



ESPE

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

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

**CARRERA DE INGENIERÍA EN ELECTRÓNICA,
AUTOMATIZACIÓN Y CONTROL**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO ELECTRÓNICO,
AUTOMATIZACIÓN Y CONTROL**

**TEMA: SISTEMA REPLICADOR DE MOVIMIENTO
ARTICULAR DE EXTREMIDAD SUPERIOR DERECHA EN
BRAZO ROBÓTICO INDUSTRIAL POR ESTUDIO
ELECTROMIOGRÁFICO Y USO DE KINECT**

**AUTORES: SANDOVAL SOCASI DANIELA ESTEFANIA
TRUJILLO MAYA ANDREA PAULINA**

DIRECTOR: ING. ERAZO ANDRÉS, M.Sc.

SANGOLQUÍ

2016



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

CERTIFICACIÓN

Certifico que el trabajo de titulación, ***“SISTEMA REPLICADOR DE MOVIMIENTO ARTICULAR DE EXTREMIDAD SUPERIOR DERECHA EN BRAZO ROBÓTICO INDUSTRIAL POR ESTUDIO ELECTROMIOGRÁFICO Y USO DE KINECT”***, realizado por las señoritas ***DANIELA ESTEFANÍA SANDOVAL SOCASI*** y ***ANDREA PAULINA TRUJILLO MAYA***, 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 Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar a las señoritas ***DANIELA ESTEFANÍA SANDOVAL SOCASI*** y ***ANDREA PAULINA TRUJILLO MAYA***, para que lo sustente públicamente.

Sangolquí, Diciembre del 2016

Ing. Andrés Erazo, M.Sc.

Director



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

AUTORÍA DE RESPONSABILIDAD

Nosotras, **DANIELA ESTEFANÍA SANDOVAL SOCASI**, con cédula de identidad N° 1721133302 y **ANDREA PAULINA TRUJILLO MAYA** con cédula de identidad N° 1718979683, declaramos que este trabajo de titulación “**SISTEMA REPLICADOR DE MOVIMIENTO ARTICULAR DE EXTREMIDAD SUPERIOR DERECHA EN BRAZO ROBÓTICO INDUSTRIAL POR ESTUDIO ELECTROMIOGRÁFICO Y USO DE KINECT**”, 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 declaramos que este trabajo es de nuestra autoría, en virtud de ello nos declaramos responsables del contenido, veracidad y alcance de la investigación mencionada.

Sangolquí, Diciembre del 2016

Daniela Estefanía Sandoval Socasi

C.C. 1721133302

Andrea Paulina Trujillo Maya

C.C. 1718979683



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

AUTORIZACIÓN

Nosotras, **DANIELA ESTEFANÍA SANDOVAL SOCASI** y **ANDREA PAULINA TRUJILLO MAYA**, autorizamos a la Universidad de las Fuerzas Armadas ESPE la publicación en la biblioteca virtual de la institución del presente trabajo de titulación **“SISTEMA REPLICADOR DE MOVIMIENTO ARTICULAR DE EXTREMIDAD SUPERIOR DERECHA EN BRAZO ROBÓTICO INDUSTRIAL POR ESTUDIO ELECTROMIOGRÁFICO Y USO DE KINECT”** cuyo contenido, ideas y criterios son de nuestra autoría y responsabilidad.

Sangolquí, Diciembre del 2016

Daniela Estefanía Sandoval Socasi

C.C. 1721133302

Andrea Paulina Trujillo Maya

C.C. 1718979683

DEDICATORIA

A las personas que más amo, mis padres quienes han brindado su amor, apoyo y motivación a lo largo de mi formación académica.

A mi madre Bertha por estar conmigo en todo momento dándome ánimos y fuerza en los momentos difíciles y enseñándome a luchar para conseguir mis metas.

A mi padre Javier por su esfuerzo y por sus valiosos consejos que me han impulsado para culminar mi carrera profesional.

A mi abuelita Bertha que desde el cielo me bendice, guía mi camino y sé que estaría muy orgullosa de mi logro.

El presente proyecto va dedicado a ustedes que han depositado su confianza en mí y con su esfuerzo y sacrificio me han sacado adelante para hoy ser una profesional.

Daniela Estefanía Sandoval Socasi

DEDICATORIA

Para las dos personas que forman una parte fundamental e irremplazable en mi vida, mis padres. Nunca olvidare sus palabras “Todo esfuerzo tiene su recompensa” y efectivamente en cada una de las etapas que he pasado junto a ustedes así ha sido, siempre demostrándoles de lo que puedo ser capaz gracias a sus enseñanzas y consejos.

A mi madre Paulina por apoyarme y ser mi mejor amiga, mi confidente, por alentarme a no rendirme durante uno de los caminos más complicados en los que te pone la vida.

A mi padre Giovanni por siempre apoyarme y enseñarme que pase lo que pase uno jamás debe rendirse, por hacerme entender que no hay que quedarse callada ante las injusticias de la vida.

Gracias a ustedes soy la mujer que soy ahora, por todo el amor, cariño, apoyo y sacrificio que han hecho para que hoy llegue a ser una profesional este proyecto va dedicado a ustedes.

Los amo y siempre estaré agradecida por darme la mejor educación tanto a nivel profesional como a nivel personal.

Andrea Paulina Trujillo Maya

AGRADECIMIENTO

A Dios, a la Virgen del Quinche por bendecirme y guiarme para cumplir este propósito tan anhelado.

Infinitamente agradecida con mis Padres por depositar su confianza en mí y por guiarme para ser una persona de bien. A mi madre por acompañarme en las noches de desvelo, por ser mi amiga y confidente, a mi padre por su esfuerzo y entrega en el hogar para que nunca nos falte nada.

A mi hermano Christian por estar siempre pendiente de mí y por brindarme su ayuda en lo posible.

A toda mi familia, abuelitos, tíos, primos por sus palabras de motivación para salir adelante. Especialmente a mis tíos Elisa y Berthi por sus consejos y apoyo en todo momento.

A mi Santy que a pesar de la distancia siempre está preocupado de mi bienestar, dándome ánimos para luchar y conseguir mis objetivos. Gracias por todo amor.

A mis amigos, por compartir estos años de agradables momentos y experiencias maravillosas, gracias por compartirme sus alegrías y tristezas, por escucharme y darme consejos.

Por último, agradezco al Ing. Andrés Erazo y al Ing. Andrés Arcentales por los conocimientos y tiempo brindado, durante la elaboración de este proyecto.

Daniela Estefanía Sandoval Socasi

AGRADECIMIENTO

Gracias a todas las personas que formaron parte de esta etapa, por brindarme, su amor, su apoyo y su amistad.

A mis padres siempre les estaré agradecida por todo lo que me han enseñado y seguiré aprendiendo de ustedes durante toda mi vida, no habría culminado mi carrera sin su apoyo, su paciencia y sacrificio.

A mis hermanos Valeria y César porque a pesar de ser mis pequeños me han ayudado en este camino, dándome su amor, sus locuras, incluso sus consejos, todo aquello que me incentiva y me da fuerzas para ser mejor persona por ustedes y para ustedes, son mi vida y los amo mucho.

A mi compañero de vida Richard, gracias por ser parte de mi vida, por tu comprensión, apoyo y por todas las experiencias buenas y malas que vivimos en esta etapa, ya somos colegas mi amor. Te amo. Gracias por todo.

A mis amigos, gracias por haber compartido este largo y duro camino, por las malas noches y tantas experiencias vividas juntos, porque de una u otra forma hicieron que esto tenga su parte divertida. Siempre estarán en mi corazón.

Finalmente, gracias al Ing. Andrés Erazo y al Ing. Andrés Arcentales por compartirnos sus conocimientos y apoyarnos durante todo el proyecto.

ÍNDICE DE CONTENIDOS

DEDICATORIA	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
AGRADECIMIENTO	vii
ÍNDICE DE CONTENIDOS	viii
ÍNDICE DE TABLAS	xiii
ÍNDICE DE FIGURAS	xiv
RESUMEN	xviii
ABSTRACT	xix
CAPÍTULO I	1
INTRODUCCIÓN	1
1.1 Antecedentes.....	1
1.2 Justificación e importancia	1
1.3 Alcance del proyecto	2
1.4 Objetivos.....	3
1.4.1 Objetivo General.....	3
1.4.2 Objetivos Específicos	3
CAPÍTULO II	5
ESTADO DEL ARTE	5
2.1 El brazo humano	5

2.1.1 Fisiología	5
2.1.2 Principales músculos	6
2.2 El brazo robótico industrial “CRS Robotics modelo A255.1”	10
2.2.2 Descripción del brazo robótico	10
2.2.3 Espacio de trabajo	11
2.2.4 Relación del espacio de trabajo Humano-Máquina	12
2.2.5 Activación de motores del brazo robótico	13
2.3 Análisis de movimientos de activación muscular	14
2.4 Electromiografía	17
2.3.1 Electroodos	17
2.3.2 Colocación de electrodos	17
2.5 Medición de posición (MPU6050)	18
2.5.2 Acelerómetro	19
2.5.3 Giroscopio	19
2.6 Representación de las señales EMG (Módulo de LEDs)	20
2.6.2 Funcionamiento	20
2.7 Tarjeta STM32F4	21
2.8 Kinect	23
2.8.2 Especificaciones técnicas	23
2.8.3 Rastreo esquelético o Skeleton Tracking	24

2.9	PC-ROS	26
CAPÍTULO III.....		29
DISEÑO E IMPLEMENTACIÓN DEL SISTEMA REPLICADOR DE MOVIMIENTOS		29
3.1	Diseño e implementación del sistema de adquisición EMG	30
3.1.1	Diseño y análisis de la etapa de protección	30
3.1.2	Diseño y análisis de la etapa de preamplificación	31
3.1.3	Diseño y análisis de la etapa de antisaturación.....	33
3.1.4	Diseño y análisis de la etapa de amplificación final	34
3.1.5	Diseño y análisis del filtro pasa banda.....	35
3.1.6	Adquisición de las señales con la tarjeta STM32-F4.....	36
3.1.7	Representación de los movimientos de activación muscular.....	37
3.2	El sistema Kinect	38
3.2.1	Prerrequisitos de instalación	38
3.2.2	OpenNi 1.5.4.....	39
3.2.3	Módulo del sensor Kinect.....	39
3.2.4	Openni_Tracker	41
3.2.5	Pruebas de instalación.....	42
3.2.6	Algoritmo de detección de las articulaciones del usuario.....	44
3.2.7	Ángulos de posicionamiento del brazo robótico a través de los datos obtenidos del Kinect.....	47

3.2.8 Implementación del algoritmo de detección de las articulaciones del usuario	50
3.3 Adquisición de datos MPU6050.....	51
3.3.1 Comunicación I2C.....	51
3.3.2 Configuración de registros.....	52
3.3.3 Cálculo de ángulo de orientación	54
3.4 Algoritmo para el movimiento del brazo robótico	54
CAPÍTULO IV	57
PRUEBAS Y RESULTADOS	57
4.1 Pruebas del sistema de adquisición de señales electromiográficas	57
4.1.1 Análisis en tiempo	58
4.1.2 Análisis en frecuencia.....	59
4.2 Pruebas del protocolo de activación muscular.....	61
4.3 Pruebas del Kinect	63
4.3.1 Reconocimiento del usuario	63
4.3.2 Ángulo del hombro 45°	63
4.3.3 Ángulo del codo 60°	65
4.3.4 Ángulo del codo -45°	65
4.4 Pruebas del procesamiento de información	67
4.5 Pruebas y resultados globales en ambiente.....	67
CAPÍTULO V.....	72
CONCLUSIONES Y RECOMENDACIONES.....	72

REFERENCIAS BIBLIOGRÁFICAS..... 74

ÍNDICE DE TABLAS

Tabla 1 Músculos del Hombro	7
Tabla 2 Músculos del Brazo	8
Tabla 3 Músculos el antebrazo	8
Tabla 4 Ángulos de movimiento de las articulaciones	13
Tabla 5 Especificaciones Técnicas STM32F4	21
Tabla 6 Variables correspondientes para la traslación del origen	48
Tabla 7 Direcciones de la aceleración en los ejes X, Y y Z	53
Tabla 8 Configuración serial	6
Tabla 9 Posición del hombro de 45° a 0°	68
Tabla 10 Posición del codo de 0° a 60°	68
Tabla 11 Posición del hombro de 0° a -45°	89

ÍNDICE DE FIGURAS

Figura 1 Partes de las extremidades superiores	5
Figura 2 Articulaciones manipulador robótico CRS A255	11
Figura 3 a) Espacio de trabajo (Vista superior).....	12
Figura 4 Driver L298N.....	14
Figura 5 Circuito de Activación de los frenos.....	14
Figura 6 Flexión y Extensión el antebrazo	15
Figura 7 Abducción y aducción del brazo	16
Figura 8 Rango de trabajo de la aducción del hombro	16
Figura 9 Colocación de electrodos a) Biceps, b) Triceps, c) Deltoides.....	18
Figura 10 Sensor MPU6050	19
Figura 11 Tira de LEDs	20
Figura 12 STM32F4-DISCOVERY	22
Figura 13 Sensor Kinect	23
Figura 14 Puntos de referencia del cuerpo humano	25
Figura 15 Posición de calibración (Pose PSI)	25
Figura 16 Sistema de referencia Kinect.....	26
Figura 17 Solicitud/Respuesta de mensajes en ROS.....	28
Figura 18 Diagrama de bloques "Sistema replicador de movimientos"	30

Figura 19 Circuito de retroalimentación.....	31
Figura 20 Circuito de protección	31
Figura 21 Circuito de preamplificación (INA128).....	33
Figura 22 Circuito antisaturación	33
Figura 23 Circuito de amplificación.....	34
Figura 24 Respuesta del filtro: a) Magnitud, b) Fase	36
Figura 25 Diagrama de adquisición señal EMG.....	37
Figura 26 Placa del sistema de adquisición de señales EMG.....	37
Figura 27 Circuito tira de LEDs	38
Figura 28 Prueba correcto funcionamiento Kinect.....	41
Figura 29 Correcta detección del Kinect	42
Figura 30 Detección /perdida de Usuario	43
Figura 31 Calibración y detección pose PSI Usuario	44
Figura 32 Diagrama de flujo Kinect.....	45
Figura 33 Script detección usuario/selección articulación	46
Figura 34 Robot antropomórfico / Ejes	47
Figura 35 Traslación del origen – torso al hombro	48
Figura 36 Detección de Usuario, ángulos codo y hombro,	50
monitorización del usuario	51

Figura 37 Ubicación de los MPU6050 en el brazo robótico	52
Figura 38 Diagrama de bloques activación hombro.....	55
Figura 39 Diagrama de bloques activación codo	56
Figura 40 Activación de los músculos: a) Bíceps-Flexión	57
del Antebrazo, b) Tríceps-Extensión del Antebrazo,	57
c) Deltoides-Aducción del Hombro.....	58
Figura 41 Representación gráfica del análisis RMS: a) Bíceps,	58
b) Tríceps, c) Deltoides	59
Figura 42 a) Espectro de frecuencia: a) Bíceps, b) Tríceps, c) Deltoides	61
Figura 43 Rangos de activación muscular.....	62
Figura 44 Representación de las activaciones de los músculos	62
Figura 45 Reconocimiento del usuario (Pose PSI).....	63
Figura 46 Ángulo del hombro 45°	64
Figura 47 Ángulo del codo 60°	65
Figura 48 Ángulo del codo - 45°	66
Figura 49 Diagrama de bloques comunicación del sistema	66
replicador de movimientos	67
Figura 50 Posición hombro del brazo humano y brazo robótico a 45°	68
Figura 51 Posición del codo del brazo humano y brazo robótico a 60°	69

Figura 52 Posición del codo del brazo humano y brazo robótico a -45° 70

RESUMEN

En el presente proyecto de investigación se desarrolla un sistema capaz de detectar el movimiento de la extremidad superior derecha de un ser humano, con el fin de reproducirlo sobre el brazo robótico industrial CRS A255.1. Para ello, se utiliza el dispositivo Kinect el cual permitirá conocer la posición en la que se encuentra el brazo humano a través de la lectura de las coordenadas del hombro y la muñeca del mismo; datos que servirán para que a través de la cinemática inversa se pueda obtener los ángulos del codo y del hombro para posteriormente mover el brazo robótico. Para garantizar la precisión del movimiento se utilizaron dos sensores (IMUS-MPU6050), los cuales leen y retroalimentan el ángulo medido continuamente con el fin de que cada articulación llegue a la posición deseada. Además, se cuenta con un sistema de adquisición de señales electromiográficas (EMG), el mismo que detectará la activación de tres músculos diferentes. Dos de ellos conforman el brazo (bíceps y tríceps) y uno el hombro (deltoides). Cuando el músculo correspondiente a cada uno de los movimientos este activado y reciba la posición del brazo humano, el brazo robótico adoptará la posición deseada. Cada uno de los músculos son representados por una tira de LEDs en las cuales, dependiendo de la fuerza aplicada al movimiento se podrá visualizar diferentes intensidades de iluminación.

Palabras Claves:

- **KINECT**
- **ELECTROMIOGRAFÍA**
- **BRAZO ROBOTICO CRS A255.1**
- (Enciclopedia Salud , s.f.)**MÚSCULOS DEL BRAZO**

ABSTRACT

This research Project develops a system able of detecting the movement of the upper right extremity of a human being, in order to reproduce it on the industrial robotic arm CRS A255.1. The Kinect device is used to know the position in which the human arm is located through reading the coordinates of the shoulder and the wrist of it; data that will serve so that through the inverse kinematics the angles of the elbow and the shoulder of the robotic arm can be calculated. To ensure precision to the movement, two sensors are used (IMUS-MPU6050), which read and send the sensed angle continuously so that each joint reaches the desired position. In addition, there is an electromyographic (EMG) signal acquisition system, which will detect the activation of 3 different muscles; where 2 of them are part of the arm (biceps and triceps) and one of them of the shoulder (deltoids). The Robotic arm is moved to the desired angle when the corresponding muscle of each movement is activated. The force applied on each of the muscles are represented by a strip of LEDs, in which depending on the force applied to the movement, it will be able to visualize different lighting intensities.

Keywords:

- **KINECT**
- **ELECTROMYOGRAPHY**
- **CRS A255.1 ROBOTIC ARM**
- **IMUS-MPU6050**
- **ARM MUSCLES**

CAPÍTULO I

INTRODUCCIÓN

1.1 Antecedentes

Con el paso de los años, el desarrollo tecnológico ha dado paso a la creación de nuevas herramientas que facilitan el trabajo del ser humano, sea en actividades de manera cotidiana o actividades industriales que tienen un grado de complejidad mayor. Este es el caso de los dispositivos robóticos, los cuales permiten desarrollar nuevas aplicaciones biomédicas, espaciales, industriales y de investigación. (Collahuazo Sinchi)

El análisis de la dinámica del cuerpo humano es un tema de interés para un amplio rango de disciplinas que abarcan desde biomecánica a ciencias del movimiento humano. Su estudio incluye aspectos tecnológicos (sensores y sistemas de medición), aspectos de procesamiento de señal (filtrado, calibración, sincronización) y aspectos de modelado (coordinación de múltiples articulaciones, redundancia, organización del sistema de control humano). Además, el análisis se centra en funciones motoras específicas (mantenimiento de postura, marcha, movimiento de brazos, y manipulación) o en diversas áreas de aplicación (De Luca, 2006)

Debido a los antecedentes mencionados, se propone en el presente trabajo desarrollar un sistema capaz de detectar el movimiento de la extremidad superior derecha y replicarlo en un brazo robótico; con un algoritmo de control a ejecutarse en tiempo real realizando un procesamiento de señales adquiridas para posteriormente transmitir las al brazo robótico.

