



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

## **EXTENSIÓN LATACUNGA**

### **DEPARTAMENTO DE ENERGÍA Y MECÁNICA**

**“DISEÑO E IMPLEMENTACIÓN DE UN GUANTE  
SENSORIZADO PARA EL CONTROL TELEOPERADO DE UN  
PROTOTIPO DE BRAZO ROBÓTICO PARA APLICACIONES  
DE MANEJO DE MATERIALES PELIGROSOS.”**

**EDISON FERNANDO HERRERA NÚÑEZ  
WELLINGTON OSWALDO ZAMBRANO CHÁVEZ**

**TESIS PRESENTADA COMO REQUISITO PREVIO A LA  
OBTENCIÓN DEL GRADO DE:**

**INGENIERO EN MECATRÓNICA**

**Año 2013**

**UNIVERSIDAD DE LA FUERZAS ARMADAS-ESPE**  
**EXTENSIÓN LATACUNGA**  
**CARRERA DE INGENIERÍA MECATRÓNICA**

**DECLARACIÓN DE RESPONSABILIDAD**

**EDISON FERNANDO HERRERA NÚÑEZ**  
**WELLINGTON OSWALDO ZAMBRANO CHÁVEZ**

**DECLARAMOS QUE:**

El proyecto de grado denominado **“DISEÑO E IMPLEMENTACIÓN DE UN GUANTE SENSORIZADO PARA EL CONTROL TELEOPERADO DE UN PROTOTIPO DE BRAZO ROBÓTICO PARA APLICACIONES DE MANEJO DE MATERIALES PELIGROSOS”**, ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas constan al pie de las páginas correspondientes, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance científico del proyecto de grado en mención.

Latacunga, Noviembre del 2013.

---

Edison Herrera N.

---

Wellington Zambrano C.

**UNIVERSIDAD DE LA FUERZAS ARMADAS-ESPE**

**EXTENSIÓN LATACUNGA**

**CARRERA DE INGENIERÍA MECATRÓNICA**

**CERTIFICADO**

ING. FRANKLIN SILVA (DIRECTOR)

ING. EDWIN PRUNA (CODIRECTOR)

**CERTIFICAN:**

Que el trabajo titulado “**DISEÑO E IMPLEMENTACIÓN DE UN GUANTE SENSORIZADO PARA EL CONTROL TELEOPERADO DE UN PROTOTIPO DE BRAZO ROBÓTICO PARA APLICACIONES DE MANEJO DE MATERIALES PELIGROSOS**”, realizado por: Edison Fernando Herrera Núñez Y Wellington Oswaldo Zambrano Chávez, ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas de la Universidad de las Fuerzas Armadas-ESPE.

Debido a que constituye un trabajo de excelente contenido científico que coadyuvará a la aplicación de conocimientos y al desarrollo profesional, **SI** recomiendan su publicación.

El mencionado trabajo consta de UN documento empastado y UN disco compacto el cual contiene los archivos en formato portátil de Acrobat (pdf). Autorizan, a los señores: EDISON FERNANDO HERRERA NÚÑEZ y WELLINGTON OSWALDO ZAMBRANO CHÁVEZ que lo entreguen al Ing. Fausto Acuña, en su calidad de Director de la Carrera de Ingeniería Mecatrónica.

Latacunga, Noviembre del 2013.

---

Ing. Franklin Silva  
DIRECTOR

---

Ing. Edwin Pruna  
CODIRECTOR

**UNIVERSIDAD DE LA FUERZAS ARMADAS-ESPE**  
**EXTENSIÓN LATACUNGA**  
**CARRERA DE INGENIERÍA MECATRÓNICA**

**AUTORIZACIÓN**

**Nosotros**, Edison Fernando Herrera Núñez y Wellington Oswaldo Zambrano Chávez.

Autorizamos a la UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE, la publicación en la biblioteca virtual de la Institución, del trabajo: **“DISEÑO E IMPLEMENTACIÓN DE UN GUANTE SENSORIZADO PARA EL CONTROL TELEOPERADO DE UN PROTOTIPO DE BRAZO ROBÓTICO PARA APLICACIONES DE MANEJO DE MATERIALES PELIGROSOS”**, cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y autoría.

Latacunga, Noviembre del 2013.

---

Edison Herrera N.

---

Wellington Zambrano C.

## DEDICATORIA

*Dedico el presente trabajo con todo mi amor a mis padres Fausto, María y a mis queridas hermanas Diana y Katty, por su infinita muestra de cariño y comprensión.*

*Edison.*

*Esta tesis la dedico con todo cariño y amor a las personas que me dieron la vida, mi mamá Libia y mi papá Oswaldo, ya que me han brindado su confianza y apoyo incondicional en todo momento y han significado el pilar fundamental para ser la persona de bien que soy hoy.*

*Wellington.*

## AGRADECIMIENTO

*Mi agradecimiento más sincero a Dios por ser luz en mi camino y ayudarme a superar cada una de las dificultades en mi vida universitaria.*

*A mis amados padres y hermanas que sin pedir nada a cambio lo han dado todo por mí, gracias por estar presentes más en los momentos no tan gratos, en los cuales sin su ayuda estuve a punto de desfallecer.*

*A toda mi familia, amigos y amigas, por ser ejemplo de unidad, fortaleza y por estar junto a mí cuando lo necesite.*

*De igual forma doy gracias a mi amigo y compañero de tesis, por toda la paciencia y apoyo brindado para cristalizar este proyecto.*

*A todas aquellas personas que se cruzaron en mi vida y que de una u otra forma ayudaron a cumplir mi objetivo.*

*Mi eterna gratitud a todos, mil gracias...*

*Edison.*

*Doy gracias principalmente a Dios por permitir que cumpla esta meta y por entregarme la fuerza y la inteligencia necesaria para superar los obstáculos durante mi carrera. Agradezco a mi familia por nunca haber dejado de confiar en mí; a mi mamá por brindarme todo ese cariño y amor que me ha servido para seguir con ganas el día a día como estudiante; a mi padre por brindarme su ejemplo en rectitud y valores que me permitieron ser mejor cada día; a mis hermanos Karen y Alex por su cariño y compañía que con los momentos compartidos aliviaban mi mente de las dificultades encontradas.*

*Gracias también a los docentes de la Escuela Politécnica del Ejército por todo su conocimiento impartido; a mis tutores de tesis por su aporte en la finalización de este trabajo; a mi amigo y compañero Edison por su gran esfuerzo y apoyo durante el desarrollo y culminación de esta tesis.*

*A todos aquellos que de alguna forma colaboraron en la realización de esta meta.*

*De corazón... muchísimas gracias.*

*Wellington.*



# ÍNDICE DE CONTENIDOS

<b>PORTADA</b>	<b>i</b>
<b>DECLARACIÓN DE RESPONSABILIDAD</b>	<b>ii</b>
<b>CERTIFICACIÓN</b>	<b>iii</b>
<b>AUTORIZACIÓN</b>	<b>iv</b>
<b>DEDICATORIA</b>	<b>v</b>
<b>AGRADECIMIENTO</b>	<b>vii</b>
<b>ÍNDICE DE CONTENIDOS</b>	<b>ix</b>
<b>RESUMEN</b>	<b>xviii</b>
<b>ABSTRACT</b>	<b>xix</b>
<b>PRESENTACIÓN</b>	<b>xx</b>
<b>CAPÍTULO I</b>	<b>1</b>
<b>FUNDAMENTOS TEÓRICOS</b>	<b>1</b>
<b>1.1 EL BRAZO HUMANO</b>	<b>1</b>
<b>1.2 MANIPULADORES ROBÓTICOS</b>	<b>2</b>
1.2.1 ROBÓTICA	2
1.2.2 ROBOT INDUSTRIAL MANIPULADOR	3
1.2.3 CINEMÁTICA DEL ROBOT	11
<b>1.3 SISTEMAS DE TELEOPERACIÓN</b>	<b>18</b>
1.3.1 TELEOPERACIÓN	18
1.3.2 ELEMENTOS DE UN SISTEMA DE TELEOPERACIÓN	19
1.3.3 TELERROBÓTICA	20
1.3.4 INTERFACES DE TELEOPERACIÓN	21
1.3.5 CONTROL EN LA TELEOPERACIÓN	25
<b>1.4 GUANTES SENSORIZADOS</b>	<b>29</b>
1.4.1 DATAGLOVE	30
1.4.2 CYBERGLOVE	30
1.4.3 POWERGLOVE	31

1.4.4	DEXTEROUS HAND MASTER	32
1.4.5	ACCELEGLOVE	32
<b>1.5</b>	<b>SENSORES INERCIALES, FLEXORESISTIVOS Y DE PRESIÓN</b>	<b>33</b>
1.5.1	SENSORES INERCIALES	33
1.5.2	SENSORES FLEX	38
1.5.3	SENSORES DE FUERZA FSR	40
<b>1.6</b>	<b>MICROCONTROLADOR</b>	<b>42</b>
1.6.1	LOS MICROCONTROLADORES PIC DE MICROCHIP	43
<b>1.7</b>	<b>TRANSMISIÓN DE DATOS</b>	<b>44</b>
1.7.1	SISTEMAS DE COMUNICACIÓN ELÉCTRICOS ELECTRÓNICOS	44
1.7.2	ELEMENTOS DE UN SISTEMA DE COMUNICACIÓN ELECTRÓNICO	44
1.7.3	COMUNICACIONES RADIOELÉCTRICAS	47
1.7.4	COMPARACIÓN DE LAS TECNOLOGÍAS INALÁMBRICAS	48
1.7.5	ZIGBEE (IEEE 802.15.4)	50
<b>1.8</b>	<b>INTERFAZ HUMANO MÁQUINA</b>	<b>52</b>
1.8.1	SOFTWARE HMI	52
1.8.2	TIPOS DE HMI (INTERFAZ HUMANO MÁQUINA)	52
1.8.3	LABVIEW	53
<b>1.9</b>	<b>MATERIALES PELIGROSOS</b>	<b>56</b>
1.9.1	CLASIFICACIÓN DE LOS MATERIALES PELIGROSOS	56
<b>CAPÍTULO II</b>		<b>61</b>
<b>DISEÑO E IMPLEMENTACIÓN</b>		<b>61</b>
<b>2.1</b>	<b>DESCRIPCIÓN DEL PROYECTO</b>	<b>61</b>
<b>2.2</b>	<b>ESTACIÓN LOCAL</b>	<b>62</b>
2.2.1	ARQUITECTURA DEL SISTEMA	62
2.2.2	ACELERÓMETRO MMA7361L	63
2.2.3	SENSOR FLEXO RESISTIVO	70
2.2.4	MICROCONTROLADOR PIC18F452	72
2.2.5	MÓDULO XBEE	74
2.2.6	GUANTE	75
2.2.7	NUNCHUK	80
2.2.8	PLACA ELECTRÓNICA PCI 1	81

<b>2.3</b>	<b>ESTACIÓN REMOTA</b>	<b>86</b>
2.3.1	ARQUITECTURA DEL SISTEMA	86
2.3.2	BRAZO ROBÓTICO LYNXMOTION AL5D	87
2.3.3	TARJETA CONTROLADORA DE SERVOS SSC16 POLOLU	91
2.3.4	SENSOR FSR 402	97
2.3.5	PLACA ELECTRÓNICA PCI 2	99
<b>2.4</b>	<b>ESTRUCTURA DE CONTROL</b>	<b>107</b>
2.4.1	SOFTWARE DE CONTROL DE LA ESTACIÓN LOCAL	107
2.4.2	SOFTWARE DE CONTROL DE LA ESTACIÓN REMOTA	113
<b>2.5</b>	<b>TRANSMISIÓN INALÁMBRICA DE DATOS</b>	<b>125</b>
2.5.1	MODO DE FUNCIONAMIENTO DE LOS MÓDULOS XBEE	126
2.5.2	DIRECCIONAMIENTO DE LOS MÓDULOS XBEE	126
<b>2.6</b>	<b>INTERFAZ HUMANO MÁQUINA</b>	<b>128</b>
2.6.1	HERRAMIENTAS USADAS EN LA APLICACIÓN	129
2.6.2	DESCRIPCIÓN DEL HMI	132
<b>2.7</b>	<b>IMPLEMENTACIÓN DE LOS COMPONENTES FÍSICOS</b>	<b>144</b>
2.7.1	ESTRUCTURA MECÁNICA DEL ENTORNO DE TRABAJO DEL ROBOT	145
2.7.2	MONTAJE DEL HARDWARE GUANTE DE DATOS (ESTACIÓN LOCAL)	146
2.7.3	ZONA DE TRABAJO	147
2.7.4	HERRAMIENTAS DE TRABAJO	148
<b>CAPÍTULO III</b>		<b>150</b>
<b>PRUEBAS Y RESULTADOS</b>		<b>150</b>
<b>3.1</b>	<b>PRUEBAS DE FUNCIONAMIENTO DEL CONTROL CON USO DE REALIMENTACIÓN</b>	<b>150</b>
<b>3.2</b>	<b>PRUEBAS DE TELEOPERACIÓN MEDIANTE EL GUANTE SENSORIZADO</b>	<b>152</b>
3.2.1	PRUEBAS DE POSICIONAMIENTO MEDIANTE EL GUANTE DE DATOS	152
3.2.2	PRUEBAS DE MANIPULACIÓN DE OBJETOS	158
<b>3.3</b>	<b>PRUEBAS DE POSICIONAMIENTO MEDIANTE EL HMI</b>	<b>162</b>
3.3.1	PRUEBAS DE RESOLUCIÓN DEL MODELO CINEMÁTICO	162

3.3.2	PRUEBAS DEL SISTEMA DE CONTROL DE POSICIÓN _____	169
<b>3.4</b>	<b>PRUEBAS DE TELEOPERACIÓN A TRAVÉS DE CÁMARAS _____</b>	<b>174</b>
3.4.1	PRUEBAS DE FUNCIONAMIENTO DE CÁMARAS _____	174
3.4.2	PRUEBAS DE TELEOPERACIÓN A TRAVÉS DE VIDEO _____	176
<b>3.5</b>	<b>ANÁLISIS TÉCNICO-ECONÓMICO _____</b>	<b>182</b>
<b>CAPÍTULO IV _____</b>		<b>184</b>
<b>CONCLUSIONES Y RECOMENDACIONES _____</b>		<b>184</b>
4.1	CONCLUSIONES _____	184
4.2	RECOMENDACIONES _____	187
<b>BIBLIOGRAFÍA _____</b>		<b>189</b>
<b>GLOSARIO _____</b>		<b>193</b>
<b>ANEXOS _____</b>		<b>200</b>

## ÍNDICE DE FIGURAS

Figura 1.1: Elementos estructurales de un robot industrial. _____	5
Figura 1.2: Semejanza de un brazo manipulador con la anatomía humana. _____	6
Figura 1.3: Los seis pares inferiores de Reuleaux. _____	6
Figura 1.4: Distintos grados de libertad (GDL) de un brazo de robot. _____	7
Figura 1.5: Configuración Cartesiana. _____	8
Figura 1.6: Configuración Cilíndrica. _____	9
Figura 1.7: Configuración Esférica. _____	9
Figura 1.8: Configuración Angular o Antropomórfica. _____	10
Figura 1.9: Configuración SCARA. _____	10
Figura 1.10: Diagrama de relación entre cinemática directa e inversa. _____	11
Figura 1.11: Esquema general de un sistema de operación. _____	19
Figura 1.12: Elementos de un sistema de teleoperación. _____	20
Figura 1.13: Brazo Maestro Mini-master. _____	22
Figura 1.14: Joystick con realimentación de fuerzas. _____	22
Figura 1.15: Space Mouse. _____	23
Figura 1.16: Guante sensorizado. _____	23
Figura 1.17: Casco de realimentación visual. _____	24
Figura 1.18: Sistema de realimentación háptica, caso particular de cinestésica. _____	27
Figura 1.19: Esquema gráfico de teleoperación con control bilateral. _____	28
Figura 1.20: DataGlove. _____	30
Figura 1.21: CyberGlove. _____	31
Figura 1.22: PowerGlove. _____	31
Figura 1.23: Dexterous Hand Master. _____	32
Figura 1.24: AcceleGlove. _____	33
Figura 1.25: Movimientos básicos asociados a la aceleración. _____	34
Figura 1.26: Aplicaciones sugeridas por freescale para los diferentes niveles “g” de los acelerómetros. _____	35
Figura 1.27: Esquema Acelerómetro Capacitivo. _____	36
Figura 1.28: Modelo físico simplificado de transductor. _____	37
Figura 1.29: Ejes de medición de un giroscopio. _____	38
Figura 1.30: Sensores Flexo Resistivos. _____	39
Figura 1.31: Sensor de fuerza resistivo rectangular. _____	40
Figura 1.32: Construcción del FSR. _____	41
Figura 1.33: El microcontrolador en un sistema cerrado. _____	42
Figura 1.34: Modelo de un sistema de comunicaciones genérico. _____	45
Figura 1.35: Velocidad de las tecnologías inalámbricas según su anchura de banda. _____	48

Figura 1.36: Alcance de las tecnologías inalámbricas	48
Figura 1.37: Consumo de potencia de las tecnologías inalámbricas,	49
Figura 1.38: Costo de la infraestructura de las tecnologías inalámbricas.	49
Figura 1.39: Paleta Robotic Arm incluida en LabVIEW Robotics.	54
Figura 1.40: NI-VISA Interfaz común entre drivers e instrumentos reales.	55
Figura 2.1: Diagrama general del proyecto.	62
Figura 2.2: Acelerómetro MMA7361L montado en PCB marca Sparkfun.	65
Figura 2.3: Esquema de la PCB Sparkfun.	66
Figura 2.4: Distribución de pines del acelerómetro MMA7361L montado en la PCB.	66
Figura 2.5: Posición del sensor.	68
Figura 2.6: Comportamiento Dinámico.	68
Figura 2.7: Comportamiento estático.	69
Figura 2.8: Flex Sensor de Spectra Symbol	70
Figura 2.9: Diagrama de dimensiones del Flex Sensor.	71
Figura 2.10: Forma de funcionamiento del Flex Sensor.	71
Figura 2.11: Microcontrolador PIC18F452.	73
Figura 2.12: Módulo XBee Serie 1.	74
Figura 2.13: Grados de libertad de las articulaciones del brazo humano.	77
Figura 2.14: Movimiento de la mano en pronación y supinación.	77
Figura 2.15: Movimiento de la mano en Flexión y Extensión	78
Figura 2.16: Movimientos del antebrazo: Flexión y extensión de la articulación del codo.	78
Figura 2.17: Movimientos del brazo: Flexión y extensión de la articulación del hombro.	79
Figura 2.18: Acelerómetros montados sobre el dispositivo guante.	79
Figura 2.19: Ubicación del sensor flex sobre el guante.	80
Figura 2.20: Nunchuk utilizado en la estación local.	80
Figura 2.21: Placa electrónica de la estación local.	81
Figura 2.22: Fuente de alimentación de la placa PCI 1.	82
Figura 2.23: Circuito regulador de tensión para la PCI 1.	82
Figura 2.24: Circuito de desacople de impedancias para los acelerómetros.	83
Figura 2.25: Conexiones realizadas en el microcontrolador de la placa PCI 1.	85
Figura 2.26: Circuito de conexión del módulo XBee con el microcontrolador de la placa PCI 1.	86
Figura 2.27: Partes del brazo robótico Lynxmotion.	87
Figura 2.28: Brazo robótico Lynxmotion AL5D con sus articulaciones.	88
Figura 2.29: Controladora serial de 16 servos marca Pololu.	91
Figura 2.30: Secuencia de bytes utilizada con la controladora de servos Pololu.	96

Figura 2.31: Sensor FSR 402 de Interlink Electronics. _____	97
Figura 2.32: Dimensiones del FSR 402. _____	98
Figura 2.33: Circuito regulador de tensión para la PCI 2. _____	100
Figura 2.34: Placa electrónica estación remota. _____	100
Figura 2.35: Circuito de acondicionamiento para el FSR. _____	101
Figura 2.36: Conexiones realizadas en el microcontrolador de la placa PCI 2. _____	103
Figura 2.37: Circuito de conexión del módulo XBee con el microcontrolador de la placa PCI 2. _____	104
Figura 2.38: Diagrama del circuito inversor de nivel. _____	105
Figura 2.39: Conexión de la PCI 2 y la controladora de servos. _____	106
Figura 2.40: Diagrama null modem 3 hilos sin control de flujo. _____	106
Figura 2.41: Estructura del algoritmo del microcontrolador de la tarjeta PCI 1. _____	107
Figura 2.42: Diagrama de flujo para la lectura de los sensores y filtro promediador. _____	108
Figura 2.43: Diagrama de flujo del funcionamiento del filtro pasa banda. _____	109
Figura 2.44: Diagrama de flujo para la creación de las tramas de datos. _____	110
Figura 2.45: Estructura del vector tipo texto usado para enviar la primera trama de datos. _____	111
Figura 2.46: Estructura del vector tipo texto usado para enviar la segunda trama de datos. _____	111
Figura 2.47: Transmisión inalámbrica de las tramas de datos. _____	112
Figura 2.48: Diagrama de flujo del algoritmo de control del microcontrolador de la estación remota. _____	113
Figura 2.49: Posición HOME del brazo robótico. _____	114
Figura 2.50: Posibles valores de los bytes de posición para la controladora de servos. _____	115
Figura 2.51: Diagrama de flujo de lectura de tramas. _____	117
Figura 2.52: Trama de datos enviada desde el HMI a la placa PCI 2. _____	118
Figura 2.53: Analogía para la interpolación. _____	120
Figura 2.54: Diagrama de flujo para el control de la base del robot. _____	121
Figura 2.55: Diagrama de flujo del control del gripper del robot. _____	122
Figura 2.56: Diagrama de flujo para la selección de la velocidad de los servomotores. _____	124
Figura 2.57: Placa XBee Explorer _____	125
Figura 2.58: Interfaz de configuración de módulos XBee en el X-CTU. _____	127
Figura 2.59: Panel Frontal del VI principal. _____	132
Figura 2.60: Event Structure _____	133
Figura 2.61: Panel frontal Supervisión 4gdl.vi _____	133
Figura 2.62: Diagrama de bloques algoritmo de visualización de cámaras. _____	134
Figura 2.63: Recursos generales del algoritmo. _____	135

Figura 2.64: Controles virtuales del parámetro velocidad. _____	135
Figura 2.65: Método de operación Ejes Específicos (°). _____	136
Figura 2.66: Límites de operación de cada articulación del manipulador. _____	137
Figura 2.67: Sistema de coordenadas del manipulador. _____	138
Figura 2.68: Método de operación Coordenadas XYZ (mm). _____	139
Figura 2.69: Diagrama con los eslabones y ejes del robot. _____	140
Figura 2.70: Sistemas de referencia y ejes del robot. _____	141
Figura 2.71: Grados 4gdl.vi _____	143
Figura 2.72: Algoritmo envío serial de datos. _____	144
Figura 2.73: Dimensiones estructura mecánica. _____	145
Figura 2.74: Vista superior de la estructura mecánica. _____	146
Figura 2.75: Montaje del hardware guante de datos. _____	146
Figura 2.76: Hardware guante de datos. _____	147
Figura 2.77: Dimensiones máximas del área de trabajo del manipulador. _____	148
Figura 2.78: Fijación de herramientas. _____	149
Figura 3.1: Acelerómetro sobre las articulaciones del codo del robot manipulador. ____	150
Figura 3.2: Diagrama de flujo del control de posicionamiento con realimentación. ____	151
Figura 3.3: Imitación de la forma del robot con el brazo humano. _____	153
Figura 3.4: Posición brazo humano /HOME robot manipulador. _____	153
Figura 3.5: Medición del efector final en el eje Z. _____	156
Figura 3.6: Peso máximo de carga del robot. _____	158
Figura 3.7: Teleoperación con manipulación de objetos aleatorios. _____	160
Figura 3.8: Diagrama de bloques de la aplicación en Math Script. _____	165
Figura 3.9: Comparación Forward Kinematics vs Math Script. _____	167
Figura 3.10: Medición del Ángulo Real del eje de cada actuador. _____	170
Figura 3.11: Curvas Grados-Valores de controladora del hombro. _____	171
Figura 3.12: Prueba de posicionamiento, posición N° 2 (263, 106, 60). _____	173
Figura 3.13: Visualización de video de las cámaras Web haciendo uso de un HUB USB. _____	175
Figura 3.14: HMI sin video al usar el HUB y extensiones USB. _____	176
Figura 3.15: Posición y visualización de las cámaras Web. _____	177
Figura 3.16: Video de cámaras con falta de información sobre orientación _____	178
Figura 3.17: Falta de información sobre profundidad al sujetar el objeto. _____	179
Figura 3.18: Falta de información sobre profundidad al colocar el objeto. _____	179
Figura 3.19: Teleoperación con el HMI haciendo uso de las cámaras web. _____	180
Figura 3.20: Teleoperación con tres cámaras web por gradilla. _____	181



## ÍNDICE DE TABLAS

<i>Tabla 1.1: Comparación entre las redes radio y las redes cableadas convencionales.</i>	47
<i>Tabla 1.2: Características básicas de Zigbee</i>	51
<i>Tabla 2.1: Descripción de pines del acelerómetro MMA7361L montado en la PCB.</i>	67
<i>Tabla 2.2: Especificaciones técnicas Flex Sensor</i>	72
<i>Tabla 2.3: Características principales PIC18F452.</i>	73
<i>Tabla 2.4: Descripción de I/O utilizadas en el microcontrolador de la placa PCI 1.</i>	84
<i>Tabla 2.5: Descripción de servomotores correspondientes a las articulaciones.</i>	89
<i>Tabla 2.6: Distancia entre ejes del Brazo robótico AL5D</i>	89
<i>Tabla 2.7: Características técnicas de servomotores Hitec.</i>	90
<i>Tabla 2.8: Especificaciones de la SSC16 Pololu</i>	92
<i>Tabla 2.9: Especificaciones técnicas del sensor FSR 402.</i>	99
<i>Tabla 2.10: Descripción de I/O utilizadas en el microcontrolador de la placa PCI 2.</i>	102
<i>Tabla 2.11: Descripción de canales libres de la placa PCI 2.</i>	103
<i>Tabla 2.12: Valores de los bytes se envían a la controladora para la posición HOME.</i>	116
<i>Tabla 2.13: Valores digitales medidos de los acelerómetros usados en la interpolación.</i>	119
<i>Tabla 2.14: Ejemplo de valores de interpolación.</i>	120
<i>Tabla 2.15: Parámetros de configuración para las direcciones de los módulos XBee.</i>	128
<i>Tabla 2.16: Parámetros D-H del robot.</i>	141
<i>Tabla 3.1: Posiciones del brazo humano para la teleoperación del robot manipulador.</i>	154
<i>Tabla 3.2: Verificación del posicionamiento del robot en el eje Z.</i>	156
<i>Tabla 3.3: Resultado de las pruebas de teleoperación con objetos aleatorios.</i>	159
<i>Tabla 3.4: Pruebas de manipulación de material peligroso.</i>	161
<i>Tabla 3.5: Prueba de validación del modelo cinemático directo.</i>	166
<i>Tabla 3.6: Rangos de movimientos de las articulaciones del manipulador</i>	167
<i>Tabla 3.7: Prueba de validación del modelo cinemático inverso.</i>	169
<i>Tabla 3.8: Valores medidos de la articulación del hombro.</i>	171
<i>Tabla 3.9: Prueba de posicionamiento del manipulador robótico.</i>	173
<i>Tabla 3.10: Costo de componentes para la implementación del proyecto.</i>	182
<i>Tabla 3.11: Costo de diseño e implementación del proyecto.</i>	183

## RESUMEN

La robótica ya no es un campo desconocido en el Ecuador y cada vez se desarrollan mejores e innovadores sistemas que aportan al desarrollo de la industria, pero para ello se necesita previa investigación que debe surgir principalmente de las Universidades. Con este concepto se busca aportar conocimientos sobre teleoperación y robótica al alumnado de la ESPE-L a través de la implementación de prototipos de teleoperación.

Este proyecto presenta el diseño e implementación de un dispositivo de control concebido como guante sensorizado que servirá como interfaz de teleoperación de un brazo robótico didáctico de 5GDL para realizar tareas de manipulación de objetos y transporte de materiales peligrosos.

La implementación del sistema parte con la investigación de acelerómetros y sensores flexoresistivos usados para sensar el movimiento del brazo humano y apertura de la mano. Los sensores son colocados en el guante y un microcontrolador se encarga de leer sus señales para enviarlas a través de un módulo Xbee. Se implementa también un pulsador de hombre-muerto y un dispositivo manual para controlar el giro de la base del robot. Un segundo módulo Xbee y microcontrolador reciben y utilizan la información para generar las señales de control para el robot manipulador a través de una SSC-16. Además, se crea un HMI en LabVIEW que resuelve el modelo cinemático inverso y directo del manipulador, y permite el control del mismo a través de un PC. Este HMI adiciona el video de cámaras web usadas para supervisión.

Es así como se obtiene un módulo de teleoperación para un manipulador robótico que combina técnicas de electrónica, programación y robótica con el uso de nuevas tecnologías en sensores para lograr una visión más amplia de la capacidad y alcances de la Ingeniería Mecatrónica, contribuyendo con beneficios en el área de teleoperación que pueden ser escalados a niveles industriales.

## **ABSTRACT**

Currently our country's industry is in full swing with the inclusion of more and better technologies in different areas. Robotics is no longer an unknown field in Ecuador and increasingly develops better and innovative systems, but this requires prior investigation should arise mainly from the universities. This concept seeks to provide knowledge about robotic arms teleoperation to related careers of the ESPE-L through the implementation of practical and modular teleoperation prototypes.

This project presents the design and implementation of a control device like a glove designed to serve as a sensorised teleoperation interface of a didactic robotic arm with five degrees of freedom to perform object manipulation and transport of hazardous materials.

The system implementation starts with the investigation of accelerometers and flex-sensors used to sense the movement of the human arm and hand opening. The sensors are placed in the glove and a microcontroller is responsible for reading their signals to be sent through an Xbee module. We also implement a dead-man button and a handheld device for controlling the robot base rotation. A second Xbee module receives the data and a second microcontroller, through a complex algorithm, uses this information to generate control signals for the robotic arm through a servo controller SSC-16. In addition, an HMI is created in LabVIEW it solves the direct and inverse kinematic model of the manipulator, and allows control it through a PC. This HMI adds video webcam used for monitoring the robot.

This is how we get a teleoperation module for a robotic arm which combines electronics, programming and robotics with the use of new technologies in sensors to achieve a broader view of the capacity and scope of Mechatronics Engineering, contributing with benefits in teleoperation area that can be scaled up to industrial levels.

## **PRESENTACIÓN**

En el presente proyecto se abordan las fases de diseño e implementación de un guante sensorizado, aplicado al control a distancia de un manipulador robótico en escala de prototipo para manipulación y transporte de materiales peligrosos.

Mediante un sistema de sensores que midan la posición y movimientos del brazo humano; se puede controlar remotamente un prototipo de brazo robótico que permita el desarrollo de tareas en un entorno que conlleve un riesgo al operador.

En el Capítulo I, se detalla el marco teórico, donde se mencionan los conceptos básicos de teleoperación, interfaces de teleoperación, robótica, sensores inerciales, microcontroladores y transmisión inalámbrica de datos; información relevante para el desarrollo del proyecto.

En el Capítulo II, se presenta las características y funciones de los elementos determinados para la implementación del proyecto; exhibiendo principalmente el diseño e integración en general de todo el hardware y software requerido en el mismo; además del desarrollo de la estructura del sistema de control, de la interfaz hombre máquina HMI y la comunicación entre estaciones.

En el Capítulo III, se exponen los resultados obtenidos de la aplicación de las pruebas experimentales realizadas al sistema de teleoperación, con lo cual se determinó el proceder del manipulador robótico en función al control a distancia mediante el guante sensorizado y con la aplicación del HMI.

En el Capítulo IV, se registran las conclusiones y recomendaciones que se obtuvieron en el transcurso del diseño e implementación del proyecto.

# CAPÍTULO I

## FUNDAMENTOS TEÓRICOS

### 1.1 EL BRAZO HUMANO

Nuestros brazos (Mendoza.edu, sf), sin duda, son mecanismos enormemente adaptables. Son capaces de maniobrar en cualquier posición que se desee, para ello, tienen dos articulaciones principales: **el hombro y el codo** (la muñeca, hasta donde la robótica trata, se considera parte del mecanismo de la mano).

#### **El Hombro**

El hombro se puede mover en dos planos, arriba y abajo, hacia detrás y hacia delante. Si se mueven los músculos del hombro hacia arriba el brazo entero se levanta separándose del cuerpo. Si se mueven los músculos del hombro hacia delante el brazo entero se mueve hacia delante.

#### **El Codo**

La articulación del codo es capaz de moverse en dos planos: atrás y adelante, arriba y abajo. Las articulaciones del brazo y su capacidad de moverse se llaman grados de libertad. El hombro ofrece dos grados de libertad por sí mismo: rotación del hombro y flexión del hombro. La articulación del codo añade un tercero y cuarto grados de libertad: la flexión del codo y la rotación del codo.

Los brazos robóticos también tienen grados de libertad. No obstante en lugar de músculos, tendones, rótulas y huesos, los brazos robóticos están hechos de metal, plástico, madera, motores, electroimanes, engranajes, poleas y otros componentes mecánicos.

Algunos brazos robóticos solo proporcionan un grado de libertad; otros proporcionan tres, cuatro, incluso cinco grados distintos de libertad.

## **1.2 MANIPULADORES ROBÓTICOS**

### **1.2.1 ROBÓTICA**

La robótica (Roboticspot, 2011) es la ciencia encaminada a diseñar y construir aparatos y sistemas capaces de realizar tareas propias de un ser humano

El término "Robótica" fue acuñado por Isaac Asimov para describir la tecnología de los robots. Él mismo predijo hace años el aumento de una poderosa industria robótica, predicción que ya se ha hecho realidad.

Recientemente se ha producido una explosión en el desarrollo y uso industrial de los robots tal que se ha llegado al punto de hablar de "revolución de los robots" y "era de los robots".

El término robótica puede ser definido desde diversos puntos de vista:

#### ***Con independencia respecto a la definición de "robot":***

"La Robótica es la conexión inteligente de la percepción a la acción"... (Brady & Paul , 1984).

#### ***En base a su objetivo:***

"La Robótica consiste en el diseño de sistemas. Actuadores de locomoción, manipuladores, sistemas de control, sensores, fuentes de energía, software de calidad--todos estos subsistemas tienen que ser diseñados para trabajar conjuntamente en la consecución de la tarea del robot"... (Jones & Flynn, 1993).

### ***Supeditada a la propia definición del término robot:***

"La Robótica describe todas las tecnologías asociadas con los robots"

La palabra robot (González, Centro de Formación del Profesorado e Innovación Educativa de Valladolid, 2002) fue usada por primera vez en el año 1921, cuando el escritor checo Karel Capek (1890 - 1938) estrena en el teatro nacional de Praga su obra Rossum's Universal Robot (R.U.R.).

El origen de la palabra "robot" proviene de la palabra eslava robota, que se refiere al trabajo realizado de manera forzada.

En las historias de robots de Isaac Asimov, éste prevé un mundo futuro en el que existen reglas de seguridad para que los robots no puedan ser dañinos para los seres humanos.

Por tal razón Asimov propuso las siguientes tres leyes de la robótica:

1ª.- Un robot no puede dañar a un ser humano o, a través de la inacción, permitir que se dañe a un ser humano.

2ª.- Un robot debe obedecer las órdenes dadas por los seres humanos, excepto cuando tales órdenes estén en contra de la primera ley.

3ª.- Un robot debe proteger su propia existencia siempre y cuando esta protección no entre en conflicto con la primera y segunda ley.

## **1.2.2 ROBOT INDUSTRIAL MANIPULADOR**

### **a. DEFINICIÓN**

Los primeros intentos (Barrientos, Peñin, Balaguer, & Aracil, 2007, págs. 17-18) de establecer una definición formal de robot, surgen en el año

1979 por parte de la RIA (*Robot Institute of America*, actualmente *Robotic Industries Association*), según la cual:

*“Un robot industrial es un manipulador multifuncional reprogramable, capaz de mover materias, piezas, herramientas o dispositivos especiales, según trayectorias variables, programadas para realizar tareas diversas”.*

La Organización Internacional de Estándares (ISO) adopta la definición de robot industrial como:

*“Manipulador de 3 o más ejes, con control automático, reprogramable, multiaplicación, móvil o no, destinado a ser utilizado en aplicaciones de automatización industrial. Incluye al manipulador (sistema mecánico y accionadores) y al sistema de control (software y hardware de control y potencia)”.*

Una definición más completa (González, Centro de Formación del Profesorado e Innovación Educativa de Valladolid, 2002) es la establecida por la Asociación Francesa de Normalización (AFNOR), que define primero el manipulador y, basándose en dicha definición, el robot:

**Manipulador:** Mecanismo formado generalmente por elementos en serie, articulados entre sí, destinado al agarre y desplazamiento de objetos. Es multifuncional y puede ser gobernado directamente por un operador humano o mediante dispositivo lógico.

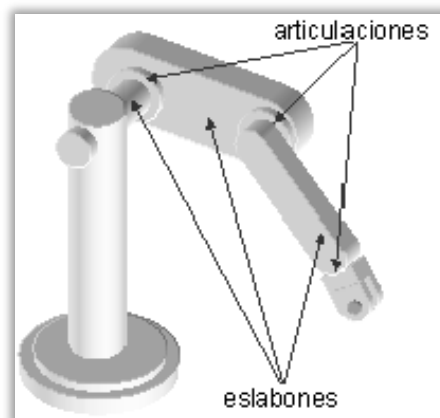
**Robot:** Manipulador automático servo-controlado, reprogramable, polivalente, capaz de posicionar y orientar piezas, útiles o dispositivos especiales, siguiendo trayectorias variables reprogramables, para la ejecución de tareas variadas. Normalmente tiene la forma de uno o varios brazos terminados en una muñeca. Su unidad de control incluye un dispositivo de memoria y ocasionalmente de percepción del entorno. Normalmente su uso es el de realizar una tarea de manera cíclica, pudiéndose adaptar a otra sin cambios permanentes en su material.



Común en todas las definiciones anteriores es la aceptación del robot industrial como un brazo mecánico con capacidad de manipulación y que incorpora un control más o menos complejo. Un sistema robotizado, en cambio, es un concepto más amplio. Engloba todos aquellos dispositivos que realizan tareas de forma automática en sustitución de un ser humano y que pueden incorporar o no a uno o varios robots, siendo esto último lo más frecuente.

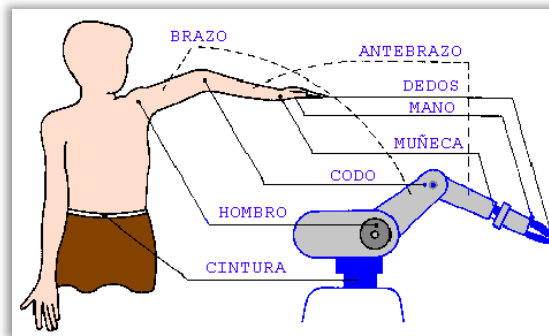
## b. ESTRUCTURA DE LOS ROBOTS INDUSTRIALES

Mecánicamente (Barrientos, Peñin, Balaguer, & Aracil, 2007, pág. 31), un robot está formado por una serie de elementos o eslabones unidos mediante articulaciones que permiten un movimiento relativo entre cada dos eslabones consecutivos (Figura 1.1).



**Figura 1.1: Elementos estructurales de un robot industrial.**

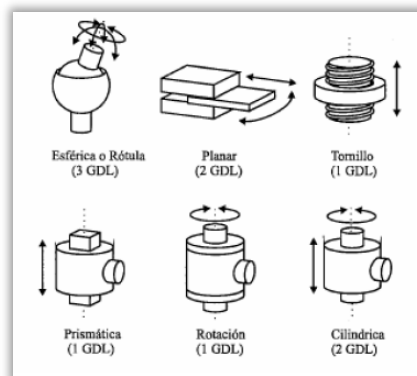
La constitución física de la mayor parte de los robots manipuladores industriales guarda cierta similitud con la anatomía del brazo humano, por lo que en ocasiones, para hacer referencia a los distintos elementos que componen el robot, se usan términos como cuerpo, brazo, codo y muñeca (Figura 1.2).



**Figura 1.2: Semejanza de un brazo manipulador con la anatomía humana.**

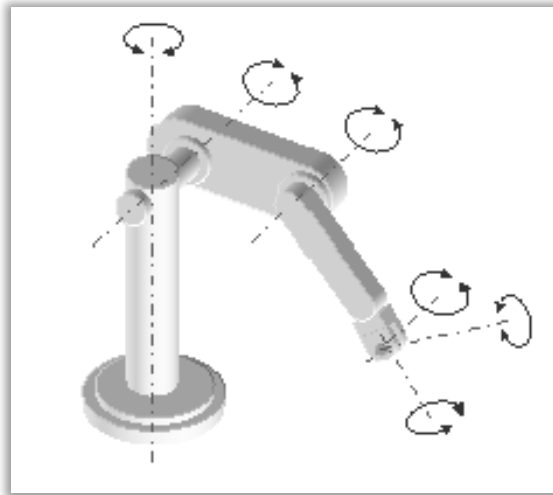
El movimiento de cada articulación puede ser de desplazamiento, de giro, o una combinación de ambos. En 1876 Franz Reuleaux identificó los posibles movimientos relativos entre dos elementos de contacto. Reuleaux denominó pares inferiores (lower pairs) a aquellos en los que el contacto se realiza entre superficies, mientras que si el contacto es puntual o lineal, los denominó pares superiores.

La Figura 1.3 muestra los seis posibles pares inferiores establecidos por Reuleaux, además del número de GDL de cada tipo de articulación de rotación correspondiente a los pares inferiores. Como se observa en ella, sólo tres tipos de articulación tienen un solo grado de libertad (estos se denominan pares lineales o de primer grado).



**Figura 1.3: Los seis pares inferiores de Reuleaux.**

Cada uno de los movimientos independientes que puede realizar cada articulación con respecto a la anterior, se denomina grado de libertad (GDL) (Figura 1.4).



**Figura 1.4: Distintos grados de libertad (GDL) de un brazo de robot.**

Una cadena cinemática, es una serie de eslabones o barras unidas por articulaciones. La estructura mecánica de un robot manipulador constituye una cadena cinemática. Al final de la cadena cinemática se puede conectar un elemento terminal o actuador final: una herramienta especial que permite al robot de uso general realizar una aplicación particular, que debe diseñarse específicamente para dicha aplicación: una herramienta de sujeción, de soldadura, etc.

### **c. TIPOS DE CONFIGURACIONES MORFOLÓGICAS**

La estructura del manipulador (González, Centro de Formación del Profesorado e Innovación Educativa de Valladolid, 2002) y la relación entre sus elementos proporcionan una configuración mecánica, que da origen al establecimiento de los parámetros que hay que conocer para definir la posición y orientación del elemento terminal.

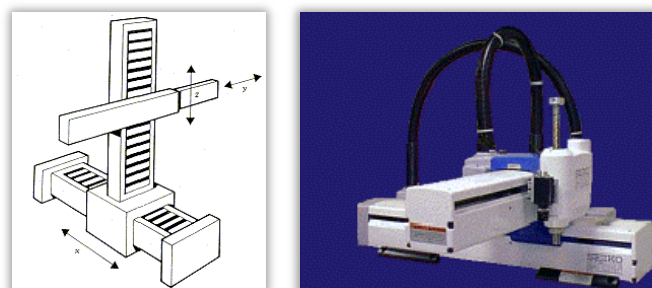
Fundamentalmente, existen cuatro estructuras clásicas en los manipuladores, que se relacionan con los correspondientes modelos de coordenadas en el espacio y que se citan a continuación: cartesianas, cilíndricas, esféricas, angulares. Y una no clásica SCARA (Selective Compliance Assembly Robot Arm).

### **Cartesiana / Rectilínea**

El posicionando se hace en el espacio de trabajo con las articulaciones prismáticas. Esta configuración se usa bien cuando un espacio de trabajo es grande y debe cubrirse, o cuando la exactitud consiste en la espera del robot. Posee tres movimientos lineales, es decir, tiene tres grados de libertad, los cuales corresponden a los movimientos localizados en los ejes X, Y y Z.

Los movimientos que realiza este robot entre un punto y otro son con base en interpolaciones lineales. Interpolación, en este caso, significa el tipo de trayectoria que realiza el manipulador cuando se desplaza entre un punto y otro.

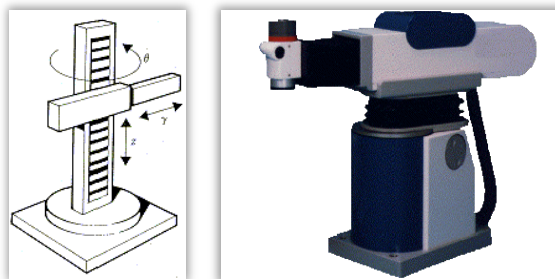
A la trayectoria realizada en línea recta se le conoce como interpolación lineal y a la trayectoria hecha de acuerdo con el tipo de movimientos que tienen sus articulaciones se le llama interpolación por articulación (Figura 1.5).



**Figura 1.5: Configuración Cartesiana.**

## Cilíndrica

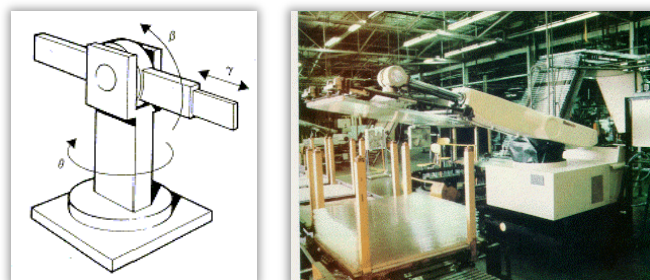
El robot tiene un movimiento de rotación sobre una base, una articulación prismática para la altura, y una prismática para el radio. Este robot ajusta bien a los espacios de trabajo redondos. Puede realizar dos movimientos lineales y uno rotacional, o sea, que presenta tres grados de libertad. Está diseñado para ejecutar los movimientos conocidos como interpolación lineal e interpolación por articulación. (Figura 1.6).



**Figura 1.6: Configuración Cilíndrica.**

## Esférica / Polar

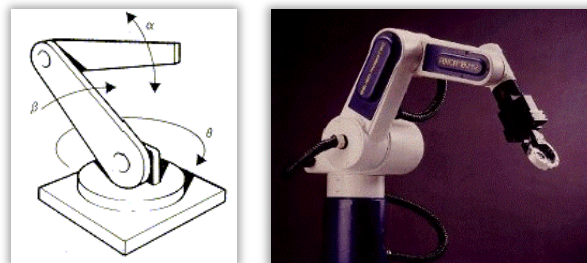
Dos juntas de rotación y una prismática permiten al robot apuntar en muchas direcciones, y extender la mano a un poco de distancia radial. Los movimientos son: rotacional, angular y lineal. El robot utiliza la interpolación por articulación para moverse en sus dos primeras articulaciones y lineal para la extensión y retracción (Figura 1.7).



