

CONTENIDO

CONTENIDO	I
RESUMEN	IV
PRESENTACIÓN	V
CAPITULO 1	1
GENERALIDADES	1
1.1 FUNDAMENTOS BÁSICOS DE LA ROBÓTICA	1
1.2 CLASIFICACIÓN DE LA ROBÓTICA	2
1.2.1 BASADO EN LA GENERACIÓN	3
1.2.1.1 Primera Generación	3
1.2.1.2 Segunda Generación	3
1.2.1.3 Tercera Generación.....	3
1.2.1.4 Cuarta Generación	3
1.2.1.5 Quinta Generación	4
1.2.2 BASADO EN EL NIVEL DE INTELIGENCIA	4
1.2.2.1 Dispositivos de manejo manual	4
1.2.2.2 Robots de secuencia arreglada	4
1.2.2.3 Robots de secuencia variable	4
1.2.2.4 Robots regeneradores.....	4
1.2.2.5 Robots de control numérico	5
1.2.2.6 Robots Inteligentes	5
1.2.3 BASADOS EN EL NIVEL DE CONTROL	5
1.2.3.1 Nivel de inteligencia artificial.....	5
1.2.3.2 Nivel de Modo control	5
1.2.3.3 Nivel de Servosistemas	5
1.2.4 BASADO EN EL LENGUAJE DE PROGRAMACIÓN	6
1.2.4.1 Sistemas guiados.....	6
1.2.4.2 Sistemas de programación de nivel-robot.....	6
1.2.4.3 Sistemas de programación de nivel-tarea.....	6
1.2.5 BASADOS EN EL TIPO DE CONTROL	6
1.2.5.1 Robots play-back	6
1.2.5.2 Robots controlados por sensores.....	7
1.2.5.3 Robots controlados por visión	7
1.2.5.4 Robots controlados adaptablemente.....	7
1.2.5.5 Robots con inteligencia artificial	7
1.3 ROBÓTICA MÓVIL	8
1.3.1 CLASIFICACIÓN DE LA ROBÓTICA MÓVIL.....	8
1.3.1.1 Robots móviles con ruedas	8
1.3.1.2 Robots móviles con patas.....	9
1.4 ROBOTS BÍPEDOS	11
1.4.1 ESTUDIOS DE LA LOCOMOCIÓN BÍPEDA 1986	11
1.4.2 DESARROLLO DEL DESPLAZAMIENTO RAPIDO (1987-1991)	12
INVESTIGACIÓN SOBRE ROBOTS HUMANOIDES TOTALMENTE INDEPENDIENTES (1993-1997)	13
1.4.2.1 Robot humanoide P1.....	13

1.4.2.2	Robot humanoide P2.....	13
1.4.2.3	Robot humanoide P3.....	13
1.4.3	HRP DE KAWADA INDUSTRIE	14
1.4.4	MORPH DE VARIAS EMPRESAS JAPONESAS	15
1.4.5	QRIO DE SONY	16
1.5	CARACTERISTICAS DE LA PLATAFORMA MOVIL.....	17
1.5.1	GRADO DE LIBERTAD EN LOS BRAZOS	18
	GRADO DE LIBERTAD EN LAS MANOS.....	19
1.5.2	GRADO DE LIBERTAD EN LA CINTURA.....	19
	GRADO DE LIBERTAD EN LAS PATAS.....	20
CAPÍTULO 2		22
DISEÑO Y ENSAMBAJE DEL PROTOTIPO.....		22
2.1	FUENTE DE ALIMENTACION.....	22
2.2	CONTROL MANUAL.....	23
2.3	PROCESAMIENTO DE DATOS.....	28
2.3.1	MICROPROCESADOR PIC16F877A	29
2.3.2	RETENCION DE DATOS.....	33
2.4	MANEJO DE LOS MOTORES DC.....	35
2.5	SENSORES DE ULTRASONIDO.....	36
2.5.1	TRANSMISOR DE ULTRASONIDO.....	37
2.5.2	RECEPTOR DE ULTRASONIDO	40
2.6	FINALES DE CARRERA	43
2.7	SENSORES DE POSICIÓN.....	44
CAPITULO 3		46
DESARROLLO DEL PROGRAMA DE CONTROL		46
3.1	PROGRAMA PRINCIPAL.....	47
3.2	SUBROUTINAS	48
3.2.1	SUBROUTINA UNO, DOS, SEIS y SIETE.....	48
3.2.2	SUBROUTINA ANALOG	49
	SUBROUTINA TRES, CUATRO, OCHO y NUEVE.....	49
3.2.4	SUBROUTINA CINCO Y DIEZ	50
3.2.5	SUBROUTINA C_SLEEP	51
3.2.6	SUBROUTINA WAKE_UP	51
3.2.7	SUBROUTINA REFRESH	52
3.2.8	SUBROUTINA NO_REFRESH.....	52
3.2.9	SUBROUTINA TRECE	52
3.2.10	SUBROUTINAS ONCE, DOCE, CATORCE Y QUINCE	52
3.2.11	SUBROUTINA ROTA	53
3.2.12	SUBROUTINA (MODO _ AUTÓNOMO).....	54
3.3	INTERRUPCIONES.....	54
3.3.1	INT_RBS.....	54

3.3.1.1	Se activo el fin de carrera del motor de la mano derecha o izquierda.....	55
3.3.1.2	El usuario esta enviando un dato (Busque_RB7).....	55
3.3.2	INT_TIMER0.....	57
3.3.3	SENSORES (Interrupción debido a obstáculos)	58
3.3.3.1	MANUAL.....	58
3.3.3.2	AUTONOMA	58
3.3.4	INT_TIMER1.....	59
3.3.5	CONVERSION	60
3.4	OTRAS SECCIONES DEL PROGRAMA	61
3.4.1	ESCOGE_OPCION.....	61
3.4.2	PRIORIDAD	62
3.4.3	FINAL.....	63
3.4.4	POP	63
3.5	DIAGRAMAS DE FLUJO	64
CAPÍTULO 4	103	
PRUEBAS Y RESULTADOS.....	103	
4.1 PRUEBAS DEL SENSOR DE ULTRASONIDO	103	
4.1.1	PLÁSTICO	103
4.1.2	MADERA.....	103
4.1.3	CUERO.....	104
4.1.4	PAPEL.....	104
4.1.5	VIDRIO	104
4.1.6	ESPONJA	104
4.2 PRUEBAS DEL MODO MANUAL DEL ROBOT	104	
4.2.1	MANEJO DE LOS BRAZOS.....	104
4.2.2	MANEJO DE LAS MANOS	105
4.2.3	MANEJO DE LAS PATAS.....	105
4.2.4	MANEJO DE LA CINTURA.....	106
4.3 PRUEBAS DEL MODO AUTOMATICO DEL ROBOT.....	106	
4.3.1	DISTANCIA DE DETECCIÓN	106
4.3.2	OBJETOS DETECTADOS	106
4.3.2.1	Persona.....	107
4.3.2.2	Silla.....	108
4.3.2.3	Mesa	109
4.3.2.4	Pared, vidrio y cuero.....	110
4.3.3	MOVILIDAD DEL ROBOT	112
4.3.3.1	En superficie plana lisa	112
4.3.3.2	En superficie plana rugosa	112
4.3.3.3	Superficie inclinada lisa	113
CAPÍTULO 5	114	
CONCLUSIONES Y RECOMENDACIONES	114	
REFERENCIAS BIBLIOGRAFICAS	117	

RESUMEN

En el presente proyecto de titulación se desarrolló el control de una plataforma robótica bípeda autónoma, que es capaz de realizar movimientos de avance, es decir el robot posee la habilidad de andar de forma repetitiva. El robot bípedo se compone de 7 grados de libertad (GDL), de los cuales 1 GDL corresponden a cada una de las patas y los otros al resto del cuerpo. Para la obtención de estos movimientos se implementó un sistema de control, que permite al robot moverse libremente en su ambiente de trabajo. El robot tiene dos modos de operación, manual a través de un control remoto infrarrojo con el cual se le envía 14 comandos para su movimiento; o autónomo, en el cual el robot es capaz de evadir obstáculos.

La plataforma bípeda empleó como elemento inteligente un microcontrolador PIC 16F87X trabajando en modo extendido, el mismo que realiza el control integral de la plataforma, verifica la posición de brazos, piernas y el estado del robot con respecto a su alrededor por medio de sensores; además se encarga de decodificar las ordenes emitidas por el control remoto, con lo cual la interrelación entre el usuario y la plataforma robótica es completa.

Para este estudio se tomaron las características del movimiento del robot cuando camina, mediante experimentos, tomando al ser humano como modelo y otros estudios de locomoción en robots bípedos. En proyectos anteriores ya se ha puesto de manifiesto la importancia de sistemas sensoriales eficaces, que doten al sistema de control de una información veraz y coherente con el entorno. Por este motivo el sistema robótico que se presenta posee una serie de sensores capaz de dotarle de una cierta capacidad para obtener información del medio que lo rodea.

PRESENTACIÓN

Al principio de nuestra vida cuando somos seres muy vulnerables, cada acción a realizar es muy compleja y demanda de toda nuestra atención, tal vez no seamos conscientes de una de las habilidades más complicadas del hombre como es caminar o andar en forma repetitiva, la cual nos permite realizar la mayoría de nuestras actividades. En la actualidad existen grandes empresas que han dedicado decenas de años en la investigación para imitar en forma perfecta esta habilidad.

Este proyecto busca controlar una plataforma bípeda autónoma, enfocándose a la parte electrónica (aprovechando el desarrollo de estructuras robóticas existentes). Se encuentra estructurado de la siguiente manera:

En el Capítulo 1 se presenta los fundamentos básicos de la robótica, su clasificación y principalmente las plataformas robóticas bípedas existentes, además se detalla el avance de los estudios de locomoción en este campo, escogiendo entre estos el más adecuado para ser empleado en esta aplicación.

Los movimientos generados en el robot requieren de un conjunto de circuitos adecuados para que el microcontrolador PIC empleado como elemento inteligente relacione correctamente cada una de las señales obtenidas desde los sensores y el control remoto; estos diseños empleados para el control de la plataforma serán analizados en el Capítulo 2.

El microcontrolador PIC cuenta con un programa de control que se encarga de decodificar las ordenes emitidas por el usuario desde el control remoto, al reconocer el comando se toma la acción correspondiente relacionando su estado actual con la orden a realizar, en el Capítulo 3 se da una explicación del programa principal y de las subrutinas empleadas, además se presentan los diagramas de flujo de cada una de ellas.

En el Capítulo 4 se detalla las pruebas del funcionamiento de la plataforma, corrigiéndose los errores que se presenten con la finalidad de tener el diseño más

adecuado y el programa que desempeñe de mejor manera los objetivos buscados.

Para finalizar en el Capítulo 5 se presentan las Conclusiones y Recomendaciones que se generen en el proyecto para futuras investigaciones.

Los anexos se componen de cuatro partes distintas las cuales son:

La información del modo de funcionamiento del robot y como utilizarlo correctamente se encuentra en el manual de usuario, que corresponde al Anexo A.

En el Anexo B se encuentran los esquemáticos, y el diseño del circuito impreso, así como un listado completo de los elementos utilizados en las mismas.

La posición y ubicación de las placas diseñadas, sensores, y fuentes de energía, sobre el robot, se las documenta en los planos de montaje, que son expuestos en el Anexo C

En el Anexo D se presentan las tramas de datos enviadas por el control remoto.

CAPITULO 1

GENERALIDADES

En este capítulo se tratará los principios básicos necesarios para la comprensión y desarrollo de la teoría así como también definiciones que ayudarán a que el lector tenga una percepción de lo que es un robot, su clasificación, la importancia en el mundo contemporáneo y otros temas relacionados al tópico.

1.1 FUNDAMENTOS BÁSICOS DE LA ROBÓTICA

La robótica desde hace cientos de años atrás nace como un anhelo de muchos pueblos de construir dispositivos que copien partes del cuerpo humano con la finalidad de sustituirlo y mejorar el rendimiento dentro de una tarea específica.

Ahora bien, según el Instituto Norteamericano del Robot y de la ISO 8373: “Un robot es un manipulador reprogramable, multifuncional, controlado automáticamente, que puede estar fijo en un sitio o moverse, y que está diseñado para mover materiales, piezas, herramientas o dispositivos especiales, por medio de movimientos variables programados para la realización de diversas tareas o trabajos” [1]. Cabe señalar que esta definición sirvió principalmente para disminuir la ambigüedad entre una máquina y un robot ya que en varias definiciones anteriores pueden entrar dentro del concepto robot máquinas domésticas sin ser estas un robot.

Para una mejor comprensión de la definición de la ISO 8373 es necesario comprender lo que significa manipulador, reprogramable y multifuncional; para lo cual se detalla sus definiciones:

Manipulador: es un mecanismo consistente en un conjunto de segmentos y uniones para mover objetos normalmente en varios grados de libertad [1].

Reprogramable: los movimientos programados o funciones auxiliares pueden modificarse sin que se realicen alteraciones en la estructura mecánica o en el sistema de control, excepto aquellas que suponen cambios de programas y memorias [1].

Multifuncional: es posible que sea adaptado a diferentes aplicaciones con alteraciones en la estructura mecánica o en el sistema de control [1].

Es necesario acotar que un robot tiene una capacidad de procesamiento inferior a la de un ser humano, pero puede realizar operaciones limitadas de forma más rápida y precisa sin importar las condiciones de trabajo y realizándolo en forma continua, sin que esto implique salirse de las tres leyes que rigen la inteligencia de los robots, las cuales son[2]:

1. Un robot no debe dañar a un ser humano o, por inacción, dejar que un ser humano sufra daño.
2. Un robot debe obedecer las órdenes que le son dadas por un ser humano, excepto cuando estas órdenes estén en oposición con la primera ley.
3. Un robot debe proteger su propia existencia hasta donde esta protección no esté en conflicto con la primera o la segunda ley.

Estas leyes fueron dadas por el científico ruso Isaac Asimos (1920-1992) en su obra "I Robot" con la finalidad de terminar con mitos populares sobre el dominio del mundo por los robots.

1.2 CLASIFICACIÓN DE LA ROBÓTICA

No se tiene una clasificación exacta dentro de la robótica ya que se puede hacer múltiples clasificaciones en función de sus características entre las que se destacan: generación, nivel de inteligencia, nivel de control, y el nivel de lenguaje de programación.

1.2.1 BASADO EN LA GENERACIÓN [1]

Según la generación los robots se clasifican en cinco generaciones las cuales se detallan a continuación:

1.2.1.1 Primera Generación

El sistema de control usado en la primera generación de robots está basado en “paradas fijas” mecánicamente. Esta estrategia es conocida como control de lazo abierto. Se puede considerar como ejemplo de esta primera etapa aquellos mecanismos de relojería que permiten mover a las cajas musicales o a los juguetes de cuerda.

1.2.1.2 Segunda Generación

Utiliza una estructura de control de ciclo abierto, pero en lugar de utilizar interruptores y botones mecánicos utiliza una secuencia numérica de control de movimientos almacenados en un disco o cinta magnética. El programa de control entra mediante la elección de secuencias de movimiento en una caja de botones o a través de palancas de control para determinar la secuencia deseada en movimientos. El mayor número de aplicaciones en los que se utilizan los robots de esta generación son de la industria automotriz, en soldadura, pintado con “spray”.

1.2.1.3 Tercera Generación

La tercera generación de robots utiliza las computadoras para su estrategia de control y tiene algún conocimiento del ambiente local a través del uso de sensores, los cuales miden el ambiente y modifican su estrategia de control, con esta generación se inicia la era de los robots inteligentes y aparecen los lenguajes de programación para escribir los programas de control. La estrategia de control utilizada se denomina de “ciclo cerrado”

1.2.1.4 Cuarta Generación

La cuarta generación de robots, ya los califica de inteligentes con más y mejores extensiones sensoriales, para comprender sus acciones y el mundo que los rodea. Incorpora un concepto de “modelo del mundo” de su propia conducta y del ambiente en el que operan. Utilizan conocimiento difuso y procesamiento dirigido

por expectativas que mejoran el desempeño del sistema de manera que la tarea de los sensores se extiende a la supervisión del ambiente global, registrando los efectos de sus acciones en un modelo del mundo y auxiliar en la determinación de tareas y metas.

1.2.1.5 Quinta Generación

Actualmente está en desarrollo esta nueva generación de robots, que pretende que el control emerja de la adecuada organización y distribución de módulos conductuales, esta nueva arquitectura es denominada arquitectura de subsumición, cuyo promotor es Rodney Brooks.

1.2.2 BASADO EN EL NIVEL DE INTELIGENCIA [1]

Para esta clasificación se toma como referencia a JIRA (Asociación de Robots Japonesa) la cual determina seis tipos de robots que son:

1.2.2.1 Dispositivos de manejo manual

Aquellos elementos y dispositivos manejados por una persona, es decir, manualmente.

1.2.2.2 Robots de secuencia arreglada

Tienen preestablecida la secuencia de acciones no modificables durante la ejecución de la tarea.

1.2.2.3 Robots de secuencia variable

Donde un operador puede modificar la secuencia fácilmente.

1.2.2.4 Robots regeneradores

Donde el operador humano conduce el robot a través de la tarea.

1.2.2.5 Robots de control numérico

Donde el operador alimenta la programación del movimiento, hasta que se enseñe manualmente la tarea.

1.2.2.6 Robots Inteligentes

Los cuales pueden entender e interactuar con cambios en el medio ambiente. Claro que dicha interacción se basa en un buen sistema de control sensorial que le permita al robot interpretar su entorno físico.

1.2.3 BASADOS EN EL NIVEL DE CONTROL [1]

Se refiere al tipo de nivel de control establecido en el controlador del robot; aquí se tiene la siguiente clasificación:

1.2.3.1 Nivel de inteligencia artificial

Donde el programa aceptará un comando y lo descompondrá dentro de una secuencia de comandos de bajo nivel basados en un modelo estratégico de tareas.

1.2.3.2 Nivel de Modo control

Donde los movimientos del sistema son modelados, para lo que se incluye la interacción dinámica entre los diferentes mecanismos, trayectorias planeadas, y los puntos de asignación seleccionados.

1.2.3.3 Nivel de Servosistemas

Donde los actuadores controlan los parámetros de los mecanismos con el uso de una retroalimentación interna de los datos obtenidos por los sensores, y la ruta es modificada sobre la base de los datos que se obtienen de sensores externos. Todas las detecciones de fallas y mecanismos de corrección son implementados en este nivel.

1.2.4 BASADO EN EL LENGUAJE DE PROGRAMACIÓN [1]

Trata sobre el tipo de programación que es la clave para una aplicación efectiva de los robots para una amplia variedad de tareas, es el desarrollo de lenguajes de alto nivel. Dentro de este grupo se clasifica a los robots en las siguientes clases:

1.2.4.1 Sistemas guiados

En el cual el usuario conduce el robot a través de los movimientos a ser realizados.

1.2.4.2 Sistemas de programación de nivel-robot

En el cual el usuario escribe un programa de computadora donde especifica el movimiento y el sensado.

1.2.4.3 Sistemas de programación de nivel-tarea

En el cual el usuario especifica la operación por sus acciones sobre los objetos que el robot manipula.

Como se puede entender existen varias formas de encasillar a los robots según sus características, es por tanto que se ha puesto dos clasificaciones más que aunque no son muy conocidas demuestran una forma diferente de categorizar.

La clasificación que se va a presentar a continuación es la más completa y estructural que se ha podido consultar, a más de ser sustentada con suficiente base teórica lo que la hace didáctica; y es [3]:

1.2.5 BASADOS EN EL TIPO DE CONTROL [3]

Tema que hace referencia al método utilizado por el robot para tomar decisiones de tarea, dentro de este ítem se tiene:

1.2.5.1 Robots play-back

Los cuales regeneran una secuencia de instrucciones grabadas. Estos robots comúnmente tienen un control de lazo abierto.

1.2.5.2 Robots controlados por sensores

Estos tienen un control en lazo cerrado de movimientos manipulados, y toman decisiones basados en datos obtenidos por sensores.

1.2.5.3 Robots controlados por visión

Donde los robots pueden manipular un objeto al utilizar información desde un sistema de visión.

1.2.5.4 Robots controlados adaptablemente

Donde los robots pueden automáticamente reprogramar sus acciones sobre la base de los datos obtenidos por los sensores. El robot tiene control adaptativo, sensorial o control mediante aprendizaje, se dice que el control es adaptativo si los parámetros del sistema de control son ajustados a partir de las condiciones detectadas durante el proceso; es sensorial, si el movimiento del robot y la fuerza de éste se ajusta de acuerdo a las salidas que proporcionan sensores externos; y, es inteligente cuando la experiencia obtenida durante ciclos o tareas anteriores se usa de modo automático para cambiar parámetros de control y/o algoritmos.

1.2.5.5 Robots con inteligencia artificial

Donde los robots utilizan las técnicas de inteligencia artificial para tomar sus propias decisiones y resolver problemas.

Y como otra forma de clasificación se tiene aquella, en donde no se toma en cuenta criterios científicos sino que se crea a partir de la perspectiva de los autores, y está en el libro "Robot y Sistemas Sensoriales" [4] los autores muestran la siguiente:

- ❖ Humanoide: un robot inspirado en la apariencia humana.
- ❖ Robot móvil: un robot dotado con movilidad.
- ❖ Robot industrial: robot diseñado para tareas industriales.

- ❖ Robot inteligente: un robot totalmente autónomo cuyo comportamiento depende de su programación y de la información adquirida en base de sus sistemas sensoriales, tiene la capacidad de retroalimentación (aprendizaje).
- ❖ Robot de servicios: robot parcial o totalmente autónomo programado para tareas útiles.

En el presente proyecto se dará más énfasis a los robots móviles y principalmente a los robots humanoides debido a que su estudio es el campo de investigación del presente trabajo.

1.3 ROBÓTICA MÓVIL

Con el propósito de aumentar la movilidad del robot y de esta manera su espacio de trabajo, se sitúa la necesidad de utilizar la robótica móvil. En este caso el robot hace uso de su sistema locomotor para interactuar con el medio que lo rodea, la movilidad presente en el robot implica que dicho elemento necesitará un sistema de sensores para participar en el entorno y redefinir sus movimientos.

Los robots móviles pueden aplicarse en un sin número de utilidades principalmente de servicio.

1.3.1 CLASIFICACIÓN DE LA ROBÓTICA MÓVIL

Básicamente se pueden clasificar los robots móviles atendiendo a su modo de desplazamiento; así, se distinguen entre robots con ruedas y con patas [4].

1.3.1.1 Robots móviles con ruedas

Frente a la otra clase de robots móviles (los robots con patas) estos disponen de un ámbito de aplicación más extenso debido a que se requiere un control sencillo, menos cantidad de energía y los movimientos no son bruscos y son rápidos.

Las características y disposición de las ruedas en el robot le van a permitir adaptarse al tipo de terreno en que desarrolle su tarea. Estos robots móviles

presentan buena estabilidad, pero habitualmente necesitan de sistemas de suspensión para asegurar que las ruedas de tracción y/o dirección estén en el suelo.

Con el objetivo de no instalar los sistemas de suspensión, es adecuado utilizar robots de tres ruedas, éstos últimos garantizan un mejor contacto con el suelo y se puede afirmar que el centro de gravedad del robot se encontrará en el centro del círculo que pasa por los tres puntos de contacto del robot con el suelo.

El uso de ruedas para realizar el desplazamiento conlleva problemas cuando se pretende realizar el desplazamiento en un terreno accidentado o escarpado, en cuyo caso las ruedas deber ser lo suficientemente grandes para pasar los obstáculos.



Figura 1.1 Robot con ruedas, Pionero 1

1.3.1.2 Robots móviles con patas

Su principal aplicación se extiende a aquellas zonas a las que un vehículo con ruedas no pueda alcanzar, por la orografía del terreno por ejemplo. El uso de patas permite al robot desplazarse en terrenos escarpados evitando y pasando por encima de los obstáculos. A pesar de la maniobrabilidad que permite este tipo

de tracción o la posibilidad de realizar tareas como subir escaleras, el uso de patas suele ser minoritario frente a la tracción con ruedas, y la mayoría de las aplicaciones existentes suelen ser de investigación.

En la actualidad es posible encontrar robots móviles que disponen desde dos a varias patas. Así dentro de éstos se pueden clasificar en robots móviles con *estabilidad estática o dinámica*. A menudo los robots con estabilidad dinámica disponen de menos de 3 patas y su estabilidad es lograda realizando los movimientos continuos del cuerpo o las patas. Estos movimientos no son requeridos en el caso de disponer de estabilidad estática, ya que el robot puede posicionarse firmemente en el suelo sin necesidad de corregir su posición continuamente.



Figura 1.2 Robot Hexápodo

1.4 ROBOTS BÍPEDOS

Son parte de la robótica móvil y más concretamente de los robots con patas; como su nombre lo indica son los que poseen dos patas para su locomoción y los más conocidos son los humanoides o androides, ya que tratan y utilizan al hombre como modelo de diseño.

Para entender aún mejor la locomoción bípeda se presenta la siguiente cronología:

1.4.1 ESTUDIOS DE LA LOCOMOCIÓN BÍPEDA 1986 [5]

Se creó un robot bípedo que camina; dando las siguientes conclusiones: la primera etapa que gravitaba en andar se cubrió satisfactoriamente aunque el robot se desplazaba muy lentamente, tardaba 5 [seg.], en cada paso. Para aumentar la velocidad de marcha o para que pudiera caminar en superficies desiguales fue necesario el desarrollo de la marcha rápida, la cual consistía en dar el mayor número de pasos en el menor tiempo posible.

En la Figura 1.3 se detalla la diferenciación de los dos tipos de marcha que se necesitó cubrir en estos estudios dando como principal conclusión el movimiento del centro de gravedad tanto en marcha lenta como en marcha rápida.

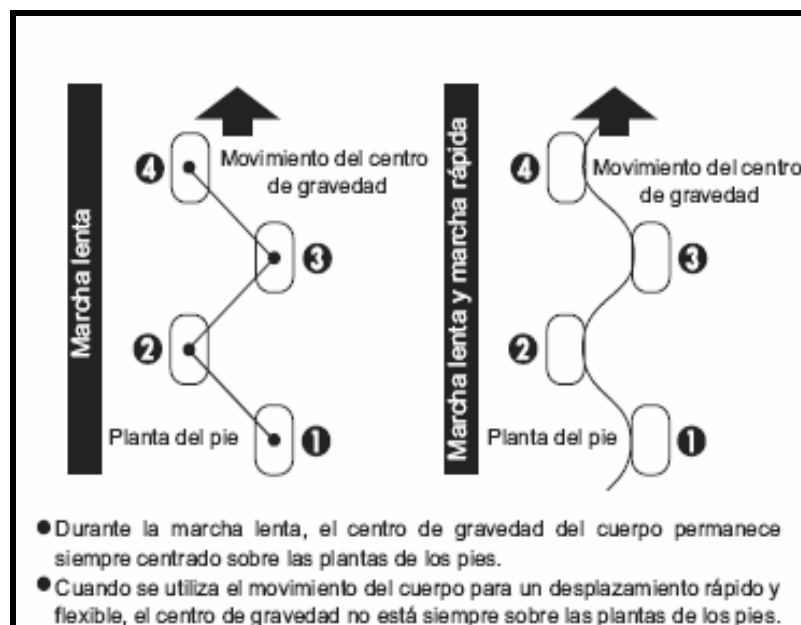


Figura 1.3 Marcha lenta y marcha rápida, tomado de [5]

1.4.2 DESARROLLO DEL DESPLAZAMIENTO RÁPIDO (1987-1991) [5]

Para conseguir un desplazamiento rápido fue necesario estudiar como andan los seres humanos, por lo que se investigó y analizó rigurosamente el caminar humano. Además del andar humano, también se estudiaron el desplazamiento animal y otras formas de desplazamiento, así como las articulaciones necesarias para caminar. Basándose en los datos derivados del desplazamiento humano, se desarrolló un programa de desplazamiento rápido, llegando a alcanzar una velocidad de marcha rápida de 1,2 [Km/h] en superficies planas para el robot E3 (Ver Figura 1.4)

En la Figura 1.4 se muestran los robots desarrollados para estos experimentos:

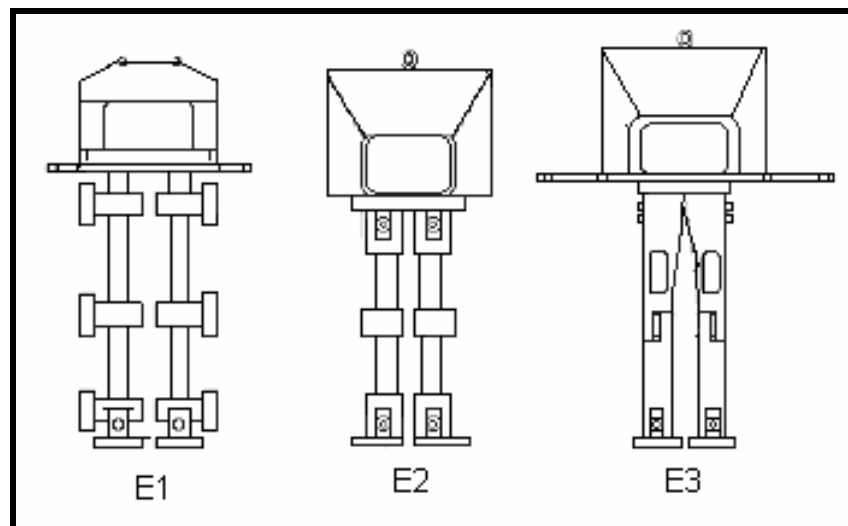


Figura 1.4 Robots desarrollados para el estudio humano, tomado de [5]

La siguiente etapa era conseguir un desplazamiento rápido y estable en un entorno de vida humana especialmente sobre superficies irregulares, pendientes y escaleras, sin que se cayera.

INVESTIGACIÓN SOBRE ROBOTS HUMANOIDES TOTALMENTE INDEPENDIENTES (1993-1997) [5]

Se desarrolló progresos en los robots humanoides, dichos progresos consistían en la interacción y utilización de los robots en tareas básicas del ser humano.

Se desarrollaron varios robots destacándose los siguientes:

1.4.2.1 Robot humanoide P1

Su altura era de 1.915 [m] y su peso de aproximadamente 175 [Kg.]. Dentro de las tareas que podía realizar estaban las de activar y desactivar interruptores eléctricos y de ordenador, abrir puertas, coger y transportar objetos. También se llevó a cabo la investigación de movimiento de brazos y piernas.

Sus detalles se pueden ver en la Figura 1.5 (izquierda).

1.4.2.2 Robot humanoide P2

Su altura era de aproximadamente 1.820 [m] y tendría un peso de 210[Kg.]; y, a diferencia del P1 el P2 tenía más adecuaciones principalmente en el torso donde se integraron, utilizando técnicas sin hilos, un ordenador, mecanismos de accionamiento motorizado, una batería. Se consiguió que, de manera autónoma y sin hilos, se desplazará independiente, subiera y bajara escaleras. Su estructura se puede ver en la Figura 1.5 (centro).

1.4.2.3 Robot humanoide P3

Fue el primer robot humanoide bípedo completamente independiente, terminado en 1997. Su altura disminuyó en comparación a los otros dos y estaba en aproximadamente 1.6 [m] dando lugar a un peso de 130 [Kg.]. Este cambio se dio principalmente ha que se habían cambiado los materiales de los componentes y descentralizado el sistema de control.

En la Figura 1.5 se muestran el robot antes detallado (derecha).

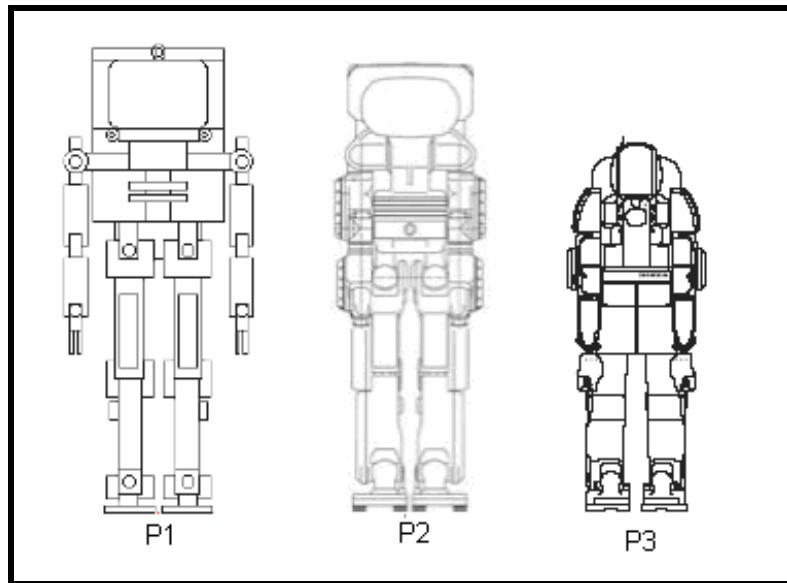


Figura 1.5 Progresos de los robots humanoides, tomado de [5]

Con esta introducción se da una idea general del desarrollo de la robótica bípeda; a continuación se presenta diferentes tipos de robots humanoides creados por diferentes marcas.

1.4.3 HRP DE KAWADA INDUSTRIE [6]

HRP fue desarrollado por el "Humanoid Research Group" del Instituto de Ciencia y Tecnología Avanzada del Japón (AIST) y por la empresa "Kawada Industries". HRP-2P mide 1.54 [m] de alto y pesa 58 [Kg] (incluyendo las baterías, ver Figura 1.6). Tiene 30 grados de libertad y el diseño de sus caderas y entrepierna le permite caminar en lugares estrechos. Se ha empleado en su diseño una electrónica de alta densidad, lo que permite que en una pequeña "mochila" se integren los ordenadores, electrónica y baterías necesarios para su funcionamiento. El HRP-2P será empleado en experimentos donde probarán el caminar en terrenos inestables o desiguales. Las caídas controladas, levantarse del suelo cuando ha caído, etc., son tareas básicas para los robots implicados en la interacción con los seres humanos.

