

**UNIVERSIDAD NACIONAL DEL CALLAO**  
**ESCUELA DE POSGRADO**  
**UNIDAD DE POSGRADO DE LA**  
**FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**



**“DESARROLLO DE UN SISTEMA SENSORIAL  
AUTOMÁTICO PARA ROBOT MÓVILES BYCIBOT  
DE APLICACIÓN SOCIAL”**

**TESIS PARA OPTAR EL GRADO ACADÉMICO DE  
MAESTRO EN CIENCIAS DE LA ELECTRÓNICA  
MENCIÓN: CONTROL Y AUTOMATIZACIÓN**

**AUTOR: ZENÓN ANDRÉS CUCHO MENDOZA**

**Callao, 2018**  
**PERÚ**

{PAGINA DE RESPETO}

## HOJA DE REFERENCIA DE MIEMBROS DEL JURADO

## **DEDICATORIA**

*Dedicado con mucho amor a Dios*

*A mis padres Juan y Dionicia.*

*A mis hijos Julio Cesar y Devin Gabriel que  
son mi inspiración y mi apoyo.*

*A mi esposa Sabina por su optimismo y  
aliento constante.*

*A mi hermano Leo, en especial*

*Y a todos mis hermanos por su ejemplo de  
tenacidad y esfuerzo para salir adelante en  
los momentos difíciles.*

*A mis amigos en general.*

## **AGRADECIMIENTO**

Finalizar un trabajo de investigación requiere dedicación propia y el apoyo de muchos colaboradores por su conocimiento en diferentes campos de la ciencia y de la vida diaria que nos transmiten los cuales me han permitido desarrollar y culminar con éxito la presente tesis. Por eso mi sincero agradecimiento de manera especial al Dr. Noe Chávez Temoche, al Dr. José Dávalos Pinto, al Dr. Jacob Astocondor Villar, al Dr. Marcelo Damas Niño, al Mag. Julio Borjas Castañeda, por su valiosa colaboración en el aspecto profesional, técnico y humano.

Al Magister Ing. Ricardo Rodríguez Bustinza por su asesoramiento y contribución en el desarrollo de la tesis.

Al decano de la Facultad de Ingeniería Eléctrica y Electrónica de la Universidad Nacional del Callao Dr. Juan Herber Grados Gamarra por su valioso apoyo para la culminación de la tesis.

El Autor

# ÍNDICE

CARÁTULA	
HOJA DE REFERENCIA DE JURADO Y APROBACIÓN	
DEDICATORIA	
AGRADECIMIENTO	
ÍNDICE	1
LISTA DE FIGURAS	6
TABLAS DE CONTENIDOS	9
RESUMEN	10
ABSTRACT	12

## CAPÍTULO I

<b>I</b>	<b>PLANTEAMIENTO DEL PROBLEMA</b>	<b>14</b>
1.1.	Determinación del problema	14
1.2.	Formulación del problema	14
1.2.1	Problema general	14
1.2.2	Problema específico	15
1.3.	Objetivos de la investigación	16
1.3.1	Objetivos generales	16
1.3.2	Objetivos específicos	16
1.4.	Justificación	17
1.4.1	Beneficios sociales	17

## CAPÍTULO II

<b>II</b>	<b>MARCO TEÓRICO</b>	<b>18</b>
2.1.	Antecedentes del estudio	18
2.1.1	Historia de la robótica	18
2.1.2	Leyes de la robótica	19
2.1.3	Aparición de los robots manipuladores	20
2.1.4	Evolución de la robótica	23

2.2.	Marco Filosófico	27
2.2.1.	Fundamento ontológico	27
2.2.2.	Fundamento epistemológico	27
2.2.3.	Fundamento metodológico	28
2.3	Clasificación de los robots	29
2.3.1	Clasificación de acuerdo a su grado de autonomía	29
2.3.1.1	Robots teleoperadores	29
2.3.1.2	Robots de funcionamiento repetitivo	30
2.3.1.3	Robots autónomos inteligentes	30
2.3.2	Clasificación en base a su arquitectura	31
2.3.2.1	Robots poliarticulados	31
2.3.2.2	Robots móviles	32
2.3.2.3	Robots androides o humanoide	32
2.3.2.4	Robots zoomórficos	33
2.3.2.5	Robots híbridos	34
2.4.	Aplicaciones del robot móvil	35
2.4.1	En la limpieza	36
2.4.2	En la seguridad	37
2.4.3	En los hospitales	38
2.4.4	En la investigación planetaria y espacial	39
2.4.5	En la actividad militar	40
2.4.6	En el sector educativo	41
2.4.7	En las fábricas, en la minería	41
2.4.8	Uso doméstico	42
2.4.9	Vehículos autónomos	42
2.5	Arquitectura de un robot móvil	44
2.5.1	Sistema móvil	44
2.5.2	Fuente de energía	44
2.5.2.1	Pilas secas	45
2.5.2.2	Baterías que se emplean en la robótica	45
2.5.3	Sistema motriz	48

2.5.4	Sistema sensorial	49
2.5.4.1	Sensor ultrasónico	49
2.5.4.2	Motor servo	52
2.5.5	Sistema de potencia	55
2.5.5.1	Circuito driver del motor DC	55
2.5.5.2	Circuito del encoder	60
2.5.5.3	Circuito de protección	61
2.6	Sistemas embebidos	61
2.7	Sistemas de control	62
2.7.1	Sistema de control en lazo abierto y lazo cerrado	63
2.7.2	Clasificación de acuerdo al tipo de energía	64
2.7.3	Clasificación según la acción de control	65
2.7.3.1	Controlador ON-OFF	66
2.7.3.2	Controlador proporcional	66
2.7.3.3	Controlador integral	67
2.7.3.4	Controlador derivativo	68
2.7.3.5	Controlador proporcional integral	68
2.7.3.6	Controlador proporcional integral derivativo	69
2.7.4	Método de sintonía de controladores Ziegler-Nichols	70
2.7.4.1	Método de respuesta al escalón	71
2.7.4.2	Método de oscilación	72
2.8	Robot de servicio	73
2.9	Robot móvil Bycibot	75
2.9.1	Circuito eléctrico	76
2.9.2	Microcontrolador TM4C123GH6PM	77
2.9.3	El módulo TIVA LaunchPAD	82
2.10	IDE Code Composer Studio	84
2.11	Obtención de la onda PWM	88
2.12	Robot evalbot	89



### **CAPÍTULO III**

<b>III</b>	<b>VARIABLES E HIPÓTESIS</b>	<b>94</b>
3.1.	Definición de las variables.	94
3.2.	Relación entre las variables de investigación	94
3.3.	Hipótesis general e hipótesis específicas.	95

### **CAPÍTULO IV**

<b>IV</b>	<b>METODOLOGÍA</b>	<b>96</b>
4.1	Tipo de investigación	96
4.2	Diseño de la Investigación del robot bycibot	96
4.2.1	Diagrama de Flujo	97
4.2.2	Sistema de señalización	98
4.2.3	Trayectoria del robot	99
4.2.4	Diagrama de bloques	101
4.2.5	Diagrama esquemático del robot	101
4.2.6	Lista de componentes	103
4.2.7	Generación de ondas PWM	103
4.2.7.1	Descripción de los principales registros	104
4.2.8	Función de transferencia del sistema en lazo cerrado	110
4.3	Población y muestra	111
4.4	Técnicas e Instrumentos de recolección de datos.	111
4.5	Procedimientos de recolección de datos	111
4.6	Procesamiento estadístico y análisis de datos	111

### **CAPÍTULO V**

<b>V</b>	<b>RESULTADOS</b>	<b>112</b>
----------	-------------------	------------

### **CAPÍTULO VI**

<b>VI</b>	<b>DISCUSIÓN DE RESULTADOS</b>	<b>116</b>
6.1.	Contrastación de la hipótesis con los resultados	116
6.2.	Contrastación de resultados con otros estudios similares	116

	<b>CAPÍTULO VII</b>	
<b>VII</b>	<b>CONCLUSIONES</b>	<b>117</b>
	<b>CAPÍTULO VIII</b>	
<b>VIII</b>	<b>RECOMENDACIONES</b>	<b>118</b>
<b>IX</b>	<b>REFERENCIAS BIBLIOGRÁFICAS</b>	<b>119</b>
	<b>ANEXOS</b>	<b>121</b>
	<b>ANEXO A:</b>	<b>122</b>
	A1) PROGRAMA FUENTE EM LENGUAJE ENSAMBLADOR PARA EL MICROCONTROLADOR ATMEGA8 DE ATMEL (MICROCHIP).	
	A2) APLICACIONES UTILIZADOS CON EL MICROCONTROLADOR TM4C123GH6PM DE TEXAS INSTRUMENTS	
	A3) PROGRAMAS EN LENGUAJE C	
	<b>ANEXO B:</b>	<b>134</b>
	ANÁLISIS DEL SISTEMA CON MATLAB	
	B1) ANALISIS CON EL RETENEDOR DE ORDEN CERO	
	B2) ANALISIS DEL SISTEMA CON CONTROL PID DISCRETO	
	<b>ANEXO C:</b>	
	MATRIZ DE CONSISTENCIA	<b>138</b>
	<b>ANEXO D:</b>	<b>139</b>
	DIAGRAMAS	
	HOJAS TÉCNICAS	

## LISTA DE FIGURAS

Fig.2.1 Isaac Asimov, padre de la Robótica	20
Fig.2.2 Primer robot industrial	21
Fig.2.3 Primer manipulador maestro esclavo	22
Fig.2.4 Evolución de la robótica	24
Fig.2.5 Cobots una nueva era en la automatización industrial	25
Fig.2.6 Robots asistentes personales	26
Fig.2.7 Robots teleoperadores	29
Fig.2.8 Robots de funcionamiento repetitivo	30
Fig.2.9 Robot inteligente autónomo: son la nueva generación	31
Fig.2.10 Robots poliarticulados	31
Fig.2.11 Robots poliarticulados y móvil	32
Fig.2.12 Robot androide y robot zoomórfico	34
Fig.2.13 Robots híbridos	35
Fig.2.14 Robot manipulador móvil	36
Fig.2.15 Robots de limpieza"	37
Fig.2.16 Robots de seguridad Knightscope K5	38
Fig.2.17 Robot aplicado a la medicina	38
Fig.2.18 Robot Sojourner	39
Fig.2.19 Robot militar	40
Fig.2.20 Robot aplicado al sector educativo	41
Fig.2.21 Robot aplicado en la minería	42
Fig.2.22 Robots domésticos y vehículos autónomos	43
Fig.2.23 Pilas secas de carbono y alcalinas	45
Fig.2.24. Tipos de baterías de níquel	46
Fig.2.25 Tipos de baterías plomo/acido	48
Fig.2.26 Forma del robot "bycibot" con tres ruedas	48
Fig.2.27 Emisión de ondas ultrasónicas	49
Fig.2.28 Sensor ultrasónico HC-SR04	50
Fig.2.29 Conexión del sensor HC-SR04	50
Fig.2.30 Diagrama de tiempos del sensor de ultrasonido	51

Fig.2.31 Motores servo	53
Fig.2.32 Servomotor desmontado	53
Fig.2.33 Control de giro del servomotor	54
Fig.2.34 Control de giro del servomotor	55
Fig.2.35 Driver del motor DC L293	56
Fig.2.36 Circuito de control para el doble giro de un motor	56
Fig.2.37 Circuito de protección del Driver L293 con diodos	57
Fig.2.38 Aplicación del driver L293B	57
Fig.2.39 Aplicación del driver L293B	58
Fig. 2.40 Driver para motores L298	59
Fig.2.41 Circuito simplificado del driver L298	59
Fig.2.42 Control de sentido de giro del motor con L298N	60
Fig.2.43 El encoder óptico	60
Fig.2.44 Circuito de protección interna del driver L293	61
Fig.2.45 Diagrama de bloques de un sistema embebido	62
Fig.2.46 Sistema de control en lazo abierto	63
Fig.2.47 Sistema de control en lazo cerrado	64
Fig.2.48 Elementos de un circuito eléctrico	65
Fig. 2.49 Diagrama de bloques de un controlador ON-OFF	66
Fig.2.50 Diagrama de bloques de un controlador Proporcional	67
Fig.2.51 Diagrama de bloques de un controlador Integral	68
Fig.2.52 Diagrama de bloques de un controlador PI	69
Fig.2.53 Diagrama de bloques de un controlador PID	70
Fig.2.54 Curva de reacción del proceso	71
Fig.2.55 Lazo cerrado con ganancia proporcional	72
Fig.2.56 Respuesta de la planta con ganancia crítica	73
Fig.2.57 Robot social	74
Fig. 2.58 Robot bycibot	76
Fig.2.59 Circuito de alimentación del robot	77
Fig.2.60 Estructura del microcontrolador TM123GH6PM	78
Fig.2.61 Configuración de pines como E/S	80

Fig.2.62 Configuración del registro de control Systick	81
Fig.2.63 Módulo Tiva C series LaunchPad	82
Fig.2.64 Diagrama de bloques del módulo LaunchPad	83
Fig.2.65 Diagrama esquemático de la Tarjeta TIVA LaunchPAD	83
Fig.2.66 Puertos de E/S disponible en la tarjeta LaunchPad	84
Fig.2.67 Code Composer Studio V6	84
Fig.2.68 Creación de un Nuevo Proyecto	85
Fig.2.69 Pantalla de Inicio para programación en C	86
Fig.2.70 Botones de compilación y programación	86
Fig.2.71 Menú desplegable para crear o añadir archivos	87
Fig.2.72 Agregar archivo tm4c123gh6pm.h al proyecto	87
Fig.2.73 Copia del archivo tm4c123gh6pm.h	87
Fig.2.74 Botones para comenzar con la ejecución del programa	88
Fig.2.75 Diagrama de bloques de módulo PWM	88
Fig.2.76 Diagrama de bloques del generador del módulo PWM	89
Fig.2.77 Sistema Evalbot con microcontrolador LM3SW9B92	90
Fig.2.78 Diagrama de bloques del sistema Evalbot	91
Fig.2.79 Distribución de los pulsadores en el módulo Evalbot	93
Fig.4.1 Diagrama de flujo	97
Fig.4.2 Señalización en el robot	98
Fig.4.3 Trayectoria del robot	99
Fig.4.4 Posición y desplazamiento del robot	100
Fig.4.5 Diagrama de bloques del robot	101
Fig.4.6 Diagrama esquemático del robot	101
Fig.4.7 Diagrama del modulo driver L298	102
Fig.4.8 Módulo de potencia para los motores con L298	102
Fig.4.9 Registro RCGC2	105
Fig.4.10 Registro GPIODIR	106
Fig.4.11 Registro GPIODEN	106
Fig.4.12 Registro GPIODR8R	107
Fig.4.13 Registro GPIO AFSEL	107

Fig.4.14 Registro GPIO PCTL	108
Fig.4.15 Registro GPIO RCG0	109
Fig.4.16 Diagrama de bloques de um sistema de control discreto	110
Fig.4.17 Modelo cinemático del robot móvil	111
Fig.5.1 Robot bycibot como modelo de estudio	112
Fig.5.2 Microcontrolador TIVA TM4C123GH6PM	113
Fig.5.3 Generador de ondas PWM con el microcontrolador TIVA	113
Fig.5.4 Respuesta del sistema discreto a diferentes periodos de muestreo	114
Fig.5.5 Respuesta a una señal escalón de una señal continua y discreta	115

## **TABLA DE CONTENIDOS**

Tabla No.2.1 Sentido de giro del motor DC	58
Tabla No.2.2 Equivalencia entre un circuito eléctrico y neumático	65
Tabla N°2.3 Método de sintonía Ziegler - Nichols	71
Tabla N°2.4 Método de sintonía para el regulador PID	71
Tabla N°2.5 Registro GPIO PCTL para configuración de pines	79
Tabla N°2.6 Especificaciones de la fuente de alimentación	92
Tabla N°4.1 Distribución de pines para la obtención de 16 ondas PWM	104

## RESUMEN

En la actualidad, es una realidad que los robots pueden realizar diversas tareas que antes eran solo realizados por el hombre y esto ha ido aumentando en la industria, en el hogar, en los hospitales, entre otros lugares debido al avance de la ciencia en todas sus disciplinas.

El presente trabajo cuyo tema de tesis es “Desarrollo de un sistema sensorial automático para Robot Mviles Bycibot de aplicación social” tiene como objetivo dar inteligencia a un robot móvil para llevar objetos a su destino reiteradas veces y sorteando obstáculos dado que la acción de llevar es repetitiva durante la jornada laboral y al ser realizado por el hombre no se está aprovechando la capacidad humana para realizar tareas más importantes, asimismo con este tipo de robots se puede apoyar a personas con discapacidad a trasladar objetos.

En la parte de diseño para que el robot se desplace automáticamente, tendrá que considerarse los principios de la navegación autónoma a través de tres preguntas: ¿dónde estoy?, que es su posición actual , ¿hacia dónde quiero ir? (planeación de tareas), y ¿cómo puedo llegar allí? (planeación de movimiento), por ello se considerará en este trabajo el uso de diferentes dispositivos electrónicos para la captación y entrega de datos, como el empleo de un microcontrolador de 32 bits quien tiene altas prestaciones para procesar la información que capta del medio ambiente por donde se desplaza con un lenguaje de programación estructurada de alto nivel como es el lenguaje C, articulando adecuadamente la parte mecánica con la parte electrónica del robot para que el robot se desplace hasta llegar a su destino.

Por ello se se detalla las principales característica de cada componente del sistema.

La implementación de un sistema sensorial con dispositivos de última generación permite desarrollar robots inteligentes de aplicación social para apoyar a las personas en particular con discapacidad para mejorar su calidad de vida en el hogar o en el trabajo.

En el trabajo se muestra el diseño de cada etapa del robot, los resultados obtenidos y las recomendaciones para un nuevo diseño.



## **ABSTRACT**

At present, it is a reality that robots can perform various tasks that were previously only carried out by man and this has been increasing in the industry, at home, in hospitals, among other places due to the advance of science in all his disciplines.

The present work whose theme of the thesis is "Development of an automatic sensory system for Mobile Robot Bycibot of social application" aims to give intelligence to a mobile robot to bring objects to their destination repeatedly and bypassing obstacles since the action of carrying it is repetitive during the working day and when carried out by man, it is not taking advantage of the human capacity to perform more important tasks, also with this type of robots you can support people with disabilities to move objects.

In the design part for the robot to move automatically, the principles of autonomous navigation will have to be considered through three questions: where am I? What is your current position? Where do I want to go? (planning of tasks), and how can I get there? (planning of movement), therefore we will consider in this work the use of different electronic devices for the capture and delivery of data, such as the use of a 32-bit microcontroller that has high performance to process the information it captures from the environment through where it moves with a high-level structured programming language such as language C, properly articulating the mechanical part with the electronic part of the robot so that the robot moves

to reach its destination. Therefore, the main characteristics of each component of the system are detailed.

The implementation of a sensorial system with state-of-the-art devices allows the development of intelligent robots of social application to support people with disabilities in particular to improve their quality of life at home or at work.

The work shows the design of each stage of the robot, the results obtained and the recommendations for a new design.

## **CAPITULO I**

### **PLANTEAMIENTO DEL PROBLEMA**

#### **1.1 Determinación del problema**

En el desarrollo de la robótica móvil, se ha visto que al inicio muchos de los robots que deberían trasladar objetos utilizaron para su desplazamiento componentes mecánicos y electrónicos que no eran apropiados ya sea por su dimensión o funcionalidad, tampoco eran robots autónomos apropiados para determinadas situaciones, siendo supervisados por un operador, limitando su operatividad para trasladar objetos hacia otro lugar. En la actualidad, con el desarrollo de la ciencias y la tecnología de la información, los robots móviles de servicio social logran gran capacidad de movimiento de forma autónoma; para ello deben contar con un sistema electrónico que sea capaz de detectar su entorno a fin de sortear obstáculos y llegar a su destino, utilizando sensores de última generación que le permitirá mejorar su inteligencia para seguir una determinada trayectoria, mejorando su performance.

#### **1.2 Formulación del problema**

##### **1.2.1 Problema general:**

**¿Cómo la falta de implementación de un robot móvil de servicio social con funcionamiento automático impide mejorar la productividad del trabajador en sus jornadas diarias?**

Existen diferentes soluciones basadas en dispositivos electrónicos de última generación para lograr el movimiento automático del robot móvil de

servicio social y apoye al hombre en sus tareas cotidianas de las jornadas laborales. El presente trabajo desarrollará la implementación de un sistema sensorial en un robot móvil para su desplazamiento repetitivo llevando objetos de un lugar a otro denominado destino, mejorando la productividad del trabajo.

### **1.2.2 Problema específico:**

#### **¿Cómo la falta de un sistema sensorial afecta el desplazamiento adecuado de un robot móvil de servicio social?**

Los robots móviles de servicio social al no contar con un adecuado algoritmo de control de sus sensores no se desplazan adecuadamente y/o no llegan a su destino, esto genera demora o pérdida de tiempo, debiéndose considerar en el diseño los sensores de detección de obstáculos, sensores de presencia existentes en el mercado.

#### **¿Cómo la falta de articulación entre las distintas piezas del robot causa dificultades en el desplazamiento del robot móvil de servicio?**

Estas dificultades pueden solucionarse si se diseña de manera óptima la estructura mecánica del robot y se desarrolla un sistema electrónico para una correcta articulación de todas sus componentes.

#### **¿Cómo la falta de empleo de dispositivos electrónicos adecuados genera un mayor costo del hardware del sistema?**

Se emplea sensores para captar datos y será un microcontrolador de 8 bits quién procesará la información, luego a través de un driver se manejará los

motores DC con señal PWM, a fin de lograr reducir el costo del sistema y mejorar la eficiencia.

**¿Cómo la falta de un eficiente algoritmo de control permite que no se produzca un adecuado manejo de la información que captan los sensores?**

Diseñar la lógica del programa con algoritmos eficientes de forma estructurada permitirá un adecuado manejo del hardware asociado al robot móvil de servicio

### **1.3 Objetivos de la investigación**

#### **1.3.1 Objetivo general**

Desarrollo de un sistema sensorial automático para un robot móvil denominado Bycibot de aplicación social el cual permita trasladar objetos siguiendo una trayectoria predefinida y evadiendo obstáculos hasta llegar a su destino.

#### **1.3.2 Objetivo específico**

- Estudiar los sensores de última tecnología que se utilizan en el sistema sensorial del robot
- Implementar un algoritmo de control discreto para orientar al robot para que pueda recibir señales y tomar una decisión.
- Implementar el algoritmo de detección de obstáculos para evadirlos y seguir su trayectoria.
- Implementar el robot, bajo los estándares de calidad, seguridad y cuidado del medio ambiente.

## **1.4 Justificación**

Se ha demostrado que el empleo de un robot móvil como apoyo en muchas labores cotidianas del hombre mejora su desempeño ya que así podrá realizar otras tareas, muchos de estos robots en su implementación deben adaptarse a necesidades específicas, por ejemplo cuando el robot móvil es diseñado para apoyar a personas con discapacidad contribuirá a mejorar su calidad de vida, pero se da el caso que muchos de estos robots móviles no se adaptan a la realidad o bien porque están sobredimensionados o no son eficientes, por tanto es importante considerar en el diseño toda la estructura de los robot móviles de servicio, considerando los elementos electrónicos y mecánicos que lo componen e implementarlo con un software estructurado interrelacionando los elementos mecánicos y electrónicos que permitirán mejorar el funcionamiento del robot, en este trabajo el robot debe realizar la tarea de trasladar objetos de un lugar a otro, evadiendo obstáculos.

### **1.4.1 Beneficios sociales**

- Mejora la labor del hombre en su trabajo, al realizar el robot las tareas repetitivas, como es lo de llevar un objeto desde un lugar a otro.
- Aliviar las dificultades que tiene muchas personas con discapacidad motriz para llevar o traer objetos.
- Protección de la vida de las personas en situaciones críticas que se presentan en actividades industriales, mineras, hospitalarias, entre otros.

## CAPÍTULO II

### MARCO TEÓRICO

#### 2.1 ANTECEDENTES DEL ESTUDIO

##### 2.1.1 Historia de la robótica

La historia de la robótica tiene su origen hace miles de años, antiguamente los robots eran unas simples máquinas, luego se les denominó autómatas. Los antiguos egipcios unieron brazos mecánicos a las estatuas de sus dioses; los griegos construyeron estatuas que operaban con sistemas hidráulicos. Durante los siglos XVII y XVIII en Europa, diversos inventores y científicos de Europa, construyeron una gran variedad de muñecos mecánicos muy ingeniosos que tenían algunas características de robots que sirvieron tanto de entretenimiento a la realeza como a las masas populares. En sus "Ensayos sobre automática" publicados por primera vez en 1914, el Ingeniero Leonardo Torres Quevedo<sup>1</sup> (1852 - 1936) formula lo que será en adelante una nueva rama de la ingeniería, la automatización, construyó el primer mando a distancia para su automóvil mediante telegrafía sin hilo, construyó una máquina analógica de cálculo, tipo mecánico llamado el ajedrecista, el primer transbordador aéreo y otros inventos. En 1920, el escritor Karel Capek<sup>2</sup> (1890-1938) al estrenar su obra de teatro *R.U.R.* (siglas de Robots Universales Rossum) aparece el

---

<sup>1</sup> Torres Quevedo, Leonardo (1914). «Ensayos sobre Automática. Su definición. Extensión teórica de sus aplicaciones». *Revista de la Real Academia de Ciencias Exactas, Físicas y Naturales* (XII): 391–419.

<sup>2</sup> Karel Capek utilizó en su obra 'Rossum's Universal Robots' la palabra 'robot', (en inglés se tradujo como robot) que en checo significa trabajador forzado

término robot que deriva de la palabra “**robota**” (que en checo significa trabajo), tuvo gran éxito y repercusión en la ciudad de Praga, luego en Londres y New York, la obra alerta sobre los peligros del maquinismo: los robots acaban exterminando a sus creadores humanos. En esta novela se usó el término **robot** para referirse a unas máquinas en forma de humanoide.

El término *robótica* es creado por Isaac Asimov<sup>3</sup> (1920-1992), escritor de ciencia ficción en su libro “**I robot**”, publicado en 1950, el término **Robótica** se refiere a la *ciencia o arte relacionada con la inteligencia artificial (para razonar) y con la ingeniería mecánica (para realizar acciones físicas sugeridas por el razonamiento).*

### **2.1.2 Leyes de la Robótica**

Las tres leyes de la robótica son un conjunto de normas elaboradas por el escritor Isaac Asimov, que la mayoría de los robots de sus novelas y cuentos están diseñados para cumplir.

1. Un robot no hará daño a un ser humano o, por inacción, permitir que un ser humano sufra daño.
2. Un robot debe hacer o realizar las órdenes dadas por los seres humanos, excepto si estas órdenes entrasen en conflicto con la 1ª Ley.
3. Un robot debe proteger su propia existencia en la medida en que esta protección no entre en conflicto con la 1ª o la 2ª Ley.

---

<sup>3</sup> Asimov I. I Robot. Edhasa. 1979. Barcelona.



Estas tres leyes surgen únicamente como medida de protección para los seres humanos. Según el propio Asimov, la concepción de las leyes de la robótica quería contrarrestar un supuesto "complejo de Frankenstein", es decir, un temor que el ser humano desarrollaría frente a unas máquinas que hipotéticamente pudieran rebelarse y alzarse contra sus creadores. De intentar siquiera desobedecer una de las leyes, el cerebro positrónico del robot resultaría dañado irreversiblemente y el robot "moriría".



**Fig.2.1** Isaac Asimov, padre de la Robótica  
Referencia: Obra "cuevas de acero" Los humanos convivían con los robots

### 2.1.3 Aparición de los robots manipuladores

Es importante también conocer **¿Cuándo se originaron los robots tal como los concebimos hoy?** <sup>4</sup>

En 1948, el inventor estadounidense George Devol<sup>5</sup>, creador del primer robot industrial, patentó un manipulador programable que fue el germen del robot industrial. Desde la llegada de la II guerra mundial, a mediados de los 40, George estaba especialmente interesado en el diseño de una máquina que

<sup>4</sup> [http://platea.pntic.mec.es/vgonzale/cyr\\_0708/archivos/\\_15/Tema\\_5.1.htm](http://platea.pntic.mec.es/vgonzale/cyr_0708/archivos/_15/Tema_5.1.htm)

<sup>5</sup> [https://es.wikipedia.org/wiki/George\\_Devol](https://es.wikipedia.org/wiki/George_Devol)

fuera de fácil manejo, adaptable a su entorno y que funcionara de forma automática. Fue en 1954 cuando Devol concibió la idea de un dispositivo de transferencia programada de artículos, objetivo que lograría tras patentar, en 1948, un manipulador programable, auténtico embrión del robot industrial, y acuñar el término automatización.



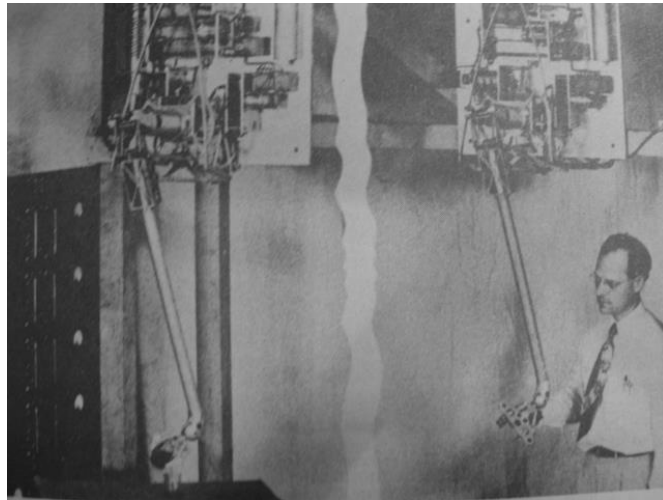
**Fig.2.2** Primer robot industrial

Referencia: Devol- Engelberger fundan Unimation (1956).

Devol predijo que el robot industrial "ayudaría al trabajador de las fábricas del mismo modo en que las máquinas de ofimática habían ayudado al oficinista".

En 1948 R.C. Goertz, del Argonne National Laboratory, desarrolló, con el objetivo de manipular elementos radioactivos sin riesgo para el operador, el primer tele manipulador denominado M1, que consistía en un dispositivo mecánico maestro-esclavo. El mecanismo de este sistema permitía que la pinza situada en el extremo del manipulador esclavo reprodujera de forma fiel los movimientos hechos por la mano del operador al extremo del manipulador maestro. Ambos manipuladores eran prácticamente iguales, y

los movimientos entre ambos se reproducían eje a eje, de tal manera que el extremo de ambos describiese la misma trayectoria.



**Fig.2.3** Primer manipulador maestro esclavo

Referencia: Fotografía correspondiente a Goertz haciendo tareas de aprehensión de objetos, 1954. Laboratorio Nacional de Argonne. USA

.En 1954 Devol concibió la idea de un dispositivo de transferencia de artículos programada que se patentó en Estados Unidos en 1961.

En 1956 **Joseph F. Engelberger** es director de ingeniería de la división aeroespacial de la empresa Manning Maxwell y Moore en Stanford, Connecticut. Juntos Devol y Engelberger comenzaron a trabajar en la utilización industrial de sus máquinas, fundando la Consolidated Controls Corporation, que más tarde se convierte en Unimation (Universal Automation), e instalando su primera máquina Unimate (1960), en la fábrica de General Motors de Trenton, Nueva Jersey, en una aplicación de fundición por inyección.

