《单片机系统设计》实验报告

正弦波信号发生器实验



**学生姓名 \_\_\_赵思蒙**

**学生学号 517021910935**

**学生班级 F1702113**

**任课教师 \_\_付庄\_\_\_**

**同组同学 陆以凡**

**实验日期 \_\_2020.10.26\_\_**

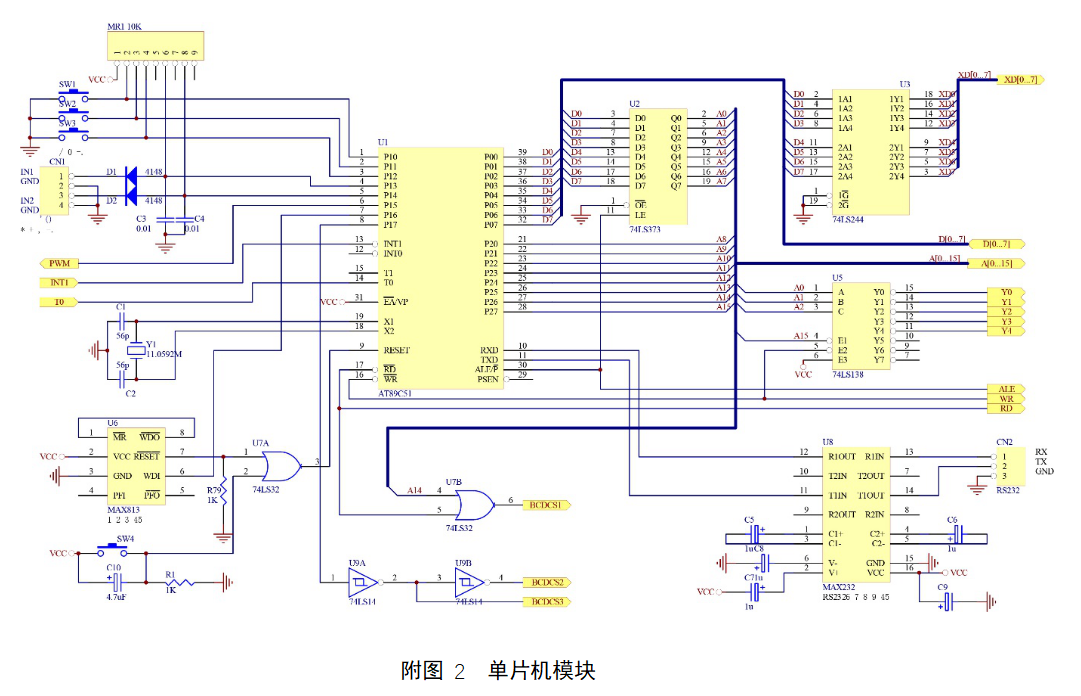
**一、实验目的**

熟悉采用C语言对51单片机进行DA转换、中断程序、键盘、LED显示、定时器/计数器、查表编程。

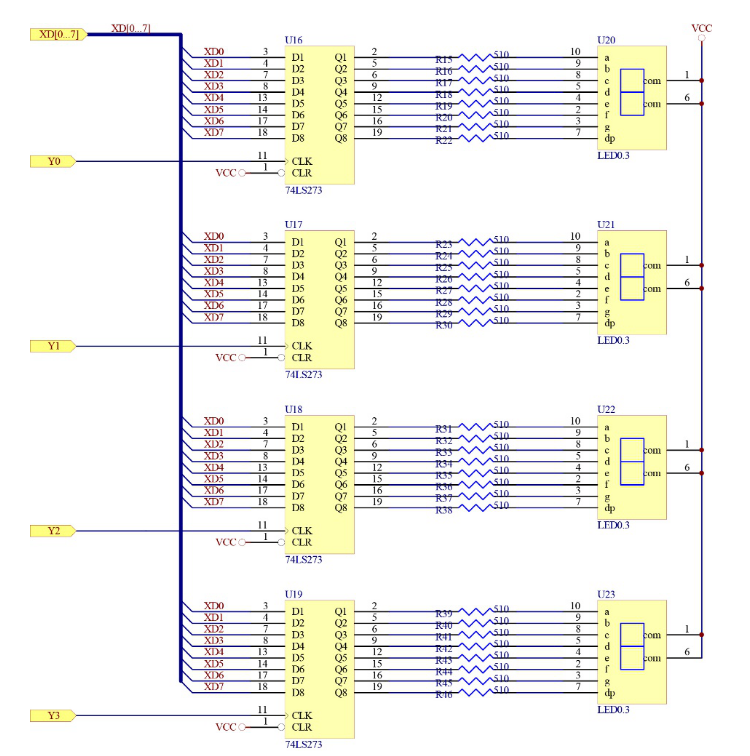
**二、实验内容**

1. 设计一幅值可调，周期固定的正弦波发生器。正弦波幅值的调节范围是Vpp5V ~10V，周期固定为100ms；
2. 正弦波幅值的调节由按键SW1完成，按键每按下一次，幅值增加1V，幅值达到10V后返回到5V，幅值在LED上以V为单位显示；
3. 信号的周期由定时器的计时周期细分并确定；
4. 定时器溢出采用中断模式，熟悉采用C语言中断函数的编写。

