ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

CONSTRUCCIÓN DE UN PROTOTIPO DE ROBOT CON UN MICROCONTROLADOR, SENSORES DE LUZ Y SERVOMOTORES QUE REALICE MOVIMIENTOS AUTÓNOMOS

PROYECTO PREVIO A LA OBTENCIÓN DEL TITULO DE TECNÓLOGO EN ELECTRÓNICA Y TELECOMUNICACIONES

IVÁN MARCELO QUINTANILLA PINOS.
Ivanmarcelo27@hotmail.com

DIRECTOR: ING. PABLO LÓPEZ
Pablo.lopez@epn.edu.ec

Quito, MARZO 2012

DECLARACIÓN

Yo, Iván Marcelo Quintanilla Pinos declaro bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

_____

Iván Marcelo Quintanilla Pinos

RESUMEN

Hoy en día microcontroladores van siendo implementados en circuitos de automatización y en prototipos de robots, son utilizados tanto en la industria como en el hogar, haciendo movimientos similares al de seres humanos y seres vivos.

Para lograr el objetivo propuesto se realizó la construcción e implementación de un prototipo de robot que realice movimientos, atrás adelante arriba, abajo y angulares se utiliza un microcontrolador 16F886 que permite mover servomotores de manera independiente y recibe de sus sensores de luz las señales analógicas convirtiéndoles en digitales y produciendo los movimientos deseados.

El proyecto está conformado por cuatro etapas definidas, la primera etapa está compuesta por la fuente de poder conformada por baterías AAA las cuales alimentan al circuito electrónico la siguiente etapa es la analógica que tiene sensores de luz llamados fototransistores y diodos emisores de luz infrarroja, la tercera etapa es la etapa de control realiza por el microcontrolador 16F886 y la siguiente etapa de movimiento realizada por los servomotores con sus respectivos componentes añadidos para que realicen movimientos de acuerdo a los factores externos.

El prototipo del robot es un sistema que puede reconocer obstáculos en la parte delantera y hacer movimientos autónomos.

En el Capítulo I se abordará algunos fundamentos teóricos que sirven para la elaboración del proyecto así también de las características de los elementos utilizados.

En el Capítulo II se explicará cómo se diseñó e implemento en éste proyecto, el software utilizado, detallando también los modos de operación que sirvieron para su correcto funcionamiento.

En el Capítulo III se exponen los resultados de las pruebas realizadas, se analizan las ventajas y desventajas y así como también las conclusiones y recomendaciones del proyecto terminado.

INTRODUCCIÓN

La robótica aplicada a la industria ha hecho posible automatizar muchos procesos, disponer de dispositivos más efectivos y medir variables como la presión con mayor exactitud que cualquier otro método disponible.

La robótica combina diversas aplicaciones como son: la mecánica, la electrónica, la informática, la inteligencia artificial y la ingeniería de control.
La robótica ha sido implementada ya en la industria reduciendo horas hombre de trabajo, los robots y los prototipos de robots pueden hacer movimientos repetitivos y más precisos que los seres humanos.

En Ecuador algunas industrias utilizan robots para embalaje, reconocimiento de cañerías y más aplicaciones reduciendo el riesgo del hombre en algunos procesos en los cuales puede ser peligroso.

El robot hoy en día cumple un factor importante, no tiene equivocaciones, no tiene horario de trabajo.

La principal aplicación de los robots tiene lugar en la industria, la repetición de tareas, como la fabricación en serie de piezas y maquinaria, lo que obliga a realizar tareas exactamente iguales. Un robot está programado para realizar los mismos movimientos y con la misma precisión, por lo que es perfecto para aplicaciones industriales.

También la industria de los juguetes ha implementado microcontroladores para sus diseños los cuales hacen juguetes más interactivos, educativos e interesantes.

CONTENIDO

CAPITULO I: FUNDAMENTOS TEÓRICOS Y CONCEPTUALES 1
1.1 El PIC 16F886 1
1.1.1 CARACTERÍSTICAS DEL PIC 16F886 1
1.1.1.1 Distribución de pines 2
1.1.1.2 Organización de memoria 3
1.2 DISPOSITIVOS OPTOELECTRÓNICOS 4
1.2.1 INTRODUCCIÓN 4
1.2.2 Fototransistores 5
1.2.2.1 FUNCIONAMIENTO 5
1.2.2.2 CARACTERISTICAS 6
1.2.3 LEDS 7
1.2.3.1 FUNCIONAMIENTO 7
1.2.3.2 CARACTERISTICAS 8
1.2.4 LED INFRARROJO IRLED 9
1.2.4.1 FUNCIONAMIENTO 9
1.2.4.2 CARACTERISTICAS 10
1.2.4.3 CIRCUITO 10
1.2.5 Servomotores. 11
1.2.5.1 FUNCIONAMIENTO 11
1.2.5.2 CARACTERISTICAS 13
1.2.5.3 CIRCUITO 14
CAPITULO II: IMPLEMENTACIÓN DEL PROTOTIPO 16
2.1 DIAGRAMA DE BLOQUES DEL CIRCUITO 16
2.2 PLANTEAMIENTO DEL PROYECTO 17
2.3 PROYECCIÓN DEL DISEÑO 17
2.4 CONSTRUCCIÓN DEL PROTOTIPO DEL ROBOT 17
2.4.1 PARTES DE LA ESTRUCTURA DEL PROTOTIPO DEL ROBOT 18
2.4.1.2 Fuente de alimentación 20
2.4.1.3 FOTOTRANSITORES Y LEDS INFRARROJOS IRLED 22
2.4.1.4 Descripción de la base del robot. 24
2.4.1.5. Diagramas de los sensores de la base dos traseros dos delanteros 25
2.4.1.6 Circuito conexiones y puertos utilizados en el microcontrolador. . 26
2.4.1.7 CIRCUITO COMPLETO DEL ROBOT. 27
2.4.2 IMPLEMENTACION DEL SOFTWARE 29
2.4.2.1 Declaraciones del compilador PBP 29
2.4.2.2 DEFINICIÓN DE CONSTANTES USADAS 32
2.4.2.3 DEFINICIÓN DE ENTRADA Y SALIDA DE PINES 33
2.4.2.4 DEFINICIÓN DE VARIABLES USADAS 33

2.4.2.5 INICIALIZACION DEL PROGRAMA. 34
2.4.2.6 RUTINA MAIN. 35
2.4.2.7 Subrutina IRojos 35
2.4.2.8 Subrutina IRrango 39
2.4.2.9. Subrutina Ojossensores. 45
2.4.2.10 Subrutina Objetodeteccion 49
2.4.2.11 Subrutina Luz chasis 50
2.4.2.12 Subrutina Senti 50
2.5 MODOS DE OPERACIÓN Y PROGRAMACIÓN 60
2.6 DESCRIPCIÓN DE SU FUNCIONAMIENTO 61
2.7 MANUAL DE USUARIO 62
CAPITULO III: ANÁLISIS DE RESULTADOS 64
3.1 COMPORTAMIENTO A FASE EXPERIMENTAL 64
3.1.1Prueba del sistema 64
3.1.2 Prueba del robot con luz directa de luz solar y luz incandescente en superficie plana. 64
3.1.3 Prueba del robot sin luz solar directa y luz incandescente en superficie plana. 64
3.2 CONCLUSIONES 65
3.3 RECOMENDACIONES 66
DETALLE DE COSTOS PARA LA IMPLEMENTACIÓN DEL SISTEMA 67
BIBLIOGRAFÍA 68
DIRECCIONES ELECTRONICAS 68

.

CAPITULO I: FUNDAMENTOS TEÓRICOS Y CONCEPTUALES

1.1 El PIC 16F886

El PIC16F886 I/sp de Microchip es un potente microcontrolador CMOS FLASH de 8 bits puede operar a frecuencias de reloj de hasta de 20 MHz. fácil de programar ya que tiene 35 instrucciones.

1.1.1 CARACTERÍSTICAS DEL PIC 16F886

El PIC16F886 i/sp tiene una memoria de datos EEPROM de 256 bytes, una memoria de programa de 8 kbytes, una memoria de datos RAM de propósito general de 256 bytes, 2 comparadores análogos, 28 pines de entrada/salida, una referencia de voltaje programable y dos temporizadores.
Conversión análoga digital de 12 canales de 10 bits.
Las ventajas de este PIC con respecto a otros PIC's se puede visualizar en la tabla 1.1

Tabla 1.1 TABLA DE LA MEMORIA DEL PIC

Device	Program Memory	Data Memory		I/O	10-bit A/D (ch)	ECCP/ CCP	EUSART	MSSP	Comparators	Timers 8/16-bit
	Flash (words)	SRAM (bytes)	EEPROM (bytes)							
PIC16F882	2048	128	128	24	11	1/1	1	1	2	2/1
PIC16F883	4096	256	256	24	11	1/1	1	1	2	2/1
PIC16F884	4096	256	256	35	14	1/1	1	1	2	2/1
PIC16F886	8192	368	256	24	11	1/1	1	1	2	2/1
PIC16F887	8192	368	256	35	14	1/1	1	1	2	2/1

1.1.1.1 Distribución de pines

La distribución de pines del PIC16F886 i/sp se muestra en la figura 1.1, y tiene lo siguiente.

-Pin 1 que es utilizado para resetear el microcontrolador.

-Las 4 entradas para los conversores análogo digital del pin 2-5 el cual tiene 14 modos y se puede tener referencias de voltajes programados. Y Tiene dos comparadores internos.

-Los puertos 6,7 que son utilizados para programación puede ser conectado al puerto serial.

-Los pines 9 y 10 son utilizados para el resonador externo.

-Los pines 11 a 14 son salidas, controlan diferentes dispositivos.

- Los pines 15-18 funcionan como entradas o salidas, pueden ser utilizados para recepción de datos o salida control de los dispositivos externos.

-Ocho salidas las cuales puede ser agrupadas de acuerdo a la necesidad y control de diferentes dispositivos y de control.

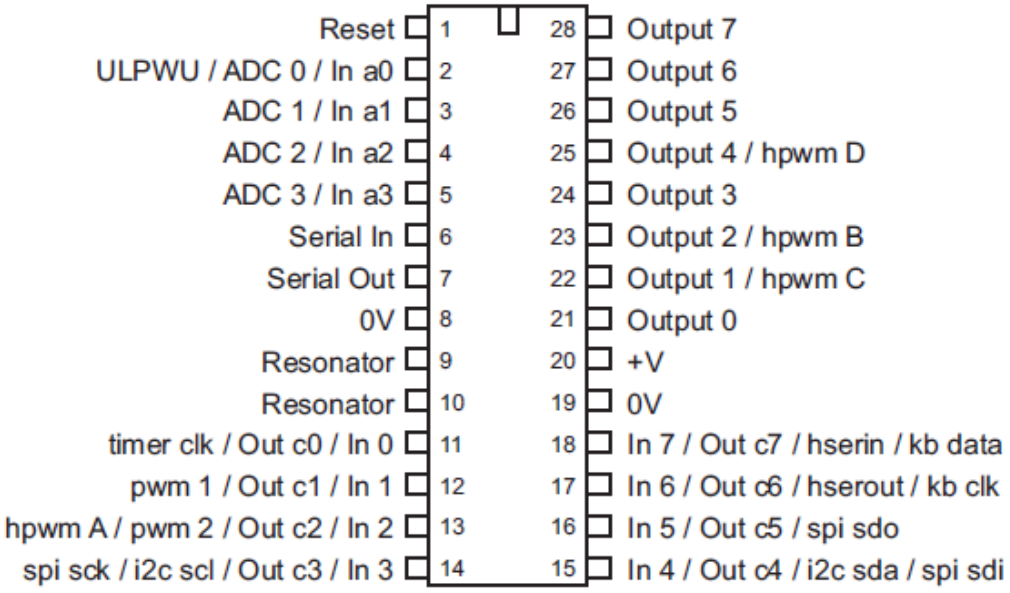


Figura 1.1 PINES MICROCONTROLADOR 16F886

1.1.1.2 Organización de memoria

El PIC16F886 i/sp posee un contador de programa de 14 bits, capaz de direccionar un espacio de memoria de 368x8 Sin embargo, únicamente los primeros 8Kx14, desde 0000h hasta 1FFFh, están implementados. Los vectores de reset e interrupción están en las direcciones 0000h y 0004h, respectivamente. La pila (stack) es de 8 niveles, lo cual significa que puede soportar hasta 8 direcciones de retorno de subrutina.
La memoria RAM está dividida en 4 bancos los cuales contienen los registros de propósito general y los registros de funciones especiales (SFR) que están ubicadas en las primeras 32 localidades de cada banco. Ver figura 1.2.

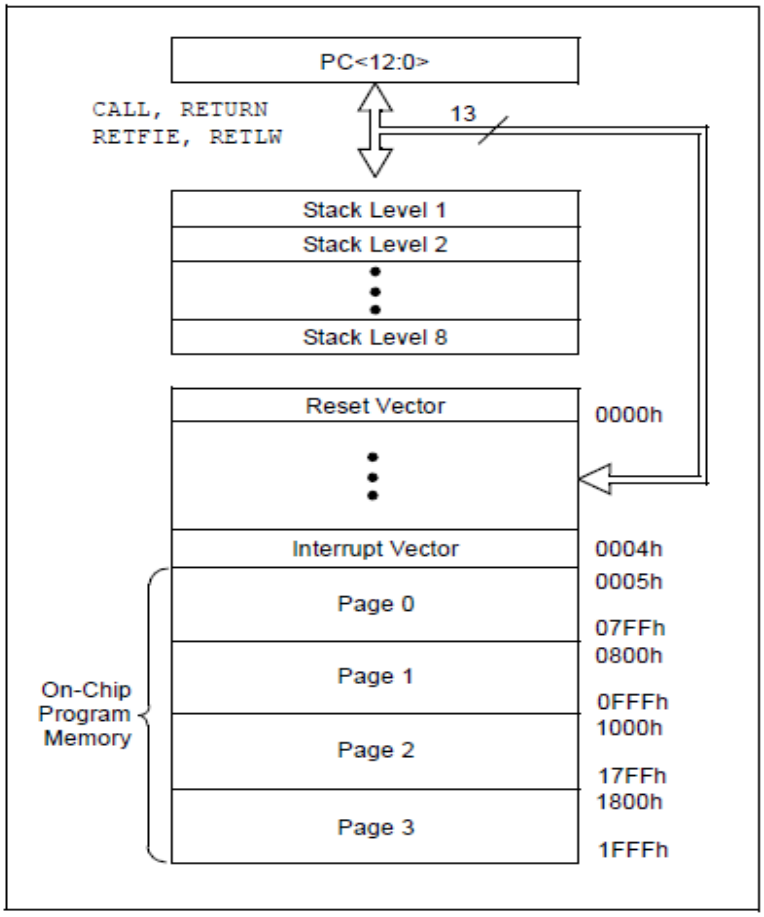


FIGURA 1.2 ORGANIZACIÓN DE LA MEMORIA DEL PIC

1.2 DISPOSITIVOS OPTOELECTRÓNICOS

1.2.1 INTRODUCCIÓN

Los dispositivos ópticos son aquellos que responden a la radiación de la luz, o que emiten radiación. Estos dispositivos responden a una frecuencia específica de radiación. Básicamente hay tres bandas en el espectro óptico de frecuencias:

Infrarrojo: Esta banda corresponde a las longitudes de onda de la luz que son muy largas y no son vistas por el ojo humano.

Visible: Corresponde a las longitudes de onda a las cuales responde el ojo humano. Comprende aproximadamente entre los 400nm y 800nm de longitud de onda. En esta banda están comprendidos todas las frecencias que el ojo humano distingue:

Ultravioleta: Longitudes de onda que son muy cortas no son vistas por el ojo humano.

Estos dispositivos se les llaman electroluminiscentes. Es importante aclarar que la mayoría de cristales de semiconductores al ser bombardeados con fotones, calor o electrones emiten luz visible o en la banda infrarroja. Sin embargo, específicamente llamamos electroluminiscentes a aquéllos que responden a la corriente eléctrica. Al aplicarle una corriente a dichos dispositivos, los electrones se mueven del material N hacia el P y se combinan con los huecos. Cuando los electrones se mueven del alto estado energético de la banda de conducción al bajo estado energético de la banda de valencia, fotones de energía son liberados. Dichos materiales pueden emitir luz visible, o como en el caso de los diodos infrarrojos luz infrarroja. Tenemos también los display , y los LEDS.

Los dispositivos llamados fotodetectores la energía que entra al cristal de semiconductor excitan a los electrones a niveles más altos de energía, dejando huecos atrás. Posteriormente estos electrones y huecos se alejan unos de otros, conformando una corriente eléctrica. El cual la radiación se representa como fotones que inciden sobre el material semiconductor, y la corriente se observa como electrones que abandonan el material.

Fotoemisores, que emiten electrones cuando existe energía radiante que incide sobre material sensible a dicha radiación. Dentro de esta categoría encontramos los fototransistores y los LASCR.

Fotoconductores, que cambian su resistencia eléctrica debido a la exposición a energía radiante tenemos los fotoresistores y los fotodiodos.

Fotovoltaicos, los cuales generan un voltaje al exponerse a energía radiante tenemos las fotoceldas.

1.2.2 Fototransistores

1.2.2.1 FUNCIONAMIENTO

El fototransistor no es muy diferente de un transistor normal, es decir, están compuestos por el mismo material semiconductor, tienen dos junturas y las mismas tres conexiones externas: colector, base y emisor. Por supuesto, siendo un elemento sensible a la luz, la primera diferencia evidente es en su cápsula, que posee una ventana o es totalmente transparente, para dejar que la luz ingrese hasta las junturas de la pastilla semiconductora y produzca el efecto fotoeléctrico.

Cuando la luz incide en este elemento se crea una corriente foto inducida que es la corriente de base

Se puede utilizar las dos en forma simultánea, aunque el fototransistor se utiliza principalmente el Pin de la base sin conectar IB=0

La corriente de base total es igual a corriente de base (modo común) + corriente de base (por iluminación). Si se desea aumentar la sensibilidad del transistor, debido a la baja iluminación, se puede incrementar la corriente de base (IB), con ayuda de polarización externa el fototransistor es un elemento muy utilizado en la robótica.

1.2.2.2 CARACTERISTICAS

Es importante decir que todos los transistores son sensibles a la luz, pero los fototransistores están diseñados para aprovechar esta característica. Existen transistores FET (de efecto de campo), que son muy sensibles a la luz, pero encontramos que la mayoría de los fototransistores consisten en una unión NPN con una región de base amplia y expuesta, como se muestra en la figura 1.3

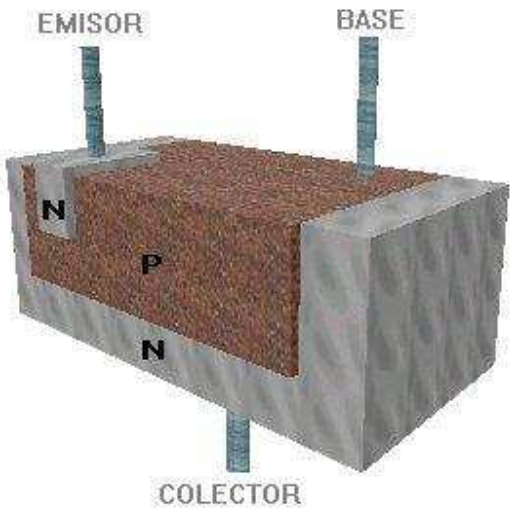


FIGURA 1.3 CAPAS DEL FOTOTRANSITOR

Al exponer el fototransistor a la luz, los fotones entran en contacto con la base del mismo, generando huecos y con ello una corriente de base que hace que el transistor entre en la región activa, y se presente una corriente de colector a emisor. Es decir, los fotones en este caso, reemplazan la corriente de base que normalmente se aplica eléctricamente. Es por este motivo que a menudo la patilla correspondiente a la base está ausente del transistor. La características más sobresaliente de un fototransistor es que permite detectar luz y amplificar mediante el uso de un sólo dispositivo (Ib = 0).

La sensibilidad de un fototransistor es superior a la de un fotodiodo, ya que la pequeña corriente foto generada es multiplicada por la ganancia del transistor.

1.2.2.3 CIRCUITOS

El circuito equivalente de un fototransistor, es un transistor común con un fotodiodo conectado entre la base y el colector, con el cátodo del fotodiodo conectado al colector del transistor y el ánodo a la base. El fototransistor es muy utilizado para aplicaciones donde la detección de iluminación es muy importante. Como el **fotodiodo**, tiene un tiempo de respuesta muy corto.

En la figura 1.4 se puede ver el circuito equivalente de un fototransistor. Se observa que está compuesto por un fotodiodo y un transistor. La corriente que entrega el fotodiodo (circula hacia la base del transistor) se amplifica ß veces, y es la corriente que puede entregar el fototransistor. Beta es la ganancia del fototransistor.

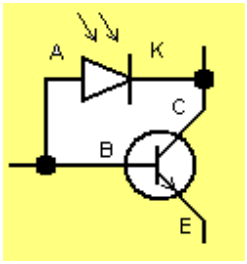


FIGURA 1.4 CIRCUITO EQUIVALENTE DEL FOTOTRANSIOR

1.2.3 LEDS

1.2.3.1 FUNCIONAMIENTO

El DIODO EMISOR DE LUZ

Los materiales semiconductores, un electrón al pasar de la banda de conducción a la de valencia, pierde energía; esta energía perdida se puede manifestar en forma de un

fotón desprendido, con una amplitud, una dirección y una fase aleatoria. El que esa energía se manifieste en (calor por ejemplo) va a depender principalmente del tipo de material semiconductor. Cuando Al polarizar directamente un diodo LED conseguimos que por la unión PN sean inyectados huecos en el material tipo N y electrones en el material tipo P; los huecos de la zona p se mueven hacia la zona n y los electrones de la zona n hacia la zona p, produciéndose por consiguiente, una inyección de portadores mayoritarios.

Ambos desplazamientos de cargas constituyen la corriente que circula por el diodo. Si los electrones y huecos están en la misma región, pueden recombinarse, es decir, los electrones pueden pasar a ocupar los huecos, cayendo" desde un nivel energético superior a otro inferior más estable, los portadores se recombinan se produce una liberación de energía una parte de esa energía produce luz. Se puede apreciar en la figura 1.5.

