



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

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA E  
INSTRUMENTACIÓN**

**PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL  
TÍTULO DE INGENIERO EN ELECTRÓNICA E  
INSTRUMENTACIÓN**

**TEMA: DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA  
PARA EL CONTROL DE UN ROBOT MÓVIL PARA ACCESO A  
LUGARES REMOTOS UTILIZANDO TECNOLOGÍA KINECT**

**AUTORES: MARÍA GABRIELA CHÁVEZ RUGEL**

**ÍVAN MARCELO TATAYO SIMBAÑA**

**DIRECTOR: MSc. EDDIE GALARZA**

**CODIRECTOR: MSc. DAVID RIVAS**

**LATACUNGA**

**2015**

UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE

INGENIERÍA ELECTRÓNICA E INSTRUMENTACIÓN

## **CERTIFICADO**

MSC EDDIE GALARZA (DIRECTOR)

MSC DAVID RIVAS (CODIRECTOR)

### **CERTIFICAN**

Que el trabajo titulado “DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA PARA EL CONTROL DE UN ROBOT MÓVIL PARA ACCESO A LUGARES REMOTOS UTILIZANDO TECNOLOGÍA KINECT” realizado por la señora María Gabriela Chávez Rugel y el señor Iván Marcelo Tatayo Simbaña , ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la Universidad de las Fuerzas Armadas ESPE.

Debido a que constituye un trabajo de excelente contenido científico y aplicable para el desarrollo profesional, se recomienda su aplicación.

El mencionado trabajo consta de un documento empastado y un disco compacto el cual contiene los archivos en formato portátil de Acrobat (PDF). Autorizan la señora María Gabriela Chávez Rugel y el señor Iván Marcelo Tatayo Simbaña, que lo entregue al Ing. Franklin Silva, en su calidad de Director de Carrera.

Latacunga, Agosto del 2015.

MSc. Eddie Galarza

Director

MSc. David Rivas

Codirector

UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE

INGENIERÍA ELECTRÓNICA E INSTRUMENTACIÓN

## **DECLARACIÓN DE RESPONSABILIDAD**

Nosotros, MARÍA GABRIELA CHÁVEZ RUGEL  
IVÁN MARCELO TATAYO SIMBAÑA

### **DECLARAMOS QUE:**

El proyecto de grado denominado “DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA PARA EL CONTROL DE UN ROBOT MÓVIL PARA ACCESO A LUGARES REMOTOS UTILIZANDO TECNOLOGÍA KINECT” ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme a las citas que constan en el pie de las paginas correspondientes, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance

Latacunga, Agosto 2015.

María Gabriela Chávez Rugel

C.C: 1803521424

Iván Marcelo Tatayo Simbaña

C.C: 1718173386

UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE

INGENIERÍA ELECTRÓNICA E INSTRUMENTACIÓN

## **AUTORIZACIÓN DE PUBLICACIÓN**

Nosotros, MARÍA GABRIELA CHÁVEZ RUGEL  
IVÁN MARCELO TATAYO SIMBAÑA

Autorizamos a la Universidad de las Fuerzas Armadas- ESPE la publicación, en la biblioteca virtual de la institución del trabajo, “DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA PARA EL CONTROL DE UN ROBOT MÓVIL PARA ACCESO A LUGARES REMOTOS UTILIZANDO TECNOLOGÍA KINECT”, cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y auditoria.

Latacunga, Agosto 2015.

María Gabriela Chávez Rugel

C.C: 1803521424

Iván Marcelo Tatayo Simbaña

C.C: 1718173386

## DEDICATORIA

*Bienaventurado el hombre que halla la sabiduría, y que obtiene la inteligencia; porque su ganancia es mejor que la ganancia de la plata, y sus frutos más que el oro fino. Proverbios 2.7*

Este logro obtenido se lo dedico a Dios porque me ha dado la vida, la fuerza, la perseverancia, la sabiduría y por nunca dejarme sola cuando más lo he necesitado. A mi madre, Melida, que con su gran ejemplo de mujer luchadora y virtuosa me enseñó que nunca debemos rendirnos y que siempre estará a mi lado para cuando la necesite, por su amor y apoyo incondicional, a mi padre, Holguer, por ser un ejemplo de lucha y constancia, a mi segunda madre y hermana Sarita y Alejandra por ser mi apoyo, mi ejemplo, mi fortaleza , a mi esposo, Josué por ser mi mejor amigo, confidente, consejero, compañero de vida por su ayuda que me han permitido creer más en mí y cumplir mis sueños y sobre todo a mi hija, Dana Noemí es mi más hermoso regalo de vida. A todas estas personas les dedico todos mis logros, porque gracias a ustedes soy lo que soy.

***María Gabriela Chávez Rugel***

Dedico este proyecto de tesis principalmente a Dios, que con su amor me iluminó para seguir el sendero del bien. Y tan grande es su amor que me obsequió a mis padres, Pablo y Rosalía, a quienes con mucho amor dedico este trabajo, que significa el fruto de su esfuerzo, de su amor, del aliento que cada día supieron dar a mi vida, que me enseñaron a nunca rendirme, a siempre ser humilde sobre todas las cosas, a siempre hacer lo correcto y hoy me dejan una gran herencia llena de valores. A mis hermanas que nunca se rindan, siempre luchen por salir adelante, por cada día avanzar un escalón más en el pasaje de la vida. Y sobre todo a mi princesa mágica Darla Jamileth que es la razón por la que día a día tenga la fortaleza de levantarme y superarme.

***Iván Marcelo Tatayo Simbaña***

## **AGRADECIMIENTO**

A Dios principalmente porque es el quien le ha dado un verdadero sentido a mi vida, a mi familia, por darme su amor incondicional, por su apoyo en todas las decisiones que he tomado.

A los ingenieros Eddie Galarza y David Rivas, director y codirector de tesis respectivamente, por sus valiosas guías para que este proyecto pueda ser culminado. Gracias a todas las personas que han sido parte de este éxito y me apoyaron en los momentos más difíciles en mi carrera.

***María Gabriela Chávez Rugel***

"El triunfo no está en vencer siempre, sino en nunca desanimarse"

Napoleón

Un profundo agradecimiento a Dios, quien me dio la vida y la vida a mis padres y siempre nos tiene en su regazo con salud y fuerzas para salir adelante. A mis padres que con su esfuerzo y apoyo diario hicieron que mis fuerzas nunca decaigan ni rendirme ante nada y siempre perseverar a través de sus sabios consejos. A mis hermanas Ivonne y Paola, por su apoyo incondicional y por demostrarme la gran fe que tienen en mí. A mi esposa por su amor y apoyo incondicional, a mi hija Darla puesto que con su ternura angelical siempre fue luz para continuar en mi camino tanto académico como intelectual.

A mi director de tesis Ing. Eddie Galarza quien me impulso con su conocimiento, paciencia y ejemplo a seguir siempre el camino correcto para alcanzar la meta propuesta siendo así posible el desarrollo total de mi proyecto. A mi codirector de tesis Ing. David Rivas por su apoyo y confianza depositado en mí para realizar el proyecto de graduación, brindándome la oportunidad para la culminación satisfactoria del mismo. Gracias a todas las personas que ayudaron directa e indirectamente en la realización de esta tesis.

***Iván Marcelo Tatayo Simbaña***

## ÍNDICE DE CONTENIDO

<b>PORTADA</b> .....	i
<b>CERTIFICADO</b> .....	ii
<b>DECLARACIÓN DE RESPONSABILIDAD</b> .....	iii
<b>AUTORIZACIÓN DE PUBLICACIÓN</b> .....	iv
<b>DEDICATORIA</b> .....	v
<b>AGRADECIMIENTO</b> .....	vi
<b>ÍNDICE DE CONTENIDO</b> .....	vii
<b>ÍNDICE DE TABLAS</b> .....	xi
<b>ÍNDICE DE FIGURAS</b> .....	xii
<b>RESUMEN</b> .....	xvi
<b>ABSTRACT</b> .....	xvii
<b>CAPÍTULO I</b> .....	1
<b>1. PLANTEAMIENTO DEL PROBLEMA EN INVESTIGACIÓN</b> .....	1
1.1 Antecedentes.....	1
1.2 Planteamiento del problema de investigación.....	2
1.3 Objetivos.....	3
1.3.1 Objetivo General: .....	3
1.4 Marco teórico .....	3
1.4.1 Kinect.....	3
a. Referencias Kinect.....	6
a.1 Aplicaciones Del Sensor Kinect Para Mejorar La Salud.....	6
a.1.1 Kinect Habilitación: .....	6
a.1.2 Aplicación Para Enfermos De Alzheimer .....	7
a.1.3 Kinect En los Quirófanos.....	7
a.1.4 Turtlebot o Camarero Robótico .....	8
a.1.5 Espejo Virtual Para Tiendas De Ropa .....	8
b. Sensor Kinect.....	8
b.1 Software Para Controlar El Sensor Kinect .....	8
b.2 Kinect Para Windows .....	9

b.3.	Componentes Hardware .....	10
b.4	Especificaciones .....	11
c.	Funcionamiento .....	13
c.1	Chip Primesense 1080 .....	15
d.	Arquitectura .....	17
d.1	Nui Api .....	18
d.2	Data Streams .....	19
d.3	Segmentación De Datos .....	19
d.4	Profundidad Extended Data .....	20
d.5	Skeletal Tracking.....	21
d.5.1	Visión de Conjunto .....	21
d.5.2	Campo de Visión.....	22
1.4.2	Arduino Xbee Shield .....	24
a.	Carga Del Programa A La Placa Arduino Uno .....	25
1.4.3	Robot Móvil .....	27
a.	Misión, Navegación y Operación .....	28
b.	Tecnologías de Construcción Y Conexión Senso-Motora .....	29
<b>CAPÍTULO II.....</b>		<b>33</b>
<b>2. DISEÑO DEL SISTEMA.....</b>		<b>33</b>
2.1	Estudio De Los Códigos Fuente SDK De Kinect .....	33
2.1.1	Skeletal Tracking.....	33
2.1.2	Camera Fundamental .....	35
2.1.3	Depth Data.....	36
2.2	Diseño Del Software .....	36
2.2.1	Diseño De La Aplicación Demostrativa En C# .....	36
a.	Formulario Carátula (Tesis GI) .....	38
a.1	Análisis Formulario Carátula (Tesis GI) .....	38
a.2	Diseño Formulario Carátula (Tesis GI).....	38
a.3	Resultado de Formulario Carátula .....	39
b.	Formulario Aplicación Demostrativa .....	39
b.1	Análisis Formulario Aplicación Demostrativa .....	39
b.2	Diseño Formulario Aplicación Demostrativa .....	40



b.3	Resultado Formulario Aplicación Demostrativa .....	41
2.2.2	Programación En Visual Studio C# .....	42
a.	Diagrama de Flujo De La Programación General.....	42
b.	Diagrama de flujo de la Codificación programa Arduino UnoR3..	43
c.	Hardware Utilizado .....	44
2.3	Diagrama de Conexiones .....	45
2.3.1	Diagrama Esquemático.....	46
2.3.2	Diseño de Circuito Impreso (PCB) .....	47
2.3.3	Diseño Robot Móvil.....	47
<b>CAPÍTULO III.....</b>		<b>50</b>
<b>3.</b>	<b>RED INALAMBRICA ZIGBEE .....</b>	<b>50</b>
3.1	Estándar Zigbee .....	50
3.2	Aplicaciones Zigbee.....	52
3.2.1	Diferentes Aplicaciones De Zigbee .....	52
a.	Automatización De Edificios .....	53
b.	Cuidado de la Salud .....	53
c.	Control Industrial .....	54
d.	Automatización de hogar.....	55
e.	Medición Automática. ....	55
f.	Telecomunicaciones.....	55
3.3	Arquitectura Zigbee.....	56
3.4	Tipos De Dispositivo Zigbee .....	57
3.5	Topologías De Red.....	59
3.5.1	Topología Estrella .....	59
3.5.2	Topología en Árbol.....	59
3.5.3	Topología en Malla.....	59
3.5.4	Ventajas Y Desventajas De Cada Topología .....	59
3.6	Seguridad .....	61
3.6.1	Arquitectura De Seguridad.....	61
3.7	Módulo Xbee.....	62
3.7.1	Características .....	62
3.7.2	Conexión Placa Arduino.....	63

a.	Cable Virtual I/O .....	63
b.	Hardware .....	65
b.1	Microcontrolador .....	65
b.2	E/S Digitales .....	65
b.3	Entradas Analógicas .....	65
b.4	Salidas PWM .....	66
b.5	Comunicaciones .....	66
c.	Software .....	66
3.7.3	Red Zigbee .....	67
a.	Instalacion Y Configuracion X-Ctu .....	67
<b>CAPÍTULO IV .....</b>		<b>70</b>
<b>4.</b>	<b>ANÁLISIS DE RESULTADOS.....</b>	<b>70</b>
4.1	Distancia Usuario - Sensor Kinect Y Estaturas.....	70
4.2	Estatura Del Usuario.....	92
4.3	Número De Usuarios.....	92
<b>CAPÍTULO V .....</b>		<b>94</b>
<b>5.</b>	<b>CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>94</b>
5.1	Conclusiones .....	94
5.2	Recomendaciones .....	95
<b>REFERENCIAS BIBLIOGRAFICAS .....</b>		<b>97</b>
<b>ANEXOS.....</b>		<b>100</b>

## ÍNDICE DE TABLAS

TABLA 3.1 Propiedades Del Estándar IEEE 802.15.4.....	51
TABLA 3.2 Ventajas Y Desventajas De Las Topologías De Red.....	60

## ÍNDICE DE FIGURAS

Figura 1.1	Kinect Xbox 360.....	4
Figura 1.2	Software De Windows Para Negocios .....	5
Figura 1.3	Kinect Windows Para Interactuar Con Un Terapeuta .....	6
Figura 1.4	Consulta De Imágenes Kinect Habilidadación Mediante El Sensor Kinect .....	7
Figura 1.1.5	Consulta De Imágenes Radiológicas Mediante El Sensor Kinect.....	7
Figura 1.6	Kit De Desarrollo Kinect O Sdk.....	9
Figura 1.7	Componente Hardware De La Luz Led.....	11
Figura 1.8	Componentes De Hardware Del Kinect .....	11
Figura 1.9	Funcionamiento Del Sensor De Movimiento En Kinect.....	13
Figura 1.10	Etapas Del Funcionamiento Del Sensor Kinect .....	14
Figura 1.11	Intensidad De Imagen Y Profundidad Codificada En Escala De Grises Para Un Retrato.....	14
Figura 1.12	Sistema Que Conformar El Chip Prime Sensor .....	16
Figura 1.13	Arquitectura Kinect.....	17
Figura 1.14	Forma De Kinect Para Reconocer Al Skeletal Tracking.....	21
Figura 1.15	Kinect Es Capaz De Reconocer A Seis Personas Y Realizar Seguimiento De Dos De Ellas .....	21
Figura 1.16	El Esqueleto De Seguimiento Está Diseñado Para Reconocer A Los Usuarios En Frente Del Sensor.....	22
Figura 1.17	Rango Del Campo De Visión Horizontal Del Sensor Kinect	23
Figura 1.18	Rango Del Campo De Visión Vertical Del Sensor Kinect....	23
Figura 1.19	Modulo Xbee.....	24
Figura 1.20	Tarjeta Arduino Uno.....	24
Figura 1.21	Xbee Shield Montado Sobre La Placa Arduino .....	25
Figura 1.22	Modelos De Robot Móvil.....	27
Figura 1.23	Sistema Autónomo Robot Móvil Más Brazo Robótico.....	27
Figura 1.24	Esquema Básico De La Arquitectura De Un Robot Móvil Para Realizar Una Misión .....	29
Figura 1.25	Odometría Del Robot Móvil.....	30
Figura 2.1	Diseño Formulario Carátula .....	38

Figura 2.2	Carátula Programada En Visual C# .....	39
Figura 2.3	Diseño Formulario Aplicación .....	40
Figura 2.4	Resultado Formulario Aplicación .....	41
Figura 2.5	Hardware Utilizado.....	44
Figura 2.6	Logo Del Software Libre Fritzing.....	45
Figura 2.7	Diagrama De Conexiones Del Robot Móvil.....	46
Figura 2.8	Diagrama Esquemático Del Robot Móvil .....	46
Figura 2.9	Diseño Pcb Vista Inferior .....	47
Figura 2.10	Diseño Pcb Vista Superior .....	47
Figura 2.11	Estructura Final Del Robot Móvil.....	48
Figura 2.12	Diseño Del Robot Móvil Interna .....	49
Figura 2.13	Diseño Del Robot Móvil Externa .....	49
Figura 3.1	Aplicaciones De Zigbee .....	52
Figura 3.2	Aplicación De Zigbee En La Automatización De Un Edificio53	
Figura 3.3	Aplicación De Zigbee En El Cuidado De La Salud.....	54
Figura 3.4	Aplicación De Zigbee En Control Industrial.....	54
Figura 3.5	Aplicación De Zigbee En Automatización De Hogar .....	55
Figura 3.6	Aplicación De Zigbee En Telecomunicaciones .....	56
Figura 3.7	Arquitectura Zigbee.....	56
Figura 3.8	Tipos De Topologías Que Se Puede Configurar En Una Red Zigbee.....	59
Figura 3.9	Esquema Para La Conexión Del Módulo Zigbee En La Placa Arduino. ....	63
Figura 3.10	Diagrama De Pines Del Módulo Xbee. Vista Superior .....	64
Figura 3.11	Instalación Del Programa X-Ctu.....	67
Figura 3.12	Instalación Del Programa X-Ctu.....	67
Figura 3.13	Instalación Del Programa X-Ctu.....	68
Figura 3.14	Instalación Del Programa X-Ctu.....	68
Figura 3.15	Instalación Del Programa X-Ctu.....	68
Figura 3.16	Instalación Del Programa X-Ctu.....	69
Figura 3.17	Pantallas Del Programa X-Ctu .....	69
Figura 4.1	Distancia Kinect (1m) Usuario1(1.2m) .....	71
Figura 4.2	Distancia Kinect (1m) Usuario1 (1.2m) .....	72
Figura 4.3	Distancia Kinect (1m) Usuario1 (1.2m) .....	72

Figura 4.4	Distancia Kinect (1m) Usuario1 (1.2m) .....	73
Figura 4.5	Distancia Kinect (1m) Usuario2 (1.6m) .....	73
Figura 4.6	Distancia Kinect (1m) Usuario2 (1.6m) .....	74
Figura 4.7	Distancia Kinect (1m) Usuario2 (1.6m) .....	74
Figura 4.8	Distancia Kinect (1m) Usuario2 (1.6m) .....	75
Figura 4.9	Distancia Kinect (1m) Usuario3 (1.75m) .....	75
Figura 4.10	Distancia Kinect (1m) Usuario3 (1.75m) .....	76
Figura 4.11	Distancia Kinect (1m) Usuario3 (1.75m) .....	76
Figura 4.12	Distancia Kinect (1m) Usuario3 (1.75m) .....	77
Figura 4.13	Distancia Kinect (2m) Usuario1 (1.2m) .....	78
Figura 4.14	Distancia Kinect (2m) Usuario1 (1.2m) .....	78
Figura 4.15	Distancia Kinect (2m) Usuario1 (1.2m) .....	79
Figura 4.16	Distancia Kinect (2m) Usuario2 (1.6m) .....	79
Figura 4.17	Distancia Kinect (2m) Usuario2 (1.6m) .....	80
Figura 4.18	Distancia Kinect (2m) Usuario2 (1.6m) .....	80
Figura 4.19	Distancia Kinect (2m) Usuario3 (1.75m) .....	81
Figura 4.20	Distancia Kinect (2m) Usuario3 (1.75m) .....	81
Figura 4.21	Distancia Kinect (2m) Usuario3 (1.75m) .....	82
Figura 4.22	Distancia Kinect (2m) Usuario (1.75m) .....	82
Figura 4.23	Distancia Kinect (3m) Usuario1 (1.2m) .....	83
Figura 4.24	Distancia Kinect (3m) Usuario1 (1.2m) .....	84
Figura 4.25	Distancia Kinect (3m) Usuario1 (1.2m) .....	84
Figura 4.26	Distancia Kinect (3m) Usuario1 (1.2m) .....	85
Figura 4.27	Distancia Kinect (3m) Usuario2 (1.6m) .....	86
Figura 4.28	Distancia Kinect (3m) Usuario2 (1.6m) .....	86
Figura 4.29	Distancia Kinect (3m) Usuario2 (1.6m) .....	87
Figura 4.30	Distancia Kinect (3m) Usuario2 (1.6m) .....	87
Figura 4.31	Distancia Kinect (3m) Usuario2 (1.6m) .....	88
Figura 4.32	Distancia Kinect (3m) Usuario3(1.75m) .....	88
Figura 4.33	Distancia Kinect (3m) Usuario3 (1.75m) .....	89
Figura 4.34	Distancia Kinect (3m) Usuario3 (1.75m) .....	89
Figura 4.35	Distancia Kinect (3m) Usuario3 (1.75m) .....	90
Figura 4.36	Grafico De Probabilidades De Funcionamiento Con Las 3 Diferentes Distancias Realizadas En El Proyecto .....	90

Figura 4.37	Grafico De Probabilidades De Funcionamiento Con Las 3 Diferentes Distancias Realizadas En El Proyecto .....	91
-------------	--	----

## **RESUMEN**

Este proyecto de desarrollo e implementación de un sistema de control para un robot móvil es manejado mediante tecnología Kinect. Kinect cuenta con una cámara RGB, un sensor de profundidad, un sensor CMOS de infrarrojos, un micrófono de múltiples matrices, luz Led, un motor y un procesador especializado Prime Sensor que capta el entorno en 3 dimensiones para traducirlas a una imagen sincronizada. Para el desarrollo de la aplicación demostrativa se empleó el análisis y estudio de los códigos fuentes SDK de Kinect, los mismos que permitirán control de un robot móvil en lugares remotos. El robot móvil consta con 4 grados de libertad y posee 4 servomotores, los mismos que permiten controlar los movimientos tanto en forma horizontal cuando el usuario del barrido horizontal de la mano derecha (eje x), el movimiento vertical, cuando el usuario se acerque o aleje del sensor Kinect utilizando el brazo izquierdo (eje y), y la rotación cuando el usuario haga un desplazamiento con la mano derecha (eje z). La interfaz y los datos que se adquieren de los distintos movimientos del usuario al mover el robot móvil, podrán ser visualizados en la aplicación desarrollada en Visual Studio 2010. Los datos son enviados por medio de la comunicación serial de Visual Studio C# a la placa Arduino, la misma que se va a encargar de recibir esos datos y enviarlos al ZigBee emisor para de esta manera conectarnos inalámbricamente al ZigBee receptor para dar el movimiento a los distintos servomotores.

### **PALABRAS CLAVE:**

- ✓ **ROBÓTICA**
- ✓ **ROBOT MOVIL**
- ✓ **TECNOLOGÍA KINECT**
- ✓ **RED INALÁMBRICA ZIGBEE**



## **ABSTRACT**

This project of development and implementation of a control system for a mobile robot is driven through Kinect technology. Kinect has a RGB camera, a depth sensor, an infrared CMOS sensor, a multiple matrix microphone, LED light, a motor and a specialized Prime Sensor processor that captures the surroundings in three dimensions and then translates it to a synchronized image. The analysis and study of root codes of the Kinect SDK codes were used for the development of the demo application, same codes that will allow the control of a mobile robot at remote places. The mobile robot has 4 degrees of freedom and process 4 servomotors which allow horizontal movements when the user sweeps his right hand, vertical movement (X axis) when the user moves closer or apart from the Kinect sensor with the left arm (Y axis), and the rotation when the user makes a displacement with the right hand (Z axis). The interface and data obtained from the different user movements as the mobile robot is moved could be visualized in the Visual Studio 2010 application. The data is sent through the serial communication of the Visual Studio C# to the Arduino plaque, which is in charge of receiving the data and sending it to the ZigBee emitter so that a wireless connection could be accomplished with the Zigbee receptor and move the servomotors.

