

INSTITUTO SUPERIOR TECNOLÓGICO

VIDA NUEVA



CARRERA:

TECNOLOGIA EN ELECTROMECAÁNICA

TEMA DEL PROYECTO DE APLICACIÓN PRÁCTICA

DISEÑO Y CONSTRUCCIÓN DE UN ROBOT MINI SUMO RADIO
CONTROLADO UTILIZANDO UNA TARJETA MINI BLACK

AUTOR:

ACARO GONZAGA VINICIO ANDRÉS

TUTOR:

ING. TOCA TRUJILLO LUIS EDISON

FECHA:

AGOSTO 2020

QUITO – ECUADOR

CESIÓN DE DERECHOS DE AUTOR

Yo, ACARO GONZAGA VINICIO ANDRÉS portador/a de la cédula de ciudadanía 171680454-5, facultado/a de la carrera **TECNOLOGÍA ELECTROMECAÁNICA**, autor/a de esta obra certifico y proveo al Instituto Tecnológico Superior Vida Nueva, usar plenamente el contenido del informe con el tema **DISEÑO Y CONSTRUCCIÓN DE UN ROBOT MINI SUMO RADIO CONTROLADO UTILIZANDO UNA TARJETA MINI BLACK** de aportar y promover la lectura e investigación, autorizando la publicación de mi proyecto de titulación en la colección digital del repositorio institucional bajo la licencia de Creative Commons: Atribución-NoComercial-SinDerivadas.

En la ciudad de Quito, del mes de Agosto de 2020

CERTIFICACIÓN

En mi calidad de Tutor del Proyecto: “DISEÑO Y CONSTRUCCIÓN DE UN ROBOT MINI SUMO RADIO CONTROLADO UTILIZANDO UNA TARJETA MINI BLACK” en la ciudad de Quito, presentado por la ciudadano ACARO GONZAGA VINICIO ANDRÉS, para optar por el título de Tecnólogo en Electromecánica, certifico, que dicho proyecto ha sido revisado en todas sus partes y considero que reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal examinador que se designe

En la ciudad de Quito, del mes de Agosto de 2020

TUTOR:

ING. TOCA TRUJILLO LUIS EDISON

APROBACIÓN DEL JURADO EXAMINADOR O TRIBUNAL

Los miembros del jurado examinador aprueban el informe de investigación, sobre el tema: “DISEÑO Y CONSTRUCCIÓN DE UN ROBOT MINI SUMO RADIO CONTROLADO UTILIZANDO UNA TARJETA MINI BLACK” en la ciudad de Quito Del estudiante: ACARO GONZAGA VINICIO ANDRÉS de la Carrera en Tecnología en Electromecánica.

Para constancia firman:

.....

C.I.:

.....

C.I.:

.....

C.I.:

.....

C.I.:

DECLARACIÓN DE AUTENTICIDAD

Yo, ACARO GONZAGA VINICIO ANDRÉS con cedula de ciudadanía 1716804545 estudiante del Instituto Superior Tecnológico Vida Nueva, declaro que he realizado este trabajo de titulación tomando en consideración citas bibliográficas que se nombran en este texto.

El Instituto Superior Tecnológico Vida Nueva puede utilizar este trabajo de titulación como una ayuda bibliográfica.

En la ciudad de Quito, del mes de Agosto de 2020

ACARO GONZAGA VINICIO ANDRÉS

C.I.: 1716804545-5

AGRADECIMIENTO

Ante todo agradezco a Dios por la oportunidad que me ha brindado de seguir estudiando y cumplir una de muchas metas, a mi madre Alba Gonzaga , a mi Padre Andrés Acaro y a mi familia que estuvo en todo momento apoyándome, que han sido una fuente de inspiración importante durante el tiempo que estado estudiando, a mi tutor de Tesis ING. Luis Toca que con su amplio conocimiento y practica en el campo de la robótica me ha guiado en este largo proceso de la realización de mi tesis y poder culminarla con éxito, también un extenso a cada uno de los docentes de esta noble Institución como es el Instituto Superior Tecnológico Vida Nueva, quienes han compartido cada uno de sus conocimientos para poder ser un profesional a futuro.

Índice

CESIÓN DE DERECHOS DE AUTOR.....	i
CERTIFICACIÓN.....	ii
APROBACIÓN DEL JURADO EXAMINADOR O TRIBUNAL	iii
DECLARACIÓN DE AUTENTICIDAD.....	iv
AGRADECIMIENTO	v
INTRODUCCIÓN.....	x
ANTECEDENTES.....	xi
1. OBJETIVOS	1
1.1 Objetivo General	1
1.2 Objetivos específicos	1
2. DESARROLLO.....	2
2.1 MARCO TEÓRICO – CONCEPTUAL	2
2.1.1 Historia del sumo robot	2
2.1.2 Reglamento.....	3
2.1.3 Tarjeta mini black.....	6
2.1.4 Micro controlador ATMEGA328P	8
2.1.5 Pololu USB AVR Programmer v2.1	10
2.1.6 Drive L293	13
2.1.7 Radio Control 2.4 GHz.....	14
2.1.8 Batería de lipo	15
2.1.9 Arduino IDE	17
2.1.10 Inventor.....	20
2.1.11 Para que sirve	20
2.1.12 Usos y funcionalidades de Inventor	20
2.2 PROCEDIMIENTO-METODOLOGIA.....	22
2.2.1 Diseño	22
2.2.2 Construcción.....	23
2.2.3 Implementación	39
5 CONCLUSIONES:.....	42
RECOMENDACIONES:.....	43

6	REFERENCIAS:.....	44
7	ANEXOS:	46

ÍNDICE DE FIGURAS:

Figura 1: Bill Harrison Hosting	2
Figura 2: Área de combate mini sumo.	3
Figura 3: Pines de Conexión de un Atmega328P.	10
Figura 4: Esquema de los pines de conexión del AVR v2.1.	12
Figura 5: Diagrama Del L293	13
Figura 6: Radio Control 2.4 GHz.....	14
Figura 7: Batería de lipo.....	15
Figura 8: Batería de una celda a 3.7V con 2000mAH.	16
Figura 9: Logo de Arduino.	18
Figura 10: Logo de Inventor.	20
Figura 11: Pantalla de Inventor 2020	22
Figura 12: Vista isométrica modelado 2d ISO	22
<i>Figura 13: Comando línea y extruir.</i>	23
Figura 14: Estructura guía del mini sumo.....	23
Figura 15: Piezas del robot cortadas a laser	24
Figura 16: Colocación de placa mini black.....	24
Figura 17: tarjeta mini black.....	24
Figura 18: Sensor JS40F	25
Figura 19: Colocación de los motores Maxon	25
<i>Figura 20: Motores Maxon</i>	25
Figura 21: Conexión de modulo digital y vinculación a electroválvulas	26
Figura 22: Esquema de conexión de cada uno de sus componentes	27
Figura 23: Esquema eléctrico terminado.....	27
Figura 24: Página oficial de Pololu.....	28
Figura 25: Archivo de instalación Pololu.	28
Figura 26: Ventana de instalación.....	29
Figura 27: Ventana emergente de instalación.....	29
Figura 28: Ventana de instalación.....	30
Figura 29: Imagen de conexión de la tarjeta programador.	30
Figura 30: Ventana de los puertos de comunicación.....	31
Figura 31: Venta de archivos de Arduino	31
Figura 32: Ventana de preferencias.....	32
Figura 33: Ventana de gestor de tarjeta.....	32
Figura 34: Ventana de paquetes Barebones.....	32
Figura 35: Ventana de Herramientas para elegir placa nueva.	33
Figura 36: Ventana de selección de Atmega328/328P.	33
Figura 37: Ventana de selección de Clock External20 MHZ	34
Figura 38: Ventana de administrador de dispositivos.....	34
Figura 39: Ventana de COM programador AVR.....	35
Figura 40: Ventana del programador Atmel STK500.....	35

Figura 41: Librería EnableInterrupt.h.	36
Figura 42: Constantes de los pines de conexión de los motores.	36
Figura 43: Programación del Mando RC.	36
Figura 44: Programación de mando Pitch.	37
Figura 45: Configuración de condiciones iniciales.	37
Figura 46: Programación del RC para el mando.	38
Figura 47: Función para manejar los motores con PWM y las señales del driver.	38
Figura 48: Perforación para cilindros	39
Figura 49: Cableado de electroválvulas y mangueras PU.....	40
Figura 50: Conexión entre el receptor y la tarjeta mini black.....	41
Figura 51: Configuración AVR Programmer.....	41
Figura 52: Datafish Atmega 328P	46
Figura 53: Conexión de la batería lipo en la placa mini black.....	46
Figura 54: Ficha técnica de la placa	47
Figura 55: Conexión entre la placa y el conector del RC	48
Figura 56: Robot mini sumo	48

INTRODUCCIÓN

La modalidad de robot mini sumo fue creada en los años ochenta en Japón por Hiroshi Nozawa, la idea fue en incentivar a estudiantes interesados en carreras de robótica o similares. Esto género que muchas Universidades e Institutos diseñen sus propios prototipos.

A partir de los años noventa fue introducido en estados Unidos la categoría de robots mini sumo lo que genero mucho interés a la comunidad tecnológica esto ha dado una pauta para la creación de nuevos sistemas para el control de robot mini sumo.

El combate de robots Mega, Mini y Micro sumos consiste de tres asaltos con una duración máxima de un minuto cada uno exceptuando el tiempo de seguridad, con un tiempo de descanso máximo de un minuto entre cada asalto. De acuerdo con las llaves establecidas cada contendiente será llamado al Dhoyo para su respectivo enfrentamiento, si transcurridos cinco minutos no se presenta el equipo quedará automáticamente eliminado. El jurado calificador observará que los robots cumplan con las especificaciones del presente reglamento, además se observará la funcionalidad del mismo.

El mini sumo RC fue diseñado para el club de robótica del INSTITUTO SUPERIOR TECNOLOGICO VIDA NUEVA de manera de incentivar a la carrera de electromecánica a generar nuevos prototipos. La misión de este prototipo como su nombre lo indica es de luchar con otros robots parecidos en su peso, dimensión, diseño para lo cual se utiliza un radio control, una placa mini black, motores y un drive que permite el control de los motores la cual será manejado desde un radio control RC que se programa a través del software Arduino IDE.

ANTECEDENTES

La modalidad de Sumo Robótico según Guillermo B. (2012), fue inventada en Japón a finales de los años ochenta por Hiroshi Nozawa, presidente de Fuji Software. Su objetivo era que los estudiantes se interesen en la robótica. El primer torneo de exhibición se realizó en 1989 y participaron 33 robots. El primer campeonato oficial fue en 1990 y lucharon 147 robots en él. Desde entonces el crecimiento del sumo robótico en Japón ha sido imparable. Más de 4000 robots participaron en la Liga de 2001.

Alfredo Piña (2012). dice que a comienzos de los años noventa el sumo robótico fue introducido en Estados Unidos por Mato Hattori. Hattori grabó una cinta con los mejores momentos de la tercera Liga de sumo. En este caso el robot es capaz de modificar su actuación en tiempo real en función de lo que ocurra en su entorno. Harrison posteriormente fue el que inventó la categoría de minisumo. Esta categoría es la que se ha hecho más popular ya que existen al menos el doble de robots de minisumo que de sumo estándar ya que su reducido tamaño facilita la construcción y ahorra costes.

Según Vanesa A (2014), La primera generación en 1960 se construye robots industriales que realizan tareas repetitivas. Luego de veinte años llega la segunda generación de robots dotados de sensores capaces de interactuar con el entorno. Posteriormente llega la última generación a partir de 1985 hasta la actualidad en la cual se construyen robots móviles. Según la clasificación de los robots por generación, la Inteligencia Artificial es el futuro próximo. Las compañías más grandes del mundo están invirtiendo en la construcción de este tipo de robot. Su objetivo es tratar de liderar el sector y poder atribuirse el descubrimiento de tecnologías de última generación esto implica al gran desarrollo de la inteligencia artificial.

Según CER (2019), los torneos nacionales de robótica se realizan cada año los torneos en Ecuador, donde se llevarán a cabo en diferentes provincias como en el caso de Concurso Ecuatoriano de Robótica (CER), entre otras, que se enfrentarán a diferentes instituciones, clubes de robótica, universidades e

identidades dedicadas al desarrollo de robots de competencias que buscan obtener los primeros lugares.

1. OBJETIVOS

1.1 Objetivo General

Diseñar y construir un robot mini sumo radio controlado utilizando una tarjeta mini black para competencias de robótica.