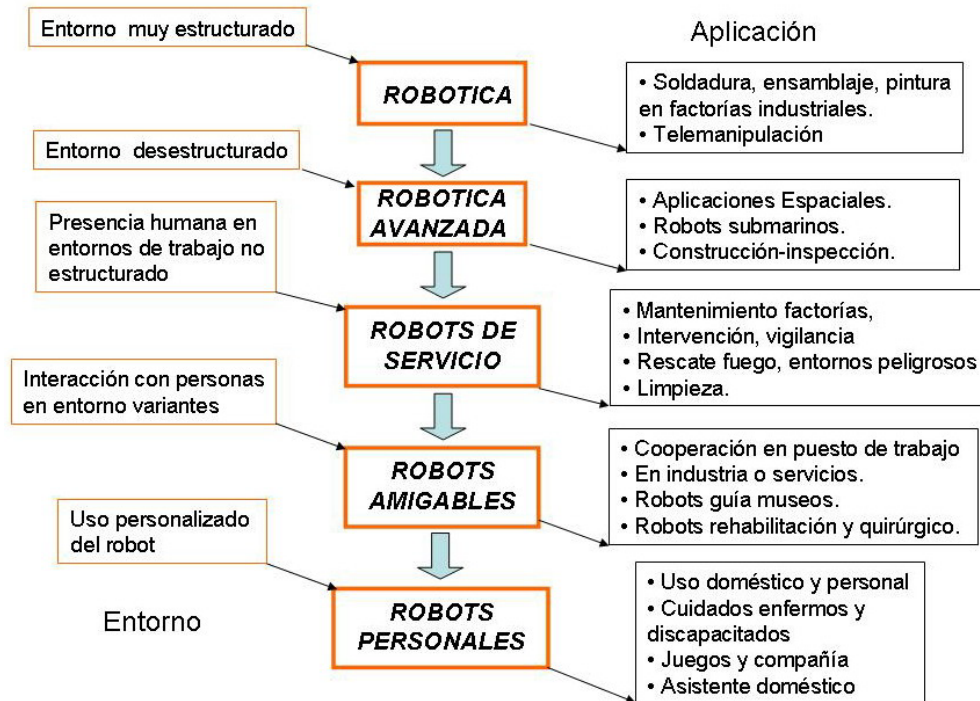
En 1968 J.F. Engelberger visitó Japón y poco más tarde se firmaron acuerdos con Kawasaki para la construcción de robots tipo Unimate. El

crecimiento de la robótica en Japón aventaja en breve a los Estados Unidos gracias a Nissan, que formó la primera asociación robótica del mundo, la Asociación de Robótica industrial de Japón (JIRA) en 1972. Dos años más tarde se formó el Instituto de Robótica de América (RIA), que en 1984 cambió su nombre por el de Asociación de Industrias Robóticas, manteniendo las mismas siglas. Por su parte Europa tuvo un despertar más tardío. En 1973 la firma sueca ASEA construyó el primer robot con accionamiento totalmente eléctrico.

#### **2.1.4 Evolución de la robótica**

La introducción de los microprocesadores desde los años 70 ha hecho posible que la tecnología de los robots haya sufrido grandes avances, se han desarrollado y tienen un "cerebro" que gobierna de manera automática sus acciones. El año 1980 fue llamado "primer año de la era robótica" porque la producción de robots industriales aumentó considerablemente ese año respecto del año anterior. Desde este año los robots se han expandido por varios tipos de industrias. El principal factor responsable de este crecimiento han sido las mejoras técnicas en los robots debidas al avance en microelectrónica e Informática. Los Estados Unidos han vendido muchas de sus empresas de robots a Europa y Japón o a sus filiales en otros países. En 1987 se fundó la [Federación Internacional de Robótica](#) (*International Federation of Robotics*, IFR) con sede en Estocolmo, Suecia.

### Evolución de la robótica



**Fig.2.4** Evolución de la robótica  
Referencia: Internet

Durante años los robots han sido considerados útiles sólo si se empleaban como manipuladores industriales. Recientemente han aparecido nuevos y variados funciones para los robots que vienen evolucionando como se muestra en la figura 2.4, a diferencia de los tradicionales robots fijos de manipulación y fabricación, estos nuevos robots móviles pueden realizar tareas en un gran número de entornos distintos y se les conoce como **robots de servicio**. Los robots de servicio se emplean en la investigación científica, en la educación, con fines de bienestar personal y social, etc. Son especialmente adecuados para el trabajo en áreas demasiado peligrosas para la vida humana o para la exploración de lugares

anteriormente vetados al ser humano. Cuando los robots de servicios tienen contacto directo con personas e interactúan con ellas, se les denomina robots asistentes. Los antecedentes de los robots asistenciales han sido los robots colaborativos o cobots. Los robots colaborativos tienen la capacidad de trabajar mano a mano con los seres humanos. Esto es posible gracias a que tienen conciencia de su entorno y a que la principal premisa en su diseño es la seguridad de los seres humanos, incorporando así la primera ley de la robótica de Asimov.



**Fig.2.5** Cobots una nueva era en la automatización industrial  
Referencia: Página web

Los robots Cobots inventados por el profesor Edward Colgate<sup>6</sup>, no pueden moverse por sí solos, por ejemplo un cobots (brazo mecánico) que ayude al hombre en las tareas de taladro y fijación. Los cobots se caracterizan por ser ligeros, flexibles y fáciles de instalar, están diseñados especialmente para interactuar con los humanos en un espacio de trabajo compartido sin necesidad de instalar vallas de seguridad, los cobots tienen como principal

---

<sup>6</sup> Edwar Colgate Profesor de la Northwestern University

característica su peso reducido entorno a los 11 Kg con una capacidad de carga de hasta 3 Kg lo cual permite minimizar las medidas de seguridad, ya que se reduce el peligro considerablemente. según el investigador Jacob Pacual<sup>7</sup>, “Ofrecen un rápido retorno de inversión, no requieren técnicos especializados para su montaje y puesta en marcha, se pueden reconfigurar para operar en diversos puntos de una línea de producción y permiten a las empresas optimizar su productividad. Representan una nueva era en la automatización industrial porque permiten la introducción de robots en sectores y procesos industriales en los que, hasta ahora, no había sido viable”.



a) PICO robot licuadora.  
Panasonic

b) Robot sirve un vaso de agua

**Fig.2.6** Robots asistentes personales  
Referencia: Página web

La evolución de estos asistentes ha dado lugar a los robots personales, dedicados a la realización de múltiples tareas, lo que supone la introducción de un “uso personalizado” del dispositivo. La robótica personal o de robots

---

<sup>7</sup> Jacob Pascual Pape, Director Sur de Europa, Universal Robots

personales, es el último escalón en la evolución de la robótica hasta nuestros días.

## **.2.2. Marco Filosófico**

### **2.2.1. Fundamento ontológico**

Desde la aparición del hombre ha buscado a lo largo del tiempo satisfacer sus necesidades básicas en alimentación, vivienda y salud, siempre ha buscado su bienestar<sup>8</sup>, reduciendo el esfuerzo físico en trabajos cotidianos e industriales con diferentes inventos y de manera impresionante aparece en nuestras vidas el robot, una máquina inteligente capaz de reemplazar al hombre en muchas tareas. En el caso particular de la salud, en los hospitales la aparición de los robots es beneficiosa siendo sus aplicaciones principales en la cirugía y terapia y hay proyectos para controlar la silla de ruedas desde el cerebro humano. Los robots móviles de servicio social ayudan a las personas discapacitadas. La convivencia con los robots es ya una realidad y los beneficios y riesgos de esta nueva tecnología están en debate. En Corea del Sur, una de las naciones más tecnificadas del planeta, el ministro de Información y Comunicaciones ha predicho que todos los hogares coreanos tendrán un robot para que les facilite las tareas diarias antes de 2020, según información periodística de El País<sup>9</sup>.

### **2.2.2. Fundamento epistemológico**

---

<sup>8</sup> Krubert(1993)

<sup>9</sup> [www.tecnologia.elpais.com](http://www.tecnologia.elpais.com)



La epistemología (del griego episteme, que significa conocimiento, logos, teoría) del presente trabajo, resalta el papel de la ingeniería en solucionar necesidades, es una ciencia aplicada y hay distinción de la ciencia pura, esta fue formulada por Mario Bunge<sup>10</sup> para utilizar leyes universales en beneficio de la humanidad. El diseño de un robot de servicio social, se basa en la segunda ley de la robótica formulado por Isaac Asimov<sup>11</sup> (1920-1992) que dice *“Un robot debe obedecer las órdenes dadas por los seres humanos, excepto si estas órdenes entrasen en conflicto con la 1ª Ley”*. Asimismo para su implementación relacionada con captar los datos del medio ambiente y realizar las tareas especificadas, se utilizará sensores de proximidad, sensores de ultrasonido, CPU de última generación como parte de un sistema embebido acoplados todos a la parte mecánica que constituye el robot móvil y pueda desplazarse adecuadamente.

### **2.2.3. Fundamento metodológico**

El presente trabajo, por ser de la rama de Ingeniería, usa como método la heurística para encontrar una solución más eficiente para apoyar en los quehaceres del hombre, en este caso particular, el robot móvil debe llevar objetos de un lugar a otro, para ello se analizará los sensores de captación y de distancia que son elementos del robot, se escogerá un microcontrolador de última generación, siendo otro de los factores el costo de los elementos y la información técnica que proporciona los fabricantes.

---

<sup>10</sup> Mario Bunge (2000). Epistemología. Buenos Aires: Siglo XXI.

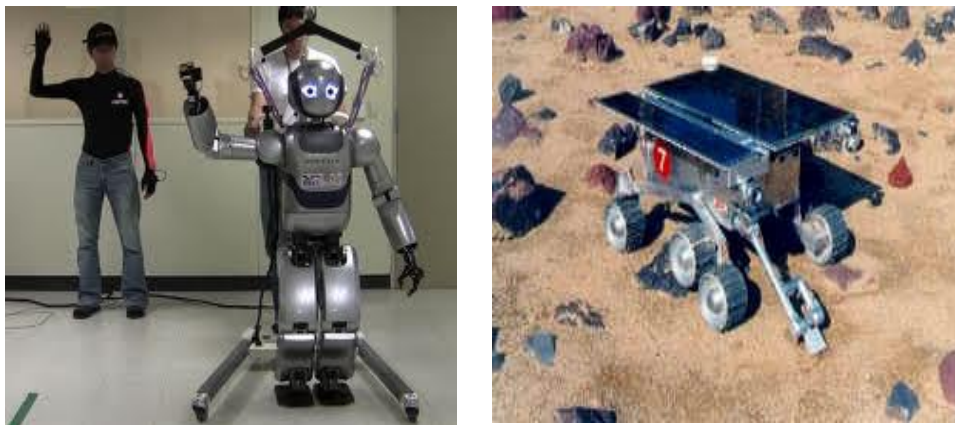
<sup>11</sup> Asimov I. Yo Robot. Edhasa. 1979. Barcelona.

Para procesar la información captada de su entorno, el programa será elaborado en lenguaje de alto nivel con algoritmos veloces para que el microcontrolador emita órdenes al robot para que se traslade de un lugar a otro. En lo que respecta a la parte mecánica podemos demostrar que la construcción de un *robot móvil* es de utilidad en los quehaceres del ser humano en todo campo. Todos estos elementos intervienen en la construcción del robot “bycibot” para desempeñarse como elemento de apoyo en las jornadas laborales.

## 2.3 Clasificación de los robots

### 2.3.1 Clasificación de acuerdo a su grado de autonomía <sup>12</sup>

**2.3.1.1 Robots teleoperadores:** Los robot teleoperados son aquellos que son controlados por un hombre a distancia desde una estación remota.



**Fig.2.7** Robots teleoperadores

Referencia: <http://todor-robots.blogspot.pe/2013/01/robots-teleoperadores.html>

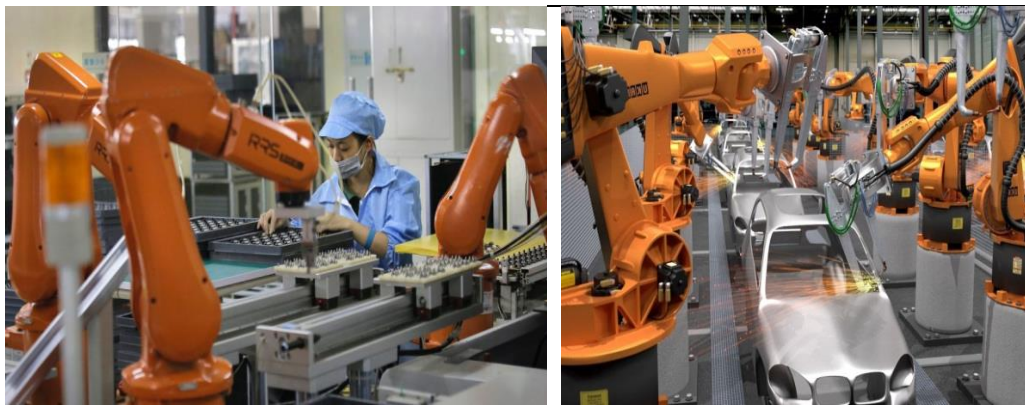
Estos robots son interesantes para trabajos en ambientes peligrosos, inestables, en aplicaciones espaciales, aplicaciones submarinas y medios

---

<sup>12</sup> Manuel Alberto Chavez Gonzales. Tesis de Maestría “Prototipo de robot móvil Teleoperado”. Instituto Politécnico Nacional Querétaro. Mexico.2012

contaminados o sustancias potencialmente peligrosos, como por ejemplo la desactivación de una bomba. El robot está controlado por medio de un programa interno en un computador.

**2.3.1.2 Robots de funcionamiento repetitivo:** Estos robots son los que se emplean mayormente en la producción industrial. Trabajan normalmente en tareas repetitivas, predecibles e invariantes con una limitada percepción del entorno. El control y la toma de decisiones lo realiza el mismo. En la siguiente figura se observa este tipo de robots que se emplean en la industria.



**Fig.2.8** Robots de funcionamiento repetitivo  
Referencia: Página web

**2.3.1.3 Robots autónomos o inteligentes:** Estos robots son máquinas capaces de percibir, modelar el entorno, planificar y actuar para alcanzar los objetivos sin la intervención o con una intervención muy pequeña de los supervisores humanos. El control de este tipo de robots es muy complejo dado el tipo de ambiente donde se desarrolla. Son sistemas dinámicos que consisten en un sistema electrónico acoplado a un cuerpo mecánico con avanzados sensores con sofisticados sistemas de control (para llevar

acciones de control cuando sea necesario). El robot tiene que reaccionar rápida e inteligentemente en un ambiente dinámico y desafiante.



**Fig.2.9** Robot inteligente autónomo: son la nueva generación  
Referencia: <https://www.tendencias21.net>

### 2.3.2 Clasificación en base a su arquitectura

**2.3.2.1 Robots poliarticulados:** En este grupo están los robots de muy diversa forma y configuración cuya característica común es la de estar básicamente en un solo lugar (aunque excepcionalmente pueden ser guiados para efectuar desplazamientos limitados) y estar estructurados para mover sus elementos terminales en un determinado espacio de trabajo según uno o más sistemas de coordenadas y con un número limitado de grados de libertad.



Robot scara de 4 ejes

**Fig.2.10** Robots poliarticulados

Referencia: Página web

En este grupo se encuentran los manipuladores, los robots industriales, y los robots cartesianos. Un robot SCARA (Selective Compliant Assembly Robot Armes) de cuatro grados de libertad con posicionamiento horizontal. Los Robots SCARA se conocen por sus rápidos ciclos de trabajo, excelente repetitividad, gran capacidad de carga y su amplio campo de aplicación.

**2.3.2.2 Robots móviles:** Son Robots con gran capacidad de desplazamiento, basada en carros o plataformas y dotada de un sistema locomotor de tipo rodante. Siguen su camino por telemando o guiándose por la información recibida de su entorno a través de sus sensores. Estos Robots aseguran el transporte de piezas de un punto a otro de una cadena de fabricación. Se explicará con mayores detalles más adelante.



a) Robot poliarticulado

b) Robot móvil

**Fig.2.11** Robots poliarticulado y movil

Referencia:Paginas web

**2.3.2.3 Robots androides o humanoides:** Son Robots que intentan reproducir total o parcialmente la forma y el comportamiento cinemática del ser humano. Actualmente los androides son todavía dispositivos muy poco evolucionados y sin utilidad práctica, y destinados, fundamentalmente, al

estudio y experimentación. Uno de los aspectos más complejos de estos Robots, y sobre el que se centra la mayoría de los trabajos, es el de la locomoción bípeda. En este caso, el principal problema es controlar dinámicamente y coordinadamente en el tiempo real el proceso y mantener simultáneamente el equilibrio del Robot. El robot Honda P3 es posiblemente el más avanzado del mundo, al poder hacer tareas únicas como subir escaleras. Su tipo es conocido como *humanoide*, debido a su aparente forma humana. Sin embargo, aunque pueda parecer que se ha obtenido un sistema completo, capaz de realizar una amplia gama de actividades. El diseño del robot y de su sistema sensorial inherente está orientado a conseguir una capacidad de desplazamiento elevada, pero los brazos han de ser controlados de forma remota. En él pueden distinguirse cuatro tipos de sensores: los *táctiles*, situados en las manos y en los pies, que le permiten controlar la fuerza de agarre de los objetos o de pisada, un sensor giroscópico en la cintura para controlar y detectar los movimientos de las piernas, un inclinómetro que les permite conocer la posición de su "columna vertebral", una cámara localizada en la cabeza, que le permite desplazarse en el entorno mediante técnicas de visión.

**2.3.2.4 Robots zoomórficos:** Los Robots zoomórficos, se caracterizan principalmente por sus sistemas de locomoción que imitan a los diversos seres vivos. En esta clasificación también se podría incluir a los androides. Se clasifican en dos categorías principales: caminadores y no caminadores. El grupo de los Robots zoomórficos no caminadores está muy poco

evolucionado. Los experimentados efectuados en Japón basados en segmentos cilíndricos acoplados axialmente entre sí y dotados de un movimiento relativo de rotación.



a) Robot androide



b) Robot zoomórfico

**Fig.2.12** Robot androide y robot zoomórfico  
Referencia: Página web

Los Robots zoomórficos caminadores múltipedos son muy numerosos y están siendo experimentados en diversos laboratorios con vistas al desarrollo posterior de verdaderos vehículos piloteando o autónomos, capaces de evolucionar en superficies muy accidentadas. Las aplicaciones de estos Robots serán interesantes en el campo de la exploración espacial y en el estudio de los volcanes.

**2.3.2.5 Robots híbridos:** Son aquellos cuya estructura se sitúa en combinación con alguna de las anteriores ya expuestas, bien sea por conjunción o por yuxtaposición. Por ejemplo, un dispositivo segmentado articulado y con ruedas, es al mismo tiempo uno de los atributos de los Robots móviles y de los Robots zoomórficos. De igual forma pueden considerarse híbridos algunos robots formados por la yuxtaposición de un

cuerpo formado por un carro móvil y de un brazo semejante al de los Robots industriales.



**Fig.2.13** Robots híbridos  
Referencia: Pagina web

## **2.4 Aplicaciones de la robótica móvil<sup>13</sup>**

La robótica móvil se considera actualmente como un área de tecnología avanzada que sirven de base para el avance en diversos campos de la industria, aportando soluciones tecnológicas innovadoras orientadas al desarrollo de mejores robots y a la ampliación del abanico de aplicaciones disponibles.

Existen muchos tipos de robots móviles y su configuración depende del tipo del medio en que se muevan (terrestres, marinos o aéreos) así como las condiciones del entorno (entorno estructurado o no estructurado). Los campos de aplicación abarcan la prospección, vigilancia, rescate y mantenimiento de estructuras hasta tareas tan diversas como la exploración espacial.

La robótica móvil es por tanto un área multidisciplinaria que agrega conocimientos muy diversos entre los que se incluye la programación, la

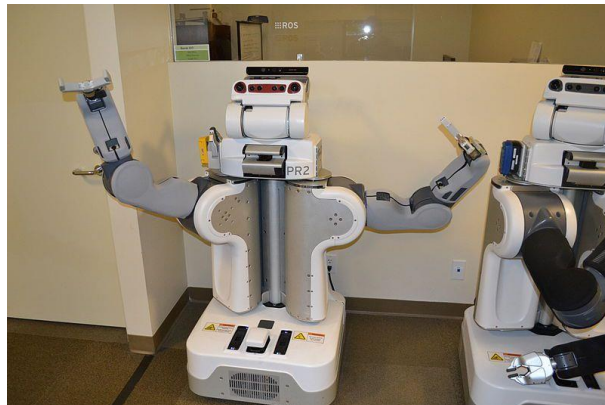
---

<sup>13</sup> <https://www.researchgate.net/>



electrónica, el control, la sensorización, la visión artificial, la generación de mapas, inteligencia artificial, etc.

Los primeros proyectos en robots móviles pueden ser encontrados en los inicios de los 70's, cuando se empieza a experimentar la interacción entre procesadores y sensores en sus bases móviles.



**Fig.2.14** Robot manipulador móvil  
Referencia: Internet

Se presenta a continuación las principales aplicaciones en el campo de la robótica móvil:

**2.4.1 En la limpieza:** Las primeras aplicaciones fueron en el campo de la limpieza; aún en el mercado actual se sigue aprovechando el potencial de robots que aspiran y limpian. En 1991 una compañía dedicada a comercializar robots móviles, la *Denning Mobile Robots y Windsor Industries*, produjo el *Roboscrub*, un robot que aspira y limpia pisos de grandes capacidades. El equipo se programó para seguir un patrón a través del espacio por limpiar, percibiendo la presencia de obstáculos para no tropezar con ellos.



**Fig.2.15** Robots de limpieza  
Referencia: Internet

Este robot, como parte de sus características, tiene incorporados transductores tipo sonar, infrarrojos y switches, y también algoritmos de navegación autónomos.

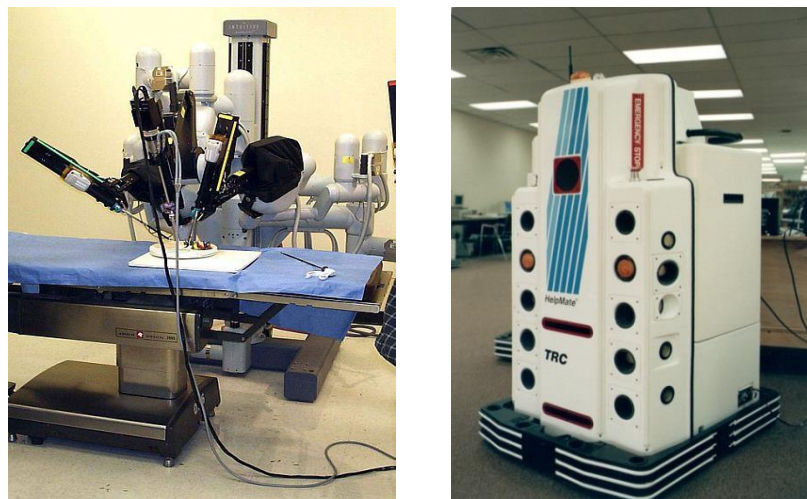
**2.4.2 En la Seguridad:** Los robots están destinados para vigilancia y seguridad tanto en el hogar como en las oficinas, centros comerciales. Un robot de seguridad fue presentado por Uber, la conocida red de transporte privado, cuenta con un nuevo vigilante en su aparcamiento de Mission Bay, San Francisco (Estados Unidos). El robot **Knight Scope K5**, tiene múltiples cámaras de alta definición que le otorgan una visión de 360 grados, una cámara térmica, un telémetro láser, un sensor meteorológico, una cámara de reconocimiento de matrículas, cuatro micrófonos, y capacidad de reconocimiento de personas. Una vez puesto en marcha en una zona geo-localizada, «deambula en busca de anomalías». K5 también tiene sistemas de inteligencia artificial para activar alarmas en caso de «ver» situaciones irregulares. Su tecnología permite leer número de

matrícula de coches para cruzar esos datos con registros de vehículos robados.



**Fig.2.16** Robots de seguridad Knightscope K5  
Referencia: Internet

**2.4.3 En los hospitales:** Se presentan los robots con aplicaciones principalmente en el campo de la cirugía, en las que se hace uso de la precisión de múltiples brazos robóticos para asistir al cirujano en las operaciones. Pueden controlarse también a distancia.



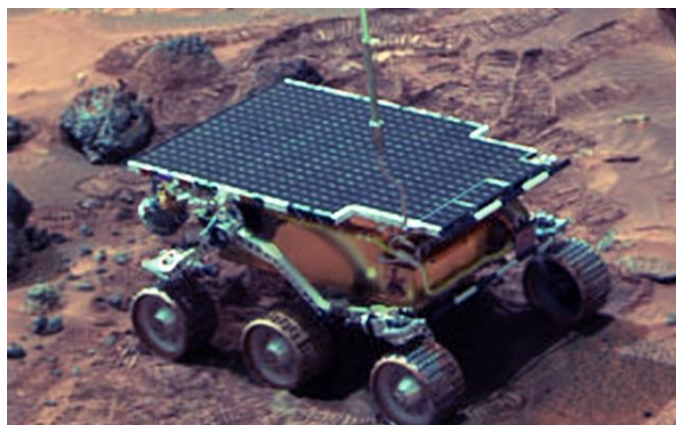
a) Robot cirujano

b) Robot Helpmate

**Fig.2.17** Robot aplicado a la medicina  
Referencia: Internet

El *Helpmate* es un robot enfermero fabricado por *Helpmate Robotics Inc.* Recorre los corredores de los hospitales combinando el sistema de navegación y la ubicación por sonar. Presionando un botón, el operario puede enviar el robot a determinado lugar del mapa y navegar de una estación programada a otra. El equipo incorpora sonares para evitar obstáculos, y un sistema de triangulación por luz estroboscópica.

**2.4.4 En la Investigación planetaria y espacial:** La misión Pathfinder, que viajó a Marte en 1997, llevó el que se puede considerar el robot móvil autónomo de mayor éxito en la historia. El robot *Sojourner* de 6 ruedas, concentró la atención de millones de personas en su exploración del planeta rojo. Ejecutó fielmente los comandos enviados desde controladores en el laboratorio de Propulsión a Cohete en California; también podía tomar acciones por sí mismo, manteniéndose alerta de obstáculos y abismos. El equipo fue el resultado del desarrollo de varios prototipos tempranos, y fue programado de acuerdo a los principios del control por comportamientos.



**Fig.2.18** Robot Sojourner

Referencia: [https://en.wikipedia.org/wiki/Sojourner\\_\(rover\)](https://en.wikipedia.org/wiki/Sojourner_(rover))

La NASA diseñó y construyó dos robots gemelos, los Errantes para la exploración marciana "Spirit" y "Opportunity". Estos Errantes fueron lanzados al espacio el 2003 y actuarían como robots geólogos mientras estén en la superficie de Marte. Estos robots están equipados con un brazo robótico, una herramienta para taladrar, 3 espectrómetros, un cuello y cabeza para sostener a las cámaras al mismo nivel del suelo que la vista humana, cuatro pares de cámaras que les permite tener a los robots una visión 3D del terreno como los humanos, además de un sistema de comunicaciones (antenas para "hablar" y "escuchar").

**2.4.5 En la actividad militar:** Otro de los robots más interesantes es el *Fletch* diseñado con el propósito de remover las pequeñas bombas y granadas de la munición antitanque que no explotan en el campo de batalla. La munición es soltada desde aeronaves y se calcula que el 25% de ella no explota, lo que genera un grave problema. La tarea de remoción es perfecta para un robot; en la evaluación se demostró que ellos pueden deshacerse incluso de explosivos de artillería.



**Fig.2.19 Robot militar**  
Referencia: Internet

El robot tuvo éxito en su navegación autónoma a un punto, en la búsqueda de munición en espiral, y en su localización, levantamiento y transporte a un punto. También se emplea en los “campos de batalla del futuro”.

**2.4.6 En el sector educativo:** La investigación y desarrollo de sistemas contribuye a la generación de conocimiento. Específicamente, los sistemas LEGO han desarrollado la robótica en todo el sentido de la palabra; ellos brindan información, elementos y partes que permiten crear cualquier sistema, constituyéndose en un apoyo educativo muy fuerte a partir de la utilización potencial del software, piezas mecánicas, sensores, sistemas de desarrollo y demás elementos necesarios para que el estudiante construya sus propios desarrollos asimilando el conocimiento requerido.



**Fig.2.20** Robot aplicado al sector educativo  
Referencia: Internet

**2.4.7 En las fábricas, en la minería:** Los robots móviles se emplean en la automatización de fábricas, donde los vehículos robóticos son usados para transportar componentes entre maquinas distantes y sitios de ensamble. Se utilizan para operaciones en ambientes peligrosos, incluyendo el

despliegue de robots móviles en excavación minera, y el uso de robots navegando autónomamente dentro de instalaciones nucleares para propósitos de inspección.



**Fig.2.21** Robot aplicado en la minería  
Referencia: Internet

**2.4.8 Uso doméstico:** Se presentan como robots aspiradores, cortadores de gras, guardianes del hogar y asistentes personales para el adulto mayor entre las aplicaciones más cotidianas, ver Fig.2.2 a y Fig.2.2 b.

**2.4.9 Vehículos autónomos<sup>14</sup>:** Un vehículo autónomo no tiene conductor, es un vehículo capaz de imitar las capacidades humanas de manejo y control. Con el objeto de poder tomar decisiones acertadas, el coche necesita recopilar toda la información disponible en su entorno. Para ello, los coches autónomos cuentan con una serie de sensores, radares, cámaras de video, GPS entre otros, que permiten la captación de información. El primer vehículo autónomo más antiguo fue presentado por

---

<sup>14</sup> [https://es.wikipedia.org/wiki/Vehículo\\_autónomo](https://es.wikipedia.org/wiki/Vehículo_autónomo)Singapur

General Motors en 1939. En el 2016, en Singapur, los primeros taxis (6) sin conductor de la Cia. UnTonomy comenzaron a circular en forma experimental en una zona de 4 Km<sup>2</sup> para recoger y dejar los pasajeros en paraderos establecidos. La automotriz estadounidense General Motors (GM) quiere tener circulando estos taxis autónomos desde 2019.



a) Robot aspiradora



b) Cortadora de césped



c) Modelo Tesla, funciona en autopista desde 2016



d) Primer taxi autónomo en Singapur

### Fig.2.22 Robots domésticos y vehículos autónomos

Referencia: Internet

De la misma manera, la compañía nipona Robot Taxi ha comenzado las pruebas de un servicio de taxis sin conductor para residentes en Fujisawa (sur de Tokio) con el objetivo de que estos vehículos de conducción automática operen en los Juegos Olímpicos de Tokio 2020 con miles de taxis a lo largo de la ciudad.



## 2.5 Arquitectura de un robot móvil

### 2.5.1 Sistema móvil

Debido a que el robot móvil debe realizar muchas tareas simultáneamente, se identifican numerosas actividades que resuelven problemas específicos. Según Iyengar<sup>15</sup>, se puede clasificar estas actividades en tres áreas conceptuales principales:

- **Percepción**, el cual comprende tareas como interpretación de sensores, integración de sensores, modelado del mundo real, y reconocimiento;
- **Planeamiento y control**, el cual desarrolla tareas de más alto nivel como planeamiento, prioridad de ejecución de tareas, y monitoreo de ejecución de todas las actividades robóticas.
- **Actuación**, el cual comprende actividades de navegación, planeamiento detallado de movimiento y acciones, y control de actuadores.

### 2.5.2 Fuente de energía

La fuente de energía para los robots autónomos son las baterías, estas deben suministrarle la suficiente potencia a todo el sistema para su buen funcionamiento a lo largo de su trayectoria. Una batería es un elemento eléctrico que transforma energía química en energía eléctrica, y viceversa. A esta transformación se la denomina proceso electroquímico y se conocen como pilas que se explican a continuación.

---

<sup>15</sup> Autonomous mobile robots. Sitharama Iyengar .1991

### 2.5.2.1 Pilas secas<sup>16</sup>:

- a) **Pila de carbono-zinc (Zn/C)** Basadas en la oxidación del zinc en medio ligeramente ácido, están compuestas por el zinc metálico, cloruro de amonio y dióxido de manganeso. Son utilizadas en aparatos sencillos y de poco consumo de corriente.



**Fig.2.23** Pilas secas de carbono y alcalinas  
Referencia: Internet

- b) **Pilas alcalinas o de zinc-dióxido de manganeso (Zn/MnO<sub>2</sub>):** La diferencia con la pila seca es el electrolito utilizado, en este caso, hidróxido de potasio, en vez de cloruro de amonio, y el zinc está en polvo. Son las de larga duración, tienen una mayor densidad de energía que las de carbono, menos tóxicas y su temperatura de trabajo es más amplia.