## **KEYWORDS**

- ✓ **ROBOTICS**
- ✓ **MOBILE ROBOT**
- ✓ **TECHNOLOGY KINECT**
- ✓ **ZIGBEE WIRELESS NETWORK**



## CAPÍTULO I

### 1. PLANTEAMIENTO DEL PROBLEMA EN INVESTIGACIÓN

#### 1.1 Antecedentes

Dentro de la tecnología Kinect Windows se han realizado varias aplicaciones de las cuales se pueden destacar el Hotel Kiosk, que es una aplicación interactiva para viajeros que utiliza nuevos gestos, incluidos los más recientes que se han integrado a Kinect para Windows SDK. Esta aplicación permite a los huéspedes conocer eventos locales y puntos de interés, así como mapas que les ayudan a encontrar lo que buscan. Dentro de la Universidad y carrera se ha implementado el tema "Desarrollo e implementación de un algoritmos para detección de objetos con tecnología Kinect", el cual tuvo como aportación la adquisición de los datos mediante esta tecnología.

La plataforma de hardware libre llamada Arduino ha desarrollado muchas aplicaciones como NetIO que es una aplicación que permite crear interfaces personalizadas basadas en componentes simples como etiquetas, botones, conmutadores. Una vez diseñada la interfaz a nuestras necesidades, cada interacción con estos componentes enviará una orden al Arduino, que actuará en consecuencia. De igual forma, también puede solicitar datos al Arduino para mostrarlos en la pantalla de nuestro móvil la diferencia es que se enfocan en una aplicación específica, Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

Dentro de la Tarjeta Arduino y la comunicación de Zigbee se han realizado diferentes aplicaciones cabe recalcar que no dentro de la

institución, como el control de un Joystick, Combine eso con el poder de la palanca de mando y un Arduino para dirigir un barco o un avión.

## **1.2 Planteamiento del problema de investigación**

En la actualidad existe dependencia el usuario y los dispositivos electrónicos, en donde para su manejo siempre ha sido necesario que el usuario realice un contacto físico con el dispositivo para que este funcione, cabe recalcar que hoy en día existen tecnologías que abren una puerta hacia la innovación y la creatividad, las cuales nos permiten una gama de herramientas para desarrollar diversos sistemas complejos de control, y reconocimiento de comandos de voz y gestos, pero que en nuestro País no se está aplicando debido al desconocimiento o falta de acceso a los mismos.

En la Carrera de Ingeniería Electrónica e Instrumentación es importante tener conocimientos tecnológicos actualizados para poder ser aplicados en las diferentes áreas, que nos son abastecidas mayormente en el ámbito universitario, con lo cual es importante generar la aplicación de investigación actual y futura.

En los laboratorios dentro de la institución Escuela Politécnica Sede Latacunga, No se han realizados aplicaciones mediante esta tecnología lo cual abre puertas para las próximas generaciones y sus aplicaciones, es importante mencionar que varias de las comunicaciones como Zigbee se han tomado en cuenta en varias asignaturas con lo cual se es una base de conocimiento, la cual debe ser complementada con las destrezas de los estudiantes y con la nueva información a plantearse.

## **1.3 Objetivos**

### **1.3.1 Objetivo General:**

Desarrollar e implementar un sistema para el control de un robot móvil para acceso a lugares remotos utilizando tecnología kinect.

### **1.3.2 Objetivos Específicos:**

- ✓ Realizar una aplicación para la detección del movimiento del cuerpo humano.
- ✓ Analizar el lenguaje de programación del SDK del sensor Kinect basado en C#.
- ✓ Establecer el control de un Robot Móvil en comunicación Inalámbrica desde el Kinect.
- ✓ Comunicar todos los dispositivos de Kinect y Robot Móvil mediante Zigbee.

## **1.4 Marco teórico**

### **1.4.1 Kinect**

Kinect Xbox 360 permite jugar con la consola y tener diferentes experiencias sin la obligación de utilizar controles. Este dispositivo utiliza micrófonos de matriz, además de un patrón de punto IR proyectado, cámara IR y una cámara RGB. El sensor Kinect puede aislar y grabar sonidos también crear un mapa a profundidad del cuarto, y generar modelos en 3D de rostros humanos y esqueletos.

Kinect brinda la oportunidad de jugar con todo el cuerpo, este dispositivo controla cabeza, brazos, piernas, rodillas, cintura, cadera, etc.

La experiencia del entretenimiento en varias áreas y la forma de jugar fue variada gracias al Kinect Xbox 360. Al modificar la forma en que las personas pueden interactuar con las computadoras y otros dispositivos de control de movimiento, con lo cual ha variado totalmente el vínculo que existía entre computador y el ser humano.[1]



**Figura 1.1 Kinect Xbox 360**

**Fuente: [1]**

En la actualidad existen gran cantidad de personas que han empezado investigaciones y programaciones de muchas aplicaciones utilizando este mecanismo y aprovechando al máximo las ventajas que van mucho más allá del pasatiempo de videojuegos, todo esto gracias al Software SDK de Kinect.

Microsoft en sus primeras versiones lanzó aplicaciones demo para comprender y desarrollar nuevas aplicaciones. Una de las más importantes es Skeletal Tracking el cual se reconoce al cuerpo humano y capta sus movimientos, Otro demo que utiliza la base de conocimiento del Skeletal Tracking es el Shape Game, Kinect Audio Demo reconoce el comando de voz para realizar una determinada acción, (disponible en inglés). La última versión presenta una mejora notoria tanto en la calidad gráfica como en la interfaz de usuario en comparación con las aplicaciones anteriores, así también se añadió nuevas opciones para el programador, como incorporar Avatares (personajes virtuales), el reconocimiento del rostro de personas en 3D, etc.

Las características principales de Kinect son:

- ✓ Sistemas operativos (Windows 7 / Windows 8 / Windows Embedded Standard 7 / Windows Embedded POSReady 7).
- ✓ 32 bits (x86) o 64 bits (x64).
- ✓ Procesador dual-core de 2,66 GHz o más rápido.
- ✓ USB 2.0.
- ✓ GB de RAM

Gran cantidad de personas en todo el mundo han conseguido disfrutar y manejar el sensor de Kinect para el kit de desarrollo de software de Windows SDK (licencia), para poder desarrollar y desplegar soluciones.



**Figura 1.2 Software de Windows para negocios**

**Fuente: [2]**

Kinect permite al usuario realizar movimientos tales como empujar su mano hacia adelante para oprimir un botón o cerrar las manos para agarrar. Kinect en el campo de terapia física permite interactuar con el ordenador sin salir del lado de su paciente



**Figura 1.3 Kinect Windows para interactuar con un terapeuta**

**Fuente: [2]**

#### **a. Referencias Kinect**

##### **a.1 Aplicaciones Del Sensor Kinect Para Mejorar La Salud**

###### **a.1.1 Kinect Habilitación:**

El campo de la salud es aquel que más beneficio y provecho ha adquirido con Kinect, ya que este tiene la capacidad de reconocer al cuerpo humano ya sea con gestos o movimientos.

En España se están realizando varios proyectos de ejercicios de rehabilitación con este periférico, un ejemplo es el de la universidad Rey Juan Carlos, en el que Kinect actúa como un completo rehabilitador en la esclerosis múltiple con el cual el paciente desde el confort de su hogar realiza video conferencia de ejercicios coordinados con su fisioterapeuta.[2]





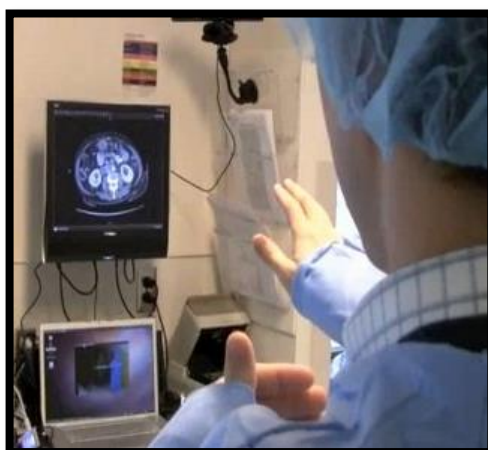
**Figura 1.4 Consulta de Imágenes Kinect habilitación Mediante El Sensor Kinect**

**Fuente: [2]**

### **a.1.2 Aplicación Para Enfermos De Alzheimer**

Dentro del campo de la medicina e investigación Kinect también es utilizado por personas enfermos de Alzheimer, existe la fundación ASISPA (Asociación de servicio integral sectorial para ancianos) cuyo centro de apoyo ayuda a mejorar la calidad de vida de los ancianos afectados por esta enfermedad.

### **a.1.3 Kinect En los Quirófanos**



**Figura 1.5 Consulta De Imágenes Radiológicas Mediante El Sensor Kinect**

**Fuente: [2]**

El sensor Kinect ha sido introducido para el uso en operaciones quirúrgicas, las cuales se han realizado mediante un ordenador. Con este dispositivo el cirujano ya no necesita manipular nada para controlar el sistema en tiempo real, con este procedimiento se reducen en un 99.99% el riesgo de infecciones. Esta aplicación fue gracias a una empresa líder en el uso de Kinect llamada TEDESYS.

**a.1.4 Turtlebot o Camarero Robótico**

Es Turtlebot un prototipo de robot económico, el cual está diseñado para captar el ambiente donde se encuentra fácilmente. El camarero robótico posee componentes básicos tales como un computador portátil, un robot Irobot Create y un sensor 3D Microsoft Kinect. Las órdenes que son enviadas al robot son ingresadas en la portátil para ser ejecutadas por el robot y así poder dejar a un lado al recurso humano optimizando tiempo y costos. [3]

**a.1.5 Espejo Virtual Para Tiendas De Ropa**

Esta aplicación no solo aporta a la tecnología sino que también es una herramienta que ayuda a la industria en este caso en tiendas de ropa, mediante un avatar 3D en tiempo real imita al cliente al probarse la vestimenta de la tienda, dando experiencia en compras virtual. Todo esto gracias a MAGIC MIRROR de Intel. [4]

**b. Sensor Kinect****b.1 Software Para Controlar El Sensor Kinect**

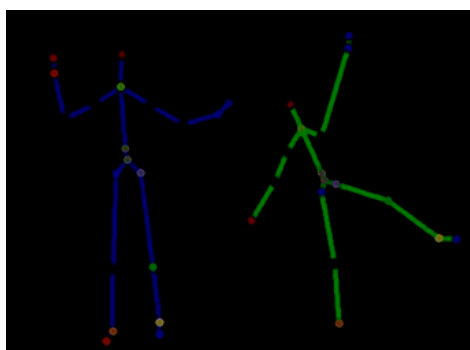
Para este software es necesario por lo menos 190 MB para su almacenamiento, el cual permite a los usuarios utilizar la interfaz de la consola Xbox 360 mediante gestos de las manos y los comandos de voz.

Kinect emplea técnicas para la identificación automática de los diferentes usuarios, para el reconocimiento de voz y facial. Entre las aplicaciones de vídeo que utiliza Kinect para realizar chat de voz o chat de vídeo con los usuarios u otros usuarios de Xbox 360 se encuentra Windows Live

Messenger. Esta aplicación puede usar la funcionalidad de seguimiento Kinect y el sensor de giro motorizado para ajustar la cámara para que el usuario se mantenga en el marco establecido, incluso en movimiento. [5]

Se han publicado aproximadamente 20 videojuegos para Kinect, a partir de septiembre del 2009 hasta el año 2015, se anunciaron desarrolladores de videojuegos para Kinect confirmados por Microsoft. Entre los cuales se encuentran Activision Blizzard, Bethesda Softworks, Capcom, Disney Interactive, Electronic Arts, Konami, MTV Games, Namco Bandai, Sega, Square Enix, THQ Inc. y Ubisoft.

Microsoft lanzó un kit oficial de desarrollo de software de Windows para su sensor Kinect o SDK. El cual según Microsoft, lo hizo más simple para los investigadores académicos y entusiastas, y así poder crear nuevas interfaces de usuarios naturales manejando la tecnología Kinect. Los desarrolladores tendrán acceso al audio, APIs y el control directo del sensor Kinect. [6]



**Figura 1.6 Kit de Desarrollo Kinect o SDK**

**Fuente: [7]**

## **b.2 Kinect Para Windows**

El Sensor Kinect para Windows está optimizado para experiencias en Windows, con Windows PC, y soportan el “Modo Cercano”.

Los diferentes desarrolladores pueden manejar cualquier sensor para empezar, pero requieren de un sensor Kinect para Windows que podría ser comprado en línea.

SDK Kinect ofrece para Windows la descarga que incluye controladores (drivers), APIs integrales para flujos de sensor no procesados, interfaces de usuario naturales, documentos de instalación, además de otros materiales de recursos.

Para usar SDK se necesita:

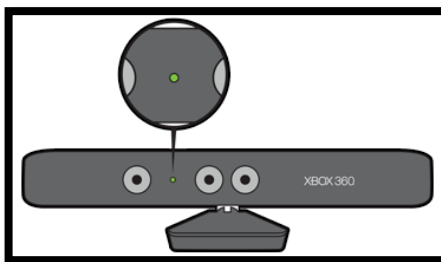
- ✓ el sensor Kinect para Xbox 360
- ✓ Windows 7 en sus versiones de 32 ó 64bit
- ✓ procesador Dual Core a 2,66 Ghz o superior
- ✓ 2 GB de RAM (recomendados 4 GB)
- ✓ Visual Studio 2010 Express
- ✓ Microsoft .NET Framework 4.0

### **b.3. Componentes Hardware [8]**

El sensor Kinect posee:

- ✓ Cámara RGB
- ✓ Sensor de profundidad
- ✓ Sensor CMOS de infrarrojos
- ✓ Matriz de múltiples micrófonos.
- ✓ Luz Led
- ✓ Acelerómetro de tres ejes
- ✓ Motor

Y en especial un procesador especializado Prime Sensor que capta el entorno en 3 dimensiones y esas captaciones las traduce a una imagen sincronizada.



**Figura 1.7 Componente Hardware de la luz led**

**Fuente: [9]**



**Figura 1.8 Componentes de Hardware del Kinect**

#### **b.4 Especificaciones**

Entre las especificaciones técnicas que definen a Kinect, se tiene:

Generales:

- ✓ Cámara de movimiento VGA color: resolución de 640x480 píxeles a 30 cuadros por segundo (fps)
- ✓ Cámara de profundidad: resolución de 640x480 píxeles a 30 fps
- ✓ Matriz de cuatro micrófonos
- ✓ Campo de vista Campo de vista horizontal: 57 grados
- ✓ Campo de vista vertical: 43 grados

- ✓ Intervalo de inclinación física:  $\pm 27$  grados
- ✓ Intervalo del sensor de profundidad: 1,2 m - 3,5 m
- ✓ Sistema de seguimiento esquelético, capacidad de seguir hasta seis personas, incluidos dos jugadores activos
- ✓ Capacidad de seguir 20 articulaciones por usuario.
- ✓ Un sistema de cancelación de eco que mejora la entrada de la voz
- ✓ Reconocimiento de voz.

#### DETALLADOS POR SECCION:

##### **Sensores**

- ✓ Lentes de color y sensación de profundidad.
- ✓ Micrófono multi-arreglo.
- ✓ Ajuste de sensor con su motor de inclinación.
- ✓ Totalmente compatible con las consolas existentes de Xbox 360.

##### **Campo De Visión**

- ✓ Campo de visión horizontal: 57 grados.
- ✓ Campo de visión vertical: 43 grados.
- ✓ rango de inclinación física:  $\pm 27$  grados.
- ✓ Rango de profundidad del sensor: 1,2 - 3,5 metros.

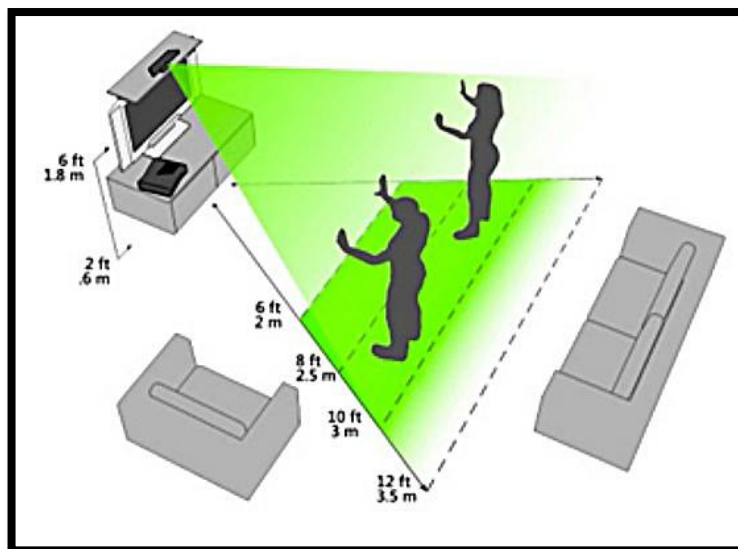
##### **Data Streams (Flujo De Datos)**

- ✓ 320 x 240 a 16 bits de profundidad @ 30fps.
- ✓ 640 x 480 32-bit de color @30fps.
- ✓ Audio de 16-bit @ 16 kHz.

##### **Sistema De Seguimiento**

- ✓ Rastrea hasta 6 personas, incluyendo 2 jugadores activos
- ✓ Rastrea 20 articulaciones por jugador activo
- ✓ Capacidad para mapear jugadores activos en Live Avatars
- ✓ Sistema de audio.

### c. Funcionamiento

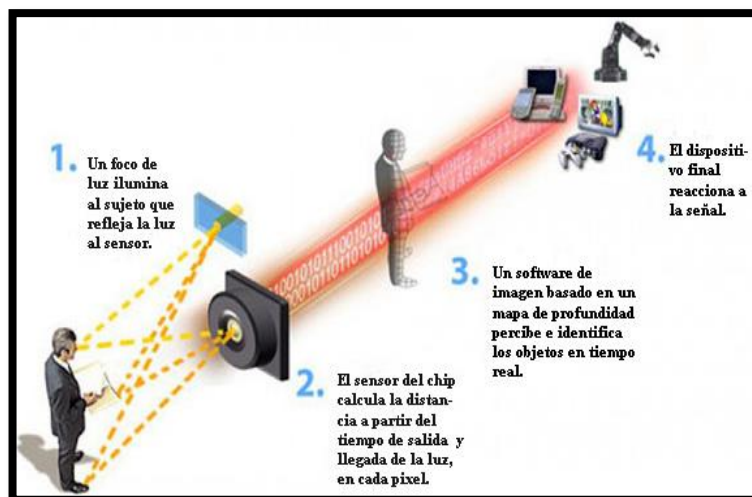


**Figura 1.9 Funcionamiento del Sensor de movimiento en Kinect**

**Fuente: [11]**

Es un periférico para videojuegos que prescinde de mandos o controles gracias a su sensor de movimiento, creado por Microsoft y previsto para ser utilizado por Windows 8, El sensor de movimiento Kinect para Xbox 360, en un principio fue conocido por el code name Project Natal. Está basado en una cámara periférica que se conecta a la videoconsola Xbox 360 reconociendo los gestos del usuario o jugador tales como rostro, voz, movimientos del cuerpo y objetos estáticos dentro del campo visual del Kinect. [10]

El sensor de Kinect es una barra horizontal conectado a un pivote, diseñado para estar en una posición longitudinal. El dispositivo tiene una cámara RGB, sensor de profundidad y un micrófono multi-array bidireccional que conjuntamente capturan el movimiento de los cuerpos en 3D, además de brindar reconocimiento facial y acepta comandos de voz. Fue creado para competir con las consolas de última generación con sensores de movimiento para videojuegos multijugador como Wii Remote y Wii MotionPlus así como PlayStation Move, de las consolas Nintendo Wii y Playstation 3.



**Figura 1.10 Etapas del funcionamiento del sensor Kinect**

**Fuente: [11]**

La cámara Kinect sirve para el reconocimiento de la imagen, Tiene dos funcionalidades principales y esenciales, genera un mapa 3D que tiene para su campo visual y reconoce humanos en movimiento con un esquema en escala de grises y de diferentes segmentos de las articulaciones.



**Figura 1.11 Intensidad de imagen y profundidad codificada en escala de grises para un retrato**

**Fuente: [11]**

En la figura 1.11 el color de fondo es muy similar al color de la persona, y es complicado para el computador segmentar desde el fondo utilizando la



intensidad de imagen, pero se realiza usando escala de grises que miden la profundidad en la imagen.

Programas de software antiguos utilizaban las diferencias en color y textura para distinguir los objetos del fondo. La cámara transmite luz invisible para nosotros, cercana en el espectro a los infrarrojos y puede conocer el tiempo que tarda la luz en volver al sensor tras reflejarse en los objetos. El sensor actúa como un sonar, la operación no es teóricamente complicada, si se conoce el tiempo de cada salida y llegada de la luz tras reflejarse en un objeto, sabiendo la velocidad absoluta de la luz, se puede tener la distancia a la cual se encuentra ese objeto. En un amplio campo visual con objetos, la cámara Kinect trata de reconocer a qué distancia están los objetos, distinguiendo el movimiento en tiempo real. Kinect puede llegar a distinguir la profundidad de cada objeto con diferencias de 1 Cm y su altura y anchura con diferencias de 3mm. El hardware de Kinect está compuesto por la cámara y el proyector de luz infrarroja, añadido al firmware y a un procesador que utiliza algoritmos para procesar las imágenes tridimensionales.

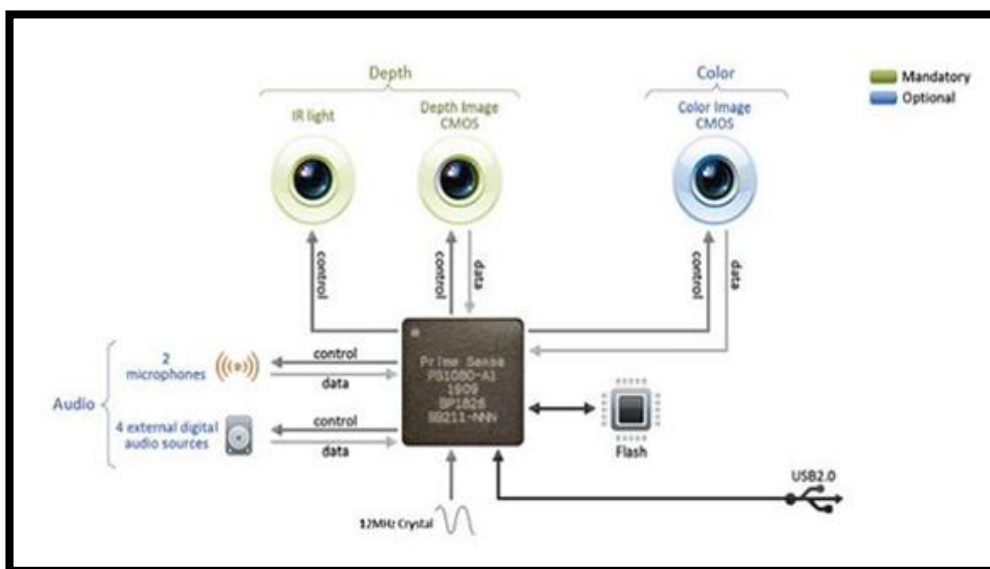
El procesador es capaz de interpretar los movimientos que se registran en los objetos capturados por la cámara de Kinect en eventos con significado que aparecen en pantalla. Los movimientos buscados por el algoritmo son contextualizados, si nos encontramos en un juego como Kinect Aventures, donde una balsa desciende por la corriente del río, si este juego requiere movimientos como agacharse o tumbarse, entonces se buscará la identificación de estos movimientos en tiempo real para producir eventos en pantalla. Si el usuario navega por el menú con interfaz gráfica de Netflix entonces se buscarán movimientos con la mano horizontal y vertical que serán registrados en los fenómenos de pantalla.

