CC461 – Embedded Systems

Frequency Generator Project

by

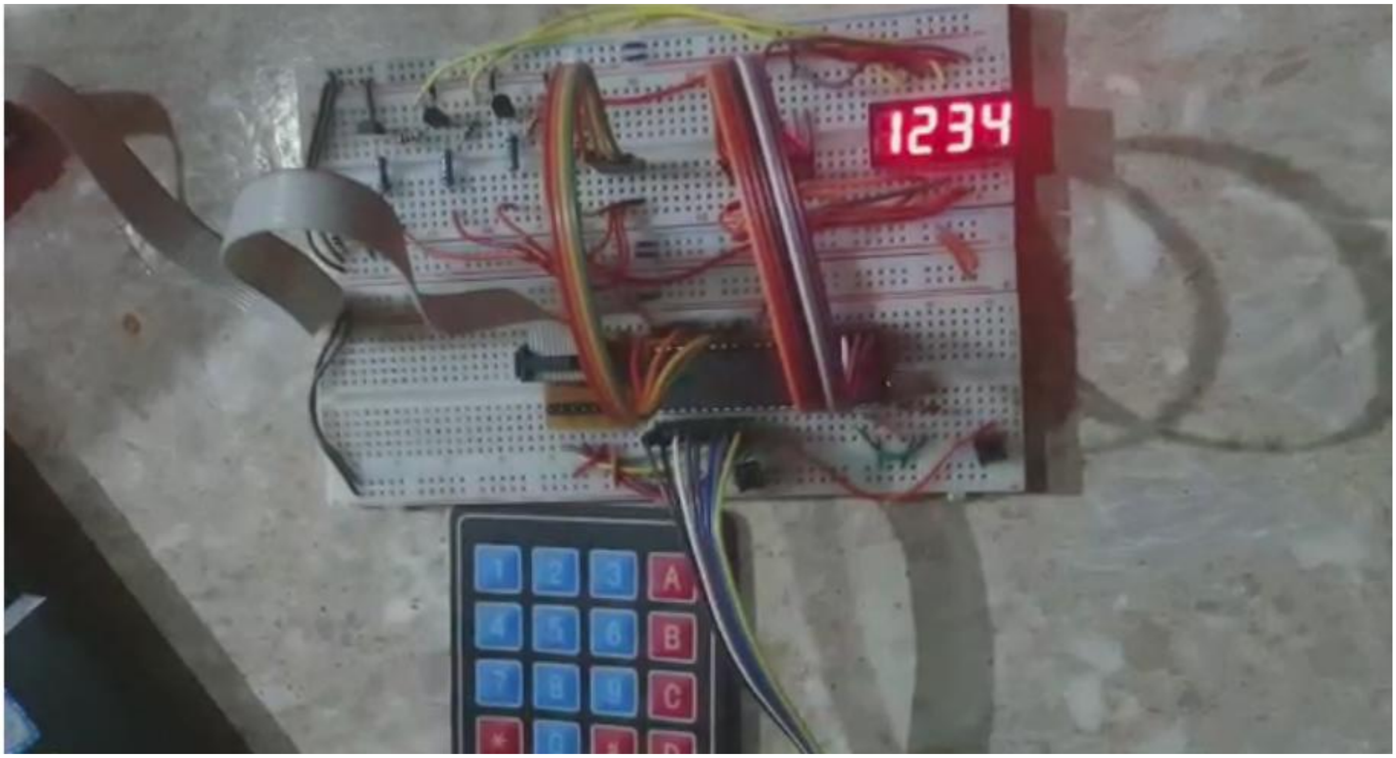
Team 14:

Adham Amr Mohamed Kamal	ID: 6713
Hossam Eldin Ahmed Shaaban	ID: 6721
Ehab Sherif Ahmed	ID: 6738
Raphael Guirguis Fawaz	ID: 7252

