connecting to power supply. The 3 LED digital display distance from the barrier. LED digital tube module can be plugged or unplugged from the sensor board. So it can be use on different occasions. Range distance is 10cm-500cm. The error is 1 ~ 2cm while ranging less than 100cm. When greater than 100cm, the error is 2 ~ 4cm. The ultrasonic sensors module can output the distance data from specific I/O interface. It can be directly installed in the robot, for application such as search object searching and obstacle avoidance detection.

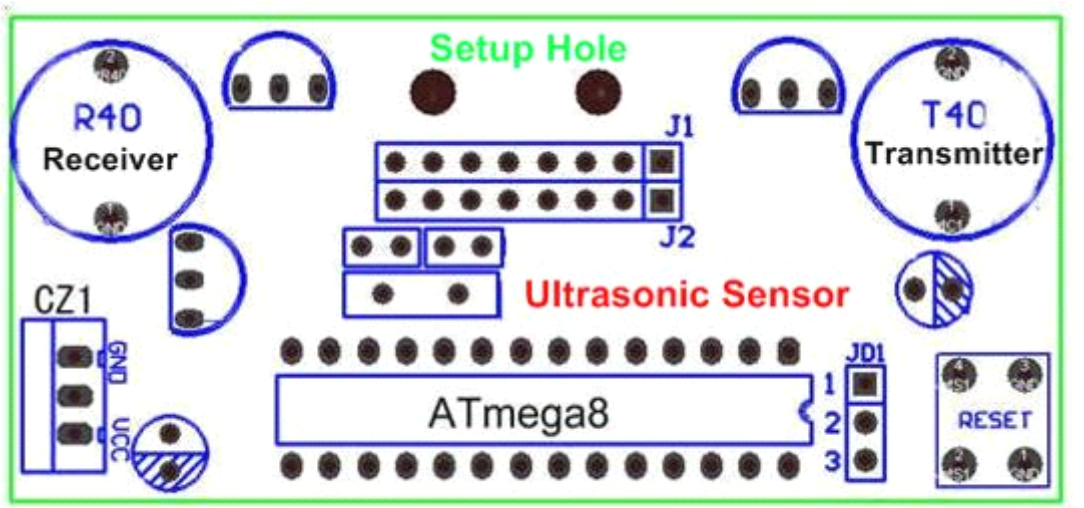


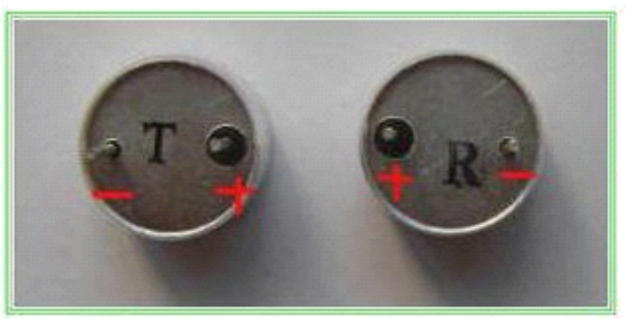


Ultrasonic sensor LED light emitting diode display measured the subsection distance scales, and can be used for robot control



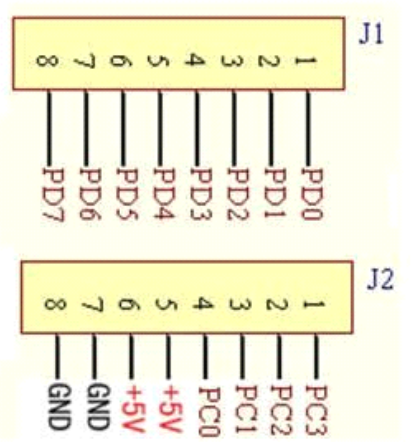
Robot Ultrasonic Ranging Shield without display.