### 2.5.2.2 Baterías que se emplean en la robótica<sup>17</sup>

- a) **Batería de Niquel-Cadmio (Ni-Cd):** Se componen de un conjunto de elementos individuales (o celdas) conectados en serie, cada uno de los cuales tiene, si está cargado, un voltaje nominal, que oscila entre 1,2 y 3,6 Voltios. Una batería recargable de Ni-CD (o NiCd) está formada por una

<sup>16</sup> [https://es.wikipedia.org/wiki/Pila\\_de\\_zinc-carbono](https://es.wikipedia.org/wiki/Pila_de_zinc-carbono)

<sup>17</sup> [http://robots-argentina.com.ar/Energia\\_baterias.htm](http://robots-argentina.com.ar/Energia_baterias.htm)

placa positiva de hidróxido de níquel y una placa negativa de hidróxido de cadmio



**Fig. 2.24.** Tipos de baterías de níquel

Referencia: [http://robots-argentina.com.ar/Energia\\_baterias.htm](http://robots-argentina.com.ar/Energia_baterias.htm)

Ambas placas están separadas por un electrolito, compuesto por una solución acuosa de potasio cáustico, contenida dentro de un tejido poroso. El efecto memoria de una batería de Ni-Cd se produce como consecuencia de sobrecargas repetidas y/o descargas parciales. El fenómeno reduce la capacidad de las baterías con cargas incompletas. Se produce porque se crean unos cristales en el interior, a causa de una reacción química indebida. Para prevenirlo se debe hacer una carga completa en cuanto sea necesario.

**b) Batería de níquel e hidruro metálico (Ni/MH):** La batería de níquel e hidruro metálico (o Ni/MH) es un tipo de batería recargable similar a una de níquel-cadmio (Ni-Cd) pero que no contiene cadmio, un metal caro y dañino para el medioambiente.

Las baterías de níquel e hidruro metálico tienden a tener una mayor capacidad que las Ni-Cd y sufren bastante menos el efecto memoria. Las baterías de níquel e hidruro metálico son más amigables con el medio

ambiente. Pueden almacenar un 30% más de energía que una de Ni-Cd equivalente, y por tanto, la carga dura más tiempo.

c) **Batería de iones de litio (Li-Ion):** Las baterías Li-Ion poseen una elevada densidad de energía, acumulando una carga mayor por unidad de volumen. Por esta razón tienen menor peso en relación a baterías de otro tipo de la misma capacidad. Se presentan en placas rectangulares, de poco espesor, de menos de 0.5 cm, lo que las hace especialmente interesantes para integrarlas en dispositivos portátiles con poco espacio.

Presentan un alto voltaje por celda; cada unidad proporciona 3.6 voltios, lo mismo que tres celdas de Ni-Cd (1.2 V cada una). Carecen de efecto memoria. Su descarga es lineal, es decir, que durante toda la descarga el voltaje de la batería apenas varía, lo que evita la necesidad de circuitos reguladores. (Se debe tener en cuenta que esto puede ser una desventaja en algunos casos, ya que hace difícil averiguar el estado de carga de la batería. Las desventajas de estas baterías de litio son: sólo tienen una vida útil de 3 años. Se pueden cargar entre 300 y 600 veces, menos que una batería de Ni-Cd o Ni/MH.

d) **Baterías plomo/ácido:** Sus elementos constitutivos son pilas individualmente formadas por un ánodo de plomo, un cátodo de óxido de plomo y ácido sulfúrico como medio electrolítico. Son las más utilizadas por su bajo coste. Pueden ser encontradas en una gran variedad de tamaños rectangulares. Su mayor inconveniente es la aparición de fugas de ácido si la batería no mantiene su posición vertical. Para eliminar este problema

aparecieron las baterías de plomo/gel, aunque estas si padecen de efecto memoria.



- a) Batería de iones de litio (Li-Ion):                      b) Baterías plomo/ácido

**Fig.2.25** Tipos de baterías plomo/ácido

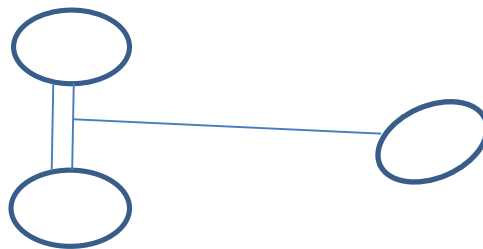
Referencia: [http://robots-argentina.com.ar/Energia\\_baterias.htm](http://robots-argentina.com.ar/Energia_baterias.htm)

### 2.5.3 Sistema motriz

El sistema motriz del robot está compuesto de los siguientes elementos:

- a) Motores de corriente continua con encoders:
- b) Ruedas
- c) Piezas de sujeción y acoplamiento

El movimiento del robot se lleva a cabo mediante tres ruedas, dos ruedas traseras fijas y una delantera como rueda motriz y directriz, configurada como se indica en la siguiente figura:



**Fig. 2.26** Forma del robot "bycibot" con tres ruedas

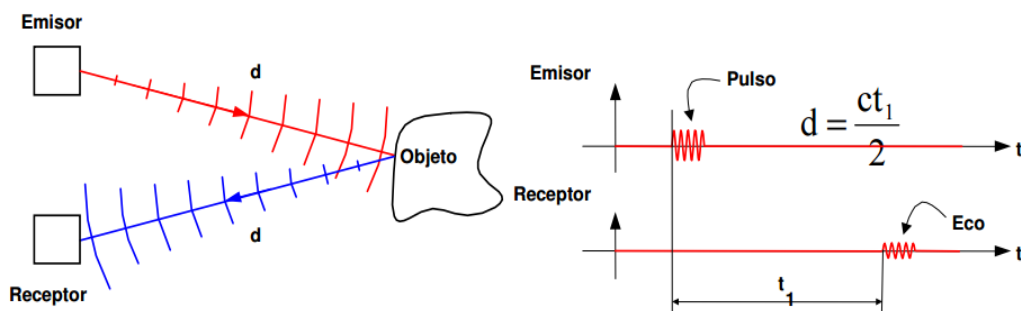
Referencia: Autoría Propia

## 2.5.4 Sistema sensorial

El sistema sensorial está compuesto por un sensor de ultrasonido a fin de detectar algún obstáculo que se presenta en su trayectoria utilizando un motor servo para hacer girar el sensor ultrasónico con un ángulo predeterminado y por los sensores de posición que le entregan señales al microcontrolador en su trayectoria.

### 2.5.4.1 Sensor ultrasónico

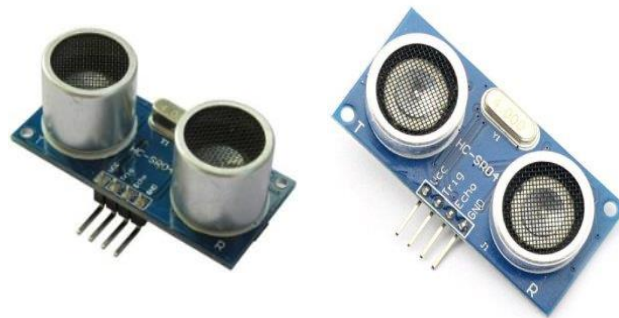
Un sensor ultrasónico es un sensor detector de proximidad, mide la distancia mediante el uso de ondas ultrasónicas. El cabezal emite una onda ultrasónica y recibe la onda reflejada que retorna desde el objeto detectado, para luego ser convertido en señales eléctricas. Los sensores ultrasónicos miden la distancia al objeto contando el tiempo entre la emisión y la recepción.



**Fig. 2.27** Emisión de ondas ultrasónicas

Referencia: [www.jameco.com](http://www.jameco.com)

En el mercado existen diversos modelos de sensores ultrasónicos, se eligió el **HC-SR04**, que es un módulo que incorpora un par de transductores de ultrasonido (Transmisión y Recepción) que se utilizan de manera conjunta para determinar la distancia entre el sensor y el objeto detectado en su trayectoria, como es nuestro caso.

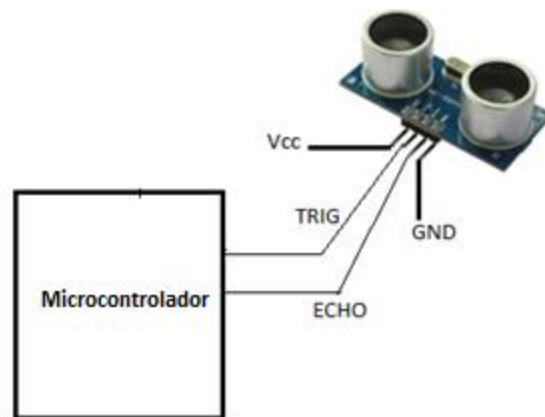


Pines del ultrasonido:

- Vcc
- Trigger (disparo)
- Echo (eco)
- GND

**Fig.2.28** Sensor ultrasónico HC-SR04  
Referencia: [www.jameco.com](http://www.jameco.com)

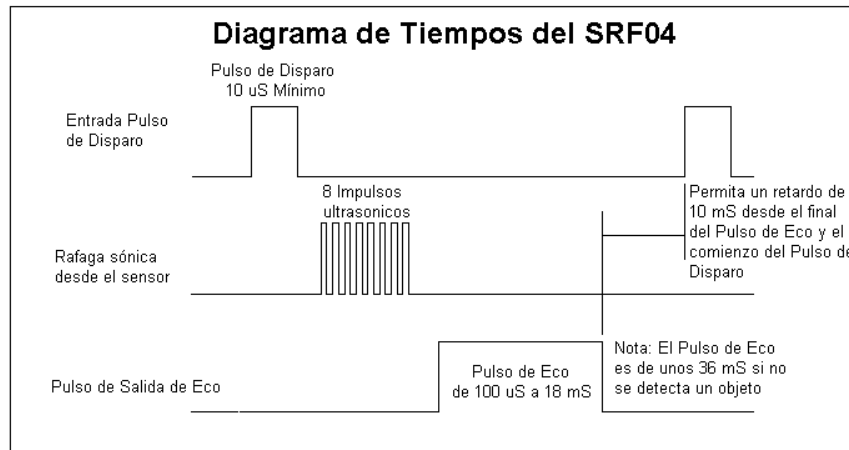
La interfaz digital del sensor ultrasónico con el microcontrolador se logra mediante los dos pines digitales: el pin de trigger (disparo) y echo (eco), cuyo funcionamiento se describe a continuación:



**Fig.2.29** Conexión del microcontrolador con el sensor HC-SR04  
Referencia: Autoría propia

- a) Se envía un pulso digital desde el microcontrolador (con un periodo mínimo de  $10\mu\text{s}$ ) al pin del disparador (Trig) del sensor ultrasónico para habilitar la medición de distancia.

- b) El sensor ultrasónico enviará 8 Pulsos de 40KHz y coloca su salida **echo** a un nivel alto, al detectar este evento se inicia un conteo de tiempo.



**Fig.2.30** Diagrama de tiempos del sensor de ultrasonido

Referencia: [www.jameco.com](http://www.jameco.com)

Por lo tanto:

Los ocho impulsos ultrasónicos demoran un periodo de:

$$8 \times (1/40K) = 8(0.025 \text{ ms}) = 200 \text{ us.}$$

La salida **“echo”** se mantendrá en nivel alto hasta recibir el eco reflejado producto de la detección del objeto (es decir se genera un ancho de pulso de tiempo T) colocándose el pin **“echo”** a un nivel bajo.

- c) La distancia es proporcional a la duración del pulso y se calcula con la siguiente formula (velocidad del sonido C= 340m/s):

$$d = \frac{CT}{2}$$

Dónde:

d= distancia a medir

C= velocidad del sonido igual a 340 m/s

T= tiempo total (salida y regreso)



Antes de comenzar a detectar un nuevo obstáculo, se recomienda dar un tiempo de aproximadamente 50 ms de espera después de terminar la cuenta.

d) Características del sensor de distancia ultrasónico: HC-SR04

- Fuente de alimentación Vcc de 5 volts
- Interfaz sencilla: Solamente 4 hilos Vcc, Trigger, Echo, GND
- Rango de medición: 2 cm a 400 cm
- Corriente de alimentación: 15 mA
- Frecuencia del pulso: 40 Khz
- Apertura del pulso ultrasónico: 15°
- Señal de disparo: 10µs

#### **2.5.4.2 Motor servo<sup>18</sup>**

Un servomotor (también llamado servo) es un dispositivo similar a un motor de corriente continua que tiene la capacidad de ubicarse en cualquier posición dentro de su rango de operación, y mantenerse estable en dicha posición. Un servomotor es un motor eléctrico que puede ser controlado tanto en velocidad como en posición. Un servo tiene un conector de 3 hilos, 5V (rojo), GND (negro o marrón) y el otro de control (amarillo o blanco). El motor servo está conformado internamente por un motor, una caja reductora y un circuito de control, que se observa en la siguientes figuras.

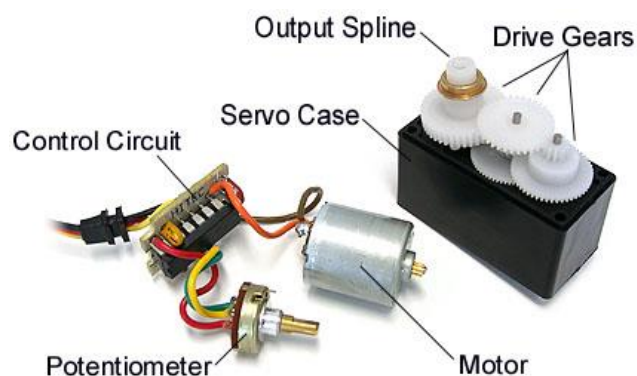
---

<sup>18</sup> <https://es.wikipedia.org/wiki/Servomotor>



**Fig.2.31** Motores servo

Referencia: <http://www.tetrixrobotics.com/>



**Fig. 2.32** Servomotor desmontado.

Referencia: <http://www.prometec.net/servos/>

La corriente que requiere depende del tamaño del servo. Normalmente el fabricante indica cuál es la corriente que consume. La corriente depende principalmente del torque y puede exceder un amperio si el servo está enclavado. Habitualmente el servo tiene un margen de operación, es decir, pueden moverse entre  $0^\circ$  y ángulo dado, que suele ser de  $180^\circ$ , pero existen modelos comerciales de todas las características imaginables (incluyendo servos de  $360^\circ$ ). Normalmente estos pequeños servos funcionan sobre 5V y el control se realiza mediante una señal de control PWM, en la que el ancho el pulso indica el ángulo que deseamos adopte el eje. Los servomotores hacen uso de la modulación por ancho de pulsos

(PWM) para controlar la dirección o posición de los motores de corriente continua. La mayoría trabaja en la frecuencia de los 50 Hz, así las señales PWM tendrán un periodo de veinte milisegundos.

En la siguiente figura, se indica el comportamiento del servomotor para diferentes anchos de pulso que se aplican.

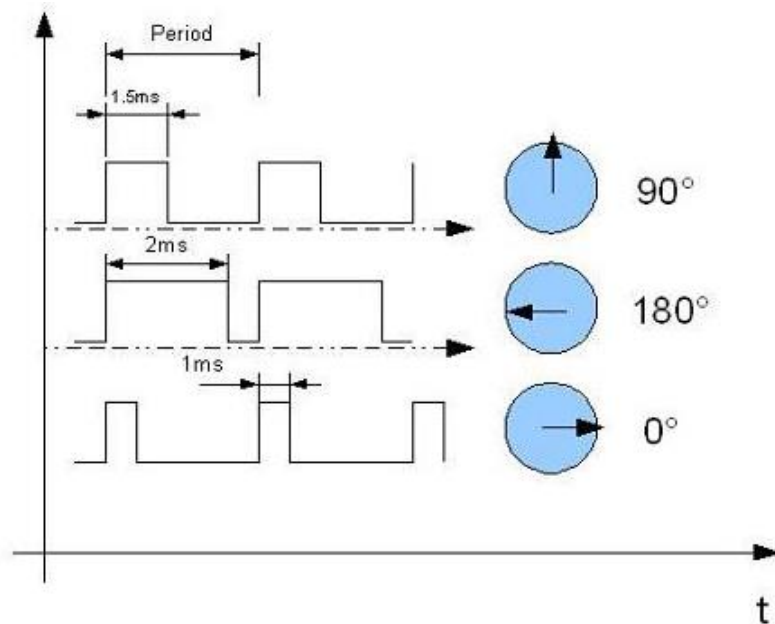


**Fig.2.33** Control de giro del servomotor

Referencia: <http://www.superrobotica.com/Servosrc.htm>

Si los circuitos dentro del servomotor reciben una señal con un ancho de pulso menor a 1.5 milisegundos, éste se moverá en sentido horario; y cuando el ancho de pulso se encuentre más de 1,6 milisegundos el servomotor se moverá en sentido antihorario; 1,5 milisegundos representan un estado neutro para los servomotores estándares. Otra particularidad que tiene este pulso es su frecuencia de refresco, que en este caso es de 50 HZ, lo que equivale a mandar un pulso de control cada 20 ms. A continuación, se tiene otro ejemplo de aplicación en la siguiente Figura 2.34, Si se mantiene el ancho del pulso a 1,5ms el motor pasa de la posición inicial a la intermedia (giro de 90°). Si se mantiene el pulso a 2,5ms el motor

gira a la posición final (giro de 180°). Si se mantuviera el pulso más tiempo de 2,5ms el motor no giraría más, ya que el potenciómetro del eje detectaría que está en la posición final, y normalmente sonaría un zumbido para indicarnos que está al final del recorrido.



**Fig.2.34** Control de giro del servomotor

Referencia: <http://www.areatecnologia.com/electricidad/servomotor.html>

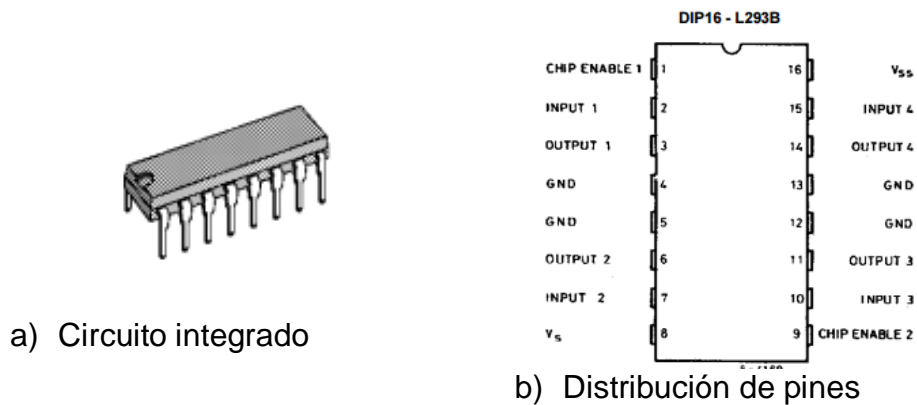
## 2.5.5 Sistema de potencia

### 2.5.5.1 Circuito driver del motor DC

Un microcontrolador que manejará los motores DC por su puerto de salida necesita de circuitos amplificadores de corriente (driver) para obtener corrientes altas para hacer funcionar apropiadamente los motores DC, por ello existen diversos circuitos integrados de potencia, entre los más conocidos tenemos el driver L293B y el driver L298 cuyo funcionamiento de cada uno de ellos se describe a continuación:

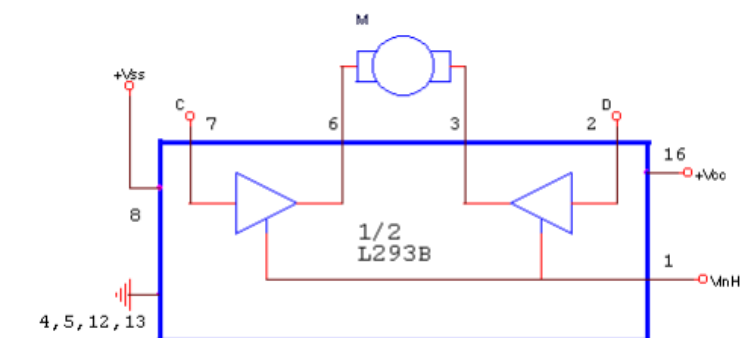
El driver L293 B, presenta las siguientes características:

- Maneja hasta 4 motores de forma independiente sin inversión de giro y puede controlar 2 motores con inversión de giro.
- Acepta como alimentación externa para los motores hasta 36 V.
- Estos drivers incluyen diodos de protección interno
- Maneja 1 amperio en cada canal de salida.



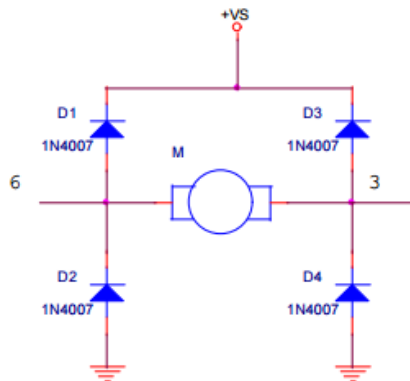
**Fig.2.35** Driver del motor DC L293  
Referencia: Internet

Un ejemplo con este circuito integrado, se observa en la siguiente figura el control de giro de un motor en los 2 sentidos.



**Fig.2.36** Circuito de control para el doble giro de un motor  
Referencia: Internet

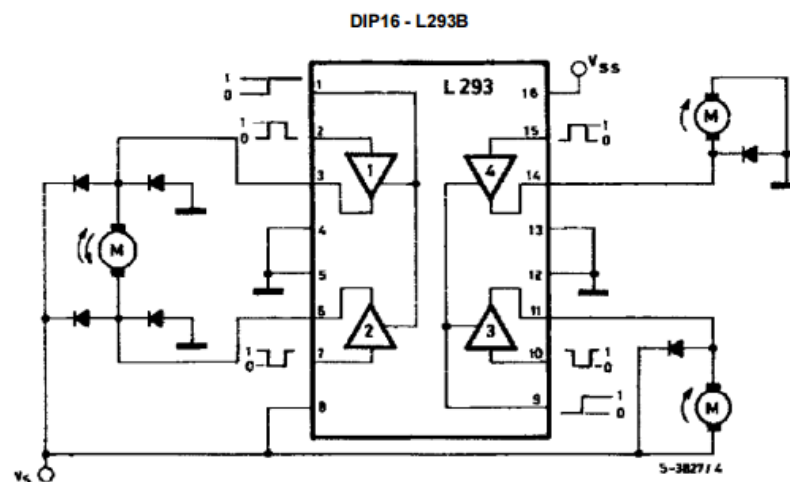
Si se quiere proteger el circuito contra posibles picos de corriente inversa cuando se arranca el motor, se conectará los diodos tal y como se muestra en la siguiente figura.



**Fig.2.37** Circuito de protección del Driver L293 con diodos

Ref: [http://www.datasheetcatalog.com/datasheets\\_pdf/L/2/9/3/L293B.shtml](http://www.datasheetcatalog.com/datasheets_pdf/L/2/9/3/L293B.shtml)

En el siguiente diagrama, se observa una aplicación del driver L293B, donde se observa el control de giro en un sentido de 2 motores DC y el control de un motor para que gire en dos sentidos horarios y antihorarios. También se observa los diodos de protección para el motor.



**Fig.2.38** Aplicación del driver L293B

Ref.: [http://www.datasheetcatalog.com/datasheets\\_pdf/L/2/9/3/L293B.shtml](http://www.datasheetcatalog.com/datasheets_pdf/L/2/9/3/L293B.shtml)

Mediante los pines 2 y 7 de controla el sentido de giro del motor M1 y con los pines 10 y 15 el motor M2. Los pines 1 y 9 activan o desactivan cada puente H de manera independiente, como se indica en la siguiente tabla.

Tabla No.2.1  
Sentido de giro del motor DC

Enable pin1	Pin2	Pin7	Sentido de giro
1	0	0	Freno libre
1	0	1	Derecha
1	1	0	Izquierda
1	1	1	Freno brusco

Ref.: [http://www.datasheetcatalog.com/datasheets\\_pdf/L/2/9/3/L293B.shtml](http://www.datasheetcatalog.com/datasheets_pdf/L/2/9/3/L293B.shtml)

A continuación se muestra una aplicación para controlar el sentido de giro de un motor c, usando el driver L293B.

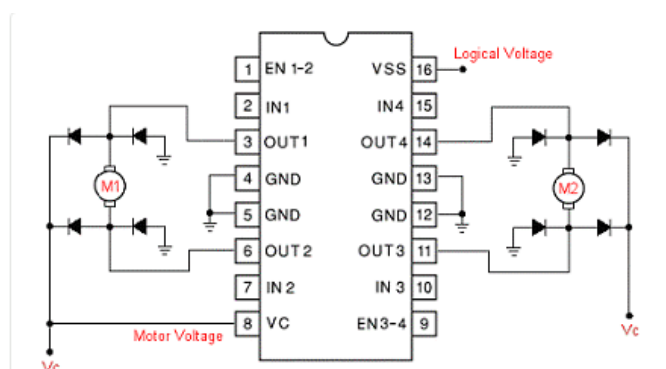


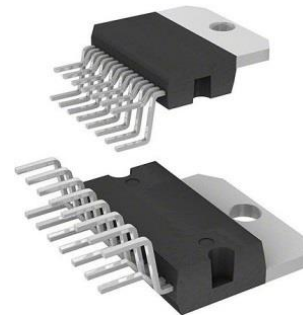
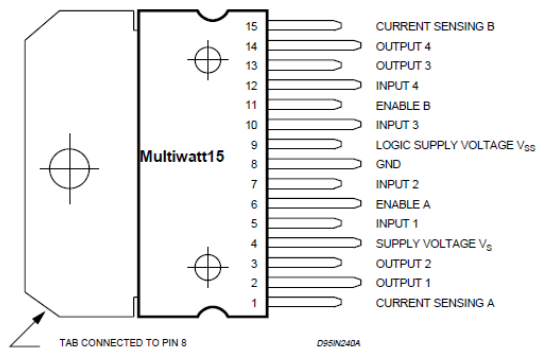
Fig.2.39 Aplicación del driver L293B

Referencia: <http://tecnodacta.com.ar/gira/tag/puente-h/>

El circuito L298 de SGS-Thomson, tiene las siguientes características:

- Es un controlador de motor de puente completo doble de alta tensión y alta corriente. El dispositivo está diseñado para aceptar cargas inductivas de accionamiento y lógica TTL estándar en relés, solenoides, motores dc y paso a paso.
- Corriente dc total de hasta 4 A

- Dispone de dos entradas de activación que permiten activar o desactivar el L298 independientemente de las señales de entrada.
- Tensión de alimentación de funcionamiento hasta 46 V.
- Protección contra sobre temperatura
- Tensión de entrada lógica "0" a 1,5 V (alta inmunidad al ruido)

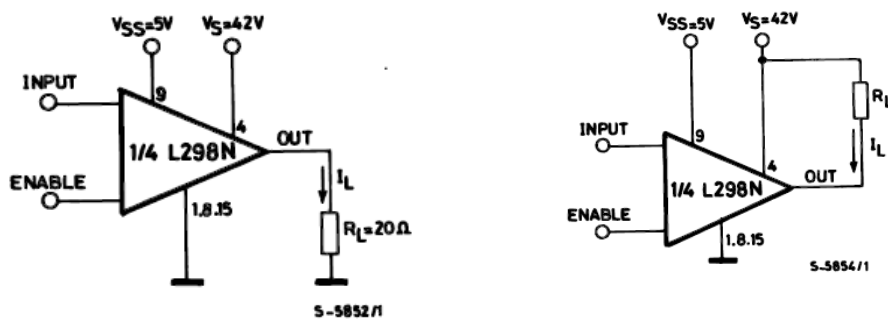


a) Tipo de montaje y pinado

b) Forma física

**Fig. 2.40** Driver L298  
Referencia: Data sheet del driver L298N

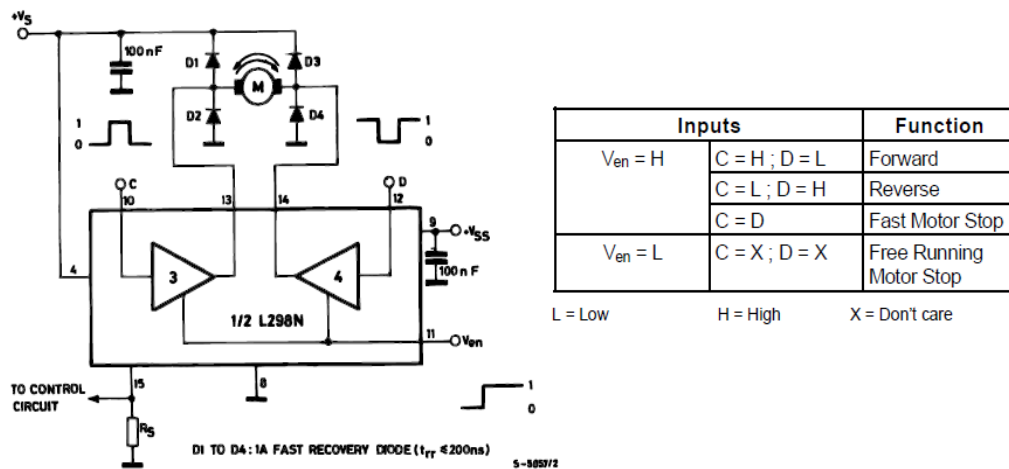
El circuito simplificado se muestra a continuación:



**Fig.2.41** Circuito simplificado del driver L298  
Referencia: Data sheet del driver L298N

Una circuito de aplicación para controlar el sentido de giro del motor en sentido horario o antihoario se muestra en la siguiente figura.

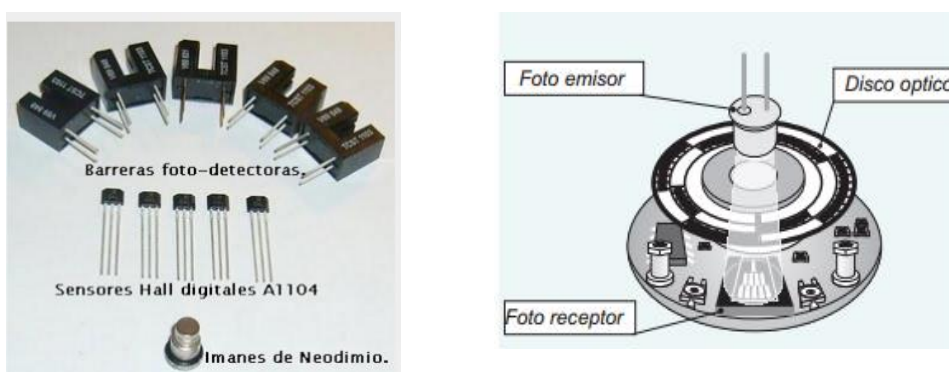




**Fig.2.42** Control de sentido de giro del motor con L298N  
 Ref.: datasheet del driver L298N.

### 2.5.5.2 Circuito del encoder

Es el circuito encargado de medir la posición en la que se encuentra el motor. Existen dos tipos de encoder: el absoluto y el incremental, el funcionamiento de ambos es similar, se trata de un disco que gira con zonas transparentes y opacas que interrumpe un haz de luz captado por fotoreceptores que transforma los impulsos luminosos en impulsos eléctricos para luego ser enviados a la salida.

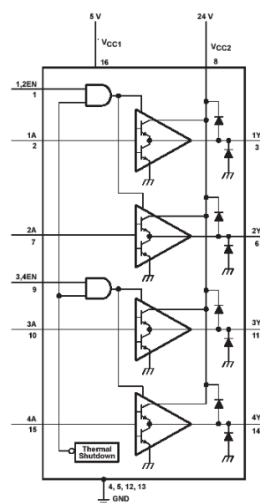


**Fig.2.43** El encoder optico

Referencia: <https://sites.google.com/site/proyectosroboticos/encoder>

### 2.5.5.3 Circuito de protección

El circuito L293 tiene sus salidas adecuadas para un manejo directo de cargas inductivas tales como relés, solenoides, motores de corriente continua y motores por pasos, ya que incorpora internamente los diodos de protección de contracorriente para cargas inductivas, como se observa en la siguiente figura.



