

Laboratorio de microprocesadores 86.07

Guía de Trabajos Prácticos

1° cuatrimestre

Año 2022

TRABAJO PRÁCTICO N° 3

Objetivo

Utilizar los distintos modos del timer para hacer parpadear un led a diferentes frecuencias y para controlar el movimiento de un servomotor con pulsadores.

Parte 1 – Timers

Descripción

Se pide hacer un programa que haga parpadear el LED conectado al PB1, en 3 frecuencias distintas o que lo deje ENCENDIDO FIJO, según los valores que haya en las entradas PD2 y PD3 según se indica en la siguiente tabla:

PD2	PD3	Estado del LED
0	0	ENCENDIDO FIJO
0	1	PARPADEA CON PRESCALER CLK/64
1	0	PARPADEA CON PRESCALER CLK/256
1	1	PARPADEA CON PRESCALER CLK/1024

Nota: si alguien presiona las teclas mientras está funcionando el micro, los leds deberán cambiar acorde a la tabla.

Para resolver el problema se deberá usar el **Timer1 y su interrupción por OVERFLOW**.

Cuando el LED esté en modo **ENCENDIDO FIJO** el timer estará apagado, en los otros 3 casos, el timer contará los pulsos de clock divididos por prescaler 64, 256 y 1024 respectivamente. Cuando se produce un overflow (desborde) deberán cambiar el estado del LED, es decir, si está prendido se apaga y viceversa.

Se deberá calcular la frecuencia o periodo con que se prenderá el LED en los 3 casos.

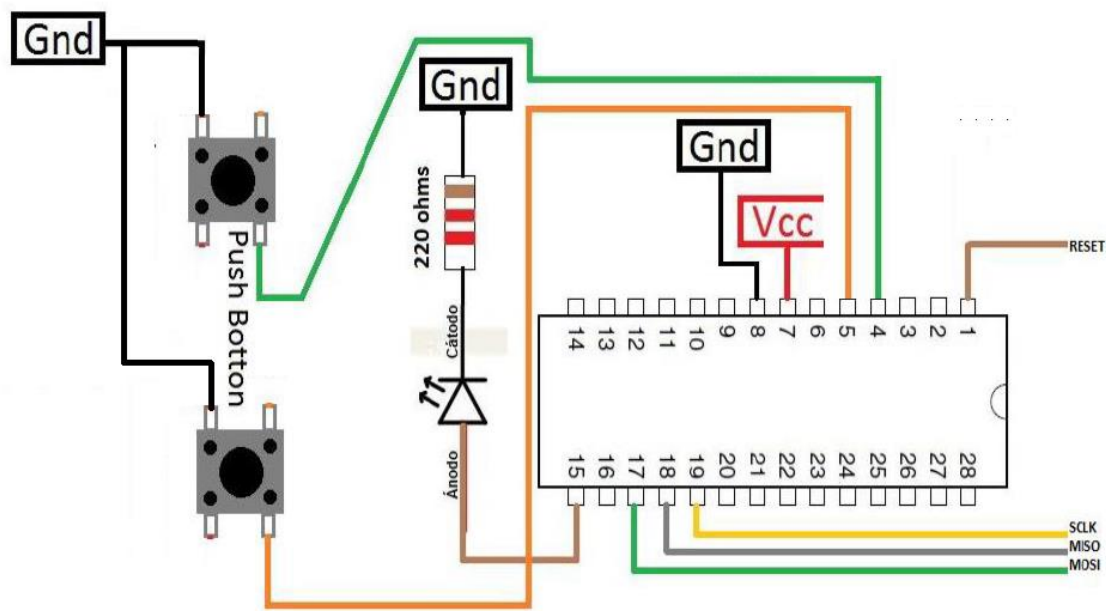
Se les recuerda que si usan la placa de Arduino, la frecuencia del clock es 16MHz.

En las entradas PD2 PD3 están conectados 2 switches, que como cualquier tecla produce rebotes al ser presionada. Deben implementar rutinas antirrebote utilizando otro timer para detectar correctamente las teclas

Materiales

- 1 LED
- 1 Resistencia de 220 Ohms
- 2 Pulsadores
- 1 Microcontrolador ATmega8
- Programador USBasp V3.0

Diagrama esquemático



Parte 2 – PWM

Descripción

Hacer un programa que mueva hacia izquierda o derecha un servomotor. Para eso se dispone de 2 pulsadores y un servomotor como indica el esquema 2.

Para lograr el movimiento del motor se deberá usar PWM, o modulación por ancho de pulso, que consiste en modificar el ciclo de trabajo de una señal (sin modificar su frecuencia). Con esta señal se alimentará el servo de forma que el valor en ON de la señal marcará la posición del indicador del motor.