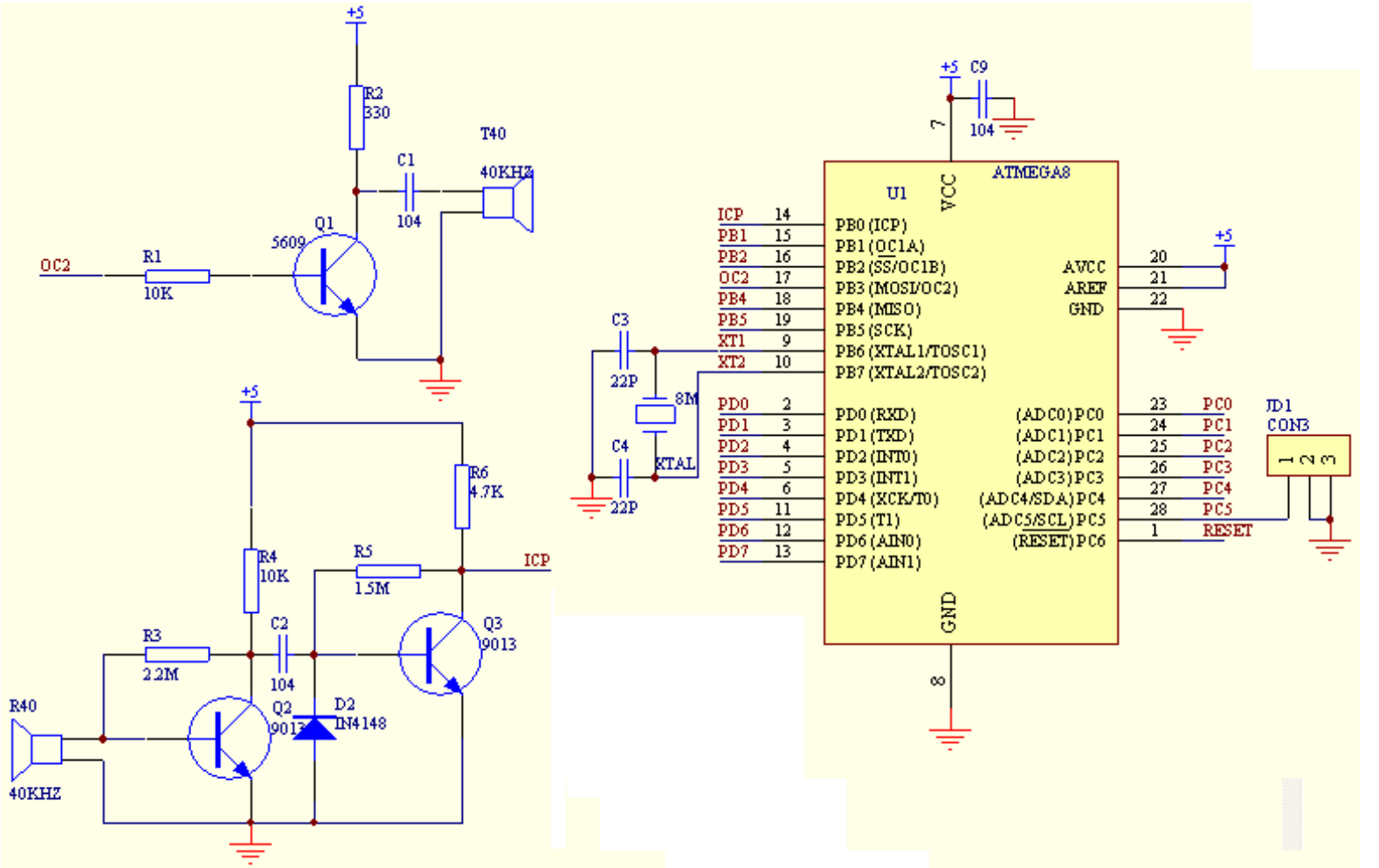
Pin JD1-2/-3 are connected to GND ; JD1-1 is connected to PC5; while short-circuit block is inserted between pin 2 and 3,LED digital display is used the display the distance. If short-circuit block inserted between the 1 and 2, LED light-emitting diode is used for display, or Ultrasonic sensor output section control signal;

J1、J2 Pin Define Function



1: J1-1-PD0 avoidance distance ranging 5 ~ 9cm
2: J1-2-PD1 avoidance distance ranging 10 ~ 19cm
3: J1-3-PD2 avoidance distance ranging 20 ~ 29cm
4: J1-4-PD3 avoidance distance ranging 30 ~ 39cm
5: J1-5-PD4 avoidance distance ranging 40 ~ 49cm
6: J1-6-PD5 avoidance distance ranging 50 ~ 99cm
7: J1-7-PD6 avoidance distance ranging 100 ~ 179cm
8: J1-8-PD7 avoidance distance ranging 180 ~ 250cm

J1、J2 pin function schematic diagram :