**Figura 1.7: Configuración Esférica.**

### De Brazo articulado / Antropomórfica / Angular

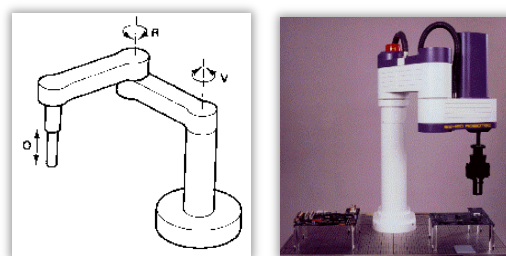
El robot usa 3 juntas de rotación para posicionarse. Generalmente, el volumen de trabajo es esférico. Estos tipos de robot se parecen al brazo humano, con una cintura, el hombro, el codo, la muñeca. Presenta una articulación con movimiento rotacional y dos angulares. Aunque el brazo articulado puede realizar el movimiento llamado interpolación lineal, el movimiento natural es el de interpolación por articulación, tanto rotacional como angular (Figura 1.8).



**Figura 1.8: Configuración Angular o Antropomórfica.**

### SCARA

Similar al de configuración cilíndrica, pero el radio y la rotación se obtiene por uno o dos eslabones. Este brazo puede realizar movimientos horizontales de mayor alcance debido a sus dos articulaciones rotacionales. El robot de configuración SCARA también puede hacer un movimiento lineal (mediante su tercera articulación) (Figura 1.9).



**Figura 1.9: Configuración SCARA.**

### 1.2.3 CINEMÁTICA DEL ROBOT

La cinemática del robot (Barrientos, Peñin, Balaguer, & Aracil, 2007, págs. 94-99) estudia el movimiento del mismo con respecto a un sistema de referencia.

Así, la cinemática se interesa por la descripción analítica del movimiento espacial del robot como una función del tiempo y en particular por las relaciones entre la posición y la orientación del extremo final del robot con los valores que toman sus coordenadas articulares.

Existen dos problemas fundamentales que se deben resolver en la cinemática del robot, los cuales se pueden observar en la figura 1.10

El primero de ellos es el **problema cinemático directo**, que consiste en determinar cuál es la posición y orientación del extremo final del robot, con respecto a un sistema de coordenadas que se toma como referencia, conocidos los valores de las articulaciones y los parámetros geométricos del robot; el **problema cinemático inverso**, resuelve la configuración que debe adoptar el robot para una posición y orientación del extremo conocidas.

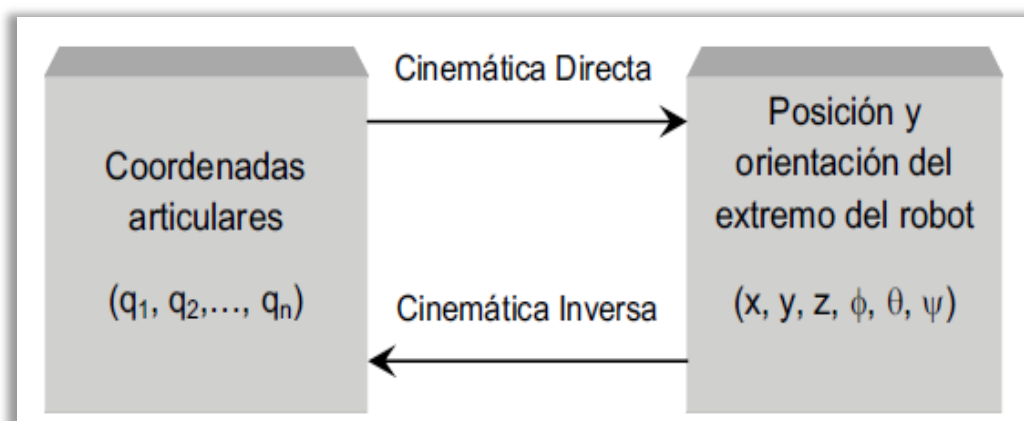


Figura 1.10: Diagrama de relación entre cinemática directa e inversa.

## a. RESOLUCIÓN DEL PROBLEMA CINEMÁTICO DIRECTO

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. No es en general complicado determinar dichas relaciones (robots de pocos GDL) mediante simples consideraciones geométricas.

Para robots de más grados de libertad puede plantearse un método sistemático basado en la utilización de las matrices de transformación homogénea, donde la matriz de transformación homogénea  $T$  relaciona la posición y orientación del extremo del robot respecto del sistema de referencia fijo situado en la base del mismo, esta matriz  $T$  será función de las coordenadas articulares.

Se denomina matriz  ${}^{i-1}A_i$  a la matriz de transformación homogénea que representa la posición y orientación relativa entre los sistemas asociados a dos eslabones consecutivos del robot; así pues  ${}^0A_2$  describe la posición y orientación del sistema de referencia solidario del segundo eslabón respecto al sistema de coordenadas de la base, esto es:

$${}^0A_2 = {}^0A_1 {}^1A_2 \quad \text{Ec. 1.1}$$

Considerando todos los grados de libertad, a la matriz  ${}^0A_i$  se la denomina  $T$ . Así, dado un robot de seis grados de libertad, se tiene que la posición y orientación del eslabón final vendrá dada por la matriz  $T$ :

$$T = {}^0A_1 {}^1A_2 {}^2A_3 {}^3A_4 {}^4A_5 {}^5A_6 \quad \text{Ec. 1.2}$$

En robótica para describir la relación que existe entre dos elementos contiguos se suele utilizar la representación Denavit-Hartenberg (D-H), la cual especifica que escogiendo adecuadamente los sistemas de coordenadas asociados a cada eslabón, será posible pasar de uno al



siguiente mediante cuatro transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón.

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 sólo si los sistemas  $\{S_{i-1}\}$  y  $\{S_i\}$  han sido definidos de acuerdo a unas normas determinadas que se expondrán posteriormente):

1. Rotación alrededor del eje  $z_{i-1}$  un ángulo  $\theta_i$ .
2. Traslación a lo largo de  $z_{i-1}$  una distancia  $d_i$ ; vector  $d_i$  (0, 0,  $d_i$ ).
3. Traslación a lo largo de  $x_i$  una distancia  $a_i$ ; vector  $a_i$  (0, 0,  $a_i$ ).
4. Rotación alrededor del eje  $x_i$  un ángulo  $\alpha_i$ .

Dado que el producto de matrices no es conmutativo, las transformaciones se han de realizar en el orden indicado. De este modo se tiene que:

$${}^{i-1}A_i = T(z, \theta_i) T(0,0,d_i) T(a_i,0,0) T(x,\alpha_i) \quad \text{Ec. 1.3}$$

Y realizando el producto entre matrices:

$${}^{i-1}A_i = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. 1.4}$$

Para reducción de la expresión,  $C\theta_i$  representa a  $\cos\theta_i$  y  $S\theta_i$  equivale a  $\sin\theta_i$ ; de manera similar con el ángulo  $\alpha_i$ .

Donde  $\theta_i$ ,  $a_i$ ,  $d_i$ ,  $\alpha_i$  son los parámetros D-H del eslabón  $i$ .

## **b. ALGORITMO DE DENAVIT-HARTENBERG PARA LA OBTENCIÓN DEL MODELO CINEMÁTICO DIRECTO**

**DH1.** Numerar los eslabones comenzando con 1 (primer eslabón móvil de la cadena) y acabando con n (último eslabón móvil). Se numerará como eslabón 0 a la base fija del robot.

**DH2.** Numerar cada articulación comenzando por 1 (correspondiente al primer grado de libertad) y acabando en n.

**DH3.** Localizar el eje de cada articulación. Si ésta 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.

**DH4.** Para i de 0 a n-1 situar el eje  $z_i$  sobre el eje de la articulación i+1.

**DH5.** Situar el origen del sistema de la base  $\{S_0\}$  en cualquier punto del eje  $z_0$ .

Los ejes  $x_0$  y  $y_0$  se situarán de modo que formen un sistema ortogonal a derechas con  $z_0$ .

**DH6.** Para 'i' de 1 a n-1, situar el origen del sistema  $\{S_i\}$  (solidario al eslabón i) en la intersección del eje  $z_i$  con la línea normal común a  $z_{i-1}$  y  $z_i$ .

Si ambos ejes se cortasen se situaría  $\{S_i\}$  en el punto de corte. Si fueren paralelos  $\{S_i\}$  se situaría en la articulación i+1.

**DH7.** Situar  $x_i$  en la línea normal común a  $z_{i-1}$  y  $z_i$ .

**DH8.** Situar  $y_i$  de modo que forme un sistema ortogonal a derechas con  $x_i$  y  $z_i$ .

**DH9.** Situar el sistema  $\{S_n\}$  en el extremo del robot de modo que  $z_n$  coincida con la dirección de  $z_{n-1}$  y  $x_n$  sea normal a  $z_{n-1}$  y  $z_n$ .

**DH10.** Obtener  $q_i$  como el ángulo que hay que girar en torno a  $z_{i-1}$  para que  $x_{i-1}$  y  $x_i$  queden paralelos.

**DH11.** Obtener  $d_i$  como la distancia, medida a lo largo de  $z_{i-1}$ , que habría que desplazar  $\{S_{i-1}\}$  para que  $x_i$  y  $x_{i-1}$  quedasen alineados.

**DH12.** Obtener  $a_i$  como la distancia medida a lo largo de  $x_i$  (que ahora coincidirá con  $x_{i-1}$ ) que habría que desplazar el nuevo  $\{S_{i-1}\}$  para que su origen coincida con  $\{S_i\}$ .

**DH13.** Obtener  $\alpha_i$  como el ángulo que habría que girar entorno a  $x_i$ , para que el nuevo  $\{S_{i-1}\}$  coincida totalmente con  $\{S_i\}$ .

**DH14.** Obtener las matrices de transformación  ${}^{i-1}A_i$  definidas en la ecuación matricial (Ec. 1.3).

**DH15.** Obtener la matriz de transformación que relaciona el sistema de la base con el del extremo del robot  $T = {}^0A_1 \cdot {}^1A_2 \cdot \dots \cdot {}^{n-1}A_n$ .

**DH16.** La matriz  $T$  define la orientación (submatriz de rotación) y posición (submatriz de traslación) del extremo referido a la base, en función de las 'n' coordenadas articulares.

Los cuatro parámetros de D-H dependen únicamente de las características geométricas de cada eslabón y de las articulaciones que lo unen con el anterior y siguiente. En concreto estos representan:

**$q_i$**  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 de 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  $x_i$  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  $z_i$ .

$\alpha_i$  Es el ángulo de separación del eje  $z_{i-1}$  y el eje  $z_i$ , medido en un plano perpendicular al eje  $x_i$ , 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 Ec. 1.4.

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.

Una vez encontrada la matriz de transformación homogénea T, suele escribirse de la siguiente manera Ec. 1.5 para la interpretación de los términos obtenidos.

$$T = \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n & o & a & P \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. 1.5}$$

Donde n, o y a es una terna ortonormal que representa la orientación y P es un vector que representa la posición.

### c. RESOLUCIÓN DEL PROBLEMA CINÉMÁTICO INVERSO

El objetivo del problema cinemático inverso consiste en encontrar los valores que deben adoptar las coordenadas articulares del robot  $q = [q_1, q_2, \dots, q_n]^T$  para que su extremo se posicione y oriente según una determinada localización espacial  $(p, [n, o, a])$ .

Así cómo es posible abordar el problema cinemático directo de una manera sistemática a partir de matrices de transformación homogénea, e independientemente de la configuración del robot; no ocurre lo mismo con el problema cinemático inverso, siendo el procedimiento de obtención de las ecuaciones fuertemente dependiente de la configuración del robot.

La determinación del modelo cinemático inverso de un robot a partir de su modelo directo se refiere claramente a conocidas las relaciones que expresan el valor de la posición y orientación del extremo del robot en función de sus coordenadas articulares, obtener por manipulación directa las relaciones inversas; las mismas relaciones que responden a ecuaciones trigonométricas acopladas cuya resolución no es trivial, siendo en muchas ocasiones tan compleja la solución que obliga a desecharla.

Existen algunos métodos genéricos susceptibles de ser programados, de modo que un computador pueda, a partir de la cinemática directa del robot (parámetros D-H), obtener la n-upla de valores articulares que posiciona y orientan el extremo; con el inconveniente de que son métodos iterativos, cuya velocidad de convergencia e incluso su propia convergencia no siempre están garantizadas.

A la hora de resolver el problema cinemático inverso es adecuado encontrar una solución cerrada. Esto es, encontrar una relación matemática explícita de la forma:

$$q_k = f_k(x, y, z, \phi, \theta, \psi); k=1 \dots n \text{ (GDL)} \quad \text{Ec. 1.6}$$

La resolución del modelo cinemático inverso por métodos geométricos se basa en encontrar suficiente número de relaciones geométricas en las que intervendrán las coordenadas del extremo del robot, sus coordenadas articulares y las dimensiones físicas de sus elementos. Se suele recurrir a la relación de triángulos formados por elementos y articulaciones.

## **1.3 SISTEMAS DE TELEOPERACIÓN**

### **1.3.1 TELEOPERACIÓN**

Se entiende por teleoperación (Gómez de Gabriel, Ollero Baturone, & García Cerezo, 2006, págs. 1-2) la extensión de las capacidades sensoriales y destrezas humanas a una localización remota.

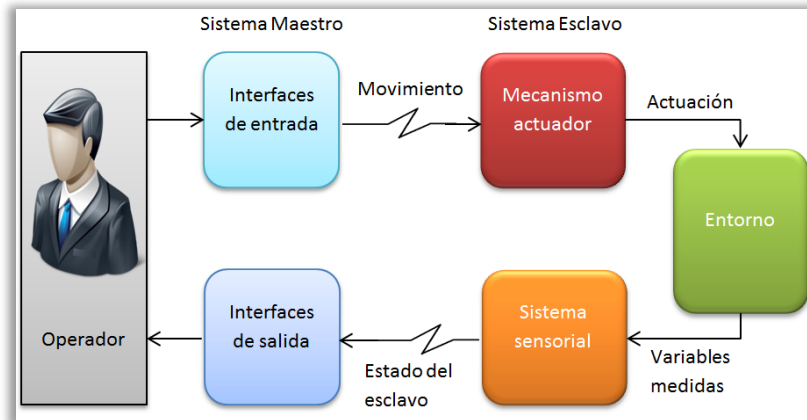
También, se habla de telemanipulación cuando el operador controla un dispositivo específico de manipulación, tal como un brazo articulado con una pinza, en una localización remota, o de teleguiado, cuando el operador realiza una tarea como la de guiado de un vehículo sin conductor.

Supuesto que en los dispositivos mencionados existen actuadores, tales como motores que generan sus movimientos, se utiliza también el término de teleactuación para referirse a los aspectos específicos de generación de órdenes a los actuadores. Así mismo, se habla de telesensorización para referirse a la captación y visualización de información sensorial en la localización remota.

La intervención del operador puede producirse en muchas formas diferentes, desde la teleoperación directa de los actuadores de las articulaciones de un brazo manipulador, hasta la simple especificación de movimientos, o incluso de tareas, las cuales se realizan de forma automática empleando para ello los sistemas informáticos de control en la estación de teleoperación y en la localización remota.

En la Figura 1.11 se muestra el esquema general de un sistema de teleoperación. El operador actúa en el sistema maestro mediante una interfaz de entrada y su movimiento se transmite al mecanismo actuador en el sistema esclavo que se encuentra en una ubicación remota,

mientras se realiza la tarea en el entorno, el sistema sensorial mide las variables que serán transmitidas hasta las interfaces de salida en el sistema maestro y así el operador pueda conocer el estado del esclavo.



**Figura 1.11: Esquema general de un sistema de operación.**

### 1.3.2 ELEMENTOS DE UN SISTEMA DE TELEOPERACIÓN

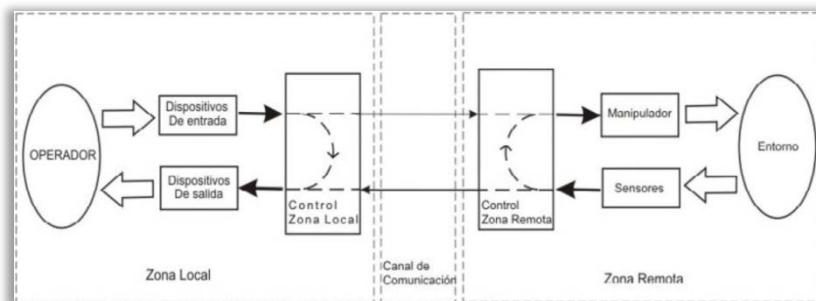
Un sistema de teleoperación (Alencastre, Muñoz, & Rudomon, 2003, págs. 314-321) consta de los siguientes elementos divididos en tres zonas (véase Figura 1.12):

1. Operador o teleoperador: es un ser humano que realiza a distancia el control de la operación. Su acción puede ir desde un control continuo hasta una intervención intermitente, con la que únicamente se ocupa de monitorizar y de indicar objetivos y planes cada cierto tiempo.
2. Dispositivo teleoperado: podrá ser un manipulador, un robot, un vehículo o dispositivo similar. Es la máquina que trabaja en la zona remota y que está siendo controlada por el operador.
3. Interfaz: conjunto de dispositivos que permiten la interacción del operador con el sistema de teleoperación. Se considera al manipulador maestro como parte del interfaz, así como a los monitores de vídeo, o

cualquier otro dispositivo que permita al operador mandar información al sistema (de entrada) y recibir información del mismo (de realimentación).

4. Control y canales comunicación: conjunto de dispositivos que modulan, transmiten y adaptan el conjunto de señales que se transmiten entre la zona remota y la local.

5. Sensores: conjunto de dispositivos que recogen la información, tanto de la zona local como de la zona remota, para ser utilizada por la interfaz y el control.



**Figura 1.12: Elementos de un sistema de teleoperación.**

### 1.3.3 TELERROBÓTICA

Los avances en la robótica (Gómez de Gabriel, Ollero Baturone, & García Cerezo, 2006) hacen posible el desarrollo de sistemas cada vez más capaces, dotados de sensores que les permiten actuar de manera autónoma para un gran número de aplicaciones. Sin embargo, esta capacidad de decisión de los sistemas robóticos no se encuentra desarrollada al nivel requerido para ejecutar determinadas tareas sin ningún tipo de supervisión humana. Por ello resulta necesario recurrir a la teleoperación, donde una persona interviene en las acciones del robot, desde un lugar alejado o separado.

Entonces se puede considerar a la telerrobótica como una forma evolucionada de la teleoperación que se caracteriza principalmente por el



aumento de la capacidad de decisión y actuación en el sistema remoto por parte del operador, el cual continúa manteniendo una intervención importante para la supervisión o teleoperación directa.

Una definición bastante concreta es la siguiente, “La telerrobótica se define como el conjunto de tecnologías que comprenden la monitorización y reprogramación a distancia de un robot por un ser humano. Se tendrá entonces la teleoperación de un robot, que se denominará telerrobot o robot teleoperado.” (Barrientos, Peñin, Balaguer, & Aracil, 2007).

#### **1.3.4 INTERFACES DE TELEOPERACIÓN**

Se entiende por interfaces de teleoperación a los dispositivos usados para controlar de manera directa al elemento teleoperado y los dispositivos que realimentan la información del entorno donde se ejecuta la tarea. Las interfaces descritas a continuación son una recopilación de los autores (Barrientos, Peñin, Balaguer, & Aracil, 2007, págs. 497-505) y (Gómez de Gabriel, Ollero Baturone, & García Cerezo, 2006, págs. 115-153).

##### **a. DISPOSITIVOS DE CONTROL**

###### **a.1. INTERFACES HÁPTICAS**

Son dispositivos bidireccionales que proporcionan sensación de fuerzas y/o tacto al operador, a través de la misma interfaz con la que envía consignas al sistema remoto. De este tipo los más comunes son los manipuladores con reflexión de fuerzas, son de dimensiones reducidas, dotados de actuadores, de diferentes grados de libertad y de configuración articular (véase Figura 1.13).



**Figura 1.13: Brazo Maestro Mini-master.**

### **a.2. DISPOSITIVOS DE SEGUIMIENTO CORPORAL O REALIDAD AUMENTADA**

Son dispositivos que permiten localizar en el espacio los diversos miembros (cabeza, brazos, piernas...), e incluso el cuerpo entero. Su principal aplicación es la interacción con entornos virtuales o en combinación con otros dispositivos, como los guantes sensorizados.

### **a.3. PALANCAS DE CONTROL**

Las palancas de control o joysticks (véase Figura 1.14) son las interfaces de entrada más utilizadas para control en velocidad o incremental, donde la posición de la palanca determina las velocidades de desplazamiento o de rotación del extremo de un manipulador o de un vehículo.



**Figura 1.14: Joystick con realimentación de fuerzas.**

#### **a.4. APUNTADORES TRIDIMENSIONALES**

Estos dispositivos son capaces de detectar los movimientos cartesianos en seis dimensiones (tres de traslación y tres de rotación) simultáneamente mediante la dirección de las fuerzas y pares que se aplican en el centro de una esfera aparentemente rígida que se maneja usando una sola mano (Véase Figura 1.15).



**Figura 1.15: Space Mouse.**

#### **a.5. GUANTES SENSORIZADOS**

También llamados guantes de datos (véase Figura 1.16). Se trata de guantes en los que se han colocado una serie de sensores que permiten captar el movimiento y la posición de cada uno de los dedos de la mano. Existen versiones en los que también es posible conocer la posición de la mano en el espacio.



**Figura 1.16: Guante sensorizado.**

## **b. DISPOSITIVOS DE REALIMENTACIÓN**

### **b.1. DISPOSITIVOS DE REALIMENTACIÓN VISUAL**

Se incluyen en este grupo todos aquellos sistemas que proyecten algún tipo de información de la zona remota para su visualización por parte del operador. Se incluyen aquí únicamente sistemas que proporcionen una imagen real de la zona remota (aunque sea en parte), y no únicamente una simulación de la misma. Pueden ser: monitores de video, pantallas de visualización tridimensional, cascos de visualización (véase Figura 1.17), visualizadores transparentes.



**Figura 1.17: Casco de realimentación visual.**

### **b.2. DISPOSITIVOS DE REALIMENTACIÓN HÁPTICA**

Son dispositivos que estimulan el sentido del tacto de diversas maneras para transmitir información al operador. La información se genera habitualmente a partir de información de contacto medida en la zona remota. Generalmente, estos dispositivos se corresponden con dispositivos de control articulados, de manera que haya una reciprocidad entre el movimiento y la realimentación. Se pueden distinguir dos grupos: Dispositivos de realimentación táctil u dispositivos de realimentación cinestésica.

Existen también otros dispositivos de realimentación, cabe destacar aquellos dispositivos relacionados con el sentido del oído, uno de los más importantes del ser humano y la forma principal de comunicación.

### **1.3.5 CONTROL EN LA TELEOPERACIÓN**

El control dentro de la teleoperación está ligado a una operación manual o directa por parte del operador, y no a una programación a distancia, como sería el caso del control supervisado o tele-programación. A continuación se describen los conceptos principales de control en la teleoperación en un orden en el cual se comprenda de mejor manera su clasificación e inclusión dentro de los mismos.

#### **a. CONTROL UNILATERAL**

En este tipo de control (Barrientos, Peñin, Balaguer, & Aracil, 2007, pág. 515), también llamado de bucle abierto (con respecto a la tarea o entorno), no existe realimentación de señal alguna desde el esclavo hacia el maestro, no estando este último motorizado en sus articulaciones. Es decir que, únicamente el maestro, mediante la interfaz de teleoperación, genera las señales de referencia o deseadas, ya sean de posición o velocidad, para los bucles de control de las articulaciones del esclavo.

#### **b. CONTROL EN POSICIÓN**

Éste (Gómez de Gabriel, Ollero Baturone, & García Cerezo, 2006, pág. 10) es el caso del control con un manipulador maestro o mediante órdenes que especifican posiciones o referencias para los sistemas de control realimentado de bajo nivel. Es el modo más simple de control y

puede ser unilateral o bilateral. En el caso de robots manipuladores estas instrucciones de posicionamiento contienen referencias articulares o cartesianas que se ejecutan de forma sencilla. Cuando se trata de robots móviles, las órdenes de posición dependen de la estructura del sistema de locomoción; éstas son típicamente de establecimiento de curvatura y velocidad de avance.

La realimentación al operador puede ser visual o cinestésica. El dispositivo de mando del operador puede ser pasivo o activo (dotado de actuadores), que traduce posiciones absolutas de la mano del operador en posiciones de referencia para el robot.

### **c. CONTROL EN VELOCIDAD**

De igual manera puede ser unilateral o bilateral. En este caso (Gómez de Gabriel, Ollero Baturone, & García Cerezo, 2006, págs. 10-11) las consignas están expresadas como velocidades cartesianas o articulares ejecutadas directamente por el controlador del robot. El tiempo de transmisión de las consignas de velocidad determina la trayectoria del robot o del vehículo, por lo que de este modo es más sensible a los retardos de comunicación, sobre todo si éstos son variables. Estas consignas incrementales poseen la ventaja de poder ser sumadas con las decisiones locales del robot para poder realizar un control compartido.

### **d. REFLEXION DE ESFUERZOS**

El proceso (Barrientos, Peñin, Balaguer, & Aracil, 2007, págs. 518-519) de transmitir al operador los esfuerzos de contacto que el manipulador esclavo realiza en el entorno remoto se conoce como reflexión de esfuerzos. Hay que tener en cuenta que se está hablando de reflexión y

no de realimentación, que sería un caso particular de reflexión. A continuación se describe brevemente cada una de las posibles formas de reflexión:

La *reflexión visual de fuerzas* se realiza presentando éstas al operador de manera gráfica a través de un visualizador. El gráfico de presentación debe incluir tanto las fuerzas ejercidas por el operador en cada uno de los ejes coordinados como los respectivos momentos.

En la *realimentación háptica* se busca aprovechar la excelente capacidad que posee el ser humano de percibir información transmitida mediante contacto o fuerzas de reacción. Este tipo de reflexión, por tanto, se realiza generando directamente una fuerza sobre la mano, excitando así dos elementos básicos de transmisión de señales hápticas hacia el sistema neurológico: los mecanorreceptores cutáneos de la mano (realimentación táctil) y los propioceptores de los músculos del brazo (realimentación de fuerzas o cinestésica) (véase Figura 1.18).

Existen también experimentos realizados *reflejando la fuerza mediante sonidos* que consisten en emitir sonidos al operador cuya intensidad está directamente relacionada con la magnitud de la fuerza que se está aplicando.



**Figura 1.18: Sistema de realimentación háptica, caso particular de cinestésica.**

### e. CONTROL BILATERAL

Control bilateral (Önal, Çağdaş, & Denizel, 2005, pág. 5) es el control bi-direccional de fuerza-posición entre dos sistemas conectados por un enlace de comunicación. Se utiliza típicamente en la teleoperación con realimentación de esfuerzos, de tal manera que el sistema principal es manejado por un operador. Los movimientos del operador son enviados hacia el sistema esclavo, generalmente remoto para el operador y las fuerzas encontradas son alimentadas de nuevo al sistema principal, lo que permite una telepresencia del operador en el entorno remoto. La necesidad de control bilateral radica en su aplicabilidad a las tareas que no pueden ser manejados por los manipuladores autónomos y / o alcanzado por los seres humanos.

De manera general la gran mayoría de autores consideran que éste control posee una realimentación cinestésica, tal como se puede observar en la definición de (Sabanovic & Ohnishi, 2011, pág. 233): El Control bilateral representa a los sistemas funcionalmente relacionados en donde el dispositivo esclavo replica el movimiento del operador en el sitio remoto y el dispositivo maestro ejerce una fuerza sobre el operador humano igual a la fuerza de interacción ejercida sobre el medio ambiente por el dispositivo esclavo (Figura 1.19)



**Figura 1.19: Esquema gráfico de teleoperación con control bilateral.**



## 1.4 GUANTES SENSORIZADOS

Los primeros guantes electrónicos (Ibarra, Ramírez, & Taboada, 2006) se crearon con la intención de reducir las limitantes que los elementos periféricos de las computadoras imponían en el uso de las manos; y la comercialización del VPL DataGlove y del Mattel Power Glove marcaron la pauta para la investigación y desarrollo de proyectos usando guantes electrónicos como interfaces para uso de la computadora o como dispositivos controlados por la computadora; sin embargo, el desarrollo de esta tecnología permitió darles aplicación en lenguaje de señas aunque sus principales usos se dirigen a la telemanipulación, realidad virtual, medicina, visualización científica, titiriteros, música y videojuegos.

Las técnicas iniciales de registro de movimientos de la mano empleaban elementos magnéticos muy precisos pero sensibles a objetos metálicos cercanos que provocan distorsión del campo y generaban errores.

Los guantes que manejaban sonido de alta frecuencia en triangulación (Logitech y Mattel Power Glove) también eran muy precisos, pero deficientes ante rebotes acústicos por las paredes y objetos cercanos.

Se desarrollaron guantes basados en la idea de dispositivos fotoeléctricos (elementos sensibles a la luz), cuyo principio de funcionamiento se asemeja al de la fibra óptica, y se basa en la percepción de un haz de luz emitido por una fuente de luz constante, donde los movimientos de flexión varían la intensidad de luz en el receptor, esto generó un método efectivo de control multidimensional.

Hoy en día también se utilizan bandas resistivas flexibles con lo cual se facilita el sensado de datos obteniéndose una buena resolución sin un hardware complicado. Adicionalmente también se diseñaron exoesqueletos los cuales tienen una estructura mecánica paralela y

sobrepuesta a la mano con rotores y sensores en cada articulación. Poseen una alta precisión por lo que se utilizan en aplicaciones delicadas.

Existen (Ortigueira, Reigosa, Rodríguez, Santamaría, & Veiga, sf) distintos tipos de guantes sensorizados, a continuación se mencionan algunos:

#### **1.4.1 DATAGLOVE**

El primer guante comercializado, y uno de los más usados es el DataGlove, creado por Zimmerman (VPL Research). Usa la fibra óptica para medir la flexión de los dedos: se emite un haz de luz en un extremo, y un sensor (diodo fotoeléctrico) detectará la intensidad en el otro extremo, que será diferente en función de la posición y flexión de los dedos.(Figura 1.20).



**Figura 1.20: DataGlove.**

#### **1.4.2 CYBERGLOVE**

El CyberGlove (Immersion Co.) emplea una tecnología diferente para conocer la flexión de los dedos, ya que en lugar de utilizar fibra óptica emplea pequeños sensores magnéticos (entre 18 y 22 por mano) que se localizan en la articulación de cada dedo y permiten medir la distancia entre ellos (Figura 1.21).



**Figura 1.21: CyberGlove.**

### **1.4.3 POWERGLOVE**

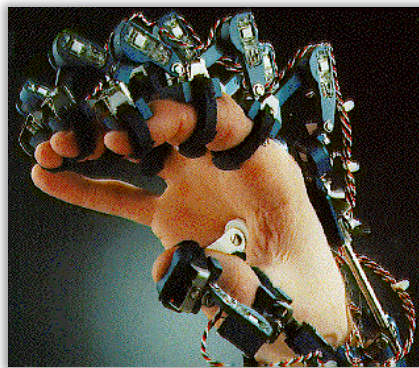
El PowerGlove no fue diseñado inicialmente para el mundo de la realidad virtual, ya que fue desarrollado por Nintendo, aunque por su reducido coste en comparación con el DataGlove y el CyberGlove, fue rápidamente incluido en sistemas de realidad virtual. Este tipo de guantes utiliza un material cuya resistencia eléctrica es variable (normalmente tinta de carbón) para medir la flexión de los dedos: al apretarlos, el carbón se contrae y disminuye su resistencia. También está dotado de trackers ultrasónicos (2 micrófonos en el guante y 3 emisores en el monitor) que mediante triangulación, como ya se ha explicado anteriormente, permiten calcular la orientación de la mano (aunque no pueden detectar el pitch, ya que solo hay 2 emisores). Aunque son mucho menos precisos que los guantes anteriores, gracias a su menor precio son muy usados (Figura 1.22).



**Figura 1.22: PowerGlove.**

#### 1.4.4 DEXTEROUS HAND MASTER

Para lograr mayor precisión se usan DHM (Dexterous Hand Master), que es como un exoesqueleto, ya que utiliza un sistema de seguimiento completamente mecánico: todos los sensores de los dedos están conectados a la máquina a través de cables, lo cual, por supuesto, lo hace mucho menos manejable y limita más el movimiento (Figura 1.23).



**Figura 1.23: Dexterous Hand Master.**

#### 1.4.5 ACCELEGLOVE

La empresa AnthroTronix (Cortes-Monroy, 2009) desarrolló inicialmente el guante junto al Departamento de Defensa de Estados Unidos para funciones relacionadas con el control robótico, pero también puede servir con videojuegos, entrenamiento deportivo (rehabilitación física), o bien traducir el lenguaje de señas (utilizado por sordo/mudos). Viene junto al SDK (kit de desarrollo de software) que le permite a los desarrolladores usar Java para programarlo en las aplicaciones que deseen.

Este guante está hecho de una mezcla de nylon transpirable ajustable con las puntas abiertas para poder escribir, cuenta con acelerómetros en los dedos y la palma que registran la orientación tridimensional. También con

un componente adicional se puede hacer el seguimiento del movimiento del brazo del usuario, este se conecta de bajo de la solapa del guante donde se encuentra el módulo que conecta los acelerómetros y el cable USB que manda la información a la computadora (Figura 1.24).



**Figura 1.24: AcceleGlove.**

## **1.5 SENSORES INERCIALES, FLEXORESISTIVOS Y DE PRESIÓN**

### **1.5.1 SENSORES INERCIALES**

Un sensor inercial es un dispositivo capaz de sensar los movimientos físicos que puede adoptar un objeto y que pueden relacionarse con la aceleración, y se utiliza en aplicaciones de captura y análisis de movimiento.

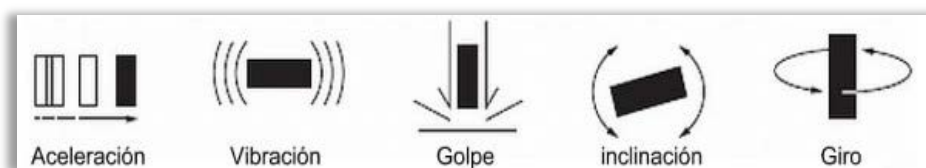
Los sensores inerciales (Groves, 2007, págs. 97-98) comprenden acelerómetros y giroscopios. Un acelerómetro mide la fuerza específica y un giroscopio mide la velocidad angular, ambos sin una referencia externa. Dispositivos que miden la velocidad, la aceleración, o la velocidad angular de un cuerpo con respecto a las características del entorno no son sensores inerciales.

La mayoría de los tipos de acelerómetros midan la fuerza específica a lo largo de un eje sensible. Del mismo modo, la mayoría de los tipos de giroscopios miden la velocidad angular alrededor de un solo eje. Una unidad de medición inercial (IMU) combina múltiples acelerómetros y giroscopios, generalmente tres de cada uno, para producir una medición de tres dimensiones de la fuerza específica y velocidad angular.

El desarrollo actual del sensor inercial se centra en la tecnología MEMS (micro-electro-mechanical systems). Los sensores MEMS son pequeños, ligeros, y muestran una mayor tolerancia a los golpes que los diseños convencionales mecánicos.

#### a. ACELERÓMETRO

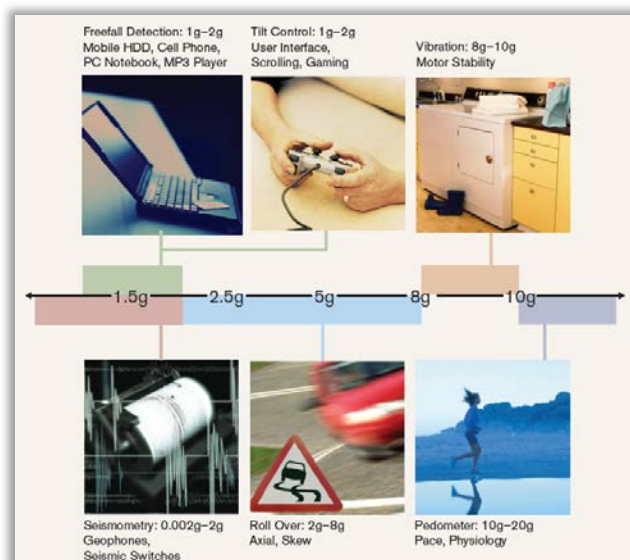
Aceleración (Sacco, 2011), vibración, golpe (shock), inclinación (tilt) y rotación (pan) son los cinco movimientos fundamentales (véase Figura 1.25) que un sistema inteligente debe detectar a todo momento para tener un control pleno sobre el objeto que desea gobernar o interpretar. Todas son, en realidad, manifestaciones diferentes de una aceleración durante períodos de tiempo distintos. Sin embargo, los seres humanos no relacionamos de manera intuitiva estos movimientos como variaciones en la aceleración/deceleración de un cuerpo. En cambio, si se tiene en cuenta y se analiza cada modalidad por separado se puede comprender de manera más sencilla muchas posibilidades que un acelerómetro puede ofrecer.



**Figura 1.25: Movimientos básicos asociados a la aceleración.**

Debido a que estos primeros cuatro modos de detección de movimiento (aceleración, vibración, choque e inclinación) se producen con la participación de determinados aspectos de la aceleración, se los mide con la unidad de la fuerza que ejerce la gravedad sobre un objeto en la Tierra, es decir, por la fuerza “g”. Un acelerómetro detecta la inclinación al medir el efecto que la fuerza de la gravedad ejerce sobre los ejes del acelerómetro que estén expuestos a esta acción en función de su posición en el espacio. Es decir, si se considera un acelerómetro de tres ejes inerciales (X, Y, y Z), se debe considerar las tres acciones de aceleración por separado para obtener los productos (resultados) de los ejes de movimiento.

Los acelerómetros MEMS (freescale, 2005) actualmente se fabrican en los rangos de detección de bajas y altas unidades de gravedad con anchos de banda mucho más amplios que antes, aumentando de este modo el campo de las aplicaciones potenciales, en la Figura 1.26 se puede apreciar las aplicaciones sugeridas por la empresa freescale para los niveles “g” de los acelerómetros.

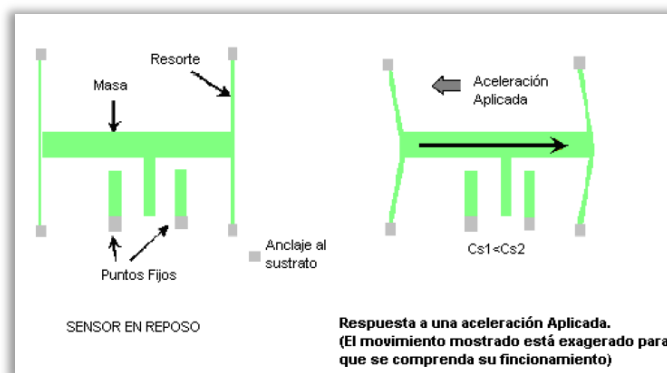


**Figura 1.26: Aplicaciones sugeridas por freescale para los diferentes niveles “g” de los acelerómetros.**

## ACELERÓMETROS CAPACITIVOS

Este tipo de dispositivo consiste de una superficie capacitiva micromecanizada de celdas de detección (g-cell) y un Circuito Integrado para Aplicaciones Específicas (o ASIC, por sus siglas en inglés) para el acondicionamiento de señal que se encuentran contenidas en un mismo encapsulado. El elemento sensor está sellado herméticamente en el nivel de oblea utilizando para ello un tapón de oblea en miniatura.

La g-cell es una estructura mecánica formada por materiales semiconductores (polisilicio) y empleando procesos semiconductores como enmascaramiento y grabado (masking y etching). Estas pueden ser modeladas como un conjunto de vigas unidas a una masa móvil central que se mueven entre haces fijos. Las vigas móviles pueden ser desviadas desde su posición de reposo por someter el sistema a una aceleración, ver Figura 1.27.

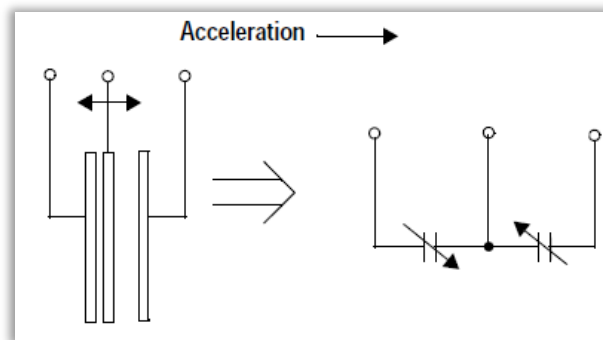


**Figura 1.27: Esquema Acelerómetro Capacitivo.**

Al moverse las vigas que se encuentran unidas a la masa central móvil, la distancia desde ellos hasta las vigas fijas situadas a un extremo se incrementará. Del mismo modo, la distancia al otro extremo decrecerá la misma magnitud. Y es precisamente este cambio en la distancia lo que nos otorga una medida de la aceleración.



Las vigas g-cell constituyen dos condensadores back-to-back, tal como se ilustra en la Figura 1.28. Cuando la viga central se desplaza con la aceleración, la distancia entre las vigas varía y, por lo tanto, la capacidad de cada condensador también cambia según la expresión  $C=A\epsilon/D$ . Donde A es el área de la viga,  $\epsilon$  es la constante dieléctrica y D la distancia entre vigas.



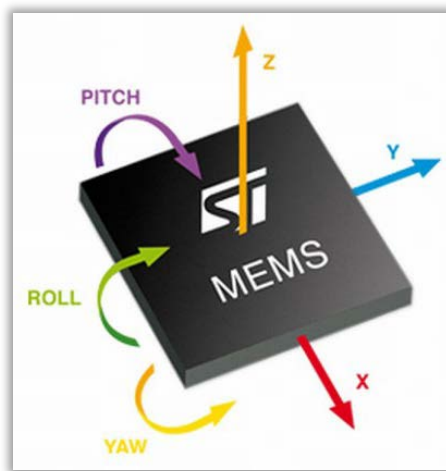
**Figura 1.28: Modelo físico simplificado de transductor.**

## b. GIROSCOPIO

Hasta aquí (Sacco, 2011) un acelerómetro se identifica hacia los movimientos de aceleración, vibración, choque e inclinación. La rotación en cambio requiere la comprensión de un movimiento angular. Este modo difiere de los anteriores porque la rotación puede tener lugar sin observarse cambios en la aceleración.

Para entender cómo funciona debemos observar la imagen de un sensor inercial de 3 ejes, es decir, debemos imaginar a los ejes del sensor X e Y en posición paralela a la superficie de la Tierra y el eje Z apuntando hacia el centro de la Tierra. En esta posición, el eje Z entrega una medición de 1 g (“g” como unidad de fuerza de gravedad), mientras que los ejes X e Y obtienen una medición de 0 g.

Girando el sensor de movimiento sólo sobre el eje Z, los ejes X e Y nunca abandonan la medida de 0 g ya que no sufren desplazamientos lineales en ninguna dirección, en tanto que el eje Z siempre continúa entregando la medida de 1 g ya que está siempre en el mismo lugar sin avanzar, ni retroceder. Por lo tanto, entendemos de este modo que para detectar los movimientos de rotación de un cuerpo (pitch, roll y yaw), se utilizan los giróscopos (véase Figura 1.29).



**Figura 1.29: Ejes de medición de un giroscopio.**

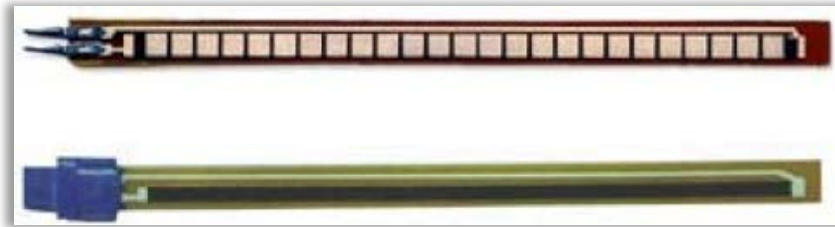
### **1.5.2 SENSORES FLEX**

Los sensores flex (Khwaja, 2013), son sensores flexo resistivos que cambian su resistencia dependiendo de la cantidad de doblez que presenten. Convierten el cambio de flexión en valor de resistencia, es decir que a más doblez la variación de la resistencia es mayor.

Estos sensores usualmente tienen forma de una tira delgada de 1 a 5 pulgadas de longitud que varían en resistencia (véase Figura 1.30).

Pueden ser usados de manera unidireccional donde solo pueden ser doblados en una dirección teniendo como resultado el aumento de la

resistencia, o de manera bidireccional donde dependiendo de la dirección de flexión la resistencia puede aumentar o disminuir.



**Figura 1.30: Sensores Flexo Resistivos.**

Comercialmente los sensores flex pueden ser adquiridos en valores de 1k $\Omega$  a 20k $\Omega$ , 20k $\Omega$  a 50k $\Omega$  y 50k $\Omega$  a 100k $\Omega$ .

#### **a. FUNCIONAMIENTO**

Los sensores flex son resistencias análogas que trabajan como un divisor variable de voltaje. El sensor está compuesto de elementos resistivos de carbono dentro de una lámina flexible de substrato. Más carbono significa menos resistencia.

Cuando el substrato es flexionado el sensor produce una salida de resistencia relativa al radio de curvatura.

#### **b. APLICACIONES**

**Robótica:** Los sensores flex pueden ser usados para determinar movimiento de articulaciones o desplazamiento.

Como **interruptores de contacto** para detección de paredes o **interruptores de presión** en gripper robóticos.

**Biomédica:** El sensor puede ser colocado en la articulación en movimiento de un equipo atlético para indicar el movimiento o desplazamiento.

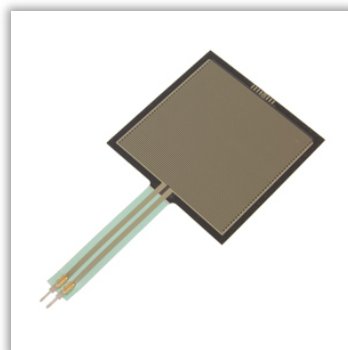
**Guantes sensorizados:** Estos sensores pueden ser incorporados dentro de un guante para la construcción de un guante de realidad virtual.

Los sensores flex también son usados en controles automáticos, productos deportivos, tecnología de asistencia, instrumentos musicales, joysticks y más.

### 1.5.3 SENSORES DE FUERZA FSR

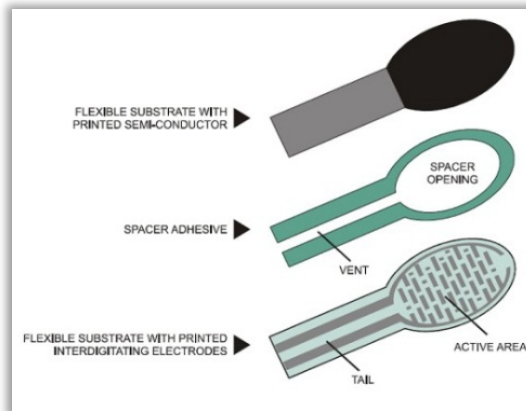
El (Carrión & Ochoa, 2009) sensor de fuerza FSR (Force Sensitive Resistor) es un dispositivo de película de polímero (PTF) que presenta una disminución de la resistencia cuando aumenta la fuerza aplicada a la superficie activa. Su sensibilidad a la fuerza está optimizada para uso en el control por toque humano de dispositivos electrónicos.