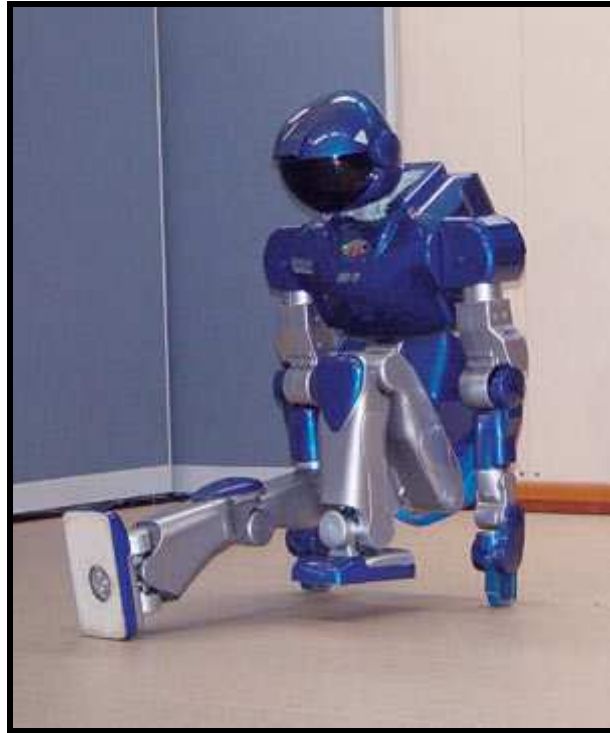


Figura 1.6 Robot HRP, tomado de [6]

1.4.4 MORPH DE VARIAS EMPRESAS JAPONESAS [6]

Morph es un desarrollo combinado de Kitano Symbiotic System Project y Leading Edge Design Corp. La estructura fue diseñada y fabricada por NICHINANCorp.

Al igual que el resto de robots, se encuentra en una versión avanzada. Esta versión pesa 2,4 [Kg] y mide 0.380 [m] de altura (ver Figura 1.7). Con respecto al diseño presenta una estructura modular con un componente central al que se le han añadido funcionalidades diversas.

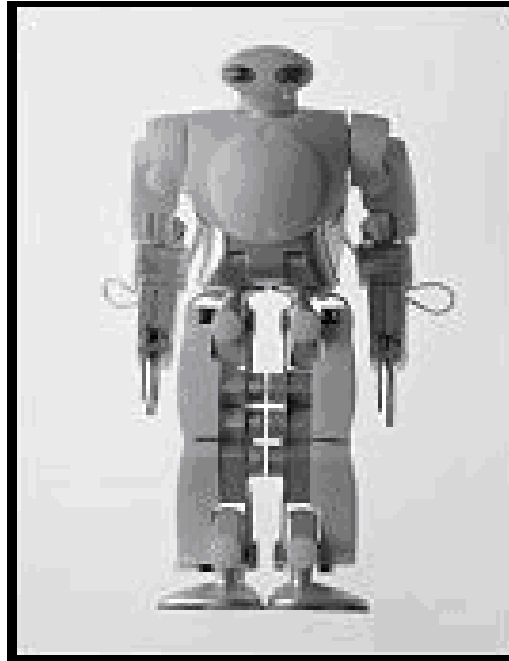


Figura 1.7 Robot Morph, tomado de [6]

1.4.5 QRIO DE SONY [6]

QRIO es un robot humanoide desarrollado por SONY. Antiguamente se denominó "SDR" y pasó por numerosas versiones: SDR-3X, SDR-4X, después SDR-4XII y finalmente fue rebautizado a mediados de 2003 como QRIO (un nombre más apropiado de cara al mercado). "QRIO" es una abreviatura para "Quest for curiosity", que en castellano significa "Búsqueda de la curiosidad". El robot de Sony apenas mide 0.60 [m] (ver Figura 1.8) en el apartado tecnológico dispone de una tecnología denominada "Intelligent Servo Actuator" que es lo que le permite andar dinámicamente (variando r.p.m. y torque en las articulaciones) y emplea una técnica denominada "Zero Moment Point" para mantener la estabilidad. Ambas tecnologías están sobradamente estudiadas y el robot de Sony (QRIO) no es el único que las emplea, otros robots japoneses como el ASIMO de Honda se basa en técnicas de semejantes principios tratando siempre de trabajar sobre una tecnología.



Figura 1.8 Robot QRIO de Sony, tomado de [6]

1.5 CARACTERISTICAS DE LA PLATAFORMA MOVIL

La plataforma bípeda utilizada consta de siete grados de libertad (7 GDL); uno en cada brazo, un GDL en cada muñeca, otro en la cintura y dos más, uno en cada pata. En la Figura 1.9 se muestran los grados de libertad que tiene la plataforma bípeda utilizada en el presente proyecto.

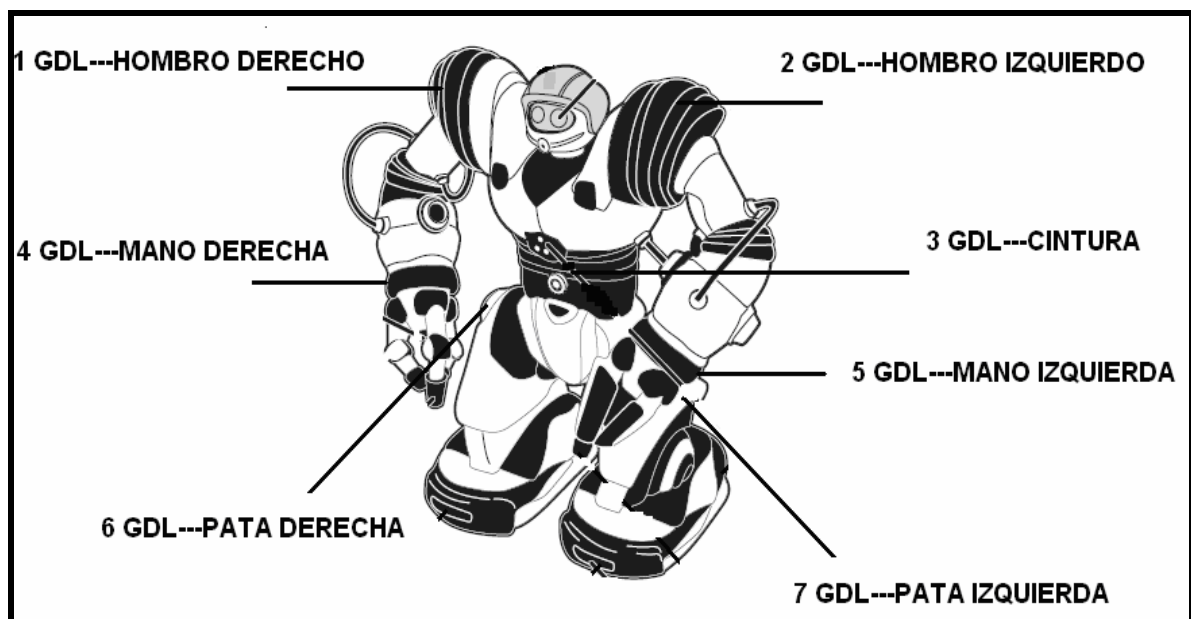


Figura 1.9 Plataforma bípeda y sus grados de libertad

Todos estos grados de libertad que posee el robot sirven para realizar los movimientos que son establecidos por el usuario mediante la utilización del control remoto.

Los grados de libertad que posee la plataforma bípeda utilizada son generados mediante dos engranajes perfectamente diseñados; donde el eje de uno de ellos es movilizado por un pequeño motor DC. Dichos engranajes se encuentran acoplados de tal forma que generen tanto el movimiento vertical como el horizontal necesario.

La plataforma bípeda al no tener un grado de libertad en la mitad de la pata para que trabaje como rodilla y así formar dos partes independientes y no solo una parte rígida, no puede alzar la pata y así pasar obstáculos.

1.5.1 GRADO DE LIBERTAD EN LOS BRAZOS

Permiten que el robot pueda movilizar el brazo derecho e izquierdo para arriba y para abajo según el comando enviado por el usuario. Para este movimiento el motor DC de cada hombro se encuentra acoplado al eje de un engrane horizontal que al moverse acciona el engrane vertical necesario para el movimiento. Dichos movimientos se pueden observar en la Figura 1.10.

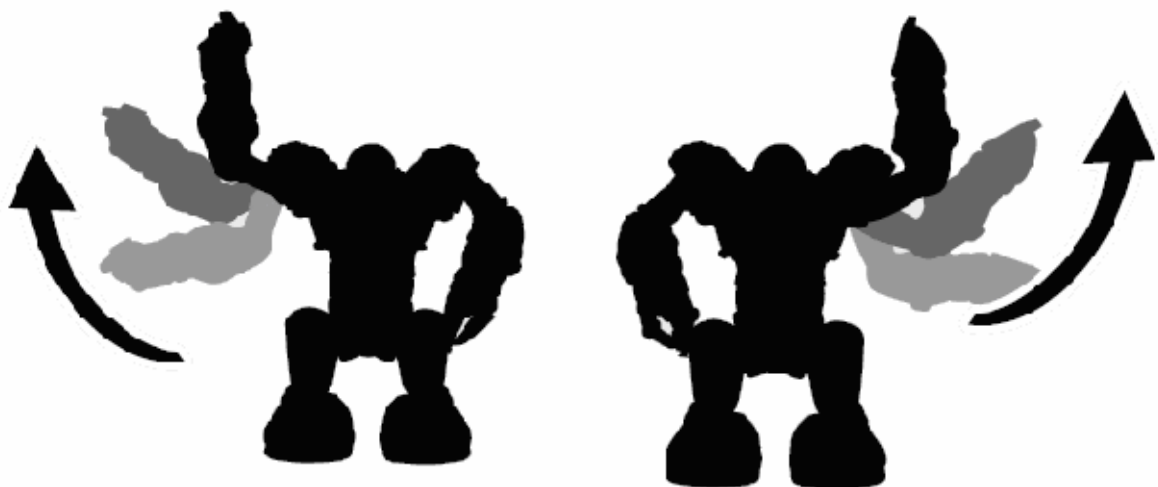


Figura 1.10 Movimientos de los brazos

GRADO DE LIBERTAD EN LAS MANOS

El grado de libertad de las manos le permiten a la plataforma bípeda moverlas hacia dentro o hacia fuera tomando como eje una unión existente en el brazo que equivale al codo del mismo. Dicho movimiento puede realizarse tanto en la mano derecha como en la izquierda; y, para conseguirlo el robot tiene un motor DC que acciona un engrane acoplado directamente a la unión interna existente en el brazo.

Cuando el movimiento es hacia fuera la mano se abre y se cierra si el movimiento es para dentro (ver Figura 1.11).

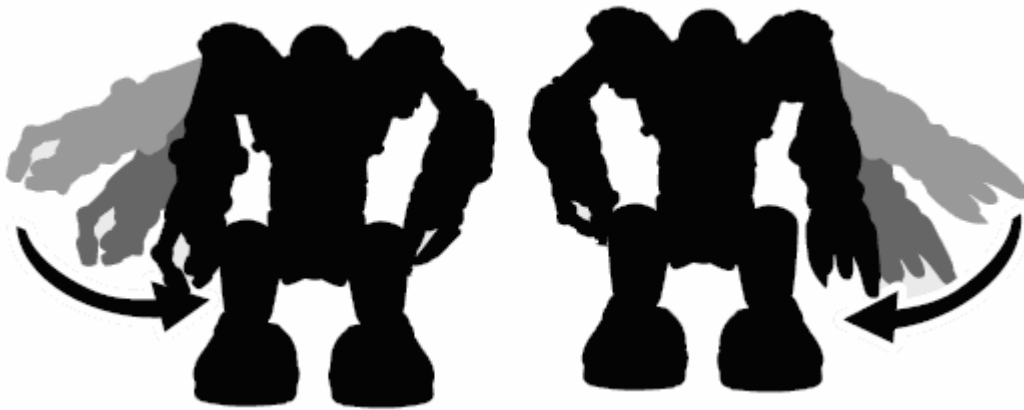


Figura 1.11 Movimiento de las manos

1.5.2 GRADO DE LIBERTAD EN LA CINTURA

En primera instancia este movimiento de la cintura servía para que el robot pueda inclinar su dorso hacia la derecha o izquierda sin necesidad de movilizar sus patas; sin embargo, esta habilidad se constituyó en una secuencia muy importante en la rutina de caminar ya que en la programación se establece un movimiento rápido de la cintura para que la plataforma bípida pueda movilizarse. El máximo desplazamiento del dorso tanto para la derecha como para la izquierda está dado por una condición de construcción del robot más que por finales de carrera o tiempo.

Dichos movimientos de la cintura se pueden apreciar en la Figura 1.12

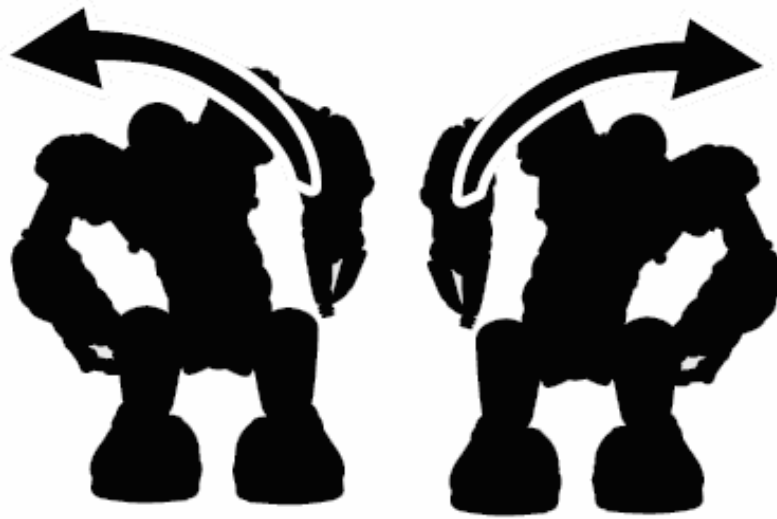


Figura 1.12 Movimiento de la cintura

GRADO DE LIBERTAD EN LAS PATAS

Dicho grado de libertad sirve para que la plataforma robótica pueda movilizarse para el frente y para atrás (ver Figura 1.13) de su trayectoria. Para generar este movimiento cada una de las patas tiene un conjunto de engranajes de plástico que son accionados por un motor de DC.

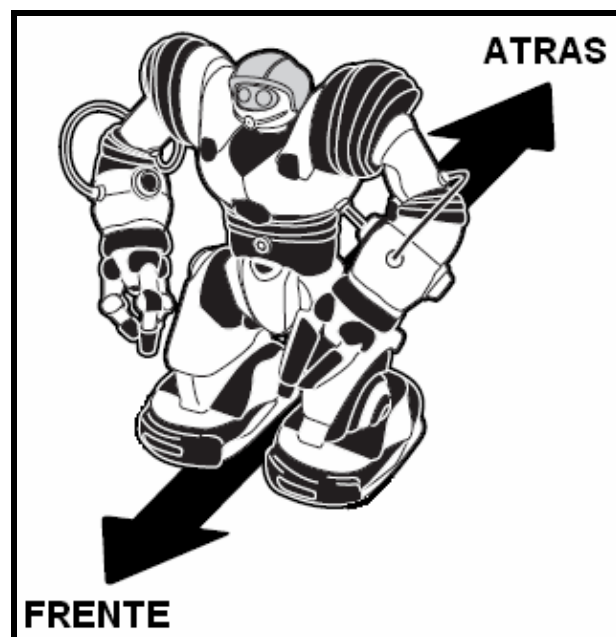


Figura 1.13 Movimiento de las patas

En el siguiente capítulo se verá con detalle el diseño del hardware utilizado para el presente proyecto, es decir, los sensores utilizados así como cada uno de los elementos que componen el hardware.

CAPÍTULO 2

DISEÑO Y ENSAMBAJE DEL PROTOTIPO

De manera general cabe indicar que en el presente proyecto se trabaja con el microprocesador PIC16F877A como cerebro del robot, se usan también resistencias variables como sensores de posición, sensores de ultrasonido y también finales de carrera en cada extremidad del cuerpo del robot para detectar objetos cuando el robot está en forma automática.

El microprocesador maneja actuadores para operar los siete motores DC que tiene el robot y los leds de señalización de la cabeza del mismo, los cuales se prenden según las secuencias de movimiento previamente enviadas por el usuario, utilizando un control remoto.

El robot tiene dos modos de funcionamiento: funcionamiento manual donde hace caso a todos los botones del control remoto; y, el modo automático donde trabaja en base de los sensores de ultrasonido y a dos botones del control remoto.

En las siguientes secciones se tratará con detalle cada uno de los componentes que forma parte de la plataforma móvil; motivo del presente trabajo.

2.1 FUENTE DE ALIMENTACION

Existe una sola fuente de alimentación de 4.8 [V] generada por 4 pilas de 1.2 [V].

El esquema general de la fuente de alimentación y los integrados que alimenta se puede apreciar en la Figura 2.1.

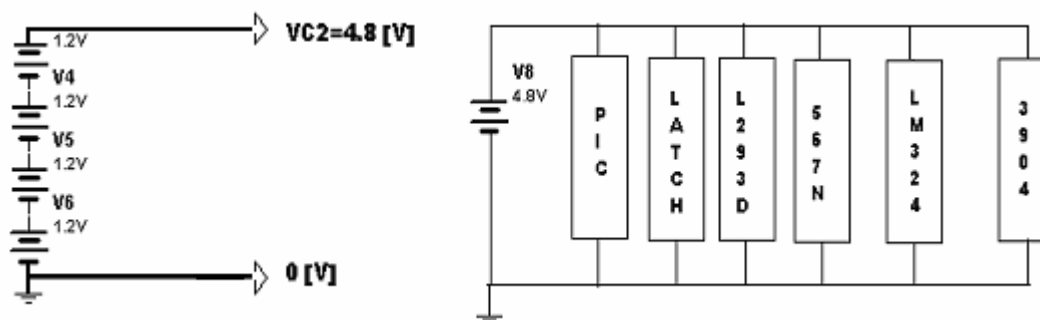


Figura 2.1 Fuentes de Alimentación

2.2 CONTROL MANUAL

El control manual se lo realiza utilizando un control remoto (Figura 2.2), el cual tiene como finalidad la transmisión de señales (creación de datos), dicho elemento consta de un transmisor de infrarrojos (ubicado en la parte superior del control) encargado de emitir varias señales las mismas que eran captadas por un receptor (ubicado en la cabeza del robot) y que van a ser interpretadas por el microcontrolador para determinar cada uno de los movimientos del robot ya que cada uno de los botones envía una señal diferente a interpretar.

Refiriéndose a la numeración de la Figura 2.2, los botones con los que se trabajó se presentan en la Tabla 2.1.

Tabla 2.1 Botones del Control Remoto

Número de Tecla	Comando
1	Levanta el brazo derecho
2	Baja el brazo derecho
3	Mueve la mano derecha hacia dentro-dos posiciones
4	Mueve la mano derecha hacia fuera-dos posiciones
5	Mueve la parte superior del cuerpo hacia la derecha
5(SLEEP)	Modo SLEEP del robot
6	Levanta el brazo izquierdo
7	Baja el brazo izquierdo
8	Mueve la mano izquierda hacia dentro-dos posiciones
9	Mueve la mano izquierda hacia fuera-dos posiciones
10	Mueve la parte superior del cuerpo hacia la izquierda
11	Camina hacia adelante
12	Gira a la derecha
13	STOP-detiene el robot y la función que este realizando
14	Gira a la izquierda
15	Camina hacia atrás
R	Activa los sensores de ultrasonido para pruebas
19	Modo autónomo
20	WAKE-UP

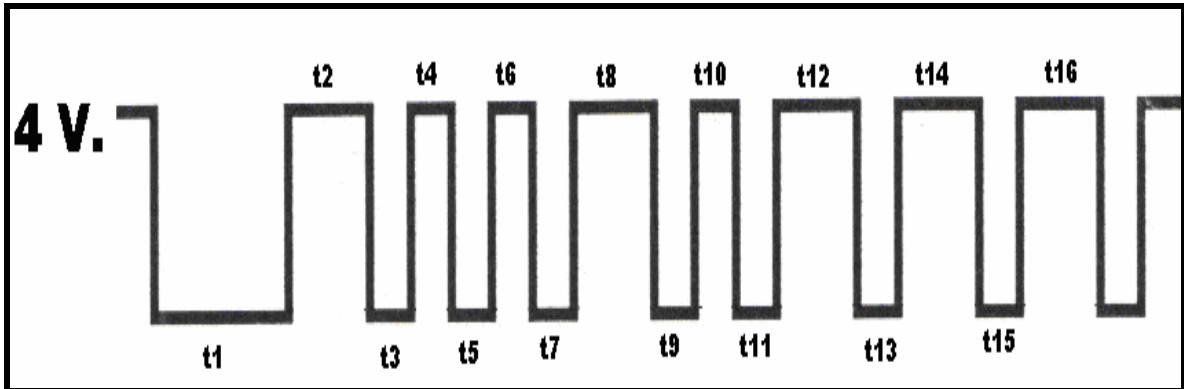


Figura 2.3 Señal emitida por el control remoto

Con un proceso experimental se registró que todas las tramas enviadas por el control remoto tenían apariencia similar a la señal de la Figura 2.3, se vuelve a recalcar que la única diferencia son los tiempos en alto que registraba cada una de ellas.

Las tramas recibidas se pueden apreciar en la Tabla 2.2 para los cuatro primeros botones del control remoto. Para entender mejor la Tabla 2.2 cabe indicar que en la columna de botón se hace referencia a la numeración dada por la Figura 2.2 y todos los tiempos desde t_1 a t_{16} están en milisegundos. Los datos que se muestran son de los cuatro primero botones ya que los demás seguían el mismo patrón.

A partir de estos valores se procedió a encontrar un patrón que genere una trama de 8 bits y satisfaga a todos los datos obtenidos, es en virtud de esto que al observar la tabla se nota que tanto el t_1 como el tiempo después del t_{16} son iguales o notablemente parecidos en todas las señales por lo que no se le toma en cuenta para obtener la trama no así los tiempos t_2 , t_3 hasta t_{15} .

Tabla 2.2 Datos obtenidos experimentalmente de la señal del control remoto

Botón	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10	t11	t12	t13	t14	t15	t16
1	6.8	3.2	0.9	0.7	0.9	0.8	0.7	0.8	0.7	0.9	0.9	0.8	0.9	0.8	3.3	0.8
2	6.6	3.2	0.9	0.9	0.8	0.8	0.8	0.9	0.8	0.9	3.3	0.9	0.9	0.8	0.8	0.8
3	6.3	0.8	0.9	0.8	0.7	0.8	0.9	0.8	0.7	0.9	0.9	0.8	0.9	0.8	3.3	0.8
4	6.6	0.8	0.9	0.8	0.9	0.8	0.9	0.8	0.7	0.9	0.9	0.8	0.9	0.8	0.9	0.8

El hecho también de que todos los tiempos en bajo de las señales son constantes ayudo a que la trama a encontrar sea más fácil por lo que se sumó cada uno de estos tiempos a su sucesor y así se pudo realizar la trama requerida considerando que los tiempos menores a 2.2 [mseg] equivalen a cero y los mayores a este tiempo equivalen a uno.

Con este elemento de juicio se establece las tramas de cada uno de las señales emitidas por el control remoto. En la Tabla 2.3 se nota dicha trama para los dos primeros botones (referirse a Figura 2.2).

Tabla 2.3 Trama de datos de 8 bits

Botón	Dato7	Dato6	Dato5	Dato4	Dato3	Dato2	Dato1	Dato0
1	1	0	0	0	0	0	0	1
2	1	0	0	0	0	1	0	0

El determinar la trama es muy importante para facilitar la programación (más detalle en el capítulo 3), sin embargo no sólo se debe interpretar la trama generada por el control sino también mantener y transmitir al PIC para su procesamiento e interpretación.

Razón por la cual en el hardware se utilizó un circuito seguidor de tensión no inversor (Figura 2.4) para lo cual se empleó el integrado LM324. El circuito se utilizó para acondicionar la señal antes de ingresar al puerto RB7 del microcontrolador para su procesamiento.

La salida del circuito está conectada a una resistencia de 100 ohmios, protección de la entrada del puerto del PIC.

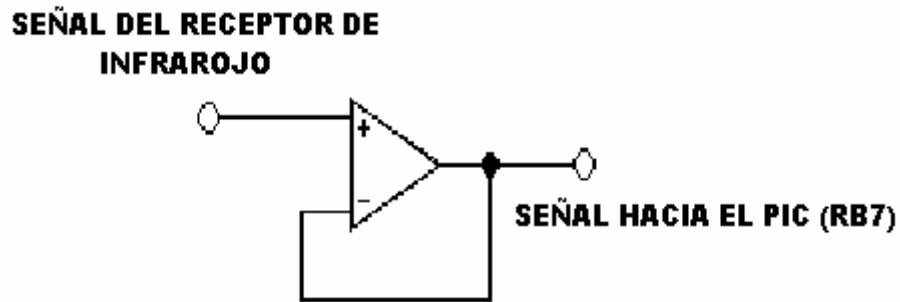


Figura 2.4 Circuito seguidor de tensión

A más de ello el integrado sirve para generar un voltaje de 4 [V] mediante un potenciómetro para alimentar el receptor de infrarrojos ubicado en la cabeza del robot, Figura 2.5.

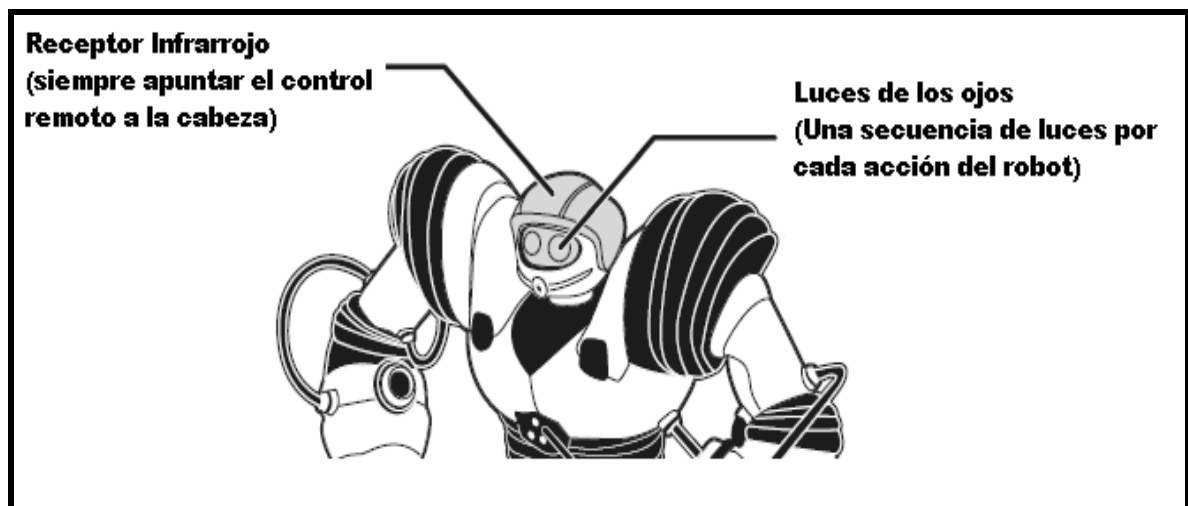


Figura 2.5 Receptor de infrarrojo y leds de la cabeza

Ya con la trama de datos en el PIC se puede proceder al manejo de los mismos.

2.3 PROCESAMIENTO DE DATOS

El proyecto desarrollado tiene como núcleo principal al microprocesador PIC16F877A (Figura 2.6) que se presenta como el cerebro del proyecto ya que es el que toma las decisiones de movimiento de los motores y funcionamiento de los sensores que complementan su accionar. Dicho dispositivo es el que procesa las señales entregadas por el control remoto del robot, funcionamiento

manual, para identificar cada una de las acciones grabadas y así generar la trama de datos necesaria para los drivers de los motores, responsables de la movilidad del robot. Cabe indicar que los motores de los hombros y patas del robot constan de sensores de posición que le ayudan a llegar a las coordenadas adecuadas para su movilidad (mayor información de los sensores de posición en la sección 2.7).

En el funcionamiento automático el robot complementa al microprocesador con las señales creadas por los sensores externos (sensores de obstáculos) y los finales de carrera propios de robot, ubicados en cada extremidad del mismo; es claro señalar que los finales de carrera trabajan tanto en el modo manual como en el automático.

2.3.1 MICROPROCESADOR PIC16F877A

El microprocesador es el responsable de interpretar todos los datos obtenidos en la interacción del robot con el entorno físico que lo rodea orientado principalmente a detectar obstáculos y los datos obtenidos en las tramas generadas por el control remoto. Su distribución de pines se puede apreciar en la Figura 2.6.

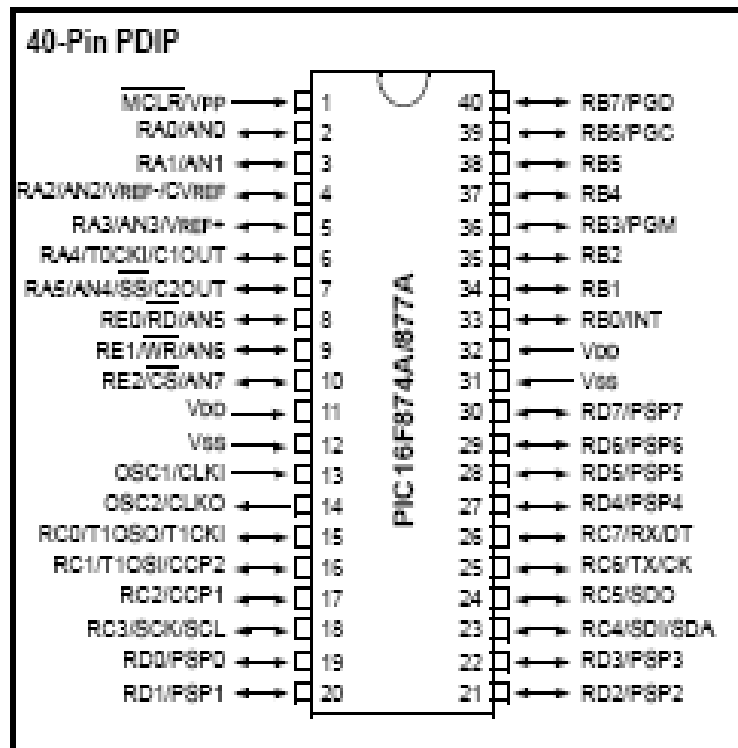


Figura 2.6 PIC 16F877A

Los puertos del PIC trabajan como entradas o salidas según como hayan sido programados, en el trabajo desarrollado se configuraron de la siguiente manera:

- ◆ Puerto A: Entrada de los sensores de posición
- ◆ Puerto B: Entrada de los sensores de obstáculos y de la señal del control remoto.
- ◆ Puerto C: Salida-generación de PWM, habilitación de latches; y, entrada de los finales de carrera.
- ◆ Puerto D: Salida de los datos para el manejo del robot- trama de 8 bits.
- ◆ Puerto E: habilita los transmisores de ultrasonido para los sensores de posición.

En la Tabla 2.4 se detalla la programación de los pines utilizados del microcontrolador PIC16F877A.

Tabla 2.4 Utilización del microprocesador

PIN	NOMBRE DEL PIN	ENTRADA O SALIDA	MODO DE EMPLEO
1	Reset	Entrada	Para resetear el programa.
2	RA0	Entrada	Posición del brazo derecho.
3	RA1	Entrada	Posición del brazo izquierdo.
4	RA2	Entrada	Posición de la pierna derecha.
5	RA3	Entrada	Posición de la pierna izquierda.
6	RA4	Entrada	No utilizado.
7	RA5	Entrada	No utilizado.
8	RE0	Salida	Habilita los sensores de obstáculos en

			el frente.
9	RE1	Salida	Habilita el sensor de obstáculo en el lado derecho.
10	RE2	Salida	Habilita el sensor de obstáculo en el lado izquierdo.
11	VDD	Entrada	Alimentación.
12	VSS	Tierra	Referencia a tierra.
13	OSC1	Entrada	Oscilador.
14	OSC2	Entrada	Oscilador.
15	RC0	Entrada	Monitorea finales de carrera de los dedos y patas
16	RC1	Salida	No utilizado
17	RC2	Salida	Genera un PWM de 40 Khz. necesario para los sensores de ultrasonido.
18	RC3	Salida	No utilizado
19	RD0	Salida	Dato 1 de la trama de 8 bits
20	RD1	Salida	Dato 2 de la trama de 8 bits
21	RD2	Salida	Dato 3 de la trama de 8 bits
22	RD3	Salida	Dato 4 de la trama de 8 bits
23	RC4	Salida	Activa el latch que maneja los leds de la cabeza.
24	RC5	Salida	Activa el latch de los motores de los brazos y manos
25	RC6	Salida	Activa el latch de los motores de la cintura y de las patas.
26	RC7	Salida	No utilizado
27	RD4	Salida	Dato 5 de la trama de 8 bits
28	RD5	Salida	Dato 6 de la trama de 8 bits
29	RD6	Salida	Dato 7 de la trama de 8 bits
30	RD7	Salida	Dato 8 de la trama de 8 bits
31	VSS	Tierra	Referencia a tierra
32	VDD	Entrada	Alimentación
33	RB0	Entrada	Anuncia obstáculo de cualquier sensor

34	RB1	Entrada	Receptor de los sensores de obstáculos del frente
35	RB2	Entrada	Receptor del sensor de obstáculo derecho
36	RB3	Entrada	Receptor del sensor de obstáculo izquierdo
37	RB4	Entrada	No utilizado
38	RB5	Entrada	Final de carrera de la mano derecha
39	RB6	Entrada	Final de carrera de la mano izquierda
40	RB7	Entrada	Entrada de la señal del control remoto

En la Figura 2.7 se muestra el microcontrolador con los elementos básicos para su funcionamiento.

