



# ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y  
TELECOMUNICACIONES

TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL  
TÍTULO DE INGENIERO EN ELECTRÓNICA Y  
TELECOMUNICACIONES

TEMA: DISEÑO E IMPLEMENTACIÓN DE UN GUANTE  
ELECTRÓNICO ORIENTADO A CONTROLAR LOS  
MOVIMIENTOS DE UN DRON

AUTOR: CONTRERAS PARREÑO, RAMIRO TERRY

DIRECTOR: SÁENZ ENDERICA, FABIÁN GUSTAVO

SANGOLQUÍ

2016



# ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA  
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

## **CERTIFICACIÓN**

Certifico que el trabajo de titulación, **“DISEÑO E IMPLEMENTACIÓN DE UN GUANTE ELECTRÓNICO ORIENTADO A CONTROLAR LOS MOVIMIENTOS DE UN DRON”**, realizado por el señor **RAMIRO TERRY CONTRERAS PARREÑO**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de las Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar al señor **RAMIRO TERRY CONTRERAS PARREÑO** para que lo sustente públicamente.

**Sangolquí, 13 de julio de 2016**

---

**MSC. ING. FABIÁN GUSTAVO SÁENZ ENDERICA  
DIRECTOR**



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

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA  
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

**AUTORÍA DE RESPONSABILIDAD**

Yo, **RAMIRO TERRY CONTRERAS PARREÑO**, con cédula de identidad N° 1711887875, declaro que este trabajo de titulación **“DISEÑO E IMPLEMENTACIÓN DE UN GUANTE ELECTRÓNICO ORIENTADO A CONTROLAR LOS MOVIMIENTOS DE UN DRON”** ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

**Sangolquí, 13 de julio de 2016**

**RAMIRO TERRY CONTRERAS PARREÑO**

C.C.: 1711887875



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

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA  
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

**AUTORIZACIÓN**

Yo, **RAMIRO TERRY CONTRERAS PARREÑO**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de mi titulación **“DISEÑO E IMPLEMENTACIÓN DE UN GUANTE ELECTRÓNICO ORIENTADO A CONTROLAR LOS MOVIMIENTOS DE UN DRON”** cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

**Sangolquí, 13 de julio de 2016**

---

**RAMIRO TERRY CONTRERAS PARREÑO**

**C.C.: 1711887875**

## DEDICATORIA

*En mi criterio, el presente proyecto no es solamente un requisito de graduación sino que representa un largo periodo de estudio, incontables horas y noches largas de trabajo realizado en una carrera de ingeniería que la palabra "exigente" no es suficiente para definirla.*

*Este mismo documento simboliza esfuerzo, capacidad, perseverancia, mucha paciencia y entereza de mi parte, tal cual nos enseñaron en Electrónica... "esta no es una carrera de velocidad sino de resistencia".*

*Entonces por la razón de que esta investigación refleja también el apoyo incondicional de varias personas relacionadas directa e indirectamente conmigo y porque sé que mis padres *Mónica* y *Terry* también anhelan que este logro cumplido sirva de motivación, inspiración, empuje, como un símbolo de que sí se pueden alcanzar los objetivos y metas... se lo dedico a las personas que crecieron conmigo y que más quiero de corazón que triunfen en este mundo:*

*A ustedes, mis hermanos: *Carlos Alberto* y *José Luis*.*

*Porque confío y tengo plena fe en ustedes, porque todo lo que he hecho en mi vida ha sido y será para apoyarles a vencer todas las adversidades... porque me han hecho feliz desde que tengo uso de razón porque estoy seguro de que nunca bajaremos los brazos y juntos lucharemos hasta lograr todos nuestros sueños como ahora *Yo* consigo uno de ellos, el inicio de lo que sé que será el éxito.*

*De su hermano: *Terry jr.**

## AGRADECIMIENTO

Al fin se termina un extenso capítulo de mi vida en el que ha habido muchos aciertos y desaciertos. Siento que he tenido algunos logros importantes en mi vida hasta el momento pero pienso que el hecho de terminar esta carrera es un logro de toda mi familia. Estaba ansioso de escribir estos pequeños párrafos de agradecimiento que son inéditos y que tratan de explicar lo que siento y vivo con cada persona. Les agradezco infinitamente, hoy estoy muy feliz y sé que todos dirán con mucho cariño: “Ya era hora, ya queríamos baile” 😊.

**A mi madre**, las letras son un instrumento casi nulo para expresar mi infinito amor hacia usted. Si existe alguien en este mundo a quien le debo todo lo que soy y lo que he logrado humildemente en mi vida ha sido usted, no necesariamente se necesitan montañas de dinero para ser amado y ser la motivación dentro de mi pecho que me empuja a alcanzar las metas. Sé que esto para nosotros no es sólo una graduación sino que es el símbolo de una *victoria* que no ha sido fácil pero que hemos alcanzado juntos. Gracias infinitamente: “Usted me da el amor que le da vida a mi alma”.

**A mi padre**, ¡la pelota está en su cancha, el partido recién comienza, no se puede alzar la copa sin jugar el campeonato! Así pues comienza la siguiente etapa en mi vida, solo el tiempo dirá si lograré el campeonato o como se dice ¡cruzar a la otra orilla del río! Si alguna vez no he cumplido tus expectativas estoy seguro que en el futuro sabré llegar al *triunfo en la vida* que es el fin tan anhelado. Agradecerte por todo el apoyo y todas las ideas brillantes que me transmites a diario y decirte que sin ti mis objetivos serían aún inciertos: “Yo soy como un lienzo en el que tu pintas el camino al éxito”.

**A mis tías Cecilia y Adriana**, como alguna vez les he dicho, ustedes son las *segundas madres* que hemos tenido. Han sabido transmitirnos el legado de los abuelitos crecimos con todo el cariño y consejos que incondicionalmente nos han sabido brindar, “Son más que sólo un granito de arena, son muchísimo más” ¡Gracias de corazón!

**A Gladis**, un refrán: ¡Con pan y vino, se anda el camino! Has sabido demostrar con hechos el *buen hacer* por todos nosotros. Eres parte fundamental del proyecto familiar y espejo de éxito en el cual reflejarse ¡Gracias por acompañarle a mi padre!

**A mi tía Magaly**, quiero agradecerte por los momentos que hemos compartido, los consejos que nos has dado y las historias de la familia que nos has contado. Para mí eres el símbolo de

*lucha y perseverancia* a seguir. Tú lograste lo que muchos han querido pero pocos han obtenido, el destino nos unirá más sin duda. ¡Un gran abrazo!

A **William Santy**, ha sido una *muy importante* compañía para mi mamá, le agradezco sobremedida toda la buena voluntad que tiene para con nosotros. Ahora que forma parte de la familia sin duda habrá nuevos proyectos para cumplir. ¡Muchas gracias!

A **Soledad Almeida**, en la parte más crítica de la carrera, cuando no había más salidas a aplicar has sabido colaborarme como *ángel guardián*. ¡Gracias totales!

Al **Ing. Darwin Alulema**, existen momentos en la carrera en los que se necesitan personas de buen corazón y buena voluntad que apoyen sin condiciones, *muchas gracias* Inge por sus buenas labores en pro del esmero e ímpetu demostrados.

A **Edison Naranjo**, una persona con muchas virtudes que supo ser *mi compañero de clases y mi amigo*, muchas gracias por todo el apoyo y todos los consejos profesora. Sin duda, ahora poder llamarte colega es una gran alegría que estábamos esperando con ansias amigo mío.

A **David Mendoza, Gabriel Torres y Diego Pabón**, una de las cosas buenas que te deja cada etapa en la vida son los *verdaderos amigos* que llegas a tener. Nunca olvidaré cuando analizábamos el *"jitter, packet loss"* en los proyectos de comunicaciones inalámbricas haciendo ingeniería de tráfico y decíamos *"ya está, ya está"* en *desmedro* de largas noches sin dormir pero a veces los datos que obteníamos eran *"falsos"*. Luego terminábamos cansados al frente de la Espe comiendo unos *"ce-vichos"*. Gracias por todo, sólo me queda decirles la frase célebre: *"Les quiero jebas"* por siempre seremos el invencible <<GRUPO 2>>.

A **los MDS**, en el honorable Colegio Montúfar aprendimos a ser señores a carta cabal, han estado para ser más que unos amigos mis brothers. MH por siempre ¡Larga vida a los MDS!

A **Karina Flores**, tu apoyo *in situ*, tu compañía, tus ganas de salir adelante y tu manera de ser me alentaron en gran parte de mis estudios, te quiero muchísimo ¡Gracias infinitas!

A **Priscila Marcillo**, la *madrina oficial* de Grupo 2, los momentos vividos, has estado en las buenas y en las malas más, gracias por estar conmigo ¡Te llevo siempre en mi corazón!

A **Yolanda Jaramillo**, indiscutiblemente el destino hace que te encuentre siempre. Las apuestas se cumplen, te quiero mucho ¡Gracias por ayudar en la presentación del proyecto!

A **Andrés Ruiz y Diego Moya**, muchísimas vivencias y excelentes experiencias. Son los mejores panas. Las frases célebres, Andrés: "Próximo gerente" y Diego: "Jeba de las jebas".

## ÍNDICE DE CONTENIDOS

<b>DEDICATORIA.....</b>	<b>v</b>
<b>AGRADECIMIENTO.....</b>	<b>vi</b>
<b>ÍNDICE DE CONTENIDOS.....</b>	<b>viii</b>
<b>ÍNDICE DE TABLAS .....</b>	<b>xii</b>
<b>ÍNDICE DE FIGURAS .....</b>	<b>xiii</b>
<b>RESUMEN.....</b>	<b>xvii</b>
<b>ABSTRACT .....</b>	<b>xviii</b>
<b>CAPÍTULO I .....</b>	<b>1</b>
<b>DISEÑO E IMPLEMENTACIÓN DE UN GUANTE ELECTRÓNICO ORIENTADO A CONTROLAR LOS MOVIMIENTOS DE UN DRON .....</b>	<b>1</b>
1.1. INTRODUCCIÓN.....	1
1.2. GUANTE ELÉCTRONICO .....	3
1.2.1. DATAGLOVE .....	4
1.2.2. CYBERGLOVE .....	5
1.2.3. ACCELEGLOVE .....	5
1.3. OBJETIVOS .....	6
1.3.1. OBJETIVO GENERAL .....	6
1.3.2. OBJETIVOS ESPECÍFICOS .....	6
1.4. JUSTIFICACIÓN E IMPORTANCIA .....	7
1.5. ALCANCE DEL PROYECTO .....	8
1.6. TRABAJOS RELACIONADOS .....	9
1.6.1. DESARROLLO DE UNA INTERFAZ PARA EL RECONOCIMIENTO AUTOMÁTICO DEL LENGUAJE DE SIGNOS.....	9
1.6.2. DISEÑO E IMPLEMENTACIÓN DE UN GUANTE SENSORIZADO PARA EL CONTROL TELEOPERADO DE UN PROTOTIPO DE BRAZO ROBÓTICO PARA APLICACIONES DE MANEJO DE MATERIALES PELIGROSOS.....	10
<b>CAPÍTULO II .....</b>	<b>11</b>
<b>DESCRIPCIÓN DE LOS COMPONENTES DEL PROYECTO .....</b>	<b>11</b>
2.1. ESTADO DEL ARTE .....	11
2.2. FUNDAMENTOS TEÓRICOS.....	12



2.2.1.	SISTEMA DE TELEOPERACIÓN .....	12
2.2.2.	ELEMENTOS DE LOS SISTEMAS DE TELEOPERACIÓN .....	13
2.2.3.	ARQUITECTURAS DE TELEOPERACIÓN.....	15
2.2.4.	INTERFACES .....	16
2.2.4.1.	INTERFACES DIRECTAS.....	16
2.2.4.2.	INTERFAZ MULTIMODAL O MULTISENSORIAL .....	16
2.2.4.3.	INTERFACES PARA CONTROL SUPERVISADO .....	17
2.2.5.	TECNOLOGÍA MEMS (SISTEMAS MICRO-ELECTRO- MECÁNICOS) .....	20
2.2.6.	ACELERÓMETRO .....	23
2.2.7.	GIROSCOPIO .....	26
2.2.8.	GRADOS DE LIBERTAD .....	28
2.2.9.	SENSOR DE FLEXIÓN.....	29
2.2.10.	POTENCIÓMETROS DIGITALES.....	31
2.2.11.	ZIGBEE.....	36
2.2.12.	PROTOCOLOS DE COMUNICACIÓN .....	49
2.2.12.1.	COMUNICACIÓN SERIAL .....	49
2.2.12.2.	COMUNICACIÓN SPI (SERIAL PERIPHERAL INTERFACE) .....	54
2.2.12.3.	COMUNICACIÓN I2C (INTER-INTEGRATED CIRCUITS) .....	60
2.2.13.	ARDUINO .....	68
2.2.14.	FRITZING .....	72
2.3.	ANÁLISIS DE LOS MOVIMIENTOS DE LA MANO Y DEL BRAZO	74
2.3.1.	LA MANO .....	74
2.3.2.	MOVIMIENTOS DEL BRAZO.....	79
2.4.	ENTORNO DE DESARROLLO Y MATERIALES PARA IMPLEMENTAR EL GUANTE ELECTRÓNICO Y EL SISTEMA DE RECEPCIÓN .....	82
2.4.1.	MPU-6050 (UNIDAD DE PROCESAMIENTO DE MOVIMIENTO) .....	82
2.4.2.	SENSORES DE FLEXIÓN .....	91
2.4.3.	ARDUINO NANO USB (V3.0) .....	92
2.4.4.	MÓDULO XBEE S1.....	94
2.4.5.	POTENCIÓMETROS DIGITALES MCP 41100 .....	98
2.4.6.	ESTUCHE CARGADOR DE CELULAR .....	100
2.4.7.	BATERÍA LIPO – LITIO POLÍMERO .....	101

2.4.8. CARGADOR BATERÍA TURNIGY .....	102
<b>CAPÍTULO III .....</b>	<b>104</b>
<b>DISEÑO Y CONSTRUCCIÓN DEL GUANTE ELECTRÓNICO .....</b>	<b>104</b>
3.1. DESCRIPCIÓN DEL PROYECTO .....	104
3.2. ARQUITECTURA DEL SISTEMA .....	106
3.3. ETAPA 1 – DISPOSITIVO MAESTRO .....	107
3.3.1. GUANTE ELECTRÓNICO.....	107
3.3.2. PROCESAMIENTO DEL DISPOSITIVO MAESTRO .....	127
3.3.3. ACONDICIONAMIENTO DE LA SEÑAL ARDUINO-XBEE.....	131
3.3.4. ESQUEMA DEL GUANTE ELECTRÓNICO .....	132
3.3.5. TRANSMISIÓN INALÁMBRICA DE DATOS .....	133
3.4. ETAPA 2 – DISPOSITIVO ESCLAVO.....	140
3.4.1. SISTEMA DE RECEPCIÓN .....	140
3.4.2. ACONDICIONAMIENTO DE LA SEÑAL XBEE-ARDUINO.....	144
3.4.3. RECEPCIÓN DE DATOS.....	145
3.4.4. PROCESAMIENTO DEL DISPOSITIVO ESCLAVO.....	147
3.4.5. ENVÍO DE DATOS.....	150
3.4.6. ESQUEMA DEL SISTEMA RECEPTOR .....	150
3.5. IMPLEMENTACIÓN .....	151
3.5.1. MONTAJE DEL GUANTE ELECTRÓNICO “WILGLOVE” .....	151
3.5.2. MONTAJE DEL SISTEMA RECEPTOR Y CONEXIÓN AL CONTROL DEL DRON .....	157
3.6. DRON X125.....	160
<b>CAPÍTULO IV.....</b>	<b>166</b>
<b>PRUEBAS Y RESULTADOS .....</b>	<b>166</b>
4.1. PRUEBAS DE FUNCIONAMIENTO DEL GUANTE ELECTRÓNICO Y EL SISTEMA RECEPTOR .....	166
4.2. ESCENARIOS DE EVALUACIÓN .....	168
4.3. RESULTADOS DE LAS PRUEBAS DE TELEOPERACIÓN DEL DRON.....	182
4.4. COSTO DEL PRODUCTO FINAL.....	183
<b>CAPÍTULO V.....</b>	<b>186</b>
<b>CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>186</b>
5.1. CONCLUSIONES .....	186

5.2. RECOMENDACIONES.....	188
<b>REFERENCIAS .....</b>	<b>190</b>

## ÍNDICE DE TABLAS

<b>Tabla 1.</b>	Tabla comparativa de potenciómetro mecánico y digital .....	32
<b>Tabla 2.</b>	Propiedades del estándar IEEE802.154.....	38
<b>Tabla 3.</b>	Modelo ISO/OSI y Modelo IEEE802.....	39
<b>Tabla 4.</b>	Comparativa de tecnologías Wireless .....	47
<b>Tabla 5.</b>	Designación de pines MPU-6050 .....	85
<b>Tabla 6.</b>	Designación de pines ADXL345.....	87
<b>Tabla 7.</b>	Designación de pines ITG-3200 .....	90
<b>Tabla 8.</b>	Designación de pines Arduino nano USB (v3.0).....	93
<b>Tabla 9.</b>	Designación de pines Módulo XBee S1 (v3.0) .....	96
<b>Tabla 10.</b>	Designación de pines Potenciómetro Digital MCP 41100.....	99
<b>Tabla 11.</b>	Tabulación de datos que retorna el ADC de los sensores de flexión .....	111
<b>Tabla 12.</b>	Mando del dedo índice para ascender/descender el Dron .....	114
<b>Tabla 13.</b>	Mando del dedo pulgar para hacer videos con el Dron .....	115
<b>Tabla 14.</b>	Movimientos de la mano para gobernar el Dron.....	127
<b>Tabla 15.</b>	Conexiones Arduino Nano con MPU-6050.....	129
<b>Tabla 16.</b>	Conexiones Arduino Nano con los sensores de flexión.....	131
<b>Tabla 17.</b>	Acondicionamiento de la señal Arduino-XBee.....	132
<b>Tabla 18.</b>	Especificaciones módulo XBEE S1, transmisión inalámbrica de datos .....	134
<b>Tabla 19.</b>	Interpretación de la cadena de caracteres del monitor serie de Arduino .....	137
<b>Tabla 20.</b>	Modo de funcionamiento .....	138
<b>Tabla 21.</b>	Parámetros de configuración para las direcciones de los módulos XBee.....	139
<b>Tabla 22.</b>	Características de envío de los módulos XBee .....	140
<b>Tabla 23.</b>	Acondicionamiento de la señal XBee-Arduino.....	145
<b>Tabla 24.</b>	Cadena de caracteres .....	146
<b>Tabla 25.</b>	Características de los potenciómetros digitales.....	147
<b>Tabla 26.</b>	Valores de inicialización de los potenciómetros digitales .....	149
<b>Tabla 27.</b>	Movimientos que se realizan mediante los potenciómetros digitales.....	149
<b>Tabla 28.</b>	Envío de datos desde el control del Dron al cuadricóptero.....	150
<b>Tabla 29.</b>	Vistas desde la cámara a bordo del Dron y cámara externa .....	170
<b>Tabla 30.</b>	Costo de los elementos y sensores usados para implementar el proyecto .....	184

## ÍNDICE DE FIGURAS

<b>Figura 1.</b>	Dron X-125, utilizado para el desarrollo del proyecto .....	2
<b>Figura 2.</b>	Diferentes tipos de Drones .....	3
<b>Figura 3.</b>	DataGlove, prototipo comercializado en la década de 1980 .....	5
<b>Figura 4.</b>	CyberGlove, con sensores magnéticos .....	5
<b>Figura 5.</b>	AcceleGlove, con acelerómetros .....	6
<b>Figura 6.</b>	Diagrama de bloques de un prototipo de teleoperación .....	13
<b>Figura 7.</b>	Telerehabilitación de la mano .....	14
<b>Figura 8.</b>	Dispositivos hápticos .....	17
<b>Figura 9.</b>	Sistema de rehabilitación de la mano .....	18
<b>Figura 10.</b>	Guante con retroalimentación de fuerza, Cybergrasp .....	19
<b>Figura 11.</b>	Guante con retroalimentación táctil, Cybertouch .....	20
<b>Figura 12.</b>	Tecnología MEMS, ejemplo de un teléfono inteligente .....	21
<b>Figura 13.</b>	Sensor inercial, fabricado por InvenSense .....	22
<b>Figura 14.</b>	Acelerómetro ADXL345 .....	24
<b>Figura 15.</b>	Principio de funcionamiento de un acelerómetro .....	25
<b>Figura 16.</b>	Movimiento del giroscopio con respecto a los ejes de rotación X, Y, Z .....	27
<b>Figura 17.</b>	Movimiento del giroscopio respecto al eje de rotación Z .....	27
<b>Figura 18.</b>	Principio de funcionamiento de un giroscopio .....	28
<b>Figura 19.</b>	Sistema de referencia global (X, Y, Z) y móvil ( $\xi$ , $\eta$ , $\zeta$ ) .....	29
<b>Figura 20.</b>	Sensor de flexión .....	30
<b>Figura 21.</b>	Funcionamiento del sensor de flexión .....	30
<b>Figura 22.</b>	Terminal de ajuste, Wiper .....	31
<b>Figura 23.</b>	Funcionamiento de los Potenciómetros Electromecánicos y Digitales .....	32
<b>Figura 24.</b>	Muestreo, cuantización y codificación de ADC (de 3 bits) .....	34
<b>Figura 25.</b>	Pasos conversión análogo digital .....	35
<b>Figura 26.</b>	Frecuencias de muestreo .....	36
<b>Figura 27.</b>	Arquitectura ZigBee .....	40
<b>Figura 28.</b>	Estructura de canales de operación ZigBee .....	41
<b>Figura 29.</b>	Tipos de paquetes ZigBee .....	43
<b>Figura 30.</b>	Topologías de redes ZigBee .....	46
<b>Figura 31.</b>	Aplicaciones ZigBee .....	48
<b>Figura 32.</b>	Paquete de datos comunicación serial .....	51
<b>Figura 33.</b>	Envío de dos bytes, comunicación serial .....	53
<b>Figura 34.</b>	Hardware de la comunicación serial .....	53
<b>Figura 35.</b>	Implementación del Hardware .....	54
<b>Figura 36.</b>	Comunicación SPI configurada en flanco de subida .....	55
<b>Figura 37.</b>	Comunicación SPI, líneas MOSI y MISO .....	56

<b>Figura 38.</b>	Comunicación SPI, línea SS (Slave Select) .....	57
<b>Figura 39.</b>	Múltiples esclavos, las líneas SS (Slave Select) separadas.....	58
<b>Figura 40.</b>	Múltiples esclavos, una sola línea SS (Slave Select) .....	59
<b>Figura 41.</b>	Conexión I2C, Master/Slave .....	61
<b>Figura 42.</b>	Topología I2C, Master/Slave.....	62
<b>Figura 43.</b>	AND cableada. (a) Una o más salidas a '0'. (b) Todas las salidas a '1' .....	63
<b>Figura 44.</b>	Secuencia de inicio y parada .....	64
<b>Figura 45.</b>	Secuencia de inicio .....	64
<b>Figura 46.</b>	Secuencia de parada .....	65
<b>Figura 47.</b>	Transferencia de Datos del protocolo I2C .....	65
<b>Figura 48.</b>	Logo Arduino .....	68
<b>Figura 49.</b>	Interfaz de programación Arduino .....	70
<b>Figura 50.</b>	Logo Fritzing .....	73
<b>Figura 51.</b>	Interfaz de desarrollo de Fritzing .....	73
<b>Figura 52.</b>	Partes de la mano .....	75
<b>Figura 53.</b>	En A) Flexión y extensión del antebrazo, B) Flexión y extensión de la mano, F) Flexión y extensión de dedos .....	76
<b>Figura 54.</b>	En G) Flexión y extensión del dedo pulgar, H) Abducción y Aducción del dedo pulgar .....	77
<b>Figura 55.</b>	Pronación y supinación del antebrazo.....	78
<b>Figura 56.</b>	Flexión del pulgar.....	79
<b>Figura 57.</b>	Pronación del pulgar .....	79
<b>Figura 58.</b>	Extensión y flexión del codo.....	80
<b>Figura 59.</b>	Movimientos del codo .....	81
<b>Figura 60.</b>	Movimiento Flexión y extensión de la muñeca .....	82
<b>Figura 61.</b>	MPU-6050.....	82
<b>Figura 62.</b>	Orientación de los ejes de sensibilidad y polaridad de rotación	83
<b>Figura 63.</b>	Pines del acelerómetro MPU-6050 .....	86
<b>Figura 64.</b>	Designación de pines ADXL345.....	88
<b>Figura 65.</b>	Orientación de los ejes de sensibilidad y polaridad de rotación	89
<b>Figura 66.</b>	Pines del giroscopio ITG-3200 .....	91
<b>Figura 67.</b>	Sensor de flexión .....	91
<b>Figura 68.</b>	Medidas del sensor de flexión.....	91
<b>Figura 69.</b>	Arduino nano USB (v3.0) .....	92
<b>Figura 70.</b>	Pines del Arduino nano USB (v3.0).....	94
<b>Figura 71.</b>	Módulo XBee S1 (izquierda) .....	95
<b>Figura 72.</b>	Placa de conexión con USB (derecha).....	95
<b>Figura 73.</b>	Dimensiones y pines del Módulo XBee S1 .....	97
<b>Figura 74.</b>	Potenciómetro digital (8 pines).....	98
<b>Figura 75.</b>	Designación de pines Potenciómetro Digital MCP 41100.....	100
<b>Figura 76.</b>	Estuche cargador de celular .....	100

<b>Figura 77.</b>	Batería LIPO .....	101
<b>Figura 78.</b>	Cargador de batería Turnigy .....	102
<b>Figura 79.</b>	Descripción del proyecto .....	105
<b>Figura 80.</b>	Arquitectura del sistema.....	106
<b>Figura 81.</b>	Sensor de flexión .....	108
<b>Figura 82.</b>	Ascenso/Descenso del Dron .....	114
<b>Figura 83.</b>	ON/OFF de video, pantalla de control del Dron.....	115
<b>Figura 84.</b>	Encapsulado MPU-6050 .....	116
<b>Figura 85.</b>	Sensor Inercial MPU-6050 .....	117
<b>Figura 86.</b>	Gravedad en los ejes del acelerómetro según la inclinación...	119
<b>Figura 87.</b>	Medición del ángulo de inclinación en un eje del acelerómetro.....	119
<b>Figura 88.</b>	Medición del ángulo de inclinación para tres ejes del acelerómetro.....	120
<b>Figura 89.</b>	Movimientos roll y pitch respecto al guante electrónico.....	122
<b>Figura 90.</b>	Movimiento de pronación de la mano.....	123
<b>Figura 91.</b>	Movimientos y ángulos de flexión-extensión de la mano.....	124
<b>Figura 92.</b>	Movimiento de supinación-pronación media de la mano.....	125
<b>Figura 93.</b>	Movimiento de pronación de la mano.....	126
<b>Figura 94.</b>	Conexiones con Arduino Nano, comunicación I2C.....	128
<b>Figura 95.</b>	Sensores de flexión con Arduino Nano, Divisor de Voltaje.....	130
<b>Figura 96.</b>	Conexiones de XBee con Arduino Nano, Regulador de Voltaje.....	132
<b>Figura 97.</b>	Dispositivo Maestro.....	133
<b>Figura 98.</b>	Comunicación XBEE (izquierda) - sistema receptor (derecha).....	136
<b>Figura 99.</b>	Monitor serie Arduino Nano, lectura de los sensores .....	137
<b>Figura 100.</b>	Interfaz de comunicación de módulos XBee en el X-CTU .....	139
<b>Figura 101.</b>	Control del Dron, joysticks y botones .....	141
<b>Figura 102.</b>	Funciones del Joystick 1 .....	142
<b>Figura 103.</b>	Funciones del Joystick 2.....	142
<b>Figura 104.</b>	Pantalla del control del Dron .....	144
<b>Figura 105.</b>	XBee con Arduino Nano, convertidor de voltaje .....	145
<b>Figura 106.</b>	Potenciómetros Digitales-Arduino Nano, Comunicación SPI.	148
<b>Figura 107.</b>	Dispositivo Esclavo .....	151
<b>Figura 108.</b>	Elementos usados .....	152
<b>Figura 109.</b>	Guante diestro marca Adidas.....	153
<b>Figura 110.</b>	Cierre y pulsera de ajuste .....	154
<b>Figura 111.</b>	Ubicación de los sensores de flexión en el guante .....	155
<b>Figura 112.</b>	Ubicación del acelerómetro y estuche cargador en el guante	155
<b>Figura 113.</b>	Logo “Wilglove MH” .....	156
<b>Figura 114.</b>	Guante electrónico “Wilglove MH” (vista superior) .....	157

<b>Figura 115.</b>	Placa electrónica de los potenciómetros digitales .....	158
<b>Figura 116.</b>	Control del Dron X125.....	158
<b>Figura 117.</b>	Conexiones a la placa del Control del Dron .....	159
<b>Figura 118.</b>	Conexiones del sistema receptor al control del Dron .....	159
<b>Figura 119.</b>	Vista de la caja de acrílico y el control del Dron .....	160
<b>Figura 120.</b>	Partes del Dron X125.....	161
<b>Figura 121.</b>	Dron X125.....	161
<b>Figura 122.</b>	Tipologías de drones y la disposición de sus motores .....	162
<b>Figura 123.</b>	Movimientos de los ángulos roll y pitch respecto del Dron ....	163
<b>Figura 124.</b>	Compensación de los motores del Dron en roll .....	164
<b>Figura 125.</b>	Compensación de los motores del Dron en pitch .....	165
<b>Figura 126.</b>	Estuche cargador usado para pruebas, dispositivo maestro .	166
<b>Figura 127.</b>	Sistema receptor usado para pruebas, dispositivo esclavo ...	166
<b>Figura 128.</b>	Escenario para el reto de la red de vóley .....	169
<b>Figura 129.</b>	Vista panorámica de la biblioteca y patio central de la ESPE	171
<b>Figura 130.</b>	Vista panorámica de la biblioteca y el edificio administrativo	172
<b>Figura 131.</b>	Vista panorámica de la biblioteca y el patio central .....	172
<b>Figura 132.</b>	Vista panorámica del patio central de la ESPE .....	173
<b>Figura 133.</b>	Vista panorámica del Logo de la ESPE.....	173
<b>Figura 134.</b>	Vista panorámica del Logo de la ESPE y paisaje.....	174
<b>Figura 135.</b>	Vista panorámica del Logo de la ESPE.....	174
<b>Figura 136.</b>	Panorámica de los Laboratorios de Ingeniería Electrónica....	175
<b>Figura 137.</b>	Panorámica de los Laboratorios de Ingeniería Electrónica....	175
<b>Figura 138.</b>	Panorámica de los Laboratorios de Ingeniería Electrónica....	176
<b>Figura 139.</b>	Panorámica de los Laboratorios de Ingeniería Electrónica....	176
<b>Figura 140.</b>	Despegue del Dron en las afueras de los laboratorios .....	177
<b>Figura 141.</b>	Parqueaderos de los Laboratorios de Ingeniería Electrónica	177
<b>Figura 142.</b>	Panorámica de las canchas del Departamento de Mecánica	178
<b>Figura 143.</b>	Panorámica de las canchas del Departamento de Mecánica	178
<b>Figura 144.</b>	Despegue del Dron en las canchas de Básquet.....	179
<b>Figura 145.</b>	Vista panorámica de las canchas de Básquet.....	179
<b>Figura 146.</b>	Vista panorámica de los parqueaderos del Bar .....	180
<b>Figura 147.</b>	Vista panorámica del Bar y el edificio central .....	180
<b>Figura 148.</b>	Foto tomada al Dron en vuelo alto .....	181
<b>Figura 149.</b>	Foto tomada al Dron en vuelo bajo .....	181



## RESUMEN

Este trabajo presenta el proceso de desarrollo de un guante electrónico orientado a controlar los movimientos de un dron, que surge de la necesidad de innovar la manera de operar un dispositivo por personas que tengan alguna discapacidad o para quienes, convenientemente, deseen optimizar un mando a distancia. Inicia con una recopilación de información del software y hardware requeridos para el diseño e implementación de un sistema MAESTRO-ESCLAVO que consta de un guante electrónico y un sistema receptor que se comunican inalámbricamente entre sí. En este sentido, se muestra cómo el sistema receptor está conectado directamente con el control remoto del dron, de ahí que se gobierna el cuadricóptero. Posteriormente, se detallan las especificaciones técnicas de los elementos que se usan dentro del proyecto. En base a esto, se realiza una esquematización de los dispositivos maestro y esclavo mediante el software libre FRITZING. Para ello, se expone como los datos son enviados inalámbricamente desde un módulo XBEE, instalado en el guante electrónico y se reciben por un segundo módulo XBEE disponible en el sistema receptor. La información se procesa a través de módulos ARDUINO NANO y XBEE gracias a una programación realizada mediante los softwares libres ARDUINO y XBEE EXPLORER respectivamente. Seguidamente, se presenta el proceso de implementación del sistema de teleoperación con un guante electrónico y un sistema receptor. Para ello, se da a explicación como en el proceso ambos insumos trabajan conjuntamente para gobernar el dron. Se presentan los testeos realizados que corroboran el correcto funcionamiento del prototipo.

### **PALABRAS CLAVE:**

- **GUANTE ELECTRÓNICO**
- **DISPOSITIVOS ARDUINO**
- **DRON**
- **DISPOSITIVOS XBEE**
- **SENSOR**

## **ABSTRACT**

This degree project contains the development process of an electronic glove designed to control the movements of a drone, which comes up from the need of innovate the way to operate a technological device for people with disabilities or simply for who would want to innovate a remote control. Firstly, all the necessary information is gathered about the software and hardware required in order to design and implement a MASTER/SLAVE system. This scheme features a master-device (electronic glove) and a slave-device (receiver system) which communicate wirelessly each other. The receiver system is connected directly to the drone's remote control, therefore the quadcopter is handled. In this sense, it is detailed the standards and technical specifications of the incumbent elements used in this project.

Then, the master-device and the slave-device are schematized with the free software FRITZING. The data are transferred via wireless from a XBEE module installed in the electronic glove and received by a second module XBEE available on the receiving system. The information is transferred, read and processed through ARDUINO NANO and XBEE modules thanks to the programs made through the free software ARDUINO and XBEE EXPLORER respectively.

Finally, it is implemented a teleoperation system with an electronic glove and a receiver system that work together to govern the drone, several tests are performed and the correct operation is confirmed. After that, the conclusions and recommendations are given.

### **KEY WORDS:**

- **ELECTRONIC GLOVE**
- **ARDUINO DEVICE**
- **DRONE**
- **XBEE DEVICE**
- **SENSOR**

## CAPÍTULO I

# DISEÑO E IMPLEMENTACIÓN DE UN GUANTE ELECTRÓNICO ORIENTADO A CONTROLAR LOS MOVIMIENTOS DE UN DRON

### 1.1. INTRODUCCIÓN

Los drones fueron inventados en el siglo XX y tuvieron un gran auge durante la segunda guerra mundial debido a que fueron utilizados con la modalidad de ser *aeronaves no tripuladas*. Esta prestación permitió atacar o hacer reconocimiento de territorio enemigo sin necesidad de un piloto, así como también fueron empleados para entrenar a los operarios de los cañones antiaéreos. A fines del siglo XX, se logró operar los ‘Drones’ mediante radio control dándoles todas las características de autonomía (Cortes, 2015).

Los ‘Drones’, sin duda, presentan una gran utilidad para manejar, proteger y transmitir la información (Guerra electrónica, criptografía) brindando una mayor seguridad en las comunicaciones haciéndolas más difíciles de interceptar. Estas capacidades se han evidenciado en gran medida en la guerra del Golfo y en la guerra de Bosnia. “Según la 23.ª Edición del Diccionario académico, la palabra *dron* (plural drones) consta como adaptación al español del sustantivo inglés *drone* (literalmente ‘zángano’), para referirse a una *aeronave no tripulada*” (Real Academia Española, 2014).

Las aeronaves no tripuladas han sido usadas ampliamente para fines militares; pero con el pasar del tiempo, las nuevas versiones han perdido el aspecto similar al de un avión. Es por esto que se han adoptado nuevas maneras de llamarlos como vehículos aéreos no tripulados o robots voladores (NX, 2014).

Entre las nuevas versiones se pueden mencionar dos: (1) los de despegue vertical que son los de ‘ala rotativa’ y los ‘auto-sustentados’ (2) los de despegue no vertical que son los de ‘ala flexible’ y ‘ala fija’. Generalmente, los que se pueden mantener estáticos en el aire cuentan con más de una

hélice y se asemejan a la forma de un zángano, por eso su nombre 'Dron', asimismo por tener cuatro hélices también se los llama 'Cuadricópteros'. (Ver figura 1).



**Figura 1 Dron X-125, utilizado para el desarrollo del proyecto**

Fuente: (Amazon Try prime, 2015)

Una de las actuales ventajas de los Drones es la manera de poder operarlos mediante un control remoto. Ésta es muy útil y brinda varias prestaciones. En algunos casos, también se les puede dar autonomía, con avanzados ordenadores y algoritmos se podría operar un dron remotamente. De igual forma, se establecen sistemas de navegación y detección para el tráfico de drones. En el caso de este proyecto, se hará referencia solamente a los 'operados por control remoto'.

En esta nueva época del siglo XXI se han superado métodos y procesos antiguos o tradicionales, gracias a la ayuda de la ingeniería, como por ejemplo en el transporte, la comunicación, la producción, en la construcción, entre otras aplicaciones. Actualmente, existen varios proyectos que buscan generar una serie de aplicaciones diversa de Drones en varios campos, no sólo desde el comercial sino inclusive hasta el artístico, como en el campo de la fotografía, buscando diferentes aplicaciones desde distribución de internet hasta salvavidas.

Las importantes prestaciones que ofrecen los nuevos materiales en el mercado, que son más livianos y resistentes, así como las baterías menos pesadas y duraderas permiten que ‘Drones’ sofisticados vuelen, giren en el aire y permitan cargar objetos de un lado a otro. Todo depende que orientación se le pueda dar a un Dron, según eso se lo construye con las características necesarias. A continuación en la figura 2 se muestran algunos drones:



**Figura 2 Diferentes Drones comerciales**

Fuente: (Gutierrez, 2015)

## **1.2. GUANTE ELÉCTRICO**

Los primeros prototipos de guantes electrónicos fueron orientados a la realidad virtual. La ‘VPL Research’ es una compañía que innovó la tendencia en el desarrollo y aplicación de estos dispositivos. Fue fundada en el año 1984 en California por Jaron Lanier. Más tarde, Thomas Zimmerman inventó el ‘DataGlove’, que usaba 6502 microcontroladores, junto con Mitch Altman lograron desarrollar el primer guante electrónico que fue comercializado principalmente en Norte América (Lanier, 2016).

Existen en la actualidad todo tipo de inventos e innovaciones en el campo de la electrónica. En general, la mayoría de las aplicaciones son provistas para la mejora de la calidad de vida de las personas. La mayoría de

inventos buscan enfoques muy específicos, tal es el caso del desarrollo de drones orientados a personas con discapacidad y otros. Entre estos se encuentra el guante electrónico, el mismo que ha tenido algunas aplicaciones en varios campos. Por ejemplo, existen guantes electrónicos fabricados para el reconocimiento y traducción de lenguaje de señas, otros aplicados en procesos de enseñanza-aprendizaje, para conductores parapléjicos, dentro del campo de la telemanipulación así como en las actividades de realidad virtual, dentro del campo médico, en la aplicación de procesos científicos, hasta actividades artísticas, como en la música y el arte en general, y sistema de videojuegos y simulaciones de comunicación en PC.

Es así como nace la idea de construir un modelo de guante electrónico de bajo coste para controlar determinados objetos como una silla de ruedas, prender o apagar una luz en una casa o para controlar algún movimiento. En sí mismo, el guante se puede adaptar para cualquier aplicación que resulte útil para la vida diaria o para personas con discapacidad. En este caso, se ha fijado la atención a la innovación en la manera de controlar los movimientos de un dispositivo electrónico, que en esta oportunidad se ha escogido un Dron por sus diversas aplicaciones con proyección a diferentes aplicaciones en el futuro.

Para este efecto, se emplea el uso sensores de flexión, acelerómetros y otros elementos electrónicos en el diseño e implementación. A continuación, se citan algunos tipos de guantes electrónicos ya existentes en el mercado.

### **1.2.1. DATAGLOVE**

En California, en la década de los 80's aparece este modelo de guante electrónico, que fue creado por Zimmerman en la empresa 'VPL Research'. Este primer prototipo de guante electrónico cuenta con tecnología de fibra óptica, misma que es usada para medir la flexión de los dedos. Su funcionamiento consiste en enviar un determinado haz de luz según como se flexionan los dedos. Seguido, con un diodo fotoeléctrico se medirá la

intensidad de luz que varía según el movimiento como se muestra en la figura 3 (Lanier, 2016).



**Figura 3 DataGlove, prototipo comercializado en la década de 1980**

Fuente: (Virtual Realitys LLC, 2016)

### 1.2.2. CYBERGLOVE

A diferencia del DataGlove, este modelo de guante electrónico llamado Cyberglove integra tecnología con sensores magnéticos en cada uno de los dedos. Estos sensores posibilitan medir la distancia que existe en medio de ellos como se observa en la figura 4 (Virtual Reality, s.f.).



**Figura 4 CyberGlove, con sensores magnéticos**

Fuente: (Virtual Reality, s.f.)

### 1.2.3. ACCELEGLOVE

La característica principal de este guante electrónico es que se lo puede programar en Java para cualquier aplicación deseada que integre el software (SDK, Drivers y API). Desarrollado por AnthroTronix, está enfocado al control robótico, videojuegos entrenamiento deportivo o para traducir el lenguaje de señas.

Este guante cuenta con acelerómetros en los dedos y palma de la mano para registrar una orientación tridimensional. El material empleado en este dispositivo es una mezcla de nylon transpirable ajustable y, a su vez, cuenta con otro componente para hacer el seguimiento de los movimientos del brazo de la persona que lo use. El costo aproximado en el mercado es de \$500 dólares. (Ver figura 5). (Cortes-Monroy, 2009)



**Figura 5 AcceleGlove, con acelerómetros**

Fuente: (Cortes-Monroy, 2009)

### **1.3. OBJETIVOS**

#### **1.3.1. OBJETIVO GENERAL**

- Implementar el diseño de un guante electrónico inalámbrico que permita controlar un Dron con movimientos de la mano.

#### **1.3.2. OBJETIVOS ESPECÍFICOS**

- Analizar el estado del arte respecto a los drones, su control y funcionamiento.
- Investigar el funcionamiento y operación de los acelerómetros y sensores de flexión.



- Realizar un diseño con sensores que permita la construcción de un guante electrónico para el manejo sencillo del Dron.
- Realizar las pruebas de funcionamiento del guante con un Dron.

#### **1.4. JUSTIFICACIÓN E IMPORTANCIA**

En el siglo XXI han surgido considerables avances y nuevas tecnologías que han revolucionado la manera de comunicación y de telecomunicación. Una de esas es el Dron que tiene gran capacidad de maniobra de portar objetos de un lugar a otro. Éste es de fácil acceso a lugares remotos. Puede retransmitir fotos y videos de planos de difícil acceso, en el mar o lugares inhóspitos, donde actúan como salvavidas, realizan ortofotos (para mapas), emiten y receptan señales de vario tipo, hacen vigilancia o monitoreo e, inclusive, sirven para entretenimiento (Anderson, 2014). Todo esto ha definido la importante utilidad de los drones debido a sus costos menos elevados.