1.2 Objetivos específicos

- Diseñar el chasis del robot mini sumo utilizando AutoCAD.
- Ensamblar e identificar los pines de la tarjeta mini black para la programación.
- Programar el radio control a través de la tarjeta mini black.
- Realizar pruebas de funcionamiento para descartar errores y realizar correcciones en la programación.

2. DESARROLLO

2.1 MARCO TEÓRICO – CONCEPTUAL

2.1.1 Historia del sumo robot

Según Guillermo B. (2012), la modalidad de sumo robot fue inventada en Japón a finales de los ochenta por Hiroshi Nozawa, presidente de Fuji software. Su objetivo era que los estudiantes se interesaran por la robótica.

Su primer torneo de exhibición se realizó en 1989 y participaron alrededor de 33 robots. El primer campeonato oficial fue en 1990 y lucharon 147 robots en él.

A comienzos de los años noventa el sumo robótico fue introducido en Estados Unidos por Mato Hattori. Hattori grabó una cinta con los mejores momentos de la tercera liga de sumo. Esta cinta llegó a manos de Bill Harrison que se involucró para difundirlo en su país. Harrison posteriormente fue el que inventó la clase de mini sumo. Esta clase es la que ha tenido mayor acogida ya que existen al menos el doble de robots de mini sumo que de sumo estándar ya que su reducido tamaño facilita la construcción y ahorra costos.



*Figura 1: Bill Harrison Hosting
Fuente: CIRC Robot Competition 2000_Sumo*

2.1.2 Reglamento

En este tema se hablará de las reglas más importantes del sumo y mini sumo.

En el área de combate robot mini sumo será de forma circular y superficie de madera, tendrá un diámetro de 75cm distribuidos de la siguiente forma.

- Un círculo interno con diámetro de 70cm de color negro mate.
- Un círculo (borde) de color blanco de 2,5cm.
- Con una altura de 5cm del piso.

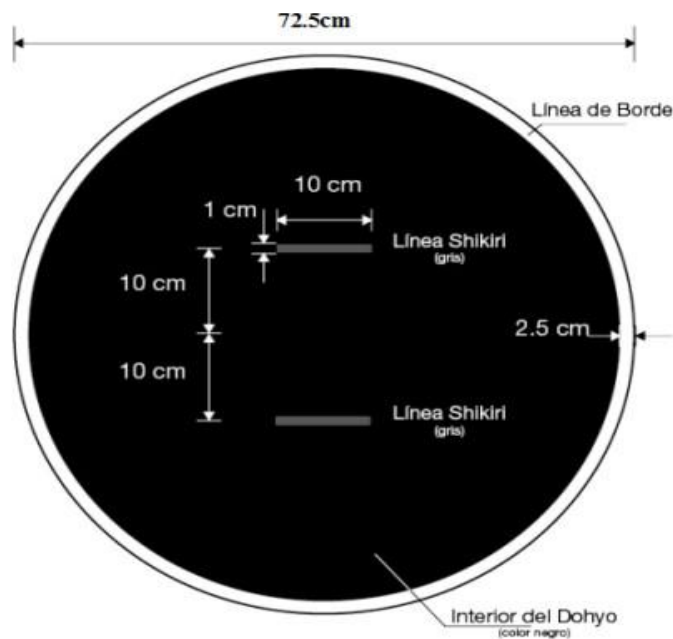


Figura 2: Área de combate mini sumo.

Fuente: Reglamento para la categoría de robot mini-sumo. Recuperado de <http://www.itc.edu.co/archives/reglamentominisumo.pdf>

Según CER (2019). La categoría Jr. está abierta únicamente en Minisumo autónomo. Esta categoría está sujeta a todas las características físicas de la categoría Mini sumo, con el limitante de edad para participantes menores a 17 años. La categoría RC está abierta para Mega y Mini sumo. La Tabla 1 muestra el detalle de categorías abiertas.

Tabla 2: Categoría habilitadas

Categorías	Autónomo	RC	Jr.
Mega sumo	x	x	x
Mini sumo	x	x	
Micro sumo	x		

Fuente: Elaboración propia (2020).

En el preámbulo de la competencia los robots deberán ser sometidos a verificación, si el jurado calificador observara el incumplimiento de alguno de los puntos detallados se proporcionará un lapso de 5 minutos para realizar las debidas correcciones, si el incumplimiento persiste el robot será descalificado.

La arquitectura física y el funcionamiento del robot deberán cumplir con los siguientes puntos como se observa en la tabla II:

Tabla 2: Dimensiones y pesos de los sumos.

Categorías	Ancho (cm)	Largo (cm)	Alto (cm)	Peso (gr)
Mega sumo	20	20	libre	3000
Minisumo	10	10	libre	500
Micro sumo	5	5	5	100

Fuente: Elaboración propia (2020).

Alimentación

Según ESPE (2019). la fuente de energía para alimentar los circuitos eléctricos de control y los motores del robot será interna, considerándose cualquier tipo de baterías de corriente continua. La duración de las baterías debe ser suficiente para desarrollar perfectamente un combate completo.

Control:

- **Sumo autónomo:** El robot sumo deberá funcionar de manera autónoma, quedando prohibido el uso de cualquier tipo de radio control o controles que afecten la funcionalidad del oponente. Solo existen mandos

inalámbricos “ON/OFF”, este tipo de control es para las categorías: Mega, Mini y Micro sumo.

- **Sumo RC:** Se pueden controlar a distancia, controladores 75Mhz están específicamente prohibidos. Todos los controles remotos deben ser pares digitalmente no acoplados. Robots de control remoto comienzan en una señal del juez de pista del torneo. Todo movimiento del robot debe ser directamente debido a la tele operación del operador.

Garantías de seguridad

Los robots Mega, Mini y Micro sumos según CER (20019) deberán ser construido bajo ciertas normas y requisitos de tal manera que brinde las garantías de seguridad para los competidores, el jurado, el público en general y el espacio físico donde se desarrolla la competencia. Entendiéndose como requisitos de seguridad los siguientes:

- El robot deberá poseer una llave “ON/OFF” externa o pulsador de emergencia visible y de fácil acceso que lleve a condiciones de paro total al sistema o mediante un mando inalámbrico, solo para la categoría Sumo autónomo es obligatorio el uso de un apagado remoto, debido a la peligrosidad que representa si uno de estos robots se sale de control.
- El prototipo debe ser diseñado para entrar en acción 5 segundos después de que el concursante presione un interruptor de inicio, bajo la indicación del árbitro de la competencia, este intervalo es el llamado Tiempo de Seguridad.
- El microprocesador y/o tarjeta de desarrollo (Arduino, Baby orangután, entre otros.) del robot podrá ser de cualquier tipo de fabricante, y se podrá usar cualquier tamaño de memoria.

DESARROLLO DE LA COMPETENCIA

- Cada combate de robots Mega, Mini y Micro sumos consiste de tres asaltos con una duración máxima de un minuto cada uno exceptuando el Tiempo de Seguridad, con un tiempo de descanso máximo de un minuto entre cada asalto. De acuerdo con las llaves establecidas cada contendiente será llamado al Dhoyo para su respectivo enfrentamiento, si transcurridos cinco minutos no se presenta el equipo quedará automáticamente eliminado.
- El jurado calificador observará que los robots cumplan con las especificaciones que se establecen en la tabla 3 del presente reglamento, además se observará la funcionalidad del mismo.

La CER (2019) dice que las dimensiones del área se encuentran especificadas en la Tabla 3 considerando las características específicas de combate para cada “Dhoyo” o área de combate según los robots Mega sumos, Mini sumos, Micro sumos y Nano sumos:

Tabla 3: Diseño del Dhoyo para cada categoría

Categorías	Diámetro (cm)	Altura (cm)	Ancho de borde (cm)	Material (gr)
Mega sumo	154,00	5,00	5,00	Metal
Mini sumo	77,00	2,50	2,50	Madera
Micro sumo	38,50	1,25	1,25	Madera

Fuente: Reglamento CER recuperado de CER

2.1.3 Tarjeta mini black

La tarjeta Mini Black es una placa diseñada principalmente para robots mini sumos de competencia. Es la tarjeta utilizada por el robot mini sumo, además de contar con una arquitectura AVR de 8 bits, para procesos de alto rendimiento.

Características:

- Micro controlador: ATMEGA328P.
- Flash: 32K Bytes.
- EEPROM: 1K Byte.
- RAM: 2K Bytes.
- Rendimiento: alto rendimiento/baja potencia.
- Alimentación de Entrada: (7 – 16) V.
- Drivers: 5A por canal (2 canales) – (2 motores).
- Pulsadores: 2.
- Leds: 3 (Leds de Usuario recomendados para visualizar estado de los sensores).
- Pines: 7 E / S digitales/Análogas (sensores).
- Regulador: Regulador de tipo Stop Down para inmunidad a caídas de tensión.
- Asociable a Arduino.
- Conector ISP de 6 pines.
- Cuenta con protecciones para cortocircuito y corriente inversa (únicamente para el micro controlador, el driver de los motores no cuenta con esta protección, por lo que se debe ser cuidadoso y conectar correctamente).

Aplicación:

- Seguidores de línea.
- Procesos de alto rendimiento.
- mini sumos.

2.1.4 Micro controlador ATMEGA328P

Según Mario Tomas (2008) los Micro controlador ATmega328P en SMD son de alto rendimiento y de bajo consumo por lo tanto el AVR micro controlador 8-Bit, es una versión mejorada del ATmega328P

Cuenta con 2 módulos de comunicación UART, 2 timer de 8 bits, 3 timer de 16 bits, 131 Instrucciones de gran alcance, hasta 20 MIPS Throughput a 20MHz. cuenta con 32 pines físicos en el encapsulado TQFP.

Según Alberto García (2020). El bootloader es parte de Arduino. Es un pequeño código almacenado en su memoria que se ejecuta al arrancar y que se encarga de leer datos por su puerto serie y almacenarlo en la memoria flash. Es decir, que permite cargar nuestro propio código sin necesidad de usar un programador externo. Los Atmega328P se pueden adquirir a muy bajo precio, pero en la mayoría de los casos vendrán sin su correspondiente bootloader (a veces incluso aunque el vendedor diga que sí). Así que, si no lo trae, lo primero que se tiene que hacer es cargar el bootloader en el microcontrolador como se observa en la figura 3.

Características:

- Fabricante: MICROCHIP (ATMEL)
- Tipo de circuito integrado: micro controlador AVR
- Capacidad de memoria EEPROM: 1kB
- Carcasa: TQFP32
- Capacidad de memoria SRAM: 2kB
- Capacidad de memoria Flash: 32kB
- Frecuencia de tictado: 20MHz
- Cantidad de canales PWM: 10
- Montaje: SMD
- Cantidad de timers 8bit: 2

- Cantidad de timers 16bit: 3
- Temperatura de trabajo: -40...85°C
- Número de cortes externos: 27
- Número de canales output compare: 10
- Cantidad de comparadores: 1
- Número de canales input capture: 3
- Cantidad de transductores A/D 10bit: 8
- Familia: ATmega
- Número de canales de capacidad: 12
- Tensión de alimentación: 1.8...5.5V
- cantidad de pines: 32

Interfaz:

- DebugWIRE
- I2Cx2
- UARTx2

Aplicación:

- Programación
- Tarjetas de desarrollo

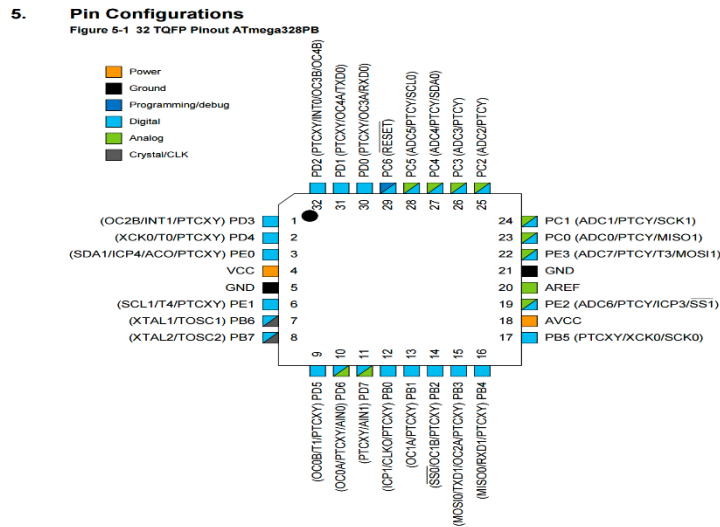


Figura 3: Pines de Conexión de un Atmega328P.

