



Diseño e implementación de un Micromouse.

Daniel Jimenez, Juan Sebastián Herrán Páez

December 13, 2024

Contents

1	Resumen	2
2	Contexto	3
3	Diseño e implementación del Micromouse	4
3.1	Diseño del esquemático y PCB	4
3.2	Problemas del diseño e implementación	5
3.2.1	Problemas del esquemático y la PCB	5
3.2.2	Soldadura, pistas, cables, cortocircuitos, encoders, mo- tores, bloqueo STM, corriente, sensores, transistores	7
3.3	Implementación final del Micromouse	8
4	Resultados	10
5	Conclusiones	13
5.1	Conclusiones Generales	13
A	Anexos	14
A.1	Vistas Micromouse	14

Chapter 1

Resumen

En este proyecto se hizo el diseño implementación y pasos a seguir para la programación del Micromouse. Se hizo el diseño del esquemático y la PCB, luego se hizo la implementación del diseño PCB artesanal, en una Baquelita de forma que se imprimió el diseño en una impresora laser con un papel termotransferible, se plancho el papel a la Baquelita y por último se sumergió en la Baquelita en agua mezclada con un polvo químico para procesar circuitos impresos en Baquelitas.

Se realizaron varios intentos de la ubicación de los componentes y se plancho varias veces, de forma que cuando se soldo fue mucho más fácil que la primera vez que se intento soldar todos los componentes, se realizaron pruebas de las soldaduras, continuidad mayoritariamente.

Finalmente con la PCB funcionando físicamente se hizo la construcción del Micromouse, de forma que se utilizaron varios materiales para adaptarlos a la construcción del mismo y se hicieron pruebas de los motores, sensores IR, entradas y salidas del microcontrolador, componentes secundarios, de los encoders y al final de todos los problemas se pudo implementar el hardware de manera que funcionó sin ningún error luego de las pruebas. Se pudo realizar una parte pequeña de la programación en la que se hicieron pasos básicos para lograr la implementación del perfil de velocidad trapezoidal.

Chapter 2

Contexto

A lo largo del semestre se aprendió sobre una materia crucial para la carrera de Ingeniería Electrónica la cual es Sistemas Embebidos. En esta materia se aprendió una gran variedad de utilidades para la vida profesional de un ingeniero electrónico como: el funcionamiento de un lenguaje de programación popular llamado C y C++, las partes básicas de un microcontrolador, el software STM32CubeIDE para la programación de la tarjeta que se utilizó durante todo el semestre y muchas formas de implementar y hacer útil un microcontrolador.

Chapter 3

Diseño e implementación del Micromouse

3.1 Diseño del esquemático y PCB

Se realizó el diseño de un esquemático como parte base para el funcionamiento del Micromouse, este diseño es crucial ya que se debe tener en cuenta todas las funciones que se van a realizar para el Micromouse, tanto las conexiones del microcontrolador que se utilizó en este caso un STM32F411CEU6, puente H, regulador, sensores, como para las conexiones de todos los demás componentes secundarios. A continuación se muestra una imagen del esquemático final utilizado en la Figura 3.1.

Este esquemático incluye todos los componentes como el microcontrolador utilizado para este Micromouse, sensores IR, tanto emisores como receptores, un puente H para el manejo de los motores, un regulador que será el VCC principal de la PCB, un buzzer, un led, un switch para el encendido y apagado del circuito y sus conexiones para que el Micromouse pueda funcionar correctamente sin ningún problema y con los componentes apropiados pensados para su objetivo principal el cual es resolver el laberinto.

Después de haber realizado el esquemático se procedió a diseñar las ubicaciones, pistas y demás características para el diseño de la PCB. A continuación se presenta el diseño de la PCB tanto de las pistas Figura 3.2 como el diseño 3D final Figura 3.3.

