



**UNIVERSIDAD NACIONAL DE CHIMBORAZO**  
**FACULTAD DE INGENIERÍA**

ESCUELA DE INGENIERÍA EN ELECTRONICA Y TELECOMUNICACIONES

Trabajo de grado previo a la obtención del Título de: **Ingeniero en Electrónica y Telecomunicaciones**

**TRABAJO DE GRADUACIÓN**

**“CONTROL DE UN BRAZO ROBÓTICO A DISTANCIA DESARROLLADO EN AMBIENTE LABVIEW, SOBRE UNA RED ETHERNET Y ATRAVES DE MENSAJES**

**DE TEXTO VIA TELEFONO CELULAR”**

**Autor:** Pablo Aníbal Tenelanda Sanaguano

**Director:** Ing. Daniel Santillán

**Riobamba – Ecuador**

**AÑO**  
**2012**

Los miembros del Tribunal de Graduación del proyecto de investigación de título: “CONTROL DE UN BRAZO ROBÓTICO A DISTANCIA DESARROLLADO EN AMBIENTE LABVIEW, SOBRE UNA RED ETHERNET Y ATRAVES DE MENSAJES DE TEXTO VIA TELEFONO CELULAR” presentado por: Pablo Aníbal Tenelanda Sanaguano y dirigida por: Daniel Santillán Haro.

Una vez escuchada la defensa oral y revisado el informe final del proyecto de investigación con fines de graduación escrito en la cual se ha constatado el cumplimiento de las observaciones realizadas, remite la presente para uso y custodia en la biblioteca de la Facultad de Ingeniería de la UNACH.

Para constancia de lo expuesto firman:

-----  
Ing. Yesenia Cevallos  
Presidente del Tribunal

-----  
Nota

-----  
Firma

-----  
Ing. Fabián Gunsha  
Miembro del Tribunal

-----  
Nota

-----  
Firma

-----  
Ing. Daniel Santillán  
Director del Proyecto

-----  
Nota

-----  
Firma

## **AUTORÍA DE LA INVESTIGACIÓN**

La responsabilidad del contenido de este Proyecto de Graduación, nos corresponde exclusivamente a: Pablo Aníbal Tenelanda Sanaguano y Daniel Santillán Haro; y el patrimonio intelectual de la misma a la Universidad Nacional de Chimborazo

## **AGRADECIMIENTO**

*Agradezco a Dios que me ha dado la sabiduría para seguir adelante.*

*Un agradecimiento muy especial a mi Mami querida pues con sus consejos, amor, comprensión y apoyo incondicional durante toda mi vida, ha sido un pilar fundamental para que pueda culminar mi carrera universitaria con éxito.*

*A mis hermanas Angelita y Marthita, que en todos estos años de Universidad me han brindado sus cuidados, consejos, amor y cariño para que pueda alcanzar este logro tan importante en mi vida.*

*A mis hermanos Freddy y Manuel que constantemente me han dado palabras de aliento para seguir adelante y vencer todas las circunstancias adversas que han surgido mientras cursaba la Universidad.*

*A mi esposa Ximena por el amor y comprensión que me ha brindado.*

*Mi agradecimiento a los Docentes de la Universidad Nacional de Chimborazo, por haber impartido sus conocimientos en las aulas para de esta manera nosotros podernos forjar un futuro mejor.*

*Finalmente el agradecimiento a la Universidad Nacional de Chimborazo de cuyas aulas llevo en mi mente los mejores recuerdos.*

## **DEDICATORIA**

*Este logro se lo dedico a mi Padre y mi Hermano que aunque no están con nosotros siempre los he llevado presentes en mi corazón.*

*A mi Mami, pues a esa persona tan maravillosa le debo todo mi ser, todo lo que tengo y todo lo que soy. A Angelita, Marthita, Freddy y Manuel, por brindarme su apoyo incondicional durante estos años.*

*A mi esposa por su amor y comprensión brindados en estos días.*

*A mis sobrinas y sobrinos que en ellos veo el reflejo de todo el amor y comprensión que hay en nuestra Familia.  
A todos Ustedes les dedico este triunfo.*

## ÍNDICE GENERAL

<i>Índice</i>	
<i>General</i> .....	<i>i</i>
<i>Índice</i>	<i>de</i>
<i>Cuadros</i> .....	<i>vi</i>
<i>Resumen</i> .....	
<i>vii</i>	
<i>Summary</i> .....	<i>x</i>
<i>iii</i>	
<i>Introducción</i> .....	<i>xiv</i>

### **CAPÍTULO I**

<b>1. MARCO REFERENCIAL</b> .....	
19	
1.1.Planteamiento del Problema .....	
19	
1.2.FORMULACIÓN DEL PROBLEMA.....	
20	
1.3.OBJETIVOS .....	
21	
1.3.1.	
General.....	21

1.3.2	Específicos: .....	21
	JUSTIFICACIÓN .....	22
2.	FUNDAMENTACIÓN TEÓRICA .....	23
2.1	ANTECEDENTES .....	23
2.2	ALCANCES DEL PROYECTO .....	23
2.3	ARQUITECTURA DE UN SISTEMA DE CONTROL AUTOMÁTICO .....	24
2.3.1	DCS. ....	26
2.	4.CRITERIOS DE DISEÑO DE UN SISTEMA DE CONTROL AUTOMÁTICO.....	26
2.4.1	Características de un sistema Automatizado .....	27
2.	Funciones Principales .....	27
2.4.3	Estación Maestra.....	28
	Características .....	29
	Infraestructura y Métodos de Comunicación .....	30
2.4.4	Arquitectura de red .....	31
	Notificaciones .....	31
2.4.6	Procedimientos .....	32
2.4.7	Plataforma utilizada en el desarrollo de aplicaciones industriales .....	32
2.4.7.1	LabVIEW .....	32
	TECNOLOGÍAS A UTILIZAR .....	33
2.5.1	Instrumentación de Control .....	33
	Compact FieldPoint Modelo 2020 .....	33
	Módulo NI cFP-DO-400 (Digital I/O) .....	35
	Módulo NI cFP-AI-100 (Analog I/O) .....	36
	Módulo NI cFP-CB-1 .....	37
2.5.2	Software .....	38
	LabVIEW 2011 .....	38

¿Cómo Trabaja LabVIEW? .....	40
Panel Frontal .....	40
Diagrama de Bloques .....	41
Paletas. ....	42
Paleta de herramientas (Tools palette) .....	42
Paleta de controles (Controlspalette) .....	43
Paleta de Funciones (functionspalette) .....	43
Programación en Labview .....	44
Ejecución de un VI .....	45
Estructuras .....	46
Arquitectura de Software para Instrumentos Virtuales (VISA) .....	48
Sistema de transmisión de datos por red celular .....	49
Telefonía celular .....	49
Red de telefonía móvil GSM .....	49
Mensajes de texto SMS .....	49
Características .....	50
Arquitectura de la red SMS .....	51
2.5.2.2.5. Aplicaciones .....	52
Modem GSM .....	53
Puerto de comunicación .....	53
MicroCode Studio .....	54
Características .....	54
Configuración de MicroCode Studio .....	54
Programador de Microcontroladores .....	55

	<i>Microcontroladores</i> .....	55
	<i>Vectir</i> .....	57
	<i>Comandos AT</i> .....	58
	<i>Descripción</i> .....	58
	<i>Objetivo de los comandos AT</i> .....	59
	<i>Comandos de configuración</i> .....	59
	<i>Comandos para envío SMS</i> .....	61
	<i>Comandos para recepción de SMS</i> .....	62
	<i>Comandos AT más utilizados</i> .....	63
2.5.3.	<i>Infraestructura de Comunicación</i> .....	63
2.5.3.3.	<i>Nivel de Campo</i> .....	63
2.5.3.3.1.	<i>Protocolo de Comunicación TCP/IP (CFP – ESTACIÓN MAESTRA)</i> .....	63
2.5.3.3.2.	<i>Partes y Elementos CFP Modelo 2020</i> .....	65
2.5.3.4.	<i>Nivel de Control Remoto y Supervisión</i> .....	67
2.5.3.4.1.	<i>Protocolo HTTP</i> .....	67
2.5.3.4.2.	<i>Web Publishing Tool LabVIEW</i> .....	68
3.	<i>METODOLOGÍA</i> .....	71
3.1	<i>TIPO DE ESTUDIO</i> .....	71
3.2	<i>COMPÁRACION ENTRE CFP Y PLC</i> .....	71
3.3	<i>DISEÑO Y DESARROLLO DEL SISTEMA DE MONITOREO Y CONTROL</i> .....	74
3.3.1.	<i>Panel de Controles en Tiempo Real</i> .....	74
	<i>Lógica de Funcionamiento</i> .....	74
	<i>Diseño e Implementación en LabVIEW</i> .....	75
	<i>Secuencia Automática</i> .....	77

3.3.1.3.1.	<i>Lógica de Funcionamiento</i> .....	78
3.3.1.3.2.	<i>Diseño e Implementación en LabVIEW</i> .....	79
3.3.1.4.	<i>Envío de Notificaciones</i> .....	80
3.3.1.4.1.	<i>Lógica de Funcionamiento</i> .....	81
3.3.1.4.2.	<i>Diseño e Implementación en LabVIEW</i> .....	81
	<i>Configuración de la Herramienta de Publicación WEB</i> .....	85
	<i>Control mediante SMS</i> .....	89
3.3.1.6.1.	<i>Selección del Microcontrolador</i> .....	89
3.3.1.6.2.	<i>Características del PIC 16F628A</i> .....	89
3.3.1.6.3.	<i>Diagrama de pines pinout</i> .....	90
3.3.1.6.4.	<i>Selección del dispositivo de comunicación GSM</i> .....	91
3.3.1.6.5.	<i>Teléfono Nokia</i> .....	92
3.3.1.6.6.	<i>Elección del terminal</i> .....	92
3.3.1.6.7.	<i>Características del Teléfono</i> .....	92
3.3.1.6.8.	<i>Descripción del puerto y cable de comunicación</i> .....	93
3.3.1.7.	<i>Funcionamiento</i> .....	94
3.3.1.7.1.	<i>Diagrama Esquemático del circuito externo</i> .....	96
3.3.1.7.2.	<i>Códigos utilizados por el sistema</i> .....	98
3.3.1.7.3.	<i>Instrucciones para la transmisión serial</i> .....	98
3.3.1.7.4.	<i>Programa en MicroCode Studio</i> .....	99
3.3.1.8.	<i>Monitoreo de los movimientos del brazo mediante celular</i> .....	100
3.3.1.8.1.	<i>Configuración de Vectir</i> .....	100
4.	<i>PRUEBAS Y RESULTADOS</i> .....	103
	<i>1.OBJETIVO:</i> .....	

4.2	PRUEBAS:	.....	104
4.2.1	Pruebas de conexión estación maestra e instrumento de campo.	.....	104
4.2.1.1.	Configuración de Adaptadores de Red	.....	104
4.2.1.2.	Pruebas de conexión	.....	107
4.2.2.	Pruebas de eficiencia en dispositivo cfp y módulos.	.....	108
4.2.2.1.	Pruebas en Módulos conectados a CFP	.....	109
4.2.2.1.1.	Pruebas en Módulo CFP-DO-400 @4	.....	110
4.2.2.1.2.	Pruebas en Módulo CFP-DO-400 @5	.....	112
3.	PRUEBAS DE CONEXIÓN ESTACION DE CONTROL Y MONITOREO CELULAR	.....	113
4.2.3.1.	Configuración de Direccionamiento IP	.....	113
4.2.3.2.	Pruebas de direccionamiento y conectividad	.....	114
4.2.3.3.	Dispositivo Celular	.....	116
4.2.4.	Pruebas de acceso al modo de control remoto	.....	116
4.2.5.	Pruebas de monitoreo celular	.....	117
4.2.6.	Pruebas de control vía SMS	.....	118
4.2.7.	Pruebas en envío de notificaciones	.....	128
4.2.7.1.	Pruebas de recepción en Buzones	.....	129
4.3	RESULTADOS	.....	131
1.	RESULTADOS: PRUEBAS DE CONEXIÓN ESTACIÓN MAESTRA E INSTRUMENTO DE CAMPO.	.....	131
4.3.2.	RESULTADOS: PRUEBAS DE EFICIENCIA EN DISPOSITIVO CFP Y MÓDULOS.	.....	132
4.3.3.	RESULTADOS: PRUEBAS DE CONEXIÓN ESTACIONE DE CONTROL REMOTO Y MONITOREO CELULAR (Estación Maestra).	.....	132
4.3.4.	RESULTADOS: PRUEBAS DE ACCESO AL MODO CONTROL REMOTO	.....	132
4.3.5.	RESULTADOS: PRUEBAS DE MONITOREO CELULAR	.....	132
4.3.6.	RESULTADOS: PRUEBAS DE CONTROL VÍA SMS	.....	133
4.3.7.	RESULTADOS: PRUEBAS EN ENVIO DE NOTIFICACIONES	.....	133
5.1.	CONCLUSIONES Y RECOMENDACIONES	.....	133
5.1.1.	CONCLUSIONES	.....	133
5.2.	RECOMENDACIONES	.....	

.....	134 Referencias Bibliográficas
.....	136 Referencias Electrónicas
.....	136

**MANUAL TÉCNICO**

1.	<i>INTRODUCCIÓN</i> .....	139
2.	<i>OBJETIVOS</i> .....	140
2.1.	<i>OBJETIVO GENERAL</i> .....	140
2.2.	<i>OBJETIVO ESPECIFICO</i> .....	140
3.	<i>CONTENIDO TECNICO</i> .....	141
3.1.	<i>MANUAL DEL SISTEMA</i> .....	141
3.1.1.	<i>ESPECIFICACIONES DE LA INTERFAZ</i> .....	141
3.1.1.1.	<i>PANEL FRONTAL DEL SISTEMA</i> .....	141
3.1.1.2.	<i>Controles en Tiempo Real</i> .....	141
3.1.1.3.	<i>Controles Automáticos</i> .....	142
3.1.1.4.	<i>Controles vía SMS</i> .....	143
3.1.1.5.	<i>Sensor de Fuerza</i> .....	143
3.1.1.6.	<i>Visión Artificial</i> .....	144
3.1.2.	<i>ESPECIFICACIONES TÉCNICAS DEL FLUJO DE DATOS</i> .....	144
3.1.2.1.	<i>PANEL FRONTAL DEL SISTEMA</i> .....	144
3.1.2.2.	<i>Controles en tiempo Real</i> .....	145
3.1.2.2.1.	<i>Funcionamiento Lógico</i> .....	145
3.1.2.3.	<i>Controles Automáticos</i> .....	149
3.1.2.3.1.	<i>Funcionamiento Lógico</i> .....	149
3.1.2.3.2.	<i>Diseño e implementación en LabVIEW</i> .....	150
3.1.2.4.	<i>Controles 3.1.</i> .....	158

3.1.2.5.	<i>Sensor de Fuerza</i> .....	158
3.1.2.6.	<i>Configuración de la Herramienta de Publicación WEB</i> .....	159
3.1.2.7.	<i>Visión Artificial</i> .....	164
3.1.2.8.	<i>Adquisición de datos VISA</i> .....	165
3.2.	<i>CONEXIÓN ELEMENTOS DE CAMPO</i> .....	166
3.2.1.	<i>Configuración de Adaptadores de Red</i> .....	166
3.2.1.1.	<i>Pruebas de conexión</i> .....	169
3.2.2.	<i>PRUEBAS DE CONEXIÓN EN DISPOSITIVO CFP Y MÓDULOS.</i> .....	170
3.2.2.2.	<i>Pruebas en Módulos conectados a CFP</i> .....	173
3.2.3.	<i>ACCESO EN ESTACIONES REMOTAS</i> .....	174
3.2.4.	<i>ACCESO DESDE TERMINALES REMOTAS</i> .....	179
1.	<i>INTRODUCCIÓN</i> .....	183
2.	<i>OBJETIVOS</i> .....	184
	<i>OBJETIVO GENERAL</i> .....	184
	<i>OBJETIVO ESPECIFICO</i> .....	184
3.	<i>CONTENIDO TECNICO</i> .....	185
	<i>INTERFAZ DE LA APLICACIÓN</i> .....	185
	<i>3.2CONTROLES EN TIEMPO REAL</i> .....	187
	<i>CONTROLES AUTOMATICOS</i> .....	188
	<i>ENVIO DE NOTIFICACIONES VIA CORREO ELECTRONICO</i> .....	189
	<i>3.5CONTROL VIA SMS</i> .....	189
	<i>3.6.SENSOR DE FUERZA</i> .....	190
	<i>3.7.Visión Artificial</i> .....	190
	<i>3.8. ACCESO VIA WEB</i> .....	191
	<i>3.8.1. ACCEDIENDO AL MODO DE MONITOREO REMOTO</i> .....	191
	<i>Anexos</i> .....	137

## ÍNDICE DE CUADROS

<i>Cuadro 2.1: Esquema de un sistema DCS.....</i>	<i>24</i>
<i>Cuadro 2.2: Partes de un CFP 2020.....</i>	<i>65</i>
<i>Cuadro 2.3: Partes de un Controlador CFP 2020.....</i>	<i>66</i>
<i>Cuadro 3.1: Elementos de Programación.....</i>	<i>74</i>
<i>Cuadro 3.2: Estado de Servomotores Ejecución 1.....</i>	<i>75</i>
<i>Cuadro 3.3: Estado de Servomotores Ejecución 2.....</i>	<i>75</i>
<i>Cuadro 3.4: Estado de Servomotores Ejecución 3.....</i>	<i>75</i>
<i>Cuadro 3.5: Estado de Servomotores Ejecución 4.....</i>	<i>76</i>
<i>Cuadro 3.6: Estado de Servomotores secuencia.....</i>	<i>78</i>
<i>Cuadro 3.7: Elementos de Programación.....</i>	<i>81</i>
<i>Cuadro 3.8: Parámetros de configuración WebServer.....</i>	<i>85</i>
<i>Cuadro 3.9: Comandos enviados por celular.....</i>	<i>98</i>
<i>Cuadro 4.1: Parámetros de configuración IP.....</i>	<i>105</i>
<i>Cuadro 4.2: Permitir Acceso en FIREWALL.....</i>	<i>111</i>
<i>Cuadro 4.3: Valores por defecto @4.....</i>	<i>111</i>
<i>Cuadro 4.4: Valores por defecto.....</i>	<i>113</i>
<i>Cuadro 4.5: Direccionamiento IP en las máquinas conectadas vía Wi-Fi.....</i>	<i>114</i>

## ÍNDICE DE FIGURAS

## Capítulo II

Figura 2.1: Niveles de una Red Industria.....	23
Figura 2.2: Esquema general de un sistema automatizado.....	24
Figura 2.3: Estación Maestra.....	26
Figura 2.4: Componentes de un Sistema de Control.....	28
Figura 2.5: Arquitectura de Comunicación Remota WAN.....	29
Figura 2.6: LabVIEW.....	30
Figura 2.7: Compact FieldPoint.....	31
Figura 2.8: Módulo NI cFP-DO-400.....	33
Figura 2.9: Módulo NI cFP-AI-100.....	34
Figura 2.10: Módulo NI cFP-CB-1.....	35
Figura 2.11: LabVIEW 2011.....	36
Figura 2.12: Panel Frontal de un VI.....	39
Figura 2.13: Diagrama de Bloques de un VI.....	40
Figura 2.14: Tools Palette.....	41
Figura 2.15: ConstrolsPalette.....	41
Figura 2.16: FunctionsPalette.....	42
Figura 2.17: Opciones de ejecución.....	43
Figura 2.18: Estructuras.....	44
Figura 2.19: Case Structure.....	45
Figura 2.20: Elementos de Case Structure.....	45
Figura 2.21: Sequence Structure.....	46
Figura 2.22: Elementos del SequenceStructure.....	46
Figura 2.23: Celdas de transmisión.....	48
Figura 2.24: Conector del teléfono Nokia 3220.....	51
Figura 2.25: MicroCode Studio.....	52
Figura 2.26: Programador de Microcontroladores.....	54
Figura 2.27: Imagen de un PIC.....	55
Figura 2.28: Vectir.....	56
Figura 2.29: Ejecución del comando AT.....	59
Figura 2.30: Ejecución del comando AT+CMGF=1.....	59
Figura 2.31: Comando AT+CMSS=1, envía SMS.....	60
Figura 2.32: Comando AT+CMGW, para cargar un SMS en el teléfono.....	61
Figura 2.33: Extracción del SMS mediante el Comando AT+CMGL="STO UNSENT".....	62
Figura 2.34: Conexión de un CFP a una estación.....	63

<i>Figura 2.35: Integración de CFP's a una topología LAN</i> .....	65
<i>Figura 2.36: Partes de un CFP 2020</i> .....	65
<i>Figura 2.37: Partes del Controlador CFP 2020</i> .....	66

### Capítulo III

<i>Figura 3.1: Diagrama De bloques del Sistema</i> .....	72
<i>Figura 3.2: Diagrama Referencial</i> .....	73
<i>Figura 3.3: Panel Frontal y Diagrama de Bloques CTR</i> .....	74
<i>Figura 3.4: Controles en Tiempo Real Ejecución 1</i> .....	75
<i>Figura 3.5: Controles en Tiempo Real Ejecución 2</i> .....	75
<i>Figura 3.6: Controles en Tiempo Real Ejecución 3</i> .....	75
<i>Figura 3.7: Controles en Tiempo Real Ejecución 4</i> .....	76
<i>Figura 3.8: Controles en Tiempo Real, Controles en Tiempo Real</i> .....	76
<i>Figura 3.9: Configuración de Key Navigation</i> .....	76
<i>Figura 3.10: Estado de Servomotores secuencia</i> .....	77
<i>Figura 3.11: Elapsed Time</i> .....	79
<i>Figura 3.12: Configuración Elapsed Time</i> .....	79
<i>Figura 3.13: Diagrama de Bloques Controles Automáticos</i> .....	80
<i>Figura 3.14: Flujograma de envío de Notificaciones</i> .....	81
<i>Figura 3.15: Información de Contactos</i> .....	82
<i>Figura 3.16: Envío de Copia a otro destinatario</i> .....	82
<i>Figura 3.17: Elaboración del Body en la Notificación</i> .....	83
<i>Figura 3.18: Autenticación de la Cuenta del Sistema</i> .....	84
<i>Figura 3.19: Diagrama completo del envío de notificaciones</i> .....	84
<i>Figura 3.20: Opciones de VI en LabVIEW2011</i> .....	85
<i>Figura 3.21: Configuración de WebServer</i> .....	86
<i>Figura 3.22: Publicación Web, selección de VI</i> .....	87
<i>Figura 3.23: Formato HTML de la publicación web</i> .....	88
<i>Figura 3.24: Detalles del archivo HTML creado</i> .....	88
<i>Figura 3.25: Visualización en Internet Explorer</i> .....	89
<i>Figura 3.26: Diagrama de pines PIC 16F628A</i> .....	90
<i>Figura 3.27: Pines de Programación del PIC 16F628A</i> .....	91
<i>Figura 3.28: Nokia 3220</i> .....	93
<i>Figura 3.29: Puerto de comunicación Nokia 3220</i> .....	93
<i>Figura 3.30: Cable DKU – 5</i> .....	94

<i>Figura 3.31: Conexión Celular hacia el PIC</i> .....	94
<i>Figura 3.32: Comando de adquisición de datos VISA</i> .....	95
<i>Figura 3.33: Activación de motores según la lectura del SMS</i> .....	96
<i>Figura 3.34: Diagrama del Circuito externo</i> .....	97
<i>Figura 3.35: Circuito externo de conexión</i> .....	98
<i>Figura 3.36: Diagrama de flujo del funcionamiento del programa</i> .....	100
<i>Figura 3.37: Pantalla principal de Vectir y formas de conexión</i> .....	101
<i>Figura 3.38: Diferentes marcas de teléfonos para trabajar</i> .....	102
<i>Figura 3.39: Escritorio remoto visto en el celular</i> .....	103

#### Capítulo IV

<i>Figura 4.1: Configuración de Dir. IP en Servidor</i> .....	105
<i>Figura 4.2: Verificación de Direccionamiento</i> .....	106
<i>Figura 4.3: Configuración de &lt;&lt;NETWORK Adapter&gt;&gt; en CFP</i> .....	106
<i>Figura 4.4: Mensaje de reinicio del CFP</i> .....	107
<i>Figura 4.5: Disponible nueva configuración de CFP</i> .....	107
<i>Figura 4.6: IPCONFIG en Server</i> .....	108
<i>Figura 4.7: Ping a CFP &lt;&lt;IP: 192.168.6.2&gt;&gt;</i> .....	108
<i>Figura 4.8: Información General del CFP</i> .....	109
<i>Figura 4.9: Módulos Asociados en CFP</i> .....	109
<i>Figura 4.10: CFP asociado y módulos utilizables</i> .....	110
<i>Figura 4.11: Canales inicializados en 0</i> .....	111
<i>Figura 4.12: Configuración Valor &lt;&lt; 1 &gt;&gt;Channel 0</i> .....	112
<i>Figura 4.13: Channel 0 ENCENDIDO</i> .....	112
<i>Figura 4.14: Configuración Valor &lt;&lt; 1 &gt;&gt;Channel 0 @5</i> .....	113
<i>Figura 4.15: Channel 0 @5 ENCENDIDO</i> .....	113
<i>Figura 4.16: Configuración IP en PCI</i> .....	114
<i>Figura 4.17: Verificación de direccionamiento PCI con Wires Shark</i> .....	114
<i>Figura 4.18: Configuración Direccionamiento Celular</i> .....	115
<i>Figura 4.19: Prueba de Conectividad Estación Maestra a PCI</i> .....	115
<i>Figura 4.20: Prueba de Conectividad Pc1 – Wi-Fi</i> .....	115
<i>Figura 4.21: Prueba de Conectividad PCI – CFP</i> .....	116
<i>Figura 4.22: Prueba de Conectividad PCI – Estación Maestra</i> .....	116
<i>Figura 4.23: Prueba de Conectividad PC – Celular</i> .....	117
<i>Figura 4.24: Monitor Remoto desde PCI en IE</i> .....	117

Figura 4.25: Monitoreo Celular.....	118
Figura 4.26: Panel de Control Vía SMS.....	119
Figura 4.27: Recepción de PC.....	119
Figura 4.28: Encendido del CH0 de DO4.....	119
Figura 4.29: Recepción de PA.....	120
Figura 4.30: Encendido del CH1 – DO4.....	120
Figura 4.31: recepción de MI.....	121
Figura 4.32: Activación del CH2 – DO4.....	121
Figura 4.33: Recepción de MD.....	122
Figura 4.34: Activación de CH3 – DO4.....	122
Figura 4.35: Recepción de CS.....	123
Figura 4.36: Activación de CH4 – DO4.....	123
Figura 4.37: Recepción de CB.....	124
Figura 4.38: Activación de CH5 –DO4.....	124
Figura 4.39: Recepción de HS.....	125
Figura 4.40: Activación de CH6 – DO4.....	125
Figura 4.41: Recepción de HB.....	126
Figura 4.42: Activación de CH7 – DO4.....	126
Figura 4.43: Recepción BI.....	127
Figura 4.44: Activación de CH0 – DO5.....	127
Figura 4.45: Recepción de BD.....	128
Figura 4.46: Activación CH1 – DO5.....	128
Figura 4.47: Sub VI Envío de Notificaciones.....	129
Figura 4.48: RECEPCIÓN Bandeja de Entrada “pablo_tenelanda@hotmail.com” .....	130
Figura 4.49: RECEPCIÓN Bandeja de Entrada “pableins666@hotmail.com”.....	131

## **RESUMEN**

Hoy en día se puede encontrar gran cantidad de industrias cuyos objetivos primordiales apuntan a la optimización de tiempo y de recursos por medio de mejores y más sofisticados sistemas de control automático, permitiendo al usuario realizar tareas de mando y supervisión de una manera mas sencilla.

El presente proyecto expone la implementación de un sistema de control remoto automático de un brazo robótico conectado a un Compact Field Point con tarjetas de entradas y salidas de datos, los mismos que componen la estación maestra del sistema de control.

Todos estos equipos mantienen una comunicación constante con otros usuarios mediante una red Ethernet de área local, estos usuarios podrán acceder de manera remota al monitoreo y control de las tareas que se encuentra realizando el brazo robótico.

Utilizando los avances de las telecomunicaciones y específicamente refiriéndose a las bondades que hoy en día tienen los teléfonos celulares se pueden conectar a la red vía Wi – Fi, y con la ayuda de un software como es el Vectir se puede acceder al monitoreo y control del proyecto desde la pantalla del celular.

La tarea de supervisión remota que se hace mediante una PC se realiza gracias a la publicación de una pagina web con la herramienta Web Publishing Tool del LabVIEW la misma que será configurada en el proyecto para que los usuarios puedan ingresar remotamente con Internet Explorer y acceder al control de la interface del software.

Además se ha adaptado a la estación maestra un circuito externo y un dispositivo celular que se encuentra operando normalmente en una compañía telefónica del País, este permitirá recibir instrucciones vía SMS para que sean interpretadas por el software y dar la orden al brazo que realice una acción específica.

## **SUMMARY**

Today we can find lots of industries whose primary objectives aimed at optimizing time and resources through better and more sophisticated automatic control systems, allowing the user to control and monitoring tasks in an easy and reliable. This design presents the implementation of an automatic remote control of a robotic arm connected to a Compact Field Point card data inputs and outputs, the same composing the master station of our control.

All these teams maintain constant communication with other users via a local area Ethernet network, these users can gain remote access to the monitoring and control of the work that is being done by the robotic arm.

Using advances in telecommunications and specifically referring to the benefits that today we have our cell phones connect to our network via Wi - Fi, and with the help of software such as Vectir can connect and access the monitoring and control project from the screen of your phone.

The remote monitoring task is done by a PC is done through the publication of a web page using the Web Publishing Tool of LabVIEW to be configured the same in our project so that users can log in remotely with Internet Explorer and access the control interface of our software.

We have also adapted our external circuit master station and a cellular device is operating normally at a telephone company in the country, this will allow us to receive instructions via SMS to be interpreted by our software and give the command to perform an action arm specified.

## **INTRODUCCIÓN**

El constante crecimiento de las empresas y los avances tecnológicos en cuanto a las telecomunicaciones se refiere brinda hoy en día varios dispositivos con grandes características, los mismos que con el desarrollo de un software adecuado permite a los desarrolladores hacer uso de todos estos elementos para que en conjunto se pueda crear sistemas de control automático más robustos y con mejores cualidades. Siendo estos sistemas amigables con el usuario permitiéndole acceder de manera fácil al monitoreo y control de distintos equipos conectados a la PC.

De la misma manera la creciente demanda de teléfonos celulares con grandes cualidades de interconexión hace que las empresas fabricantes de teléfonos saquen al mercado nuevos equipos con mejores características, similares a una PC normal. Estas características permiten utilizar terminales de este tipo para incorporarlos a proyectos de esta magnitud con el objetivo de tener acceso total al ordenador desde donde se encuentren.

# CAPITULO I

## 1. MARCO REFERENCIAL

### 1.1. Planteamiento del Problema

En la actualidad, uno de los objetivos primordiales en el ámbito empresarial e industrial es optimizar los sistemas de control, permitiendo aligerar tareas de mando y supervisión con un control más sofisticado, complementándose con sistemas de telecomunicaciones y herramientas de visualización que permitan efectivizar los procesos realizados.

Para lo cual se debe tener en cuenta cada una de las herramientas que se tiene a disposición como es el caso de los dispositivos de comunicación celular debido a que su desarrollo es considerable y brinda la posibilidad de tener acceso a redes inalámbricas como por ejemplo Wi – Fi, datos de internet inalámbrico que ofrece la operadora, etc. de esta manera se puede tener acceso y control directo a diversos equipos que se encuentren conectados a la red.

En vista de ello es posible encontrar varias plataformas capaces de permitir al desarrollador construir y elaborar potentes sistemas de cómputo encargados de gestionar no solo datos e información sino además: equipo, maquinaria, herramientas, materiales, etc.

La interacción de herramientas tecnológicas hoy por hoy es un tema de mucha importancia, es por ello que la gran mayoría de soluciones en el campo de la telecomunicaciones basan su éxito en la interacción con uno o varios sistemas en tiempo real, siendo pilares fundamentales de la automatización en el sector industrial ayudando a establecer soluciones prácticas, eficaces y confiables a gran variedad de procesos.

Años atrás las tareas automatizadas de control y visualización se efectuaban fundamentalmente con PLC's (Controladores Lógicos Programables o Autómatas). Esta tecnología fue eficiente por mucho tiempo pero al transcurrir varios años la industria enfrentó cambios realmente drásticos para poder competir con las demás del medio, siendo necesario implementar mayor cantidad de recursos tanto humano, material, tecnológico para abastecer la demanda de un mayor número de consumidores que requieren un servicio o un producto de mayor calidad.

El principal problema que enfrentaban años atrás varios de los sistemas de automatización basados en dispositivos PLC's, era la falta de accesibilidad tanto para tareas de mantenimiento, monitoreo y reparación de daños y errores.

Al momento se puede encontrar gran variedad de sistemas y tecnologías disponibles para proporcionar al entorno industrial sistemas realmente eficientes capaces de controlar y automatizar gran parte de los procesos que son realizados en una planta, las mismas tecnologías que hoy por hoy permiten al usuario controlar y supervisar el normal desenvolvimiento de un equipo en cualquier lugar dentro o fuera de la planta de producción.

## **1.2. FORMULACIÓN DEL PROBLEMA**

La creciente demanda de profesionales con conocimientos de automatización industrial y con ello diseñadores y desarrolladores de sistemas basados en instrumentación virtual(V.I.) desarrollados en LabVIEW y principalmente utilizando la herramienta de paneles remotos hace posible que los profesionales de hoy deban cumplir con cierto nivel de conocimientos aplicables al área industrial, la falta de herramientas tecnológicas incorporadas a los laboratorios de control imposibilita que los estudiantes tengan una percepción clara y precisa de los elementos e infraestructura que debe cumplir un sistema de estas características

Los procesos industriales han sido por décadas un tema de gran interés tanto para investigadores como para los mismos dueños de los recursos, es por ello que se ha planteado un sinnúmero de alternativas para que un proceso sea realizado en el menor tiempo posible y con un alto índice de calidad.

En el sector industrial se pueden encontrar numerosos ejemplos en los cuales el hombre está imposibilitado para realizar y cumplir con ciertos trabajos, sea este el caso de la industria ensambladora de vehículos como automóviles en donde se debe trabajar con partes de la carrocería que son extremadamente grandes y pesadas, o por el contrario el caso de una planta refinadora de petróleo en donde se debe controlar adecuadamente las temperaturas a las cuales se somete el oleoducto para obtener sus derivados como combustibles, puede también ser un ejemplo específico el caso de una industria procesadora de alimentos que deba controlar la temperatura exacta de cocción de la materia prima. Para varias de estas incógnitas la tecnología en primera instancia apuro con circuitos integrados y maquinaria capaz de facilitar gran parte de tareas realizadas por el hombre sin embargo el avance de la informática

y la computación ha posibilitado la creación de potentes sistemas de cómputo capaces de trabajar con maquinaria e instrumentos electrónicos para control y monitoreo de procesos, con el crecimiento desmedido de la industria se vio además de ello la necesidad de que no solo sean ciertos procesos controlados por modernos sistemas de cómputo sino que además estos sistemas sean capaces de recibir, analizar y comparar la información recibida para elaborar posibles estudios de calidad y rendimiento. El aumento de infraestructura y la necesidad de expansión para alcanzar un mayor acaparamiento del mercado obligó a la industria a crear varias plantas de producción y con ello estos sistemas se vieron en la necesidad migrar a plataformas distribuidas que permitieran su acceso ya no a una sola persona sino a varias, clasificándolas por sus funciones y estableciendo permisos para la operatividad de las herramientas controladas.

### **1.3. OBJETIVOS**

#### **1.3.1. General:**

Desarrollar un sistema elaborado en ambiente de programación LabVIEW capaz de controlar y monitorear los movimientos de un brazo robótico desde un dispositivo de comunicación celular GSM y además trabaje sobre una red ETHERNET de área local.

#### **1.3.2. Específicos:**

Diseñar y programar un sistema de monitoreo y control para un brazo robótico en ambiente de programación LabVIEW 2011.

Diseñar un sistema que permita la comunicación entre un celular y la PC para poder controlar el brazo mediante SMS.

Desarrollar un sistema básico de visión a través de las cámaras web, para visualizar los movimientos ejecutados por el brazo, tanto en un computador como en el dispositivo de celular.

### **1.4. JUSTIFICACIÓN**

Hoy en día los avances de las Telecomunicaciones brinda varias herramientas para utilizar y sacar el mayor provecho de cada una de ellas como con los dispositivos de comunicación celular y buscar una aplicación en el campo industrial, es común encontrar a un operador realizando manualmente un trabajo o en su defecto lo vemos frente a una determinada máquina para desempeñar una función o llevar a cabo un proceso, si hablamos de automatización estaremos diciendo que el proceso podrá ejecutarse con o sin la supervisión de un encargado (INDIVIDUO) ya que el proceso se encuentra totalmente realizado por una máquina y a su vez esta monitoreado por un sistema informático el mismo que contempla medidas de detección y corrección de errores.

Con la aplicación de tecnologías como LabVIEW para el desarrollo de aplicaciones industriales sumado a ello la gran variedad de dispositivos compatibles con la plataforma de desarrollo es fácil crear y desarrollar un sistema de control y monitoreo aplicable en una gran cantidad de procesos industriales.

Por otro lado los estudiantes de la escuela de ingeniería en Electrónica y Telecomunicaciones cuentan actualmente con equipos de control para realizar pruebas de este tipo de sistemas, de esta manera la implementación de un proyecto con estas características los alumnos de esta escuela contará con un sistema de pruebas para control automático y monitoreo de elementos electrónicos.

Este proyecto combina gran cantidad de conocimientos y destrezas adquiridas a lo largo de la carrera permitiendo al estudiante crear una herramienta de igual o mejores características que la que se describe en este proyecto.

## **CAPITULO II**

### **2. FUNDAMENTACIÓN TEÓRICA**

#### **2.1. ANTECEDENTES**

El presente proyecto de titulación toma como punto de partida el Diseño e Implementación de un Sistema de Control desarrollado en ambiente de programación

LabVIEW, y los avances de la tecnología en cuanto a telecomunicaciones para monitorear y controlar remotamente los movimientos de un brazo robótico.

Además se incorpora al proyecto un dispositivo de comunicación celular de última generación y comprobar su efectividad para integrarlo a un sistema con estas características para visualizar los movimientos del brazo para de esta manera garantizar así el control del miembro robótico.

Al este sistema se busca convertirlo en una herramienta de telecontrol, debido a que ocuparemos los equipos existentes en la UNACH para dar instrucciones al brazo y que de esta manera pueda ser controlado únicamente con la pc o a su vez mediante mensajes de texto enviados por el celular, además utilizaremos protocolos de comunicación TCP/IP para realizar un enlace ETHERNET para poder acceder desde otros usuarios al sistema.

LabVIEW como plataforma de desarrollo para aplicaciones industriales incorpora un elemento de gran importancia con el nombre "WEB Publishing Tool" que consiste en subir la aplicación desarrollada a un servidor web propio de LabVIEW, permitiendo el acceso a la aplicación a través de Internet Explorer.

## **2.2. ALCANCES DEL PROYECTO**

Para alcanzar el objetivo descrito, es decir desarrollar un sistema que permita monitorear y controlar los movimientos de un brazo robótico se ha considerado lo siguiente:

El diseño de la Interfaz Hombre Maquina (HMI) bajo plataforma LabVIEW versión 2011, desde esta aplicación el usuario podrá ejecutar movimientos en tiempo real y movimientos automatizados previamente establecidos desde la programación.

El estudio de la herramienta WebPublishingTool para alojar aplicaciones desarrolladas en LabVIEW y permitir su posterior visualización y control desde un navegador web como Internet Explorer.

El diseño e implementación de un circuito capaz de establecer comunicación directa entre un celular y la PC capaz de recibir las instrucciones enviadas mediante SMS y sean ejecutadas por el brazo.

### 2.3. ARQUITECTURA DE UN SISTEMA DE CONTROL AUTOMATICO

Un sistema de control automático, limitada antes a comunicar los diferentes dispositivos de campo (transductores y transmisores con actuadores) han ido evolucionando para poder procesar los datos que una planta moderna debe generar para ser competitiva, segura y confiable.

Así mismo, deben desarrollarse para poder satisfacer las necesidades de información que ahora se tiene no solo a nivel de procesos sino también a nivel de gerencia y administración.

*Nivel de Gestión:* Se encarga de integrar los niveles inferiores a una estructura organizada y jerárquica. Las máquinas en este nivel sirven de enlace entre el proceso productivo y el área de gestión, en la cual se requiere información sobre ventas, tiempos de producción, repuestos en bodega, etc. Emplean redes tipo LAN y WAN que funcionan bajo protocolos como Ethernet, por dar un ejemplo.

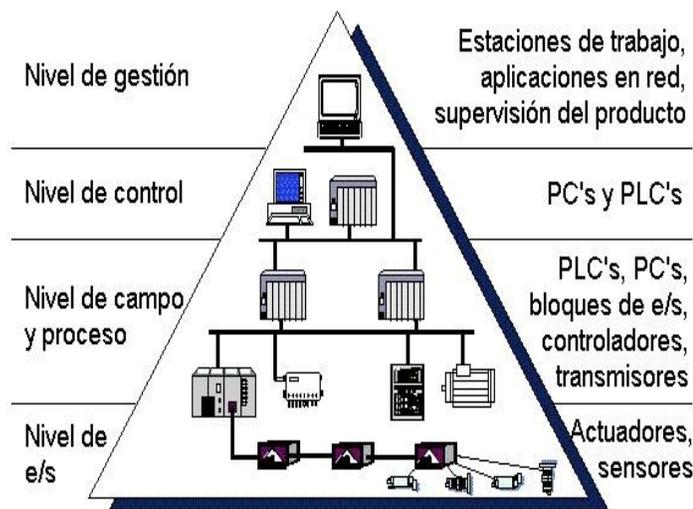
*Nivel de Control:* Se encarga de enlazar y controlar los distintos procesos, líneas de producción de una planta industrial. A este nivel se sitúan los PLC's de gran desempeño y poder, así como computadoras destinadas a diseño, control de calidad, programación. Suelen emplear redes tipo LAN que funcionan bajo el protocolo Ethernet.

*Nivel de Campo y Proceso:* Aquí se realiza la integración de la información generada y requerida por los procesos de campo automáticos y controlados que utilizan PLC's y controladores, multiplexores de Entrada/Salida (I/O) que están conectados en subsistemas. Aquí es frecuente encontrar uno o varios autómatas modulares, actuando como maestros. En este nivel se emplean los buses o redes industriales de campo que funcionan bajo protocolos como Fieldbus, Profibus, por mencionar algunos.

*Nivel de I/O:* Es el nivel más próximo a las variables físicas de la planta. Aquí se hallan los sensores (transmisores) y actuadores encargados de medir y controlar los procesos productivos, respectivamente. Basados en la información que se recoge en

este nivel. Aplicaciones de control toman las decisiones necesarias que garanticen una correcta automatización y supervisión.

Una planta industrial organizada de esta forma requiere de sistemas: CONTROL, DCS, Multiplexores y HMI's. Vale indicar que un diagrama representativo de una planta organizada de esta forma suele parecer ser simples líneas que unen un bus con otro, ocultando la verdadera complejidad que en realidad existe. Desde ya debe entenderse que la unión no es solamente asunto de unir eléctricamente un bus con otro, sino lograr que dos protocolos diferentes puedan comunicarse entre sí. En la siguiente imagen se muestran los niveles de aplicación de un sistema de control.



*Figura 2.1: Niveles de una Red Industria <sup>1</sup>*

### 2.3.1. DCS.

El término DCS, viene de las siglas Distributed Control System, es un sistema de control que cumple con funciones de control a través de una serie de módulos de control automáticos e independientes, distribuidos en una planta o proceso.

La filosofía de funcionamiento de esta arquitectura es evitar que el control de toda la planta esté centralizado en una sola unidad, que es lo que se busca en sistema de este tipo. De esta forma, si una unidad de control falla, el resto de unidades podría seguir funcionando.

---

<sup>1</sup> <http://www.profesores.frc.utn.edu.ar/industrial/sistemasinteligentes/UT5/INTRODUCCI%C3%93N%20GENERAL.html>.

Los sistemas DCS se desarrollan sobre la base de dispositivos de control, tales como Controladores o PLC's, que se fundamenta en un programa de control encargado de tomar decisiones dependiendo de los datos que reciben en sus entradas. Las decisiones son enviadas hacia actuadores que son los que se encargan de mantener las variables del proceso bajo control en los valores establecidos.