Actualmente, los drones tienen muchos usos, tales como se listan a continuación:

- Agricultura
- Servicios Forestales
- Seguridad
- Socorrista
- Cartografía
- Espionaje
- Audiovisual
- Distribución de Internet (idea planteada por Google)
- Monitoreo de tráfico vehicular
- Diversión.

Una vez verificada la proyección a futuro de los 'Drones' y con el conocimiento de sus prestaciones, el objeto del proyecto es innovar la manera de controlarlos. Por ende, se persigue desarrollar un 'guante electrónico' que sea didáctico, fácil de manejar y, al mismo tiempo, inalámbrico, con motivo de

controlar un Dron que sirva para controlar un remoto mediante un simple movimiento de la mano hacia arriba o abajo. De esta forma, se espera prescindir de botones o palancas usualmente utilizados en remotos.

En este sentido, se propone el desarrollo de una versión con materiales de uso de bajo coste, optimizando la presentación y modelo. Así, el diseño electrónico contará con la capacidad de proporcionar movimientos suficientes para controlar el Dron. De esta manera, el presente proyecto de investigación también coincide con las líneas de investigación en TICs de Protocolos de Comunicación de la Universidad de las Fuerzas Armadas – ESPE.

Se incluyen algunos otros posibles campos beneficiados con la innovación planteada, tales como:

- Personas que no puedan manejar un Joystick o control
- Personas que no gocen de pleno movimiento de los dedos de la mano
- Personas con discapacidad

### **1.5. ALCANCE DEL PROYECTO**

Este proyecto tiene como objetivo la implementación del diseño de un 'guante electrónico' para controlar un Dron. Este guante funciona con movimientos de la mano, los mismos que permiten manejar fácilmente un Dron. Para este efecto, se emplean acelerómetros para tener un control de la maniobrabilidad del guante y de esta manera ordenar un movimiento al Dron. Además, se complementa el guante con sensores de flexión para tener un mejor dominio del movimiento propiamente dicho, estos sensores cambian su resistencia según su flexión. La versatilidad del prototipo es su implementación inalámbrica que permitirá que el guante se comunique de manera remota con el Dron.

## **1.6. TRABAJOS RELACIONADOS**

### **1.6.1. DESARROLLO DE UNA INTERFAZ PARA EL RECONOCIMIENTO AUTOMÁTICO DEL LENGUAJE DE SIGNOS.**

Este proyecto presenta una propuesta de que promueve una interfaz gráfica con el objeto de reconocer automáticamente el lenguaje de señas, mismo que se expresa en un lenguaje de programación en JAVA. Esta propuesta se fundamenta en análisis efectuados en individuos que presentan sordera con necesidades de aprendizaje de escritura y lectura. Así, este grupo de personas precisan de una instrucción primaria en el "...abecedario basado en procesos cognitivos del aprendizaje de una persona sorda..." (Chacón Mosquera, 2013, pág. vi). Una de las principales características en este grupo de personas es que presentan una predominancia en su memoria visual. Por tanto, la asociación gráfica que se consigue entre el signo efectuado y la letra representada en el alfabeto tiene sentido para la imagen de la letra del alfabeto que se simboliza.

En este sentido, este proyecto presenta una propuesta de interfaz, que se emplearía para el reconocimiento de señales y signos básicos. Este estudio se apalanca como línea base para una siguiente propuesta, una que incluya un sistema de traducción. Así, éste buscaría generar traducción oral simultánea de lenguaje de señas frente al castellano. La propuesta explica que, para que se dé el reconocimiento se necesita que el guante electrónico esté constituido por seis (6) giroscopios colocados cada uno en uno de los sensores, mismos que ayudarán a la representación del signo. Cada signo representado es comparado dentro de una base de datos para la identificación de cada signo a una letra o palabra. En vista que no todas las palabras pueden estar registradas, el programa permite agregar nuevas palabras gracias a una opción con la que el programa cuenta (Chacón Mosquera, 2013, pág. vi).

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

Este proyecto presenta el diseño y la fabricación de un dispositivo de control concebido de forma de un guante sensorizado, mismo que sirve 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 de la investigación de acelerómetros y sensores flexoresistivos, los que son usados para reconocer y censar 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. Adicional, se implementa 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, el que resuelve el modelo cinemático inverso y directo del manipulador. Éste permite el control del mismo a través de un PC. Este HMI adiciona el video de cámaras web usadas para supervisión (Herrera Ñunez & Zambrano Chávez, 2013, pág. xviii).

## **CAPÍTULO II**

### **DESCRIPCIÓN DE LOS COMPONENTES DEL PROYECTO**

#### **2.1. ESTADO DEL ARTE**

Existen grandes empresas de tecnología que han desarrollado diferentes guantes electrónicos. La mayoría de estos han sido enfocados a la realidad virtual. Además de estos, existen otros varios proyectos de guantes electrónicos que se han realizado para la robótica entre otras aplicaciones varias. No obstante, muchos de estos presentan condiciones en su uso. Las principales limitantes de estos ingeniosos dispositivos electrónicos son los costos elevados y la orientación de su aplicación de uso para la que son desarrollados. De esta condición surge la necesidad de desarrollar una aplicación de similares características con elementos de bajo coste, orientada a controlar algún objeto que, como en el caso de personas con discapacidad frente a una silla de ruedas, ayude a facilitar el uso del objeto. En este proyecto se ha considerado controlar los movimientos de un dron.

De esta forma, el propósito de este proyecto es gobernar el dispositivo mediante la capacidad de movimiento de las extremidades superiores, específicamente la palma y dedos. Para el cumplimiento de este objetivo se considera el uso de un guante habitual; el mismo que, dotado de elementos electrónicos adecuados, cumpla con la finalidad de realizar una comunicación entre el control, mismo que está vinculado con el dron, y el guante electrónico. Así, se plantea el diseño de una interfaz inalámbrica mano-control, aplicando elementos electrónicos de código abierto, multiplataforma y de bajo costo. Con esto se pretende dar la pauta para el uso de un guante electrónico de fácil manejo para futuras aplicaciones.

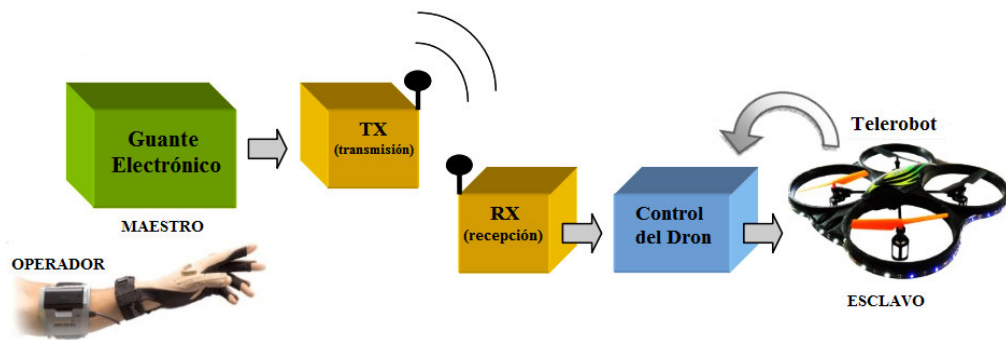
## 2.2. FUNDAMENTOS TEÓRICOS

### 2.2.1. SISTEMA DE TELEOPERACIÓN

Para iniciar con el análisis de sistemas de teleoperación, es imprescindible iniciar con la presentación de su definición.

***Teleoperación: conjunto de tecnologías que comprenden la operación o gobierno a distancia de un dispositivo por un ser humano. Por tanto, teleoperar es la acción que realiza un ser humano de operar o gobernar a distancia un dispositivo; mientras que un sistema de teleoperación será aquel que permita teleoperar un dispositivo, que se denominará dispositivo teleoperado (Nuño Ortega & Basañez Villalengua, 2004, pág. 5).***

En los sistemas de teleoperación normalmente existen dos elementos electrónicos fundamentales. El primer elemento es conocido como maestro o *master*. El otro está denominado como esclavo o *slave*, el que usualmente está ubicado en una zona lejana o en lugares que se deba proteger al operador de algún ambiente hostil (Aliaga, 2000). Se concluye, por tanto, que los sistemas de teleoperación son sistemas que permiten a una persona operar un dispositivo maestro para controlar el movimiento de un dispositivo esclavo. La relación de ambos dispositivos responde a la capacidad de interacción en un espacio específico para el cometido de una actividad concreta. Este tipo de sistema tiene la capacidad de proveer al operador la realimentación sensorial del entorno, tales como imágenes, movimientos, fuerza, distancia, entre otras variables, según se grafica en la figura 6 (Padilla, 2014, pág. 12).



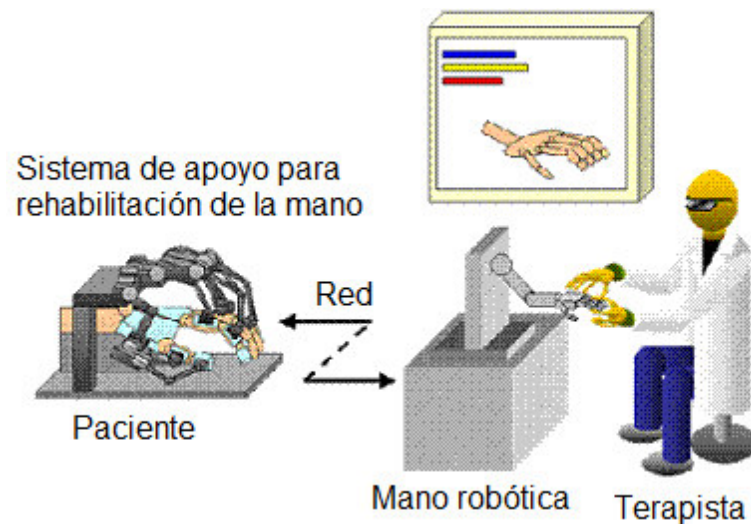
**Figura 6 Diagrama de bloques de un prototipo de teleoperación**

Fuente: (Padilla, 2014, pág. 23)

Existen variables importantes a considerar en un sistema de teleoperación. Se puede citar, por ejemplo, la capacidad de procesamiento de las señales en la etapa de transmisión de información. Esto debido a que algún retraso puede jugar un papel determinante en el sistema y, con ello, variar la precisión y coordinación hombre-dispositivo. Si bien es cierto que la responsabilidad del sistema recae en el operador, porque el desenvolvimiento del dispositivo maestro depende de la experticia y destrezas del sujeto; no menos es cierto que el diseño de la interface hombre-máquina es de alta relevancia para el buen desempeño del sistema (Padilla, 2014, pág. 20).

### **2.2.2. ELEMENTOS DE LOS SISTEMAS DE TELEOPERACIÓN**

Los elementos de este tipo de sistemas generalmente están constituidos por dos dispositivos. El primero es un manipulador local. El segundo es, en este sentido, un robot remoto. Además, estos sistemas presentan un canal de comunicación, el medio ambiente con el que interactúa el dispositivo remoto y el operador humano (Padilla, 2014, pág. 14). Estos elementos se los puede identificar en la figura 7:



**Figura 7 Telerehabilitación de la mano**

Fuente: (Hand telerehabilitation system, 2015)

- **Operador o teleoperador:** Es un individuo que realiza la acción de control a distancia. Puede ser un control continuo, de monitorización o de una intervención intermitente, según las características de cada sistema.
- **Dispositivo teleoperado:** Es una máquina o dispositivo que está siendo controlada por el operador. Trabaja en una zona remota y puede ser un robot de manipulación o terminal de características equivalentes.
- **Interfaz:** Es un dispositivo maestro, así como los elementos que permiten que puedan interactuar el sujeto operador con el dispositivo esclavo. Estos están considerados en lo que se denomina interfaz.
- **Control y Canales de Comunicación:** Son todos los dispositivos que permiten que se transmita y se acoplen el total de las señales transmitidas bidireccionalmente, justo entre el dispositivo remoto frente al dispositivo local.
- **Sensores:** Es el total de los dispositivos que reciben la información remitida, sea por el dispositivo remoto o local. Esta información es suministrada a la interfaz así como del control.



### 2.2.3. ARQUITECTURAS DE TELEOPERACIÓN

De acuerdo a estudios previos, existen diferentes maneras de establecer un sistema de teleoperación y éstas “...se diferencian por la información que se intercambia entre el [maestro] y el [esclavo] y por el tipo de detección que se requiere...” (Padilla, 2014, pág. 15). Se clasifican en:

- **Esquema posición-posición:** No existe necesidad de sensores de fuerza. La colocación del esclavo y maestro se establece en función de ambos.
- **Esquema fuerza-posición:** La colocación del dispositivo esclavo se establece toda vez se coloque el robot master. Así, las fuerzas que se reproducen en el esclavo permiten los cálculos según los cuales el dispositivo maestro puede accionar los sensores. De esta forma, es imprescindible medio cada fuerza que se reproduce durante este esquema.
- **Esquema fuerza-fuerza:** La relación maestro-esclavo está establecida por el tipo y características de sus fuerzas. Esto permite que se establezca, entonces, la colocación de la posición de ambos dispositivos.
- **Esquema de cuatro canales:** Se efectúa por la reciprocidad de información de fuerzas y colocación entre los dispositivos. La condición teórica de este esquema resulta en una transparencia infinita.

En este sentido, se puede concebir que las arquitecturas de teleoperación se relaciona, principalmente, en la estabilidad y la transparencia en el sistema y permiten diferentes grados de telepresencia. La estabilidad es esencial en los sistemas de teleoperación. Ésta está sujeta al ruido eléctrico y los retrasos en las señales de comunicación. La transparencia por su parte se refiere a la capacidad que tenga el operador para precisar toda fuerza recibida por el remoto. Esto hace referencia a lo que se conoce como interfaces hápticas.

## 2.2.4. INTERFACES

Según estudios previos, toda vez las interfaces son las partes del sistema que toman contacto directo con el hombre se pueden encontrar tres categorías (Nuño Ortega & Basañez Villalengua, 2004, págs. 19-21), que son de este tipo:

- Interfaces directas
- Interfaces multimodal o multisensorial
- Interfaz de control supervisado

### 2.2.4.1. INTERFACES DIRECTAS

Entre las interfaces directas se pueden identificar los controladores de mano. El ejemplo más palpable son los *joysticks*, mismos que pueden ser reconocidos como *applets* de java. En este caso, quien opera tiene la capacidad de controlar el vehículo o manipulador. Usualmente, estas son las interfaces tradicionales. Lo característico de estas interfaces es que el usuario actúa en tiempo real en todo momento, durante el proceso teleoperado, lo que le hace sentir que está en el lugar remoto. Al mismo tiempo, el sujeto que opera el dispositivo presenta contacto visual mediante un circuito audiovisual. Este tipo de interfaces se efectúan mediante la aplicación de un alto nivel de ancho de banda para la comunicación, así como retardos casi nulos. Por sus características, resultan ideales los sistemas que trabajan en tiempo real.

### 2.2.4.2. INTERFAZ MULTIMODAL O MULTISENSORIAL

Este tipo de interfaces proporcionan al operador diferentes maneras de control. Algunos ejemplos típicos de estas diferentes maneras de control son actuadores individuales con realimentación gráfica y realimentación visual.

### 2.2.4.3. INTERFACES PARA CONTROL SUPERVISADO

En estas interfaces se encuentran incluidos los dispositivos hápticos con retroalimentación de fuerza y torque, tales como guantes o similares, así como una variedad de *trackers* o cualquier interfaz que reproduzca una telepresencia del máximo nivel en el sitio remoto (Ver figura 8).



**Figura 8 Dispositivos hápticos**

Fuente: (Nuño Ortega & Basañez Villalengua, 2004, pág. 21)

### INTERFAZ HÁPTICA

El término háptico viene de la palabra *hapteshai* que significa tocar, palabra proveniente del idioma griego (Nuño Ortega & Basañez Villalengua, 2004, pág. 6). Las interfaces hápticas inducen al operador en la capacidad de manipular e, inclusive, sentir ambientes y sus contenidos dentro de contextos

virtuales, además de poder controlar teleoperaciones, tal como se muestra en la figura 9. Entre los campos de aplicación de estas interfaces se encuentra:

- Medicina
- Educación
- Entretenimiento
- Industria
- Artes gráficas



**Figura 9 Sistema de rehabilitación de la mano**

Fuente: (Franco, 2007)

## **CLASIFICACIÓN DE LAS INTERFACES HÁPTICAS**

De acuerdo a las condiciones que se buscan condicionar en el sistema, se puede determinar la información que se debe enviar al usuario (Martín Doñate, 2004). Esta información puede clasificarse en:

- Retroalimentación de fuerza
- Retroalimentación táctil
- Retroalimentación propio receptiva

La retroalimentación de fuerza proporciona la información suficiente y concierne a las condiciones "...la dureza, peso e inercia del objeto virtual..." (Martín Doñate, 2004) (Ver figura 10). La retroalimentación táctil, en este sentido, permite que se acopie información imprescindible, tales como "...la geometría del objeto virtual, su rigurosidad y temperatura entre otros..." (Martín Doñate, 2004) (Ver figura 11). Este tipo de interfaz, entonces, permite un adecuado nivel de retroalimentación, además de proporcionar la información correspondiente a la colocación a detalle del usuario, inclusive.

El tipo de interfaces que proporcionan una retroalimentación de fuerza son de este tipo (Véase figura 10):

- **Dispositivo de Escritorio:** Este tipo de interfaz reproduce la interacción del sistema, tal es el caso de un joystick por ejemplo.
- **Guante:** Este tipo de interfaz genera una manipulación dexterizada, puesto que permite que se generen varios puntos de contacto. Estos puntos son los que generan los retornos de fuerza.



**Figura 10 Guante con retroalimentación de fuerza, Cybergrasp**

Fuente: (Martín Doñate, 2004)

Por otro lado, existen interfaces con retroalimentación táctil, como las siguientes:

- **Dispositivo de Escritorio:** Este tipo de interfaz induce la capacidad de sensación de la fuerza en el operador sentir, tales como la estructura, el contorno, o cualquier tipo de elementos localizados en dos dimensiones.
- **Guantes:** Este tipo de interfaz induce a la representación real de movimientos al momento que se produce el contacto con los cuerpos virtuales. Además, permite el reconocimiento de condiciones como la contextura (véase la figura 11).



**Figura 11 Guante con retroalimentación táctil, Cybertouch**

Fuente: (Martín Doñate, 2004)

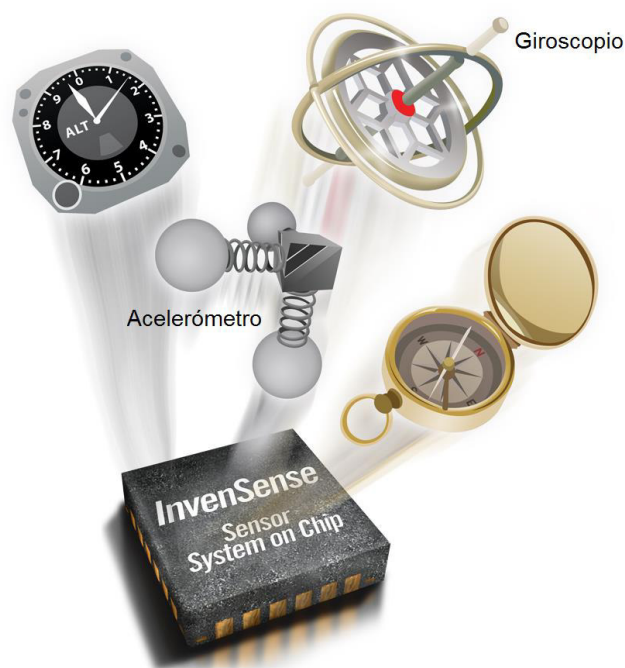
### **2.2.5. TECNOLOGÍA MEMS (SISTEMAS MICRO-ELECTRO-MECÁNICOS)**

Esta tecnología aprovecha las prestaciones que ofrecen el micromaquinado y la microelectrónica. Es decir, se integra elementos electrónicos y mecánicos en una base de silicio sobre un mismo sustrato (ver figura 12). Una de las ventajas de los MEMS es su tamaño, que va desde las décimas de micra hasta los cientos de micras. Esto permite su uso e implementación en sistemas



## SENSORES INERCIALES

Los sensores inerciales se usan fundamentalmente para la captura y análisis de movimiento. Estos son dispositivos capaces de censar los movimientos físicos que pueden adoptar los objetos (Neoteo, 2011). Lo importante de resaltar de estos dispositivos es que actúan sin una referencia externa, puesto que no trabajan respecto a las características de un entorno<sup>1</sup>. Así pues, se puede nombrar a los siguientes sensores como magnetómetro, acelerómetro y giroscopio. *Para este estudio, sólo se hará referencia al acelerómetro y giroscopio (Ver figura 13).*



**Figura 13 Sensor inercial, fabricado por InvenSense**

Fuente: (InvenSense, InvenSense Motion, s.f.)

Para combinar estos dos dispositivos inerciales se usa la denominada Unidad de Medición Inercial, conocida en inglés como *Inertial measurement unit* - IMU, la misma que está basada en la tecnología MEMS, con características

<sup>1</sup> Generalmente tienen 6 grados de libertad, que se explica más adelante



más ligeras y más pequeñas que las habituales. Ésta sirve para producir una medición en tres dimensiones de la fuerza específica y la velocidad angular.

Con motivo de generar control, en los sistemas inteligentes se identifican los siguientes movimientos fundamentales: la aceleración, vibración, golpe (shock), inclinación (tilt) y rotación (pan) (Neoteo, 2011). Este tipo de movimientos son expresiones reales de la aceleración, por tiempo y espacios. Si se analiza cada movimiento por separado se puede asimilar de mejor manera las posibilidades que un acelerómetro puede ofrecer (Neoteo, 2011).

Se sabe que la unidad de medida de la velocidad es  $\frac{m}{s}$ . Así, la aceleración es una relación proporcional entre el tiempo como velocidad y una unidad de espacio. Por otro lado, la acción gravitatoria se involucra en la relación objeto-inclinación. Debido a que los modos de detección de movimientos –tales como el golpe, el grado de inclinación y otros– se efectúan en determinados aspectos de la aceleración, y a saber que la *gravedad* ejerce sobre los mismos, son objetos de medición por la fuerza “g” (gravedad es igual a  $9,8\frac{m}{s^2}$ ).

Por el momento, se han analizado los movimientos lineales que se tienen con un acelerómetro (aceleración, vibración, golpe, inclinación). Seguido, en el movimiento de rotación se “...requiere la comprensión de un movimiento angular. E[ste] modo difiere de los anteriores porque la rotación puede tener lugar sin observarse cambios en la aceleración...” (Neoteo, 2011). Para medir estos valores se usa el giroscopio.

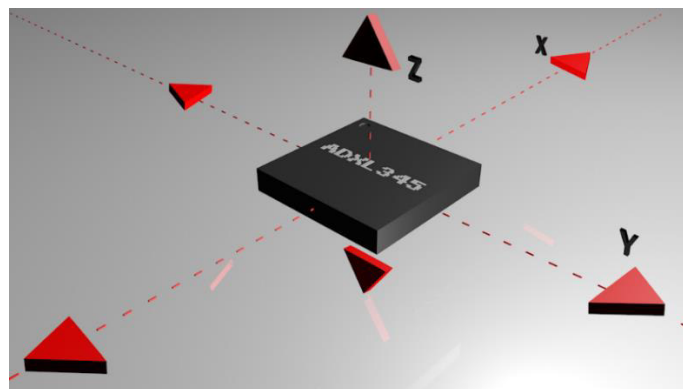
### 2.2.6. ACELERÓMETRO

Este tipo de sensores electromecánicos han sido diseñados para medir la fuerza de aceleración de un objeto según un eje referencial X, Y o Z (Sparkfun,

2012). La aceleración se puede relacionar con el movimiento de un objeto. A estos movimientos se les llaman fuerzas dinámicas<sup>2</sup> o fuerzas estáticas.

Para medir el efecto de cada movimiento (vibración, golpe e inclinación) el acelerómetro toma en cuenta la fuerza gravitacional “g” (una unidad g es igual a  $9,8 \text{ m/s}^2$ )<sup>3</sup>. Esto a decir verdad, puesto la unidad de fuerza que ejerce la gravedad sobre los objetos es tomada en cada eje del espacio X, Y y Z (Neoteo, 2011).

Por lo tanto, el acelerómetro está diseñado para responder a las diferentes maneras asociadas a un movimiento. Según dicho movimiento se genera una respuesta. De esta manera, si se tiene un acelerómetro de tres ejes (ver figura 14), se puede obtener un resultado o una medida en cada uno de ellos, siendo esta medida la que representa el movimiento que se produce.



**Figura 14 Acelerómetro ADXL345**

Fuente: (Sparkfun, 2012)

El número de ejes que tiene un acelerómetro es importante al momento de seleccionar el dispositivo. Existen algunos tipos, según el siguiente detalle:

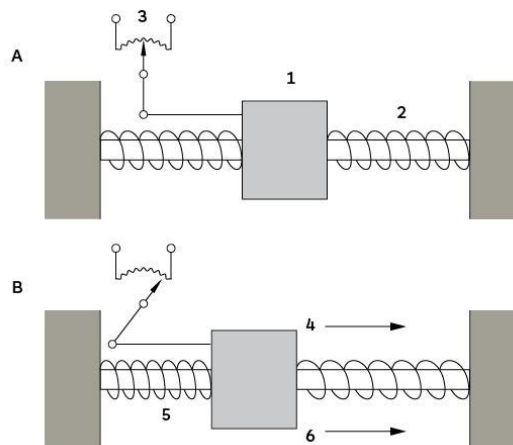
<sup>2</sup> Para el guante electrónico de datos, se utilizan las fuerzas dinámicas

<sup>3</sup> La gravedad en el planeta es equivalente a  $9,8 \frac{\text{m}}{\text{s}^2}$ , pero esta puede variar ligeramente con la altitud.

- De un solo eje, comúnmente usados para medir niveles de vibración simples;
- De dos ejes, miden la aceleración o vibración en los ejes X y Y, hacia adelante y atrás, y de lado a lado;
- De tres ejes, es el mismo de dos ejes añadido el eje Z. Estos son capaces de medir complejas posiciones tridimensionales.

## FUNCIONAMIENTO DEL ACELERÓMETRO

Para medir diferencias en la aceleración se muestra el desplazamiento de una masa sólida con respecto a su contenedor (Ver figura 15). En la parte A, la masa se mantiene en el centro de las dos barras, siendo que el contenedor está en reposo. El acelerómetro arroja un voltaje intermedio en este estado. En la parte B se observa que la inercia de la masa se queda ralentizada en la parte izquierda. Esto ocurre porque el contenedor acelera hacia la derecha, produciendo una compresión del resorte y estrechándolo delante de la masa (IHS Engineering, 2015). Cuando esto ocurre, el dispositivo registra un voltaje más alto relativo a la medida de la aceleración. De acuerdo a la desaceleración, la masa regresa a su estado de reposo y la salida de voltaje disminuye correspondientemente. A continuación se describe la operación de un acelerómetro lineal electromecánico básico:



**Figura 15 Principio de funcionamiento de un acelerómetro**

Fuente: (IHS Engineering, 2015)

Estos resortes minúsculos se mueven internamente a medida que las fuerzas actúan sobre el sensor. Existen acelerómetros que internamente contienen placas capacitivas que son adjuntadas a estos resortes minúsculos. A medida que las placas se mueven en relación de la una con la otra, la capacitancia entre ellas cambia y, según estos cambios en capacitancia, se puede determinar la aceleración. Otros acelerómetros usan materiales piezoeléctricos. Estos diminutos cristales estructuran una carga eléctrica de salida cuando son expuestos a una presión mecánica.

Ahora, considerando la gravedad al momento de realizar las mediciones se pueden considerar las siguientes afirmaciones:

- Se obtiene una aceleración igual a la gravedad cuando está en reposo,
- En caída libre se lee una aceleración de cero,
- Se obtiene la aceleración expresada en las coordenadas (cambio de velocidad en el dispositivo) haciendo una compensación de la gravedad (se resta la gravedad de la salida obtenida), y
- Se obtiene un valor según el efecto de gravedad que se produce por la rotación del dispositivo.

### **2.2.7. GIROSCOPIO**

Los giroscopios se encargan de medir la rotación alrededor de cada uno de los ejes X, Y y Z. Ésta se representa según la velocidad angular "...en grados por segundo ( $^{\circ}/s$ ) o en revoluciones por segundo (RPS). La velocidad angular es simplemente una medida de velocidad de rotación" (Ronzo, s.f.) (Ver figura 16). Los giroscopios son normalmente usados en los sistemas de navegación y en robótica. Usualmente, estos sirven para determinar la orientación de un objeto en un sistema autónomo.



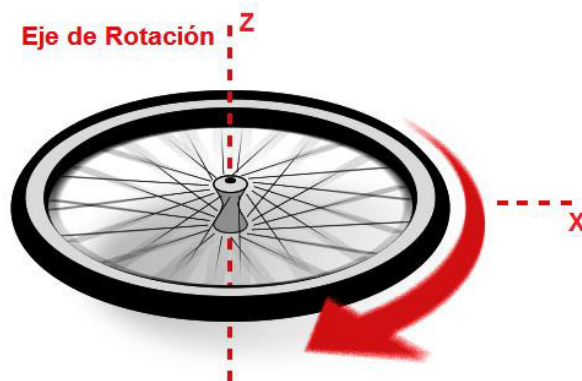
**Figura 16 Movimiento del giroscopio con respecto a los ejes de rotación X, Y, Z**

Fuente: (InvenSense, InvenSense Motion, s.f.)

Por ejemplo si se tiene una moto robótica (tamaño pequeño) y se quiere mantenerla en equilibrio, se puede usar giroscopios; puesto que estos permiten la medición de la rotación de posición balanceada con motivo de retroalimentar errores y permitir la estabilidad del motor.

### **FUNCIONAMIENTO DEL GIROSCOPIO**

En la figura 17 se puede observar que la rueda gira alrededor del eje Z. A esta rotación se la puede medir en RPS (revoluciones por segundo) o en grados por segundo ( $^{\circ} / s$ ). A esto se lo llama *velocidad angular*.

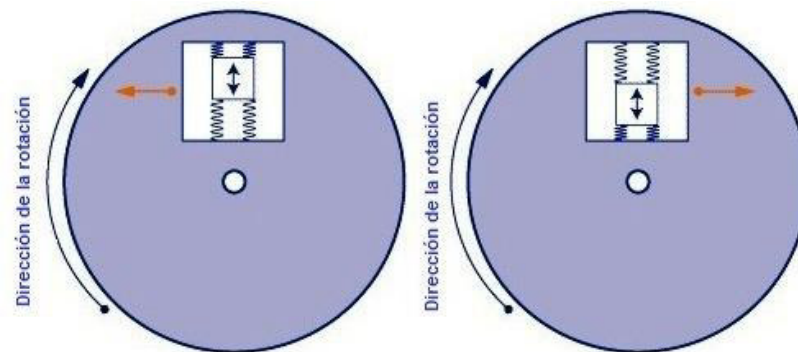


**Figura 17 Movimiento del giroscopio con respecto al eje de rotación Z**

Fuente: (Ronzo, s.f.)

El ITG-3200 puede analizar y efectuar una medición de rotación en los tres ejes, largo ancho y profundidad. Cuando el giroscopio está rotado, una pequeña masa es desplazada a medida que la velocidad angular cambia. Este hecho es una clara señal de una posible baja de corriente eléctrica. Esto puede ser amplificado y medido por un dispositivo microcontrolador.

El sensor MEMS es de tamaño muy pequeño y está dentro del giroscopio<sup>4</sup>. Éste se usa una pequeña masa de resonancia para medir la velocidad angular. Este valor se expresa en señales eléctricas cuando se hace girar el giroscopio. (Ver figura 18).



**Figura 18 Principio de funcionamiento de un giroscopio**

Fuente: (Ronzo, s.f.)

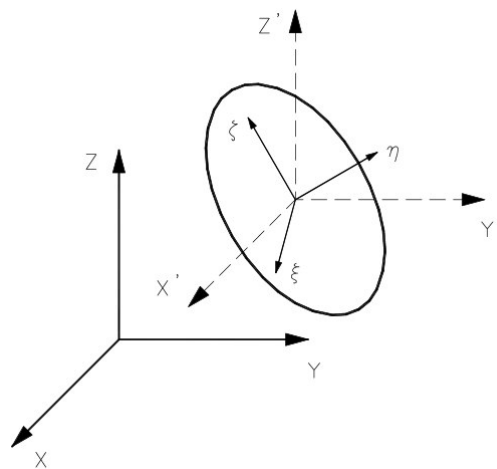
### 2.2.8. GRADOS DE LIBERTAD

Todo cuerpo ubicado en el espacio presenta 6 tipos de grados de libertad. En consecuencia, se necesitan, entonces, seis coordenadas independientes para determinar su situación. De las seis coordenadas, tres son necesarias para especificar los movimientos de traslación y las otras tres son necesarias para los giros. Las seis coordenadas definen la localización de un sistema de coordenadas cartesiano, mismo que está fijado en el cuerpo respecto a un sistema de coordenadas global fijo, el que se encuentra fuera del cuerpo. (Pillajo Obando & Robayo Cajamarca, 2012, pág. 44).

<sup>4</sup> Entre 1 a 100 micrómetros, el tamaño de un cabello humano

Las coordenadas de traslación son las que definen la posición del origen del sistema de referencia, mismo que se mueve con el cuerpo. Las coordenadas de rotación son las que se necesitan para definir la orientación del sistema local respecto al global. En la figura 19 se muestra un cuerpo y dos sistemas de referencia:

- Sistema global o de referencia fijo ( $X, Y, Z$ ) en el exterior, al que se denominará sistema global; y
- Sistema local o móvil ( $\xi, \eta, \zeta$ ), que acompaña al cuerpo en su movimiento y que se denominará sistema local.



**Figura 19 Sistema de referencia global ( $X, Y, Z$ ) y móvil ( $\xi, \eta, \zeta$ )**

Fuente: (Pillajo Obando & Robayo Cajamarca, 2012, pág. 44)

### 2.2.9. SENSOR DE FLEXIÓN

Este tipo de sensores llevan su nombre por la capacidad de flexionarse que tienen y son unidireccionales, puesto que sólo se pueden flexionar en una dirección (Grusin, Sparkfun, 2011). A medida que se flexiona el sensor, éste arrojará un valor mayor. Los valores que resultan de estos sensores son en resistencia; es decir, que funcionan como las resistencias variables tradicionales pero con la diferencia que son alargados y se pueden adaptar a diferentes superficies según su aplicación. Esta tecnología está patentada por

Spectra Symbol y han sido usados en el guante electrónico llamado Nintendo Power Glove. (Ver figura 20).

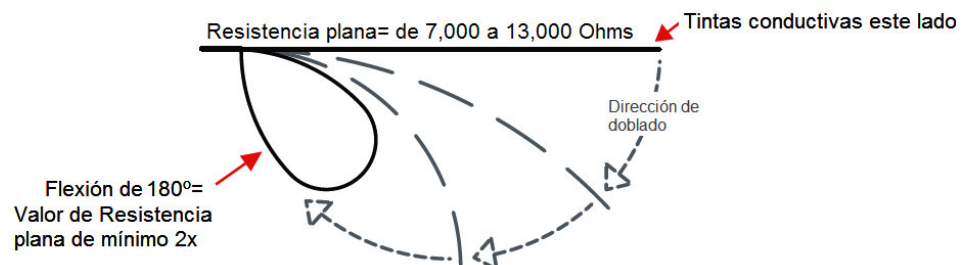


**Figura 20 Sensor de flexión**

Fuente: (Fritzing, 2015)

## FUNCIONAMIENTO DE LOS SENSORES DE FLEXIÓN

Se los debe flexionar hacia la parte activa, donde se encuentran los sensores. Estos tienen una flexión de 180°, porque son unidireccionales. En la figura 21 se indica la dirección del flexionamiento (SpectraSymbol, 2013).



**Figura 21 Funcionamiento del sensor de flexión**

Fuente: (SpectraSymbol, 2013)

## POSIBLES USOS

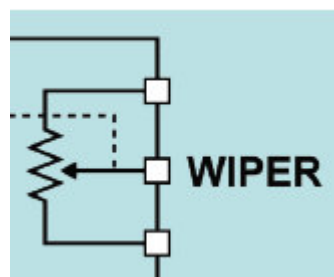
- Robótica
- Gaming <sup>5</sup>
- Dispositivos médicos
- Periféricos de una PC
- Instrumentos musicales
- Terapia física

<sup>5</sup> Juegos Virtuales



### 2.2.10. POTENCIÓMETROS DIGITALES

Los potenciómetros digitales son circuitos integrados que tienen secciones resistivas internas. Estas secciones son controladas y accedidas por una señal digital de 5V o 0V. Este tipo de potenciómetros están conformados por un divisor resistivo de  $n+1$  resistencias. En este caso, existen tantos puntos intermedios según sean requeridos para el sistema. Estos puntos son "...conectados a un multiplexor analógico que selecciona la salida" (SpectraSymbol, 2013). Estos se usan para reemplazar a los potenciómetros mecánicos<sup>6</sup> (Ver figura 22). Por tanto, cumplen la misma función que un potenciómetro mecánico e igualmente utiliza tres pines (Hernández, 2011).



**Figura 22 Terminal de ajuste, Wiper**

Fuente: (Hernández, 2011)

Existe una terminal llamada *Wiper* que se encuentra entre las dos terminales. Este elemento sirve para ajustar los valores de resistencia. A diferencia de los potenciómetros mecánicos, los potenciómetros digitales se comunican a través de una interfaz serie como SPI, I2C, Up/Down.

### FUNCIONAMIENTO

En la figura 23 se muestra el funcionamiento de los dos tipos de potenciómetros, el electromecánico y el digital. Se puede observar que el potenciómetro electromecánico varía en sus valores de resistencia, mientras

<sup>6</sup> Potenciómetro mecánico: resistor que se le puede variar el valor de su resistencia.

que el potenciómetro digital varía en sus valores de voltaje. Esto puede ser relativo según la ley de Ohm<sup>7</sup>.



**Figura 23 Funcionamiento de los Potenciómetros Electromecánicos y Digitales**

Fuente: (Hernández, 2011)

A continuación se muestra una comparación entre el potenciómetro mecánico y digital.

**Tabla 1**

**Tabla comparativa de potenciómetro mecánico y digital**

Potenciómetro Mecánico	Potenciómetro Digital
Resistencia del ajuste (Wiper) insignificante	Resistencia del ajuste (wiper) cada 100 $\Omega$
Resistencia bien controlada de extremo a extremo	De un extremo al otro depende de la resistencia. Las relaciones de la posición del ajuste (wiper) constante
Interfaz mecánica	Pueden ser controlados por microprocesadores o botones a través de las interfaces serie: I2C, SPI, Up/Down

Fuente: (Hernández, 2011)

<sup>7</sup> Ley de Ohm:  $V=R \cdot I$  (Voltaje es igual a resistencia por corriente)

## CONVERSIÓN ANÁLOGO-DIGITAL CON ARDUINO (ADC)

En la actualidad, se vive en un mundo análogo, en el que la mayoría de las cosas que medimos en general se lo hace análogamente (Gómez, 2013). Este es el caso de los sensores que se usan en el guante electrónico, por ejemplo. Este guante permite que, mediante los sensores flexoresistivos, se obtengan valores con una variación de resistencia<sup>8</sup> según como se flexionan los dedos. Es decir, a mayor flexión se obtiene mayor valor de resistencia y viceversa. Ésta es una medida análoga.

Los datos que se obtienen de una señal eléctrica analógica se caracterizan porque varían constantemente y pueden tomar cualquier valor. Se tiene que el microcontrolador de un sistema de control (en este caso Arduino) no trabaja con señales analógicas sino con señales digitales. Así, se debe transformar estas señales análogas en digitales para que el microcontrolador los pueda entender. Por esa razón se hace uso de un convertidor análogo-digital<sup>9</sup>.

La conversión análogo-digital llamada también ADC por sus siglas en inglés (*Analog to digital converter*) hace referencia a convertir señales análogas<sup>10</sup> en señales digitales. Así, se puede cuantificar una señal, la que se puede asignar un valor. Este valor se lo identifica toda vez se haya concluido y se lo representa en formato digital<sup>11</sup>. Se usarán valores de voltaje para controlar los movimientos del Dron. Lo interesante de Arduino es que cuenta con ciertos ADC, los que ayudan a convertir los niveles de voltaje a valores digitales. Para obtener datos precisos de un ADC, es necesario tomar en cuenta cual es la resolución del convertidor, la precisión de la medición, el tiempo de conversión y la frecuencia de muestreo.

---

<sup>8</sup> Los valores obtenidos son lineales.

<sup>9</sup> Módulo incluido de fábrica en Arduino.

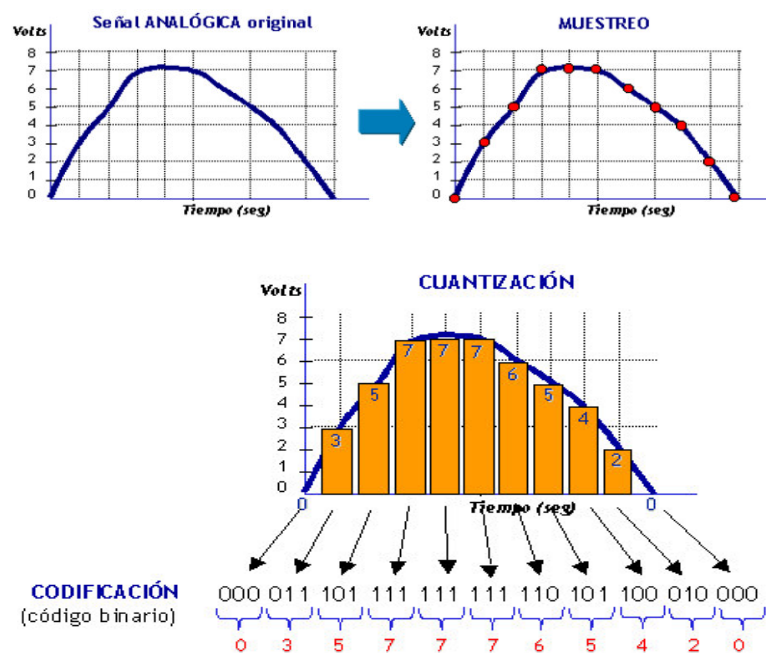
<sup>10</sup> Que pueden ser en voltaje, resistencia, corriente representadas como temperatura, presión, flexión etc.

<sup>11</sup> Las señales digitales son las que identificamos con 1's o 0's y las señales análogas tienen valores continuos.

## LA RESOLUCIÓN DEL CONVERTIDOR

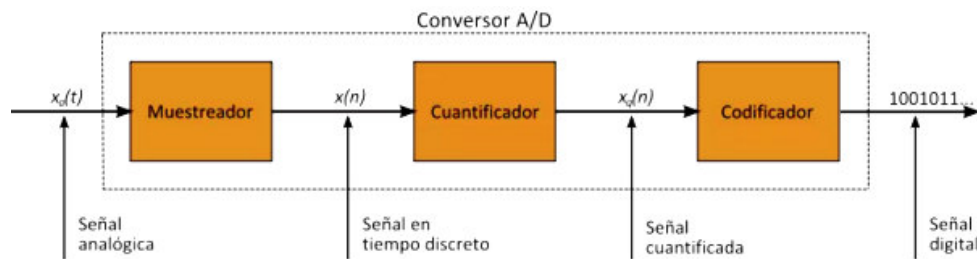
El microcontrolador de Arduino trabaja solamente con señales digitales, por esta razón se realiza una conversión de señales análogas a digitales. Esta conversión es posible mediante el proceso llamado “cuantificación”. En este proceso se convierten señales que por naturaleza son continuas a un dato discreto. Para la cuantificación antes se debe muestrear la señal (ver figura 24) (Gómez, 2013).