**Fig.2.44** Circuito de protección interna del driver L293  
Referencia: [http://robots-argentina.com.ar/MotorCC\\_L293D.htm](http://robots-argentina.com.ar/MotorCC_L293D.htm)

## 2.6 Sistema Embebidos

Un Sistema embebido es un sistema electrónico diseñado para realizar funciones específicas en tiempo real, a diferencia de lo que ocurre con las computadoras, las cuales tienen un propósito general, ya que están diseñadas para cubrir un amplio rango de necesidades.

La característica principal es que emplea para realizar las tareas asignadas uno o varios procesadores digitales (CPUs), generalmente formados por un microprocesador, microcontrolador, DSP o FPGA, lo que le permite aportar 'inteligencia' al sistema electrónico del que forma parte.

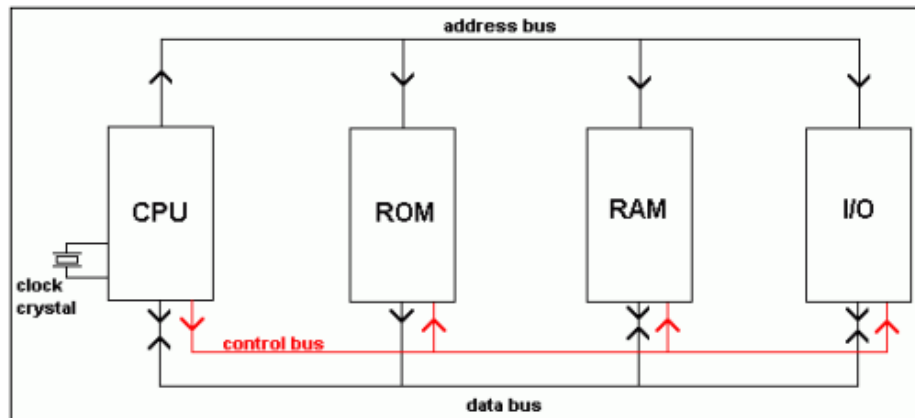


Figura 1: Detalle de MP configuración mínima

**Fig.2.45** Diagrama de bloques de un sistema embebido  
Referencia: Internet

Un sistema embebido interactúa continuamente con el entorno para controlar algún proceso mediante una serie de sensores y actuadores que están conectados a sus módulos de entrada y salida. Para realizar la programación los fabricantes brindan herramientas de software de desarrollo potentes y de acceso libre, se pueden programar directamente utilizando el lenguaje ensamblador o el lenguaje C o C++ y en algunos casos el lenguaje Java. Los programas se graban en las memorias no volátiles (Flash, ROM y EEPROM) y los datos temporales en la memoria SRAM. Los programas almacenados en la memoria no volátil, se conocen como “Firmware” que controlarán al hardware del sistema.

## 2.7 Sistema de control<sup>19</sup>

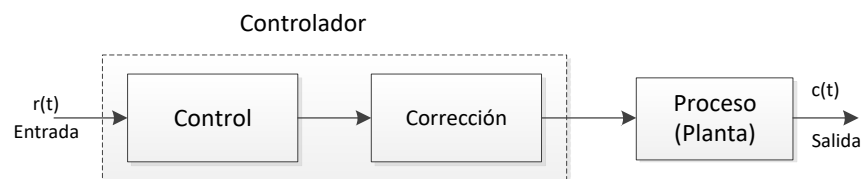
Un sistema de control es una interconexión de componentes que forman una configuración del sistema que proporcionará una respuesta deseada. La base para el análisis de un sistema es el fundamento proporcionado por

<sup>19</sup> Sistemas de control moderno. Richard Dorf

la teoría de los sistemas lineales que supone una relación entre causa y efecto para sus componentes.

### 2.7.1 Sistema de control en lazo abierto y lazo cerrado

Un sistema de control a lazo abierto utiliza un regulador o actuador de control para obtener la respuesta deseada. Un sistema de lazo abierto es un sistema sin realimentación. En un sistema de control en lazo abierto no se mide la salida ni se realimenta para compararla con la entrada. Así, a cada entrada de referencia le corresponde una condición de operación fija; como resultado de ello la precisión del sistema depende de la calibración o corrección.



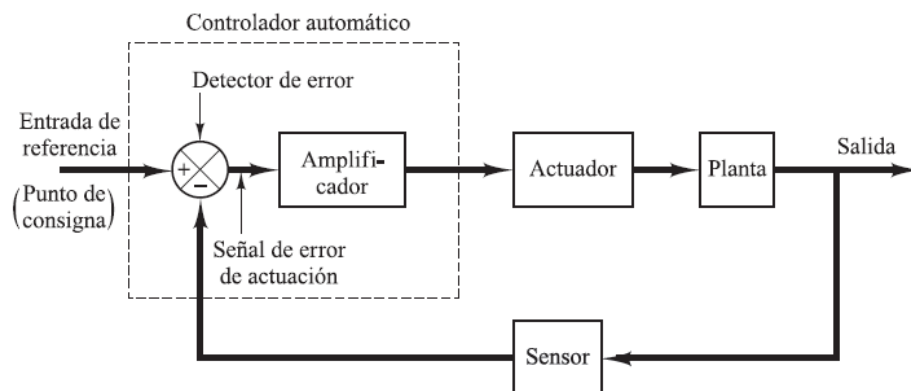
**Fig.2.46** Sistema de control en lazo abierto  
Referencia: autoría propia

En un sistema de control en lazo cerrado usa una medida de la salida y la realimentación de esta señal para compararla con la entrada de referencia (set point). La señal de error es la diferencia entre la señal de entrada y la señal de realimentación, con el fin de reducir el error y llevar la salida del sistema al valor deseado (setpoint)..

La siguiente figura es un diagrama de bloques de un sistema de control industrial<sup>20</sup> que consiste en un controlador automático, un actuador, una

<sup>20</sup> Ingeniería de control moderno. Katsuhiko Ogata

planta y un sensor (elemento de medición). El controlador detecta la señal de error, que por lo general, está en un nivel de potencia muy bajo, y la amplifica a un valor suficientemente alto.



**Fig.2.47** Sistema de control en lazo cerrado  
Referencia: Ingeniería de control moderno. Katsuhiko Ogata

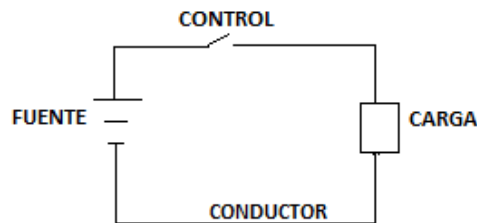
En la Figura 2.47, se observa que la salida del controlador automático alimenta a un actuador (formado por un motor o una válvula neumática, un motor hidráulico o un motor eléctrico) y el sensor convierte la variable de salida en otra variable manejable como temperatura, desplazamiento, una presión o un voltaje, que pueda usarse para comparar la salida con la señal de entrada de referencia. Para elegir un tipo de controlador se debe considerar la naturaleza de la planta, las condiciones de operación, seguridad, costo, disponibilidad, fiabilidad, precisión, peso y tamaño.

### 2.7.2 Clasificación de los controladores por el tipo de energía:

Los controladores de acuerdo al tipo de energía que utiliza la planta para operar, se clasifican en:

- Neumáticos
- Hidráulicos y
- Electrónicos

Los sistemas neumáticos e hidráulicos tienen los mismos elementos que cualquier otro tipo de circuito, como por ejemplo los eléctricos. En estos, la pila suministra la energía, que es conducida por los cables, y controlada por los interruptores hasta llegar al elemento receptor o carga.



**Fig.2.48** Elementos de un circuito eléctrico  
Referencia: autoría propia

En el siguiente cuadro se muestra la equivalencia entre los elementos de un circuito eléctrico y uno neumático o hidráulico

Tabla No.2.2  
Equivalencia entre un circuito eléctrico y neumático

CIRCUITO ELECTRICO	CIRCUITO NEUMATICO O HIDRAULICO
Pila	Compresora o bomba que genera el fluido a presión
Cable	Tubo por donde circula el fluido a presión
Interruptores	Válvulas de control
Carga(receptor)	Cilindros neumáticos o hidráulicos

Referencia: Internet

### 2.7.3 Clasificación de los controladores según la acción de control

Según la acción de control que se aplique a la planta, los controladores se clasifican en:

- Controlador ON-OFF
- Controlador proporcional
- Controlador integral
- Controlador proporcional integral (PI)
- Controlador derivativo
- Controlador proporcional derivativo (PD)
- Controlador proporcional integral derivativo (PID)

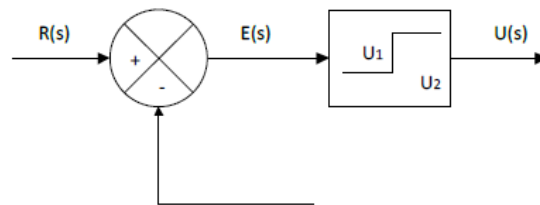
### 2.7.3.1 Controlador ON-OFF

En este controlador el elemento de actuación solo tiene dos posiciones, que son el encendido y apagado.

Ley de control:

$$u(t) = u_1 \quad \text{para } e(t) > 0$$

$$u(t) = u_2 \quad \text{para } e(t) < 0 \quad \text{generalmente } u_2 = 0 \text{ o } u_2 = -u_1$$



**Fig. 2.49** Diagrama de bloques de un controlador ON-OFF  
Referencia: Ingeniería de control moderno. Katsuhiko Ogata

### 2.7.3.2 Controlador proporcional

El control proporcional tiene como entrada la señal de error y la salida del controlador es proporcional a la entrada, es decir:

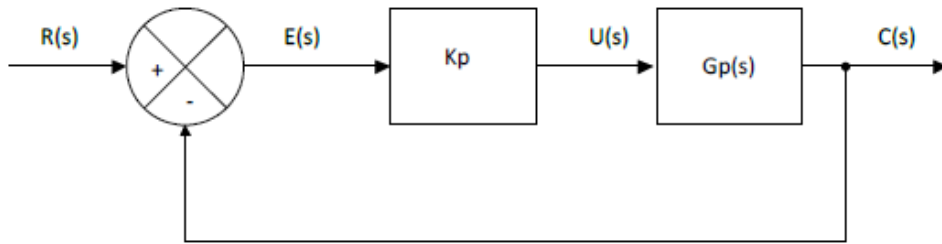
$$u(t) = K_p e(t) \quad (\text{donde } K_p: \text{ganancia proporcional})$$

La función de transferencia del controlador proporcional se define como:

$$G_c(s) = K_p$$

La ganancia constante tiende a existir solo sobre cierto rango de errores que se conoce como banda proporcional.

El controlador proporcional determina la ubicación de los polos en lazo cerrado, el sistema no cambia de tipo, es decir, si un sistema es de tipo 1, el sistema permanece en tipo 1. El controlador proporcional no añade polos o ceros en la trayectoria directa.



**Fig.2.50** Diagrama de bloques de un controlador Proporcional  
Referencia: Ingeniería de control moderno. Katsuhiko Ogata

### 2.7.3.3 Controlador integral

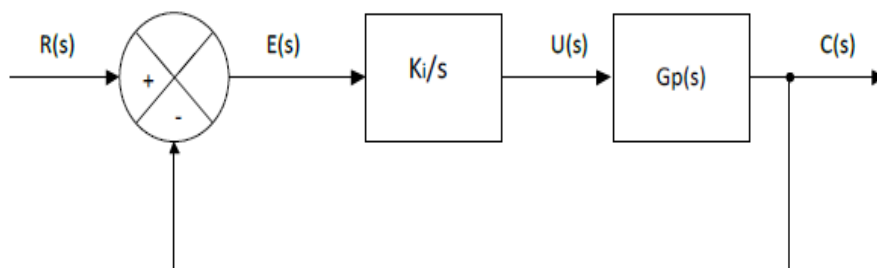
El controlador integral tiene como señal de salida la integral de la señal de error, es decir, la salida en cualquier tiempo es proporcional a la acumulación de los efectos de los errores pasados.

$$u(t) = K_i \int_0^t e dt \quad (K_i: \text{ganancia integral})$$

La función de transferencia del controlador integral es la siguiente:

$$G_c(s) = \frac{K_i}{s}$$

El controlador integral aumenta el tipo de sistema en 1. Además al introducir un polo en tiene como efecto disminuir la estabilidad relativa y disminuir el tiempo de asentamiento.



**Fig.2.51** Diagrama de bloques de un controlador Integral  
Referencia: Ingeniería de control moderno. Katsuhiko Ogata



#### 2.7.3.4 Controlador derivativo

La señal a la salida del controlador derivativo es proporcional a la razón de cambio del error. Cuando el error es grande, el controlador tiene una salida grande si el error tiene una razón de cambio grande. El controlador derivativo no tiene acción de control sobre señales de error constantes o que varíen lentamente.

La ley de control para un controlador derivativo es:

$$u(t) = K_d \frac{de}{dt} \quad K_d: \text{ganancia derivativo.}$$

La función de transferencia del controlador PD es:

$$G_c(s) = K_d s$$

El controlador agrega un cero en la trayectoria directa por lo cual disminuye en 1 el tipo de sistema. Estos controladores se usan en combinación con otras formas de controlador.

#### 2.7.3.5 Controlador proporcional integral (PI)

La señal de salida de un controlador proporcional integral es la suma de los efectos del controlador proporcional y el controlador integral, es decir:

$$u(t) = K_p e + K_i \int_0^t e dt$$

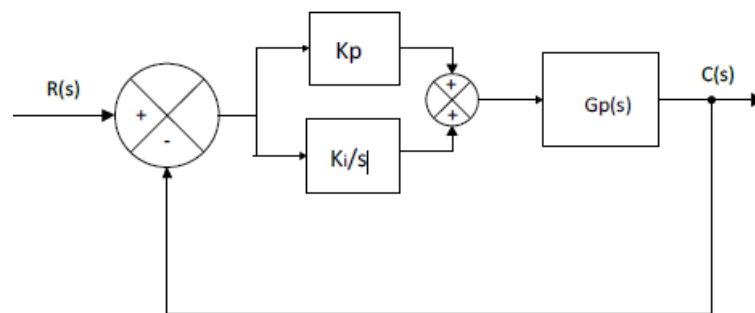
La función de transferencia del controlador proporcional integral es:

$$G_c(s) = K_p \left(1 + \frac{1}{T_i s}\right)$$

Donde:  $T_i = \frac{K_i}{K_p}$  (constante de tiempo integral)

El controlador proporcional integral añade a la trayectoria directa un polo y

un cero. El controlador proporcional integral conlleva a un aumento en el tipo de sistema en 1 y a un corrimiento del lugar geométrico de las raíces hacia la derecha lo cual origina una disminución en la estabilidad relativa, pero es menor que la que se obtiene con el controlador integral. El controlador proporcional integral es un compensador de atraso o retardo



**Fig.2.52** Diagrama de bloques de un controlador PI  
Referencia: Control. Ogata.

### 2.7.3.6 Controlador proporcional integral derivativo (PID)

La señal de salida de un controlador PID es la suma de los efectos de los controladores por separado, es decir:

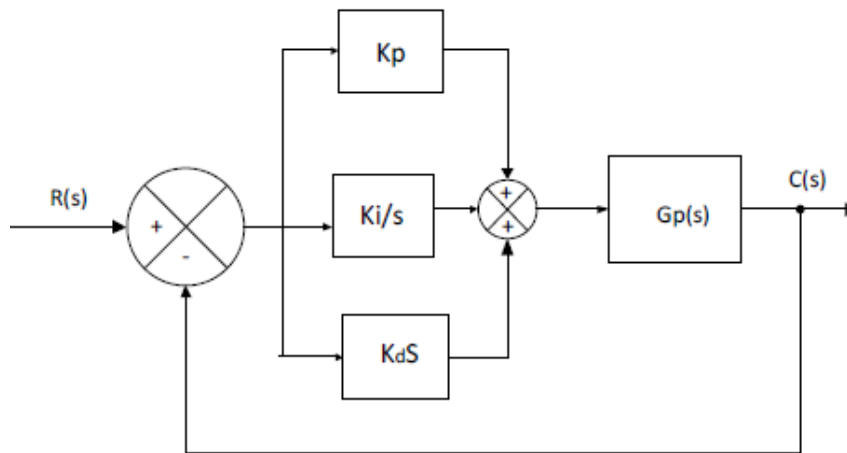
$$u(t) = K_p e + K_i \int_0^t e dt + K_d \frac{de}{dt}$$

La función de transferencia para un controlador PID es la que se muestra a continuación:

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right)$$

El controlador PID es un compensador de atraso-adelanto. La acción PD ocurre en frecuencias altas y la acción PI ocurre en la región de frecuencias bajas. Este controlador se usa cuando se necesitan mejoras en el

comportamiento transitorio y es estado estacionario. Además incrementa el número de ceros en dos y el de polos en uno en la trayectoria directa, en consecuencia se incrementa el tipo de sistema en 1 mejorando la estabilidad del sistema.



**Fig.2.53** Diagrama de bloques de un controlador PID  
Referencia: Control moderno. Ogata.

#### 2.7.4 Método de sintonía de controladores Ziegler - Nichols

El método fue propuesto en 1942 por Jhon G. Ziegler y Nataniel B Nichols para el control de servomecanismos hidráulicos en baterías antiaéreas empleadas en la segunda guerra mundial.

Existen dos métodos de ajuste.

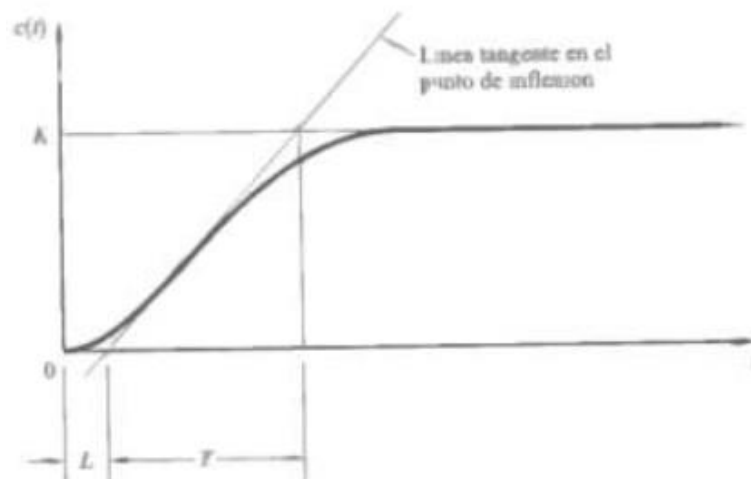
**2.7.4.1 Método de Respuesta al escalón:** consiste en alimentar a la planta a lazo abierto con un escalón unitario inicial, hasta llevar a la planta a un punto de operación normal, luego con un cambio de entrada escalón se debe lograr que se establezca un nuevo punto de operación, como se muestra en la Figura 2.54. Esta curva se llama curva de reacción del proceso.

Para obtener los valores de los parámetros del controlador se usan los valores establecidos en la tabla 2.3

Tabla No.2.3  
Método de la curva de reacción por tipo de controlador

Tipo de controlador	$K_p$	$T_i$	$T_d$
P	$T/L$	$\infty$	0
PI	$0.9T/L$	$L/0.3$	0
PID	$1.2T/L$	$2L$	$0.5L$

Tabla 3.1 Valores de los parámetros por el método de la curva de reacción



**Fig.2.54** Curva de reacción del proceso  
Referencia: Ogata Libro de control

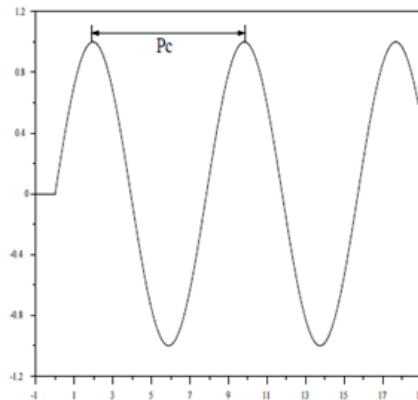
**2.7.4.2 Método de oscilación:** Este método es válido solo para plantas estables a lazo abierto y se lleva a cabo siguiendo los siguientes pasos:



**Fig.2.55** Lazo cerrado con ganancia proporcional  
Referencia: Control moderno. Ogata

Utilizando solo control proporcional, comenzando con un valor de ganancia pequeño, incrementar la ganancia hasta que el lazo comience a oscilar de forma sostenida. Si la salida no presenta oscilaciones sostenidas cual sea el valor de  $K_p$ , entonces no se puede aplicar este método. Registrar la ganancia crítica del controlador  $K_p = K_c$  y el periodo de oscilación de la salida del controlador,  $P_c$ . (en el diagrama de Nyquist, corresponde a que  $K_c G(j\omega)$  cruza el punto  $(-1, 0)$  cuando  $K_p = K_c$ ).

- Ganancia proporcional crítica ( $K_c$ ).- Es la ganancia de un controlador solo proporcional, que provoca que el sistema sea oscilatorio (críticamente estable).
- Periodo de oscilación sostenida ( $P_c$ ).- Es el periodo de oscilación que se consigue con la ganancia crítica.



**Fig.2.56** Respuesta de la planta con ganancia crítica  
Referencia: Control moderno. Ogata

En base a ellas, las expresiones de sintonía para un regulador PID propuestas por Ziegler y Nichols se muestran en la siguiente tabla, y son función de la Ganancia proporcional crítica ( $K_c$ ) y del periodo de oscilación sostenida ( $P_c$ ) del sistema en cuestión:

Tabla No.2.4

Método de sintonía para el regulador PID

Tipo de controlador	$K_p$	$T_i$	$T_d$
P	$0.5K_{cr}$	$\infty$	0
PI	$0.45K_{cr}$	$(5/6)P_{cr}$	0
PID	$0.6K_{cr}$	$0.5 P_{cr}$	$0.125 P_{cr}$

Tabla 3.2 Valores de los parámetros del controlador a partir de  $K_{cr}$  y  $P_{cr}$

Referencia: Control moderno. Ogata

## 2.8 Robot de servicio

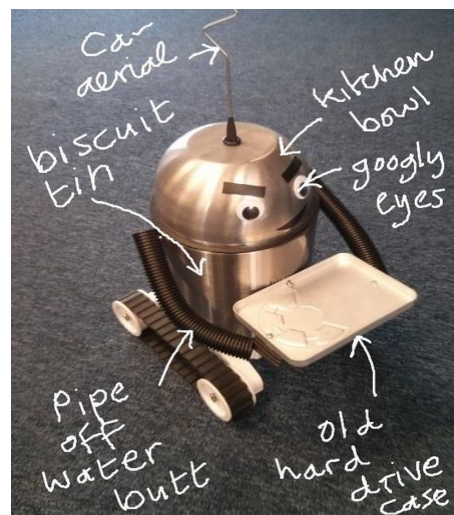
El concepto de robot de servicio ha sido definido de manera tentativa por la Federación Internacional de Robótica (IFR): "*un robot de servicio es un*

*robot que opera semi o totalmente autónomo para realizar servicios útiles para el bienestar de los seres humanos y equipos, con exclusión de las operaciones de fabricación<sup>21</sup>.*

Los robots de servicio tienen aplicaciones mayormente en la industria, en el hogar, en los hospitales, en laboratorios, en las universidades o centros de investigación y en especial para apoyo a personas discapacitadas. Muchas tareas repetitivas que son realizadas por el hombre como cortar el césped en el hogar, analizar una base de datos en los laboratorios, inspeccionar zonas de difícil acceso, trasladar cargas entre áreas diferentes, etc., ahora son realizadas por un robot de servicio como la aspiradora Roomba, los robots mayordomos para asistencia a personas discapacitadas, enfermos o al adulto mayor, entre otros.



a) Aspiradora “Roomba”



b) Robot mayordomo

Fig.2.57 Robot social  
Referencia: Internet

<sup>21</sup> Provisional definition of Service Robots English, 27th of October 2012

Se describe a continuación dos tipos de robots implementados para la realización del presente trabajo de investigación:

- **Robot bycibot:** Implementado con un microcontrolador de 8 bits de la compañía Atmel, con tres ruedas para desplazarse de un determinado lugar a otro, considerando los obstáculos que debe sortear para avanzar. Este robot permitirá apoyar en los trabajos repetitivos de llevar objetos a un determinado lugar, inicialmente fue construido solamente con elementos lógicos sin microcontrolador
- **Robot Evalbot:** implementado con un microcontrolador de 32 bits de Texas Instruments, con dos ruedas que de manera autónoma puede desplazarse sorteando obstáculos a lo largo de su trayectoria hasta llegar al lugar programado. Se busca que el robot apoye en las labores de inspección de las condiciones ambientales de un determinado lugar, como por ejemplo la medición de temperatura y humedad relativa.

## 2.9 Robot móvil Bycibot

El robot móvil denominado “Bycibot” tiene las siguientes características:

- Circuito eléctrico aislado
- Tiene un microcontrolador ATmega8
- Presenta driver L293 para accionar los tres motores que controlan las tres ruedas.
- Programa en lenguaje ensamblador, el código fuente se encuentra en la Parte de Anexos A1.





Fig. 2.58 Robot bycibot  
Referencia: Laboratorio de electrónica.PUCP

Con el nuevo diseño en la parte de hardware, se modifica para obtener mejor funcionabilidad, con los siguientes elementos:

- Microcontrolador TIVA TM4C123GH6PM de Texas Instruments
- Motor servo para controlar el ángulo de giro
- Sensor ultrasónico HC-SR04 para detectar los obstáculos del entorno en donde se desplaza el robot.
- Mando de control remoto con Bluetooth

### **2.9.1 Circuito eléctrico**

La fuente de alimentación está constituida por una batería de 12 V, que se conecta a la entrada del regulador de voltaje C.I.7805 para obtener los 5 voltios que permite alimentar al sensor de ultrasonido, al módulo bluetooth y al motor servo. Los drivers de potencia que se conectan a los motores de corriente continua son alimentados con 12 voltios. El microcontrolador es alimentado por el regulador de 3.3 V.

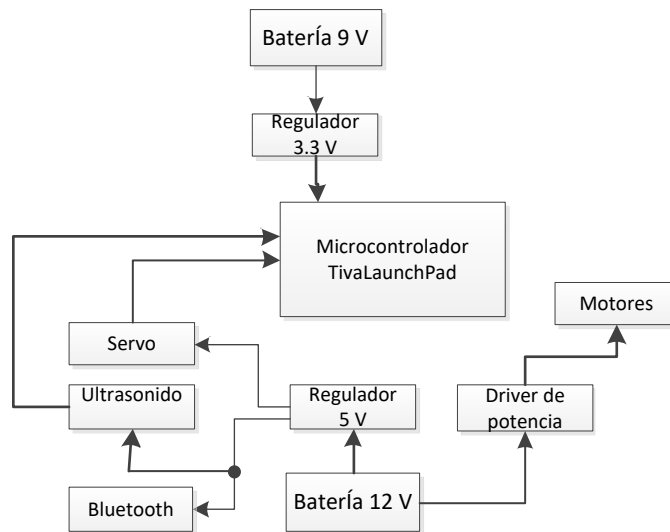


Fig.2.59 Circuito de alimentación del robot  
Referencia: Autoría propia

### 2.9.2 Microcontrolador TM4C123GH6PM<sup>22</sup>

El microcontrolador TM4C123GH6PM, es de Texas Instruments, tiene un bus de datos de 32 bits, con arquitectura ARM-Cortex-M4, tipo Harvard es decir el CPU del microcontrolador está conectada a dos memorias independientes, la memoria de datos (SRAM) y la memoria de programas (FLASH), separados cada uno con sus propios buses de datos y direcciones y lo puede manejar simultáneamente, esto lo hace más eficiente. En cuanto a sus instrucciones es de tipo RISC (Reduced Instrucción Set Computer), y se ejecutan en un solo ciclo de máquina y ocupan una sola dirección en la memoria flash. Entre sus características principales del microcontrolador se indica que tiene una memoria flash de

<sup>22</sup> Pagina web de Texas Instrument. [www.ti.com](http://www.ti.com)

256 Kbytes, con una memoria SRAM de 32 Kbytes, una memoria EEPROM de 2 Kbytes y con frecuencia de operación igual a 80 Mhz.

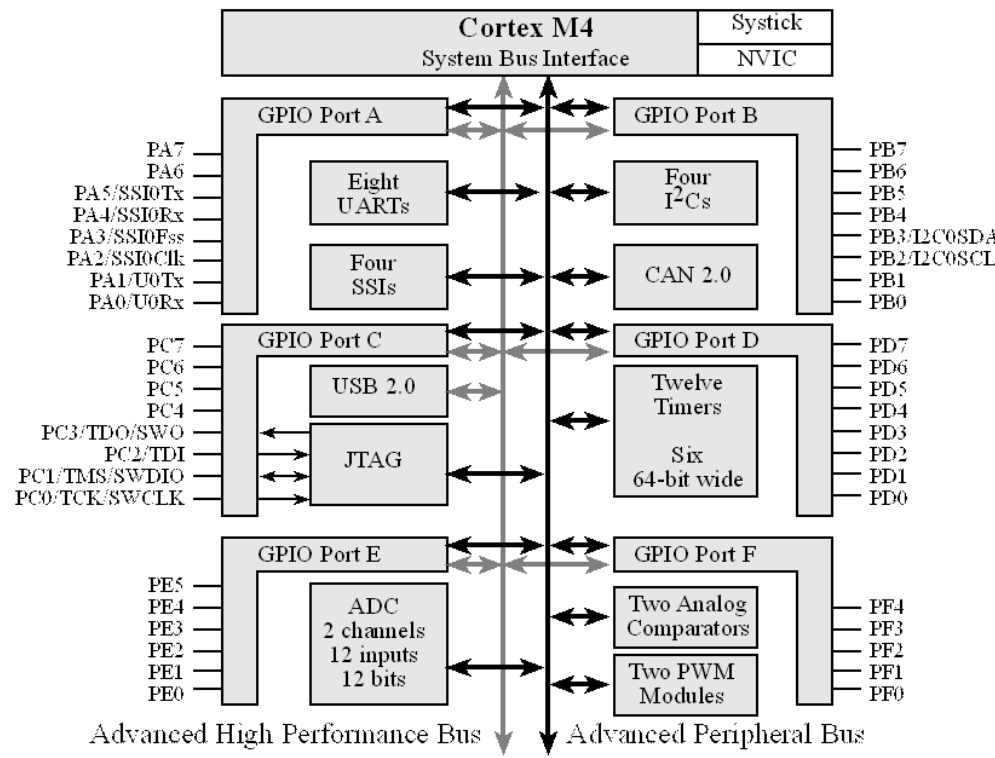


Fig.2.60 Estructura del microcontrolador TM123GH6PM  
Referencia: Data Sheet Texas Instruments [www.ti.com](http://www.ti.com)

En la familia de microcontroladores TM4C, Cada pin del puerto puede ser configurado como pin digital E/S u otra función alternativa como entrada analógica, timer E/S o serial E/S, por ejemplo el pin PA0 puede ser E/S digital o entrada serial (U<sub>0</sub>Rx), para ello se utiliza determinados registros como AMSEL y el registro GPIOPCTL. El microcontrolador TM4C123GH6PM tiene 8 módulos para la comunicación serial, 12 temporizadores donde 6 de ellos son de 64 bits, 12 entradas analógicas donde 2 de ellos son de 12 bits, tiene dos módulos PWM generando hasta 16 salidas PWM. Cada convertidor ADC puede alcanzar hasta 1M

muestras/segundo. La siguiente tabla resume el funcionamiento de cada pin del microcontrolador.