Los FSRs no son células de carga o galgas extensiométricas aunque tengan propiedades similares. En la Figura 1.31 se puede apreciar un FSR.



**Figura 1.31: Sensor de fuerza resistivo rectangular.**

El FSR está constituido por dos sustratos flexibles, el primero con electrodos impresos y el segundo con material semiconductor, unidos estos por una membrana adhesiva con un espacio abierto para la separación de los sustratos (véase Figura 1.32).



**Figura 1.32: Construcción del FSR.**

#### **a. Características de la señal**

La (Acuña, 2012) gama usable de la señal de salida de un FSR es casi lineal. Si se aplica bastante fuerza, su respuesta se convierte en no lineal debido a la saturación del sensor.

Aunque el FSR es considerado como sensor de presión no es recomendable que se use para mediciones reales de presión, ya que posee una baja exactitud, con errores de hasta un 25% de salida. Sin embargo, es excelente para aplicaciones donde se necesita una medición cualitativa.

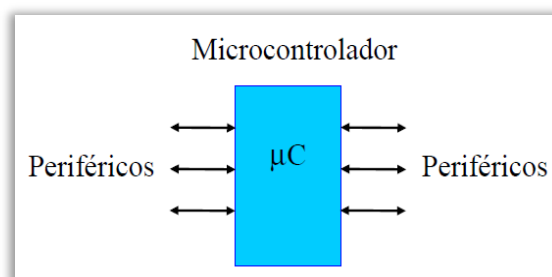
La exactitud de la fuerza se extiende de aproximadamente del  $\pm 5\%$  al  $\pm 25\%$  dependiendo de la consistencia de la medida y la actuación del sistema, de la tolerancia de la capacidad de repetición llevada a cabo en la fabricación, y de la calibración del sensor.

## 1.6 MICROCONTROLADOR

El microcontrolador (Angulo Martínez & Angulo Usategui, 2003, págs. 1-3) es un circuito integrado programable que contiene todos los componentes de un computador. Se emplea para controlar el funcionamiento de una tarea determinada y, debido a su reducido tamaño, suele ir incorporado en el propio dispositivo al que gobierna. Esta última característica es la que le confiere la denominación de «controlador incrustado» (embedded controller).

El microcontrolador es un computador dedicado. En su memoria sólo reside un programa destinado a gobernar una aplicación determinada; sus líneas de entrada/salida soportan el conexionado de los sensores y actuadores del dispositivo a controlar, y todos los recursos complementarios disponibles tienen como única finalidad atender sus requerimientos. Una vez programado y configurado el microcontrolador solamente sirve para gobernar la tarea asignada.

Un microcontrolador es un computador completo, aunque de limitadas prestaciones, que está contenido en el chip de un circuito integrado y se destina a gobernar una sola tarea. (Figura 1.33).



**Figura 1.33: El microcontrolador en un sistema cerrado.**

Todas las partes del computador están contenidas en su interior y sólo salen al exterior las líneas que gobiernan los periféricos.

## **1.6.1 LOS MICROCONTROLADORES PIC DE MICROCHIP**

### **a. CARACTERÍSTICAS GENERALES**

Los microcontroladores PIC (Mandado, Menéndez, Ferreira, & Matos, 2007, págs. 47-48) poseen las siguientes características generales:

#### **ARQUITECTURA INTERNA**

Utilizan la arquitectura Harvard que se caracteriza porque la memoria de programa genera la dirección de la memoria de datos. La memoria de datos está estructurada en palabras de ocho bits y la memoria de programa en palabras de 12, 14 ó 16 bits.

En la ejecución de las instrucciones se aplica la segmentación (pipeline) búsqueda y ejecución simultánea de instrucciones, lo que permite al procesador realizar al mismo tiempo la ejecución de una instrucción y la búsqueda de la siguiente. Esta característica hace que todas las instrucciones, excepto las de salto, se ejecuten siempre en un ciclo de instrucción.

#### **REPERTORIO DE INSTRUCCIONES**

Son procesadores del tipo RISC porque el número de instrucciones de su repertorio es reducido. El número de instrucciones es diferente para cada gama y varía entre 33 y 77.

#### **MEMORIA**

Poseen una memoria de datos volátil (RAM) de pequeña capacidad comprendida entre 25 y 3840 octetos, según el modelo, y direccionada por la memoria no volátil (EPROM, OTP, EEPROM o FLASH) que contiene el programa. Algunos modelos poseen además una memoria de datos no volátil (EEPROM).

## **PROGRAMACIÓN**

Su programación se realiza mediante un canal de comunicación serie denominado ICSP (In-Circuit Serial Programming), compuesto fundamentalmente por una línea de datos y otra de reloj.

## **1.7 TRANSMISIÓN DE DATOS**

### **1.7.1 SISTEMAS DE COMUNICACIÓN ELÉCTRICOS ELECTRÓNICOS**

La comunicación (Blake, 2004, págs. 1-3) fue una de las primeras aplicaciones de la tecnología eléctrica, su práctica empezó en 1873 con el sistema del telégrafo de Samuel Morse.

De manera general es posible definir a un sistema de comunicación como aquel que tiene como finalidad transmitir información desde un origen o fuente a un destino por medio de un canal.

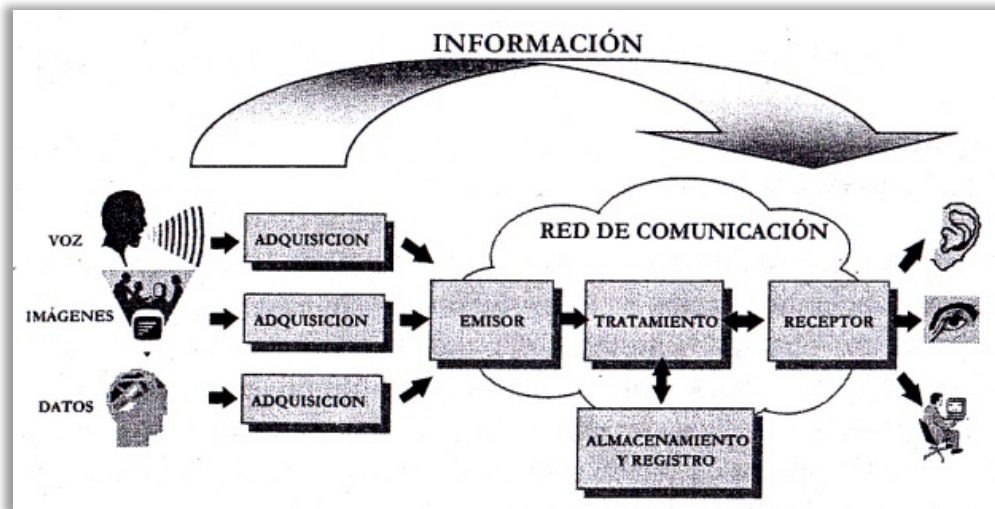
### **1.7.2 ELEMENTOS DE UN SISTEMA DE COMUNICACIÓN ELECTRÓNICO**

Todos los sistemas electrónicos de comunicaciones (Frenzel, 2003, págs. 4-7) tienen los componentes básicos mostrados en la Figura 1.34: un transmisor, un medio o canal de comunicación y un receptor. El proceso de comunicación empieza cuando alguien genera algún tipo de mensaje, datos u otra señal de inteligencia que debe ser recibida por los demás.

El mensaje, en la forma de una señal electrónica, es alimentado al transmisor, el cual se encarga de transmitirlo por medio de un canal de comunicaciones. El mensaje es captado por el receptor y transferido a



otro elemento. En este proceso se capta el ruido en el canal de comunicación y en el receptor. Ruido es el término general aplicado a cualquier fenómeno que degrada o interfiere la señal de información transmitida.



**Figura 1.34: Modelo de un sistema de comunicaciones genérico.**

#### **a. TRANSMISOR**

El primer paso para enviar un mensaje es convertirlo en una forma electrónica adecuada para su transmisión.

El transmisor por sí mismo es una colección de componentes y circuitos electrónicos diseñados para convertir la señal eléctrica en una forma adecuada para transmitirse a través de un medio de comunicación determinado. La señal original de inteligencia, modula usualmente a una portadora senoidal de mayor frecuencia generada en el transmisor, a la combinación se le aumenta la amplitud mediante amplificadores dando por resultado una señal que es compatible con el medio seleccionado para su transmisión.

## **b. CANAL DE COMUNICACIONES**

El canal de comunicaciones es el medio por el cual la señal electrónica se envía de un lugar a otro. En los sistemas de comunicaciones se utilizan muchos medios de diferentes tipos, como los siguientes:

**Conductores Eléctricos.** En su forma más sencilla, el medio puede ser un par de alambres que llevan la señal de voz de un micrófono a unos audífonos; podría ser un cable coaxial como, o un cable con un par de hilos trenzados, utilizado en una red de área local para las computadoras personales.

**Medio óptico.** El medio de comunicación puede ser un cable de fibra óptica que transmite el mensaje en una onda de luz. En la actualidad dicho cable es ampliamente utilizado por las compañías telefónicas y de internet para la transmisión de gran cantidad de información a largas distancias y elevado ancho de banda.

**Espacio Libre.** Cuando éste es el medio, el sistema resultante se conoce como radio. Radio es el término general aplicado a cualquier forma de comunicación inalámbrica de un punto a otro. La radiocomunicación hace uso del espectro electromagnético. Las señales de inteligencia se convierten en campos eléctricos y magnéticos que se propagan libremente en el espacio a través de grandes distancias.

## **c. RECEPTORES**

Un receptor es una colección de componentes electrónicos que acepta el mensaje transmitido del canal y lo convierte en una forma inteligible para los humanos o para el sistema destino.

### 1.7.3 COMUNICACIONES RADIOELÉCTRICAS

Las comunicaciones radioeléctricas (Roldán Martínez, 2005, págs. 1-2) o redes inalámbricas se caracterizan por el empleo del aire y las ondas de radio como soporte de comunicación.

A diferencia de lo que ocurre con sus homólogas cableadas, no requieren un medio físico, como un cable de cobre o una fibra óptica, para el establecimiento de la comunicación.

En efecto la idea que subyace a toda red radio es la de conectividad total, tanto temporal que es conexión disponible en cualquier momento, como espacial que se refiere conexión disponible en cualquier lugar.

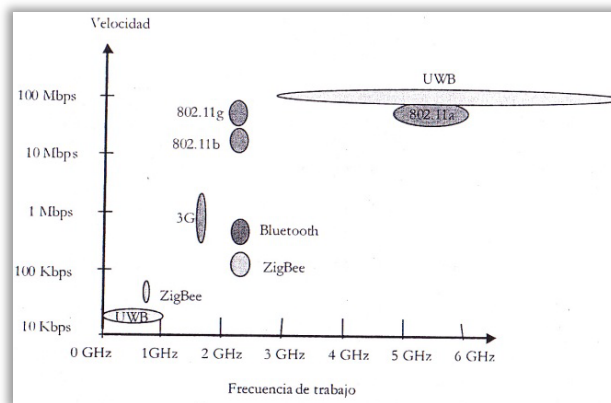
La tabla 1.1 resume los aspectos como ventajas, inconvenientes y seguridad que poseen las redes inalámbricas. No obstante, siempre es posible combinar en un mismo entorno una red radio con una cableada y aprovecharse de las ventajas que ofrecen ambas.

**Tabla 1.1: Comparación entre las redes radio y las redes cableadas convencionales.**

	<b>Red radio</b>	<b>Red cableada</b>
<b>Costes de instalación</b>	Medio	Alto
<b>Movilidad</b>	Sí	No
<b>Flexibilidad</b>	Muy alta	Baja
<b>Escalabilidad</b>	Alta	Muy alta
<b>Seguridad</b>	Alta	Alta
<b>Demanda</b>	Baja	Muy alta
<b>Config. e instalación</b>	Fácil	Media
<b>Pres. en empresas</b>	Baja	Alta
<b>Integridad</b>	Alta	Alta
<b>Licencia</b>	Depende de los casos	No

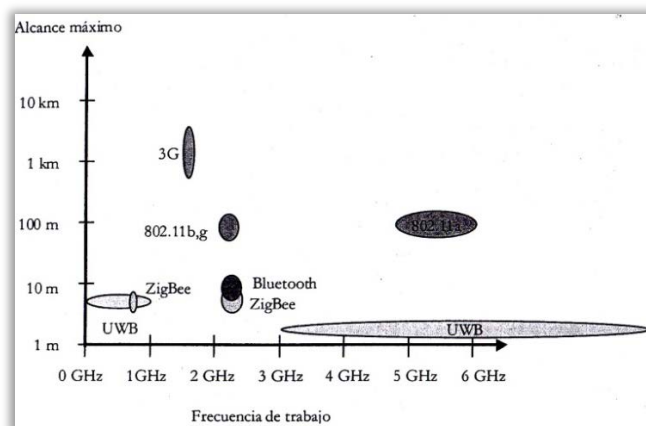
### 1.7.4 COMPARACIÓN DE LAS TECNOLOGÍAS INALÁMBRICAS

Desde el punto de vista velocidad (Roldán Martínez, 2005, págs. 7-9), se encuentran (véase Figura 1.35) la banda estrecha (10-100 Kbps), banda media (100-2000 Kbps) y banda ancha inalámbrica (2-100 Mbps).



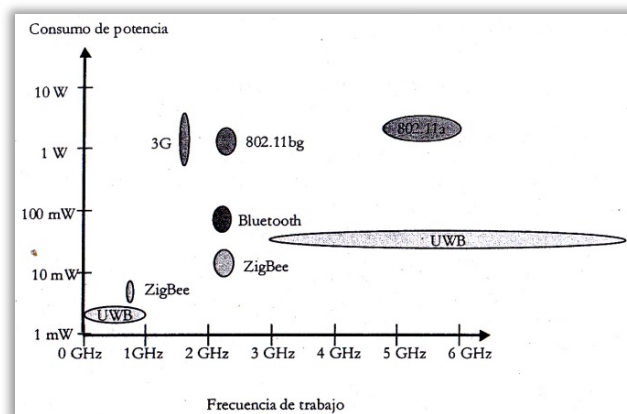
**Figura 1.35: Velocidad de las tecnologías inalámbricas según su anchura de banda.**

Los requisitos de alcance que una aplicación exija determinan otro punto en la selección de la tecnología. Como norma general, cuanto mayor sea la frecuencia de trabajo, menor será la distancia máxima que la tecnología en cuestión es capaz de cubrir. (véase Figura 1.36).



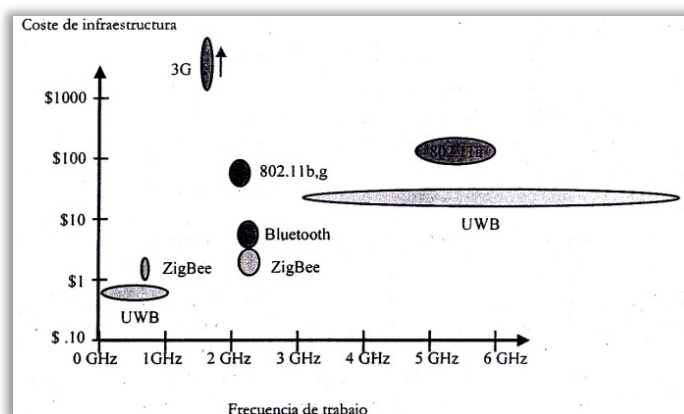
**Figura 1.36: Alcance de las tecnologías inalámbricas**

Por otra parte, existen aplicaciones en las que el consumo de potencia de los dispositivos inalámbricos constituye un factor determinante a la hora de escoger una tecnología. Como se indica en la Figura 1.37, cuando mayor sea la frecuencia de operación mayor será la energía que se requiere para hacer funcionar el dispositivo.



**Figura 1.37: Consumo de potencia de las tecnologías inalámbricas,**

Una de las ventajas de las comunicaciones inalámbricas, independientemente de la tecnología concreta, es su bajo coste de infraestructura en comparación con los tradicionales sistemas de comunicación basados en cableado. (véase Figura 1.38)



**Figura 1.38: Costo de la infraestructura de las tecnologías inalámbricas.**

### **1.7.5 ZIGBEE (IEEE 802.15.4)**

En algunas aplicaciones (Roldán Martínez, 2005, pág. 210) en las que participan pequeños dispositivos como sencillos sensores o actuadores no suelen ser necesarias altas tasas binarias. Estos sistemas se caracterizan por comunicaciones esporádicas en las que el mayor volumen de información apenas supera unas pocas decenas o centenas de kilobits por segundo y por requerir un alcance limitado a pocas decenas de metros, lo que facilita la portabilidad de la red y la instalación de este tipo de dispositivos. En estos entornos, por otra parte, resulta más crítico minimizar costes, tamaño y consumo de potencia de los dispositivos.

Zigbee (Oyarce, 2009) es un protocolo de comunicaciones inalámbrico basado en el estándar de comunicaciones para redes inalámbricas IEEE 802.15.4, permite que dispositivos electrónicos de bajo consumo puedan realizar sus comunicaciones inalámbricas. Es útil para redes de sensores en entornos industriales, médicos y, sobre todo, domóticos.

Las comunicaciones Zigbee se pueden realizar en tres bandas 868 MHz para Europa, 915 MHz para Norte América y 2.4 GHz para todo el mundo. A diferencia de bluetooth, este protocolo realiza las comunicaciones a través de una única frecuencia, es decir, de un canal. Normalmente puede escogerse un canal de entre 16 posibles para la banda libre de 2.4 GHz, 10 canales para la banda de 915 MHz y 1 solo canal para 868 MHz. El alcance depende de la potencia y antena del dispositivo. La velocidad de transmisión de datos de una red Zigbee es de hasta 256kbps. Una red Zigbee la pueden formar, en teoría, hasta 65535 equipos, es decir, el protocolo está preparado para poder controlar en una red esta cantidad de dispositivos. La tabla 1.2 resume las características de Zigbee.

Una red Zigbee la forman básicamente 3 tipos de elementos. Un único dispositivo Coordinador, dispositivos Routers y dispositivos finales.

### **El Coordinador.**

Es el nodo de la red que tiene la única función de formar una red. Es el responsable de establecer el canal de comunicaciones y del PAN ID (identificador de red) para toda la red. Una vez establecidos estos parámetros, el Coordinador puede formar una red, permitiendo unirse a él a dispositivos Routers y End Points. Una vez formada la red, el Coordinador hace las funciones de Router.

### **Los Routers.**

Es un nodo que crea y mantiene información sobre la red para determinar la mejor ruta para transmitir un paquete de información. Lógicamente un router debe unirse a una red Zigbee antes de poder actuar como Router retransmitiendo paquetes de otros routers o de End points.

### **End Device.**

Los dispositivos finales no tienen capacidad de enrutar paquetes. Deben interactuar siempre a través de su nodo padre, ya sea este un Coordinador o un Router, es decir, no puede enviar información directamente a otro end device. Normalmente estos equipos van alimentados a baterías. El consumo es menor al no tener que realizar funciones de enrutamiento.

**Tabla 1.2: Características básicas de Zigbee**

Velocidad de transmisión	250 Kbps (2,4 GHz), 40 Kbps (915 MHz) y 20 Kbps (868 MHz)
Acceso al medio	CSMA-CA
Topologías	Estrella, punto a punto, mallada
Espacio de direcc.	64 bits
Alcance	Entre 5 y 500 m dependiendo del entorno, aunque un valor típico son 50 m.

## **1.8 INTERFAZ HUMANO MÁQUINA**

El HMI es un sistema que sirve de interfaz entre el humano y la máquina, estos suelen usar indicadores y comandos, pilotos, paneles, pulsadores, displays LCD, pantallas TouchScreen, etc., en la actualidad estos procesos están implementados con controladores y dispositivos electrónicos y que dejan disponibles puertas de comunicación.

### **1.8.1 SOFTWARE HMI**

Los software HMI permiten realizar varias funciones entre las que se encuentran las interfaces graficas de modo de poder ver el proceso e interactuar con el registro en tiempo real de modo que se puede manejar el sistema desde una interfaz interactuando con el proceso.

Al igual que en las terminales, los operadores requieren de una herramienta de diseño o desarrollo, el cual es usado para configurar la aplicación deseada, esperando que este haga o realice el proceso desde dispositivos externos, o bien lo puede hacer por medio de una interfaz o tarjeta de adquisición de datos.

### **1.8.2 TIPOS DE HMI (INTERFAZ HUMANO MÁQUINA)**

En general podemos describir dos grandes clasificaciones: aquellas cuyo software de programación es propio de cada marca y sus limitantes están fijadas por esta misma, ejemplo de ellos serían CTC, GOT1000, Panel View.

Otras nos permiten una mayor versatilidad ya que trabajan en ambiente windows y se pueden programar en lenguajes como Visual C, C++,



Labview, entre otros, y por lo mismo sus capacidades son mayores, ejemplo de estas serían HPX, computadoras industriales entre otras.

### **1.8.3 LABVIEW**

LabView (Albornoz Carrasco & Lagos Norambuena, 2008, págs. 4-6) es el acrónimo de Laboratory Virtual Instrument Engineering Workbench. Es un lenguaje y a la vez un entorno de programación gráfica en el que se pueden crear aplicaciones de una forma rápida y sencilla.

LabVIEW es un revolucionario ambiente de desarrollo gráfico con funciones integradas para realizar adquisición de datos, control de instrumentos, análisis de mediciones y presentaciones de datos. LabVIEW da la flexibilidad de un poderoso ambiente de programación sin la complejidad de los ambientes tradicionales.

LabVIEW entrega herramientas poderosas para crear aplicaciones sin líneas de texto de código. Con LabVIEW se puede colocar objetos ya construidos para rápidamente crear interfaces de usuario. Después se especifica la funcionalidad del sistema armando diagramas de bloques.

Originalmente este programa estaba orientado a aplicaciones de control de instrumentos electrónicos usadas en el desarrollo de sistemas de instrumentación, lo que se conoce como instrumentación virtual.

Por este motivo los programas creados en LabVIEW se guardarán en ficheros llamados VI y con la misma extensión, que significa instrumento virtual (Virtual Instruments).

También relacionado con este concepto se da nombre a sus dos ventanas principales: un instrumento real tendrá un Panel Frontal donde estarán sus botones, pantallas, etc. y una circuitería interna. En LabVIEW estas

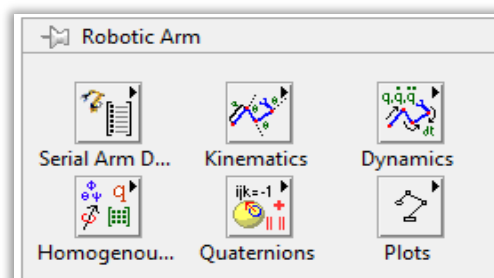
partes reciben el nombre de Panel Frontal y Diagrama de Bloques respectivamente.

En general LabVIEW incluye una diversidad de módulos y toolkits para distintas aplicaciones, en el control, simulación, robótica, interfaz de sensores y actuadores, protocolos de comunicación, herramientas para sistemas de visión, entre muchos más dependiendo de la aplicación específica.

### a. ROBOTICS MODULE

El Módulo LabVIEW Robotics (ni.com, 2013) es un paquete de software que le permite desarrollar aplicaciones de robótica con LabVIEW. El módulo de robótica proporciona algoritmos relacionados con la robótica, los controladores que permiten controlar ciertos sensores, hardware y software de asistentes de configuración y VIs de ejemplo que demuestran conceptos de robótica.

LabVIEW Robotics ofrece muchas funciones que son útiles para los brazos robóticos como cinemática, dinámica, y la generación de trayectoria. Las bibliotecas de los brazos robóticos (Figura 1.39) son útiles para la simulación, así como para el análisis de los resultados de los experimentos con robots reales.



**Figura 1.39: Paleta Robotic Arm incluida en LabVIEW Robotics.**

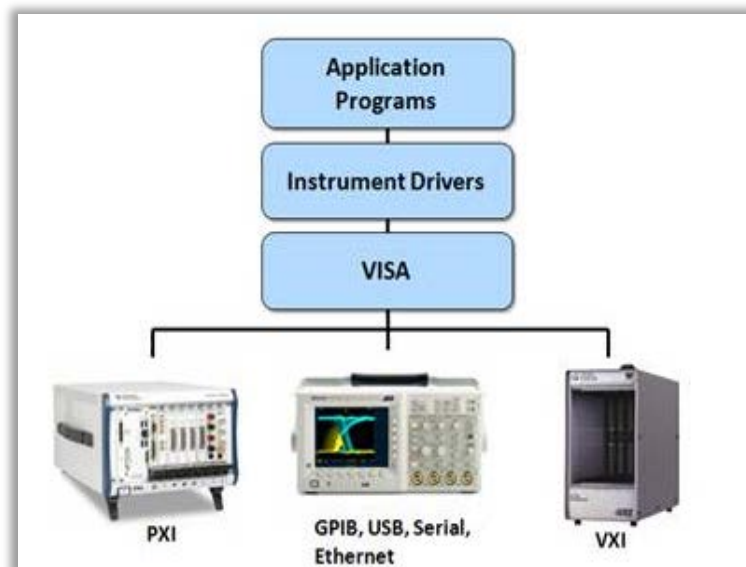
## b. NI-VISA

VISA (Virtual Instrument Software Architecture) (Lajara Vizcaíno & Pelegrí Sebastián, 2011) es una librería desarrollada por varios fabricantes de equipos que proporcionan un software estándar para las operaciones de lectura, escritura y manejo de eventos de instrumentación.

La gran ventaja de VISA es que el medio de comunicación es transparente para el programador, es decir, un programa no tendría que cambiar nada tanto si se usa RS-232 como si se usa GPIB o TCP/IP.

NI-VISA es la implementación de National Instruments de este estándar. Puede establecer comunicaciones a través de GPIB, serie, PXI, VXI o Ethernet.

Como VISA soporta varias interfaces de comunicación, suele ser el método elegido para crear drivers de control de equipos reales, véase la Figura 1.40.



**Figura 1.40: NI-VISA Interfaz común entre drivers e instrumentos reales.**

## **1.9 MATERIALES PELIGROSOS**

Un material peligroso (Wikipedia, 2012) es toda sustancia sólida, líquida o gaseosa que por sus características físicas, químicas o biológicas puede ocasionar daños al ser humano, al medio ambiente y a los bienes. También llamado por su sigla en inglés Hazmat (hazard material).

### **1.9.1 CLASIFICACIÓN DE LOS MATERIALES PELIGROSOS**

La Organización de las Naciones Unidas clasifica en esta forma los materiales y desechos peligrosos:

#### **a. Clase 1: Explosivos**

Comprende de sustancias explosivas, artículos explosivos y sustancias que producen efecto explosivo pirotécnico. Se subdivide en seis subclases:

- Materiales y artículos que presentan riesgo de explosión de toda la masa (como la nitroglicerina y la dinamita).
- Materiales y artículos que presentan riesgo de proyección, pero no de explosión de toda la masa.
- Materiales y artículos que presentan riesgo de incendio y de que se produzcan pequeños efectos de onda de choque o proyección, pero no un riesgo de explosión de toda la masa.
- Materiales y artículos que no presentan riesgos notables. Generalmente se limita a daños en el embalaje.
- Materiales muy poco sensibles que presentan riesgo de explosión de toda la masa pero que la posibilidad de explosión es remota.

- Materiales extremadamente insensibles que no presentan riesgo de explosión de toda la masa.

#### **b. Clase 2: Gases**

Se refiere a cualquier tipo de gas comprimido, licuado o disuelto bajo presión. Se distinguen tres subclases:

- Gases inflamables. Incluyen generalmente a hidrocarburos procedentes de la destilación del petróleo o de fuentes de gas natural (propano, hidrógeno).
- Gases no inflamables, no venenosos y no corrosivos. Son gases que no se queman con facilidad, y la combustión puede llevarse a cabo solo en condiciones extremas (nitrógeno, helio).
- Gases venenosos. Conformado por mezclas estables de gases, pero capaces de reaccionar con los compuestos orgánicos de las células produciendo la muerte (Cloro, fosgeno).
- Gases corrosivos.

#### **c. Clase 3: Líquidos**

Son líquidos, mezclas de líquidos, o líquidos conteniendo sólidos en solución o suspensión, que liberan vapores inflamables a temperaturas relativamente bajas. Estas se clasifican de acuerdo al Punto de Inflamabilidad, esto es, la temperatura más baja a la que el líquido desprende vapores en cantidad suficiente para formar una mezcla inflamable en las proximidades de su superficie (gasolina).

- Punto de inflamabilidad bajo (inferior a  $-18^{\circ}\text{C}$ ).

- Punto de inflamabilidad medio (igual o superior a  $-18^{\circ}$  C e inferior a  $23^{\circ}$  C)
- Punto de inflamabilidad alto (igual o superior a  $23^{\circ}$  C e inferior a  $61^{\circ}$  C)

En esta clase también se incluyen igualmente las materias sólidas en estado fundido cuyo punto de inflamación es superior a  $61^{\circ}$  C y que sean entregadas al transporte o transportadas en caliente a una temperatura igual o superior a su punto de inflamación.

También se incluyen las materias líquidas explosivas desensibilizadas (materias líquidas explosivas preparadas en solución o en suspensión en agua o en otros líquidos de modo que formen una mezcla líquida homogénea exenta de propiedades explosivas).

#### **d. Clase 4: Sólidos**

Incluye a las sustancias espontáneamente inflamables y sustancias que en contacto con el agua emiten gases inflamables. Son las sustancias que se encienden con facilidad, y que en consecuencia representan un peligro de incendio bajo las condiciones industriales normales.

- Sólidos inflamables. Son sólidos que en condiciones normales de transporte son inflamables y pueden favorecer incendios por fricción (magnesio, Fósforo rojo).
- Sustancias que pueden presentar combustión espontánea. Son espontáneamente inflamables en condiciones normales de transporte o al entrar en contacto con el aire (Fósforo blanco).
- Sustancia que en contacto con el agua despiden gases inflamables o tóxicos (sodio, potasio).

#### **e. Clase 5: Oxidantes**

- Oxidantes. Son sustancias que, aun sin ser combustibles, causan o contribuyen a la combustión al liberar oxígeno. No se confunda con las sustancias oxidantes o receptoras de electrones en reacciones químicas (ver Reducción-oxidación)(nitrato de amonio, peróxido de hidrógeno).
- Peróxidos orgánicos. Compuestos orgánicos con estructura bivalente O-O, térmicamente inestables, capaces de descomponerse en forma explosiva y violenta. Son sensibles al calor o a la fricción.

#### **f. Clase 6: Venenos**

- Sustancias venenosas. Son sólidos o líquidos que pueden causar efectos graves y perjudiciales para la salud del ser humano si se inhalan sus vapores, se ingieren o entran en contacto con la piel o las mucosas (cianuro de potasio, Cloruro de mercurio (I), Cloruro de mercurio (II)).
- Sustancias infecciosas. Son materiales que contienen microorganismos patógenos viables o toxinas de los que se sabe o se sospecha pudieran originar enfermedades en humanos y en animales (ántrax, VIH).

#### **g. Clase 7: Radiactivos**

Se entiende por material radiactivo a todos aquellos que poseen una actividad mayor a 70 kBq/kg (kilobequerelios por kilogramo) o su equivalente de 2 nCi/g (nanocurios por gramo) (Uranio, Plutonio).

#### **h. Clase 8: Corrosivos**

Son sustancias ácidas o básicas que causan lesiones visibles en la piel y otros tejidos vivos o corroen los metales. Algunas de estas sustancias son volátiles y desprenden vapores irritantes; pueden desprender gases tóxicos cuando se descomponen (hidróxido de Sodio, ácido sulfúrico).

#### **i. Clase 9: Mezclas peligrosas**

Son sustancias que presentan peligros para el hombre y el medio ambiente, pero sus efectos sobre éstos no clasifican como ninguna de las clases anteriores (por ejemplo el hielo seco).

- Cargas peligrosas que están reguladas en su transporte pero no pueden ser incluidas en ninguna de las clases antes mencionadas (asfalto caliente).
- Sustancias peligrosas para el medio ambiente.
- Residuos peligrosos.



## **CAPÍTULO II**

### **DISEÑO E IMPLEMENTACIÓN**

Una vez comprendidas las nociones básicas necesarias para la realización de este proyecto, en este capítulo se detalla la implementación e instrumentación tanto del guante sensorizado como del manipulador robótico, así como todo el hardware y software necesario para el control teleoperado del robot.

#### **2.1 DESCRIPCIÓN DEL PROYECTO**

El presente proyecto consta de dos partes principales la estación local y la estación remota, cada una formada por los siguientes componentes, que en conjunto permiten la teleoperación del manipulador robótico.

En la estación local se encuentra el guante sensorizado, el cual incorpora acelerómetros y un sensor flexo resistivo que detectan los movimientos del brazo humano y los transforman en señales eléctricas. Además se dispone de un dispositivo manual con un stick analógico y un pulsador de hombre-muerto.

Las señales eléctricas de los sensores del guante y del dispositivo manual son tratadas por un microcontrolador para posteriormente enviar la información de los sensores inalámbricamente mediante un módulo XBee. Hasta este punto el operador humano que realice la teleoperación se encuentra en una posición completamente independiente al lugar en donde se encuentre el manipulador robótico, pudiendo operarlo de manera remota.

Las señales inalámbricas que viajan a través del aire son captadas en la estación remota por un segundo módulo XBee, éste las transmite a otro

microcontrolador que tiene la finalidad de interpretar estas señales y enviar a la tarjeta controladora de servomotores la información necesaria sobre la posición que debe tomar el robot.

Adicionalmente se dispone de un HMI en un computador, gracias al cual es posible observar en tiempo real al robot manipulador a través de imágenes de video proporcionadas por cámaras ubicadas en la estación remota. También con éste HMI se puede realizar un control directo del manipulador robótico. El computador con la aplicación HMI se comunica con la tarjeta electrónica de la estación remota haciendo uso también de la tecnología XBee. (véase Figura 2.1)



**Figura 2.1: Diagrama general del proyecto.**

## 2.2 ESTACIÓN LOCAL

### 2.2.1 ARQUITECTURA DEL SISTEMA

El modelo de guante es un dispositivo provisto de sensores capaz de determinar la posición en cinco movimientos de las articulaciones del

brazo diestro. Para el control del movimiento de la base del robot se adiciona un stick analógico (palanca de dirección) en un dispositivo para la mano izquierda el cual incluye también un pulsador de hombre-muerto. La señal de los sensores y del joystick es acondicionada, procesada y enviada solo si el hombre muerto se encuentra presionado.

El hardware está conformado por tres acelerómetros MMA7361L y un Sensor Flex, montados estratégicamente en un guante diestro y una codera, también se implementa una tarjeta electrónica que será conocida de aquí en adelante como “PCI 1”, la cual aloja un microcontrolador PIC18F452, un módulo inalámbrico XBee Serie 1 y demás elementos electrónicos necesarios para la adquisición y transmisión de datos.

El microcontrolador previamente mencionado realiza la conversión de las señales analógicas acondicionadas de los sensores por medio del módulo conversor A/D (análogo-digital).

Una vez digitalizados los datos estos son procesados por un algoritmo lógico y actualizados continuamente para posteriormente ser transmitidos a través de la interfaz inalámbrica.

## **2.2.2 ACELERÓMETRO MMA7361L**

Los acelerómetros elegidos son de la compañía Freescale Semiconductor, estos acelerómetros son del tipo capacitivos y basan su funcionamiento en la variación de la capacidad entre dos o más conductores (entre los que se encuentra un dieléctrico), en respuesta a la variación de la aceleración.

El acelerómetro utilizado es un circuito integrado miniaturizado y de montaje en superficie sobre el PCB.

## a. CARACTERÍSTICAS DEL ACELERÓMETRO MMA7361L

Este sensor de tres ejes requiere muy poca energía y puede ser configurado para realizar mediciones entre  $\pm 1.5g$  y  $\pm 6g$ . Otras características que posee son modo sleep, acondicionamiento de señal, filtro pasa bajo de un polo, compensación de temperatura, autoprueba, y detección de 0g para caída libre.

Este sensor funciona con alimentaciones entre 2.2 y 3.6VDC (3.3V es el valor óptimo), y consume solamente 400 $\mu$ A de corriente. Los tres ejes tienen su propia salida analógica. A continuación se detallan las características del acelerómetro:

- Medición de Tres ejes X, Y, Z
- Dimensiones: 3mm x 5mm x 1.0mm LGA-14 Paquete
- Bajo consumo de corriente: 400  $\mu$ A
- Modo de espera: 3  $\mu$ A
- Funcionamiento de baja tensión: 2.2 V – 3.6 V
- Alta sensibilidad (800 mV/g a 1.5g)
- Sensibilidad seleccionable ( $\pm 1.5$  g,  $\pm 6$  g)
- Rápido tiempo de encendido (0.5 ms de tiempo de respuesta)
- Autoprueba para detección de caída libre
- Detección de 0g para caída libre.
- Acondicionamiento de Señal con filtro de paso bajo
- Diseño robusto de alta resistencia al impacto.

En nuestro caso, implementaremos una configuración por defecto de 1.5g y 800mV/g para el rango de aceleraciones y la sensibilidad respectivamente.

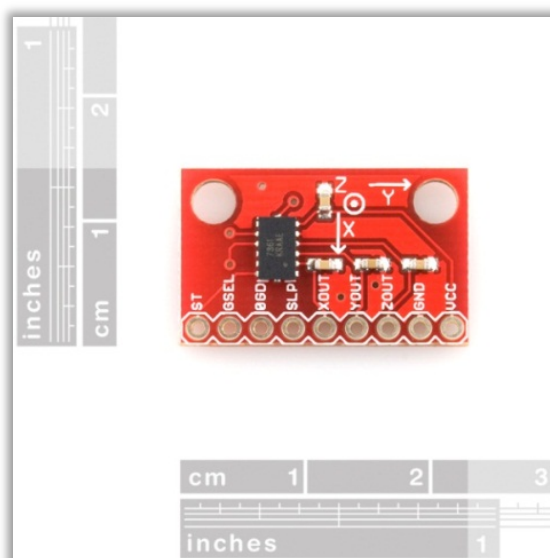
En lo que al consumo de energía se refiere, se alimenta el dispositivo con una tensión de alimentación general VCC (3.3V), común para todos los dispositivos inerciales presentes en el proyecto.

## b. Tarjeta Adaptadora (PCB).

Los sensores basados en tecnología MEMS son sensibles a los procesos de soldadura de las placas de circuito impreso (PCB). Para obtener el óptimo valor de offset del cero-g, se debe tener cuidado con el diseño de la PCB y las condiciones de soldadura, el montaje en la superficie de diseño de la placa es una parte crítica total del diseño.

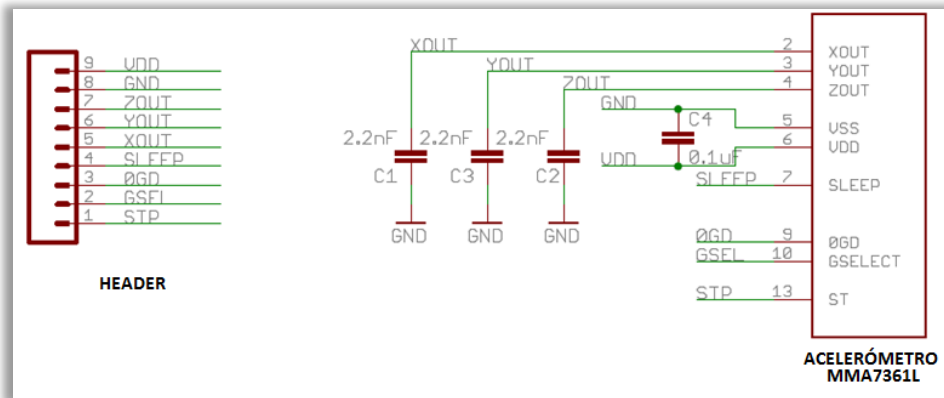
Para garantizar una interfaz adecuada de conexión de soldadura entre la placa y el acelerómetro (evitando la formación de puentes y cortocircuitos entre almohadillas de soldadura) se optó por adquirir de fábrica el acelerómetro previamente montado en su tarjeta adaptadora (PCB), con las garantías de obtener lecturas fiables.

La PCB utilizada es una tarjeta adaptadora amigable de la compañía Sparkfun, la misma que tiene disponibles todos los pines del acelerómetro MMA7361L en un header de 9 pines con espaciamiento de 0.1" para fácil montaje. Sus dimensiones son: 0.90 x 0.50". Figura 2.2.



**Figura 2.2: Acelerómetro MMA7361L montado en PCB marca Sparkfun.**

Para el uso de aplicaciones de consumo, el fabricante del acelerómetro recomienda ciertas pautas en el montaje de la interfaz del acelerómetro con el microcontrolador, que la tarjeta adaptadora ya contempla, véase la Figura 2.3.



**Figura 2.3: Esquema de la PCB Sparkfun.**

### Descripción De Pines

En la figura 2.4 se aprecia la distribución de pines del sensor MMA7361L montado en su PCB y a continuación la tabla 2.1 con la descripción específica de los pines del dispositivo.



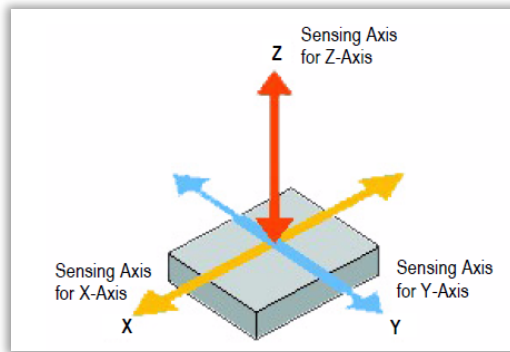
**Figura 2.4: Distribución de pines del acelerómetro MMA7361L montado en la PCB.**

**Tabla 2.1: Descripción de pines del acelerómetro MMA7361L montado en la PCB.**

<b>PIN N°</b>	<b>NOMBRE DEL PIN</b>	<b>DESCRIPCIÓN</b>
1	ST	Pin de entrada para iniciar la función de auto verificación (verificación de la integridad mecánica y eléctrica del acelerómetro).
2	GSEL	Selector de resolución, normalmente esta en bajo por lo tanto la resolución es de +/-1.5g (800mv/g) si se pone un alto de 3.3V se obtiene resolución de +/-6g (206mv/g).
3	0GD	Detección de caída libre (Cuando detecta caída libre se genera una señal de alto en este pin).
4	SLP	Entrada lógica para activar el modo de suspensión (sleep).
5	XOUT	Salida de lectura aceleración EJE X.
6	YOUT	Salida de lectura aceleración EJE Y.
7	ZOUT	Salida de lectura aceleración EJE Z.
8	GND	Tierra o GND.
9	VCC	Voltaje de alimentación.

**c. Direccionamiento del sensor:**

En la figura 2.5 se muestra como se establecen los ejes de coordenadas tridimensionales para poder realizar un correcto estudio y posicionamiento del acelerómetro.

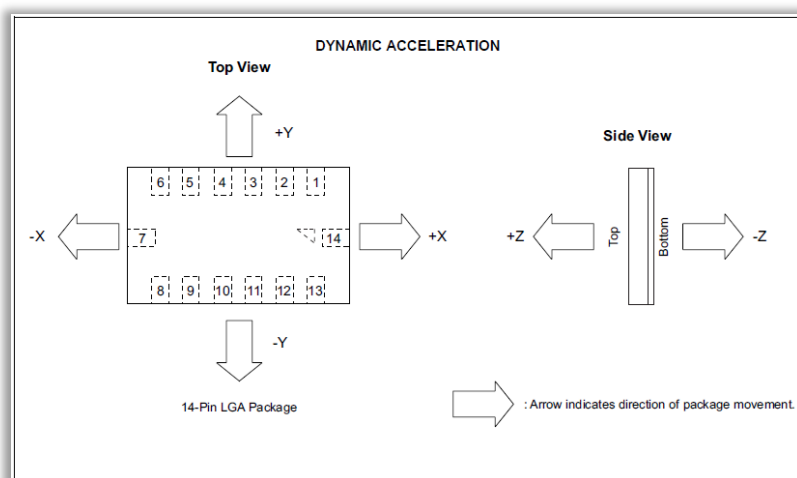


**Figura 2.5: Posición del sensor.**

El acelerómetro puede ser utilizado para medir tanto aceleraciones dinámicas (p.e; vibración) como estáticas (p.e; gravedad), según la aplicación particular requerida, éstos corresponden a los funcionamientos en presencia o ausencia de movimiento.

En cuanto al comportamiento dinámico, el sensor está preparado para medir señales de aceleración que varían rápidamente en el tiempo.

En la figura 2.6 se muestran las especificaciones dadas por el fabricante correspondiente a la orientación y el sentido de la aceleración dinámica en cada uno de los ejes.



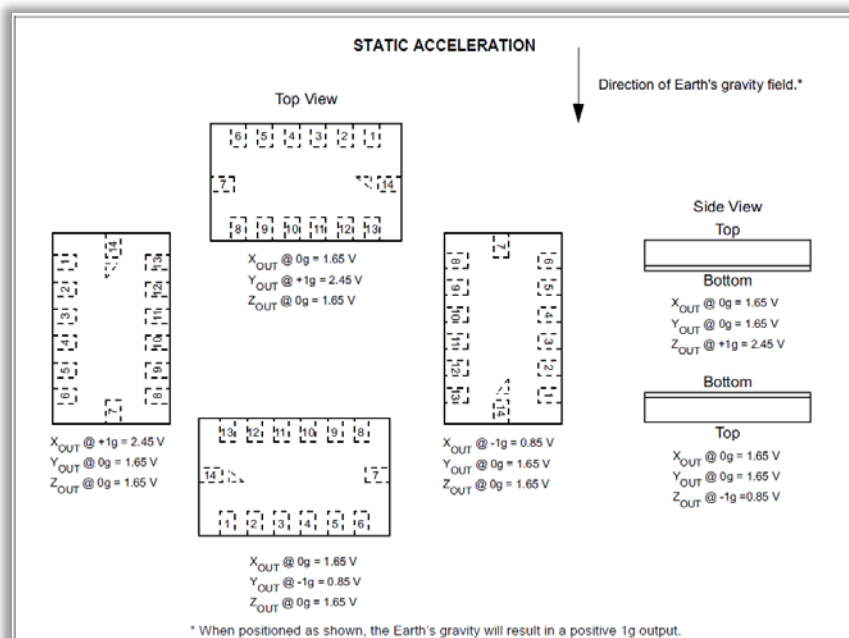
**Figura 2.6: Comportamiento Dinámico.**



Con respecto al comportamiento estático, el sensor ofrece respuesta ante aceleraciones que no varían en el tiempo o ante aceleraciones de muy baja frecuencia o frecuencia nula. La inclinación es una medición estática, donde se mide la aceleración de la gravedad y es la aplicación particular escogida para la realización del proyecto.

En la figura 2.7 se muestra un ejemplo de la orientación, sentido y magnitud en la medición de la aceleración estática, provista por el fabricante.

En caso de que el sensor se encuentre en la posición donde los ejes X e Y sean perpendiculares a la gravedad su valor de salida será de “0 g”, mientras que el valor del eje Z será de “+1 g” si la muesca es vista por el usuario, o “-1g” si la muesca no es vista. Para el resto de casos es similar solo hay que tener claro el sistema de coordenadas de la figura, donde se ilustra un ejemplo tomado en condiciones estándar (rango de 1.5g seleccionado y alimentación de 3.3V).