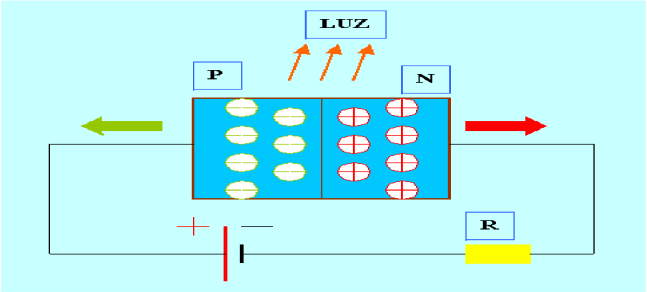


Figura 1.5 Diodo emisor de luz polarizado directamente

1.2.3.2 CARACTERISTICAS

El diodo LED tiene mayor eficiencia energética, mayor resistencia a las vibraciones, mejor visión ante diversas circunstancias de iluminación, menor disipación de energía, menor riesgo para el medio ambiente, capacidad para operar de forma intermitente de modo continuo, respuesta rápida. Con LED’s se puede producir luces de diferentes colores con un rendimiento luminoso elevado, a diferencia de muchas de las lámparas

incandescentes utilizadas hasta ahora, que tienen filtros para lograr un efecto similar (lo que supone una reducción de su eficiencia energética). Todo ello pone de manifiesto las numerosas ventajas que los LED ofrecen. También se utilizan en la emisión de señales de luz que se trasmiten a través de fibra óptica.

Los LED tienen diferentes tamaños, formas y colores. Tenemos LED redondos, cuadrados, rectangulares, triangulares y con diversas formas.

Los colores básicos son rojo, verde y azul, aunque podemos encontrarlos naranjas, amarillos incluso hay un Led de luz blanca.

Las dimensiones en los LED redondos son 3mm, 5mm, 10mm y uno gigante de 20mm. Los de formas poliédricas suelen tener unas dimensiones aproximadas de 5x5mm.

1.2.3.3 CIRCUITO

La energía contenida en un fotón de luz es proporcional a su frecuencia, es decir, su color. Cuanto mayor sea el salto de banda de energía del material semiconductor que forma el LED, más elevada será la frecuencia de la luz emitida. Se puede apreciar el circuito de polarización del dio LED en la figura 1.6.

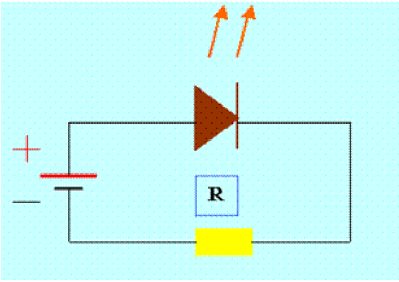


FIGURA 1.6 CIRCUITO DE POLARIZACIÓN DEL DIODO LED

1.2.4 LED INFRARROJO IRLED

1.2.4.1 FUNCIONAMIENTO

Un LED IR, siglas en inglés de Infra-Red-Emitting-Diode (diodo emisor de luz infrarroja) es un dispositivo semiconductor (diodo) que emite luz casi-monocromática, es decir, con un espectro muy angosto, cuando se polariza de forma directa y es atravesado por una corriente eléctrica. El color, longitud de onda, depende del material semiconductor

empleado en la construcción del diodo, pudiendo variar desde el ultravioleta, pasando por el espectro de luz visible, hasta el infrarrojo, recibiendo éstos últimos la denominación de IRLED Diodo emisor de luz infrarroja.

1.2.4.2 CARACTERISTICAS

Los diodos infrarrojos (IRLED) se emplean en mandos a distancia de televisores, habiéndose generalizado su uso en otros electrodomésticos como equipos de aire acondicionado, equipos de música, etc. y en general para aplicaciones de control remoto, así como en dispositivos detectores, además de ser utilizados para transmitir datos entre dispositivos electrónicos como en redes de computadoras y dispositivos como teléfonos móviles, computadoras de mano. Podemos apreciar en la figura 1.7, las parte de un diodo IRLED.

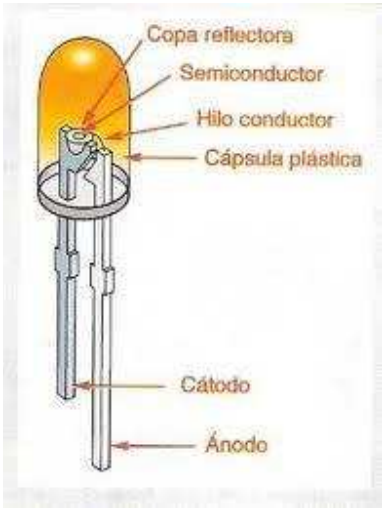


FIGURA 1.7 DIODO LED INFRARROJO

1.2.4.3 CIRCUITO

Podemos ver en la figura 1.8 el circuito de un IRLED con un fototransistor para su activación para que el fototransistor entre en corte y saturación es un circuito muy empleado en la robótica para detectar objetos.

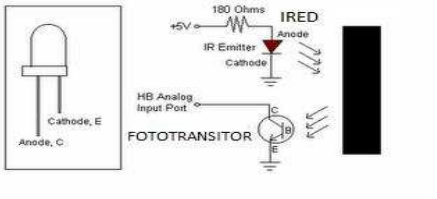


Figura 1.8 IRLED CON FOTOTRANSITOR

1.2.5 Servomotores.

1.2.5.1 FUNCIONAMIENTO

Un Servo es un dispositivo pequeño que tiene un eje de rendimiento controlado. El servomotor puede ser llevado a posiciones angulares específicas al enviar una señal codificada. Con tal de que una señal codificada exista en la línea de entrada, el servo mantendrá la posición angular del engranaje. Cuando la señal codificada cambia, la posición angular de los piñones cambia. En la práctica, se usan servos para posicionar superficies de control como el movimiento de palancas, pequeños ascensores y timones. Ellos también se usan en radio control, títeres, y por supuesto, en robots.

La modulación por ancho de pulso, PWM (*Pulse Width Modulation*), es una de las empleadas para el control de servos. Este sistema consiste en generar una onda cuadrada en la que se varía el tiempo que el pulso está a nivel alto, manteniendo el mismo período normalmente, con el objetivo de modificar la posición del servo según se desee como muestra en la figura 1.9.

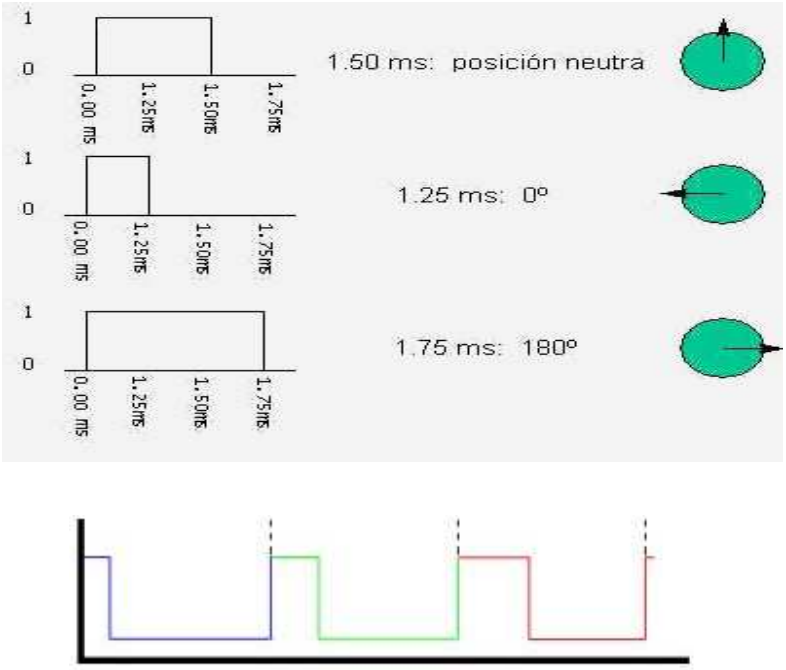


Figura 1.9 PULSOS PARA SERVO

El motor del servo tiene algunos circuitos de control y un potenciómetro (una resistencia variable) esta es conectada al eje central del servo motor. En la figura se puede observar al lado derecho del circuito. Este potenciómetro permite a la circuitería de control, supervisar el ángulo actual del servo motor. Si el eje está en el ángulo correcto, entonces el motor está apagado. Si el circuito chequea que el ángulo no es el correcto, el motor girará en la dirección adecuada hasta llegar al ángulo correcto. El eje del servo es capaz de llegar alrededor de los 180 grados. Normalmente, en algunos llega a los 210 grados, pero varía según el fabricante. Un servo normal se usa para controlar un movimiento angular de entre 0 y 180. Puede apreciar la parte interna del servomotor en la figura 1.10

La cantidad de voltaje aplicado al motor es proporcional a la distancia que éste necesita viajar. Así, si el eje necesita regresar una distancia grande, el motor regresará a toda velocidad. Si este necesita regresar sólo una pequeña cantidad, el motor correrá a una velocidad más lenta. A esto se le llama control proporcional

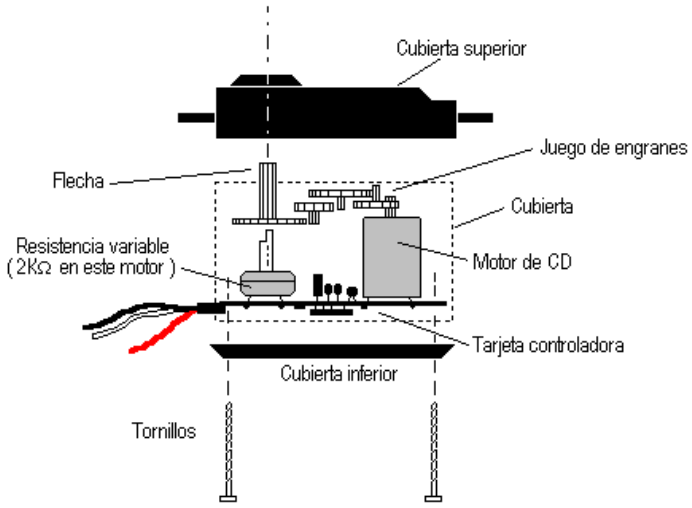


Figura 1.10 Parte interna servomotor

1.2.5.2 CARACTERISTICAS

Para el control de un servo se limita a indicar en qué posición se debe situar. Esto se lleva a cabo mediante una serie de pulsos tal que la duración del pulso nos indica el ángulo de giro del motor. Cada servo tiene sus márgenes de operación, que se corresponden con el ancho del pulso máximo y mínimo que el servo. Los valores más generales se corresponden con pulsos de entre 1 ms y 2 ms de anchura, que dejarían al motor en ambos extremos (0º y 180º). El valor 1.5 ms indicaría la posición central o neutra (90º), mientras que otros valores del pulso lo dejan en posiciones intermedias. Estos valores suelen ser los recomendados, sin embargo, es posible emplear pulsos menores de 1 ms o mayores de 2 ms, pudiéndose conseguir ángulos mayores de 180°. Si se sobrepasan los límites de movimiento del servo, éste comenzará a emitir un zumbido, indicando que se debe cambiar la longitud del pulso. El factor limitante es el tope del potenciómetro y los límites mecánicos constructivos.

El período entre pulso y pulso (tiempo de OFF) no es crítico, e incluso puede ser distinto entre uno y otro pulso. Se suelen emplear valores ~ 20 ms (entre 10 ms y 30

ms). Si el intervalo entre pulso y pulso es inferior al mínimo, puede interferir con la temporización interna del servo, causando un zumbido, y la vibración del eje de salida.

Es importante destacar que para que un servo se mantenga en la misma posición durante un cierto tiempo, es necesario enviarle continuamente el pulso correspondiente. De este modo, si existe alguna fuerza que le obligue a abandonar esta posición, intentará resistirse. Si se deja de enviar pulsos (o el intervalo entre pulsos es mayor que el máximo) entonces el servo perderá fuerza y dejará de intentar mantener su posición, de modo que cualquier fuerza externa podría desplazarlo. El servo tiene tres cables el rojo de polaridad positiva el negro GND y el blanco de control. Los pulsos de control servo motor puede apreciarse en la figura 1.11.

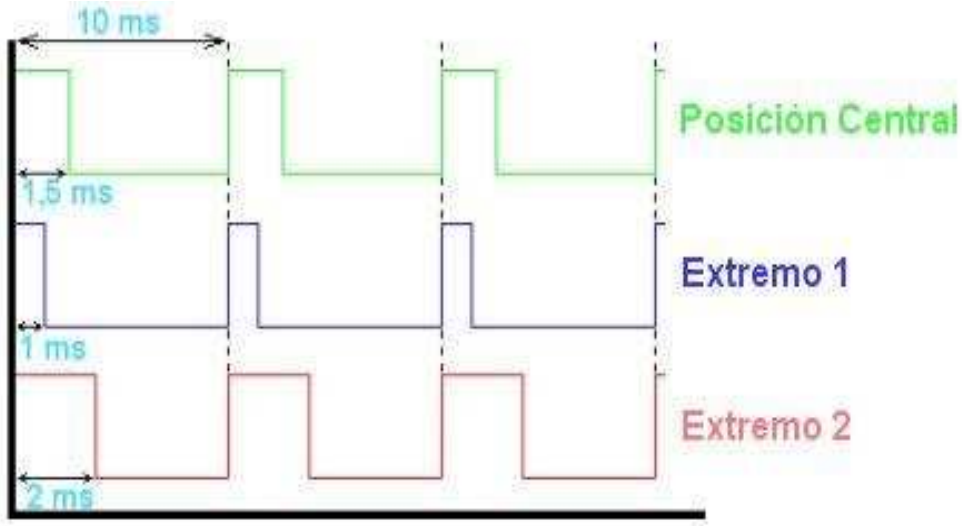


Figura 1.11 PULSOS PARA EL CONTROL SERVOMOTOR

1.2.5.3 CIRCUITO

En la figura 1.12 circuito podemos apreciar un circuito para poder controlar por pulsos al servomotor.

El circuito emplea un 555 conectado de tal forma de generar un tren de pulsos ajustable por medio del potenciómetro del 10K. El transistor conectado a la salida amplia la capacidad de manejo de corriente. A medida que se gira el potenciómetro el

tren de pulsos es modificado con lo que se logra alterar el estado del servo el cual desplaza su eje en función del potenciómetro. Es recomendable emplear un potenciómetro lineal, para que el efecto sea igual en cualquier parte del recorrido del mismo.

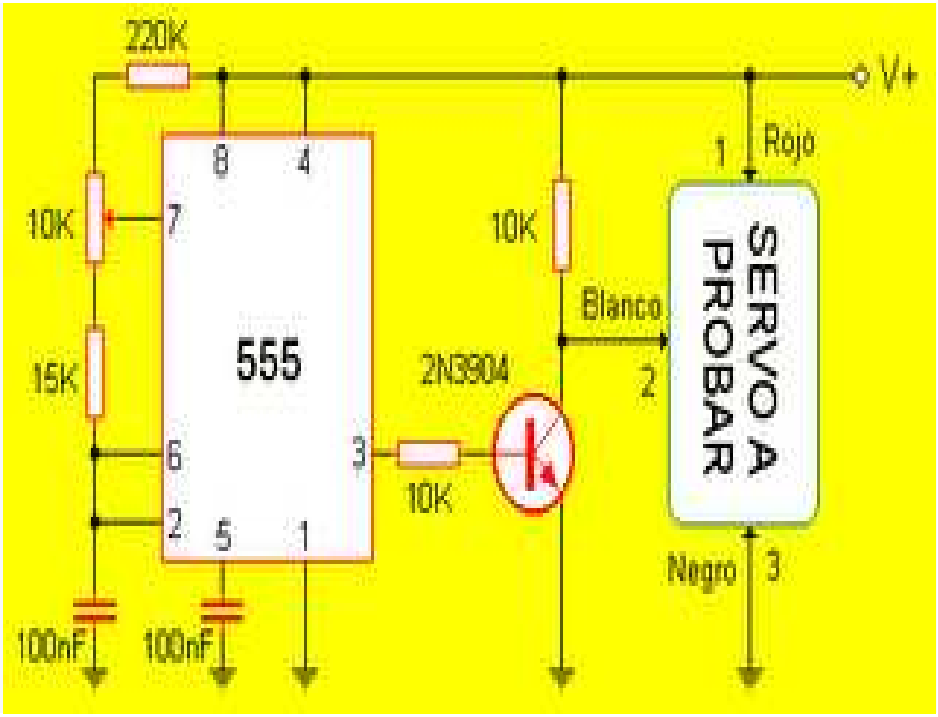


FIGURA 1.12 CIRCUITO PARA CONTROL SERVOMOTOR

REFERENCIA BIBLIOGRAFICA:WIKIPEDIA

CAPITULO II: IMPLEMENTACIÓN DEL PROTOTIPO

2.1 DIAGRAMA DE BLOQUES DEL CIRCUITO

En la figura 2.1 está especificado el diagrama de bloques del prototipo de robot que funciona con dos pares de pilas la cual alimenta al circuito, también disponen de los sensores de luz los cuales están conectados directamente al microcontrolador que permite mandar señal de control para los dos servomotores que ordenan el movimiento de atrás, adelante, circular y los otros dos servomotores que permiten el movimiento de arriba, abajo, izquierda y derecha del prototipo del robot.

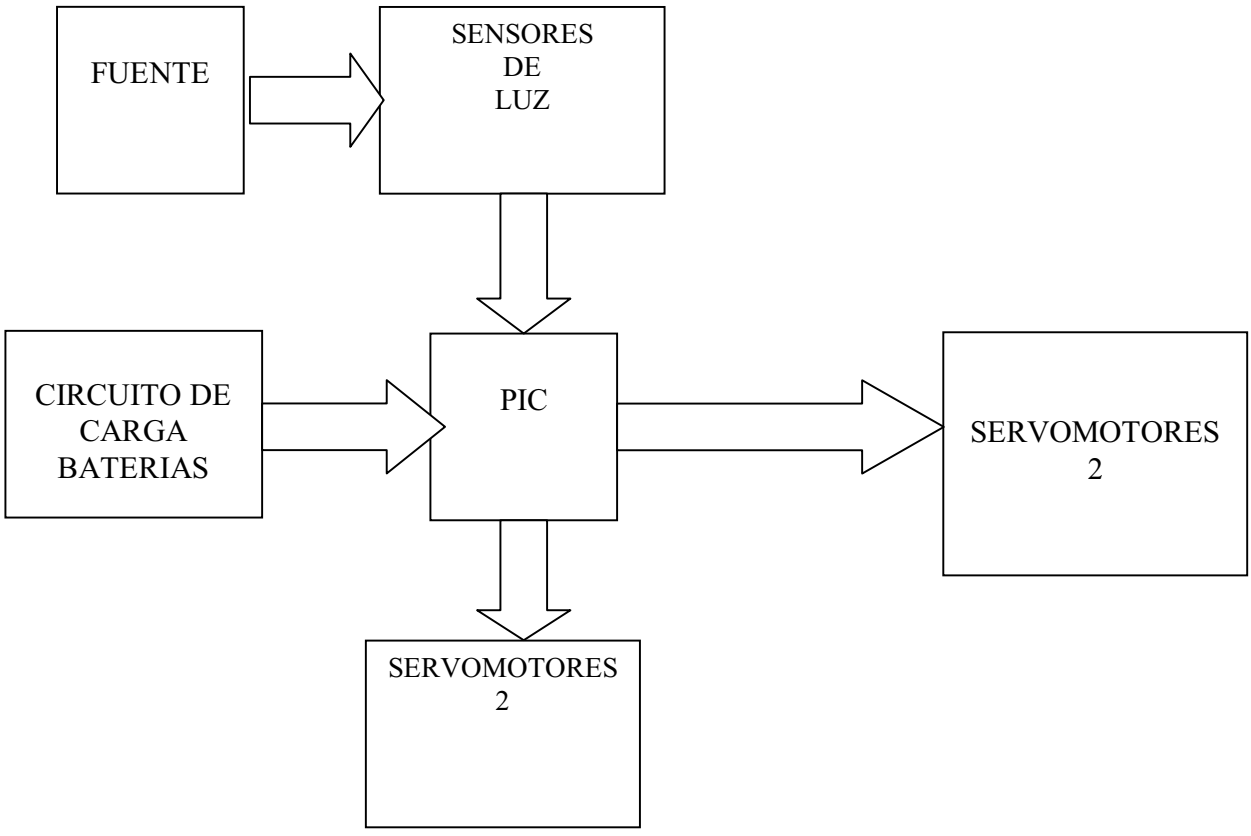


FIGURA 2.1 DIAGRAMA DE BLOQUES ROBOT

2.2 PLANTEAMIENTO DEL PROYECTO

El prototipo del robot, es un dispositivo el cual se mueve de manera autónoma de acuerdo como se presente un obstáculo frente del robot, tiene una base y una cabeza.
La base tiene movimientos atrás, adelante, y circular.
Tiene una cabeza que realiza movimientos izquierda, centro, derecha, arriba y abajo.

El robot se mueve con servomotores el microprocesador determina la velocidad y el ángulo de giro permitiendo tener movimientos autónomos.
Los servomotores actúan de manera independiente puede tener ángulos de giros de hasta 180 grados de acuerdo a la incidencia directa de los IRLED a los fototransistores.

2.3 PROYECCIÓN DEL DISEÑO

El proyecto está orientado a un prototipo el cual pueda orientar en un futuro al diseño y construcción de prototipos, robot que imiten movimientos de seres humanos o seres vivos los cuales sean autónomos y sean automáticos.
Algunas empresas están implementando la robótica para reemplazar a los seres humanos para hacer movimientos precisos y automáticos, es orientado para el estudio de prototipos robóticos y ser implementado a la industria en general.

2.4 CONSTRUCCIÓN DEL PROTOTIPO DEL ROBOT

Los siguientes temas están destinados al explicar de manera detalla del proyecto tanto del hardware como del software. Vendrá detallado en conjunto pero será explicado por partes y por módulos del robot, los programas del microcontrolador será explicado por módulos y diagramas que expliquen su respectivo funcionamiento.

2.4.1 PARTES DE LA ESTRUCTURA DEL PROTOTIPO DEL ROBOT

El robot esta fromado de varios componentes mecánicos como servomotores, piñones, montura de los sensores de luz, servomotores, ruedas, conectores. Figura 2.2, 2.3, 2.4,2.5,2.6, respectivamente.