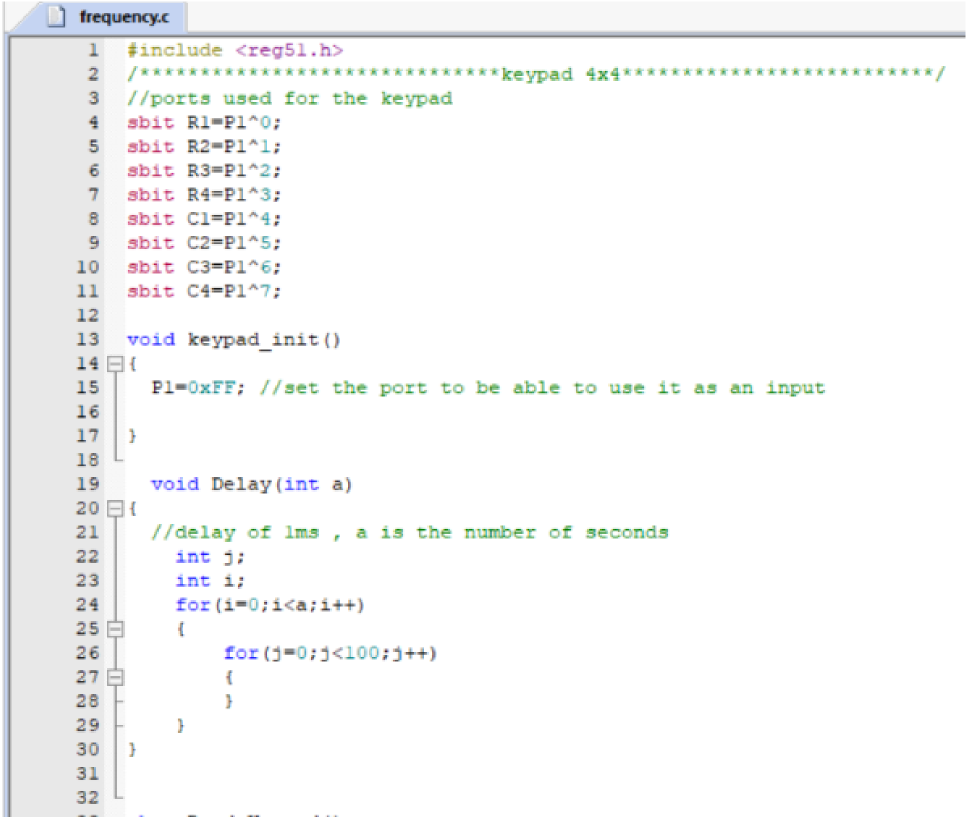
Project Summary:

The frequency generator project is a system that utilizes an 8051 microcontroller to generate a square wave with different frequencies up to 9999 Hz. The system takes in the frequency value via a hex keypad connected to P1 and displays it on a 4-digit seven-segment display connected to P2 with control of the four common anodes done using P0.0 to P0.3 via 2N3906 transistors. Timer 0 interrupt is used to refresh the display. After entering the frequency value, a push button connected to INT0 is pressed to enable the generation of the square wave on pin P3.7. If a new frequency value is to be entered, the switch connected to INT0 is pressed again. The crystal used with the project is 11.0592 MHz.

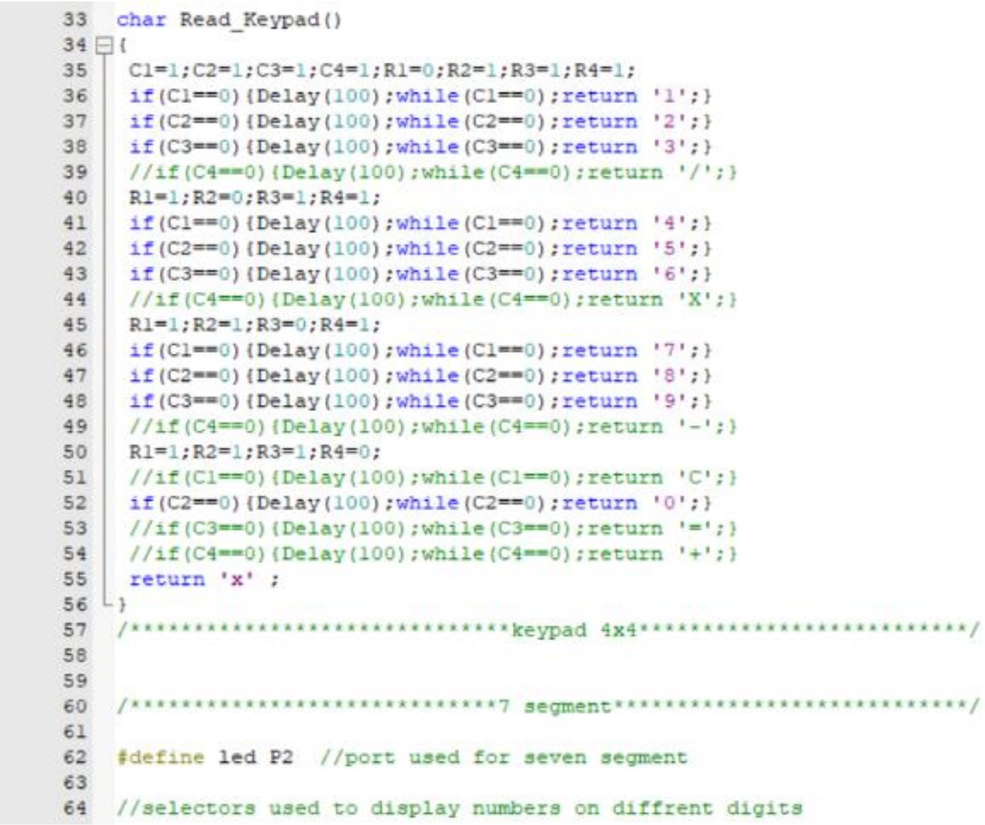
Circuit:



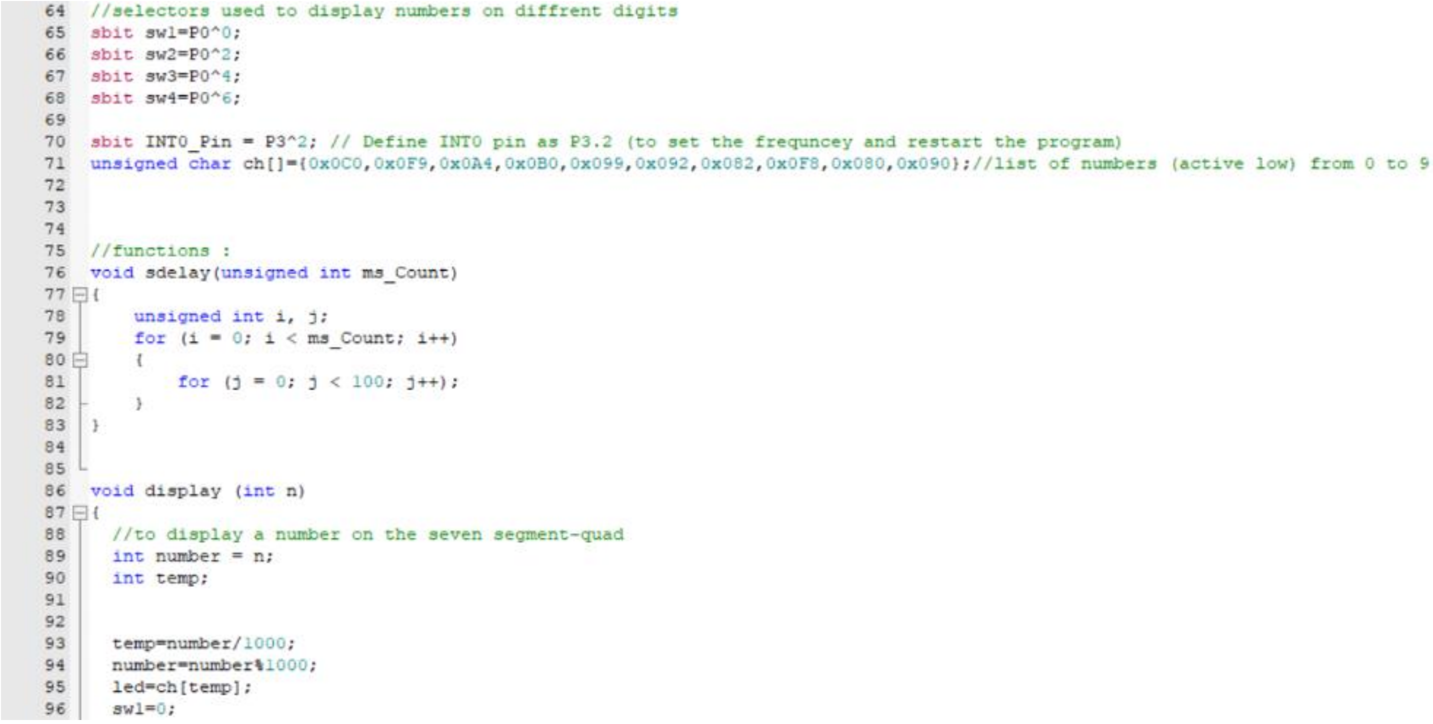
Source code:



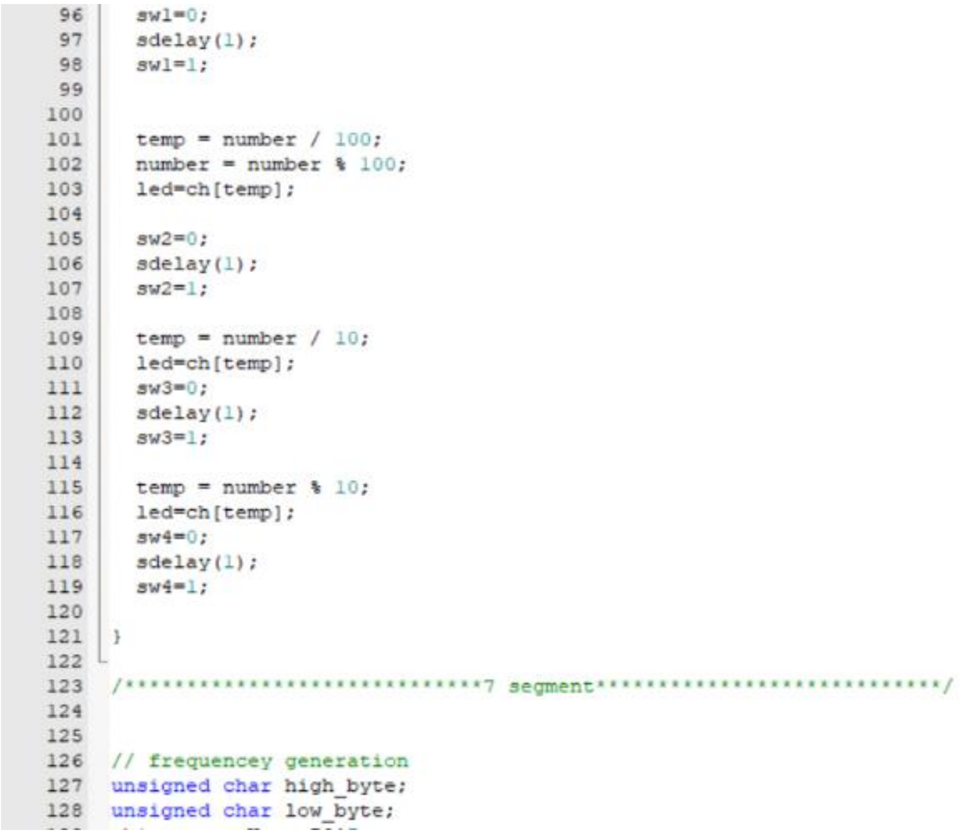
```c
#include <reg51.h>
/*****************************keypad 4x4**************************/
//ports used for the keypad
sbit R1=P1^0;
sbit R2=P1^1;
sbit R3=P1^2;
sbit R4=P1^3;
sbit C1=P1^4;
sbit C2=P1^5;
sbit C3=P1^6;
sbit C4=P1^7;

void keypad_init()
{
  P1=0xFF; //set the port to be able to use it as an input

}

  void Delay(int a)
{
  //delay of 1ms , a is the number of seconds
    int j;
    int i;
    for(i=0;i<a;i++)
    {
        for(j=0;j<100;j++)
        {
        }
    }
}