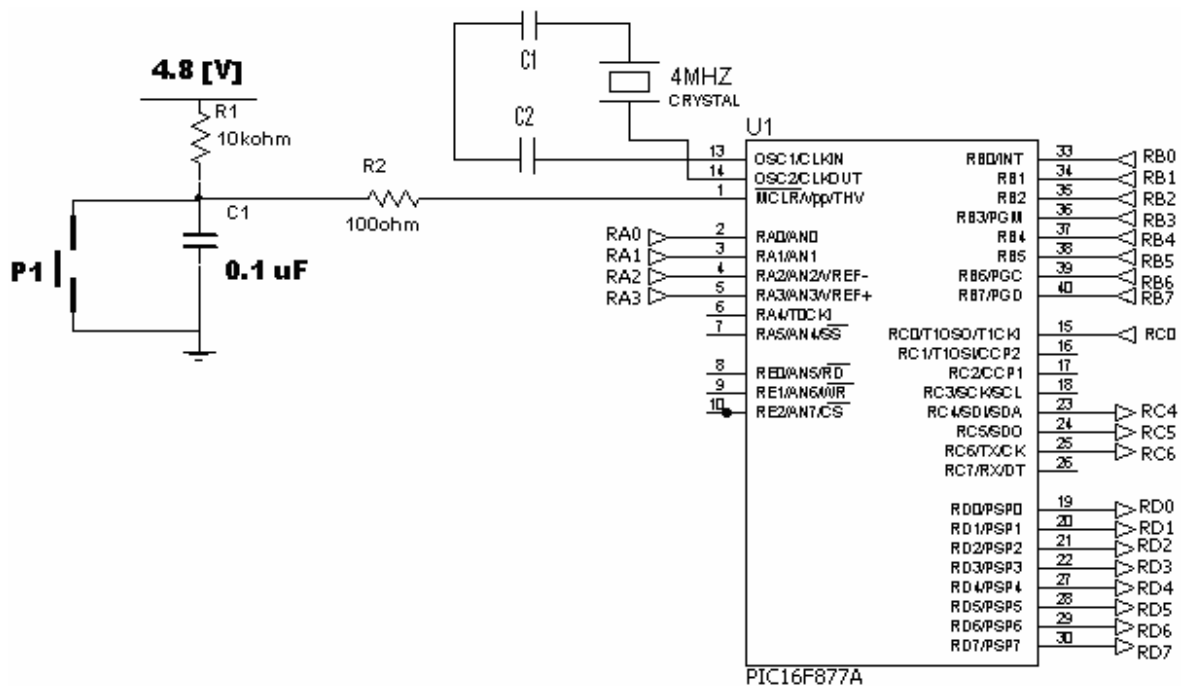


Figura 2.7 PIC 16F877A, reset y cristal

El pulsante conectado al pin1 del microcontrolador sirve para generar el cero lógico necesario para la señal del reset. Para eliminar los rebotes que pueden producirse con la implementación de pulsadores se utiliza el capacitor C1 de un valor de 0.1 [uF] y con el fin de limitar la corriente que entra en el PIC en cada

reset se emplea una resistencia de 10K[ohm] en serie con otra de 100[ohm] que se coloca en cada entrada de los puertos del chip.

Dicha corriente se obtiene a partir de la Fórmula 2.1 [7].

$$I = \frac{V}{R} = \frac{VC2}{R_1 + R_2} \quad \text{Fórmula 2.1}$$

$$I_{reset} = \frac{4.8V}{10100} = 0,4752[mA]$$

Los capacitores que se ponen en los pines 13 y 14 del microcontrolador (ver Figura2.7) son de 22 [pF], valor recomendado por el manual de usuario del PIC16F877A para cristales de 4 [MHz] (ver Anexo B).

2.3.2 RETENCION DE DATOS

El puerto D es utilizado en su totalidad para transmitir los datos necesarios para manejar los siete motores DC y los ocho leds de la cabeza. Como en el proyecto se trabaja con varios elementos que necesitan los datos, fue necesario utilizar un chip que retenga dichos datos para no perder información. Es por ello que se emplearon los latches SN74LS373 cada uno de ellos de 8 bits de entrada (puerto D) y 8 bits de salida. Los latches fueron utilizados según la Tabla 2.5; y habilitados por los pines RC4, RC5 y RC6 del puerto C.

Tabla 2.5 Utilización de los latches

Latch	Bit de Habilitación	Uso
1	RC6	Datos de los motores de las patas y cintura
2	RC5	Datos de los motores de los brazos
3	RC4	Datos de los leds

A la salida del latch se colocan resistencias de 330 [ohm] para que pase la corriente necesaria para encender los leds de la cabeza del robot, los cuales se prenden según cada instrucción que recibe (ver Figura 2.5), es fundamental aclarar que dichos leds se prendían enviado a cero lógico (0V) la salida del latch ya que son de ánodo común.

Para determinar el valor de las resistencias se asume que por los leds circula una corriente de 15 [mA] y utilizando una variación de la Fórmula 2.1 [7] se tiene que:

$$I = \frac{VC2}{R}$$

$$R = \frac{VC2}{I} = \frac{4.8[V]}{15[mA]} = 320[\text{ohm}]$$

Normalizando se tiene:

$$R = 330 [\text{ohm}]$$

Las salidas de los latches 2, 3 se conectaron a los drivers de los motores DC sin ninguna resistencia ya que esto ocasionaría una caída de voltaje, lo cual no permitiría accionar al manejador L293D.

2.4 MANEJO DE LOS MOTORES DC

El robot tiene siete motores dos en sus patas, uno en la cintura, dos en los hombros y dos en las manos, todos de corriente continua.

Para manejar los motores DC se utilizó el circuito integrado L293D, el cual permite usando dos entradas de control realizar la inversión de giro del motor.

El L293D es un driver de 4 canales para manejar dos motores DC con una corriente de hasta 1 [A], trabaja con niveles de voltaje específicos por lo que se alimenta directamente de la fuente VC2, cada L293D puede manejar hasta 2 motores, por lo que en el presente proyecto se utilizaron cuatro L293D.

En la Figura 2.8 se muestra la forma de utilizar el driver para distintos tipos de aplicaciones, entre ellas manejar los dos sentidos de giro de un motor, o sólo un giro de otro motor.

Los pines 1, 9, el pin 8 equivalente a los pines de habilitación y, el pin 16 correspondiente al VC1 están conectados directamente a la fuente de 4.8 [V].

Los pines 2, 6, 10 y 15 que son los pines de las entradas son las salidas de cada uno de los latches (referirse a la Tabla 2.4).

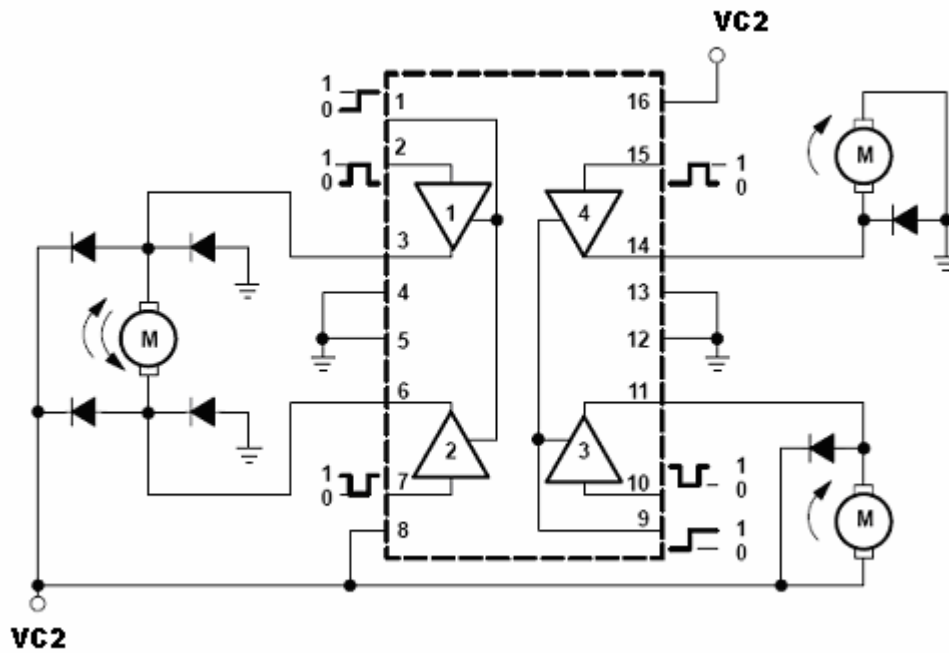


Figura 2.8 Aplicaciones del CI L293D

2.5 SENSORES DE ULTRASONIDO

Para que el robot detecte los objetos durante su movilización se utilizaron sensores de ultrasonido como detectores de obstáculos. El sensor consta de dos partes el emisor y el receptor, el principio de funcionamiento se basa en una señal emitida por el transmisor la cual, al existir un cuerpo externo dentro del alcance provoca que la señal se refleje o de un eco; el mismo que es captado por el receptor.

Los pines 8,9, y 10 correspondientes al puerto E (Figura 2.6) sirven para habilitar a cada uno de los emisores de ultrasonido ya que cada uno de estos se encuentra haciendo una operación AND lógico con el pin RC2 generador de la señal de 40KHz (señal PWM), para el funcionamiento del sensor.

El sensor de ultrasonido utilizado se muestra en la Figura 2.9; dicho sensor consta de un par emisor-receptor, las características serán presentadas más adelante.



Figura 2.9 Sensores de Ultrasonido (Emisor-Receptor)

2.5.1 TRANSMISOR DE ULTRASONIDO

Las características principales del transmisor de ultrasonido son las detalladas en la Tabla 2.6 [8].

Tabla 2.6 Características del Transmisor de Ultrasonido

Característica	Valor
Frecuencia Central	40KHz. \pm 1.0 KHz.
Nivel de presión del sonido	119 dB/40 \pm 1 KHz.
Ancho de Banda	5.0 KHz./100dB
Capacitancia	2000 pF \pm 20%

En la Figura 2.10 en los literales a y b [8] se puede apreciar que para el óptimo funcionamiento del sensor de ultrasonido es necesario ajustar la frecuencia central en un valor de 40 KHz., por lo que fue preciso generar esta señal en el microcontrolador.

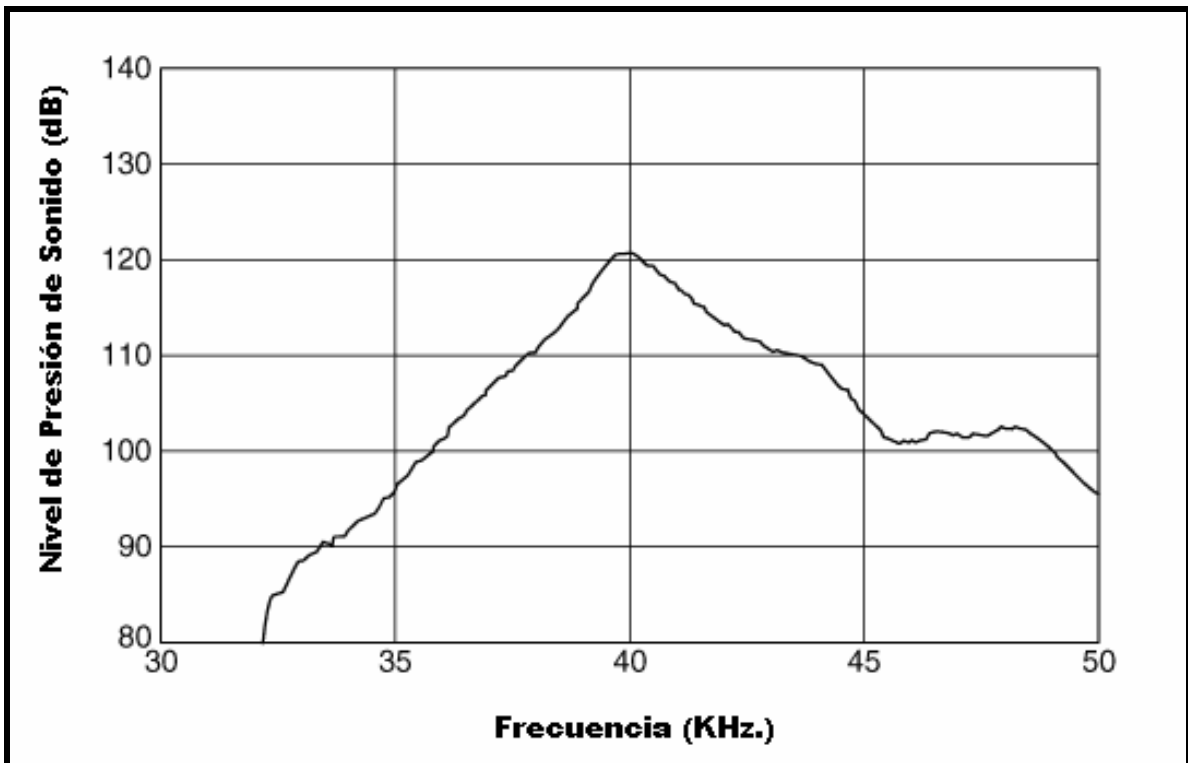


Figura 2.10a. Característica del sensor ante la presión de sonido, tomado de [8]

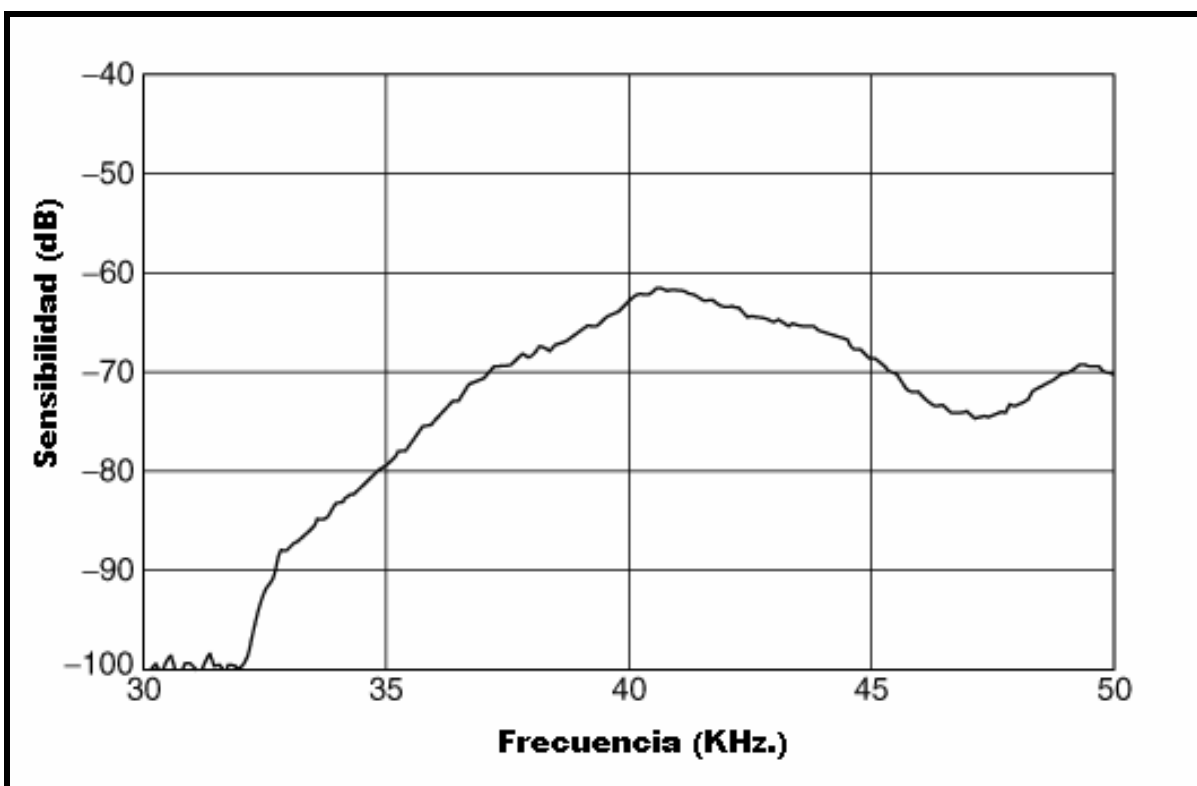


Figura 2.10b. Características de sensibilidad del sensor de Ultrasonido, tomado de [8]

Esta señal se conecta a cada uno de los transmisores a través de una compuerta que realiza la operación AND con cada una de las señales de habilitación (ver Tabla 2.7) con la finalidad de que no funcionen todos los transmisores al mismo tiempo.

A la salida de esta compuerta se colocó una resistencia de 330 ohmios para proteger la base del transistor (Q1 en la Figura 2.11), empleado antes del emisor de ultrasonido.

El transistor se utiliza para amplificar la señal, ya que esta se atenúa al pasar por la compuerta.

En la Figura 2.11 se puede observar el circuito implementado; donde los terminales T1 y T2 son los empalmes del transmisor de ultrasonido y REX corresponde a RE0, RE1 o RE2 según el sensor que se habilite.

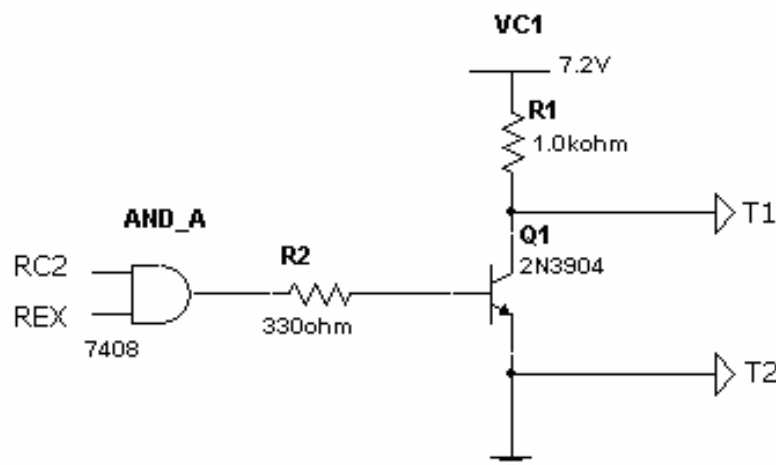


Figura 2.11 Circuito para el Transmisor de Ultrasonido

La Tabla 2.7 muestra el uso de cada pin del puerto E para habilitar el sensor de ultrasonido correspondiente.

Tabla 2.7 Habilitación de los Sensores de Ultrasonido

Pin del Puerto	Funcionalidad
RE0	Habilita los transmisores de obstáculos en el frente del robot.
RE1	Habilita el transmisor de obstáculo en el lado derecho del robot.
RE2	Habilita el transmisor de obstáculo en el lado izquierdo del robot.

2.5.2 RECEPTOR DE ULTRASONIDO

Los receptores utilizados en el presente proyecto presentan las características, detalladas en la Tabla 2.8 [8].

Tabla 2.8 Características del Receptor de Ultrasonido

Característica	Valor
Frecuencia Central	40 KHz. \pm 1.0 KHz.
Sensibilidad Mínima	65 dB/40 \pm 1 KHz.
Ancho de Banda	3.5 KHz./(at-71dB)
Capacitancia	2000 pF \pm 20%

Para el acondicionamiento del receptor se utilizó el detector de tonos LM567N [9], ya que es muy sencillo de implementar (ver Figura 2.13) y fácil de interpretar ya que genera una señal de cero lógico (0_L , pin 8 del integrado de la Figura 2.13) cuando un elemento es detectado dentro del perímetro de actividad del sensor ubicado en el robot.

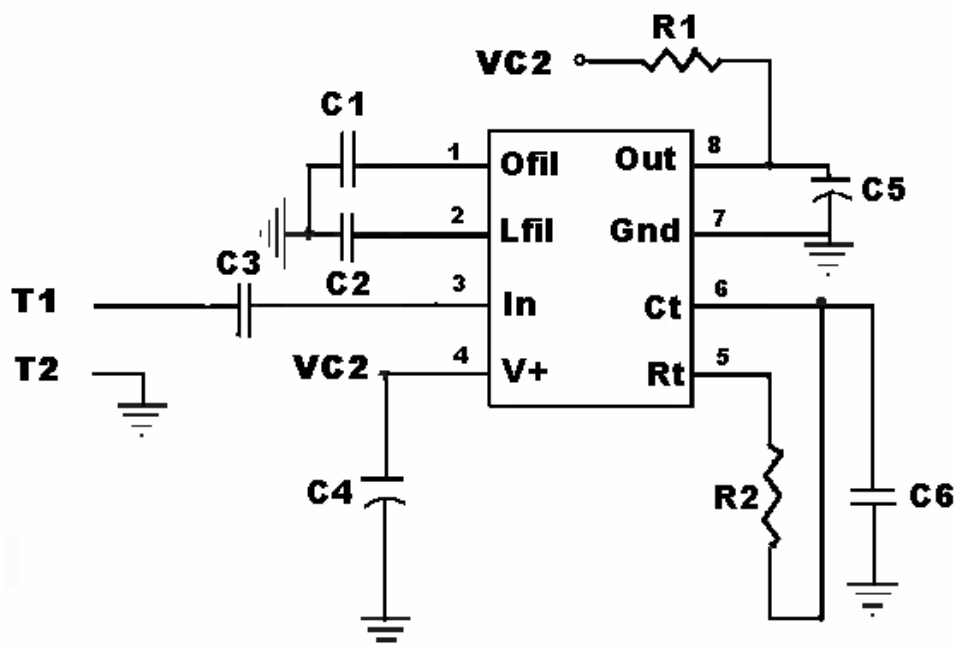


Figura 2.12 Circuito de Acondicionamiento del Receptor de Ultrasonido

Para que el LM567N acondicione la señal emitida por el transmisor de ultrasonido (frecuencia establecida previamente en 40 [Khz]) debe tener los siguientes valores en sus capacitores y resistencias (Figura 2.12).

Los capacitores C1, C2, C3 y C4 tienen un valor ya establecido en las hojas del manual de elemento, por lo que:

$$C1 = 22 \text{ [pF]}$$

$$C2 = 4.7 \text{ [nF]}$$

$$C3 = 0.01 \text{ [uF]}$$

$$C4 = 1.0 \text{ [uF]}$$

La resistencia R1 es la establecida para la carga del circuito de acondicionamiento, las hojas técnicas del elemento establecen un valor de 1K[ohm] pero, luego de las pruebas de funcionamiento se implanta el valor de 3.3 K[ohm] como valor de la resistencia R1 [9].

Todos los valores antes detallados son ya preestablecidos por el fabricante del circuito integrado en el manual para la aplicación [9]; sin embargo, el valor del

capacitor C6 y de la resistencia R2 determinan la frecuencia central de trabajo a partir de la Fórmula 2.2 [9]:

$$f_o = \frac{1}{1.1 * R2 * C6} = \text{frecuencia central} \quad \text{Fórmula 2.2}$$

Como la frecuencia central es de 40Khz. y asumiendo un valor de 0.01 uF para el capacitor C6, se puede calcular la resistencia R2, a partir de:

$$R2 = \frac{1}{1.1 * f_o * C6} = \frac{1}{1.1 * 40000Hz * 0.01 * 10^{-6} F} = 2272.72[ohm]$$

Cuyo valor normalizado con una tolerancia del 10% es de 2.2K[ohm].

$$R2 = 2.2K[ohm]$$

El capacitor C5 de 10 uF se utiliza para eliminar la señal que se genera en el receptor a causa de la emisión del ultrasonido cerca de él [9].

$$C5 = 10 \text{ uF}$$

Este acondicionamiento de señal al trabajar con la frecuencia a detectar no permite modificar la distancia de detección del obstáculo, por lo que para modificar este parámetro es necesario disminuir la potencia entregada al sensor.

Dentro de la plataforma robótica utilizada la ubicación de estos sensores de ultrasonido para detectar obstáculos mediante el sistema de eco se pueden observar en la Figura 2.13

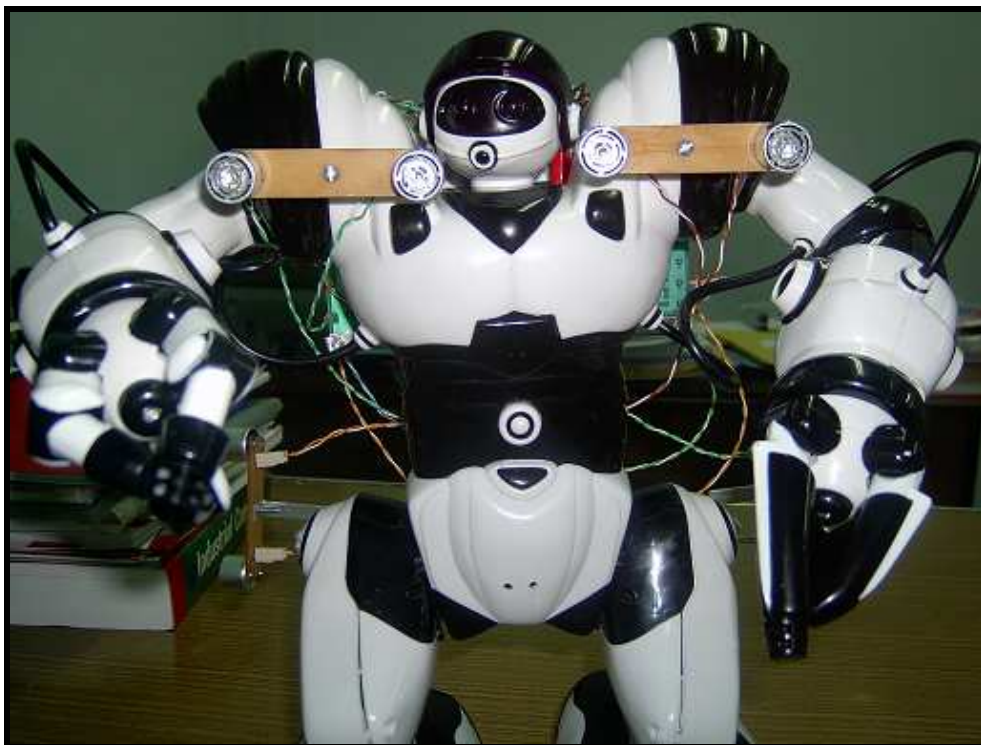


Figura 1.13 Ubicación de los sensores de ultrasonido

2.6 FINALES DE CARRERA

En el robot existen seis finales de carrera ubicados como se muestra en la Tabla 2.9 y cuyo diagrama se puede apreciar en la Figura 2.14. Su operación se basa en un contacto normalmente abierto que al ser activado manda una señal de continuidad en los dos terminales de empalme.

Tabla 2.9 Función y ubicación de los finales de carrera

Final de Carrera	Ubicación	Funcionalidad
Final de carrera1	Dedo Derecho	Todos detectan el obstáculo y detienen el robot.
Final de carrera2	Dedo Izquierdo	
Final de carrera3	Al frente de la pata Derecha	
Final de carrera4	Atrás de la pata Derecha	
Final de carrera5	Al frente de la pata Izquierda	
Final de carrera6	Atrás de la pata Izquierda	

Dicha señal que se obtiene de estos finales de carrera entran en el puerto RC0 adicionalmente cuando cualquiera de ellos es presionado se genera una señal en el pin RB0 con la finalidad de realizar una interrupción externa en el microcontrolador.

Esta interrupción es necesaria para informar al robot que un obstáculo ha sido detectado.

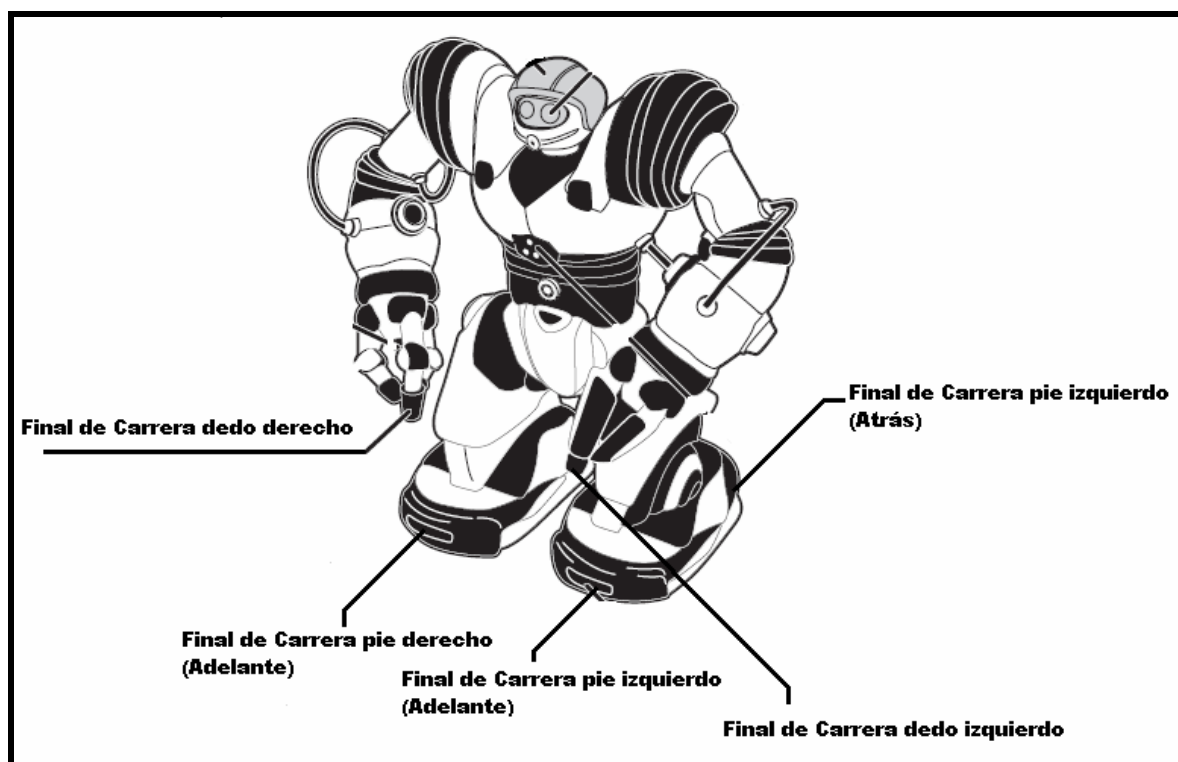


Figura 2.14 Ubicación de finales de carrera del robot

2.7 SENSORES DE POSICIÓN

Para detectar la posición de cada uno de los ejes de los motores se utilizó, acoplados al mismo, potenciómetros, los cuales están conectados al VC2 (4.8 [V]) para generar diferentes valores de voltaje, los mismos que ingresan al microcontrolador a través de los canales analógicos AN0, AN1, AN2 y AN3.

El circuito de los sensores de posición se presenta en la Figura 2.15 (es similar el circuito para todos), donde a partir de cada salida se conecta una resistencia de 100 [ohm] para proteger al microcontrolador.

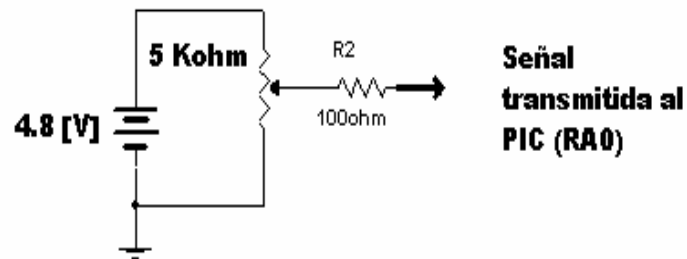


Figura 2.15 Circuito utilizado como sensor de posición

Los potenciómetros utilizados en el presente proyecto son lineales, los cuales se mueven con el eje del engrane de cada pata y brazo. Los potenciómetros utilizados se pueden observar en la Figura 2.16; de este tipo fueron utilizados cuatro.

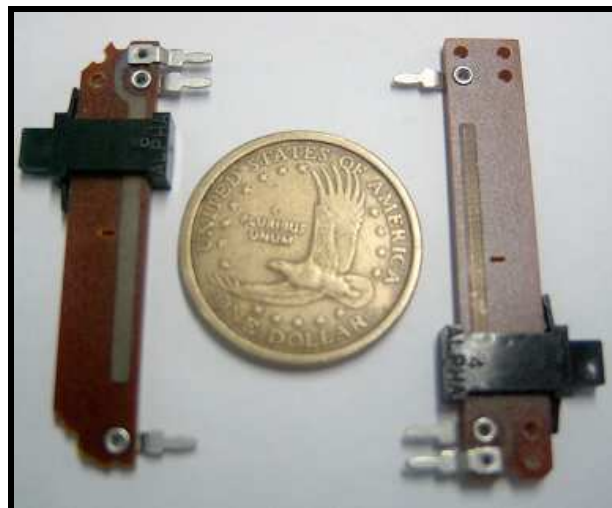


Figura 2.16 Potenciómetros lineales utilizados como sensores de posición

En el capítulo tres se detalla el software necesario para el funcionamiento de la plataforma bípeda, diagramas de flujo y generalidades del programa.

CAPITULO 3

DESARROLLO DEL PROGRAMA DE CONTROL

El programa de control se lo realizó en lenguaje ensamblador para microcontroladores PIC; y esta basado en la arquitectura de la Figura 3.1 en la cual se puede ver que el microcontrolador interactúa con los sensores actuadores y control remoto, el cual emite una trama de datos que es recibida y procesada, luego este comanda a los actuadotes para que el robot realice diferentes secuencias de movimiento.