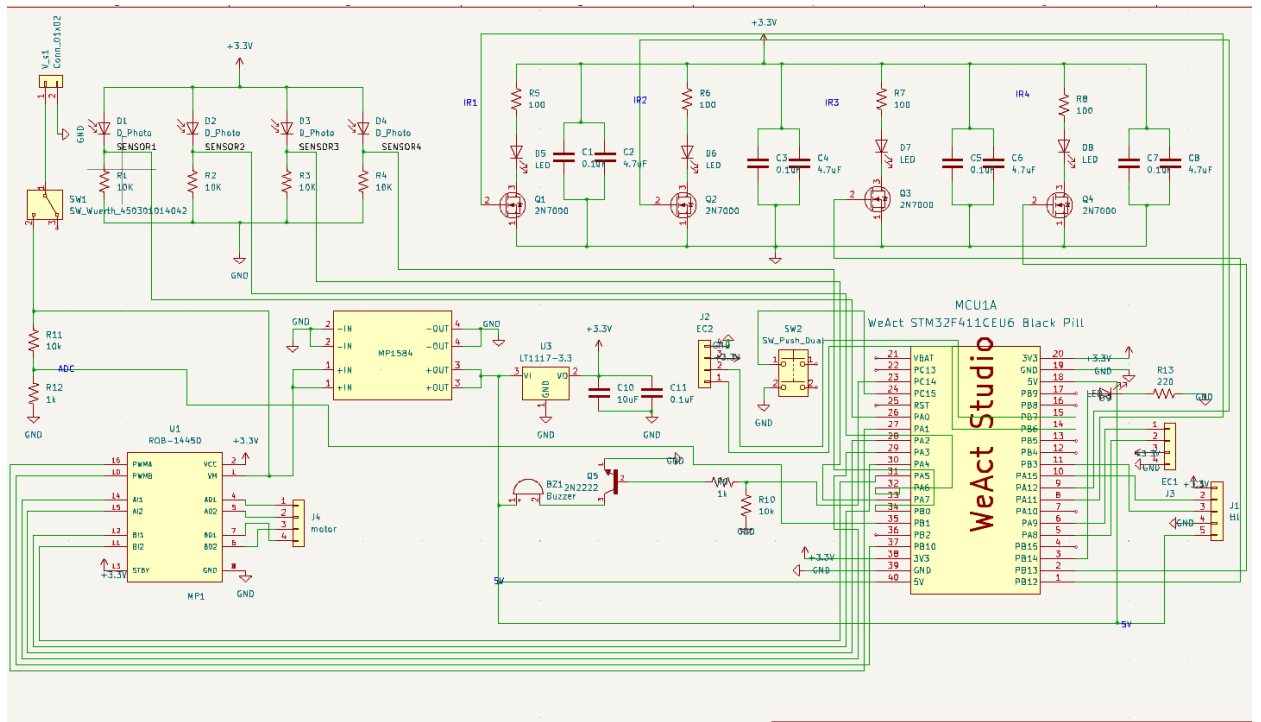


Figure 3.1: Esquemático final

3.2 Problemas del diseño e implementación

3.2.1 Problemas del esquemático y la PCB

Estos problemas realmente fueron los que conllevaron a una cadena de problemas que también generaron más problemas. Esto puesto a que si no se tiene de partida una solución apropiada y que facilite los siguientes pasos para el objetivo final, surgirán una cantidad de problemas que llevarán a una perdida de tiempo. Los problemas comunes para el caso de este proyecto del diseño del esquemático y de la PCB fueron:

Problemas con el diseño de algunas huellas y symbols que se intentaron buscar y no se adecuaban al circuito, la falta de conocimiento sobre los headers y la ubicación de los sensores que se pensó que tenía que ser lo más cercanos a los transistores condensadores y resistencias, de modo que se tuvieron que añadir bastantes conexiones que al final terminaron siendo cables y a su vez esto dificultó la soldadura para muchos componentes y una cadena de cortocircuitos que dificultaban aún más el funcionamiento correcto de la

