

(6609) LABORATORIO DE MICROCOMPUTADORAS

Proyecto:
(*Manejo de Puertos*)

Profesor:	Ing. Guillermo Campiglio
Cuatrimestre / Año:	1C 2022
Turno de clases prácticas:	Miércoles 19-22hs
Jefe de Trabajos Prácticos:	Pedro Ignacio Martos
Docente guía:	Fernando Pucci

Autores			Seguimiento del proyecto									
Nombre	Apellido	Padrón										
Juan Cruz	Larraya	104162										

Observaciones:

Fecha de aprobación		

Firma J.T.P.

COLOQUIO	
Nota final	
Firma Profesor	

Índice

1. *Objetivo*
2. *Descripción del proyecto*
3. *Diagrama de conexiones*
4. *Esquemático*
5. *Lógica del programa*
6. *Resultados*
7. *Conclusiones*
8. *Apéndice*

1. Objetivo

El objetivo de este trabajo práctico es aprender acerca del uso y la configuración de los puertos del microcontrolador ATmega328p utilizando los registros I/O del mismo. Utilizar la resistencia de pull-up interna.

2. Descripción del proyecto

El programa consiste en hacer parpadear un LED de forma que este prendido y apagado durante 500ms. El LED parpadea cuando se aprieta el botón 1 y se detiene cuando se aprieta el botón 2. Para la conexión de cada pulsador se usó una configuración distinta donde en el caso del pulsador 2 se usó la resistencia de pull-up.

3. Diagrama de conexiones

En la figura 3.1 a continuación se observa el diagrama de conexiones del circuito.

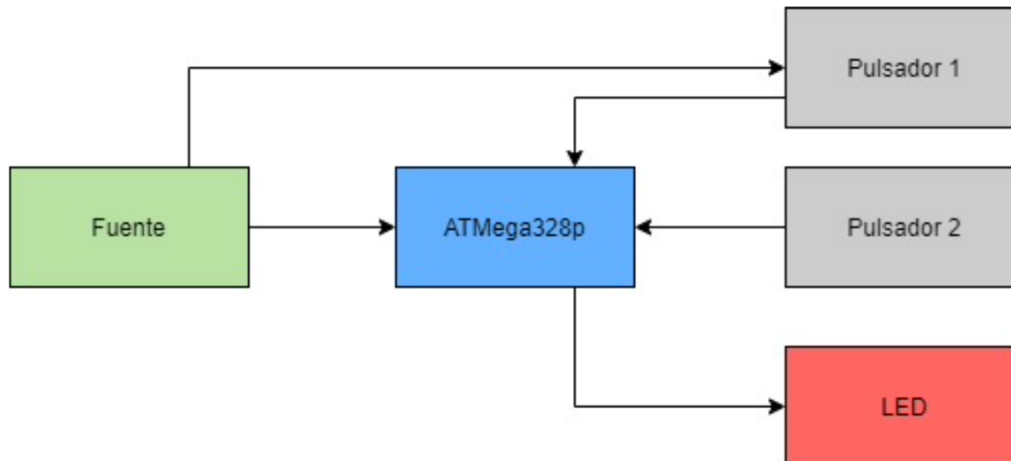


Figura 3.1: Diagrama de bloques

Se puede observar cómo tanto el microcontrolador como el pulsador 1 son alimentados por la fuente. Además se observa como los pulsadores corresponden a una entrada del ATmega328p mientras que el LED es una salida

4. Esquemático

A continuación en la figura 4.1 se muestra el esquemático de las conexiones eléctricas del circuito.

