



INSTITUTO POLITÉCNICO  
NACIONAL  
ESCUELA SUPERIOR DE  
CÓMPUTO



## **PRÁCTICA 8: CRONÓMETRO DE 59.9 SEGUNDOS**

ALUMNO: MALAGON BAEZA ALAN ADRIAN  
MARTINEZ CHAVEZ JORGE ALEXIS

GRUPO: 6CM3

U.A: SISTEMAS EN CHIP

PROFESOR: FERNANDO AGUILAR SÁNCHEZ

FECHA DE ENTREGA: 30 DE ABRIL DE 2023

## OBJETIVO

Al término de la sesión, los integrantes del equipo contarán con la habilidad de realizar un cronómetro de 59.9 segundos.

## INTRODUCCIÓN TEÓRICA

### CRONÓMETRO DIGITAL

La palabra cronómetro proviene de la mitología griega, el nombre se le dio por el Dios griego Cronos que era el Dios del tiempo. Es un reloj o una función de reloj que sirve para medir fracciones de tiempo, normalmente cortos y con exactitud.<sup>1</sup>

Los cronómetros son instrumentos usados para medir intervalo de tiempo, el cual es definido como el lapso de tiempo entre dos eventos.<sup>1</sup>

Un ejemplo de intervalo de tiempo es nuestra edad, cual es solo un lapso de tiempo desde que nacimos. A diferencia de un reloj convencional el indicador muestra el tiempo de cada día en horas, minutos y segundos de una época absoluta un punto de inicio (así como el principio del día o el año), un cronómetro o temporizador simple mide e indica el periodo de tiempo desde un punto de inicio arbitrario.<sup>1</sup>

La unidad del intervalo de tiempo es el segundo (s). El SI define al segundo propiamente del átomo de cesio y por esta razón los osciladores de cesio son considerados así patrones primarios para intervalo de tiempo y frecuencia.<sup>1</sup>

El segundo es la duración de 9 192 631 770 periodos de la radiación asociada a la transición hiperfina del estado base del átomo de Cesio-133.<sup>1</sup>

## RESOLUCIÓN

La resolución de un cronómetro o temporizador representa el periodo de tiempo más pequeño que el instrumento puede medir o indicar. Una resolución común para cronómetros digitales es de 1 ms (0,001 s) o mejores, y para cronómetros analógicos es de 1/5 de segundo, ó 0,2 s.<sup>1</sup>

## BASE DE TIEMPO

La base de tiempo es la que produce la señal de frecuencia usada para los cronómetros o temporizadores para medir el intervalo de tiempo. Hoy en día estos dispositivos por lo general utilizan osciladores de cristal de cuarzo. La frecuencia más común utilizada en los cristales de cuarzo es de 32 768 Hz.<sup>1</sup>

## COMPONENTES DEL CRONÓMETRO

Cada cronómetro está compuesto por cuatro elementos:

- Fuente de poder,
- Base de tiempo,
- Contador
- Un indicador.

El diseño y construcción de cada componente depende del tipo de cronómetro.

Para el cronómetro digital la fuente de poder es usualmente una celda de plata o una batería alcalina que alimenta el oscilador y la circuitería del contador y el indicador. Usualmente la base de tiempo es un oscilador de cristal de cuarzo, con una frecuencia nominal de 32 768 Hz. En la Figura 1 se pueden apreciar los componentes de un cronómetro digital.<sup>1</sup>



*Figura 1. Componentes del cronómetro digital*

## **MATERIALES Y EQUIPO EMPLEADO**

- ✓ CodeVision AVR
- ✓ AVR Studio 4
- ✓ Microcontrolador ATmega 8535
- ✓ 3 Display cátodo común
- ✓ 21 Resistores de 330  $\Omega$  a 1/4 W
- ✓ 3 Push Botón

## DESARROLLO EXPERIMENTAL

1. Diseñe un programa en el que coloque dos Displays, uno en el Puerto A y el otro en el Puerto B y con una terminal del Puerto D detecte la cuenta a través de un par infrarrojo.