Fuente: Atmel atmega 328P Recuperado de <http://ww1.microchip.com/downloads/en/DeviceDoc/40001906C.pdf>

2.1.5 Pololu USB AVR Programmer v2.1

Según Pololu Corporation (2020) dice que el programador integrado en el sistema se puede usar para programar microcontroladores AVR y placas de control basadas en AVR, como nuestro A-Star 328PB Micro, controladores de robot Orangután y el robot 3pi. El programador emula un STK500 en un puerto serie virtual, lo que lo hace compatible con el software de programación AVR estándar, y es compatible con dispositivos que funcionan a 3,3 V o 5 V. El programador también cuenta con un puerto serie de nivel TTL, lo que permite la comunicación en serie de uso general para depurar o programar microcontroladores con un cargador de arranque en serie. Este programador completamente ensamblado incluye un cable ISP y un encabezado macho de doble cara de 1 x 6.

El programador proporciona una interfaz para transferir un programa AVR compilado de su computadora a la memoria no volátil del AVR objetivo, lo que le permite ejecutar el programa.

El programador se comunica con su software de programación, como Atmel Studio, AVRDUDE o Arduino IDE, a través de un puerto COM virtual usando el

protocolo STK500. El programador se conecta a su dispositivo de destino a través de un cable de programación ISP de 6 pines incluido (las conexiones ISP de 10 pines más antiguas no son compatibles directamente, pero es fácil crear o comprar un adaptador ISP de 6 pines a 10 pines). El programador también actúa como un adaptador en serie de USB a TTL, proporcionando un puerto en serie de nivel TTL que se puede utilizar para comunicarse con otros dispositivos en serie desde su computadora. Se envía completamente ensamblado con conectores soldados como se muestra.

Características:

- Se conecta a una computadora a través de USB a través de un cable USB A Micro-B (no incluido)
- Emula un programador STK500 a través de la interfaz de puerto COM virtual
- Funciona con el software de programación AVR estándar, incluidos Atmel Studio, AVRDUDE y Arduino IDE
- Software de configuración disponible para Windows, Mac OS X y Linux
- Admite dispositivos de 3,3 V y 5 V; puede cambiar automáticamente el voltaje de funcionamiento según el VCC objetivo detectado
- Opcionalmente, puede alimentar el objetivo a 3.3 V o 5 V en sistemas de baja potencia
- Funcionalidad de adaptador serial USB a TTL para comunicación serial de uso general
- Proporciona una salida de reloj de 100 kHz, que puede ser útil para revivir AVR mal configurados
- Todos los pines de E / S están protegidos con resistencias de 470 Ω
- Cable ISP de 6 pines y un conector macho de doble cara 1 x 6 incluido

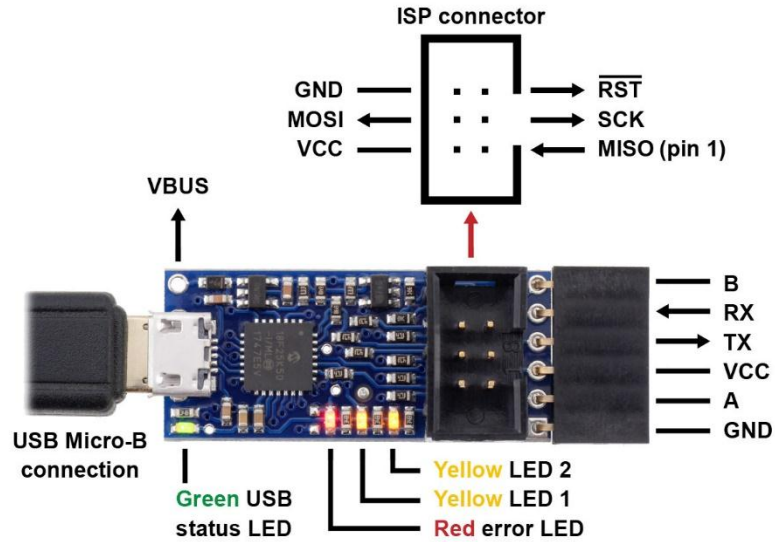


Figura 4: Esquema de los pines de conexión del AVR v2.1.
Fuente: Pololu AVR (BITS) Recuperado de <https://www.pololu.com/product/3172>

Características del puerto de serie de nivel TTI

Este programador funciona como un adaptador USB a serie. Para Pololu Corporation (2020) dice que el programador se instala como dos puertos COM virtuales: uno para comunicarse con el software de programación y otro para comunicaciones seriales de uso general. Esto significa que puede cambiar sin problemas entre programar un AVR y depurarlo a través del puerto serie TTL sin tener que abrir y cerrar su programa de terminal. Además de las líneas de transmisión en serie (TX) y de recepción (RX), el programador le permite usar A y B como líneas de comunicación en serie que se pueden configurar con nuestro software.

En la configuración predeterminada, el pin B se usa como salida DTR (y el pin A es una entrada no utilizada), lo que le permite conectar el programador directamente a una variedad de placas compatibles con Arduino (incluido nuestro Micro A-Star 328PB) y usar para cargar programas a la placa a través de un cargador de arranque en serie.

2.1.6 Drive L293

Jesús (2017) dice que el L293D es un driver de 4 canales capaz de proporcionar una corriente de salida de hasta 600mA por canal y puede soportar picos de hasta 1.2 A. Cada canal es controlado por señales TTL y cada pareja de canales dispone de una señal de habilitación para conectar o desconecta las salidas.

Tiene la disponibilidad de poder utilizar dos tensiones diferentes, una para el propio circuito integrado y otra para la alimentación del motor, cosa que facilita. El dispositivo es un controlador monolítico integrado

El dispositivo es un controlador monolítico integrado de cuatro canales de alto voltaje y alta corriente diseñado para que acepte niveles lógicos de estándares DTL o TTL y unidades de cargas inductivas (como relés solenoides, DC y motores paso a paso) y conmutadores como se observa en la figura 5.

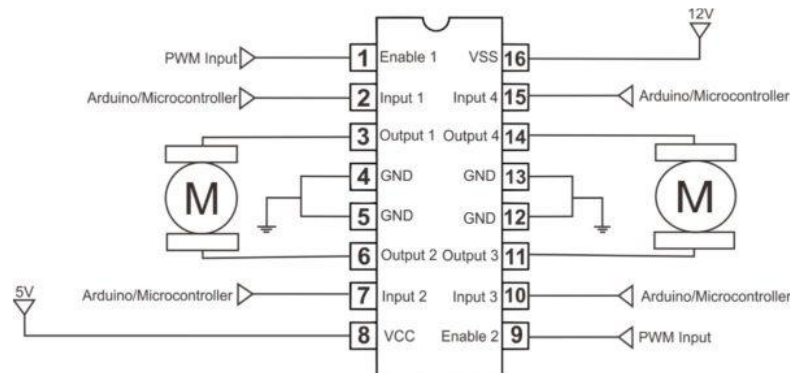


Figura 5: Diagrama Del L293
Fuente: Datasheet catalogo Recuperado de

http://www.datasheetcatalog.com/info_redirect/datasheet/stmicroelectronics/1330.pdf.shtml

Característica:

- Capacidad de salida por canal (600 mA).
- Salida de corriente pico No repetitiva por canal (1.2 A).
- Protección a altas temperaturas.
- Lógica "0" tensión de entrada hasta 1.5 V (INMUNIDAD DE ALTO RUIDO).
- Contiene Diodos de pinza interna.

- Facilidad de conexión.
- Capacidad de motores (2).

Aplicaciones:

- Seguidores de línea.
- Mini sumo.
- Control de motores de velocidad.

2.1.7 Radio Control 2.4 GHz

Radio Control tipo pistola de tres canales. Ideal para las categorías Minisumo RC, Sumo RC, Robot de Batalla (Warbot). Incluye el receptor.

Características:

- Número de canales: 3
- Alimentación: 4 Baterías AA (No incluidas)
- Frecuencia: 2.4 GHz

Aplicación:

- Minisumo
- Sumo
- Robots de guerra



Figura 6: Radio Control 2.4 GHz.

Fuente: ja-bot Recuperado de <https://ja-bots.com/producto/radio-control-2-4ghz/>

2.1.8 Batería de lipo

La batería LiPo es una batería de almacenamiento y suministro de energía eléctrica para proyectos de electrónica, robótica, Drones, sistemas embebidos y artículos de hobby, por mencionar algunos. Tiene la capacidad de almacenar altas densidades de energía gracias a su componente químico, el Polímero de iones de Litio o comúnmente conocido como Polímero de Litio. De este componente químico se derivan las contracciones Li-Poli, LiPo, LiP, Li-Pol, entre otros.

La batería LiPo tiene amplia densidad de energía, lo que le permite mantener mayor energía durante más tiempo en comparación con otros elementos químicos que conforman a otro tipo de baterías. Con una batería LiPo se va obtener más energía que una batería convencional del mismo peso.



Figura 7: Batería de lipo.

Fuente: Teslabem Recuperado de <https://teslabem.com/blog/como-usar-y-cuidar-las-baterias-lipo/>.

Número de celdas y configuración de una batería LiPo.

Una batería LiPo está conformada por una o varias celdas rectangulares, las cuales están interconectadas entre sí para formar la batería completa. Una celda también puede ser una batería por sí sola. Cada celda tiene un voltaje de 3.7V. En seguida un ejemplo de una única celda de LiPo.



Figura 8: Batería de una celda a 3.7V con 2000mAH.

Fuente: Teslabem Recuperado de <https://teslabem.com/blog/como-usar-y-cuidar-las-baterias-lipo/>.

El número de celdas

Si se conectan más celdas en serie se obtendrá entonces mayor voltaje. Es por eso que se puede encontrar baterías LiPo de 3.7v, 7.4v, 11.1v, 14.8v y así sucesivamente. Colocando más celdas en serie los voltajes individuales de cada celda se suman.

Ejemplo:

- Batería de 1 Celda = 3.7V.
- Batería de 2 Celdas = $3.7V + 3.7V = 7.4V$.
- Batería de 3 Celdas = $3.7V + 3.7V + 3.7V = 11.1V$.
- Batería de 4 Celdas = $3.7V + 3.7V + 3.7V + 3.7V = 14.8V$.

Voltaje ideal de una batería LiPo

El rango de voltaje ideal de una batería LiPo es:

El voltaje mínimo en una celda debe ser de 3.0V

El voltaje máximo en una celda de 4.2V

Es importante cuidar estos rangos de voltaje en las celdas de tus baterías. Si lo haces, te va a permitir darle un tiempo de vida útil mucho más duradero.

También por eso es que se recomienda siempre utilizar cargadores especializados en la carga de baterías de Polímero de Litio. Existen algunos

accesorios que son de gran ayuda para lograr que nuestra batería esté siempre bien cargada y balanceada aún y cuando se está utilizando

Ventajas sobre las baterías Lipo tradicionales

- La densidad de potencia alcanza los 7,5 kw / kg.
- Menor caída del voltaje durante la descarga de alta velocidad, lo que da más potencia bajo carga.
- La impedancia interna puede alcanzar tan solo 1.2mOhm en comparación con la de 3mOhm de un Lipoly estándar.
- Mayor control térmico, el paquete por lo general no supera los 60 ° C
- La hinchazón durante una carga pesada no supera el 5%, en comparación con el 15% de un Lipoly normal.
- Mayor capacidad durante altas descargas. Más del 90% a una tasa del 100%.
- Capacidad de carga rápida, hasta 30 °C en algunas baterías.

Vida útil más larga, casi el doble que la tecnología lipo estándar

2.1.9 Arduino IDE

Según Yübal (2018) dice que Arduino es una plataforma de creación electrónica de código abierto, la cual está basada en hardware y software libre y flexible y fácil de utilizar. Esta plataforma ayuda a crear diferentes tipos de microordenadores de una sola placa a los que la comunidad de creadores que pueden darle diferentes usos. El hardware libre son los dispositivos cuyas especificaciones y diagramas son de acceso público, de manera que cualquiera puede replicarlos. Esto quiere decir que Arduino ofrece las bases para que cualquier otra persona o empresa pueda crear sus propias placas, pudiendo ser diferentes entre ellas, pero igualmente funcionales al partir de la misma base.

El software libre son los programas informáticos cuyo código es accesible por cualquiera para que quien quiera pueda utilizarlo y modificarlo. Arduino ofrece la

plataforma Arduino IDE (Entorno de Desarrollo Integrado), de tal manera se le puede dar todo tipo de utilidades.



Figura 9: Logo de Arduino.

Fuente: Hacedores recuperado de <https://hacedores.com/arduino-lanza-logo-open-source-para-comunidades/>

Como funciona Arduino

Para Santillana (2017) dice que se trata de una plataforma de hardware libre. Cuenta con un circuito integrado mediante el cual se pueden grabar instrucciones. A su vez, estas instrucciones se escriben usando un lenguaje de programación que permite al usuario establecer programas que interactúan con circuitos electrónicos.