**Figura 2.7: Comportamiento estático.**

Nótese que jamás los tres ejes coincidirán en valor, ya que es imposible si el desfase entre ellos es de  $90^\circ$ . Por tanto, la igualdad de valores en las salidas será de dos ejes como máximo.

### 2.2.3 SENSOR FLEXO RESISTIVO

Para realizar el control de la pinza del robot manipulador se decidió que la apertura de la mano sea reflejada con precisión al servomotor. Para cumplir lo dicho se necesita de un dispositivo capaz de sensorar este movimiento y que además se preste para su fácil implementación en el guante.

Para ello se utilizó el sensor flexo resistivo “Flex Sensor” (Figura 2.8) de la compañía Spectra Symbol, el cual funciona de manera que al ser flexionado, su resistencia aumenta.

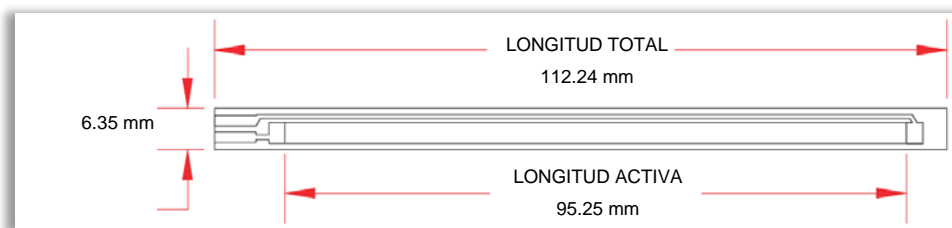


**Figura 2.8: Flex Sensor de Spectra Symbol**

La tecnología del Flex Sensor se basa en elementos resistivos de carbono que se presentan como una resistencia variable a la deformación, impresa sobre un sustrato delgado y flexible. Cuando el sustrato está doblado, el sensor produce una resistencia de salida correlacionada al radio de curvatura, mientras menor es el radio, mayor es el valor de la resistencia.

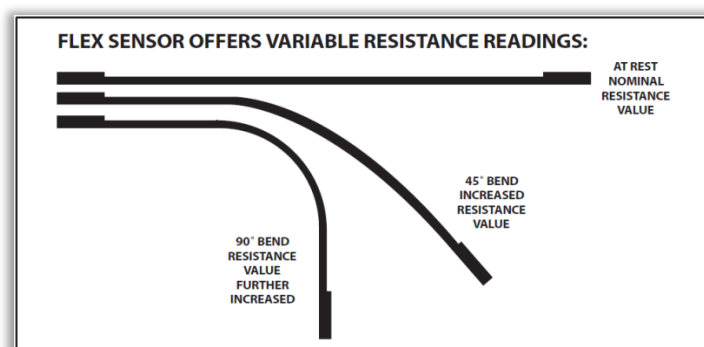
## a. CARACTERÍSTICAS Y ESPECIFICACIONES

El sensor presenta 10K en estado plano o nominal y tiene bajos requerimientos de energía. La resistencia puede aumentar hasta 5 veces su lectura en estado nominal. El perfil bajo del sensor de flexión permite que se envuelva alrededor de las superficies o para caber en espacios reducidos, sus dimensiones pueden ser apreciadas en la Figura 2.9. Con un ciclo de vida de más de 1 millón de flexiones, su durabilidad y forma es adecuada para nuestra aplicación.



**Figura 2.9: Diagrama de dimensiones del Flex Sensor.**

Este sensor es unidireccional, es decir que solo puede ser flexionado en una única dirección. Para obtener la correcta resistencia variable el sensor debe ser flexionado con las láminas resistivas hacia el exterior de la curva, su funcionamiento puede apreciarse en la Figura 2.10.



**Figura 2.10: Forma de funcionamiento del Flex Sensor.**

Las especificaciones del sensor se detallan en la siguiente tabla:

**Tabla 2.2: Especificaciones técnicas Flex Sensor**

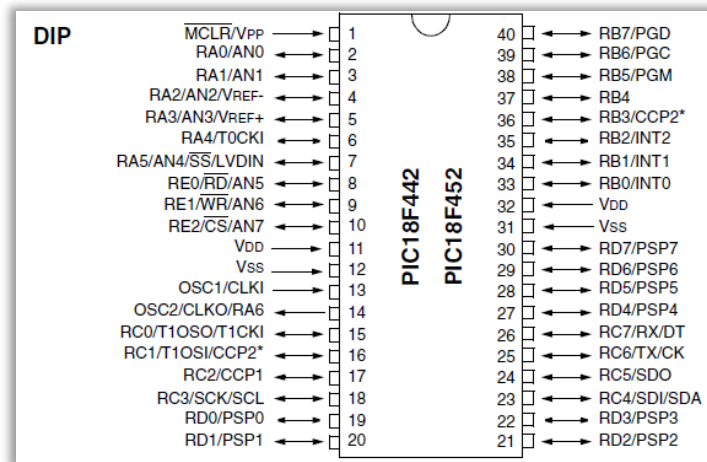
Ciclo de vida	1 millón
Grosor	$\leq 0.43\text{mm}$
Rango de Temperatura	$-35^{\circ}\text{C}$ a $80^{\circ}\text{C}$
Resistencia Nominal (Plano)	10K $\Omega$
Tolerancia de la Resistencia	$\pm 30\%$
Rango de Resistencia al Doblar	60K a 110K $\Omega$
Consumo de energía	0.50 Watts continuos. 1 Watt en picos

Para poder obtener la señal de voltaje que dependa de la variación de la resistencia del sensor se utiliza un divisor de voltaje, donde el sensor actúa como resistencia variable del circuito.

#### **2.2.4 MICROCONTROLADOR PIC18F452**

El microcontrolador PIC18F452 llamado posteriormente "PIC1", es de la empresa Microchip basado en tecnología CMOS. Su funcionamiento requiere bajo consumo de energía, cuenta con una mayor memoria de programa (32K) y 1.5K bytes para memoria de datos, además posee un amplio rango de tensión de 2.0V a 5.5V. Se seleccionó este microcontrolador debido a sus características y disponibilidad inmediata. La Figura 2.11 muestra la distribución de pines del PIC18F452.

El microcontrolador en esta etapa del proyecto tiene como finalidad adquirir las señales analógicas de los sensores montados en el guante, convertirlos en señales digitales para almacenarlos en su memoria, tratarlos mediante algoritmos de filtrado para finalmente enviarlos mediante comunicación serial al módulo XBee.



**Figura 2.11: Microcontrolador PIC18F452.**

En la siguiente tabla se resumen las características principales del microcontrolador PIC18F452.

**Tabla 2.3: Características principales PIC18F452.**

CARACTERÍSTICAS	PIC18F452
Frecuencia de operación	DC-40MHz
Memoria de programa Flash (bytes)	32K
Memoria de datos RAM (bytes)	1536
Memoria de datos EEPROM (bytes)	256
Número de Instrucciones (instrucciones)	16384
Fuentes de interrupción	18
Puertos de I/O	5 Puertos (A,B,C,D,E)
Timers	4
Módulos PWM (CCP)	2
Comunicación serial	MSSP (SPI/I2C), USART
Comunicación en paralelo	1 PSP
Módulo ADC (10 bits)	8 canales de entrada
Set de instrucciones	75 Instrucciones (14 bits cada una)
Corriente máxima en puertos	25mA
Encapsulado	40-pin DIP

## 2.2.5 MÓDULO XBEE

En la selección del dispositivo para la transmisión inalámbrica de datos se tomó en cuenta los siguientes requerimientos: bajo consumo de energía, hardware compacto, corto alcance entre estaciones

Se optó por utilizar el estándar de comunicaciones IEEE 802.15.4 para la transmisión inalámbrica de datos donde el dispositivo seleccionado para tal cometido fue el módulo XBee Serie 1 de la compañía MaxStream (véase Figura 2.12). Cabe recalcar que estos módulos XBee son los más populares en el mercado de la electrónica, caracterizados principalmente por ser económicos, poderosos y fáciles de usar.



**Figura 2.12: Módulo XBee Serie 1.**

### CARACTERÍSTICAS MÓDULO XBEE SERIE 1

El uso del estándar 802.15.4 va desde remplazar un cable por una comunicación serial inalámbrica, como lo es en el caso del presente proyecto, hasta el desarrollo de configuraciones punto a punto, multipunto, o redes complejas de sensores como las redes Mesh (protocolo Zigbee).

Actualmente existen dos series de módulos XBee, donde su principal diferencia es que la serie 2 puede hacer redes mesh y la serie 1 no. Por

consiguiente los dispositivos XBee Serie 1 se basan únicamente en el estándar de comunicaciones IEEE 802.15.4, que opera en las capas inferiores del modelo OSI.

Cada módulo XBee tiene una dirección única. En el caso de los módulos seleccionados cada uno de ellos tiene una dirección única de 64bits que viene grabada de fábrica.

El módulo XBee Serie 1 tiene un alcance en línea de vista de 100 metros y en interiores hasta 30 metros. Trabajan dentro de la banda libre de 2.4Ghz, por lo que son útiles para aplicaciones en cualquier lugar del mundo sin necesidad de licencias.

Entre las necesidades que satisface el módulo se hallan: bajo costo, ultrabajo consumo de potencia, instalación barata y simple. También existen los llamados módulos XBee PRO de la Serie 1 que se diferencian en la capacidad de alcance, permitiendo en algunos casos doblar la distancia de transmisión, ya que poseen una mayor potencia en la señal.

El módulo requiere una alimentación desde 2.8 a 3.4 V, la conexión a tierra y las líneas de transmisión de datos por medio del UART (TXD y RXD) para comunicarse con un microcontrolador, o directamente a un puerto serial utilizando algún conversor adecuado para los niveles de voltaje.

### **2.2.6 GUANTE**

Este tipo de dispositivo particular está formado por dos partes: un guante diestro y una codera.

Se considera este tipo de configuración porque no existe un dispositivo de una sola pieza que presente características de comodidad y facilidad para el posicionamiento de los sensores.

### **a. Características**

El guante diestro es de la marca Black Diamont, modelo HeavyWeigth, que está confeccionado en 300 g/m<sup>2</sup> de tejido Polartec Power Stretch, donde su capa exterior de nylon es duradera y lisa y su capa interior de poliéster es suave y cómoda, en ambas capas combinando la flexibilidad del elastán. El tejido patentado por la empresa Polartec presenta bielasticidad y es altamente transpirable, lo que provee al guante alta flexibilidad manteniendo seca la mano del usuario en todo momento con tan solo 58 g de peso. Además sus costuras están hechas con Kevlar, la cual es una fibra textil muy ligera con excepcional resistencia a la rotura, generalmente usada en chalecos antibalas.

La codera adquirida es una codera deportiva ajustable marca Dalps confeccionada con nylon elástico relleno de neopreno, materiales que hacen que la codera se ajuste cómodamente al portador manteniendo su flexibilidad y elasticidad.

Todas estas características hacen que la elección del guante y la codera sean las mejores para la concepción del dispositivo guante que será usado en el brazo derecho como medio de teleoperación.

### **b. Ubicación de los sensores**

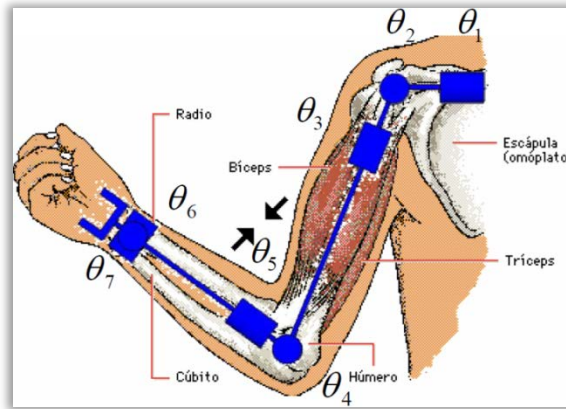
La disposición de todos los sensores fue determinada al analizar de manera simple la relación de los movimientos del brazo humano con los movimientos del robot manipulador.

#### **Disposición de los acelerómetros**

Para realizar el control de cada grado de libertad del manipulador robótico mediante el brazo humano, se optó por determinar la variación de los



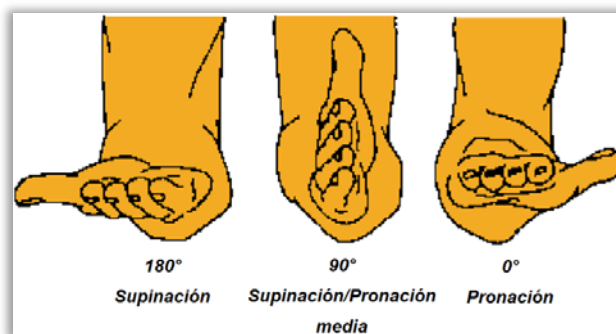
ángulos  $\theta_2$ ,  $\theta_5$ ,  $\theta_6$  y  $\theta_7$  que se muestran en la Figura 2.13 y así relacionarlos con los movimientos del robot.



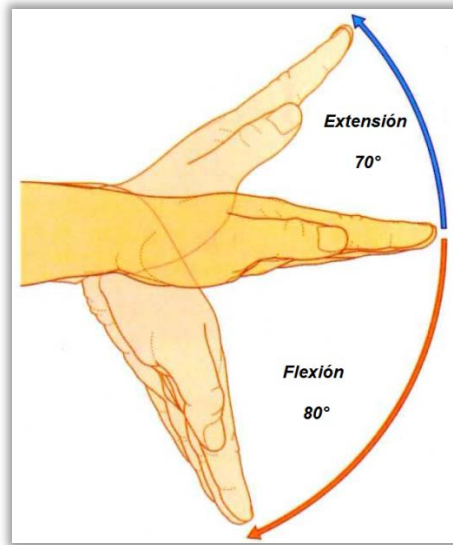
**Figura 2.13: Grados de libertad de las articulaciones del brazo humano.**

Para ello se utilizan 3 acelerómetros donde su ubicación es esencial debido a que deben estar correctamente relacionados con la ubicación de las articulaciones del brazo para poder sensor los movimientos de las mismas.

El acelerómetro 1 fue ubicado en la muñeca para que sea capaz de sensor los movimientos de flexión y extensión con el eje X, pronación y supinación con el eje Y. En las Figuras 2.14 y 2.15 se observa el rango de variación en ángulos de estos movimientos.

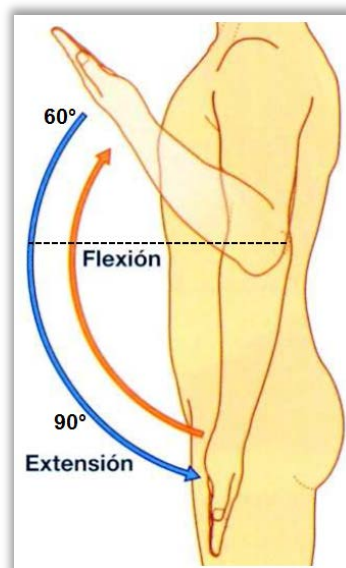


**Figura 2.14: Movimiento de la mano en pronación y supinación.**



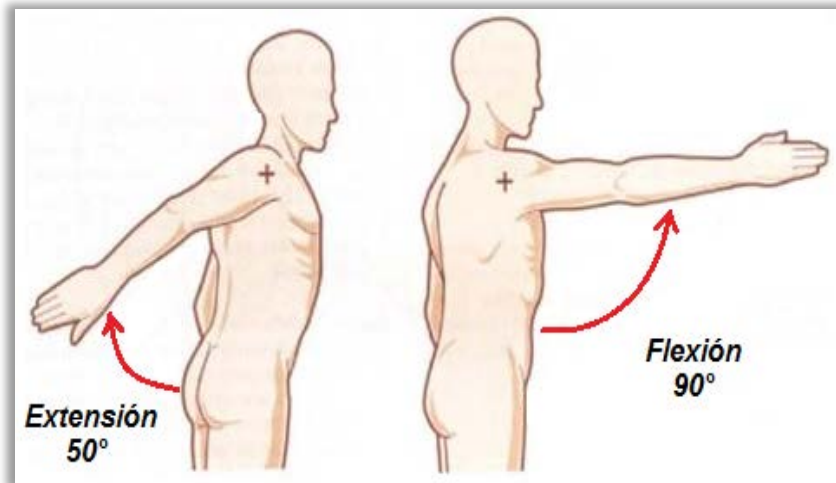
**Figura 2.15: Movimiento de la mano en Flexión y Extensión**

El acelerómetro 2 fue ubicado en el antebrazo, éste sensor utiliza únicamente el eje X para detectar la inclinación del antebrazo en los movimientos de flexión y extensión del codo, como se observa en la Figura 2.16.



**Figura 2.16: Movimientos del antebrazo: Flexión y extensión de la articulación del codo.**

El último sensor también utiliza el eje X y fue ubicado en el brazo para determinar la inclinación del mismo únicamente mientras se encuentra en flexión, ya que la extensión del brazo se realiza hacia atrás y este movimiento no es necesario para el control del robot. Figura 2.17.



**Figura 2.17: Movimientos del brazo: Flexión y extensión de la articulación del hombro.**

En la figura 2.18 se puede apreciar la disposición final de los acelerómetros montados sobre el dispositivo guante.



**Figura 2.18: Acelerómetros montados sobre el dispositivo guante.**

### Disposición del Flex Sensor.

El Flex sensor fue colocado en el guante sobre el dedo índice, para que de esta manera sea posible sentir el cierre y la apertura de la mano. Figura 2.19.



Figura 2.19: Ubicación del sensor flex sobre el guante.

### 2.2.7 NUNCHUK

Este dispositivo tiene dos finalidades, la primera es la lectura del movimiento en el eje Y del stick analógico utilizada para mover el servomotor de la base del robot. Y la segunda es la utilización del botón Z como pulsador de hombre-muerto para el sistema. Véase Figura 2.20.



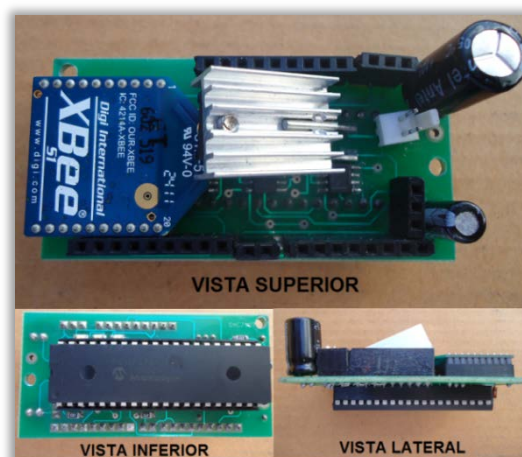
Figura 2.20: Nunchuk utilizado en la estación local.

El nunchuk es utilizado mediante la mano izquierda del operador y no representa ningún grado de dificultad a la hora de realizar la teleoperación.

## 2.2.8 PLACA ELECTRÓNICA PCI 1

La placa diseñada para usarse en la estación local, de manera general está compuesta de tres secciones principales: Acondicionamiento de señales, disposición de pines del microcontrolador y transmisión inalámbrica; y como en toda placa electrónica esta también dispone de una zona de alimentación.

Cada una de estas secciones se compone de diversos elementos electrónicos que se encuentran integrados en una única tarjeta electrónica con diseño a doble lado con unas dimensiones de 68x34mm. En la Figura 2.21 se puede observar la placa PCI 1 con todos sus elementos montados.



**Figura 2.21: Placa electrónica de la estación local.**

Los esquemas electrónicos fueron realizados en el software Proteus ISIS 7, los cuales se utilizaron para el diseño real de la placa gracias a las

herramientas del software Ares. En los siguientes literales se detallarán los elementos utilizados y sus conexiones en cada una de las secciones.

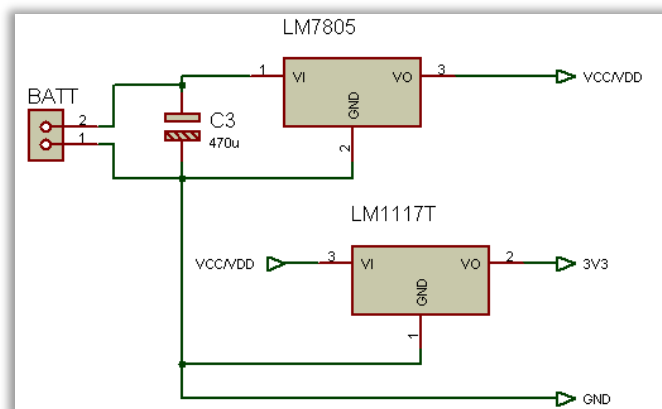
La energía necesaria para el funcionamiento de la tarjeta electrónica PCI 1 es suministrada por una batería recargable de Litio-Polímero de dos celdas marca Turnigy de 7.4 V y 2200 mAh. Véase Figura 2.22.



**Figura 2.22: Fuente de alimentación de la placa PCI 1.**

La batería se implementa con un circuito regulador de tensión para obtener los voltajes de alimentación constantes requeridos por los dispositivos de la tarjeta, los cuales son 5V y 3.3V.

En la Figura 2.23 se muestra el circuito regulador de tensión usado en la tarjeta PCI 1.



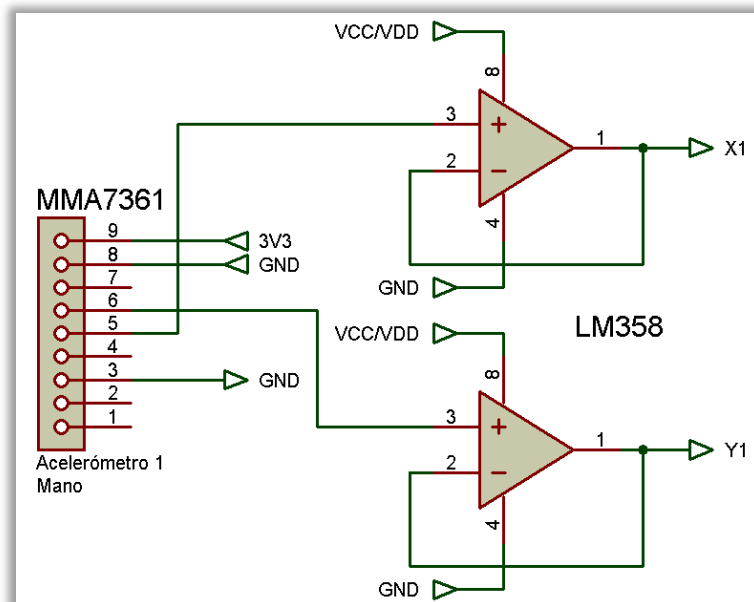
**Figura 2.23: Circuito regulador de tensión para la PCI 1.**

## a. ACONDICIONAMIENTO DE SEÑALES

En la placa PCI 1 se conectan extensiones de las señales obtenidas por los sensores (acelerómetros y sensor flex), por el stick analógico y un botón del nunchuk.

Como ya se detalló en un apartado anterior, de los tres acelerómetros incorporados en el guante, se obtienen y utilizan un total de cuatro señales analógicas.

Cada una de estas señales antes de conectarse al microcontrolador, pasan por un circuito de desacoplamiento de impedancias para evitar caídas de tensión y falsas lecturas de los acelerómetros. Dicho circuito está formado por un amplificador operacional en configuración seguidor de voltaje, uno por cada señal. En la figura a continuación se puede observar cómo se han realizado estas conexiones en uno de los acelerómetros.



**Figura 2.24: Circuito de desacople de impedancias para los acelerómetros.**

Las señales del sensor flex, el stick analógico y el botón del nunchuk se conectan directamente a las entradas del microcontrolador, como se observa en la Figura 2.25.

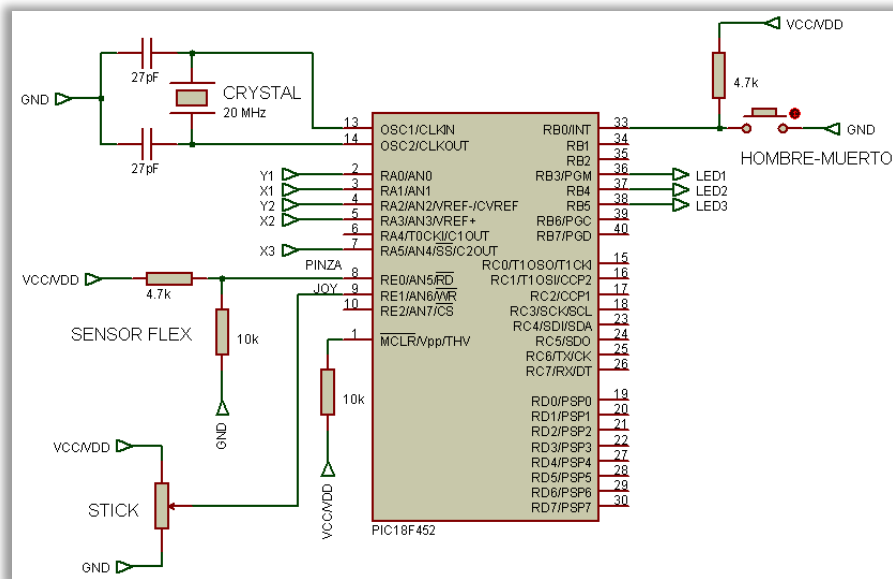
## **b. DISPOSICIÓN DE PINES DEL MICROCONTROLADOR**

El microcontrolador PIC18F452 de esta placa maneja todas las señales del guante sensorizado. En total al microcontrolador se conectan siete entradas análogas y cuatro elementos I/O del puerto B, en la Tabla 2.4 se detallan estas conexiones. En la Figura 2.25 se puede apreciar más claramente cómo se realizaron dichas conexiones.

**Tabla 2.4: Descripción de I/O utilizadas en el microcontrolador de la placa PCI 1.**

<b><i>Número de pin</i></b>	<b><i>Descripción del pin</i></b>	<b><i>Señal Conectada</i></b>
2	AN0	Acel. 1 eje Y / Mano (Y1)
3	AN1	Acel. 1 eje X / Mano (X1)
4	AN2	Acel. 2 eje Y / Codo (Y2)
5	AN3	Acel. 2 eje X / Codo (X2)
7	AN4	Acel. 3 eje X / Hombro (X3)
8	AN5	Sensor Flex (Pinza)
9	AN6	Stick Analógico (Joy)
33	RB0	Pulsador Hombre-Muerto
36	RB3	LED 1
37	RB4	LED 2
38	RB5	LED 3





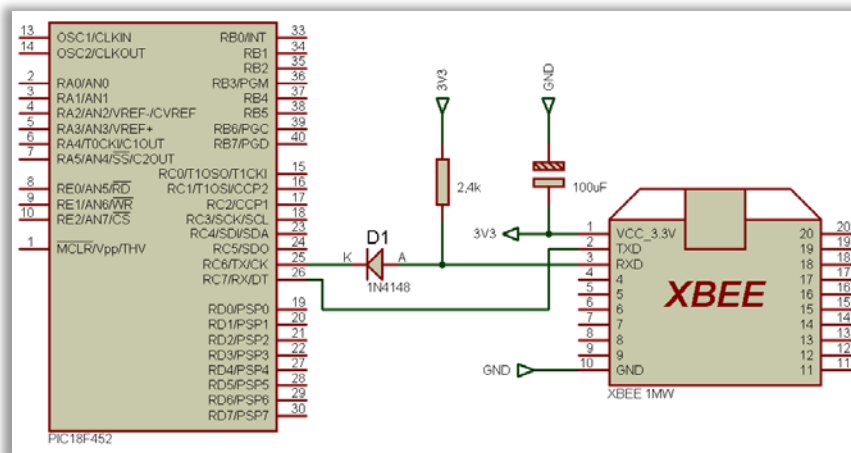
**Figura 2.25: Conexiones realizadas en el microcontrolador de la placa PCI 1.**

### c. CONEXIÓN DEL MÓDULO XBEE

Para la transmisión inalámbrica, el microcontrolador envía los datos de manera serial al módulo XBee, como la comunicación entre estos dos dispositivos no es a grandes distancias, no es necesaria la utilización de ningún estándar de transmisión serial.

Debido a que los niveles del voltaje en los que trabajan el microcontrolador y el XBee son distintos, la comunicación serial desde el microcontrolador hacia el módulo inalámbrico se realiza a través de un sencillo circuito que cambia el nivel de voltaje lógico de 5V a 3.3V (Figura 2.26).

Aunque no es necesaria la transmisión de datos desde el módulo XBee hacia el microcontrolador, se ha conectado el pin receptor del microcontrolador (RX) con el pin transmisor del XBee (TX) para que esté disponible si se llegase a implementar una aplicación en el futuro.



**Figura 2.26: Circuito de conexión del módulo XBee con el microcontrolador de la placa PCI 1.**

## 2.3 ESTACIÓN REMOTA

### 2.3.1 ARQUITECTURA DEL SISTEMA

La estación remota comprende todos los elementos necesarios para recibir la información del guante sensorizado enviada por la estación local y utilizarla de tal manera que el manipulador robótico reciba las órdenes exactas sobre la posición en la que debe encontrarse y efectúe correctamente dichas órdenes, dependiendo de las acciones del teleoperador en la estación local.

El hardware de la estación remota está conformado por un Manipulador Robótico Lynxmotion AL5D, una tarjeta controladora de servos SSC16 Pololu y una tarjeta electrónica llamada PCI 2, la cual incorpora un microcontrolador PIC18F452 y un módulo XBee Serie 1.

La estación remota desempeña su función empezando por el módulo XBee que recibe los datos inalámbricamente, luego el microcontrolador se

encarga mediante algoritmos lógicos de procesar esta información para luego enviar los comandos de posición a la SSC y esta a su vez envíe las señales que posicionen correctamente al robot.

### 2.3.2 BRAZO ROBÓTICO LYNXMOTION AL5D

El modelo del brazo robótico utilizado para el proyecto es el AL5D de la marca Lynxmotion, el cual está constituido principalmente de piezas de aluminio y plástico para ganar flexibilidad y aligerar su peso, las mismas que se acoplan directamente a los servomotores para generar los movimientos de las articulaciones.

#### a. CARACTERÍSTICAS

En la Figura 2.27 se puede observar las partes constitutivas del hardware del robot: base, columna giratoria, brazo de oscilación, brazo, muñeca central y efector final (gripper o pinza).

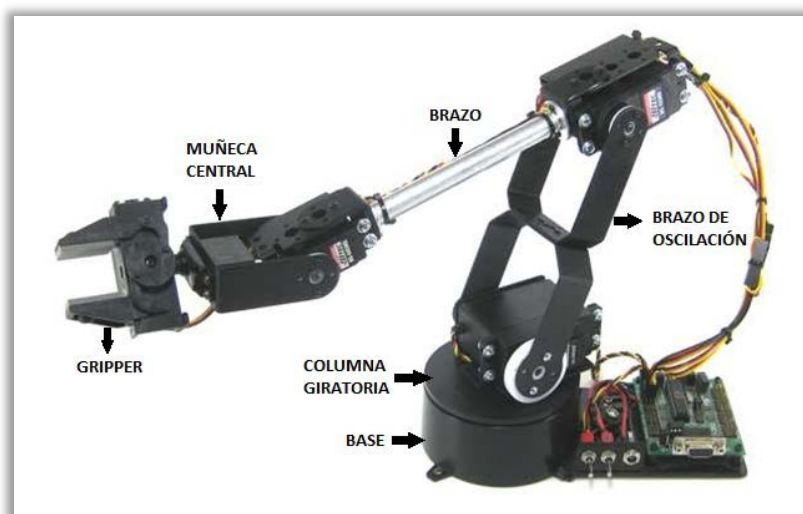


Figura 2.27: Partes del brazo robótico Lynxmotion.

El manipulador robótico posee cinco grados de libertad que son los movimientos de la base, hombro, codo, inclinación de la muñeca y rotación de la muñeca, además del efector final que refiere al movimiento de apertura y cierre del gripper.

Cada articulación del brazo robótico (Figura 2.28) utiliza un servomotor de la marca HITEC para ejecutar sus movimientos, excepto la articulación de la muñeca que necesita de dos servomotores para sus movimientos de inclinación y rotación respectivamente.



**Figura 2.28: Brazo robótico Lynxmotion AL5D con sus articulaciones.**

Los servomotores permiten el movimiento de cada eje en un rango de 180°, sin embargo por condiciones mecánicas de la estructura del robot, no todas las articulaciones pueden alcanzar la totalidad de ese rango de giro.

En la Tabla 2.5 se muestra la serie de los servomotores correspondiente a cada articulación.

**Tabla 2.5: Descripción de servomotores correspondientes a las articulaciones.**

<b>ARTICULACIÓN</b>	<b>SERVOMOTOR</b>
BASE	HS-485HB
HOMBRO	HS-805BB
CODO	HS-805MG
INCLINACIÓN DE LA MUÑECA	HS-645MG
ROTACIÓN DE LA MUÑECA	HS-485HB
GRIPPER	HS-311

El brazo robótico tiene un peso aproximado de 31 oz, equivalente a 850.49 gramos, alcanza una altura máxima de 529 mm y levanta una carga máxima con el brazo extendido de 4.18 oz (130 g) y una carga neta de 9.65 oz (300 g). Además la apertura del gripper es de 1.22”, equivalente a 31 mm.

A continuación se muestra la Tabla 2.6 con las dimensiones entre ejes de cada articulación del robot.

**Tabla 2.6: Distancia entre ejes del Brazo robótico AL5D**

<b>ARTICULACIONES</b>	<b>DISTANCIA (in)</b>	<b>DISTANCIA (mm)</b>
Base-Hombro	2.76	70
Hombro-Codo	5.69	144.5
Codo-Muñeca	7.34	186.5
Muñeca-Rotación de Muñeca	2.74	69.5
Rotación de Muñeca-Gripper	2.36	60
Muñeca-Extremo del Gripper	5.098	129.5

## b. CARACTERÍSTICAS TÉCNICAS DE LOS SERVOMOTORES

La Tabla 2.7 muestra un resumen de las características más importantes de operación de los servomotores utilizados en el brazo robótico.

Los servomotores encargados de los movimientos del manipulador son de la marca Hitec con un rango de operación de 4.8V a 6.0 V, se aprecia en las características que el valor de alimentación de voltaje es directamente proporcional al valor del torque e inversamente proporcional al valor de la velocidad, es en ese sentido que para obtener un mayor torque se establece un valor de alimentación de 6.0V para los servomotores.

**Tabla 2.7: Características técnicas de servomotores Hitec.**

SERIE	DIMENSIONES	PESO	DBW	VOLT	TORQUE	VELOCIDAD SEC/60°
HS-485HB	1.57x0.78x1.49 in 39.8x19.8x38.0 mm	1.59 oz 45.0 g	8usec	4.8V	66.65 oz-in 4.8 kg-cm	0.22
				6.0V	83.32 oz-in 6.0 kg-cm	0.18
HS-805MG	2.59x1.18x2.26 in 65.8x30.0x57.4 mm	6.95 oz 197.0 g	8usec	4.8V	275.0 oz-in 19.8 kg-cm	0.19
				6.0V	343.0 oz-in 24.7 kg-cm	0.14
HS-805BB	2.59x1.18x2.26 in 65.8x30.0x57.4 mm	5.36 oz 152.0 g	8usec	4.8V	275.0 oz-in 19.80 kg-cm	0.19
				6.0V	343.0 oz-in 24.70 kg-cm	0.14
HS-645MG	1.59x0.77x1.48 in 40.4x19.6x37.6 mm	1.95 oz 55.2 g	8usec	4.8V	106.9 oz-in 7.7 kg-cm	0.24
				6.0V	133.3 oz-in 9.6 kg-cm	0.20
HS-311	1.57x0.78x1.43 in 39.9x19.8x36.3 mm	1.52 oz 43.0 g	5usec	4.8V	42.0 oz-in 3.0 kg-cm	0.19
				6.0V	48.6 oz-in 3.5 kg-cm	0.15

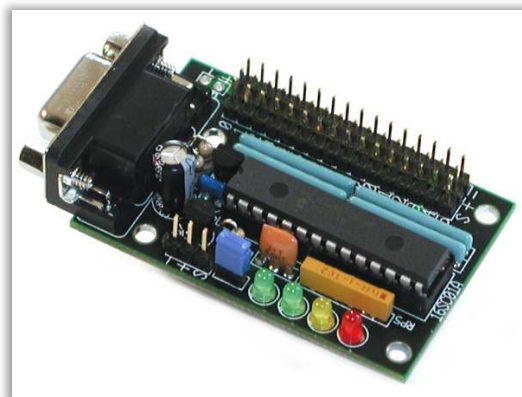
\*DBW: Dead Band Width (ancho de banda muerto).

### 2.3.3 TARJETA CONTROLADORA DE SERVOS SSC16 POLOLU

Anteriormente se pudo contemplar que el brazo robótico utilizado dispone de seis servomotores, a los cuales es posible controlar directamente mediante el microcontrolador de la estación remota, pero el problema radica que para controlar servomotores se requiere la creación de un algoritmo de control de alto nivel el cual utiliza técnicas avanzadas para realizar las tareas de sincronizar, almacenar posiciones, y mantener el torque de varios servomotores al mismo tiempo, sin tomar en cuenta que dicho microcontrolador ya se encuentra enfocado a otras tareas.

Tal proceso estaría más expuesto a errores y tendría dificultades innecesarias para el proyecto, el cual se basa principalmente en el tema de teleoperación, por tal motivo para el control de varios servomotores se tiende a usar tarjetas controladoras disponibles en el mercado las cuales son versátiles y fáciles de usar.

En el mercado local existen tarjetas controladoras de 8, 16, 32 o más servomotores, con interfaces de comunicación serial o USB. Debido a los requerimientos del proyecto se eligió la tarjeta controladora de servomotores SSC16 (Serial Servo Controller, max. 16 Servos) de marca Pololu, que puede apreciarse en la Figura 2.29.



**Figura 2.29: Controladora serial de 16 servos marca Pololu.**

### a. CARACTERÍSTICAS DE LA SSC16 POLOLU

La Controladora Serial Pololu de 16 Servos, permite controlar hasta dieciséis servomotores desde casi cualquier controlador de robot o computadora. La interfaz con la controladora de servos es un puerto serie estándar RS-232 o una línea serie de nivel TTL, utilizando cualquier velocidad de transmisión entre 1200 y 38400 baudios. En la siguiente tabla se detallan las especificaciones.

**Tabla 2.8: Especificaciones de la SSC16 Pololu**

Tamaño del PCB	1.50" x 2.275"
Número de puertos para servos	16
Rango de ancho de pulso	0.25 – 2.75 ms
Resolución	1 microsegundo (0.1 grado)
Voltaje de alimentación	5.6 – 20 V
Voltaje I/O	0 y 5 V
Velocidad de transmisión	1200 – 38400 (auto detección)
Consumo de corriente	15mA (promedio)

### OPCIONES DE INTERFAZ

Se puede comunicar con la controladora de servos usando uno de los dos protocolos de comunicación disponibles.

Uno de estos dos modos de interfaz es escogido cambiando el estado del jumper J1 mientras la servo controladora no se encuentre encendida; no es posible cambiar los modos sin resetear la servo controladora de modo que primero se apague y luego se encienda.

**Modo Pololu:** El modo por defecto, cuando jumper J1 está abierto, es el protocolo Pololu usado para controlar múltiples dispositivos seriales. En



este modo, la servocontroladora puede estar en la misma línea serial junto con otros dispositivos. Este modo también permite el acceso a todas las características especiales de la controladora de servomotores, como son ajustes de velocidades, rangos, y ajustes neutrales.

**Modo Mini SSC II:** Este modo es seleccionado colocando el jumper sobre los dos pines J1. Este ajuste permite que la servocontroladora responda al protocolo usado por la servocontroladora Mini SSC II fabricada por Scott Edwards Electronics. Este protocolo es más simple, pero solo permite al usuario especificar la posición deseada de los servos en una sola dirección. En este modo, la controladora de servos no es compatible con otros productos o periféricos seriales Pololu.

## **b. MODO DE FUNCIONAMIENTO POLOLU**

Se seleccionó este modo de operación para el control de los servomotores del brazo robótico, debido a la facilidad de uso y a la posibilidad de acceder a mayor cantidad de configuraciones para los servomotores. En este modo existen varias opciones para controlar los servos. Los distintos comandos lidian con el establecimiento de los valores de ancho de pulso para los servos.

Con los comandos absolutos, simplemente se establece el valor para cada servo. En modos de 7 y 8 bits, se establece el valor neutral, rango y dirección para cada servo; luego, cuando se envía un comando de posición de 7 u 8 bits, la controladora de servos combina todos los parámetros para obtener la posición actual del servo. Tanto si se utiliza un modo absoluto o no, es posible controlar la velocidad de cada servo.

**Velocidad de transmisión.** El rango de velocidad disponible en este modo es aproximadamente 2000 a 40000 baudios. Es posible para la controladora de servos operar a tasas de hasta 57600 baudios, pero el

éxito del intento dependerá de la velocidad exacta de la fuente de control en serie. En el caso del presente proyecto se utiliza 9600 baudios de velocidad de transmisión.

**Protocolo.** Para comunicarse con la controladora de servos, se envía una secuencia de cinco a seis bytes. El primer byte es un valor de sincronización que siempre debe ser 0x80 (128). El byte 2 es el número del tipo de dispositivo Pololu, que es 0x01 para la controladora de 16 servos. El byte 3 es uno de los seis valores para los diferentes comandos de la controladora. El byte 4 es el servo al cual se le pretende aplicar el comando. El byte 5 y posiblemente el 6 son el valor dado por el comando. En todos los bytes excepto en el byte de inicio, el bit siete debe ser 0. Por lo tanto, el rango de valores para los bytes 2 al 6 es de 0x00-0x7F (0 a 127 en decimal).

#### **Comando 0: Establecer parámetros (1 data byte)**

Bit 6 especifica tanto si un servo esta encendido o no; con 1 enciende el servo, y con 0 (por defecto) lo apaga.

Bit 5 establece la dirección del movimiento del servo, que solo aplica a los comandos de posición de 7 y 8 bits. Si el bit es 0 (por defecto), un número alto produce un pulso de salida alto; si el bit es 1, un número de posición alto hace que el pulso de salida sea corto.

Bits 0-4 son el rango a través del cual el servo se mueve en los comandos de 7 y 8 bits. Un valor elevado dará un rango elevado, y estableciendo el rango a 0 el servo estará siempre neutral. Para una misma configuración de rango, un comando de 8 bits moverá el servo a través de dos veces el intervalo de un comando de 7 bits.

El rango establecido por defecto es 15, que dará aproximadamente 180 grados en comandos de 8 bits y 90 grados en comandos de 7 bits.

#### **Comando 1: Establecer velocidad (1 data byte)**

Este comando permite establecer la velocidad con la que el servo se mueve. Si la velocidad se establece en 0 (por defecto), el pulso de salida cambiará instantáneamente a la posición establecida.

Si la velocidad es un valor diferente de 0, el pulso cambia gradualmente desde la posición anterior a la nueva posición. Si se establece una velocidad de 1, el ancho de pulso cambia a 50us por segundo, sube a una máxima velocidad de 6.35ms por segundo con una velocidad establecida de 127.

#### **Comando 2: Establecer Posición, 7 bits (1 data byte)**

Cuando este comando es enviado, el valor es multiplicado por el rango establecido para el correspondiente servo y ajustado para la configuración neutral.

#### **Comando 3: Establecer Posición, 8 bits (2 data byte)**

Este comando es como la versión de 7 bits, excepto que dos bytes de datos deben ser enviados para transferir 8 bits. Bit 0 del dato 1 contiene el bit más significativo (bit 7), y los bits menos significativos del dato 2 contienen los 7 bits menos significativos restantes.

#### **Comando 4: Establecer Posición, Absoluto (2 data byte)**

Este comando permite un control directo de la variable interna de posición del servo. Configuración neutral, rango y dirección no afectan este comando. El dato 2 contiene los 7 bits más bajos, y el dato 1 contiene los más altos. El rango de valores válidos va desde 500 a 5500. Configurar un servo a una posición automáticamente lo encenderá.

#### **Comando 5: Establecer el valor neutral (2 data bytes)**

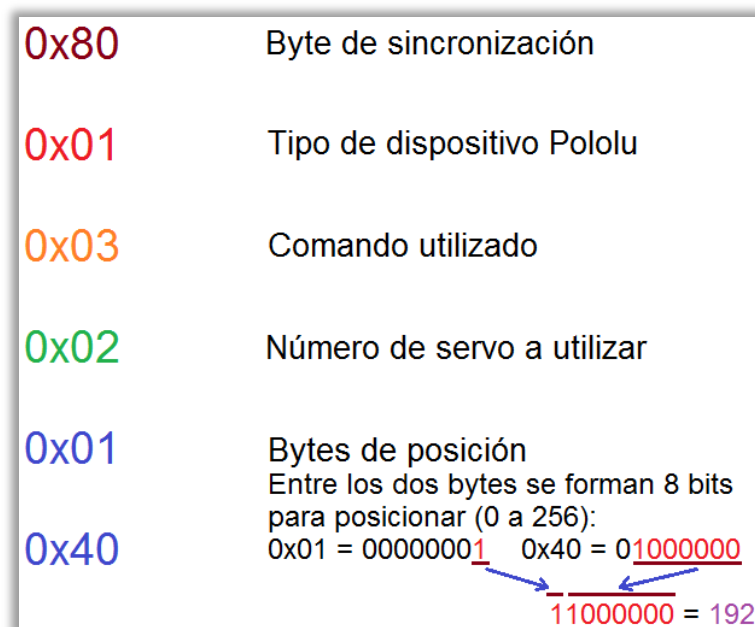
Establecer el valor neutral solo es aplicable a los comandos de 7 y 8 bits. El valor neutral establece la mitad del rango, y corresponde a un valor de 53.5 en una posición de 7 bits o a un valor de 127.5 en una posición de 8

bits. La posición neutral es una posición absoluta, y configurando la posición neutral el servo se moverá a esa posición. El valor por defecto es 3000.

### EMPLEO DEL MODO DE FUNCIONAMIENTO POLOLU

Para realizar el control de las posiciones que deben tomar los servomotores, el microcontrolador de la estación remota envía a la tarjeta controladora de servos una secuencia de seis bytes en los cuales se incluyen los dos bytes del “Comando 3” (Establecer Posición).

Al utilizar el comando 3 que usa 8 bits para establecer una posición, se tiene un total de 0 a 256 variaciones para un rango de 0 a 180 grados en el servomotor. Por ejemplo, si se deseara posicionar el segundo servomotor conectado a la controladora a 135 grados, se debe enviar el valor decimal de 192 o su equivalente en binario o hexadecimal. En la siguiente figura se indica los seis bytes necesarios para este ejemplo.



**Figura 2.30: Secuencia de bytes utilizada con la controladora de servos Pololu.**

Al no enviar ninguna secuencia anterior a la secuencia de posición, que establezca los valores para los comandos 0 (parámetros), 1 (velocidad) y 5 (valor neutral), automáticamente la tarjeta controladora al recibir el dato de posición utiliza los valores por defecto para los comandos mencionados.

#### 2.3.4 SENSOR FSR 402

La pinza del robot requiere un dispositivo que indique que un objeto fue atrapado y evitar así forzar el servomotor. Para cumplir con esta necesidad se utiliza un sensor FSR, como se ha explicado en el capítulo uno, este dispositivo es optimizado para aplicaciones de presión cualitativas, como es el caso.

