

**DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**

**CONTROLADOR DE BRAZO ROBOT**

# **T E S I S**

**PRESENTADA COMO REQUISITO PARCIAL  
PARA OBTENER EL GRADO DE:**

**MAESTRO EN INGENIERÍA EN ELECTRÓNICA**

**ING. JUAN MANUEL SOUFFLEE HERNÁNDEZ**

**DIRECTOR DE TESIS:  
M.C. RAFAEL ARMANDO GALAZ BUSTAMANTE**

**HERMOSILLO, SONORA, MÉXICO.**

**DICIEMBRE DE 2016**



INSTITUTO TECNOLÓGICO DE HERMOSILLO

## *Resumen*

Instituto Tecnológico de Hermosillo  
Departamento de Eléctrica y Electrónica

Maestría en Ingeniería en Electrónica

### **Controlador para brazo robot**

by Juan Manuel SOUFFLEE HERNÁNDEZ

El objetivo de este trabajo es desarrollar un controlador para un brazo robot Scorbob de 5 grados de libertad, para ello se implanta un controlador PID en cada una de las articulaciones para el control de velocidad de los motores utilizando modulación por ancho de pulso con inversores de puente completo y se desarrolla la cinemática directa por medio del algoritmo Denavit-Hartenberg para el control de posición y orientación del extremo final del robot. Posteriormente se realiza la programación del dispositivo FPGA que se encarga de ejecutar el controlador. Por último se realizan ensayos para concluir con los resultados obtenidos.

## *Acknowledgements*

'Yo tengo juicio ya, libre y claro, sin las sombras caliginosas de la ignorancia, que sobre él me pusieron mi amargo ir y devenir'.....'Ya conozco mi necedad ...; ya, por misericordia de Dios, escarmentando en cabeza propia estoy'

# Índice general

<b>Resumen</b>	<b>I</b>
<b>Acknowledgements</b>	<b>II</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Antecedentes . . . . .	1
1.2. Justificación de la Tesis . . . . .	3
1.3. Planteamiento del problema . . . . .	3
1.4. Objetivos General y Particulares . . . . .	4
1.4.1. Objetivo General . . . . .	4
1.4.2. Objetivos Particulares . . . . .	4
1.5. Metas y alcances . . . . .	4
<b>2. Marco Teórico</b>	<b>5</b>
2.1. Actuadores . . . . .	5
2.2. Sensores . . . . .	7
2.2.1. Fotodiodos . . . . .	9
2.2.2. Fototransistores . . . . .	10
2.2.3. Encoders . . . . .	10
2.3. Convertidores de potencia . . . . .	12
2.3.1. Inversores . . . . .	13
2.3.2. Funcionamiento básico . . . . .	13
2.3.3. Inversores monofásicos controlados por voltaje . . . . .	15
Modulación por ancho de un solo pulso . . . . .	16
2.4. Proceso de diseño de inversor . . . . .	16
2.5. Robótica . . . . .	16
2.6. Cinemática de Robots manipuladores . . . . .	18
Matrices de rotación . . . . .	19
Matrices de transformación homogénea . . . . .	22
2.7. Sistemas de control . . . . .	25
2.7.1. Respuesta transitoria y en estado estable . . . . .	28
2.7.2. Sistemas de primer orden . . . . .	29
2.7.3. Sistemas de segundo orden . . . . .	32
2.8. Lógica digital con dispositivos FPGA . . . . .	33
2.8.1. Características de una implementación con FPGA . . . . .	34
2.8.2. Lenguaje de descripción de hardware . . . . .	35
2.8.3. Estructura de VHDL . . . . .	35

<b>3. Análisis cinemático</b>	<b>41</b>
3.1. Obtención del modelo Denavit-Hartenberg . . . . .	42
3.2. Implementación del modelo . . . . .	44
3.3. Manipulación del modelo desarrollado . . . . .	45
<b>4. Implementación</b>	<b>49</b>
4.1. Inversor de puente completo L298 . . . . .	50
4.2. Tarjeta FPGA Nexys 2 . . . . .	51
4.3. Fabricación de tarjeta con optoacopladores . . . . .	51
4.4. Software de control . . . . .	51
4.4.1. Diseño de una memoria ROM con valores de la función Seno	51
4.4.2. Obtención de la posición mediante matrices de transfor-	
mación . . . . .	52
<b>5. Resultados y conclusiones</b>	<b>54</b>
<b>Bibliografía</b>	<b>56</b>

# Índice de figuras

1.1. Clasificación de los Robots . . . . .	2
2.1. Clasificación de los Actuadores . . . . .	5
2.2. Clasificación de los motores de CD . . . . .	6
2.3. Modelo de motor de CD en Serie . . . . .	7
2.4. Cuadrantes de operación de un motor de CD . . . . .	7
2.5. Clasificación de los sensores . . . . .	8
2.6. Circuito básico con fotodiodo . . . . .	9
2.7. Fototransistor . . . . .	10
2.8. Señales de salida de un encoder incremental . . . . .	10
2.9. Encoder óptico . . . . .	11
2.10. Disco de un Encoder absoluto . . . . .	12
2.11. Convertidor según el cuadrante de operación . . . . .	13
2.12. Circuito inversor de medio puente y forma de onda de voltaje de salida . . . . .	14
2.13. Corriente en de carga altamente inductiva . . . . .	14
2.14. Circuito inversor de puente completo . . . . .	15
2.15. Filtros típicos de salida . . . . .	16
2.16. Elementos de un brazo robot . . . . .	18
2.17. Representación de un sistema en lazo abierto . . . . .	27
2.18. Representación de un sistema en lazo cerrado . . . . .	27
2.19. Sistema de control con un controlador en cascada . . . . .	28
2.20. Señales típicas de prueba . . . . .	29
2.21. Diagrama a bloques de un sistema de primer orden . . . . .	29
2.22. Respuesta a impulso unitario de un sistema de primer orden . . . . .	30
2.23. Curva de respuesta exponencial . . . . .	31
2.24. Respuesta una entrada rampa de un sistema de primer orden . . . . .	32
2.25. Diagrama a bloques de un sistema de segundo orden . . . . .	33
2.26. Unidades fundamentales del código VHDL . . . . .	36
2.27. Estructura de las librerías y los paquetes . . . . .	36
2.28. Representación de una entidad . . . . .	37
2.29. Código VHDL y representación esquemática . . . . .	37
2.30. Descripción estructural . . . . .	39
3.1. Brazo Robot Scorbot ER-V Plus . . . . .	41
3.2. Elementos de un brazo robot . . . . .	42
3.3. Medidas del robot Scorbot ER-V Plus . . . . .	42
3.4. Secuencia del algoritmo DH . . . . .	43
3.5. Posición inicial del Robot . . . . .	45
3.6. Coordenadas articulares . . . . .	45

3.7. Variando $q_1$ 90 . . . . .	46
3.8. Coordenadas articulares $q_1$ rotado 90 . . . . .	46
3.9. Variando $q_1$ y $q_2$ 90 . . . . .	47
3.10. Coordenadas articulares $q_1$ y $q_2$ rotado 90 . . . . .	47
3.11. Variando $q_3$ 90 . . . . .	48
3.12. Coordenadas articulares $q_3$ rotado -90 . . . . .	48
4.1. Sistema de control para el brazo Robot . . . . .	49
4.2. Inversor Keyes L298 . . . . .	50
4.3. Tarjeta Nexys 2 . . . . .	51
4.4. Construcción simplificada de la tabla de la función seno . . . . .	53

# Índice de cuadros

3.1. Parámetros D-H para el robot Scorbot ER-V plus . . . . .	43
4.1. Necesidades de los motores contra características del inversor . . .	50

# List of Symbols

$a$	distancia	m
$P$	potencia	W ( $\text{Js}^{-1}$ )
$\omega$	frecuencia angular	rad

# Capítulo 1

## Introducción

Los seres humanos a través de su historia siempre han intentado desarrollar artefactos que les ayuden en sus tareas, cada avance que se ha dado a lo largo de este recorrido ha contribuido al desarrollo de la ciencia y la tecnología; desde la invención del ábaco, ideada por mesopotámicos en el 3500 AC, el cual se supone fue el primer instrumento que ayudó al ser humano en la realización de cálculos, también se han encontrado mecanismos usados por los egipcios para extracción de agua que datan del 3000 AC, así podemos seguir hasta el siglo XVII, donde se comienza a dar una gran transformación en todos los aspectos de la vida del hombre.

Todas estas contribuciones dieron un gran impulso a la humanidad durante la revolución industrial, en donde se utilizaron todas las técnicas hasta entonces conocidas y nuevas que comenzaron a desarrollarse para mecanizar procesos, aparece por primera vez en la historia la especialización del trabajo y se comenzaron a estudiar las tareas repetitivas en las que el hombre podía ser reemplazado por máquinas.

Más adelante con el desarrollo de la tecnología y de las técnicas de control en la segunda mitad del siglo XX, el uso de automatismos industriales ha crecido hasta llegar a ser de uso común dentro de la industria.

La robótica, término acuñado por Isac Asimov; es un área de estudio nueva dentro de la ciencia y la tecnología, existen distintas definiciones, y en todas siempre se está hablando de máquinas automáticas que ejecutan un programa para realizar alguna función o tarea, estas máquinas tendrán una parte mecánica, un sistema de control electrónico, un software, e interfaces; si bien es cierto para el enfoque que se le quiera dar siempre habrá que ajustar nuestra definición a lo que queremos lograr, así que habrá diferencias en las definiciones que se encuentren. [4]

### 1.1. Antecedentes

El área de Robótica es multidisciplinaria, esto es, abarca muchas áreas de conocimiento, desde la teoría de control, el control de motores, cinemática, dinámica, resistencia de materiales, diseño estructuras y mecanismos, software para simulación y control, sensores y actuadores, etc. [18] [15]

Los robots se han diversificado y se pueden encontrar muchas clasificaciones según su tipo de aplicación o área de interés, en forma generalizada se han distinguir algunos tipos: [18]

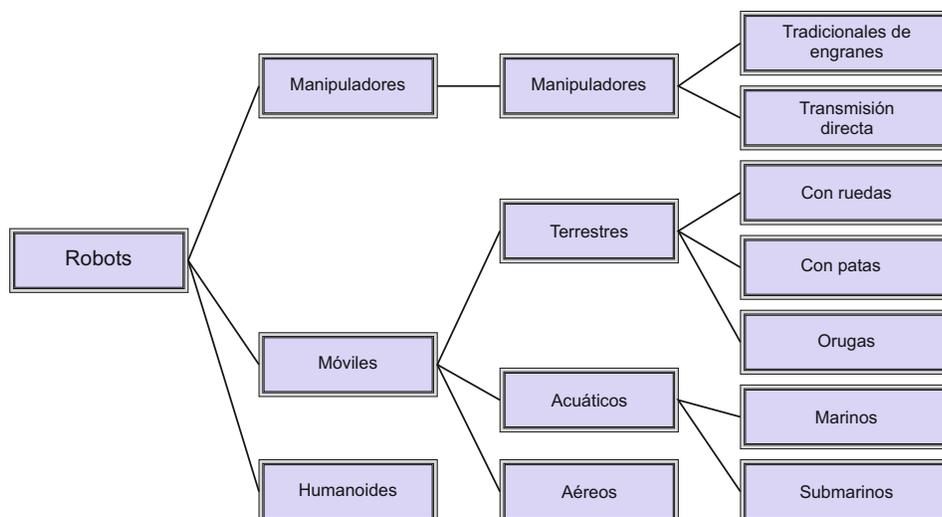


FIGURA 1.1: Clasificación de los Robots

Por ello es indispensable el estudio y desarrollo de los sistemas de control que han de manipular a un robot así como los modelos matemáticos que proporcionen una base sólida de conocimiento que ayuden a describir el funcionamiento, operación y control de ellos, la descripción del funcionamiento y control de los motores los cuales son elementos actuadores, en ocasiones se puede llevar rutinas programadas de accionamientos o estar controlados por un dispositivo de control manual ya sea que este directamente conectado o sea un mando a distancia. El control de motores es una parte vital en el funcionamiento de un robot y es en esta área de conocimiento donde profundizaremos en el desarrollo de esta tesis.

Los robots industriales actualmente son los más utilizados por la amplia variedad de aplicaciones que se le dan dentro de la industria. Dichos robots son llamados también brazos robots por analogía con el brazo humano. Son muchas las compañías que actualmente fabrican este tipo de robots, entre ellas se pueden mencionar ABB, FANUC, Honda, Bosh, MOTOMAN, etc.; usando cualquier navegador y haciendo una solicitud de búsqueda de fabricantes de robots industriales se pueden hallar más de trescientas empresas alrededor del mundo que fabrican este tipo de automatismos.

Los brazos robots se usan dentro de la industria para automatización de procesos, como son transporte de partes de un sistema, soldadura, pintura, selección y ordenamiento de piezas, para ensamble de vehículos, en medicina para cirugías, etc. [18]

Por su interés práctico y uso más generalizado se estudiara en detalle el brazo robot, el cual en forma general está formado por:

- Articulaciones
- Actuadores
- Sensores

- Interface de potencia
- Interface de control
- Software para el control

## 1.2. Justificación de la Tesis

En la actualidad el uso y aplicación de los robots ya no solo está restringido a la industria, hay aplicaciones a la medicina, al hogar, a la educación desde temprana edad, en aplicaciones de investigación científica; hoy día los robots no solo se usan con un determinado programa y realizan una sola función, sino que se están utilizando en sistemas de inteligencia artificial, en redes industriales, y el grado de automatización que se ha alcanzado en algunas aplicaciones hace que la intervención humana este relegada solo a actividades de supervisión, monitoreo y mantenimiento de éstos equipos, y en ocasiones esto se hace vía remota. [4]

El estar a la vanguardia en el uso de tecnologías nos permite tener una ventaja competitiva en el mercado laboral el cual es cada vez más competido y exigente, se ha demostrado que el uso de robots incrementa la productividad reduciendo tiempos de ensamblado, costos por mano de obra, se mejora la calidad, ya que es un programa el que realiza la tarea y no un trabajador que se agota a lo largo de una jornada de trabajo. El uso y conocimiento de tecnologías permite no solo al individuo que los posee avanzar, sino que ayuda a su entorno al poder hacer uso y traer tecnologías que mejoren el nivel de vida de las personas. Es importante para nuestra región que se tengan conocimientos profundos en el uso de los robots, y de las disciplinas que son necesarias para su entendimiento (llámense matemáticas, mecánica, electrónica de potencia, software, microcontroladores, interfaces, acondicionadores de señal, etc.), el uso robots se esta generalizando ya que éstos han reducido sus costos y el avance de la tecnología también los ha hecho mejores, más rápidos, más fiables y más robustos. [4]

## 1.3. Planteamiento del problema

El rápido desarrollo de las tecnologías hace necesario que recursos didácticos sean implementados en la enseñanza superior (llámense universidades, tecnológicos o centros de investigación), por ello resulta indispensable el funcionamiento apropiado de todos los recursos con que cuente cada institución, el poseer elementos de enseñanza permite a los centros educativos poder transmitir el conocimiento y si este es transmitido por medio de distintos recursos didácticos que incluyan tanto la teoría como la práctica, el aprovechamiento en el alumno se fortalece. El área de robótica y en esencia el de los brazos industriales está comprobado que tienen una rápida expansión por ello es tan indispensable su comprensión y el dominio de las técnicas matemáticas y físicas que describen su funcionamiento, así como el conocimiento de las tecnologías necesarias para su implementación. Actualmente el laboratorio cuenta con una estructura mecánica de un brazo robot y los motores son de cd, se necesita hacerlo operacional, así que el primer obtaculo es hacer el control de motores de cd, esto es requerimos

del desarrollo de la etapa de potencia en una primera instancia, para después pasar al modelado y desarrollar la etapa de control mediante un software.

## 1.4. Objetivos General y Particulares

### 1.4.1. Objetivo General

En gran medida la expansión del uso e implementación de brazos robóticos en la industria se debe a que el costo de estos equipos ha disminuido con el avance de la tecnología, y al hecho de que en la industria han sido probados durante más de dos décadas con excelentes resultados. El uso eficiente de los recursos el aprovechamiento y reutilización de los equipos de laboratorio permite tener mayores recursos didácticos al alcance de todos. Es por eso que el objetivo de este proyecto es desarrollar un controlador que permita hacer funcional la estructura mecánica del brazo robot que se encuentra en el laboratorio de electrónica, usando para ello un dispositivo FPGA de bajo costo.

### 1.4.2. Objetivos Particulares

El objetivo primordial de este proyecto se basa en la construcción de la etapa de potencia de un brazo robot que será utilizado para fines educativos, para el modelado y construcción del sistema de potencia se utilizara programación VHDL en dispositivos FPGA. Para ellos se pretende:

- Modelar los motores de CD de la estructura.
- Implementar un PID para que el movimiento de las articulaciones sea suave.
- Utilizar las herramientas matemáticas para la localización espacial.
- Describir la cinemática del robot.
- Obtener la tabla de parámetros del brazo robot utilizando la convención Denavit-Hartenberg.
- Realizar la implementación del controlador en el FPGA usando programación VHDL.

## 1.5. Metas y alcances

Implementar la etapa de potencia, establecer el control de motores con PWM mediante el FPGA y verificar que el movimiento sea suave, realizar el modelado matemático para la obtención del modelo cinemático directo, y poder hacer manipulaciones del robot que le permitan moverse a puntos específicos dentro de su espacio de trabajo.