1.2 Justificación e importancia

En la industria, los diferentes procesos cada vez se vuelven más automatizados. En varios de los procesos como el corte de varios materiales, manejo de metales, puntos de suelda, manipulación de diferentes productos, entre otros; se necesita

realizar actividades con precisión y de cierta forma con una fuerza determinada. Sin embargo, debido a limitaciones que se dan por varios factores como la seguridad de los operadores, espacio, falla humana, entre otros, es necesario colocar brazos robóticos los cuales serán controlados por la persona encargada de operarlos. De esta manera, se toma en cuenta el valioso trabajo humano sin necesidad de reemplazarlo por una máquina, llegándose a obtener un trabajo conjunto del operador con el brazo robótico para mejorar los procesos industriales debido a la incorporación de características humanas de precisión, exactitud y control.

El prototipo a diseñarse podrá ser utilizado en procesos en donde se deba manipular materiales o productos finales, los cuales deban ser manejados con cuidado en ambientes abiertos con tráfico de personas u objetos aleatorios. Si estos procesos son programados en un brazo robótico común, se da la posibilidad de incurrir en choques o paradas de emergencia imprevistas, debido a que no se controlan en detalle variables como la trayectoria, por lo cual el brazo robótico se mueve de un lado a otro en un entorno cambiante.

Por todo lo mencionado anteriormente, surge la idea como proyecto de investigación para la obtención del título de grado, un enfoque para replicar los movimientos de una extremidad superior humana; controlando la trayectoria por la cual se ejercen los movimientos a través de la adquisición de las señales electromiográficas de los músculos más representativos del brazo humano, y reforzando el control de la posición y ángulo en la que se desea que se encuentre el brazo robótico industrial CRS A255.1 a cada instante a través del Kinect y sensores de posición.

1.3 Alcance del proyecto

El presente trabajo de investigación, tiene como objetivo la generación de un sistema capaz de detectar los movimientos de un brazo humano y replicarlo en un brazo robótico CRS A255.1. El trabajo está previsto realizarlo en cuatro etapas, las mismas que son descritas a continuación:

- En la primera etapa se realizará el diseño y la implementación del hardware de adquisición de señales electromiográficas (EMG). Se realizará un diseño basado en dispositivos de bajo costo, tomando en cuenta protecciones de seguridad para el sujeto. El circuito permite adquirir las señales de varios músculos en tiempo real.
- En la siguiente etapa de este trabajo de investigación se realizará el procesamiento digital de las señales adquiridas; en el cual se analizará la señal en tiempo, frecuencia y amplitud, con el fin de conocer las activaciones de los diferentes músculos al realizar el movimiento de contracción y relajación del brazo.
- Posteriormente se desarrollará un programa basado en el uso del Kinect, para detectar el sentido, dirección y ángulo del movimiento del brazo humano; con el fin de definir movimientos con diferentes inclinaciones, desviaciones y réplicas con el brazo robótico industrial.
- Finalmente se realizará la comunicación entre las señales EMG adquiridas del brazo humano, y las señales censadas por el dispositivo Kinect, para enviarlas hacia el brazo robótico donde se reproducirá el movimiento del usuario mediante el diseño de un algoritmo de control.

1.4 Objetivos

1.4.1 Objetivo General

Diseñar e implementar un sistema para replicar los movimientos de un brazo humano en un brazo robótico industrial CRS A255.1.

1.4.2 Objetivos Específicos

- Desarrollar un sistema de adquisición multicanal para señales EMG, considerando premisas tales como bajo costo y seguridad del dispositivo.

- Analizar las señales EMG adquiridas tanto en tiempo como en frecuencia
- Definir la relación existente entre la activación de un grupo de músculos y el movimiento de las articulaciones del brazo robótico industrial.
- Utilizar el dispositivo Kinect para el establecimiento de la posición y el ángulo de ubicación del brazo humano.

CAPÍTULO II

ESTADO DEL ARTE

2.1 El brazo humano

El brazo humano es un órgano externo y articulado que se usa para el movimiento de objetos y la locomoción del cuerpo humano. Este comprende desde la mano hasta el omoplato y está formado por músculos y huesos.

Los músculos de las extremidades superiores son de los más potentes del cuerpo humano, por lo que los brazos son muy útiles para actividades que requieren fuerza física; y más, aún gracias al sistema de balanceo corporal intervienen directamente en las funciones de equilibrio.

2.1.1 Fisiología

Los brazos están conectados a la parte superior del tronco y tienen la función de permitir la movilidad para coger, sujetar y manejar objetos al realizar diferentes actividades. Estos están compuestos por cuatro partes: Mano, antebrazo, brazo, cintura escapular (Ver Figura 1).



Figura 1 Partes de las extremidades superiores

Fuente: (Enciclopedia Salud , s.f.)

- **Mano:** Es el órgano efector de la extremidad superior que empieza desde la muñeca y está conformada por 5 dedos. Esta tiene dos funciones principales, la prensión y el tacto. Es importante mencionar que el sentido del tacto desarrolla totalmente la capacidad de la mano, debido a que sin éste sería imposible medir la fuerza prensora. La parte más importante de la mano es el dedo pulgar, ya que sin éste la capacidad de la mano se reduce hasta en un 40%; afectando principalmente a la capacidad de agarre y manipulación. (Bernal, 2012)
- **Antebrazo:** Este se conoce como la región de la extremidad superior comprendida entre el codo y la muñeca; y al igual que las otras partes del miembro superior está constituido por una serie de huesos, músculos, nervios y vasos que trabajan en conjunto. El antebrazo presenta dos huesos: el cubito y el radio. (Sanz, 2014)
- **Brazo:** El brazo está comprendido desde el codo hasta el hombro y se compone de un hueso que forma la parte superior del mismo, el húmero.
- **Cintura escapular:** Sirve como estructura de anclaje de las extremidades superiores al esqueleto axial; es decir, esta fija el hombro al tronco de manera que constituye la comunicación entre este y el brazo. Además, en ella se fijan muchos músculos que mueven los brazos, siendo una estructura muy ligera que permite a las extremidades superiores un grado de movimiento que no se tiene en ningún otro lugar del cuerpo (Corsino, 2001).

2.1.2 Principales músculos

Los músculos del brazo se dividen en una región anterior y posterior. En términos generales todo lo que está por anterior, es flexor y por posterior es extensor.

A continuación, se muestra los músculos de cada una de las partes que forman la extremidad superior con el fin de elegir los más importantes de acuerdo a los movimientos específicos que se realizan para este proyecto.

2.2.2.1 Músculos del hombro

En la Tabla 1 se puede observar los músculos que conforman el hombro, su ubicación y su función principal.

Tabla 1

Músculos del hombro

Músculos	Ubicación	Función
Deltoides	Cara superior del hombro	-Flexión y extensión el brazo -Abducción y aducción -Rotación interna y externa
Infraespinoso	Por detrás del músculo supraespinoso	-Rotación externa del hombro
Redondo Menor	Cara posterior del hombro	-Estiramiento hacia atrás -Rotación externa -Aducción
Redondo Mayor	Región posterior del hombro	-Rotación interna -Aducción
Supraespinoso	En la zona superior de la articulación del hombro	-Protege el músculo deltoides del movimiento de la separación del brazo (Abducción)
Subescapular	Encima del redondo mayor	-Estiramiento hacia atrás -Rotación interna

2.2.2.2 Músculos del brazo

En la Tabla 2 se puede observar los músculos que conforman el brazo, su ubicación y su función principal.

Tabla 2**Músculos del brazo**

Músculos	Ubicación	Función
Bíceps	Región anterior del brazo	-Flexor y supinador del antebrazo -Elevador y abductor del brazo
Braquial Anterior	En la región anterior e inferior del brazo	-Flexor del antebrazo sobre el brazo
Coracobraquial	Forma la pared lateral de la axila y se encuentra en la región anterior del brazo	-Flexión y aducción de hombro.
Tríceps	En la región posterior del brazo	-Extensor del antebrazo. -Aductor del brazo

2.2.2.3 Músculos del antebrazo

En la Tabla 3 se puede observar los músculos que conforman el antebrazo, su ubicación y su función principal.

Tabla 3**Músculos el antebrazo**

Músculos	Ubicación	Función
Ancóneo	En la región posterior del antebrazo	-Extensor y pronador del antebrazo
Abductor largo del pulgar	En la capa profunda del antebrazo	-Abductor del pulgar y de la mano -Extensor de la mano
		CONTINÚA 

Cubital Anterior	En la parte interna y anterior del antebrazo.	-Flexión y aductor de la muñeca
Cubital Posterior	Región posterior del antebrazo	-Extensor y aductor de la mano
Extensor propio del menique	En la región palmar interna	-Extensor del dedo menique
Extensor de los dedos	En la capa superficial de la región posterior del antebrazo	-Extensor de las falanges sobre los metacarpianos y estos sobre el antebrazo.
Extensor largo del pulgar	En la capa profunda de la región posterior del antebrazo	-Extensor del pulgar
Extensor corto del pulgar	En la capa profunda de la región posterior del antebrazo	-Extensor y abductor del pulgar
Extensor del índice	En la capa profunda de la región posterior del antebrazo	-Extensor del dedo índice
Extensor radial largo del carpo	En la región externa del antebrazo, debajo del braquioradial	-Extensión y abducción de la muñeca
Flexor propio del pulgar	En la región anterior del antebrazo.	-Flexor de las falanges del pulgar
Flexor profundo de los dedos	En la región anterior del antebrazo.	-Flexor de la tercera falange
Flexor superficial de los dedos	En el segundo plano de la región anterior del antebrazo	-Flexor de los dedos sobre la mano

CONTINÚA



Flexor radial del carpo (Palmar Mayor)	En la región anterior del antebrazo	-Flexor de la mano y del antebrazo
Flexor largo del pulgar	En la región anterior del antebrazo en tercer plano	Flexor de la falange distal -Ayuda a la función de pinza del pulgar con el índice
Palmar largo (Palmar Menor)	En la región anterior del antebrazo	-Flexor de la mano
Pronador redondo	En la parte externa de la región anterior del antebrazo	-Pronador y extensor del antebrazo
Pronador cuadrado	Ubicado en la región de la muñeca	-Pronación del antebrazo
Supinador Largo (Braquioradial)	En la región posteroexterna del antebrazo	-Flexión y supinación del antebrazo
Supinador Corto	Dispuesto alrededor y sobre la cara lateral del codo y la parte superior del antebrazo	-Supinación del antebrazo

2.2 El brazo robótico industrial “CRS Robotics modelo A255.1”

El manipulador robótico CRS A255 de cinco grados de libertad es ideal para varias aplicaciones las cuales incluye asistencia a máquinas, manipulación de material liviano, así como también recogida y colocación de objetos.

2.2.2 Descripción del brazo robótico

El manipulador robótico CRS A255 es un sistema mecánico basado en la estructura de un brazo humano. Está formado por cinco grados de libertad: la base, el hombro, el codo, la muñeca y la rotación del elemento terminal. Cada articulación posee diferente grado de desplazamiento angular. (A255 Robot Arm User Guide, 2000) (Ver Figura 2)

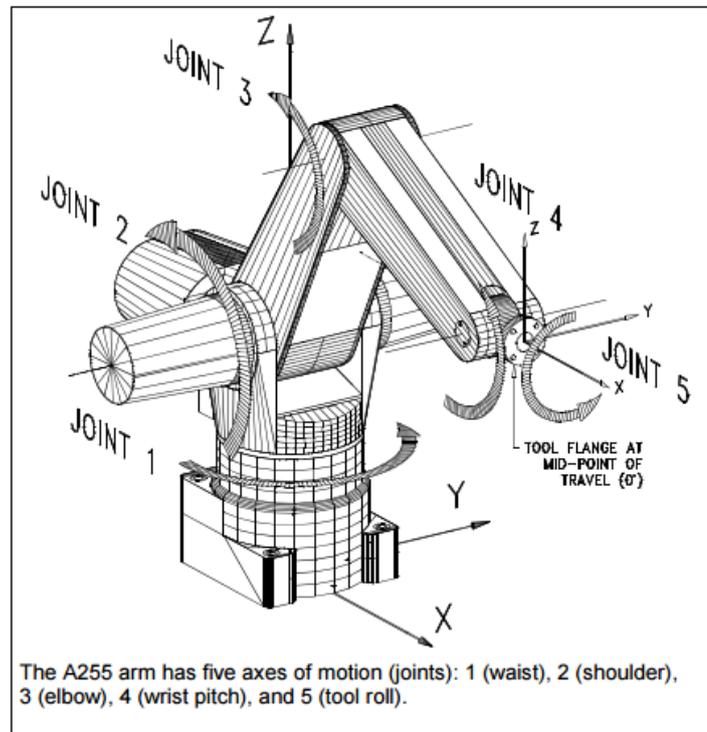


Figura 2 Articulaciones manipulador robótico CRS A255

Fuente: (Robotics Corporation CRS, 2000)

El movimiento se efectúa por medio de servomotores, los mismos que trabajan en conjunto con cajas reductoras, rodamientos, tornillos sin fin, acopladores armónicos y codificadores ópticos; para formar la estructura mecánica del sistema.

2.2.3 Espacio de trabajo

El espacio de trabajo del brazo robótico está en función del desplazamiento radial y vertical. En forma radial, el brazo puede desplazarse hasta 350 grados; y en forma vertical, el brazo puede tener un desplazamiento de hasta 120 grados. (A255 Robot Arm User Guide, 2000) (Ver Figura 3).

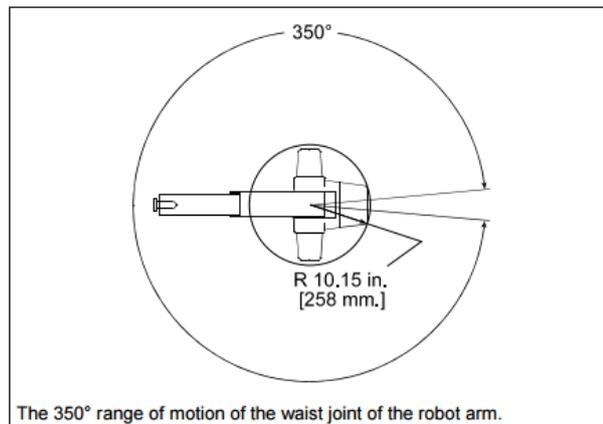
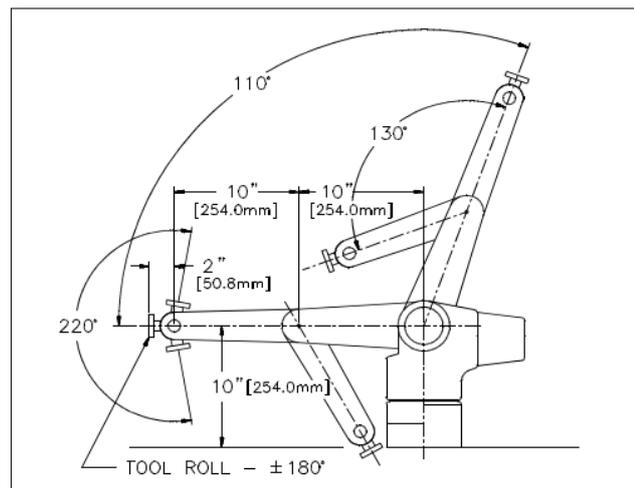


Figura 3 a) Espacio de trabajo (Vista superior)

Fuente: (Robotics Corporation CRS, 2000)



b) Espacio de trabajo (Vista lateral)

Fuente: (Robotics Corporation CRS, 2000)

2.2.4 Relación del espacio de trabajo Humano-Máquina

La Tabla 4 muestra una comparación entre los ángulos que puede mover las diferentes articulaciones tanto del brazo humano como el robótico.

Tabla 4**Ángulos de movimiento de las articulaciones**

Articulación	Ángulo máximo brazo robótico	Ángulo máximo brazo humano
Hombro	110°	180°
	130°	140°
Cintura	350°	+45°/-45°

2.2.5 Activación de motores del brazo robótico

El Brazo Robótico CRS A255 cuenta con motores y frenos que se activan con 12 Voltios DC en cada una de sus articulaciones; por lo que es necesario realizar una etapa de potencia para activar los frenos, motores y para poder controlar el giro de los mismos. Para ello, se utiliza la etapa de potencia desarrollada sobre el brazo robótico en el proyecto de investigación “DISEÑO E IMPLEMENTACIÓN DE TÉCNICAS DE CONTROL PARA EL CONTROLADOR CAD DEL MANIPULADOR ROBÓTICO CRS A255.1”, desarrollado por el Ing. Patricio Rivera. Cabe mencionar que dicho proyecto incorporaba un controlador fuzzy, el cual no fue utilizado debido a que el mismo ocupaba gran cantidad de recursos de la tarjeta STM32F4, lo cual reducía el trabajo en tiempo real de nuestra aplicación; por lo que fue necesario desarrollar un algoritmo de control para el movimiento de los motores.

De la etapa de potencia, solo se utiliza dos drivers para controlar el giro de 2 motores y el circuito correspondiente para la activación de los frenos de los mismos. El driver utilizado es el L298N; en la Figura 4 se puede observar la distribución del mismo.