El sensor utilizado es el modelo FSR 402 (de forma circular de 0.5" de diámetro) (ver Figura 2.31) de la empresa Interlink Electronics, el cual varía su valor de resistencia en una relación inversamente proporcional a la presión o fuerza aplicada a la sección circular.

Es un pequeño sensor de fuerza resistivo con una superficie de contacto redonda de 1.8 cm de diámetro fabricado de película de polímero.

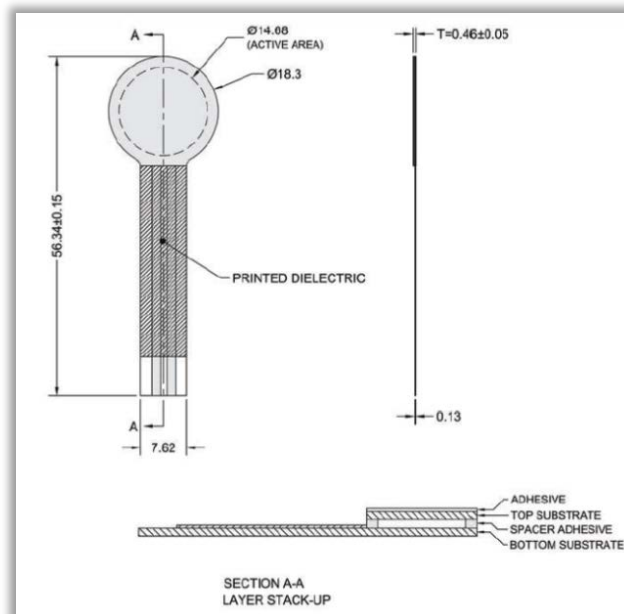


**Figura 2.31: Sensor FSR 402 de Interlink Electronics.**

### a. CARACTERÍSTICAS Y ESPECIFICACIONES

Mientras más se presione en la zona activa del sensor, menor será su valor de resistencia en respuesta a la fuerza aplicada. Cuando no es presionado, su resistencia es superior a  $1M\Omega$ . Su rango de presión varía de 0.2 N a 20 N (20.4 gf a 2.039 kgf).

El sensor es un dispositivo muy delgado y robusto con un diámetro de 18.29 mm y una longitud de 56.34mm, todas sus dimensiones pueden ser apreciadas en la figura 2.32.



**Figura 2.32: Dimensiones del FSR 402.**

Tiene una extensión que termina en dos puntas que se pueden soldar cómodamente. Es muy sencillo de utilizar y puede tener multitud de aplicaciones en muchos proyectos electrónicos que no requieran de gran precisión.

A continuación se detalla la tabla con las especificaciones técnicas del sensor.

**Tabla 2.9: Especificaciones técnicas del sensor FSR 402.**

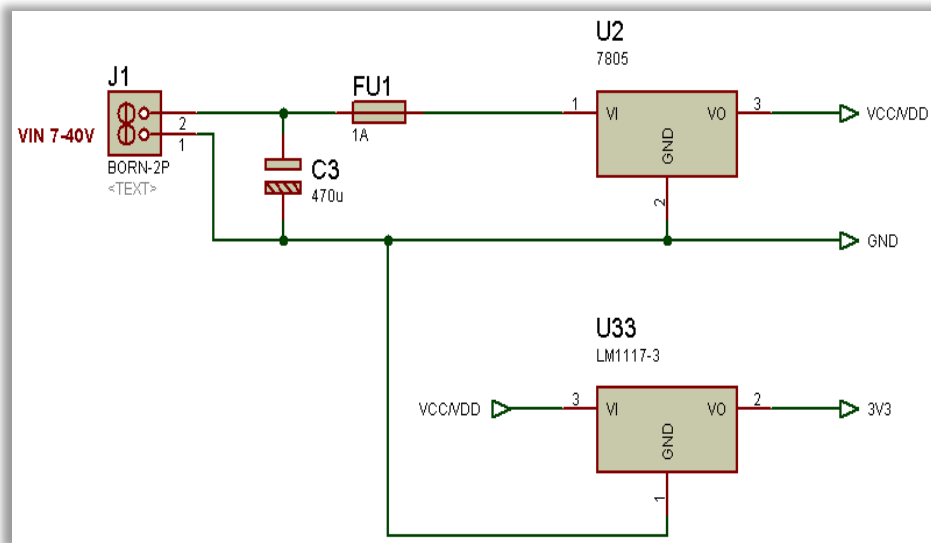
Rango de fuerza	0.2 N a 20 N (20.4 gf a 2.039 kgf)
Resistencia sin actuación	> 10 MΩ
Repetibilidad (mismo elemento)	± 2%
Repetibilidad (entre diferentes elementos)	± 6%
Diámetro total	18.29 mm
Diámetro área sensible	14.68 mm
Grosor	0.46 mm
Recorrido para activarse	0.15mm
Vida útil	10 000 000 de activaciones
Rango de temperatura	-30 ° C a +70 ° C
Máxima corriente	1mA/cm <sup>2</sup> de la fuerza aplicada

### **2.3.5 PLACA ELECTRÓNICA PCI 2**

La placa electrónica de la estación remota llamada PCI 2 consta de cuatro secciones principales que resumen los recursos utilizados para su funcionamiento, estas son acondicionamiento de señales, disposición de pines del microcontrolador, transmisión inalámbrica y conexión de la tarjeta controladora.

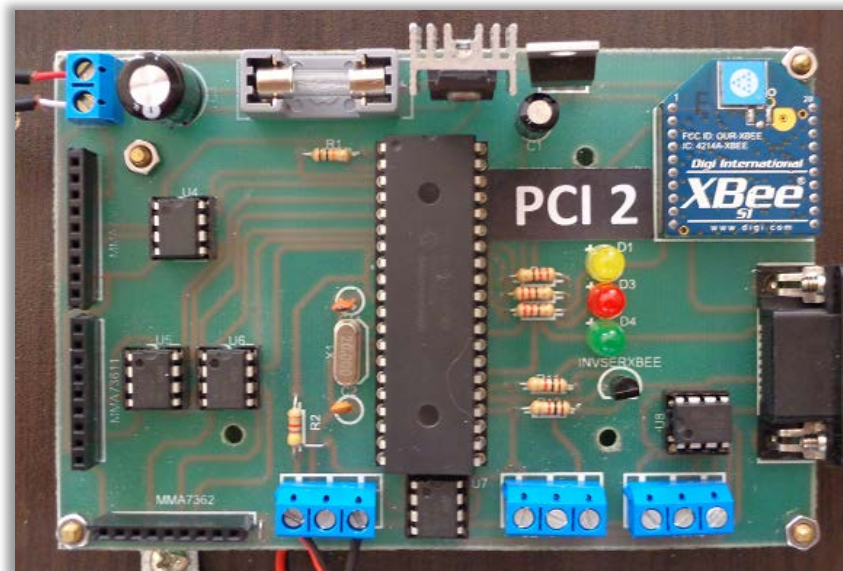
La PCI 2 es una tarjeta electrónica con diseño de un solo lado y de dimensiones de 125x80mm, realizada en Ares (Advanced Routing and Editing Software) módulo del sistema Proteus que permite la creación de circuitos impresos.

La energía necesaria para el funcionamiento de la Tarjeta electrónica PCI 2 es suministrada por una fuente de voltaje exclusiva. Se implementa a continuación de la fuente un circuito regulador de tensión para obtener los voltajes de alimentación constantes requeridos por los dispositivos de la tarjeta, los cuales son 5V y 3.3V. En la Figura 2.33 se muestra el circuito regulador de tensión.



**Figura 2.33: Circuito regulador de tensión para la PCI 2.**

En la Figura 2.34 se puede observar la tarjeta PCI 2 con sus elementos montados.



**Figura 2.34: Placa electrónica estación remota.**

A continuación se describe los componentes y conexionado de la tarjeta en cada una de sus secciones:

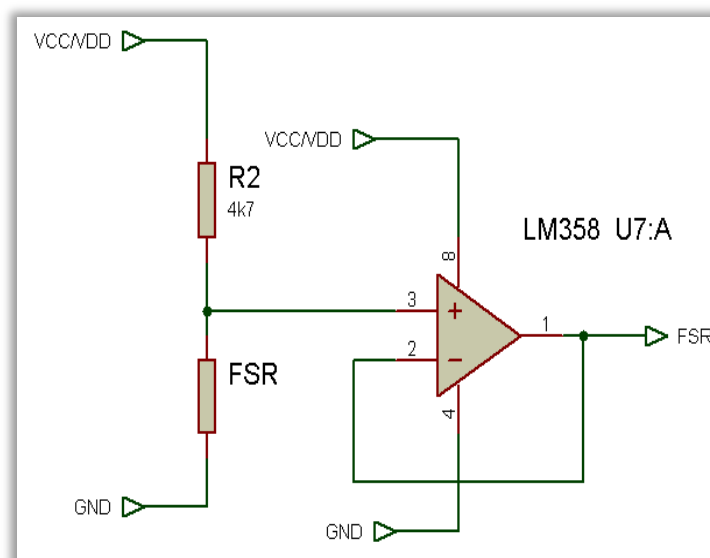


### a. ACONDICIONAMIENTO DE SEÑALES

La tarjeta electrónica PCI 2 está diseñada de tal manera que se pueda conectar hasta ocho señales análogas de sensores, de los cuales se utiliza un canal para el sensor de fuerza FSR (usado para precisar la fuerza de cierre del gripper) y los restantes siete son canales que pueden servir para ampliación en un futuro.

Cada uno de los canales análogos tiene implementado un amplificador operacional en configuración seguidor de tensión (véase la figura 2.24) para eliminar efectos de carga que pueden alterar los valores de medición de sensores.

Con relación a la obtención del valor de medición del sensor FSR se utiliza un circuito simple denominado divisor de voltaje, el cual transforma la respuesta resistiva del sensor a una señal de voltaje, la Figura 2.35 muestra el circuito de acondicionamiento (divisor de tensión y seguidor de tensión) implementado para el sensor de fuerza FSR.



**Figura 2.35: Circuito de acondicionamiento para el FSR.**

## **b. DISPOSICIÓN DE PINES DEL MICROCONTROLADOR**

El elemento principal encargado de leer las señales, de forma general, en la PCI 2 es el microcontrolador denominado de aquí en adelante PIC2, este microcontrolador es el PIC18F452, uno de la misma serie de la placa de la estación local.

Este microcontrolador es el encargado de realizar el algoritmo de control en base a los datos obtenidos a través de la comunicación inalámbrica y la señal del sensor FSR. La tabla 2.10 muestra los recursos utilizados de este microcontrolador en esta etapa del proyecto.

**Tabla 2.10: Descripción de I/O utilizadas en el microcontrolador de la placa PCI 2.**

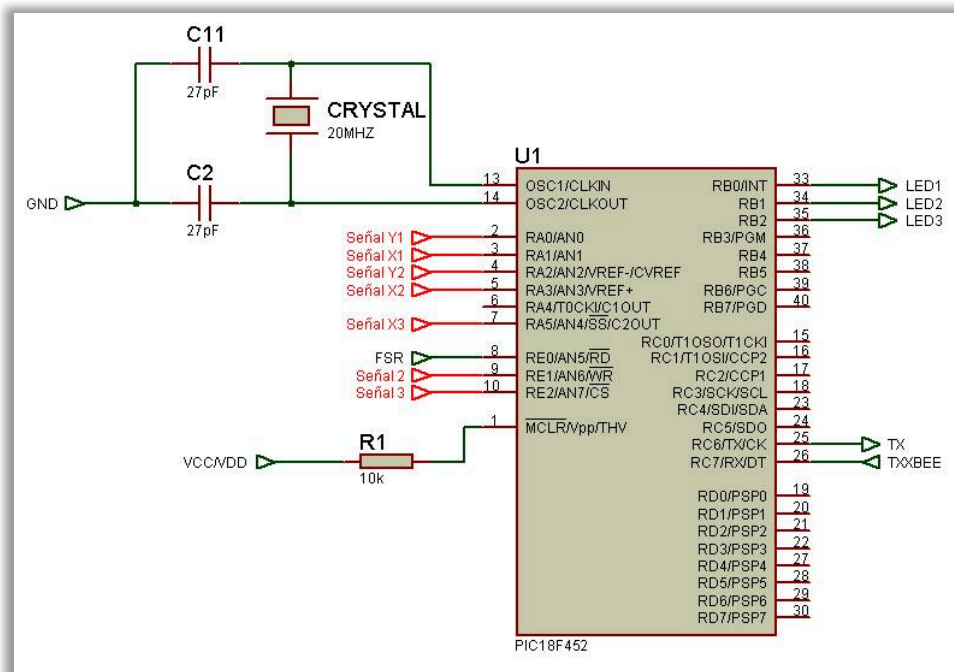
<b><i>Número de pin</i></b>	<b><i>Descripción del pin</i></b>	<b><i>Dispositivo Conectado</i></b>	<b><i>Función</i></b>
8	AN5	Sensor de Fuerza FSR	Señal fuerza de cierre del Gripper
33	RB0	LED 1(Amarillo)	Indicador de encendido
34	RB1	LED 2(Rojo)	Indicador de operación por guante sensorizado
35	RB2	LED 3(Verde)	Indicador operación vía HMI

Adicionalmente como se mencionó la PCI 2 tiene capacidad de agregar hasta siete señales análogas extras, ya que físicamente tiene montados conectores ligados con los canales del módulo convertidor Análogo Digital (A/D) del microcontrolador, de los cuales existen tres conectores para sensores que trabajen con una alimentación de 3.3V (sensores inerciales)

y dos conectores para sensores con alimentación de 5V. En la tabla 2.11 se resumen las conexiones de los canales libres de la PCI 2.

**Tabla 2.11: Descripción de canales libres de la placa PCI 2.**

<b>Número de pin</b>	<b>Canal (A/D)</b>	<b>Alimentación del Sensor</b>	<b>Señal</b>
2	Canal 0(AN0)	3.3V	Señal Y1
3	Canal 1(AN1)	3.3V	Señal X1
4	Canal 3(AN2)	3.3V	Señal Y2
5	Canal 4(AN3)	3.3V	Señal X2
7	Canal 5(AN4)	3.3V	Señal X3
9	Canal 6(AN6)	5V	Señal 2
10	Canal 7(AN7)	5V	Señal 3

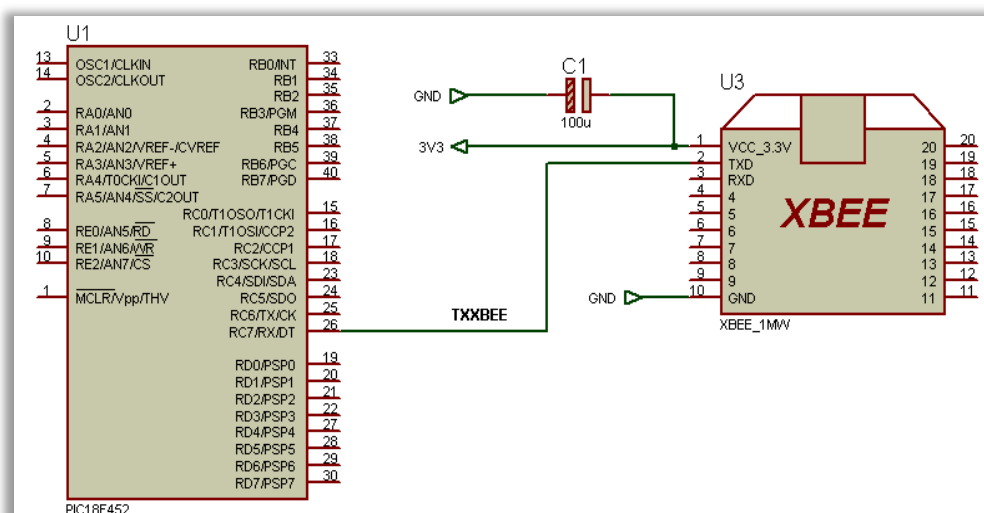


**Figura 2.36: Conexiones realizadas en el microcontrolador de la placa PCI 2.**

### c. CONEXIÓN DEL MÓDULO XBEE

El módulo XBee en la PCI 2 está encargado de recibir los datos enviados inalámbricamente desde el módulo XBee de la tarjeta electrónica PCI 1 y enviarlos al puerto de recepción RX (comunicación serial) del microcontrolador PIC2.

Cómo se especificó en la conexión del XBee con el microcontrolador PIC1, los niveles de voltaje a los que trabajan los dos dispositivos son distintos, por lo cual se utilizaba un circuito básico para cambiar el nivel de voltaje lógico de 5V a 3.3V para evitar que el módulo XBee sufriera daño permanente, pero en el caso de la PCI 2 se conecta directamente el terminal de transmisión TXD del XBee al puerto de recepción RX del microcontrolador (véase la Figura 2.37), ya que los niveles lógicos van de 3.3V a 5V y no existe riesgo de daños como en el caso anterior, además con respecto a la información, la misma no se ve afectada ya que los niveles TTL del microcontrolador consideran un valor lógico alto desde 2.4V en adelante.



**Figura 2.37: Circuito de conexión del módulo XBee con el microcontrolador de la placa PCI 2.**

#### d. CONEXIÓN CON LA TARJETA CONTROLADORA DE SERVOS

La tarjeta controladora de servos es la encargada de posicionar los servomotores del robot, según las exigencias del algoritmo realizado en el microcontrolador de la estación remota.

Para unir la etapa de control de la PCI 2 con la controladora de servos se utilizó el puerto serie estándar RS232 de la controladora COM3, el mismo que tiene incorporado un circuito adaptador de niveles RS232 a TTL (véase Anexo B), utilizado generalmente para conectar directamente dispositivos con niveles RS232 (como la PC) con la controladora de servos que maneja niveles TTL.

En aplicación tanto la etapa de control de la PCI 2 como la controladora de servos, se encuentran en niveles TTL, y por lo descrito anteriormente para la conexión entre las mismas se debe invalidar la función del adaptador de nivel (RS232-TTL) incorporado de fábrica en la controladora de servos. Por tal motivo se implementó un circuito inversor de nivel (véase Figura 2.38) similar al adaptador de nivel incorporado en la controladora.

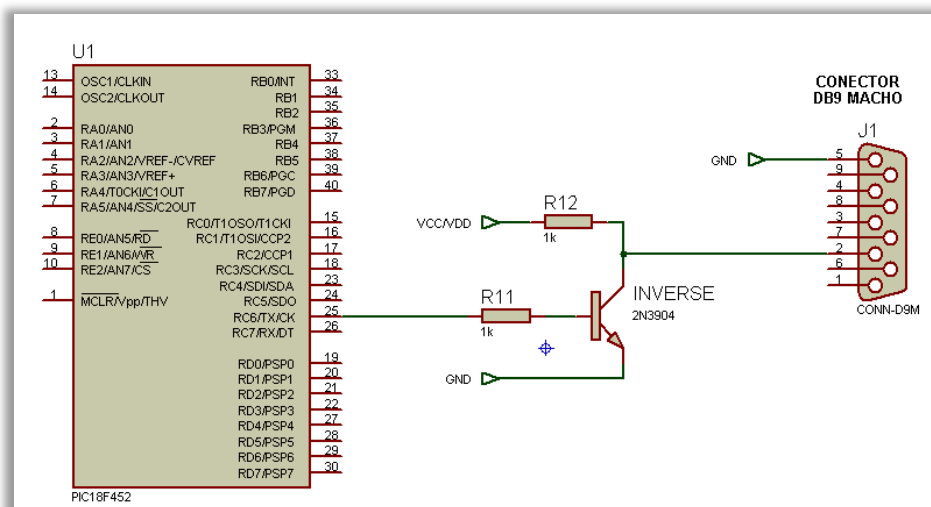
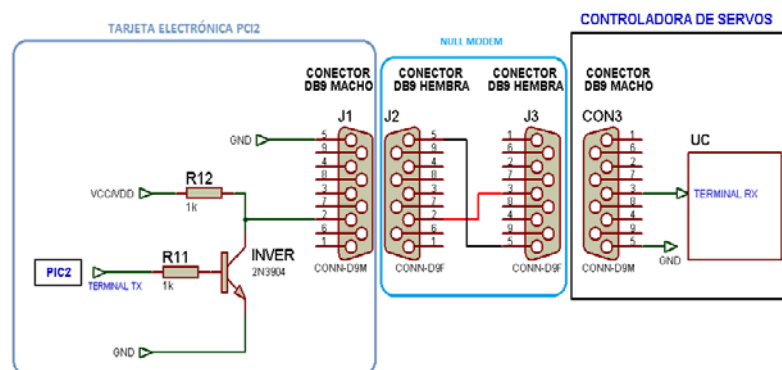


Figura 2.38: Diagrama del circuito inversor de nivel.

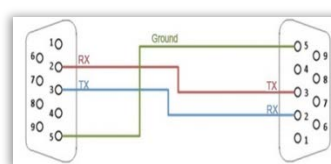
Todo lo desarrollado anteriormente muestra las conexiones físicas realizadas en la PCI 2, además de estas conexiones se implementa un cable denominado null modem (utilizado para aplicaciones de comunicación serie) para interconectar la PCI 2 con la tarjeta controladora de servos. La Figura 2.39 indica la conexión de la PCI 2 y la controladora de servos.

Cabe recalcar que la comunicación serial se realiza en un solo sentido desde la PCI 2 hacia la controladora de servos, así también la distancia del cable null modem está en el orden de unos pocos centímetros ya que la PCI 2 está montada muy cerca de la tarjeta controladora de servos.



**Figura 2.39: Conexión de la PCI 2 y la controladora de servos.**

El cable null modem tiene las líneas de transmisión y recepción entrelazadas, el cable empleado en la aplicación es el null modem serie de 3 hilos sin control de flujo, por lo cual utiliza solo los pines 2, 3 y 5. A continuación se muestra en la Figura 2.40 la conexión del cable null modem de 3 hilos sin control de flujo utilizado.



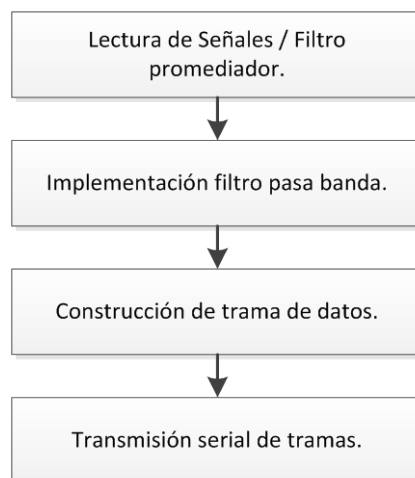
**Figura 2.40: Diagrama null modem 3 hilos sin control de flujo.**

## 2.4 ESTRUCTURA DE CONTROL

El algoritmo de control para el microcontrolador tanto de la estación local como de la estación remota fue realizado mediante el software de programación “MikroC Pro for PIC”, el cual se eligió debido a la facilidad de programación en lenguaje C y a las prestaciones generales de dicho software.

### 2.4.1 SOFTWARE DE CONTROL DE LA ESTACIÓN LOCAL

El desarrollo del algoritmo del microcontrolador de la placa PCI 1 se encuentra estructurado de la siguiente manera:



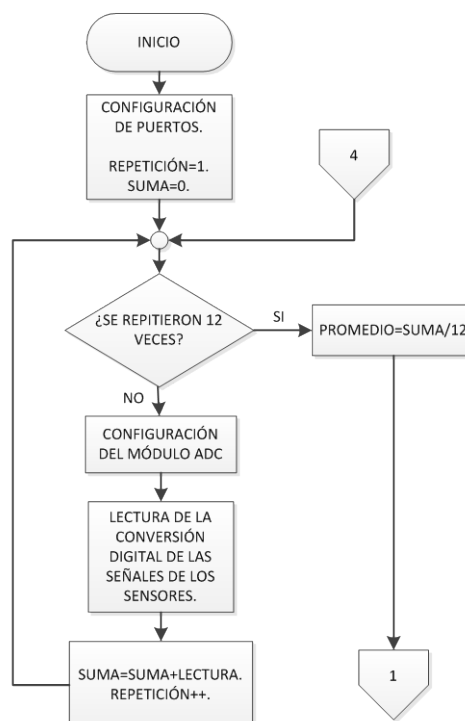
**Figura 2.41: Estructura del algoritmo del microcontrolador de la tarjeta PCI 1.**

A continuación se detalla el funcionamiento de cada una de estas secciones de código. Nótese que en conjunto, los diagramas de flujo de cada una de las secciones conforman el diagrama de flujo total del algoritmo del microcontrolador de la estación local, véase desde la Figura 2.42 a la Figura 2.47.

### a. LECTURA DE SEÑALES

Después de realizar la respectiva declaración de variables y configuración de puertos comunes, el programa principal del microcontrolador inicia con la lectura de las cuatro señales de los acelerómetros, la señal del sensor flex y la señal del eje Y del stick del nunchuk. Se utiliza el módulo ADC con configuración de 10 bits y con referencia Vdd y Vss, lo que hace que 5 voltios en una de las entradas de estos canales sean representados por un valor digital de 1023.

Para obtener un valor más preciso de las señales de los sensores, se aplica un filtro digital promediador justo después de obtener la lectura de las señales con el módulo ADC. El filtro promediador trabaja de tal manera que realiza un promedio de doce lecturas para cada una de las señales. La Figura 2.42 muestra el diagrama de flujo correspondiente.

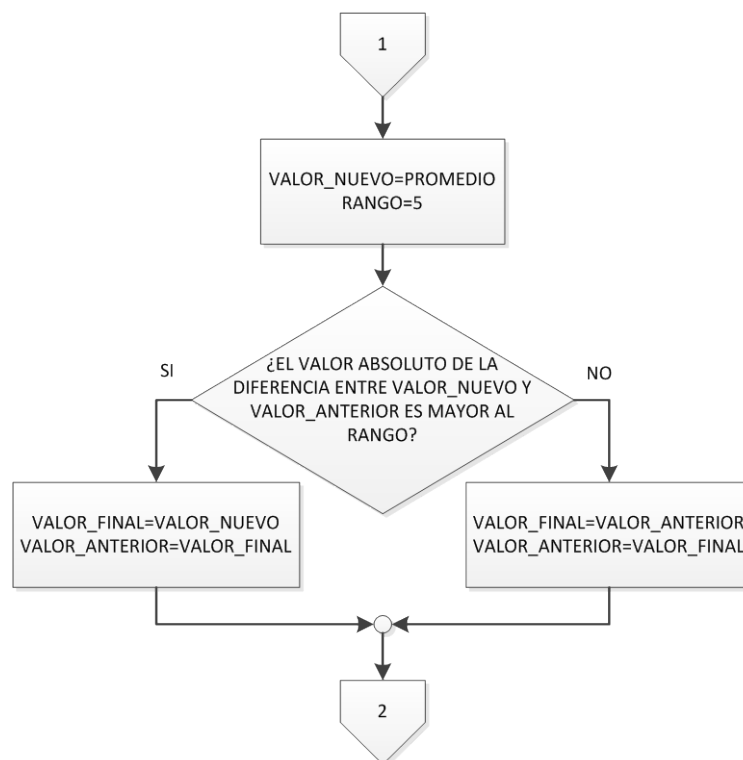


**Figura 2.42: Diagrama de flujo para la lectura de los sensores y filtro promediador.**



## b. FILTRO PASA BANDA

Debido a la sensibilidad de los sensores del guante, especialmente de los acelerómetros, al momento en que el operador desee mantener el brazo con el guante en una posición sin moverse (estática), los sensores emitirán cambios en su señal ya que es imposible para una persona evitar totalmente vibraciones o movimientos milimétricos involuntarios de sus articulaciones. Estos cambios de señal representarían cambios de posición o vibración para el robot si se usan directamente. Es por eso, que a continuación del filtro promediador se implementa al algoritmo un filtro digital pasa banda.



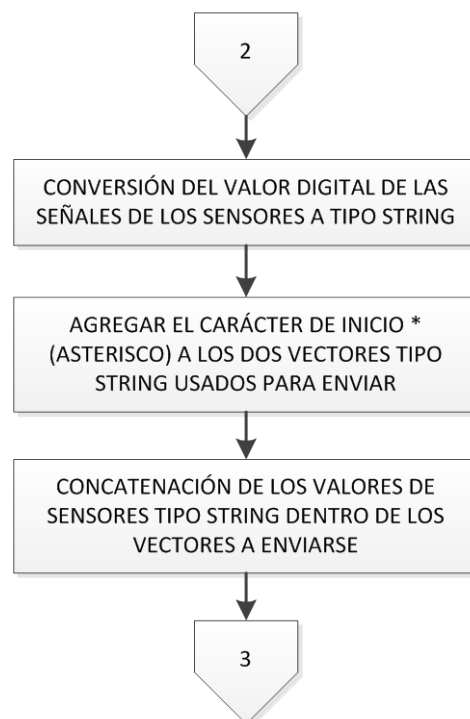
**Figura 2.43: Diagrama de flujo del funcionamiento del filtro pasa banda.**

El filtro pasa banda discrimina aquellos valores de los sensores que estén dentro de un rango específico, el valor numérico de dicho rango

representa la cantidad de variación de la señal de los sensores en estado estático. Al discriminar estos valores se elimina aquellos movimientos indeseados del robot mientras se mantiene una posición estática en el guante, lo que se traduce a: si no hay movimiento en el guante no habrá movimiento del robot. En la figura 2.43 se observa de manera general el diagrama de flujo del filtro.

### c. TRAMA DE DATOS

Para transmitir la información de los sensores de manera serial hacia el microcontrolador, se crean dos tramas de datos tipo string (cadena de texto), formadas por los valores de los sensores organizados de manera estructurada. En la Figura 2.44 se puede observar el diagrama de flujo que indica cómo se han desarrollado las tramas.



**Figura 2.44: Diagrama de flujo para la creación de las tramas de datos.**

Cada una de las tramas es almacenada en vectores tipo string, siendo el primer elemento del vector (posición cero) el símbolo \* (asterisco) el cual representa el inicio y el tipo de trama.

Para este caso en particular símbolo asterisco representa al tipo de control será por medio del guante sensorizado.

La primera trama está formada por los valores de las señales de los sensores que representan el movimiento de la mano, el giro de la muñeca y el movimiento del codo respectivamente como se puede apreciar en la Figura 2.45.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
*				3	5	8	@				3	3	1	%				3	3	0	#

**Figura 2.45: Estructura del vector tipo texto usado para enviar la primera trama de datos.**

En la Figura 2.46 se indica la segunda trama formada por las señales restantes que son el movimiento del hombro, el stick del nunchuk, y el movimiento del sensor flex.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
*				3	3	4	!			1	0	2	0	)				7	5	0	(

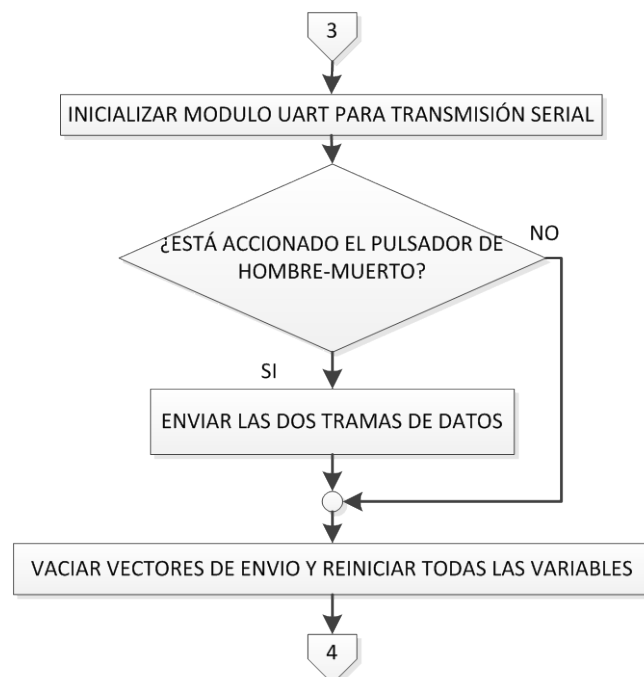
**Figura 2.46: Estructura del vector tipo texto usado para enviar la segunda trama de datos.**

Los símbolos posicionados a continuación de los valores numéricos son utilizados únicamente como analogía de “a qué señal” pertenecen los valores que los anteceden, además el símbolo “#” y el símbolo “(” sirven en la estación remota para indicar que es el final de las tramas respectivamente.

#### d. TRANSMISIÓN SERIAL DE TRAMAS

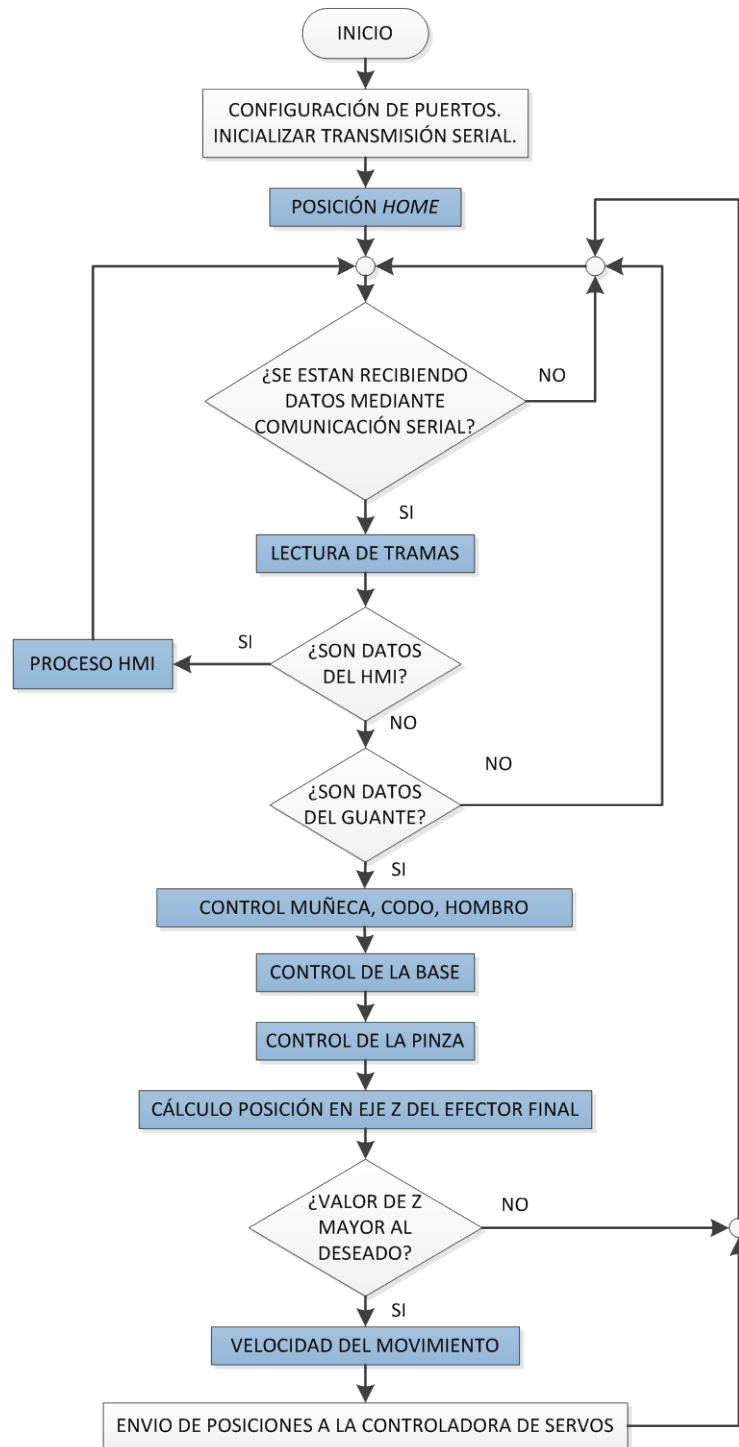
Para realizar la transmisión serial de datos al módulo XBee se inicializa el módulo UART del microcontrolador a una velocidad de transmisión de 9600 Baudios. Las dos tramas de datos que contienen la información de los sensores son únicamente enviadas si el botón Z del nunchuk se encuentra presionado. Este botón se utiliza como un pulsador de hombre-muerto, en caso de ocurrir algún inconveniente que pueda afectar la integridad del robot o de la operación de transporte del material, únicamente se deja de presionar el botón y el robot automáticamente deja de moverse. Esta medida se utiliza como seguridad en el control de robots industriales para evitar accidentes.

La Figura 2.47 muestra el diagrama de flujo que indica la transmisión serial de datos y el funcionamiento del pulsador de hombre-muerto. Una vez terminado el proceso, el código vuelve a ejecutarse en un lazo infinito iniciando nuevamente con la lectura de los sensores en la Figura 2.42.



**Figura 2.47: Transmisión inalámbrica de las tramas de datos.**

## 2.4.2 SOFTWARE DE CONTROL DE LA ESTACIÓN REMOTA



**Figura 2.48: Diagrama de flujo del algoritmo de control del microcontrolador de la estación remota.**

El desarrollo del algoritmo de control de la estación remota es mucho más complicado que el de la estación local, por tal motivo para su explicación se presenta el diagrama de flujo completo en la Figura 2.48, con la característica de que aquellos recuadros color azul serán explicados más detalladamente en los siguientes postulados.

#### **a. POSICIÓN HOME**

El robot cuenta con una posición HOME dada por los siguientes ángulos en sus articulaciones: base a  $0^\circ$ , hombro a  $90^\circ$ , codo a  $-90^\circ$ , giro de la muñeca a  $0^\circ$ , inclinación de la muñeca a  $0^\circ$  y apertura de la pinza a 16mm. Con estas posiciones de los servomotores la posición HOME del robot adopta la postura de la Figura 2.49.

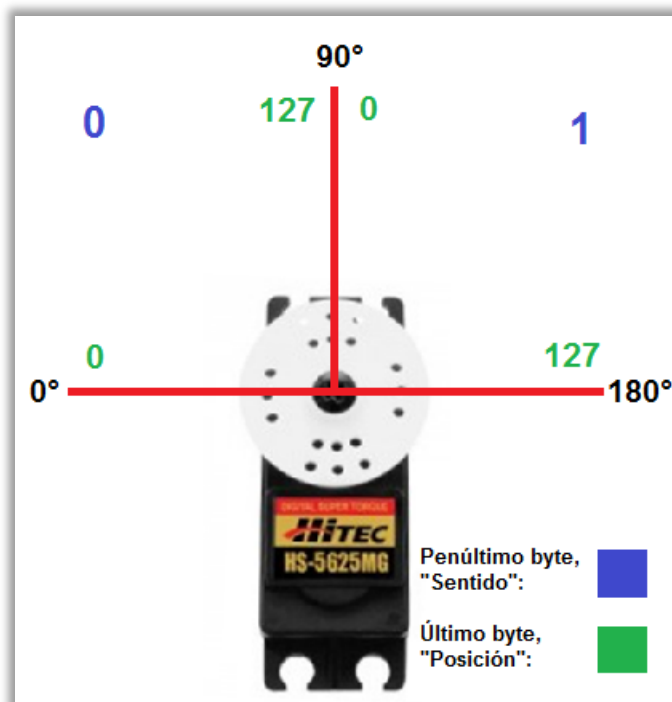


**Figura 2.49: Posición HOME del brazo robótico.**

Como se explica en el literal “b” del apartado 2.3.3 de este documento, para que los servomotores trabajen se debe enviar una secuencia de seis

bytes a la tarjeta controladora donde los dos últimos bytes forman la posición del servomotor.

Para mayor facilidad se toma al penúltimo byte como “dirección de movimiento del servomotor”, teniendo dos únicos valores decimales posibles, cero (0) o uno (1) los cuales representan movimiento del centro a la derecha o del centro a la izquierda. Al último byte se lo toma como “posición del movimiento”, el cual representa la posición que va tomar el servomotor una vez elegido el sentido, dicha posición tendrá un rango decimal de 0 a 127 según el caso. En la Figura 2.50 se puede apreciar los valores decimales que pueden tomar los dos bytes en un rango de movimiento de 180° del servomotor.



**Figura 2.50: Posibles valores de los bytes de posición para la controladora de servos.**

En el algoritmo de control, para mover a cualquier posición los servomotores del robot se utiliza una subrutina que envía de manera

serial hacia la controladora de servos los seis bytes necesarios y donde es posible modificar los tres últimos bytes correspondientes a número de servo, sentido y posición.

En la siguiente tabla se exponen los valores de los tres últimos bytes modificables para optar la posición HOME.

**Tabla 2.12: Valores de los bytes se envían a la controladora para la posición HOME.**

<i>Servomotor</i>	<i>Cuarto Byte: Número de servo</i>	<i>Quinto Byte: Sentido</i>	<i>Sexto Byte: Posición</i>
Base	5	1	0
Hombro	4	1	1
Codo	3	0	126
Inclin. muñeca	2	1	0
Giro muñeca	1	1	14
Pinza	0	1	11

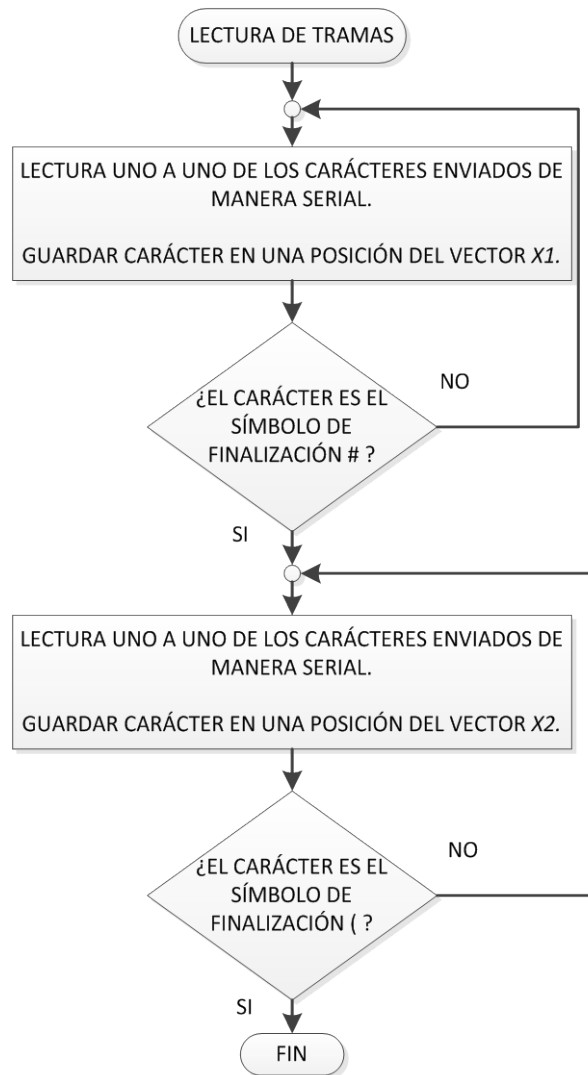
#### **b. LECTURA DE TRAMAS**

El XBee de la estación remota recibe dos tramas tipo texto desde la estación local, se pueden apreciar estas tramas en las Figuras 2.45 y 2.46, pero además para controlar el brazo robótico desde el HMI, la estación remota recibe información de posición para los servomotores mediante una trama enviada desde dicho HMI, esta trama se puede observar en la Figura 2.52.

Cualquiera que sea el origen de las tramas, éstas son guardadas en vectores mediante el mismo algoritmo ya que los símbolos de finalización de las tramas son los mismos para el HMI como para el guante



sensorizado, este proceso se puede observar en el diagrama de flujo de la Figura 2.51.

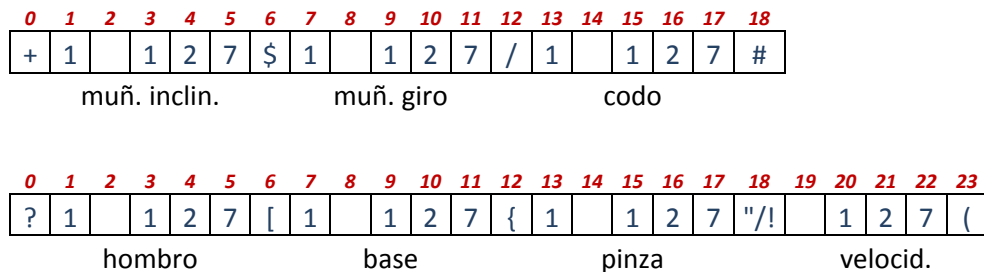


**Figura 2.51: Diagrama de flujo de lectura de tramas.**

### c. PROCESO HMI

Se sabe que la trama guardada en los vectores es del HMI cuando su primer símbolo es el “+” (más), esta trama posee información directa sobre el sentido y posición del movimiento de los servomotores, siendo “el

sentido” el valor decimal después del símbolo y los siguientes tres valores decimales “la posición” (véase Figura 2.52).



**Figura 2.52: Trama de datos enviada desde el HMI a la placa PCI 2.**

El algoritmo realiza dos procesos, el primero es determinar si la trama trae información sobre si se desea cambiar la velocidad, dependiendo de ello se realiza la lectura y se envían de manera serial los bytes a la controladora para cambiar la velocidad de los servomotores.

El segundo es la lectura en la trama de los parámetros de sentido y posición de los servomotores, para a continuación enviar los bytes de posicionamiento de cada servomotor a la controladora.

#### **d. CONTROL MUÑECA, CODO, HOMBRO**

Si se determinó que la trama guardada en los vectores fue enviada desde la placa PCI 1, quiere decir que el contenido de la trama son los valores de los sensores y dispositivos del guante. Esta sección del algoritmo utiliza la conversión de las señales del guante para determinar los valores necesarios de posición y sentido para así mover los servomotores mediante su tarjeta controladora.

Para el control de los servomotores de las articulaciones inclinación muñeca, giro muñeca, codo y hombro se utiliza un proceso idéntico por cada una. La parte principal del proceso consiste en realizar una

interpolación lineal para obtener el valor del byte de posición, conociendo de antemano los valores del acelerómetro correspondientes a los extremos de las posiciones de los servomotores en cada sentido. Los valores medidos de la señal de los ejes de los acelerómetros se listan en la siguiente tabla.

**Tabla 2.13: Valores digitales medidos de los acelerómetros usados en la interpolación.**

<i>Servomotor</i>	<i>Dirección</i>	<i>Ángulo</i>	<i>Sentido</i>	<i>Posición</i>	<i>Valor acelerómetro</i>
Rot. Muñeca	Arriba	90°	1	127	156
	Centro	0°		0	0
	Abajo	-90°	127		0
Inclinación Muñ.	Arriba	90°	1	127	190
	Centro	0°		0	0
	Abajo	-90°	127		0
Codo	Arriba	0°	0	0	170
	Centro	-90°		1	127
	Abajo	-180°	0		127
Hombro	Arriba	180°	1	127	161
	Centro	90°		0	0
	Abajo	0°	127		0

Por ejemplo (véase Figura 2.53), para el rango de movimiento de 0° a 90° en la inclinación de la muñeca del robot, el servomotor se encuentra en el sentido 1, con un valor de posición de 0 a 127 respectivamente. Por lo tanto el eje del acelerómetro que accione este servomotor debe realizar un movimiento dentro de ese rango angular, entregando un valor digital de su señal de 190 para 90° y 329 para 0°. Dando como resultado que el

valor del sensor en 190 corresponde a un valor de posición de 127 y el valor 329 para un valor de posición de 0, dentro del sentido 1 (véase Tabla 2.14). Si la posición del eje del acelerómetro de este caso se encuentra dentro de este rango entregando un valor de su señal de 200, para obtener el valor de posición correspondiente se utiliza interpolación lineal como se indica en la ecuación 2.1.