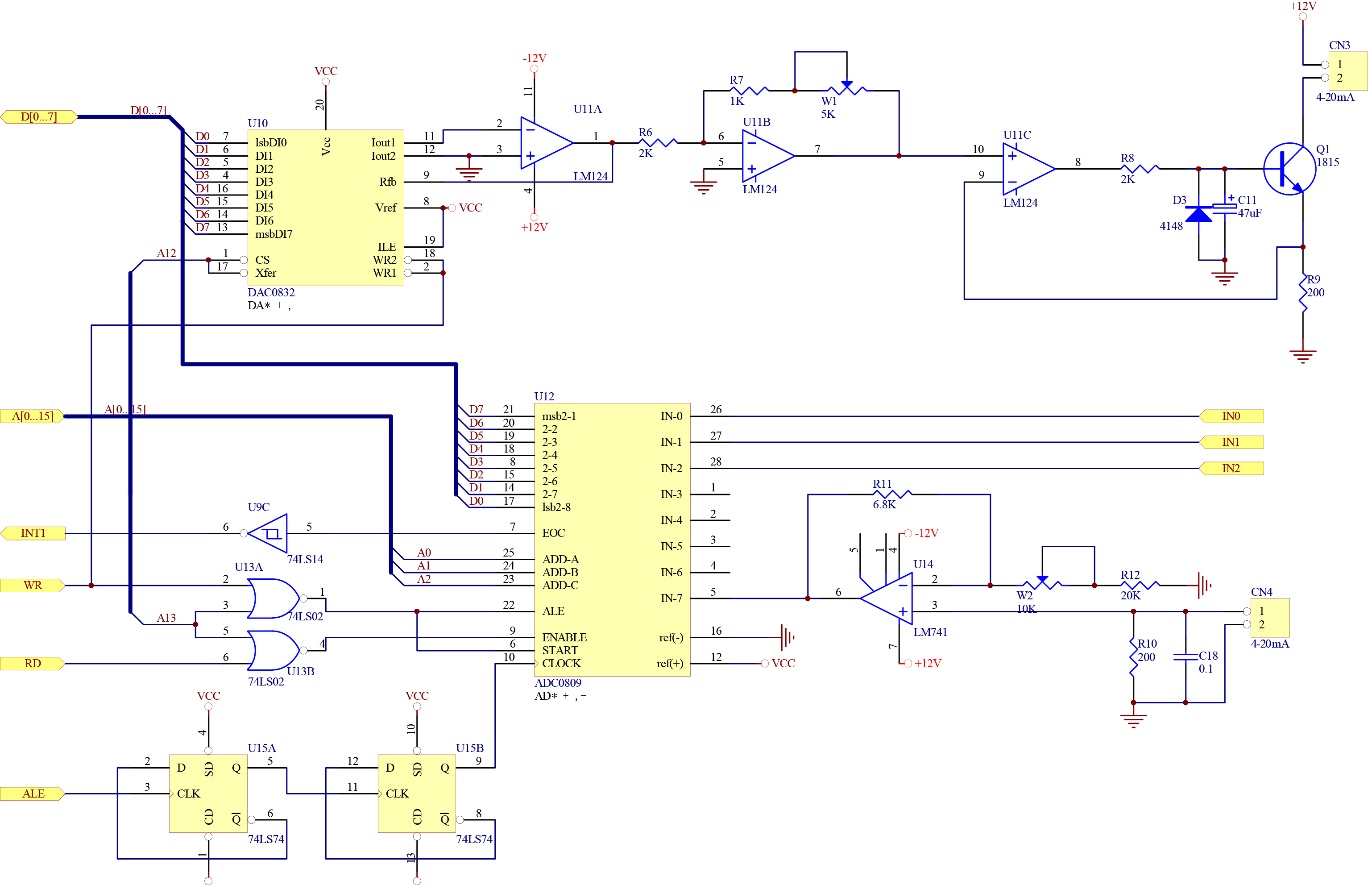
**三、实验电路**



单片机电路



LED电路



AD、DA模块



输出元件

1. 编程流程

首先通过matlab获取100个sin函数值，储存在数组中。

然后按主程序和中断程序分别编程。

|  |  |
| --- | --- |
| 主程序 | 中断子程序 |
|  |  |

1. 程序代码

#include<AT89X51.h>

#include<absacc.h>

#include<stdio.h>

#include<math.h>

#define LED0 XBYTE[0x7FF8]

#define LED1 XBYTE[0x7FF9]

#define LED2 XBYTE[0x7FFA]

#define LED3 XBYTE[0x7FFB]

#define DAC XBYTE[0xEFFF]

#define H 0xFF

#define L 0x9C

#define uchar unsigned char

#define uint unsigned int

char code dx516[3] \_at\_ 0x003b;

uchar code sin\_tab[]={64,68,72,76,80,84,88,91,95,98,102,105,108,111,113,116,118,120,122,124,125,126,127,127,128,128,128,127,127,126,125,124,122,120,118,116,113,111,108,105,102,98,95,91,88,84,80,76,72,68,64,60,56,52,48,44,40,37,33,30,26,23,20,17,15,12,10,8,6,4,3,2,1,1,0,0,0,1,1,2,3,4,6,8,10,12,15,17,20,23,26,30,33,37,40,44,48,52,56,60};

uchar data Tcount;

uchar data v; //times v=5~10

void LEDfun(uchar y,uchar x,bit d);

void show(uint x);

void delay(uint t);

void main(void)

{

//LEDfun(3,0,0);LEDfun(2,0,0);LEDfun(1,0,0);LEDfun(0,0,0);

uint i;

IE=0x8A;//EA=1;ET1=1;ET0=1;

TMOD=0x11;//T0 MODE 1 ;T1 MODE 1

//TH0=0x24;TL0=0x00;//10ms,use for delay

TH1=H;TL1=L;//1ms

Tcount=0;//0-100

//TCON=0x04;

v=5;

TR1=1;//start T1

while(1)

{

show(v);

if(P1\_0==0)//SW1 MAYBE DOWN

{

for(i=0;i<1000;++i);

if(P1\_0==0)

{

if(v<10){++v;}

else{v=5;}

while(P1\_0==0);

}

}

}

}

void show(uint x) //4

{

LEDfun(3,x/1000,0);

LEDfun(2,(x/100)%10,0);

LEDfun(1,(x/10)%10,0);

LEDfun(0,x%10,0);