La resolución del convertidor se refiere a la “...cantidad de valores discretos en los que un ADC puede traducir una señal análoga a digital y se mide según el número de bits que el ADC tiene a su salida. El número de bits representa la cantidad máxima de valores discretos o de pasos que un ADC puede tener” (Gómez, 2013). En este sentido, la entrada analógica permite el cálculo codificado de cualquier valor digital. Esta medición de la relación se escribe según la cantidad N de bits existentes (Ver figura 25).



**Figura 24 Muestreo, cuantización y codificación de un ADC (de 3 bits)**

Fuente: (Rueda, 2010)



**Figura 25 Pasos conversión análogo digital**

Fuente: (Rueda, 2010, pág. 60)

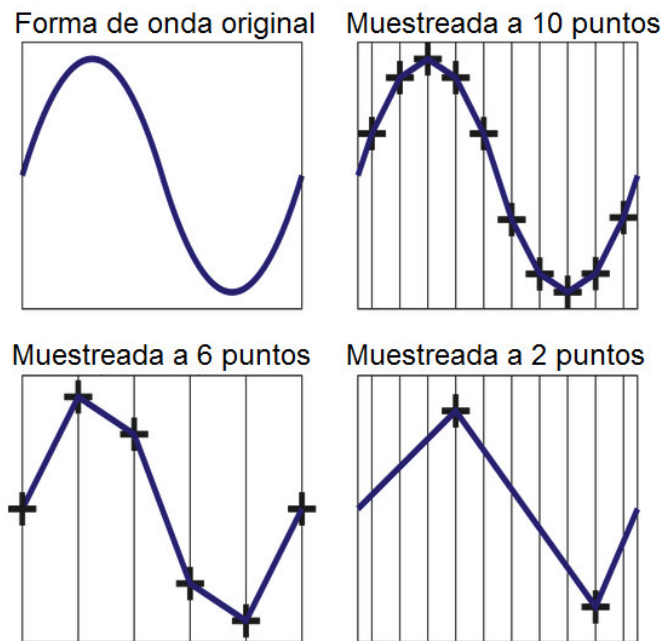
## PRECISIÓN DE LA MEDICIÓN

De acuerdo a investigaciones previas, existen dispositivos que cuentan con 8, 10 y hasta 15 bits de resolución para el ADC. Lo que indica esta medida es que cuanto mayor sea el número de bits mayor será el número de intervalos (cuantificación), y menor será el ancho del intervalo y por lo tanto mejor la precisión de la medición<sup>12</sup> (Gómez, 2013). La precisión de la señal es el ancho de este intervalo y es medido en mV (milivoltios).

## TIEMPO DE CONVERSIÓN Y FRECUENCIA DE MUESTREO

El convertidor tarda un tiempo determinado en efectuar la conversión, mismos que tiene un lapso muy corto. Dicho proceso permite un número pequeño de conversiones. Este proceso de ADC ha sido denominado frecuencia de muestreo o "*sampling rate*" por sus siglas en inglés (Ver figura 26). (Gómez, 2013).

<sup>12</sup> El número de bits necesario depende de la aplicación que se desee realizar.



**Figura 26 Frecuencias de muestreo**

Fuente: (Labtronix, s.f.)

### 2.2.11. ZIGBEE

El zigbee fue desarrollado por *Zigbee Alliance*. Este sistema fue desarrollado según el IEEE 802.15.4 de redes inalámbricas de área personal, conocido como WPAN en inglés. Entre varias empresas crearon este dispositivo con motivo de promover un estándar de comunicaciones, vía radio y bidireccional. Asimismo, el zigbee permite comunicaciones de altos niveles de seguridad con niveles mínimos de transferencia de información. Esto permite incrementar el tiempo de vida útil de sus baterías. Entre los ejemplos más visibles se pueden citar los dispositivos de domótica (*Zigbee* y Domótica), los periféricos para PC, sensores aplicados en el sector productivo o en la medicina, a más de aplicaciones para hogares, juguetería, entre otras. Por tanto, se deduce que este sistema se creó con la intención de reemplazar los sensores/actuadores individuales.

Actualmente, la utilidad de este tipo de dispositivos se ha centrado en aplicaciones de *Internet of things*, que se refiere al internet de las cosas. En

un futuro cercano, este tipo de dispositivos estarán presentes en cualquier hogar para su uso cotidiano; así como también el *Do-it-yourself-er* o hágalo usted mismo. Este tipo de dispositivos son empleados de formas diversas, especialmente en empresa de telecomunicaciones. Entre las principales se encuentran Comcast, EchoStar, Time Warner Cable, DirecTV, Charter, Rogers, Deutsche Telekom, Videocon (ZigBee Alliance, s.f.).

#### **ESTÁNDAR IEEE 802.15.4**

Con revisión aprobada al 2006, este estándar permite "...el nivel físico y el control de acceso al medio de redes inalámbricas de área personal con tasas bajas de transmisión de datos (*low-rate wireless personal area network, LR-WPAN*)" (Pillajo Obando & Robayo Cajamarca, 2012, págs. 99-100). El protocolo de este estándar pretende la obtención de costos de fabricación mucho más bajos entre las WPAN's (*Wireless Personal Area Network*). Esta particularidad fundamental permite sencillez tecnológica. Así el propósito del protocolo es "...definir los niveles de red básicos para dar servicio a un tipo específico de red WPAN centrada en la habilitación de comunicación entre dispositivos con bajo costo y velocidad" *Ibidem*. Para este fin, y por peculiaridad del modelo, se ha desarrollado la implementación de bandas libres denominadas *Industrial, Scientific and Medical*. Estas bandas han sido implementadas en condiciones no comerciales, especialmente en ámbitos como la salud, la academia, entre otros.

#### **CARACTERÍSTICAS**

- Bandas libres ISM de 2,4 Ghz, 868 Mhz (Europa) y 915 Mhz (EEUU), con velocidades entre 20 kB/s y 250 kB/s con rangos de 10m a 1200m.
- Aumenta la confiabilidad de la comunicación en redes de alta densidad, gracias a la gran cantidad de nodos que es capaz de operar. Esto garantiza que cada paquete llegue a su destino debido a que existen mayor números de rutas alternas.

- Aumenta las zonas de alcance de una red. Esto a decir verdad, ya que cuenta con un protocolo de comunicación multi-salto, que es capaz de comunicar dos dispositivos incluso si uno de ellos no se encuentra en el rango de la red, tomando en cuenta que deben existir otros nodos intermedios que los interconecten.
- Por su baja tasa de transmisión y peculiaridades propias del objeto, puede coexistir con varias redes simultáneas en la misma frecuencia, ya sean estas de *WiFi* o *Bluetooth*, sin que su desempeño se afecte.
- El ZigBee presenta un identificador de red exclusivo, por lo que pueden funcionar inclusive con 16000 redes simultáneamente, puesto que se conforma por 65000 nodos.
- Gracias a su topología de tipo malla (*mesh*), el ZigBee puede generar auto-reparación, causados durante el ejercicio de transferencia de datos. Este proceso auto-regenerativo incrementa la confiabilidad del ZigBee.

Las propiedades se sintetizan en la siguiente tabla:

**Tabla 2**

**Propiedades del estándar IEEE802.154**

Propiedad	Rango
Rango de transmisión de datos	868 MHz (20kb/s); 915 MHz (40kb/s); 2.4 Ghz (250 kb/s)
Alcance	10–50 m (según el entorno)
Latencia	Debajo de los 15ms
Canales	868/915 Mhz (11 canales). 2.4 GHz (16 canales)
Bandas de frecuencia	Dos PHY: 868/915 MHz y 2.4 GHz
Direccionamiento	Cortos de 8 bits o 64 bits IEEE
Canal de acceso	CSMA-CA y CSMA-CA ranurado

Fuente: (Maza Alverca & Berrazueta Solórzano, 2005)



## ARQUITECTURA

### MODELO DE REFERENCIA ISO/OSI

Según la Organización Internacional de Normalización ISO y el Sistema Abierto de Interconexiones, las OSI han sido definidas por el estándar IEEE 802 en las siguientes capas:

- Capa Física PHY o *Physical Layer*,
- Capa de Enlace de Datos o MAC o *Medium Access Control* (Pillajo Obando & Robayo Cajamarca, 2012, págs. 100-101).

Las demás capas son normalmente descritas por el consorcio industrial formado por compañías interesadas en la fabricación y uso del estándar, y no se especifican en el mismo. El modelo de referencia ISO/OSI simplificado de cinco capas define los perfiles de aplicación. Esto permite que ZigBee Alliance dirija el desarrollo de las capas superiores para el estándar IEEE 802.15.4, como se sintetiza en la tabla 3.

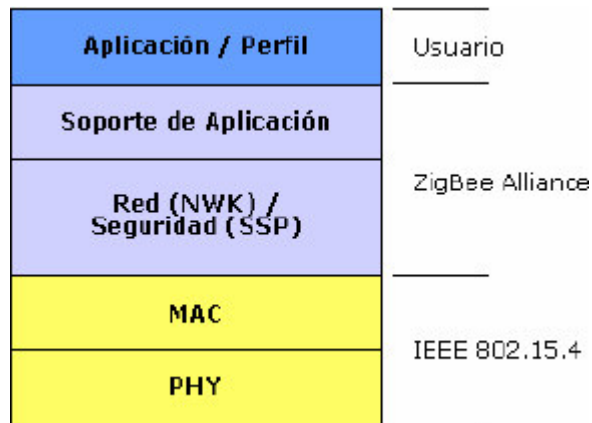
**Tabla 3**

#### Modelo ISO/OSI y Modelo IEEE802

MODELO OSI-ISO	MODELO OSI-ISO SIMPLIFICADO	MODELO IEEE 802
7. Aplicación	Aplicación	Capas superiores
6. Presentación	Perfil de aplicación	
5. Sesión		
4. Transporte		
3. Red	Red	Control de Enlace Lógico (MAC)
2. Enlace de datos	Enlace de datos	Control de acceso al medio (MAC)
1. Física	Física	Física PHY

Fuente: (Pillajo Obando & Robayo Cajamarca, 2012, pág. 101)

En este sentido, se da como resultado esta estructura donde se ven las capas que se le atribuyen a la ZigBee Alliance, mismas que son las que manejan el estándar 802.15.4 y las capas superiores. En la figura 27 se muestran las capas con que se conforma la pila de protocolos para ZigBee.



**Figura 27 Arquitectura ZigBee**

Fuente: (Marla & Moreno, 2012)

## CAPAS DE LA ARQUITECTURA ZIGBEE

### CAPA FÍSICA

Esta capa es el primer componente del modelo ISO/OSI, pues provee la interfaz con el medio físico donde ocurre la comunicación, sus principales funciones son:

- Control<sup>13</sup> de transmisor-receptor
- Detección de energía<sup>14</sup>
- Selección de canales
- Calidad del enlace
- Medición de variables
- Transmisión y recepción de los paquetes de mensajes a través del medio

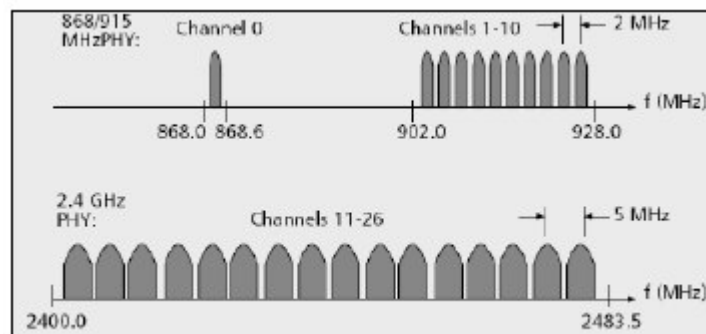
<sup>13</sup> Activación, desactivación

<sup>14</sup> Dependiendo del transmisor-receptor

- Asignación de canales

## CANALES DE OPERACIÓN

Se enumeran de 0 a 26 en tres bandas de frecuencia con un total de 27 canales, según se detalla en la figura 28.



**Figura 28 Estructura de canales de operación ZigBee**

Fuente: (Pillajo Obando & Robayo Cajamarca, 2012, pág. 102)

## CAPA DE ENLACE DE DATOS MAC (*Medium Access Control*)

Para evitar las colisiones con otros transmisores se verifica la disponibilidad del canal antes de transmitir. Para ello, el estándar IEEE 802.15.4 usa el algoritmo de Acceso múltiple con un mecanismo que evita las colisiones de datos (Pillajo Obando & Robayo Cajamarca, 2012, pág. 102). Según corresponda, éste usa una de las tres topologías disponibles: topología en estrella, en árbol o en malla (ibídem). La subcapa MAC se encarga de las siguientes funciones:

- Asociación/Disociación,
- Control de Seguridad,
- Generación de tramas de acuse de recibo (*Acknowledgment Frames*).

## **CAPA DE RED**

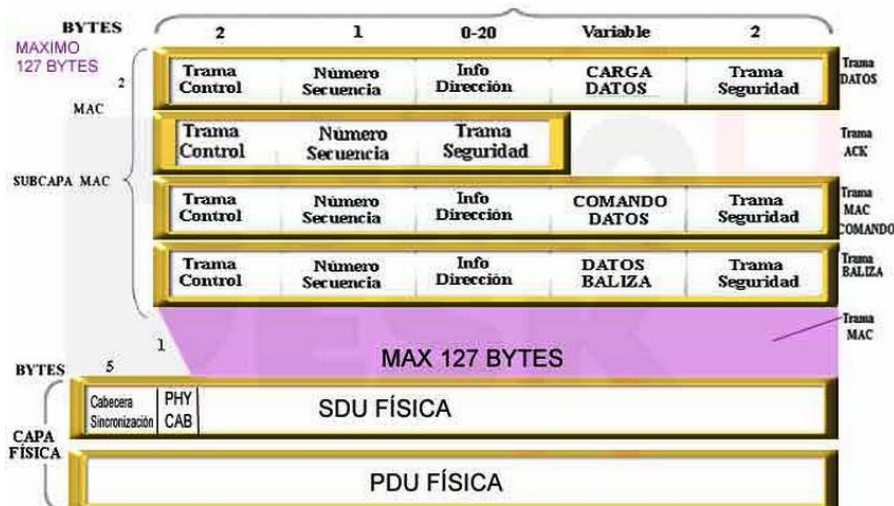
De acuerdo investigaciones previas, la capa de red presenta, principalmente, algunas actividades, tales como "...iniciar la red, unirse a la red, enrutar paquetes dirigidos a otros en la red, proporcionar a los medios para garantizar la entrega del paquete al destinatario final, filtrar paquetes recibidos, cifrarlos y autentificarlos" (Pillajo Obando & Robayo Cajamarca, 2012, pág. 103). De esta forma, se evidencia que la capa de aplicación usa la interfaz que de la red. Esto conduce a un adecuado empleo del MAC. De esta forma, otra asignación que recae en esta capa es el establecimiento de una red, desde la que se permita la asignación de direcciones a todos sus dispositivos. Seguido, toda vez se efectúe "...la función de unir o separar dispositivos a través del controlador de red [...], ejecuta la implementación...] de seguridad y encamina tramas a sus respectivos destinos (Pillajo Obando & Robayo Cajamarca, 2012, pág. 103).

## **CAPA DE APLICACIÓN**

Los ZDO (*ZigBee Device Objects*) son propios de esta capa, mismos que son los que determinan si el dispositivo actúa como coordinador, ruteador o dispositivo final. A esta capa se le considera como la aplicación que se le da a la red y como la de más alto nivel (Pillajo Obando & Robayo Cajamarca, 2012, pág. 103).

## **EMPAQUETAMIENTO Y DIRECCIONAMIENTO**

Para ZigBee se tienen cuatro tipos diferentes de paquetes, en la siguiente figura 29 se muestra lo siguiente: datos, ACK, MAC y baliza.



**Figura 29 Tipos de paquetes ZigBee**

Fuente: (Domodesk, 1999)

Según esta apreciación se puede deducir que este paquete de datos representa un total no mayor a 104 bytes de carga. Esta trama presenta su propia numeración, mismo que busca cerciorar que cada paquete arribó a su destino. Luego de ello, se espera una confirmación que el paquete no presenta error alguno. Esta confirmación recae en el campo a cargo de este proceso. Entonces, la retroalimentación receptor-emisor, proveniente del paquete de reconocimiento ACK, es una afirmación de la recepción del paquete sin error. Entre cada trama se encuentra un tiempo de silencio. Todo este proceso genera un alto patrón de confiabilidad de cada condición dentro de un proceso de transmisión.

Por tanto, este tipo de paquete MAC es implementado en procesos de configuración de nodos y dispositivos, así como para acciones a control remoto. Este proceso se da mediante una red centralizada, misma que permite generar configuración de redes a distancia y, por tanto, cumplir con el objetivo del ZigBee. En este sentido, todo dispositivo puede escuchar<sup>15</sup> a la vez que dormirse<sup>16</sup> luego de ninguna acción sonora. Este proceso es

<sup>15</sup> Recibir datos o información

<sup>16</sup> Está a la espera de ser activado por un Router o Coordinador Zigbee.

denominado despertar<sup>17</sup> y se encuentra a cargo del paquete baliza, mismo que permite la sincronización de los nodos de los dispositivos con un bajo uso de batería.

## DISPOSITIVOS QUE CONSTITUYEN UNA RED ZIGBEE

En base al trabajo de Pillajo y Robayo, se los puede definir los dispositivos según el papel y su funcionalidad en la red.

### PAPEL QUE DESEMPEÑAN

**COORDINADOR ZIGBEE (ZC):** controlan y coordinan la red y los caminos que deben seguir los dispositivos para conectarse entre sí, existen un ZC en cada red Zigbee;

**ROUTER ZIGBEE (ZR):** interconectan los dispositivos separados en la topología de la red, además de ofrecer un nivel de aplicación para la ejecución de código de usuario;

**DISPOSITIVO FINAL (ZED, Zigbee End Device):** este dispositivo representa las principales características de la tecnología ZigBee, porque se caracteriza por el bajo costo y el bajo consumo de recursos. Se comunican con su nodo padre, Router ZigBee o Coordinador ZigBee pero no pueden transmitir información a otros dispositivos. Por esta razón la mayoría del tiempo este dispositivo está a la espera de ser activado aumentando así la duración de sus baterías. Su costo comercial es reducido gracias a que no utiliza tantos recursos de memoria (Pillajo Obando & Robayo Cajamarca, 2012, pág. 106).

### FUNCIÓN QUE DESARROLLAN EN LA RED

**DISPOSITIVO DE FUNCIONALIDAD COMPLETA (FDD, FULL-FUNCTION DEVICE):** actúan como Coordinador o Router ZigBee en una red de área personal (PAN, personal area network), conocidos también como nodo activo.

**DISPOSITIVO DE FUNCIONALIDAD REDUCIDA (RFD, REDUCED-FUNCTION DEVICE):** constituyen los sensores de la red, conocidos como nodos pasivos. Se comunican solamente con los dispositivos FFD, por su capacidad y funcionalidad limitada, no pueden ser coordinadores (Pillajo Obando & Robayo Cajamarca, 2012, pág. 106).

---

<sup>17</sup> Activar el dispositivo

## MODOS DE FUNCIONAMIENTO DE ZIGBEE

ZigBee se caracteriza por el bajo consumo de sus nodos. Esto ocurre en gran medida gracias a que permanece dormido la mayor parte del tiempo, tanto los nodos activos como pasivos. Para cuando se requiere usarlo, el módulo ZigBee despierta en aproximadamente 15ms, a lo que se puede decir que se trata de un tiempo casi nulo. Esta característica lo hace eficiente y asimismo para volver a dormirse.

## ESTRATEGIAS DE CONEXIÓN EN UNA RED ZIGBEE

Existen dos formas de realizar una conexión en una red ZigBee, tanto con balizas y sin balizas, según el siguiente detalle:

**CON BALIZAS:** *se usan para poder sincronizar todos los dispositivos que conforman la red, y les permite saber cuándo transmitir identificando la red domótica y describiendo la estructura de la supertrama<sup>18</sup>. Es un mecanismo de control del consumo de potencia en la red. Los intervalos de las balizas son asignados por el coordinador de red y varían desde los 15ms hasta los 4 minutos.*

**SIN BALIZAS:** *se usa el acceso múltiple al sistema ZigBee en una red punto a punto cercano. Donde, cada dispositivo es autónomo, pudiendo iniciar una conversación, en la cual los otros pueden interferir. A veces, ocurre que el dispositivo destino no oye la petición, o el canal está ocupado (Pillajo Obando & Robayo Cajamarca, 2012, pág. 106).*

## ACCESO AL MEDIO

Para acceder al medio físico se debe emplear el uso de CSMA/CA como recurso. En este sentido, “[l]as redes que no utilizan las balizas hacen uso de una variación del mismo basada en la escucha del medio, temporizada por un algoritmo de *backoff*<sup>19</sup>, salvo en el caso de las confirmaciones (ACK, *Acknowledgement*) (Pillajo Obando & Robayo Cajamarca, 2012, pág. 108).

---

<sup>18</sup> Una supertrama está formada por dieciséis slots de igual capacidad, que pueden dividirse en una parte activa y otra pasiva, en la que el coordinador puede ahorrar energía ya que no tendrá que realizar labores de control.

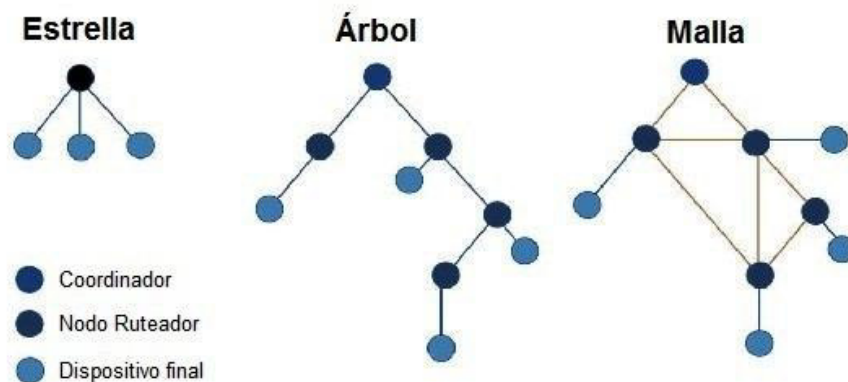
<sup>19</sup> Tiempo de espera exponencial aleatorio

En algunos casos, estos mensajes de confirmación pueden ser opcionales. El éxito del envío se verifica con la recepción de una confirmación. No obstante, no es confirmado su recibimiento cuando el dispositivo no logra completar una trama. “Pueden realizarse reintentos basados en *timeout*<sup>20</sup> un cierto número de veces, tras lo cual se decide si seguir intentando o dar error de transmisión” (Pillajo Obando & Robayo Cajamarca, 2012, pág. 108).

## TOPOLOGÍA DE REDES ZIGBEE

El estándar de la tecnología ZigBee permite tres tipos de tecnologías (ver figura 30):

- Topología en estrella,
- Topología en árbol (Cluster Tree),
- Topología en malla (Pillajo Obando & Robayo Cajamarca, 2012, pág. 108).



**Figura 30 Topologías de redes ZigBee**

Fuente: (National Instruments, 2009)

<sup>20</sup> Define en segundos el tiempo que el servidor esperará por recibir y transmitir durante la comunicación. Por defecto está configurado a 300 segundos, apropiado para la mayoría de las situaciones.



## CARACTERÍSTICAS DE LOS MEDIOS DE TRANSMISIÓN DE DATOS

Dentro de tabla 4 se muestra las características de los medios de transmisión de datos Bluetooth, WiFi y ZigBee. Para que se tenga una mejor perspectiva de las razones para usar ZigBee en este proyecto.

**Tabla 4**

### Comparativa de tecnologías Wireless

	ZigBee	Bluetooth	WiFi
<b>Aplicaciones</b>	Monitorización y control	Reemplazo de cable	Web, email, video
<b>Frecuencia de Radio</b>	868 MHz, 915 MHz, 2.4 GHz	2.4 GHz	2.4 – 5 GHz
<b>Ancho de Banda</b>	20 – 250 kbps	1 Mbps	11 – 108 Mbps
<b>Dimensión de red</b>	>65000	7 - 8	30
<b>Rango de transmisión (metros)</b>	1 - 75	1 - 10	1 -100
<b>Topologías de red</b>	Estrella, mallado y árbol de clúster	Ad hoc, piconet	Point-to-point, multipoint
<b>Complejidad</b>	Baja	Media - alta	Alta
<b>Consumo energético</b>	Bajo	Medio	Alto
<b>Puntos fuertes</b>	Consumo, coste, robustez, seguridad	Coste, seguridad	Velocidad, flexibilidad

Fuente: (Maza Alverca & Berrazueta Solórzano, 2005)

## VENTAJAS DE LAS REDES ZIGBEE

Entre las principales ventajas existentes del ZigBee, se pueden listar las siguientes:

**Conexiones punto-punto y punto-multipunto; direccionamiento de información y el refrescamiento de la red; óptimo para redes de baja tasa de transferencia de datos; economía y construcción sencilla; bajo consumo de energía, reduce tiempos de espera en el envío y recepción de paquetes, consta de más de 65000 nodos distribuidos en subredes de 255 nodos, soporte de múltiples topologías de red, operación en banda libre de ISM 2.4 GHz para conexiones inalámbricas, y alojamiento de 16 bits a 64 bits de dirección extendida (Pillajo Obando & Robayo Cajamarca, 2012, pág. 108).**

## DESVENTAJAS DE LAS REDES ZIGBEE

Entre las principales desventajas existentes del ZigBee, se pueden listar las siguientes:

**Tasa de transferencia es muy baja; menor cobertura por ser un tipo de red inalámbrica tipo WPAN; manipula textos pequeños comparados con otros estándares inalámbricos; es incompatible con Bluetooth, porque no llegan a tener las mismas tasas de transferencia, ni la misma capacidad de soporte para nodos (Pillajo Obando & Robayo Cajamarca, 2012, pág. 108).**

## APLICACIONES ZIGBEE

En la siguiente figura 31 se muestra las diferentes aplicaciones que tiene ZigBee:



**Figura 31 Aplicaciones ZigBee**

Fuente: (Marla & Moreno, 2012)

## 2.2.12. PROTOCOLOS DE COMUNICACIÓN

### 2.2.12.1. COMUNICACIÓN SERIAL

En electrónica se usa un protocolo de comunicación para envío de datos con el fin de poder interrelacionar o comunicar circuitos electrónicos embebidos, procesadores o circuitos integrados. Se puede decir que existen varios tipos de protocolos para intercambio de datos de sistemas embebidos como RS-232, RS-485, Serial, Ata, Pcie, Express o Firewire pero los más conocidos o usados actualmente son los USB (*Universal serial/bus*) y *Ethernet* (Jimbo, s.f.).

En general, se los puede dividir en grupos sincrónicos u asincrónicos. La característica principal de la interfaz serial síncrona es que ésta usa señal de reloj<sup>21</sup> común para compartir sus datos. Comúnmente, esto logra una comunicación más rápida, además de aumentar al menos un cable extra de conexión. La comunicación I2C y SPI son un claro ejemplo de interfaces seriales *síncronas*. En vista de sus beneficios, se considera importante su aplicación en este trabajo de investigación de guante electrónico. Se incluye información de estas más adelante.

Por otro lado, una comunicación asíncrona se refiere a datos transferidos sin necesidad de una señal de reloj externa. Por tanto, se disminuye el cable extra, poniendo mayor énfasis para una mejor y confiable transmisión/recepción de datos. Este protocolo sin señal de reloj externa se usa en módulos GPS, Bluetooth, Xbee entre otros. En algunas ocasiones se puede referir a los puertos serie como UART. “La UART (*universally asynchronous receiver/transmitter*) es una unidad que incorporan ciertos procesadores, encargada de realizar la conversión de los datos a una secuencia de bits y transmitirlos o recibirlos a una velocidad determinada” (Pillajo Obando & Robayo Cajamarca, 2012, pág. 108). Esta clase de comunicación de tipo serial es conocida como transistor-transistor logic o por sus siglas TTL. “Esto significa que la comunicación se realiza mediante

---

<sup>21</sup> La señal de reloj es aquella que sirve para coordinar las acciones de varios circuitos, también llamada pulsos de reloj. Mediante la frecuencia de reloj se a entender el ritmo con el que se transmiten los datos.

variaciones en la señal entre 0 voltios y  $V_{cc}$  (donde  $V_{cc}$  suele ser 3.3V o 5V). Por el contrario, otros sistemas de transmisión emplean variaciones de voltaje de  $-V_{cc}$  a  $+V_{cc}$  [por ejemplo, los puertos RS-232 típicamente varían entre -13V a 13V]” (Pillajo Obando & Robayo Cajamarca, 2012, pág. 108).

## **REGLAS DE COMUNICACIÓN SERIAL**

Existen ciertas reglas y mecanismos incorporados en esta comunicación, tales como:

- Bits de datos
- Bits de sincronización
- Bits de paridad
- Baud rate o Velocidad de Transmisión

El protocolo serial se puede configurar de algunas maneras; pero lo que se debe prestar atención es que ambos dispositivos deben ser configurados para usar exactamente los mismos protocolos en la interfaz de comunicación<sup>22</sup>.

## **BAUD RATE O VELOCIDAD DE TRANSMISIÓN**

El Baud Rate establece la velocidad de transmisión en la comunicación. Por tanto, determina qué tan rápido se enviarán los datos en una interfaz serial. Esto se lo expresa en bits-por-segundo (bps). Asimismo, éste determina cuanto tiempo el transmisor mantiene en alto/bajo la interfaz serial o a que período<sup>23</sup> el dispositivo receptor muestrea en su interfaz. Se puede hallar el valor de cuánto tiempo se demora en transmitir un bit, si se invierte el valor del Baud Rate.

El requerimiento principal para los valores a los que se puede configurar el Baud Rate no tiene limitaciones (dentro de lo real). No obstante,

---

<sup>22</sup> Se debe tener exactamente los mismos parámetros en la interfaz para asegurarse que exista la comunicación.

<sup>23</sup> Período se refiere a la cantidad de tiempo a la que se muestrea la señal.

es imprescindible tener este valor en las interfaces de ambos dispositivos receptor/transmisor. Algunos valores estándar son 1200, 2400, 4800, 19200, 38400, 57600 y 115200; aunque y especialmente para cosas simples, donde la velocidad no es crítica, se puede usar 9600 bps. Si el valor es más alto la transmisión/recepción será más rápida; pero existen límites en que tan rápidos pueden ser. Por ejemplo, no es usual poner más de un valor de 115200, porque para la mayoría de microcontroladores es demasiado rápido y pueden presentar tener errores en la recepción.

## PAQUETE DE DATOS

En la figura 32 se muestra un bloque de transmisión de datos (usualmente un byte). Este se envía en un paquete o *frame*<sup>24</sup> de bits. Los paquetes están compuestos por bits de sincronización y bits de paridad.



**Figura 32 Paquete de datos comunicación serial**

Fuente: (Jimbo, s.f.)

## BITS DE DATOS

En esta parte del paquete de datos se envía la información que se necesita transmitir y puede ser configurada de 5 a 9 bits. Ciertamente, el número de bits estándar es de 8 bits (un byte<sup>25</sup>). Una vez que se define el número de bits que se enviarán, se define también el formato según el cual los datos serán recopilados y acopiados. Así, estos se envían desde el menos significativo (*lsb, least-significant*) al más significativo (*msb, most-significant*) o viceversa.

<sup>24</sup> Frame of bits: significa paquete de datos

<sup>25</sup> Un byte es 8 bits.

De no especificarse el tramo de envío, usualmente se asume que van desde el bit menos significativo.

### **BITS DE SINCRONIZACIÓN**

Se caracterizan por ser los bits de inicio y parada de una transmisión de datos. Estos indican cuando se comienza y detiene cada paquete de datos. Existe siempre un bit de inicio, pero se pueden presentar uno o hasta dos bits de parada. El bit de inicio va del estado de 1 a 0, mientras que el bit de parada se mantiene en el estado de 1.

### **BITS DE PARIDAD**

La paridad es una manera muy simple de bajo nivel de detección de errores. Se lo define como par o impar. Para verificar el bit de paridad se envían los datos (5-9 bits) y la igualdad de la suma de estos bits decide si el bit es configurado o no. Por ejemplo, asumiendo una paridad que es configurada en par y fue añadida al byte de datos de 0b01011101, el cual tiene un número impar de 1s (5), entonces, el bit de paridad debería ser configurado a 1. Inversamente, si el modo de paridad fue configurado a impar, el bit de paridad debería ser 0.

Un dispositivo transmitiendo el carácter ASCII O y K tendría que crear dos paquetes de datos. El valor ASCII de O es 79, el cual se convierte en un valor binario de 8 y sería 01001111, mientras el valor de K sería 01001011. Todo esto es anexado a los bits de sincronización. No está especificado, pero se asume que los datos son transferidos primero con el bit menos significativo. Nótese en la figura 33 que cada uno de los dos bytes son enviados como se lee de derecha a izquierda.



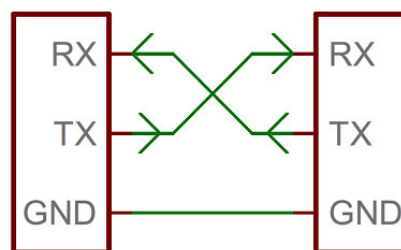
**Figura 33 Envío de dos bytes, comunicación serial**

Fuente: (Jimbo, s.f.)

Si se toma el valor de velocidad de transmisión de 9600 bps, el tiempo que se mantiene enviando en alto o bajo esos bits sería  $1/(9600)$  bps o  $104\mu\text{s}$  por bit. Por cada byte de datos transmitido hay 10 bits siendo enviados, de los cuales se distribuyen "...un bit de inicio, 8 bits de datos y un bit de parada" (Pillajo Obando & Robayo Cajamarca, 2012, pág. 108). Entonces, a 9600 bps se envían 9600 bits por segundo o 960 (9600/10) bytes por segundo.

## HARDWARE DE LA COMUNICACIÓN SERIAL

Para la comunicación serial se necesitan dos cables. El primero es concerniente para la transmisión TX y otro para recepción RX, según se describe en la figura 34.



**Figura 34 Hardware de la comunicación serial**

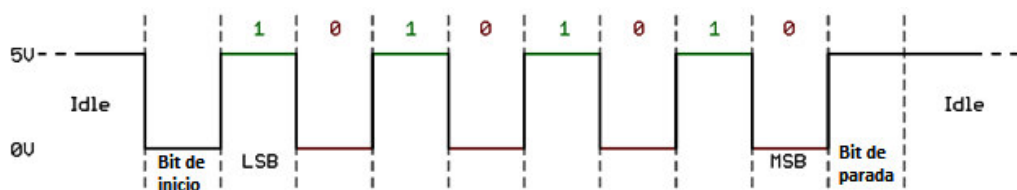
Fuente: (Jimbo, s.f.)

En la figura precedente se puede apreciar que el pin RX de un dispositivo se relaciona con el pin TX del otro dispositivo y viceversa. Una interface serial puede funcionar enviando y recibiendo datos. Esta puede ser de manera *full-*

*duplex* o *half-duplex*. El *Full-duplex* se refiere a que ambos dispositivos pueden enviar datos simultáneamente. Por otro lado, el *half-duplex* quiere decir que cada dispositivo debe esperar su turno para enviar y recibir datos. También existen dispositivos que son <<todo oídos>> y no necesitan comunicarse con su transmisor, puesto que solamente reciben datos. A esta manera de transmitir se la conoce como *simplex*.

## IMPLEMENTACIÓN DEL HARDWARE

Cuando el microcontrolador precisa de enviar datos a un dispositivo, mediante la comunicación serial, lo hace por medio de un nivel TTL (*transistor-transistor logic*). Estas señales existen usualmente en el rango de 0V a 3.3V o 5V. Además, pueden indicar una línea libre, un bit de valor 1 o un bit de parada. El GND o tierra puede representar un bit de inicio o un bit de datos de valor 0 (Ver figura 35).



**Figura 35 Implementación del Hardware**<sup>26</sup>

Fuente: (Jimbo, s.f.)

### 2.2.12.2. COMUNICACIÓN SPI (SERIAL PERIPHERAL INTERFACE)

De acuerdo a Grusin, el bus de interfaz de periféricos serie, conocido por sus siglas en inglés como SPI o *Serial Peripheral Interface*, es una interfaz serial utilizada para enviar datos entre microcontroladores y pequeños periféricos

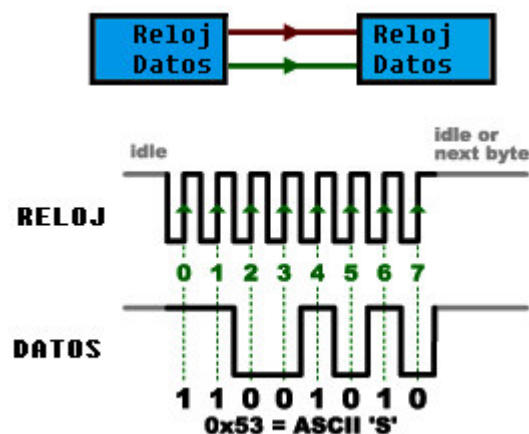
<sup>26</sup> Se asume que la comunicación empieza desde el bit menos significativo



como sensores, tarjetas SD u otros dispositivos. Estos presentan las siguientes características:

- Usa una línea de señal de reloj separada,
- Una línea para datos entrantes y salientes,
- Una línea de selección (*chip select*), para escoger el dispositivo con el que se desea comunicar.

A diferencia de la comunicación serial, el SPI es una comunicación síncrona que utiliza una línea de señal de reloj separada. Esta línea es la que lleva la sincronización perfecta de la transmisión mediante pulsos de reloj que le dicen al receptor cuando exactamente muestrear los bits en la línea de datos. Los dispositivos pueden leer estas señales de reloj en diferentes formas. Estas señales pueden ser cuando el reloj oscila en el flanco de subida (de bajo a alto) o en el flanco de bajada (de alto a bajo), según se configure el protocolo o especifique el datasheet<sup>27</sup>. Cuando el dispositivo receptor detecta este dato, inmediatamente mira la línea de datos para leer el siguiente bit. (Ver figura 36).



**Figura 36 Comunicación SPI configurada en flanco de subida**

Fuente: (Grusin M. , 2010)

<sup>27</sup> Datasheet es una hoja de datos, en la que se describe las características técnicas de los dispositivos

En esta se envía el dato de 0x53 desde el bit menos significativo. Así, se envía el valor más bajo 0011=0x3 y el valor más alto 0101=0x5. Por esta razón, queda de esta manera 11001010.

## RECEPCIÓN DE DATOS

En la comunicación SPI existe un solo dispositivo (que normalmente es un microcontrolador). Esta genera la señal de reloj para sincronizar los datos de envío. A este dispositivo se lo conoce con el nombre de *master* (CLK, SCK o *serial clock*). Asimismo, al dispositivo que recibe estos datos se lo conoce con el nombre de *slave* (esclavo). Se puede tener varios esclavos pero un solo máster.

El proceso cuando se envían los datos desde el *master* al *slave* se lo denomina con las siglas en inglés MOSI (*Master Out Slave In*). Al contrario cuando se envían datos desde el dispositivo *Slave* al *Master* se lo denomina con las siglas en inglés MISO (*Master In Slave Out*). Nótese de mejor manera en la figura 37, en donde se indica que el SPI es *full duplex*. Éste tiene dos líneas de datos. Por un lado, la primera se emplea para el envío y, la otra, sirve para recepción de datos. Por esta razón, en algunas ocasiones se puede recibir y enviar datos al mismo tiempo.

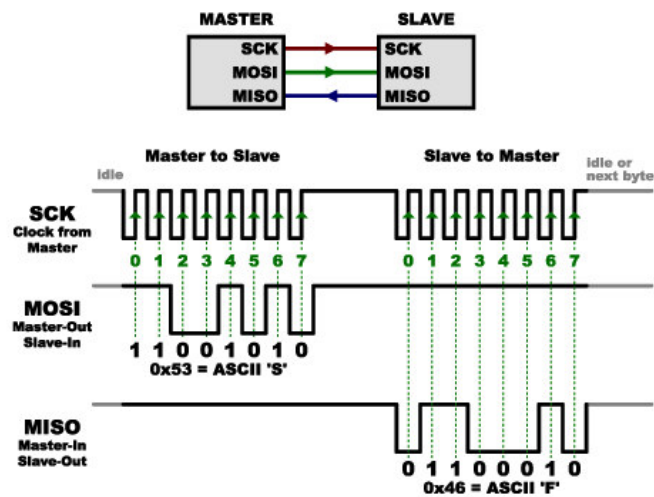


Figura 37 Comunicación SPI, líneas MOSI y MISO

Fuente: (Grusin M. , 2010)

## SELECCIÓN DEL ESCLAVO (*SS*, *SLAVE SELECT*)

Para escoger o seleccionar el dispositivo con el que se desea comunicar se usa la línea *SS* o *Slave Select*, que despierta al dispositivo para que reciba o envíe datos, así como se puede seleccionar al dispositivo que se desee dentro de un grupo de dispositivos. La línea *SS* normalmente se activa en bajo (estado en un 0 'cero' lógico). Por tanto, al momento que el dispositivo máster envía información hacia el esclavo, la línea de datos se activa y por ende activa al dispositivo esclavo. Al contrario, al momento que el dispositivo máster termina de enviar la información, la línea de datos vuelve al estado "alto" (estado en un 1 'uno' lógico) y se mantiene en este estado hasta que el Master quiera enviar información (Grusin M. , 2010). (Ver figura 38).

