

Competencia de Robótica

R2-D2 2014

Categoría: Velocista

Nombre del Robot: Miyagui

Institución: UTN-FRA

Participantes:

Calvo, Juan Ignacio
Schuster, Mariela
Medina, Sergio Daniel



Índice

1. Introducción	3
1.1. Descripción básica de funcionamiento	3
1.2. Objetivos	3
2. Mecánica	3
2.1. Descripción de la estructura mecánica	3
2.2. Especificaciones Técnicas	3
3. Electrónica	3
3.1. Descripción del circuito y mención de los integrados utilizados	3
3.2. Esquemático y/o PCB	5
PCB Lado superior (Doble Capa)	5
PCB Lado inferior (Doble Capa)	5
Máscara de componentes:	6
Esquemático General	7
4. Programación	8
4.1. Método de programación y programador utilizado	8
4.2. Descripción de la lógica del código y lenguaje utilizado	8
5. Conclusión	9

1. Introducción

1.1. Descripción básica de funcionamiento

El Robot cuenta con una estructura de pcb espumado, dos motores, uno por cada rueda trasera, y un soporte de apoyo en su parte delantera. También con 3 sensores IR en su parte delantera, con los que sensa la línea blanca, a partir de sus lecturas, se varían las velocidad de los motores con los que se corrige el recorrido. Su control se lleva a cabo por un microcontrolador, ATmega128.

1.2. Objetivos

El objetivo del diseño fue lograr una estructura resistente y a su vez fácil de armar, un material sencillo de trabajar como lo es el pbc espumado, económico y simple.

2. Mecánica

2.1. Descripción de la estructura mecánica

La transmisión de los motores se realiza a través de una corona al eje. Cada uno de los motores traseros tiene su propia corona y rueda. En la parte delantera, solo tiene un punto de apoyo.

2.2. Especificaciones Técnicas

- 2 Motores Mabuchi: 4W 8000rpm
- 1 batería lipo de 7,4 volts 1200mah

3. Electrónica

3.1. Descripción del circuito y mención de los integrados utilizados

Básicamente el circuito consta de las siguientes etapas:

- **Etapas de Alimentación:** Es la etapa que se encarga de regular la tensión para que el resto de las etapas funcionen correctamente.
- **Logica de control:** Basicamente es la etapa donde se encuentra el microcontrolador, con los diferentes elementos básicos para su correcto funcionamiento.
- **Etapas de sensado:** Es la etapa que se encarga de captar la línea a seguir.
- **Etapas de Potencia:** Es la etapa donde se realiza el control de los motores usados.

A continuación serán detalladas estas etapas más en profundidad.

Etapas de Alimentación:

Es la etapa que se encarga de regular la tensión para que el resto de las etapas funcionen correctamente. En un principio, la alimentación general se realiza con una batería de 7,4 volts. Será necesario reducir esta tensión a 5V para que el funcionamiento de las próximas etapas. Para ello se utilizará un regulador de tensión, basado en el integrado LM2576HV, que básicamente permite implementa

una fuente switching simple y con pocos componentes externos (Solo una bobina externa, un diodo zener y un capacitor).

Logica de Control:

Basicamente es la etapa donde se encuentra el microcontrolador, con los diferentes elementos basicos para su correcto funcionamiento. El microcontrolador elegido es un ATMEGA128, trabajando a una frecuencia de 16MHz, por medio de un cristal externo. En ésta etapa alrededor del microcontrolador hay toda una serie de conectores que facilitan cualquier expansión futura, además de los pines reservados para el programador. Cuenta con un pulsador de reset manual.