### **c.1 Chip Primesense 1080**

Este chip captura su entorno en tres dimensiones y lo transforma a imágenes sincronizadas en 3D, sirve también para reconstituir una captura de movimiento 3D de la escena frente al Kinect.

El Chipp primesense fue desarrollado por una empresa Israelita PrimeSense, este ha sido su mejor y mayor logro ya que permite una infinidad de posibilidades.

Sus dos elementos están constituidos por un sensor 3D, (proyector de infrarrojos y sensor de profundidad) y el chip PrimeSense, los cuales trabajan en conjunto.



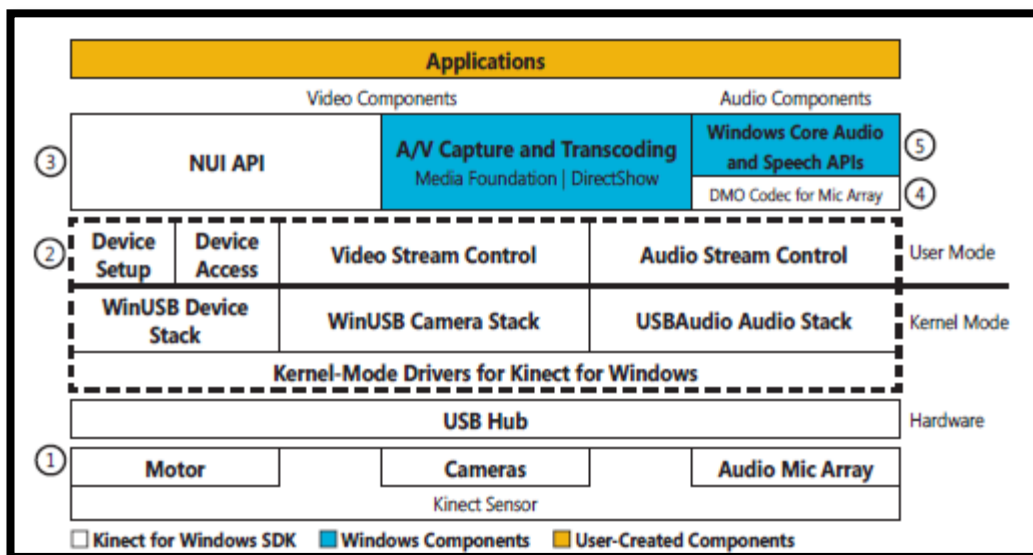
**Figura 1.12 Sistema que conforma el chip Prime Sensor**

**Fuente: [11]**

Los creadores de Kinect tienen una idea diferente con respecto a la utilización y desarrollo para los videojuegos, ya que estos no podrán ser detectados si los usuarios no se encuentran erguidos o rectos y en una zona despejada para jugar, es decir no se podrá estar sentado en un sofá, para usar Kinect se deberá estar de pie. Kinect cuenta también con un micrófono estéreo el cual sirve para chatear en las videollamadas y por supuesto para efectuar comandos de voz. Esta tecnología es diferente a la de los micrófonos integrados en la webcam o sobremesa debido a que no poseen la cancelación de ruido ya que no necesitan capturar sonidos cercanos como las de la webcam o teléfono móvil sino del habla de una o varias personas en una sala a una distancia corta.

Kinect está buscando un nuevo tipo de interfaz gráfica que según Microsoft en un futuro será lo que se imponga para ordenadores, televisión, videoconsolas de juegos cooperativos o familiares.

#### d. Arquitectura [12]



**Figura 1.13 Arquitectura Kinect**

**Fuente: [12]**

A continuación se presenta la descripción de los elementos de la figura 1.13, los números encerrados en un círculo de la figura corresponden a lo siguiente:

1. Hardware de Kinect: los componentes de hardware, incluido el sensor Kinect y el hub USB, a través del cual se conecta el sensor al equipo.
2. Controladores de Kinect de Microsoft: los controladores de Windows 7 para el sensor Kinect se instalan como parte del proceso de configuración de la versión beta de SDK.

Los controladores de Kinect de Microsoft son compatibles con:

- ✓ La matriz de micrófonos del sensor Kinect como un dispositivo de audio en modo kernel al cual se puede acceder a través de las API estándar de audio en Windows.

- ✓ Imagen de transmisión y datos de profundidad.
  - ✓ Funciones de enumeración de dispositivos que le permite a una aplicación usar más de un sensor Kinect conectados al equipo.
3. NUI API: conjunto de API que recopila datos de los sensores de imágenes y controla los dispositivos de Kinect.
4. KinectAudio DMO: El KinectAudio DMO proporciona todas las características del estándar de Windows junto con características especiales de la matriz de micrófonos.
5. API estándar de Windows 7: las API de audio, voz y medios en Windows 7, tal como se describen en el SDK de Windows y el SDK de Microsoft Speech (Guía de programación de la versión beta del SDK de Kinect para Windows).

#### **d.1 Nui Api [13]**

##### Interfaz de Usuario Natural (NUI) para Kinect para Windows

La NUI es el núcleo de la Kinect para Windows API (Interfaz de programación de aplicaciones) con la cual se ingresa los datos que provienen de los sensores los cuales son:

- ✓ Datos de imagen de color y datos de imágenes profundidad
- ✓ Los datos de audio se transmiten por el flujo de audio.

Además de las capacidades de hardware, el tiempo de ejecución de software Kinect implementa:

- ✓ El software puede reconocer y realizar un seguimiento de un cuerpo humano. El tiempo de ejecución convierte la información de profundidad en las articulaciones del esqueleto en el cuerpo humano. Esto hace posible un seguimiento de dos personas delante de la cámara.
- ✓ La integración con la API de Microsoft habla de modo que puede implementar un motor de reconocimiento de voz en su aplicación

compatible con Kinect. Esto hace que sea posible añadir los comandos de voz, como "Tracking Inicio" o "Seguimiento Stop", para su aplicación.

- ✓ Una estrecha integración con el seguimiento SDK, lo que hace posible el seguimiento de rostros humanos.

El Kinect para Windows Developer Toolkit incluye Kinect Explorer - WPF C # Muestra, que demuestran cómo utilizar la API de NUI en aplicaciones.

La Interfaz de Usuario Natural se divide en los subsistemas Data Streams, Skeletal Traking y Speach.

## **d.2 Data Streams [14]**

Si el flujo de datos se activa, Kinect puede capturar audio, color y datos de profundidad, Estos datos generan datos esqueleto. El sensor proporciona los datos para su aplicación en forma de un flujo de datos. La NUI API permite usar la programación de control y acceso. Los procesos de aplicación de cada fotograma deben ser comunicados de manera oportuna.

El flujo de datos de profundidad combina dos tipos de datos:

- ✓ Los datos de profundidad (mm).
- ✓ Datos de segmentación del usuario.

Los datos de la profundidad corresponden a la distancia en milímetros. La imagen de fondo está disponible en 3 diferentes resoluciones: 640x480 (por defecto), 320x240 y 80x60 como se especifica mediante el DepthImageFormat Enumeración. El ajuste del rango, especificado mediante el DepthRange Enumeración, determina la distancia desde el sensor en la que se reciben los valores de profundidad.

## **d.3 Segmentación De Datos**

Los procesos de ejecución Kinect pueden identificar hasta seis usuarios en un mapa de segmentación. El mapa de segmentación es un mapa de bits con valores de píxel correspondiente al índice de la persona en el campo de

visión que está más cercana a la cámara en esa posición de píxel. Datos de segmentación del usuario sólo están disponibles en la corriente del sensor de cuando el seguimiento del esqueleto está habilitado.

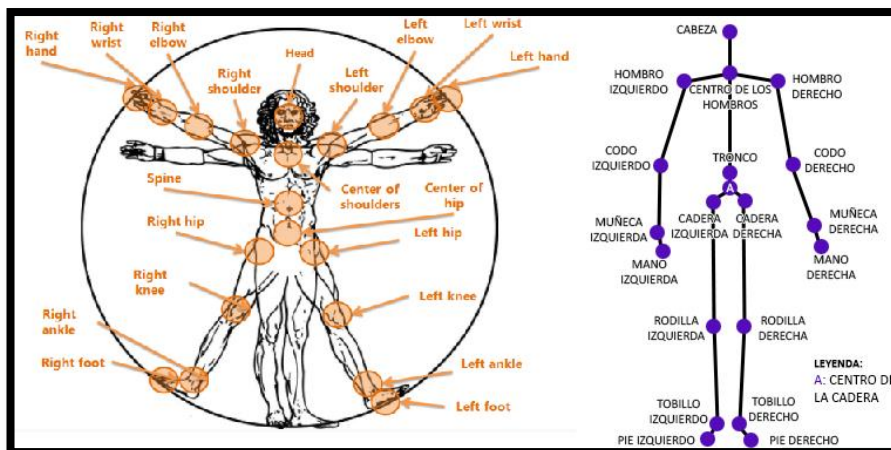
Los datos de segmentación del usuario también se conocen comúnmente como datos del índice de usuario. Aunque estos datos son un flujo lógico separado, los datos de profundidad y los datos de segmentación se fusionan en un único valor de píxel para cada cuadro. El valor "0" indica que no identifico a ninguna persona, los valores de "1" a "6" están definidos cuando identifico usuarios.

#### **d.4 Profundidad Extended Data**

Algunas aplicaciones necesitan datos de profundidad más allá del límite", es decir que datos que se encuentre "demasiado lejos" o más cerca del límite "demasiado cerca", incluso si la resolución o la precisión disminuyen. Los datos de profundidad y de segmentación pueden ser recuperados en cualquiera de los dos formatos:

- ✓ Toda la información de profundidad - Cada píxel está representado por una estructura con dos campos: una profundidad de 16 bits y un índice de usuario 16-bit. Se notifican todos los valores de profundidad detectadas, incluidos los que están fuera del intervalo de confianza. Píxeles cuya profundidad es desconocida (no pueden ser detectados) se reportan con un valor de profundidad de "0".
- ✓ Pic información de profundidad - Cada píxel está representado por un valor de 16 bits. Los primeros 13 bits contienen el valor de profundidad y los 3 bits de orden inferior contienen el índice del usuario. Cualquier valor de profundidad fuera de la gama aceptable se reemplaza con un valor especial para indicar que está demasiado cerca, demasiado lejos, o no se conoce su valor.

## d.5 Skeletal Tracking

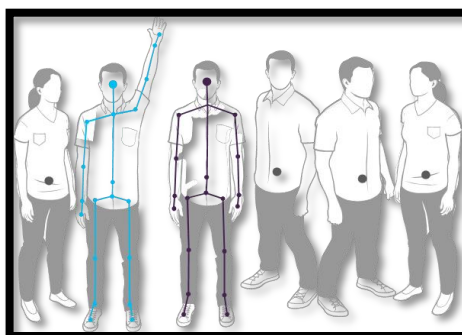


**Figura 1.14 Forma de Kinect para reconocer al Skeletal Tracking**

**Fuente: [16]**

### d.5.1 Visión de Conjunto

El Seguimiento esquelético permite a Kinect reconocer o registrar a las personas para seguir los movimientos del usuario. Usando el sensor infrarrojo (IR) de la cámara, Kinect puede reconocer hasta seis usuarios en el campo de visión del sensor. De los cuales dos pueden ser rastreados con precisión. Una aplicación puede localizar las articulaciones de los usuarios rastreados en el espacio y realizar un seguimiento de sus movimientos en el tiempo.[15]

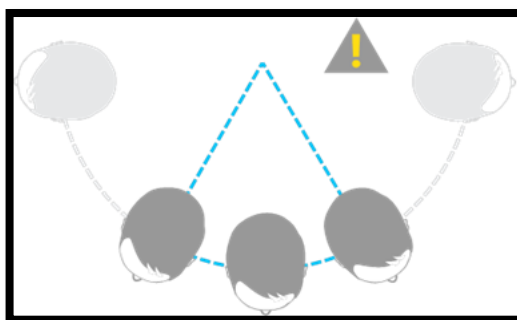


**Figura 1.15 Kinect es capaz de reconocer a seis personas y realizar seguimiento de dos de ellas**

**Fuente: [17]**

Es importante aclarar que el Seguimiento esquelético está optimizado para reconocer a los usuarios en dos posiciones de pie o sentado, al encontrarse al frente o de lado del Kinect, es posible que existan algunas dificultades como por ejemplo que el usuario no sea visible para el sensor.

Para que el sensor Kinect reconozca al usuario, este debe estar en frente del sensor, garantizando que el sensor observe la cabeza y la parte superior del cuerpo.



**Figura 1.16 El Esqueleto de seguimiento está diseñado para reconocer a los usuarios en frente del sensor**

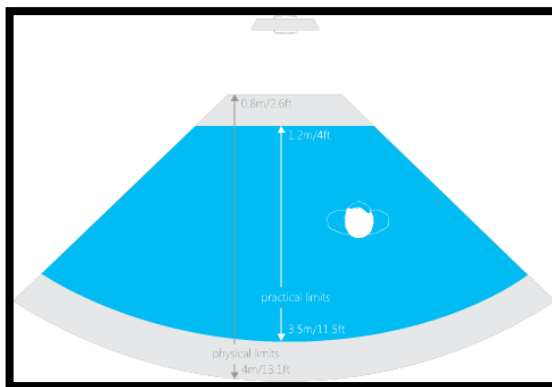
**Fuente: [17]**

#### **d.5.2 Campo de Visión**

El campo de visión de los usuarios que establece Kinect está determinado por los ajustes de la cámara de infrarrojos, que se establecen con el DepthRange Enumeración.

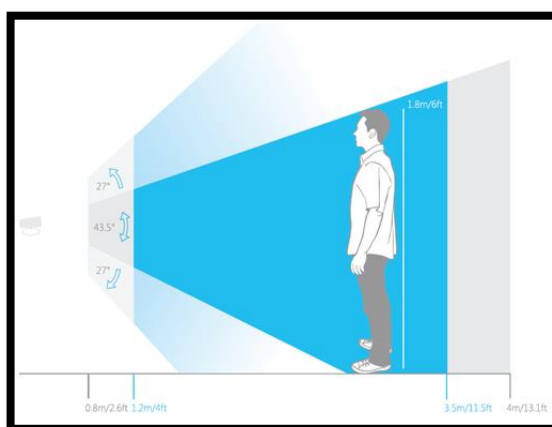
El rango del sensor Kinect puede observar al usuario de pie entre 0,8 metros (2,6 pies) y 4,0 metros (13,1 pies) de distancia, lo que sugiere un rango práctico de 1.2 a 3.5 metros. Para el rango más cercano, Kinect puede ver usuarios de pie entre 0,4 metros (1,3 pies) y 3,0 metros (9,8 pies); que tiene un intervalo práctico del 0,8 a 2,5 metros.





**Figura 1.17 Rango del campo de visión horizontal del sensor Kinect**

**Fuente: [17]**



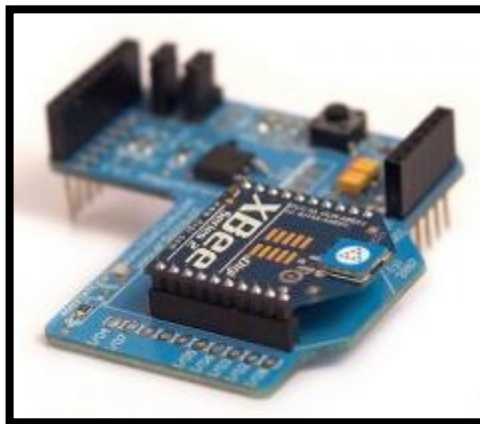
**Figura 1.18 Rango del campo de visión vertical del sensor Kinect**

**Fuente: [17]**

El emisor de infrarrojos de un sensor de Kinect proyecta un patrón de luz infrarroja. Este patrón de luz se utiliza para calcular la profundidad de las personas en el campo de visión que permite el reconocimiento de los diferentes usuarios y sus partes del cuerpo. Si se utiliza más de un sensor Kinect para iluminar la zona de destino, existe una reducción en la exactitud y precisión de seguimiento esquelético por la interferencia con las fuentes de luz infrarroja. Para disminuir la posibilidad de interferencia, es recomendable que no exista más de un sensor Kinect (o fuente de luz infrarroja) que apunte a un campo de visión donde se realiza el seguimiento del esqueleto.

### 1.4.2 Arduino Xbee Shield [18]

Los módulos XBee utilizan el protocolo IEEE 802.15.4 conocido también como ZigBee. El cual es creado para implementar redes de sensores (red tipo mesh) con propiedades de recuperación y bajo consumo de energía.



**Figura 1.19 Modulo Xbee**

**Fuente: [19]**



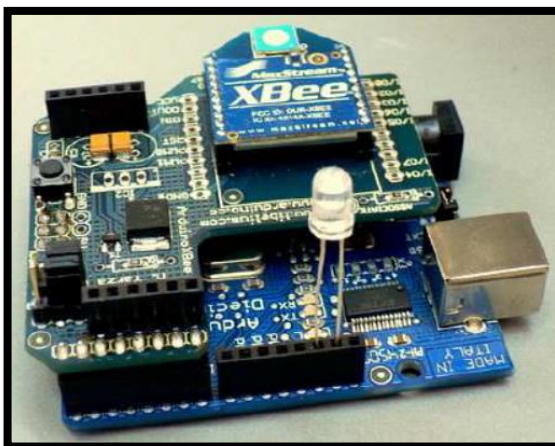
**Figura 1.20 Tarjeta Arduino Uno**

**Fuente: [20]**

La figura 1.20 muestra la tarjeta Arduino Uno y en la figura 1.19 su módulo Xbee la unión de estas tarjetas conforma la tarjeta Arduino Xbee Shield la cual es utilizada en este proyecto.

Las distancias a las cuales los módulos XBee de MaxStream permiten enlaces seriales de señales TTL son de 30 metros en interiores, 100 metros en exteriores con línea de vista y hasta 1.5 Km con los módulos Pro.

El XBee shield para Arduino permite la comunicación de la tarjeta Arduino de forma inalámbrica usando Zigbee, la cual fue desarrollada en colaboración de Libelium. Para esta conexión es necesario tener dos placas Arduino con Xbee shield comunicándose una con otra sin configuración, manejando sólo los comandos estándares de Arduino.



**Figura 1.21 Xbee Shield montado sobre la placa Arduino**

**Fuente: [21]**

La figura 1.21 es el montaje del módulo Xbee y la Tarjeta Arduino Uno.

#### **a. Carga Del Programa A La Placa Arduino Uno**

Para cargar un programa a la placa Arduino con XBee shield, es necesario poner los dos jumpers del shield en la posición USB o quitarlos completamente. Posteriormente se debe cargar un programa con normalidad desde el Arduino IDE. En este caso, se carga el programa Comunicación | Physical Pixel a una de las placas. El cual ordena a la placa encender el LED conectado al pin 13 cuando recibe 'H' por el puerto serie y lo apaga cuando recibe 'L'. Se puede probar conectando la placa con el monitor de puerto serie de Arduino (debe estar configurado a 9600 baudios), escribiendo H y INTRO (o pinchando en "send") el LED se deberá encender. Al enviar L el LED deberá apagarse. Si no ocurre nada, es posible que se tenga un Arduino que no tiene el LED conectado al pin 13 montado en la

placa (es recomendable observar el índice de placas), en este caso se tendrá que conectar su propio led.

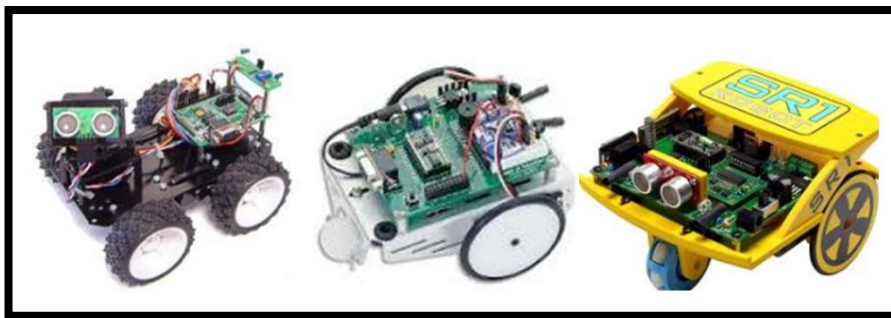
Una vez que se ha cargado el programa Physical Pixel y comprobado que funciona, desconectar el primer Arduino del ordenador. Conecta los jumpers en la posición XBee (en la posición más alejada del borde de la placa). Posteriormente se necesita cargar otro programa a la otra placa. Asegurándose de que sus jumpers están en la posición USB. Cuando se haya cargado el programa, se puede comprobar que funciona con el monitor de puerto serie de Arduino. Se observa los H's y L's llegando cada segundo. Se debe apagar el monitor de puerto serie y desconectar la placa. Cambiar los jumpers a la posición XBee. Después se debe conectar las dos placas al ordenador, después de unos segundos, observar el LED de la primera placa encenderse y apagarse cada segundo (el LED de la placa Arduino, no el LED de la placa XBee shield, que proporciona información sobre el estado del módulo XBee). Si funciona significa que las placas Arduino se están comunicando inalámbricamente.

Se puede usar cualquiera de los comandos serie estándar de Arduino con el XBee shield. Con los jumpers del shield en la posición XBee, los comandos print y println enviarán los datos al XBee shield y al puerto USB (o lo que es lo mismo a otros XBee shields y al ordenador al mismo tiempo). Con esta configuración, sin embargo, la placa sólo recibirá datos del XBee shield, no de la conexión USB (es necesario cambiar la posición de los jumpers para permitir que la placa reciba datos del ordenador).

El módulo XBee del shield está configurado para trabajar a 9600 baudios por defecto, a menos que se cambie la configuración, se deberá asegurar de llamar a la función Serial.begin() con el parámetro 9600 (Serial.begin(9600);) en el programa.

Para permitir que el ordenador se comunique directamente con el XBee shield, Se debe conectar a una placa Arduino sin microcontrolador y colocar los jumpers en la posición USB, entonces se podrá enviar y recibir datos desde el módulo XBee utilizando cualquier terminal de puerto serie. Esto permite ver los datos que el módulo está recibiendo de otros XBee shields.

### 1.4.3 Robot Móvil



**Figura 1.22 Modelos de Robot Móvil**

**Fuente: [22]**

En la figura 1.22 se pueden observar los diferentes tipos de robot móviles que existen dentro del mercado o han sido creados por el humano.

Los robots móviles son máquinas preparadas para desplazarse por el suelo, lugares con obstáculos, sin actuación del ser humano. Los robots móviles se han desarrollado grandemente en los últimos años de investigación. Esto se debe en gran parte a la reducción de costos en el hardware indispensable para su construcción y a la nueva concepción industrial, ya que dentro de sus ventajas este es un sistema más económico, fácil de implementar y de diseño simple. Requiere de una reconfiguración de la secuencia de acciones necesarias para una producción variada. Los Robots móviles son capaces de navegación genérica en un medio dado a partir de órdenes de alto nivel; a la secuencia de tales órdenes se la suele llamar el plan de la misión. [22]



**Figura 1.23 Sistema autónomo robot móvil más brazo robótico**

**Fuente: [22]**

En la figura 1.23 se tiene un robot móvil que posee un sistema autónomo, es decir, realiza su propia gestión del tráfico y el cual será guiado en forma remota.