Para la activación de los frenos se utiliza el circuito mostrado en la Figura 5, con el fin de separar la activación de los frenos de las señales de control y evitar que en la tarjeta utilizada para el control ingresen corrientes parásitas. (Rivera, 2015)

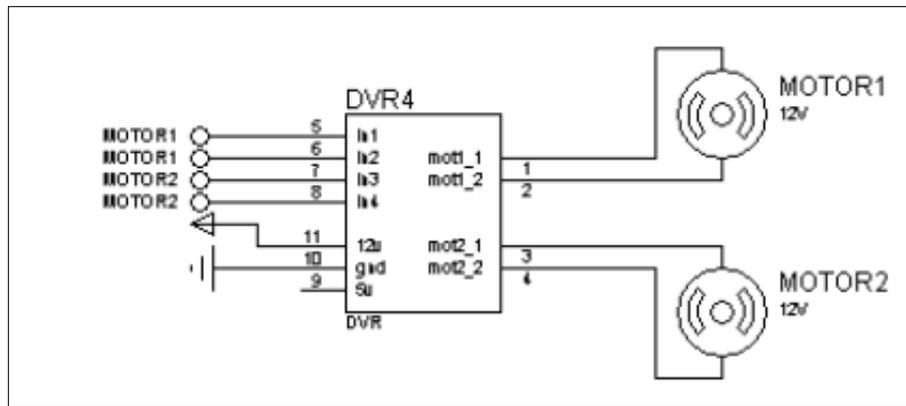


Figura 4 Driver L298N

Fuente: (Rivera, 2015)

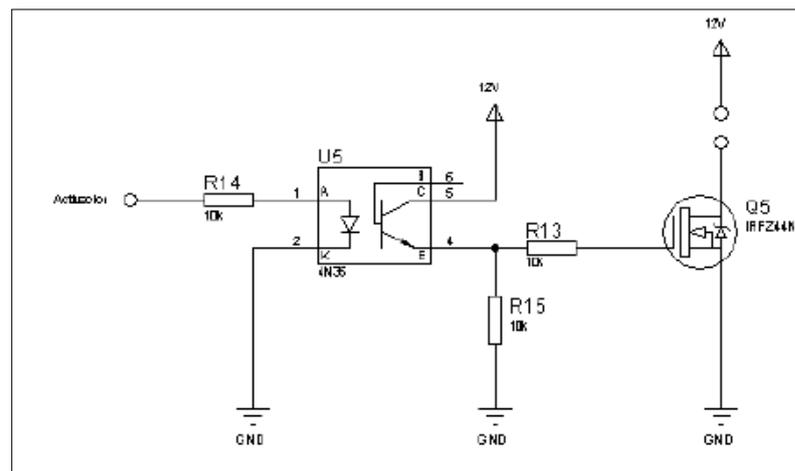


Figura 5 Circuito de Activación de los frenos

Fuente: (Rivera, 2015)

2.3 Análisis de movimientos de activación muscular

Los movimientos de las extremidades superiores dependen de la contracción y la relajación coordinada de los músculos opuestos. Una vez analizado las funciones de los músculos y los movimientos que se pueden realizar con el manipulador robótico, tomando en cuenta las limitaciones de hardware, se decide realizar 3 movimientos específicos:

- **Flexión y extensión del antebrazo**

Este ejercicio activa los músculos bíceps y tríceps, dependiendo de si se ejerce el movimiento de flexión o extensión. Cuando se realiza la flexión del brazo se activa el músculo bíceps y se relaja el tríceps, mientras que si se extiende la articulación se relaja el bíceps y se activa el tríceps. (Ver Figura 6)



Figura 6 Flexión y Extensión el antebrazo

Fuente: (*Ciencias Biológicas y Educación para la Salud, s.f.*)

- **Aducción el hombro**

Este ejercicio consiste en desplazar el brazo desde arriba hacia abajo. Con este movimiento, el músculo deltoides y el supraespinoso se activan, reforzándose y generando robustez gracias a la cintura escapular; logrando así ir desde un ángulo de 0° a 180°. (Ver Figura 7)



Figura 7 Abducción y aducción del brazo

Fuente: (Martín, 1998)

Es importante mencionar que en este caso se trabajará de 45° a 0° , tomando como 0° cuando el brazo este colocado horizontalmente. (Ver Figura 8)

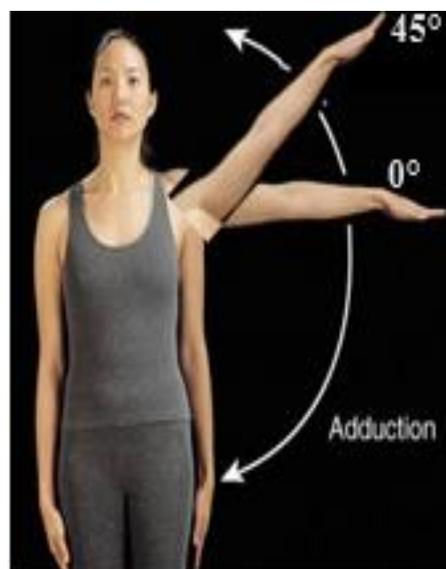


Figura 8 Rango de trabajo de la aducción del hombro

Una vez detallado los movimientos que se realizarán, se puede observar que se trabajará únicamente con tres músculos de la extremidad superior derecha; el bíceps y tríceps correspondientes al brazo, y el deltoides correspondiente al hombro.

2.4 Electromiografía

La electromiografía es el registro de la actividad eléctrica del sistema nervioso periférico y los músculos, el mismo que sirve para realizar un diagnóstico de posibles afecciones al sistema nervioso periférico. Se lo realiza mediante la colocación de electrodos en los músculos y se registran la actividad eléctrica de los mismos (Electromiograma, s.f.).

2.3.1 Electroodos

En el campo de la bioelectricidad, se utiliza los electrodos como un transductor, debido a que mediante este dispositivo se puede transformar una magnitud bioeléctrica en otra magnitud puramente eléctrica. Los electrodos hacen un cambio iónico del tejido vivo del cuerpo humano hacia un dispositivo electrónico (Pardell), el cual se encarga de procesar las señales y obtener características útiles para su estudio. Entre las señales biológicas con las que trabajan estos dispositivos se encuentran las señales: Electrocardiográficas (ECG), Electroencefalografías (EEG) y Electromiográficas (EMG).

El registro de estas señales se realiza especialmente con dos tipos de electrodos, los electrodos de superficie y los electrodos invasivos. Los electrodos de superficie son colocados en la superficie de la piel y son capaces de tomar registros de la actividad bioeléctrica; mientras que los electrodos invasivos son incrustados en el tejido para registrar directamente la diferencia de potencial existente entre la membrana celular y la piel (Neuman, 2000). Cabe recalcar que para el presente estudio se utilizó electrodos de superficie.

2.3.2 Colocación de electrodos

La buena ubicación de los electrodos es importante para obtener una señal correcta. Los electrodos bipolares EMG de superficie deben ser colocados en cada músculo que se desea examinar, a una distancia de 2 a 3 centímetros. (Ver Figura 9).

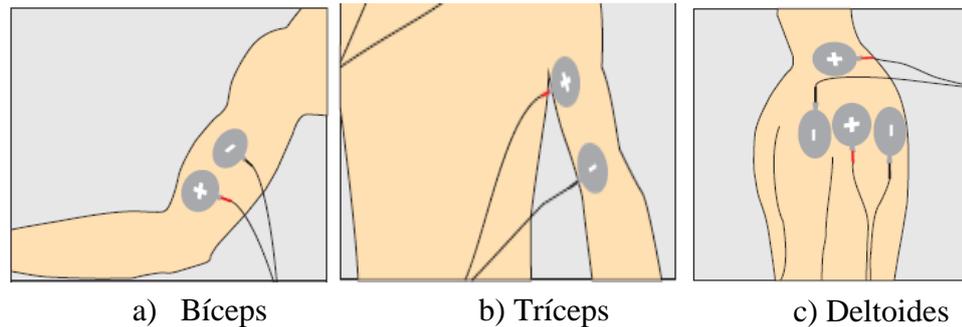


Figura 9 Colocación de electrodos a) Bíceps, b) Tríceps, c) Deltoides

Fuente: (Massó & Rey, 2010)

Adicionalmente se debe colocar un electrodo de referencia lejos del área de registro y en un tejido eléctricamente neutro (Massó & Rey, 2010).

2.5 Medición de posición (MPU6050)

EL MPU6050 es una unidad de medición inercial (IMU- Inertial Measurement Units) de 6 grados de libertad (DoF), la cual dispone de un acelerómetro de 3 ejes, un giroscopio de 3 ejes y un procesador de movimiento digital. Este sensor es útil en aplicaciones de navegación, goniometría, estabilización, entre otros. (Tutorial MPU6050, Acelerómetro y Giroscopio, 2015).

El sensor utiliza el protocolo de comunicación I2C para la transmisión de datos, el mismo que maneja dos líneas a las que se les conoce como bus I2C. En dicho protocolo por una línea se envía una señal de reloj (SCL) para la sincronización del sistema y por la otra línea se enviarán o recibirán los datos (SDA). (Carletti, 2007) (Ver Figura 10).

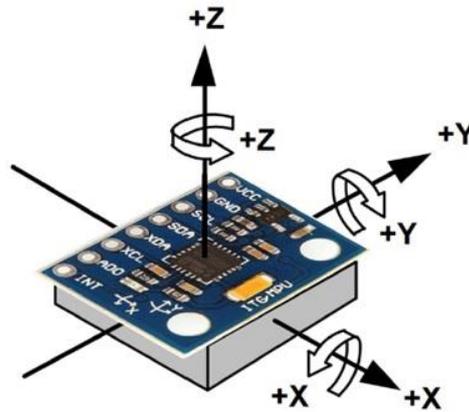


Figura 10 Sensor MPU6050

Fuente: (Tutorial MPU6050, Acelerómetro y Giroscopio, 2015)

2.5.2 Acelerómetro

Los acelerómetros son sensores inerciales capaces de medir la aceleración de una masa de prueba respecto a la gravedad de la tierra. Debido a la gravedad terrestre (9.8m/s^2), se puede calcular el ángulo respecto al eje X o al eje Y; (Tutorial MPU6050, Acelerómetro y Giroscopio, 2015) donde conociendo las medidas de los tres ejes del acelerómetro se puede conocer los ángulos de inclinación aplicando las siguientes formulas:

$$\text{Angulo } Y = \tan^{-1} \left(\frac{x}{\sqrt{y^2 + z^2}} \right) \quad (1)$$

$$\text{Angulo } X = \tan^{-1} \left(\frac{y}{\sqrt{x^2 + z^2}} \right) \quad (2)$$

2.5.3 Giroscopio

Los giroscopios son sistemas capaces de medir la velocidad angular más no la orientación en sí. La velocidad angular es la tasa de cambio del desplazamiento

angular por unidad de tiempo, es decir que tan rápido gira un cuerpo alrededor de su eje.

Si se conoce el ángulo inicial de la IMU, se le puede sumar el valor que marca el giroscopio para saber su nuevo ángulo (Tutorial de Arduino y MPU 6050, 2014).

$$\text{Angulo } Y = \text{Angulo } Y \text{ anterior} + \text{Giroscopio } Y \cdot \Delta t \quad (3)$$

2.6 Representación de las señales EMG (Módulo de LEDs)

Las tiras de LEDs son dispositivos que tienen diversas configuraciones dependiendo del uso que se requiera darles: iluminación intensa o tenue de diferentes ambientes, variación de colores en un amplio espectro usando un controlador RGB, entre otros. (Ver Figura 11)

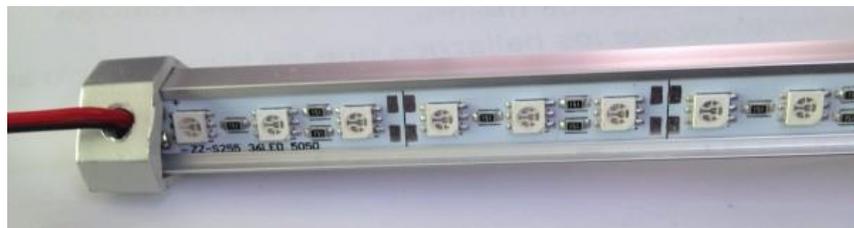


Figura 11 Tira de LEDs

Fuente (www.prometec.net)

2.6.2 Funcionamiento

Las tiras están formadas por LEDs con encapsulado SMD de dos tamaños:

- SMD 3528 (3,5mm x 2,8mm): Cada LED emite entre 3 y 5 Lúmenes (Tiras LEDs, 2012)
- SMD 5050 (5,0mm x 5,0mm): Cada LED emite entre 10 y 15 Lúmenes (Tiras LEDs, 2012)

Los LEDs SMD funcionan con corriente continua; es decir, necesitan una fuente de alimentación de 12VDC para su funcionamiento a corriente nominal.

2.7 Tarjeta STM32F4

La tarjeta STM32F4–Discovery es una herramienta que cuenta con un desarrollo versátil. La misma permite evaluar las características del controlador digital de señales (DSC) de alto desempeño ARM Cortex–M4F. El diseño de la tarjeta permite aprovechar las principales características del microcontrolador, así como la reprogramación de manera fácil y rápida mediante la unidad de programación-depuración ST–LINK integrada en la tarjeta (Sistema de desarrollo STM32F4–Discovery, s.f.).

Además, esta cuenta con la ventaja de que instalando las librerías de Waijung y U3CM en Matlab/Simulink, se puede cargar el programa elaborado en este último directamente.

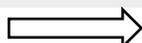
La Tabla 5 presenta las características técnicas de la Tarjeta STM32F4–Discovery

Tabla 5

Especificaciones Técnicas STM32F4

N°	DESCRIPCIÓN
1	Alimentación por puerto USB 5VDC.
1	Suministro externo de 5VDC y 3.3VDC.
1	Microcontrolador STM32F407, ARM Cortex–M4F
1	Memoria de datos 192kB + 4kB (SRAM, RTCRAM).
1	Oscilador interno PLL hasta 168MHz (210 DMIPS).
14	Temporizadores / Contadores con PWM para control de motores, encoder incremental.
1	Temporizador / Contador de 24 bit + WatchDog.

CONTINÚA



6	UART, 2 CAN, 3 SPI (Master/Slave), 3 I2C, 2 I2S.
16	Convertidores ADC de 12 bit (2 MS/s).
2	Convertidores DAC de 12 bit.
1	Puerto USB OTG HS/FS, 1 Puerto Ethernet.
8	LEDs de diagnóstico o estado (4 programables).
2	Pulsadores
1	Acelerómetro digital de tres ejes LIS302DL.
1	Amplificador digital de audio CS43L22.
1	Conector Jack para audio.
1	Conector Mini-USB para programación/depuración.
1	Conector Micro-USB OTG “Host” o “Device”

En la Figura 12 se puede observar la tarjeta STM32F4 y sus componentes mencionados anteriormente.

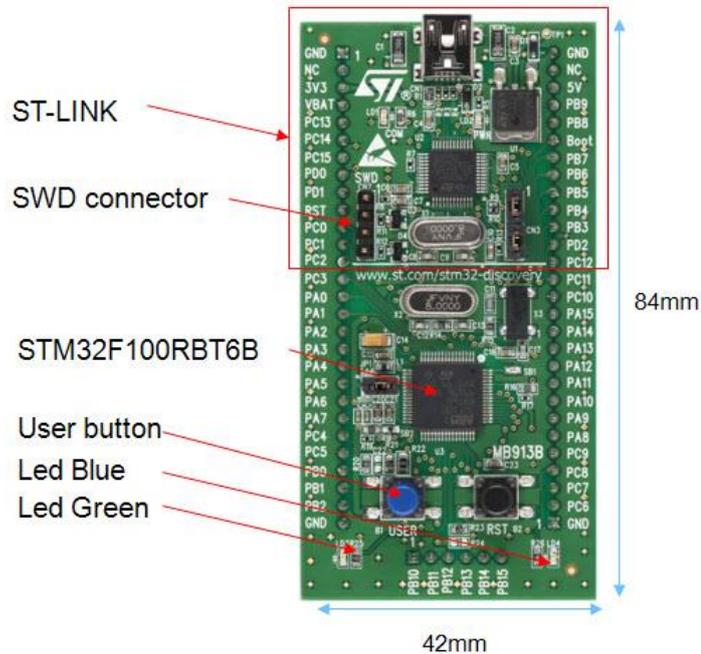


Figura 12 STM32F4-DISCOVERY

Fuente: (STM32F4-DISCOVERY, s.f.)

2.8 Kinect

Kinect es un controlador de juego libre creado por Alex Kipman desarrollado y auspiciado por Microsoft para la video consola Xbox 360. El dispositivo permite a los usuarios interactuar y manejar la videoconsola con movimientos de su cuerpo, comandos de voz, objetos e imágenes; para lo cual utiliza cámaras, sensores de profundidad y un micrófono multi-array. (Ñacato, 2014)

2.8.2 Especificaciones técnicas

El dispositivo cuenta con una cámara RGB, un sensor de profundidad, un micrófono de múltiples matrices y un procesador personalizado. Dichos componentes trabajan en conjunto y permiten realizar la captura de movimiento de todo el cuerpo en 3D incluido el reconocimiento del esqueleto, facial y voz. (Ñacato, 2014) (Ver Figura 13)

El dispositivo posee además un campo de visión angular de 57° horizontalmente y 43° verticalmente, además de contar con una base motorizada que le permite inclinar el sensor 27° verticalmente en ambas direcciones. La distancia correcta para la identificación esta entre 1.2 a 3.5 metros. (Ballesteros, 2012)



Figura 13 Sensor Kinect

Fuente: (Kinect XBOX-360, s.f.)

2.8.2.1 Cámara RGB

La cámara RGB posee una resolución de 8-bits VGA (640x480 pixeles), fabricada con un sensor CMOS y capaz de emitir 30fps (imágenes por segundo). Permite la captura y transmisión datos de video desde el entorno al computador. (Ballesteros, 2012)

2.8.2.2 Sensor de profundidad

El sensor de profundidad está formado por: un emisor de rayos infrarrojo y un sensor CMOS monocromático.

El emisor de rayos infrarrojo permite capturar datos de video en 3D bajo cualquier condición lumínica; este opera con una resolución VGA (640x480) con 16 bits de profundidad a 30fps. El sensor CMOS monocromático se encarga de calcular la distancia en función del tiempo que tarda en reflejar la luz en el entorno de captura. (Ñacato, 2014)

2.8.2.3 Micrófono multi-array

El micrófono multi-array consta de cuatro micrófonos ubicados en el extremo inferior del Kinect. Cada uno trabaja de manera independiente, procesa 16-bits y un rango de muestreo de 16kHz. El sensor reconoce la ubicación del sonido y elimina el ruido del exterior para utilizarlo en aplicaciones de reconocimiento de voz.

2.8.3 Rastreo esquelético o Skeleton Tracking

El sensor Kinect permite realizar el seguimiento del cuerpo, mediante la detección del esqueleto del usuario en tiempo real; logrando adquirir datos que incluyen la ubicación de las articulaciones y las posiciones en las que se encuentra el esqueleto del usuario (Ver Figura 14).

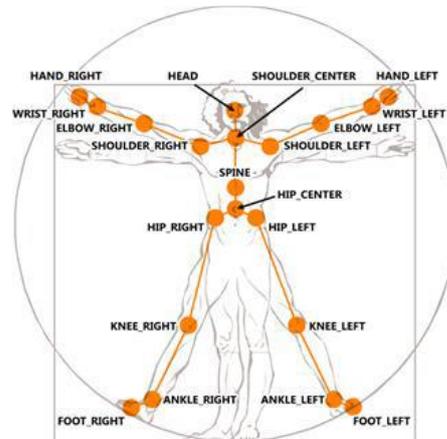


Figura 14 Puntos de referencia del cuerpo humano

Fuente: (Kinect for Windows SDK)

Una vez que el Kinect pueda reconocer un usuario, para que empiece la calibración del mismo, es necesario que el usuario adopte una posición específica para posteriormente continuar con la lectura de las coordenadas de la articulación especificada (Ver Figura 15).

