

UNIVERSIDAD NACIONAL DEL ALTIPLANO

FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA



**“DESARROLLO E IMPLEMENTACIÓN DE UN MÓDULO DE
ENTRENAMIENTO DE BRAZO ROBÓTICO DE CINCO GRADOS
DE LIBERTAD MONITORIZADO MEDIANTE REDES ZIGBEE”**

TESIS

PRESENTADO POR:

VICTOR RAUL QUISPE RONCALLA

RICARDO GERMAN ESCOBEDO GIL

**PARA OPTAR POR EL TÍTULO PROFESIONAL DE:
INGENIERO ELECTRÓNICO**

PUNO - PERÚ

2014

UNIVERSIDAD NACIONAL DEL ALTIPLANO

FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA

“DESARROLLO E IMPLEMENTACIÓN DE UN MÓDULO DE ENTRENAMIENTO DE
BRAZO ROBÓTICO DE CINCO GRADOS DE LIBERTAD MONITORIZADO
MEDIANTE REDES ZIGBEE”

TESIS PRESENTADA POR:

VICTOR RAUL QUISPE RONCALLA

RICARDO GERMAN ESCOBEDO GIL

PARA OPTAR POR EL TÍTULO PROFESIONAL DE:

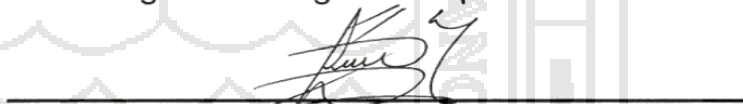
INGENIERO ELECTRÓNICO

APROBADA POR EL JURADO REVISOR CONFORMADO POR:

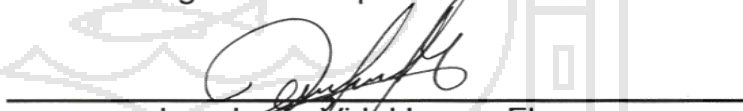
Presidente :


Mg. Eudes Rigoberto Apaza Estaño

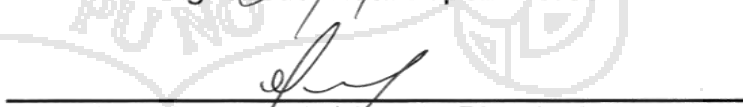
Primer miembro :


Ing. Luis Enrique Baca Wiese

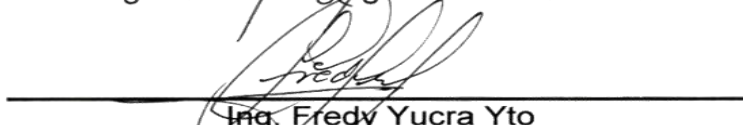
Segundo miembro :


Ing. Jesús Vidal Lopez Flores

Director de tesis :


Ing. Ferdinand Edgardo Pineda Ancco

Asesor de tesis :


Ing. Fredy Yuca YtoPUNO - PERÚ
2014

ÁREA: Automatización e instrumentación

TEMA: Robótica

AGRADECIMIENTOS

Agradecemos en primer lugar a Dios quien nos dio la vida y la ha llenado de bendiciones en todo momento, a Él que con su infinito amor nos ha dado la sabiduría suficiente para culminar nuestra carrera universitaria.

Queremos expresar nuestro más sincero agradecimiento, reconocimiento y cariño a nuestros padres por todo el esfuerzo que hicieron para dar osuna profesión y hacer de nosotros, personas de bien, gracias por los sacrificios y la paciencia que demostraron todos estos años; gracias a ustedes hemos llegado a donde estamos.

Gracias a nuestros hermanos y hermanas quienes han sido nuestros amigos fieles y sinceros, en los que hemos podido confiar y apoyarnos para seguir adelante.

Gracias a todas aquellas personas que de una u otra forma nos ayudaron a crecer como personas y como profesionales.

Agradecemos también de manera especial a nuestro director de tesis quién con sus conocimientos y apoyo supo guiar el desarrollo de la presente tesis desde el inicio hasta su culminación.

De igual manera agradecer a todos esos docentes que fungieron de guías y amigos, durante este tiempo dentro de la universidad, a aquellos amigos y compañeros de salón, de desveladas y de travesuras en la EPIE.

“Ahora podemos decir que todo lo que somos es gracias a todos ustedes”



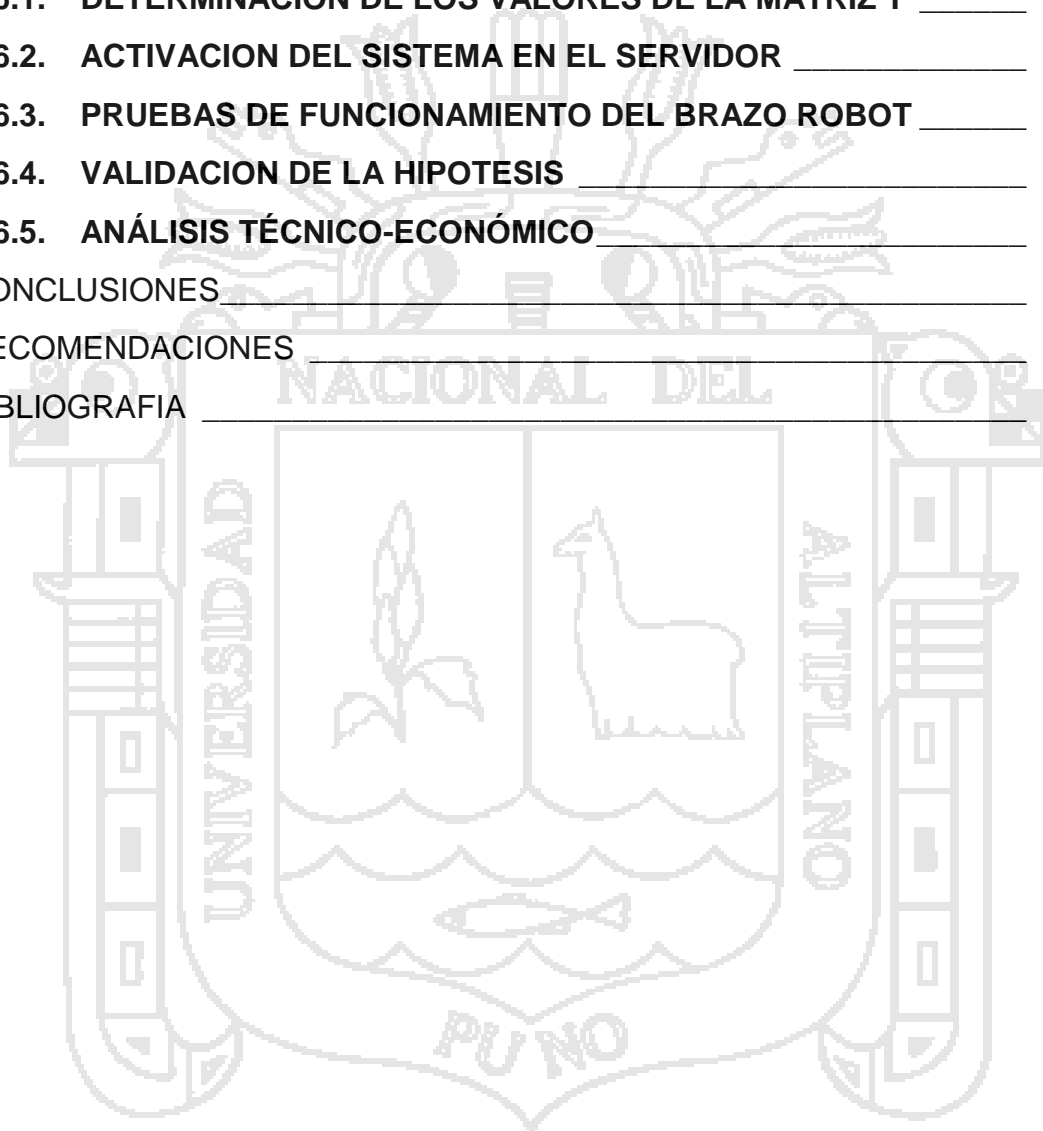
Dedicado a nuestras familias, amigos y sobre todo a Dios.

INDICE

RESUMEN _____	11
ABSTRACT _____	12
INTRODUCCIÓN _____	13
CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES Y OBJETIVOS DE LA INVESTIGACION _____	15
1.1. PLANTEAMIENTO DEL PROBLEMA. _____	16
1.2. ANTECEDENTES DE LA INVESTIGACION. _____	17
1.3. OBJETIVOS DE LA INVESTIGACION. _____	22
Objetivo general: _____	22
Objetivos específicos: _____	22
CAPÍTULO II: MARCO TEORICO, MARCO CONCEPTUAL E HIPOTESIS DE LA INVESTIGACION _____	24
2.1. MARCO TEORICO _____	25
2.2. MARCO CONCEPTUAL _____	27
2.2.1. Robótica _____	27
a) Estructura del robot manipulador _____	28
b) Partes de los robots _____	29
c) Morfología de los robots _____	30
d) Tipos de articulaciones _____	30
e) Estructuras básicas _____	31
f) Localización espacial _____	33
<input type="checkbox"/> Representaciones de la posición _____	34
<input type="checkbox"/> Sistema cartesiano de referencia. _____	35
<input type="checkbox"/> Coordenadas cartesianas. _____	36
<input type="checkbox"/> Coordenadas polares y cilíndricas. _____	36
<input type="checkbox"/> Coordenadas esféricas. _____	37
g) Representación de la orientación _____	38
<input type="checkbox"/> Matrices de rotación _____	40
h) La cinemática _____	42
<input type="checkbox"/> Matrices de transformación homogéneas _____	43
<input type="checkbox"/> Composición de matrices de transformación homogéneas _____	45
o Submatriz de rotación _____	45
o Submatriz de traslación _____	47
<input type="checkbox"/> Representación de las Matrices de Rotación y Traslación según el Modelo de Matriz de Transformación Homogénea _____	48
<input type="checkbox"/> Localización espacial _____	49
<input type="checkbox"/> Desarrollo de la cinemática directa _____	49
i) Modelo cinemático directo de robots manipuladores _____	50
<input type="checkbox"/> Modelo cinemático directo del manipulador _____	50

□	Modelo de un robot manipulador con n articulaciones	51
□	Resolución del problema cinemático directo mediante matrices de transformación homogénea.	52
□	Algoritmo de Denavit-Hartenberg para la obtención del modelo cinemático directo	54
2.2.2.	Redes inalámbricas	58
a)	Espectro electromagnético	58
b)	Protocolos de estandarización	60
c)	Ventajas de las redes inalámbricas	62
d)	Desventajas de las redes inalámbricas	63
□	Módulos XBEE	63
o	Redes ZIGBEE	64
o	Clasificación de nodos ZIGBEE	65
o	Protocolo ZigBee	67
o	Estrategias de conexión de los dispositivos en un red ZigBee	68
2.3.	HIPOTESIS DE LA INVESTIGACION	70
CAPÍTULO III: METODO DE LA INVESTIGACION		71
3.1.	VARIABLES DE LA INVESTIGACION	72
a)	Variable independiente:	72
b)	Variable dependiente:	72
3.2.	METODOLOGIA DE INVESTIGACION	72
3.3.	TIPO DE INVESTIGACION	72
3.4.	NIVEL DE INVESTIGACION	72
3.5.	FUENTES DE RECOLECCION DE DATOS	73
3.6.	ANALISIS E INTERPRETACION DE DATOS	73
CAPÍTULO IV: CARACTERIZACION DEL AREA DE INVESTIGACION		74
4.1.	UBICACIÓN GEOGRAFICA DE LA PROVINCIA DE PUNO	75
CAPÍTULO V: IMPLEMENTACION DEL MODULO DE ENTRENAMIENTO DE BRAZO ROBOTICO		77
5.1.	CIRCUITO DE CONTROL	79
5.1.1.	MICROCONTROLADOR PIC16F877A	79
5.1.2.	LÓGICA DE POSICIONAMIENTO	80
5.1.3.	CONTROL MANUAL	81
5.1.4.	PUERTO SERIE Y NIVELES DE TENSION	84
5.1.5.	INDICADORES LED	85
5.1.6.	CONECTORES	86
5.2.	MONTAJE DEL PROTOTIPO	87

5.3. MICROPROGRAMA	96
5.3.1. HERRAMIENTAS DE PROGRAMACIÓN	96
5.3.2. PROGRAMACIÓN MODO MANUAL	98
5.4. APLICACIÓN EN LABVIEW	102
5.4.1. LABVIEW PANEL FRONTAL	103
5.4.2. LABVIEW DIAGRAMA DE BLOQUES	106
CAPÍTULO VI: EXPOSICION Y ANALISIS DE LOS RESULTADOS	119
6.1. DETERMINACIÓN DE LOS VALORES DE LA MATRIZ T	120
6.2. ACTIVACION DEL SISTEMA EN EL SERVIDOR	122
6.3. PRUEBAS DE FUNCIONAMIENTO DEL BRAZO ROBOT	123
6.4. VALIDACION DE LA HIPOTESIS	124
6.5. ANÁLISIS TÉCNICO-ECONÓMICO	125
CONCLUSIONES	130
RECOMENDACIONES	131
BIBLIOGRAFIA	132



ÍNDICE DE FIGURAS

<i>Figura 1 Sistema de operación manual mediante instrumentos</i>	17
<i>Figura 2 Tableros de control y medida</i>	18
<i>Figura 3 Esquema de brazo robótico simulado en Autodesk Inventor</i>	26
<i>Figura 4 Partes de la estructura del brazo robótico.</i>	30
<i>Figura 5 Articulaciones con diferentes grados de libertad.</i>	31
<i>Figura 6 Configuraciones más usadas en robots manipuladores.</i>	32
<i>Figura 7 Sistema de coordenadas Vectorial</i>	35
<i>Figura 8 Sistema de coordenadas cartesianas</i>	36
<i>Figura 9 Sistema de coordenadas polares</i>	37
<i>Figura 10 Representación Coordenadas cilíndricas.</i>	38
<i>Figura 11 Ubicación efector final mediante un sistema de referencia</i>	39
<i>Figura 12 Sistema de coordenadas rotadas uno respecto al otro</i>	40
<i>Figura 13 Sistema de coordenadas solidarios.</i>	42
<i>Figura 14 Sistemas de coordenadas coincidentes</i>	45
<i>Figura 15 Representación gráfica y matricial de un giro α alrededor del eje X.</i>	47
<i>Figura 16 Representación gráfica y matricial de un giro β alrededor del eje Y.</i>	47
<i>Figura 17 Representación gráfica y matricial de un giro γ alrededor del eje z.</i>	47
<i>Figura 18 Representación gráfica y matricial de la submatriz de rotación.</i>	48
<i>Figura 19 Matriz de Transformación Homogénea representando una traslación espacial.</i>	49
<i>Figura 20. Relación cinemática inversa y directa.</i>	50
<i>Figura 21 Sistema de coordenadas para localizar un objeto.</i>	51
<i>Figura 22 Robot planar de 2 grados de libertad.</i>	53
<i>Figura 23 Representación parámetros de D_H</i>	56
<i>Figura 24 Robot cilíndrico</i>	57
<i>Figura 25 Espectro electromagnético</i>	59
<i>Figura 26 Red ZigBee, de acuerdo a su papel en la red</i>	66
<i>Figura 27 Mapa distrital de Puno</i>	75
<i>Figura 28 Croquis de la Universidad Nacional del Altiplano - Puno</i>	75
<i>Figura 29 E/S PIC 16F877A</i>	79
<i>Figura 30 Control de pulsos</i>	81
<i>Figura 31 Jostick con pulsadores</i>	82
<i>Figura 32 Circuito de control de la Pinza.</i>	82
<i>Figura 33 Adaptador USB XBee</i>	84
<i>Figura 34 MAX232.</i>	85
<i>Figura 35 LEDs indicadores</i>	86
<i>Figura 36 Circuitería final</i>	87
<i>Figura 37 Visión global de la parte mecánica.</i>	88
<i>Figura 38 Imagen de la base (Parte A).</i>	88
<i>Figura 39 Interior de la Parte A.</i>	89
<i>Figura 40 Base (zona inferior de la parte B).</i>	90
<i>Figura 41 Soporte multifuncional para servomotores.</i>	90

<i>Figura 42</i>	<i>Proceso de montaje del servomotor 5.</i>	91
<i>Figura 43</i>	<i>a) Soporte en largo para servomotor b) Montaje servomotor 5.</i>	91
<i>Figura 44</i>	<i>Tornillería para fijar soporte multifuncional y soporte.</i>	92
<i>Figura 45</i>	<i>Soportes largos acoplados.</i>	92
<i>Figura 46</i>	<i>Montaje del servomotor 4.</i>	93
<i>Figura 47</i>	<i>Soporte en C adosado a soporte en Largo.</i>	93
<i>Figura 48</i>	<i>Extremo de soporte en C con bloques multifuncionales superpuestos.</i>	94
<i>Figura 49</i>	<i>Pinza manipuladora</i>	94
<i>Figura 50</i>	<i>Etapas de montaje: (a) Desarrollo, (b) simulación e (c) implementación.</i>	95
<i>Figura 51</i>	<i>Aspecto del MPLAB X.</i>	97
<i>Figura 52</i>	<i>Vistas de PicKit2 clone.</i>	97
<i>Figura 53</i>	<i>Conexionado Correcto PicKit2 clone-Microcontrolador.</i>	98
<i>Figura 54</i>	<i>Cabecera del Microprograma.</i>	99
<i>Figura 55</i>	<i>Bits de configuración.</i>	100
<i>Figura 56</i>	<i>Asignación de patillas de entrada y salida.</i>	102
<i>Figura 57</i>	<i>Definición de variables y constantes.</i>	102
<i>Figura 58</i>	<i>Panel Frontal del programa en LabVIEW.</i>	103
<i>Figura 59</i>	<i>Selección del puerto.</i>	104
<i>Figura 60</i>	<i>Pestaña de acciones.</i>	104
<i>Figura 61</i>	<i>Pulsador de carga de secuencia.</i>	105
<i>Figura 62</i>	<i>Tabla de coordenadas.</i>	105
<i>Figura 63</i>	<i>Barra de posicionamiento.</i>	105
<i>Figura 64</i>	<i>Panel de información general.</i>	106
<i>Figura 65</i>	<i>Esquema general del programa.</i>	107
<i>Figura 66</i>	<i>Variables para el panel frontal.</i>	107
<i>Figura 67</i>	<i>Configuración del puerto serie en LabVIEW.</i>	108
<i>Figura 68</i>	<i>Bloque A del diagrama de bloques.</i>	109
<i>Figura 69</i>	<i>Elementos iniciales del Bloque B.</i>	110
<i>Figura 70</i>	<i>Bloque B.</i>	111
<i>Figura 71</i>	<i>Primera parte del bucle “while” del Bloque B.</i>	112
<i>Figura 72</i>	<i>Contenido del primer bucle “for” del Bloque B.</i>	113
<i>Figura 73</i>	<i>Contenido del segundo bucle “for” del Bloque B.</i>	114
<i>Figura 74</i>	<i>Estructura “case” del Bloque B.</i>	115
<i>Figura 75</i>	<i>Bloque de ayuda.</i>	116
<i>Figura 76</i>	<i>Bloque “Ord o Man?”</i>	116
<i>Figura 77</i>	<i>Bloque “Escribe en Puerto Serie”.</i>	117
<i>Figura 78</i>	<i>Bloque “PIC”.</i>	117
<i>Figura 79</i>	<i>Bloque “Lectura Byte Control”.</i>	118
<i>Figura 80</i>	<i>Análisis de los cinco grados de libertad para determinar valores de matriz homogénea T</i>	120

ÍNDICE DE TABLAS

<i>Tabla 1 Clasificación cronológica de manipuladores</i> _____	22
<i>Tabla 2 Parámetros D-H para el robot cilíndrico de la figura 18.</i> _____	58
<i>Tabla 3: Error relativo promedio de posicionamiento</i> _____	124
<i>Tabla 4: Análisis técnico</i> _____	128
<i>Tabla 5: Costo de realización</i> _____	129



RESUMEN

El presente proyecto desarrolla un brazo robot que se podrá controlar en modo manual y automático, para realizar tareas repetitivas, específicamente tareas de paletización; haciendo referencias al área de robótica y de redes, para lo cual se utilizó recursos accesibles que no requieran una gran inversión, por ello se desarrolla por completo todo el sistema: La parte mecánica, la de control y la interfaz que permita el monitoreo mediante redes Zigbee. Los diseños, métodos y herramientas utilizados buscan facilitar la comprensión de su funcionamiento para la elaboración de trabajos futuros y disminuir su complejidad.

Para lograr el fin mencionado se eligió un diseño tipo articulado antropomórfico y que a su vez representa el modelo más comercial y práctico de construir, la construcción de dicho brazo robótico consta de 5 grados de libertad, cuyo control residirá en una computadora que dotará al brazo de la capacidad - mediante redes Zigbee- para realizar algunas tareas, las cuales a gran escala podrían ser idóneas para industrias.

Palabras claves: Brazo, robot, redes, inalámbricas, articulación, paletización, industrialización.

ABSTRACT

This project develops a robot arm can be controlled in manual and automatic mode, to perform repetitive tasks, specifically palletizing tasks; making reference to the area of robotics and networks, for which accessible resources that do not require a large investment was used, so the whole system is fully developed: The mechanical part, the control and interface to monitoring through networks Zigbee. Designs, methods and tools used are intended to facilitate the understanding of its operation for the development of future work and decrease their complexity.

To achieve the aim, to be elected an anthropomorphic articulated type design which in turn represents the commercial and practical model building, the construction of the robotic arm has 5 degrees of freedom, whose control will reside on a computer that will provide the arm capacity -by Zigbee networks - to perform some tasks, which a large scale could be suitable for industries.

Keywords: arm, robot, networks, wireless, articulation, palletizing, industrialization.

INTRODUCCIÓN

La automatización de los procesos industriales a través de los años ha dado lugar a un avance espectacular de la industria. Todo ello ha sido posible gracias a una serie de factores entre los que se encuentran las nuevas tecnologías en el campo mecánico, la introducción de las computadoras, la regulación de sistemas y procesos y también el hecho, en algunos casos, de ser controlados y monitoreados remotamente. Controlar un proceso a distancia, es relativamente algo nuevo, por varias razones, como el desconocimiento de esta tecnología, el costo que involucra implementar esta tecnología en las industrias, dar mayor seguridad a empleados o supervisores que trabajan en zonas de riesgo.

La incorporación de las computadoras en la producción, la automatización de los procesos industriales y la robotización han permitido el aumento de la productividad y la mejora del producto, hecho que conlleva la incorporación de robots industriales.

Dichos robots cumplen con tareas de un alto nivel de complejidad, precisión, rapidez y efectividad.

Un robot industrial es un manipulador reprogramable con funciones múltiples, diseñado para mover materiales, partes, herramientas o dispositivos especializados a través de movimientos programados variables para el desempeño de una gran diversidad de tareas. La mayoría de las industrias automatizadas cuenta con robots manipuladores para asistir o sustituir al operador en trabajos peligrosos, monótonos o de alta complejidad.

En el presente trabajo se muestran aplicaciones que involucran el control de un manipulador a distancia comenzado con una breve explicación del contenido de los capítulos que conforman este documento.

En el Capítulo I, se muestran temas referidos al planteamiento del problema, antecedentes de la investigación y los objetivos.

En el Capítulo II, se desarrolla el marco teórico, conceptual, conjuntamente a la hipótesis de la investigación.

En el Capítulo III, se contempla la metodología implementada para el diseño y desarrollo del presente proyecto de investigación, tanto a nivel perceptivo como material.

En el Capítulo IV, describe características del área de investigación, así como datos relevantes como el problema objeto de investigación.

En el Capítulo V, se detalla la implementación del módulo de entrenamiento, y los procesos que conllevan, tales como, circuitería, montaje del prototipo, desarrollo del microprograma y la aplicación en LabView.

En el Capítulo VI, se detallan y analizan los resultados obtenidos a partir de la hipótesis.

Posteriormente se resumen las recomendaciones que se proponen, así como las conclusiones a las que se llegaron, y finalmente, se presentan la bibliografía que sirvieron como soporte a esta investigación.



PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES Y OBJETIVOS DE LA INVESTIGACIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA.

La robótica es uno de los recursos más utilizados pero a la vez, uno de los más costosos aunque muchas industrias ya utilicen sistemas muy avanzados, también existen pequeñas industrias que desean mejorar su productividad y mejorar la calidad de los procesos sin tener acceso a tecnologías.

Antes de llevar a cabo el desarrollo de un sistema que ayude a agilizar, estandarizar, aumentar la productividad y disminuir residuos así como costos de producción es necesario dar respuesta a la siguiente pregunta.

¿Cómo pueden llevarse a cabo las operaciones automatizadas de elementos mecánicos que desempeñan una función dinámica durante el proceso, y garantizar la estandarización, uniformidad y calidad del proceso?

Es por dicho motivo que el impacto que ha tomado la robótica en el mundo obliga a investigar y a solucionar problemas de la vida cotidiana, los robots han demostrado ser una herramienta de alta calidad que permiten realizar un trabajo dado.

Debido a lo anterior, resulta evidente la necesidad de desarrollar un sistema que permita realizar las operaciones de soldadura de manera estandarizada, continua y con la calidad que demanda el desempeño del armamento.

El presente proyecto de tesis tiene como objeto el desarrollo e implementación de un brazo robótico de 5 grados de libertad y además de la circuitería y la programación necesaria para controlar un brazo robot remotamente con redes ZigBee, por medio de un microcontrolador, el control se puede realizar

manualmente o por el contrario controlarse remotamente desde un ordenador y mediante una interfaz desarrollada con el software LabView.

1.2. ANTECEDENTES DE LA INVESTIGACION.

En los primeros tiempos de la industrialización las plantas eran supervisadas y controladas manualmente, basándose en las indicaciones de instrumentos instalados en campo, como muestra la Figura 1 la supervisión requería que el operador estuviera en planta para llevar a cabo el control manual del proceso.



Figura 1 Sistema de operación manual mediante instrumentos

Fuente: Acedo Sánchez., Instrumentación y Control avanzado de procesos, 2006.

Desarrollos posteriores en la instrumentación, tal como sensores con posibilidad de transmitir las principales variables de proceso (temperatura, presión, nivel, caudal, etc.), así como controladores mecánicos, hidráulicos y neumáticos, contribuyeron en los años cuarenta a la automatización gradual de las plantas, la tendencia de automatización continuó durante los años cincuenta al aparecer los instrumentos electrónicos. Todo ello condujo a la centralización en salas donde se ubicaron los elementos necesarios para llevar a cabo la supervisión y control. La figura 2 muestra dos paneles con instrumentos analógicos de diferentes tamaños de acuerdo a la evolución en la tecnología.

Con la introducción de señales estándar para transmisión, tanto neumáticas (3 a 15 PSI) como electrónicas (4 a 20 mA), los distintos fabricantes hicieron compatibles los elementos que componen un lazo de control, como sensor, transmisor, controlador, convertidor, elemento final, indicador, registrador, etc. Esto simplificó el diseño, instalación operación y mantenimiento de los sistemas de control industriales.

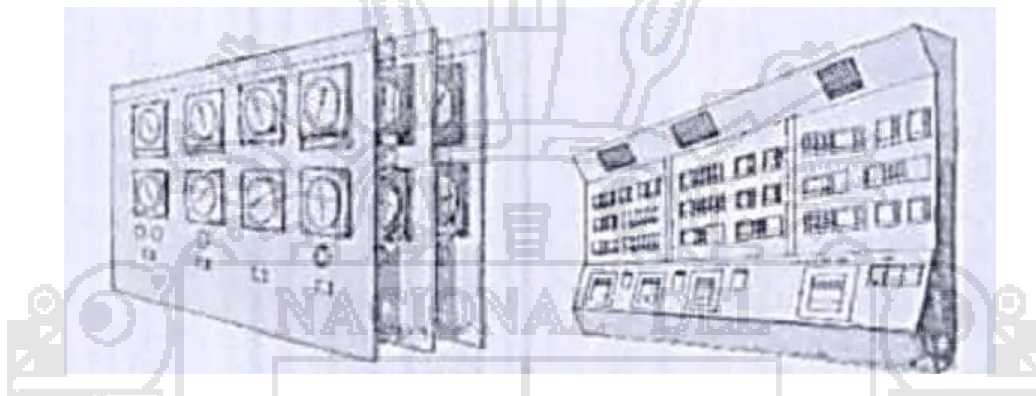


Figura 2 Tableros de control y medida

Fuente: Acedo Sánchez., Instrumentación y Control avanzado de procesos, 2006.

A medida de los años veinte se utilizó en la industria el control todo-nada, y al final de los años veinte el control proporcional. Los controladores con acciones proporcional, integral y derivativa fueron de uso común en los años treinta. Los valores de referencia (puntos de consigna o SP), podían ser fijados por el operador de planta. La necesidad de comprender los problemas inherentes al ajuste o sintonía de los controladores PID, hizo que apareciera la simulación del lazo de control. Este fue el principio de la teoría sobre sistemas de control. Para el ajuste de los parámetros de control se empezaron a utilizar las reglas de ZieglerNichols, basada en la sensibilidad del lazo de control.

Los sistemas teleoperadores se desarrollaron en los años 40 para manejar materiales radiactivos. Consistían en un par de pinzas “maestra” y “esclava” acopladas por mecanismos que permitían que la pinza “esclava”, en contacto

con el material peligroso, reprodujera los movimientos de la pinza “maestra” accionada por un operador detrás de un muro protector con ventanas apropiadas para observar la operación.

El primer teleoperador accionado por servomecanismos eléctricos se presentó en 1947. Poco después en 1948, se introdujeron servosistemas con realimentación de fuerza hacia la pinza “maestra” para permitir que el operador percibiera el esfuerzo desarrollado.

A nivel histórico se pueden distinguir cinco fases relevantes en el desarrollo de la robótica industrial:

- El laboratorio ARGONNE diseñó, en 1950, manipuladores amo-esclavo para manejar material radioactivo.
- Unimation, fundada en 1958 por Engelberger y hoy absorbida por Whestinghouse, realiza los primeros proyectos de robots a principios de la década de los sesentas de nuestro siglo, instalando el primero en 1961 y posteriormente, en 1967, un conjunto de ellos en una factoría de General Motors. Tres años después, se inicia la implantación de los robots en Europa, especialmente en el área de fabricación de automóviles. Japón comienza a implementar esta tecnología hasta 1968.
- Los laboratorios de la Universidad de Standford y del MIT (Instituto Tecnológico de Massachusetts) acometen, en 1970, la tarea de controlar un robot mediante computador.
- En el año de 1975, la aplicación del microprocesador, transforma la imagen y las características del robot, hasta entonces grande y costoso.
- A partir de 1980, el fuerte impulso en la investigación, por parte de las empresas fabricantes de robots, otros auxiliares y diversos

departamentos de Universidades de todo el mundo, sobre la informática aplicada y la experimentación de los sensores, cada vez más perfeccionados, potencian la configuración del robot inteligente capaz de adaptarse al ambiente y tomar decisiones en tiempo real, adecuarlas para cada situación.

