



UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN

ROBOT MÓVIL DE ENTREGA AUTÓNOMO

T E S I S

QUE PARA OBTENER EL TÍTULO DE

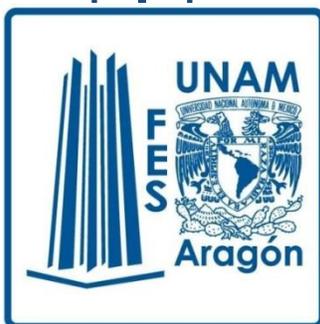
INGENIERO ELÉCTRICO ELECTRÓNICO

P R E S E N T A

JOSÉ RODRIGO FLORES ESPINO

DIRECTOR DE TESIS

ING. JOSÉ RAYMUNDO REYNA OROZCO



Ciudad Nezahualcóyotl, Estado de México, 2019



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

Gracias de corazón, a mi asesor José Raymundo Reyna Orozco, por su criterio, dedicación, paciencia y motivación. Gracias por confiar en mi proyecto y darle una aplicación más amplia. Ha sido un privilegio contar con su guía y ayuda.

De la manera más sincera y cordial le concedo las gracias al ingeniero Joel García Zárraga, por su tiempo, guía y continúa aportación a la mejora de éste proyecto.

Gracias a mi hermana Sheila por siempre confiar en mí, por siempre ofrecerme su apoyo, por ser una persona que me mantiene trabajando en mis metas y aspiración, su vida profesional me motiva e incentiva con el propósito de pensar más a fondo las acciones que uno realiza.

Gracias a mi sobrina Denisse por mostrarme su cariño y respeto, su tenacidad para enfrentar las adversidades de la vida realmente me inspiró a terminar este proyecto, esta Tesis se la dedicó con mucho afecto.

Gracias a todas las personas que me ayudaron en la realización de este proyecto brindándome su apoyo, orientación y consejos.

Resumen de contenido

Capitulo C	Conceptos principales.....	5
C.1	Justificación.....	5
C.2	Introducción.....	5
C.2.1	Antecedentes históricos.....	6
C.2.2	Antecedentes de los leguajes de programación.....	8
Capitulo 1	Softwares utilizados.....	10
1.1	Proteus.....	10
1.2	Mikro C for Pic.....	11
Capitulo 2	Desarrollo conceptual.....	12
2.1	Aplicaciones.....	12
2.2	Ventajas.....	13
2.3	Formación del proyecto.....	13
2.3.1	Implementación de las estructuras electrónicas externas.....	15
Capitulo 3	Elementos de las estructuras electrónicas externas.....	17
3.1	Sensores.....	17
3.2	Transistores.....	17
3.3	Relevadores.....	18
3.4	Fototransistores.....	19
3.5	Microcontroladores.....	20
3.5.1	Estructura del microcontrolador	20
Capitulo 4	Estructura electrónica externa 1.....	21
4.1	Circuito de enclave.....	21
4.2	Circuito de enclave con relevador y transistor.....	22
4.3	Simulación.....	24
4.3.1	Circuito completo.....	27
Capitulo 5	Estructura electrónica externa 2.....	28
5.1	Código.....	29
5.2	Simulación.....	30
5.2.1	Circuito completo.....	32

Capitulo 6	Designación de la estructura electrónica externa.....	33
6.1	Implementación física.....	33
6.2	Diagrama electrónico íntegro.....	34
Capitulo 7	Robot móvil.....	36
7.1	Diagramas de flujo.....	38
7.1.1	Diagrama de flujo 1.....	40
7.1.2	Diagrama de flujo 2.....	42
7.2	Configuración del microcontrolador.....	44
7.3	Código de programación 1.....	44
7.4	Código de programación 2.....	47
7.5	Simulación.....	49
7.5.1	Circuito electrónico del robot.....	51
7.6	Implementación física.....	51
7.6.1	Circuito electrónico del robot íntegro.....	53
Capitulo 8	Electroimán.....	54
8.1	Implementación.....	55
Capitulo 9	Diseño del chasis.....	57
9.1	Diseño de la base para los motores.....	58
9.2	Implementación.....	59
Capitulo 10	Conclusiones.....	61
Capitulo 11	Referencias.....	62
11.1	Bibliografía.....	62
11.2	Ciberografía.....	62

C

Conceptos principales

C.1 **Justificación**

Los requerimientos de la sociedad actual, giran en torno a satisfacer la necesidad de aumentar la producción de diversos artículos, a reducir los tiempos de ejecución y a tener una calidad uniforme en dichos artículos.

Esto hace que las industrias miren cada vez más hacia una automatización basada en el uso de computadoras. La inflexibilidad y generalmente el alto coste de las maquinas encargadas de las diversas labores en las industrias, a menudo llamadas *sistemas de automatización duros*, han llevado a un interés creciente en el uso de robots, los cuales muchos de estos están disponibles comercialmente.

El desarrollo presentado en este trabajo, pretende satisfacer o mitigar la labor relacionada al transporte de materiales a una determinada zona, donde se utilizaran posteriormente en diferentes procesos.

C.2 **Introducción**

Durante la historia de la humanidad el hombre se ha esforzado por mejorar su entorno para poder adaptarse y tener una vida más cómoda. Desde entonces han surgido mecanismos y máquinas primitivas que han ayudado al hombre con esta labor.

En el transcurso de la Segunda Guerra Mundial, la sociedad se empezó a dar cuenta de la gran importancia que tenían ciertas disciplinas, como lo son la eléctrica, la electrónica, la automatización y el control. Es en esta época de la historia dónde los sistemas de control y automatización tuvieron un gran auge.

Esta influencia de automatizar procesos, llegó de manera general a la sociedad, al término de la guerra, con aparatos que facilitarían las tareas cotidianas de la gente. Al tener tanta demanda estas tecnologías, se empezaron a integrar más mecanismos y por ende a incorporar más disciplinas en la fabricación de éstos dispositivos.

Al establecerse de manera más íntegra la automatización y el control, se ha definido como una área científico-tecnológica clave para el desarrollo de la sociedad. Con beneficios directos en todos los ámbitos de ésta, como lo son en la medicina, educación, gobierno, agricultura, seguridad, etc.

Por esta razón en esta sociedad moderna se necesita la elaboración de procesos cada vez más detallados y precisos para la realización de diversos productos que satisfagan alguna necesidad.

Uno de los sectores donde más se utiliza esta rama de la ingeniería, puede ser el económico, ya que en varias empresas la optimización de costes, la reducción de tiempos de ejecución y la disposición de recursos, son factores determinantes en la medición de su eficiencia, al solucionar problemas y/o necesidades de la población.

Es aquí donde la automatización y los procesos de control brindan una solución o alternativa viable a la mejora de estos procesos. Ya que algunas de estas actividades son repetitivas, monótonas y arduas, por lo tanto puede ser pertinente suplir a las personas encargadas a estas labores, por un robot.

C.2.1 **Antecedentes históricos**

La palabra *robot* tiene un origen checo, relacionada de la palabra *robota*, que significa trabajo. El diccionario Webster define el concepto de robot como un dispositivo automático que ejecuta funciones ordinariamente asignada a los seres humanos.

La definición dada por el Robot Institute of America de la palabra robot es, un manipulador reprogramable multifuncional, diseñado para mover materiales, piezas o dispositivos especializados a través de movimientos programados variables para la realización de una diversidad de tareas. De forma genérica un robot es un manipulador reprogramable de uso general con sensores externos y que pueden efectuar diferentes tareas. Cabe destacar que el robot debe poseer "inteligencia" que se debe normalmente a los algoritmos de computador asociados con su sistema sensorial y de control.

La palabra robot se introdujo en la lengua inglesa en 1921 con el drama satírico R.U .R. de Karel Capek (Rossum Universal Robots). En este trabajo, los robots son maquinas que se asemejan a los seres humanos, pero que trabajan sin descanso. Inicialmente, los robots se fabricaron como ayuda para sustituir a los operarios humanos, pero posteriormente los robots se vuelven contra sus creadores, aniquilando a toda la raza humana. Después de esta representación, diversos personajes aportaron su conceptualización de ésta palabra, algunos de estos ejemplos es el alemán Fritz Lang, con su película *Metrópolis* de 1926 que reforzaba la idea que tenía Capek acerca de los robots.

Los primeros trabajos que condujeron a los robots industriales de hoy día, se remonta al periodo que siguió inmediatamente después a la Segunda Guerra Mundial, pero unos años antes algunos científicos ya habían hecho avances. Durante los años finales de la década de los cuarenta, comenzaron programas de investigación en Oak Ridge y Argonne National Laboratories para desarrollar manipuladores mecánicos controlados de forma remota para manejar materiales radioactivos. Estos sistemas eran del tipo *maestro-esclavo*, diseñados para reproducir fielmente los movimientos de mano y brazo realizados por un operario, el manipulador maestro era guiado por el usuario a través de una secuencia de movimientos, mientras el manipulador esclavo duplicaba a la unidad maestra tan fidedignamente como le era posible.

El trabajo sobre manipuladores maestro-esclavo fue seguido rápidamente por sistemas más sofisticados, capaces de operaciones repetitivas autónomas. A mediados de los años cincuenta, George C. Devol desarrollo un dispositivo el cual nombro como *dispositivo de transferencia programada articulada*, un manipulador cuya operación podía ser programada y por lo tanto modificada, también podía seguir una secuencia de movimientos, determinados por las instrucciones del programa. Posteriores desarrollos de este concepto por Devol y Joseph F. Engelberger condujo al primer robot industrial, introducido por Unimation Inc, en 1959. La clave de este dispositivo era el uso de una computadora en conjunción con un manipulador para producir una máquina que podía ser “enseñada” para realizar una variedad de tareas de forma automática.

Se hizo visible en los años sesenta que la flexibilidad de estas máquinas podía mejorar significativamente mediante el uso de una realimentación sensorial. Al comienzo de esta década, H. A. Ernst [1962] publicó el desarrollo de una mano mecánica controlada por computador con sensores táctiles. Este dispositivo, llamado MH-1 podía “sentir” bloques y usar esta información para controlar la mano, de manera que apilaba los bloques sin la ayuda de un operario. Este trabajo es uno de los primeros ejemplos de un robot capaz de tener una conducta adaptiva en un entorno razonablemente no estructurado.

Durante el mismo período Tomovic y Boni [1962] desarrollaron una mano prototipo provista con un sensor de presión que detectaba el objeto y proporcionaba una señal de realimentación de entrada a un motor para iniciar uno de dos modos de aprehensión. Una vez la mano estaba en contacto con el objeto, se enviaba a una computadora información proporcional a su tamaño y peso mediante estos elementos sensibles a la presión.

A finales de los años sesenta, Mc Carthy [1968] y sus colegas en el Stanford Artificial Intelligence Laboratory publicaron el desarrollo de una computadora con manos, ojos y oídos, es decir, manipuladores, cámaras de TV y micrófonos. Demostraron un sistema que reconocía mensajes hablados, “veía” bloques distribuidos sobre una mesa y los manipulaba de acuerdo a instrucciones. Mientras otros países en particular Japón comenzaron a ver el potencial de los robots industriales.

En las posteriores décadas, se fueron haciendo progresivamente más avances, como el hecho por Cincinnati Milacron en 1974, introdujo su primer robot industrial controlado por computadora capaz de levantar más de 100 libras así como seguir a objetos móviles en una línea de montaje. También los hechos por IBM en [1975] gracias a Will y Grossman, desarrollaron un manipulador controlado por computadora con sensores de contacto y fuerza para realizar montajes mecánicos en una máquina de escribir de veinte piezas.

Hoy en día vemos la robótica como un campo de trabajo mucho más amplio, en una serie de aéreas interdisciplinarias, que incluyen cinemática, dinámica, planificación de sistemas de control, sensores, lenguajes de programación e inteligencia artificial.

C.2.2

Antecedentes de los lenguajes de programación

Los algoritmos utilizados para cumplir con diferentes funciones suelen estar comprendidos en los módulos de programación.

Un gran obstáculo cuando se usan manipuladores como máquinas de uso general es la dificultad de una comunicación eficiente y adecuada entre el usuario y el sistema del robot, de tal forma que el usuario pueda dirigir al manipulador para cumplir la tarea asignada. Hay varias maneras de comunicarse con un robot, y tres soluciones generales para lograrlo, que son: *reconocimiento de palabras separadas, enseñanza-repetición y lenguajes de programación de alto nivel.*

Los sistemas de reconocimiento de voz antiguos son bastante primitivos y suelen depender de quien habla. Estos sistemas podían reconocer un conjunto de palabras concretas de un vocabulario muy limitado y en general exigen al usuario una pausa entre las palabras, aunque actualmente es posible reconocer las palabras separadas en tiempo real debido a los cada vez más rápidos componentes de las computadoras y algoritmos de procesamiento más eficientes.

La enseñanza-repetición, también conocido como “guiado”, era la solución más común utilizada en los robots industriales. Este método implica enseñar al robot los movimientos que el usuario desea que realice. La enseñanza-repetición se lleva a cabo normalmente con los siguientes pasos:

- 1) Dirigiendo al robot con un movimiento lento utilizando el control manual para realizar la tarea completa y grabando los ángulos del movimiento.
- 2) Reproduciendo y repitiendo el movimiento enseñado.
- 3) Si el movimiento enseñado es el correcto, entonces se hace funcionar al robot a la velocidad correcta en el modo repetitivo.

Guiar al robot en movimiento lento, puede ser en general llevado a cabo de varias maneras usando un joystick, un conjunto de botones (uno para cada movimiento) o incluso un sistema de manipulación maestro-esclavo.

Los lenguajes de programación de alto nivel suministran una solución más general para resolver el problema de comunicación hombre-robot. La utilización de robots para llevar a cabo diferentes tareas requieren técnicas de programación en lenguajes de alto nivel ya que ciertos robots en particular los encargados en las líneas de producción suele confiar en la realimentación de los sensores y este tipo de interacción solo puede ser mantenida por métodos de programación que contengan condiciones.

Las soluciones de la programación se pueden clasificar en dos grandes categorías: *la programación orientada al robot y orientada a objetos, con la programación a nivel de tarea.*

En la programación orientada al robot, cada tarea de la línea de producción se describe específicamente como una secuencia de movimientos del robot, se guía y se controla al robot durante toda la realización de la tarea de tal forma que, por lo general cada línea de programación se corresponde con una acción del robot.

Por otra parte la **programación a nivel de tarea** se describe como una secuencia de objetivos a lograr con el posicionamiento de los objetos y no como un conjunto de movimientos necesarios para que se logren estos objetivos, por lo tanto, ningún movimiento concreto del robot queda especificado de esta manera.

1

Softwares utilizados

1.1 Proteus

Proteus es un entorno integrado diseñado para la realización completa de proyectos de construcción de equipos electrónicos en todas sus etapas: diseño, simulación y construcción.

El software se compone de varias herramientas perfectamente integradas entre sí:

- Diseñador del esquema electrónico y simulación de circuitos electrónicos.
- Diseñador de la placa de circuito impreso.
- Visor tridimensional.
- Módulo de gestión de la lista de materiales.
- Módulo de programación y simulación de microcontroladores.

Proteus dispone de un módulo para la elaboración avanzada de esquemas electrónicos, que incorpora una librería de más de 6,000 modelos de dispositivos digitales y analógicos.

El módulo diseño PCB es la herramienta para la elaboración de placas de circuito impreso con posicionador automático de elementos y generación automática de pistas, que permite el uso de hasta 16 capas. (véase imagen 1.0)

El módulo de visualización 3D nos proporciona una vista en tres dimensiones de nuestro diseño y nos facilita una imagen de la placa del circuito impreso con la misma apariencia que en la vida real.(véase imagen 1.1)

Éste software fue diseñado por la compañía Labcenter Electronics Ltd. Es una compañía fundada en 1998. Desde hace más de 25 años desarrolla y comercializa este software, actualmente se vende en 50 países.

Imagen 1.0

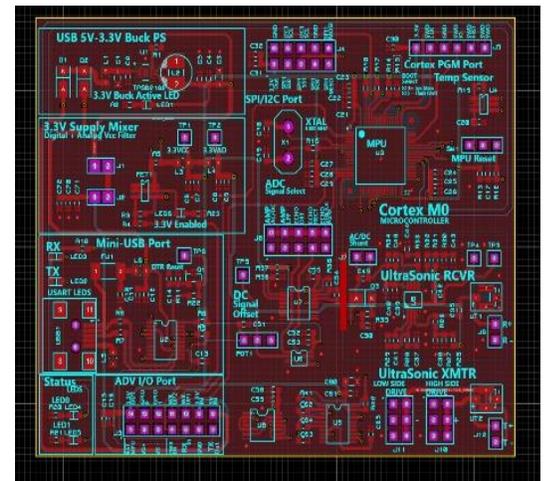


Imagen 1.1

1.2 Mikro C for PIC

MikroC Pro for PIC es un potente y versátil compilador de microcontroladores PIC de Mikroelektronika. Es fácil de aprender y fácil de usar con un entorno de desarrollo integrado (IDE) altamente avanzado, compilador que cumple con ANSI, un amplio conjunto de bibliotecas de hardware y software fáciles de usar, documentación completa y muchos ejemplos listos para usar. (véase la imagen 1.2)

Mikro C Pro for PIC admite microcontroladores PIC de 8 bits de Microchip, como las series de microcontroladores PIC12, PIC16 y PIC18.

Una versión demo totalmente funcional limitada a 2Kb de tamaño de código de salida se puede descargar de forma gratuita desde el sitio web de Mykroelektronika.

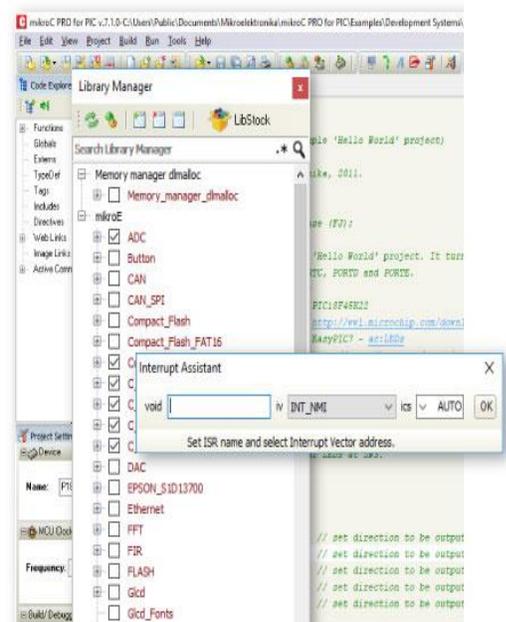


Imagen 1.2

Cuenta con:

- 1000 Funciones de biblioteca. (véase la imagen 1.3)
- 153 Ejemplos.
- 619 Microcontroladores soportados.
- Posibilidad crear interfaces de usuario gracias al TFT visual integrado.

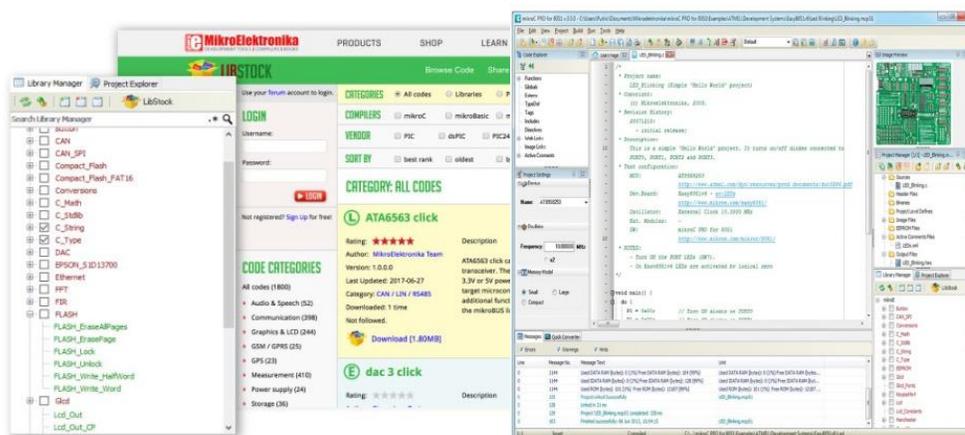


Imagen 1.3

2

Desarrollo conceptual

Este trabajo presentado, aborda la problemática que se tiene en centros sanitarios u hospitales dónde se necesita la transportación de materiales y/o equipo a una determinada zona para el uso del personal.

Aunque si bien este proyecto puede ser implementado en otros entornos, dónde se requiera el transporte de objetos o materiales a destinos específicos de una manera constante.

Este proyecto se logrará de dos maneras diferentes las cuáles van dirigidas a objetivos específicos.