Etapa de sensado:

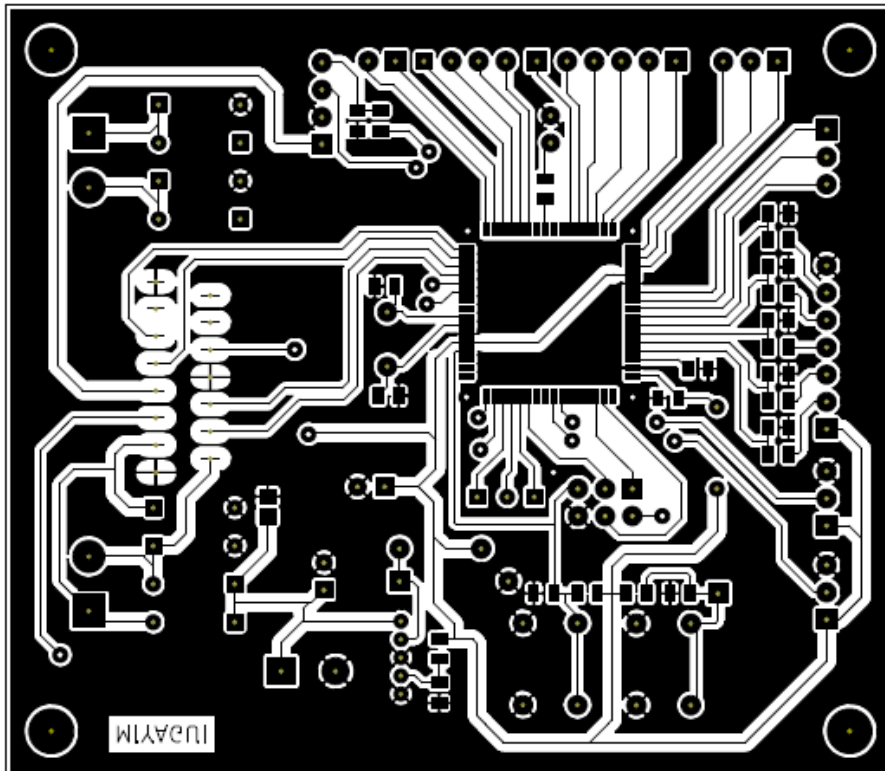
Es la etapa que se encarga de captar la linea a seguir. La idea de esta etapa es, según que tan sobre la linea se encuentre el robot, obtener una medida analógica de ésta posición. Ésto se consigue mediante el uso del sensor TCRT1000, un sensor óptico reflectivo, que en el mismo encapsulado posee el transmisor y receptor (fototransistor). Se agrego proximo a los sensores TCRT1000, un diodo IR a cada uno para aumentar la sensibilidad de las mediciones.

Etapa de Potencia:

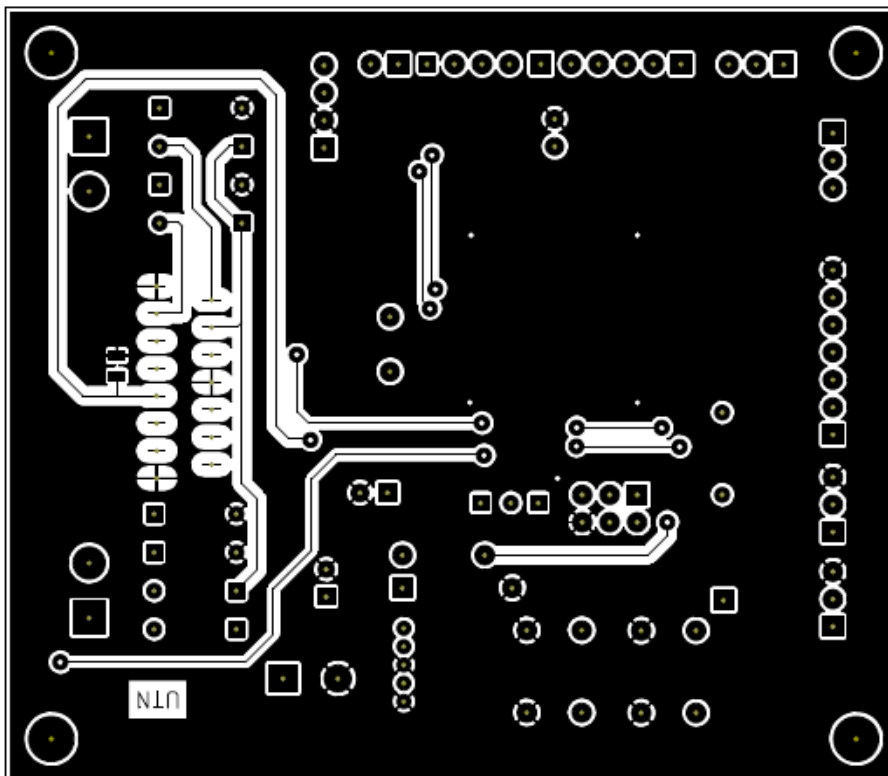
Es la etapa donde se realiza el control de los motores usados. Para ello se emplea el control del tipo puente h, con lo cual se decidió usar el driver L298. A la entrada de dicho integrado se deberá ingresar la señal de control para cada motor (modulado por el microcontrolador) y la señal de selección (entradas digitales). Se agregan diodos "freewheel" para eliminar las tensiones inversas que, a travez de la carga, pueden dañar el sistema.