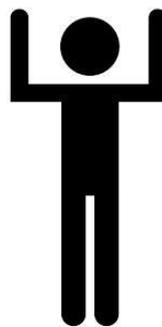


Figura 15 Posición de calibración (Pose PSI)

Fuente: (www.yanamal.com)

Estos datos son enviados en coordenadas cartesianas en 3 dimensiones; es decir, tendrá los puntos de cada articulación en x, y, z. Estas variables serán tomadas con respecto al sensor 3D del Kinect, el mismo que se alinea con el torso de la persona; por lo que en el usuario este se vuelve el punto de referencia. Debido a esto, los

puntos de las diferentes articulaciones serán tomados respecto al torso, tomando en cuenta que los ejes del Kinect tienen la distribución mostrada en la Figura 16

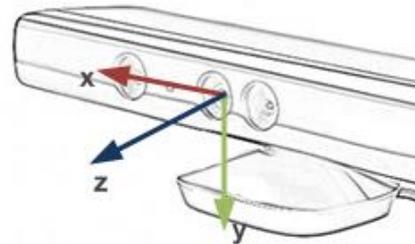


Figura 16 Sistema de referencia Kinect

Fuente: (Integración de dispositivos Leap y Kinect, 2013)

2.9 PC-ROS

ROS (Robotic Operating System) provee librerías y herramientas para ayudar a los desarrolladores de software a crear aplicaciones para robots. ROS provee abstracción de hardware, controladores de dispositivos, librerías, herramientas de visualización, comunicación por mensajes, administración de paquetes y más. ROS fue diseñado bajo la licencia open source, BSD. (Velasco, s.f.).

ROS cuenta con computación gráfica, la cual es una red de procesos que hace que los datos en conjunto se procesen peer to peer. Se maneja varios conceptos que define a ROS:

- **Nodos:** Son algoritmos independientes y ejecutables que utiliza ROS para poder realizar tareas específicas. La principal característica es su diseño modular, donde se ejecutan los nodos individualmente; gestionados por su nodo principal ROS Master. Estos nodos se comunican a través de mensajes y tópicos para cumplir con la tarea de cada uno de ellos y llegar al objetivo común.

Cada nodo está escrito usando la librería cliente de ROS. Mediante el uso de esta librería no solo se puede crear nodos sino también los publicadores y suscriptores de los mensajes y tópicos en los nodos, además de los servicios y sus parámetros. Se puede utilizar esta librería tanto en C++ como en Python. (Ramarras, 2014)

- **Mensajes:** Son tipos de estructuras de datos que se usan para la comunicación entre nodos, estos tipos de datos pueden ser:

- Tipos primitivos estándar: (enteros, punto flotante, booleanos, entre otros).
- Estructuras y arreglos

- **Topics:** Los nodos pueden publicar los mensajes a través de un topic, es decir estos se enrutan a través de un sistema como publicación/suscripción. El topic es un nombre que se utiliza para identificar el contenido del mensaje. Un nodo envía un mensaje mediante su publicación a un topic determinado, donde si un nodo está interesado en un determinado tipo de dato, se suscribirá al topic correspondiente. (Ramarras, 2014)

Un topic se puede considerar como un bus de mensajes. Es importante mencionar que cualquier persona se puede conectar al bus para recibir o enviar mensajes, siempre y cuando se configuren para recibir el dato correcto. (Aaron, 2014)

- **ROS Master:** Es un nodo maestro y su función principal es permitir que los nodos se localicen unos a otros. Además, proporciona el registro de nombres y consulta para el resto de la computación gráfica. Sin este, los nodos no podrían encontrar los mensajes de otros, e invocar a los servicios.

Para que funcione este nodo y se pueda ejecutar toda la lógica realizada en los demás nodos, se debe ejecutar el comando roscore.

- **Servicios:** Estos se realizan a través del concepto de solicitud/respuesta, es decir se manda un mensaje para pedir y otro para responder; generándose una arquitectura Cliente/Servidor. En general, un nodo servidor ofrece un servicio con un nombre y un nodo cliente solicita el servicio con el mismo nombre mediante el envío del mensaje de solicitud, esperando la respuesta del mismo. (Aaron, 2014)

En la Figura 17 se puede observar de forma gráfica el proceso de comunicación entre nodos con una publicación y suscripción.

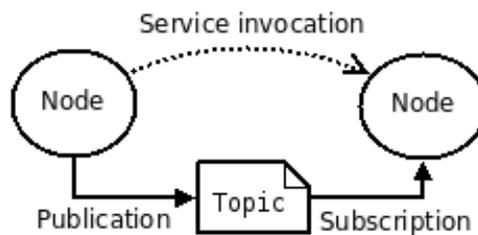


Figura 17 Solicitud/Respuesta de mensajes en ROS

Fuente: (Aaron, 2014)

ROS funciona sobre plataformas Windows o Linux. Se trabajó sobre Ubuntu Trusty 14.04, para la cual se utiliza los pasos propuestos en el sitio web de ROS.org, para su correcta configuración. Es importante mencionar que ROS cuenta con varias versiones y en el presente proyecto se utilizó ROS indigo, la cual permite utilizar paquetes y librerías para el funcionamiento del Kinect; además de permitir utilizar y ejecutar códigos en Python o en C++, los cuales son utilizados por el Kinect.

CAPÍTULO III

DISEÑO E IMPLEMENTACIÓN DEL SISTEMA REPLICADOR DE MOVIMIENTOS

El sistema replicador de movimientos está compuesto por tres etapas de instrumentación, una de actuadores y una de procesamiento.

La primera etapa es el sistema de adquisición de señales electromiográficas (EMG), en la cual mediante el uso de electrodos superficiales se detecta la activación de los músculos (bíceps, tríceps y deltoides); los mismos que son adquiridos y analizados con la tarjeta STM32F4. Mediante esta medición, se indica la activación de los movimientos a realizarse en el robot y son representados mediante la variación de intensidad en tiras de LEDs.

La segunda etapa de instrumentación la componen una PC con sistema operativo Linux (Ubuntu Trusty 14.04) y el Kinect; en la cual mediante el sistema ROS y programas desarrollados en Python y C++ se adquieren las coordenadas X, Y, Z del hombro y muñeca para su posterior uso en el desarrollo de la cinemática inversa, para adquirir los ángulos de movimiento de las articulaciones (hombro y codo).

La tercera etapa comprende el uso de los sensores MPU6050 los cuales miden el ángulo en el cual se encuentran las articulaciones (hombro, codo) del brazo robótico CRS A-255.1. Estos se comunican a través de la tarjeta STM32F4 mediante comunicación I2C para entregar la información y procesarla.

El procesamiento de toda la información se la realiza la tarjeta STM32F4 Discovery, en la misma que llega la información del Kinect, los datos del MPU6050 y las señales EMG; con los cuales se realiza la activación de los motores del robot con el comportamiento deseado. (Ver Figura 18)

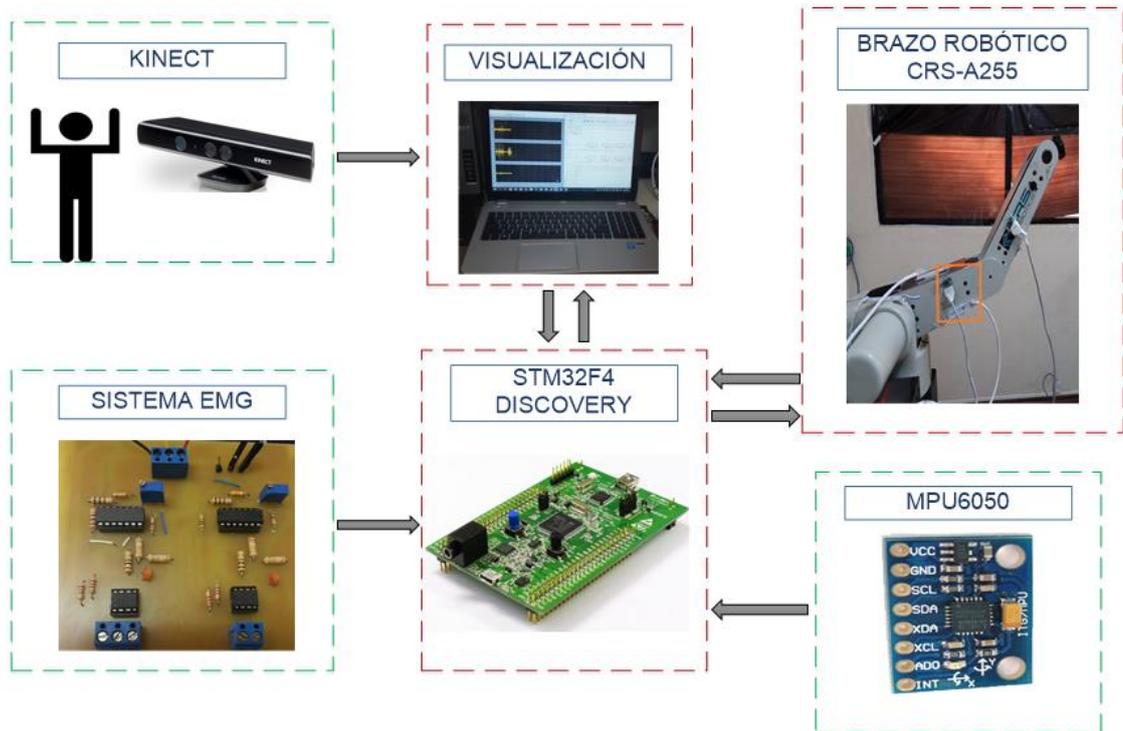


Figura 18 Diagrama de bloques "Sistema replicador de movimientos"

3.1 Diseño e implementación del sistema de adquisición EMG

Para el diseño del circuito se tomó en cuenta las características de la señal electromiográfica:

- Espectro de importancia (0 - 500Hz).
- Energía dominante (20 – 150Hz).
- Amplitudes (0 – 1mVrms aproximadamente).

Para el registro de las señales EMG se realizó el circuito compuesto por cinco etapas: Protección, preamplificación, antisaturación, amplificación final.

3.1.1 Diseño y análisis de la etapa de protección

Se realizó el circuito de retroalimentación o driver de pierna derecha propuesto por el fabricante Texas Instruments, utilizado para registrar señales de

electrocardiografía o ECG (ver Figura 19). Se seleccionó este circuito debido a que el mismo evita las corrientes de desbalance y compensa problemas de ruido en modo común en la entrada diferencial del amplificador de instrumentación.

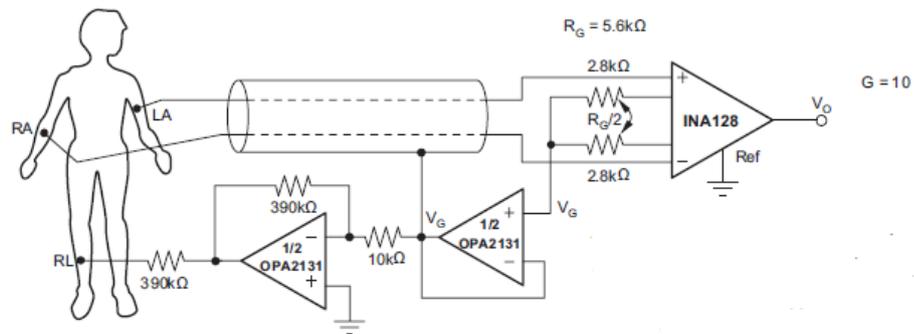


Figura 19 Circuito de retroalimentación

Fuente: (Texas Instrument)

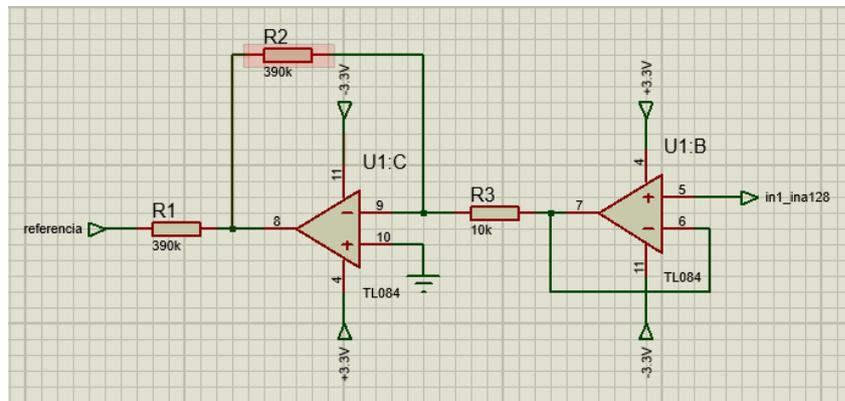


Figura 20 Circuito de protección

3.1.2 Diseño y análisis de la etapa de preamplificación

En la etapa de preamplificación se empleó un amplificador de instrumentación debido a que el mismo cumple con características importantes para el buen registro de las señales biológicas; como son:

- Impedancia de entrada de $10^9 - 10^{12} \Omega \parallel 2 - 10 \text{pF}$.
- $\text{CMRR} \geq 90 \text{dB}$
- Corriente de polarización de entrada $\leq 20 \text{nA}$.

Considerando estas características, se seleccionó el amplificador de instrumentación INA128 de Texas Instruments, el mismo que presenta el siguiente comportamiento:

- Corriente de polarización de entrada: 5 nA máx
- CMRR : 120 dB mín.
- Fuente de alimentación: $\pm 2.25 \text{ V}$ a $\pm 18 \text{ V}$
- Impedancia de entrada diferencial: $10^{10} \Omega \parallel 2 \text{pF}$.

El cálculo de resistencias para la etapa de preamplificación se determina mediante la siguiente ecuación:

$$G = 1 + \frac{50 \text{k}\Omega}{R_G} \quad (4)$$

A partir de esta relación, y tomando en cuenta que la ganancia de preamplificación es de 10, tenemos:

$$10 = 1 + \frac{50 \text{k}\Omega}{R_G} \quad (5)$$

$$R_G = \frac{50 \text{k}\Omega}{10 - 1} \quad (6)$$

$$R_G = (R_4 + R_5) = \mathbf{5,56 \text{k}\Omega} \quad (7)$$

La Figura 21 muestra el circuito implementado.

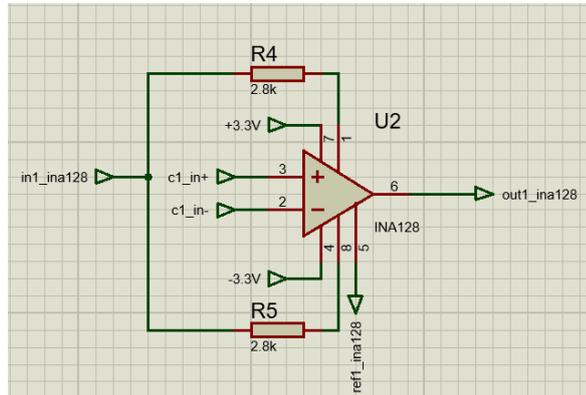


Figura 21 Circuito de preamplificación (INA128)

3.1.3 Diseño y análisis de la etapa de antisaturación

Se empleó un circuito integrador (ver Figura 22) propuesto por Texas Instrument para disminuir el voltaje de offset de salida del amplificador de instrumentación, evitando además de esta manera que el circuito entre en saturación.

$$f_{-3dB} = \frac{1}{2\pi R_6 C_1} = 1,59Hz \quad (8)$$

$$R_6 C_1 = 0,1 \approx (1M\Omega)(0.1\mu F) \quad (9)$$

La Figura 22 muestra el circuito implementado.

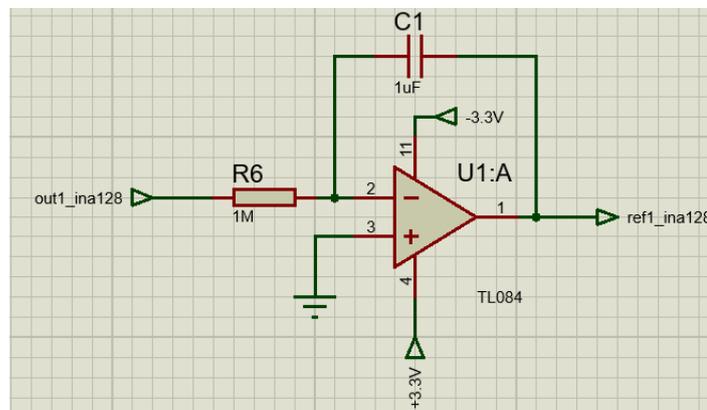


Figura 22 Circuito antisaturación

3.1.4 Diseño y análisis de la etapa de amplificación final

En esta etapa se realizó un amplificador no inversor. A continuación, se indican los cálculos.

Los valores seleccionados para las resistencias son:

$$R_9 = 56k\Omega \text{ y } R_7 = 1k\Omega$$

La siguiente relación permite obtener la ganancia del amplificador:

$$G = \frac{R_9}{R_7} + 1 \quad (10)$$

$$G = \frac{56k\Omega}{1k\Omega} + 1 \quad (11)$$

$$G = 57 \quad (12)$$

La Figura 23 muestra el circuito implementado.

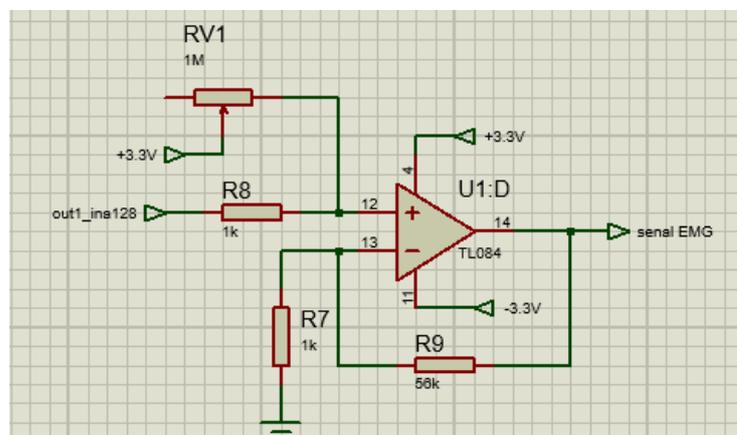


Figura 23 Circuito de amplificación

3.1.5 Diseño y análisis del filtro pasa banda

Se utilizó filtros digitales debido a las ventajas que presenta frente a un filtro analógico; estas son:

- El rendimiento de los filtros digitales es estable (no cambian por factores ambientales), lo cual evita la necesidad de calibrar periódicamente.
- La respuesta de frecuencia de un filtro puede ser ajustada usando un procesador programable.
- Las señales originales y filtradas pueden almacenarse para el estudio o análisis posterior.
- Los filtros digitales son recomendables para trabajar con bajas frecuencias y además son encontrados en muchas aplicaciones biomédicas.

Para realizar el diseño del filtro se utilizó el software Matlab/Simulink, el mismo que con la herramienta “Digital Filter Design” (Ver Figura 24), nos permite especificar el tipo, el orden, la frecuencia de muestreo y la frecuencia de corte del filtro que se desea implementar.

