

PARCIAL TERCER CORTE - SISTEMAS EMBEBIDOS

Wendy Chaparro, Cristopher Ramírez

Escuela de Ciencias Exactas e Ingeniería

Universidad Sergio Arboleda - Bogotá, Colombia

wendy.chaparro01@usa.edu.co, cristopher.ramirez01@usa.edu.co

1. INTRODUCCIÓN

El proyecto micromouse combina diseño electrónico, programación y control avanzado para desarrollar un robot autónomo capaz de navegar en laberintos. Utilizando una tarjeta STM32F411, sensores infrarrojos (IR) y motores con encoders, se configuró el hardware para controlar motores mediante PWM, leer distancias con sensores IR a través del ADC, y transmitir datos mediante comunicación serial TTL.

La calibración de los sensores y motores permitió obtener parámetros clave para implementar un sistema de odometría y control PID. Estos elementos aseguran un desplazamiento preciso y estable del robot, optimizando su rendimiento en entornos controlados. Este documento describe de manera detallada el desarrollo del sistema de control, desde las configuraciones iniciales hasta la integración del PID y la odometría.

2. MATERIALES

- Circuito PCB del robot
- Placa de desarrollo STM32F411.
- Condensadores cerámicos y electrolíticos.
- Resistencias.
- STM32CubeIDE
- base y estructura del robot
- regulador MP1584
- Encoder de cuadratura
- Puente H
- motores DC
- TTL
- Baterías de litio recargables

3. ANÁLISIS Y DESARROLLO

El proceso de configuración del robot Micromouse comenzó con el diseño del circuito en KiCad, el cual fue impreso en baquela para implementarlo en la estructura del robot. Una vez ensamblado el hardware, se soldaron los componentes y se configuró la tarjeta STM32F411 para el control de los motores, el manejo de sensores IR, y la generación de PWM, así como para habilitar los contadores en modo encoder.

El programa que se implementa es un sistema de control para el funcionamiento del micromouse que se basa en la lectura de los sensores IR, control de motores, y el uso de un perfil de velocidad trapezoidal para ejecutar movimientos precisos dentro de un laberinto. Este sistema combina la calibración y uso de sensores infrarrojos, cálculos de odometría y algoritmos de control PID para guiar al robot de manera controlada y exacta.

Descripción del Algoritmo para el Control del Micromouse

1. Sensores y Toma de Decisiones:

- El micromouse cuenta con cuatro pares de sensores infrarrojos dos frontales y dos para laterales, uno para la izquierda y el otro para la derecha, que envían información con respecto a los obstáculos presentes y posición del micromouse respecto a las paredes del laberinto para la toma de decisiones
- Los sensores se calibraron midiendo distancias en incrementos de 5 mm hasta 100mm como se muestra en la grafía 3. Medidas que fueron ajustadas mediante la conversión de valores ADC a voltaje y a distancia por medio de una interpolación lineal.

2. El Control de Movimiento y Perfil de Velocidad:

La velocidad del micromouse se calcula con un perfil de velocidad trapezoidal que consta de tres etapas:

- **Aceleración:** Los motores comienzan con una velocidad mínima de 100mm/s , el micromouse aumenta su velocidad con una aceleración de 700mm/s hasta alcanzar una velocidad máxima de 250mm/s .

- **Velocidad constante:** Mantiene la velocidad máxima durante la mayor parte de la distancia recorrida.
- **Desaceleración:** Reduce su velocidad a $700mm/s$ para volver a $100mm/s$ antes de detenerse.

El objetivo de este perfil es garantizar que el movimiento sea suave, reduciendo la posibilidad de errores en el posicionamiento debido a aceleraciones o frenadas bruscas.

3. Cálculo de Distancia :

El programa utiliza odometría para calcular la distancia recorrida mediante encoders. La odometría fue configurada mediante una librería personalizada que calcula la posición y orientación del robot con base en los pulsos de los encoders de los motores. se determinaron parámetros clave como el diámetro de las ruedas ($35mm$), los pulsos generados por vuelta de las llantas (780), los pulsos por milímetro. asociados a las ruedas del micromouse (7) y la longitud de arco de ($109,9mm$), información que permitió convertir la distancia objetivo ($59.6 mm$ para giros o $180 mm$ para avanzar) a pulsos y así determinar el número de pulsos que debe generar cada encoder para cumplir con el movimiento requerido y la distancia lineal que se ha recorrido al final y durante cada movimiento.