- 1) Implementar el robot con una estructura electrónica externa conformada de dispositivos electrónicos como los son: transistores, relevadores, sensores. Esta estructura es también llamada *estructura electrónica numero 1*, está pensada para un transporte de materiales de manera continua es decir, que exista un tiempo de transición mínimo entre llenar la carga a llevar y la solicitud de requerimiento de material.
- 2) Implementar el robot con una estructura electrónica externa conformada de microcontroladores y sensores, Esta estructura es también llamada *estructura electrónica numero 2*, va dirigida hacia la flexibilidad y comodidad del usuario, para tener mayor control en los artículos que se enviarán a las diferentes zonas de uso.

2.1 Aplicaciones

Algunos de los entornos donde este trabajo puede implementarse, son: almacenes de industrias, oficinas, hogar.

- **Almacenes de industrias.**
Se podría implementar en el traslado de materiales desde el muelle de descarga, al almacén. También se utilizaría en la siguiente etapa, llevando del almacén hasta las diferentes zonas de implementación como, llenando las cintas ó bandas transportadoras. Algunos de estos materiales acarreados serían: líquido de frenos, refrigerantes, tornillos, material para baterías etc.
- **Oficinas.**
En este sector económico podría implementarse para enviar documentos a una cierta habitación, al igual puede ser utilizado llevando material de oficina como lápices, plumas, hojas, engrapadoras a dónde se requiera.

- **Hogar.**

Es posible utilizar este proyecto transportando todo tipo de comida y materiales de primera necesidad a habitaciones específicas donde se carece de personal que lo realice.

2.2 Ventajas

Al separar a los individuos encargados del transporte de materiales, estos pueden ocuparse de otras actividades donde se requiera con mayor importancia el personal humano.

Este proyecto se implementara con el uso de celdas solares con baterías recargables, para reducir en la manera de lo posible el impacto ambiental, al igual se utilizaran luces intermitentes para facilitar la visualización del robot.

Un objetivo que hace que este proyecto destaque sobre otros similares es que se planea llevarlo a cabo con costos de fabricación reducidos y que también pueda utilizarse en conjunto como flota, lo cual le otorga a este proyecto factibilidad y a su vez versatilidad.

2.3 Formación del proyecto

Se pretende dar una solución adaptable para una movilidad sustentable de materiales a través de la implementación de un robot del tipo móvil autómatas que se encargara de realizar esta tarea.

Lo que hará este robot es empezar en un punto establecido donde se pueda recorrer todo el pasillo, he ir dejando el material en la habitación correspondiente.

Se plantea para su análisis un corredor mostrado en la figura 2.0.

Este robot operará a nivel de piso y puede llevar cargas moderadas de 10 kg aproximadamente. Para llevar a cabo este proyecto se utilizará un microcontrolador del tipo "PIC" el cual será programado en "lengua C".



Figura 2.0

Este microcontrolador realizara todas las funciones concernientes que le permitirán al robot y a una plataforma electrónica externa llevar a cabo con éxito el transporte de materiales.

Se planea utilizar un “driver” como el” L292” ó el “UNL2003A” para la etapa de potencia, aunque es posible que se requiera la implementación de más componentes electrónicos para mejorar su desempeño.

Los motores con los que cuenta este proyecto son del tipo “Reductores o Engranajes” que son motores menos precisos, pero son cómodos de usar para el ambiente de programación utilizado, este tipo de motores ofrecen un buen torque y una buena capacidad para mover cargas pesadas.

Se pretende implementar un control de velocidad basado en el valor analógico leído por un potenciómetro para ajustar la velocidad, para una buena optimización y funcionamiento.

Poniendo como ejemplo el pasillo de la figura 2.1.



1. El robot empezará en la posición de inicio apagado.
2. El personal que requiera material en alguna habitación de dicho pasillo activará una alarma que envíe una solicitud al cuarto de control.
3. Después el robot se moverá hacia adelante.
4. Al encontrar el marco de la puerta girará a la derecha, hasta encontrar el cuarto correcto.
5. Ahí tendrá que moverse de frente, desactivar la carga y dejarla adentro de la habitación.
6. Después se moverá hacia atrás.
7. Al encontrar el marco de la puerta tendrá que moverse a la izquierda, hasta encontrar el punto de inicio.
8. De ahí se moverá hacia atrás.
9. Asegurar otra carga y se moverá de frente pero ahora en dirección a otro destino donde se requiera el material.

Lo anterior mencionado se podrá lograr con las dos diferentes estructuras eléctricas externas, estas brindaran las señales que faciliten el posicionamiento continuo del robot, también por obvias razones se necesita un algoritmo de programación para interpretar dichas señales.

Figura 2.1

Entonces de forma general el proyecto consiste de dos partes:

1. Una estructura electrónica externa la cual brinda las restricciones y el control correspondientes al robot. Esta estructura se elegirá de las dos posibles, con respecto a la arquitectura y a las necesidades del complejo donde se instalará.
2. El robot que se enfocará en el transporte del material.

2.3.1 Implementación de las estructuras electrónicas externas

Las infraestructuras eléctricas se implementaran con base en la figura 2.2.

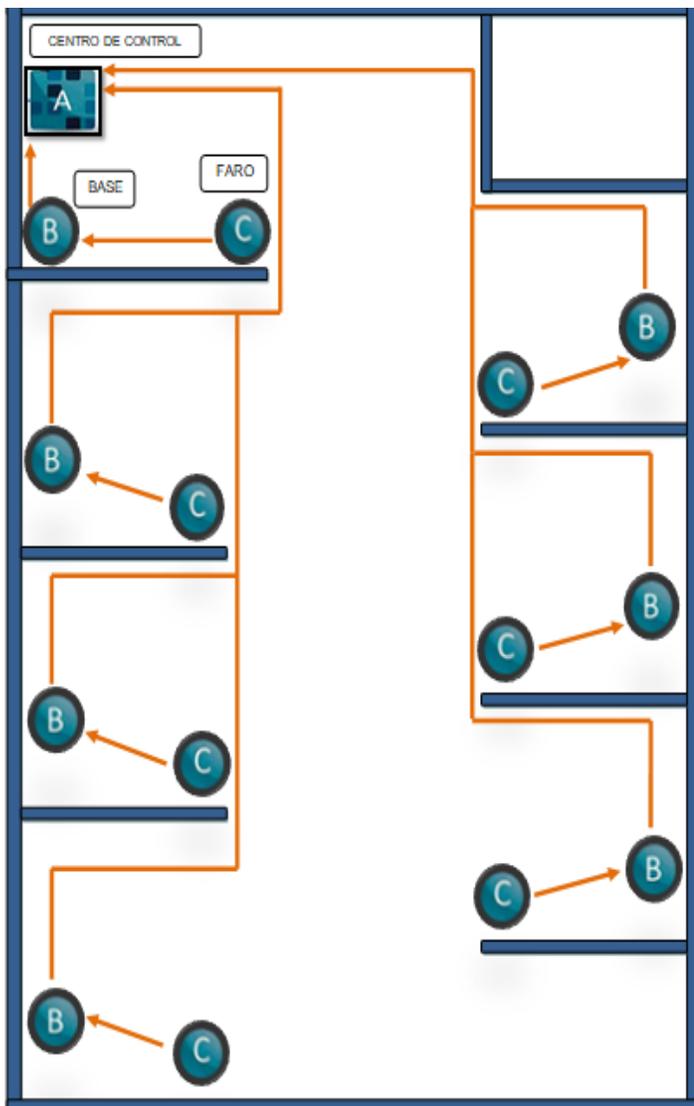


Figura 2.2

La primera parte "*Centro de Control*" es la cual recibirá toda la interconexión con la demás partes de la estructura y será de donde partirá el robot.

El apartado número dos $Base_x$ corresponde a un conjunto de sensores emisores $Alarma_x$ y una pareja de sensores emisor-receptor "*Detector_x*".

La tercera parte corresponde al "*Faro_x*" al igual que el apartado anterior estará constituida por un grupo de sensores emisores y una pareja de sensor emisor-receptor.

Se le agregan subíndices ya que existirá uno diferente por cada habitación, pero el comportamiento es el mismo.

Para el apartado *Centro de Control*, este se llevará a cabo mediante el uso de un panel, este panel mostrará el cuarto del pasillo donde se requiera material a través del uso de "leds". Para el panel correspondiente con la estructura electrónica 1, éste contará con un led que indicará la necesidad de material en alguna habitación. (véase la figura 2.3).

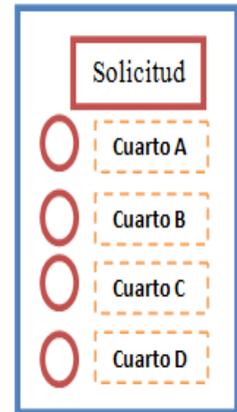


Figura 2.3

Para el panel correspondiente con la estructura electrónica 2, éste contará también con un led que mostrará la necesidad de material en alguna habitación, pero también contará con un botón por cada cuarto que al ser presionado le indicará al robot que habitación requiere entrega de material. (véase la figura 2.4).

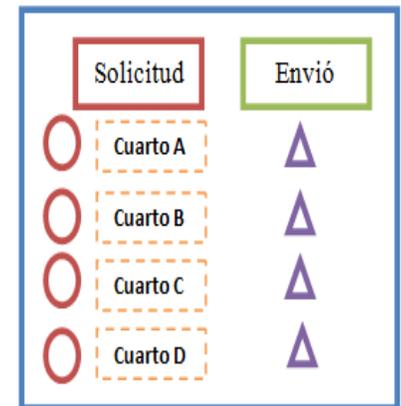


Figura 2.4

Para el apartado de $Base_x - Alarma_x$ esta se montará en la pared que se encuentre enfrente de la puerta de la habitación. Mientras que el apartado $Base_x - Detector_x$ se acoplará en el piso de la respectiva habitación a una distancia no muy lejana del sensor $Alarma_x$. (véase la figura 2.5). Para la sección de $Faro_x$ se instalará sobre el marco de la puerta como lo muestra la figura 2.6. De manera conjunta ambas secciones. (véase la figura 2.7).

Cabe mencionar que para la parte de "*Centro de Control*" también contará con su $Alarma_x$, $Detector_x$ y $Faro_x$ que llamaremos $E1$, $E2$ y $E3$ respectivamente.

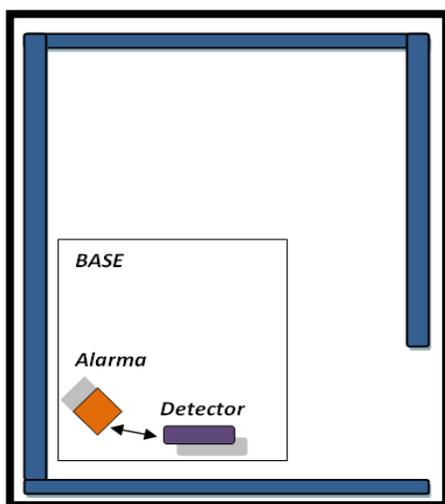


Figura 2.5

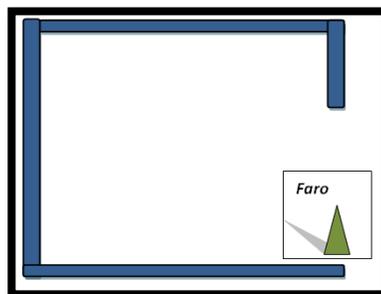


Figura 2.6

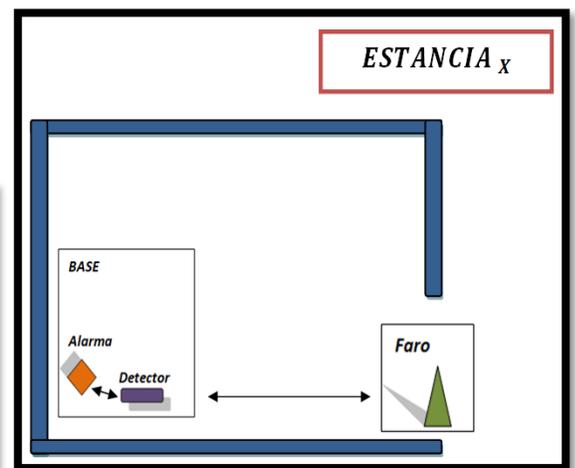


Figura 2.7

3

Elementos de las estructuras electrónicas externas

Estas dos estructura electrónica estarán conformadas por varios elementos electrónicos como lo son los siguientes:

- ❖ Infrared Optical Sensor CNY 70 (Similares).
- ❖ Transistores tipo NPN 2N222A.
- ❖ Relevadores SUN HOLD (RAS - 0510 de 5 Volts DC).
- ❖ Microcontroladores(PIC).

Será pertinente una breve introducción del funcionamiento de éstos dispositivos que se dará a continuación.

3.1 Sensores

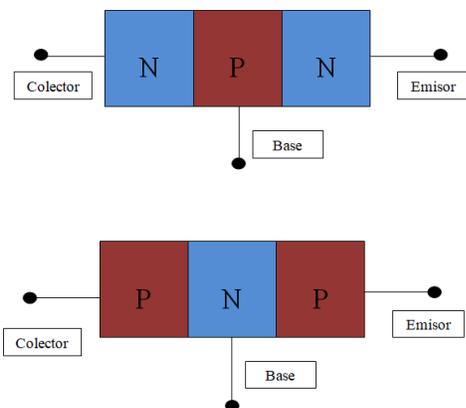


El término sensor se refiere a un elemento que produce una señal relacionada con la cantidad que se está midiendo.

Regularmente se utiliza con mayor frecuencia la palabra *transductor*, que es un elemento que al someterlo a un cambio físico experimenta un cambio relacionado proporcionalmente. (véase la imagen 3.0).

Imagen 3.0

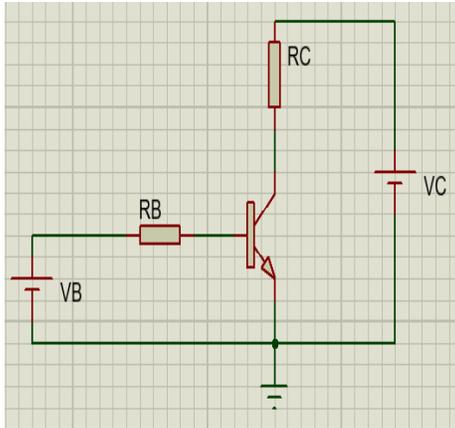
3.2 Transistores



Son dispositivos *semiconductores*, esto quiere decir que es un dispositivo que dependiendo su polarización se comportará como un material aislante (madera, mica) y/o como un material conductor (oro, plata, cobre).

Figura 3.1

Figura 3.2



Está compuesto por la unión de finas láminas de material P (exceso de carga positiva) con láminas de material N ó mayoría de electrones, en diferentes configuraciones. (véase la figura 3.1).

Al polarizar el transistor como se muestra en la figura 3.2, donde VC es mucho mayor a VB, se genera una corriente que circula entre las terminales base-emisor, sin embargo esta corriente crea un efecto de

“avalancha” haciendo que la corriente de la base se multiplique varias veces estableciendo una corriente elevada entre el emisor y el colector.

Básicamente el transistor tiene dos funciones la primera deja pasar señales eléctricas a partir de una señal de mando es decir, funciona como un interruptor controlado y la segunda función es que se puede utilizar como elemento amplificador de señales.

Para éste desarrollo se utilizará al transistor como interruptor.

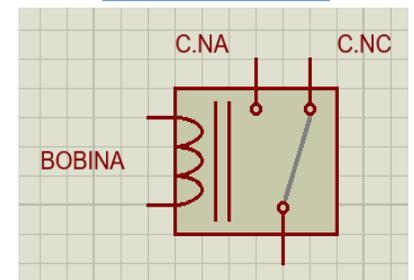
3.3 Relevadores

El relé o relevador es un dispositivo electromagnético, que funciona como un interruptor que se controla eléctricamente.

Este dispositivo está constituido de forma generalizada por tres partes. Un contacto normalmente abierto (C.NA), un contacto normalmente cerrado (C.NC) y una bobina, (véase la figura 3.3).

Cuando circula corriente a través de la bobina, esta se excitará, cambiando de posición sus contactos, el contacto normalmente cerrado pasará a ser un contacto abierto y el contacto normalmente abierto, pasará a ser un contacto cerrado. Este efecto será anulado cuando la corriente que éxito a la bobina deje de fluir.

Figura 3.3



Esté dispositivo es muy útil cuando se utiliza en su modo de auto enclave, (véase la figura 3.4). Esta configuración sirve para suministrar corriente a otra parte del circuito sin la necesidad de mantener siempre presionado el botón de arranque y se desactivará cuando se oprima botón de paro.

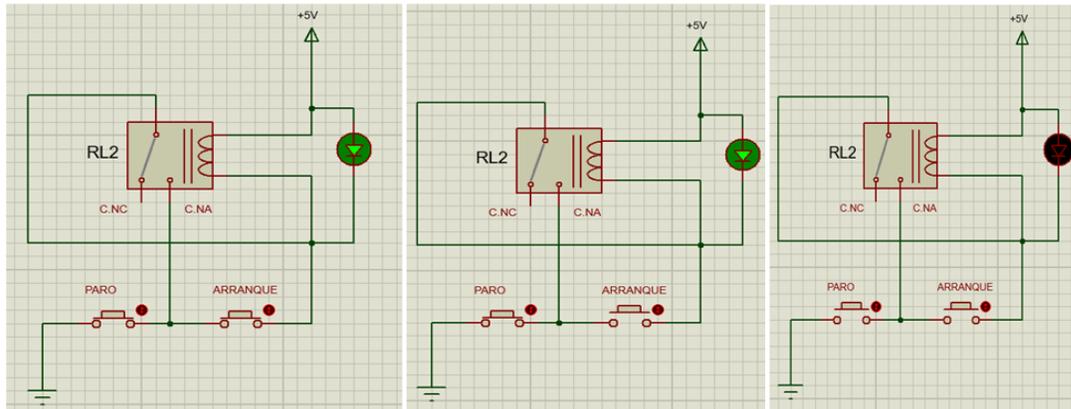


Figura 3.4

3.4 Fototransistores

Son dispositivos *semiconductores* que tienen una unión base-colector P-N sensible a la luz.

Cuando la luz no incide, la corriente colector emisor es muy pequeña, pero al incidir la luz se produce una corriente de base proporcional a la intensidad luminosa, debido a ello se produce una corriente de colector que es una medida de la intensidad luminosa. (véase la imagen 3.5).

Imagen 3.5



3.5 Microcontroladores

Definido de forma generalizada un microcontrolador es un circuito integrado capaz de realizar diversas tareas lógicas, estas tareas son escritas o programadas en su memoria y solo puede tener cargado un conjunto de funciones o programa en dicha memoria.

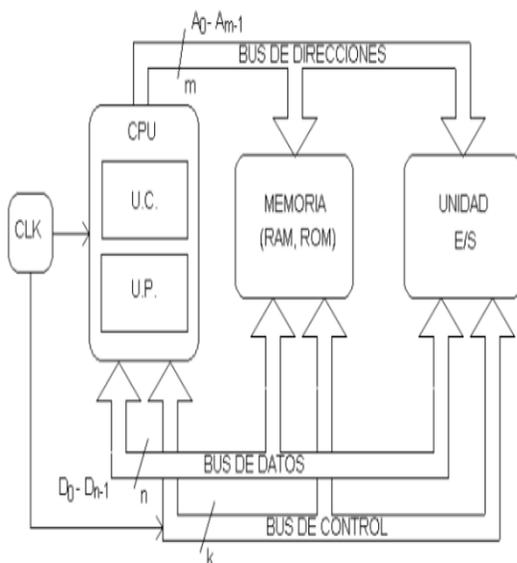
Otra interpretación de un microcontrolador es, un dispositivo lógico de propósito general formado por la unión de otros circuitos como los son, decodificadores, contadores, comparadores, registros, flip-flops, etcétera.

Los primeros microcontroladores creados como lo son el del tipo "PIC" necesitan tanto un software como un hardware para poder ser utilizados. Este hardware fundamental que debe de llevar cada microcontrolador PIC, se puede definir en tres elementos:

- 1) Circuito de reloj.
- 2) Módulos de entradas y salidas.
- 3) Fuente de alimentación.

3.5.1 Estructura del microcontrolador

La estructura general de un microcontrolador es la que se muestra en la imagen siguiente.



Unidad E/S: Dispositivos especializados en mandar y recibir información entre el CPU y el mundo exterior.

Memoria: Circuitos donde se almacenan los códigos de las instrucciones del programa y los datos.

CPU: (Central Process Unit), realiza las operaciones lógicas y aritméticas previstas en el repertorio de las instrucciones del microcontrolador.