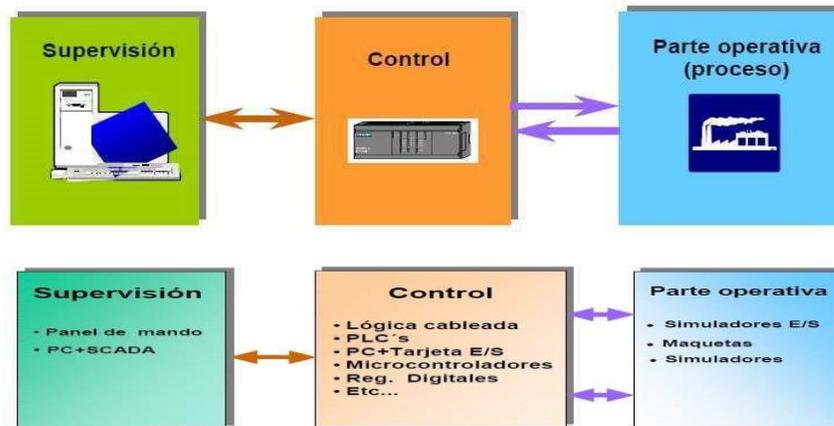
ASPECTO	DCS
TIPO DE ARQUITECTURA	DISTRIBUÍDA
TIPO DE CONTROL PREDOMINANTE	REGULATORIO: Lazos de control cerrados automáticamente por el sistema. Adicionalmente: control secuencial, batch, algoritmos avanzados, etc.
TIPOS DE VARIABLES	ACOPLADAS
ÁREA DE ACCIÓN	Área de la planta.
UNIDADES DE ADQUISICIÓN DE DATOS Y CONTROL	Controladores de lazo, PLCs.
MEDIOS DE COMUNICACIÓN	Redes de área local, conexión directa.
BASE DE DATOS	DISTRIBUÍDA

*Cuadro 2.1: Esquema de un sistema DCS*

#### **2.4. CRITERIOS DE DISEÑO DE UN SISTEMA DE CONTROL AUTOMÁTICO**

Los sistemas de control automático son aquellos dedicados a obtener la salida deseada de un sistema o proceso. En un sistema general se tienen una serie de entradas que provienen del sistema a controlar, llamado planta, y se diseña un sistema para que, a partir de estas entradas, modifique ciertos parámetros en el sistema planta, con lo que las señales anteriores volverán a su estado normal ante cualquier variación.

Un sistema de control básico es mostrado en la siguiente figura:



*Figura 2.2: Esquema general de un sistema automatizado*

### 2.4.1. Características de un sistema Automatizado

Ser un sistema de arquitectura abierta, capaz de crecer o adaptarse según las necesidades del usuario.

Poder comunicarse con total facilidad, y de forma transparente al usuario, con el módulo y con el resto del mundo.

Tener programas sencillos de instalar, sin excesivas exigencias de hardware, y fáciles de utilizar para el usuario.

### 2.4.2. Funciones Principales

Adquisición de datos, se lo hace mediante el protocolo de comunicación I/O Server Mbenet, y se almacena en los logs del Intouch.

Supervisión, se lo realiza a nivel local o remoto a través de la internet.

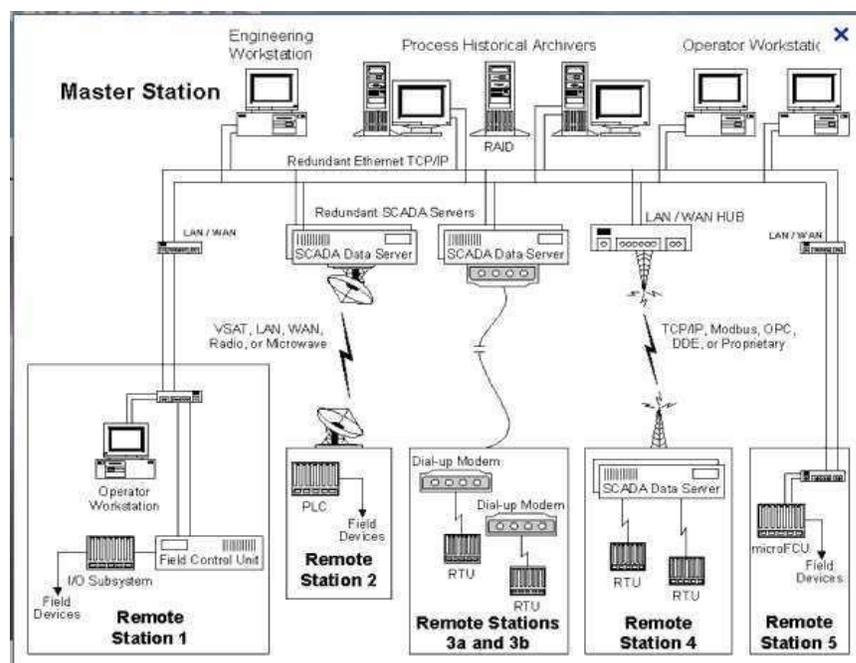
Control, se puede modificar la evolución del proceso, actuando directamente sobre las salidas conectadas.

### 2.4.3. Estación Maestra

El término "Estación Maestra" se refiere a los servidores y al software responsable para comunicarse con el equipo del campo (RTUs, PLCs, etc) en estos se encuentra el software HMI corriendo para las estaciones de trabajo en el cuarto de control, o en

cualquier otro lado. En un sistema de control pequeño, la estación maestra puede estar en un solo computador. A gran escala, en los sistemas de control la estación maestra puede incluir muchos servidores, aplicaciones de software distribuido, y sitios de recuperación de desastres.

El sistema de control usualmente presenta la información al personal operativo de manera gráfica, en forma de un diagrama de representación. Esto significa que el operador puede ver un esquema que representa la planta que está siendo controlada. Por ejemplo un dibujo de una bomba conectada a la tubería puede mostrar al operador cuanto fluido está siendo bombeado desde la bomba a través de la tubería en un momento dado o bien el nivel de líquido de un tanque o si la válvula está abierta o cerrada. Los diagramas de representación pueden consistir en gráficos de líneas y símbolos esquemáticos para representar los elementos del proceso, o pueden consistir en fotografías digitales de los equipos sobre los cuales se animan las secuencias.



*Figura 2.3: Estación Maestra*<sup>2</sup>

<sup>2</sup> <http://scada-tx-de-datos.wikispaces.com/>

#### **2.4.3.1. Características**

Configuración: permite definir el entorno de trabajo de un sistema de control, adaptándolo a la aplicación particular que se desea desarrollar.

Interfaz gráfica del operador: proporciona al operador las funciones de control y supervisión de la planta, permitiendo al usuario tener un control total sobre los procesos que realiza el sistema.

Módulo de proceso: ejecuta las acciones de mando pre-programadas a partir de los valores actuales de variables leídas. Gestión y archivo de datos: almacenamiento y procesado ordenado de datos, de forma que otra aplicación o dispositivo pueda tener acceso a ellos.

Comunicaciones: transferencia de información entre la planta y la arquitectura de hardware que soporta el Sistema de Control, y también entre ésta y el resto de elementos informáticos de gestión. El paquete HMI para el sistema de control típicamente incluye un programa de dibujo con el cual los operadores o el personal de mantenimiento del sistema pueden cambiar la apariencia de la interfaz. Estas representaciones pueden ser tan simples como unas luces de tráfico en pantalla, las cuales representan el estado actual de un campo en el tráfico actual, o tan complejas como una pantalla de multiproyector representando posiciones de todos los elevadores en un rascacielos o todos los trenes de una vía férrea.

Plataformas abiertas como GNU/Linux que no eran ampliamente usados inicialmente, se usan debido al ambiente de desarrollo altamente dinámico y porque un cliente que tiene la capacidad de acomodarse en el campo del hardware y mecanismos a ser controlados que usualmente se venden UNIX o con licencias OpenVMS. Hoy todos los grandes sistemas son usados en los servidores de la estación maestra así como en las estaciones de trabajo HMI.

#### **2.4.3.2. Infraestructura y Métodos de Comunicación**

Los sistemas de control tienen tradicionalmente una combinación de radios y señales directas seriales o conexiones de módem para conocer los requerimientos de comunicaciones, incluso Ethernet e IP sobre SONET (fibra óptica) es también frecuentemente usada en sitios muy grandes como ferrocarriles y estaciones de

energía eléctrica. Es más, los métodos de conexión entre sistemas pueden incluso que sea a través de comunicación wireless (por ejemplo si queremos enviar la señal a una PDA, a un teléfono móvil,...) y así no tener que emplear cables.



Figura 2.4: Componentes de un Sistema de Control<sup>3</sup>

#### 2.4.4. Arquitectura de red

Arquitecturas distribuidas cliente/servidor pueden ser configuradas usando protocolos TCP/IP. Estructuras multi maestro en las cuales cada terminal puede comunicarse con los otros equipos de tal manera que le es posible la lectura y la escritura.

Tanto las tecnologías inalámbricas como las redes celulares pueden ser usadas para comunicación con dispositivos remotos o enviar SMS (Short Message Service) a los operadores de mantenimiento; de esta manera el centro de supervisión puede proveer un servicio de asistencia remota a los sitios perimetrales alrededor del País.

<sup>3</sup> <http://www.tecnologiajavier.es/4eso/t04robotica/04aplicaciones.html>

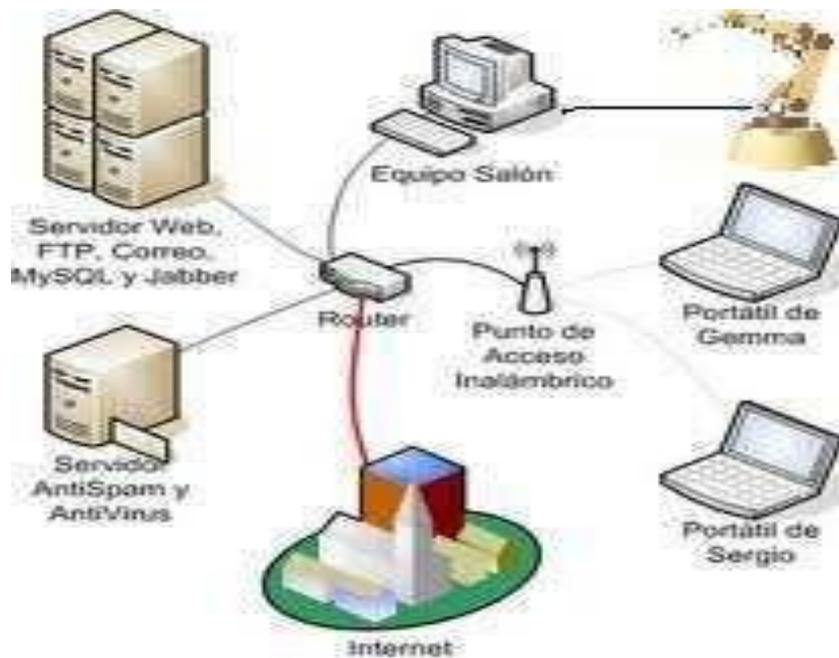


Figura 2.5: Arquitectura de Comunicación Remota WAN

#### 2.4.5. Notificaciones

Notificaciones pueden ser activadas como función de valores límites tomados por el sistema de control. Las alarmas son señaladas directamente en un área especialmente reservada de la pantalla, y la confirmación por el operador es también controlada. El estado de información (eventos activos) e historial (dato, tiempo, evento) pueden ser ingresadas según varias categorías libremente definidas (Ej. Prioridad y lugar).

#### 2.4.6. Procedimientos

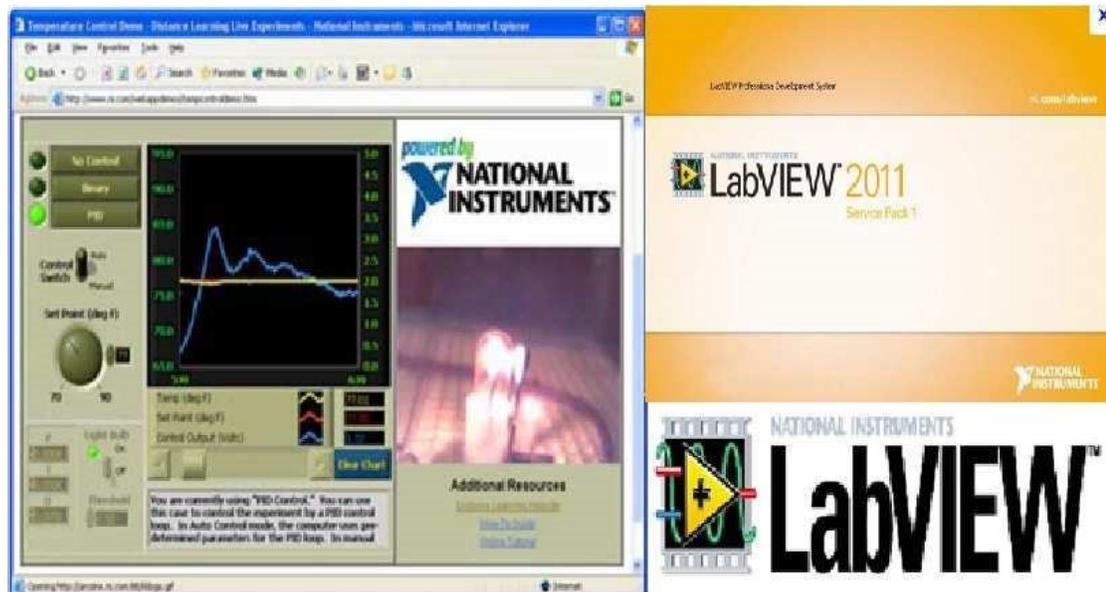
Los procedimientos son sets de características, valores de un proceso de trabajo configuración de la planta. Los nuevos procedimientos pueden ser creados y existen unos con actualizaciones (copiados, borrados, renombrados, impresos y modificados) así como los que son importados y exportados del proceso en sí.

La aplicación de un procedimiento puede ser comandado por un operador, ejecutado automáticamente en el arranque o ejecutados por el programa.

#### 2.4.7. Plataforma utilizada en el desarrollo de aplicaciones industriales

### 2.4.7.1. LabVIEW <sup>4</sup>

LabVIEW, como en el resto de los lenguajes de programación se basa en el concepto de programación modular descansa sobre el elemento función. Esto quiere decir que la programación modular se basa en la programación con funciones, osea que la función es la base de la programación modular. Muchas de estas funciones son proporcionadas por el propio fabricante de software, y se presentan ordenadas por tipos en las librerías de funciones. Estas librería pueden ser vistas si se pulsa con el botón derecho del ratón sobre cualquier punto del diagrama de la aplicación, a continuación se pulsa en la opción final del menú flotante (doble ^ o extensión), con lo que se muestra el listado de las librerías de funciones. La mayoría de las funciones que utilizaremos en este curso, serán las de la librería *Programming* (programación).  
Figura xxx Interfaz de LabVIEW.



*Figura 2.6: LabVIEW*

Los diferentes programas hacen posible configurar una arquitectura cliente/servidor con protocolo TCP/IP o red intranet/Internet o crear aplicaciones web accesibles desde buscadores estándar.

## 2.5. TECNOLOGÍAS A UTILIZAR

### 2.5.1. Instrumentación de Control

<sup>4</sup> <http://www.ni.com/labview/>

Se denominan instrumentos de control a todos los dispositivos eléctricos o electrónicos que actúan como parte esencial en la arquitectura de un sistema de control automático, estos instrumentos de control son la plataforma física de conexión entre los elementos de campo: sensores, actuadores, relés, etc, suministrando conexión con la estación maestra que alberga la aplicación desde la cual se administran los recursos.

Como principal instrumento de control en el desarrollo del prototipo tenemos a un CFP Modelo 2020 adquirido por la Facultad de Ingeniería y que permanece en los laboratorios para realización de prácticas de laboratorio.

### 2.5.1.1. Compact FieldPoint Modelo 2020

#### Compact Field Point

Es un controlador de automatización programable (PAC) fácil de usar y altamente expansible compuesto de módulos de E/S e interfaces de comunicación inteligentes. Los módulos de E/S filtran, calibran y escalan señales de sensores sin procesar a unidades de ingeniería así como realizar auto diagnósticos para encontrar problemas y publica automáticamente medidas en la red Ethernet.



*Figura 2.7: Compact FieldPoint*

Es un controlador de automatización programable (PAC) fácil de usar y altamente expansible compuesto de módulos de E/S robustos e interfaces de comunicación inteligentes. Los módulos de E/S del Compact FieldPoint filtran, calibran y escalan señales de sensores sin procesar a unidades de ingeniería así como realizar auto diagnóstico para encontrar problemas como, por ejemplo, un termopar abierto. A través de sus servidores integrados de Web y archivos, la interfaz Compact FieldPoint publica automáticamente medidas en la red Ethernet.

Como características fundamentales tenemos:

Despliegue de controladores embebidos en tiempo real para registro de datos autónomos, análisis avanzado y control de procesos.

Obtención de acceso a puntos de E/S cercanos o remotos en la red usando la misma simple estructura de software de lectura y escritura.

Facilidad de conexión de cualquier sensor directamente a la amplia variedad de módulos de E/S analógicos y digitales de alta precisión.

Fácil integración de la aplicación NI LabVIEW al controlador embebido para una operación confiable y autónoma.

Otorga herramientas de desarrollo de sistemas de control o de mediciones fáciles y simples de utilizar. Cumple con estándares internacionales de seguridad y resistencia industrial tal como: aislamiento para 2300Vrms, rango de tolerancia de temperatura entre 40° C hasta 70° C y montaje en riel. Tiene módulos E/S inteligentes, que se pueden escoger de acuerdo con las necesidades.

Acceso vía HTTP a través de un servidor Web comunicándose a 10 y 100 Mbps en Ethernet.

Tolerancia a la vibración de 5g de vibración e inmunidad al ruido eléctrico.

#### **2.5.1.2. Módulo NI cFP-DO-400 (Digital I/O)**

El cFP-DO-400 de National Instruments cuenta con ocho canales de entrada o salida de señales digitales. Cada canal es compatible con voltajes de 5 a 30 VDC y puede proporcionar hasta 2 A por canal con un máximo de 9 A cuadrados por módulo. (La suma de los cuadrados de las corrientes de salida de los ocho canales no debe ser superior a 9.) Cada canal tiene un LED para indicar el canal de encendido / apagado. El módulo cuenta con 2300 V de aislamiento de tránsito entre los canales de salida y la placa madre. Para hacer la instalación y configuración más fácil la NI cFP-DO-400 cuenta con HotPnP (plug and play) la operación y es automáticamente detectado e identificado por el software de configuración.

Especificaciones Técnicas:

Se monta al plano trasero en medio de los módulos de E/S.

Se requiere un cFP-CB-1 para cada módulo de E/S Compact FieldPoint. Liberación de tensión y ranuras integradas para ataduras de cables.

Facilidad de cableado con las terminales con codificación en colores para conexiones de voltaje y conexiones comunes.



Figura 2.8: Módulo NI cFP-DO-400

### **2.5.1.3. Módulo NI cFP-AI-100 (Analog I/O)**

El cFP-AI-100 de National Instruments es un módulo de entrada para 8 canales de una sola terminal con fines de medición directa de mili voltios, baja tensión, media tensión, o miliamperios, señales de corriente de una variedad de sensores y transductores. El NI cFP-AI-100 ofrece entradas analógicas de alta velocidad (2,8 actualización ms) con una resolución de 12 bits, y las características Superación, HotPnP (plug and play), operación y diagnóstico a bordo para asegurar una instalación libre de problemas y mantenimiento, viene con un certificado de calibración NIST con el objetivo de garantizar que las medidas analógicas sean precisas y confiables.

Especificaciones Técnicas:

8 entradas de voltaje o corriente

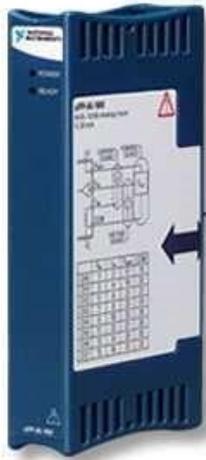
Mide señales de corriente de mili voltios, bajo voltaje, voltaje medio o miliamperios

Resolución de 12 bits

Operación Hot PnP (plug-and-play)

Rango de operación de -40 a 70 °C

Rango de entrada configurable por software (por canal)



*Figura 2.9: Módulo NI cFP-AI-100*

#### **2.5.1.4. Módulo NI cFP-CB-1**

El Cfp-CB-1 es un elemento que permite la entrada y salida de señales, a través de cables hacia el dispositivo Compact FieldPoint. El Cfp-CB-1 se monta en la placa madre del CFP y ofreciendo una manera fácil y conveniente de conectar sensores y actuadores. El bloque de conectores viene con dos barras de alivio de tensión - uno para hilos individuales y uno para cables más gruesos. El cFP-CB-1 tiene ranuras para los lazos de alambre para cables de más tamaño y cuenta con terminales codificados por color para facilitar el cableado, además cuenta con un termistor para compensación de unión fría cuando se utiliza con termopares.

#### **Especificaciones Técnicas:**

Se monta al plano trasero en medio de los módulos de E/S.

Se requiere un cFP-CB-1 para cada módulo de E/S Compact FieldPoint.

Liberación de tensión y ranuras integradas para ataduras de cables.

Facilidad de cableado con las terminales con codificación en colores para conexiones de voltaje y conexiones comunes.



*Figura 2.10: Módulo NI cFP-CB-1*

## 2.5.2. Software

### 2.5.2.1. LabVIEW 2011 <sup>5</sup>



*Figura 2.11: LabVIEW 2011 <sup>6</sup>*

Este programa fue creado por National Instruments (1976) para funcionar sobre máquinas MAC, salió al mercado por primera vez en 1986. Ahora está disponible para

<sup>5</sup>

<http://www.ni.com/labview/>

<sup>6</sup> <http://electronbits.blogspot.com/2011/03/curso-de-sistemas-de-medida-y.html>

las plataformas Windows, UNIX, MAC y Linux. La versión actual 8.6, publicada en agosto de 2008, cuenta también con soporte para Windows Vista.

Los programas desarrollados con LabVIEW se llaman Instrumentos Virtuales o Vi's, y su origen provenía del control de instrumentos, aunque hoy en día se ha expandido ampliamente no sólo al control de todo tipo de electrónica (Instrumentación electrónica) sino también a su programación embebida. Un lema tradicional de LabVIEW es: "La potencia está en el Software", que con la aparición de los sistemas multinúcleo se ha hecho aún más patente. Entre sus objetivos están el reducir el tiempo de desarrollo de aplicaciones de todo tipo (no sólo en ámbitos de Pruebas, Control y Diseño) y el permitir la entrada a la informática a profesionales de cualquier otro campo. LabVIEW consigue combinarse con todo tipo de software y hardware, tanto del propio fabricante -tarjetas de adquisición de datos, PAC, Visión, instrumentos y otro Hardware- como de otros fabricantes.

LabVIEW es una herramienta gráfica para pruebas, control y diseño mediante la programación. El lenguaje que usa se llama lenguaje G, donde la G simboliza que es lenguaje Gráfico, es un entorno de programación destinado al desarrollo de aplicaciones, similar a los sistemas de desarrollo comerciales que utilizan el lenguaje C o BASIC. La diferencia de dichos programas en un importante aspecto es: los citados lenguajes de programación se basan en líneas de texto para crear el código fuente del programa en cambio LabVIEW emplea la programación gráfica para crear programas basados en diagramas de bloques.

Sin embargo, LabVIEW se, mientras que Para el empleo de LabVIEW no se requiere gran experiencia en programación, ya que se emplean iconos, términos e ideas familiares a científicos e ingenieros, y se apoya sobre símbolos gráficos en lugar de lenguaje escrito para construir las aplicaciones. Por ello resulta mucho más intuitivo que el resto de lenguajes de programación convencionales. LabVIEW posee extensas librerías de funciones y subrutinas. Además de las funciones básicas de todo lenguaje de programación, LabVIEW incluye librerías específicas para la adquisición de datos, control de instrumentación VXI, GPIB y comunicación serie, análisis presentación y guardado de datos. LabVIEW también proporciona potentes herramientas que facilitan la depuración de los programas.

Las ventajas que proporciona el empleo de LabVIEW se resumen en las siguientes:

Se reduce el tiempo de desarrollo de las aplicaciones al menos de 4 a 10 veces, ya que es muy intuitivo y fácil de aprender.

Dota de gran flexibilidad al sistema, permitiendo cambios y actualizaciones tanto del hardware como del software.

Da la posibilidad a los usuarios de crear soluciones completas y complejas.

Con un único sistema de desarrollo se integran las funciones de adquisición, análisis y presentación de datos.

El sistema está dotado de un compilador gráfico para lograr la máxima velocidad de ejecución posible.

Tiene la posibilidad de incorporar aplicaciones escritas en otros lenguajes.

#### **2.5.2.1.1. ¿Cómo Trabaja LabVIEW?**

Los programas desarrollados mediante LabVIEW se denominan Instrumentos Virtuales (VI's), porque su apariencia y funcionamiento imitan los de un instrumento real. Sin embargo son análogos a las funciones creadas con los lenguajes de programación convencionales. Los VI's tienen una parte interactiva con el usuario y otra parte de código fuente, y aceptan parámetros procedentes de otros VI's.

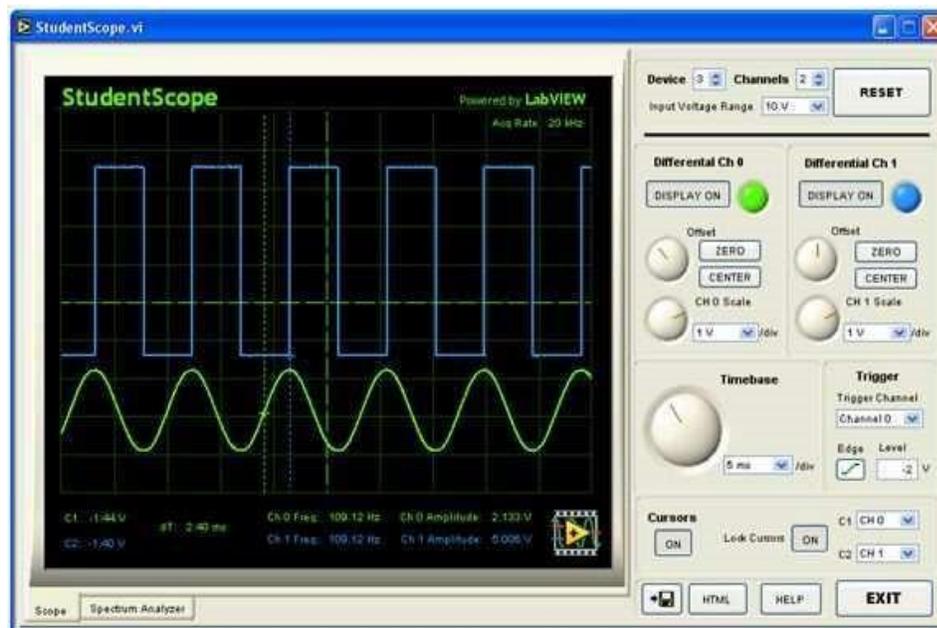
Todos los VI's tienen un panel frontal y un diagrama de bloques. Las paletas contienen las opciones que se emplean para crear y modificar los VI's. A continuación se procederá a realizar una somera descripción de estos conceptos.

#### **2.5.2.1.2. Panel Frontal**

Se trata de la interfaz gráfica del VI con el usuario. Esta interfaz recoge las entradas procedentes del usuario y representa las salidas proporcionadas por el programa. Un panel frontal está formado por una serie de botones, pulsadores, potenciómetros, gráficos, etc.

Cada uno de ellos puede estar definido como un control (a) o un indicador (b). Los primeros sirven para introducir parámetros al VI, mientras que los indicadores se

emplean para mostrar los resultados producidos, ya sean datos adquiridos o resultados de alguna operación.



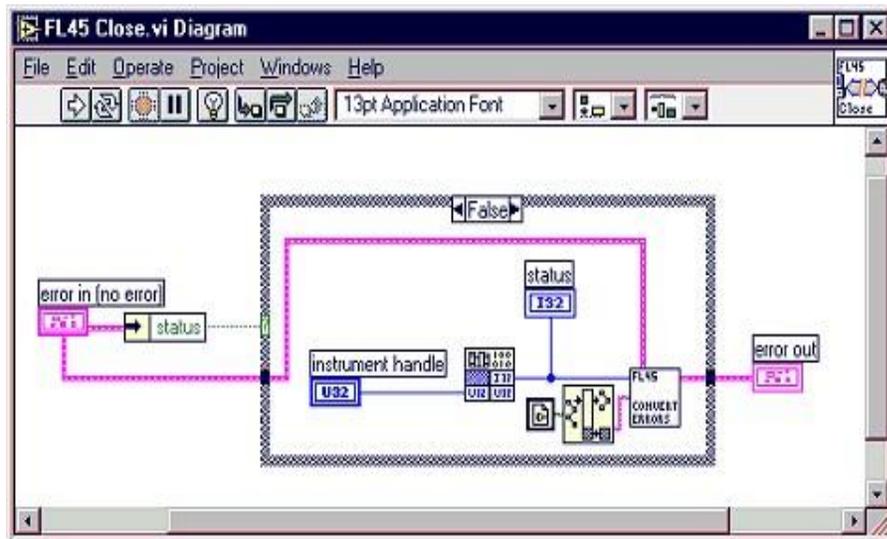
*Figura 2.12: Panel Frontal de un VI*

### 2.5.2.1.3. Diagrama de Bloques

El diagrama de bloques constituye el código fuente del VI. En el diagrama de bloques es donde se realiza la implementación del programa del VI para controlar o realizar cualquier procesamiento de las entradas y salidas que se crearon en el panel frontal.

El diagrama de bloques incluye funciones y estructuras integradas en las librerías que incorpora LabVIEW. En el lenguaje G las funciones y las estructuras son nodos elementales. Son análogas a los operadores o librerías de funciones de los lenguajes convencionales.

Los controles e indicadores que se colocaron previamente en el Panel Frontal, se materializan en el diagrama de bloques mediante los terminales. A continuación se presenta un ejemplo de lo recién citado:



*Figura 2.13: Diagrama de Bloques de un VI*

El diagrama de bloques se construye conectando los distintos objetos entre sí, como si de un circuito se tratara. Los cables unen terminales de entrada y salida con los objetos correspondientes, y por ellos fluyen los datos.

#### 2.5.2.1.4. Paletas.

Las paletas de LabVIEW proporcionan las herramientas que se requieren para crear y modificar tanto el panel frontal como el diagrama de bloques. Existen las siguientes paletas:

#### 2.5.2.1.5. Paleta de herramientas (Tools palette)

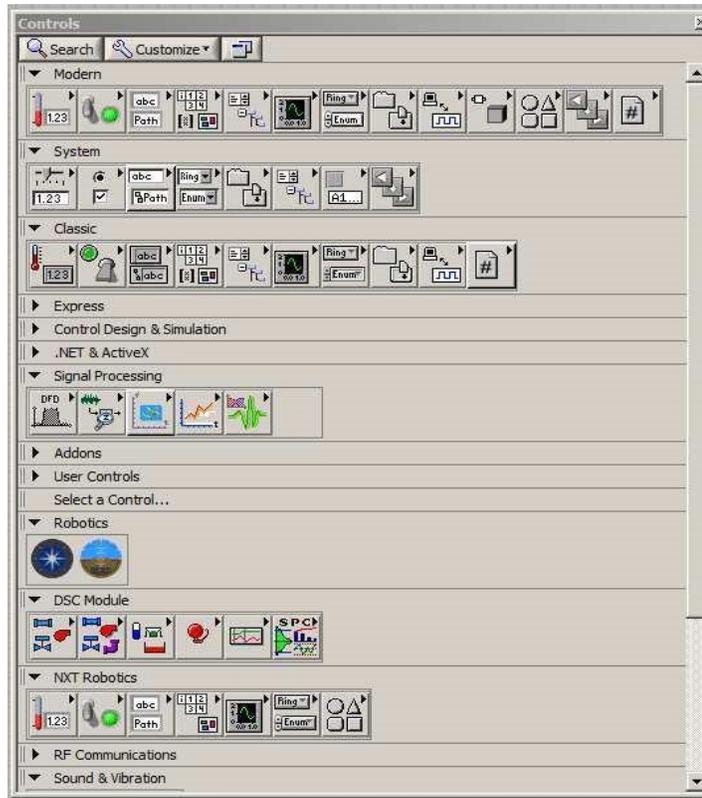
Se emplea tanto en el panel frontal como en el diagrama de bloques. Contiene las herramientas necesarias para editar y depurar los objetos tanto del panel frontal como del diagrama de bloques.



*Figura 2.14: Tools Palette*

### 2.5.2.1.6. Paleta de controles (Controlspalette)

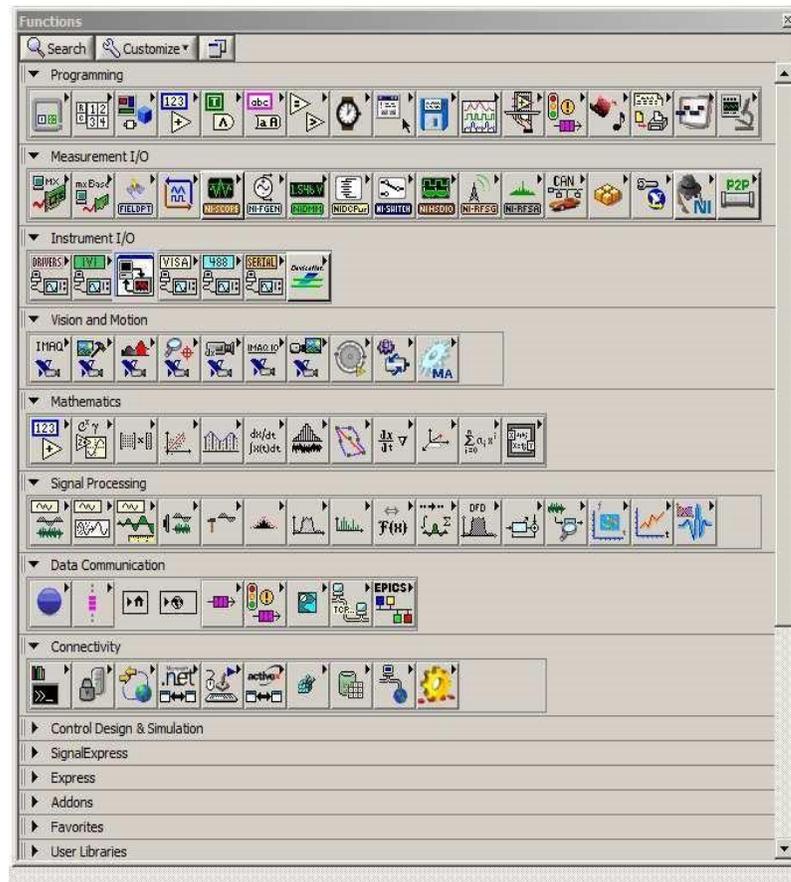
Se utiliza únicamente en el panel frontal. Contiene todos los controles e indicadores que se emplearán para crear la interfaz del VI con el usuario.



*Figura 2.15: ConstrolsPalette*

### 2.5.2.1.7. Paleta de Funciones (functionspalette)

Se emplea en el diseño del diagrama de bloques. La paleta de funciones contiene todos los objetos que se emplean en la implementación del programa del VI, ya sean funciones aritméticas, de entrada/salida de señales, entrada/salida de datos a fichero, adquisición de señales, temporización de la ejecución del programa.



*Figura 2.16: FunctionsPalette*

### 2.5.2.1.8. Programación en Labview

Con el entorno gráfico de programación de LabVIEW se comienza a programar a partir del panel frontal. En primer lugar se definirán y seleccionarán de la paleta de controles todos los controles (entradas que dará el usuario) e indicadores (salidas que presentará en pantalla el VI) que se emplearán para introducir los datos por parte del usuario y presentar en pantalla los resultados.

Para facilitar la tarea de conexión de todos los terminales, en el menú “Help” puede elegirse la opción “Show Help”, con lo que al colocar el cursor del ratón sobre un elemento aparece una ventana con información relativa a éste (parámetros de entrada y salida). Además, si se tiene seleccionado el cursor de cableado, al situar éste sobre un elemento se muestran los terminales de forma intermitente.

### 2.5.2.1.9. Ejecución de un VI

Una vez se ha concluido la programación del VI se debe proceder a su ejecución. Para ello la ventana activa debe ser el panel frontal (si se está en la ventana del diagrama de bloques, se debe seleccionar la opción Show Panel de la barra de menú). Una vez situados en el panel frontal, se pulsará el botón de Run, situado en la barra de herramientas.



*Figura 2.17: Opciones de ejecución*

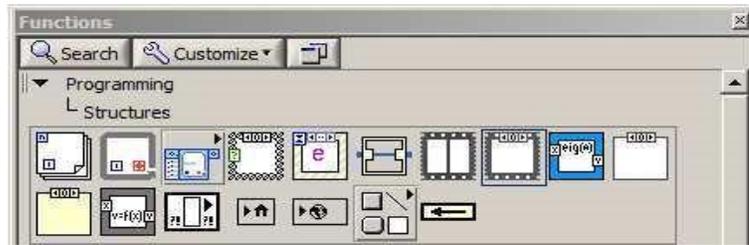
El programa comenzará a ejecutarse. Mientras dura la ejecución del mismo, la apariencia del botón de Runes la que se muestra a continuación:

De este modo el programa se ejecutará una sola vez. Si se desea una ejecución continua, se pulsará el botón situado a la derecha del de Run(ContinuousRun). Si durante el funcionamiento continuo del programa se vuelve a pulsar el citado botón, se finalizará la última ejecución del mismo, tras lo cual el programa se parará.

Para finalizar la ejecución de un programa se puede operar de dos formas. La primera, y la más aconsejable, es emplear un botón en el panel frontal del VI, cuya pulsación produzca la interrupción del bucle de ejecución de la aplicación. La segunda forma de detener la ejecución del VI es pulsando el botón de pausa o el de stop. La diferencia entre ambos es que si se pulsa stop, la ejecución del programa finaliza inmediatamente, mientras que si se pulsa pausa, se produce una detención en el funcionamiento del programa, retomándose su ejecución una vez se vuelve a pulsar el mismo botón.

#### **2.5.2.1.10. Estructuras**

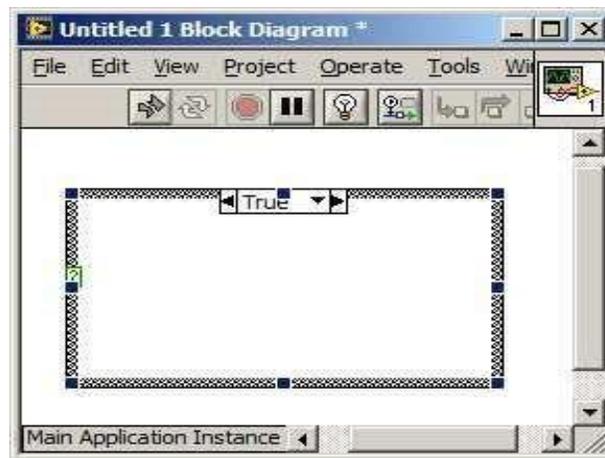
En la paleta de funciones la primera opción es la de las estructuras. Éstas controlan el flujo del programa, bien sea mediante la secuenciación de acciones, ejecución de bucles, etc.



*Figura 2.18: Estructuras*

Las estructuras se comportan como cualquier otro nodo en el diagrama de bloques, ejecutando automáticamente lo que está programada en su interior una vez tiene disponibles los datos de entrada, y una vez ejecutadas las instrucciones requeridas, suministran los correspondientes valores a los cables unidos a sus salidas. Sin embargo, cada estructura ejecuta su subdiagrama de acuerdo con las reglas específicas que rigen su comportamiento, y que se especifican a continuación.

Un subdiagrama es una colección de nodos, cables y terminales situados en el interior del rectángulo que constituye la estructura. El For Loop y el While Loop únicamente tienen un subdiagrama. El Case Structure y el Sequence Structure, sin embargo, pueden tener múltiples subdiagramas, superpuestos como si se tratara de cartas en una baraja, por lo que en el diagrama de bloques únicamente será posible visualizar al tiempo uno de ellos. Los subdiagramas se construyen del mismo modo que el resto del programa. Las siguientes estructuras se hallan disponibles en el lenguaje G.



*Figura 2.19: Case Structure*

Al igual que otras estructuras posee varios subdiagramas, que se superponen como si de una baraja de cartas se tratara. En la parte superior del subdiagrama aparece el identificador del que se está representando en pantalla. A ambos lados de este

identificador aparecen unas flechas que permiten pasar de un subdiagrama a otro. En este caso el identificador es un valor que selecciona el subdiagrama que se debe ejecutar en cada momento.

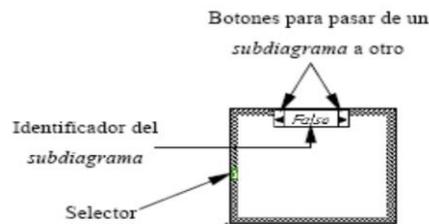


Figura 2.20: Elementos de Case Structure

La estructura Case tiene al menos dos subdiagramas (True y False). Únicamente se ejecutará el contenido de uno de ellos, dependiendo del valor de lo que se conecte al selector.

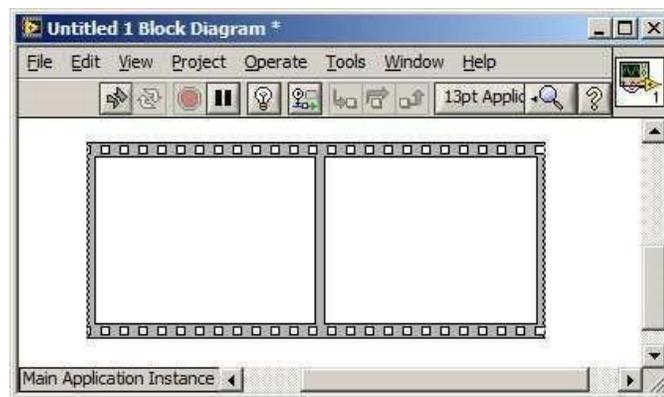


Figura 2.21: Sequence Structure

De nuevo, este tipo de estructuras presenta varios subdiagramas, superpuestos como en una baraja de cartas, de modo que únicamente se puede visualizar una en pantalla. También poseen un identificador del subdiagrama mostrado en su parte superior, con posibilidad de avanzar o retroceder a otros subdiagramas gracias a las flechas situadas a ambos lados del mismo

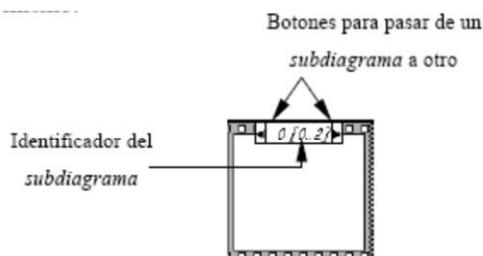


Figura 2.22: Elementos del Sequence Structure

Esta estructura secuencia la ejecución del programa. Primero ejecutará el subdiagrama de la hoja (frame) nº0, después el de la nº 1, y así sucesivamente. Para pasar datos de una hoja a otra se pulsará el botón derecho del ratón sobre el borde de la estructura, seleccionando la opción Add sequence local.

#### **2.5.2.1.11. Arquitectura de Software para Instrumentos Virtuales (VISA)**

VISA es una librería de interfaz simple para controlar VXI, GPIB, RS-232, y otros tipos de instrumentos en las plataformas de LabVIEW. VISA es el estándar utilizado por la Alianza de Sistemas de VXI plug & play, que incluye más de 35 de las compañías más grandes en la industria de instrumentación.

El estándar VISA unifica la industria para hacer software que pueda ser interpretado y reusado por más tiempo, sin importar el tipo de operación de su instrumento. NI-VISA es un software controlador que está disponible en la página web del fabricante.

#### **2.5.2.2. Sistema de transmisión de datos por red celular**

##### **2.5.2.2.1. Telefonía celular**

La telefonía celular permite tener una comunicación en tiempo real, transmitiendo voz y datos, gracias a la velocidad con la que viaja la información por las redes existentes que día a día ofrecen mas y eficientes beneficios para los usuarios.

La red de telefonía móvil celular consiste en un sistema telefónico en la que mediante la combinación de una red de estaciones transmisoras – receptoras de radio llamadas estaciones base y una serie de centrales telefónicas de conmutación, se posibilita la comunicación entre terminales telefónicas portátiles y teléfonos de la red fija tradicional.

La telefonía móvil celular se basa en un sistema de áreas de transmisión llamadas células o celdas, que abarcan grandes áreas comprendidas entre 1,5 y 5 km, dentro de las cuales existen una o varias estaciones repetidoras que trabajan con una determinada frecuencia, que debe ser diferente de la célula circundantes.

En la siguiente figura se puede observar la distribución de las celdas y la central e comunicación.

### 2.5.2.2. Red de telefonía móvil GSM

GSM es una tecnología digital inalámbrica de segunda generación que presta servicios de voz de alta calidad, así como servicios de datos conmutados por circuitos en una amplia gama de bandas de espectro. Es un sistema de comunicación basado en el uso de células digitales que se desarrolla para crear un sistema para móviles único que sirva de estándar para todo el mundo y compatibles con los servicios existentes y futuros sobre una red digital de servicios integrados.

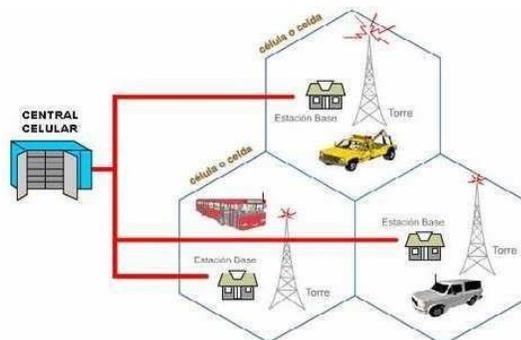
Con el GSM no es el teléfono que contiene los datos del abonado, sino más bien “una tarjeta inteligente” denominada SIM Card, que se inserta en el aparato desde el cual se desea llamar.