Tabla N°2.5  
Registro GPIOCTL para configuración de pines

IO	Ain	0	1	2	3	4	5	6	7	8	9	14
PA0		Port	U0Rx							CAN1Rx		
PA1		Port	U0Tx							CAN1Tx		
PA2		Port		SSI0Clk								
PA3		Port		SSI0Fss								
PA4		Port		SSI0Rx								
PA5		Port		SSI0Tx								
PA6		Port			I <sub>2</sub> C1SCL		MIPWM2					
PA7		Port			I <sub>2</sub> C1SDA		MIPWM3					
PB0		Port	U1Rx						T2CCP0			
PB1		Port	U1Tx						T2CCP1			
PB2		Port			I <sub>2</sub> C0SCL				T3CCP0			
PB3		Port			I <sub>2</sub> C0SDA				T3CCP1			
PB4	Ain10	Port		SSI2Clk		M0PWM2			T1CCP0	CAN0Rx		
PB5	Ain11	Port		SSI2Fss		M0PWM3			T1CCP1	CAN0Tx		
PB6		Port		SSI2Rx		M0PWM0			T0CCP0			
PB7		Port		SSI2Tx		M0PWM1			T0CCP1			
PC4	C1-	Port	U4Rx	U1Rx		M0PWM6		IDX1	WT0CCP0	U1RTS		
PC5	C1+	Port	U4Tx	U1Tx		M0PWM7		PhA1	WT0CCP1	U1CTS		
PC6	C0+	Port	U3Rx					PhB1	WT1CCP0	USB0epen		
PC7	C0-	Port	U3Tx						WT1CCP1	USB0pflt		
PD0	Ain7	Port	SSI3Clk	SSI1Clk	I <sub>2</sub> C3SCL	M0PWM6	MIPWM0		WT2CCP0			
PD1	Ain6	Port	SSI3Fss	SSI1Fss	I <sub>2</sub> C3SDA	M0PWM7	MIPWM1		WT2CCP1			
PD2	Ain5	Port	SSI3Rx	SSI1Rx		M0Fault0			WT3CCP0	USB0epen		
PD3	Ain4	Port	SSI3Tx	SSI1Tx				IDX0	WT3CCP1	USB0pflt		
PD4	USB0DM	Port	U6Rx						WT4CCP0			
PD5	USB0DP	Port	U6Tx						WT4CCP1			
PD6		Port	U2Rx			M0Fault0		PhA0	WT5CCP0			
PD7		Port	U2Tx					PhB0	WT5CCP1	NMI		
PE0	Ain3	Port	U7Rx									
PE1	Ain2	Port	U7Tx									
PE2	Ain1	Port										
PE3	Ain0	Port										
PE4	Ain9	Port	U5Rx		I <sub>2</sub> C2SCL	M0PWM4	MIPWM2			CAN0Rx		
PE5	Ain8	Port	U5Tx		I <sub>2</sub> C2SDA	M0PWM5	MIPWM3			CAN0Tx		
PF0		Port	U1RTS	SSI1Rx	CAN0Rx		MIPWM4	PhA0	T0CCP0	NMI	C0o	
PF1		Port	U1CTS	SSI1Tx			MIPWM5	PhB0	T0CCP1		C1o	TRD1
PF2		Port		SSI1Clk		M0Fault0	MIPWM6		T1CCP0			TRD0
PF3		Port		SSI1Fss	CAN0Tx		MIPWM7		T1CCP1			TRCLK
PF4		Port					MIPFault0	IDX0	T2CCP0	USB0epen		

Referencia: Data Sheet Texas Instruments [www.ti.com](http://www.ti.com)

La lectura y escritura de puertos GPIO en el microcontrolador presenta las siguientes características:

- Los pines del microcontrolador son en su mayoría bidireccionales.
- EL pin para ser configurado como salida se escribe un “1” lógico en su bit correspondiente en el registro de dirección y el microcontrolador podrá escribir un dato digital por dicho pin.

- El pin para ser configurado como entrada se escribe un “0” en el bit correspondiente del registro de dirección. En este caso el microcontrolador podrá leer en dicho pin la señal digital, como se muestra en la siguiente figura.

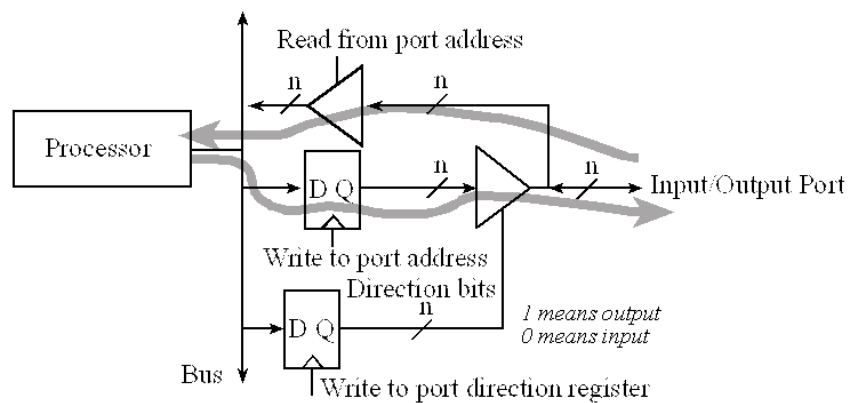


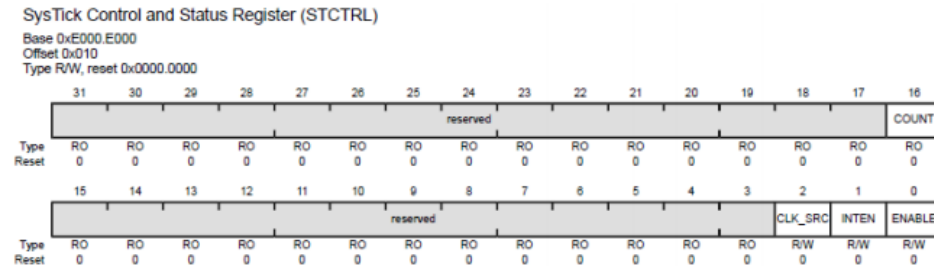
Fig.2.61 Configuración de pines como E/S  
Referencia: Data Sheet Texas Instruments [www.ti.com](http://www.ti.com)

Otro periférico interno del microcontrolador es el temporizador SysTick Timer cuyas características principales son:

- El temporizador es de 24 bits y el SysTick cuenta descendientemente desde el valor programado hasta 0, cuando pasa de “1” a “0” se activa la señal de fin de cuenta y se reinicia desde su valor inicial.
- El temporizador consta de tres registros:
  - SysTick Control and Status (STCTRL): Registro de configuración, para configurar su reloj, activar el contador, permitir la interrupción SysTick, y determinar el estado del contador.
  - SysTick Reload Value (STRELOAD): Se utiliza para recargar el valor del contador.

SysTick Current Value (STCURRENT): Contiene el valor actual del contador

**SysTick Control and Status (STCTRL):**



Donde:

<p><b>COUNT</b>          0: La cuenta no ha llegado a 0          1: La cuenta ha llegado a 0</p>	<p><b>ENABLE</b>          0: SysTick desactivado          1: SysTick activado</p>
<p><b>CLK_SRC (Clock source)</b>          0: Reloj externo (NO IMPLEMENTADO)          1: Reloj del sistema (usar este valor)</p>	<p><b>INTEN (Interrupt enable)</b>          0: No genera interrupciones          1: Interrupciones habilitadas</p>

Fig.2.62 Configuración del registro de control SysTick  
 Referencia: Data Sheet Texas Instruments [www.ti.com](http://www.ti.com)

- El valor para programar el SysTick<sup>23</sup> en su registro STRELOAD se determina por:

$$frecuencia = \frac{f_{CLK}}{RELOAD + 1}$$

$$Ttemp = TCLK (RELOAD + 1)$$

Donde:

$f_{CLK}$  : Frecuencia de reloj del SysTick, en Hz.

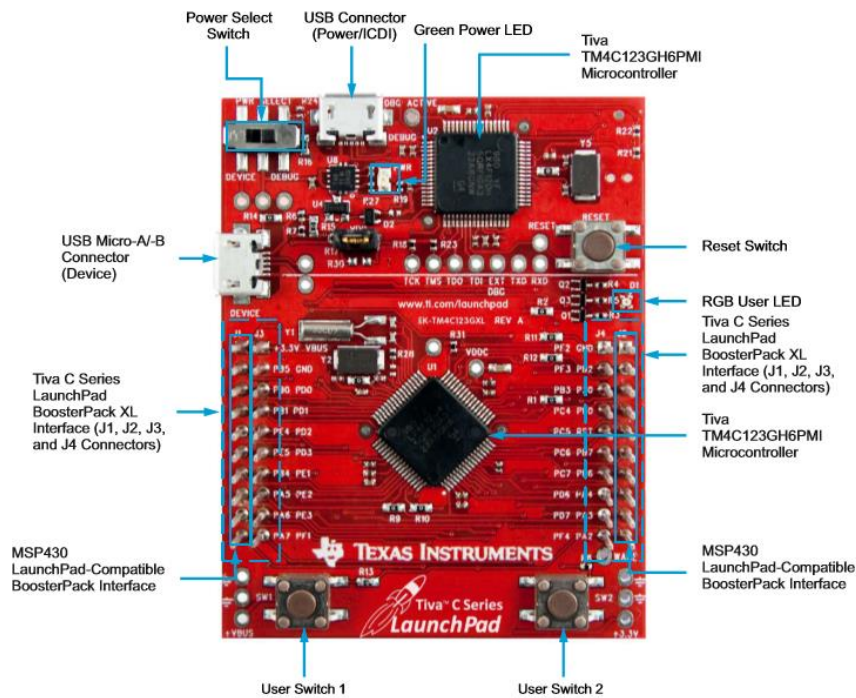
RELOAD: es el valor Inicial a grabar

Para programar el SysTick, primeramente se desactiva este temporizador, luego asignar el valor al registro RELOAD con la formula anterior.

<sup>23</sup> Separatas del microcontrolador TivaLaunchPad. Ing.Hugo Pratt. PUCP.

### 2.9.3 El módulo TIVA LaunchPAD

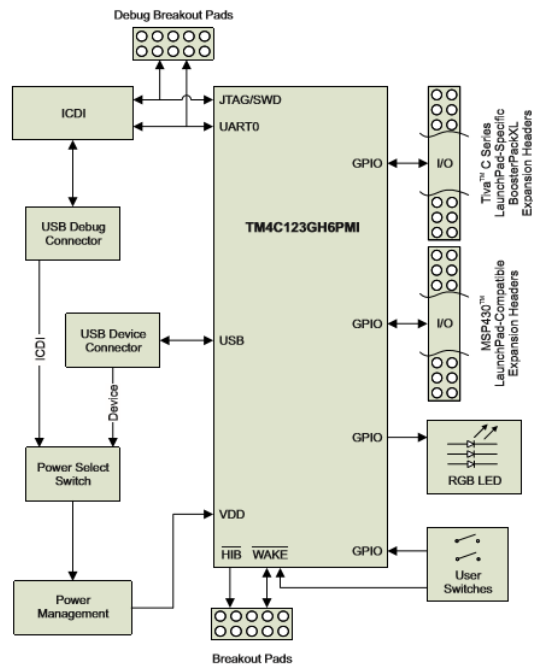
El módulo de Texas Instruments empleado corresponde al modelo Tiva LaunchPAD de la serie C TM4C123G mostrado en la siguiente figura y cuyas características son:



**Fig.2.63** Módulo Tiva C series LaunchPad  
Referencia: Data Sheet Texas Instruments [www.ti.com](http://www.ti.com)

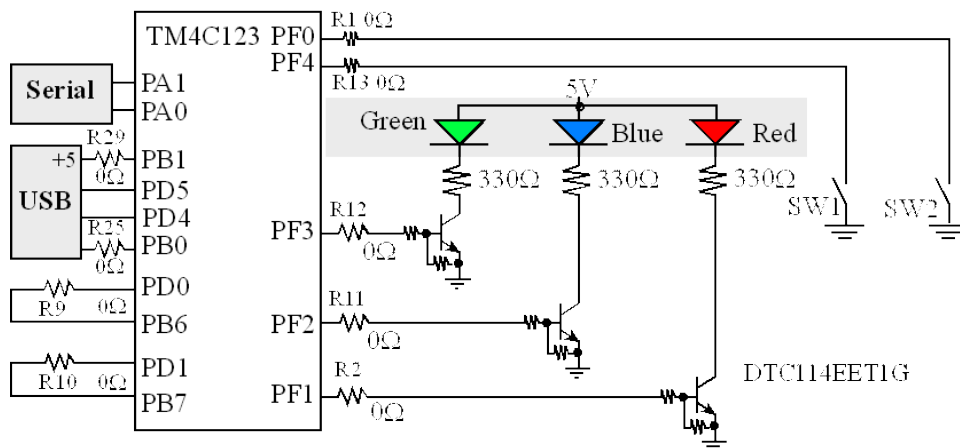
- Microcontrolador Tiva TM4C123GH6PM
- Interfaz de dispositivo USB 2.0
- Dos pulsadores y un LED RGB.
- Un pulsador de reinicio (Reset)
- Conector Micro USB
- Interfaz de depuración (ICDI)
- Fuente de alimentación seleccionable: ICDI y dispositivo USB.

El Diagrama de Bloques de la tarjeta está indicado por la siguiente figura:



**Fig.2.64** Diagrama de bloques del módulo LaunchPad Tiva  
Referencia: Data Sheet Texas Instruments [www.ti.com](http://www.ti.com)

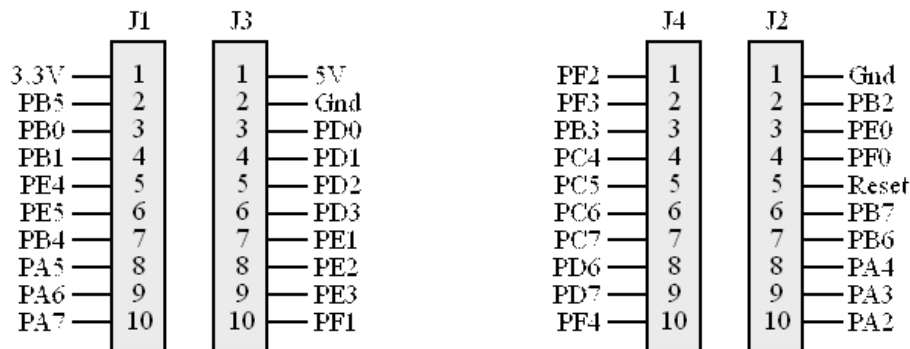
El diagrama esquemático de la tarjeta Tiva indicada se muestra a continuación:



**Fig.2.65** Diagrama esquemático de la Tarjeta TIVA LaunchPAD  
Referencia: Data Sheet TEXAS Instruments. [www.ti.com](http://www.ti.com)

La tarjeta tiene disponible sus puertos de E/S a través de sus conectores tipo “macho” distribuidos en la siguiente forma:





**Fig.2.66** Puertos de E/S disponible en la tarjeta LaunchPad

Referencia: Libro Valvano

[http://users.ece.utexas.edu/~valvano/Volume1/E-Book/C6\\_MicrocontrollerPorts.htm](http://users.ece.utexas.edu/~valvano/Volume1/E-Book/C6_MicrocontrollerPorts.htm)

## 2.10 IDE Code Composer Studio

El IDE (Integrated Development Environment) Code Composer Studio (CCS), es un software de desarrollo para programar microprocesadores y microcontroladores de la familia de Texas Instruments. En esta plataforma debemos seleccionar el tipo de lenguaje que utilizaremos entre el lenguaje ensamblador o lenguaje C (para nuestro proyecto escogemos el lenguaje C). Al abrir la aplicación Code Composer, se muestra una pantalla de inicio como en la siguiente figura.



**Fig.2.67** Code Composer Studio V6

Referencia: Data Sheet Texas Instruments.

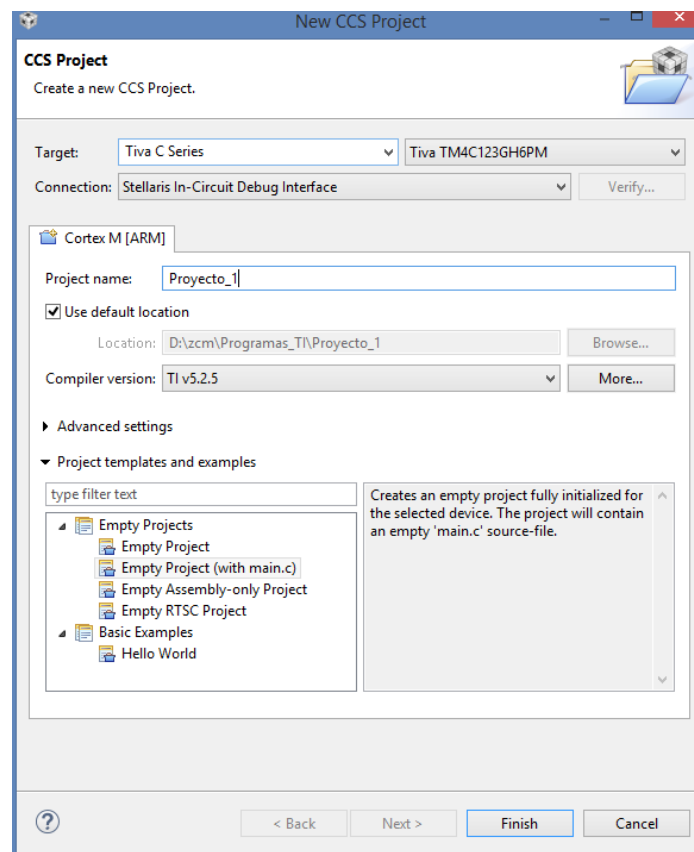
A continuación, para crear un nuevo proyecto se realiza lo siguiente:

En **Target** seleccione la familia **Tiva C Series** y el dispositivo **Tiva TM4C123GH6PM**.

En **Connection** seleccione **Stellaris In-Circuit Debug Interface**.

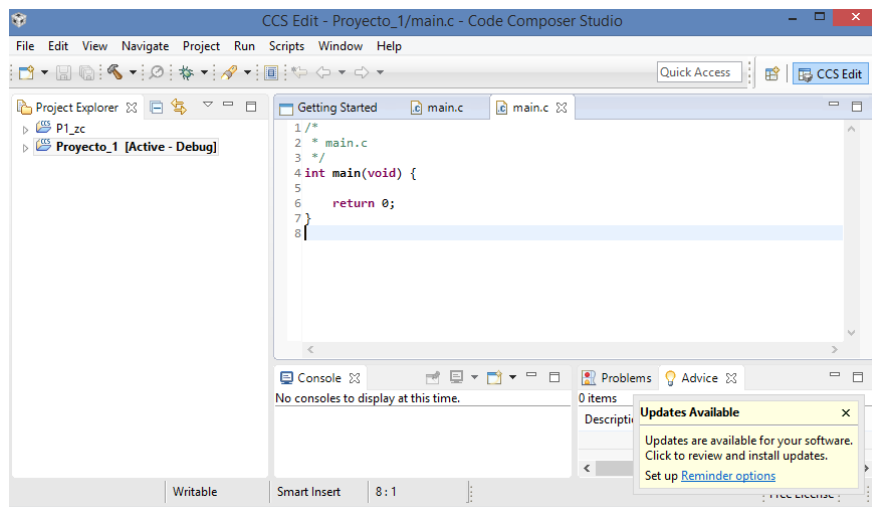
Se ingresa el nombre del proyecto, por ejemplo **proyecto\_1** y luego se asegura en el cuadro inferior izquierdo **Project templates and examples** esté seleccionada la opción **Empty Project(with main.c)**.

Luego de ingresar todos estos datos haga clic en el cuadro **finish**.



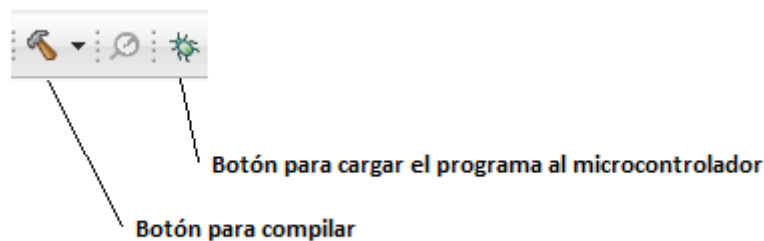
**Fig.2.68** Creación de un Nuevo Proyecto  
Referencia: Data Sheet Texas Instruments.

Y aparecerá una plantilla de inicio para editar un programa en lenguaje C, como se muestra en la siguiente figura:



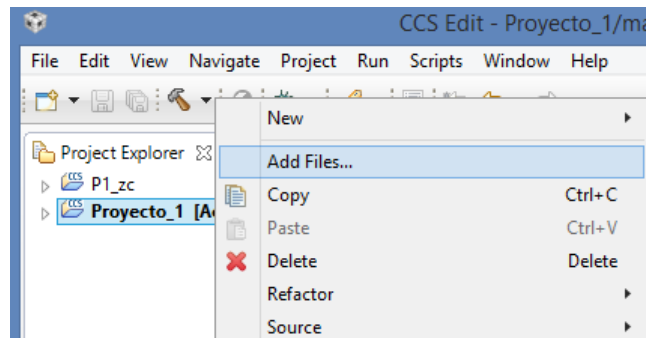
**Fig.2.69** Pantalla de Inicio para programación en C  
Referencia: Data Sheet Texas Instruments.

Se muestra en la siguiente figura los botones que se utiliza para compilar y cargar el programa compilado a la memoria flash del microcontrolador.

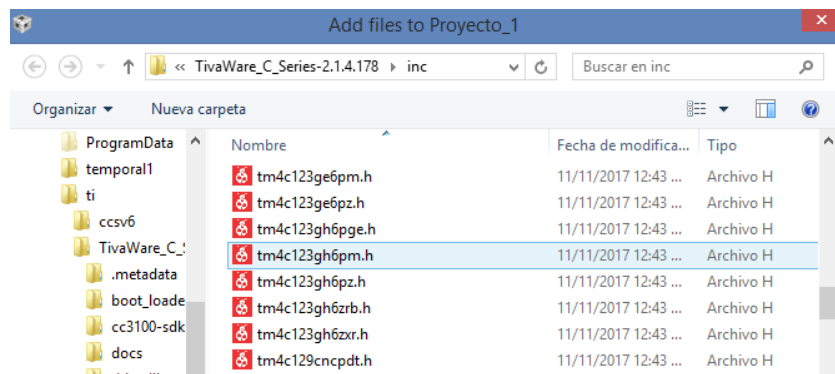


**Fig.2.70** Botones de compilación y programación  
Referencia: Data Sheet Texas Instruments.

Para agregar archivos al Proyecto creado, en el lado izquierdo, se posiciona el mouse sobre el nombre del Proyecto (en el ejemplo Proyecto\_1) y se presiona el botón derecho del mouse, aparecerá un menú desplegable y se selecciona Add Files, escogiendo a continuación el archivo tm4c123gh6pm.h, que se encuentra dentro de la carpeta Inc, como se muestran en las siguientes figuras.

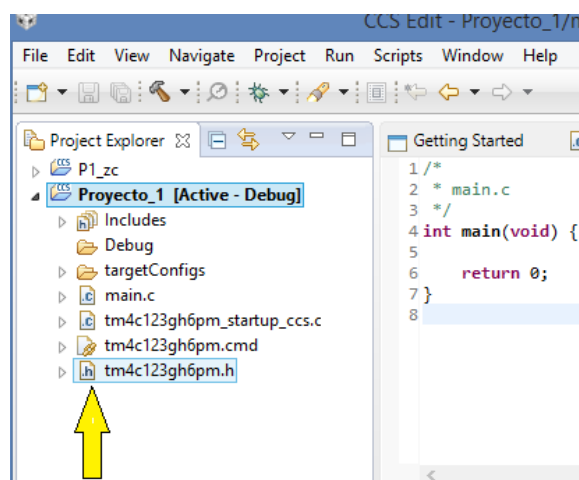


**Fig.2.71** Menú desplegable para crear o añadir archivos al proyecto  
Referencia: Autoría propia



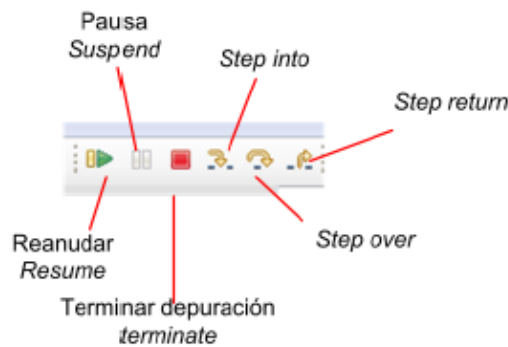
**Fig.2.72** Se selecciona archivo tm4c123gh6pm.h para agregarlo al proyecto  
Referencia: Autoría propia

Finalmente se comprueba que el archivo tm4c123gh6pm.h ha sido copiado dentro de la carpeta Proyecto\_1, observar la siguiente figura.



**Fig.2.73** Copia del archivo tm4c123gh6pm.h  
Referencia: Autoría propia

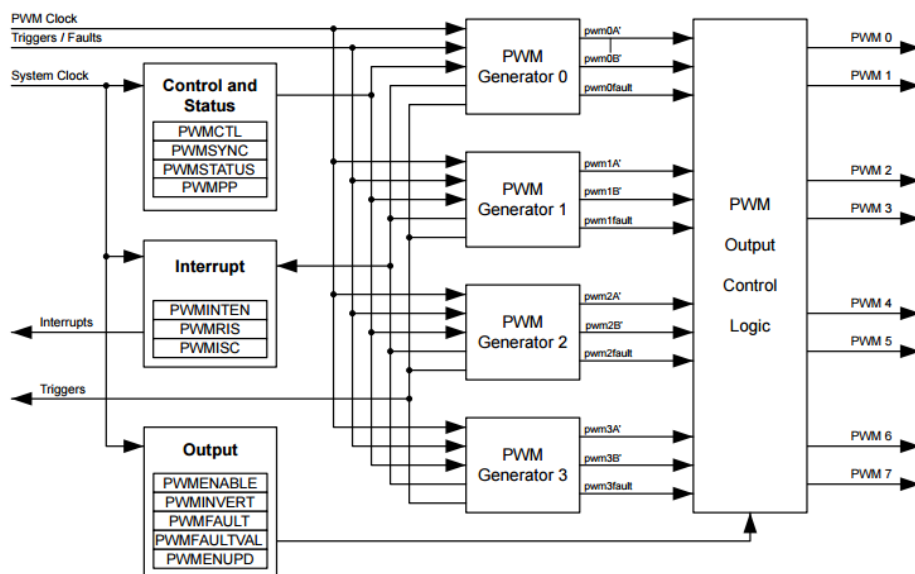
Luego se escribe el programa en el editor, se compila y se transfiere el programa al microcontrolador, obteniéndose los siguientes botones para comenzar al ejecución del programa:



**Fig.2.74** Botones para comenzar con la ejecución del programa  
Referencia: Autoría propia

## 2.11 Obtención de la onda PWM

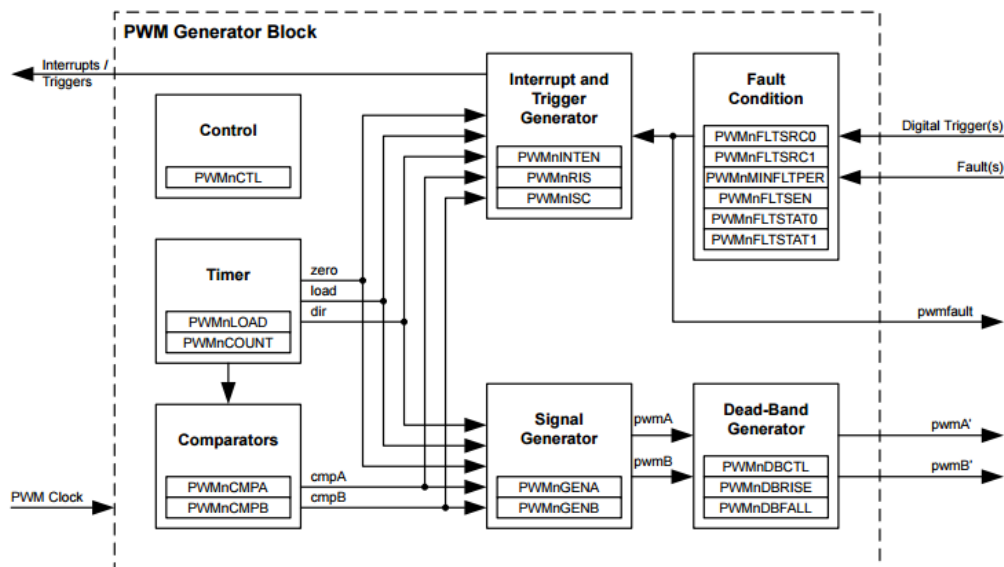
El microcontrolador TM4C123GH6PM tiene dos módulos PWM. Cada módulo tiene 4 sub-módulos generadores de PWM (Generador 0, generador 1, generador 2 y generador 3), como se muestra en la siguiente figura.



**Fig.2.75** Diagrama de bloques del módulo PWM  
Referencia: Data Sheet Texas Instruments.

Cada uno de estos sub-módulos del microcontrolador pueden generar dos ondas PWM independientes a la misma frecuencia, siendo en total 8 ondas PWM para el módulo 0 y otras 8 ondas PWM para el módulo 1, siendo en total 16 ondas PWM.

A continuación, se presenta el diagrama de bloques de un generador PWM, donde se observa los bloques de control, el Timer y dos comparadores que permiten generar la señal PWM.



**Fig.2.76** Diagrama de bloques del generador del módulo PWM  
Referencia: Data Sheet Texas Instruments.

## 2.12 Robot evalbot

El robot móvil de Texas Instruments denominado EVALBOT es un módulo de evaluación basado en el microcontrolador LM3S9B92 de 32 bits de la familia Stellaris, con núcleo ARM, con memoria flash de 256 KB y 96KB de memoria SRAM, tiene salida Ethernet. Además contiene puertos de entrada

y salida, diversos periféricos internos como el módulo serial UART, el temporizador, interrupciones entre otros.