En el sistema de adquisición EMG se implementó un filtro con las siguientes características:

- Tipo de filtro: Filtro pasabanda
- Método de filtrado: Filtro IIR, configuración Butterwork
- Orden del filtro: 10
- Frecuencia de corte: De 20 a 150 Hz
- Frecuencia de muestreo: 1kHz

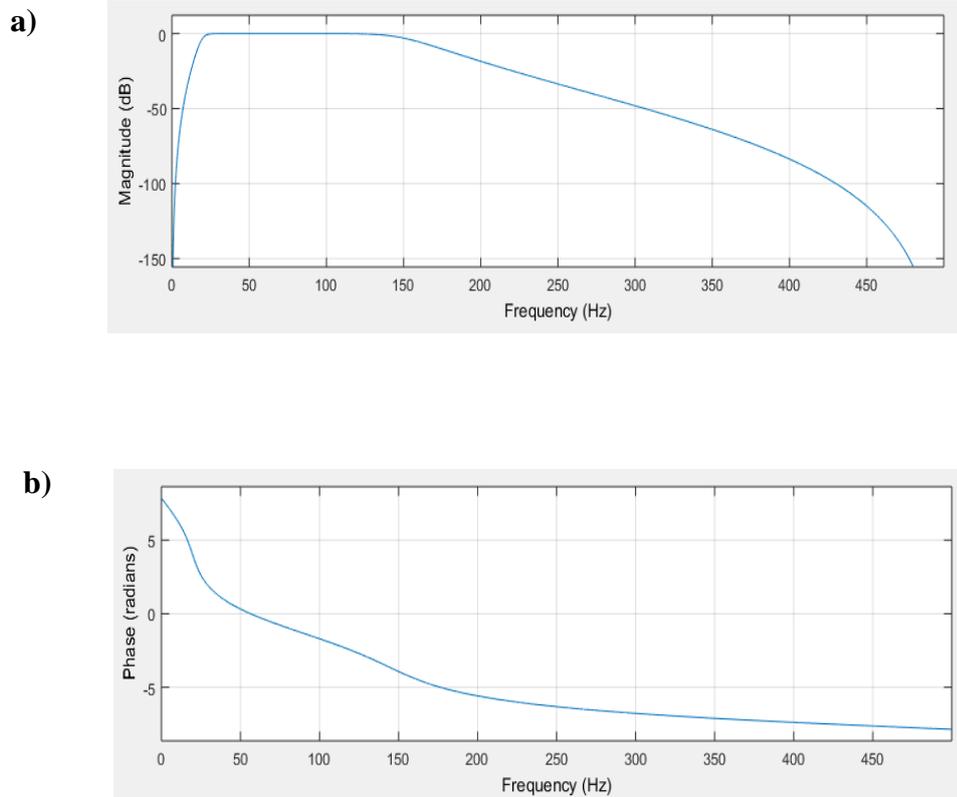


Figura 24 Respuesta del filtro: a) Magnitud, b) Fase

3.1.6 Adquisición de las señales con la tarjeta STM32-F4

Para la adquisición y procesamiento de las señales EMG se utilizó la tarjeta STM32F4. Para esto, primero se configuró el bloque Regular Analog to Digital Converter (ADC), el mismo que muestra los datos puros de la señal EMG con una resolución de 0 a 4095 bits. Para cambiar estos datos a un valor de tensión se utilizó el bloque de ganancia con un valor de $3.3/4095$. Posteriormente estos valores se almacenan en un buffer y el siguiente bloque detrend permite eliminar el offset. Esta señal pasa por un filtro pasabanda para elegir las frecuencias de interés, para finalmente entrar a un buffer donde se concatena el vector y se muestra la señal EMG filtrada (Ver Figura 25).

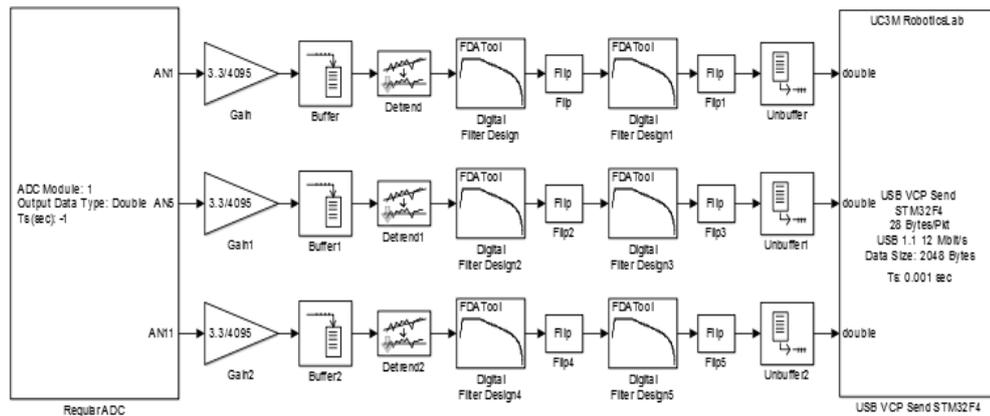


Figura 25 Diagrama de adquisición señal EMG

De acuerdo a las especificaciones y cálculos realizados en cada etapa se desarrolló el primer prototipo de adquisición de señales EMG (Ver Figura 26).

3.1.7 Representación de los movimientos de activación muscular

Las fuerzas de los movimientos de activación muscular van a ser representados en la variación de intensidad de luz de las tiras de LEDs.

Debido a que la intensidad de luz en un led es proporcional a la intensidad de corriente que circula en el mismo, es necesario utilizar un transistor para controlar este flujo. En esta aplicación se utilizó un transistor de potencia Darlington (TIP120), de tipo NPN, el mismo que soporta hasta 5 amperios y una tensión de 60V.

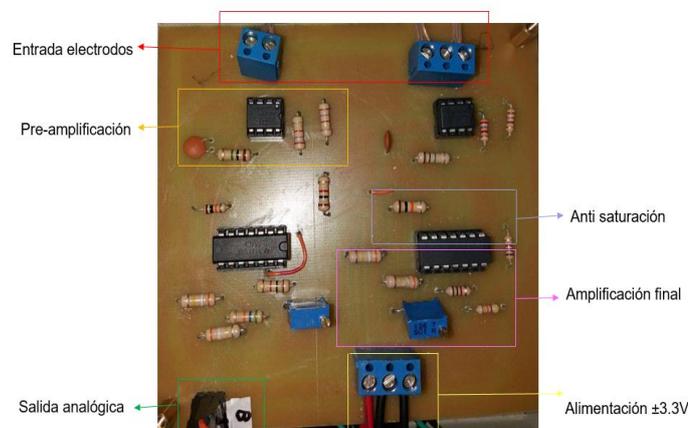


Figura 26 Placa del sistema de adquisición de señales EMG

Mediante este transistor se controló la carga que se transmite a la tira de LEDs. Esta señal es enviada desde un pin PWM de la tarjeta STM32-F4 a la base del transistor.

La Figura 27 muestra el circuito de control para la visualización de los LEDs.

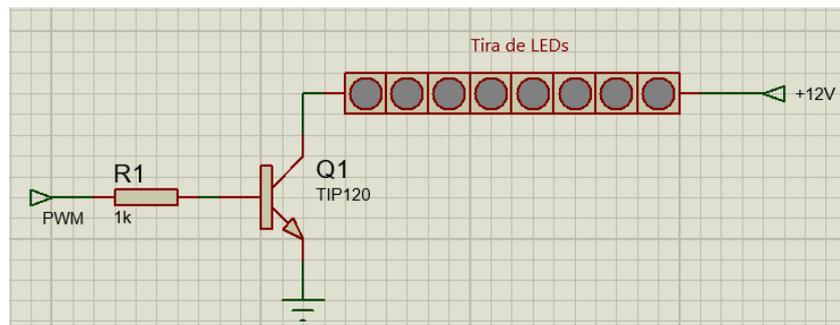


Figura 27 Circuito tira de LEDs

3.2 El sistema Kinect

Este sistema se utiliza para la detección de las coordenadas de las articulaciones del hombro y de la muñeca de la extremidad superior derecha del usuario, las cuales serán utilizadas en la cinemática inversa para obtener los ángulos a los cuales se moverá la parte del codo y el hombro. Todo este sistema se trabajó sobre ROS en Linux (Ubuntu Trusty 14.04) debido a la fácil manipulación y rendimiento de todos los paquetes y librerías necesarias para el correcto funcionamiento del Kinect.

3.2.1 Prerrequisitos de instalación

Es necesaria su instalación por el terminal de Ubuntu. Primero se ingresa el comando indicado en (1) para instalar los drivers para el reconocimiento de dispositivos USB, ya que el Kinect cuenta con este tipo de conector. Además, se debe instalar “doxygen graphviz”, el cual permite leer documentación de C, C++, Python y Java mediante el comando indicado en (2).

- (1) “sudo apt-get install git build-essential python libusb-1.0-0-dev freeglut3-dev openjdk-7-jdk()” (Marner, 2011)
- (2) “sudo apt-get install doxygen graphviz mono-complete” (Marner, 2011)

3.2.2 OpenNi 1.5.4

OpenNI (*Open Natural Interaction*) es un framework open source que sirve para trabajar y desarrollar aplicaciones que utilicen interacción con los usuarios. Esta interacción está basada en los sentidos humanos como pueden ser el oído o la visión. OpenNI facilita la comunicación con los sensores de audio, video y profundidad de los dispositivos hardware compatibles como Kinect o ASUS Xtion (ambos basados en el chip PS1080 de PrimeSense) (OpenNI Programmer Guide, 2012)

Para instalarlo se debe colocar en el terminal de Ubuntu los siguientes comandos:

1. “git clone https://github.com/OpenNI/OpenNI.git”
2. “cd OpenNI”
3. “git checkout Unstable-1.5.4.0”
4. “cd Platform/Linux/CreateRedist”
5. “chmod +x RedistMaker”
6. “./RedistMaker”
7. “cd OpenNI/Platform/Linux/Redist/OpenNI-Bin-Dev-Linux-x64-v1.5.4.0”

Tras lo cual se generan las librerías respectivas para su utilización en cualquier aplicación dentro de este entorno de trabajo.

3.2.3 Módulo del sensor Kinect

Es necesario instalar todos los paquetes para reconocer el sensor Kinect. Estos pueden ser descargados de la página de “<https://github.com>” o clonados de la misma

a través del terminal, siguiendo los pasos que se muestran a continuación para su instalación:

1. `“git clone https://github.com/avin2/SensorKinect”`
2. `“cd SensorKinect”`
3. `“cd Platform/Linux/CreateRedis”`
4. `“chmod +x RedisMaker”`
5. `“./RedisMaker”`
6. `“cd SensorKinect/Platform/Linux/Redis/Sensor-Bin-Linux-x64-v5.1.2.1”`
7. `“chmod +x install.sh”`
8. `“sudo ./install.sh”`

Además, es necesario ejecutar los siguientes comandos para que la cámara y el sensor del Kinect se instalen correctamente y más adelante no exista problemas en el momento de su ejecución.

1. `“sudo apt-get install ros-indigo-openni-launch”`
2. `“sudo apt-get install ros-indigo-openni-camera”`

Para comprobar que todo se instaló correctamente se debe conectar el Kinect al computador y ejecutar el comando que se muestra a continuación:

1. `“cd OpenNI/Platform/Linux-x86/Bin/Release”`
2. `“./Sample-NiSimpleViewer”`

Al correr estos dos comandos el Kinect es capaz de identificar objetos y personas como se muestra en la Figura 28; donde la imagen debe salir en combinación negro con amarillo.



Figura 28 Prueba correcto funcionamiento Kinect

3.2.4 Openni_Tracker

Se debe instalar `openni_tacker`, debido a que este cuenta con un programa desarrollado en C++, el cual permite leer la presencia de un nuevo usuario, así como la pose PSI y la calibración necesaria para proceder con la lectura de la ubicación de las diferentes articulaciones. Además, dentro de este programa se puede seleccionar la articulación de la cual se desea obtener sus coordenadas. Para su instalación es necesario seguir los pasos mostrados a continuación:

1. Descargar NITE 1.5.2.23 o una versión mayor. Es importante mencionar que sobre una versión menor existen varios problemas de instalación o ejecución del programa. NITE es un complemento que necesita `openni_tracker` para poder ejecutarse y realizar la lectura de las coordenadas de las articulaciones.
2. Una vez descargado NITE, se debe descomprimirlo, entrar a la carpeta y correr el script `install.sh` para instalar el paquete. Es importante instalar el programa de raíz.
3. Crear un espacio de trabajo dentro de ROS y clonar `openni_tracker` del repositorio en la carpeta `src/`, colocando el comando “`git clone https://github.com/ros-drivers/openni_tracker.git.`”

4. En la raíz del espacio de trabajo, ejecutar el comando “catkin_make” para compilar el paquete. (Skeleton Tracking, 2015)

3.2.5 Pruebas de instalación

Una vez instalados todos los paquetes mencionados en los puntos anteriores, se procede a realizar las respectivas pruebas para corroborar que todo se instaló correctamente.

Para comprobar que el OpenNi 1.5.4 fue correctamente instalado, se debe conectar el Kinect al computador y ejecutar el siguiente comando:

1. “roslaunch openni_launch openni.launch”

Al ser ejecutado se debe obtener lo que se muestra en la Figura 29. Es importante mencionar que en el Kinect se prenderá un foco de color verde de forma continua, lo que indica que se detectaron todos sus componentes de forma correcta.

```

/opt/ros/indigo/share/openni_launch/launch/openni.launch http://localhost:11311
process[camera_disparity_registered_hw-20]: started with pid [16195]
process[camera_base_link-21]: started with pid [16197]
process[camera_base_link1-22]: started with pid [16208]
process[camera_base_link2-23]: started with pid [16227]
process[camera_base_link3-24]: started with pid [16232]
[ INFO] [1464148272.166890690]: Initializing nodelat with 4 worker threads.
Warning: USB events thread - failed to set priority. This might cause loss of data...
Warning: USB events thread - failed to set priority. This might cause loss of data...
[ INFO] [1464148275.121447230]: Number devices connected: 1
[ INFO] [1464148275.121575338]: 1. device on bus 001:36 is a SensorKinect (Zae)
from PrimeSense (45e) with serial id '0000000000000000'
[ INFO] [1464148275.122366019]: Searching for device with index = 1
[ INFO] [1464148276.142968233]: Opened 'SensorKinect' on bus 1:36 with serial number '0000000000000000'
[ INFO] [1464148276.326681077]: rgb_frame_id = 'camera_rgb_optical_frame'
[ INFO] [1464148276.326740870]: depth_frame_id = 'camera_depth_optical_frame'
[ WARN] [1464148276.329646631]: Camera calibration file /home/tesis/.ros/camera_info/rgb_0000000000000000.yaml not found.
[ WARN] [1464148276.329690693]: Using default parameters for RGB camera calibration.
[ WARN] [1464148276.329712864]: Camera calibration file /home/tesis/.ros/camera_info/depth_0000000000000000.yaml not found.
[ WARN] [1464148276.329730238]: Using default parameters for IR camera calibration.

```

Figura 29 Correcta detección del Kinect

Para ejecutar el `openni_tracker` es necesario tener corriendo el “`roslaunch openni_launch openni.launch`”, ya que este es el nodo master al cual accede `openni_tracker`. Además, se debe ejecutar los siguientes pasos en el orden mostrado a continuación:

1. Ingresar al espacio de trabajo creado por el terminal: “`cd nombre_espacio_trabajo`”
2. Ejecutar el comando “`catkin_make`”
3. Ejecutar el comando “`source devel/setup.bash`”
4. Ejecutar el commando “`roslaunch openni_tracker openni_tracker`”

Después de ingresar todos los comandos mencionados, se observa que se prende un foco de color rojo en el Kinect, el cual representa el sensor de profundidad (infrarrojo). Una vez ocurrido esto, el usuario se debe colocar frente al Kinect adoptando la pose PSI a la distancia de más o menos 1 metro y medio, tomando en cuenta que la cámara capte desde su torso para arriba. Si el Kinect detecta un usuario, aparecerá un mensaje en el terminal diciendo `new user #`; cuando este se quite de la cámara aparece el mensaje `lost user #`, (Ver Figura 30). Al obtenerse la pose PSI de forma correcta y la calibración esté lista, aparecerá un mensaje como el que se muestra en la Figura 31. En este punto se empezará a escanear el cuerpo para detectar la articulación escogida.

```

tesis@tesis-HP-ENVY-TS-15-Notebook-PC:~/catkin_ws$ source devel/setup.bash
tesis@tesis-HP-ENVY-TS-15-Notebook-PC:~/catkin_ws$ roslaunch openni_tracker openni
tracker
[ INFO] [1464149483.588702942]: New User 1
[ INFO] [1464149486.557429020]: New User 2
[ INFO] [1464149500.702879627]: Lost user 1
[ INFO] [1464149500.703155862]: Lost user 2

```

Figura 30 Detección /perdida de Usuario

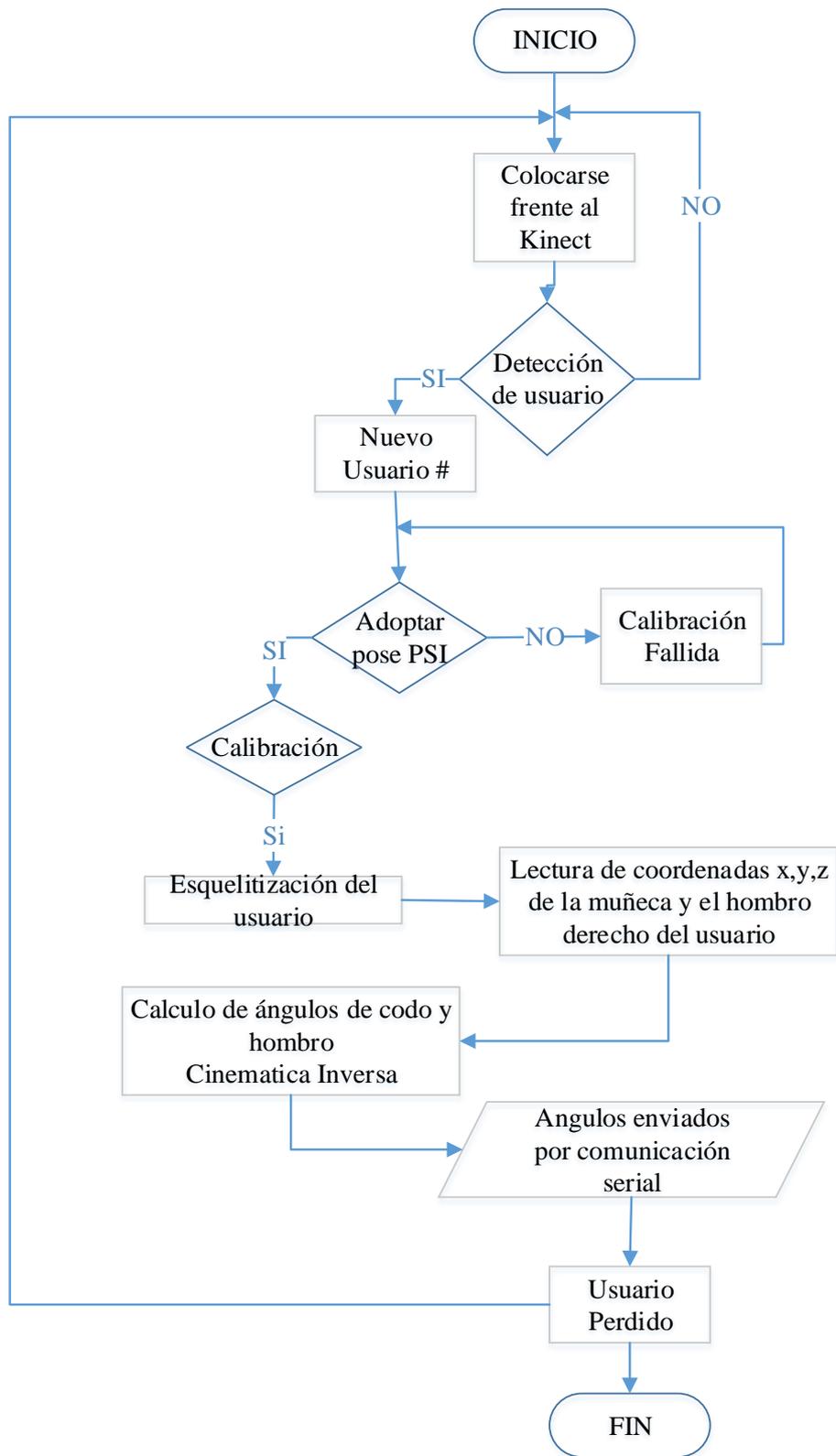


Figura 32 Diagrama de flujo Kinect

```

class Kinect:
    def __init__(self, name='openni_tracker', user=1):
        rospy.init_node(name, anonymous=True)
        self.listener = tf.TransformListener()
        self.user = user

```

Figura 33 Script detección usuario/selección articulación

En los dos scripts es necesario importar ciertas librerías para el correcto funcionamiento de los programas. Además, para que puedan ser ejecutados como archivos de Python, deben incluir siempre en la primera línea “#!/usr/bin/env/Python”. Estos dos scripts deben estar guardados como “nombre_script.py” respectivamente en: “espacio_trabajo/src/openni_tracker/src”. Cabe mencionar que una vez desarrollados estos scripts, es necesario volverlos ejecutables, para que luego sean ejecutados a través del comando rosrún; para ello, se debe ingresar a la carpeta donde están guardados a través del terminal y colocar el comando “**chmod +x nombre_script.py**”.