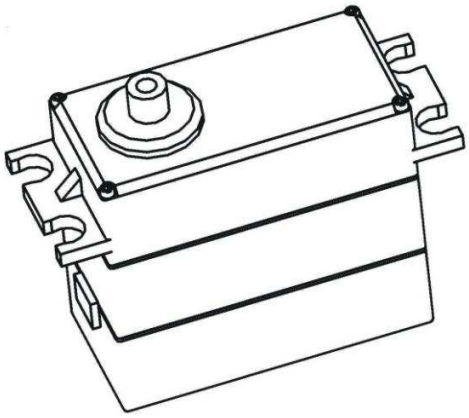


FIGURA 2.2 SERVOMOTOR CON ENGRANAJES

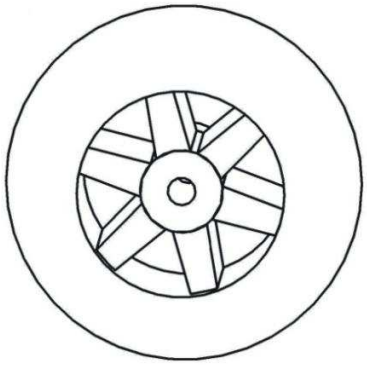


FIGURA 2.3 RUEDAS

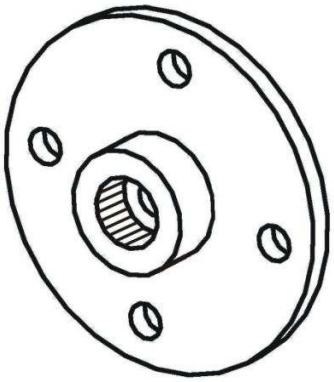


FIGURA 2.4 PIÑONES DELANTERO

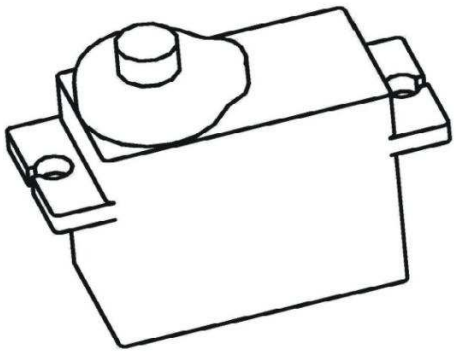


FIGURA 2.5 SERVOMOTOR

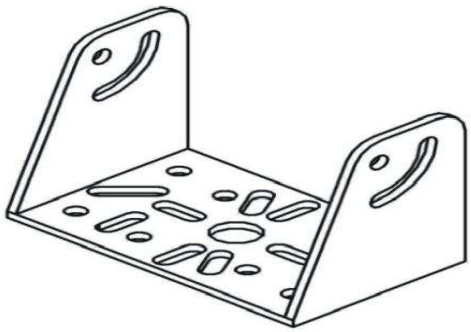


FIGURA 2.6 PLACA INFERIOR PARA SUJETAR EL SERVO DE LA CABEZA

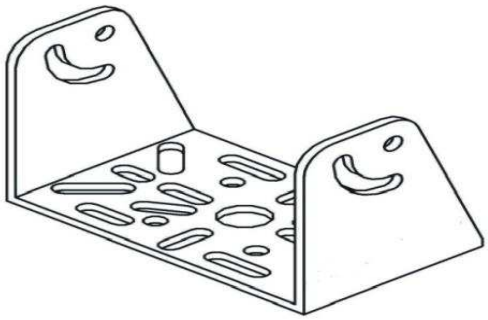


FIGURA 2.7 PLACA SUPERIOR PARA SUJETAR EL SERVOMOTOR DE LA CABEZA

Podemos apreciar en la figura 2.8 ensamblado la cabeza del robot

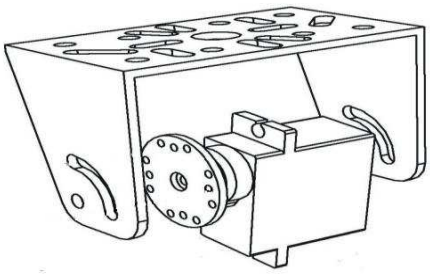


Figura 2.8 CABEZA ROBOT

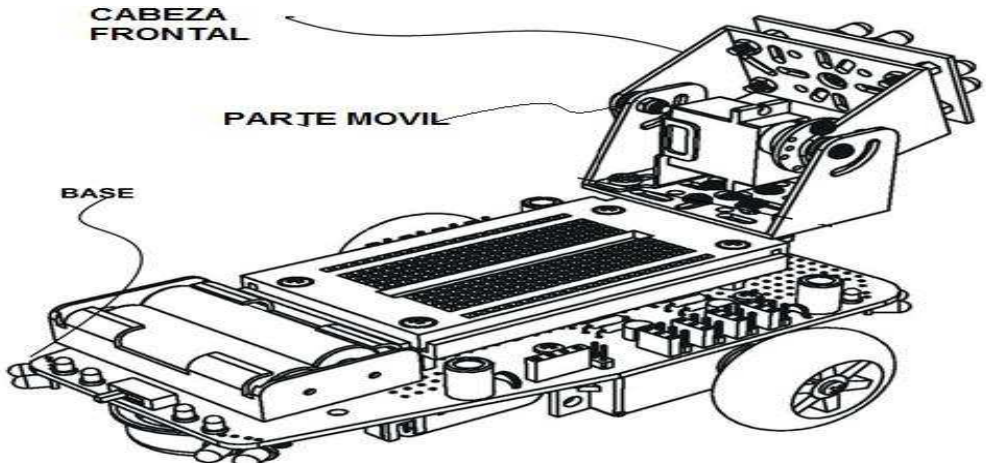


FIGURA 2.9 ROBOT EMSAMBLADO

2.4.1.2 Fuente de alimentación

Tenemos dos parte en el circuito de alimentación son las baterías recargables que alimentan a todo el robot y también el cargador externo de 9 voltios que alimenta al circuito y los jumper donde están conectados los servomotores.
La fuente de alimentación tiene un interruptor de dos posiciones el cual permite alimentar por medio de las baterías y cuando está apagado el robot permite la carga de las mismas tenemos en conexión 4 baterías AAA las cual alimentan el circuito cuando está en movimiento también un LED que nos indica cuando están en funcionamiento de color rojo.
Hemos puesto dos diodos en serie para que permita la carga en un solo sentido y no regrese el voltaje de las baterías.
Los condensadores conectados nos sirven de filtros y para mantener una carga constante a los servomotores permitiendo que no existan interferencias y no afecte el microcontrolador en su funcionamiento.

Para la carga de las baterías se usa una fuente externa de 9 voltios y una corriente aproximada para carga de 250 mA

R=V/I = 4.8V/250mA= 13.2 OHM resistencia puesta 18 OHM .Los diodos 1N4001 cumple con los requerimientos para el circuito la corriente y el voltaje de polaridad inversa ya que la batería si alimentara el circuito seria 4.8V y el diodo sería suficiente para soporta la corriente y el voltaje inverso
Así mimos la resistencia para el led seria ya que circula una corriente aproximada de 25 mA.
Rled= 4.8V-1.5V/ 20mA= 165Ω se le puso 180Ω.
Ver en la Figura 2.10

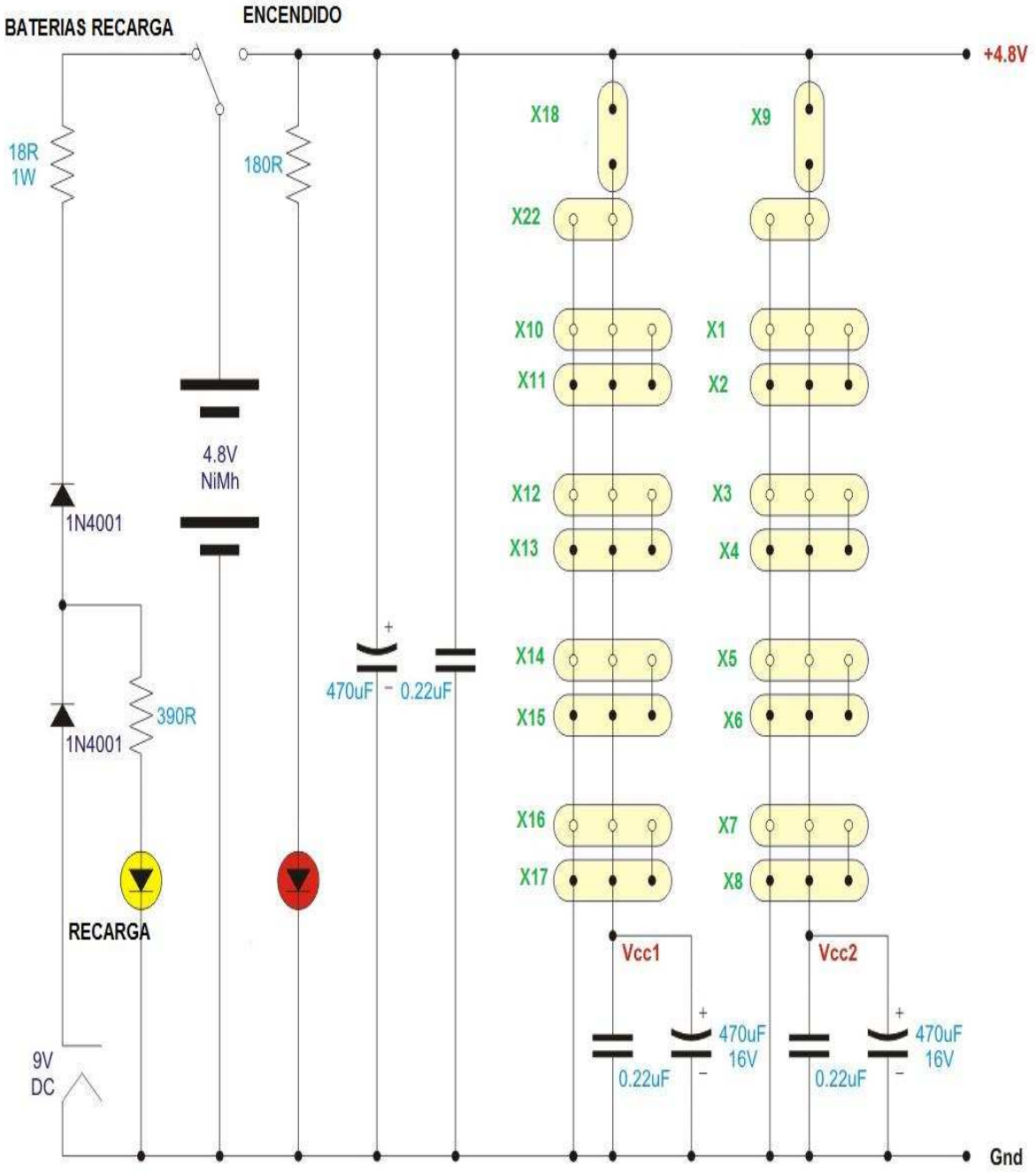


Figura 2.10 DIAGRAMA DEL CIRCUITO DE LA BASE

2.4.1.3 FOTOTRANSITORES Y LEDS INFRARROJOS IRLED

Hardware de la cabeza.

Para darle más sensibilidad al sistema se utiliza 2 grupos de 2 IRLED's en serie y 4 grupos de 2 fototransistores en cascada como se puede apreciar en la figura 2.11 .
La parte frontal funciona con luz infrarroja los IRLED's emiten luz, esta choca con algún obstáculo y se refleja, la luz reflejada es captada por los fototransistores los cual envían una señal analógica la microcontrolador.

El fototransistor a utilizarse es el PT334, y el Led infrarrojo es el IR333C, los fototransistores están conectados de par en par en emisor común para dar mayor sensibilidad y así poder detectar obstáculos. Ver en la figura 2.11

Los fototransistores trabajan en corte y saturación, cuando está en corte el voltaje en la resistencia 4K7 es igual a cero y cuando está en saturación se tiene un voltaje de 4.8 voltios.

La resistencia de protección en los LED's es de 39 OHM de acuerdo al siguiente calculo: R=(4.8V-3V)/50mA = 36OHM se aproximó a 39 OHM. Se ha puesto de ese valor para que emitan la mayor cantidad de luz infrarroja posible. A menor resistencia mayor es el brillo de diodos IRLED. De acuerdo al siguiente calculo

Para que los IRLED's se enciendan es necesario que envié una orden que sature el transistor PN100.

Hay que tener en cuenta que la resistencia para los fototransistores resistencia de 47 KΩ funciona como carga del circuito. Ver figura 2.11

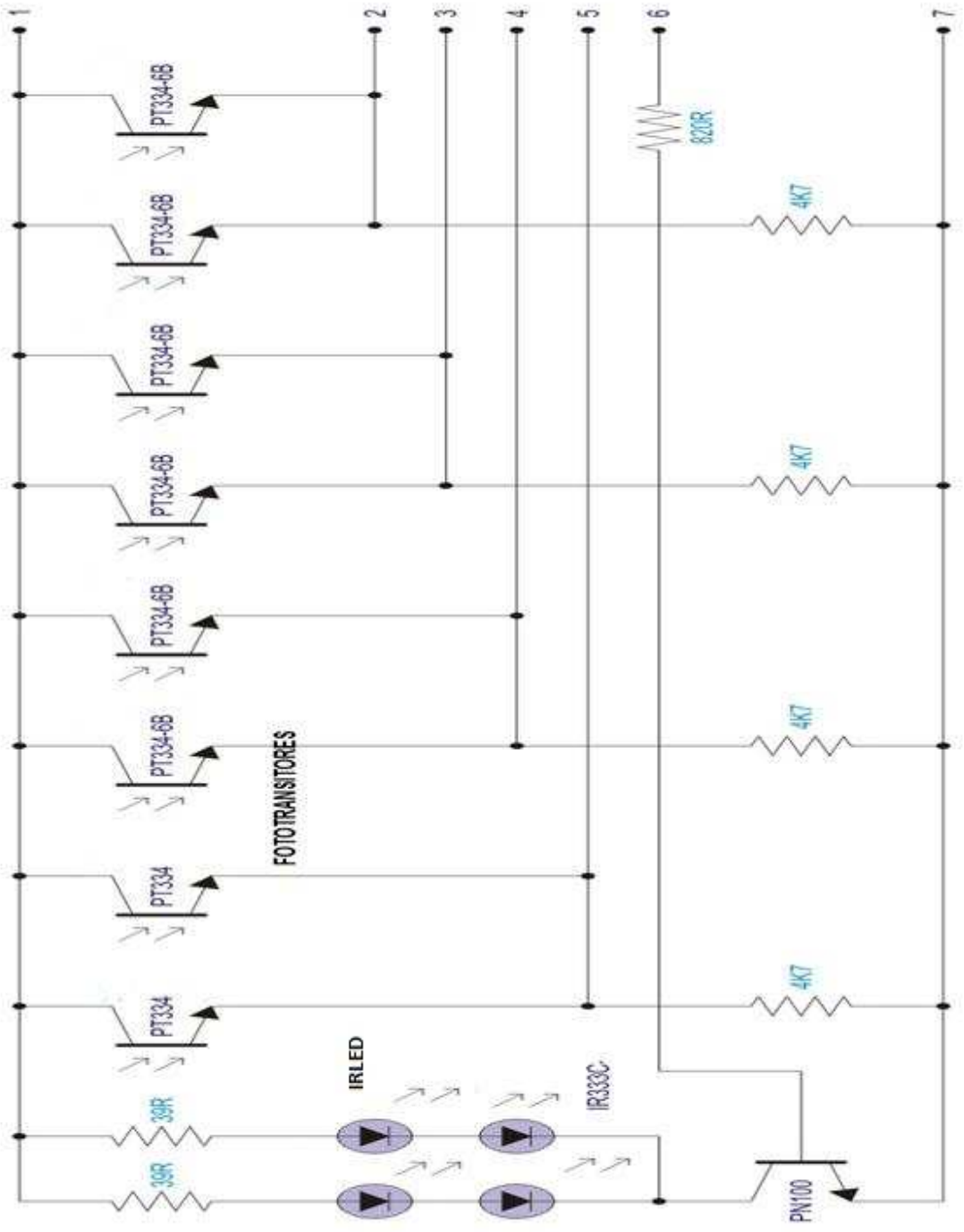


FIGURA 2.11 DIAGRAMA DE LA CABEZA ROBOT

2.4.1.4 Descripción de la base del robot.

En la figura 2.12 podemos apreciar la distribución de los sensores tanto los LEDS como los fototransistores están distribuidos y los jumper como están distribuidos.

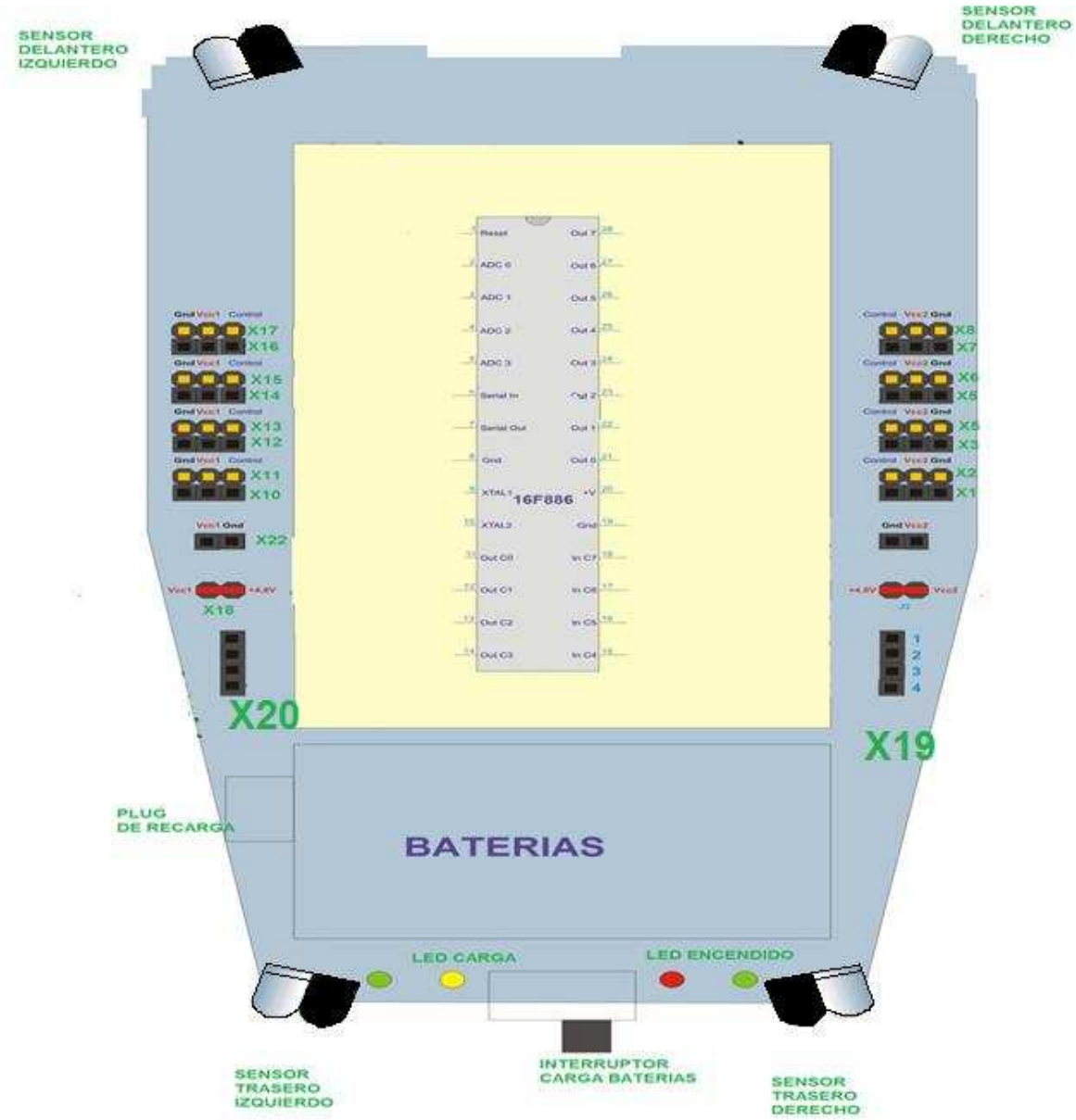


FIGURA 2.12 BASE DEL ROBOT

2.4.1.5. Diagramas de los sensores de la base dos traseros y dos delanteros.

Está conectado el IRLED y el fototransistor para que funcione de acuerdo a la incidencia de luz funcionan como detectores de objetos, los diodos IRLED emiten luz los fototransistores detectan esa luz que es reflejada por el objeto es el circuito de IRLED del lado derecho de del robot figura 2.13.
Este circuito vienen a funcionar como detectores de objetos para poder calcular la resistencia aproximadamente la circulación máxima de este circuito es de 50mA
R fotodiodo = (4.8V-1.5V)/50mA=72 Ω resistencia aproximada 100Ω que son resistencia comerciales.

La resistencia del LED R=(4.8V-1.5V)/20mA= 165 OHM se aproximó a 180 OHM.

La resistencia se le ha puesto una de 100KΩ que funciona como carga del circuito.
Cuando el IRLED se enciende también se va encender el diodo led de color verde de acuerdo a la señal que envié el microcontrolador, el BC547 transistor va estar en estado de conducción y de no conducción. Ver figura 2.13.

Para los cuatro sensores de la base el circuito es similar no varía. Ver en la figura 2.13

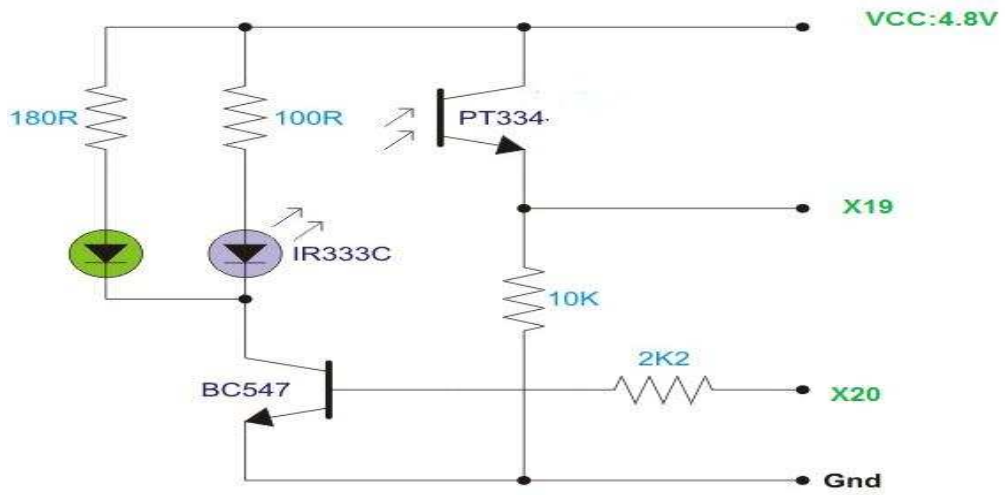


Figura 2.13 DIAGRAMA DE LOS 4 SENSORES DE LAS ESQUINAS

2.4.1.6 Circuito conexiones y puertos utilizados en el microcontrolador.

En la figura 2.15 podemos apreciar las conexiones del microcontrolador a los servomotores y los puertos que se utilizó.

