1.led blinking

Circuit.



Mikro c code

simple

void main() {

Trisb=0x00;

portb=0x00;

portb.f0=0xff;

delay\_ms(2000);

portb.f0=0x00;

}

////////////////////////////

For lop

void main() {

int i;

Trisb=0x00;

portb=0x00;

for (i=0;i<50;i++)

{

portb.f0=0xff;

delay\_ms(2000);

portb.f0=0;

delay\_ms(1000);

}

}

Circuit:



Code : button

void main() {

int i, bt\_zero = 0;

Trisb = 0x00;

trisc = 0x01; // Set port C bit 0 as input

portb = 0x00;

portc = 0x00;

while (1) {

if (portc.f0 == 1) {

delay\_ms(150);

if (portc.f0 == 1) {

bt\_zero++;

if (bt\_zero == 10) {

bt\_zero = 0;

}

}

}

if (portc.f0 == 1) {

for (i = 0; i < 50; i++) {

portb.f0 = 1;

delay\_ms(1000);

portb.f0 = 0;

delay\_ms(1000); // LED on for 1 second, off for 1 second

}

}

}

}

2.seven segment display

Circuit



Code.

void main()

{

trisb=0x00;

portb=0xff;

}

Counting 0-9

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

char arrayca[]={0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};

void main()

{

int i=0;

trisb=0x00;

portb=0x00;

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

{

portb=arraycc[i];

delay\_ms(500);

}

}

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

char arrayca[]={0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};

void main()

{

int i=0;

trisb=0x00;

portb=0x00;

for(i=9;i>=0;i--)

{

portb=arraycc[i];

delay\_ms(500);

}

}

1 digit 7 segment display cc



Code

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

char arrayca[]={0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};

void main()

{

int i = 0;

int bt\_zero = 0;

TRISB = 0x00;

TRISD = 0x01;

PORTB = 0x3F;

PORTD = 0x00;

while (1) // Infinite loop to keep checking the button

{

if (PORTD.F0 == 1) // Check if the button is pressed

{

if (PORTD.F0 == 1) // Check again after debounce

{

bt\_zero++;

if (bt\_zero == 10)

{

bt\_zero = 0;

}

}

}

if (PORTD.F0 == 1) // Check if the button is pressed

{

PORTB = arraycc[bt\_zero]; // Display the current value of bt\_zero

delay\_ms(400);//display time

}

}

}

For ca display



Code.

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

char arrayca[] = {0xC0, 0xF9, 0xA4, 0xB0, 0x99, 0x92, 0x82, 0xF8, 0x80, 0x90};

void main()

{

int i = 0;

int bt\_zero = 0;

TRISB = 0x00;

TRISD = 0x01;

PORTB = 0xC0; // Initialize to display '0' on common anode display

PORTD = 0x00;

while (1) // Infinite loop to keep checking the button

{

if (PORTD.F0 == 1) // Check if the button is pressed

{

if (PORTD.F0 == 1) // Check again after debounce

{

bt\_zero++;

if (bt\_zero == 10)

{

bt\_zero = 0;

}

}

}

if (PORTD.F0 == 1) // Check if the button is pressed

{

PORTB = arrayca[bt\_zero]; // Display the current value of bt\_zero

delay\_ms(400); // Display time

}

}

}

Two digit display.



Code for counting 0-99

char arraCC[] = { 0xBF, 0x86, 0xDB, 0xCF, 0xE6, 0xED, 0xFD, 0x87, 0xFF, 0xEF };

void main()

{

int mod = 0, res = 0, i = 0, k = 0;

// Set port directions for controlling the display

TRISB = 0x00;

TRISC = 0x00;

TRISD = 0x00;

// Initialize port values

portb = 0x00;

portc = 0x00;

portd = 0x00;

while(1)

{

// Loop through numbers from 0 to 99

for(i = 0; i <= 99; i++)

{

res = i / 10; // Calculate tens digit

mod = i % 10; // Calculate units digit

// Display each digit for a brief delay

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

{

portc.f0 = 0x00; // Activate power for left digit

portb = arraCC[res]; // Set segment data for tens digit

delay\_ms(10); // Delay

portc.f0 = 0xff; // Deactivate power for left digit

portc.f1 = 0x00; // Activate power for right digit

portb = arraCC[mod]; // Set segment data for units digit

delay\_ms(10); // Delay

portc.f1 = 0xff; // Deactivate power for right digit

}

}

}

}

Two digit seven segment display common anode counting 99-0



Code

char arrayca[] = { 0x40, 0x79, 0x24, 0x30, 0x19, 0x12, 0x02, 0x78, 0x00, 0x10 };

void main() {

int k=0,i,result=0,mod=0;