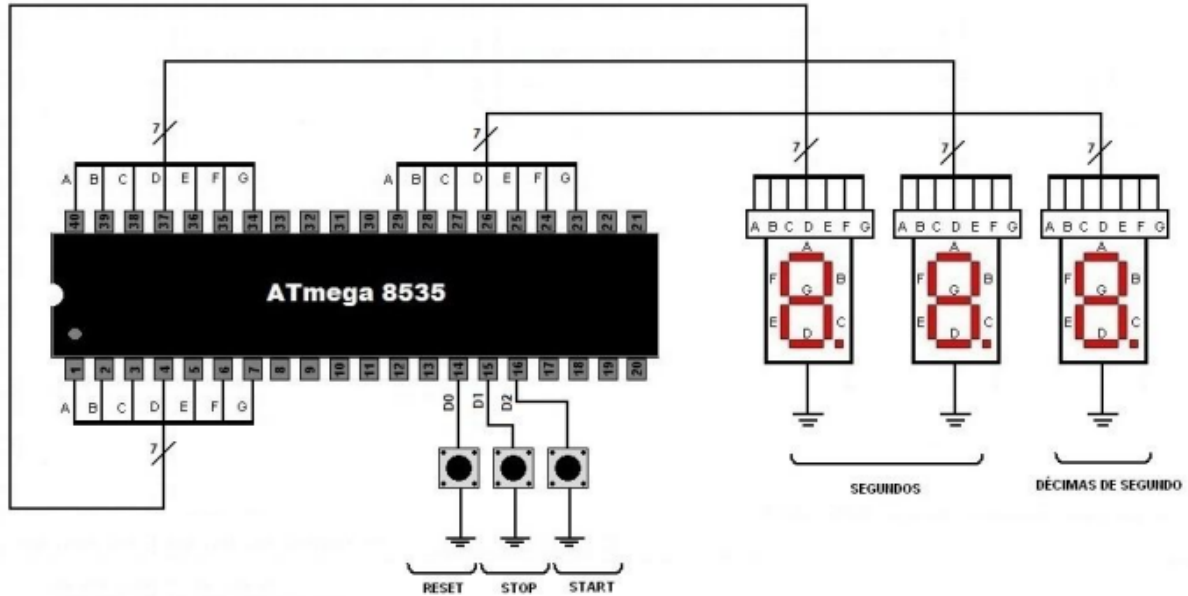


Figura 2. Circuito para el cronómetro

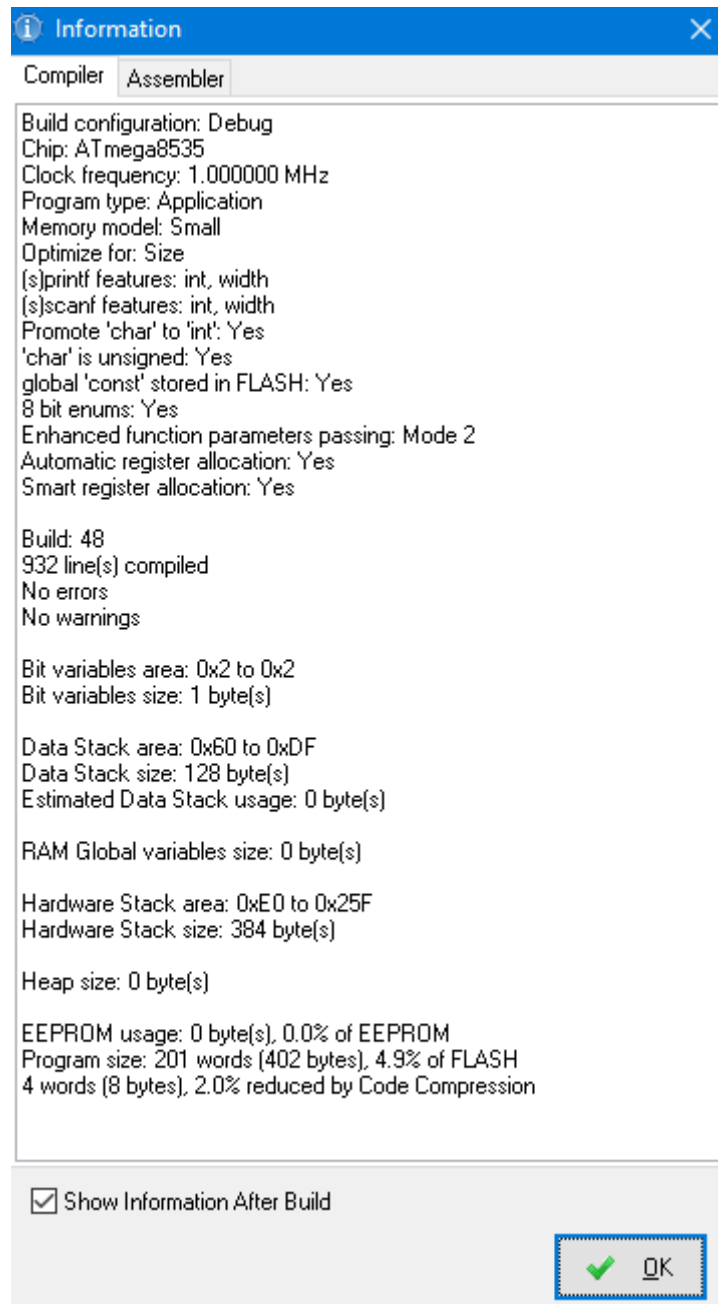


Figura 3. Compilación exitosa en CodeVision

## CÓDIGO GENERADO POR CODEVISION

```

/*****
This program was created by the CodeWizardAVR V3.47
Automatic Program Generator
© Copyright 1998-2021 Pavel Haiduc, HP InfoTech S.R.L.
http://www.hpinfotech.ro

Project :
Version :
Date    :
Author  :

```

Company :  
Comments:

Chip type : ATmega8535  
Program type : Application  
AVR Core Clock frequency: 1.000000 MHz  
Memory model : Small  
External RAM size : 0  
Data Stack size : 128

\*\*\*\*\*/

```
#include <mega8535.h>
#include <delay.h>
#define start PIND.2 //Botón inicio
#define stop PIND.1 //Botón paro
#define reset PIND.0 //Botón reinicio
```

```
bit botona;
bit botonp = 0;
bit conteo = 1; //Bandera para conteo
```

```
const char tabla7segmentos[10] = {0x3f, 0x06, 0x5b, 0x4f, 0x66,
0x6d, 0x7d, 0x07, 0x7f, 0x6f};
unsigned char decenas,unidades,milesimas;
```

```
void main(void)
{
// Declare your local variables here
```

```
// Input/Output Ports initialization
// Port A initialization
// Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=Out
Bit1=Out Bit0=Out
DDRA=(1<<DDA7) | (1<<DDA6) | (1<<DDA5) | (1<<DDA4) | (1<<DDA3) |
(1<<DDA2) | (1<<DDA1) | (1<<DDA0);
// State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=0 Bit1=0 Bit0=0
PORTA=(0<<PORTA7) | (0<<PORTA6) | (0<<PORTA5) | (0<<PORTA4) |
(0<<PORTA3) | (0<<PORTA2) | (0<<PORTA1) | (0<<PORTA0);
```

```
// Port B initialization
// Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=Out
Bit1=Out Bit0=Out