En esta fase que dura desde 1975 hasta 1980, la conjunción de los efectos de la revolución de la Microelectrónica y la revitalización de las empresas automovilísticas, produjo un crecimiento acumulativo del parque de robots, cercano al 25%.

La evolución de los robots industriales desde sus principios ha sido vertiginosa. En poco más de 30 años las investigaciones y desarrollos sobre robótica industrial han permitido que los robots tomen posiciones en casi todas las áreas productivas y tipos de industria. En pequeñas o grandes fábricas, los robots pueden sustituir al hombre en aquellas áreas repetitivas y hostiles, adaptándose inmediatamente a los cambios de producción solicitados por la demanda variable.

Actualmente existen industrias que utilizan como herramientas de trabajo robots tipo antropomórfico, ya que conlleva una mayor eficiencia en el área de trabajo, ya sea para diferentes procesos en los cuales se necesitan, por mencionar algunos, mejoras en la fabricación de los materiales, procesos de corte, pintado, así como la precisión del manejo de objetos, manipulación de materiales tóxicos que podrían resultar peligrosos para el operario, entre otras.

Entre los antecedentes o proyectos similares en el cual se basa el presente proyecto de tesis, se puede mencionar:

- Implementación de brazos robóticos diseñados por los estudiantes de la Escuela Profesional de Ingeniería Electrónica (Jhoni Saolin Hanco Vilca – Ángel Calsina Quispe).
- Brazo robótico para realizar movimientos de piezas de ajedrez, por Giancarlo Berenz, independiente México.
- Implementación de brazos robóticos por los estudiantes de la Universidad de Medellín – Colombia.
- Desarrollo del prototipo de un brazo robótico para la manipulación de objetos, por Ezequiel Ruiz Nicolini, Instituto Balseiro-Argentina.
- Brazo robótico acoplado a una silla de ruedas que ayuda a las personas con discapacidades, desarrollado por la empresa japonesa AIST
- Robot paletizador KR desarrollado por la empresa E-Market Kplace Packaging.
- Brazo robótico CEBEK, desarrollado y comercializado por Robótica Educativa y Presencial de España.

La clasificación de los manipuladores según T. M. Knasel vendría dada de acuerdo al siguiente orden:

Generación	Nombre	Tipo de control	Grado de movilidad	Usos más frecuentes
1982	Pick and Place	Fines de carrera, aprendizaje	Ninguno	Manipulación, servicio de maquinas
1984	Servo	Servocontrol, trayectoria continua,	Desplazamiento por vía	Soldadura, pintura

		programación condicional		
1989	Ensamblado	Servos de precisión	Guía por vía	Ensamblado, desbardado
2000	Móvil	Sensores inteligentes	Patas y ruedas	Construcción, mantenimiento
2010	Especiales	Controlados con técnicas de IA	Andante, saltarín	Militar, espacial

Tabla 1 Clasificación cronológica de manipuladores

1.3. OBJETIVOS DE LA INVESTIGACION.

Objetivo general:

Desarrollar y construir el módulo de entrenamiento de un brazo robótico de cinco grados de libertad, diseñado en una escala de 1 a 100, implementado en LabView monitorizado mediante redes ZigBee, para emular tareas de paletización, tanto en modo manual como en modo automático.

Objetivos específicos:

Obtener experiencia práctica sobre la comunicación serie entre un microcontrolador y un ordenador y ampliar conocimientos sobre el campo de la robótica, para así realizar el control del modulo por medio de una interfaz applicativa de LabView

Realizar un prototipo de brazo robot de unos 50 cm de altura, que cuente con la circuitería necesaria para el control manual y automático (previamente programadas) del módulo, y que a la vez posea cinco grados de libertad y una

pinza manipuladora, cuyo control se realice a través de un microcontrolador y el cual dota de movimiento a 6 servomotores.

Desarrollar un modo aprendizaje para el robot, que le permita memorizar posiciones consecutivas colocando a priori el brazo robot en una posición inicial pre-programada, con la finalidad de que el sistema sea capaz de realizar, a pequeña escala, algunos de los movimientos que realizan los brazos robóticos industriales.





**CAPÍTULO II: MARCO TEORICO, MARCO CONCEPTUAL E
HIPOTESIS DE LA INVESTIGACION**

MARCO TEORICO, MARCO CONCEPTUAL E HIPOTESIS DE LA INVESTIGACION

2.1. MARCO TEORICO

El presente proyecto de tesis comenzó con el desarrollo de un brazo robótico simple para satisfacer la demanda de un proyecto realizable para el área de Electrónica, tanto en sistemas de control, como en comunicación de datos, de la carrera Ingeniería Electrónica.

Los avances en la tecnología han permitido desarrollar un sin número de aplicaciones en los que se destaca los sistemas de control modernos utilizados en las industrias, la medicina, el agro, en nuestra vida cotidiana y en fin en todo el entorno en el que nos desenvolvemos. Pasando del control realizado por un operador humano en una consola de control (sistemas HMI) en el sitio de la planta a sistemas controlados automáticamente mediante PLCs, autómatas robotizados hasta el control automático efectuado de forma remota por un operador ubicado a varios metros de la planta de producción.

El prototipo que se desarrolla a lo largo de este proyecto pertenece, dentro de la escala cronológica, a un robot de segunda generación, el cual se desarrollara conceptualmente en el ítem posterior Capítulo II parte 2.2.

El desarrollo resultó un proyecto integrador ya que en el mismo se utilizaron los conocimientos de distintas materias como la de electrónica de potencia; resistencia de materiales; transmisión de datos y sistemas de control, máquinas y computadoras y elementos de máquinas.

El diseño general del brazo posee cinco grados de libertad, los cuales lo hacen muy versátil, y sus partes componentes son la base, el brazo, el antebrazo y la mano. El diseño se realizó mediante el software Autodesk Inventor.

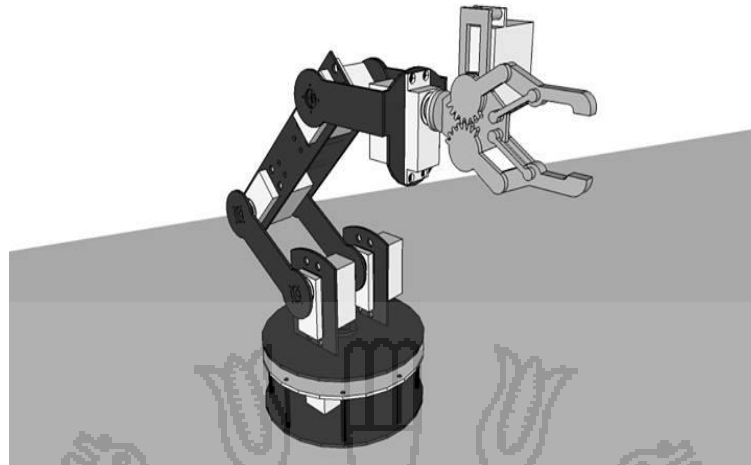


Figura 3 Esquema de brazo robótico simulado en Autodesk Inventor

Con respecto al diseño estructural, se busco obtener poca masa y buena resistencia. Para tener poca masa se utilizó para la estructura, aluminio -por ser un material ligero y maleable-, y para aumentar la resistencia se le confirió un perfil con un mayor momento de inercia en el sentido de la realización de movimientos.

Motores y transmisión, para favorecer aún más la reducción de masa se optó por colocar los servomotores distribuidos en las articulaciones; el movimiento se transmite remotamente por las redes ZigBee -cuyos protocolos son muy similares a los desarrollados en los conceptos TCP/IP-, mediante una la interfaz creada en LabView. Los motores utilizados son de corriente continua dotados de una reducción epicicloidal, con este tipo de reducción se logra obtener muy buen par en un espacio reducido.

El software se desarrolla en LabView, un programa que permite al manipulador gobernar movimientos del brazo robótico, el cual ofrece, básicamente dos modos de movimientos, manual y automático

El regulador de velocidad-posición, se constituye principalmente a un microcontrolador PIC, que es el encargado de generar la modulación por ancho

de pulsos (PWM), además se considera a los sensores y efectores como herramientas para percepción y ejecución los que se convierten en niveles lógicos TTL, que previa a la etapa de potencia actúa sobre los motores para un uso maniobrable.

Las redes ZigBee (emisor y receptor), los cuales pertenecen al estándar IEEE 802.15.4 son los encargados de la manipulación remota desde la PC hasta el modulo de entrenamiento, el que se define en 16 canales en el rango de 2.4 GHz, cada uno de ellos con un ancho de banda de 5MHz, por lo que resulta ideal para redes estáticas y escalables.

2.2. MARCO CONCEPTUAL

2.2.1. Robótica

La robótica es la ciencia encaminada a diseñar y construir aparatos y sistemas capaces de realizar tareas propias de un ser humano. Se combinan diversas disciplinas como son la mecánica, la electrónica, la informática, la inteligencia artificial y la ingeniería de control.

Los robots se pueden clasificar según su cronología y según su arquitectura. Según su cronología se clasifican los siguientes tipos:

1ª Generación: Manipuladores. Son sistemas mecánicos multifuncionales con un sencillo sistema de control, bien manual, de secuencia fija o de secuencia variable.

2ª Generación: Robots de aprendizaje. Repiten una secuencia de movimientos que ha sido ejecutada previamente por un operador humano. El modo de hacerlo es a través de un dispositivo mecánico. El operador realiza los movimientos requeridos mientras el robot le sigue y los memoriza.

3ª Generación: Robots con control sensorizado. Los robots se vuelven reprogramables, usan controladores o computadoras para analizar la información captada de su entorno mediante sensores (cabe mencionar que se desarrolla la visión artificial) y aparecen los lenguajes de programación.

4ª Generación: Robots inteligentes. Son similares a los anteriores, pero además poseen sensores que envían información a la computadora de control sobre el estado del proceso. Esto permite una toma inteligente de decisiones y el control del proceso en tiempo real.

a) Estructura del robot manipulador

Cuando se habla de la estructura del robot, se habla de la forma que debe tener el brazo del robot manipulador, forma que debe ser acorde a su aplicación. Mecánicamente, la estructura típica de un manipulador industrial se compone de un brazo y una muñeca formados por elementos con articulaciones entre ellos. Un extremo se encuentra fijo a una base soporte, mientras que el extremo final se encuentra libre y equipado con una herramienta para manipular los objetos o realizar tareas de montaje.

El manipulador se diseña con el objetivo de alcanzar una pieza dentro de su espacio de trabajo. El espacio de trabajo alcanzable se refiere al espacio dentro del cual puede desplazarse el extremo de su muñeca.

Dependiendo de la estructura del manipulador, el espacio de trabajo puede ser de forma regular o irregular. Un manipulador industrial convencional es una cadena cinemática abierta formada por un conjunto de eslabones o elementos interrelacionados mediante articulaciones o uniones, que permiten el movimiento controlado entre dos eslabones sucesivos. Cada

uno de los movimientos independientes que realiza cada una de las articulaciones se denomina grado de libertad (GDL).

Para posicionar y orientar un cuerpo en cualquier punto del espacio de trabajo, un manipulador debe de tener cinco GDL, tres para posicionamiento y dos para orientación.

El movimiento de cada articulación puede ser de rotación, desplazamiento o una combinación de ambos. Bajo esta premisa existen seis tipos diferentes de articulaciones: De rotación, prismática, cilíndrica, planar, esférica o rotula y de tornillo. Las diferentes combinaciones de articulaciones en los robots, dan lugar a que se puedan realizar diferentes configuraciones para el brazo del manipulador, características que se deben de tener en cuenta para su diseño. Las configuraciones más comunes son: Cartesiana, cilíndrica, polar o esférica, angular o rotacional o antropomórfica

b) Partes de los robots

Los robots disponen de cuerpo rígido, en el que hay eslabones con movimiento.

Los eslabones se unen entre sí mediante articulaciones, que permiten el movimiento, al tratarse de un automatismo similar a un brazo humano articulado su esquema y partes quedarían distribuidas de acuerdo a la siguiente figura:

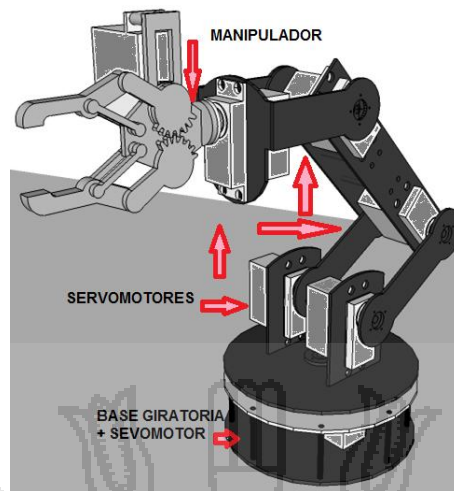


Figura 4 Partes de la estructura del brazo robótico.

c) Morfología de los robots

La morfología comprende la descripción de las características básicas de la estructura de los robots incluyendo tanto a manipuladores como móviles. En robots manipuladores se consideran los tipos de articulaciones así como las configuraciones clásicas que para este tipo de robots son empleadas. Un robot está formado por los siguientes elementos: estructura mecánica, transmisiones, sistema de accionamiento, sistema sensorial, sistema de control y elementos terminales.

Un robot manipulador es una cadena cinemática abierta formada por un conjunto de eslabones o elementos de la cadena interrelacionados mediante articulaciones o pares cinemáticos, articulaciones que permiten el movimiento relativo entre los sucesivos eslabones.

La constitución física de la mayor parte de los robots industriales guarda cierta similitud con la anatomía del brazo humano, por lo que en ocasiones, para hacer referencia a los distintos elementos que lo componen se usan términos como cuerpo, brazo, codo y muñeca.

d) Tipos de articulaciones

Cada uno de los movimientos independientes que puede realizar cada articulación con respecto a la anterior, se denomina grado de libertad (GDL). En la figura 5 se indica el número de GDL de cada tipo de articulación.

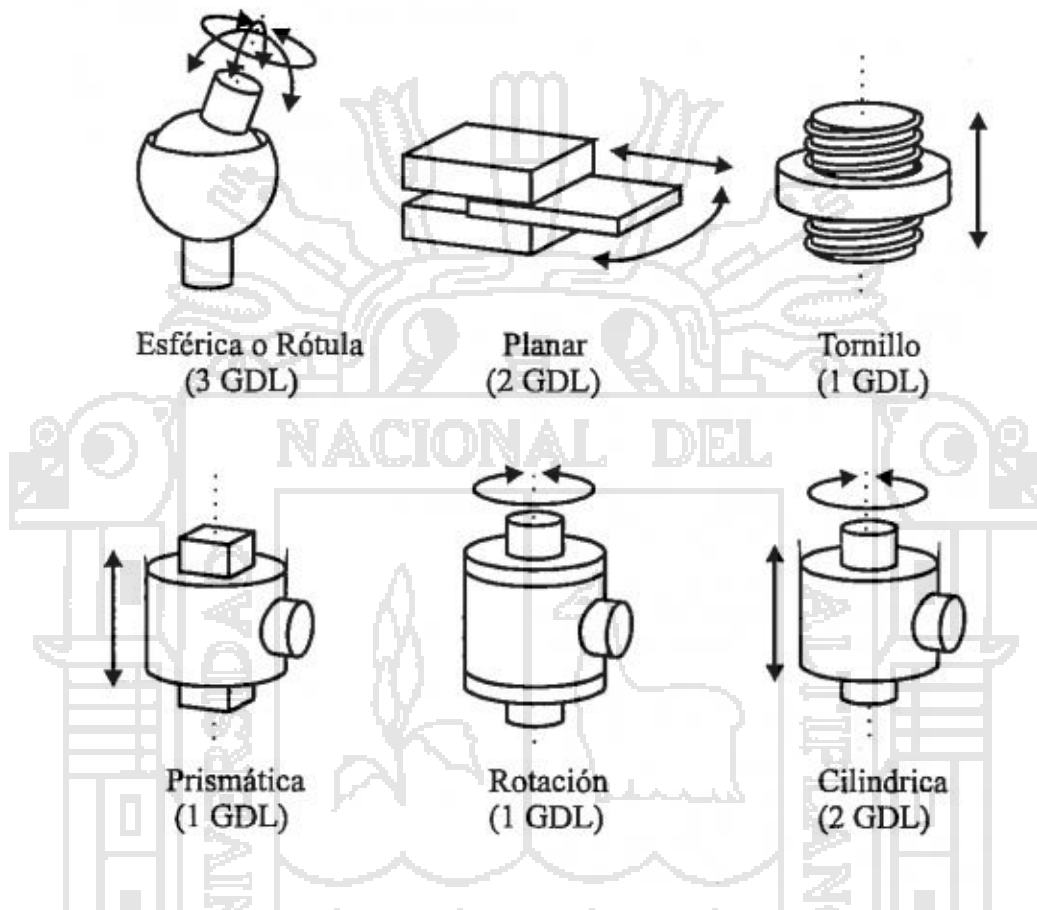


Figura 5 Articulaciones con diferentes grados de libertad.

Fuente: Barrientos., Fundamentos de Robótica., 1996.

El número de grados de libertad del robot viene dado por la suma de los grados de libertad de las articulaciones que lo componen.

e) Estructuras básicas

El empleo de diferentes combinaciones de articulaciones en un robot, da lugar a diferentes configuraciones, con características a tener en cuenta tanto en el diseño y construcción del robot como en su aplicación. Las combinaciones más frecuentes son las representadas en la figura 6 que

muestra las tres primeras articulaciones del robot, que son las más importantes a la hora de posicionar su extremo en un punto del espacio.

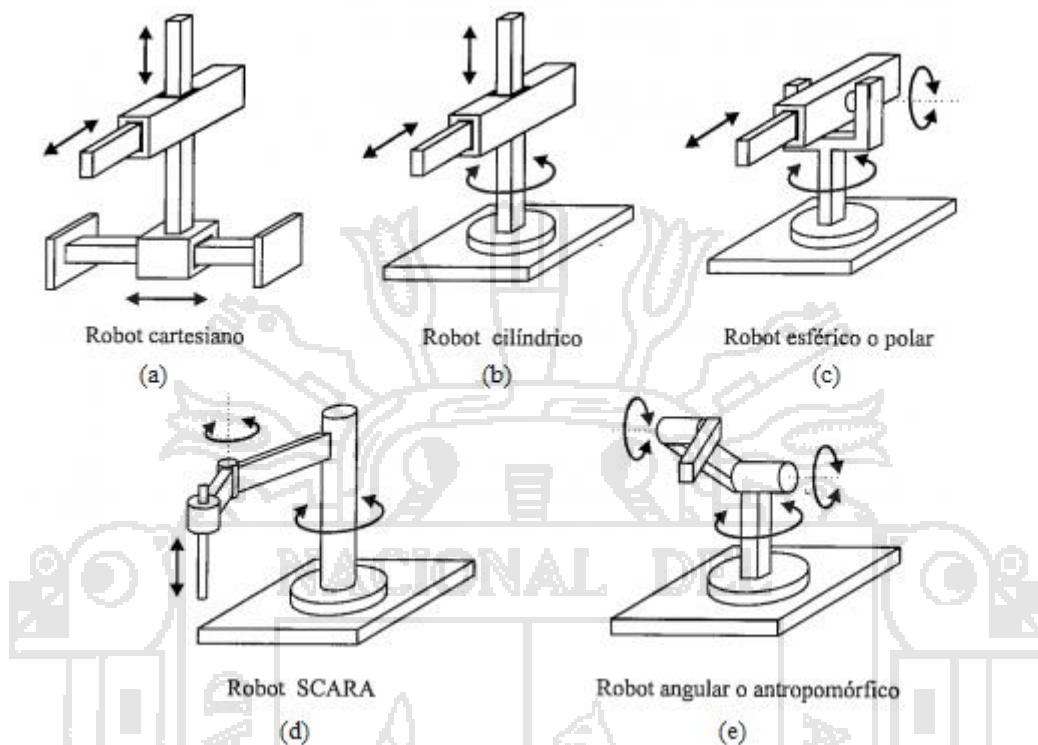


Figura 6 Configuraciones más usadas en robots manipuladores.

Fuente: Barrientos., Fundamentos de Robótica., 1996.

Puesto que para posicionar y orientar un cuerpo de cualquier manera en el espacio son necesarios seis parámetros, tres para definir la posición y tres para la orientación, si se pretende que un robot posicione y oriente su extremo (y con él, la pieza o herramienta manipulada) de cualquier modo en el espacio, se precisaran al menos seis DOF.

- **Configuración cartesiana.** Se ilustra en la figura 6.a. La configuración tiene tres articulaciones prismáticas (3D o estructura PPP). Esta configuración es bastante usual en estructuras industriales.
- **Configuración cilíndrica.** Esta configuración tiene dos articulaciones prismáticas y una de rotación (2D, 1G). La primera articulación es

normalmente de rotación (estructura PPP), como se muestra en la figura 6.b. La posición se especifica de forma natural en coordenadas cilíndricas.

- **Configuración polar o esférica.** Esta configuración se caracteriza por dos articulaciones de rotación y una prismática (2G, 1D o estructura RRP). En este caso, las variables particulares expresan la posición del extremo del tercer enlace en coordenadas polares, tal como se muestra en la figura 6.c.
- **Configuración Scara.** Esta especialmente diseñada para realizar tareas de montaje en un plano. Está constituida por dos articulaciones de rotación con respecto a dos ejes de paralelos, y una de desplazamiento en sentido perpendicular al plano figura 6.d.
- **Configuración angular o antropomórfica.** Esta configuración es una estructura con tres articulaciones de rotación (3G o RRR), tal como se muestra en la figura 6.e. La posición del extremo final se especifica de forma natural en coordenadas angulares. La estructura tiene un mejor acceso a espacios cerrados y es fácil desde el punto de vista constructivo. Es muy empleada en robots manipuladores industriales, especialmente en tareas de manipulación que tengan una cierta complejidad.

f) Localización espacial

La manipulación de piezas llevada a cabo por un robot implica el movimiento espacial de su extremo. Asimismo, para que el robot pueda recoger una pieza, es necesario conocer la posición y orientación (localización) de esta con respecto a la base del robot. Se aprecia entonces

la necesidad de contar con una serie de herramientas matemáticas que permitan especificar la posición y orientación en el espacio de piezas, herramientas y, en general, de cualquier objeto.

- **Representaciones de la posición**

En un plano el posicionamiento tiene dos grados de libertad, y por tanto la posición de un punto vendrá definida por dos componentes independientes. En el caso de un espacio tridimensional será necesario emplear tres componentes.

Un punto queda totalmente definido en el espacio a través de los datos de suposición (6), así, una vez que se establece un sistema de coordenadas, podemos ubicar cualquier punto en el universo con un vector de posición de orden 3×1 . Como es común que definamos muchos sistemas de coordenadas, además del sistema de coordenadas universal, los vectores deben etiquetarse con información que identifique en cual sistema están definidos. Así los vectores se describen con un subíndice a la izquierda que indica el sistema de coordenadas al que hacen referencia (a menos que quede claro según el contexto); por ejemplo, AP , esto significa que los componentes detienen valores numéricos que indican distancias sobre los ejes de $\{A\}$. Cada una de estas distancias sobre un eje puede definirse como el resultado de proyectar el vector sobre el eje correspondiente.

La figura 7 representa el dibujo de un sistema de coordenadas llamado $\{A\}$, con tres vectores unitarios mutuamente ortogonales con puntas sólidas.

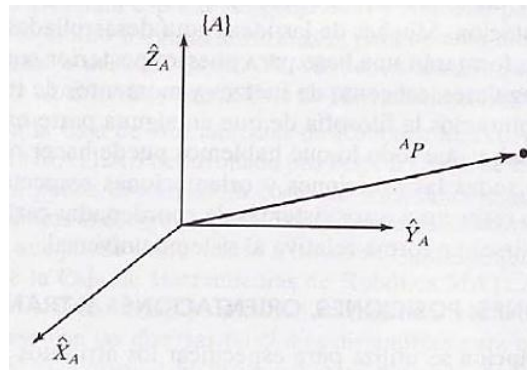


Figura 7 Sistema de coordenadas Vectorial

Fuente: Craig, J. Introducción a la robótica., 2006

Un punto se representa como un vector y puede definirse de manera equivalente como una posición en el espacio, o simplemente como un conjunto ordenado de tres números. Dados los subíndices x, y y z, los elementos individuales de un vector son:

$$AP = \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix}$$

Es decir; describe la posición de un punto en el espacio con un vector de posición.

- **Sistema cartesiano de referencia.**

El sistema de referencia OXY correspondiente queda definido por dos vectores coordenados OX y OY perpendiculares entre sí con un punto de intersección común O figura 8.a

Si se trabaja en el espacio (tres dimensiones), el sistema cartesiano OXYZ está compuesta por una terna ortonormal de vectores coordenados OX, OY y OZ, tal y como se ve en la figura 8.b. Se trata de una terna ortonormal a derechas.

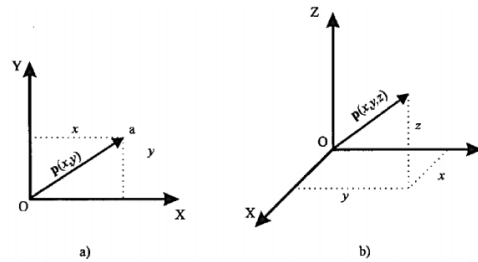


Figura 8 Sistema de coordenadas cartesianas

Fuente: Barrientos., Fundamentos de Robótica., 1996.

- **Coordenadas cartesianas.**

Si se trabaja en un plano, con su sistema coordenado OXY de referencia asociado, un punto a vendrá expresado por las componentes (x,y) correspondientes a los ejes coordenados del sistema OXY. Este punto tiene asociado un vector $p(x,y)$, que va desde el origen O del sistema OXY hasta el punto a (ver figura 8.a). Por tanto, la posición del extremo del vector p está caracterizado por las dos componentes (x,y) , denominadas coordenadas cartesianas del vector y que son las proyecciones del vector p sobre los ejes OX y OY.

En el caso de que se trabaje en tres dimensiones, un vector tiene definido con respecto al sistema de referencia OXY mediante las coordenadas correspondientes a cada uno de los ejes coordenado. En el caso de la figura 8.b, el vector p estará definido por las componentes cartesianas (x, y, z) .

- **Coordenadas polares y cilíndricas.**

Para un plano, es posible también caracterizar la localización de un punto o vector p respecto a un sistema de ejes cartesianos de referencia OXY utilizando las denominadas coordenadas polares $p(r,\Theta)$ figura 9.a. En esta representación, r representa la distancia desde el

origen o del sistema hasta el extremo del vector p , mientras que Θ es el ángulo que forma el vector p con el eje OX .

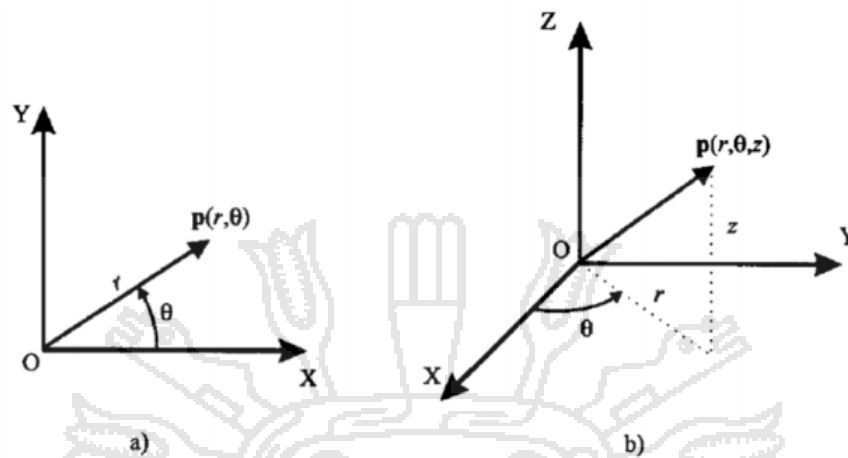


Figura 9 Sistema de coordenadas polares

Fuente: Barrientos., Fundamentos de robótica., 1996

En el caso de trabajar en tres dimensiones, un vector p podrá expresarse con respecto a un sistema de referencia $OXYZ$, mediante las coordenadas cilíndricas $p(r, \Theta, z)$ figura 9.b. Las componentes r y Θ tienen el mismo significado que en el caso de coordenadas polares, aplicado el razonamiento sobre el plano OXY , mientras que la componente z , expresa la proyección sobre el eje OZ del vector p .

- **Coordenadas esféricas.**

Utilizando el sistema de referencia $OXYZ$, el vector p tendrá como coordenadas esféricas (r, Θ, Φ) , donde la componente r es la distancia desde el origen O hasta el extremo del vector p ; la componente Θ es el ángulo formado por la proyección del vector p sobre el plano OXY con el eje OX ; y la componente Φ es el ángulo formado por el vector p con el eje OZ figura 10.

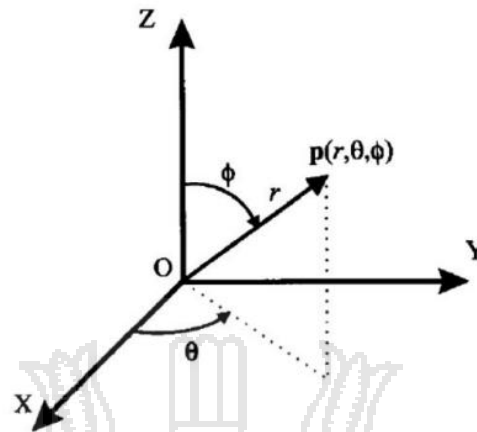


Figura 10 Representación Coordenadas cilíndricas.