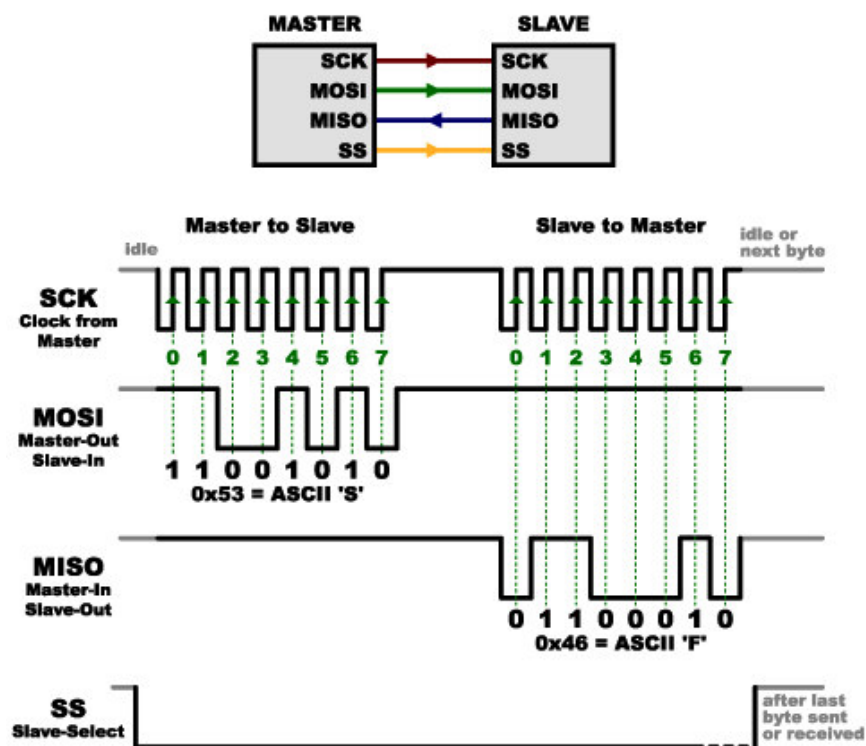
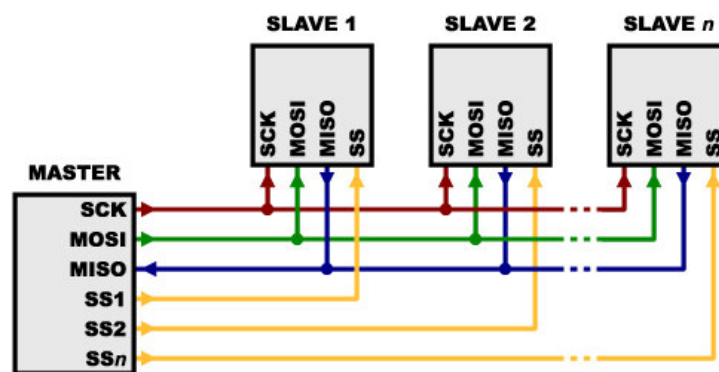


Figura 38 Comunicación SPI, línea *SS* (*Slave Select*)

Fuente: (Grusin M. , 2010)

## MÚLTIPLES ESCLAVOS

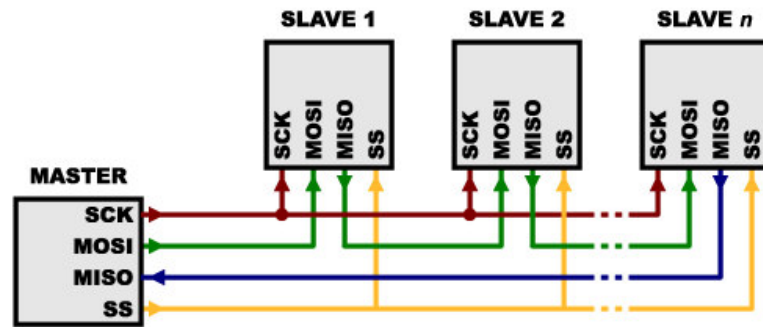
La comunicación SPI tiene la característica de poder conectar múltiples esclavos y se lo puede hacer de dos maneras. En la primera, cada esclavo necesitará una línea SS (*Slave Select*) separada. Para comunicarse con un esclavo en particular se debe activar solamente la línea SS de este esclavo en bajo y mantener el resto de líneas SS en alto. Se debe activar un esclavo a la vez, caso contrario puede existir errores en la recepción de datos. (Ver figura 39).



**Figura 39 Múltiples esclavos, las líneas SS (*Slave Select*) separadas**

Fuente: (Grusin M. , 2010)

En el segundo caso, todos los esclavos van conectados a una misma línea SS (*Slave Select*). Todos los esclavos están conectados en cadena margarita o por sus siglas en inglés *daisy-chained*. También, la salida MISO de cada uno de los esclavos va conectada a la entrada MOSI del siguiente esclavo y así sucesivamente. Por el hecho de tener una sola línea SS para todos los esclavos, los dispositivos son activados simultáneamente y una vez enviados los datos, la línea SS se pone en estado alto. (Ver figura 40).



**Figura 40 Múltiples esclavos, una sola línea SS (*Slave Select*)**

Fuente: (Grusin M. , 2010)

Nótese que, para este formato, los datos fluyen desde un esclavo al siguiente. Entonces, para enviar una orden a cualquier esclavo se debe transmitir suficientes datos para alcanzar a *todos* los dispositivos esclavos. Este formato generalmente es usado en los casos en que no se necesita recibir una retroalimentación o algún dato de retorno, por ejemplo cuando se usan leds.

## VENTAJAS

- Soporta múltiples esclavos,
- Es más rápido que la comunicación serial asíncrona,
- El hardware que recibe los datos puede ser un simple registro de desplazamiento.

## DESVENTAJAS

- Requiere más líneas de señal que otras comunicaciones seriales;
- El dispositivo maestro debe controlar todas las comunicaciones (los esclavos no se pueden comunicar directamente el uno al otro);
- Usualmente requieren líneas SS separadas para cada esclavo, lo cual puede ser problemático si se necesitan varios dispositivos esclavos;
- La comunicación debe ser bien definida; mas no se puede enviar cantidades aleatorias de datos.

### 2.2.12.3. COMUNICACIÓN I2C (INTER-INTEGRATED CIRCUITS)

Según Sparkfun, la comunicación I2C fue desarrollada por la empresa Phillips en el año 1982 con la finalidad de comunicar una PC con los diferentes periféricos. La idea fue simplificar las conexiones entre periféricos y controlar sistemas de audio y video. Actualmente, empresas como Siemens, Intel, TI, Atmel, Maxim, Xicor, entre otras, diseñan dispositivos basados en este protocolo I2C (Sparkun, 2011). El Bus I2C, en principio, precisa exclusivamente de dos líneas de señal, además de una en común. En este sentido su comunicación es sincrónica, así como se realiza en serie. Así, una de las señales es implementada para intercambiar datos, a diferencia de la otra que sirve como para marcar el tiempo.

Originalmente, éste fue desarrollado para comunicaciones dentro del espectro de 100KHz y provisto solamente de direcciones de 7 bits; limitando el número de dispositivos en el bus a 112. En 1992, fue añadido el *Fast-mode* a 400KHz y se expandió con espacio para direcciones de 10 bits. En la actualidad este tipo de dispositivos pueden transmitir valores mayores. (Ver figura 41).

#### CARACTERÍSTICAS

- Está presenta dos hilos, conocidas como líneas que conectan dispositivos;
- Controlada por una señal de reloj común, la comunicación se transmite de forma síncrona;
- Se transmiten dos señales, una por cada línea. Posee una comunicación serial, bit a bit, *half-duplex*.

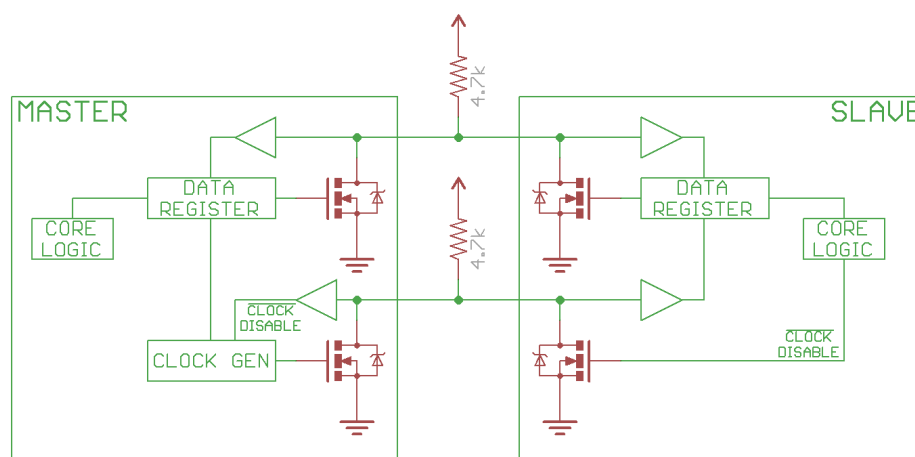
#### VELOCIDAD DE TRANSMISIÓN

- Estándar: hasta 100Kbits/s,
- *Fast*: hasta 400 Kbits/s,

- *High-speed*: hasta 3,4 Mbits/s (Sparkun, 2011).

## LÍNEAS DE COMUNICACIÓN

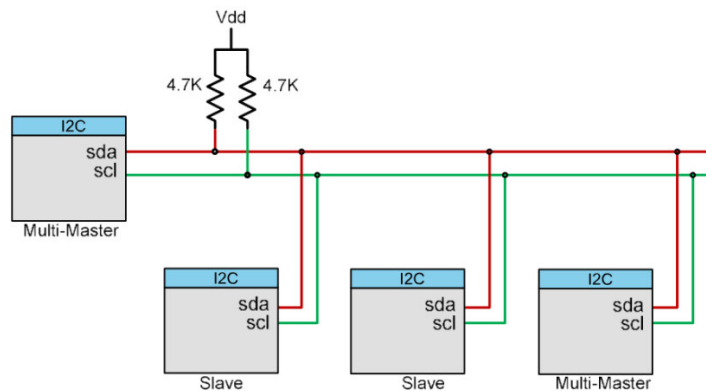
- **SDA (*Serial Data Line*)**, línea para la transferencia serial de datos;
- **SCL (*Serial Clock Line*)**, señal de reloj utilizada para la sincronización de los datos (Sparkun, 2011).



**Figura 41 Conexión I2C, Master/Slave**

Fuente: (Sparkun, 2011)

En este protocolo existe la comunicación del tipo maestro/esclavo (*master/slave*), que es la que maneja dispositivos conectados al bus I2C; también soporta protocolo multimaestro y cada uno de los dispositivos tiene una dirección única. Las funciones del maestro y del esclavo se definen cuando “[e]l circuito maestro inicia y termina la transferencia de información además de controlar la señal de reloj. Normalmente es un microcontrolador. [Además, e]l esclavo es el circuito direccionado por el maestro” (Sparkun, 2011).



**Figura 42 Topología I2C, Master/Slave**

Fuente: (Semiconductor, 2012)

En la figura 42 se muestra la topología I2C. Una de las ventajas de I2C es que se puede incorporar varios másteres, aunque sólo uno de ellos tendrá el control a la vez. La capacidad máxima de los dispositivos que se pueden conectar al bus no debe superar los 400pF y no superar una distancia de 2 o 3 metros. Además, las líneas de SDA y SCL están conectadas a tensión positiva de alimentación con resistencias de *Pull Up*<sup>28</sup>. Los dispositivos maestros o esclavos pueden ser transmisores o receptores. Esto depende de la función del dispositivo debido a que la línea SDA es bidireccional. El bus I2C se considera libre si las líneas SDA y SCL se encuentran en estado lógico uno como resultado de la presencia de las resistencias de *Pull-up*.

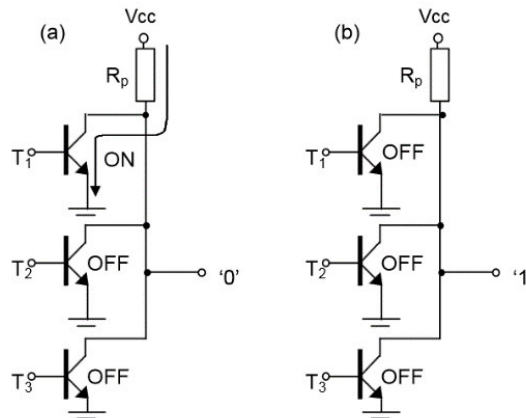
## HARDWARE DEL BUS I2C

En la figura siguiente se muestra el Hardware del Bus I2C. Entonces, toda vez que dos dispositivos son registrados simultáneamente predominan los ceros. Así, se aplica lo que se denomina como AND cableada. “Si un dispositivo escribe un nivel alto pero lee un cero indica que otro dispositivo está también accediendo al bus” (Pillajo Obando & Robayo Cajamarca, 2012, pág. 93). A

<sup>28</sup> Las resistencias de Pull Up son importantes en la comunicación I2C ya que son resistencias normales que cumplen la función de mantener el voltaje requerido sobre las líneas SDA y SCL, se conectan entre una señal lógica y el positivo. No permiten que existan valores entre cero y uno con la finalidad de que no se confunda su estado.



continuación se describen las características de una conexión de colector abierto (TTL) o drenador abierto (CMOS). (Ver figura 43).



**Figura 43 AND cableada. (a) Una o más salidas a '0'. (b) Todas las salidas a '1'**

Fuente: (Electrónica Digital, 2010)

Dependiendo del estado del transistor de salida de cada dispositivo, la And Cableada<sup>29</sup> suele resultar en algunos de estos dos casos:

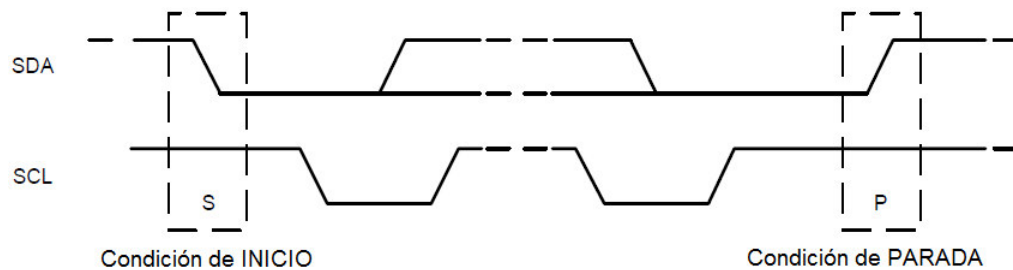
1. Que el transistor esté saturado, con lo cual lleva a nivel bajo o "0", a la línea correspondiente, independiente del estado de los otros transistores. Es decir, el bus está ocupado a nivel bajo.
2. Que el transistor esté en corte (estado de alta impedancia) con lo cual el estado de la línea depende de los otros transistores. Así, el bus está libre y, de no haber ningún otro transistor saturado, la línea estará alta mediante la resistencia de Pull-up, misma que está conectada a la alimentación.

## CONDICIONES DE INICIO (S) Y PARADA (P)

Cuando las señales SDA y SCL están en estado lógico alto significa que la condición inicial es de bus libre. Esto quiere decir que los transistores de salida

<sup>29</sup> La función AND cableada resulta de particular interés cuando se deben combinar muchas entradas, pues se elimina la necesidad de disponer de puertas de muchas entradas. En todos los circuitos de AND cableada se requiere una resistencia externa.

de todos los dispositivos conectados al bus I2C deben estar en alta impedancia (Coquet, 1999). (Ver figura 44).



**Figura 44 Secuencia de inicio y parada**

Fuente: (Invensense, 2013, pág. 34)

### CONDICIÓN DE INICIO (S)

Cuando en estado alto se encuentra la línea de datos SCL y la SDA se encuentra baja, cualesquier maestro lo ocupa. Así, resulta la condición de *inicio (start)*. Obsérvese la figura 45:

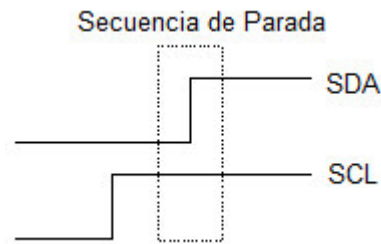


**Figura 45 Secuencia de inicio**

Fuente: (Carletti, 2007)

### CONDICIÓN DE PARADA (P)

Cuando la línea de datos SDA y SCL están en estado alto, se establece la llamada condición de *parada (stop)*. (Ver figura 46).

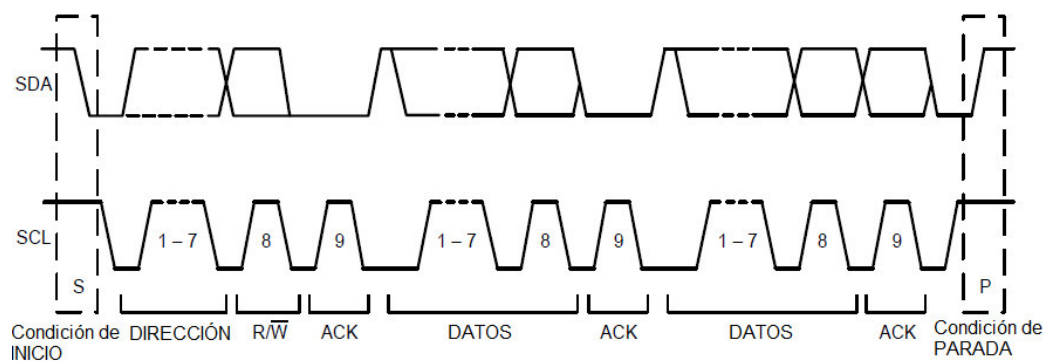


**Figura 46 Secuencia de parada**

Fuente: (Carletti, 2007)

## TRANSFERENCIA DE DATOS

Para efectuar la definición del bus I2C se ha planteado dos secuencias especiales. La primera tiene que ver la secuencia de inicio y la otra con su parada. Estas se convierten en secuencias imprescindibles para la transferencia de datos, en vista que induce a un cambio de línea de SDA a SCL cuando existe una variación de datos de cantidad baja a alta. Cada una de estas señales representa todo el proceso de las transferencias, desde su inicio hasta su conclusión frente a los esclavos, tal como se muestra en la figura 47.



**Figura 47 Transferencia de Datos del protocolo I2C**

Fuente: (Invensense, 2013, pág. 35)

Dentro de la secuencia de 8 bits se identifica una transferencia de datos. Ésta comienza por el bit de más pesado (*MSB, Most Significant Bit*). Estos bits se

colocan en la línea SDA. En efecto, la SCL cambia a condición alta. Luego que se da la respuesta de confirmación por el dispositivo correspondiente, es necesario enviar 9 pulsos de reloj por cada byte o 8 bits de datos. De esta manera, sólo cuando el estado se encuentra bajo, después de la recepción de un bit, significa que éste puede receptor otro bite.

Por otro lado, si se recibe un bit en estado alto quiere decir que no puede realizar más transferencia de datos. Por tanto, el máster termina su envío con una señal de secuencia de alto (*stop*). Esta condición obliga al dispositivo maestro a quedarse en espera. La transferencia de datos quedará reanudada toda vez el esclavo sea capaz de proseguir con la envío de más datos.

En resumen, se puede determinar los siguientes pasos para realizar una comunicación mediante el protocolo I2C:

1. Envía un bit de *start*. Éste señala el comienzo de una transferencia de datos;
  2. Se envía 7 bits, mismos que son provenientes del esclavo;
  3. Se envía 1 bit de lectura/escritura (R/W), mismo que permite establecer la cualidad de transmisor del esclavo.
  4. Se envía un bit de reconocimiento ACK (*Acknowledgment*), mismo que es enviado por el esclavo;
  5. Se envía el mensaje dividido en bytes (8 bits);
  6. Por último, se envía un bit de *stop* que indica el fin de la comunicación.
- (Pillajo Obando & Robayo Cajamarca, 2012, pág. 96)

En este sentido, es importante prestar atención a ciertas condiciones a tomar en cuenta para el adecuado funcionamiento. En inicio, se debe mencionar que los 7 bits iniciales son condición que establece la direccionalidad del esclavo. Como resultado se establece la dirección de datos por el 8 bit. De esta manera se concluye que:

- Si  $R/W=0$  entonces el esclavo trabaja como receptor y el máster escribe los datos recibidos;
- Si  $R/W=1$  entonces el esclavo trabaja como transmisor y el máster recibe o lee información.

El proceso del bus I2C comienza con el envío de 1 bit start por parte del máster, lo que alerta a los esclavos para comenzar la transferencia de datos. Como efecto, cada esclavo analiza la correspondencia de la información frente al máster, que es el que establece la dirección del dispositivo para que el esclavo reconozca dicha información. Cabe mencionar que cada dispositivo presenta su dirección de manera propia y exclusiva.

## **TIPOS DE FORMATO DE TRANSFERENCIA**

### **MAESTRO-EMISOR**

Transmite al esclavo-receptor. Si el bit 8 es de escritura ( $R/W=0$ ).

### **MAESTRO-RECEPTOR**

En este formato, el maestro recibe del esclavo inmediatamente toda información después del byte inicial, siempre y cuando el byte 8 sea de lectura ( $R/W=1$ ).

### **COMBINADO**

Este formato se refiere a cuando el maestro mantiene comunicación con un esclavo, incluso después de haber enviado la condición de *stop*. Durante este proceso, el bit  $R/W$  se invierte, el mismo momento en que se reanuda la dirección del esclavo así como el estado de inicio-*start*. A este proceso se le conoce como cambio de dirección de transferencia. Puede n cantidad de

combinaciones efectuarse, tanto de escritura como lectura, al momento de transferencia.

## APLICACIONES

- Sistemas de configuración y supervisión en servidores,
- Sistema de gestión de alimentación,
- Tarjetas chip,
- Conexión en serie de dispositivos externos a un ordenador, y
- Bus de interconexión entre dispositivos en una tarjeta o equipo. (Pillajo Obando & Robayo Cajamarca, 2012, pág. 98)

### 2.2.13. ARDUINO

Inició en Italia (Ivrea) en el año 2005. Éste es creado por Massimo Banzi y David Cuatrecasas. Arduino está basado en una plataforma de hardware y software *easy-to-use* (fácil de usar), misma que se implementa con el fin de ayuda al usuario a realizar proyectos interactivos en el campo de la electrónica (Banzi, 2008). (Ver figura 48).



**Figura 48 Logo Arduino**

Fuente: (Banzi, 2008)

Una de las ventajas significativas que presta Arduino es el *open-source* o código abierto, con acceso al código fuente, de fácil distribución y desarrollo libre. Esto permite que la persona que lo utiliza pueda orientar su desarrollo para una necesidad personal o aplicación según le convenga, libremente sin

licencias o permisos. El software se descarga de la web gratuitamente, inclusive. Así, para comunicar a Arduino al entorno existen una variedad de sensores, motores y otros artefactos que envían diferentes tipos de señales a las entradas que dispone el dispositivo.

## **CARACTERÍSTICAS**

- Programación simple
- Software código abierto
- Hardware recurso abierto
- Bajo costo
- Multiplataforma

Pese a que la mayoría de este tipo de micro-controladores se desarrolla en plataforma Windows, el Arduino corre inclusive en Macintosh OSX y GNU/Linux. Arduino es de código abierto y se encuentra disponible tanto para programadores experimentados como para principiantes. Además, este usa los ATMEGA3258P, que son microcontroladores de Atmel.

## **HARDWARE ARDUINO**

De acuerdo a la descripción de Arduino, se incluye en su hardware una diecimila, nano (se conecta con el computador usando un cable USB Mini-B), bluetooth (permite la comunicación y programación sin cables), Lily Pad, Mini, Serial y Serial Single Sided (Pillajo Obando & Robayo Cajamarca, 2012, págs. 112-118).

## **SOFTWARE ARDUINO**

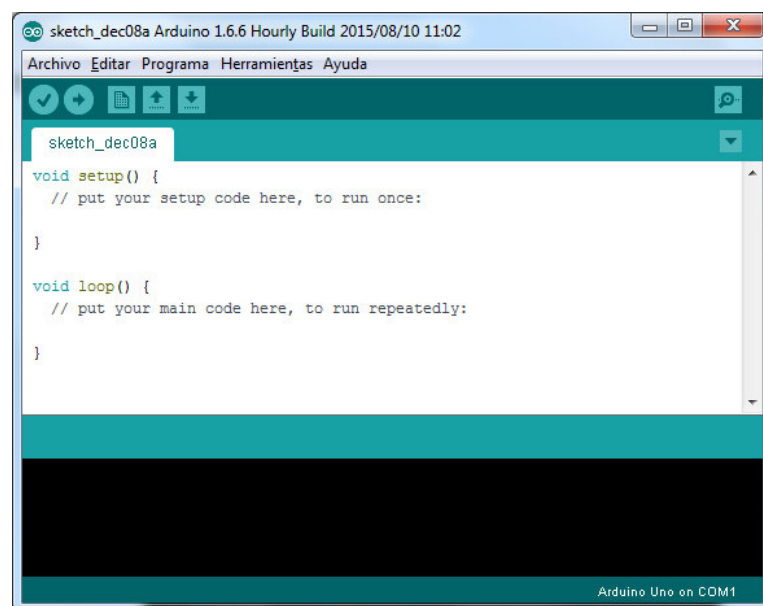
Arduino se establece de un entorno de desarrollo (IDE), además de un núcleo de bibliotecas. El IDE está escrito en Java y las bibliotecas centrales están escritas en C y C++.

## AVR libraries

Estas bibliotecas tienen el potencial de extender el lenguaje Arduino. El sistema de arduino está basado en el compilador `avr-gcc` y hace uso de las librerías `standard AVR Libc`. Estas son librerías C open-source, específicamente escritas por el *Hardware Atmel*, que es el creador de los chips sobre los que corre Arduino. Estas bibliotecas se constituyen de tres partes: `Avr-binutils`, `Avr-gcc` y `Avr-libc`. Este último es de software libre, puesto que busca elevar la calidad de almacenamiento de la biblioteca C. Su aplicación se da en el compilador `AVRGCC` sobre microcontroladores Atmel AVR.

## ENTORNO ARDUINO

El Arduino permite una familiarización rápida y óptima con cualquier usuario, ya que presenta una interfaz de fácil acceso y un menú que cuenta con las opciones: crear nuevo, abrir, guardar, parar, reproducir y exportar. (Ver figura 49).



**Figura 49 Interfaz de programación Arduino**

Fuente: (Arduino, s.f.)



## PROGRAMA

**Verify/Compile:** Verifica y compila errores en la programación.

**Incluir Librería:** Este comando añade cabeceras, según se necesite en la aplicación.

**Mostrar carpeta de programa:** Muestra las carpetas donde están las librerías de Arduino, mismas que creó el usuario o propias de Arduino.

**Añadir fichero:** Añade fichero.

## HERRAMIENTAS

**Auto Formato:** El uso de este comando da una sangría en el código, alineando las llaves de apertura y cierre del texto con las instrucciones, asegurándose que estén dentro de las llaves. Formatea el código.

**Placa:** Se usa para controlar y compilar la rutina, se selecciona la placa Arduino.

**Puerto:** Se elige los puertos reales o virtuales de la PC.

**Quemar Bootloader:** Sirve para grabar un Bootloader.

## BOOTLOADER

Es un gestor de arranque que ejecuta un programa y carga los bits y registros necesarios. Esto ocurre debido a que el microcontrolador Atmega 328P lo tiene precargado. Cuando el bootloader es energizado permite inicializar todo los recursos del sistema. Éste es el primer programa que se ejecuta para dar paso al programa principal (registros, sentencias de programa, programas de inicio). Mediante un cable USB se programa el microcontrolador Atmega 328P a través del uso del bootloader. Esto sin necesidad de usar un programador externo.

## ESTRUCTURA DE UN PROGRAMA EN ARDUINO

A continuación se presenta la estructura básica para la programación en Arduino:

### **setup()**

Llama a funciones de configuración y se realiza una sola vez. En el programa es lo primero que se ejecuta. Se inicializan o se asignan las funcionalidades, por ejemplo, cuando se asigna la función de cualquiera de los pines.

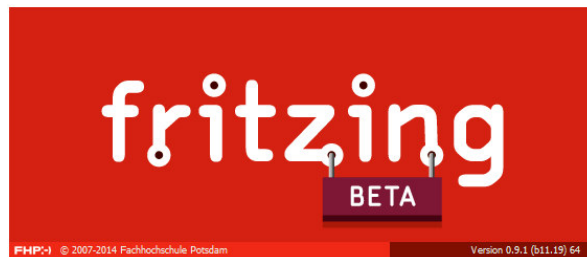
### **loop()**

Es el núcleo de los programas de Arduino. Su característica es que se ejecuta indefinidamente (bucle). Lee las entradas y activa las salidas.

#### **2.2.14. FRITZING**

Fritzing es una iniciativa de software de código abierto, ya que presenta acceso al código fuente. Es distribuido y desarrollado libremente para investigadores, desarrolladores y artistas. Es una herramienta de software creada según el principio de procesamiento y los criterios del Arduino. El Fritzing busca, entonces, "...documentar prototipos basados en Arduino y crear esquemas de circuitos impresos para su posterior fabricación..." (Postdam, 2014). El Fritzing se puede ver como una herramienta de automatización de diseño electrónico EDA. Su última versión fue actualizada en abril de 2015 y corre sobre los sistemas operativos MAC, Linux y Windows.

Para la esquematización del dispositivo maestro y dispositivo esclavo como de todos los elementos y su adhesión al guante electrónico como se puede observar en la figura 50, se ha utilizado el software gratuito *Fritzing* desarrollado por el *Interaction Design Lab* de la Universidad de Ciencias Aplicadas o *Fachhochschule de Postdam, Alemania*.



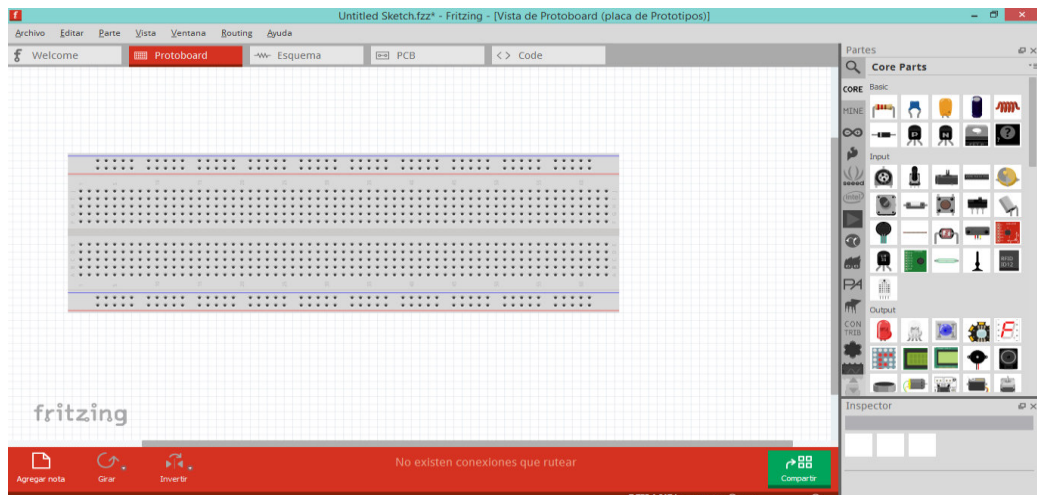
**Figura 50 Logo Fritzing**

Fuente: (Postdam, 2014)

## CARACTERÍSTICAS

- Entorno de procesamiento y Arduino
- Software de código abierto
- Software libre
- Permite crear esquemas de circuitos impresos
- Multiplataforma (Windows, Macintosh OSX y GNU/Linux)

## ENTORNO FRITZING



**Figura 51 Interfaz de desarrollo de Fritzing**

Fuente: (Postdam, 2014)

## 2.3. ANÁLISIS DE LOS MOVIMIENTOS DE LA MANO Y DEL BRAZO

### 2.3.1. LA MANO

Según el estudio de Moore and Agur (2008, págs. 5-7) "...las manos están localizadas en los extremos de los antebrazos, son prensiles y tienen cinco dedos cada una, siendo el cuarto segmento del miembro superior torácico". Por tanto, las manos se incluyen desde el inicio de la muñeca inclusive las yemas de los dedos de las extremidades superiores. Las manos se componen de huesos, tendones, músculos, ligamentos y un sistema nervioso sensorial. Dentro de la mano, el miembro más estratégico de la mano es el pulgar, en vista que "...es indispensable para realizar las pinzas pulgar-digitales con cada uno de los otros dedos, en particular el índice y también para la constitución de una toma de fuerza con los otros cuatro dedos" (Moore & Agur, 2008, págs. 5-7). Cada dedo posee dos grados de libertad en la base con excepción del pulgar, puesto que éste tiene cinco grados de libertad y dos articulaciones tipo bisagra que proporcionan los movimientos de flexión y extensión.

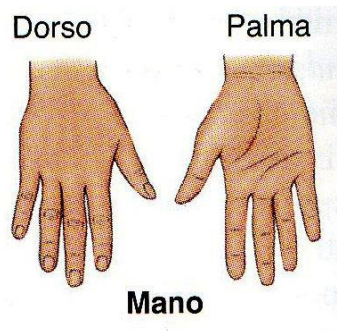
### HUESOS DE LA MANO

**Carpó:** Está compuesto por ocho huesos dispuestos en filas, proximal y distal, de cuatro huesos. Estos pequeños huesos proporcionan flexibilidad al carpo. De lateral a medial, los cuatro huesos de la fila proximal del carpo son: escafoides, semilunar, piramidal, pisiforme. Las caras proximales de la fila distal de huesos del carpo se articula con la fila proximal de huesos del carpo y sus caras distales se articulan con los huesos metacarpianos. De lateral a medial, los cuatro huesos de la fila distal del carpo son: trapecio, trapezoide, capitado o grande, ganchoso.

**Metacarpo:** Constituye el hueso de la palma entre el carpo y las falanges. Está constituido por los cinco huesos metacarpianos. Cada metacarpiano consta de una base, un cuerpo y una cabeza. Cada dedo posee tres falanges (proximal, media y distal), excepto el primero (pulgar) que tiene sólo dos (proximal y distal). Esto permite que la mano tenga un alto número de grados de libertad, permitiéndole múltiples configuraciones de aprensión y manipulación, por poseer articulaciones tipo bisagra.

### MOVIMIENTOS DE LA MANO

Según Morey & Agur (2008, págs. 5-7), la anatomía humana es peculiar y amplia, por lo que es imprescindible identificar cada parte del esqueleto para vacilar errores. “Esta nomenclatura permite la comunicación precisa entre los profesionales sanitarios de todo el mundo, así como entre estudiantes de ciencias de la salud, básicas, aplicadas y otros” (Moore & Agur, 2008, págs. 5-7). En la figura 52 se muestran las partes de la mano:



**Figura 52 Partes de la mano**

Fuente: (Moore & Agur, 2008, págs. 5-7)

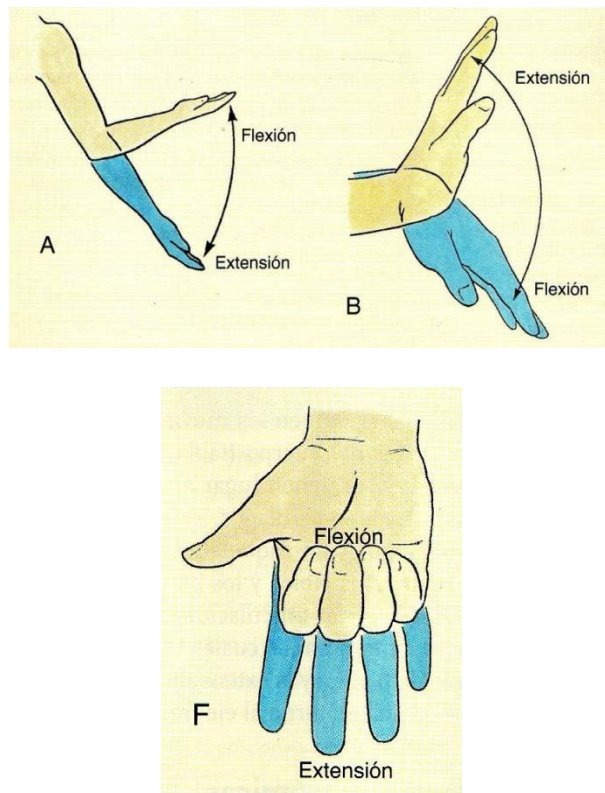
**Dorso:** Hace referencia a la superficie superior o dorsal de cualquier parte que protruye anteriormente desde el cuerpo, como el dorso del pie, de la mano o de la lengua.

**Palma:** Hace referencia a la parte plana de la mano, excluidos los cinco dedos. Es lo opuesto al dorso.

## TÉRMINOS DE MOVIMIENTO

**Flexión:** Significa doblar una parte o reducir su ángulo.

**Extensión:** Significa enderezar una parte o incrementar su ángulo entre sus partes. La única excepción que se efectúa, durante el plano sagital<sup>30</sup>, es durante el movimiento extensión-flexión del pulgar. (Ver figura 53).



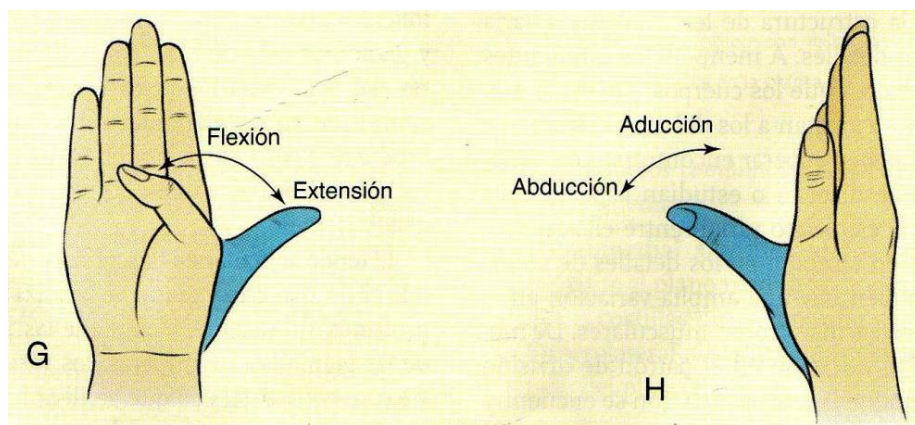
**Figura 53 En A) Flexión y extensión del antebrazo, B) Flexión y extensión de la mano, F) Flexión y extensión de los dedos**

Fuente: (Moore & Agur, 2008, págs. 5-7)

<sup>30</sup> El plano sagital también llamado plano anteroposterior pasa desde el frente hasta la espalda del cuerpo, creando un lado izquierdo y un lado derecho del cuerpo, divide el cuerpo en segmentos iguales con exactamente la mitad del cuerpo en cada lado del plano cardinal.

**Abducción:** “Significa alejarse del plano medio del cuerpo en el plano frontal” (Moore & Agur, 2008, págs. 5-7).

**Aducción:** “Significa aproximarse al plano medio del cuerpo en el plano frontal (coronal)” (Moore & Agur, 2008, págs. 5-7). Cuando se refiere a los dedos (de la mano y del pie), la abducción significa separarlos mientras que la aducción hace referencia a juntarlos. (Ver figura 54).

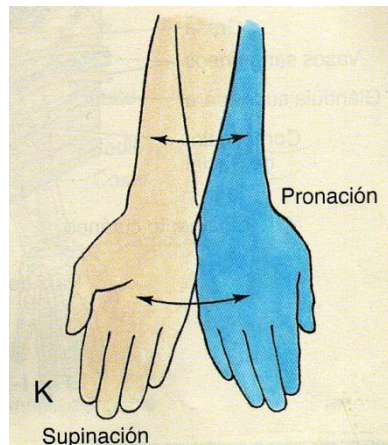


**Figura 54 En G) Flexión y extensión del dedo pulgar, H) Abducción y Aducción del dedo pulgar**

Fuente: (Moore & Agur, 2008, págs. 5-7)

**La pronación:** Es una “...rotación interna del antebrazo y la mano de manera que la palma quede orientada posteriormente” (Moore & Agur, 2008, págs. 5-7).

**La supinación:** Es una “...rotación externa del antebrazo y la mano, de tal forma que la palma quede orientada anteriormente” (Moore & Agur, 2008, págs. 5-7), como en la posición anatómica. (Ver figura 55).



**Figura 55 Pronación y supinación del antebrazo**

Fuente: (Moore & Agur, 2008, págs. 5-7)

## **EL PULGAR**

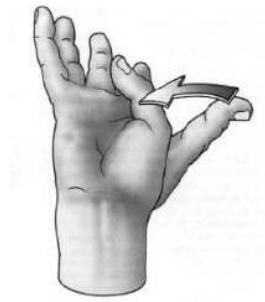
Esta parte de la mano sirve como mecanismo, principalmente, para la motricidad de pinzas entre sí y los demás dedos, desde la que se ejecuta, seguidamente, la fuerza general de la mano al juntarse con los restantes cuatro. De esta forma, el pulgar presenta algunas responsabilidades tales como el movimiento de oposición, mismo que consiste en juntar, de forma aislada o global, los cinco dedos; el movimiento de contra-oposición, que permite (para aliviar los dedos cuando estos están sujetando; y la flexibilidad funcional (Kapandji, 2006).

## **COMPONENTES DE MOVIMIENTO DEL PULGAR**

### **FLEXIÓN**

La flexión desplaza toda la columna del pulgar hacia dentro, denominado también abducción. Esta flexión involucra a todas las articulaciones del pulgar. (Ver figura 56).



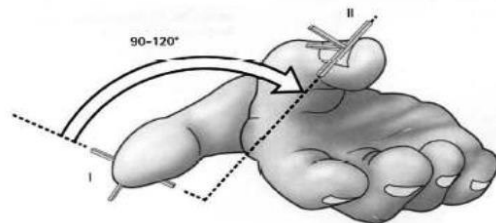


**Figura 56 Flexión del pulgar**

Fuente: (Kapandji, 2006, pág. 159)

## **PRONACIÓN**

Este movimiento determina un cambio "...de la última falange del pulgar que mira en direcciones diferentes, según su grado de rotación" (Kapandji, 2006, pág. 159). Básicamente, éste es un factor importante que compone esta parte de la mano, ya que permite a las yemas de los dedos contactarse con el pulgar. (Ver figura 57).



**Figura 57 Pronación del pulgar**

Fuente: (Kapandji, 2006, pág. 159)

## **2.3.2. MOVIMIENTOS DEL BRAZO**

### **MOVIMIENTO DEL CODO**

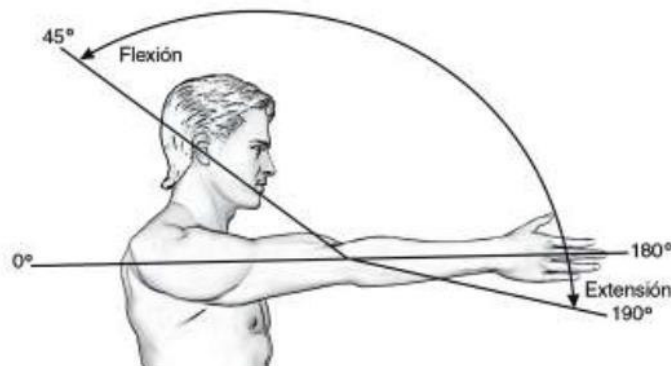
El codo es la articulación intermedia del miembro superior que permite la orientación en los tres planos del espacio con la ayuda del hombro y desplazando la mano del cuerpo.

## Movimiento del codo

Anatómicamente, el codo cuenta con una sola articulación; sin embargo, se definen otros movimientos que se logra con la ayuda del hombro.

### Flexión-Extensión

Flexión es el movimiento que permite acercar las caras anteriores del brazo y del antebrazo. El retorno del movimiento de flexión se denomina extensión. (Ver figura 58).

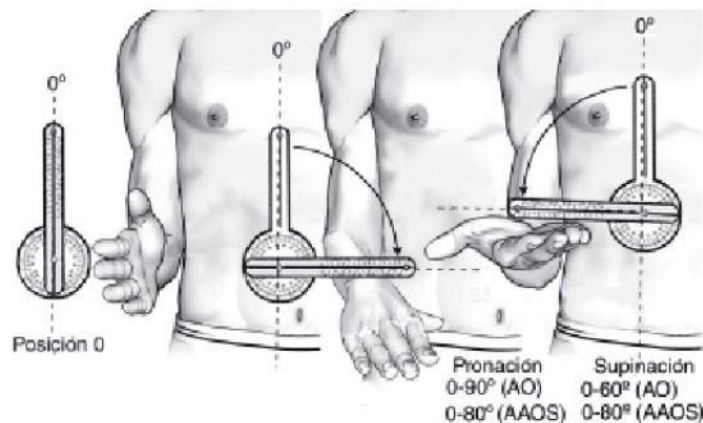


**Figura 58 Extensión y flexión del codo**

Fuente: (Taboadela, 2007, pág. 37)

### Rotación o Prono-Supinación

Éste tipo de rotación se efectúa exclusivamente en el área del codo, mismo que se realiza entre la membrana interósea y muñeca (Taboadela, 2007, pág. 73). (Ver figura 59).