PIN número 1 que es de reset, resetea el circuito cuando se aplica un voltaje entre 2.5V Y 3V. Ver figura 2.15

Los pines 2,3,4,5, son entradas desde la A0 hasta la A3 y también funcionan como conversor de análogo a digital desde ADC1 HASTA ADC3.
También se utilizó los pines del 6-9 que son para programar el PIC.

Los pines 9 y 10 que son para el resonador o cristal cerámico para este caso utilizamos uno de 16 MHZ.

También se utilizan los pines 11-14 , estas funcionan como entradas y salidas de acuerdo a la instrucción que se le dé al programa como quiere que trabaje hay que tener en cuenta que aquí en el pin 11-12 funciona la función PWM1 y PWM2 , para el circuito utilizaremos como entradas .Ver figura 2.14

Los pines del 15-18 funcionan como entradas y salidas de acuerdo a la instrucción que estemos utilizando como salidas.

Los pines 19-20 son de polarización del Pic.
Los pines 21-24 son salidas del Pic. el cual es utilizado para control de los servomotores. Ver figura 2.14

El pin 25 también es utilizado para control sirve para activar los IRLED’s de la parte frontal del robot ver figura 2.15

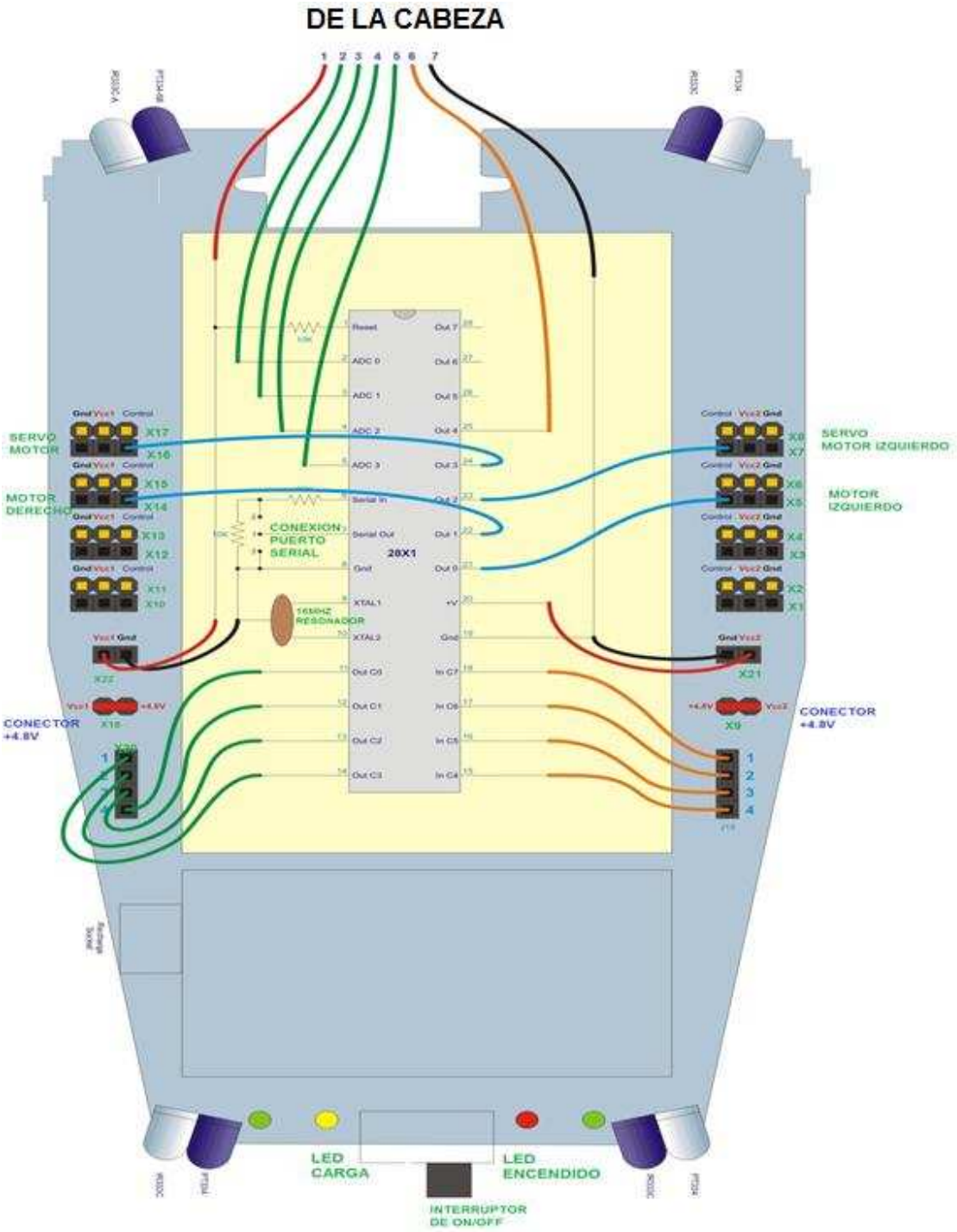


Figura 2.14 CIRCUITO MICROCONTROLADOR

2.4.1.7 CIRCUITO COMPLETO DEL ROBOT.

A continuación podemos apreciar el circuito completo del robot.

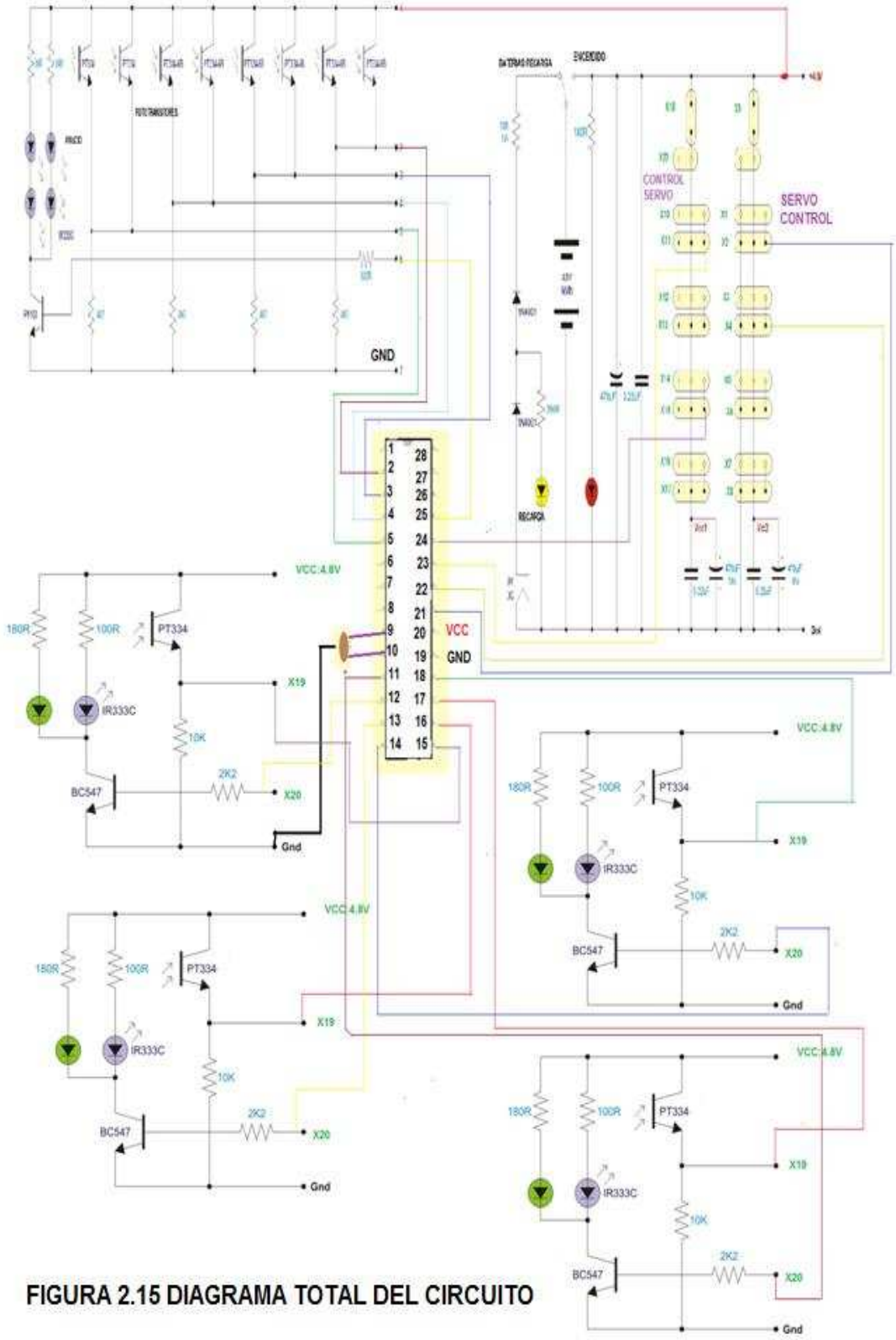


FIGURA 2.15 DIAGRAMA TOTAL DEL CIRCUITO

2.4.2 IMPLEMENTACION DEL SOFTWARE

Explicaremos paso a paso la implementación del software que está en el anexo 1 el cual tiene su programa principal y sus subrutinas le hemos dado nombres a las rutinas principales y a las subrutinas las cuales están especificadas a continuación.
Para el funcionamiento del robot con el del PIC 16F886 i/sp, se puede utilizar lenguajes de alto o bajo nivel.
El lenguaje de alto nivel es el que se acerca al lenguaje humano, como por ejemplo el BASIC, mientras que el bajo nivel es el más próximo al lenguaje de máquina, como código binario, ensamblador, hexadecimal. etc.

El presente proyecto y por la facilidad de programación del PIC se utilizó el lenguaje programación PIC BASIC PRO (PBP), por tener declaraciones especiales entre ellas: PWM, las cuales son muy importante para la programación de cualquier Pic.

2.4.2.1 Declaraciones del compilador PBP

El PBP es un lenguaje de programación tipo BASIC el cual tiene más de 80 instrucciones disponibles que se detallan en la tabla 2.1 con una breve explicación:

Tabla 2.1 COMANDOS PARA PROGRAMACIÓN DEL PIC

DECLARACION	APLICACIÓN
@	Inserta una línea en código ensamblador
ADCIN	Lee el conversor analógico
ASM… ENDASM	Inserta una sección de código ensamblador
BRANCH	GOTO computado (equivale a ON.. GOTO)
BRANCHL	BRANCH fuera de página (BRANCH largo)
BUTTON	Anti-rebote y auto-repetición de entrada en el pin especificado
CALL	Llamada a subrutina de ensamblador
CLEAR	Hace cero todas las variables

CLEARWDT	Hace cero el contador del Watchdog Timer
COUNT	Cuenta el número de pulsos en un pin
DATA	Define el contenido inicial en un chip EEPROM
DEBUG	Señal asincrónica de salida en un pin fijo y baud
DEBUGIN	Señal asincrónica de entrada en un pin fijo y baud
DISABLE	Deshabilita el procesamiento de ON INTERRUPT, ON DEBUG
DISABLE DEBUG	Deshabilita el procesamiento de ON DEBUG
DISABLE INTERRUPT	Deshabilita el procesamiento de ON INTERRUPT
DTMFOUT	Produce tonos telefónicos en un pin
EEPROM	Define el contenido inicial en un chip EEPROM
ENABLE	Habilita el procesamiento de ON INTERRUPT, ON DEBUG
ENABLE DEBUG	Habilita el procesamiento de ON DEBUG
ENABLE INTERRUPT	Habilita el procesamiento de ON INTERRUPT
END	Detiene la ejecución e ingresa en modo de baja potencia
FOR… NEXT	Ejecuta declaraciones en forma repetitiva
FREQOUT	Produce hasta dos frecuencias en un pin
GOSUB	Llama a una subrutina en BASIC en la línea especificada
GOTO	Continúa la ejecución en la línea especificada
HIGH	Saca un 1 lógico (5 V.) por un pin
HPWM	Salida de hardware con ancho de pulsos modulados
HSERIN	Entrada serial asincrónica (hardware)
HSEROUT	Salida serial asincrónica (hardware)
I2CREAD	Lee bytes de dispositivos I2C
IC2WRITE	Graba bytes de dispositivos I2C
IF..THEN..ELSE.. ENDIF	Ejecuta declaraciones en forma condicional
INPUT	Convierte un pin en entrada
LCDIN	Lee caracteres desde una RAM de un LCD

LCDOUT	Muestra caracteres en un LCD
LET	Asigna el resultado de una expresión a una variable
LOOKDOWN	Busca un valor en una tabla de constantes
LOOKDOWN2	Busca un valor en una tabla de constantes o variables
LOOKUP	Obtiene un valor constante de una tabla
LOOKUP2	Obtiene un valor constante o variable de una tabla
LOW	Hace 0 lógico (0 V.) un pin específico
NAP	Apaga el procesador por un cierto período de tiempo
ON DEBUG	Ejecuta un DEBUG en BASIC
ON INTERRUPT	Ejecuta una subrutina BASIC en un interrupt
OUTPUT	Convierte un pin en salida
OWIN	Entrada de dispositivos un - alambre
OWOUT	Salida de dispositivos un - alambre
PAUSE	Demora con resolución de 1 milisegundo
PAUSEUS	Demora con resolución de 1 microsegundo
PEEK	Lee un byte del registro
POKE	Graba un byte en el registro
POT	Lee el potenciómetro en el pin especificado
PULSIN	Mide el ancho de pulso en un pin
PULSOUT	Genera pulso hacia un pin
PWM	Salida modulada en ancho de pulso por un pin especificado
RANDOM	Genera número pseudo-aleatorio
RCTIME	Mide el ancho de pulso en un pin
READ	Lee byte de un chip EEPROM
READCODE	Lee palabra desde un código de memoria
RESUME	Continúa la ejecución después de una interrupción
RETURN	Continúa en la declaración que sigue al último GOSUB
REVERSE	Convierte un pin de salida en entrada, o viceversa
SELECT CASE	Compara una variable con diferentes valores

SERIN	Entrada serial asincrónica (tipo BASIC Stamp 1)
SERIN2	Entrada serial asincrónica (tipo BASIC Stamp 2)
SEROUT	Salida serial asincrónica (tipo BASIC Stamp 1)
SEROUT2	Salida serial asincrónica (tipo BASIC Stamp 2)
SHIFTIN	Entrada serial sincrónica
SHIFTOUT	Salida serial sincrónica
SLEEP	Apaga el procesador por un período de tiempo
SOUND	Genera un tono o ruido blanco en un pin
STOP	Detiene la ejecución del programa
SWAP	Intercambia los valores de dos variables
TOGGLE	Hace salida a un pin y cambia su estado
USBIN	Entrada de USB
USBINIT	Inicializar USB
UBSOUT	Salida USB
WHILE… WEND	Ejecuta declaraciones mientras la condición sea cierta
WRITE	Graba bytes en un chip EEPROM
WRITECODE	Escritura palabra en código de memoria
XIN	Entrada X – 10
XOUT	Salida X – 10

2.4.2.2 DEFINICIÓN DE CONSTANTES USADAS

En el programa definimos constantes, al definir constantes damos valores máximos y mínimos que pueden operar el servomotor eso nos permitirá determinar el movimiento máximo de la cabeza del robot y de la base del robot , por eso los valores se han puesto de manera decimal indicándole al servomotor donde es el centro de referencia y para que empiece a moverse, indicándole cual es el movimiento máximo y mínimo que puede realizar, esto permitirá que el servomotor no funcione fuera de rango .

REFERENCIA BIBLIOGRAFICA: MANUAL PIC16F886

factorizderecha = 10 movimiento izquierda derecha y factor utilizado de escala para la velocidad del servo primario.
factorarribaabajo=14 factor de escala del servo arriba, abajo del servo inclinación servos primarios.
distanciamaxima=200 determinación de la distancia del objeto debe estar dentro de esta distancia
distanciarobot= 550 Robot trata de mantener una distancia cuando esta movimiento.
centroderecho=145 posición del servomotor de la cabeza al centro
centroizquierdo=135 posición del servomotor de la cabeza al centro
motorparadaizq=143 posición del servomotor de la base al centra
motorparadadercha=151 posición del servomotor de la base derecho
servomaximo=centroderecho+70 valor máximos del servomotor izquierda derecha.
servominimo=centroderecho-70 valor mínimo del servomotor de la cabeza.
aximainclinacion= centroizquierdo+90 máximo valor del servomotor de la base derecho.
inimainclinacion= centroizquierdo-60 máximo valor del servomotor dela base izquierdo.

2.4.2.3 DEFINICIÓN DE ENTRADA Y SALIDA DE PINES

En esta parte hemos definido las entradas y salida a utilizarse tanto las entradas analógicas y las salidas digitales hemos dado nombres a continuación explicaremos
basefotoizquierdo=1 ADC 1; basefotoderecho = 3 ADC 3; fotoarriba =0 ADC 0,
fotoabajo =2 ADC 2 , definimos entradas analógicas .
servoizquierda= 0 salida digital 0;servoderecha= 1 salida digital 1;
servoarribaabajo= 2 salida digital 2; servoizqderecha = 3 salida digital 3;
sensoresluz= 4 salida digital 4

2.4.2.4 DEFINICIÓN DE VARIABLES USADAS

Definimos variables para hacer más fácil la programación y poder encontrar futuras fallas en el programa, define variables para evaluar y testear.
tiempo= b0 variable usada para resetear el contador y dar saltos loops..
tecnirobot= b1 contador utilisado en la subrutina sentí.

servoprimario= b2 siguiente posición del servomotor de la cabeza izquierda derecha
siguienteposicion = b3 siguiente posición del servomotor
escaladelprimario= b4 factor de escala para cálculos.
escalasiguienteposicion= b5 factor de escala para la siguiente posición.
velocidadizq= b6 evalua el servomotor izquierdo velocidad y dirección. De la base
velocidaddercha= b7 evalua el servomotor derecho velocidad y dirección de la base
evaluandoled= b8 variable utilizada para los Irled's
testsensorluz= b9 variable usada en detección testsensorluz
sensoresluztodos= b10 datos de todos los testsensorluz sensores y detección
distanciaanaloga= w6 b12:b13 promedio de las entradas analógicas de los IRLED's
temporalcalculo= w7 b14:b15 variable temporal utilizada en cálculos
arribaabajosensor = w8 b16:b17 escala diferencia evalúa los sensores de luz de la base traseros
izquierdaderechasensor= w9 b18:b19 escala diferencial evalúa los sensores de luz de la base delanteros.
izquierdaIRsensor = w10 b20:b21 datos del IREDS DE LA BASE DERECHO
derechaIRsensor = w11 b22:b23 datos del IRLEDS de la base izquierdo
arribaIRsensor= w12 b24:b25 Datos de los IREDs de la cabeza
abajoIRsensor = w13 'b26:b27 Datos de los IREDS frontales

2.4.2.5 INICIALIZACION DEL PROGRAMA.

Utilizamos el comando dirsc, enciende los IRED's de la cabeza, y mueve izquierda derecha y al centro. Calculamos la variable b2 si el igual a 14 esto mueve el servomotor arriba abaja y al centro. Calculamos b3 si esto es igual a 15 es un factor del servo motor esto previene que el servo salga del rango de trabajo cuando se mueve a la izquierda o a la derecha la cabeza.
Calculamos b4 y b5 si es igual a 15 esto previene que se salga del rango de trabajo cuando se mueve hacia arriba o hacia abajo.
Activamos el los servomotores de la base izquierdo y derecho mandamos valores a los para iniciar en el centro.

Calculamos b6 si es igual a 143 y b7 es igual a 151 esto hace que se inicialice los servomotores de la base.
Activamos los servomotores de la base para ser movidos da el rango de inicio, este programa inicializa todo los servomotores. Ver Figura 2.16

2.4.2.6 RUTINA MAIN.

La salida W0 incrementamos el valor en pasos empieza desde 0.para este caso va a ser desde 0 hasta 65535 ,ponemos los loops va a incrementarse uno a uno.. Igualamos w7 a b0 y hacemos una operación lógica AND 7 y si W7 es igual a 0 manda a llamar a la subrutina luzchasis, hay que tener en cuenta que W7 nos va a servir en el programa ya que es un variable y para hacer cálculos temporales. Termina esta subrutina si el servomotor de la cabeza está a la izquierda , o el servomotor de la base se ha salido de la posición anterior ,después manda a llamar a las subrutina Senti,IRojos e IRrango. Si la entrada b0 supera un valor mayor que dos llama a las subrutinas Ojossensores,Objetodeteccion para corregir errores o detectar un objeto adelante .Ver Figura 2.17

2.4.2.7 Subrutina IRojos

Enviamos un pulso a la salida 4 del pin 25 del microcontrolador el cual pone al transistor PN100 en saturación y enciende los IRLED's de la cabeza.
Lee las entradas analógicas del microcontrolador a 10 bits, utilizamos para eso el comando readadc10, lee toda la información de los fototransistores de acuerdo a la luz emitida por los IRLED's reflejada en el objeto que esté al frente o no. Luego manda un OL a la misma salida apagando los IRLED's,
Después calculamos la variables w10 , el valor de w10 lo restamos de la variable w7 esto nos va a permitir eliminar alguna luz externa incidida en los fototransistores y tener la mayor parte de luz de los IRLED's . Para ello calculamos las variables w11, w12,w13 y siempre restamos de la variable w7 así captamos en los fototransistores la mayor luz infrarroja emitida por los IRLED's.
Después calculamos la variable w6 con los valores de w10 hasta w13 y la dividimos para 4 esto hace tener la lectura de todos los fototransistores.