//Set port direction to control the display

TrisB=0x00;

TrisC=0x00;

//initalize the port

PortB=0x00;

portc=0x00;

while(1)

{

for(i=99;i>=0;i--) //counting 99-0

{

result=i/10;

mod=i%10;

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

{

portc.f0=1;

portb=arrayca[result];

delay\_ms(10);

portc.f0=0;

portc.f1=1;

portb=arrayca[mod];

delay\_ms(10);

portc.f1=0;

}

}

}

}

Button



Code

char arraycc[] = { 0xBF, 0x86, 0xDB, 0xCF, 0xE6, 0xED, 0xFD, 0x87, 0xFF, 0xEF };

//char arrayca[] = { 0x40, 0x79, 0x24, 0x30, 0x19, 0x12, 0x02, 0x78, 0x00, 0x10 };

void main() {

int i;

int bt\_zero = 0, bt\_one = 0, input\_value,res,mod,k;

TRISB = 0x00;

TRISC = 0x00;

TRISD.f0 = 0xff;

TRISD.f1 = 0xff;

TRISD.f2 = 0xff;

TRISD.f3 = 0xff;

TRISD.f4 = 0x00;

portb = 0x00;

portc = 0x00;

portd.f4=0x00;

while(1)

{

if(portd.f1 == 0xff) // click initialize

{

delay\_ms(150);

if(portd.f1 == 0xff) // click stability check

{

bt\_zero++; // digit increment

if(bt\_zero == 10) // after 9, next is 0

{

bt\_zero = 0;

}

}

}

if(portd.f0 == 0xff)

{

delay\_ms(150);

if(portd.f0 == 0xff)

{

bt\_one++;

if(bt\_one == 10)

{

bt\_one = 0;

}

}

}

portc.f0 = 0x00; // active power for digit left

portb = arrayCC[bt\_zero]; // provide data for left digit

delay\_ms(10);

portc.f0 = 0xff; // Deactive power for digit left

portc.f1 = 0x00; // active power for second digit

portb = arrayCC[bt\_one]; // provide data for second digit

delay\_ms(10);

portc.f1 = 0xff; // deactive power for second digit

}

}

Button use for 2 digits seven segment display Ca common anode



Code

char arrayca[] = { 0x40, 0x79, 0x24, 0x30, 0x19, 0x12, 0x02, 0x78, 0x00, 0x10 };

void main() {

int bt\_zero = 0, bt\_one = 0;

// Set port direction to control the display

TrisB = 0x00;

TrisC = 0x00;

TrisD = 0x03; // Set D0 and D1 as input;

// Initialize the ports

PortB = 0x00;

PortC = 0x00;

PortD = 0x00;

while (1) {

if (PortD.F0 == 1) // Button first

{

Delay\_ms(150);

if (PortD.F0 == 1) {

bt\_zero++;

if (bt\_zero == 10) {

bt\_zero = 0;

}

}

}

if (PortD.F1 == 1) // Button second

{

Delay\_ms(150);

if (PortD.F1 == 1) {

bt\_one++;

if (bt\_one == 10) {

bt\_one = 0;

}

}

}

// Display first digit on the seven-segment

PortC.F0 = 1;

PortB = arrayca[bt\_zero];

Delay\_ms(10);

PortC.F0 = 0;

// Display second digit on the seven-segment

PortC.F1 = 1;

PortB = arrayca[bt\_one];

Delay\_ms(10);

PortC.F1 = 0;

}

}

Four digit CA display counting 0-9999



Code

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

char arrayca[]={0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};

void main() {

int d\_zero,d\_one,d\_two,d\_three,i,j;

Trisb=0x00;

Trisc=0x00;

portb=0x00;

portc=0x00;

while(1)

{

for(i=0;i<=9999;i++) //counting 0-9999

{

d\_zero=i/1000; // for first digit of display

d\_one=((i/100)%10); // for second digit

d\_two=((i/10)%10); // for third digit

d\_three=i%10;

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

{

portc.f0=1;

portb=arrayca[d\_zero];//display

delay\_ms(10);

portc.f0=0;

portc.f1=1;

portb=arrayca[d\_one];//display

delay\_ms(10);

portc.f1=0;

portc.f2=1;

portb=arrayca[d\_two];//display

delay\_ms(10);

portc.f2=0;

portc.f3=1;

portb=arrayca[d\_three];//display

delay\_ms(10);

portc.f3=0;

}

}

}

}

Four digit seven segment CC display



Code

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

char arrayca[]={0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};

void main() {

int d\_zero,d\_one,d\_two,d\_three,i,j;