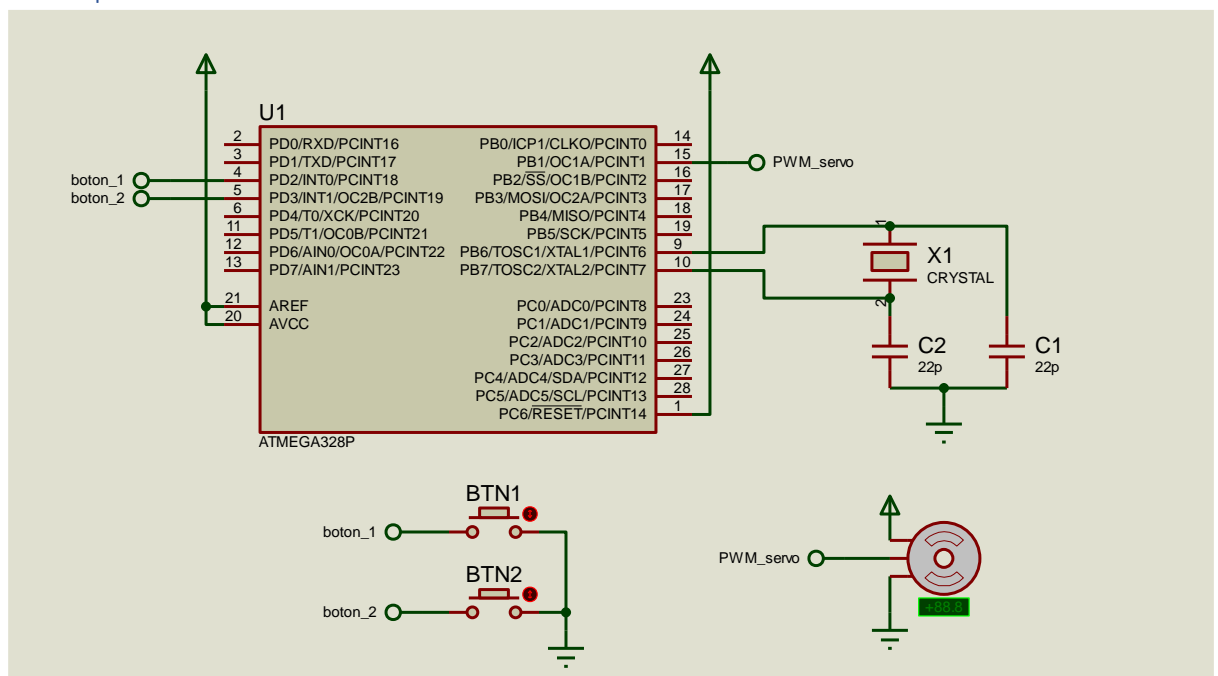
Consideraciones:

- Con cada presión del botón 1 se moverá el motor hacia la izquierda hasta llegar a su tope
- Con cada presión del botón 2 se moverá el motor hacia la derecha hasta llegar a su tope
- Se deben tener al menos 16 pasos de movimiento entre extremos
- **Si se mantiene apretado algún pulsador se tendrá que ir moviendo el motor en la dirección que corresponda**

Materiales

- 1 Servomotor
- 2 Pulsadores
- 1 Microcontrolador ATmega8
- Programador USBasp V3.0