3.2. Esquemático y/o PCB

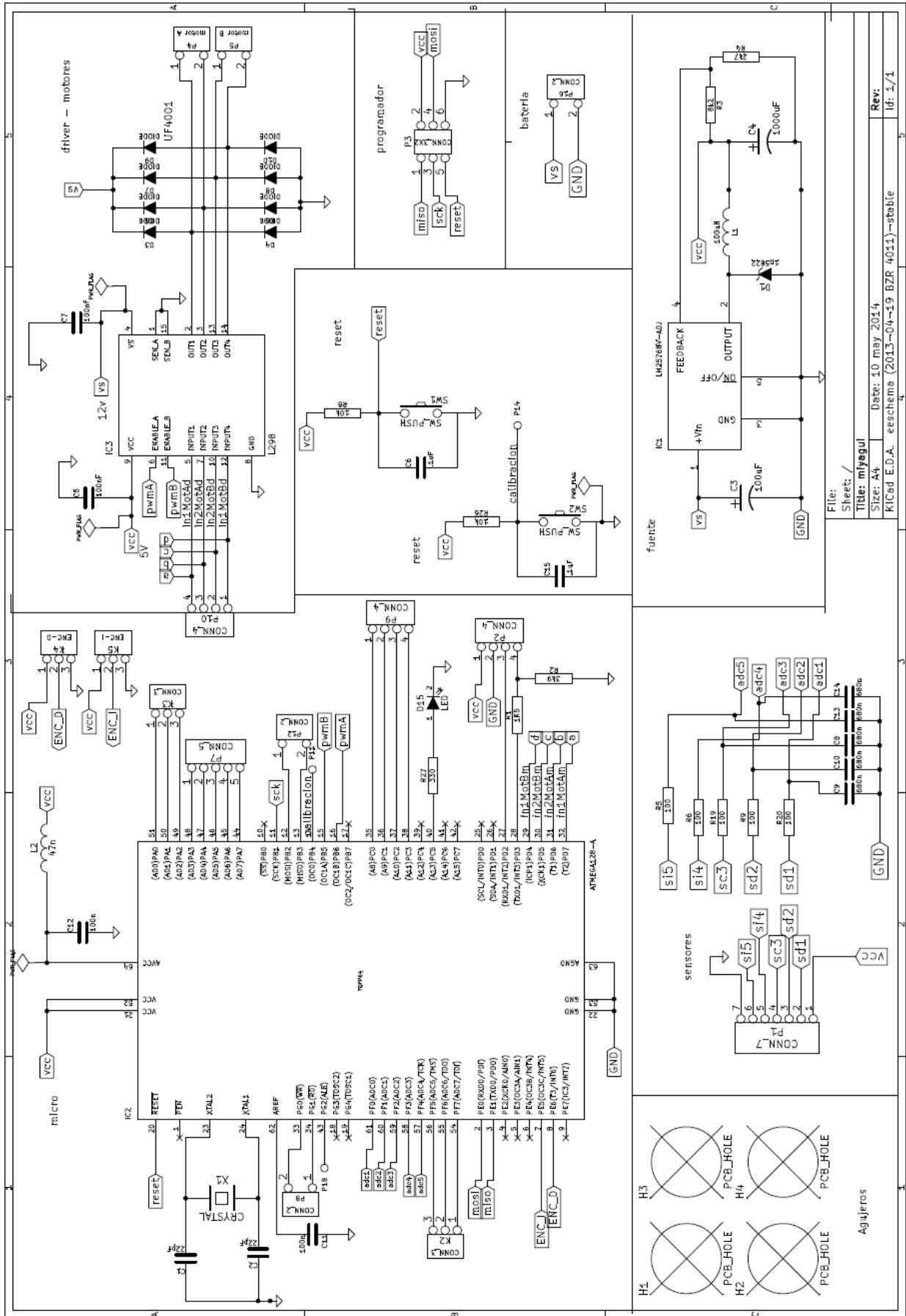
PCB Lado superior (Doble Capa)