Fuente: Barrientos., Fundamentos de robótica., 1996

g) Representación de la orientación

Para el caso de un sólido, es necesario además definir cuál es su orientación con respecto a un sistema de referencia. En el caso de un robot, no es suficiente con especificar cuál debe ser la posición de su extremo, sino que en general, es también necesario indicar su orientación. Una orientación en el espacio tridimensional viene definida por tres grados de libertad o tres componentes linealmente independientes. Para poder describir de forma sencilla de un objeto respecto a un sistema de referencia, es habitual asignar solidariamente al objeto un nuevo sistema, y después estudiar la relación espacial existente entre los dos sistemas. De forma general, esta relación vendrá dada por la posición y orientación del sistema asociado al objeto respecto al de referencia. Para el análisis de los distintos métodos de representar orientaciones se supondrá que ambos sistemas coinciden en el origen, y que por tanto no existe cambio alguno de posición entre ellos.

A menudo necesitamos representar no solamente un punto en el espacio, sino también describir la orientación de un cuerpo en el espacio. Por ejemplo, si el vector AP de la figura 11 ubica el punto directamente entre las puntas de los dedos de la mano de un manipulador, la ubicación completa de la mano no se especifica sino hasta que se proporciona también su orientación. Suponiendo que el manipulador tiene un número suficiente de articulaciones, la mano podría orientarse arbitrariamente y al mismo tiempo podría mantenerse el punto entre las puntas de los dedos en la misma posición.

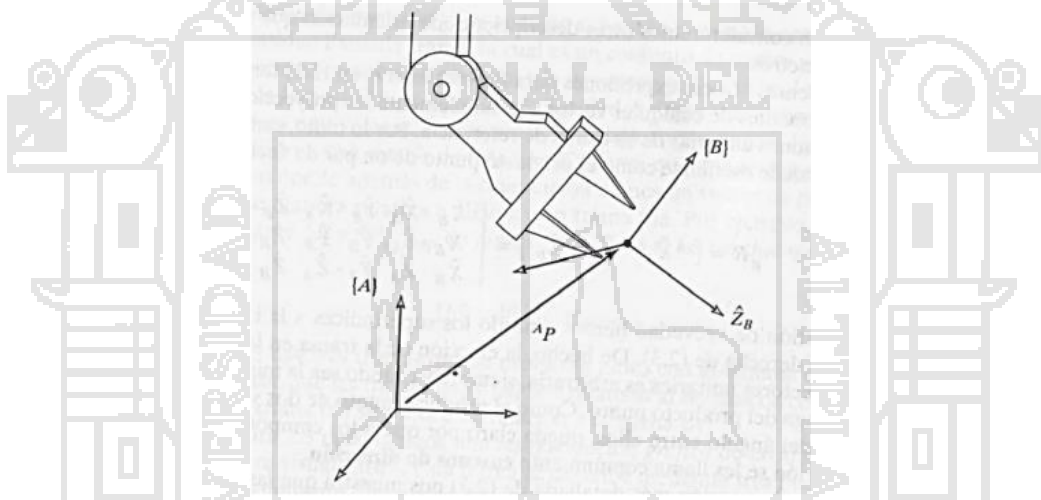


Figura 11 Ubicación efector final mediante un sistema de referencia

Fuente: Craig J., Introducción a la robótica., 2006.

En el espacio; para describir la orientación de un cuerpo, adjuntaremos un sistema de coordenadas al cuerpo y luego daremos una descripción de este sistema de coordenadas relativo al sistema de referencia. En la figura 11 se ha adjuntado el sistema de coordenadas (B) al cuerpo de una manera conocida. Ahora basta con una descripción de (B) relativo al sistema de coordenadas (A) para dar la orientación del cuerpo.

Por ende, las posiciones de los puntos se describen con vectores, y las orientaciones de los cuerpos con un sistema de coordenadas adjunto. Una manera de describir el sistema de coordenadas (B) adjunto al cuerpo es escribiendo los vectores unitarios de sus tres ejes principales en términos del sistema de coordenadas (A).

- **Matrices de rotación**

Las matrices de rotación son el método más extendido para la descripción de orientaciones, debido principalmente a la comodidad que proporciona el uso del álgebra matricial.

Supóngase que se tiene en el plano dos sistemas de referencia OXY y OUV con un mismo origen O, siendo el sistema OXY el de referencia fijo y el sistema OUV el móvil solidario al objeto (figura 12).

Los vectores unitarios de los ejes coordenados del sistema OXY son i_x, j_y , mientras que los del sistema OUV son i_u, j_v .

Un vector p del plano se puede representar en ambos sistemas como:

$$P_{xy} = [p_x, p_y]^T = p_x \cdot i_x + p_y \cdot j_y$$

$$P_{uv} = [p_u, p_v]^T = p_u \cdot i_u + p_v \cdot j_v$$

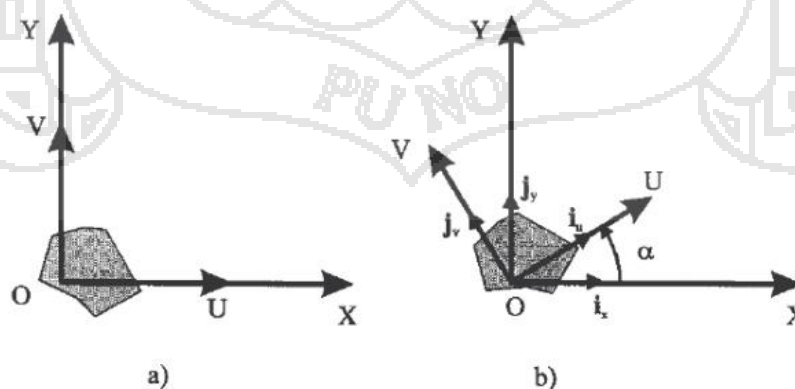


Figura 12 Sistema de coordenadas rotadas uno respecto al otro

Fuente: Craig J., Introducción a la robótica., 2006.

Realizando una sencilla serie de transformaciones se puede llegar a la siguiente equivalencia:

$$\begin{bmatrix} P_x \\ P_y \end{bmatrix} = R \begin{bmatrix} P_u \\ P_v \end{bmatrix}$$

dónde:

$$R = \begin{bmatrix} i_x i_u & i_x j_v \\ j_y i_u & j_y j_v \end{bmatrix}$$

es la llamada matriz de rotación. También recibe el nombre de matriz de cosenos directores. Es fácil de comprobar que se trata de una matriz ortonormal, tal que $R^{-1} = R^T$

En el caso de dos dimensiones, la orientación viene definida por un único parámetro independiente. Si se considera la posición relativa del sistema OUV girado un ángulo sobre el OXY figura 12.b, tras realizar los correspondientes productos escalares, la matriz R será de la forma.

$$R = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}$$

Para el caso en que $\alpha=0$, en el que los ejes coordenados de ambos sistemas coinciden, la matriz R corresponderá a la matriz unitaria.

En un espacio tridimensional, el razonamiento a seguir es similar, siendo el OXYZ el sistema de referencia fijo, y el OUVW el solidario al objeto cuya orientación se desea definir, (figura 13). Los vectores unitarios del sistema OXYZ serán $\hat{i}_x, \hat{j}_y, \hat{k}_z$ mientras que los del OUVW serán $\hat{i}_u, \hat{j}_v, \hat{k}_w$. Un vector p del espacio podrá ser referido a cualquiera de

los sistemas de la siguiente manera:

$$P_{uvw} = [p_u, p_v, p_w]^T = p_u \cdot i_u + p_v \cdot j_v + p_w \cdot k_w$$

$$P_{xyz} = [p_x, p_y, p_z]^T = p_x \cdot i_x + p_y \cdot j_y + p_z \cdot k_z$$

Y al igual que en dos dimensiones, se puede obtener la siguiente equivalencia:

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = R \begin{bmatrix} P_u \\ P_v \\ P_w \end{bmatrix}$$

Dónde:

$$R = \begin{pmatrix} i_x i_u & i_x j_v & i_x k_w \\ j_y i_u & j_y j_v & j_y k_w \\ k_z i_u & k_z j_v & k_z k_w \end{pmatrix}$$

es la matriz de rotación que define la orientación del sistema OUVW con respecto al sistema OXYZ, también recibe el nombre de matriz de cosenos directores y se trata de una matriz ortonormal, tal que la inversa de la matriz R es igual a su transpuesta $R^{-1} = R^T$.

La principal utilidad de esta matriz de rotación corresponde a la representación de la orientación de sistemas girados únicamente sobre uno de los ejes principales del sistema de referencia.

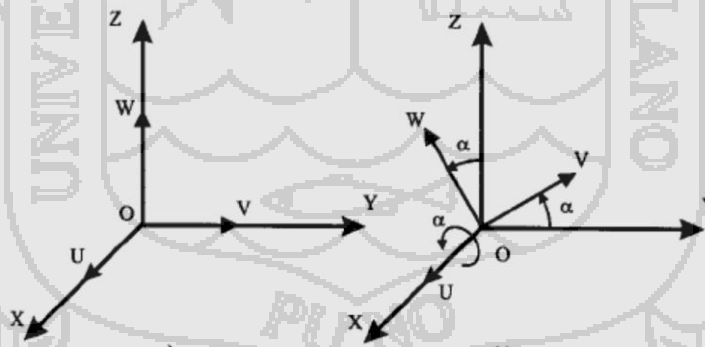


Figura 13 Sistema de coordenadas solidarios.

Fuente: Barrientos., Fundamentos de Robótica., 1996.

h) La cinemática

Una herramienta fundamental en el diseño y control de los manipuladores es sin duda la cinemática. Un robot manipulador típico, se caracteriza por una estructura serial (ciclo abierto) de ligaduras rígidas, denominada

Cadena Cinemática Abierta. Estas ligaduras están conectadas físicamente en puntos específicos denominados articulaciones, las cuales pueden tener movimientos de rotación o traslación. Cada articulación indica un grado de libertad. En la cadena cinemática abierta, uno de los extremos está sujeto a la base del robot, mientras que el otro está libre y puede ser unido a una herramienta para realizar tareas de ensamble o de manipulación de objetos. El extremo libre de la cadena se conoce con el nombre de elemento terminal del robot o endeffector.

El estudio del problema de la cinemática directa consiste en que, dado un manipulador, la posición y la orientación de su elemento terminal pueden ser determinadas por un conjunto de transformaciones de los valores de los ángulos de sus articulaciones con respecto a un sistema de coordenadas de referencia. Es así como el problema de la cinemática directa se reduce a encontrar una expresión que relacione el sistema de coordenadas del elemento terminal con el sistema de coordenadas de referencia.

Para la definición de esta expresión es posible seguir la formulación de Denavit- Hartenberg [Barrientos, 1997]. Esta formulación permite obtener una serie de expresiones que determinan la cinemática directa en base a las características del manipulador en estudio. Sin embargo, al implementar esta formulación en el ROBOT se encontraron algunas divergencias en los resultados. Se optó por analizar la formulación de Denavit-Hartenberg desde sus orígenes en las matrices de transformaciones homogéneas, y utilizar este método para obtener las expresiones que modelarían la cinemática directa del manipulador.

- **Matrices de transformación homogéneas**

Una matriz de transformación homogénea (T) es una matriz de 4x4 (4 filas, 4 columnas) que permite representar un vector en diferentes sistemas de coordenadas en el espacio.

Esta matriz de 4x4 está constituida por 4 submatrices, tal como se muestra a continuación:

$$T = \begin{bmatrix} R(3X3) & P(3X1) \\ F(1X3) & W(1X1) \end{bmatrix} = \begin{bmatrix} ROTACION & TRASLACION \\ PERSPECTIVA & ESCALADO \end{bmatrix}$$

$$T = \begin{pmatrix} R(3X3) & P(3X1) \\ F(1X3) & W(1X1) \end{pmatrix} = \begin{pmatrix} ROTACION & TRASLACION \\ PERSPECTIVA & ESCALADO \end{pmatrix}$$

La matriz de rotación representa la rotación espacial de un sistema de coordenadas. El vector de traslación, representa la traslación espacial del mismo sistema de coordenadas. Los términos de perspectiva y escalado son colocados para completar la matriz y darle una forma cuadrada. En aplicaciones robóticas el término de perspectiva es un vector de 3 componentes nulo (0,0,0) mientras que el término escalado representa un escalado global igual a la unidad. De estas observaciones, la matriz de transformación en robótica tiene una forma como esta:

$$T = \begin{pmatrix} R(3X3) & P(3X1) \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} ROTACION & TRASLACION \\ 0 & 1 \end{pmatrix}$$

Esta matriz representara la rotación (submatriz de rotación) y traslación (submatriz de traslación) de un sistema de coordenadas O'X'Y'Z respecto a un sistema de referencia OXYZ.

El uso de matrices de transformación homogéneas nos permite:

- Representar en una sola expresión la posición y orientación de un sistema de coordenadas que se ha trasladado y girado en el espacio respecto a un sistema de referencia.
- Representar un vector con coordenadas en el sistema $O'X'Y'Z'$, en el sistema $OXYZ$ o sistema de referencia.
- Rotar y trasladar un vector con respecto a un sistema de referencia fijo.
- Combinar traslaciones y rotaciones de sistemas consecutivos, multiplicando sus matrices de transformación correspondiente.
- **Composición de matrices de transformación homogéneas**

Una transformación compleja, puede descomponerse en la aplicación consecutiva de transformaciones movimientos simples (giros básicos y traslaciones).

Estos giros básicos y traslaciones son los que constituirán la matriz de transformación homogénea. Según la estructura ya mostrada, procederemos a mostrar la forma de generar las submatrices de rotación y traslación.

- **Submatriz de rotación**

A manera de explicar la obtención de esta submatriz, se empezara por representar un subsistema u,v,w que rotara, o se trasladara, con respecto a un sistema de referencia fijo X,Y,Z .



Figura 14 Sistemas de coordenadas coincidentes

La matriz representativa del subsistema u,v,w será la que se muestra en la fórmula anterior, en donde U_x representa la componente del vector u sobre el eje X, U_y la componente sobre el eje Y, y finalmente U_z la componente sobre el eje Z. De igual manera están representados los ejes v y w .

La matriz representativa del subsistema u,v,w será la que se muestra seguidamente, en donde U_x representa la componente del vector u sobre el eje X, U_y la componente sobre el eje Y, y finalmente U_z la componente sobre el eje Z. De igual manera están representados los ejes v y w .

$$T = \begin{pmatrix} U_x & V_x & W_x \\ U_y & V_y & W_y \\ U_z & V_z & W_z \end{pmatrix}$$

Simplificando la matriz representada anteriormente, para que coincida con los valores mostrados en los sistemas de coordenadas coincidentes, se obtiene la matriz representada como sigue:

$$T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

La forma que toma la matriz representada se debe a que el vector u se encuentra íntegramente sobre el eje x , no presentando componentes proyectadas sobre los otros ejes. De igual manera ocurre con los ejes v y w .

Tomando como referencia la submatriz de rotación que se acaba de revisar, se procede a representar los posibles movimientos que se pueden generar, girando alrededor de cada uno de los ejes de coordenadas independientemente. Estos movimientos se resumen como

los giros alrededor del eje X, alrededor del eje Y, y alrededor del eje Z. En las Figuras 15, 16, y 17 se representan gráfica y matricialmente los giros mencionados.

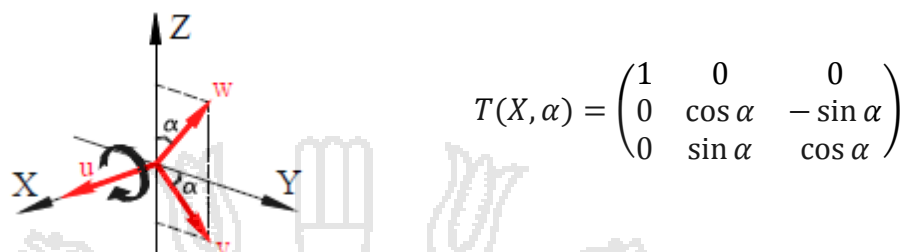


Figura 15 Representación gráfica y matricial de un giro α alrededor del eje X.

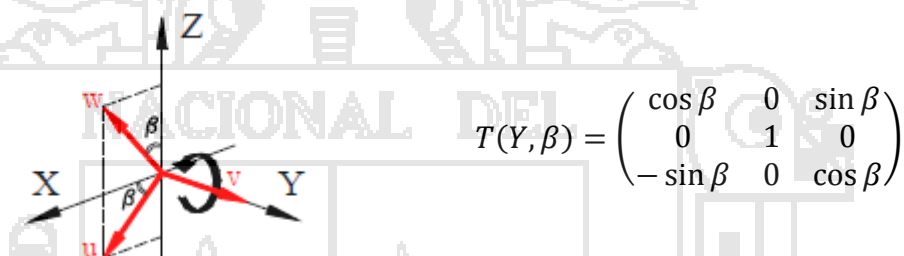


Figura 16 Representación gráfica y matricial de un giro β alrededor del eje Y.

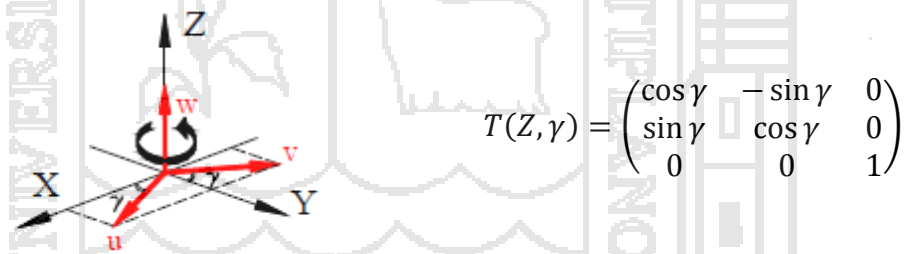


Figura 17 Representación gráfica y matricial de un giro γ alrededor del eje z.

○ **Submatriz de traslación**

Esta submatriz estará representada por el vector que une los sistemas de coordenadas relacionados, es decir, el que se traslada y el de la base.

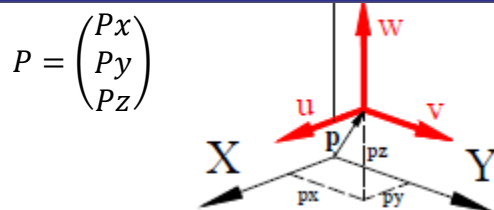


Figura 18 Representación gráfica y matricial de la submatriz de rotación.

- **Representación de las Matrices de Rotación y Traslación según el Modelo de Matriz de Transformación Homogénea**

Considerando que estos movimientos definirán la posición del sistema de coordenadas con respecto al sistema fijo, ejecutando giros y traslaciones espaciales, resultara más eficiente representar estos movimientos en una sola expresión, una matriz de transformación homogénea.

Para ello, primero se le dan forma a las matrices definidas anteriormente de tal manera que pasen a ser matrices cuadradas de 4x4. Para ello, se le asignan valores a las submatrices de traslación, escalado y perspectiva, obteniendo expresiones como las representadas en las fórmulas siguientes.

$$T(X, \alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T(Y, \beta) = \begin{pmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T(Z, \theta) = \begin{pmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$P = \begin{pmatrix} 1 & 0 & 0 & Px \\ 0 & 1 & 0 & Py \\ 0 & 0 & 1 & Pz \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Figura 19 Matriz de Transformación Homogénea representando una traslación espacial.

- **Localización espacial**

A continuación, se definen el número de movimientos necesarios para determinar la localización espacial de un sistema de coordenadas con respecto a otro, en el manipulador de 5 GDL articulado verticalmente.

Estos movimientos son tres:

- Un giro alrededor del eje z del sistema de la base $T(z, \theta)$.
- Una traslación espacial (p).
- Un giro para dar la posición final, girando alrededor de un eje diferente de z en el sistema de coordenadas que posee la posición espacial final.

Multiplicando las matrices correspondientes a estos movimientos, se genera una expresión que constituye la matriz de transformación homogénea para el sistema que se está analizando.

$$\begin{aligned} \text{Matriz de transformación homogénea} &= T(Z, \theta) \times P \times T(X, \alpha) \\ &= \begin{pmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 0 & Px \\ 0 & 1 & 0 & Py \\ 0 & 0 & 1 & Pz \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos \theta & -\sin \theta \cdot \cos \alpha & \sin \theta \cdot \sin \alpha & Px \cdot \cos \theta + (-Py \cdot \sin \theta) \\ \sin \theta & \cos \theta \cdot \cos \alpha & -\cos \theta \cdot \sin \alpha & Px \cdot \sin \theta + Py \cdot \cos \theta \\ 0 & \sin \alpha & \cos \alpha & Pz \\ 0 & 0 & 0 & 0 \end{pmatrix} \end{aligned}$$

- **Desarrollo de la cinemática directa**

Para definir la cinemática será necesario contar con un modelo que defina la apariencia del manipulador. Este modelo debe mostrar la configuración que tendrá, es decir como estarán distribuidos los grados

de libertad, la posición y giro de los actuadores, la longitud de los eslabones, etc.

i) Modelo cinemático directo de robots manipuladores

Existen dos problemas fundamentales a resolver en la cinemática del robot figura 10 el primero de ellos se conoce como el problema cinemático directo, y consiste en determinar cuál es la posición y orientación del extremo final del robot, con respecto a un sistema de coordenadas que toma como referencia, conocidos los valores de las articulaciones y los parámetros geométricos de los elementos del robot; el segundo, denominado problema cinemático inverso, resuelve la configuración que debe adoptar el robot para una posición y orientación del extremo conocidas.

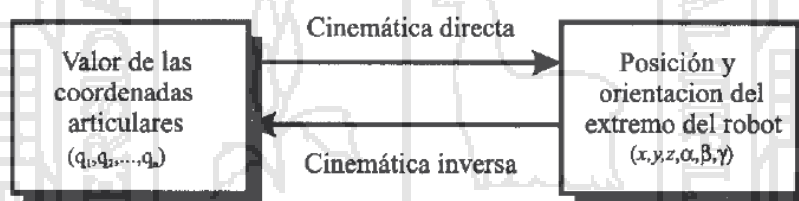


Figura 20. Relación cinemática inversa y directa.

Fuente: Barrientos., Fundamentos de robótica., 1996

Denavit y Hartenberg propusieron un método sistemático para describir y representar la geometría espacial de los elementos de una cadena cinemática, y en particular de un robot, con respecto a un sistema de referencia fijo, reduciéndose el problema cinemático directo a encontrar una matriz de transformación homogénea 4x4 que relaciona la localización espacial del extremo del robot con respecto al sistema de coordenadas de su base.

- **Modelo cinemático directo del manipulador**

Viene dado por una función que permite expresar la posición y orientación del sistema de referencia objetivo en el espacio cartesiano (operacional o de la tarea) p en términos de q :

Siendo ϕ un conjunto de funciones no lineales. Conocidas las variables articulares (entradas), se desea saber dónde está el manipulador.

Para construir el modelo directo existen dos alternativas típicas:

- Utilización de relaciones específicas del robot: Consiste en el empleo de las relaciones geométricas que puedan establecerse en el robot de que se trate.
- Método general: Se trata de emplear matrices de transformación que relacionan sistemas de referencia tal como se ha expuesto anteriormente.
- **Modelo de un robot manipulador con n articulaciones**

Un robot articulado puede describirse definiendo cuatro magnitudes asociadas a cada articulación. Una de estas magnitudes es la variable de la articulación y las restantes son parámetros fijos para cada robot figura 15.

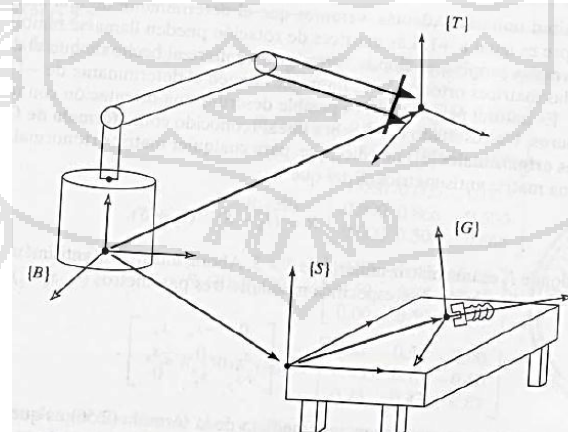


Figura 21 Sistema de coordenadas para localizar un objeto.

Fuente: Barrientos., Introducción a la robótica., 2006

La grafica anterior muestra estos parámetros que permiten definir una representación de las relaciones de traslación y rotación entre los enlaces adyacentes. A esta relación se le denomina representación de Denavit-Hartenberg.

- **Resolución del problema cinemático directo mediante matrices de transformación homogénea.**

La resolución del problema cinemático directo consiste en encontrar las relaciones que permiten conocer la localización espacial del extremo del robot a partir de los valores de sus coordenadas articulares.

La obtención de estas relaciones no es en general complicada, siendo incluso en ciertos casos (robot de pocos GDL) fácil de encontrar mediante simples consideraciones geométricas. Por ejemplo, para el caso de un robot de 2GDL figura 21, es fácil comprobar que:

Para robots de más grados de libertad puede plantearse un método sistemático basado en la utilización de las matrices de transformación homogénea.

$$\begin{aligned}
 {}^{i-1}A_i &= \begin{bmatrix} \cos\theta & -\text{sen}\theta & 0 & 0 \\ \text{sen}\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\text{sen}\alpha & 0 \\ 0 & \text{sen}\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} \cos\theta & -\text{sen}\theta & \text{sen}\alpha \text{sen}\theta & \text{a}\cos\theta \\ \text{sen}\theta & \cos\theta & -\text{sen}\alpha \cos\theta & \text{a}\text{sen}\theta \\ 0 & \text{sen}\alpha & \cos\alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

En general, un robot de n grados de libertad está formado por n eslabones unidos por n articulaciones, de forma que cada par articulación-eslabón constituye un grado de libertad. A cada eslabón se le puede asociar un sistema de referencia solidario a él y, utilizando las transformaciones homogéneas, es posible representar las rotaciones y traslaciones relativas

entre los distintos eslabones que componen el robot. Normalmente, la matriz de transformación homogénea que representa la posición y orientación relativa entre los sistemas asociados a dos eslabones consecutivos del robot se suele denominar matriz ${}^{i-1}A_i$. Así pues 0A_1 , describe la posición y orientación del sistema de referencia solidario al primer eslabón con respecto al sistema de referencia a la base, 1A_2 describe la orientación y posición del segundo eslabón respecto del primero, etc. Del mismo modo, denominando 0A_k a las matrices resultantes del producto de las matrices ${}^{i-1}A_i$ con i desde 1 hasta k , se puede representar de forma total o parcial la cadena cinemática que forma el robot. Así, por ejemplo, la posición y orientación del sistema solidario con el segundo eslabón del robot con respecto al sistema de coordenadas de la base se puede expresar mediante la matriz 0A_2 :

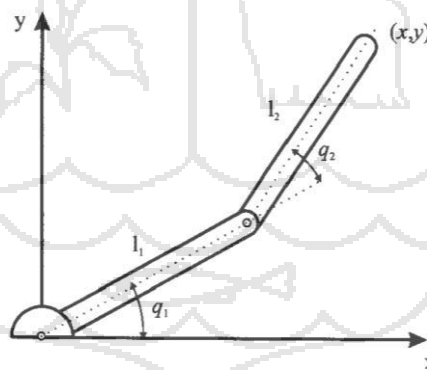


Figura 22 Robot planar de 2 grados de libertad.

Fuente: Barrientos., Introducción a la robótica., 2006

$${}^0A_2 = {}^0A_1 {}^1A_2$$

De manera análoga, la matriz representa la localización del sistema del tercer eslabón:

$${}^0A_3 = {}^0A_1 {}^1A_2 {}^2A_3$$

Cuando se consideran todos los grados de libertad, a la matriz 0A_n se le suele denominar T. Así dado un robot de seis grados de libertad, se tiene que la posición y orientación del eslabón final vendrá dada por la matriz T:

$$T = {}^0A_6 = {}^0A_1 {}^1A_2 {}^2A_3 {}^3A_4 {}^4A_5 {}^5A_6$$

Aunque para describir la relación que existe entre dos elementos contiguos se puede hacer uso de cualquier sistema de referencia ligado a cada elemento, la forma habitual que se suele utilizar en robótica es la representación de Denavit-Hartenberg (D-H).

Estas transformaciones básicas consisten en una sucesión de rotaciones y traslaciones que permiten relacionar el sistema de referencia del elemento i con el sistema del elemento i-1. Las transformaciones en cuestión son las siguientes (es importante recordar que el paso del sistema $\{S_{i-1}\}$ al $\{S_i\}$ mediante estas cuatro transformaciones está garantizado solo si los sistemas $\{S_{i-1}\}$ y $\{S_i\}$ han sido definidos de acuerdo a unas normas determinadas).

- **Algoritmo de Denavit-Hartenberg para la obtención del modelo cinemático directo**

Denavit y Hartenberg propusieron en 1955 un método matricial que permite establecer de manera sistemática un sistema de coordenadas $\{S_i\}$ ligado a cada eslabón i de una cadena articulada, pudiéndose determinar a continuación las adecuaciones cinemáticas de la cadena completa.

D-H 1. Numerar los eslabones comenzando con 1 (primer eslabón móvil de la cadena) y acabando con n (último eslabón móvil). Se enumerara como eslabón 0 a la base fija del robot.

D-H 2. Numerar cada articulación comenzando por 1 (la correspondiente al primer grado de libertad) y acabando en n .

D-H 3. Localizar el eje de cada articulación. Si esta es rotativa, el eje será su propio eje de giro. Si es prismática, será el eje a lo largo del cual se produce el desplazamiento.

D-H 4. Para i de 0 a $n-1$ situar el eje sobre el eje de la articulación $i+1$.

D-H 5. Sitúa el origen del sistema de la base S_0 en cualquier punto del eje z_0 , los ejes x_0 e y_0 se situarán del modo que formen un sistema dextrógiro con z_0 .

D-H 6. Para i de 1 a $n-1$, situar el sistema S_i en la intersección z_i del eje con la línea normal a z_{i-1} y z_i . Si ambos ejes se cortasen se situaría S_i en el punto de corte. Si fuesen paralelos S_i se situaría en la articulación $i+1$.

D-H 7. Situar x_i en la línea normal común a z_{i-1} y z_i .

D-H 8. Situar de modo que forme un sistema dextrógiro con x_i y z_i .

D-H 9. Situar el sistema $\{S_n\}$ en el extremo del robot de modo que coincida con la dirección de z_{n-1} y x_n sea normal a z_{n-1} y z_n .

D-H 10. Obtener θ_i como el ángulo que hay que girar en torno a z_{n-1} para que x_{i-1} y x_i queden paralelos.

D-H 11. Obtener d_i como la distancia medida a lo largo de, que habría que desplazar S_{i-1} para que x_i y x_{i-1} quedasen alineados.

D-H 12. Obtener α_i como la distancia medida a lo largo de (que ahora coincidiría con x_{i-1}) que habría que desplazar el nuevo $\{S_{i-1}\}$ para que su origen coincidiese con $\{S_i\}$.