4

Estructura electrónica externa 1

El algoritmo que seguirá esta estructura para cumplir de forma óptima su misión de acotar el comportamiento del robot es la siguiente:

- A) La $Alarma_x$ se enciende manualmente (esto quiere decir que se requiere material en alguna habitación)
- B) Al encender $Alarma_x$ se enciende también $Detector_x$
- C) Cuando $Detector_x$ reciba presencia (reflejo o conmute) apagar $Alarma_x$ y activara $Faro_x$
- D) Cuando $Faro_x$ conmute encenderá $E1, E2$ y volverá a habilitar la opción de activar de manera manual la $Alarma_x$ (reiniciar)
- E) Al conmutar $E2$ se desactivará $E1$ y encenderá $E3$
- F) Al reflejar $E3$ se reiniciará $E1$ y $E2$

El diagrama eléctrico que satisface las condiciones antes mencionadas es el siguiente.

4.1

Circuito de enclave

Al presionar el switch principal el circuito se cerrará y permitirá excitar a la bobina número uno (B1).

Con lo cual ocasionará que el contacto normalmente abierto uno (C.N.A.1) pase a ser un contacto cerrado brindando otro camino para cerrar el circuito, sin la necesidad de mantener el "Switch Principal" presionado.

Este circuito se desactivará al energizar la bobina del relevador B5, lo que hará que el contacto normalmente cerrado cinco (C.N.C.5) pase a ser un contacto abierto. Este circuito eléctrico sirve como fuente de alimentación controlada para el resto del circuito. (véase la figura 4.0).

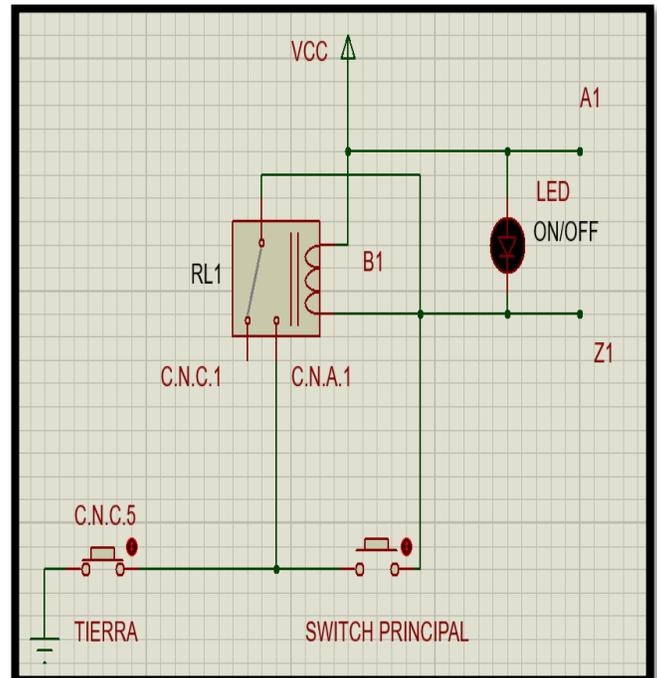


Imagen 4.0

En la parte numero dos del circuito, al tener polarizado las borneras A1 y Z1, se energizará los sensores “Alarma” y “Detector”.

“Alarma” se apagará cuando la bobina B2 se polarice y por consecuencia el contacto normalmente cerrado (C.N.C.2) pase a ser un contacto abierto.

Cuando el sensor “Detector” se excite cerrará su transistor interno y dejará pasar corriente a través de Y1. Siempre y cuando el sensor “Detector” refleje o conmute. (véase la figura 4.1).

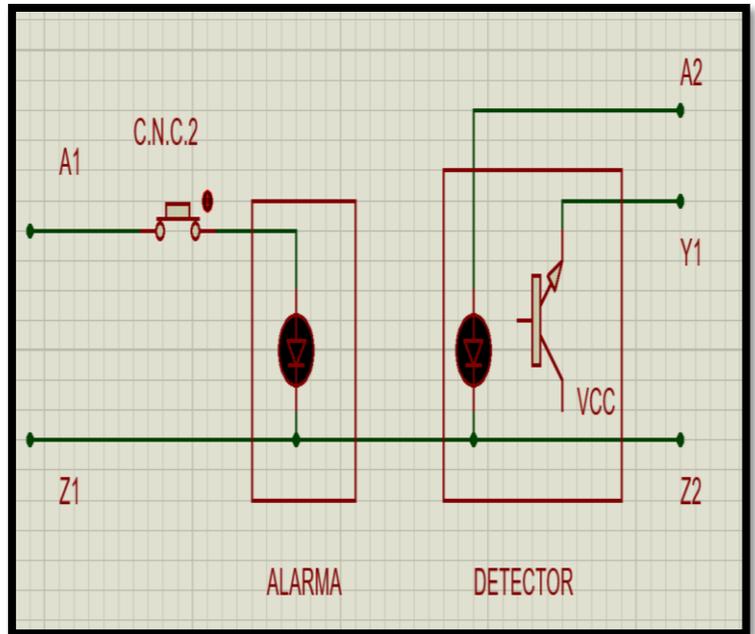


Figura 4.1

4.2

Circuito de enlace con relevador y transistor

Para la tercera parte del circuito eléctrico: cuando el transistor T1 reciba corriente en su base, dicho de otra manera que exista corriente eléctrica en Y1, el transistor conmutará.

Con lo cual las bobinas B3 y B2 se energizaran. Cerrado a su vez el C.N.A.3 y enclavando el sistema, polarizando las borneras A3 y Z3. Sin importar si la corriente en Y1 deja de fluir, lo que se traduce a que el transistor T1 deje de conmutar en colector-emisor. (véase la figura 4.2).

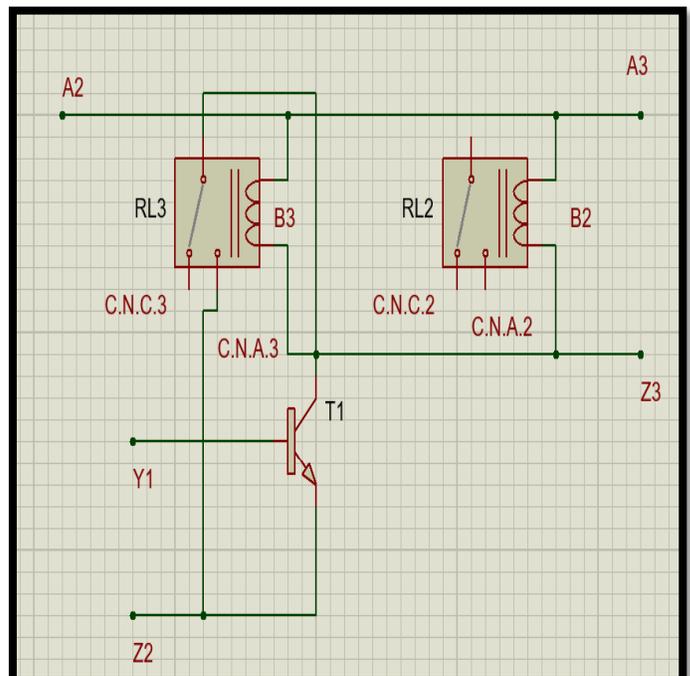


Figura 4.2

La parte número cuatro del circuito es muy parecida a la anterior, al polarizar las borneras A3 y Z3 se podrá energizar el sensor "Faro".

Cuando el sensor "Faro" conmute habrá flujo de corriente en Y2, con lo cual excitará al transistor T2 haciéndolo que conmute. Esto hará que se cierre el circuito y por ende las bobinas B4 y B5 reciban corriente.

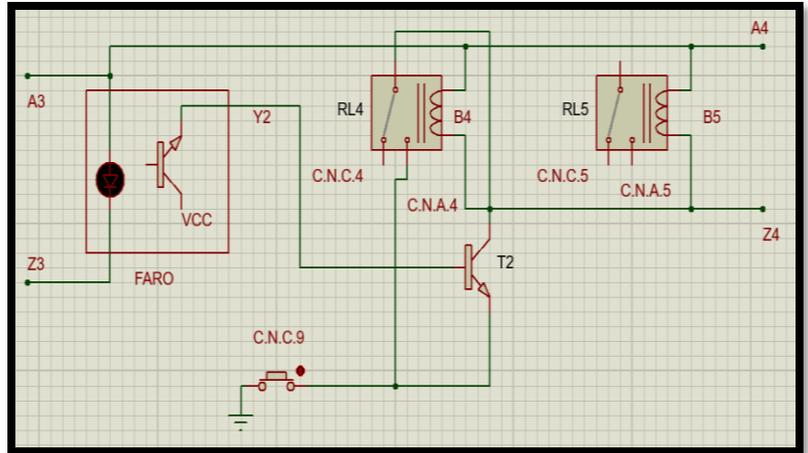


Figura 4.3

Cuando la bobina B4 reciba corriente, enclavará el sistema permitiendo polarizar las borneras A4 y Z4. (véase la figura 4.3).

Al tener polarizadas las borneras A4 y Z5, en la parte número cinco del circuito se podrán activar los sensores E1 y E2 respectivamente.

Cuando el sensor E2 detecte presencia de algún objeto o conmute permitirá la existencia de corriente en Y3 con lo cual excitará el transistor T3. Permitiendo el flujo de corriente hasta Z4 autorizando de nuevo cerrar el circuito. Esto energizará la bobina B6 enclavando el sistema.

A la par que lo anterior ocurre, se polarizará la bobina B7, es decir el contacto normalmente cerrado (C.N.C.7) se convertirá en un contacto abierto desactivando el sensor E1. (véase la figura 4.4).

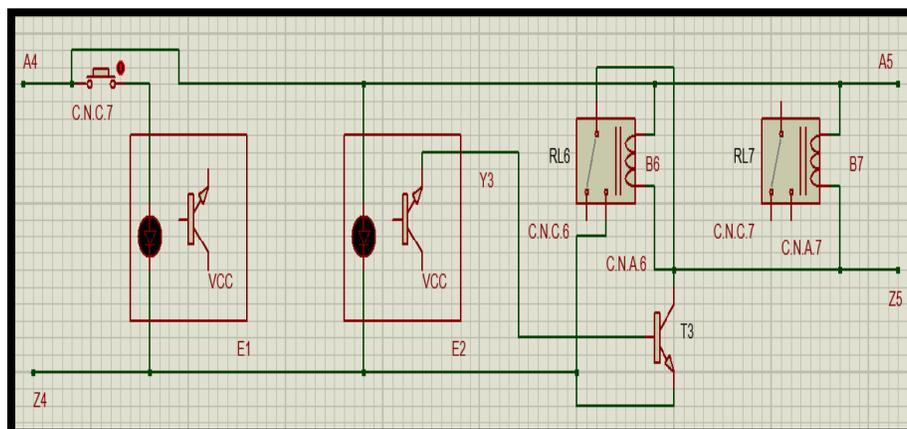


Figura 4.4

Al tener polarizadas las borneras A5 y Z5 en la última parte del circuito se podrá activar el sensor E3.

Cuando este sensor conmute, en la bornera Y4 habrá flujo de corriente, por ello excitará al transistor T4, cerrando el circuito y enclavando el sistema a través de la bobina B8.

Esto hará que también la bobina B9 se polarice, a su vez propiciara que toda la estructura del circuito eléctrico se reinicie. (véase la figura 4.5).

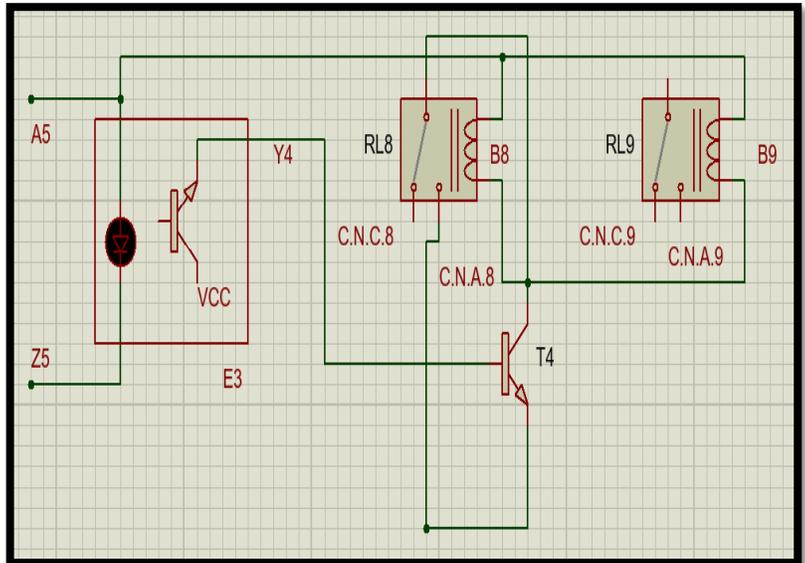
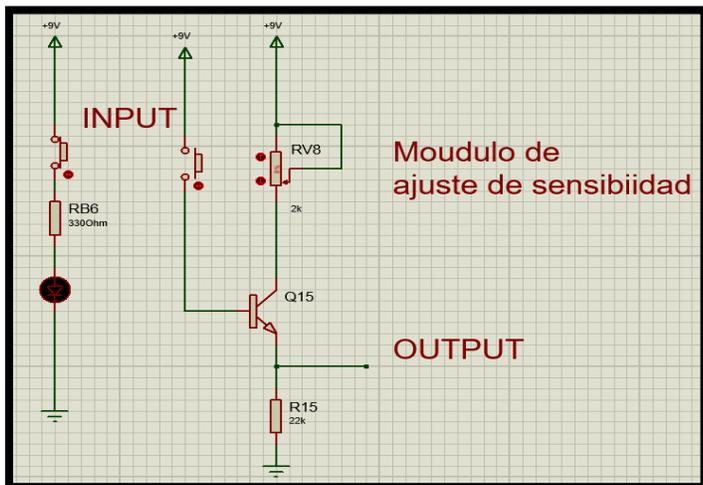


Figura 4.5

4.3 Simulación

Para la simulación de esta estructura se plantean dos problemas. 1) Toda la estructura antes mencionada solo esta de forma representativa y de fácil comprensión, ya que el sensor que se utilizara "CNY 70" o similares, no está disponible en el software empleado. 2) La estructura solo muestra su aplicación en una habitación y en el cuarto de control, para agregar esa estructura a múltiples habitaciones se necesita agregar un módulo.

Para la primera parte, se implemento un arreglo que asemeja el comportamiento que se tendría utilizando este tipo de sensores, como se muestra en la figura 4.6.



El arreglo consta de:

- 1) Un potenciómetro de 2 kΩ, que ayudan a ajustar la sensibilidad del sensor.
- 2) Una resistencia de 330Ω para no dañar el sensor.
- 3) Un "Push Button" que al ser presionado realiza la simulación de cuando el sensor conmute.

Figura 4.6

Este tipo de sensores funcionan en su mayoría con dos leds infrarrojos, uno emisor y otro receptor, en caso de que el led receptor reciba señal, significa que existe un objeto frente a los sensores, con esto el transistor que forma parte de su estructura interna, entrará a la zona de saturación y se comportará como un switch dejando circular una corriente del colector hacia el emisor.

Para la segunda parte se agregara el siguiente modulo.(véase la figura 4.7).

Donde $X_1 - X_2$ representan un cuarto, al igual que $y_1 - y_2$. En $Z_1 - Z_2$ representan las salidas al cuarto de control.

Este arreglo de diodos es fundamental para que la estructura eléctrica funcione de manera óptima y que sin ella, una señal podría activar otra secuencia no deseada.

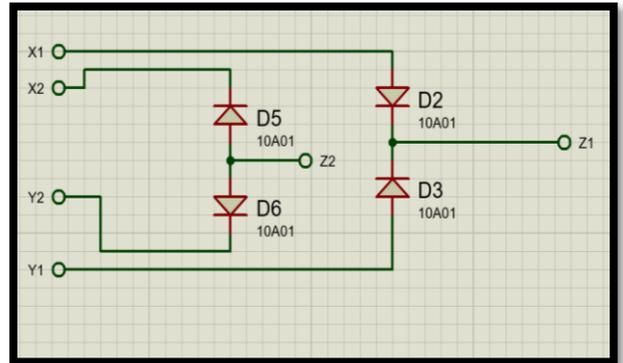
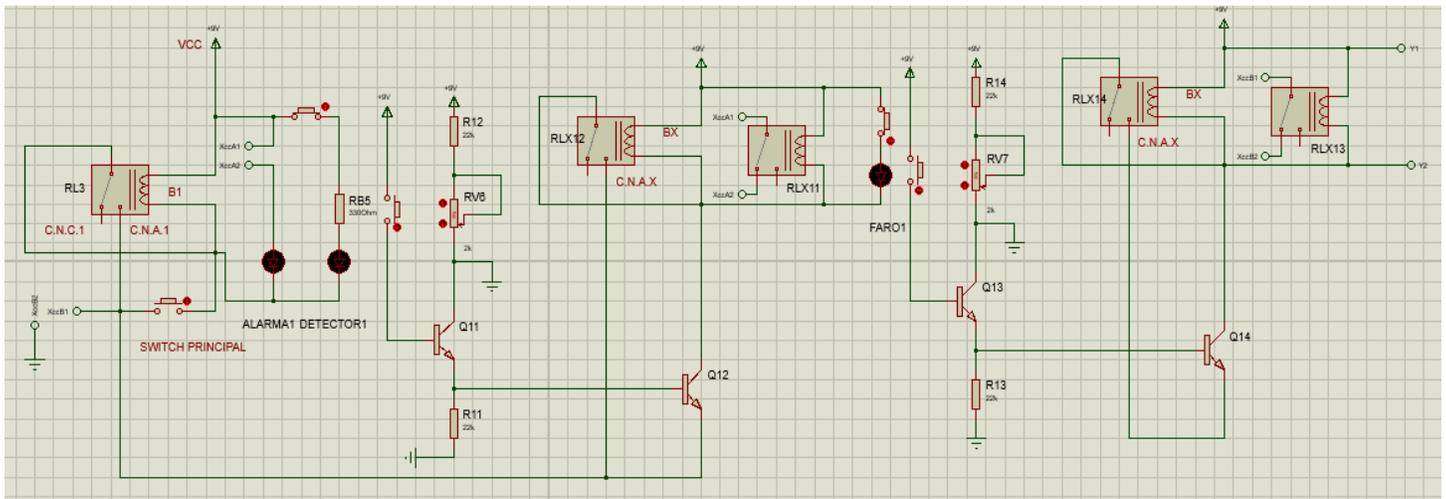
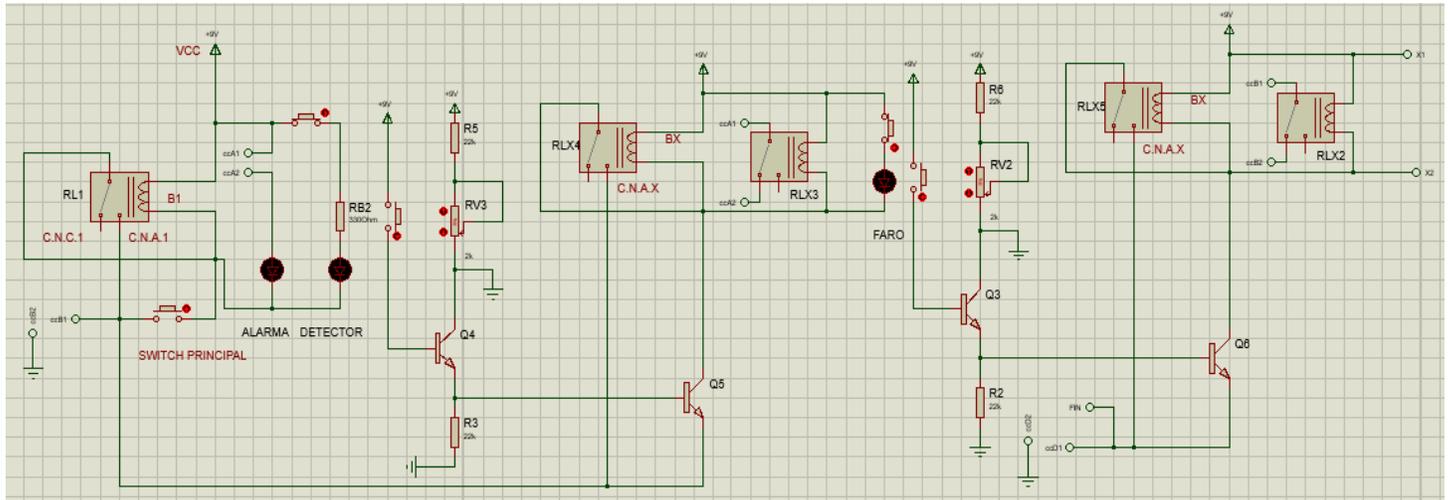


Figura 4.7

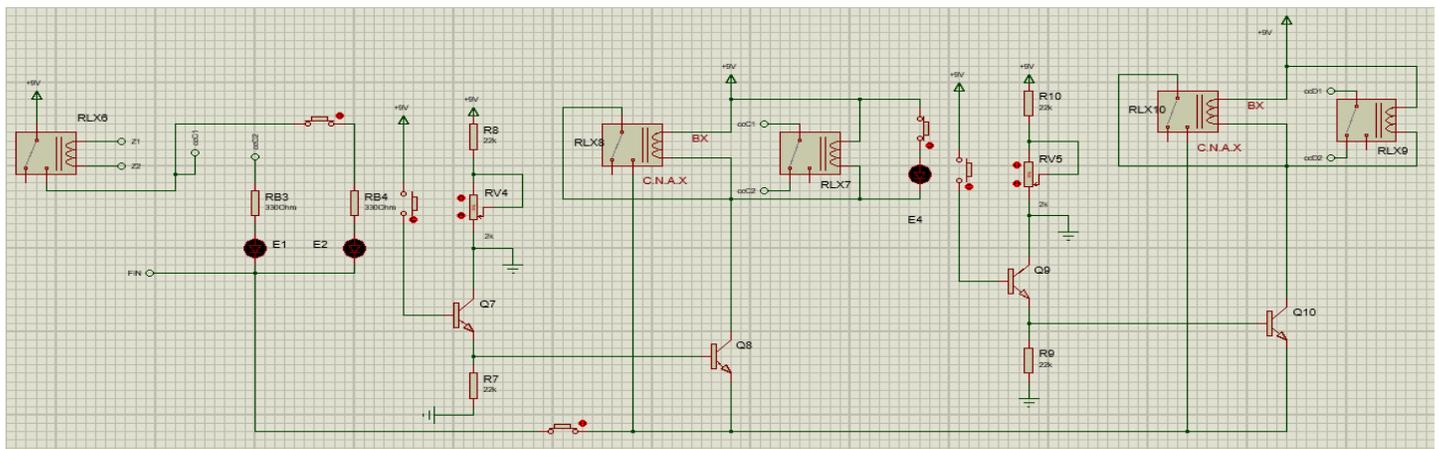
El circuito con todos los elementos unidos que representa la habitación "x" es el siguiente.