### 2.5.2.2.3. Mensajes de texto SMS

El servicio de mensajes SMS (Short Menssages Services), es una red digital que permite a los usuarios de teléfonos celulares enviar y recibir mensajes de texto, un mensaje de texto puede ser enviado desde un teléfono celular, un modem o desde una dirección IP, cada mensaje tiene una longitud de hasta 160 caracteres, estos 160 caracteres pueden ser palabras, números o una combinación alfanumérica.

Los SMS son enviados por la red móvil celular y son transferidos entre teléfonos por medio del Centro de Servicio de Mensajes Cortos (SMSC).

El SMSC es un software de la operadora de red que se encarga de manejar, procesar y facturar los mensajes. El despacho de los mensajes se lo realiza en colas de espera de tal forma que el mensaje tarda un tiempo en llegar al usuario destino el cual depende de la cantidad de mensajes y de la velocidad del software de la operadora. La interoperabilidad posibilita al cliente utilizar SMS de la misma forma que el servicio de voz, es decir se puede enviar y recibir mensajes de texto de un teléfono a otro en un tiempo muy corto.



### Figura 2.23: Celdas de transmisión<sup>7</sup>

Cada estación base está situada en una de estas celdas y tiene asignado un grupo de frecuencias de transmisión y recepción propio.

Como el número de frecuencias es limitado, es posible reutilizar las mismas frecuencias en otras células, siempre que no sean adyacentes, para evitar interferencias entre ellas, permitiendo que miles de personas puedan utilizar los teléfonos al mismo tiempo.

#### **2.5.2.2.4. Características**

El servicio de mensajería corta utiliza los canales de señalización SACCH (Slow Associated Control Channel) y SDCCH (Slow Dedicated Control Channel).

El SACCH es usado durante el transcurso de una llamada y el SDCCH es usado si el receptor está libre. Las características generales del SMS son:

- **Concatenación**  
Se puede concatenar algunos SMS estándar para formar un mensaje largo, se puede concatenar hasta 255 mensajes. Cuando esta característica es usada se debe incluir información adicional para que la aplicación pueda reenzambler correctamente los mensajes cortos.
- **Compresión**  
Permite comprimir los datos de usuario del mensaje, esta característica es opcional, y se basa en un algoritmo donde la longitud de la secuencia de salida es inversamente proporcional a la frecuencia con que el carácter ocurre en la secuencia de entrada.
- **Mensajería Binaria**  
El SMS puede ser combinado en el modo carácter o binario. El modo binario permite mejorar la eficiencia de los datos transmitidos.
- **Soporte de diferentes alfabetos**  
Se puede utilizar dos alfabetos: UCS2 y alfabeto por defecto GSM de 7 bits. El UCS2 (Universal Multiple Octet Coded Character Set 2) incorpora todos los principales lenguajes de todo el mundo.

---

<sup>7</sup> <http://sx-de-tx.wikispaces.com/Tecnologia+Celular>

El alfabeto por defecto de 7 bits se deriva del conjunto de caracteres ASCII. También se puede utilizar un alfabeto GSM de 8 bits para datos en modo binario.

#### **2.5.2.2.5. Arquitectura de la red SMS**

- Entidad de envío de mensajes cortos SME  
SME (Short Messaging Entity) entidad que puede enviar o recibir mensajes cortos, pudiendo estar localizada en la red fija, una estación móvil, u otro centro de servicios.
- Centro de Servicio de Mensajes SMSC  
SMSC (Short Message Service Center) es el responsable de la transmisión y almacenamiento de un mensaje corto, entre el SME y una estación móvil.
- Centro de Conmutación Móvil MSC MSC (Mobile Switching Center) lleva cabo funciones de conmutación del sistema y del control de llamadas a y desde otro teléfono del sistema de datos.
- SMS – Gateway/Interworking MSC (SMS - GMSC)

Es un SMC capaz de recibir un mensaje corto de un SMSC, interrogando al HLR (Home Location Register) sobre la información de encaminamiento y enviando el mensaje corto al MSC.

#### **2.5.2.2.5. Aplicaciones**

Las principales aplicaciones basadas en SMS son:

- **Mensajes de persona a persona**  
Los usuarios de telefonía móvil utilizan comúnmente el servicio de mensajería corta para comunicarse con otro usuario móvil de su misma operadora e incluso de una operadora diferente.
- **Servicios de notificación**  
Permite el envío de mensajes a ciertos usuarios que constan en una base de datos específica tales como: clientes de compañías de televisión, clubs deportivos, supermercados y otros minoristas, aerolíneas y bancos.
- **Servicios de localización**

Aplicado a la localización de vehículos, integra un GPS. Los datos de longitud y latitud son transferidos a un terminal móvil. El terminal por medio de un SMS envía a estos datos a un servidor donde se procesan para indicar la localización actual del vehículo en un mapa geográfico.

- **Supervisión remota**

El servicio de mensajería corta puede usarse para gestionar máquinas en ambientes de supervisión remota. Esta aplicación proporciona valiosa información sobre el estado o suceso de algún evento ocurrido sobre la máquina, que el usuario precisa saber.

- **Comercio electrónico**

Se puede llevar a cabo transacciones financieras a través del terminal móvil, para la cual será necesario tener convenios con algunas instituciones bancarias.

### 2.5.2.3. Modem GSM

#### 2.5.2.3.1. Puerto de comunicación

El puerto de comunicación del teléfono GSM (Nokia 3220), se describe a continuación de manera detallada, para poder identificar los pines que se utiliza para la comunicación e intercambio de datos.



*Figura 2.24: Conector del teléfono Nokia 3220* <sup>8</sup>

Este conector está conformado por 14 pines los cuales tienen una función específica. De este puerto los pines que se deben identificar son el de transmisión, recepción y tierra (Tx, Rx, GND), los cuales permitirán la transferencia de datos.

De todos los pines utilizados en el pop port del celular, los que utilizaremos son los pines 6, 7 y 8.

---

<sup>8</sup> [http://www.latest-mobile.com/articles/nokia/nokia-3220\\_3236](http://www.latest-mobile.com/articles/nokia/nokia-3220_3236)

Estos pines permitirán establecer una comunicación serial tal como se detalla a continuación.

- Pin 6 es Rx.
- Pin 7 es Tx.
- Pin 8 es GND.

La transferencia de los datos se genera con los siguientes parámetros los cuales deberán ser configurados en el dispositivo con el cual se va a comunicar.

- Velocidad de transmisión 9600 Baudios
- 8 bits de datos
- Bits de paridad "Ninguna"
- Bits de parada "1"

Los niveles de voltaje que utiliza este pÓrtico de conexión son, para el cero lógico 0 voltios y para el uno lógico 5 voltios, es decir trabaja con niveles TTL.

#### **2.5.2.4. MicroCode Studio <sup>9</sup>**



*Figura 2.25: MicroCode Studio*

##### **2.5.2.4.1. Características**

Microcode es un programa editor de texto similar a un bloc de notas, pero con la diferencia que esta hecho exclusivamente para facilitar la programación de los microcontroladores como los PIC.

---

<sup>9</sup> <http://www.mecanique.co.uk/code-studio/>

Los procedimientos para programar son muy sencillos, los cuales deben estar acorde con los modelos de microcontrolador a utilizarse, de esta manera debemos elegir el microcontrolador que utilizaremos en el proyecto es el 16F628A, este programa de escritura no trabaja por si solo, necesita de un compilador, ensamblador y un programador para que se encuentre completo.

#### **2.5.2.4.2. Configuración de MicroCode Studio**

Para poder utilizar de forma fácil y aprovechar las ventajas de este editor de texto programador se debe acoplar diferentes programas, como son el compilador y el ensamblador que juntos conforman el programa PicBasicPro y un programador como es IC-Prog.

Los cuales son sumamente necesarios para una correcta generación de código hexadecimal, el cual va a ser utilizado por el microcontrolador PIC.

La instalación del programa Microcode es similar a la de cualquier programa la diferencia radica en que se debe intercalar los dos programas restantes, una vez instalado el programa se genera una carpeta con el nombre Mecanique en la unidad C de memoria, es muy importante encontrar esta carpeta puesto que es la que a a contener los dos programas restantes.

#### **2.5.2.4.3. Programador de Microcontroladores**

Programador de Microcontroladores PIC: es una herramienta indispensable con la cual se puede grabar el código generado por el compilador PicBasic para poner en funcionamiento cada uno de los proyectos propuestos en cada capítulo. Existen en internet una gran cantidad de modelos de programadores para microcontroladores PIC, de muy bajo costo y fácil construcción. Consideramos una buena experiencia realizar el montaje de cualquiera de estos diseños, aunque en esta oportunidad es recomendable utilizar el P16Pro/Picallw.



Figura 2.26: Programador de Microcontroladores

#### **2.5.2.4.4. Microcontroladores**

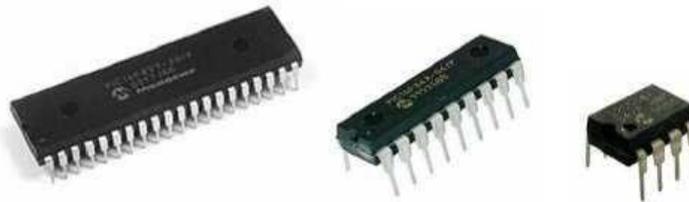
Los PIC's son una familia de microcontroladores desarrollados y fabricados por la empresa Microchip Technologies Inc., los cuales cuentan con una tecnología tipo RISC (*Reduced Instruction Set Computer*) y poseen en su arquitectura interna características especiales que varían según el modelo de PIC que deseamos utilizar.

Podríamos decir que estos dispositivos se asemejan a una computadora pero de tamaño muy reducido, ya que cuentan con casi los mismos recursos que éstas, es decir, poseen memoria de programa, memoria RAM, memoria de datos, puertos de entrada o salida, temporizadores y en algunos casos cuentan con recursos adicionales como convertidores A/D, comparadores, USART (Universal Synchronous/Asynchronous Receiver/Transmitter), comunicación serie I2C, entre otros.

Con todas estas características es lógico pensar que este dispositivo pasa a ser el corazón del circuito a ser controlado. Esto significa que el microcontrolador es el encargado de dirigir todos los procesos de un circuito electrónico, en base a las instrucciones de programa o rutinas que definen funciones específicas de control, donde las mismas serán realizadas en lenguaje Basic para microcontroladores PIC. Es por esta razón que consideramos muy importante estudiar la arquitectura interna del microcontrolador que se desea programar y aunque esta tarea pueda parecer difícil, el Lenguaje Basic para microcontroladores PIC la hace sumamente sencilla.

El diseño de programas para microcontroladores PIC va acompañado normalmente con un previo estudio del diseño del hardware que hará que los proyectos se pongan en marcha. Es decir, resulta absolutamente necesario saber cual será la función específica de cada pin; por ejemplo, en el caso de los puertos I/O (IN/OUT) a ser utilizados en el microcontrolador, es importante definir sus funciones antes de empezar a programar, ya que éstos pueden ser configurados a conveniencia como entrada o como salida de datos de forma independiente. También se puede destinar un puerto completo del microcontrolador para el control de dispositivos periféricos como pantallas LCD, teclados, motores paso a paso, leds, servomotores entre otros. De ahí la importancia de establecer cual será la función de cada puerto del microcontrolador PIC elegido para los proyectos.

Otra decisión importante será elegir convenientemente el modelo de microcontrolador a ser utilizado, ya que hay una gran gama de modelos que pueden ser adaptados a necesidades específicas de diseño.



*Figura 2.27: Imagen de un PIC*

#### 2.5.2.5. Vectir <sup>10</sup>



*Figura 2.28: Imagen de un PIC*

Vectir es una aplicación cliente/servidor que permite controlar ciertas aplicaciones del ordenador desde el móvil, el tipo de comunicación que utiliza puede ser mediante infrarrojo, Bluetooth o Wi-fi.

---

<sup>10</sup> <http://www.vectir.com/>

El server deberá ser instalado previamente en la PC y la aplicación JAVA Cliente en el celular, el cliente se conecta a la pc a través del Server y realiza las funciones instantáneamente en el ordenador.

Este software esta diseñado exclusivamente para que la PC pueda interactuar con el celular de una manera inteligente y amigable para realizar un control remoto al ordenador. Tiene además soporte para una amplia gama de dispositivos celulares.

Vectir hoy en día a sido utilizado tanto en casa como en el sector industrial y empresarial brindando soluciones rápidas y oportunas a diversas áreas de trabajo.

Esta aplicación también ha sido utilizada por varias personas para controlar remotamente al ordenador con un teléfono celular, de esta manera se puede tener un control preciso y rápido de documentos en el ordenador tales como presentaciones de PowerPoint o controlar su reproductor multimedia, Windows Media Player, iTunes, winamp, mouse. Vectir también hace que sea fácil de ajustar el comportamiento existente se adapte a sus necesidades personales. Cambia los tamaños de fuente y colores, añade nuevos comandos y elegir qué teclas del mando a teléfono funciones seleccionadas. Con la ayuda de los comandos de teclado, puede añadir soporte básico de control remoto para prácticamente cualquier aplicación que se ejecuta en la PC.

Además de esto también brinda una aplicación de escritorio remoto. Esta función le permite ver de forma remota y controlar el escritorio de los ordenadores desde un teléfono móvil a través de Bluetooth o Wi – Fi. Utilizando las teclas de navegación o la pantalla táctil para desplazarse en el teléfono sobre el escritorio del ordenador, ampliar, reducir y controlar el ratón.

Una amplia gama de teléfonos son compatibles con la función de Bluetooth o Wi-fi Remote Control. Nokia, Sony Ericsson, LG, Motorola, Samsung, Benq Siemens, HTC y Blackberry han sido probados con éxito. Siempre y cuando su teléfono tiene Bluetooth o conectividad WiFi y corre Android o soportes de aplicaciones JAVA, hay una buena probabilidad de que va a funcionar.

## **2.5.2.6. Comandos AT <sup>11</sup>**

### **2.5.2.6.1. Descripción**

Los comandos AT son instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un Terminal MODEM.

Los comandos AT fueron desarrollados en 1977 por Dennis Hayes como un interfaz de comunicación con un MODEM para así poder configurarlo y proporcionarle instrucciones, tales como marcar un número de teléfono. Más adelante, con el avance del baudío, fueron las compañías Microcomm y US Robotics las que siguieron desarrollando y expandiendo el juego de comandos hasta universalizarlo.

Los comandos AT se denominan así por la abreviatura de attention.

Aunque la finalidad principal de los comandos AT es la comunicación con modems, la telefonía móvil GSM también ha adoptado como estándar este lenguaje para poder comunicarse con sus terminales. De esta forma, todos los teléfonos móviles GSM poseen un juego de comandos AT específico que sirve de interfaz para configurar y proporcionar instrucciones a los terminales, permiten acciones tales como realizar llamadas de datos o de voz, leer y escribir en la agenda de contactos y enviar mensajes SMS, además de muchas otras opciones de configuración del terminal.

Es claro que la implementación de los comandos AT corresponde a los dispositivos GSM y no depende del canal de comunicación a través del cual estos comandos sean enviados, ya sea cable de serie, canal Infrarrojos, Bluetooth, etc.

### **2.5.2.6.2. Objetivo de los comandos AT**

Los comandos AT deben ser utilizados para el desarrollo de nuevas tecnologías y programas de comunicaciones y ajustar propiedades avanzadas del teléfono y módems inalámbricos.

Entre las funciones más usadas de los comandos AT se tienen:

- Configurar el teléfono para una conexión inalámbrica, a través de infrarrojos o por el sistema de bus o cable.
- Configurar el MODEM interno del teléfono para una conexión inalámbrica, a través de infrarrojos o por un sistema de bus o cable.

---

<sup>11</sup> <http://bluehack.elhacker.net/proyectos/comandosat/comandosat.html>

- Solicitar información sobre la configuración actual o estado operativo del teléfono o MODEM.

### 2.5.2.6.3. Comandos de configuración

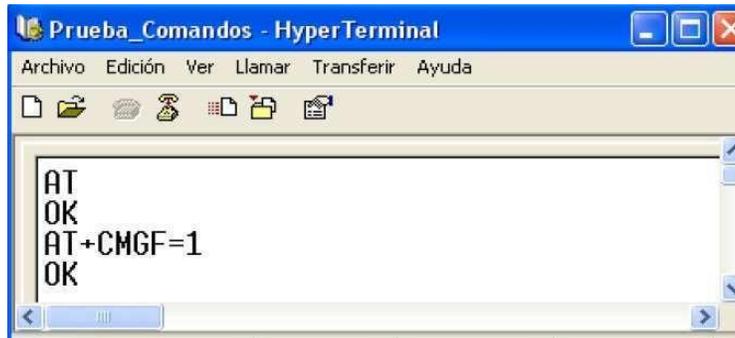
Este tipo de comandos permiten cambiar la configuración interna del MODEM integrado en el teléfono entre los que se tiene:

- **AT**, este es un comando de atención, cuya función es monitorear si existe una buena conexión en el canal de comunicación, si la conexión es buena el teléfono responde OK, en la figura se pone el código de programación que permite realizar esta acción a través del sistema microprocesado.



*Figura 2.29: Ejecución del comando AT*

- **AT+CMGF="A"**, este comando permite elegir el modo de interpretación de los datos por parte del teléfono, si la equivalencia de A es igual a "1", los datos son interpretados en modo texto, es decir la secuencia de caracteres que envía al teléfono son ASCII normales. Si la equivalencia de A es igual a "0", los datos son interpretados en modo PDU (Protocol Data Unit), por lo tanto los datos enviados al teléfono deben ser interpretados con caracteres HEX (Hexadecimales), para el sistema se ha tomado en cuenta la primera opción, debido a que la comunicación entre el sistema y el teléfono será monitoreada, por tanto la interpretación de datos será mas sencilla. El teléfono al recibir los datos responde OK, indicando que la petición ha sido aceptada. En la imagen se ilustra el código de programación que permite realizar esta acción.



*Figura 2.30: Ejecución del comando AT+CMGF=1*

#### 2.5.2.6.4. Comandos para envío SMS <sup>12</sup>

Después de que el teléfono ha sido configurado en el modo texto, una de las necesidades del sistema es el envío de SMS, mediante el cual podrá responder a un evento solicitado.

Se analiza el menú de mensajes en el software de cualquier teléfono celular, se observa que posee básicamente tres librerías como son: Buzón de Entrada, Buzón de Mensajes Enviados, Buzón de Mensajes o Elementos no Enviados. Cuando un mensaje llega a un teléfono celular este se almacena en la carpeta de Buzón de Entrada, dentro de esta carpeta el mensaje aparecerá como un mensaje no leído y una vez que es abierto el mensaje pasará a ser un mensaje leído, este caso dentro de los comandos AT es interpretado como REC UNREAD y RED READ respectivamente.

Al hablar del tema, se trata de indicar que si el sistema desea enviar o recibir un mensaje, debe saber a que librería debe apuntar en el teléfono para alcanzar su objetivo, pero como se utiliza un teléfono GSM, hay que tomar en cuenta que posee dos memorias que tienen las mismas librerías, por lo tanto el sistema también tiene que identificar cual es la memoria en la que se esta trabajando. Para enviar un SMS, el teléfono requiere de dos comandos que son: el AT+CMGW y AT+CMSS.

---

<sup>12</sup> Tesos: *Estudio de la Comunicación con comandos AT y Microcontroladores caso practico de implementación de un sistema de gestión de alarmas para viviendas con monitoreo mediante telefonía celular.* Autor: Martha Alulema. Año 2010. ESPOCH

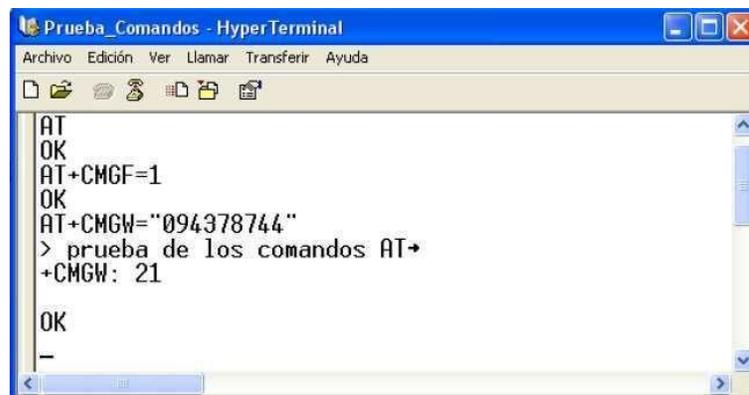


```
AT
OK
AT+CMGF=1
OK
AT+CMGW="094378744"
> prueba de los comandos AT+
+CMGW: 22

OK
AT+CMSS=22_
```

*Figura 2.31: Comando AT+CMSS=1, envía SMS*

El comando **AT+CMGW="#"Telefónico"**, permite cargar en el teléfono la información del SMS, al número que se especifica dentro de los parámetros del comando (#Telefónico).



```
AT
OK
AT+CMGF=1
OK
AT+CMGW="094378744"
> prueba de los comandos AT+
+CMGW: 21

OK
-
```

*Figura 2.32: Comando AT+CMGW, para cargar un SMS en el teléfono*

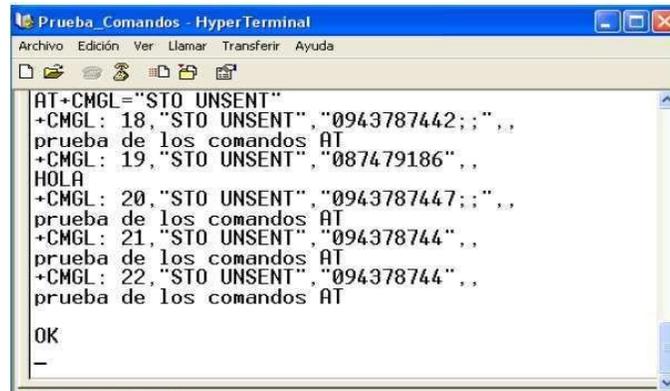
Cuando el comando es enviado con el número telefónico, el teléfono responde con un signo ">", de esta manera indica que se debe ingresar la información del mensaje, una vez que la información a sido ingresada se adhiere el carácter ctrl.Z, de esta manera indica el final del mensaje al teléfono, por lo tanto si la ejecución del comando ha sido correcta el teléfono responde OK.

El comando **AT+CMPS="LOCALIDAD MEM"**, permite enviar el SMS cargado en el teléfono, al igual que lo haría la tecla "Send", El dominio "Localidad del MEM", indica la posición del mensaje dentro de la memoria (ya sea en la memoria interna del teléfono o en la SIM), al recibir este comando el teléfono responde OK, cuando se ejecuta en forma correcta.

#### **2.5.2.6.5. Comandos para recepción de SMS**

Una vez que se carga un SMS en el teléfono celular, este se guarda dentro de la librería de Buzón de Mensaje no enviado en la tarjeta SIM, para el caso de recepción de mensajes, necesariamente el sistema debe apuntar a la memoria del teléfono donde el mensaje recibido es guardado.

A continuación se muestra la extracción del SMS, debemos tener en cuenta que no se realizó ninguna instrucción previa para cambiar de memoria, antes de la extracción del SMS.



```
Prueba_Comandos - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
AT+CMGL="STO UNSENT"
+CMGL: 18,"STO UNSENT","0943787442;";",,
prueba de los comandos AT
+CMGL: 19,"STO UNSENT","087479186",,
HOLA
+CMGL: 20,"STO UNSENT","0943787447;";",,
prueba de los comandos AT
+CMGL: 21,"STO UNSENT","094378744",,
prueba de los comandos AT
+CMGL: 22,"STO UNSENT","094378744",,
prueba de los comandos AT
OK
-
```

*Figura 2.33 Extracción del SMS mediante el Comando AT+CMGL="STO UNSENT"*

#### 2.5.2.6.6. Comandos AT más utilizados

Estos son los comandos mas comunes en la mayoría de los modems y los que más se usan.

#### **Comandos para SMS**

- AT+CPMS: Seleccionar lugar de almacenamiento de los SMS.
- AT+CMGF: Seleccionar formato de los mensajes SMS.
- AT+CMGR: Leer un mensaje SMS almacenado.
- AT+CMGL: Listar los mensajes almacenados.
- AT+CMGS: Enviar mensaje SMS.
- AT+CMGW: Almacenar mensaje en memoria.
- AT+CMSS: Enviar mensaje almacenado.
- AT+CSCA: Establecer el Centro de mensajes a usar.
- AT+ WMSC: Modificar el estado de un mensaje.

#### 2.5.3. Infraestructura de Comunicación

### 2.5.3.3. Nivel de Campo

#### 2.5.3.3.1. Protocolo de Comunicación TCP/IP (CFP – ESTACIÓN MAESTRA)

TCP/IP es el protocolo común utilizado por todos los ordenadores conectados a Internet, de manera que éstos puedan comunicarse entre sí.

Hay que tener en cuenta que en Internet se encuentran conectados ordenadores de clases muy diferentes y con hardware y software incompatibles en muchos casos, además de todos los medios y formas posibles de conexión. Aquí se encuentra una de las grandes ventajas del TCP/IP, pues este protocolo se encargará de que la comunicación entre todos sea posible. TCP/IP es compatible con cualquier sistema operativo y con cualquier tipo de hardware.



*Figura 2.34: Conexión de un CFP a una estación*

TCP/IP no es un único protocolo, sino que es en realidad lo que se conoce con este nombre es un conjunto de protocolos que cubren los distintos niveles del modelo OSI. Los dos protocolos más importantes son el TCP (Transmission Control Protocol) y el IP (Internet Protocol), que son los que dan nombre al conjunto. En Internet se diferencian cuatro niveles o capas en las que se agrupan los protocolos, y que se relacionan con los niveles OSI de la siguiente manera:

Aplicación: Se corresponde con los niveles OSI de aplicación, presentación y sesión. Aquí se incluyen protocolos destinados a proporcionar servicios, tales como correo electrónico (SMTP), transferencia de ficheros (FTP), conexión remota (TELNET) y otros más recientes como el protocolo HTTP (Hypertext Transfer Protocol).

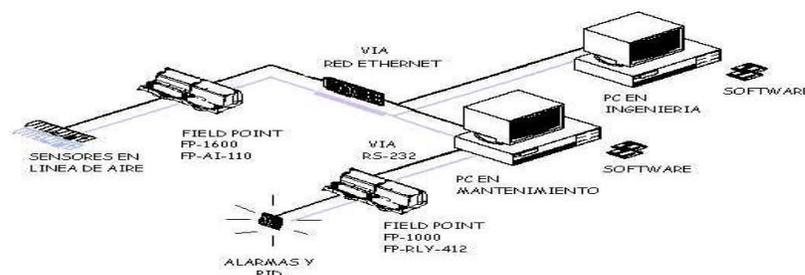
Transporte: Coincide con el nivel de transporte del modelo OSI. Los protocolos de este nivel, tales como TCP y UDP, se encargan de manejar los datos y proporcionar la fiabilidad necesaria en el transporte de los mismos.

Internet: Es el nivel de red del modelo OSI. Incluye al protocolo IP, que se encarga de enviar los paquetes de información a sus destinos correspondientes. Es utilizado con esta finalidad por los protocolos del nivel de transporte.

Enlace: Los niveles OSI correspondientes son el de enlace y el nivel físico. Los protocolos que pertenecen a este nivel son los encargados de la transmisión a través del medio físico al que se encuentra conectado cada host, como puede ser una línea punto a punto o una red Ethernet.

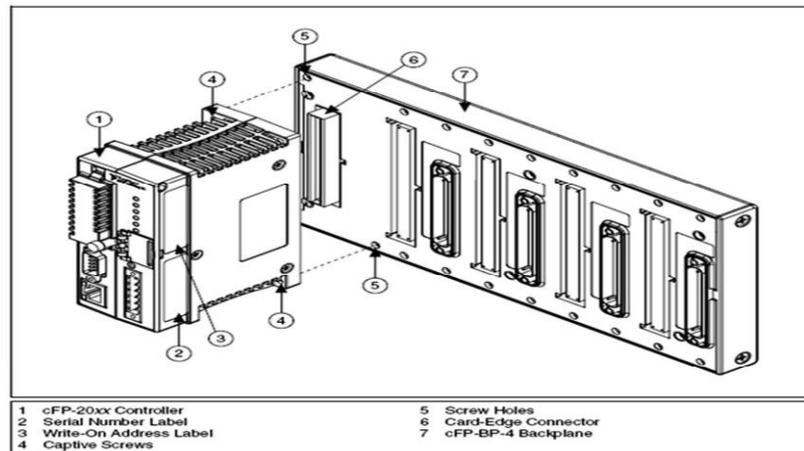
El TCP/IP necesita funcionar sobre algún tipo de red o de medio físico que proporcione sus propios protocolos para el nivel de enlace de Internet. Por este motivo hay que tener en cuenta que los protocolos utilizados en este nivel pueden ser muy diversos y no forman parte del conjunto TCP/IP. Sin embargo, esto no debe ser problemático puesto que una de las funciones y ventajas principales del TCP/IP es proporcionar una abstracción del medio de forma que sea posible el intercambio de información entre medios diferentes y tecnologías que inicialmente son incompatibles.

Para transmitir información a través de TCP/IP, ésta debe ser dividida en unidades de menor tamaño. Esto proporciona grandes ventajas en el manejo de los datos que se transfieren y, por otro lado, esto es algo común en cualquier protocolo de comunicaciones. En TCP/IP cada una de estas unidades de información recibe el nombre de "datagrama" (datagram), y son conjuntos de datos que se envían como mensajes independientes.



*Figura 2.35: Integración de CFP's a una topología LAN*

**2.5.3.3.2. Partes y Elementos CFP Modelo 2020** Composición de un Compact FieldPoint modelo 2020.



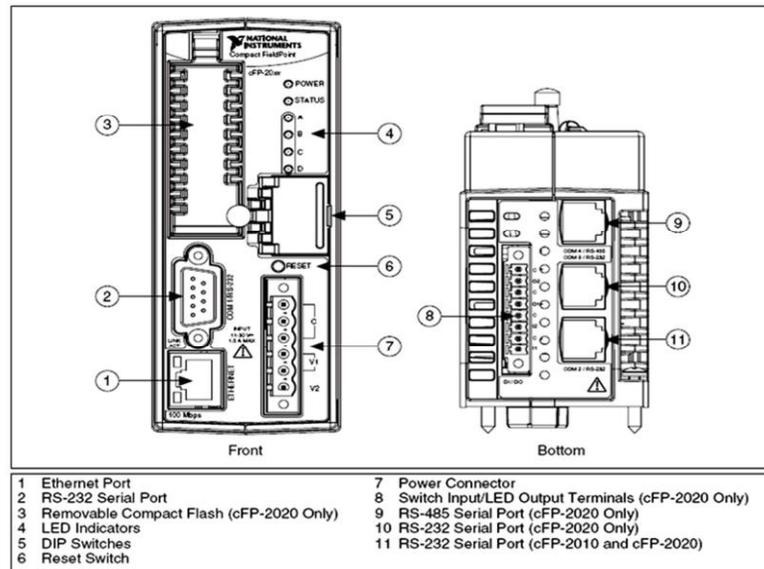
*Figura 2.36: Partes de un CFP 2020*

Nº	CARACTERÍSTICA – DESCRIPCIÓN
1	CFP-20XX Controller: Controlador CFP
2	Etiqueta con Número de Serie
3	Etiqueta con Direccionamiento del Equipo
4	Tornillos de sujeción
5	Agujeros para tornillos
6	Conector para el Controlador CFP
7	Placa del circuito CFP

*Cuadro 2.2: Partes de un CFP 2020*

El controlador CFP 2020 es un minicomputador capaz de captar y arrojar señales que junto con el software de programación LabView posibilita el desarrollo de sistemas de gran complejidad, capaces controlar y monitorear procesos en tiempo real ajustado a estándares de calidad y seguridad industrial.

Está principalmente compuesto por el CPU o controlador propiamente dicho.



**Figura 2.37: Partes del Controlador CFP 2020**

Nº	CARACTERÍSTICA – DESCRIPCIÓN
1	Puerto Ethernet
2	Puerto Serial RS-232
3	Compact Flash extraíble
4	Indicadores LED
5	Interruptores DIP
6	Interruptor de Reseteo
7	Conector de energía
8	Interruptor de entrada/LED Terminales de salida
9	Puerto Serial RS-485
10	Puerto Serial RS-232
11	Puerto Serial RS-232

**Cuadro 2.3: Partes de un Controlador CFP 2020**

### 2.5.3.4. Nivel de Control Remoto y Supervisión

#### 2.5.3.4.1. Protocolo HTTP

El protocolo de transferencia de hipertexto (HyperText Transfer Protocol) es un protocolo del nivel de aplicación usado para la transferencia de información entre sistema, de forma clara y rápida. Este protocolo ha sido usado por el World-Wide Web desde 1990.

Este protocolo de transferencia permite usar una serie de métodos para indicar la finalidad de la petición. Se basa en otros conceptos y estándares como Uniform Resource Identifier (URI), Uniform Resource Location (URL) y Uniform Resource Name (URN), para indicar el recurso al que se hace referencia la petición. Los mensajes se pasan con un formato similar al usado por el Internet Mail y el Multipurpose Internet Mail Extensions (MIME).

El protocolo HTTP se basa en un paradigma de peticiones y respuestas. Un cliente envía una petición en forma de método, una URI y una versión de protocolo seguida de los modificadores de la petición de forma parecida a un mensaje MIME, información sobre el cliente y al final un posible contenido. El servidor contesta con una línea de estado que incluye la versión del protocolo y un código que indica éxito o error, seguido de la información del servidor en forma de mensaje MIME y un posible contenido.

Generalmente es el cliente quien inicia la comunicación HTTP y consiste en la petición de un recurso del servidor. Puede hacerse de forma directa al servidor o a través de intermediarios.

Sintaxis de petición:

El esquema “http” se usa para localizar recursos en la red por medio de protocolo http.

La sintaxis de la petición es la siguiente:

**http://”dirección [:puerto][path]”**

Donde dirección es el nombre del dominio de Internet o una dirección IP, el puerto es un número que indica el puerto al que se envía la petición y el path indica el recurso al que se accede.

Si no se indica un número de puerto, por defecto se supone que se accede al puerto 80.

Si no se indica un path, entonces se supone que este es “/”.

#### **2.5.3.4.2. Web Publishing Tool LabVIEW**

La Herramienta de Publicación web permite al desarrollador propiciar un ambiente ideal para poner al alcance de los usuarios una herramienta de total control y monitoreo en tiempo real de la aplicación que se ha configurado, todo esto a través de una conexión a internet por medio de Microsoft Internet Explorer.

Desde la interfaz principal de LabVIEW es posible configurar todos los parámetros de seguridad, puertos de comunicación así como el acceso a la aplicación o a las aplicaciones que posteriormente podrán visualizarse en Internet Explorer, algo muy interesante y que es de vital importancia a la hora de desarrollar sistemas de control es el nivel de acceso que deben tener los usuarios separándolos en dos grupos:

- Acceso Limitado:

Usuarios que únicamente recurren al sistema para verificar el normal desempeño de los procesos ejecutados sin poder realizar utilizar los controles que se hallan implementados en el panel de control.

- Acceso Total:

Usuarios que por sus funciones están en condiciones de interactuar de manera total con el sistema, interactuando de manera directa con todos los elementos que presenta el sistema.

Es necesario tener instalado y configurado como servidor web a Internet Information Server (IIS) ya que de no ser así la herramienta de publicación web no podría trabajar, adicional a ello también es necesario tener él complemento propio para ejecutar aplicaciones desarrolladas bajo el entorno de LabVIEW, este complemento es el RUN TIME ENGINE que permitirá a Internet Explorer mostrar el contenido de los paneles frontales de la aplicación publicada.

**Run TIME ENGINE:** Conjunto de librerías que permiten al sistema operativo Windows ejecutar VI's, aún a pesar de que no se halle instalada la Suite de LabVIEW en el equipo.

## **CAPITULO III**

### **3. METODOLOGÍA**

Para la consecución del proyecto como se tiene establecido se ha utilizado un amplio estudio bibliográfico enfocado al análisis, diseño y desarrollo de sistemas de control automático, y además se ha tomado en cuenta las bondades que ofrece hoy en día las telecomunicaciones para aprovecharlas en el campo de la industria.

El método de investigación experimental será utilizado para determinar cuáles son las herramientas más idóneas para ser utilizadas en LabVIEW para desarrollo de sistemas de monitoreo y control.

Para la comunicación del teléfono celular con la Pc se ha utilizado el programa MicroCode Studio y el puerto serial adaptado a los USB de la laptop.

La investigación se desarrolló en función del cronograma de trabajo, planificado para cubrir los objetivos planteados. Se empezó analizando los diversos tipos de control que existen actualmente, y el software necesario que podamos utilizar para la culminación del proyecto.

Los resultados se obtuvieron de la experimentación y recolección de muestras y ensayos, describiendo los fenómenos que se van presentando a medida que avanzó el proceso.

#### **3.1. TIPO DE ESTUDIO**

El sistema de control basa su funcionamiento, arquitectura e infraestructura en información recogida de proyectos realizados tanto experimentales como de aquellos desarrollados e implementados ya en el sector industrial.

Cada sección y módulo que conforma el proyecto se encuentra fundamentado en un estudio teórico práctico de las tecnologías a aplicar para la consecución del mismo, se ha tomado como referente libros y materiales especializados en sistemas de control como: manuales y guías de usuario para manejo y desarrollo de aplicaciones con LabVIEW con el fin de entender y comprender el funcionamiento de las herramientas y ventajas que brinda al programador.

#### **3.2. COMPÁRACION ENTRE CFP Y PLC**

Los PLCs se han utilizado en aplicaciones industriales de control y automatización desde que fueron introducidos en los años 60. Primero fueron introducidos como un remplazo para los circuitos de relés secuenciales usados para el control de

maquinarias. La secuencia en la cual sus salidas se encienden o apagan se define por software.

Comparado a los primeros PLCs, que se basaban en CPUs de bit-slice y por lo tanto eran solamente capaces de E/S digitales, los modernos PLCs hacen uso las tecnologías de microprocesador más recientes. Esto permite que incorporen características más avanzadas tales como entradas y salidas analógicas, comunicación en red y lenguajes de programación de mayor nivel.

El paso evolutivo obvio en estas herramientas era el desarrollo de los productos que ofrecieron la ventaja de ambas plataformas. Compañías tales como Rockwell, Siemens, GE Fanuc y National Instruments comprendieron la necesidad de estos dispositivos y ofrecen hoy plataformas enteras basadas en este concepto.

Los nuevos controladores resultantes, diseñados para tratar usos más especializados, combinan las mejores características del PLC con las mejores características de la PC. El analista de industria ARC denominó a estos dispositivos como los controladores de automatización programables, o PACs.

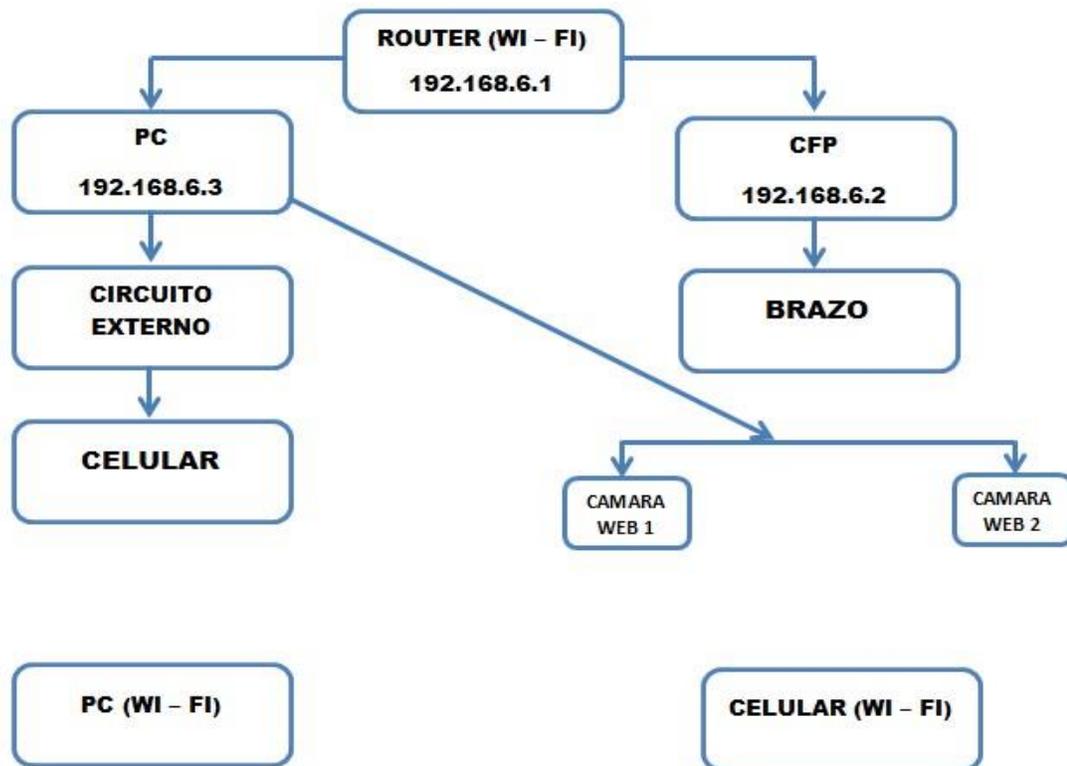
Los PACs y los PLCs poseen varias cosas en común: internamente ambos incluyen una fuente de energía, una CPU, una base para E/S, módulos y poseen registros de memoria que reflejan los canales individuales de E/S sobre los módulos. No obstante ello, las diferencias son bastante significativas.

En su estudio denominado "Visión Mundial de los Controladores Lógicos Programables", ARC identificó cinco principales características de los PACs:

- Funcionalidad de dominios múltiples, al menos dos de lógica, movimiento, control PID, manejadores y procesos en una sola plataforma.
- Plataforma de desarrollo simple multidisciplinaria, incorporando etiquetado común y una sola base de datos para acceder a todos los parámetros y funciones.
- Herramientas de software que permiten el diseño por flujo de proceso a través de diversas máquinas o unidades de procesos, junto con la norma IEC 611313, guía del usuario y administración de datos.
- Arquitecturas abiertas, modulares, que reflejen las aplicaciones industriales a partir de planos de maquinarias en fábricas a operaciones unitarias en plantas de procesos.
- Utilización de normas para interfaces de redes, lenguajes, etc., tales como TCP/IP, OPC y consultas XML y SQL.

Los PACS de National Instruments se basan en la tecnología NI LabVIEW, incluyendo el LabVIEW Real-Time y LabVIEW FPGA. Con LabVIEW, se pueden crear sistemas personalizados de medición y control y descargarlos en objetivos confiables embebidos corriendo un sistema operativo de tiempo real o aún embeberlos en silicio sin necesidad de conocer la programación VHDL.

### Diagrama de bloques del sistema



*Figura 3.1: Diagrama de bloques del sistema*

### DESCRIPCIÓN DEL FUNCIONAMIENTO DEL SISTEMA

La estación maestra está compuesta por la PC, las dos cámaras web, circuito externo, teléfono celular, router, CFP y brazo robótico.

En la PC que compone la estación maestra se encuentra el software con la interfaz Hombre Máquina ejecutándose para dar servicio a las demás estaciones de trabajo. Las cámaras web nos permitirán crear un sistema básico de monitoreo de los movimientos del brazo.

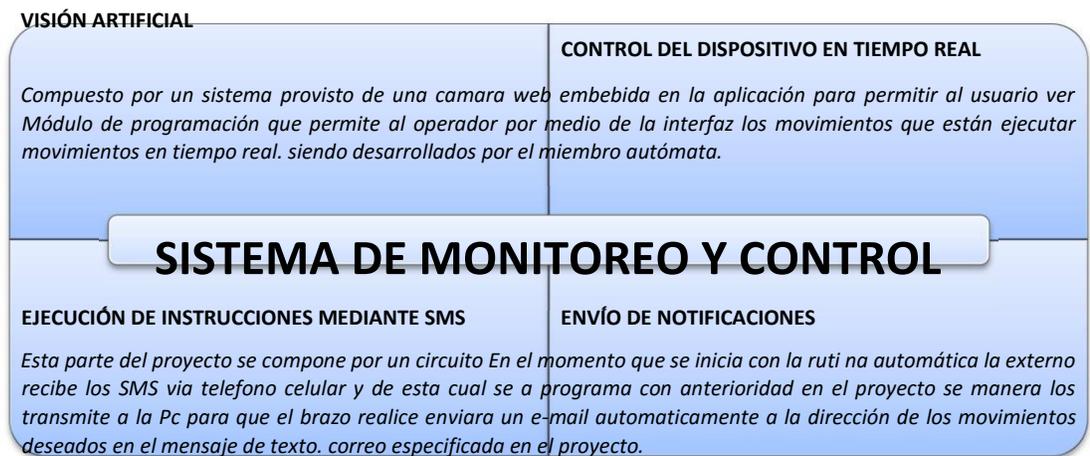
El router será el encargado de mantener comunicado a la PC y al CFP por medio de cable Ethernet, y por medio de su Wi-Fi brindará conexión a las estaciones de trabajo y a nuestro celular para el monitoreo.

El CFP cumplirá la función de instrumento de campo, el mismo que será el encargado de sacar al mundo exterior las instrucciones de la PC.