}

void LEDfun(uchar y,uchar x,bit d) //共阳极 d=1表示需要小数点

{

uchar xdata \*led\_adr;

uint dd;

if(d==1)

{

dd=0x7F;

}

else

{

dd=0xFF;

}

switch(y)

{

case 0:{led\_adr=&LED0;break;}

case 1:{led\_adr=&LED1;break;}

case 2:{led\_adr=&LED2;break;}

case 3:{led\_adr=&LED3;break;}

default:{break;}

}

switch(x)

{

case 0:{\*led\_adr=0xC0&dd;break;}

case 1:{\*led\_adr=0xF9&dd;break;}

case 2:{\*led\_adr=0xA4&dd;break;}

case 3:{\*led\_adr=0xB0&dd;break;}

case 4:{\*led\_adr=0x99&dd;break;}

case 5:{\*led\_adr=0x92&dd;break;}

case 6:{\*led\_adr=0x82&dd;break;}

case 7:{\*led\_adr=0xF8&dd;break;}

case 8:{\*led\_adr=0x80&dd;break;}

case 9:{\*led\_adr=0x90&dd;break;}

default:{\*led\_adr=0x88&dd;break;}

}

}

void T1\_int(void) interrupt 3

{

//x=sin(Tcount\*2\*3.1416/100)\*v;

uint output;

//float th=Tcount\*0.063467;

output=sin\_tab[Tcount]\*v\*0.177;

//output=(sin(th)+1)\*10.5\*v;

//output=Tcount\*2;

DAC=output;

//show(v);

//write sin to dac

Tcount++;

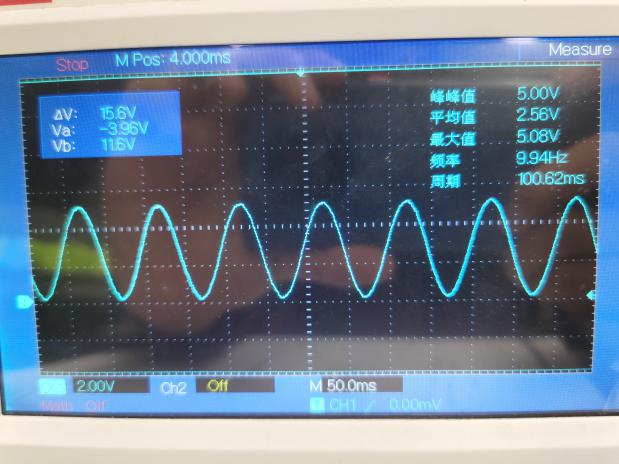
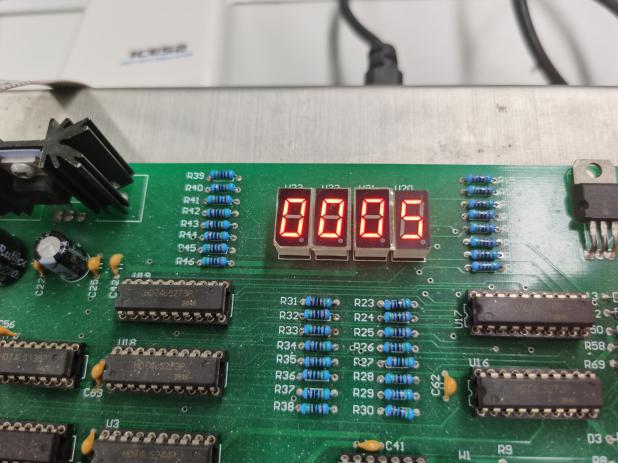
if(Tcount==100){Tcount=0;}