D-H 13. Obtener α_i como el ángulo que habría que girar en torno a x_i (que ahora coincidiría con x_{i-1} para que el nuevo $\{S_{i-1}\}$ coincidiese totalmente con $\{S_i\}$.

D-H 14. Obtener las matrices de transformación ${}^{i-1}A_i$ definidas.

D-H 15. Obtener la matriz de transformación que relaciona el sistema de la base con el del extremo del robot $T = {}^0A_1, {}^1A_2, {}^2A_3, \dots, {}^{n-1}A_n$.

D-H 16. La matriz T define la orientación (sub matriz de rotación) y posición (submatriz de traslación) del extremo referido a la base en función de las n coordenadas articulares figura 22.

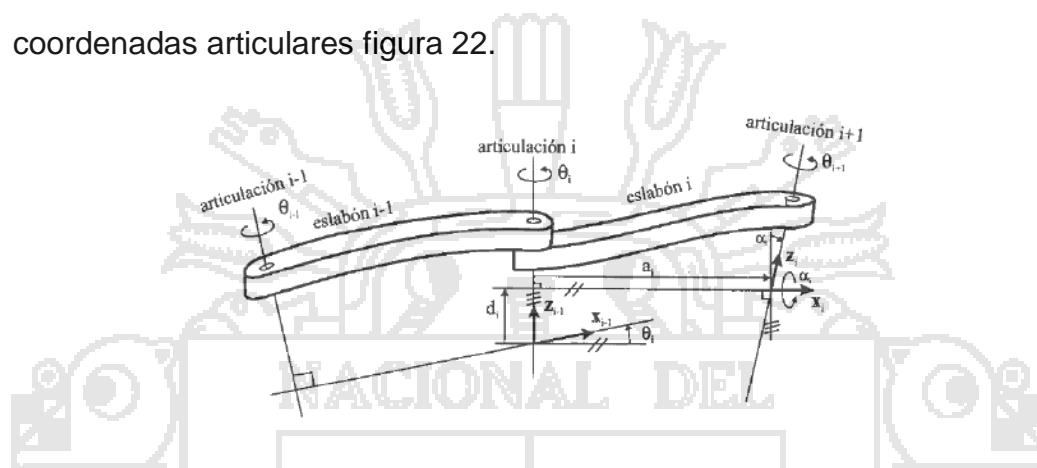


Figura 23 Representación parámetros de D_H

Fuente: Barrientos., Introducción a la robótica., 2006

Los cuatro parámetros de D-H ($\theta_i, d_i, a_i, \alpha_i$) dependen únicamente de las características geométricas de cada eslabón y de las articulaciones que le unen con el anterior y siguiente:

θ_i Es el ángulo que forman los ejes x_{i-1} y x_i medido en un plano perpendicular al eje z_{i-1} , utilizando la regla de la mano derecha. Se trata de un parámetro variable en articulaciones giratorias.

d_i Es la distancia a lo largo del eje z_{i-1} desde el origen del sistema de coordenadas $(i-1)$ ésimo hasta la intersección del eje z_{i-1} con el eje x_i , se trata de un parámetro variable en articulaciones prismáticas.

a_i Es la distancia a lo largo del eje x_i , que va desde la intersección del eje z_{i-1} con el eje hasta el origen del sistema i ésimo, en el caso de

articulaciones giratorias. En el caso de articulaciones prismáticas, se calcula como la distancia más corta entre los ejes z_{i-1} y z_i .

α_i Es el ángulo de separación del eje z_{i-1} y el eje z_i , medido en un plano perpendicular al eje x , utilizando la regla de la mano derecha.

Una vez obtenidos los parámetros D-H, el cálculo de las relaciones entre los eslabones consecutivos del robot es inmediato, ya que vienen dadas por las matrices A, que se calculan según la expresión general. Las relaciones entre eslabones no consecutivos vienen dadas por las matrices T que, como ya se comentó anteriormente, se obtienen como producto de un conjunto de matrices A. Obtenida la matriz T, esta expresará la orientación (sub matriz (3x3) de rotación) y posición (sub matriz (3x1) de traslación) del extremo del robot en función de sus coordenadas articulares, con lo que quedara resuelto el problema cinemático directo.

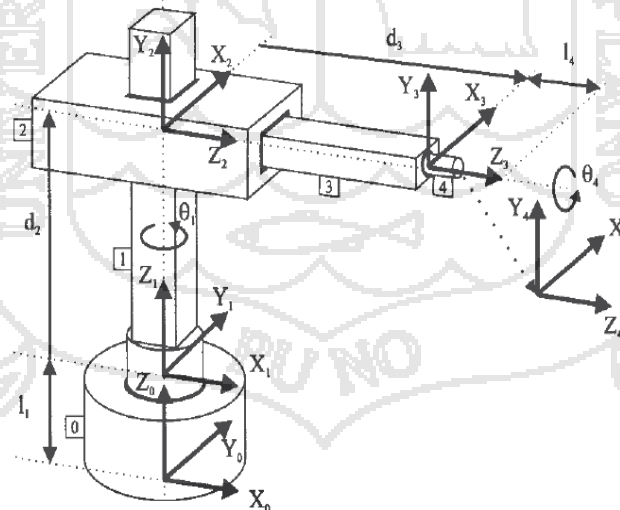


Figura 24 Robot cilíndrico

Fuente: Barrientos., Introducción a la robótica., 2006

Articulación	θ_i	d_i	a_i	α_i
1	q_1	l_1	0	0
2	90	d_2	0	90
3	0	d_3	0	0
4	q_4	l_4	0	0

Tabla 2 Parámetros D-H para el robot cilíndrico de la figura 18.

Fuente: Barrientos., Introducción a la robótica., 2006

2.2.2. Redes inalámbricas

El término red inalámbrica (Wireless network en inglés) es un término que se utiliza en informática para designar la conexión de nodos sin necesidad de una conexión física (cables), ésta se da por medio de ondas electromagnéticas. La transmisión y la recepción se realizan a través de puertos.

a) Espectro electromagnético

El espectro electromagnético es el rango de todas las radiaciones electromagnéticas posibles. El espectro de un objeto es la distribución característica de la radiación electromagnética de ese objeto.

El espectro electromagnético se extiende desde las bajas frecuencias usadas para la radio moderna (extremo de la onda larga) hasta los rayos gamma (extremo de la onda corta), que cubren longitudes de onda de entre miles de kilómetros y la fracción del tamaño de un átomo. Se piensa que el límite de la longitud de onda corta está en las cercanías de la longitud de Planck, mientras que el límite de la longitud de la onda larga es el tamaño del universo en sí, aunque en principio el espectro sea infinito y continuo.

El espectro electromagnético cubre la energía de ondas electromagnéticas que tienen longitudes de onda diferentes. Las frecuencias de 30 Hz y más bajas pueden ser producidas por ciertas nebulosas estelares y son importantes para

su estudio. Se han descubierto frecuencias tan altas como 2.9×10^{27} Hz a partir de fuentes astrofísicas.

La energía electromagnética en una longitud de onda particular λ (en el vacío) tiene una frecuencia asociada f y una energía fotónica E . Así, el espectro electromagnético puede expresarse en términos de cualquiera de estas tres variables que están relacionadas mediante ecuaciones.

De este modo, las ondas electromagnéticas de alta frecuencia tienen longitud de onda corta y energía alta; las ondas de frecuencia baja tienen longitud de onda larga y energía baja.

Las longitudes de onda de la radiación electromagnética, sin importar el medio por el que viajen, son, por lo general, citadas en términos de longitud de onda en el vacío, aunque no siempre se declara explícitamente.

Por lo general, la radiación electromagnética se clasifica por la longitud de onda; ondas de radio, microondas, infrarroja y región visible (luz, rayos ultravioleta, rayos X y rayos gamma)

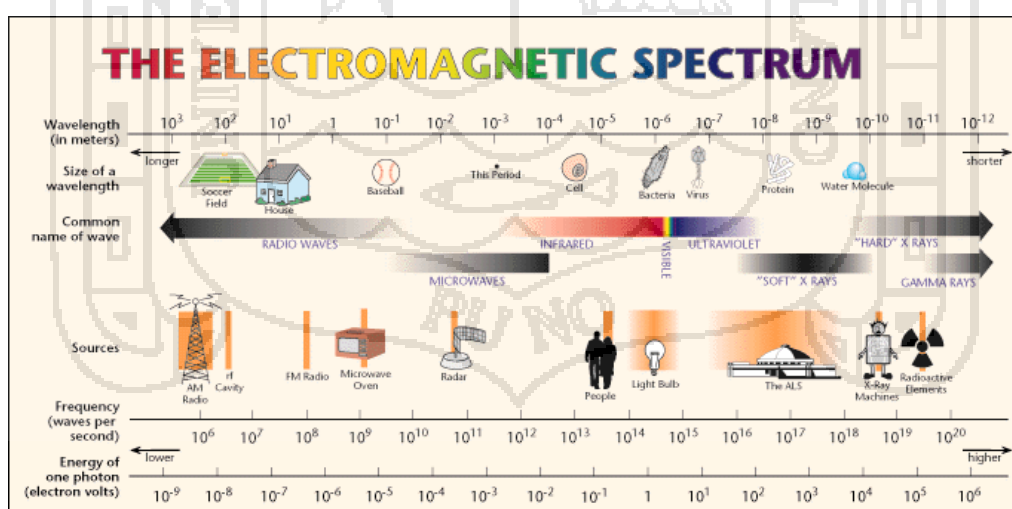


Figura 25 Espectro electromagnético

Fuente: <http://espectroelectromagnetico.blogspot.com/2007/04/el-espectro-electromagnetico.html>

b) Protocolos de estandarización

IEEE 802.11, estándar de redes inalámbricas en una red de área local o WLAN (Wireless Local Area Network) para dispositivos con certificación Wi-Fi (Wireless Fidelity), se trata de frecuencias incluidas en la banda ISM (Industrial, Scientific and Medical), bandas reservadas internacionalmente para su uso no comercial de radiofrecuencia, dentro de dicho estándar existen categorías:

IEEE 802.11a, con un rendimiento total es de 54 Mbps, este estándar provee hasta ocho canales de radio en la banda de frecuencia de 5 GHz, posee una modulación tipo OFDM.

IEEE 802.11b, con un rendimiento total es de 11 Mbps, se encuentra en la frecuencia de 2.4 GHz, posee una modulación DSSS.

IEEE 802.11g, con un rendimiento total de 54 Mbps, está en el rango de 2.4 GHz con una modulación OFDM, es compatible con dispositivos que admiten el estándar IEEE 802.11b.

IEEE 802.11n, con un rendimiento que va desde 100 Mbps a 600 Mbps, cuenta con una modulación MIMO.

IEEE 802.11ac, desde enero de 2014, tiene un rendimiento de hasta 1Gbps, dentro de la banda de 5GHz, con flujos MIMO y modulación de alta densidad 256 QAM.

IEEE 802.15.1, estándar de redes inalámbricas para redes personales o WPAN (Wireless Personal Area Network), para dispositivos con certificación Bluetooth, también incluidas en la banda ISM.

IEEE 802.15.4, estándar de redes inalámbricas para redes personales o WPAN, para módulos de radio, se puede tomar en cuenta los siguientes:

Protocolos MIWI, con tasas de transferencias bajas aproximadamente de 250Kbps, con distancias máximas de 100 metros sin obstáculos, desarrollado por Microchip Technology, proyecto de código abierto con la única limitación de la obligación de utilizar microcontroladores de Microchip Technology y su transceptor MRF24J4, capaz de formar redes punto a punto, de estrella o malla; capaz de tener 8 coordinadores como máximo por red, y cada uno de estos 127 hijos, haciendo un total de 1024 nodos por red.

Protocolos ZIGBEE, cuyo objetivo es el de aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías, es decir, cuenta con claves de seguridad de 128 bits, hasta 25 Kbps y de bajo consumo de energía.

Por considerar otros que corresponden al presente protocolo se pueden hacer mención a:

- ZIGBIT de Atmel
- ZICM* de California Eastern Laboratories
- deRFmega128-22M00 de Dresden Elektronik
- EMB-ZRF231PA, EMB-Z2530PA de Embit
- JN5139-xxx-M0x de Jennic
- VersaNode 210, VersaNode 220 de Nivis LLC.
- Pan45xx de Panasonic
- RC220x de Radiocrafts
- ZMN24xx de RF Monolithics
- MXR-EM20 de Radios Inc.
- Pixie, Pixie Lite de FlexiPanel Ltd.
- ETRXxxx de Telegesis Ltd.

- ProBee de Sena Technologies

Z357PA10 de MMB Research, entre otros cuya producción se discontinuó.

IEEE 802.16, desarrollada como alternativa de red inalámbrica al acceso de banda ancha DSL, y cable, y una forma de conectar nodos Wi-Fi en una red de área metropolitana (MAN), con un radio de acción de 50 Km., dentro de dicho estándar existen categorías:

IEEE 802.16; 10-66 GHz, modulación QAM

IEEE 802.16a; 2-11 GHz, modulación OFDM y OFDMA

IEEE 802.16b/c; Interoperable con la especificación de certificaciones.

IEEE 802.16e; añade movilidad

c) Ventajas de las redes inalámbricas

Como se vio en el ítem anterior, las redes inalámbricas están basadas en estándares internacionales comprobados a nivel de industria de transmisión de datos.

Instalación simple, ya que debe ser del tipo plug and play, tomando minutos su instalación.

Robusta y confiable, consideradas en especial aquellas que tienen alcances no mayores a 100 metros.

Escalabilidad, pues los concentradores inalámbricos pueden soportar sin problemas 60 usuarios simultáneamente, lo que permite expandir su red con efectivos de costos.

Facilidad de uso, ya que permite movilidad, lo que no se dan en redes cableadas.

Servidor web para una administración más fácil, un usuario simplificará la administración de su red inalámbrica si selecciona un punto de acceso con un servidor web incorporado.

Seguridad, si el usuario escoge una red que otorgue múltiples niveles de seguridad, encriptación y autenticación de usuarios.

Existen aplicaciones que detectan localidades, para poder tener acceso a concentradores próximos y permanecer conectado.

Costo de propiedad reducido, mientras que el costo de inversión inicial requerida para una red inalámbrica puede ser más alta que el costo en hardware de una red cableada, la inversión de toda la instalación y el costo durante el ciclo de vida pueden ser significativamente inferiores, ya que en ambientes dinámicos se requieren movimientos frecuentes, lo que abarata los costos debido a que no se cuenta con una instalación física.

Fácil configuración para el usuario, para esto solo basta con tener en cuenta los códigos de acceso a la red inalámbrica.

d) Desventajas de las redes inalámbricas

Interferencias, se pueden dar por teléfonos inalámbricos que operen en la misma frecuencia, también puede ser por redes inalámbricas cercanas, o inclusive por otros equipos conectados en la misma red.

Seguridad, en una red cableada es necesario tener acceso al medio que transmite la información mientras que en la red inalámbrica el medio de transmisión es el aire.

- **Módulos XBEE**

El presente proyecto de investigación trata de una red ZIGBEE para el control y monitoreo de un brazo robótico, por lo que en el apartado redes

inalámbricas se dará especial énfasis en la red en mención.

XBee es el nombre comercial de DIGI INTERNACIONAL, para una familia de módulos de radio compatibles con factor de forma. Las primeras radio Xbee se introdujeron el año 2005.

Los módulos XBee puede ser utilizados con el mínimo de cuatro numero de conexiones, potencia (3.3v), planta, datos de entrada y datos de salda, con otras conexiones adicionales como Reset, Sleep.

A partir de 2010 muchos módulos XBee constan de un control de flujo de E/S, A/D y líneas indicadoras construidas en una versión XBee programable con un procesador adicional.

- **Redes ZIGBEE**

ZigBee es el nombre de la especificación de un conjunto de protocolos de alto nivel de comunicación inalámbrica para su utilización con radiodifusión digital de bajo consumo, basada en el estándar IEEE 802.15.4 de redes inalámbricas de área personal (WPAN). Su objetivo son las aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías.

Por lo visto anteriormente, la relación entre IEEE 802.15.4 y ZigBee es parecida a la existente entre IEEE 802.11 y Wi-Fi Alliance. La especificación 1.0 de ZigBee se aprobó el 14 de diciembre de 2004 y está disponible a miembros del grupo de desarrollo (ZigBee Alliance). Un primer nivel de suscripción, denominado adopter, permite la creación de productos para su comercialización adoptando la especificación de 3500 dólares anuales. Esta especificación está disponible al público para fines no comerciales.

ZigBee utiliza la banda ISM para usos industriales, científicos y médicos en concreto 868 MHz en Europa, 915 en Estados Unidos y 2.4 GHz en todo el mundo. Sin embargo, a la hora de diseñar dispositivos, las empresas optaran prácticamente siempre por la banda de 2.4 GHz, por ser libre en todo el mundo. El desarrollo de la tecnología se centra en la sencillez y el bajo costo mas que otras redes inalámbricas semejantes de la familia WPAN. El nodo ZigBee mas completo requiere en teoría cerca del 10% del hardware de un nodo Wi-Fi típico, esta cifra baja al 2% para los nodos más sencillos.

○ **Clasificación de nodos ZIGBEE**

Se definen tres tipos distintos de módulos XBee, según su papel en la red:

- **Coordinador ZigBee, (ZigBee Coordinator, ZC).** El tipo de dispositivo más completo. Debe existir al menos un ZC por red. Sus funciones son las de encargarse de formar y controlar la red, es decir, es el ZC es el único de los nodos que puede crear una red ZigBee, maneja las direcciones, y la auto-regeneración y los caminos que debe seguir los dispositivos para conectarse entre ellos.
- **Router ZigBee, (ZigBee Router, ZR).** Interconecta dispositivos separados en la topología de la red, a partir de estar unido a una red formada, además de ofrecer un nivel de aplicación para la ejecución de código y configuración del usuario.
- El ZR sirve de intermediario entre nodos distantes, y resaltando que su uso es opcional (si no hay nodos distantes, su uso se puede obviar).

- Dispositivo terminal, (ZigBee End Device, ZED). Se trata de una versión simplificada de un router, ya que posee la funcionalidad necesaria para comunicarse con su nodo padre (el coordinador o un router, para conectarse a una red ZigBee), pero no puede transmitir información destinada a otros dispositivos XBee. De esta forma, este tipo de nodo, puede trabajar intermitentemente, pudiendo estar dormido la mayor parte del tiempo, aumentando la vida media de sus baterías.



Figura 26 Red ZigBee, de acuerdo a su papel en la red

Adicionalmente se definen dos tipos distintos de módulos Xbee, según su funcionalidad:

Dispositivo de funcionalidad completa (FFD): También conocidos como nodo activo. Es capaz de recibir mensajes en formato 802.15.4. Gracias a la memoria adicional y a la capacidad de computar, puede funcionar como Coordinador o Router ZigBee, o puede ser usado en dispositivos de red que actúen de interfaz con los usuarios.

Dispositivo de funcionalidad reducida (RFD): También conocido como nodo pasivo. Tiene capacidad y funcionalidad limitadas (especificada en

el estándar) con el objetivo de conseguir un bajo coste y una gran simplicidad. Básicamente, son los sensores/actuadores de la red.

Un nodo ZigBee (tanto activo como pasivo) reduce su consumo gracias a que puede permanecer dormido la mayor parte del tiempo (incluso muchos días seguidos). Cuando se requiere su uso, el nodo ZigBee es capaz de despertar en un tiempo ínfimo, para volverse a dormir cuando deje de ser requerido. Un nodo cualquiera despierta en aproximadamente 15 ms. Además de este tiempo, se muestran otras medidas de tiempo de funciones comunes:

Nueva enumeración de los nodos esclavo (por parte del coordinador):
aproximadamente 30 ms.

Acceso al canal entre un nodo activo y uno pasivo: aproximadamente 15 ms.

○ **Protocolo ZigBee**

El protocolo ZigBee posibilita tres funciones importantes en una red:

- Ruteo, posibilita el reenvío de mensajes entre módulos distantes.
- Creación de redes Ad Hoc, instantáneamente sin intervención humana (espontáneas)
- Auto Regeneración, identificación automática de nodos y rutas faltantes, reconfiguración de rutas.

○ **Topología**

ZigBee permite tres topologías de red:

- Topología en estrella: el coordinador se sitúa en el centro.
- Topología en árbol: el coordinador será la raíz del árbol.

- Topología de malla: al menos uno de los nodos tendrá más de dos conexiones.

La topología más interesante (y una de las causas por las que parece que puede triunfar ZigBee) es la topología de malla. Ésta permite que si, en un momento dado, un nodo del camino falla y se cae, pueda seguir la comunicación entre todos los demás nodos debido a que se rehacen todos los caminos. La gestión de los caminos es tarea del coordinador.

- **Estrategias de conexión de los dispositivos en un red ZigBee**

Las redes ZigBee han sido diseñadas para conservar la potencia en los nodos 'esclavos'. De esta forma se consigue el bajo consumo de potencia. La estrategia consiste en que, durante mucho tiempo, un dispositivo "esclavo" está en modo "dormido", de tal forma que solo se "despierta" por una fracción de segundo para confirmar que está "vivo" en la red de dispositivos de la que forma parte. Esta transición del modo "dormido" al modo "despierto" (modo en el que realmente transmite), dura unos 15ms, y la enumeración de "esclavos" dura alrededor de 30ms, como ya se ha comentado anteriormente.

En las redes Zigbee, se pueden usar dos tipos de entornos o sistemas:

Con balizas; es un mecanismo de control del consumo de potencia en la red. Permite a todos los dispositivos saber cuándo pueden transmitir. En este modelo, los dos caminos de la red tienen un distribuidor que se encarga de controlar el canal y dirigir las transmisiones. Las balizas que dan nombre a este tipo de entorno, se usan para poder sincronizar todos los dispositivos que conforman la red, identificando la red doméstica, y describiendo la estructura de la "supertrama". Los intervalos de las balizas

son asignados por el coordinador de red y pueden variar desde los 15ms hasta los 4 minutos.

Este modo es más recomendable cuando el coordinador de red trabaja con una batería. Los dispositivos que conforman la red, escuchan a dicho coordinador durante el "balizamiento" (envío de mensajes a todos los dispositivos -broadcast-, entre 0,015 y 252 segundos). Un dispositivo que quiera intervenir, lo primero que tendrá que hacer es registrarse para el coordinador, y es entonces cuando mira si hay mensajes para él. En el caso de que no haya mensajes, este dispositivo vuelve a "dormir", y se despierta de acuerdo a un horario que ha establecido previamente el coordinador. En cuanto el coordinador termina el "balizamiento", vuelve a "dormirse".

Sin balizas; se usa el acceso múltiple al sistema Zigbee en una red punto a punto cercano. En este tipo, cada dispositivo es autónomo, pudiendo iniciar una conversación, en la cual los otros pueden interferir. A veces, puede ocurrir que el dispositivo destino puede no oír la petición, o que el canal esté ocupado.

Este sistema se usa típicamente en los sistemas de seguridad, en los cuales sus dispositivos (sensores, detectores de movimiento o de rotura de cristales), duermen prácticamente todo el tiempo (el 99,999%). Para que se les tenga en cuenta, estos elementos se "despiertan" de forma regular para anunciar que siguen en la red. Cuando se produce un evento (en nuestro sistema será cuando se detecta algo), el sensor "despierta" instantáneamente y transmite la alarma correspondiente. Es en ese momento cuando el coordinador de red, recibe el mensaje enviado por el

sensor, y activa la alarma correspondiente. En este caso, el coordinador de red se alimenta de la red principal durante todo el tiempo.

2.3. HIPOTESIS DE LA INVESTIGACION

Se desarrollará un prototipo de brazo robótico en Autodesk Inventor, para implementarlo físicamente con materiales ligeros y resistentes, el cual mediante la programación de una interfaz en LabView pueda simular movimientos a escala de automatismos industriales, que trabajara por medio de comunicaciones inalámbricas en el estándar IEEE 802.15.4, ó el correspondiente a las redes ZigBee, mediante un microcontrolador PIC16F877A, con la finalidad de emular labores de riesgo industriales, manipulación de objetos o de precisión.





CAPÍTULO III: METODO DE LA INVESTIGACION

METODO DE LA INVESTIGACION

3.1. VARIABLES DE LA INVESTIGACION

La hipótesis general planteada para el proyecto es:

“Desarrollar y construir el módulo de entrenamiento de un brazo robótico de cinco grados de libertad, diseñado en una escala de 1 a 100, implementado en LabView monitorizado mediante redes ZigBee, para emular tareas de paletización, tanto en modo manual como en modo automático.”

De dicha hipótesis se obtienen las siguientes variables:

a) Variable independiente:

- Diseño e implementación de un brazo robótico.
- Factor humano en el proceso de implementación

b) Variable dependiente:

- Paletizado de objetos
- Redes inalámbricas
- Programación orientada a objetos
- Servomotores
- Voltajes

3.2. METODOLOGIA DE INVESTIGACION

Inductivo-deductivo

3.3. TIPO DE INVESTIGACION

La presente investigación es de tipo aplicativo.

3.4. NIVEL DE INVESTIGACION

Se trata de un proyecto de tesis de nivel experimental, debido a la complejidad del proyecto en sí, así como la eficacia y la eficiencia alcanzada durante el desarrollo del mismo.

3.5. FUENTES DE RECOLECCION DE DATOS

Medidas experimentales y simulaciones, así como los conocimientos previos al desarrollo del proyecto de investigación.

3.6. ANALISIS E INTERPRETACION DE DATOS

Para el desarrollo del presente proyecto de investigación se recurrirá a un análisis descriptivo para la obtención de resultados previos, lo que servirá para formular una conclusión parcial.

Dichas conclusiones obtenidas servirán para verificar la hipótesis.

Posteriormente la interpretación de los datos permitirá comparar los resultados obtenidos con los resultados de otras investigaciones de la misma índole de relación al presente tema.



**CAPÍTULO IV: CARACTERIZACION DEL AREA DE
INVESTIGACION**



CARACTERIZACION DEL AREA DE INVESTIGACION

4.1. UBICACIÓN GEOGRAFICA DE LA PROVINCIA DE PUNO



Figura 27 Mapa distrital de Puno

Fuente: <http://e-callao.net/2013/02/ejecutivo-declara-en-emergencia-districtos-en-puno-huanuco-y-huancavelica-ante-riesgo-de-deslizamientos/>

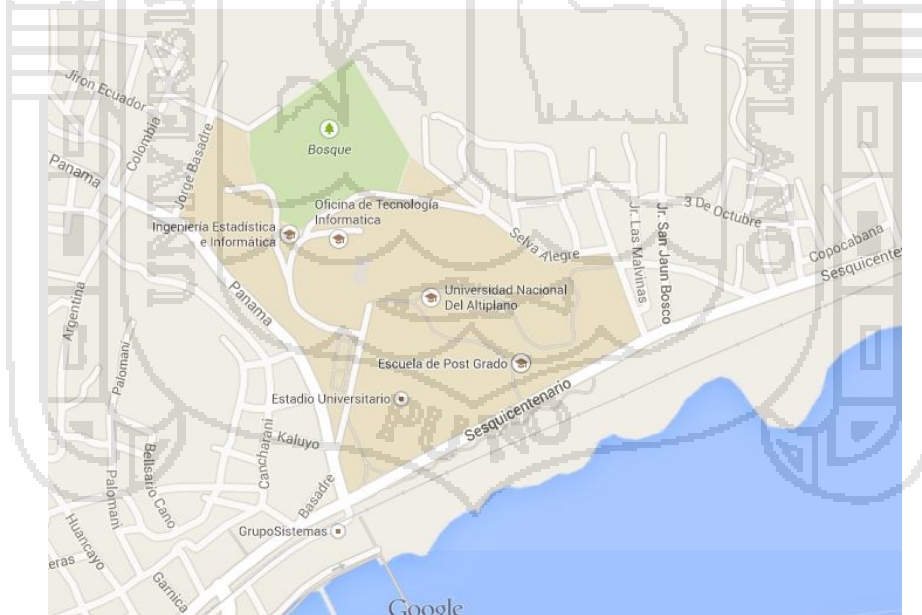


Figura 28 Croquis de la Universidad Nacional del Altiplano - Puno

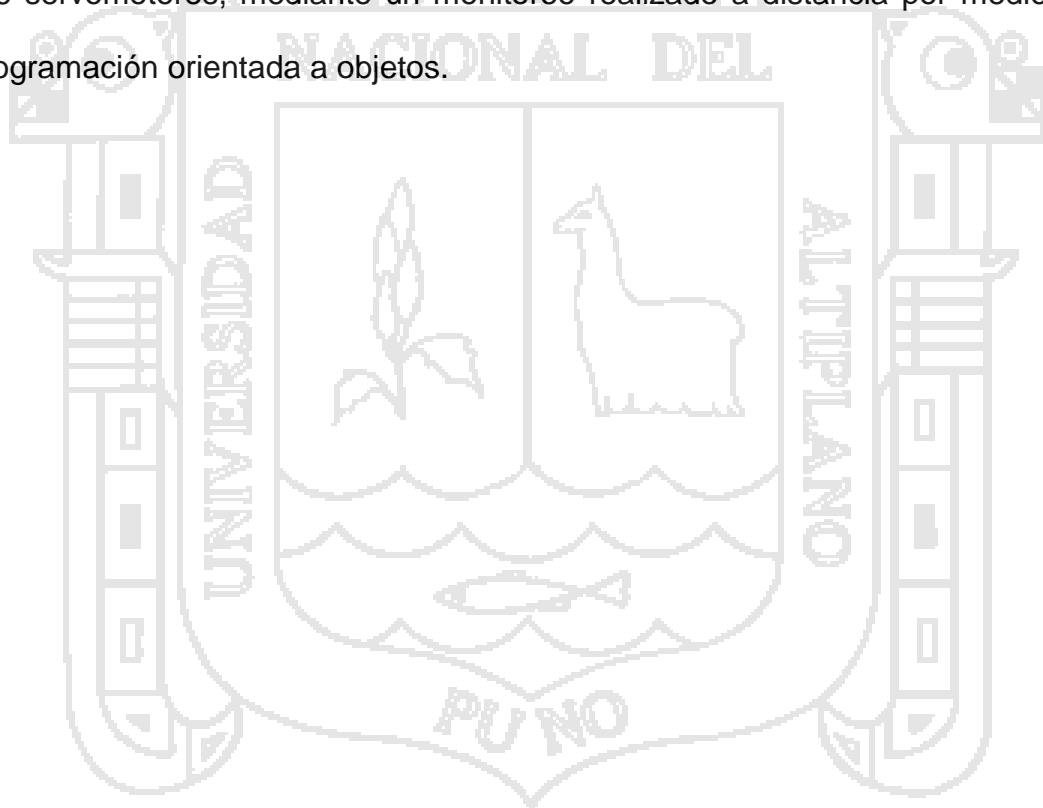
Fuente: <https://www.google.com.pe/maps/@-15.8239867,-70.0153836,16z>

La concepción inicial del presente proyecto se dio a raíz de proyectos en el área de robótica, además de la significancia a nivel de la industria de

manipuladores robóticos, surgidos anteriormente en nuestra Primera Casa de Estudios del Departamento de Puno, la Universidad Nacional del Altiplano.

