

Universidad Autónoma de Baja California
Facultad de Ciencias Químicas e Ingeniería



MICROCONTROLADORES

Practica No. 3

**Programación en Lenguaje Ensamblador del
ATMega328p y retardos por software**

Docente: Castro Gonzalez Ricardo

Alumno: Gómez Cárdenas Emmanuel Alberto

Matricula: 01261509

Objetivo:

Mediante esta práctica el alumno analizará la implementación de retardos por software, así como también se familiarizará con la configuración y uso de puertos

Material:

- Computadora personal con AVR Studio, AVRDUDE y tarjeta Tjuino (Atmega328p)

Teoría:

- **Análisis y cálculos de retardos de la práctica (Subido a github)**
- **Teoría sobre puertos de E/S del Atmega328p**

En el contexto de la programación de microcontroladores, es fundamental comprender el papel de los puertos de entrada/salida, que actúan como interfaces entre el microcontrolador y su entorno físico. Este dispositivo cuenta con tres puertos principales: Puerto B, C y D. Es importante destacar que Puerto B y Puerto D son de 8 bits, mientras que Puerto C consta de 7 bits.

Estos puertos son altamente versátiles y de propósito general, lo que significa que los usuarios tienen la capacidad de configurarlos tanto para recibir datos como para enviar señales de salida según sus necesidades específicas. Sin embargo, cada terminal en estos puertos puede tener varias funciones alternativas. Esta versatilidad se vuelve esencial debido a la necesidad de cumplir con diversas funciones internas del microcontrolador. Por ejemplo, el puerto serie requiere terminales específicas para la recepción y transmisión de datos, que son funciones alternativas asignadas a PD0 y PD1, respectivamente. Además, estas terminales pueden tener otras funciones alternativas, como la detección de cambios.

El manejo de cada puerto implica la manipulación de tres registros clave en el espacio de Registros I/O. Estos registros desempeñan funciones cruciales en la configuración y operación de los puertos.

El registro **DDRx** determina la dirección del puerto, permitiendo que se comporte como entrada o salida. Un valor lógico de 1 activa el puerto como salida, mientras que un valor lógico de 0 lo configura como entrada. Es importante destacar que cada terminal en el puerto puede tener su propia configuración, lo que significa que la dirección de una terminal puede diferir de otras, incluso dentro del mismo puerto.

El registro **PORTx** está directamente relacionado con las terminales del puerto. Cuando el puerto se configura como salida, el registro **PORTx** refleja lo que se envía a las terminales externas. En cambio, cuando el puerto es una entrada, el registro **PORTx** se utiliza para habilitar resistores pull-up.

El registro **PINx** es relevante cuando el puerto actúa como entrada. Permite realizar lecturas directas de las terminales. Sin embargo, cuando el puerto es una salida, escribir un valor lógico de 1 en este registro invierte el valor almacenado en el registro **PORTx**, cambiando de 1 a 0 o viceversa.

Esta comprensión de los puertos de entrada/salida y su manipulación mediante registros es esencial para desarrollar aplicaciones efectivas con microcontroladores como el ATmega328P.

Conclusiones y comentarios:

Esta práctica me ha permitido aprender sobre la implementación de retardos por software y familiarizarme con la configuración y uso de los puertos en el microcontrolador Atmega328p. He comprendido la importancia de los puertos de entrada/salida y cómo manipular registros como DDRx, PORTx y PINx para controlar la dirección, el estado y la lectura de las terminales. Estos conocimientos son esenciales para futuros proyectos de programación de microcontroladores como el Atmega328p.

Bibliografía:

Santiago, F. (2021). *El Microcontrolador ATmega328P de Microchip: Programación en Ensamblador, Lenguaje C y un enlace con Arduino*. Universidad Tecnológica de la Mixteca.