Calculamos w7 con w6 antes calculado esto hace que ajustemos la distancia si un objeto esta demasiado lejos no vamos a tener tanta incidencia de luz infrarroja emitida por IRLED's hacia los fototransistores, cuando vamos a tener un objeto cerca si vamos a tener mayor incidencia de luz. Ver en la figura 2.18

INICIALIZACION DEL PROGRAMA

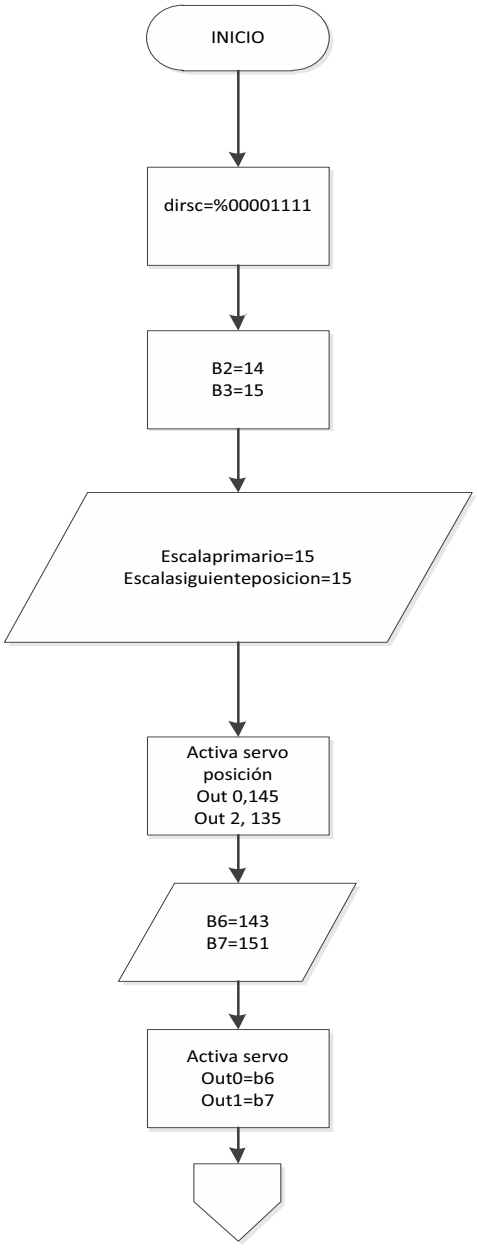


FIGURA 2.16 INICIALIZACION DEL PROGRAMA

RUTINA MAIN

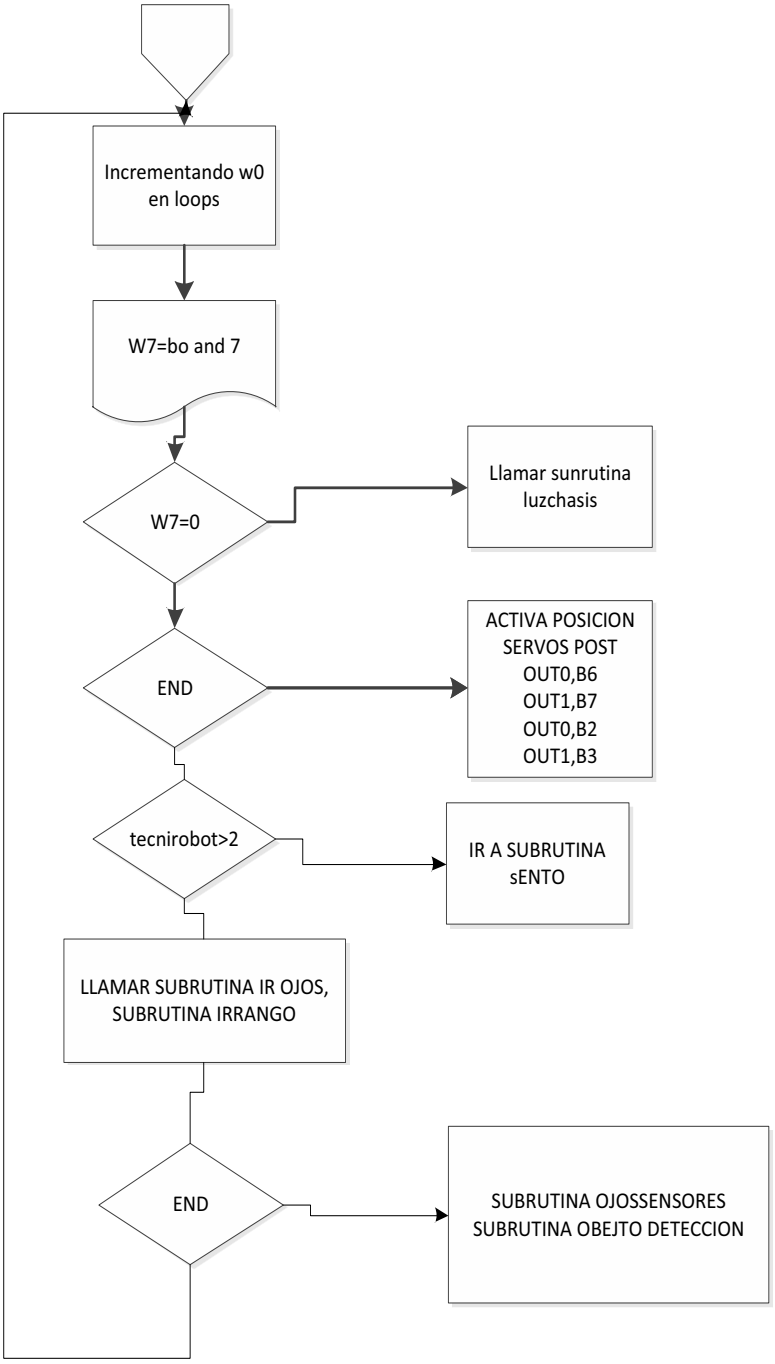


FIGURA 2.17 RUTINA MAIN

SUNRUTINA IROJOS

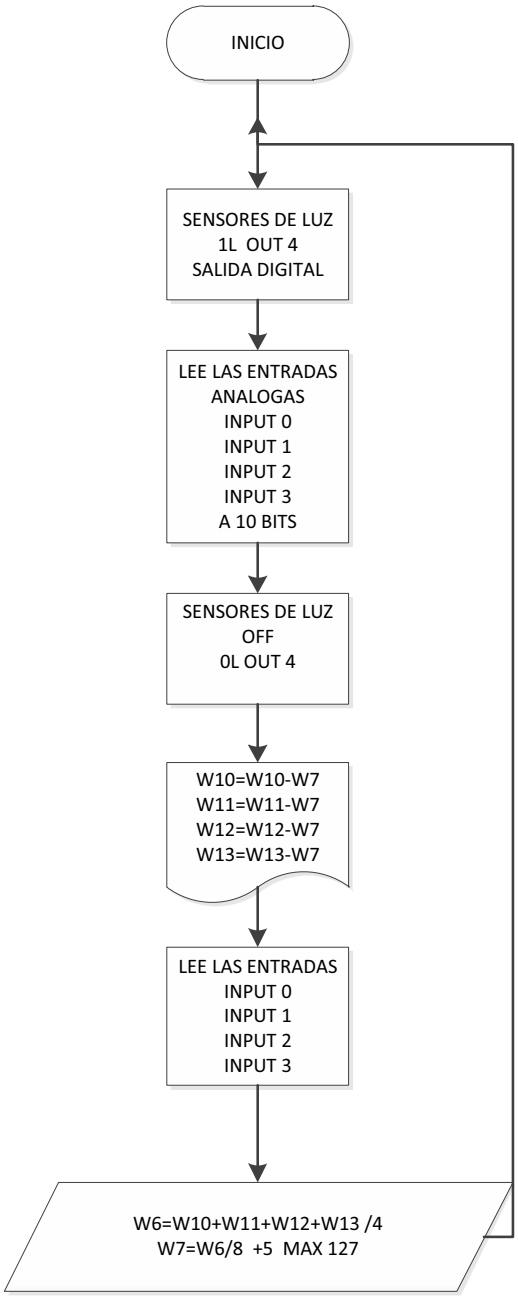


FIGURA 2.18 SUBRUTINA IROJOS

2.4.2.8 Subrutina IRrango

Primero calculamos la variable w6 si es menor que 200 l objeto no está al frente de la cabeza pone el servomotor al centro.
Utilizamos las variables temporales b2, b3 y b6. Para cálculos de posición de los servomotores de la cabeza.
Para este caso corrige los valores si el servomotor de la cabeza está a la izquierda lo mantiene a la izquierda sin que se salga de rango para ello calculamos b2, lo mismo sucede si el motor esta al centro lo mantiene en el centro lo mantendrá para ello calculamos b3, y lo mismo sucede si el motor de la cabeza está a la derecha lo mantiene a la derecha sin procurando que no salga de rango para ello calculamos b3 Si b3 es menor que 135 mueve a la siguiente posición no va a dejar que el servomotor se quede solo a la derecha Si lo valores son iguales esto indica que los servomotores están al límite y lo pone en stop para eso calculamos las variables b6 y b7 y después regresa a la subrutina . Ver en la figura 2.19.

Siguiendo la subrutina calculamos w7 sumamos los valores w10 y w11 lo dividimos para el factor izquierda con esto podemos regular la velocidad con la que se mueve el servomotor de la cabeza a la derecha y a la izquierda, iguala las variables b4 y w7 esto permite que el servomotor no se salga del rango de trabajo.
Calcula w9 para este caso si detecta un objeto a la izquierda los IRLED's va a mover la cabeza hacia la izquierda.
Compara los valores w10y w11 si w10 es menor que w11 calcula w9 esto va hacer si un objeto está a la derecha va a poner el servo motor a la derecha.
Compramos el valor temporal w12 y w13 si el valor temporal w12 es mayor que w13 calculamos w8. Esto hace si detecta un objeto abajo mueve la cabeza hacia abajo pero siempre compara los valores los iguala y resta del valor calculado w8 esto va a permitir que no se vaya el servomotor del rango de trabajo. Podemos apreciar en la figura 2.20.

Para este caso utilizamos la variable w12 y w13 compramos si w12 es menos que w13, calculamos el valor temporal w8 esto hace que el servomotor motor se mueva hacia

arriba si detecta un objeto arriba los IRled's. Compra los valores temporales y resta del valor calculado w8 para que el servomotor no sobrepase el valor del trabajo.
Si w9 es mayor que 3 y w8 es mayor que 3 pone el valor temporal en 0 esto hace que si el objeto se mueve o no detecta algún objeto se paren los servomotores y ponga al centro.
Compara b6 , b7 esto hace que el servomotor de la base empiece desde 0 para nuestro caso seria 0 grados.
Calculamos w7 si este valor es menor que 50 hace que el servomotor de la cabeza se mueva a la derecha después calculamos el valor temporal b6 esto hace que detecte si esta todo a la derecha lo que hace b7 es regresar el servomotor hacia el otro lado lo va a mover entre 10 y 15 grados la velocidad está controlado por el valor temporal b7 .Ver en la figura 2.21

Siguiendo con la subrutina , calculamos w7 es mayor que 50 detecta que el servomotor está a la izquierda calculamos w7 esto detecta que el servo está demasiado a la izquierda para eso calculamos el valor temporal w6 que detecta que el servomotor está demasiado a la izquierda para eso calculamos el valor temporal b6 que hace regrese al otro lado unos grados y se mantenga a la izquierda y b7 va a controlar esa velocidad de retorno pero cuidando que el servomotor no sobrepase el valor de trabajo .
Después calculamos variable temporal w6 si es mayor que 550 calculamos la variable b7 que hace que detecte si un objeto está al frente o no lo compraran esos valores y calcula la distancia ideal del objeto al frente. Esto logramos calculando la variable b6.
Nuevamente calculamos la variable b6 y b7 esto hace que los servomotores de la base no se muevan si esta fuera de rango el objeto al frente de la cabeza y si está en el rango mueve los servomotores hacia atrás siempre controlando la velocidad dada por el valor temporal w7.
Si el objeto está al frente en el rango va a mover los servomotores de la base hacia el frente .Ver en la figura 2.22

