

Práctica 2

Hello World en microcontroladores



Imparte:

DANIEL GIOVANNI MARTÍNEZ SANDOVAL

I7266 - Programación de Sistemas Embebidos - D10
Hecho en \LaTeX

1. Objetivos

1.1. Generales

- 1) Que el alumno se familiarice con el proceso de programación del microcontrolador que se usará durante el curso.

1.2. Particulares

1. Que el alumno desarrolle, en lenguaje C, una serie de programas que fortalezcan los conocimientos básicos de la arquitectura del ATmega328P.
2. Que el alumno, al practicar, desarrolle familiaridad hacia la estructura básica de un programa en C orientado a un microcontrolador.
3. Que el alumno aprenda a realizar un proceso básico de desarrollo y programación de un microcontrolador.

2. Indicaciones

- Una vez instalado el entorno de desarrollo descrito en [este enlace](#), y en el editor de texto de su preferencia, codificar un programa en lenguaje C para cada objetivo, de forma que el ATmega328P realice lo que se indica cada uno de los siguientes enunciados:
 1. Encender y apagar un LED conectado al pin PC5 del ATmega328P con una frecuencia de $1.0 \pm 0.1\text{Hz}$.
 2. Encender y apagar de manera secuencial cada uno de los pines del puerto C del ATmega328P, cambiando de LED a una frecuencia de $1.0 \pm 0.1\text{Hz}$.
 3. Encender un led en el pin PC5 del ATmega328P y mantenerlo siempre encendido, excepto cuando se presione un pushbutton conectado a pin PD7.

- Solo se aceptarán las prácticas que se realicen en lenguaje C y compilables con `avr-gcc`, para lo cual se sugiere el siguiente flujo de trabajo:
 1. Crear una nueva carpeta (sugerencia extra: numerada), por ejemplo: `021_Patron1`.
 2. Dentro de la carpeta, colocar el archivo de código fuente correspondiente y nombrarlo `main.c`.
 3. Copiar la dirección de la carpeta, la cual se puede obtener desde la barra de dirección del explorador de archivos.
 4. Abrir una nueva instancia de Powershell (desde el menú de Windows).
 5. Utilizar el comando: `cd "C:\direccion\de\la\carpeta"` (substituyendo entre las comillas por la dirección que se copió).
 6. Dentro de la carpeta, usar los comandos mencionados en la guía:
 - Para compilar:
`avr-gcc main.c -mmcu=atmega328p -o main.elf`
 - Para convertir a formato intel hex:
`avr-objcopy -O ihex main.elf main.hex`
 - Para cargar el código al microcontrolador:
`avrdude -p atmega328p -c usbasp -U flash:w:main.hex`
 7. **RECOMENDACIÓN:** se sugiere **NO** copiar y pegar los comandos aquí mencionados, más bien, escribirlos manualmente en la consola, ya que se pueden copiar de manera errónea debido al formato PDF.
- En Classroom, se deberá cargar un reporte de actividad que incluya lo siguiente:
 1. Portada u hoja de presentación que incluya (por lo menos): Título, nombre de los autores, nombre de la materia, nombre y escudo de la universidad.
 2. Introducción.
 3. Objetivos.
 4. Marco teórico: una breve síntesis de los temas vistos en clase que se requirieron para entender y realizar la actividad.
 5. Desarrollo: un resumen del procedimiento que se siguió para realizar la actividad.
 6. Conclusión: un breve texto que señale los resultados y aprendizajes de la actividad.

Generalidades

- Las actividades escritas se deberán cargar a Classroom en un documento en formato PDF.
- Los archivos de código fuente también se deberán cargar a Classroom en su formato nativo, en caso de ser múltiples, se pueden cargar en un formato comprimido (por ejemplo: zip).
- Las actividades no serán válidas hasta que se entreguen todos los requisitos, además, los archivos de código fuente que se entreguen deberán ser compilables sin errores (warnings sí están permitidas) y los binarios deberán realizar la funcionalidad descrita por la actividad.
- El profesor se reserva el derecho de solicitar (ya sea de manera grupal o individual) que los alumnos presenten su actividad funcional como parte de la evaluación de la misma actividad.

- En trabajos que incluyan investigación, es obligatorio añadir las fuentes; el formato es libre, pero se recomienda IEEE o APA.