Una vez realizado todo este proceso, para correrlos se debe ejecutar los siguientes comandos:

1. “cd espacio_trabajo”
2. “catkin_make”
3. “source devel/setup.bash”
4. “rosrún openni_tracker nombre_script.py”

Es importante mencionar que ROS da la facilidad de autocompletar el comando que se desea escribir en el terminal presionando la tecla “tab”, lo que hace más fácil colocar todos los comandos durante todo el proceso de instalación y ejecución de los programas de manera más rápida y eficiente.

3.2.7 Ángulos de posicionamiento del brazo robótico a través de los datos obtenidos del Kinect

Una vez obtenidas las coordenadas de las diferentes articulaciones, la posición de la muñeca derecha y del hombro derecho, es necesario realizar la cinemática inversa para poder obtener los ángulos de movimiento del codo y el hombro respectivamente, para luego estos ser enviados al brazo robótico para que adopte la posición deseada.

Se realiza los cálculos de cinemática inversa para un brazo robótico antropomórfico y se definen 3 grados de libertad, debido a que solo se realizará 3 movimientos específicos (Ver Figura 34).

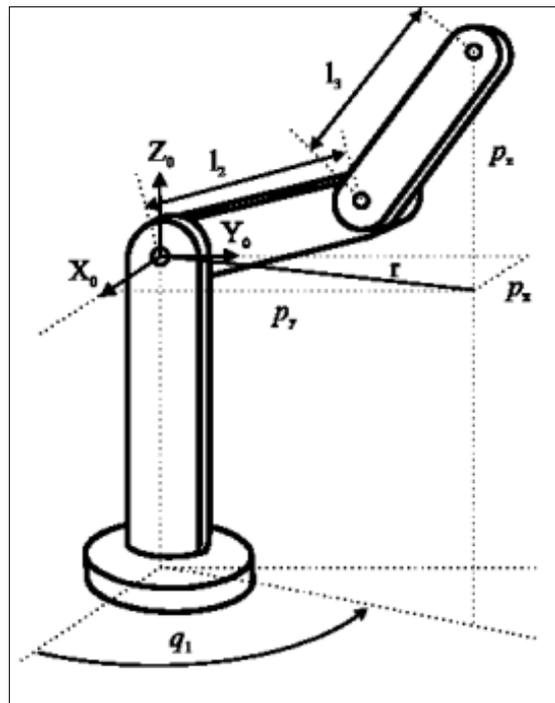


Figura 34 Robot antropomórfico / Ejes

Fuente: (Ramos & Iles)

Se desarrolla la cinemática inversa, debido a que se tiene como datos las coordenadas a las que se desea llegar. Se utilizó el método geométrico debido a que solo se utilizan tres grados de libertad, por el número de movimientos que realiza.

Para realizar estos cálculos es necesario contar con las medidas del largo del brazo y del antebrazo, las cuales son de 26 cm cada uno. Debido a que el Kinect lee las coordenadas en metros, estas medidas son transformadas para tener todo el sistema en el mismo sistema de unidades. Otro factor a considerarse es que se va a tomar como referencia al hombro y las coordenadas que da el Kinect son respecto al torso; por lo tanto, es necesario trasladar las coordenadas al nuevo punto de referencia.

A continuación, en la Figura 35 se muestra como se hace esta traslación de coordenadas.

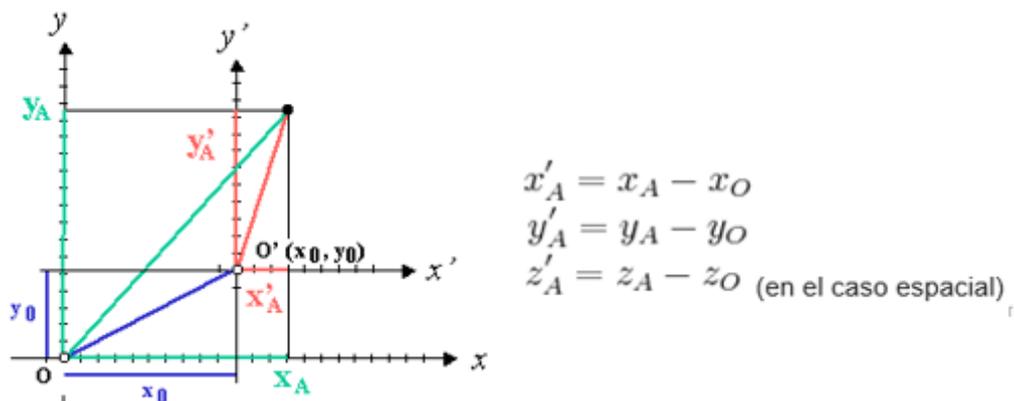


Figura 35 Traslación del origen – torso al hombro

Fuente: (Anonimo, 2011)

En la Tabla 6 se puede observar las variables que se muestran en la Figura 35 con su correspondiente posición de acuerdo a los datos obtenidos por el Kinect, tomando en cuenta que se obtiene como dato final la posición de la muñeca con respecto al hombro.

Como se mencionó anteriormente, los ejes de referencia del Kinect y del brazo robótico son diferentes, por lo que es necesario hacer el respectivo cambio para trabajar con las coordenadas definitivas para aplicar las fórmulas de la cinemática inversa por el método geométrico ($x_{robot}=z'A$, $y_{robot}=x'A$, $z_{robot}=y'A$).

Tabla 6

Variables correspondientes para la traslación del origen

Variable	Posición
O	Origen – Torso
O'	Origen – Hombro
x0	Coordenadas en x torso-hombro
y0	Coordenadas en y torso-hombro
z0	Coordenadas en z torso-hombro
xa	Coordenadas en x torso-muñeca
ya	Coordenadas en y torso-muñeca
Za	Coordenadas en z torso-muñeca
x'A	Posición final de la muñeca en x
y'A	Posición final de la muñeca en y
z'A	Posición final de la muñeca en z

Una vez realizado todas las traslaciones necesarias, se aplica las fórmulas mostradas a continuación para obtener los ángulos de orientación de las articulaciones deseadas:

$$q_1 = \frac{P_y}{P_x} \quad (13)$$

$$\cos(q_3) = \frac{P_x^2 + P_y^2 + P_z^2 - l_2 - l_3}{2l_2l_3} \quad (14)$$

$$q_2 = \beta - \alpha \quad (15)$$

$$\alpha = \arctan\left(\frac{l_3 \sin q_3}{l_2 + l_3 \cos q_3}\right) \quad (16)$$

$$\beta = \arctan\left(\frac{P_z}{\sqrt{P_x^2 + P_y^2}}\right) \quad (17)$$

Las ecuaciones (13-17) (Ramos & Iles) entregan los ángulos q_1 =ángulo cintura, q_2 = ángulo hombro y q_3 =ángulo codo; además de las longitudes l_2 =longitud brazo, l_3 =longitud antebrazo; y las coordenadas a las que se desea que llegue el brazo robótico P_x , P_y , P_z . Todos estos cálculos son realizados en el segundo script desarrollado en Python. Finalmente, los ángulos respectivos son enviados por comunicación serial al programa general para realizar el movimiento de las articulaciones del brazo robótico.

3.2.8 Implementación del algoritmo de detección de las articulaciones del usuario

Con los datos enviados por el puerto serial a la computadora, se realiza el procesado de los mismos, para luego ser enviados a la tarjeta STM32F4 y posteriormente al robot.

Es importante mencionar que para vigilar que la posición del usuario sea la correcta y la deseada, ROS permite obtener una monitorización en tiempo real de lo que el Kinect está capturando (Ver Figura 36). Para que esto sea posible, es necesario instalar a través de un terminal el paquete respectivo colocando el comando “sudo apt-get install ros-indigo-image-view”. Una vez realizada la instalación, se debe introducir el siguiente comando “roslaunch image_view image_view

image:=/camera/rgb/image_color” para obtener un cuadro con el cual se visualiza lo que la cámara del Kinect está capturando.

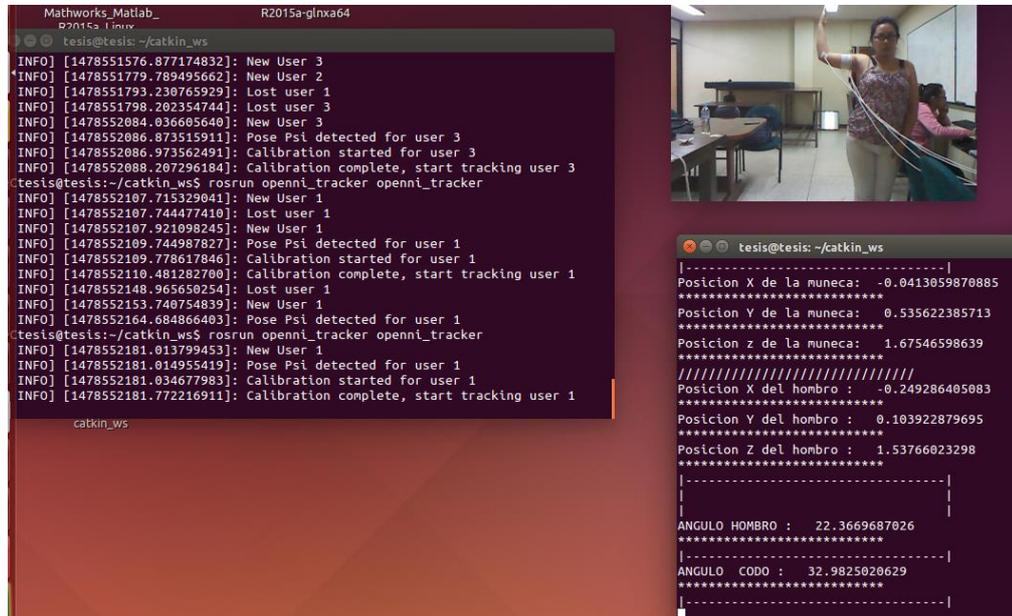


Figura 36 Detección de Usuario, ángulos codo y hombro, monitorización del usuario

3.3 Adquisición de datos MPU6050

La adquisición de los datos del MPU6050 se efectuó mediante la tarjeta STM32-F4 y un programa desarrollado en Matlab/Simulink. En el mismo se configuró la comunicación I2C y los registros internos del sensor inercial, permitiendo adquirir el posicionamiento del robot mediante la variación del ángulo de las articulaciones (hombro y codo). A continuación, se detalla cada una de las etapas de configuración.

3.3.1 Comunicación I2C

El módulo transmite datos en serie mediante la comunicación I2C, a una velocidad máxima de 400Kbits/s. Para esto se utiliza dos líneas: SDA (Serial Data Line) por la que se envían y reciben datos, y SCL (Serial Clock Line) en la que se envía la señal de reloj.

En este sistema se utilizaron dos módulos MPU6050, debido a que se va medir la posición del hombro y codo, los mismos que son ubicados en las dos articulaciones respectivamente. (Ver Figura 37). Estos módulos son conectados en paralelo a los pines B9 (SDA) y B8 (SCL) de la tarjeta STM32F4. Adicionalmente, se conecta el pin AD0 a VCC o GND respectivamente de cada módulo.



Figura 37 Ubicación de los MPU6050 en el brazo robótico

3.3.2 Configuración de registros

En el software Matlab/Simulink se programó la adquisición de los datos del módulo MPU6050, mediante los registros que tiene internamente este dispositivo.

Primero se debe configurar un protocolo de comunicación, definiendo dispositivos maestros y esclavos. Tomando en cuenta que solo los dispositivos maestros pueden iniciar una comunicación; se definió como dispositivo master a la tarjeta STM32F4 la cual inicia la transferencia en el bus y genera la señal de reloj. Los dispositivos esclavos serán los módulos MPU6050.

El direccionamiento del bus I2C implica 7 bits; es decir, se puede conectar hasta 128 dispositivos sobre un bus I2C. Se debe tener en cuenta el bit de lectura y escritura para poder iniciar la comunicación. El bit extra es necesario para saber si el

dispositivo esclavo va leer o escribir datos del dispositivo maestro. Si el bit de lectura/escritura (R/W) es cero, el dispositivo maestro está escribiendo en el esclavo, caso contrario está leyendo. El bit de lectura/escritura es el bit menos significativo.

Las direcciones del MPU6050 son 0x68 y 0x69; sin embargo, al adicionarle el bit de lectura las nuevas direcciones son 0xD1 y 0xD3 respectivamente.

Para obtener los valores puros de la aceleración en los tres ejes X, Y, Z se los adquirió de los registros internos. Las direcciones de los mismos se las obtuvo del Datasheet y de las librerías MPU6050.h y MPU6050.cpp.

Se obtuvo estos datos de forma dividida. Primero los 8 bits más significativos y aparte los 8 bits menos significativos debido a que el bloque de Simulink solo permite la configuración de 8 bits, agrupándose los datos para formar el registro de 16bits. La Tabla 7 muestra las direcciones que se utilizó, las cuales se obtuvieron de la librería del MPU6050

Tabla 7

Direcciones de la aceleración en los ejes X, Y y Z

Registros lsb: menos significativo msb: más significativo	Direcciones hexadecimales
x_lsb	0x3C
x_msb	0x3B
y_lsb	0x3E
y_msb	0x3D
z_lsb	0x40
z_msb	0x3F

Fuente: Librería MPU6050.h

<https://github.com/jrowberg/i2cdevlib/blob/master/Arduino/MPU6050/MPU6050.h>

3.3.3 Cálculo de ángulo de orientación

Una vez adquirida la aceleración en los ejes X, Y y Z, por trigonometría se calcula el ángulo de inclinación del MPU. De acuerdo a la posición que se ubicó los sensores en el robot, para el codo y hombro se logró adquirir el ángulo en Y.

$$\text{Angulo } Y = \tan^{-1} \left(\frac{x}{\sqrt{y^2+z^2}} \right) \quad (18)$$

3.4 Algoritmo para el movimiento del brazo robótico

Para realizar los movimientos de las articulaciones del robot (hombro, codo) se debe cumplir ciertas condiciones. Los diagramas de flujo de las Figuras 38 y 39 representan estas condiciones.