Así mismo se incluyeron conocimientos del área de telecomunicaciones, así referencia específica al telemando o telecontrol.

De esta manera y bajo el preámbulo precedente se realizó un prototipo de brazo robot de unos 50 cm de altura, que cuente con la circuitería necesaria para el control manual y automático (previamente programadas) del módulo, y que a la vez posea cinco grados de libertad y una pinza manipuladora, cuyo control se realice a través de un microcontrolador y el cual dota de movimiento a 6 servomotores, mediante un monitoreo realizado a distancia por medio de programación orientada a objetos.





IMPLEMENTACION DEL MODULO DE ENTRENAMIENTO DE BRAZO ROBOTICO

Durante los periodos de elaboración, desarrollo, ensamblaje y demás, los autores del presente proyecto de tesis, intervinieron sobre el objeto de estudio logrando modificaciones directas o indirectas para acondicionamiento necesario del propósito en mención: Prototipo físico a escala de un brazo robótico de 5 GDL.

Además de haberse realizado dentro de las condiciones previstas, las cuales se dieron durante el lugar y tiempo determinados del fenómeno estudiado

En el desarrollo del presente trabajo de investigación se pueden diferenciar claramente cuatro etapas fundamentales:

La primera etapa cubre el diseño y desarrollo del circuito con el cual se controla el brazo robot, para realizar este diseño se ha de comprender perfectamente cómo funcionan los servomotores y como gestiona su movimiento, además también hay que tener claro cómo se realiza la comunicación con el ordenador y de que entradas se va a dotar al sistema para el control.

En la segunda etapa se procede a montar el brazo robot, para poder realizar sobre él posteriormente las pruebas pertinentes. En este caso se trata de un conjunto de piezas que se adquieren ya orientadas a la construcción mecánica del brazo robot.

En la tercera etapa se ha desarrollado la programación necesaria para que el brazo robot pueda funcionar de manera manual sin necesidad de un ordenador.

Una vez programado el microcontrolador se realizan las pruebas de movimiento correspondientes. Adicionalmente en esta etapa se crea la parte de

programación correspondiente a la comunicación serie, aunque no se depura hasta la etapa siguiente.

En esta última etapa se depura la programación necesaria para poder comunicar el brazo robot con el ordenador. Para poder realizar una correcta depuración de la programación se realiza, a partir de la herramienta de programación orientada a objetos LabVIEW, una aplicación que permite la comunicación entre el microcontrolador y el ordenador y que permite desarrollar todas las habilidades del brazo robot.

5.1. CIRCUITO DE CONTROL

5.1.1. MICROCONTROLADOR PIC16F877A

Hay diversas razones que han llevado a utilizar este microcontrolador en este trabajo. En primer lugar la gran cantidad de información y aplicaciones disponibles para adquirir conocimientos sobre el dispositivo en cuestión, un precio razonable y sobre todo las características técnicas que presenta. En la figura 29 se pueden observar todas las entradas y salidas disponibles del microcontrolador PIC 16F877A.

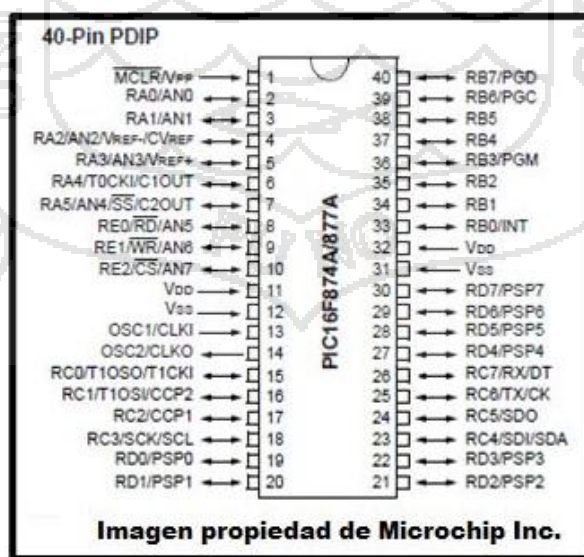


Figura 29 E/S PIC 16F877A

Fuente: Microchip Inc.

Este dispositivo es capaz de trabajar a una frecuencia máxima de 20MHz. Puesto que no se encuentra dentro de los objetivos reducir el consumo y por tanto, tampoco ajustar la velocidad del microcontrolador a la mínima necesaria, en este proyecto se utilizan los 20MHz como frecuencia de funcionamiento del sistema. Por otro lado, también dispone de una serie de entradas analógicas que resultaban bastante interesantes tanto a nivel de diseño como de aprendizaje práctico, para este trabajo se ha configurado concretamente una de las entradas analógicas. Es importante, que tenga la capacidad de comunicarse por medio de un puerto serie. Cuenta con numerosas entradas (en este caso se necesitan siete digitales y una analógica) y salidas (6 digitales para este trabajo) para evitar complicar el diseño con técnicas de multiplexado. Si se desea estudiar en profundidad las características de este dispositivo, se ha añadido a los documentos adjuntos en el CD de este proyecto, sus hojas de características.

5.1.2. LÓGICA DE POSICIONAMIENTO

Para posicionar los servomotores, como se ha descrito anteriormente, se obtienen señales de entre 1ms y 3ms de ancho de pulso al inicio del ciclo con un periodo de 20ms, donde 1ms equivale a 0° y 3ms equivale a 180° . Para realizar el control se utiliza un temporizador (TMR1) y seis salidas digitales, el temporizador es el encargado de colocar un nivel alto o nivel bajo en dichas salidas según corresponda. Para un control de 6 servomotores se necesitan 12 flancos (1 de subida y 1 de bajada por salida digital) como muestra la figura 30. Esto equivale a 12 interrupciones por parte del temporizador TMR1. Puesto que para este proyecto solo se utiliza un temporizador, no es posible posicionar dos servomotores al mismo tiempo.

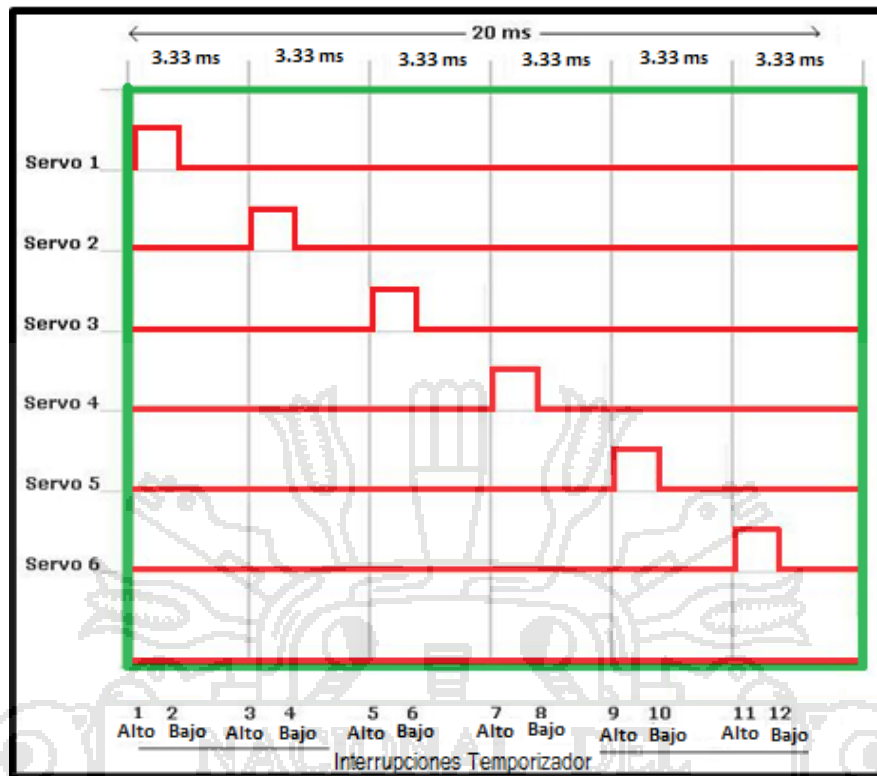


Figura 30 Control de pulsos

Ya que se han de controlar 6 servomotores, estos 20ms se han de repartir entre los 6 servomotores (como se muestra en la figura 30, además el tiempo de control del ángulo de giro (tiempo de la señal a nivel alto) de un servomotor no puede coincidir con el tiempo de control del siguiente. Por lo tanto, a cada servomotor le corresponde 3.33ms de control, sin embargo, no todo el tiempo es utilizado

5.1.3. CONTROL MANUAL

Para realizar el control manual en este proyecto se ha decidido trabajar con una entrada analógica para aprovechar las prestaciones del microcontrolador utilizado y 7 entradas digitales cuyo funcionamiento se explica un poco más adelante. El motivo de usar una entrada analógica y el resto digitales es el dotar a la pinza manipuladora de una mayor precisión. A continuación se muestra, por cuestiones de claridad, una imagen que corresponde al diseño final donde aparecen pulsadores y un potenciómetro (Figura 31.).



Figura 31 Jostick con pulsadores

La entrada analógica se realiza gracias a un divisor de tensión, en el cual, con un potenciómetro lineal y 2 resistencias fijas, se controla la tensión de entrada del canal analógico, los valores y circuitos utilizados para el cálculo se muestran en la Figura 32.

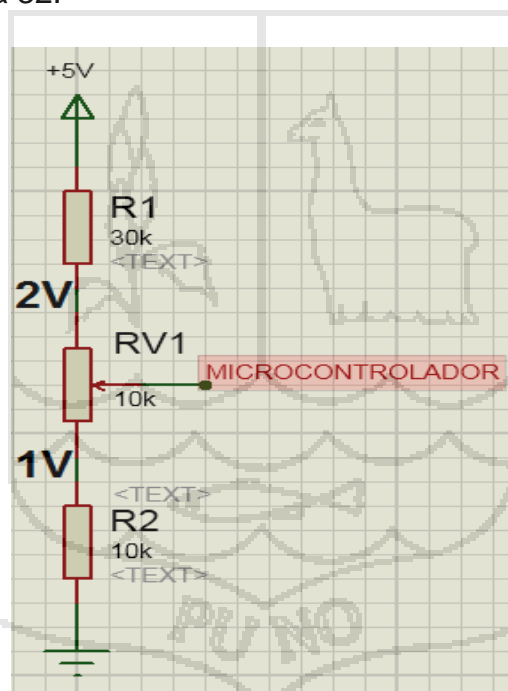


Figura 32 Circuito de control de la Pinza.

El valor analógico que se obtiene, es procesado por el microcontrolador para posteriormente actuar sobre el servomotor que controla la pinza del brazo robot. Para adquirir el dato analógico el convertidor del microcontrolador puede configurarse con una resolución de 8 bits y de 10 bits. Para obtener una mayor

precisión se configura con una resolución de 10 bits, más adelante se indica como configurar los distintos parámetros de la adquisición analógica en la programación del microcontrolador. Para una resolución de 10 bits, la precisión en términos de tensión se determina mediante la expresión 3.3.

$$1LSB = \frac{V_{ref}}{2^{10bits-1}} = \frac{5}{1024} = 4.8mV \quad \dots \text{Ecuación (3.3)}$$

Es decir, se pueden discriminar variaciones de 4.8mV, que es totalmente válido para la aplicación.

Los valores de las resistencias presentes en la figura 32, se han elegido siguiendo el siguiente criterio: A la entrada del conversor del microcontrolador se proporciona una tensión de entre 1V y 2V. Con este margen y puesto que el sistema es capaz de discriminar 4,8 mV se tienen un total de 208.3 posiciones, lo que supone que para 135° el servomotor controlado analógicamente tenga un paso de 0,64°, lo cual es una resolución bastante aceptable para el proyecto.

Para obtener una caída de tensión de dos voltios en el punto 3 de la figura V.4, se realiza el cálculo correspondiente al divisor de tensión entre R1 y (RV2+R12). Si a la resistencia R1 se le da un valor de 30 KΩ el resultado de RV2+ R12 se obtiene en la ecuación 3.4. La impedancia de entrada del microcontrolador es infinita y por tanto la corriente que circula por ella es despreciable.

$$V3 = 5 \times \frac{RV1+R2}{R1+(RV1+R2)} = 2V \quad \uparrow RV1 + R2 = 2000\Omega \quad \dots \text{Ecuación (3.4)}$$

A continuación, para obtener una caída de tensión de un voltio desde el punto uno de la figura V.4 y dando a RV2 un valor de 10KΩ, se determina el valor de R12 mediante el divisor de tensión de la ecuación 3.5.

$$V3 = 2 \times \frac{R2}{R2+RV1} = 1V \quad \uparrow R2 = 10000\Omega \quad \dots \text{Ecuación (3.5)}$$

Con lo que queda definido el circuito de control de la pinza que discrimina valores analógicos de entre 1 y 2 voltios con una precisión de 4,8 mV.

Las entradas digitales son las encargadas de controlar el servomotor que se desea mover (excluyendo la pinza) y la dirección de movimiento. Siempre que se desee mover uno de los servomotores se tiene que mantener pulsado el pulsador correspondiente al servomotor que se desea mover a la vez que el correspondiente a la dirección deseada.

5.1.4. PUERTO SERIE Y NIVELES DE TENSION

Para este trabajo se plantea un problema común hoy en día con los nuevos ordenadores, ya que no dispone de puerto serie para realizar la comunicación. Sin embargo, este problema se puede resolver mediante un adaptador USB Xbee como el que se muestra en la figura 33.

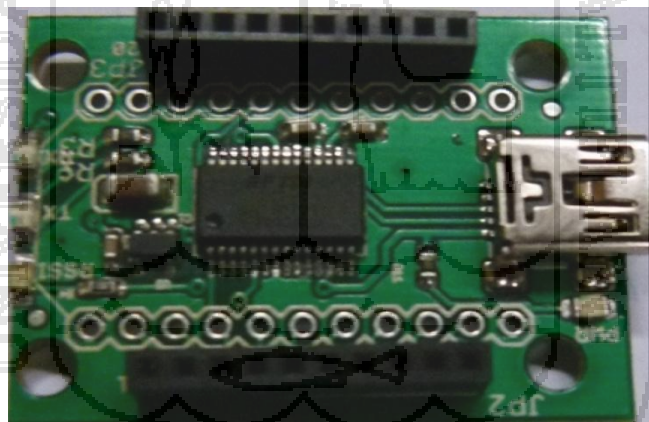


Figura 33 Adaptador USB XBee

El microcontrolador PIC 16F877A dispone de 2 patillas para la comunicación serie, sin embargo existe una pequeña limitación. Los niveles de tensión TTL con los que funciona el PIC tienen que adaptarse a los niveles de tensión con los que trabaja el ordenador. Para adaptar los niveles de tensión, en este proyecto, se utiliza el dispositivo MAX232, que se ha configurado siguiendo las instrucciones de su hoja de características. En la figura siguiente (figura 34) se

muestra la configuración utilizada para el MAX232. Para conocer más datos sobre este dispositivo, se adjuntan sus hojas de características en formato digital.

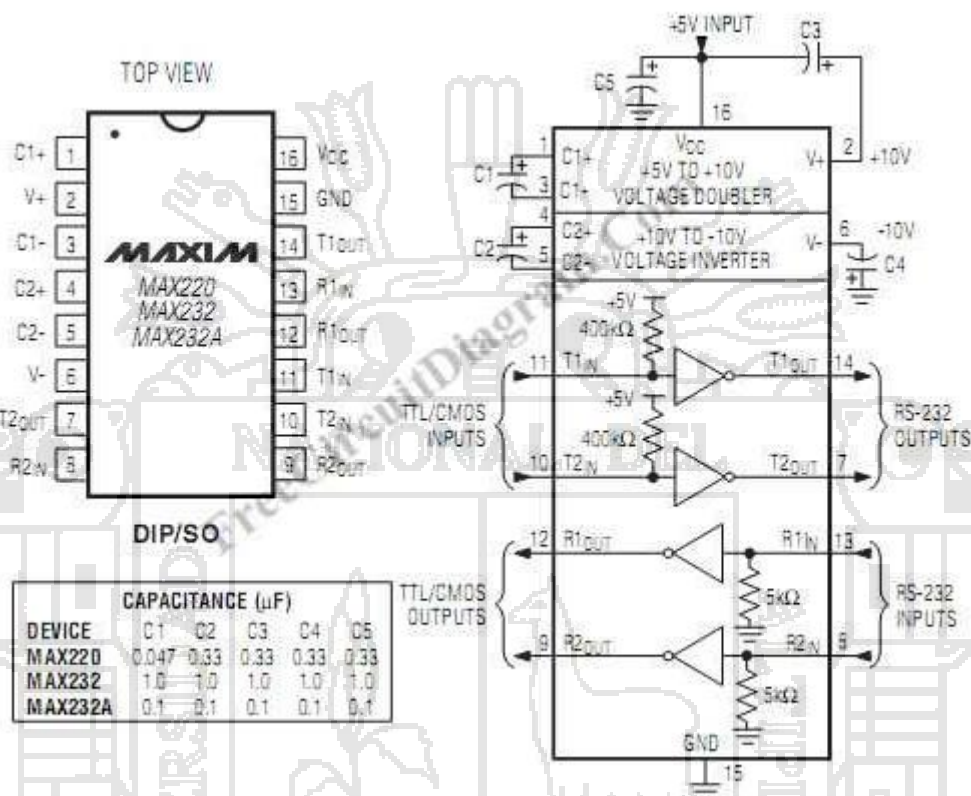


Figura 34 MAX232.

Fuente: www.freecircuitdiagram.com

5.1.5. INDICADORES LED

Para este trabajo solo se requieren dos señales luminosas, ya que casi todos los estados del sistema se pueden reconocer fácilmente desde la aplicación realizada en LabVIEW. En este proyecto, el circuito de control se compone de 2 diodos LEDs, uno rojo y otro verde. El diodo LED rojo se ha dispuesto para mostrar si se le proporciona alimentación al circuito e indica también con un parpadeo de 1 segundo de periodo, cuando inicializamos el programa de comunicación con el ordenador, que no quiere decir que se inicie el control del

brazo robótico. El LED verde indica esto último, es decir, indica que se está realizando un control serie del brazo robot desde el ordenador. En este momento están intercambiando datos de posicionamiento entre ordenador y brazo. En la figura 35 se pueden apreciar los LEDs utilizados en el diseño final.

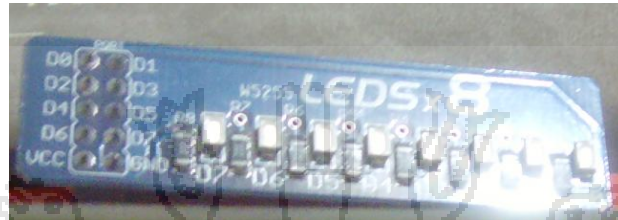


Figura 35 LEDs indicadores

5.1.6. CONECTORES

Se ha utilizado, para la realización de este trabajo, una serie de conectores que corresponden a la alimentación, la comunicación con el brazo robot y el conector de programación.

Para la alimentación se necesitan 2 conectores, uno para la alimentación de los servomotores, y otro para la de la placa de control. Ambas alimentaciones disponen de un circuito de protección contra inversión de la polaridad y condensadores de desacoplo de 47nF entre las entradas de alimentación y el potencial de referencia de las tensiones utilizadas en el microcontrolador (0 Voltios).

La comunicación del brazo robot a la placa de control se consigue simplemente con una tira de 6 conectores en línea de patilla torneada.

Por último se ha diseñado el sistema para que la programación se pueda realizar por conector RJ11 que permite utilizar un programador ICD2, ICD...etc. o por conector ICSP que permite utilizar programadores como el PicKit2, PicKit, GTPUSB...etc.

El aspecto final de la circuitería es el que se muestra en la figura 36. En ella se muestran todas las partes explicadas anteriormente.

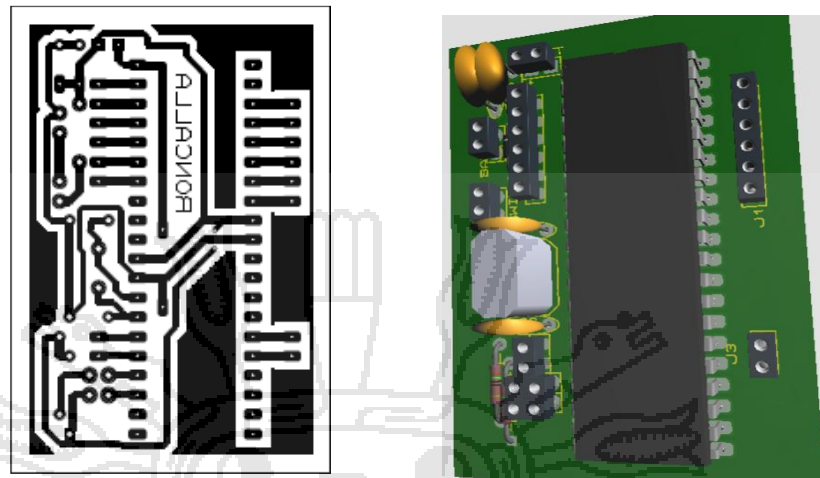


Figura 36 Circuitería final

5.2. MONTAJE DEL PROTOTIPO

Para realizar el montaje de la estructura mecánica se utilizan una serie de elementos metálicos especialmente diseñados para este tipo de estructuras. También se disponen los actuadores (servomotores) que son los responsables directos del movimiento de la estructura, cuya tornillería de fijación se proporciona junto con los servomotores. El diseño de toda la estructura es personalizado, por lo que no existe ningún esquema de proveedor para realizar el montaje. Por esta razón se procede a explicar a continuación cada uno de los pasos a realizar para su montaje. La base de la estructura está compuesta por dos piezas de fibra de vidrio las cuales se superponen y giran una respecto a la otra.

A continuación en la figura 37 se muestra una imagen general de la parte mecánica, que sirve para tener una visión global de la misma.

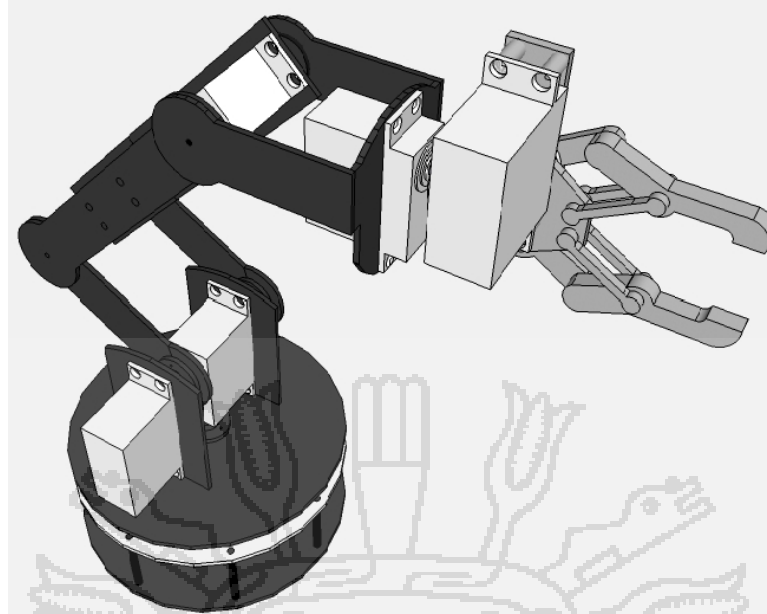


Figura 37 Visión global de la parte mecánica.

La base contiene una base fija, o parte A, a la que se le acoplan unos rodamientos para facilitar el movimiento giratorio de la pieza superpuesta o parte B. La parte A y sus rodamientos se pueden observar en la figura 38.



Figura 38 Imagen de la base (Parte A).

En la figura 38 se pueden apreciar unas marcas en los rodamientos, estas marcas se han realizado gracias a una lija de grano como indican las

instrucciones de montaje para esta pieza. El interior de la parte A de la base es hueco y es el lugar donde se dispone el servomotor 6, encargado de realizar el movimiento giratorio de la base. En la siguiente figura (figura 39) se presenta el interior hueco de la parte A y el servomotor 6 correctamente fijado.



Figura 39 Interior de la Parte A.

En la figura 40 se muestra la parte B, se aprecia que esta parte presenta un orificio central donde encaja el eje de giro del servomotor 6 que se aparece en la parte central del elemento de la figura 38.



Figura 40 Base (zona inferior de la parte B).

Los tornillos y tuercas que se observan en la figura 40 corresponden a la fijación de la parte B de la base con la pieza que se observa en la figura 41 (Soporte Multifuncional para Servomotores) que se encuentra por la otra cara. Esta pieza sirve como fijación del siguiente servomotor (servomotor 5).



Figura 41 Soporte multifuncional para servomotores.

En la siguiente imagen (figura 42.) se muestra un esquema de montaje del conjunto formado por la figura 41, la base en su totalidad y el servomotor 5.

Para fijar los servomotores a los soportes multifuncionales se utilizan unos remaches de nylon que fijan por presión.

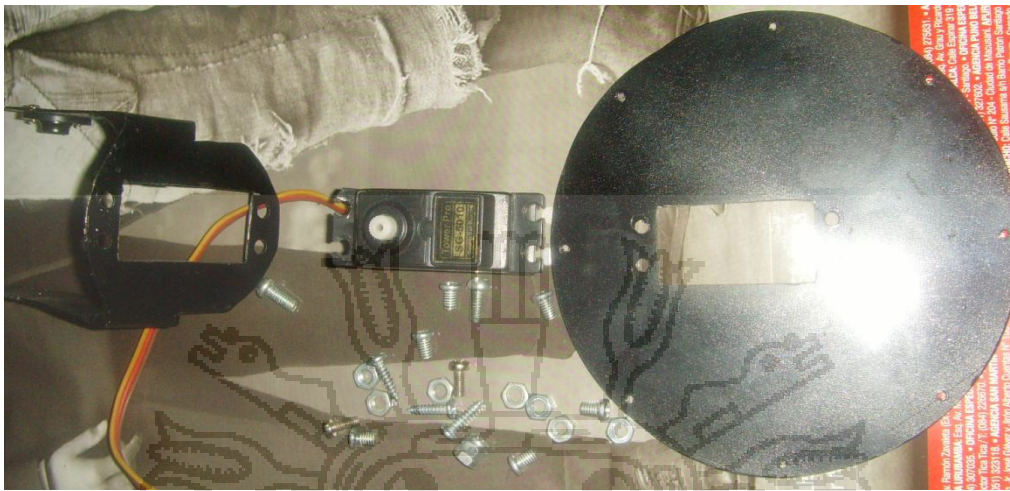


Figura 42 Proceso de montaje del servomotor 5.

En siguiente paso, se acopla el elemento de la figura 43 (a), al servomotor y al soporte multifuncional montado anteriormente como se muestra en la figura 43 (b).



Figura 43 a) Soporte en largo para servomotor b) Montaje servomotor 5.

Para atornillar los soportes en C a los soportes multifuncionales se utilizan los elementos de la figura 44 en el orden que se indica.



Figura 44 Tornillería para fijar soporte multifuncional y soporte.

A continuación se fija otro soporte largo anterior como se muestra en la siguiente imagen (figura 45), en la cual no aparece el soporte multifuncional que aparecía ya acoplado a uno de los dos soportes para una mayor claridad de montaje.



Figura 45 Soportes largos acoplados.

Seguidamente, para colocar el servomotor 4, se fija al último soporte colocado, un nuevo soporte multifuncional y en el lateral de este soporte multifuncional un

soporte en largo. Una vez montadas estas 2 estructuras se instala en el soporte multifuncional el servomotor 4, con lo que se obtiene la estructura de la siguiente imagen. (Figura 46).



Figura 46 Montaje del servomotor 4.

Una vez montada esta última parte se atornilla el soporte al extremo plano de un soporte largo, por su parte exterior como se muestra en la figura 47.

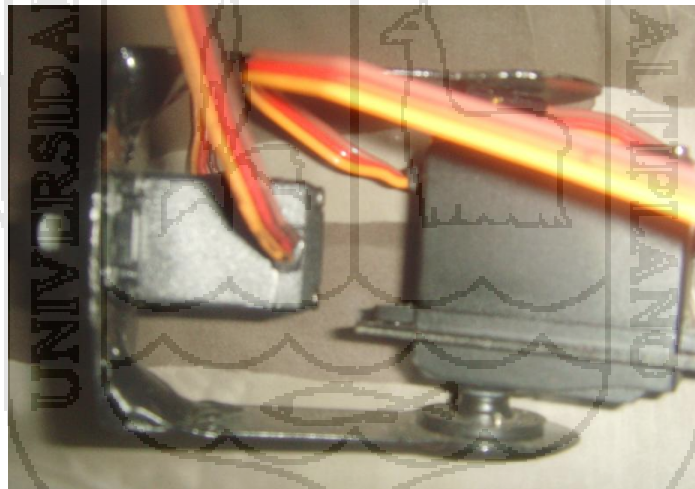


Figura 47 Soporte en C adosado a soporte en Largo.

En el extremo de este último soporte en C se acoplan dos soportes multifuncionales ambos superpuestos entre sí, como se aprecia en la figura 48.

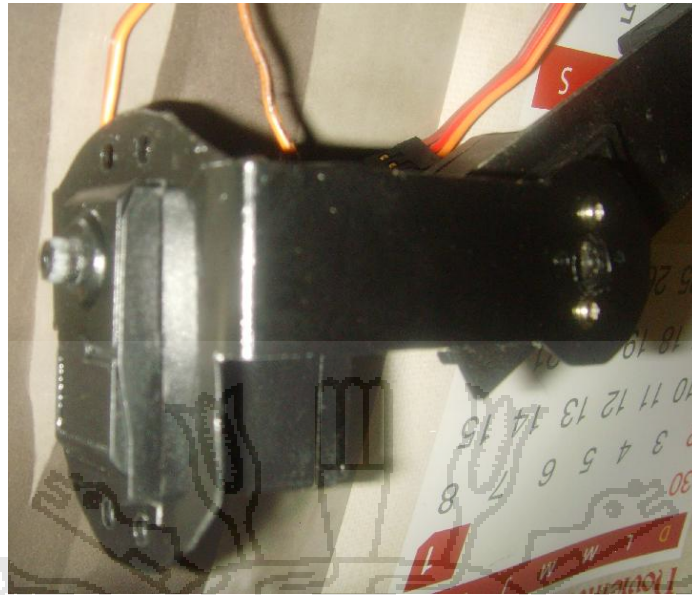


Figura 48 Extremo de soporte en C con bloques multifuncionales superpuestos.

A cada uno de los bloques multifuncionales se le instala un servomotor (servomotor 3 y servomotor 2). Una vez hecho esto, se procede a la instalación de la pinza manipuladora que se muestra en la figura 49, la cual dispone de un hueco en la parte inferior para colocar el servomotor 1 y un orificio para colocar el eje de giro del servomotor 2 en la parte opuesta a la pinza.



Figura 49 Pinza manipuladora

Realizados todos los pasos, el diseño final de la parte mecánica queda como se muestra en la figura 37.

Una vez terminada de montar toda la parte mecánica se comprueba que los servomotores se han colocado en la posición correcta, es decir, que el ángulo de giro que se puede realizar con ellos es el máximo permitido por la propia mecánica. Después de comprobar esto manualmente, si alguno de los servomotores no se encuentra en su posición más óptima, se desatornilla su eje de giro de la parte fija que lo sujeta y una vez puesto dicho servomotor en la posición deseada se vuelve a atornillar a la parte metálica que fija el eje de giro.

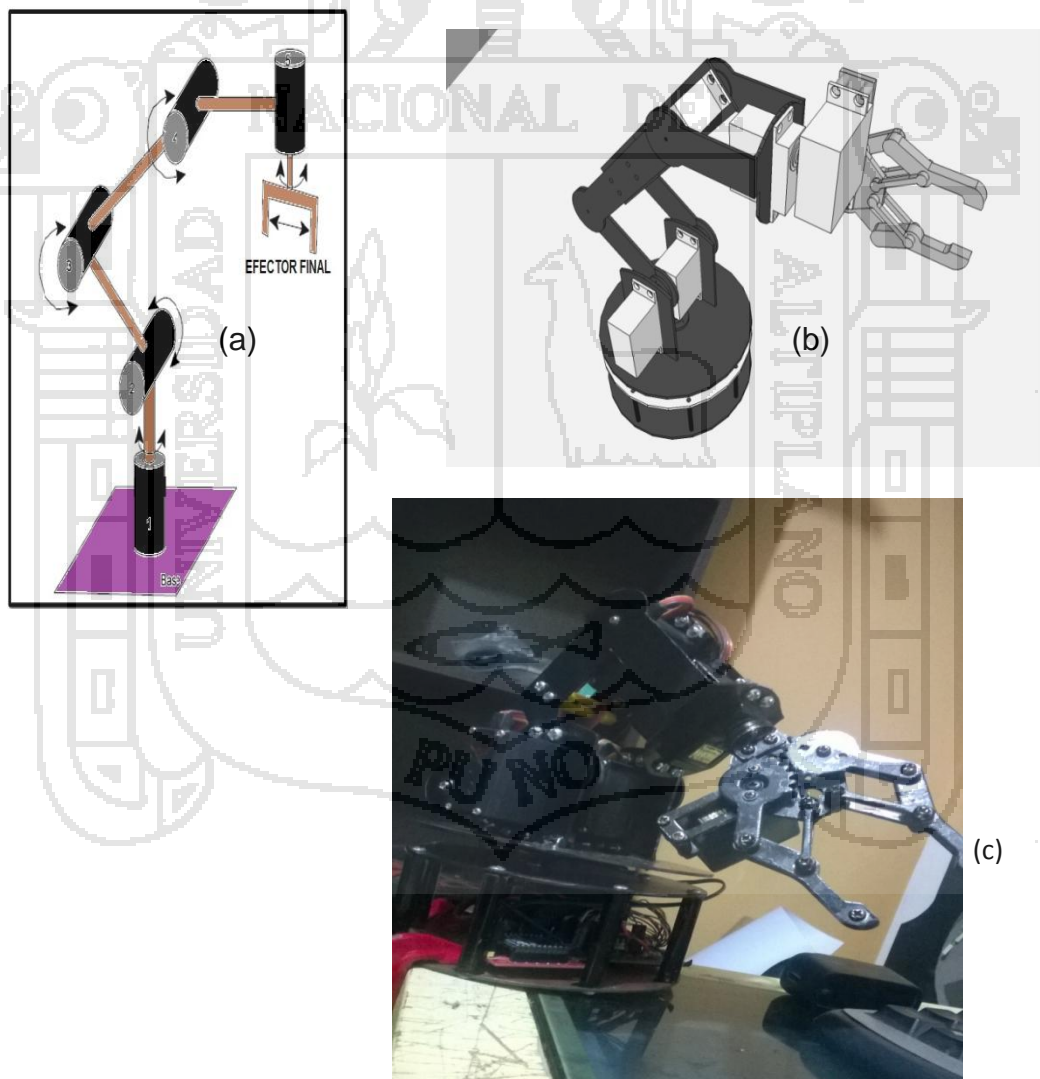


Figura 50 Etapas de montaje: (a) Desarrollo, (b) simulación e (c) implementación.

5.3. MICROPROGRAMA

En este apartado se describe detalladamente el microprograma desarrollado para el control del brazo robótico, además para implementar el microprograma se requiere de una serie de herramientas que son descritas a continuación. Dicho microprograma está formado por dos partes: la encargada de realizar el control manual mediante los pulsadores y el potenciómetro de la circuitería, y por otro lado existe una parte encargada de gestionar el control por ordenador. Para realizarlo se explotan algunos de los recursos que presenta el microcontrolador como es el temporizador (TMR1), el cual consta de 16 bits y el conversor analógico/digital de 10 bits. También intervienen en esta parte de la programación distintas entradas y salidas digitales.

A lo largo del microprograma, la parte dedicada a un algoritmo que interpreta el dato recogido por el conversor analógico/digital, se encarga de transformarlo en un ancho de pulso determinado para uno de los servomotores (pinza manipuladora). Por último, se crea otra parte que se encarga del protocolo de comunicación entre el ordenador y la circuitería.

5.3.1. HERRAMIENTAS DE PROGRAMACIÓN

Para llevar a cabo el microprograma se necesita de la herramienta MPLAB X de Microchip, en este caso se ha utilizado la versión 8.36. Al MPLAB X se le ha añadido una expansión (Se adjunta en el CD) que permite utilizar como compilador de C el programa PIC C Compiler. El aspecto del programa MPLAB X se puede ver en la figura 51. Se trata de un paquete de programas que tiene diversas funciones como simulación, depuración en línea y programación de dispositivos, entre otras. Es un programa muy versátil ya que soporta distintos tipos de programadores, depuradores y emuladores.



Figura 51 Aspecto del MPLAB X.

Además también se requiere de un programador/depurador que sirve para programar el microprograma en la memoria del microcontrolador y depurarlo en línea. En la figura 52. Se muestra el programador/depurador utilizado para este trabajo, como se puede observar, se trata de un PicKit2 clone también de Microchip y es totalmente compatible con el microcontrolador PIC16F877A.



Figura 52 Vistas de PicKit2 clone.

Como se puede observar en la figura 33, este equipo cuenta con un conector USB para su comunicación con el ordenador. Además cuenta, por el lado opuesto al conector USB, de un conector RJ11 por medio del cual se programa el microcontrolador gracias a la circuitería desarrollada.

Para conectar el PicKit2 clone a la circuitería se ha seguido el esquema proporcionado con el propio dispositivo. En la figura 53. Se puede observar la forma de conectar de las patillas del PicKit2 clone al circuito y los elementos mínimos para una correcta programación y/o depuración. Una vez realizada la correcta programación y/o depuración se puede comprobar las conexiones entre la circuitería y el resto de elementos del proyecto (Servomotores, Ordenador, Programador/Depurador).

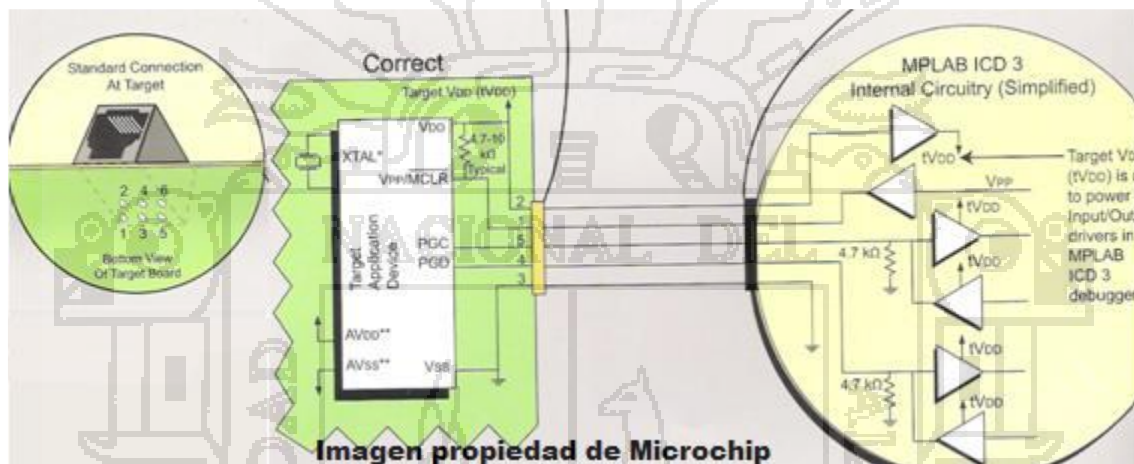


Figura 53 Conexión Correcta PicKit2 clone-Microcontrolador.

El lenguaje utilizado para la programación, es en su totalidad lenguaje C. Se ha utilizado este lenguaje ya que, aunque no se puede conseguir el código más óptimo en lo que respecta al tiempo, proporciona una mayor claridad y facilidad de comprensión.

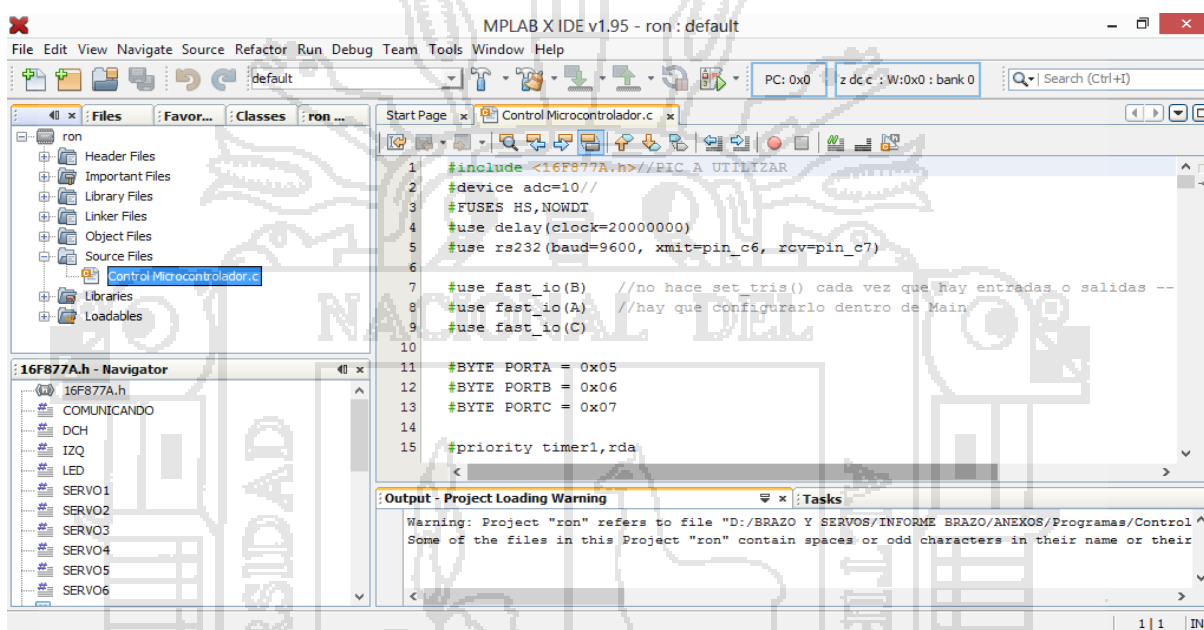
Tanto la circuitería de programación como el programa necesitan de otra circuitería que ha de ser programada.

5.3.2. PROGRAMACIÓN MODO MANUAL

La parte de microprograma correspondiente al modo manual, es la encargada de posicionar cada uno de los seis servomotores que contiene el brazo robótico. Tiene la capacidad de gestionar cada una de las señales de entrada que son introducidas por el usuario manualmente como son: Movimiento de la

pinza robótica manipuladora (servomotor 1) mediante una entrada analógica, la selección de los servomotores del dos al seis y la selección de la dirección de movimiento hacia 0° o hacia 135°. Esta limitación de movimiento hasta los 135° es debida a la propia mecánica del brazo robot.

En primer lugar se presenta la cabecera del programa, la cual se muestra en la figura 54.



```

1 #include <16F877A.h> //PIC A UTILIZAR
2 #device adc=10//
3 #FUSES HS,NOWDT
4 #use delay(clock=2000000)
5 #use rs232(baud=9600, xmit=pin_c6, rcv=pin_c7)
6
7 #use fast_io(B) //no hace set tris() cada vez que hay entradas o salidas --
8 #use fast_io(A) //hay que configurarlo dentro de Main
9 #use fast_io(C)
10
11 #BYTE PORTA = 0x05
12 #BYTE PORTB = 0x06
13 #BYTE PORTC = 0x07
14
15 #priority timer1,rda

```

Output - Project Loading Warning

Warning: Project "ron" refers to file "D:/BRAZO Y SERVOS/INFORME BRAZO/ANEXOS/Programas/Control
Some of the files in this Project "ron" contain spaces or odd characters in their name or their

Figura 54 Cabecera del Microprograma.

Para este apartado, interesa conocer solo ciertos elementos, el resto se explican en el apartado correspondiente a la comunicación serie. Al principio se incluye la librería "16F877A H" correspondiente al microcontrolador PIC 16F877A, la cual es responsable de que las funciones que se utilizan en el programa, predefinidas en C y preparadas para introducir los argumentos, funcionen correctamente.

En la línea 2 se define la resolución del convertor analógico/digital del microcontrolador, en este caso la resolución seleccionada es de 10 bits, que es la máxima capaz de proporcionar el microcontrolador.

Antes de comenzar cualquier microprograma, es necesario configurar los Bits de configuración. En este caso es necesario configurar el tipo de reloj, con el cual se indica la velocidad del reloj (HS para 20MHz según la documentación del microcontrolador) y desactivar el WatchDog (NOWDT). Existen otros bits de configuración para este microcontrolador, para configurarlos existen una serie palabras de configuración como las que se muestran en la figura 55. El compilador se encarga de cargar una configuración por defecto a los bits de configuración que no se han declarado.

PIC16F877A	
1.00	LP Low power osc < 200 khz
1.00	XT Crystal osc <= 4mhz
1.00	HS High speed Osc (> 4mhz)
1.00	RC Resistor/Capacitor Osc with CLKOUT
1.02	NOWDT No Watch Dog Timer
1.02	WDT Watch Dog Timer
1.03	NOPUT No Power Up Timer
1.03	PUT Power Up Timer
1.04	PROTECT Code protected from reads
1.04	NOPROTECT Code not protected from reading
1.06	NOBROWNOUT No brownout reset
1.06	BROWNOUT Reset when brownout detected
1.07	NOLVP No low voltage prgmg, B3(PIC16) or B5(PIC18) used for I/O
1.07	LVP Low Voltage Programming on B3(PIC16) or B5(PIC18)
1.08	NOCPD No EE protection
1.08	CPD Data EEPROM Code Protected
1.09	WRT_5% Lower 255 bytes of Program Memory is Write Protected
1.09	WRT_25% Lower quarter of Program Memory is Write Protected
1.09	WRT_50% Lower half of Program Memory is Write Protected
1.09	NOWRT Program memory not write protected
1.11	NODEBUG No Debug mode for ICD
1.11	DEBUG Debug mode for use with ICD

Figura 55 Bits de configuración.

Las funciones que se observan en la figura 54 y que son nombradas como "fast_io(X)", dotan de velocidad al programa ya que solo permite funcionar a los registros de entrada y salida (A,B,C) según se configure en el programa principal, es decir, no se puede realizar la salida de un dato por un registro o por un bit del registro que está programado como entrada sin antes

programarlo como bit o registro de salida, lo que quiere decir que el registro no se autoconfigura con esta forma de programarlo.

A continuación se asignan las direcciones de los registros PORTA, PORTB y PORTC a las constantes "PORTA", "PORTB" y "PORTC" para mayor claridad en el resto de la programación, como se muestra en las líneas 11,12 y13.

La última línea concede niveles de prioridad a las interrupciones, que este proyecto son de dos tipos: el TMR1 (temporizador, "timer1") y USART (puerto serie, "rda"). Esta línea concede mayor prioridad a las interrupciones que se encuentran más a la izquierda. En este caso la interrupción del TMR1 es la primera en atenderse.

A continuación se definen una serie de constantes que hacen referencia a bits de entrada y salida del microcontrolador. Al definirlos con un nombre concreto se consigue una mayor claridad a la hora de realizar e interpretar el programa. Se definen así los servomotores, los pulsadores que definen el movimiento en un sentido o en otro y por último los 2 LEDs indicadores. La figura 56 muestra la asignación explicada.

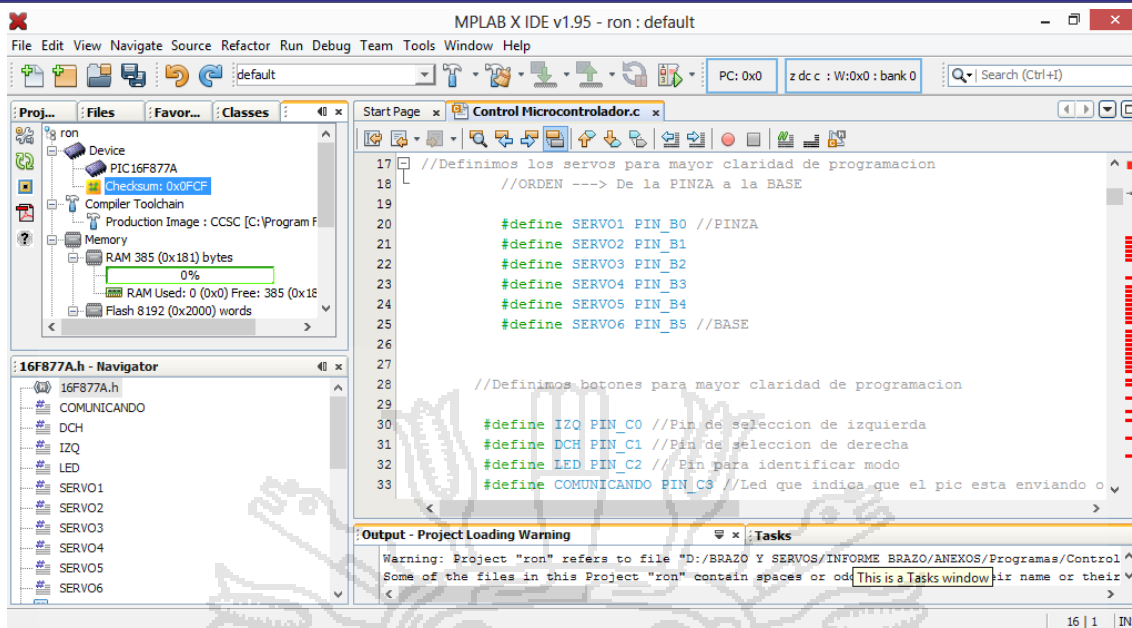


Figura 56 Asignación de patillas de entrada y salida.

En la siguiente imagen (figura 57), se muestran una serie de variables las cuales se utilizan para el desarrollo del programa, además de unas constantes que definen las temporizaciones del sistema.

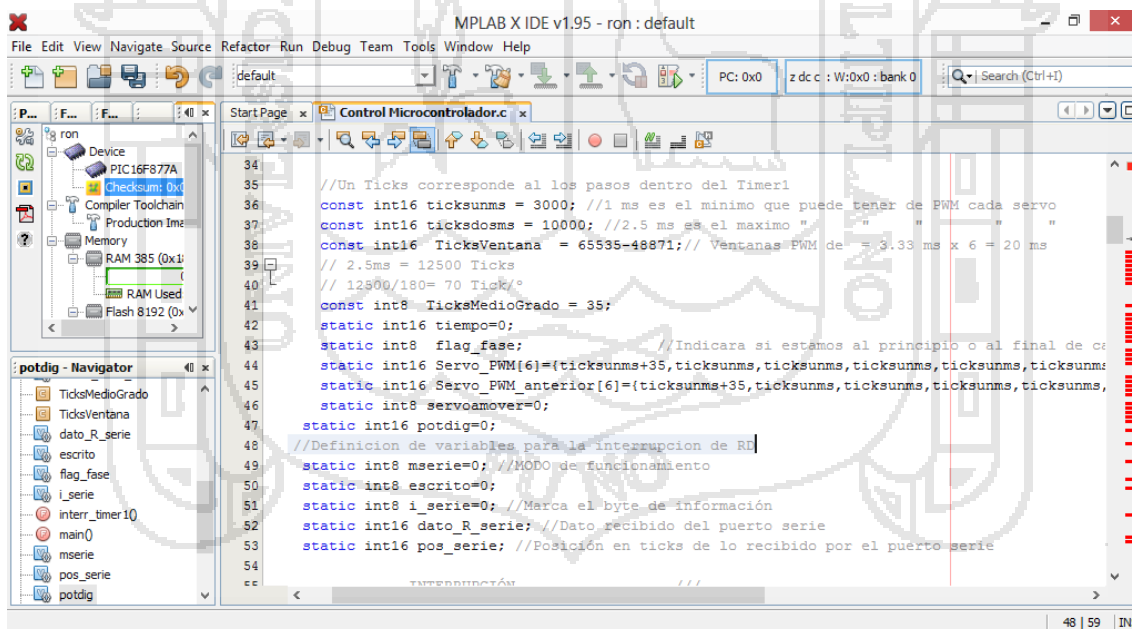


Figura 57 Definición de variables y constantes.

5.4. APLICACIÓN EN LABVIEW

Para resolver la comunicación con el ordenador se realiza una aplicación en la herramienta de programación LabVIEW, cuya metodología de programación es

la programación orientada a objetos. Para acceder a la comunicación se genera un panel frontal con el que el usuario puede interactuar y que lleva asociado un diagrama de bloques que es transparente para el mismo.

5.4.1. LABVIEW PANEL FRONTAL

En la figura se muestra el panel frontal con el que el usuario interacciona para controlar el movimiento del brazo robot y generar las secuencias que se deseen. En la figura 58 se muestra el panel frontal de control.

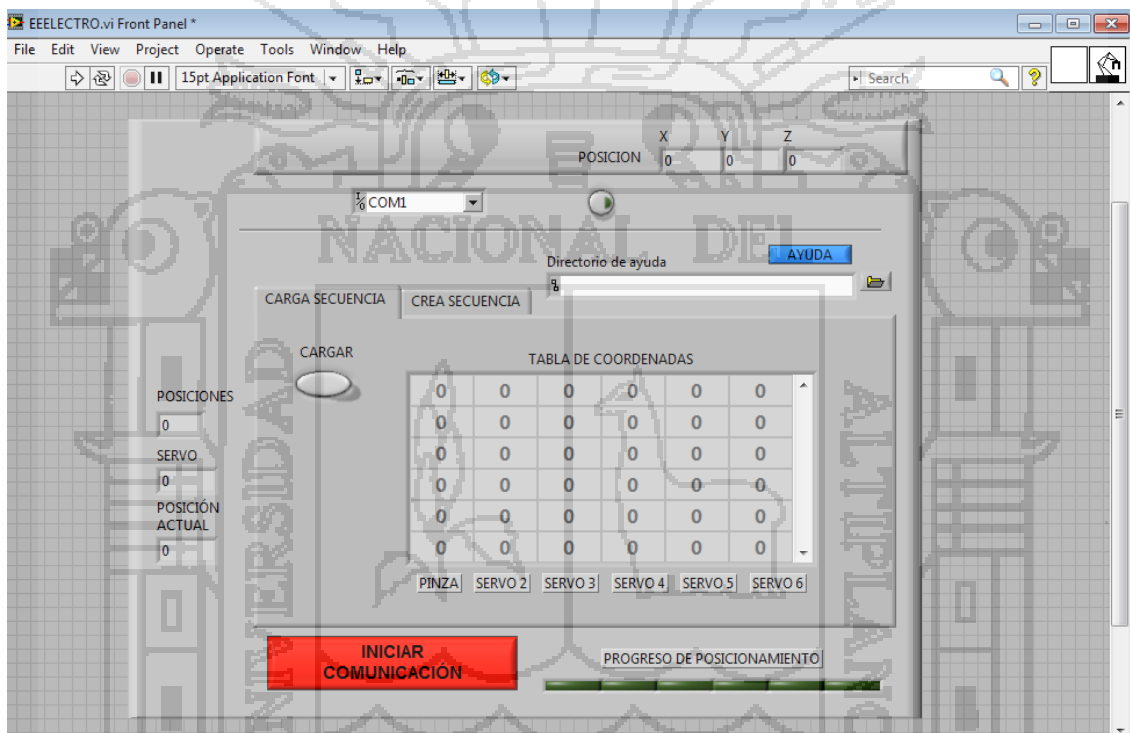


Figura 58 Panel Frontal del programa en LabVIEW.

En el panel de la figura 58 se diferencian varias partes. En la parte superior se ha dispuesto un recuadro en el que se pueden desplegar un panel con los distintos puertos disponibles para seleccionar. Una vez seleccionado el puerto al que se conecta el cable DB9-USB, se hace un clic en el pulsador que se encuentra a la derecha confirmando el puerto de comunicación como se muestra en la figura 59.

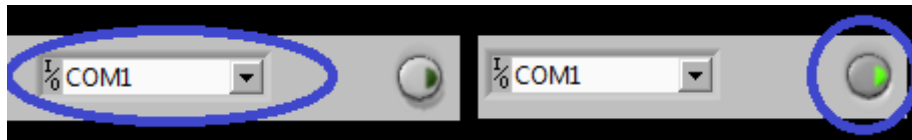


Figura 59 Selección del puerto.

Tras seleccionar el puerto existen diversas opciones de funcionamiento; en primer lugar, se puede seleccionar el modo de cargar una secuencia creada previamente y en segundo lugar existe también la opción de generar una secuencia de movimiento. Estas opciones se seleccionan a partir de las dos pestañas que se muestran en la figura 60.

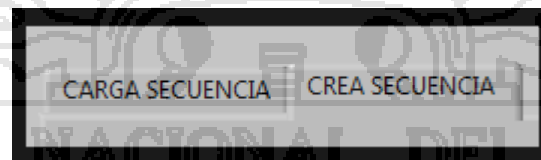


Figura 60 Pestaña de acciones.

El recuadro verde contiene el pulsador con el que se añade una posición nueva a la secuencia que se genera dentro de un fichero *.txt, esta posición va después de la posición anterior. El recuadro azul contiene la posición del brazo que se guarda en la última posición y en la que se ha de posicionar el brazo robot. Finalmente el recuadro amarillo contiene 6 controles deslizantes, uno para cada servomotor, con los que se posiciona al brazo robot y la pinza manipuladora en la posición deseada. Todas las posiciones con los que se configuran los servomotores son posiciones relativas a 0° y como se ha dicho anteriormente, los valores no pueden ser superiores a 135° , ya que la propia mecánica no lo permite. Por otro lado, las posiciones solo pueden tener pasos de 1° , es decir, números enteros de 0° a 135° donde para la pinza manipuladora 0° es la posición en la que se encuentra completamente abierta. Para realizar la carga de una secuencia que se haya guardado previamente se ha de pulsar el botón "CARGAR" que se presenta en la figura 61.

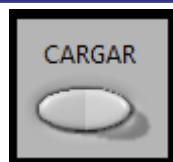


Figura 61 Pulsador de carga de secuencia.

Una vez que la secuencia se ha cargado, en la tabla que aparece en la parte central del panel, se representan, por filas, cada una de las articulaciones del brazo robot, expresadas en grados y, por columnas, cada una de las posiciones de los servomotores como se muestra en la figura 62.

TABLA DE COORDENADAS					
0	0	60	0	0	0
0	0	0	50	120	100
90	0	60	80	0	100
90	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
PINZA	SERVO 2	SERVO 3	SERVO 4	SERVO 5	SERVO 6

Figura 62 Tabla de coordenadas.

Una vez que se ha cargado la secuencia en la tabla basta con pulsar el pulsador rojo con el nombre "INICIAR COMUNICACIÓN" que se encuentra en la parte inferior del panel frontal. Cuando la comunicación se ha iniciado, en la parte inferior izquierda, se ve cual es el servomotor que está actualizando su posición mediante una barra de indicadores rectangulares de color verde como los que se muestran en la figura 63.

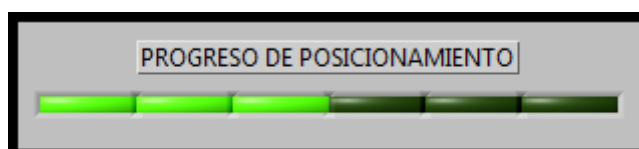


Figura 63 Barra de posicionamiento.

Existen una serie de elementos informativos a la izquierda del panel frontal. Estos elementos se encargan de informar de las posiciones que toma el brazo a lo largo de toda la secuencia, del servomotor que se está controlando en ese momento y de la posición actual en la que se encuentra el brazo robot. La figura 64 muestra dicho panel.

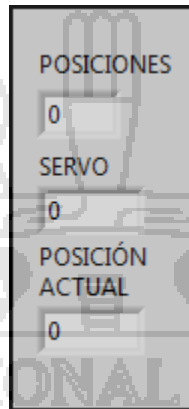


Figura 64 Panel de información general.

Finalmente se ha incluido dentro de la interfaz un pulsador por medio del cual se puede acceder a un fichero en formato PDF de ayuda. Junto a este pulsador aparece un pequeño recuadro en el que se puede cargar el nombre del fichero de ayuda, la razón por la que se ha decidido incluir este recuadro es debida a que el programa final no queda en un solo ejecutable, por lo que el directorio del archivo de ayuda seguramente cambie y puede crear problemas al intentar cargarlo, de esta forma, simplemente hay que cargar previamente el archivo con el botón de ayuda que aparece en la pantalla principal de la figura 58.