DDRB=(1<<DDB7) | (1<<DDB6) | (1<<DDB5) | (1<<DDB4) | (1<<DDB3) |
(1<<DDB2) | (1<<DDB1) | (1<<DDB0);
// State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=0 Bit1=0 Bit0=0
PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) |
(0<<PORTB3) | (0<<PORTB2) | (0<<PORTB1) | (0<<PORTB0);
```

```

// Port C initialization
// Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=Out
// Bit1=Out Bit0=Out
DDRC=(1<<DDC7) | (1<<DDC6) | (1<<DDC5) | (1<<DDC4) | (1<<DDC3) |
(1<<DDC2) | (1<<DDC1) | (1<<DDC0);
// State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=0 Bit1=0 Bit0=0
PORTC=(0<<PORTC7) | (0<<PORTC6) | (0<<PORTC5) | (0<<PORTC4) |
(0<<PORTC3) | (0<<PORTC2) | (0<<PORTC1) | (0<<PORTC0);

// Port D initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In
// Bit0=In
DDRD=(0<<DDD7) | (0<<DDD6) | (0<<DDD5) | (0<<DDD4) | (0<<DDD3) |
(0<<DDD2) | (0<<DDD1) | (0<<DDD0);
// State: Bit7=P Bit6=P Bit5=P Bit4=P Bit3=P Bit2=P Bit1=P Bit0=P
PORTD=(1<<PORTD7) | (1<<PORTD6) | (1<<PORTD5) | (1<<PORTD4) |
(1<<PORTD3) | (1<<PORTD2) | (1<<PORTD1) | (1<<PORTD0);

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=0xFF
// OC0 output: Disconnected
TCCR0=(0<<WGM00) | (0<<COM01) | (0<<COM00) | (0<<WGM01) | (0<<CS02)
| (0<<CS01) | (0<<CS00);
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Disconnected
// OC1B output: Disconnected
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=(0<<COM1A1) | (0<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) |
(0<<WGM11) | (0<<WGM10);
TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (0<<WGM12) | (0<<CS12)
| (0<<CS11) | (0<<CS10);
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;