El circuito con todos los elementos unidos que representa la habitación "y" es el siguiente.

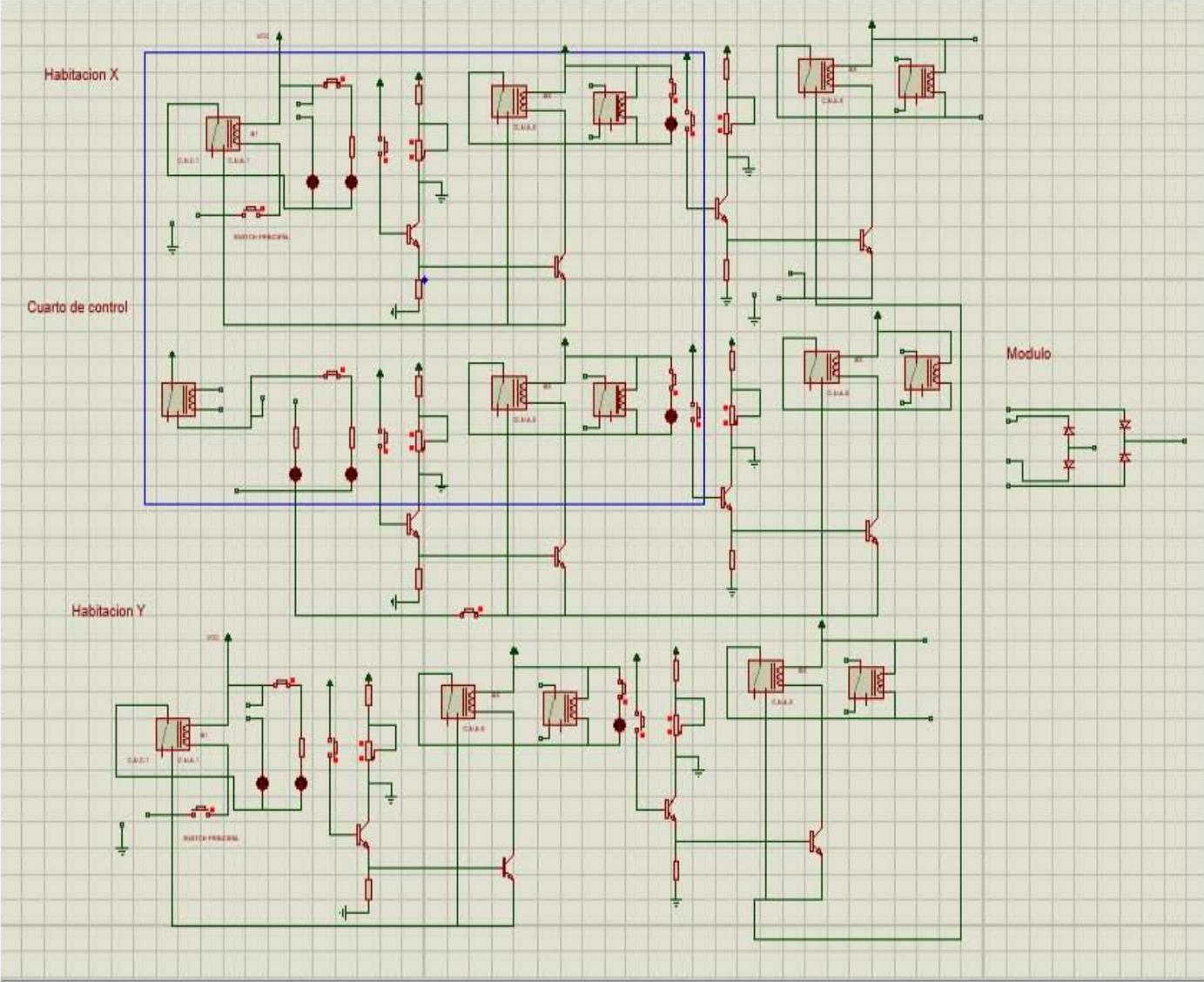


El circuito con todos los elementos unidos que representa el cuarto de control es el siguiente.



4.3.1

Circuito completo



5

Estructura electrónica externa 2

La anterior estructura electrónica se puede implementar con un microcontrolador del tipo *Pic*, más fácilmente. Esta nueva estructura ofrece mayores comodidades como lo son: la utilización de un microcontrolador por piso, mayor fiabilidad, reducción de número de elementos utilizados y reducción de tiempo de alambrado.

Lo que se debe hacer es estructurar un código de programación que satisfaga las condiciones mencionadas al inicio de este tema pero ahora con delimitaciones extras:

- A) La *Alarma_x* se enciende manualmente (esto indica que se necesita material en alguna habitación)
- B) Al encender *Alarma_x* se enciende también *E3* y *Detector_x*.
- C) Cuando *Detector_x* reciba presencia (refleje o conmute) apagar *Alarma_x* y activara *Faro_x*.
- D) Cuando *Faro_x* conmute, prenderá *E1, E2* y apagará *E3*
- E) Al conmutar *E2* se reiniciará todo el sistema.

Para lograr esto, se debe comprender que se utilizarán sensores los cuales necesitan una señal de activación para que funcionen, dicha señal será dada por el microcontrolador a través de la señal principal de control *HX*, esto delimitará a los sensores (los correspondientes al cuarto “A” o los correspondientes al cuarto “B” etc.) a su vez estos sensores brindarán señales de salida cuando estos conmuten (reflejen su señal a través de un objeto) esta señal de salida será también ocupada por el microcontrolador como entrada para brindar una señal de activación al resto de los sensores (*Detector_x*, *Faro_x*, *E1*, *E2*, *E3*). La figura 5.0 expresa lo anteriormente mencionado.

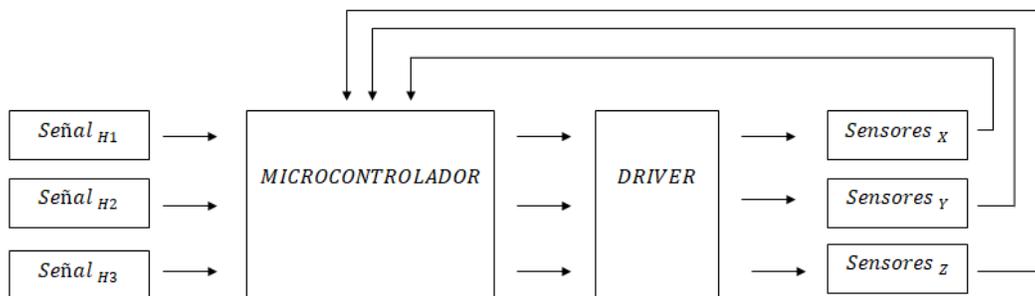


Figura 5.0

5.1 Código

El código que sigue el microcontrolador para que la estructura anterior mencionada funcione es el siguiente:

```
void ca() //Funciones para activar los sensores correspondientes
{
  portc=0b00000011;portd=0b00000010;
}
void cb()
{
  portc=0b00000110;portd=0b00000011;
}
void cc()
{
  porte=0b00000011;portc=0b00000000;portd=0b00000000;
}
void ce()
{
  portc=0b00011000;portd=0b00000010;
}
void cf()
{
  portc=0b00110000;portd=0b00000110;
}

void main()
{
  short a; //Variable de rango - 128 .. 127
  short K;
  short L;
  PORTA=0; // Todos los pines del puerto A se ponen a cero
  PORTB=0; // Todos los pines del puerto B se ponen a cero
  PORTC=0; // Todos los pines del puerto C se ponen a cero
  PORTD=0; // Todos los pines del puerto D se ponen a cero
  PORTE=0; // Todos los pines del puerto E se ponen a cero
  ANSELH=0; // Configura pins como entradas digitales
  ANSEL=0;
  TRISA=1; //Puerto A como entrada
  TRISB=1; //Puerto B como entrada
  TRISC=0; //Puerto C como salida
  TRISD=0; //Puerto D como salida
  TRISE=0; //Puerto E como salida

  while(1) //Para anclar el programa
  {
    while(PORTA.RA0 == 0) //Mientras el pin 0 del puerto A sea 0 realizas lo siguiente FUNCIONA COMO ON/OFF
    {
      K=0;
      L=0;
      while(PORTA.RA1 == 1) //Mientras el pin 1 del puerto A sea 1 realizas lo siguiente
      {
        //sirve para seleccionar las instrucciones del cuarto X
        while((PORTB.RB0 == 0) && (K == 0)) //Mientras el pin 0 del puerto B sea 0 y k sea igual a 0
        {
          //llamas a la funcion "ca" durante el todo el ciclo "for"
          for (a=0;a<4; a=a+1)
          {
            ca();
            delay_ms(5); //Retraso de 5 milisegundos
          }
        }
      }
    }

    while((PORTB.RB0 == 1) && (PORTB.RB1 == 0)) //Mientras el pin 0 del puerto B sea 1
    {
      K=1; // y el pin 1 del puerto B sea 0 //llamas a la funcion "cb" durante el todo el ciclo "for"
      for (a=0;a<4; a=a+1)
      {
        cb();
        delay_ms(5);
      }
    }

    while((PORTB.RB0 == 0) && (PORTB.RB1 == 1) && (PORTB.RB2 == 0))
    {
      for (a=0;a<4; a=a+1)
      {
        cc();
        delay_ms(5);
      }
    }
  }
}
```

```

• while(PORTA.RA2 == 1) //Mientras el pin 2 del puerto A sea 1 realizas lo siguiente
• { //sirve para seleccionar las instrucciones del cuarto Y
•   while((PORTB.RB3 == 0) && (L == 0))
•   {
•     for (a=0;a<4; a=a+1)
•     {
•       ce();
•       delay_ms(5);
80     }
•   }
•   while((PORTB.RB3 == 1) && (PORTB.RB4 == 0))
•   {
•     L=1;
•     for (a=0;a<4; a=a+1)
•     {
•       cf();
•       delay_ms(5);
•     }
90   }

```

```

• while((PORTB.RB3 == 0) && (PORTB.RB4 == 1) && (PORTB.RB2 == 0))
• {
•   for (a=0;a<4; a=a+1)
•   {
•     cc();
•     delay_ms(5);
•   }
• }
• }
100 }

```

5.2 Simulación

Para la simulación de ésta estructura electrónica se uso el software *Proteus*. Se implementó un arreglo de transistores que simularán el comportamiento de los sensores ya que este software no dispone de estos dispositivos (véase la figura 5.1).

Este arreglo funciona de manera manual, el sensor emisor y receptor se activarán a través de la señal del *Driver*, esta señal será brindada por el microcontrolador, cuando el push button sea presionado significará que existe conmutación en el sensor CNY70 o similares.

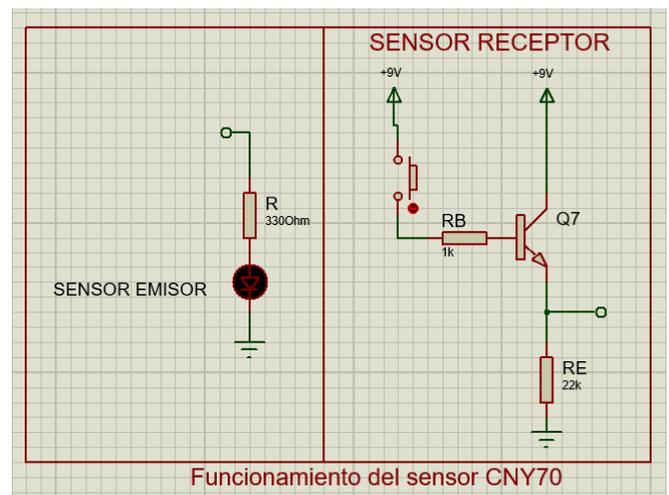
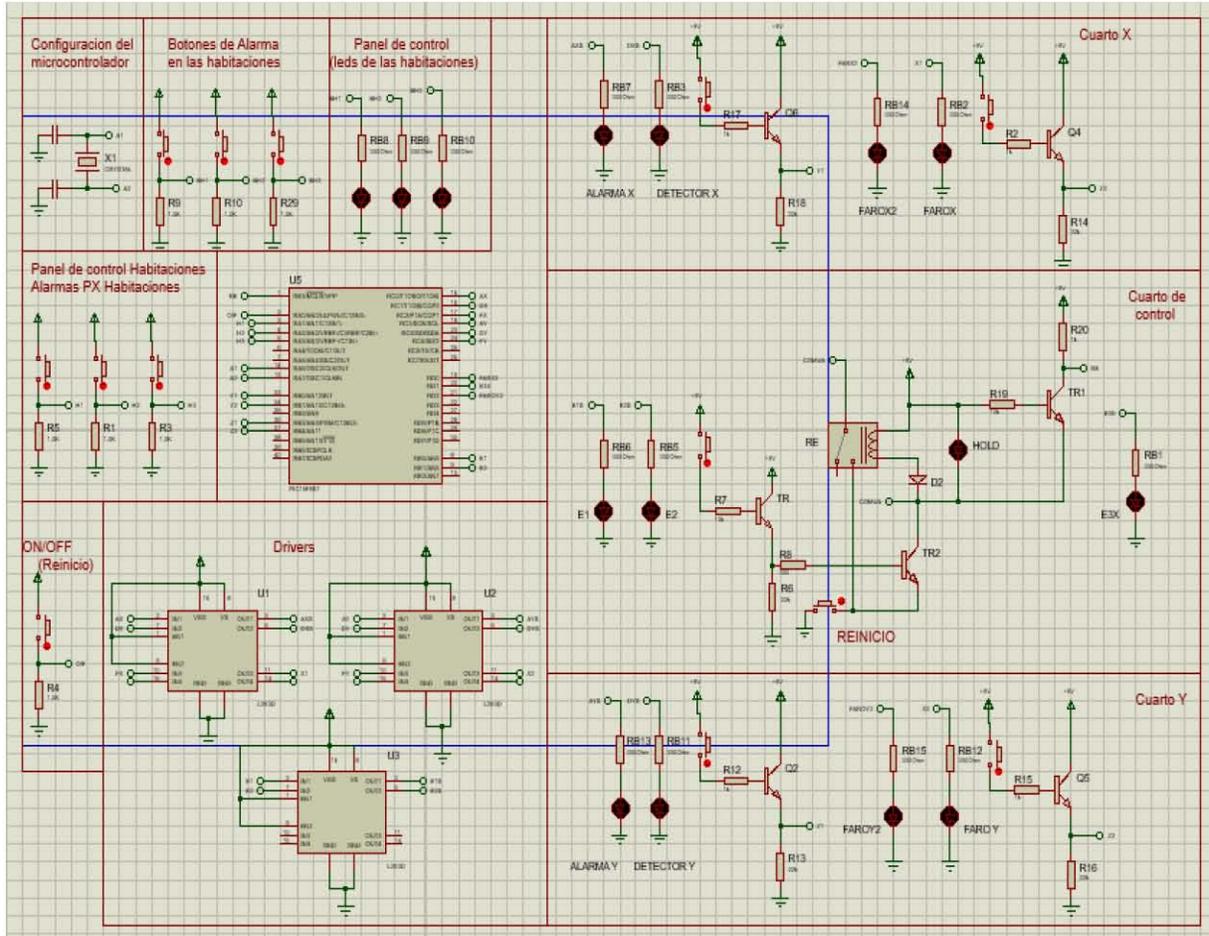


Figura 5.1

5.2.1

Circuito completo



6 Designación de la estructura electrónica externa

Para optar por una de las dos posibles estructura electrónicas externas es necesario comparar las ventajas y desventajas de cada una.

Estructura electrónica externa 1		Estructura electrónica externa 2	
Ventajas	Desventajas	Ventajas	Desventajas
Los componentes son muy conocidos en el ámbito de la ingeniería electrónica desde hace bastantes años y por ende su funcionamiento es intuitivo.	Su implementación es bastante robusta e impráctica.	Implementación práctica, ligera y de fácil modificación.	Requiere tener conocimiento en la programación de microcontroladores.
No se necesita tener conocimientos en la programación de microcontroladores.	Requerimiento de usar varias fuentes de voltaje simultáneamente.	Uso de solo dos fuentes de alimentación.	