SUBRUTINA IRRANGO

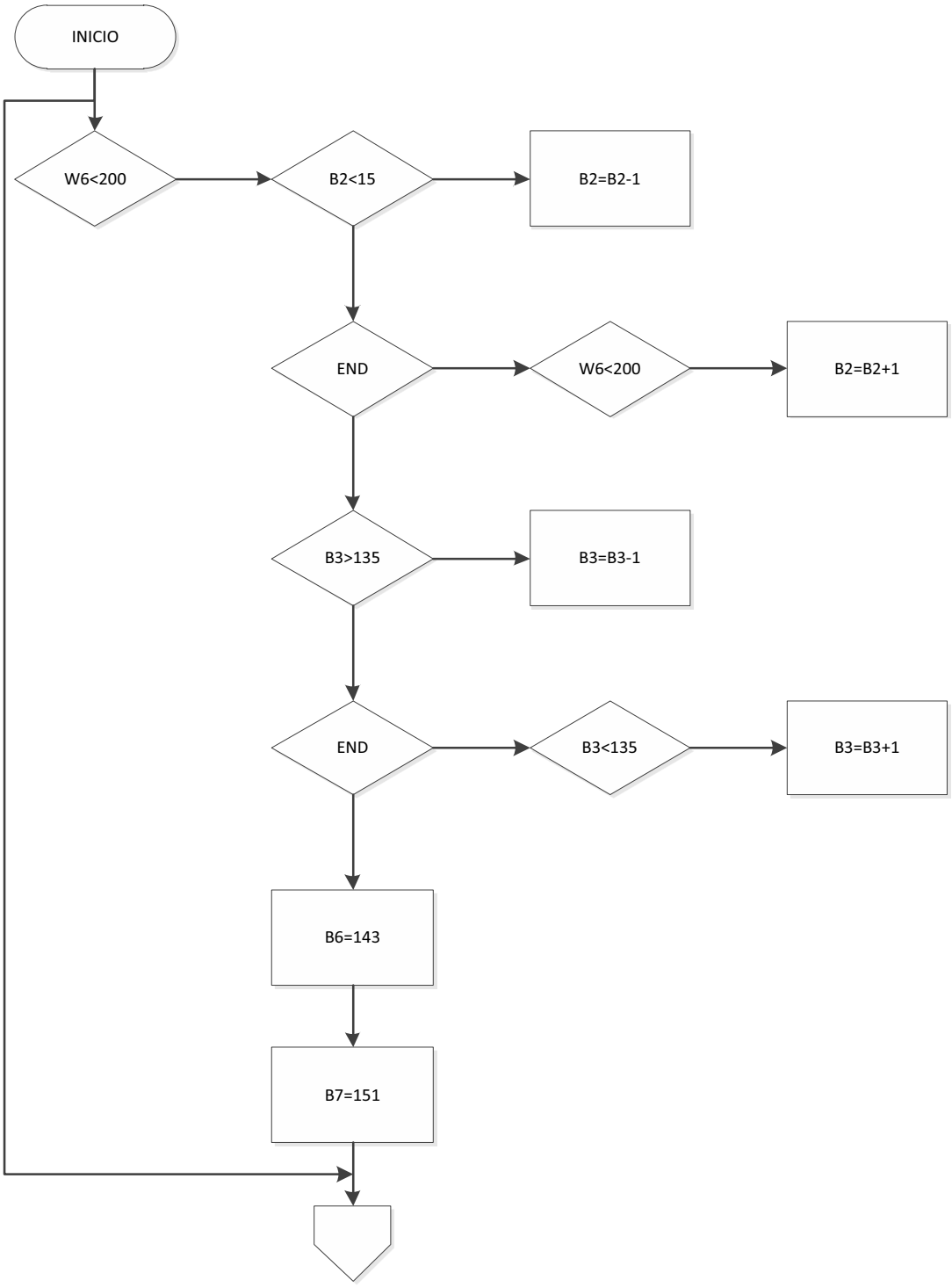


FIGURA 2.19 SUBRUTINA IRRANGO

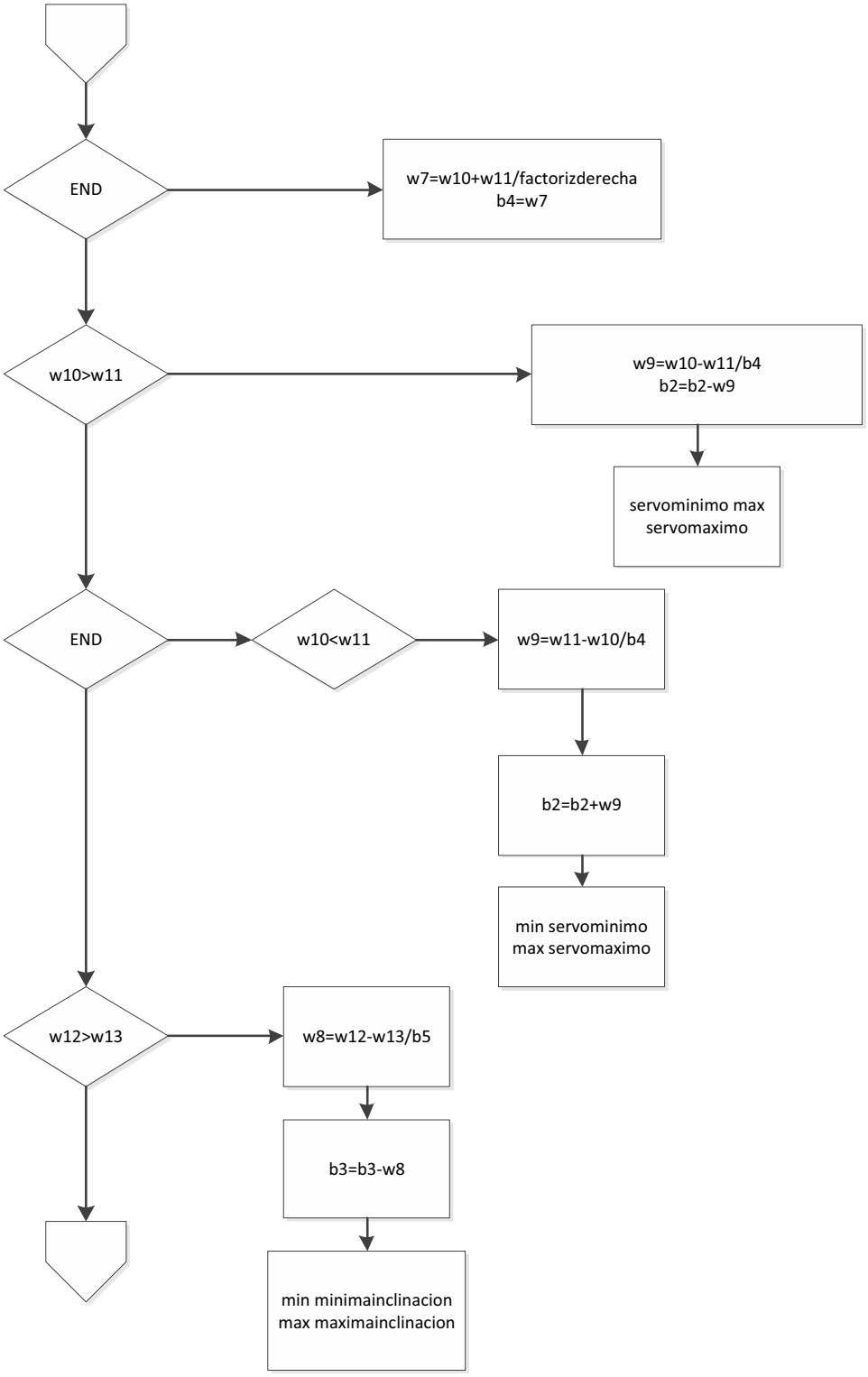


FIGURA 2.20 SUBRUTINA IRRANGO

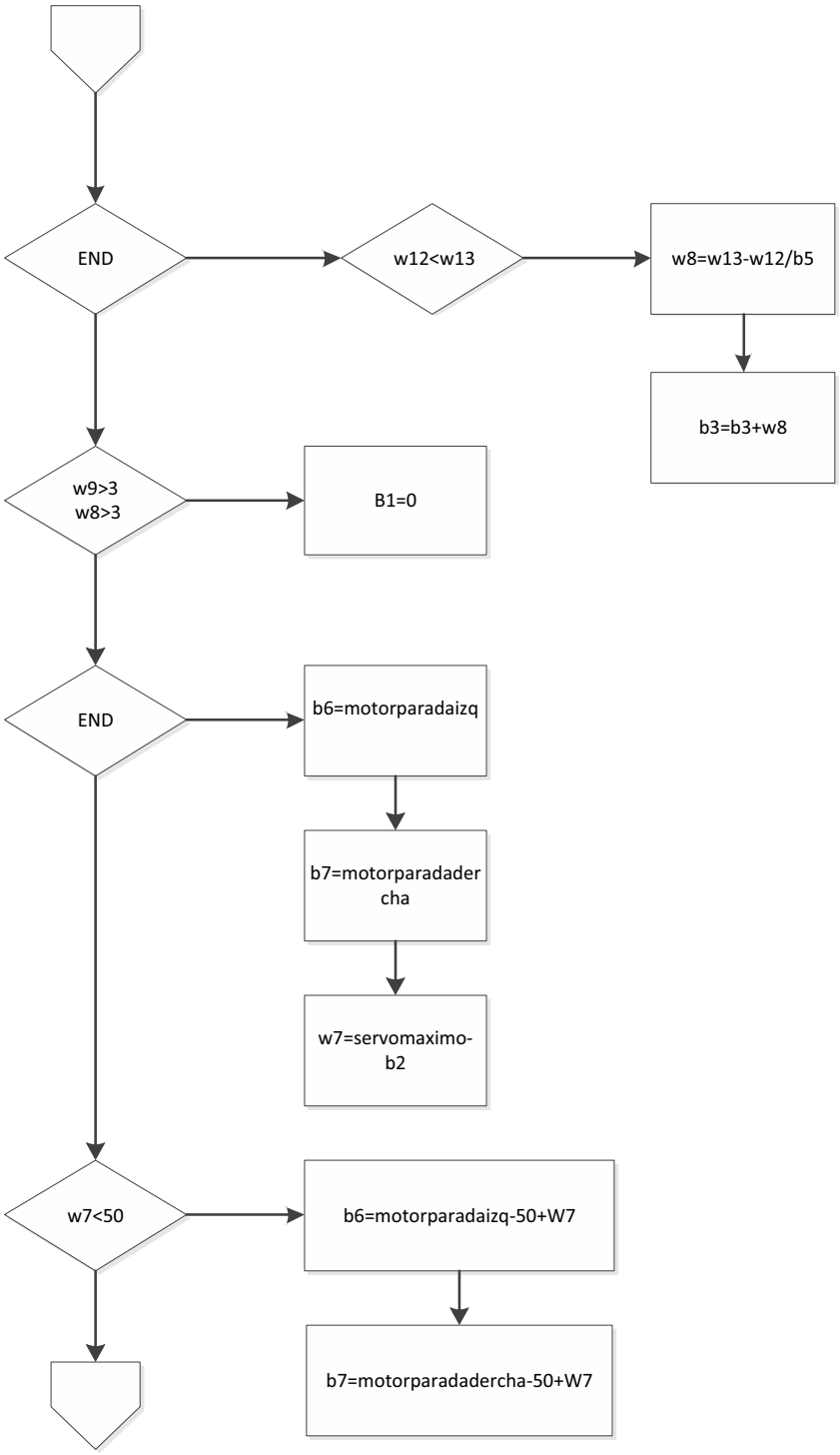


Figura 2.21 SUBRUTINA IRRANGO

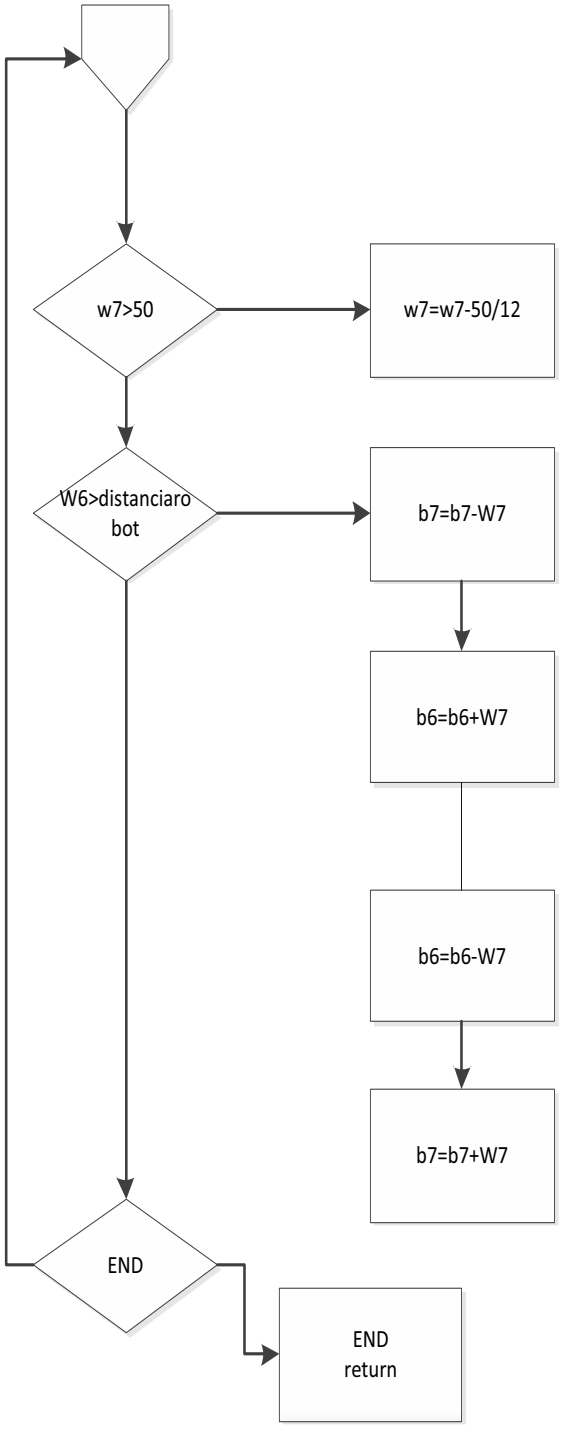


FIGURA 2.22 SUBRUTINA IRRANGO

2.4.2.9. Subrutina Ojossensores.

Encendemos los diodos IRED's de la base. Leemos las entradas digitales del microprocesador vamos a leer fototransistores de la base.
Apagamos los IRED's , Calculamos b10 pero si se aproxima a 240 , o se aproxima al valor de b10 esto quiere decir que un objeto ha sido detectado en la base esto hace encender el LED e ha sido detectado hacemos una operación XOR y calculamos w7 si w7 es igual a b10 este valor es mayor que 4 va a encender el LEd de la esquina que este con el objeto si un objeto es detectado en las 4 esquinas va encender los 4 LEDS.
Va encender los LED's de la base, y si no detecta algún objeto apaga los LEDS.
Después leemos el IRED de la esquina derecha trasera calculando la variable b9 con b10 y una operación AND. Ver en la figura 2.23.
Después de leer IRED de la base subrutina calculamos b9 si es igual a 0 esto hace que el servomotor de la base del lado derecho entre en reversa para esto comparamos el valor b7 si es menor se mueve el servomotor si es igual el servomotor se va a parar esto quiere decir que no hay ningún objeto en la esquina derecha.
Después calculamos b6 esto hace si detecta un objeto en la esquina trasera izquierda lo va poner en stop los servomotores de la base esto indica que está en las dos esquinas traseras un objeto. Y si no detecta ningún objeto termina la subrutina
Calculamos b9 esto hace leer el IRED trasero izquierdo de la base, hacemos una operación AND 32 con b10. Si detecta un objeto en la parte trasera izquierda de la base se mueve el servomotor en reversa de la base izquierda para ello utilizamos la variable b6 pero si no detecta ningún objeto en la base lo pone en stop.
Calculamos b7 si detecta un objeto en la base trasera derecha pone en reversa el servomotor de la base derecha, y si no detecta un objeto pone en stop el servomotor.
Ver en la Figura 2,24.
Siguiendo con la subrutina calculamos la variable b9 hace una operación AND 64 con b10, esto lee el IRED frontal izquierdo de la base, calculamos b9 si es igual a 0 . eso hace que el servomotor de la izquierda se ponga en reversa, después calculamos b6 esto hace si no detecta algún objeto para el servomotor calculamos b6 esto lee el led frontal derecho si detecta algún objeto pone en reversa el servomotor si detecta un

objeto en ambos va poner los dos servomotores de la base en reserva y si no detecta algún objeto en stop.

Si b7 es mayor que 7 esto sobrepasaría el valor entonces resetearía el contador poniendo w0 en 0 esto indica que el objeto está moviéndose. Y terminaría la subrutina. Después calculamos b9 y hacemos la operación AND con b10 esto hace leer el IRED frontal derecho de la base si el objeto es detectado por el fototransistor pone en reversa el servomotor de la derecha, calculamos b7 y si no es detectado ningún objeto pone el servomotor en parada. Después si el objeto es detectado en la parte frontal derecha e izquierda pone los dos servomotores en reversa y también compara y pone w0 igual a cero para resetear el contador. Esto se puede apreciar en la Figura 2.25

SUNRUTINA Ojossensores

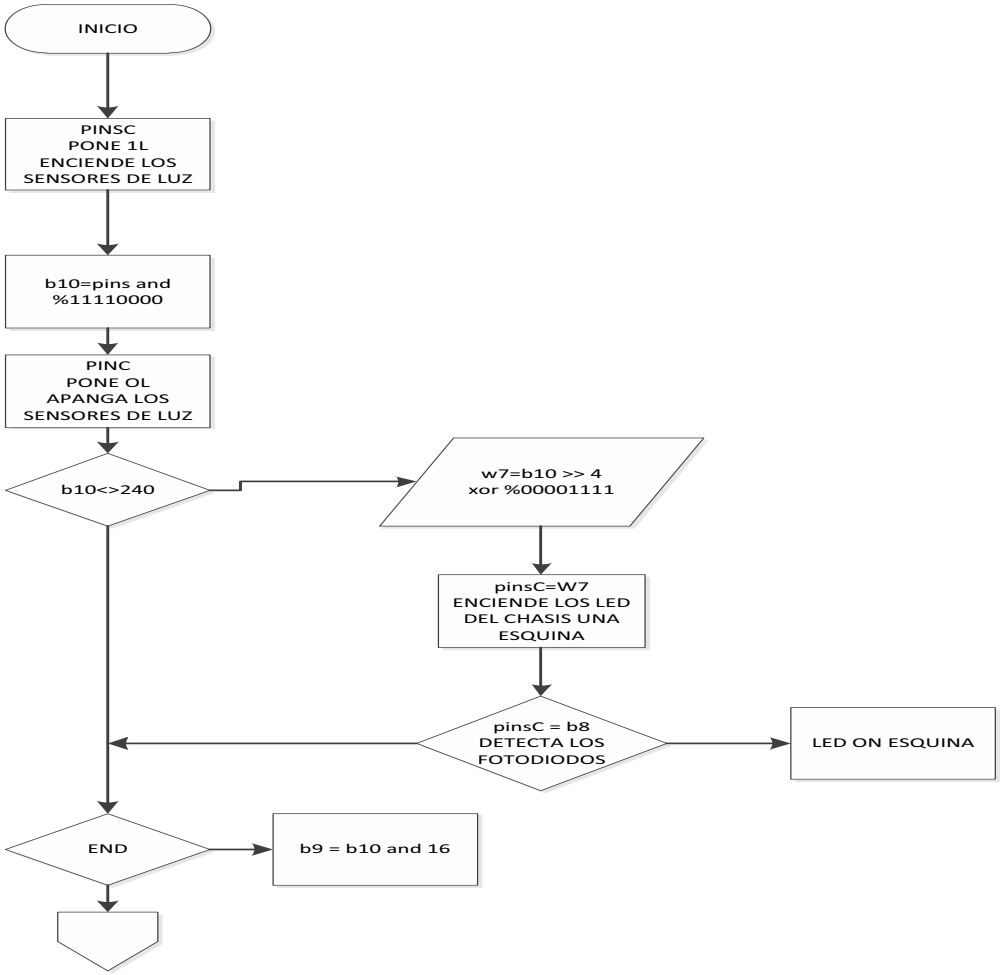


FIGURA 2.23 SUBRUTINA OJOSSENSORES

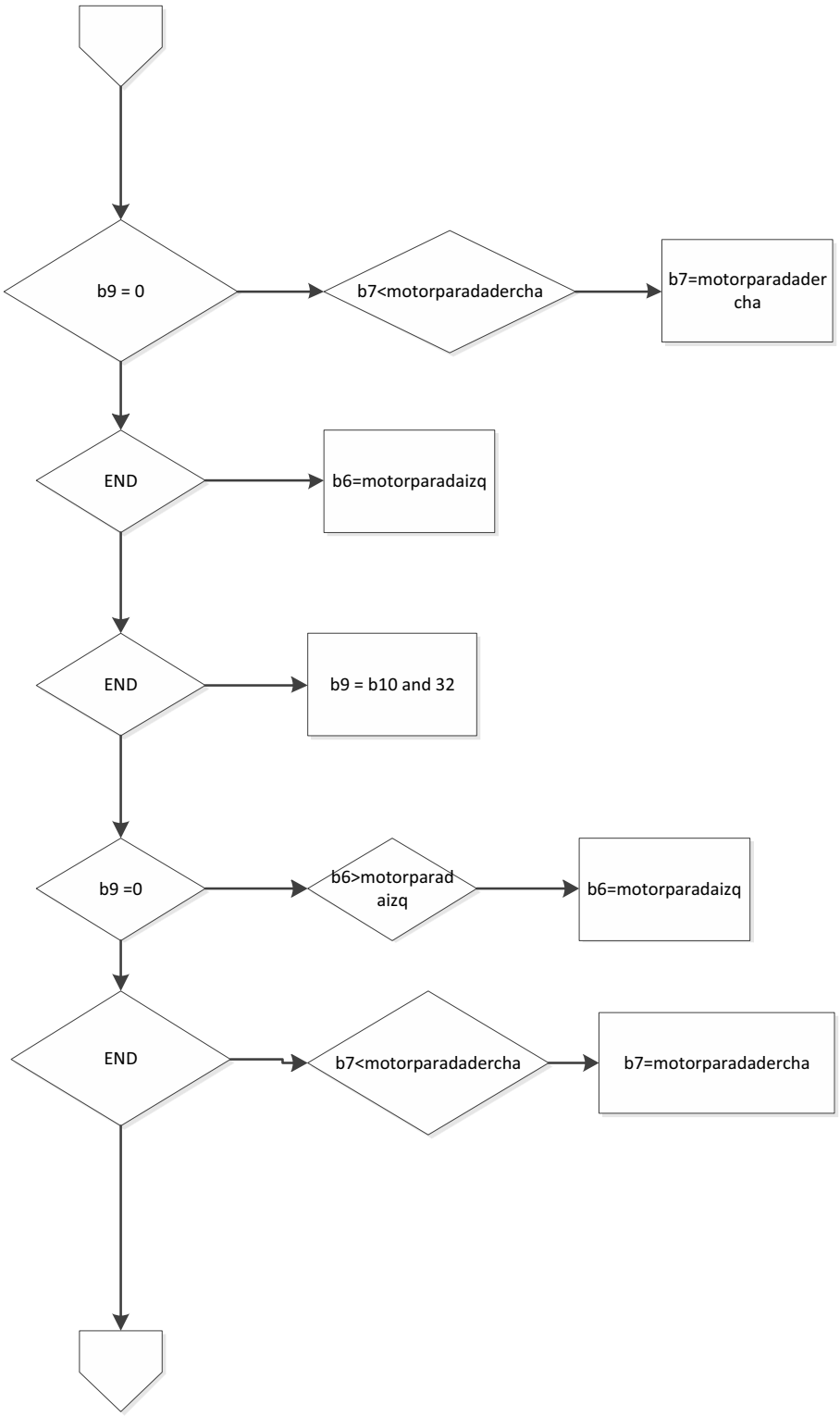


FIGURA 2.24 SUBRUTINA OJOSSENSORES

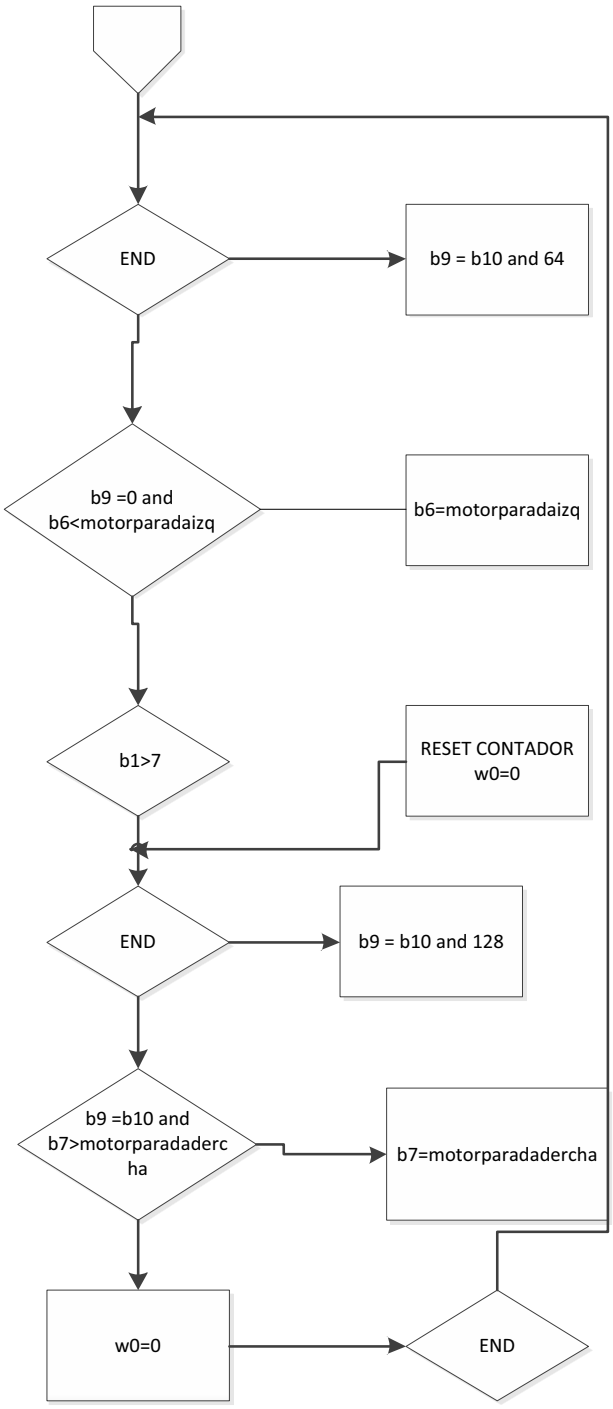


FIGURA 2.25 SUBRUTINA OJOSSENSORES

2.4.2.10 Subrutina Objetodeteccion

Encendemos los IRED's de la base con el comando pinsc.
Comparamos la variable b10 es igual a pins hacemos una operación AND 11110000 esto lee la entrada digital. Luego apagamos los IRED's.
Calculamos la variable w7 si es igual a b10 y esta es mayor que 4 esto sirve para detectar si un objeto está cerca graba los 4 bits .Este va estar leyendo las 4 esquinas de los fototransistores verificando si un objeto está cerca o no.
Calculamos b10 pero si este valor es igual a cero quiere decir que un objeto no es detectado o no hay objeto frente.
Comparamos pinC y comparamos con b8 si son iguales esto hace que regrese al subrutina nuevamente..
Si pinsC es igual es igual a w7 esto quiere decir que el objeto ha sido detectado y enciende el indicador LED.
Calculamos la variable b9 y hacemos una operación AND 16, lee el fototransistor trasero derecho de la base. Si b9 es igual a 16 esto quiere decir que el objeto ha sido detectado. Calculamos b7 si es menor que motorparada izquierda (143) pone el servomotor derecho en reversa calculamos la variable b7 y si es igual a motorparadaderecha (153), esto hace poner el motor en stop.Ver en la Figura 2.26
Calculamos la variable b9 y hacemos una operación AND 32 con b10, lee el fototransistor trasero izquierdo de la base. Si b9 es igual a 32 esto quiere decir que el objeto ha sido detectado. Calculamos b6 si es mayor que motorparada izquierda (143) pone el servomotor de la base izquierdo en reversa, calculamos la variable b6 y si es igual a motorparadaderecha (153), esto hace poner el motor en stop.
Calculamos b7 si es menor que motorparadaderecha esto quiere decir que el motor derecho está en reversa calculamos b7 si es igual a motorparadaderecha le pone en stop el servomotor derecho.

Calculamos la variable b9 y hacemos una operación AND 64 con b10, lee el fototransistor delantero izquierdo de la base. Si b6 es igual a 64 esto quiere decir que el objeto ha sido detectado, adelante y el servomotor izquierdo se pone en reversa

Calculamos b1 si es mayor 7 resetea el contador ya que el robot está en movimiento ,esto significa que el robot también ha detectado en la parte trasera un objeto y otro en la parte delantera por lo que calculamos w0 si es igual a 0 entonces pone los servomotores tanto derecho e izquierdo en parada. Ver en la figura 2.27

Calculamos la variable b9 y hacemos una operación AND 128 con b10, lee el fototransistor delantero derecho de la base. Si b9 es igual a 128 hacemos una operación and con b7 es mayor que motorparadaderecha esto quiere decir que el objeto ha sido detectado pero está en movimiento el robot y va poner en stop el servomotor esto quiere decir que ha detectado un objeto atrás reseteando el contador si cumple esto termina la subrutina si no empieza a leer la subrutina desde el principio. la parte delantera por lo que calculamos w0 si es igual a 0 entonces pone los servomotores tanto derecho e izquierdo en parada. Ver en la figura 2.28

2.4.2.11 Subrutina Luz chasis

Desplaza a la derecha un bit, sirve para encender los IRLED's de la base de manera secuencial de acuerdo a la subrutina. Para eso utilizamos los valores temporales de b8 esta subrutina va a encender los IRLED's de manera secuencial va a desplazar un bit y si b8 compara los valores si no termina la subrutina puede apreciar 2.29.

2.4.2.12 Subrutina Senti

Esta subrutina utilizamos el comando case b1 esto permite detectar si la cabeza detecta un objeto adelante.
Utilizamos el comando case 3.4, esto hace volver lentamente el servomotor de la base utilizamos .Calculamos b7 esto hace que ajuste la velocidad del servomotor de la base cuando este en reversa, Calculamos b6 esto hace que ajuste la velocidad del servomotor de la base cuando este en reversa.
Utilizamos case 5 calculamos b7 si es igual a motorparadaderecha pone en stop el servomotor derecho, calculamos b6 si es igual motorparadaizquierdo pone el servomotor de la base en stop. Calculamos la variable b2 iguala y compara si es igual

enciende la cabeza y la mueve a la izquierda Leemos la subrutina IRojos. Calculamos b3 y compara si es igual la cabeza está a del centro a la izquierda.
Enviamos a la Subrutina IRojos, grabamos el valor en la RAM de la subrutina IRojos para eso utilizamos poke 80. Ver en la Figura 2.30 y Figura 2.31

Utilizamos el comando Case 6, regula la velocidad del servomotor de la cabeza
Calculamos b2 esto mueve la cabeza a la derecha, calculamos b3 esto mueve la cabeza y esta del centro a la derecha.
Mandamos a leer la subrutina IRojos. Utilizamos ese valor y lo grabamos en la memoria Ram para este caso utilizamos poke 82 . Ver figura 2.31.
Utilizamos Case 7 encuentra un dirección en la memoria y la utiliza.
Calculamos b2 esto pone la cabeza al centro del robot.
Leemos la memoria RAM con el comando peek 80 y 82 lo datos compramos los datos w6 y w7 de la memoria esto calcula y gira la cabeza a la izquierda si no hay más espacio entonces calculamos b6 esto hace que el motor se ponga en reversa lentamente para ello sumamos 5 , y si el motor está a la derecha y no hay más espacio lo pone en reversa lentamente para esto calculamos b6 pero restamos 5.
Después calculamos b6 sumamos nuevamente 5 esto hace que el motor hace mover hacia la derecha lentamente nuevamente.
Calculamos la b6 hace que la cabeza se mueva hacia arriba después calculamos b7 esto hace que la cabeza se mueva hacia abajo. Si detecta que la cabeza está arriba termina la subrutina.
Utilizamos case 8 to 12, esto hace que se mueva los servomotores de la base lentamente hacia adelante.
Calculamos b3 para eso restamos 15 esto hace que la cabeza se mueva hacia abajo lentamente.
Si b1 es mayor que b10 lee los fototransistores de la cabeza, y manda a leer después llamamos a la subrutina IRojos. Si no termina si no cumple. Ver Figura 2.32
Calculamos w6 si es mayor igual a distanciarobot, esto hace si un objeto está demasiado cerca de la cabeza calculamos w0 igualamos a 0.esto hace que los servos de la cabeza se ponga en stop

Después calculamos b6 y b7 esto hace que los servomotores de la base se pongan en stop.

Calculamos w7 teniendo en cuenta que mínimo 0 y máximo 20 esto va a ser regular la velocidad de los servomotores de la base de acuerdo si un objeto está cerca o lejos.

Calculamos b6 si el objeto está lejos mueve los servomotores de la base hacia delante.