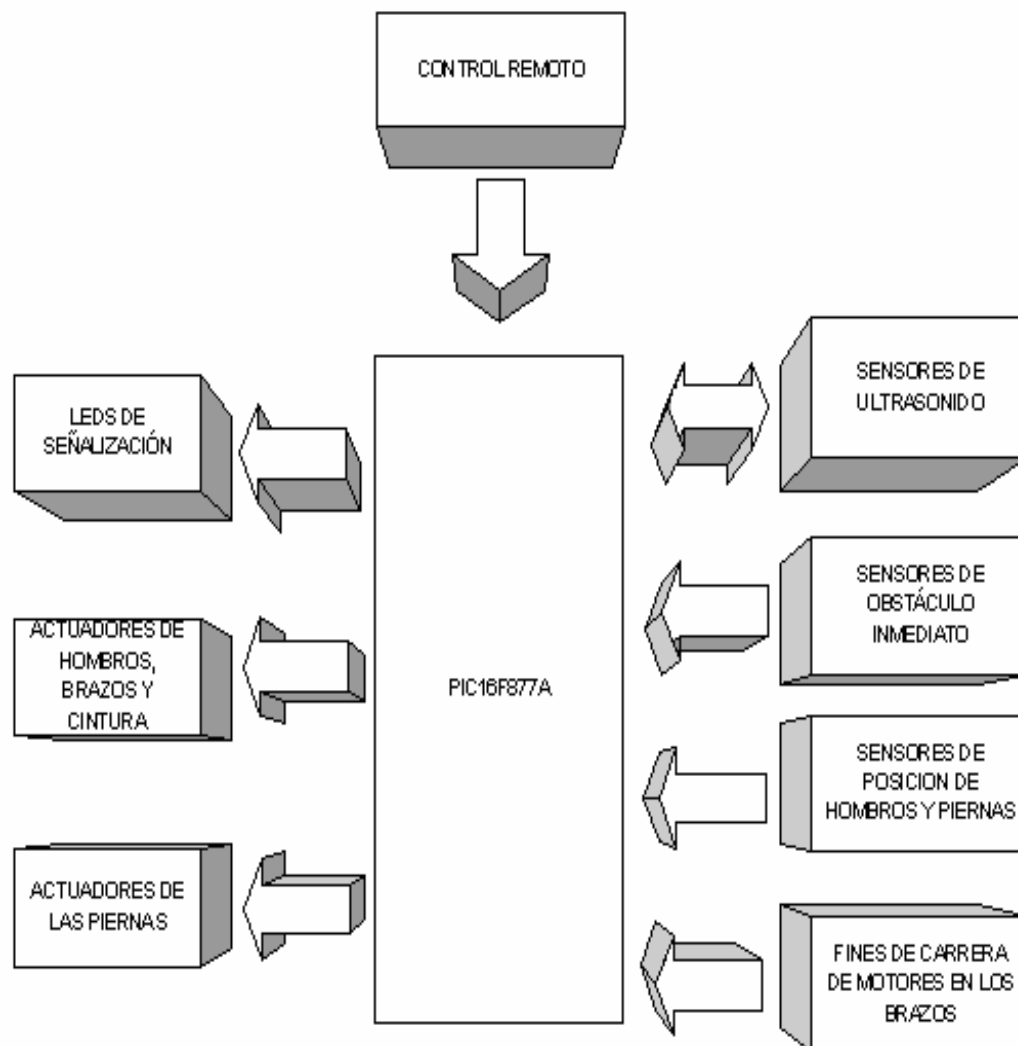


Figura 3.1 Arquitectura del Programa

3.1 PROGRAMA PRINCIPAL

El programa principal se encarga de configurar al microcontrolador llamando a siete subrutinas distintas, las mismas que se resumen a continuación:

Tabla 3.1 Subrutinas de Programa Principal

Subrutina	Función
PUERTOS	Configura a los puertos PORTB, PORTA como entradas y PORTE, PORTC, PORTD como salidas, por medio de los TRISX (donde X=A, B, C, D, E), todas estas digitales a excepción de PORTA que es análogo.
ANALOG	Configura al conversor A/D (ADCON1) para utilizar la fuente del PIC como Voltaje de referencia y el número de canales analógicos hace que el resultado de la conversión se deposite en el ADRESH perdiendo los dos bits menos significativos de la conversión, finalmente deja encendido el conversor A/D.
VARIABLES	Prepara los valores iniciales de cada registro auxiliar, borra el contenido de los registros AUXD, AUXD1, AUXT, AUXB, AUXMA, PORTC, PORTE, MOTOR A, MOTOR B, setea AUXT_1A=D'72' AUXT_1B= D'08' LEDS=B'11110101', finalmente se debe llamar a la subrutina refresh para actualizar estos cambios en los latch.
PWMFUN	Configura el registro CCP2CON para que el timer 2 funcione en modo PWM
INTERRUPCIONES	Al inicio se habilita solo ciertas interrupciones configurando al INTCON y al PIE1. Las interrupciones permitidas son: Internas.- Por desbordamiento del TIMER1 (PIE1.TMR1IE), Conversión A/D completa (PIE1.ADIE). Externas.- Cambio de estado en los MSB del PORTB RBU (INTCON.RBIE), interrupción en RBO (INTCON.INTE). Además se borra todas las banderas, y se habilita el funcionamiento de las interrupciones tanto periféricas (INTCON.PEIE) como globales (INTCON.GIE).
INTERRUPCIONES	Al inicio se habilita solo ciertas interrupciones configurando al INTCON y al PIE1. Las interrupciones permitidas son: Internas.- Por desbordamiento del TIMER1 (PIE1.TMR1IE). Conversión A/D completa (PIE1.ADIE). Externas.- Cambio de estado en los MSB del PORTB RBU (INTCON.RBIE), interrupción en RBO (INTCON.INTE). Además se borra todas las banderas, y se habilita el funcionamiento de las interrupciones tanto periféricas (INTCON.PEIE) como globales (INTCON.GIE).
WAKE UP	Esta subrutina se explicara más adelante

Tabla 3.1 Continuación Subrutinas de Programa Principal

Timers	Configura los timers del microcontrolador,		
	Timer	Registros configurados	Función
	0	OPTION_REG, TMRO	Se utiliza para obtener bases de tiempo de 2.2m[s] y 1 m[s] dependiendo el caso, necesaria para la recepción de los datos enviados desde el control remoto.
	1	T1CON, TMR1L, TMR1H	Por medio de este timer, se crea una base de tiempo de 8.6m[s] que sirve para cronometrar distintos tiempos desde 0.5 [s], hasta 5 [min.].
	2	T2CON, TMR2, PR2, CCPR2L, CCPR2H	Genera una señal PWM de 40[Khz.] que sale por el pin RC2, la cual es utilizada para los emisores de ultrasonido.

Luego de completar todas las subrutinas del programa principal el microcontrolador permanece en un lazo que se denomina principal (LAZO_PRINCIPAL) en la espera de la llegada de un dato o la orden de realizar la conversión A/D. En el primer caso si la bandera de llegada de dato AUXT,6=1 se produce un salto a **ESCOGE _ OPCION**; en cambio para el segundo caso en la cual la bandera de auxiliar de conversión AUXT,3=1, se genera un retardo y luego del mismo se setea la bandera de interrupción de conversión A/D (PIR1.ADIF=1) propia del microprocesador produciendo una interrupción a **CONVERSION**.

3.2 SUBRUTINAS

A continuación se detallaran las subrutinas utilizadas fuera del programa principal:

3.2.1 SUBRUTINA UNO, DOS, SEIS y SIETE

Estas subrutinas permiten manejar el movimiento en los hombros sentido horario u antihorario. Para encender sus actuadores se utiliza el registro *MOTOR_A*; dependiendo de la posición inicial del brazo a comandar, que se la

obtiene llamando a la subrutina **ANALOG** (previamente se debe setear el canal de conversión correspondiente al sensor del hombro en *ADCON0*).

En caso de que el brazo suba, si su posición es menor a la altura del hombro, este podrá moverse solamente hasta dicha altura, caso contrario logrará subir hasta la posición máxima. Por otra parte si el brazo esta bajando y su colocación es mayor a la altura del hombro este alcanzará moverse hasta esta altura, caso contrario conseguirá bajar hasta la posición mínima.

Cuando ya se determinó hasta que posición se desea llegar se guarda esta información en el auxiliar de posición sea *AUXCONV1* o *AUXCONV2* y finalmente se produce un salto a **PRIORIDAD** para determinar el nivel de prioridad en la conversión con *AUXP*.

Tabla 3.2 Resumen de Subrutinas Uno, Dos, Seis y Siete

SUBROUTINA	UNO	DOS	SEIS	SIETE
MOTOR_A	B'10XXXXXX'	B'01XXXXXX'	B'XX10XXXX'	B'XX01XXXX'
MOVIMIENTO (Visto de frente)	Antihorario (sube)	Horario (baja)	Horario (sube)	Antihorario (baja)
HOMBRO	Derecho		Izquierdo	
Auxiliar de posición	AUXCONV1		AUXCONV2	
Canal A/D	0		1	
ADCON0	B'XX000XXX'		B'XX001XXX'	
AUXP	B'1XXX0001'		B'1XXX0010'	

X= valor que no se modifica

3.2.2 SUBROUTINA ANALOG

En esta subrutina se inicia la conversión A/D seteando el bit *ADCON0,GO/DONE*, a continuación se llama a la SUBROUTINA **REFRESH**, al regresar de la subrutina se espera hasta que la conversión este completa y finalmente se devuelve el valor obtenido en el *ADRESH* por medio del acumulador *W*.

SUBROUTINA TRES, CUATRO, OCHO y NUEVE

Este grupo de subrutinas tienen el mismo principio de funcionamiento para lograr el movimiento de los brazos, abriendo y cerrando las manos. Básicamente se enciende el actuador manejado por el registro *MOTOR_A* con lo que se logra un giro horario o antihorario, luego se configura al

microcontrolador para que mantenga encendidos los brazos 1,5 [s] aproximadamente o hasta que su posición llegue al fin de carrera.

Los registros que permiten mantener encendido el motor determinado tiempo son: *AUXT_1M3* y *AUXT_1M4* cada uno asignado a un brazo, los mismos que deben ser configurados con un valor igual a D'3' ya que el Timer1 se encuentra encargado de generar bases de tiempo de 0,5[s]; en cada desborde se decrementan una unidad, y cuando lleguen a cero los registros se ordena detener el brazo. Por otra parte existen dos banderas para saber que la muñeca se encuentra en movimiento, estas banderas y registros serán vistas en la Tabla 3.3.

Tabla 3.3 Resumen de Subrutinas Tres, Cuatro, Ocho y Nueve

SUBRUTINAS	TRES	CUATRO	OCHO	NUEVE
MOTOR_A	B'XXXX10XX'	B'XXXX01XX'	B'XXXXXX10'	B'XXXXXX01'
MOVIMIENTO (Visto de frente)	Antihorario (Abriéndose)	Horario (Cerrándose)	Horario (Abriéndose)	Antihorario (Cerrándose)
MUÑECA	Derecha		Izquierda	
AUXILIAR TIMER 1	AUXT_1M3		AUXT_1M4	
BANDERA DE TIMER 1	AUXB,6		AUXB,5	
BANDERA DE MOVIMIENTO DE MANO	AUXT,1		AUXB,7	

X= valor que no se modifica

3.2.4 SUBRUTINA CINCO Y DIEZ

El objetivo de estas subrutinas es lograr el movimiento de la cintura, su funcionamiento es similar a las subrutinas TRES, CUATRO, OCHO y NUEVE, pero los registros utilizados cambian, por lo cual se presenta directamente su resumen en la Tabla 3.4.

Tabla 3.4 Resumen de Subrutinas Cinco y Diez

SUBROUTINAS	CINCO	DIEZ
MOTOR_B	B'10XXXXXX'	B'01XXXXXX'
MOVIMIENTO (Visto de frente)	Antihorario (Derecha)	Horario (Izquierda)
AUXILIAR TIMER 1	AUXMA,3	
BANDERA DE TIMER 1	AUXB,6	
BANDERA DE MOVIMIENTO DE CINTURA	AUXT,2	

X= valor que no se modifica

3.2.5 SUBROUTINA C_SLEEP

Esta subrutina puede ser llamada por dos eventos: el primero, si el usuario selecciona este comando; y el segundo se efectúa siempre que no se haya recibido ninguna orden los últimos cinco minutos; colocando al robot en modo sleep (dormido). Es decir no toma en cuenta ninguna orden del usuario a menos que se lo despierte con el comando Wake_UP, para esto se setea la bandera AUXB,4 de desborde de 5 minutos. Adicionalmente se efectúa un movimiento simultaneo en los brazos hacia abajo utilizando las subrutinas DOS y SIETE, posteriormente se configura para que se muevan a su posición mínima en los registros de posición AUXCONV1 y AUXCONV2. (Ver subrutinas DOS y SIETE)

3.2.6 SUBROUTINA WAKE_UP

La subrutina WAKE_UP es la única que se puede elegir cuando el robot se encuentra en modo Sleep, esta se encarga de despertar al robot borrando la bandera AUXB,4; y además realiza el movimiento de subir los brazos hacia arriba, llamando a las subrutinas UNO y SEIS. Los brazos deberán llegar hasta la posición máxima, por lo que se deben configurar nuevamente los registros AUXCONV1 y AUXCONV2

3.2.7 SUBROUTINA REFRESH

En Refresh se refrescan los datos actuales de los registros MOTOR_A, MOTOR_B y LEDS en cada uno de los Latch. Para esto se sigue el siguiente orden:

Se habilita el latch 1 con RC6, luego se escribe el dato de MOTOR_A en el PORTD, se espera un tiempo de 30u[s] y luego se deshabilita el latch. Para los latch siguientes se repite el mismo procedimiento, tomando en cuenta los registros mostrados en la Tabla 3.5:

Tabla 3.5 Resumen Refresh

Latch	Bit Habilita/Deshabilita	Dato a escribirse en el Latch
1	RC6	MOTOR_A
2	RC5	MOTOR_B
3	RC4	LEDS

3.2.8 SUBROUTINA NO_REFRESH

Esta subrutina evita refrescar los datos en los latch, hasta que las dos piernas alcancen una posición previamente configurada; para lograr esto, al iniciar la subrutina se setea AUXB,0=1 (bandera que no permite refrescar los datos), y se espera hasta que los todos los bits más significativos de MOTOR_B sean igual a cero, al cumplirse esta condición se borra el registro MOTOR_B y la bandera AUXB,0.

3.2.9 SUBROUTINA TRECE

La subrutina trece hace que el robot se detenga por completo, es decir es un comando de PARADA (STOP). La parada puede ocurrir debido a dos causas: el robot detecto un obstáculo o el usuario emitió este comando.

3.2.10 SUBROUTINAS ONCE, DOCE, CATORCE Y QUINCE

Estas subrutinas generan ordenes o conjunto de secuencias necesarias para que el robot camine; la dirección creada es distinta y en cada caso son: caminar hacia adelante, girar a la derecha, virar a la izquierda, caminar para atrás; estas secuencias necesitan de tres movimientos: el de la cintura y los de

cada pierna; por lo que es necesario llamar a las subrutinas convenientes (A_P_D_U, A_P_D_D, A_P_I_U, A_P_I_D), las mismas que tiene un funcionamiento similar a las subrutinas uno, dos, seis y siete que permiten el movimiento de los hombros, pero en este caso se pretende controlar las piernas.

Tabla 3.6 Resumen de Subrutinas A_P_D_U, A_P_D_D, A_P_I_U, A_P_I_D

SUBROUTINA	A_P_D_U	A_P_D_D	A_P_I_U	A_P_I_D
MOTOR_B	B'10XXXXXX'	B'01XXXXXX'	B'XX10XXXX'	B'XX01XXXX'
MOVIMIENTO (Vista lateral)	Antihorario (adelante)	Horario (atrás)	Horario (adelante)	Antihorario (atrás)
PIERNA	Derecho		Izquierdo	
Auxiliar de posición	AUXCONV3		AUXCONV4	
Canal A/D	2		3	
ADCON0	B'XX010XXX'		B'XX011XXX'	
AUXP	B'1XXX0100'		B'1XXX1000'	

X= valor que no se modifica

3.2.11 SUBROUTINA ROTA

Esta subrutina es utilizada en la comunicación, y permite saber la posición del bit que esta llegando en ese instante por medio de *AUXD1*. Como su nombre lo indica esta subrutina se encarga de rotar a la derecha el registro *AUXD1* con el carry previamente seteado. Luego de realizar esta acción, se revisa si el carry es igual a cero con lo que se finaliza la subrutina; el carry seteado indica que los bits del registro *AUXD1* han sido rotados ocho veces por lo que se debe finalizar la comunicación, entonces se borra la bandera de comunicación *AUXT,7*. Luego se guarda la orden recibida en *DATO* (*DATO=AUXD*) y se verifica si el comando recibido es el de **WAKE_UP** o el de **STOP**.

Para la orden **WAKE_UP** se borra el bit del modo **SLEEP** (*AUXB,4=0*) para despertar al robot, recordando que solo este comando puede ser utilizado para esta tarea. Por otra parte si es el comando **STOP** se debe setear la bandera de **STOP** (*AUXT,5=1*), para que cualquier proceso se detenga.

Finalmente se anuncia la llegada de dato con la bandera *AUXT,6=1* y se termina la subrutina.

3.2.12 SUBROUTINA (MODO _ AUTÓNOMO)

En el modo autónomo el robot habilita el sensor de ultrasonido delantero ($RE0=1$), y sube la bandera de tipo de movimiento ($AUXMA,0=1$), para indicar la autonomía del robot, posteriormente se llama a la subrutina **ONCE** (Robot camina hacia adelante); al presentarse un obstáculo, se genera la interrupción de **SENSORES**, en la cual se obtiene la información necesaria para cambiar de dirección al robot, por lo que es de importancia realizar un determinado número de llamadas a la subrutina DOCE o CATORCE (girando al robot); cuando el robot posea la posición de deseada, se volverá al inicio de este proceso.

3.3 INTERRUPCIONES

Al producirse una interrupción automáticamente el microcontrolador deja lo que se encuentra realizando ese momento, salta a la dirección 0X04h (*Etiqueta INTER*), guarda el contenido del acumulador *W* en el registro auxiliar *W_AUX*, y empieza a realizar un testeo de cada bandera de interrupción, al encontrar el BIT seteado se producirá un salto a la etiqueta correspondiente, el orden en la Tabla 3.7 refleja la prioridad que se le dio a la interrupción.

Tabla 3.7 Etiquetas de Interrupciones y su Prioridad

BANDERA DE INTERRUPCIÓN	ETIQUETA DE LLEGADA
INTCON,0 (RBIF)	INT_RBS
INTCON,2 (TOIF)	INT_TIMER0
INTCON,1 (INTF)	SENSORES
PIR1,6 (ADIF)	CONVERSION
PIR1,0 (TMR1IF)	INT_TIMER1

3.3.1 INT_RBS

Al ingresar en esta opción se puede tener tres posibilidades las mismas que son:

1. Se activo el fin de carrera del motor de la mano derecha (RB5),
2. Se activo el fin de carrera del motor de la mano izquierda (RB6) o
3. El usuario esta enviando un dato (RB7).

3.3.1.1 Se activo el fin de carrera del motor de la mano derecha o izquierda

El propósito de estas opciones es detener el movimiento de la mano, en la Tabla 3.8 se observa las condiciones y acciones necesarias para que se cumpla este objetivo.

Tabla 3.8 Resumen de Activado de los Fines de Carrera en las Manos.

MANO	DERECHA	IZQUIERDA
CONDICIONES	AUXT,1=1 Pin RB5=0	AUXB,7=1 Pin RB6=0
ACCIONES MOTOR_A igual a	B'XXXX00XX'	B'XXXXXX00'

X= valor que no se modifica

3.3.1.2 El usuario esta enviando un dato (Busque_RB7)

Para atender esta interrupción y obtener el dato, es necesario, conocer la trama que se va a recibir. La trama comienza con un bit de inicio en bajo que dura aproximadamente 3,2m[s], luego se produce cambios de alto a bajo sucesivamente hasta completar 8 periodos completos, que se relacionan directamente con los 8 bits que pertenecen al dato recibido, el tiempo en bajo T.B.=1,1m[s] es igual en cada periodo; pero el tiempo en alto T.A. es variable y determina el estado lógico del bit, si T.A.=2.2m[s] el bit es uno lógico, pero si T.B.=1m[s] el bit es cero. En la Tabla 3.9 se observa la trama de datos enviada por el control remoto; además, para conocer el funcionamiento de esta parte del programa se debe revisar el diagrama de flujo detallado de **BUSQUE_RB7** (Figura 3.2).

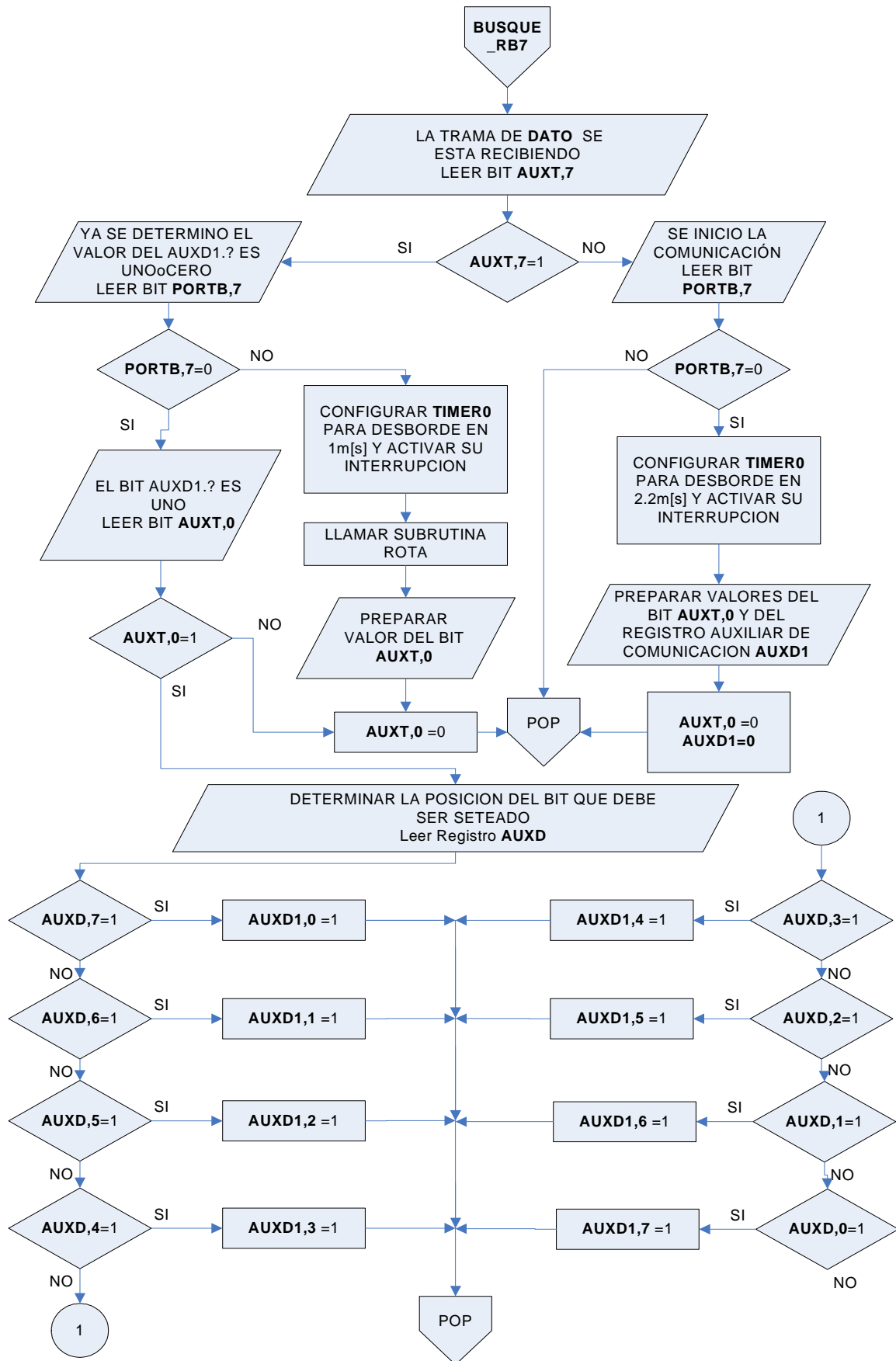



Figura 3.2 Diagrama de Flujo Detallado de BUSQUE_RB7

Tabla 3.9 Trama Enviada por el Control Remoto



T.A. [ms]	0	2.2	1	1	2.2	1	2.2	2.2	2.2	
T.B [ms]	6.8	1								
	Bit inicio	B7	B6	B5	B4	B3	B2	B1	B0	FIN
		BYTE								
DATO		1	0	0	1	0	1	1	1	

3.3.2 INT_TIMER0

Este timer se lo utiliza como generador de bases de tiempo para la comunicación. Al principio, al detectar el flanco de bajada en el bit de inicio, se activa la interrupción del timer0 (INTCON, TOIE=1) y se configura para que se desborde en 2.2m[s]; al generarse la interrupción, se setea (**AUXT,0=1**) bandera de desborde de timer por 1era. vez desde su última configuración, a partir de este momento puede ocurrir dos eventos: el cambio de estado del bit RB7 (flanco positivo) o el desborde por segunda vez del timer desde la última vez que se configuró, en este caso la comunicación falla, es cancelada y la ininterrupción del timer0 es deshabilitada. En cambio si sucede el primer evento, quiere decir que se va a recibir el dato, por lo tanto se borra el bit **AUXT,0=0** y se activa el bit de inicio de comunicación (AUXT,7=1), luego se vuelve a configurar el desborde del timer0 pero esta vez para un tiempo de 1,2m [s], si el timer se desborda (entra a la interrupción) antes que el pin RB7 cambie de estado (flanco negativo) se setea (**AUXT,0=1**), y con esto se da a conocer el estado lógico actual del bit leído. Este proceso se inicia cada vez que RB7 detecta un flanco positivo y solamente cuando se encuentra en la recepción de bits (B7, B6,, B0). Finalmente luego de atender cada interrupción se realiza un salto a POP para reestablecer el valor original del acumulador W antes que se atienda esta interrupción.

Tabla 3.10 Registros Utilizados en la Recepción de Datos

REGISTRO	FUNCION
DATO	Guardar el comando recibido al final de la comunicación
AUXT,7	Bandera de comunicación
AUXT,0	Bandera de desborde de timer por 1era. Vez
AUXT,6	Bandera que anuncia llegada del dato
AUXD1	Determina la posición en el registro AUXD del bit que se esta recibiendo
AUXD1	Registro en el que se almacenan los bits de llegada

3.3.3 SENSORES (Interrupción debido a obstáculos)

Se tiene dos opciones que dependen de la bandera de movimiento *AUXMA,0*:

Tabla 3.11 Opciones de Movimiento

AUXMA,0	OPCION
0	MANUAL
1	AUTONOMA

3.3.3.1 MANUAL

Al activarse cualquiera de los sensores de obstáculo (fines de carrera ubicados en las patas y manos) pin RC0=0, se procede a simular la llegada del comando **STOP** y se alza la bandera *AUXT,5* para que el robot se detenga. Por último se genera un salto a POP que devuelve el valor que tenía el acumulador *W* antes de ingresar a la interrupción.

3.3.3.2 AUTONOMA

Esta opción es habilitada solamente cuando el robot se encuentra en modo autónomo y se produce al detectar un obstáculo por medio de los sensores de ultrasonido delanteros. Al ingresar a la interrupción se activa la bandera de obstáculo al frente (*AUXMA,0=1*), se detiene al robot borrando los registros *MOTOR_A*, *MOTOR_B* y llamando a la subrutina **REFRESH**. Luego se apaga el sensor de ultrasonido delantero (*PIN RE0=0*) y se genera un retardo mayor al tiempo de vuelo máximo, seguidamente se encienden los sensores de ultrasonido laterales (*PINES RE1=RE2=1*) para detectar algún objeto a la

derecha o izquierda del robot y nuevamente se genera el mismo tiempo de retardo. A continuación se realiza un barrido de los pines *RB1* y *RB2* para indicar en que sentido el robot puede girar (Bandera de dirección *AUXMA,2*) y cuantos veces (*NUM_GIRO*) debe llamar a la subrutinas **DOCE** (*Auxma,2=0*) o **CATORCE** (*Auxma,2=1*), antes de continuar caminando.

Tabla 3.12 Resumen de Acciones para el Barrido de RE1 y RE2.

PIN RE1	PIN RE2	Obstáculo Lateral D o I	Bandera AUXMA,2	NUM_GIRO	Dirección de Giro
0	0	No	0	25	DERECHA 90°
0	1	Izquierda	0	25	DERECHA 90°
1	0	Derecha	1	25	IZQUIERDA 90°
1	1	I y D	0	50	DERECHO 180°

Al terminar se borra la bandera de interrupción por RBO (*INTCON,RBOF=0*) y se produce un salto a POP que devuelve el valor original del acumulador antes de entrar a la interrupción.

3.3.4 INT_TIMER1

El timer1 es el encargado de generar una base de tiempo de 0,5 segundos cada vez que se produce la interrupción y con la ayuda de registros auxiliares es posible que se generen cinco alternativas:

1. Son cinco minutos desde la última orden dada por el usuario
2. A pasado 1,5[s] desde que se inicio el movimiento en la mano derecha
3. A pasado 1,5[s] desde que se inicio el movimiento en la mano izquierda
4. Son 1,5[s] a partir que se inicio el movimiento cintura
5. El comando STOP fue llamado hace 1,5[s]

A continuación se resumen estas opciones en la Tabla 3.13

Tabla 3.13 Resumen para las Opciones del Timer1

OPCION	ACCION	AUXILIAR DE CONTEO (Valor inicial)	BANDERA AUXILIAR	¿Se cumplió la opción ? Si, entonces:
1	Simular la llegada del comando Sleep	$AUXT_1A=D'72'$ $AUXT_1B=D'08'$	$AUXB,4$	$DATO=$ $B'10100011'$
2	Apagar el motor de la mano derecha	$AUXT_1M3=$ $D'03'$	$AUXB,6$	$MOTOR_A=$ $B'XXXX00XX'$
3	Apagar el motor de la mano izquierda	$AUXT_1M4=$ $D'03'$	$AUXB,5$	$MOTOR_A=$ $B'XXXXXX00'$
4	Apagar el motor de la cintura	$AUXT_1M5=$ $D'03'$	$AUXMA,3$	$MOTOR_B=$ $B'XXXX00XX'$
5	Colocar los leds en estado de reposo	$AUXT_1=$ $D'03'$	$AUXB,3$	$LEDS=$ $B'11110101'$

Las banderas sirven para avisar al microcontrolador que se debe restar una vez al auxiliar de conteo correspondiente, cuando dicho auxiliar llegue a cero se deberá realizar la acción oportuna, luego se debe refrescar los datos en los latch por lo que hay que llamar a la subrutina **REFRESH** y finalmente realizar un salto a POP la misma que restituye el valor del acumulador W antes de ingresar a esta interrupción.

3.3.5 CONVERSION

Como su nombre lo indica esta parte del programa se encarga de conocer el estado de los sensores de posición realizando una conversión A/D. Para efectuar esto se debe conocer la prioridad en la cual los sensores deben ser leídos, pero antes es preciso comprobar que el registro $PRIOR_1$ este lleno ($PRIOR_1,7=1$), si este bit esta vacío se procede a subir un nivel a todas las prioridades y nuevamente se testa el bit antes mencionado, si nuevamente esta vacío se borra la bandera de $AUXT,3=0$ auxiliar de conversión y se finaliza la interrupción con un salto a POP devolviendo al acumulador W su valor antes de atender la interrupción. Para la primera opción cuando $PRIOR_1,7=1$, se realiza un barrido de los bits bajos de $PRIOR_1$ (tomando en cuenta el byte de prioridad presentado en la subrutina prioridad), dependiendo de el caso se procederá a seleccionar el canal A/D deseado a través del $ADCON0$ y para obtener su estado se llama a la subrutina **ANALOG**, al alcanzar la posición deseada (dada por los registros $AUXCONVG$ donde $G=1,2,3y4$) se debe

apagar el actuador correspondiente para detener el movimiento (utilizando *MOTOR_A* o *MOTOR_B*), además de llamar a la subrutina **REFRESH**. Si el registro de *PRIOR_1* ha sido satisfecho se borra el BIT *PRIOR_1,7=0* y se finaliza la interrupción con un salto a POP que restaura el valor que inicialmente tenía el acumulador antes de atender esta interrupción.

Tabla 3.18 Resumen para el Barrido de *PRIOR_1*

BIT SETEADO	<i>PRIOR_1,3</i>	<i>PRIOR_1,2</i>	<i>PRIOR_1,1</i>	<i>PRIOR_1,0</i>
Actuador	Pierna Izquierda	Pierna Derecha	Hombro Izquierdo	Hombro Derecho
Canal A/D	3	2	1	0
ADCON igual	B'XX011XXX'	B'XX010XXX'	B'XX001XXX'	B'XX000XXX'
Auxiliar de posición	AUXCONV4	AUXCONV3	AUXCONV2	AUXCONV1
¿Es la posición deseada? Si, entonces:	<i>MOTOR_B</i> = B'XX00XXXX'	<i>MOTOR_B</i> = B'XXXX00XX'	<i>MOTOR_A</i> = B'XX00XXXX'	<i>MOTOR_A</i> = B'00XXXXXX'

X= valor que no se modifica

3.4 OTRAS SECCIONES DEL PROGRAMA

3.4.1 ESCOGE_OPCION

Antes de determinar la orden generada por el usuario, es necesario resetear la cuenta de los cinco minutos sin comunicación (encargada al timer1), para evitar que el robot entre al modo **SLEEP**, para esto los valores de los registros auxiliares del *TIMER_1* son restaurados: *AUXT_1A*= D'72' y *AUXT_1B*=D'8'. Inmediatamente se verifica si la trama que llega es verdadera observando que *DATO,7* se encuentre seteado caso contrario se realiza un salto a **FINAL**. Al ser verdadero se compara en forma sucesiva el registro auxiliar *AUXD1*, que contiene el número del comando, finalmente se procede a identificar el color del comando es decir si es: rojo, verde o naranja, esto se identifica en el registro *DATO* de la siguiente forma:

Tabla 3.14 Identificación de los Comandos

comando	NARANJA	VERDE	ROJO
0	L_STRIKE3	L_TURN_STEP	DOCE
1	L_SWEEP	L_THUMP	UNO
2	L_BURP	L_THROW	CUATRO
3	L_STRIKE2	C_SLEEP	CINCO
4	HIGH5	L_PICK_UP	DOS
5	L_STRIKE1	LEAN_BW	TRES
6	BULLDOZER	STEP	ONCE
7	DOPS	BACK_STEP	QUINCE
8	R_STRIKE3	R_TURN_STEP	CATORCE
9	R_SWEEP	R_THUMP	SEIS
10	WHISTLE	R_THROW	NUEVE
11	R_STRIKE	LISTEN	DIEZ
12	TALK_BACK	R_PICK_UP	SIETE
13	R_STRIKE1	LEAN_FWD	OCHO
14	ROAR	RESET	TRECE
15	ALL_DEMO	EXECUTE	DIECINUEVE
16	POWER_OFF	WAKE_UP	VEINTE
17	DEMO_1	RIGHT	DIECISEIS
18	DEMO_2	LEFT	DIECIOCHO
19	DANCE	SONIC	DIECISIETE

Nota: las palabras sombreadas no realizan ninguna función

Tabla 3.15 Forma de Determinar el Color del Dato

DATO igual a	COLOR
B'X00XXXXX'	Rojo
B'X01XXXXX'	Verde
B'X10XXXXX'	Naranja

Una vez que se determinó el comando y su color se procede a llamar a la subrutina o salto correspondiente. Al terminar las subrutinas correspondientes se produce un salto a FINAL.