```



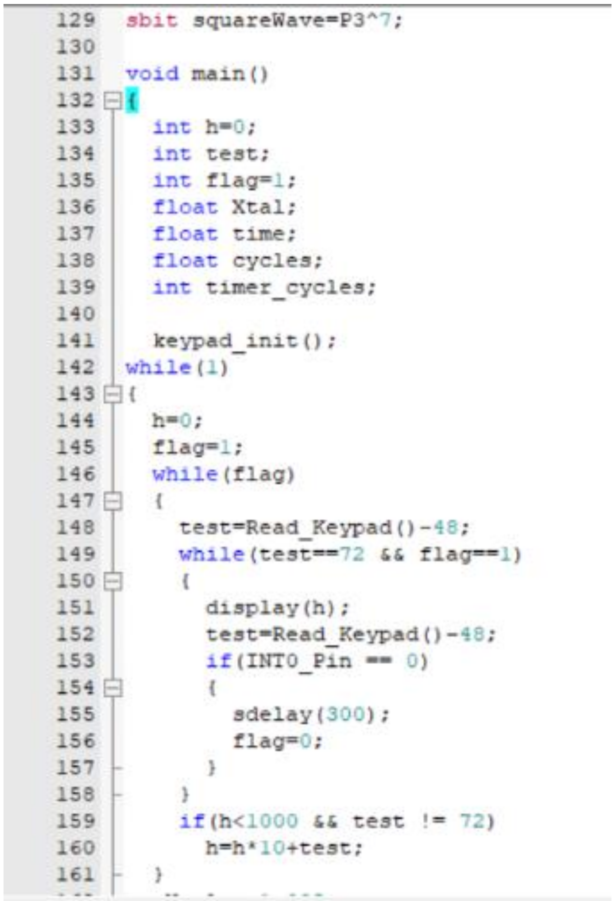
```c
char Read_Keypad()
{
 C1=1;C2=1;C3=1;C4=1;R1=0;R2=1;R3=1;R4=1;
 if(C1==0){Delay(100);while(C1==0);return '1';}
 if(C2==0){Delay(100);while(C2==0);return '2';}
 if(C3==0){Delay(100);while(C3==0);return '3';}
 //if(C4==0){Delay(100);while(C4==0);return '/';}
 R1=1;R2=0;R3=1;R4=1;
 if(C1==0){Delay(100);while(C1==0);return '4';}
 if(C2==0){Delay(100);while(C2==0);return '5';}
 if(C3==0){Delay(100);while(C3==0);return '6';}
 //if(C4==0){Delay(100);while(C4==0);return 'X';}
 R1=1;R2=1;R3=0;R4=1;
 if(C1==0){Delay(100);while(C1==0);return '7';}
 if(C2==0){Delay(100);while(C2==0);return '8';}
 if(C3==0){Delay(100);while(C3==0);return '9';}
 //if(C4==0){Delay(100);while(C4==0);return '-';}
 R1=1;R2=1;R3=1;R4=0;
 //if(C1==0){Delay(100);while(C1==0);return 'C';}
 if(C2==0){Delay(100);while(C2==0);return '0';}
 //if(C3==0){Delay(100);while(C3==0);return '=';}
 //if(C4==0){Delay(100);while(C4==0);return '+';}
 return 'x' ;
}
/********************************keypad 4x4*************************/


/******************************7 segment***************************/

#define led P2  //port used for seven segment

//selectors used to display numbers on diffrent digits
```



