#include<reg51.h> //header file included

void delay(unsigned int time); //delay function declaration

void main()

{

unsigned int i;

while(1) //infinite loop for continuous saw tooth waveform generation

{

P2 = 0x55;

delay(100);//delay function call

P2 = 0xAA;

delay(100);//delay function call

}

}

void delay(unsigned int time) //for delay generation

{

unsigned int i,j;

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

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

}

#include<reg51.h>

void delay(long int x)

{

int i,j;

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

{

}

}

unsigned char arr[10] = {0xF6,0x90,0xE5,0xB5,0x93,0x37,0x77,0x94,0xF7,0xB7};

void main()

{

while(1)

{

int i;

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

{

P0 = ~arr[i];

delay(5000);

}

}

}

#include<reg51.h>

unsigned int STEP[] = {8,4,2,1};

void delayms(unsigned long x)

{

unsigned int i;

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

}

void main(void)

{ unsigned char k;

while(1)

{ for(k=0; k<3;k++)

{

P1= ~STEP[k];

delayms(5);

}

}

}

/\*------------------LCD---------------------\*/

/\*

Interfacing Pins :

P0.0 thru P0.7 interfaced to D0 thru D7.

P3.2 ----> LCD\_RS (Register Select).

P3.3 ----> LCD\_EN (Enable).

-----------------------------------------------------------------------------

Jumper Settings:

J1:2-3 J2:1-2 J3:2-3 J4:1-2 J5:1-2 J6:1-2

=============================================================================

\*/

#include<REG51.h>

//#define Enable 0x08

//#define RS 0x04

sbit en = P3^3;

sbit rs = P3^2;

void DelayMs(delay)

{

int i,j;

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

{

for(j=0;j<100;j++);

}

}

void write\_lcd\_data(value)

{

/\* data for lcd \*/

P0 = value;

rs = 1;

en = 1;

DelayMs(10);

en = 0;

}

void write\_lcd\_command(value)

{

/\* command for lcd \*/

P0 = value;

rs = 0;

en = 1;

DelayMs(10);

en = 0;

}

void main(void)

{

P0 = 0x00;

P3 = 0x00;

while(1)

{

write\_lcd\_command(0x38); //function set

DelayMs(100);

write\_lcd\_command(0x0E); //display off

DelayMs(100);

write\_lcd\_command(0x01); //display clear

DelayMs(100);

write\_lcd\_command(0x06); //entry mode set

DelayMs(100);

write\_lcd\_command(0x0F); //display on

DelayMs(100);

write\_lcd\_command(0x80); //set address counter value

DelayMs(100);

write\_lcd\_data('M');

DelayMs(100);

write\_lcd\_data('I');

DelayMs(100);

write\_lcd\_data('C');

DelayMs(100);

write\_lcd\_data('R');

DelayMs(100);

write\_lcd\_data('O');

DelayMs(100);

write\_lcd\_data('C');

DelayMs(100);

write\_lcd\_data('O');

DelayMs(100);

write\_lcd\_data('N');

DelayMs(100);

write\_lcd\_data('T');

DelayMs(100);

write\_lcd\_data('R');

DelayMs(100);

write\_lcd\_data('O');

DelayMs(100);

write\_lcd\_data('L');

DelayMs(100);

write\_lcd\_data('L');

DelayMs(100);

write\_lcd\_data('E');

DelayMs(100);

write\_lcd\_data('R');

DelayMs(100);

}

}