Cuenta con una interfaz de entrada. Esta puede estar directamente unida a los periféricos, o conectarse a ellos a través de puertos.

La interfaz de entrada tiene como objetivo trasladar la información al microcontrolador. El microcontrolador es la pieza que se encarga de procesar esos datos. Además, varía dependiendo de las necesidades del proyecto en el que se desee usar la placa, y existe una gran variedad de fabricantes y versiones disponibles.

También cuenta con interfaz de salida. Este se encarga de llevar la información procesada a los periféricos autorizados de hacer el uso final de esos datos. En algunos casos puede tratarse de otra placa en la que se centraliza y procesa la

información de forma totalmente renovada, o sencillamente, puede ser una pantalla o un altavoz encargado de mostrar la versión final de los datos.

Lenguaje de la programación con Arduino: C++

¿Qué lenguaje utiliza este tipo de programación? La plataforma Arduino se programa con un lenguaje propio basado en el lenguaje de programación de alto nivel Processing, lo que significa que es similar a C++.

Para que sirve Arduino

Arduino es utilizado para miles de proyectos electrónicos y de aplicaciones diferentes para Arduino CC (2020) dice que el software Arduino es fácil de usar y se puede ejecutar en MAC, Windows y Linux. Arduino se puede utilizar para crear elementos autónomos, conectándose a dispositivos que pueden interactuar tanto con el hardware como con el software. Arduino puede controlar luces y dispositivos o cualquier otro tipo de dispositivo optando a dar solución basada a Arduino.

Por qué usar Arduino

Para Frank Mecafenix (2017) dice que existen muchas plataformas microcontroladoras disponibles para computación física. Parallax Basic Stamp, Netmedia's Bx-24 y muchas otras de funcionalidad similar pero todas estas herramientas toman de forma desordenada los detalles de la programación de microcontrolador y la encierra en un paquete más fácil de usar.

Arduino también ayuda a simplificar los procesos de trabajo y ofrece más ventajas a los profesores, estudiantes y aficionados.

Ventajas

- Multiplataforma
- Entorno de programación simple y claro.
- Código abierto y software extensible.
- Código abierto y hardware extensible.
- Barato.

2.1.10 Inventor

Para Carlos Contreras (2018) inventor es un programa de dibujo asistido en computadora CAD 2 y 3 dimensiones, puedes crear dibujos o planos genéricos, proyectos de ingeniería, arquitectura, mapas o sistemas de información geográfica por mencionar algunas industrias y aplicaciones.



Figura 10: Logo de Inventor.

Fuente: Autodesk Recuperado de https://latinoamerica.autodesk.com/?mktvar002=afc_latam_deeplink&AID=13955714&PID=8299320&SID=jkp_Cj0KCQiAkK nyBR DwA RIs ALtx e7j XSjviL1 e m Ur Z

2.1.11 Para que sirve

Se trata de un programa diseñado específicamente para la creación de planos. Sin embargo, gracias a sus modernas versiones, inventor incluye concepto de espacio papel y espacio modelo.

Además, se ha convertido también en un programa de referencia en materia de diseño gráfico, diseño gráfico e interiorismo y decoración de interiores.

2.1.12 Usos y funcionalidades de Inventor

Funcionalidades que hace que Inventor sea un programa favorito para los diseñadores, arquitectos e ingenieros entre otros.

- Funcionalidades de texto, columnas y contornos.
- Acceso a comandos y menús contextuales, lo que permite tener un ritmo de trabajo más ágil y fluido.
- Rápida recuperación de vistas al permitir guardarlas por nombre.
- Extracción de vistas de un sólido tridimensional rápido.
- Proyección real de vistas creadas anteriormente.
- Para ingenieros, la orden propis permite devolver las propiedades físicas de sólidos 3D y regiones.
- Transparencia de objetos particulares o de capas contenedoras.
- Filtrar objetos según características y propiedades.
- Posibilidad de recuperar proyectos y archivos en múltiples dispositivos gracias a AutoCAD 360.

Ventajas:

- Se relaciona con múltiples plataformas.
- Es posible dibujar en 3D y 2D.
- Es factible práctico para trabajar.
- Menor tiempo de trabajo.
- Es utilizado en muchas ramas
- Permite intercambiar información.

2.2 PROCEDIMIENTO-METODOLOGIA

2.2.1 Diseño

2.2.1.1 Diseño en Inventor

1. El programa utilizado para el diseño de la estructura del robot mini sumo es en Autodesk Inventor 2020. El cual es un software que permite realizar el modelado en 3d y 2d, además, no da una vista del plano para realizar el objeto de mejor manera, como se muestra en la figura 11.

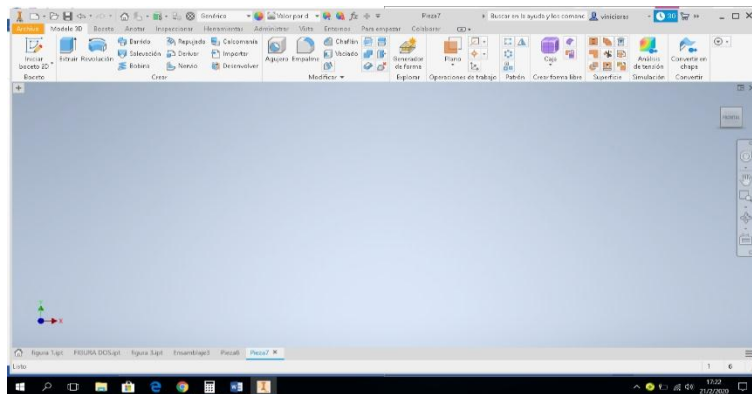


Figura 11: Pantalla de Inventor 2020
Fuente: Elaboración propia (2020)

2. Seleccionar la opción vista de modelo personalizado en 2d en la cual se tendrá una mejor visión para el diseño tanto en el plano del eje x, y, z, como muestra en la figura 12.

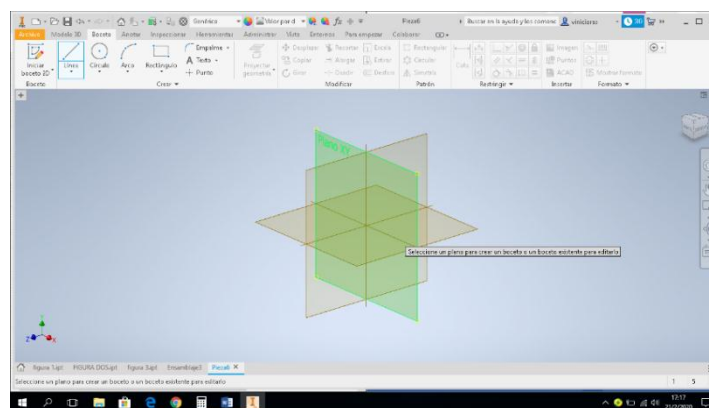


Figura 12: Vista isométrica modelado 2d ISO
Fuente: Elaboración propia (2020)

3. Utilizar los comandos línea, extruir, que se muestran en la barra de herramientas esto facilitara que el diseño del chasis como se muestra en la figura 13.

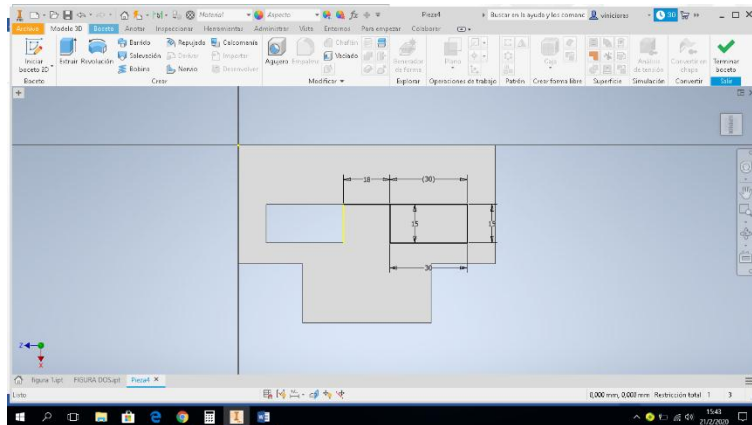


Figura 13: Comando línea y extruir.

Fuente: Elaboración propia (2020)

4. Observar el diseño final de la estructura del mini sumo y realizar correcciones en caso de ser necesario a la medida de la figura de ensamble, como muestra en la figura 14.

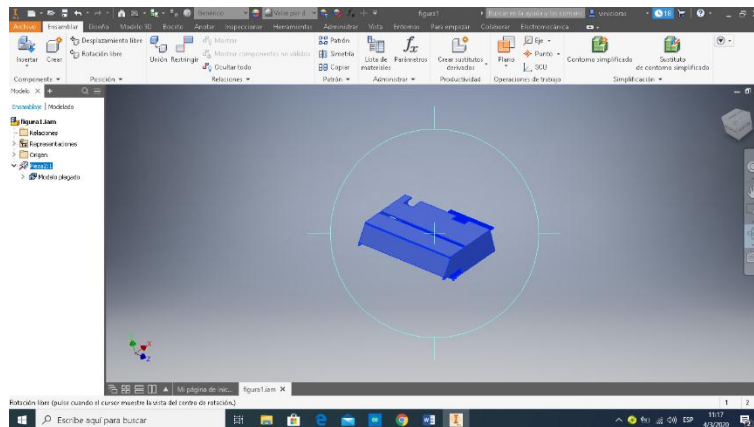


Figura 14: Estructura guía del mini sumo.

Fuente: Elaboración propia (2020)

2.2.2 Construcción

2.2.2.1.1 Ensamble

1. Una vez elaborada las piezas correspondientes se procede al ensamble del robot mini sumo como se observa en la figura 15.



Figura 15: Piezas del robot cortadas a laser
 Fuente: Elaboración propia (2020)

2. Coloca la placa mini black y los sensores Sharp en cada lugar en la base del chasis del robot mini sumo asegurando para que no se caiga la placa, como muestra en la figura 16.



Figura 16: Colocación de placa mini black.
 Fuente: Elaboración propia (2019)

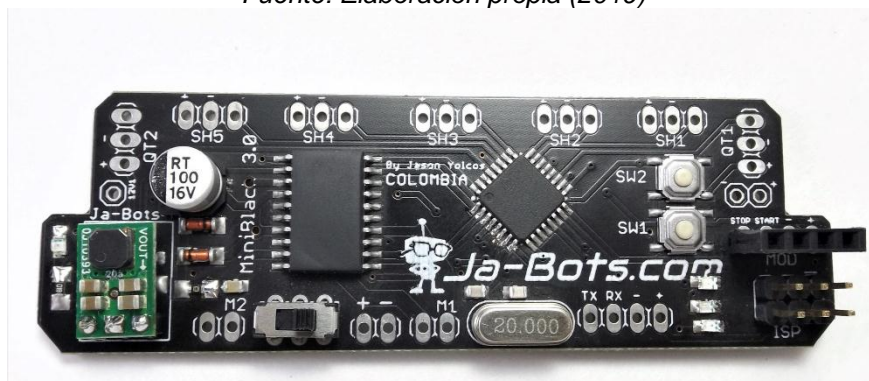


Figura 17: tarjeta mini black.
 Fuente: Elaboración propia (2020)



Figura 18: Sensor JS40F
Fuente: Elaboración propia (2020)

3. Se coloca los motores uno en cada lado en el rectángulo hueco para su fijación como muestra en la figura 19.



Figura 19: Colocación de los motores Maxon
Fuente: Elaboración propia (2020)



Figura 20: Motores Maxon
Fuente: Elaboración propia (2020)

4. Se fija los demás componentes del chasis en sus respectivos sitios, como muestra en la figura 21.



*Figura 21: Conexión de modulo digital y vinculación a electroválvulas
Fuente: Elaboración propia (2020)*

2.2.2.1.2 Tarjeta mini black:

La tarjeta mini black es diseñada para robots mini sumo de competencia, la cual es muy útil. Tiene una arquitectura AVR de 8 bits, para procesos de alto rendimiento y cuenta con un diseño que se adapta a todas las necesidades.

Descripción:

- Microcontrolador: ATMEGA328P.
Flash: 32K Bytes.
EEPROM: 1K Byte.
RAM: 2K Bytes.
- Rendimiento: alto rendimiento/baja potencia.
- Alimentación de Entrada: (7 – 16) V.
- Drivers: 5A por canal (2 canales) – (2 motores).
- Pulsadores: 2.
- Leds: 3 (Leds de propósito general recomendados para visualizar estado de los sensores).

- Entrada: módulo de inicio.
- Regulador: Regulador de tipo Stop Down para inmunidad a caídas de tensión.
- Asociable a Arduino IDE, como se observa en la figura 22.