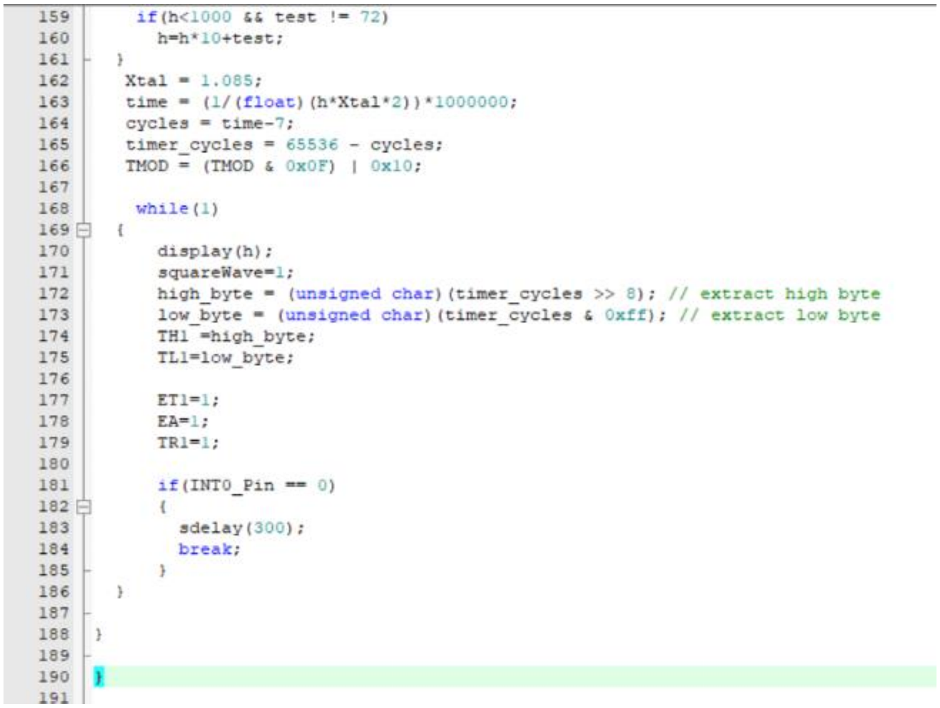
```c
//selectors used to display numbers on diffrent digits
sbit sw1=P0^0;
sbit sw2=P0^2;
sbit sw3=P0^4;
sbit sw4=P0^6;

sbit INT0_Pin = P3^2; // Define INT0 pin as P3.2 (to set the frequncey and restart the program)
unsigned char ch[]={0x0C0,0x0F9,0x0A4,0x0B0,0x099,0x092,0x082,0x0F8,0x080,0x090};//list of numbers (active low) from 0 to 9



//functions :
void sdelay(unsigned int ms_Count)
{
    unsigned int i, j;
    for (i = 0; i < ms_Count; i++)
    {
        for (j = 0; j < 100; j++);
    }
}


void display (int n)
{
  //to display a number on the seven segment-quad
  int number = n;
  int temp;


  temp=number/1000;
  number=number%1000;
  led=ch[temp];
  sw1=0;
```



```c
  sw1=0;
  sdelay(1);
  sw1=1;


  temp = number / 100;
  number = number % 100;
  led=ch[temp];

  sw2=0;
  sdelay(1);
  sw2=1;

  temp = number / 10;
  led=ch[temp];
  sw3=0;
  sdelay(1);
  sw3=1;

  temp = number % 10;
  led=ch[temp];
  sw4=0;
  sdelay(1);
  sw4=1;

}

/******************************7 segment****************************/


// frequencey generation
unsigned char high_byte;
unsigned char low_byte;
```