3.4.2 PRIORIDAD

Esta subrutina permite asignar la prioridad de conversión para cada uno de los sensores de Posición, dependiendo del orden de accionamiento de cada uno de los actuadores, es decir que el primer motor accionado va a ser el primero en apagarse, siempre y cuando no vuelva a ser accionado antes de que se apague ya que su prioridad será cambiada.

Para esto se tiene cinco registros auxiliares *AUXP*, *PRIOR_1*, *PRIOR_2*, *PRIOR_3*, *PRIOR_4*, los mismos que dan la libertad de realizar hasta cuatro movimientos distintos y guardar el orden en que los sensores de posición deben ser leídos.

El primer registro auxiliar *AUXP* se encarga de trasladar el nuevo requerimiento de prioridad, en el cual solamente uno de los bits menos significativos del byte de prioridad podrá estar seteado.

Tabla 3.16 Byte de Prioridad

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Prioridad Ocupada	----	----	----	Prioridad Sensor Pierna Izquierda	Prioridad Sensor Pierna Derecha	Prioridad Sensor Hombro Izquierdo	Prioridad Sensor Hombro Derecho

Este valor será colocado en cualquiera de los niveles de prioridad, siempre que no se encuentren ocupados o que la prioridad del sensor no se repita, en último caso (si se repite la prioridad) esta será guardada en el último registro disponible y se procederá a bajar un nivel de prioridad a todas hasta sobrescribir el nivel que estaba repetido.

Los cuatro últimos registros representan niveles de atención jerarquizados en la siguiente forma:

Tabla 3.17 Jerarquía de los Registros Auxiliares de Prioridad.

Registro	Nivel	Jerarquía
PRIOR_1	1er	prioridad máxima
PRIOR_2	2do	prioridad media
PRIOR_3	3ero	prioridad baja
PRIOR_4	4to	prioridad mínima

3.4.3 FINAL

Se borra el bit *AUXT,6* (bandera de llegada de dato) y se retorna al lazo principal con un salto al **Lazo_Principal**.

3.4.4 POP

En POP se devuelve el valor que tenía el acumulador (*W*) antes de ingresar a cualquiera de las interrupciones, $W=W_AUX$. y se retorna de la interrupción.

3.5 DIAGRAMAS DE FLUJO

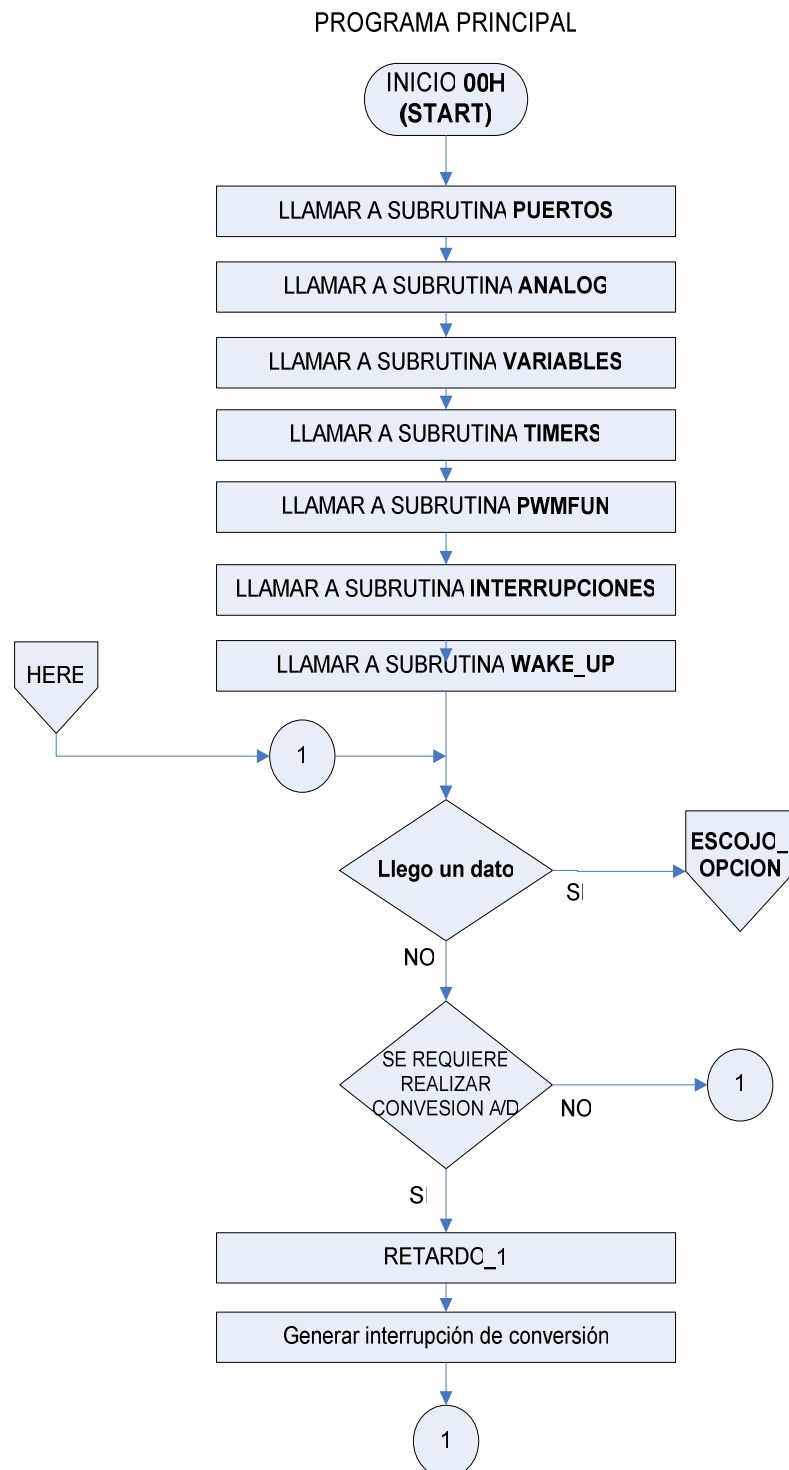


Figura 3.3 Programa Principal (Lazo Principal)