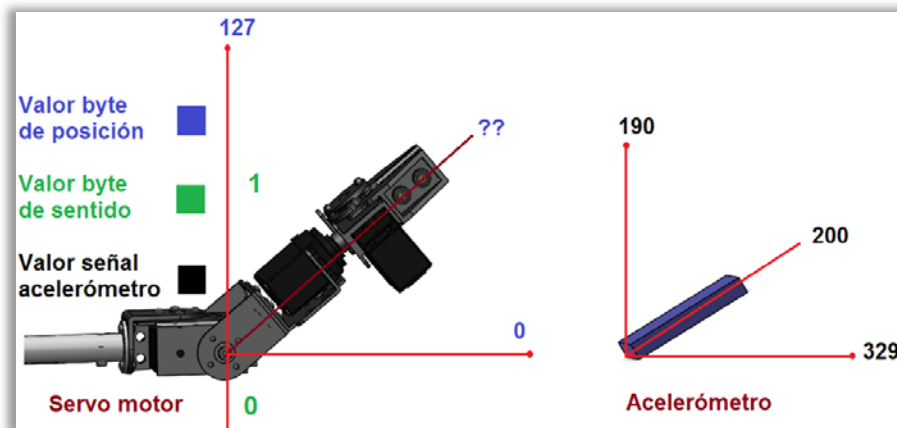


Figura 2.53: Analogía para la interpolación.

Tabla 2.14: Ejemplo de valores de interpolación.

<i>X</i>	<i>Y</i>
<i>Valor de sensor</i>	<i>Valor de posición</i>
190	127
200	V. pos.
329	0

$$y = y_1 + (x - x_1) \frac{(y_2 - y_1)}{(x_2 - x_1)} \quad \text{Ec. 2.1}$$

$$V. \text{ posición} = 127 + (200 - 190) \frac{(0 - 127)}{(329 - 190)} \quad \text{Ec. 2.2}$$

$$V. \text{ posición} = 117.8 \quad \text{Ec. 2.3}$$

Una vez interpolado el valor, si el valor de posición es un valor posible dentro del rango de movimiento mecánico del robot, este valor se almacena junto con el valor de sentido correspondiente.

### e. CONTROL BASE

El servomotor de la base es controlado por el eje Y del stick analógico del nunchuk, de tal manera que si el stick se mueve a la derecha la base del robot se moverá a la derecha, si el stick se mueve a la izquierda la base del robot se moverá hacia ese sentido y si el stick permanece centrado el robot no se moverá quedando en la última posición.

Este control se realiza incrementando o disminuyendo el valor del byte de posición dependiendo de la señal que entregue el stick del nunchuk. En la Figura 2.54 se presenta un diagrama de flujo que representa más claramente el algoritmo utilizado.

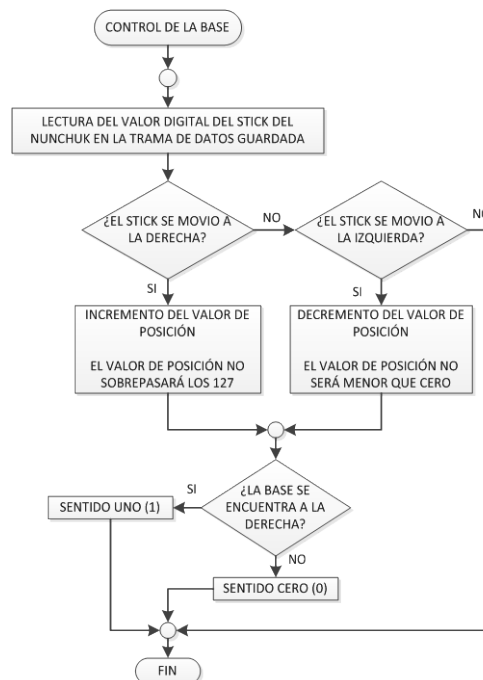


Figura 2.54: Diagrama de flujo para el control de la base del robot.

## f. CONTROL DEL GRIPPER O PINZA

Para el control de la pinza se utiliza el mismo proceso de interpolación del literal “d”, habiendo medido previamente los valores entregados por el guante cuando está cerrado completamente, abierto completamente y en un punto intermedio.

Como la pinza tiene la finalidad de coger objetos, para el control de la misma es necesario agregar un sensor de fuerza que indique que el objeto manipulado ha sido presionado con una fuerza mínima de sujeción y así evitar forzar al servomotor.

El diagrama de flujo de la Figura 2.55 indica el proceso que realiza el algoritmo en el control del gripper.

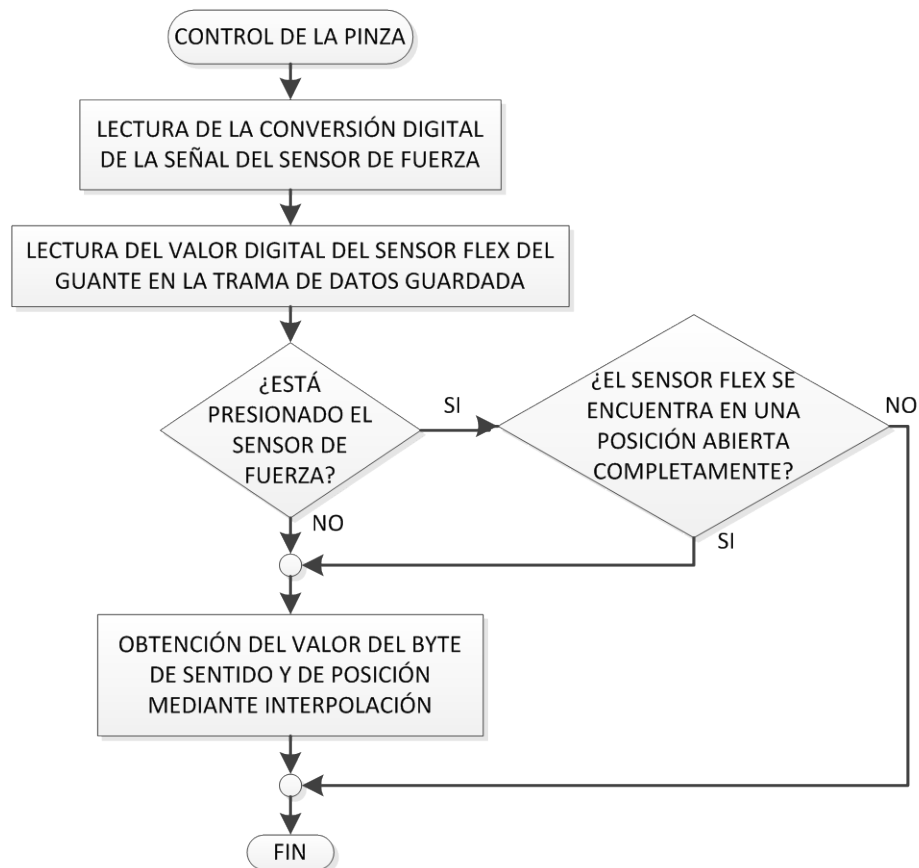


Figura 2.55: Diagrama de flujo del control del gripper del robot.

## **g. CÁLCULO DE LA POSICIÓN DE LA PINZA EN EL EJE Z**

Para evitar colisiones accidentales del robot con el suelo, se toma como medida de seguridad el cálculo de la posición en la que se encuentra el extremo del efector final en el eje Z, esto se realiza mediante una condicionante en el algoritmo de control evitar que el robot pueda moverse a menos de un valor específico dentro del eje Z.

Mediante el algoritmo de Denavit-Hartenberg se calculan los parámetros D-H del robot para posteriormente con ayuda del software MatLab resolver el problema cinemático directo y obtener la ecuación que represente a la posición del efector final en el eje Z en función a la posición angular de sus articulaciones.

La teoría sobre la resolución del algoritmo de Denavit-Hartenberg se encuentra más detallada en el apartado 1.2.3, los parámetros obtenidos para el robot en particular se indican en el apartado 2.6.2 de este documento. La ecuación utilizada se muestra a continuación:

$$Z = l1 + l2 \text{sen}(\theta_2) + l3 \text{sen}(\theta_2 + \theta_3) + l4 \text{sen}(\theta_2 + \theta_3 + \theta_4) \quad \text{Ec. 2.4}$$

Siendo L1, L2, L3 y L4 la distancia de los eslabones del robot (véase Figura 2.69), y  $\theta_2$ ,  $\theta_3$ ,  $\theta_4$  los valores angulares en grados de las articulaciones del robot como se indican en la Figura 2.70.

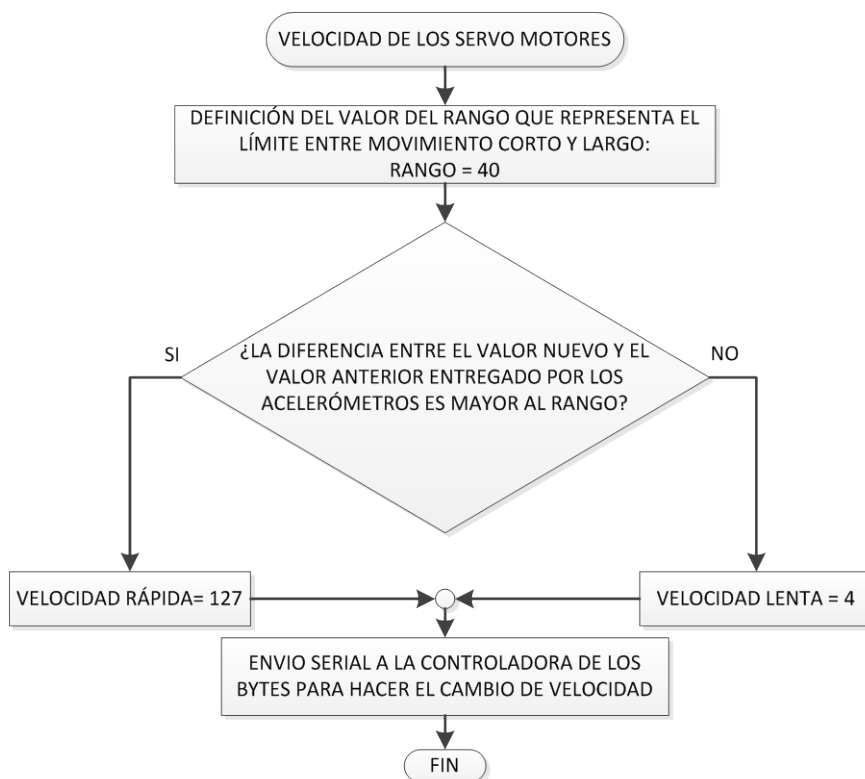
Debido a que las posiciones de los servomotores obtenidas mediante el algoritmo de control están expresadas por el valor del byte de sentido y el valor del byte de posición, se realiza físicamente la medición del correspondiente ángulo en grados para cada valor del byte de sentido y posición de los servomotores (véase Anexo A).

Dichas mediciones son guardadas en el software del microcontrolador y utilizadas para la resolución de la ecuación en el algoritmo de control.

## h. VELOCIDAD DEL MOVIMIENTO DE LOS SERVOMOTORES

Si el movimiento del guante sensorizado se traduce hacia el robot manipulador como una posición segura (mayor al valor deseado de Z), antes de enviar todos los valores de posicionamiento hacia la controladora de servos, primeramente se calcula la velocidad a la cual se posicionaran los servomotores.

Para el movimiento de los servomotores mediante el guante sensorizado existen dos posibles velocidades que se eligen automáticamente por el algoritmo de control dependiendo del rango de movimiento del guante. Si se leen movimientos largos la velocidad será rápida y si se leen movimientos cortos la velocidad será lenta. En el diagrama de flujo de la Figura 2.56 se puede observar este proceso.



**Figura 2.56: Diagrama de flujo para la selección de la velocidad de los servomotores.**

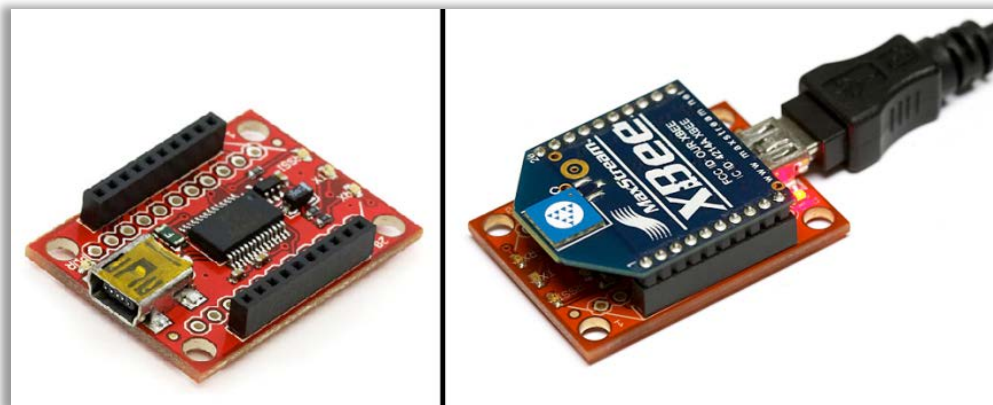


## 2.5 TRANSMISIÓN INALÁMBRICA DE DATOS

Existen tres módulos XBee utilizados en todo el sistema: el de la tarjeta electrónica de la estación local, el de la tarjeta electrónica de la estación remota y el del computador con el HMI.

El XBee de la estación local recibe información de manera serial desde el microcontrolador PIC1 y la envía inalámbricamente mediante una trama RF (radio frecuencia) hacia el módulo XBee ubicado en la estación remota.

De igual manera el módulo XBee del computador puede enviar inalámbricamente hacia la estación remota la información recibida desde el HMI. Para comunicar este módulo XBee con el computador se utiliza la placa XBEE EXPLORER la cual se conecta al computador mediante un puerto USB (ver Figura 2.57). Esta placa especial simula una conexión serial del módulo XBee con el computador.



**Figura 2.57: Placa XBee Explorer**

El XBee de la estación remota recibe la trama RF de cualquiera de las dos fuentes y envía la información de manera serial hacia el microcontrolador PIC2.

### **2.5.1 MODO DE FUNCIONAMIENTO DE LOS MÓDULOS XBEE**

Debido a los sencillos requerimientos de comunicación entre las estaciones, todos los módulos XBee utilizados trabajan en modo Transparente o también llamado modo transmitir/recibir, mediante una topología de comunicación punto a punto.

En este modo, todo lo que ingresa por el pin 3 (Data in), es guardado en el buffer de entrada y luego transmitido inalámbricamente y todo lo que ingresa como paquete RF, es guardado en el buffer de salida y luego enviado por el pin 2 (Data out). El modo Transparente viene establecido por defecto en los módulos XBee.

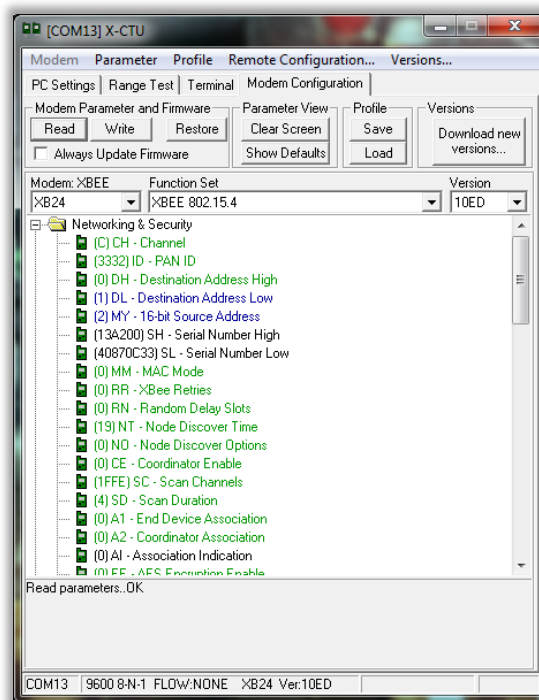
Este modo es el más sencillo de usar está destinado principalmente a la comunicación punto a punto, donde no es necesario ningún tipo de control. También se usa para reemplazar alguna conexión serial por cable, ya que es la configuración más sencilla posible y no requiere una mayor configuración.

### **2.5.2 DIRECCIONAMIENTO DE LOS MÓDULOS XBEE**

Los módulos permiten dos tipos de direccionamiento, el de 16 bit y el de 64 bits. La principal diferencia es que en el de 64 bit, es posible obtener una mayor cantidad de direcciones. Son a través de estas direcciones que los módulos se comunican entre sí.

Debido a que se utilizan únicamente tres módulos XBee se configuran direcciones de 16 bits en cada uno de los módulos. La configuración puede ser realizada mediante comandos AT enviados por el microcontrolador o a través del software X-CTU haciendo uso de la tarjeta XBee Explorer.

El software X-CTU presenta un entorno de configuración completo y de fácil utilización, es por esto que se utilizó este método para configurar las direcciones de los módulos XBee. Una vez ejecutado el software y conectado el XBee al computador es posible mediante la pestaña “Modem Configuration” visualizar, editar y guardar parámetros de configuración del módulo. La Figura 2.58 presenta la pestaña de configuración en el X-CTU.



**Figura 2.58: Interfaz de configuración de módulos XBee en el X-CTU.**

El parámetro MY, define un número de 16 bits como dirección del módulo dentro de la red. Su posible rango se encuentra entre 0x0 y 0xFFFFE. Para definirlo se ingresa el número en formato decimal. Por ejemplo si a un módulo se le desea asignar la dirección 0x3BF1, entonces se debe ingresar 15345 que es su equivalente en decimal.

Para el direccionamiento de 16 bits el parámetro DH debe ser 0x0. Mientras que parámetro DL, permite definir un número de 16 bits como

dirección del módulo de destino dentro de la red al cual se va a realizar la comunicación. Su rango debe estar entre 0x0 y 0xFFFFE.

No se permite usar las direcciones 0xFFFF ni 0xFFFFE para el direccionamiento de 16 bits ya que estas son utilizadas en el direccionamiento de 64 bits. Resulta más práctico tratar con direcciones decimales y no tomar en cuenta sus equivalentes en hexadecimal, es así que el rango posible de direcciones es de 1 a 65534.

La Tabla 2.15 indica los valores de las direcciones configuradas para cada módulo XBee.

**Tabla 2.15: Parámetros de configuración para las direcciones de los módulos XBee.**

	<b>Parámetro MY</b>	<b>Parámetro DL</b>
<b>XBee estación local</b>	1603	2501
<b>XBee HMI</b>	1625	2501
<b>XBee estación remota</b>	2501	1625

## **2.6 INTERFAZ HUMANO MÁQUINA**

Además de la implementación del guante sensorizado como medio principal de teleoperación del manipulador robótico, se ha desarrollado una interfaz amigable que permite interactuar a distancia al operador humano con el robot, haciendo uso de un computador.

Para realizar este tipo de interfaz se ha decidido utilizar el software LabVIEW 12.0 (2012) desarrollado por National Instruments, por la facilidad que este presenta en el desarrollo de aplicaciones y la gran potencialidad de combinarse con todo tipo de software y hardware, tanto del propio fabricante como de otros fabricantes.

La aplicación desarrollada está centrada en el usuario de una manera intuitiva y sencilla, la misma cuenta con controles e indicadores que permiten al operador inmiscuirse rápidamente en el proceso mental de la interfaz, sin crear confusión.

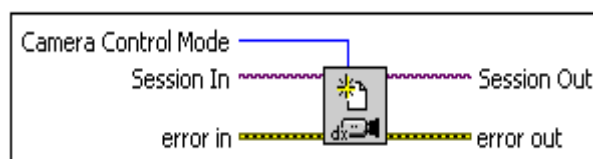
Cabe mencionar los dos pilares principales del software LabVIEW utilizados para desarrollar el HMI, estos son el Módulo de Robótica (Robotics Module) y el NI-VISA que es un soporte para dispositivos y comunicaciones.

Antes de describir las secciones del HMI, se realiza una revisión general de las herramientas del software LabView utilizadas en la aplicación.

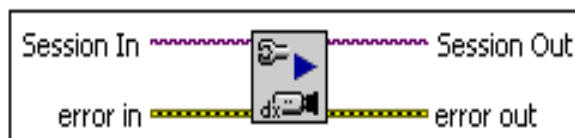
### 2.6.1 HERRAMIENTAS USADAS EN LA APLICACIÓN

Entre las herramientas utilizadas para el diseño de la aplicación de control del manipulador a distancia por medio del computador están:

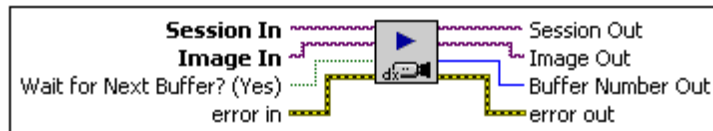
**IMAQdx Open Camera:** Abre una cámara, consulta sus capacidades, carga un archivo de configuración de la cámara, y crea una referencia única.



**IMAQdx Configure Grab:** Configura e inicia una adquisición de grabación que se repite en un lazo continuo. Adquiere imágenes a alta velocidad.



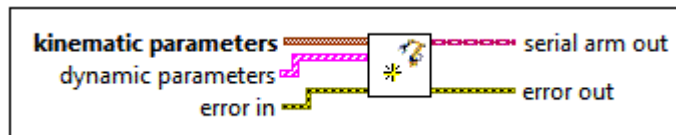
**IMAQdx Grab:** Adquiere los cuadros más recientes de la imagen de salida. Si el tipo de imagen no corresponde con el formato de la cámara de video, este VI cambia el tipo de imagen a un formato adecuado.



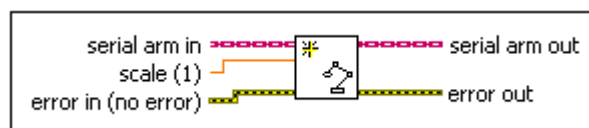
**IMAQdx Close Camera:** Detiene la adquisición y se cierra la sesión de la cámara especificada.



**Initialize Serial Arm:** Establece las propiedades de un brazo robótico y genera una representación del mismo. Existen algunos modelos ya planteados, y si se desea se puede generar un brazo genérico en el cual se debe seleccionar manualmente las instancias de sus parámetros.



**Initialize Plot:** Inicializa un objeto 3D que representa un brazo robótico.



**Update Plot:** Devuelve una escena 3D con el objeto 3D que representa un brazo robótico. Debe utilizar Initialize Plot VI para inicializar el objeto 3D antes de llamar a este VI por primera vez.

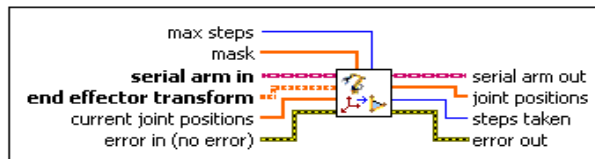
Puede utilizar este VI en un bucle con registros de desplazamiento para actualizar las posiciones comunes del brazo robótico en el objeto 3D y mostrar el movimiento del brazo entre posiciones.



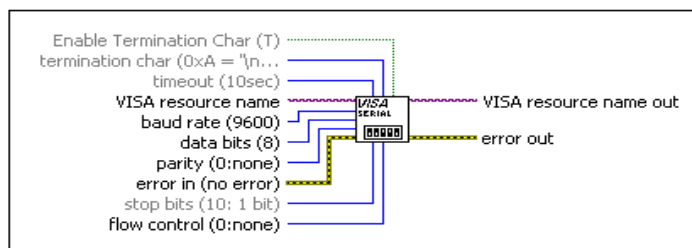
**Forward Kinematics:** Devuelve una transformación homogénea que representa la posición del extremo del efector del brazo robótico con respecto a la base del brazo.



**Inverse Kinematics:** Devuelve los ángulos de las articulaciones de un brazo robótico, dada la ubicación del efector final con respecto a la base del brazo.



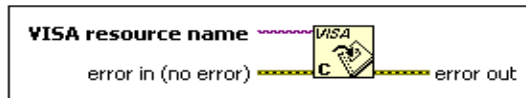
**VISA Configure Serial Port:** Inicializa el puerto serie especificado por el VISA resource name con las configuraciones elegidas.



**VISA Write Function:** Escribe los datos de write buffer en el dispositivo o interfaz especificado en VISA resource name.



**VISA Close Function:** Cierra la sesión del dispositivo especificado por VISA resource name.



## 2.6.2 DESCRIPCIÓN DEL HMI

La interfaz básicamente está desarrollada en forma de un menú interactivo, está dividida en tres secciones: un VI principal y dos SUBVIs que son llamados o invocados desde el VI principal. A continuación se describe cada VI:



### Main 4gdl.vi

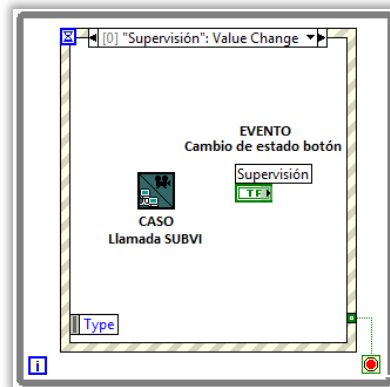
Este VI es el nodo principal de la aplicación, el mismo que permite invocar de manera instantánea a los SUBVIs y finalizar la aplicación, mediante los botones del panel frontal del VI, veáse Figura 2.59.



Figura 2.59: Panel Frontal del VI principal.



Para realizar el menú interactivo, se utilizó “Event Structure”, ya que esta estructura espera a que suceda un evento (en nuestro caso acción de los botones), a continuación ejecuta el caso apropiado para manejar el evento (llamada de SUBVIs o Stop) véase la Figura 2.60.

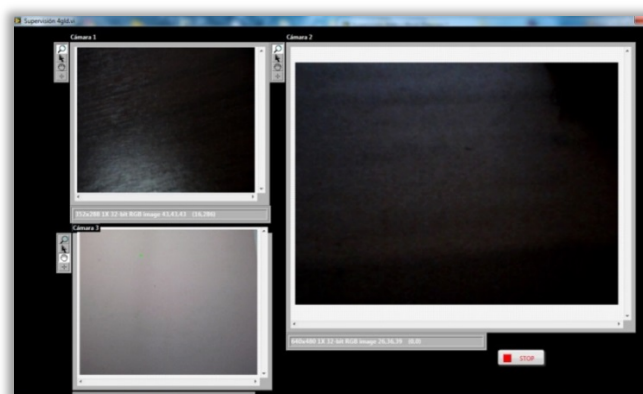


**Figura 2.60: Event Structure**



### **Supervisión 4gdl.vi**

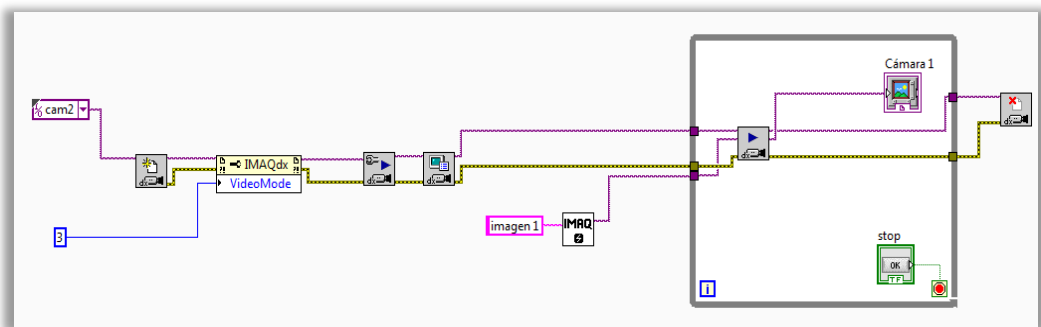
Este SUBVI tiene en su panel frontal dispuestos tres visualizadores de imágenes, que muestran las imágenes de video captadas por las webcams conectadas al computador vía USB, véase Figura 2.61.



**Figura 2.61: Panel frontal Supervisión 4gdl.vi**

Como su nombre lo indica solo sirve para supervisar la interacción del brazo robótico dentro de su entorno por medio de las tres cámaras ubicadas estratégicamente.

Para obtener las imágenes de las cámaras se utiliza un algoritmo básico desarrollado con las herramientas del IMAQdx, en la Figura 2.62 se puede observar el diagrama de bloques del algoritmo.



**Figura 2.62: Diagrama de bloques algoritmo de visualización de cámaras.**



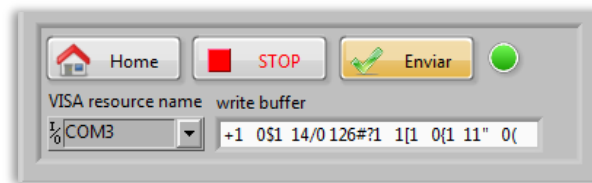
### **Control Robot AL5D (CD y CI) 4gdl.vi**

En sí este es el VI más importante de la aplicación, ya que dentro del mismo está implementada una serie de algoritmos que permiten el control del brazo robótico por medio del computador.

Como parte inicial, el software de control de posición configura parámetros esenciales de la comunicación con el manipulador, parámetros del sistema mecánico como velocidades de los servomotores y referencias para ubicar al robot en una posición inicial.

Es de ese modo que se tiene dispuestos los siguientes recursos mostrados en la Figura 2.63.

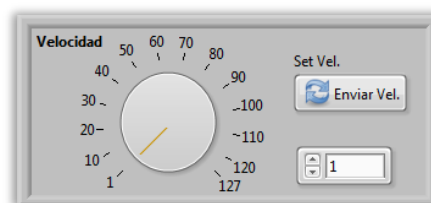
Tres botones de izquierda a derecha: HOME que sirve para posicionar al manipulador en una posición inicial, STOP que termina la aplicación y Enviar que debe permanecer activado para permitir el envío de la trama de datos hacia el robot. Además de lo indicado un control que permite seleccionar el puerto destinado para la comunicación y un visualizador de la trama a enviar.



**Figura 2.63: Recursos generales del algoritmo.**

Con respecto a la velocidad de operación de los actuadores del robot se creó dos controles virtuales, una perilla que permite establecer el valor de velocidad y un botón para confirmar el envío del valor de la velocidad al robot, véase la Figura 2.64.

La velocidad de los servomotores se configura mediante el envío de un valor entre 1 y 127 al controlador de servomotores. El valor de '1' implica que el servomotor se mueve a su mínima velocidad; mientras que el valor de '127' representa la velocidad máxima.



**Figura 2.64: Controles virtuales del parámetro velocidad.**

Con el desarrollo de la aplicación se permite al usuario controlar al manipulador robótico mediante dos métodos de operación que pueden ser combinados para facilitar la manipulación del robot en su entorno.

Los métodos de operación del manipulador son:

**Ejes Específicos (°).**- Este método permite desplazar en forma independiente cada uno de los ejes o articulaciones del robot, en dirección horaria o anti horaria.

Para variar la configuración angular del robot y la apertura-cierre de la pinza se utiliza controles slider, los mismos que están limitados por los valores angulares de los extremos.

Tanto para los dos métodos de control del manipulador se implementó un modelo virtual que permite tener noción de las posiciones que adoptará el robot real. Dicho modelo virtual utiliza un visualizador que representa un brazo robótico interpretando la escena 3D provista por Update Plot VI.

En este modo de operación (ejes específicos) se hace uso de la herramienta Forward Kinematics VI, el mismo que permite conocer la posición en la que se encuentra el gripper tras cada variación angular que se realice, todo lo anterior descrito se puede observar en la Figura 2.65.

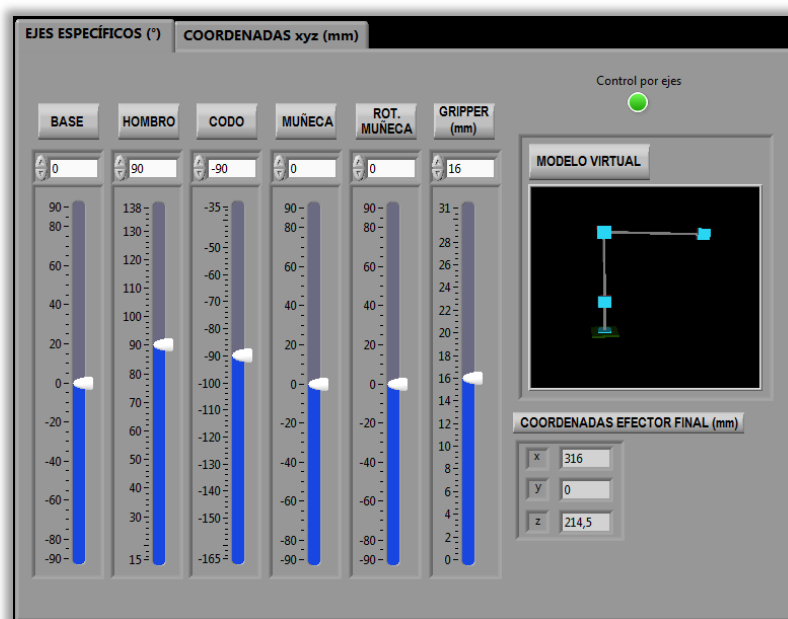
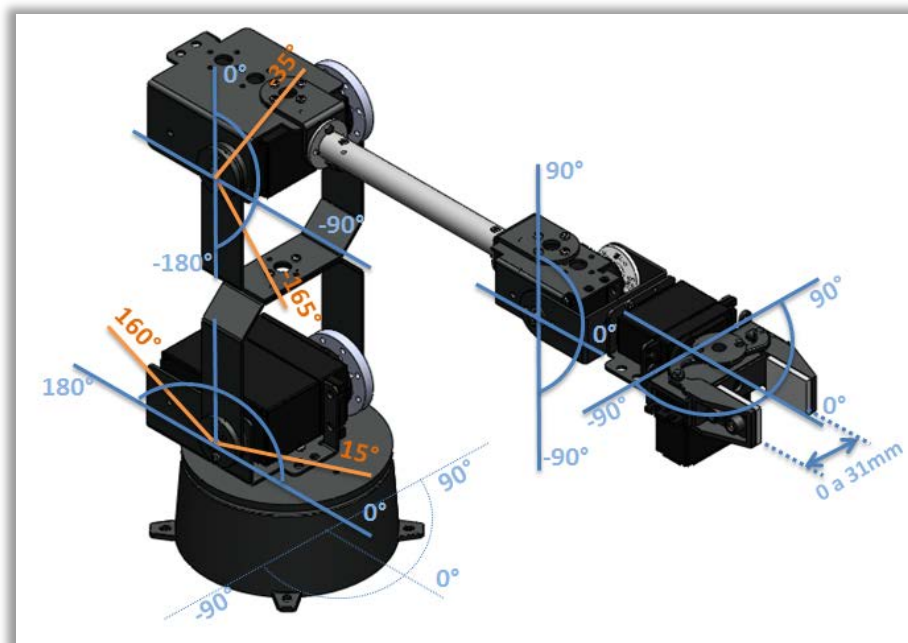


Figura 2.65: Método de operación Ejes Específicos (°).

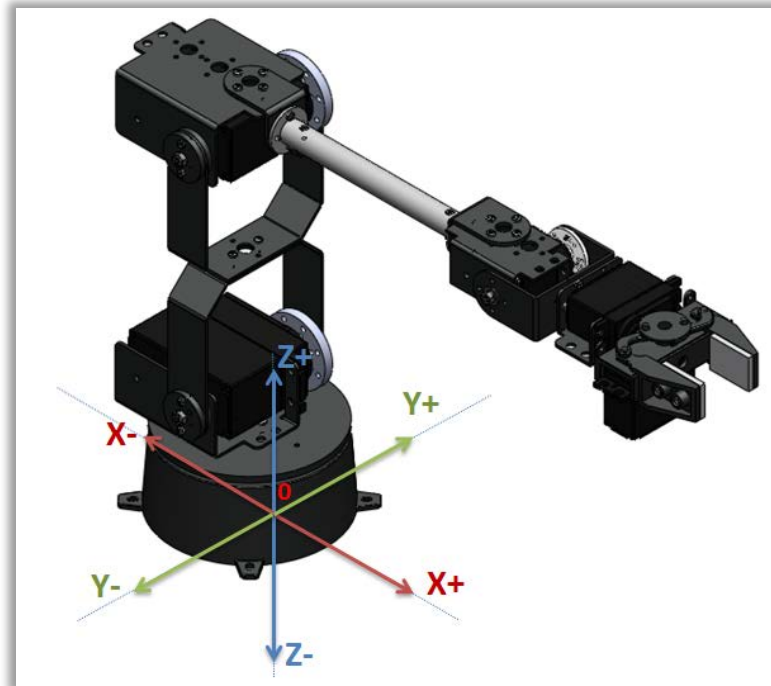
Para la manipulación del robot moviendo cada articulación se establece la siguiente convención de direcciones de ángulos y sus valores basados en la resolución de los problemas cinemáticos. Además debido a restricciones mecánicas del manipulador se ha establecido límites de los movimientos de cada articulación del robot, tal como se puede observar en la Figura 2.66.



**Figura 2.66: Límites de operación de cada articulación del manipulador.**

**Coordenadas xyz (mm).**- Este método permite al usuario desplazar el punto central del efector final en dirección positiva o negativa a lo largo de los ejes del sistema de coordenadas situado al pie del robot, véase la Figura 2.67.

Para el desplazamiento del robot mediante este método, se utilizan tres controles slider para mover en dirección de los ejes X Y y Z, además de dos controles para la orientación de la muñeca (Muñeca Rot) y la apertura-cierre del gripper.



**Figura 2.67: Sistema de coordenadas del manipulador.**

Este método de operación del manipulador básicamente es la resolución del problema cinemático inverso mediante la herramienta Inverse Kinematics VI que provee el módulo de robótica de LabVIEW.

Dicha herramienta facilita cálculos y provee de eficiencia a la aplicación ya que posee algoritmos estructurados que buscan la solución óptima al problema cinemático inverso, todo ello embebido en la misma.

Así se obtienen los valores angulares que las articulaciones deben adoptar para lograr que el actuador final se ubique en una posición concreta definida por los controles slider (xyz) según una determinada localización espacial (coordenadas xyz al pie de la base del robot).

Debido a que la herramienta Inverse Kinematics trabaja siempre dependiente de una posición inicial proporcionada iterativamente, no siempre se obtendrá una solución para distintas posiciones del efector final, ya que dichas posiciones pueden estar demasiado alejadas del área

de trabajo, la posición deseada no es factible, el área es inalcanzable en el proceso de convergencia entre la posición actual y la nueva posición, es por ello que este VI devuelve un error si no puede converger a una solución dentro de los pasos máximos. Es por ello que en la aplicación se ha creado un indicador luminoso que nos advierte que estamos llegando al límite de extensión del robot o al límite del área de trabajo donde el VI puede converger una solución al problema cinemático inverso véase la Figura 2.68.

La solución de la cinemática inversa funciona mejor cuando se está resolviendo sobre una sucesión de posiciones relativamente cercanas, al igual que cualquier algoritmo solucionador este tiene considerables fluctuaciones que dependen de la selección apropiada del número máximo de pasos de análisis.

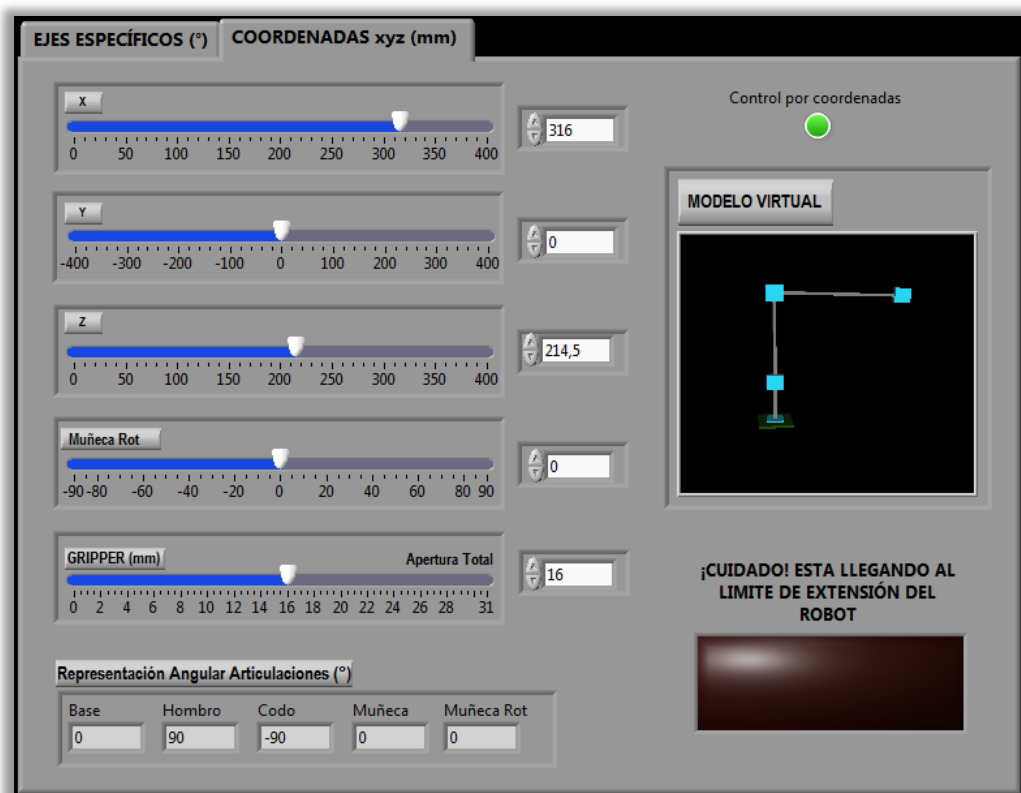
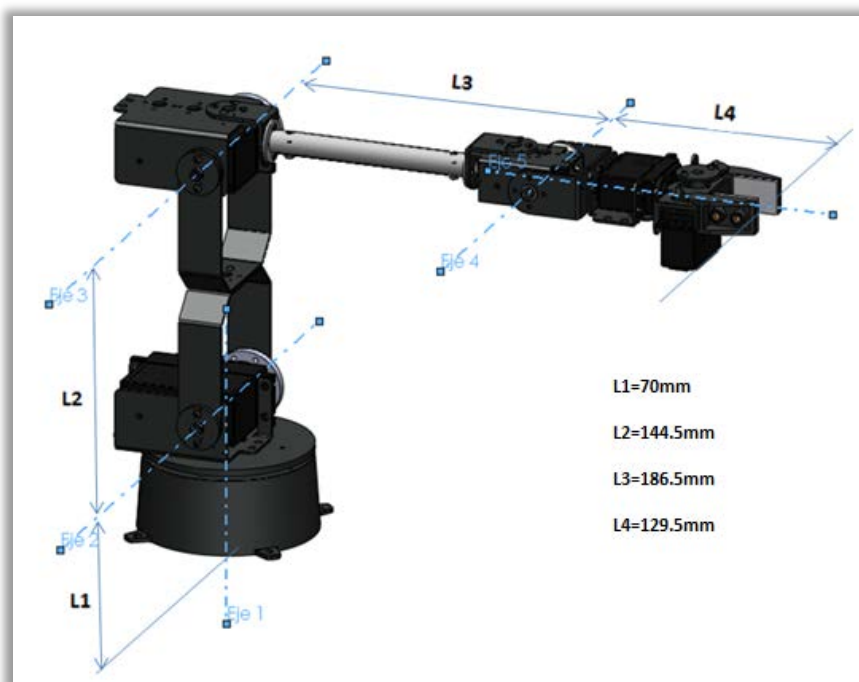


Figura 2.68: Método de operación Coordenadas XYZ (mm).

Tanto el método de operación de Ejes Específicos como el de Coordenadas XYZ utilizan las herramientas del módulo de robótica de LabVIEW para solucionar los problemas cinemáticos directo e inverso respectivamente, las herramientas mencionadas obligatoriamente necesitan parámetros de referencia del robot (parámetros D-H) para la utilización de sus algoritmos.

Es así que se debe identificar la cadena cinemática del robot para determinar dichos parámetros, a continuación se muestra un diagrama esquemático de los ejes del brazo robótico además de la distancia entre eslabones.



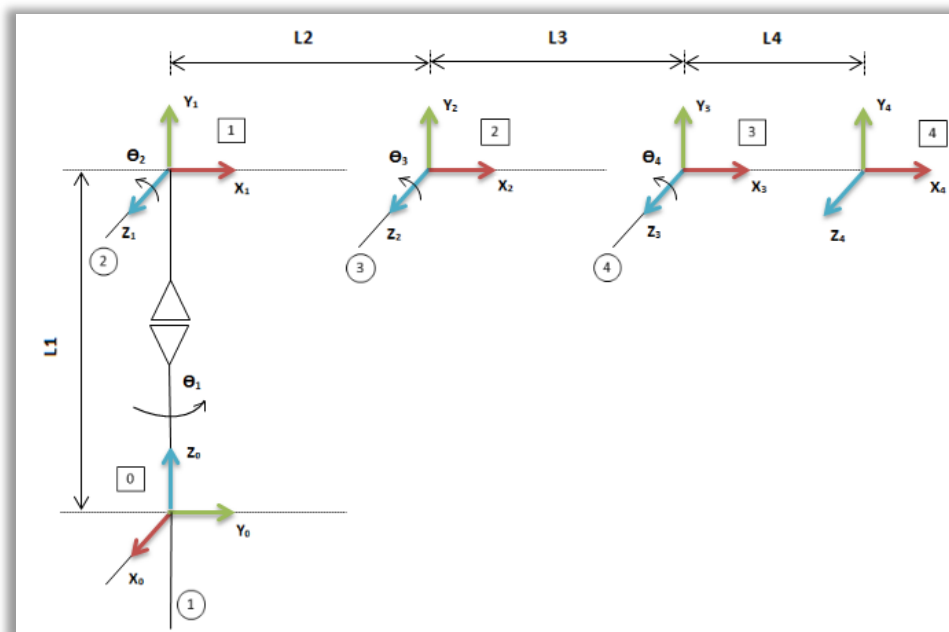
**Figura 2.69: Diagrama con los eslabones y ejes del robot.**

Con el algoritmo propuesto por Denavit-Hartenberg, se procede a seguir cada uno de los ítems del algoritmo para el robot antropomórfico que será analizado como un robot de 4 grados de libertad para facilidad de cálculos y debido a que el movimiento de rotación de la muñeca (eje 5) no se



encarga de posicionar el elemento terminal del robot sino de orientarlo, además no se considera a la pinza como parte de la cadena cinemática del manipulador.

En primer lugar, se localizan los sistemas de referencia de cada una de las articulaciones del robot como se muestra en la Figura 2.70.



**Figura 2.70: Sistemas de referencia y ejes del robot.**

Siguiendo los pasos marcados por el Algoritmo de Denavit-Hartenberg se ha calculado los parámetros correspondientes y se disponen en la siguiente tabla.

**Tabla 2.16: Parámetros D-H del robot.**

ARTICULACIÓN	$\theta$	$d$	$a$	$\alpha$
1	$\theta_1$	L1	0	$90^\circ$
2	$\theta_2$	0	L2	$0^\circ$
3	$\theta_3$	0	L3	$0^\circ$
4	$\theta_4$	0	L4	$0^\circ$

Una vez determinados los parámetros de Denavit-Hartenberg del robot, la aplicación es la encargada de resolver los problemas cinemáticos directo e inverso para controlar la posición del manipulador robótico.