La implementación experimental de pequeños robots móviles en laboratorios universitarios y en diversos campos de la robótica y a nivel de aficionado o microbots está admitiendo que presente un tipo de investigación que engloba los aspectos de conexión senso-motora (aspecto más fundamental de la Robótica) desde un punto de vista diferente a aproximaciones anteriores y que conlleva también un cambio de visión en la concepción clásica de la Inteligencia Artificial, la cual se aborda intentando construir vida artificial.

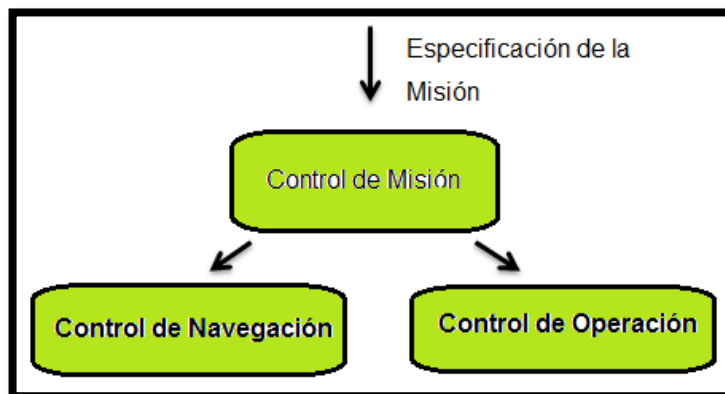
Existen varias diferencias entre los robots manipuladores industriales y los móviles. Para que un robot manipulador sea útil y capaz de evolucionar en el espacio 3D, este debe tener varios grados de movilidad, es decir grados de libertad, sin embargo, un robot móvil con solamente dos grados de libertad se encuentra apto para realizar actividades interesantes sobre una superficie de trabajo. Los robots manipuladores generalmente trabajan en espacios altamente estructurados y requieren de un sistema de percepción mucho menos sofisticado.

**a. Misión, Navegación y Operación**

El robot móvil se caracteriza por ejecutar diferentes movimientos de navegación y por llevar a cabo una interacción con distintos elementos de su entorno de trabajo (operación) que conllevan el cumplimiento de una serie de objetivos impuestos según cierta especificación que será establecida por el operador. La misión de un robot móvil se define como “la realización conjunta de una serie de objetivos de navegación y operación”.

Un robot móvil debe tener una arquitectura que coordine los distintos elementos de a bordo (sistema sensorial, control de movimiento y operación) de forma correcta y eficaz para la realización de una misión. El diseño de

esta arquitectura depende mucho de su aplicación en particular, pero un esquema básico de los principales módulos que la componen y la interacción que tienen estos son presentados en la figura 1.24.



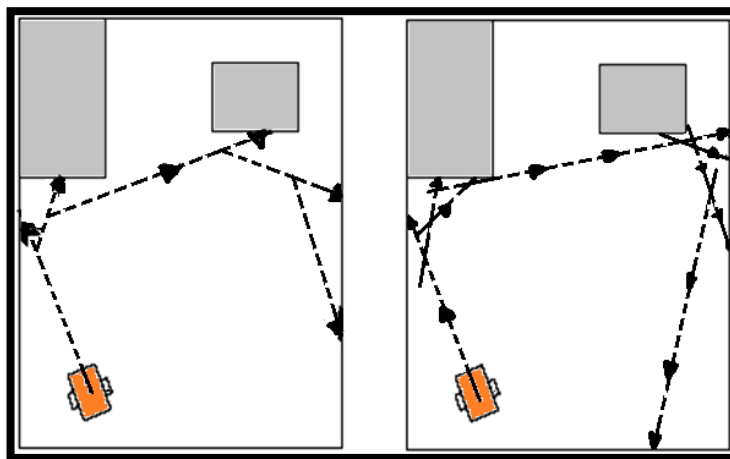
**Figura 1.24 Esquema básico de la arquitectura de un robot móvil para realizar una misión**

#### **b. Tecnologías de Construcción Y Conexión Senso-Motora**

Un robot móvil, como otros tipos de robot, puede considerarse como un sistema (una colección de sensores, actuadores, y elementos computacionales) organizados de tal modo que exhiban una acción inteligente en respuesta a ciertos estímulos. Tal sistema no tiene por qué ser complicado, de hecho, algunos de los robots móviles más sencillos responden al siguiente esquema: Están constituidos por dos motores que hacen girar sendas ruedas independientemente, no llevan procesador, todo su hardware consiste en los drivers de potencia de los motores, y en dos contadores programables que comienzan a contar al recibir una señal externa. Durante el tiempo que dure su cuenta, hasta un valor predefinido, generan una señal que indica al motor que se mueva en sentido opuesto. El resto del tiempo cada motor se mueve hacia adelante.

El robot, al mantener un movimiento de los motores a una misma velocidad y sentido, este avanza. Al chocar el robot móvil con un obstáculo, ambos motores invierten su sentido, y el robot retrocede. Pero el contador de uno de ellos está ajustado a un tiempo menor que el del otro, con lo cual al invertirse ambos giran en distinto sentido durante un instante, Hace que el

robot gire sobre su propio eje, con lo que es posible que ya no apunte hacia el obstáculo. Posteriormente ambos motores vuelven a girar hacia adelante, y se continúa el avance en línea recta. Si el tiempo (ángulo) de giro no hubiese sido suficiente, el robot volvería a chocar con el obstáculo, y repetiría la misma maniobra. La sección de la parte izquierda de la figura 1.25, indica esta situación.



**Figura 1.25 Odometría del Robot Móvil**

Si a uno de los motores se le hiciera girar un poco más lento en el sentido derecho, tendríamos que el robot retrocedería describiendo un arco. Esto haría que siguiese aproximadamente las paredes, como se observa en el recuadro derecho de la figura 1.25. En ese caso, el observador externo asignaría un propósito al robot, cuando en realidad el "programa" no lo establece explícitamente. De hecho, ni siquiera se puede decir que haya un programa en sentido tradicional: todo esto se puede implantar con circuitería analógica, aunque sea mejor hacerlo digitalmente, siempre que se conserve el espíritu de simplicidad. Este ejemplo sirve para ilustrar una conexión directa y de bajo nivel entre percepción y acción, pero esta no es la única manera. También se puede (usando sensores más sofisticados) tratar de localizar aspectos físicos externos conocidos por un mapa prealmacenado, realizar una planificación del camino por los medios antes vistos, y seguirlo usando referencias establecidas u odometría.

Respecto a la tecnología usada, debemos referirnos a los sensores y a los actuadores.



Entre los **sensores** más utilizados se encuentran:

- ✓ **Fotoresistencias o fototransistores**, que se usan para implementar fototaxias (seguimiento de fuentes de luz). Su salida se conecta a un conversor A/D, o a un simple comparador, dependiendo del uso que se quiera hacer (si importa el valor de la señal, o sólo si ésta es superior a un umbral).
- ✓ **Sensores de proximidad por infrarrojos**: Son sensibles a radiación alrededor de los 880 nm. Existen detectores encapsulados que contienen emisor y receptor; modulan la emisión y responden sólo a ese patrón de modulación con lo que evitan interferencias de fuentes externas de infrarrojos. El hardware que necesitan es un oscilador (de cuarzo, o a estable) para el emisor, y un conversor A/D o comparador para el receptor.
- ✓ **Sensores piroeléctricos**: son resistencias variables con la temperatura. Se usan para seguir fuentes de calor.
- ✓ **Sensores de contacto por doblez**: constan de un eje metálico con una capa de pintura conductora que varía su resistencia al doblarse. Se conectan a un conversor A/D.
- ✓ **Microinterruptores de choque** (bumpers): se usan con una palanca que los activa al chocar el robot con algún obstáculo. Se conectan directamente a entradas digitales del microcontrolador del robot.
- ✓ **Sonares**: El modelo más usado es el Polaroid TM. El hardware que usan es un contador, para saber el tiempo transcurrido entre la ida y la vuelta del impulso ultrasónico.
- ✓ **Codificadores ópticos**: normalmente de tipo incremental, se instalan en todas o algunas de las ruedas tanto en el eje de giro como en el de guiado. Requieren un hardware específico para la cuenta de pulsos, aunque ésta se pueda hacer también por software, conectado las señales de cada canal a puertos de entrada, y manteniendo un proceso dedicado a monitorizarlos.
- ✓ **Giróscopos**: son análogos a los usados en los sistemas de navegación inercial de los aviones, pero algo más simples. Son raramente usados por su alto precio. Existen versiones electrónicas baratas basadas en sensores de

estado sólido que sólo miden la velocidad de giro, pero no la orientación absoluta.

- ✓ **Inclinómetros:** se basan en un codificador óptico en posición vertical con un péndulo colgado de él, o bien en una gota de mercurio sobre un platillo horizontal con contactos repartidos regularmente alrededor de ella.
- ✓ **Brújulas:** indican la orientación absoluta usando el campo magnético terrestre. No son muy usadas, aunque en lugares exteriores dan medidas aceptables, en interiores y sobre todo con campos magnéticos provocados por la circuitería o maquinaria circundante no son fiables.
- ✓ **Cámaras de TV:** se suelen usar modelos en miniatura, de tipo CCD. Tienen los inconvenientes de requerir un hardware más complicado (una placa digital de imagen) y generar un volumen de información difícilmente tratable en tiempo real sin hardware específico.

En cuanto a los actuadores, se suelen usar siempre motores eléctricos de CC, por su facilidad de control. Se conectan a engranajes reductores para disminuir la velocidad y aumentar la potencia. Si la corriente que necesitan no es muy alta (robots pequeños no muy pesados con dos motores pueden consumir de 0.5 a 2 A por motor), existen reguladores encapsulados que pueden proporcionarla, los cuales se controlan por modulación por ancho de pulso (PWM).

Los lazos de realimentación para el control de las ruedas se suelen realizar por software, el cual lee los registros asociados a los codificadores, y envía una señal digital que luego se convierte en analógica y activa los dispositivos de potencia. El control es, por supuesto, siempre discreto.

Los sistemas de locomoción son variados. El modelo más común consiste en usar dos ruedas motrices independientes con sus ejes alineados perpendicularmente a la dirección de avance. La forma del robot suele ser compacta, mejor circular, para ganar maniobrabilidad.

La alimentación es un punto muy problemático. Hasta ahora, las baterías que podían dar suficiente corriente eran muy pesadas; hoy día existen modelos recargables de Ni-Cd más ligeros.

## CAPÍTULO II

### 2. DISEÑO DEL SISTEMA

#### 2.1 Estudio De Los Códigos Fuente SDK De Kinect

El SDK o Kit de Desarrollo de Kinect principalmente fue orientado a la investigación académica, pero con el pasar de los días tuvo una gran acogida por programadoras particulares; es por esta razón que en este capítulo se procederá a estudiar los códigos fuentes del *coding4fun* de la versión beta 2.

Entre los códigos fuente a estudiar tenemos:

- ✓ Skeletal Tracking
- ✓ Camera fundamental
- ✓ Depth Data

##### 2.1.1 Skeletal Tracking

Los códigos fuentes utilizados en el proyecto son:

##### **Microsoft.Research.Kinect.Nui:**

Al momento de crear un proyecto, esta librería es esencial para empezar a programar y utilizar el paquete de controles que tiene el sensor kinect; ya que permite crear la interface de comunicación entre el sensor kinect y Visual Studio C#.

##### **Coding4Fun.Kinect.Wpf:**

Este código fuente hace referencia a un toolkit el cual contiene métodos para trabajar con el sensor de profundidad y los datos del Skeletal tracking, debido a esta herramienta se ha logrado simplificar el arduo trabajo en la programación y mejorar la rapidez en el procesamiento de imágenes.

**System.IO.Ports:**

Esta librería permite habilitar los puertos de entrada y salida para comunicación serie de Visual Studio C#.

**Runtime:**

Este código permite inicializar el sensor Kinect; al cual se le puede dar un nombre de variable pública, para este proyecto se lo ha denominado como Kinect\_GI.

**Kinect\_GI.Initialize:**

Código para dar inicio al uso del Skeletal Tracking de Kinect.

**RuntimeOptions:**

Instrucción mediante la cual podemos acceder a todas las funciones del Kinect; para hacer uso del Skeletal tracking se selecciona la función (UseSkeletalTracking).

**Kinect\_GI.SkeletonFrameReady:**

Permite crear un evento en el cual se pueda adquirir los valores entregados por el Skeletal Tracking.

**New EventHandler<SkeletonFrameReadyEventArgs>():**

Este código permite dar nombre al método, para este proyecto se lo llamará Kinect\_esqueleto.

**Void Kinect \_esqueleto:**

En este método se cargarán todos los eventos generados por el SkeletonFrameReady.

**SkeletalTrackingState.Tracked:**

Por medio de este código se obtendrán datos emitidos por el Skeletal Tracking siempre y cuando el usuario este en movimiento.

**Posicion\_eclipse(eclipse\_cabeza,gi\_esqueletor.Joints[JointID.Head])**

Enlaza los datos de la cabeza del usuario a un objeto *ellipse*, los mismos que serán enviados a un evento *posicion\_eclipse*, en la cual constan las escalas y la posición en la que el cuerpo humano se debe mover.

**var escala\_puntos\_eclipse = puntos\_eclipse.ScaleTo(320, 368):**

La resolución de la imagen utilizada en este proyecto es de 320\*368 pixeles, rango en el cual los distintos puntos captados por el movimiento del cuerpo puedan moverse sin ningún conflicto y a la vez sin alterar el diseño de la pantalla de presentación.

**Canvas.SetLeft-Canvas.SetTop:**

Estos métodos contienen los valores de la posición tanto en el eje X como en el eje Y, los mismo que han sido usados en la pantalla de presentación de acuerdo a la posición de los puntos entregados por el Skeletal tracking.

**2.1.2 Camera Fundamental****kinect\_GI.VideoFrameReady:**

En este evento se procede activar la cámara RGB de Kinect.

**kinect\_GI.VideoStream.Open:**

Por medio de esta instrucción se da apertura al uso de la cámara de video, la cual va enviar el flujo de datos captados a la aplicación realizada, también se puede especificar la resolución a trabajar, así como el tipo de flujo de imagen y para mayor percepción se debe especificar el tipo de imagen como "Color".

**RuntimeOptions:**

Instrucción mediante la cual podemos acceder a todas las funciones del kinect; para hacer uso de la cámara se selecciona la función (UseColor).

**PlanarImage:**

Código el cual permite crear una variable en la que se va a cargar los datos capturados por la cámara.

**BitmapSource.Create():**

Permite especificar el ancho y alto de la imagen en la cual se va a poder visualizar los datos capturados por parte de la cámara, de la misma forma se puede escoger el formato de pixeles el cual para este proyecto es "Bgr32" que es el formato de pixeles que se van almacenar; también podemos asignar colores los diferentes elementos percibidos por la cámara, mediante la mezcla del ancho de la imagen "imageData.width" y la paleta de pixeles de la imagen "imageData.BytesPerPixel"

**2.1.3 Depth Data****RuntimeOptions:**

Instrucción mediante la cual podemos acceder a todas las funciones del kinect permitiendo definir cuantas personas se encuentran en el rango del sensor de profundidad, para este proyecto la instrucción que capta los datos de profundidad es "UseDepthAndPlayerIndex".

**Kinect\_GI.DepthFrameReady:**

Por medio de esta instrucción se adquieren los datos de profundidad del Skeletal tracking.

**2.2 Diseño Del Software****2.2.1. Diseño De La Aplicación Demostrativa En C#**

Para el desarrollo de la aplicación demostrativa se empleó el análisis y estudio de los códigos fuentes SDK de Kinect (Camera Fundamental, Depth

Data, Skeletal Tracking), los mismos que permitirán controlar al robot móvil en lugares remotos.

El robot móvil tiene 4 grados de libertad ya que posee 4 servomotores, los mismos que van a permitir controlar los movimientos tanto en forma horizontal cuando el usuario ordene el barrido horizontal con la mano derecha (eje x), el movimiento vertical, cuando el usuario se acerque o aleje del sensor Kinect utilizando el brazo izquierdo (eje y), y la rotación cuando el usuario haga un desplazamiento con la mano derecha (eje z). Adicionalmente, al proyecto se agregó una aplicación extra que consiste en que el tercer y cuarto servomotor asumen el movimiento de la cabeza y al ser acoplada la cámara inalámbrica permitirá mediante la pantalla visualizar de forma exacta la ubicación del robot móvil, la cual utilizará los valores de movimiento del usuario; ya que al mover la cabeza en posición horizontal, el tercer servomotor producirá un desplazamiento en el eje X en un rango de movimiento de 0-180 grados; de misma forma al mover la cabeza en posición vertical el cuarto servomotor se desplazara en el eje Y en un rango de 0-180 grados.

La interfaz y los datos que se adquieren de los distintos movimientos que realice el usuario al mover el robot móvil, podrán ser visualizados en la aplicación desarrollada en Visual Studio 2010 en el lenguaje de programación C#.

Los datos son enviados por medio de la comunicación serial de C# a la placa Arduino, la misma que se va a encargar de recibir esos datos y enviarlos al ZigBee emisor para de esta manera conectarnos inalámbricamente al ZigBee receptor para dar el movimiento a los distintos servomotores.

La placa Arduino tiene conexión USB, lo que facilita conectar a cualquier dispositivo que tenga conexión vía USB.

## a. Formulario Carátula (Tesis GI)

### a.1 Análisis Formulario Carátula (Tesis GI)

Formulario en el cual se realiza la presentación del trabajo práctico; detallando así el logotipo a la Institución perteneciente, el tema el cual es el “Desarrollo e implementación de un sistema para el control de un robot móvil para acceso a lugares remotos utilizando tecnología Kinect”; los autores del proyecto de graduación, los tutores designados para guiar el proyecto de graduación., además consta de un botón el cual al dar un click nos permitirá ocultar la ventana de carátula y pasar a otra ventana donde se pondrá en ejecución la aplicación demostrativa de Visual Studio C#.

### a.2 Diseño Formulario Carátula (Tesis GI)

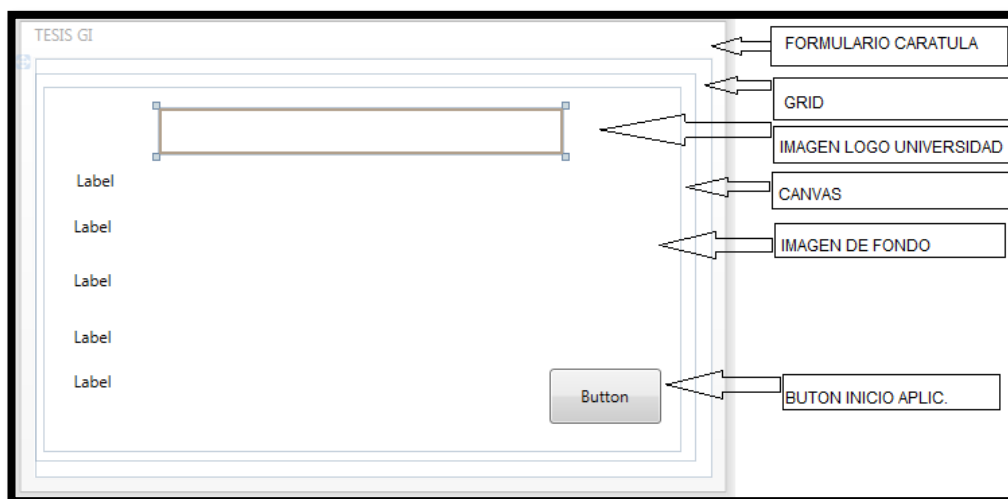


Figura 2.1 Diseño Formulario Carátula



### a.3 Resultado de Formulario Carátula



Figura 2.2 Carátula Programada en Visual C#

### b. Formulario Aplicación Demostrativa

#### b.1 Análisis Formulario Aplicación Demostrativa

Formulario en el cual se realiza la presentación de la aplicación demostrativa; va a contener dos canvas y una imagen la cual va a permitir visualizar el cambio de dirección del robot móvil mediante diferentes imágenes presentadas, además consta de etiquetas o "label" las cuales detallan la Carrera a la que se pertenece, así como la posición en la que está ubicado el usuario, indicando por medio de mensajes si la posición es la correcta o si tiene que moverse hacia atrás, o hacia adelante.

En el primer canvas se va a poder visualizar las distintas utilidades que contiene el SDK de Kinect, en la imagen1 que está acoplada al canvas1 se va a desplegar el conjunto del Skeletal Tracking y la cámara de video RGB; también en la imagen2 que está contenida en el canvas2, se va a poder desplegar el conjunto del Skeletal tracking, además se tiene en cada imagen objetos eclipse los cuales se van a enlazar con los puntos del Skeletal Tracking, utilizando estos "Joint" se procede a programar y asociar los datos

entregados por el Kinect con el lenguaje de Programación C#; para después enviarlos por el protocolo serial a la tarjeta Arduino, en donde mediante el protocolo inalámbrico ZIGBEE, se enviarán los datos al receptor del robot móvil para poder operarlo mediante el movimiento de las manos, se tendrá un voltaje a cada servomotor el cual me permitirá controlar y acceder a lugares remotos con el robot. De la misma manera, para tener la visibilidad del lugar en donde se encuentra, se ha acoplado una cámara inalámbrica la cual enviará datos a un televisor dando la ubicación exacta del robot. El diseño del formulario se lo puede visualizar en la figura 2.3.