El celular conectado a la PC es el cual va a recibir los SMS enviados a través de la operadora celular (en este caso claro), los mismos que van a ser codificados por el PIC que conforman el circuito externo y los enviará hacia al ordenador para que nuestro software interprete dichas instrucciones y mande al brazo a ejecutar el movimiento correspondiente.

### 3.3. DISEÑO Y DESARROLLO DEL SISTEMA DE MONITOREO Y CONTROL.

LabVIEW en su entorno de programación y desarrollo de aplicaciones ofrece un conjunto de herramientas que facilita al programador: diseñar e implementar sistemas de extensa complejidad.



*Figura 3.2: Diagrama Referencial*

#### 3.3.1. Panel de Controles en Tiempo Real

Esta sección del sistema permite al usuario ejecutar movimientos que no han sido establecidos, para realizar un movimiento el usuario tiene a disposición 2 botones por servomotor, uno de los cuales activa de manera inversa (cierra, giro izquierda, bajar) y otro de manera directa (abre, giro derecha, subir).

##### 3.3.1.1. Lógica de Funcionamiento

El brazo robótico que utilizamos está constituido por cinco servomotores que pueden rotar en dos direcciones por tanto los controles que se han establecido desde la interfaz controlan el sentido de operatividad.

La condición fundamental es que un servomotor únicamente se lo puede activar al presionar un solo botón ya que de presionar los dos al mismo tiempo ninguno resultará activo.

A continuación se puede observar el flujograma con el desempeño de las variables a utilizar.

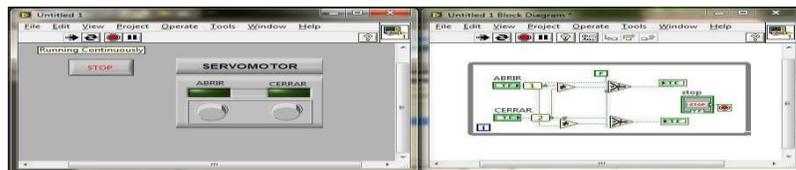
##### 3.3.1.2. Diseño e Implementación en LabVIEW

Siguiendo la lógica de programación anteriormente expuesta se ha utilizado varios elementos de programación propios de LabVIEW, como:

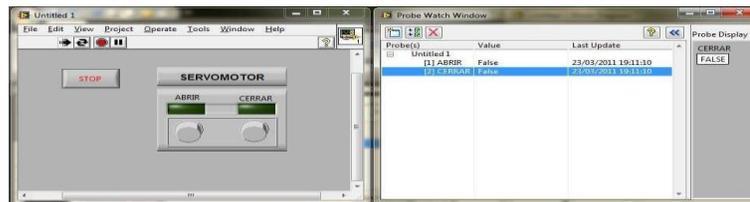
NOMBRE	ELEMENTO	DESCRIPCIÓN
CONTROL BOOLEANO (BOTON)		Enciende o apaga un servomotor.
CONDICIONAL (NO ES IGUAL)		Arroja un valor booleano (1 – 0) al ejecutar la condición.
CONTANTE BOOLEANA (FALSE)		Es una constante booleana que siempre arroja el valor FALSE.
INDICADOR BOOLEANO (LED)		Indicador que se enciende o apaga de ser el caso.

**Cuadro 3.1: Elementos de Programación**

En la siguiente tabla de referencia se especifican los botones que se encuentran configurados en la interface así como los servomotores que se activan con cada uno de ellos y la tecla específica que lo enciende o apaga.



**Figura 3.3: Panel Frontal y Diagrama de Bloques CTR**



**Figura 3.4: Controles en Tiempo Real Ejecución 1**

BOTON	VALOR	ESTADO
ABRIR	False	APAGADO
CERRAR	False	APAGADO

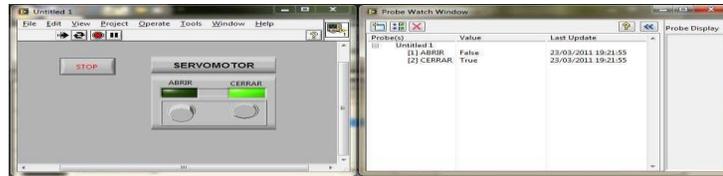
**Cuadro 3.2: Estado de Servomotores Ejecución 1**



**Figura 3.5: Controles en Tiempo Real Ejecución 2**

BOTON	VALOR	ESTADO
ABRIR	True	ENCENDIDO
CERRAR	False	APAGADO

**Cuadro 3.3: Estado de Servomotores Ejecución 2**



*Figura 3.6: Controles en Tiempo Real Ejecución 3*

BOTON	VALOR	ESTADO
ABRIR	False	APAGADO
CERRAR	True	ENCENDIDO

*Cuadro 3.4: Estado de Servomotores Ejecución 3*



*Figura 3.7: Controles en Tiempo Real Ejecución 4*

BOTON	VALOR	ESTADO
ABRIR	True	APAGADO
CERRAR	True	APAGADO

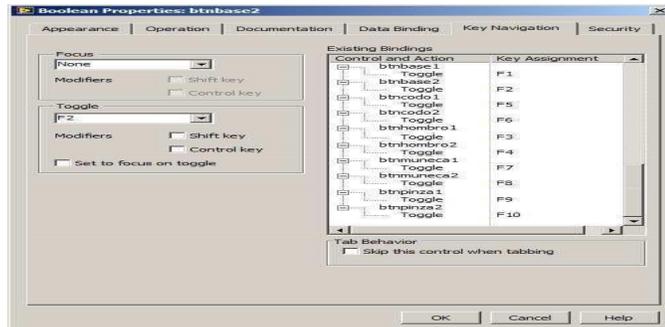
*Cuadro 3.5: Estado de Servomotores Ejecución 4*

De esta manera se halla constituido dos controles por cada servomotor utilizando la misma lógica de programación expuesta anteriormente.



*Figura 3.8: Controles en Tiempo Real, Controles en Tiempo Real*

Para mejorar la operatividad de cada control se ha establecido un comando configurado en KEY NAVIGATION al interior de las propiedades de cada control (botón), presentando una mayor accesibilidad a los controles a través del teclado. A continuación se muestra los controles que activan los servomotores que propician los movimientos de los distintos servomotores.



*Figura 3.9: Configuración de Key Navigation*

### 3.3.1.3. Secuencia Automática



*Figura 3.10: Estado de Servomotores secuencia*

El botón de activación de la secuencia automática fue diseñado con el objetivo fundamental de permitir al usuario ejecutar rutinas sin que necesariamente se encuentre frente al sistema, es justamente ahí que en conjunto con los sensores el sistema es capaz de terminar si existen problemas con los distintos elementos que trabaja como elementos electrónicos y medios de comunicación.

Como se puede observar en la interface del programa se puede ingresar direcciones de e-mail para que el programa envíe notificaciones de correo electrónico, los mismos que se dispararan al inicio y al terminar la secuencia.

A continuación se expone una de las cuatro rutinas automáticas ya que básicamente son las mismas y funcionan de manera similar en cuanto a estructura, lógica y programación.

#### 3.3.1.3.1. Lógica de Funcionamiento

Para desarrollar una rutina de movimientos se ha procedido a realizar un análisis de tiempos en vista de los servomotores y canales que serán activados. A continuación se presenta los valores configurados y los elementos que trabajaran:

Nº	CFP	CHANNEL	TIEMPO (s)	MOVIMIENTO
1	5	1	5	BASE DERECHA
2	4	7	5	HOMBRO BAJA
3	4	4	8	CODO BAJA
4	4	1	2	PINZA ABRE
5	4	0	2	PINZA CIERRA
6	4	6	2	HOMBRO SUBE
7	5	0	10	BASE IZQUIERDA
8	4	7	2	HOMBRO BAJA
9	4	1	2	PINZA ABRE
10	4	6	2	HOMBRO SUBE
11	5	1	10	BASE DERECHA
12	4	7	2	HOMBRO BAJA
13	4	0	2	PINZA CIERRA
14	4	6	2	HOMBRO SUBE
15	5	0	10	BASE IZQUIERDA
16	4	7	2	HOMBRO BAJA
17	4	1	2	PINZA ABRE
18	4	6	2	HOMBRO SUBE
19	5	1	6	BASE DERECHA
20	4	5	9	CODO SUBE
21	4	6	4	HOMBRO SUBE
22	4	0	2	PINZA CIERRA

*Cuadro 3.6: Estado de Servomotores secuencia*

Lo esencial para la programación, es entonces tomar un tiempo inicial y sumar cada uno de los valores siguientes para ir moviendo de manera coordinada un servomotor por vez de tal manera que al finalizar la secuencia de movimientos pueda repetirse sin interrupciones.

### 3.3.1.3.2. Diseño e Implementación en LabVIEW

Tomando en cuenta los tiempos y movimientos anteriormente expuestos es necesario programarlos en LabVIEW de acuerdo a las herramientas que nos ofrece en casos de control de tiempo.

El elemento más importante será por tanto el que se muestra a continuación:



*Figura 3.11: Elapsed Time*

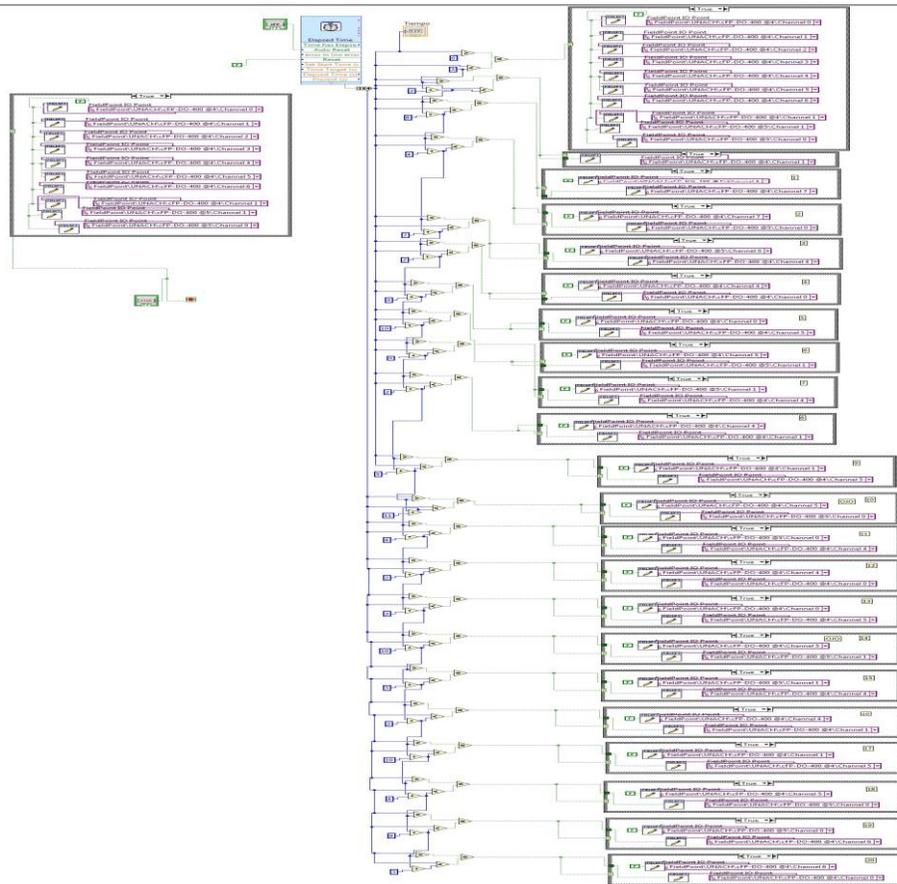
Que se encargará de ejecutar un lapso de tiempo repetitivo o no, dependiendo de su configuración.

Como se supo indicar anteriormente la suma de todos los tiempos es igual a 78 segundos, por tanto una vez concluido este periodo la secuencia se repetirá dependiendo de la rutina.



*Figura 3.12: Configuración Elapsed Time*

A continuación se muestra cual es el esquema de la programación en diagramas de bloques con toda la lógica ya transcrita a LabVIEW.



*Figura 3.13: Diagrama de Bloques Controlés Automáticos*

### 3.3.1.4. Envío de Notificaciones

Este sistema de control está compuesto por un módulo de envío de notificaciones configuradas y establecidas, en base a lo expuesto se ha incorporado una sección que incorpora cuentas de correo destino, descripción de la tarea realizada y su procedencia.

#### 3.3.1.4.1. Lógica de Funcionamiento

La función principal que desempeña la sección de envío de notificaciones es informar al usuario la presencia de errores a través de correos electrónicos enviados a las cuentas de usuarios configuradas.

El proceso se inicia con el ingreso de las cuentas de correo (remitente, destinatario), posteriormente se puede establecer si se desea incorporar una nueva cuenta a la que

se enviará una copia del correo para constancia del error.  
 copia del correo para constancia del error.

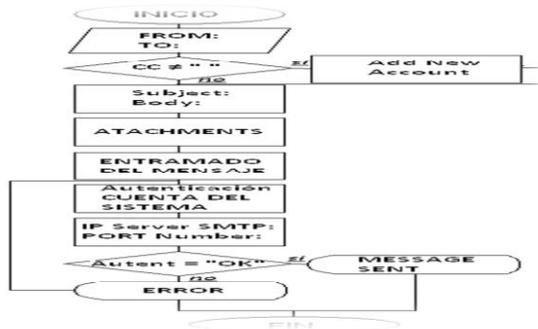


Figura 3.14: Flujo de envío de Notificaciones.

### 3.3.1.4.2. Diseño e Implementación en LabVIEW

Siguiendo la lógica de programación anteriormente expuesta se ha utilizado varios elementos de programación propios de LabVIEW, como:

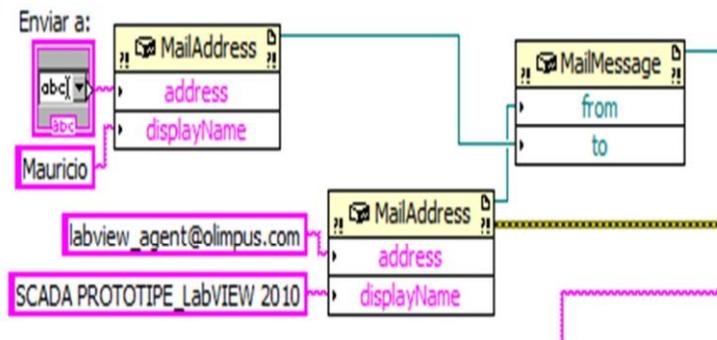
NOMBRE	ELEMENTO	DESCRIPCIÓN
Get Date / Time String		Toma la hora que marca el sistema.
Constructor Node		Constructor .NET que permite al desarrollador ingresar una cuenta de correo electrónico.
Constructor Node		Constructor. NET: configurado para ingresar una cuenta de correo electrónico.
Constructor Node		Constructor .NET que permite al desarrollador ingresar una cuenta de correo electrónico.
Property Node		InvokeNode, permite enviar una copia del correo electrónico a otra cuenta de correo.
Constructor Node		Constructor .NET permite ingresar el nombre de cuenta y contraseña de autenticación.
Constructor Node		Recibe la Dirección IP del servidor SMTP
Property Node		Especifica el Número de puerto por el que transmite la información el servidor SMTP
Property Node		Autentica los parámetros ingresados de inicio de sesión de una cuenta en el servidor SMTP
Invoke Node		Concluye el proceso y envía el mensaje.

Cuadro 3.7: Elementos de Programación

a) Ingreso de Información (Contactos, Correo)

Como parte inicial en el proceso de envío de notificaciones vía correo electrónico es indispensable especificar las cuentas de correo de el usuario que envía así como del que recibe la información.

En la siguiente imagen se muestra la programación realizada en el diagrama de bloques que permite al usuario especificar cuáles son las cuentas de usuario que estarán configuradas.

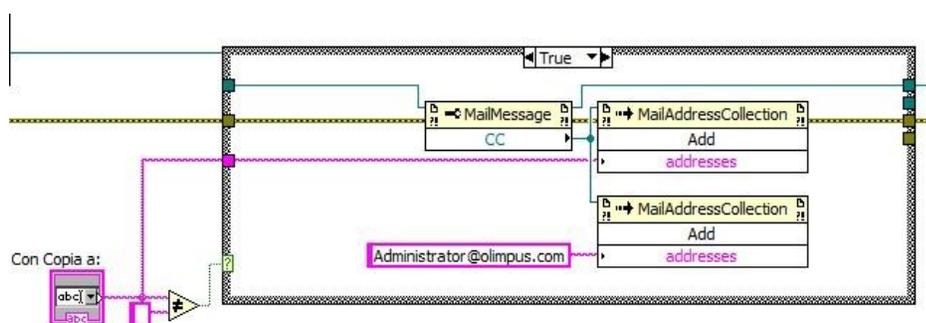


*Figura 3.15: Información de Contactos*

b) Envío de una Copia del Correo Electrónico

En caso de requerirse, se ha especificado en el sistema enviar una copia a otro destinatario.

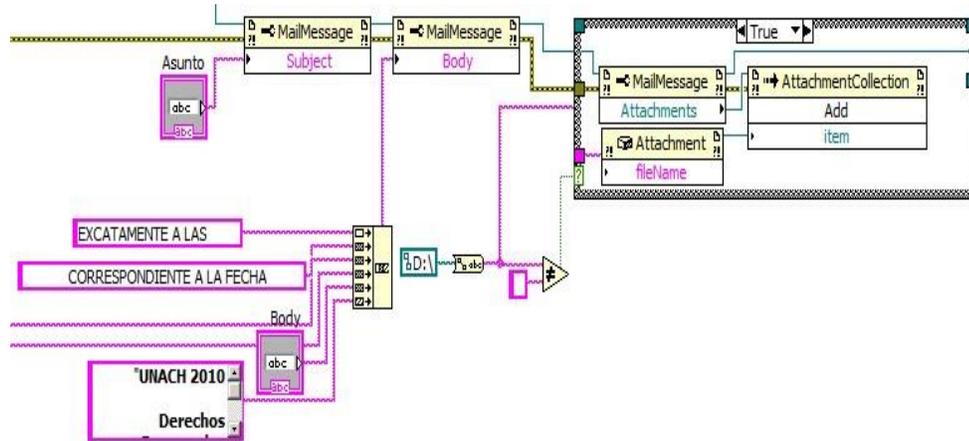
Los elementos que se han utilizado son los mismos que se utilizaron anteriormente. La representación en el diagrama de bloques de esta sección se aprecia a continuación:



*Figura 3.16: Envío de Copia a otro destinatario*

c) Elaboración del Body en el Correo Electrónico

Parte de la información a ser enviada al interior de la notificación será: fecha y hora del error, posible causa y además se adjuntará un documento con el reporte de los últimos procesos realizados.



*Figura 3.17: Elaboración del Body en la Notificación*

d) Autenticación Cuenta del Sistema

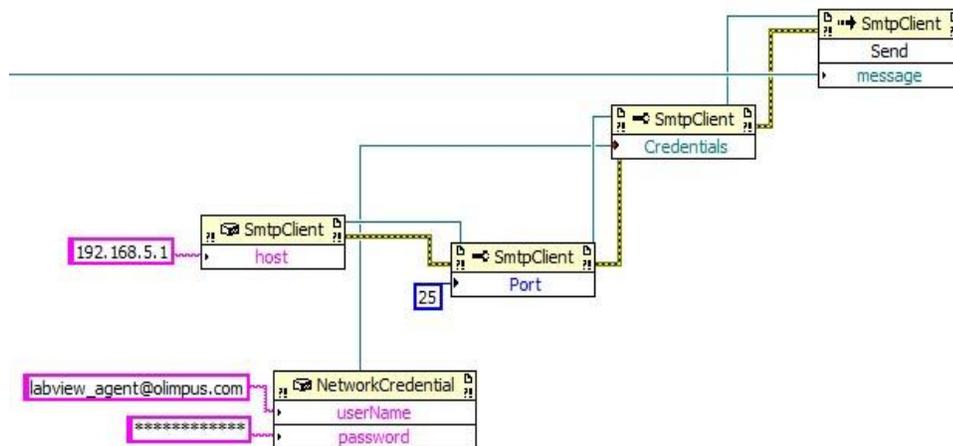
Para completar el envío de la notificación es necesario autenticar la cuenta del sistema en el servidor SMTP siendo indispensable ingresar información como:

Dirección IP del Servidor SMTP.

Puerto de transmisión.

Id de la Cuenta y su Contraseña.

Con todos los valores anteriormente enunciados configurados ingresados de manera correcta el sistema procederá finalmente a enviar la notificación a las cuentas establecidas.



*Figura 3.18: Autenticación de la Cuenta del Sistema*

En la siguiente figura se muestra la programación completa con la cual estamos trabajando en el proyecto para que funcione correctamente el envío de notificaciones vía e-mail.

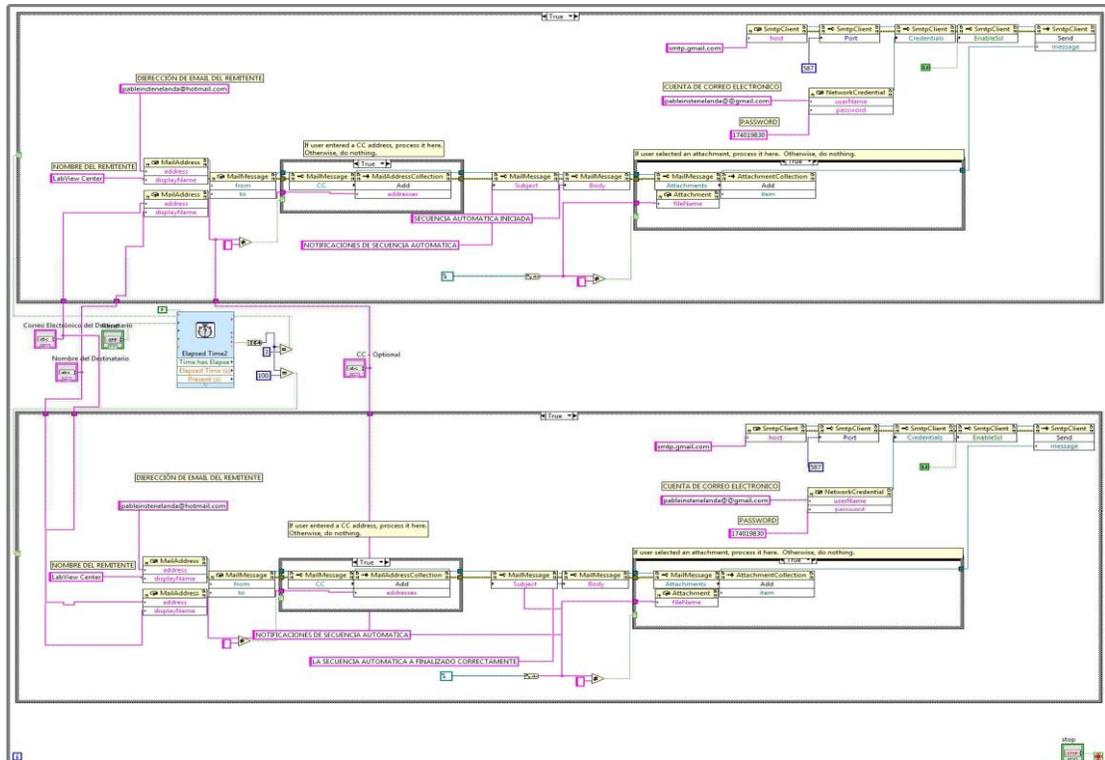


Figura 3.19: Diagrama completo del envío de notificaciones

### 3.3.1.5. Configuración de la Herramienta de Publicación WEB

**Web Server LabVIEW:** Es una herramienta que forma parte del entorno de programación a partir de la versión 8.6 que permite al desarrollador publicar vi's a través de internet, siendo de mucha utilidad en sistemas de monitoreo y control remoto.

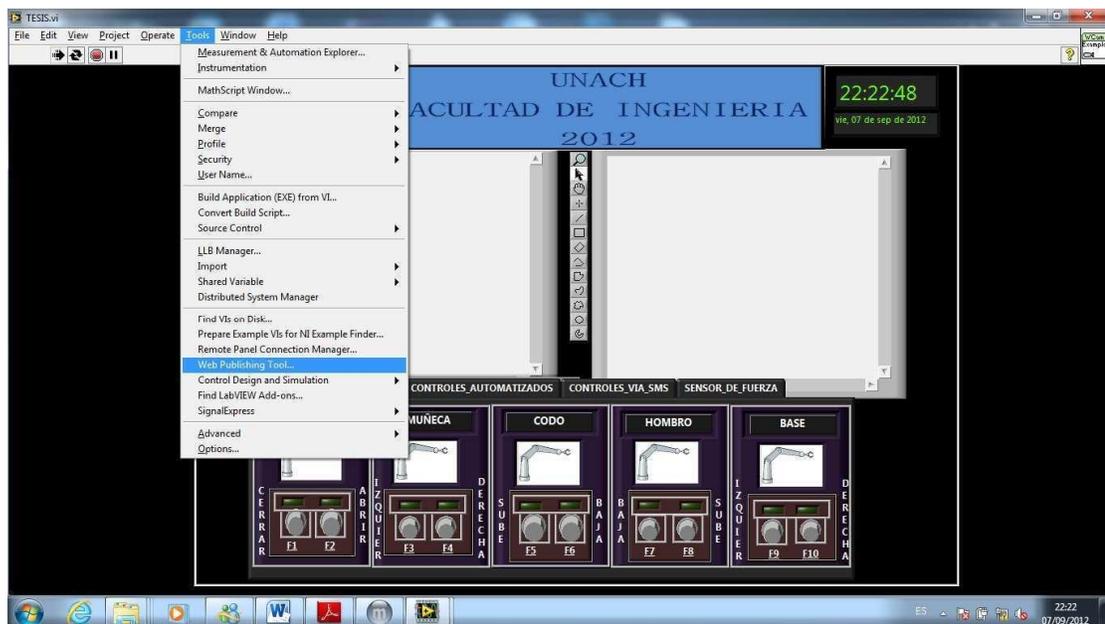
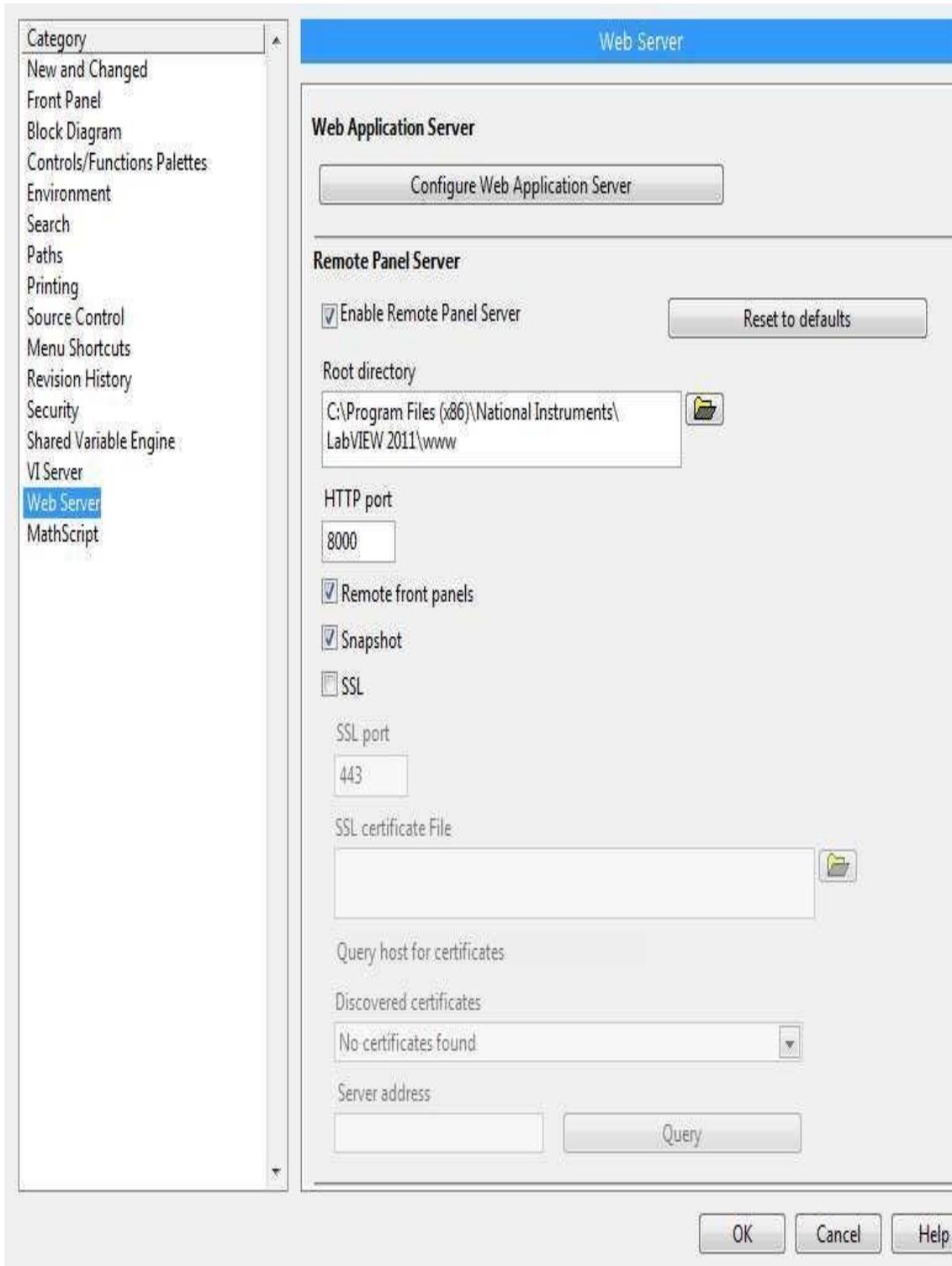


Figura 3.20: Opciones de VI en LabVIEW2011

Configuración del entorno Web Server.

CARACTERÍSTICA	VALOR
Activación del Servidor del Paneles Remotos:	Enable
Directorio Raíz:	
Puerto:	8000
Encriptación:	NO
Directorio archivo LOG:	
VI's Visibles:	* (all)
VI's con Acceso en Navegador Web:	* (all)

Cuadro 3.8: Parámetros de configuración WebServer



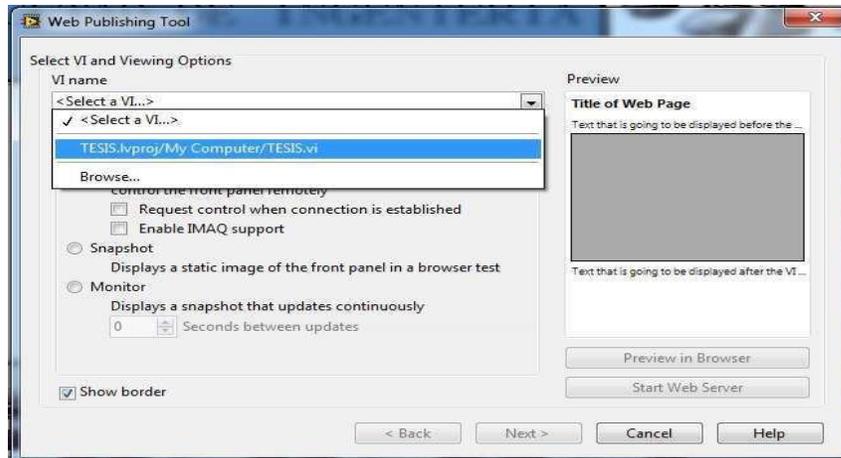
*Figura 3.21: Configuración de WebServer*

Desde la barra de menú se dirigen hacia la sección de TOOLS posteriormente escogemos la opción Web Publishing Tools iniciamos el proceso de publicación web para la aplicación.

Entre las configuraciones que debemos realizar consta:

Selección del VI que será publicado vía web.

## Opciones de visualización.



*Figura 3.22: Publicación Web, selección de VI*

### Opciones de visualización:

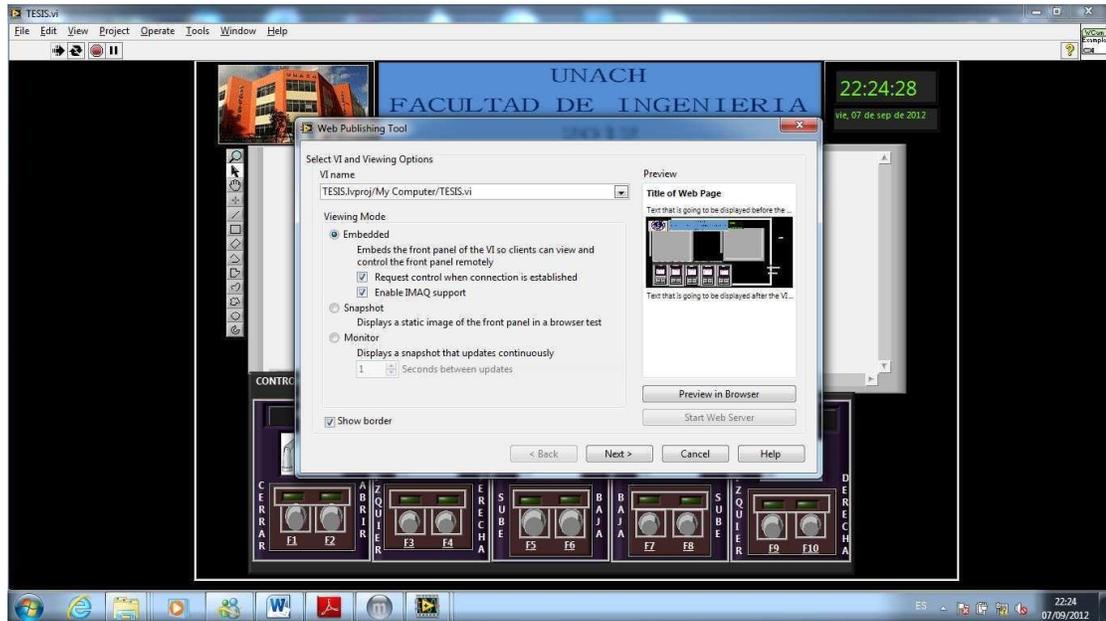
**Embebido:** Muestra la aplicación incrustada en una página web que permite al usuario operar con todas las herramientas disponibles en el panel frontal.

**Snapshot:** Muestra en una página web la imagen con el panel frontal de la aplicación.

**Monitor:** Muestra en una página web la imagen con el panel frontal de la aplicación capas de actualizarse cada un periodo de tiempo determinado, que puede ser utilizado para fines de supervisión.

### Selección de la salida en HTML.

Es básicamente el formato y el contenido que aparecerá en el archivo html que será visualizado a través de Internet Explorer.



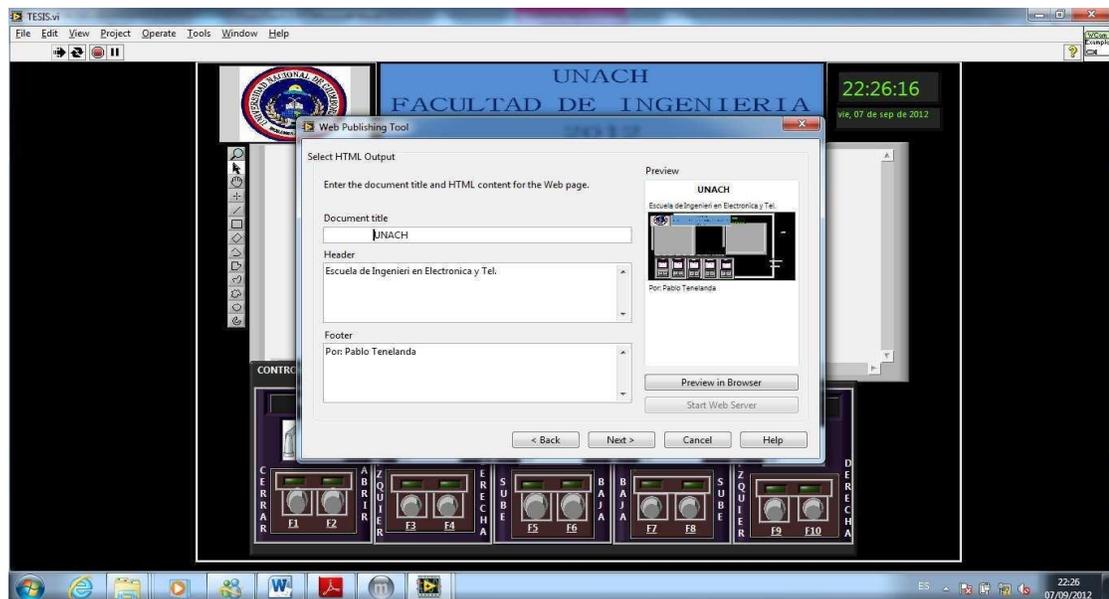
*Figura 3.23: Formato HTML de la publicación web*

Opciones almacenamiento:

Directorio de almacenamiento.

Nombre de archivo.

URL, de acceso en Internet Explorer.



*Figura 3.24: Detalles del archivo HTML creado*

Visualización del panel frontal en Internet Explorer.



*Figura 3.25: Visualización en Internet Explorer*

### 3.3.1.6. Control mediante SMS

El proyecto consta además en su interface en LabVIEW con un sistema diseñado para leer los mensajes de texto el cual son enviados vía teléfono celular y recibidos por un circuito externo el cual permite codificar la información para que el proyecto ejecute las instrucciones especificadas en el mensaje.

#### 3.3.1.6.1. Selección del Microcontrolador

Dentro del mercado existe una gran cantidad de, microcontroladores con diferentes arquitecturas. Se pueden clasificar dependiendo del numero de bits destinado al direccionamiento de la memoria en:

8, 16 y 32 bits.

Los microcontroladores de 16 y 32 bits son los de mayor rendimiento con la desventaja de ser mas caros. La mayoría de aplicaciones se desarrollan con microcontroladores de 8n bits, siendo estos los más populares.

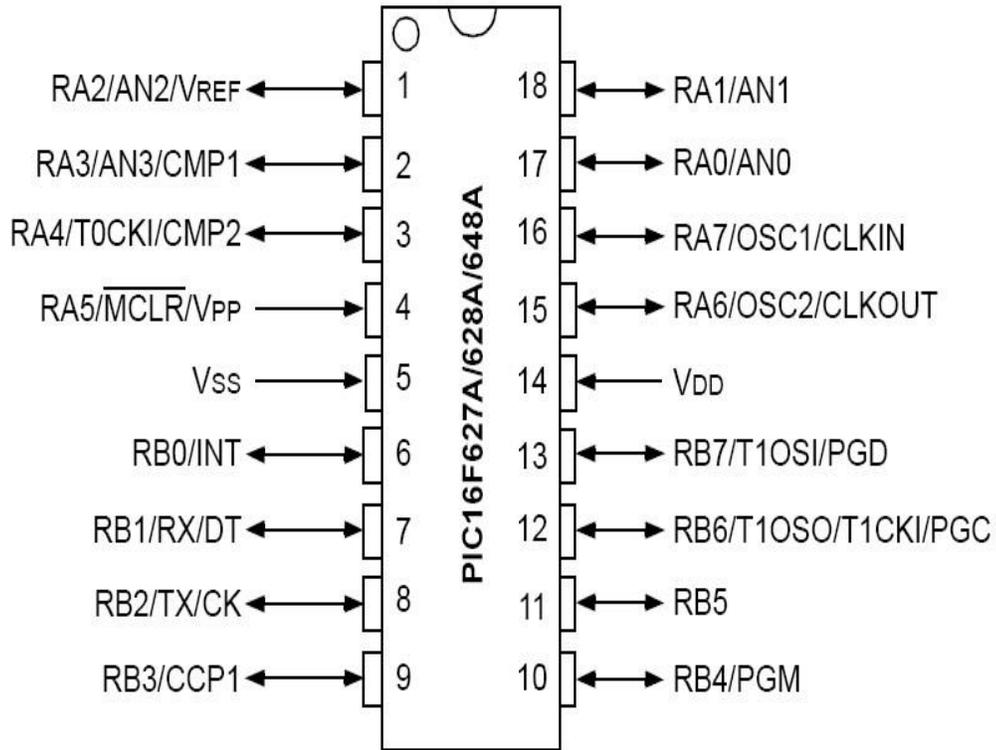
#### 3.3.1.6.2. Características del PIC 16F628A

A continuación se muestra las características del microcontrolador PIC16F628A:

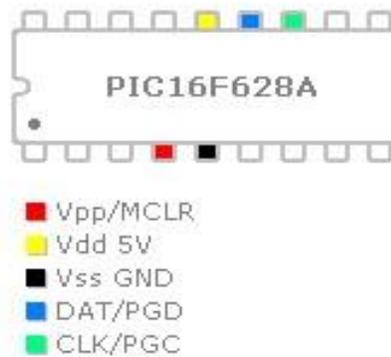
- Microcontrolador de 8 Bits.
- Memoria de programa tipo Flash de 1024 palabras de 14 bits.

- Memoria RAM de 68 bytes.
- Memoria EEPROM de datos de 64 bytes.
- Velocidad de operación de hasta 20 Mhz.
- Cuatro fuentes de interrupción.
- Posee 13 pines I/O (pines de entrada o salida).

### 3.3.1.6.3. Diagrama de pines pinout



*Figura 3.26: Diagrama de pines PIC 16F628A<sup>13</sup>*



<sup>13</sup> <http://www.bilbaoelectronics.com/pines-16f628a.html>

#### **3.3.1.6.4. Selección del dispositivo de comunicación GSM**

Para la selección del dispositivo es importante considerar: la interfaz de comunicación, los protocolos de comunicación, el modo de manejo de datos y la disponibilidad en el mercado.

En el mercado se pueden encontrar modelos de teléfonos los cuales debemos tomar en cuenta que solo ciertas marcas de ellos ofrecen un puerto de comunicación que permitan interactuar con un microcontrolador.

Entre las principales marcas en el mercado se puede distinguir tanto a Sony Ericsson como Nokia las cuales poseen modelos con puertos manejables.

Las principales características que se deben considerar para la selección de los terminales son: puerto serial de comunicaciones, capacidad de envío y recepción de mensajes y protocolo de comunicaciones.

- **Puerto Serial de comunicaciones.-** es necesario que el dispositivo celular posea un puerto de comunicaciones accesible para la aplicación, mediante algún tipo de conector. De igual forma debe existir la suficiente información técnica del dispositivo de manera que no cause ningún tipo de daño en manipulación. El puerto del dispositivo debe permitir la comunicación bidireccional con el sistema microprocesado.
- **Capacidad de envío y recepción de mensajes (Short Message Service)**  
Uno de los servicios que ofrece GSM es la posibilidad de envío de mensajes SMS. Al elegir un terminal GSM se asegura que tanto el terminal de envío como el de recepción tenga la capacidad de recibir mensajes SMS.
- **Protocolo de comunicaciones**  
El protocolo de comunicación del terminal debe permitir la comunicación entre el microcontrolador – teléfono y teléfono – computador.  
Debe ser estandarizado. De acuerdo a los diferentes tipos de marcas de teléfono se ha desarrollado diferentes protocolos de comunicaciones siendo los más utilizados los comandos AT.

#### **3.3.1.6.5. Teléfono Nokia**

---

<sup>14</sup> <http://www.bilbaoelectronics.com/pines-16f628a.html>

Algunos modelos Nokia permiten comunicación por medio de comandos AT y otros por comandos FBUS y MBUS. Posee puertos de comunicaciones y todos los Nokia GSM permiten servicio de mensajería corta. No poseen para el desarrollo aplicaciones con sus protocolos FBUS y MBUS al ser estos propietarios.

- **Protocolo MBUS.-** MBUS utiliza un solo pin para la transmisión y recepción. La transmisión es halfduplex. Son utilizados dos pines Dta y GND. La comunicación con el teléfono se ha 9600 bps, 8 bits de datos, paridad impar, y un bit de parada.
- **Protocolo FBUS.-** utiliza un pin de transmisión de datos, un pin de recepción, y un pin de tierra. Se parece a un puerto de transmisión serial estándar. Trabaja a 115200 bps, 8 bits de datos sin paridad y 1 bit de parada. Tanto la transmisión de MBUS como FBUS se da mediante un formato de trama propietario de Nokia.

#### 3.3.1.6.6. Elección del terminal

Dentro del mercado se pueden encontrar fácilmente terminales tanto Sony Ericsson como Nokia.

Debido a que existe una mayor cantidad de soporte para aplicaciones con terminales Nokia se ha elegido a un teléfono de esta marca para el desarrollo del proyecto.



*Figura 3.28: Nokia 3220*<sup>15</sup>

#### 3.3.1.6.7. Características del Teléfono

El teléfono Nokia 3220m funciona en la red inalámbrica GSM a 850/1800/1900 MHz.

---

<sup>15</sup> [http://www.latest-mobile.com/articles/nokia/nokia-3220\\_3236](http://www.latest-mobile.com/articles/nokia/nokia-3220_3236)

Tiene conexión y transmisión de datos PC Suite para el Nokia 3220 la transmisión que soporta es EGPRS hasta 167.6 Kbps, GPRS hasta 80 Kbps y (HS) CSD hasta 43.2 Kbps.

#### **3.3.1.6.8. Descripción del puerto y cable de comunicación**

Para la comunicación entre el celular y el PIC se requiere de tres hilos, Rx, Tx y GND. En el puerto de comunicación del teléfono estos corresponden a los pines 6, 7 y 8, respectivamente.

Para la conexión se hace uso del cable de conexión de datos correspondiente al Nokia 3220 que es el DKU – 5 (CA – 42). El extremo donde esta el conector USB se corto para llegar e identificar los pines 6, 7 y 8.