- a) Distancia recorrida por cada rueda:

$$d = \frac{2\pi r \cdot \text{pulsos}}{\text{pulsos por revolución}}$$

- b) Velocidad lineal:

$$v_{\text{lineal}} = \frac{\Delta d}{\Delta t}, \quad v_{\text{angular}} = \frac{d_r - d_l}{\text{base entre ruedas}}$$

4. Control de Motores:

El control de los motores se realiza utilizando las siguientes herramientas:

Función Feed-Forward: Proporciona una aproximación de la velocidad de los motores a una velocidad deseada por medio del uso de una función que permite pasar de volt a PWM para así ser controlada más fácil por el PID, esta función se calcula por medio de la calibración del motor en términos de velocidad vs voltios 4.

filtro FIR: es un tipo de filtro digital que se utiliza para suavizar señales o eliminar ruidos no deseados. Un FIR es aquel que su respuesta a una entrada dada tiene una duración limitada, definida por el

número de coeficientes del filtro, que para este caso estuvieron entre 7 y 20 como se muestra en las gráficas 6.

- Con 7 o 10 coeficientes, el filtro reacciona más rápido a cambios en la velocidad, pero suaviza menos los ruidos.
- Con 20 o más coeficientes, se logra un mayor nivel de suavizado, pero con una respuesta más lenta ante cambios rápidos en la velocidad.

el micromouse usa el filtro FIR para eliminar ruidos o fluctuaciones de la señal de velocidad real, proporcionando un valor más estable y confiable, que es fundamental para el control de los motores por medio de PID.

Control PID: Para mantener la velocidad deseada de cada motor, se implementó un controlador PID que ajuste la velocidad de los motores en base al error con respecto a la velocidad calculada, para minimizar errores en la trayectoria.

5. ImplementacionControl PID:

- **Inicialización del PID:** Se configuraron los parámetros K_p , K_i , K_d con una serie de pruebas un límite de saturación del error acumulado.
- **Actualización del PID:** El cálculo del control PID se hace por medio del uso de dos parámetros fundamentales. El error de la velocidad y un delta de tiempo constante para un óptimo cálculo de la velocidad por medio de los pulsos generados por los encoders cada delta de tiempo.

- a) Proporcional:

$$P = K_p \cdot \text{error}$$

- b) Integral:

$$I = K_i \cdot \int \text{error} dt$$

Este término se acumula y se limita para evitar sobrecarga.

- c) Derivativo:

$$D = K_d \cdot \frac{\Delta \text{error}}{\Delta t}$$

El resultado se combina para generar la salida:

$$\text{Salida PID} = P + I + D$$

La salida que arroja el PID se suma a la velocidad actual de los motores para llegar a la misma velocidad calculada.

6. **Transmisión de Datos:** Se emplea una comunicación serial Uart y un TTL. Este método permite la transmisión de variables en tiempo real del sistema con el fin de hacer los ajustes necesarios

4. RESULTADOS

El micromouse analiza su entorno con los sensores infrarrojos y según la información obtenida, decide avanzar o girar. Calcula el perfil de velocidad de forma adecuada para el movimiento requerido sin embargo se comoetieron los siguientes errores en su desarrollo.

- **Delta de tiempo en integración de funciones:** Surgieron inconvenientes al controlar el delta de tiempo para la integración en múltiples funciones, lo que dificultó la sincronización adecuada entre las mediciones de velocidad y las etapas del movimiento.
- **Integración de distancia real y perfil de velocidad:** Inicialmente, se estaba trabajando con tiempo acumulado para calcular los cambios en el perfil de velocidad. Sin embargo, esto resultó inadecuado, ya que no se comparaban las etapas del perfil en función de la distancia real recorrida por el micromouse. Esto llevó a ajustes para asegurar que los cálculos de distancia y velocidad estuvieran correctamente integrados.

Análisis de desempeño: El micromouse demostró capacidad para optimizar su velocidad en diferentes etapas del perfil trapezoidal. Sin embargo, los problemas iniciales relacionados con la distancia real destacan la importancia de calibrar cuidadosamente cada etapa del perfil y el cálculo de los coeficientes de PID.

La resolución del laberinto: Para resolver el laberinto de forma eficiente, es fundamental implementar un algoritmo como Flood Fill, que permita al micromouse determinar la mejor ruta hacia el objetivo basándose en el análisis de los caminos disponibles.

5. CONCLUSIONES

1. **Fiabilidad en entornos controlados:** Aunque el micromouse opera eficientemente en condiciones ideales, sería interesante evaluar cómo integrando un algoritmo de resolución y el control mejorado se desempeñan en entornos mas demandantes.
2. **Calibración detallada:** La calibración de los sensores infrarrojos y encoders permitió medir distancias con exactitud, lo que fue clave para calcular la odometría y asegurar movimientos controlados.
3. **La implementación del filtro FIR** garantizó una mayor precisión en el error arrojado por el PID, demostrando la importancia del suavizado de señales.
4. **El uso del controlador PID y función feedforward** junto con el perfil de velocidad trapezoidal, permitió movimientos más suaves, estables y precisos, reduciendo los errores de trayectoria.