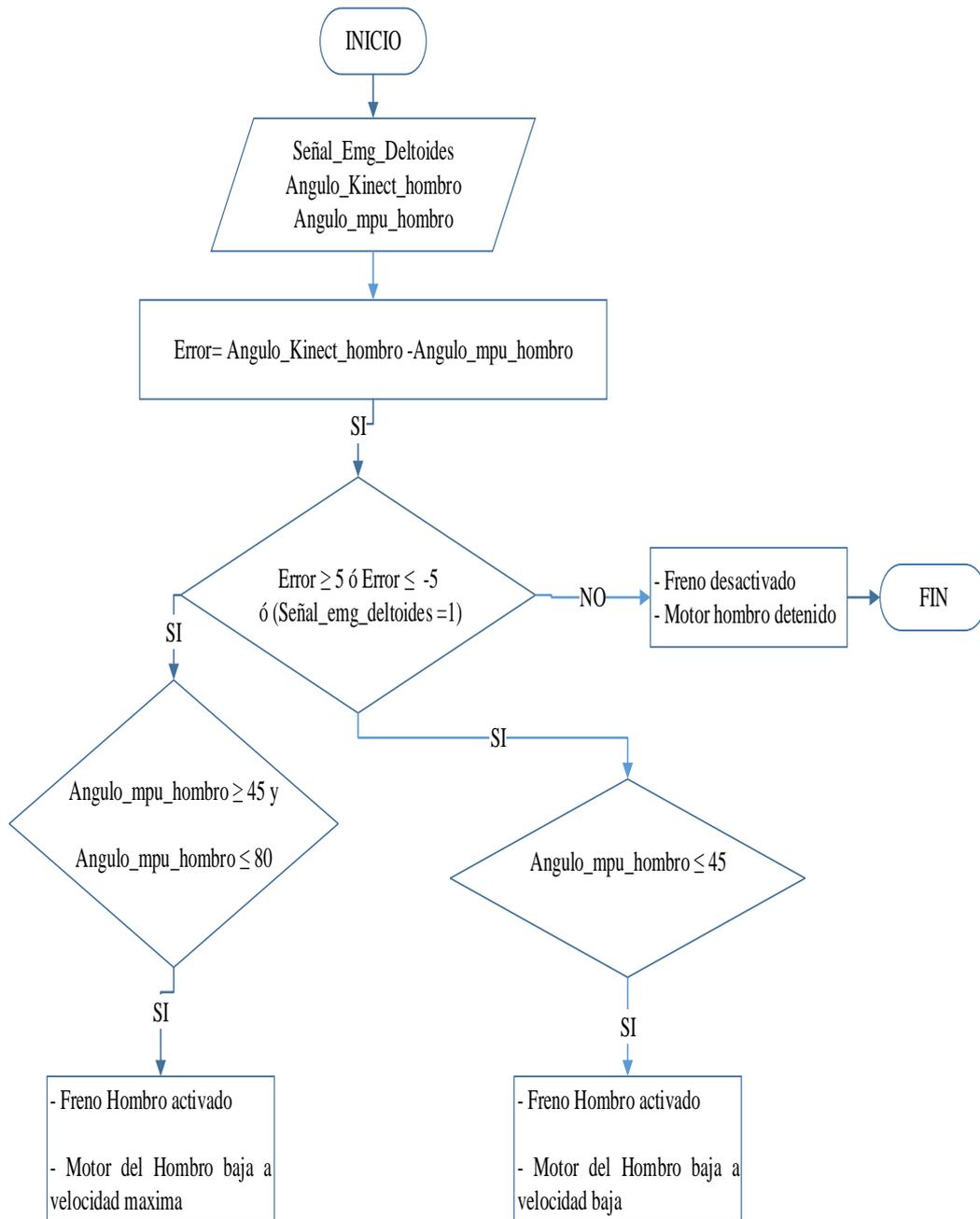


Figura 38 Diagrama de bloques activación hombro

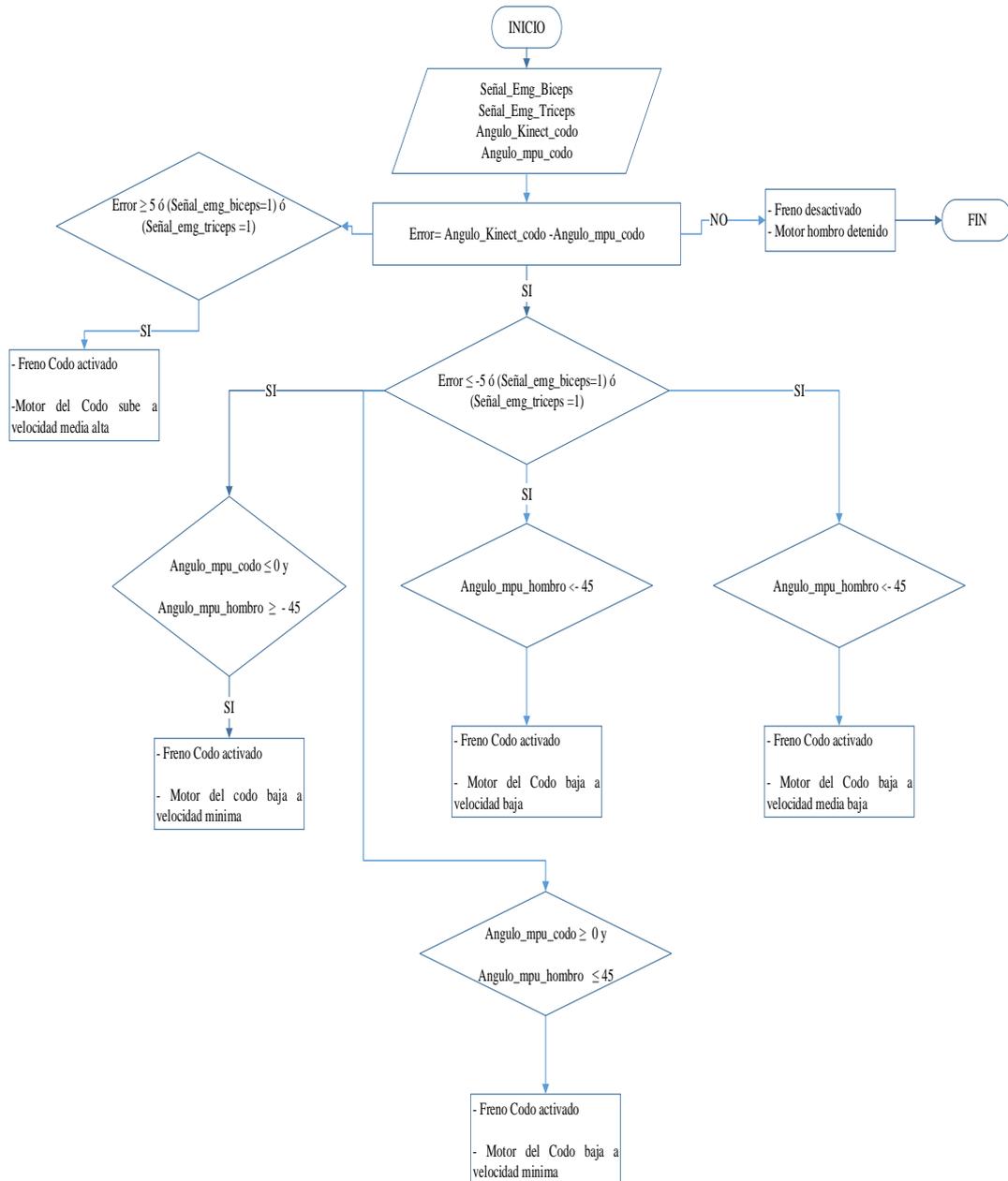


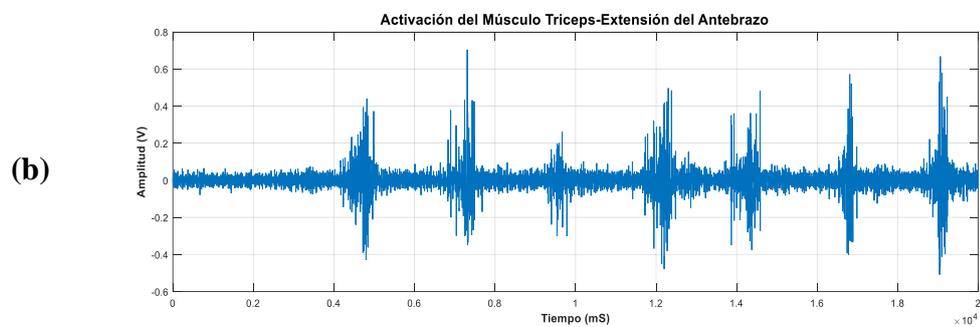
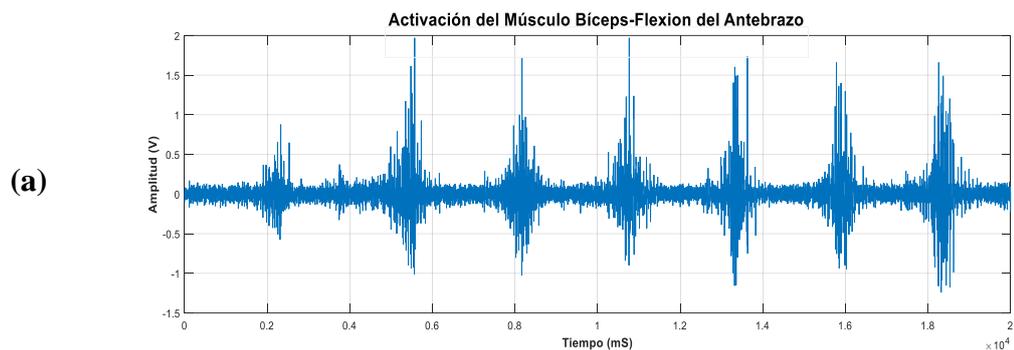
Figura 39 Diagrama de bloques activación codo

CAPÍTULO IV

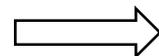
PRUEBAS Y RESULTADOS

4.1 Pruebas del sistema de adquisición de señales electromiográficas

Con las señales musculares adquiridas, se procedió a realizar su análisis y ver los músculos que se activan al realizar los diversos movimientos propuestos en capítulos anteriores; obteniendo como resultado que al realizar el movimiento de flexión del antebrazo se activa el músculo bíceps y si se realiza la extensión del mismo se activa el músculo tríceps. Por el contrario, si se realiza el movimiento de abducción del hombro, se activa el músculo deltoides. (Ver Figura 40).



CONTINÚA



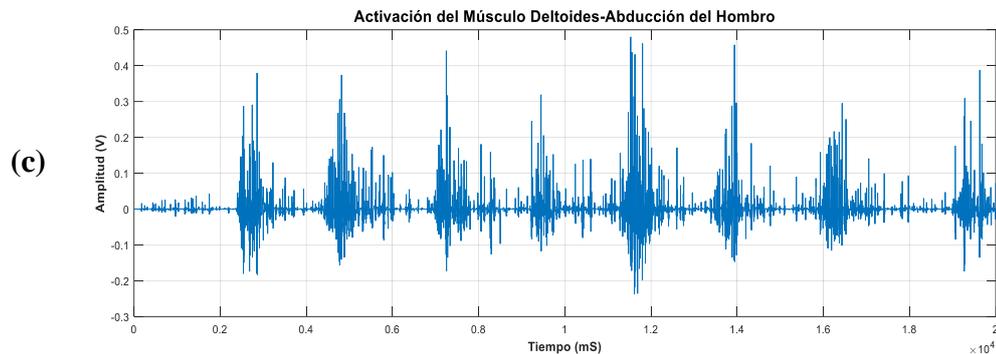


Figura 40 Activación de los músculos: a) Bíceps-Flexión del Antebrazo, b) Tríceps-Extensión del Antebrazo, c) Deltoides-Aducción del Hombro

Cabe mencionar que los análisis que se realizaron son importantes para conocer las características y los rangos en los que existe una activación de cada uno de los músculos, ya que esta es una de las condiciones para que se activen las articulaciones en el brazo robótico. Es importante mencionar el uso de tiras de LEDs, ya que cuando se realiza uno de los movimientos, los indicadores luminosos permiten definir qué músculo se activó y con qué intensidad.

4.1.1 Análisis en tiempo

En este análisis se toma en cuenta como varía el voltaje con respecto al tiempo, con el fin de identificar en donde inicia la activación muscular. La Figura 41 presenta varios picos en los diferentes músculos los cuales indican que se dio actividad muscular; tomando en cuenta que para cada señal existe un valor medio a partir del cual es evidente que el músculo está activado. Los picos pequeños muestran cuando el músculo se encuentra relajado.

Para realizar este análisis se utiliza un algoritmo en Matlab (Arcentales, 2016), el cual aplica el método RMS a cada una de las señales; bíceps, tríceps y deltoides. En base al ruido, se calculan los valores umbrales, comparando 1 segundo donde no hubo actividad muscular, con aquellos tramos donde sí existió.

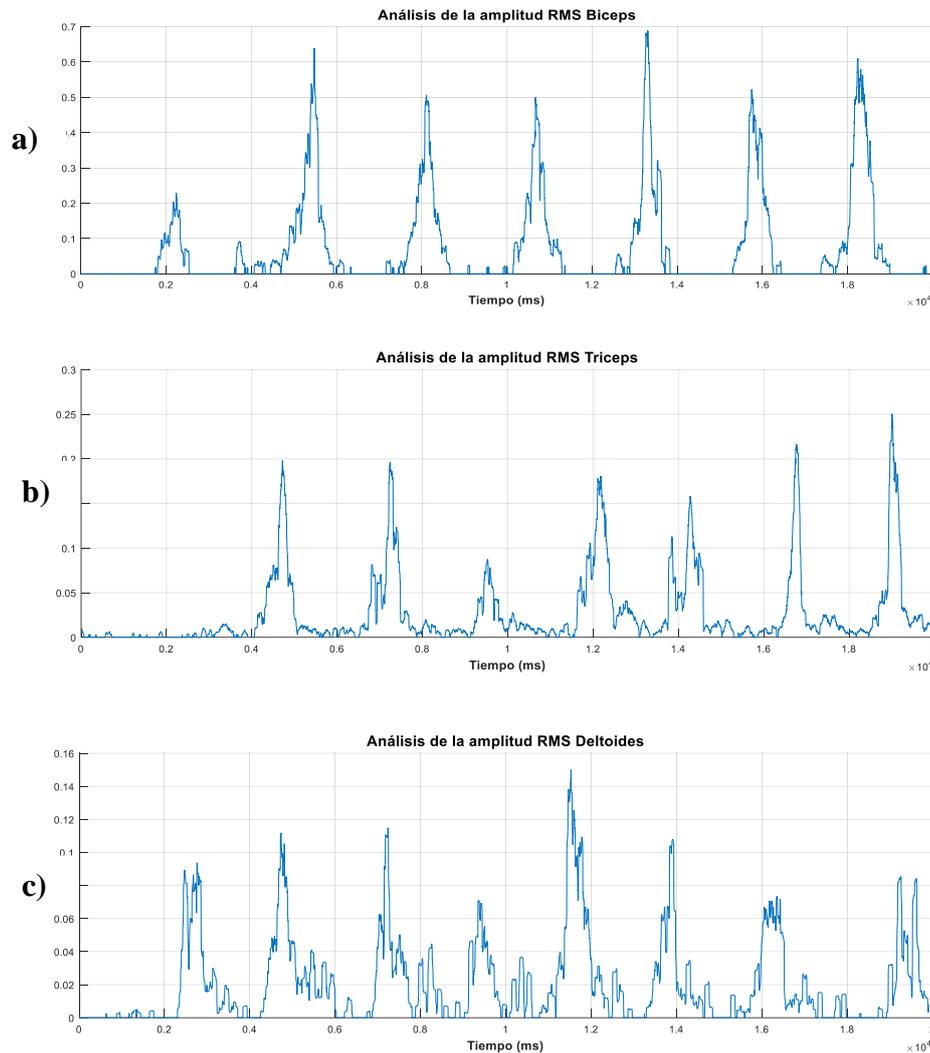


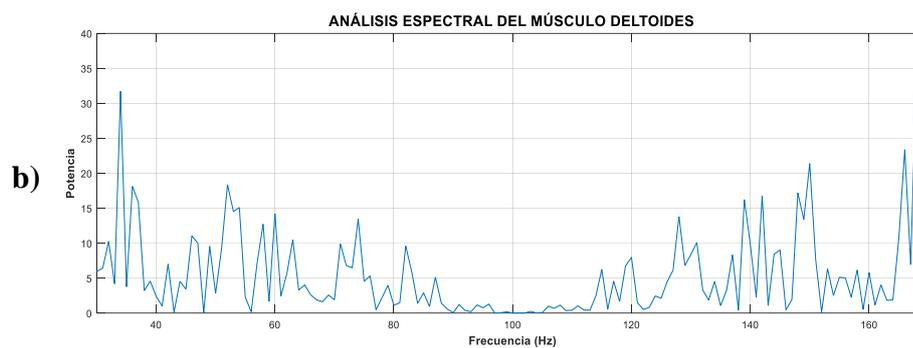
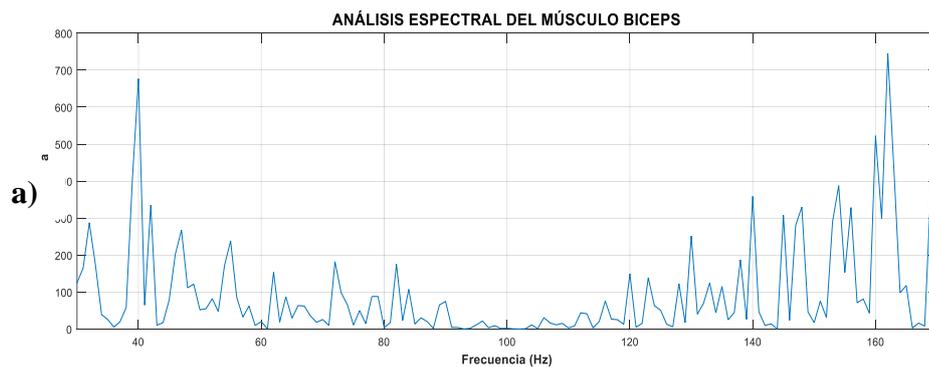
Figura 41 Representación gráfica del análisis RMS: a) Bíceps, b) Tríceps, c) Deltoides

4.1.2 Análisis en frecuencia

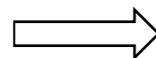
En esta etapa se realiza un análisis espectral, que consiste en realizar una gráfica de la señal de Potencia vs Frecuencia. Esta puede ser estudiada a través de la transformada de Fourier, la cual proporciona un método para la caracterización de la máxima contracción de la señal obtenida acorde al espectro de potencia. De forma general, este espectro se asemeja a una "U" con un pico alrededor de los 100-200 Hz y un rango aproximado entre 10 y 2 000 Hz. (Richfield EK, 1981)

Este análisis es muy utilizado en fatiga muscular ya que se puede observar que las bajas frecuencias aumentan cuando el sujeto de prueba se somete experimentalmente a fatiga, observando la modificación del espectro durante el registro de la señal. (Collahuazo Sinchi)

Se utilizó la función fft de Matlab, el cual da como resultado el espectro de cada señal obtenida en cada uno de los músculos en los que se trabaja. Como se trabaja con los músculos del brazo, la mayor información se concentra entre los 50 y 150 Hz. En los espectros se puede observar el pico de potencia de cada una de las señales, el cual indica que hubo una contracción del músculo. (Ver Figura 42).



CONTINÚA



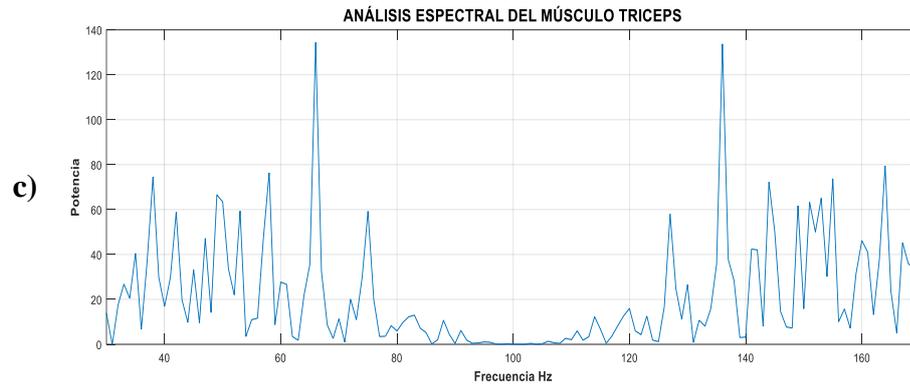


Figura 42 a) Espectro de frecuencia: a) Bíceps, b) Tríceps, c) Deltoides

4.2 Pruebas del protocolo de activación muscular

La activación muscular se basó en la variación de fuerza, para lo cual se calcula el valor cuadrático medio (RMS) de la señal electromiográfica de cada uno de los músculos, empleando la siguiente formula:

$$RMS = \left[\frac{1}{N} \sum_{i=0}^{N-1} x_i^2 \right]^{1/2} \quad (26)$$

Donde:

N es el valor de una ventana de la señal EMG

x_i es cada uno de los puntos de la señal muestreada.

Mediante este análisis se determina el umbral de activación de la señal EMG en cada uno de los músculos, el mismo que nos permite delimitar rangos alto, medio y bajo de amplitud, para ser visualizados en la variación de intensidad de los LEDs.

La Figura 43 muestra la señal EMG del bíceps aplicada el RMS en la cual se puede apreciar los rangos alto medio y bajo.