## Capítulo 2

# Marco Teórico

### 2.1. Actuadores

La función de un actuador convertir algún tipo de energía en energía mecánica. Los actuadores se clasifican según el tipo de energía con la que se alimentan: actuadores neumáticos, eléctricos e hidráulicos; y por el tipo de movimiento que generan, si este es un desplazamiento longitudinal o rotatorio. El tipo de actuador que nos interesa investigar es el actuador eléctrico ya que sus características de control sencillo, velocidad de respuesta y amplia variedad los hace los más utilizados en la industria y estos se clasifican según el tipo de energía eléctrica con la que se alimentan, por el movimiento que generan y por su forma de excitación, comúnmente les llamamos motores eléctricos.[19],[2]

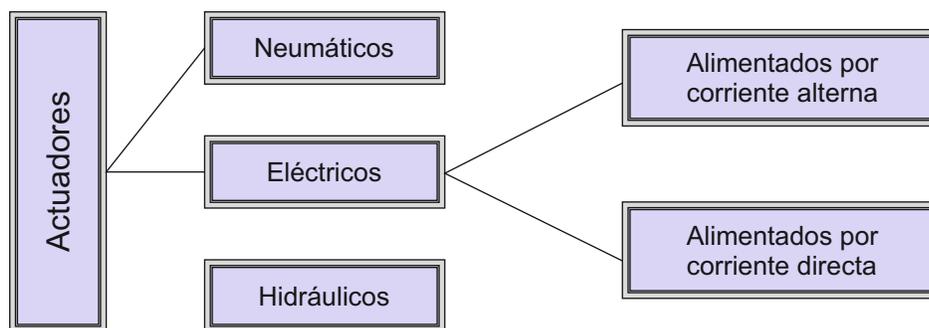


FIGURA 2.1: Clasificación de los actuadores

Los motores de corriente directa, que son los que se utilizan en esta investigación constan de un rotor y un estator. El rotor es la parte móvil y es el que empuja a la carga para generar el movimiento, y el estator es la parte fija que proporciona el empuje magnético que induce la fuerza electromotriz. [17]

En general los motores eléctricos tienen velocidades de rotación muy elevadas hasta 30000 rpm, en la mayoría de las aplicaciones de los brazos robots no se necesita de tal velocidad bastaría con velocidades de 120 rpm pero que tengan la fuerza necesaria para mover cargas pesadas que es algo que no posee un motor tradicional. Para adaptar estas características de los motores a la robótica lo que se hace es acoplar al rotor un sistema de engranes que permita disminuir la velocidad de giro pero que aumente el par de salida o la capacidad de carga que tiene el motor, lo que se hace es sacrificar velocidad y aumentar la fuerza en el par. [9]

Los motores de corriente directa se pueden controlar de diferentes formas: podemos variar la corriente en el campo, por demanda de par manteniendo fija la corriente de campo y variando la corriente de armadura, y por último, controlando el voltaje de armadura. La velocidad del motor sin carga que se obtiene aplicando el voltaje especificado para la armadura, la corriente especificada para la armadura y la corriente especificada para el campo y a la velocidad obtenida se le llama velocidad nominal o velocidad base.[16] [17]

Una clasificación de los motores de CD se puede apreciar en la figura 2.2.

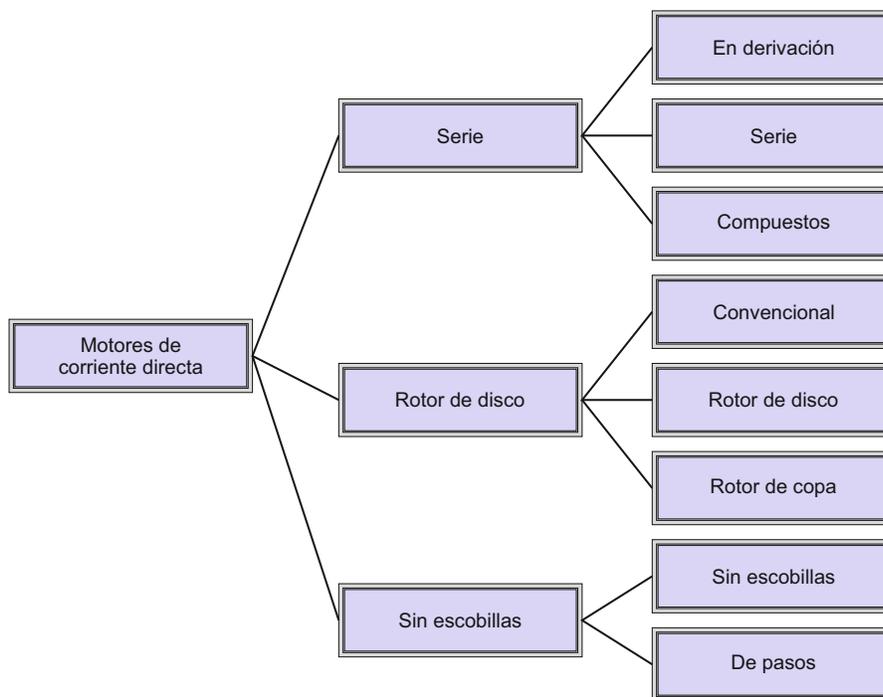


FIGURA 2.2: Clasificación de los motores de CD

El circuito equivalente de un motor serie de corriente directa se muestra en la figura 2.3, y puede apreciarse que el devanado de campo queda en serie con el devanado de la armadura, el hecho de que quede en serie hace que el devanado de un motor serie se utilice un alambre más grueso que el que se utiliza en el motor en derivación, ya que conduce la corriente de la armadura y no solo su propia corriente de campo, como la corriente es más grande, genera un campo más grande, esto hace que el devanado de campo tenga menos vueltas para lograr esto.[10]

Los motores de corriente directa pueden trabajar en cuatro modos de operación dependiendo del voltaje aplicado ya sea negativo o positivo y del sentido en que fluya la corriente, estos modos son motorización, frenado regenerativo, frenado dinámico o de recuperación, frenado con reversa.[16] En resumen un motor de corriente directa debe ser capaz de operar en los cuatro cuadrantes y para las operaciones en reversa debe de invertir la excitación del campo, para así poder invertir la fem inducida.[16],[17]

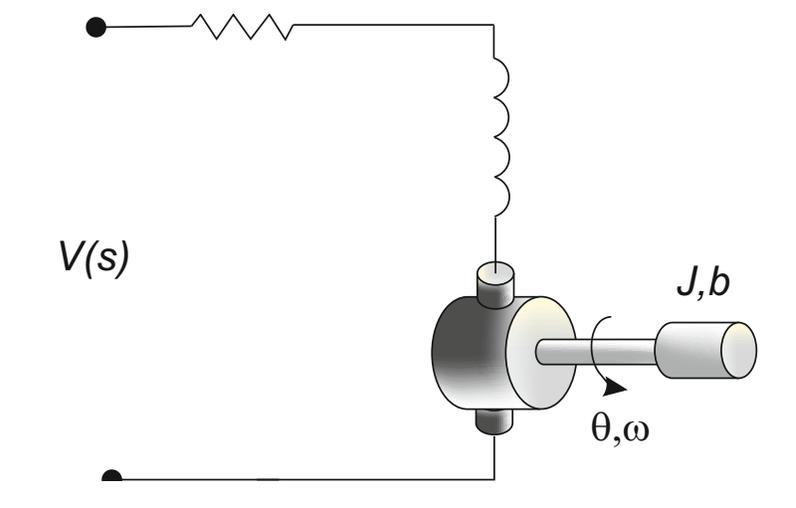


FIGURA 2.3: Modelo de motor de CD en Serie

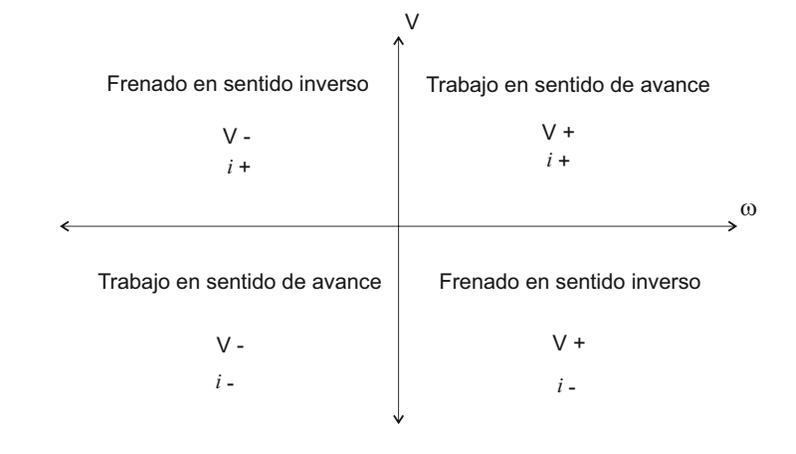


FIGURA 2.4: Cuadrantes de operación de un motor de CD

## 2.2. Sensores

Para dar una definición correcta de lo que es un sensor, primero hay que recordar que es un transductor. La función de un transductor es la de convertir una variable física en otra que tiene un dominio diferente, entonces un transductor puede ser un sensor pero también un actuador. Para realizar esta función un transductor se vale de algún principio físico de transformación de energía, y estos son: piezoresistivo, capacitivo, ultrasónico, magnético, térmico, fotoeléctrico, electroquímico, resistivo. Si clasificamos a los sensores por el principio de transducción se encuentran: piezoresistivo, capacitivo, piezoeléctrico, ultrasónico, magnético, termoelectrónico, fotoeléctrico, químico. En la figura 2.5 podemos ver la clasificación de los sensores por el tipo de variable medida.[19]

Es importante analizar las características de los sensores y estas son estáticas y dinámicas, las primeras no cambian con el tiempo y las segundas están en función de este. Dentro de las características estáticas de un sensor están la:

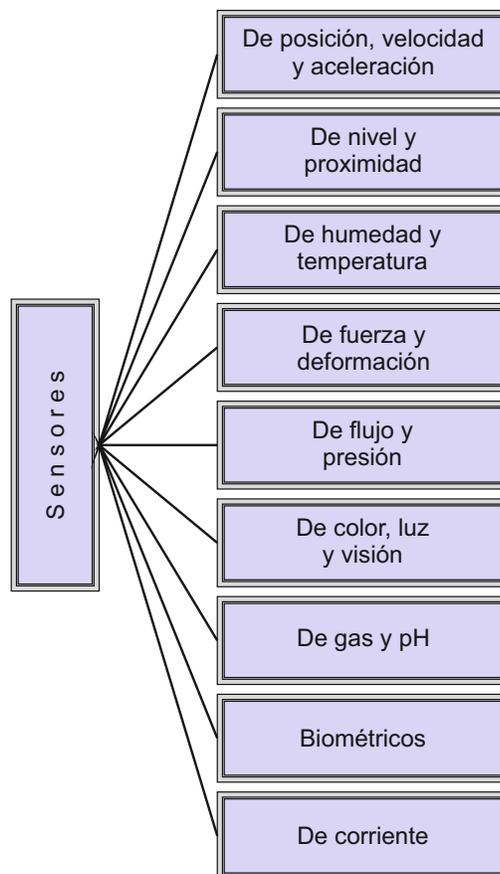


FIGURA 2.5: Clasificación de los sensores

Sensibilidad, que se define como la entrada mínima que requiere el sensor para dar una salida. El Rango, es el intervalo entre el valor mínimo y el valor máximo de la variable que se está midiendo. La Precisión, se refiere a cuando el sensor lee el mismo valor de entrada, este siempre dé el mismo valor a la salida. La Exactitud, es la diferencia entre la lectura que da el sensor y el valor real de la variable medida. Linealidad estática, esta depende de factores ambientales, que tanto se ve afectado el sensor por la contaminación ambiental a su alrededor, como por ejemplo ruido, vibraciones, temperatura, humedad, etc. El Offset, es un corrimiento en el valor que da salida y siempre es el mismo para cierto rango de operación, es como agregar una componente de  $V_{dc}$  en la salida. La Resolución, es la capacidad que tiene el sensor de registrar el cambio más pequeño en la variable de salida. El Error Estático, es error que existe al hacer una medición, siempre existirán errores en las lecturas, los sensores arrojan una cantidad y esa normalmente se ajusta a una tabla de valores donde se realiza alguna interpolación. Las características dinámicas son: El Tiempo de Respuesta, es el tiempo que transcurre desde que se presenta un cambio en la variable que se mide y el tiempo que tarda el sensor en registrar dicho cambio. Histéresis, es la capacidad que tiene el sensor para seguir la curva de salida ideal, cuando se tiene un sensor se linealiza la salida aunque en la práctica todos presentan no linealidades. La

Linealidad Dinámica, es la capacidad que tiene el sensor de seguir fielmente la curva de salida que viene en la hoja de especificaciones ante los cambios en la variable física de entrada, cuando existen cambios bruscos en la variable censada, la salida experimenta una no linealidad dinámica. Error Dinámico, normalmente se deben a errores en las mediciones por cargas inducidas debido a los mismos aparatos de medición.[19]

Actualmente la tendencia es desarrollar sensores cada vez más pequeños, con mejoras en todas sus características estáticas y dinámicas, este tipo de sensores llamados sensores inteligentes hacen uso de la tecnología para integrar dentro del sensor salidas lineales sin aumentar el número de conexiones que se requieren para poner el sensor dentro de una placa física, los sensores inteligentes fueron propuestos alrededor de 1980 y están definidos por la IEEE en el estándar 1452.1. [19]

Debido a que en este trabajo se pretende implementar un controlador para la estructura mecánica existente de un brazo robot Scorbot V, este está equipado solo con sensores de final de carrera, encoders y ópticos solo se pondrá una descripción de ellos ya que existe una amplia variedad en el mercado.

### 2.2.1. Fotodiodos

Los sensores de temperatura y los fotodiodos están basados en uniones p-n, en un fotodiodo en el cual incide una radiación óptica con longitud de onda dentro de la banda aceptada por el diodo en esta se genera una corriente que se suma a la generada por la unión p-n, que para todos los diodos depende de la temperatura existente en la unión la cual es no lineal con respecto al voltaje de la propia unión. Entonces cuanto mayor radiación luminosa que está dentro del espectro de frecuencias aceptados por el diodo mayor será la corriente inversa del diodo, lo que sucede dentro del fotodiodo es que cuando el rayo luminoso incide produce electrones libres y huecos, mientras más luz mayor es el número de portadores minoritarios y mayor será la corriente inversa del diodo. En la figura 2.6 se muestra un circuito típico para la polarización de un fotodiodo. [5],[12]

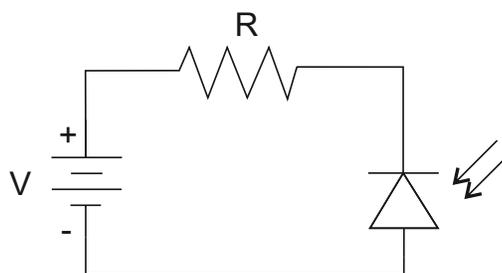


FIGURA 2.6: Circuito básico con fotodiodo

### 2.2.2. Fototransistores

Los fototransistores tienen más sensibilidad a la luz que los fotodiodos, como la conexión de la base está abierta, toda la corriente inversa entra a la base del transistor, lo que sucede dentro del fototransistor es que la corriente de colector es mayor a la corriente inversa multiplicada por un factor  $\beta$ . La unión p-n fotosensible es la de colector a base, entonces, la diferencia entre un fotodiodo y un fototransistor es el factor o ganancia de corriente  $\beta$ , y es que la misma cantidad de energía electromagnética produce en el transistor  $\beta$  veces más corriente en un fototransistor, la desventaja es que la velocidad de respuesta del fototransistor mientras que el fotodiodo tiene corrientes de salida de micro amperes y conmutación del orden de nanosegundos en el fototransistor se tienen corrientes en el orden de mili amperes y tiempos de conmutación de microsegundos.[21]

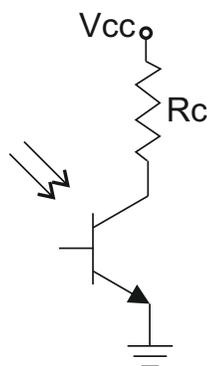


FIGURA 2.7: Fototransistor

### 2.2.3. Encoders

Si se desea conocer la posición angular de un motor de CD normalmente se utilizan los encoders, también se les llama codificadores ópticos, hay dos tipos: Encoders incrementales y encoders absolutos, los encoders incrementales utilizan dos emisores y dos receptores y los encoders absolutos utilizan el código gray. Un encoder óptico es un arreglo de dos diodos LED y dos fotoreceptores, por entre los cuales se hace pasar un disco con ranuras que al girar el motor genera un tren de pulsos y permite detectar el sentido de giro, como se muestra en la figura ?? [1] [18]

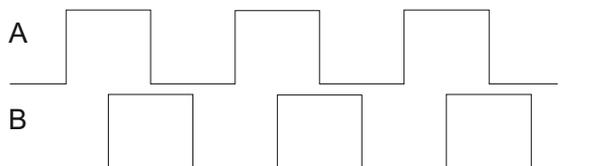


FIGURA 2.8: Señales de salida de un encoder incremental

Los encoders incrementales también llamados relativos como el de la figura 2.9 constan de un disco con una serie de huecos o marcas para presentaciones de mayor resolución, que están colocadas radialmente y son equidistantes entre sí, tienen un sistema que es un emisor de luz y un elemento fotoreceptor, el eje va acoplado al disco para poder medir su posición, con este acomodo al girar el eje se generan pulsos en el receptor cada vez que un hueco o marca permita el paso de luz, si llevamos la cuenta de éstos pulsos se puede obtener la posición del eje. La resolución de este sensor depende directamente del número de marcas que se ponen en el disco, se han desarrollado encoders que pueden contar hasta 100,000 pulsos por vuelta. Con este tipo de encoders no se puede saber la posición absoluta, esta se puede conocer mediante un circuito que cuente el número de pulsos, para entonces determinar la posición angular del motor.[1] [18] [? ]

La resolución de estos encoders ópticos se calcula con la fórmula:

$$R = \frac{\pi D}{2a} \quad (2.1)$$

Donde:

$R$  = Resolución

$D$  = Diámetro

$a$  = Ancho de ranura

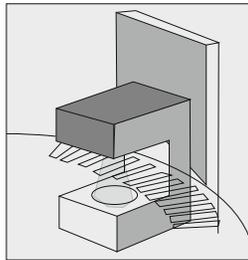


FIGURA 2.9: Encoder óptico

Los encoders absolutos son parecidos a los encoders incrementales ya que tienen emisores de luz, fotoreceptores, un disco con graduaciones o marcas, en este caso el disco no es una sucesión de marcas iguales y equidistantes, son conjunto de secciones codificadas en código binario, las resoluciones van desde  $2^8$  a  $2^{19}$  bits. Para este tipo de encoders no es necesario un circuito contador que lleve la cuenta del número de pulsos generados por los fotoreceptores ya que cada posición esta codificada en forma absoluta. [2],[1]

En la figura 2.10 se muestra un disco de un encoder absoluto con una resolución  $2^4$ .