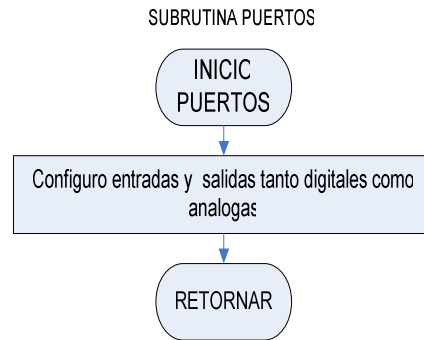


Figura 3.4 Subrutina Puertos

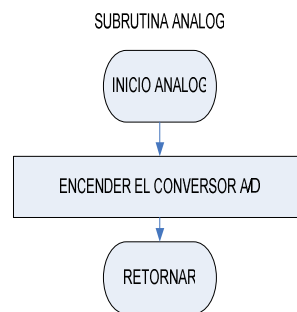


Figura 3.5 Subrutina Analog

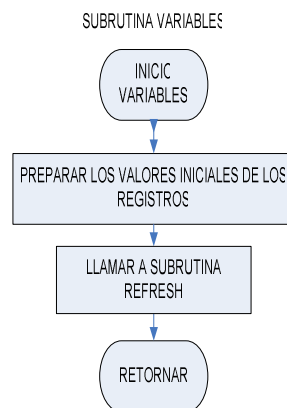


Figura 3.6 Subrutina Variables

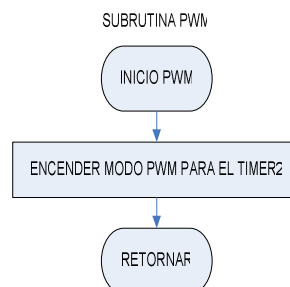


Figura 3.7 Subrutina PWM

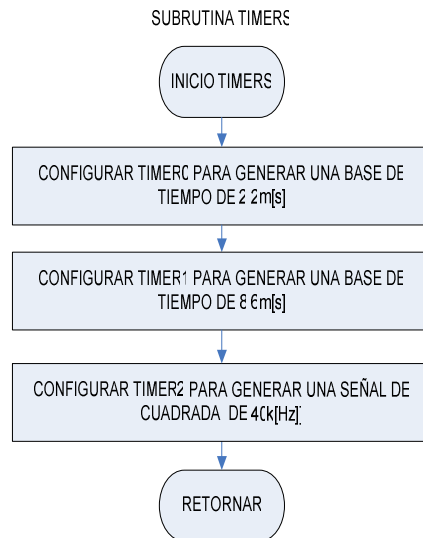


Figura 3.8 Subrutina Timers

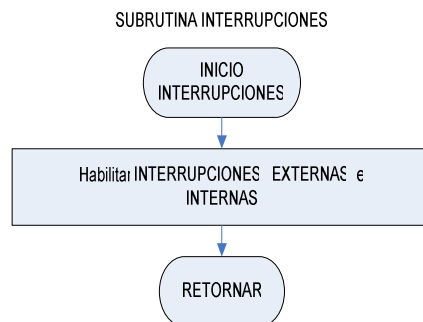


Figura 3.9 Subrutina Interrupciones

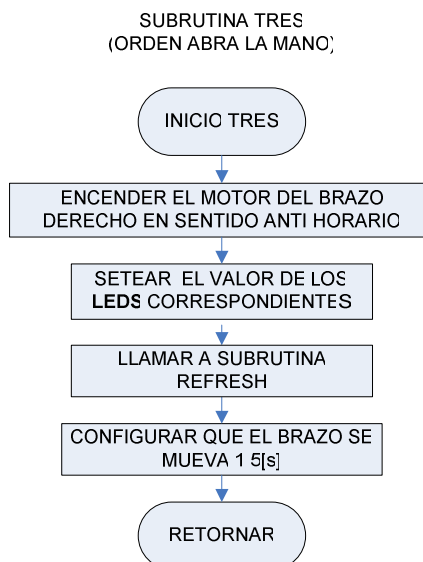


Figura 3.10 Subrutina Tres

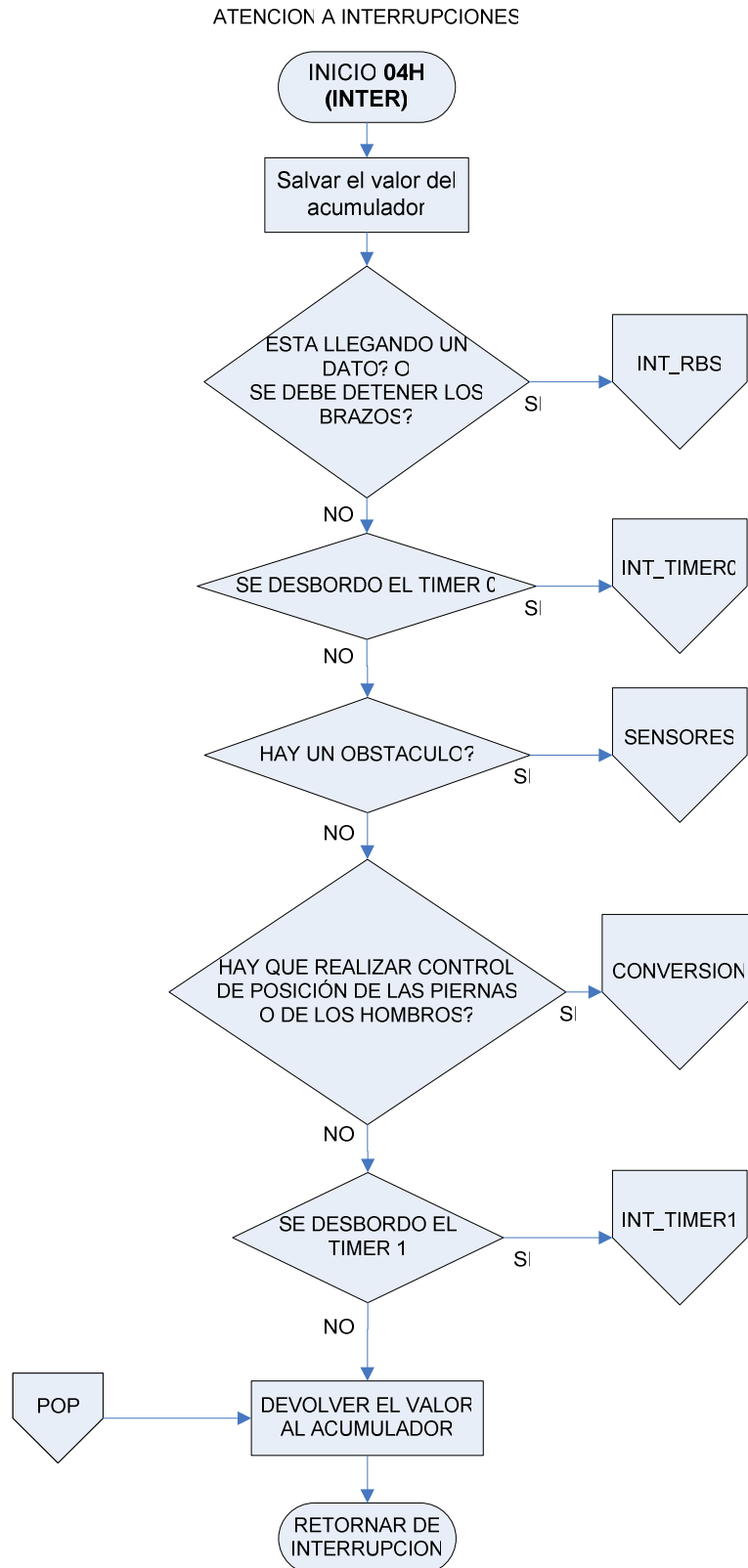


Figura 3.11 Atención a Interrupciones

INTERRUPCIONES EN RBS

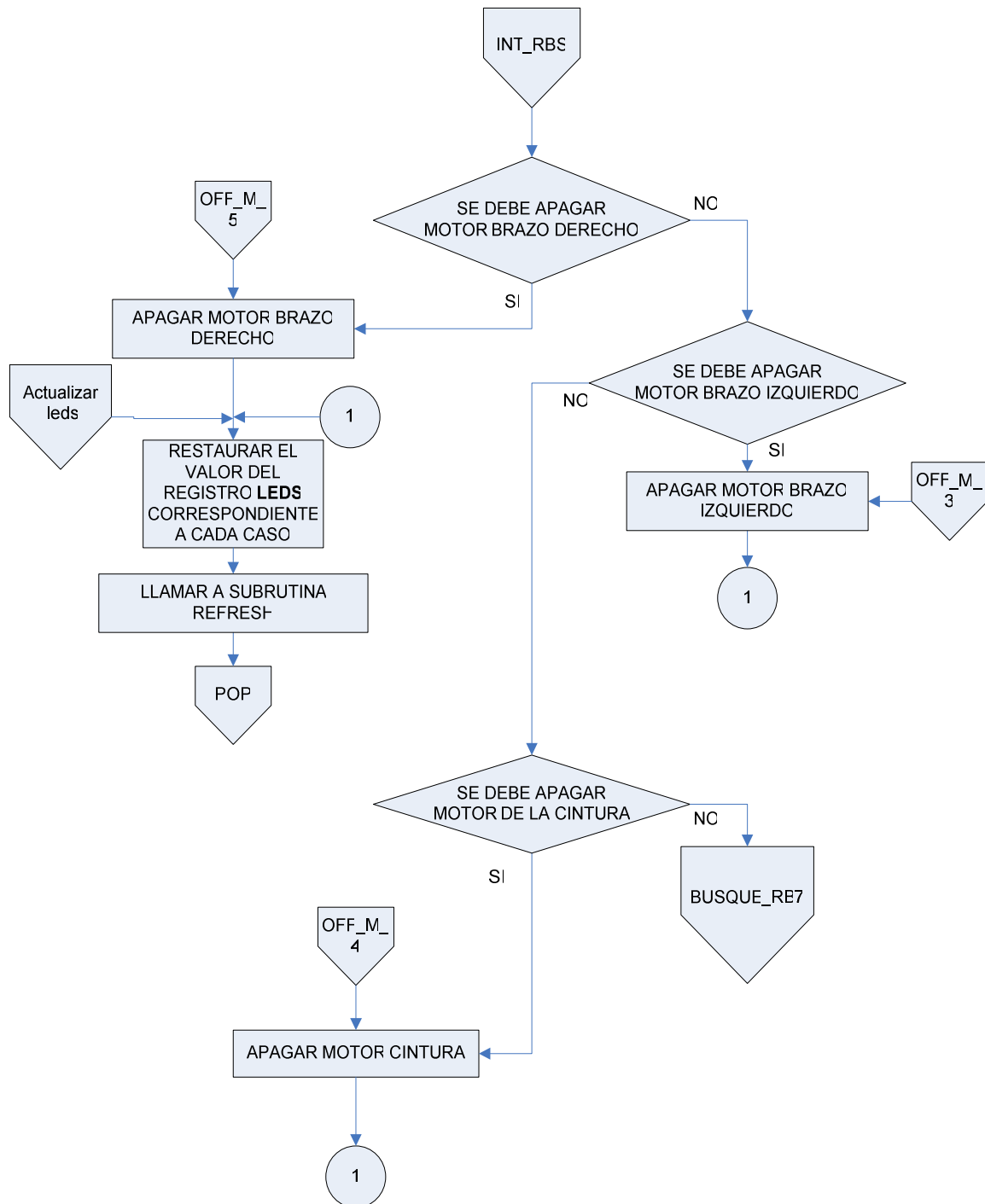


Figura 3.12 Interrupción en RBS

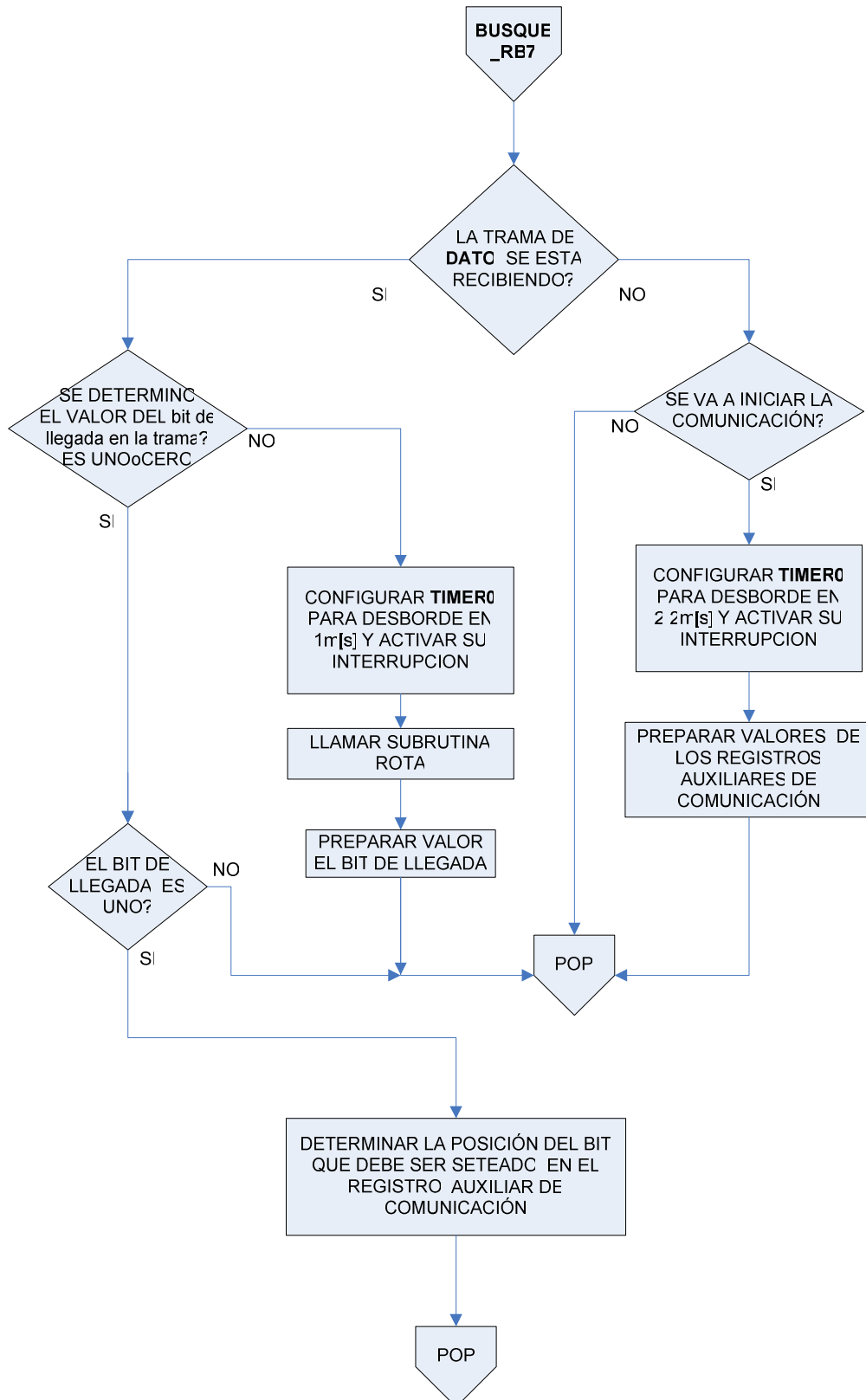


Figura 3.12 Continuación Interrupción en RBS

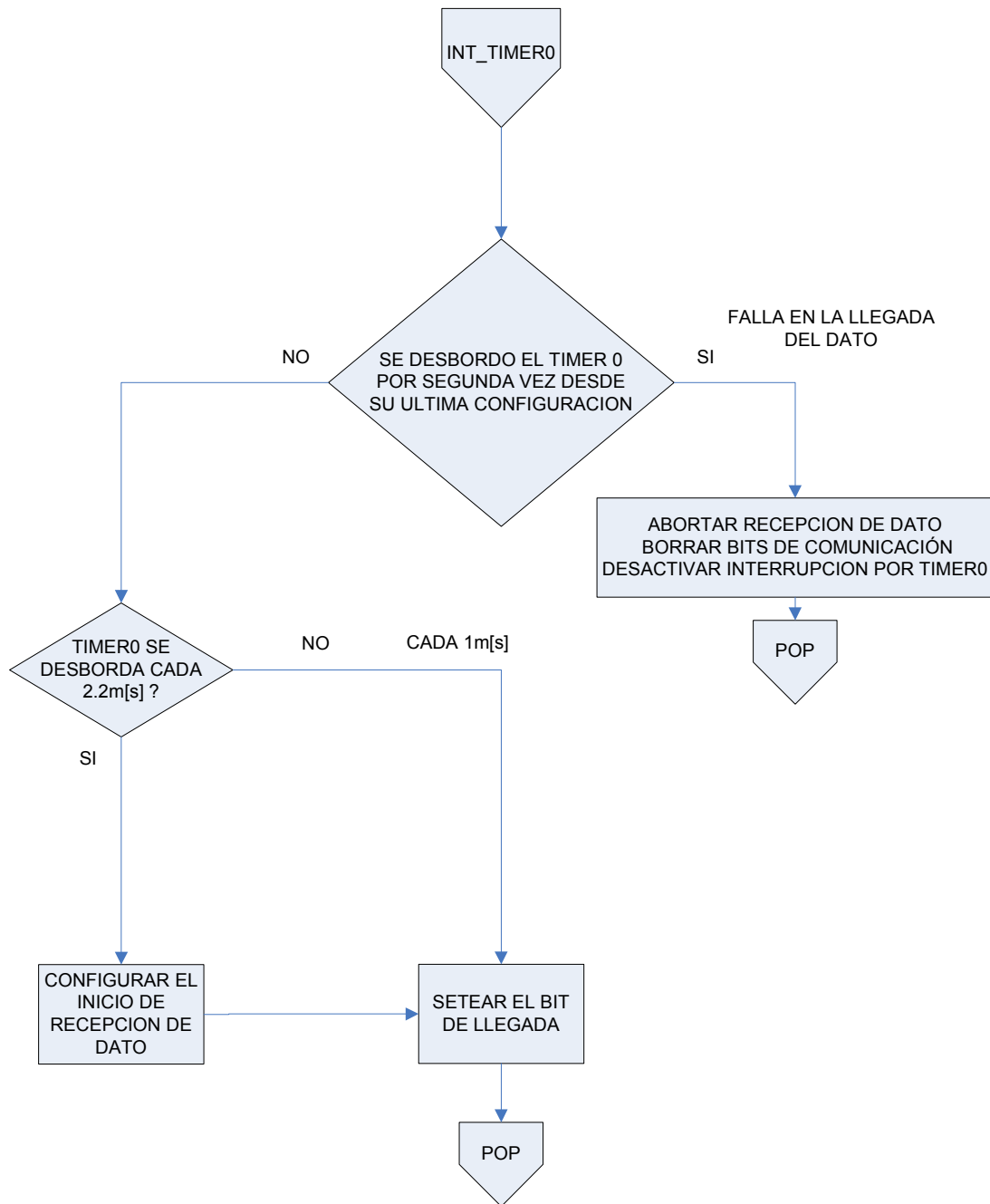


Figura 3.13 Interrupción Timer0

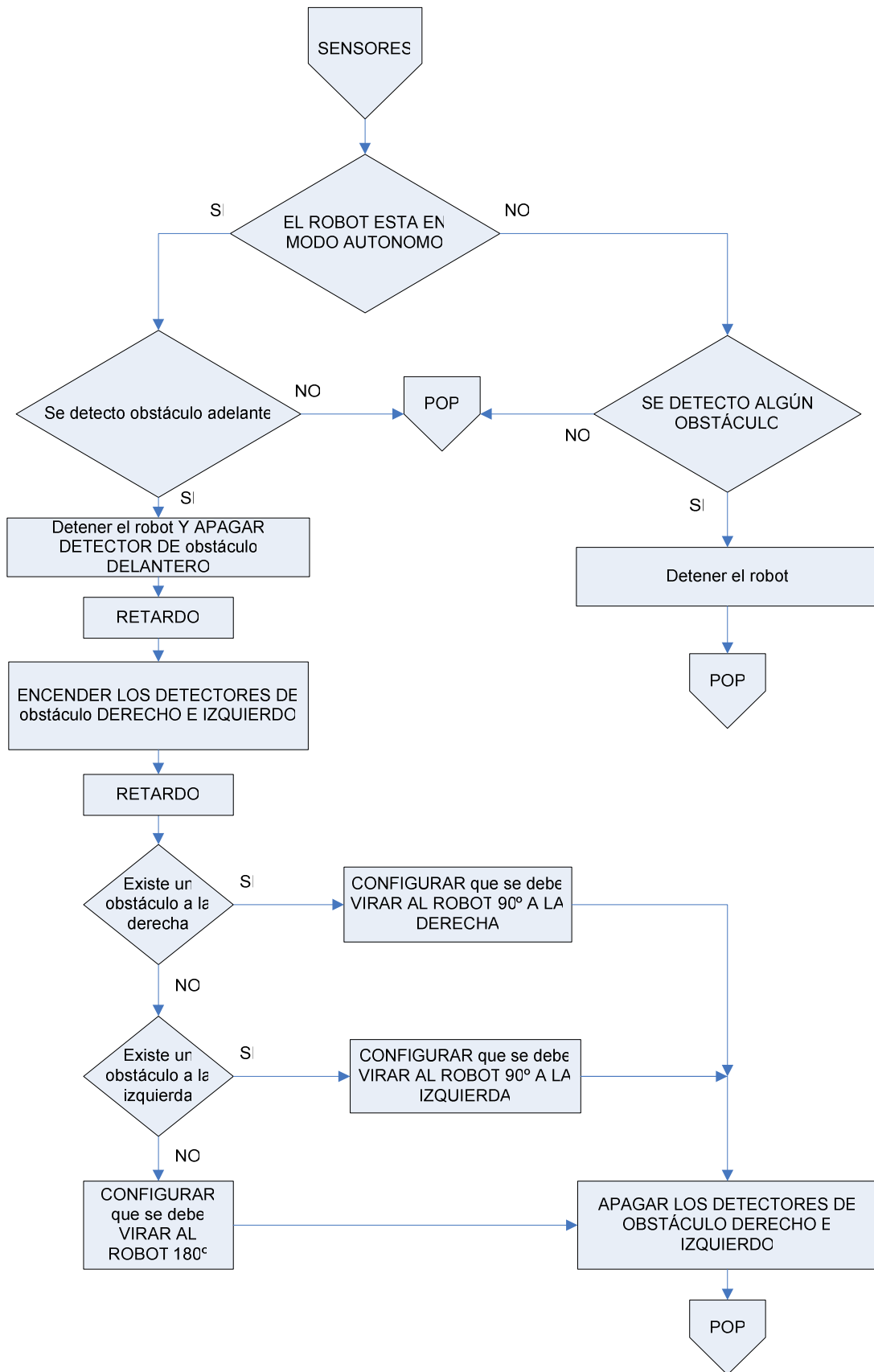


Figura 3.14 Interrupción Sensores

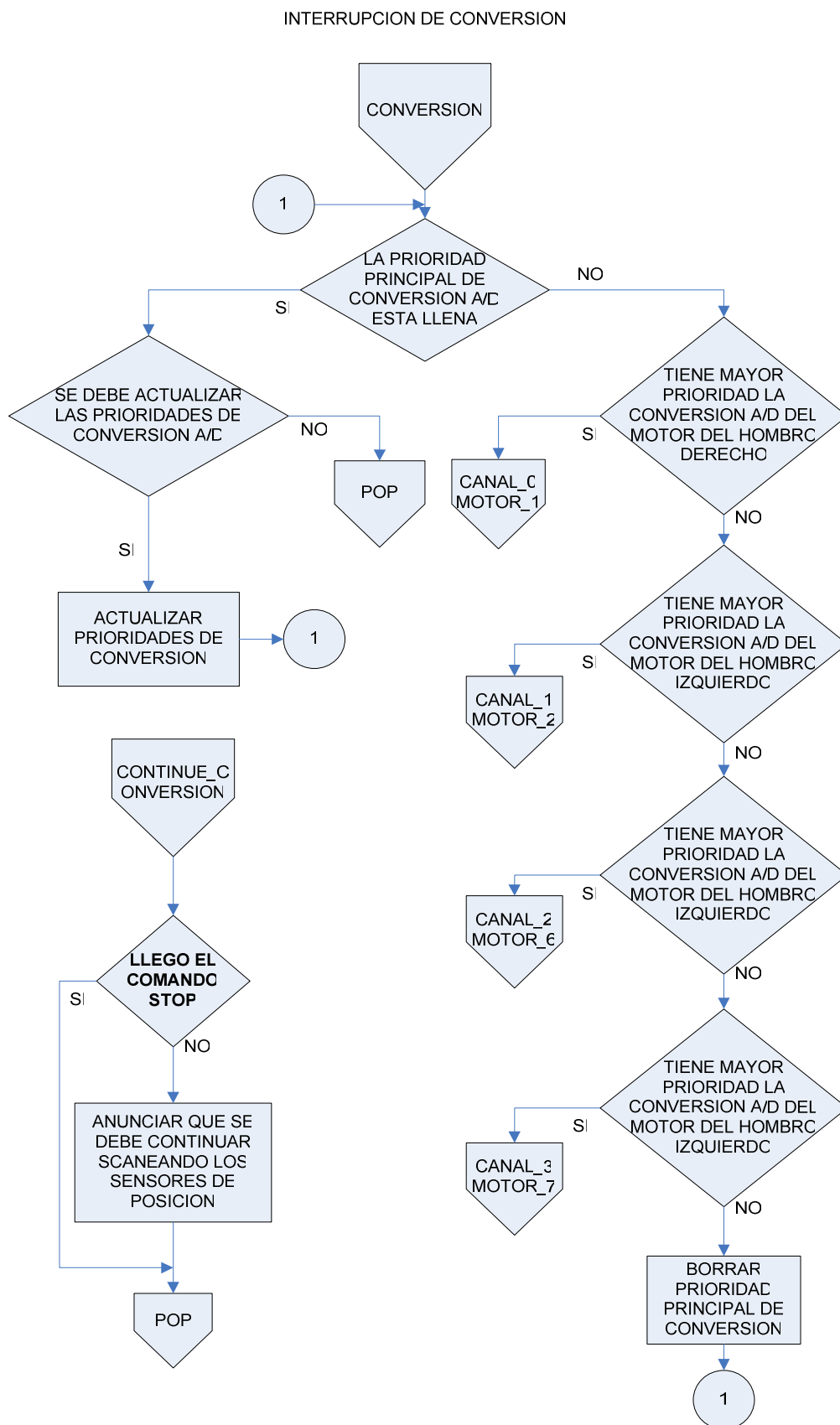


Figura 3.15 Interrupción Conversión

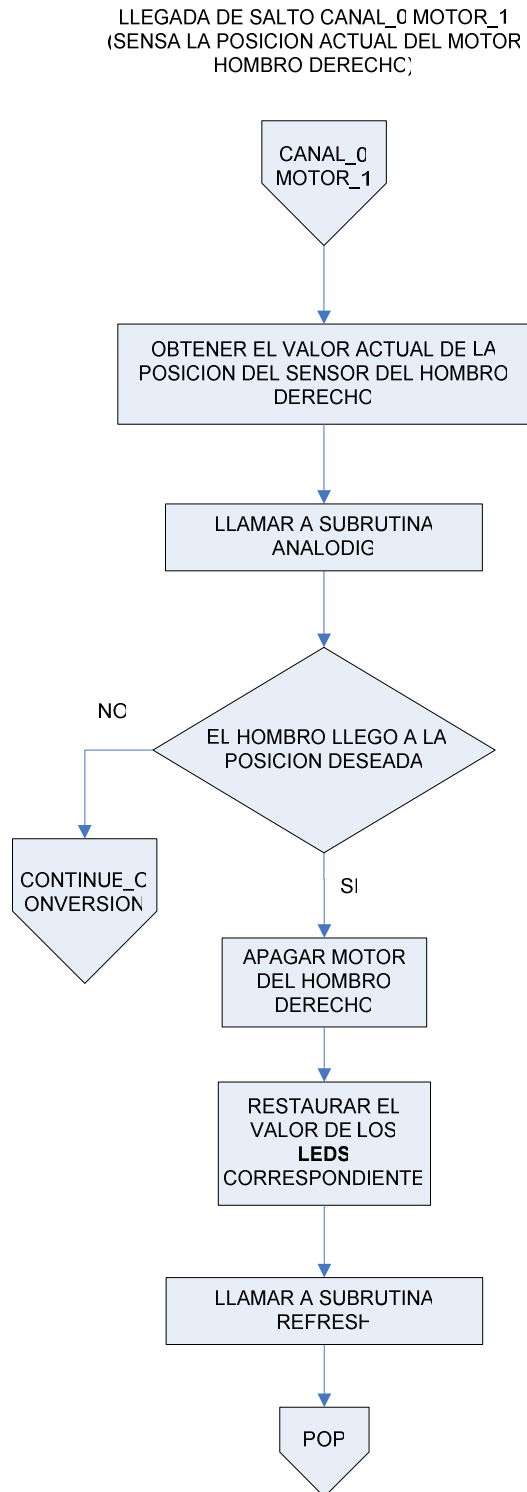


Figura 3.15 Continuación Interrupción Conversión

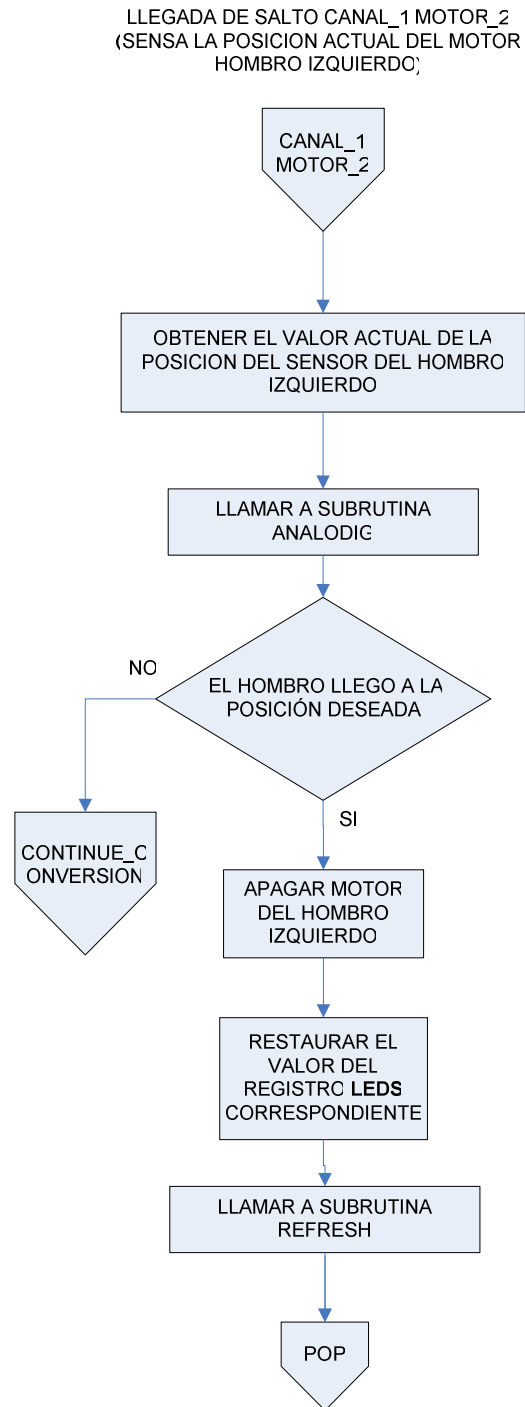


Figura 3.15 Continuación Interrupción Conversión

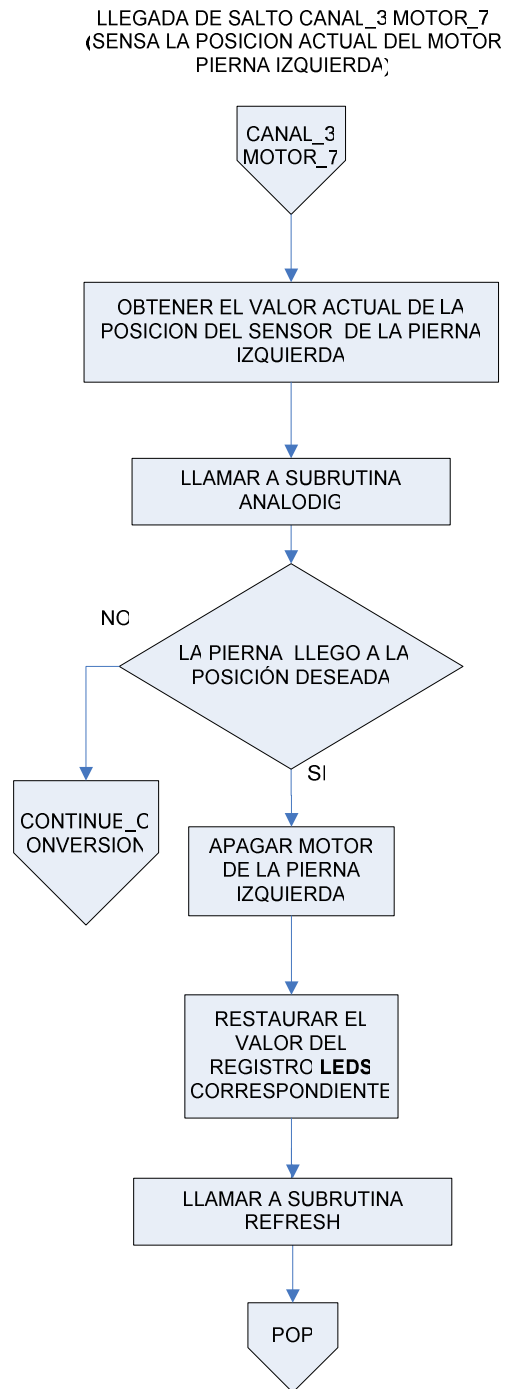


Figura 3.15 Continuación Interrupción Conversión

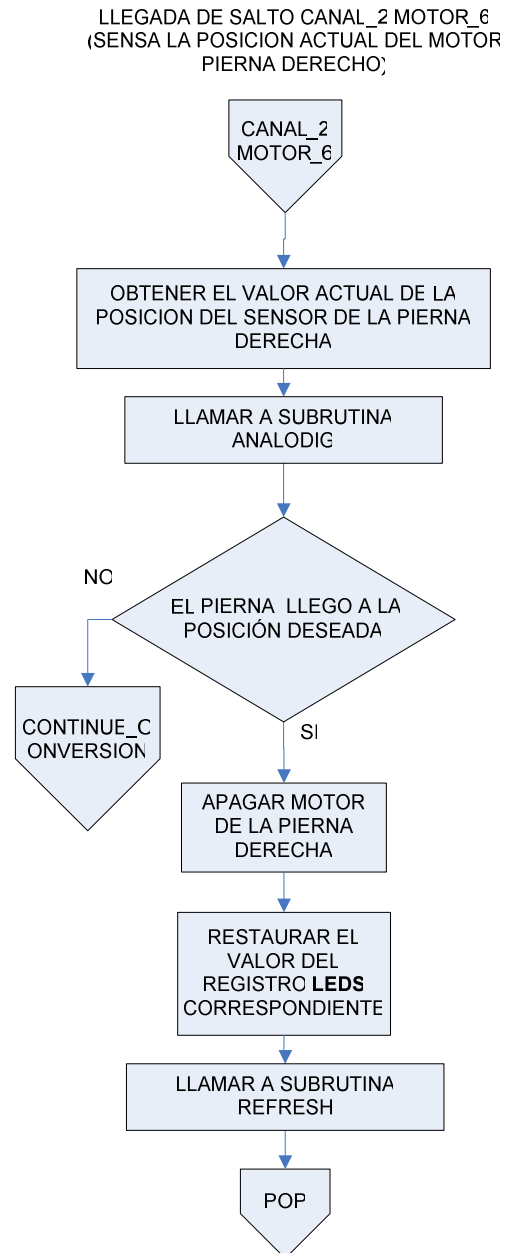


Figura 3.15Continuación Interrupción Conversión

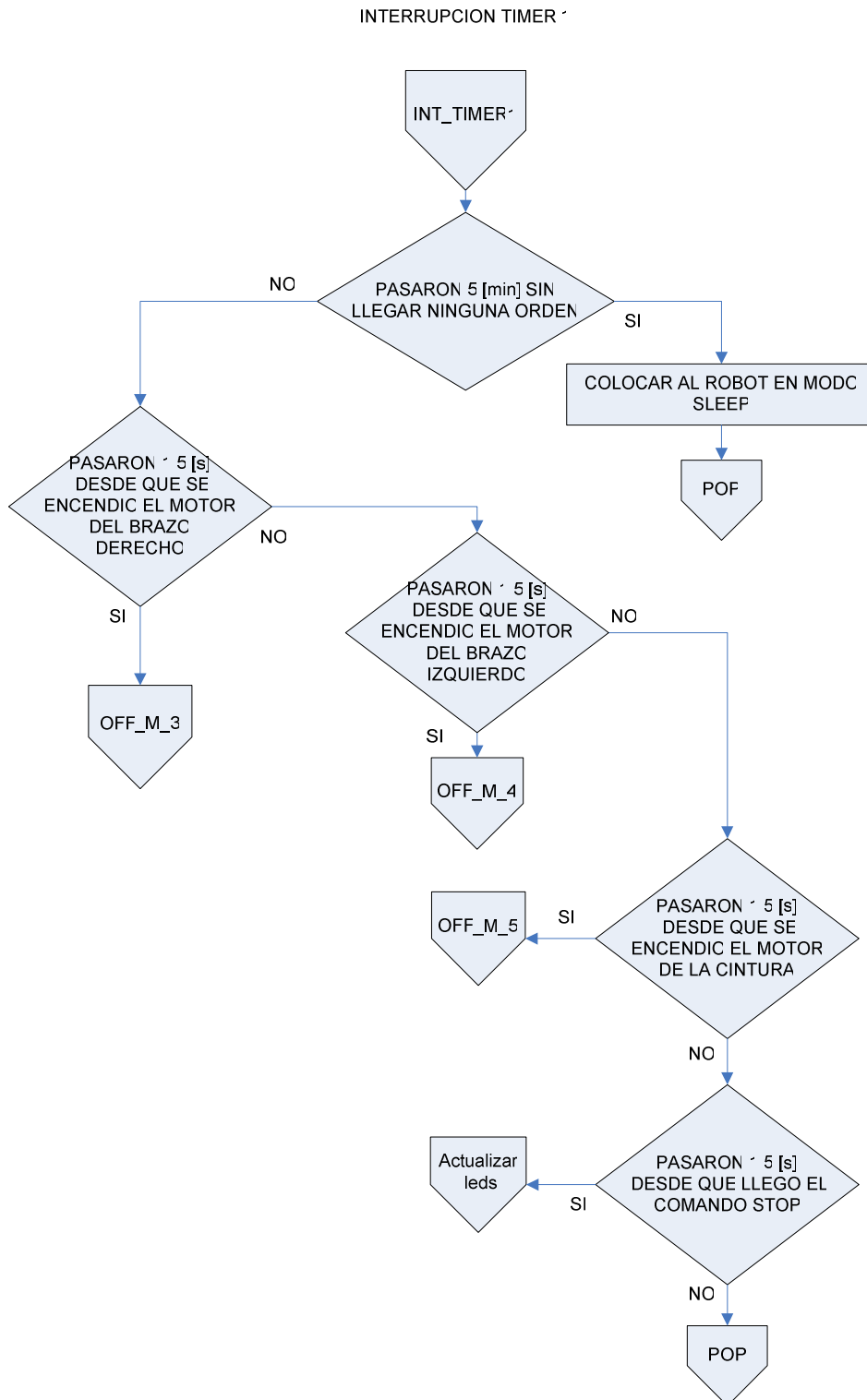


Figura 3.16 Interrupción Timer1

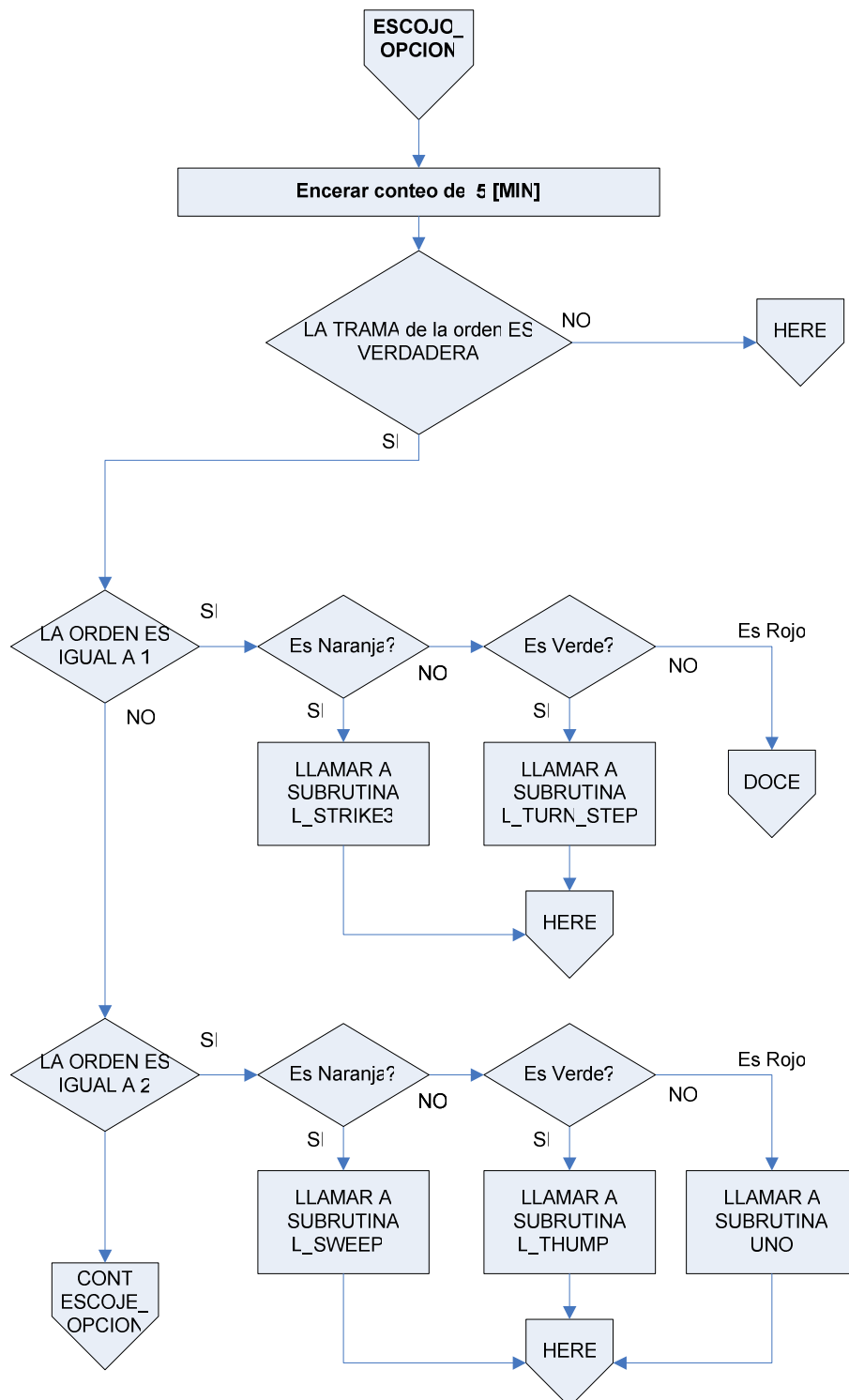


Figura 3.17 ESCOJO OPCION

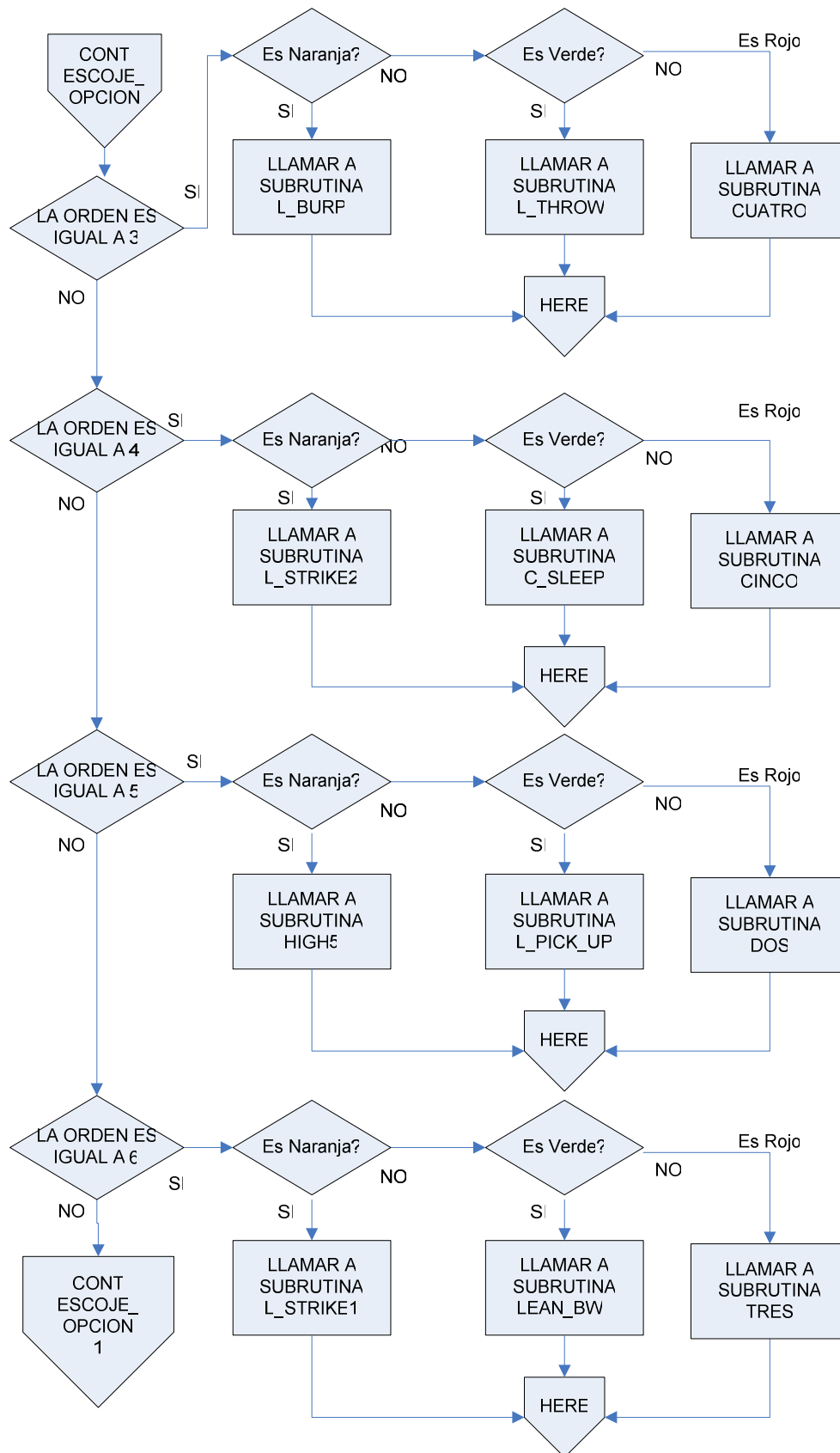


Figura 3.17 Continuación ESCOJE OPCION

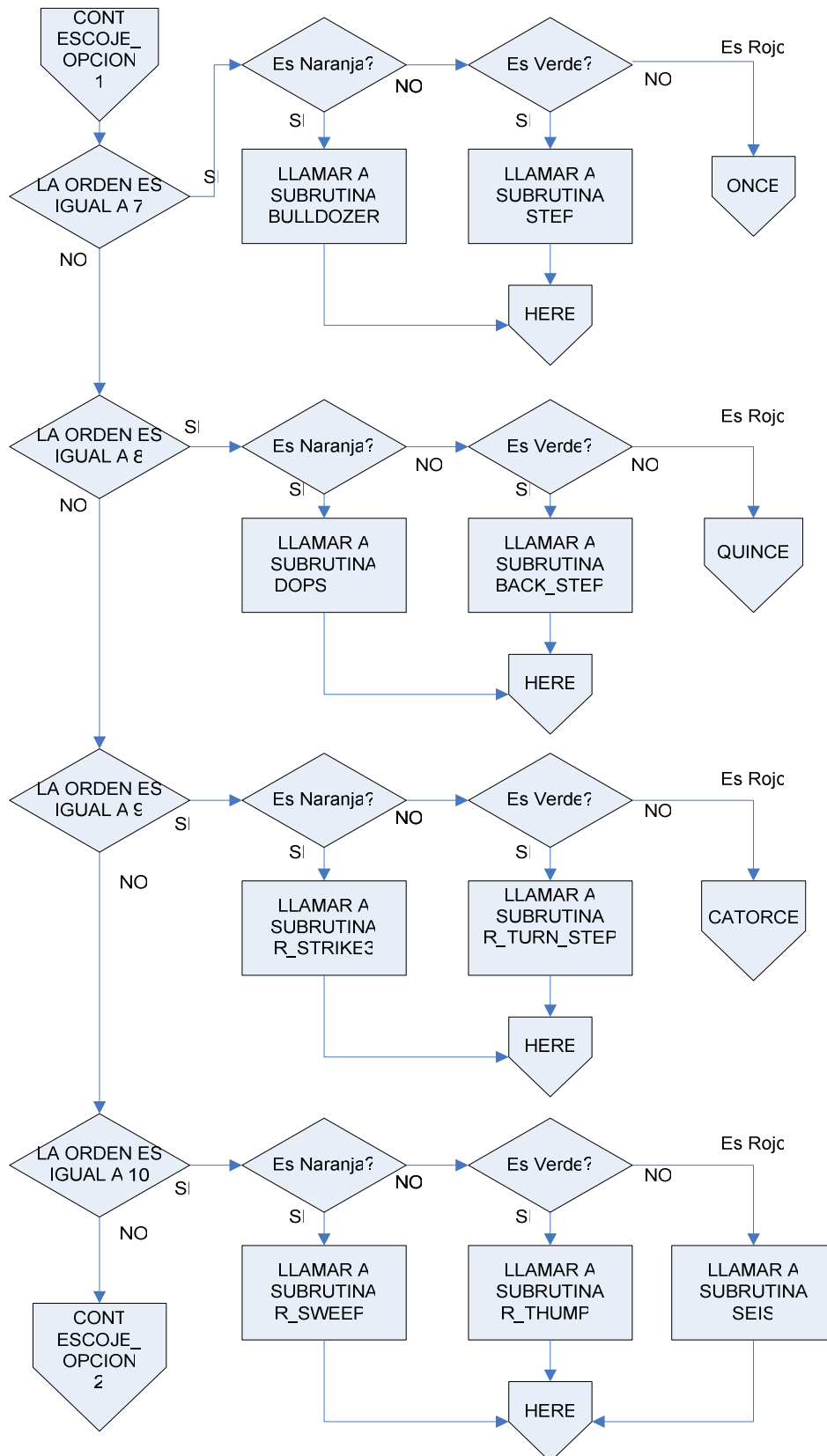