*Figura 3.29: Puerto de comunicación Nokia 3220*<sup>16</sup>

El cable de datos del Nokia se lo pudo adquirir localmente, lo cual fue una gran ayuda pues facilito la conexión al puerto de datos para el celular; caso contrario, se habría optado por soldar cables al referido puerto.



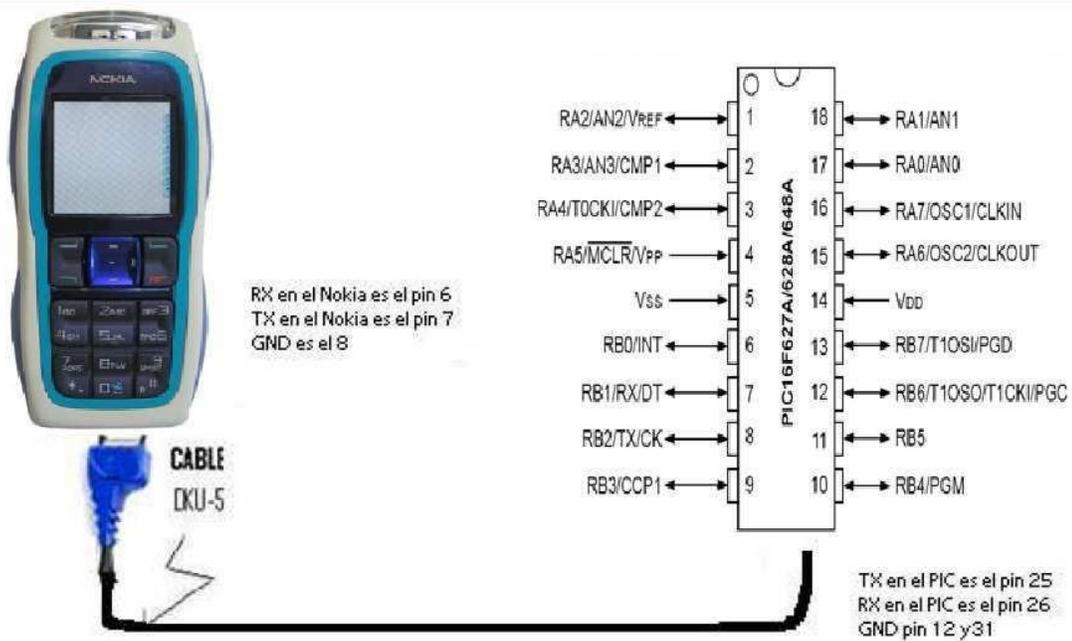
*Figura 3.30: Cable DKU – 5*<sup>17</sup>

Vale la pena mencionar que para la conexión de los terminales se debe cruzar la conexión; esto quiere decir que el Tx del Nokia se debe conectar con el Rx del PIC y viceversa.

---

<sup>16</sup> [http://www.latest-mobile.com/articles/nokia/nokia-3220\\_3236](http://www.latest-mobile.com/articles/nokia/nokia-3220_3236)

<sup>17</sup> <http://ucables.com/ref/CA-42>



*Figura 3.31: Conexión Celular hacia el PIC*

### 3.3.1.7. Funcionamiento

El diseño del circuito se basa en las necesidades del proyecto se debe controlar el brazo con instrucciones vía SMS, para lo cual utilizamos un celular cualquiera para él envío de las instrucciones y para la recepción un Nokia 3220, previamente se ha analizado sus salidas con el cable de datos para obtener únicamente los datos necesitados, el PIC 16F628A, el mismo que se encargara de codificar las instrucciones que arrojará el celular utilizando comandos AT, para que el proyecto ejecute dichos movimientos.

El circuito externo se comunicara con la Pc mediante cable serial adaptado al USB y enviara los datos para que sean leídos por el programa en LabVIEW y de esta manera ejecute los movimientos deseados.

La función VISA del LABVIEW es aquella que permitirá adquirir los datos que provienen desde el teléfono celular, estos datos son adquiridos por el puerto serial COM6 como se muestra en este ejemplo, para que de esta manera el programa ejecute las instrucciones especificadas en el mensaje de texto enviado.

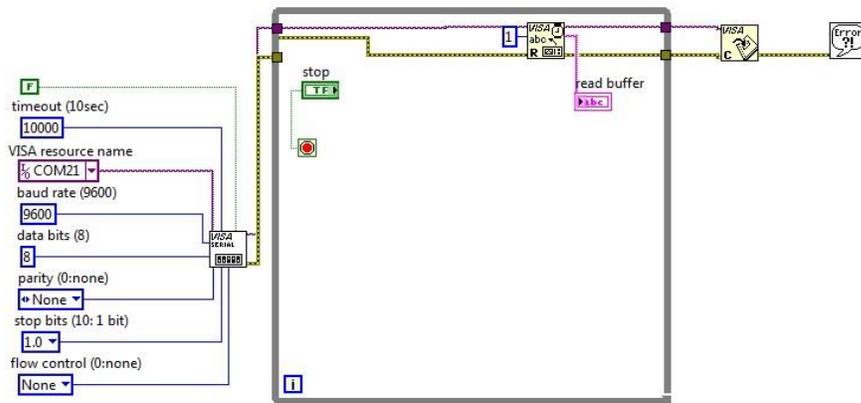
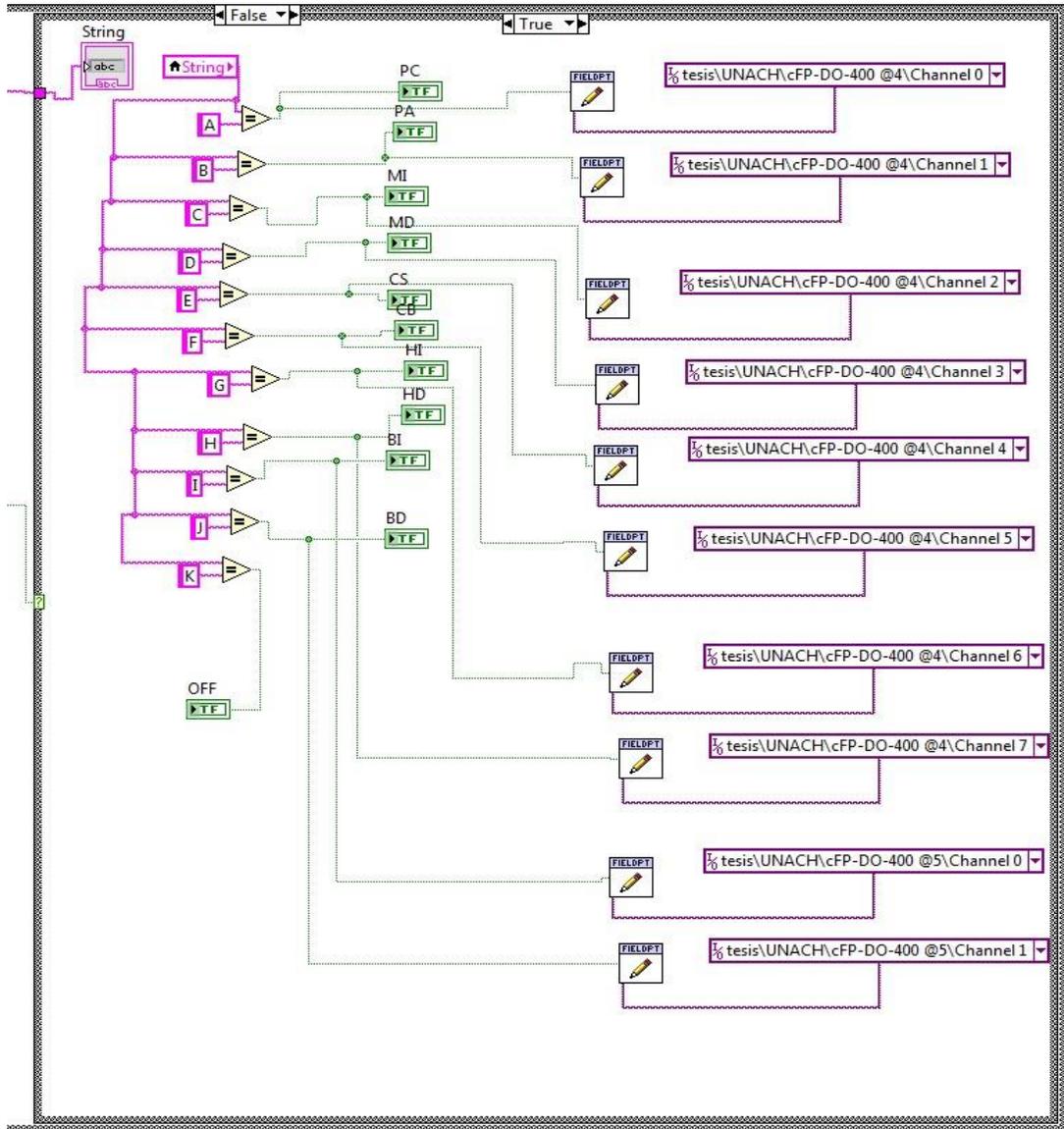


Figura 3.32: Comando de adquisición de datos VISA

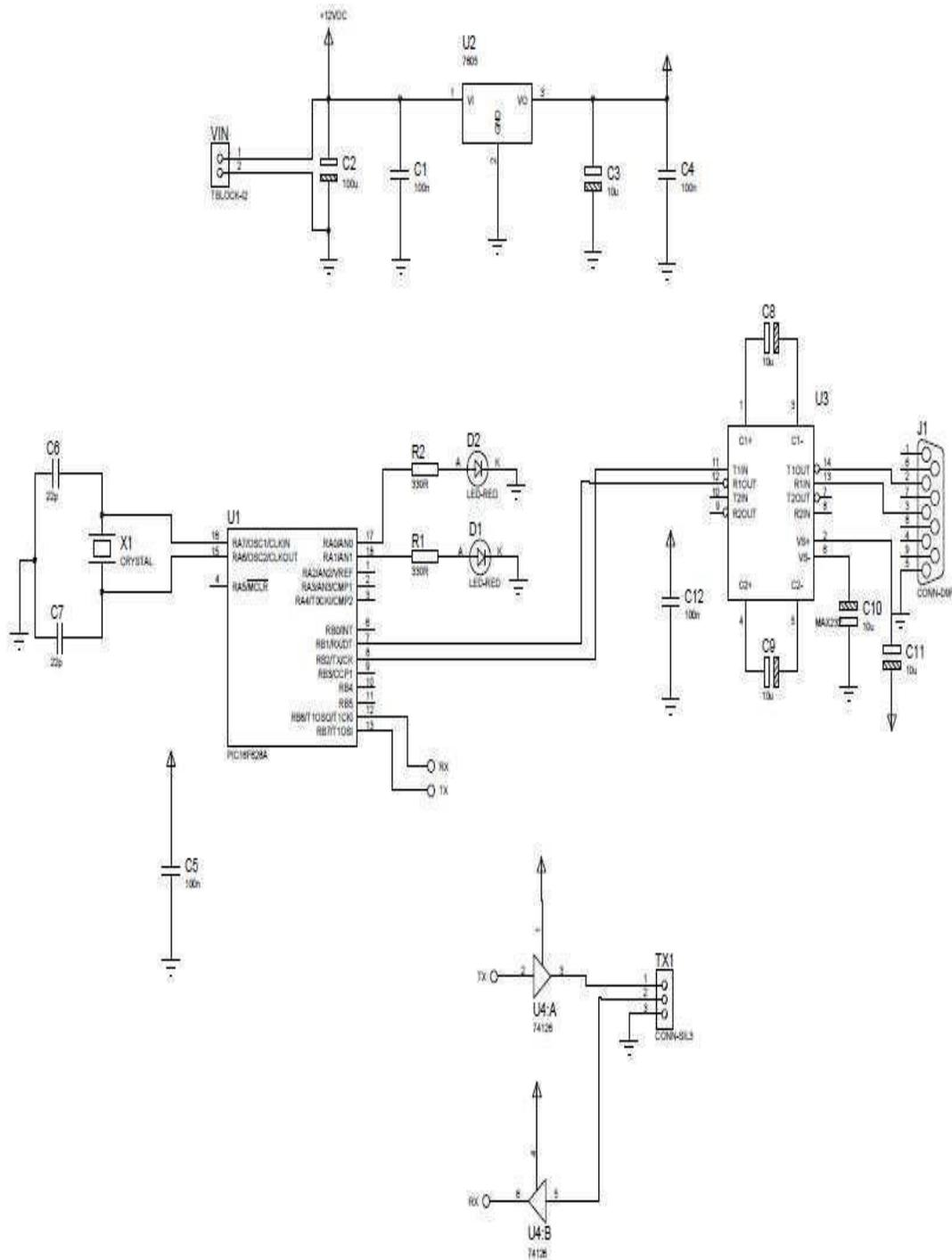
Una vez que el programa a leído el mensaje de texto utilizamos sentencias de comparación para ver cual de los servomotores tiene que ser activado por el proyecto y poner de esta manera un 1 lógico en la salida del CFP y específicamente en el canal que corresponda a el servomotor para que ejecute la instrucción.



*Figura 3.33: Activación de motores según la lectura del SMS*

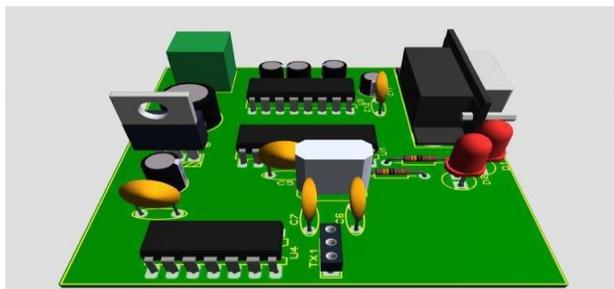
### 3.3.1.7.1. Diagrama Esquemático del circuito externo

En la siguiente figura se puede observar el diagrama de la placa que se ha elaborado para el correcto funcionamiento del proyecto, en el mismo se encuentra un microcontrolador el cual se encargara de hacer la conexión entre el celular y la estación maestra.



**Figura 3.34: Diagrama del Circuito externo**

Una vez montado el circuito en la plaqueta tenemos el dispositivo para conectar el celular con la PC para que puedan ser ejecutadas las instrucciones enviado mediante SMS desde otro celular.



*Figura 3.35: Circuito externo de conexión*

### 3.3.1.7.2. Códigos utilizados por el sistema

Los códigos que se va a enlistar a continuación deberán ser enviados por el usuario con el fin de que el modem los decodifique para que el brazo robótico pueda realizar las acciones correspondientes.

Es necesario mencionar que en este punto los comandos deberán ser escritos por el usuario tal como se muestra en el listado, esto quiere decir, si el comando contiene letras minúsculas se escribirá con letras minúsculas, y si están escritas con letras mayúsculas se deberá escribir con letras mayúscula.

Comando	Acción a ejecutar
Activar pc	Cerrar Pinza
Activar pa	Abrir Pinza
Activar mi	Muñeca Izquierda
Activar md	Muñeca Derecha
Activar cs	Codo Sube
Activar cb	Codo Baja
Activar hs	Hombro Sube
Activar hb	Hombro Baja
Activar bi	Base Izquierda
Activar bd	Base Derecha

*Cuadro 3.9: Comandos enviados por celular*

### 3.3.1.7.3. Instrucciones para la transmisión serial

El programa que se desarrolla en esta aplicación tiene como base la comunicación serial asincrónica y para este propósito el programa MicroCode Studio es de mucha utilidad debido a que se puede emplear una sola instrucción para ejecutar la comunicación serial con un dispositivo, además la salida o la entrada de datos se

efectúe por el pin que se especifique y no es necesario emplear driver RS – 232 debido a que PBP puede invertir la señal de los bits.

Para la con el celular se emplearon los siguientes comandos de configuración:

- **HSEROUT**

Con HSEROUT se envía uno o más ítems el puerto serial de hardware en dispositivos que soportan comunicación serial asincrónica. La instrucción es la siguiente:

HSEROUT [ítem (,item)]

Al trabajar con HSEROUT se tiene que definir en la parte inicial del programa la velocidad de transferencia, setear el registro de transmisión y habilitar el puerto de comunicación serial asincrónico.

- **HSERIN**

Recibe uno o más ítems de un port serial (de hardware) en dispositivos que soportan comunicaciones seriales asincrónicas por hardware.

HSERIN es una de varias funciones seriales asincrónicas pre – construidas. Solo puede ser usada por dispositivos que posean hardware USART. Vea la hoja de datos del dispositivo para información de los pin seriales de entrada y otros. Loas parámetros seriales y el baud – rate son especificados usando DEFINE:

Colocar registro receptor en receptor habilitado

DEFINE HSER\_RCSTA 90h

Coloque el registro de transmisión en recepción habilitada

DEFINE HSER\_TSTA 20h

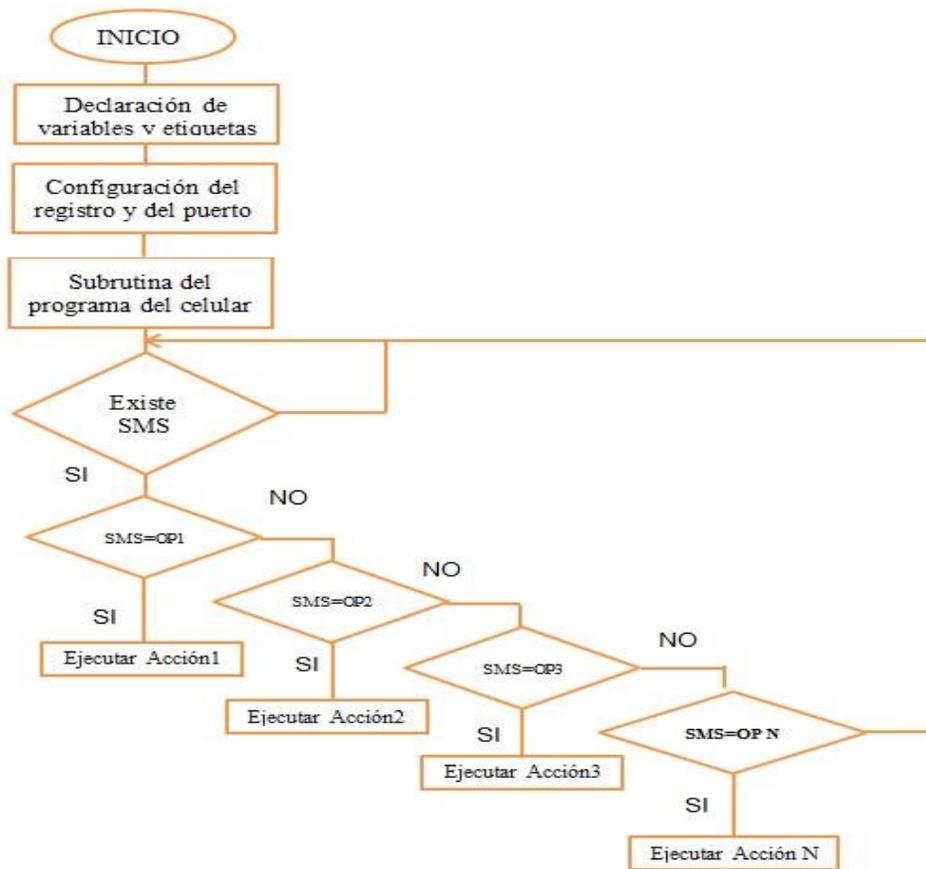
Colocar baud rate

DEFINE HASE\_BAUD 9600

HSERIN asume un oscilador de 4MHZ cuando se calcula el baud rate, para mantener una relación de baud rate apropiada con otros valores de oscilador, use DEFINE para especificar el nuevo valor OSC.

#### **3.3.1.7.4. Programa en MicroCode Studio**

A continuación presentamos el diagrama de flujo del funcionamiento general del programa principal que se ha desarrollado en MicroCode Studio para el funcionamiento.



*Figura 3.36: Diagrama de flujo del funcionamiento del programa*

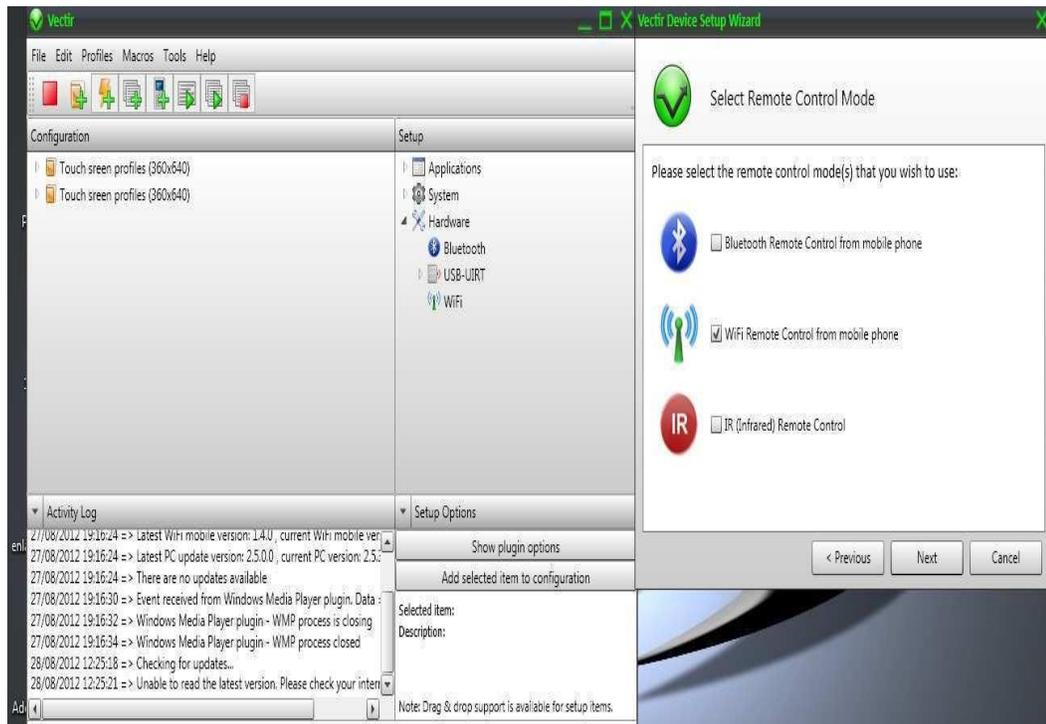
### 3.3.1.8. Monitoreo de los movimientos del brazo mediante celular

Para el monitoreo mediante el celular en el proyecto realizado se ha utilizado el software llamado Vectir el cual permite realizar un escritorio remoto a la PC y de esta manera poder observar y controlar lo que esta sucediendo remotamente con los movimientos del brazo y todo esto en tiempo real.

#### 3.3.1.8.1. Configuración de Vectir

La configuración del software debe ser realizada simultáneamente tanto en la aplicación que se ha instalado previamente en el celular como en la que esta instalada en el ordenador, para que estos dos dispositivos se sincronicen inmediatamente.

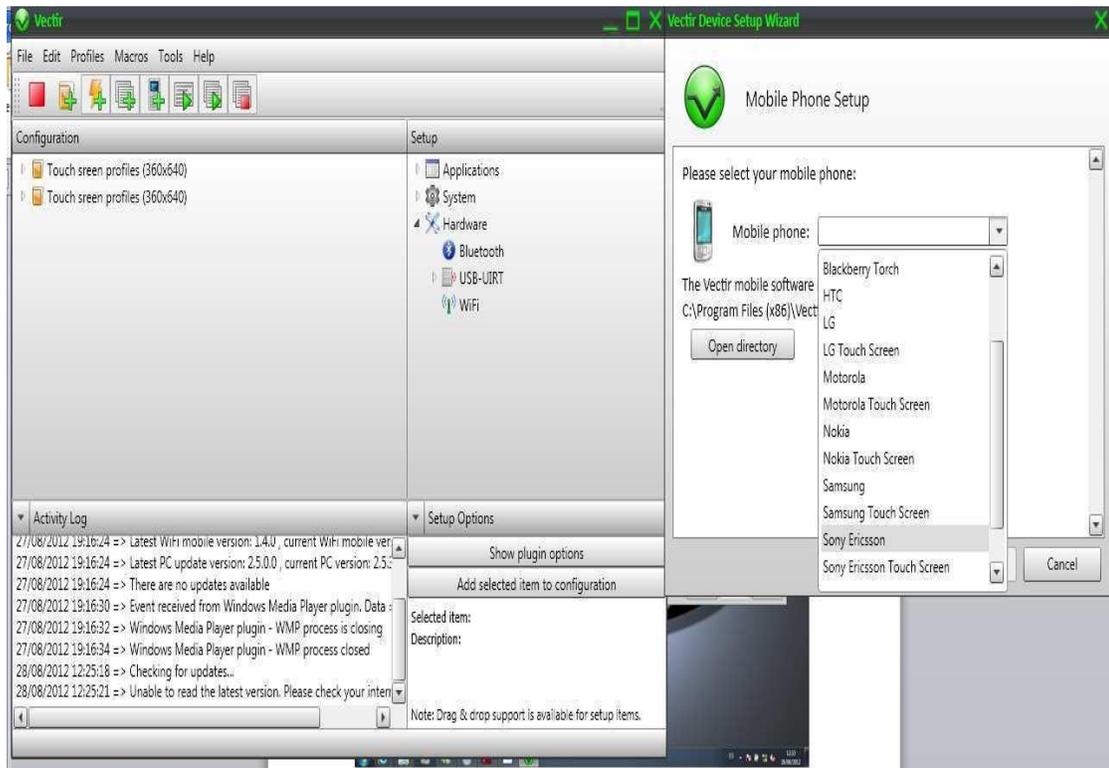
En la siguiente imagen mostramos las bondades de conexión con las cuales dispone este software para la interconexión entre el PC y el teléfono.



*Figura 3.37: Pantalla principal de Vectir y formas de conexión*

Debido a que toda la red de área local será gestionada por un router inalámbrico escogeremos la opción que dice Wi-fi Remote Control From mobile phone, puesto que utilizaremos el wi-fi que esta integrado en el teléfono para que se conecte con el router y de esta manera conectarse a la red para que podamos observar el escritorio de la computadora remotamente en el celular.

El software utilizado es compatible con varias empresas fabricantes de teléfonos celulares por lo cual lo convierte en un pilar fundamental para la ejecución del proyecto.



*Figura 3.38: Diferentes marcas de teléfonos para trabajar*

Luego de escoger la marca del celular con el cual se va a trabajar procedemos a la configuración en el teléfono para lo cual abrimos en el celular la aplicación instalada de vectir, automáticamente la aplicación se abre y como ya esta corriendo la aplicación en la PC el celular se conecta y presenta en su pantalla las opciones para trabajar.

En la siguiente imagen se puede observar la laptop con el programa en LabVIEW corriendo y en la parte inferior derecha se encuentra el celular con el cual se va a trabajar y en su pantalla aparece la imagen del escritorio de la PC.



*Figura 3.39: Escritorio remoto visto en el celular*

## CAPITULO IV

### 4. PRUEBAS Y RESULTADOS

#### 4.1. OBJETIVO:

Determinar la capacidad de trabajo que tiene el sistema de control y verificar el cumplimiento de los objetivos para los cuales fue desarrollado.

#### 4.2. PRUEBAS:

**4.2.1. Pruebas de conexión estación maestra e instrumento de campo.** Parte de la infraestructura integrada que se compone por; una PC que actúa como estación maestra que mantiene el sistema y el servidor web que alojará la aplicación para su publicación y acceso a través de Internet Explorer, un instrumento de campo el cual provee de canales de entrada y salida de información, un circuito diseñado para recibir los mensajes de texto y enviar las instrucciones al programa en LabVIEW, un switch que conecta los dispositivos que contarán con acceso a la red local, un teléfono celular de última generación que permitirá ingresar a la red vía Wi-fi para realizar el monitoreo y control de los movimientos del brazo.

Las pruebas realizadas estrictamente corresponden a la conexión existente entre la estación maestra y el dispositivo CFP conectados a través de un switch y las instrucciones ejecutadas mediante SMS.

Se presenta por tanto a continuación los elementos que intervienen en la conexión:

- Cables de conexión directo UTP
- Switch
- PC Estación Maestra
- Cfp Modelo 2020
- Circuito externo de comunicación
- Dispositivo Celular
- Cable serial a USB

A continuación en la siguiente figura se muestra la topología de red utilizada para conectar ambos dispositivos

#### **4.2.1.1. Configuración de Adaptadores de Red**

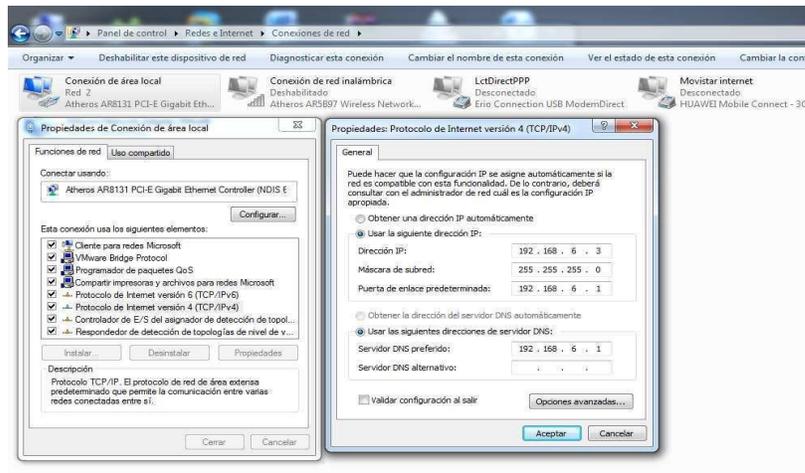
A continuación se muestra los valores de direccionamiento IP que deberán estar presentes en los dispositivos conectados:

<b>CARACTERISTICAS</b>	<b>ROUTER</b>	<b>CFP</b>	<b>ESTACIÓN MAESTRA</b>
Dir IPV4	192.168.6.1	192.168.6.2	192.168.6.3
Máscara	255.255.255.0	255.255.255.0	255.255.255.0
Default Gateway	-	-	-
Primary DNS	192.168.6.1	192.168.6.1	192.168.6.1

*Cuadro 4.1: Parámetros de configuración IP*

Configuración del Direccionamiento IP en Estación Maestra:

1) Panel de Control -> Conexiones de Red -> Conexión de Área Local.



*Figura 4.1: Configuración de Dir. IP en Servidor*

IPCONFIG en Símbolo del sistema para comprobar el direccionamiento IP configurado:

```
C:\Users\Pablo>ipconfig

Configuración IP de Windows

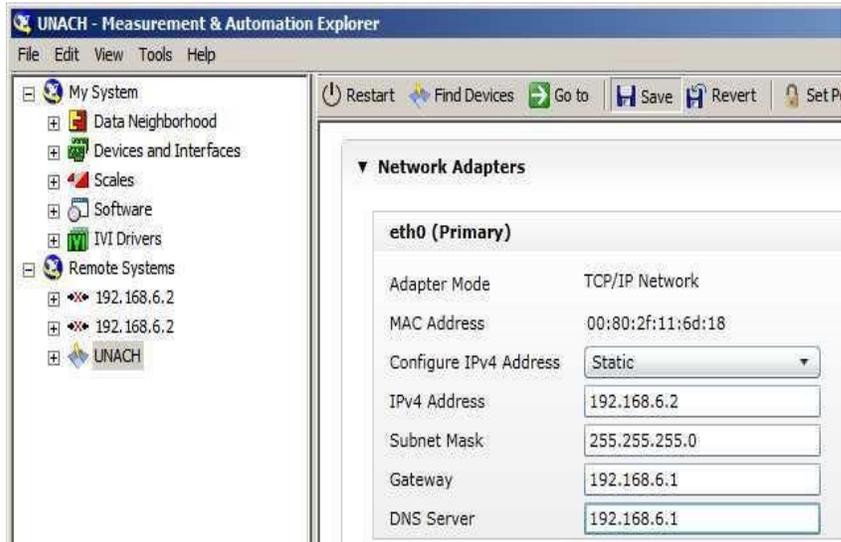
Adaptador de Ethernet Conexión de área local:

    Sufijo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . . . : fe80::8c4d:9c4d:2a96:9957%11
    Dirección IPv4. . . . . : 192.168.6.3
    Máscara de subred. . . . . : 255.255.255.0
    Puerta de enlace predeterminada. . . . . : 192.168.6.1
```

*Figura 4.2: Verificación de Direccionamiento*

2) Measurement Automation Explorer

Antes de empezar a utilizar el proyecto tenemos que configurar al CFP con la dirección IP con la cual se va a trabajar, en el siguiente grafico se puede observar con la cual debe estar equipo.



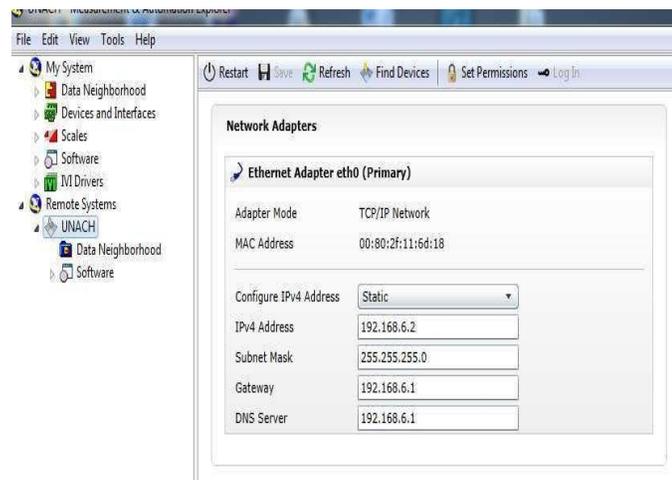
*Figura 4.3: Configuración de <<NETWORK Adapter>> en CFP*

El sistema a través de un mensaje indica que el sistema remoto debe ser reiniciado.



*Figura 4.4: Mensaje de reinicio del CFP*

Al momento de arrancar nuevamente el sistema ya podremos ver el nuevo direccionamiento IP configurado.



*Figura 4.5: Disponible nueva configuración de CFP*

#### 4.2.1.2. Pruebas de conexión

Para realizar las pruebas de conectividad entre los dos dispositivos centrales es necesario validar el direccionamiento que poseen, esto se lo realiza fácilmente en símbolo del sistema en la estación maestra a través del comando **ipconfig-all** que ingresado mostrará en la consola la siguiente información:

```
Adaptador de Ethernet Conexión de área local:

Sufijo DNS específico para la conexión. . . :
Vínculo: dirección IPv6 local. . . . . : fe80::8c4d:9c4d:2a96:9957%11
Dirección IPv4. . . . . : 192.168.6.3
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . : 192.168.6.1
```

*Figura 4.6: IPCONFIG en Server*

Para las pruebas de conectividad optamos por realizarlas con el comando <<ping IP DESTINO>> desde Símbolo del Sistema en la PC.

Verificando que se realizó de manera exitosa.

```
C:\Users\Administrator>ping 192.168.6.2

Pinging 192.168.6.2 with 32 bytes of data:
Reply from 192.168.6.2: bytes=32 time=1ms TTL=60
Reply from 192.168.6.2: bytes=32 time=1ms TTL=60
Reply from 192.168.6.2: bytes=32 time<1ms TTL=60
Reply from 192.168.6.2: bytes=32 time<1ms TTL=60

Ping statistics for 192.168.6.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

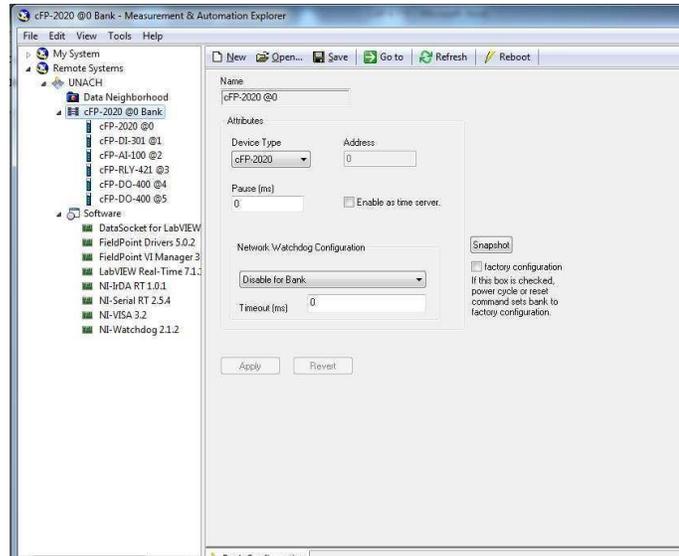
*Figura 4.7: Ping a CFP <<IP: 192.168.6.2>>*

#### 4.2.2. Pruebas de eficiencia en dispositivo cfp y módulos.

Para acceder a la configuración del Módulo CFP es necesario utilizar la herramienta Measurement Automation Explorer.

**MeasureAutomation Explorer:** National Instruments proporciona Measurement & Automation Explorer (MAX), que es una interfaz gráfica que permite al usuario configurar los dispositivos físicos conectados de manera que se puedan integrar de manera fácil a los VI's desarrollados. MAX se instala normalmente con el entorno de desarrollo de aplicaciones LabVIEW.

Podremos ver entonces que la interfaz principal de MAX ha cambiado totalmente, identificamos que en la parte derecha en <<REMOTE SYSTEMS>> encontramos asociado el Compact FieldPoint con la dirección <<192.168.6.2>> y todos los módulos que se hallan acoplados en el hardware.



*Figura 4.8: Información General del CFP*



*Figura 4.9: Módulos Asociados en CFP*

#### 4.2.2.1. Pruebas en Módulos conectados a CFP

Para verificar que los módulos están conectados de manera correcta y verificar que cada uno de los canales esté en perfecto estado se enviará un pulso y se verificará que el LED indicador de cada canal se encienda correctamente en los módulos.

Es necesario especificar que dos canales activaran un motor, tanto en dos sentidos respectivamente.

Para las pruebas a realizar es necesario ubicarse en:

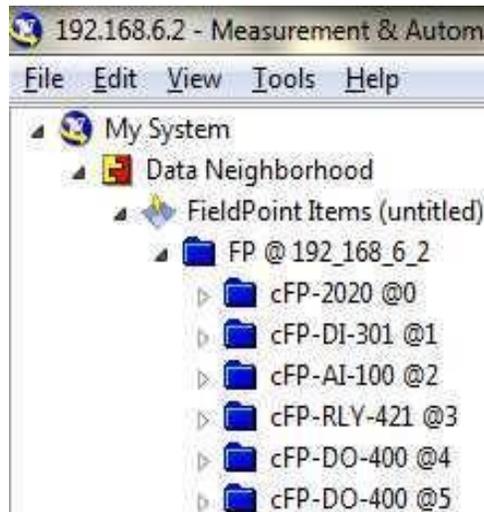
- 1) MySystem

2) Data Neighborhood 3) FieldPointItems

4) FP @192\_168\_6\_2

Por cuestiones de configuraciones y trabajo específicamente afines al proyecto se utilizarán las posiciones 4 y 5 de que contienen los módulos DO 400 que activarán los servomotores para proveer de movimiento al brazo robótico. 5) CFP-DO-400 @4 y CFP-DO-400 @5

Con un total de 5 servomotores utilizaremos 10 canales, 5 de ellos ubicados en el primer módulo DO-400 y los dos restantes del segundo módulo.



*Figura 4.10: CFP asociado y módulos utilizables*

#### 4.2.2.1.1. Pruebas en Módulo CFP-DO-400 @4

Como anteriormente se especificó en el capítulo 2 el Módulo DO 400 tiene a disposición 8 canales para salida de señales digitales, por tanto los utilizaremos para mover 4 de los servomotores que mueven al robot.

Los canales están asociados de la siguiente manera al movimiento de los servomotores:

<u>Nº DE CANAL</u>	<u>SERVOMOTOR</u>	<u>CARACTERISTICA</u>
Channel 0	PINZA	CERRAR
Channel 1	PINZA	ABRIR
Channel 2	MUÑECA	GIRO IZQUIERDA
Channel 3	MUÑECA	GIRO DERECHA
Channel 4	CODO	BAJA
Channel 5	CODO	SUBE
Channel 6	HOMBRO	SUBE
Channel 7	HOMBRO	BAJA

Cuadro 4.2: Permitir Acceso en FIREWALL

Los valores iniciales en **CFP-DO-400 @4** corresponden a <<0>> que significa que no existe salida de señal e información por tanto todos los LED's indicadores del módulo estarán apagados como se aprecia a continuación:



*Figura 4.11: Canales inicializados en 0*

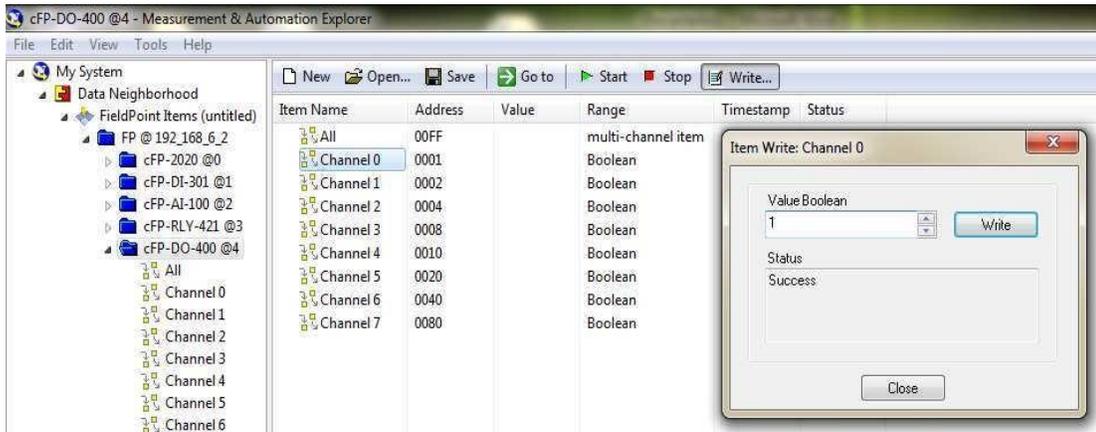
Antes de continuar es necesario verificar que los LED's indicadores de <<POWER>> y <<READY>> se hallen encendidos en cada uno de los módulos.

Cfp-DO-400 @4		
<u>Nº DE CANAL</u>	<u>VALOR INICIAL</u>	<u>ESTADO</u>
Channel 0	0	APAGAGO
Channel 1	0	APAGAGO
Channel 2	0	APAGAGO
Channel 3	0	APAGAGO
Channel 4	0	APAGAGO
Channel 5	0	APAGAGO
Channel 6	0	APAGAGO
Channel 7	0	APAGAGO

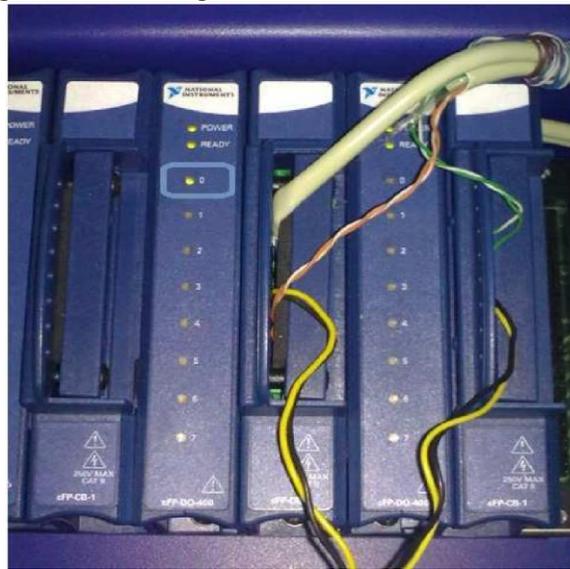
*Cuadro 4.3: Valores por defecto @4*

Para verificar que todos los canales se encuentran funcionando correctamente es necesario asignar el valor <<1>> a cada uno de ellos y verificar que efectivamente el LED correspondiente se encienda, pues de esta manera estaremos comprobando que existe el envío de información.

**CHANEL 0**



*Figura 4.12: Configuración Valor << 1 >>Channel 0*



*Figura 4.13: Channel 0 ENCENDIDO*

Este proceso lo realizamos para cada uno de los canales de salida de la tarjeta de datos para verificar su correcto funcionamiento.

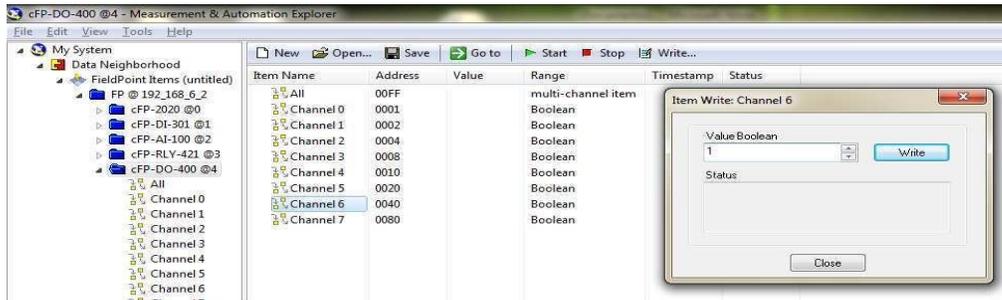
#### 4.2.2.1.2. Pruebas en Módulo CFP-DO-400 @5

En módulo DO 400 situado en la posición 5 los canales están asociados de la siguiente manera al movimiento de los servomotores:

Nº DE CANAL	SERVOMOTOR	CARACTERISTICA
Channel 0	BASE	GIRO DERECHA
Channel 1	BASE	GIRO IZQUIERDA

*Cuadro 4.4: Valores por defecto*

### CHANNEL 0



*Figura 4.14: Configuración Valor << 1 >>Channel 0 @5*



*Figura 4.15: Channel 0 @5 ENCENDIDO*

### 4.2.3. PRUEBAS DE CONEXIÓN ESTACION DE CONTROL Y MONITOREO CELULAR (Estación Maestra y CFP).

#### 4.2.3.1. Configuración de Direccionamiento IP

A continuación se muestra un cuadro con direccionamiento correspondiente a cada estación:

<u>CARACTERISTICA</u>	<u>PC 1</u>	<u>CELULAR</u>
Funcionalidad Remota	MONITOR	CONTROL
Dir IP V4	192.168.6.100	192.168.6.101
Máscara	255.255.255.0	255.255.255.0
Default Gateway	-	-
Primary DNS	192.168.6.1	192.168.6.1

*Cuadro 4.5: Direccionamiento IP en las máquinas conectadas vía Wi-Fi*

Este procedimiento lo realizamos desde la conexión de área local y verificamos con que valores están establecidos tanto para la PC1 como para el Celular, y las IP's que tenemos en la estación maestra y en el CFP.