Apéndice

Figura 1: Esquemático del robot micromouse

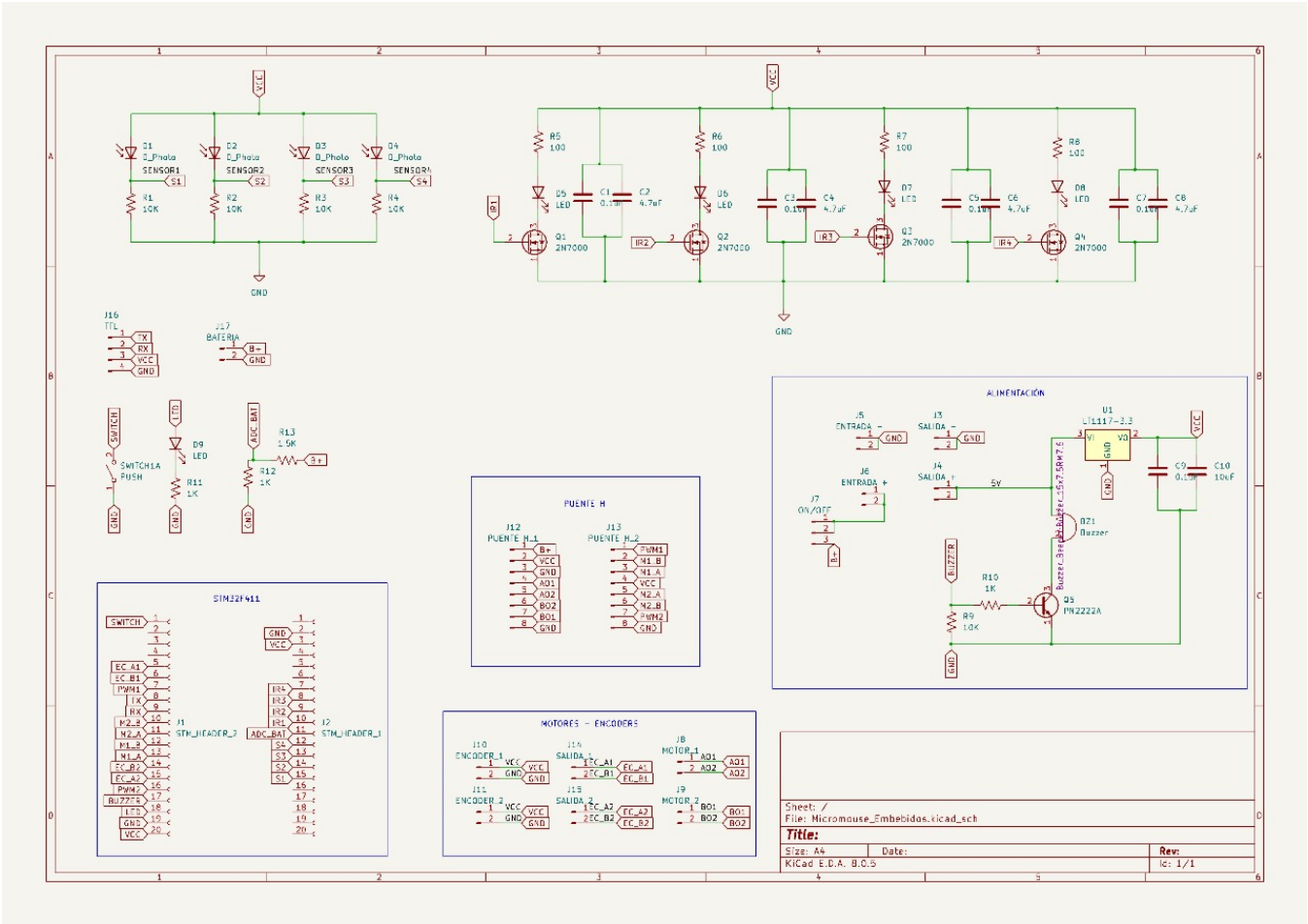


Figura 2: PCB del robot micromouse

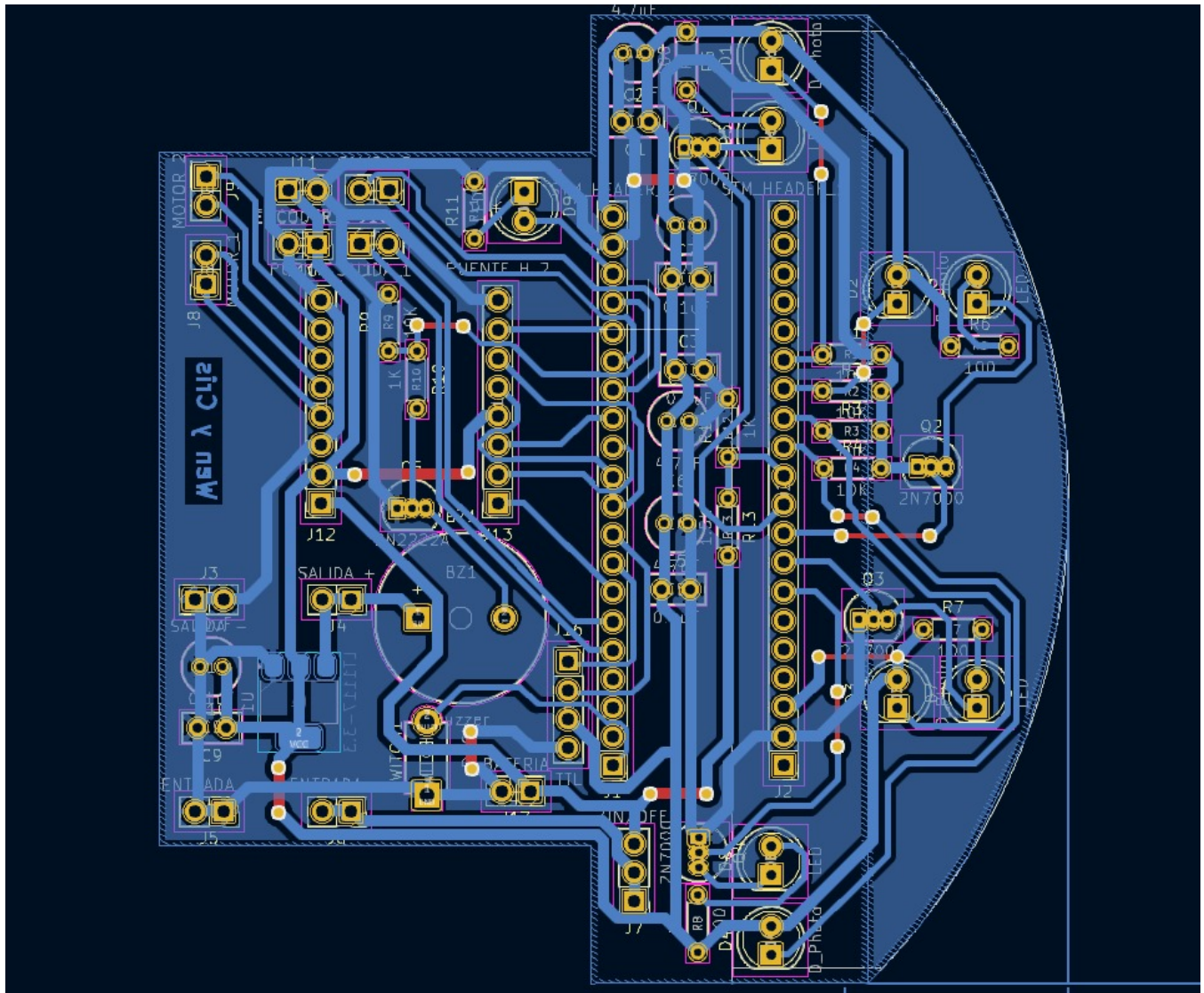