La fuente de alimentación está compuesta por tres pilas AA para suministrar energía a los motores, los puertos de comunicaciones y a los sensores que dispone el módulo. Evalbot



**Fig.2.77** Sistema Evalbot con microcontrolador LM3SW9B92 de 32 bits.  
Fuente: <http://www.ti.com/>

Evalbot, cuyo diagrama de bloques se muestra en la siguiente figura, tiene un software incorporado para efectos de demostración del funcionamiento de cada uno de sus módulos que lo componen. Se conecta a través de su puerto USB a la computadora para efectos de edición, depuración y ejecución de los programas. Siendo sus características principales:

- Microcontrolador Stellaris® LM3S9B92
- Conector de tarjeta MicroSD
- Códec de audio I2S con altavoz
- Conectores de dispositivo y host USB

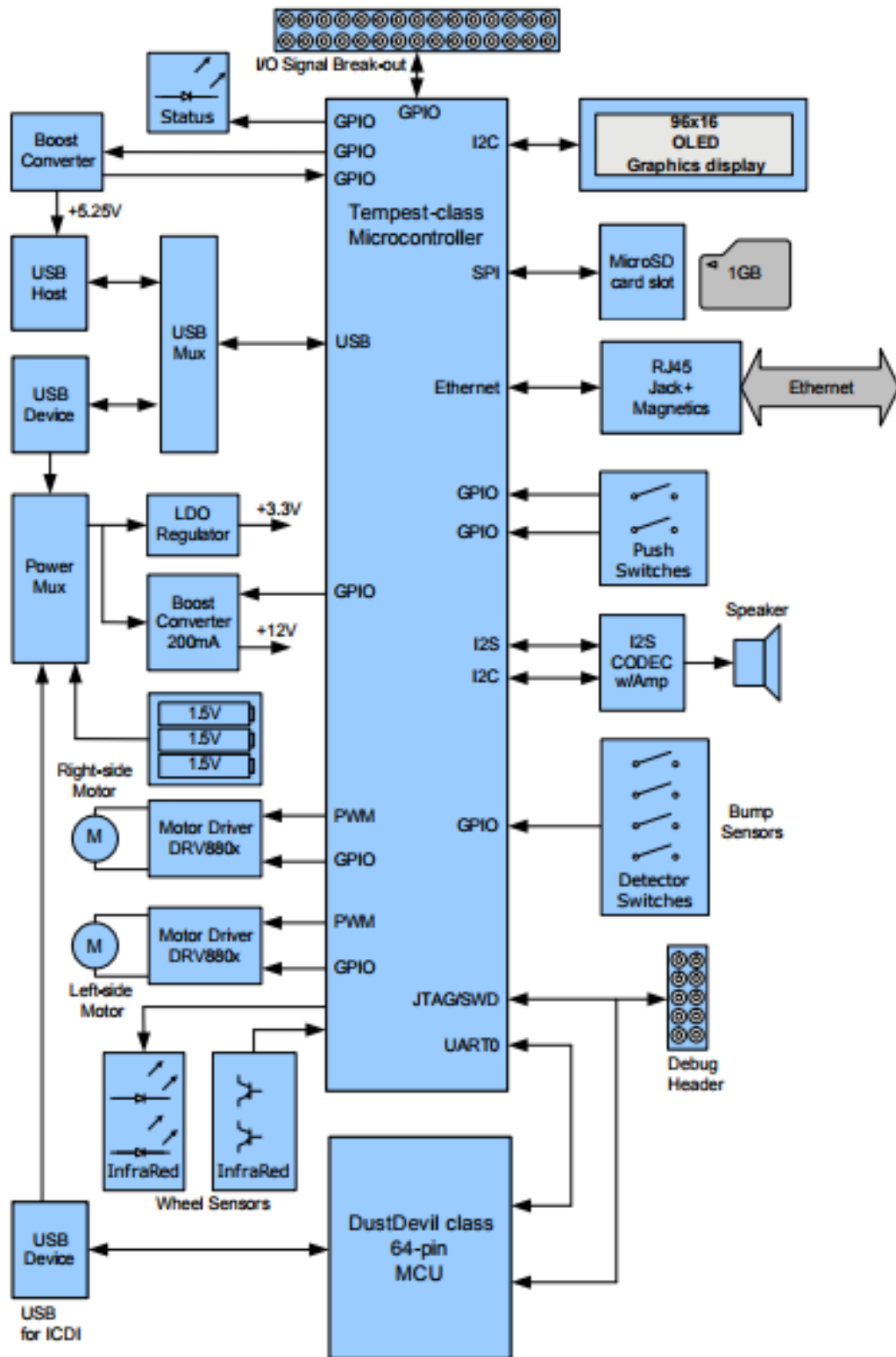


Fig.2.78 Diagrama de bloques del sistema Evalbot

Fuente: <http://www.ti.com/>



- Conector RJ45 Ethernet
- Display OLED con brillo de 96 x 16 en azul
- mediante batería (3 baterías AA) o USB
- Interfaz de depuración en circuito (ICDI)
- Puerto de expansión de comunicación inalámbrica
- Dos motores de engranajes dc proporcionan activación y regulación
- Los sensores ópticos detectan la rotación de la rueda con resolución de 45°.
- Un sensor (parte delantera) para la detección de "choques"

En la siguiente tabla se muestra las especificaciones eléctricas de la fuente de alimentación que soporta el módulo Evalbot

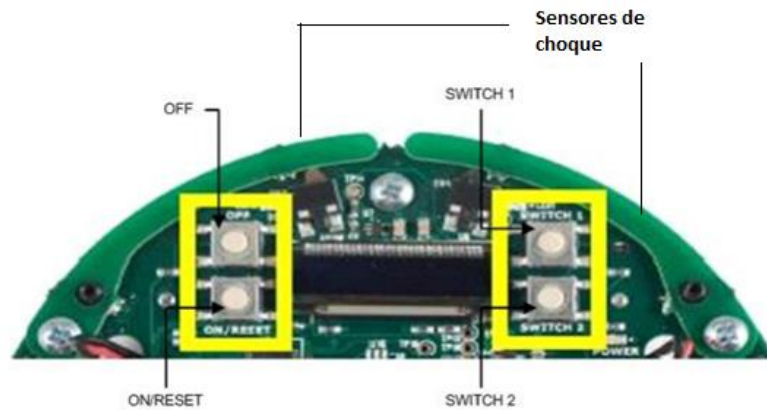
**TABLA N°2.6**  
**Especificaciones eléctricas de la fuente de alimentación**

Parameter	Min	Typical	Max
Battery Supply Voltage	3.5 V	4.5 V	5.0 V
USB Supply Voltage	4.0 V	5.0 V	5.25 V
Battery current (typical stationary)	–	100 mA	–
Battery current (typical in motion)	–	200 mA	–
Power down supply current		0.5 $\mu$ A	
AA Alkaline Battery Capacity (typ)	–	2.5 A/Hr <sup>a</sup> )	–
Reverse Battery Protection		No	
Allowable Battery/USB Current			0.5A

a. From Energizer E91 data sheet.

Referencia: Evalbot data sheets 128420

El módulo contiene 4 pulsadores y los sensores de choque distribuidos en la siguiente forma:



**Fig.2.79** Distribución de los pulsadores en el módulo Evalbot

Fuente: <http://www.ti.com/Stellaris>

Donde:

Botón OFF: El módulo está apagado.

Botón ON/RESET: pulsadores para encender y reiniciar el sistema

SWITCH 1: Para iniciar la ejecución de un programa de aplicación

SWITCH 2: Para detener la ejecución de la aplicación.

Sensores de choque: estos sensores en número de 2 (Lado derecho e izquierdo) permiten detectar el choque con algún objeto.

Para efectos de la investigación se ha desarrollado un software que permite sortear obstáculos hasta llegar a la meta llevando objetos. Se adjunta el programa en la parte de anexos.

## **CAPITULO III**

### **VARIABLES E HIPÓTESIS**

#### **3.1 Definición de las variables**

##### **3.1.1 Variable independiente X.**

- Sistema sensorial
- Sistema de visión.
- Trayectoria a seguir.
- Costo

##### **3.1.2 Variable dependiente Y**

- Detección de obstáculos
- Señalización de la trayectoria
- Ubicación del punto de llegada que es la meta.
- Retorno al punto de partida

#### **3.2 Relación entre las variables de investigación**

##### **3.2.1 Indicadores de la variable independiente:**

- Funcionabilidad del sistema. X1
- Tiempo de trayectoria. X2
- Programación modular del software. X3

##### **3.2.2 Indicadores de la variable dependiente:**

- Facilidad de implementación con sistemas embebidos. Y1
- Uso de Sensores Y2

### **3.3 Hipótesis general y Hipótesis específica**

#### **3.3.1 Hipótesis general**

Desarrollando un sistema sensorial para un robot móvil de servicio social denominado Bycibot permitirá trasladar de manera automática objetos desde un lugar a otro, siguiendo una trayectoria predefinida, evadiendo obstáculos hasta llegar a su destino.

#### **3.3.2 Hipótesis específica**

- Estudiar los sensores de última tecnología para desarrollar el sistema sensorial del robot para mejorar la funcionalidad del sistema.
- Implementar un algoritmo de control para orientar al robot al recibir señales y seguir una trayectoria.
- Conocer los dispositivos electrónicos que emplean los robots para detectar obstáculos, evadirlos y continuar con su trayectoria.
- El robot llevará objetos de un lugar a otro para mejorar la productividad de la empresa.

## **CAPITULO IV**

### **METODOLOGIA**

#### **4.1 Tipo de investigación**

El desarrollo del presente trabajo es de tipo metodológica y aplicada, se hizo un estudio descriptivo del estado actual de la robótica desde su creación, de su desarrollo a lo largo del tiempo y sus aplicaciones de los robots móviles en diferentes campos de nuestra vida cotidiana. En particular se desarrolló el sistema sensorial de un robot móvil para una aplicación social que ayude al hombre a realizar sus tareas.

#### **4.2 Diseño de la Investigación**

La metodología empleada es la siguiente:

- Se planteó el problema a resolver utilizando un robot móvil
- Primero se inició con la investigación de los diferentes tipos de robots desde sus inicios y sus aplicaciones en la actualidad.
- Se investigó el comportamiento de un robot móvil, sus características de diseño en hardware y software del sistema.
- Se planteó la hipótesis de investigación
- Se buscó algunas alternativas de solución, escogiendo lo más adecuado.
- Se realizó pruebas con tarjetas con microcontroladores y dispositivos de potencia, analizando los resultados obtenidos y planteando recomendaciones para un mejor trabajo.

### 4.2.1 Diagrama de Flujo del sistema

Se presenta a continuación el diagrama de flujo del sistema en la siguiente figura Fig.4.1

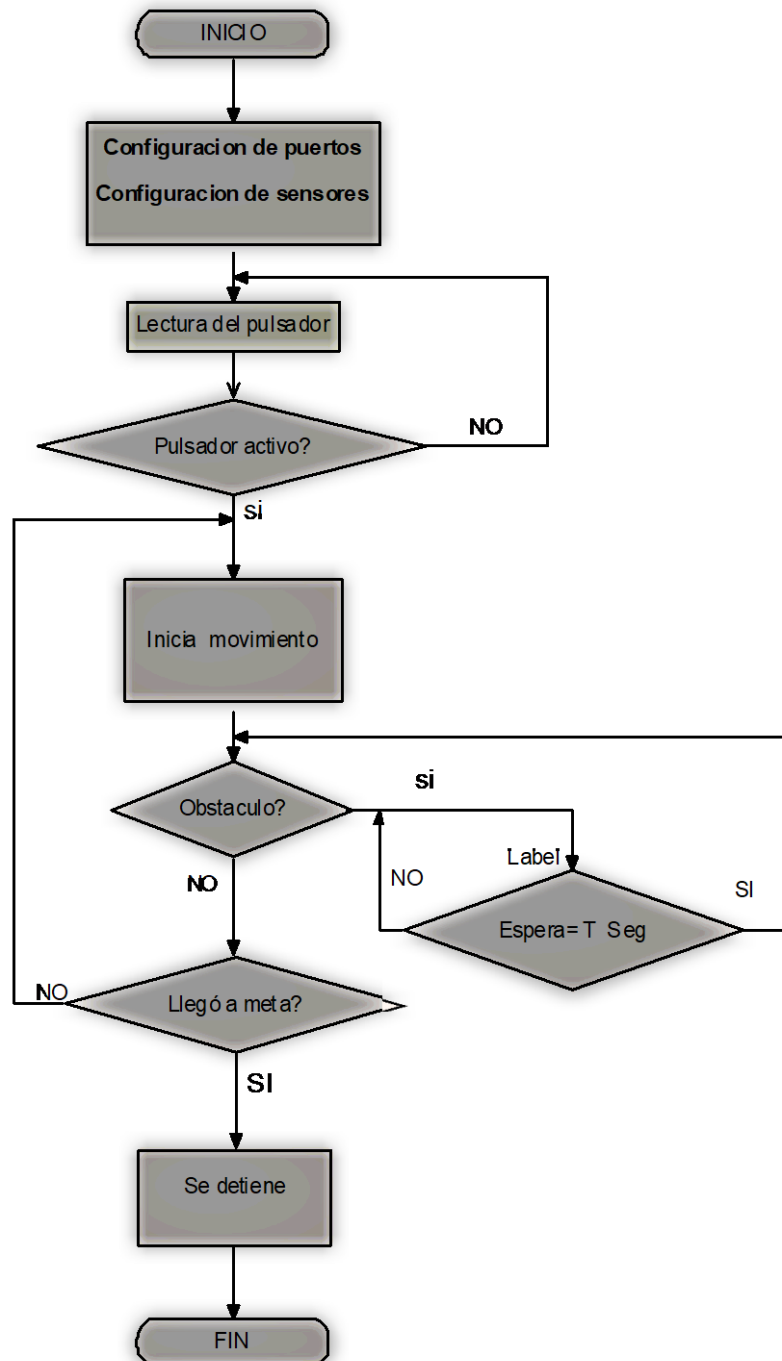


Fig.4.1 Diagrama de flujo  
Referencia: autoría propia

#### 4.2.2 Diseño del sistema de señalización

El robot móvil en su trayectoria debe ser capaz de recibir y entregar la información sin perturbaciones, en forma adecuada, con los niveles eléctricos permitidos, con detección de obstáculos y señalizaciones durante su movimiento hasta llegar a la meta. En la siguiente figura se presenta las señalizaciones que tiene el robot móvil propuesto.

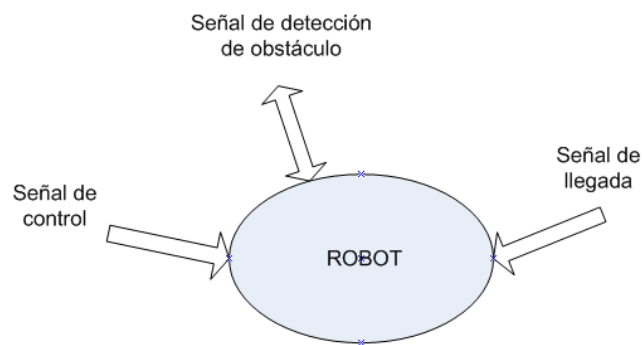


Fig.4.2 Señalización en el robot  
Referencia: autoría propia

Las señales consideradas son:

- a) Señales de control de encendido, apagado, parada y restablecimiento, para que el robot móvil basado en un microcontrolador pueda interpretar dichas señales se realizan mediante un mando inalámbrico compuesto por el módulo bluetooth utilizando su puerto serial.
- b) Señal de detección de obstáculo. La ubicación apropiada de los sensores de ultrasonido en la parte delantera del robot, al avanzar permite detectar obstáculos que se pueden presentar por el tránsito de las personas u otra razón deteniéndose hasta quedar libre la trayectoria y reanudando su movimiento llegar a su destino.

- c) Señales de control y chequeo del robot para su reinicio es importante en caso se detecten fallas en su movimiento

### 4.2.3 Trayectoria del robot

La trayectoria del ambiente de trabajo del robot está indicada en la Fig. 4.3, donde se aprecia lo siguiente:

- Partida: Lugar donde se encuentra inicialmente el robot, se ha definido como el ambiente A.
- Trayectoria: Se Inicia en el punto A, luego pasa por el punto B y C, para llegar al punto M.
- Destino: Es el lugar final donde se dirige el robot, se define como el ambiente M.

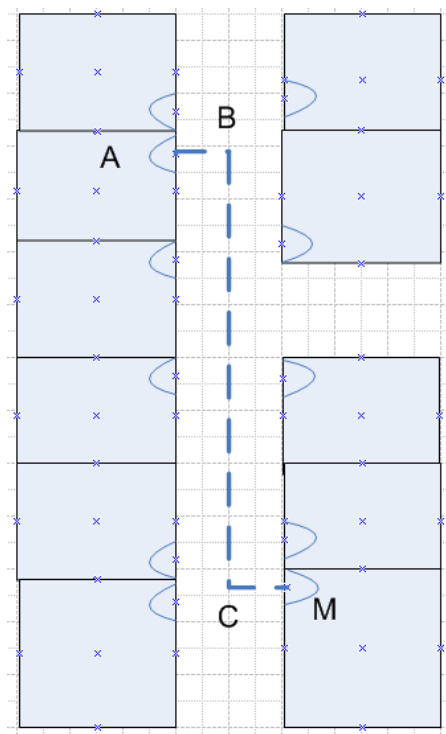


Figura 4.3 Trayectoria del robot  
Referencia: autoría propia



Por consiguiente, el robot móvil que parte de un lugar determinado, debe saber ¿Dónde estoy? ¿Dónde voy? ¿Cómo llego a mi destino? En el primer caso el robot parte de un lugar fijo denominado A (está dentro del ambiente de trabajo del usuario), luego el robot se dirige al punto B (se encuentra en el pasillo) para luego girar 90 grados a la derecha y seguir una trayectoria rectilínea conocida, sorteando obstáculos hasta llegar a su destino denominado C y controlado desde el punto M.

En la siguiente figura N°4.4, se muestra las posiciones que adopta el robot móvil desde su posición inicial, el sentido de su desplazamiento (Norte, Este, Sur) hasta llegar a su posición final, donde gira 180° y se detiene esperando una orden para volver a su lugar de inicio.

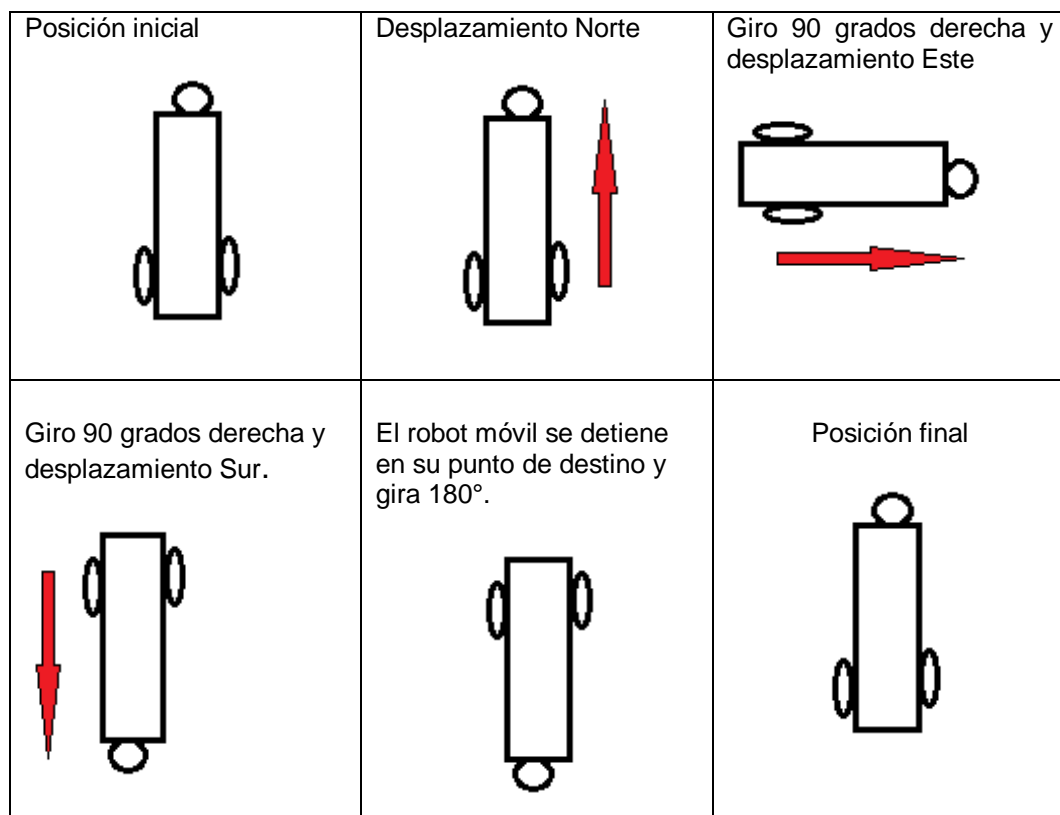


Figura 4.4 Posición y desplazamiento del robot  
Referencia: autoría propia

#### 4.2.4 Diagrama de bloques del robot móvil

El diagrama de bloques del robot móvil se muestra a continuación:

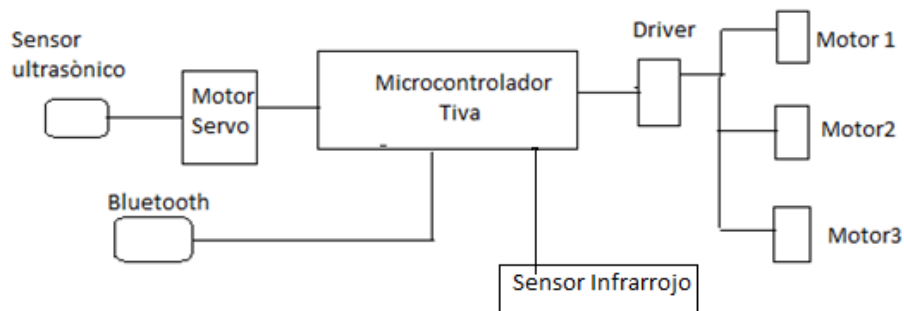


Fig.4.5 Diagrama de bloques del robot móvil con LaunchPadTiva  
Referencia: autoría propia

El sensor ultrasónico determina si hay obstáculos para que el robot siga avanzando. Si encuentra un obstáculo, el robot se detiene y el motor servo comienza a escanear verificando si la vía está libre otra vez para continuar con su recorrido hasta llegar a la meta determinado por los sensores infrarrojos.

#### 4.2.5 Diagrama esquemático del robot

El diagrama esquemático del circuito está indicado en la figura 4.6, donde se observa el pinado<sup>24</sup> completo del microcontrolador Tiva LaunchPad y las que se utiliza para la conexión a los módulos de ultrasonido, bluetooth y el motor servo.

El driver que se utiliza para conectar a los motores será basado en el circuito con puente H el driver L298 de la Fig.4.7. Esta posee 2 entradas de alimentación una de 5V para controlar la parte lógica y otra para energizar las salidas del motor a 12 V, empleándose el módulo de potencia tal como se indica en la Figura 4.8 donde se muestra las entradas para los motores y sus habilitadores.

<sup>24</sup> <http://embedded-lab.com/blog/tiva-c-gpios/>

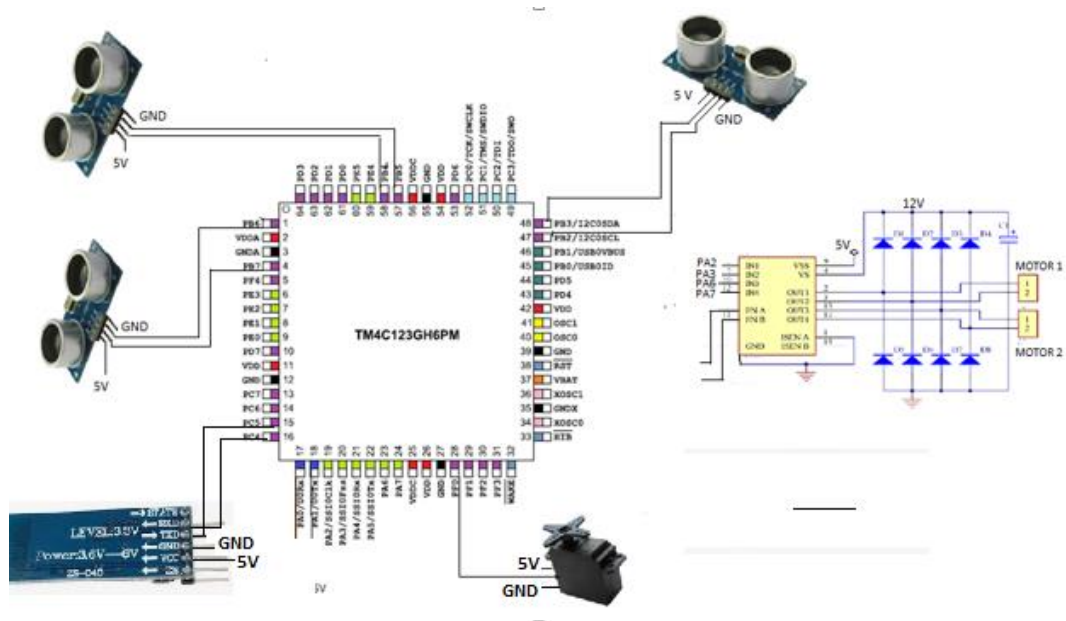


Fig.4.6 Diagrama esquemático simplificado del circuito  
Referencia: autoría propia



Fig.4.7 Driver L298  
Referencia: Hojas técnicas

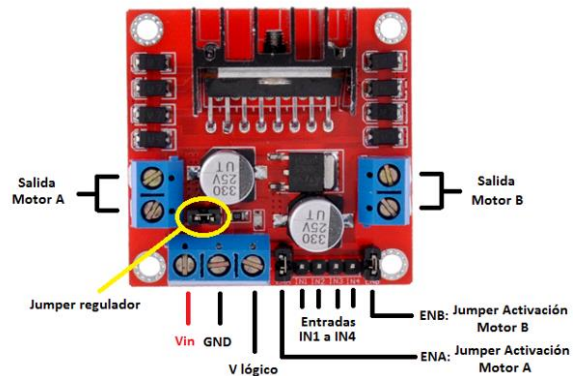


Fig.4.8 Módulo de potencia para los motores con L298  
Referencia: Hojas técnicas

Características de la tarjeta del driver L298:

- Corriente máxima 2 Amperios.
- Posee 6 entradas de control (ver tabla de control)
- Admite entradas de señal PWM para el control de velocidad.
- Dimensiones: 43 mm x 23,9 mm x 43 mm.
- Salidas: para 2 motores de DC

#### **4.2.6 Lista de componentes:**

- 1 tarjeta LaunchPad Tiva
- 1 driver 293
- 2 motores DC
- 1 motor servo
- 1 sensor infrarrojo
- 1 módulo bluetooth
- 1 sensor ultrasónico

#### **4.2.7 Generación de ondas PWM**

Para generar la onda PWM, se hace uso de:

- Un contador de 16 bits: Funciona en modo ascendente o descendente y la frecuencia de salida depende del valor de carga de 16 bits.
- Dos comparadores PWM: para actualizar los valores cuando se produce la comparación exitosa.
- Un generador de señal PWM: se produce la onda PWM como resultado del contador y el comparador PWM.

Para la programación se utiliza la siguiente tabla para obtener las ondas PWM utilizando el microcontrolador Tiva LaunchPad, dado este tiene dos generadores M0PWM y M1PWM donde cada uno puede generar 8 ondas hacen un total de 16 ondas PWM, cuya salida es utilizando los siguientes pines mostradas en la siguiente tabla N° 4.1

**Tabla N° 4.1**

Distribución de pines para la obtención de 16 ondas PWM

Pin Name	Pin Number	Pin Mux / Pin Assignment	Pin Type	Buffer Type <sup>a</sup>	Description
M0FAULT0	30 53 63	PF2 (4) PD6 (4) PD2 (4)	I	TTL	Motion Control Module 0 PWM Fault 0.
M0PWM0	1	PB6 (4)	O	TTL	Motion Control Module 0 PWM 0. This signal is controlled by Module 0 PWM Generator 0.
M0PWM1	4	PB7 (4)	O	TTL	Motion Control Module 0 PWM 1. This signal is controlled by Module 0 PWM Generator 0.
M0PWM2	58	PB4 (4)	O	TTL	Motion Control Module 0 PWM 2. This signal is controlled by Module 0 PWM Generator 1.
M0PWM3	57	PB5 (4)	O	TTL	Motion Control Module 0 PWM 3. This signal is controlled by Module 0 PWM Generator 1.
M0PWM4	59	PE4 (4)	O	TTL	Motion Control Module 0 PWM 4. This signal is controlled by Module 0 PWM Generator 2.
M0PWM5	60	PE5 (4)	O	TTL	Motion Control Module 0 PWM 5. This signal is controlled by Module 0 PWM Generator 2.
M0PWM6	16 61	PC4 (4) PD0 (4)	O	TTL	Motion Control Module 0 PWM 6. This signal is controlled by Module 0 PWM Generator 3.
M0PWM7	15 62	PC5 (4) PD1 (4)	O	TTL	Motion Control Module 0 PWM 7. This signal is controlled by Module 0 PWM Generator 3.
M1FAULT0	5	PF4 (5)	I	TTL	Motion Control Module 1 PWM Fault 0.
M1PWM0	61	PD0 (5)	O	TTL	Motion Control Module 1 PWM 0. This signal is controlled by Module 1 PWM Generator 0.
M1PWM1	62	PD1 (5)	O	TTL	Motion Control Module 1 PWM 1. This signal is controlled by Module 1 PWM Generator 0.
M1PWM2	23 59	PA6 (5) PE4 (5)	O	TTL	Motion Control Module 1 PWM 2. This signal is controlled by Module 1 PWM Generator 1.
M1PWM3	24 60	PA7 (5) PE5 (5)	O	TTL	Motion Control Module 1 PWM 3. This signal is controlled by Module 1 PWM Generator 1.
M1PWM4	28	PF0 (5)	O	TTL	Motion Control Module 1 PWM 4. This signal is controlled by Module 1 PWM Generator 2.
M1PWM5	29	PF1 (5)	O	TTL	Motion Control Module 1 PWM 5. This signal is controlled by Module 1 PWM Generator 2.
M1PWM6	30	PF2 (5)	O	TTL	Motion Control Module 1 PWM 6. This signal is controlled by Module 1 PWM Generator 3.
M1PWM7	31	PF3 (5)	O	TTL	Motion Control Module 1 PWM 7. This signal is controlled by Module 1 PWM Generator 3.

a. The TTL designation indicates the pin has TTL-compatible voltage levels.

Fuente: data sheet del módulo Launch Pad Tiva

#### 4.2.7.1 Descripción de los principales registros

Para la programación del microcontrolador Tiva LaunchPad entorno al lenguaje C, se detalla las características de los principales registros.

- **Registro RCGC2 (Run Mode Clock Gating Control Register 2):**

El registro RCGC2, es utilizado para activar el reloj de un puerto del microcontrolador. Los puertos disponibles son Puerto A, Puerto B, Puerto C, Puerto D, Puerto E y Puerto F y cada uno tiene su propio reloj.

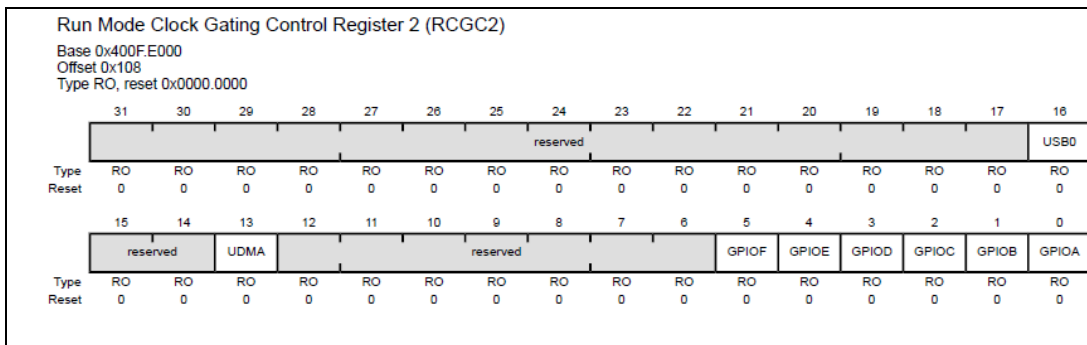


Figura N° 4.9 Registro RCGC2

Fuente: Data sheet del microcontrolador Tiva Launch Pad

Por ejemplo, para habilitar el reloj del puerto A:

`SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOA, o`

`SYSCTL_RCGC2_R |= 0x0001`

Por ejemplo, para habilitar el reloj del puerto B:

`SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOB, o`

`SYSCTL_RCGC2_R |= 0x0002`

- **Registro GPIO DIR**

El registro GPIODIR es el registro de dirección de datos. Colocar un "1" en un bit del registro GPIODIR configura el pin correspondiente como una salida, mientras que escribir un "0" lógico configura el pin correspondiente como una entrada. Todos los bits se borran mediante un reset, lo que significa que todos los pines GPIO son entradas por defecto.

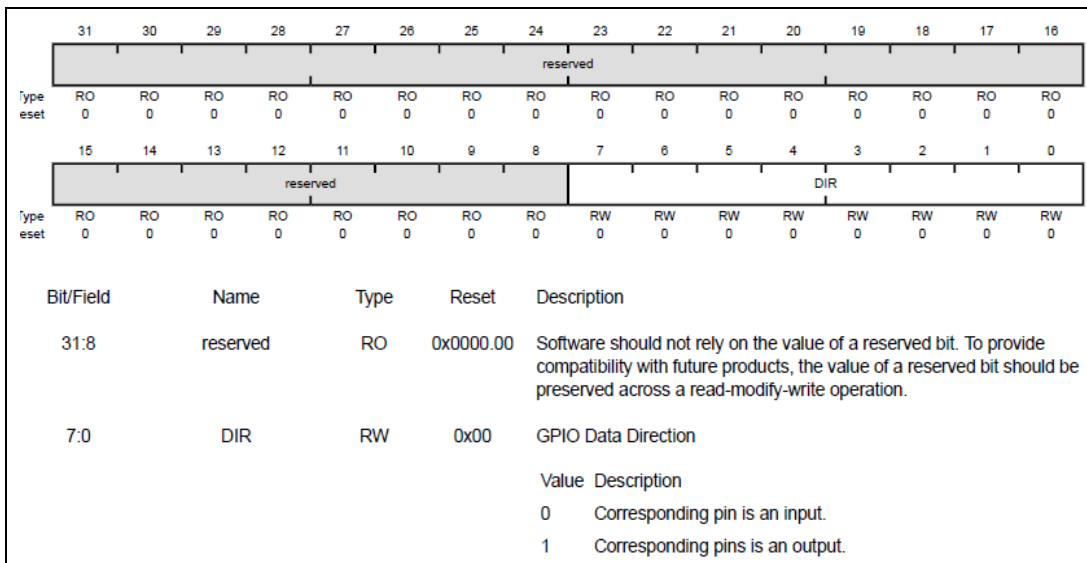


Figura N° 4.10. Registro GPIODIR

Fuente: Data sheet del microcontrolador Tiva Launch Pad

Ejemplo:

Para configurar el pin 6 y 7 del puerto B como salida:

`GPIO_PORTB_DIR_R |= 0xC0;`

• **Registro para habilitar la salida digital (GPIO Digital Enable)**

Registro GPIODEN: Se habilitan con “1” lógico y con “0” lógico están deshabilitados.