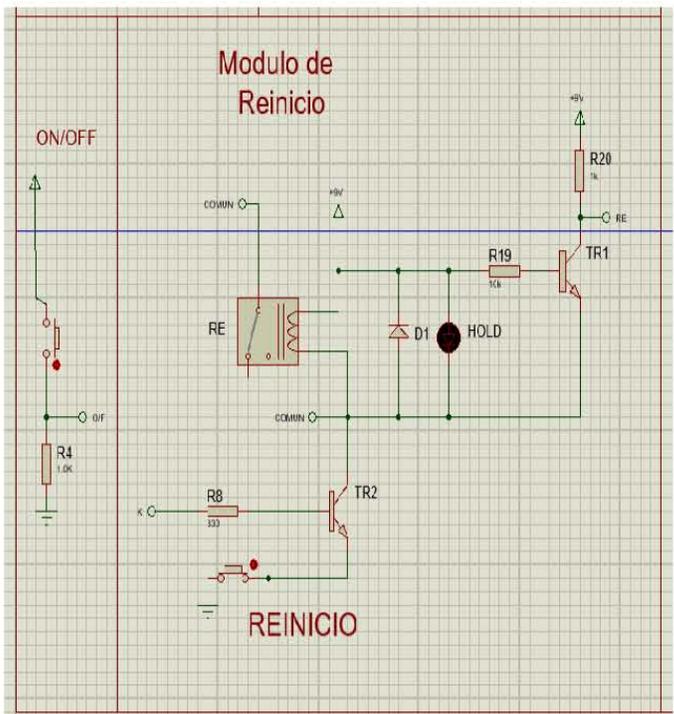
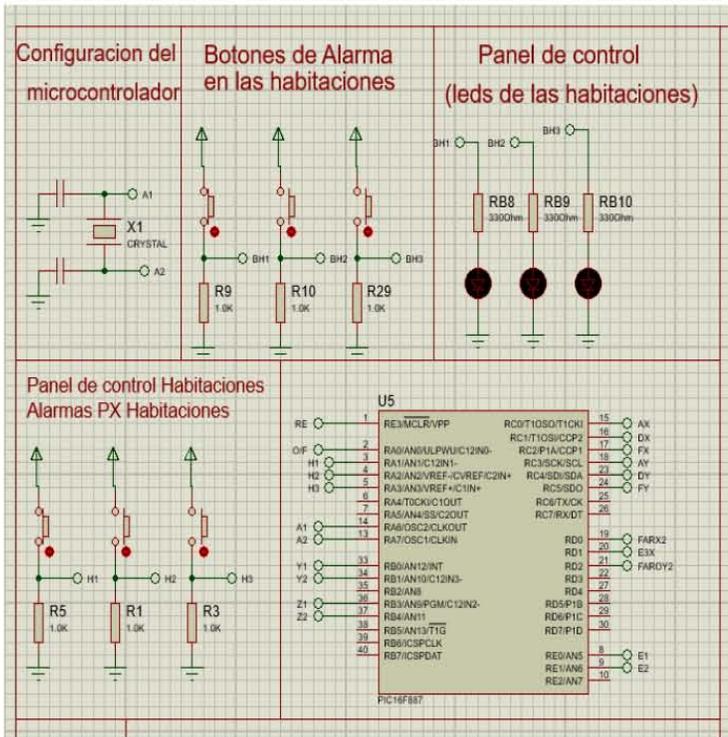
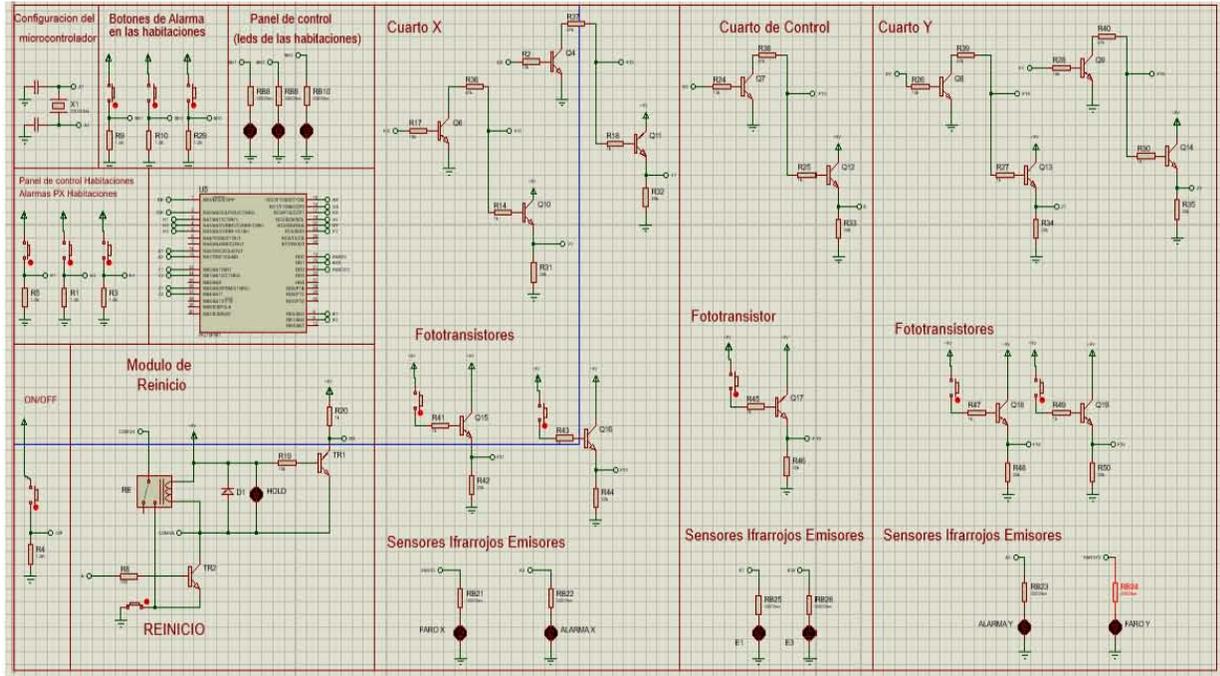
Al analizar la tabla anterior se concluye que la mejor opción de implementación para este caso, es la segunda estructura electrónica.

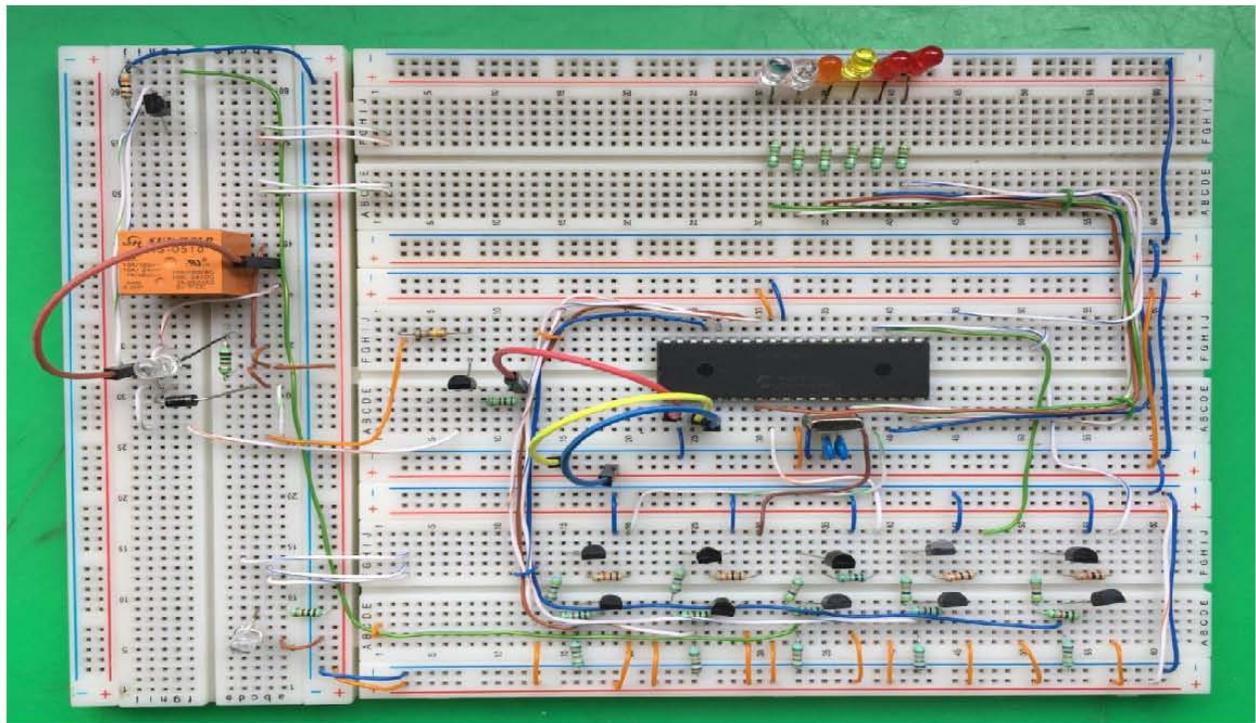
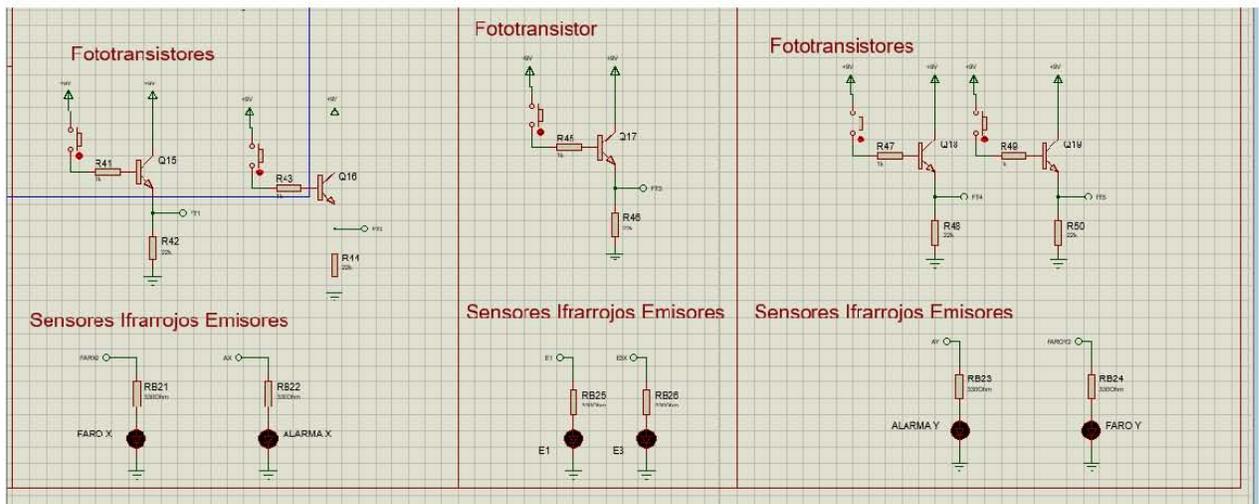
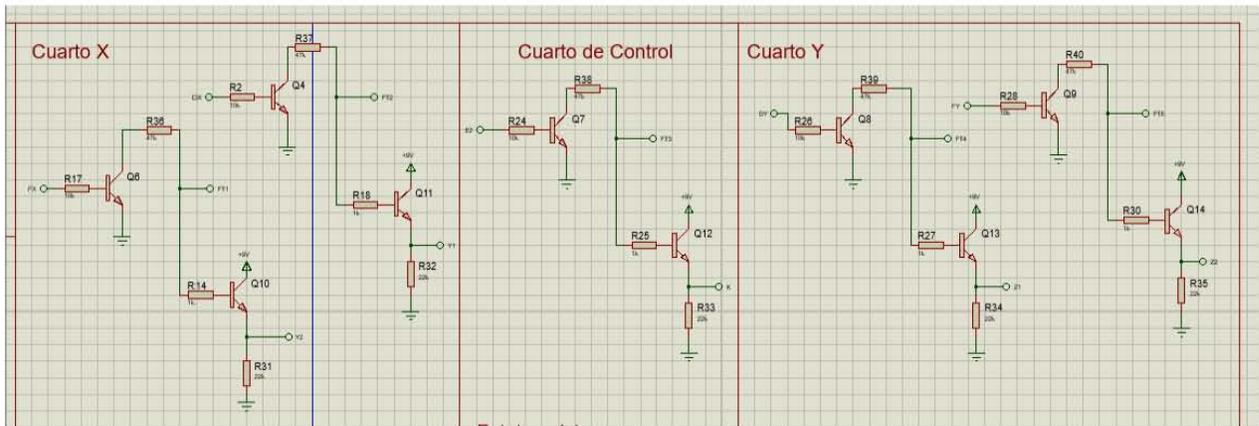
6.1 Implementación física

Al realizar dicha implementación física se notaron varias discrepancias entre la simulación y la práctica. Primeramente la simulación está pensada para una implementación de alta potencia esto quiere decir que está realizada para componentes cuyos voltajes de operación oscilan entre 15 y 20 volts, esto para asegurar la transportación de materiales pesados. Segundo, en la implementación física se opto por hacerla de forma representativa de fácil comprensión, esto para asegurar que el sistema funciona correctamente y que podrá operar con compontes más industriales.

6.2

Diagrama eléctrico integro



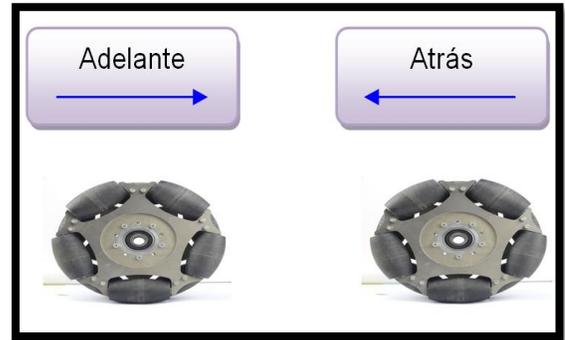


7 Robot móvil

Figura 7.0

Como primera instancia, debemos averiguar con que arreglo, el juego de ruedas puede ir en las diferentes direcciones.

Una rueda solo puede girar en dos sentidos, adelante y atrás, esto se debe a que un motor sólo puede girar en esas dos direcciones debido al sentido de la corriente. (véase la figura 7.0).

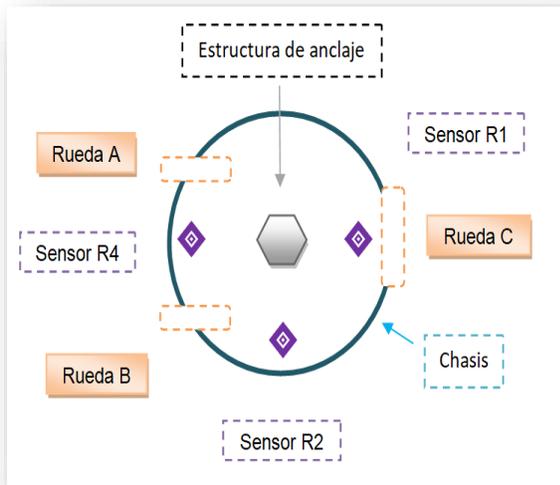


Para poder girar en más direcciones se necesita un arreglo básico de 3 ruedas, dependiendo el sentido de giro de cada rueda, el movimiento resultante será diferente.

Rueda A	Rueda B	Rueda C	Dirección Resultante
<div style="text-align: center;">Adelante</div> 	<div style="text-align: center;">Adelante</div> 		Adelante
<div style="text-align: center;">Atrás</div> 	<div style="text-align: center;">Atrás</div> 		Atrás
<div style="text-align: center;">Adelante</div> 	<div style="text-align: center;">Atrás</div> 	<div style="text-align: center;">Atrás</div> 	Izquierda
<div style="text-align: center;">Atrás</div> 	<div style="text-align: center;">Adelante</div> 	<div style="text-align: center;">Adelante</div> 	Derecha

El robot entonces estará conformado por tres ruedas omnidireccionales dispuestas en una forma triangular sobre un chasis circular. Como se muestra en la figura 7.1.

Figura 7.1

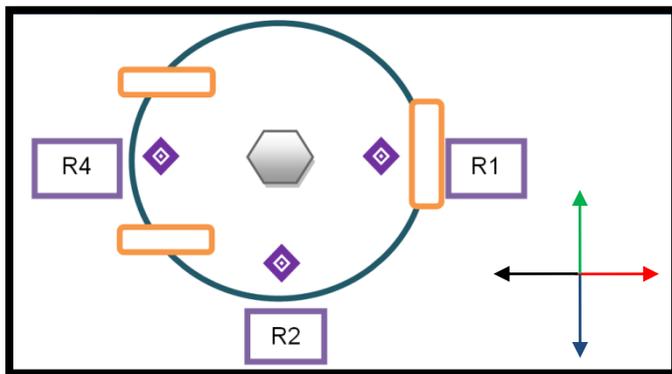


Tendrá dos secciones la primera pertenece al área del anclaje ubicada en la parte superior del chasis, que servirá para asegurar la carga, esto se pretende implementar con la ayuda de un electroimán.

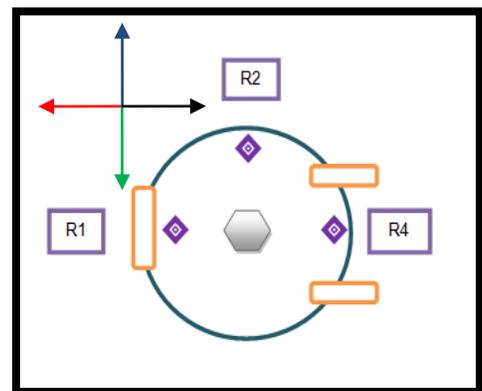
La segunda es la estructura de recepción que estará dispuesta a lo largo de la circunferencia del robot y contará con tres sensores ubicados estratégicamente alrededor de esta.

Como las ruedas que se van a utilizar serán omnidireccionales, cabe destacar que no girarán en el sentido tradicional de la palabra, sino que se moverán exclusivamente, horizontal y verticalmente según corresponda el caso.

	Adelante
	Derecha
	Izquierda
	Atrás



Posición de inicio, cuando el punto de partida está en la parte superior de un corredor o pasillo.



Posición de inicio, cuando el punto de partida está en la parte inferior de un corredor o pasillo.

7.1 Diagramas de flujo

Estos diagramas de flujo tienen que acoplarse a las dos estructuras eléctricas externas correspondientemente.

Recordando el seudocódigo para llegar a cualquier habitación del corredor, comenzando desde un punto de inicio, el cual se encuentra en la parte superior del corredor. (véase la figura 7.2).

- 1) El robot empezará en la posición de inicio.
- 2) La carga será posicionada encima del robot.
- 3) Después se moverá hacia adelante.
- 4) Al encontrar el marco de la puerta girará a la derecha, hasta encontrar el cuarto correcto.
 - 4.A) Si el destino está del lado derecho del corredor
 - 4.A1) Ahí tendrá que moverse de frente, desactivar la carga y dejarla adentro del cuarto.
 - 4.A2) Después se moverá hacia atrás.
 - 4.B) Si el destino por lo contrario se encuentra en la parte izquierda del corredor.
 - 4.B1) Tendrá que moverse hacia atrás, desactivar la carga y dejarla adentro del cuarto.
 - 4.B2) Después se moverá de frente.
- 5) Al encontrar el marco de la puerta tendrá que moverse a la izquierda, hasta encontrar el punto de inicio. Ahí se moverá hacia atrás.
- 6) Asegurar otra carga y moverse de frente pero ahora en dirección a un diferente destino donde se requiera el material.

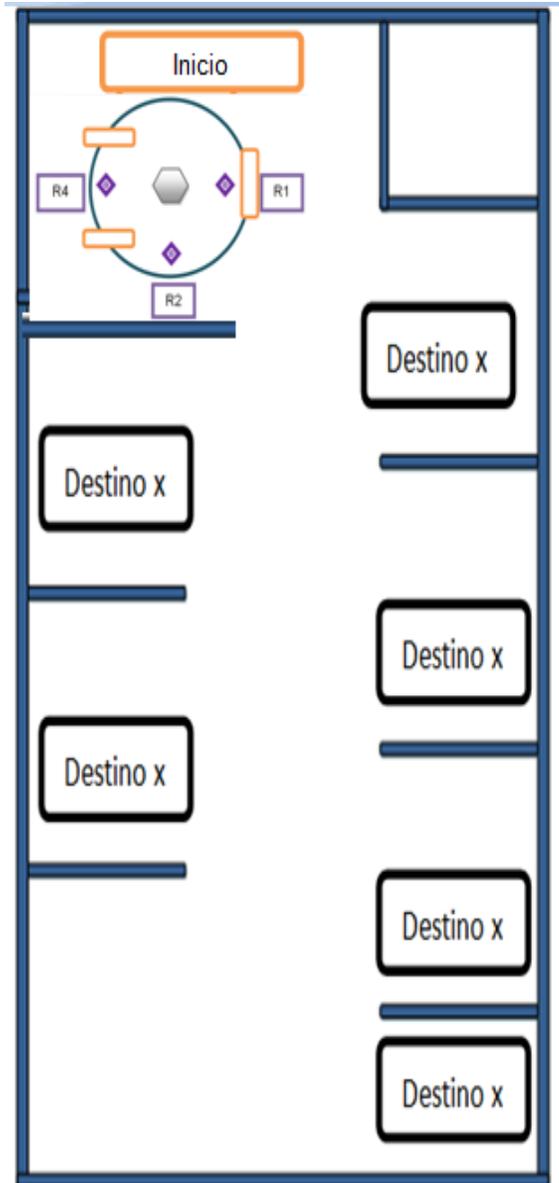
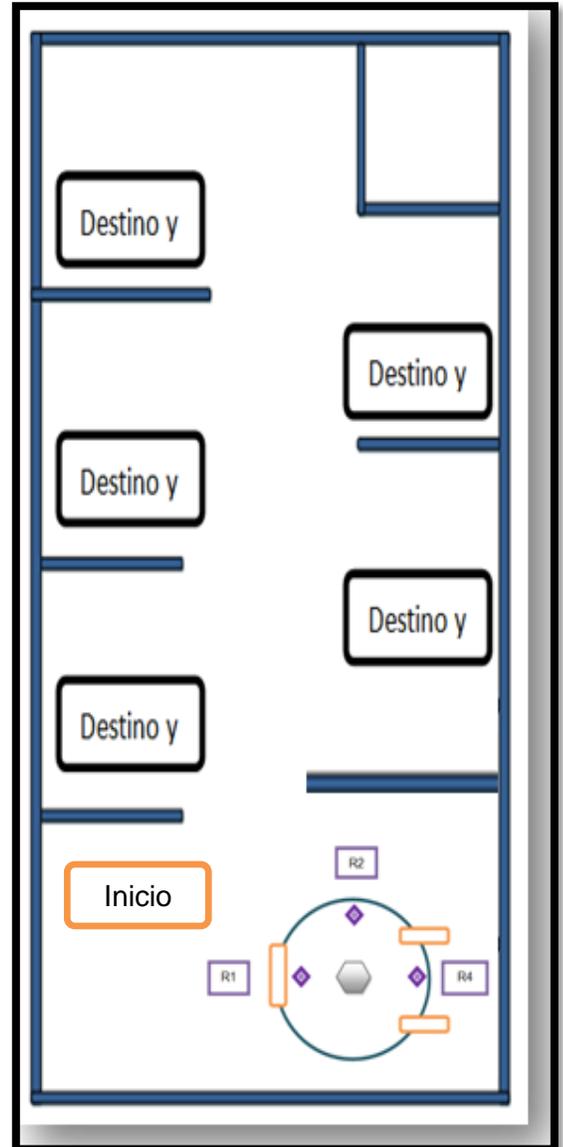


Figura 7.2

Ahora bien para crear el seudocódigo y posteriormente el diagrama de flujo que satisfaga la entrega de material a cualquier habitación del corredor, comenzando desde un punto de inicio, el cual se encuentra en la parte inferior del corredor, solo basta con cambiar de posición el robot y el algoritmo será el mismo. (véase la figura 7.3).