PCB Lado inferior (Doble Capa)



Esquemático General



4. Programación

4.1. Método de programación y programador utilizado

Se decidió usar un micro de la empresa ATMEL, en particular el microcontrolador Atmega128, de 8 bits. Para ello se debería usar un programador compatible con éste, en este caso, el USBtiny, al ser el más accesible. Optamos por usar un IDE, en nuestro caso el Eclipse, que nos permitiría trabajar como un proyecto, hacer control de versiones, programación, etc, además de contar con complementos para tal fin.

4.2. Descripción de la lógica del código y lenguaje utilizado

El control del seguimiento de línea del proyecto se basa en un control del tipo PID, que básicamente es un control por realimentación que mediante el calculo del error respecto a un valor deseado, permitirá hacer un ajuste adecuado. Para ello el PID trabaja con tres parámetro: el proporcional, el integral, y el derivativo.

- Proporcional determina la reacción del error actual.
- Integral genera una corrección proporcional a la integral del error (el error de seguimiento se reduce a cero al aplicar la correccion adecuada).
- Derivativo determina la reacción del tiempo en el que el error se produce.

La suma de estas tres parámetros es usada para ajustar la velocidad general del sistema.

En nuestro caso particular, para generar una variacion de velocidad se deberá ir alternando las velocidades, de la rueda izquierda o derecha, según corresponda.

La velocidad final estará definida por el grado de modulación del PWM que controla los motores, con lo cual la velocidad del cada motor está relacionada con el valor de los registros de control dedicados a tal fin en el microcontrolador.

Según éste criterio, el control PID se encarga de definir el valor que tendrá la salida de PWM del microcontrolador para cada motor. Se definió un modulo de Timer para el control de los PWM de los motores, y se lo configuró para trabajar en tal modo (Modo PWM). Cuatro pines de puertos de I/O se destinaron a la selección para el driver de los motores.

En cuanto a las entradas medidas en la etapa de sensado, se realiza una conversión del valor obtenido por medio del periférico de conversión Analógico-Digital. Son tres las entradas analógicas usadas (Para la línea central y las dos laterales).

5. Conclusión

El proyecto que en un principio parecía tener un mayor grado de dificultad, a medida que fuimos avanzando en el desarrollo, vimos cómo se fue simplificando en gran medida. En un principio optamos por una estructura ligera y resistente, que resulto relativamente económica y viable y que cumplió muy bien con las exigencias que le dimos. Posteriormente hicimos la elección de los motores, una tarea que resulto simple. Para hacer el control de ellos empleamos un puente h, el cual fue implementado mediante un driver accesible y que nos evitó la tarea de desarrollarlo con MOSFET (restándole posible problemas relacionados a esto), ya desde el primer momento, una vez montado el microcontrolador en la placa principal pudimos empezar a hacer pruebas sobre ellos. Implementamos una etapa de sensado prácticamente con la misma facilidad, y con esto ya teníamos datos para el control.

Desde el punto de vista del hardware, llevar a delante un seguidor de línea de éstas características resulta muy viable.

En cuanto al firmware que deberá desarrollarse para el microcontrolador, se pueden conseguir resultados muy buenos implementando un simple control on-off de lo más básico, y hay una amplia variedad de técnicas para el control del proceso, como implementar un PID, para asegurar un mejor rendimiento. De este modo la mayor dificultad, y un punto a seguir trabajando, es el software.