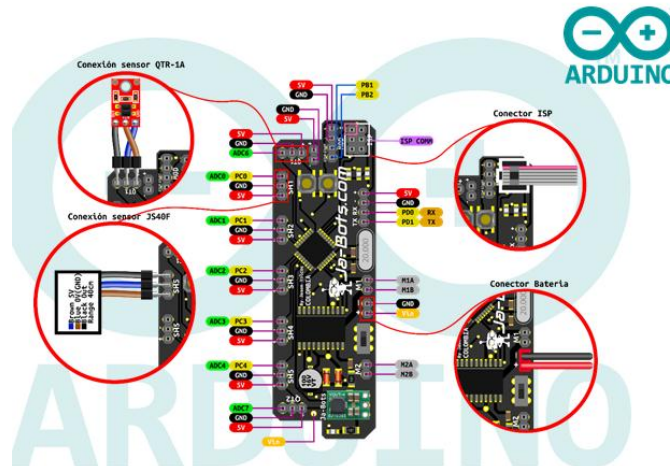


Figura 22: Esquema de conexión de cada uno de sus componentes
Fuente: Elaboración propia (2020)

2.2.2.1.3 Configurar Arduino IDE tarjeta mini black

1. Para poder programar en la tarjeta mini black se debe tener instalado el software Arduino IDE, como se observa en la figura 23.

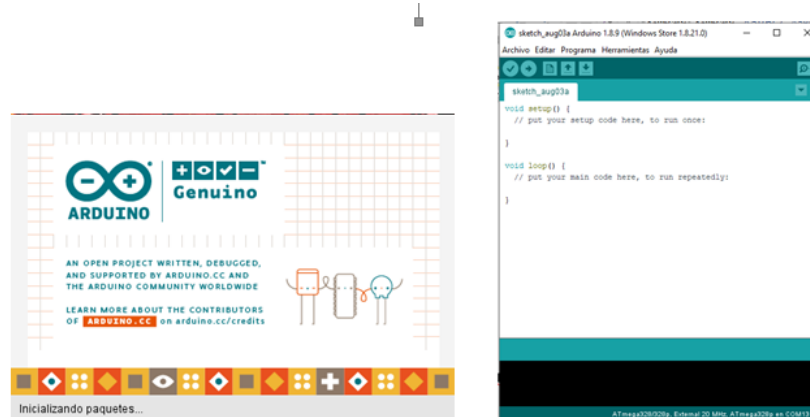


Figura 23: Esquema eléctrico terminado
Fuente: Elaboración propia (2020)

2. Instalación del drive para el programador de Pololu AVR V2 como tal, se debe instalar los drivers necesarios, los cuales se puede encontrar en la página oficial de Pololu, como se observa en la figura 24.



Figura 24: Página oficial de Pololu.
Fuente: Elaboración propia (2020)

3. Una vez ahí, se desliza así abajo hasta encontrar la subsección de “File downloads” y descargas el archivo que se ajuste a tus necesidades dependiendo del sistema operativo. Luego se descarga, y se da click sobre el archivo para realizar la instalación, cómo se observa en la figura 25.

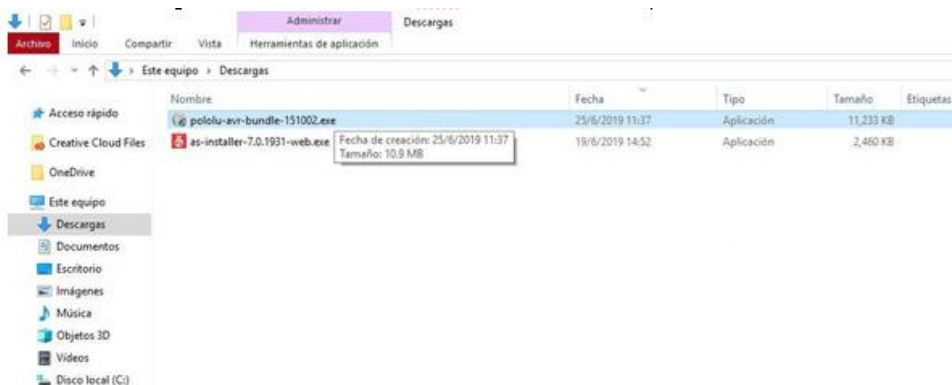


Figura 25: Archivo de instalación Pololu.
Fuente: Elaboración propia (2020)

4. Se da permisos para realizar los cambios en el equipo y luego se mostrará la siguiente ventana donde preguntará que componentes se debe instalar,

para esto se deja todo seleccionado y se le da click en “install” para comenzar con la instalación, como se observa en la figura 26.

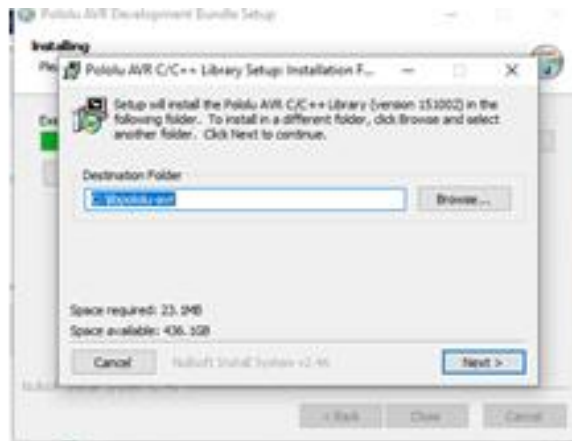


Figura 26: Ventana de instalación.
Fuente: Elaboración propia (2020)

5. Luego se abre otra ventana emergente donde instalara el controlador del USB Programmer, se deja la ruta por defecto y le se da Click en “install” para continuar. Luego se espera que termine el proceso de instalación y luego le se da click en “close” para continuar, como se observa en la figura 27.

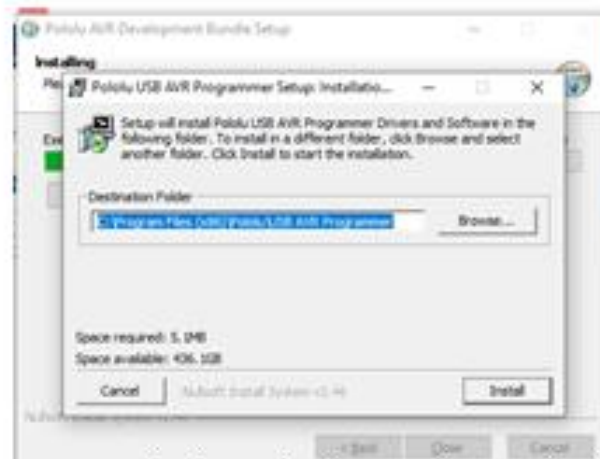


Figura 27: Ventana emergente de instalación
Fuente: Elaboración propia (2020)

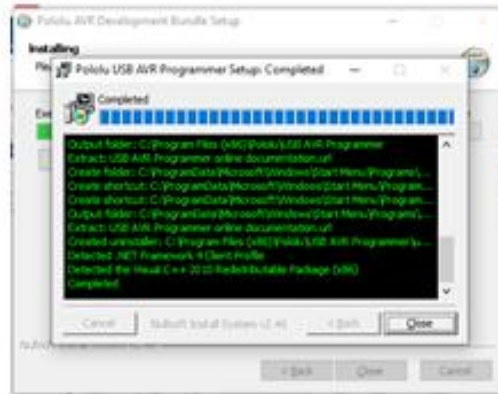


Figura 28: Ventana de instalación.
Fuente: Elaboración propia (2020)

2.2.2.1.4 Conexión para programar

1. Luego de tener el software totalmente instalado, se conecta el programador con la tarjeta mediante el puerto de programación ISP y a su vez el programador mediante USB al computador, como se observa en la figura 29.

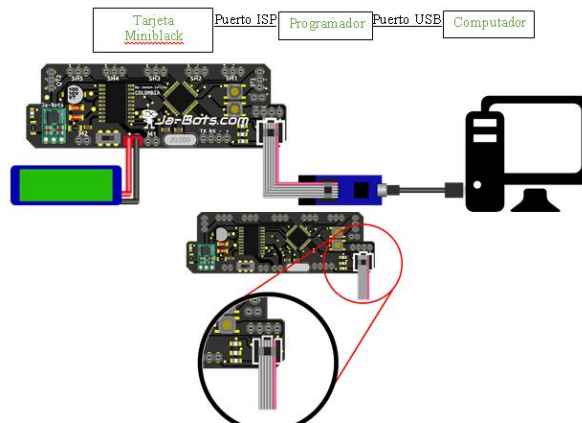


Figura 29: Imagen de conexión de la tarjeta programador.
Fuente: Elaboración propia (2020)

2. Luego de realizar la anterior conexión, se dirige al administrador de dispositivos del computador y se hace doble click en puertos (COM Y LPT) y se verifica que el programador se encuentre conectado, como se observa en la figura 30.

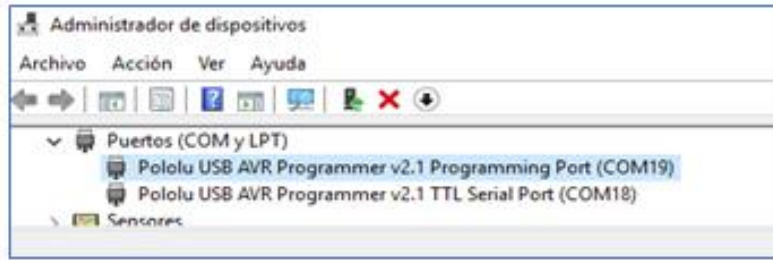


Figura 30: Ventana de los puertos de comunicación
Fuente: Elaboración propia (2020)

3. Se configura la aplicación Arduino IDE para lo cual se dirige a archivos y luego se da click en preferencias, cómo se observa en la figura 31.

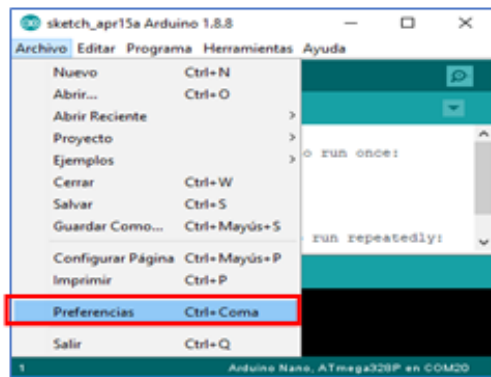


Figura 31: Venta de archivos de Arduino
Fuente: Elaboración propia (2020)

1. Se abrirá una ventana como la que se muestra a continuación en donde se copiará el siguiente enlace.

https://raw.githubusercontent.com/carlosefr/atmega/master/package_carlosefr_atmega_index.json en *Gestor de URL Adicionales de Tarjetas*, esto con el fin de que Arduino IDE pueda reconocer y hacer uso del programador de Pololu AVR, así como del microcontrolador que se está usando, el ATmega328P, como se observa en la figura 32

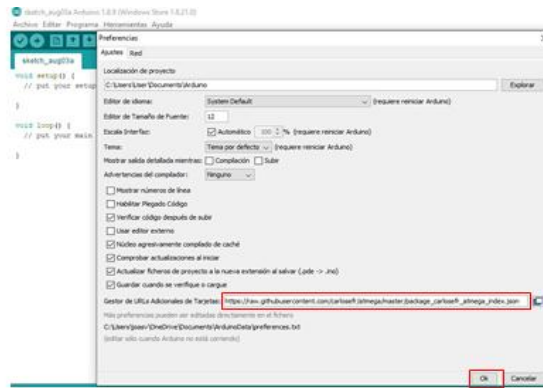


Figura 32: Ventana de preferencias.
Fuente: Elaboración propia (2020)

4. A continuación, se dirige a herramientas y luego se hace click en gestor de tarjeta donde aparecerá una ventana en donde se instalará Barebones Atmega chips (no bootloader), como se observa en la figura 33.