## b.2 Diseño Formulario Aplicación Demostrativa

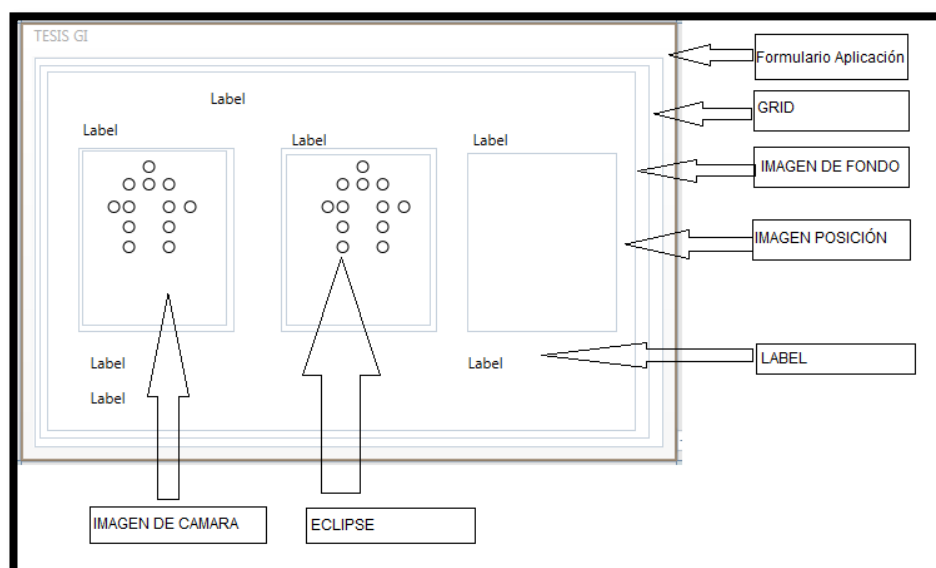


Figura 2.3 Diseño Formulario Aplicación

### b.3 Resultado Formulario Aplicación Demostrativa

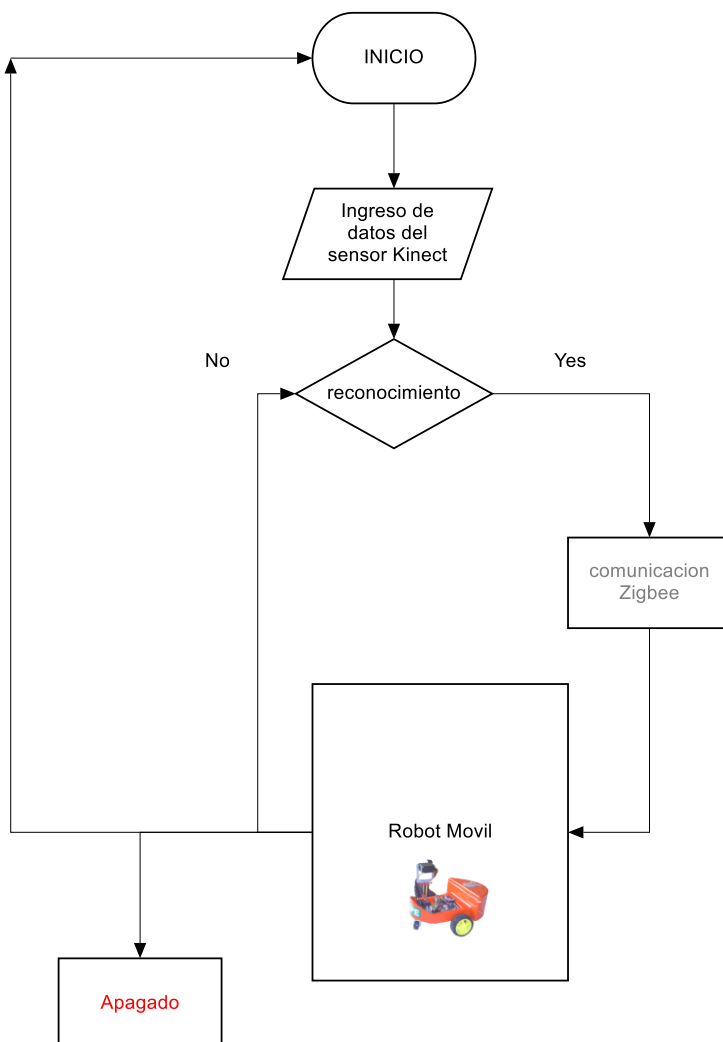


Figura 2.4 Resultado Formulario Aplicación

## 2.2.2 Programación En Visual Studio C#

El presente proyecto fue desarrollado en el lenguaje de programación C#

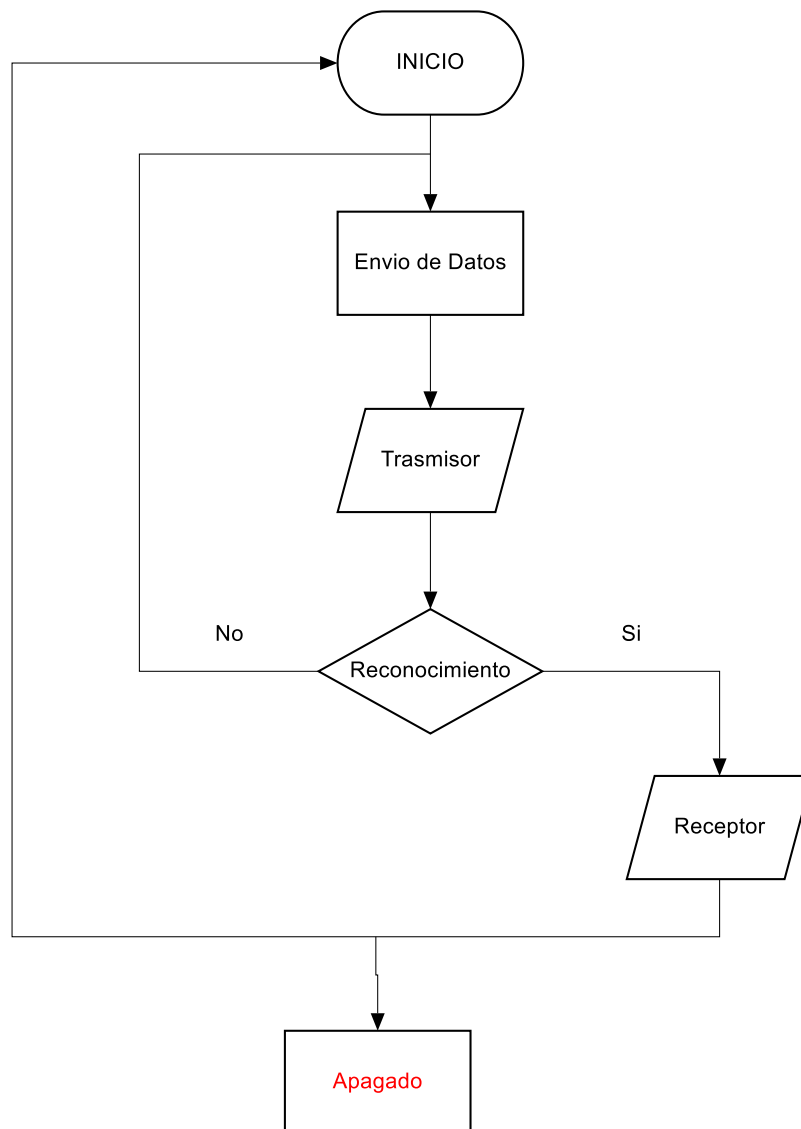
### a. Diagrama de Flujo De La Programación General



En el diagrama de flujo se puede observar que al iniciar el programa, se toman los datos enviados por el sensor Kinect, en el cual, si se reconoce al cuerpo humano, se tiene un correcto flujo de datos, dando paso a la comunicación ZigBee la cual es configurada mediante la tarjeta Arduino Uno. Luego de desarrollar la lógica de programación se envían los datos desde el emisor hasta el receptor para que de esta manera se tenga el control sobre el robot móvil. Caso contrario entraría en bucle hasta reconocer al usuario.

### b. Diagrama de flujo de la Codificación programa Arduino UnoR3

En el diagrama de flujo de Arduino se puede ver el transmisor y receptor en el envío de datos.

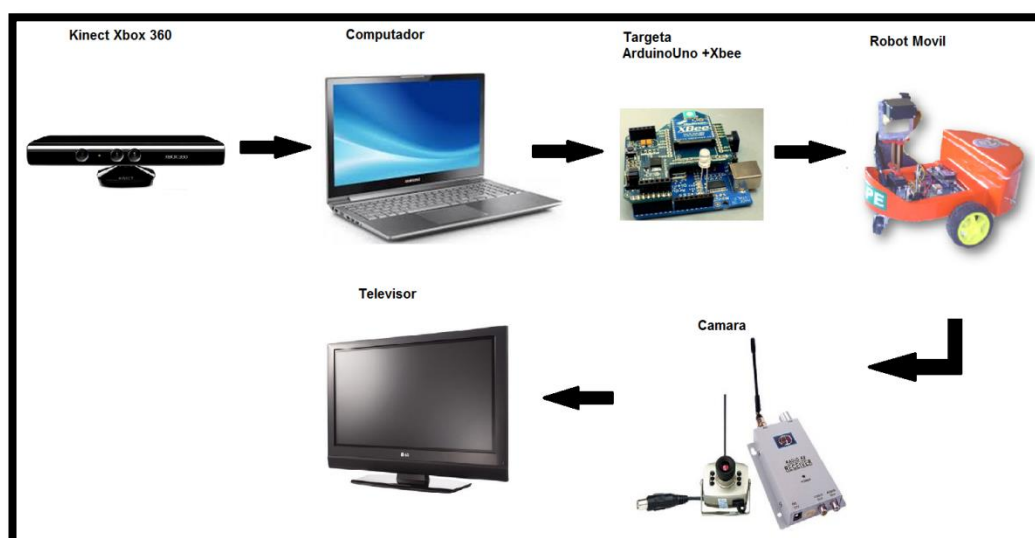


En el diagrama de flujo correspondiente al programa denominado Arduino UnoR3, se presenta el inicio y un apagado para su finalización. Al empezar el envío de datos que va hacia el transmisor, lo cual está configurado en la tarjeta Arduino Uno, se programa para que exista la comunicación con el receptor. Al existir la comunicación, los datos son enviados hacia el receptor que está ubicado en el robot móvil, con lo cual podrá usuario realizar el control correspondiente.

### c. Hardware Utilizado

El hardware utilizado consta varias partes primero, el Kinect que es el encargado de captar los movimientos del cuerpo humano, tales como los movimientos de los dos brazos y la cabeza necesarios para el control. En un computador con el SDK de Kinect previamente instalado mediante la aplicación que realiza en WPF C# reconocerá los datos enviados por el Kinect, los cuales se visualizara en la pantalla del programa , La tarjeta Arduino UNO que está conectada al computador vía USB, permitirá transmitir los datos adquiridos desde el sensor Kinect hacia el robot móvil, el cual cuenta con el receptor de Arduino los cuales están comunicados inalámbricamente con su respectivo circuito y programación en el robot móvil.

En la figura 2.5 podemos observar de una mejor manera como está desarrollado el hardware del proyecto, del Kinect se envían los datos hacia el computador, el cual mediante el programa de Visual C y la comunicación con la tarjeta Arduino Uno son enviados del transmisor al receptor que se encuentra en el robot móvil. En el robot móvil está instalada una cámara de baja resolución la cual muestra en donde se encuentran el robot, y su señal es enviada para una mejor visualización a un televisor.



**Figura 2.5 Hardware Utilizado**

### 2.3 Diagrama de Conexiones

Los diagramas fueron diseñados en Fritzing que es un software libre para diseño electrónico, que permite simular el funcionamiento de La Tarjeta Arduino Uno, ZigBee y los servomotores.



**Figura 2.6 Logo del Software Libre Fritzing**

Antes de realizar las conexiones se debe verificar los pines de la tarjeta Arduino Uno, tanto como los disponibles y los que se van a utilizar.

Para este proyecto se utilizan cuatro servomotores, por lo que se usará las salidas 3, 5, 6 y 9 para controlar el robot móvil, dependiendo de los datos entregados inalámbricamente por el emisor.

En el diagrama que se presenta en la Figura 2.7 se puede visualizar que consta de dos transistores NPN (2n3904), los cuales permitirán amplificar la corriente para los servomotores. Es posible observar el diagrama de conexiones del robot móvil realizado en el programa Fritzing, donde fácilmente se puede observar todos los elementos utilizados para este diagrama.

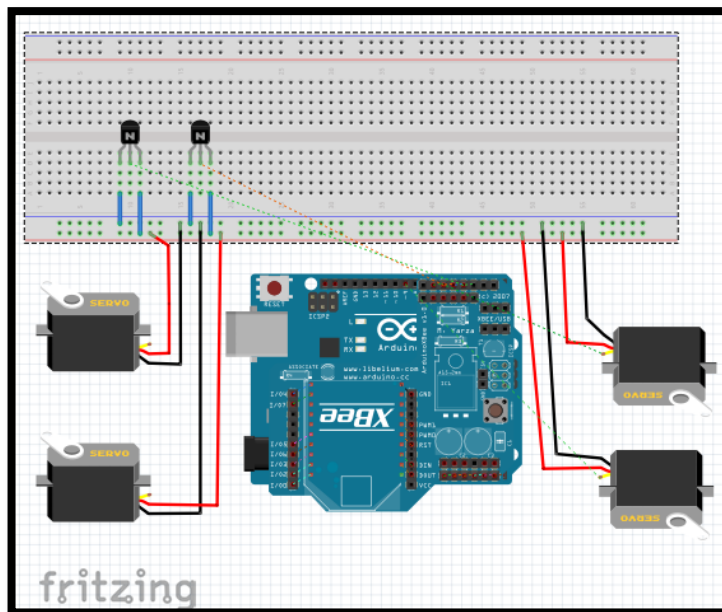


Figura 2.7 Diagrama de conexiones del robot móvil

### 2.3.1 Diagrama Esquemático

Mediante este diagrama podemos visualizar las conexiones de los servomotores hacia la placa Arduino con su respectivo shield para ZigBee. La figura 2.8 presenta el diagrama esquemático, el cual es directamente realizado por el programa Fritzing al realizar el diagrama de conexiones.

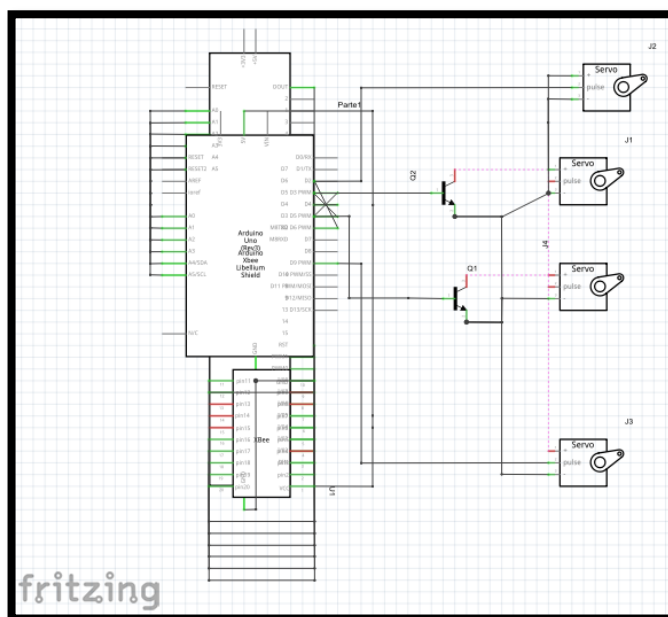


Figura 2.8 Diagrama Esquemático del robot móvil



### 2.3.2 Diseño de Circuito Impreso (PCB)

En las figuras 2.9 y 2.10 se puede verificar el circuito impreso diseñado para las conexiones entre los servomotores y la placa Arduino Uno, su uso es muy sencillo ya que consta de borneras para facilitar la conexión.

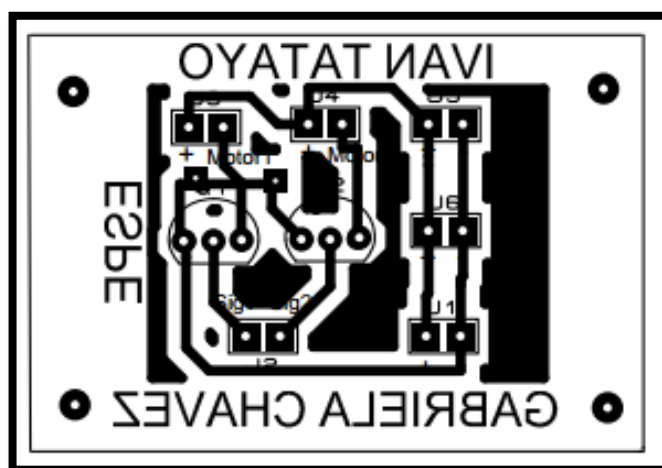


Figura 2.9 Diseño PCB vista inferior

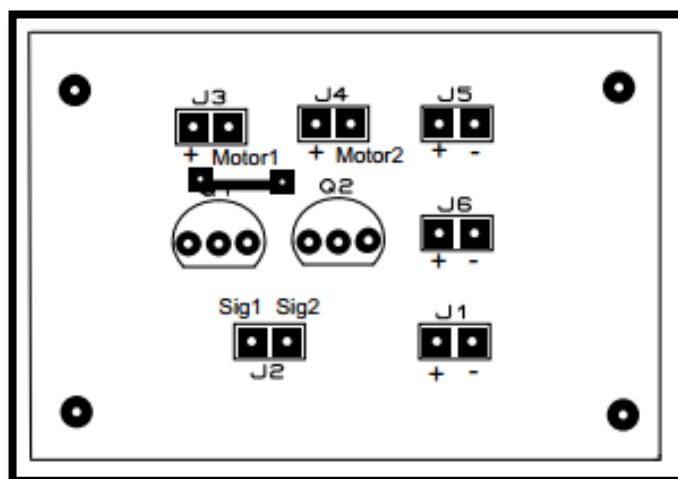


Figura 2.10 Diseño PCB vista superior

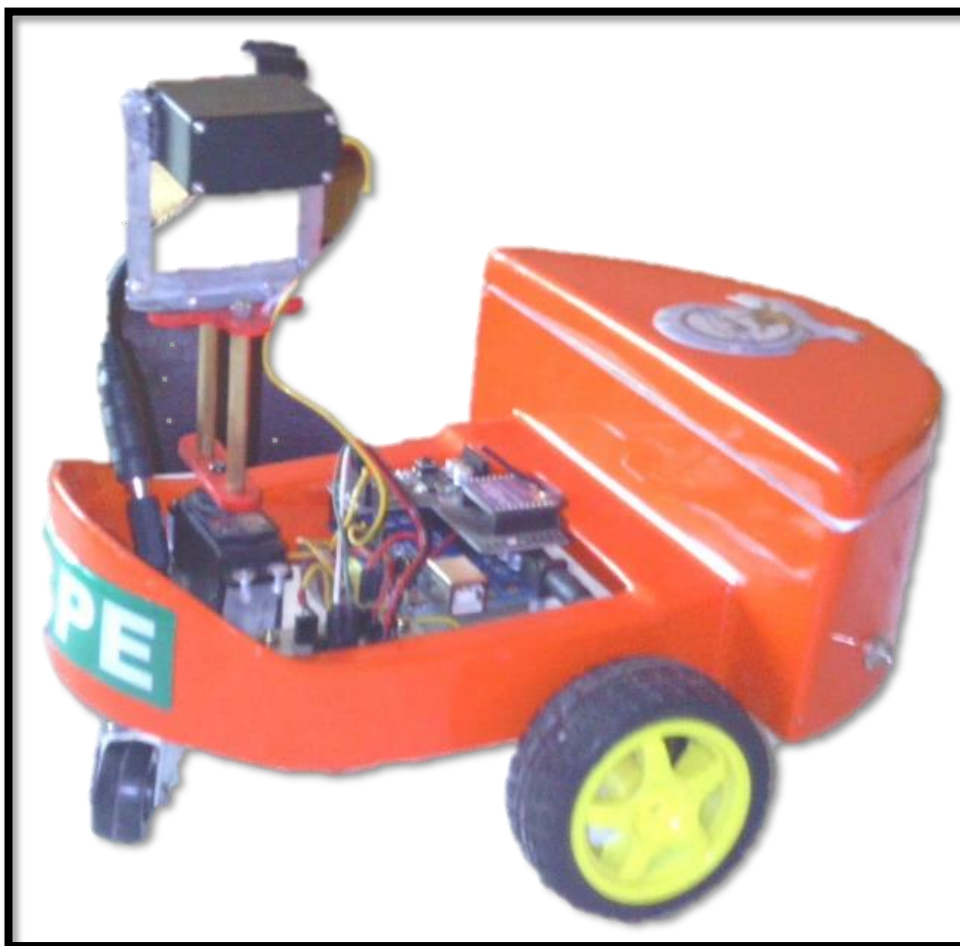
### 2.3.3 Diseño Robot Móvil

Para la aplicación demostrativa, se procedió a construir un prototipo de robot móvil, el cual se moverá de acuerdo al movimiento realizado por el usuario con sus brazos y cabeza. Tiene cuatro grados de libertad, el

movimiento de la mano derecha sobre el eje x, permite que el robot móvil gire de derecha a izquierda en un rango de 360 grados; así como el movimiento de la mano izquierda sobre el eje z, permite que el robot móvil se desplace hacia adelante o simplemente frene.

Adicionalmente los servomotores tercero y cuarto son controlados por el movimiento de la cabeza tanto en el eje X como en el eje Y respectivamente.

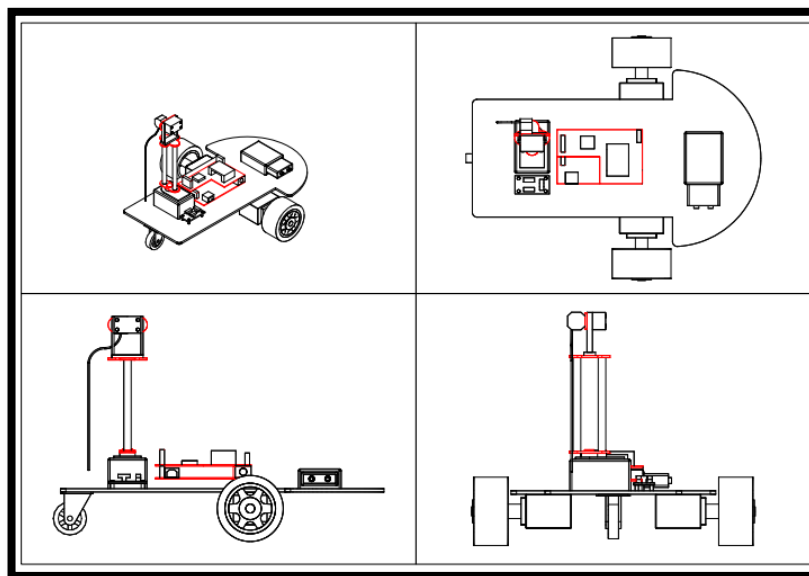
La estructura del robot está construida en acrílico, los servomotores se encuentran ubicados cada uno en una llanta diferente, consta también de una “rueda loca” que permite soportar el peso de la parte posterior. Cada dispositivo es acoplado al robot de una manera sencilla y rápida debido a que toda su estructura es desarmable.



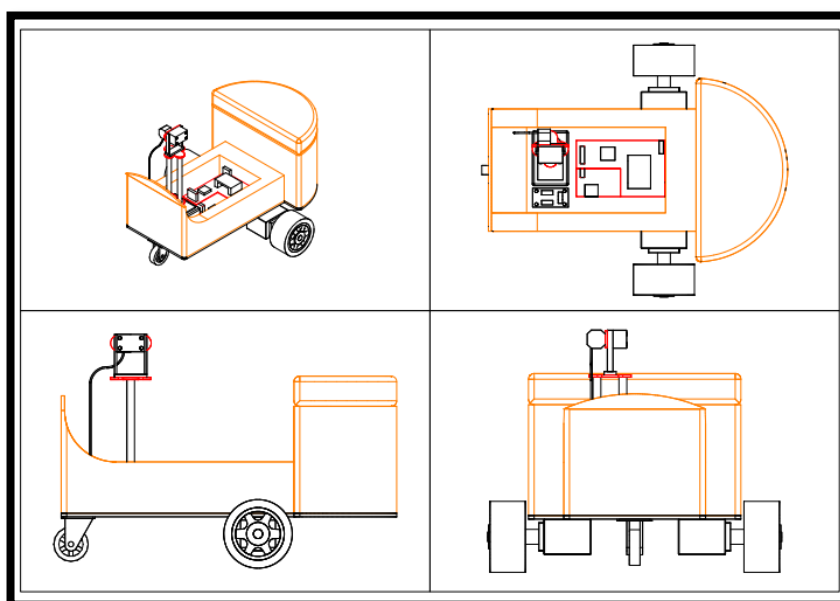
**Figura 2.11 Estructura final del Robot Móvil**

A continuación se presentan las diferentes vistas del robot móvil, las cuales fueron diseñadas en Auto CAD.

En la figura 2.12 se puede observar las vistas del robot móvil, tales como vista de lado superior inclinado, vista de lado superior, vista de lado y vista de frente. La figura 2.12 muestra estas vistas sin la carcasa mientras que la figura 2.13 muestra las mismas vistas pero añadida la carcasa.



**Figura 2.12 Diseño del Robot Móvil Interna**



**Figura 2.13 Diseño del Robot Móvil Externa**

## CAPÍTULO III

### 3. RED INALAMBRICA ZIGBEE

Este capítulo describe la forma correcta de uso de Zigbee y su adquisición de datos, para así poder realizar la construcción del hardware y software correspondiente.