5.4.2. LABVIEW DIAGRAMA DE BLOQUES

A continuación se representan los distintos bloques que forman parte de la programación del sistema de comunicación por puerto serie y de la gestión de los movimientos para el brazo robot. El esquema general del sistema se

muestra en la figura 65, donde aparece dentro de un bucle “while” que se repite permanentemente al iniciar el programa, una secuencia de dos elementos (Bloque A y Bloque B). Ambos elementos solo se ejecutan bajo una condición que habilita al “case”.

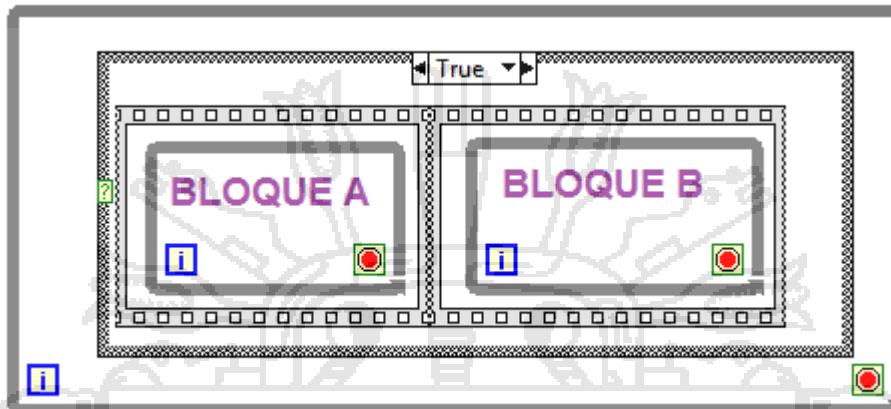


Figura 65 Esquema general del programa.

Analizando de manera más exhaustiva el sistema, cabe destacar en primer lugar la creación de una serie de variables encargadas de mostrar el estado del panel frontal tanto del servomotor que se está posicionando como de la posición en la que se encuentra el brazo. La figura 66 muestra dichos elementos.

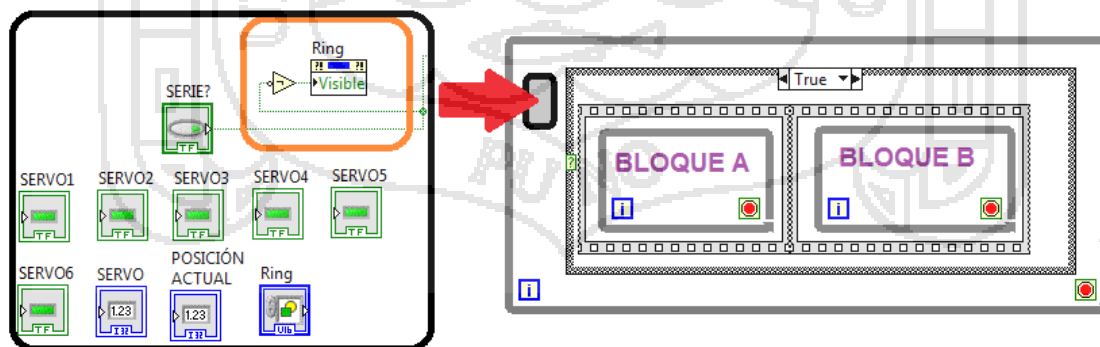


Figura 66 Variables para el panel frontal.

El pulsador con el nombre de “SERIE?” es el que verifica que se ha seleccionado un puerto (figura 59). La zona remarcada con un color naranja

permite que una vez seleccionado el puerto, se muestre la pantalla completa del panel frontal, de manera que solo en el momento de seleccionar el puerto se pueden acceder a las opciones de funcionamiento.

Una vez se ha seleccionado el puerto de transmisión, se ha de configurar este de manera que las características de transmisión en el microcontrolador y en el ordenador sean iguales. La velocidad configurada es de 9600 baudios, sin bits de paridad, un único bit de parada y 8 bits de datos. En la figura 67 se muestra el bloque funcional que realiza la configuración, como se aprecia en el montaje de la figura, la parte recuadrada en negro, precede a la estructura secuencial.

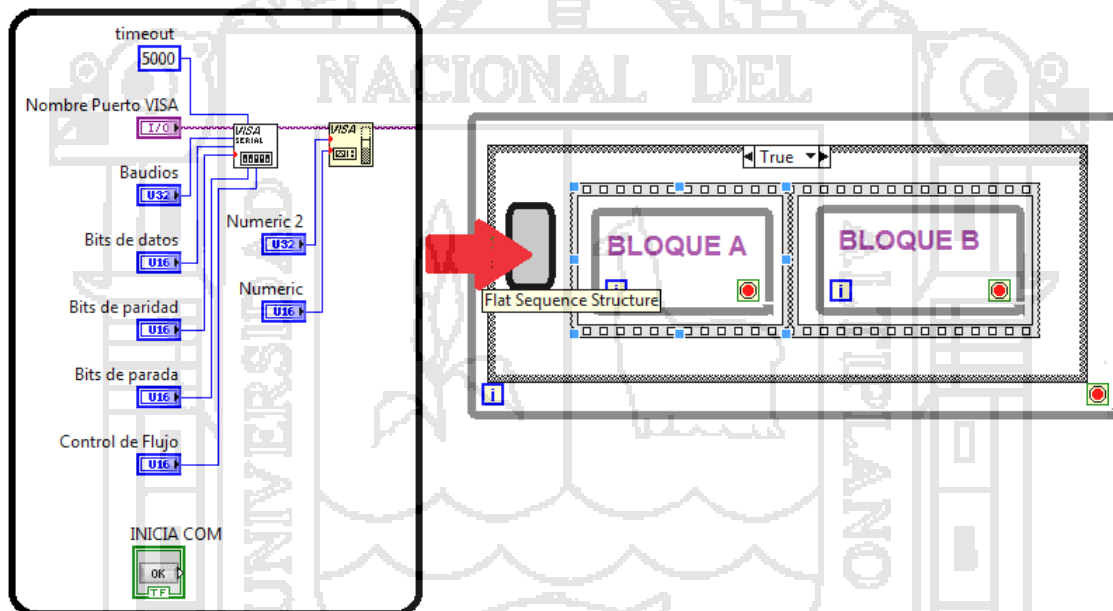


Figura 67 Configuración del puerto serie en LabVIEW.

En la figura 66 se muestra otro elemento a continuación del bloque de configuración del puerto serie que se encarga de vaciar el buffer del puerto serie. Además, aparece un pulsador con el nombre “INICIA COM” que representa al pulsador “INICIAR COMUNICACIÓN” de la figura 58.

El bloque A, que se muestra en las figuras anteriores (64, 65, 66), contiene los elementos que se muestran en la figura 68. Los elementos que se encuentran

recuadrados en rojo, se encargan de apagar todos los LEDs, los cuales indican el servomotor que se está modificando (de color verde) y de poner a cero las variables que muestran los datos en el panel de información general (coloreados en azul).

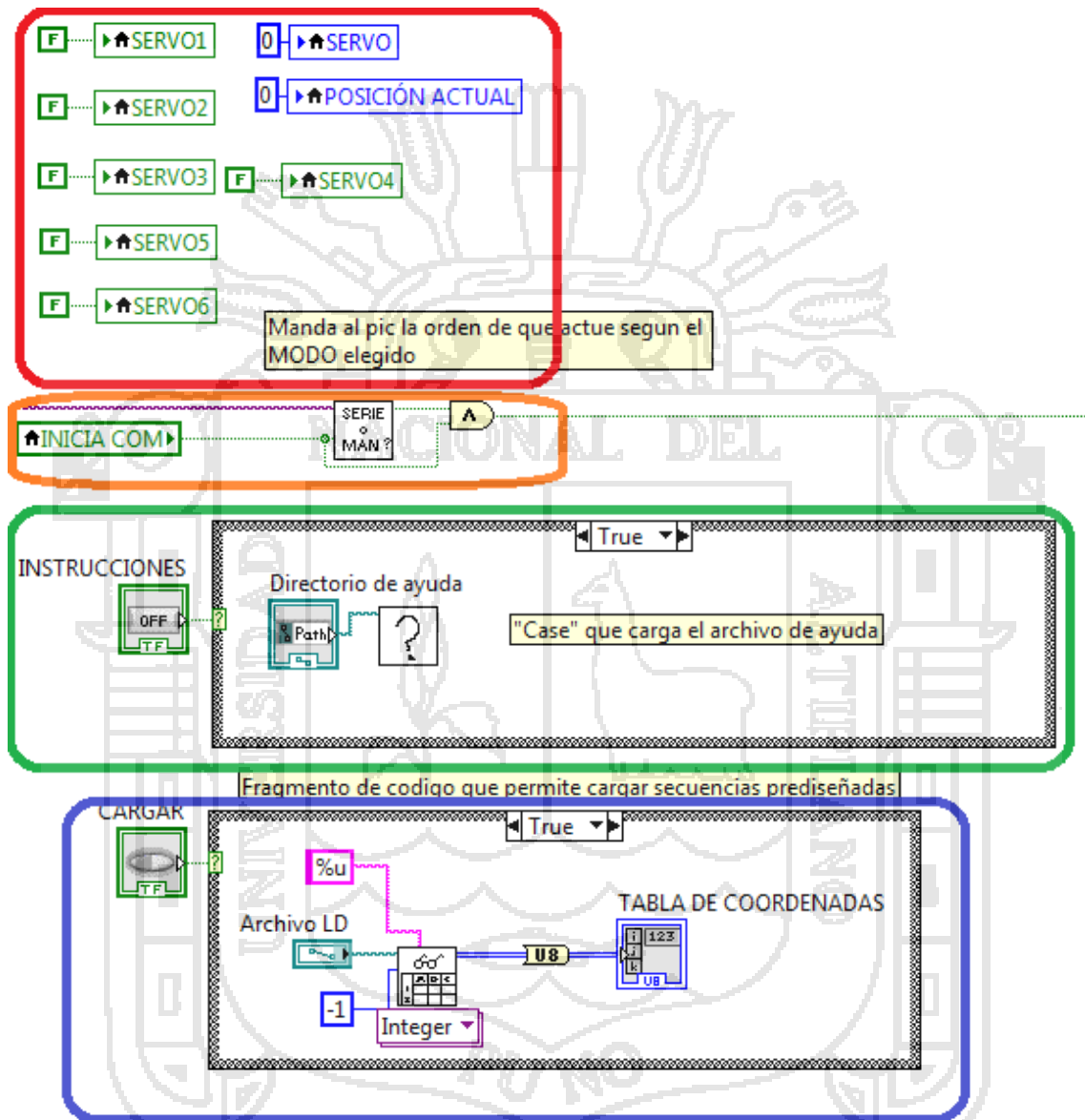


Figura 68 Bloque A del diagrama de bloques.

La zona naranja contiene al bloque que decide si el control se realiza manualmente o por el contrario, el brazo robot pasa a estar controlado por el ordenador. Este hecho determina si se pasa a la secuencia del bloque B o se continúa en la misma.

Dentro del recuadro de color verde se dispone el código que permite acceder a la ayuda del programa, la cual, se suministra en formato PDF en la documentación virtual adjunta con el CD-ROM. Esta parte contiene un subprograma encargado de la monitorización de la ayuda.

Finalmente, al final de la figura 68 se distribuyen una serie de elementos dentro del recuadro azul. Esta parte es la encargada de permitir la carga de secuencias en la tabla de coordenadas a la que puede acceder el usuario desde el panel frontal. Como se ha descrito en el apartado 5.4.1.

Una vez iniciada la comunicación se accede al bloque B. Ya en el bloque B los primeros elementos que se ejecutan son los que se muestran en la figura 69. (Antes de acceder al bucle “while”).

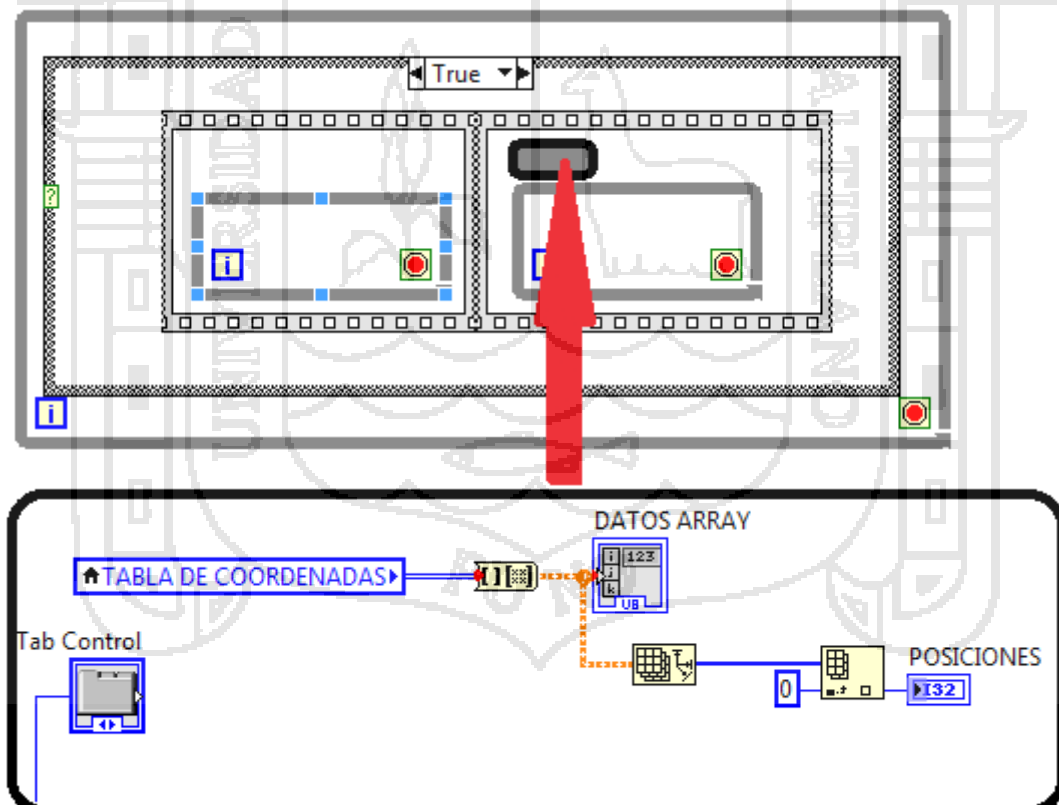


Figura 69 Elementos iniciales del Bloque B.

Lo primero que se realiza al acceder a la comunicación con el brazo robot desde el ordenador, es pasar los datos de la matriz que forma la tabla de coordenadas a un array al que se le llama "DATOS ARRAY", de la que se obtiene el número de filas que se guarda en la variable "POSICIONES" En este punto se declara el bloque (Tab Control) que permite crear diversas pestañas como las que se muestran en la figura 60.

Para tener una visión más global del funcionamiento del bloque B, a continuación en la figura 70 se muestra la estructura general de lo que hay contenido en la estructura "while" del bloque B.

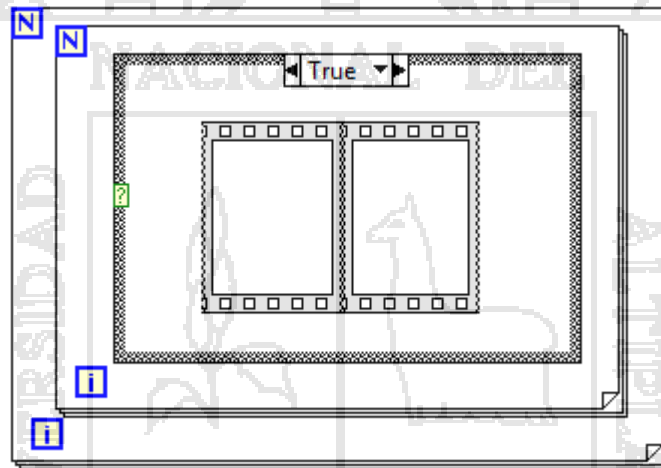


Figura 70 Bloque B.

Dentro del bucle "while" del bloque B los primeros elementos que aparecen son los que se muestran en la figura 71. En esta parte se evalúa si se crea una secuencia nueva o se desarrolla una secuencia previamente cargada. A continuación se explica cada uno de los recuadros de color.

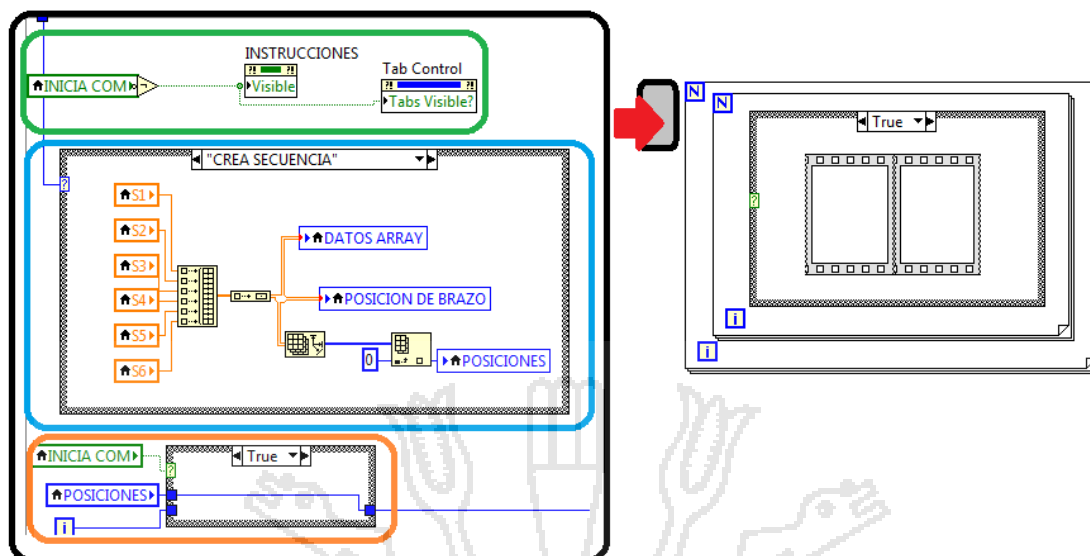


Figura 71 Primera parte del bucle “while” del Bloque B.

En la parte superior, el contenido del recuadro en color verde hace desaparecer el botón de ayuda, de manera que no se pueda acceder a la ayuda cuando se realiza una transferencia continua de datos al microcontrolador.

La estructura “case” incluida en el recuadro azul, para el caso de que se seleccione el modo de crear secuencia, actualiza todas las variables coloreadas en azul dentro del recuadro con los valores de las pestañas deslizantes que aparecen en la figura 59. Estas pestañas se nombran como “Sx” donde “x” es la pestaña que regula la posición del servomotor “x”.

Por último, dentro del recuadro naranja queda definido el número de repeticiones que se generara en un bucle “for” que se ejecuta posteriormente, que tendrá un valor “posiciones”, cuyo valor hace referencia a las distintas posiciones que pueden adquirir las articulaciones dentro de una secuencia pre-programada.

A continuación se accede a la estructura de programación que se presenta en la figura 70. El primer bucle “for” (Contenido en el Bloque B) tiene un número

de repeticiones que está definido por uno de los elementos de la figura 71 como se ve en el recuadro naranja de dicha figura.

Dentro del primer bucle “for” se colocan los elementos presentes en la figura 72 donde, para facilitar la visión general de la estructura, se realiza un montaje con un recuadro naranja donde todos los elementos que hay dentro de este están señalados con una flecha. Estos elementos se encargan de actualizar el panel que muestra la posición del brazo robótico en el momento que este está siendo posicionado. Además esta parte de programa selecciona la fila que posteriormente se manda con los valores de posición desde el ordenador al microcontrolador. Las variables coloreadas en verde son utilizadas para actualizar el valor de los LEDs de manera que los ponen todos a nivel bajo (apagados).

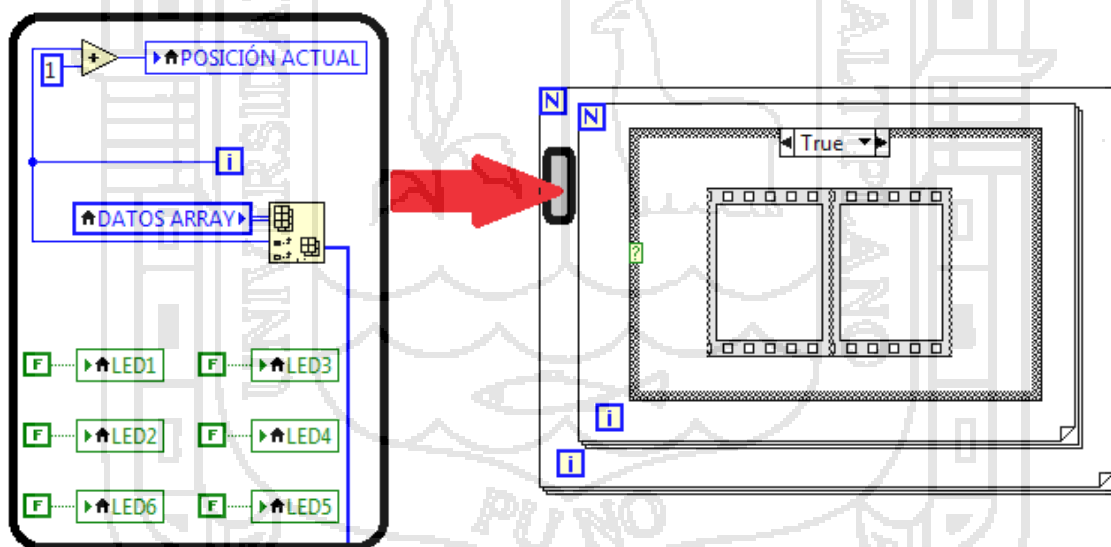


Figura 72 Contenido del primer bucle “for” del Bloque B.

El segundo bucle “for” tiene el número de repeticiones definido por el tamaño del vector de datos obtenido del bloque funcional de la figura 72. En este caso el número de datos que hay dentro del vector de datos es seis siempre, puesto que se corresponden con las posiciones de los seis servomotores que se

envían a continuación. Los elementos dentro de la segunda estructura “for” antes de pasar a la estructura case son los que se muestran en la figura 73.

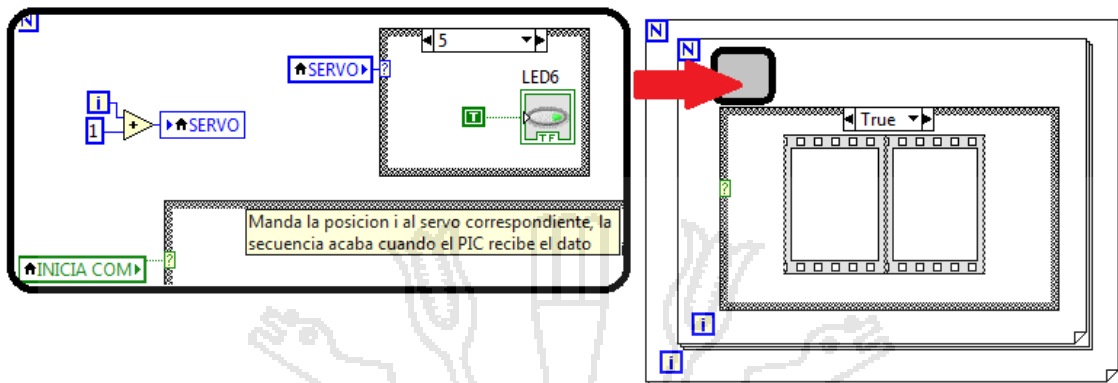


Figura 73 Contenido del segundo bucle “for” del Bloque B.

Básicamente, la función de estos elementos es la de encender el LED correspondiente al servomotor que se va a posicionar y mantenerlo encendido hasta que se vuelva a realizar el bucle con la siguiente posición de las articulaciones.

Dentro de la estructura “case” de la figura 73 es donde se encuentra la parte que realiza la transmisión de las posiciones de los servomotores desde el ordenador al microcontrolador por medio del puerto serie. La figura 74 muestra esta parte con todos los elementos claramente diferenciados por medio de recuadros de colores.

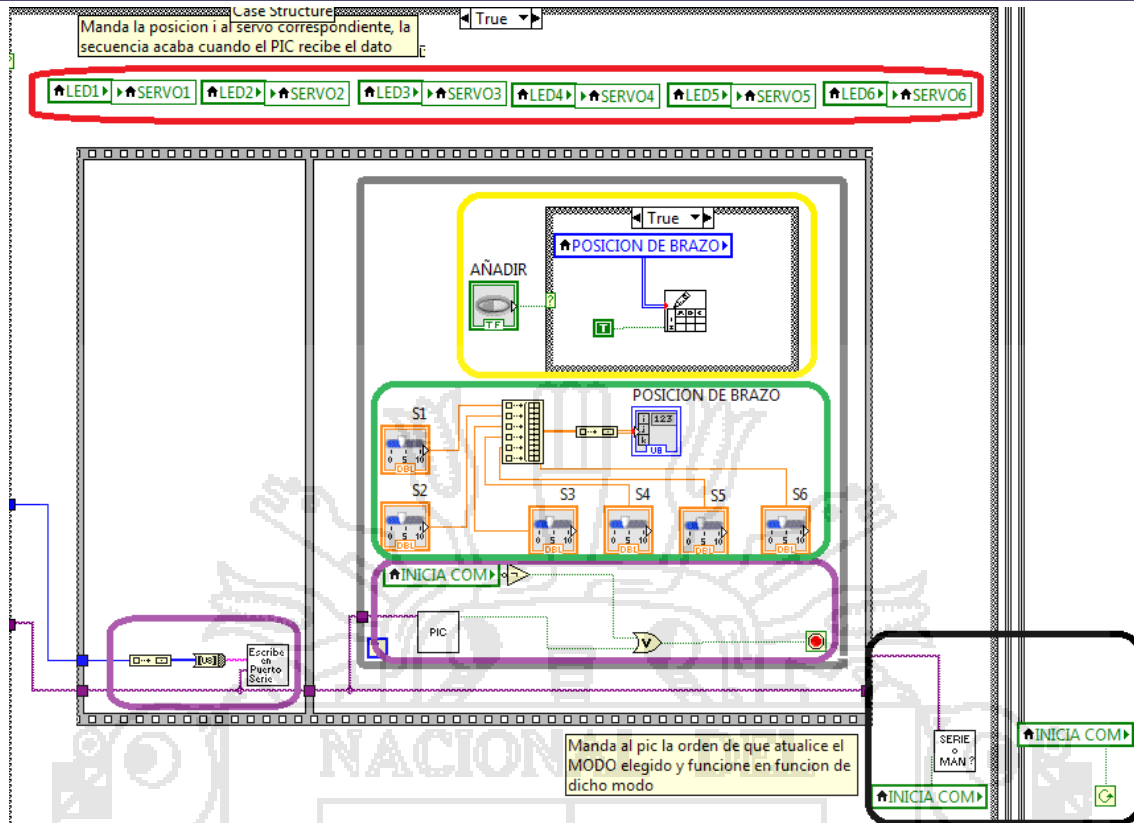


Figura 74 Estructura “case” del Bloque B.

Los elementos recogidos en el recuadro rojo actualizan el estado de los LEDs en función del estado en el proceso de posicionamiento, de manera que el usuario puede ver cuál es el servomotor al que se le está actualizando la posición.

La parte correspondiente al envío del dato, donde en este caso el dato equivale a un valor de posicionamiento, está situada en el recuadro azul. El bloque que se aprecia dentro del recuadro pertenece a un subprograma que se explica en el siguiente apartado.

El recuadro de color morado contienen los elementos que analizan la respuesta desde el microcontrolador. Una vez que el servomotor se ha posicionado correctamente, el bloque “PIC” (subprograma) devuelve un valor 1 (“True”) permitiendo salir del bucle y continuar con el resto del programa.

En el caso de que se seleccione el modo de crear secuencia, dentro de este bucle “while”, el contenido que recoge el recuadro de color amarillo, almacena la posición del brazo dentro de un bloc de notas de forma consecutiva, dependiendo del valor de las pestañas deslizantes.

Finalmente, un vez se s le del bucle “while”, dentro del recuadro negro, el programa analiza si se siguen transmitiendo datos de posicionamiento por el puerto serie o se pasa de nuevo al modo manual.

SUBPROGRAMAS

En la figura 68 aparecen 2 subprogramas. En el recuadro verde aparece un símbolo de interrogación en uno de los bloques, el contenido de ese bloque se presenta en la figura 75 y se encarga de abrir el archivo con extensión *.PDF que contiene el manual de ayuda de la aplicación.

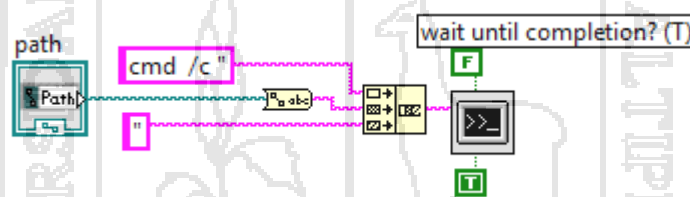


Figura 75 Bloque de ayuda.

En el recuadro naranja se puede apreciar un bloque con el nombre “ORD o MAN?” cuyo formato interno se muestra en la figura siguiente (Figura 76).

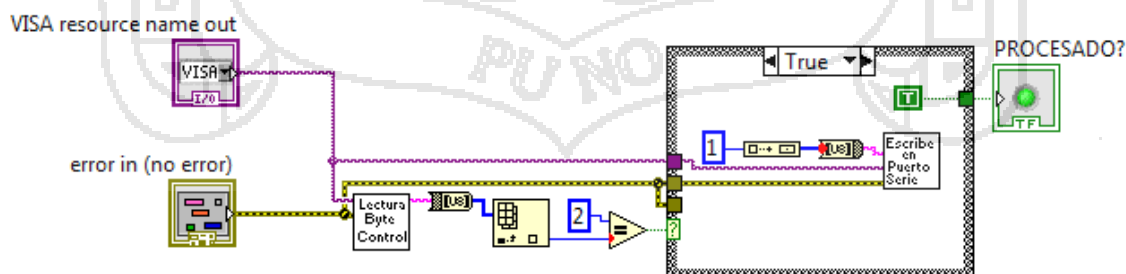


Figura 76 Bloque “Ord o Man?”

