

## PRACTICA N° 6

### TEMPORIZADORES POR DESBORDAMIENTO DE UN MICROCONTROLADOR ATMEGA328P

---

#### OBJETIVOS.

HABILITAR Y USAR LOS TEMPORIZADORES 0 Y 2 MEDIANTE DESBORDAMIENTO

#### INTRODUCCION

Un microcontrolador de 8 bits cuenta con dos temporizadores de 8 (T0 y T2) y uno de 16 bits (T1), este temporizador está compuesto por una unidad de comparación de ocho bits. Esta unidad de comparación se dedica a contar eventos, estos pueden ser generados internamente por el reloj del sistema vía el preescalador o por una señal de reloj externa aplicada a la terminal T0 (PB0).

Cuando el registro TCNT0 llega a su cuenta máxima, que es 0xFF en hexadecimal o 255 en decimal, al siguiente evento a contar se desborda y se pondrá la cuenta de este registro en ceros, ese evento de desbordamiento es el que genera una llamada de interrupción por desbordamiento de Temporizador y a su vez hará que se active la bandera de desbordamiento en el registro TIFR, la interrupción será atendida inmediatamente por el CPU y saltará a la sección del código que el usuario haya designado cada vez que exista un desbordamiento.

Los desbordamientos cuando están regidos por el reloj del microcontrolador pueden dar distintos tiempos para que se desborde el registro esto dependerá de la frecuencia del oscilador del sistema, por ejemplo si se emplea un cristal de 4 MHz el ciclo máquina (*CM*) será de:

$$CM = \frac{1}{4000000} = 0.25\mu s$$

De acuerdo a esto, el preescalador cuando es igual al ciclo máquina incrementará uno cada 0.25μs y si se hace la cuenta máxima que es de 256 cuentas el tiempo máximo que contara antes de desbordarse será de 64 μs en cambio si el preescalador está en 1024 quiere decir que pasaran 1024 ciclos máquina para incrementar un uno, y suponiendo que se realiza la máxima cuenta el tiempo máximo para el desbordamiento será de 65.5 ms. El resumen de estos tiempos se muestra en la siguiente tabla (5.1).

**TABLA 5.1 TIEMPOS DE LA FUENTE DE RELOJ PARA EL TEMPORIZADOR**

Preescala (Pscla)	$Pscla \times CM \times \text{Numero de Cuentas} = TM$	Tiempo Maximo (TM)
<b>1:1</b>	$1 \times 0.25\mu s \times 256$	$64\mu s$
<b>1:8</b>	$8 \times 0.25\mu s \times 256$	$512\mu s$
<b>1:64</b>	$64 \times 0.25\mu s \times 256$	$4.096ms$
<b>1:256</b>	$256 \times 0.25\mu s \times 256$	$16.384ms$
<b>1:1024</b>	$1024 \times 0.25\mu s \times 256$	$65.536ms$

### Registro e Control del Temporizador/Contador (TCCR0B)

BIT							
7	6	5	4	3	2	1	0
FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00

Los Bits CS02, CS01 y CS00 se emplean para seleccionar el reloj que emplea el temporizador para su cuenta, es decir es la pre escala que se le da al reloj en la tabla 5.2

**TABLA 5.2 TIEMPOS DE LA FUENTE DE RELOJ PARA EL TEMPORIZADOR**

CS02	CS01	CS00	Descripción
0	0	0	Ningún fuente de reloj (Temporizador detenido)
0	0	1	Sin pre escalador 1:1
0	1	0	Pre escalador a 1:8
0	1	1	Pre escalador a 1:64
1	0	0	Pre escalador a 1:256
1	0	1	Pre escalador a 1:1024
1	1	0	Fuente externa de reloj en T0 - Reloj en Flanco de bajada
1	1	1	Fuente externa de reloj en T0 - Reloj en Flanco de subida

### Registro de la Máscara de Interrupción del Temporizador/Contador(TIMSK0)

BIT							
7	6	5	4	3	2	1	0
-	-	-	-	-	OCIE0B	OCIE0A	TOIE0

El bit 6 **TOIE2**, es el bit de habilitación de la interrupción del temporizador 2 por desbordamiento, cuando este bit esta en uno y junto con el bit de interrupción general en registro de estado también esta en uno la interrupción por desbordamiento del Temporizador 2 es habilitada. Es decir si ocurre un desbordamiento en el registró TCNT2 del temporizador 2 se ejecutara la interrupción

El bit 2 **TOIE1**, es el bit de habilitación de la interrupción del temporizador 1 por desbordamiento, cuando este bit esta en uno y junto con el bit de interrupción general en registro de estado también esta en uno la interrupción por desbordamiento del Temporizador 1 es habilitada. Es decir si ocurre un desbordamiento en el registró TCNT1H y TCNT1L del temporizador 1 se ejecutara la interrupción

El bit 0 **TOIE0**, es el bit de habilitación de la interrupción del temporizador 0 por desbordamiento, cuando este bit esta en uno y junto con el bit de interrupción general en registro de estado también esta en uno la interrupción por desbordamiento del Temporizador 0 es habilitada. Es decir si ocurre un desbordamiento en el registró TCNT0 del temporizador 0 se ejecutara la interrupción.