#### 3.1 Estándar Zigbee

El estándar Zigbee es una tecnología para redes inalámbrica en el área personal, con estándar de comunicación de la IEEE 802.15.4. La cual permite transmisión de datos a una frecuencia de 2.4 GHz. Tiene una tasa de datos de 250 kbps, corriente de transmisión de 35mA, batería de duración de varios años , topologías redes malladas, en reposo su corriente es de 3µA y posee aplicaciones dentro de la domótica y sensores.

El estándar de comunicación IEEE 802.15.4 define el nivel físico y el control de acceso al medio de redes inalámbricas de área personal con tasas bajas de transmisión de datos.

ZigBee es perfecto para procesos de automatización, ya que por sus propiedades puede no utilizar cableado en una red lo cual facilita su montaje, obteniendo ahorro de dinero y tiempo, que son dos puntos importantes dentro de un proyecto o trabajo.[23]

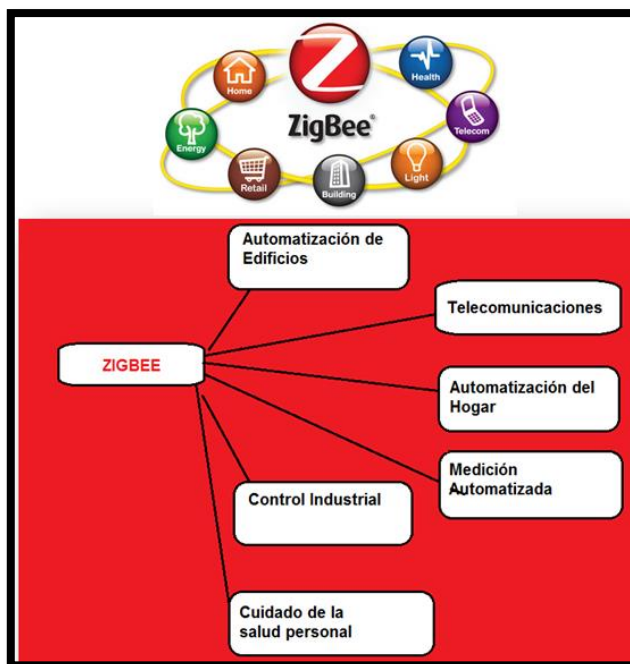
En la tabla 3.1 especifica las características con sus respectivos rangos de trabajo basado en el estándar IEEE 802.15.4.

Tabla 3.1

## Propiedades del Estándar IEEE 802.15.4.

CARACTERÍSTICAS	RANGO
Bandas de Frecuencia	868 MHz 915 MHz 2.4 GHz
Alcance	10-20 metros
Retardo(Latencia)	Menor a 15 ms
Número de Canales	868/915 MHz: 11 canales 2.4 GHz: 16 canales
Rango de Transmisión de datos	868 MHz: 20 Kb/s 915 MHz: 40 Kb/s 2.4 GHz: 250 Kb/s
Direccionamiento	Corto de 8 bits o de 64 bits IEEE
Canal de acceso	CSMA-CA CSMA-CA ranurado
Temperatura	Rango de temperatura industrial - 40°C a 85°C

## 3.2 Aplicaciones Zigbee



**Figura 3.1 Aplicaciones de Zigbee**

**Fuente: [24]**

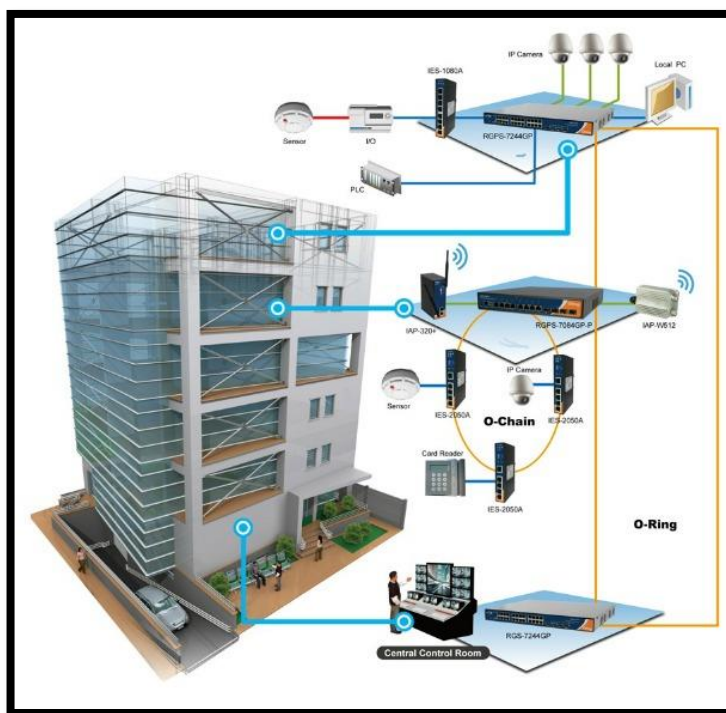
La figura 3.1 Describe las aplicaciones del estándar Zigbee es decir donde se puede aplicar, tales como la automatización de edificio y hogares, la atención médica, control industrial, medición automática y telecomunicaciones.

### 3.2.1 Diferentes Aplicaciones De Zigbee [25]

El estándar Zigbee tiene su principal aplicación cuando sus nodos se unen para trabajar juntos y configurar lo que se denomina “Red de Sensores”. Zigeo se ha hecho a medida para aplicaciones de control y su monitoreo, como la automatización de edificio (Inmótica) y hogares (Domótica), la atención médica, control industrial, medición automática y telecomunicaciones.

### a. Automatización De Edificios

Control de luces, control de acceso, control de temperatura, seguridad.



**Figura 3.2 Aplicación de Zigbee en la Automatización de un Edificio**

**Fuente: [25]**

En la figura 3.2 se puede observar un edificio que cuenta con la automatización mediante el estándar ZigBee, es visible la variedad y cantidad de equipos conectados en red ZigBee, con lo cual mediante esta red ayuda al control y al mejor desempeño.

### b. Cuidado de la Salud

Monitoreo de pacientes, control de estado físico, sin necesidad de que el personal médico interrumpa sus demás labores. El campo de la salud ofrece un gran número de oportunidades para las aplicaciones inalámbricas entre las cuales podemos destacar 3 áreas principales

- ✓ Seguimiento de enfermedades crónicas.
- ✓ Supervisión del bienestar personal.
- ✓ Personal fitness.

Los dispositivos del ZigBee Health Care Profile se subdividen en tres categorías según el tipo de seguimiento que necesite el paciente:

- ✓ Dispositivos para seguimiento episódico del paciente.
- ✓ Dispositivos para seguimiento continuo del paciente.
- ✓ Dispositivos para seguimiento del paciente con alarmas.



**Figura 3.3 Aplicación de Zigbee en el Cuidado de la Salud**

**Fuente: [25]**

La Figura 3.3 muestra los equipos que ayudan al control y monitoreo de la salud de una persona los cuales están configurados gracias al estándar ZigBee.

### **c. Control Industrial**

Rastreo de equipos, control de energía, control de procesos, control de ambiente.



**Figura 3.4 Aplicación de Zigbee en Control Industrial**

**Fuente: [25]**



En la figura 3.4 se puede observar una industria grande la cual posee un control industrial, en la parte derecha de la imagen se puede ver uno de los equipos que ayuda a la red.

#### **d. Automatización de hogar.**

Control de iluminación, control de acceso, seguridad, ventilación, irrigación de jardines.



**Figura 3.5 Aplicación de ZigBee en automatización de hogar**

**Fuente: [25]**

Es posible observar en la figura 3.5 una casa automatizada la cual es controlada mediante un equipo inteligente. En la parte derecha se visualizan los instrumentos o equipos de la casa los cuales pueden ser controlados para mayor tranquilidad del usuario dentro de su hogar.

#### **e. Medición Automática.**

Control de consumo, mediciones inteligentes, control de clima.

#### **f. Telecomunicaciones.**

Servicios de red, La figura 3.6 muestra una red mediante el estándar ZigBee.

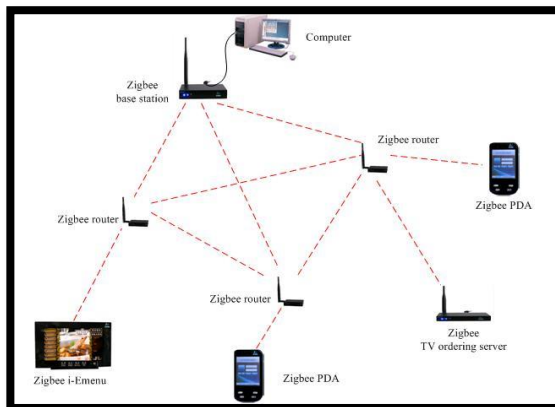


Figura 3.6 Aplicación de ZigBee en telecomunicaciones

Fuente: [25]

### 3.3 Arquitectura Zigbee

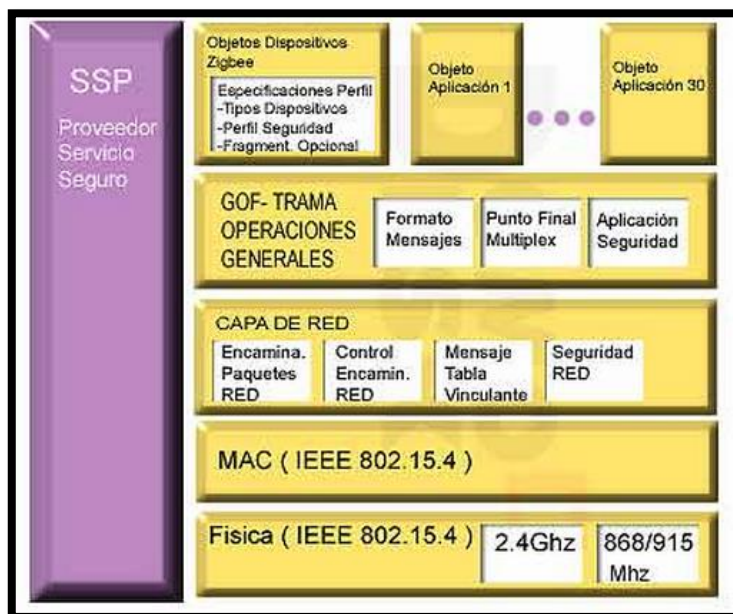


Figura 3.7 Arquitectura ZigBee

Fuente: [26]

Siguiendo el estándar del modelo de referencia OSI (Organización Internacional de Normalización), en la figura 3.7, aparece la arquitectura en capas.

La primera capa es la física y la segunda, la de acceso al medio MAC, estas son definidas por el estándar IEEE 802.15.4. Las capas superiores son

definidas por la Alianza ZigBee y corresponden a las capas de red y de aplicación, las cuales contienen los perfiles del uso, ajustes de la seguridad y la mensajería.

La función principal de la capa de red es permitir el correcto uso del subnivel MAC y entregar una interfaz adecuada para el nivel superior que corresponde a la trama de operaciones. Así también el control puede detectar así mismo la presencia de receptores, lo que posibilita la comunicación directa y la sincronización a nivel MAC.

El GOF o trama general de operaciones que se encuentra entre la capa de aplicaciones y la capa de red, cubre diversos elementos tales como modos de direccionamientos, subdireccionamiento y la descripción de dispositivos. La trama general especifica métodos, eventos y formatos de datos que son utilizados para constituir comandos.

La capa de aplicación es la más alta descrita por la especificación y, por tanto, la interfaz efectiva entre el nodo ZigBee y sus usuarios. En esta capa se encuentran la mayor parte de los componentes tanto los objetos de dispositivo ZigBee, procedimientos de control y objetos de aplicación.

### **3.4 Tipos De Dispositivo Zigbee**

ZIGBEE ALLIANCE define tres tipos de dispositivos según el papel que realizan o ejecutan en la red:

- ✓ Coordinador ZigBee
- ✓ Router ZigBee
- ✓ Terminal o Dispositivo final

**Coordinador ZigBee** es el tipo de dispositivo más completo. Existe o debe existir UNO solo, independiente de qué tipo de red se esté utilizando. Controla la red y se encarga de que los dispositivos se conecten entre ellos es decir la asociación o des-asociación de dispositivos.

**Router ZigBee** conecta dispositivos separados en la topología de la red. Es posible que trabaje como coordinador al igual que se asocia con uno en la red u otro router ZigBee.

**Terminal o Dispositivo final** Posee la funcionalidad necesaria para comunicarse con su nodo principal o maestro (el coordinador o un router), pero no puede transmitir información destinada a otros dispositivos. Este nodo puede estar dormido la mayor parte del tiempo o en estado "sleep" incrementando así la vida media de sus baterías. No es necesario gran capacidad de memoria ya que tiene requerimientos mínimos por lo cual es mucho más económico. Los dispositivos finales están siempre ubicados en los extremos de la red y es gracias a esto su nombre.

De acuerdo a su funcionabilidad es posible clasificarlos en:

- ✓ Dispositivo de funcionalidad completa (FFD)
- ✓ Dispositivo de funcionalidad reducida (RFD)

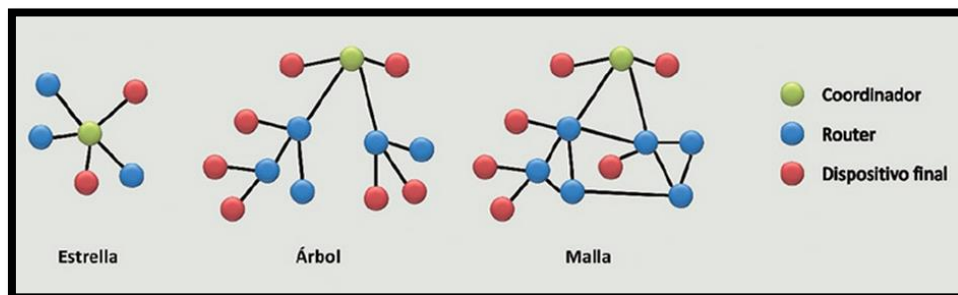
FFD: Conocido como nodo activo y se apoya en las funciones y características del estándar IEEE 802.15.4. Gracias a la memoria adicional y a su capacidad puede funcionar como coordinador o router.

RFD: Es conocido como nodo pasivo, opera con la mínima implementación del protocolo IEEE 802.15.4, a diferencia del FFD tiene capacidad y funcionalidad limitada, tiene como objetivo ser simple y de bajo precio, además solo pueden asociarse a un FFD a la vez.

Los FFDs utilizan cualquier tipo de topología y se comunican con cualquier otro dispositivo mientras que Los RFD están limitados a utilizar solo la topología estrella, por lo cual no pueden ser ni coordinadores ni ruteadores.

### 3.5 Topologías De Red

Existen 3 topologías recogidas por Zigbee: malla, árbol y estrella. Las cuales se pueden observar en la figura 3.8.



**Figura 3.8 Tipos de Topologías que se puede configurar en una red Zigbee**

Fuente: [23]

#### 3.5.1 Topología Estrella

El coordinador de la red se sitúa en el centro de la red y el resto de dispositivos pertenecientes a la red, sean FFD o RFD se comunican directamente con él.

#### 3.5.2 Topología en Árbol

El coordinador de la red se comunica con dispositivos FFDs, los cuales a su vez comunican de forma directa con los nodos de su clúster.

#### 3.5.3 Topología en Malla

Los nodos tendrán más de una conexión, con lo cual existen muchas direcciones o caminos para que lleguen los mensajes al coordinador.

#### 3.5.4 Ventajas Y Desventajas De Cada Topología

En la tabla 3.2 se muestran algunas ventajas y desventajas que existen en las diferentes topologías.

Tabla 3.2

## Ventajas y Desventajas de las topologías de Red.

TOPOLOGÍAS	VENTAJAS	DESVENTAJAS
Estrella	<ul style="list-style-type: none"> <li>✓ Baja latencia.</li> <li>✓ Topología muy robusta.</li> <li>✓ Fiabilidad muy alta.</li> <li>✓ Sencillez y rapidez en el desarrollo.</li> <li>✓ Gasto energético homogéneo.</li> </ul>	<ul style="list-style-type: none"> <li>✓ No siempre es posible desplegar una topología de este tipo.</li> <li>✓ Escalabilidad baja.</li> <li>✓ Si falla el nodo central cae toda la red.</li> </ul>
Árbol	<ul style="list-style-type: none"> <li>✓ Mantiene todas las ventajas de la topología Estrella.</li> <li>✓ Alta escalabilidad.</li> <li>✓ Menor porcentaje de colisiones.</li> </ul>	<ul style="list-style-type: none"> <li>✓ La inclusión de routers puede encarecer significativamente la solución final.</li> <li>✓ Puede caer una parte significativa de la red al caer un router.</li> <li>✓ Costoso y difícil de desarrollar el algoritmo de enrutado dinámico.</li> <li>✓ Baja fiabilidad de los algoritmos de enrutamiento.</li> </ul>
Malla	<ul style="list-style-type: none"> <li>✓ Menor coste: no es necesaria una cantidad tan grande de routers para alcanzar una gran escalabilidad.</li> <li>✓ Pueden caer uno o varios nodos que la información seguirá obteniéndose, al existir rutas alternativas.</li> </ul>	<ul style="list-style-type: none"> <li>✓ Complejidad del sistema alta.</li> <li>✓ Alta cantidad de colisiones.</li> <li>✓ Empeora la latencia de la red.</li> <li>✓ Significativas diferencias entre las pruebas en laboratorio y el despliegue real, con lo que se obtienen despliegues muy costosos.</li> </ul>

### **3.6 Seguridad**

Dentro de ZigBee y de cualquier protocolo de comunicación es importante y elemental la seguridad. ZigBee maneja un modelo de seguridad de la subcapa MAC IEEE 802.15.4, la cual establece 4 servicios como:

- ✓ Control de accesos: El dispositivo mantiene una lista de los dispositivos comprobados en la red.
- ✓ Datos Encriptados: Los cuales usan una encriptación con un código de 128 bits.
- ✓ Integración de tramas: Protegen y garantizan que los datos no puedan ser modificados.
- ✓ Secuencias de refresco: Examinan que las tramas no hayan sido reemplazadas por otras. El controlador de red verifica estas tramas para ver si son las esperadas.

#### **3.6.1 Arquitectura De Seguridad**

ZigBee en sus mecanismos de seguridad maneja claves de 128 bits. Una clave puede asociarse a una red o a un enlace. Las claves de enlace se establecen en base a una clave maestra que controla la correspondencia entre claves de enlace. Como mínimo la clave maestra inicial debe obtenerse por medios seguros ya que la seguridad de toda la red depende de ella en últimas circunstancias.

Es claro y necesario que la distribución de claves es una de las funciones de seguridad más importantes. Una red segura encarga a un dispositivo especial la distribución de claves: el denominado centro de confianza o trust center. Se puede admitir debilidades o vulnerabilidades momentáneas. Las aplicaciones que no requieran un nivel especialmente alto de seguridad utilizarán una clave enviada por el centro de confianza a través del canal inseguro transitorio.

Los niveles de seguridad de la arquitectura están divididos de la siguiente forma:

El subnivel MAC puede llevar a cabo comunicaciones fiables de un solo salto. Utiliza el nivel de seguridad indicado por los niveles superiores.

El nivel de red gestiona el ruteo, procesando los mensajes recibidos. Las tramas salientes usarán la clave de enlace correspondiente al ruteo realizado, si está disponible; en otro caso, se usará la clave de red.

El nivel de aplicación ofrece servicios de establecimiento de claves al ZDO y las aplicaciones, y es responsable de la difusión de los cambios que se produzcan en sus dispositivos a la red. También guía las peticiones de los dispositivos al centro de seguridad y propaga a todos los dispositivos las renovaciones de la clave de red.

ZDO: (ZigBee Device Object – Objetos de dispositivos ZigBee): Es el encargado tanto de definir el rol de cada uno de los dispositivos de la red como de descubrir nuevos dispositivos conectados y su papel en la red.

### **3.7 Módulo Xbee**

Es el módulo más importante, ya que es el encargado de recibir y transmitir los datos cumpliendo las normas del estándar IEEE 802.15.4. Los demás bloques ayudan de soporte para que este dispositivo trabaje correctamente. En el mercado se puede encontrar distintos fabricantes que tienen módulos que cumplen con éste estándar, entre los más importantes se destacan los de MaxStream.

XBEE-ZB. Son módulos de radio frecuencia que proporcionan conectividad inalámbrica rentable a redes malladas Zigbee. Son compatibles con otros dispositivos de Zigbee incluyendo productos de otros fabricantes.

XBEE de DIGI estos módulos se encuentran disponibles en una variedad de protocolos y frecuencias. La programación de estos dispositivos se hace directamente en el módulo eliminando la necesidad de configuración de un procesador independiente, debido a que el software inalámbrico no está aislado y las aplicaciones pueden desarrollarse sin riesgo para la seguridad y en el rendimiento del módulo.

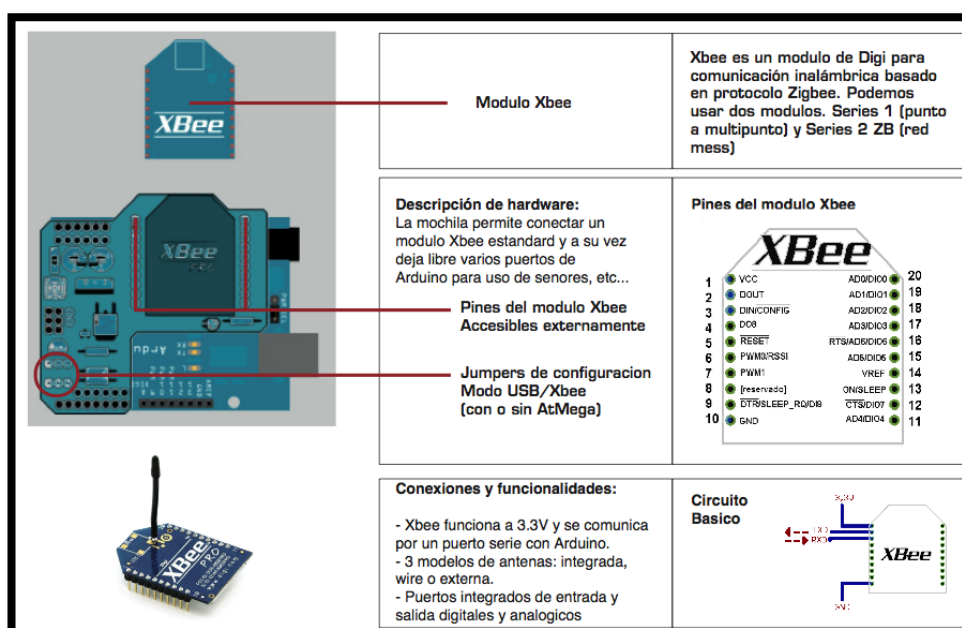
#### **3.7.1 Características**

- ✓ Interoperabilidad con otros dispositivos compatibles con ZigBee.



- ✓ Posee variedad en módulos en radiofrecuencia.
- ✓ Poseen un protocolo de red de malla que ayuda con la gestión de tráfico, actualización de Firmware y la Auto sanación de los dispositivos en caso de fallos.
- ✓ Existe varias versiones programables correspondientes a cada dispositivo XBEE con lo cual permite el desarrollo de cada aplicación personalizada con este tipo de módulos.