Los demás ítems del algoritmo de Denavit-Hartenberg referidos al cálculo de las matrices de transformación de un sistema a otro ( ${}^{i-1}A_i$ ) y la determinación de la matriz de transformación (T) entre la base y el extremo del robot se las verán en el capítulo de pruebas ya que sirvieron para validar la solución de las herramientas del módulo de robótica con las ecuaciones determinadas mediante el algoritmo D-H.

Implicítamente en el VI Control Robot AL5D (CD y CI) 4gdl se encuentra un algoritmo muy importante que traduce la información suministrada por los modelos virtuales ya analizados en datos que los controladores del manipulador entenderán.

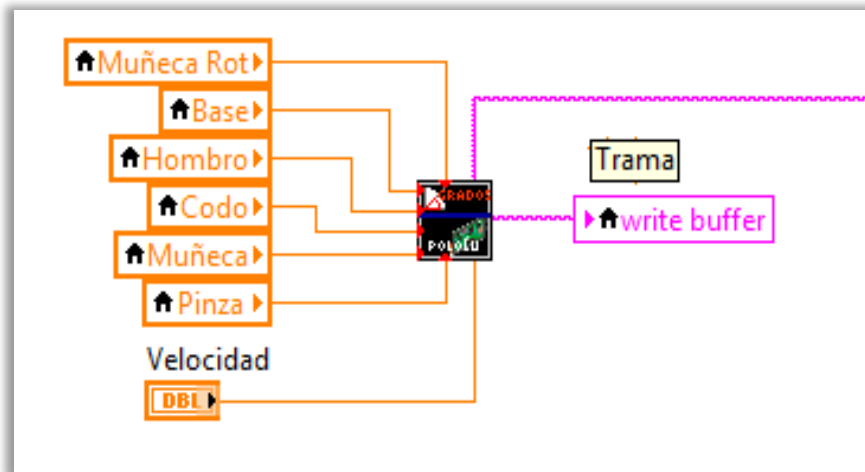


#### **Grados 4gdl.vi**

Una vez que se encuentran resueltos los problemas cinemáticos y la visualización virtual, todo en valores angulares se implementa un nuevo algoritmo para transformar dichos valores angulares en valores de controladora de servomotores, los cuales posicionaran al robot.

Este algoritmo opera de manera que se introducen los valores angulares del robot, la apertura-cierre del gripper en milímetros y el valor de velocidad de operación de servomotores en el VI Grados, el mismo que convierte dichos datos en su correspondiente valor de posición de controladora de servomotores (previamente medidos), luego se ordena todos estos datos en una trama de caracteres ASCII similar a la trama que fabrica el microcontrolador PIC1 para enviar los valores de los sensores a la PCI 2.

En la Figura 2.71 se puede observar el VI Grados con sus entradas y la trama a su salida.



**Figura 2.71: Grados 4gdl.vi**

### **Interfaz HMI-Robot**

Para terminar de describir el HMI solo resta indicar como se realiza la comunicación entre la estación del HMI con la estación remota del robot.

El envío de la trama de datos ASCII desde la aplicación en el computador hacia el controlador del robot se lo realiza inalámbricamente con el tercer módulo XBee montado sobre la tarjeta XBee Explorer USB.

Solo se necesita conectar el módulo XBee en la base del Explorer y conectar el cable USB al computador, de esta manera se tiene acceso a los pines seriales para operación y configuración del módulo.

Una vez implementada la interfaz serial entre el ordenador y el módulo XBee se realiza el algoritmo de la Figura 2.72, para el envío de la trama de datos en forma 8N1, que es la abreviatura de la configuración del puerto serie en modo asíncrono donde hay ocho bits de datos, ningún bit de paridad, un bit de parada y la misma velocidad de transmisión de datos con la que se trabaja en todo el proyecto.

El módulo inalámbrico de la estación del HMI está direccionado para enviar la información al módulo inalámbrico montado en la PCI 2 de la estación remota, el mismo que transferirá la trama hacia el microcontrolador PIC2 que será el encargado de realizar el control del robot, del mismo modo como lo hace con los datos enviados del guante sensorizado.

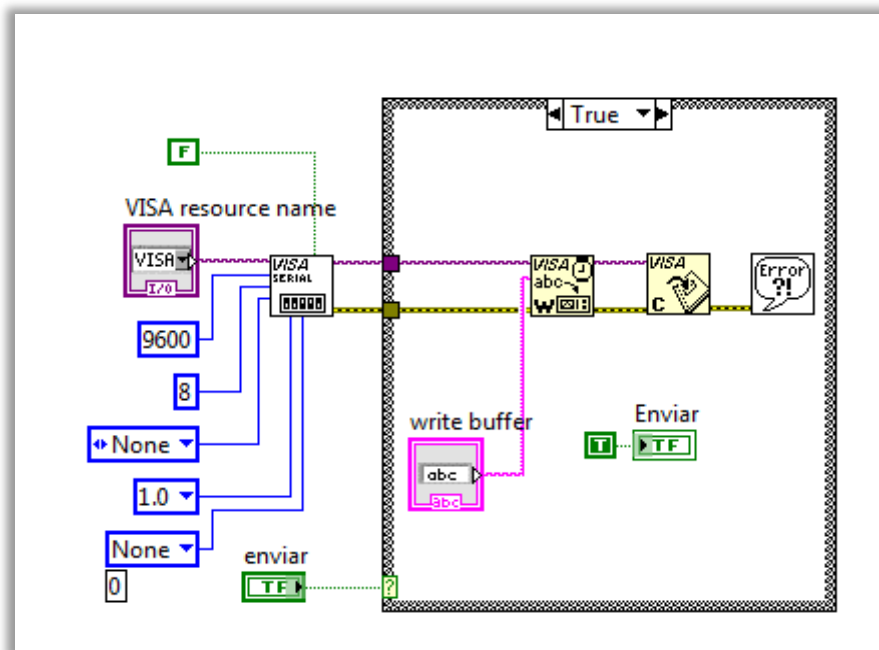


Figura 2.72: Algoritmo envío serial de datos.

## 2.7 IMPLEMENTACIÓN DE LOS COMPONENTES FÍSICOS

Para respaldar el entorno de trabajo del manipulador y los componentes del hardware en general del proyecto se debe considerar ciertos factores, como: la estructura mecánica del prototipo, el área de trabajo, la disposición de las herramientas de trabajo, el soporte para los elementos electrónicos y los compartimentos para almacenaje y protección de los elementos.

### 2.7.1 ESTRUCTURA MECÁNICA DEL ENTORNO DE TRABAJO DEL ROBOT

La estructura sobre la cual se posiciona al manipulador y se delimita su operación de trabajo está construida con perfiles de aluminio de color negro para los soportes, vidrio de 3 líneas para sus caras y un tablero MDF como base.

Las dimensiones de la estructura mecánica exteriormente son 114x83x69cm tal como se muestra en la Figura 2.73.



**Figura 2.73: Dimensiones estructura mecánica.**

Adicionalmente sobre la base de la estructura se encuentran montados dos compartimentos en forma de cajas, uno de ellos con dimensiones de 62x14x8cm hecho de policarbonato, el cual alberga y protege toda la circuitería de la estación remota, el segundo compartimento fabricado de madera con dimensiones de 30x15x8.7cm, con la función de almacenar todo el hardware del guante sensorizado véase la Figura 2.74.



**Figura 2.74: Vista superior de la estructura mecánica.**

## **2.7.2 MONTAJE DEL HARDWARE GUANTE DE DATOS (ESTACIÓN LOCAL)**

Se optó por adquirir una caja de montaje electrónico de dimensiones 13x7x4.7cm, para situar y mantener los elementos y circuitería general del guante sensorizado. Además se adaptó a la caja un clip de correa para montar y desmontar el hardware del guante sensorizado en el operador y así hacer más fácil la operación de teleoperar, véase la Figura 2.75.



**Figura 2.75: Montaje del hardware guante de datos.**

Una vez instalado la PCI 1, su fuente de alimentación, las extensiones de los sensores y del stick analógico, los elementos de protección y de activación, se tiene todo el hardware del guante sensorizado como se muestra en la Figura 2.76.



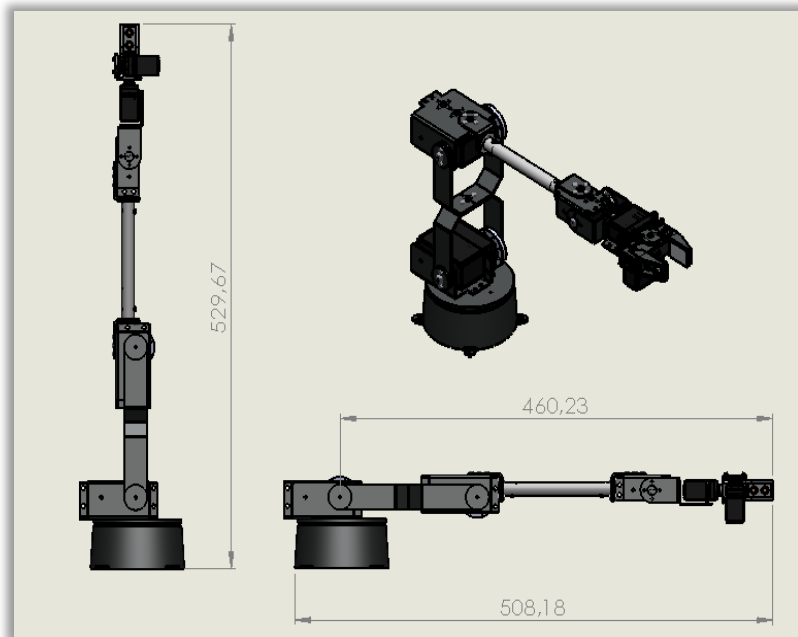
**Figura 2.76: Hardware guante de datos.**

### **2.7.3 ZONA DE TRABAJO**

La zona de trabajo del manipulador está determinada por las características físicas del robot (eslabones, tamaño y forma), esta área se refiere al volumen espacial al que puede llegar el extremo del robot.

Las dimensiones máximas del manipulador indican que se puede alcanzar hasta una altura máxima de 529.67mm y extendido completamente el brazo en forma horizontal desde el centro de la base (origen de coordenadas) un alcance de 460.23mm.

Las dimensiones máximas que puede alcanzar el manipulador se muestra en la Figura 2.77.



**Figura 2.77: Dimensiones máximas del área de trabajo del manipulador.**

Hay que aclarar que las medidas indicadas no contemplan la restricción mecánica que ejerce el resorte, el mismo que reduce el área de trabajo del robot pero no en manera tan significativa.

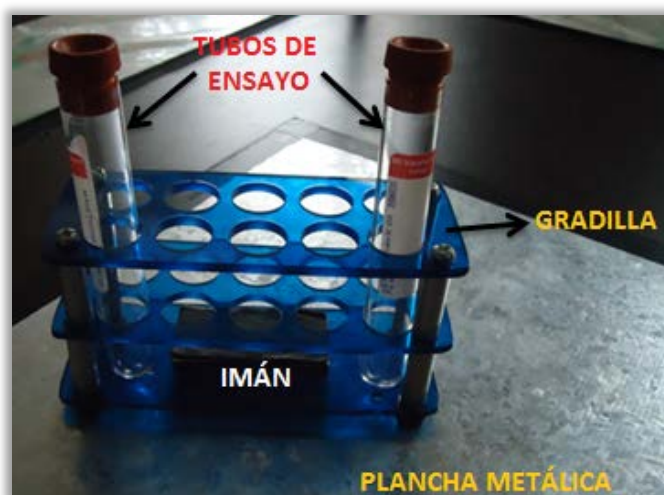
#### **2.7.4 HERRAMIENTAS DE TRABAJO**

Como se habrá notado en la Figura 2.74, se encuentran dispuestos dentro del área de trabajo del manipulador, herramientas de laboratorio denominadas gradillas y sobre ellas tubos de ensayo.

Dichas herramientas son usadas comúnmente en laboratorios clínicos y en laboratorios de investigación, en el caso de la aplicación específica del proyecto se utilizan los tubos de ensayo para contener el material peligroso y las gradillas para sostener los tubos y así facilitar su manejo y almacenaje.



Previamente se pensó en fijar las gradillas en la base de la estructura, pero con ese hecho se tendría un ambiente totalmente definido y conocido con lo cual la potencialidad de la aplicación sería limitada básicamente a una tarea repetitiva. Por tal motivo no se fija de manera constante las gradillas en una sola posición, se adiciona un imán a cada gradilla y planchas metálicas en la base de la estructura (dentro del área de trabajo del robot) que suponen un modo de fijación no permanente véase la Figura 2.78, con lo cual se provee de facilidad para simular un ambiente no estructurado con respecto a la ubicación de las gradillas y el contenedor del material peligroso.



**Figura 2.78: Fijación de herramientas.**

## CAPÍTULO III

### PRUEBAS Y RESULTADOS

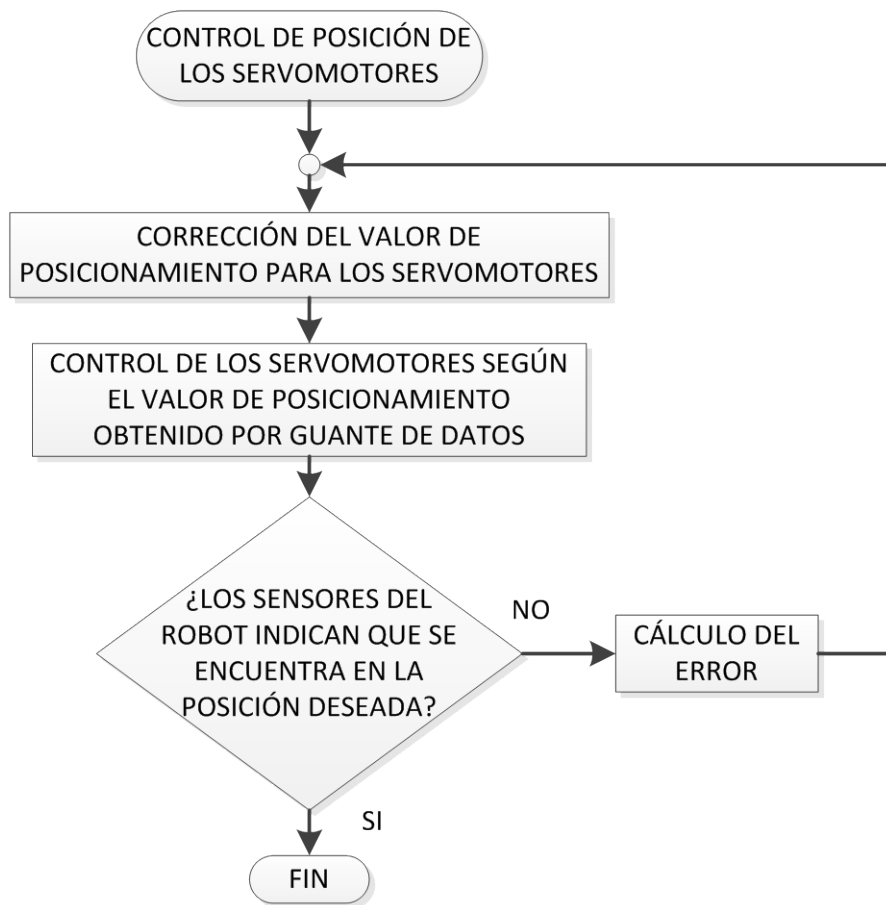
#### 3.1 PRUEBAS DE FUNCIONAMIENTO DEL CONTROL CON USO DE REALIMENTACIÓN

A pesar que realizar el posicionamiento de los servomotores no requiere un control en lazo cerrado, se optó durante el proceso de diseño instrumentar el robot con acelerómetros que entreguen una señal que represente la posición de las articulaciones del robot (véase Figura 3.1).



**Figura 3.1: Acelerómetro sobre las articulaciones del codo del robot manipulador.**

Se realizaron pruebas con la articulación del codo y la inclinación de la muñeca adhiriendo al software del microcontrolador de la estación remota el algoritmo descrito en el diagrama de flujo de la figura 3.2 para así agregar un control en lazo cerrado para el posicionamiento.



**Figura 3.2: Diagrama de flujo del control de posicionamiento con realimentación.**

En teoría el algoritmo debe funcionar, pero en la práctica esto no ocurre debido a que se utilizan acelerómetros para obtener el valor de realimentación.

Durante el posicionamiento del robot, su movimiento es instantáneo y los sensores captan la aceleración lineal y no lo estática (inclinación), con este valor la corrección se vuelve completamente errónea entregando valores falsos lo que hace imposible la teleoperación.

De todas maneras cabe recalcar, que así funcionase la realimentación, ésta entregaría valores iguales o muy cercanos a los deseados ya que se utilizan servomotores los cuales en si ya poseen un control interno de

posición, y además en el presente caso no existe perturbación externa que cambie la posición de estos dispositivos. Todo ello al final de cuentas hace que el uso de un control en lazo cerrado sea innecesario para el posicionamiento del robot.

Por estas razones, finalmente el robot manipulador no posee un control en lazo cerrado para el posicionamiento de sus servomotores.

## **3.2 PRUEBAS DE TELEOPERACIÓN MEDIANTE EL GUANTE SENSORIZADO**

Para el éxito de la teleoperación del robot manipulador mediante el guante sensorizado se debe verificar el correcto funcionamiento del control de los servomotores del robot mediante el guante de datos y el nunchuk.

### **3.2.1 PRUEBAS DE POSICIONAMIENTO MEDIANTE EL GUANTE DE DATOS**

#### **POSICIÓN DE HOME**

La posición **HOME** es la posición inicial de operación de todo robot manipulador, dicho esto, para realizar la teleoperación mediante el guante sensorizado es necesario representar ésta posición por una posición adecuada del brazo humano.

Colocar el brazo humano de manera idéntica a la del brazo robótico, como se indica en la figura 3.3, resulta totalmente incomodo, sin tener en cuenta que se debería pasar por esta posición o similares en muchas ocasiones durante la teleoperación, por tal razón esta opción ha sido descartada completamente.



**Figura 3.3: Imitación de la forma del robot con el brazo humano.**

Se determinó que la manera más cómoda de mantener el brazo en una posición que represente el **HOME** y que desde ésta resulte fácil el control del robot es la que se muestra en la figura 3.4.



**Figura 3.4: Posición brazo humano /HOME robot manipulador.**

Una vez establecida la posición para el **HOME**, se lee los valores que entregan los sensores del guante sensorizado y se ingresan al microcontrolador. El algoritmo de control calcula las posiciones para los servomotores partiendo de los valores de los sensores del guante en la posición que representa al **HOME**.


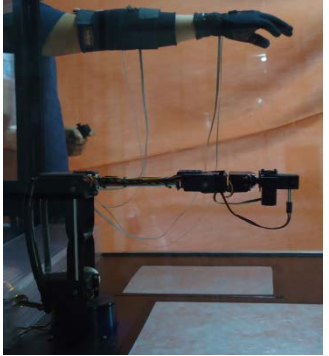



### **POSICIONAMIENTO DEL ROBOT MANIPULADOR**

Dependiendo de la inclinación del brazo, antebrazo y mano del operador se tendrá una respuesta similar en la posición de los eslabones del robot,

pero ya que la forma del brazo humano que representa el **HOME** del robot no es idéntica se deberá verificar las posiciones que tome el robot manipulador dependiendo de las diferentes configuraciones del brazo del operador humano.

A continuación en la tabla 3.1 se puede apreciar el resultado del control con el guante de datos probando las diferentes posiciones del brazo del operador.

**Tabla 3.1: Posiciones del brazo humano para la teleoperación del robot manipulador.**

<b>A</b>	<b>B</b>	<b>C</b>
		
<b>Brazo recto inclinado</b>	<b>Brazo recto horizontal</b>	<b>Brazo recto declinado</b>
<b>D</b>		<b>E</b>
		
<b>Brazo con doblez de codo arriba</b>		<b>Brazo con doblez de codo abajo</b>

La figura A, B y C demuestran que si se inclina el brazo recto del operador hacia alguna dirección el robot realizará movimientos similares en cada uno de sus eslabones.

En la figura D se puede apreciar que doblando el codo del operador hacia arriba el robot toma una posición tal que el efector final se acerca a la base del manipulador. En la figura E se observa que doblando el codo del operador hacia abajo el robot toma una posición que aleja de la base el efector final.

Se determinó que el operador debe empezar con el brazo recto y hacer uso de los dos últimos movimientos (D, E) para dependiendo del caso alejarse o acercarse al objeto a manipular y realizar un correcto posicionamiento del efector final. Si el objeto a manipular se encuentra en una posición ni muy alejada ni muy cercana a la base se puede hacer uso del movimiento C y A.

Todas estas configuraciones del brazo con el guante sensorizado deben combinarse y realizarse con mucho cuidado para adecuarse a la correcta forma de teleoperación del manipulador robótico.

## **POSICIONAMIENTO Y SEGURIDAD EN EL EJE Z**

Como se ya se ha dicho, el robot cuenta con una seguridad mediante software que evita que la punta del efector final se desplace a menos una posición especificada en el eje Z.

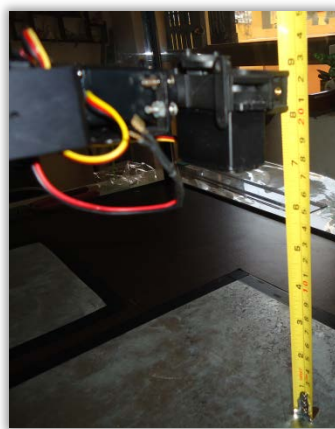
Este concepto es usado principalmente para evitar que el robot colisione con el suelo por acción descuidada del operador.

Su funcionamiento depende de la exactitud con la que el robot se posicione. Las pruebas se realizaron utilizando el HMI para posicionar el robot manipulador en diferentes configuraciones y en valores iguales del eje Z.

Acto seguido, en cada una de las configuraciones se midió físicamente el valor real en el eje Z en el cuál se encuentra la punta del efector final (Véase Figura 3.5). En la siguiente tabla se listan los resultados de las mediciones así como el error.

**Tabla 3.2: Verificación del posicionamiento del robot en el eje Z.**

Posición deseada (HMI) x,y,z	Posición real en Z (Medición)	Error Absoluto (mm)	Error Relativo
<b>Cerca de la base</b>			
50,-112, <b>20</b>	21	1	5%
237,0, <b>110</b>	106	-4	3.6%
180,108, <b>200</b>	196	-4	2%
90,160, <b>280</b>	272	-8	2.8%
<b>Lejos de la base</b>			
250,0, <b>20</b>	22	2	10%
170,240, <b>110</b>	117	3	2.7%
163,-195, <b>200</b>	194	-6	3%
80,-162, <b>280</b>	283	3	1%
<b>Muy lejos de la base</b>			
110,-338, <b>20</b>	26	6	30%
324,180, <b>110</b>	84	-36	32.7%
390,-90, <b>200</b>	170	-30	15%
200,322, <b>280</b>	211 (75)	-69 (-205)	24% (73.2%)



**Figura 3.5: Medición del efector final en el eje Z.**



Cuando el robot opera dentro de un rango de trabajo no muy alejado de la base el posicionamiento en Z tiene errores muy bajos que se representan por unos pocos milímetros, sin embargo si se opera el robot en un área bastante alejada de la base, físicamente el robot toma una posición estirada de sus eslabones provocando esfuerzo inadecuado para cada una de sus articulaciones lo que hace que no logre posicionarse correctamente por la falta de potencia de sus servomotores.

Es por esto que en este caso se observa un error muy elevado. Se comprobó que el robot nunca debe trabajar en posiciones muy alejadas de la base ya que no fue diseñado para tal operación.

Para posicionarse en la última medición de la tabla, si ésta se inicia desde una posición previa que tenga el valor en el eje Z mayor a 280mm el robot termina con una posición de 211mm, pero si se inicia el posicionamiento con un valor en Z casi cercano al suelo el robot no es capaz de subir el efector final quedando únicamente a 75mm lo que provoca demasiado error. Gracias a esto también se pudo determinar que el robot al manipular un objeto pesado cercano al suelo no debe subir directamente en el eje Z, primero se deben contraer sus articulaciones para acercarse a la base y luego proceder a subir en el eje Z.

En aquellas mediciones cercanas al suelo (20mm) se puede observar que el robot se posiciona con un valor mayor al deseado en unos pocos milímetros, y ya que el error no es demasiado se estableció dentro del algoritmo para la seguridad en el eje Z un valor de 10mm.

En general se puede observar en la tabla que las mediciones físicas son diferentes a las que indica el HMI, esto se debe a factores como el diseño impreciso de las articulaciones del robot, a que los servomotores presentan pequeños errores en sus posicionamientos y a posibles pequeños errores en las mediciones angulares. Estos errores aunque

pequeños, en conjunto provocan que la posición del efector final tenga variaciones.

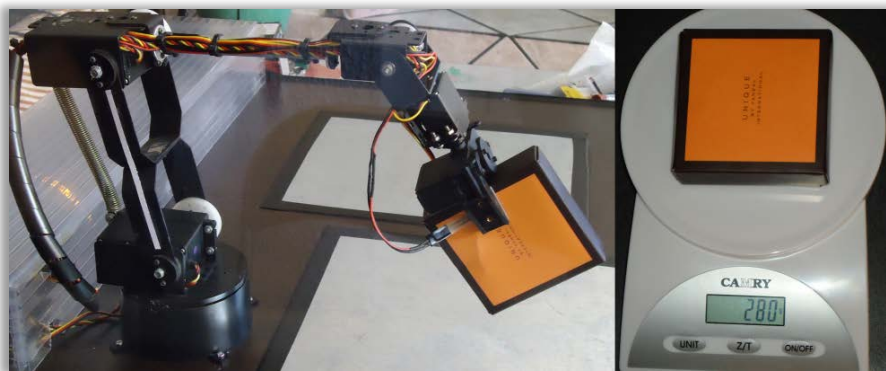
### 3.2.2 PRUEBAS DE MANIPULACIÓN DE OBJETOS

Con el sistema de teleoperación en correcto funcionamiento se procede a realizar pruebas que verifican la capacidad de manipulación del robot haciendo uso del guante sensorizado y analizan el nivel de dificultad de cada operación.

#### OBJETOS ALEATORIOS

Primeramente se debe verificar físicamente la carga máxima que puede levantar el robot manipulador, teniendo en cuenta que mientras más cerca se encuentre el objeto a la base del robot más peso podrá levantar.

La carga máxima que levanta el robot cerca de la base es de 280 gramos (aproximadamente dos candados metálicos tamaño estándar), como se observa en la figura 3.6.



**Figura 3.6: Peso máximo de carga del robot.**

La carga máxima que puede levantar el mientras se encuentre extendido es de 130 gramos. Una vez conocido con exactitud el peso máximo

posible a levantar se realizan pruebas de manipulación de objetos aleatorios en diferentes posiciones dentro del área de trabajo del robot manipulador, analizando para cada una el nivel de dificultad que presenta para el operador en un rango de 1 a 10, tomando a 10 como “Imposible de manipular” y 1 como “Completamente fácil de manipular”. En la tabla 3.3 se puede apreciar el resultado de cada una de estas pruebas.

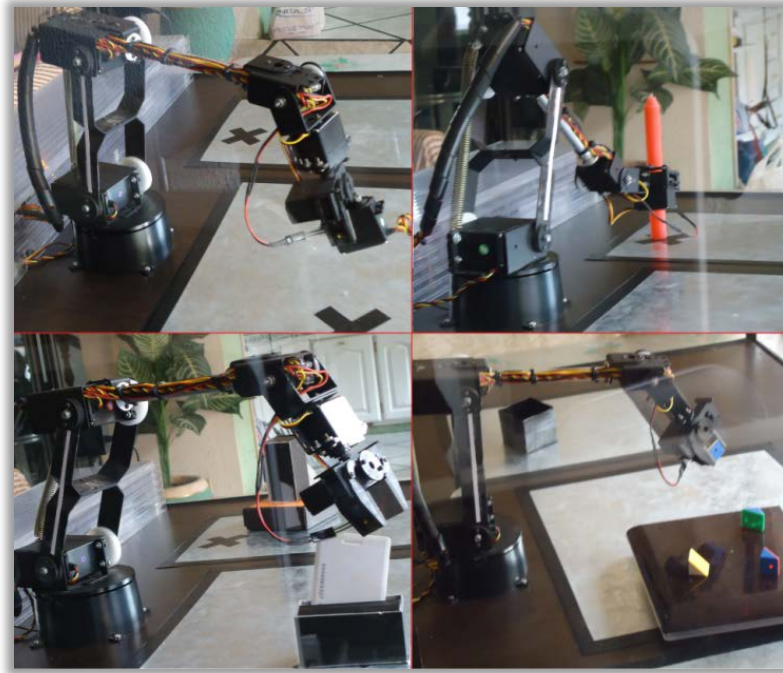
**Tabla 3.3: Resultado de las pruebas de teleoperación con objetos aleatorios.**

Prueba	Objeto	Peso	Nivel de dificultad	
			Operador 1	Operador 2
<b>A</b>	Servomotor	55 g	2	4
<b>B</b>	Vela	29 g	6	7
<b>C</b>	Tarjeta	9 g	8	9
<b>D</b>	Figuras	1 g	4	5
<b>E</b>	Tubo ensayo	12 g	5	6

La prueba A consiste únicamente en transportar el servomotor de una ubicación a otra. Esta prueba representa el transporte simple de objetos y no representa mayor dificultad. La prueba B consiste en transportar la vela de un lugar a otro de tal manera que el operador pueda colocarla de forma correcta sin dejarla caer. En esta prueba el operador debe ser capaz de colocar el objeto con precisión para evitar el colapso del mismo.

En la prueba C el operador debe llevar la tarjeta desde una posición vertical a una posición horizontal dentro de unos compartimentos haciendo uso del giro de la muñeca. Esta prueba representa un nivel de dificultad exigente ya que se debe colocar el objeto de una manera diferente a la que se lo sujetó. La prueba D se trata de transportar pequeños objetos hacia una caja, simulando la operación de paletizado. Aquí el operador pone a prueba su capacidad para controlar el robot por

un largo periodo de trabajo. Finalmente la prueba E representa la operación para la cual fue concebido el proyecto, dicha operación se analizará posteriormente. En la figura 3.7 se observa la manipulación de algunos objetos de las pruebas.



**Figura 3.7: Teleoperación con manipulación de objetos aleatorios.**

En las pruebas realizadas el Operador 1 posee mayor experiencia de trabajo con el sistema de teleoperación que el Operador 2, es por esto que se observa una diferencia significativa en el nivel de dificultad de cada operación, evidenciando la necesidad de que un nuevo operador deberá practicar lo suficiente para conseguir un total control del robot manipulador con el guante sensorizado.

### **TUBOS DE ENSAYO CON MATERIAL PELIGROSO**

Transportar material peligroso desde un lugar a otro es el principal objetivo en la teleoperación del manipulador robótico mediante el guante sensorizado.

Como ya se ha explicado anteriormente dicho proceso se lo realiza colocando gasolina como material peligroso dentro de los tubos de ensayo sellados y se transporta de una gradilla a otra dentro del entorno de trabajo del robot manipulador. Realizar esta operación requiere que durante el proceso se cumpla con dos requisitos esenciales: lo más rápido posible y con la mayor precisión posible para evitar movimientos del material.

Se realizaron pruebas considerando cada uno de los factores por separado y en conjunto, y de todas ellas se cronometró el tiempo. En la tabla 3.4 se puede observar estas pruebas.

Antes que nada cabe indicar que la operación de sujetar, manipular y colocar el tubo de ensayo no es una operación extremadamente difícil, obviamente tiene un nivel de dificultad considerable y requiere que el operador sepa utilizar correctamente el guante sensorizado.

**Tabla 3.4: Pruebas de manipulación de material peligroso.**

<b>Prueba</b>	<b>Solo Rapidez</b>	<b>Solo Precisión</b>	<b>Distancias entre gradillas</b>	<b>Tiempo</b>
1	X		32 cm	34 s
2		X		1min 42s
3	X	X		1min
4	X		60cm	36s
5		X		1min 27s
6	X	X		1min 1s
7	X		75cm	50s
8		X		1min 33s
9	X	X		1min 19s

Se observa que en el transporte donde se llevó rápido el tubo de ensayo, el material peligroso se agita demasiado, si no se encontraran tapados los contenedores el material se hubiera derramado.

Este inconveniente se corrige al transportar el material de una manera muy lenta para ganar estabilidad y precisión de los movimientos. Si se combinan de manera eficiente la velocidad y la precisión, se obtiene el mejor de los resultados ya que se optimiza el tiempo para el transporte del material y no existen movimientos bruscos que resulten como desperdicio de material y peligro para el entorno de teleoperación.

### **3.3 PRUEBAS DE POSICIONAMIENTO MEDIANTE EL HMI**

Esta sección se enfoca en el posicionamiento del manipulador robótico en el espacio de trabajo mediante la operación del HMI. Para este fin se evidencian los aspectos de la resolución del modelo cinemático del manipulador y el sistema de control de posicionamiento del mismo.

#### **3.3.1 PRUEBAS DE RESOLUCIÓN DEL MODELO CINEMÁTICO**

Para el caso particular de la aplicación mediante el HMI, se considera de manera primordial el posicionamiento del extremo del efector final, más no su orientación.

En consecuencia, para la resolución del problema cinemático del manipulador se lo analizó como una cadena cinemática de cuatro grados de libertad, apartando del análisis el movimiento de rotación de la muñeca, el mismo que forma parte de la orientación del efector más no de su posicionamiento.

El movimiento de apertura-cierre del gripper tampoco es considerado como parte de la cadena cinemática para el análisis.

Al realizar este desacople cinemático entre posición y orientación del manipulador, se facilita la resolución del problema cinemático como se verá a continuación.

Una vez determinados los parámetros Denavit-Hartenberg del manipulador (Tabla 2.15) se calculan las matrices  ${}^{i-1}A_i$ , sustituyendo los parámetros D-H en la ecuación matricial (Ec. 1.4) de la siguiente manera:

$${}^0A_1 = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & L_2C_2 \\ S_2 & C_2 & 0 & L_2S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2A_3 = \begin{bmatrix} C_3 & -S_3 & 0 & L_3C_3 \\ S_3 & C_3 & 0 & L_3S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3A_4 = \begin{bmatrix} C_4 & -S_4 & 0 & L_4C_4 \\ S_4 & C_4 & 0 & L_4S_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Para calcular la matriz de transformación homogénea T, se utiliza la Ecuación 1.2, que para los 4 GDL del modelo analizado se expresa de la siguiente manera:

$$T = {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3 \cdot {}^3A_4$$

Una vez realizadas las operaciones de las matrices, se obtiene la siguiente expresión:

$$T = \begin{bmatrix} C_1 C_{234} & -C_1 S_{234} & S_1 & L_2 C_1 C_2 + L_3 C_1 C_{23} + L_4 C_1 C_{234} \\ S_1 C_{234} & -S_1 S_{234} & -C_1 & L_2 S_1 C_2 + L_3 S_1 C_{23} + L_4 S_1 C_{234} \\ S_{234} & C_{234} & 0 & L_1 + L_2 S_2 + L_3 S_{23} + L_4 S_{234} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Expresando la matriz de transformación homogénea T en forma de la Ecuación 1.5 se tiene:

$$\begin{aligned} n_x &= C_1 C_{234} & n_y &= S_1 C_{234} & n_z &= S_{234} \\ o_x &= -C_1 S_{234} & o_y &= -S_1 S_{234} & o_z &= C_{234} \\ a_x &= S_1 & a_y &= -C_1 & a_z &= 0 \end{aligned}$$

$$\left. \begin{aligned} P_x &= L_2 C_1 C_2 + L_3 C_1 C_{23} + L_4 C_1 C_{234} \\ P_y &= L_2 S_1 C_2 + L_3 S_1 C_{23} + L_4 S_1 C_{234} \\ P_z &= L_1 + L_2 S_2 + L_3 S_{23} + L_4 S_{234} \end{aligned} \right\} \text{ Ec. 3.1}$$

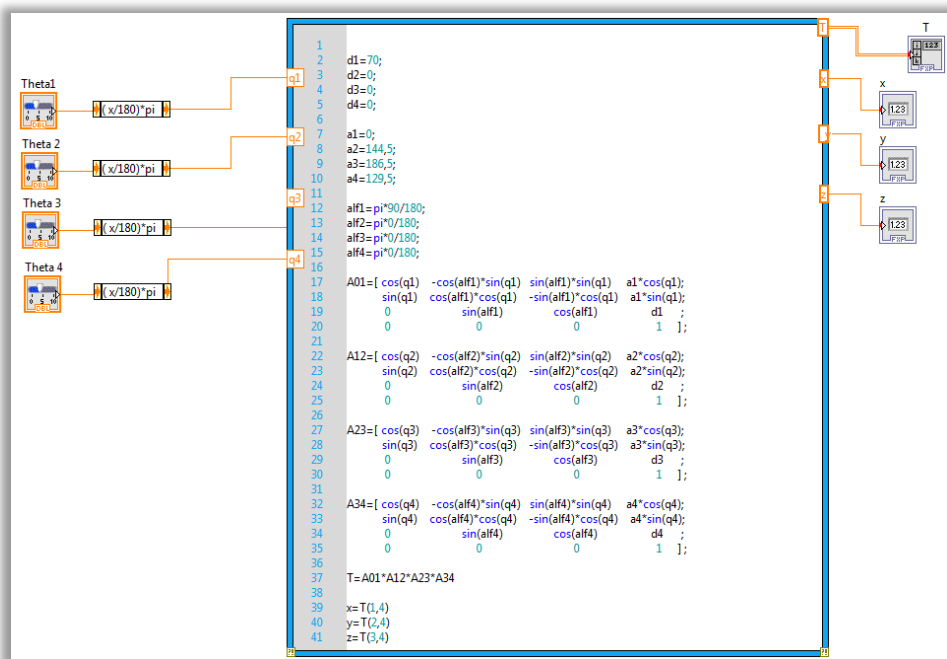
El resultado de las ecuaciones obtenidas muestra el vector de la posición (Px, Py, Pz) que es justo lo que se busca para esta aplicación y adicionalmente la terna ortonormal que representa la orientación (n, o, a), todo en función de las coordenadas angulares ( $\theta_1, \theta_2, \theta_3, \theta_4$ ) y las longitudes de los eslabones del robot.

Las ecuaciones determinadas permiten posicionar el extremo del robot respecto a un sistema de coordenadas rectangulares ( $x_0, y_0, z_0$ ) situada al centro de la base del robot.

Una vez resuelta la cinemática directa del manipulador se obtuvieron las Ecuaciones 3.1, para validar las mismas se emplea la estructura MathScript Node, en la cual se pueden ingresar fórmulas de tipo texto en lenguaje de Matlab, directamente dentro del diagrama de bloques.



El MathScript es una herramienta del módulo NI LabVIEW MathScript RT, la cual añade programación textual orientada a matemáticas al entorno de desarrollo gráfico. Dicha herramienta permite evaluar las ecuaciones resultantes introduciendo los valores numéricos por los términos expresados en variables ( $\theta_1, \theta_2, \theta_3, \theta_4$ ). (Véase figura 3.8).



**Figura 3.8: Diagrama de bloques de la aplicación en Math Script.**

De esta forma se realizó un script para la evaluación de las ecuaciones obtenidas mediante la matriz de transformación homogénea. La aplicación tiene declarados los parámetros constantes como son las longitudes de los eslabones ( $L_1 = 70\text{mm}$ ,  $L_2 = 144.5\text{mm}$ ,  $L_3 = 186.5\text{mm}$ ,  $L_4 = 129.5\text{mm}$ ) y los ángulos  $\alpha$  en radianes, todos dispuestos de manera que indican los parámetros D-H de la tabla 2.16.

Como se puede notar, Theta 1, Theta 2, Theta 3, y Theta 4 son las únicas variables y son los ángulos de cada una de las articulaciones en grados, los cuales se transforman en radianes y son guardados en las variables  $q_1, q_2, q_3$  y  $q_4$  respectivamente para posteriormente ser evaluados.

Finalmente la aplicación al evaluar las ecuaciones, entrega como resultado los valores de la matriz de transformación homogénea T y separando de la misma se obtiene las coordenadas x, y, z del vector posición del extremo del efector final, situado en un espacio cartesiano.

En la Tabla 3.5 se expone el resultado de contrastar las ecuaciones evaluadas mediante el **Math Script** con la resolución del problema cinemático directo mediante la herramienta **Forward Kinematics VI** utilizada en el HMI.

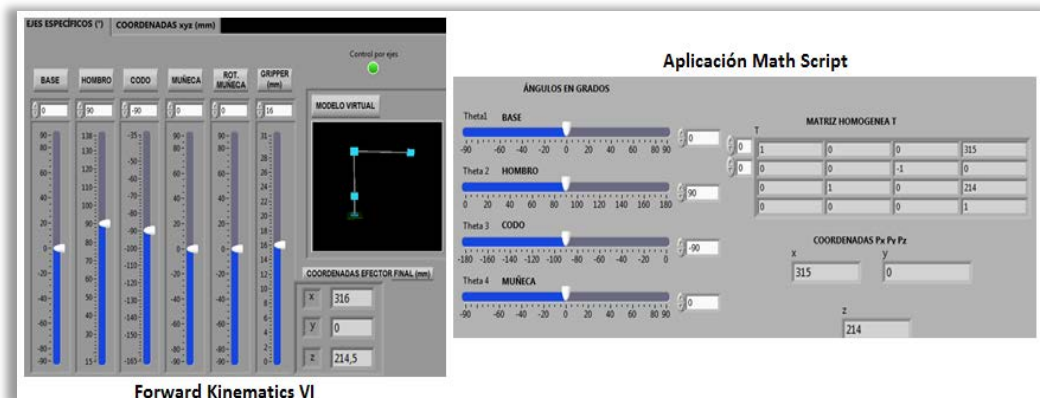
Para la prueba se utilizó cinco configuraciones angulares que definen distintas posiciones del robot, dónde se considera que la posición ideal es la obtenida mediante las ecuaciones evaluadas en el Math Script.

**Tabla 3.5: Prueba de validación del modelo cinemático directo.**

COORDENADAS ÁNGULARES				COORDENADAS VIRTUALES DEL EFECTOR FINAL						ERROR RELATIVO PROMEDIO %
BASE	HOMBRO	CODO	MUÑECA	Forward Kinematics VI			Aplicación con Math Script			
( $\theta_1$ )	( $\theta_2$ )	( $\theta_3$ )	( $\theta_4$ )	x	y	z	x	y	z	
0	90	-90	0	316	0	214.5	315	0	214	0.184
45	70	-100	60	228.456	228.456	177.286	228	227.723	176.816	0.263
-45	50	-50	90	197.553	-197.553	310.193	197	-196.97	309.31	0.287
-90	60	-70	-90	0	-233.429	35.2227	0	-232.77	35.3689	0.232
0	40	-35	90	285.197	0	308.145	284	0	307.281	0.234

Como se observa en la tabla, el error obtenido es ínfimo, por lo que se valida el modelo cinemático directo del manipulador desarrollado con la herramienta Forward Kinematics VI del HMI.

En la figura 3.9 se observa los resultados de las componentes del vector posición de la pinza, tanto en la aplicación utilizada para el HMI y la aplicación mediante el Math Script, con respecto a la primera configuración angular utilizada en la prueba de la Tabla 3.5.



**Figura 3.9: Comparación Forward Kinematics vs Math Script.**

Con la resolución de la cinemática directa del robot mediante la aplicación desarrollada en LabVIEW, se considera de forma ideal cualquier valor angular ingresado para posicionar las articulaciones del manipulador; sin embargo las articulaciones integradas por los actuadores tienen un rango de giro de  $0^\circ$  a  $180^\circ$ , además se debe de considerar las restricciones mecánicas propias del robot que impiden el movimiento en el rango especificado.

De tal manera se determina de manera experimental rangos de valores angulares en los que las articulaciones pueden moverse de manera independiente (véase la tabla 3.6).

**Tabla 3.6: Rangos de movimientos de las articulaciones del manipulador**

ARTICULACIÓN	RANGO DE MOVIMIENTO
BASE	$90^\circ$ a $-90^\circ$
HOMBRO	$15^\circ$ a $160^\circ$
CODO	$-35^\circ$ a $-165^\circ$
MUÑECA	$90^\circ$ a $-90^\circ$
ROTACIÓN MUÑECA	$90^\circ$ a $-90^\circ$
GRIPPER	0mm a 31mm

Véase la Figura 2.66 donde se muestra la convención de las direcciones y valores de los rangos de movimiento de cada articulación.

Para comprobar la validez del modelo cinemático inverso se verifican algunas posiciones de interés del efector final con respecto al origen de coordenadas al pie del robot, mediante el uso del modelo cinemático directo.

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

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 3.7, se realizó la prueba para siete valores de interés.

En los resultados de esta prueba se puede notar que las articulaciones en forma virtual cumplen con las coordenadas rectangulares del efector final dadas con un error máximo de 1,191%, lo cual permite validar el sistema

de control angular implementado para la resolución del problema cinemático inverso.

**Tabla 3.7: Prueba de validación del modelo cinemático inverso.**

COORDENADAS xyz (mm) (INVERSE KINEMATICS VI)				Aplicación con Math Script			ERROR RELATIVO PROMED. (%)
Coordenas Rectangulares			Configuración Angular (°)	Coordenas Rectangulares			
x	y	z	[θ1,θ2,θ3,θ4]	x	y	z	
316	0	214,5	[0°,90°,-90°,0°]	315	0	214	0.183
268	167	100	[32°,76°,-112°,36°]	267	166.561	100.395	0.344
301	-130	78	[-23°,66°,-108°,42°]	300	-127.298	77.0923	1.191
201	-98	262	[-26°,123°,-101°,-22°]	200	-97.7691	260.445	0.442
275	223	158	[39°,73°,-82°,0°]	275	222.291	158.431	0.197
312	0	0	[0°,40°,-107°,71°]	312	0	0.346098	0.000
150	-92	268	[-32°,138°,-104°,-35°]	149	-93.3551	268.113	0.727

### 3.3.2 PRUEBAS DEL SISTEMA DE CONTROL DE POSICIÓN

La resolución de los problemas cinemáticos suministran los valores angulares que cada una de las articulaciones del manipulador debe adoptar para su posicionamiento, en tanto que la apertura-cierre del gripper se expresa en milímetros.

Ahora para posicionar de forma correcta los actuadores encargados de mover las articulaciones del robot, se convierten los valores obtenidos por los modelos cinemáticos del manipulador en valores que la controladora de servomotores pueda interpretar.

Debido a la misma tolerancia, rangos de operación de ancho de pulso y rangos de banda muerta (DBW) de cada uno de los servomotores utilizados, se realiza de manera experimental una serie de mediciones para hallar los valores de controladora correspondientes a los valores angulares provistos por los modelos cinemáticos.

Para las mediciones se tuvo en cuenta la forma que facilita la visualización de los dos últimos bytes de la trama enviada a la tarjeta controladora mediante el “Comando 3” mostrada en la Figura 2.50 del apartado 2.4.2.

Es así que para establecer una posición se definió al penúltimo byte como dirección de movimiento del servomotor que puede tener dos posibles direcciones en rango decimal (0-1) y al último byte como la posición del movimiento en rango decimal (0-127), teniendo esto en cuenta se procede a realizar las mediciones.