TH1=H;TL1=L;//1ms

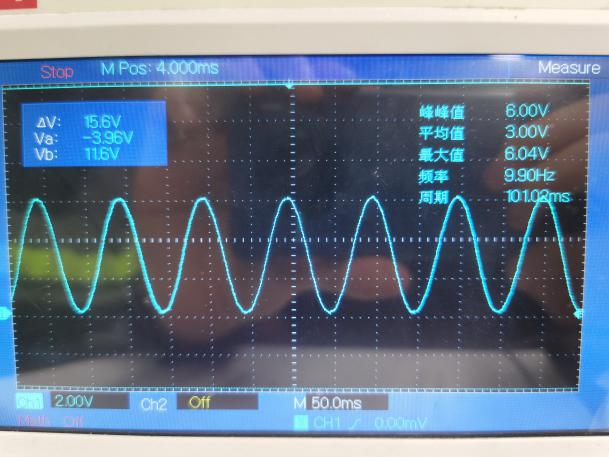
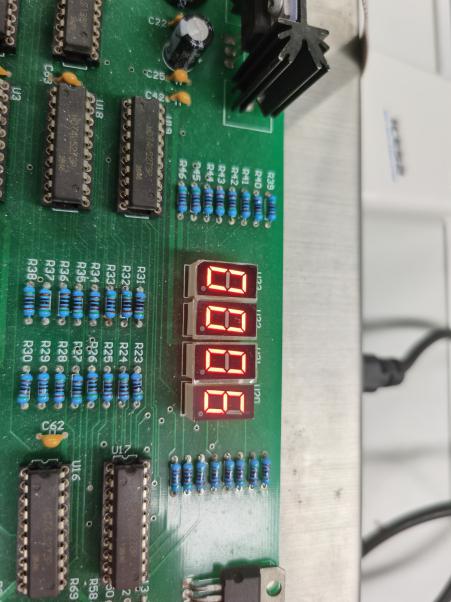
}

1. 结果(周期100ms)

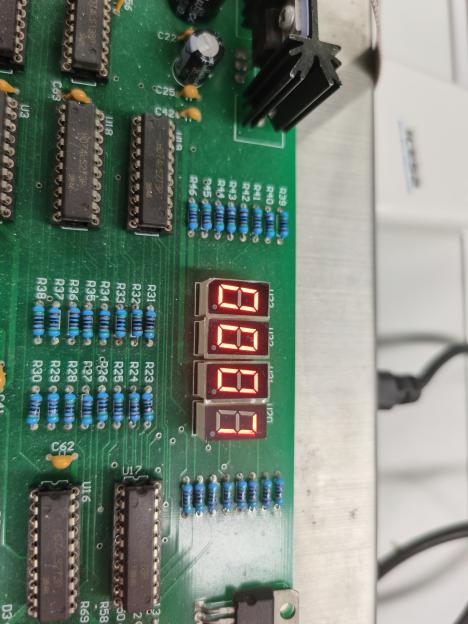
5V：



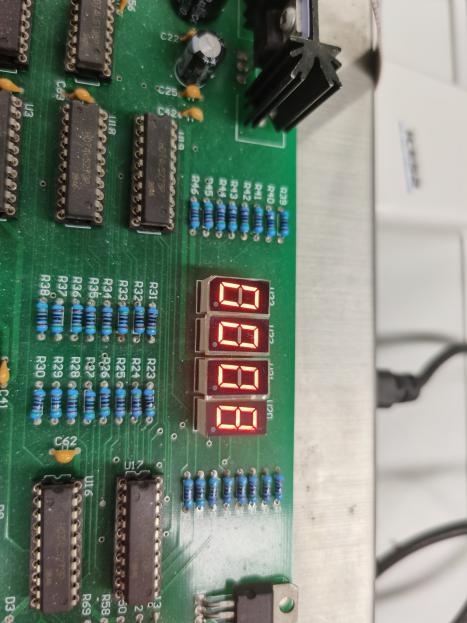
6V:



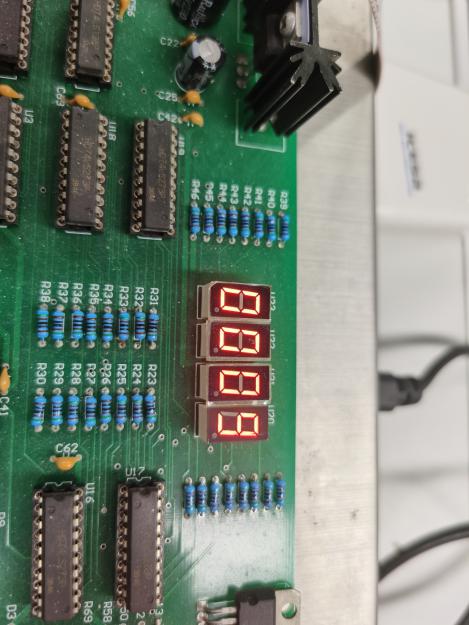
7V:



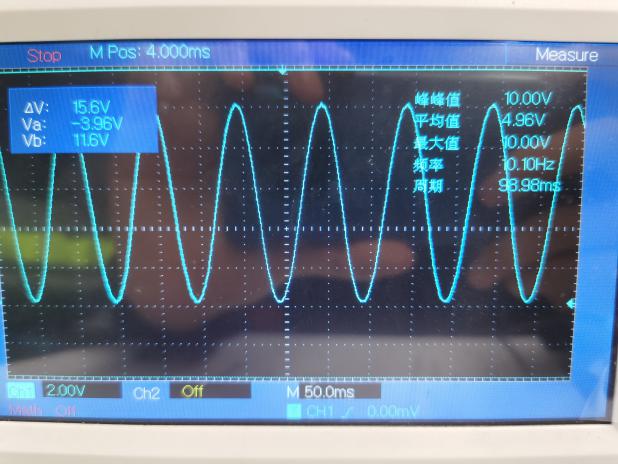
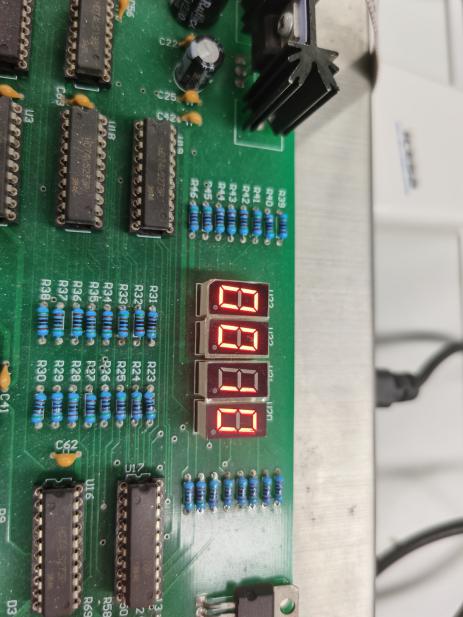
8V:



9V:



10V:



1. 总结以及反思

本次实验编程中主要遭遇了以下几个问题。

问题一：正弦信号有负值，D/A转换中无法实现

解决方案：将整个sin函数值加一，这样函数值在0-2内变化。可以实现D/A转换，且不影响峰峰值。

问题二：信号幅值偏小

解决方案：电压在电路中与理想情况有偏差，可能是电路中有损耗电阻。通过增大设置的电压参数解决问题。

问题三：周期远高于设定值

解决方案：第一次运行时，我们将sin 函数的计算过程放在了中断函数中，由于浮点数的运算时间较长，输出函数的周期到了600ms. 后改为直接输入存有函数值的数组，大大减少了运算时间，但在设定每个函数值1ms延迟的情况下，周期仍达到了200ms,推测为实验中的其它运算也有耗时。最终将人工设定的延时减小到0.3ms后，周期达到了100ms附近。 但在8V 和 10V时周期会略短于100ms,推测偶数的预算可能更为迅速。

本次实验圆满完成了任务目标。熟悉了D/A转换的使用方法，以及参数调整的思路。感谢老师的指导和同组同学的帮助