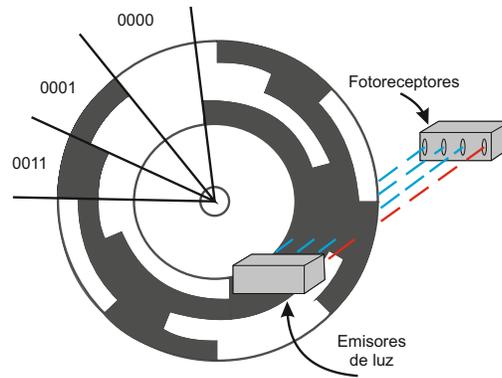


FIGURA 2.10: Disco de un Encoder absoluto

### 2.3. Convertidores de potencia

Un sistema de potencia está conformado por una fuente, una carga, un interruptor y el sistema de control. El sistema de potencia ideal será aquel en el que la potencia suministrada a la entrada sea igual a la potencia entregada. [16]

$$P_{in} = P_{out} \quad (2.2)$$

Para que esto sea cierto debe de existir un interruptor ideal, el cual será un dispositivo con dos estados de operación, uno en circuito abierto y el otro en corto circuito, esto es; un dispositivo de conmutación. [16],[17]

Por lo tanto, la interfaz de potencia es aquella en la que se estudian los sistemas de conversión de la energía eléctrica, es decir, los llamados convertidores estáticos. Existen dos clasificaciones para los convertidores estáticos, según su función y según el cuadrante de operación. [16],[17]

Los tipos de convertidores según su funcionamiento son:

- Convertidor de cc-cc
- Convertidor de cc-ca
- Convertidor de ca-ca
- Convertidor de ca-cd

Los tipos de convertidores según el cuadrante de operación son:

- $V_{out}$  es unipolar y la  $I_{out}$  es unidireccional.
- $V_{out}$  es unipolar y la  $I_{out}$  es bidireccional.
- $V_{out}$  es bipolar y la  $I_{out}$  es unidireccional.
- $V_{out}$  es bipolar y la  $I_{out}$  es bidireccional.

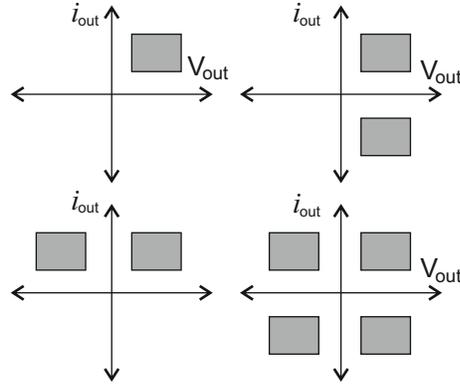


FIGURA 2.11: Convertidor según el cuadrante de operación

### 2.3.1. Inversores

En esta sección se describe el funcionamiento de los convertidores CD-CA llamados inversores, específicamente se llega al circuito inversor de puente completo monofásico con modulación de ancho de pulso. Este tipo de convertidor es capaz de generar un voltaje de CA simétrico a partir de una fuente de entrada de CD, donde el voltaje de salida y la frecuencia pueden ser variables. Para variar este voltaje a la salida, se puede variar el voltaje a la entrada que normalmente es fijo o se puede implementar alguna técnica de modulación por ancho de pulso (PWM). En el primer método la ganancia del inversor es unitaria y en el segundo la ganancia del inversor es variable estableciendo un control PWM, así la relación de la ganancia en el inversor es una relación entre el voltaje de CD a la entrada y el voltaje CA de la salida.[16],[17]

Idealmente la forma de onda de un inversor a la salida debería ser sinusoidal, en la práctica esto no sucede así y son salidas que además contienen armónicas que deberán filtrarse, para aplicaciones de baja y mediana potencia pueden aceptarse formas no sinusoidales, pero para altas potencias si son necesarias y las distorsiones deben ser mínimas.[16],[17]

Existen inversores monofásicos y trifásicos y son controlados mediante señales de control PWM para generar el voltaje de salida CA, los inversores son ampliamente utilizados en la industria para control de motores de CA, fuentes ininterrumpibles de energía, hornos de inducción, etc. [16],[17]

### 2.3.2. Funcionamiento básico

Para explicar el funcionamiento de un inversor se usa el circuito de la figura 2.12, en la cual podemos ver el circuito básico de potencia y la forma del voltaje de salida en la carga cuando esta se trate de una carga puramente resistiva.[16],[17]

El valor del voltaje RMS (Root Mean Square-Raíz media cuadrática) de la forma de onda de un inversor de medio puente se obtiene con 2.3 [17]

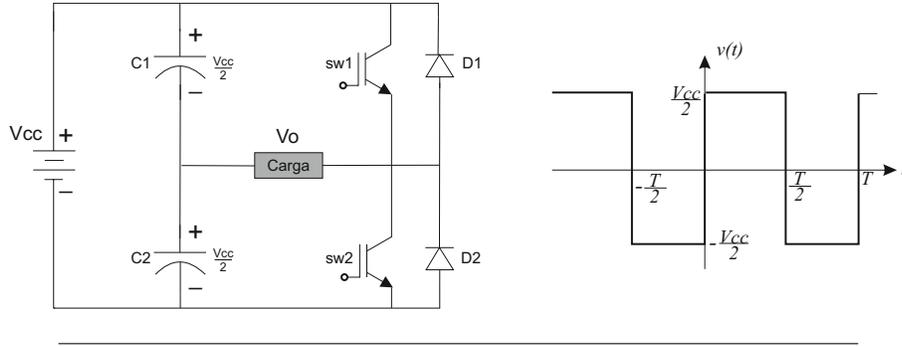


FIGURA 2.12: Circuito inversor de medio puente y forma de onda de voltaje de salida

$$V_0 = \sqrt{\frac{2}{T} \int_0^{\frac{T}{2}} \frac{V_{cc}^2}{4} dt} = \frac{V_{cc}}{2} \quad (2.3)$$

El voltaje de salida para cualquier instante  $t$  se calcula analizando su forma de onda mediante series Fourier obtenemos 2.4 [17]

$$V_0 = \sum_{n=1,3,5,\dots}^{\infty} \frac{2V_{cc}}{n\pi} \text{sen}(n\omega t) \quad (2.4)$$

Donde  $\omega = 2\pi f$ , y  $f$  es la frecuencia de  $V_0$  en radianes por segundo. Del análisis de Fourier resulta que la función es impar, de tal modo para valores de  $n = 2, 4, 6 \dots$  el  $V_0 = 0$  lo que indica que las armónicas pares no existen y los valores de  $a_0$  y  $a_n$  son cero por la simetría de la forma de onda de la salida. [17]

Cuando la carga es resistiva la corriente en la salida está en fase con el voltaje, pero cuando la carga es altamente inductiva la forma de onda de la corriente es como en la figura 2.13. [17]

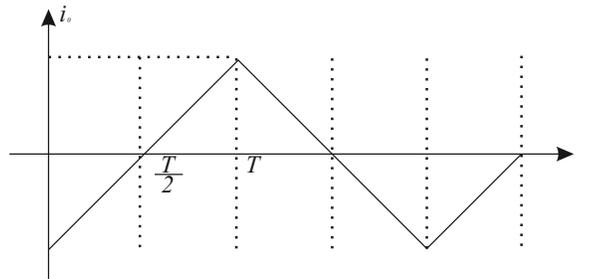


FIGURA 2.13: Corriente en de carga altamente inductiva

Un inversor de puente completo monofásico se muestra en la figura 2.14, a diferencia del inversor de medio puente este tiene cuatro interruptores cuando  $sw1$  y  $sw2$  se activan el voltaje de alimentación  $V_{cc}$  es igual a  $V_o$ , luego al apagar  $sw1$  y  $sw2$  y al “mismo tiempo” encender  $sw3$  y  $sw4$  en este caso  $V_{cc} = -V_o$ , el tiempo que transcurre desde que se apagan  $sw1$  y  $sw2$  y se encienden  $sw3$  y  $sw4$  debe ser igual o mayor que el tiempo de apagado de los interruptores. [17]

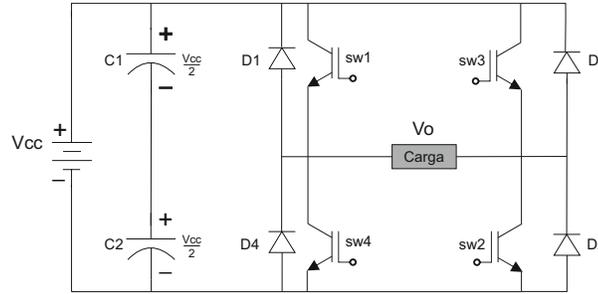


FIGURA 2.14: Circuito inversor de puente completo

El voltaje RMS de salida se obtiene de con la ecuación 2.5. [17]

$$V_0 = \sqrt{\frac{2}{T} \int_0^{\frac{T}{2}} V_{cc}^2 dt} = V_{cc} \quad (2.5)$$

De la serie de Fourier obtenemos el voltaje instantáneo de salida 2.6. [17]

$$V_0 = \sum_{n=1,3,5,\dots}^{\infty} \frac{4V_{cc}}{n\pi} \text{sen}(n\omega t) \quad (2.6)$$

Para  $n = 1$ , la ecuación anterior representa la componente fundamental 2.7

$$V_1 = \frac{4V_{cc}}{\sqrt{2}\pi} = 0,90V_{cc} \quad (2.7)$$

La corriente de salida instantánea para una carga RL será 2.8. [17]

$$i_0 = \sum_{n=1,3,5,\dots}^{\infty} \frac{4V_{cc}}{n\pi \sqrt{R^2 + (n\omega L)^2}} \text{sen}(n\omega t - \theta_n) \quad (2.8)$$

donde  $\theta_n = \tan^{-1} \frac{n\omega L}{R}$ .

Al comparar los inversores de medio puente y puente completo se observa que el voltaje de bloqueo inverso de cada interruptor y la calidad del voltaje de salida son iguales, pero en los inversores completos la potencia de salida es cuatro veces mayor y componente fundamental de la salida es del doble en el inversor de puente completo que en el inversor de medio puente. [17]

### 2.3.3. Inversores monofásicos controlados por voltaje

Existen diferentes técnicas para controlar el voltaje a la salida de un inversor, que pueden ser variar el voltaje de alimentación, regular el voltaje del inversor y mantener voltaje y frecuencia constantes. El método que existe más eficiente para controlar la ganancia es la modulación por ancho de pulso (PWM) en los inversores. Las técnicas que se utilizan para lograr esto son: [16],[17]

- Modulación por ancho de un solo pulso
- Modulación por ancho de múltiples pulsos

- Modulación por ancho de pulso sinusoidal
- Modulación por ancho de pulso sinusoidal modificado
- Control por desplazamiento de fase

### Modulación por ancho de un solo pulso

Este tipo de control se ejerce mediante un único pulso en cada medio ciclo haciendo variar su ancho para de esta manera controlar el voltaje de la salida del inversor. Las señales de disparo se generan comparando una señal rectangular de referencia  $V_r$  con un onda portadora triangular  $V_t$ , la frecuencia de la señal de referencia es la que determina la frecuencia fundamental del voltaje de salida. [17]

## 2.4. Proceso de diseño de inversor

Para determinar las capacidades de voltaje y corriente de los dispositivos de potencia que se utilizan en el circuito inversor depende directamente de la clase de inversor, de la carga y del sistema de control que se implemente. Para el diseño se requiere conocer las ecuaciones de la corriente instantánea de la carga y graficar las formas de onda de corriente para cada dispositivo, una vez que se conoce lo anterior se pueden seleccionar los dispositivos de potencia. Para determinar las capacidades de voltaje primero se necesita establecer los voltajes inversos de cada uno de los dispositivos. [17]

Para reducir las componentes no deseadas en la señal de salidas se colocan filtros a la salida que pueden ser un filtro C, uno LC sintonizado o en el mejor de los casos un filtro CLC que resulta más efectivo, como los que se muestran en la figura 2.15. [17]

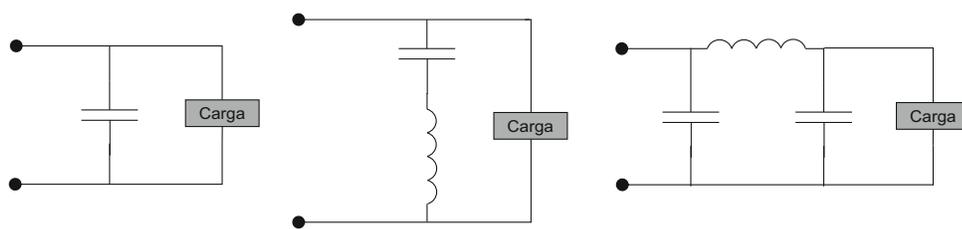


FIGURA 2.15: Filtros típicos de salida

## 2.5. Robótica

Cuando escuchamos la palabra robot lo primero que se nos viene a la mente es la imagen de un sistema humanoide, cuando escuchamos la palabra robótica lo que vemos es una estructura que semeja a un brazo humano, esto es debido a la influencia de los medios de comunicación y lo que ellos nos proyectan. En

los inicios de la robótica se pensaba que éstos sistemas superarían las labores humanas y que seríamos reemplazados rápidamente, la realidad es que se necesita un nivel de avance tecnológico que no se tiene y no se tendrá en mucho tiempo, así que los robots están relegados en la actualidad a tareas repetitivas y que pueden ser programadas por los seres humanos, lo anterior no significa que dichas tareas sean sencillas.

Son muchas las disciplinas que intervienen en el desarrollo de la robótica y mientras más complejo es un sistema se hacen necesarias más áreas del conocimiento, una definición formal de la palabra Robot tomada del diccionario de la Real Academia Española es:

*Máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes reservadas solo a las personas.*

El robot que en la industria es el más utilizado es el llamado de cadena cinemática abierta cuyos elementos principales son los eslabones y las articulaciones. Los eslabones son los elementos que le dan forma a la estructura del robot, y las articulaciones son las uniones que existen entre dos eslabones, un robot de cadena cinemática abierta es aquel que al unir sus eslabones mediante articulaciones el último eslabón es aquel que no se conecta y dicho extremo queda libre. El movimiento de los eslabones está determinado por el tipo de articulación y estas pueden ser rotacionales ó traslacionales, llamadas también rotativas ó prismáticas respectivamente. [?] [15] [12]

Los eslabones se enumeran desde la base hasta el efector final utilizando la letra  $l$  y el subíndice  $i$  para enumerarlos  $l_i$ , y por convención se toma al eje  $z$  como el eje sobre el cual rota o se realiza el movimiento de traslación la articulación. Entonces, si el movimiento es rotacional se utiliza la variable  $\theta_i$  y si el movimiento es traslacional se utiliza la variable  $d_i$ . Una vez que ya se tienen determinados cuales son los eslabones y las articulaciones se puede determinar los grados de libertad (GDL) que tiene el robot. [12]

Los grados de libertad de un robot se refieren al conjunto de variables que describen el movimiento del robot y con las cuales se puede obtener la posición y orientación final del robot. Para que el robot se mueva necesita tener acoplado en sus articulaciones a los actuadores quienes son los encargados de proveer la fuerza necesaria para realizar los movimientos. Cuando se da el caso de que las articulaciones son independientes el número de dichas articulaciones coincide con el número de GDL del sistema. Cuando se da el caso de que una articulación tenga más de un tipo de movimiento con respecto a un eje determinado, dicha articulación tendrá más de un grado de libertad. Por ejemplo, una articulación esférica podría tener hasta 3-GDL. [?] [12]

Para que un brazo robot ubique su extremo final en un punto en el espacio son necesarios que se tengan al menos 3 GDL y si además se desea una determinada orientación se deberán tener otros 3 GDL sumando 6 GDL como mínimo para poderse ubicar en un punto en el espacio con una orientación determinada y que estos puntos estén dentro del espacio de trabajo alcanzable por el brazo robot. [12] [4]

Como se mencionó en el capítulo uno un brazo robot está formado por varios elementos, un ejemplo puede ser el sistema mostrado en la figura ??.

