C code for voting machine

SUBMITTED BY :
Philip Manoj-(Roll no.42)
Anjali Nair- (Roll no.13)
Mohammed Mushraf-(Roll no.34)
Akshay A J -(Roll no.04)

```
// Program to make a voting machine using LCD

#include<reg51.h>

#define msec 50

#define lcd_data_str_pin P2

sbit rs = P3^0; //Register

select (RS) pin  sbit rw = P3^1;

//Read write(RW) pin  sbit en

= P3^6; //Enable(EN) pin

sbit ini_pin = P1^0; // Start

voting pin  sbit stop_pin =

P1^5; // Stop voting pin


sbit          candidate_1=P1^1;

//Candidate1                  sbit
```

```
candidate_2=P1^2;

//Candidate2                sbit

candidate_3=P1^3;

//Candidate3                sbit

candidate_4=P1^4;

//Candidate4  int max = 0;

int carry = 0;

int arr[4];


int vote_amt[3],j;

unsigned int vote_1,vote_2,vote_3,vote_4;


void delay(int delay_time) // Time

delay function  {

int j,k;
for(j=0;j<=delay_time;j++)

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

}


void lcd_cmd(unsigned char cmd_addr) //Function to send
```

```
command to LCD  {

lcd_data_str_pin = cmd_addr;

en = 1;

rs = 0;

rw = 0;

delay(1);

en = 0;

return;

}


void lcd_data_str(char str[50]) //Function to send string

{

int p;

for (p=0;str[p]!='\0';p++)

{

 lcd_data_str_pin = str[p];

 rw = 0;

 rs = 1;

 en = 1;

 delay(1);
 en = 0;
```

```
}
return;
}

void lcd_data_int(unsigned int vote) //Function to send 0-9
character values  {
char dig_ctrl_var;
int p;
for (j=2;j>=0;j--)
{
 vote_amt[j]=vote%10;
 vote=vote/10;
}

for (p=0;p<=2;p++)
{
 dig_ctrl_var = vote_amt[p]+48;
 lcd_data_str_pin = dig_ctrl_var;
 rw = 0;
 rs = 1;
```

```
 en = 1;
 delay(1);
 en = 0;
}
return;
}

void vote_count() // Function to count votes
{
while (candidate_1==0 && candidate_2==0 && candidate_3==0
&& candidate_4==0);  if (candidate_1==1)
{
 while (candidate_1 == 1);
 {
 vote_1 = vote_1 + 1;
 }
}

if (candidate_2==1)
{
 while (candidate_2 == 1);
```

```
 {
 vote_2 = vote_2 + 1;
 }
}

if (candidate_3==1)
{
 while (candidate_3 == 1);
 {
 vote_3 = vote_3
+ 1;   }
}

if (candidate_4==1)
{
 while (candidate_4
== 1);   {
 vote_4 = vote_4
+ 1;   }
}
```

```
}

void lcd_ini()
{
 lcd_cmd(0x38);
delay(msec);
lcd_cmd(0x0E);
delay(msec);
lcd_cmd(0x01);
delay(msec);
lcd_cmd(0x81);
delay(msec);
lcd_data_str("Welcome!!!");
delay(100);
lcd_cmd(0x01);
delay(msec);
lcd_cmd(0x80);
delay(msec);
lcd_data_str(
"Press" );
```

```
delay(msec);
lcd_cmd(0x14);
delay(msec);
lcd_data_str("but
ton");
delay(msec);

delay(msec);
lcd_cmd(0xC0);
delay(msec);
lcd_data_str("
to");
delay(msec);
lcd_cmd(0x14);
delay(msec);
lcd_data_str("v
ote");
delay(100);
lcd_cmd(0x01);
```

```
delay(msec);

lcd_cmd(0x80);

delay(msec);
lcd_data_str("

P1");

delay(msec);

lcd_cmd(0x84);

delay(msec);

lcd_data_str("

P2");

delay(msec);

lcd_cmd(0x88);

delay(msec);

lcd_data_str("

P3");

delay(msec);

lcd_cmd(0x8C);

delay(msec);

lcd_data_str("
```

```
P4");
delay(msec);

vote_count();
lcd_cmd(0x01);
delay(msec);
lcd_cmd(0x85);
delay(msec);
lcd_data_str("Th
ank");
delay(msec);
lcd_cmd(0x14);
delay(msec);
lcd_data_str("You!!");
delay(100);
}

void results() // Function to
show results  {
int i;
```

```
carry = 0;
lcd_cmd(0x01);
delay(msec);
lcd_cmd(0x80);
delay(msec);
lcd_data_str("Results");
