

7mo7ma

# Electrónica Digital IV

## Trabajo Práctico N°1

---

### Actividad:

#### TRABAJO PRÁCTICO N°1

##### Tanteador de fútbol

/\* Tanteador para fútbol con display de 7 segmentos, incrementos, decrementos y reset usando pulsadores

Placa Arduino UNO

Dos displays para el marcador de localía y dos para el marcador de visitas

Programa que utilice el puerto D del Arduino Uno para los 7 segmentos del display

Toda rutina que se deba repetir en el programa, realizarla en una función

Los segmentos conectarlos desde el pin 0 del portD para el segmento a (Pin de salida 0 de la placa), hasta el pin 6 del Port D para el segmento g (pin de salida 6 de a placa)

El pulsador de IncrementoLocal al pin 14 (A0), setearlo como pinMode (14, INPUT) o con su nombre dado en el #define si se ha utilizado

El pulsador de DecrementoLocal al pin 15 (A1), setearlo como pinMode (15, INPUT) o con su nombre dado en el #define si se ha utilizado

El pulsador de IncrementoVisita al pin 16 (A2), setearlo como pinMode (16, INPUT) o con su nombre dado en el #define si se ha utilizado

El pulsador de DecrementoVisita al pin 17 (A3). setearlo como pinMode (17, INPUT) o con su nombre dado en el #define si se ha utilizado

Las salidas para los cuatro transistores de multiplexación serán 10, 11, 12, 13

La eliminación de rebotes debe hacerse con una demora multiplexada

Como los pines analógicos que se usan como digitales no tienen pullup, colocarlas por fuera en A0, A1, A2, A3.

Se debe poder resetear las cuentas oprimiendo simultáneamente los dos pulsadores de incremento

\*/

### Resultado:

#### Código:

```
bbyte Digit[] = {0x3F,0x6,0x5B,0x4F,0x66,0x6D,0x7D,0x7,0x7F,0x67};  
int decena = 0;  
int unidad = 0;  
int numero = 0;  
int decena2 = 0;
```

---

---

```
int unidad2 = 0;
int numero2 = 0;
int cantmultiplexados = 0;
#define TransistorDecenas 13 //
#define TransistorUnidades 12 //
#define TransistorUnidades2 11//
#define TransistorDecenas2 10//

void setup()
{
  DDRD=B11111111;
  pinMode(14, INPUT);
  pinMode(15, INPUT);
  pinMode(16, INPUT);
  pinMode(17, INPUT);
  pinMode(TransistorUnidades, OUTPUT);
  pinMode(TransistorDecenas, OUTPUT);
  pinMode(TransistorUnidades2, OUTPUT);
  pinMode(TransistorDecenas2, OUTPUT);
}
void loop(){

  //Reset
  if (digitalRead(14)== 1 & digitalRead(15)== 1 )
  {
    numero = 0;
  }
  if (digitalRead(16)== 1 & digitalRead(17)== 1 )
  {
    numero2 = 0;
  }

  //Incremento
  if (digitalRead(14)== 1)
  {
    numero++;
    Limite ();
  }

  if (digitalRead(16)== 1)
  {
    numero2++;
    Limite ();
  }

  //Decremento
  if (digitalRead(15)== 1)
  {
```

---

```

    numero--;
    Limite ();
}
if (digitalRead(17)== 1)
{
    numero2--;
    Limite ();
}

Parte_2();
}
void Limite ()
{
    if (numero > 99){numero = 0;}
    if (numero < 0){numero = 99;}
    if (numero2 > 99){numero2 = 0;}
    if (numero2 < 0){numero2 = 99;}
}
void Parte_2()
{
    decena = numero / 10 ;
    unidad = numero % 10 ;
    decena2 = numero2 / 10 ;
    unidad2 = numero2 % 10 ;
    for (cantmultiplexados=0; cantmultiplexados < 10; cantmultiplexados++)
    {
        PORTD=Digit [unidad]; //Coloca el valor de la unidad en el puerto
        digitalWrite(TransistorUnidades ,1);
        delay(2); //cuando termina de escribir cada unidad, espera 5 0 10 ms
        para poderla visualizar
        digitalWrite(TransistorUnidades ,0);
        PORTD=Digit [decena]; //Coloca el valor de la decena en el puerto
        digitalWrite(TransistorDecenas,1);
        delay(2); //cuando termina de escribir cada decena, espera 5 0 10 ms
        para poderla visualizar
        digitalWrite(TransistorDecenas,0);
        PORTD=Digit [unidad2];
        digitalWrite(TransistorUnidades2 ,1);
        delay(2);
        digitalWrite(TransistorUnidades2 ,0);
        PORTD=Digit [decena2];
        digitalWrite(TransistorDecenas2,1);
        delay(2);
        digitalWrite(TransistorDecenas2,0);
    }
}

```

---

## Captura de pantalla:

