

---

---

INSTITUTO TECNOLÓGICO JOSÉ MARIO MOLINA PASQUEL Y HENRÍQUEZ

UNIDAD ACADÉMICA CHAPALA



TECNOLÓGICO SUPERIOR DE JALISCO  
MARIO MOLINA

Ingeniería Mecatrónica | Robótica

Reporte de Proyecto

**BattleBot “Buggy”**

Número de control y Nombre de los Alumnos.

Jorge Satomi Minami Aguilera NC:18031197

Jennifer Cassandra González Osuna NC: 18031192

María del Carmen Ramírez Rosales NC: 18031206

Karen Samantha Padilla Mojarrero NC:18031201

Hugo Alberto Henríquez Rentería NC. 17030151

9° Semestre

Docente: Omar Pirul Mendo

*Chapala, Jalisco.*

*Diciembre 2022.*

---

## Preliminares

### 2. Agradecimientos

Hugo Alberto Enríquez Rentería

Primero que nada agradezco a todos los profesores en general por haberme compartido sus conocimientos, que los cuales fueron muy útiles para realizar este proyecto, ya que si no fuera por ellos no hubiéramos tenido idea de cómo hacer este prototipo, Gracias al Profesor Omar Pirul por habernos impartido la materia de electrónica Analógica y digital que la cual fue muy necesarias implementar todo el conocimiento en este proyecto, al Profesor Víctor Gonzalez por habernos impartido las materias de programación básica, Micros, y embebidos que fue donde sacamos lo mejor de nuestros conocimientos al momento de elaborar circuitos ya profesionales y que al implementar todo en este proyecto ya es más fácil para nosotros llevarlo a cabo, al Profesor Martin Rojas por habernos impartido la materia de Auto CAD que gracias a esa materia podemos fabricar algunas piezas como engranajes, poleas, etc. que tengamos que crear a nuestras necesidades. Al profesor Brandon Malagón por habernos dado varios consejos, cursos e ideas para mejorar nuestro proyecto, y cómo utilizar el EasyEDA que el cual es un software muy necesario para crear placas electrónicas que necesitemos tener para el funcionamiento de nuestros proyectos y además de cómo nos orientó para manejar el torno y así poder desbastar, hacer roscas, orificios y demás cosas. Gracias al profesor Juan Carlos Desales por habernos impartido la materia de potencia la cual para este proyecto fue muy necesaria y requerida ya que manejamos corriente eléctrica que en otros proyectos no habíamos implementado. Al profesor Gerardo Rodríguez por haber impartido la materia de mecánica ya que al igual como este proyecto es un robot se necesitó utilizar algunos engranajes y flechas, al igual que mucha fuerza para el movimiento de este. Agradezco a todos mis compañeros por haber compartido algo de sus conocimientos que tienen al igual que todas las ideas buenas para la mejora del robot y sobre todo el compañerismo y el apoyo que hemos llevado a cabo a lo largo de los años.

Jorge Satomi Minami Aguilera

Estoy muy agradecido y doy gracias a mis profesores del semestre cursado (Omar Pirul, Martin Rojas y Víctor Villanueva) por haberme brindado su apoyo y conocimiento para la realización del presente proyecto, además de agradecer a Omar Orozco por compartirnos sus conocimientos en procesos de fabricación y de materiales para la realización de nuestra arma, por haber ofrecido su ayuda para elaborar la misma, de igual manera quiero agradecer al profesor Brandon Malagón por haberme enseñado a utilizar de manera correcta el torno que posee la Universidad, al Profesor Jesus González por haberme compartido su ayuda y conocimientos en las distintas etapas de elaboración del proyecto.

---

---

También agradezco a mis compañeros de equipo, Jennifer, Maricarmen, Samantha y Hugo por su apoyo y ayuda en cada etapa del proceso que conlleva la elaboración de nuestro proyecto, por haberme tenido paciencia y tolerancia pues sin ellos sería extremadamente difícil haber llegado a los resultados que conseguimos en nuestro proyecto, los cuales considero grandiosos.

También agradezco a Molly y a Oslo (mis perros) por haber sido mis compañeros fieles y pilares de apoyo en todo este proceso.

#### Jennifer Cassandra González Osuna

Quiero agradecer a mis profesores del semestre Omar Pirul, Victor Villanueva y el profesor Martin Rojas por el impulso que nos dan al realizar esta clase de proyectos más avanzados, ya que gracias a ellos nos involucramos y envolvemos en diversos temas retándonos a nosotros mismos y a nuestro aprendizaje, lo cual nos ayuda a expandir más nuestros conocimientos y a mantenernos en constante aprendizaje.

También agradezco a mis compañeros del proyecto, Maricarmen, Samantha, Satomi y Hugo que sin ellos hubiera sido imposible realizar el robot sola, gracias por su ayuda, paciencia y perseverancia.

A mi padre por su apoyo moral en cada proyecto que realizó y que cada día sigue impulsándome para poder llegar más lejos, aceptando cada reto y siempre dar lo mejor de mí para seguir mejorando día con día.

A mis amigos e ingenieros Alejandro Martínez y Erick Rubén por su constante apoyo y por compartirme de sus conocimientos.

Por último, y no menos importante, a Sparky gracias por ser mi acompañante y amigo incondicional que siempre está ahí acompañándome en las noches de desvelo.

#### Maria del Carmen Ramírez Rosales

Agradezco a mis profesores que me impartieron en este semestre (Martin Rojas, Omar Pirul, Juan Carlos Desales, Víctor Villanueva) por darnos el espacio en sus materias para poder llevar a cabo nuestro proyecto.

También Agradezco a mis compañeros de equipo, (Jennifer, Samantha, Satomi y Hugo), ya que sin ellos no hubiera sido posible terminar este proyecto, cada uno pusimos nuestro granito de arena para poder terminarlo.

---

---

Y finalmente y no menos importante a mi familia ya que sin ellos no me sería posible seguir estudiando. Gracias por su apoyo y cuidado de mi hija.

Karen Samantha Padilla Mojarro

quiero agradecer a mis compañeros de equipo que si ellos un proyecto de magnitud tan grande como lo es un robo de pelea no se hubiera podido lograr si ellos y sus diferentes habilidades y conocimientos, también quiero agradecer a los profesores que nos impartieron dichos conocimientos ya que sin ellos nada de lo aquí mencionado hubiese sido posible, el profesor Martin Rojas, Víctor Gonzalez, por último pero no men0oos importante Omar Pirul, que nos apoyaron en todo momento durante el proceso de realización de dicho proyecto.

También quiero agradecer a mis padres que me apoyan moral y económicamente para poder realizar este tipo de retos y proyectos.

GRACIAS...

---

### 3. Resumen

En el presente documento se muestra el desarrollo de un robot de combate y todo su proceso de creación desde los bocetos, las ideas hasta un diseño terminado junto con una simulación del robot en software de autoCAD. Al igual que la programación utilizada en lenguaje de programación C++ en el IDE Arduino con un microcontrolador atmega328p, el cual cumple con los requisitos necesarios para el funcionamiento deseado del robot, tal como es el uso de un control de radiofrecuencia para el uso inalámbrico del robot.

Se muestra todo el proceso llevado a cabo en la construcción del robot, la parte mecánica, las piezas y materiales utilizados, así como los circuitos diseños para su uso, las simulaciones realizadas para verificar conexiones y posibles fallas o cortocircuitos entre otras. Se detallan los drivers utilizados para el control y alimentación de los motores para mover el robot, así como el motor del arma.

A lo largo de todo el se encuentran conclusiones, fallas y problemáticas que hubo durante el desarrollo del prototipo, así como diversas modificaciones que se realizaron y cambios imprevistos, todo esto para garantizar un buen funcionamiento del robot.

Y por último, se encuentran algunos anexos, en los que abriendo las ligas se puede dirigir a los videos de construcción y funcionamiento del robot.

---

---

## 4. Índice

### Contenidos

Preliminares	2
2. Agradecimientos	2
3. Resumen	5
4. Índice	6
Generalidades del proyecto	7
5. Introducción	7
6. Descripción de la empresa u organización y del puesto o área del trabajo del estudiante	10
7. Problemas a resolver	11
8. Objetivos	11
<b>Objetivo general:</b>	11
<b>Objetivo específico:</b>	11
9. Justificación	12
Marco Teórico	13
10. Marco Teórico	13
Especificaciones:	18
Desarrollo	26
11. Procedimientos y descripción de las actividades realizadas	26
Resultados	72
12. Resultados, planos, gráficos, prototipos, manuales, programas análisis estadísticos, modelos matemáticos, simulaciones, normatividades, regulaciones y restricciones, entre otros.	72
13. Actividades sociales realizadas en la empresa u localización	83
Conclusiones	86
14. Conclusiones del proyecto, recomendaciones y experiencia profesional personal adquirida	86
Competencias desarrolladas	87
15. Competencias desarrolladas y/o aplicadas	87
Fuentes de información	89
16. Fuentes de información	89
Anexos	90

---

## Generalidades del proyecto

### 5. Introducción

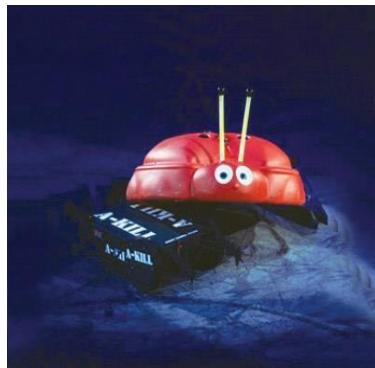
El final de la década de 1990 fue una época turbulenta para el combate de robots. Marc Thorpe, el fundador de la popular competencia Robot Wars, se vio envuelto en batallas legales por la marca Robot Wars. Se planearon varios eventos en San Francisco, CA y luego se cancelaron debido a problemas legales. Sin embargo, finalmente, un par de constructores veteranos, Trey Roski y Greg Munson, recibieron el visto bueno legal para organizar un evento. Unos meses después, en agosto de 1999, se llevó a cabo la primera competencia oficial de BattleBots en Long Beach, CA.

El evento tuvo tres clases de peso: Kilobots (25-55 lbs con ruedas, 25-83 lbs con patas), Megabots (56-109 lbs con ruedas, 84-164 lbs con patas) y Gigabots (110-200 lbs con ruedas, 165-300 lbs patas). Tenga en cuenta que los bots con patas, como el impresionante Mechadon de Mark Setrakian de 450 lb (Figura 5.1), podían ser mucho más pesados que sus contrapartes tradicionales con ruedas. Debido a la dificultad que implica construirlos, sólo se inscribieron en la competencia unos pocos bots con patas.



*Figura 5.1 Mechado de 450 lb (Mark Setrakian, 1999).*

Estaban los robots divertidos/tontos obligatorios, como Stuffie, un megabot que lucía el armazón de un jeep a control remoto con un osito de peluche al volante, y Tentomushi (Figura 5.2), un adorable robot con temática de mariquita que se ganó a la multitud para reclamar victoria en dos partidos separados.



*Figura 5.1 Tentomushi*

El final de la serie de televisión en 2002 no fue de ninguna manera el final de BattleBots. Desde el final del programa, BattleBots ha seguido organizando torneos en todo el país. BotsIQ, una competencia patrocinada por BattleBots, se enfoca en enseñar robótica, así como otras áreas STEM a estudiantes de secundaria y preparatoria.

Hoy en la actualidad las competencias de robot de combate han mejorado de una manera impresionante y podemos ver que este evento se realiza cada año, con competidores nuevos o incluso equipos que regresan con la revancha con un robot completamente mejorado.

Uno de los robots de combate más poderosos en la actualidad es el SawBlaze, el cual fue utilizado como inspiración para crear el robot de este proyecto llamado “Buggy”, modificando cosas de diseño, pero manteniendo técnicas que el equipo original implementó a su SawBlaze para hacerlo uno de los más poderosos.

SawBlaze es un robot pesado construido por Jamison Go del Team SawBlaze, anteriormente Limites Robotics, que ha competido en todas las temporadas del reinicio de BattleBots desde la temporada 2 de ABC.

El capitán del equipo, Jamison Go, anteriormente formó parte del equipo JACD y Overhaul en la temporada 1 de ABC, pero se disolvió para encabezar su propio equipo en futuras temporadas de BattleBots. Al igual que el hijo de puta de la era de Comedy Central, SawBlaze fue diseñado para deslizarse debajo de los robots para atraparlos en su lugar, luego bajar un brazo con un disco giratorio montado en el extremo para causar daño desde arriba. También podría equiparse con varios accesorios de tenedor y pala dependiendo de su oponente. En temporadas posteriores, el armamento del disco de corte podría intercambiarse con un disco asimétrico que gira hacia abajo para dar mejores golpes, llamado "sierra de martillo". SawBlaze también se combinó con un lanzallamas montado en la parte delantera, aunque esto apenas funcionó en sus primeras peleas. Sin embargo, demostró arrojar llamas verdes en temporadas más recientes.

---

---

SawBlaze debutó exclusivamente como un robot de sierra en la temporada 2 de ABC, perdiendo su partido clasificatorio ante Razorback por decisión de los jueces después de que su hoja de sierra relativamente delgada se dobrara y atascara. A pesar de esto, SawBlaze regresó para una pelea a tres bandas, que ganó. Este principio básico seguiría siendo el mismo en los próximos años, pero evolucionó con el tiempo para volverse más eficiente y confiable.

En Discovery Season 3, SawBlaze lucía un disco de corte con bordes de diamante más resistente y una cabeza de dragón más grande, así como un color verde lima más vibrante en todo el bot. El brazo articulado se actualizó para girar más rápido y con más potencia, ya que se dobló como un auto enderezador cuando SawBlaze se encontró volcado. El equipo encontró mucho más éxito esta temporada, obteniendo un récord de 3-1 en las rondas de Fight Card, perdiendo solo un Evento Principal ante el ex semifinalista Bronco. Hizo lo suficiente para ganar un lugar en el Top 16, donde finalmente perdió ante Monsoon.

SawBlaze regresó para la temporada 2019, con la esperanza de respaldar una temporada exitosa la última vez. La cabeza de dragón se cambió por un método más funcional para evitar que la sierra golpee el suelo cuando se volteó. En las rondas de Fight Card, SawBlaze solo perdió un evento principal ante el ex campeón de BattleBots Tombstone y terminó 3-1 como resultado. SawBlaze dominó al ex campeón de BattleBots, Son of Whyachi, en el Top 16, solo para ser derrotado en los cuartos de final por el eventual subcampeón Witch Doctor.

Jamison Go y SawBlaze regresaron una vez más para Discovery Season 5, con mejoras de armas que le permitieron girar y reducir la velocidad en menos tiempo, lo que le dio al equipo un mayor control de su arma principal. SawBlaze terminó 2-1 en las rondas de Fight Card, incluida una victoria rotunda sobre el eventual subcampeón Whiplash. SawBlaze llegó al Top 16 por tercera temporada consecutiva, vengando su derrota contra Witch Doctor antes de perder ante Tantrum y caer en los cuartos de final en temporadas sucesivas.

SawBlaze luchó en la temporada 6 de BattleBots on Discovery. El mayor cambio visual parece estar relacionado con la horquilla central del robot, que tenía una tendencia a enroscarse hacia arriba y obstaculizar el juego terrestre de SawBlaze contra oponentes con armas giratorias. Esta horquilla ahora está dividida por el centro y es más robusta que en temporadas anteriores. El equipo también confirmó

---

---

una nueva caja de cambios que impulsa el brazo del arma, diseñada con la ayuda del miembro del equipo Jake Chesin. Esta actualización se realizó para hacer que el brazo del arma sea más resistente y más resistente a los grandes impactos, ya que es vital para la eficiencia del arma principal. La temporada de SawBlaze comenzó con una victoria dominante sobre el subcampeón de la temporada 2018, Minotaur, respaldada por una victoria en el Evento Principal por nocaut sobre MadCatter. Sin embargo, no pudo terminar con un récord perfecto después de perder ante los actuales campeones End Game y nuevamente terminar 2-1 después de Fight Night. SawBlaze continuó con su dominio en el grupo del torneo, derrotando a Hijinx, P1 y Riptide. A pesar de llegar a las semifinales por primera vez, SawBlaze finalmente fue derrotado por Witch Doctor, pero recogió el Premio del Fundador al final de la competencia.

Jamison Go se burló de las actualizaciones de SawBlaze en septiembre de 2022 antes del Campeonato Mundial VII, la más notable de las cuales se refería a las nuevas ruedas de Brasil, que son más pequeñas que las ruedas anteriores de SawBlaze pero considerablemente más pesadas. Fabricados con caucho vulcanizado, se han utilizado con gran eficacia en robots como Whiplash y Black Dragon en las últimas temporadas. El equipo también hizo una revisión de su sistema de transmisión, instalando motores de transmisión completamente nuevos, así como acopladores de entrada personalizados.[1] El equipo confirmó su inclusión en octubre de 2022, mostrando el envío de sus cajas antes de la filmación de la competencia.

## 6. Descripción de la empresa u organización y del puesto o área del trabajo del estudiante

El Instituto Tecnológico José Mario Molina Pasquel y Henríquez "Campus Zapopan" (también, el Tecnológico Mario Molina Zapopan o simplemente el Tec MM Campus Zapopan) es una institución pública de educación de nivel superior de Ingeniería, ubicada en Zapopan, Jalisco, México, y gestionado por el gobierno del estado. El Tec Mario Molina Campus Zapopan, fundada en el año 1999, el cual tiene la finalidad de prestar servicio de educación superior tecnológica en el Estado de Jalisco.

Desde 1999 la Universidad (Tec MM Campus Zapopan) opera a través de un modelo de red en el cual consiste en una administración general impartido por el Tecnológico Nacional de México (TecNM) y sectorizado a la Secretaría de Innovación Ciencia y Tecnología (Conacyt) que engloba un conjunto de establecimientos de educación superior pública de la República Mexicana. El Instituto Tecnológico José Mario Molina Pasquel y Henríquez opera como un

---

---

sistema en red de los 13 Institutos Tecnológicos Superiores actuales en el estado de Jalisco, los cuales cuentan con un Sistema de Educación Superior, un Sistema de Universidad Virtual, y la Administración General de la Institución.

La administración general tiene como función primordial impulsar las políticas coordinadas en lo académico, la gestión administrativa y la vinculación productiva de cada ITS para eficientar su operación y potenciar el impacto del sistema en los retos de la educación superior y desarrollo tecnológico en Jalisco.

Todos los integrantes de este proyecto son estudiantes que se encuentran en cursando el noveno semestre en la carrera Ingeniería mecatrónica en el campus Chapala.

## 7. Problemas a resolver

Se debe diseñar y construir un arma que sea capaz de destruir o dañar a otro robot de tal manera que no se dañe a él mismo. Tomando como consideración de alto rango la sincronización del control con el robot mediante radiofrecuencia para lograr con éxito el control total del robot, donde la programación influye en su mayoría, ya que, si el robot no es capaz de sincronizarse con el control por algún rango de error, el robot no debe de ser capaz de manipularse el solo, es decir no debe ser autónomo.

Una vez sincronizados, permitirá al capitán del equipo manipular el robot móvil para encenderlo o apagarlo, cambiar de dirección, activar y desactivar el arma.

El arma debe de ser capaz de no detenerse (una vez activada) al tocar otro objeto, pues lo mejor para el funcionamiento del robot será que el arma muestre alto torque al impactar con otro robot.

## 8. Objetivos

### **Objetivo general:**

Diseñar y crear un robot de batalla con un arma giratoria.

### **Objetivo específico:**

Planificar y diseñar PCB, código de programación, mecanismos, arma giratoria y carcasa lo suficientemente capaz de aplicar como un robot de combate a destrucción, el cual dependerá de un arma giratoria para su éxito en el campo de batalla.

---

## 9. Justificación

Se requiere cumplir con la construcción de un robot de pelea para la acreditación de la materia de robótica, con el objetivo de cumplir las competencias y objetivos requeridos para ser acreedores de la dicha. Así como estar en constante aprendizaje para poder desarrollar las habilidades en el área de robótica y poder aplicar los conocimientos adquiridos en el área laboral.

Ya que la robótica es propicia para apoyar las habilidades productivas, creativas y la programación, y se convierte en un motor para la innovación cuando produce cambios en las personas en las ideas y actitudes, en las relaciones y modos de actuar y pensar de los estudiantes.

El propósito educativo de esta experiencia es promover la creación de robots móviles para la materia de robótica, donde también se da importancia a las asignaturas de matemáticas, pues dentro de la programación o incluso cálculos para el funcionamiento del robot son necesarias el uso de las matemáticas. La robótica como área de conocimiento nueva requiere del respaldo financiero y académico a largo plazo, por esta razón las experiencias con estudiantes y educadores no deben ser cortas si no sostenidas y continuas.

# Marco Teórico

## 10. Marco Teórico

### Arduino UNO

Arduino UNO es una placa basada en el microcontrolador ATmega328P. Tiene 14 pines de entrada/salida digital (de los cuales 6 pueden ser usados con PWM), 6 entradas analógicas, un cristal de 16Mhz, conexión USB, conector jack de alimentación, terminales para conexión ICSP y un botón de reseteo. Tiene toda la electrónica necesaria para que el microcontrolador opere, simplemente hay que conectarlo a la energía por el puerto USB o con un transformador AC-DC. El pinout de este microcontrolador se muestra en la figura 10.1.

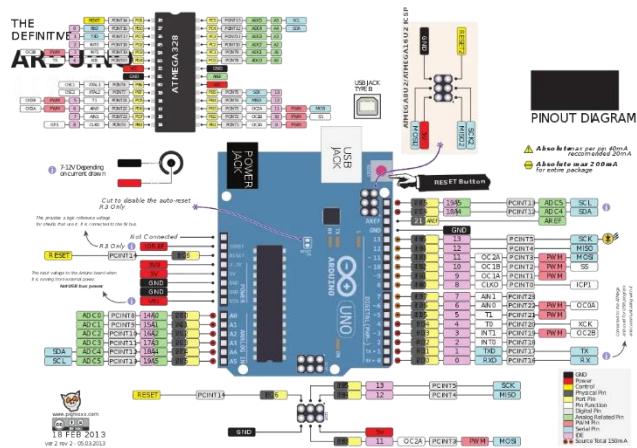


Figura 10.1 Diagrama de la placa ARDUINO y del ATMega328PU (Arduino.cl, 2013).

### Comunicación de Radiofrecuencia

El término se aplica para definir una parte del espectro electromagnético. Concretamente, la parte con menos energía de este. La transmisión de las ondas se produce al generar una corriente a través de un conductor, y se recibe con una antena. El ejemplo más claro es el de una estación de radio y un aparato receptor, como el de nuestro coche.

### Control de radiofrecuencia Attop

Modelo: DRON CYCLONE ADVENTURE FORCE RC TITANIUM AIR

Modelo: A8

Control: 2.4GHz

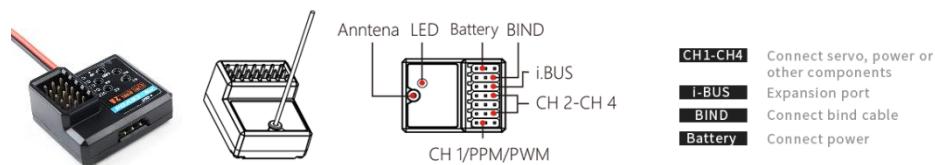
Este modelo de dron junto con su control de radiofrecuencia es mostrado en la figura 10.2.



*Figura 10.2 Dron Cyclone Adventure Force RC Titanium Air (NINCO AIR, 2016).*

## FGr4P

Control receptor de hasta 8 canales se muestra en la figura 10.3 y a continuación se muestran sus características técnicas para entender más sobre su función.



*Figura 10.3 FGr4P Receptor (FlySky).*

PWM channels: 4  
Wireless frequency: 2.4GHz  
Wireless protocol: AFHDS 3  
Range:> 200m (ground)  
Antenna type: single antenna (150mm)  
Power: 3.5-8.4V  
RSSI: Supported  
Data port: PWM / PPM / i.bus / s.bus  
Temperature range: -10 °C – + 60 °C  
Humidity range: 20% -95%  
Online Update: Yes  
Dimensions: 29 \* 22 \* 16.2mm  
Weight: 8g  
Certification: CE, FCC ID: N4ZFGFR4P00

## IDE Arduino

El IDE de Arduino puede implementarse en los sistemas operativos Windows, Mac y Linux. La mayoría de sus componentes están escritos en JavaScript para facilitar su edición y compilación. Aunque su intención principal es la de facilitar la escritura de códigos, tiene otras funciones que vale la pena destacar.

El código fuente para el IDE se publica bajo la [Licencia Pública General de GNU](#), versión 2. El IDE de Arduino admite los lenguajes C y C++ utilizando reglas especiales de estructuración de códigos.



Figura 10.4 IDE Arduino (Arduino, 2022).

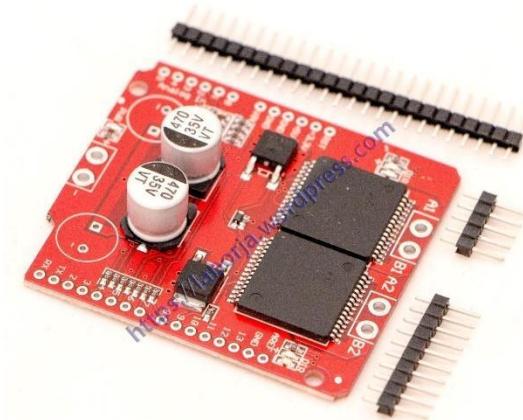
## Monster Moto Shield VNH2SP30

Esto es esencialmente una versión incrementada del motor Ardumoto. Por esta Sparkfun Monster Moto Shield ha reemplazado el L298 H-puente con un par de conductores de motor de puente completo VNH2SP30. También ha reforzado la circuitería de soporte por lo que esta placa es capaz de conducir un par de motores de alta corriente. El VIN. A continuación, se muestran sus características y en la figura 10.5 se muestra el módulo.

### Características:

- Voltaje máximo: 16V
- Valor máximo de corriente: 30 A
- Práctica corriente continua: 14 A
- La detección de corriente disponible para Arduino pin analógico
- MOSFET de resistencia: 19 mΩ (por trayecto)
- frecuencia PWM máxima: 20 kHz

- apagado térmico
- Baja tensión y Desconexión por sobretensión



*Figura 10.5 Monster Moto Shield (wordpress, 2022).*

### **Motorreductores 12Kg 300 rpm**

Motor reductor de 12V 300 rpm con una fuerza nominal de 1,3 Kg·cm y 4 kg·cm a máxima eficiencia mostrado en la figura 10.6. Este motor de alta velocidad destaca por ser silencioso, su pequeño tamaño y reducido consumo. Resulta idóneo para aplicaciones en las que se necesita una alta velocidad de giro. Tensión: 12V. Consumo: 800 mA. Fuerza: 0,13 N·m. Diámetro del eje: 6mm

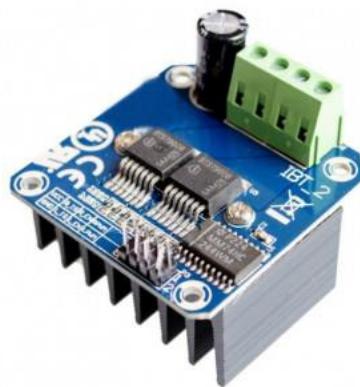
Tensión Nominal	12V
Velocidad Sin carga	300 rpm
Consumo sin carga	140 mA
Consumo nominal	< 800 mA
Velocidad nominal	220 rpm
Fuerza nominal	1,3 kgf·cm / 0,13 N·m
Fuerza a máxima eficiencia	4 kgf·cm / 0,39 N·m
Consumo en frenado	3000 mA
Potencia de salida	2,9 W
Diámetro Máximo	37 mm
Diámetro Eje	6 mm
Rosca fijación	4 x M3



*Figura 10.6 Motor Reductor de 12V 300 rpm (2022).*

### BTS7960 Driver motor 43A de alta potencia

En este proyecto se utilizó el driver BTS7960 que se muestra en la figura 10.7 para controlar los motores debido a que este driver es de alta potencia y resiste hasta 43 Amperios.



*Figura 10.7 Driver BTS7960 (2022).*

### Características:

- Alimentación: 5.5 a 27V
- Corriente en reposo: 7 uA a 25 ° C.
- Capacidad de PWM de hasta 25 kHz combinada con activación libre.
- Protección de sobrecorriente
- Bloqueo de sobretensión
- Intensidad máxima: 43 Amperios
- Circuito digital de control con entradas de nivel lógico
- Velocidad de giro ajustable por PWM (hasta 25 KHz)

---

### Servomotor Tower Pro MG996R

Este servo es esencialmente una versión mejorada del famoso servo MG995. Ahora tiene un sistema de control de PCB e IC rediseñado que lo hace mucho más preciso. Su engranaje y motor internos también se actualizan para mejorar el centrado.

Especificaciones:

- Voltaje de operación: 4.8 V a 7.2 V
- Velocidad de operación: 0.17 s/60° (4.8 V), 0.14 s/60° (6 V)
- Torque detenido: 9.4 kgf·cm (4.8 V ), 11 kgf·cm (6 V)
- Con rodillo doble
- Ángulo de rotación: 120° aprox.
- Banda muerta: 5  $\mu$ s
- Peso: 55 g
- Dimensiones: Largo 40.7 mm, ancho 19.7 mm, altura 42.9 mm aprox.
- Largo del cable: 31 cm aprox.



Figura 10.8 Servomotor MG996R (2022).

### MEGA RUEDA ROBOT TODO TERRENO 120 MM S360168

Este par de mega ruedas de 120 mm proporcionan la locomoción idónea para robots que se aventuran en el exterior mostradas en la figura 10.7. La rueda está formada por una llanta de pvc y una cubierta de caucho ideal para todo terreno gracias a sus grandes estrías, que están llenas de goma espuma. La rueda proporciona una gran tracción en terrenos accidentados. La rueda puede fijarse sobre un eje de 6 mm con la ayuda de los casquillos de fijación hexagonales S360187.



*Figura 10.9 Mega Rueda Robot Todo Terreno (2022).*

### **MOTOR (MAQUINA ELECTRICA)**

Motores reconstruidos de los cuales se desconoce su origen, sin embargo se tiene una estimación que pertenecían a una máquina eléctrica, son de alto consumo de amperaje +10A, bajo torque, altas revoluciones como se muestra en la figura 10.10



*Figura 10.10 Motores de alto amperaje (Karen Samantha, 2022).*

### **MOTORREDUCTOR**

Para el motor del arma se seleccionó una caja de engranajes Kelvin K80 en conjunto con un motor ETRM 48V 1.05A 3000 RPM de 0.03Kw, se desconoce el origen del motor, sin embargo fue la mejor opción para el arma debido a su alto torque. Este motor se muestra en la figura 10.11 mientras que en la figura 10.12 se muestran sus dimensiones.



Figura 10.11 Motor ETRM 48V (Satomi Minami, 2022).

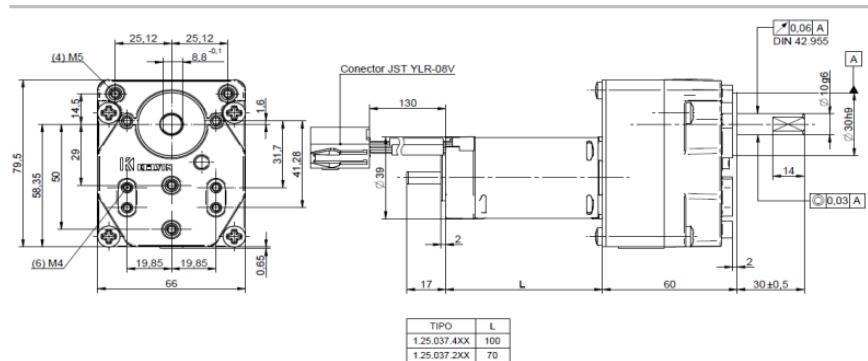


Figura 10.12 Motor ETRM 48V Dimensiones (ETRM).

### BATERIA (ETP)

Es una batería para máquinas eléctricas de 20V/7V con una capacidad de duración de 12000 mAh como se muestra en la figura 10.13.



Figura 10.13 Batería 20V a 1200 mA (Satomi Minami, 2022).

---

## BATERIA 12 CELDAS 24V

Contamos con 12 celdas 18650 de 2200 mAh cada una, están divididas en conjunto de 6 baterías donde nos entregan 12V, las uniremos para conseguir 24 Volts necesarios para nuestro carro, consiguiente un voltaje nominal de 13.6A, estas batería se muestran en la figura 10.14.



*Figura 10.14 Celdas 18650 (Satomi Minami, 2022).*

## MOTORES BRUCK PARABRISAS

Motor de múltiples velocidades para limpia parabrisas con una alimentación máxima de 24V amperaje nominal 2.2A y un torque no especificado por el fabricante, usualmente empleados en proyectos caseros como asadores. La marca BRUCK GERMANY es una de las marcas de repuestos automotrices más reconocidas en México. Su catálogo abarca miles de aplicaciones; y todos sus productos son fabricados bajo las normas de calidad más estrictas de la Unión Europea. Este motor se muestra en la figura 10.15.



*Figura 10.15 Motores Bruck Germany de Parabrisas (Jennifer González, 2022).*

### Características:

Modelo: GOLF JETTA A4

Número de parte: 1J0-955-119

## FUSION 360.

Fusion 360 es una plataforma de software CAD, CAM, CAE y de circuitos impresos de modelado 3D basada en la nube para el diseño y la manufactura de productos.

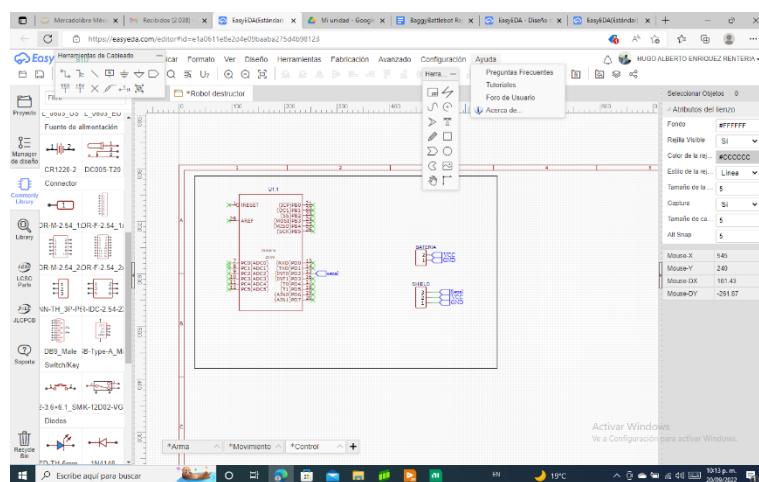
Diseña y proyecta productos del modo que deseas para garantizar su ajuste, estética, forma y función. Proyecta, diseña y crea lo que quieras con los amplios componentes electrónicos y las herramientas de diseño de circuitos impresos. Se muestra el interfaz demostrativo en 3D en la figura 10.16.



Figura 10.16 Fusion 360 (2022).

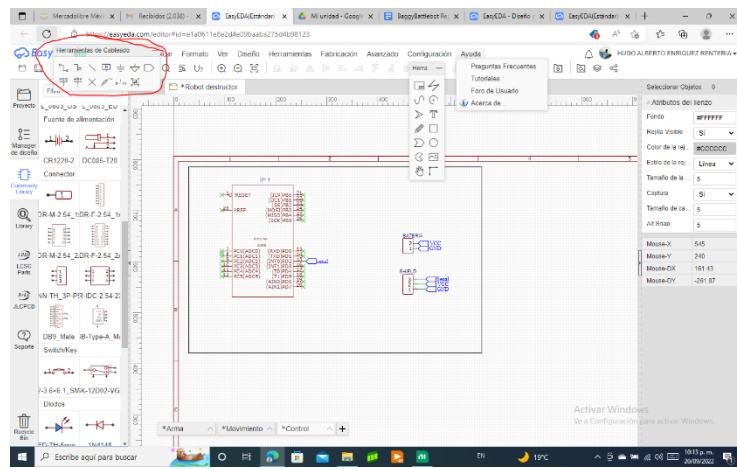
## Easy EDA

EasyEDA es un software de diseño de PCB fácil de usar (mostrado en la figura 10.17). Sin embargo, puede encontrar un problema o dos, especialmente si es la primera vez que usa la herramienta. Intentamos identificar algunos problemas relacionados con el esquema de PCB y ofrecer soluciones. EasyEDA es una herramienta de diseño de PCB de última generación diseñada para adaptarse tanto a principiantes como a expertos.



*Figura 10.17 EasyEDA (Hugo, Henríquez, 2022).*

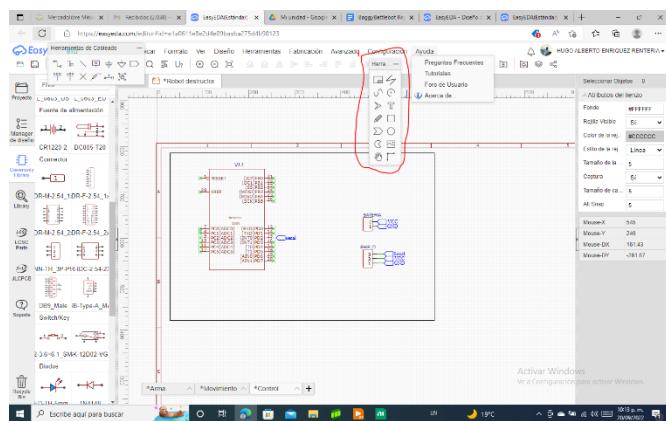
En la figura 10.18 se muestra el panel principal del software que utilizaremos donde vienen las herramientas para agregar y conectar los circuitos como se ha planeado para el funcionamiento del proyecto.



*Figura 10.18 EasyEDA Panel principal (Hugo Henríquez, 2022).*

En el círculo rojo marcado en la figura 10.19 nos muestra una de las herramientas de donde podemos apreciar que nos da varias opciones, en las cuales son:

- Hacer conexiones
- Terminal de voltaje fijo a 5v
- Terminal de VCC(voltaje corriente continua)
- Puertos de Red (para solo tener etiquetas sin tener que conectar todo)
- Tache para los pines o conexiones que no se utilizaran.
- Para hacer el agrupamiento de varias conexiones
- entrada de un pin
- Terminal a tierra.



*Figura 10.19 EasyEDA Herramienta Panel principal (Hugo Henríquez, 2022).*

En otra herramienta nos las opciones diferentes que sirven para:

- Agrupar esquemas cuando está muy grande el circuito
- Dibujar libre
- ponerles texto
- polígono
- arrastrar (algún diagrama)
- cabeza de flecha
- volver al origen (vista principal o inicial)
- elipse
- Tarta

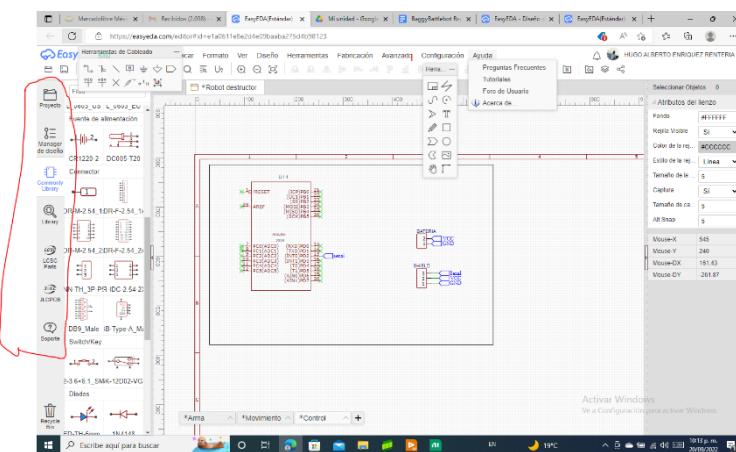
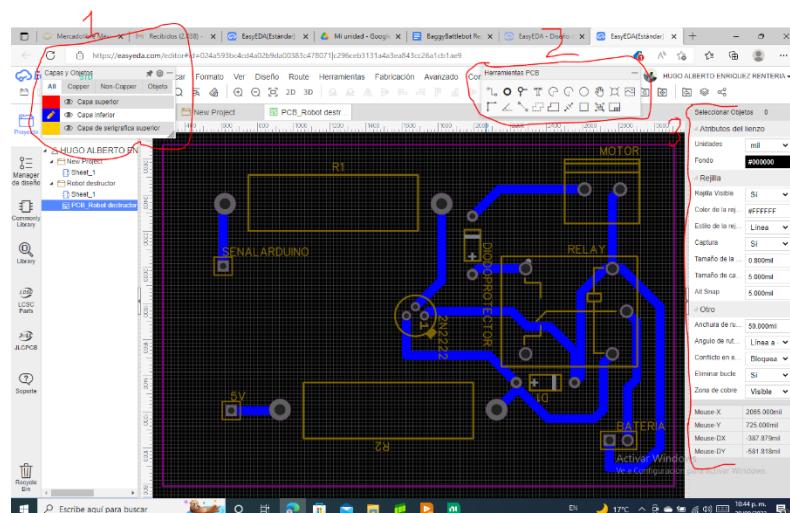


Figura 10.20 EasyEDA Panel de herramientas (Hugo Henríquez, 2022).

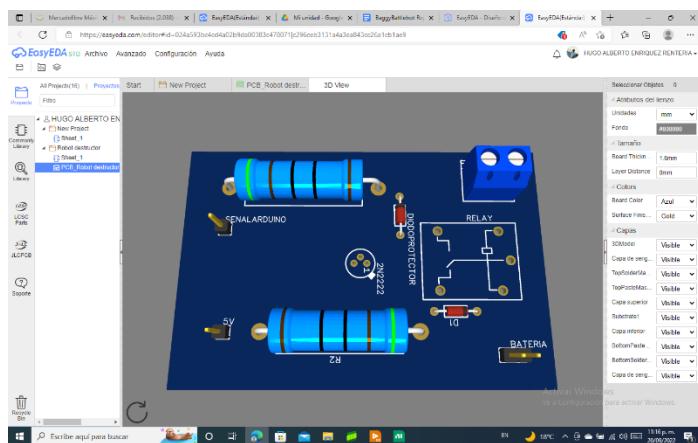
De este lado, en la figura 10.20 nos encontramos con las herramientas para la búsqueda de componentes y donde mandar a fabricar las PCB 'S que tengamos, así como crear proyectos nuevos, guardarlos o abrirlos.



*Figura 10.21 EasyEDA Crear una PCB (Hugo Henríquez, 2022).*

En la siguiente imagen figura 10.21 se muestra como es el siguiente paso al concluir con tus diagramas, en esta parte ya se diseña una PCB, tanto como las pistas, el acomodo de los componentes, y las dimensiones de la PCB. Marcamos las herramientas de esta parte con números

1. Tenemos la herramienta para ver por qué parte de la baquela quitaremos o haremos las conexiones para las conexiones y al momento de mandar a maquinaria nos quedará el resultado como lo hemos hecho aquí.
2. Tiene herramientas para hacerle unas cosas y personalizar tus PCB, por ejemplo, si tienes que hacer un puente por la parte de superficial tienes la opción de generar un hoyo, o si quieres ponerle el nombre a tu PCB tienes la herramienta de texto entre otras.
3. Aquí podemos modificar cualquier componente que tengamos seleccionado, o alguna pista, por ejemplo, en las pistas que es lo que más se usa, podemos modificar la anchura de la pista, en los hoyos su anchura y su queremos que tenga más shape o su forma del orificio.



*Figura 10.22 EasyEDA Vista previa de la PCB diseñada (Hugo Henríquez, 2022).*

En la figura 10.22 podemos ver como sería la vista previa de lo que será el circuito ya montado en la baquela, a veces si saldrá con todos los componentes, pero en algunos casos solo saldrá esto (un dibujo) donde se verá dónde estaría ese componente. al lado derecho tendremos lo que son las especificaciones y algunas herramientas para cambiar algunas cosas básicas como el color de la PCB, etc.

## Driver TB6600.

Permite controlar motores de hasta 3.5A por canal (4A max). Este driver tiene limitación de corriente ajustable, protección contra sobre corriente y 7 resoluciones diferentes de microstepping.

---

---

El TB6600 es un controlador profesional para motores paso a paso bipolares. Es compatible con microcontroladores como Arduino y otros que puedan generar señales de pulsos de 5V. El TB6600 soporta una gran variedad de voltajes de entrada de **9 a 42V DC**. Es capaz de proporcionar hasta 3.5A de corriente de forma continuada y 4A de pico por cortos periodos de tiempo. De esta forma puede controlar una gran variedad de motores.

El controlador soporta el control de dirección y de paso como en todos los controladores de éste tipo. También puede configurarse para microstepping mediante unos micro interruptores incluidos. Hay 7 valores posibles: 1, 2 / A, 2 / B, 4, 8, 16 y 32 además de 8 posiciones para el ajuste de corriente: 0.5A, 1A, 1.5A, 2A, 2.5A, 2.8A, 3.0A y 3.5A. Todas las señales están protegidas internamente mediante optoacopladores de alta velocidad para evitar interferencias y mejorar el aislamiento del circuito de control. Viene ensamblado en una caja de metal para medir refrigeración.

## Desarrollo

### 11. Procedimientos y descripción de las actividades realizadas

#### 1. Bocetos

Después de una larga lluvia de ideas entre todos los miembros del equipo, se llegó a un diseño y se hizo del bosquejo en base a los deseos e ideas de todos. Este se muestra a continuación en la figura 11.1.1.

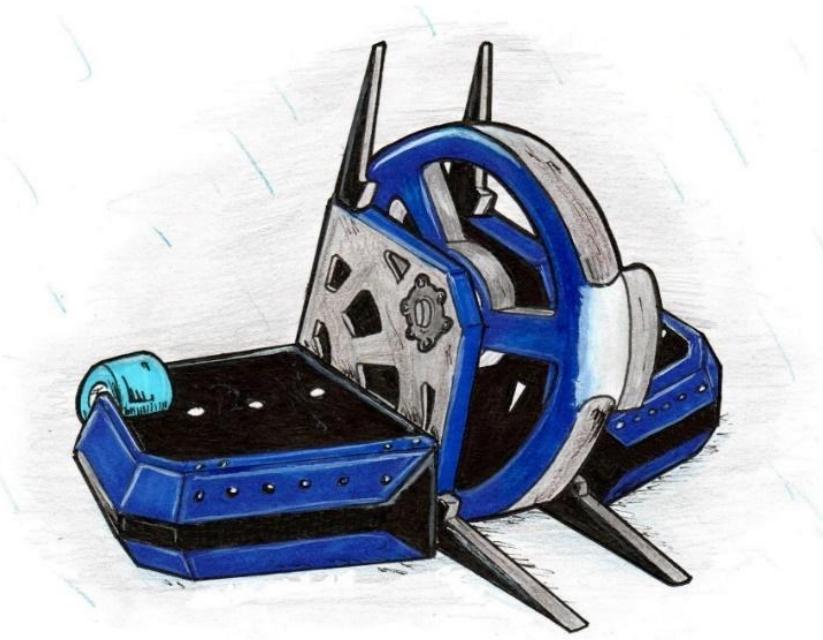


Figura 11.1.1. Boceto del robot ‘Baggy’ de combate (Jennifer González, 2022).

Después se realizó el bosquejo del arma, una clase de rodillo. El diseño que se busca implementar es como una clase de hacha o cuchillo delgado pero lo suficientemente filoso para hacer daño al primer contacto. La primera idea del arma se muestra en la figura 11.1.2.

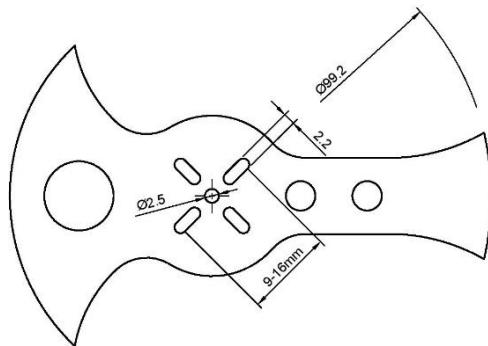


Figura 11.1.2. Boceto del arma del robot (Ramirez Rosales María del Carmen, 2022).

El arma fue diseñada en fusión 360. Para el boceto se dibujó un círculo con diámetro de 300 mm. Dentro del círculo en el punto central se dibujó otro círculo más pequeño con diámetro de 60 mm aprox. Se unió una línea curvada de un extremo al otro extremo del círculo. El círculo pequeño sólo se utilizó como base ya que después fue eliminado como se muestra en la figura 11.1.3.

Mientras que en la figura 11.1.4 se muestra cuando se terminó el boceto se extruye solo la parte del arma, por lo tanto, se borró lo restante del círculo. La extrusión fue de 30 mm de grosor aproximadamente.

Por último, en esta sección tenemos la figura 11.1.5 el cual muestra un plano con los bocetos del arma.

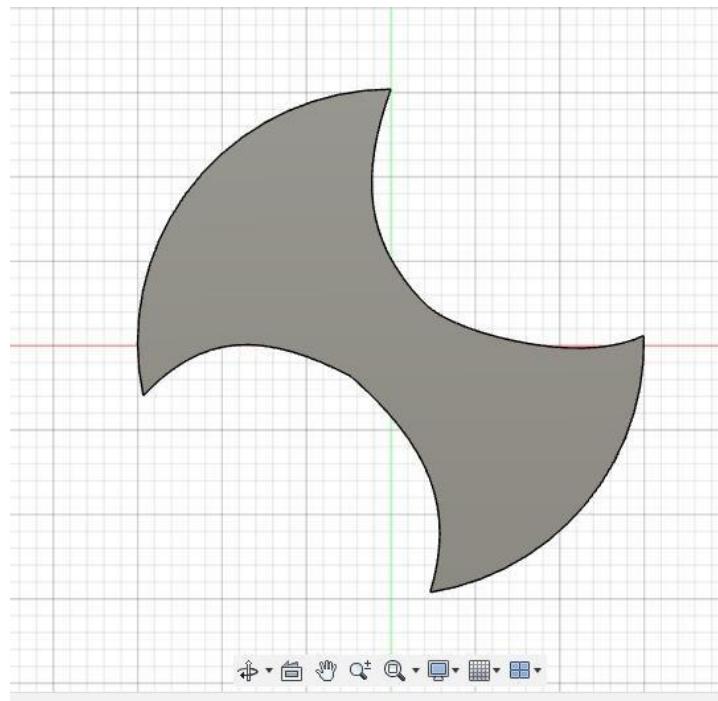


Figura 11.1.3. Boceto del arma propuesta 2 Fusion (Ramirez Rosales María del Carmen, 2022).

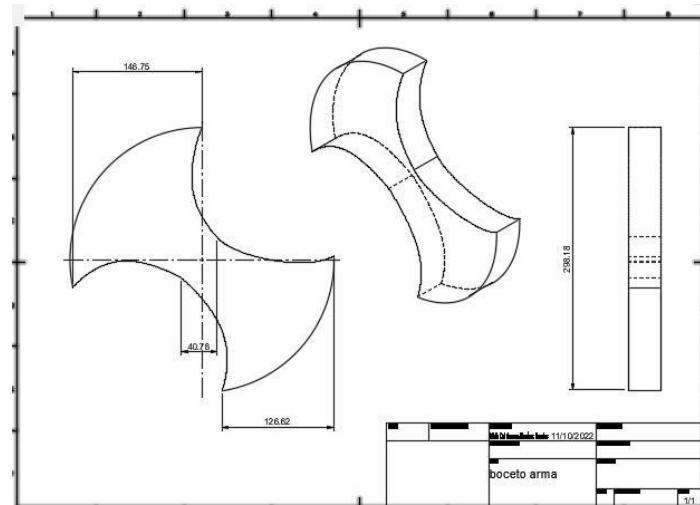


Figura 11.1.4. Plano del arma propuesta 2 (Ramirez Rosales María del Carmen, 2022).

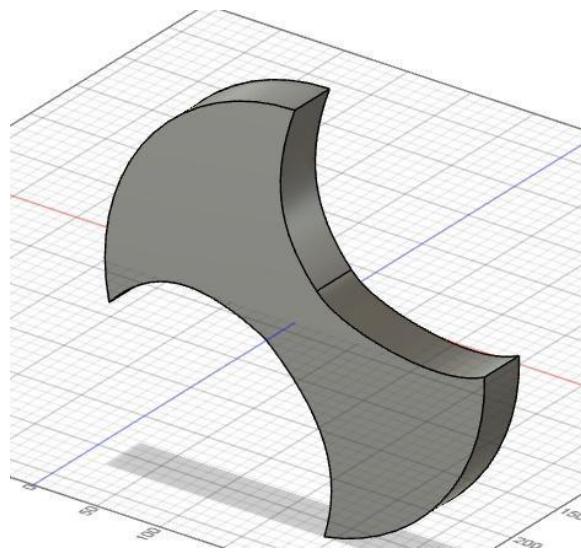


Figura 11.1.5. Boceto del arma del robot (Ramirez Rosales María del Carmen, 2022).

## Arma 2.0

Segunda versión del arma donde hemos pulido a mejor detalle las dimensiones y hemos considerado de manera más óptima los materiales que emplearemos



Figura 11.1.6 Arma 2.0 (Satomi Minami, 2022).

Las dimensiones del arma se mantienen en un largo de 30 cm mientras que el espesor ha sido modificado a  $\frac{1}{2}$  in o 1.27 cm, además se han añadido diversos círculos a lo largo del amar con el fin de obtener menos peso sin perder rigidez

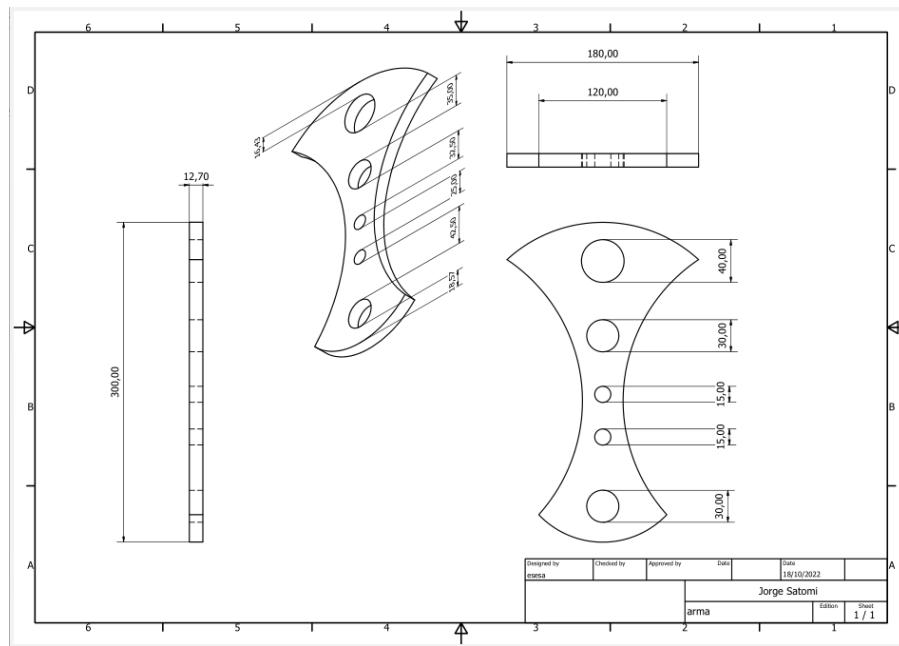


Figura 11.1.7 Boceto arma 2.0 (Satomi Minami, 2022).

## Arma 3.0

Sin embargo, debido a las pruebas realizadas con los motores, entre otras cuestiones como el material para crear el arma se cambio completamente el arma pensando en hacer un rodillo con estrellas filosas (se muestra en la sección de carcasa más adelante). Y el resto del diseño se cambio debido a los problemas que presentaríamos con la movilidad del primer diseño planteado, este diseño se muestra en el desarrollo y en los anexos junto con todos los planos utilizados.

### 2. Dimensiones.

Las dimensiones que tenemos contempladas para el robot son de 55cm de largo por 15 cm de altura por 25 cm de ancho.

Puede que cambiemos medidas, todo depende del tamaño de los componentes ya que las medidas se tomaron a consideración del tamaño de los componentes que estamos probando.

Por el momento la carcasa se mantiene como se muestra en la figura 11.2.1. En la parte de enmedio se planea poner el arma con sus respectivas bases y en los 2 huecos extremos corresponden a las llantas.

Pueden realizarse cambios sin previo aviso.

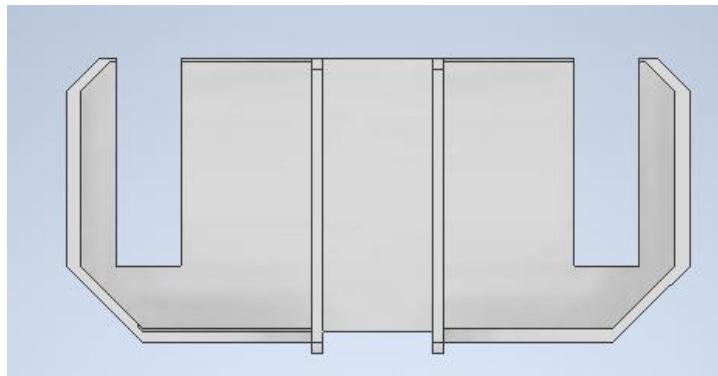


Figura 11.2.1 Base del robot boceto (Maricarmen Ramirez, Autodesk Inventor, 2022).

La figura 11.2.2 muestra el chasis del robot, se pueden apreciar los soportes del arma y su cuerpo externamente, pues así quedaría. (puede modificarse sin previo aviso

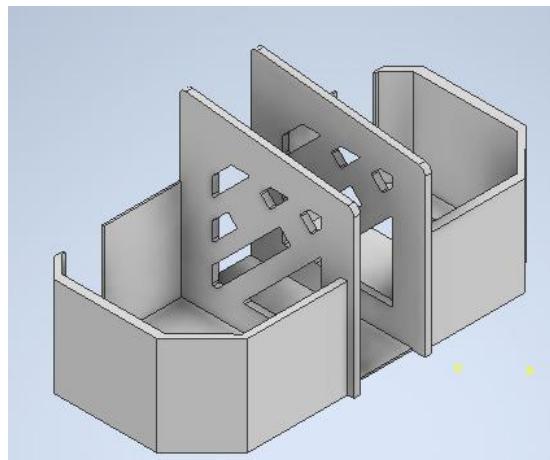
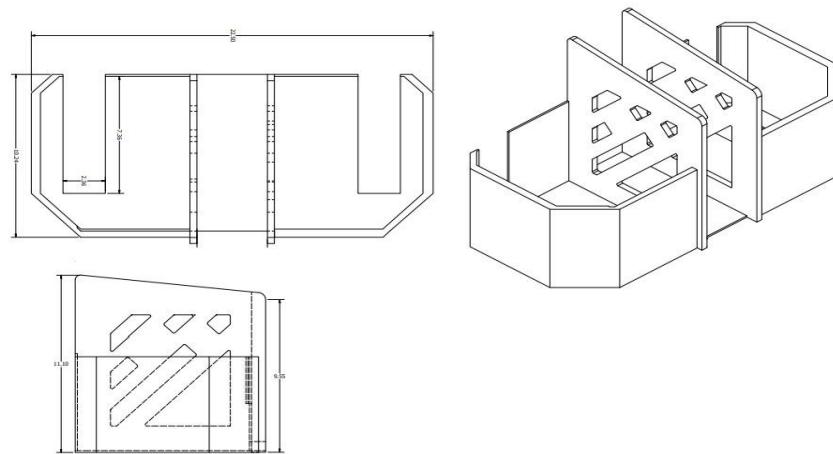


Figura 11.2.2 Chasis.

La figura 11.2.3 Muestra el boceto de la carcasa con sus respectivas dimensiones.

quedando como 22.50 in de largo x 10.24 in de ancho. Con un espacio para las llantas de 2.36 in de ancho x 7.36 in de largo y con un espacio para el arma de 3.93 in de ancho con 10.24 in de largo.

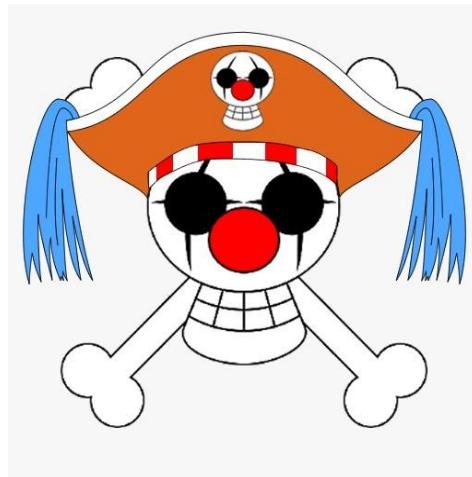
Los soportes del arma la parte trasera es más larga, queda con 11.10 in y por delante queda con 9.55 in de alto y de largo sus respectivos 10.24 in.



*Figura 11.2.3 Dimensiones (AutoDesk Inventor, 2022).*

# BAGGY

*Figura 11.2.3 Molde de letras Buggy para el robot (Photoshop, Jennifer González, 2022).*



*Figura 11.2.3 Buggy logo (One Piece, Eiichiro Oda 1997).*

### 3. Pruebas de radiofrecuencia

Para el primer componente, el FGR4, se utilizaron las siguientes conexiones mostradas en la figura 11.3.1.

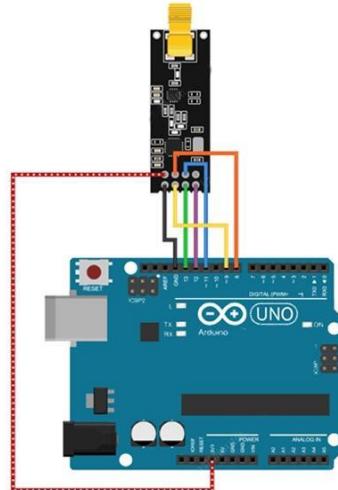


Figura 11.3.1 Conexiones del módulo NRF24L01 de prueba (Fritzing, 2022).

Pero el resultado de esta prueba fue que hay ocasiones que el módulo de radiofrecuencia se desconecta sin motivo aparente, y podría causar fallos cuando el robot esté siendo controlado. Por lo que utilizar esta opción se volvió descartable.

En la siguiente prueba se utilizó un control de radiofrecuencia Attop del dron Adventure Force RC Titanium A8. Para poder controlarlo se tuvo que buscar una manera de hackear la placa electrónica del dron, ya que está no tenía un receptor, sino que está en un circuito integrado y para ello se tuvo que analizar la placa del circuito del dron como se observa en la figura 11.3.2.

Mientras que en la figura 11.3.3 se pueden observar algunos circuitos integrados en la placa de circuito del dron. La tarjeta contaba con 4 pequeños controladores de los 4 motores del dron, el receptor y un amplificador de señal.

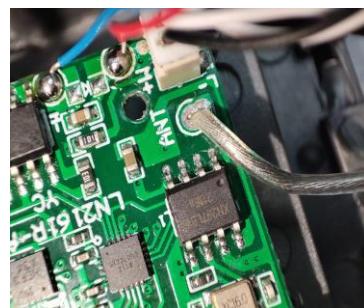


Figura 11.3.2 Transceiver chip XN297LB de tarjeta electrónica de dron (Jennifer González, 2022).

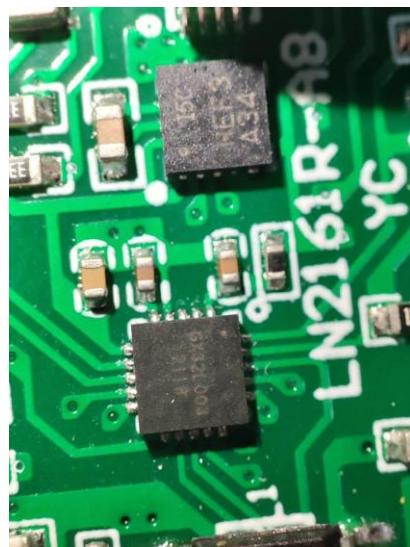


Figura 11.3.3 Circuito integrado GX32L003 del dron (Jennifer González, 2022).

Se busco el circuito integrado que nos entregará la señal de salida del radiofrecuencia (XN297) para poder utilizarla, sin embargo no pude encontrar el circuito integrado llamado “GX32L003” el cual parece ser que amplifica la señal pero se desconoce su datasheet y su funcionamiento ya que se observó que las salidas están conectadas a él, así que la única opción era checar las señales por un osciloscopio y buscar la manera de poder decodificar la señal para obtener una más comprensible para nuestro robot, sin embargo esto implicaría no solo poner en riesgo el robot a causa del proceso no se complete con éxito, sino que también implicaría echar a perder por completo la tarjeta electrónica del dron, por lo cual el dron quedaría inservible. Es por eso que se optó por comprar un nuevo control de rf con su propio receptor.

El control que nuevo que se adquirió fue un FlySky FS-i6 DE 2.4GHz el cual cuenta con 6 canales junto con su módulo receptor. Por lo tanto, se hizo el bosquejo de conexión para probar el control con arduino el cual se muestra en la figura 11.3.4.

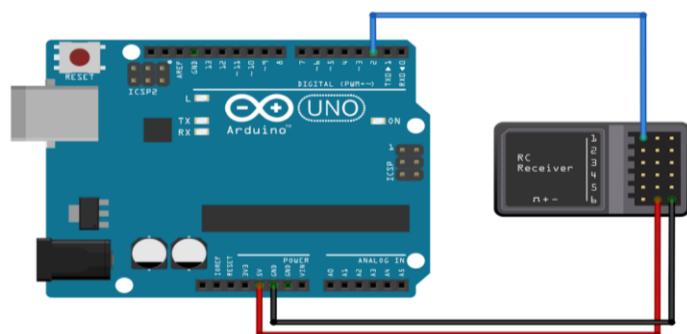
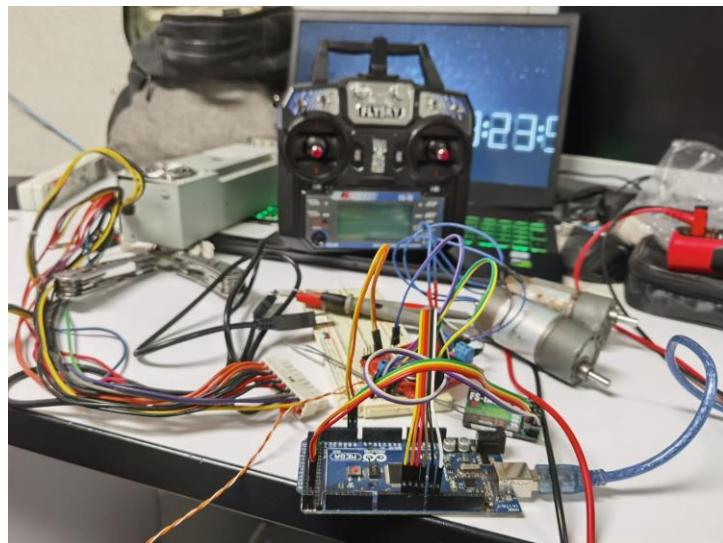


Figura 11.3.4 Conexiones del FGR4 de prueba (Fritzing, 2022).

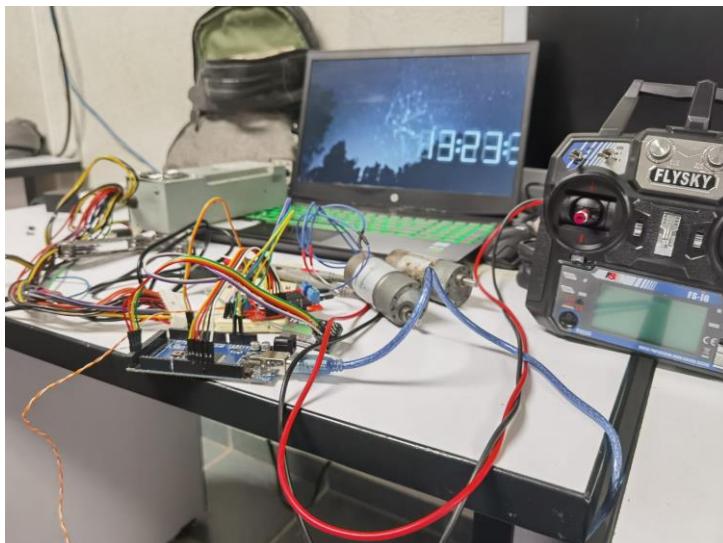
---

---

Mientras que en las figuras 11.3.5 y 11.3.6 se muestran las pruebas del control FlySky junto con los motores y los servomotores, este procedimiento se realizó una vez que se identificaron todos los canales del control y sus respectivos valores.



*Figura 11.3.5 Prueba de control de radiofrecuencia (Jennifer González, 2022).*



*Figura 11.3.6 Prueba de control de radiofrecuencia con motores (Jennifer González, 2022).*

#### 4. Programación

Se comenzó con la estructura del código del robot en general, enfocándose primeramente en la movilidad del robot, es decir para controlar las direcciones de

este, arriba, abajo, izquierda, derecha, arriba derecha, arriba izquierda, abajo derecha y abajo izquierda. Una vez obtenida las direcciones se asignó un canal a la dirección de los motores, pero ya con el driver Monter moto shield, y el código realizado en lo mencionado anteriormente, fue el siguiente.

```
#include <Servo.h>

// FS-IA6 Receiver Test

#define N 6      // Define channels

#define ch1 22   // Motors Left & Right

#define ch2 24   // Down & Up

#define ch3 26   // Servo Down & Up

#define ch4 28   // Weapon speed

#define ch5 30   // Active Weapon

#define ch6 32   // Turn On/Off Robot Battle

// 8, 7, 4, 9, 6, 5 // Digital Pin for Monster Moto Shield

// MONSTER MOTO SHIELD

int D[2] = { 9, 8 };          // seleccion de la direccion

int M[2] = { 4, 7 };          // seleccion del motor

int pwmpin[2] = { 5, 6 };    // 5 motor1 y 6 motor2

int en[2] = { A0, A1 };

int ch[N]; // Array

//Servomotors

Servo servoRight;
```

```
Servo servoLeft;

int SERVO = 0;      //Store grade data

float grados = 0;  //Store

int lectura = 0;   //Store analog value


void setup() {

    Serial.begin(115200);

    pinMode(ch1, INPUT);

    pinMode(ch2, INPUT);

    pinMode(ch3, INPUT);

    servoRight.attach(2);

    servoLeft.attach(3);

    for (int i = 0; i < 2; i++) {

        pinMode(M[i], 1);

        pinMode(D[i], 1);

        pinMode(pwmpin[i], 1);

    }

}

void loop() {

    ch[0] = pulseIn(ch1, HIGH); // Motors Left & Right

    ch[1] = pulseIn(ch2, HIGH); // Down & Up

    ch[2] = pulseIn(ch3, HIGH); // Servo Down & Up

    //ch[3] = pulseIn(ch4, HIGH); // Weapon speed

    // ch[4] = pulseIn(ch5, HIGH); // Active Weapon
```

```
//  ch[5] = pulseIn(ch6, HIGH); // Turn On/Off Robot Battle

robotMove();

/*
Serial.print("Channel 1 = :");
Serial.print(ch[0]);
Serial.print(", ");
Serial.print("Channel 2 = :");
Serial.print(ch[1]);
Serial.print(", ");
Serial.print("Channel 3 = :");
Serial.print(ch[2]);
Serial.print(", ");
Serial.print("Channel 4 = :");
Serial.print(ch[3]);
Serial.print(", ");
Serial.print(" -----> ")*/
}

void robotMove() {
//Send servos in their position
//Channel 2 UP AND DOWN
if (ch[1] > 1390 && ch[1] < 2000) { // UP
    Serial.println("ARRIBA");
}
```

```
    digitalWrite(M[0], 1);          //4
    digitalWrite(M[1], 1);          //7
    digitalWrite(D[0], 0);          //9
    digitalWrite(D[1], 0);          //8
    analogWrite(pwmpin[0], 1023);  //5
    analogWrite(pwmpin[1], 1023);  //6
}

if (ch[1] > 980 && ch[1] < 1300) { // DOWN
    Serial.println("ABAJO");
    digitalWrite(M[0], 0);          //4
    digitalWrite(M[1], 0);          //7
    digitalWrite(D[0], 1);          //9
    digitalWrite(D[1], 1);          //8
    analogWrite(pwmpin[0], 1023);  //5
    analogWrite(pwmpin[1], 1023);  //6
}

if (ch[1] > 1380 && ch[1] < 1400 || ch[0] > 1370 && ch[0] < 1400) { // STOP
    Serial.println("STOP");
    digitalWrite(M[0], 0);          //4
    digitalWrite(M[1], 0);          //7
    digitalWrite(D[0], 0);          //9
    digitalWrite(D[1], 0);          //8
    analogWrite(pwmpin[0], 1023);  //5
    analogWrite(pwmpin[1], 1023);  //6
}

if (ch[0] > 990 && ch[0] < 1300) { // LEFT
```

```

Serial.println("IZQUIERDA");

digitalWrite(M[0], 0);           //4
digitalWrite(M[1], 1);           //7
digitalWrite(D[0], 1);           //9
digitalWrite(D[1], 0);           //8
analogWrite(pwmpin[0], 1023);   //
analogWrite(pwmpin[1], 1023);   //6

}

if (ch[0] > 1381 && ch[0] < 2000) { // RIGHT
    Serial.println("DERECHA");

    digitalWrite(M[0], 1);           //4
    digitalWrite(M[1], 0);           //7
    digitalWrite(D[0], 0);           //9
    digitalWrite(D[1], 1);           //8
    analogWrite(pwmpin[0], 1023);   //5
    analogWrite(pwmpin[1], 1023);   //6

}
}

void arma() {
    int val= analogRead(ch4);
    val=map(val,0,1023,0,180); // Conversión de valores
}

void OnOff() { // channel 6
    if (ch[5] = pulseIn(ch6, HIGH)); {

```

```
analogWrite(pwmpin[0], 1023);

analogWrite(pwmPin[1], 1023;)

return HIGH;

}

if (ch[5] = pulseIn(ch6, LOW);) {

analogWrite(pwmpin[0], 0);

analogWrite(pwmPin[1], 0);

return LOW;

}

}
```

Y a continuación se muestra la prueba de los servos con el 3er canal del control de radiofrecuencia:

```
#include <Servo.h>

#define N 4

int ch[N]; // Array

Servo servoRight; // create servo object to control a servo

Servo servoLeft;

#define ch3 26 // Servo Down & Up

float lectura = 0;

int SERVO = 0;

//function

int servoRange(int reciveValue = 0);

void setup() {
```

```
Serial.begin(115200);

servoRight.attach(2); // attaches the servo on pin 9 to the servo object
servoLeft.attach(3);

}

void loop() {

ch[2] = pulseIn(ch3, HIGH); // Servo Down & Up
Serial.println(ch[2]);
servoRange(ch[2]);
}

int servoRange(int receiveValue = 0) {

//-----Mindle to Down range-----

if (receiveValue >= 1400 && receiveValue <= 1600) { // midle = 90°
servoLeft.write(90);
servoRight.write(90);
Serial.println("ESTOY EN EL PRIMER RANGO");
delay(10);
}

if (receiveValue >= 1300 && receiveValue <= 1395) {
servoLeft.write(72);
servoRight.write(108);
Serial.println("ESTOY EN EL SEGUNDO RANGO");
delay(10);
}
```

```
if (receiveValue >= 1200 && receiveValue <= 1295) {  
    servoLeft.write(54);  
    servoRight.write(126);  
    Serial.println("ESTOY EN EL TERCER RANGO");  
    delay(10);  
}  
  
if (receiveValue >= 1100 && receiveValue <= 1195) {  
    servoLeft.write(36);  
    servoRight.write(144);  
    Serial.println("ESTOY EN EL CUARTO RANGO");  
    delay(10);  
}  
  
if (receiveValue >= 1000 && receiveValue <= 1095) {  
    servoLeft.write(18);  
    servoRight.write(162);  
    Serial.println("ESTOY EN EL QUINTO RANGO");  
    delay(10);  
}  
  
if (receiveValue >= 900 && receiveValue <= 995) {  
    servoLeft.write(0);  
    servoRight.write(180);  
    Serial.println("ESTOY EN EL SEXTO RANGO");  
    delay(10);  
}
```

```
//-----Mindle to up range-----  
  
if (receiveValue >= 1610 && receiveValue <= 1695) { // midle = 90°  
    servoRight.write(90);  
    servoLeft.write(90);  
    Serial.println("ESTOY EN EL PRIMER RANGO");  
    delay(10);  
}  
  
if (receiveValue >= 1700 && receiveValue <= 1795) { // midle = 90°  
    servoRight.write(72);  
    servoLeft.write(108);  
    Serial.println("ESTOY EN EL PRIMER RANGO");  
    delay(10);  
}  
  
if (receiveValue >= 1800 && receiveValue <= 1895) {  
    servoRight.write(54);  
    servoLeft.write(126);  
    Serial.println("ESTOY EN EL SEGUNDO RANGO");  
    delay(10);  
}  
  
if (receiveValue >= 1900 && receiveValue <= 1995) {  
    servoRight.write(54);  
    servoLeft.write(144);  
    Serial.println("ESTOY EN EL TERCER RANGO");  
    delay(10);  
}
```

```
}
```

Código para probar la activación del pin encargado de activar/desactivar el arma

```
void setup() {  
    pinMode(7, OUTPUT);  
    Serial.begin(9600);  
}  
  
void loop() {  
    int lec = analogRead(A2);  
    Serial.println(lec);  
    if (lec > 700) {  
        digitalWrite(7, HIGH);  
        Serial.println(lec);  
    } else {  
        digitalWrite(7, LOW);  
    }  
}
```

Código para probar los motores

```
//Motor Right  
#define LPWM 9  
#define RPWM 10  
#define EN 11  
  
//Motor Left
```

```
#define RPWM2 6
#define LPWM2 7
#define EN2 8

void setup() {
    pinMode(EN, OUTPUT);
    pinMode(EN2, OUTPUT);

    digitalWrite(EN, 1);
    digitalWrite(EN2, 1);

}

void loop() {
    //Adelante
    //MOTOR 1
    analogWrite(LPWM, 0);
    analogWrite(RPWM, 255);

    // MOTOR 2
    analogWrite(LPWM2, 0);
    analogWrite(RPWM2, 255);
```

```
}
```

## Código para todos los canales

```
// FS-IA6 Receiver Test

#define N 6      // Define channels

#define ch1 22   // No activated

#define ch2 24   // Down & Up

#define ch3 26   // Motors Left & Right

#define ch5 30   // Active Weapon

#define ch6 32   // Turn On/Off Robot Battle

int ch[N]; // Array

void setup() {

    Serial.begin(115200);

    pinMode(ch1, INPUT);

    pinMode(ch2, INPUT);

    pinMode(ch3, INPUT);

    pinMode(ch5, INPUT);

    pinMode(ch6, INPUT);

}

void loop() {

    ch[0] = pulseIn(ch1, HIGH); // Channel no activated

    ch[1] = pulseIn(ch2, HIGH); // Down & Up

    ch[2] = pulseIn(ch3, HIGH); // Motors Left & Right

    ch[4] = pulseIn(ch5, HIGH); // Active Weapon
```

```
ch[5] = pulseIn(ch6, HIGH); // Turn On/Off Robot Battle

Serial.print("Channel 1 = :");
Serial.print(ch[0]);
Serial.print(", ");
Serial.print("Channel 2 = :");
Serial.print(ch[1]);
Serial.print(", ");
Serial.print("Channel 3 = :");
Serial.print(ch[2]);
Serial.print(", ");
Serial.print("Channel 5 = :");
Serial.print(ch[4]);
Serial.print(", ");
Serial.print("Channel 6 = :");
Serial.println(ch[5]);
}
```

## 5. Prueba de Motores

En esta prueba lo que buscamos es poner a funcionar los motores, así como controlarlos con el fin de posteriormente desarrollar las habilidades de movimiento que podremos aplicar a nuestro robot de combate como se muestra en la figura 11.5.1.

```

int D[2] = {9, 8}; // seleccion de la direccion
int M[2] = {4, 7}; // seleccion del motor
int pwmpin[2] = {5, 6}; // 5 motor1 y 6 motor2
int en[2] = {A0,A1};

void setup() {
    for (int i = 0; i < 2; i++) {
        pinMode (M[i], 1);
        pinMode (D[i], 1);
        pinMode(pwmpin[i], 1);
    }
}
void loop() [
    //Motor 2 giro antihorario
    digitalWrite (M[0], 1); //4
    digitalWrite (M[1], 0); //7
    digitalWrite (D[0], 0); //9
    digitalWrite (D[1], 0); //8
    analogWrite(pwmpin[0], 255); //5
    analogWrite(pwmpin[1], 255); //6
    delay(1000);

    //Motor 1 giro antihorario
    digitalWrite (M[0], 0); //4
    digitalWrite (M[1], 0); //7
    digitalWrite (D[0], 0); //9
    digitalWrite (D[1], 1); //8
    analogWrite(pwmpin[0], 255); //5
    analogWrite(pwmpin[1], 255); //6
    delay(1000);

    //Motor 2 giro horario
    digitalWrite (M[0], 0); //4
    digitalWrite (M[1], 1); //7
    digitalWrite (D[0], 1); //9
    digitalWrite (D[1], 1); //8
    analogWrite(pwmpin[0], 255); //5
    analogWrite(pwmpin[1], 255); //6
    delay(1000);

    //Motor 1 giro horario
    digitalWrite (M[0], 1); //4
    digitalWrite (M[1], 1); //7
    digitalWrite (D[0], 1); //9
    digitalWrite (D[1], 0); //8
    analogWrite(pwmpin[0], 255); //5
    analogWrite(pwmpin[1], 255); //6
    delay(1000);
]

```

Figura 11.5.1 Código para prueba de motores Monster Moto Shield (Arduino, Satomi Minami, 2022).

En este apartado se realizaron pruebas a diferentes motores para descifrar cuales tenían mejor torque, la prueba se realizó poniendo un disco de cierra adaptado eje del motor así se realizaron cortes a diferentes materiales como metal, madera y acrílico.

Se pudo obtener el resultado de dos motores que trabajan a 12v 3a que soportan bastante peso y es difícil frenarlos cuando se les aplica peso ya que tiene un aguante de un poco más de 40 kilos, los motores mencionados se muestran a continuación en la figura 11.5.2.



Figura 11.5.2 Prueba de motores 12V 3<sup>a</sup> (Samantha Padilla, 2022)

Los motores antes mencionados figura 11.5.2 reprobaron en su totalidad las pruebas realizadas puesto que no lograron mover el arma colocada en una maqueta controlada, además de obtener consumos exagerados de corriente, +10A, la prueba consistió en fijar el motor a un tablón donde se encontraba fijada el arma, ambos una estrella para cadena, se le colocó la cadena quedando ajustada sin ser apretado y se les aplicó un voltaje de una fuente capaz de proporcionar 12V a 9A, se encendió la fuente, el motor no giro más sin embargo tensó la cadena hasta conseguir que la fuente se apagara por sí sola, al tercer intento apago la fuente quemando un par de cables en su interior, al reemplazarlos la fuente de alimentación funcionó y se concluyó en el descarte de los motores. El procedimiento de pruebas se muestra en las figuras 11.5.3 y 11.5.4.

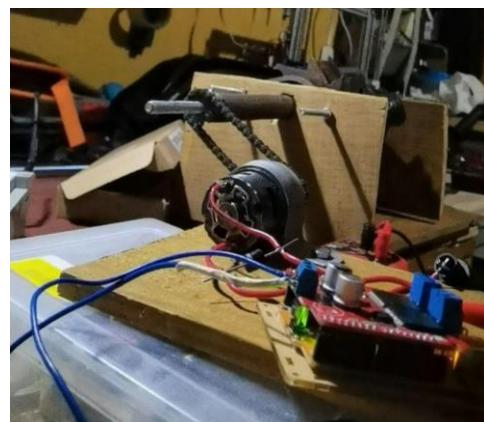


Figura 11.5.3 Prueba 1 de torque con el motor 12V (Satomi Minami, 2022).



Figura 11.5.4 Prueba 2 de torque con el motor 12V (Satomi Minami, 2022).

### Motorreductor arma

Es un motor con una caja de engranajes de la marca Kelvin modelo K80 el cual cuenta con un motor de 48V a 1.05A a 0.03Kw el cual en nuestras pruebas logró levantar 22.4 kilogramos sin problema alguno, el motor funcionando sin esfuerzo consume un promedio de 228mA como se muestra en la figura 11.5.5.

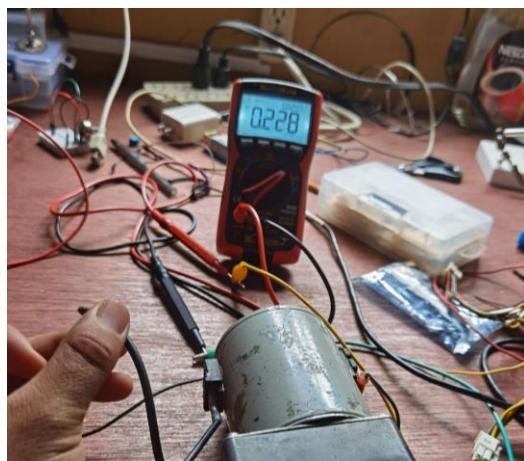


Figura 11.5.5 Prueba 1 de Amperaje del motor K80 (Satomi Minami, 2022).

Mientras que cargando los 22.44 kilogramos obtuvimos un consumo promedio de 357 mA como se muestra en la figura 11.5.6.

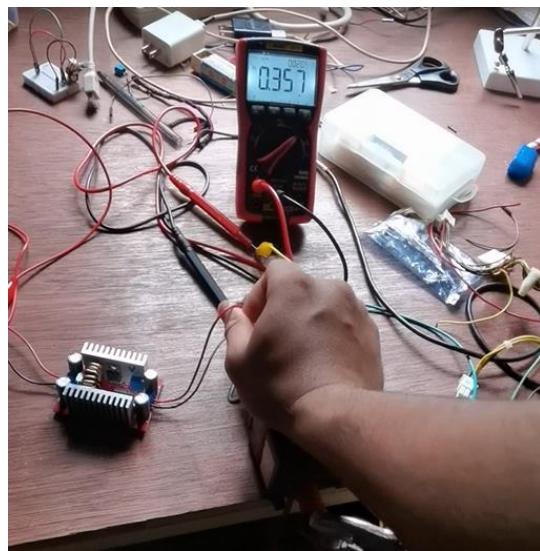


Figura 11.5.5 Prueba 2 de Amperaje del motor K80 (Satomi Minami, 2022).

Enseguida se realizó una prueba para levantar peso, en este caso utilizamos una pesa de 9.22 kg como se muestra en la figura 11.5.6. y 11.5.7



Figura 11.5.6 Pesando pesos (Satomi Minami, 2022).

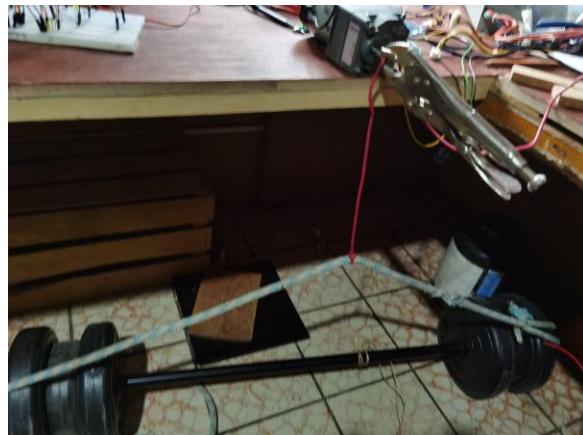


Figura 11.5.7 Prueba para levantar peso (Satomi Minami, 2022).

La forma en que se comprobó el funcionamiento del arma fue de la siguiente forma, se fijó el motor con su estrella, se fijó el arma en un mismo tablón ambos como se muestra en la figura 11.5.8.

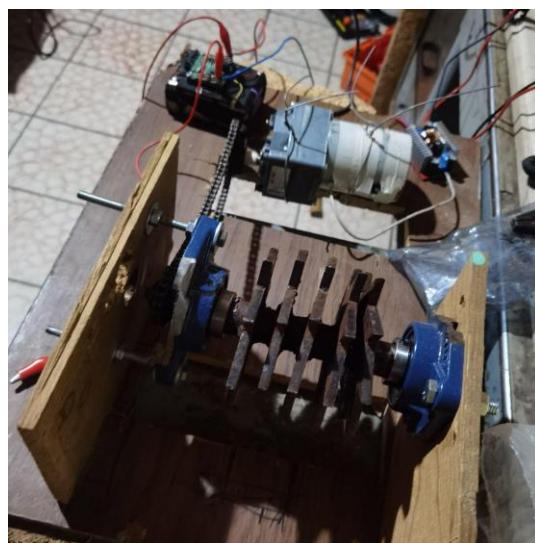
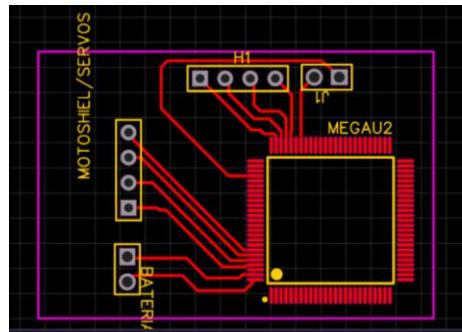


Figura 11.5.8 Maqueta del mecanismo del arma (Satomi Minami, 2022).

## 6. Circuitos electrónicos

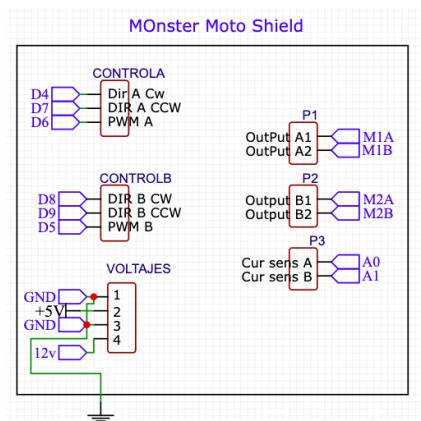
### Microcontrolador (salidas y entradas)

Se definieron las entradas digitales en el arduino para controlar los PWM para el moto shield desde el pin 4 hasta el 9. También se utilizaron 4 pines analogicos el A0 y A1 para la activación de los motores en la shield y el A2 y A3 para la activación de dos servos que se utilizarán, así como salidas de voltaje a 5V para la alimentación de los servos y de la moto shield como se muestra en la figura 11.6.1.



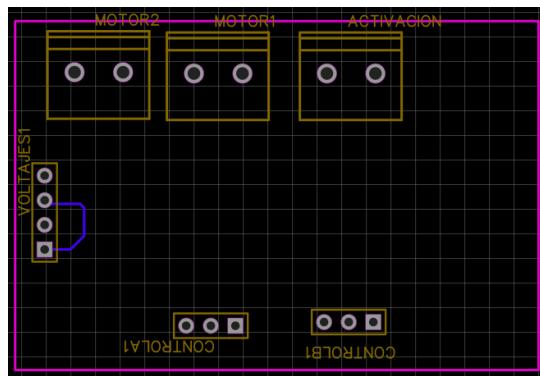
**Figura 11.6.1** Desarrollo de la PCB, Borrador 1 (Hugo Henriquez, 2022).

Creación de PCB del microcontrolador, en esta parte se realizó la simulación de las conexiones de la PCB del microcontrolador y las salidas que tiene, aunque usaremos la placa Arduino pero no nos deja crearla ya que son módulos y EAsyEDA no nos deja, así que recurrimos a utilizar el microcontrolador por separado y hacerle las conexiones que llevará como se muestra en la figura 11.6.2.



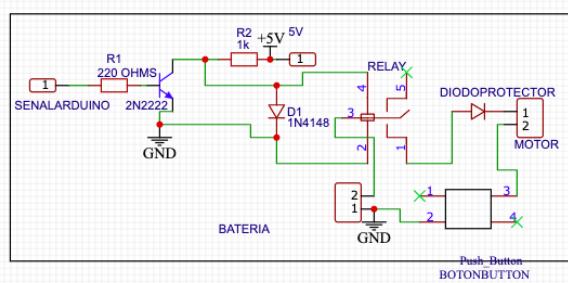
**11.6.2** Desarrollo de la PCB, Borrador 2 (Hugo Henríquez, 2022).

Se tuvieron que poner las entradas en el moto shield y solo poner las conexiones simulando el módulo, al igual que en el Arduino para el momento de hacer las conexiones en físico no haya confusiones y basarnos en este diagrama, al igual que en el Arduino también se utilizara un fuente de voltaje extra para la alimentación ahora de los motores de 12v y se les pusieron sus respectivas salidas hacia los motores, así como las entradas analógicas para la activación de los mismos como se muestra en la figura 11.6.3.



*Figura 11.6.3. Desarrollo de la PCB, Borrador 2 (Hugo Henriquez, 2022).*

Esta parte al igual que Arduino solo es una simulación del módulo del Monster Moto Shield ya que aquí se conectarán por cables tanto las entradas y las salidas de los motores, controles, voltajes y activación del módulo.



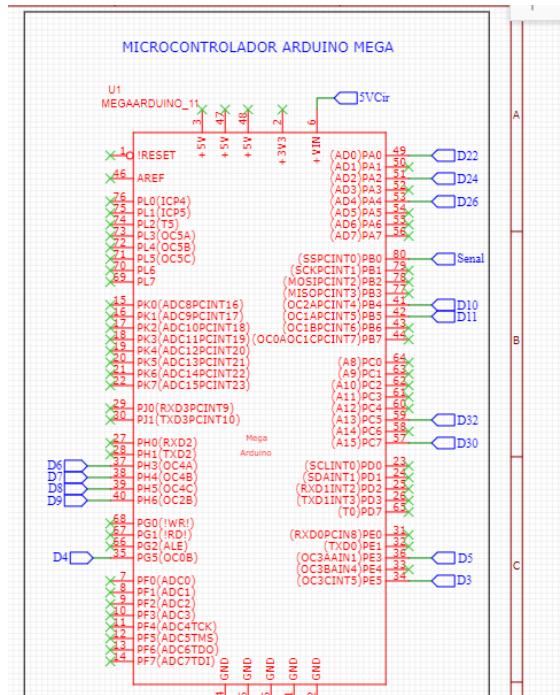
*Figura 11.6.4 Desarrollo del circuito del arma, Borrador 1 (Hugo Henriquez, 2022).*

Para este circuito se adaptaron los componentes a la corriente que se utilizará para que el arma gire, así como una protección para cuando golpee y haya picos de voltaje y no se regresen y puedan causar algún daño se activará por medio de la señal del control como se muestra en la figura 11.6.4.

Después de realizar algunas pruebas tuvimos que rectificar los circuitos del arma para la protección de la misma y para controlar manualmente el arma. Y con los pines ya definidos ahora ya tenemos los circuitos completos y necesarios.

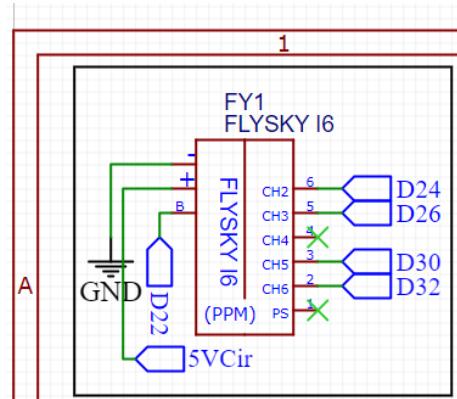
Se hicieron algunas modificaciones en los circuitos una vez que se definieron exactamente cuales se utilizarán ya que ya se hicieron pruebas para ver que la conexiones que se hicieron y las pruebas que se realizaron funcionaran correctamente como se muestra en las figuras 11.6.5, 11.6.6 y 11.6.7.

- Circuitos en el microcontrolador
- Circuitos en el Motor Shield (puente H)
- Circuito para activación del arma



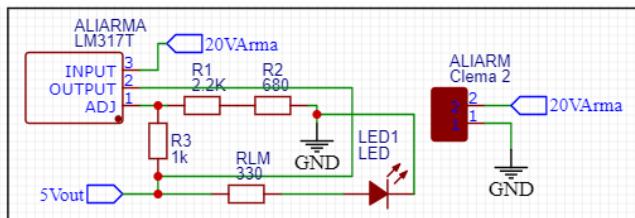
**Figura 11.6.4 Desarrollo del circuito del microcontrolador, Borrador 1 (Hugo Henríquez, 2022).**

En la figura 11.6.4 tenemos los pines que se utilizarán para el control de todo el robot que los cuales verán para que son cada uno cuando describamos las partes de los circuitos.



**Figura 11.6.5 Desarrollo de la comunicación RF, Borrador 1 (Jennifer González, 2022).**

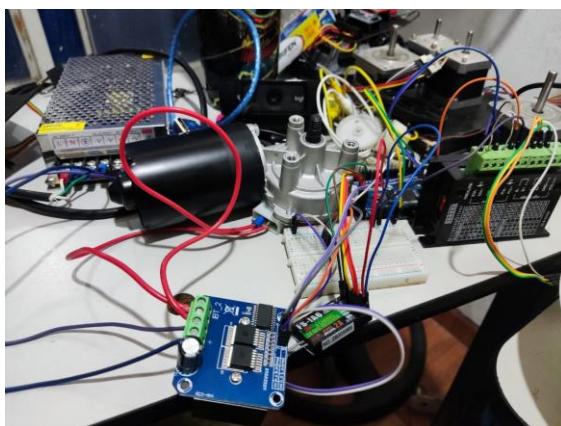
En la figura 11.6.5 tenemos el receptor del control que utilizaremos del cual solo sacaremos 4 canales con los respectivos pines del arduino y su alimentación a 5v.



*Figura 11.6.6 Desarrollo del circuito del arma, Borrador 2 (Hugo Henriquez, 2022).*

En la figura 11.6.6 tenemos un reductor de voltaje donde conectaremos a la clema (aliarm) el voltaje de nuestras baterías y de ahí le entra a un LM317(ALIARMA) que el cual tiene un arreglo de resistencias que el cual reduce el voltaje de entrada de 20v a un voltaje de salida de 5v.

Los 3 diagramas anteriores (figura 11.6.6, 11.6.5 y 11.6.4) se realizaron las conexiones en físico con los drivers y el código, comprobando que todas las conexiones fueran correctas y una vez comprobadas maquilar la PCB. Estas conexiones en físico se muestran en la figura 11.6.7.



*Figura 11.6.7 Conexiones finales del robot (Hugo Henríquez).*

## 7. Carcasa

En esta base (figura 11.7.1) será donde irán montados los servos, los motores, algunas de las PCB y donde sentará la base del arma, así como el chasis (versión 1).

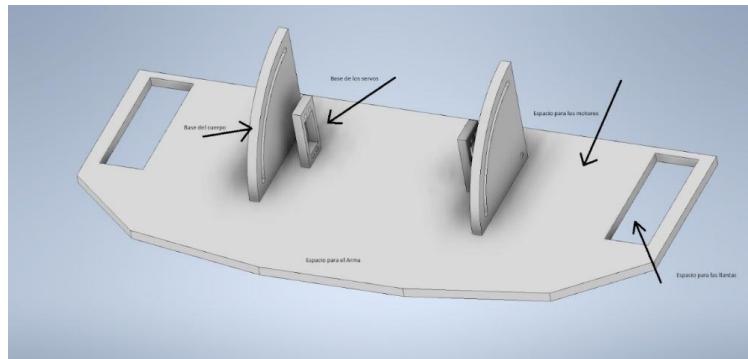


Figura 11.7.1 Modelo para dimensionar el robot (Inventor AutoDesk, Jorge Satomi, 2022)

Base del cuerpo: Sitio en el lugar donde la base se guiará para desplazarse según lo indiquen los servos.

Base de los servos: Sitio donde se fijarán los servos.

Espacio de los motores: Lugar donde estarán fijados los motores (falta colocar los agujeros para atornillarlo pues que no se ha definido un motor exacto).

Espacio para las llantas: es un espacio diseñado para una llanta de hule de 5in estándar.

Espacio por el arma: Es el espacio dedicado para el arma, no presenta información concreta puesto que se carece de una elección concreta.

Al final se realizaron cambios sumamente radicales con el diseño del robot debido a que las pruebas que se realizaron con los motores y los servomotores no eran suficientes para que nuestro diseño principal funcionará, por lo tanto, se decidió fijar el arma y hacer un barril giratorio con picos. A continuación, en la figura 11.7.2 se muestra el diseño de la carcasa con todas las partes ensambladas, mientras que en la figura 11.7.3 el diseño de la rampa.

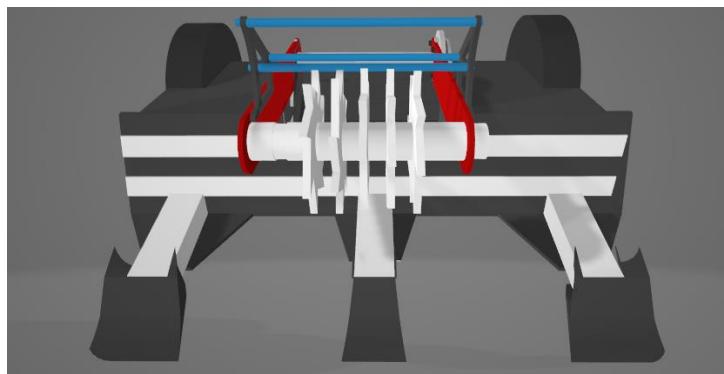
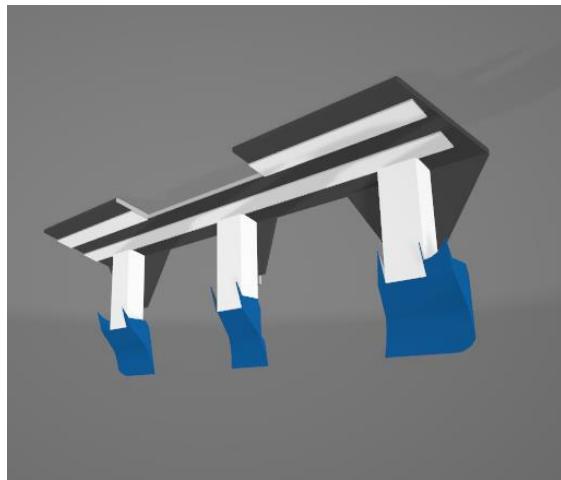


Figura 11.7.2 Diseño de la carcasa (AutoDesk Inventor, Jennifer González, 2022).

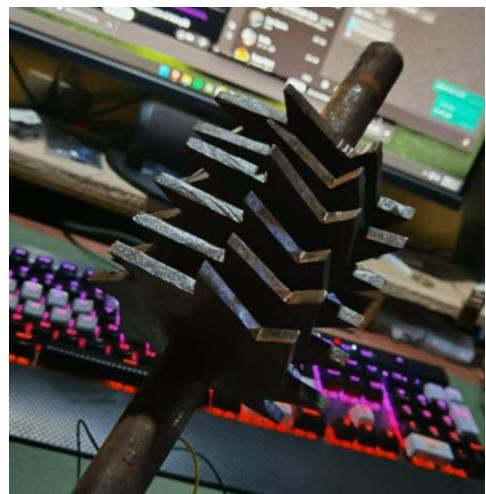


*Figura 11.7.2 Diseño de la rampa (AutoDesk Inventor, Jennifer González, 2022).*



*Figura 11.7.3 Diseño del arma (AutoDesk Inventor, Maricarmen Ramírez, 2022).*

Una vez que todo el equipo estuvo de acuerdo con el boceto y el modelo presentado, se comenzó con el desarrollo físico del chasis, comenzando por el arma la cual se muestra en la figura 11.7.4, mientras que en la 11.7.5 donde se puede observar el arma ya armada, con la pieza (picos) en el eje, sus chumaceras y el piñón correspondiente. Mientras que el resto de las figuras (11.7.6 - 11.7.20) se observa el resto del procedimiento al armar el chasis, que implicó tareas como cortar, marcar, soldar, remachar y pintar.



*Figura 11.7.4 Arma (Hugo Henriquez, 2022).*



*Figura 11.7.5 Arma con mecanismo (Satomi Minami, 2022).*



*Figura 11.7.6 Buje de los motores con torno (Satomi Minami y Hugo Henriquez, 2022).*



*Figura 11.7.7 Cortando la base del robot (Satomi Minami, 2022).*



*Figura 11.7.8 Cortando el eje del arma (Hugo Henriquez, 2022).*



*Figura 11.7.9 Cortando el eje del arma (Hugo Henriquez, 2022).*



*Figura 11.7.10 Cortando lámina para los picos (Samantha Padilla, 2022).*



*Figura 11.7.11 Picos cortados y soldados (Samantha Padilla, 2022).*



*Figura 11.7.12 Quitando rebabas de las paredes (Karen Samantha, 2022).*



Figura 11.7.13 Puliendo piezas(Hugo Henriquez, 2022).



Figura 11.7.14 Doblando paredes del robot (Jennifer González, 2022).



Figura 11.7.15 Pesando el material a utilizar (Maricarmen Ramirez, 2022).



**Figura 11.7.16** Acomodando partes del robot (Jennifer González, Samantha Padilla, Maricarmen Ramírez, 2022).



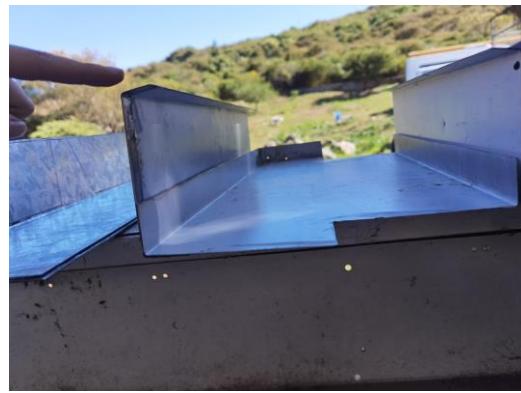
**Figura 11.7.17** Partes del robot acomodadas (Jennifer González, Samantha Padilla, Maricarmen Ramírez, 2022).



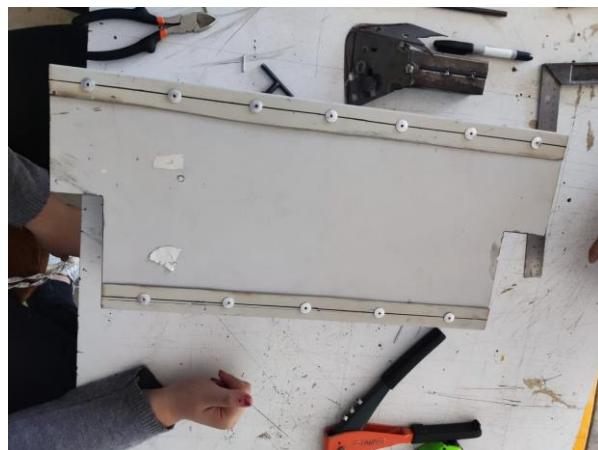
**Figura 11.7.18** Quitando el carbón de las piezas (Maricarmen Ramirez, 2022).



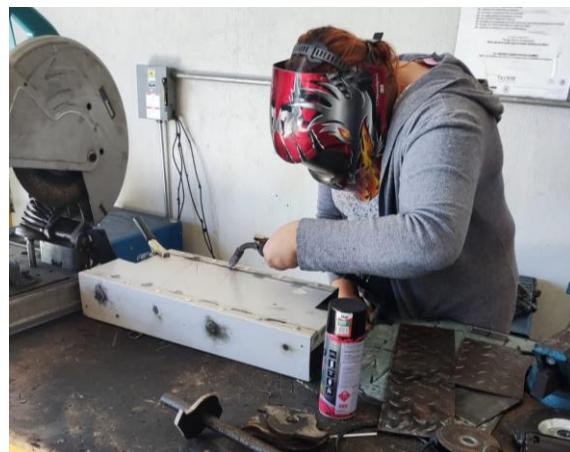
**Figura 11.7.19** Bases de los motores (Satomi Minami, 2022).



**Figura 11.7.20** Colocando paredes (Jennifer González, 2022).



**Figura 11.7.21** Remachando el robot (Maricarmen Ramírez, Jennifer González y Samantha Padilla, 2022).



**Figura 11.7.22** Soldando con microalambre (Samantha Padilla, 2022).



**Figura 11.7.23** Remache de la rampa al robot (Satomi Minami y Samantha Padilla, 2022).



*Figura 11.7.24 Soldando baleros a la pared (Jennifer González, 2022).*



*Figura 11.7.25 Soldando los picos a la rampa (Jennifer González, 2022).*



**Figura 11.7.26** Soldando el resto de las paredes (Satomi Minami, 2022).



**Figura 11.7.27** Pintando el robot (Jennifer González, 2022).



**Figura 11.7.28** Pintando la rampa del robot (Jorge Satomi, 2022).



*Figura 11.7.29 Pintando el arma (Jennifer González, 2022).*



*Figura 11.7.30 Logo del robot en MDF (Jennifer González, 2022).*



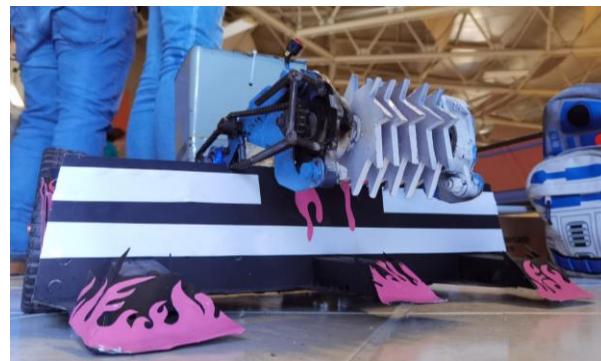
*Figura 11.7.31 Caja de acrílico para proteger el proyecto (Jennifer González, 2022).*



*Figura 11.7.32 Colocando cadenas en el mecanismo (Satomi Minami y Hugo Hernriquez, 2022).*



*Figura 11.7.33 Probando la fuerza del arma (Buggy Team, 2022).*



*Figura 11.7.34 Buggy Battlebot terminado (Buggy Team, 2022).*



*Figura 11.7.35 Pesando el robot (Buggy Team, 2022).*



*Figura 11.7.36 Comprobando dimensiones del battlebot (Buggy Team, 2022).*

## 8. Prueba Llantas

En estas pruebas sabiendo que si conectamos el motor a la carcasa y las llantas al motor prácticamente el motor sufriría de la máxima comprensión que causa el peso de la carcasa, además que el motor se ve obligado a realizar un mayor esfuerzo por lo que nos dimos a la tarea de pensar e idear una forma de evitar que lo mencionado ocurra, por lo que pensamos que colocando unos baleros Sobre una base que se encuentre fija al chasis de la carcasa y los baleros a esta misma donde los baleros sean la base del eje de nuestras llantas, lo que causaría que el robot

---

---

pueda rodar ya sea con o sin motor por lo que al conectarle el motor el mismo solo se encargaría de liberar la presión que existe en las llantas y no con todo el peso del carro, se realizó una maqueta para poder poner a prueba lo antes mencionado como se muestra en la figura 11.8.1.



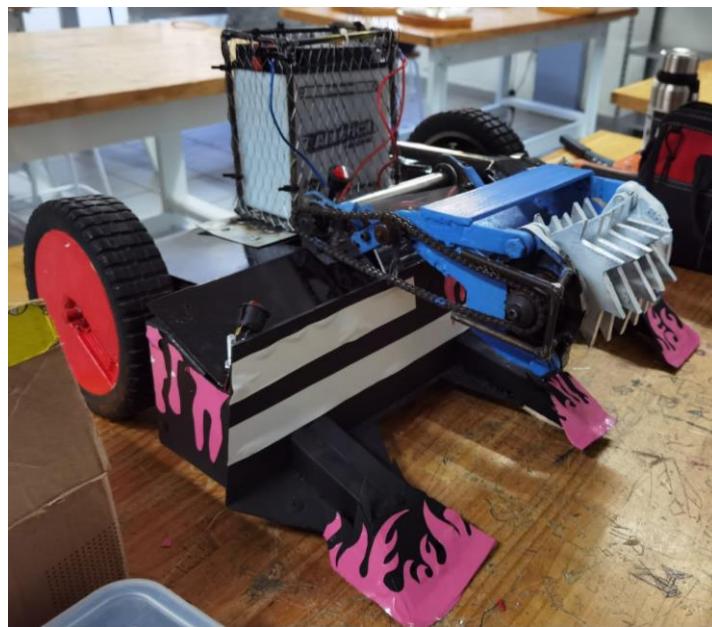
*Figura 11.8.1 Maqueta del eje de las llantas (Satomi Minami, 2022).*

## Resultados

12. Resultados, planos, gráficos, prototipos, manuales, programas análisis estadísticos, modelos matemáticos, simulaciones, normatividades, regulaciones y restricciones, entre otros.

### Battlebot Buggy

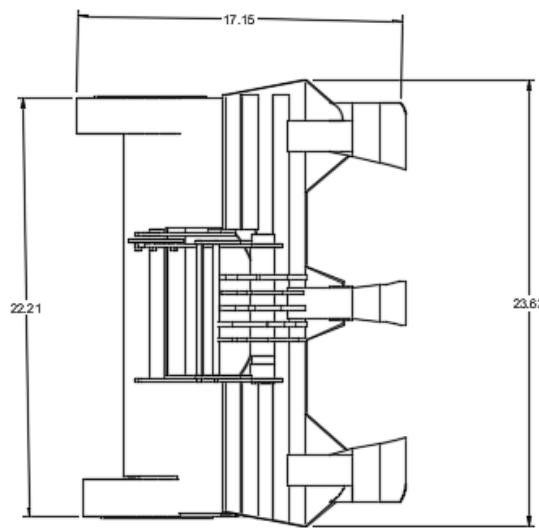
El robot obtuvo un peso total de 28.7 kilogramos y se muestra terminado en la figura 12.1.0.



*Figura 12.1.0 Buggy Battlebot (Satomi Minami, 2022).*

### Planos.

El plano del diseño del robot final se muestra en la figura 12.1.19.



*Figura 12.1.19. Ensamble del robot (Maricarmen Ramírez, 2022).*

PCB

Resultados de la PCB de la parte del arma presentada en 3D simulando en cómo se vería físicamente en la figura 12.1.20.

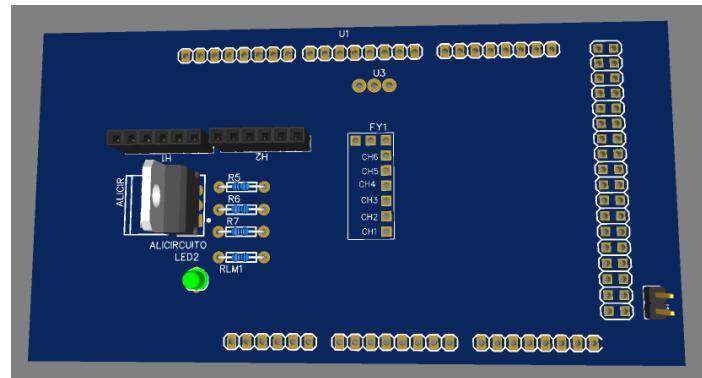


Figura 12.1.20 PCB principal en 3D (EasyEDA, 2022).

Resultados de la PCB controladora general en la simulación 3D la cual llevará el control de todo el robot se muestra en la figura 12.1.21.

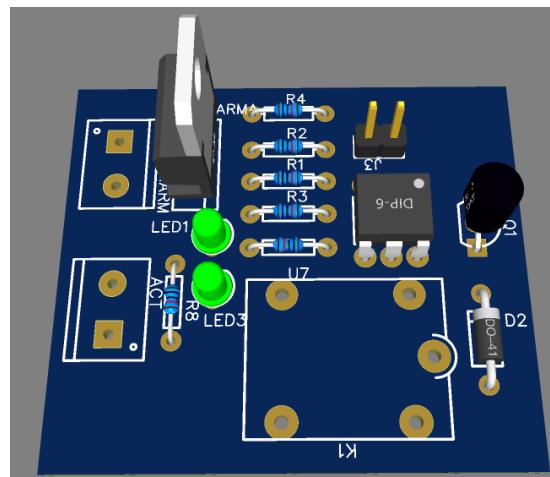


Figura 12.1.21 PCB de activación del arma, simulación 3D(EasyEDA, 2022).

Ahora se muestra el diagrama final del circuito implementado en el robot así como la PCB finalizada que se utilizó para el funcionamiento del proyecto, los cuales se observan en la figura 12.1.22, 12.1.23 y 12.1.24.

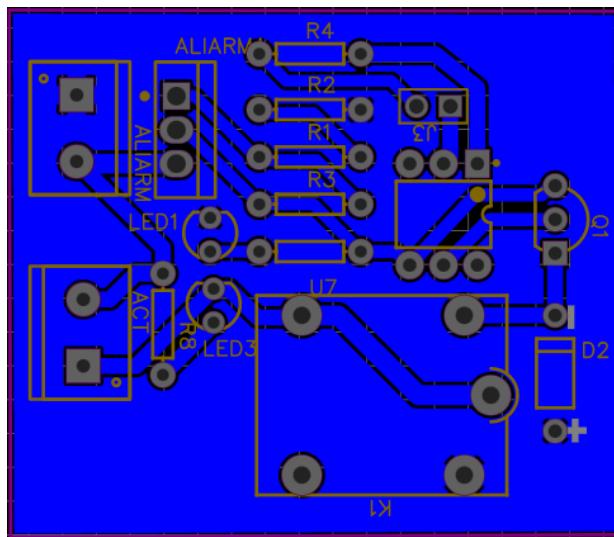


Figura 12.1.23 PCB de activación del arma (EasyEDA, 2022).