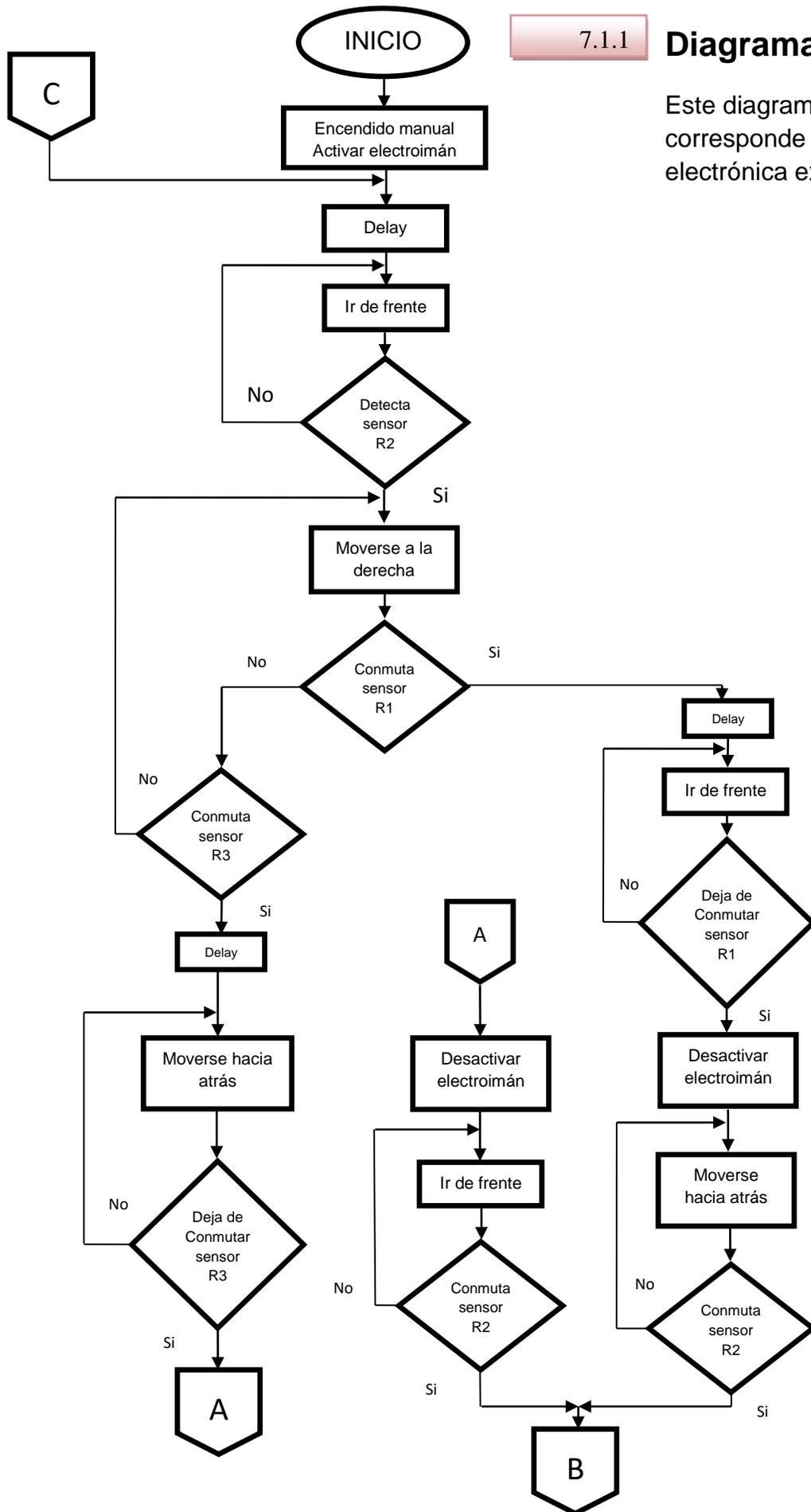
Figura 7.3

- 1) El robot empezará en la posición de inicio.
- 2) La carga será posicionada encima del robot.
- 3) Después se moverá hacia adelante.
- 4) Al encontrar el marco de la puerta girará a la derecha, hasta encontrar el cuarto correcto.
 - 4.A) Si el destino está del lado derecho del corredor.
 - 4.A1) Ahí tendrá que moverse hacia atrás, desactivar la carga y dejarla adentro del cuarto.
 - 4.A2) Después se moverá de frente.
 - 4.B) Si el destino por lo contrario se encuentra en la parte izquierda del corredor.
 - 4.B1) Tendrá que moverse de frente, desactivar la carga y dejarla adentro del cuarto.
 - 4.B2) Después se moverá hacia atrás.
- 5) Al encontrar el marco de la puerta tendrá que moverse a la izquierda, hasta encontrar el punto de inicio. Ahí se moverá hacia atrás.
- 6) Asegurar otra carga y moverse de frente pero ahora en dirección a un diferente destino donde se requiera el material.



Como se puede observar es el mismo seudocódigo, sólo que este último está invertido en los puntos 4.A y 4.B con respecto al primero, esto no causa ninguna alteración o mal funcionamiento en el robot, entonces podemos concluir que el seudocódigo funciona sin importar si cambiamos el punto de partida.

Este diagrama de flujo corresponde a la estructura electrónica externa número 1



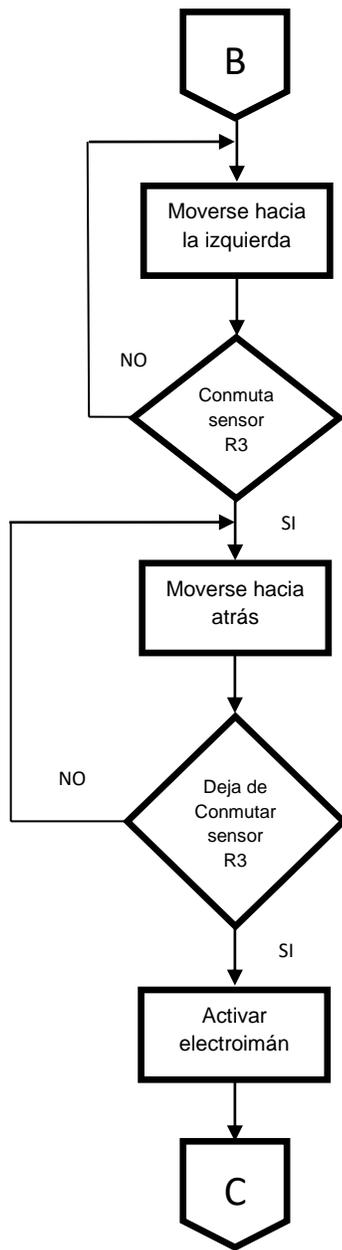
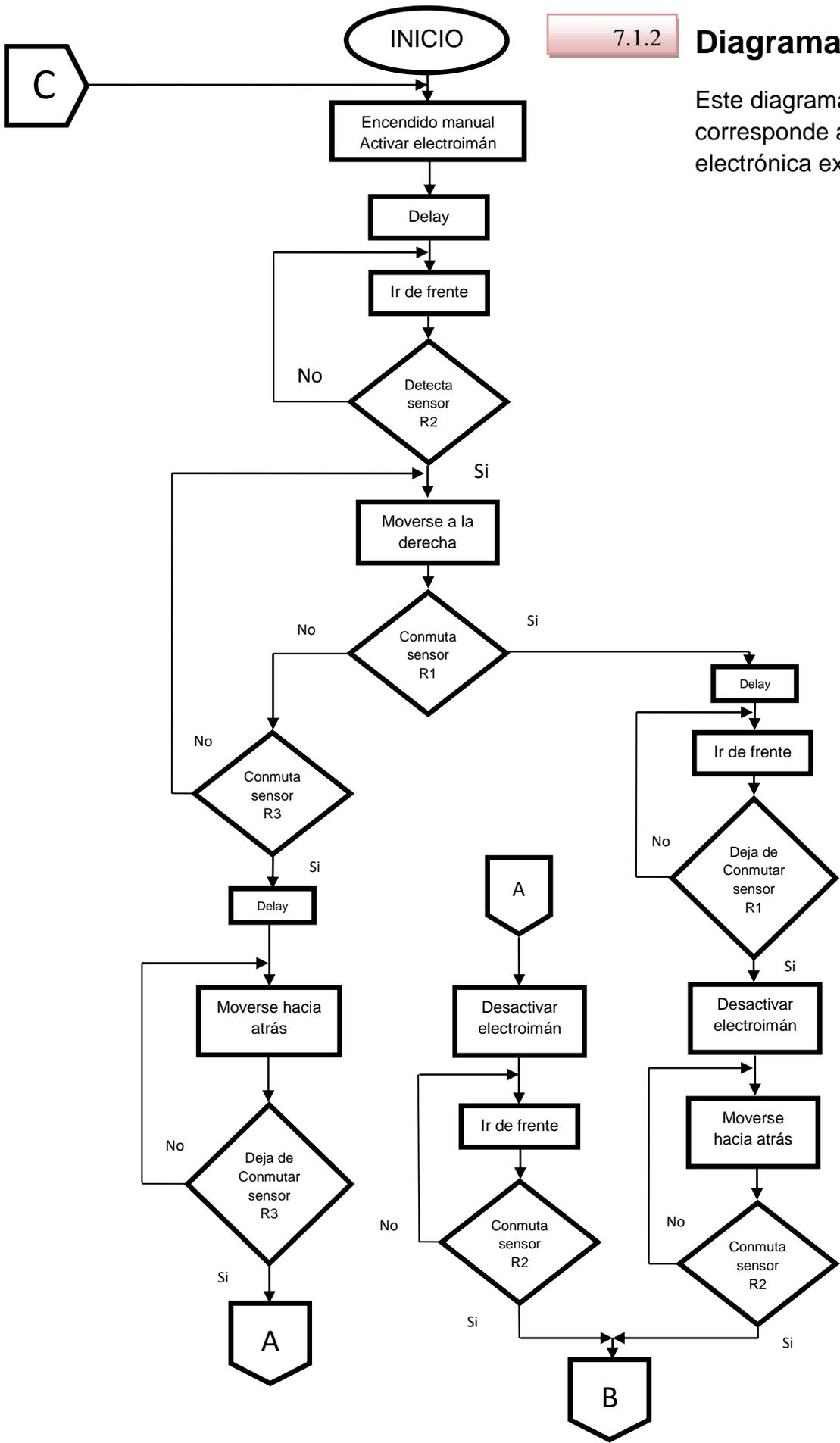
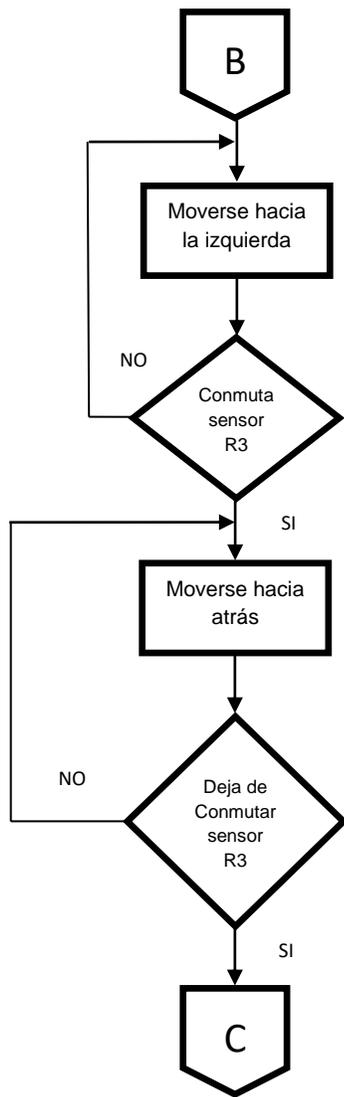


Diagrama de flujo 2

Este diagrama de flujo corresponde a la estructura electrónica externa número 2

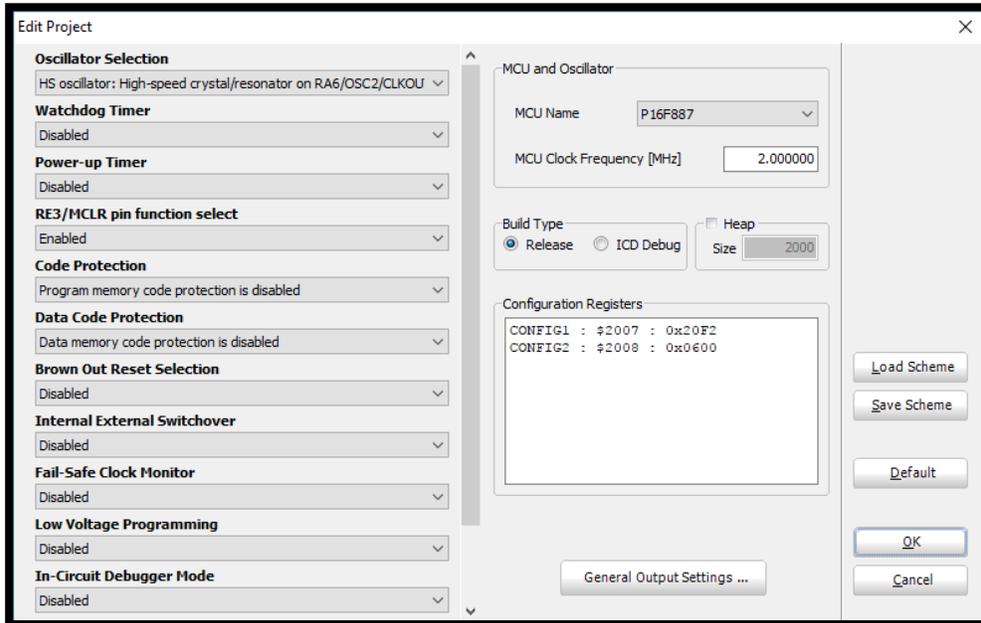




7.2

Configuración del microcontrolador

Este diagrama se transcribirá en el ambiente de programación denominado “Mikro C” como ya se había mencionado anteriormente.



7.3

Código de programación 1

El código de programación basado en el diagrama de flujo número1 es el siguiente.

```

• unsigned short speed ; // Variable de 0 a 255 sin signo
• int x=0; //Variable de -32768 .. 32767
• //int delay_val-Variable de -32768 .. 32767 que va a obtener su valor a partir de la lectura de voltaje analogico
- void cw(int delay_val) { //Funcion: ir Adelante-ElectroIman Activo
•
• if(delay_val >0){
• portc=0b000001000;portd=0b10001000;for( x=0;x<delay_val;x++) {delay_ms(5);} //Funciones que muestra en los diferentes puertos (C,B y D)
• portc=0b000000100;portd=0b10000100;for( x=0;x<delay_val;x++) {delay_ms(5);} //la secuencia necesaria para mover el motor en una determinada
10 portc=0b000000010;portd=0b10000010;for( x=0;x<delay_val;x++) {delay_ms(5);} // direccion
• portc=0b000000001;portd=0b100000001;for( x=0;x<delay_val;x++) {delay_ms(5);} //El valor del retraso o delay esta determinada por
• //el valor de la variable "delay_val" y "x"
• }
• //En el bucle for "x" pretende alcanzar el valor de la variable "delay_val"
}

```



```

while((PORTE.RE1 == 1) ^ (I == 1) ^ (PORTE.RE0 == 1) ^ (PORTE.RE2 == 1)) //Funciona como SWITCH-Compuerta XOR ""
{
  //Si dos de las sentencias anteriores son verdaderas se saldra de la rutina, al igual que si todas las sentencias son falsas
  for (a=0;a<4; a=a+1)
  {
    sr2(speed); // Llama a la funcion "sr2" con el valor de "speed" obtenido analogicamente
    delay_ms(5);
  }
  delay_ms(10);
  I=1; //Sirve para anclar la rutina sin la necesidad de mantener el "pin 1" del puerto "E" en "1" o en "alto"
}

while(PORTE.RE0 == 1) //Caundo el "pin 0" del puerto "E" reciba un "1" realizas lo siguiente: //CAMINO A 1.0
{
  I=0;
  J=1;
  for (a=0;a<4; a=a+1)
  {
    cw(speed); // Llama a la funcion "cw" con el valor de "speed" obtenido analogicamente
    delay_ms(5);
  }
}

```

```

while(PORTE.RE2 == 1) //Caundo el "pin 2" del puerto "E" reciba un "1" realizas lo siguiente: //CAMINO B 1.0
{
  I=0;
  F=1;
  for (a=0;a<4; a=a+1)
  {
    sr3(speed); // Llama a la funcion "cw" con el valor de "speed" obtenido analogicamente
    delay_ms(5);
  }
}

while((PORTE.RE2 == 0) && (F == 1) && (PORTE.RE1 == 0)) // Compuerta AND //CAMINO B 1.1 //Seccion ElectroIman Desactivado
{
  //Si alguna de estas tres sentencias no se cumple, se saldra de la rutina
  for (a=0;a<4; a=a+1)
  {
    ncw(speed); // Llama a la funcion "ncw" con el valor de "speed" obtenido analogicamente
    delay_ms(5);
  }
}

```

```

while((PORTE.RE0 == 0) && (J == 1) && (PORTE.RE1 == 0)) //CAMINO A 1.1
{
  //Si alguna de estas tres sentencias no se cumple se saldra de la rutina
  for (a=0;a<4; a=a+1)
  {
    xcw(speed); // Llama a la funcion "xcw" con el valor de "speed" obtenido analogicamente
    delay_ms(5);
  }
}

while((PORTE.RE1 == 1) ^ (M == 1) ^ (PORTE.RE2 == 1)) //CAMINO A 1.2
{
  //Si dos de las sentencias anteriores son verdaderas se saldra de la rutina al igual que si todas las sentencias son falsas
  J=0;
  F=0;
  L=1;
  M=1;
  for (a=0;a<4; a=a+1)
  {
    nsr3(speed); // Llama a la funcion "nsrv3" con el valor de "speed" obtenido analogicamente
    delay_ms(5);
  }
}

```

```

while((PORTE.RE2 == 1) && (L == 1)) //CAMINO A 1.3
{
  //Si alguna de estas dos sentencias no se cumple se saldra de la rutina
  for (a=0;a<4; a=a+1)
  {
    xcw(speed); // Llama a la funcion "xcw" con el valor de speed obtenido analogicamente
    delay_ms(5);
  }
}

```

Código de programación 2

El código de programación basado en el diagrama de flujo número 2 es el siguiente.

```

.
. unsigned short speed ; // Variable de 0 a 255 sin signo
. int x=0; //Variable de -32768 .. 32767
. //int delay_val-Variable de -32768 .. 32767 que va a obtener su valor a partir de la lectura de voltaje analogico
- void cw(int delay_val) { //Funcion: ir Adelante-ElectroIman Activo
.
. if(delay_val >8){
. portc=0b00001000;portd=0b10001000;for( x=0;x<delay_val;x++) {delay_ms(5);} //Funciones que muestra en los diferentes puertos (C,B y D)
. portc=0b00000100;portd=0b10000100;for( x=0;x<delay_val;x++) {delay_ms(5);} //la secuencia necesaria para mover el motor en una determinada
10 portc=0b00000010;portd=0b10000010;for( x=0;x<delay_val;x++) {delay_ms(5);} // direccion
. portc=0b00000001;portd=0b10000001;for( x=0;x<delay_val;x++) {delay_ms(5);} //El valor del retraso o delay esta determinada por
. } //el valor de la variable "delay_val" y "x"
. } //En el bucle for "x" pretende alcanzar el valor de la variable "delay_val"
}

- void sr2(int delay_val) { //Funcion: ir a la Derecha-ElectroIman Activo
.
. if(delay_val >8){
. portc=0b00000001;portd=0b10001000;portb=0b00001000;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0b00000010;portd=0b10000100;portb=0b00000100;for( x=0;x<delay_val;x++) {delay_ms(5);}
20 portc=0b00000100;portd=0b10000010;portb=0b00000010;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0b00001000;portd=0b10000001;portb=0b00000001;for( x=0;x<delay_val;x++) {delay_ms(5);}
. }
. }

- void sr3(int delay_val) { //Funcion: ir a la Atras-ElectroIman Activo
.
. if(dclay_val >8){
. portc=0b00000001;portd=0b10000001;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0b00000010;portd=0b10000010;for( x=0;x<delay_val;x++) {delay_ms(5);}
30 portc=0b00000100;portd=0b10000100;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0b00001000;portd=0b10001000;for( x=0;x<delay_val;x++) {delay_ms(5);}
. }
. }

- void xcw(int delay_val) { //Funcion: ir Atras-ElectroIman Desactivado
.
. if(delay_val >8){
. portc=0b00000001;portd=0b00000001;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0x00000010;portd=0b00000010;for( x=0;x<delay_val;x++) {delay_ms(5);}
40 portc=0b00000100;portd=0b00000100;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0b00001000;portd=0b00001000;for( x=0;x<delay_val;x++) {delay_ms(5);}
. }
. }

- void ncr3(int delay_val) { //Funcion: ir a la Izquierda-ElectroIman Desactivado
.
. if(delay_val >8){
. portc=0x00001000;portd=0b00000001;portb=0b00000001;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0b00000100;portd=0b00000010;portb=0b00000010;for( x=0;x<delay_val;x++) {delay_ms(5);}
50 portc=0b00000010;portd=0b00000100;portb=0b00000100;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0b00000001;portd=0b00001000;portb=0b00001000;for( x=0;x<delay_val;x++) {delay_ms(5);}
. }
. }

- void ncw(int delay_val) { //Funcion: ir Adelante-ElectroIman Desactivado
.
. if(delay_val >8){
. portc=0b00001000;portd=0b00001000;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0b00000100;portd=0b00000100;for( x=0;x<delay_val;x++) {delay_ms(5);}
60 portc=0b00000010;portd=0b00000010;for( x=0;x<delay_val;x++) {delay_ms(5);}
. portc=0b00000001;portd=0b00000001;for( x=0;x<delay_val;x++) {delay_ms(5);}
. }
. }