```

Adaptador de LAN inalámbrica Conexión de red inalámbrica:

Sufijo DNS específico para la conexión. . . :
Vínculo: dirección IPv6 local. . . : fe80::659a:1554:821e:af61%12
Dirección IPv4. . . . . : 192.168.6.100
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . : 192.168.6.1

```

*Figura 4.16: Configuración IP en PC1*

Time	Source	Destination	Protocol	Length	Info
580	195.011028	192.168.6.100	ICMP	74	ECHO (ping) request id=0x0001, seq=633/30978, ttl=128
587	195.011140	192.168.6.3	ICMP	74	Echo (ping) reply id=0x0001, seq=633/30978, ttl=128
588	195.512997	192.168.6.100	ICMP	74	ECHO (ping) request id=0x0001, seq=634/31234, ttl=128
589	195.513123	192.168.6.3	ICMP	74	Echo (ping) reply id=0x0001, seq=634/31234, ttl=128
590	195.869904	192.168.6.3	SSDP	488	HTTP/1.1 200 OK
591	197.013849	fe80::8c4d:9c4d:2a9ff02::c	SSDP	208	M-SEARCH * HTTP/1.1

*Figura 4.17: Verificación de direccionamiento PC1 con Wireshark*

Time	Source	Destination	Protocol	Length	Info
4387	405.039476	192.168.6.3	TCP	1514	58687 > 34966 [ACK] Seq=2145313 Ack=1269 Win=65280 Len=1448 TSval=202719
4388	405.059483	192.168.6.3	TCP	666	58687 > 34966 [PSH, ACK] Seq=2144961 Ack=1269 Win=65280 Len=600 TSval=20
4389	405.059510	192.168.6.101	TCP	66	34966 > 58687 [ACK] Seq=1269 Ack=2139417 Win=92320 Len=0 TSval=174419 Tsv
4390	405.059672	192.168.6.3	TCP	1514	58687 > 34966 [ACK] Seq=2145561 Ack=1269 Win=65280 Len=1448 TSval=202719
4391	405.059675	192.168.6.3	TCP	666	58687 > 34966 [PSH, ACK] Seq=2147009 Ack=1269 Win=65280 Len=600 TSval=20
4392	405.059813	192.168.6.3	TCP	1514	58687 > 34966 [ACK] Seq=2147609 Ack=1269 Win=65280 Len=1448 TSval=202719
4393	405.059817	192.168.6.3	TCP	188	58687 > 34966 [PSH, ACK] Seq=2149057 Ack=1269 Win=65280 Len=122 TSval=20

*Figura 4.18: Configuración Direccionamiento Celular*

#### 4.2.3.2. Pruebas de direccionamiento y conectividad

A continuación verificaremos que el direccionamiento configurado en los adaptadores de red sean los indicados, para lo cual deberemos hacer uso de comando **<<ipconfig DIR IP>>** que muestra el detalle de los valores que se tienen configurados para el establecimiento de comunicación.

#### Ping Estación Maestra a PC1 <<ping 192.168.6.100>>

```

C:\Users\Pablo>ping 192.168.6.100 -t

haciendo ping a 192.168.6.100 con 32 bytes de datos:
tiempo de espera agotado para esta solicitud.
Respuesta desde 192.168.6.100: bytes=32 tiempo=2ms TTL=128
Respuesta desde 192.168.6.100: bytes=32 tiempo=1ms TTL=128
Respuesta desde 192.168.6.100: bytes=32 tiempo=168ms TTL=128
Respuesta desde 192.168.6.100: bytes=32 tiempo=1ms TTL=128
Respuesta desde 192.168.6.100: bytes=32 tiempo=1ms TTL=128
Respuesta desde 192.168.6.100: bytes=32 tiempo=1ms TTL=128

```

*Figura 4.19: Prueba de Conectividad Estación Maestra a PC1*

**Ping a Router <<ping 192.168.6.1>>**

```
C:\Users\Pablo>ping 192.168.6.1
Haciendo ping a 192.168.6.1 con 32 bytes de datos:
Respuesta desde 192.168.6.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.6.1: bytes=32 tiempo=13ms TTL=64
Respuesta desde 192.168.6.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.6.1: bytes=32 tiempo=49ms TTL=64

Estadísticas de ping para 192.168.6.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 3ms, Máximo = 49ms, Media = 17ms
```

*Figura 4.20: Prueba de Conectividad Pc1 – Wi-Fi*

**Ping a CFP <<ping 192.168.6.2>>**

```
C:\Users\Pablo>ping 192.168.6.2
Haciendo ping a 192.168.6.2 con 32 bytes de datos:
Respuesta desde 192.168.6.2: bytes=32 tiempo=3ms TTL=68
Respuesta desde 192.168.6.2: bytes=32 tiempo=3ms TTL=68
Respuesta desde 192.168.6.2: bytes=32 tiempo=7ms TTL=68
Respuesta desde 192.168.6.2: bytes=32 tiempo=3ms TTL=68

Estadísticas de ping para 192.168.6.2:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 3ms, Máximo = 7ms, Media = 4ms
```

*Figura 4.21: Prueba de Conectividad PC1 – CFP*

**Ping a Server <<ping 192.168.6.3>>**

```
C:\Users\Pablo>ping 192.168.6.3
Haciendo ping a 192.168.6.3 con 32 bytes de datos:
Respuesta desde 192.168.6.3: bytes=32 tiempo=28ms TTL=128
Respuesta desde 192.168.6.3: bytes=32 tiempo=2ms TTL=128
Respuesta desde 192.168.6.3: bytes=32 tiempo=3ms TTL=128
Respuesta desde 192.168.6.3: bytes=32 tiempo=8ms TTL=128

Estadísticas de ping para 192.168.6.3:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 2ms, Máximo = 28ms, Media = 8ms

C:\Users\Pablo>
```

*Figura 4.22: Prueba de Conectividad PC1 – Estación Maestra*

**4.2.3.3. Dispositivo Celular**

Para verificar la dirección IP del teléfono celular para proceder con las pruebas de conectividad utilizaremos el programa wireshark, que en una de sus aplicaciones sirve para ver el tráfico que esta atravesando por el router de tal manera que según van

conectándose mas maquinas a la red en la pantalla va a aparecer la dirección IP con la cual esta configurada cada una de ellas.

Como ya vimos anteriormente en la Figura 4.19, la dirección IP del celular es 192.168.6.101, entonces procedemos a realizar la prueba de ping correspondiente hacia esta dirección.

#### Ping al Celular <<ping 192.168.6.101>>

```
C:\Users\Pablo>ping 192.168.6.101

Haciendo ping a 192.168.6.101 con 32 bytes de datos:
Respuesta desde 192.168.6.101: bytes=32 tiempo=1ms TTL=69
Respuesta desde 192.168.6.101: bytes=32 tiempo=1ms TTL=69
Respuesta desde 192.168.6.101: bytes=32 tiempo=5ms TTL=69
Respuesta desde 192.168.6.101: bytes=32 tiempo=2ms TTL=69

Estadísticas de ping para 192.168.6.101:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 1ms, Máximo = 5ms, Media = 2ms