### 3.7.2 Conexión Placa Arduino [27]



**Figura 3.9 Esquema para la conexión del módulo ZigBee en la placa Arduino.**

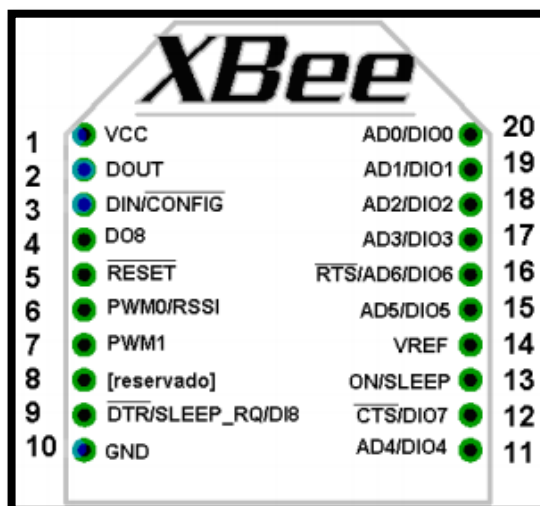
**Fuente: [28]**

Para la conexión de la placa es importante tomar en cuenta algunas especificaciones y configuraciones de la placa como se muestra en la figura 3.9.

#### a. Cable Virtual I/O

Esta opción de configuración permite crear los llamados Cables Virtuales. Se utilizan para crear un canal de comunicación de manera transparente entre los pines de un módulo y otro.

Cada pin de entrada tiene su propio pin de salida ya definido entre nodos, esto permite una forma totalmente simple de enviar información, controlar o medir de manera sencilla y rápida, sin necesidad de complicadas configuraciones.



**Figura 3.10 Diagrama de pines del módulo Xbee. Vista Superior**

**Fuente: [28]**

En la figura 3.10 se puede observar el esquemático de los pines del módulo Xbee. El envío es totalmente transparente para el usuario, ya que el paquete recibido puede o no ser entregado por el pin DOUT (pin 3), lo que permite utilizar ese pin para el envío de otro tipo de información e incluso seguir recibiendo desde otros módulos.

Para poder crear un cable virtual, es necesario tener en cuenta los pines de entrada/salida o I/O, que deben estar asociados entre sí, es decir, vienen en pares. Por lo cual sólo se pueden crear cables entre estos pares de pines de distintos módulos.

Un ejemplo que muestra esta especificación es que en una entrada digital DI5, sale por una salida digital DO5 y una entrada analógica ADC 0, sale por una salida analógica PWM 0. Algunas entradas y salidas tienen el mismo pin asignado, pero en otros como los analógicos, éstas se encuentran en diferentes pines.

## **b. Hardware**

El hardware son los componentes electrónicos y físicos de la placa Arduino. Entre ellos vamos a utilizar los siguientes:

### **b.1 Microcontrolador**

El microcontrolador es el dispositivo electrónico programable de la placa Arduino, es decir, que ejecuta un programa que previamente hemos escrito sus instrucciones para que realice una tarea de manera autónoma.

El microcontrolador es del fabricante Atmel y en la placa Arduino UNO y otras tiene el modelo Atmel328. El microcontrolador dispone de unos pines de entrada/salida que se encargan de comunicarse con el exterior. En las patillas de entrada podemos conectar sensores para recibir datos del exterior, y en las patillas de salida podemos conectar actuadores para mandar órdenes e interactuar con el entorno físico.

### **b.2 E/S Digitales**

Los pines de entrada/salida digitales son los que trabajan con valores binarios: "1" corresponde a 5V y "0" corresponde a 0V.

La placa Arduino tiene 14 pines configurables como entrada o salida digital que operan a 0 o 5 voltios. Cada pin puede proporcionar o recibir como máximo de corriente 40 mA. Se puede ampliar 6 pines digitales más utilizando los pines de las entradas analógicas como digitales.

### **b.3 Entradas Analógicas**

Los pines de entrada analógica son 6 que utiliza un convertidor analógico/digital de 10 bits. Por defecto miden de 0V hasta 5V.

La resolución de los 10 bits del convertidor A/D quiere decir que en los programas ante un valor de 0V en la entrada analógica se lee un valor de 0 y si en la entrada analógica se tiene 5V se lee un valor de 1023, que corresponde a 2 elevado a 10 menos 1.

#### **b.4 Salidas PWM**

Los pines de la placa Arduino 3, 5, 6, 9, 10 y 11 proporcionan una señal PWM (Pulse Width Modulation), señal modulada por anchura de pulso.

Una señal PWM es una señal periódica cuadrada con una amplitud de 5V a la cual podemos controlar su ciclo de trabajo para tener a la salida una señal casi continua (señal analógica) entre 0V y 5V. Esta variación de la señal continua es el resultado del valor medio de la señal periódica al variar su ciclo de trabajo.

#### **b.5 Comunicaciones**

La placa Arduino tiene diferentes maneras de comunicarse con otros dispositivos, siendo la más utilizada la comunicación serie a través del conector USB.

La conexión entre la placa Arduino y el PC utilizando el puerto USB crea una conexión virtual serie asociada a un puerto de comunicaciones del PC que permite cargar los programas con el IDE. También se puede utilizar esta conexión serie para recibir y enviar datos a los programas que se ejecutan en la placa Arduino por medio del monitor serie del entorno de programación.

#### **c. Software**

El software es la parte lógica de la placa Arduino con ella se escribe, verifica y carga en la memoria del microcontrolador el programa que quiere ejecutar.

Los programas se deben cargar en la placa Arduino se escriben y verifican en una aplicación informática que se instala en la PC denominada IDE (Integrated Development environment) de Arduino. Este entorno o interfaz de programación es multiplataforma es decir es compatible con windows, Linux y mac , se comunica con la placa Arduino a través de un puerto USB. Una vez cargado el software en la memoria del microcontrolador y los componentes electrónicos conectados a la placa Arduino.

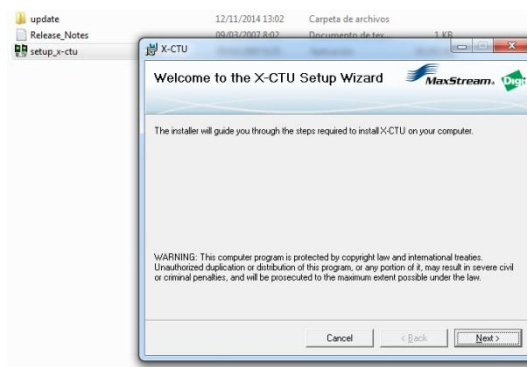
Es posible ejecutarlo de manera autónoma, desconectando el cable USB del PC, si se conecta una fuente de alimentación externa de pilas o ejecutarlo con el cable USB conectado al PC sin necesidad de alimentación externa.

### 3.7.3 Red Zigbee

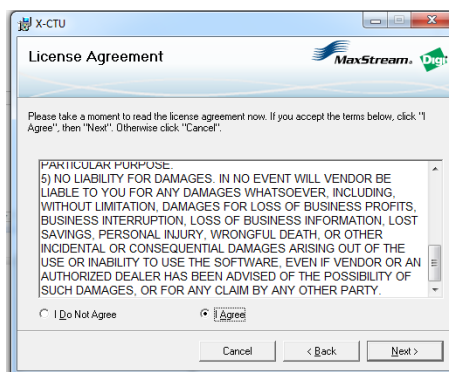
Para realizar la red Zigbee primero es necesario configurar los módulos, y cambiar los Firmware de los módulos XBee, para ello es necesario instalar X-CTU, con lo cual es posible usar como terminal serie para mandar y recibir datos por el módulo desde el PC.

#### a. Instalacion Y Configuracion X-Ctu

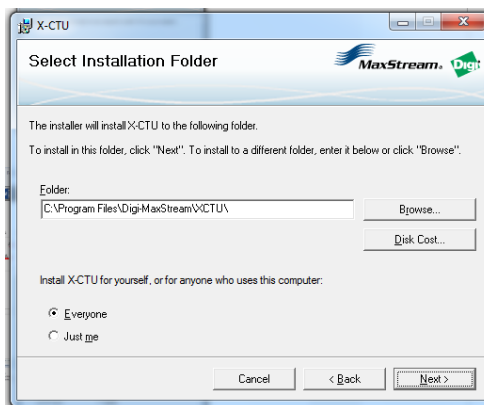
Dentro de la configuración de X-CTU que es sencilla, se debe tomar en cuenta ciertos requerimientos y dar “next” a las condiciones, sin problemas se debe instalar el programa. De la Figura 3.11 a la Figura 3.16 Se puede visualizar los pasos de la instalación.



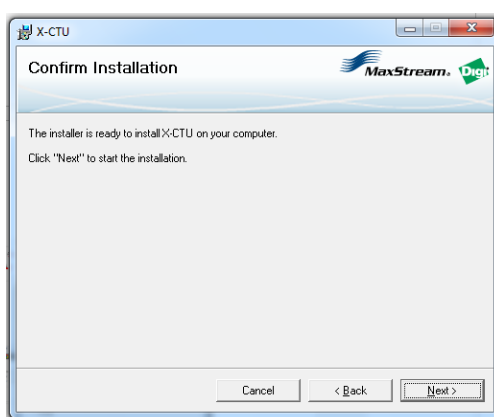
**Figura 3.11 Instalación del programa X-CTU**



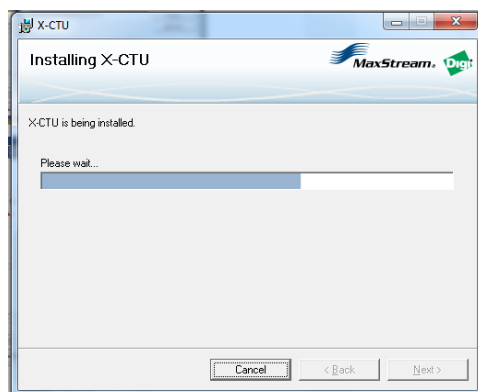
**Figura 3.12 Instalación del programa X-CTU**



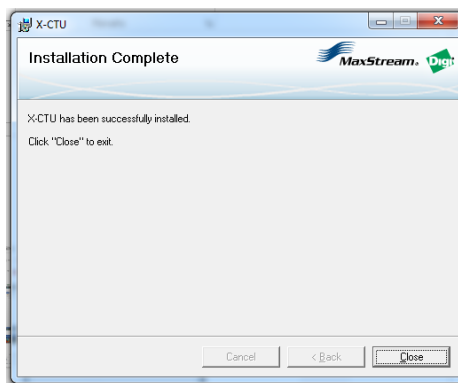
**Figura 3.13** Instalación del programa X-CTU



**Figura 3.14** Instalación del programa X-CTU

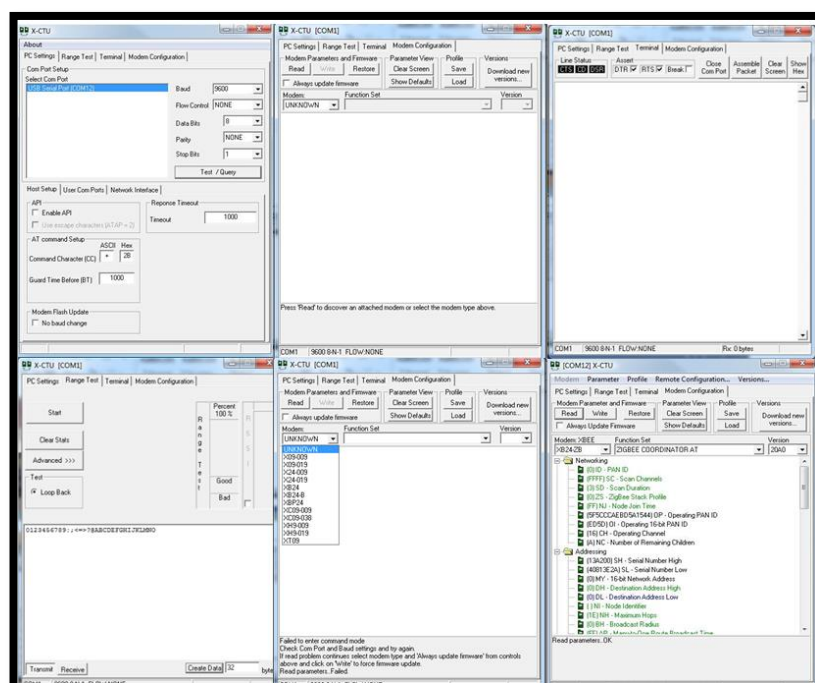


**Figura 3.15** Instalación del programa X-CTU



**Figura 3.16 Instalación del programa X-CTU**

En el programa se puede visualizar las diferentes pantallas que tiene X-CTU para poder ser utilizadas.



**Figura 3.17 Pantallas del programa X-CTU**

Una vez instalado el programa se conecta la placa XBee Explorer USB con uno de los módulos XBee al puerto USB y se deberá abrir el programa X-CTU.

Se debe establecer la comunicación Zigbee una vez conseguido esto se podrá dar paso a la transferencia de datos que serán enviados desde el sensor Kinect para este proyecto.

## CAPÍTULO IV

### 4. ANÁLISIS DE RESULTADOS

En el análisis de resultados es necesario profundizar en varios parámetros que afectan el funcionamiento adecuado de la aplicación, tales como distancia entre el usuario y el sensor Kinect, la estatura y el número de los usuarios.

Para un correcto uso y que el usuario no tenga ningún problema con la aplicación es imprescindible que mediante el análisis sean tomados en cuenta las condiciones de funcionamiento y las posiciones que requiere el sensor Kinect.

Es importante mencionar la distancia en la cual el sensor Kinect está ubicado con respecto al piso, en varias pruebas realizadas el sensor está a una distancia de 1.18m y la distancia del sensor Kinect al usuario es de 1.80m la cual es considerada la óptima.

#### 4.1 Distancia Usuario - Sensor Kinect Y Estaturas.

Para obtener la distancia correcta entre el sensor Kinect y el usuario, es necesario tomar en cuenta los datos obtenidos por el sensor, es decir que sea detectado eficazmente. Así también como considerar las distancias no aceptadas como muy cercanas o muy lejanas, sin dejar de tomar en cuenta las estaturas de los usuarios.

Las posiciones, distancias y estaturas tomadas en cuenta dentro de este proyecto son:

- ✓ Posiciones: Manos y cabeza (ejes x, y, z).
- ✓ Distancias: 1m, 2m y 3m.
- ✓ Estaturas: 1.2m, 1.6m y 1.75m.



Se debe considerar las distancias y estaturas del usuario para que esta aplicación funcione de la mejor manera, con lo cual mediante el análisis obtuvimos los siguientes resultados:

## RESULTADOS:

### DISTANCIA = 1m

Al encontrarse el sensor Kinect a una distancia de 1 metro (distancia1), es posible visualizar que no es una distancia perfecta ya que ni el rostro, ni los hombros son detectados de manera correcta en ninguno de los 3 usuarios, con lo cual no es suficiente para graficar el Skeleton (puntos del esqueleto del usuario), ni para tener un funcionamiento óptimo.

### Usuario1 (1.2m)

De las figuras 4.1 a 4.4 el usuario 1 de una estatura de 1.2m, a la distancia de 1 m no es posible que el sensor Kinect lo observe de manera adecuada, ya que como se menciona anteriormente no es suficiente para graficar el Skeleton, con lo cual no es considerada ni una distancia ni estatura adecuada para un funcionamiento óptimo.

En la figura 4.1 el usuario 1 busca ser detectado por el sensor y mostrar una posición correcta.



Figura 4.1 Distancia Kinect (1m) Usuario1(1.2m)

En la figura 4.2 el usuario 1 da la orden de acelerar sin ser detectado por el sensor con lo cual es posible observar que el robot móvil sigue en la misma posición.



**Figura 4.2 Distancia Kinect (1m) Usuario1 (1.2m)**

En la figura 4.3 el usuario 1 da la orden de ir hacia la izquierda sin ser detectado por el sensor con lo cual es posible observar que el robot móvil sigue en la misma posición.



**Figura 4.3 Distancia Kinect (1m) Usuario1 (1.2m)**

En la figura 4.4 el usuario 1 da la orden de ir hacia la derecha sin ser detectado por el sensor con lo cual es posible observar que el robot móvil sigue en la misma posición.



**Figura 4.4 Distancia Kinect (1m) Usuario1 (1.2m)**

### Usuario 2 (1.6 m)

En las imágenes de las figuras 4.5 a 4.8, el usuario 2 de una estatura de 1.6m a la distancia de 1 m no es posible que el sensor Kinect lo observe de manera adecuada, ya que como se menciona anteriormente no es suficiente para graficar el Skeleton. Con lo cual no es considerada una distancia adecuada ya que los puntos del Skeleton no están en una posición correcta para su funcionamiento, aunque la estatura es óptima.

En la figura 4.5 el usuario 2 busca ser detectado por el sensor y mostrar una posición correcta.



**Figura 4.5 Distancia Kinect (1m) Usuario2 (1.6m)**

En la figura 4.6 el usuario 2 da la orden de ir hacia la derecha sin ser detectado por el sensor con lo cual es posible observar que el robot móvil sigue en la misma posición.



**Figura 4.6 Distancia Kinect (1m) Usuario2 (1.6m)**

En la figura 4.7 el usuario 2 da la orden de ir hacia la izquierda sin ser detectado por el sensor con lo cual es posible observar que el robot móvil sigue en la misma posición.



**Figura 4.7 Distancia Kinect (1m) Usuario2 (1.6m)**

En la figura 4.8 el usuario 2 da la orden de acelerar sin ser detectado por el sensor con lo cual es posible observar que el robot móvil sigue en la misma posición.



**Figura 4.8 Distancia Kinect (1m) Usuario2 (1.6m)**

### Usuario 3 (1.75m)

De las figuras 4.9 a 4.12, el usuario 3 de una estatura de 1.75m a la distancia de 1 m no es posible que el sensor Kinect lo observe de manera adecuada sin obtener correctamente el Skeleton como también ocurría con los usuarios anteriores.

En la figura 4.9 el usuario 3 busca ser detectado por el sensor y mostrar una posición correcta.



**Figura 4.9 Distancia Kinect (1m) Usuario3 (1.75m)**

En la figura 4.10, el usuario 3 da la orden de ir hacia la derecha sin ser detectado por el sensor con lo cual es posible observar que el robot móvil sigue en la misma posición.



**Figura 4.10 Distancia Kinect (1m) Usuario3 (1.75m)**

En la figura 4.11 el usuario 3 da la orden de ir hacia la izquierda sin ser detectado por el sensor con lo cual es posible observar que el robot móvil sigue en la misma posición.



**Figura 4.11 Distancia Kinect (1m) Usuario3 (1.75m)**

En la figura 4.12 el usuario 3 da la orden de acelerar sin ser detectado por el sensor con lo cual es posible observar que el robot móvil sigue en la misma posición.



**Figura 4.12 Distancia Kinect (1m) Usuario3 (1.75m)**

### **DISTANCIA = 2m**

El sensor Kinect se encuentra a una distancia de 2 metros, a esta distancia la cámara detecta por completo al usuario y es posible tomar los respectivos datos para el Skeletal, su funcionamiento en comparación con el anterior caso (distancia de 1m) es notorio, el rostro y el cuerpo de los 3 usuarios es detectado en su totalidad, obteniendo el control deseado para el robot móvil.

### **Usuario 1 (1.2m)**

De las figuras 4.13 a 4.15, el usuario 1 de una estatura de 1.2m a la distancia de 2 m, muestran que es posible tomar los respectivos datos para el Skeletal, su funcionamiento es correcto y recibe las señales para el robot móvil. Es detectado correctamente en todas las posiciones del usuario.

En la figura 4.13 el usuario 1 busca ser detectado por el sensor y mostrar una posición correcta.



**Figura 4.13 Distancia Kinect (2m) Usuario1 (1.2m)**

En la figura 4.14 el usuario1 da la orden de ir hacia la derecha y es detectado por el sensor con lo cual es posible observar que el robot móvil obedece a la orden del usuario1 y va hacia la derecha.



**Figura 4.14 Distancia Kinect (2m) Usuario1 (1.2m)**

En la figura 4.15 el usuario1 da la orden de ir hacia la izquierda y es detectado por el sensor con lo cual es posible observar que el robot móvil obedece a la orden del usuario1 y va hacia la izquierda.



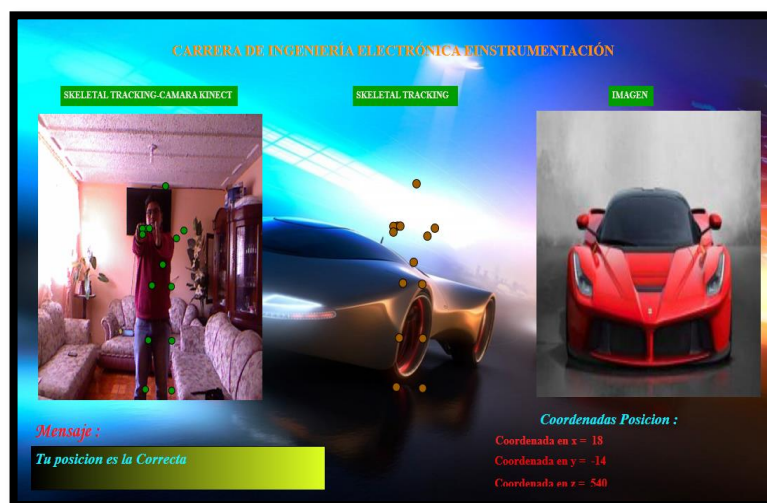


**Figura 4.15 Distancia Kinect (2m) Usuario1 (1.2m)**

### Usuario 2(1.6m)

De las figuras 4.16 a 4.1.8 el usuario 2 de una estatura de 1.6 m a la distancia de 2 m, muestran que es posible tomar los respectivos datos para el Skeletal, su funcionamiento es correcto y recibe las señales para el robot móvil. Es detectado correctamente en todas las posiciones del usuario.

En la figura 4.16 el usuario 2 busca ser detectado por el sensor y mostrar una posición correcta.



**Figura 4.16 Distancia Kinect (2m) Usuario2 (1.6m)**

En la figura 4.17 el usuario2 da la orden de ir hacia la derecha y es detectado por el sensor con lo cual es posible observar que el robot móvil obedece a la orden del usuario1 y va hacia la derecha.



**Figura 4.17 Distancia Kinect (2m) Usuario2 (1.6m)**

En la figura 4.18 el usuario2 da la orden de ir hacia la izquierda y es detectado por el sensor con lo cual es posible observar que el robot móvil obedece a la orden del usuario1 y va hacia la izquierda.



**Figura 4.18 Distancia Kinect (2m) Usuario2 (1.6m)**

### **Usuario 3 (1.75m)**

De las figuras 4.19 a 4.22, el usuario 3 de una estatura de 1.75m a la distancia de 2 m, muestran que es posible tomar los respectivos datos para el Skeletal, su funcionamiento es correcto y recibe las señales para el robot móvil. No es detectado en todas las posiciones del usuario.

En la figura 4.19 el usuario 3 busca ser detectado por el sensor y mostrar una posición correcta.



**Figura 4.19 Distancia Kinect (2m) Usuario3 (1.75m)**

En la figura 4.20 el usuario 3 da la orden de ir hacia la derecha pero no es detectado por el sensor con lo cual es posible observar que el robot móvil no obedece a la orden del usuario 3 para ir hacia la derecha.



**Figura 4.20 Distancia Kinect (2m) Usuario3 (1.75m)**

En la figura 4.21 el usuario 3 da la orden de ir hacia la izquierda y es detectado por el sensor con lo cual es posible observar que el robot móvil obedece a la orden del usuario1 y va hacia la izquierda.



**Figura 4.21 Distancia Kinect (2m) Usuario3 (1.75m)**

En la figura 4.22 el usuario 3 da la orden de acelerar y es detectado por el sensor con lo cual es posible observar que el robot móvil cambia de posición.