Figura 3.17 Continuación ESCOGE_OPCION

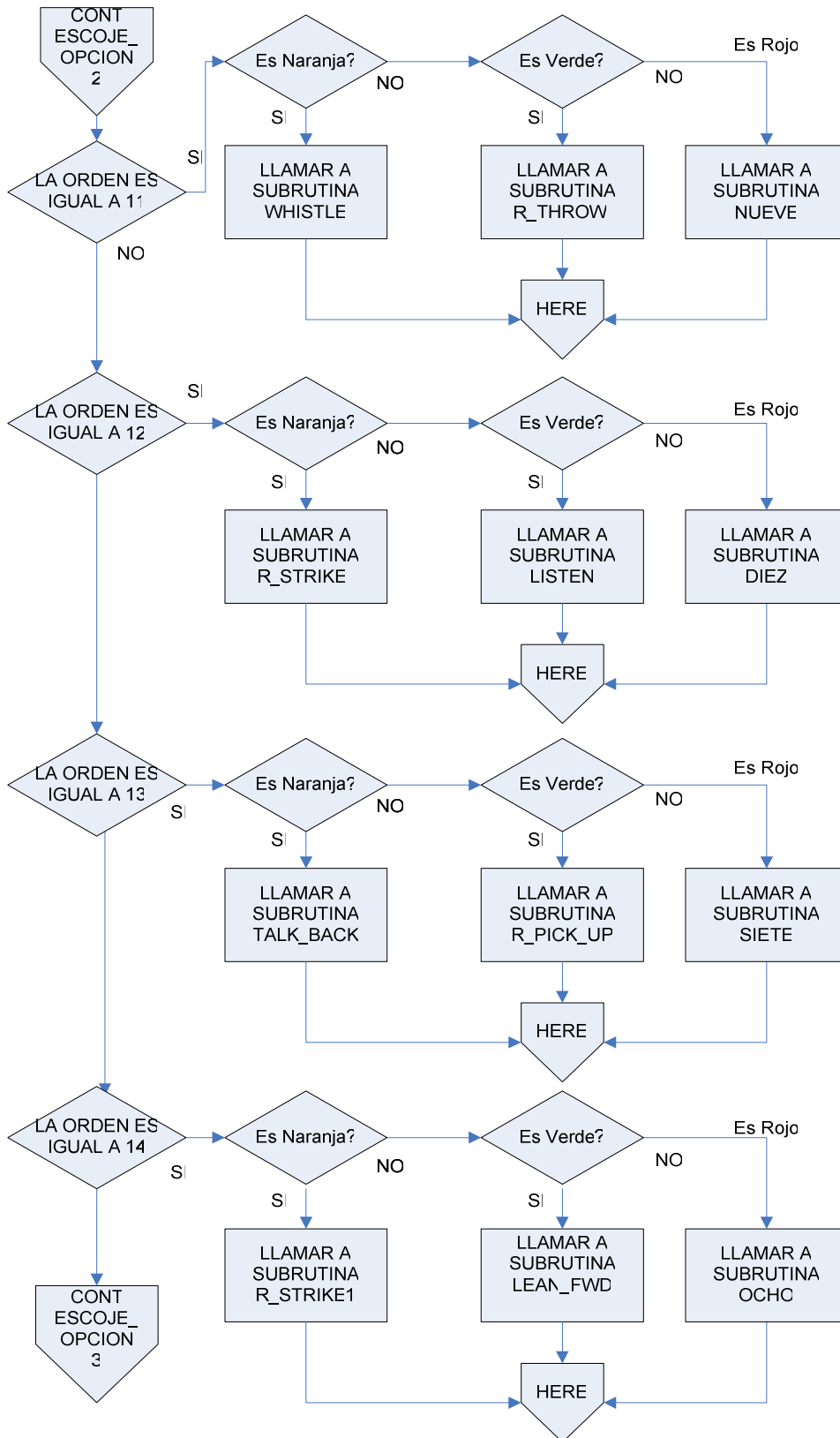


Figura 3.17 Continuación ESCOGE_OPCION

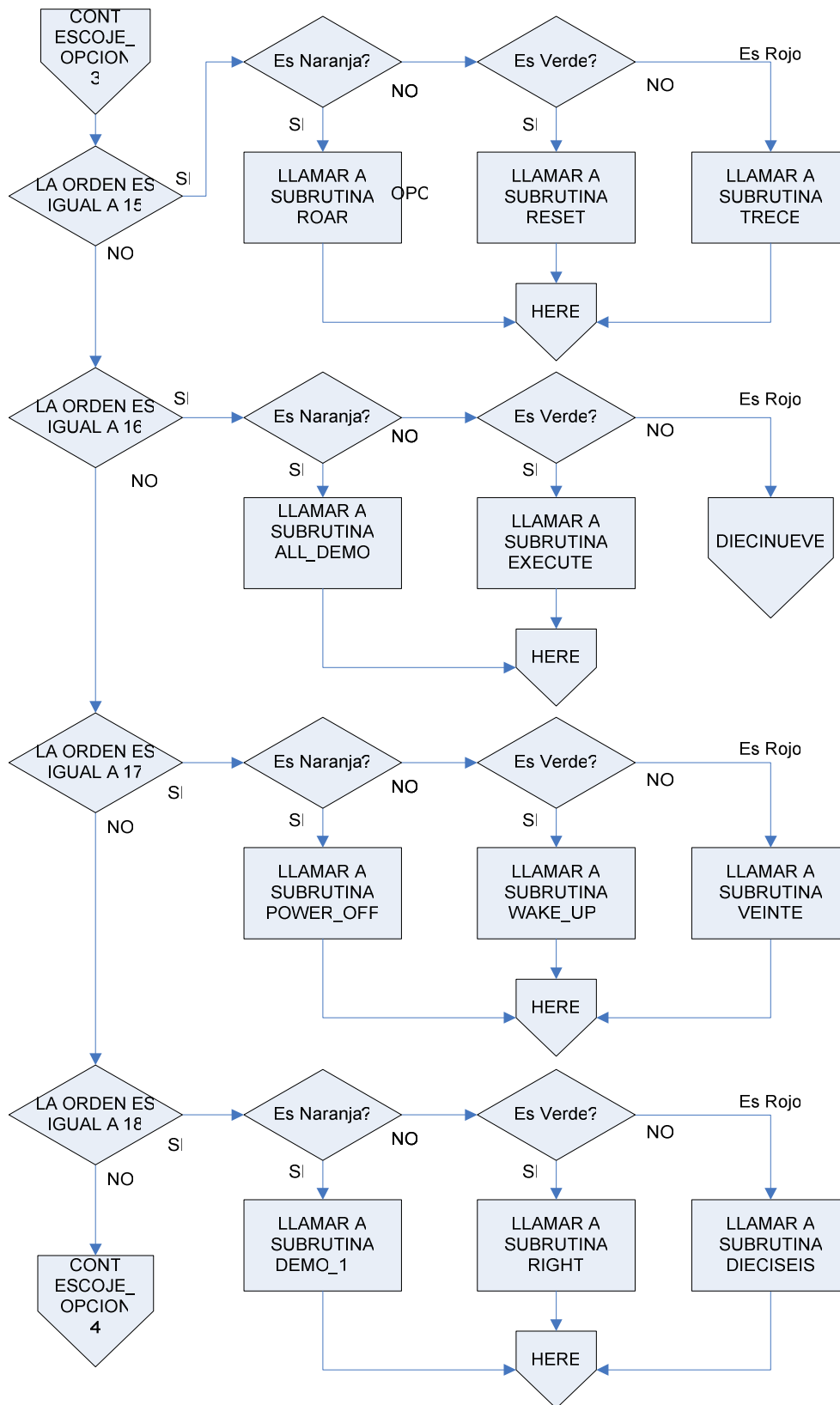


Figura 3.17 Continuación ESCOGE_OPCION

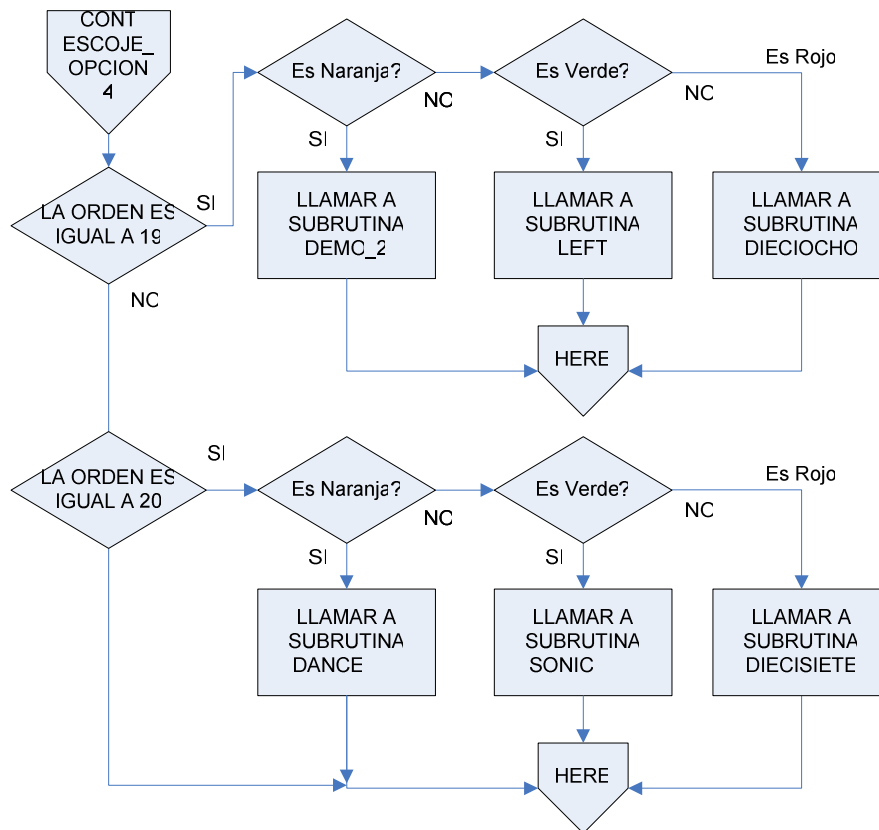


Figura 3.17 Continuación ESCOGE OPCION

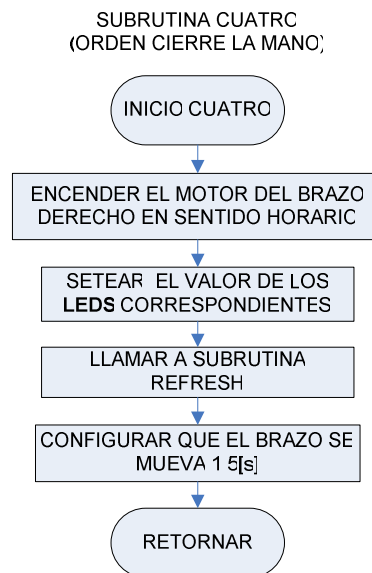


Figura 3.18 Subrutina Cuatro

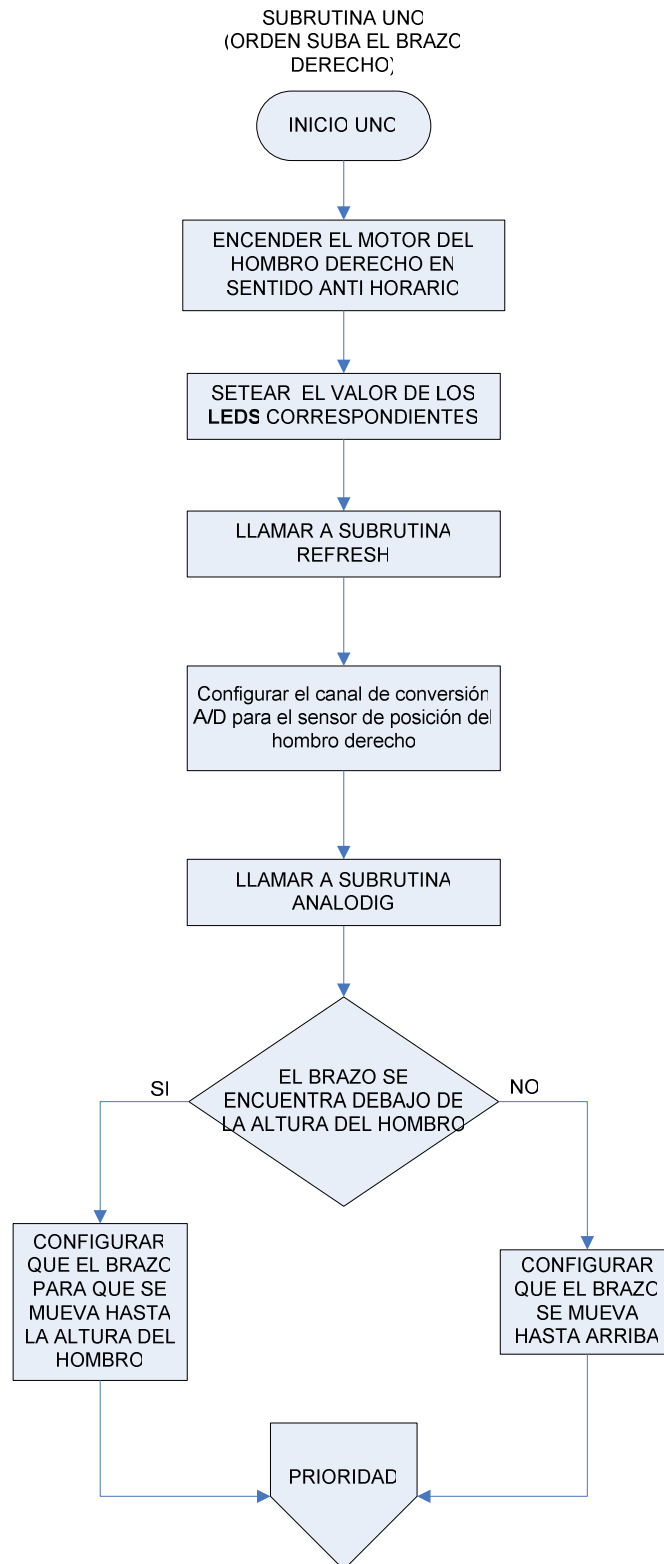


Figura 3.19 Subrutina Uno

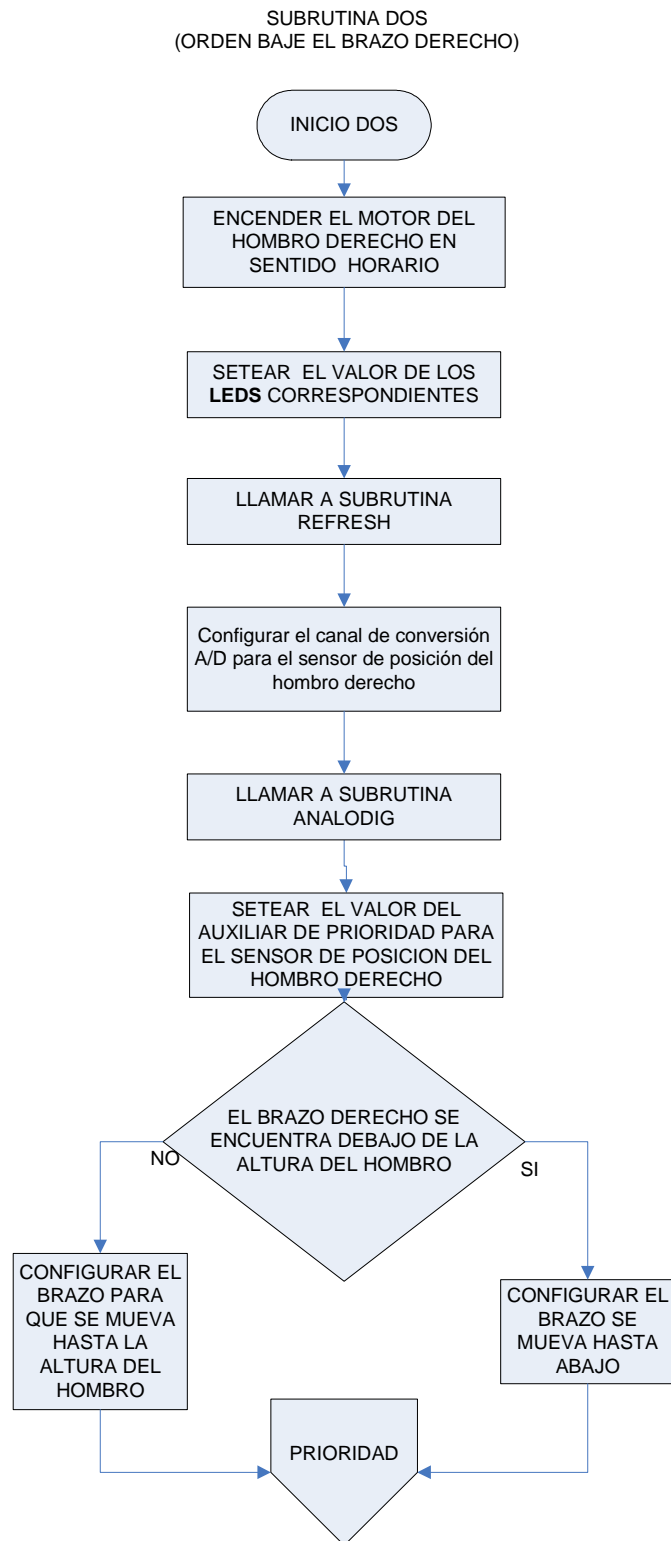


Figura 3.20 Subrutina Dos

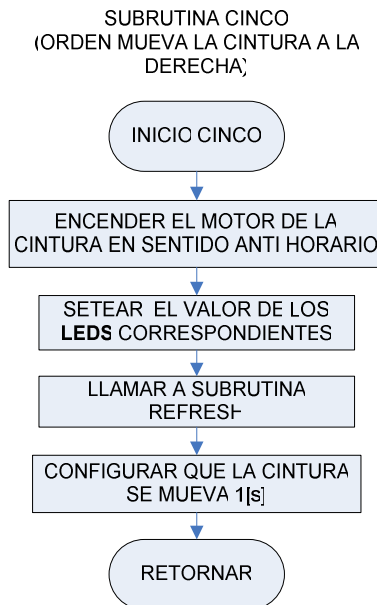


Figura 3.21 Subrutina Cinco

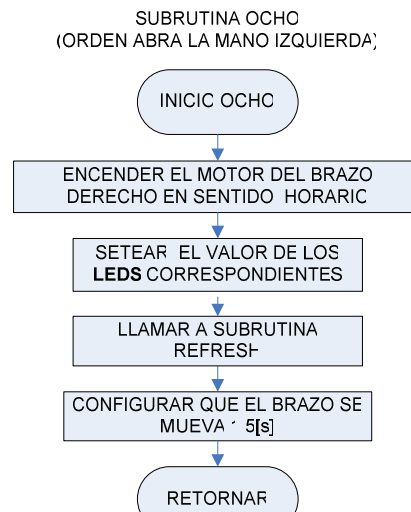


Figura 3.22 Subrutina Ocho

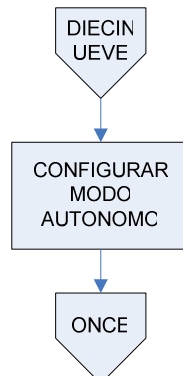


Figura 3.23 Subrutina Diecinueve

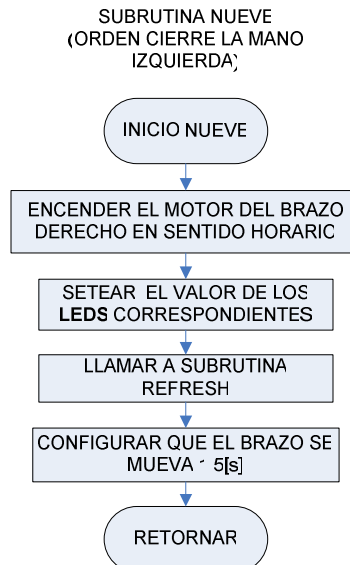


Figura 3.24 Subrutina Nueve

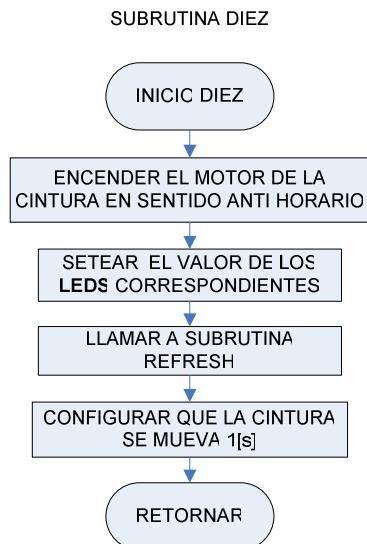


Figura 3.25 Subrutina Diez

LLEGADA DE SALTO ESCOJA DIRECCION
(ESCOJE LA DIRECCION QUE DEBE TOMAR
EL ROBOT);

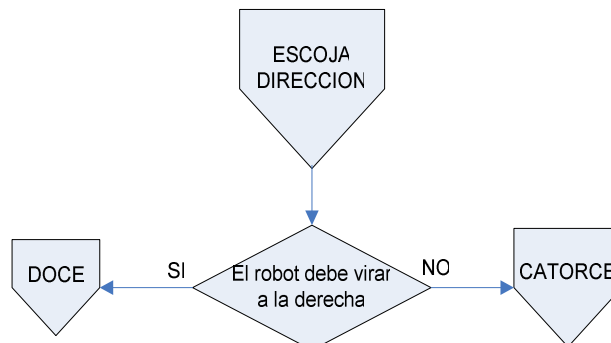


Figura 3.26 Escoja Dirección (Al encontrar obstáculo en Modo Autónomo)

SUBROUTINA SEIS
(ORDEN SUBA EL BRAZO IZQUIERDO)

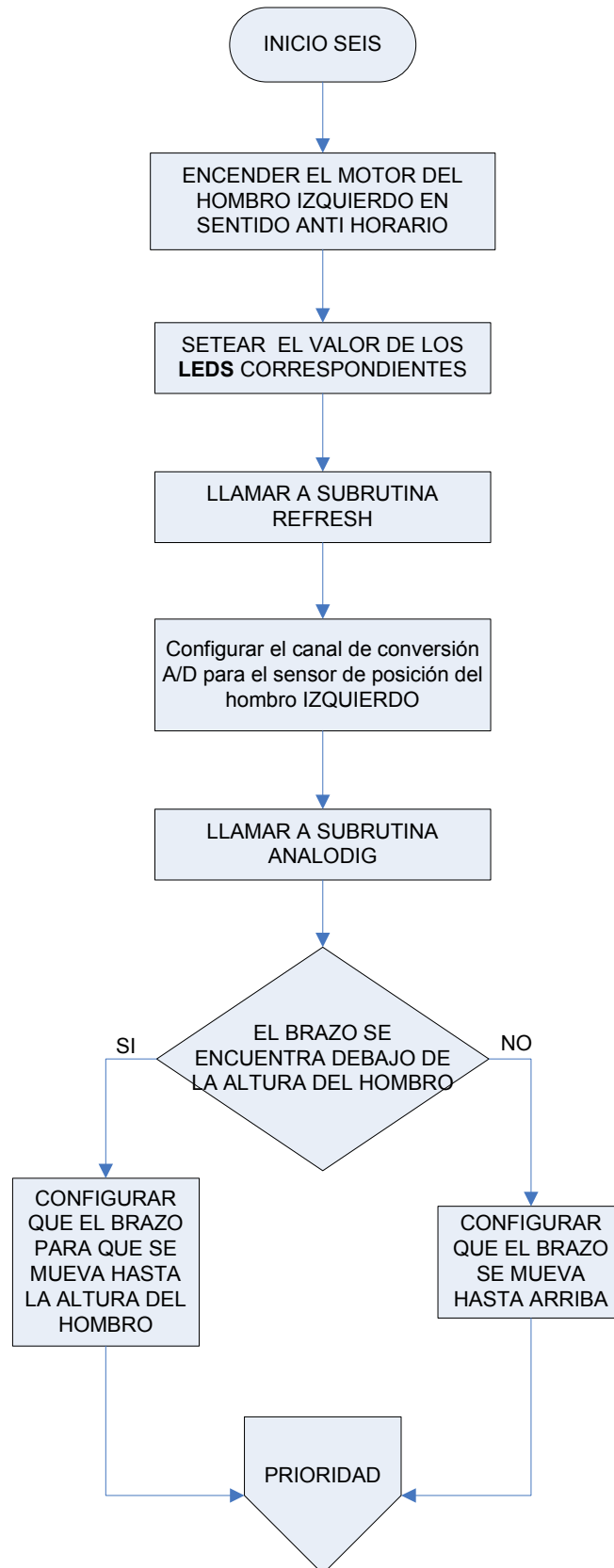


Figura 3.27 Subrutina Seis

SUBROUTINA SIETE
(ORDEN BAJE EL BRAZO IZQUIERDO)

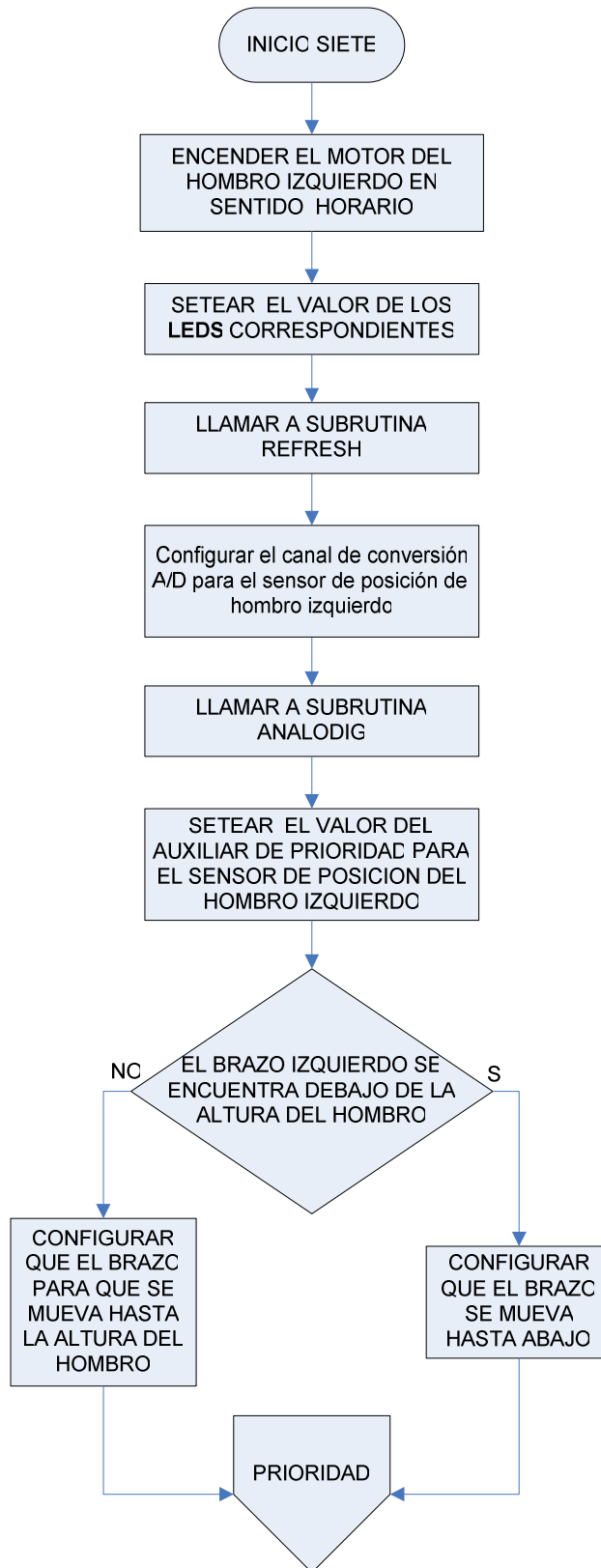


Figura 3.28 Subrutina Siete

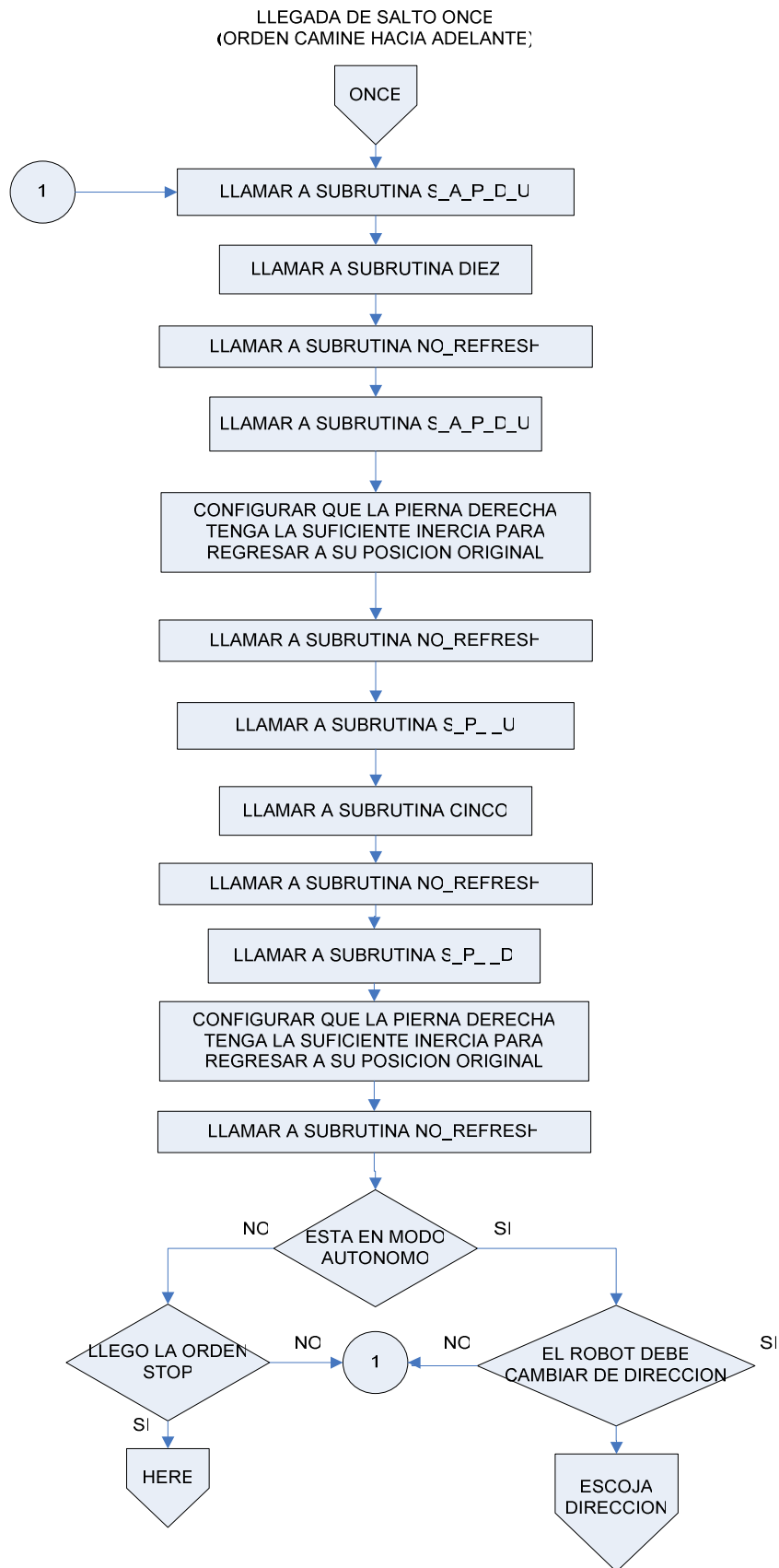


Figura 3.29 Subrutina Once

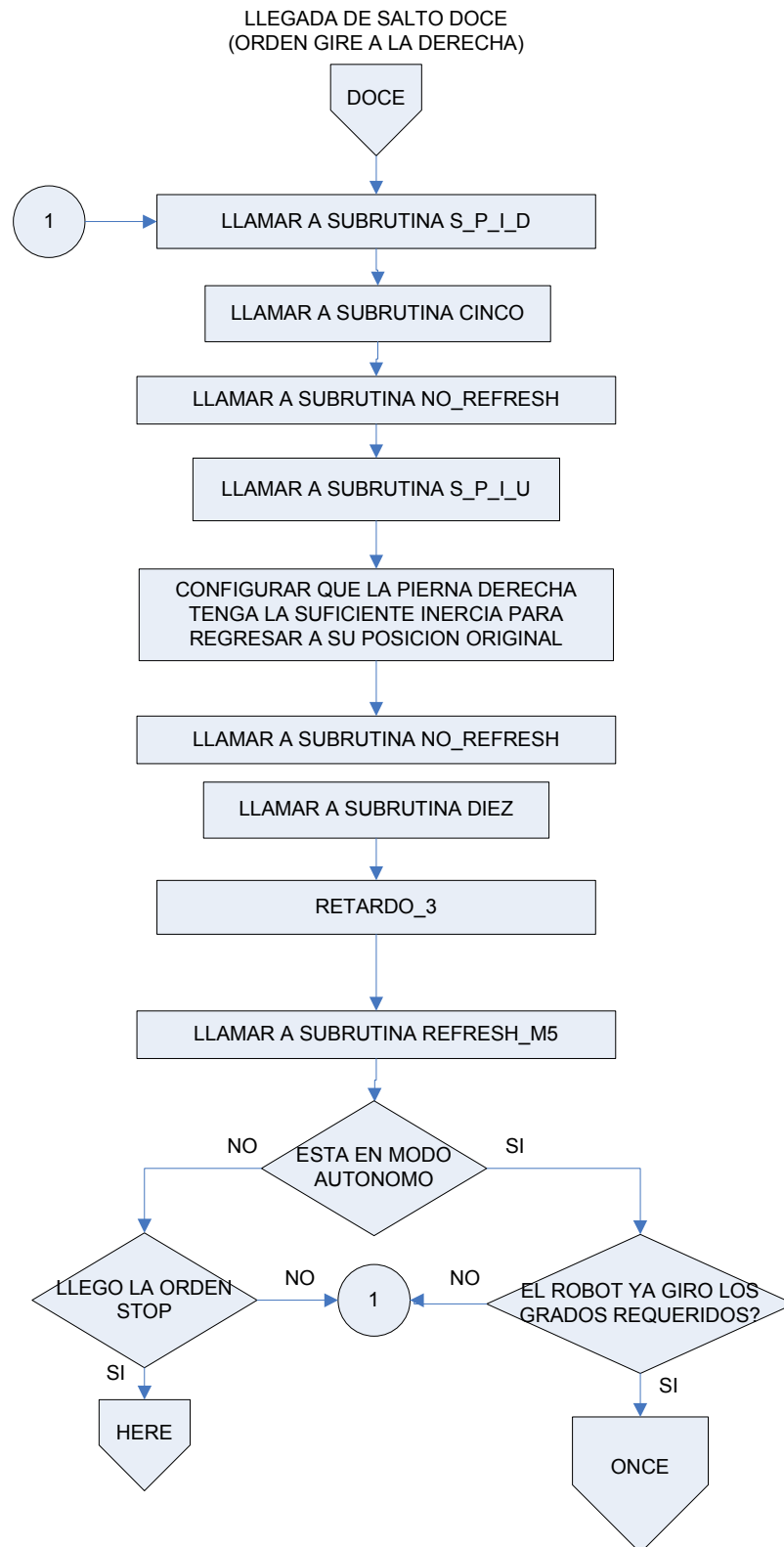


Figura 3.30 Subrutina Doce

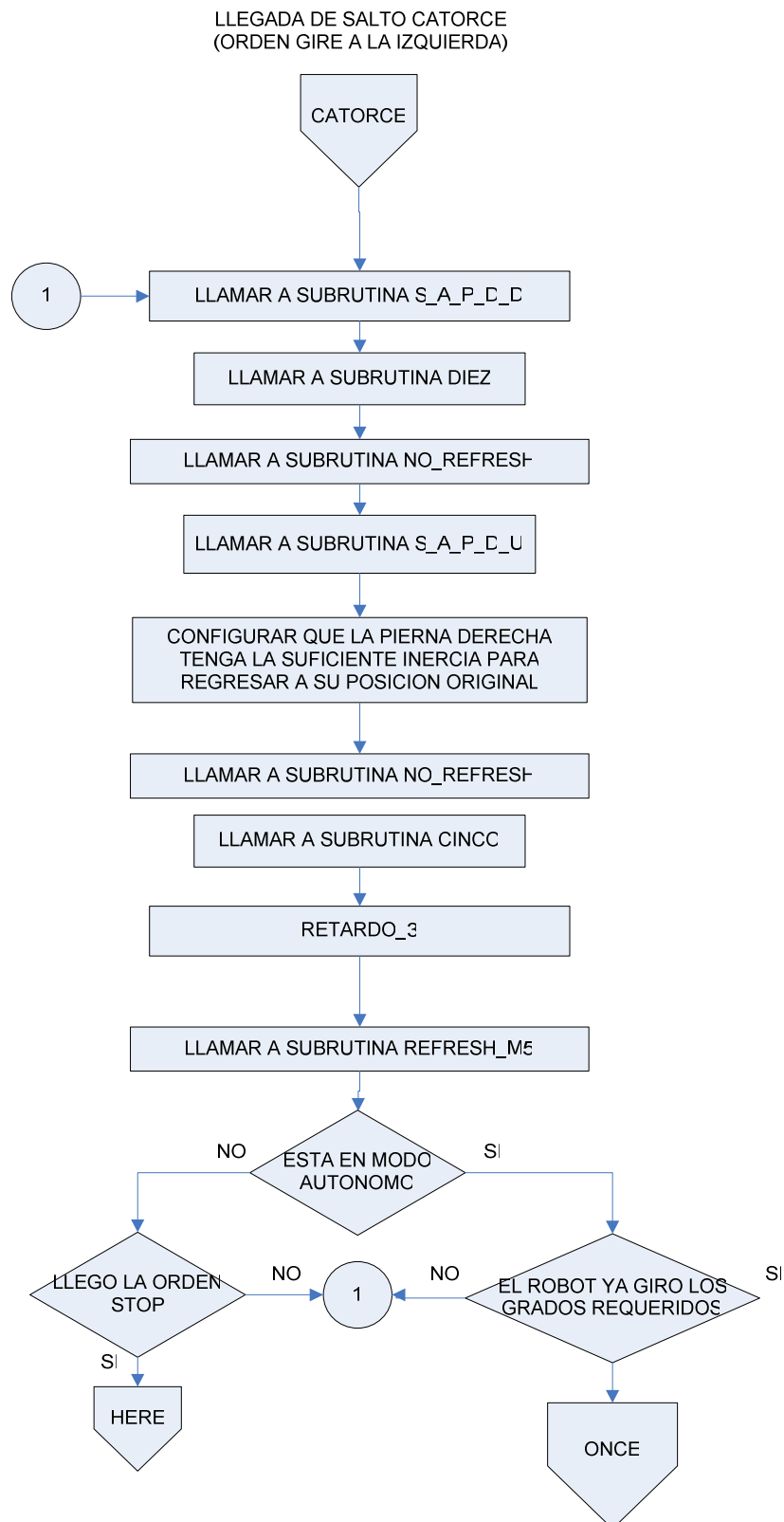


Figura 3.31 Subrutina Catorce

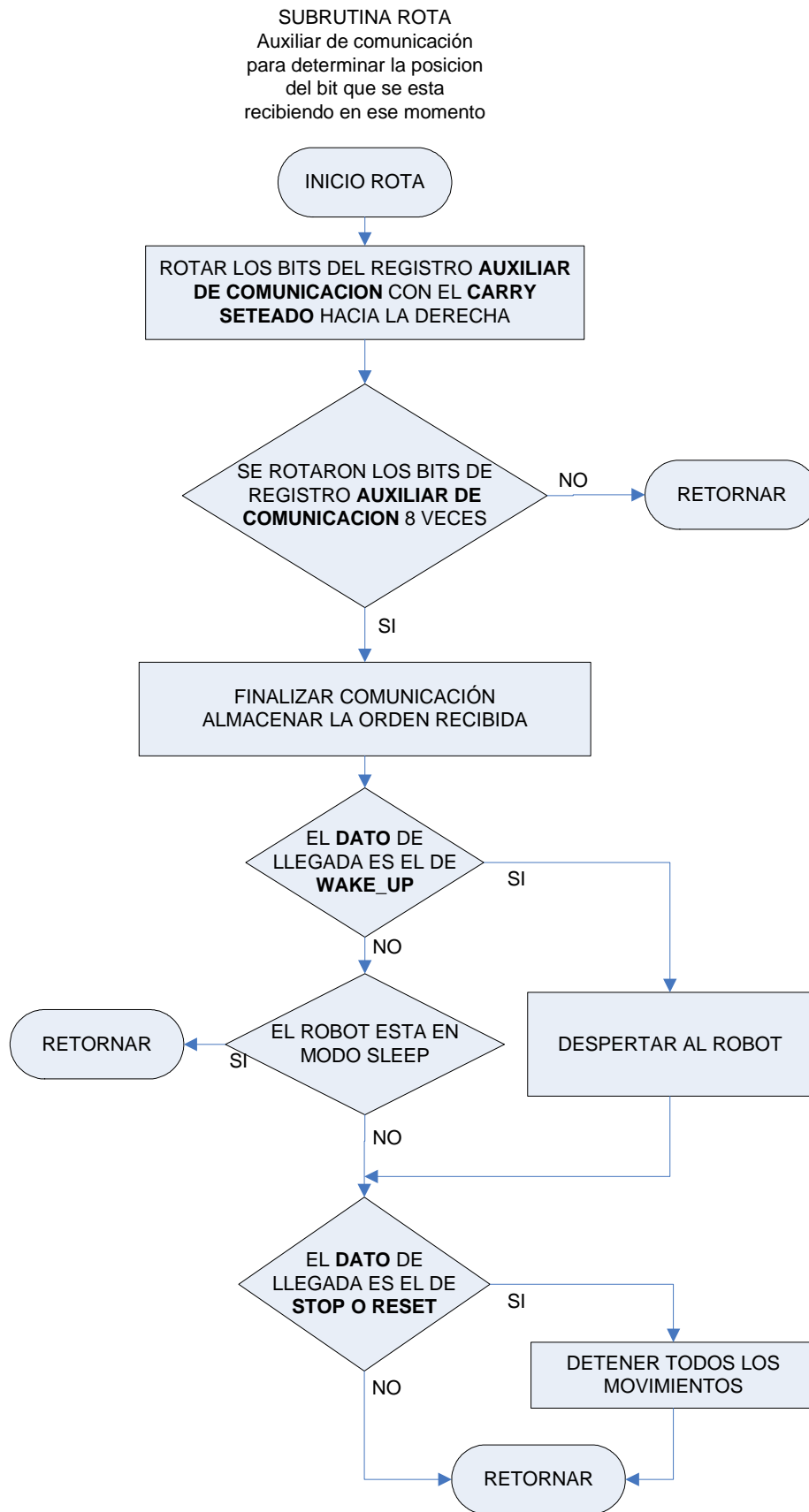


Figura 3.32 Subrutina Rota

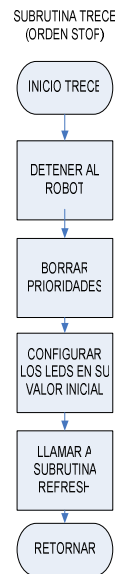


Figura 3.33 Subrutina Trece (STOP)