**Figura 59 Movimientos del codo**

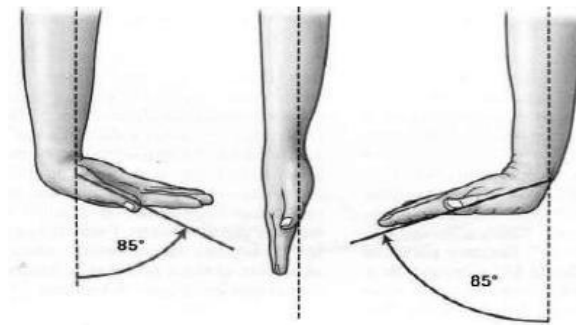
Fuente: (Taboadela, 2007, pág. 73)

## LA MUÑECA

La muñeca es responsable de los movimientos, así como la posición óptima de presión en las sujeciones. La muñeca posee dos grados de libertad que otorgan a la mano tanto el movimiento de abducción-aducción además del de flexión-extensión. Combinados ejecutan el movimiento de circunducción, siendo éste el tercer GDL (grado de libertad), mismo que recorre una superficie en forma de cono no regular y su vértice corresponde al centro de la muñeca.

## FLEXIÓN-ÉXTENSIÓN DE LA MUÑECA

La flexión se calcula en sentido anatómico, puesto que se relaciona la posición "...la muñeca alineada, cara dorsal de la mano en la prolongación de la cara posterior del antebrazo" (Kapandji, 2006, pág. 159). La flexión activa es de 85°, por lo que no alcanza un grado mayor a 90°. La extensión también es de 85° y no alcanza los 90°. (Ver figura 60).



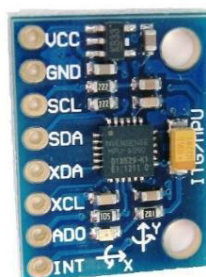
**Figura 60 Movimiento Flexión y extensión de la muñeca**

Fuente: (Kapandji, 2006, pág. 159)

## 2.4. ENTORNO DE DESARROLLO Y MATERIALES PARA IMPLEMENTAR EL GUANTE ELECTRÓNICO Y EL SISTEMA DE RECEPCIÓN

### 2.4.1. MPU-6050 (UNIDAD DE PROCESAMIENTO DE MOVIMIENTO)

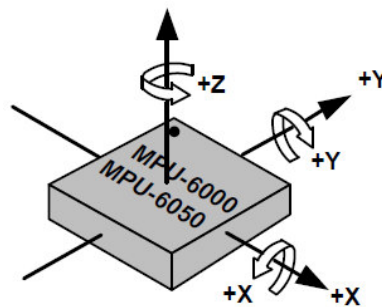
Estos dispositivos están dentro de los llamados sensores IMU (Inertial measurement unit), y han sido desarrollados por InvenSense. Estos miden o informan acerca de la velocidad, orientación y/o fuerzas gravitacionales de un aparato. Usualmente, este tipo de dispositivos consisten de dos o más partes. Citando por prioridad son el acelerómetro, giroscopio, magnetómetro y altímetro (Sparkfun, 2013). (Ver figura 61).



**Figura 61 MPU-6050**

Fuente: (Arduino, s.f.)

El MPU-6050 es un sensor IMU de seis ejes, lo cual significa que da seis valores en la salida, tres valores del acelerómetro y tres del giroscopio. El MPU-6050 es un sensor basado en tecnología MEMS (*Micro Electro Mechanical Systems*). Además, ambos, tanto el acelerómetro (ADXL 345) como el el giroscopio (ITG-3200), están embebidos dentro de este chip. (Ver figura 62 y 63).



**Figura 62 Orientación de los ejes de sensibilidad y polaridad de rotación**

Fuente:(InvenSense, MPU6000/MPU6050 9-axis Evaluation Board User guide, 2011, pág. 6)

## CARACTERÍSTICAS GENERALES

Este dispositivo (MPU-6050) cuenta con las siguientes características generales (Ver tabla 5):

- Salida digital de seis ejes (acelerómetro y giroscopio),
- 16 bits de salida para los dos sensores,
- Procesador DMP (*Digital Motion Processor*),
- Selección del rango de medición por el usuario de 2g a 16g,
- Plataforma Android, Linux, Windows,
- Dimensiones: 25.5 x 15.2 x 2.48mm,
- Fuente de alimentación: 2.37-3.46V (un adaptador interno permite hasta 5V),
- Comunicación I2C con todos los registros hasta 400kHz en modo rápido,

- Filtros digitales para el acelerómetro, giroscopio y sensor de temperatura, y
- Autodiagnóstico (InvenSense, MPU-6000 and MPU-6050 Product specification Revision 3.4, 2013, pág. 10).

### **CARACTERÍSTICAS DEL GIROSCOPIO DEL MPU-6050**

- Salida digital de rango de velocidad de  $\pm 250$ ,  $\pm 500$ ,  $\pm 1000$ ,  $\pm 2000$  °/s.
- Pin FSYNC para sincronización externa, se usa para la estabilización de imágenes en video y GPS.
- Bias mejorado y estabilidad en la sensibilidad de la temperatura que reduce la necesidad de la calibración por parte del usuario.
- Filtro digital paso bajo programable.
- Factor de escala predefinido por el fabricante.

### **CARACTERÍSTICAS DEL ACELERÓMETRO DEL MPU-6050**

- Salida digital de tres ejes con un rango de  $\pm 2g$ ,  $\pm 4g$ ,  $\pm 8g$ ,  $\pm 16g$ .
- Salida de 16 bits
- Funciona normalmente a  $500\mu A$
- A bajo voltaje funciona del siguiente modo:
  - 10  $\mu A$  a 1.25 Hz
  - 20  $\mu A$  a 5Hz
  - 110  $\mu A$  a 40Hz
- Interrupciones programables por el usuario
- Interrupciones de caída libre
- Interrupción a alta gravedad
- Interrupción de cero movimientos.

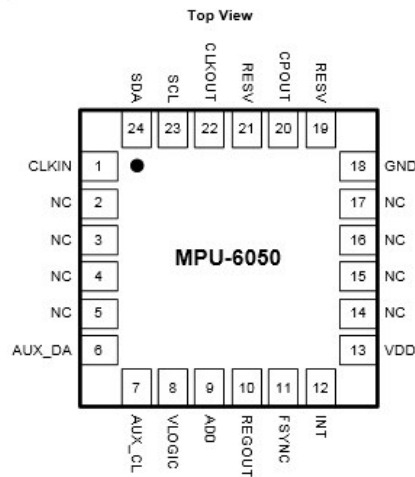
## DESIGNACIÓN DE PINES DEL MPU-6050

**Tabla 5**

### Designación de pines MPU-6050

PIN N°	NOMBRE	DESCRIPCIÓN
1	CLKIN	Referencia opcional externa entrada de reloj. (Conecte a tierra si no se utiliza)
6	AUX_DA	I2C datos, para la conexión de sensores externos
7	AUX_CL	I2C reloj, para la conexión de sensores externos
8	VLOGIC	Entrada o salida digital de suministro de voltaje
9	AD0/SDO	Dirección I2C, cuando el dispositivo trabaja como esclavo
10	REGOUT	Conexión al condensador regulador del filtro
11	FSYNC	Marco de entrada de sincronización digital. (Conecte a tierra si no se utiliza)
12	INT	Interrupción de salida digital (drenaje abierto)
13	VDD	Tensión de alimentación E/S digital
18	GND	Tierra
20	CPOUT	Conexión al capacitor que anula el alto voltaje generado por la bomba de carga.
22	CLKOUT	Sistema de salida de reloj
23	SCL	Señal de reloj I2C
24	SDA	Línea de datos I2C

Fuente: (InvenSense, MPU-6000 and MPU-6050 Product Specification, 2013, pág. 21)



**Figura 63 Pines del acelerómetro MPU-6050**

Fuente: (InvenSense, MPU-6000 and MPU-6050 Product Specification, 2013, pág. 21)

## ACELERÓMETRO ADXL345

Este acelerómetro presenta:

...una estructura micromaquinada suspendida sobre una oblea de silicio policristalino y capacitores diferenciales, de tecnología MEMS” (Devices, 2009, pág. 5). Un plato de cada capacitor es conectado a la estructura suspendida de forma independiente. Así la desviación de la estructura cambia la distancia entre los platos del capacitor, modificando la capacitancia. Un circuito en el interior del chip mide la diferencia entre estas capacitancias y el resultado es una amplitud proporcional de la aceleración. Esta amplitud es convertida en una señal digital compatible con el protocolo I2C (Devices, 2009, pág. 5).

## CARACTERÍSTICAS

El ADXL345 es un acelerómetro de 3 ejes (X, Y, Z), pues mide aceleración estática de gravedad en aplicaciones de detección de movimiento y aceleración dinámica para detectar movimientos o golpes con salida digital, con las siguientes características (ver tabla 6):



- Voltaje de alimentación de 2.0V a 3.6V
- Interfaces digitales SPI de 3 y 4 hilos, I2C.
- Temperatura de trabajo (-40°C a +85°C)
- Dimensiones de encapsulado 3mm x 5mm x 1mm
- Selección del ancho de banda por comandos seriales
- Pila FIFO de 32 registros
- Sensibilidad del sensor<sup>31</sup> de 4 mg/LSB, permite detectar cambios de inclinación de hasta 1°.
- Selección de la resolución de la salida desde 10bits – 16 bits
- Selección del rango de medición por el usuario 2g, 4g, 8g, 16g
- Consumo de energía 23µA a 2.5V

**Tabla 6****Designación de pines ADXL345**

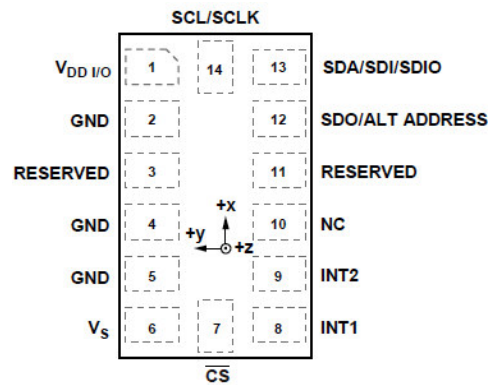
PIN Nº	NOMBRE	DESCRIPCIÓN
1	VDD I/O	Alimentación de voltaje
2	GND	Pin para conexión a tierra
3	RESERVED	Debe estar conectado a Vs o desconectado
4	GND	Pin para conexión a tierra
5	GND	Pin para conexión a tierra
6	Vs	Pin para alimentación de voltaje
7	CS	Entrada digital
8	INT1	Salida de interrupción 1
9	INT2	Salida de interrupción 2

Continúa  


<sup>31</sup> Mg/LSB (miligravedades/Least Significant Bit) quiere decir que mide una miligravedad por cada bit menos significativo de envío.

10	NC	Sin conexión interna
11	RESERVED	Debe estar conectado a $V_s$ o desconectado
12	SDO/ALT ADDRESS	Salida de datos serial (SPI 4 hilos)/dirección I2C
13	SDA/SDI/SDIO	Entrada de datos I2C, serial SPI 4 hilos
14	SCL/SCLK	Comunicación serial del reloj. SCL para I2C y SCLK para SPI

Fuente: (Devices, 2009, pág. 5)



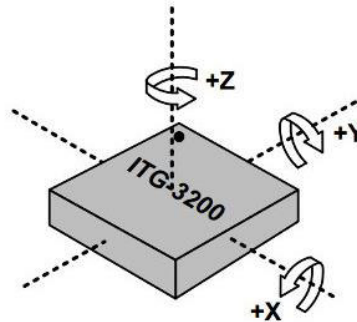
**Figura 64 Designación de pines ADXL345**

Fuente: (Devices, 2009, pág. 5)

## GIROSCOPIO ITG-3200

El giroscopio ITG-3200 está compuesto de 3 giroscopios independientes en cada eje que detectan la velocidad de rotación de coordenada (X, Y y Z) del sistema en un instante, fabricado por InvenSense con tecnología MEMS. (Ver figura 65). La señal que resulta luego de aplicar un movimiento al sensor es amplificada, demodulada y filtrada. Éste reproduce "...un voltaje proporcional

a la velocidad angular. Este voltaje es digitalizado utilizando un convertidor analógico digital de 16 bits” (InvenSense, ITG-3200 Product Specification Revision 1.4, 2010, pág. 5). Dicho convertidor se encarga de hacer un muestreo en cada eje (X, Y y Z). (Ver tabla 7).



**Figura 65 Orientación de los ejes de sensibilidad y polaridad de rotación**

Fuente: (InvenSense, ITG-3200 Product Specification Revision 1.4, 2010, pág. 13)

## CARACTERÍSTICAS

- Filtro programable digitalmente de paso bajo
- Amplio rango de suministro de voltaje de 2.1V a 3.6V
- Consumo de corriente de 6.5mA (mayor duración de la batería)
- Salida digital en (X, Y y Z) en un circuito integrado
- Sensibilidad de 14 LSBs por °/s
- Presenta un rango completo de medición de +- 2000 grados cada segundo
- Interface serial de modo rápido I2C
- Entradas opcionales de reloj externo para sincronizar con el reloj del sistema
- Sensor de temperatura con salida digital

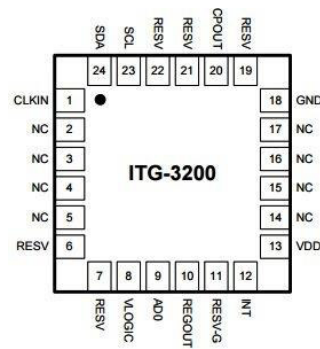
Tabla 7

## Designación de pines ITG-3200

PIN N <sup>o</sup>	NOMBRE	DESCRIPCIÓN
1	CLKIN	Referencia opcional externa entrada de reloj. (Conecte a tierra si no se utiliza)
8	VLOGIC	Alimentación digital IO tensión de VLOGIC debe ser $\leq$ VDD en todo momento
9	AD0	I2C LSB
10	REGOUT	Conexión del capacitor regulador del filtro
12	INT	Interrupción de salida digital
13	VDD	Entrada de voltaje
18	GND	Entrada de suministro de tierra
11	RESV-G	Conexión a tierra (reservada)
20	CPOUT	Conexión a condensador
23	SCL	Reloj I2C
24	SDA	Datos I2C

Fuente: (InvenSense, ITG-3200 Product Specification Revision 1.4, 2010, pág. 13)

En la figura 66 se muestra los pines del giroscopio ITG-3200:



**Figura 66 Pines del giroscopio ITG-3200**

Fuente: (InvenSense, ITG-3200 Product Specification Revision 1.4, 2010, pág. 13)

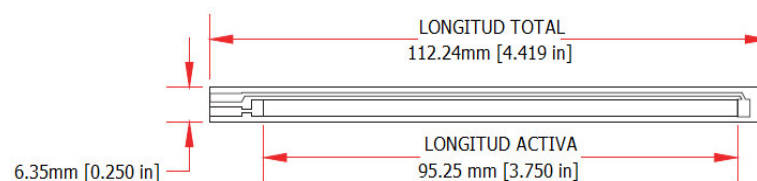
### 2.4.2. SENSORES DE FLEXIÓN

La tecnología de este tipo de sensores se basa en elementos resistivos de carbono que se presentan como una resistencia variable a la deformación, impresa sobre un sustrato delgado y flexible. Éste es unidireccional, pues trabajan en un sólo sentido y se debe tener cuidado con los conectores que se encuentran en la base del sensor, puesto que estos se pueden deteriorar o romper y se perdería la funcionalidad del dispositivo (Sparkfun, s.f.). (Ver figura 67 y 68).



**Figura 67 Sensor de flexión**

Fuente: (Sparkfun, s.f.)



**Figura 68 Medidas del sensor de flexión**

Fuente: (Symbol, 2014)

## CARACTERÍSTICAS GENERALES

- Miden el ángulo de desplazamiento, y
- Dispositivo de movimiento se doblan y flexionan físicamente.

## ESPECIFICACIONES ELÉCTRICAS

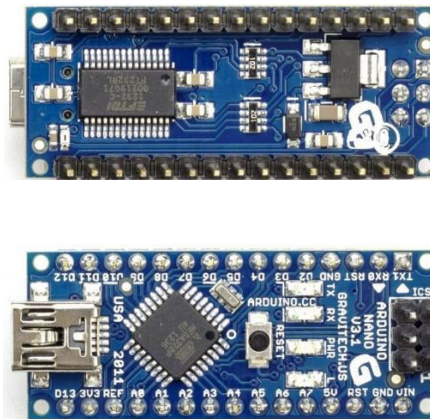
- Resistencia plana:  $10K\Omega \pm 30\%$  (tolerancia)
- Modelo Lineal
- Rango de resistencia flexionada: de  $60K\Omega$  a  $110k\Omega$
- Potencia nominal: 0.5 W continua; 1 W pico.

## ESPECIFICACIONES MECÁNICAS

- Ciclo de vida: > 1 millón,
- Altura:  $\leq 0.43\text{mm}$  (0.017 in), y
- Rango de Temperatura:  $-35^{\circ}\text{C}$  a  $+80^{\circ}\text{C}$  (Symbol, 2014).

### 2.4.3. ARDUINO NANO USB (V3.0)

Se le llama también placa Arduino Nano USB (V3) y está basado en el microcontrolador ATmega328P. (Ver figura 69).



**Figura 69 Arduino nano USB (v3.0)**

Fuente: (Arduino, s.f.)

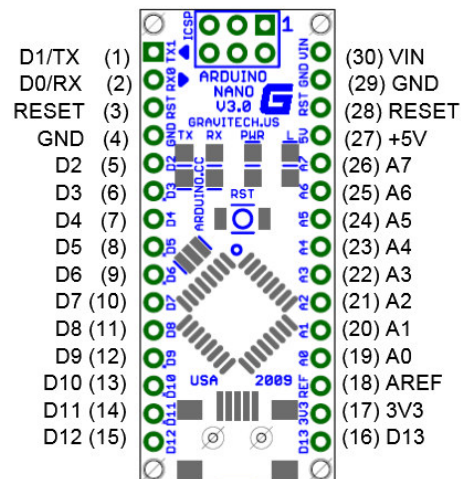
A continuación se muestra en la tabla 8 la designación de pines de Arduino Nano:

**Tabla 8**  
**Designación de pines Arduino nano USB (v3.0)**

PIN Nº	NOMBRE	TIPO	DESCRIPCIÓN
<b>1-2, 5-16</b>	D0-D13	I/O	Salida y entrada digital puerto 0-13
<b>3, 28</b>	RESET	Input	Reset (activado en bajo)
<b>4, 29</b>	GND	PWR(led indicador)	Suministro de tierra
<b>17</b>	3V3	Output	+3.3V salida (para FTDI)
<b>18</b>	AREF	Input	Referencia ADC
<b>19-26</b>	A0-A7	Input	Entrada analógica
<b>27</b>	+5V	Output or Input	+5V de salida del regulador interno  +5V de entrada desde una alimentación externa
<b>30</b>	VIN	PWR (led indicador)	Suministro de voltaje

Fuente: (Gravitech)

Pines en la figura 70:



**Figura 70 Pines del Arduino nano USB (v3.0)**

Fuente: (Gravitech)

## 2.3 DESCRIPCIÓN

- Microcontrolador Atmel “ATmega328P”
- Tensión de Operación (nivel lógico) 5V.
- Tensión de Entrada (recomendado) 7-12V.
- Tensión de Entrada (límites) 6-20V
- Pines E/S Digitales 14 (de los cuales 6 están provistos con salida PWM)
- Corriente máxima por cada PIN de E/S 40mA.
- Entradas Analógicas: 8
- Frecuencia de reloj: 16Mhz
- Memoria: 32KB de los cuales 2KB son usados por el bootloader.
- RAM: 2 KB - EEPROM : 1KB
- Conversor FDTI FT232RL.

### 2.4.4. MÓDULO XBEE S1

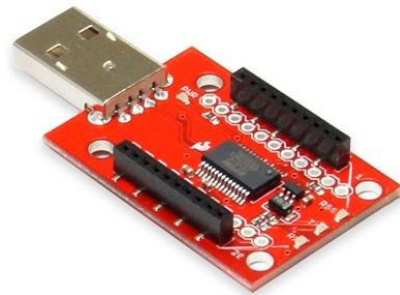
Este módulo usa el estándar IEEE 802.15.4 y se usa para enviar datos desde el transmisor del guante electrónico hacia el sistema receptor. (Ver figura 71 y 72).





**Figura 71 Módulo XBee S1 (izquierda)**

Fuente: (Alibaba global trade, 2015)



**Figura 72 Placa de conexión con USB (derecha)**

Fuente: (Myduino, 2016)

## CARACTERÍSTICAS

- Entrada: 2.8 – 3.4 V
- 9 entradas/salidas con entradas analógicas y digitales.
- Alcance de 100m con línea de vista.
- Interfaz serial
- Fáciles de integrar
- Consumo de corriente: Bajo consumo <50mA, cuando está en modo sleep <10μA, cuando transmite 45mA, cuando recibe 50mA
- 65000 direcciones para cada uno de los 16 canales disponibles. Se tienen muchos de estos dispositivos en una misma red.

## DESIGNACIÓN DE PINES

(Ver tabla 9).

**Tabla 9**

### Designación de pines Módulo XBee S1 (v3.0)

Pin #	Nombre	Dirección	Descripción
1	VCC	-	Terminal de alimentación
2	DOUT	Output	Terminal salida UART
3	DIN/CONFIG	Input	Terminal entrada UART
4	D08	Output	Salida digital 8
5	RESET	Input	Terminal de reset
6	PWM0/RSSI	Output	Salida PWM 0/RX muestra la intensidad de la señal
7	PWM1	Output	Salida 1 PWM
8	(RESERVADO)	-	No conectado
9	DTR/SLEEP_RQ/D18	Input	Control sleep o entrada digital 8
10	GND	-	Tierra
11	AD4/DI04	Either	Entrada analógica 4 o Digital I/O 4
12	CTS/DI07	Either	Digital I/O 7

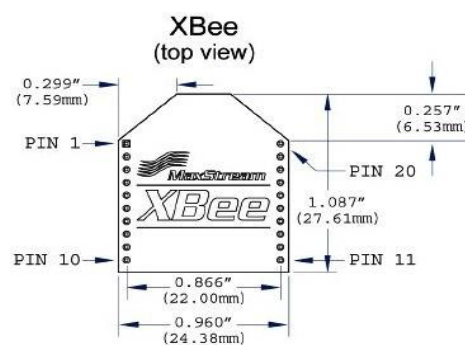
Continúa



13	ON/SLEEP	Output	Módulo indicador de status
14	VREF	Input	Voltaje de referencia para entradas A/D
15	Associate/AD5/DI05	Either	Entrada/salida digital 5
16	RTS/AD6/DI06	Either	Entrada analógica 5 o I/O Digital 5
17	AD3/DI03	Either	Entrada analógica 3 o I/O Digital 3
18	AD2/DI02	Either	Entrada analógica 2 o I/O Digital 2
19	AD1/DI01	Either	Entrada analógica 1 o I/O Digital 1
20	AD0/DIO0	Either	Entrada analógica 0 o I/O Digital 0

Fuente: (Digi International, 2009, pág. 7)

En la figura 73 se presentan las dimensiones y pines del módulo XBee S1:



**Figura 73 Dimensiones y pines del Módulo XBee S1**

Fuente: (Digi International, 2009, pág. 6)

### 2.4.5. POTENCIÓMETROS DIGITALES MCP 41100

Circuito integrado de 8 pines que se usa en reemplazo de los potenciómetros mecánicos. (Ver figura 74).



**Figura 74 Potenciómetro digital (8 pines)**

Fuente: (Microchip, 2004)

#### **CARACTERÍSTICAS**

- 256 pasos para cada acelerómetro
- Valor del potenciómetro para resistencia de 100k $\Omega$
- Rango de voltaje de operación 2.7V a 5.5V
- Resistencia baja de Wiper: 75 $\Omega$
- Configuración del potenciómetro con divisor de voltaje
- Utiliza la interfaz serial de comunicación SPI (10Mhz, modos 0,0 & 1,1)
- Multiplexación SDI/SDO
- Rango de temperatura de uso (-40 $^{\circ}$ C a +125  $^{\circ}$ C)

## DESIGNACIÓN DE PINES

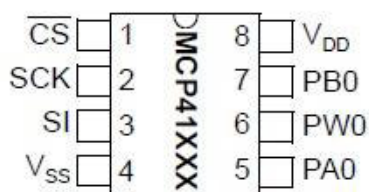
**Tabla 10**

### Designación de pines Potenciómetro Digital MCP 41100

PIN N°	NOMBRE	TIPO	DESCRIPCIÓN
1	CS	Input	(Chip Select) selección del dispositivo
2	SCK	Input	Señal de reloj externa enviada desde el dispositivo master
3	SDI/SDO	Input or Output	(MOSI, MISO) entrada y salida de datos desde el dispositivo master respectivamente
4	Vss	PWR	Tierra
5	P0A	Input	Terminal A del potenciómetro
6	P0W	Input	Terminal Wiper del potenciómetro
7	P0B	Output o Input	Terminal B del potenciómetro
8	Vdd	PWR	Suministro de voltaje 2.7V a 5.5V

Fuente: (Microchip Technology Inc, 2003, pág. 12)

En la figura 75 se muestra la designación de pines del MCP 41100:



**Figura 75 Designación de pines Potenciómetro Digital MCP 41100**

Fuente: (Microchip Technology Inc, 2003, pág. 1)

#### 2.4.6. ESTUCHE CARGADOR DE CELULAR

Es un cargador de batería externo para un teléfono móvil S3 mini GT-i8190. Cuenta con un botón de encendido/apagado de fácil operación. Adicional a esto, el estuche cargador funciona también como protector. Es muy práctico y cuenta con características de portabilidad y protección. Por esto, este dispositivo es elegido, en vista que ayuda a alimentar el circuito del guante electrónico. Brinda 5Voltios con una capacidad de 2000mAh, suficientes para energizar el guante. Además, sirve para albergar y a la vez proteger la placa electrónica del dispositivo maestro (Amazon, 2015). (Ver figura 76).



**Figura 76 Estuche cargador de celular**

Fuente: (Amazon, 2015)

## CARACTERÍSTICAS TÉCNICAS

- Entrada: 5V DC; 0.5 Ah
- Salida: 5V DC; 0.5Ah
- Capacidad: 2000mAh descarga continua
- Química: Polímero de Litio (Lithium-ion)
- Tamaño: 13.2 cm (largo), 6.5 cm (ancho), 1.8cm (alto)
- Peso neto: 70 g
- Recargable
- Cargador: Cable micro USB data (sync) con cualquier adaptador PC/MAC/AC.
- Temperatura de funcionamiento: -25°C a 60°C

### 2.4.7. BATERÍA LIPO – LITIO POLÍMERO

Una de las razones por las que se usa estas baterías es por su excelente recursividad de energía. Este tipo de baterías son muy livianas y proporcionan bastante poder. Normalmente cuestan más que las baterías normales y se debe tener mucho cuidado al usarlas. (Ver figura 77). (Amazon, 2015)



**Figura 77 Bateria LIPO**

Fuente: (Amazon, 2015)

## CARACTERÍSTICAS

- Salida: 7.4V
- Tipo de dos piezas en serie

- Capacidad: 1500mAh descarga continua
- Química: Polímero de Litio (Lithium-ion)
- Tamaño: 6.8 cm (largo), 3.7 cm (ancho), 1.9cm (alto)
- Peso neto: 92g
- Recargable: conector JST-BEC (12cm)
- Cargador: conector JST-XH (9cm)
- Temperatura de funcionamiento: -25°C a 60°C

### **CUIDADO DE LAS BATERÍAS LIPO (POLÍMERO DE LITIO)**

Para un excelente uso y aplicación de la batería a lo largo de su vida se recomienda no descargar cualquier tipo de unidad cuando se encuentre a menos de 3V. Asimismo, para elevar su calidad de uso, este tipo de batería debe llegar a un máximo de 60°C durante su uso, así como se recomienda utilizar un cargador específico para esta tecnología.

#### **2.4.8. CARGADOR BATERÍA TURNIGY**

Cargador de baterías LIPO con configuración 2S o 3S puede cargar alrededor de 800mAh. (Ver figura 78).



**Figura 78 Cargador de batería Turnigy**

Fuente: (Ebay, s.f.)



**CARACTERÍSTICAS**

- Entrada: 11-14 V
- Salida: 8.4 V – 12.6 V DC 800mAh
- Tamaño: 7.4 cm (largo), 5 cm (ancho), 2.5 cm (alto)
- Peso neto: 37g
- Indicador LED: rojo para poder, verde para carga
- Puede ser usado con batería de auto

## CAPÍTULO III

### DISEÑO Y CONSTRUCCIÓN DEL GUANTE ELECTRÓNICO

#### 3.1. DESCRIPCIÓN DEL PROYECTO

De acuerdo al esquema de un sistema de teleoperación<sup>32</sup>, se diseñan e implementan dos etapas en el presente proyecto: el dispositivo maestro y el dispositivo esclavo. Ambos mecanismos trabajan en conjunto para gobernar los movimientos de un Dron.

El dispositivo maestro es un guante electrónico sensorizado e inalámbrico. El cual extiende las capacidades y destrezas de la mano (diestra) humana para manipular un Dron. Este mando a distancia se implementa con dos sensores de flexión, uno en el dedo índice y otro en el dedo pulgar para efectivamente detectar la flexión de cada dedo. Adicional, se adhiere un sensor acelerómetro en el dorso del guante para detectar la variación de los movimientos al inclinar la mano en las diferentes direcciones posibles. En este sentido, se integra un estuche cargador de celular en el antebrazo del guante para alimentar el circuito, mismo que alberga un módulo de Arduino Nano y un módulo XBee S1. El Arduino Nano permite leer e interpretar los datos recibidos por los sensores de flexión y el acelerómetro, para posteriormente enviarlos al módulo XBee S1. Por su parte, el módulo XBee S1 se encarga de transmitir inalámbricamente los datos al dispositivo esclavo.

El dispositivo esclavo consta de un sistema de recepción y del control propio que viene con el Dron. Es decir, el sistema receptor está conectado directamente a la placa electrónica del control remoto del Dron, formando un solo dispositivo esclavo entre el sistema receptor y el control remoto propio del Dron. El sistema receptor está conformado por un segundo módulo XBee S1. El segundo módulo está constituido por un Arduino Nano, una placa de potenciómetros digitales y una batería recargable de alimentación con su respectivo cargador. El sistema receptor, como su nombre lo dice, se encarga

---

<sup>32</sup> Capítulo II, páginas 11,12

de realizar la etapa de recepción de datos que han sido enviados desde el guante electrónico. Una vez que el dispositivo maestro envía información hacia el dispositivo esclavo, el módulo XBee S1 es el encargado de recibir los datos, los mismos que son enviados al módulo Arduino Nano. Luego de ello, el Arduino lee y procesa la información, para seguido interpretar los datos. Así, el Arduino se comunica con la placa de potenciómetros digitales. Seguido, esta placa envía las órdenes recibidas del Arduino al control remoto propio del Dron.

El control remoto se comunica mediante una interfaz directa con el Dron, puesto que este control remoto envía directamente las órdenes del dispositivo esclavo al cuadricóptero. De esta manera se tiene un control de los movimientos de un Dron, desde un guante electrónico en tiempo real. (Ver figura 79).



**Figura 79 Descripción del proyecto**

### 3.2. ARQUITECTURA DEL SISTEMA

Según las arquitecturas de teleoperación, la arquitectura del sistema está basado en un esquema de fuerza-posición. La posición del esclavo se determina por el seguimiento del dispositivo maestro. Las fuerzas que aparecen sobre el esclavo se miden y se generan en el maestro mediante sus sensores. En la figura 80 se puede observar la arquitectura del sistema. (Padilla, 2014, pág. 15).



**Figura 80 Arquitectura del sistema**

De acuerdo a la arquitectura planteada se establecen dos etapas definidas:

- La Etapa 1 consiste en la fase de detección, procesamiento y transmisión de datos a la Etapa 2, mediante un guante electrónico.
- La Etapa 2 corresponde a la recepción de datos, procesamiento y envío de datos al control remoto del Dron, mediante un sistema receptor.

La Etapa 1 y la Etapa 2 se comunican inalámbricamente.

### **3.3. ETAPA 1 – DISPOSITIVO MAESTRO**

#### **3.3.1. GUANTE ELECTRÓNICO**

Se denomina dispositivo maestro al guante, ya que es el que da las órdenes en el sistema planteado. En el campo de la tecnología, los guantes electrónicos se clasifican dentro de las interfaces hápticas<sup>33</sup>. De acuerdo a esta clasificación, el guante electrónico de este proyecto está catalogado como una interfaz háptica con retroalimentación de fuerza<sup>34</sup>. Esto quiere decir que es un dispositivo que le permite al operador gobernar los movimientos del Dron según la fuerza de inclinación que aplique el operador con la mano, así como también al flexionar sus dedos.

#### **CARACTERÍSTICAS DEL GUANTE ELECTRÓNICO**

El diseño del guante electrónico conforma la parte fundamental del proyecto. En este sentido, el objeto principal de la investigación es la elaboración de un prototipo de bajo coste y con un alto grado de innovación respecto a la forma de controlar los movimientos de un Dron. Por tal razón, luego de un análisis se eligen los siguientes elementos electrónicos:

- 2 sensores de flexión de 100k $\Omega$
- 1 acelerómetro MPU-6050 (ADXL345/ITG3200), 6 grados de libertad
- 1 módulo XBee S1
- 1 módulo Arduino Nano

#### **FASE DE DETECCIÓN**

Para realizar la fase de detección se requieren elementos que sean de fácil uso y adaptación al guante, puesto que es importante que sean capaces de percibir y representar los movimientos de la mano, especialmente el de la

---

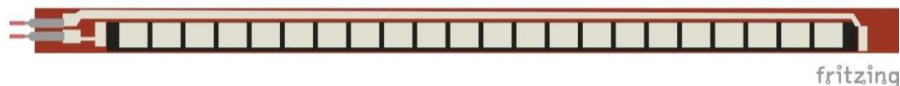
<sup>33</sup> Capítulo II, páginas 17, 18

<sup>34</sup> Capítulo II, página 18

inclinación o flexión de los dedos. Para esto se hace uso del guante electrónico, el mismo que está dotado con dos sensores de flexión y un acelerómetro.

## SENSORES DE FLEXIÓN

El sensor flexo-resistivo es, básicamente, una resistencia variable que reacciona al flexionamiento. Se ha decidido usar este elemento porque tiene las siguientes medidas: 95.25 mm de largo y 6.35mm de ancho, que para esta ocasión se adapta perfectamente a los dedos de la mano promedio, lo cual los hace muy útiles para este prototipo. El flexionamiento del dispositivo es detectado solamente en una dirección, acoplándose precisamente a la flexión unidireccional de los dedos de la mano. (Ver figura 81).



**Figura 81 Sensor de flexión**

Fuente: (Postdam, Fritzing, 2014)

Estos dispositivos funcionan de manera que, al ser flexionados, su resistencia aumenta. Es decir, el sensor produce una resistencia de salida correlacionada al radio de curvatura, mientras menor es el radio, mayor es el valor de la resistencia. Estos valores de resistencia varían desde 10K $\Omega$  en estado plano o nominal hasta 110k $\Omega$  cuando se flexiona el sensor 90°. Su durabilidad es de más de 1 millón de flexiones, de esta manera se garantiza el uso del guante por un tiempo razonable.

## RESOLUCIÓN

De acuerdo a la presente investigación, al momento que el Arduino Nano recibe e interpreta los valores de los sensores de flexión se considera el parámetro de resolución, el mismo que se define según el valor del convertidor análogo digital (ADC) que es el que determina dichos valores (Gómez, 2013). Para calcular el número de valores discretos que el ADC es capaz de convertir se usa la siguiente fórmula:

$$\text{valores discretos} = 2^{\text{resoluciónbits}} \rightarrow$$

Donde, la resolución de bits para Arduino Nano es de 10 bits:

$$\text{valores discretos} = 2^{10} \rightarrow$$

$$\text{valores discretos} = \mathbf{1024}$$

Esto significa que se mapearan los voltajes de entrada entre 0 y 5 voltios a valores enteros desde 0 a 1023.

## PRESICIÓN DE LA MEDICIÓN

Lo que indica este parámetro es que cuanto mayor sea el número de bits, mayor será el número de intervalos (cuantificación), menor será el ancho del intervalo y por lo tanto mejor la precisión de la medición<sup>35</sup> (Gómez, 2013). Se tiene una precisión en la medición de los sensores de flexión de:

$$\frac{5 \text{ voltios}}{1024 \text{ unidades}} = 0.0049 \text{ voltios por unidad} \equiv \mathbf{5 \text{ mV por unidad}}$$

---

<sup>35</sup> El número de bits necesario depende de la aplicación que se desee realizar.

En este sentido, se puede relacionar el valor que retorna el ADC con respecto al voltaje. Para el guante electrónico se trabaja con Arduino Nano. La técnica que aplica Arduino para el ADC se basa en usar el voltaje análogo para cargar un capacitor interno y luego, mediante una resistencia, interna medir el tiempo que le toma al capacitor descargarse. El microcontrolador de Arduino Nano monitorea el número de ciclos de reloj que pasan antes de que el capacitor se descargue. Este número de ciclos es el número que retorna el ADC una vez que se complete la conversión<sup>36</sup>. En este caso el convertidor asume que 5 voltios son 1023 unidades y cualquier valor por debajo de 5 voltios será un valor entre 0 y 1023.

De acuerdo a Seidle, mediante la siguiente fórmula se puede simplificar la Lectura del ADC:

$$\frac{\text{Resolución del ADC}}{\text{Sistema de Voltaje}} = \frac{\text{Lectura del ADC}}{\text{Voltaje análogo medido}} \rightarrow$$

Donde se tiene que la resolución del ADC es 10 bits, el sistema de voltaje de 0 a 5 voltios. Para este ejemplo, se usará un voltaje análogo medido de 2.5 voltios:

$$\frac{1024 \text{ unidades}}{5 \text{ V}} = \frac{\text{Lectura del ADC}}{2.5\text{V}} \rightarrow$$

Simplificando:

$$\text{Lectura del ADC} = \frac{1024 \text{ unidades}}{5 \text{ V}} * 2.5\text{V} \rightarrow$$

$$\text{Lectura del ADC} = \mathbf{512 \text{ unidades}}$$

---

<sup>36</sup> Proceso interno de Arduino



Entonces, cuando el sensor de flexión se encuentra en un valor medio de flexión entre 0 y 5 voltios (2.5V), el ADC reportará una medida de 512 unidades. En la siguiente tabla se hace una referencia a todos los demás valores posibles. (Ver tabla 11).

**Tabla 11**

**Tabulación de datos que retorna el ADC de los sensores de flexión.<sup>37</sup>**

Entrada análoga (V)*	Salida digital
0.000 – 0.0049	0
1.245 – 1.250	256
2.495 – 2.500	512
3.745 – 3.750	768
4.995 – 5.000	1023

\*valores digitales cada 0.0049 V (4.9 mV)

## TIEMPO DE CONVERSIÓN Y FRECUENCIA DE MUESTREO

De acuerdo Gómez (2013), la frecuencia de muestreo máxima del ADC se calcula utilizando el inverso del tiempo de conversión:

$$F_{\text{muestreo máx}} = \frac{1}{t_{\text{conversión}}} \rightarrow$$

Para la función *analogRead(sensorPin)* de Arduino Nano se tiene que la frecuencia de muestreo es 9600 Hz, simplificando:

<sup>37</sup> Se tomaron solamente cinco valores de referencia

$$t_{conversión} = \frac{1}{9600Hz} \rightarrow$$

$$t_{conversión} \cong 100\mu s$$

Así, se presenta un tiempo de conversión muy pequeño, suficiente para la aplicación del guante electrónico, debido que se usa la velocidad de reacción de los movimientos de los dedos para esa medición.

### DIVISOR DE VOLTAJE

Según la ley de Ohm, 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 flexo resistivo actúa como resistencia variable del circuito. En este caso, por el pin analógico del Arduino, deben entrar valores con tensión entre 0V y 5V. Estos valores serán traducidos por el ADC integrado de Arduino Nano (10 bits). Para determinar los valores de voltaje que se producen en los sensores de flexión se usa un divisor de voltaje que está formado por una resistencia de 22K $\Omega$  y la lectura del sensor de flexión, a continuación se muestra los valores máximos y mínimos que se pueden obtener con la siguiente relación:

$$V_{out} = V_{in} * \frac{\text{Lectura del Sensor de flexión}}{22k\Omega + \text{Lectura del Sensor de flexión}} \rightarrow$$

Entonces con un valor de resistencia en ambos dedos de 30k $\Omega$ , que se obtiene en el sensor de flexión cuando se encuentra estirado, el valor de voltaje es:

$$V_{out} = 5V * \frac{30K\Omega}{22k\Omega + 30k\Omega} \rightarrow$$

Simplificando:

$$V_{\text{sensor estirado}} = 2.88V$$

Así, el valor de voltaje que se tiene en el sensor de flexión del dedo índice cuando está flexionado es:

$$V_{\text{out}} = 5V * \frac{80K\Omega}{22k\Omega + 80k\Omega} \rightarrow$$

$$V_{\text{sensor flexionado dedo índice}} = 3.9V \approx 4V$$

De igual manera, el valor de voltaje que se tiene en el sensor de flexión del dedo pulgar cuando está flexionado es:

$$V_{\text{out}} = 5V * \frac{50K\Omega}{22k\Omega + 50k\Omega} \rightarrow$$

$$V_{\text{sensor flexionado dedo pulgar}} = 3.47V$$

### DEDO ÍNDICE: ALTURA DE VUELO

El sensor ubicado en el dedo índice del guante es el que controla el ascenso/descenso del Dron. Este mando es el encargado de hacer despegar al cuadricóptero. Asimismo, cumple la función de acelerador, por lo que regula la altura de vuelo según su flexión. A más flexión se genera más altura. Por el contrario, a menos flexión se evidencia menos altura. Cuando el dedo índice está completamente estirado, el dron se encuentra detenido y, viceversa, cuando el dedo está totalmente flexionado el dron aplica su máxima velocidad y altura. (Ver figura 82). (Ver tabla 12).

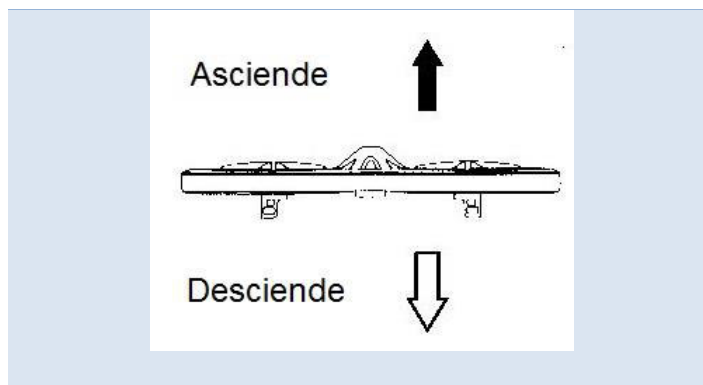


Figura 82 Ascenso/Descenso del Dron

Tabla 12

Mando del dedo índice para ascender/descender el Dron\*

SENSOR DE FLEXIÓN PARA ASCENDER/DESCENDER EL DRON				
Dedo	Movimiento	Altura del dron (m)	Voltaje (V)	Resistencia (Ohmios)
Índice	Extendido	0	2.88	Mínimo 30K
	Flexionado totalmente	Máxima 100	4	Máximo 80K

\*precisión de la medición cada 0.0049 V (4.9 mV por unidad)

### DEDO PULGAR: VIDEO ON/OFF

El sensor ubicado en el dedo pulgar del guante es el que activa o desactiva el botón para hacer videos del control del Dron. (Ver figura 83). Cada vez que se flexiona el dedo pulgar se activa o desactiva el video. Existe un sonido que emite el control del dron cada vez que se flexiona el dedo. Como resultado,

aparece un ícono en la pantalla del control del dron, indicando que se está grabando o no, con estos indicadores se tiene una perfecta noción del momento de grabado. Así, se puede iniciar y detener un video en cualquier momento del vuelo del cuadricóptero. (Ver tabla 13).