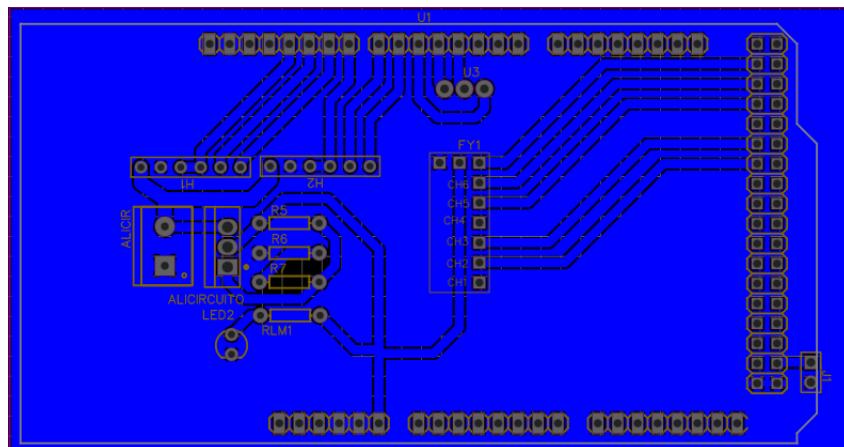
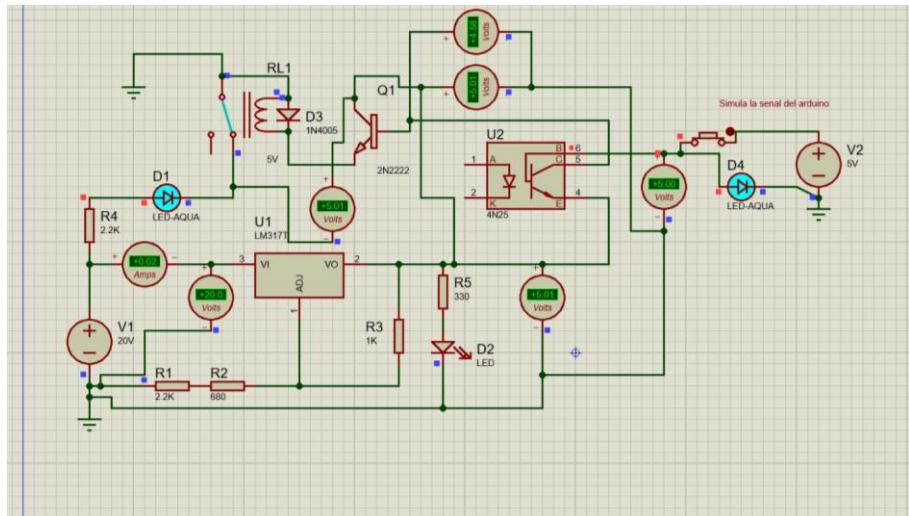


Figura 12.1.24 PCB de activación del arma, simulación 3D (EasyEDA, 2022).

Simulación Circuito de activación del arma.

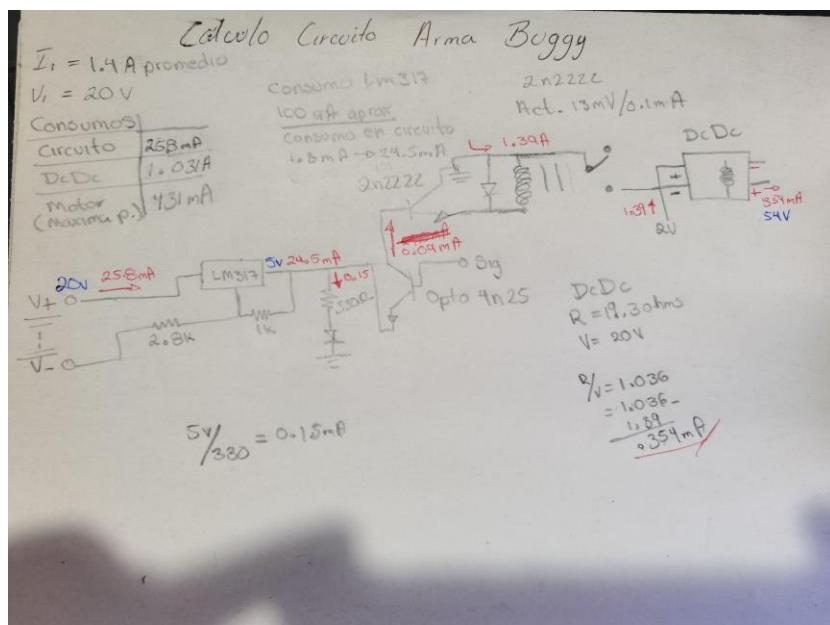
Se pasó a realizar la simulación del arma con el fin de comprobar su funcionamiento, consumos, y viabilidad del mismo, la simulación se muestra a continuación en la figura 12.1.25.



**Figura 12.1.25 Simulación del circuito de activación para el arma (Isis Proteus, 2022).**

### Cálculo del consumo del circuito

En la siguiente imagen se muestran los cálculos realizados del consumo de amperaje del circuito de arma, debido a que necesitábamos saber el consumo de amperaje para no tener problemas a la hora de alimentarlo, mientras que el resto (el circuito del microcontrolador) no fue necesario, pues cada driver en su datasheet muestra cuánta corriente consume y el voltaje de entrada que se administra. Los cálculos realizados se muestran en la figura 12.1.26.



**Figura 12.1.26 Cálculo del circuito del arma (Satomi Minami, 2022).**

---

---

## Código

```
// FS-IA6 Receiver Test

#define N 6      // Define channels

#define ch1 22   // No Activated

#define ch2 24   // Down & Up

#define ch3 26   // Motors Left & Right Down & Up

#define ch5 30   // Active Weapon

#define ch6 32   // Turn On/Off Robot Battle

int ch[N]; // Channel Array

//Motor Right

#define LPWM 9

#define RPWM 10

#define EN 11

//Motor Left

#define RPWM2 6

#define LPWM2 7

#define EN2 8

//Weapon

#define weaponPin0 53

#define weaponPin 49

#define weaponPin2 47

#define weaponPin3 45
```

```
#define weaponPin4 43
#define weaponPin5 41
#define weaponPin6 39
#define weaponPin7 37

void setup() {
    Serial.begin(115200);
    //Iniciar canales
    pinMode(ch1, INPUT);
    pinMode(ch2, INPUT);
    pinMode(ch3, INPUT);
    pinMode(ch5, INPUT);
    pinMode(ch6, INPUT);

    pinMode(EN, OUTPUT);
    pinMode(EN2, OUTPUT);

    //WEAPON
    pinMode(weaponPin, OUTPUT);
    pinMode(weaponPin0, OUTPUT);
    pinMode(weaponPin2, OUTPUT);
    pinMode(weaponPin3, OUTPUT);
    pinMode(weaponPin4, OUTPUT);
```

```
pinMode(weaponPin5, OUTPUT);
pinMode(weaponPin6, OUTPUT);
pinMode(weaponPin7, OUTPUT);

}

void loop() {
    // put your main code here, to run repeatedly:
    ch[0] = pulseIn(ch1, HIGH); // Motors Left & Right
    ch[1] = pulseIn(ch2, HIGH); // Motors Down & Up
    ch[2] = pulseIn(ch3, HIGH); // Weapon Down & Up
    ch[4] = pulseIn(ch5, HIGH); // Active/Desactive Weapon
    ch[5] = pulseIn(ch6, HIGH); // Turn On/Off Robot Battle

    //If buggy robot is on, active all the channels of control rf
    if (ch[4] < 1400) {
        robotMove();
        activeWeapon();
        Serial.println("ROBOT ON");
        digitalWrite(EN, 1);
        digitalWrite(EN2, 1);
    } //if not, keep all channels desactivate
    else {
        analogWrite(EN, 0);
        analogWrite(EN2, 0);
    }
}
```

```
    Serial.println("ROBOT OFF");

}

}

void activeWeapon() {
    if (ch[5] < 1400) {

        Serial.println("WEAPON ON");

        digitalWrite(weaponPin0, HIGH);
        digitalWrite(weaponPin, HIGH);
        digitalWrite(weaponPin2, HIGH);
        digitalWrite(weaponPin3, HIGH);
        digitalWrite(weaponPin4, HIGH);
        digitalWrite(weaponPin5, HIGH);
        digitalWrite(weaponPin6, HIGH);
        digitalWrite(weaponPin7, HIGH);

    } else {

        Serial.println("WEAPON OFF");

        digitalWrite(weaponPin0, LOW);
        digitalWrite(weaponPin, LOW);
        digitalWrite(weaponPin2, LOW);
        digitalWrite(weaponPin3, LOW);
        digitalWrite(weaponPin4, LOW);
        digitalWrite(weaponPin5, LOW);
        digitalWrite(weaponPin6, LOW);
        digitalWrite(weaponPin7, LOW);

    }
}
```

```
        }

    }

}

void robotMove() {

    //Channel 2 UP AND DOWN

    if (ch[1] > 1400 && ch[1] < 2000) { // FORWARD

        Serial.println("FORWARD");

        //MOTOR 1

        analogWrite(LPWM, 200);

        digitalWrite(RPWM, 0);

        // MOTOR 2

        analogWrite(LPWM2, 200);

        digitalWrite(RPWM2, 0);

    }

    if (ch[1] > 980 && ch[1] < 1200) { // BACK

        Serial.println("BACK");

        //MOTOR 1

        digitalWrite(LPWM, 0);

        analogWrite(RPWM, 200);
```

```
// MOTOR 2

digitalWrite(LPWM2, 0);
analogWrite(RPWM2, 200);

}

if (ch[1] > 1380 && ch[1] < 1400 && ch[2] > 1500 && ch[2] < 1580) { // STOP

Serial.println("STOP");

//MOTOR 1

analogWrite(LPWM, 0);
analogWrite(RPWM, 0);

// MOTOR 2

analogWrite(LPWM2, 0);
analogWrite(RPWM2, 0);

}

if (ch[2] > 1000 && ch[2] < 1350) { // LEFT

Serial.println("LEFT");

//MOTOR 1

analogWrite(LPWM, 200);
digitalWrite(RPWM, 0);

// MOTOR 2

digitalWrite(LPWM2, 0);
analogWrite(RPWM2, 200);
```

```
}

if (ch[2] > 1600 && ch[2] < 2000) { // RIGHT

    Serial.println("RIGHT");

    //MOTOR 1

    digitalWrite(LPWM, 0);

    analogWrite(RPWM, 200);

    // MOTOR 2

    analogWrite(LPWM2, 200);

    digitalWrite(RPWM2, 0);

}

}

// Made by Jennifer González
```

### 13. Actividades sociales realizadas en la empresa u localización

Las actividades realizadas para el desarrollo de “Buggy” se muestran en el cronograma de la figura 12.1.27 donde más adelante se detallan las tareas en casa actividad.

## Cronograma

Actividades	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8	Semana 9	Semana 10	Semana 11	Semana 12	Semana 13	Semana 14	Semana 15	Semana 16
Lluvia de ideas																
Planeacion																
Adquisicion de Materiales y componentes																
Prueba de componentes																
Programacion control RF																
Primer Diseno																
Elaboracion de simulacion de pruebas																
Correccion 1																
Elaboracion de los circuitos																
Prueba de circuitos																
Corte de piezas																
Union de piezas																
Correccion 2																
Elaboracion de puebas con robot armado																
Montado, pintado y Correccion 3																
Ajustes y presentacion																

**Figura 12.1.27 Cronograma de actividades (Excel, 2022).**

## Descripción de Actividades (cronograma)

### 1. Lluvia de ideas

Cada miembro del equipo proporciona ideas (sin importar lo locas que resulten), se realizará una recolección de las mejores, se evaluará qué ideas son las posibles de realizar y que se ajusten al presupuesto propuesto.

### 2. Planeación

A Partir de lo obtenido en la lluvia de ideas se pasará a planear la realización del robot, es decir, se realizará evaluación de pesos, motores, materiales a implementar, que arma se construirá, las actividades que se desempeñarán durante el proceso, la estructura del proceso los componentes que se utilizarán, probabilidades e investigaciones (precios, componentes, funciones, que tan complicado y costoso es conseguir “n” artículo...).

### 3. Adquisición de materiales y componentes

Se comenzará buscando los diversos materiales por internet, se capturaron, posterior se realizarán investigaciones sobre dónde adquirirlos físicamente para asistir a los a esos sitios a buscarlos, comparar precios y determinar dónde adquirir cada elemento y analizar dónde es más viable (dicha actividad se realizará en el momento de estar presente y solo se asistirá si se obtiene recomendaciones o ningún tipo de comentario negativo).

### 4. Prueba de componentes

Se realizará el funcionamiento de cada componente, como primera etapa se tendrá que ponerlo a funcionar de manera básica para posteriormente controlar y dominar su funcionamiento.

---

---

## 5. Programación control RF

Se estructurará la programación del control con su módulo, con el fin de determinar qué canales utilizar con que botones, con ello se experimentará con los componentes programables que llevará el proyecto

## 6. Primer diseño

Como primer proyecto se adentró una propuesta final sobre cómo quedará el robot de forma gráfica, con ello se determinarán cambios, nuevas propuestas y sugerencias

## 7. Elaboración de simulación de pruebas

se elaborará maquetas sobre el sistema de movimiento del robot, así como del arma y lo que sea considerado necesario con el fin de determinar qué mecanismos, estructuras y componentes si funcionan y cuales tendrán que ser descartados, modificados o cambiados

## 8. Corrección 1

Sobre esta corrección realizará el protocolo para realizar los posteriores cambios y/o modificaciones encontradas durante los anteriores procesos hasta el actual paso

## 9. Elaboración de los circuitos

Se desarrollarán los circuitos necesarios para controlar lo necesario en el proyecto

## 10. Elaboración de piezas

A partir del diseño antes mencionado se marcarán y cortaran y/o desarrollarán las piezas necesarias para la estructura, mecanismos, etc. que sean necesarias

## 11. Unión de piezas

Todas las piezas obtenidas comenzarán a montarse para conseguir la estructura planeada del proyecto, así como se montarán los diversos elementos que el mismo poseerá y se realizarán las pruebas necesarias

## 12. Corrección 2

A partir de lo obtenido en el desarrollo del paso anterior se realizarán los cambios y/o modificaciones necesarias

## 13. Elaboración de pruebas con robot armado

Una vez montado toda la estructura del robot se realizarán las pruebas necesarias para así poder exponer la veracidad de lo antes planeado

---

#### 14. Montado, pintado y corrección

Se realizarán todos los cálculos de diseño que se han planeado en el proceso y se realizará toda modificación necesaria

#### 15. Ajustes y presentación

Se pulirán los últimos detalles necesarios y se realizará como será presentado.

## Conclusiones

### 14. Conclusiones del proyecto, recomendaciones y experiencia profesional personal adquirida

Problemas que se presentaron a lo largo del proyecto y se tuvieron que cambiar por otros elementos.

1. Motores: 1. 2 motores a 12V a 2.2A de 12 Kg fuerza teóricos, no podrían con un contrincante, 2. Motor del arma 12V, amperaje variable, no logró mover el arma
2. Arma: muy pesada y robusta
3. Brazo del arma, crear mecanismo y motor de movimiento.
4. Señal de activación para el arma sufrió caída de tensión causando la no activación del arma pese a que la señal se envíe, es decir, esperamos una señal de 5V, donde en su lugar recibimos una señal de 3.8V lo que impedía la activación del nuestro transistor
5. Las llantas derrapan más de lo normal en cualquier superficie
6. los BTS7960 tuvieron un comportamiento inusual al cambio de dirección en cada motor al trabajar simultáneamente 2 BTS7960
7. El circuito del arma al poseer bobinas produce campos magnéticos nos intervienen con el circuito principal porque se tuvo que colocar estratégicamente en el chasis para evitar la interferencia de los mismos, se consideró encerrarlos en una jaula de Faraday, sin embargo, no fue necesario una vez ya colocados
8. desde el diseño principal al no poder mover cada lateral del robot se consideró modificar el diseño para que posea una rampa y el brazo, por ende, se tuvo que realizar un rediseño en su gran mayoría
9. fijación de arma puesto que los motores nema 23 no lograron cargar el peso del arma.
10. Mecanismo de movimiento del proyecto (como montar las llantas), para ello realizamos una investigación sobre cómo lograr que los motores no lidian directamente con el peso de proyecto, concluimos que al colocar un balero

---

---

que en el mismo siente el peso del proyecto y por el orificio interno valla la flecha de la llanta, lograremos que las llantas queden fijas y el motor solo se conecte a la flecha consiguiendo que el motor solo lidie con la presión sobre la llanta y no con el peso del proyecto. dicha investigación fue compartida con más equipos que presentaron múltiples problemas para fijar sus llantas con el motor.

### Conclusiones del proyecto

El proyecto resulta ser un proyecto muy dinámico en el que se requiere tener múltiples personalidades a cargo del mismo en diversas áreas como lo son:

- Mecanismos
- Estructuras
- Diseño
- Electrónica
- Programación

Por lo que tener por lo menos 1 persona a cargo de cada área se vuelve muy ágil para el proyecto pues se tiene concentración sobre esa área y reduce las posibilidades de que ocurra un error.

## Competencias desarrolladas

### 15. Competencias desarrolladas y/o aplicadas

Específicas:

Diseña algoritmos computacionales, así como logra su implementación utilizando un lenguaje de programación de alto nivel.

Utiliza la computadora de manera adecuada, particularmente para el uso de herramientas de programación y busca información en los medios electrónicos.

Manipula un robot industrial en forma manual a través de sus diversos sistemas de referencia para ubicar la herramienta de trabajo en puntos de interés.

---

## 1. Morfología del robot

- Identifica los componentes de un robot industrial para determinar los grados de libertad y espacio de trabajo de dicho manipulador y las disciplinas que intervienen en su análisis y diseño.
- Manipula un robot industrial en forma manual a través de sus diversos sistemas de referencia para ubicar la herramienta de trabajo en puntos de interés.

Genéricas:

- Capacidad de análisis y síntesis.
- Capacidad de organizar y planificar
- Toma de decisiones
- Fundación
- Planeación
- Ejecución
- Capacidad de análisis y síntesis
- Capacidad de organización y planificación
- Conocimientos básicos de la carrera
- Habilidades básicas en el modelado de sistemas
- Habilidades básicas de manejo de la computadora
- Habilidad de manejo de software de Ingeniería
- Habilidad para buscar y analizar
- Trabajo en equipo
- Habilidad para la búsqueda y análisis de información proveniente de fuentes de información.

---

---

## Fuentes de información

### 16. Fuentes de información

Aguayo, P. (2019, 23 septiembre). Arduino UNO. Arduino.cl - Compra tu Arduino en Línea.

DRIVER MOTOR PASO A PASO MONSTER MOTO SHIELD VNH2SP30. (s. f.). MACTRONICA.

*¿Cómo funciona la radiofrecuencia? | Telectrónica.* (s. f.).

Dorf R. (1997) "Sensors Nanoscience, Biomedical Engineering, and Instruments", *The Electrical Engineering Handbook Third Edition*.

Mayo Joe. (2010) "A beginner's guide Microsoft Visual Studio 2010", McGraw-Hill Estados Unidos  
Mirin Konstantin, (2009) "MySQL Workbench - The Database Modeling Tool for MySQL",  
programmersnotes.

Webster John G. (2006) *Encyclopedia of Medical Devices and Instrumentation Second Edition Volume 1*.

Aubry Christophe (2008). "Dreamweaver CS3: Para PC-MAC" primera edición. Barcelona: Ediciones ENI.

Daza Lemes (2008). "Evaluación clínico-funcional del movimiento corporal humano", editorial Panamericana, Colombia.

Disselhorst RAU G (2000). *Movimiento biomecánico: de la pierna hasta el brazo*, Mc Graw Hill, México.

Anibal Pliego Baturone (2011). *Robótica. Manipuladores y Robots Móviles*.

Antonio Barrientos, Luis Felipe Peñin, Carlos Balaguer, Rafael Aracil (2007). *Fundamentos de Robótica*. Mc Graw Hill, España - 2da Edición.

Comité Español Automático (CEA), 2008, *Libro Blanco de la Robótica*.

José Angulo Usategui (2005). *Introducción a la Robótica*. Thomson. España.