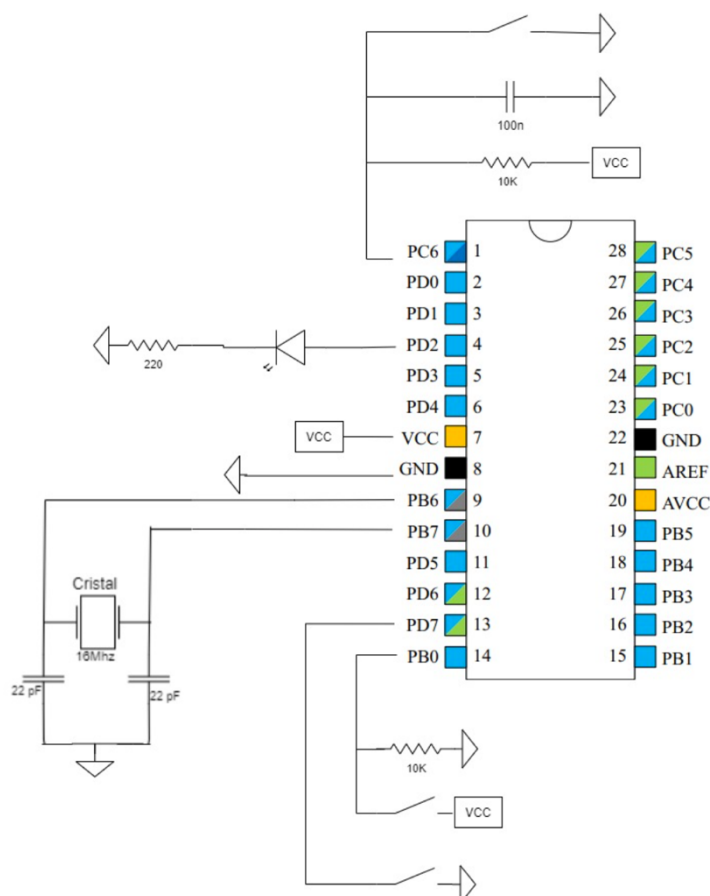


Figura 4.1: Esquemático del circuito

Se conecta un LED al pin 4 (PD2) con una resistencia de 220Ω en serie, que se encenderá cuando haya un 1 lógico en el pin. Luego, el pulsador 1 está conectado al pin 14 (PB0) y el pulsador 2 al pin 13 (PD7). Como se mencionó antes, se usó un tipo de configuración distinta para cada botón. En el caso del botón 1, se lo conecto directamente a VCC, de forma que imponga un 1 lógico cuando esté presionado y un 0 lógico cuando no, por lo que se lo conecto a una resistencia de $10K\Omega$ para que

funcione como una resistencia de pull-down externa. Para el botón 2, los niveles lógicos funcionan de manera opuesta, por lo que para esto se conectó el pulsador a tierra y el pin de microcontrolador se configura para que active la resistencia de pull-up interna. Por último, también se encuentran el circuito de reset y el cristal de 16 MHz.

5. Lógica del programa

En la figura 6.1, se observa el diagrama de flujo del programa.

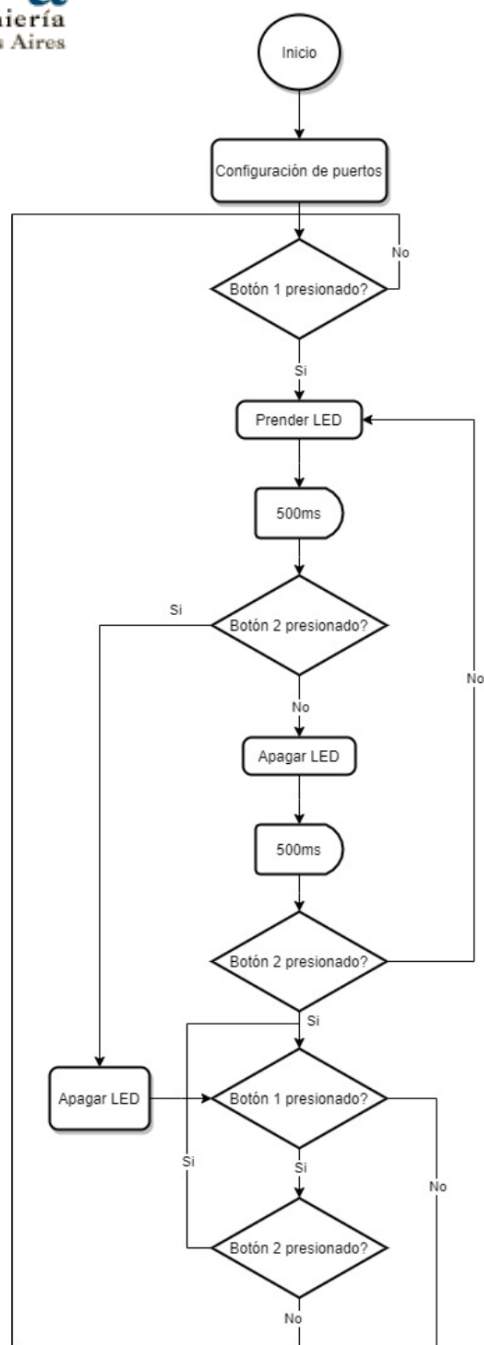


Figura 6.1: Diagrama de flujo

El programa fue diseñado de forma que el LED comience apagado y que los botones deban ser solamente presionados para que tengan un efecto en el circuito.

Al momento de verificar si cualquier botón fue presionado, se implementó una rutina anti rebote para confirmar que el botón haya sido realmente presionado

donde se espera un delay de 40 ms antes de comprobar nuevamente el estado del botón.

Un punto importante a destacar es que la verificación de si se apretó el botón 2 mientras el zLED se encuentra titilando ocurre durante el delay de 500 ms. Para esto se lo chequea regularmente cada aproximadamente 12 milisegundos, y si está apretado setea el flag Z en 0 e interrumpe el delay.

Una vez que se aprieta el pulsador 2, el programa comprueba si se están apretando ambos botones a la vez. En caso de ser así se mantiene con el LED apagado hasta que se deje de presionar uno de los botones y vuelve a esperar que se apriete el pulsador 1.

6. Resultados

Los objetivos de este trabajo práctico se alcanzaron de forma exitosa, así como se verificó el funcionamiento correcto del microcontrolador.

7. Conclusiones

Este trabajo fue útil para desarrollar el conocimiento en el manejo de puertos del microcontrolador como también me resultó un muy buen primer acercamiento a la programación en código Assembly.