**Figura 83 ON/OFF de video, pantalla de control del Dron**

**Tabla 13**

**Mando del dedo pulgar para hacer videos con el Dron\***

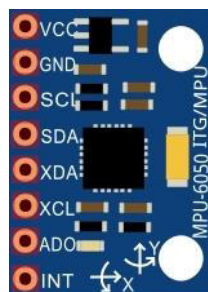
Flexión/Extensión del dedo	Botón del control del Dron	Resistencia (Ohmios)	Voltaje (V)
Pulgar	ON/OFF de Video	Mínimo 30K	2.88
		Máximo 50 K	3.47

\*precisión de la medición cada 0.0049 V (4.9 mV por unidad)

**UNIDAD DE PROCESAMIENTO DE MOVIMIENTO “MPU-6050”**

Se hace un análisis de acuerdo a las características necesarias para la presente aplicación, considerando los rangos de funcionamiento, sensibilidad y tamaño. Se selecciona la unidad MPU-6050 porque presenta una

sobresaliente linealidad, la misma que reduce significativamente el error de desalineación que se produce en los sensores IMU. Por sus dimensiones cabe sin inconvenientes en el dorso de la mano. Este dispositivo es una unidad de 6 grados de libertad o 6 DOF (*Degrees of freedom*). Esto significa que lleva un acelerómetro y un giroscopio, ambos de 3 ejes (3+3=6 DOF). Este sensor opera a 3.3 voltios, aunque tiene un regulador que permite conectarla a 5V. Asimismo, el MPU-6050 utiliza el protocolo de comunicación I2C (*Inter-Integrated Circuit*) denominado como bus de comunicaciones en serie dedicado y cuenta con tecnología de procesamiento de movimiento DMP (*Digital Motion Processing*). (Ver figura 84).



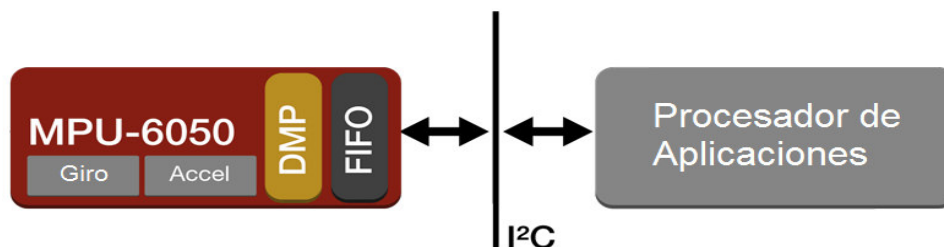
**Figura 84 Encapsulado MPU-6050**

Fuente: (Torres, 2014)

El *Digital Motion Processing* es una tecnología de procesamiento de movimiento que permite recolectar los datos del sensor con una salida de 16 bits, tanto para los 3 ejes del acelerómetro y los 3 ejes del giroscopio, sincronizándolos a una velocidad que se ha determinado en 9600 Baudios (velocidad estándar escogida). Además, cuenta con un *buffer* FIFO (First Input First Output)<sup>38</sup> de 1024 bytes, que guarda el conjunto completo de datos colectados. Esto ayuda a reducir los requisitos de sincronización en el procesador del sistema al momento de leer la ráfaga de datos recibidos. Una vez que se pueden leer los datos, el procesador del sistema puede ahorrar

<sup>38</sup> Buffer FIFO (First Input First Output) se refiere a que el MPU-6050 lo usa para enviar el primer dato que recibió en su entrada. Es decir, el primer dato que recibió es el primer dato que envía, mientras guarda los siguientes datos entrantes.

energía mediante la introducción de un modo de espera de bajo consumo (*low-power*), mientras que el MPU recoge más datos. Por su parte, la comunicación I2C funciona a 400kHz en modo rápido (*Fast-mode*) para comunicarse con todos los registros. (Ver figura 85).



**Figura 85 Sensor Inercial MPU-6050**

Fuente: (InvenSense, 2014)

Sin embargo, por razones de utilidad para esta aplicación en particular, se reciben naturalmente en el procesador del Arduino los datos de los 3 ejes del acelerómetro y de los 3 ejes del giroscopio, pero solamente se usan los datos que envía el 'acelerómetro' (porque la aceleración puede tener lugar sin observarse cambios en la rotación). Asimismo, se trabaja únicamente con los valores recibidos en los ejes 'X' e 'Y' del acelerómetro, porque los movimientos de la mano para esta aplicación solo exigen los valores que se obtienen en el plano 'XY'. Entonces, de ahora en adelante solo se hará referencia a los parámetros concretamente del acelerómetro del sensor inercial MPU-6050.

### **LSB (LEAST SIGNIFICANT BIT)**

De acuerdo a Pillajo Obando & Robayo Cajamarca (2012, pág. 68), para una mejor comprensión de los siguientes parámetros, se especifica el concepto de LSB. Con lo que concierne a los ADC, el LSB es un indicador que permite establecer el cambio de voltaje toda vez existe una variación gradual del ADC, además de determinar el voltaje designado a cada bit. Su valor depende del rango máximo de voltaje y del número de bits del ADC.

## SENSIBILIDAD DEL ACELERÓMETRO

Este factor es importante al momento de experimentar los movimientos de la mano, porque es el que indica la cantidad de medida que percibe el sensor en función de la magnitud física que se aplica (inclinación). La unidad MPU-6050 tiene una sensibilidad de 16384 LSB/g, en donde el LSB representa el bit menos significativo y se usa con respecto a la magnitud física aceleración “ $g$ ” ( $1g = 9.81 \text{ m/s}^2$ ).

## RANGO DE MEDIDA

El MPU-6050 tiene un acelerómetro de rango de “bajo- $g$ ” (de  $\pm 2g$  a  $\pm 16g$ ). Se considera que el rango de gravedad ( $1g = 9.81 \text{ m/s}^2$ ) que puede experimentar un cuerpo humano sin sufrir daños oscila entre los  $\pm 5g$ . En este caso, la sensibilidad del sensor MPU-6050 se ajusta a  $\pm 2g$ , porque se estima que es suficiente para esta aplicación y se lo realiza mediante programación, escribiendo en los registros del controlador. Para esta configuración, el acelerómetro envía valores a Arduino en un rango de números enteros entre  $[-16384, +16384]$  (Pillajo Obando & Robayo Cajamarca, 2012, pág. 68).

## PRESICIÓN

Con los parámetros mencionados del rango de medida en el inciso anterior, cada variación de una unidad digital será equivalente a una variación de  $61 \mu\text{g}/\text{LSB}$ , en donde ( $1g = 9.81 \text{ m/s}^2$ ) y el LSB representa el bit menos significativo.

## INCLINACIÓN DEL ACELERÓMETRO

Los acelerómetros miden el efecto de la fuerza de gravedad sobre ellos mismos. El aumento de percepción de gravedad en uno de los ejes indica que



éste se está poniendo en vertical, incidiendo toda la fuerza de gravedad sobre él. Asumiendo que el acelerómetro está alineado perfectamente con el suelo el eje 'Z' marcará  $9.81 \text{ m/s}^2$  y los otros dos ejes marcarán 0 (cero). En cambio, si se gira el MPU-6050 a  $90^\circ$  ahora es el eje 'X' el que está perpendicular al suelo. Sólo entonces, este eje marcará  $9.81 \text{ m/s}^2$  (Robologs, 2014). (Ver figura 86).

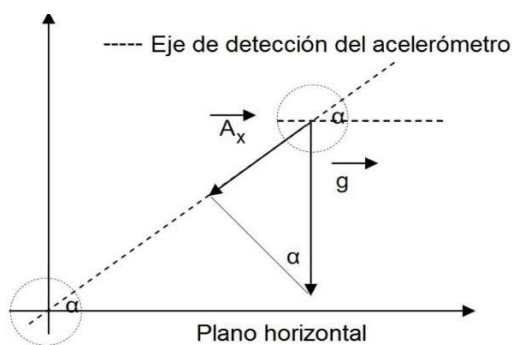


**Figura 86 Gravedad en los ejes del acelerómetro según la inclinación**

Fuente: (Robologs, 2014)

El ángulo de giro de un eje, vendrá definido por una combinación del peso percibido en cada uno de los 3 ejes.

## DETECCIÓN DE LA INCLINACIÓN PARA UN EJE



**Figura 87 Medición del ángulo de inclinación en un eje del acelerómetro**

Fuente: (STMicroelectronics, 2010, pág. 10)

En la figura 87 se muestra un eje de detección del acelerómetro para la medición de la inclinación. El acelerómetro mide la proyección del vector de gravedad sobre el eje de detección. La amplitud de la aceleración detectada cambia de acuerdo al sentido del ángulo  $\alpha$  entre el eje sensitivo y el plano horizontal (STMicroelectronics, 2010, pág. 10).

$$A_x = g \times \sin(\alpha)$$

Usando la ecuación antes mostrada, es posible estimar el ángulo de inclinación ( $\alpha$ ), despejando:

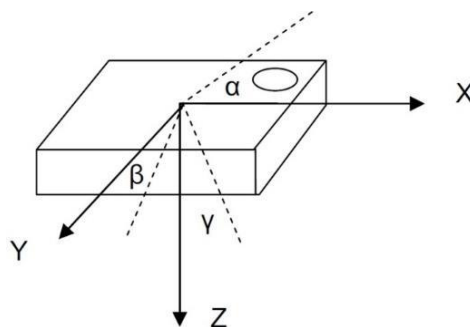
$$\alpha = \arcsin\left(\frac{A_x}{g}\right)$$

Dónde:

- $A_x$  = aceleración medida
- $g$  = Vector de la gravedad de la tierra

### DETECCIÓN DE LA INCLINACIÓN PARA 3 EJES

Con un acelerómetro de tres ejes, el usuario puede usar el eje Z para combinarlo con los ejes X e Y para detectar la inclinación con la finalidad de mejorar la sensibilidad de inclinación y precisión. (Ver figura 88).



**Figura 88 Medición del ángulo de inclinación para tres ejes del acelerómetro**

Fuente: (STMicroelectronics, 2010, pág. 13)

Las ecuaciones que permiten hallar los ángulos de inclinación *roll* y *pitch* con respecto a cada uno de los planos:

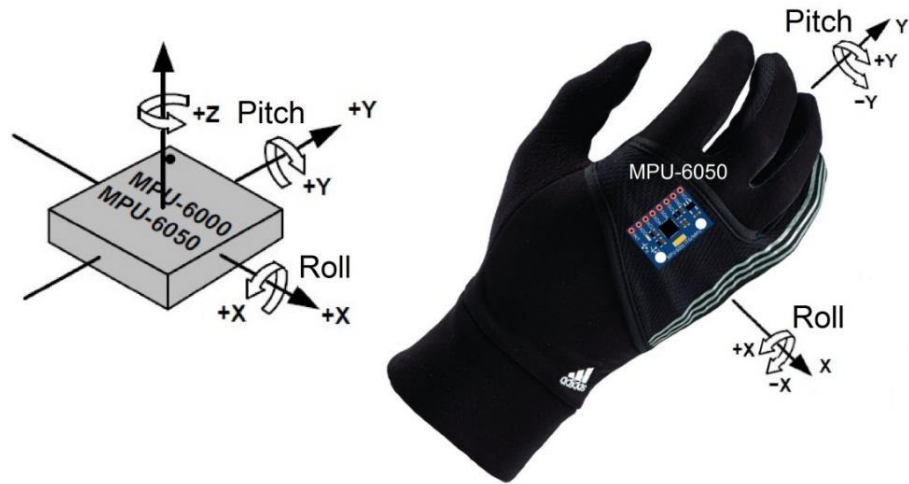
$$Pitch = \alpha = \arctan\left(\frac{A_{x1}}{\sqrt{(A_{y1})^2 + (A_{z1})^2}}\right)$$

$$Roll = \beta = \arctan\left(\frac{A_{y1}}{\sqrt{(A_{x1})^2 + (A_{z1})^2}}\right)$$

Una vez definidos estos ángulos se procede a realizar la configuración de los valores de *pitch* y *roll* con respecto al guante electrónico, según los movimientos posibles de la mano.

### **CONFIGURACIÓN DEL ACELERÓMETRO DE ACUERDO A LOS MOVIMIENTOS DE LA MANO DIESTRA**

Para la configuración se consideran parámetros como los niveles de aceleración que se van a aplicar, el rango de frecuencia y la resolución del acelerómetro. El acelerómetro puede ser utilizado para medir tanto: aceleraciones dinámicas (vibración) como estáticas (gravedad), según la aplicación requerida. El tipo de fuerzas que se producen por movimientos de la mano son denominadas como aceleraciones estáticas. Es así que para el caso del guante electrónico, el acelerómetro realiza una medición de sus fuerzas estáticas. (Ver figura 89).



**Figura 89 Movimientos en *roll* y *pitch* respecto al guante electrónico**

En el capítulo II se analizó los movimientos de la mano. Según este análisis se obtienen los valores de los ángulos que se puede girar la mano naturalmente. De acuerdo a estos ángulos, que son tomados respecto al eje transversal<sup>39</sup> del cuerpo humano, se configura el acelerómetro con la finalidad de obtener una lectura lo más precisa posible según los diferentes desplazamientos de la mano.

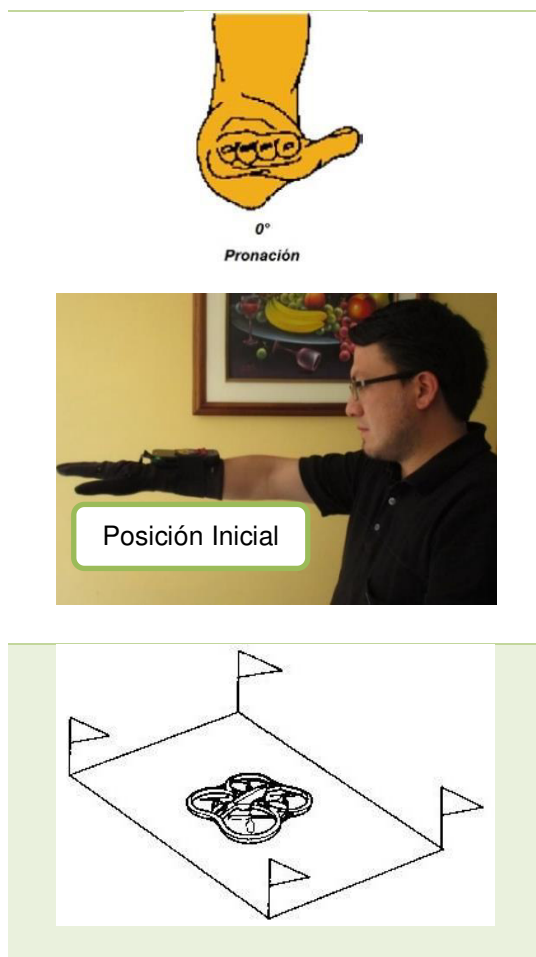
Para emular el funcionamiento del control del dron de acuerdo a la inclinación del acelerómetro mediante el guante electrónico se hace una analogía simple:

### **POSICIÓN INICIAL**

La mano tiene la posición llamada pronación con un ángulo de  $0^\circ$  respecto al eje transversal de la mano, tal como se ejemplifica en la figura 90. Esta condición se aprovecha en este proyecto, de tal manera que, para cuando el guante electrónico está en pronación el Dron, se mantiene paralelo al suelo u

<sup>39</sup> Es un eje paralelo al suelo, que cruza al cuerpo de derecha a izquierda.

horizontal debido a la posición del acelerómetro. Esta postura se la puede denominar posición neutra o home del Dron.

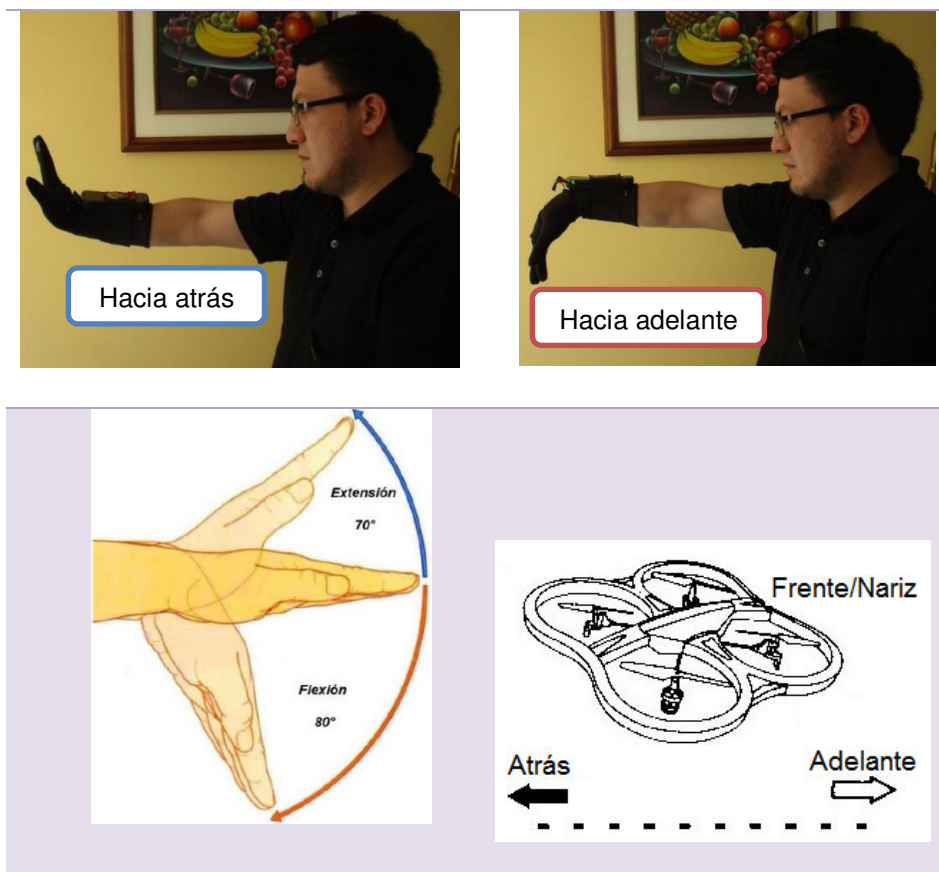


**Figura 90** Movimiento de pronación de la mano

### **MANDO HACIA ADELANTE Y ATRÁS**

La mano tiene también los movimientos de flexión y extensión. Estas posturas tienen un ángulo de  $80^\circ$  para la posición llamada flexión y un ángulo de  $70^\circ$  para la posición llamada extensión. Las mismas tomadas respecto al eje transversal de la mano como se muestra en la figura 91. Estos movimientos se aprovechan de tal manera que, para cuando se flexiona la mano, el Dron se mueve hacia adelante. Cuando se extiende la mano, el Dron se mueve hacia atrás. Esta variación de ángulos de la mano se toma de referencia para

el ángulo determinado como *pitch* del acelerómetro, según la configuración planteada para este proyecto. (Ver figura 91).



**Figura 91 Movimientos y ángulos de flexión-extensión de la mano**

### **MANDO HACIA LA DERECHA**

La mano cuenta con otras posturas como la posición llamada supinación o pronación media con un ángulo de  $90^\circ$  respecto al eje transversal de la mano. En el proyecto, esta condición se usa para gobernar el desplazamiento del Dron a la derecha, tomando esta variación del ángulo de la mano de referencia para el ángulo determinado como roll del acelerómetro según la configuración planteada. (Ver figura 92).

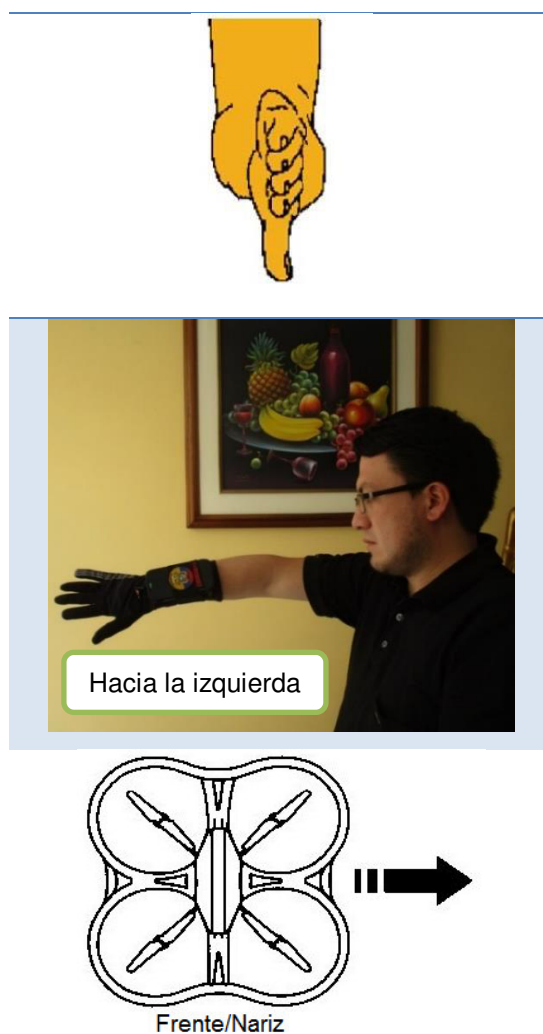


**Figura 92 Movimiento de supinación-pronación media de la mano**

### **MANDO HACIA LA IZQUIERDA**

Ahora, para cuando se desea desplazar el Dron hacia la izquierda se requieren dos movimientos combinados: uno de la mano y otro del codo. Se parte desde un movimiento de pronación de la mano y luego se realiza una rotación interna del codo con un ángulo de  $-90^\circ$ , con respecto al eje transversal de la mano, dando como resultado la posición que está en la figura 93 con el pulgar apuntando hacia el suelo. Asimismo, en esta postura se toma de

referencia el cambio en el ángulo de la mano para el ángulo determinado como *roll* del acelerómetro, según la configuración planteada para este proyecto. (Ver figura 93).



**Figura 93 Movimiento de pronación de la mano**

Una vez expuesta la manera como se gobierna el Dron mediante las posturas de la mano con el guante electrónico, se tiene un resumen de los movimientos de la mano y la variación de los ángulos del guante electrónico respectivamente, como se demuestra en la siguiente tabla.



**Tabla 14****Movimientos de la mano para gobernar el Dron\***

MPU-6050		
Movimientos de la mano	Control del Dron	Ángulo de giro (º)
Pronación	Neutro	0º
Supinación/ Pronación media	A la derecha	90º Roll
Combinado (mano-codo)	A la izquierda	-90º Roll
Flexión	Hacia adelante	-80º Pitch
Extensión	Hacia atrás	70º Pitch

\*El MPU-6050 trabaja con una resolución de 1º para cada ángulo.

### 3.3.2. PROCESAMIENTO DEL DISPOSITIVO MAESTRO

Al guante electrónico se adhiere un Estuche cargador, el mismo que alberga un módulo Arduino Nano. Este módulo se encarga de realizar el procesamiento de los datos recibidos por los sensores descritos en la fase de detección.

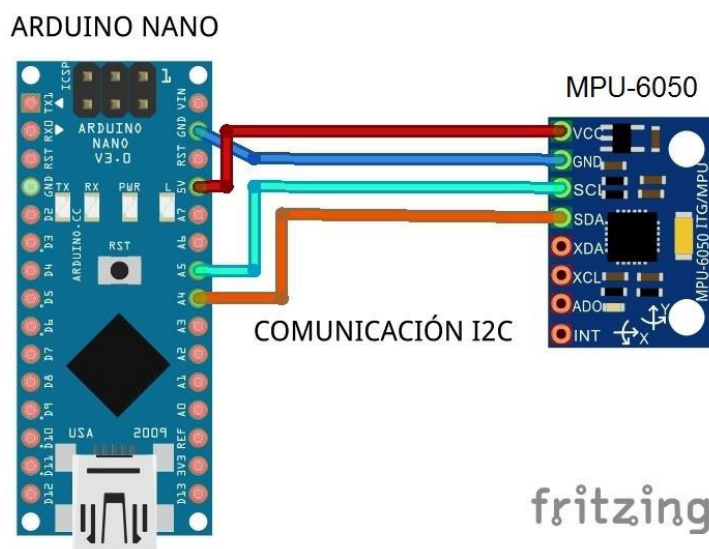
#### ARDUINO NANO

Para realizar la lectura y el procesamiento del guante electrónico, así como del sistema receptor, se seleccionó el módulo Arduino Nano por su tamaño y prestaciones.

#### LECTURA DE LOS SENSORES DEL GUANTE ELECTRÓNICO MEDIANTE ARDUINO NANO

Para el desarrollo del código de programación se comienza con la comunicación entre el Arduino Nano y el MPU-6050. Para esto, se usa el bus

I2C (*Inter-Integrated Circuit*), que tan sólo dispone de dos conexiones. Primero, el SDA por donde se transmiten los datos y, luego, el SCL por el que se transmite la señal de reloj. El Arduino Nano tiene sus conexiones I2C en sus pines A4 y A5, así como también 5V y tierra. (Ver figura 94).



**Figura 94 Conexiones con Arduino Nano, comunicación I2C**

Para establecer la configuración necesaria del sensor primero se declaran las librerías *Wire.h*, *MPU6050.h*, e *I2Cdev.h*. Luego, ya en el bucle de configuración se inicia la comunicación I2C mediante la orden `Wire.begin()`. Al dejar el campo `Wire.begin()` entre paréntesis en blanco se configura el dispositivo como maestro.

Ahora, el MPU-6050 se monta sobre el dorso de la mano derecha. Trabaja en un rango de gravedad de  $\pm 2g$  y envía valores al Arduino en un rango de números enteros entre  $[-16384, +16384]$ . Estos valores se obtienen considerando un ángulo de variación de  $360^\circ$ . Para las condiciones de la mano se tiene otra variación de ángulos. Se realizó la lectura de los movimientos posibles de la mano diestra considerados como *roll* y *pitch* para este proyecto en las posturas necesarias, con una resolución de  $1^\circ$  para cada movimiento en los ejes 'X' e 'Y', respectivamente.

Entonces, de acuerdo al guante electrónico según las posturas de pronación, supinación/pronación media, combinado (mano-codo), extensión, flexión de la mano, se considera únicamente una variación promedio de un rango de números enteros entre  $[-10000,+10000]$  que arroja el MPU-6050 por cada eje, los mismos que mediante la orden *map( )* se han mapeado para trabajar en un rango de valores enteros de 0 a 255 (8 bits). (Ver tabla 15). Este rango de números enteros son con los que trabaja Arduino como se puede observar en el *Monitor serie de Arduino Nano*. (Ver figura 99).

**Tabla 15**

**Conexiones Arduino Nano con MPU-6050**

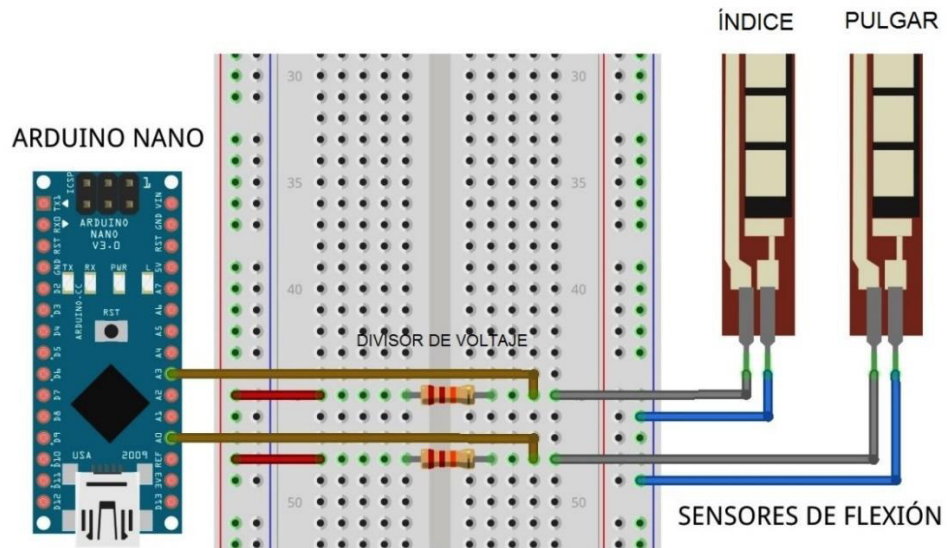
MPU-6050				
<i>Pines Arduino Nano</i>	<b>Comunicación</b>	<b>Resolución (bits)</b>	<b>Precisión (g/LSB)</b>	<b>Sensibilidad (LSB/g)</b>
A4	SDA	8 bits	61 $\mu$	$\pm 10000$
A5	SCL			

\*El MPU-6050 trabaja en un rango de gravedad de  $\pm 2g$

Adicionalmente, se montan dos sensores de flexión en el guante, uno ubicado en el dedo índice y otro en el dedo pulgar, de manera que se pueda doblar en el mismo ángulo que en que se flexionan los dedos. En la figura 95 se observan las conexiones respectivas con las entradas analógicas de Arduino Nano, A0 para el sensor del dedo pulgar y A3 para el sensor del dedo índice. Estos sensores resistivos envían valores a Arduino en un rango de números enteros entre  $[0,1023]$ . Estos valores se obtienen considerando un ángulo de variación de  $180^\circ$ .

Considerando las condiciones de los dedos de la mano, se tiene otra variación de ángulos. Se realizó la lectura de las flexiones posibles de los

dedos de la mano diestra, considerados como asciende/desciende (índice) y *video* (pulgar) para este proyecto, con una precisión de 4.9mV por unidad.



**Figura 95 Sensores de flexión con Arduino Nano, Divisor de Voltaje**

En Arduino Nano declara las variables *sf0* para el dedo pulgar y *sf3* para el dedo índice. Para realizar la lectura de estos sensores se usa la orden *analogRead( )* y en el campo del paréntesis se pone A3 o A0 según el dedo que se desea leer. Por ejemplo, cuando se realiza la flexión/extensión del sensor ubicado en el dedo índice del guante electrónico y se leen los valores recibidos en el pin analógico A3 de Arduino Nano se obtienen valores digitales que oscilan entre 750 y 980. Estos son los valores mínimos y máximos posibles que se producen por la flexión del dedo índice.

Estos valores son realmente la variación útil que se tiene y son, aproximadamente, 230 valores (980-750), los mismos que mediante la orden *map( )* se han mapeado para trabajar en un rango de valores enteros de 0 a 255 (8 bits). (Ver tabla 16). Este rango de números enteros es con el que trabaja Arduino, como se puede observar en el Monitor serie de Arduino Nano. (Ver figura 99).

Tabla 16

## Conexiones Arduino Nano con los sensores de flexión

Sensor de flexión			
<i>Pines</i> <i>Arduino</i> <i>Nano</i>	Dedo de la mano diestra	Resolución (bits)	Precisión (mV)
A0	Pulgar	8 bits	4.9
A3	Índice		

Para el caso del sensor del dedo pulgar, simplemente se emula la función de un botón pulsador o switch. Se debe flexionar y extender el dedo pulgar para que el sensor sobrepase su valor medio que oscila en 900. Para cuando esto sucede, se envía un 1 lógico a Arduino Nano activando la cámara del Dron, la misma que grabará hasta cuando se vuelva a flexionar el dedo sobrepasando un valor de 900 y se envíe un 0 lógico desactivando la cámara del Dron. Si se mantiene extendido el dedo pulgar, la cámara no se activará. Existe un sonido que emite el control del Dron cada vez que se flexiona el dedo y aparece un ícono en la pantalla del mismo, indicando que se está grabando o no. Con estos indicadores se tiene una perfecta noción del momento de grabado.

### 3.3.3. ACONDICIONAMIENTO DE LA SEÑAL ARDUINO-XBEE

El módulo Arduino Nano y XBee reposan sobre el estuche cargador, este cargador suministra a Arduino con un voltaje de 5V. (Ver figura 96). (Cable color rojo: 5V, azul: tierra). A su vez, el módulo XBee se alimenta del pin 3V3 de Arduino (cable color amarillo: 3V, azul tierra).

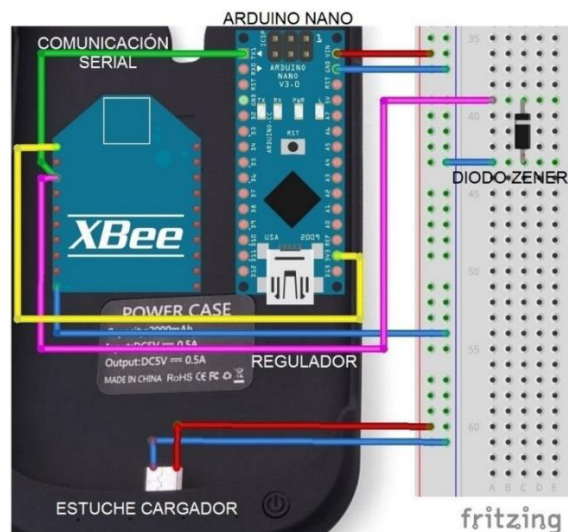
Por el pin TX1 de Arduino Nano (cable color verde: transmisión) se envían los datos con un voltaje de 5V hacia el pin de recepción de XBee (cable color: verde). No obstante, en vista que el módulo XBee trabaja solamente con un voltaje de entre 2.8 a 3.4 V, el primero acondiciona la señal con un diodo

zener 1N4728A. Éste regula el voltaje para que XBee solo obtenga en su entrada 3V (cable color: rosado). (Ver tabla 17).

**Tabla 17**

**Acondicionamiento de la señal Arduino-XBee**

ARDUINO NANO (Envía)	REGULADOR	XBEE (Recibe)
5V	5V => 3V	3V



**Figura 96 Conexiones de XBee con Arduino Nano, Regulador de Voltaje**

### 3.3.4. ESQUEMA DEL GUANTE ELECTRÓNICO

En la siguiente figura se muestra todas las etapas anteriormente expuestas del guante electrónico o dispositivo maestro, tal como serán implementadas posteriormente en un circuito impreso y acopladas al guante diestro Adidas. Realizado en el software gratuito *Fritzing*. (Ver figura 97).



**Figura 97 Dispositivo Maestro**

### 3.3.5. TRANSMISIÓN INALÁMBRICA DE DATOS

El estuche cargador, anteriormente citado, alberga también un módulo XBee S1, que cumple la función de transmitir inalámbricamente los datos hacia el dispositivo esclavo.

## MÓDULO XBEE

Se optó por utilizar el módulo XBEE que cuenta con el estándar de comunicaciones IEEE 802.15.4 para la transmisión inalámbrica de datos, donde el dispositivo fue seleccionado por su bajo consumo de energía, alcance y hardware pequeño compacto para la comunicación entre los dispositivos maestro y esclavo. (Ver tabla 18).

**Tabla 18**

### Especificaciones módulo XBEE S1, transmisión inalámbrica de datos

ESPECIFICACIÓN	XBEE
<b>Performance (Rendimiento)</b>	
<b>Indoor (en interiores)</b>	Hasta 30 metros
<b>Outdoor (en exteriores) RF con línea de vista</b>	Hasta 90 metros
<b>Transmit Power Output (Potencia de transmisión de salida)</b>	1mW (0 dBm)
<b>RF Data Rate (Rango de Datos RF)</b>	250000 bps
<b>Serial Interface Data Rate (Rango de datos de la interfaz serial)</b>	1200 bps – 250 kbps
<b>Sensibilidad del receptor</b>	-92 dBm
<b>Requerimientos de energía</b>	
<b>Voltaje</b>	2.8 – 3.4 V
<b>Corriente de transmisión (típica)</b>	45mA (a 3.3V)

Continúa 

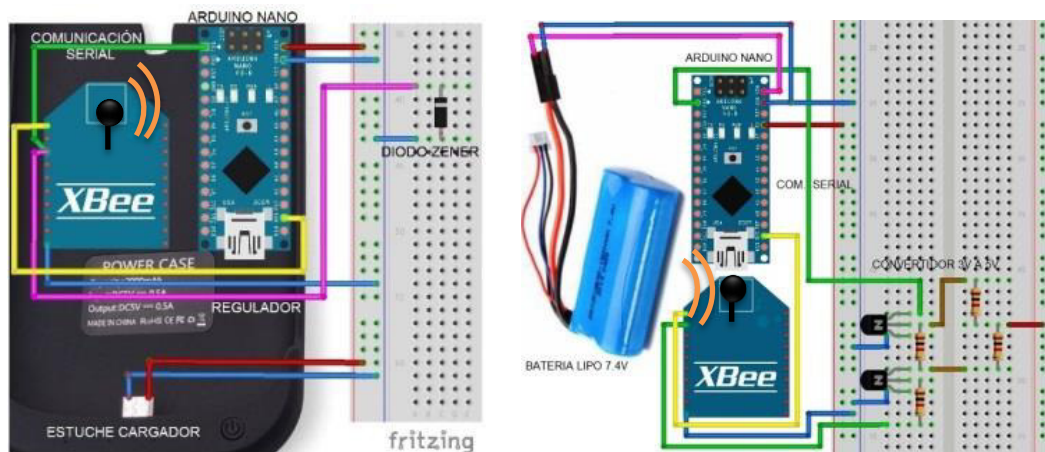


<b>Corriente de recepción (típica)</b>	50mA (a 3.3V)
<b>General</b>	
<b>Frecuencia de Operación</b>	2.4 GHz
<b>Dimensiones</b>	2.438cm x 2.761 cm
<b>Temperatura de operación</b>	-40° a 85° C
<b>Networking &amp; Seguridad</b>	
<b>Soporta las topologías de red</b>	Punto a punto, Punto a multipunto y Peer to Peer
<b>Número de canales</b>	16 secuencias directas de canales
<b>Opciones de direccionamiento</b>	PAN ID, canal y direcciones

Fuente: (Digi International, 2009, pág. 5)

## **ENVÍO DE DATOS DEL GUANTE ELECTRÓNICO AL SISTEMA RECEPTOR**

Un principal objetivo del proyecto ha sido implementar un módulo de comunicación inalámbrica XBee, con motivo de comunicar el dispositivo maestro con el dispositivo esclavo, dotando autonomía al guante electrónico. Para esto, se dispone de 2 módulos XBee S1 fabricados por Digi International y 2 módulos Arduino Nano. Uno de los módulos XBee S1 se encuentra en el estuche cargador que va en el guante electrónico, el mismo que se comunica mediante el puerto serie con uno de los módulos Arduino Nano, según se mostró anteriormente en la figura 96. El segundo módulo XBee S1 se conecta mediante el puerto serie con el segundo módulo Arduino Nano, ambos módulos se encuentran en la caja de acrílico del sistema receptor. (Ver figura 98).



**Figura 98 Comunicación XBEE (izquierda) - sistema receptor (derecha)**

El objetivo es establecer una conexión inalámbrica en tiempo real entre el guante electrónico y el sistema receptor, sistema que a su vez controla los movimientos del Dron. Para esto se prepara un código sencillo en el módulo Arduino Nano, mismo que se comunica por el puerto serie con el módulo XBee del guante electrónico. Estos módulos trabajan conjuntamente. En Arduino Nano se usan las librerías `#include "I2Cdev.h"` e `#include "MPU6050.h"`. Estas fueron desarrolladas por Jeff Rowberg. Así, se declara la variable `char buffer [100]` para almacenar la cadena de caracteres a enviar y luego mediante la instrucción `serial.print(buffer)` se envían los datos al módulo XBee del sistema receptor con un retardo de 100 milisegundos. En esta cadena de caracteres se encuentran los datos de los sensores de flexión y el acelerómetro, tal como se puede ver en la figura 99 en el siguiente formato:



**Figura 99 Monitor serie Arduino Nano, lectura de los sensores**

En el monitor serie se imprime la cadena de caracteres que se obtiene del dispositivo maestro. Se indica un recuadro marcado (ver tabla 19) donde se puede apreciar:

**Tabla 19**

**Interpretación de la cadena de caracteres del monitor serie de Arduino**

		Delay (retardo)
<b>Sensor de flexión del dedo índice (asciende/desciende)</b>	A: 0	
<b>Acelerómetro eje 'X' (roll)</b>	X: 68	
<b>Acelerómetro eje 'Y' (pitch)</b>	Y: 48	100 ms
<b>Sensor de flexión del dedo pulgar (video)</b>	V: 0	
<b>Velocidad de transmisión</b>	9600 baudios	

## MODO DE FUNCIONAMIENTO DE LOS MÓDULOS XBEE

Debido a los sencillos requerimientos de comunicación, entre el guante electrónico y el sistema receptor, se usa una topología punto a punto que no requiere mayor configuración. El módulo que se encuentra en el guante electrónico debe ser configurado como Coordinador de la red Zigbee, debido a que se encuentra en el dispositivo maestro. Éste controla y coordina que los dispositivos puedan conectarse entre sí. Mientras que el módulo que se encuentra en la caja de acrílico del sistema receptor se configura como Dispositivo Final o End Device, mismo que está a la espera de ser activado, aumentando así la duración de sus baterías. En la siguiente tabla 20 se muestran los modos de funcionamiento.

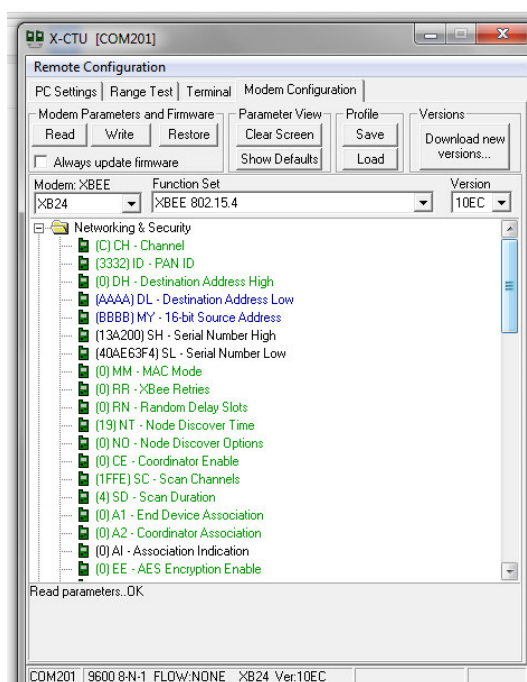
**Tabla 20**

### Modo de funcionamiento

	Guante electrónico	Sistema receptor
<b>Xbee</b>	Coordinador	Dispositivo final

## DIRECCIONAMIENTO DE LOS MÓDULOS XBEE

Para la configuración de los módulos XBee existe un software proporcionado por el fabricante llamado X-CTU, que es una herramienta gratuita que facilita la configuración de los mismos al establecer un contacto directo con el módulo de forma automática y presentar sus distintas características. Esta configuración es posible gracias a la placa con conexión USB, que al momento de conectar el módulo XBee con esta placa el software X-CTU presenta un entorno de configuración completo y de fácil utilización como se muestra en la figura 100. Se hace clic en la pestaña de *Modem Configuration*, visualizar, editar y, luego, guardar los parámetros del módulo.