Diagrama esquemático

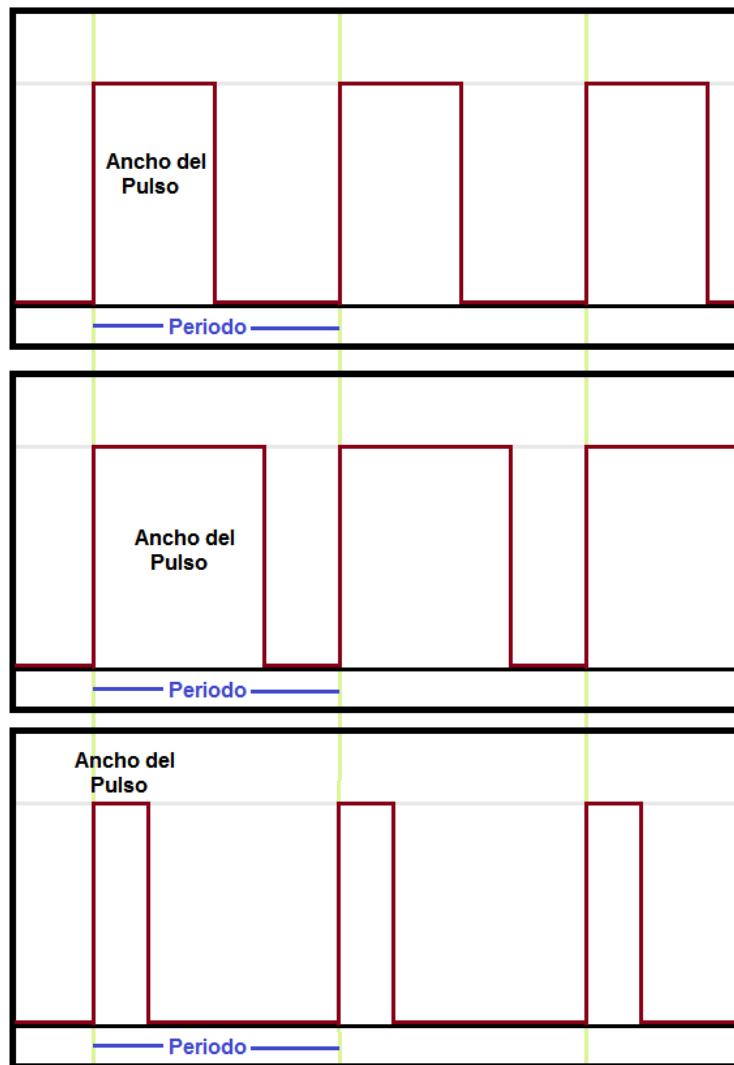


PWM

Modulación por ancho de pulso o PWM (Pulse Width Modulation), de una señal, es cuando se modifica el ciclo de trabajo o el ancho del pulso de una señal periódica, en este caso representado por una señal cuadrada, uno de los usos del PWM entre muchos otros, es controlar la cantidad de energía, en este caso el voltaje promedio es mayor conforme aumenta el ciclo de trabajo.

En la imagen se puede observar, que el período de la señal permanece fijo, por lo tanto, la frecuencia también, solamente cambia el ciclo de trabajo, en la primera se observa que el ciclo de trabajo es de aproximadamente 50% lo cual nos indica que es el porcentaje de voltaje promedio entregado a la carga.

El PWM se puede utilizar en varias cosas, como el control de la velocidad de motores de DC, la posición de un servomotor, fuentes conmutadas, entre otras cosas más.

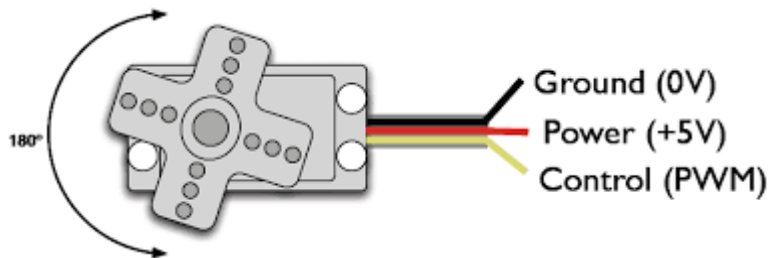
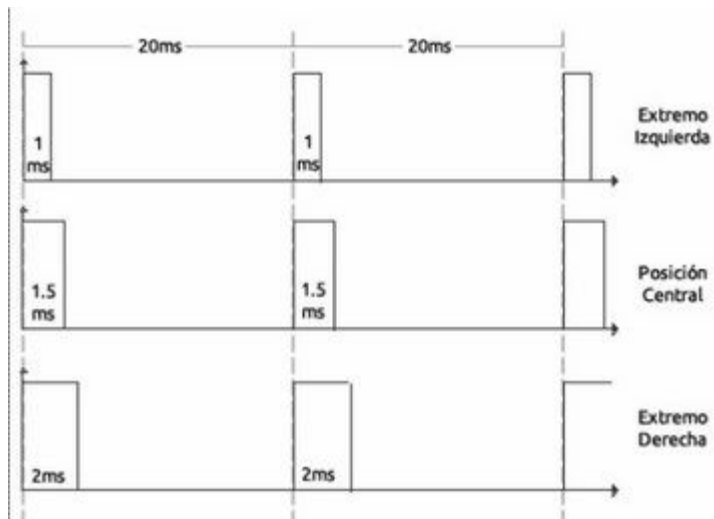


Un servomotor (también llamado servo) es un dispositivo similar a un motor de corriente continua pero que tiene la capacidad de ubicarse en una posición específica dentro de su rango de operación. Además, puede mantenerse estable en dicha posición ya que dispone de un circuito electrónico de control. Dependiendo del tipo de servomotor, este puede ser controlado tanto en velocidad como en posición.

El PWM enviado al motor determina la posición del eje, y se basa en la duración del pulso enviado a través del cable de control; el rotor girará a la posición deseada.

El servomotor espera ver un pulso cada 20 milisegundos (ms) y la longitud del pulso determinará hasta dónde gira el motor. Por ejemplo, un pulso de 1.5ms hará que el motor gire a la posición de 90°.

Si el tiempo es inferior a 1,5 ms, se mueve en sentido contrario a las agujas del reloj hacia la posición de 0°, y si el tiempo es superior a 1,5 ms, el servo girará en sentido de las agujas del reloj hacia la posición de 180°.



[Lectura recomendada](#)

"The AVR microcontroller and embedded systems. Embedded system using Assembly and C".

Autores: MUHAMMAD ALI MAZIDI, SARMAD NAIMI, SEPEHR NAIMI

Capítulos 9, 15 y 16