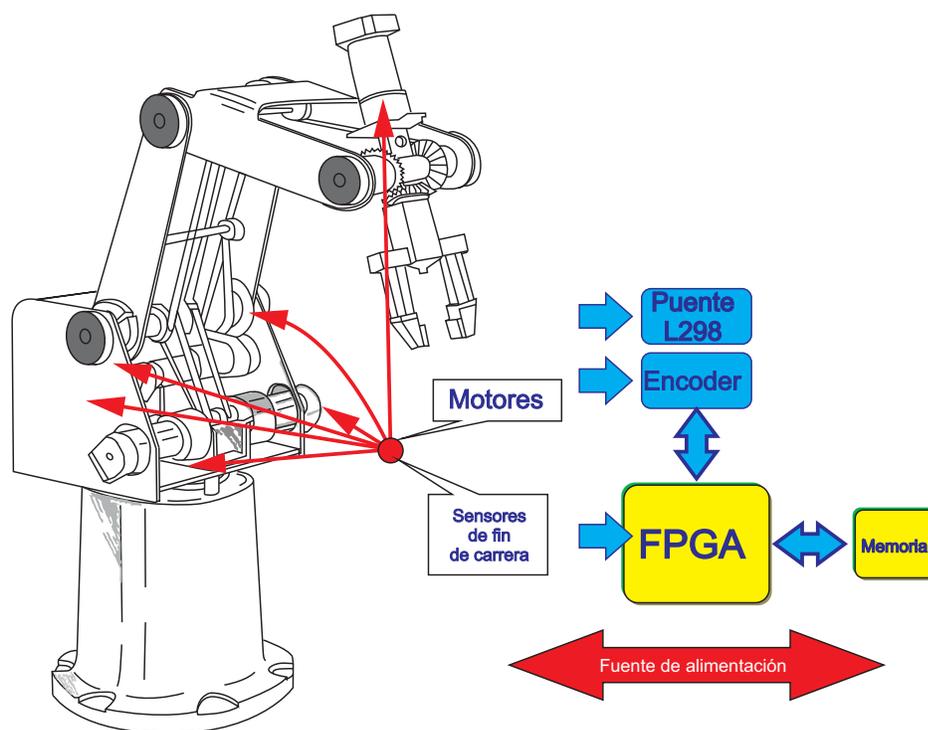


FIGURA 2.16: Elementos de un brazo robot

Este sistema contiene como elementos

- Un sistema mecánico.
- Un sistema de potencia.
- Sensores y actuadores.
- Un sistema de control.

## 2.6. Cinemática de Robots manipuladores

La cinemática es la parte de la física que estudia al movimiento sin tomar en cuenta las causas que lo originan, en ella se realiza la descripción analítica del movimiento, y establece las relaciones entre cada articulación con respecto a un sistema de referencia cartesiano, esto resulta de vital importancia ya que con ella se puede determinar la posición y orientación del extremo final del robot y define el espacio de trabajo del robot, que son todas aquellas posiciones que el robot es capaz de alcanzar.[2][10]

Un robot manipulador de cadena cinemática abierta como ya se había definido anteriormente está conformado por un conjunto de cuerpos rígidos, que son eslabones unidos por articulaciones que pueden ser rotacionales, prismáticas, cilíndricas, planares, de tornillo o esféricas. Para analizar este tipo de estructuras

es necesario definir que son los grados de libertad. Un grado de libertad es una coordenada independiente sobre la cual se describe la configuración del robot manipulador.

En robótica un modelo cinemático debe ser capaz de entender los movimientos de rotación y traslación de los eslabones, y éstos movimientos deben ser descritos en forma adecuada para que se establezca un orden, al establecer un conjunto de sistemas de coordenadas en cada una de las articulaciones y que queden establecidas las relaciones de éstos sistemas, ya que con ello se determinan las posiciones y orientaciones que se pueden alcanzar al quedar determinado el espacio de trabajo del robot.

Los brazos robóticos están formados por eslabones y a la unión de cada eslabón se le llama articulación, el número de articulaciones que posea el robot define el número de grados de libertad que tendrá el sistema, las articulaciones solo pueden realizar dos tipos de movimientos: movimiento lineal también llamado prismático o movimiento rotacional. [10]

Los grados de libertad (DOF por sus siglas en inglés: degree-freedom) representan cada uno una coordenada independiente de cada una de las posiciones que puede alcanzar esa articulación y permite definir la configuración del sistema. Para definir los grados de libertad que posee un sistema es necesario identificar aquellas articulaciones que proveen información que ayude a definir la configuración de un sistema. [3]

De acuerdo con la definición de un cuerpo rígido, cada elemento de un sistema debe tener un sistema coordenado que defina su posición y orientación, con respecto a un sistema coordenado base. Si se cumple con la definición de cuerpo rígido es posible aplicar la segunda ley de Newton que nos dice que la fuerza sobre una partícula se calcula a partir de la masa y la derivada de la velocidad y los postulados de Euler que permiten modelar el movimiento rotacional de un cuerpo considerando sus propiedades inerciales. [Fundamentos de robotica]

Cuando se trata de ubicar el extremo final del brazo robot se necesita solo un punto en el espacio y este queda definido por sus componentes en  $x$ ,  $y$ ,  $z$  del espacio cartesiano, además en un brazo robot también debemos definir la orientación con respecto al sistema de referencia que estemos usando. Para definir la orientación del extremo final del robot también se requieren de tres componentes linealmente independientes y se define un nuevo sistema de referencia que esta relacionado con el anterior sistema de referencia que define el punto final del brazo. [2]

## Matrices de rotación

Es necesario ahora dar una definición de sistema de referencia, se debe entender que un sistema de referencia es un sistema de coordenadas  $x$ ,  $y$ ,  $z$  el cual tiene un origen  $o$ . Cuando expresamos el punto final de un robot lo que hacemos es definir un vector y cuando hacemos operaciones con vectores se requiere que hagan referencia al mismo sistema coordenado o si los vectores pertenecen a sistemas coordenados distintos éstos deberán ser paralelos. Así que si se realizan operaciones con vectores que están en distintos sistemas de referencia pero

que dichos sistemas utilizan un sistema de referencia común es posible expresar sus las operaciones matriciales utilizando matrices de rotación y matrices de transformación. [2]

Las matrices de rotación son el sistema más utilizado para describir la orientación de un brazo robot, ya que hacen uso del álgebra matricial básico. Supongamos que tenemos dos sistemas de referencia OXY y OUV los cuales comparten el mismo origen, si utilizamos el sistema OXY como fijo y el sistema OUV como móvil, entonces los vectores unitarios del sistema fijo son  $i_x, j_y$  y los vectores unitarios del sistema móvil son  $i_u, i_v$ . Entonces el vector en el plano del sistema móvil se puede expresar como:[2]

$$P = p_u i_u + p_v j_u \quad (2.9)$$

$$\begin{cases} p_x = i_x P \\ p_y = j_y P \end{cases} \quad (2.10)$$

Sustituyendo 2.9 en 2.10 se obtiene:[2]

$$\begin{bmatrix} p_x \\ p_y \end{bmatrix} = R \begin{bmatrix} p_u \\ p_v \end{bmatrix} \quad (2.11)$$

donde:

$$R = \begin{bmatrix} i_x i_u & i_x j_u \\ j_y i_u & j_y j_u \end{bmatrix} \quad (2.12)$$

A la ecuación 2.12 se le conoce como matriz de rotación, y de esta manera queda definida la orientación del sistema OUV con respecto al sistema OXY, y se trata de una matriz ortonormal, de tal forma que  $R^{-1} = R^T$ . [2]

Si consideramos el sistema OUV girado un ángulo  $\alpha$  con respecto al sistema OXY, la matriz de rotación  $R$  queda como sigue: [2]

$$R = \begin{bmatrix} \cos \alpha & -\text{sen } \alpha \\ \text{sen } \alpha & \cos \alpha \end{bmatrix} \quad (2.13)$$

Cuando los sistemas coordenados están en el espacio el sistema OXYZ es  $i_x, j_y, k_z$ , y el sistema OUVW es  $i_u, j_v, k_w$ . Por lo tanto, un vector definido en este espacio tridimensional se podrá definir para ambos sistemas de referencia como: [2]

$$\begin{aligned} P_{xyz} &= [p_x, p_y, p_z]^T = p_x \cdot i_x + p_y \cdot j_y + p_z \cdot k_z \\ P_{uvw} &= [p_u, p_v, p_w]^T = p_u \cdot i_u + p_v \cdot j_v + p_w \cdot k_w \end{aligned} \quad (2.14)$$

Y como se hizo en el caso de dos dimensiones, ahora tenemos: [2]

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = R \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} \quad (2.15)$$

Entonces la matriz R para el espacio tridimensional es: [2]

$$R = \begin{bmatrix} i_x i_u & i_x j_v & i_x k_w \\ j_y i_u & j_y j_v & j_y k_w \\ k_z i_u & k_z j_v & k_z k_w \end{bmatrix} \quad (2.16)$$

Es es la matriz de rotación donde se define la orientación del sistema  $OUVW$  con respecto al sistema de referencia  $OXYZ$ . Y como se vió en el espacio bidimensional, también se cumple que  $R^{-1} = R^T$ . [2]

Si hacemos girar el sistema  $OUVW$  sobre su eje  $OU$  y que este siempre se matenga sobre el eje  $OX$  del sistema  $OXYZ$ , entonces la matriz de rotación con respecto a  $x$  es: [2]

$$Rot x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\text{sen } \alpha \\ 0 & \text{sen } \alpha & \cos \alpha \end{bmatrix} \quad (2.17)$$

De nuevo si giramos el sistema  $OUVW$  sobre el eje  $OV$  y este se matenga sobre el eje  $OY$ , nos da la matrix de rotación con respecto a  $y$ ,

$$Rot y(\phi) = \begin{bmatrix} \cos \phi & 0 & \text{sen } \phi \\ 0 & 1 & 0 \\ -\text{sen } \phi & 0 & \cos \phi \end{bmatrix} \quad (2.18)$$

Por último rotamos el sistema  $OUVW$  sobre su eje  $OW$  y que coincide con el eje  $OZ$ , la matriz de rotación que obtenemos es con respecto al eje  $z$ , [2]

$$Rot z(\theta) = \begin{bmatrix} \cos \theta & -\text{sen } \theta & 0 \\ \text{sen } \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.19)$$

Es posible poder utilizar estas tres matrices de rotación y expresarlas como una multiplicación de matrices que representan una suceción de rotaciones de tal forma que si tenemos una matriz que es rotada primero sobre el eje  $OW$ , posteriormente sobre el eje  $OV$  y por último sobre el eje  $OU$  que se pueda representar como: [2]

$$T = Rotz(\theta)Roty(\phi)Rotx(\alpha) \quad (2.20)$$

$$\begin{aligned} &= \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\phi & 0 & S\phi \\ 0 & 1 & 0 \\ -S\phi & 0 & C\phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\alpha & -S\alpha \\ 0 & S\alpha & C\alpha \end{bmatrix} \\ &= \begin{bmatrix} C\theta C\phi & -S\theta C\alpha + C\theta S\phi S\alpha & S\theta S\alpha + C\theta S\phi C\alpha \\ S\theta C\phi & C\theta C\alpha + S\theta S\phi S\alpha & -C\theta S\alpha + S\theta S\phi C\alpha \\ -S\phi & C\phi S\alpha & C\phi C\alpha \end{bmatrix} \end{aligned}$$

Aquí es muy importante el orden en que se realizan las rotaciones, ya que el producto de matrices no es conmutativo y estas quedan definidas por la regla de la mano derecha. [2],[12]

Esta matriz 2.22 también es conocida como representación Roll-Pitch-Yaw (RPY)

Ahora supongamos que se tiene un vector  $P^1 = [3, 3, 0]^T$  lo cual significa que el vector está en el sistema de referencia  $OUVW$  y deseamos conocer la posición de ese vector en el sistema de referencia  $OXYZ$  y que el ángulo  $\alpha = 45^\circ = \frac{\pi}{4}$ , entonces el vector  $P^0$  es: [2],[12]

$$P^0 = R_1^0 P^1 = \text{Rot } x(45^\circ) \begin{bmatrix} 3 \\ 3 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \sqrt{18} \\ 0 \end{bmatrix}$$

### Matrices de transformación homogénea

Conocemos las operaciones para representar rotaciones, pero también es necesario conocer aquellas operaciones en las que existen desplazamientos. De esta manera las matrices de transformación nos permiten conocer y determinar de manera conjunta la posición y orientación del efector final del brazo.

En un robot manipulador es necesario poder representar la posición y orientación del extremo final del robot y esto se hace en el sistema de coordenadas cartesianas. Así una matriz de transformación homogénea involucra tanto operaciones de traslación como de rotación. Para poder representar un vector de posición de  $n$  componentes por un vector de  $n + 1$  componentes y es como se forma la matriz de transformación homogénea. Y el vector  $P(x, y, z)$  queda representado por  $P(wx, wy, wz, w)$ , donde  $w$  es un escalar. Entonces un vector  $P = ai + bj + ck$ , donde  $i, j, k$  son vectores unitarios de los ejes  $OX, OY, OZ$  del sistema de referencia fijo  $OXYZ$ , sus coordenadas homogéneas son: [2],[12]

$$P = \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} aw \\ bw \\ cw \\ w \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix} \quad (2.21)$$

De esta definición de coordenada homogénea, podemos representar direcciones con la forma  $[a, b, c, 0]^T$ . Entonces para definir una matriz de transformación homogénea  $T$  queda como: [2],[12]

$$T = \begin{bmatrix} R_{3x3} & P_{3x1} \\ f_{1x3} & w_{1x1} \end{bmatrix} = \begin{bmatrix} \text{Rotación} & \text{Traslación} \\ \text{Perspectiva} & \text{Escalado} \end{bmatrix} \quad (2.22)$$

Si se supone nula la perspectiva y el escalado unitario, la matriz  $T$  es:

$$T = \begin{bmatrix} R_{3x3} & P_{3x1} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \text{Rotación} & \text{Traslación} \\ 0 & 1 \end{bmatrix} \quad (2.23)$$

Hay que considerar que el sistema  $O'UVW$  está trasladado por un vector  $P = p_x i + p_y j + p_z k$  respecto al sistema fijo  $OXYZ$ , que es la matriz básica de traslación: [2],[12]

$$T(p) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.24)$$

Entonces el vector  $r$  en el sistema  $O'UVW$ , sus componentes con respecto al sistema  $OXYZ$  son: [2],[12]

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_u \\ r_v \\ r_w \\ 1 \end{bmatrix} = \begin{bmatrix} r_u + p_x \\ r_v + p_y \\ r_w + p_z \\ 1 \end{bmatrix} \quad (2.25)$$

De la misma forma, un vector  $r$  en el sistema  $OXYZ$  desplazado según la traslación  $T$  tendrá como componentes  $r'$ : [2],[12]

$$\begin{bmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} r_x + p_x \\ r_y + p_y \\ r_z + p_z \\ 1 \end{bmatrix} \quad (2.26)$$

Las rotaciones de un vector alrededor de cada uno de los tres ejes  $OX, OY, OZ$  en un ángulo  $\alpha$ , se pueden realizar por las transformaciones de rotación, entonces la transformación de rotación alrededor del eje  $x$  es: [2],[12]

$$Rot\ x(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\text{sen } \alpha & 0 \\ 0 & \text{sen } \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.27)$$

$$Rot\ y(\phi) = \begin{bmatrix} \cos \phi & 0 & \text{sen } \phi & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen } \phi & 0 & \cos \phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.28)$$

$$Rot\ z(\theta) = \begin{bmatrix} \cos \theta & -\text{sen } \theta & 0 & 0 \\ \text{sen } \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.29)$$

Podemos calcular la representación del vector  $r$  cuyas coordenadas  $(r_x, r_y, r_z)$  que esta definido en el sistema de referencia  $OXYZ$ , realizando la multiplicación de la matriz  $T$  por las coordenadas  $(r_u, r_v, r_w)$  del sistema girado  $O'UVW$  de la siguiente forma: [2],[12]

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = T \begin{bmatrix} r_u \\ r_v \\ r_w \\ 1 \end{bmatrix} \quad (2.30)$$

Es igualmente usada para representar la rotación y la traslación de un vector respecto del sistema de referencia fijo  $OXYZ$  dado por: [2],[12]

$$\begin{bmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{bmatrix} = T \begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} \quad (2.31)$$

### Rotación seguida de traslación

Para representar la rotación de un ángulo  $\phi$  sobre el eje  $OX$  seguido de una traslación de vector  $P$  definido en el sistema  $OXYZ$  se tiene que: [2],[12]

$$T(x)Rot x(\phi) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & \cos \phi & -\text{sen } \phi & p_y \\ 0 & \text{sen } \phi & \cos \phi & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.32)$$

Para representar la rotación de un ángulo  $\theta$  sobre el eje  $OY$  seguido de una traslación de vector  $P$  definido en el sistema  $OXYZ$  se tiene que: [2],[12]

$$T(p)Rot y(\theta) = \begin{bmatrix} \cos \theta & 0 & \text{sen } \theta & p_x \\ 0 & 1 & 0 & p_y \\ -\text{sen } \theta & 0 & \cos \theta & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.33)$$

Para representar la rotación de un ángulo  $\psi$  sobre el eje  $OZ$  seguido de una traslación de vector  $P$  definido en el sistema  $OXYZ$  se tiene que: [2],[12]

$$T(p)Rot z(\psi) = \begin{bmatrix} \cos \psi & -\text{sen } \psi & 0 & p_x \\ \text{sen } \psi & \cos \psi & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.34)$$

### Traslación seguida de rotación

Si se desea realizar primero la traslación y posteriormente la rotación sobre cada uno de los ejes del sistema  $OXYZ$ , las matrices homogéneas que se obtienen son: [2],[12]

Primero la matriz de traslación del vector  $p$  y la rotación de un ángulo  $\phi$  sobre el eje  $OX$  es:

$$Rot x(\phi)T(p) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & \cos \phi & -\text{sen } \phi & p_y \cos \phi - p_z \text{sen } \phi \\ 0 & \text{sen } \phi & \cos \phi & p_y \text{sen } \phi + p_z \cos \phi \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.35)$$

Ahora la matriz de traslación de  $p$  y la rotación de un ángulo  $\theta$  sobre  $OY$  es:

$$Rot x(\theta)T(p) = \begin{bmatrix} \cos \theta & 0 & \sen \theta & p_x \cos \theta + p_z \sen \theta \\ 0 & 1 & 0 & p_y \\ -\sen \theta & 0 & \cos \theta & p_z \cos \theta - p_x \sen \theta \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.36)$$

Y por último la matriz de traslación de  $p$ .

$$Rot x(\psi)T(p) = \begin{bmatrix} \cos \psi & -\sen \psi & 0 & p_x \cos \psi - p_y \sen \psi \\ \sen \psi & \cos \psi & 0 & p_x \sen \psi + p_y \cos \theta \\ 0 & 0 & 0 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.37)$$

\*\*\*\*\*

## 2.7. Sistemas de control

Actualmente se utiliza la teoría de control clásico, teoría de control moderno y la teoría de control robusto. El control automático es hoy en día parte integral en sistemas industriales, en sistemas robóticos, en control de vehículos, etc. Existe existen dos tipos de sistemas de control, los de lazo cerrado y los de lazo abierto. [5]

*Sistemas de control de lazo cerrado.* Este tipo de sistemas de control son los que manejan retroalimentación, en este sistema el controlador toma como señal de entrada la diferencia entre la señal de entrada y la señal de retroalimentación, con el propósito de minimizar el error y mantener la salida dentro de un valor deseado. [5]

*Sistemas de control de lazo abierto.* En este tipo de sistemas no existe ningún tipo de retroalimentación y la señal de salida no afecta a la entrada, lo que sucede es que a cada señal de entrada le corresponde una salida prefijada y la precisión del sistema depende de su calibración, son sistemas que operan con una base de tiempo, al igual que lo ciclos de una lavadora.[13],[5]

Ventajas de un sistema de control de lazo abierto son:

- Un sistema de control en lazo abierto tiene menos elementos que un sistema en lazo cerrado por lo tanto, es más simple su construcción y mantenimiento.
- Su costo es menor.
- No hay problemas de estabilidad, ya que se conoce la salida y se evitan las perturbaciones
- Son convenientes cuando no se precisa medir la salida o cuando medir la salida implica un costo económico alto.