Trisb=0x00;

Trisc=0x00;

portb=0x00;

portc=0x00;

while(1)

{

for(i=0;i<=9999;i++) //counting 0-9999

{

d\_zero=i/1000; // for first digit of display

d\_one=((i/100)%10); // for second digit

d\_two=((i/10)%10); // for third digit

d\_three=i%10;

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

{

portc.f0=0;

portb=arraycc[d\_zero];//display

delay\_ms(10);

portc.f0=1;

portc.f1=0;

portb=arraycc[d\_one];//display

delay\_ms(10);

portc.f1=1;

portc.f2=0;

portb=arraycc[d\_two];//display

delay\_ms(10);

portc.f2=1;

portc.f3=0;

portb=arraycc[d\_three];//display

delay\_ms(10);

portc.f3=1;

}

}

}

}

4 digit cc button



Code

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

char arrayca[]={0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};

void main() {

int d\_zero,d\_one,d\_two,d\_three,i,j;

int bt\_zero=0,bt\_one=0,bt\_two=0,bt\_three=0;

Trisb=0x00;

Trisc=0x00;

Trisd=1;

portb=0x00;

portc=0x00;

while(1)

{

if(portd.f0==0)

{

delay\_ms(150);

if(portd.f0==0)

{

bt\_zero++;

if(bt\_zero==10)

{

bt\_zero=0;

}

}

}

if(portd.f1==0)

{

delay\_ms(150);

if(portd.f1==0)

{

bt\_one++;

if(bt\_one==10)

{

bt\_one=0;

}

}

}

if(portd.f2==0)

{

delay\_ms(150);

if(portd.f2==0)

{

bt\_two++;

if(bt\_two==10)

{

bt\_two=0;

}

}

}

if(portd.f3==0)

{

delay\_ms(150);

if(portd.f3==0)

{

bt\_three++;

if(bt\_three==10)

{

bt\_three=0;

}

}

}

portc.f0=0;

portb=arraycc[bt\_zero];//display

delay\_ms(10);

portc.f0=1;

portc.f1=0;

portb=arraycc[bt\_one];//display

delay\_ms(10);

portc.f1=1;

portc.f2=0;

portb=arraycc[bt\_two];//display

delay\_ms(10);

portc.f2=1;

portc.f3=0;

portb=arraycc[bt\_three];//display

delay\_ms(10);

portc.f3=1;

}

}



Code

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

int cnt = 0; // Initialize cnt to 0

int bt\_zero = 0;

int bt\_one = 0;

int bt\_two = 0;

int bt\_three = 0;

int button\_response(int bt) {

if (portd.f0 == 0xff) // click initialize

{

delay\_ms(100);

if (portd.f0 == 0xff) // click stability check

{

bt++; // digit increment

if (bt == 10) // after 9, next is 0

{

bt = 0;

}

}

}

return bt;

}

void button\_click() {

if (portd.f1 == 0xff)

{

delay\_ms(100);

if (portd.f1 == 0xff)

{

cnt++;

if (cnt == 4)

{

cnt = 0;

}

}

}

switch (cnt)

{

case 0:

portd.f4 = 0; // active power for digit left

portb = 0x7F; // provide data for left digit

delay\_ms(10);

portd.f4 = 1; // Deactive power for digit left

bt\_zero = button\_response(bt\_zero);

break;

case 1:

portd.f5 = 0xff; // active power for second digit

portb = 0x7F; // provide data for second digit

delay\_ms(10);

portd.f5 = 0x00; // deactive power for second digit

bt\_one = button\_response(bt\_one);

break;

case 2:

portd.f6 = 0xff; // active power for third digit

portb = 0x7F; // provide data for third digit

delay\_ms(10);

portd.f6 = 0x00; // Deactive power for third digit

bt\_two = button\_response(bt\_two);

break;

case 3:

portd.f7 = 0xff; // active power for digit right

portb = 0x7F; // provide data for right digit

delay\_ms(10);

portd.f7 = 0x00; // Deactive power for digit right

bt\_three = button\_response(bt\_three);

break;

}

}

void main() {

Trisb=0x00;

Trisc=0x00;

Trisd=0xff;

portb=0x00;

portc=0x00;

while (1) {

button\_click();

portc.f0=0;

portb=arraycc[bt\_zero];

delay\_ms(10);

portc.f0=1;

portc.f1=0;

portb=arraycc[bt\_one];

delay\_ms(10);

portc.f1=1;

portc.f2=0;

portb=arraycc[bt\_two];

delay\_ms(10);

portc.f2=1;

portc.f3=0;

portb=arraycc[bt\_three];

delay\_ms(10);

portc.f3=1;