Calculamos b7 si el objeto está cerca mueve el servomotor hacia atrás. Si w0 es igual a cero pone en stop los servomotores de la base

Después calculamos case 12 si es mas a 12 pone en stop los servos y después retorna a la subrutina. Ver en la Figura 2.33

.SUBRUTINA ObjectoDeteccion

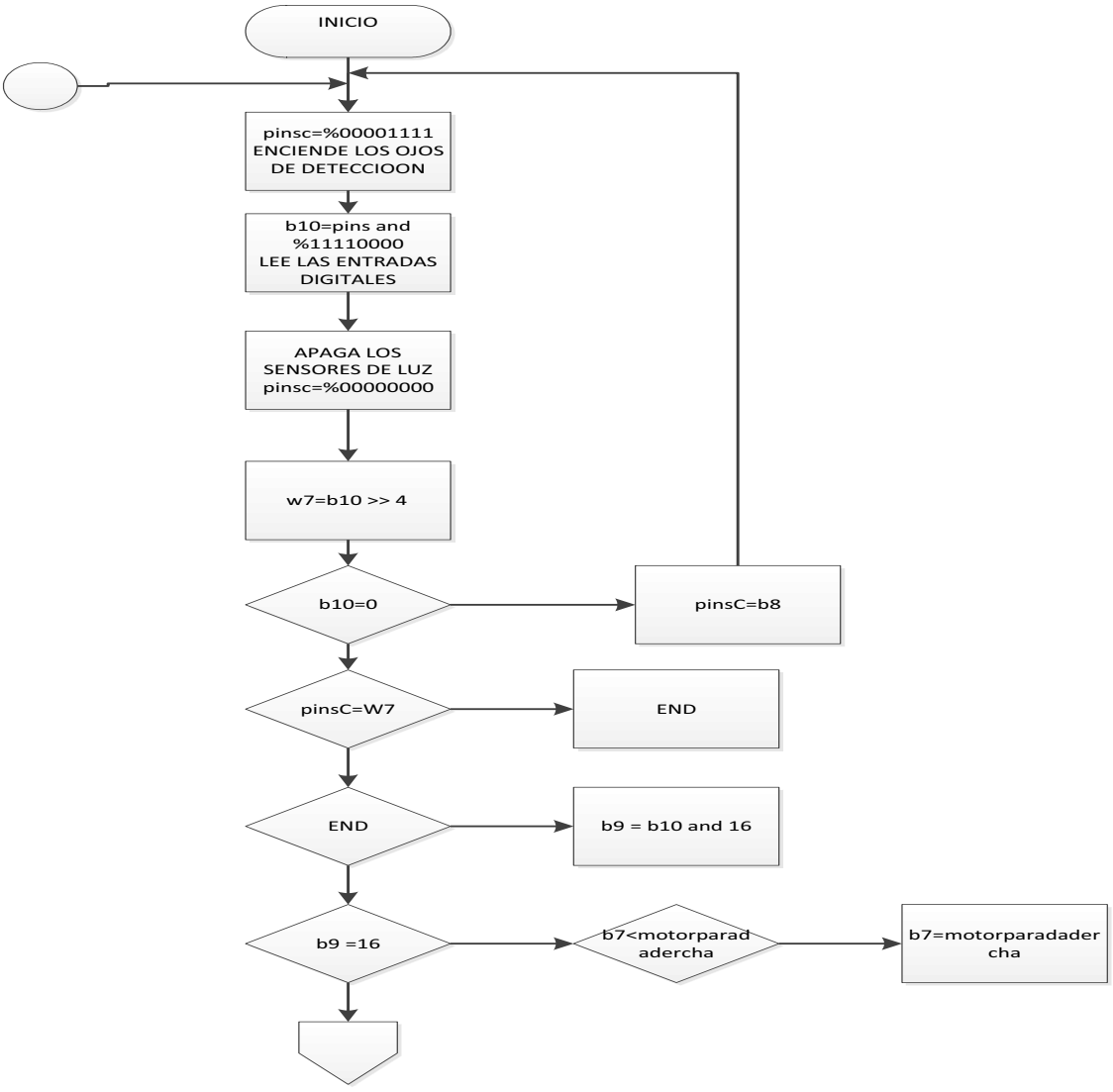


FIGURA 2.26 SUBRUTINA OBJETODETECCION

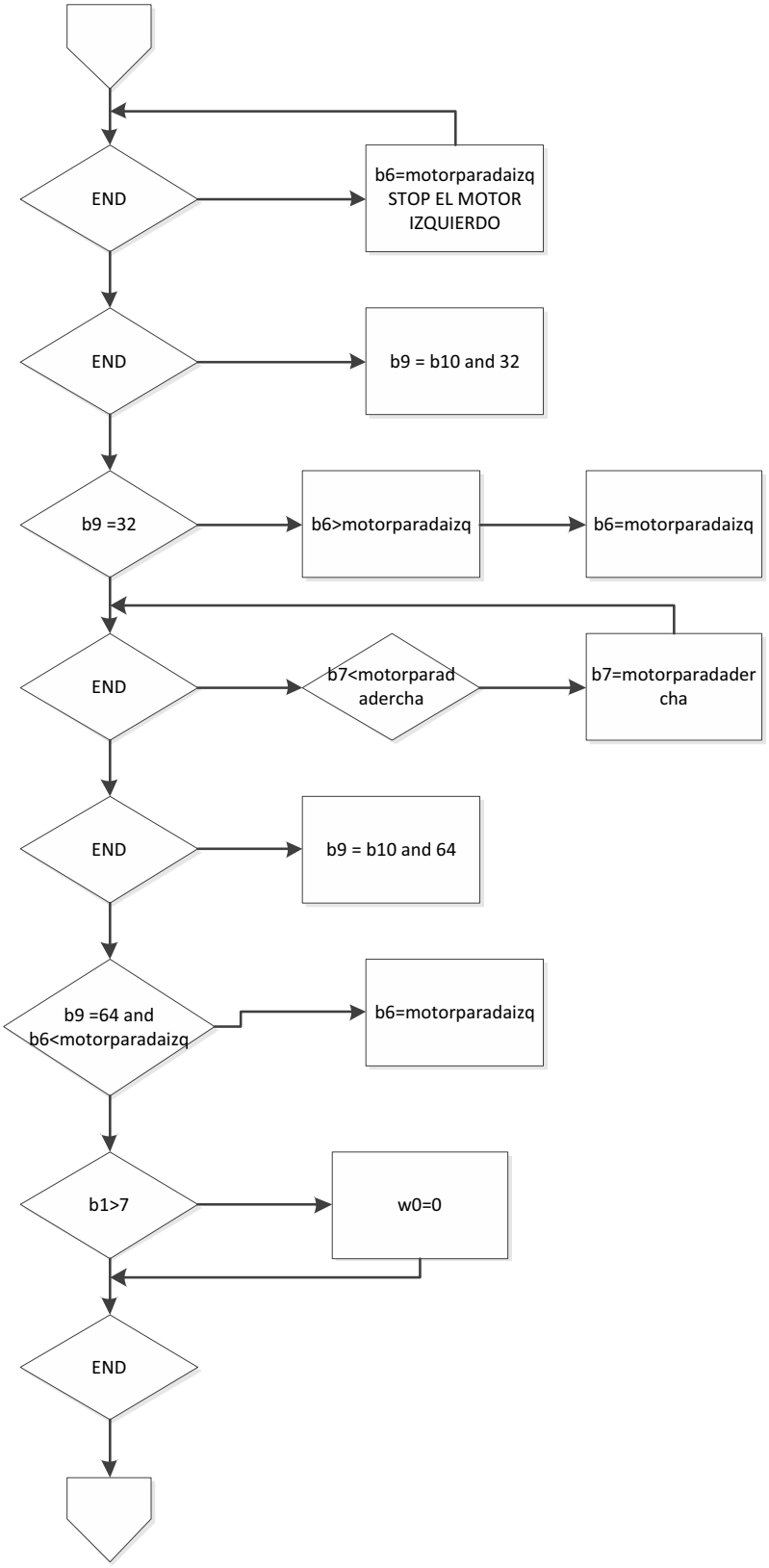


FIGURA 2.27 SUBRUTINA OBJETODETECCION

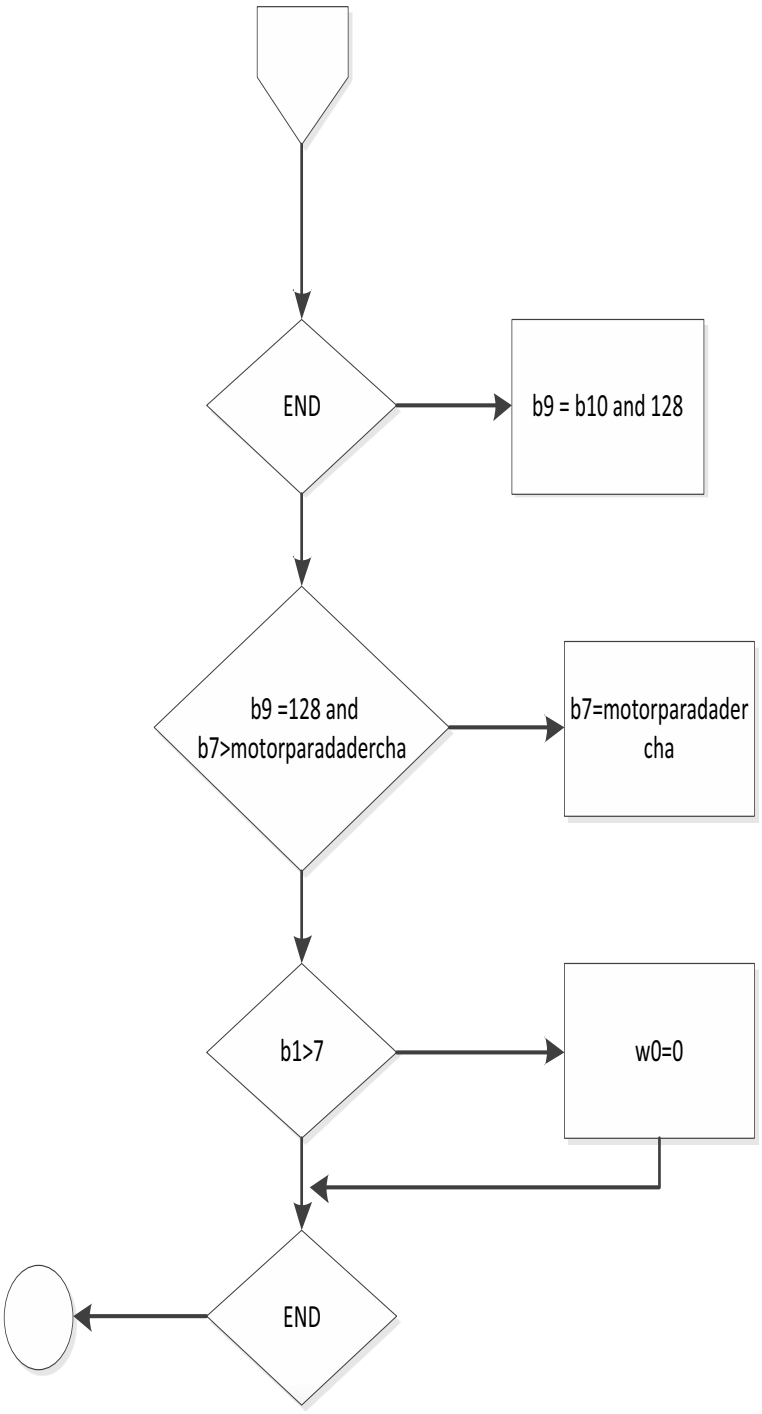


FIGURA 2.28 SUBRUTINA OBJETODETECCION

SUBRUTINA LUZ CHASIS

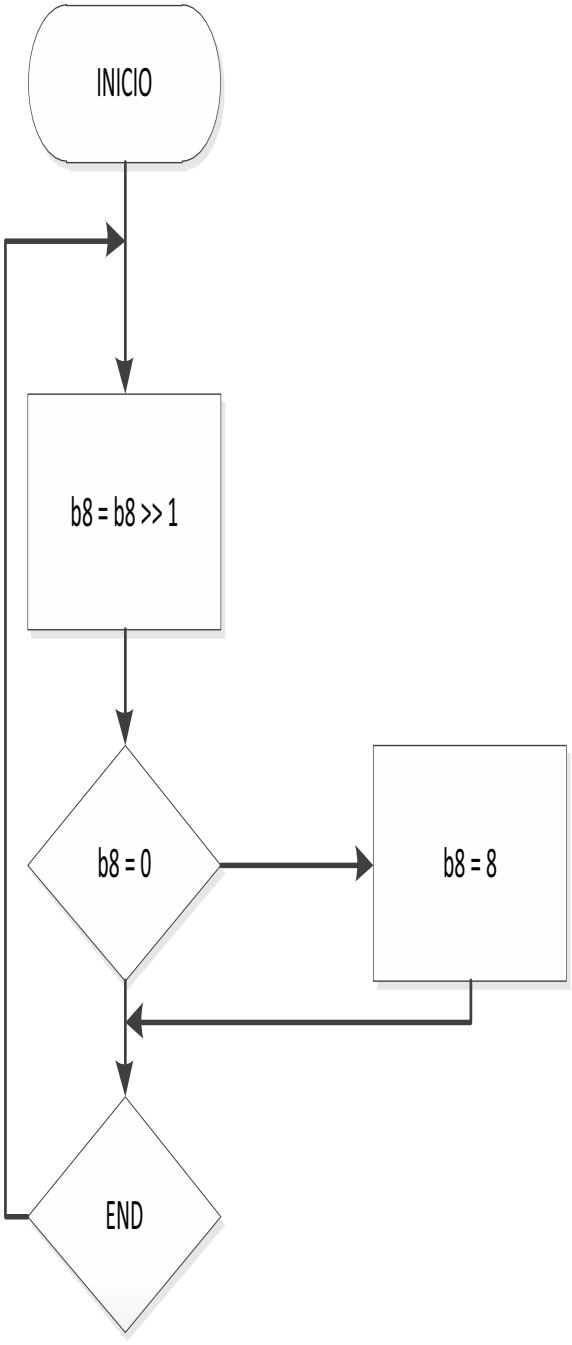


FIGURA 2.29 SUBRUTINA LUZ CHASIS

SUBRUTINA SENTI

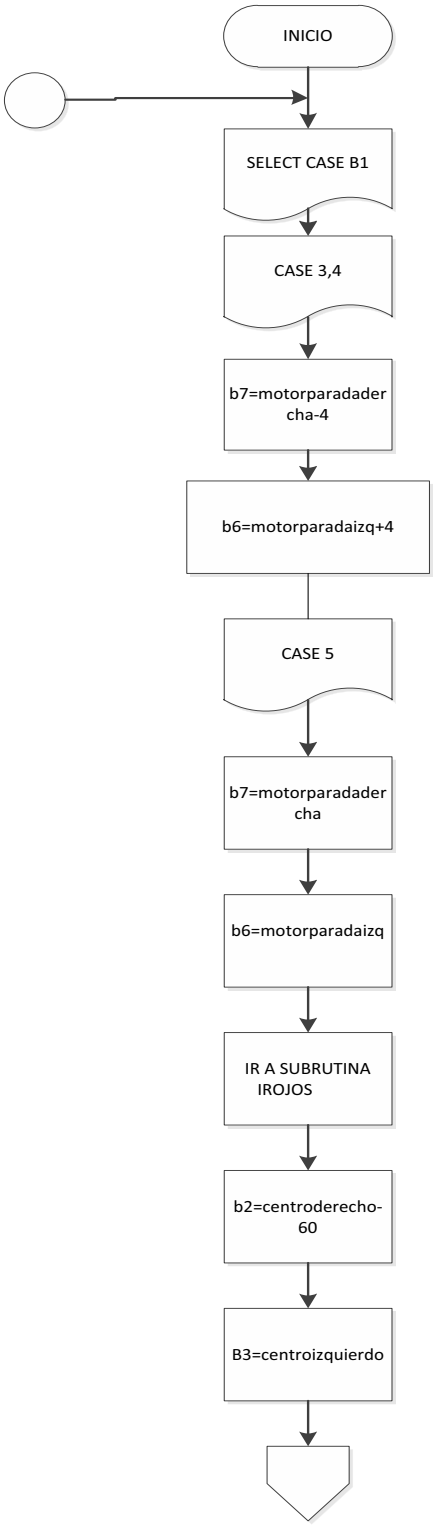


FIGURA 2.30 SUBRUTINA SENTI

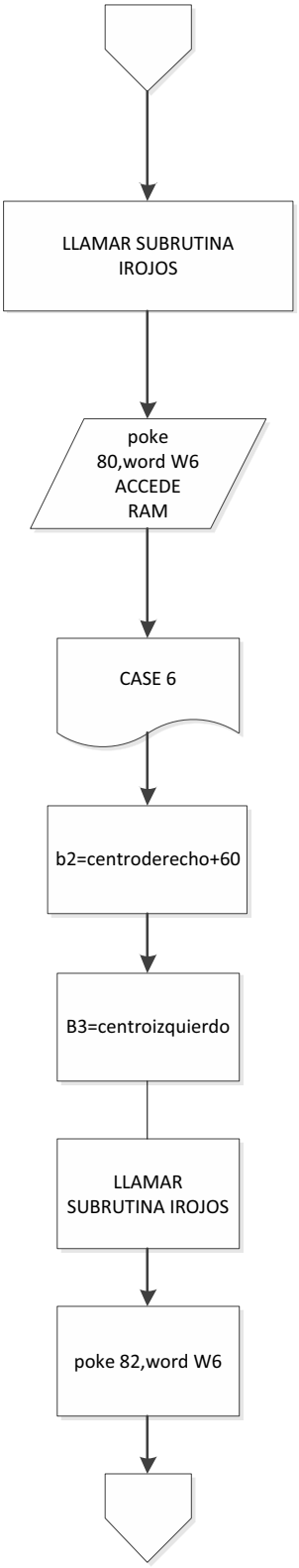


FIGURA 2.31 SUBRUTINA SENTI

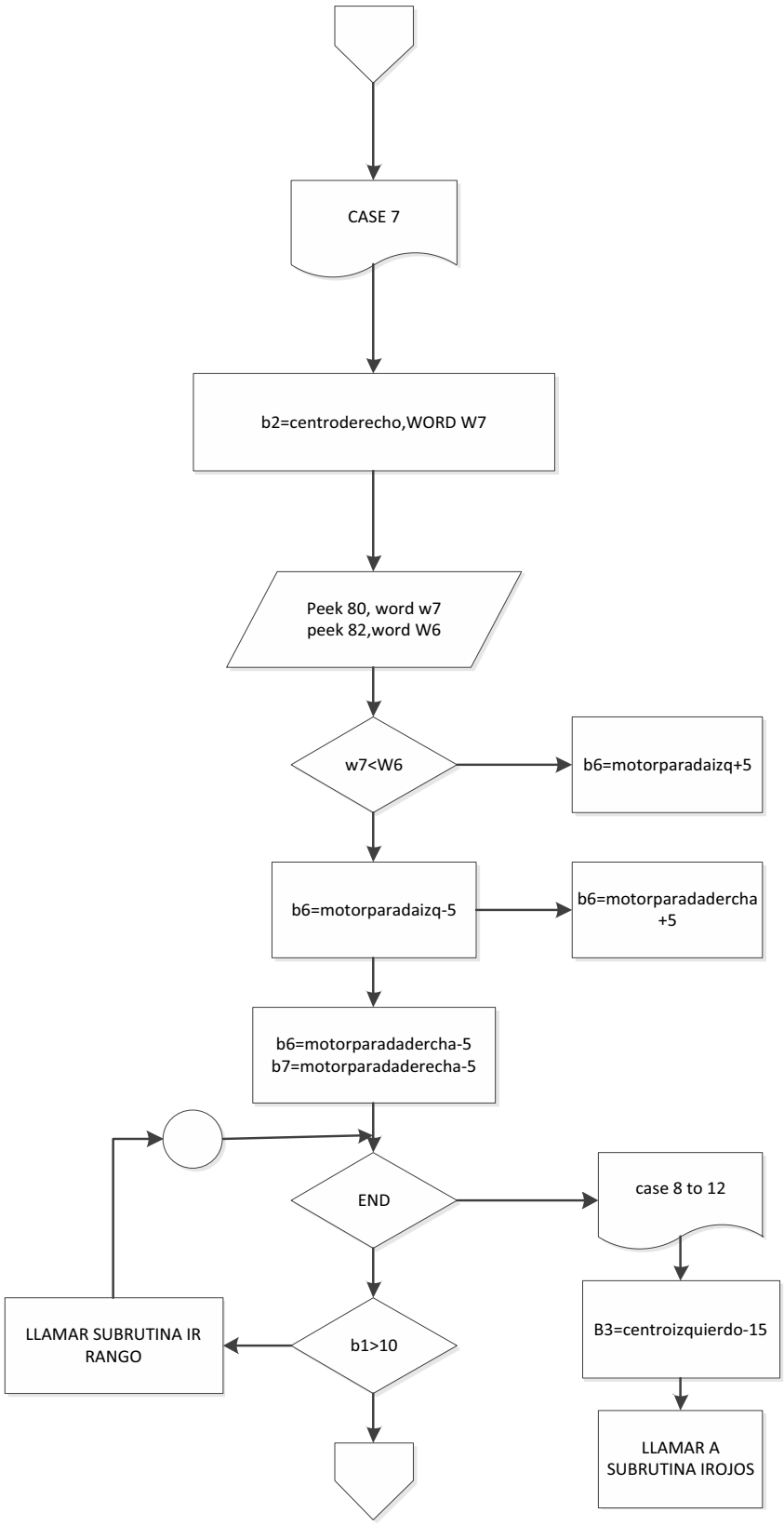


FIGURA 2.32 SUBRUTINA SENTI

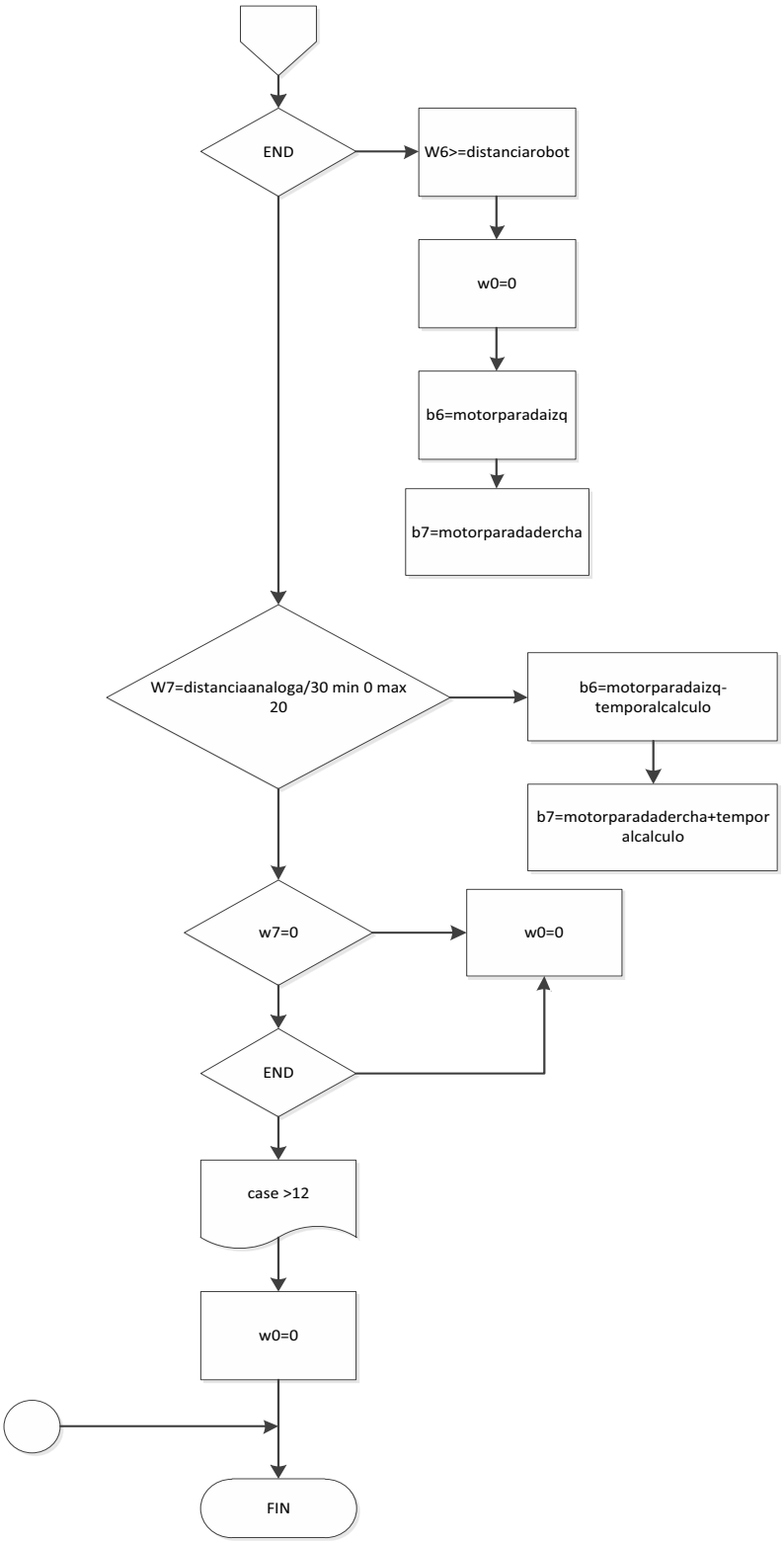


FIGURA 2.33 Subrutina TECNIROBOT

2.5 MODOS DE OPERACIÓN Y PROGRAMACIÓN

El micro controla todo el trabajo del robot, el robot tienen movimientos circulares, atrás, adelante, y la cabeza tiene movimientos arriba abajo, izquierda, derecha debemos tomar en cuenta que la cabeza no debe ser manipulada o topada cuando se esté moviendo ya que los piñones del servo motor puede averiarse o romperse, El microcontrolador controla los movimientos tanto de la cabeza como se indica en la figura 2.34 y los servo motores que mueven adelante y atrás. Ver en la figura 2.35

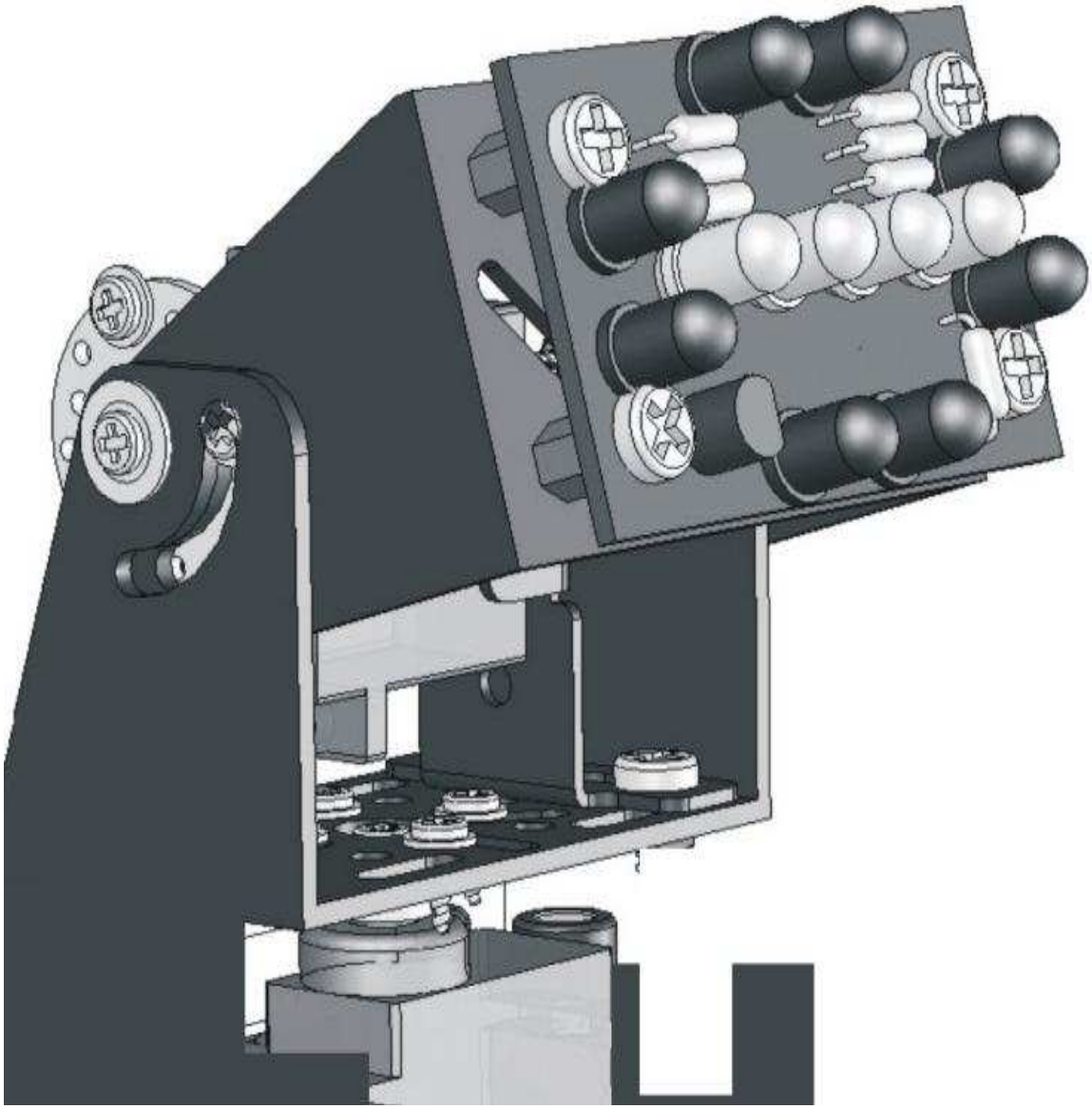


FIGURA 2.34 PARTE FRONTAL ROBOT

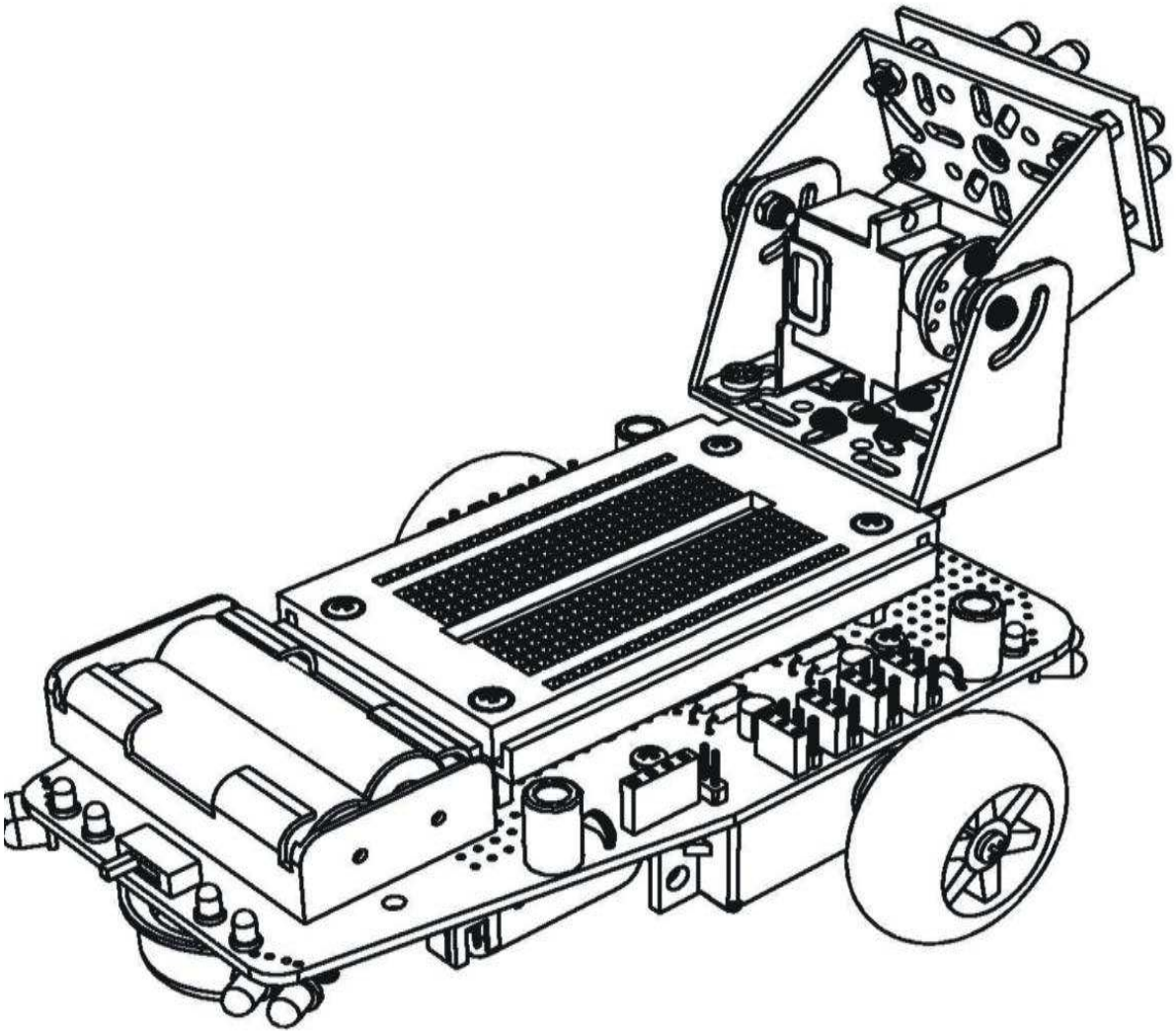


FIGURA 2.35 PARTES DEL ROBOT EMSAMBLADO

2.6 DESCRIPCIÓN DE SU FUNCIONAMIENTO

Una vez puesto las baterías y plenamente cargadas, se enciende el robot con el interruptor de inicio, procurar no topar la cabeza del robot ni sus llantas ya que puede ser forzados los piñones del servomotor, el robot puede detectar objetos hasta 5cm y puede hacer movimientos circular, adelante, atrás y la cabeza arriba abajo, el robot también posee un botón de reset es muy útil ya que permite reiniciar el programa del microprocesador . El robot puede detectar objetos adelante atrás o en cada esquina independiente ,el robot se ha implementado la parte mecánica y la parte electrónica

para su funcionamiento óptimo se requiere no utilizar con incidencia del sol directo ya que puede variar en las lecturas haciendo que salga del rango y producir error en su funcionamiento , se ha diseñado el circuito que cuando los servos salgan de límite de funcionamiento del robot entre en reposo para que vuelva a funcionar se necesita de reiniciarlo con el botón de reinicio que va directo al microprocesador .

2.7 MANUAL DE USUARIO

A continuación se describe en resumen el manual de operación del robot.

1. Colocar las baterías AAA
2. Prender el interruptor como muestra en la figura 2.36
3. Poner en una superficie plana
4. Colocar de preferencia no en lugar donde haya mucha incidencia de la luz solar
5. Presiona reset.
6. Cercar la mano en la parte frontal en la cabeza del robot este se va mover de acuerdo a los movimientos que uno haya con la mano.

NOTA: Una vez arrancado el robot puede presionar el botón de reset o apagar el robot si no se incide ningún objeto el robot se mueve de manera mínima, no va a moverse y se podrá apagarlo.

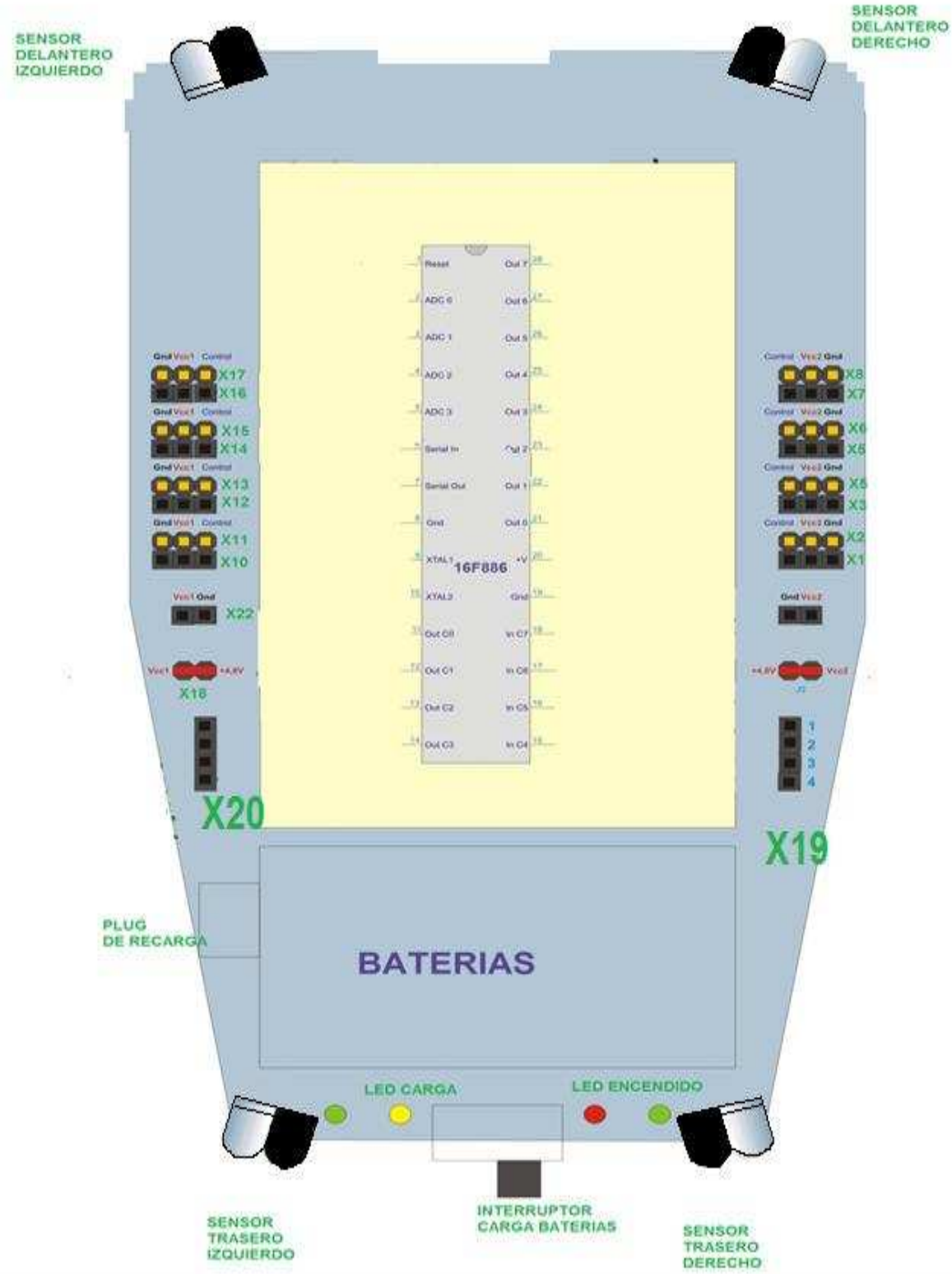


Figura 2.36 BASE DEL ROBOT

CAPITULO III: ANÁLISIS DE RESULTADOS

3.1 COMPORTAMIENTO A FASE EXPERIMENTAL

3.1.1Prueba del sistema.

Una vez ensamblado el robot se pudo conectar la fuente de poder extrema la cual cargaba correctamente las baterías al momento de encender el robot se hizo las pruebas de campo, hemos incidido en dos aspectos con luz directa de sol o luz incandescente directa en superficie plana y en una superficie rugoso pudiendo obtener los resultados a continuación.

3.1.2 Prueba del robot con luz directa de luz solar y luz incandescente en superficie plana.

Al momento de hacer la prueba con luz incidente de sol el robot no reacciona de manera correcta ya que la luz hace que los valores varíen de manera inadecuada, haciendo que el robot no funcione de la manera correcta incluso se clava la cabeza en una sola posición haciendo que funcione fuera del límite , teniendo que presionar el botón de reseteo para que vuelva a funcionar por lo que es recomendable no utilizar con luz directa del sol.