Las desventajas de un sistema de lazo abierto son:

- Las perturbaciones pueden afectar el sistema, y los cambios en la calibración pueden provocar que la salida ya no sea la deseada.

- Es necesario que el sistema se este recalibrando constantemente.

Ventajas de un sistema de lazo cerrado son:

- Es más exacto al igualar los valores de la variable real y el que se necesita.
- Es menos sensible a perturbaciones y a los cambios de cada uno de sus elementos
- Su respuesta es más rápida y su ancho de banda es mayor

Las desventajas de un sistema de lazo cerrado son:

- Existe una pérdida en la ganancia, ya que el denominador de la función de transferencia en lazo cerrado es  $1 + GH$  como se observa en la ecuación 2.40.
- Es posible que al agregar la retroalimentación el sistema se vuelva inestable.
- Al existir más elementos de control el sistema se vuelve complejo, caro y propenso a fallos.

En teoría de control se utilizan las funciones de transferencia para establecer la relación que existe entre la entrada y la salida de un sistema. Una función de transferencia de un sistema esta descrito por una ecuación diferencial lineal e invariante en el tiempo y esta definida como el cociente de la transformada de Laplace de la salida y la transformada de Laplace de la función de entrada. [13] [5]

Si tenemos a un sistema lineal e invariante en el tiempo descrito con la siguiente ecuación diferencial:

$$a_0 y^n + a_1 y^{n-1} + \dots + a_{n-1} \dot{y} + a_n y = b_0 u^m + b_1 u^{m-1} + \dots + b_{m-1} \dot{u} + b_m u \quad (2.38)$$

Si llevamos esta ecuación diferencial al dominio de la frecuencia, obtenemos de ella la función de transferencia del sistema:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \dots + b_{m-1} s + b_m}{a_0 s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n} \quad (2.39)$$

donde  $n \geq m$ .

*Características de la función de transferencia.*

- Es un modelo matemático porque en ella queda expresada la ecuación diferencial que relaciona la entrada y la salida de un sistema.
- Es una propiedad del sistema que es independiente de la función de entrada.
- Establece la relación entre la entrada y salida del sistema, pero no ofrece una descripción de este. Varios sistemas físicos distintos pueden tener la misma función de transferencia.

- Al conocer la función de transeferencia de un sistema, es posible estudiar la salida aplicando diferentes señales de entrada, para conocer la naturaleza del sistema.
- Si se desconoce la función de transeferencia, es posible obtenerla experimentalmente aplicando señales de entrada conocidas y estudiando la salida.

Un sistema de control esta formado por varios elementos, donde cada uno de ellos realiza una determinada función. Es posible representar al sistema mediante bloques funcionales donde se muestre el flujo de la señal, este tipo de representación es el más utilizado; cada bloque representa la descripción del comportamiento de cada elemento dentro del sistema de control. [13],[5]



FIGURA 2.17: Representación de un sistema en lazo abierto

Para el caso lineal de un sistema de control en lazo cerrado este puede ser representado en un diagrama de bloques por puntos de suma, bloques y ramificaciones.

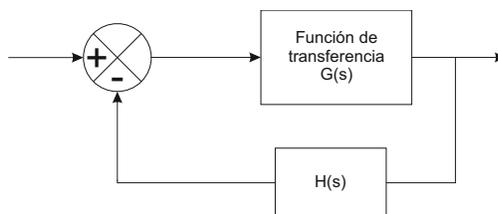


FIGURA 2.18: Representación de un sistema en lazo cerrado

Para el sistema de la figura 2.18 la función de transeferencia del sistema es:

$$\frac{Y(s)}{U(s)} = \frac{G(s)}{1 + G(s)H(s)} \quad (2.40)$$

### Sensibilidad de un sistema de control

Todo sistema del tipo que sea, está sujeto a condiciones climáticas, al paso del tiempo, al desconocimiento de los parámetros exactos del proceso, etc. En un sistema de lazo abierto, estas afectaciones dan lugar a salidas erróneas. En un sistema de lazo cerrado, el sistema intenta regular estos cambios para que no afecten a la salida. Por lo tanto, la *sensibilidad* de un sistema de control de un sistema de lazo cerrado es su capacidad para reaccionar a estos cambios. [13], [5]

La capacidad que presenta un sistema de control de lazo cerrado para reducir las variaciones de los parámetros de operación es una ventaja con respecto a los sistemas de lazo abierto, entonces para que los sistemas de lazo cerrado sean

muy precisos, los elementos de control de lazo abierto deben ser apropiadamente seleccionados.[13],[5]

### 2.7.1. Respuesta transitoria y en estado estable

Para entender la dinámica de un sistema es necesario entender el cambio de comportamiento que sufre un sistema al aplicarle una señal de entrada. Esta respuesta se divide en dos partes: la respuesta transitoria y la respuesta en estado estable. La parte de más interés se centra en la respuesta transitoria, es aquí donde se analiza como responde el sistema ante las perturbaciones a las que es sometido, al conocer de manera precisa se podrá controlar la respuesta del sistema y llevarlo al lugar que se requiere en estado estable. En general la respuesta del sistema para todo el tiempo  $t$  se puede escribir como:[15],[13]

$$y(t) = y_{transitoria}(t) + y_{estadoestable}(t) \quad (2.41)$$

Lo que se busca de un sistema de control es que la respuesta sea la deseada, para esto la respuesta transitoria del sistema es la que se manipula hasta que alcanza el estado estable. Si un sistema en lazo abierto no ofrece la salida que se desea es necesario reemplazarlo con otro que si ofrezca la salida que se busca. En cambio un sistema de lazo cerrado que no ofrece la salida deseada, este es modificado hasta obtener la respuesta deseada. Esto se hace generalmente colocando un controlador antes de la planta dentro del lazo principal del sistema como se muestra en la figura 2.19. [5]

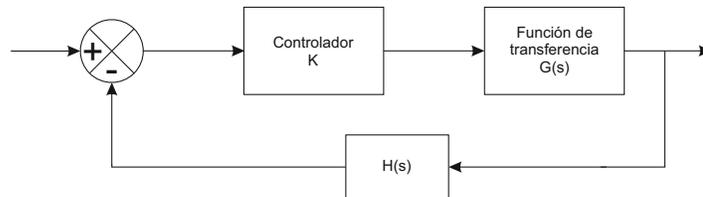


FIGURA 2.19: Sistema de control con un controlador en cascada

La principal ventaja de los sistemas de control en lazo cerrado sobre los de lazo abierto es la capacidad que se tiene de ajustar el sistema para llevarlo al punto de operación deseado. Ajustando su respuesta transitoria que es la que lleva al sistema desde su estado inicial hasta el final y desaparece con el tiempo, y la respuesta en estado estable es la que existe durante largo tiempo conforme  $t$  tiende al infinito. Lo primero que se tiene que hacer es determinar si el sistema es estable, para esto se utilizan señales de prueba para analizar como responde ante señales conocidas, comunmente se utiliza la señal impulso, escalón, rampa y parabólica. En la práctica realmente no se conoce la señal de entrada y se utilizan las señales de prueba ya que aportan características de comportamiento similares a las que se tienen ante las señales de entrada reales.[13],[5]

Una de las ventajas de las señales de prueba es que es posible realizar análisis matemáticos y experimentar con los sistemas de control, ya que éstas señales son

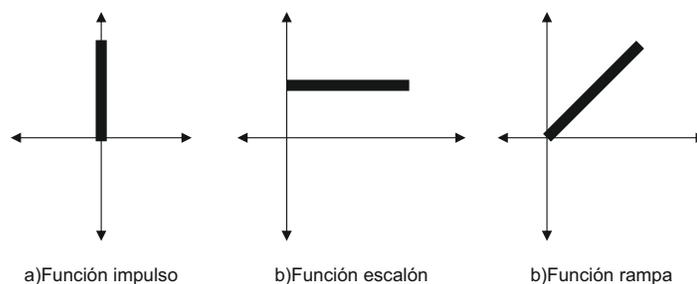


FIGURA 2.20: Señales típicas de prueba

funciones en el dominio del tiempo muy sencillas, en la figura 2.20 se muestran ejemplos de las funciones prueba más utilizadas.[13]

### 2.7.2. Sistemas de primer orden

Un sistema de primer orden es aquel que se puede representar por una ecuación diferencial de primer orden:

$$a_{n-1}\dot{y} + a_n y = b_m u \tag{2.42}$$

Transformando esta al dominio de la frecuencia, dejando la ecuación en términos de la ganancia y de la constante de tiempo, esto queda:

$$\frac{Y(s)}{U(s)} = \frac{k}{\tau s + 1} \tag{2.43}$$

donde  $k$  es la ganancia del sistema y  $\tau$  es la constante de tiempo.

La representación en diagrama de bloques para la ecuación 2.43 se muestra en la figura 2.21.

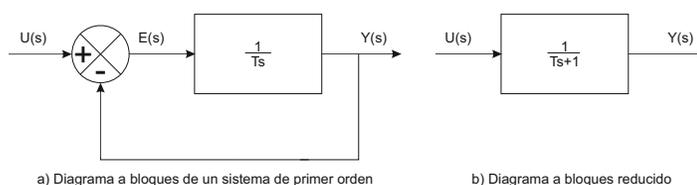


FIGURA 2.21: Diagrama a bloques de un sistema de primer orden

### Respuesta impulso unitario de sistemas de primer orden

Cuando la entrada del sistema es una función impulso unitario, que se define como:

$$u(t) = \begin{cases} \lim_{t \rightarrow \infty} \frac{A}{t_o} & \text{para } 0 < t < t_o \\ 0 & \text{para } t < 0, t_o < t \end{cases}$$

Utilizando transformada de Laplace, la entrada es  $U(s) = 1$  para cuando  $A = 1$ , sustituyendo el valor de  $U(s)$  en la ecuación 2.43 y aplicando la transformada inversa de Laplace se obtiene:

$$y(t) = \frac{1}{\tau} e^{-\frac{t}{\tau}} \quad (2.44)$$

La curva de la ecuación 2.44 es la respuesta del sistema en el dominio del tiempo  $(t)$ , la podemos observar en la figura 2.22.

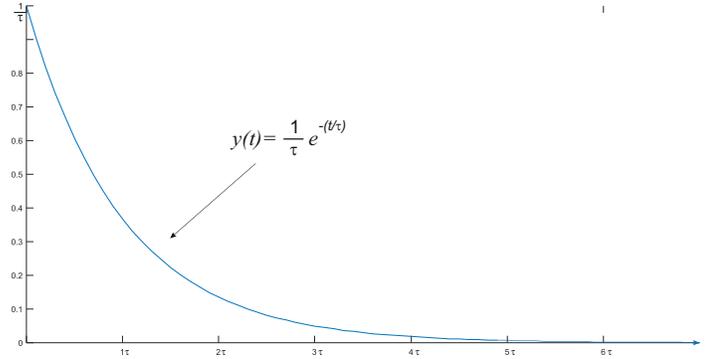


FIGURA 2.22: Respuesta a impulso unitario de un sistema de primer orden

### Respuesta escalón unitario de sistemas de primer orden

La función escalón unitario en el dominio de la frecuencia es  $1/s$ , entonces el sistema de primer orden ante una entrada escalón unitario, queda como en la ecuación 2.45.

$$Y(s) = \frac{k}{\tau s + 1} \frac{1}{s} \quad (2.45)$$

Al resolver la ecuación 2.45, obtenemos la ecuación 2.46 para  $t \geq 0$ .

$$y(t) = 1 - e^{-\frac{t}{\tau}} \quad (2.46)$$

La curva de la ecuación 2.46 es la respuesta del sistema en el dominio del tiempo  $(t)$ , la podemos observar en la figura 2.23.

Como se puede apreciar en la figura 2.23 para cuatro constantes de tiempo, el valor de la respuesta alcanza el 98 % del valor de estado estable, para fines prácticos se considera la respuesta transitoria del sistema va de 0 hasta  $4\tau$ , y para valores superiores de  $t > 4\tau$  tenemos la respuesta en estado estable. [13]

### Respuesta rampa unitaria de sistemas de primer orden

La función rampa unitaria en el dominio de la frecuencia es  $1/s^2$ , reemplazamos el valor de  $U(s)$  en la ecuación 2.43 y la salida  $Y(s)$  queda como:

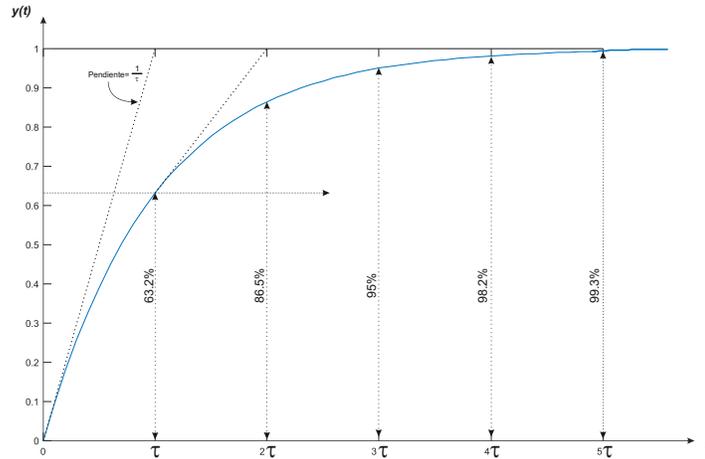


FIGURA 2.23: Curva de respuesta exponencial

$$Y(s) = \frac{k}{\tau s + 1} \frac{1}{s^2} \tag{2.47}$$

Resolviendo esta ecuación para  $t$  utilizando la transformada inversa de Laplace, tenemos para  $t \geq 0$ :

$$y(t) = t - \tau + \tau e^{-\frac{t}{\tau}} \tag{2.48}$$

Si graficamos la ecuación 2.48 y la comparamos con la entrada, se puede apreciar el error en la salida es igual a:

$$\begin{aligned} e(t) &= u(t) - y(t) \\ &= \tau(1 - e^{-\frac{t}{\tau}}) \end{aligned}$$

Para este caso el error  $e(t) = 0$  cuando  $t = \infty$ .

### Propiedad de los sistemas lineales invariantes en el tiempo

Como se puede apreciar en la figura 2.24 mientras más pequeña es la constante de tiempo  $\tau$  menor es el error en estado estacionario.

Para sistemas invariantes en el tiempo de primer orden se puede observar que cuando la entrada es la función rampa, la salida es

$$y(t) = t - \tau + \tau e^{-\frac{t}{\tau}}$$

Si derivamos la ecuación 2.48 con respecto al tiempo, obtenemos la salida 2.46 de un sistema de primer orden cuando la entrada es la función escalón

$$y(t) = 1 - e^{-\frac{t}{\tau}}$$

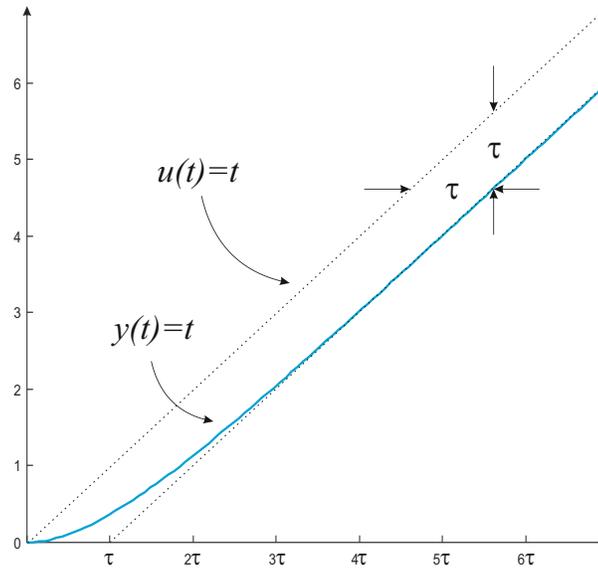


FIGURA 2.24: Respuesta a una entrada rampa de un sistema de primer orden

De nuevo, si se deriva la ecuación 2.46 obtenemos la salida de una función impulso para un sistema de primer orden representado por la ecuación 2.44.

$$y(t) = \frac{1}{\tau} e^{-\frac{t}{\tau}}$$

Esta propiedad solo existe en los sistemas lineales e invariantes en el tiempo. [13]

### 2.7.3. Sistemas de segundo orden

La ecuación diferencial 2.49 representa a un sistema de segundo orden, es el más típico que existe.

$$a_0 \ddot{y} + a_2 \dot{y} + a_3 y = ku(t) \tag{2.49}$$

Transformado la ecuación 2.49 al dominio de la frecuencia para obtener su función de transferencia tenemos:

$$\frac{Y(s)}{U(s)} = \frac{k}{s(s+p)} \tag{2.50}$$

Donde  $k = \text{Ganancia}$  y  $p = \text{Polo del sistema}$ , la representación en lazo cerrado se muestra en la figura 2.25.