Este bloque se encarga de transmitir por el puerto serie el valor de comunicación correspondiente según se requiera un control remoto por

ordenador o volver al modo manual. Una vez el microcontrolador recibe esta información, este responde verificando que ha recibido la información y actuando en consecuencia a la información recibida. Se puede apreciar que dentro de este bloque aparecen otros dos sub-bloques, los cuales se explican en los siguientes párrafos.

Empezando desde la izquierda aparece un bloque con el nombre “Escribe en Puerto Serie” cuyo contenido se muestra en la figura 77. Este bloque limpia el buffer de escritura cada vez que se envía un dato desde el ordenador y espera 10ms después de cada escritura para posteriormente cerrar la comunicación desde la aplicación con el puerto serie.

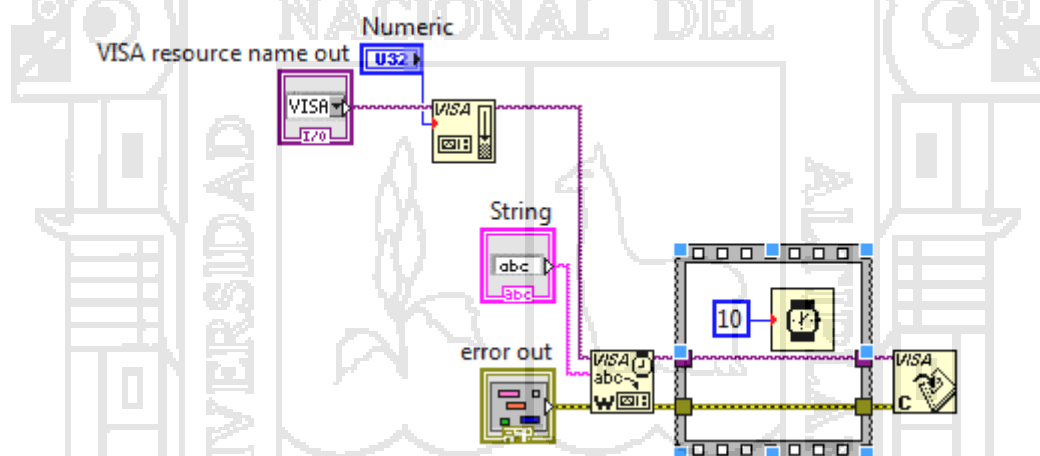


Figura 77 Bloque “Escribe en Puerto Serie”.

Más a la derecha (en la figura 76) hay un bloque con el nombre de “PIC” Este bloque contiene los elementos de la figura 78.

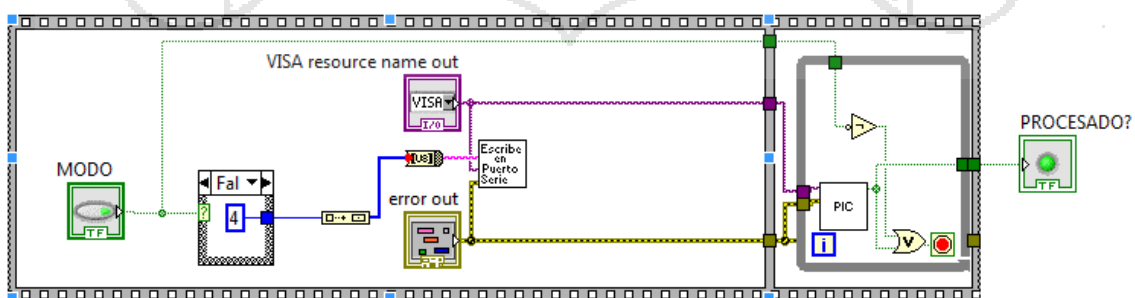


Figura 78 Bloque “PIC”.

Este bloque lee un dato enviado desde el microcontrolador y en el caso de que ese dato sea 0x02 la aplicación de LabVIEW responde que el dato ha sido recibido correctamente mediante el envío de un dato con el valor 0x01, esto provoca que la salida de este bloque tenga un valor afirmativo (true). En caso de que el valor no se haya recibido correctamente, la respuesta de este bloque es un valor negativo (false).

Finalmente aparece otro bloque dentro de la figura 78 con el nombre “Lectura Byte Control” que como su nombre indica es el encargado de leer los bytes de control que el microcontrolador manda mediante la comunicación serie al ordenador para que este, mediante la aplicación en LabVIEW, los interprete y actúe en consecuencia. La figura 79 muestra el contenido de este último módulo de la categoría clasificada como subprograma.

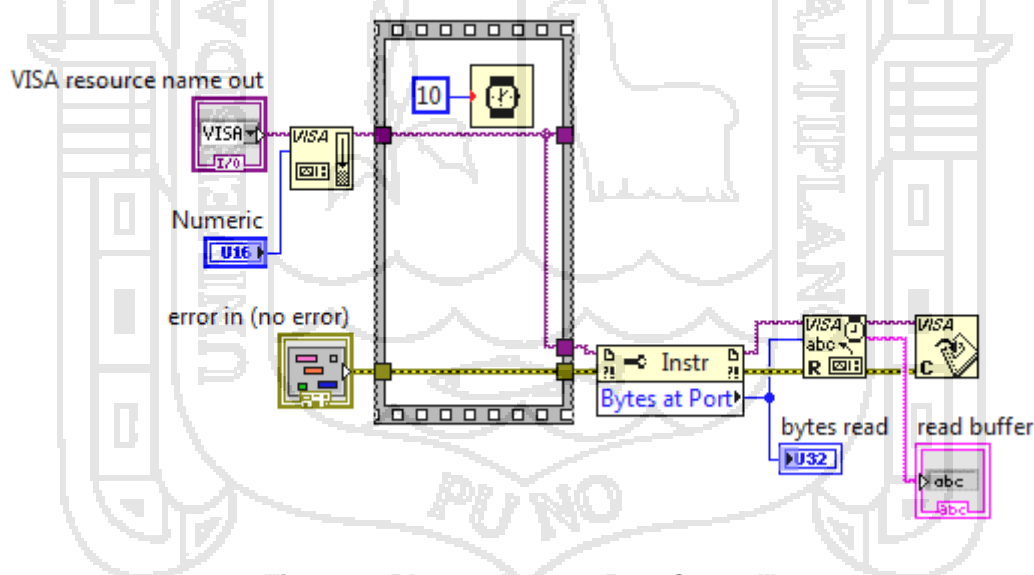


Figura 79 Bloque “Lectura Byte Control”.

En este bloque se vacía el buffer de lectura y se espera un tiempo de seguridad de 10ms para leer correctamente el dato. Una vez leído el dato se cierra la comunicación para que el puerto seleccionado pueda ser utilizado por otros dispositivos si fuese necesario.

CAPÍTULO VI: EXPOSICION Y ANALISIS DE LOS RESULTADOS



EXPOSICION Y ANALISIS DE LOS RESULTADOS

6.1. DETERMINACIÓN DE LOS VALORES DE LA MATRIZ T

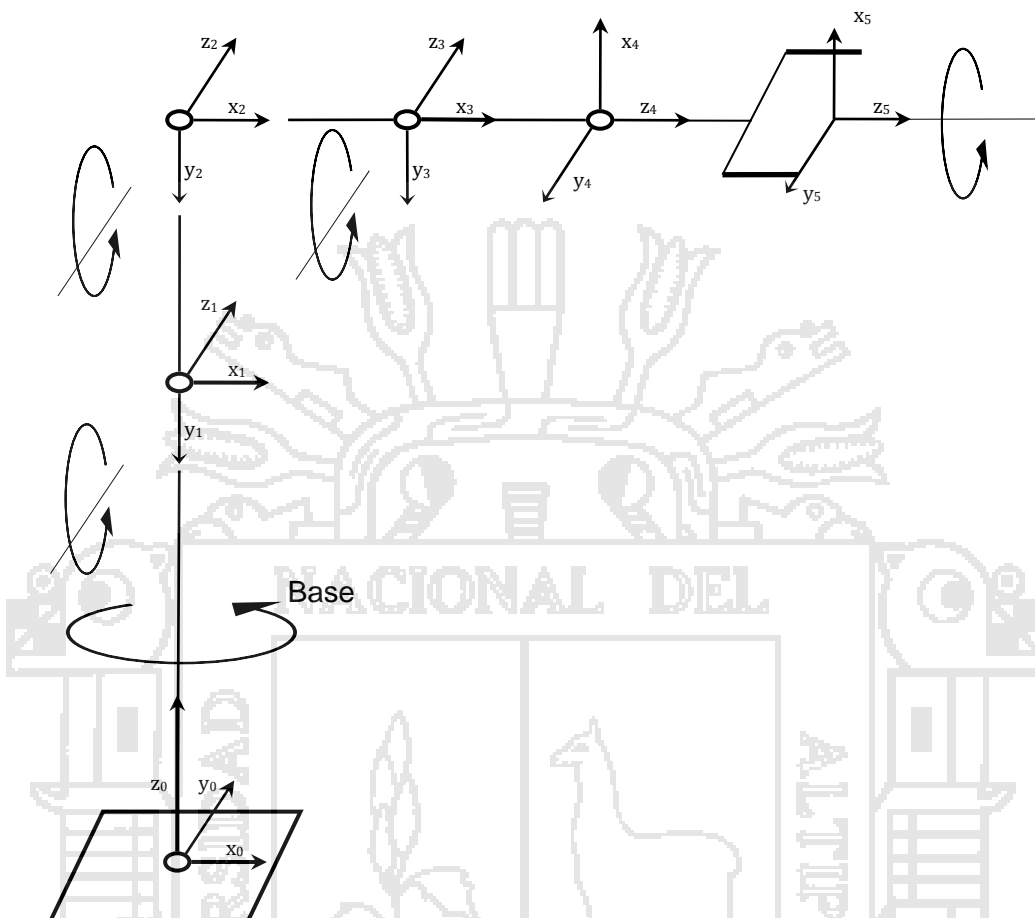


Figura 80 Análisis de los cinco grados de libertad para determinar valores de matriz

homogénea T

$${}^0A_1 = \begin{pmatrix} \cos_1 & 0 & \text{sen}_1 & 0 \\ \text{sen}_1 & 0 & \cos_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$${}^1A_2 = \begin{pmatrix} \text{sen}_1 & \cos_2 & 0 & d_2 \\ \cos_1 & \text{sen}_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Articulación	θ	d	a	α
1	q1	d1	0	-90

2	q2	0	d2	0
3	q3	0	d3	0
4	q4-90	0	d4	-90
5	q5	d5	0	0

$${}^2A_3 = \begin{pmatrix} \cos_3 & -\text{sen}_3 & 0 & d_3 \\ \text{sen}_3 & \cos_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$${}^3A_4 = \begin{pmatrix} \text{sen}_4 & 0 & \cos_4 & d_4 \text{sen}_4 \\ -\cos_4 & 0 & \text{sen}_4 & -d_4 \cos_4 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$${}^4A_5 = \begin{pmatrix} \cos_5 & -\text{sen}_5 & 0 & 0 \\ \text{sen}_5 & \cos_5 & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Para obtener la matriz de transformación (T) entre la base y el extremo del robot hay que multiplicar por las diferentes matrices de transformación entre el sistema 0 y el sistema 6. Se procederá de la siguiente forma:

$$T = {}^1A_2 \cdot {}^2A_3 \cdot {}^3A_4 \cdot {}^4A_5 = {}^0A_5$$

Entonces la matriz de transformación homogénea quedaría de la siguiente forma:

$$T = \begin{pmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & w \end{pmatrix}$$

$$(1, 1) = \cos_5 \text{sen}_4 \cos_1 \cos_2 \cos_3 - \cos_5 \text{sen}_4 \cos_1 \text{sen}_2 \text{sen}_3 + \cos_1 \cos_2 \cos_4 \cos_5 \text{sen}_3 + \cos_1 \text{sen}_2 \cos_3 \cos_4 \cos_5 + \text{sen}_1 \text{sen}_5$$

$$(1, 2) = -\text{sen}_5 \text{sen}_4 \cos_1 \cos_2 \cos_3 + \text{sen}_4 \text{sen}_5 \cos_1 \text{sen}_2 \text{sen}_3 - \cos_1 \cos_2 \cos_4 \text{sen}_5 \text{sen}_3 + \cos_1 \text{sen}_2 \cos_3 \cos_4 \text{sen}_5 + \text{sen}_1 \cos_5$$

$$(1, 3) = \cos_4 \cos_1 \cos_2 \cos_3 - \cos_4 \cos_1 \text{sen}_2 \text{sen}_3 - \cos_1 \cos_2 \cos_4 \text{sen}_3 - \cos_1 \text{sen}_2 \cos_3 \text{sen}_4$$

$$(1, 4) = d_4 \text{sen}_4 \text{cos}_1 \text{cos}_2 \text{cos}_3 - d_4 \text{sen}_4 \text{cos}_1 \text{sen}_2 \text{sen}_3 + d_4 \text{cos}_1 \text{cos}_2 \text{cos}_4 \text{sen}_3 + d_3 \text{cos}_1 \text{cos}_2 + d_2 \text{cos}_1$$

$$(2, 1) = \text{cos}_5 (\text{sen}_1 \text{sen}_4 \text{cos}_2 \text{cos}_3 - \text{sen}_4 \text{sen}_1 \text{sen}_2 \text{sen}_3 + \text{cos}_2 \text{cos}_4 \text{sen}_1 \text{sen}_3 + \text{sen}_1 \text{cos}_3 \text{cos}_4 \text{sen}_2) + \text{sen}_1 \text{sen}_5$$

$$(2, 2) = -\text{sen}_5 (\text{sen}_1 \text{sen}_4 \text{cos}_2 \text{cos}_3 - \text{sen}_4 \text{sen}_1 \text{sen}_2 \text{sen}_3 + \text{cos}_2 \text{cos}_4 \text{sen}_1 \text{sen}_3 + \text{sen}_1 \text{cos}_3 \text{cos}_4 \text{sen}_2) + \text{sen}_1 \text{cos}_5$$

$$(2, 3) = \text{sen}_1 \text{cos}_4 \text{cos}_2 \text{cos}_3 - \text{cos}_4 \text{sen}_1 \text{sen}_2 \text{sen}_3 - \text{cos}_2 \text{sen}_4 \text{sen}_1 \text{sen}_3 - \text{sen}_1 \text{sen}_2 \text{sen}_4 \text{cos}_3$$

$$(2, 4) = d_4 \text{sen}_1 \text{sen}_4 \text{cos}_2 \text{cos}_3 - d_4 \text{sen}_4 \text{sen}_1 \text{sen}_2 \text{sen}_3 + d_4 \text{cos}_2 \text{cos}_4 \text{sen}_1 \text{sen}_3 + d_4 \text{sen}_1 \text{sen}_3 \text{cos}_2 \text{cos}_4 + d_3 \text{sen}_1 \text{cos}_2 + d_2 \text{sen}_1$$

$$(3, 1) = \text{sen}_1 (-\text{sen}_4 \text{sen}_2 \text{cos}_3 - \text{sen}_4 \text{cos}_2 \text{sen}_3 + \text{sen}_2 \text{cos}_4 \text{sen}_3 + \text{cos}_2 \text{cos}_3 \text{cos}_4)$$

$$(3, 2) = -\text{sen}_5 (-\text{sen}_4 \text{sen}_2 \text{cos}_3 - \text{sen}_4 \text{cos}_2 \text{sen}_3 - \text{sen}_2 \text{cos}_4 \text{sen}_3 + \text{cos}_2 \text{cos}_3 \text{cos}_4)$$

$$(3, 3) = (-\text{cos}_4 \text{sen}_2 \text{cos}_3 - \text{cos}_4 \text{cos}_2 \text{sen}_3 + \text{sen}_2 \text{sen}_3 \text{sen}_4 - \text{cos}_2 \text{sen}_4 \text{cos}_3)$$

$$(3, 4) = d_4 (-\text{sen}_4 \text{sen}_2 \text{cos}_3 - \text{sen}_4 \text{cos}_2 \text{sen}_3 - \text{sen}_2 \text{cos}_4 \text{sen}_3 + \text{cos}_2 \text{cos}_3 \text{cos}_4) - d_3 \text{sen}_2 + d_1$$

$$(4, 1) = 0$$

$$(4, 2) = 0$$

$$(4, 3) = 0$$

$$(4, 4) = 1$$

6.2. ACTIVACION DEL SISTEMA EN EL SERVIDOR

Para comprobar el funcionamiento entre el computador que realizara la función de servidor, y el brazo robótico una vez conectado al nodo ZigBee y la tarjeta de control con el microcontrolador PIC16F877A, se pondrá en funcionamiento el sistema sin conectar al PC cliente y determinar la respuesta de los servomotores cuando las articulaciones se pongan en movimiento y determinar la relación entre el ángulo recorrido y el ángulo medido por los sensores.

Así en el microprograma principal se probara cada articulación y se registraran estos valores para determinar la precisión del sistema teniendo en cuenta el ángulo máximo que puede recorrer cada articulación.

Para poner en funcionamiento todo el sistema se inició por conectar las tarjetas al brazo robótico y alimentar cada una de ellas, seguidamente la conectorización a las E/S, y por último la conexión de las señales provenientes de la interfaz teniendo en cuenta los voltajes de referencia de estas señales.

6.3. PRUEBAS DE FUNCIONAMIENTO DEL BRAZO ROBOT

A pesar de que para dicho proceso no se requiere un control en lazo cerrado, se optó durante el proceso de diseño mediante la aplicación de LabVIEW, para monitorizar y poder recibir una señal que represente la posición de las articulaciones del robot.

Se realizaron pruebas con la articulación del codo y la inclinación de la muñeca adhiriendo al microprograma desarrollado en la estación.

La prueba en mención consiste en comparar los valores establecidos en coordenadas rectangulares de la posición del efector final del modo operación COORDENADAS xyz (en mm) con las coordenadas rectangulares obtenidas mediante el uso de la aplicación Math Script que ya se utilizó para validar el modelo matemático.

La aplicación de control del HMI en modo COORDENADAS xyz (mm) basada en el uso de la herramienta Inverse Kinematics, evalúa los valores de las coordenadas rectangulares del efector final y obtiene como resultado los correspondientes valores articulares del manipulador que es el objetivo de la resolución del modelo cinemático inverso.

Dichos valores angulares obtenidos son introducidos en la aplicación Math Script que resuelve el problema cinemático directo, con la evaluación de las coordenadas angulares obtenemos los correspondientes valores rectangulares de la posición en la que se encuentra el extremo del efector final.

La comparación de las coordenadas rectangulares de la posición del efector final entre las dos aplicaciones anteriormente mencionadas se muestra en la tabla 3, se realizó la prueba para siete valores de interés.

Coordenadas xyz (mm) (Inverse Kinematics)			Aplicación con Math Script			Error relativo promedio (%)	
Coordenadas rectangulares		Configuración angular	Coordenadas rectangulares				
x	y	Z	$(\theta_1, \theta_2, \theta_3, \theta_4)$	x	y	z	
316	0	214.5	$(0^\circ, 90^\circ, -90^\circ, 0^\circ)$	315	0	214	0.183
268	167	100	$(32^\circ, 76^\circ, -112^\circ, 36^\circ)$	267	166.561	100-395	0.344
301	-130	78	$(-23^\circ, 66^\circ, -108^\circ, 42^\circ)$	300	-127.298	77-0923	1.191
201	-98	262	$(-26^\circ, 123^\circ, -101^\circ, 22^\circ)$	200	-97.7691	260.445	0.442
275	223	158	$(39^\circ, 73^\circ, -82^\circ, 0^\circ)$	275	222.291	158.431	0.197
312	0	0	$(0^\circ, 40^\circ, -107^\circ, 71^\circ)$	312	0	0.346098	0.000
150	-92	268	$(-32^\circ, 138^\circ, -104^\circ, -35^\circ)$	149	-93.3551	268.113	0.727

Tabla 3: Error relativo promedio de posicionamiento

6.4. VALIDACION DE LA HIPOTESIS

La hipótesis general planteada para el proyecto es:

“Desarrollar y construir el módulo de entrenamiento de un brazo robótico de cinco grados de libertad, diseñado en una escala de 1 a 100, implementado en LabView monitorizado mediante redes ZigBee, para emular tareas de paletización, tanto en modo manual como en modo automático”.

Se utiliza una prueba de hipótesis para las proporciones (Ordoñez, 2009), en base a los resultados visuales que se obtuvieron considerando:

Los resultados indican que de 55 piezas para paletizar, el prototipo de brazo robótico de imágenes paletizó 52, entonces se puede afirmar que al menos el 96% de las piezas pudieron ser paletizadas, es decir:

$$n = 55$$

$$x = 52$$

$$p = \frac{52}{55}$$

$$p_0 = 96\% \text{ ó } 0.96$$

Se asume una confiabilidad de 95%, el valor correspondiente a Z en la distribución normal es -1.64

$$\alpha = 0.05 \rightarrow z = -1.645$$

Hipótesis:

$$H_0 = p > p_0$$

$$H_1 = p < p_0$$

$$z_0 = \frac{\frac{x}{n} - p_0}{\sqrt{\frac{p_0(1-p_0)}{n}}}$$

$$z_0 = \frac{\frac{52}{55} - 0.96}{\sqrt{\frac{0.96(1-0.96)}{55}}} = -0.55048188$$

$$\therefore -0.55048188 > -1.645$$

Lo que sugiere que: “No hay razón para rechazar la hipótesis H_0 ”

Por consiguiente, con una confiabilidad del 95% se concluye que al menos el 96% de las piezas pueden ser paletizadas por el prototipo de brazo robot diseñado e implementado

6.5. ANÁLISIS TÉCNICO-ECONÓMICO

N°	Cant.	Ítem	Descripción	Características	Precio unitario	Importe Total
1	2	J3.0	Conector macho contactos	Diámetro 5mm	1.85	3.70

2	2	J3.1	Conector hembra 2 contactos	Diámetro 5mm	1.85	3.70
3	2	SW8	Interruptor de 2 patillas	Interruptor simple	2.10	4.20
4	2	D3	Diodo	1A, 50V	1.10	2.20
5	2	C3, CC4	Condensador	1.5 μ F, 25 V, aluminio	1.08	2.16
6	1	OSC1	Oscilador	30MHz 5V	5.45	5.45
7	2	C1, C2	Condensador	22pF, 25V cerámico disco	1.03	2.06
8	1	RV1	Potenciómetro	10 K Ω . 5%, 1/8W	3.50	3.50
9	1	R1	Resistencia	30 K Ω . 5%, 1/8W	1.03	1.03
10	5	SW1, SW2, SW3, SW4, SW5	Micropulsador	4mm, 12V	1.10	5.50
11	3	SW6, SW7 RESET	Micropulsador	10mm, 12V	1.10	3.30
12	7	R2, R3, R4, R5, R6, R7, R8	Resistencia	1 K Ω . 5%, 1/8W	1.03	7.21
13	1	R11, R12	Resistencia	10 K Ω . 5%, 1/8W	1.03	1.03
14	1	U1	Microcontrolador	PIC 16F877A	20.20	20.20
15	2	J2	Conector	Tira de patillas, 0.5mm con patilla torneada	5.25	10.50
16	1	J1	Conector	Conector serie DB9	1.50	1.50
17	1	D1	LED	3mm, 25° verde	1.40	1.40
18	1	D2	LED	3mm. 25° rojo	1.20	1.20
19	2	R9. R10	Resistencia	150 Ω . 5%, 1/8W	1.03	2.06

20	1	U2	Circuito integrado	Max232	5.67	5.67
21	4	C5, C6, C7	Condensador electrolítico	1 μ F, 25V electrolítico	1.10	4.40
22	1	PCI	Placa perforada	Contactos metálicos por una cara 10x10cm	20.00	20.00
23	3	SV1, SV2, SV3	Servomotor	HitechHS-422	60.00	180.00
24	3	SV4, SV5, SV6	Servomotor	Estándar Hitech	50.00	150.00
25	1	MEC	Estructura mecánica	Piezas de acrílico para construir la parte mecánica	200.00	200.00
26	3	ZigBee	Módulos de emisión, recepción y USB ZigBee	Transmisión inalámbrica	95.00	285.00
27	1	Autodesk	Entorno de desarrollo (aplicación Windows)	Desarrollo	0.00	0.00
28	1	PicKit 2	Depurador/grabador para μ C	Desarrollo	170.00	170.00
29	1	Cable Serie-USB	Cable de comunicación	Desarrollo	20.00	20.00
30	1	Soldador	Equipo de laboratorio	Desarrollo	40.00	40.00
31	1	Desoldador	Equipo de laboratorio	Desarrollo	12.00	12.00
32	1	Osciloscopio	Equipo de laboratorio	Desarrollo	700.00	700.00
33	3 (meses)	Internet	Conexión ADSL para búsqueda de información en la red	Desarrollo	130.00	390.00
34	3 (meses)	Energía eléctrica	Para el funcionamiento de los diversos dispositivos y pruebas	Desarrollo	30.00	90.00
35	1	LabView 2013	Aplicación de instrumentación virtual	Desarrollo	4350.00	4350.00
36	1	Ordenador	Computador personal para desarrollo	Desarrollo	900.00	900.00
37	1	Juego de destornillad	Paquete de destornillador con distintas puntas de	Desarrollo	20.00	20.00

		ores	precisión			
38	1	Alimentación	Fuente de alimentación regulable	Desarrollo	70.00	70.00
39	4	Impresión	Tinta de impresora	Desarrollo	90.00	360.00
40	2 millares	Papel bond	Para impresiones de informes finales	Desarrollo	15.00	30.00
41	6	Recargas	Recargas virtuales para terminales móviles	Desarrollo	10.00	60.00
42	2	Pendrive	Memorias flash para llevar información	Desarrollo	20.00	40.00
TOTAL						8438.97

Tabla 4: Análisis técnico

Se abordan temas referentes a los gastos que el proyecto implica en cuanto a componentes necesarios, así como los gastos de ingeniería e implementación del mismo.

En la Tabla 4 se presenta el costo de los componentes necesarios para realizar la implementación del prototipo de clasificación y paletizado con robot.

Para la estimación de costos de Ingeniería, así como del software; se aplica la regla de Cocomo (Gavilánez y Albán, 2007) la cual se expresa en la ecuación 5.1.

$$\text{Costo}(\text{Nuevos Soles}) = k * N_{\text{horas persona}}$$

K = Valor hora profesional en Nuevos Soles. Estimado en 40.00 N.S.

Para calcular el costo de la mano de obra del montaje e implementación, se utiliza el mismo criterio, pero asignando a K un valor de 4.00 N.S.

En la Tabla 4, evidencia el costo de inversión del proyecto en cuanto a personal, considerando que para el rubro de ingeniería interviene una persona con una carga laboral neta de 30 días y 8 horas diarias.

Para el montaje e implementación, se requiere de una persona con una carga laboral de 25 días y 8 horas diarias.

Ítem	Descripción	K (N.S.)	N (horas de trabajo)	Subtotal
1	Ingeniera del proyecto	40.00	240	9600.00
2	Montaje e implementación	4.00	200	800.00
			TOTAL	10400.00

Tabla 5: Costo de realización

En base a los resultados de la Tabla 4 y la Tabla 5, el costo de inversión del proyecto “Desarrollo e implementación de un módulo de entrenamiento de brazo robótico de cinco grados de libertad monitorizado mediante redes ZigBee” es de 18838.97 nuevos soles.

CONCLUSIONES

La base del manipulador debe concentrar la mayor inercia posible para darle la rigidez necesaria a la estructura, cuando esta se encuentra levantando una carga en movimiento.

En este trabajo desarrollado a manera de prototipo se puede apreciar la gran importancia y potencialidad del campo de la teleoperación en aplicaciones que involucran la interacción del ser humano con materiales peligrosos que atenten de alguna manera a la integridad de los mismos.

Al ser el brazo robótico un sistema didáctico, este presenta varias condiciones que influyen en el error de posicionamiento mediante el sistema de control basado en la interfaz humano maquina, entre las que están de manera general; imperfecciones en la estructura mecánica, diseño mecánico impreciso, servomotores de modelismo.

De manera general, la dificultad de teleoperar el robot depende de la capacidad del operador para manejar el sistema, si el operador no conoce completamente la forma de uso y no ha practicado su uso, tendrá mas difícil la tarea de manipulación y monitoreo.

RECOMENDACIONES

Se recomienda que se realicen mas proyectos con otro tipo de objetivos pudiendo tomar como referencia este proyecto

Se deben determinar los sistemas de referencia siguiendo criterios establecidos por un determinado autor ya que existen distintos criterios para realizar la asignación de los sistemas de referencia.

La correcta asignación y representación de los sistemas de referencia al brazo robótico permitirá una correcta resolución de la matriz de transformación homogénea.

Para mejorar el desempeño del presente proyecto se sugiere experimentar con módulos diferentes.

La Escuela Profesional de Ingeniería Electrónica debería realizar convenios con más instituciones, tanto públicas como privadas, para que sus estudiantes puedan realizar mayores investigaciones, en las tres especialidades por igual, lo mismo que se debería alentar a los estudiantes para poder realizarlas.

BIBLIOGRAFIA

- [1].Barrientos, Antonio (2007). Fundamentos de la Robótica (2da edición)
España: McGraw-Hill.
- [2].Ollero Baturone, Aníbal (2001). Robótica, manipuladores y robots
móviles (1ra ed.) España: Marcombo Boixareu Editores.
- [3].Craig, John J. (2006). Robótica (1ra ed.) Stanford University: Pearson
Prentice Hall.
- [4].Craig, John J. (2008). Introduction to robotics: Mechanics and control
(3rd ed.) Stanford University: Pearson Prentice Hall.
- [5].Vivas Alban, Oscar Andres (2010) Diseño y control de robots
industriales: teoría y práctica (1ra ed.) Colombia: Elaleph.com S.R.L.
- [6].Palacios, Enrique & Remiro, Fernando & López, Lucas (2009)
Microcontrolador PIC16F84, desarrollo de proyectos (3ra ed.) España:
Alfaomega RA-MA
- [7].Gislason, Drew (2008). Zigbee wireless networking (1st ed.) Oxford,
Reino Unido: Newnes-Elsevier.
- [8].Mendoza Sánchez Eduardo Augusto (2004) Control de un Robot
Manipulador (tesis de Maestría), Universidad de las Américas, Puebla-
México.
- [9].González, Víctor R (2007) Curso 2007: Fundamentos de Robótica
(disertación doctoral) Centro de Formación del Profesorado e Innovación
Educativa Valladolid II, Valladolid-España.
- [10]. Parra Márquez, Juan Carlos & Cid Cifuentes, Karina Pilar (2007);
Implementación de Robot Scorboter-vplus, para docencia, en realidad
virtual, Revista Digital Universitaria UNAM, 8 (5) ,14-19.