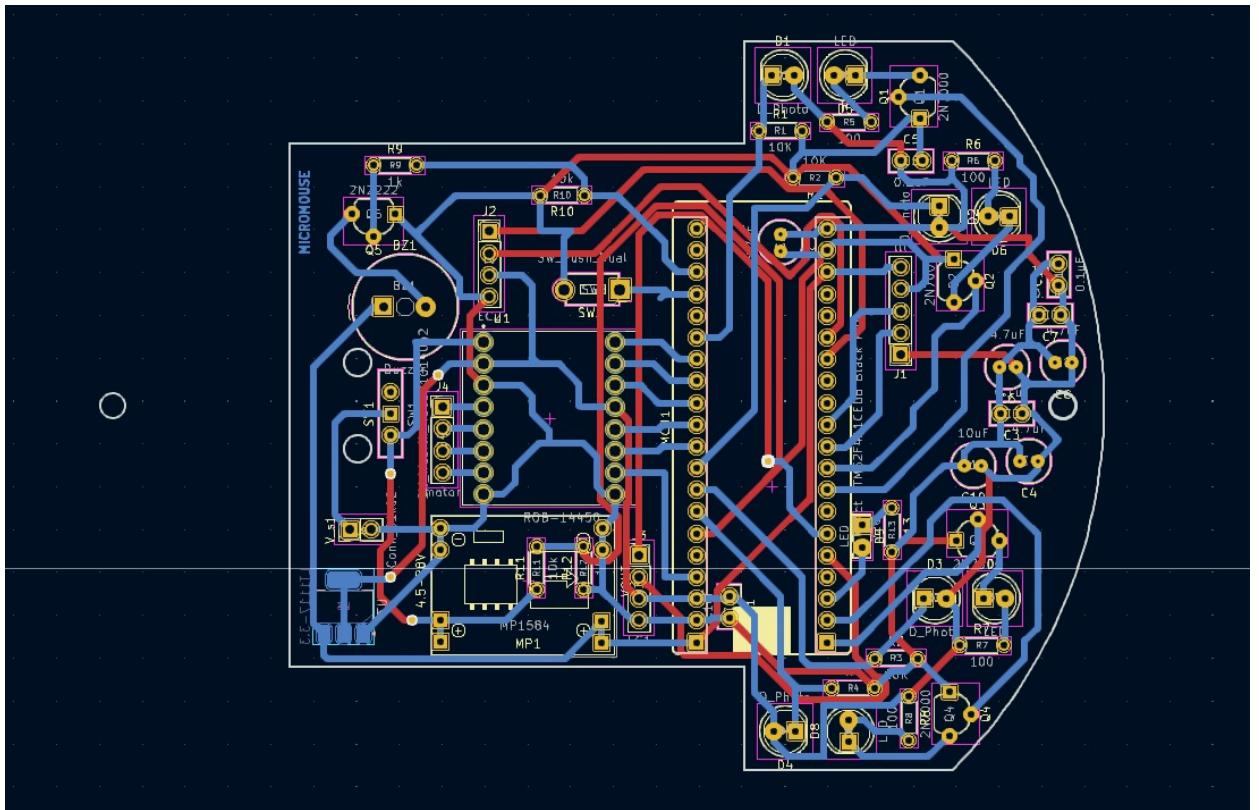


Figure 3.2: Diseño PCB pistas

PCB.

Finalmente después de una cantidad de problemas bastante extensa en la primera implementación de la PCB con componentes soldados y un diseño de la ubicación de los componentes de la PCB bastante limitada por cuestiones de tamaño se realizó una segunda versión (aparte de las versiones descartadas) que se implementó y se planchó hasta 5 veces con el objetivo de que todas las pistas quedaran adheridas correctamente para que no se tuviera problemas de cortocircuito entre las pistas o que fueran problemas mínimos y fáciles de corregir.