C:\Users\Pablo>_
```

*Figura 4.23: Prueba de Conectividad PC – Celular*

#### 4.2.4. Pruebas de acceso al modo de control remoto

Para acceder al modo de modo de monitoreo y control remoto el usuario únicamente debe digitar la siguiente información en la barra de dirección en el Internet Explorer. <http://192.168.6.3:8000/tesis.html>

Este será la URL que inmediatamente al localizar el servicio mostrará una pantalla en la cual el usuario podrá ver cual es el estado del sistema y además podrá informarse si al momento está funcionando o no de manera correcta, es necesario especificar que este modo de acceso no permite interactuar con la aplicación:



*Figura 4.24: Monitor Remoto desde PC1 en IE*

#### 4.2.5. Pruebas de monitoreo celular

Una vez que tenemos toda la infraestructura de red implementada y el celular listo para ser utilizado como monitor y conectado al router inalámbrico se puede conectar con la estación maestra para poder observar en la pantalla del celular los movimientos del brazo robótico.

En la siguiente figura se puede observar en la parte izquierda la estación maestra en funcionamiento y en la parte derecha la pantalla del celular con el respectivo monitoreo del proyecto.



*Figura 4.25: Monitoreo Celular*

Gracias al software configurado en el celular utilizado para la realización del proyecto se puede dar cuenta que realmente es una herramienta muy útil pues tenemos una visión en tiempo real de lo que sucede con el brazo robótico remotamente y además se puede ejercer un control total sobre el funcionamiento de la estación maestra.

#### 4.2.6. Pruebas de control vía SMS

Ahora procederemos a poner en funcionamiento del proyecto vía mensajes de texto enviados mediante celular.

Procedemos a enviar cada uno de los códigos de activación vía SMS y su verificación en la interface en LabVIEW que esta encendiendo los motores correspondientes a cada código de mensaje enviado.



*Figura 4.26: Panel de Control Vía SMS*



*Figura 4.27: Recepción de PC*



*Figura 4.28: Encendido del CH0 de DO4*



*Figura 4.29: Recepción de PA*



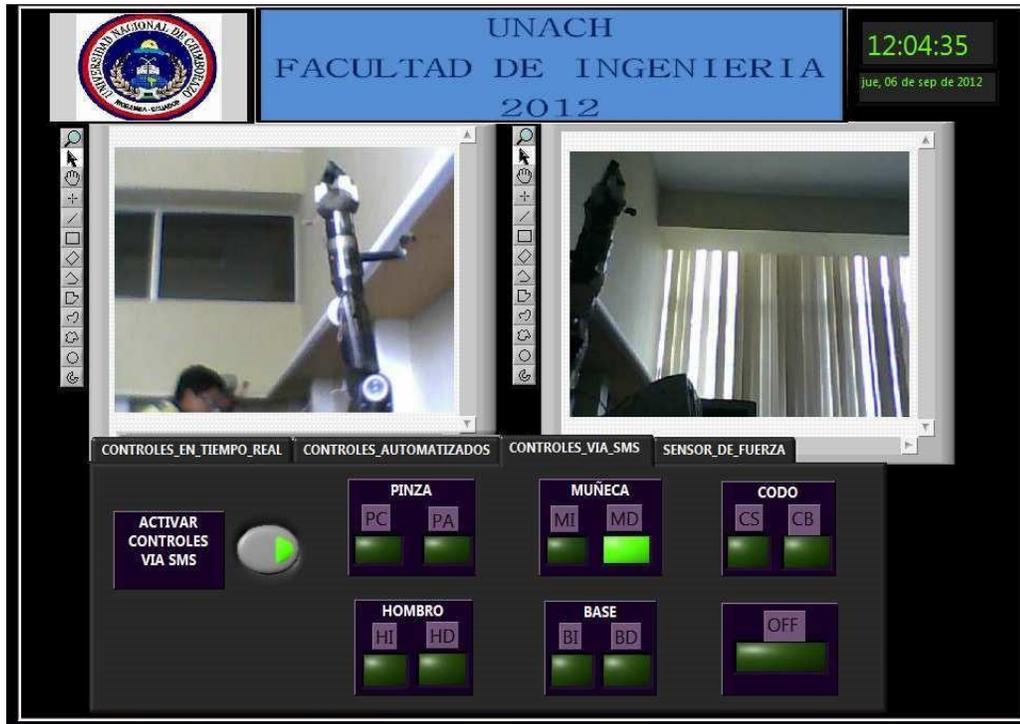
*Figura 4.30: Encendido del CH1 – DO4*



*Figura 4.31: recepción de MI*



*Figura 4.32: Activación del CH2 – DO4*



*Figura 4.33: Recepción de MD*



*Figura4.34: Activación de CH3 – DO4*



*Figura 4.35: Recepción de CS*



*Figura 4.36: Activación de CH4 – DO4*



*Figura 4.37: Recepción de CB*



*Figura 4.38: Activación de CH5-DO4*



*Figura 4.39: Recepción de HS*



*Figura 4.40: Activación de CH6 – DO4*



*Figura 4.41: Recepción de HB*



*Figura 4.42: Activación de CH7 – DO4*



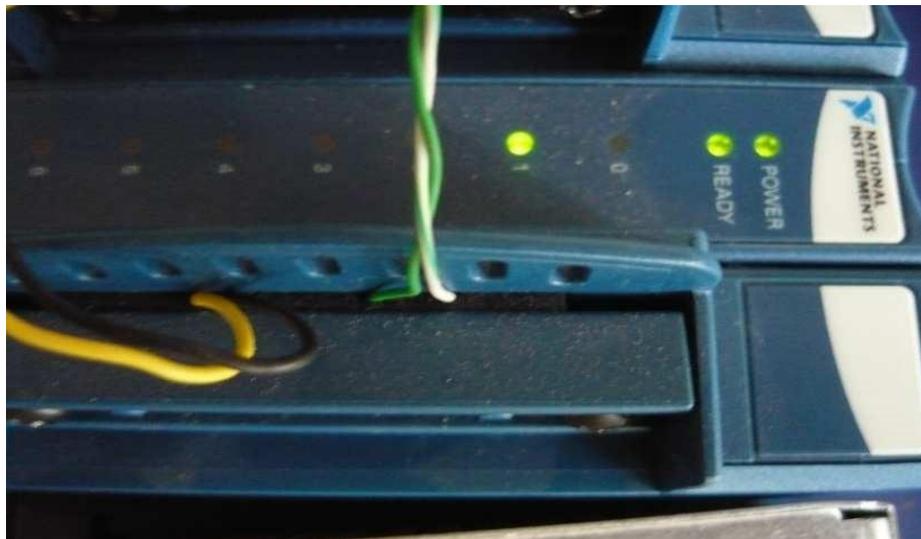
*Figura 4.43: Recepción BI*



*Figura 4.44: Activación de CH0 – DO5*



*Figura 4.45: Recepción de BD*



*Figura 4.46: Activación CH1 – DO5*

#### 4.2.7. Pruebas en envío de notificaciones

Esta funcionalidad permite a la aplicación generar automáticamente una notificación que será enviada a distintas cuentas de correo configuradas en la interfaz de la aplicación en LabVIEW.

A continuación se puede ver cuál es el extracto de la interfaz encargada de generar la notificación:



*Figura 4.47: Sub VI Envío de Notificaciones*

Estas notificaciones serán enviadas al iniciar la secuencia automática y al finalizar la misma, a las cuantas de usuario escritas en el VI.

#### **4.2.7.1. Pruebas de recepción en Buzones**

Una vez iniciada la secuencia que debe cumplir el proyecto automáticamente se dispara un correo electrónico el mismo que contiene el texto, "La secuencia automática a iniciado".

Inmediatamente al accionarse el mecanismo en un máximo de 3(s) la notificación llega hasta las direcciones de correo electrónico que con anterioridad se ha programado en la interfaz.

De igual manera se ha configurado el proyecto para que al terminar la rutina automática se envíe otro correo con el texto, "La secuencia automática a finalizado correctamente".

Como pudimos apreciar anteriormente el correo fue enviado a dos cuentas de correo distintas: pablo\_tenelanda@hotmail.com y pableins666@hotmail.com por tanto procederemos a verificar que efectivamente el mail remitido se haya depositado en los buzones de manera correcta.

Ahora procedemos a la verificación de la notificación en el PC1 a la segunda dirección de correo ingresada para comprobar que las notificaciones están siendo generadas correctamente.

### Tiempo de Retardo: 3(s)

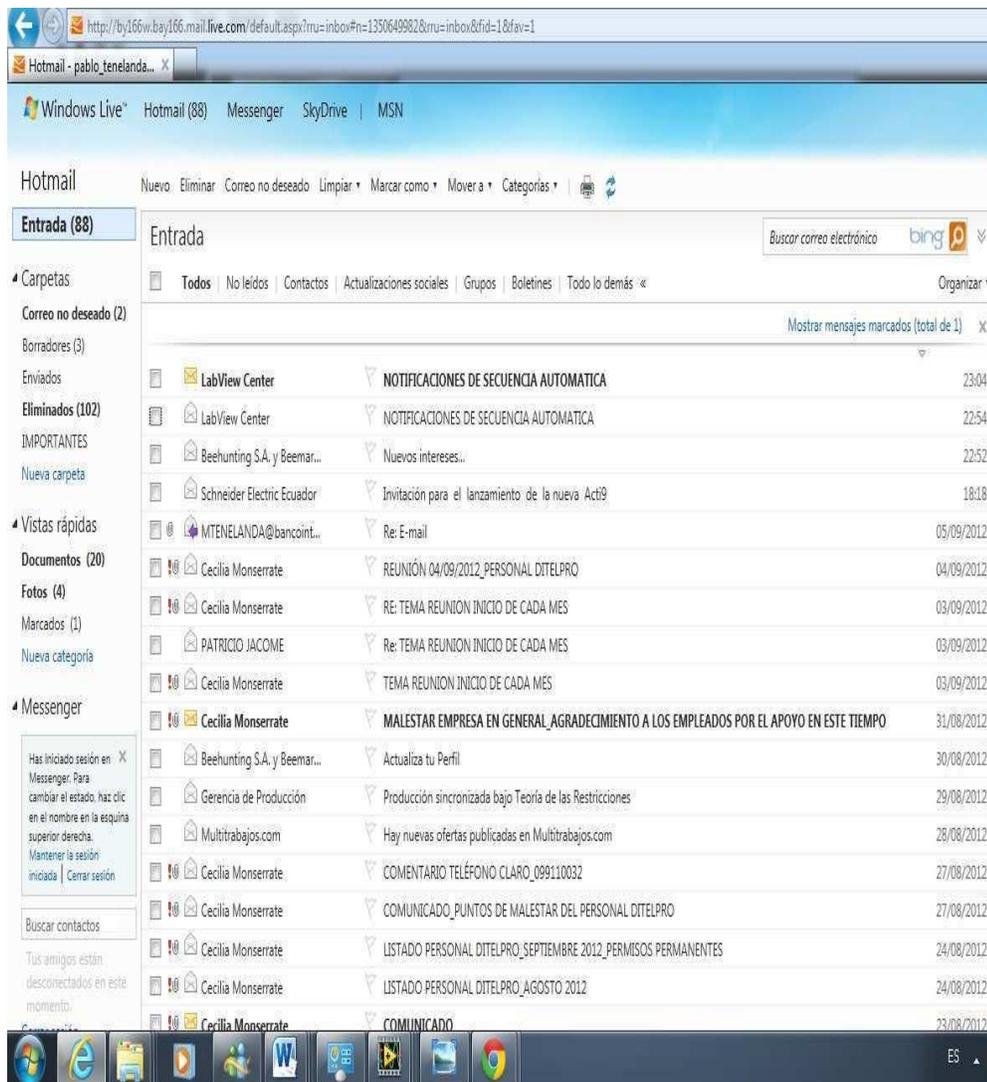
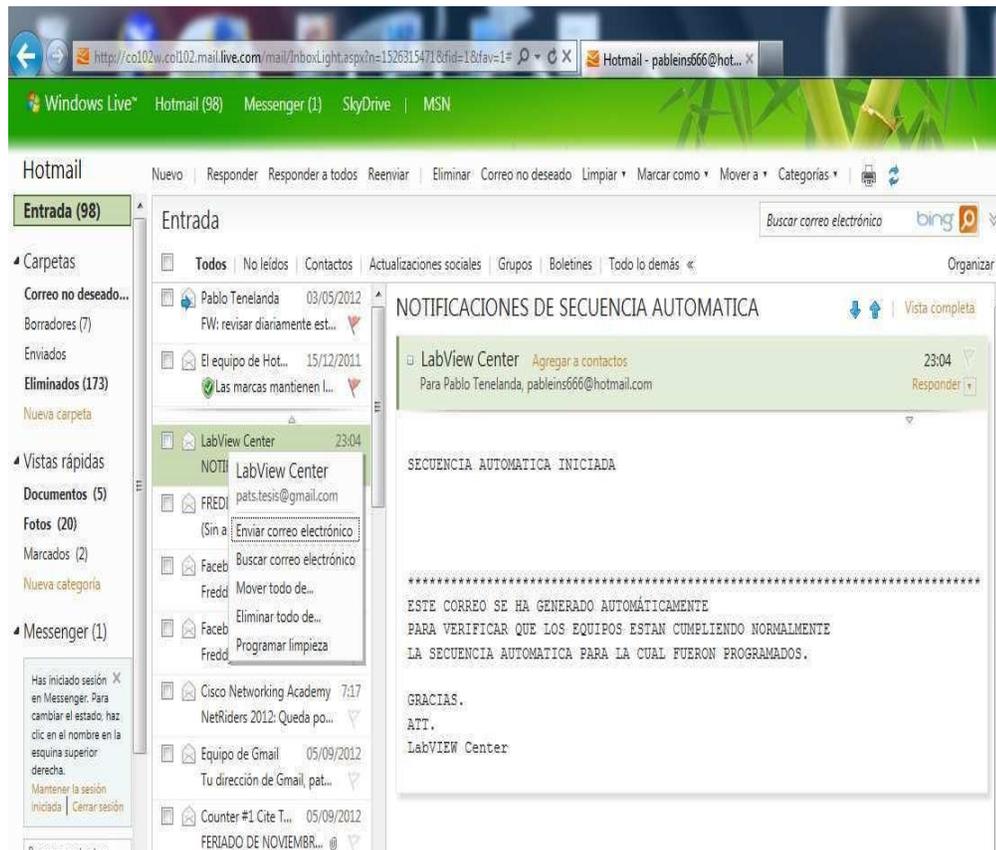


Figura 4.48: RECEPCIÓN Bandeja de Entrada “pablo\_tenelanda@hotmail.com”

### Tiempo de Retardo: 3(s)



**Figura 4.49: RECEPCIÓN Bandeja de Entrada “pableins666@hotmail.com”**

Para los tres casos es necesario validar que la hora y fecha sean correctas y entonces podremos concluir que efectivamente el sistema está emitiendo de manera adecuada las notificaciones.

### 4.3. RESULTADOS

#### 4.3.1. RESULTADOS: PRUEBAS DE CONEXIÓN ESTACIÓN MAESTRA E INSTRUMENTO DE CAMPO.

Debido a que el CFP Modelo 2020 es un dispositivo robusto y de uso industrial, fácil de incorporar a una infraestructura de red ETHERNET debidamente configurado con un direccionamiento IP válido, se puede concluir que la conexión que se realizó con la Estación Maestra fue de lo más sencilla y fácil de realizar.

Algo que es necesario mencionar es la ayuda que brinda el Measurement Automation Explorer más conocido como MAX ya que pone a disposición en su interfaz todas las herramientas necesarias no solo para configurar el direccionamiento IP sino los dispositivos y módulos que se hallan remotamente conectados.

#### **4.3.2. RESULTADOS: PRUEBAS DE EFICIENCIA EN DISPOSITIVO CFP Y MÓDULOS.**

Las pruebas realizadas con cada uno de los Módulos DO 400 y sus respectivos canales fueron realizadas en MAX ya que es necesario verificar el normal funcionamiento de cada uno de ellos antes de integrarlos al proyecto en LabVIEW.

Las pruebas se realizaron de manera exitosa en cada canal y se verificó que ya conectado a la interfaz que comunica el brazo robótico y el CFP efectivamente los servomotores respondían a la salida de las señales digitales.

#### **4.3.3. RESULTADOS: PRUEBAS DE CONEXIÓN ESTACIONE DE CONTROL REMOTO Y MONITOREO CELULAR (Estación Maestra).**

Las pruebas únicamente consistían en asignar un direccionamiento IP valido que permitiese un intercambio de datos entre los distintos equipos, lo que primordialmente se requería es que las estaciones clientes cuenten con acceso hacia el servidor y de igual manera que los puertos de comunicación que son necesarios para el intercambio de información estén abiertos.

#### **4.3.4. RESULTADOS: PRUEBAS DE ACCESO AL MODO CONTROL REMOTO**

El sistema de control en su modo monitor tenía la función específica de otorgar al usuario la posibilidad de observar el panel frontal de la aplicación y de igual manera observar cual es el trabajo que se está realizando sin que este a su vez pueda interactuar de manera directa con la aplicación, siendo totalmente eficiente al momento de realizar las pruebas de acceso vía web

No hay que olvidar tomar en cuenta el número de puerto que se configuró en el WEB SERVER de LabVIEW ya que de ingresar el número de puerto incorrecto no podremos tener a disposición la aplicación.

#### **4.3.5. RESULTADOS: PRUEBAS DE MONITOREO CELULAR**

Las pruebas que se ha realizado para el análisis de los resultados con para el monitoreo remoto de los movimientos del brazo con el celular son exitosas, para lo cual se ha verificado y comprobado que las imágenes que se presentan tanto en la

pantalla de la estación maestra como del celular son idénticas y de la misma manera las dos se mueven simultáneamente con un mínimo retardo en la pantalla del celular.

#### **4.3.6. RESULTADOS: PRUEBAS DE CONTROL VÍA SMS**

En las pruebas realizadas en el laboratorio acerca del funcionamiento del proyecto con las instrucciones enviadas desde un teléfono celular vía mensajes de texto logramos verificar que el proceso se cumple con normalidad las instrucciones son ejecutadas por el miembro robótico con un retardo de aproximadamente 8 segundos en el proceso envió recepción de mensajes entre los dos celulares, la activación de cada uno de los motores se realiza con normalidad, de tal manera que se ha cumplido con el objetivo.

#### **4.3.7. RESULTADOS: PRUEBAS EN ENVIO DE NOTIFICACIONES**

Para las pruebas realizadas con el envió de notificaciones se verificó en todos los casos que estas sean enviadas y depositadas en las cuentas de correo adecuadas, además se pudo observar un retardo que va de 3 a 6(s) en llegar un correo hasta su destinatario.

## **CAPITULO V**

### **5.1. CONCLUSIONES Y RECOMENDACIONES**

#### **5.1.1. CONCLUSIONES**

El software de desarrollo LabVIEW ayuda a los estudiantes a familiarizarse con plataformas de desarrollo aptas para diseño y desarrollo de aplicaciones y sistemas de control industrial dando como resultado sistemas robustos, eficientes y fáciles de utilizar.

Se diseñó un software de monitoreo y control desarrollado en ambiente de programación LabVIEW para el proyecto.

Se diseñó e implementó satisfactoriamente al sistema un circuito externo el cual permite mantener una comunicación estable con el ordenador y además se acoplo sin ningún problema al software desarrollado.

Se implementó un prototipo para monitoreo y control del brazo, cumpliendo los objetivos propuestos como estudiante y de tal manera reafirmamos los conocimientos adquiridos en la Escuela de Ingeniería en electrónica y Telecomunicaciones de la UNACH.

Para desarrollar el proyecto se trabajo con un celular que opera normalmente en la red de Claro pero puede ser sustituido por otro chip de cualquier operadora del país.

## **5.2. RECOMENDACIONES**

Se recomienda leer el manual de usuario y el manual técnico del proyecto, ya que de esta manera se evita causar daños serios a los distintos equipos que se están utilizando, especialmente en los existentes en los laboratorios de la escuela de Ingeniería en Electrónica y Telecomunicaciones.

Si se va a realizar alguna mejora con respecto al software del presente trabajo se debe poseer buenos conocimientos en LabVIEW para poder ingresar o modificar la programación que controla el brazo robótico.

Si el puerto del celular Nokia 3220 no se activa correctamente con el cable de datos se recomienda utilizar una maquina con Windows XP y utilizar el hyperterminal para comprobar que el puerto esta funcionando correctamente, o a su vez instalar una maquina virtual en una PC con Windows 7 para realizar este proceso.

Es aconsejable borrar los mensajes del buzón de entrada del celular que actúa como receptor para ejecutar las instrucciones esto sirve para que tenga una mejor comunicación.

Para ejecutar el hyperterminal en la PC recomendamos instalar antes los drivers del cable que se va a utilizar (DK-U5).

A futuro se podría conectar al sistema una serie de sensores para hacer el sistema más autónomo

Para cualquier cambio a realizarse a la infraestructura del proyecto es necesario que la alimentación eléctrica este completamente apagada y presionar el botón que para la ejecución del mismo.

## **BIBLIOGRAFÍA**

### **Referencias Bibliográficas**

LABVIEW: ENTORNO GRAFICO DE PROGRAMACION

Autor: Lajara Vizcaíno, José Rafael / PelegríSebastiá, José

Editorial: MARCOMBO, S.A.

ISBN: 9788426716965

Páginas: 470

Edición: 2ª

EAN: 9788426716965

PROGRAMACION GRÁFICA PARA INGENIEROS

Autor: Molina Martínez, Jose Miguel / Jimenez Buendía, Manuel

Editorial: MARCOMBO, S.A.

ISBN: 9788426716767

Páginas: 252

Edición: 1ª

EAN: 9788426716767

MICROCONTROLADORES PIC

Autor: Carlos A. Reyes

Edición: 3ª

Estudio de la comunicación con comandos AT y Microcontroladores, caso práctico implementación de un prototipo sistema de gestión de alarmas para viviendas con monitoreo mediante telefonía celular.

Autor: Martha Alulema

Escuela Superior Politécnica de Chimborazo

### **Referencias Electrónicas**

<http://www.wikipedia.com>

<http://www.ni.com/labview/esa/>

<http://www.ni.com/compactfieldpoint/esa/>

<http://www.monografias.com/trabajos68/controlador-motores-dc/controladormotores-dc.shtml>

<http://vnc.softonic.com/descargar>

<http://bluehack.elhacker.net/proyectos/comandosat/comandosat.html>

<http://www.dlinkla.com/home/productos/producto.jsp?idp=70>

[http://www.procobre.org/archivos/peru/conductores\\_electricos.pdf](http://www.procobre.org/archivos/peru/conductores_electricos.pdf)

# ANEXOS

## ANEXO 1 MANUAL TÉCNICO

UNIVERSIDAD NACIONAL DE CHIMBORAZO



**FACULTAD DE INGENIERÍA**  
Escuela de Ingeniería en Electrónica y Telecomunicaciones

CONTROL DE UN BRAZO ROBÓTICO A DISTANCIA DESARROLLADO EN AMBIENTE LABVIEW, SOBRE UNA RED ETHERNET Y ATRAVES DE MENSAJES DE TEXTO VIA TELEFONO CELULAR.

Riobamba, 10 de septiembre del 2012

Desarrollado por:  
Pablo Tenelanda Sanaguano

## Contenido

<b>1. INTRODUCCIÓN</b> .....	135
<b>2. OBJETIVOS</b> .....	136
<b>2.1. OBJETIVO GENERAL</b> .....	<b>136</b>
<b>2.2. OBJETIVO ESPECIFICO</b> .....	<b>136</b>
<b>3. CONTENIDO TECNICO</b> .....	136
<b>3.1. MANUAL DEL SISTEMA</b> .....	<b>136</b>
<b>3.1.1. ESPECIFICACIONES DE LA INTERFAZ</b> .....	136
<b>3.1.1.1. PANEL FRONTAL DEL SISTEMA</b> .....	136
<b>3.1.1.2. Controles en Tiempo Real</b> .....	137
<b>3.1.1.3. Controles Automáticos</b> .....	138
<b>3.1.1.4. Controles vía SMS</b> .....	139
<b>3.1.1.5. Sensor de Fuerza</b> .....	139
<b>3.1.1.6. Visión Artificial</b> .....	140
<b>3.1.2. ESPECIFICACIONES TÉCNICAS DEL FLUJO DE DATOS</b> ...	140
<b>3.1.2.1. PANEL FRONTAL DEL SISTEMA</b> .....	140
<b>3.1.2.2. Controles en tiempo Real</b> .....	141
<b>3.1.2.2.1. Funcionamiento Lógico</b> .....	141
<b>3.1.2.3. Controles Automáticos</b> .....	145
<b>3.1.2.3.1. Funcionamiento Lógico</b> .....	145
<b>3.1.2.3.2. Diseño e implementación en LabVIEW</b> .....	146
<b>3.1.2.4. Controles 3.1</b> .....	155
<b>3.1.2.5. Sensor de Fuerza</b> .....	155
<b>3.1.2.6. Configuración de la Herramienta de Publicación WEB</b> .....	156
<b>3.1.2.7. Visión Artificial</b> .....	161
<b>3.1.2.8. Adquisición de datos VISA</b> .....	162

<b>3.2. CONEXIÓN ELEMENTOS DE CAMPO .....</b>	<b>163</b>
<b>3.2.1. .... Configuración de Adaptadores de Red ....</b>	<b>163</b>
<b>3.2.1.1. .... Pruebas de conexión.....</b>	<b>166</b>
<b>3.2.2.PRUEBAS DE CONEXIÓN EN DISPOSITIVO CFP Y MÓDULOS.</b>	
170 .....	167
<b>3.2.2.2. .... Pruebas en Módulos conectados a CFP .....</b>	<b>170</b>
<b>3.2.3. .... ACCESO EN ESTACIONES REMOTAS ....</b>	<b>171</b>
<b>3.2.4. .... ACCESO DESDE TERMINALES REMOTAS ....</b>	<b>176</b>

## **1. INTRODUCCIÓN**

El sistema que se describe a continuación tiene por objetivo fundamental monitorear y controlar remotamente un brazo robótico, a través de su interfaz el usuario podrá realizar movimientos en tiempo real, movimientos automáticos, enviar notificaciones vía e-mail cuando inicia y termina una secuencia automática, el usuario podrá acceder a la aplicación a través de Internet Explorer propiciando un acceso remoto a la aplicación.

Además nuestro proyecto esta adaptado a un circuito externo y un celular Nokia el cual nos permitirá recibir instrucciones vía SMS los mismos que serán interpretados por LabVIEW para dar instrucciones a ejecutar al brazo robótico.

El sistema está desarrollado totalmente en LabVIEW 2011, y se ha utilizado software como MicroCode Studio para la programación del PIC, además del Vectir un software para poder acceder remotamente a nuestra Pc obteniendo un control total sobre nuestro proyecto.

## **2. OBJETIVOS**

### **2.1. OBJETIVO GENERAL**

Monitorear y controlar remotamente los movimientos de un brazo robótico

### **2.2. OBJETIVO ESPECIFICO**

Permitir al usuario activar los distintos servomotores que forman parte de la estructura electrónica del brazo robótico.

Iniciar una secuencia automática de movimientos del brazo robótico y verificar que las notificaciones vía e – mail lleguen a los correos electrónicos ingresados en la interface.

Permitir el acceso remoto de usuarios vía web a través de Internet Explorer.

Recibir instrucciones a través de SMS enviados mediante teléfono celular.

Obtener monitoreo en tiempo real de los movimientos del brazo robótico en la pantalla de nuestro celular.

## **3. CONTENIDO TECNICO**

### **3.1. MANUAL DEL SISTEMA**

#### **3.1.1. ESPECIFICACIONES DE LA INTERFAZ**

##### **3.1.1.1. PANEL FRONTAL DEL SISTEMA**



Panel Frontal del Sistema

El panel frontal del sistema ofrece al usuario la posibilidad de interactuar de manera directa con el dispositivo electrónico a través del establecimiento de comunicación con el dispositivo de campo, todo esto se logra a través de la utilización de los distintos componentes y herramientas que se pone a disposición el ambiente de desarrollo. Es importante identificar las secciones más importantes con las que el usuario podrá interactuar:

1. Controles en Tiempo Real.
2. Controles Automáticos.
3. Controles vía SMS.
4. Sensor de Fuerza

### 3.1.1.2. Controles en Tiempo Real

A través de esta sección se ha establecido la interacción del usuario con cada uno de los servomotores que son accionados para mover el brazo robótico, se ha establecido por tanto 5 elementos físicos con los cuales se ha trabajado:

- a) Pinza
- b) Muñeca
- c) Codo
- d) Hombro
- e) Base

Cada una de estas partes puede rotar de izquierda a derecha, subir o bajar según corresponda el caso.

Para permitir al usuario realizar movimientos se ha establecido en la interfaz de 10 botones que activan los distintos servomotores, en la interfaz se puede visualizar cual es el movimiento que es realizado y a través del LED indicador que se halla en la parte superior a cada botón el usuario puede saber si en verdad se está ejecutando el movimiento establecido.



*Panel Frontal de Controles en Tiempo Real*

La única condición que se ha tomado en cuenta para la programación de esta parte del sistema es que en el caso de estar dos botones activados en una sola parte no se realice ninguna acción.

### **3.1.1.3. Controles Automáticos**

Esta parte del sistema ejecuta de manera automática una rutina de movimientos al pulsar el botón inicio, la rutina ya ha sido previamente establecida desde la programación.

Los focos que podemos observar en nuestra interface nos indicara que motor esta activo y cual es la acción que esta ejecutando.

Además en esta sección de nuestra interface podemos observar que tenemos la posibilidad de ingresar direcciones de e – mail los cuales nos avisaran el momento que inicia y termina la secuencia automática.



*Panel Frontal de Ejecución Automática*

### 3.1.1.4. Controles vía SMS

La siguiente pestaña es la de los controles vía teléfono celular con mensajes de texto, una vez que activamos los controles con el botón el brazo robótico procederá a realizar la instrucciones que han sido enviados al numero designado para recibir los comandos.



*Panel Frontal Controles vía SMS*

### 3.1.1.5. Sensor de Fuerza

En esta pestaña se encuentra un visualizador que nos permite obtener los valores de voltaje que nos envía desde el exterior un sensor de fuerza que se encuentra colocado en la pinza de nuestro brazo robótico.



*Panel Frontal Sensor de Fuerza*

### **3.1.1.6. Visión Artificial**

Esta parte de nuestra interface nos permite ver en tiempo real los movimientos que esta ejecutando el brazo robótico, obteniendo de esta manera un monitoreo preciso de la actividad del proyecto, con la ayuda de dos cámaras web..

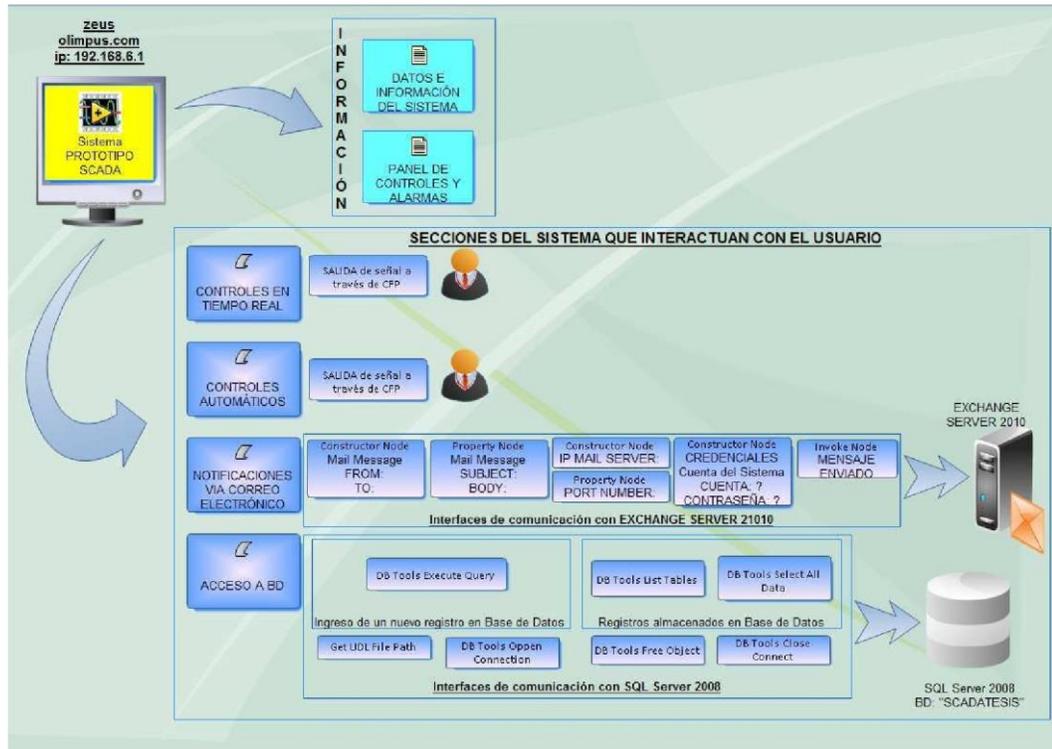


*Panel Frontal Visión Artificial*

## **3.1.2. ESPECIFICACIONES TÉCNICAS DEL FLUJO DE DATOS**

### **3.1.2.1. PANEL FRONTAL DEL SISTEMA**

El sistema descrito a través de su interfaz presenta al usuario dos secciones que están totalmente definidas: la primera encargada de mostrar información como fecha y hora del sistema así como los indicadores de alarmas y en segunda instancia las secciones que actúan directamente con el usuario y que ya han sido descritas anteriormente. A continuación podemos apreciar cual es el flujo de datos e información que existe y las interfaces que han sido utilizadas para comunicar la aplicación otros servicios como el de Base de Datos y el de Correo Electrónico.



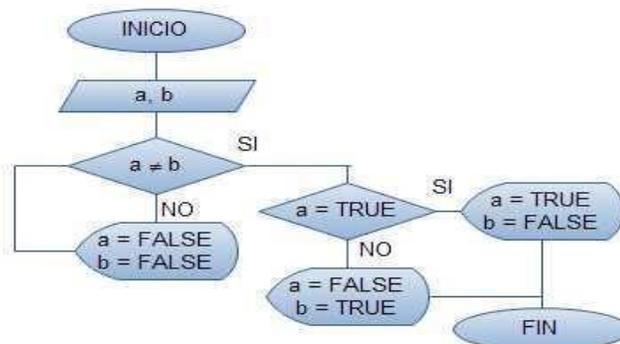
### GESTIÓN DE INFORMACIÓN DEL SISTEMA

#### 3.1.2.2. Controles en tiempo Real

##### 3.1.2.2.1. Funcionamiento Lógico

Esta parte es quizás la más básica pues interactúa de manera directa con el instrumento de campo ya que instantáneamente al ser activado uno motores a través de los controles utilizados en la interfaz el usuario estará moviendo el brazo robótico.

La única condición que se aplica es que en ninguno de los casos se puede activar los dos sentidos para un solo motor, partiendo entonces de este principio el sistema se halla fundamentado en el siguiente esquema



**3.1.2.2.2. Diseño e implementación en LabVIEW**

Siguiendo la lógica de programación anteriormente expuesta hemos utilizado varios elementos de programación propios de LabVIEW, como:

<b>NOMBRE</b>	<b>ELEMENTO</b>	<b>DESCRIPCIÓN</b>
CONTROL BOOLEANO (BOTON)		Enciende o apaga un servomotor.
CONDICIONAL (NO ES IGUAL)		Arroja un valor booleano (1 – 0) al ejecutar la condición.
CONSTANTE BOOLEANA (FALSE)		Es una constante booleana que siempre arroja el valor FALSE.
SELECT		Retorna un valor dependiendo de la condicional especificada.
INDICADOR BOOLEANO (LED)		Indicador que se enciende o apaga de ser el caso.

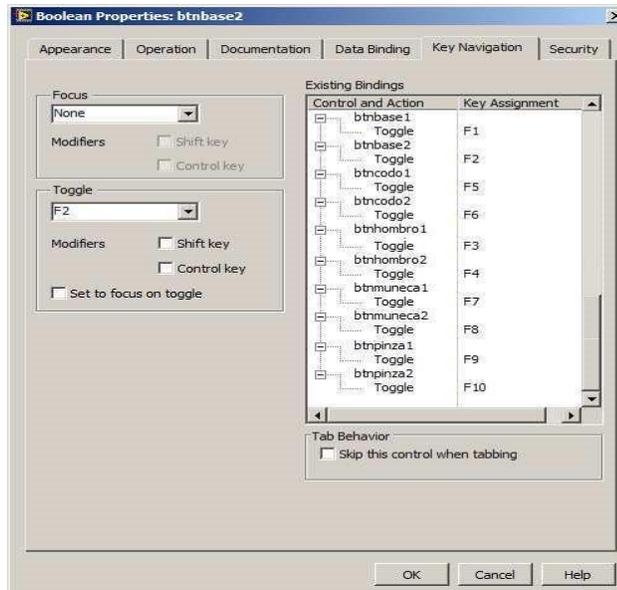
*Elementos de Programación*

En la siguiente tabla de referencia se especifican los botones que se encuentran configurados en la interface así como los servomotores que se activan con cada uno de ellos y la tecla específica que lo enciende o apaga.



*Panel Frontal, Controles en Tiempo Real*

A continuación se muestra los controles que activan los servomotores que propician los movimientos de los distintos servomotores.



Configuración de Key Navigation

BOTON	VALOR	ESTADO
ABRIR	False	APAGADO
CERRAR	False	APAGADO

BOTON	VALOR	ESTADO
ABRIR	True	ENCENDIDO
CERRAR	False	APAGADO

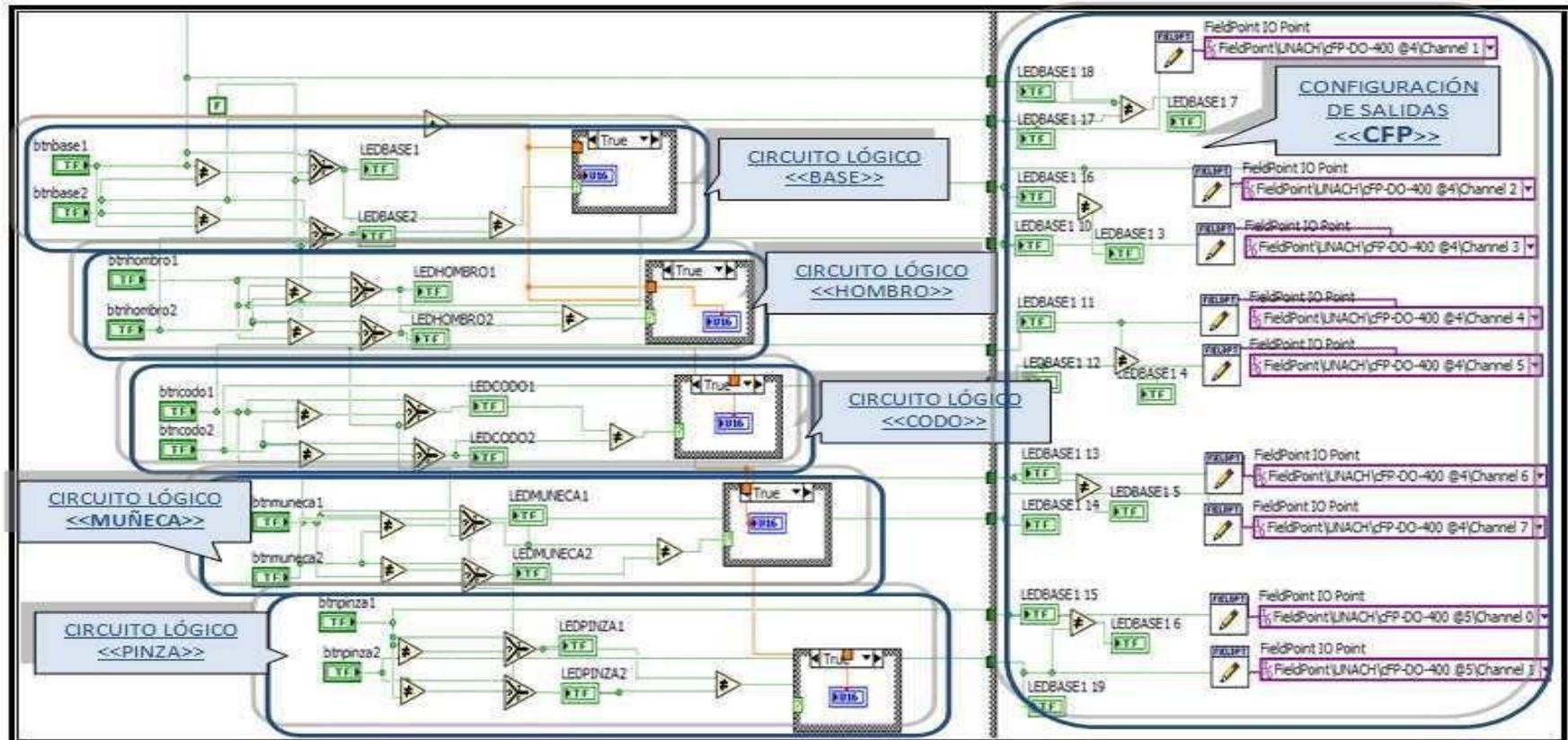
BOTON	VALOR	ESTADO
ABRIR	False	APAGADO
CERRAR	True	ENCENDIDO

Estados de motores en ejecución

El diagrama de bloques con la programación de esta sección se presenta en la siguiente imagen, en la cual podemos encontrar la representación de los botones utilizados para activar los motores y las salidas de información que se ha utilizado para activar los distintos canales en el módulo DO 400.

## DIAGRAMA DE BLOQUES

### PANEL DE CONTROLES EN TIEMPO REAL



### 3.1.2.3. Controles Automáticos

#### 3.1.2.3.1. Funcionamiento Lógico

El panel de controles automáticos fue diseñado con el objetivo fundamental de permitir al usuario ejecutar una rutina automática siendo innecesaria la presencia de un operador.

Para desarrollar una rutina de movimientos se ha procedido a realizar un análisis de tiempos en vista de los motores y canales que serán activados. A continuación se presenta los valores configurados y los elementos que trabajaran:

Nº	CF P	CHANN EL	TIEMPO (s)	MOVIMIEN T O
1	5	1	5	BASE DERECHA
2	4	7	5	HOMBRO BAJA
3	4	4	8	CODO BAJA
4	4	1	2	PINZA ABRE
5	4	0	2	PINZA CIERRA
6	4	6	2	HOMBRO SUBE
7	5	0	10	BASE IZQUIERDA
8	4	7	2	HOMBRO BAJA
9	4	1	2	PINZA ABRE
10	4	6	2	HOMBRO SUBE
11	5	1	10	BASE DERECHA
12	4	7	2	HOMBRO BAJA
13	4	0	2	PINZA CIERRA
14	4	6	2	HOMBRO SUBE
15	5	0	10	BASE IZQUIERDA

16	4	7	2	HOMBRO BAJA
17	4	1	2	PINZA ABRE
18	4	6	2	HOMBRO SUBE
19	5	1	6	BASE
				DERECHA
20	4	5	9	CODO SUBE
21	4	6	4	HOMBRO SUBE
22	4	0	2	PINZA CIERRA

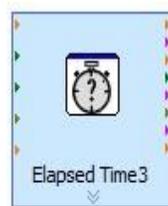
Estado de Servomotores en secuencia

Lo esencial para la programación, es entonces tomar un tiempo inicial y sumar cada uno de los valores siguientes para ir moviendo de manera coordinada un motor por vez de tal manera que al finalizar la secuencia de movimientos pueda repetirse sin interrupciones.

### 3.1.2.3.2. Diseño e implementación en LabVIEW

Tomando en cuenta los tiempos y movimientos anteriormente expuestos es necesario programarlos en LabVIEW de acuerdo a las herramientas que nos ofrece en casos de control de tiempo.

El elemento más importante será por tanto el que se muestra a continuación:



Elapsed Time

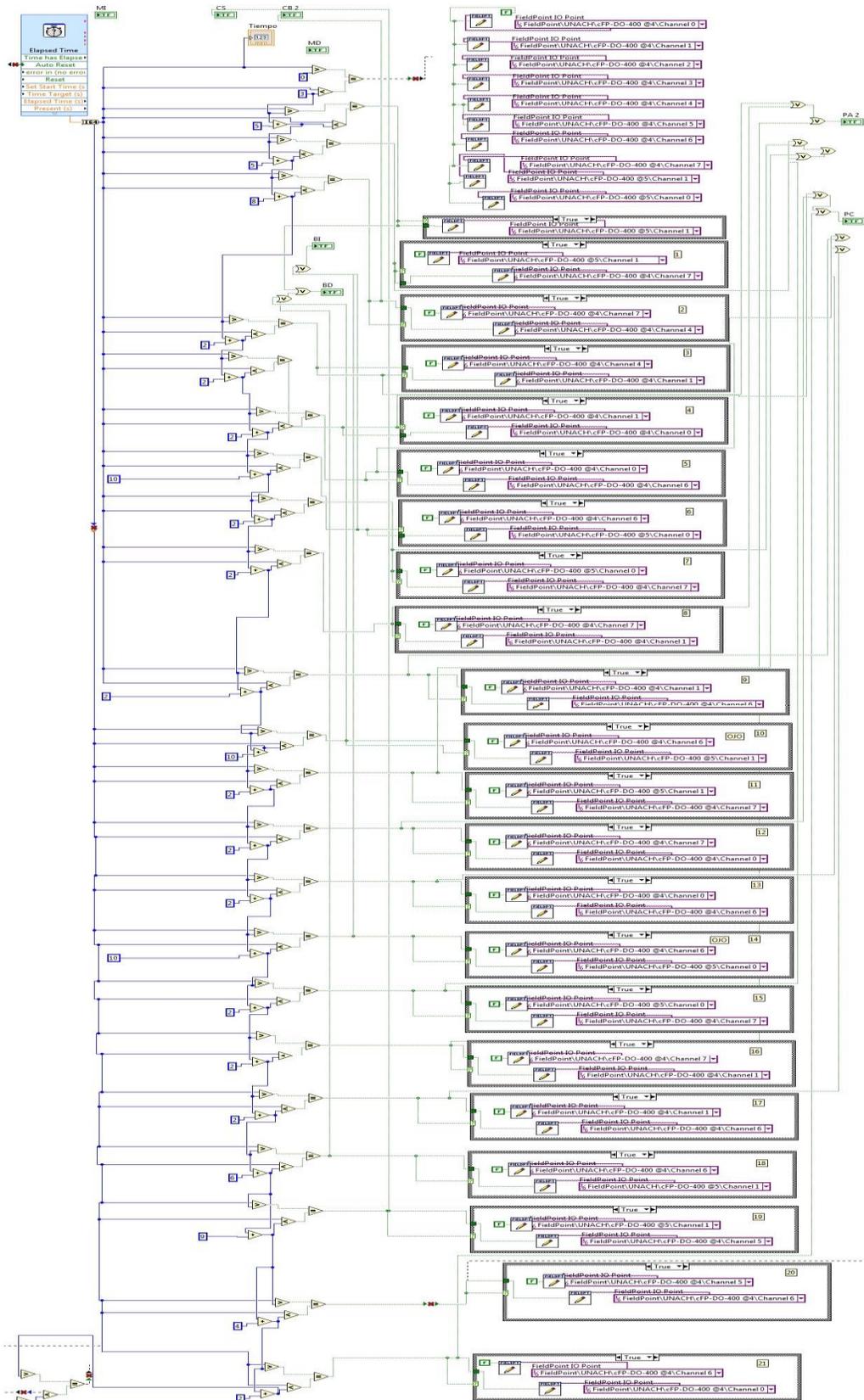
Que se encargará de ejecutar un lapso de tiempo repetitivo o no, dependiendo de su configuración.

Como se supo indicar anteriormente la suma de todos los tiempos es igual a 78 segundos, por tanto una vez concluido este periodo la secuencia se repetirá dependiendo de la rutina.



### Configuración Elapsed Time

A continuación se muestra cual es el esquema de la programación en diagramas de bloques con toda la lógica ya transcrita a LabVIEW.



*Diagrama de Bloques Controles Automáticos*

## **Detalles de Notificaciones Vía Correo Electrónico**

### **Funcionamiento Lógico**

Este sistema de control está compuesto por un módulo de envío de notificaciones vía e – mail debidamente configuradas y establecidas, en base a lo expuesto se ha incorporado una sección que incorpora cuentas de correo destino, descripción del estado de la secuencia.

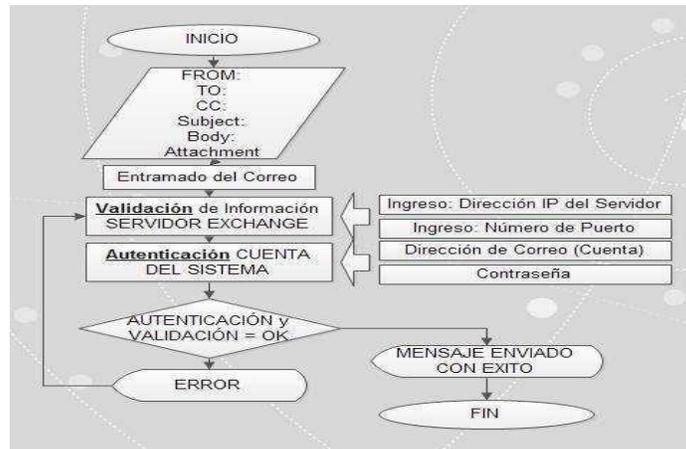
La función principal que desempeña la sección de envío de notificaciones es informar al usuario como esta desempeñando sus tareas el brazo, y esto lo realizamos a través de correos electrónicos enviados a las cuentas usuarios configurados.

El proceso se inicia con el ingreso de las cuentas de correo (remitente, destinatario), posteriormente se puede establecer si se desea incorporar una nueva cuenta a la que se enviará una copia del correo para constancia del error.

El Subject y el Body indicará información adicional del proceso que realiza el proyecto, la hora y fecha de identificación y adicional a todo ello se adjunta un archivo con el informe del último proceso culminado, para finalmente empaquetar la información o realizar el entramado del mensaje.

Por separado se debe realizar la autenticación de la cuenta de correo del sistema especificando cual es la dirección IP del servidor de correo y el puerto de comunicación, una vez que se ha realizado la autenticación exitosamente el mensaje puede ser enviado a las cuentas configuradas.

A continuación se presenta el flujograma que muestra cual el orden de pasos que cumple esta sección del sistema.



Flujograma, Envío de Notificaciones vía Correo Electrónico

### Diseño e implementación en LabVIEW

Siguiendo la lógica de programación anteriormente expuesta hemos utilizado varios elementos de programación propios de LabVIEW, como:

NOMBRE	ELEMENTO	DESCRIPCIÓN
Get Date / Time String		Toma la hora que marca el sistema.
Constructor Node		Constructor .NET que permite al desarrollador ingresar una cuenta de correo electrónico.
Constructor Node		Constructor. NET: configurado para ingresar una cuenta de correo electrónico.
Constructor Node		Constructor .NET que permite al desarrollador ingresar una cuenta de correo electrónico.
PropertyNode		InvokeNode, permite enviar una copia del correo electrónico a otra cuenta de correo.
Constructor Node		Constructor .NET permite ingresar el nombre de cuenta y contraseña de autenticación.
Constructor Node		Recibe la Dirección IP del servidor SMTP
PropertyNode		Especifica el Número de puerto por el que transmite la información el servidor SMTP

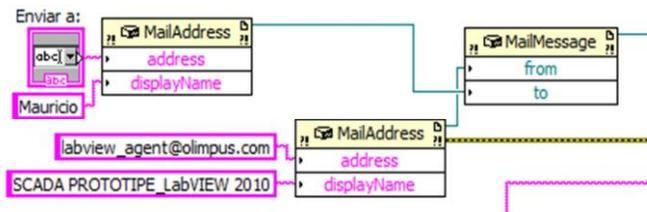
PropertyNode		Autentica los parámetros ingresados de inicio de sesión de una cuenta en el servidor SMTP
InvokeNode		Concluye el proceso y envía el mensaje.

Elementos de Programación

e) Ingreso de Información (Contactos, Correo)

Como parte inicial en el proceso de envío de notificaciones vía correo electrónico es indispensable especificar las cuentas de correo de el usuario que envía así como del que recibe la información.

En la siguiente imagen se muestra la programación realizada en el diagrama de bloques que permite al usuario especificar cuáles son las cuentas de usuario que estarán configuradas.



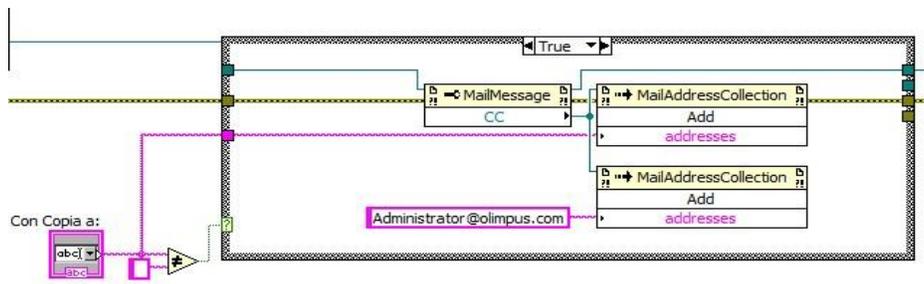
Información de Contactos

f) Envío de una Copia del Correo Electrónico

En caso de requerirse, se ha especificado en el sistema enviar una copia a otro destinatario.

Los elementos que se han utilizado son los mismos que se utilizaron anteriormente.

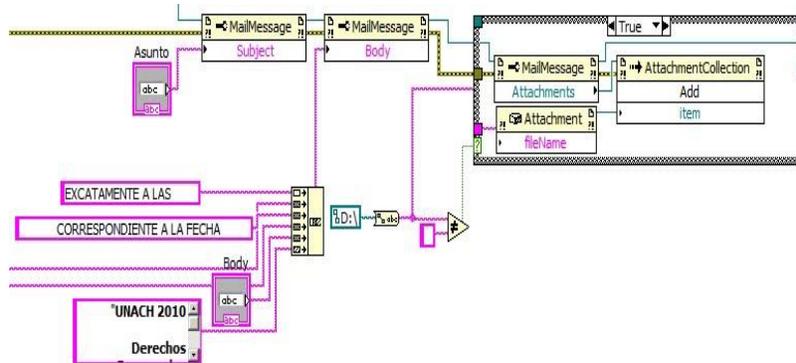
La representación en el diagrama de bloques de esta sección se aprecia a continuación:



## Envío de Copia a otro destinatario

### g) Elaboración del Body en el Correo Electrónico

Parte de la información a ser enviada al interior de la notificación será: fecha y hora del error, posible causa y además se adjuntará un documento con el reporte de los últimos procesos realizados.



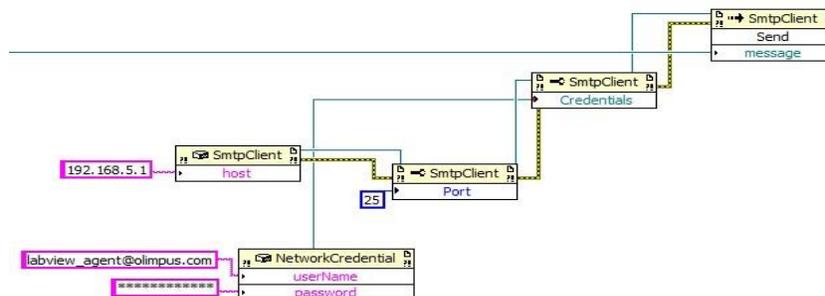
### Elaboración del Body en la Notificación

#### h) Autenticación Cuenta del Sistema

Para completar el envío de la notificación es necesario autenticar la cuenta del sistema en el servidor SMTP siendo indispensable ingresar información como:

- Dirección IP del Servidor SMTP.
- Puerto de transmisión.
- Id de la Cuenta y su Contraseña.

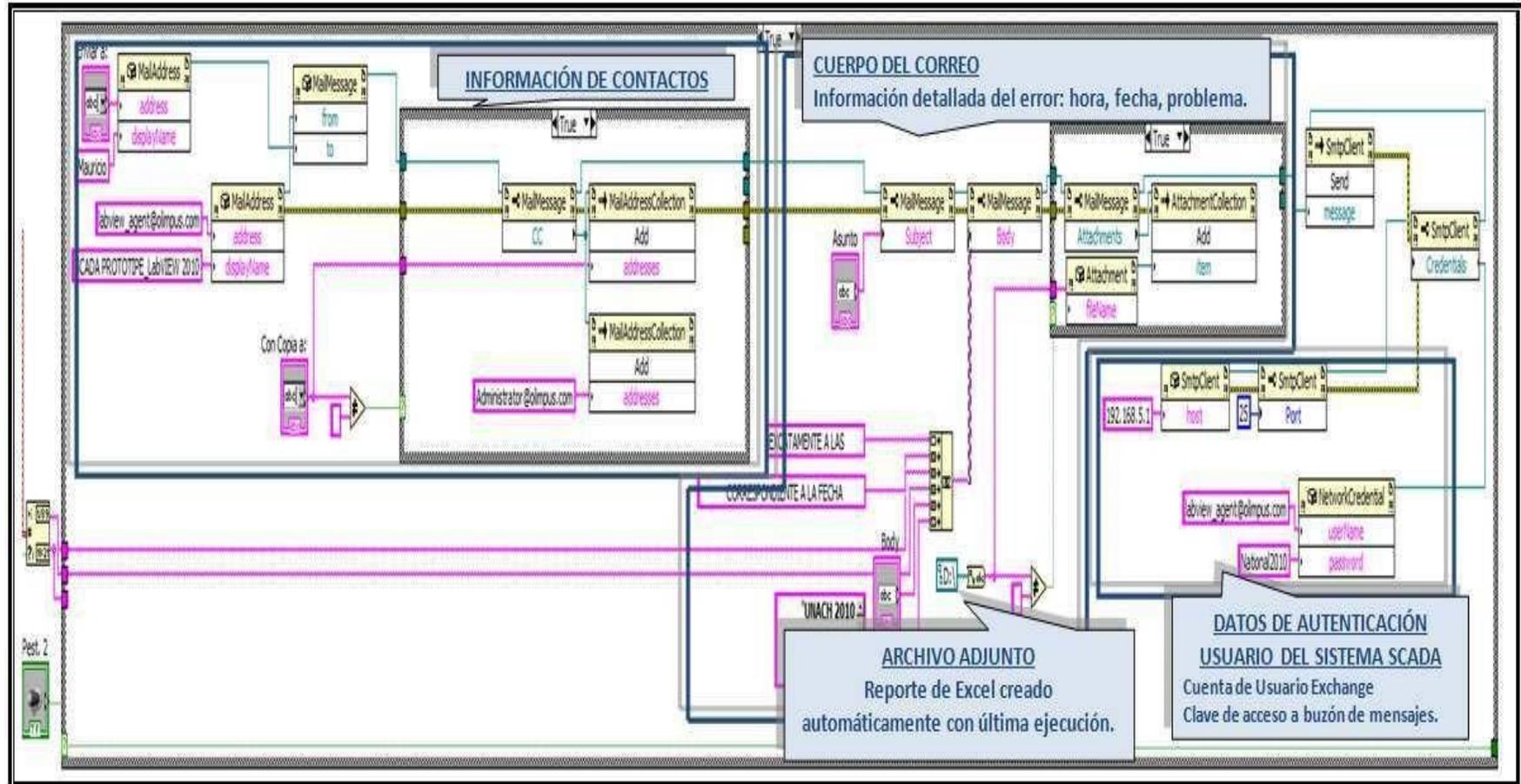
Con todos los valores anteriormente enunciados configurados ingresados de manera correcta el sistema procederá finalmente a enviar la notificación a las cuentas establecidas.



## Autenticación de la Cuenta del Sistema

## DIAGRAMA DE BLOQUES

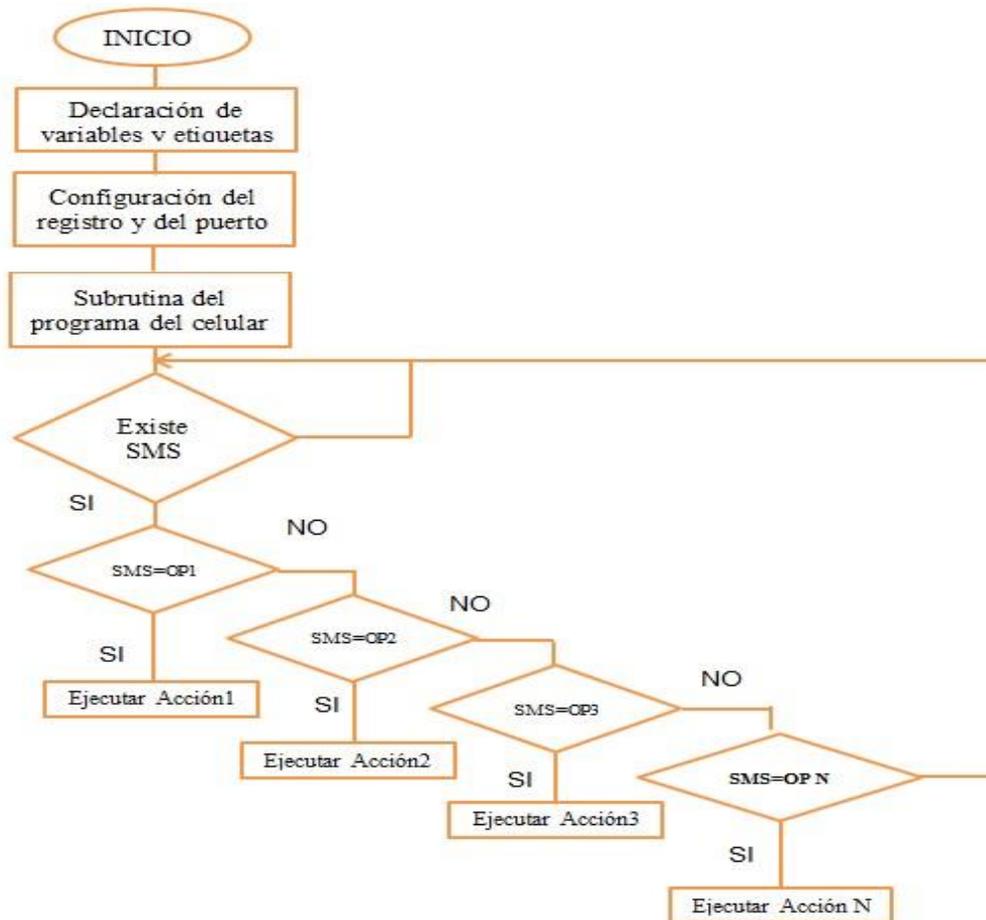
### PANEL ENVIÓ DE NOTIFICACIONES VIA MAIL



### 3.1.2.4. Controles 3.1.

#### Lógica de Funcionamiento

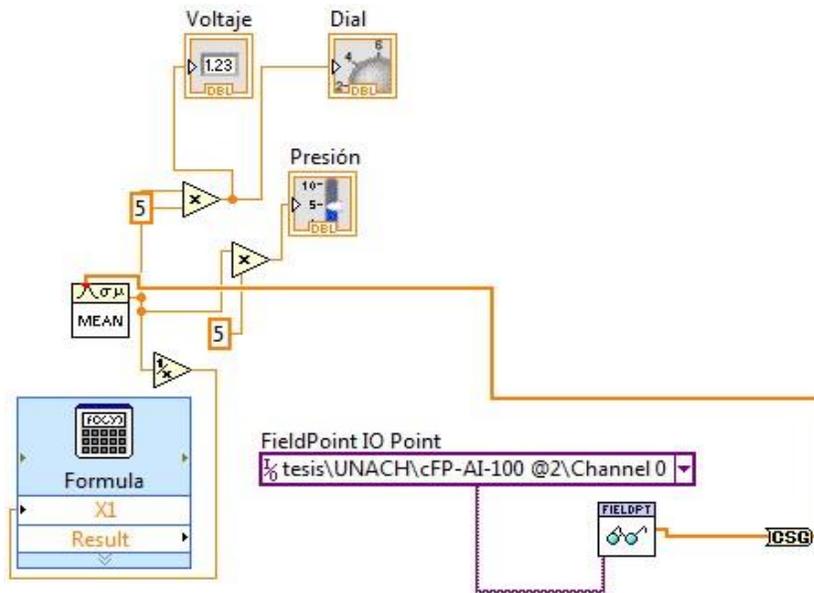
Para la realización de nuestro circuito externo de comunicación entre nuestro dispositivo celular y nuestro programa en LabVIEW, hemos tomado en cuenta la siguiente lógica de funcionamiento con la cual hemos elaborado el programa en MicroCode Studio.



Flujograma, Controles vía SMS

### 3.1.2.5. Sensor de Fuerza

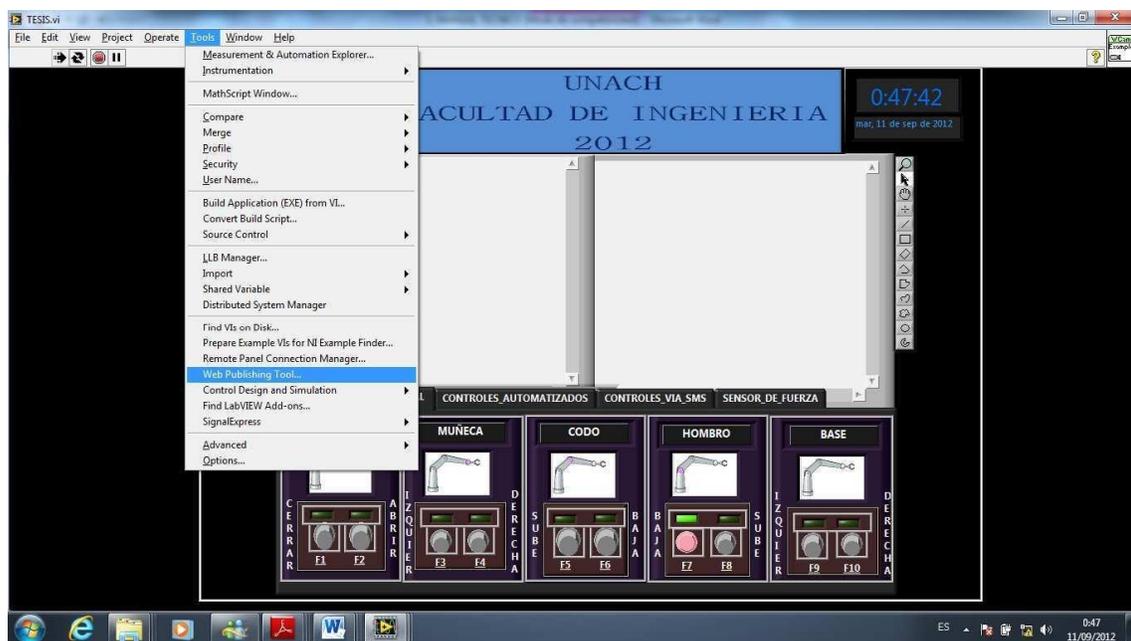
Para obtener los datos enviados desde el exterior ha nuestra PC se utilizó un sensor de fuerza resistivo FSR y con la ayuda de una tarjeta AI 100 del CFP los datos los podemos ver en nuestra interface.



*Diagrama de la visualización del control de fuerza*

### 3.1.2.6. Configuración de la Herramienta de Publicación WEB

**Web Server LabVIEW:** Es una herramienta que forma parte del entorno de programación a partir de la versión 8.6 que permite al desarrollador publicar vi's a través de internet, siendo de mucha utilidad en sistemas de monitoreo y control remoto.

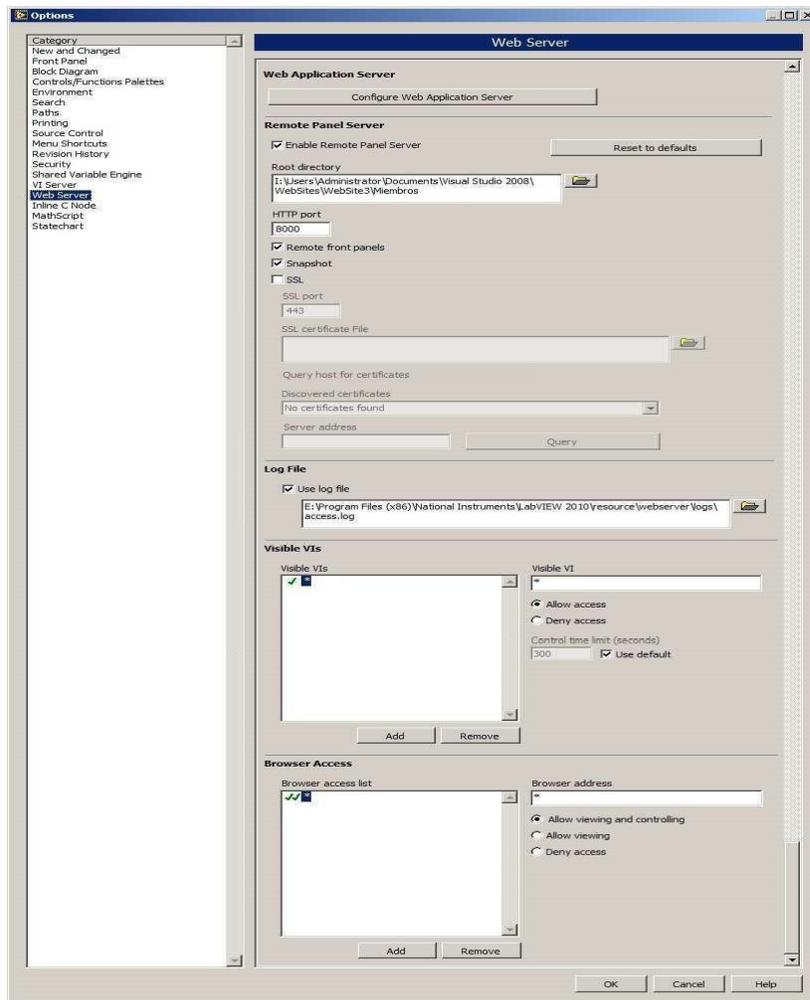


*Opciones de VI en LabVIEW2011*

Configuración del entorno Web Server.

CARACTERÍSTICA	VALOR
Activación del Servidor del Paneles Remotos:	Enable
Directorio Raíz:	
Puerto:	8000
Encriptación:	NO
Directorio archivo LOG:	
VI's Visibles:	* (all)
VI's con Acceso en Navegador Web:	* (all)

*Parámetros de configuración WebServer*



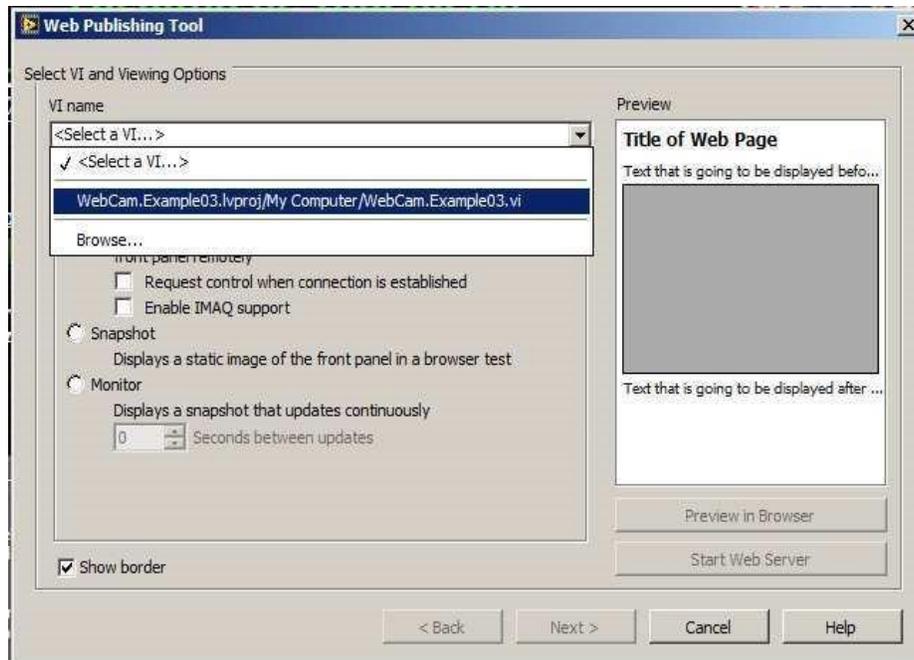
### Configuración de WebServer

Desde la barra de menú nos dirigimos hacia la sección de TOOLS posteriormente escogemos la opción Web Publishing Tool iniciamos el proceso de publicación web para nuestra aplicación.

Entre las configuraciones que debemos realizar consta:

Selección del VI que será publicado vía web.

Opciones de visualización.



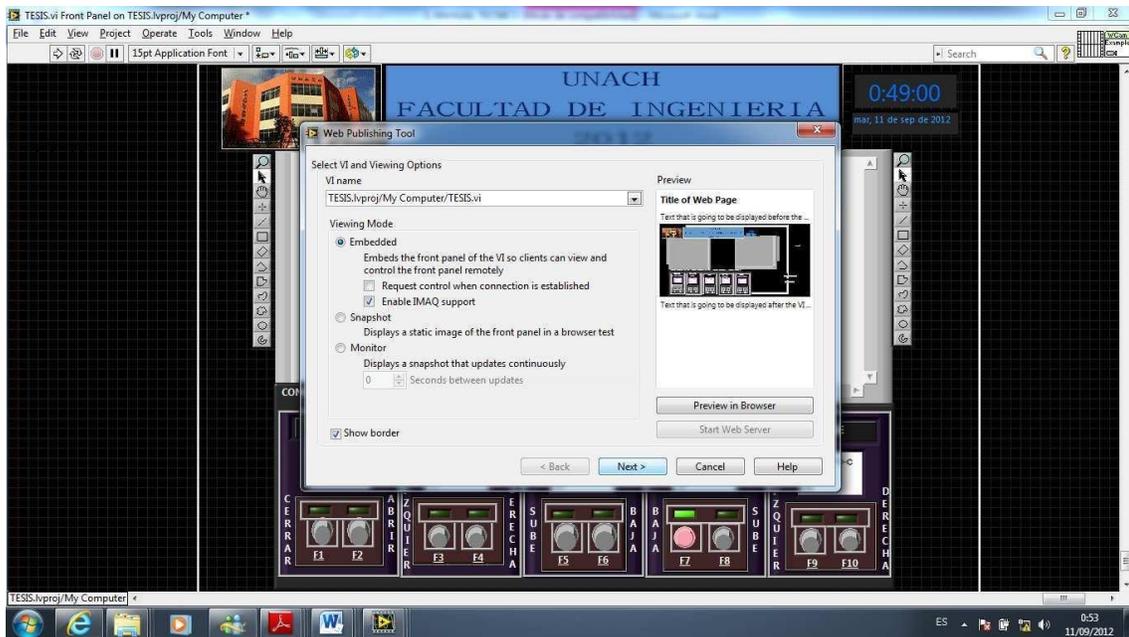
*Publicación Web, selección de VI*

### **Opciones de visualización:**

Embebido: Muestra la aplicación incrustada en una página web que permite al usuario operar con todas las herramientas disponibles en el panel frontal.

Snapshot: Muestra en una página web la imagen con el panel frontal de la aplicación.

Monitor: Muestra en una página web la imagen con el panel frontal de la aplicación capas de actualizarse cada un periodo de tiempo determinado, que puede ser utilizado para fines de supervisión.



### Modo Visualización de VI en Web Browser

#### **Selección de la salida en HTML.**

Es básicamente el formato y el contenido que aparecerá en el archivo html que será visualizado a través de Internet Explorer.

Opciones almacenamiento:

- Directorio de almacenamiento.

- Nombre de archivo.

- URL, de acceso en Internet Explorer.

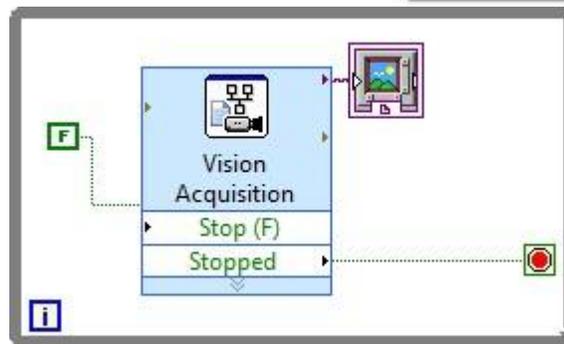
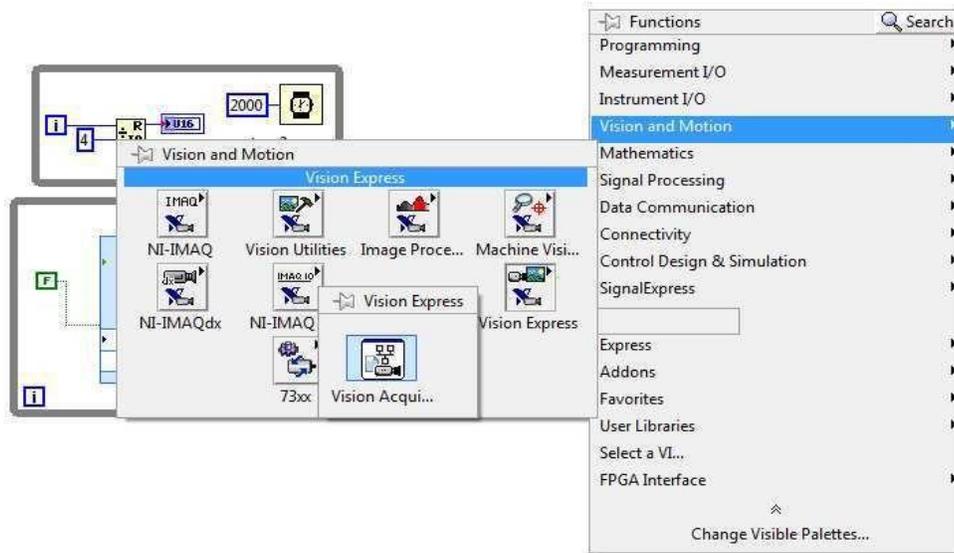
Visualización del panel frontal en Internet Explorer.



Visualización en Internet Explorer

### 3.1.2.7. Visión Artificial

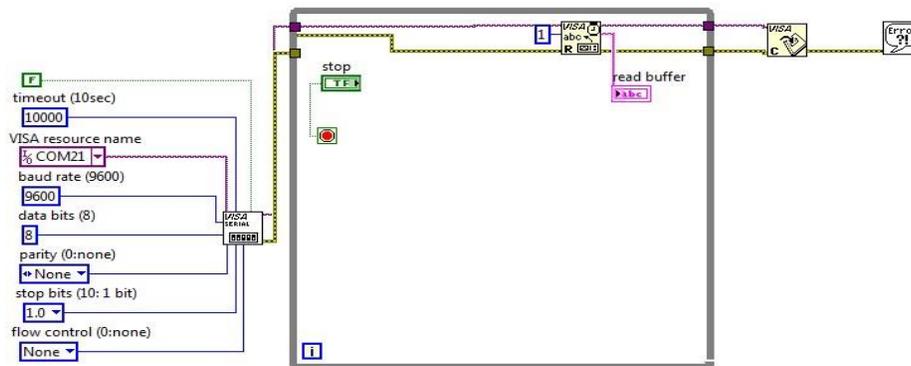
Para acceder a la visualización de nuestra visión artificial en tiempo real procedemos a conectar cámaras web a nuestra PC y con ayuda de la función



*Herramienta para obtener Visión Artificial*

### 3.1.2.8. Adquisición de datos VISA

Para leer los datos de nuestro programa utilizamos la función VISA del LabVIEW la misma que nos permite configurar el puerto serial para la intercomunicación con el celular.



*Diagrama de configuración VISA*

### 3.2. CONEXIÓN ELEMENTOS DE CAMPO

Parte de la infraestructura integrada que se compone por; una PC que actúa como estación maestra que mantiene el sistema y el servidor web que alojará la aplicación para su publicación y acceso a través de Internet Explorer, un instrumento de campo el cual provee de canales de entrada y salida de información, un router inalámbrico que conecta los dispositivos que contarán con acceso a la red local, y dos estaciones que serán contarán con acceso al modo de monitoreo y control remoto.

Las pruebas realizadas estrictamente corresponden a la conexión existente entre la estación maestra y el dispositivo CFP conectados a través de un switch.

Se presenta por tanto a continuación los elementos que intervienen en la conexión:

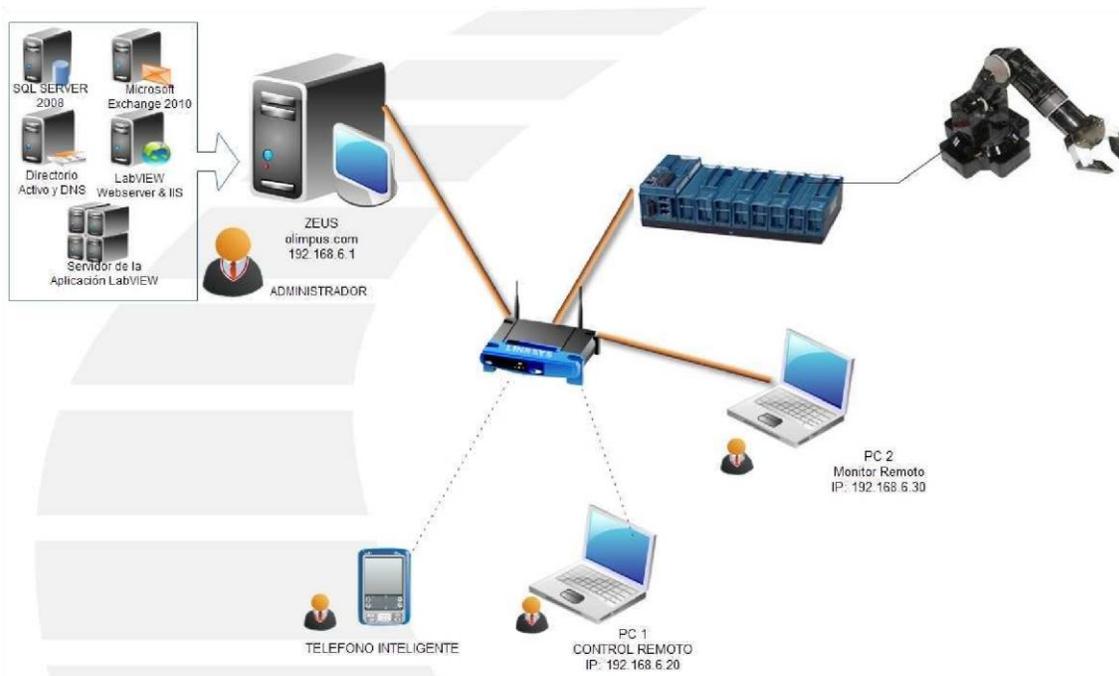
Cables de conexión directo UTP.

Switch

PC Estación Maestra

Cfp Modelo 2020

A continuación en la siguiente se muestra la topología de red utilizada para conectar todos los dispositivos.



#### 3.2.1. Configuración de Adaptadores de Red

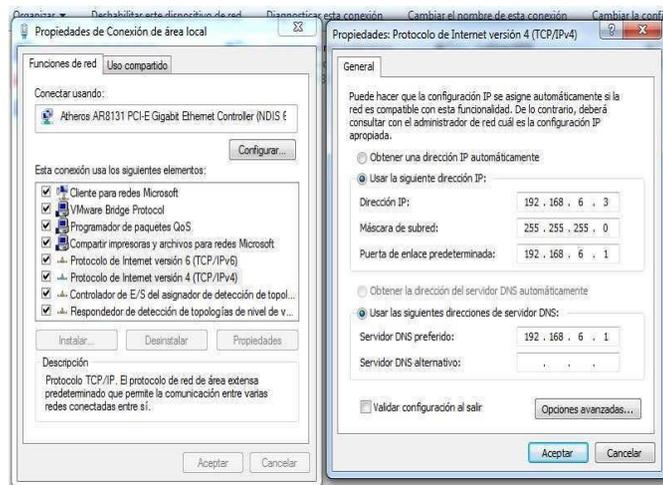
A continuación se muestra los valores de direccionamiento IP que deberán estar presentes en los dispositivos conectados:

<u>CARACTERÍSTICA</u>	<u>ESTACIÓN MAESTRA</u>	<u>CFP</u>
Dir IP V4	192.168.6.3	192.168.6.2
Máscara	255.255.255.0	255.255.255.0
Default Gateway	-	-
Primary DNS	192.168.6.1	192.168.6.1

Parámetros de configuración IP

Configuración del Direccionamiento IP en Estación Maestra:

- 3) Panel de Control -> Conexiones de Red -> Conexión de Área Local.



Configuración de Dir. IP en Servidor

IPCONFIG en Símbolo del sistema para comprobar el direccionamiento IP configurado:

```
C:\Users\Pablo>ipconfig

Configuración IP de Windows

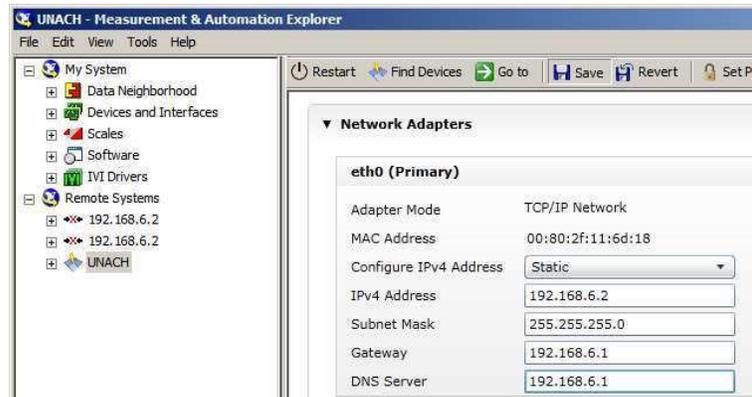
Adaptador de Ethernet Conexión de área local:

Sufijo DNS específico para la conexión. . . :
Vínculo: dirección IPv6 local. . . . . : fe80::8c4d:9c4d:2a96:9957%11
Dirección IPv4. . . . . : 192.168.6.3
Máscara de subred. . . . . : 255.255.255.0
Puerta de enlace predeterminada. . . . . : 192.168.6.1
```

Verificación de Direccionamiento

#### 4) Measurement Automation Explorer

Para configurar el direccionamiento IP en nuestro Compact FieldPoint es necesario ir hasta el elemento REMOTE SYSTEM en la parte derecha de la pantalla, seleccionamos el Compact FieldPoint encontrado y nos dirigimos hasta el menú de NETWORK SETTINGS para configurar el direccionamiento con los parámetros requeridos para el proyecto.



Configuración de <<NETWORK Adapter>> en CFP

El sistema a través de un mensaje nos indica que el sistema remoto debe ser reiniciado.



Mensaje de reinicio del CFP

Al momento de arrancar nuevamente el sistema ya podremos ver el nuevo direccionamiento IP configurado.



Disponible nueva configuración de CFP

### 3.2.1.1. Pruebas de conexión

Para realizar las pruebas de conectividad entre nuestros dos dispositivos centrales es necesario validar el direccionamiento que poseen, esto se lo realiza fácilmente en símbolo del sistema en la estación maestra a través del comando **ipconfig-all** que ingresado mostrará en la consola la siguiente información:

```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix . . . : fe80::7482:459a:827:9944x11
    Link-local IPv6 Address . . . . . : fe80::7482:459a:827:9944x11
    IPv4 Address. . . . . : 192.168.6.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 0.0.0.0
                                   192.168.6.1
  
```

IPCONFIG en Router

Para las pruebas de conectividad optamos por realizarlas con el comando <<ping IP DESTINO>> desde Símbolo del Sistema en nuestra PC.

Verificando que se realizó de manera exitosa.