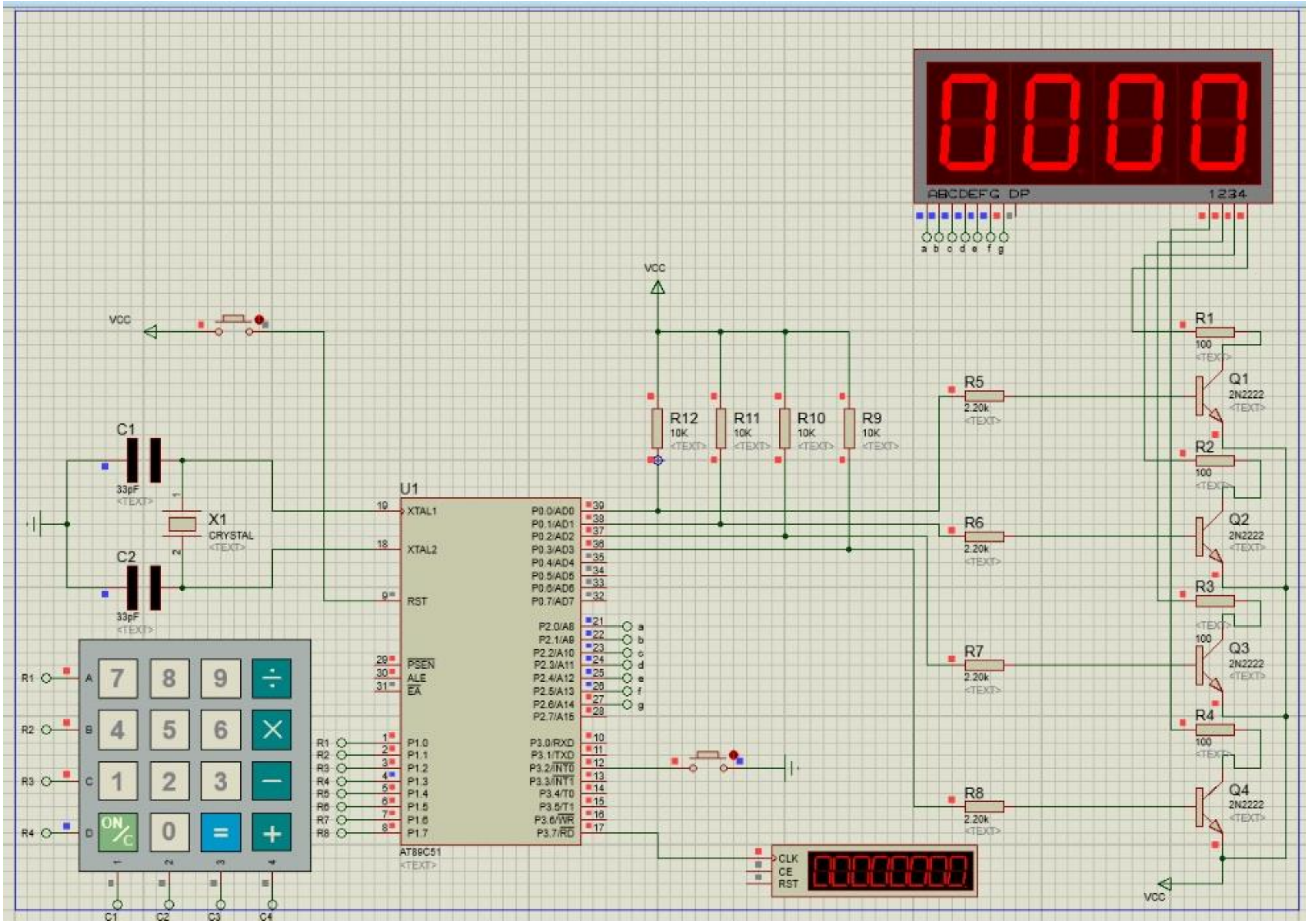


```c
sbit squareWave=P3^7;

void main()
{
  int h=0;
  int test;
  int flag=1;
  float Xtal;
  float time;
  float cycles;
  int timer_cycles;

  keypad_init();
while(1)
{
  h=0;
  flag=1;
  while(flag)
  {
    test=Read_Keypad()-48;
    while(test==72 && flag==1)
    {
      display(h);
      test=Read_Keypad()-48;
      if(INT0_Pin == 0)
      {
        sdelay(300);
        flag=0;
      }
    }
    if(h<1000 && test != 72)
      h=h*10+test;
  }
```



```c
    if(h<1000 && test != 72)
      h=h*10+test;
  }
   Xtal = 1.085;
   time = (1/(float)(h*Xtal*2))*1000000;
   cycles = time-7;
   timer_cycles = 65536 - cycles;
   TMOD = (TMOD & 0x0F) | 0x10;

    while(1)
  {
      display(h);
      squareWave=1;
      high_byte = (unsigned char)(timer_cycles >> 8); // extract high byte
      low_byte = (unsigned char)(timer_cycles & 0xff); // extract low byte
      TH1 =high_byte;
      TL1=low_byte;

      ET1=1;
      EA=1;
      TR1=1;

      if(INT0_Pin == 0)
      {
        sdelay(300);
        break;
      }
  }

}

}
```

Proteus Simulation:


VCC
C1
33pF
X1
CRYSTAL
C2
33pF
U1
19 XTAL1
18 XTAL2
9 RST
29 PSEN
30 ALE
31 EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
R1
R2
R3
R4
R5
R6
R7
R8
P0.0/AD0 39
P0.1/AD1 38
P0.2/AD2 37
P0.3/AD3 36
P0.4/AD4 35
P0.5/AD5 34
P0.6/AD6 33
P0.7/AD7 32
P2.0/A8 21
P2.1/A9 22
P2.2/A10 23
P2.3/A11 24
P2.4/A12 25
P2.5/A13 26
P2.6/A14 27
P2.7/A15 28
P3.0/RXD 10
P3.1/TXD 11
P3.2/INT0 12
P3.3/INT1 13
P3.4/T0 14
P3.5/T1 15
P3.6/WR 16
P3.7/RD 17
AT89C51
a b c d e f g
R12 10K
R11 10K
R10 10K
R9 10K
R5 2.20k
R6 2.20k
R7 2.20k
R8 2.20k
R1 100
R2 100
R3 100
R4 100
Q1 2N2222
Q2 2N2222
Q3 2N2222
Q4 2N2222
ABCDEFG DP
1234
0000
CLK
CE
RST
7 8 9 ÷
4 5 6 ×
1 2 3 −
ON/C 0 = +
A B C D
1 2 3 4
C1 C2 C3 C4