Figura 3: *ADC Vs mm*

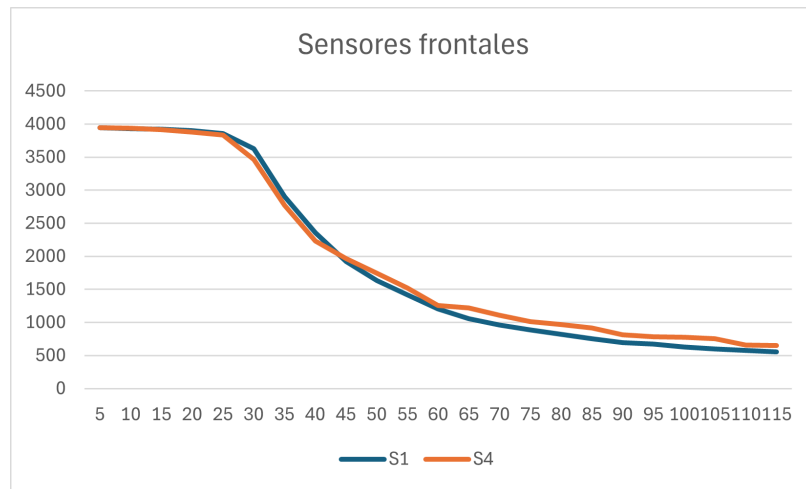


Figura 4: *mm/s Vs Volts*

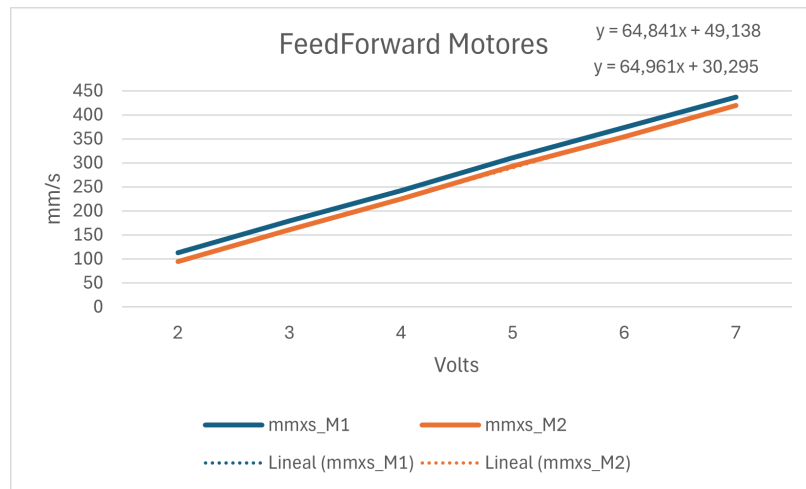