Si obtenemos las raíces de la función de transferencia en lazo cerrado estas son:

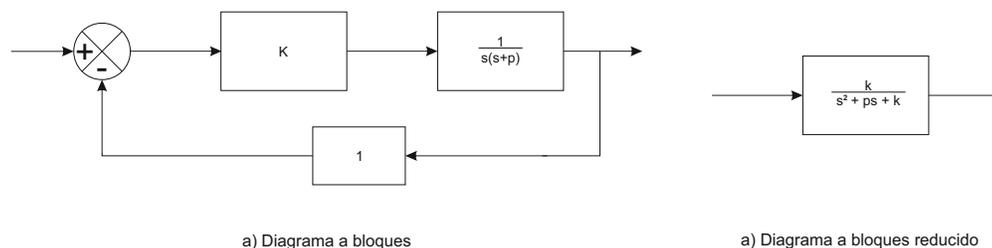


FIGURA 2.25: Diagrama a bloques de un sistema de segundo orden

## 2.8. Lógica digital con dispositivos FPGA

La tendencia hoy en día es que los dispositivos digitales sean cada vez más veloces como una de sus características principales, y la velocidad de un dispositivo se tiene que ver desde tres puntos de vista: rendimiento, latencia y temporización. Desde el punto de vista del procesamiento de información rendimiento se refiere a la capacidad que tiene un dispositivo para procesar datos en cada ciclo de reloj. Latencia es el retardo que existe desde el momento en que se recibe una señal de entrada y se obtiene una salida y también se cuenta en ciclos de reloj. La sincronización se refiere al retardo que existe entre cada uno de los elementos debido a sus propias características de respuesta en el tiempo. La métrica para la temporización son los ciclos de reloj y la frecuencia. [7]

Un diseño de alto rendimiento tiene que ver con la cantidad de datos que pueden ser procesados sin atender a los tiempos de conmutación de los elementos que lo forman. Lo que interesa es que mientras que un proceso tiende a ser más rápido que otro el sistema pueda estar ejecutando otros procesos para que al final la respuesta general del sistema sea la más rápida que se pueda obtener, es el mismo concepto que se utiliza en las ensambladoras de autos, si un proceso te consume más tiempo que otros, se debe compensar ese retraso realizando otras tareas en paralelo para que el tiempo final del proceso sea menor. En el campo de procesamiento de datos se utiliza el término: pipeline.[7]

Una ventaja de los sistemas pipeline es se pueden estar procesando datos nuevos antes de que los anteriores se terminen de procesar, tal como ocurre en las líneas de ensamble de autos. Desde el punto de vista algorítmico un diseño pipeline lo que busca es desentramar los bucles; un bucle es un algoritmo interactivo donde se accesa una y otra vez a las mismas variables y las mismas direcciones de memoria hasta completar un cálculo, en este caso no hay uso del paralelismo porque un microprocesador sólo puede procesar una instrucción a la vez. Dentro de un algoritmo optimizado los bucles son desentramados y con esto aumenta el rendimiento, aunque es muy probable que se necesite incrementar el número de variables, registros u operaciones para realizar los cálculos esto no debe preocupar ya que existen métodos para hacer más eficientes este tipo de algoritmos. [7]

Un diseño de baja latencia es aquel que una vez que tiene los datos en la entrada la salida se muestra lo más pronto posible. Un diseño de este tipo sin duda requiere de paralelismos y eliminar todo aquello que pueda reducir la

velocidad máxima de reloj dentro del diseño como es el uso de registros, pero existe un castigo para esto ya que aumenta el retardo combinatorio entre registros restantes, ya que se tiene que aumentar el tamaño del bus y se deben utilizar operaciones extra para unir los procesos.[7]

### 2.8.1. Características de una implementación con FPGA

La temporización se refiere a la velocidad de reloj de un diseño. El retardo que existe entre dos o más elementos que estén secuencialmente ordenados determina la velocidad máxima de reloj. En una implementación pipeline no se puede saber si funcionará más rápido que una iterativa sin conocer los detalles de la implementación. Un buen diseño en la temporización es cuando se cumple que el retardo máximo entre dos elementos secuenciales es menor que un tiempo en el ciclo de reloj.[7] Para realizar estas mejoras se puede:

- Agregar capas de registro ya que estas mejoran el tiempo al dividir la ruta crítica en rutas con un menor retardo.
- Al descomponer funciones lógicas en otras más pequeñas que se puedan evaluar en paralelo, esto conlleva a reducir el retardo en la trayectoria más larga de las subestructuras.
- Eliminar las codificaciones de prioridad en las que no se necesitan, esto contrae la estructura lógica y reduce el retardo de la trayectoria.
- Balancear los registros mejora la sincronización moviendo la lógica combinatoria de la ruta crítica a una ruta adyacente.
- La sincronización se puede mejorar al reordenar las trayectorias que se combinan con la trayectoria crítica de tal forma que parte de la lógica que compone a la ruta crítica se coloca más cerca del registro de destino.

La implementación de una topología de área es aquella donde se pone por encima de la velocidad, la reutilización de recursos y se utilizan algoritmos recursivos. En este tipo de implementaciones utilizan más recursos para retener valores intermedios y se replican estructuras que se ejecutan en paralelo, lo que ocurre generalmente es que hay lógica duplicada.[7] Otra técnica que se utiliza es reutilizar la lógica, cuando la lógica compartida es mayor que la lógica de control es común utilizar máquinas de estado como entrada adicional a la lógica. Por último, cuando se comparten recursos se refiere a los bloques funcionales son reutilizables aún cuando éstos se encuentren en distintos módulos, hay que llevar a que éstos recursos sean globales.[7]

Los dispositivos FPGA (Field Programmable Gate Array) no son adecuados para hacer implementación de bajo consumo de energía, aunque algunos fabricantes ofrecen CPLDs (complex programmable logic devices) como alternativas, lo cierto es que éstos dispositivos están muy limitados en tamaño y capacidad y no son adecuados cuando se requiere potencia de cálculo.[7]

### 2.8.2. Lenguaje de descripción de hardware

Debido a la creciente complejidad de los dispositivos FPGA, así como a la sofisticación y capacidad de las herramientas de diseño, pasar de un diseño basado en esquemas a HDL (Hardware Description Language) fué todo un hito, lo que permitió a los programadores describir módulos de manera funcional que son independientes de la tecnología utilizada. [7]

De esta tecnología nació VHDL que significa VHSIC (Very High Speed Integrated Circuit) HDL, que traducido al español es lenguaje de descripción de hardware de circuitos integrados de muy alta velocidad. Originalmente VHDL fué diseñado para simulación de sistemas digitales, actualmente es también utilizado para síntesis de circuitos. Debido esta concepción original de simulación algunas descripciones no pueden ser sintetizadas, y por otro lado VHDL es muy flexible y permite distintas descripciones para una misma funcionalidad. VHDL presenta tres tipos de descripción de circuitos: estructural, comportamental o algorítmico y funcional. Estos estilos tienen relación con el nivel de abstracción que se desarrolle.[7] [6]

Las ventajas de VHDL son:

- Es un lenguaje basado en un estándar del (IEEE Std 1076-1987, IEEE Std 1076-1993), por tanto, es un lenguaje normalizado, compatible con la mayoría de las herramientas de diseño.
- Permite utilizar distintos estilos de diseño facilitando la tarea del ingeniero de diseño.
- Un módulo creado en VHDL puede ser utilizado en distintos diseños, así que una reutilización de código es posible.
- Permite Mejorar la calidad y el tiempo de un diseño.
- Permite usar módulos, lo cual es utilizado para reducir las unidades de diseño, con esto se logran crear bloques de programación que facilitan el desarrollo por parte del ingeniero.
- VHDL permite hacer simulaciones y mejorar la funcionalidad.

En general, VHDL permite tres tipos de descripción de hardware: [3]

- Estructural, donde se especifican los componentes del sistema y sus interconexiones.
- Funcional, no se necesita conocer la estructura interna de un circuito y es posible describirlo mediante su comportamiento funcional.
- Flujo de datos.

### 2.8.3. Estructura de VHDL

El lenguaje de programación VHDL se estructura en módulos o unidades funcionales, donde cada módulo tiene una secuencia de instrucciones. La estructura de cada programa invariablemente esta formada por una Entidad (*Entity*)

y una Arquitectura (*Architecture*), además como puede contener Bibliotecas (*Library*), Paquetes (*Package*) y Componentes (*Component*), que sirven para optimizar y reutilizar código. Todos éstos módulos representan las unidades fundamentales del código VHDL. [11]

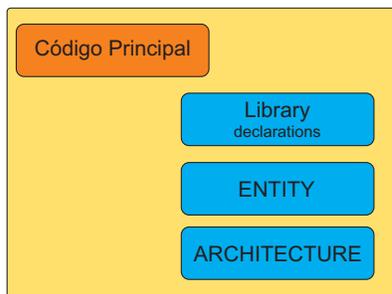


FIGURA 2.26: Unidades fundamentales del código VHDL

### Librerías y paquetes

Las librerías y los paquetes son usados en los lenguajes de programación, y son mecanismos que sirven para reutilizar código. Las librerías pueden contener uno o más paquetes y a su vez los paquetes pueden ser creados por el usuario o estar definidos por algún mecanismo regulador o a través de estándares como el IEEE (Institute of Electrical and Electronics Engineers) como parte de los estándares que forman al lenguaje VHDL. [3] [6]

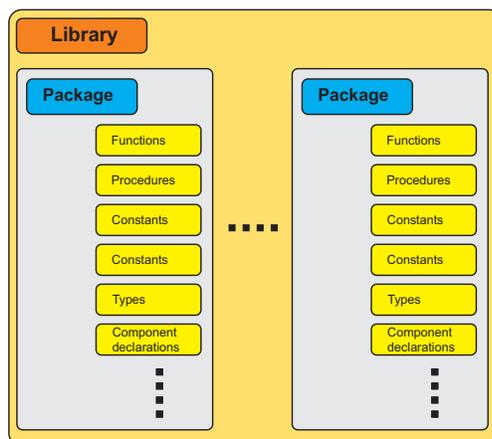


FIGURA 2.27: Estructura de las librerías y los paquetes

### Entidad(entity)

La entidad es la estructura fundamental y describe características físicas del dispositivo como lo son sus entradas y salidas. La entidad es un bloque del diseño que es tratado como un elemento individual, y puede representar una

compuerta, un decodificador, una ALU (Arithmetic Logic Unit), y se le puede considerar un sistema al relacionar las entradas y salidas, que representan puntos de conexión con otros elementos del sistema que se esté desarrollando.[3] [6] Cada una de las señales que entran o salen de una entidad se le conoce como puerto, el cual equivale a decir que es una terminal de salida o entrada en un símbolo esquemático. Todo puerto que es declarado en la entidad debe tener un *nombre*, un *modo* y debe especificar un *tipo de dato* que lo define. El *nombre* se utiliza para poder nombrar al puerto, el *modo* establece la dirección de flujo de datos, si son datos de entrada, de salida o ambos; el *tipo de dato* se refiere a la clase de datos que serán transmitidos o recibidos por ese puerto. [3]

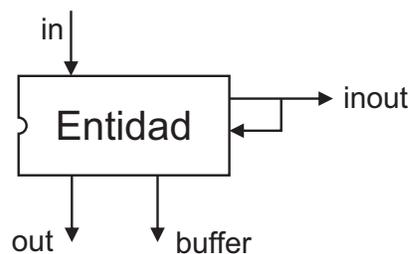


FIGURA 2.28: Representación de una entidad

Para declarar una entidad se da una descripción de las entradas y las salidas con los que cuenta el circuito que se está diseñando. Por ejemplo, si se desea diseñar un multiplexor de dos entradas y una salida, con una señal para el selector, se tiene que la representación esquemática del multiplexor corresponde a la figura 2.29. [3]

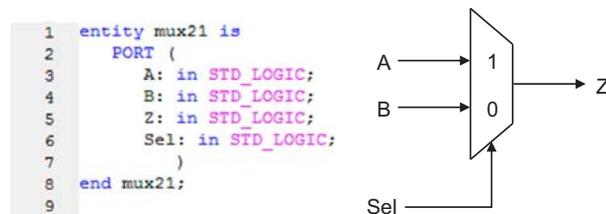


FIGURA 2.29: Código VHDL y representación esquemática

### Arquitectura(architecture)

Definida la entidad ahora es posible definir la arquitectura cuyo propósito es definir las funciones que realizará el módulo que se describió en la entidad, por tanto, contiene algoritmos que ejecutan las instrucciones que se deben llevar a cabo. Es aquí donde entra la creatividad del ingeniero de diseño y podrá hacer las descripciones de hardware por el método que más se le facilite, existen muchas formas de implementar una solución a un problema, y depende del tipo de aplicación, el tamaño de esta y del estilo del programador. [3],[14]

## Descripción por flujo de datos

Para describir la arquitectura mediante una descripción de flujo de datos es necesario mostrar la como se da la transferencia de información entre las entradas y salidas de una entidad. Este tipo de descripción es usada cuando se utilizan tablas de verdad y ecuaciones lógicas como parte principal de la descripción de un circuito. Un ejemplo de esto es cuando se realiza la descripción de una compuerta lógica dentro de una arquitectura, por ejemplo en la figura 2.28 tenemos una representación de una y se da el código en VHDL de esa entidad que representa las entradas y salidas de un multiplexor, ahora se muestra la descripción en lenguaje VHDL de la arquitectura para esa entidad. [3],[14]

```
10:  -- Esta es la arquitectura de un Mux 2 a 1
11:  Architecture circuito of Mux21 is
12:      begin
13:          Z<=A when Sel='1' else B;
14:      end circuito;
```

Esta codificación o estilo de programación se basa en la utilización de los operadores lógicos del álgebra booleana que son parte de las instrucciones básicas del lenguaje VHDL que están definidos dentro de los distintos tipos de datos (bit, boolean y std-logic), cuando se utilizan éstos operadores lógicos realizan operaciones uno a uno, esto significa, que el tamaño o longitud de la palabra debe ser el mismo entre las variables sobre las cuales se realiza la operación. El orden de prioridad de los operadores lógicos es: [3]

- Paréntesis.
- Complementos.
- Función And.
- Función or.

## Descripción funcional

En este tipo de descripción se puede suponer a las entidades como una especie de caja negra, en donde no importa su estructura interna y lo que realmente interesa es lo que se obtiene a la salida. Primordialmente este tipo de descripción se basa en el uso de procesos y declaraciones secuenciales, esta descripción funcional permite un rápido modelado de la función. Supongamos que deseamos hacer una descripción funcional de la compuerta Nand, cuya función de transferencia se conoce y se muestra en lenguaje VHDL. [3] [6]

## Descripción estructural

Este tipo de descripción se basa en la utilización de estructuras o módulos ya definidas que se vuelven a utilizar para crear un sistema más grande, esto significa que un módulo se convierte en componente de otro módulo. Al realizar una descripción estructural se obtienen múltiples niveles de jerarquía y los componentes pueden ser agregados no importando que la descripción de éstos módulos sea de otro tipo. [6] [11]

Los componentes de una descripción estructural se pueden ver en la figura 2.30

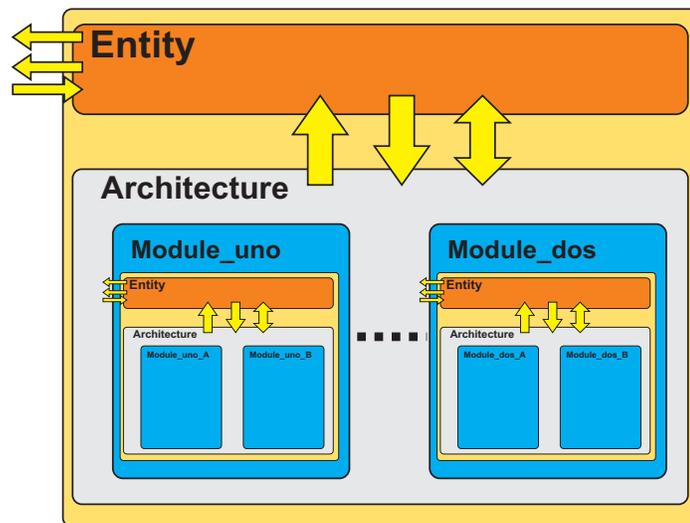


FIGURA 2.30: Descripción estructural

Tomando la figura 2.30 y escribiendo su representación en código, obtenemos:

```

1: entity moduloprincipal is
2:   Port ( x1 : in  STD_LOGIC; --entradas
3:         x2 : in  STD_LOGIC;
4:         y1 : in  STD_LOGIC; --entradas
5:         y2 : in  STD_LOGIC;
6:         z  : out STD_LOGIC);--salidas
7: end entity moduloprincipal;
8:
9: Architecture Behavioral of moduloprincipal is
10:
11:   component Module_uno is
12:     port ( a1 : in  STD_LOGIC;
13:          a2 : in  STD_LOGIC;
14:          a3 : out STD_LOGIC);
15:   end component Module_uno;
16:
17:   component Module_dos is
18:     port ( b1 : in  STD_LOGIC;
19:          b2 : in  STD_LOGIC;
20:          b3 : out STD_LOGIC);
21:   end component Module_dos;

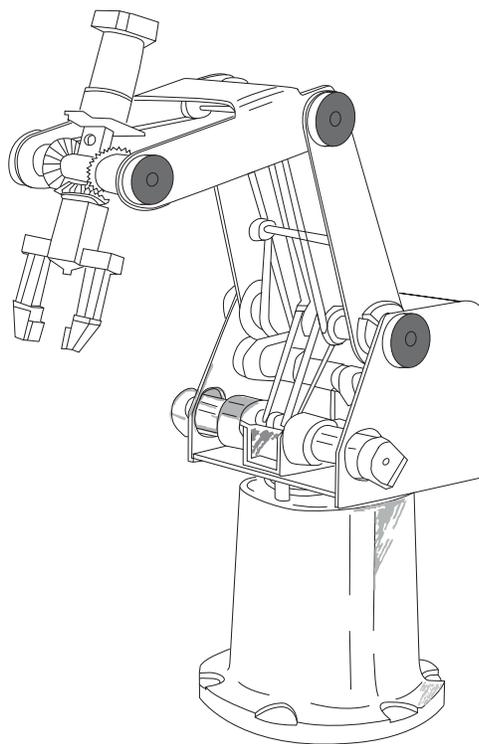
```

```
22:
23:   signal z1, z2 : std_logic := 'U'; --conectando se ales
24:
25:   begin
26:
27:       sub1: Module_uno port map (a1=>x1, a2=>x2, a3=>z1);
28:       sub2: Module_dos port map (b1=>y1, b2=>y2, b3=>z2);
29:       z <= z1 xor z2;
30:
31:   end Behavioral;
```