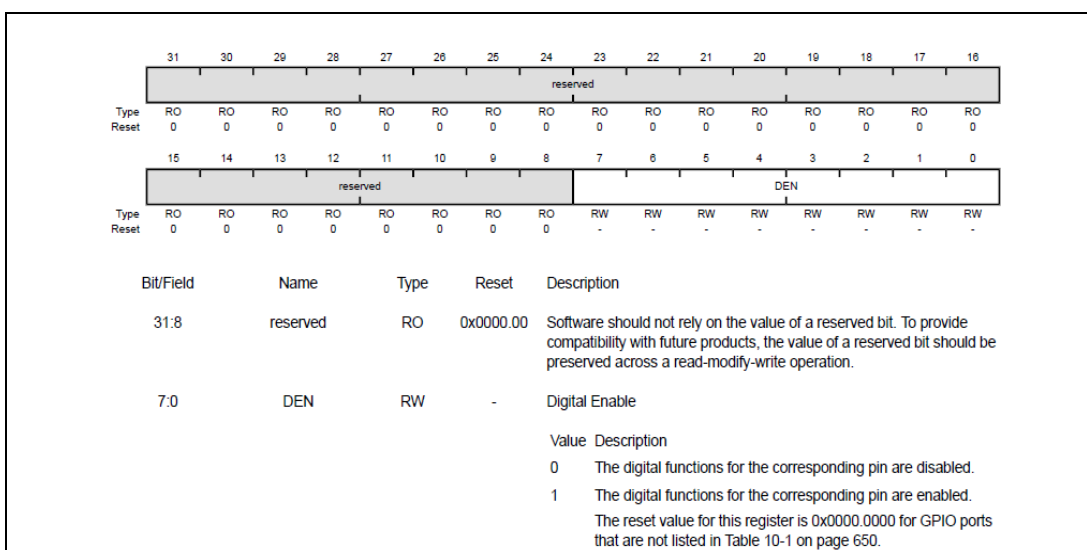


Figura N° 4.11. Registro GPIODEN

Fuente: Data sheet del microcontrolador Tiva Launch Pad

- **Registro para habilitar la salida del driver de 8 mA (GPIO Drive Selec).** Se denomina registro GPIO DR 8R

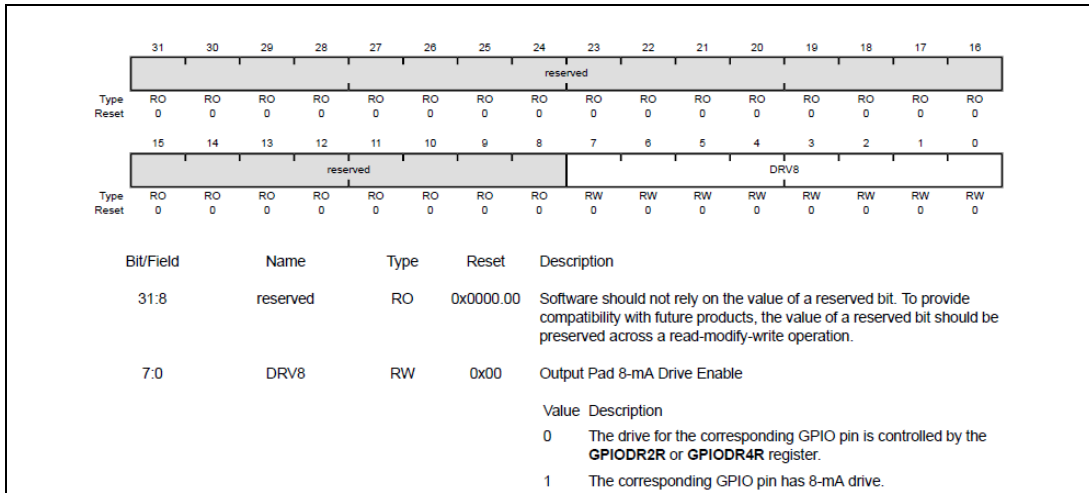


Figura N° 4.12 Registro GPIO DR 8R

Fuente: Data sheet del microcontrolador Tiva Launch Pad

- **Registro para habilitar la salida del puerto o la función alternativa (GPIO AFSEL)**

El registro GPIOAFSEL es el registro de selección para estar en modo GPIO o como una función alternativa.

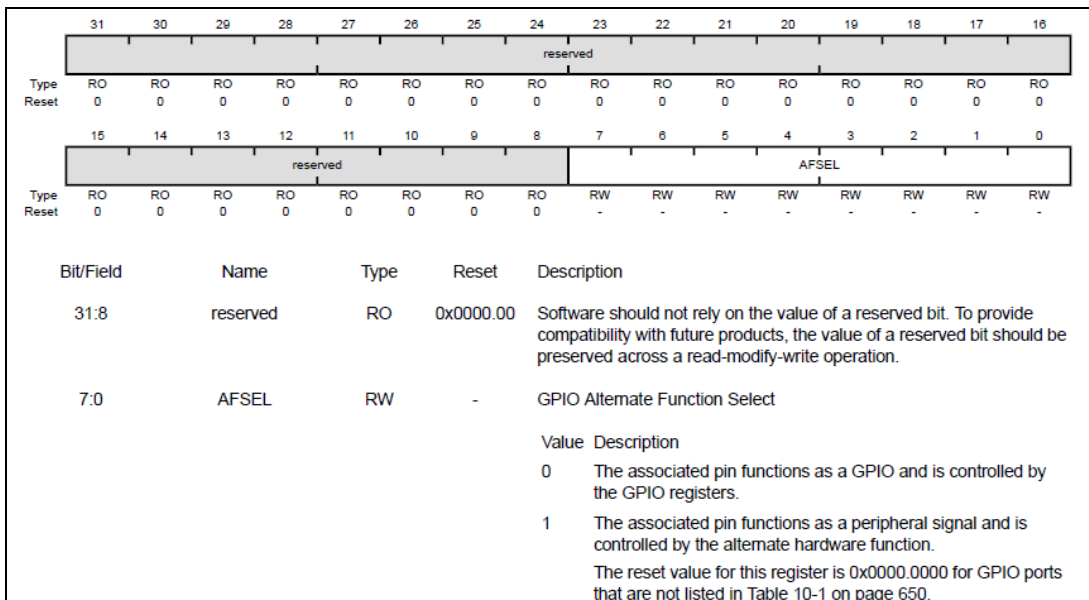


Figura N° 4.13. Registro GPIO AFSEL

Fuente: Data sheet del microcontrolador Tiva Launch Pad



Como se observa en el registro GPIO AFSEL, si un bit es cero, el pin se usa como un GPIO. Si se escribe un “1” lógico en un bit de este registro configura la línea GPIO correspondiente para ser controlada por un periférico asociado, haciendolo jnto con el registro GPIO PCTL.

- **Registro de control GPIO PCTL**

El registro GPIO PCTL se utiliza junto con el registro GPIOAFSEL y cuando se usa en el modo alternativo selecciona la señal específica del periférico que se desea controlar.

Bit/Field	Name	Type	Reset	Description
31:28	PMC7	RW	-	Port Mux Control 7 This field controls the configuration for GPIO pin 7.
27:24	PMC6	RW	-	Port Mux Control 6 This field controls the configuration for GPIO pin 6.
23:20	PMC5	RW	-	Port Mux Control 5 This field controls the configuration for GPIO pin 5.
19:16	PMC4	RW	-	Port Mux Control 4 This field controls the configuration for GPIO pin 4.
15:12	PMC3	RW	-	Port Mux Control 3 This field controls the configuration for GPIO pin 3.
11:8	PMC2	RW	-	Port Mux Control 2 This field controls the configuration for GPIO pin 2.
7:4	PMC1	RW	-	Port Mux Control 1 This field controls the configuration for GPIO pin 1.
3:0	PMC0	RW	-	Port Mux Control 0 This field controls the configuration for GPIO pin 0.

Figura N° 4.14 Registro GPIO PCTL

Fuente: Data sheet del microcontrolador Tiva Launch Pad

El registro GPIOAFSEL se borra al restablecerse, por lo tanto, la mayoría de los pines GPIO se configuran como GPIO por defecto. Cuando se

establece un bit en el registro GPIOAFSEL, se controla la señal GPIO correspondiente al periférico asociado. El registro GPIOCTL selecciona uno de un conjunto de funciones periféricas para cada GPIO.

- **Registro de control RCGC0**

El registro RCGC0, permite activar el reloj del generador PWM0 escribiendo un "1" lógico en el registro RCGC0.

Run Mode Clock Gating Control Register 0 (RCGC0)																
Base 0x400FE000																
Offset 0x100																
Type RO, reset 0x0000.0040																
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	reserved			WDT1	reserved		CAN1	CAN0	reserved			PWM0	reserved		ADC1	ADC0
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	reserved				MAXADC1SPD		MAXADC0SPD		reserved	HIB	reserved		WDT0	reserved		
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Figura N° 4.15 Registro RCGC0  
Fuente: Data sheet del microcontrolador Tiva Launch Pad

Determinación del periodo PWM para el generador 0

$$Periodo = \frac{PWM0.LOAD + 1}{fcpu/DIV}$$

Donde:

DIV:valor del preescalador. Se configura en el registro RCC.Su valor predefinido es 1.

Determinación del ciclo de trabajo (C.T.) de la onda PWM para el generador 0	
$C.T. = \frac{PWM0LOAD - PWM0CMPA}{PWM0LOAD + 1} \cdot 100\%$	$C.T. = \frac{PWM0CMPA + 1}{PWM0LOAD + 1} \cdot 100\%$

Fuente: Ing. Hugo Pratt: Módulo PWM en TM4C123GH6PM

#### 4.2.8 Función de transferencia del sistema en lazo cerrado

Para el análisis de utilizó los conceptos de control discreto, cuyo diagrama de bloques se muestra en la siguiente figura 4.16, a fin de analizar la estabilidad del diseño propuesto.

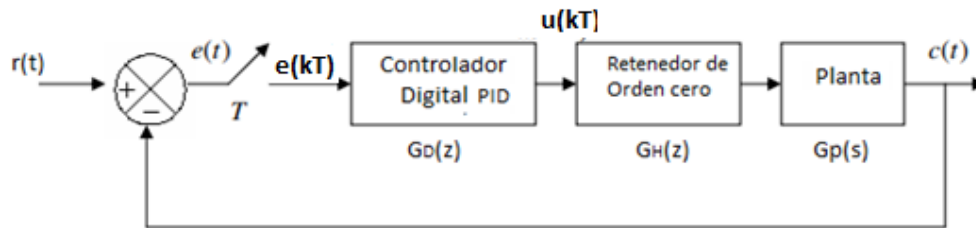


Figura N° 4.16 Diagrama de bloques de un sistema de control discreto  
Fuente: Ogata Katsuhiko. Sistemas de Control en tiempo discreto

Donde  $G_D(z)$  es la función de transferencia del controlador digital,  $G_H(z)$  corresponde al retenedor de orden cero y  $G_p$  a la planta definido como un sistema de primer orden y cuya función de transferencia relaciona la velocidad con la que avanza el robot con la tensión de entrada:

$$G_p = \frac{K}{1 * TS}$$

Considerando:  $G(z) = G_H(z) * G_p(z)$

La función de transferencia del sistema estará dado por:

$$\frac{C(z)}{R(z)} = \frac{G_D(z)G(z)}{1 + G_D(z)G(z)}$$

Y retenedor de orden cero, con periodo T es igual a:

$$G_H(s) = \frac{1 - e^{-sT}}{s}$$

Cuando la entrada al sistema  $G(s)$  es una señal muestreada mediante impulsos la función de transferencia será:

$$G(z) = Z[G(s)]$$

Para el análisis del robot móvil se considera la velocidad que tendrá cada rueda, cuyos componentes se muestran en el siguiente gráfico:

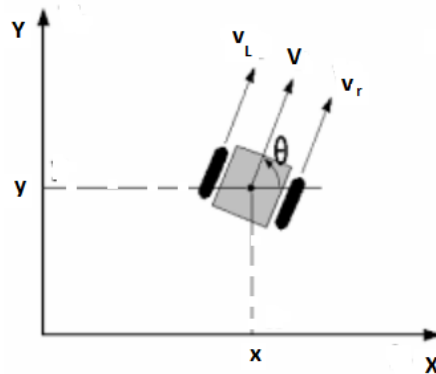


Figura N° 4.17 Modelo cinemático del robot móvil  
Fuente: Robótica Manipuladores y robots móviles

Donde:

$$V_l = W_l * r \quad ; \quad v = \frac{V_r + V_l}{2} \quad ; \quad w = \frac{V_r - V_l}{l};$$

$$V_r = W_r * r$$

A partir de estas ecuaciones obtenemos:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -\text{sen}\theta & 0 \\ \text{cos}\theta & 0 \\ 0 & 1 \end{bmatrix} * \begin{bmatrix} v \\ w \end{bmatrix}, \quad \text{Luego:} \quad \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -r\text{sen}\theta/2 & -r\text{sen}\theta \\ r\text{cos}\theta/2 & r\text{cos}\theta \\ -r/l & r/l \end{bmatrix} * \begin{bmatrix} w_l \\ w_r \end{bmatrix}$$

#### 4.3 Población y muestra

La población de los robots aumenta cada día por cada aplicación en diversos campos. Se presenta en la industria, en el hogar y en la actividad comercial como los robots móviles que vemos a diario.

#### 4.4 Técnicas e instrumentos de recolección de datos

Se utiliza la información la bibliografía y publicación relacionada a la robótica móvil principalmente de entidades competentes, tesis publicadas y temas de investigación en dicha área.

#### 4.5 Procedimientos de recolección de datos

No se aplica.

#### 4.6 Procesamiento estadístico y análisis de datos

No corresponde

## CAPITULO V

### V RESULTADOS

- a. Se ha realizado las pruebas de funcionamiento del robot móvil con el microcontrolador ATmega8 de AVR ATMEL. El programa se implementó en lenguaje ensamblador utilizando interrupciones externas. El robot se desplazó de acuerdo a las indicaciones del mando. El programa en lenguaje ensamblador se encuentra en Anexos.



Fig.5.1 Robot bycibot como modelo de estudio  
Fuente: Laboratorio de proyectos electrónicos. PUCP

- b. Para mejorar el sistema sensorial del robot móvil se diseña con el microcontrolador TIVA TM4C123GH6PM de Texas Instrumentes que se muestra en la figura 5.2 con mejores prestaciones para detectar obstáculos con sensores ultrasónicos, un mando automático y parada automática utilizando un sensor bluetooth. Los

objetos que traslade el robot en una plataforma (canastilla) serán pequeños y de poco peso no mayor a 500 gr. Los programas para el microcontrolador indicado se encuentran en anexos.

- c. Utilizando ondas PWM para los motores de corriente continua tendrá un mejor desplazamiento como se observa en la figura Fig.5.3, toda que se podrá regular la velocidad al variar su ciclo de trabajo.

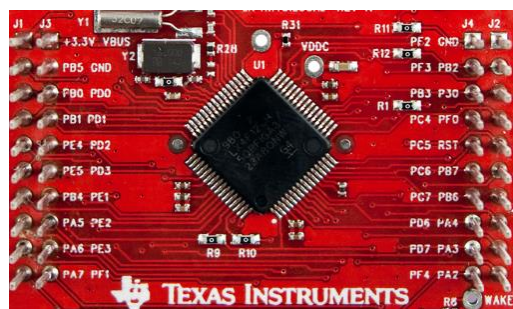


Fig.5.2 Microcontrolador TIVA TM4C123GH6PM  
Fuente: Página web de Texas Instruments

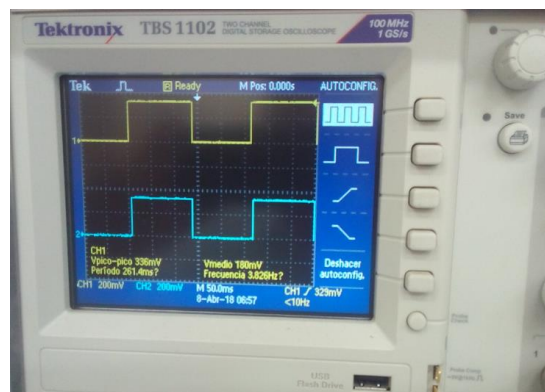
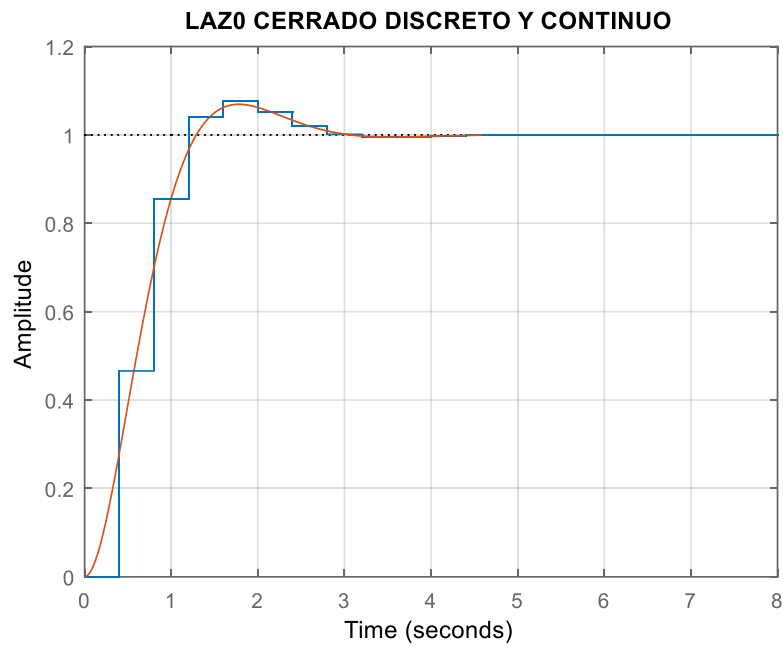


Fig.5.3 Generación de ondas PWM con el microcontrolador TIVA  
Fuente: autoría propia.

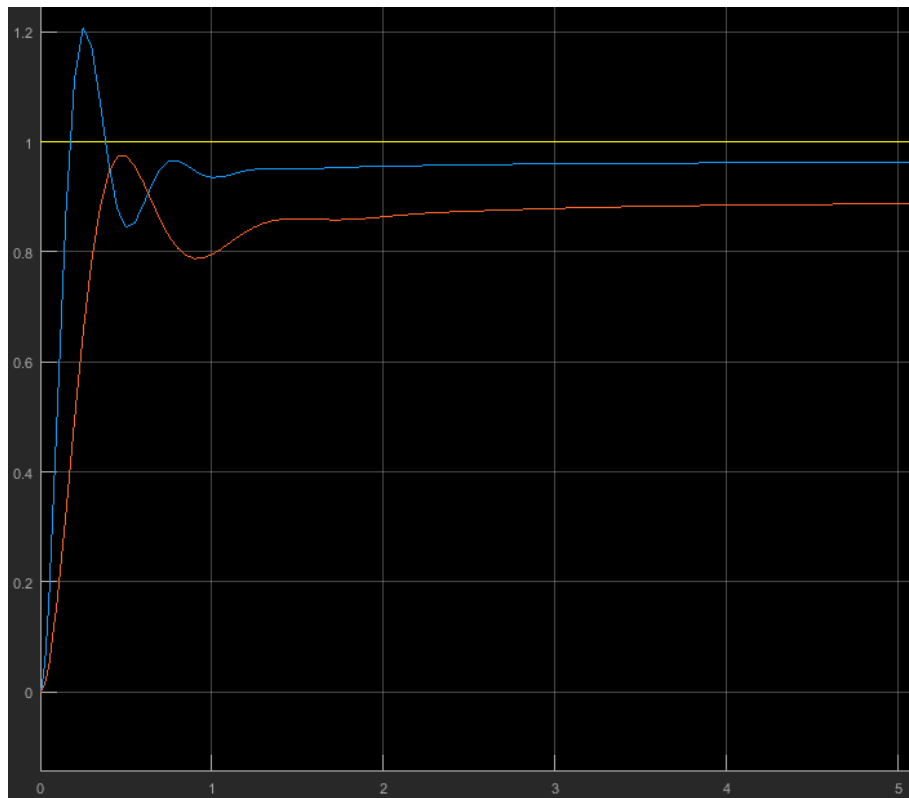
- d. El mando inalámbrico para dar inicio al movimiento del robot móvil se realiza con el sensor bluetooth que se coloca en el robot móvil y durante su desplazamiento pueda ser a la vez detenido y reiniciado cuando sea necesario.

- e. El algoritmo de la programación es modular a fin de permitir colocar otro tipo de sensores utilizando funciones en lenguaje C y dar mayor inteligencia al robot durante su trayectoria.
- f. Se mejora el movimiento del robot durante su trayectoria si se utiliza el control PID discreto, cuyos resultados del análisis mediante técnicas computacionales Matlab y simulink se indican en las siguientes figuras Fig.5.4 y Fig.5.5, para ello se utilizó un controlador discreto PID, un retenedor de orden cero, con un sistema de primer orden compensado y su respuesta es considerando la señal escalón como entrada.



**Fig.5.4** Respuesta del sistema discreto a diferentes periodos de muestreo con retenedor de orden cero

Fuente: autoría propia.



**Fig.5.5** Respuesta a una señal escalón de una señal continua y discreta  
Fuente: autoría propia.

- g. La presencia de obstáculos en su trayectoria ha sido posible detectarlos a una distancia establecida, dando un tiempo suficiente para que el camino quede libre y el robot móvil pueda continuar con su movimiento.



## **CAPITULO VI**

### **VI DISCUSIÓN DE RESULTADOS**

#### **6.1 Contrastación de la hipótesis con los resultados**

En el capítulo III se planteó la Hipótesis de la investigación de la siguiente manera: “Desarrollando un sistema sensorial para un robot móvil de servicio social denominado Bycibot permitirá trasladar de manera automática objetos desde un lugar a otro, siguiendo una trayectoria predefinida, evadiendo obstáculos hasta llegar a su destino”.

Se ha considerado en el desarrollo del robot el empleo de diversos sensores indicado en el capítulo IV colocado para un desplazamiento adecuado llevando objetos, evitando chocar hasta llegar al lugar designado. Se ha verificado el funcionamiento de cada parte del robot en forma modular para su acoplamiento y seguir mejorando en futuras versiones.

#### **6.2 Contrastación de resultados con otros estudios similares**

En otros robots anteriores han utilizado microcontroladores de 8 bits de otros fabricantes cuya prestación es menor con respecto al microcontrolador de 32 bits de Texas Instruments utilizado en el diseño del presente trabajo a diferencia de otros robots similares el presente robot tiene mayormente una aplicación de servicio.

## VII CONCLUSIONES

- La implementación de un sistema sensorial brinda mayor grado de inteligencia al robot móvil con su entorno utilizando para ello componentes electrónicos de última generación para desplazarse siguiendo una trayectoria predefinida, evadiendo obstáculos y llevando objetos hasta llegar a su destino.
- El empleo de un microcontroladores de 32 bits, el TM4C123H6PM de Texas Instruments, permite disminuir el hardware del sistema al contar con periféricos internos que son programados en lenguaje C o en lenguaje ensamblador aplicado al diseño de robots móviles.
- El diseño de robots móviles debe seguir estándares de calidad y seguridad previendo cualquier accidente, lo cual se logra al utilizar un microcontrolador de última generación y contar con sensores externos para la marcha del robot, detección de obstáculos, proximidad y las señales internas que controlan los periféricos internos para manejar los motores con ondas PWM e indicar alarma en casos imprevistos.
- El monitoreo del entorno donde se desplaza el robot móvil utilizando técnicas de control digital para controlar su desplazamiento tiene mayores ventajas que un control analógico, está conformado por el subsistema PID discreto, retenedor de orden cero, y la planta compensado en lazo cerrado, habiéndose comprobado con herramientas computacionales.

## **VIII RECOMENDACIONES**

- El ambiente de trabajo del robot debe tener señalizaciones para dejar libre la trayectoria del robot y evitar lecturas erróneas.
- En los futuros diseños se debe incluir mayores sensores de sonido de audio y voz para un mejor desenvolvimiento de un robot.
- Se debe plantear para futuros diseños una localización y recorrido más complejo utilizando el concepto de Odometría en robots móviles.
- Se puede considerar la alternativa de utilizar los conceptos de redes neuronales para implementar la inteligencia artificial del robot en manejar su entorno y navegar en forma autónoma.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] Bermejo, S. (2003). *Desarrollo de robots basados en el Comportamiento*. Catalunya: Ediciones de la Universidad Politécnica de Catalunya.
- [2] Iyengar, S. (1991). *Autonomous Mobile Robots*. (Vol 2). U.S.: IEEE Computer Society Press.
- [3] Fahimi, F. (2009). *Autonomous robots: modeling, path planning, and control*. Boston: Springer
- [4] Rashid S. (2004). *Electrónica de Potencia Circuito, Dispositivos y Aplicaciones*. 3<sup>ra</sup> edición. México: Pearson Educación S.A.
- [5] Gracia, L. (2006). *Modelado cinemático y control de robots móviles con ruedas* (Disertación doctoral). Universidad Politécnica de Valencia. España.
- [6] Corrales, A. (2012). *Sistema de Navegación para Robots Sociales Basado en Señales* (Disertación doctoral). Universidad Carlos III de Madrid. Madrid.
- [7] Ogata Katsuhiko.(2002). *Sistemas de Control en Tiempo Discreto*. 2<sup>da</sup>. Edición. México: Prentice Hall Hispanoamericana S.A.
- [8] Dorf Richard C. y Bishop Robert H. (2005). *Sistemas de control moderno*. 10<sup>a</sup> edición. Madrid: Pearson Educación S.A.
- [9] Hernández, R. (2014). *Metodología de la Investigación*. 6<sup>a</sup> edición. México: McGraw-Hill.
- [10] Bernal, C. (2014). *Metodología de la Investigación*. 3<sup>a</sup> edición. Bogotá: Pearson Educación.

- [11] Ogata K. y Prentice C. (2003). *Ingeniería de Control Moderna*. 4ª edición. México: Prentice Hall Hispanoamericana S.A.
- [12] Fu k.S. González R.C. (1993). *Robótica: Control, Detección, Visión e Inteligencia*. 1ª edición. Mexico: McGraw-Hill Interamericana editores, S.A.
- [13] Ollero A. (2001). *Robótica: Manipuladores y robots móviles*. 1ª edición. Barcelona: Marcombo, S.A.
- [14] Fernández del Busto R. (2013). *Análisis y diseño de sistemas de control digital*. 4ª edición. . México: McGraw-Hill Interamericana editores, S.A.
- [15] Lewis D. (2013). *Fundamentals of embedded software with the ARM Cortex-M3*. 2<sup>th</sup> edition. New Jersey: Pearson Education, Inc.

**PAGINAS WEB:**

- [1] <http://www.monografias.com/trabajos83/la-robotica/la-robotica.shtml>
- [2] <http://www.zav.es/docs/portillo/BarrerasMD.pdf>
- [3] <http://wiki.robotica.webs.upv.es/>
- [4] [www.tendencias21.net](http://www.tendencias21.net)
- [5] [http://robots-argentina.com.ar/Energia\\_baterias.htm](http://robots-argentina.com.ar/Energia_baterias.htm)
- [6] <http://publimetro.pe/vida-estilo/noticia-medicina-robotica-futuro-salud-44740>
- [7] [www.atmel.com](http://www.atmel.com)
- [8] [www.ti.com](http://www.ti.com) ; de Texas Instruments
- [9] <http://cortex-m.com/tm4c-lessons/>
- [10] <http://users.ece.utexas.edu/~valvano/Volume1/E-Book/>
- [11] <http://embedded-lab.com/blog/tiva-c-gpios/>
- [12] [http://portal.ku.edu.tr/~cbasdogan/Courses/Robotics/projects/Discrete\\_PID.pdf](http://portal.ku.edu.tr/~cbasdogan/Courses/Robotics/projects/Discrete_PID.pdf)

## **ANEXOS**

### **ANEXO A:**

A1) PROGRAMA FUENTE EM LENGUAJE ENSAMBLADOR PARA EL MICROCONTROLADOR ATMEGA8 DE ATMEL (MICROCHIP).

A2) APLICACIONES UTILIZADOS CON EL MICROCONTROLADOR TM4C123GH6PM DE TEXAS INSTRUMENTS

A3) PROGRAMAS EN LENGUAJE C

### **ANEXO B:**

ANÁLISIS DEL SISTEMA CON MATLAB

B1) ANALISIS CON EL RETENEDOR DE ORDEN CERO

B2) ANALISIS DEL SISTEMA CON CONTROL PID DISCRETO

### **ANEXO C:**

MATRIZ DE CONSISTENCIA.