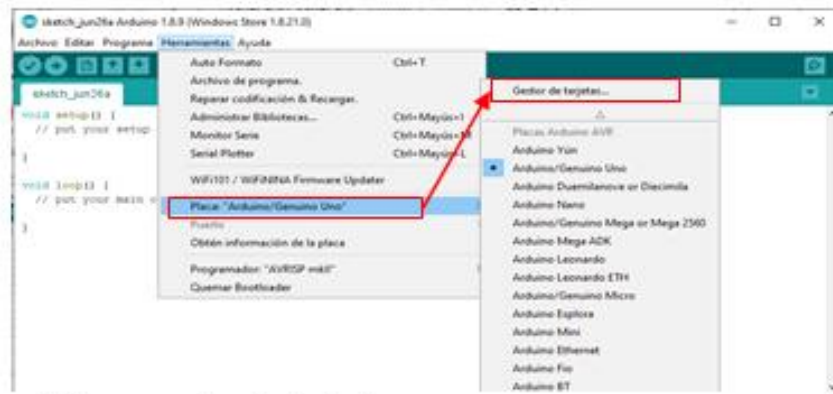


Figura 33: Ventana de gestor de tarjeta.
Fuente: Elaboración propia (2020)

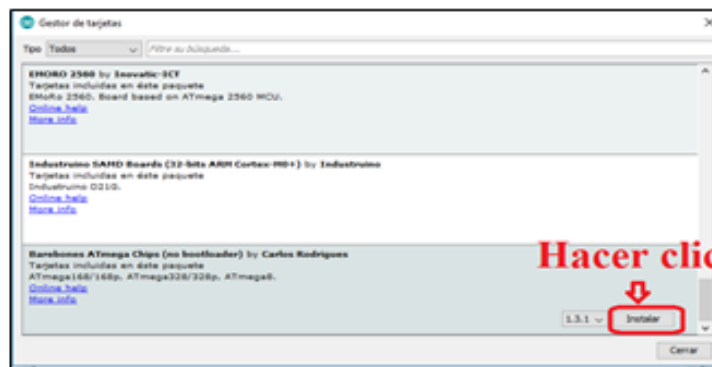


Figura 34: Ventana de paquetes Barebones
Fuente: Elaboración propia (2020)

5. Una vez hecho esto cargara una tarjeta nueva la cual se va a elegir atmega328/328P, está se la encuentra en herramientas/Placa: y se hace click en Atmega328/328P para elegir, como se observa en la figura 35.

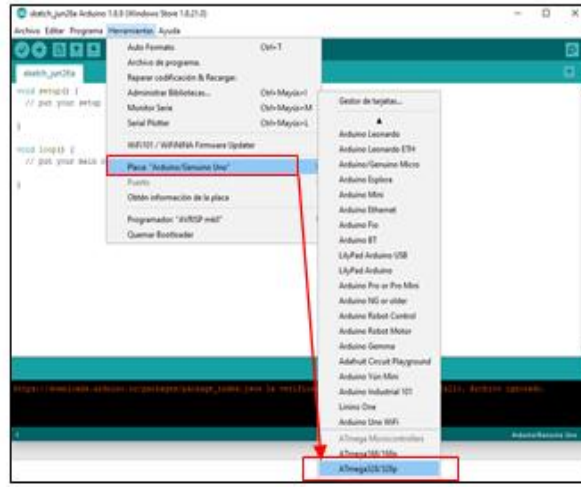


Figura 35: Ventana de Herramientas para elegir placa nueva.
Fuente: Elaboración propia (2020)

6. Luego ingresa a procesadores y se hace click en Atmega328/328P, también se selecciona el reloj que usa la tarjeta mini black, para ello se dirige a Herramienta/Clock y selecciona External20 MHZ, como se observa en la figura 36.

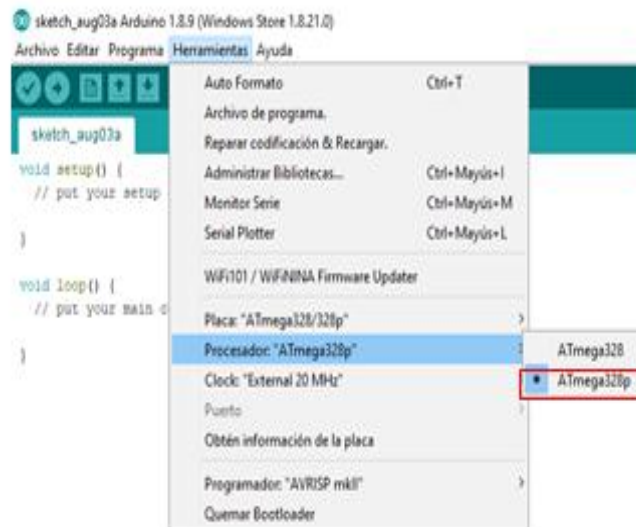


Figura 36: Ventana de selección de Atmega328/328P.
Fuente: Elaboración propia (2020)

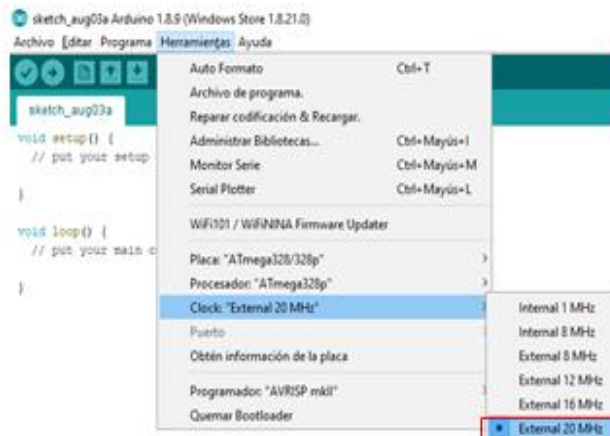


Figura 37: Ventana de selección de Clock External20 MHZ
Fuente: Elaboración propia (2020)

2.2.2.1.5 Configurar programador Pololu AVR

1. Para aquello se debe buscar el puerto COM que se va a utilizar, este está determinado por la asignación del computador. Entonces se dirige a administrador de dispositivos y se verifica el COM del programador, como se observa en la figura 38.

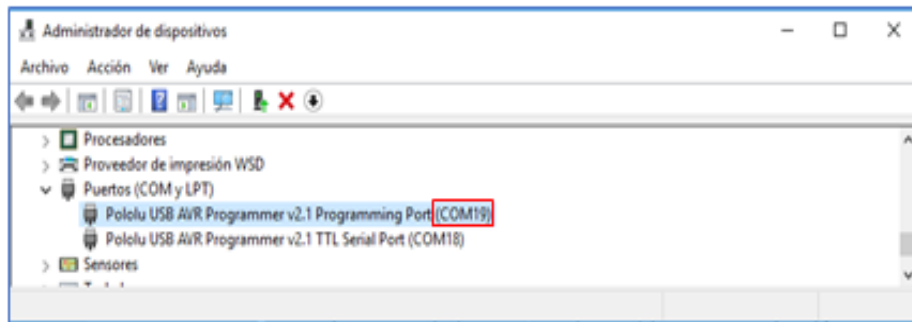


Figura 38: Ventana de administrador de dispositivos.
Fuente: Elaboración propia (2020)

2. Luego en la plataforma Arduino IDE y Herramientas y se selecciona el COM que pertenece al programador Port (en caso que aparece como COM 19 o COM 3, COM 5 es porque el computador automáticamente le da el nombre como se observa en la figura 39).

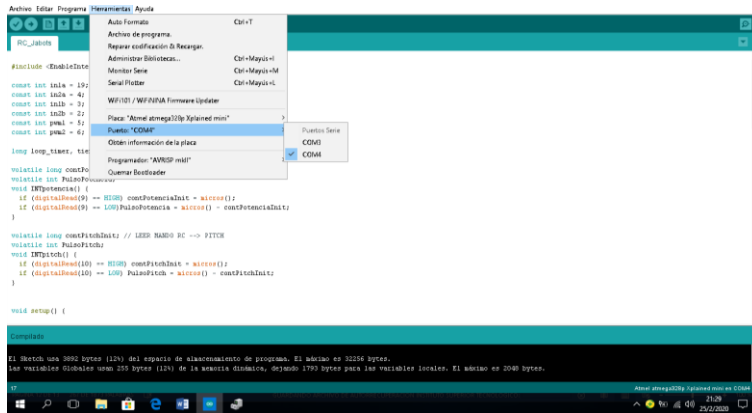


Figura 39: Ventana de COM programador AVR.
Fuente: Elaboración propia (2020)

3. Se procede a escoger el programador, para eso se dirige a Herramientas/programador y se selecciona Atmel STK500 development board, como se observa en la figura 40.

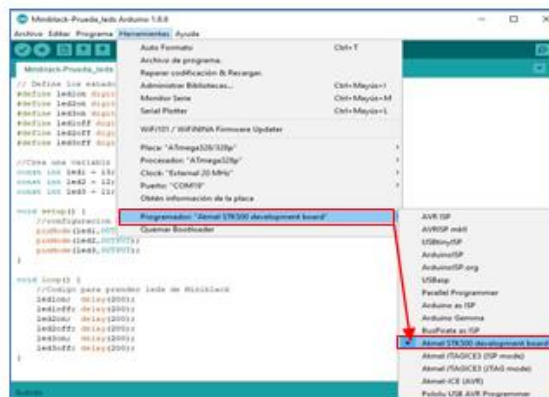


Figura 40: Ventana del programador Atmel STK500.
Fuente: Elaboración propia (2020)

2.2.2.1.6 Programación


1. Una vez ya instalado los programadores se realiza la programación del robot mini sumo RC.
2. Para la programación se va a utilizar la librería **<EnableInterrupt.h>** que proporciona una API consistente en chips. Hace cambios externos y de pines en la serie Atmega, y las interrupciones de pines debidas un Código optimizado para la velocidad; incluye un modo de velocidad extra alta para el código crítico de velocidad como se observa en la figura 41.



```
RC_Jabots Arduino 1.8.12 (Windows Store 1.8.33.0)
Archivo Editar Programa Herramientas Ayuda
RC_Jabots $
#include <EnableInterrupt.h>
#include <EnableInterrupt.h>
```

Figura 41: Librería *EnableInterrupt.h*.
Fuente: Elaboración propia (2020)

- Una vez insertada la librería se procede a dar una constante a los pines de comunicación el cual será la dirección del motor y la velocidad de los motores y el tiempo de ejecución, como se observa en la figura 42.



```
RC_Jabots Arduino 1.8.12 (Windows Store 1.8.33.0)
Archivo Editar Programa Herramientas Ayuda
RC_Jabots $
#include <EnableInterrupt.h>
const int in1a = 19; //direccion motor 1 derecha
const int in2a = 4; //direccion motor 2 derecha
const int in1b = 3; //direccion motor 1 izquierda
const int in2b = 2; //direccion motor 2 izquierda
const int pwm1 = 5; //PWM motor 1 Derecho
const int pwm2 = 6; //PWM motor 2 Izquierdo
long loop_timer, tiempo_ejecucion;
```

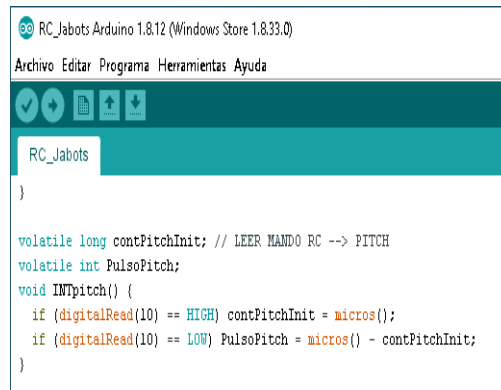
Figura 42: Constantes de los pines de conexión de los motores.
Fuente: Elaboración propia (2020)

- Ahora se procede a la programación del mando RC en el cual se utilizará `volatile long` que será la constante de potencia del RC eso ayudará a controlar la velocidad de los motores, como se observa en la figura 43.

```
volatile long contPotenciaInit; // LEER MANDO RC --> POTENCIA
volatile int PulsoPotencia;
void INTpotencia() {
  if (digitalRead(9) == HIGH) contPotenciaInit = micros();
  if (digitalRead(9) == LOW) PulsoPotencia = micros() - contPotenciaInit;
}
```

Figura 43: Programación del Mando RC.
Fuente: Elaboración propia (2020)

5. Luego de realizar se configura el PITCH el que manda la señal para adelante o para atrás el cual se configura como la programación de mando de potencia, como se observa en la figura 44.



```
RC_Jabots Arduino 1.8.12 (Windows Store 1.8.33.0)
Archivo Editar Programa Herramientas Ayuda
RC_Jabots
}

volatile long contPitchInit; // LEER MANDO RC --> PITCH
volatile int PulsoPitch;
void INTpitch() {
  if (digitalRead(10) == HIGH) contPitchInit = micros();
  if (digitalRead(10) == LOW) PulsoPitch = micros() - contPitchInit;
}
```

Figura 44: Programación de mando Pitch.
Fuente: Elaboración propia (2020)