```

```

void main()
{
short a;      // Variable de - 128 .. 127
short I;     // Variable de - 128 .. 127
short F;     // Variable de - 128 .. 127
70 short J;   // Variable de - 128 .. 127
short M;     // Variable de - 128 .. 127
short L;     // Variable de - 128 .. 127
PORTA=0;    // Todos los pines del puerto E se ponen a cero
PORTE=0;    // Todos los pines del puerto B se ponen a cero
PORTD=0;    // Todos los pines del puerto E se ponen a cero
PORTB=0;    // Todos los pines del puerto B se ponen a cero
PORTC=0;    // Todos los pines del puerto E se ponen a cero
ANSEL=0;    // Configura AN pins como I/O digitales
ANSELH=0;
80 TRISA=1;   //Puerto A como entrada
TRISE=1;    //Puerto B como entrada SENSORES
TRISD=0;    //Puerto D como salida MOTOR B
TRISB=0;    //Puerto D como salida MOTOR C
TRISC=0;    //Puerto D como salida MOTOR A

```

```

ADCON1 = 0; // CONFIGURO TODOS LOS CANLES COMO ANALOGICOS Y VOLTAJE DE REFERENCIA SE TOMA DEL MICROCONTROLADOR
while(1)    //Para anclar el programa
{
//El 0 es pin del puerto A en el que obtiene el valor analogico
90 speed = ADC_read(0)/4; //Al obtener el valor analogico de ese pin lo divido entre 4 por lo tanto speed = (valor analogico en RA0/4)
while(PORTA.RA1 == 0) //Mientras el pin 1 del puerto A sea 0 realizas lo siguiente FUNCIONA COMO ON/OFF
{
I=0;
F=0; // Inicializo todas la variables
J=0;;
M=0;
L=0;
while(PORTE.RE1 == 0) // Mientras el pin 1 del puerto B este en 0 y k sea 0 realizas lo siguiente: "IR ADELANTE"
{
//El bucle for sirve para ancla la subrutina
100 for (a=0;a<4; a=a+1) //Para la variable "a" que vale 0 y "a" es menor que 4 a la variable "a" le sumas 1 conforme a cada ciclo
{
cw(speed); // Llama a la funcion "cw", el valor de speed se lo asigno a la variable "delay_val" (delay_val = valor analogico)
delay_ms(5); //Retrazo la instruccion 5 mili segundos
}
}
}

```

```

while((PORTE.RE1 == 1) ^ (I == 1) ^ (PORTE.RE0 == 1) ^ (PORTE.RE2 == 1)) //Funciona como SWITCH-Compuerta XOR ""
{
//Si dos de las sentencias anteriores son verdaderas se saldra de la rutina, al igual que si todas las sentencias son falsas
for (a=0;a<4; a=a+1)
{
110 sr2(speed); // Llama a la funcion "sr2" con el valor de "speed" obtenido analogicamente
delay_ms(5);
}
delay_ms(10);
I=1; //Sirve para anclar la rutina sin la necesidad de mantener el "pin 1" del puerto "E" en "1" o en "alto"
while(PORTE.RE0 == 1) //Cuando el "pin 0" del puerto "E" reciba un "1" realizas lo siguiente: //CAMINO A 1.0
{
I=0;
J=1;
120 for (a=0;a<4; a=a+1)
{
cw(speed); // Llama a la funcion "cw" con el valor de "speed" obtenido analogicamente
delay_ms(5);
}
}
}

```

```

while(PORTE.RE2 == 1) //Cuando el "pin 2" del puerto "E" reciba un "1" realizas lo siguiente: //CAMINO B 1.0
{
I=0;
F=1;
130 for (a=0;a<4; a=a+1)
{
sr3(speed); // llama a la funcion "cw" con el valor de "speed" obtenido analogicamente
delay_ms(5);
}
}
while((PORTE.RE2 == 0) && (F == 1) && (PORTE.RE1 == 0)) // Compuerta AND //CAMINO B 1.1 //Seccion ElectroIman Desactivado
{
//Si alguna de estas tres sentencias no se cumple, se saldra de la rutina
for (a=0;a<4; a=a+1)
{
140 ncw(speed); // Llama a la funcion "ncw" con el valor de "speed" obtenido analogicamente
delay_ms(5);
}
}
}

```

```

while((PORTE.RE0 == 0) && (J == 1) && (PORTE.RE1 == 0)) //CAMINO A 1.1
{
    //Si alguna de estas tres sentencias no se cumple se saldra de la rutina
    for (a=0;a<4; a=a+1)
    {
        xcw(speed); // Llama a la funcion "xcw" con el valor de "speed" obtenido analógicamente
        delay_ms(5);
    }
}

while((PORTE.RE1 == 1) ^ (M == 1) ^ (PORTE.RE2 == 1)) //CAMINO A 1.2
{
    //Si dos de las sentencias anteriores son verdaderas se saldra de la rutita al igual que si todas las sentencias son falsas
    J=0;
    F=0;
    L=1;
    M=1;
    for (a=0;a<4; a=a+1)
    {
        nsrw3(speed); // Llama a la funcion "nsrw3" con el valor de "speed" obtenido analógicamente
        delay_ms(5);
    }
}

```

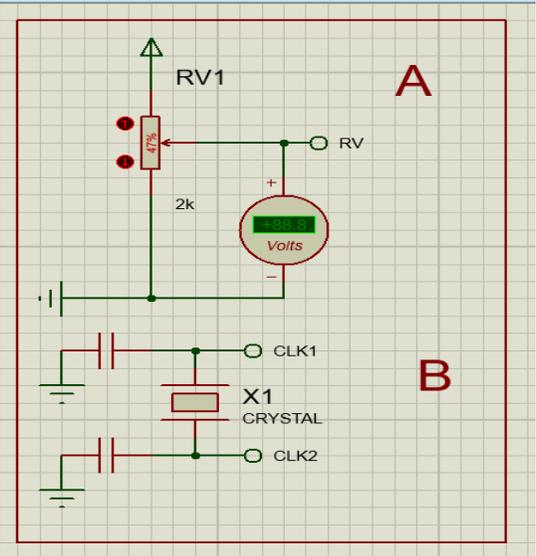
```

while((PORTE.RE2 == 1) && (L ==1)) //CAMINO A 1.3
{
    //Si alguna de estas dos sentencias no se cumple se saldra de la rutina
    for (a=0;a<4; a=a+1)
    {
        xcw(speed); // Llama a la funcion "xcw" con el valor de speed obtenido analógicamente
        delay_ms(5);
    }
}

```

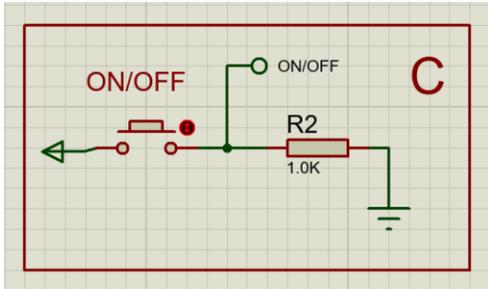
7.5 Simulación

Para simular los códigos de programación y cotejar que estos realmente funcionan, se utilizó el software “Proteus”. Los sensores fueron sustituidos por “Push Buttons” para facilitar la simulación. Ambos códigos se simularon de la misma manera.

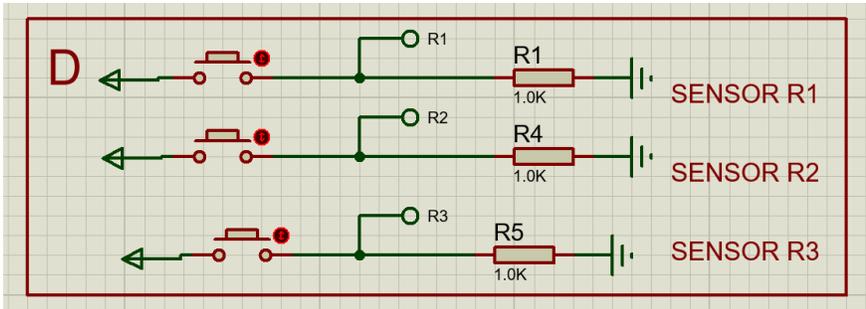


(A) Módulo donde se obtiene el valor analógico del voltaje para el ajuste de la velocidad a los motores o el retraso

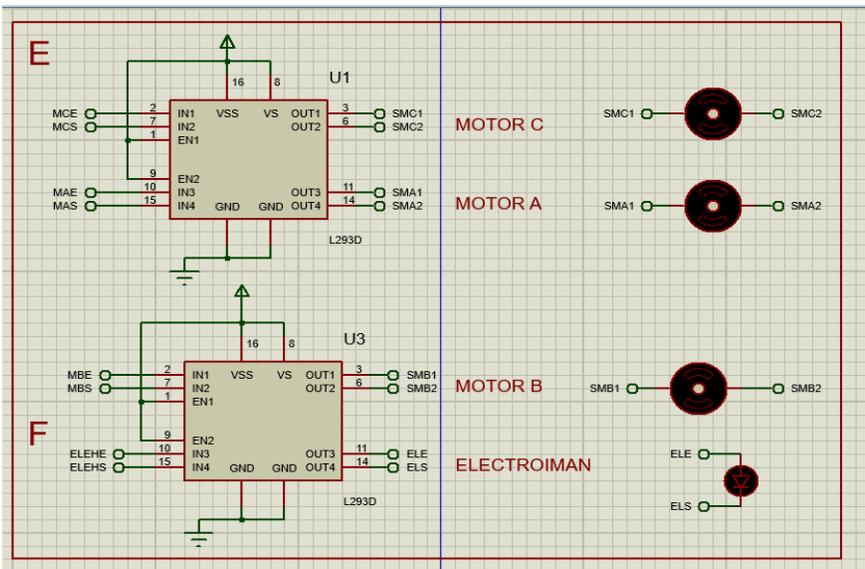
(B) Configuración básica del microcontrolador



(C) Módulo externo de encendido y apagado

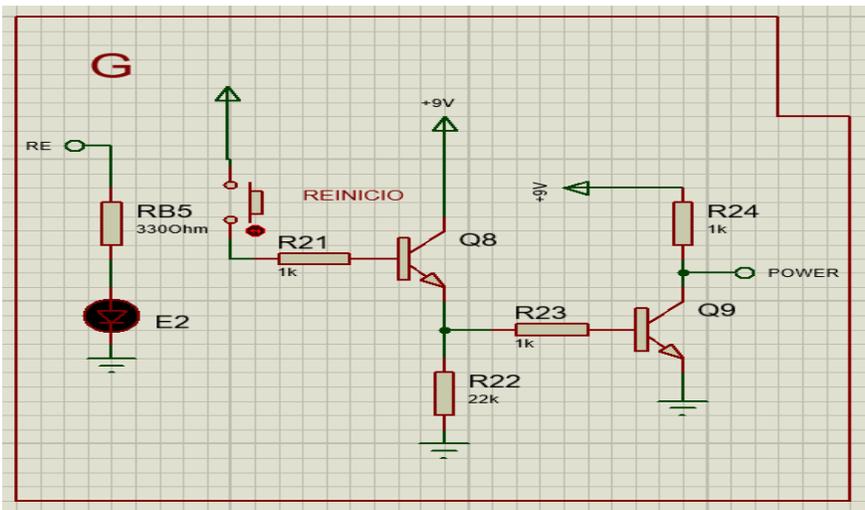


(D) Módulo de sensores



(F) Representación del sistema "Electroimán"

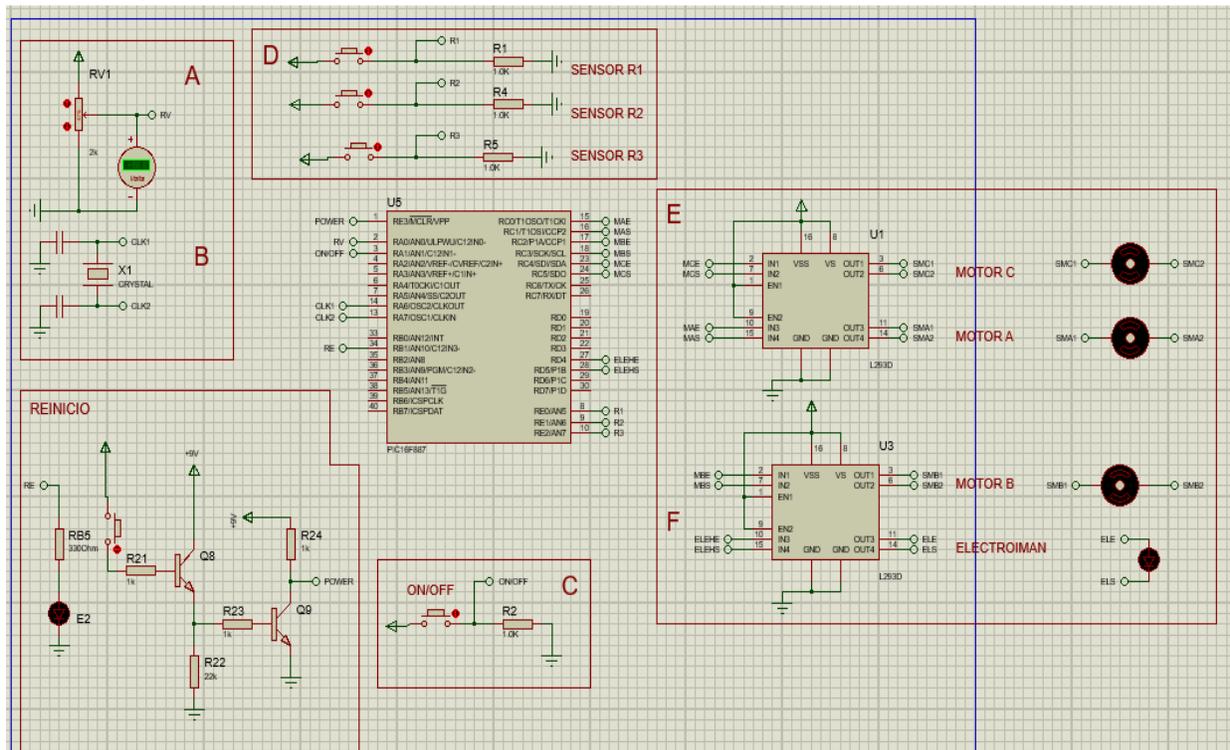
(E) Módulos de potencia y su conexión con los motores



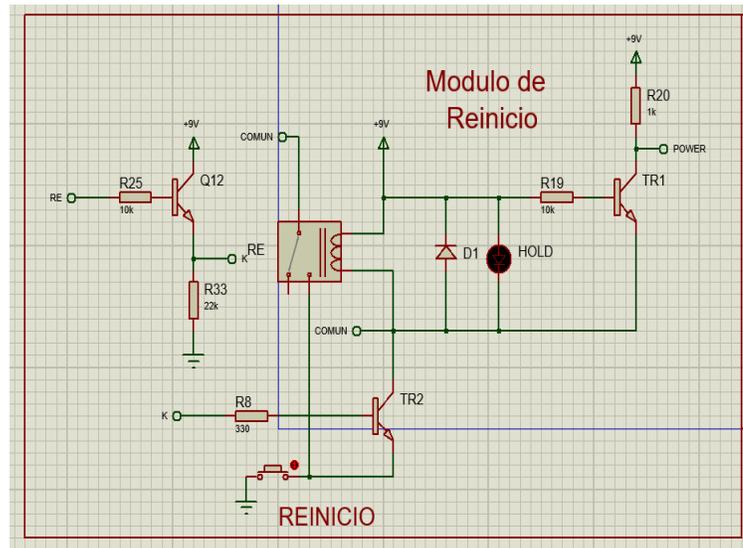
(G) Módulos de reinicio

7.5.1

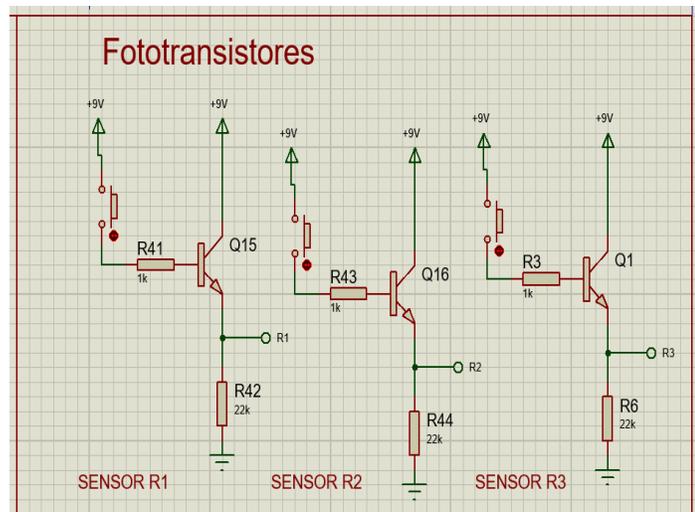
Circuito electrónico del robot



Este módulo cuenta con un circuito de enclave con relevador para mantener apagado el microcontrolador una vez que finalice la tarea y pueda ser reiniciado por un operador manualmente a través de un switch.