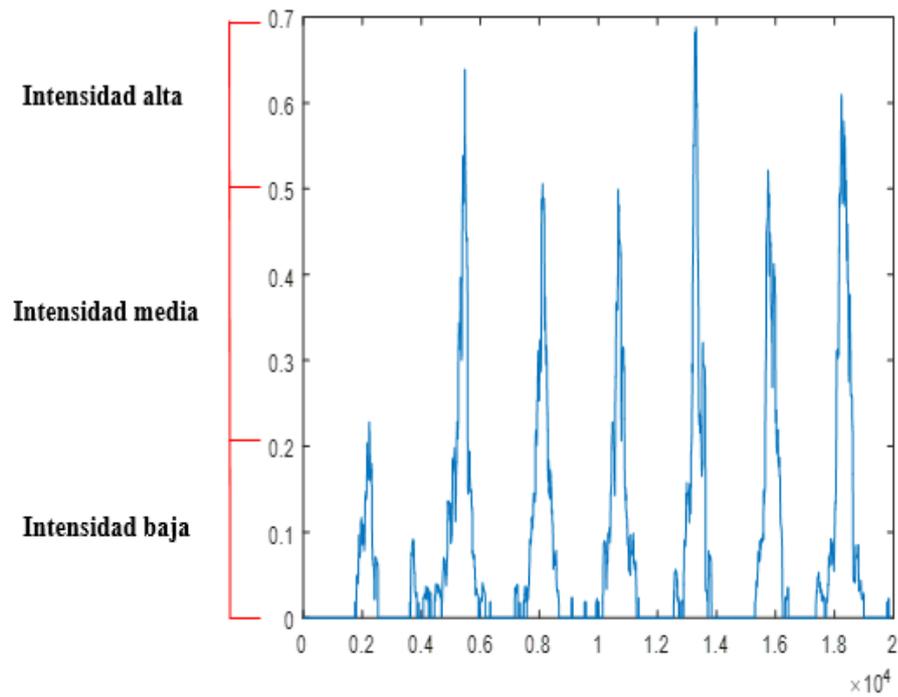


Figura 43 Rangos de activación muscular

Una vez desarrollado el algoritmo RMS y definido los rangos, en la Figura 44 se puede observar la representación de la activación de los músculos mediante tiras de LEDs.

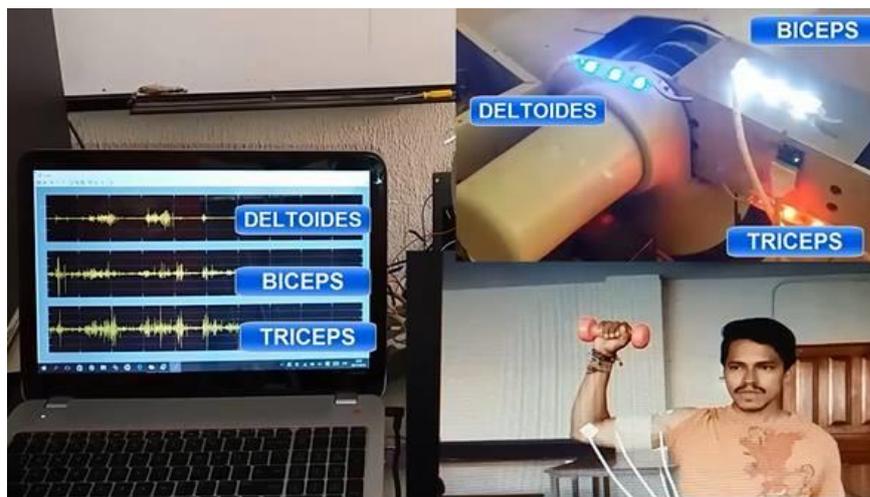


Figura 44 Representación de las activaciones de los músculos

4.3 Pruebas del Kinect

Se realizó una serie de pruebas para validar el funcionamiento del Kinect y verificar el algoritmo implementado de cinemática inversa.

4.3.1 Reconocimiento del usuario

El usuario adopta la pose PSI con la cual el Kinect lo detecta y permite el reconocimiento de la extremidad superior derecha para su posterior análisis (Ver Figura 45).



```
[ INFO] [1478708804.112754506]: New User 1  
[ INFO] [1478708819.900665594]: Pose Psi detected for user 1  
[ INFO] [1478708819.932963069]: Calibration started for user 1  
[ INFO] [1478708820.638795726]: Calibration complete, start tracking user 1
```

Figura 45 Reconocimiento del usuario (Pose PSI)

4.3.2 Ángulo del hombro 45°

La Figura 46 muestra la posición del hombro a 45° con respecto a la horizontal.



Figura 46 Ángulo del hombro 45°

A continuación, se muestra los cálculos del error relativo entre la posición que adopta el usuario y los cálculos que se realizan de acuerdo a la posición captada por el Kinect.

$$Error = \frac{Valor\ verdadero - valor\ aproximado}{valor\ verdadero} * 100\% \quad (19)$$

$$Error = \frac{45 - 44,8}{45} * 100\% \quad (20)$$

$$Error = 0,4\% \quad (21)$$

4.3.3 Ángulo del codo 60°

La Figura 47 nos permite visualizar la posición del codo a 60°.



Figura 47 Ángulo del codo 60°

Para el presente angulo generado, el calculo del error relativo se muestra a continuacion:

$$Error = \frac{60 - 59,03}{60} * 100\% \quad (22)$$

$$Error = 1,61\% \quad (23)$$

4.3.4 Ángulo del codo -45°

Como se puede observar en la Figura 48, la posición del codo se encuentra en 45°.

El error que se obtiene al realizar esta posición se presenta a continuación.

$$Error = \frac{-45 - (-45,04)}{-45} * 100\% \quad (24)$$

$$Error = 0,08\% \quad (25)$$



Figura 48 Ángulo del codo - 45°

La validación de los ángulos se determinó calculando del error relativo entre el ángulo de la extremidad superior derecha del usuario y el ángulo proporcionado por el Kinect luego de realizada la cinemática inversa; de lo cual se puede concluir que tiene un error considerable menor al 0,7% en promedio, el mismo que se da por movimientos del usuario mientras se realiza el procesamiento de los datos y por la exactitud con la que trabaja la cámara y el algoritmo aplicado sobre la Kinect.

4.4 Pruebas del procesamiento de información

Para el envío y recepción de información del sistema replicador se implementó dos tipos de comunicación: comunicación serial y comunicación I2C. Los parámetros de configuración de la comunicación serial se los puede observar en la Tabla 8.

Tabla 8

Configuración serial

Velocidad de transferencia	9600 baudios
Bits de datos	8
Bits de paro	1
Paridad	Ninguna
Timeout	6

Los datos del MPU6050 son transmitidos por medio de comunicación I2C utilizando una velocidad de transmisión de 400Kbits/s, mientras que los datos del Kinect son transmitidos por medio de comunicación serial. (Ver Figura 49)

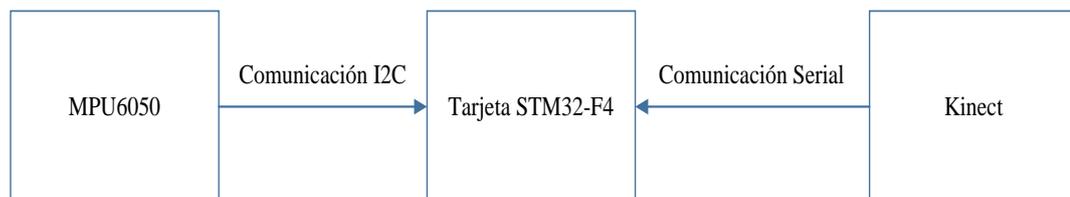


Figura 49 Diagrama de bloques comunicación del sistema replicador de movimientos

4.5 Pruebas y resultados globales en ambiente

Para validar el correcto funcionamiento del sistema replicador implementado, se realizó movimientos repetitivos de la extremidad superior derecha del usuario y se determinó el error absoluto y el tiempo en que el robot adopta dicha posición.

En la Tabla 9 se muestra los resultados de cinco repeticiones de la posición del hombro de 45° a 0° , en la cual se determina que el error absoluto en promedio es de 2,21. Esto indica que existe una buena precisión en el posicionamiento del usuario y el brazo robótico. También se realizó el cálculo del tiempo de respuesta en adoptar la posición, el mismo que en promedio es 1,26 segundos, lo cual es un retardo considerable para que el brazo robótico adopte su posición (Ver Figura 50).



Figura 50 Posición hombro del brazo humano y brazo robótico a 45°

Tabla 9

Posición del hombro de 45° a 0°

Hombro 45° a 0°				
Ángulo Kinect (grados)	Ángulo mpu inicial (grados)	Ángulo mpu final (grados)	Error absoluto (grados)	Tiempo(s)
0,32	44,8	0,11	0,21	1,22
0,58	46,53	-2,74	3,31	1,25
1,26	46,38	-0,14	1,39	1,26
2,39	42,76	-0,10	2,48	1,32
1,06	45,7	-2,59	3,65	1,27
Error promedio			2,21	
Tiempo promedio				1,26

Cabe mencionar que debido a fallas mecánicas, el motor del hombro solo puede moverse en un sentido, por lo que solo se puede realizar movimientos de 70° a 0° .

En la Tabla 10 se muestra los resultados de cinco repeticiones de la posición del codo de 0° a 60° , en la cual se determina que el error absoluto en promedio es de 3,30. Este error se da debido a que en el movimiento del codo se implementó diferentes velocidades en diferentes rangos de ángulos, por lo cual también existe un tiempo de respuesta alto de 29,68 segundos para el brazo robótico adopte la posición del usuario (Ver Figura 51).



Figura 51 Posición del codo del brazo humano y brazo robótico a 60°

Tabla 10

Posición del codo de 0° a 60°

Codo 0° a 60°				
Ángulo Kinect (grados)	Ángulo mpu inicial (grados)	Ángulo mpu final (grados)	Error absoluto (grados)	Tiempo(s)
49,99	0,2143	48,72	1,27	29
50,23	3,315	45,96	4,27	28,66
CONTINÚA				➡

51,82	1,395	49,49	2,32	30
49,06	2,485	44,35	4,71	31,34
50,12	3,65	46,17	3,95	29,4
Error promedio			3,30	
Tiempo promedio				29,68

En la Tabla 11 se muestra los resultados de cinco repeticiones de la posición del codo de 0° a -45° , en la cual se determina que el error absoluto en promedio es de 1,80. En comparación con el error de la prueba anterior es menor, debido a que al realizar un movimiento menor a 0° influye el peso y la gravedad para el movimiento se realice más rápido. En esta prueba se obtuvo un tiempo de respuesta promedio de 9,15 segundos (Ver Figura 52).



Figura 52 Posición del codo del brazo humano y brazo robótico a -45°

Tabla 11**Posición del hombro de 0° a -45°**

Codo 0° a -45°				
Ángulo kinect	Ángulo mpu inicial	Ángulo mpu final	Error absoluto	Tiempo(s)
-44,87	1,433	-41,82	-3,05	8,56
-46,03	0,955	-43,82	-2,21	9,03
-45,09	1,19	-43,38	-1,71	9,21
-44,54	1,142	-43,39	-1,15	9,54
-46,26	1,137	-45,37	-0,89	9,42
Error promedio			1,80	
Tiempo promedio				9,15

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- En la industria el presente proyecto sería de gran ayuda debido a que existen lugares en que los operadores no pueden acceder y se debe realizar movimientos controlados de los robots para las diferentes actividades que se realiza en los procesos de producción.
- Al integrar todo el sistema, es importante tomar en cuenta los tiempos de respuesta con los que trabaja la tarjeta STM32f4, así como también el tiempo que se demora en enviar el Kinect y recibir los datos Matlab por el puerto serial; principalmente para la sincronización de todos los actuadores y sensores que conforman el sistema.
- Se ha seleccionado el diseño más adecuado y seguro para el usuario en la adquisición de las señales EMG, logrando obtener un sistema que detecta las señales de 3 músculos.
- El uso del dispositivo Kinect genera una versatilidad de opciones para poder trabajar más allá de los datos de las coordenadas de las articulaciones que se puede obtener de un usuario. Más aun, el skeleton tracking permite detectar más de un usuario a la vez. También, el micrófono permite utilizar comandos de voz, lo cual incrementa la variedad de aplicaciones o mejoras que se puede realizar en el sistema implementado.
- El sistema replicador desarrollado es capaz de realizar los tres tipos de movimientos: flexión/extensión del antebrazo y abducción del hombro;

obteniéndose un error del $\pm 5^\circ$ cuando el brazo robótico adopta la posición del brazo humano.

- El sistema EMG indica el inicio de la activación y su potencia de acuerdo a los movimientos de las articulaciones, los mismos que se visualizan en la variación de intensidad de las tiras LEDs.

5.2 Recomendaciones

- El registro de señales EMG está inmersa a diferentes factores que alteran la señal y generan ruido, por lo que se recomienda utilizar cables blindados y entorchados para disminuir interferencias magnéticas; además es importante una correcta ubicación de los electrodos.
- Para trabajos futuros se sugiere realizar una clasificación y entrenamiento de la señal EMG con diferentes pesos para establecer patrones y características de cada señal.
- Se debe tener presente los sistemas de referencia con los que trabaja el Kinect, brazo robótico y MPUS, para realizar las traslaciones y transformaciones necesarias para manejar los 3 componentes sobre el mismo sistema y así obtener los datos necesarios de forma correcta.
- En un sistema con diferentes protocolos de comunicación es necesario sincronizar el tiempo de transmisión y recepción del flujo de información.

REFERENCIAS BIBLIOGRÁFICAS

- A255 Robot Arm User Guide*. (2000). Robotics Corporation, CRS.
- Aaron, M. (21 de Junio de 2014). *Robotic Operating System*. Obtenido de ROS.org:
<http://wiki.ros.org/ROS/Concepts>
- Anonimo. (29 de 11 de 2011). *Traslacion de sistemas de coordenadas*. Obtenido de
http://enciclopedia.us.es/index.php/Sistema_de_coordenadas_cartesianas
- Arcentales, A. (2016). Detección de contracciones señales EMG. Quito, Pichincha, Ecuador.
- Ballesteros, S. (2012). *Sistema de teleoperación mediante el Kinect*. España.
- Bernal, J. L. (2012). *Universidad Nacional de Colombia*. Obtenido de
<http://www.bdigital.unal.edu.co/42964/1/80492327.2013.pdf>
- Carletti, E. (2007). *Robots Comunicación - Bus I2C*. Obtenido de http://robots-argentina.com.ar/Comunicacion_busI2C.htm
- Ciencias Biológicas y Educación para la Salud*. (s.f.). Obtenido de
http://hnnbiol.blogspot.com/2008/01/sistema-muscular_21.html
- Collahuazo Sinchi, J. P. (s.f.). Diseño y construcción de una mano robot activada por señales electromiográficas.
- Corporation, C. R. (2000). *A255 Robot Arm User Guide*.
- Corsino, E. L. (3 de Febrero de 2001). *Saludmed- Partes de la extremidad superior del ser humano*. Obtenido de
<http://www.saludmed.com/CsEjerci/Cinesiol/CinEscap.html>
- De Luca, C. (2006). *Electromyography. Encyclopedia of Medical Devices and Instrumentation* .

ECAM estetic. (s.f.). Instructivo de colocacion de Electrodo. *ECAM estetic*, 15.

Electromiograma. (s.f.). Obtenido de <http://www.webconsultas.com/pruebas-medicas/electromiograma-12209>

Enciclopedia Salud. (s.f.). Obtenido de <http://educacion.uncomo.com/articulo/cuales-son-las-extremidades-del-cuerpo-humano-24392.html>

Integración de dispositivos Leap y Kinect. (2013). Obtenido de <http://animusproject.wix.com/web/apps/blog/integraci%C3%B3n-de-dispositivos-leap-y-kinect>

Kinect XBOX-360. (s.f.). Obtenido de Microsoft: <http://www.xbox.com/en-US/xbox-360/accessories/kinect>

Marner, M. (30 de Junio de 2011). *Prerrequisitos de instalación Kinect*. Obtenido de <http://www.20papercups.net/programming/kinect-on-ubuntu-with-openni/>

Martín, J. L. (Febrero de 1998). *Aducción del Hombro*. Obtenido de ELSEVIER: <http://www.elsevier.es/en-revista-fisioterapia-146-articulo-isocineticos-el-hombro-13008935>

Massó, N., & Rey, F. (2010). Aplicaciones de la electromiografía de superficie en el deporte. *Apunts Medicina de L'Esport*, 10.

Neuman, M. R. (2000). *The Biomedical Engineering Handbook*. CRC Press LLC.

Ñacato, D. (2014). *Diseño e implemntacion de un sistema de teleoperacion para controlar un robot humanoide mediante un sensor Kinect*. Ecuador.

Ochoa, M., Aguiar, G., & Erazo, A. (2015). RHINO – an autonomous interactive surveillance robot for the needed. *MATEC Web of Conferences*, 5.

OMANGIN. (2011). *Openni Tracker*. Obtenido de GitHubGist: <https://gist.github.com/omangin/2414166>

OpenNI Programmer Guide. (11 de Mayo de 2012). Obtenido de <http://openni.org/Documentation/ProgrammerGuide.html>.

Pardell, X. (s.f.). *Apuntes de electromedicina*. Obtenido de <http://www.pardell.es/electromiografo.html>

Ramarras, J. R. (27 de Agosto de 2014). *Blog de Programacion y Robotica*. Obtenido de <https://jjromeromarras.wordpress.com/2014/08/27/conceptos-basicos-de-ros/>

Ramos, G., & Iles, P. (s.f.). *Cinematica Inversa-Robot Antropomorfico*. Obtenido de Repositorio UTN: <http://repositorio.utn.edu.ec/bitstream/123456789/1100/8/04%20ISC%2005%20Cap%20II.pdf>

Richfield EK, C. B. (1981). Review of quantitative and automated meedle electromyographic analysis. *IEEE Trans Biomed Eng*, 28:506-14.

Rivera, P. (2015). *Diseño e Implementación de técnicas de control para el controlador CAD del manipulador Robótico CRS A255*. Quito.

Sanz, I. (6 de Noviembre de 2014). *Partes del Brazo*. Obtenido de SPORHUESCA: <http://www.sporhuesca.com/fracturas-de-radio/>

Sistema de desarrollo STM32F4-Discovery. (s.f.). Obtenido de Siconetics: http://www.siconetics.com/wfiles/Afiche_Stm32F4-Discovery.pdf

Skeleton Tracking. (5 de Noviembre de 2015). Obtenido de ROS ANSWERS: <http://answers.ros.org/question/214421/how-can-i-setup-skeleton-tracking-using-a-kinect-and-ros-indigo-on-ubuntu-1404/?comment=228387#post-id-228387>

STM32F4-DISCOVERY. (s.f.). Obtenido de Electronics, Mouser: <http://www.mouser.com/Electronic-Components/?Stocked=False&RoHSCompliant=False>

Tiras LEDs. (2012). Obtenido de Meetthings:
<http://meetthings.com/blog/index.php/02-11-2011/como-elegir-y-como-se-conectan-las-tiras-de-leds/>

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

Tutorial MPU6050, Acelerómetro y Giroscopio. (2015). Obtenido de Mechatronics, Naylamp:
http://www.naylampmechatronics.com/blog/45_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html

Velasco, G. (s.f.). *Robotic Operating System.* Obtenido de ROS.org:
<http://wiki.ros.org/es>