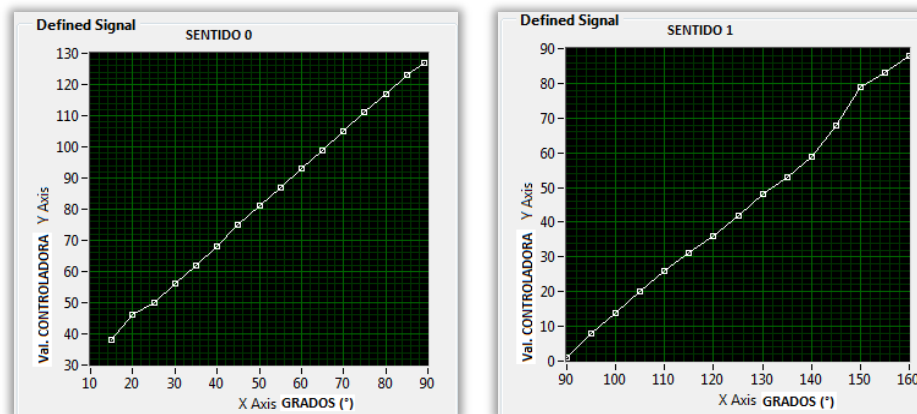
Las mediciones se realizan enviando señales seriales de control a la tarjeta controladora en forma de la secuencia de bytes utilizada en la Figura 2.30 con lo cual se permite posicionar el eje de cada actuador. Además se tiene muy en cuenta los límites de desplazamiento de cada uno de los servomotores indicados en la tabla 3.6. Para la medición del ángulo real del eje del servomotor, se utilizó círculos graduados y un transportador, que se coloca de tal forma que se pueda observar el ángulo en el que se encuentra cada uno de los ejes de los servomotores, véase la figura 3.10.



**Figura 3.10: Medición del Ángulo Real del eje de cada actuador.**

Se utiliza estos métodos de medición para una variación de cada cinco grados, con lo cual se obtienen suficientes datos y mediante la herramienta de interpolación lineal que incorpora el software LabVIEW se determinan las curvas **Grados-Valores de controladora**.

En la Figura 3.11 se presentan las respectivas curvas determinadas mediante interpolación lineal de los datos medidos (véase la tabla 3.8) para el servomotor del Hombro.



**Figura 3.11: Curvas Grados-Valores de controladora del hombro.**

**Tabla 3.8: Valores medidos de la articulación del hombro.**

<b>HOMBRO</b>			
<b>SENTIDO 0</b>		<b>SENTIDO 1</b>	
<b>GRADOS(°)</b>	<b>V. CONTROLADORA</b>	<b>GRADOS(°)</b>	<b>V. CONTROLADORA</b>
15	38	90	1
20	46	95	8
25	50	100	14
30	56	105	20
35	62	110	26
40	68	115	31
45	75	120	36
50	81	125	42
55	87	130	48
60	93	135	53
65	99	140	59
70	105	145	68
75	111	150	79
80	117	155	83
85	123	160	88
89	127		

Los valores medidos y sus curvas, correspondientes a los demás servomotores de cada una de las articulaciones del manipulador, se muestran en el anexo A.

Al analizar los valores medidos de controladora para cada servomotor y sus respectivas representaciones gráficas, se constata que los valores obtenidos no son completamente lineales como se esperaría, lo cual agrega error significativo a nuestro sistema de control de posición.

Además de ciertas restricciones mecánicas propias de la constitución del robot como el resorte mecánico, aspectos dimensionales y también la distribución del peso de las articulaciones, dificultan posicionar de manera exacta al efector final.

Todo descrito anteriormente se debe a la implementación del proyecto con un manipulador robótico didáctico que no ofrece las mismas características de precisión de movimientos y estabilidad que provee un robot industrial.

Para verificar el posicionamiento del manipulador sobre el área de trabajo, el usuario mediante el modo **COORDENADAS xyz (mm)** del HMI, define los valores de las coordenadas rectangulares hacia los cuales debe dirigirse el extremo del robot.

Para estimar la operación del manipulador, se han definido diez valores de coordenadas en el espacio (x,y,z), y se verifica que el extremo del robot se dirija hacia dichas coordenadas.

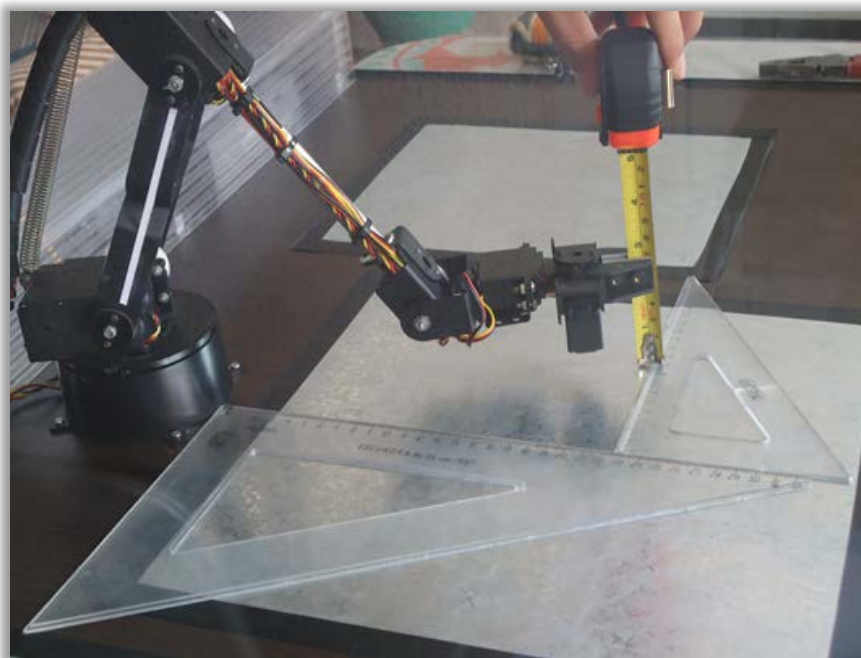
En la Tabla 3.9 se presentan las coordenadas de referencia definidas, y las coordenadas en las que se posiciona el extremo del robot.

En la Figura 3.12 se muestran las mediciones tomadas para la posición número dos de la prueba de posicionamiento del manipulador robótico.



**Tabla 3.9: Prueba de posicionamiento del manipulador robótico.**

Posición N°	Posiciones deseadas del efector final (mm)			Posiciones reales del extremo del efector final (mm)			Error Relativo Promedio (%)
	x	y	z	x	y	z	
1	32	-218	149	26	-232	142	9.96
2	263	106	60	275	102	65	5.56
3	146	156	300	135	150	290	4.90
4	124	-233	30	129	-229	24	8.58
5	124	-233	70	118	-225	68	3.71
6	106	308	0	113	305	7	2.53
7	106	308	170	110	299	162	3.80
8	132	-364	20	130	-365	22	3.93
9	197	357	140	195	355	105	8.86
10	316	0	214.5	313	1	214	0.39
							5.22



**Figura 3.12: Prueba de posicionamiento, posición N° 2 (263, 106, 60).**

### **3.4 PRUEBAS DE TELEOPERACIÓN A TRAVÉS DE CÁMARAS**

Se ha mencionado que las cámaras implementadas en el entorno de trabajo del robot no son usadas para realizar la teleoperación usando como realimentación visual las imágenes de video en el HMI, esta decisión se tomó debido varios factores que se analizaron en las pruebas.

Se adquirieron dos cámaras Web marca Omega y una cámara INS, con una resolución de 5 megapíxeles. Debido a la distancia entre las cámaras y el computador se adquirió tres extensiones USB de 1.8 metros (las más cortas disponibles) y ya que posiblemente el computador a usarse con el HMI posea sus puertos USB ocupados se adquirió un HUB USB estándar. A continuación se analizan las pruebas realizadas para la teleoperación mediante las cámaras Web.

#### **3.4.1 PRUEBAS DE FUNCIONAMIENTO DE CÁMARAS**

Primeramente se procede a determinar el correcto funcionamiento de las tres cámaras en el computador. Para ello se utilizan cada uno de los elementos adquiridos y se analiza los resultados.

##### **USANDO EXTENSIONES USB**

Se conectan las cámaras Web a las extensiones USB, y estas a su vez directamente a los puertos USB del computador. Luego utilizando la visualización de cámaras vía el HMI se observa que no se visualiza el video de todas las cámaras y cada vez que se ejecuta el programa se visualizan dos o tres cámaras de manera aleatoria. Se puede verificar que todas las cámaras han sido reconocidas usando el administrador de tareas de Windows o el software Measurement & Automation Explorer.

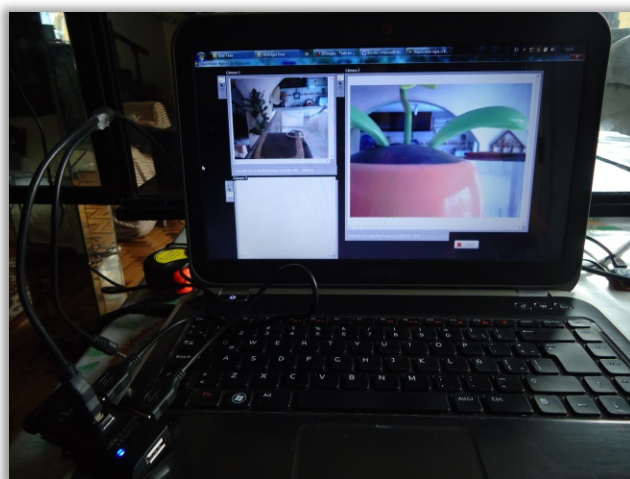
Esto se debe a que para utilizar extensiones USB en cámaras Web se recomienda de manera general y dependiendo del tipo de cámara que la distancia total del cable más la extensión no sobrepase los 3 metros, ya que si la distancia es muy grande el puerto USB no alcanza a suministrar la energía necesaria para que las cámaras funcionen correctamente. En nuestro caso, la distancia total es de 3.1 m que da como resultado el problema indicado.

Es por esto que el uso de extensiones USB resulta como uno de los inconvenientes que imposibilitan la teleoperación mediante cámaras.

### **USANDO EL HUB USB**

Debido a que es muy posible que no se disponga en el computador de los tres puertos USB necesarios para conectar las cámaras se optó por usar un HUB USB al que únicamente se conectarían las cámaras USB.

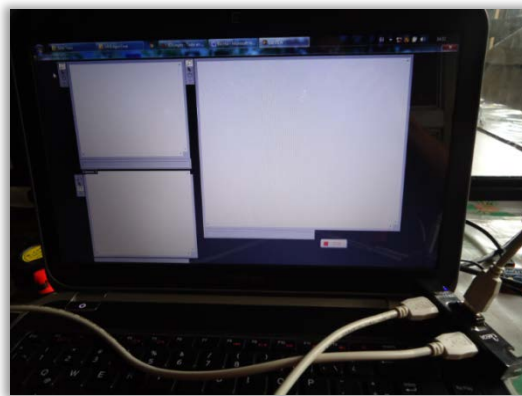
Se conectaron las tres cámaras USB directamente al HUB USB y este se conectó a un puerto USB del computador. Luego utilizando la visualización de cámaras vía el HMI se observa que únicamente se visualiza video de una o dos de las cámaras USB (Véase Figura 3.13).



**Figura 3.13: Visualización de video de las cámaras Web haciendo uso de un HUB USB.**

Se determinó que este problema se debe a que las cámaras Web usan una tasa de transferencia de datos muy elevada y constante que hace que no se puedan utilizar varias cámaras con el HUB USB ya que al final solo se transmitirían los datos de una sola de las cámaras, y pocas veces dependiendo de la cámara se recibía imágenes de dos cámaras. Lo que se traduce como: una Web Cam por puerto estándar USB del computador.

Está por demás acotar que si se utiliza el HUB USB y las extensiones USB el resultado es la nula recepción de video de las cámaras WEB en el HMI (Véase Figura 3.14).



**Figura 3.14: HMI sin video al usar el HUB y extensiones USB.**

Debido a que no se pueden usar HUB USB con las cámaras USB, es posible que se dé el inconveniente de falta de puertos USB en el computador con el HMI implementado.

### **3.4.2 PRUEBAS DE TELEOPERACIÓN A TRAVÉS DE VIDEO**

Está claro que los inconvenientes analizados en las pruebas anteriores son suficientes para imposibilitar la teleoperación a través de cámaras Web.

A pesar de esto, por motivos de análisis, las cámaras Web con las extensiones USB se conectaron directamente a los puertos USB 3.0 de un computador portátil. Debido a las mejores características de este puerto se puede suministrar la energía necesaria para que las cámaras funcionen con las extensiones. Gracias a ello se realizaron las pruebas de control del robot haciendo uso de realimentación visual únicamente con el video proporcionado por las cámaras, es decir que el operador controlará el robot haciendo uso del video de las cámaras sin mirar directamente al robot.

### **TELEOPERACIÓN CON EL GUANTE SENSORIZADO**

Para realizar las pruebas de teleoperación con el guante sensorizado haciendo uso de la realimentación visual a través de las cámaras web se utiliza la aplicación “Supervisión” en el HMI, esta aplicación muestra el video de las tres cámaras, posicionadas una para cada gradilla y una de enfoque total del área de trabajo (Véase Figura 3.15).



**Figura 3.15: Posición y visualización de las cámaras Web.**

Las pruebas simplemente consisten en realizar el procedimiento de transportar un tubo de ensayo de una gradilla a otra, mirando las imágenes en la aplicación de “Supervisión”. Al final dichas pruebas entregaron como resultado la existencia de dos grandes inconvenientes que provocan que el operador tenga evidentes dificultades para el

posicionamiento del efector final y realizar el correcto proceso de sujetar y colocar los tubos de ensayo.

**En primer lugar** el operador debe orientar la base del robot hacia la dirección que se encuentre el objeto a manipular, durante este proceso se utiliza en mayor grado las imágenes de video de la cámara superior, pero resulta difícil para el operador saber con seguridad el lugar exacto hacia donde se encuentra el robot ya que una imagen superior y dos dentro del área no son capaces de suministrar la suficiente información visual sobre orientación, en la Figura 3.16 se indica la falta de información de video para determinar la posición correcta de la base del robot.



**Figura 3.16: Video de cámaras con falta de información sobre orientación**

**En segundo lugar** una vez orientado el robot, se debe acercar la punta del efector final hacia el objeto a manipular, y posteriormente cuando se requiera colocar el objeto también se debe proceder a acercar el efector final. En este proceso, con las imágenes de video proporcionadas el operador tiende a confundirse creyendo que ya se ha llegado al lugar deseado, esto se debe a que el video proporcionado no entrega información completa para conocer la profundidad al posicionar la pinza del robot

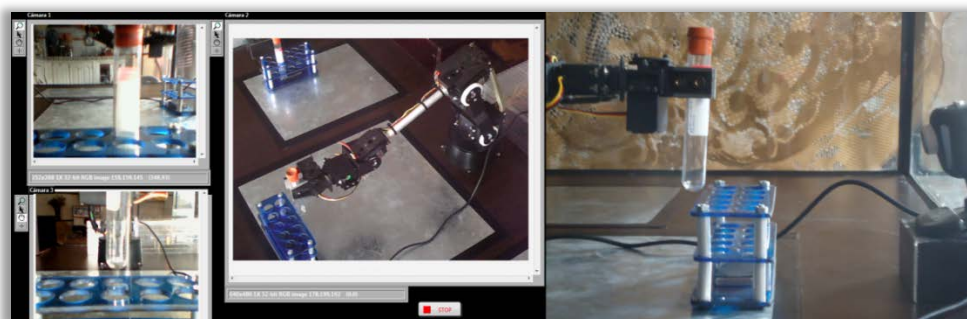


En la Figura 3.17, en el HMI en la cámara 1 se observa que la pinza ya ha llegado al tubo de ensayo pero la cámara 2 muestra sin exactitud que aún no se llega al objeto, la fotografía tomada desde el costado muestra de mejor manera que en realidad aún faltan varios milímetros para llegar al objeto.



**Figura 3.17: Falta de información sobre profundidad al sujetar el objeto.**

Además la falta de profundidad se repite a la hora de colocar los tubos de ensayo en las gradillas, haciendo que el operador crea que ya se encuentra sobre uno de los agujeros y posiblemente suelte el tubo con el material peligroso sobre el suelo. En la Figura 3.18 se puede observar que en el video de la cámara 3 se puede considerar que el tubo ya se encuentra sobre las gradillas, pero la fotografía revela que en realidad existe una distancia aun por recorrer.



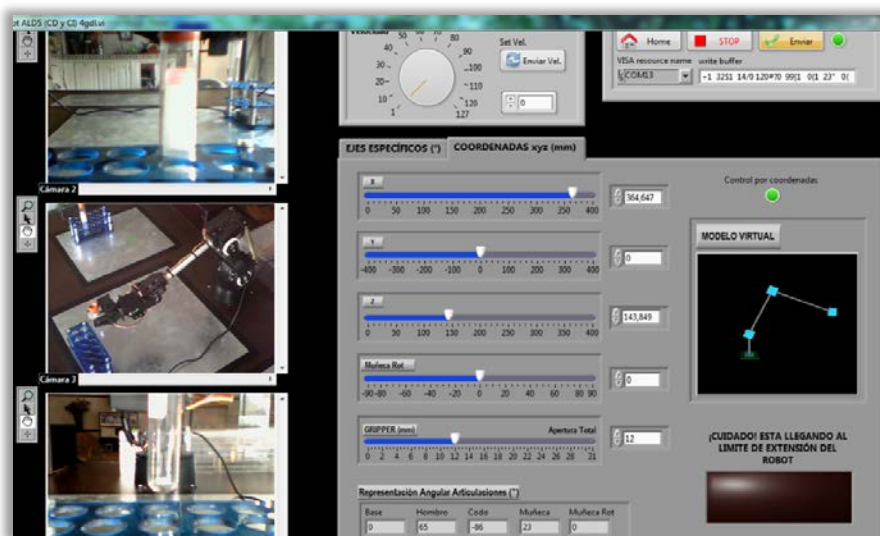
**Figura 3.18: Falta de información sobre profundidad al colocar el objeto.**

## TELEOPERACIÓN CON EL HMI

La aplicación para el control del HMI muestra también el video de las tres cámaras Web, las pruebas se realizaron controlando el robot desde el HMI y usando únicamente la realimentación visual proporcionada por las cámaras.

De igual manera se obtuvieron los mismos resultados que con las pruebas con el guante, y que evidenciaron los problemas indicados anteriormente, la única diferencia es que en este caso a la hora de teleoperar el operador puede tomarse el tiempo que quiera para realizar los movimientos, a diferencia que con el guante el operador estaba obligado a realizar las acciones el tiempo que su brazo pudiera soportar.

En la Figura 3.19, se evidencia nuevamente la falta de información de profundidad entregada por el video de las cámaras a la hora de colocar el tubo de ensayo en las gradillas.



**Figura 3.19: Teleoperación con el HMI haciendo uso de las cámaras web.**

Como se ha podido observar en estas pruebas, la teleoperación con realimentación visual con las cámaras Web es demasiado complicada y



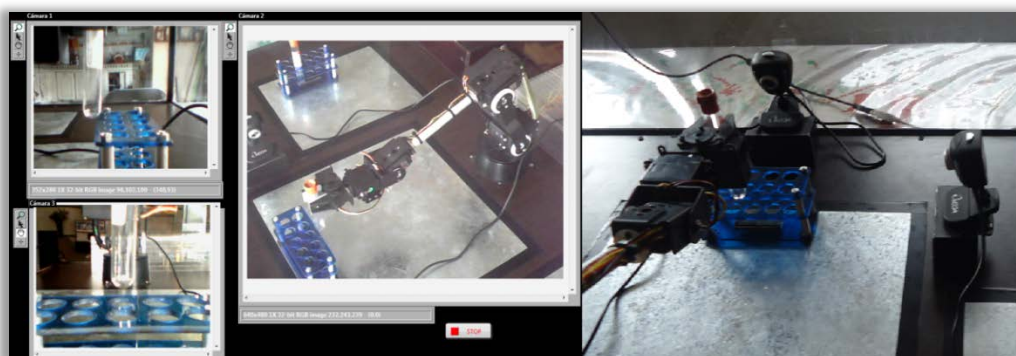
provoca mucha imprecisión al operador para realizar las tareas de manipulación de objetos. Es necesario para el operador conocer exactamente la ubicación de cada una de las articulaciones y del efector final del robot dentro del área de trabajo, para ello se necesita que obligadamente el operador tenga contacto visual con el robot.

Pero si se desease realizar la teleoperación únicamente con las cámaras, se necesita un mayor número de estas para conocer con exactitud la posición del robot.

Para el caso de este proyecto en particular se necesitarían tres cámaras por gradilla (un total de seis cámaras), la primera con vista aérea, la segunda con vista frontal y la tercera con vista lateral a una de las gradillas. Con esto se obtendría la información visual necesaria para realizar sin errores la operación de manipulación.

En la figura 3.20 se puede observar que las tres cámaras Web disponibles se colocan en la configuración adecuada para así posicionar el robot sin errores cerca de una de las gradillas.

El proyecto no hace uso de seis cámaras (tres por gradilla) debido a la falta de puertos USB y falta de espacio en la pantalla del HMI para visualización de video.



**Figura 3.20: Teleoperación con tres cámaras web por gradilla.**

### 3.5 ANÁLISIS TÉCNICO-ECONÓMINO

En la siguiente tabla se detalla la lista de materiales, elementos electrónicos y demás elementos utilizados en la implementación del presente proyecto, además se determina el costo total del mismo, en dólares americanos.

**Tabla 3.10: Costo de componentes para la implementación del proyecto.**

No	CANT.	DETALLE	VALOR UNITARIO (\$)	VALOR TOTAL (\$)
1	1	Brazo robótico Lynxmotion AL5D (incluido actuadores)	560,00	560,00
2	1	Fuente de Alimentación Triple (+5V +12V -12V)	65,00	65,00
3	1	Serial 16-Servo Controller Pololu	59,00	59,00
4	1	Estructura Mecánica (Incluido materiales)	200,11	200,11
5	1	Guante Black Diamond (Heavy Weight Black MD)	35,00	35,00
6	1	Codera marca Dalps	28,00	28,00
7	1	Control Nunchuk	19,00	19,00
8	3	Acelerómetro - MMA7361L	35,00	105,00
9	1	Acelerómetro - MMA7361 - 148861	25,00	25,00
10	1	Sensor Flex 4.5"	21,50	21,50
11	1	Sensor de Fuerza RED	9,90	9,90
12	3	XBEE 1MW CHIP ANTENNA-2.4GHZ-100M	36,00	108,00
13	1	XBEE EXPLORER USB	35,50	35,50
14	1	Cable mini USB	2,45	2,45
15	1	Batería de Litio 7.4V Turnigy	24,51	24,51
16	1	Cargador de Batería Turnigy	26,00	26,00
17	2	PIC18F452	9,25	18,50
18	1	Elementos Electrónicos	104,89	104,89
19	1	Elaboración de Tarjetas Electrónicas	108,00	108,00
20	1	Cámara WEB INS USB TIPO PINZA 5MPX	7,99	7,99
21	2	Cámara WEB OMEGA USB 5MPX	9,50	19,00
22	3	Cable de extensión USB OMEGA 6 pies	1,68	5,04
23	1	Mini HUB USB OMEGA 4 puertos con cable	4,50	4,50
24	1	Varios	80,00	80,00
<b>TOTAL:</b>				<b>1671,88</b>

El costo de materiales y elementos para la construcción del prototipo es de 1671,88 dólares americanos.

Para conocer el costo total del proyecto, se debe añadir el rubro de mano de obra de ingeniería y montaje del proyecto.

Para la estimación del costo de ingeniería y software se usa la regla de Cocomo. La misma que establece la siguiente expresión:

$$\text{Costo (USD)} = K * \text{No horas persona} \quad (\text{Ec. 3.2})$$

Dónde:

K = Valor hora profesional en USD, estimado en 15.00 USD para rubro de ingeniería.

Para cuantificar el costo del montaje e implementación del proyecto, se utiliza la misma ecuación, pero asignando a K un valor de 2.00 USD.

En la tabla 3.11, se detalla el coste de personal del proyecto, considerando dos personas con una carga laboral neta de 30 días y 5 horas diarias para el rubro de ingeniería.

Para el montaje de igual manera se contempla el trabajo de dos personas, con una carga laboral de 15 días y 5 horas diarias cada uno.

**Tabla 3.11: Costo de diseño e implementación del proyecto.**

ÍTEM	DESCRIPCIÓN	No. Horas Hombre	K (USD)	SUBTOTAL (USD)
1	Ingeniería del proyecto	300	15	4500
2	Montaje del proyecto	150	2	300
			<b>TOTAL:</b>	<b>4800</b>

Por lo tanto el costo total de inversión del prototipo es de 6471,88 dólares americanos.

## **CAPÍTULO IV**

### **CONCLUSIONES Y RECOMENDACIONES**

#### **4.1 CONCLUSIONES**

- Se ha realizado de manera satisfactoria la implementación de un sistema de teleoperación para un robot manipulador mediante un guante sensorizado; capaz de interpretar los movimientos del brazo humano y traducirlos a movimientos del robot y de esta manera realizar tareas que impliquen coger, manipular y colocar objetos dentro de un área de trabajo permitida.
- El transporte de tubos de ensayo con material peligroso mediante teleoperación con el guante sensorizado se realiza de forma normal y segura, sin poner en riesgo la integridad de algún elemento del sistema.
- Si la manipulación de los tubos ensayo mediante el guante sensorizado se realiza de manera rápida se pierde mucha precisión en los movimientos del robot y el operador puede llegar a perder el control del manipulador haciendo que la tarea realizada tarde más tiempo del que hubiese tomado si se hubiese realizado la operación con lentitud y precisión.
- De manera general la dificultad de teleoperar el robot con el guante sensorizado para manipular objetos depende de la capacidad del operador para manejar el sistema, si el operador no conoce completamente la forma de uso y no ha practicado los movimientos tendrá dificultades para realizar tareas de manipulación.

- El diseño del guante sensorizado no es complejo y utiliza elementos previamente fabricados para otros fines, pero satisface la necesidad de posicionar los sensores sobre el brazo humano, además sus características físicas hacen que el guante sensorizado no presente ninguna molestia al momento de teleoperar el brazo robótico.
- Se instaló un detector de fuerza en la parte lateral interna de la pinza del robot con el fin de obtener la información necesaria de la fuerza de cierre, para que el efector final del robot pueda sostener y manipular diferentes objetos sin deformarlos o en su defecto dañar el actuador de la pinza al forzar su funcionamiento.
- Debido al requerimiento de alto consumo de potencia de los actuadores del robot y además evitando posibles interferencias con la etapa de control, se ha dispuesto el uso de una fuente de alimentación independiente para la etapa de potencia del robot.
- No es necesario un control en lazo cerrado para el posicionamiento del robot ya que los servomotores se colocan siempre en el lugar deseado dependiendo de las acciones del operador y el correcto posicionamiento de los servomotores depende únicamente del algoritmo de control que interpreta las acciones del operador.
- Retroalimentar una señal que indique la posición de los actuadores de un robot únicamente es necesario si dichos actuadores desconocen la posición a la que deben llegar. Este no es el caso ya que los servomotores internamente poseen su control de posición.
- Se desarrolló de manera adicional una interfaz amigable al usuario en la plataforma de programación visual gráfica LabVIEW, que también

permite teleoperar el manipulador robótico mediante la aplicación HMI ejecutada en cualquier computador que tenga instalado el software y los toolkits necesarios.

- El HMI implementado permite el control del brazo robótico con un nivel de precisión satisfactorio y emula en gran parte la capacidad de posicionamiento de los sistemas de control de los robots industriales.
- Al ser el brazo robótico un sistema didáctico, este presenta varias condiciones que influyen en el error de posicionamiento mediante el sistema de control del HMI, entre las que están de manera general: imperfecciones en la estructura mecánica, diseño mecánico impreciso, servomotores de modelismo.
- Los servomotores de modelismo utilizados en el robot no tienen una relación exactamente lineal entre la señal de control aplicada y la respuesta de posición angular que adopta su eje. A razón de que hay intervalos de señal de control que no producen movimiento al eje del actuador, es decir, el circuito de control interno de los servomotores posee baja sensibilidad y también la misma caja reductora no permite mover el eje en incrementos más pequeños de los que permita el juego del sistema de engranajes.
- Las cámaras web incorporadas al sistema de teleoperación son usadas únicamente para supervisión ya que el operador deberá observar directamente la tarea que realice el robot manipulador.
- Las cámaras web implementadas no posibilitan al operador a usar solo el video de las mismas para teleoperar el robot debido a las deficiencias encontradas, pero hacen referencia para trabajos futuros

a la inclusión de más y mejores cámaras para el control del robot sin visualización directa de la tarea por parte del operador.

## **4.2 RECOMENDACIONES**

- El usuario que vaya a utilizar el módulo de teleoperación debe leer y comprender lo expuesto en el manual para de esta manera realizar el adecuado uso del guante sensorizado y HMI, y así garantizar el correcto funcionamiento del sistema de teleoperación.
- Realizar la teleoperación con el guante sensorizado de manera lenta y cautelosa para evitar perder el control y ganar precisión en los movimientos del robot manipulador.
- Para futuros proyectos se recomienda que el guante a implementarse en lo posible de fabricación propia preferiblemente de una pieza y diseñado tomando en cuenta las necesidades planteadas para así tener una mejor forma de posicionar los sensores sobre el guante.
- No excederse en el tiempo de operación de los servomotores al teleoperar el brazo robótico tanto con el uso del guante sensorizado como con el HMI, ya que los servomotores son de modelismo y no están diseñados para trabajo en periodos largos de tiempo; los mismos pueden sobrecalentarse y sufrir daño permanente.
- Recordar que el robot es didáctico, por ende no mantener por mucho tiempo al robot en posiciones críticas en el que el peso del mismo pueda afectar demasiado la operación de los servomotores; es decir

observar que el robot no opere de manera forzada con la emisión de ruidos y vibración.

- Si en futuros proyectos se pretende teleoperar un robot con motores de corriente continua convencionales se necesita obligadamente un control en lazo cerrado para el posicionamiento del robot y para ello se recomienda que para obtener la señal realimentada se usen potenciómetros lineales en las articulaciones del robot y no acelerómetros para evitar problemas con la lectura de señales.
- Para ampliaciones del proyecto se recomienda implementar un mejor método de realimentación de información visual hacia usuario en la aplicación HMI, ya que solo las tres cámaras hacen imposible la teleoperación del robot sin línea de vista directa al módulo. Con esta mejora se podría utilizar módulos de comunicación inalámbricos de mayor alcance que permitirían teleoperar al manipulador robótico sin línea de vista directa incluso con el guante sensorizado.
- Si en futuro se desea cambiar o agregar funciones al sistema que implique nueva información para el microcontrolador de la estación remota se recomienda reemplazar el microcontrolador actual por uno de mayor memoria RAM.



## BIBLIOGRAFÍA

- Acuña, J. (2012). *Scribd*. Recuperado el 01 de 10 de 2012, de <http://es.scribd.com/doc/50785531/38370133-Biopotenciales-1>
- Albornoz Carrasco, J., & Lagos Norambuena, P. (2008). *GUIA DE INICIACION EN LABVIEW 7.1*. Recuperado el 5 de Agosto de 2013, de [http://www.inele.ufro.cl/apuntes/LabView/Sesiones\\_Oficial\\_pdf/Guia\\_de\\_Iniciacion\\_en\\_LabVIEW.pdf](http://www.inele.ufro.cl/apuntes/LabView/Sesiones_Oficial_pdf/Guia_de_Iniciacion_en_LabVIEW.pdf)
- Alencastre, M., Muñoz, L., & Rudomon, I. (2003). Teleoperating Robots in Multiuser Virtual Environments. *Proceedings of the Fourth Mexican International Conference on Computer Science*. Mexico.
- Angulo Martínez, I., & Angulo Usategui, J. M. (2003). *MICROCONTROLADORES PIC. Diseño práctico de aplicaciones*. Madrid: McGRAW-HILL.
- Barrientos, A., Peñin, L., Balaguer, C., & Aracil, R. (2007). *Fundamentos de Robótica*. Madrid: McGraw-Hill.
- Blake, R. (2004). *Sistemas electrónicos de comunicaciones*. México: Thomson.
- Brady, M., & Paul, R. (1984). Robotic Research. *The First International Symposium*. Cambridge MA: The MIT Press.
- Carrión, L., & Ochoa, D. (2009). *Data Lights*. Recuperado el 01 de 10 de 2012, de [http://www.datalights.com.ec/site2/images/stories/robotica/nap/nap\\_fsr.pdf](http://www.datalights.com.ec/site2/images/stories/robotica/nap/nap_fsr.pdf)
- Cortes-Monroy, P. (07 de 07 de 2009). *FayerWayer*. Recuperado el 01 de 10 de 2012, de *AcceleGlove: Guante programable con*

acelerómetro: <http://www.fayerwayer.com/2009/07/acceleglove-guante-programable-con-acelerometro/>

freescale. (2005). *freescale*. Recuperado el 27 de 09 de 2012, de [http://cache.freescale.com/files/sensors/doc/fact\\_sheet/ACCELERS\\_ENSRFS.pdf?fsrch=1&sr=1](http://cache.freescale.com/files/sensors/doc/fact_sheet/ACCELERS_ENSRFS.pdf?fsrch=1&sr=1)

Frenzel, L. E. (2003). *Sistemas electrónicos de comunicaciones*. México: Alfaomega.

Galicia Balcázar, G. (sf). *CURSO BÁSICO DE MICROCONTROLADORES PIC*. Recuperado el 25 de 09 de 2012, de <http://www.slideshare.net/guestd2dada1/pic-16-f84-a-2899778>

Gómez de Gabriel, J. M., Ollero Baturone, A., & García Cerezo, A. J. (2006). *Teleoperación y Telerrobótica*. Madrid: Pearson Educación S.A.

González, V. R. (2002). *Centro de Formación del Profesorado e Innovación Educativa de Valladolid*. Recuperado el 24 de 09 de 2012, de Introducción a la Robótica: [http://cfievalladolid2.net/tecno/cyr\\_01/robotica/intro.htm](http://cfievalladolid2.net/tecno/cyr_01/robotica/intro.htm)

González, V. R. (2002). *Centro de Formación del Profesorado e Innovación Educativa de Valladolid*. Recuperado el 27 de 09 de 2012, de Clasificación del robot industrial: [http://cfievalladolid2.net/tecno/cyr\\_01/robotica/industrial.htm#clasificacion](http://cfievalladolid2.net/tecno/cyr_01/robotica/industrial.htm#clasificacion)

González, V. R. (2002). *Centro de Formación del Profesorado e Innovación Educativa de Valladolid*. Recuperado el 28 de 09 de 2012, de Estructura de un robot industrial: [http://cfievalladolid2.net/tecno/cyr\\_01/robotica/sistema/morfologia.htm](http://cfievalladolid2.net/tecno/cyr_01/robotica/sistema/morfologia.htm)

- Groves, P. D. (2007). *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*. Londres: Artech House.
- Ibarra, J. L., Ramírez, M. d., & Taboada, V. (25 de 08 de 2006). *e-Journal. Revistas especializadas de prestigio en formato electrónico*. Recuperado el 27 de 09 de 2012, de Sensor Fotoeléctrico Aplicado al Movimiento de los Dedos de laa Manos: <http://www.ejournal.unam.mx/cys/vol10-01/CYS10105.pdf>
- Jones, J. L., & Flynn, A. M. (1993). *Mobile Robots: Inspirations to implementation*. A K Peters Ltd.
- Khwaja, H. J. (2013). *Santa Clara University*. Recuperado el 01 de 05 de 2013, de <http://mech207.engr.scu.edu/SensorPresentations/Jan - Flex Sensor Combined.pdf>
- Lajara Vizcaíno, J. R., & Pelegrí Sebastiá, J. (2011). *LabVIEW Entorno gráfico de programación* (Segunda ed.). España: marcombo, S.A. ediciones técnica.
- Mandado, E., Menéndez, L., Ferreira, L., & Matos, E. (2007). *Microcontroladores PIC. Sistema integrado para el autoaprendizaje*. España: Editorial Marcombo, Ediciones Técnicas.
- Mendoza.edu. (sf). *Mundo Joven*. Recuperado el 07 de Septiembre de 2012, de Zona Robótica: [http://www.roboticajoven.mendoza.edu.ar/rob\\_mec7.htm#brazo](http://www.roboticajoven.mendoza.edu.ar/rob_mec7.htm#brazo)
- ni.com. (2013). *Overview of the LabVIEW Robotics Module*. Recuperado el 4 de Agosto de 2013, de <http://www.ni.com/white-paper/11564/en>
- Önal, Çağdaş, & Denizel. (2005). *Bilateral Control: A Sliding Mode Control Approach*. Istanbul, Turkey: Sabancı University.
- Ortigueira, C., Reigosa, M., Rodríguez, M., Santamaría, C., & Veiga, J. (sf). *Sistemas Adaptativos y Bioinspirados en Inteligencia Artificial*.

Recuperado el 01 de 10 de 2012, de Realidad Virtual:  
<http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/3D/Realidad%20Virtual/web/dispositivos/guantes.html>

Oyarce, A. (2009). *xbee.cl, olimex.cl*. Recuperado el 22 de 12 de 2012, de  
[http://www.olimex.cl/pdf/Wireless/ZigBee/XBee-Guia\\_Usuario.pdf](http://www.olimex.cl/pdf/Wireless/ZigBee/XBee-Guia_Usuario.pdf)

Roboticspot. (2011). Recuperado el 07 de Septiembre de 2012, de  
<http://www.roboticspot.com/robotica.php>

Roldán Martínez, D. (2005). *Comunicaciones inalámbricas. Un enfoque aplicado*. México: Alfaomega.

Sabanovic, A., & Ohnishi, K. (2011). *Motion Control Systems*. Singapore: John Wiley & Sons.

Sacco, M. (2011). *NeoTeo - Revista de tecnología*. Recuperado el 27 de 09 de 2012, de <http://www.neoteo.com/-21690-sensores-inerciales-el-mundo-en-movimiento>

Wikipedia. (2012). *Wikipedia*. Recuperado el 2012 de 10 de 18, de  
[http://es.wikipedia.org/wiki/Material\\_peligroso](http://es.wikipedia.org/wiki/Material_peligroso)

Wikipedia. (2012). Recuperado el 27 de 09 de 2012, de  
[http://es.wikipedia.org/wiki/Red\\_inal%C3%A1mbrica](http://es.wikipedia.org/wiki/Red_inal%C3%A1mbrica)

## GLOSARIO

### A

**AFNOR:** (Asociación Francesa de Normalización). Es la organización nacional francesa para la estandarización y miembro de la Organización Internacional para la Estandarización.

**ASCII:** Acrónimo inglés de American Standard Code for Information Interchange - Código Estándar Estadounidense para el Intercambio de Información. Es un código de caracteres basado en el alfabeto latino, tal como se usa en inglés moderno y en otras lenguas occidentales.

**ASIC:** Un Circuito Integrado para Aplicaciones Específicas (o ASIC, por sus siglas en inglés) es un circuito integrado hecho a la medida para un uso en particular, en vez de ser concebido para propósitos de uso general.

### B

**Baudio:** Es una unidad de medida, usada en telecomunicaciones, que representa el número de símbolos por segundo en un medio de transmisión digital. Cada símbolo puede codificar 1 o más bits dependiendo del esquema de modulación.

**Bielasticidad:** Característica de aquellos tejidos cuyo sentido de elasticidad se localiza en ambos sentidos o longitudinal o transversal. Ello proporciona un muy elevado nivel de comodidad en el usuario de la prenda.

**Buffer:** En informática, un buffer de datos es una ubicación de la memoria en un Disco o en un instrumento digital reservada para el almacenamiento temporal de información digital, mientras que está esperando ser procesada.

## C

**Cinestesia:** Rama de la ciencia que estudia el movimiento humano. Proviene del griego κίνησις, "movimiento", y αἴσθησις /aísthesis/ "sensación", es decir, etimológicamente "sensación o percepción del movimiento".

**CMOS:** Complementary metal-oxide-semiconductor. Es una de las familias lógicas empleadas en la fabricación de circuitos integrados. Su principal característica consiste en la utilización conjunta de transistores de tipo pMOS y tipo nMOS configurados de tal forma que, en estado de reposo, el consumo de energía es únicamente el debido a las corrientes parásitas.

**Convergencia:** Unión de dos o más cosas que confluyen en un mismo punto.

## D

**DBW:** Dead Band Width. A veces llamado una zona neutral, es un intervalo de una señal de dominio o banda donde no se produce ninguna acción (el sistema es muerto). En este rango el servomotor no responde al cambio de posición y se configura en microsegundos, "us".

**D-H:** Denavit-Hartenberg, parámetros que permiten la obtención del modelo cinemático de un robot.

## E

**Elementos resistivos de carbono:** Son elementos comúnmente usados para la fabricación de resistencias de todo tipo.

## G

**GDL:** (Grado de libertad). Se refiere a la capacidad de movimiento en un espacio tridimensional, equivale a decir número y tipo de movimientos del manipulador.

## H

**Háptico:** No aparece en el diccionario de la Real Academia Española y proviene del griego háptō que significa relativo al tacto.

## I

**Interpolación:** Obtención de nuevos puntos partiendo del conocimiento de un conjunto discreto de puntos. Es frecuente disponer de un cierto número de puntos obtenidos por muestreo o a partir de un experimento y pretender construir una función que los ajuste.

**ISO:** (Organización Internacional de Estándares). Es un organismo encargado de promover el desarrollo de normas internacionales de fabricación.

## K

**KCP:** NAVEGADOR KCP (KUKA CONTROL PANEL). Es la unidad manual de programación del sistema de robots de fabricantes de robots KUKA.

## L

**LCD:** (Liquid Crystal Display) Pantalla de cristal líquido formada por un número de píxeles en color o monocromos colocados delante de una fuente de luz o reflectora.

**Local:** Local es todo lo que se refiere o es relativo a un Lugar determinado.

## M

**MEMS:** Sistemas Microelectromecánicos (Microelectromechanical Systems, MEMS) se refieren a la tecnología electromecánica, micrométrica y sus productos, y a escalas relativamente más pequeñas (escala nanométrica).

## N

**Normal:** Perpendicular a otra recta o plano tangentes.

**Nunchuk:** Es una expansión para el mando inalámbrico de la consola de videojuegos Wii, tiene forma ovalada y posee dos botones y un stick análogo.

## O

**Ortogonal:** Adjetivo que se emplea para nombrar a aquello que se encuentra en un ángulo de  $90^\circ$ .

**Ortonormal:** Un conjunto de vectores es ortonormal si es a la vez un conjunto ortogonal y la norma de cada uno de sus vectores es igual a 1.

**OSI:** (Open System Interconnection) Es el modelo de red descriptivo, que fue creado por la Organización Internacional para la Estandarización. Es un marco de referencia para la definición de arquitecturas en la interconexión de los sistemas de comunicaciones.

## P

**Paletizado:** Es la acción y efecto de disponer mercancía sobre un palé para su almacenaje y transporte. Las cargas se paletizan para conseguir uniformidad y facilidad de manipulación; así se ahorra espacio y se rentabiliza el tiempo de carga, descarga y manipulación.



**Perfil Bajo:** En el las áreas de diseño y fabricación, es un término usado para dar a conocer que el elemento en cuestión tiene características físicas externas que lo hacen sencillo, imperceptible o discreto.

**Polivalente:** Que posee varios valores o que tiene varias aplicaciones, usos, etc.

**Pulsador de hombre-muerto:** Interruptor redundante de seguridad, se lo utiliza en caso de ocurrir algún inconveniente que pueda afectar la integridad del robot o de la operación de transporte del material, únicamente se deja de presionar el botón y el robot automáticamente deja de moverse.

## R

**R.U.R.:** (Robots Universales de Rossum), obra teatral de ciencia ficción conocida por contener la primera aparición del término «robot». Escrita por el checo Karel Čapek en 1920.

**Ratiométrico:** La ratiometricidad simplemente hace referencia a la propiedad por la que el offset de la tensión de salida y la sensibilidad están escalados con la tensión de la fuente de alimentación. Es decir, si la tensión de alimentación aumenta, la sensibilidad y el offset aumentarán linealmente, mientras que si la tensión de alimentación disminuye, ambos disminuirán linealmente.

**Redes Mesh:** O redes en malla, es una topología de red en la que cada nodo está conectado a todos los nodos. De esta manera es posible llevar los mensajes de un nodo a otro por distintos caminos. Si la red de malla está completamente conectada, no puede existir absolutamente ninguna interrupción en las comunicaciones.

**Remoto:** Permite referirse a aquello que se encuentra a una cierta distancia, retirado o alejado.

**Repetibilidad:** Capacidad de reproducción de las mediciones realizadas con un dispositivo al medir repetidamente valores idénticos de la variable en las mismas condiciones de servicio y en el mismo sentido de variación. Se considera en general su valor máximo y se expresa en porcentaje del alcance (span).

**Resolución:** Cantidad más pequeña de incremento posible de determinar, siendo por tanto la cantidad de medida más pequeña a detectar.

**RF:** El término radiofrecuencia, también denominado espectro de radiofrecuencia o RF, se aplica a la porción menos energética del espectro electromagnético, situada entre unos 3 kHz y unos 300 GHz. Las ondas electromagnéticas de esta región del espectro, se pueden transmitir aplicando la corriente alterna originada en un generador a una antena.

**RIA:** (Robot Institute of America), actualmente Robotic Industries Association.

## S

**Sensibilidad:** Es la razón de cambio de la salida frente a cambios en la entrada. También es definida como la salida obtenida por unidad de entrada.

**SSC:** (Serial Servo Controller). Controlador serie de servomotores.

**Sustrato:** El sustrato es el soporte donde se monta el circuito electrónico, compuesto por conductores, dieléctricos, y componentes integrados en el sustrato, o montados. La elección del sustrato depende de las características térmicas, mecánicas y eléctricas del circuito.

## T

**Teleoperación:** Conjunto de tecnologías que comprenden la operación o gobierno a distancia de un dispositivo por un ser humano.

**TTL:** Es la sigla en inglés de transistor-transistor logic, es decir, «lógica transistor a transistor». Es una familia lógica o lo que es lo mismo, una tecnología de construcción de circuitos electrónicos digitales. En los componentes fabricados con tecnología TTL los elementos de entrada y salida del dispositivo son transistores bipolares.

## **U**

**UART:** Son las siglas de "Universal Asynchronous Receiver-Transmitter" (en español, Transmisor-Receptor Asíncrono Universal). Éste controla los puertos y dispositivos serie. Se encuentra integrado en la placa base o en la tarjeta adaptadora del dispositivo.

## **V**

**VI:** (Virtual Instrument) Programa desarrollado bajo la plataforma LabVIEW. Puede ser usado como un subVI o como una función más dentro de un programa.

# **ANEXOS**

**Elaborado por:**

---

Edison Fernando Herrera Núñez   Wellington Oswaldo Zambrano Chávez

**Aprobado por:**

---

Ing. Fausto Acuña

DIRECTOR DE LA CARRERA DE INGENIERÍA MECATRÓNICA

**Certificado por:**

---

Dr. Freddy Jaramillo Checa

SECRETARIO ACADÉMICO