```

```

OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=0xFF
// OC2 output: Disconnected
ASSR=0<<AS2;
TCCR2=(0<<WGM20) | (0<<COM21) | (0<<COM20) | (0<<WGM21) | (0<<CS22)
| (0<<CS21) | (0<<CS20);
TCNT2=0x00;
OCR2=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) |
(0<<OCIE1B) | (0<<TOIE1) | (0<<OCIE0) | (0<<TOIE0);

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=(0<<ISC11) | (0<<ISC10) | (0<<ISC01) | (0<<ISC00);
MCUCSR=(0<<ISC2);

// USART initialization
// USART disabled
UCSRB=(0<<RXCIE) | (0<<TXCIE) | (0<<UDRIE) | (0<<RXEN) | (0<<TXEN) |
(0<<UCS22) | (0<<RXB8) | (0<<TXB8);

// Analog Comparator initialization
// Analog Comparator: Off
// The Analog Comparator's positive input is
// connected to the AIN0 pin
// The Analog Comparator's negative input is
// connected to the AIN1 pin
ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) |
(0<<ACIC) | (0<<ACIS1) | (0<<ACIS0);
SFIOR=(0<<ACME);

// ADC initialization
// ADC disabled
ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADATE) | (0<<ADIF) | (0<<ADIE) |
(0<<ADPS2) | (0<<ADPS1) | (0<<ADPS0);

// SPI initialization
// SPI disabled
SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) |

```



```

(0<<CPHA) | (0<<SPR1) | (0<<SPR0);

// TWI initialization
// TWI disabled
TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);

while (1)
{
    //Condiciones de los botones start, stop y reset
    if(start == 0){ //No se inicia el cronómetro
        botona = 0;
    }
    else if(stop == 0){ //No se detiene el cronómetro
        botona = 1;
        conteo = 0;
    }
    else if(reset == 0){ //No se reinicia el cronómetro
        botona = 1;
        conteo = 1;
    }
    else{
        botona = botonp;
    }

    //Funcionamiento del cronómetro
    if(botona == 0){ //START
        if(milesimas == 9){
            if(unidades == 9){
                decenas++;
                unidades = 0;
                PORTA.7 = 1; //Punto del display de unidades
                if(decenas == 6 && unidades == 0){
                    decenas = 0;
                    unidades = 0;
                }
            }
            else{
                unidades++;
                PORTA.7 = 1;
            }
            milesimas = 0;
            delay_ms(100); //Visualización de la información en
Los displays
        }
        else{
            milesimas++;
            delay_ms(100);
        }
    }
}

```

```

else{           //STOP
    if(conteo == 0){
        unidades = unidades;
        decenas = decenas;
        milesimas = milesimas;
        PORTA.7 = 1;
    }
    else{       //RESET
        decenas = 0;
        unidades = 0;
        milesimas = 0;
    }
}
PORTA = tabla7segmentos[unidades];
PORTB = tabla7segmentos[decenas];
PORTC = tabla7segmentos[milesimas];
botonp = botona;
}
}

