[Micropython]TPYBoard 超声波测距

1. ****实验目的****

**1. 学习在PC机系统中扩展简单I/O 接口的方法。**

**2. 进一步学习编制数据输出程序的设计方法。**

**3. 学习超声波模块的测距原理。**

**4. 学习LCD5110接线方法**

**5. 学习TPYboard控制超声波模块测距。**

****2.所需元器件****

**超声波模块一个**

**TPYBoard板子一块**

**5110LCD显示屏一个**

**数据线一条**

**杜邦线若干**

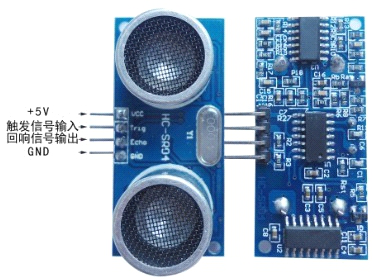
****3.超声波模块工作原理****

**(1)采用IO口TRIG触发测距，给最少10us的高电平信呈。**

**(2)模块自动发送 8 个 40khz 的方波，自动检测是否有信号返回。**

**(3)有信号返回，通过 IO 口 ECHO 输出一个高电平，高电平持续的时间就是超声波从发射到返回的时间。测试距离=(高电平时间\*声速(340M/S))/2。**

**如下图接线，VCC 供 5V电源， GND 为地线，TRIG 触发控制信号输入，ECHO 回响信号输出等四个接口端。**



1. ****控制5110显示屏显示6x8字符****

**先看一下LCD5110针脚含义吧（注意：LCD5110的针脚有些不一样的）**

**TPYBoard的针脚与5110的针脚对应关系如下：**

**TPYBoard       LCD5110    memo**

**————————————————————————————**

**# any   Pin        => RST       Reset pin (0=reset, 1=normal)**

**# any   Pin        => CE        Chip Enable (0=listen for input,   1=ignore input)**

**# any   Pin        => DC        Data/Command (0=commands, 1=data)**

**#   MOSI          => DIN       data flow (Master out, Slave in)**

**#   SCK           => CLK       SPI clock**

**# 3V3   or any Pin  => VCC       3.3V logic voltage (0=off, 1=on)**

**# any   Pin        => LIGHT     Light (0=on, 1=off)**

**#   GND           => GND**

**还是看不明白的话，直接上针脚编号吧**

**TPYBoard       LCD5110    memo**

**————————————————————————————**

**Y10        => RST       Reset pin (0=reset, 1=normal)**

**Y11        => CE        Chip Enable (0=listen for input, 1=ignore input)**

**Y9         => DC        Data/Command (0=commands, 1=data)**

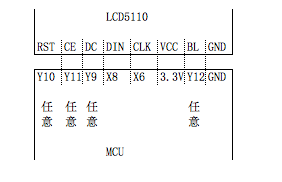
**X8         => DIN       data flow (Master out, Slave in)**

**X6         => CLK       SPI clock**

**VCC**

**Y12        => LIGHT     Light (0=on, 1=off)**

**GND**



**接线ok后，并且导入font.py文件和upcd8544.py文件，编写main.py将测到的距离显示在5110显示屏上，运行main.py就ok了。(font.py和upcd8544.py可以从官网上下载，最后会告诉下载地址)。**

****5.源代码****

**import pyb**

**from pyb import Pin**

**from pyb import Timer**

**import upcd8544**

**from machine import SPI,Pin**

**Trig = Pin('X2',Pin.OUT\_PP)**

**Echo = Pin('X1',Pin.IN)**

**num=0**

**flag=0**

**run=1**

**def start(t):**

**global flag**

**global num**

**if(flag==0):**

**num=0**

**else:**

**num=num+1**

**def stop(t):**

**global run**

**if(run==0):**

**run=1**

**start1=Timer(1,freq=10000,callback=start)**

**stop1=Timer(4,freq=2,callback=stop)**

**while True:**

**if(run==1):**

**SPI = pyb.SPI(1) #DIN=>X8-MOSI/CLK=>X6-SCK**

**#DIN =>SPI(1).MOSI 'X8' data flow (Master out, Slave in)**

**#CLK =>SPI(1).SCK 'X6' SPI clock**

**RST = pyb.Pin('Y10')**

**CE = pyb.Pin('Y11')**

**DC = pyb.Pin('Y9')**

**LIGHT = pyb.Pin('Y12')**

**lcd\_5110 = upcd8544.PCD8544(SPI, RST, CE, DC, LIGHT)**

**Trig.value(1)**

**pyb.udelay(100)**

**Trig.value(0)**

**while(Echo.value()==0):**

**Trig.value(1)**

**pyb.udelay(100)**

**Trig.value(0)**

**flag=0**

**if(Echo.value()==1):**

**flag=1**

**while(Echo.value()==1):**

**flag=1**

**if(num!=0):**

**#print('num:',num)**

**distance=num/10000\*34000/2**

**print('Distance')**

**print(distance,'cm')**

**lcd\_5110.lcd\_write\_string('Distance',0,0)**

**lcd\_5110.lcd\_write\_string(str(distance),6,1)**

**lcd\_5110.lcd\_write\_string('cm',58,1)**

**lcd\_5110.lcd\_write\_string('This is a test of Distance',0,2)**

**flag=0**

**run=0**