```

C:\Users\Administrator>ping 192.168.6.2

Pinging 192.168.6.2 with 32 bytes of data:
Reply from 192.168.6.2: bytes=32 time=1ms TTL=60
Reply from 192.168.6.2: bytes=32 time=1ms TTL=60
Reply from 192.168.6.2: bytes=32 time<1ms TTL=60
Reply from 192.168.6.2: bytes=32 time<1ms TTL=60

Ping statistics for 192.168.6.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
  
```

Ping a CFP <<IP: 192.168.6.2>>

### 3.2.2. PRUEBAS DE CONEXIÓN EN DISPOSITIVO CFP Y MÓDULOS.

Para acceder a la configuración del Módulo CFP es necesario utilizar la herramienta Measurement Automation Explorer.

**MeasureAutomation Explorer:** National Instruments proporciona Measurement & Automation Explorer (MAX), que es una interfaz gráfica que permite al usuario configurar los dispositivos físicos conectados de manera que se puedan integrar de manera fácil a los VI's desarrollados. MAX se instala normalmente con el entorno de desarrollo de aplicaciones LabVIEW.

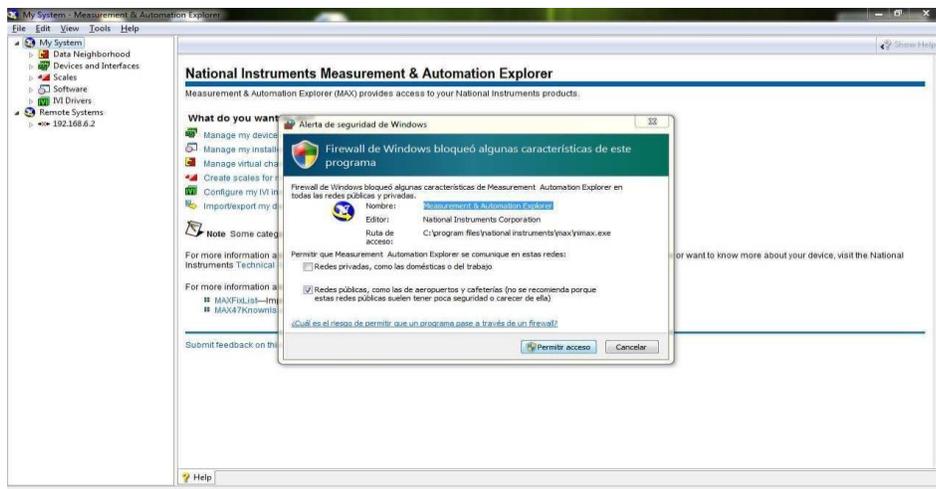
#### 3.2.2.1. Ingreso a MAX y Acceso al Módulo CFP

Como se mencionó anteriormente MAX viene incorporado en el conjunto de herramientas que se instala con LabVIEW 2010 su icono de representativo que se localiza en el escritorio es el siguiente:



Icono de MAX

Al ingresar a la aplicación dependiendo de nuestras configuraciones establecidas para el FIREWALL será necesario otorgar permisos de acceso.



Interfaz Principal de MAX

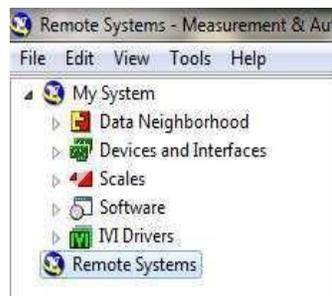
Por tanto es necesario permitir todo el tráfico e intercambio de datos que surja con esta aplicación (MAX).



### Permitir Acceso en FIREWALL

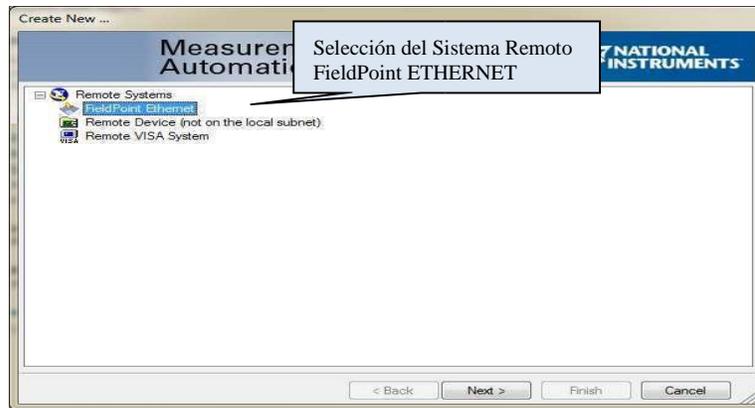
Para iniciar con la configuración de nuestro dispositivo es necesario que nuestra aplicación establezca comunicación con el CFP conectado a la infraestructura de red y este previamente configurado con el direccionamiento IP correcto.

Para reconocer el dispositivo deberemos escoger la opción REMOTE SYSTEMS que identifica los dispositivos asociados con anterioridad compatibles con National Instruments y que están integrados en nuestra infraestructura de red.



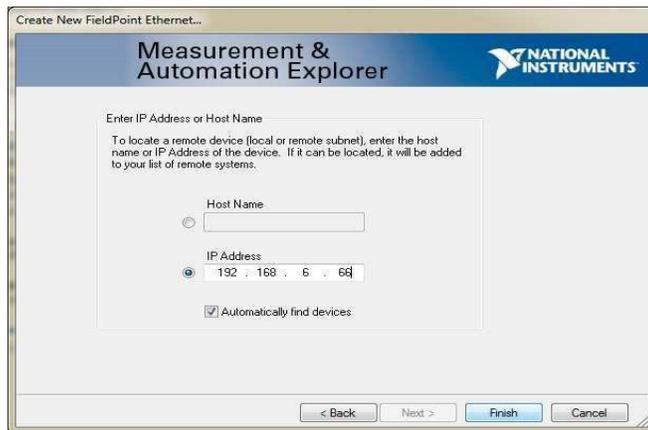
### Sistemas Remotos

Para agregar un nuevo dispositivo deberemos dar un click derecho y seleccionar la opción <<CREATE NEW>> aparece entonces una nueva ventana con los posibles instrumentos remotos que podrán ser conectados como se muestra a continuación:



### Reconocimiento de un Nuevo Dispositivo

Ubicamos en equipo del listado disponible y posteriormente deberemos especificar la dirección IP configurada.



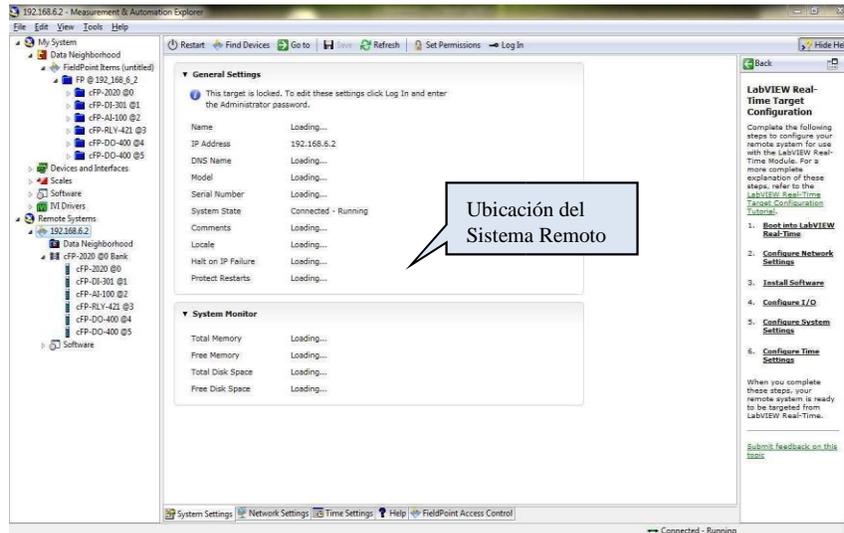
### Direccionamiento IP del Dispositivo

De ser exitosa la conexión el sistema mostrará la siguiente información indicando que se ha asociado de manera correcta el dispositivo.



### Establecimiento de Conexión al Dispositivo

Podremos ver entonces que la interfaz principal de MAX ha cambiado totalmente, identificamos que en la parte derecha en <<REMOTE SYSTEMS>> encontramos asociado el Compact FieldPoint con la dirección <<192.168.6.2>>y todos los módulos que se hallan acoplados en el hardware.



### Información General del cFP



### Módulos Asociados en cFP

#### 3.2.2.2. Pruebas en Módulos conectados a CFP

Para verificar que los módulos están conectados de manera correcta y verificar que cada uno de los canales esté en perfecto estado se enviará un pulso y se verificará que el LED indicador de cada canal se encienda correctamente en los módulos. Es necesario especificar que dos canales activaran un motor, tanto en dos sentidos respectivamente.

Para las pruebas a realizar es necesario ubicarnos en:

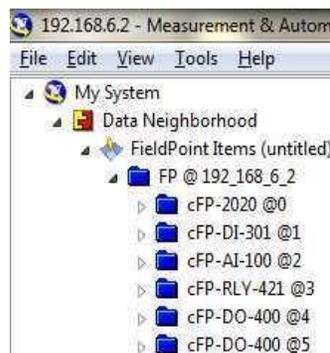
- 6) MySystem
- 7) Data Neighborhood

- 8) FieldPointItems
- 9) FP @192\_168\_6\_2

Por cuestiones de configuraciones y trabajo específicamente afines a nuestro proyecto se utilizarán las posiciones 4 y 5 de que contienen los módulos DO 400 que activarán los motores para proveer de movimiento a nuestro brazo robótico.

- 10) CFP-DO-400 @4 y CFP-DO-400 @5

Con un total de 5 motores utilizaremos 10 canales, 5 de ellos ubicados en el primer módulo DO-400 y los dos restantes del segundo módulo.



4.21: cFP asociado y módulos utilizables

### 3.2.3. ACCESO EN ESTACIONES REMOTAS

#### Instalación de LVRTE

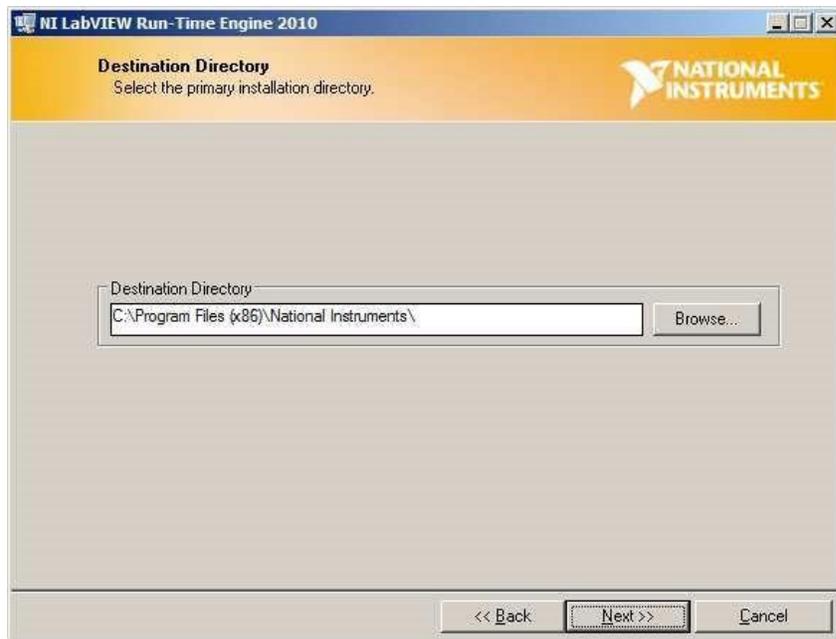


El componente se lo puede descargar desde la página oficial de National Instruments y no requiere de activación.

Para ejecutar la instalación es necesario descomprimir el instalador del archivo descargado, una vez descomprimido la instalación continua automáticamente.



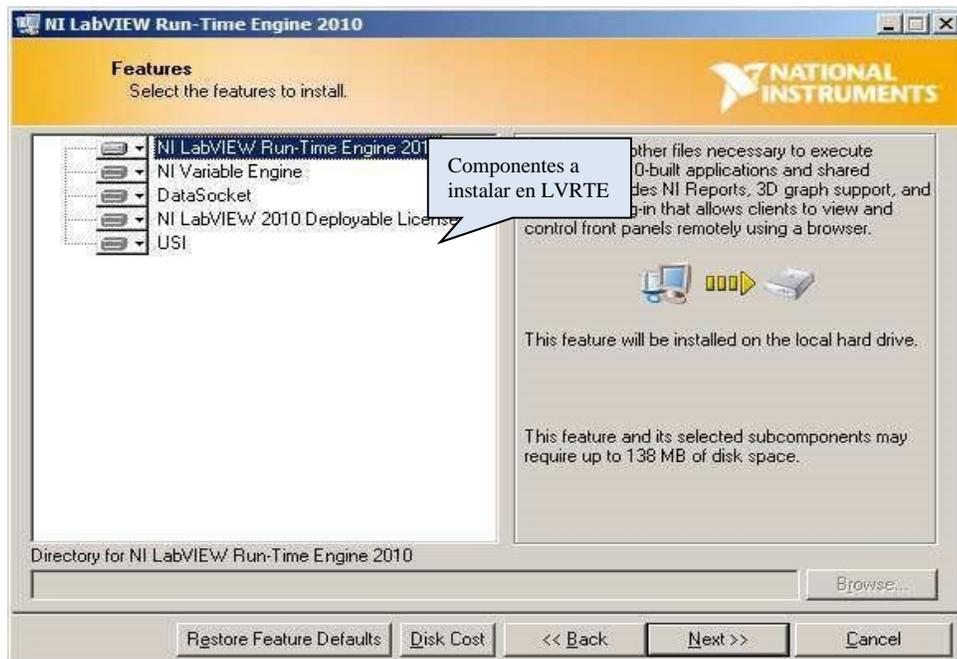
El sistema especifica el directorio de instalación para los componentes.



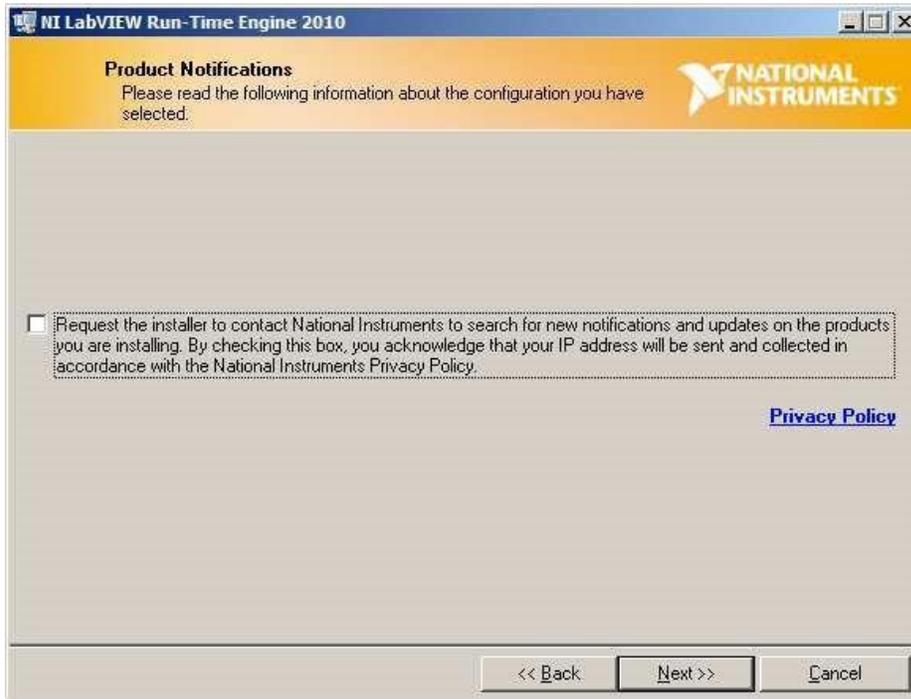
En caso de requerir una instalación personalizada, el sistema permite seleccionar los componentes que serán instalados con el sistema.

Dependiendo de las necesidades escogemos la opción indicada, para nuestro caso es únicamente necesario NI LabVIEW Run-Time Engine 2010.

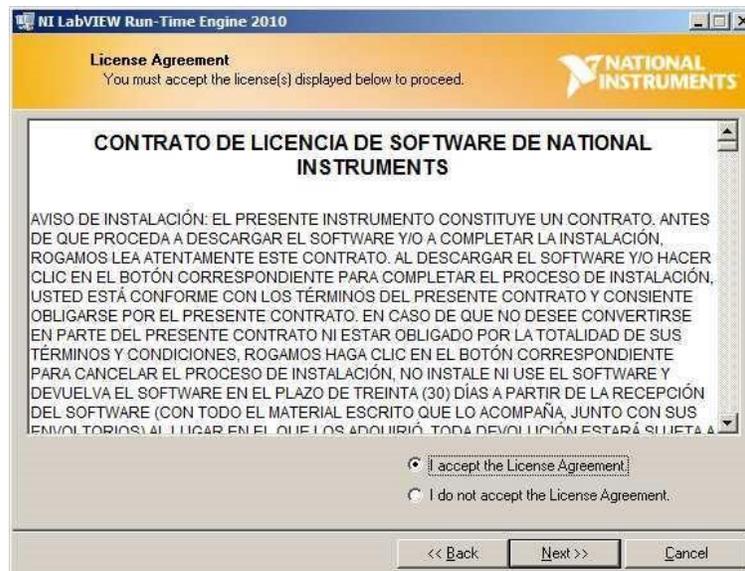
Es necesario mencionar que por cada versión de LabVIEW existe un paquete instalador de RunTime Engine, y que no son compatibles versiones distintas.



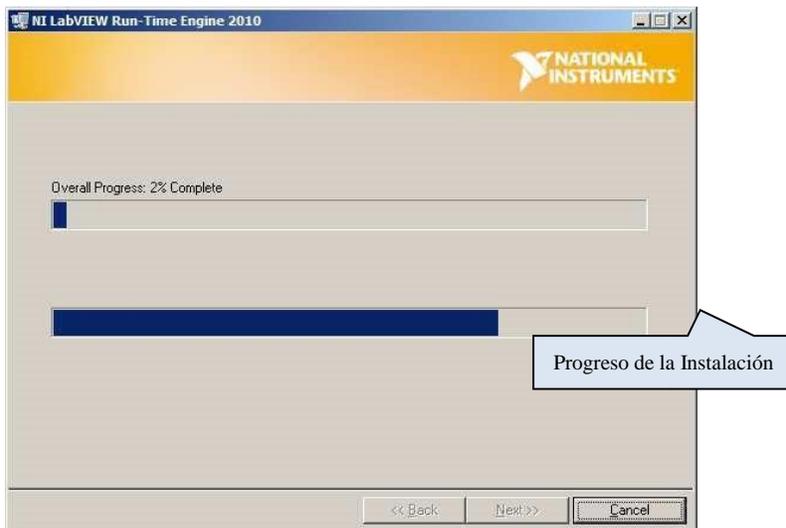
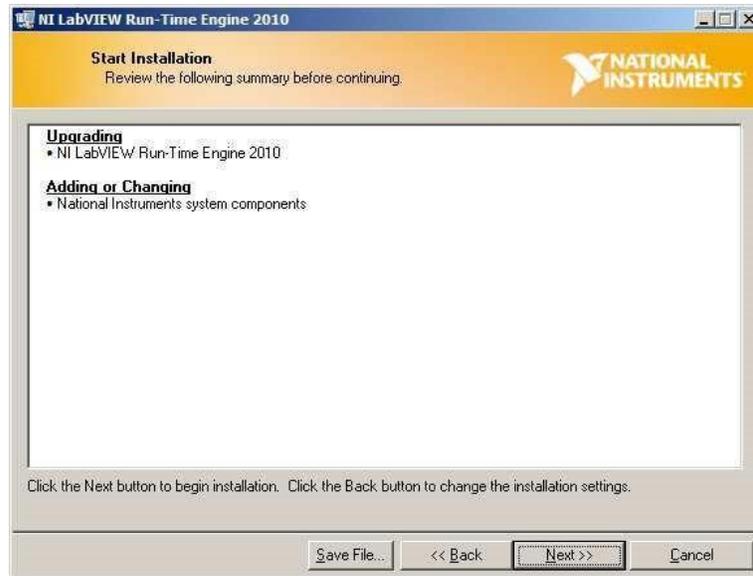
En caso de ser necesario especificamos si deseamos que National Instruments realice actualizaciones automáticas al producto instalado.



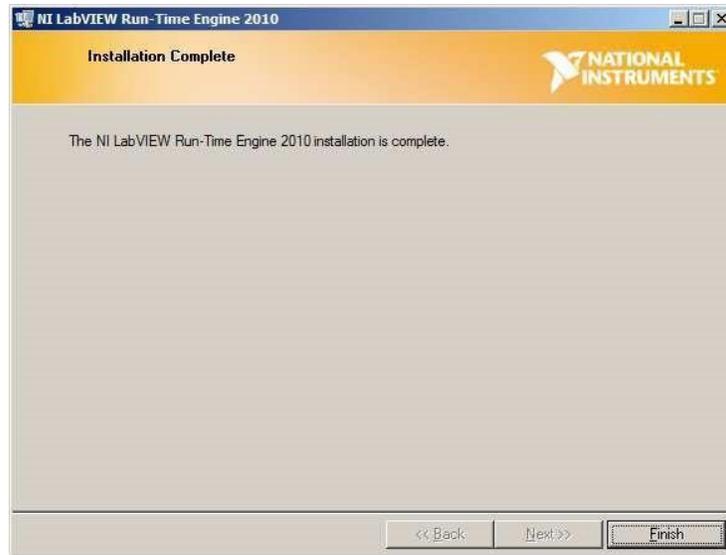
Aceptamos el contrato de licencias de software de National Instruments.



Iniciamos con la instalación de los componentes seleccionados



El sistema instalación muestra que se ha concluido con el proceso.



Para completar la instalación es necesario reiniciar el sistema.



### 3.2.4. ACCESO DESDE TERMINALES REMOTAS

#### Acceso de Usuario de Monitoreo Remoto

Para acceder al modo de modo de monitoreo y control remoto el usuario únicamente debe digitar la siguiente información en la barra de dirección en el Internet Explorer.

<http://192.168.6.3:8000/tesis.html>

Este será la URL que inmediatamente al localizar el servicio mostrará una pantalla en la cual el usuario podrá ver cual es el estado del sistema y además podrá informarse si al momento está funcionando o no de manera correcta, es necesario especificar que este modo de acceso no permite interactuar con la aplicación:



Monitor

### Remoto en IE



### Aplicación controlada desde LabVIEW

#### **Acceso de Usuario Control Remoto**

Para hacer uso de la aplicación web que está alojada en el servidor web de LabVIEW el usuario debe conocer cuál es la dirección URL para acceder al recurso, a continuación se especifica cuál es el formato de la dirección que será introducida en la barra de direcciones de Internet Explorer.

<http://192.168.3:8000/tesis.html>

Inmediatamente se cargará una página web que nos muestra la aplicación que tenemos remotamente controlando a todo el proyecto.

**ANEXO 2**  
**MANUAL DE USUARIO**

**UNIVERSIDAD NACIONAL DE CHIMBORAZO**



**FACULTAD DE INGENIERIA**

Escuela de Ingeniería en Electrónica y Telecomunicaciones

CONTROL DE UN BRAZO ROBÓTICO A DISTANCIA DESARROLLADO EN AMBIENTE LABVIEW, SOBRE UNA RED ETHERNET Y ATRAVES DE MENSAJES DE TEXTO VIA TELEFONO CELULAR.

Riobamba, 10 de septiembre del 2012

Desarrollado por:  
Pablo Tenelanda Sanaguano

**Contenido**

1.	INTRODUCCIÓN .....	183
2.	OBJETIVOS .....	184

2.1. OBJETIVO GENERAL	184
2.2. OBJETIVO ESPECIFICO	184
3. CONTENIDO TECNICO	185
3.1. INTERFAZ DE LA APLICACIÓN	185
3.2. CONTROLES EN TIEMPO REAL	187
3.3. CONTROLES AUTOMATICOS	188
3.4. ENVIO DE NOTIFICACIONES VIA CORREO ELECTRONICO	189
3.5. CONTROL VIA SMS	189
3.6. SENSOR DE FUERZA	190
3.7. Visión Artificial	190
3.8. ACCESO VIA WEB	191
3.8.1. ACCEDIENDO AL MODO DE MONITOREO REMOTO	191

## 1. INTRODUCCIÓN

El sistema que se describe a continuación tiene por objetivo fundamental monitorear y controlar remotamente los movimientos de un brazo robótico, a través de su interfaz el usuario podrá realizar movimientos en tiempo real, movimientos automáticos, enviar notificaciones el momento que inicia y termina la rutina automática, control del brazo mediante SMS enviados vía teléfono celular, medir la fuerza que se ejerce en la pinza mediante un sensor de fuerza ubicado en la pinza del brazo.

Adicional a ello el usuario podrá acceder a la aplicación a través de Internet Explorer propiciando un acceso remoto a la aplicación.

El sistema está desarrollado totalmente en LabVIEW 2011.

Para la programación del PIC hemos utilizado Microcode Studio.

Para el monitoreo del proyecto en la pantalla del celular utilizaremos el Vectir.

## **2. OBJETIVOS**

### **2.1. OBJETIVO GENERAL**

Monitorear y controlar remotamente los movimientos de un brazo robótico

### **2.2. OBJETIVO ESPECIFICO**

Permitir al usuario activar los distintos servomotores que forman parte de la estructura electrónica del brazo robótico.

Iniciar una secuencia automática de movimientos del brazo robótico y verificar que las notificaciones vía e – mail lleguen a los correos electrónicos ingresados en la interface.

Permitir el acceso remoto de usuarios vía web a través de Internet Explorer.

Recibir instrucciones a través de SMS enviados mediante teléfono celular.

Obtener monitoreo en tiempo real de los movimientos del brazo robótico en la pantalla de nuestro celular.

## **3. CONTENIDO TECNICO**

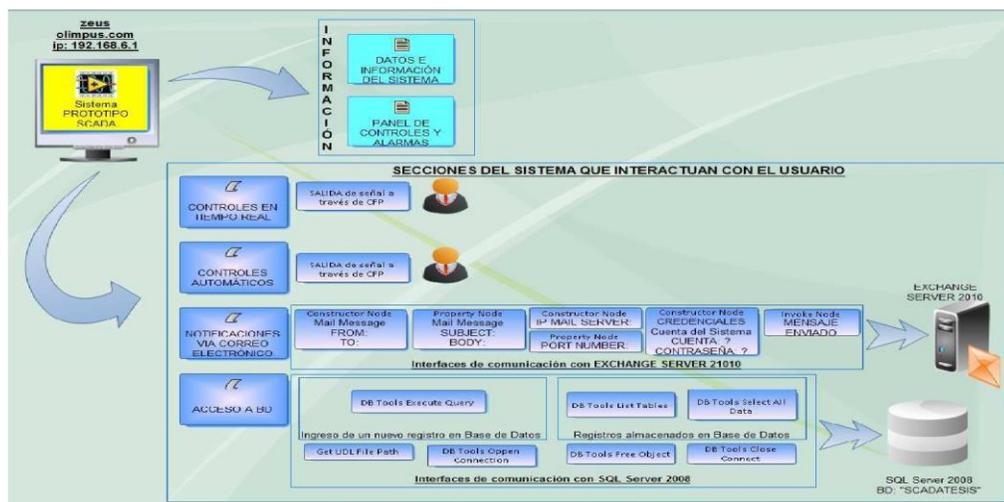
### 3.1. INTERFAZ DE LA APLICACIÓN



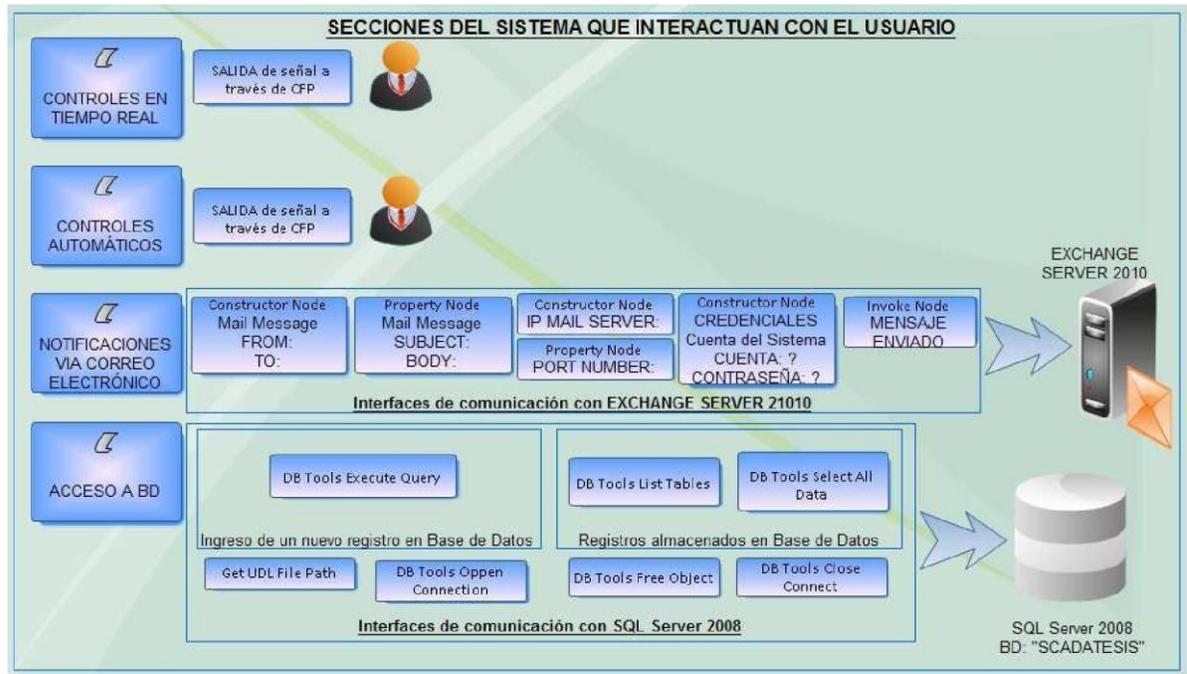
*Panel Frontal del Sistema*

El sistema presenta una sola interfaz a través de la cual el sistema puede interactuar con el usuario, esta interfaz es conocida en LabVIEW como el Panel Frontal y pone a disposición del usuario todos los controles configurados e implementados, además de ello se puede distinguir indicadores como leds o también paneles exclusivamente utilizados para mostrar información como la hora, la fecha y los tiempos de ejecución programados.

En la siguiente representación se puede apreciar cuales son las secciones que interactúan de manera directa con el usuario.



### Información gestionada por la aplicación



### Secciones que interactúan con el Usuario

Desde la interfaz del sistema el usuario principalmente distinguirá cuatro secciones bien definidas por sus funciones, siendo estas:

- Controles en Tiempo Real.
- Controles Automáticos
- Controles vía SMS
- Sensor de Fuerza

Todas estas secciones se encuentran al interior de un contenedor (TAB CONTROL) que activa una opción por ves, de manera que el usuario no podrá trabajar de manera simultánea con varias secciones.

### **3.2. CONTROLES EN TIEMPO REAL**

Esta sección del programa presenta en su interfaz un conjunto de botones que actúan como controles para activar los motores que están conectados a las salidas DO 400, estos botones pueden también ser activados por las teclas F1 hasta F10, la interfaz de esta sección es totalmente intuitiva y fácil de operar, en ella se encuentran etiquetas

para indicar a cuál de los sentidos corresponde cada botón y la operación que estará ejecutando el brazo robótico.



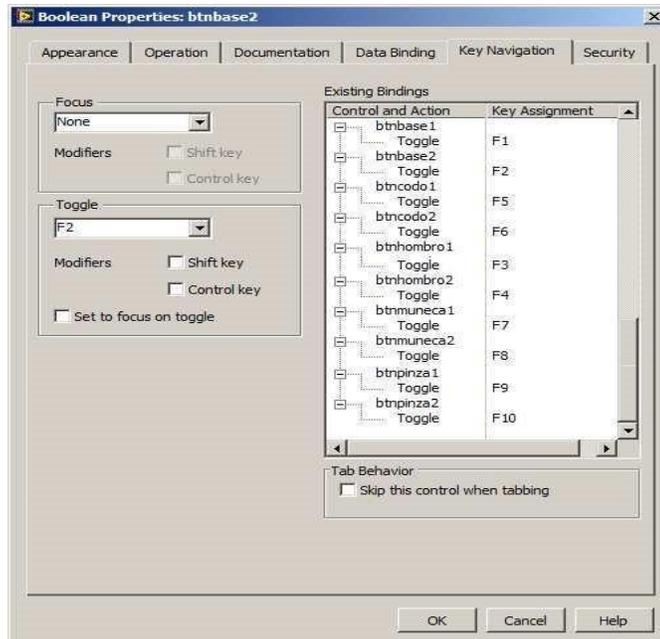
*Panel de Controles en Tiempo Real*



*Control de un motor "PINZA"*

Etiquetas: PINZA (Abrir-F1, Cerrar-F2),

Para realizar un movimiento el usuario deberá presionar un solo botón, es necesario entender que no se pueden presionar dos botones ya que el motor asociado no ejecutaría ningún movimiento, el indicador LED que se encuentra en la parte superior a cada botón indica si el motor ha sido correctamente activado y que la salida de la señal digital se ha realizado de manera correcta. Este concepto es aplicable a los 5 motores que controlan las distintas partes del brazo robótico



Configuración de Key Navigation

BOTON	VALOR	ESTADO
ABRIR	False	APAGADO
CERRAR	False	APAGADO

BOTON	VALOR	ESTADO
ABRIR	True	ENCENDIDO
CERRAR	False	APAGADO

BOTON	VALOR	ESTADO
ABRIR	False	APAGADO
CERRAR	True	ENCENDIDO

Estados de motores en ejecución

### 3.3. CONTROLES AUTOMATICOS

Para activar esta sección de la aplicación es necesario primero oprimir el BOTON DE INICIO y que se ubica en la parte superior izquierda del TAB CONTROL, entonces la rutina automática empezara de inmediato.



Panel de Control Movimientos Automáticos

Para activar una secuencia automática el usuario debe colocar el botón de INICIO, entonces el brazo robótico comenzará a moverse de acuerdo a la programación configurada.

### 3.4. ENVIO DE NOTIFICACIONES VIA CORREO ELECTRONICO

En esta parte del sistema el usuario puede configurar el envío de las notificaciones vía correo electrónico.

El usuario puede configurar:

- Cuentas destinatarias
- Asunto del mensaje
- Cuerpo del mensaje.

### 3.5. CONTROL VIA SMS

Nuestro proyecto además puede ser manipulado con mensajes de texto enviado vía teléfono celular, las instrucciones llegan a un modem con el cual vamos a transferir los datos a nuestra PC para ejecutar las acciones expresadas en el SMS.



*Panel Frontal Control Vía SMS*

### 3.6. SENSOR DE FUERZA

En la siguiente pestaña podemos apreciar la fuerza que esta ingresando a través de nuestro sensor localizado en la pinza de nuestro brazo, esto nos permitirá hacer una relación de fuerza para que la pinza coja algún objeto sin lastimarlo.



*Panel Frontal Sensor de Fuerza*

### **3.7. Visión Artificial**

Nuestro proyecto nos permite ver en tiempo real los movimientos de un brazo robótico lo cual hace de el un instrumento muy importante para la implementación de proyectos de este tipo.



*Panel Frontal Visión Artificial*

### **3.8. ACCESO VIA WEB**

#### **3.8.1. ACCEDIENDO AL MODO DE MONITOREO REMOTO**

Para acceder al modo de modo de monitoreo y control remoto el usuario únicamente debe digitar la siguiente información en la barra de dirección en el Internet Explorer.

<http://192.168.6.3:8000/tesis.html>

Este será la URL que inmediatamente al localizar el servicio mostrará una pantalla en la cual el usuario podrá ver cuál es el estado del sistema y además podrá informarse si al momento está funcionando o no de manera correcta, es necesario especificar que este modo de acceso no permite interactuar con la aplicación:



Monitor Remoto en IE

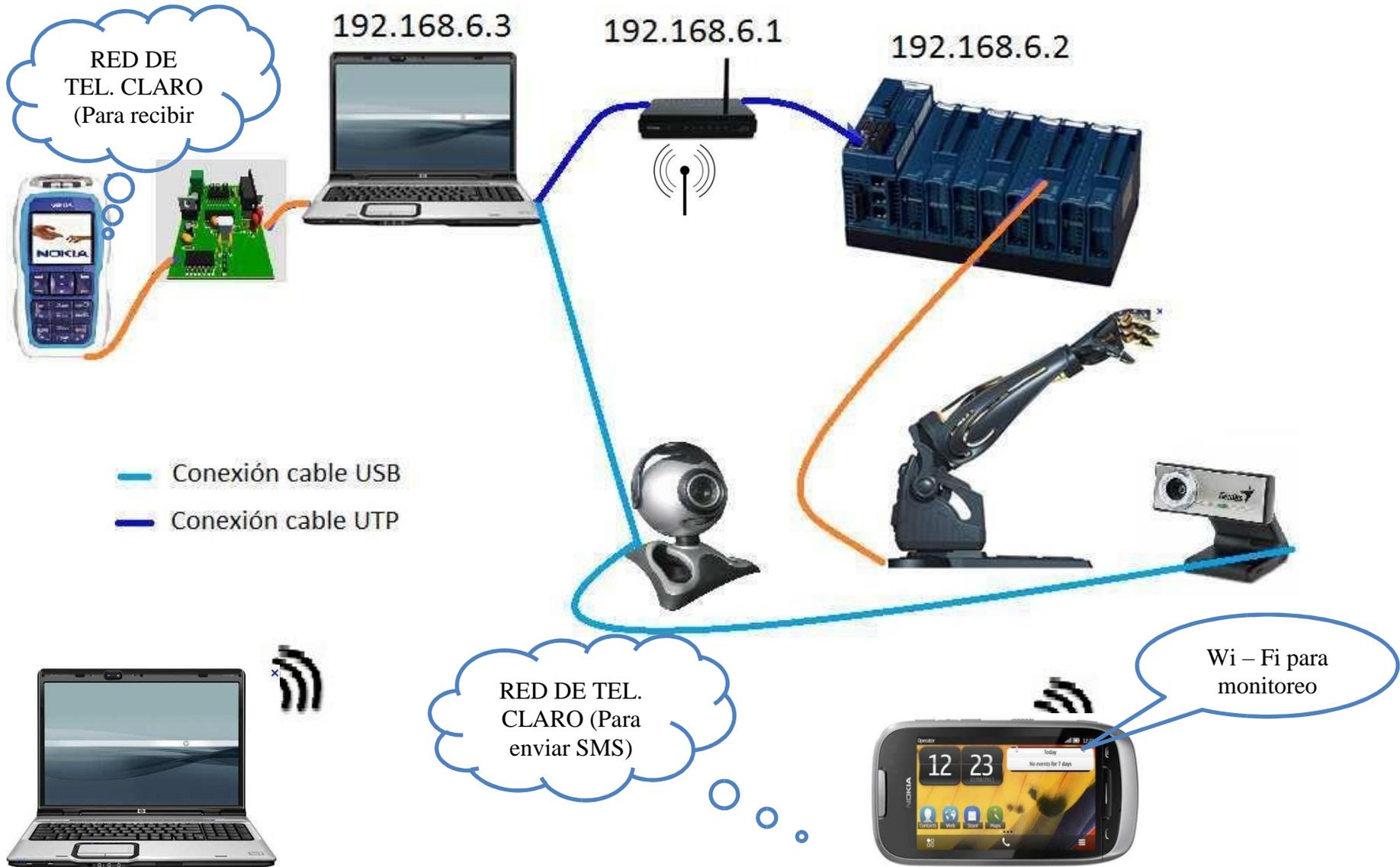


Aplicación controlada desde LabVIEW

La aplicación a instalar se la denomina Run Time Engine y se la puede descargar gratuitamente de la página oficial de National Instruments.

Una vez instalado y listo el equipo es necesario ingresar la siguiente URL para poder tener acceso a la aplicación.

# DIAGRAMA DE CONEXIÓN DE LOS EQUIPOS ESTACIÓN MAESTRA



# DIAGRAMA DE CONEXIÓN DE LOS EQUIPOS ESTACIÓN MAESTRA