6. Se realiza la configuración de condiciones iniciales en los pies del motor y del drive I293, esto ayudara a la comunicación entre l placa y el RC, como se observa en la figura 45.



```
RC_Jabots Arduino 1.8.12 (Windows Store 1.8.33.0)
Archivo Editar Programa Herramientas Ayuda
RC_Jabots

void setup() {

  //configuracion de condiciones iniciales
  pinMode(in1a,OUTPUT);
  pinMode(in2a,OUTPUT);
  pinMode(in1b,OUTPUT);
  pinMode(in2b,OUTPUT);
  pinMode(pwm1,OUTPUT);
  pinMode(pwm2,OUTPUT);
  pinMode(10, INPUT_PULLUP); // YAW
  enableInterrupt(10, INTpitch, CHANGE);
  pinMode(9, INPUT_PULLUP); // POTENCIA
  enableInterrupt(9, INTpotencia, CHANGE);
  //Serial.begin(115200);
  delay(100);
}
```

Figura 45: Configuración de condiciones iniciales.
Fuente: Elaboración propia (2020)

7. Se realiza la programación final en el cual va ayudar a la comunicación con el RC, esto ayudara a que controle el RC si va para adelante, atrás, izquierda y derecha, como se observa en la figura 46.

```

RC_Jabots Arduino 1.8.12 (Windows Store 1.8.33.0)
Archivo Editar Programa Herramientas Ayuda
RC_Jabots

while (micros() - loop_timer < 10000);
tiempo_ejecucion = (micros() - loop_timer) / 1000;
loop_timer = micros();

if(PulsoPotencia >=1560) //adelante
{
    loop_timer = micros();
    motors(250,250 );
}
else if(PulsoPotencia <=1150) //atras
{
    loop_timer = micros();
    motors(-100,-100 );
}
else if(PulsoPitch <=1500) //izquierda
{
    loop_timer = micros();
    motors(-100,100 );
}
else if(PulsoPitch >=1700) //derecha
{
    loop_timer = micros();
    motors(100,-100 );
}
else if((PulsoPitch >=1500) && (PulsoPotencia <=1560)) //stop
{
}

```

Figura 46: Programación del RC para el mando.
Fuente: Elaboración propia (2020)

- El Paso siguiente es la función de manejar los motores con PWM y la señal del driver, esto ayuda a la velocidad de los motores tanto para ir para adelante atrás o de izquierda a derecha, como se observa en la figura 47.

```

RC_Jabots Arduino 1.8.12 (Windows Store 1.8.33.0)
Archivo Editar Programa Herramientas Ayuda
RC_Jabots

    loop_timer = micros();
    motors(0,0);
}
//Funcion para manejar los motores con pwm y la señales del driver
int motors(int speedL, int speedR) {
    if (speedL>0){
        digitalWrite(in2b,LOW);
        digitalWrite(in1b,HIGH);
    }
    else{
        digitalWrite(in2b,HIGH);
        digitalWrite(in1b,LOW);
        speedL=speedL*-1;
    }
}

if (speedR>0) {
    digitalWrite(in2a,LOW);
    digitalWrite(in1a,HIGH);
}
else{
    digitalWrite(in2a,HIGH);
    digitalWrite(in1a,LOW);
    speedR=speedR*-1;
}
analogWrite(pwm1, speedR);
analogWrite (pwm2, speedL);

```

Figura 47: Función para manejar los motores con PWM y las señales del driver.
Fuente: Elaboración propia (2020)

2.2.3 Implementación

2.2.3.1.1 Tarjeta mini black

1. La tarjeta mini black está diseñada especialmente para robot mini sumo, está constituida por un Atmega y cuenta con una arquitectura de AVR de 8 bits, con sus respectivas descripciones, como se muestra en la figura 48.

- Microcontrolador: ATMEGA328P.
 - Flash: 32K Bytes.
 - EEPROM: 1K Byte.
 - RAM: 2K Bytes.
 - Rendimiento: alto rendimiento/baja potencia.
- Alimentación de Entrada: (7 – 16) V.
- Drivers: 5A por canal (2 canales) – (2 motores).
- Pulsadores: 2.
- Leds: 3 (Leds de propósito general recomendados para visualizar estado de los sensores).
- Entrada: módulo de inicio.
- Regulador: Regulador de tipo Step Down para inmunidad a caídas de tensión.
- Asociable a Arduino IDE.

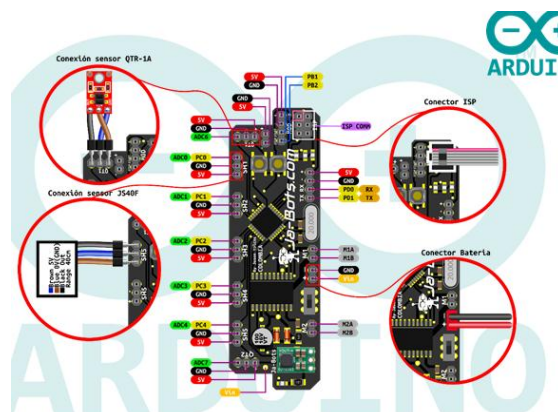


Figura 48: Perforación para cilindros
Fuente: Elaboración propia (2019)

2. Puertos de comunicación de la tarjeta según su especificación de la tarjeta como se observa en la figura 48. .

Pines de comunicación:

- Serial RS232 (PD0 y PD1).
- ISP para programación
- Pines PB1 y PB2 son de propósito general, pero en este diseño se usan, bien para conectar el módulo de inicio, necesario en las competencias de minisumo, o para ser usado mediante radio control de dos canales para manipular el robot de forma remota. Esto para las competencias de minisumo RC.

3. Pines de conexión para la programación según su especificación de la tarjeta como se observa en la figura 49.

Pines Programables:

- pines E/S análogos, 5 los cuales a su vez son digitales
- 4 pines E/S de propósito general.
- 2 pares de pines PWM cada uno para un motor.
- 3 pines asociados a Leds.
- 2 pines asociados a pulsadores.

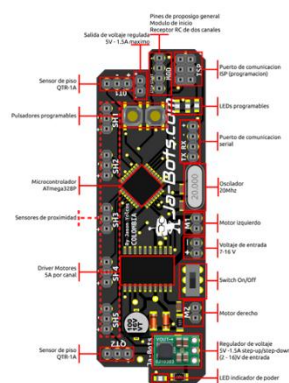


Figura 49: Cableado de electroválvulas y mangueras PU
Fuente: Elaboración propia (2019)

2.2.3.1.2 Radio control 2.4GHz

El radio control tipo pistola consiste en tres canales, que son ideales para categorías de mini sumo RC el cual incluye un receptor con el que se va a comunicar con la tarjeta mini black, como se muestra en la figura 50.

Características:

- Número de canales: 3
- Alimentación: 4 Baterías AA
- Frecuencia 2.4GHz

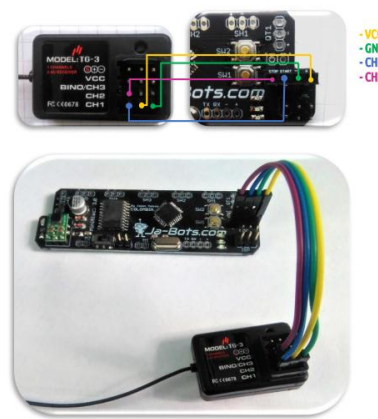


Figura 50: Conexión entre el receptor y la tarjeta mini black.
Fuente: Elaboración propia (2020)

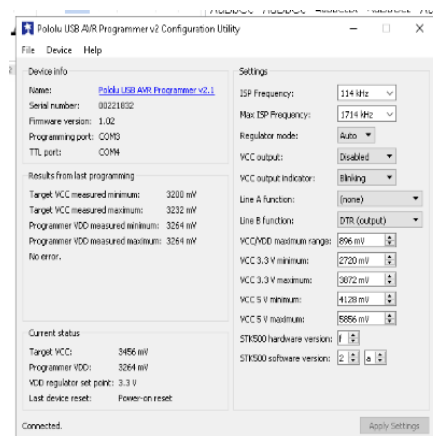


Figura 51: Configuración AVR Programmer
Fuente: Elaboración propia (2020)

5 CONCLUSIONES:

La importancia fundamental es que el proyecto ha sido realmente educativo e instructivo el cual se ha tenido que seleccionar componentes a utilizar que ha llevado investigaciones tales como proyectos de titulación, artículos entre otros. Después de haber revisado estos textos, se procede a la descripción de los fundamentos de los microcontroladores atmega de marca Atmel esto ayudara a entender el funcionamiento del mismo

En diseño y mecánica se ha trabajado en un programa de diseño en 3D o 2D llamado Inventor lo que ayudara a diseñar el chasis del robot mini sumo y poder comprobar que todas las piezas encajen en el modelo antes de construirlo el cual se ha utilizado las más sencillas herramientas y eficaz para el prototipo.

Para la programación se ha creado un programa desde cero para el robot RC en Arduino que es fácil de programar, pero posee errores leves en la recopilación el cual genera desconexión entre la placa y el RC y no es muy confiable ya que al rato de la recopilación tiene inconvenientes.

RECOMENDACIONES:

Para proyectos de competencia de robot mini sumo demanda un alto nivel de programación para aquello es recomendable programar en el software atmel ya que ese nos ayuda con mayor variedad de librerías a diferencia de arduino que da algunos inconvenientes al rato de recopilar la información ya que de eso dependerá que el robot tenga un buen funcionamiento.

Se recomienda que para programación de robot de competencia usar lo que es Atmel estudio ya que este software presenta mejor desempeño en su recopilación de programa y posee más variedad de librerías a diferencia de Arduino.

Realizar una programación de manera ordenada y evitar la utilización de elementos innecesarios esto puede hacer del programa algo extenso y muy difícil de corregir o modificar una función en caso de ser necesario.

Es recomendable observar e identificar los pines de conexión de la placa mini black, si existe una mala conexión puede tener inconvenientes en la recopilación del programa y no va tener un buen funcionamiento.

Se recomienda realizar la conexión bien de la batería antes de encender la placa mini black ya que si no se conecta bien puede dañar el drive que controla los motores ya que no consta con protección.

.

6 REFERENCIAS:

6.1 Bibliografía

Sánchez, J. A. (2002). Avances en robótica y visión por computador. En U. d.

Mancha. España.

Proyecto mini sumo. (S. f.). (23 de julio de 2019). Obtenido de

<https://es.scribd.com/document/363688314/proyecto-minisumo>

Santillana (2017) Que es la programación de Arduino recuperado de

<https://www.bejob.com/que-es-la-programacion-con-arduino-y-para-que-sirve/>

Martin tomas (2008). Microcontroladores: Lenguaje C para AVR GCC Tutorial recuperado de

Bolton, W. (24 de agosto de 2013). Mecatrónica. México: Alfa omega.

Calaza, G. (23 de julio de 2014). Taller de Arduino. Barcelona: Marcombo.

Alfredo P. (2012) Proyecto de Arduino Pamplona España

Ocaña, G., & Ocaña, G. (2015). Robótica Educativa. Madrid: Dextra.

Aprendiendo Arduino (27 de marzo de 2019) Construcción y Programación de Robot

Minisumo obtenido de <https://aprendiendoarduino.wordpress.com/2019/05/06/construcción-y-programación-de-robot-minisumo/#comments>

Arduino. CC. Guía Introducción (17 de febrero de 2010) Que es Arduino recuperado de <https://www.arduino.cc/en/Guide/Introduction>

Frank Mecafenix (abril de 2017) Como funciona Arduino La enciclopedia de la ingeniería recuperado de <https://www.ingmecafenix.com/electronica/arduino/>

Vanesa A. (2014) La evolución de la robótica en la industria a lo largo de la historia

Madrid España

Capek, C. (31 de enero de 1938). Robots en el mundo. Obtenido de

<http://blog.belzec.net/historia-de-los-robots-el-primer-robot/>.

Cuartielles, D. (1999). Professional Android Open Accessory Programming. Estados

Unidos: Piliphins.

ESCUELA DE NEGOCIOS BUSINNES SCHOOL. (3 de AGOSTO de 2018).

esneca. Obtenido de <https://www.esneca.com/>: <https://www.esneca.com/>

XATAKA BASICS (3 de AGOSTO de 2018). Obtenido de

<https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno>

Alcalde P.S (2014). Electrónica. Madrid España Editorial Paraninfo.

Jeremy Blum (2014). Arduino a Fondo. Madrid España Editorial Anaya Multimedia.

Guillermo Barbadillo (2012). Proyecto de Arduino Sumo Robótico Pamplona España.

Carlos contreras (2010) Manual de AutoCAD Barcelona España

Sánchez-Martín F.M. (2014). Historia de la robótica: de Arquitas de Tarento al Robot da Vinci (Parte II) Barcelona España.