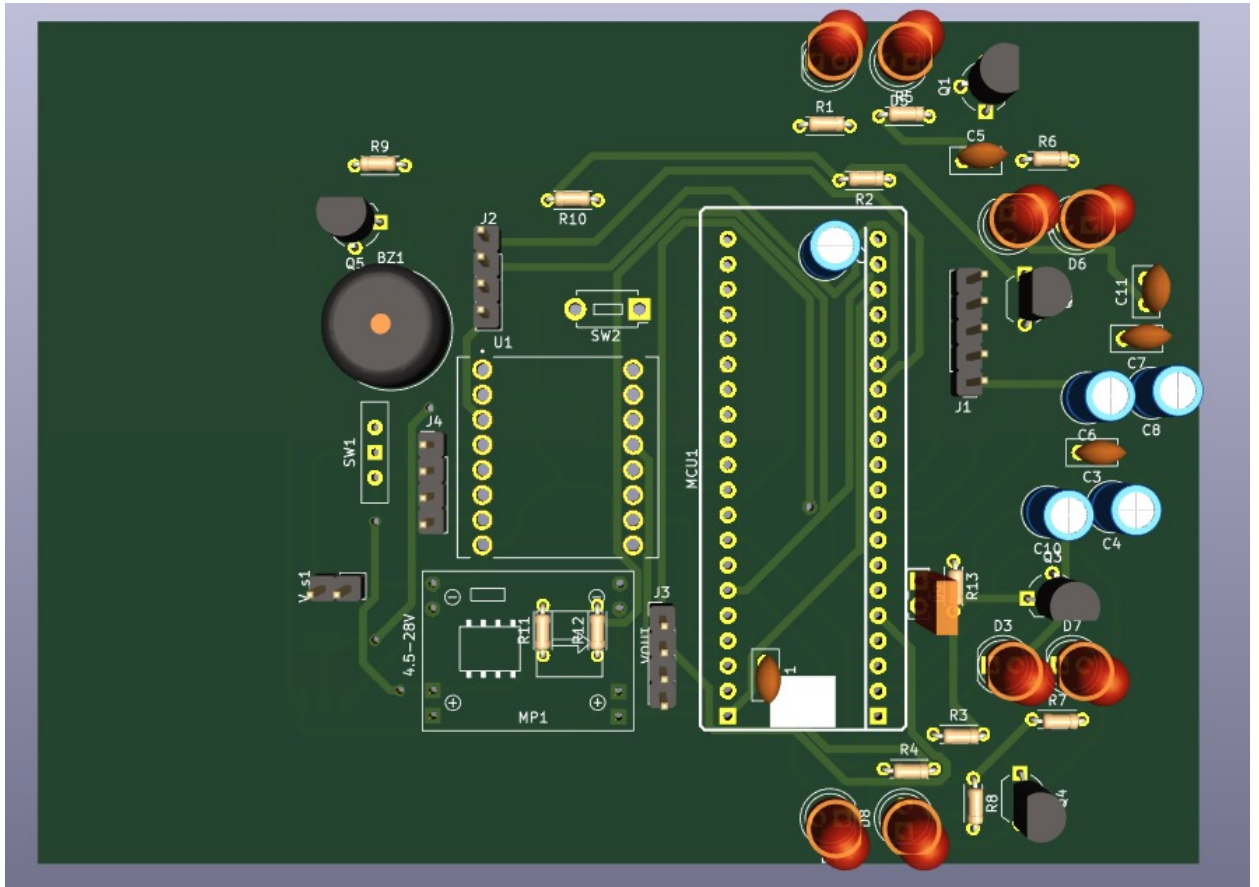


Figure 3.3: Diseño PCB 3D

3.2.2 Soldadura, pistas, cables, cortocircuitos, encoders, motores, bloqueo STM, corriente, sensores, transistores

Problemas de este tipo sucedieron todo el tiempo, principalmente en la primera implementación fallida del Micromouse, puesto que a esta implementación fue a la que le dedicamos más tiempo, tiempo que al final después de tantos intentos fue desperdiciado. De igual forma después de lograr correctamente la funcionalidad de la PCB se presentaron errores en los encoders que al final después de hacer pruebas de posibles errores se encontró que en la tarjeta que se tiene para los encoders en una tarjeta de las 2 que se tenía se desoldo un sensor que funciona para la lectura de estos encoder, de modo que solo generaba resultados de 0 y 1 es decir no contaba los pulsos correctamente.

Posiblemente el bloqueo de la primera STM utilizada fue debido a un corto que se hizo midiendo con el multímetro y al moverse un poco se cortocircuitó. El segundo bloqueo que generó la inhabilitación de la segunda STM se cree que fue por un error al poner los cables que se conectan al encoder, pues un cortocircuito entre gnd y vcc parece que fue el posible problema o quizás debido a la potencia que requería el motor cuando se estaba probando los pulsos del encoder.

Adicionalmente errores o falta de recursos que no se podían conseguir de forma rápida debido a su costo elevado también generó pérdida de tiempo que se podía haber aprovechado si se hubiese tenido ciertos materiales a la mano.

En realidad de estos problemas se dedicó mucho tiempo y eso perjudicó en gran medida el progreso del Micromouse. Aunque hubieron infinitas de problemas nunca se quiso abandonar el proyecto, pues realmente se quería llegar al objetivo final que es resolver el laberinto.

3.3 Implementación final del Micromouse

Aunque surgieron una variedad de problemas innumerables se logró obtener la implementación final del Micromouse Figura 3.4 de modo que funcionó correctamente la PCB, los motores, encoders, sensores, buzzer, botón auxiliar, switch para el encendido y apagado del circuito y demás funciones sin necesidad de mover nada.