```

## CIRCUITO ELECTRICO EN PROTEUS

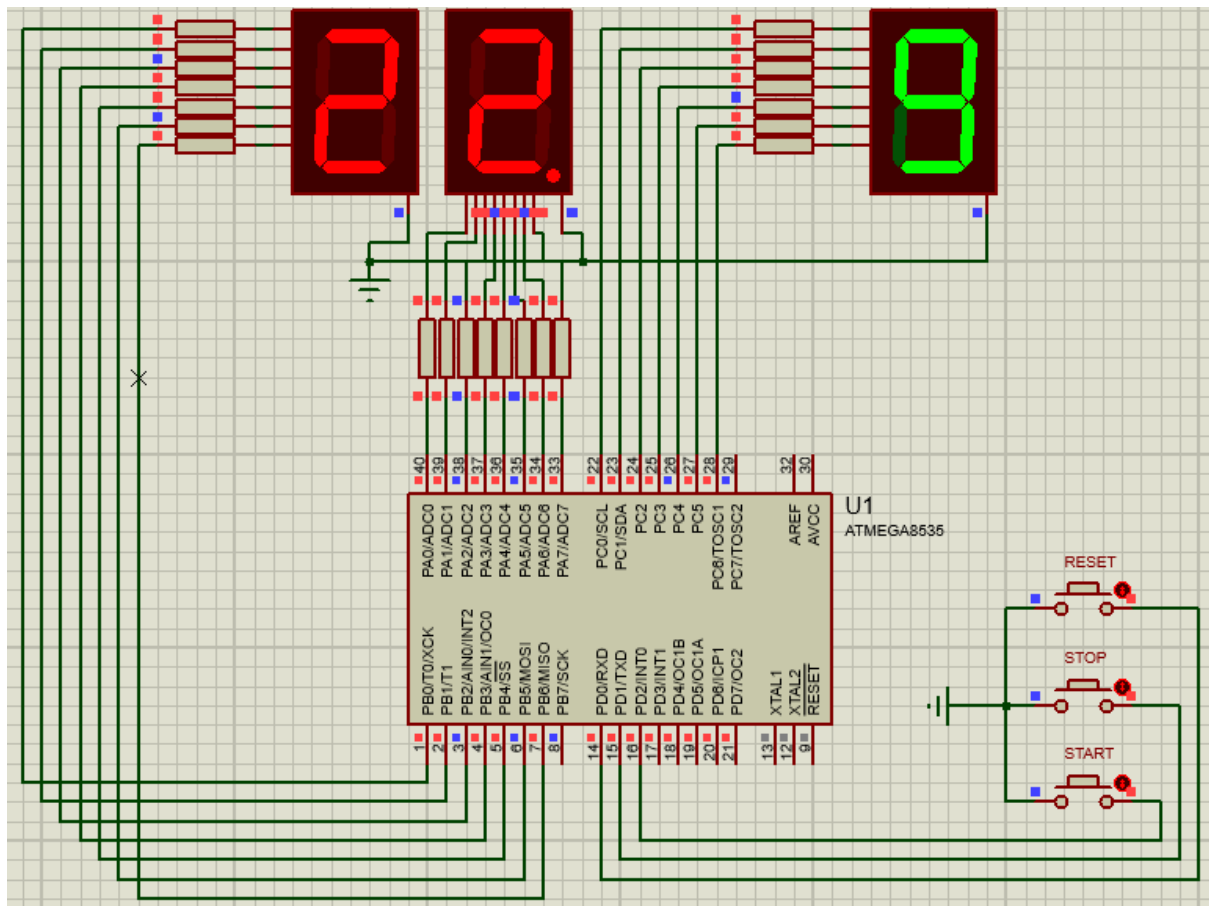
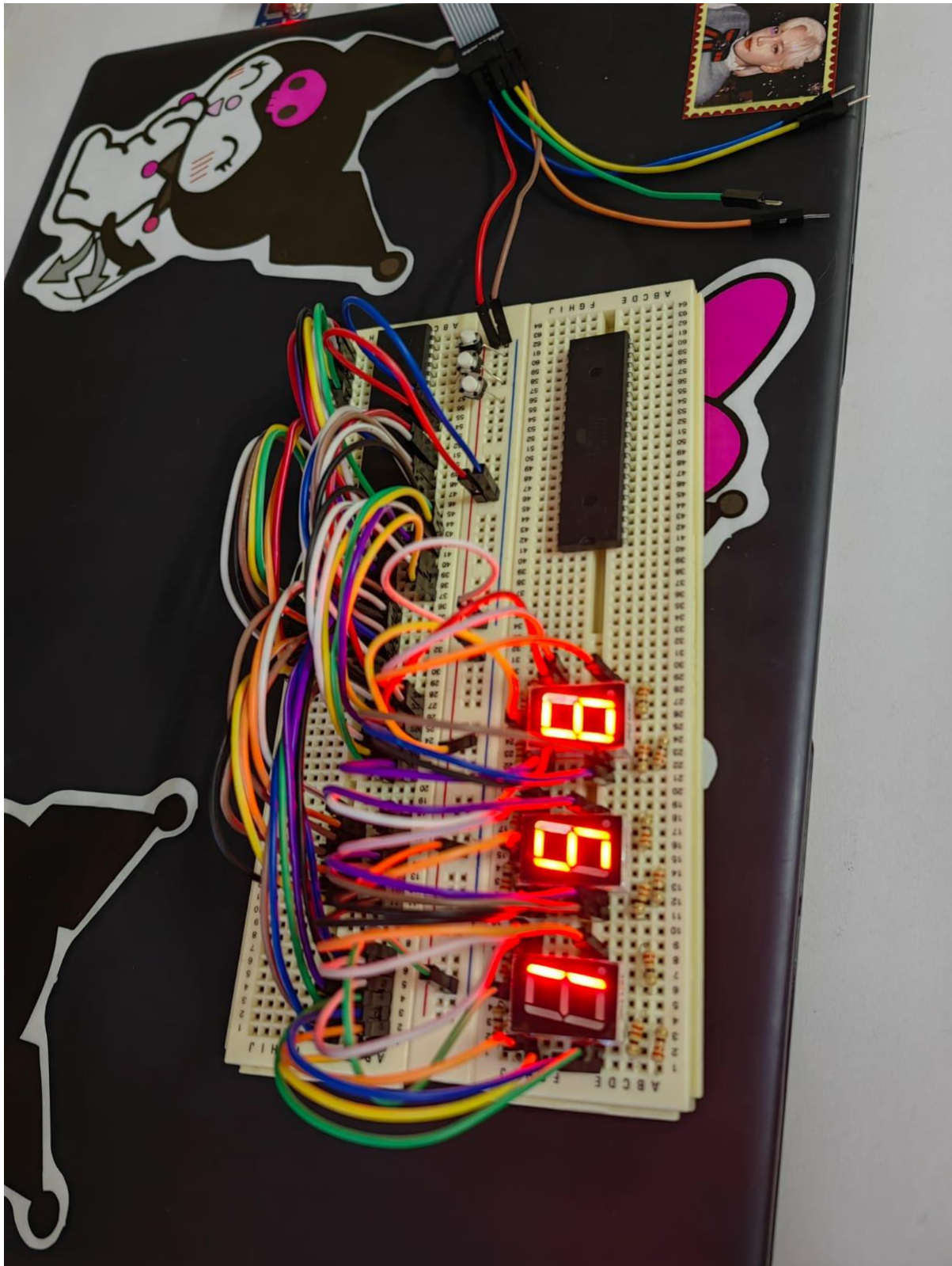