**Figura 4.22 Distancia Kinect (2m) Usuario (1.75m)**

### **DISTANCIA = 3m**

El sensor Kinect se encuentra a una distancia de 3 metros, el usuario a esta distancia es reconocido, pero no correctamente, al estar alejado del sensor Kinect es posible visualizar que los puntos del Skeletal se encuentran fuera de los 3 usuarios, es decir los puntos del Skeleton no están bien ubicados en el cuerpo del usuario. A esta distancia es posible tener un control adecuado en el robot móvil.

## Usuario 1 (1.2m)

De las figuras 4.23 a 4.26, el usuario 1 de una estatura de 1.2m a la distancia de 3 m, muestran que es posible ser reconocido pero no correctamente siendo visualizados los puntos del Skeletal fuera del usuario. No es detectado correctamente en todas las posiciones del usuario.

En la figura 4.23 el usuario 1 busca ser detectado por el sensor y mostrar una posición correcta.



**Figura 4.23 Distancia Kinect (3m) Usuario1 (1.2m)**

En la figura 4.24 el usuario 1 da la orden de ir hacia la derecha pero no es detectado por el sensor con lo cual es posible observar que el robot móvil no obedece a la orden del usuario 1 en ir hacia la derecha.



**Figura 4.24 Distancia Kinect (3m) Usuario1 (1.2m)**

En la figura 4.25 el usuario 1 da la orden de ir hacia la izquierda y es detectado por el sensor con lo cual es posible observar que el robot móvil obedece a la orden del usuario1 y va hacia la izquierda.



**Figura 4.25 Distancia Kinect (3m) Usuario1 (1.2m)**

En la figura 4.26 el usuario 1 da la orden de acelerar y es detectado por el sensor con lo cual es posible observar que el robot móvil cambia de posición.



**Figura 4.26 Distancia Kinect (3m) Usuario1 (1.2m)**

Nota: Es importante mencionar que en esta distancia (3m), las ordenes que se dan con el movimiento de la cabeza y que el robot móvil obedece, solo se produce en la posición en x y queda deshabilitado por la estatura.

### **Usuario 2(1.6m)**

De las figuras 4.27 a 4.31, el usuario 2 de una estatura de 1.6m a la distancia de 3 m muestra que es posible ser reconocido pero no correctamente, siendo visualizados los puntos del Skeletal fuera del usuario. No es detectado correctamente en todas las posiciones del usuario.

En la figura 4.27 el usuario 2 busca ser detectado por el sensor y mostrar una posición correcta.



**Figura 4.27 Distancia Kinect (3m) Usuario2 (1.6m)**

En la figura 4.28 el usuario 2 da la orden de ir hacia la derecha y es detectado por el sensor, con lo cual es posible observar que el robot móvil obedece a la orden del usuario 2 en ir hacia la derecha.



**Figura 4.28 Distancia Kinect (3m) Usuario2 (1.6m)**

En la figura 4.29 el usuario 2 da la orden de ir hacia la izquierda y es detectado por el sensor con lo cual es posible observar que el robot móvil obedece a la orden del usuario 2 y va hacia la izquierda.





**Figura 4.29 Distancia Kinect (3m) Usuario2 (1.6m)**

En la figura 4.30 el usuario 2 da la orden de acelerar y es detectado por el sensor con lo cual es posible observar que el robot móvil cambia de posición.



**Figura 4.30 Distancia Kinect (3m) Usuario2 (1.6m)**

En la figura 4.31 es posible observar la orden dada por el usuario2 que es el movimiento de cabeza, la cual consiste en el movimiento de la cámara pero esta no es detectada a la distancia de 3m ya que es considerada muy lejana.



Figura 4.31 Distancia Kinect (3m) Usuario2 (1.6m)

### Usuario 3 (1.75m)

De las figuras 4.32 a 4.35, el usuario 3 de una estatura de 1.75 m a la distancia de 3 m, muestran que es posible ser reconocido pero no correctamente siendo visualizados los puntos del Skeletal fuera del usuario. No es detectado correctamente en todas las posiciones del usuario.

En la figura 4.32 el usuario 3 da la orden de ir hacia la derecha pero no es detectado por el sensor con lo cual es posible observar que el robot móvil obedece a la orden del usuario 3 en ir hacia la derecha.



Figura 4.32 Distancia Kinect (3m) Usuario3(1.75m)

En la figura 4.33 el usuario 3 da la orden de ir hacia la izquierda y es detectado por el sensor, con lo cual es posible observar que el robot móvil obedece a la orden del usuario1 y va hacia la izquierda.



**Figura 4.33 Distancia Kinect (3m) Usuario3 (1.75m)**

En la figura 4.34 el usuario 3 da la orden de acelerar y es detectado por el sensor con lo cual es posible observar que el robot móvil cambia de posición.



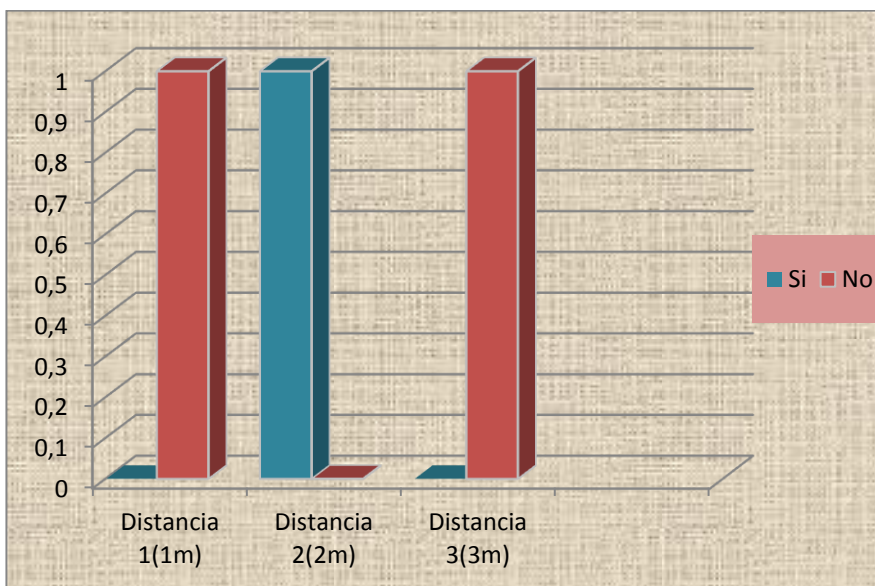
**Figura 4.34 Distancia Kinect (3m) Usuario3 (1.75m)**

En la figura 4.35 presenta la misma observación que la figura 4.31 que es posible observar el orden dado por el usuario3 que es el movimiento de cabeza la cual consiste en el movimiento de la cámara pero esta no es detectada por la distancia de 3m ya que es considerada muy lejana.

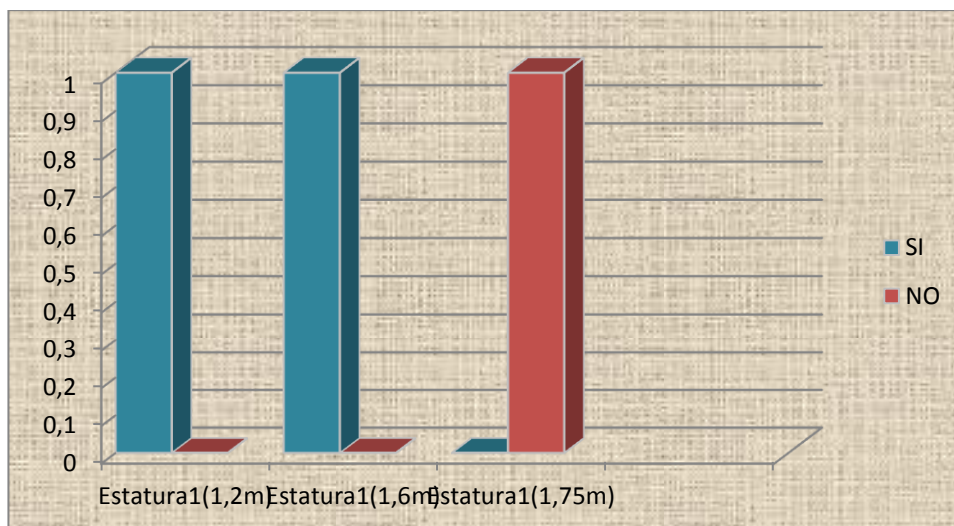


**Figura 4.35 Distancia Kinect (3m) Usuario3 (1.75m)**

Mediante los datos obtenidos y la visualización de los resultados se determinó una distancia óptima entre el sensor y el usuario debe estar en el rango de 1.8m a 2m para que la aplicación funcione, las distancias menores no son las adecuadas ya que el sensor Kinect no detecta al usuario, en distancias superiores los puntos del Skeletal no se encuentran dentro del cuerpo del usuario sino fuera de él, es decir tampoco es detectado en su totalidad.

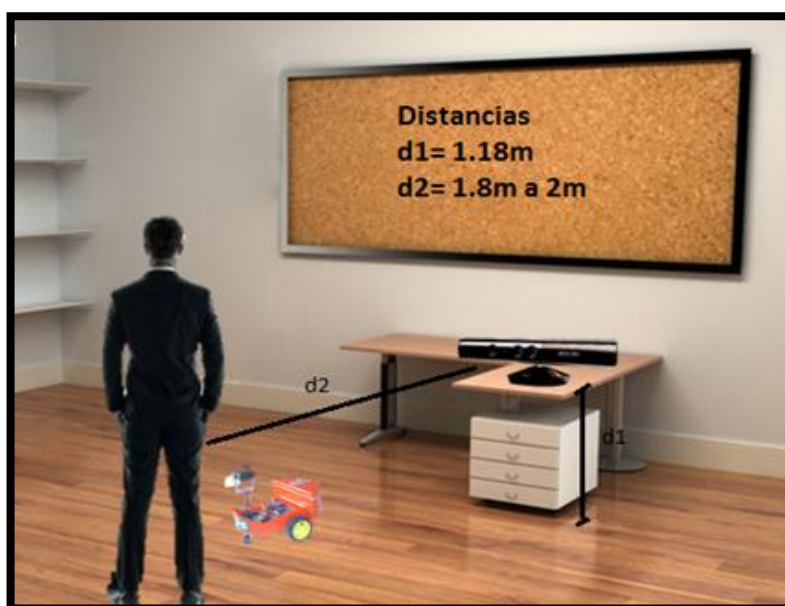


**Figura 4.36 Grafico de Probabilidades de Funcionamiento con las 3 Diferentes Distancias Realizadas en el Proyecto**



**Figura 4.37 Grafico de Probabilidades de Funcionamiento con las 3 Diferentes Distancias Realizadas en el Proyecto**

En la figura 4.36 Se puede observar las tres distancias establecidas en el proyecto tales como 1, 2 y 3 metros las cuales muestra si son o no reconocidas. Así también la figura 4.37 Muestra con las tres estaturas parámetro para este proyecto. Con lo cual es posible mirar que es reconocido a una distancia de 2m y a una estatura de 1.6m. Por lo cual en la figura 4.38 Se muestra la distancia entre el usuario y el sensor Kinect y la distancia del piso hacia el sensor para un funcionamiento óptimo.



**Figura 4.38 Distancias Kinect-Usuario.**

A la aplicación se añadieron mensajes que explican al usuario si su posición es correcta o si debe dar un paso atrás o un paso adelante.

Al encontrarse a una distancia menor que la de 2 m el mensaje que se presenta es que debe dar un paso atrás, si su posición, a pesar de dar el paso hacia atrás sigue en una posición menor, seguirá dando este mensaje hasta que el usuario se encuentre en la posición y distancia correcta, de la misma forma pasará si la distancia esta más de 2m, el mensaje en este caso será que debe dar un paso adelante.

Una vez que la posición sea la adecuada el mensaje será “TU POSICIÓN ES LA CORRECTA” y se podrá dar paso al control del robot móvil.

#### **4.2 Estatura Del Usuario**

Se realizaron varias pruebas a una distancia de 1.8m entre el usuario y el sensor Kinect que se determinó era la óptima pero con diferentes estaturas de los usuarios, y así poder observar el comportamiento de la aplicación con estas variaciones.

Al ser sometido a estas pruebas, es importante observar los cambios significativos que existen entre cada uno de los usuarios al momento de manipular o poner en práctica la aplicación.

Para realizar esta prueba se consideraron 3 usuarios con diferentes estaturas: 1.20m, 1.60m y 1.75m.

#### **4.3 Número De Usuarios**

La aplicación fue diseñada para que un solo usuario esté al frente del sensor Kinect. Como se explicó inicialmente, el seguimiento esquelético permite a Kinect reconocer hasta seis usuarios en el campo de visión del sensor pero de los cuales solo dos pueden ser rastreados con precisión, en esta aplicación se reconoce al que primero ingresa en el rango de visión de la cámara.

Si existe un segundo usuario, Kinect seguirá reconociendo al primer usuario, y los puntos del Skeleton se mantendrán sobre el usuario reconocido como se muestra en la figura 4.37, por lo que es recomendable que para la manipulación de esta aplicación solo se encuentre una persona frente al sensor Kinect.



Figura 4.39 Reconocimiento de usuarios del sensor Kinect.

## CAPÍTULO V

### 5. CONCLUSIONES Y RECOMENDACIONES

#### 5.1 Conclusiones

- ✓ Por medio del estudio de los códigos fuentes SDK de Kinect, se logró desarrollar una aplicación demostrativa que es capaz de capturar los movimientos de brazos, manos y cabeza; cada uno plasmado en los diferentes ejes de coordenadas x, y, z. Estos valores entregados aportaron a la implementación del robot móvil la cual permite que la comunicación entre el lenguaje de programación y sensor Kinect sea eficiente.
- ✓ Los códigos fuentes SDK fueron estudiados en tres segmentos los cuales son Skeletal Traking, Depth Data y Cámara RGB; los mismos que de forma detallada se especifican sus funciones características. El Skeletal Tracking se enlaza mediante puntos o “Joint” a la estructura del cuerpo humano para graficarlo, el sensor 3D permite definir la profundidad o la distancia a la que se encuentra el usuario de la misma forma la cámara fundamental RGB entrega un entorno visual de trabajo. Esta información queda como sustento para futuros estudios y mejoras en nuevos procesos.
- ✓ Mediante la captura de los datos obtenidos por el sensor kinect, los cuales enviados por el puerto serial desde la computadora hacia los servomotores a través de los módulos ZigBee, los mismos que no presentan pérdida de datos siempre y cuando la red establecida esté configurada eficazmente entre emisor y receptor. De esta forma el control del robot móvil es eficiente.



- ✓ Se realizó la comunicación inalámbrica entre robot móvil receptor y emisor de una forma eficaz ya que el emisor se comunica con la computadora mediante comunicación serial la cual envía los datos de una manera muy fiable y segura.
- ✓ Por medio de las pruebas realizadas se determinó una distancia óptima entre el usuario y el sensor Kinect, mediante varias de estas pruebas con diferentes usuarios, se encontró que la distancia que mejor resultados obtiene en detección del usuario para un funcionamiento adecuado fue la distancia de 1.8m.
- ✓ Así también mediante las pruebas se obtuvo que la estatura optima del usuario para una mejor detección de este, es de 1.6 m.

## **5.2 Recomendaciones**

- ✓ Usar la distancia establecida como óptima dentro del proyecto de 1.8m a 2 m, ya que a esta distancia el sensor Kinect capta correctamente todos los datos del usuario, es decir, hay un funcionamiento adecuado, todas las órdenes dadas por el usuario sean claramente aceptadas.
- ✓ La interfaz gráfica fue realizada con una resolución de 1100 x 600 pixeles, es preferible que al momento de utilizar la aplicación la resolución de pantalla sea 1280 x 720.
- ✓ Tener en cuenta que la aplicación demostrativa fue elaborada en el lenguaje C, razón por la cual se cargaron librerías como Microsoft.Research.Kinect.Nui y Coding4Fun.Kinect.Wpf, ya que si estas librerías no constan en el proyecto este no tendrá un correcto funcionamiento.

- ✓ La aplicación demostrativa fue elaborada para que solo un usuario esté al frente de Kinect, y este pueda captar los movimientos y controlar el robot móvil razón por la cual si existiera otros usuarios el Kinect no podría reconocer y los valores captados serían imprecisos en la captación del movimiento del cuerpo humano.
  
- ✓ Este proyecto ha sido posible gracias a la tecnología de Kinect que nos permite conseguir un esqueleto del usuario fácil y fielmente. Aun así el tratamiento de los datos para que fuera útiles requirió un esfuerzo considerable con lo cual ayudará a nuevos proyectos que necesiten añadir o implementar lo realizado en el presente proyecto.

## REFERENCIAS BIBLIOGRAFICAS

- [1] N.R.Velasco, « <http://repositorio.espe.edu.ec/handle/21000/6155>,» [En línea]. [Último acceso: 28 JULIO 2014].
- [2] E. TIEMPO, «<http://m.eltiempo.com/tecnologia/entretenimiento-tecnologia/el-kinect-ofrece-ventajas-en-distintos-campos-profesionales-como-la-salud-la-ciencia-el-comercio-y-la-educacin/11583682>,» .[En línea]. [Último acceso: 2 Agosto 2014].
- [3] OSRFOUNDATION, «<http://www.turtlebot.com/>,»TURTLEBOT 2. [En línea]. [Último acceso: 15 Octubre 2014].
- [4] E. TIEMPO, « <http://m.eltiempo.com/buscador/CMS-11583682/1>,» El Kinect ofrece ventajas en distintos campos profesionales. [En línea]. [Último acceso: 16 Octubre 2014].
- [5] MICROSOFT, « [http://www.pcactual.com/articulo/laboratorio/analisis/perifericos/otros\\_perifericos/10527/kinect\\_para\\_windows\\_control\\_movimiento\\_mas\\_alla\\_de\\_los\\_juegos.html](http://www.pcactual.com/articulo/laboratorio/analisis/perifericos/otros_perifericos/10527/kinect_para_windows_control_movimiento_mas_alla_de_los_juegos.html), » Kinect para Windows: control de movimiento más allá de los juegos. [En línea]. [Último acceso: 9 Noviembre 2014].
- [6] ISLABIT, « <http://www.islabit.com/20561/microsoft-confirma-kinect-windows-sdk.html>, » Microsoft confirma Kinect Windows SDK. [En línea]. [Último acceso: 16 Octubre 2014].
- [7] MICROSOFT, « <http://blogs.windows.com/latam/2012/07/20/el-poder-de-kinect-para-windows/>,» El poder de Kinect para Windows. [En línea]. [Último acceso: 11 Noviembre 2014].
- [8] MICROSOFT, « <http://support.xbox.com/en-US/xbox-360/kinect/kinect-sensor-components>,» Kinect sensor for Xbox 360 components. [En línea]. [Último acceso: 12 Noviembre 2014].
- [9] MICROSOFT, « <http://support.xbox.com/es-ES/xbox-360/kinect/speech-recognition>,» Reconocimiento de voz y charla en Kinect para Xbox 360. [En línea]. [Último acceso: 12 Noviembre 2014].
- [10] IDEASGEEK, « <http://www.ideasgeek.net/2010/11/10/funcionamiento-del-sensor-de-movimiento-en-kinect/>,» Funcionamiento del sensor de movimiento en Kinect .[En línea]. [Último acceso: 15 Noviembre 2014].

- [11] MICROSOFT, « <http://es.engadget.com/2010/11/10/microsoft-kinect-para-xbox-360-un-analisis-movidito-de-lo-nuevo/>,» Microsoft Kinect para Xbox 360: Un análisis movidito de lo nuevo de Redmond .[En línea]. [Último acceso: 15 Noviembre 2014].
- [12] MICROSOFT, «Diálogo que tenga en cuenta el contexto con Kinect,» [En línea]. Available: <https://msdn.microsoft.com/es-es/magazine/hh882450.aspx>. [Último acceso: 15 Noviembre 2014].
- [13] MICROSOFT, « <https://msdn.microsoft.com/en-us/library/hh973075.aspx>,» Data Streams .[En línea]. [Último acceso: 15 Noviembre 2014].
- [14] MICROSOFT, « <https://msdn.microsoft.com/en-us/library/jj131028.aspx>,» Depth Stream. [En línea]. [Último acceso: 15 Noviembre 2014].
- [15] MICROSOFT, « <https://msdn.microsoft.com/en-us/library/jj131025.aspx>,» Tracking Users with Kinect Skeletal Tracking .[En línea]. [Último acceso: 17 Noviembre 2014].
- [16] MICROSOFT. « <https://praveenitech.wordpress.com/tag/skeleton-tracking/>,» [En línea]. [Último acceso: 17 Noviembre 2014].
- [17] MICROSOFT. «<https://msdn.microsoft.com/en-us/library/hh973074.aspx>,» [En línea]. [Último acceso: 17 Noviembre 2014].
- [18] ARDUINO, « <https://www.arduino.cc/en/Guide/ArduinoXbeeShield>,» Arduino XBee Shield. [En línea]. [Último acceso: 16 Enero 2015].
- [19] ALPHACRUCIS, « <http://www.alpha-crucis.com/es/arduino/718-arduino-shield-xbee-en-3700386600074.html>,» Arduino Shield - Xbee .[En línea]. [Último acceso: 17 Enero 2015].
- [20] D. DINER, « <http://digitaldiner.blogspot.com/2012/10/arduino-uno-vs-beaglebone-vs-raspberry.html>,» Arduino Uno vs BeagleBone vs Raspberry Pi . [En línea]. [Último acceso: 17 Enero 2015].
- [21] S. ELECTRONIC, « <http://savageelectronics.blogspot.com/2011/09/configurando-radios-xbee.html>,» Configurando Radios XBee .[En línea]. [Último acceso: 17 Enero 2015].
- [22] J. D. ESTEVE, «

[http://platea.pntic.mec.es/vgonzale/cyr\\_0708/archivos/\\_15/Tema\\_5.5.htm](http://platea.pntic.mec.es/vgonzale/cyr_0708/archivos/_15/Tema_5.5.htm),» Robots móviles. [En línea]. [Último acceso: 21 Enero 2015].

- [23] J. LONGARES, « <http://www.javierlongares.com/arte-en-8-bits/introduccion-a-zigbee-y-las-redes-de-sensores-inalambricas/>,» Introducción a Zigbee y las redes de sensores inalámbricas .[En línea]. [Último acceso: 21 Enero 2015].
- [24] C.I.r.E, « <http://webdelcire.com/wordpress/archives/1714>,» Comenzando con ZigBee. [En línea]. [Último acceso: 21 Enero 2015].
- [25] I. V. SALGADO. « <http://www.dea.icaei.upco.es/sadot/Comunicaciones/avanzadas/Zigbee%20y%20sus%20aplicaciones.pdf>,» [En línea]. [Último acceso: 10 Febrero 2015].
- [26] DOMODESK, « <http://www.domodesk.com/a-fondo-zigbee>,» Zigbee .[En línea]. [Último acceso: 19 Febrero 2015].
- [27] SCRIB, « <http://es.scribd.com/doc/139000918/XBee-Guia-Usuario-pdf#scribd>,» XBEE Series 1. [En línea]. [Último acceso: 15 Marzo 2015].
- [28] CAPSULABS, « <https://hangar.org/webnou/wp-content/uploads/2012/01/Capsulab081.pdf>,» Xbee y Arduino. [En línea]. [Último acceso: 16 Marzo 2015].

## **ANEXOS**

**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE**

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN**

**CERTIFICACIÓN**

Se certifica que el presente trabajo fue desarrollado por la Sra. MARÍA GABRIELA CHÁVEZ RUGEL y el Sr. IVÁN MARCELO TATAYO SIMBAÑA.

MSc. EDDIE, GALARZA

**DIRECTOR**

MSc. DAVID, RIVAS

**CODIRECTOR**

ING. FRANKLIN, SILVA

**DIRECTOR DE LA CARRERA**

DR. RODRIGO, VACA

**SECRETARIO ACADÉMICO**