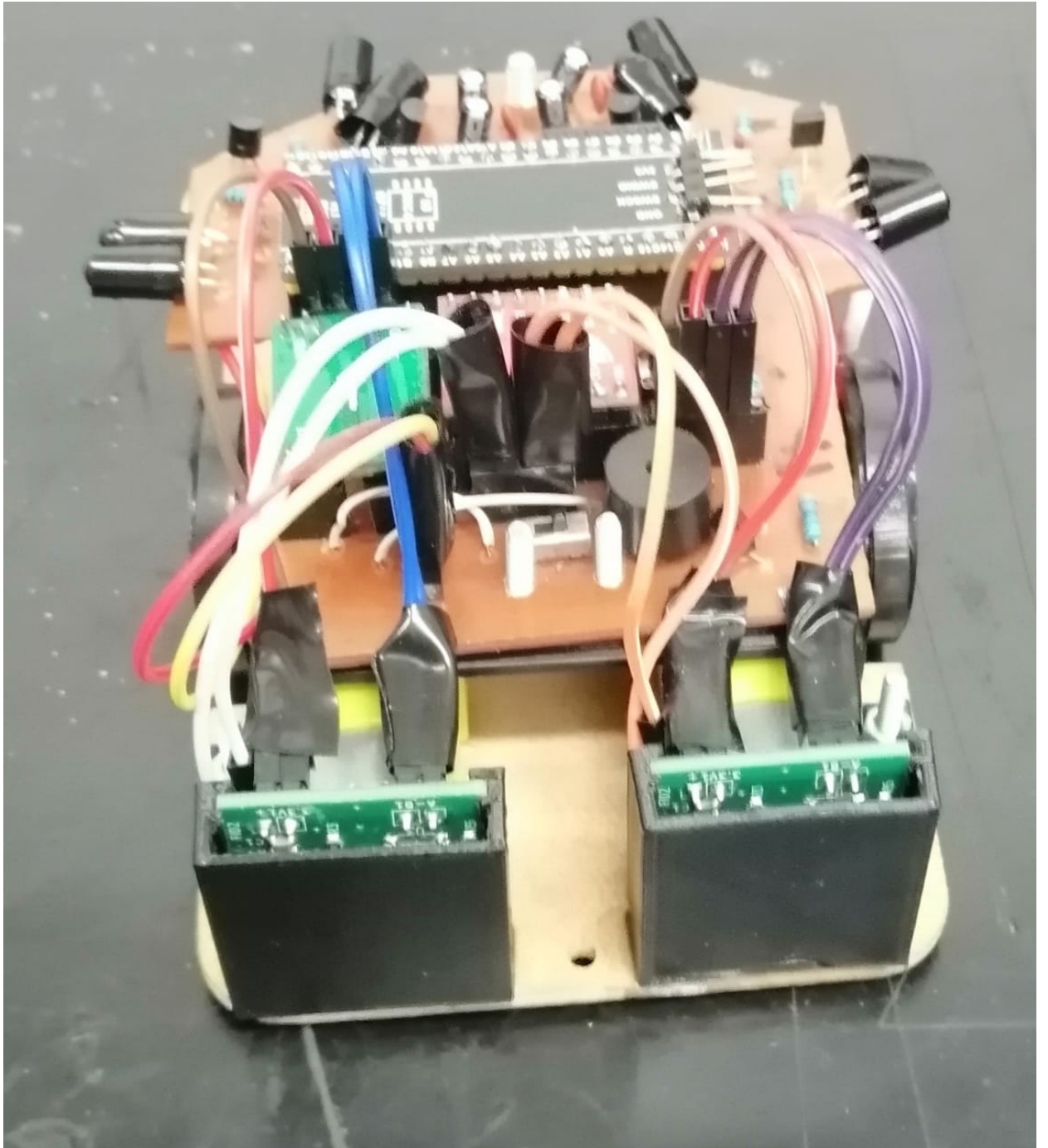


Figure 3.4: Implementación final Micromouse

Chapter 4

Resultados

A pesar de los problemas mencionados anteriormente, se comprobó el buen funcionamiento del producto, haciendo pruebas de funcionamiento de los motores, encoders, sensores, etc, llevando a la parte de programación del Micromouse. Por el corto tiempo que se dispuso para la programación, se logró obtener el cálculo de la velocidad en milímetros por segundos de cada motor, además de establecer las variables clave para poder realizar una función para establecer la velocidad requerida en cada movimiento que hiciera el Micromouse, para esto, se tienen las siguientes gráficas (Figura 4.1 y 4.2) que representan la velocidad máxima de cada motor, según su voltaje de entrada.

