Practical 2

Single LED

#include<reg51.h>

void delay()

{

int i;

for (i=0; i<20; i++)

{

TMOD = 0X01;

TL0=0XB0;

TH0=0X3C;

TR0 = 1;

while(TF0==0)

TF0= 0;

TR0=0;

}

}

void main()

{

while (1)

{

P2 = 0x00;

delay();

P2= 0xFF;

delay();

}

}

Practical 3,4

8-LED

#include<reg51.h>

void delay()

{

int i;

for(i=0;i<30000;i++);

}

void main(){

P2=0X01;

delay();

P2=0X02;

delay();

P2=0X04;

delay();

P2=0X08;

delay();

P2=0X80;

delay();

P2=0X40;

delay();

P2=0X20;

delay();

P2=0X10;

delay();

}

Practical 5

Stepper Motor

#include<reg51.h>

sbit motp=P1^0;

sbit motn=P1^1;

void main()

{

unsigned int i;

motp=motn=0;

while(1)

{

motp=1;

motn=0;

for(i=0;i<60000;i++);

motp=0;

motn=1;

for(i=0;i<60000;i++);

}

}

Practical 6

7 segment

#include <reg51.h>

void delay(unsigned int ms)

{

unsigned int i,j;

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

for(j=0;j<1275;j++);

}

void main(void)

{

char number[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};

int i,j;

P2=0x00;

P3=0x00;

while(1)

{

for(i=0;i<=9;i++)

{

P2=number[i];

for(j=0;j<=9;j++)

{

P3=number[j];

delay(50);

}

}

}

}

Practical 7

Traffic

#include <reg51.h>

sbit red=P1^0;

sbit yellow=P1^1;

sbit green=P1^2;

void main()

{

unsigned int i;

red=yellow=green=0;

while(1)

{

red = 1;

for(i=0; i< 60000; i++);

red = 0;

yellow=1;

for (i=0; i<60000; i++);

yellow=0;

green=1;

for(i=0; i<60000; i++);

for(i=0; i<60000; i++);

green=0;

}

}

Pratical 8

Square wave

#include<reg51.h>

void delay();

void main()

{

while(1)

{

P1=0xFF;

delay();

P1=0x00;

delay();

}

}

void delay()

{

unsigned int i,j,k;

for(i=0;i<10;i++);

for(j=0;j<200;j++);

for(k=0;k<300;k++);

}

Pratical 9

Triangular wave

#include<reg51.h>

void main(void)

{

P1=0x00;

while(1)

{

P1+=0x05;

}

while(P1<0xFF)

do

{

P1-=0x05;

}

while(P1>0x00);

}

Practical 10

Sine wave

#include<reg51.h>

#include<intrins.h>

void main()

{

unsigned i;

unsigned int wavevalue [16]={128,192,224,255,240,224,192,128,64,32,60,10,0,32,64};

while(1)

{

for(i=0;i<16;i++)

{

P1=wavevalue[i];

\_nop\_();

\_nop\_();

\_nop\_();

\_nop\_();

\_nop\_();

\_nop\_();

}

}

}