**Figura 100 Interfaz de comunicación de módulos XBee en el X-CTU**

En la figura 100 se muestra la interfaz de comunicación de módulos XBee en el X-CTU. En ésta el parámetro MY precisa un número de los bits que, en total, resulta de 16. Este establece la dirección en la que el módulo está integrado en sistema. Por otro lado, también se establece el parámetro DL, con el mismo valor numérico del MY. Éste determina la trayectoria del módulo respecto a su destino dentro del sistema. Se precisa tener cuidado que en ambos canales tengan asignado el mismo PAN ID (por defecto) y la misma tasa de transmisión de datos que es de 9600 baudios para este proyecto. (Ver tabla 21).

**Tabla 21**

**Parámetros de configuración para las direcciones de los módulos XBee**

	Parámetro my	Parámetro dl
<b>Xbee guante electrónico</b>	BBBB	AAAA
<b>Xbee sistema receptor</b>	AAAA	BBBB

La ventaja para el guante electrónico es que el usuario puede ubicar a placer el control del Dron en una posición con línea de vista al guante, para una mejor comunicación, a pesar que la distancia también es mínima entre el operador y el control del Dron. Esto permite la comunicación sin ningún problema entre el guante electrónico y el sistema receptor. (Ver tabla 22).

**Tabla 22**

**Características de envío de los módulos XBee**

	Frecuencia de trabajo (hz)	Transmisión de datos (kb/s)	Distancia entre dispositivos (m)	Velocidad de transmisión de datos (bps)
<b>XBee guante electrónico</b>	2.4 GHz	250 kb/s	Hasta 30 m	9600 bps
<b>XBee sistema receptor</b>				

### 3.4. ETAPA 2 – DISPOSITIVO ESCLAVO

#### 3.4.1. SISTEMA DE RECEPCIÓN

Se lo denomina dispositivo esclavo porque es el que recibe las órdenes enviadas por el dispositivo maestro en el sistema planteado. Está conformado por una caja de acrílico transparente y el control del Dron, de hecho la caja y el control están conectados directamente.

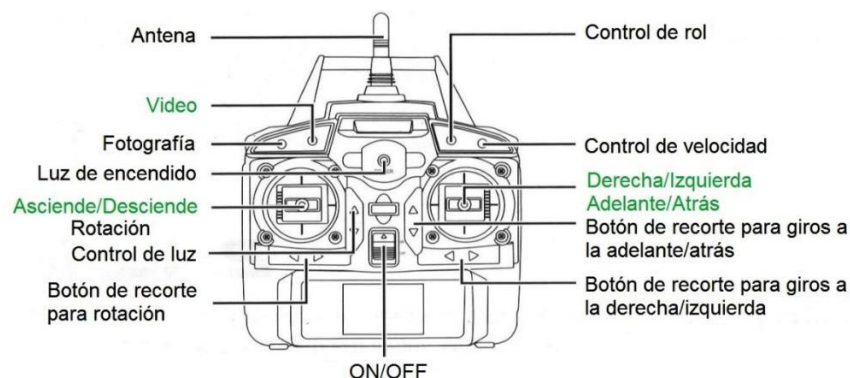
## CARACTERÍSTICAS DEL SISTEMA DE RECEPCIÓN

El sistema receptor es muy liviano. Ambos, la caja y el control, son fáciles de trasladar. El denominado dispositivo esclavo es de fácil manejo por el hecho que solo tiene un botón de encendido/apagado en todo el sistema y cuenta con la pantalla del control para monitorear el estado de la señal, la carga de la batería, el encendido/apagado de la cámara, entre otros. Para lograr esto, se hizo un análisis y se eligen los siguientes elementos electrónicos:

- 3 potenciómetros digitales de 100k $\Omega$  de 256 pasos
- 1 batería recargable LIPO de 7.4V 1500mAh
- 1 cargador para la batería LIPO
- 1 control del Dron X125
- 1 módulo XBee S1
- 1 módulo Arduino Nano

## FUNCIONAMIENTO DEL CONTROL DEL DRON

Se reemplaza el funcionamiento del control del Dron mediante el guante electrónico. Este guante envía las órdenes a los motores del cuadricóptero. Para hacer esto posible, primero se analizó el funcionamiento del control del Dron, al mismo que se le interviene los dos *joysticks* y el botón de video (ver figuras 101, 102, 103 y 104):



**Figura 101 Control del Dron, joysticks y botones<sup>40</sup>**

Fuente: (Shiye, 2015)

<sup>40</sup> Lo que está etiquetado en color verde, son las funciones que se controlan mediante el guante electrónico

Botón de video:

- Para encendido/apagado de la cámara

Joystick 1

- Para ascender/descender el Dron

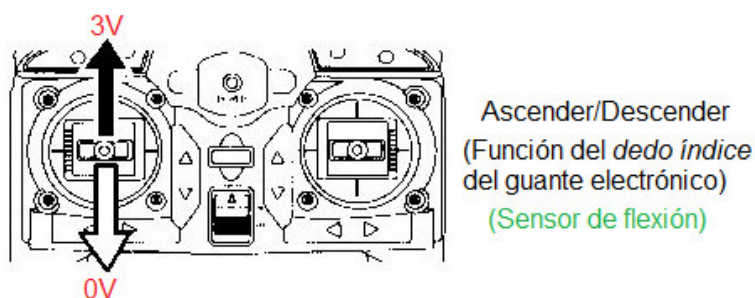


Figura 102 Funciones del Joystick 1

Joystick 2

- Para girar hacia adelante y atrás
- Para girar hacia la derecha e izquierda

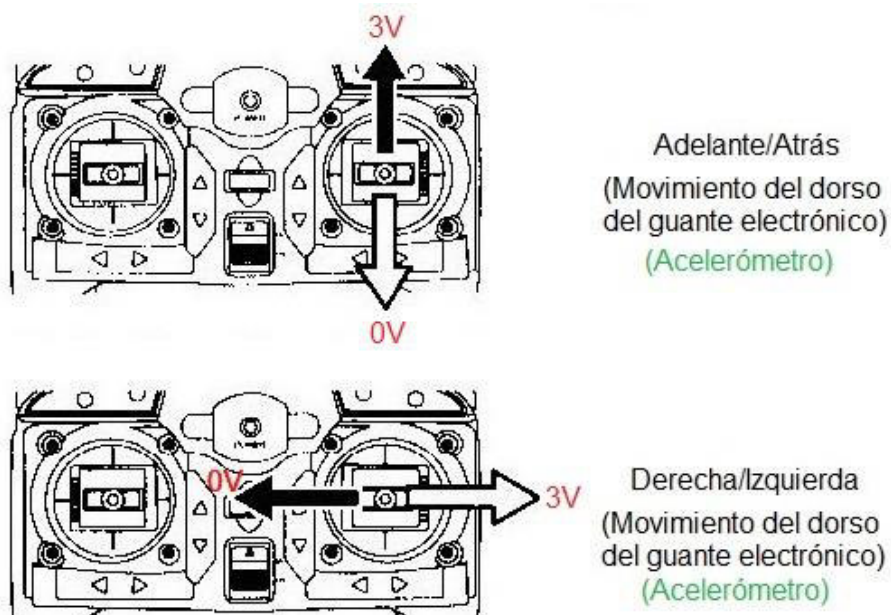


Figura 103 Funciones del Joystick 2

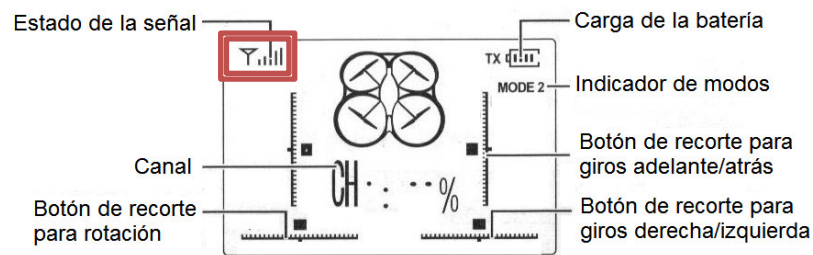


Los *joysticks* del control del Dron funcionan mediante potenciómetros mecánicos, los que tienen una variación desde 0 a 3 voltios. El joystick 2 viene diseñado para mantenerse en una posición neutra. Se coloca en el medio. En esta posición se tiene un voltaje de 1.5V. Esto ocurre para que al momento de emparejar el Dron, éste se mantenga en una posición neutra.

## **EMPAREJAMIENTO DEL CONTROL Y EL DRON**

La preparación para el despegue del Dron se realiza mediante un sencillo emparejamiento entre el control y el Dron, el mismo que es necesario para que el Dron reconozca su mando. En pasos simples sería:

1. Tome el Dron y conecte el switch de la batería.
2. Coloque el Dron sobre el suelo.
3. Asegúrese que usted se encuentra al menos 2 metros detrás del cuadricóptero (las hélices de la cola del Dron deben estar apuntando hacia usted).
4. Inspeccione la ubicación y asegúrese que el Dron se encuentra lejos de las personas, animales o de algún obstáculo.
5. Asegúrese que el control del Dron tenga puestas las baterías AA.
6. Encienda el control del Dron. El indicador de estado de la señal se pondrá a titilar como se muestra en la figura [estado de la señal], ahora presione el joystick 1 (ascender/descender), hasta la parte superior. Luego hale hasta la parte inferior, una subida/bajada del joystick 1. Cuando el indicador de estado de la señal deja de titilar, se produce un tono que indica que el Control del Dron y el Dron se han emparejado exitosamente.
7. Está listo para volar.



**Figura 104 Pantalla del control del Dron**

Una vez emparejados, se enciende el guante electrónico y se puede volar el Dron sin ningún inconveniente debido a que el guante electrónico y el control del Dron están conectados sin necesidad de realizar ningún proceso adicional. Una vez determinados el emparejamiento del dron y el control, así como el funcionamiento de los joysticks del mismo, se procede a explicar el funcionamiento del sistema receptor o dispositivo esclavo.

### 3.4.2. ACONDICIONAMIENTO DE LA SEÑAL XBEE-ARDUINO

Primero se realiza el acondicionamiento de la señal que se recibe del dispositivo maestro mediante el módulo XBee y Arduino Nano. Estos módulos se adhieren a la caja de acrílico y están alimentados por la batería LIPO. Esta batería provee por el pin  $V_{IN}$  a Arduino con un voltaje de 7.4V (cable color rosado: 7.4V, azul: tierra) como se muestra en la figura 107. A su vez, el módulo XBee se alimenta del pin 3V3 de Arduino (cable color amarillo: 3V, azul tierra). (Ver figura 105).

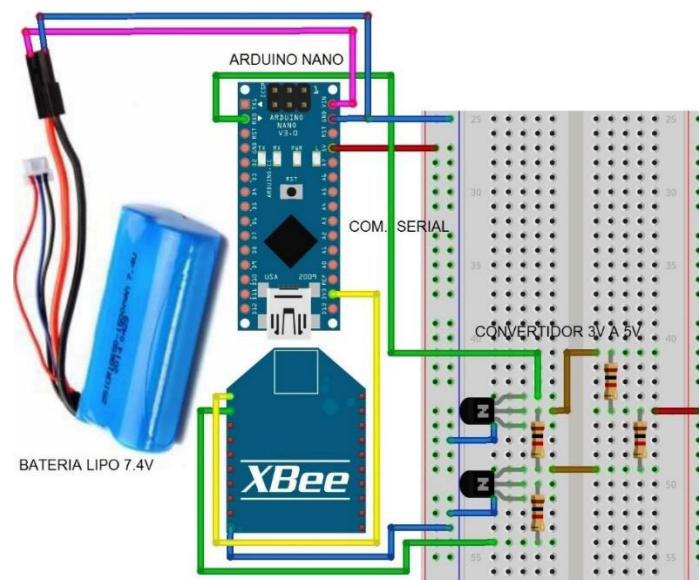
Por el pin RX0 de Arduino Nano (cable color verde: recepción) se deben recibir los datos con un voltaje de 5V desde el pin de transmisión de XBee (cable color: verde), tal como se muestra en la figura [dispositivo esclavo]. Si el módulo XBee envía la señal con un voltaje de entre 2.8 a 3.4V, entonces primero se acondiciona la señal con un convertidor lógico usando transistores 2N3904 y resistencias de 1k para amplificar de 3V a 5V. Este voltaje llegará a

Arduino y recibirá en el pin RX0 solamente una señal de 5V (cable color: verde). (Ver tabla 23).

**Tabla 23**

**Acondicionamiento de la señal XBee-Arduino**

XBEE (Envía)	CONVERTIDOR	ARDUINO NANO (Recibe)
3V	3V => 5V	5V



**Figura 105 XBee con Arduino Nano, convertidor de voltaje**

### 3.4.3. RECEPCIÓN DE DATOS

Los datos son enviados desde el dispositivo maestro al dispositivo esclavo, esta información llega como una cadena de caracteres que se recibe mediante el módulo XBee por comunicación serial previamente inicializada con el comando *Serial.begin(9600)*. Esta cadena de 14 caracteres tiene un

formato como el mostrado en la tabla 24, la misma que se guarda en una variable llamada *inbluet*.

**Tabla 24**

**Cadena de caracteres**

---

A: 0	X: 68	Y: 48	V: 0
------	-------	-------	------

---

Luego para poder interpretar estos datos (A: [0,255], X: [0,255], Y: [0,255], V: [0,1]), se debe hacer una segmentación de la cadena de caracteres mediante el comando *inbluet.substring( )*. Posteriormente, se toman solamente los valores enteros de cada variable. Estos valores enteros como se había explicado antes varían entre 0 a 255 y serán enviados a los potenciómetros digitales. Estos potenciómetros son los que controlaran digitalmente los Joysticks del control del dron.

## POTENCIÓMETROS DIGITALES

Para el control a distancia de los Joysticks 1 y 2, desde el dispositivo maestro se necesita de un elemento que pueda variar su voltaje sin necesidad de tener un contacto manual. Éste debe estar conectado directamente a la placa electrónica del control del Dron. Esa razón ha inspirado al uso de los potenciómetros digitales MCP41100, que se comunican a través de una interfaz SPI. Estos dispositivos realizan la misma función que los potenciómetros mecánicos e, igualmente, tienen tres pines de variación.

Los potenciómetros digitales que se usan en este proyecto son de 100k $\Omega$ . Tienen una resolución de 256 pasos cada 75 $\Omega$  y se configuran para que tenga una variación de 0 a 3V, tal como trabajaban los potenciómetros que se reemplazaron del control del Dron. Como se puede observar en la tabla 25.

Tabla 25

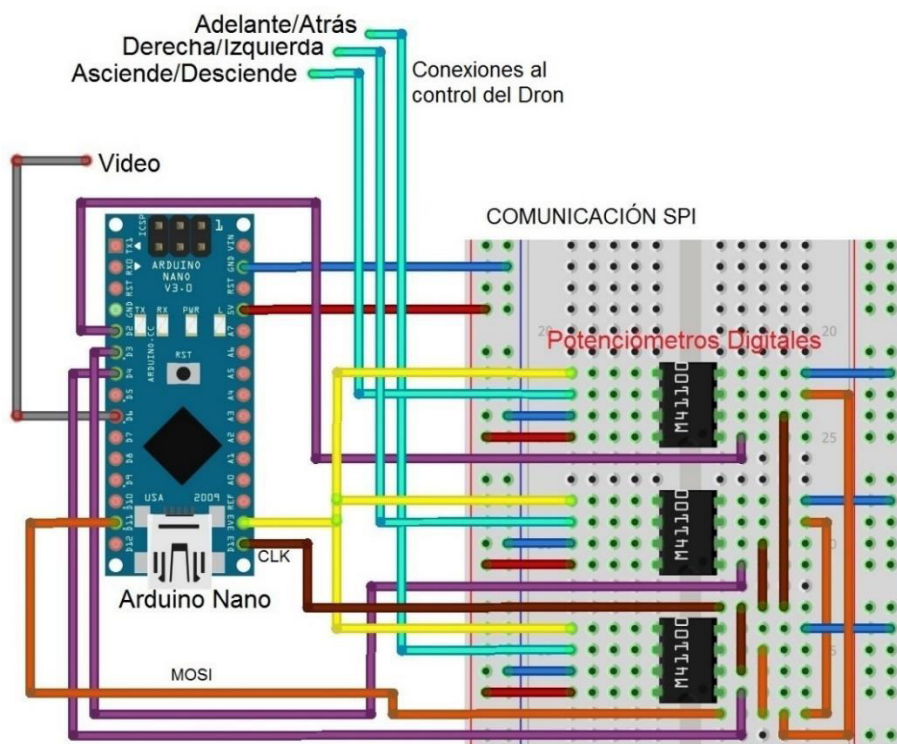
**Características de los potenciómetros digitales**

	Resistencia ( $\Omega$ )	Pasos	Wiper ( $\Omega$ )	Variación de voltaje (V)
<b>Potenciómetros Digitales MCP41100</b>	100K	256	75 $\Omega$	De 0 a 3

**3.4.4. PROCESAMIENTO DEL DISPOSITIVO ESCLAVO****ARDUINO NANO**

Se ha emulado el funcionamiento del control del Dron X125 mediante el guante electrónico, según las funciones más relevantes del control a las que se le asigna un movimiento del guante electrónico. Así, como la flexión de los dedos pulgar e índice como se indicó en el inciso de fase de detección y en funcionamiento del control del dron.

Esta asignación de los movimientos al guante electrónico se realiza a través del dispositivo esclavo, dispositivo que tiene integrado tres potenciómetros digitales conectados directamente a la placa electrónica del control del Dron. Estos elementos reemplazan a los potenciómetros mecánicos del control del dron, cumpliendo la misma función que estos pero de manera digital. Los potenciómetros digitales se comunican a través de la interfaz de comunicación SPI como se muestra en la figura 106.



**Figura 106 Potenciómetros Digitales-Arduino Nano, Comunicación SPI**

Para la transmisión de datos entre Arduino Nano y los potenciómetros digitales denominados como *cs1*, *cs2* y *cs3*, se usa la librería `#include <SPI.h>` y se inicializa la interfaz SPI con el comando `SPI.begin()`. Además, se deben configurar los pines *D2*, *D3* y *D4* de Arduino Nano. Estos pines fueron configurados como salidas mediante el comando `pinMode()` y conectados a los potenciómetros *cs1*, *cs2* y *cs3* respectivamente. Para la etapa de emparejamiento del control del Dron con el cuadricóptero se lo configura mediante el comando `SPI.transfer()`, según los valores máximo y mínimo del *joystick1*. Para poder simular que la subida/bajada necesaria para el emparejamiento exitoso, se ejecuta el potenciómetro *cs1*. Esta etapa de emparejamiento se efectúa automáticamente.

Posteriormente, se necesita que los potenciómetros *cs1*, *cs2* y *cs3* se inicialicen con un cierto valor para asegurar que, al momento de hacer el emparejamiento del Dron con el sistema receptor y a su vez con el guante electrónico, no existan movimientos no deseados. Para esto se realiza la

función *setini()* mediante código en Arduino, que se encarga de enviar un valor de inicialización para cada potenciómetro digital como se muestra en la tabla 26.

**Tabla 26**

**Valores de inicialización de los potenciómetros digitales**

Potenciómetros Digitales	Valor de inicialización (valores enteros)	Asignación de las funciones del control
cs1	0	Ascender/Descender
cs2	128	Derecha/Izquierda
cs3	128	Atrás/Adelante

Luego del envío de los valores de inicialización se procede a enviar los valores que se generan normalmente por los movimientos y flexión del guante electrónico respectivamente, cambiando los valores de los potenciómetros digitales del control del Dron y controlando así los movimientos del Dron. Para esto se tiene la analogía de la tabla 27:

**Tabla 27**

**Movimientos que se realizan mediante los potenciómetros digitales**

Valores enteros	Movimientos del Dron según los Potenciómetros Digitales		
	cs1	cs2	cs3
0	Altura cero	Izquierda	Atrás
128	Altura media	neutro	neutro
255	Máxima altura	Derecha	Adelante

### 3.4.5. ENVÍO DE DATOS

El sensor de flexión ubicado en el dedo pulgar lee los datos cada segundo con un delay(1000), mientras que el sensor de flexión ubicado en el dedo índice, así como los valores que se envían del acelerómetro, se leen cada 10ms. Estos valores son leídos y procesados por el Arduino Nano del sistema receptor y enviados a los potenciómetros digitales que han sido incorporados al control del Dron.

El control del Dron envía las órdenes recibidas por el guante electrónico hacia el cuadricóptero gobernando de esta manera los movimientos. La comunicación entre el control y el Dron se la realiza mediante señal de radio con una frecuencia de 2.4Ghz y un alcance de entre 80 a 100 metros según las condiciones del medio.

**Tabla 28**

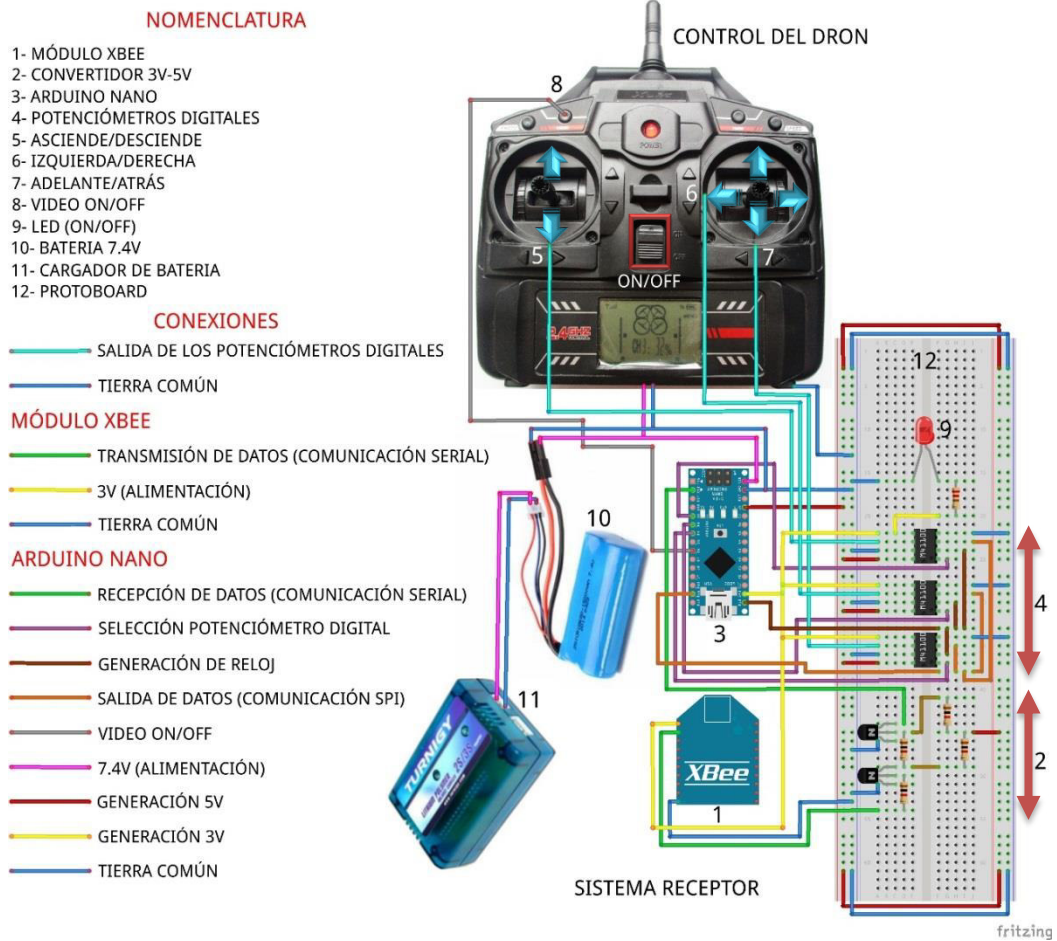
**Envío de datos desde el control del Dron al cuadricóptero**

<i>Control del Dron</i>	
Señal de radio (Hz)	2.4G
Alcance (m)	Entre 80 y 100

### 3.4.6. ESQUEMA DEL SISTEMA RECEPTOR

En la siguiente figura se muestran todas las etapas anteriormente expuestas del sistema receptor o dispositivo esclavo, tal como serán implementadas posteriormente en un circuito impreso y acopladas a la caja de acrílico. Este esquema es realizado en el software gratuito *Fritzing*. (Ver figura 107).





**Figura 107 Dispositivo Esclavo**

### 3.5. IMPLEMENTACIÓN

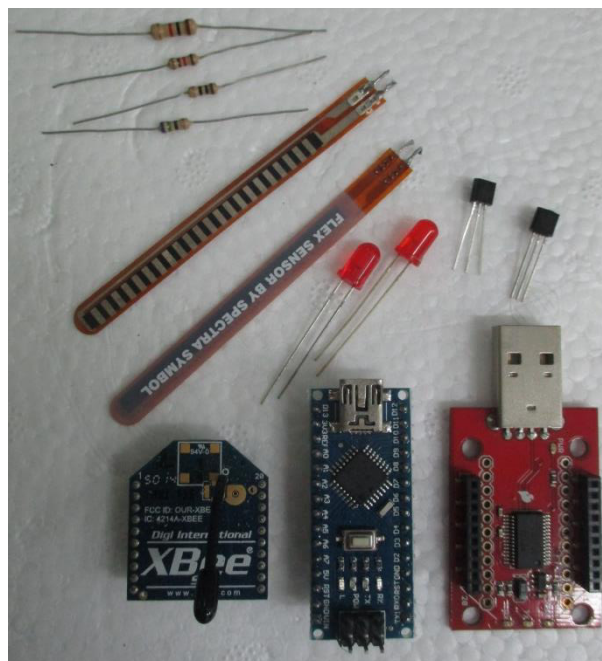
#### 3.5.1. MONTAJE DEL GUANTE ELECTRÓNICO “WILGLOVE”

##### ELEMENTOS USADOS

Para el proceso de implementación, se recurre a los siguientes elementos tales como el módulo XBee y Arduino Nano, mismos que están afianzados sobre una placa electrónica que se encentra sobre el estuche cargador para facilitar su movilidad (ver figura 108). Los insumos a detalle son los siguientes:

- Guante de polyester, diestro
- Estuche cargador de celular S3 mini

- 2 sensores de flexión de 100k $\Omega$
- 1 acelerómetro, IMU MPU-6050 6 grados de libertad ADXL345/ITG3200
- 1 módulo XBee S1
- 1 módulo Arduino Nano
- Resistencias
- Cables



**Figura 108 Elementos usados**

## **GUANTE DIESTRO**

Se toma en cuenta las características de los diferentes elementos que se usan en el desarrollo del guante electrónico con la finalidad de buscar un modelo de guante que reúna todas las exigencias. Se eligió el par de guantes “Adidas AWP 2.0” de color negro, hechos de polyester, puesto que le brindan al prototipo elasticidad, además que son ligeros. La característica más significativa es que tienen un bolsillo oculto en el dorso del guante, que permite cubrir los sensores por protección y estética. Adicional, este tipo de guante

cuenta con extrema transpirabilidad a través de los paneles de los dedos, lo que no interfieren con la ubicación de los sensores de flexión. (Ver figura 109).



**Figura 109 Guante diestro marca Adidas**

### **CARACTERÍSTICAS**

- Guante diestro.
- Material de la cara dorsal de polyester
- Bolsillo oculto en el dorso

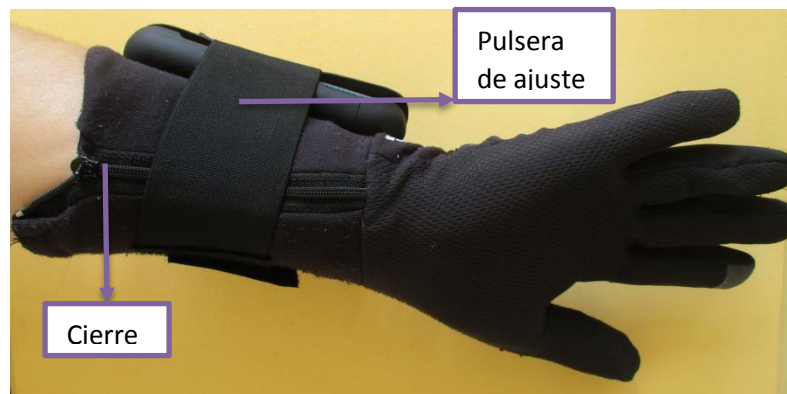
Se optimiza el guante con dos bolsillos adicionales muy pequeños para el tamaño del sensor de flexión. Estos bolsillos se colocan sobre los dedos índice y pulgar, con la finalidad que el usuario no note la presencia de los sensores de flexión, dándole comodidad y una buena imagen por fuera.

### **ESTUCHE CARGADOR DE CELULAR**

Para seguridad del operador se realizó una extensión del guante diestro con el objetivo de dar espacio para añadir el estuche cargador. Esta fuente de poder está adherida al guante con velcro, además usa una pulsera de ajuste.

## CIERRE Y PULSERA DE AJUSTE

El guante cuenta con un cierre ajustable para mayor facilidad al momento de ponérselo como se muestra en la figura 110. La idea es brindar una mayor factibilidad de uso con el objetivo de que esté al alcance de grandes y chicos. Se adapta a la muñeca de la persona que lo usa.



**Figura 110 Cierre y pulsera de ajuste**

Complementario a esto, el guante cuenta con una pulsera de ajuste que es un elástico ancho que está cocido al guante y se usa para sostener fijamente la fuente de poder o estuche cargador (que alberga la placa electrónica). Adicional, la pulsera usa velcro para estar adherido al guante y para dar flexibilidad de ajuste al operador, según el ancho de su antebrazo. La finalidad de este accesorio es dar seguridad al operador para realizar los movimientos de fuerza moderada, sin temer que se desajuste el dispositivo maestro y gobernar el Dron a cabalidad.

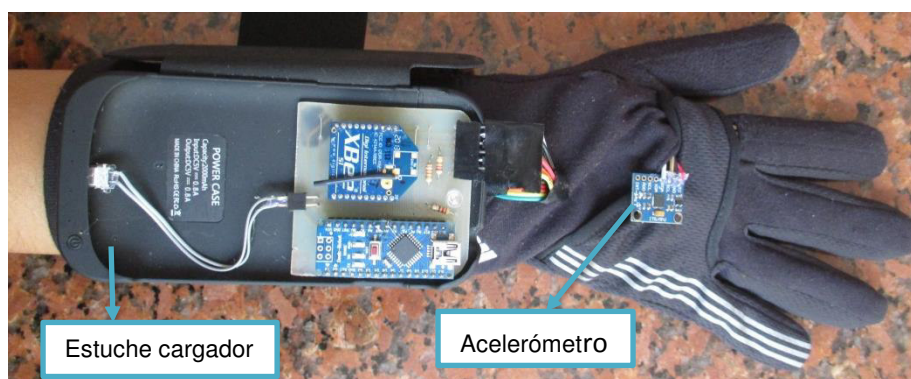
## DISPOSICIÓN DE LOS SENSORES Y DEL ESTUCHE CARGADOR DE CELULAR

Se consideran dos sensores de flexión y un acelerómetro estratégicamente ubicados. (Ver figura 111). El acelerómetro se encuentra en la parte central

del dorso del guante con la finalidad de obtener más precisión al momento de recibir los datos generados por el sensor. Los sensores de flexión, en cambio, son ubicados en el dedo índice y pulgar para facilidad de manejo del operador. Aprovechando el espacio del guante alargado se coloca el estuche cargador a la altura de la muñeca-antebrazo. El propósito es dar libertad en el manejo. Esta ubicación brinda mejores movimientos de la mano al estar su dorso y palma libre de peso. (Ver figura 112).



**Figura 111 Ubicación de los sensores de flexión en el guante**



**Figura 112 Ubicación del acelerómetro y estuche cargador en el guante**

## LOGO “WILGLOVE MH”

Con la finalidad de darle el toque personal al guante electrónico, se realizó un logo, el mismo que se muestra en la figura 113:



Figura 113 Logo “Wilglove MH”

De dónde:

- Wilglove: es el la unión de las palabras *Wilson* y *Glove* (guante en inglés).
- MH: significa *Montúfar Honorable* en honor al Colegio “Juan Pío Montúfar”.

En la figura 114 se presenta el guante terminado y se le llamará “*Wilglove*” de aquí en adelante.



**Figura 114 Guante electrónico “Wilglove MH” (vista superior)**

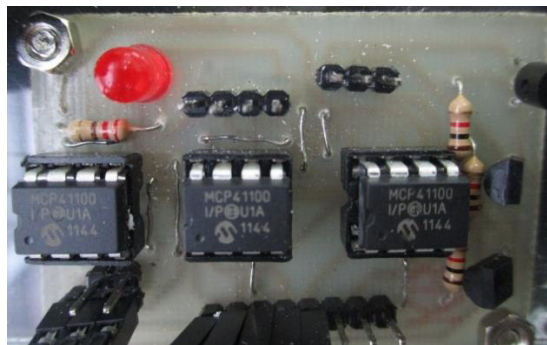
### **3.5.2. MONTAJE DEL SISTEMA RECEPTOR Y CONEXIÓN AL CONTROL DEL DRON**

#### **ELEMENTOS USADOS**

Para la elaboración de este sistema se emplean elementos tales como el módulo XBee y Arduino Nano. Estos están afianzados sobre una placa electrónica que se encuentra dentro de una caja de acrílico, la misma que está adherida al control del Dron para facilitar su movilidad. (Ver figura 115 y 116):

- 1 caja de acrílico (transparente)
- 3 potenciómetros digitales de 100k $\Omega$  (256 pasos)

- 1 batería recargable LIPO de 7.4V 1500mAh
- 1 cargador para la batería LIPO
- 1 control del Dron X125
- 1 módulo XBee S1
- 1 módulo Arduino Nano
- Resistencias
- Transistores
- Tornillos
- Cables



**Figura 115 Placa electrónica de los potenciómetros digitales**

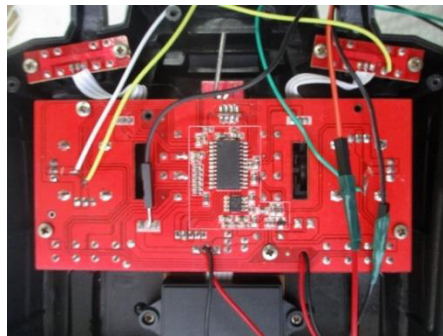


**Figura 116 Control del Dron X125**

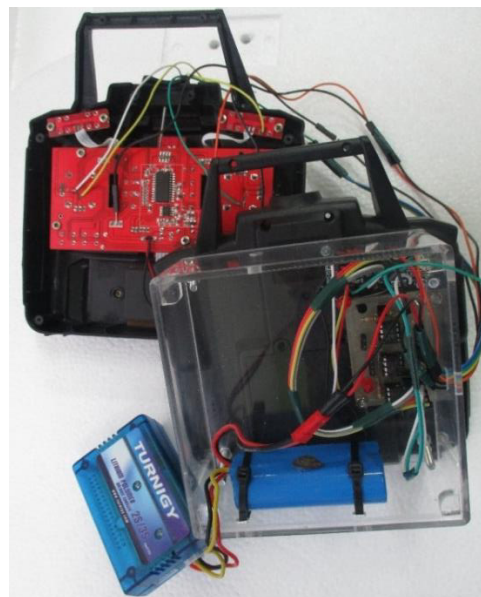


## CAJA DE ACRÍLICO

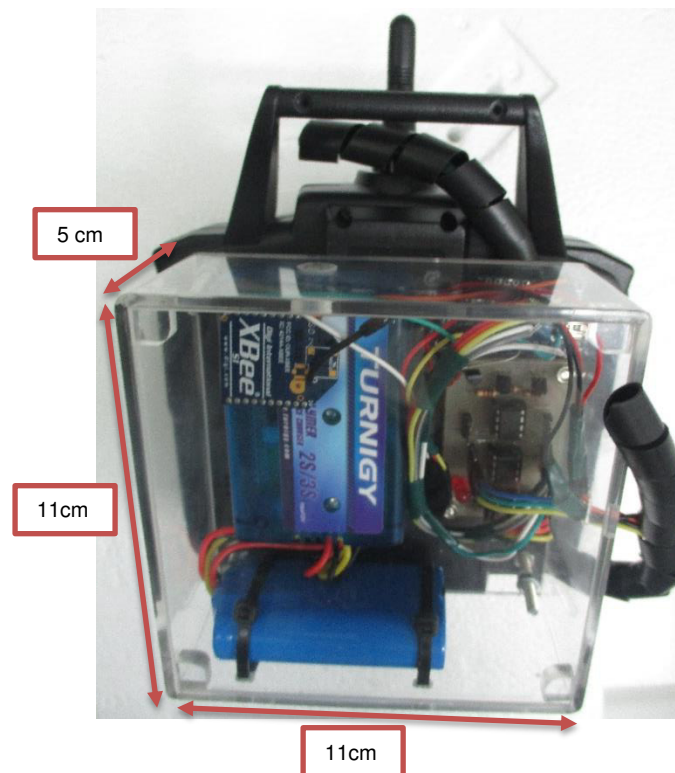
Se considera el tamaño y las características de los diferentes elementos que se usan en el desarrollo del sistema receptor con la finalidad de buscar un modelo de caja óptima que reúna las exigencias. Se eligió una caja transparente, hecha de acrílico que brinde una libre visión y sea liviana. La característica más significativa es brindar protección a los materiales usados y que sea estética. Adicional, esta caja cuenta con una tapa ajustada con tornillos de fácil uso para tener la opción a revisar algún circuito o cambiar la batería si fuere necesario. (Ver figura 119).



**Figura 117 Conexiones a la placa del Control del Dron**



**Figura 118 Conexiones del sistema receptor al control del Dron**



**Figura 119 Vista de la caja de acrílico y el control del Dron**

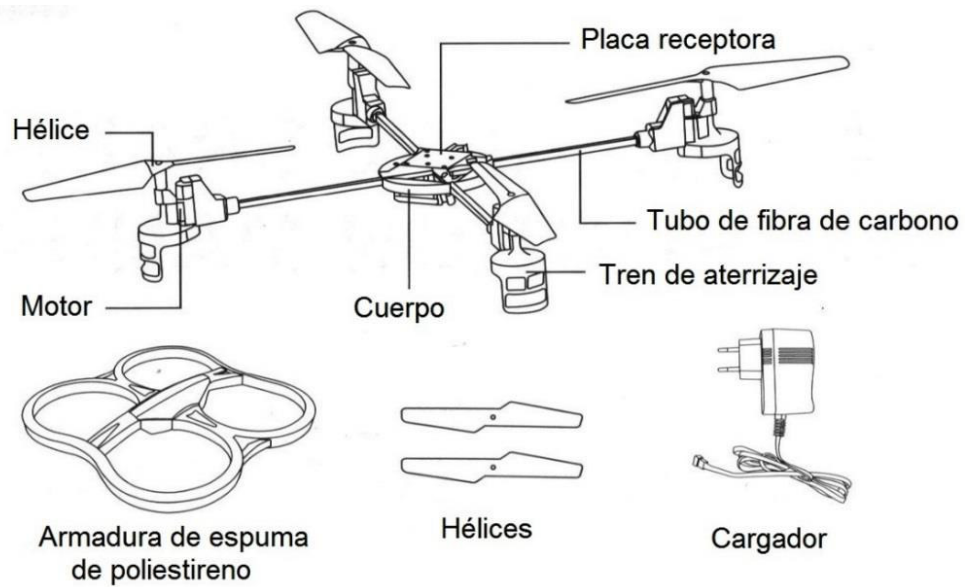
### **CARACTERÍSTICAS**

- Material acrílico
- Transparente para libre visión
- Tapa de fácil uso

### **3.6. DRON X125**

Las drones o naves no tripuladas son capaces de rotar alrededor de tres ejes perpendiculares entre sí y, como cualquier aeronave, tienen un punto de intersección que está situado sobre su centro de gravedad. Estos ejes son los denominados ejes longitudinal, transversal y vertical. Para el caso del *Quadcopter X125*, se controla su orientación variando la dirección del sistema de propulsión. Este sistema se controla modificando el ángulo de ataque de las hélices o la inclinación de los motores provocando los movimientos de *roll*

y *pitch* (el movimiento de *yaw* no se analiza en esta oportunidad). (Ver figura 120 y 121).



**Figura 120 Partes del Dron X125**

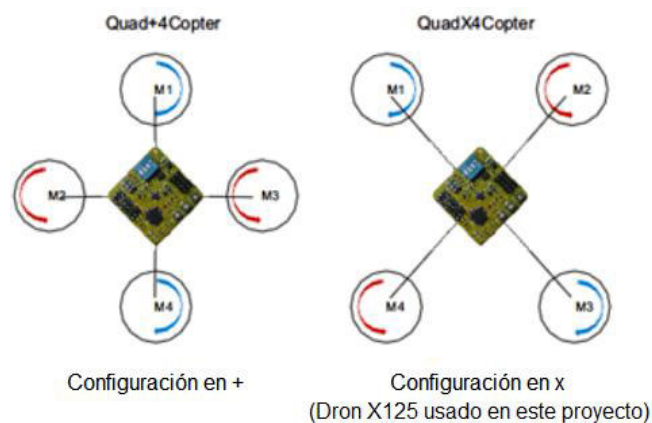
Fuente: (Shiye, 2015)



**Figura 121 Dron X125**

## CONFIGURACIÓN EN “X”

Los drones pueden usar distintas estructuras para el chasis pero las más usadas son en “x” o en “+”(Etxeberria Méndez & Goicoechea Fernández, 2015, pág. 9) como se muestra en la figura 122. La configuración en “x” es la que predomina, ya que presenta brazos que son iguales y el control se sitúa en el centro de gravedad de la estructura. Dentro de las modalidades en “x” las más utilizadas son las de 4, 6 y 8 brazos.



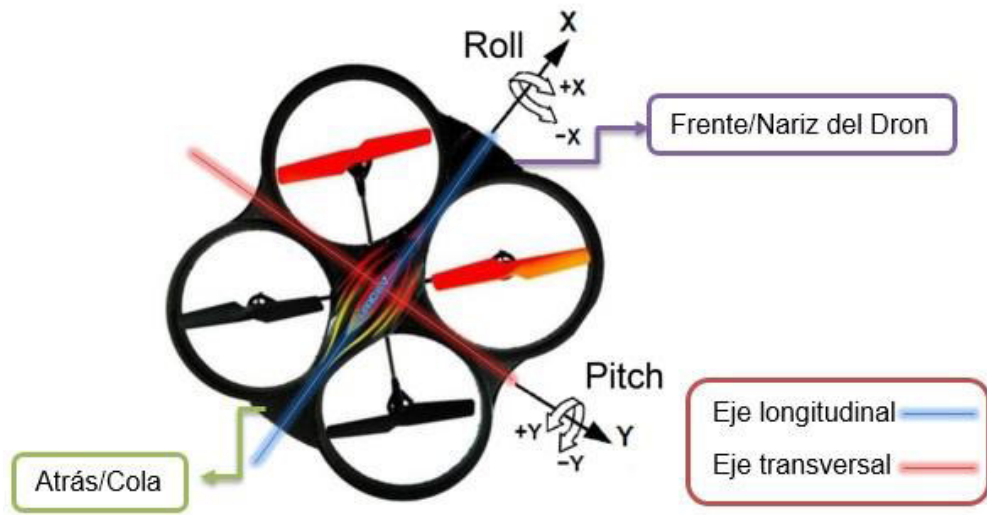
**Figura 122 Tipologías de drones y la disposición de sus motores**

Fuente: (Etxeberria Méndez & Goicoechea Fernández, 2015, pág. 9)

El Dron X125, usado en este proyecto, tiene una configuración en “x”. En detalle se puede acotar que con dicha configuración el aparato gana estabilidad porque se genera un mejor torque y la cámara tiene un mejor ángulo de visión.