## Capítulo 3

# Análisis cinemático

Para construir el modelo cinemático que describe el movimiento del robot, se tiene que analizar la estructura del robot para la cual se construye dicho modelo, el robot utilizado es el Scorbot ER-V Plus y se analizan 3-gdl, nos interesan las posiciones del extremo final del segundo eslabón del robot, al observar la figura podemos ir identificando cada una de sus partes. [8]



---

FIGURA 3.1: Robot Scorbot ER-V Plus

Al desglosar las partes del brazo robot podemos identificar sus eslabones y sus articulaciones para poder entonces poder numerar los sistemas de referencia y el tipo de movimiento que tiene cada articulación. Como se puede observar en la figura 3.2 se muestran los eslabones y las articulaciones que posee la estructura. [8]

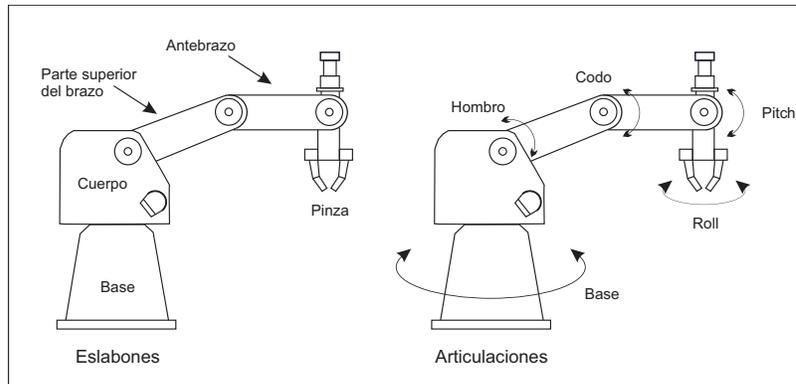


FIGURA 3.2: Elementos de un brazo robot

### 3.1. Obtención del modelo Denavit-Hartenberg

Para utilizar la metodología Denavit-Hartenberg, lo primero que se hace es describir la convención que se usa para establecer los sistemas de referencia de cada uno de los eslabones del robot que se está analizando.

Como ya se tiene identificada cada una de las articulaciones y de los eslabones que la forman se pueden establecer los sistemas de referencia asociados a cada una de ellas utilizando el algoritmo Denavit-Hartenberg. Es necesario medir las distancias entre los ejes como se muestra en la figura 3.3. [8]

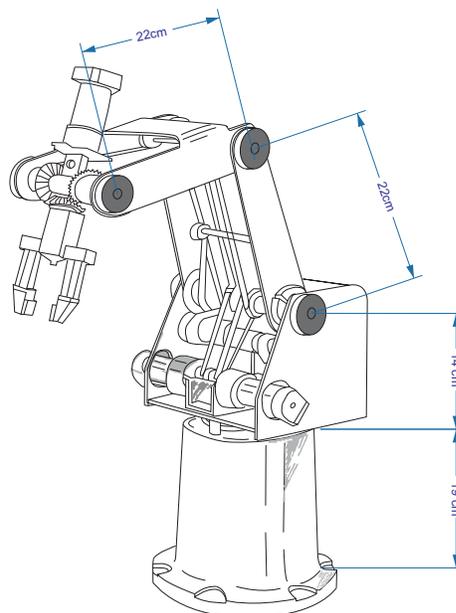


FIGURA 3.3: Medidas del robot Scorbobot ER-V Plus

Se procede a aplicar la secuencia de pasos de del algoritmo del DH1 al DH9

- DH1. Se parte del primer eslabón móvil de la cadena numerándolo, la base se numera con cero ya que es fija.
- DH2. Se numera la primera articulación, que corresponde al primer gdl y se termina en  $n$ .
- DH3. Se localiza cada eje de cada articulación.
- DH4. Hay que situarse en el eje  $Z_0$ .
- DH5. Situar el origen  $S_0$  en cualquier punto sobre el eje  $Z_0$ .
- DH6. Situar  $S_1$  utilizando el punto donde se corta con  $S_0$ , y se repite esto para cada articulación.
- DH7. Situar  $X_1$  en la línea normal que es común a  $Z_0$  y  $Z_1$ , repetir esto para todas las articulaciones.
- DH8. Situar  $Y_0, Y_1$ , y continuar para cada una de las articulaciones.
- DH9. Situar el último sistema de referencia en la última articulación del sistema.

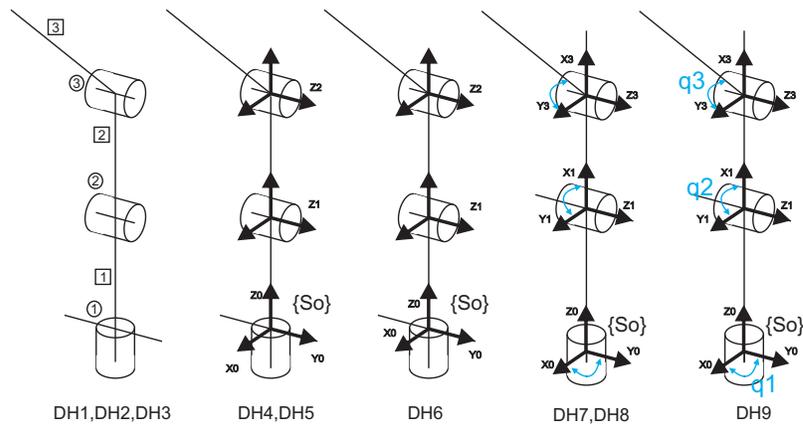


FIGURA 3.4: Secuencia del algoritmo DH para el robot Scorbot ER-V Plus

Ya que se tienen definidos los sistemas de referencia del brazo, se puede obtener la tabla de parámetros DH que corresponde a los pasos DH10 al DH13 3.1.

CUADRO 3.1: Parámetros D-H para el robot Scorbot ER-V plus

Articulación	$\theta$	$d$	$a$	$\alpha$
1	q1	14	0	-90
2	q2	0	22	0
3	q3	0	22	0

Con la tabla 3.1 se obtienen las matrices:

$$[H]^0 A_1 = \begin{bmatrix} \cos(q_1) & 0 & -\text{sen}(q_1) & 0 \\ \text{sen}(q_1) & 0 & \cos(q_1) & 0 \\ 0 & -1 & 0 & 14 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

$$[H]^1 A_2 = \begin{bmatrix} \cos(q_2) & -\text{sen}(q_2) & 0 & 22\cos(q_2) \\ \text{sen}(q_2) & \cos(q_2) & 0 & 22\text{sen}(q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

$$[H]^2 A_3 = \begin{bmatrix} \cos(q_3) & -\text{sen}(q_3) & 0 & 22\cos(q_3) \\ \text{sen}(q_3) & \cos(q_3) & 0 & 22\text{sen}(q_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.3)$$

De la multiplicación de las matrices 3.1,3.2,3.3 se obtiene la matriz  $T$ .

$$[H]T = \begin{bmatrix} n & o & a & p \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^0A_3 = {}^0A_1 {}^1A_2 {}^2A_3$$

Realizada la operación de multiplicación podemos observar los valores  $x$ ,  $y$ ,  $z$  de las posiciones finales del extremo del brazo robot y corresponden a los tres primeros valores de la cuarta columna de la matriz  $T$  resultante.

$$\begin{aligned} x &= 22 * \cos(q_1) * \cos(q_2) + 22 * \cos(q_1) * \cos(q_2) * \cos(q_3) - 22 * \cos(q_1) * \sin(q_2) * \sin(q_3) \\ y &= 22 * \cos(q_2) * \sin(q_1) - 22 * \sin(q_1) * \sin(q_2) * \sin(q_3) + 22 * \cos(q_2) * \cos(q_3) * \sin(q_1) \\ z &= 14 - 22 * \cos(q_2) * \sin(q_3) - 22 * \cos(q_3) * \sin(q_2) - 22 * \sin(q_2) \end{aligned}$$

## 3.2. Implementación del modelo

Para poder visualizar éstos resultados utilizamos las siguientes instrucciones que tiene integradas el Robotics Tool Box de Matlab para generar la simulación correspondiente:

```

1  syms q1 q2 q3 l1 l2
2  pi1=sym('pi')
3  A01=mdh(q1,14,0,-pi/2)
4  A12=mdh(q2,0,22,0)
5  A23=mdh(q2,0,22,0)
6  T=A01*A12*A23
7  l1=22;l2=22
8  L1=link([pi/2 0 0 14])
9  L2=link([0 22 0 0])
10 L3=link([0 22 0 0])
11 RITH=robot({L1,L2,L3},'Robot ITH')
12 drivebot(RITH)

```

### 3.3. Manipulación del modelo desarrollado

Una vez que se ha creado el robot en Matlab se despliegan dos ventanas: la primera con la estructura del robot simplificada 3.5 y la segunda 3.6 con los valores de los ángulos  $q_1$ ,  $q_2$  y  $q_3$  y de las posiciones que va tomando el robot.

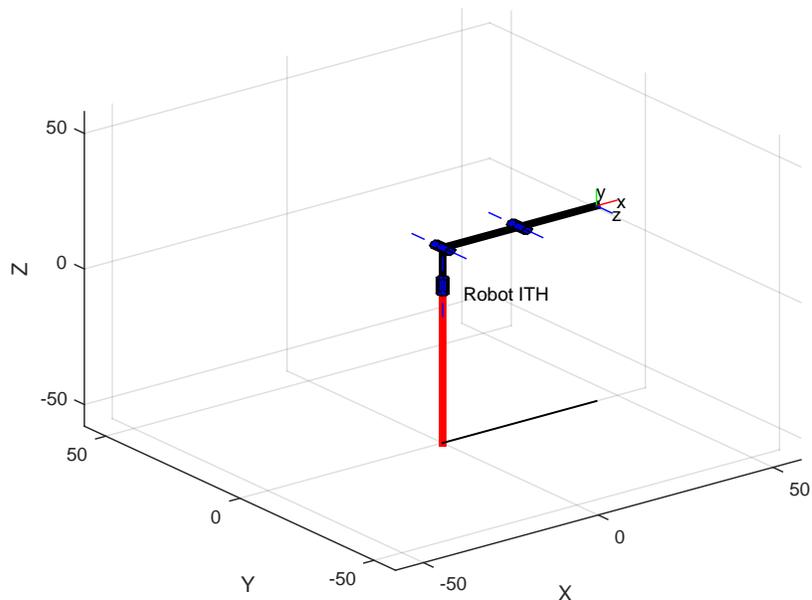


FIGURA 3.5: Posición inicial del Robot

Esta gráfica 3.6 muestra los valores de las coordenadas articulares.

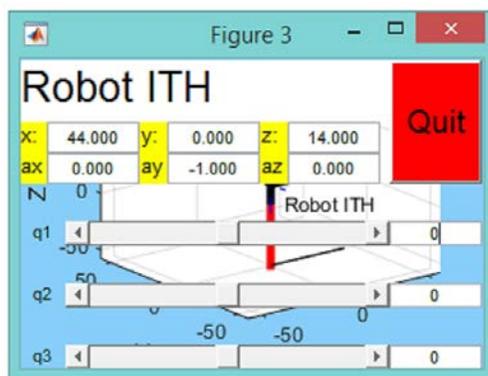


FIGURA 3.6: Coordenadas articulares

Ya que se ha identificado la posición inicial, como corresponde a la convención Denavit-Hartenberg que se ha hecho, se procede a manipular los ángulos  $q_1$ ,  $q_2$ ,

$q_3$  para observar las posiciones que va tomando el extremo final del robot. Al variar  $q_1$  90 en sentido contrario a las manecillas del reloj observamos que el robot se mueve como se muestra en la figura 3.7.

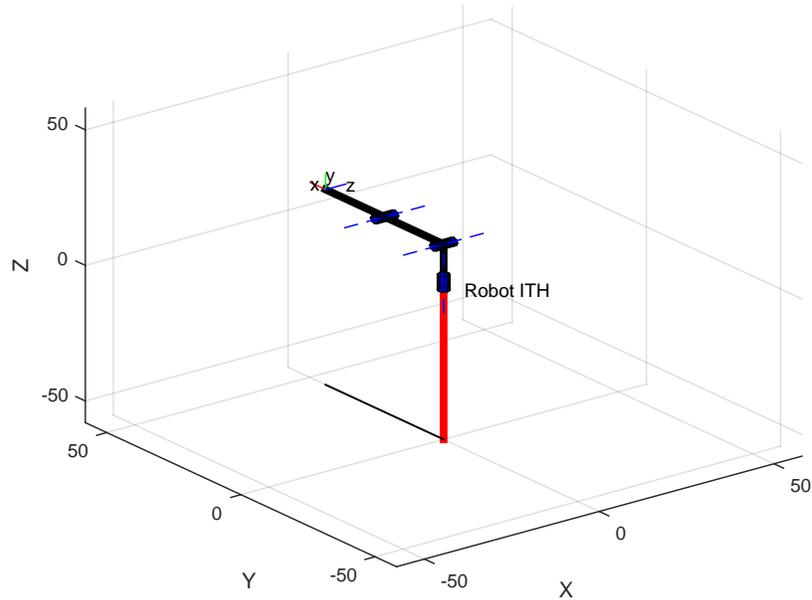


FIGURA 3.7: Variando  $q_1$  90

Esta gráfica 3.8 muestra las coordenadas articulares.

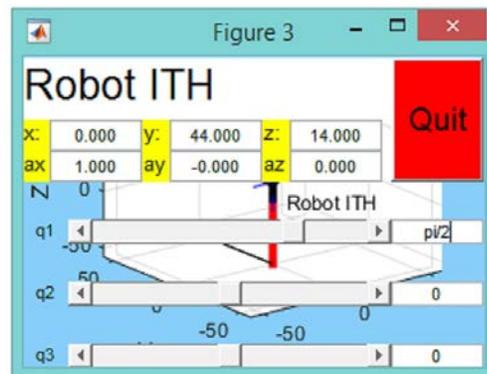


FIGURA 3.8: Coordenadas articulares  $q_1$  rotado 90

Ahora al mover  $q_2$  90 el robot alcanza las nuevas posiciones como se muestra en la figura 3.9, y se mantiene  $q_1$  rotado 90 grados.

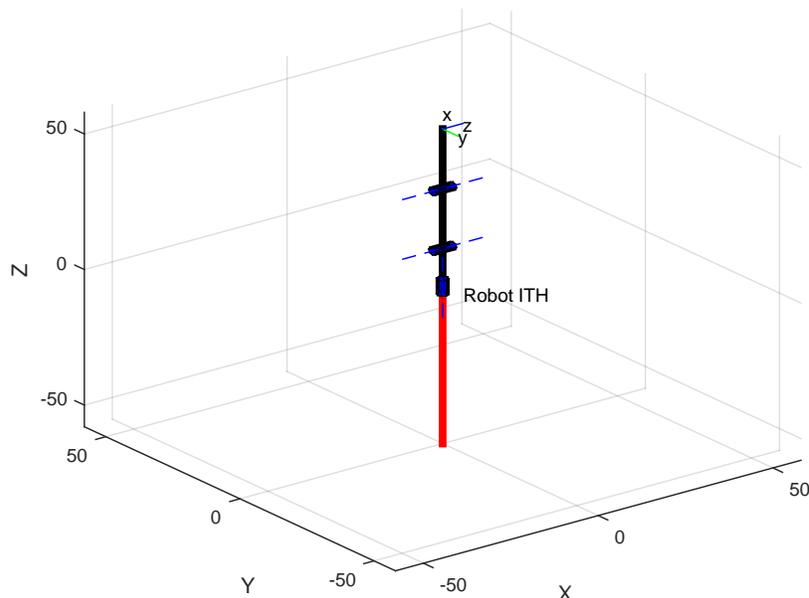


FIGURA 3.9: Variando  $q_1$  y  $q_2$  90

Esta gráfica 3.10 muestra las coordenadas articulares y podemos observar que posición en  $z = 58$  que corresponde a la suma algebraica del eslabón uno  $l_1 = 22\text{cm}$  el eslabón dos  $l_2 = 22\text{cm}$ , más la altura de la base para la cual tomamos la distancia que hay entre el origen donde se ubica nuestro primer sistema de referencia y el segundo sistema de referencia donde se ubica la primera articulación cuya distancia es 14cm.