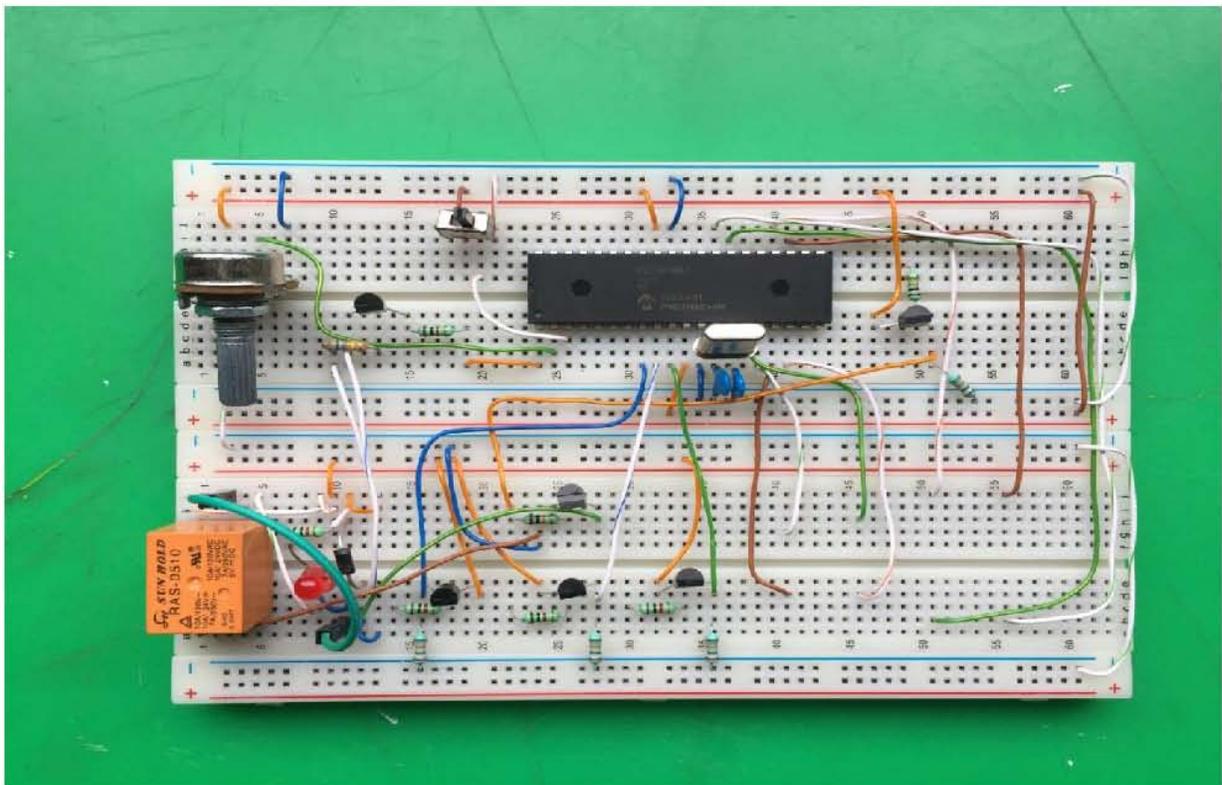
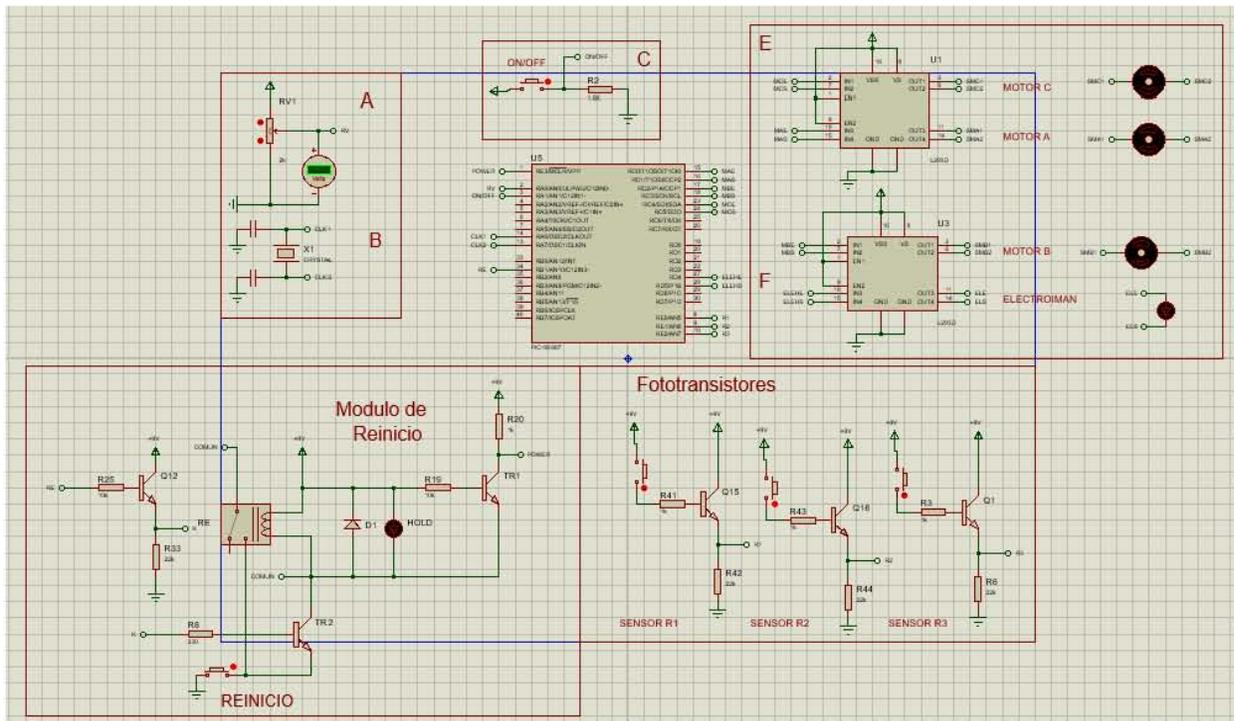


Este módulo fue agregado para la representación de los fototransistores ocupados para este proyecto, este módulo cuenta con una etapa de recepción de señal y acoplamiento de dicha señal.



7.6.1

Circuito electrónico del robot integro

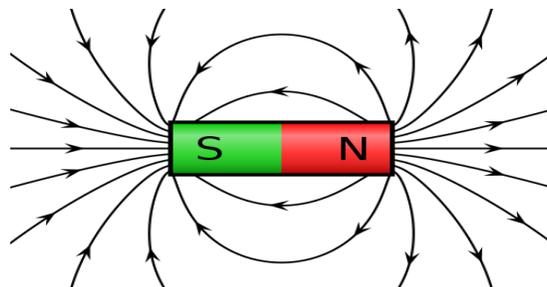


8 Electroimán

Ahora para implementar el sistema que corresponde al electroimán se hará de la siguiente manera. Primero tenemos que saber que es un electroimán y como poder usarlo.

Muchos de los aparatos que potencian nuestra vida tecnológica, dependen de la relación fundamental entre el movimiento y dos propiedades básicas de la materia, la electricidad y el magnetismo.

Los polos norte y sur en un imán son los puntos de concentración del campo magnético que lo rodea. Este campo magnético se puede utilizar para producir movimiento atrayendo a otros materiales magnéticos cuando los polos magnéticos son distintos y repeliéndolos cuando los polos son iguales. (véase las figuras 8.0).



Figuras 8.0



Movimiento



Movimiento

La corriente eléctrica también puede producir campo magnético, la corriente eléctrica que fluye por un cable, es un constante movimiento de electrones, diminutas unidades de carga. Esto hace que se forme un campo magnético a su alrededor. (véase la figura 8.1).

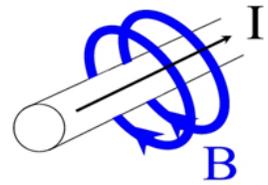


Figura 8.1

Cuando la corriente deja de fluir el campo se desvanece, cuando empieza a fluir de nuevo la corriente, el campo magnético reaparece.

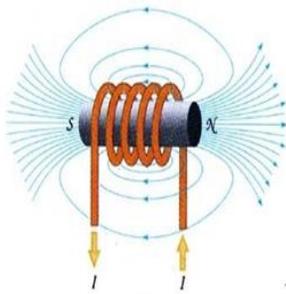


Figura 8.2

Como ya lo habíamos mencionado dos campos magnéticos interactuando pueden producir movimiento.

El campo magnético que se produce en un cable cuando fluye corriente, se intensifica si este cable forma una espiral y cuando el centro de la espiral es de hierro se intensifica aun más, al concentrarse el magnetismo. Esto es un electroimán y es mucho más potente que un imán común. Los electroimanes son muy útiles por que se pueden encender y apagar. (véase la figura 8.2). La relación entre la electricidad, magnetismo y movimiento es tan íntima que dos de ellas producen la tercera.

8.1 Implementación

Teniendo en cuenta lo anterior, ahora podemos implementar un electroimán que nos ayude al anclaje del robot y la carga a transportar. Este sistema constará de un electroimán previamente construido como se muestra en la imagen 8.3

La parte inferior de dicho electroimán es la que se imanta, lo que permite el anclaje del robot a la carga. (véase la imagen 8.4).



Imagen 8.3



Voltaje de alimentación: 12 volts de corriente directa para funcionar a su máxima capacidad de carga.

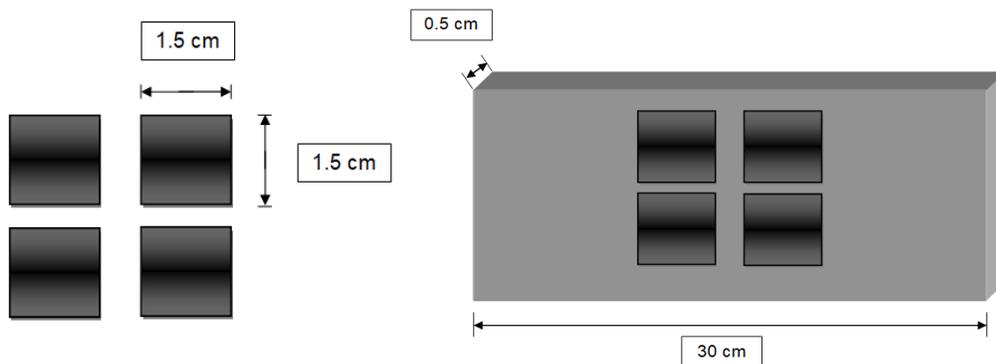
Capacidad de carga: 50 Kilogramos.

Imagen 8.4

El "Rolling Trolley" en el cual se depositará el material que se transportará a las diferentes ubicaciones (véase las imagen 8.5), será de una altura de 50 cm. Se agregaran 4 segmentos cuadrados de hierro de 1.5 cm alto y 1.5 cm de ancho, estos se encontrarán por debajo de la última charola, al activarse el electroimán, la superficie circular anteriormente mencionada quedará unida con los segmentos cuadrados de hierro. (véase las figuras 8.6).

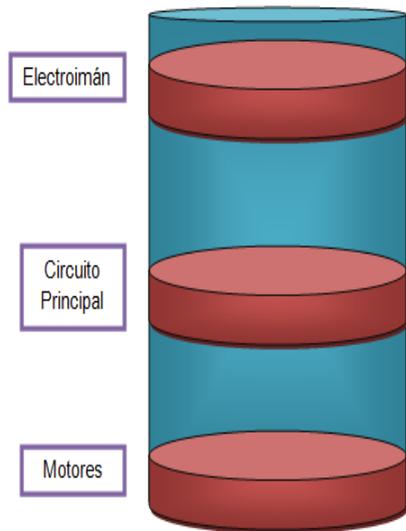


Imagen 8.5



Figuras 8.6

9 Diseño del chasis



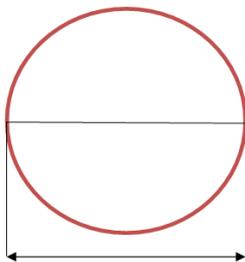
El diseño del chasis del robot se hará como se muestra en la figura 9.0.

Sera un chasis cilíndrico con tres bases dentro del mismo las cuales servirán para colocar los elementos correspondientes al electroimán, el circuito principal y los motores.

El material seleccionado para la implementación física, es la madera, este elemento es versátil ya que ofrece una buena resistencia y poco peso, además es un buen aislante térmico y eléctrico. El tipo de madera que se empleara será del tipo “triplay aglomerado”.

Figura 9.0

Diámetro de las bases



17.5 cm

Espesor de las bases



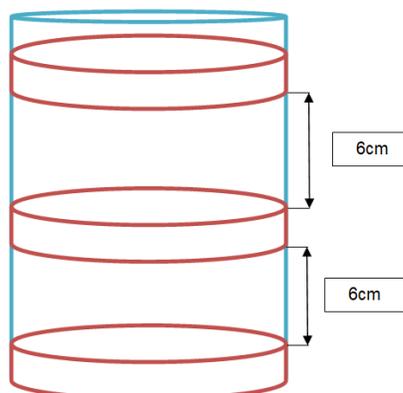
0.4 cm

Medida vertical del chasis



18 cm

Distancia de separación entre bases

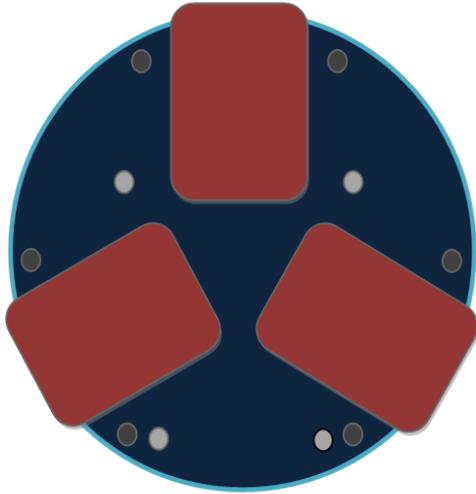


6cm

6cm

9.1

Diseño de la base para los motores



El diseño de la base para los motores se muestra en la figura 9.1.

* En los rectángulos carmesí es donde se insertaran los tres motores.

* Los círculos negros representan los hoyos en los cuales se colocarán tornillos para asegurar los motores.

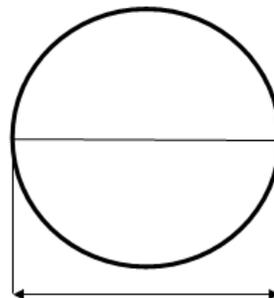
Figura 9.1

Espacio para los motores



2.5 cm

Diámetro de los tornillos



0.5 cm

Implementación

Se realizara la siguiente secuencia de instrucciones para elaborar el chasis del prototipo, como es mencionado anteriormente, en las páginas 57 y 58 respectivamente.

- 1) Se cortarán 3 trozos de madera con las medidas previamente mencionadas en la página 57 (véase las imagen 9.2).



Imagen 9.2

- 2) Con el fin de ensamblar las tres bases entre sí, se optó por utilizar 4 fragmentos rectangulares de madera para formar columnas (véase las imagen 9.3).

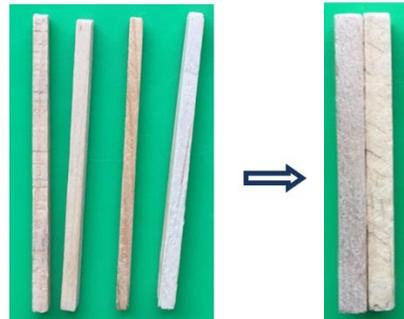


Imagen 9.3

- 3) Para lograr unir todos los elementos correspondientes en su lugar, se recomienda hacer uso de un adhesivo resistente ya sea “UHU” o “Resistol 850”, como se muestra en la imagen 9.4.

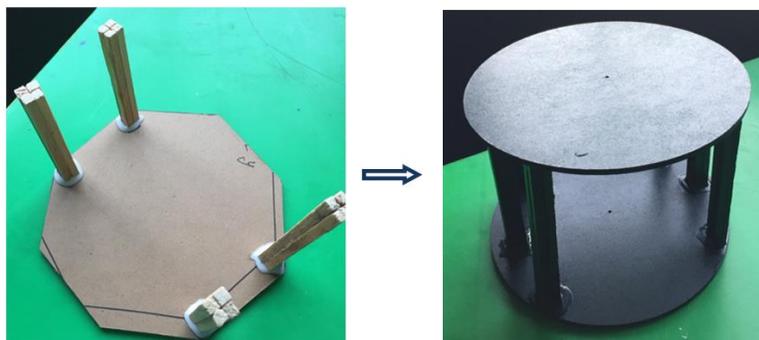
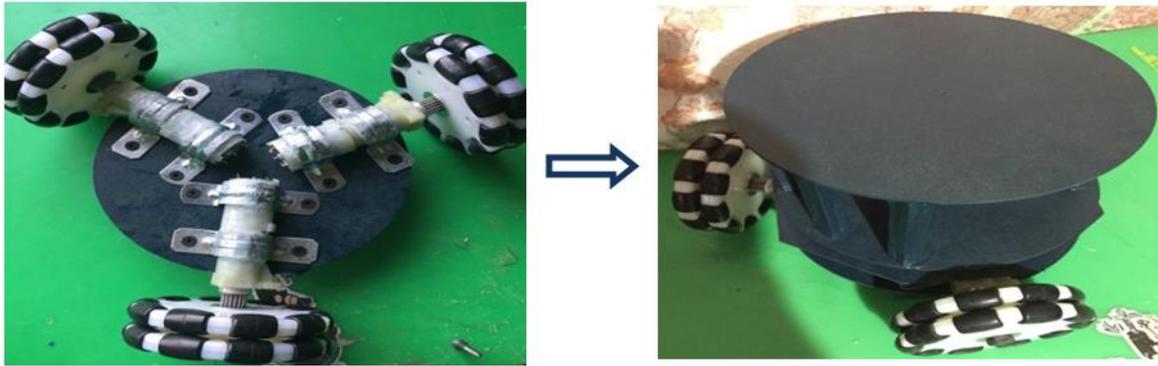


Imagen 9.4

- 4) Ahora con la finalidad de implementar la base de los motores, se emplearan “pijas” y “agarraderas” para mantener los motores sostenidos a la base, se sugiere que el grosor de esta base sea aproximado de 2.5 cm o bien la utilización de “pijas” con una longitud corta. La instalación de los motores con la base se ve a continuación.



El chasis terminado con todos sus elementos ya montados como se muestra en la imagen 9.5.

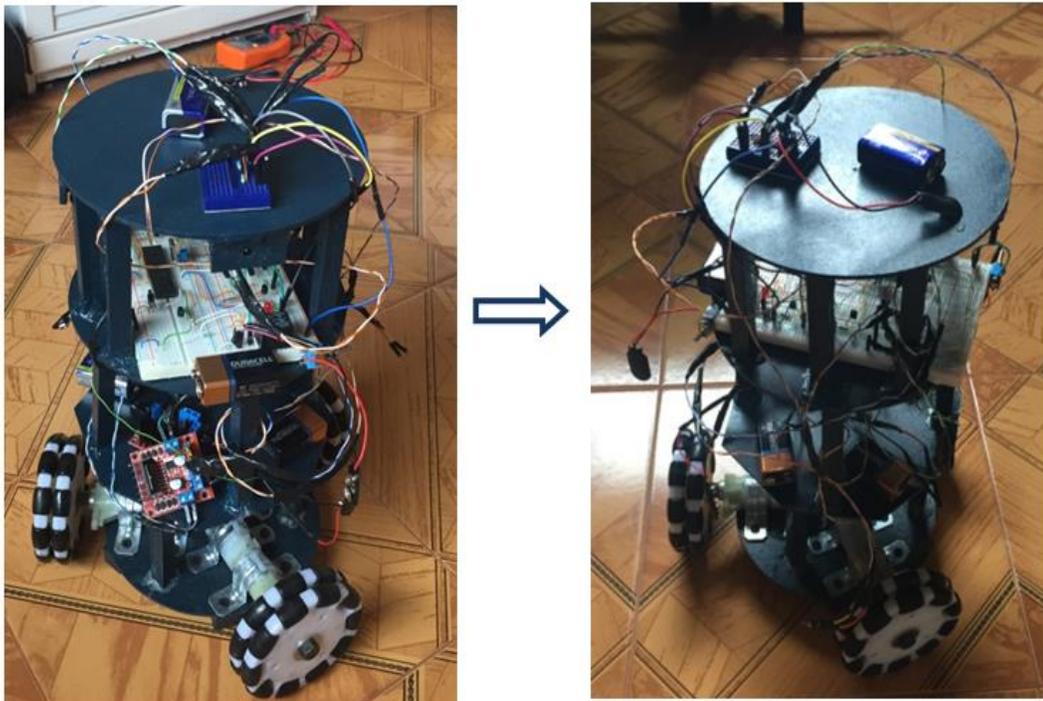


Imagen 9.5

10 Conclusiones

En este proyecto se probó el rendimiento de las diferentes estructuras electrónicas, planteamiento y solución dada por el autor a la problemática del transporte de materiales, esta comprobación resultó satisfactoria, así que es posible utilizar este modelo para ayudar a esta labor. Es recomendable recordar que éste diseño tiene restricciones y protocolos que se deben proseguir (repetir) para un buen funcionamiento, todo esto es mencionado en los diferentes temas y subtemas del presente proyecto.

Este diseño fue implementado a manera de prototipo de fácil comprensión ya que para el transporte de cargas pesadas de 20 a 40kg aproximadamente se requieren componentes más industriales, al optar por esta forma de ejecución se pudo invertir más continuidad en el apartado de la programación, lógica, algoritmos, planteamiento, implementación y solución, ya que todo esto puede ser trasladado a componentes más eficaces, pero su proceso en general sería el mismo.

Este desarrollo no pretende remover a los trabajadores encargados en la transportación de materiales, sino a agilizar dicha labor y también al estar este personal desocupado, se puede encargar de otras tareas en las cuales se requiera con mayor necesidad su presencia.

Una de las aplicaciones que posee este proyecto es que, se puede utilizar en varios entornos y por lo tanto satisfacer diferentes problemáticas en diversos ambientes.

El proyecto está abierto a sugerencias de mejoramiento y fue realizado también como una guía y apoyo a proyectos similares.

11 Referencias

11.1

Bibliografía

- W. Bolton. Mecánica, Sistemas de Control Eléctrico en la Ingeniería Mecánica y Eléctrica. 4ª Edición. Omega
- R.C. Fernando. Mecatrónica, Control y Automatización. 2ª Edición. Alfa Omega
- Josep Balcells, José Luis Romeral. Autómatas Programables. AlfaOmega
- K.S. FU, R.S. González, C.S.G. Lee. Robótica, Control, Detección, Visión e Inteligencia. McGrawHill
- Fernando E. Valdés Pérez, Ramón Pallás Areny. Microcontroladores: Fundamentos y aplicación con PIC. Marcombo.

11.2

Ciberografía

- <https://www.mikroe.com/mikroc-pic>
- <http://www.studentcompanion.net/en/introduction-to-mikroc-pro-for-pic-compiler/>
- <https://www.labcenter.com>
- <https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxkcm9jdGF2aW9kaWF6aGVybmFuZGV6fGd4OjVknGI2MGMwZTA1YmMxNTk>
- <http://www.hubor-proteus.com>