### Registro Temporizador/Contador ( TCNT0)

BIT							
7	6	5	4	3	2	1	0

Este registro sirve para llevar la cuenta del temporizado, al ser un registro de 8 bits cuenta desde 0 hasta 255 antes de desbordarse y llevar el registro de nueva cuenta a ceros, el cambio de un número a otro dentro del registro dependerá de el preescalador y el frecuencia del reloj.

### Registro de Banderas de Interrupción del Temporizador/Contador (TIFR0)

BIT							
7	6	5	4	3	2	1	0
					OCF0A	OCF0A	TOV0

Los bits TOV2, TOV1 y TOV0, se ponen a uno cuando ocurre el desbordamiento del temporizador correspondiente, estas bandera se limpian por hardware cuando se ejecuta la interrupción correspondiente.

## MATERIAL Y EQUIPO

1 Protoboard  
1 C.I. AVR ATMEGA 16  
8 LEDs  
8 resistencias de 330Ω  
Fuente de Alimentación Regulada de 5 volts  
Cables para conexión rápida  
1 Multímetro

### **MATERIA ALTERNO**

1 C.I. AVR ATMEGA 16

TARJETA DE DESARROLLO UPIITA-F3

TARJETA VISUALIZADOR.

PROGRAMADOR AVR ISP.

CABLE PLANO CON CONECTORES DE 10 PINES DOBLE FILA HEMBRAS.

CONVERTIDOR DE VOLTAJE DE 127 VAC A 9 VDC.

1 Multímetro

## **DESARROLLO**

### **Planteamiento del problema**

Se requiere escribir un programa en el cual se ejecute una cuenta ascendente, esta asciende cada vez que se desborde el temporizador y se despliega en el PUERTO B. Se debe de emplear una pre escala de 1024 y con el sysclock a 1 MHz.

### **Procedimiento**

1. Realizar el diagrama de flujo.
2. Incluir las bibliotecas necesarias
3. Desactivar la interrupción global
4. Definir la pre escala
5. Habilitar la interrupción por desbordamiento del temporizador 0
6. Habilitar la interrupcion global
7. En la función de interrupción sacar a puerto la variable que incrementa
8. Incrementar la variable

## Diagrama del circuito

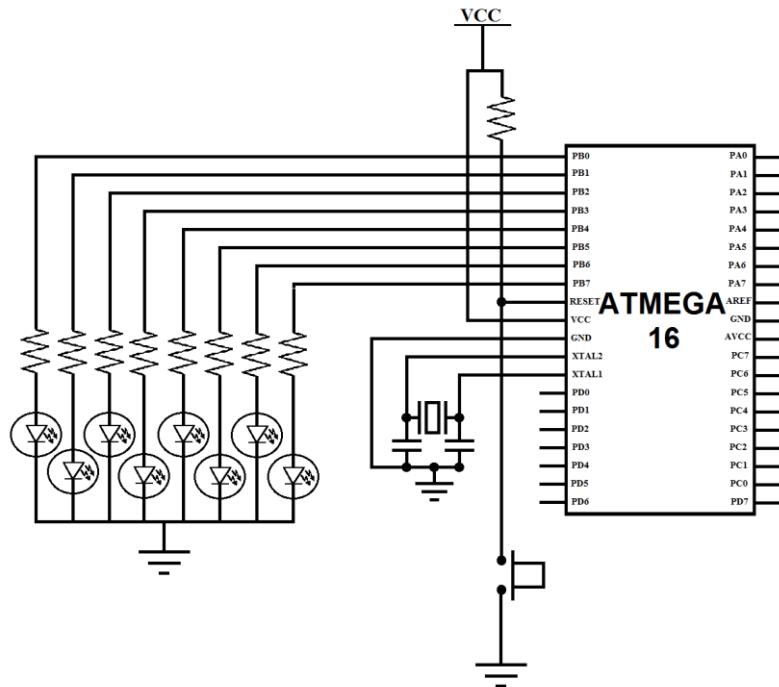


Figura 5.1 Diagrama del circuito para la practica 5.

## CODIGO EN LENGUAJE C.

```
#include <avr/io.h>
#include <avr/interrupt.h>

short int i=0;
ISR (TIMER0_OVF_vect) {
    PORTD = i;
    i++;
}

int main(void)
{
    cli();
    TCCR0B = (1<<CS02)|(0<<CS01)|(1<<CS00); //PREESCALA A 1024
    TIMSK0 = (1<<TOIE0); // HABILITO LA INTERRUPCION POR SOBREFLUJO
    DEL TIMER 0
    DDRD = 0xFF; //
    sei();
    while(1)
    {
        //bucle infinito
    }
}
```

## **ACTIVIDADES DE TRABAJO AUTÓNOMO.**

- 1) Practica propuesta por el profesor en clase.