delay(msec);
lcd_cmd(0x14);
delay(msec);
lcd_data_str("Are");
delay(msec);
lcd_cmd(0x14);
delay(msec);
lcd_data_str("Out");
delay(msec);

 lcd_cmd(0x01);
delay(msec);
lcd_cmd(0x80);
delay(msec);
```

```
lcd_data_str("
P1");
delay(msec);
lcd_cmd(0x84);
delay(msec);
lcd_data_str("
P2");
delay(msec);
lcd_cmd(0x88);
delay(msec);
lcd_data_str("
P3");
delay(msec);
lcd_cmd(0x8C);
delay(msec);
lcd_data_str("
P4");
delay(msec);
```

```
lcd_cmd(0xC0);
delay(100);
lcd_data_int(vot
e_1);
delay(msec);

lcd_cmd(0xC4);
delay(msec);
lcd_data_int(vot
e_2);
delay(msec);
lcd_cmd(0xC8);
delay(msec);
lcd_data_int(vote_3);
delay(msec);

lcd_cmd(0xCC);
delay(msec);
lcd_data_int(vote_4);
```

```
delay(300);

arr[0] = vote_1;
arr[1] = vote_2;
arr[2] = vote_3;
arr[3] = vote_4;

for( i=0; i<4; i++)
{
 if(arr[i]>=max)
 max = arr[i];
}

if ( (vote_1 == max) && ( vote_2 != max) && (vote_3 !=
max)&& (vote_4 != max) )  {
 carry = 1;
 lcd_cmd(0x01);
 delay(msec);
 lcd_cmd(0x82);
 delay(msec);
 lcd_data_str("Hurray!!!");
```

```
delay(50);
lcd_cmd(0xC4);
delay(msec);
lcd_data_str("P1");
delay(msec);
lcd_cmd(0x14);
delay(msec);
lcd_data_str("wins");
delay(msec);
}

if ( (vote_2 == max) && ( vote_1 != max) && (vote_3 !=
max)&& (vote_4 != max) )  {
carry = 1;
lcd_cmd(0x01);
delay(msec);
lcd_cmd(0x82);
delay(msec);
lcd_data_str("Hurray!!!");
delay(50);
lcd_cmd(0xC4);
```

```
 delay(msec);
 lcd_data_str("P2");
 delay(msec);
 lcd_cmd(0x14);
 delay(msec);
 lcd_data_str("wins");
 delay(msec);
}

if ( (vote_3 == max) && ( vote_2 != max) && (vote_1 !=
max)&& (vote_4 != max) )  {
 carry = 1;
 lcd_cmd(0x01);
 delay(msec);
 lcd_cmd(0x82);
 delay(msec);
 lcd_data_str("Hurray!!!");
 delay(50);
 lcd_cmd(0xC4);
 delay(msec);
```

```
lcd_data_str("P3");
delay(msec);
lcd_cmd(0x14);
delay(msec);
lcd_data_str("wins");
delay(msec);
}

if ( (vote_4 == max) && ( vote_2 != max) && (vote_3 !=
max)&& (vote_1 != max) )  {
carry = 1;
lcd_cmd(0x01);
delay(msec);
lcd_cmd(0x82);
delay(msec);
lcd_data_str("Hurray!!!");
delay(50);
lcd_cmd(0xC4);
delay(msec);
lcd_data_str("P4");
```

```
 delay(msec);
 lcd_cmd(0x14);
 delay(msec);
 lcd_data_str("wins");
 delay(msec);
}

if (carry==0)
{
 lcd_cmd(0x01);
 delay(msec);
 lcd_cmd(0x82);
 delay(msec);

lcd_data_str("cla
sh");   delay(50);
 lcd_cmd(0x14);
 delay(msec);

lcd_data_str("betwee
```

```
n!!!");   delay(50);
 if(vote_2 == max)
 {
 lcd_cmd(0xC5);
 lcd_data_str("P2");
 delay(50);
 }
 if(vote_3 == max)
 {
 lcd_cmd(0xC9);
 lcd_data_str("P3");
 delay(50);
 }
 if(vote_4 == max)
 {
 lcd_cmd(0xCD);
 lcd_data_str("P4");
 delay(50);
 }
}
}
```

```
void main()
{
ini_pin = stop_pin = 1;
vote_1 = vote_2 = vote_3 = vote_4 = 0;
candidate_1 = candidate_2 = candidate_3 =
candidate_4 = 0;  lcd_cmd(0x38);
delay(msec);
lcd_cmd(0x0E);
delay(msec);
lcd_cmd(0x01);
delay(msec);
lcd_cmd(0x80);
delay(msec);
lcd_data_str( "Press" );
delay(msec);
lcd_cmd(0x14);
delay(msec);
lcd_data_str("init");
delay(msec);
delay(msec);
```

```
lcd_cmd(0xC0);
delay(msec);
lcd_data_str("
to");
delay(msec);
lcd_cmd(0x14);
delay(msec);
lcd_data_str("be
gin");
delay(100);
while(1)
{
 while(ini_pin
!= 0)   {
 if (stop_pin
== 0)   break;
 }
 if (stop_pin
```

```
== 0)   {

 break;

 }

 lcd_ini();

}


while(1)
{

results

();  }

}
```