7 ANEXOS:

Anexo1: Materiales y componentes del robot mini sumo

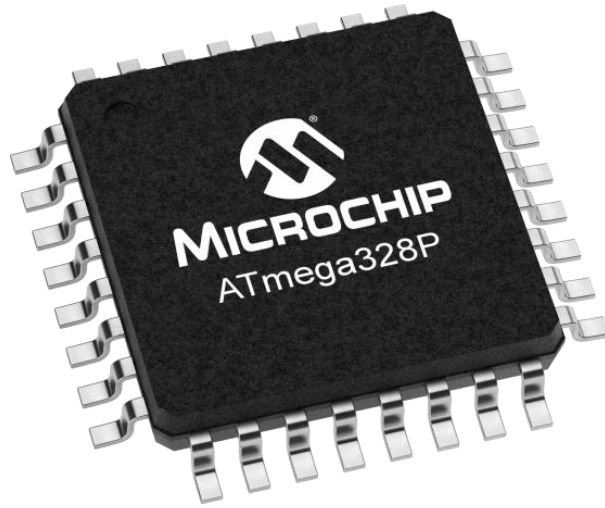


Figura 52: Datafish Atmega 328P
Fuente: Elaboración propia (2020)

Anexo2: Conexión de la batería en placa mini black

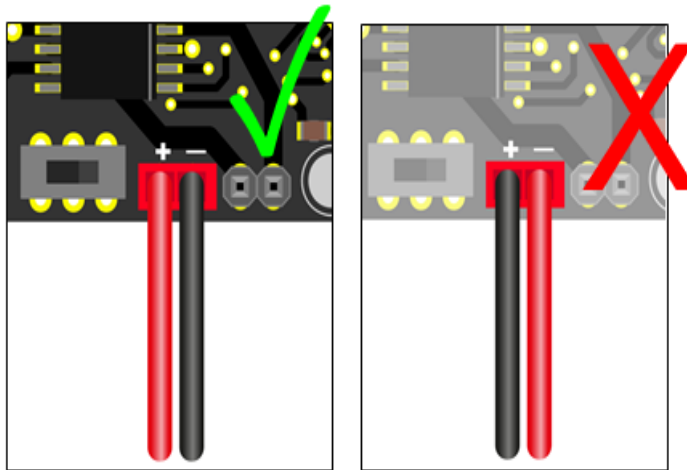


Figura 53: Conexión de la batería lipo en la placa mini black
Fuente: Elaboración propia (2020)

Anexo3: Ficha técnica de la placa mini black.

PORT	Physic PIN	PIN	Atmel Name	Arduino Name	Función
B	12	PB0	LED1	8	LED 1
	13	PB1	LED2	9	LED 2 / Salida Digital
	14	PB2	LED3	10	LED 3 / Salida Digital
	15	PB3	TX2	11	TX 2/ Salida Digital
	16	PB4	RX2	12	RX 2/ Salida Digital
	17	PB5	LED 4	13	LED 4

C	23	PC0	S1	A0	Sensor 1/digital o análogo
	24	PC1	S2	A1	Sensor 2/digital o análogo
	25	PC2	S3	A2	Sensor 3/digital o análogo
	26	PC3	S4	A3	Sensor 4/digital o análogo
	27	PC4	S5	A4	Sensor 5/digital o análogo
	28	PC5	S6	A5	Sensor 6/digital o análogo

E	19	ADC 6	S7	A6	Sensor 7/ Análogo
	22	ADC 7	S8	A7	Sensor 8 / Análogo

D	30	PD0	RX	0	Comunicación Serial
	31	PD1	TX	1	Comunicación Serial
	32	PD2	INT 0	2	Entrada Digital
	1	PD3	INT 1	3	Entrada Digital
	2	PD4	SW2	4	Pulsador 2
	9	PD5	SW1	5	Pulsador 1
	10	PD6	MSTO	6	Módulo stop / Digital
	11	PD7	MSTA	7	Módulo start / Digital

Figura 54: Ficha técnica de la placa
Fuente: Elaboración propia (2020)

Anexo4: Conexión entre la placa y el mando de comunicación del RC

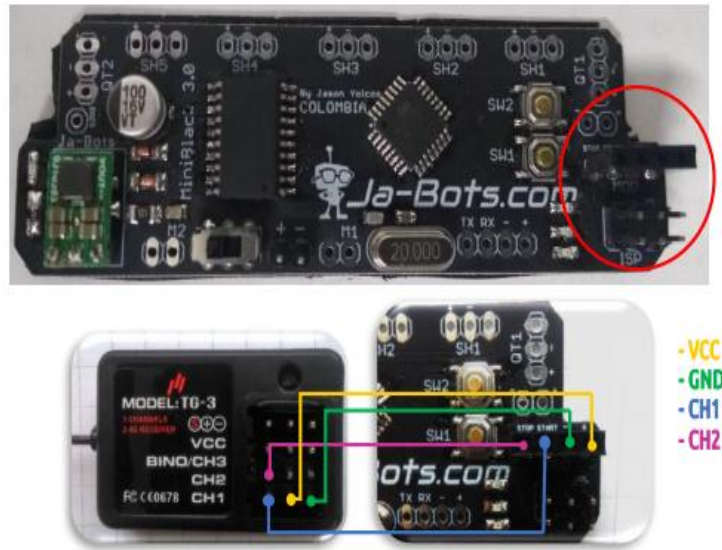


Figura 55: Conexión entre la placa y el conector del RC
Fuente: Elaboración propia (2020)

Anexo5: Robot mini sumo



Figura 56: Robot mini sumo
Fuente: Elaboración propia (2020)

Anexo5: Programación mini sumo RC

```
#include <EnableInterrupt.h>

const int in1a = 19; //direccion motor 1 derecha

const int in2a = 4; //direccion motor 2 derecha

const int in1b = 3; //direccion motor 1 izquierda

const int in2b = 2; //direccion motor 2 izquierda

const int pwm1 = 5; //PWM motor 1 Derecho

const int pwm2 = 6; //PWM motor 2 Izquierdo

long loop_timer, tiempo_ejecucion;

volatile long contPotencialnit; // LEER MANDO RC --> POTENCIA

volatile int PulsoPotencia;

void INTpotencia() {

    if (digitalRead(9) == HIGH) contPotencialnit = micros();

    if (digitalRead(9) == LOW)PulsoPotencia = micros() - contPotencialnit;

}

volatile long contPitchInit; // LEER MANDO RC --> PITCH

volatile int PulsoPitch;

void INTpitch () {

    if (digitalRead(10) == HIGH) contPitchInit = micros();

    if (digitalRead(10) == LOW) PulsoPitch = micros() - contPitchInit;

}
```

```

void setup() {

  //configuracion de condiciones iniciales

  pinMode(in1a,OUTPUT);

  pinMode(in2a,OUTPUT);

  pinMode(in1b,OUTPUT);

  pinMode(in2b,OUTPUT);

  pinMode(pwm1,OUTPUT);

  pinMode(pwm2,OUTPUT);

  pinMode(10, INPUT_PULLUP);          // YAW

  enableInterrupt(10, INTpitch, CHANGE);

  pinMode(9, INPUT_PULLUP);          // POTENCIA

  enableInterrupt(9, INTpotencia, CHANGE);

  //Serial.begin(115200);

  delay(100);

}

void loop() {

  while (micros() - loop_timer < 10000);

  tiempo_ejecucion = (micros() - loop_timer) / 1000;

  loop_timer = micros();

  if(PulsoPotencia >=1560) //adelante

```



```

{
    loop_timer = micros();
    motors(250,250 );
}

else if(PulsoPotencia <=1150) //atras

{
    loop_timer = micros();
    motors(-100,-100 );
}

else if(PulsoPitch <=1500) //izquierda

{
    loop_timer = micros();
    motors(-100,100 );
}

else if(PulsoPitch >=1700) //derecha

{
    loop_timer = micros();
    motors(100,-100 );
}

else if((PulsoPitch >=1500) && (PulsoPotencia <=1560)) //stop

{

```

```

        loop_timer = micros();

        motors(0,0);

    }

else if((PulsoPitch >=1600) && (PulsoPitch <=1640)) //stop

    {

        loop_timer = micros();

        motors(0,0);

    }

}

//Funcion para manejar los motores con pwm y la señales del driver

int motors(int speedL, int speedR){

    if(speedL>0){

        digitalWrite(in2b,LOW);

        digitalWrite(in1b,HIGH);

    }

    else{

        digitalWrite(in2b,HIGH);

        digitalWrite(in1b,LOW);

        speedL=speedL*-1;

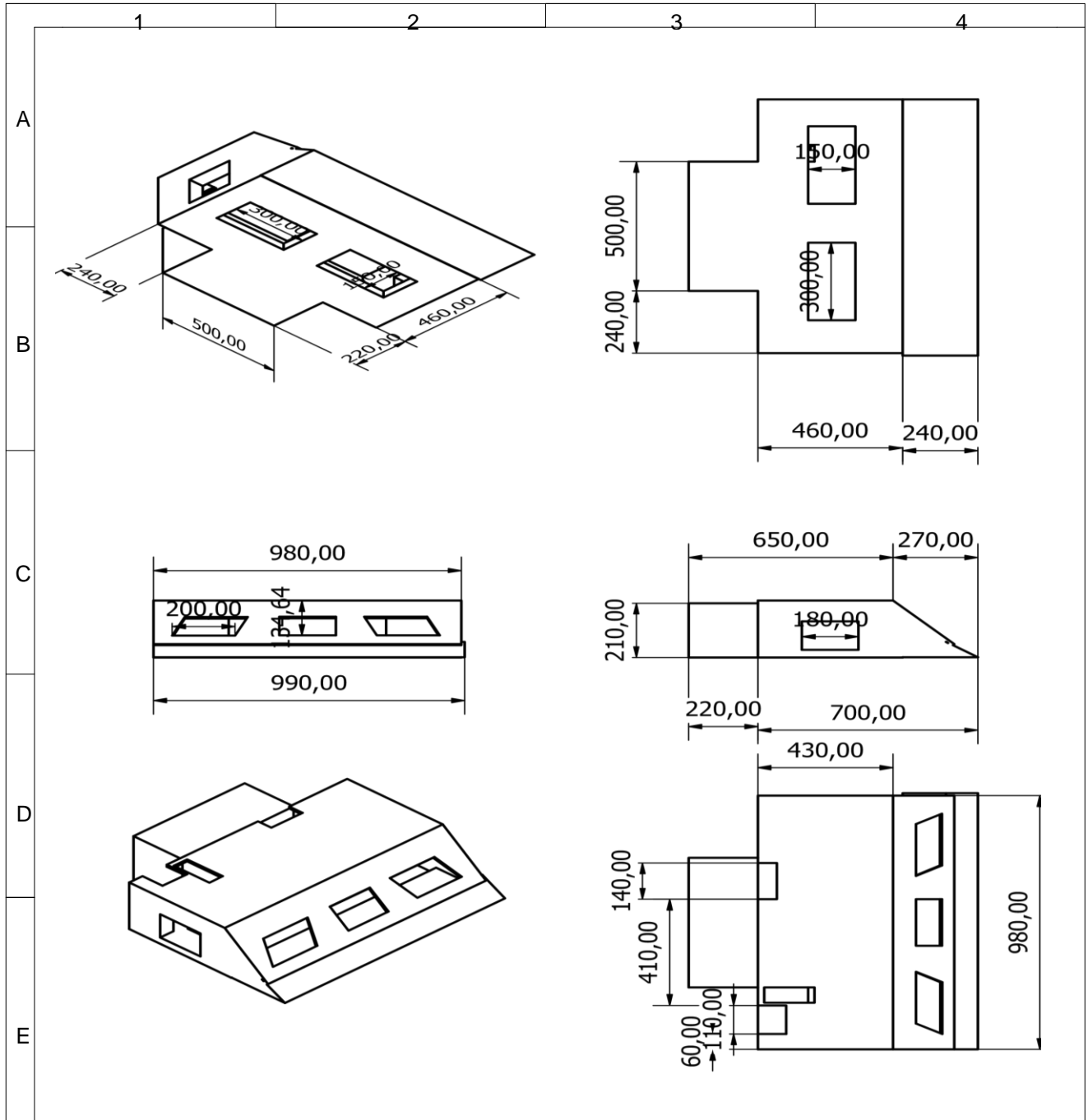
    }


    if(speedR>0){

```

```
    digitalWrite(in2a,LOW);  
    digitalWrite(in1a,HIGH);  
}  
else{  
    digitalWrite(in2a,HIGH);  
    digitalWrite(in1a,LOW);  
    speedR=speedR*-1;  
}  
analogWrite(pwm1,speedR);  
analogWrite(pwm2,speedL);  
}
```

Anexo6: Diseño del robot mini sumo RC



				Tolerancia		Materia	Acero Inoxidable
						Denominació	Chasis robotmini sumo
							Escal
							1:2
						Número de	01 de 01
							
Edició	Modificació	Fecha	Nombre			Sustitució	
					Instituto Superior Tecnológico Vida Nueva		