

main.c

```
// miAT2
// SETA pruebas xsetaseta@gmail.com

#include <avr/io.h>
#include <stdio.h>

#define Set_Datos          PORTB|=_BV(5)
#define Clear_Datos       PORTB&=~_BV(5)
#define Set_Enable        PORTB|=_BV(4)
#define Clear_Enable      PORTB&=~_BV(4)

#define F_CPU              1000000

void LCD_INI(void);
void LCD_CLS(void);
void LCD_HOME(void);
void LCD_AT(char x);
void Pon4bits(char x);
void ENABLE(void);
void SENDI(char x);
void SENDCHAR(char x);
void SENDCADE(char *x);
void PrintAtNum(unsigned char x,int numero);
void PrintAtStr(unsigned char x,char *string);

void delay_ms(unsigned char time_ms);
void delay_10us(unsigned char time_10us);

char VARI[17]; //Modificar para numeros mayores de 16

int main(void)
{
  uint8_t countval;
  unsigned char voltios;
  int temp;

  PORTC=0x00;
  PORTB=0x00;
  PORTD=0x00;
  DDRD=0xff; //puerto D como salida
  DDRC=0x00; //puerto C como entrada
  DDRB=0xff; //puerto B como salida

  LCD_INI();
  LCD_CLS();
  LCD_HOME();

  ADMUX=0xE0; //2.56V,IZ,ADC0
  ADCSRA=0xA0;

  PrintAtStr(0,"voltmetro 0-1v");
  PrintAtStr(75,"8bits");
  countval = 0;
  while(1)
  {
    delay_ms(250);
    ADCSRA|=_BV(6); // inicio_ADC
                    //no se mira si ha terminado la conversión

    delay_ms(250);
    voltios=ADCH;
    temp=voltios*110;
    temp=temp/256;
    temp=temp*10;

    LCD_AT(64);
    sprintf(VARI,"%0.3d",temp);
    SENDCADE(VARI);
    SENDCADE(" mv");
    PORTD = countval++;
  }
}
```

1

main.c

```
//-----
}

void PrintAtStr(unsigned char x,char *string)
{
  LCD_AT(x);
  SENDCADE(string);
}
void PrintAtNum(unsigned char x,int numero)
{
  LCD_AT(x);
  sprintf(VARI,"%d",numero);
  SENDCADE(VARI);
}
void LCD_INI(void)
{
  delay_ms(250);
  Pon4bits(0x03);
  Clear_Datos;
  ENABLE();
  Set_Datos;
  delay_ms(30);
  Pon4bits(0x03);
  Clear_Datos;
  ENABLE();
  Set_Datos;
  delay_ms(30);
  Pon4bits(0x03);
  Clear_Datos;
  ENABLE();
  Set_Datos;
  delay_ms(30);

  Pon4bits(0x02); // modo 4 bits
  Clear_Datos;
  ENABLE();
  Set_Datos;

  SENDI(0x2c); //modo 4 bits, dos lineas
  SENDI(0x0f); // cursor con parpadeo
  SENDI(0x04);
}

void Pon4bits(char x)
{
  char z;
  z=PORTB & 0xf0;
  x= x & 0x0f;
  PORTB= x |z;
}
void LCD_CLS(void)
{
  SENDI(1); //borra pantalla
}
void LCD_HOME(void)
{
  SENDI(2); //cursor al inicio
}
void LCD_AT(char x) //0 Comienzo línea1, 64 Comienzo línea2
{
  x=x | 128;
  SENDI(x);
}
void SENDCADE(char *x)
{
  char z;
  for(z=0;z<33 && *x!=0;z++,x++)
  {SENDCHAR(*x);}
}
void SENDCHAR(char x)
{
  char z;
  z=x >> 4;
}
```

2

```
Pon4bits(z); //4 bytes de mas peso
ENABLE();
Pon4bits(x); //4 bytes de menos peso
ENABLE();
}
//Manda datos de control
void SENDI(char x)
{
    Clear_Datos;
    SENDCHAR(x);
    delay_10us(250); //Tiempos para reformar
    Set_Datos;
}
void ENABLE(void)
{
    Set_Enable;
    delay_10us(250); //Tiempos para reformar
    Clear_Enable;
}

// 4 ciclos*delay*time_10us+5*time_10us
void delay_10us(unsigned char time_10us)
{
    unsigned short delay_count = F_CPU / 400000; //para 1Mhz->2

    unsigned short cnt;
    asm volatile (
        "\n"
        "L_d11%=: \n\t"
        "mov %A0, %A2 \n\t"
        "mov %B0, %B2 \n\t"
        "L_d12%=: \n\t"
        "sbw %A0, 1 \n\t"
        "brne L_d12%= \n\t"
        "dec %1 \n\t"
        "brne L_d11%= \n\t"
        : "=&w" (cnt)
        : "r"(time_10us), "r"((unsigned short) (delay_count))
    );
}

// 4 ciclos*delay*time_ms+5*time_ms
void delay_ms(unsigned char time_ms)
{
    unsigned short delay_count = F_CPU / 4000; //para 1Mhz->250

    unsigned short cnt;
    asm volatile (
        "\n"
        "L_d11%=: \n\t"
        "mov %A0, %A2 \n\t"
        "mov %B0, %B2 \n\t"
        "L_d12%=: \n\t"
        "sbw %A0, 1 \n\t"
        "brne L_d12%= \n\t"
        "dec %1 \n\t"
        "brne L_d11%= \n\t"
        : "=&w" (cnt)
        : "r"(time_ms), "r"((unsigned short) (delay_count))
    );
}
}
```