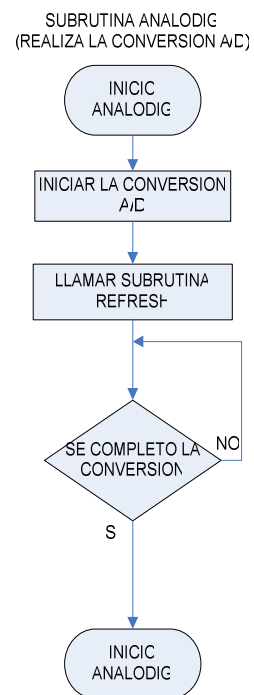


Figura 3.34 Subrutina Analog

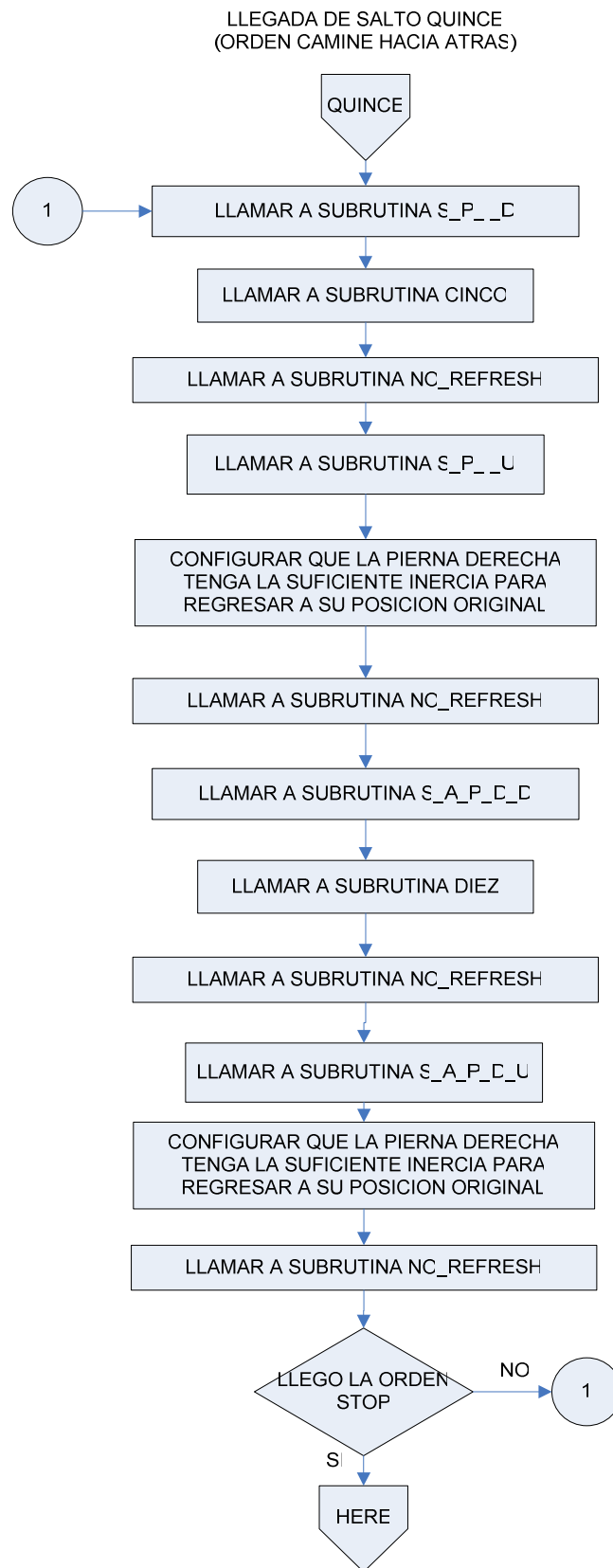


Figura 3.35 Subrutina Quince

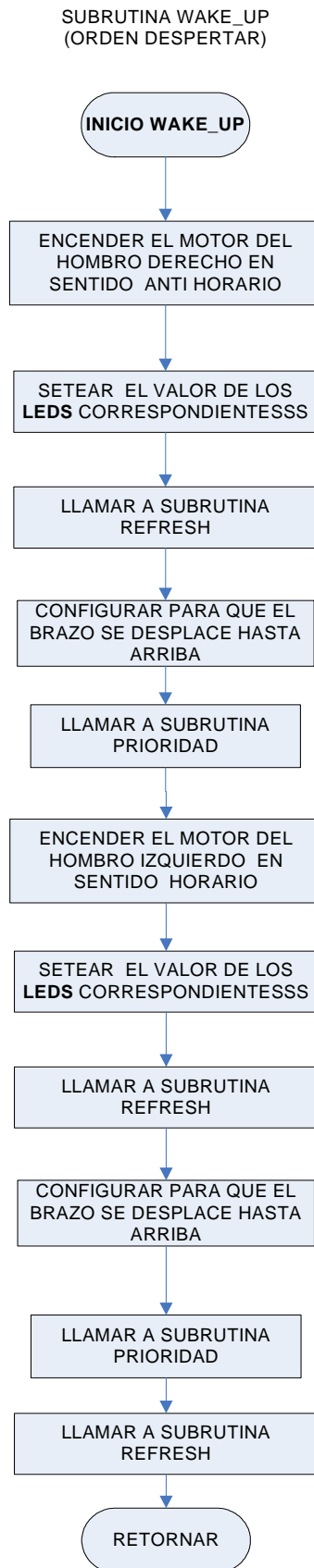


Figura 3.36 Subrutina Wake_Up

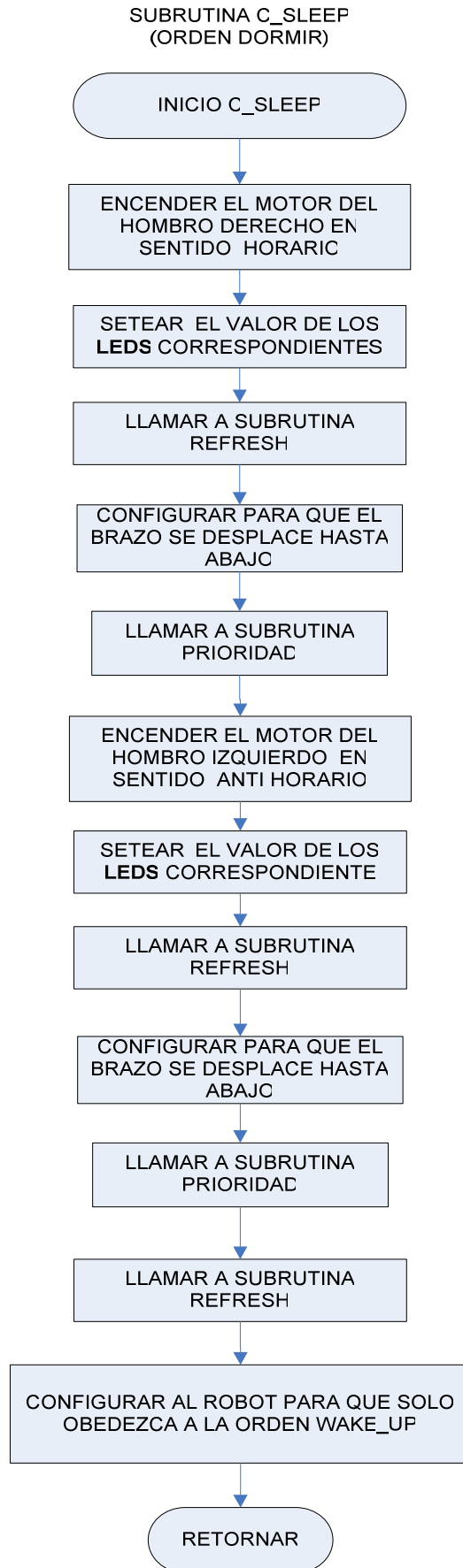


Figura 3.37 Subrutina C_Sleep

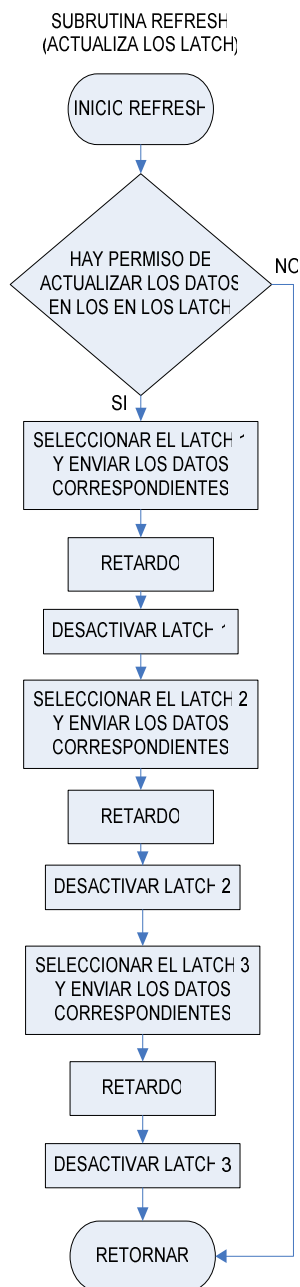


Figura 3.38 Subrutina Refresh

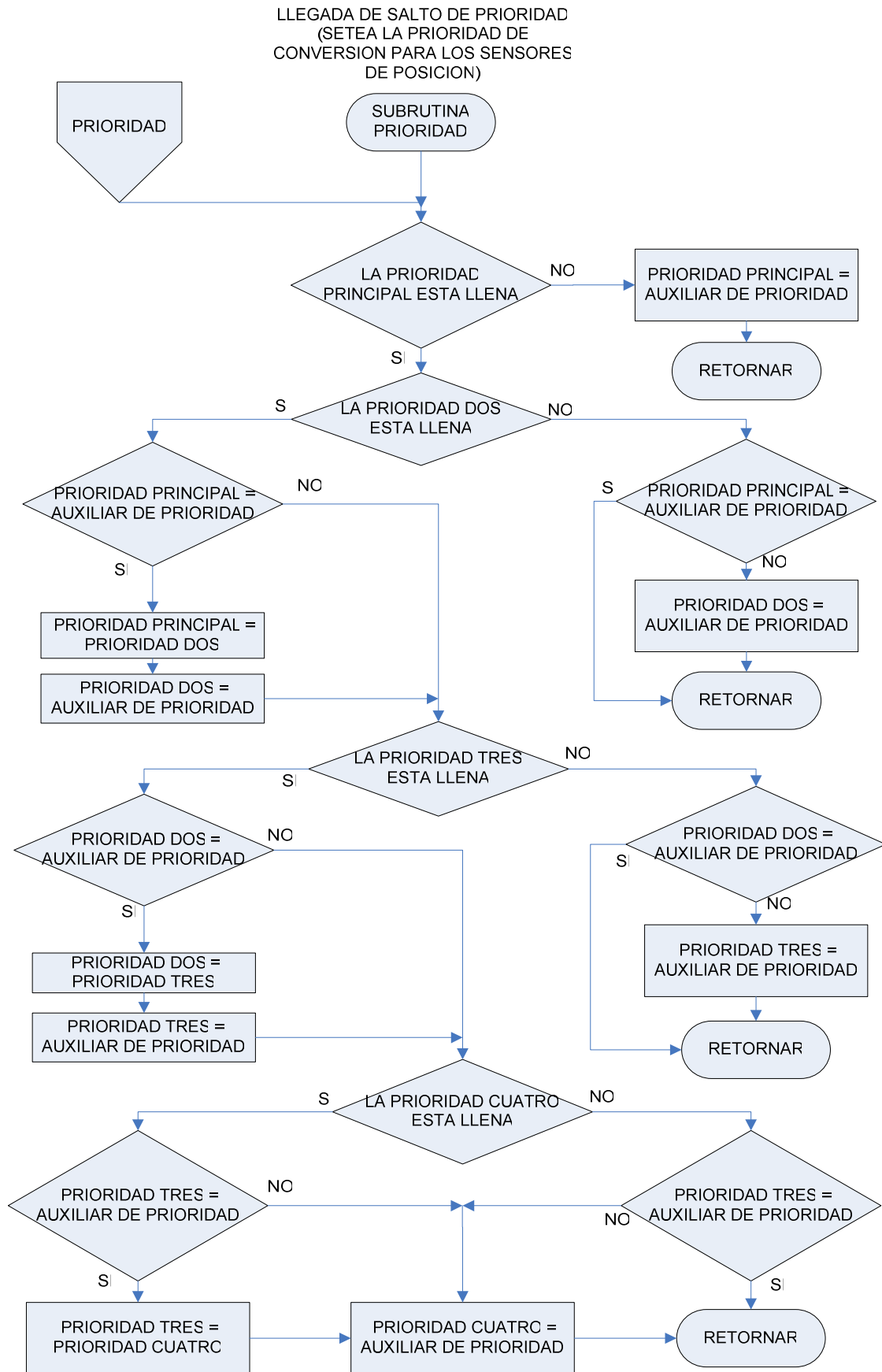


Figura 3.39 Subrutina Prioridad

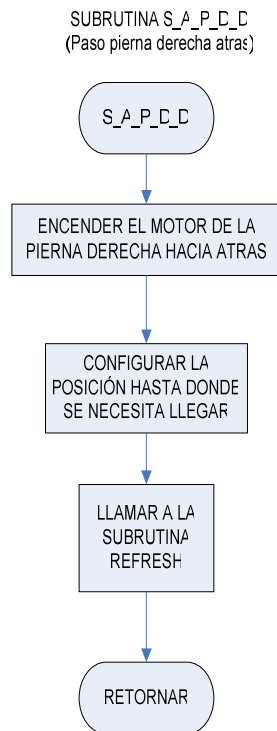


Figura 3.40 Subrutina S_A_P_D_D

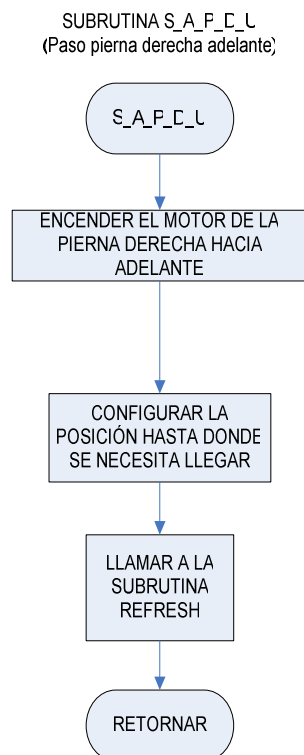


Figura 3.41 Subrutina S_A_P_D_U

SUBROUTINA S_A_P__C
(Paso pierna izquierda atrás)

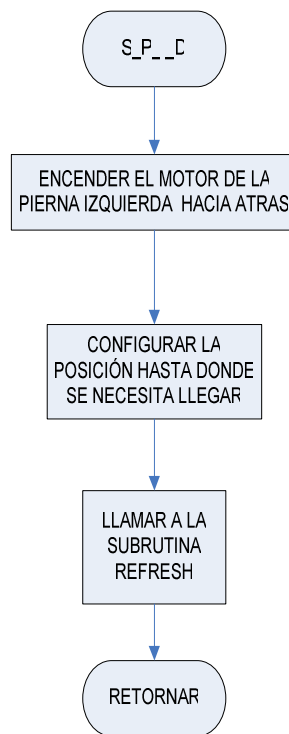


Figura 3.42 Subrutina S_A_P_I_D

SUBROUTINA S_A_F__U
(Paso pierna izquierda adelante)

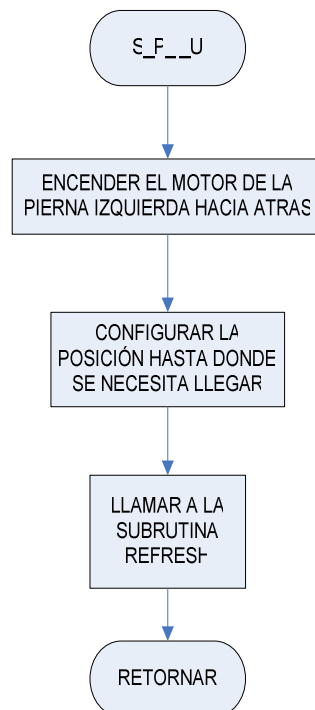


Figura 3.43 Subrutina S_A_P_I_U

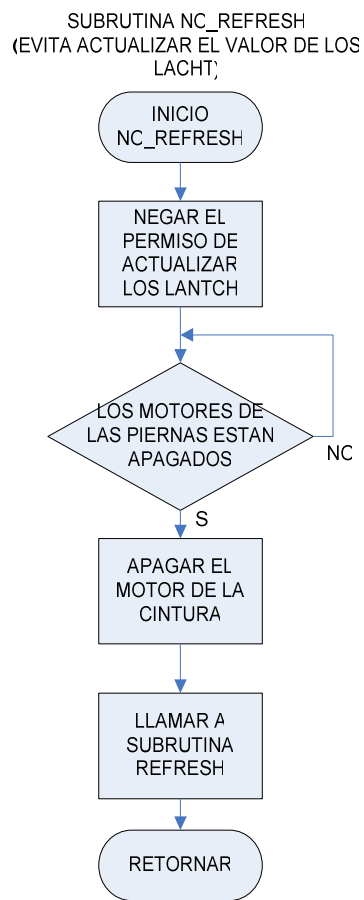


Figura 3.44 Subrutina No_Refresh

En el siguiente capítulo se detalla las pruebas realizadas en la plataforma bípeda para obtener un buen funcionamiento.

CAPÍTULO 4

PRUEBAS Y RESULTADOS

En el presente capítulo se muestran todas las pruebas desarrolladas a la plataforma robótica para determinar las condiciones necesarias para su buen funcionamiento.

4.1 PRUEBAS DEL SENSOR DE ULTRASONIDO

El objetivo de estas pruebas era determinar los tipos de materiales que pueden detectar los sensores de ultrasonido y la distancia de detección por lo que se procedió en todos los casos a poner los materiales (cuya superficie fue considerable) al frente de los sensores desde una distancia mínima hasta obtener la distancia máxima de detección.

Los materiales y los resultados obtenidos de las pruebas se detallan a continuación.

4.1.1 PLÁSTICO

La materia utilizada en esta prueba fue una funda plástica; y, todos los sensores de ultrasonido detectaron el obstáculo.

La distancia máxima de detección de los sensores de adelante fue de 33 [cm.], del sensor de la derecha la distancia fue 37.8 [cm.] y del sensor de la izquierda el valor fue de 39 [cm.]. Todas las pruebas fueron realizadas con una superficie grande del material.

4.1.2 MADERA

La madera es uno de los materiales de más fácil detección a pesar de la distancia, debido a que los sensores basan su funcionalidad en la emisión de ondas; y, al ser la madera un elemento compacto y plano (en el caso de la prueba) hizo que el eco de la onda regrese al receptor.

La distancia máxima de detección fue superior a la del plástico y fue de 41 [cm.] para los sensores delanteros y 60 [cm.] para los sensores laterales.

4.1.3 CUERO

El cuero a pesar de ser un material denso no presenta las ventajas de estructuración que posee la madera, es por ello que la distancia de localización del objeto disminuye a un valor de 39 [cm.] en los sensores frontales y en los laterales el valor fue de 55 [cm.].

4.1.4 PAPEL

El papel es otro de los materiales que se los puede detectar a una distancia considerable del robot; esta claro que para la óptima detección del material, este debe estar paralelo a todas las parejas de sensores.

Procurando respetar la consideración anterior, los sensores del pecho del robot detectaron la hoja de papel a una distancia de 42 [cm.]; y, los sensores de los costados izquierdo y derecho lo hicieron a 45 [cm.] y 47 [cm.] respectivamente.

4.1.5 VIDRIO

El vidrio es uno de los materiales más fáciles de detectar, su constitución compacta hace que el funcionamiento del sensor sea óptimo, los cuales detectan una distancia de 65 [cm.] en los sensores laterales y 42 [cm.] en los frontales.

4.1.6 ESPONJA

La esponja es de los pocos materiales que por su naturaleza, no puede ser detectado por los sensores de ultrasonido.

4.2 PRUEBAS DEL MODO MANUAL DEL ROBOT

Las pruebas del modo manual, se ejecutaron satisfactoriamente con todos los comandos enviados por el usuario a través del control remoto.

4.2.1 MANEJO DE LOS BRAZOS

La finalidad de la prueba era comprobar la ejecución de los comandos 1, 2, 6 y 7 (referido a la Tabla 2.1).

Luego del enviar estos comandos a través del control remoto, el robot levanto y bajo el brazo derecho e izquierdo. Cabe decir que el robot al levantar el brazo izquierdo no lo hace con tanta facilidad como el brazo derecho, debido a la

constitución mecánica de los brazos, ya que el brazo derecho tiene mayor facilidad de movimiento. Otra acotación importante es que el sensor de posición del brazo izquierdo, debido a su ubicación, no permite que el brazo recorra todo su trayecto, alcanzando solamente el 90% de su totalidad, lo que no ocurre en el brazo derecho que alcanza el 98%.

4.2.2 MANEJO DE LAS MANOS

Con la ejecución de las teclas 3, 4, 8 y 9 del control remoto (Tabla 2.1) se observó el funcionamiento de ambas manos del robot.

El robot, en ambas manos, tiene dos posiciones, tanto al abrir así como al cerrar, por lo que en estos comandos es necesario pulsar dos veces cada una de las teclas para la ejecución total de los mismos.

4.2.3 MANEJO DE LAS PATAS

En el manejo de las patas se buscó comprobar que el robot ejecute correctamente los comandos de caminar tanto para adelante como para atrás.

Para realizar esta prueba se pulsó las teclas 11 y 15 (Tabla 2.1), luego de lo cual el robot se coloca en una posición inicial, la misma que es muy importante para facilitar el caminar del robot. La posición inicial para caminar es brazo derecho arriba con su mano cerrada y el brazo izquierdo abajo con su mano semi-abierta. (Figura 4.1).

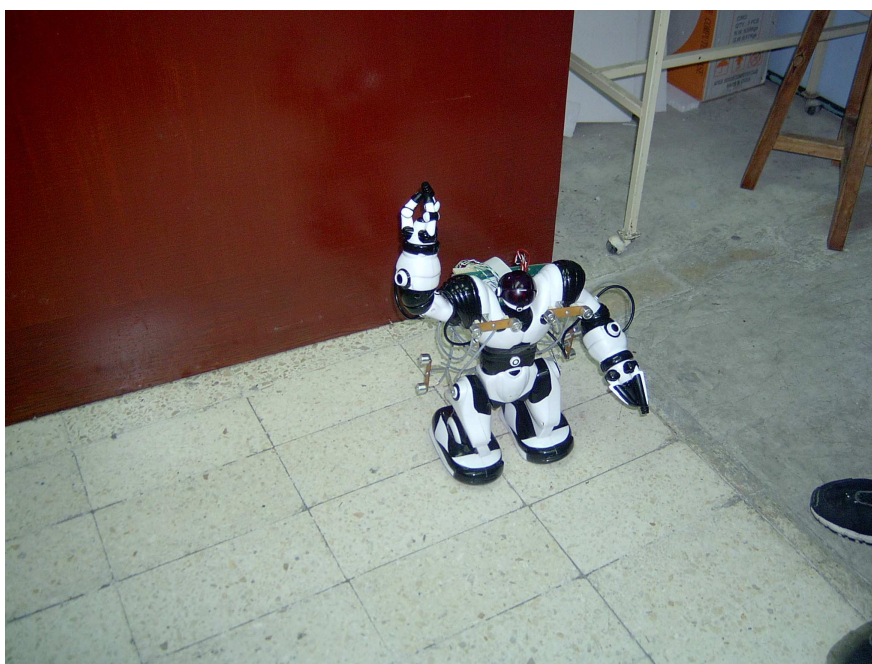


Figura 4.1 Posición inicial del robot antes de empezar a caminar

Otra de las pruebas, dentro del manejo de las patas, fue para determinar la ejecución de los comandos girar, por lo que al pulsar las teclas 12 y 14 (Tabla 2.1) el robot giró correctamente hacia la derecha e izquierda respectivamente. Para la ejecución de estas órdenes se requirió nuevamente de una posición inicial; para girar se necesitó que el brazo derecho esté abajo, la mano derecha cerrada y que el brazo izquierdo esté abajo con su mano semi-abierta.

4.2.4 MANEJO DE LA CINTURA

El anhelo de esta prueba fue determinar la ejecución de los comandos para mover el dorso del robot. Por lo que al pulsar las teclas 5 y 6 (Tabla 2.1) la plataforma bípeda movió su dorso a lo programado, a la derecha e izquierda correspondientemente.

4.3 PRUEBAS DEL MODO AUTOMATICO DEL ROBOT

4.3.1 DISTANCIA DE DETECCIÓN

El robot debe detectar obstáculos a una distancia de 25 [cm.]. Para comprobar esta hipótesis se ejecutó el modo automático del robot y se utilizó obstáculos de diferente constitución.

De estas pruebas se obtuvo la Tabla 4.1.

Tabla 4.1 Distancia de detección de objetos

Material	Distancia (cm.)
Concreto	32
Vidrio	42
Madera	39
Cuero	41

Como se puede apreciar en la Tabla 4.1 la distancia de detección en el modo automático depende del material del obstáculo y de la superficie del mismo.

4.3.2 OBJETOS DETECTADOS

En el modo automático el robot tiene cierta dificultad en detectar los obstáculos que no son superficies planas y perpendiculares a la señal de sus sensores de

ultrasonido, por lo que se realizó pruebas de detección de obstáculos específicos como: personas, pared, mesas, gradas, puertas, dando los resultados que se muestran en las Figuras 4.2, 4.3, 4.4, 4.5, 4.6 y 4.7.

4.3.2.1 Persona

Como se muestra en la Figura 4.2a y 4.2b el robot detecta a la persona por lo que se detiene a la distancia programada de los sensores de ultrasonido.



Figura 4.2a Robot detectando persona y realizando la acción evasiva



Figura 4.2b Robot alejándose de la persona

4.3.2.2 Silla

Las sillas al no tener una superficie relativamente grande (superficie de las patas) dificultó totalmente al robot detectarla como se muestra en la Figura 4.3a y 4.3b . A pesar de ello el robot se detuvo ya que los finales de carrera de los dedos tocaron el objeto lo que provocó que el robot retroceda y cambie de dirección para continuar caminando.



Figura 4.3a Robot acercándose a la silla, sin detectarla.



Figura 4.3b Robot impactándose con la silla

4.3.2.3 Mesa

Como se indica en la Figura 4.4a y 4.4b las patas de la mesa presentan la misma dificultad que la silla por lo que el robot tampoco la detectó.



Figura 4.4a Robot dirigiéndose hacia una de las patas de la mesa



figura 4.4b Choque del robot con la mesa.

4.3.2.4 Pared, vidrio y cuero

En estos tres casos de superficie el robot detectó completamente deteniéndose a las distancias detalladas en la Tabla 4.1 Todo esto se puede apreciar en las Figuras 4.5, 4.6a, 4.6b, 4.7a, y 4.7b.)



Figura 4.5 Robot deteniéndose ante una superficie de vidrio,

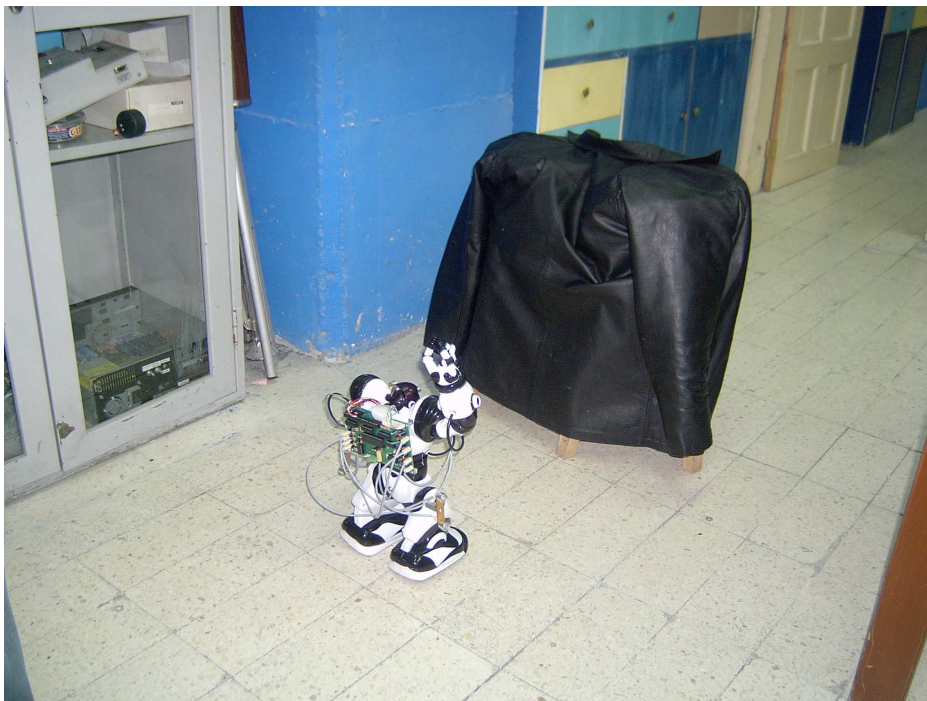


Figura 4.6a Robot detectando una superficie de cuero.



Figuras 4.6b Evasiva del robot luego de detectar obstáculo



Figura 4.7a Plataforma robótica girando hacia la derecha luego de encontrarse con una pared.



Figura 4.7b robot preparando su posición inicial para dirigirse hacia delante

Todos estos resultados se pueden comprobar cuando el elemento a detectar tiene una superficie paralela a los sensores, caso contrario el robot topa el objeto y su detención depende exclusivamente de los finales de carrera de sus extremidades.

4.3.3 MOVILIDAD DEL ROBOT

4.3.3.1 En superficie plana lisa

En este tipo de superficie el robot camina fácilmente y con relativa velocidad; los giros se ejecutan con un alto nivel de eficacia.

4.3.3.2 En superficie plana rugosa

El movimiento hacia delante y atrás funciona adecuadamente, pero presentó dificultades cuando las patas no tenían total contacto con el suelo, es decir, existían grietas considerables en el camino para el tamaño del robot.

Cabe indicar que cuando el robot realiza los giros hacia los costados no se mueve con la misma facilidad que en una superficie lisa e inclusive puede llegar a desobedecer al microcontrolador. Esto es relativamente normal ya que inclusive el ser humano, presenta dificultad en esta clase de superficie irregular.

4.3.3.3 Superficie inclinada lisa

En experimentos de caminata realizados en este tipo de superficie el robot se cae debido a la inestabilidad que presenta al mover las patas. Cabe mencionar que en superficies menores a 10° de inclinación el robot anda perfectamente.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

La versatilidad que poseen hoy en día los microcontroladores, permite realizar desde pequeñas aplicaciones hasta controles muy complejos, esto es muy evidente en el presente trabajo ya que con un solo microcontrolador se dio movimiento a toda la estructura robótica, sin dejar a un lado la comunicación con el medio externo.

El uso de elementos de interrelación entre el hombre y la máquina no es un concepto nuevo, pero aquí se ratifica que el elemento más cómodo de usar ya que es universal, es un control remoto, por lo que el manejo de la plataforma es fácil y sencilla. Lo único que es necesario conocer, son las tramas de datos emitidas por el control remoto, una vez echo esto, la decodificación del comando se la programa generando una base de tiempo adecuada y observando el estado lógico de la trama.

En el control de los motores se uso el driver LM293, el mismo que simplifico enormemente el encendido y las inversiones de giro en los motores, pero a la vez resto torque debido a que trabaja con niveles de corriente más bajos que los nominales de los motores.

El uso del modo extendido en cualquier tipo de aplicación incrementa la capacidad de manejo de periféricos ya sea de entrada o salida, en este proyecto se aumento al doble el número de salidas digitales, con lo cual se prescindió de otro microcontrolador que únicamente habría ocupado más espacio sin estar encargado de realizar funciones de importancia.

Dentro de la programación se comprobó nuevamente que el uso adecuado de las interrupciones y subrutinas hace más sencillo la elaboración del código fuente, adicionalmente se incluyo un criterio de prioridad variable en la

atención de ciertos eventos, de esta forma se logro que en hombros y patas el primer miembro en accionarse sea el primero en ser apagado.

El costo del prototipo fue relativamente bajo, todos los elementos se los encontró dentro del país, cumpliéndose uno de nuestros objetivos personales planteados “Controlar una plataforma bípeda” con elementos que se encuentren dentro del país.

El uso de sensores de posición permiten un control exacto y minucioso de cualquier estructura, en este proyecto se observa claramente la diferencia entre las articulaciones que poseen sensores de posición (hombros y piernas), finales de carrera (manos) y la que no tienen ningún tipo de sensor (cintura). Las primeras articulaciones dan seguridad total en los movimientos y se puede indicar hasta que punto se desea mover la articulación. En el caso de las manos que tienen final de carrera, la incertidumbre de la posición esta presente, pero se la puede solucionar colocando a la articulación en una posición inicial. En el caso de la cintura, al no tener un sensor o final de carrera, no se podría ejercer control de posición en esta parte de la estructura, pero la constitución mecánica del robot, permitió intuir cual es la posición estándar de la articulación, y en base a experimentación se genero bases de tiempo que dan la movilidad que se necesita.

El diseño y elaboración de las placas tomaron mucho tiempo, ya que se hicieron varias consideraciones de viabilidad, técnicas como estéticas, que al iniciar el proyecto se las desconocía, por lo que es recomendable que dentro de la carrera exista una materia que permita al estudiante construir cualquier placa con excelentes condiciones técnicas, fácilmente y en poco tiempo.

Los costos de las plataformas bípedas bordean en su mayoría 600 dólares, y se encuentran fuera del país, seria recomendable que se aplique proyectos conjuntos con la carrera de Mecánica, así se impulsaría de mejor forma la investigación no solamente en este tipo de estructuras bípedas sino en cualquier estructura robótica.

Las sugerencias para mejorar este prototipo en el futuro son:

Generar el código fuente que permita al robot realizar acciones más humanas, como por ejemplo bailar, levantar objetos, memorizar un trayecto, o incluso poder salir de un laberinto.

Crear un control remoto que sea accionado mediante la voz o el computador, y emita los comandos a la plataforma bípeda.

Realizar la simulación y visualización en tiempo real de la plataforma, observando los movimientos y reacciones desde un monitor, de esta forma se podrá comprobar los algoritmos de control antes de implementados en la plataforma.

Se considera muy importante la investigación de este tipo de robots en la Escuela Politécnica Nacional, ya que en los últimos años ha habido un auge en la investigación de humanoides debido a que su apariencia humana es un atractivo innato para las personas, y va a permitir crear cada vez más "máquinas que no son máquinas", adicionalmente lograr imitar la habilidad de caminar en el hombre es un reto muy interesante, ya que esto es el inicio de una nueva rama investigativa dentro del mundo.

REFERENCIAS BIBLIOGRAFICAS

- [1] Torres, F.; Pomares, J.; Gil, P.; Puerta, S.; Aracil, R., "Robots y Sistemas Sensoriales," Editorial Pearson Educación S.A., Madrid 2002.
- [2] GROOVER M.; WERSS M.; NAGEL R.; ODREY N., "Industrial Robotics-Technology, Programming and Applications," Editorial McGraw-Hill, 1986.
- [3] Juan José Ramos, René Alberto González Florentino y Samuel Meléndez, ROBOTEC Tecnología Robótica, <http://robotec11.tripod.com/id4.html>
- [4] Universidad de Mendoza-Argentina, Diferentes tipos de robots - clasificación general, <http://www.roboticajoven.mendoza.edu.ar>
- [5] Honda, History ASIMO, <http://asimo.honda.com/asimos>
- [6] RoboticSpot, Robótica por Ingenieros desde Madrid para España y el Mundo, <http://www.roboticspot.com>
- [7] Walter Fendt, traducción por Juan Muñoz, Ley de Ohm, http://www.walter-fendt.de/ph11s/ohmslaw_s.htm
- [8] Jameco, Sensores de Ultrasonido 136653, <http://www.jameco.com>
- [9] Cueva, J., Retroban, M., "Control y ensamblaje de un robot que emule un perro doméstico," EPN, Quito, Noviembre 2004.
- [10] Microchip Technology Inc., "PIC16F87XA Data Sheet, 28/40/44-Pin Enhanced Flash Microcontrollers," 2003
- [11] Ñiacasha, N., "Desarrollo de un Dispositivo que Mida la Distancia a un Objeto Emulando el Efecto de un Bastón Blanco para Personas Invidentes," EPN, Quito, Mayo 2004