Test Program Code For AVR:

```
/*****************************************************************
Project : Robot Ultrasonic Ranging Shield
Chip type : ATMEGA8
Clock frequency: 8.000000MHz
*****************************************************************/
#include<iom8v.h>
#include<macros.h>
#pragma interrupt_handler intt0:10 //T0 INTO
#pragma interrupt_handler icp_timer1:6 //T1 trapped-programinterrupt
#pragma data:code //set data area as program code memory
const unsigned char tab1[]={0X28,0XEE,0X32,0XA2,0XE4,
                            0XA1,0X21,0XEA,0X20,0XA0};//7 segment coding font table (lm,cm)
const unsigned char tab2[]={0X08,0XCE,0X12,0X82,0XC4,
                            0X81,0X01,0XCA,0X00,0X80};//7 segment coding font table (m)
#pragma data:data // set data area back to program code memory
unsigned char ledbuff[]={0X08,0X28,0X28};//display buffer area
unsigned char count;
unsigned char newcount;
unsigned char oldcount;
void intt0(void)//T0 break off timing routine    timing 58US , and ranging    1CM.
{
TCNT0=0XC6;
count++;
}
void icp_timer1(void)//ICP1 trapped-programinterrupt，frequency 38.5～41.6
{
if (96<=ICR1<=104)
{
SREG&=0X7f;
newcount=count-13;
ICR1=0;
TCNT1=0;
TCCR1B=0X81;
}
}
void delay_1us(void)//1us
{
asm("nop");
}
void delay_us(unsigned int t)//tus
```

```
{
unsigned int i=0;
for(i=0;i<t;i++)
delay_1us();
}
void delay_1ms(void)//1ms
{
unsigned int i;
for(i=1;i<1142;i++);
}
void delay_ms(unsigned char t)
{
unsigned char i=0;
for(i=0;i<t;i++)
{
delay_1ms();
}
}
void send40kHz(void) //send 40KHz ultrasonic wave
{
count=0;
TIMSK&=0XDF;//ban ICP1 capable
OCR2=0X64;
TCCR2=0X19;
}
void close40kHz(void) // stop sending ultrasonic wave
{
TCCR2=0X00;
}
void hextobcd(unsigned char m)//count and transform into LED 7 segment code
{
unsigned char temp;
temp=m%10;
ledbuff[0]=tab1[temp];//mm pin
m=m/10;
temp=m%10;
ledbuff[1]=tab1[temp];//lm pin
temp=m/10;
ledbuff[2]=tab2[temp];//m pin
}
void display3led(void) //LED indication
{
unsigned char i;
for(i=0;i<3;i++)
```

```
{
PORTD=ledbuff[i];
PORTC=~(1<<i);//PC0-mm bit，PC1-lm pin，PC2-m pin
delay_1ms();
PORTC=(1<<i);
}
}
void display8led(void) //8LED display
{
if(newcount<10)
PORTD=0XFE;
else if (newcount<20)
PORTD=0XFD;
else if (newcount<30)
PORTD=0XFB;
else if (newcount<40)
PORTD=0XF7;
else if (newcount<50)
PORTD=0XEF;
else if (newcount<100) PORTD=0XDF;
else if (newcount<180)
PORTD=0XBF;
else
PORTD=0X7F;
}
void mcu_init(void)//MCU initialization
{
DDRD=0XFF;
PORTD=0XFF;
DDRC=0XDF; //PC5 input
PORTC=0XFF;
DDRB=0XFE;
PORTB=0XF7;
TCNT2=0X00;
TCNT0=0XC6; //T0 timing 58US
TCCR0=0X02; //T0 timer 1/8 fractional frequency
TCNT1=0X00;
TCCR1A=0X00;
TCCR1B=0X81; //input arrest noise suppression permission，ICP1 failling edge trigger，system clock
TIMSK=0X01; //enable T0 timing overflow
}
void main(void)
{
unsigned char i;
```

```
mcu_init();
for(;;)
{
send40kHz();
SREG|=0X80;
delay_us(50);
close40kHz();
delay_us(20);
TIMSK=0X21;
if(!(PINC&0X20)) //if insert JD1, and the LED will display the distance
{
if (newcount!=oldcount)
oldcount=newcount;
hextobcd(oldcount); for(i=0;i<20;i++)
display3led();
}
else
{ PORTC=0XFF;
display8led();
delay_ms(100);
}
}
```