## Anexos

### 17. Anexos

En los siguientes links, se anexa el contenido más importante del proyecto:

- Construcción del robot:

[https://drive.google.com/file/d/1NfZMrrlf79idvVVKP8r47cw9q3RCO2i2/view?usp=share\\_link](https://drive.google.com/file/d/1NfZMrrlf79idvVVKP8r47cw9q3RCO2i2/view?usp=share_link)

- Funcionamiento del robot:

[https://drive.google.com/file/d/1NgSvFpLYcjgxWmk9p7nhg1pbUp0RofXL/view?usp=share\\_link](https://drive.google.com/file/d/1NgSvFpLYcjgxWmk9p7nhg1pbUp0RofXL/view?usp=share_link)

- Enlace al cronograma:

<https://mega.nz/file/leMm1KBb#OXVI-yC6aBRm8ZsXUtFSG317jxu4gCD8I2Fy6aRwheY>

- Proyecto del EasyEDA:

<https://u.eeasyeda.com/account/user/projects/index/detail?project=6da09b0a36134cd2905c5bf1a711b86f&folder=all>

## Diagrama

En la figura 12.1.22 se muestra el diagrama general utilizado en Buggy el battlebot, el cual se diseño mediante cálculos realizados y mostrados en el desarrollo del proyecto.

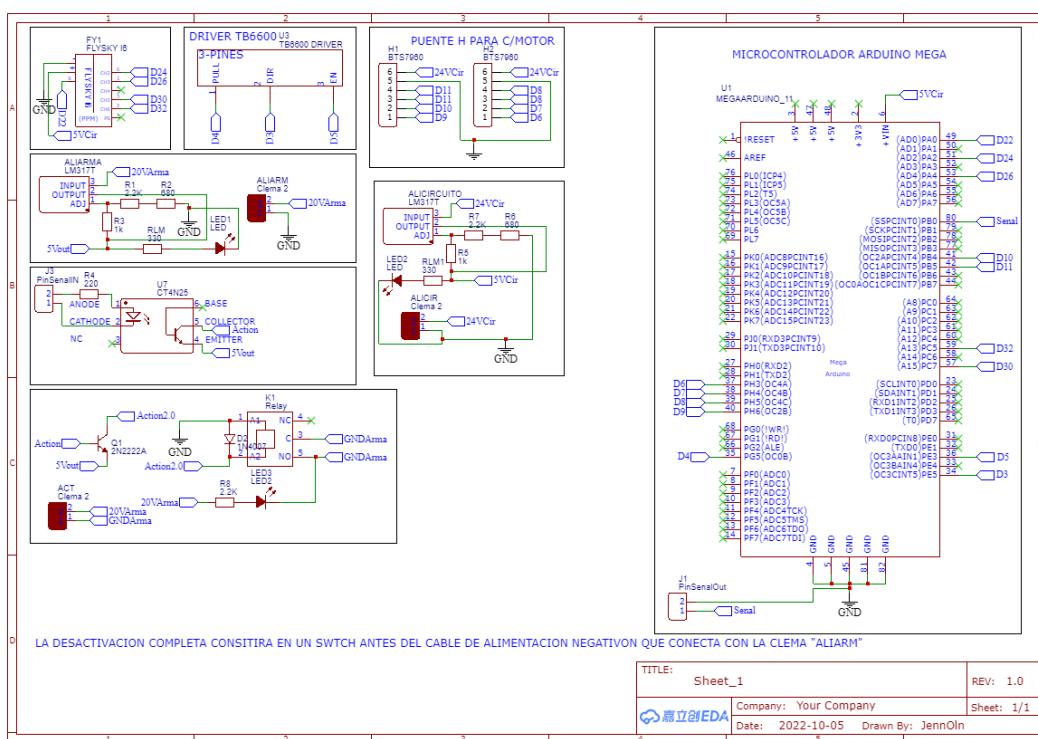


Figura 12.1.22 Diagrama de conexión del robot (EasyEDA, 2022).

## Planos

Los planos finalizados se muestran a partir de la figura 12.1.1 a la figura 12.1.19.

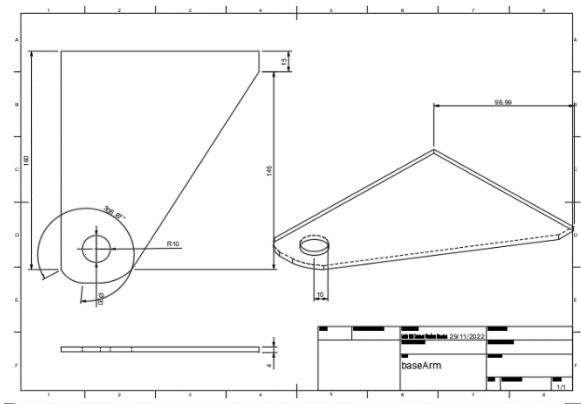


Figura 12.1.1. Soporte de arma (Maricarmen Ramírez, 2022).

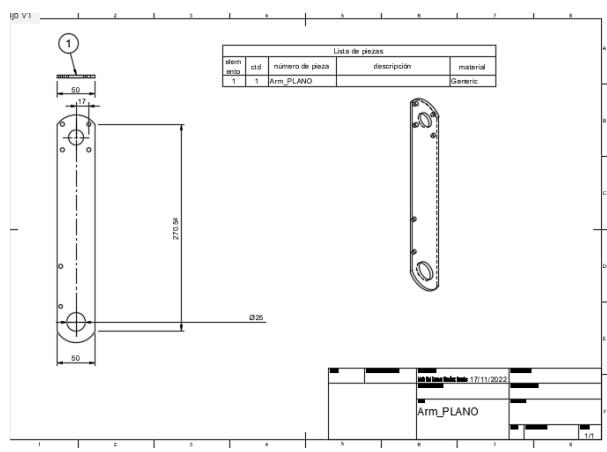


Figura 12.1.2. Plano arma (Maricarmen Ramírez, 2022).

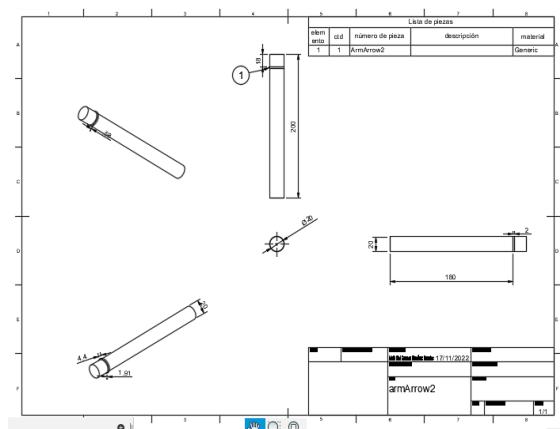


Figura 12.1.3. Arrow arma (Maricarmen Ramírez, 2022).

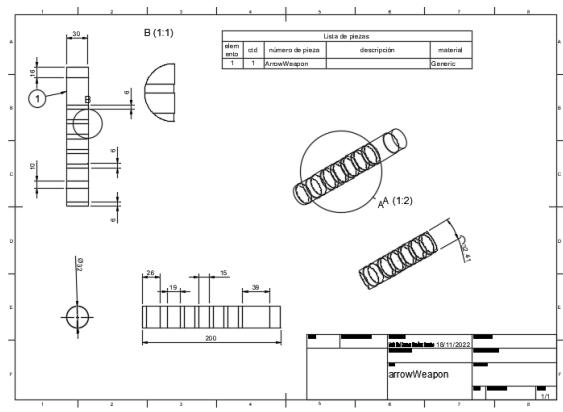


Figura 12.1.5. Arrow Weapon (Maricarmen Ramírez, 2022).

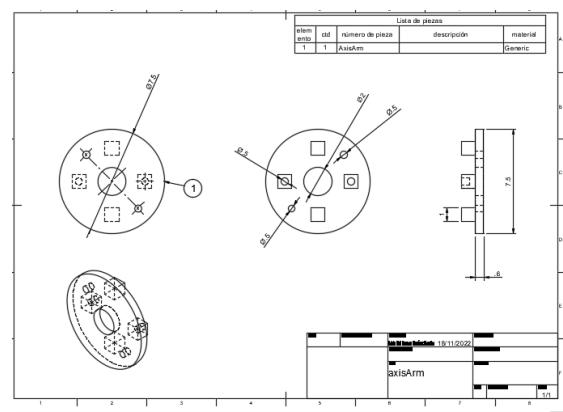


Figura 12.1.5. Arrow Weapon (Maricarmen Ramírez, 2022).

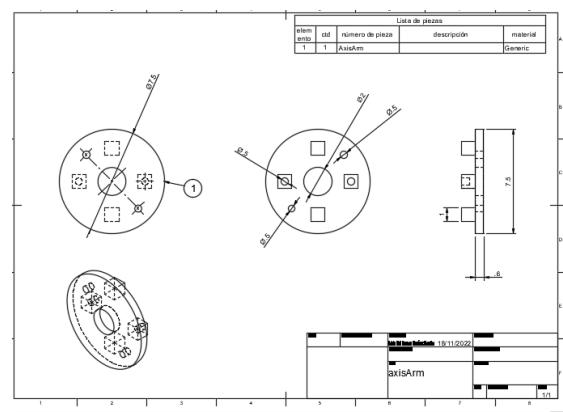


Figura 12.1.6. Axis Arm (Maricarmen Ramírez, 2022).

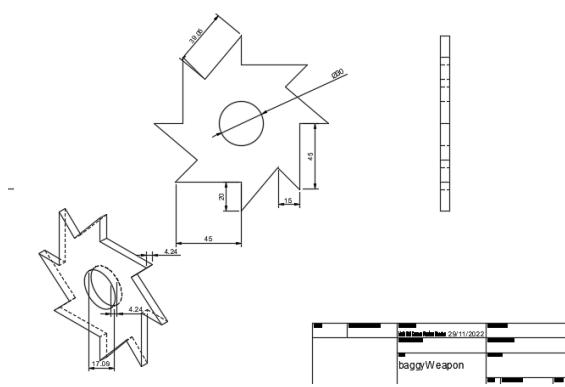
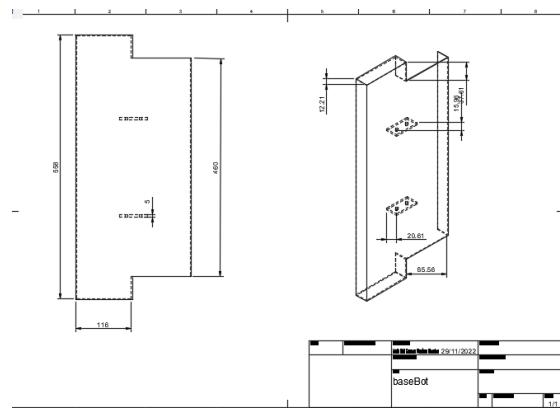
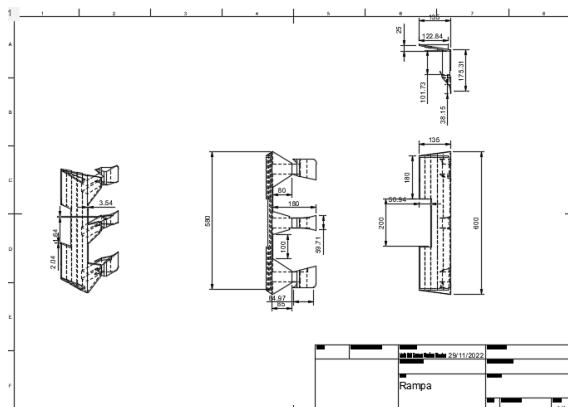


Figura 12.1.7. Arma (Maricarmen Ramírez, 2022)



*Figura 12.1.8. Base (Maricarmen Ramírez, 2022).*



*Figura 12.1.9. Rampa (Maricarmen Ramírez, 2022).*

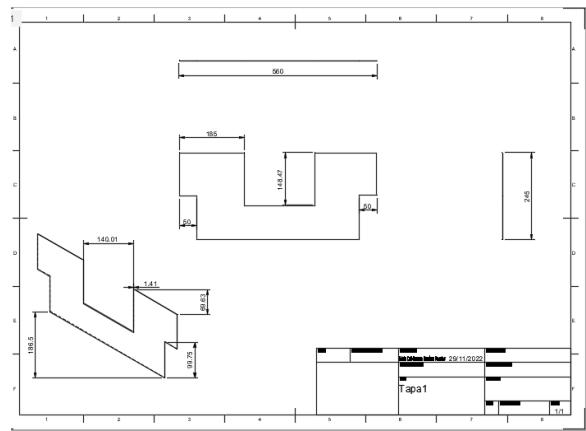


Figura 12.1.10. Tapa. (Maricarmen Ramírez, 2022).

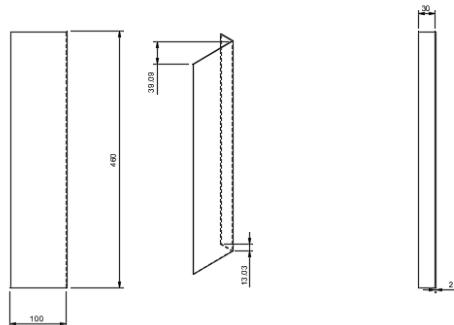


Figura 12.1.11. Wall (Maricarmen Ramírez, 2022).

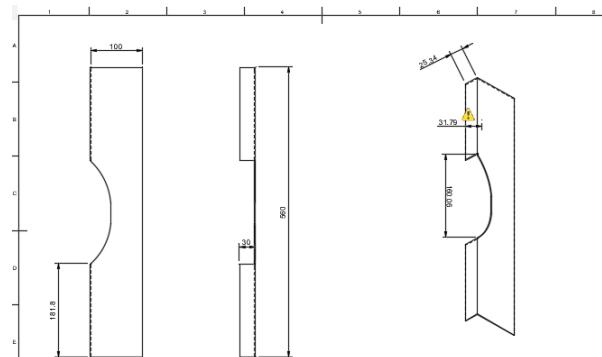


Figura 12.1.14. Wall part II (Maricarmen Ramírez, 2022).

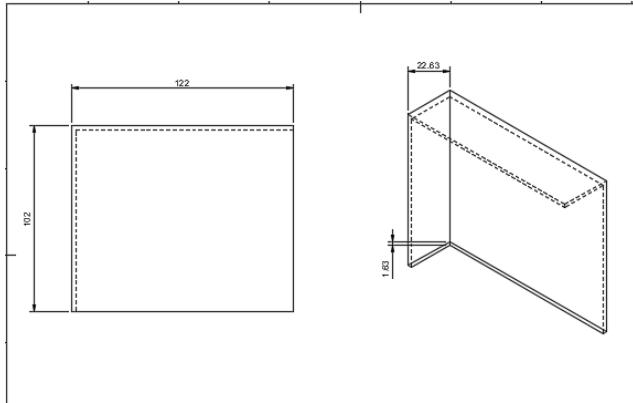


Figura 12.1.15. Wall part III (Maricarmen Ramírez, 2022).

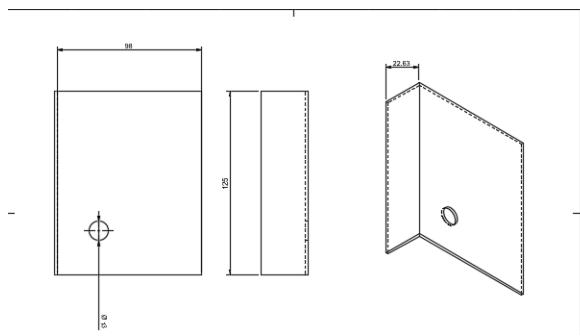


Figura 12.1.16. Wall Left (Maricarmen Ramírez, 2022)

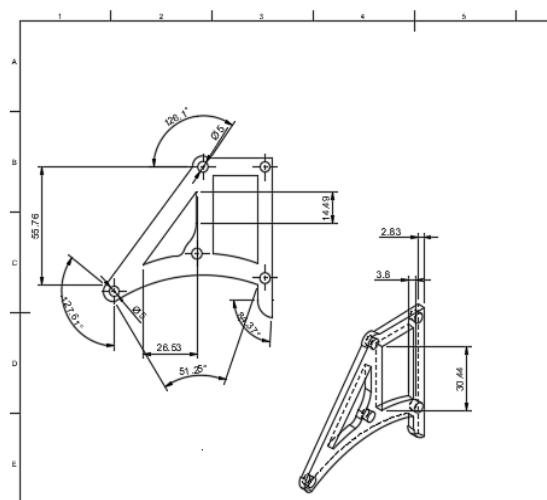


Figura 12.1.17. Weapon Base (Maricarmen Ramírez, 2022).

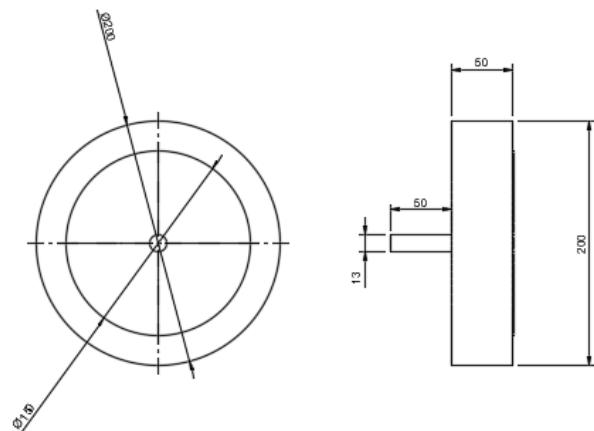


Figura 12.1.18. Llanta (Maricarmen Ramírez, 2022).

#### Listado de siglas o abreviaturas.

PCB	"Printed Circuit Board" en inglés. Se define como un circuito cuyos componentes y conductores están contenidos dentro de una estructura mecánica. Las funciones conductoras incluyen trazas de cobre, terminales, disipadores de calor o conductores planos. La estructura mecánica se hace con material laminado aislante entre capas de material conductor. A su vez, la estructura general es chapada y cubierta con una máscara de soldadura no conductora y una pantalla de impresión para la ubicación de leyenda de componentes electrónicos.
RF	“Radiofrecuencia”. Las ondas de radiofrecuencia se generan cuando una corriente alterna pasa a través de un conductor. Las ondas se caracterizan por sus frecuencias y longitudes. La frecuencia se mide en hercios (o ciclos por segundo) y la longitud de onda se mide en metros (o centímetros).

## Lista de materiales:

	Elemento	Cantidad	Coste
1	Motoreductor	1	950
2	Motor con espiral	2	1440
3	Suministrador DCDC	1	350
4	Chumaceras	2	230
5	Estrellas	5	500
6	Celdas	12	360
7	Bateria 14A picos 300A	1	830
8	Engranes	2	200
9	LM317	2	80
10	resistencias	9	9
11	Reley 10A 5V	1	44
12	CT4N25	2	26
13	Tuercas 1/2	8	64
14	tuerca 2/8	8	28
15	Varilla	1	78
16	Abrazadera motorees	3	60
17	Arandelas	6	12
18	Baleros	4	120
19	flecha 30CM	1	0
20	Lamina 3mm	1pliego	0
21	Lamina Galvanizada	1/2 pliego	0
22	Lamina acero	pedacera	0
23	Sopleteada	servicio	50
24	tornillos 1/2	4	32
25	remaches	2 paquetes	60
26	Electrodos 1013	1kg	61
27	Maquilado PCB	2 servicios	100
28	llantas	2	0
29	Lamina acero 1/4	1	0
30	Flecha 1in	1	0
31	pinturas	2 latas	120
32	perfil viga	1tramo	0
33	Madera	1tablon	50
34	madera Maquetas	1pliego 1x1m	0
35	rollo microalambre	1	375
36	BTS7960	4	860
37	acrilico	1lamina	0
38	arduino mega	2	470
39	Cable	5 mts	200
40	control RF	1	1000
41	Cadena	1metro	0
42	petroleo	1L	65
43			
44			
45			
46			
47	TOTAL:		8824

*Tabla 1.1 Lista de materiales utilizados (2022).*