3.1.3 Prueba del robot sin luz solar directa y luz incandescente en superficie plana.

El robot pudo trabajar de manera correcta ya que se encuentra en los niveles establecidos pudiendo mover la cabeza de manera adecuada y moverse atrás adelante sin ningún problema es la manera óptima que puede trabajar el robot y detectar los obstáculos y moverse de manera adecuada.

3.2 CONCLUSIONES

1. El microcontrolador es el cerebro de todo robot el cual hace todo el trabajo y es muy preciso, recepta datos externos procesarlos de manera rápida y tener respuesta inmediata haciendo todo el trabajo de acuerdo como este programado, realizando movimientos autónomos.

2. Es necesario utilizar fototransistores y los IRLED para el proyecto los cuales son fáciles de conseguir y fácil su entendimiento para poderlos hacerles funcionar como sensores de luz y haciendo que el robot haga movimientos apropiados.

3. El robot es controlado con un microcontrolador el cual activa a los servomotores, estos deben ser activados de manera independiente por eso se ha utilizado las 4 salidas del PIC.

4. Para el correcto funcionamiento del PIC16F886 es necesario configurar los puertos C ya que estos funciona tantos como entradas y como salidas, en el programa le hemos configurado para que trabajen en ambos input y ouput.

5. Para poder utilizar el microcontrolador de manera adecuada hay que configurar el conversor análogo digital ya que este nos permite la lectura de los sensores de luz y permite tener lecturas óptimas.

6. En el robot hemos implementado los sensores de luz más adecuados como son los diodos IRLED DIODOS LEDS INFRAQRROJOS y los fototransistores permitiendo el adecuado funcionamiento.

3.3 RECOMENDACIONES

1. Las baterías que están implementadas son suficiente para abastecer prototipo durante 5 a 10 minutos, si se desearía aumentar algunos otros servomotores o otros dispositivos se debería pensar en aumentar ya sea la capacidad en amperaje de las baterías o poner otro par de baterías en paralelo para aumentar la corriente del circuito, ya que si las baterías están bajas el circuito no funciona adecuadamente.

2. El robot también dispone de una salida serial de comunicación que puede ser utilizada para comunicación con otro micro controlador o para programación con el programador, puede implementarse un circuito de comunicación con un diodo Led IR y se puede trasmitir datos entre dos micro controladores, siendo esto muy interesante para futuros proyectos.

3. En este robot se puede implementar circuitos como cámaras o implementar otros detectores de objetos de mayor sensibilidad para futuros proyectos.

4. El robot tiene gran acogida como un prototipo de investigación pudiendo ser mejorado ya que el programa puede ser cambiado y ser reprogramado según las necesidades.

DETALLE DE COSTOS PARA LA IMPLEMENTACIÓN DEL SISTEMA

A continuación se detalla el costo

Cantidad	Código	Valor	Precio unitario	Precio total
1	PIC16F886 is/p		$ 7,50	$ 7,50
2	Servomotor estándar		$ 20	$ 40
2	Servomotor		$ 28	$ 56
12	Fototransistores		$ 0.45	$ 8.10
8	Opto acoplador PC817		$ 0.25	$ 2
25	Resistencias		$ 0.04	$ 1
2	Portabaterias AAA		$ .50	$ 1
1	Protoboard mini		$ 15	$ 15
2	Diodo rectificador		$ 0.30	$ 0.30
2	Interruptor		$ 0.65	$ 0,65
2	Llantas para servo		$ 2	$ 4
4	Capacitor	470uF / 16V	$ 0.40	$ 1.60
4	Capacitor cerámico	0.22uF /	$ 0,30	$ 1.20
10	Conector hembra	3 y 4 pines	$ 0.20	$ 2
6	Leds colores	Rojo,azul,verde	$ 0,15	$ 0,60
10	Soportes 4mm		$0 .25	$ 2.50
2	Soporte 5mm		$ 0.25	$ 0.50
1	Juego de cable		$ 5	$ 5
1	Tornillos		$1	$ 1
1	Diodo 1N4007		$ 0,12	$ 0,12
1	Baquelita		$4.50	$4.50
1	Conector carga		$2	$2
1	Varios		$15	$15

Subtotal	$ 171.57
IVA 12%	$ 20.58
TOTAL	$ 192.15

BIBLIOGRAFÍA

[1] WILLIAMS, Arthur. (1994). *Manual de Circuitos Integrados: selección, diseño y aplicaciones.* 1ra edición. MacGraw-Hill. México.

[2] Grob, Bernard. (1982). *Circuitos electrónicos y sus aplicaciones.* Editorial Interamericana, S.A. México.

[3] Herrera Pérez Enrique, 1983 *Fundamentos de ingeniería electrónica,* Limusa. México. Mx.

[4] MCGRAW-HILL Science/Engineering/Math; 8 edition (January 27, 1997)

[5] NEWNES; microcontroladores principios y aplicaciones 2 edition .

DIRECCIONES ELECTRONICAS

- http://www.mecanique.co.uk/code-studio/index.html
- http://www.futurisima.com