

TRABAJO PRÁCTICO N° 1

Temas: RISC Microchip. Manejo de instrucciones y memoria. Polling.

Objetivos:

- Programación en lenguaje C, y debugging en MPLAB.X
- Técnica de programación por polling. Manejo de puertos de I/O.

Material de soporte:

- Para resolver el práctico se propone consultar el “**Material para clases prácticas**”, como así también la sección correspondiente a este práctico, disponible en el Aula Virtual.
- Utilizar el proyecto base **proyectobase1.X** también disponible en el Aula Virtual.

Ejercicio 1: Polling

Realizar una rutina que consista en encender y apagar dos LEDs después de presionar y soltar el pulsador correspondiente a cada uno. Deberá inspeccionar ambos pulsadores en forma *secuencial*. Un pulsador y su LED correspondiente son independientes el uno del otro. Los pulsadores estarán por defecto abiertos con valor binario **1** (al pulsar van a **0**) y se encontrarán dispuestos en los pines **12** y **13** del **PORTD**. Los LEDs se encenderán con nivel lógico 1, y estarán ubicados en los pines **4** y **5** del **PORTA**.

Ejercicio 2: Automatismo

Un sistema de calefacción de una fábrica posee un **termostato variable** y cinco sensores de detección de apertura de accesos (portones y puertas al exterior de la fábrica). El sistema de control debe encender un motor que impulsa aire caliente y abrir la válvula de salida de gas, si se cumplen a la vez las siguientes condiciones:

1. El termostato variable desciende por debajo de 22°C, detectable por un sensor que entrega valores en 16 bits por un puerto. El sensor trabaja en un rango de 0 a 100°C.
2. Por lo menos 3 de los 5 accesos se encuentran cerrados, detectable por un sensor del tipo ON/OFF (*cuando el sensor detecta que la puerta/portón está cerrado, estará en ON*).
3. Hay presión de gas suficiente por encima del umbral mínimo, detectable por un sensor de nivel del tipo ON/OFF.
4. Hay energía eléctrica, detectable por un sensor de presencia de tensión del tipo ON/OFF.

Nota: La lectura de entradas debe hacerse por **polling**. Otorgar en lo posible al programa una estructura en bloques tal que:

1. En el primer bloque se tomen las variables de entrada y se las deposite en memoria.
2. En el segundo bloque se instrumente la lógica consignada, utilizando las variables de memoria.
3. En el tercer bloque se actualicen las salidas y sólo se escriba una salida cuando exista un cambio respecto a su estado inmediato anterior (no escribir permanentemente).