}

}

4.LcD display



Code:

// LCD module connections

sbit LCD\_RS at RB2\_bit;

sbit LCD\_EN at RB3\_bit;

sbit LCD\_D4 at RB4\_bit;

sbit LCD\_D5 at RB5\_bit;

sbit LCD\_D6 at RB6\_bit;

sbit LCD\_D7 at RB7\_bit;

sbit LCD\_RS\_Direction at TRISB2\_bit;

sbit LCD\_EN\_Direction at TRISB3\_bit;

sbit LCD\_D4\_Direction at TRISB4\_bit;

sbit LCD\_D5\_Direction at TRISB5\_bit;

sbit LCD\_D6\_Direction at TRISB6\_bit;

sbit LCD\_D7\_Direction at TRISB7\_bit;

// End LCD module connections

char a[] = "Hey This is AL AMIN ";

char b[] = "The Student of ICE,PUST";

void main() {

int i, j;

Lcd\_Init(); // Initialize LCD

Lcd\_Cmd(\_LCD\_CLEAR); // Clear display

Lcd\_Out(1, 4, a); // Write text 'Hey This is AL AMIN' in the first row

Lcd\_Out(2, 4, b); //Write the text "The Student of Pabna university of Science and Technology"

delay\_ms(5000);

while (1) {

// Scroll the first string (a)

for (i = 0; i < 100; i++) {

Lcd\_Cmd(\_LCD\_SHIFT\_LEFT);

Delay\_ms(500);

}

Lcd\_Cmd(\_LCD\_CLEAR); // Clear display

// Scroll the second string (b)

for (j = 0; j < 100; j++) { // Adjust the loop limit based on string length

Lcd\_Cmd(\_LCD\_SHIFT\_LEFT);

Delay\_ms(500);

}

Lcd\_Cmd(\_LCD\_CLEAR); // Clear display

}

}

Steper motor

Circuit:



Code:

void main() {

Trisb=0b00000000;//Port b as output;

portb=0b11111111;

do{

Portb=0b00000011;

delay\_ms(500);

portb=0b00000110;

delay\_ms(500);

portb=0b00001100;

delay\_ms(500);

portb=0b00011000;

delay\_ms(500);

}

while(1);//loop excuted infinite

}

Servo



Code

void rotateLeft90();

void rotateZero();

void rotateRight90();

int i;

void main() {

     TRISB = 0x00;

     portb = 0x00;

     while(1)

     {

        rotateLeft90();

        delay\_ms(2000);

        rotateZero();

        delay\_ms(2000);

        rotateRight90();

        delay\_ms(2000);

     }

}

void rotateLeft90()

{

    for(i=0;i<50;i++)

    {

        portb.f0=1;

        delay\_us(800);

        portb.f0=0;

        delay\_us(19200);

    }

}

void rotateZero()

{

    for(i=0;i<50;i++)

    {

        portb.f0=1;

        delay\_us(1500);

        portb.f0=0;

        delay\_us(18500);

    }

}

void rotateRight90()

{

    for(i=0;i<50;i++)

    {

        portb.f0=1;

        delay\_us(2200);

        portb.f0=0;

        delay\_us(17800);

    }

}

Relay control



Code

void main() {

    int bt=0;

    Trisb=0x00; //as output

    Trisc=0xff; // as input

    portb=0x00;

    while(1)

     {

       //button

       if(portc.f0==1)

       {

        delay\_ms(150);

        if(portc.f0==1)

         {

          bt++;

          if(bt==10)

          {

            bt=0;

          }

         }

       }

      if(portc.f0==1)

      {

        portb.f0=1;//turn on relay

       delay\_ms(10000);  // relay on for 10 second

       portb.f0=0;//turn off relay

       delay\_ms(10000);  // off for 10 second

      }

     }

}

Dotmatrix Display



Code

void MSDelay(unsigned char Time)

{

unsigned char y,z;

for(y=0;y<Time;y++);

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

}

void main()

{

TRISC=000;

TRISD=0x00;

while(1) {

PORTD=0x80;

PORTC-0x00;

MSDelay(10);

PORTD = 0x40;

PORTC= 0xff;

MSDelay(10);

PORTD=0x20;

PORTC=0xff;

MSDelay(10);

PORTD=0x10;

PORTC=0x18;

MSDelay(10);

PORTD=0x08;

PORTC=0x18;

MSDelay(10);

PORTD=0x04;

PORTC=0x00;

MSDelay(10);

PORTD=0x02;

PORTC= 0xff;

MSDelay(10);

}

}