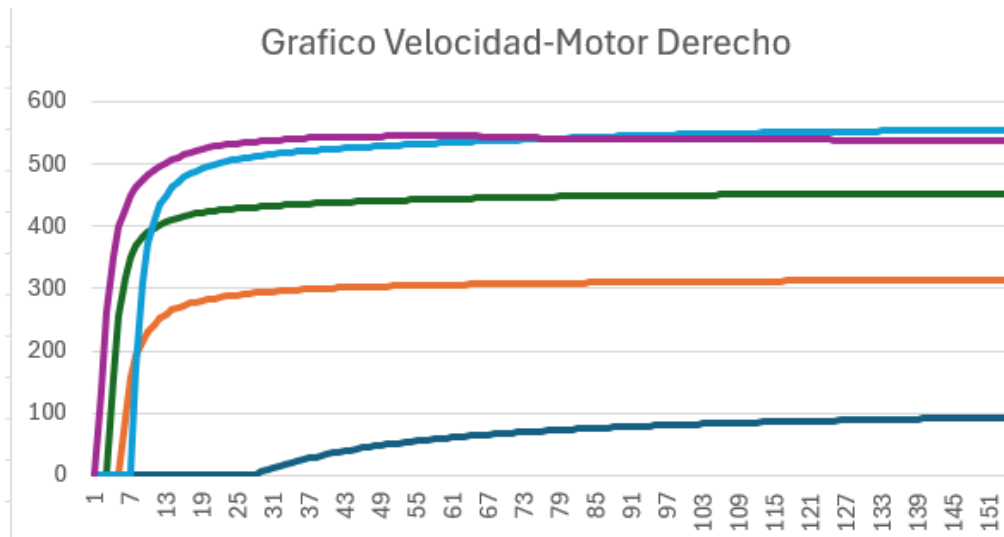


Figure 4.1: Gráfica de velocidad-Motor derecho

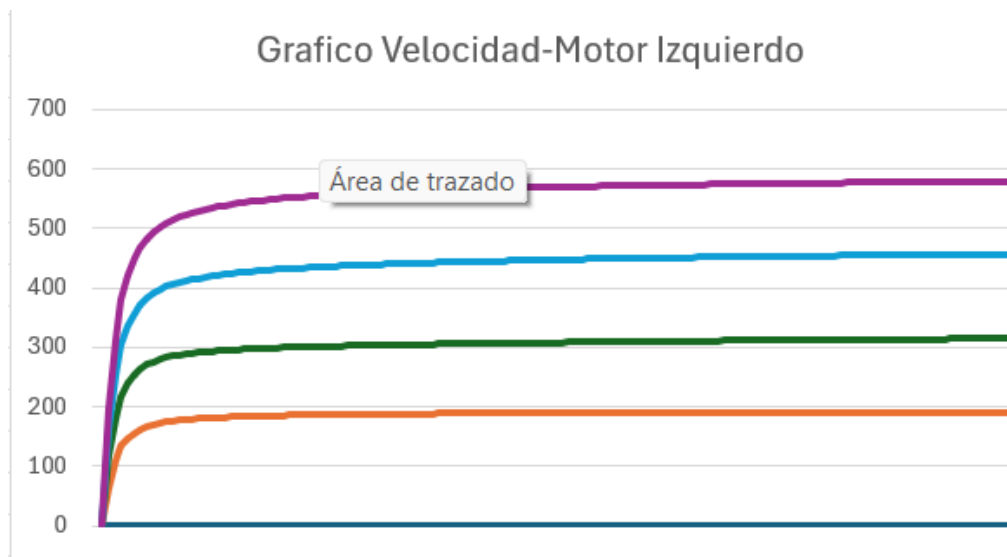


Figure 4.2: Gráfica de velocidad-Motor Izquierdo

Como se puede observar, los motores se comportan de manera distinta, incluso de manera irregular, observando el motor derecho que se visualiza un comportamiento no regular en comparación al motor izquierdo, esto es debido a la construcción física de cada motor reductor implementado en el diseño final de micromouse.

Al tener esta gráficas, se pueden sacar valores puntuales para construir un modelo matemático que relacione la velocidad del motor con su voltaje aplicado, para así poder tener un control mas preciso al hacer su recorrido, esto también requisito para poder tener un perfil de velocidad trapezoidal, esto para controlar la aceleración y la desaceleración en cada movimiento del Micromouse.

Al tener el control de los motores listo, se iría a revisar la parte de los sensores IR, que son los ojos del Micromouse, se colocaron dos sensores mirando hacia el frente, y otros dos mirando a cada lado del Micromouse, estos sensores nos ayudan a medir la distancia que tienen el robot con respecto a las paredes del laberinto,, esto se implementaba a partir de las mediciones análogas que se obtenían de los pines ADC configurados. Al igual que los motores, se le hacia mediciones a cada uno de los sensores para establecer un modelo matemático que relacionaba un valor ADC con una distancia, para que el Micromouse pudiera frenar a cierta distancia de una pared, además de

poder corregir su dirección si se acercaba demasiado a una pared por los lados.

Para tener un mejor control en la velocidad y en la medición ante las colisiones, se aplica un controlador PID, esto con el fin de controlar mejor la velocidad de los motores y al establecer la distancia adecuada para que el Micromouse no chocara con ninguna pared.

Ya teniendo todo esto implementado, se le incluía al Micromouse una lógica para la resolución del laberinto, esto llamado Flood-Fill, este algoritmo de búsqueda ayuda al Micromouse a encontrar la solución más sencilla al laberinto, esto basado en árboles de búsqueda implementado en cada celda del laberinto.