Figura 5: 7 Coeficientes IR

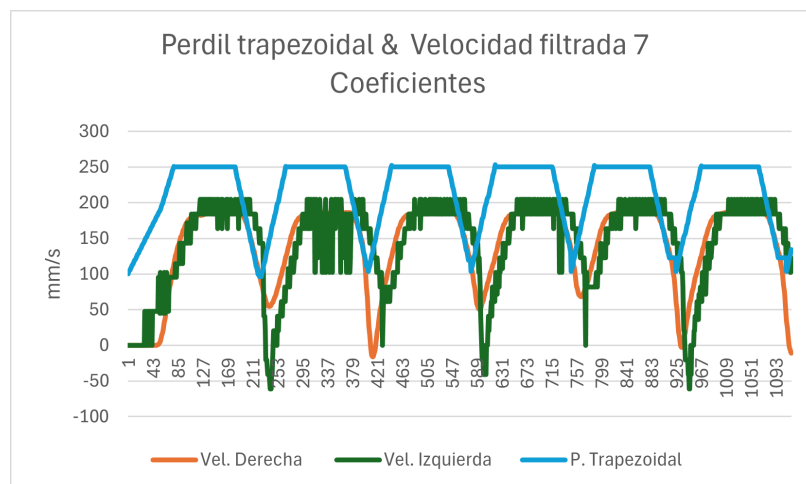


Figura 6: 15 Coeficientes IR

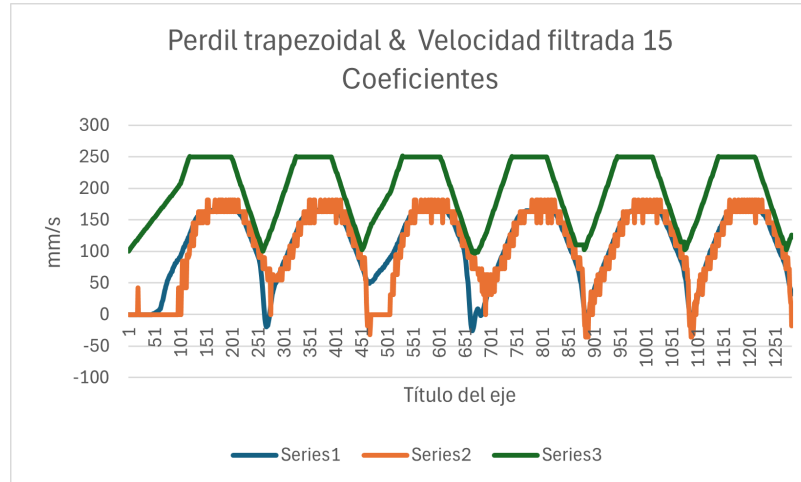


Figura 7: 20 Coeficientes IR