### **ANEXO D:**

DIAGRAMAS

HOJAS TÉCNICAS

## ANEXO A:

A1) Programa fuente em lenguaje ensamblador para el microcontrolador Atmega8 de ATMEL(microchip).

```

; *****
; Programa: Robot_movil_1
; Autor: Zenón Cucho Mendoza
; Microcontrolador: ATmega8
; Descripción:
; Mueve 3 motores DC para desplazamiento del robot
; Conexiones:
; PB0 Y PB1 AL MOTOR 1
; PB2 Y PB3 AL MOTOR 2
; PB4 Y PB5 AL MOTOR 3
; *****
.include "C:\include\m8def.inc"

.CSEG
.ORG $00
rjmp INICIO

; Interrupciones
.ORG $01
rjmp Int_INT0 ; Vector del INTO

; *****
; ***** RUTINA DE SERVICIO DE INTERRUPCION *****
; *****
Int_INT0:
    push R16 ; Salvamos registros en pila
    in R16,SREG
    push R16
    LDI r19,5
    pop R16
    out SREG,R16
    pop R16 ; Recuperamos registros del pila
    reti ; Salimos de la interrupción

; *****
; ***** Subrutinas *****
; *****

Init_INT0:    ; interrupción externa INTO
    in r16,MCUCR ; Solicitud de Interrupción en flanco de
    ori r16,$03 ; subida
    out MCUCR,r16
    in r16,GICR ; Habilitamos la interrupción externa 0
    ori r16,$40
    out GICR,r16
    ret

```

```

Init_Puertos:
    LDI R16, $00 ; PINES DE ENTRADA
    OUT DDRC,R16
    OUT DDRD,R16 ;
    LDI R18,$FF
    OUT DDRB, R18 ; PINES DE SALIDA PARA LOS MOTORES
    OUT PORTB,R16
    RET
; Programa principal
INICIO:
    cli
    ldi r16,high(RAMEND)      ; Define puntero de pila
    out SPH,r16
    ldi r16,low(RAMEND)
    out SPL,r16
    rcall Init_Puertos ; Inicializamos el sistema
    rcall Init_INT0
    LDI R19,0
EsperaPulso_Start:  ; Lectura del pulsador
    sbis     PIND,3
    rjmp     EsperaPulso_Start
EsperaSoltar:
    sbic     PIND,3
    rjmp     EsperaSoltar
    SEI
LECTURA1:
    CPI R19,5
    BRNE LECTURA2

;CONDICION INICIAL:MOTORES GIRANDO
    LDI R16,0
    OUT PORTB,R16
    LDI R19,0
    RJMP EsperaPulso_Start
LECTURA2:
    IN R20, PINC ; LECTURA DE 4 INTERRUPTORES
    ANDI R20,$03 ; filtro 2 pines b0 y b1 de la rotacion
    CPI R20,01
    BREQ M1H_M2A
    CPI R20,02
    BREQ M1A_M2H

```



```

;Motores_Detenidos:
;      LDI R16,$00 ; Detiene los 2 motores
;      OUT PORTB,R16
;      RJMP LECTURA1 ;Condicion_Inicial
M1H_M2A:
      LDI R16,0b00000110      ;b0 y b1 para el motor 1, b2 y b3 para el motor 2
      OUT PORTB,R16

EsperaSoltar_P1:
      sbic      PINC,0
      rjmp EsperaSoltar_P1
      RJMP LECTURA1
M1A_M2H:
      LDI R16,0b00001001      ;b0 y b1 para el motor 1, b2 y b3 para el motor 2
      OUT PORTB,R16

EsperaSoltar_P2:
      sbic      PINC,1
      rjmp EsperaSoltar_P2

      RJMP LECTURA1

```

## A2) APLICACIONES UTILIZADOS CON EL MICROCONTROLADOR TM4C123GH6PM DE TEXAS INSTRUMENTS

### COMUNICACION SERIAL

#### PuTTY

Es una aplicación portable que nos permite conectarnos entre dos computadoras que tienen diferentes CPUs, en este caso el Módulo TIVA TM4C123H6PM y la PC. La aplicación es gratuita, con interfaz sencilla que se descarga de Internet

#### Pasos a seguir:

1. Ejecute la aplicación Putty

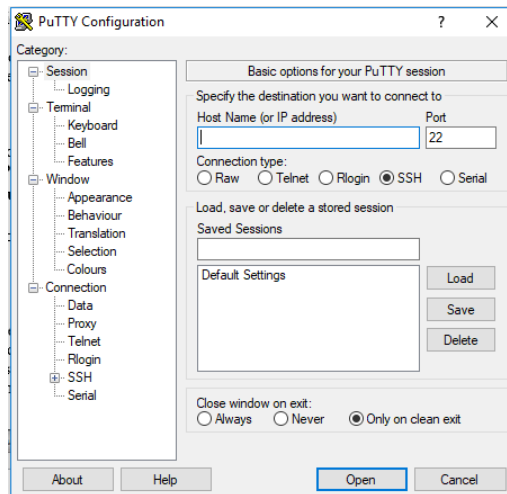


Fig.1

2. Crea una sesión de nombre Serial\_Tiva, seleccione en **Connection** la pestaña **Serial** e ingrese los parámetros de Transmisión serial del UART del TIVA y la PC, como se observa en las siguientes figura: (el puerto COM6 puede cambiar)

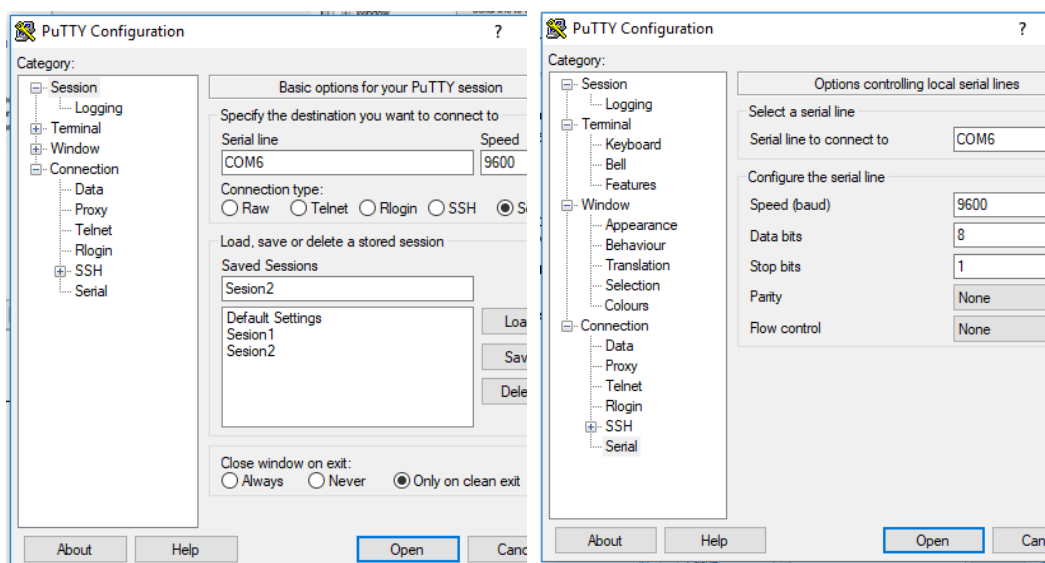


Figura 2

Figura 3

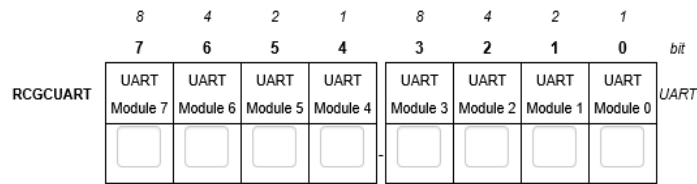
## UART Initialization & Configuration

The following steps show how to initialize PWM peripheral.

### 1. Enable Clock to the UART Modules (RCGCUART register)

The first step to configure UART is to control the clocking for the UART modules. When enabled, an UART module is provided a clock and accesses to module registers are allowed. When disabled, the clock is disabled to save power and accesses to module registers generate a bus fault.

- ▶ 0: UART module is disabled
- ▶ 1: Enable and provide a clock to UART module in RUN mode



SYSCTL->RCGCUART =  MyDefines.h

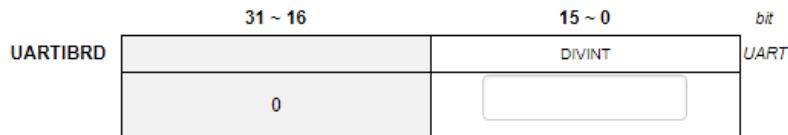
After enable clock signal, check the PRUART register until the corresponding bit set to 1.

In C:

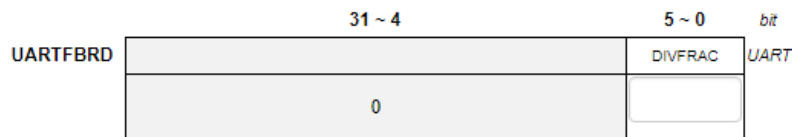
```
while ( (SYSCTL->PRUART & (____) ) != (____) ) {};
```

There are four possible sequences to update the baud-rate registers:

- ▶ UARTIBRD write, UARTFBRD write, and UARTLCRH write
- ▶ UARTFBRD write, UARTIBRD write, and UARTLCRH write
- ▶ UARTIBRD write and UARTLCRH write
- ▶ UARTFBRD write and UARTLCRH write



UART  ->IBRD =  decimal



UART  ->FBRD =  decimal

## El temporizador del sistema (*SysTick timer*)

El microcontrolador tiene un temporizador de sistema de 24 bits, el SysTick, que cuenta descendentemente hasta cero y cuando pasa de “1” a “0”, se activa la señal fin de cuenta, luego en el siguiente flanco de reloj se reinicia desde el valor programado en el registro NVIC\_ST\_RELOAD\_R (abreviado STRELOAD) otra vez en forma descendente.

El temporizador Systick, consta de tres registros:

**SysTick Control and Status (STCTRL):** Registro de configuración, para configurar su reloj, activar el contador, permitir la interrupción SysTick, y determinar el estado del contador.

**SysTick Reload Value (STRELOAD):** Se utiliza para recargar el valor del contador.

**SysTick Current Value (STCURRENT):** Contiene el valor actual del contador.

### SysTick Control and Status (STCTRL):

#### SysTick Control and Status Register (STCTRL)

Base 0xE000.E000

Offset 0x010

Type R/W, reset 0x0000.0000

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	reserved															COUNT	
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	reserved													CLK_SRC	INTEN	ENABLE	
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Donde:

<p><b>COUNT</b>  0: La cuenta no ha llegado a 0  1: La cuenta ha llegado a 0</p>	<p><b>ENABLE</b>  0: SysTick desactivado  1: Systick activado</p>
<p><b>CLK_SRC</b> (Clock source)  0: Reloj externo (NO IMPLEMENTADO)  1: Reloj del sistema (usar este valor)</p>	<p><b>INTEN</b> (Interrupt enable)  0: No genera interrupciones  1: Interrupciones habilitadas</p>

### A3) PROGRAMAS EN LENGUAJE C

```
//Descripción: comunicación serial entre la PC y el módulo Tiva
//Autor: Zenon Cucho
//Referencia:Valvano.J.W.

#include"tm4c123gh6pm.h"
#include"stdint.h"

/*Funcion de configuración del UART*/

void config_UART0 (void){
    unsigned long retardo1;
    // Se activa el reloj del UART0
    SYSCTL_RCGC1_R |= SYSCTL_RCGC1_UART0;
    retardo1 = SYSCTL_RCGC2_R; // Espera unos ciclos de reloj
    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOA; // Se activa el reloj del puerto A
    // PA0 (U0Rx) PA1( U0Tx)
    retardo1 = SYSCTL_RCGC2_R; // Espera unos ciclos de reloj
    // Se desactiva el UART
    UART0_CTL_R &= ~ UART_CTL_UARTEN;
    // Configuración del UART0
    //Reloj del UART a 16 Mhz --> Parte entera es 16MHz/(16*9600)
    UART0_IBRD_R = 104; // IBRD = int(16,000,000 / (16 * 9600)) = int(104.1616)
    UART0_FBRD_R = 11; // FBRD = round(0.1666 * 64 + 0.5 ) = 11

    // Se configuran los bits de datos, 1 bit de parada, sin paridad y habilita
    // el FIFO

    UART0_LCRH_R = ((UART0_LCRH_R & ~0x000000FF)|(UART_LCRH_WLEN_8)|(UART_LCRH_FEN));
    // Se habilita el UART
    UART0_CTL_R |= UART_CTL_UARTEN;
    GPIO_PORTA_AFSEL_R |= 0x03; // Se activan las funciones alternas de PA0 y PA1.
    GPIO_PORTA_DEN_R |= 0x03; // Habilitación PA0 y PA1 para señales digitales.
    GPIO_PORTA_PCTL_R = (GPIO_PORTA_PCTL_R&0xFFFFF00)+0x00000011;
    GPIO_PORTA_AMSEL_R &= ~0x03; // Se desactiva la función analógica del
    // puerto A
}

int main(void){
    config_UART0();
}
```

//COMUNICACIÓN BLUETOOTH

//Utilizando UART1

//Parametros de Tx:9600 bps,8bits de datos, sin paridad, 1 bit de parada.

```

//***** CONFIGURACION UART1: COMUNICACION MEDIANTE BLUETOOTH*****
// Pin PC4: Rx datos y Pin PC5 tx datos

void conf_UART1(void){

    SYSCTL_RCGC1_R |= SYSCTL_RCGC1_UART1; // Se activa la señal de reloj del UART1
    while((SYSCTL_PRUART_R&0x02)==0);
    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOC; //activar la señal de reloj del puerto C
    while((SYSCTL_PRGPIO_R&0x04)==0); // espera
    UART1_CTL_R &= ~UART_CTL_UARTEN; // deshabilitar el módulo UART
    //Configurando para 9600 bps a f=16Mhz
    // 8 bits de data y un bit de parada
    UART1_IBRD_R = (UART1_IBRD_R & ~0xFFFF)|104; //Parte entera
    UART1_FBRD_R = (UART1_FBRD_R & ~0x3F)|11; //parte fraccionaria
    UART1_LCRH_R = (UART1_LCRH_R|0x70) &~ 0x0A;
    UART1_CTL_R = (UART1_CTL_R|0x301) &~0x20; // habilitar uart y reg.FIFO
    GPIO_PORTC_AFSEL_R |= 0x30; //Se activan las funciones alternas de PC4 y PC5
    //conectamos UART a PC4 y PC5
    GPIO_PORTC_PCTL_R = ((GPIO_PORTC_PCTL_R & 0xFF00FFFF)|0x00110000);
    GPIO_PORTC_DEN_R |= 0x30; //Habilitación PC4 Y PC5 para señales digitales.
    GPIO_PORTC_AMSEL_R &= ~0x30; // desactivar señal analógica
    }

    unsigned char esperaDato_UART1(void){
    unsigned char datoRx;
    while((UART1_FR_R & UART_FR_RXFE)!=0); // ¿Buffer de Rx esta vacío?
    datoRx=UART1_DR_R;
    return datoRx;
    }

void TransmiteByte(unsigned char datoTx){
    while(UART1_FR_R & UART_FR_TXFF);
    UART1_DR_R = datoTx;
    }

void TransmiteCadena(const unsigned char CadenaASCII[]){
    int i;
    for(i=0; CadenaASCII[i]; i++)
        TransmiteByte(CadenaASCII[i]);
    }

```

```
// ++++++
```

```

//***** FUNCION DISTANCIA_ULTRASONIDO *****

int distancia_ultrasonido (void)
{
    uint32_t tiempo;
    unsigned int cuenta ,f;
    uint8_t distancia;
    cuenta=0;
    int trigger=4;
    int echo=5;
    //config_timer(); //funcion timer
    GPIO_PORTB_DATA_R&=(0<<trigger) ; // Inicialmente salida en "0"
    delay_Systick_10us(); //delay
    GPIO_PORTB_DATA_R|= (1<<trigger) ;// Salida en "1" lógico para
        //el flanco de subida
    delay_Systick_10us(); //tiempo del pulso de disparo
    GPIO_PORTB_DATA_R|= ~(1<<trigger) ;//Pulso a cero
    TIMER0_ICR_R |= 0x01;// nos aseguramos que este Limpio el RIS
    while (( GPIO_PORTB_DATA_R & (1<<echo) ) ==0 ){ } ;//espera
    TIMER1_CTL_R |= (TIMER_CTL_TAEN); //Activación del timer 1
    while((GPIO_PORTB_DATA_R &( 1<<echo)) != 0); //espero un flanco de
        // bajada por parte del "echo"
    TIMER1_CTL_R &= ~(TIMER_CTL_TAEN); //Desactivación del timer
    cuenta = TIMER1_TAR_R; //
    //F=16 Mhz
    tiempo = ((0xFFFFFFFF-cuenta)*0.0625); // T=1/F en uSeg.;
    distancia=tiempo/29/2; // distancia en centímetros
        // (Velocidad del sonido=340 m/s)
    TIMER1_TAILR_R=0xFFFFFFFF; //inicializamos el valor de la carga.
    return distancia ;

}

//Configura el temporizador del SYstick a 10 us, para el sensor
//ultrasonico
void delay_Systick_10us(void) {
    // Deshabilita el Systick para proceder a la configuración
    NVIC_ST_CTRL_R &= ~NVIC_ST_CTRL_ENABLE;
    / Carga el tiempo entre encendido y apagado del led.
    NVIC_ST_RELOAD_R|= 0x4F; //10us=(1/16Mhz)(contador+1)

//Habilitar el Systick y la señal de reloj

```

```

NVIC_ST_CURRENT_R = 0;           // Renicia la cuenta actual
/
NVIC_ST_CTRL_R |= NVIC_ST_CTRL_ENABLE + NVIC_ST_CTRL_CLK_SRC;
NVIC_ST_CTRL_R |= NVIC_ST_CTRL_ENABLE + NVIC_ST_CTRL_CLK_SRC;
}

// Configura el Timer1 para cuenta descendente
void Configura_Timer1(void){
    unsigned long temp;
    SYSCTL_RCGC1_R = SYSCTL_RCGC1_TIMER1;
    temp=SYSCTL_RCGC1_R;
    TIMER1_CTL_R &= ~TIMER_CTL_TAEN;
    TIMER1_CFG_R = 0x00;
    TIMER1_TAMR_R = 0x02;
    TIMER1_TAILR_R = ~0;
    TIMER1_ICR_R |= 0x01;
}

```

```

//*****
//GENERACION DE ONDAS PWM
//Referencias: Valvano, J.W. Embedded systems
//              Laboratorio de Sistemas digitales-PUCP
//*****

void GeneraPWM(void){
//Se configura el pin de salida PD0 para la onda PWM
//Se utiliza el generador PW1_0
    unsigned long temp;
//*****se habilita el puerto PD0 *****
//Configuracion para la señal de reloj en puerto D
    SYSCTL_RCGC0_R |= SYSCTL_RCGC0_PWM0;
    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOD; //Para PWM en PD0
    temp = SYSCTL_RCGC2_R; //retardos
    GPIO_PORTD_DIR_R |= 0x01;
    GPIO_PORTD_AMSEL_R &= ~(0x01);
    GPIO_PORTD_AFSEL_R |= 0x01;
// PD0 conectado a M1PWM0
    GPIO_PORTD_PCTL_R = (GPIO_PORTD_PCTL_R & 0xFFFFF0) | 0x05;
    GPIO_PORTD_DEN_R |= 0x01;
}

```



```

// Activamos señal de reloj para módulo PWM1
SYSCTL_RCGCPWM_R |= SYSCTL_RCGCPWM_R1;
while(!(SYSCTL_PRPWM_R & SYSCTL_PRPWM_R1));
// Configuramos divisor de frecuencia con 4
SYSCTL_RCC_R |= SYSCTL_RCC_USEPWMDIV;
SYSCTL_RCC_R &= ~SYSCTL_RCC_PWMDIV_M;
SYSCTL_RCC_R |= SYSCTL_RCC_PWMDIV_4;
// Generador 0: Modo cuenta descendente, inhabilitado
PWM1_0_CTL_R &= ~(0x03FF);
PWM1_0_GENA_R = (PWM1_0_GENA_R & 0xFFFFF000) | 0x08C;
// frecuencia base = 100Hz
PWM1_0_LOAD_R = (PWM1_0_LOAD_R & 0xFFFF0000) | 0X3E7F;
// Ciclo de trabajo = 50%
PWM1_0_CMPA_R = (PWM1_0_CMPA_R & 0xFFFF0000) | 0X1F3F;
// Habilitamos el generador 0
PWM1_0_CTL_R |= PWM_0_CTL_ENABLE;
// Habilitamos la señal PWM
PWM1_ENABLE_R |= PWM_ENABLE_PWM0EN;
}

```

```

//*****
//GENERACION DE ONDAS PWM
//Referencias: Valvano, J.W. Embedded systems
//              Laboratorio de Sistemas digitales-PUCP
//*****
void Genera2PWM(void){
    unsigned long temp2;
    //Obtencion de la onda PWM en el puerto PB6
    //señal de reloj activado en puerto B
    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOB;
    //Se activa la función alterna para trabajar con PWM
    GPIO_PORTB_AFSEL_R |= 0x40; //en PB6
    //Configura el registro PCTL para elegir la función
    GPIO_PORTB_PCTL_R &= ~0x0F000000;
    GPIO_PORTB_PCTL_R |= 0x04000000; // Configura PB6 como PWM0
    //Deshabilita funciones analógicas sobre PB6
    GPIO_PORTB_AMSEL_R &= ~0x40;
    //Habilita la función digital sobre el pin PB6
    GPIO_PORTB_DEN_R |= 0x40;
    // Configurar el divisor PWM: USEPWMDIV = 1.
    //Dividir 16 MHz/64 = Frecuencia de reloj del módulo PWM = 250 KHz
    SYSCTL_RCC_R|= SYSCTL_RCC_USEPWMDIV;
    temp2= SYSCTL_RCC_R;    //retardos de estabilizacion
    temp2= SYSCTL_RCC_R;
}

```

```

//Deshabilitar generador PWM0 previo a la configuración
  PWM0_0_CTL_R &= ~(PWM_0_CTL_ENABLE);
//Configurar el generador PWM0 para modo de cuenta descendente
  PWM0_0_CTL_R &= ~(0x3F); //
// Se limpia el registro PWM0_0_GENA_R antes de configurar
  PWM0_0_GENA_R &= ~(0xFFF);
//Se configura el estado de la señal PWM0_0, se pone a "0"
//(PWM0_0_LOAD_R) o a "1" (PWM0_0_CMPA_R)
  PWM0_0_GENA_R = (PWM_0_GENA_ACTCMPAD_ONE|PWM_0_GENA_ACTLOAD_ZERO);

//Configurar el periodo de la señal PWM0_0 con el GENA (PWM0_0)
//La fuente de reloj del generador PWM0 es 250 KHz y por ende el
//contador PWM0_0_COUNT_R se
//decrementa a dicha frecuencia. Entonces, una cuenta se efectúa con
// T= 1/250 KHz, o 4 us.
//El máximo periodo se calcula: Max_Periodo = 4 us * (LOAD+1), LOAD =
// 65535 o (2^16 - 1)
// Max_Periodo = 262140 us. El valor LOAD se carga en el registro
// PWM0_0_LOAD_R de 16 bits
// Se limpia el registro PWM0_0_LOAD_R
  PWM0_0_LOAD_R &= ~(PWM_0_LOAD_M);
// Se carga el valor LOAD = 65535
  PWM0_0_LOAD_R |= 0xFFFF; //
// Ciclo de trabajo de la señal PWM0= 80%
//(CMPA + 1) = 0.5 (LOAD + 1), CMPA = 52428
//Se limpia el registro PWM0_0_CMPA_R
  PWM0_0_CMPA_R &= ~(PWM_0_CMPA_M);
  PWM0_0_CMPA_R |= 52428;
//Se habilita el generador PWM0
  PWM0_0_CTL_R |= PWM_0_CTL_ENABLE;
//Se activa la salida de la señal PWM0_0 en PB6
  PWM0_ENABLE_R |= PWM_ENABLE_PWM0EN;
}

```

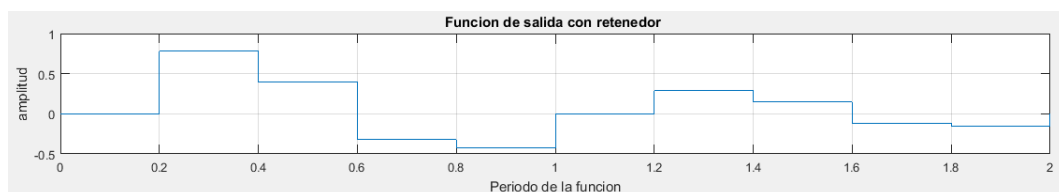


```

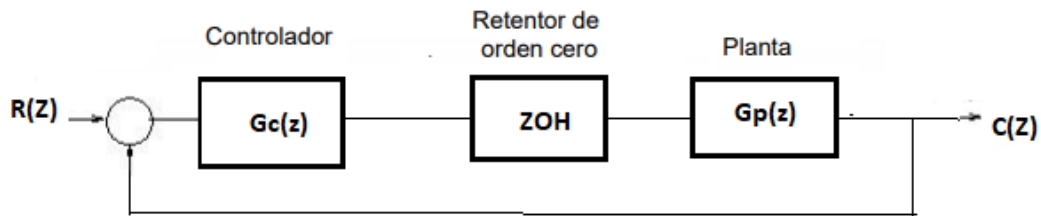
title('Funcion continua')
xlabel('Periodo de la funcion')
ylabel('amplitud')
subplot(3,1,2)
stem(t,y); %funcion de muestreo
grid
title('Funcion muestreada')
xlabel('Periodo de la funcion')
ylabel('amplitud')
subplot(3,1,3)
stairs(t,y);
grid
title('Funcion de salida con retenedor ')
xlabel('Periodo de la funcion')
ylabel('amplitud')

```

Si se utiliza una frecuencia inferior a la frecuencia de la señal, la señal de la salida no será como la original. En la Figura siguiente se muestra para una frecuencia de muestreo de 5 hz.



## B2) ANALISIS DEL SISTEMA CON CONTROL PID DISCRETO



Se demuestra para el sistema PID planteado, la rapidez de la estabilidad depende de los valores de las constantes  $K_p$ ,  $K_i$  y  $K_d$ . Inicialmente se observa un sobreimpulso y tiempo de establecimiento elevado, los cuales se reducen al variar los parámetros  $K_p$ ,  $K_i$  y  $K_d$ ., como se indica en los siguientes graficos

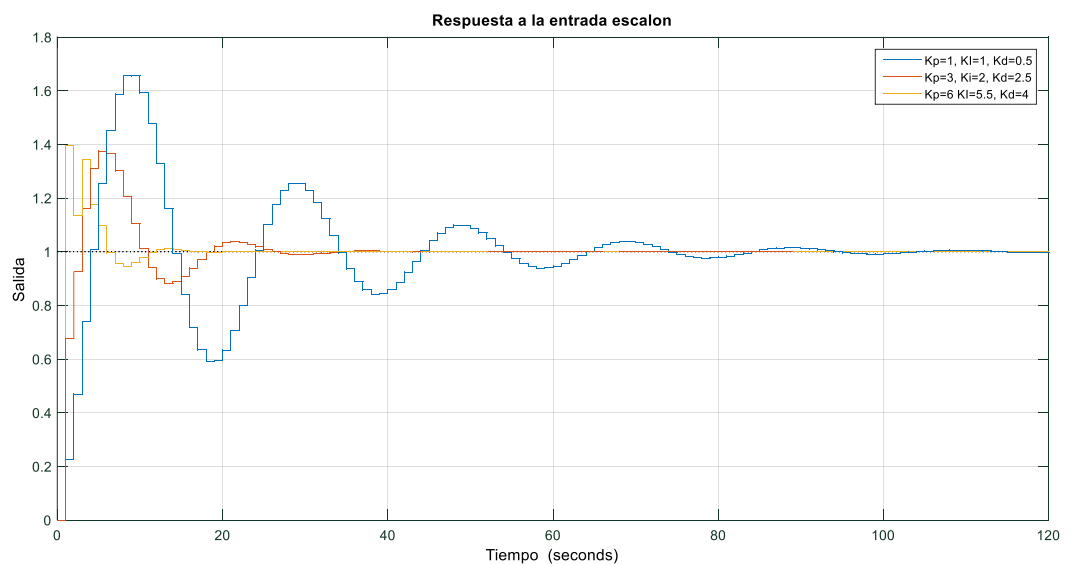


Figura1: Respuesta de diferentes controladores discretos

## Código resumido en matlab

```
%Controlador discreto
%Entrada: escalón
%Calculo de la funcion de transferencia
%Calculo de la transformada z de la planta y el retenedor
%Respuesta a una señal escalón
%Referencia: Ogata Katsuhiko. Sistemas de Control en Tiempo Discreto.
%           Instituto Tecnológico de Culiacán. Rodríguez Robledo.

clear;
clc;

for bucle = 1:1:2

Kp=input('Ingrese Kp:');

Ki=input('Ingrese Ki:');

Kd=input('Ingrese Kd:');
T=input('Ingrese Tiempo de muestreo (Seg):');

%Definiendo la funcion de la planta Gp(s)
Gp = zpk([], [0 -10], 1)
%Discretizacion del modelo continuo:
Gz =c2d(Gp, T, 'zoh'); %

Gz = tf (Gz);

GPID =tf([ Kp+Ki+Kd -(Kp +2*Kd) Kd], [1 -1 0], T);

Lazo_abierto = GPID*Gz;

%Sistema realimentado
Lazo_cerrado =feedback(Lazo_abierto,1)

step(Lazo_cerrado)

grid;

title('Respuesta a la entrada escalon')

xlabel('Tiempo (Seg)')
ylabel('Salida')

hold on

end
```

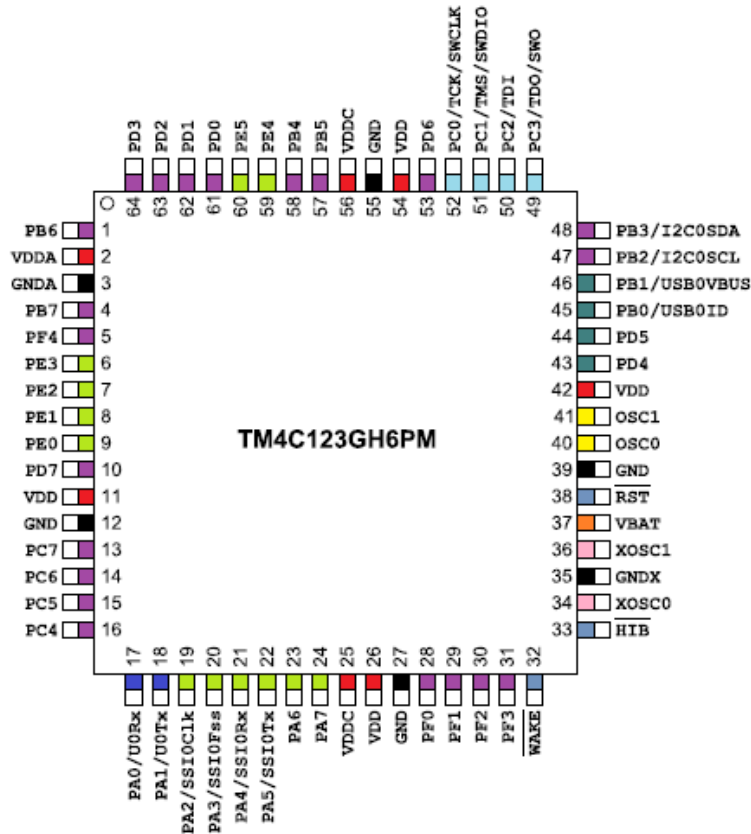
**PARTE C:  
MATRIZ DE CONSISTENCIA**

PROBLEMA	OBJETIVO	HIPOTESIS	VARIABLE DE ESTUDIO	INDICADORES	DISEÑO DE LA INVESTIGACION
<p><b>1.Problema general:</b> ¿Cómo la falta de implementación de un robot de servicio móvil social impide optimizar el tiempo útil en las jornadas diarias?</p> <p><b>2.Problema específico:</b> ¿Cómo la falta de un sistema sensorial afecta el desplazamiento adecuado de un robot móvil de servicio?  ¿Cómo la falta de articulación entre las distintas piezas del robot con el área de trabajo dificulta su desplazamiento óptimo?</p>	<p><b>Objetivo General:</b> Desarrollo de un sistema sensorial para un robot móvil de servicio para realizar trabajos de apoyo al hombre.</p> <p><b>2.Objetivos específicos:</b></p> <ul style="list-style-type: none"> <li>• Estudiar los sensores de última tecnología que se utilizan en el sistema sensorial del robot</li> <li>• Implementar un algoritmo de control para orientar al robot para que pueda recibir señales y tomar una decisión.</li> <li>• Implementar el algoritmo de detección de obstáculos para evadirlos y seguir su trayectoria.</li> </ul>	<p><b>Hipótesis General</b> Desarrollando un sistema sensorial para un robot móvil de servicio social permitirá optimizar las funciones del robot para apoyar los trabajos manuales del hombre mejorando su calidad de vida.</p> <p><b>Hipótesis Específica</b></p> <ul style="list-style-type: none"> <li>• Empleando los sensores de última tecnología en los robots mejorará su funcionalidad para seguir una trayectoria.</li> <li>• Desarrollando un algoritmo con lazos de control orientará al robot al a seguir la trayectoria correcta.</li> <li>• Empleando un lenguaje de programación de alto nivel facilitará el diseño modular del software.</li> </ul>	<p><b>Variable independiente X.</b></p> <ul style="list-style-type: none"> <li>• Sistema de visión.</li> <li>• Trayectoria a seguir.</li> <li>• Sistema</li> <li>• Costo</li> </ul> <p><b>Variable dependiente Y</b></p> <p>Velocidad adecuada en su recorrido. Ubicación del punto de llegada que es la meta. Programación modular</p>	<p><b>Indicadores:</b></p> <ul style="list-style-type: none"> <li>- Funcionabilidad del sistema. X1</li> <li>- Apoyo al trabajo del hombre. X2</li> <li>- Programación modular del software. X3</li> </ul> <p><b>Indicadores:</b></p> <ul style="list-style-type: none"> <li>- Desplazamiento del robot movilY1</li> <li>- Detección de obstáculos con sensores. Y2</li> <li>- Control de la velocidad del robotl. Y3</li> </ul>	<ul style="list-style-type: none"> <li>a. Estudio del estado actual de la robótica móvil de aplicación social.</li> <li>b. Introducción al sistema de sistemas embebidos y sensores</li> <li>c. Estudio de un robot móvil de tres ruedas con nueva tecnología electrónica.</li> <li>d. Aplicación de las herramientas de software para programar microcontroladores.</li> <li>e. Desarrollo del sistema robótico móvil.</li> <li>f. Comprobación de los objetivos propuestos.</li> </ul>

## ANEXO D:

- DIAGRAMAS
- HOJAS TÉCNICAS

### Diagrama del microcontrolador TM\$C123GH6PM



### DIAGRAMA DE LA TARJETA LaunchPad Tiva C