Chapter 5

Conclusiones

5.1 Conclusiones Generales

El aprendizaje más valioso de esta materia fue la aplicación del conocimiento de muchas áreas de todo tipo aplicado para la resolución de un problema, de forma que se desarrolló un ejercicio mental para llegar a la solución de un problema, pensando de una forma que se modelaron soluciones propias adaptadas para cada problema específico, se aprendió mucho el pensamiento de resolución de problemas y se descubrieron habilidades que se pudieron implementar en el transcurso del semestre.

Se puede concluir que al realizar un proyecto de este calibre se tiene que tener en cuenta mucho la disposición del tiempo y saber aprovecharlo al máximo, esto para poder desarrollar y cumplir con los objetivos de manera mas eficaz y no tener los problemas como se llegaron a atener en el desarrollo de este proyecto del Micromouse.

Se puede llegar a resolver el laberinto con el Micromouse actual, pero todavía se tiene que implementar y hacer pruebas de los siguientes pasos en terminos de la programación, para llegar al objetivo de resolver el laberinto.

Appendix A

Anexos

A.1 Vistas Micromouse

Se incluyen vistas adicionales del Micromouse.

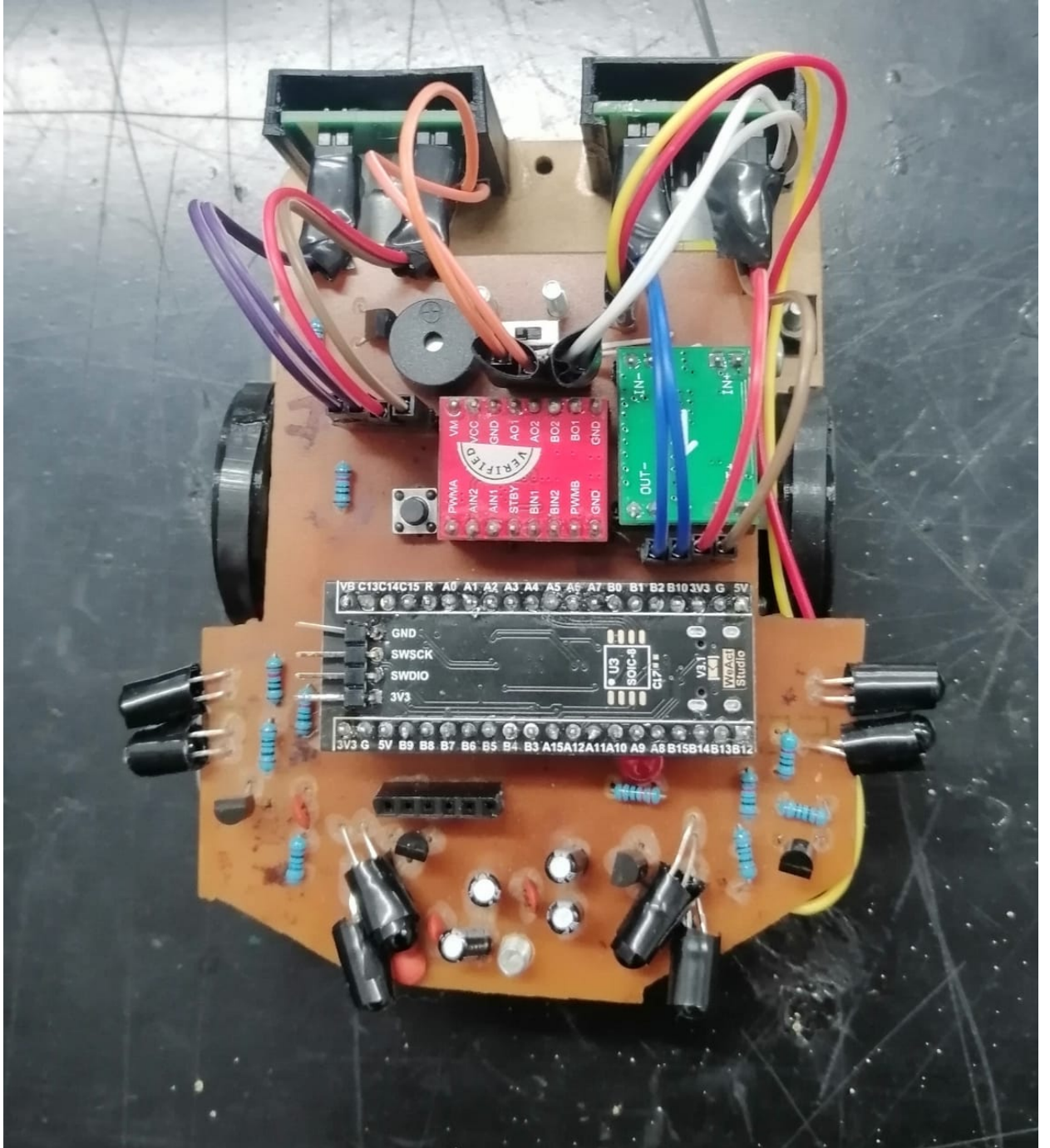
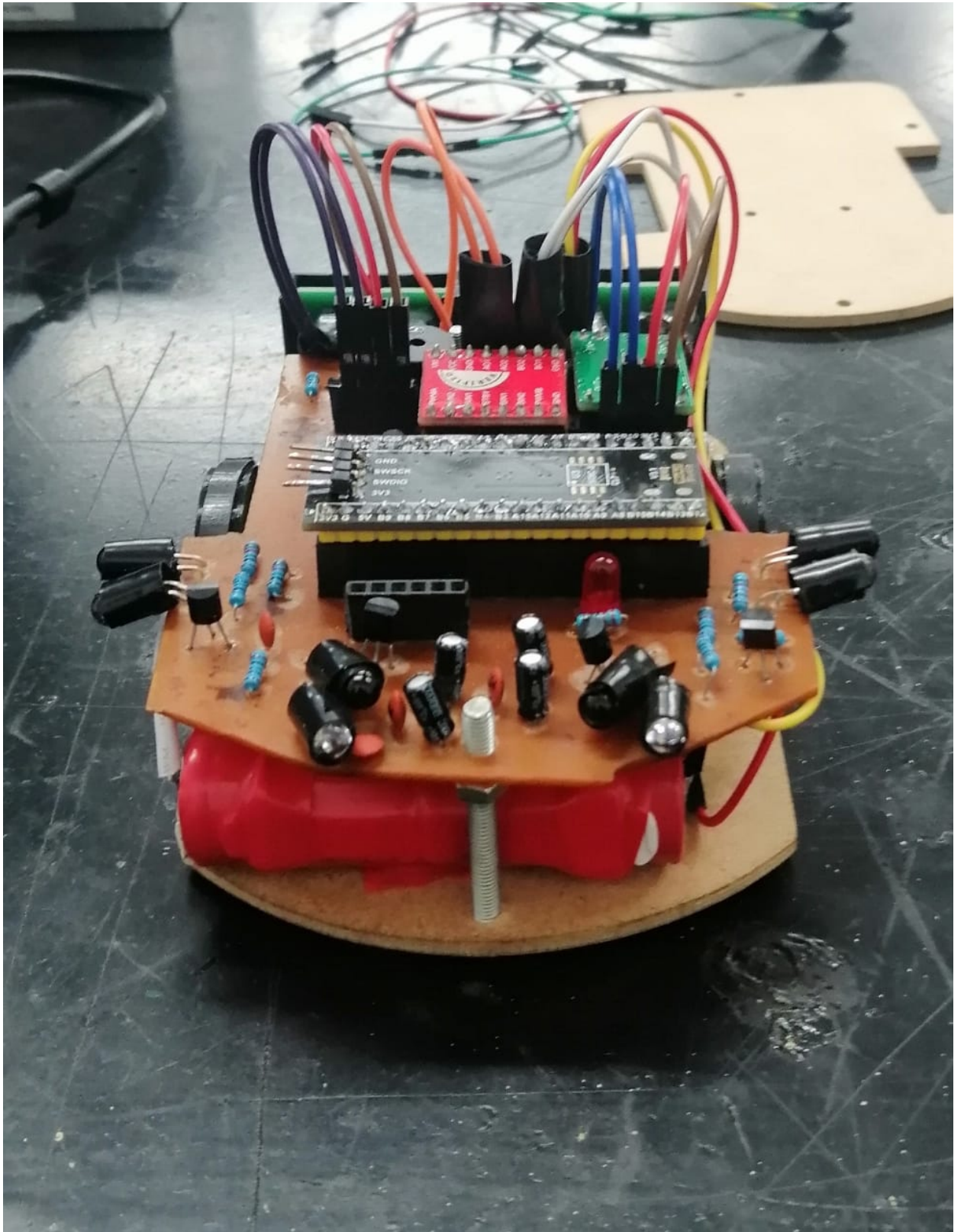


Figure A.1: Micromouse vista 1



16

Figure A.2: Micromouse vista 2