Figura 4. Circuito simulado en Proteus

## CIRCUITO EN EL PROTO ARMADO



# OBSERVACIONES Y CONCLUSIONES INDIVIDUALES

Malagón Baeza Alan Adrián

El cronómetro digital es un dispositivo muy utilizado para medir un intervalo de tiempo desde el segundo 0 hasta el fin de un suceso que se desee medir. En los deportes es utilizado para medir el intervalo de tiempo que tarda el atleta en completar su ejercicio con el fin de establecer marcas a romper para competiciones.

El cronómetro digital existe como un dispositivo físico que se puede adquirir en distintas tiendas, el cuál consta en su aspecto físico de al menos un botón y un display en dónde se muestra el conteo. Sin embargo, es posible implementar un cronómetro digital a través del uso de un microcontrolador, pulsadores y displays.

Para poder realizar el cronómetro se deben de colocar la cantidad de displays que sean necesarios para la cuenta que deseamos llevar. Como el microcontrolador que utilizamos sólo cuanta con 4 puertos configurables, sólo podemos usar 3 de estos para conectar displays y 1 para conectar los pulsadores para el funcionamiento del cronómetro.

Martínez Chávez Jorge Alexis

Al tener 3 puertos disponibles podemos implementar un cronómetro de 2 dígitos y un 3er dígito para la cuenta de los milisegundos. Para poder identificar los milisegundos se implementó un display de color verde a diferencia de los 2 displays rojos para el conteo de las décimas y unidades. Además, en el display de las unidades se optó por usar un display con punto decimal para poder visualizar el punto que separa las unidades de las milésimas.

Para tener un funcionamiento correcto del cronómetro se tuvo que establecer una cuenta del 0-9 para los milisegundos y para las unidades, mientras que para las decenas se tuvo que realizar una cuenta del 0-6 debido a que en el sistema de segundos sólo se pueden contar hasta 60 segundos (59.9 segundos en la práctica) para pasar al siguiente dígito que representan los minutos.

## BIBLIOGRAFÍA

[1] F. J. Jiménez Tapia, "Funcionamiento y tipos de Cronómetros," [www.cenam.mx](https://www.cenam.mx/eventos/enme/docs/38%20Funcionamiento%20y%20Tipos%20de%20Cronometros%20).  
<https://www.cenam.mx/eventos/enme/docs/38%20Funcionamiento%20y%20Tipos%20de%20Cronometros%20> (accessed Mar. 05, 2022).