## FUNCIONAMIENTO DEL DRON MEDIANTE EL GUANTE ELECTRÓNICO

Para tener una mejor comprensión de la relación entre el guante y el Dron se presenta en la figura 123 la configuración de los ángulos *pitch* (eje longitudinal) y *roll* (eje transversal) respecto del Dron:



**Figura 123 Movimientos de los ángulos *roll* y *pitch* respecto del Dron**

Se puede observar la figura 89 con los ejes de referencia del guante electrónico para una mejor comprensión de la relación.

### **EJE LONGITUDINAL (ROLL)**

El eje roll se proyecta a partir de la frente/nariz del Dron hasta la parte trasera o de la cola como se muestra en la figura 124. “El movimiento que realiza el dron alrededor de este eje se denomina alabeo” (Quadcopter Drone Flyers, 2015).

Este alabeo se produce gracias a la compensación que existe entre los motores. Cuando se gira el guante electrónico hacia la derecha, en un determinado ángulo, provoca que los motores de la parte izquierda del Dron produzcan un aumento de sustentación y se eleve, así como también hace que los motores de la parte derecha del Dron produzcan una disminución de sustentación haciendo que descienda. Esto desplaza el Dron a la derecha. Así, realiza el mismo mecanismo pero a la inversa para cuando desea girar a la izquierda.

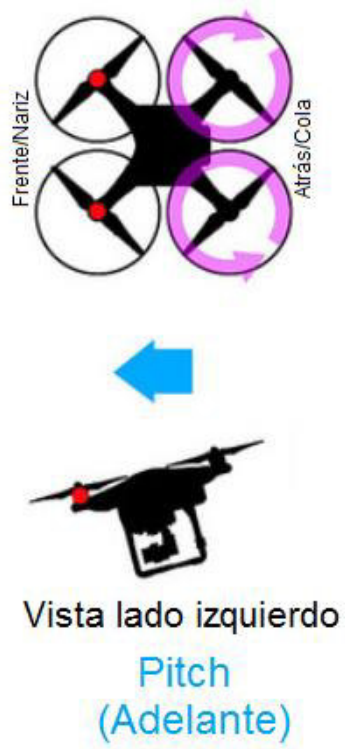


**Figura 124 Compensación de los motores del Dron en *roll***

Fuente: (Stephens, 2014)

### **EJE TRANSVERSAL (PITCH)**

Conocido como eje lateral se proyecta a partir de punta a punta de los lados del Dron como se muestra en la figura 125. Este movimiento que realiza el Dron se denomina cabeceo (Quadcopter Drone Flyers, 2015). Este cabeceo se produce gracias a la compensación que existe entre los motores. Cuando se gira el guante electrónico hacia la adelante, en un determinado ángulo, provoca que los motores de la parte trasera del Dron produzcan un aumento de sustentación y se eleve. Además, hace que los motores de la parte delantera del Dron produzcan una disminución de sustentación haciendo que descienda, desplazando el Dron hacia adelante. Así, realiza el mismo mecanismo pero a la inversa para cuando desea girar hacia atrás.



**Figura 125 Compensación de los motores del Dron en *pitch***

Fuente: (Stephens, 2014)

## CAPÍTULO IV

### PRUEBAS Y RESULTADOS

#### 4.1. PRUEBAS DE FUNCIONAMIENTO DEL GUANTE ELECTRÓNICO Y EL SISTEMA RECEPTOR

Para una exitosa teleoperación del Dron se debe verificar el correcto funcionamiento del guante electrónico, así como también del sistema receptor. Como se puede observar en las figuras 126 y 127 se tienen los circuitos iniciales con los que se realizaron las pruebas preliminares.

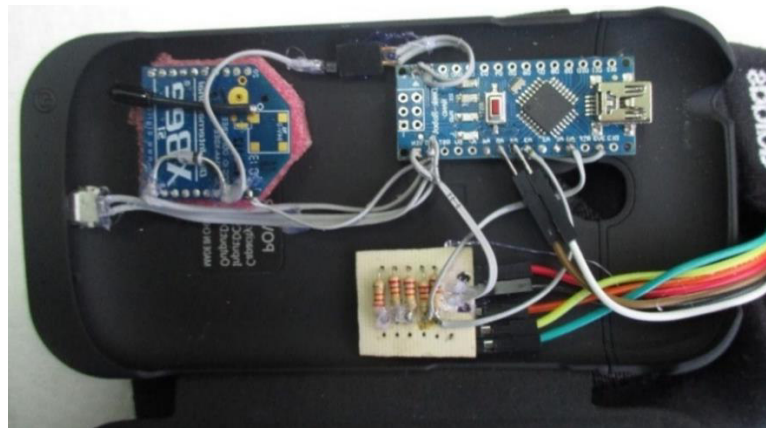


Figura 126 Estuche cargador usado para pruebas, dispositivo maestro

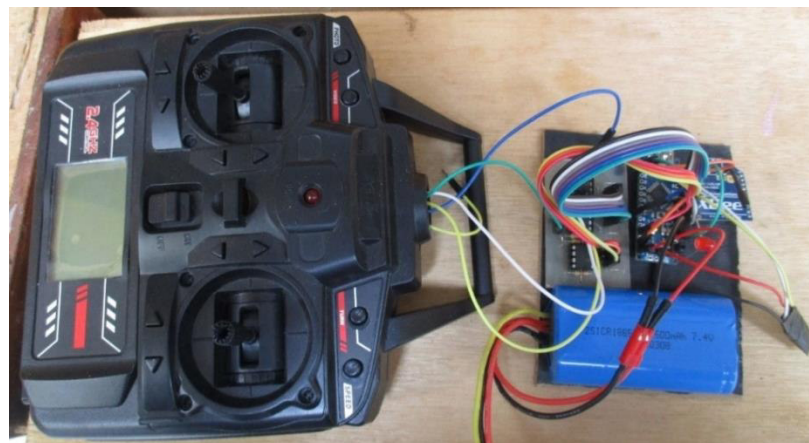


Figura 127 Sistema receptor usado para pruebas, dispositivo esclavo



Para la elaboración del guante electrónico Wilglove se efectuaron pruebas de movimiento y flexionamiento de los sensores mediante el monitor serie de Arduino, con objeto de evaluar y verificar una correcta configuración. Esto fue posible gracias a una conexión directa del guante con la computadora a través del puerto de configuración de Arduino Nano. Posteriormente, se efectuaron pruebas de comunicación con el sistema receptor. Una vez verificada la comunicación entre el Wilglove y el sistema receptor, se procede a realizar las pruebas de comunicación con el Dron.

## **PRUEBAS DE TELEOPERACIÓN DEL DRON MEDIANTE EL GUANTE ELECTRÓNICO Y EL SISTEMA RECEPTOR**

Para esta parte es necesario un encendido del sistema previo para poder realizar las pruebas de vuelo del Dron.

### **ENCENDIDO DEL SISTEMA**

Para el uso correcto del sistema se debe seguir esta guía de pasos:

1. Prender el Dron.
2. Ubicar el Dron en el suelo con las hélices de color negro hacia el usuario. Se recomienda ubicarlo a 2 metros de distancia con el usuario. Verificar que no existan obstáculos para un libre vuelo del cuadricóptero.
3. Encender el control del Dron y esperar el aviso, al menos 2 segundos de espera, mismo que se activa mediante un sonido que produce el control.
4. Ponerse el guante electrónico en la mano derecha, cerrar el cierre y el velcro de sujeción. Ubicar la mano en posición de pronación, con los dedos estirados.
5. Encender el guante electrónico.
6. Flexionar el dedo índice suavemente hasta que el Dron se eleve moderadamente.

7. Volar el Dron libremente, controlando sus movimientos con el dorso de la mano y su altura con el dedo índice. La distancia máxima es de 80-100 metros desde el sistema receptor al Dron.
8. Flexionar el dedo pulgar cuando se desee iniciar o detener videos.

Una vez hecho el encendido del sistema y realizado una verificación del correcto funcionamiento se procede a evaluar el proyecto en diferentes escenarios.

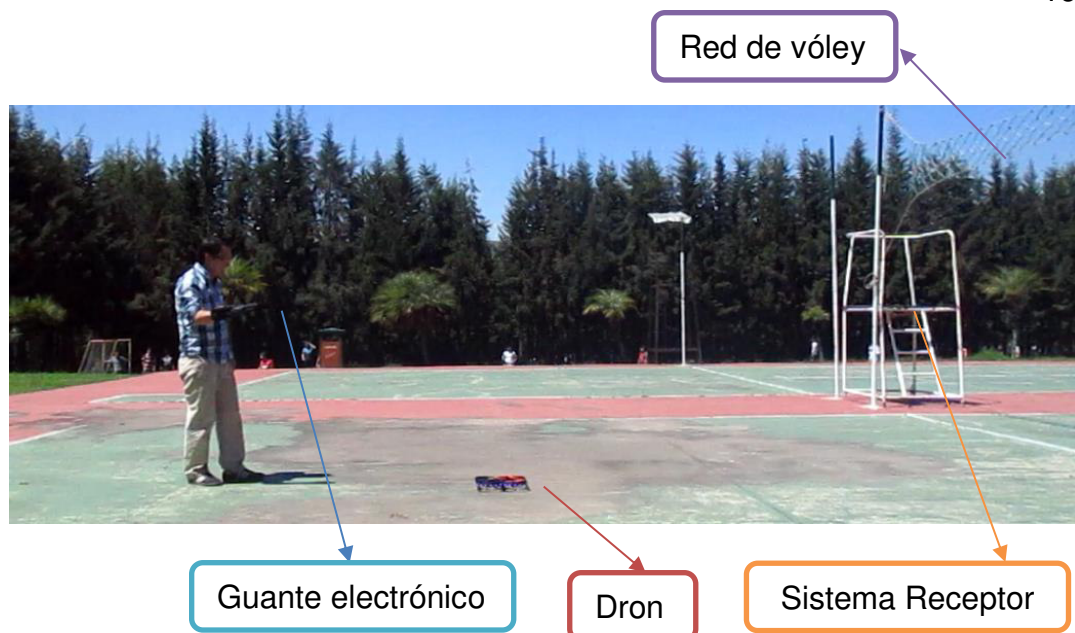
#### **4.2. ESCENARIOS DE EVALUACIÓN**

Se realizaron varias pruebas en distintos espacios de la Universidad de las Fuerzas Armadas – ESPE. Entre los espacios se pueden citar:

- Las afueras de los laboratorios del Departamento de Eléctrica y Electrónica
- Patio central de la Universidad
- Canchas de básquet
- Canchas de fútbol
- Dentro del coliseo Miguel Iturralde
- Las afueras del coliseo Miguel Iturralde
- Parqueaderos

#### **RETO RED DE VÓLEY (VUELO BAJO)**

Entre las pruebas realizadas se planteó el reto de pasar el dron por encima de una red de vóley desde un lado de la cancha y aterrizar sin inconvenientes en el otro lado de la misma.



**Figura 128 Escenario para el reto de la red de vóley**

En la figura 128 se observa el escenario en el que se hace el reto. Se ubica el sistema receptor en la estructura para la visera del juez de vóley, el Dron en un lado de la cancha y el operador está a dos metros de distancia del cuadricóptero, como es lo recomendado. El operador controla el Dron desde el guante electrónico. En la Tabla 29, se tienen dos columnas. La columna de la izquierda muestra las imágenes del reto de la red de vóley desde una cámara externa. La columna de la derecha muestra a su vez las imágenes del reto desde la cámara a bordo del Dron consecutivamente, es decir, cada imagen de la columna de la izquierda concuerda con la de la derecha.

Tabla 29

## Vistas desde la cámara a bordo del Dron y una cámara externa



Como se puede observar en las imágenes, el Dron se eleva 4 metros, aproximadamente. Luego, sobrepasa la red de vóley y aterriza sin inconvenientes en la cancha.

## **PANORÁMICAS DESDE LA CÁMARA A BORDO DEL DRON (VUELO ALTO)**

Las siguientes fotografías se tomaron con el Dron en vuelo. En las figuras 129, 130, 131 y 132 se muestran las fotos tomadas desde la cámara a bordo del Dron. Las imágenes panorámicas son obtenidas a una altura aproximada de 80 metros desde el patio central de la Universidad de las Fuerzas Armadas-ESPE. En las cuales se puede observar la biblioteca, el edificio administrativo y el patio central.



**Figura 129 Vista panorámica de la biblioteca y patio central de la ESPE**



**Figura 130 Vista panorámica de la biblioteca y el edificio administrativo**



**Figura 131 Vista panorámica de la biblioteca y el patio central**



**Figura 132 Vista panorámica del patio central de la ESPE**

En las figuras 133, 134 y 135 se presentan las vistas panorámicas obtenidas a una altura de aproximadamente 60 y 70 metros. Las mismas que fueron tomadas mediante la cámara a bordo del Dron. Estas, muestran el sello característico de la ESPE del edificio central desde un ángulo interesante.



**Figura 133 Vista panorámica del Logo de la ESPE**



**Figura 134** Vista panorámica del Logo de la ESPE y paisaje



**Figura 135** Vista panorámica del Logo de la ESPE



Asimismo, en las figuras 136, 137, 138 y 139 se puede observar los laboratorios de ingeniería electrónica. Estas imágenes fueron tomadas durante un solo vuelo del Dron a una altura de 20 metros, aproximadamente. Se puede observar como el dron se aleja del techo del laboratorio paulatinamente. Estas fotos panorámicas son sin duda una vista privilegiada que se ha obtenido a través de la cámara a bordo del Dron.



**Figura 136 Panorámica de los Laboratorios de Ingeniería Electrónica**



**Figura 137 Panorámica de los Laboratorios de Ingeniería Electrónica**



**Figura 138** Panorámica de los Laboratorios de Ingeniería Electrónica



**Figura 139** Panorámica de los Laboratorios de Ingeniería Electrónica

Como se puede observar en la figura 140, el dron despegue desde la hierba de las afueras de los laboratorios de electrónica.



**Figura 140 Despegue del Dron en las afueras de los laboratorios**

En la figura 141, se toma una foto de las afueras del laboratorio de ingeniería electrónica a una altura de 5 metros aproximadamente.



**Figura 141 Parqueaderos de los Laboratorios de Ingeniería Electrónica**

En las figuras 142 y 143 se pueden observar las canchas de ingeniería mecánica. Las imágenes fueron obtenidas a una altura de 30 metros, aproximadamente. En particular en la figura 143 se tiene un hermoso paisaje y se puede ver gran parte de la universidad.

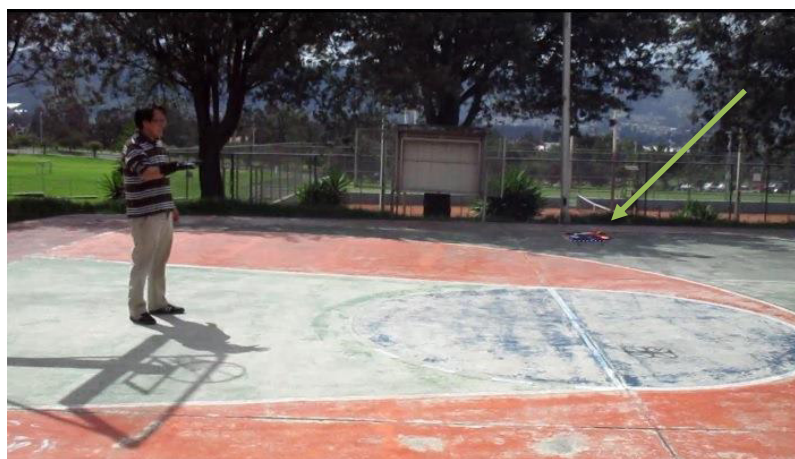


**Figura 142 Panorámica de las canchas del Departamento de Mecánica**



**Figura 143 Panorámica de las canchas del Departamento de Mecánica**

En la figura 144, el Dron se encuentra despegando en las canchas de básquet. Se puede observar un vuelo a baja altura de un metro aproximadamente, realizado por el operador.



**Figura 144 Despegue del Dron en las canchas de Básquet**

Así también, en las mismas canchas de básquet se hizo un vuelo de reconocimiento. Como se observa en las figura 145, se obtiene una vista panorámica a una altura aproximada de 30 metros.



**Figura 145 Vista panorámica de las canchas de Básquet**

Para complementar las vistas obtenidas de toda la universidad, se realizó un vuelo desde la parte de atrás del bar a una altura de unos 25-30 metros aproximadamente. En las figuras 146 y 147 se puede observar parte del parqueadero y el edificio del bar. En la figura 147 particularmente se observa gran parte del edificio central.



**Figura 146 Vista panorámica de los parqueaderos del Bar**



**Figura 147 Vista panorámica del Bar y el edificio central**

## FOTOS TOMADAS DEL VUELO DEL DRON DESDE UNA CÁMARA EXTERNA

A continuación, se muestran fotografías del Dron tomadas desde un punto externo (ver figura 149). En la figura 148 se observa el Dron volando a una altura aproximada de 60 a 70 metros (se señala con una flecha roja en la imagen 148).



**Figura 148 Foto tomada al Dron en vuelo alto**



**Figura 149 Foto tomada al Dron en vuelo bajo**

### **4.3. RESULTADOS DE LAS PRUEBAS DE TELEOPERACIÓN DEL DRON**

Después de haber realizado varias pruebas de vuelo, se pueden presentar los siguientes datos:

- Autonomía del Dron de 15 minutos de vuelo aproximadamente,
- Maniobrabilidad de un 90% respecto al control original del Dron,
- Fácil manejo, y
- Tiempo de respuesta del Dron con respecto al guante electrónico, oscila en el rango de los microsegundos.

Las pruebas se realizaron a vuelo bajo y vuelo alto. En ambientes externos e internos.

#### **VUELO BAJO**

Para un vuelo bajo del Dron se realizaron pruebas en ambientes controlados, que presenten las mejores condiciones. Como el patio de una casa o en el coliseo Miguel Iturralde de la universidad. En el coliseo por ejemplo, no existe la influencia de un viento fuerte, lo que permitió un control absoluto de las maniobras del cuadricóptero. Se puede decir que este ambiente es ideal para el manejo puesto que no se tiene viento fuerte que pueda desequilibrar el Dron al momento que está en el aire. El vuelo que se realizó con estas condiciones fue muy preciso y estable a una altura aproximada de 8 metros.

#### **VUELO ALTO**

Para un vuelo alto del Dron se realizaron pruebas al aire libre. Como las canchas de fútbol o de básquet, el patio central, las afueras de los laboratorios de ingeniería electrónica entre otros. Para estos ambientes, debe haber un viento moderado para poder hacer un vuelo preciso con el cuadricóptero.



Debido a que el Dron se puede desviar o desequilibrar en alturas mayores a 10 metros, aproximadamente. Con viento moderado se puede controlar el Dron sin problemas. Se debe tener en cuenta la distancia límite de 100 metros, porque una vez pasada esta altura se puede perder el control del Dron.

## **DESPEGUE**

El despegue es importante al momento de volar el cuadricóptero. Se realizaron pruebas de despegue desde cemento, hierba y tierra. En el caso de la tierra, se tuvo muchos problemas debido a que al momento de volar se levantaba mucho polvo y esto causaba una grave interferencia en las ranuras de las hélices y motores. Para el caso de la hierba, no se presentaron mayores problemas, debido a que fue en hierba corta. El inconveniente para este escenario fue que solían quedarse residuos de hierba en las rendijas de las hélices. Sin duda, el mejor escenario para despegar fue el cemento, debido a que no existió ningún elemento que haya podido causar daños al Dron.

## **4.4. COSTO DEL PRODUCTO FINAL**

En la siguiente tabla se detalla la lista de materiales, elementos electrónicos y otros elementos utilizados en la implementación del presente proyecto. Además, se determina el costo total del mismo.

**Tabla 30****Costo de los elementos y sensores usados para implementar el proyecto**

No.	CANT.	DETALLE	VALOR UNITARIO (\$)	VALOR TOTAL (\$)
1	1	Dron Quadcopter X125	120,00	120,00
2	1	Guantes Adidas AWP 2.0	20,99	20,99
3	1	Caja de Acrílico	20,00	20,00
4	1	Estuche cargador S3 mini	11,00	11,00
5	2	Arduino nano	19,00	38,00
6	2	XBee s1	39,50	79,00
7	1	XBee Explorer USB	35,00	35,00
8	1	Cable mini USB	2,50	2,50
9	1	Acelerómetro MPU-6050	14,50	14,50
10	1	Diodo zener 3V	0,15	0,15
11	10	Resistencias ¼ W	0,02	0,20
12	4	Capacitores	0,15	0,60
13	3	Transistores 2N2904	0,10	0,30

Continua



<b>14</b>	5	Zócalos de 8 pines	0,12	0,60
<b>15</b>	3	Header regleta	0,60	1,80
<b>16</b>	3	Potenciómetros Digitales	2,50	7,50
<b>17</b>	2	Sensores de flexión 4.5"	10,00	20,00
<b>18</b>	1	Batería de Litio 7.4V Turnigy	18,00	18,00
<b>19</b>	1	Cargador de Batería Turnigy	20,00	20,00
<b>20</b>	1	Adaptador 12V	4,00	4,00
<b>21</b>	1	Otros (cables, tornillos, amarras)	10,00	10,00
<b>22</b>	4	Pilas AA	3,50	3,50
<b>23</b>	3	Elaboración de las Placas Electrónicas	4,00	12,00
<b>TOTAL</b>				<b>439,64</b>

## CAPÍTULO V

### CONCLUSIONES Y RECOMENDACIONES

#### 5.1. CONCLUSIONES

1. El campo de la tecnología se ha desarrollado exponencialmente a lo largo del siglo XXI. Es por esto que, a través de este proyecto se aprovecha las prestaciones que brindan los drones actuales y los guantes electrónicos de nueva generación con motivo de dar ayuda y prestar atención a ciertas necesidades específicas de las personas.
2. Para el desarrollo del prototipo del guante electrónico y el dron se ha implementado el uso satisfactorio de información vinculante a dispositivos electrónicos, sistemas de teleoperación, interfaces, protocolos de comunicación y estándares requeridos dentro del proyecto.
3. La identificación de las condiciones naturales de movimiento corporal humano es imprescindible para la configuración del guante electrónico, especialmente a decir verdad, ya que el prototipo debe reconocer condiciones como flexión de los dedos, movimientos posibles de la mano, entre otro tipo de particularidades específicas de este miembro corpóreo.
4. Por sus condiciones y peculiaridades, el presente prototipo puede ser usado por cualquier persona sin previo conocimiento o capacitación en el control de un master de esta calidad. En este sentido, su empleo puede estar orientado a un público con alguna discapacidad motriz, por lo que el guante electrónico sería un recurso de extensión de la mano que facilite la manipulación de objetos.
5. Los sensores y dispositivos como Arduino Nano, XBee, los sensores de flexión y el estuche cargador son recursos de fácil acceso y manipulación, que en su conjunto conforman el sistema de

teleoperación que induce al máster a controlar y definir el comportamiento del esclavo.

6. Se ha realizado de manera exitosa la implementación de un guante electrónico y un sistema receptor para la teleoperación de un Dron, usando un acelerómetro, dos sensores de flexión, Arduino Nano y XBee entre otros materiales de bajo coste; capaces de interpretar los movimientos de la mano humana y traducirlos a movimientos del Dron y de esta manera gobernar su desplazamiento e incluso realizar videos desde la cámara a bordo del cuadricóptero.
7. Se han implementado dos tarjetas de circuitos impresos, tanto para el guante electrónico como para el sistema receptor. Estos esquemas eléctricos de conexiones se emplean mediante el software Fritzing y Proteus, para luego ser montados y conectados con el guante electrónico y el sistema receptor, respectivamente.
8. La programación del código en Arduino Nano permite la actuación de los distintos dispositivos y sensores de forma coordinada tanto en el guante electrónico como en el sistema receptor, tal es el caso de la comunicación entre los sensores de flexión y acelerómetro en el guante electrónico, así como para comunicarse con XBee.
9. La configuración XBee ha sido implementada con éxito permitiendo la comunicación inalámbrica entre el dispositivo maestro y el dispositivo esclavo. Esto permite el flujo de información inalámbrica entre el guante electrónico y el Dron mediante el sistema de teleoperación.
10. El guante electrónico es el dispositivo que le permite al usuario interactuar con el Dron mediante el sistema de teleoperación, debido a que la dificultad de controlar el Dron depende de la destreza que tiene el usuario para utilizar el guante. Pese a que el operador no requiere de una capacitación en el mando, si debe practicar los movimientos con el dispositivo para su adiestramiento, así como reconocer el sistema antes de realizar un vuelo con el Dron.

11. Los ensayos preliminares mediante los circuitos de prueba conllevaron al análisis y perfeccionamiento en cada etapa de desarrollo. Al momento de realizar los primeros vuelos se obtuvieron varias ideas de cómo mejorar el manejo del Dron, las mismas que se implementaron durante el proyecto. Entre ellas se puede mencionar la mejora en el ajuste del guante a la mano, mediante la fijación de éste con la pulsera de ajuste y velcro. Asimismo, esto induce en una mejora de la configuración de un rango límite de movimiento en los valores del acelerómetro y sensores de flexión según los dedos y dorso de la mano humana promedio.

## **5.2. RECOMENDACIONES**

1. Existen varios tipos de drones desarrollados en base a los requerimientos de este proyecto. En este sentido, se debe incursionar en el uso de otros tipos o clases de mecanismos, sean en variedad como en prototipo, mismos que sean alternativos o suplementarios al guante electrónico.
2. Con motivo de dar uso en otro tipo de espacios y para otro tipo de usos, se recomienda identificar diferente información referente a dispositivos electrónicos, sistemas de teleoperación, interfaces, protocolos de comunicación y estándares requeridos según sea el caso y la necesidad.
3. Se recomienda analizar no sólo movimientos de mano, sino inclusive otros como condiciones sensoriales más pequeñas y complejas, en vista que, este tipo de guantes se pueden utilizar en personas con discapacidades de movilidad corpórea de manos. En este sentido, la investigación se expande a condiciones que van mucho más allá de un simple movimiento.
4. Se recomienda testear este guante con otro tipo de objeto esclavo a control remoto, así como probarlo como una extensión de otra parte del cuerpo humano a discreción.

5. Se recomienda promover el testeado de esclavos a través de la implementación alternativa de masters tales como joystick, brazos robóticos o cualquier otro con este principio.
6. En la implementación se realizó un proceso técnico interesante, el mismo que se basó en las ideas innovadoras para minimizar el uso de dispositivos para la comunicación inalámbrica, así como también de los sensores con el fin de optimizar el costo de construcción. Se recomienda adquirir los materiales en tiendas online como Amazon, Sparkfun o eBay.
7. Se recomienda indagar en la implementación de otro tipo de tarjetas y sistemas; tales como las Tarjetas de Desarrollo FPGA u otra versión de Arduino, que presenten mejores niveles de procesamiento, así como calidad de sus prestaciones, y que promuevan una mejor eficiencia y calidad del proceso de control a distancia; así como otro tipo de plataformas, como es el caso de LabVIEW, que permite simular sistemas de mayor complejidad.
8. Se recomienda implementar el uso de un dispositivo embebido que permita procesar la información y realizar la comunicación inalámbrica.
9. Se recomienda usar una configuración XBee que permita una comunicación o configuración punto a multipunto, con motivo de experimentar la posibilidad de que el control pueda controlar más de un dispositivo esclavo.
10. Se recomienda mediante programación, realizar una calibración automática del guante electrónico, a los movimientos que realice el usuario.
11. Para una futura aplicación se recomienda testear un guante con mayor flexibilidad que pueda acoplarse en manos de diferente tamaño y que evite causar problemas en la configuración de los límites de los rangos de movimientos.

## REFERENCIAS

- Aliaga, I. (Diciembre de 2000). *Arquitecturas de teleoperación con reflexión de fuerza sobre un sistema master-slave de 2GDL*. Obtenido de <http://www4.tecnun.es/asignaturas/control1/proyectos/teleop2D/teleoperacion.htm>
- Alibaba global trade. (2015). Obtenido de [https://www.alibaba.com/product-detail/new-products-2015-xbee-pro-rf\\_60144257718.html](https://www.alibaba.com/product-detail/new-products-2015-xbee-pro-rf_60144257718.html)
- Amazon. (2015). Obtenido de [http://www.amazon.com/XCSOURCE%C2%AE-2000mah-External-Battery-GT-i8190/dp/B00O9JYVZI/ref=sr\\_1\\_3?ie=UTF8&qid=1439425176&sr=8-3&keywords=battery+case+s3+mini](http://www.amazon.com/XCSOURCE%C2%AE-2000mah-External-Battery-GT-i8190/dp/B00O9JYVZI/ref=sr_1_3?ie=UTF8&qid=1439425176&sr=8-3&keywords=battery+case+s3+mini)
- Amazon. (2015). Obtenido de [http://www.amazon.com/1500mAh-2x18650-Connector-Lithium-Polymer/dp/B00XOFUI2S/ref=sr\\_1\\_17?ie=UTF8&qid=1449548626&sr=8-17&keywords=7.4v+1500mah](http://www.amazon.com/1500mAh-2x18650-Connector-Lithium-Polymer/dp/B00XOFUI2S/ref=sr_1_17?ie=UTF8&qid=1449548626&sr=8-17&keywords=7.4v+1500mah)
- Amazon Try prime. (2015). Obtenido de [http://www.amazon.com/6-Axis-Remote-Control-Quadcopter-MicroSD/dp/B00WUTVCVU/ref=sr\\_1\\_2?ie=UTF8&qid=1436735926&sr=8-2&keywords=Quadcopter+x125](http://www.amazon.com/6-Axis-Remote-Control-Quadcopter-MicroSD/dp/B00WUTVCVU/ref=sr_1_2?ie=UTF8&qid=1436735926&sr=8-2&keywords=Quadcopter+x125)
- Anderson, C. (2014). *MIT Technology Review*. Obtenido de <http://www.technologyreview.com/featuredstory/526491/agricultural-drones/>
- Anónimo. (s.f.). Obtenido de <http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/3D/Realidad%20Virtual/web/dispositivos/guantes.html>
- Arduino. (s.f.). Obtenido de <https://www.arduino.cc/en/Main/ArduinoBoardNano>
- Arduino. (s.f.). *Arduino*. Obtenido de <http://playground.arduino.cc/Main/MPU-6050>
- Banzi, M. (2008). *Arduino*. Obtenido de <https://www.arduino.cc/en/Main/AboutUs>
- Carletti, E. (2007). *Robots Argentina*. Obtenido de Comunicación Bus I2C: [http://robots-argentina.com.ar/Comunicacion\\_busI2C.htm](http://robots-argentina.com.ar/Comunicacion_busI2C.htm)
- Chacón Mosquera, E. F. (2013). *Desarrollo de una interfaz para el reconocimiento automático del lenguaje de signos*. Sangolquí: ESPE.



- Coquet, E. (1999). *Electrónicos*. Obtenido de <http://www.comunidadelectronicos.com/articulos/i2c.htm>
- Cortes, J. P. (2015). *Academia*. Obtenido de Academia: [http://www.academia.edu/7946946/QU%C3%89\\_ES\\_UN\\_DRONE](http://www.academia.edu/7946946/QU%C3%89_ES_UN_DRONE)
- Cortes-Monroy, P. (4 de Julio de 2009). *Fayerwayer*. Obtenido de <https://www.fayerwayer.com/2009/07/acceleglove-guante-programable-con-acelerometro/>
- Devices, A. (2009). *Digital Accelerometer*. Norwood: ANALOG DEVICES.
- Digi International, D. (2009). *Xbee/Xbee-PRO Modules*. Minnetonka: DIGI INTERNATIONAL.
- Domodesk. (3 de Septiembre de 1999). Obtenido de <http://www.domodesk.com/a-fondo-zigbee>
- Ebay. (s.f.). Obtenido de <http://www.ebay.com/itm/Turnigy-12V-2S-3S-Lipo-Battery-Balance-Charger-For-RC-Helicopter-Quadcopter-NEW-/281704256991?var=&hash=item4196e1d5df:m:mLit7pDPeWUiJDxOavHHIDw>
- (2010). Electrónica Digital. En *Electricidad* (págs. 13-14). Cartagena: UNIVERSIDAD POLITÉCNICA DE CARTAGENA.
- Etxeberria Méndez, J., & Goicoechea Fernández, J. (2015). *Implementación de un dron cuadricóptero con Arduino*. Pamplona : UNIVERSIDAD PÚBLICA DE NAVARRA.
- Franco, S. (27 de Mayo de 2007). *Locos robots para rehabilitación y fisioterapia en Gifu, Japón*. Obtenido de [http://www.rehabilitacionblog.com/2011\\_05\\_01\\_archive.html](http://www.rehabilitacionblog.com/2011_05_01_archive.html)
- Frank, R. (30 de Junio de 2010). *SensorTips* . Obtenido de <http://www.sensortips.com/featured/yole-developpement-forecasts-growth-for-mems-sensors-in-smartphones/>
- Fritzing. (2015). Dortmund, Alemania.
- Gómez, M. (23 de Abril de 2013). Obtenido de <http://fuenteabierta.teubi.co/2013/04/el-convertidor-analogo-digital-en.html>
- Gómez, M. (23 de abril de 2013). *Código Tips*. Obtenido de <http://fuenteabierta.teubi.co/2013/04/el-convertidor-analogo-digital-en.html>
- Grad. (10 de Enero de 2011). *Sparkfun*. Obtenido de <https://www.sparkfun.com/tutorials/240>
- Gravitech. (s.f.). *Arduino Nano 3.0 User Manual*. CREATIVE COMMONS.

- Grusin, M. (2010). *Sparkfun*. Obtenido de <https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi>
- Grusin, M. (9 de Agosto de 2011). *Sparkfun*. Obtenido de <https://www.sparkfun.com/tutorials/270>
- Gutierrez, J. (23 de Diciembre de 2015). *UniversoDrones*. Obtenido de <http://universodrones.com/>
- Hand telerehabilitation system*. (2015). Obtenido de <http://4.bp.blogspot.com/-KeJkNtj870/TdpT6gHwwwl/AAAAAAAAAK68/CFRhnUNZm3U/s1600/Hand+Tele-rehabilitation+System.GIF>
- Hernández, J. (6 de Octubre de 2011). *Electrónica: Entendiendo los potenciómetros digitales*. Obtenido de <http://smdelectronicayalgomas.blogspot.com/2011/10/entendiendo-los-potenciometros.html#.VxD1-PnhDIX>
- Herrera Ñunez, E. F., & Zambrano Chávez, W. O. (2013). *Diseño e Implementación de un guante sensorizado para el control teleoperado de un prototipo de brazo robótico para aplicaciones de manejo de materiales peligrosos*. Latacunga: ESPE.
- IHS Engineering*. (28 de Julio de 2015). Obtenido de <http://insights.globalspec.com/article/1263/specifying-an-accelerometer-function-and-applications>
- InvenSense. (2010). *ITG-3200 Product Specification Revision 1.4*. Sunnyvale: INVENSENSE INC.
- InvenSense. (2011). *MPU6000/MPU6050 9-axis Evaluation Board User guide*. San José: INVENSENSE.
- InvenSense. (2013). *MPU-6000 and MPU-6050 Product Specification*. Sunnyvale: INVENSENSE INC.
- InvenSense. (2013). *MPU-6000 and MPU-6050 Product specification Revision 3.4*. Sunnyvale: INVENSENSE INC.
- Invensense. (2013). *MPU-6050 Product Specification*. Sunnyvale: Invensense.
- InvenSense. (2014). Obtenido de <http://www.invensense.com/products/motion-tracking/6-axis/mpu-6050/>
- InvenSense. (s.f.). *InvenSense Motion*. Obtenido de <http://www.invensense.com/technology/motion/>
- Jimbo. (s.f.). *Sparkfun*. Obtenido de Serial Communication: <https://learn.sparkfun.com/tutorials/serial-communication>

- Kapandji, A. (2006). *Fisiología Articular: hombro, codo, mano, muñeca*. EDITORIAL MÉDICA PANAMERICANA.
- Labtronix. (s.f.). Obtenido de <http://labtronix.co.uk/drupal/content/about-oscilloscope-sample-rate>
- Lanier, J. (9 de Enero de 2016). *Virtual Reality Society*. Obtenido de <http://www.vrs.org.uk/virtual-reality-profiles/vpl-research.html>
- Marla , G., & Moreno, J. (23 de Mayo de 2012). *Zigbee*. Obtenido de <http://sx-de-tx.wikispaces.com/ZIGBEE>
- Martín Doñate, C. (2004). *Interfaces hápticos. Aplicación en entornos virtuales*. Zaragoza: INGEGRAF.
- Maza Alverca, Y., & Berrazueta Solórzano, F. (2005). *Zigbee como una aplicación a la domotica y el estandar IEEE 802.15.4*. Quito: ESCUELA POLITÉCNICA NACIONAL.
- Microchip. (4 de Junio de 2004). Obtenido de <http://www.microchip.com/wwwproducts/en/MCP41100>
- Microchip Technology Inc, M. T. (2003). *MCP 41100*. MICROCHIP.
- Moore, K., & Agur, A. (2008). *Fundamentos de anatomía con orientación clínica*.
- Myduino. (2016). Obtenido de [http://www.myduino.com/index.php?route=product/product&product\\_id=178](http://www.myduino.com/index.php?route=product/product&product_id=178)
- National Instruments. (13 de Agosto de 2009). Obtenido de <http://www.ni.com/white-paper/8939/es/>
- Neoteo. (20 de febrero de 2011). Obtenido de <http://www.neoteo.com/-21690-sensores-inerciales-el-mundo-en-movimiento>
- Nuño Ortega, E., & Basañez Villalengua, L. (2004). *Teleoperación: técnicas, aplicaciones, entorno sensorial y teleoperación inteligente*. Barcelona: UNIVERSIDAD POLITÉCNICA DE CATALUÑA.
- NX. (30 de Julio de 2014). *Tecnonauta*. Obtenido de <http://www.tecnonauta.com/notas/1939-drones-que-es>
- Padilla. (2014). *La Catarina, publicación estudiantil de la Universidad de las Américas de Puebla*. Obtenido de [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lmt/padilla\\_m\\_o/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lmt/padilla_m_o/capitulo2.pdf)
- Pillajo Obando, M., & Robayo Cajamarca, S. (2012). *Desarrollo de un guante "electrónico de datos", con sensores inerciales, herramientas Opensource y comunicación inalámbrica que interactue con imágenes*

*del cuerpo humano en 3D para el proyecto "Sistema de entrenamiento virtual para medicina".* Quito: UNIVERSIDAD POLITÉCNICA SALESIANA.

Postdam, F. (2014). *Fritzing*. Postdam, Brandenburg, Alemania.

Postdam, F. (2014). *Fritzing*. Obtenido de <http://fritzing.org/home/>

*Quadcopter Drone Flyers.* (2015). Obtenido de <http://www.quadcopterflyers.com/2015/02/quadcopters-yaw-roll-and-pitch-defined.html>

Real Academia Española . (18 de Diciembre de 2014). *Fundéu BBVA*. Obtenido de <http://www.fundeu.es/recomendacion/dron-adpatacion-al-espanol-de-drone/>

*Robologs.* (15 de Octubre de 2014). Obtenido de <http://robologs.net/2014/10/15/tutorial-de-arduino-y-mpu-6050/>

Ronzo. (s.f.). *Sparkfun.* Obtenido de [https://learn.sparkfun.com/tutorials/gyroscope?\\_ga=1.39319975.521213662.1436031708](https://learn.sparkfun.com/tutorials/gyroscope?_ga=1.39319975.521213662.1436031708)

Rueda, J. (18 de diciembre de 2010). *Scribd.* Obtenido de <https://es.scribd.com/doc/45559974/25/Teorema-del-muestreo-Teorema-de-Nyquist-Shannon>

Seidle, N. (s.f.). *Sparkfun.* Obtenido de [https://learn.sparkfun.com/tutorials/analog-to-digital-conversion?\\_ga=1.73932054.521213662.1436031708](https://learn.sparkfun.com/tutorials/analog-to-digital-conversion?_ga=1.73932054.521213662.1436031708)

Semiconductor, C. (2012). *I2C Master/Multi-Master/Slave*. San José: Cypress Perform.

Shiye, X. (2015). *4-axis areocraft instruction manual*. ShenZhen.

*Sparkfun.* (s.f.). Obtenido de <https://www.sparkfun.com/products/10264>

*Sparkfun.* (s.f.). Obtenido de <https://www.sparkfun.com/products/8606>

*Sparkfun.* (2012). Obtenido de <https://learn.sparkfun.com/tutorials/accelerometer-basics>

*Sparkfun.* (2013). Obtenido de [https://www.sparkfun.com/pages/accel\\_gyro\\_guide](https://www.sparkfun.com/pages/accel_gyro_guide)

*Sparkun.* (2011). Obtenido de <https://learn.sparkfun.com/tutorials/i2c>

*Sparkun.* (2012). Obtenido de [https://learn.sparkfun.com/tutorials/gyroscope?\\_ga=1.39319975.521213662.1436031708](https://learn.sparkfun.com/tutorials/gyroscope?_ga=1.39319975.521213662.1436031708)

- SpectraSymbol. (2013). Obtenido de [https://www.ini.uzh.ch/~ppyk/BasicsOfInstrumentation/Flex\\_Sensor\\_Spectra\\_Symbol.pdf](https://www.ini.uzh.ch/~ppyk/BasicsOfInstrumentation/Flex_Sensor_Spectra_Symbol.pdf)
- Stephens, R. (2014). *Quadcopter Axes and Motion*. Obtenido de [http://flickrhivemind.net/blackmagic.cgi?id=16989545847&url=http%3A%2F%2Fflickrhivemind.net%2FUser%2Fstephens%2520planning%2FTimeline%3Fsearch\\_type%3DUser%26textinput%3Dstephens%2520planning%26photo\\_type%3D250%26method%3DGET%26noform%3Dt%26sort%3DDate%252](http://flickrhivemind.net/blackmagic.cgi?id=16989545847&url=http%3A%2F%2Fflickrhivemind.net%2FUser%2Fstephens%2520planning%2FTimeline%3Fsearch_type%3DUser%26textinput%3Dstephens%2520planning%26photo_type%3D250%26method%3DGET%26noform%3Dt%26sort%3DDate%252)
- STMicroelectronics. (Abril de 2010). *Tilt measurement using a low-g-3-axis accelerometer*. Obtenido de ST: [http://www.st.com/web/en/resource/technical/document/application\\_note/CD00268887.pdf](http://www.st.com/web/en/resource/technical/document/application_note/CD00268887.pdf)
- Symbol, S. (2014). *Flex Sensor FS*. Spectra Symbol.
- Taboadela, C. (2007). *Goniometría: una herramienta para la evaluación de las incapacidades laborales*. Buenos Aires: ASOCIART ART.
- Torres, C. (2014). *Fritzing*. Obtenido de <http://fritzing.org/projects/mpu-6050-board-gy-521-acelerometro-y-giroscoPIO>
- Virtual Reality*. (s.f.). Obtenido de <http://www.bienetec.es/producto/cyberglove-systems-cyberglove-ii>
- Virtual Realitys LLC*. (2016). Obtenido de <http://www.vrealities.com/products/data-gloves/5dt-data-glove-14-ultra-2>
- ZigBee Alliance*. (s.f.). Obtenido de <http://www.zigbee.org/zigbee-for-developers/zigbee3-0/>