

Práctica 1

Tarjeta de Desarrollo para un Microcontrolador

M.I. Marco Negrete

Entrega: 06-02-2014.

1. Objetivos

- Diseñar una tarjeta para un microcontrolador que sea útil en el desarrollo de un robot para laberinto.
- Aprender a elegir el mejor microcontrolador de acuerdo con el problema que se quiere resolver.
- Escribir un programa para el microcontrolador elegido que sirva como entrenamiento en el uso de dicho dispositivo.

2. Introducción

Una parte importante del robot para resolver un laberinto que se presentará como primer proyecto es el microcontrolador. En él se implementarán los controles de bajo nivel y la lectura de todos los sensores (infrarrojos y encoders en cuadratura).

Se deja a elección del alumno la marca y modelo del microcontrolador a usar, sin embargo, es necesario que dicho dispositivo cuente con ciertos periféricos necesarios para el buen desempeño del robot.

Para el control de velocidad de los motores es necesario un módulo que pueda generar una señal PWM, por lo que el microcontrolador elegido debe contar con al menos dos módulos PWM. Asimismo, para la lectura de los sensores de distancia, es necesario un convertidor analógico-digital de al menos tres canales (o bien, tantos canales como sensores tenga el robot). La lectura de los encoders en cuadratura se puede realizar sólo con pines lógicos de entrada-salida de propósito general, sin embargo, la lectura se simplifica en buena medida si el dispositivo elegido cuenta con interrupciones externas o temporizadores para detectar los flancos de subida de las fases del encoder. Por último, se requiere que el microcontrolador cuente con un puerto de comunicación RS232 o USB. Este último, aunque no es indispensable para el funcionamiento del robot, es de gran ayuda para depurar los programas desarrollados.

3. Desarrollo

3.1. Circuito impreso para el microcontrolador

Diseñar una tarjeta en circuito impreso para el microcontrolador elegido. La tarjeta debe tener al menos los siguientes elementos:

- Led de encendido
- Conector para alimentación
- Conector para comunicación serial (preferentemente, conector tipo DB9) o para USB
- Al menos tres pines conectados a los convertidores A/D para medir la señal analógica generada por los sensores de distancia.
- Dos conjuntos de tres pines para el control de los motores (es decir, estos pines se conectarán al puente H que se implementará en la práctica 2). De cada conjunto, un pin debe ir a una salida de PWM y dos a salidas lógicas.
- Dos grupos de cuatro pines para alimentación y lectura de los encoders en cuadratura. Cada grupo de 4 pines debe tener: Vcc, tierra, A y B. Los pines A y B son para las fases del encoder y deben ir conectados a alguna interrupción externa o *timer* del microcontrolador.
- Al menos cuatro pines lógicos (conectados a cualquier pin I/O del microcontrolador) que servirán para comunicarse con el CPLD (que se usará en la práctica 5).

Nota: Es deseable que el alumno diseñe su propia tarjeta de desarrollo para el microcontrolador elegido, sin embargo, se acepta el uso de cualquier tarjeta comercial siempre y cuando tenga todo lo necesario para el funcionamiento del robot.

3.2. Programación del microcontrolador

Escribir un programa que realice lo siguiente:

- Al inicio, enviar, por puerto RS232 o USB, una cadena que indique que el programa ha comenzado a ejecutarse, por ejemplo “HAL9000.->Hello Dr. Chandra...”.
- Llevar un contador que se incremente cada 500 [ms]. Éste será el contador 'A' y debe comenzar a incrementarse cuando se inicie el programa y continuar incrementándose indefinidamente (no importa que se desborde).
- Llevar otro contador 'B', que será operado con los siguientes comandos (cadenas de texto enviadas por USB o RS232):
- ++ Incremento en uno.
- -- Decremento en uno.

- `set value` Fija en el contador B el número `value`, que puede ser cualquier entero entre 0 y 255.
- B Envía por serial el valor del contador B.
- A Envía por serial el valor del contador A.
- Todos los comandos deben ejecutarse al enviar el caracter de “enter”.

4. Elementos a evaluar

- El programa debe funcionar COMPLETAMENTE, esto es, ejecutar correctamente todos los comandos. De acuerdo con las reglas del curso, no se aceptan prácticas que funcionen a medias.
- Al momento de mostrar la práctica, se pedirá al alumno que realice alguna modificación sencilla al código, por ejemplo, cambiar la cadena que se envía al principio.
- Si la tarjeta requiere de alimentación externa, es responsabilidad del alumno llevar todo lo necesario (pilas, conectores, etc). No se acepta el uso de caimanes o protoboards. Se deben usar conectores adecuados.