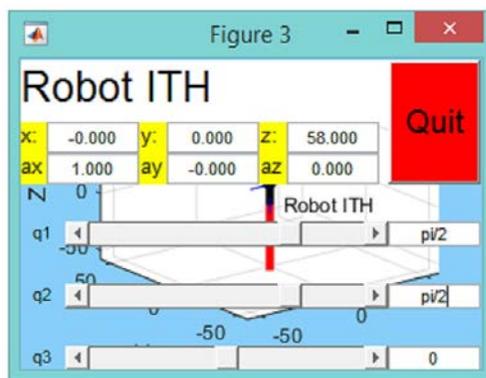


FIGURA 3.10: Coordenadas articulares  $q_1$  y  $q_2$  rotado 90

Por último se varía  $q_3$  a  $-90$  ya que el movimiento es en el sentido de las manecillas del reloj, los ángulos  $q_1$  y  $q_2$  se dejan en su posición inicial, físicamente

$q_3$  sólo se puede variar en el sentido de las manecillas del reloj así que se trabaja con valores negativos de su ángulo de movimiento, figura 3.12.

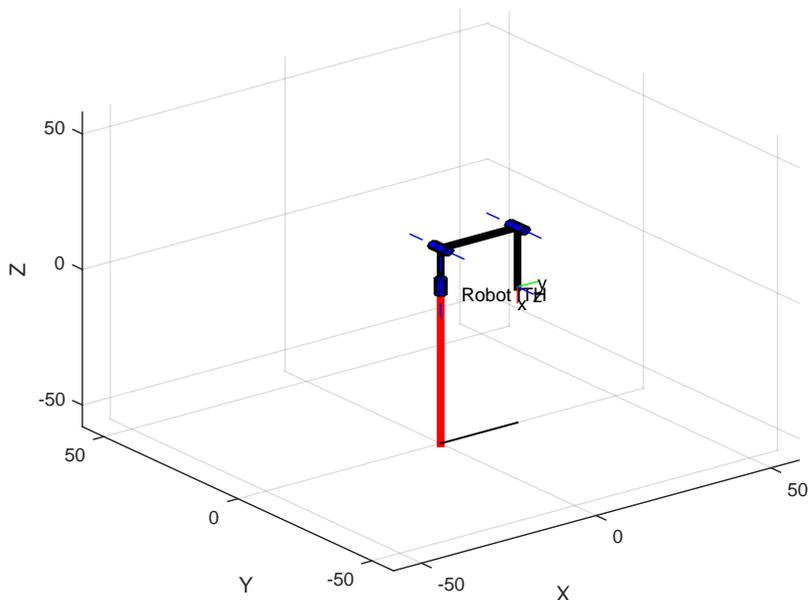


FIGURA 3.11: Variando  $q_3$  90

Esta gráfica 3.12 muestra las coordenadas articulares y nos permite cambiar la posición del Robot, variando los ángulos  $q_1, q_2$  y  $q_3$ .

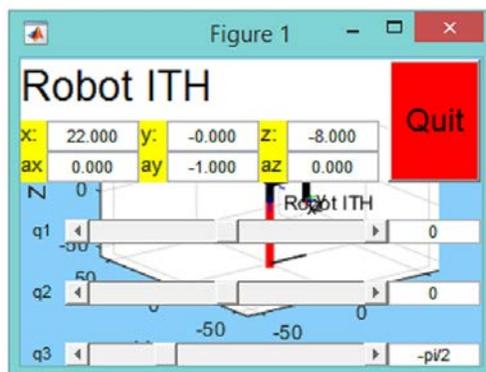


FIGURA 3.12: Coordenadas articulares  $q_3$  rotado -90

Dentro de la simulación podemos dar cualquier valor a los ángulos  $q_1, q_2$  y  $q_3$ ; ya en el sistema físico se está restringido el movimiento por la estructura física del brazo, esta herramienta de simulación permite observar las posiciones que alcanza el brazo y resulta muy útil.

## Capítulo 4

# Implementación

Para la implementación física de éste controlador se tienen varias alternativas para el control como son la utilización de plataformas como Arduino, Rasberry, u otros similares a este tipo de tecnologías; para este trabajo se ha seleccionado un dispositivo FPGA Nexys 2, ya que se busca que la programación sea concurrente y trabaje en paralelo, no es el alcance de este trabajo presentar las ventajas o desventajas de los distintas tecnologías con que puede desarrollarse el controlador.

Para la etapa de potencia se utiliza un inversor comercial L298 que cumple con las especificaciones suficientes para brindar una adecuada señal de potencia hacia los actuadores que trae integrados la estructura mecánica del brazo robot y se diseño una tarjeta con optoacopladores para proteger la FPGA del sistema de potencia.

La estructura de control que se utiliza es la que se muestra en la figura 4.1.

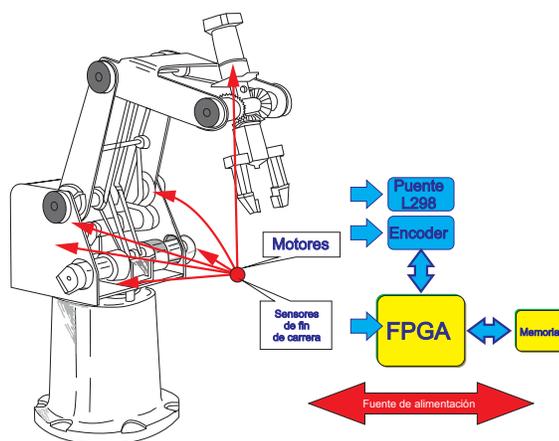


FIGURA 4.1: Sistema de control para el brazo Robot

El flujo de la señal se genera dentro del FPGA, donde primero se lee el valor de las entradas de los sensores de límite que están conectados a las entradas PMOD de la tarjeta Nexys para sensar si alguno esta en su posición inicial, aquellas articulaciones que no estén en su origen se les envía una señal que va hacia los inversores para que a su vez los actuadores lleven la articulación hasta el sensor de límite y que dicha articulación quede en su posición de inicio. Una vez que el robot esta en su posición inicial los valores de los ángulos  $q_1, q_2$  y  $q_3$

se pueden calcular a partir de la posición de referencia inicial dada por el análisis Denavit-Hartenberg según la convención que se usó para establecer los sistemas de referencia.

#### 4.1. Inversor de puente completo L298

Los actuadores que le dan la fuerza mecánica al brazo con la cual se generan los movimientos, está compuesta de motores de corriente directa, por tanto, el movimiento se efectúa controlando el voltaje aplicado a los motores, y el sentido de estos se hace invirtiendo la polaridad que se aplica al motor. Para lograrlo se utilizan inversores de puente completo L298 que proporcionan todas las características de control requeridas para los motores que se tienen, esta relación se muestra en la tabla 4.1.

CUADRO 4.1: Necesidades de los motores contra características del inversor

[HTML]9AFF99[HTML]009901 Características de los motores	[HTML]9AFF99[HTML]009901 Para
[HTML]9AFF99 $V_{in} = 12V_{DC}$	[HTML]9AFF99 $V_{ss} = 50V_{DC}$
[HTML]9AFF99 $I_{out} = 2Amp$	[HTML]9AFF996
[HTML]9AFF99	[HTML]9AFF99

El inversor utilizado es una tarjeta comercial Keyes L298 que se muestra en la figura 4.2, y como se explicó anteriormente un motor de corriente continua va girar en función de voltaje aplicado entre sus terminales y esto se logra activando las terminales de control de la tarjeta del circuito inversor en las combinaciones que se requieran, estas señales se generan en el FPGA y se hacen llegar a través de los optoacopladores.

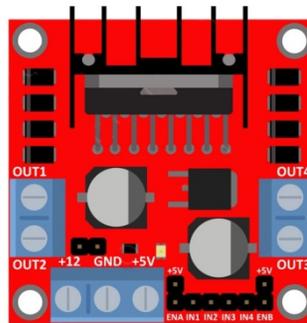


FIGURA 4.2: Inversor Keyes L298

La señal que se hace llegar al inversor es una señal PWM, en donde lo que se hace es variar el ciclo de trabajo, como la velocidad del motor depende de la potencia y esta a su vez del voltaje y corriente aplicado por lo que para variar la potencia modifica la potencia media con la que se alimenta a los motores y para esto lo adecuado es variar el voltaje, ya que por razones de diseño hay que mantener la corriente constante hasta donde esto sea posible, por tanto, el

control de velocidad de un motor de corriente directa se basa en el control de voltaje promedio aplicado.

## 4.2. Tarjeta FPGA Nexys 2

La tarjeta Nexys, es una tarjeta completa y esta lista para usarse en la plataforma de desarrollo de Xilinx Spartan 3E FPGA, y el software que se utiliza es el ISE Web Pack de Xilinx y todas sus características son descritas en el manual de referencia. [Reference Manual Nexys]

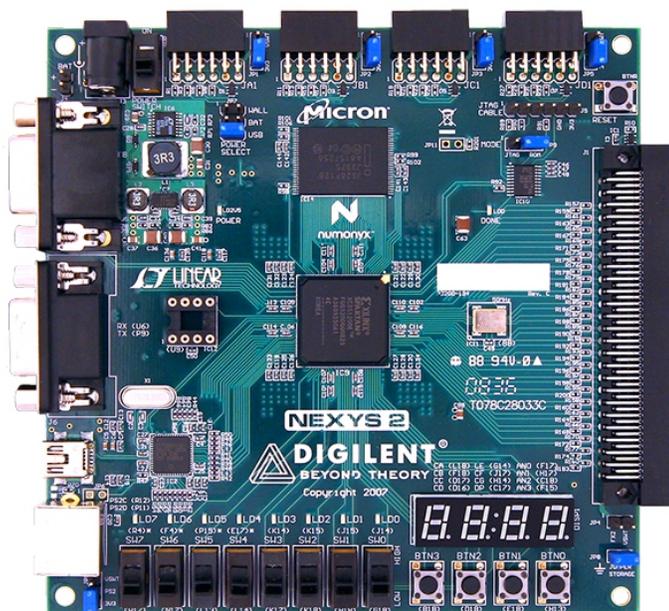


FIGURA 4.3: Tarjeta Nexys 2

## 4.3. Fabricación de tarjeta con optoacopladores

## 4.4. Software de control

### 4.4.1. Diseño de una memoria ROM con valores de la función Seno

Para los propósitos de este trabajo se requiere una memoria de sólo lectura y sin señal de reloj, lo que se está formando es una pila de valores pre-almacenados los cuales se leen cuando se ingresa su dirección de almacenamiento. Cuando se diseña una ROM, los registros no son inferidos, porque no existe una asignación de una señal a otra en la transición de estado, y son compuertas lógicas las que forman una LUT (LookUp Table).

```

1: --Descripción de la LookUp Table para la función Seno
2: library IEEE;
3: use IEEE.STD_LOGIC_1164.ALL;
4:
5: entity seno is
6:     GENERIC (bits: INTEGER :=10; --n mero de bits por palabra
7:             words: INTEGER :=8); -- n mero de palabras en la
           memoria
8:     PORT ( angulo: IN INTEGER RANGE 0 TO words-1;
9:           valor_seno: OUT STD_LOGIC_VECTOR (bits-1 DOWNT0 0));
10: end seno;
11:
12: architecture seno of seno is
13:     TYPE vector_array IS ARRAY (0 TO words-1) OF
14:     STD_LOGIC_VECTOR (bits-1 DOWNT0 0);
15:     CONSTANT memory: vector_array := (
16:         "0000000000", --seno de 0 =0.000
17:         "0101111111", --seno de pi/8 =0.383
18:         "1011000011", --seno de 2pi/8=0.707
19:         "1110011100", --seno de 3pi/8=0.924
20:         "1111101000", --seno de 4pi/8=1.000
21:         "1110011100", --seno de 5pi/8=0.924
22:         "1011000011", --seno de 6pi/8=0.707
23:         "0101111111"); --seno de 7pi/8=0.383
24: begin
25:     valor_seno <= memory(angulo);
26: end seno;

```

Para esto cuando se declaran los valores de que se utilizaran en la tabla de senos, estos deben ser declarados como constantes dentro de la entidad. Para este trabajo se requiere una tabla adicional para cosenos la cual tiene la misma estructura y formato sólo que al ingresar un ángulo esta devuelve el valor de la función coseno.

#### 4.4.2. Obtención de la posición mediante matrices de transformación

Como

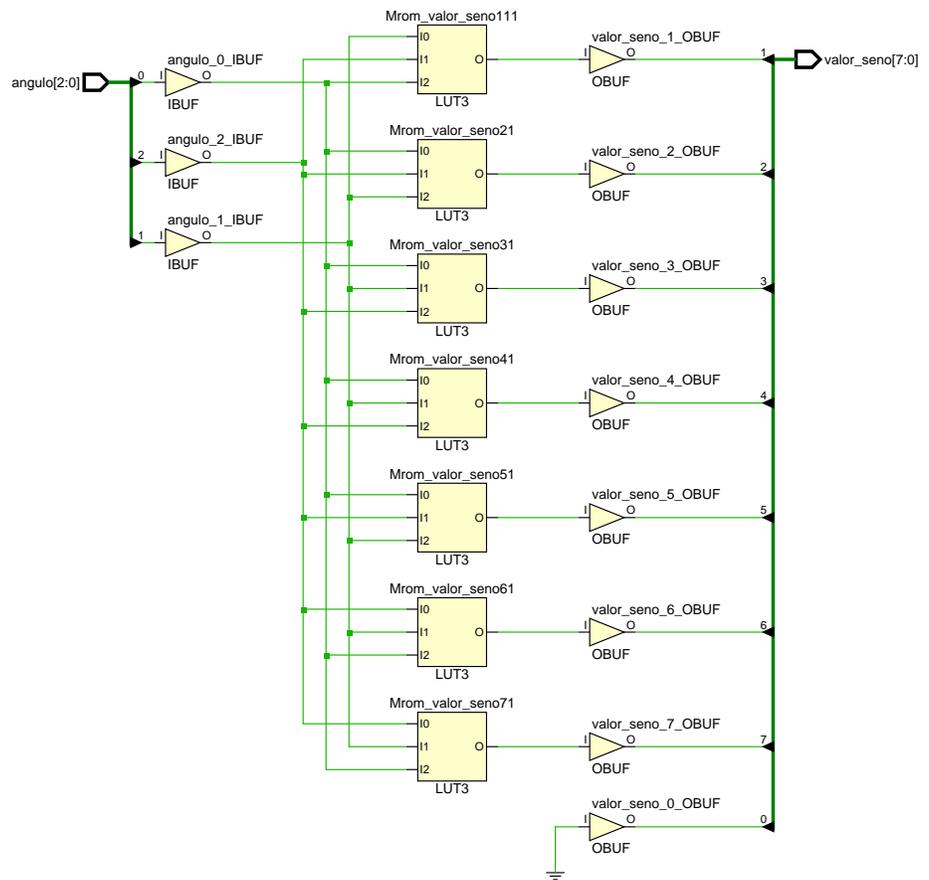


FIGURA 4.4: Construcción de la memoria ROM de la función SENO

## Capítulo 5

# Resultados y conclusiones

# Bibliografía

- [1] Ramón Pallás Areny. *Sensores y acondicionadores de señal*. Marcombo, 2004. 10, 11
- [2] Antonio Barrientos, Luis Felipe Peñin, Carlos Balaguer, and Rafael Aracil. *Fundamentos de robótica*. McGraw-Hill, Interamericana de España, 2007. 5, 11, 19, 20, 21, 22, 23, 24
- [3] Fernando Pardo Carpio and José A Boluda Grau. *VHDL: Lenguaje para síntesis y modelado de circuitos*. Ra-ma, 1999. 35, 36, 37, 38
- [4] Reyes Cortes. *MATLAB aplicado a Robótica y Mecatrónica*. Alfaomega, 2012. 1, 3, 17
- [5] Richard Dorf. *Sistemas de control moderno*, volume 10. Pearson, 2005. 9, 25, 26, 27, 28
- [6] William Kafig. *VHDL 101, Everything you need to know to get started*. Elsevier, 2011. 35, 36, 37, 38, 39
- [7] Steve Kilts. *Advanced FPGA. Design, Architecture, Implementation, and Optimization*. John Wiley Sons, 2007. 33, 34, 35
- [8] Eshed Robotec Limited. *Scorbot-ER Vplus User's Manual*, volume 3. Eshed Robotec Limited, 1998. 41, 42
- [9] Timothy J Maloney. *Electrónica industrial moderna*. Pearson Educación, 2006. 5
- [10] Albert Paul Malvino and David J Bates. *Principios de electrónica*, volume 2. McGraw-Hill, 1991. 6
- [11] David Jaime González Maxinez. *Programación de sistemas digitales con VHDL*. Grupo Editorial Patria, 2013. 36, 39
- [12] Roger Miranda. *Cinemática y dinámica de robots manipuladores*, volume 1. Alfaomega, 2016. 9, 17, 21, 22, 23, 24
- [13] Katsuhiko Ogata. *Ingeniería de control moderna*, volume 5. Pearson, 2010.
- [14] Volnei A. Pedroni. *Circuit Design with VHDL*. MIT Press, 2004. 37, 38
- [15] Daniel Saldivar Pérez Marco, Cuevas Erick. *Fundamentos de Robótica y mecatrónica con Matlab y SimuLink*. Alfaomega, 2011. 1, 17, 28

- 
- [16] Eduardo Ballester Portillo, Robert Piqué, and Robert Piqué López. *Electrónica de potencia: principios fundamentales y estructuras básicas*, volume 3. Marcombo, 2011. 6, 12, 13, 15
- [17] Muhammad Rashid. *Electrónica de potencia: circuitos, dispositivos y aplicaciones*. Pearson Educación, 2004. 5, 6, 12, 13, 14, 15, 16
- [18] Fernando Reyes Cortes. *Robótica: control de robots manipuladores*. Alfaomega, 2011. 1, 2, 10, 11
- [19] Fernando Reyes Cortés, JAIME Cid Monjaraz, and E Vargas Soto. *Mecánica, Control y Automatización*. 2013. 5, 7, 9