



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TEMA:

**Diseño e implementación de un algoritmo de control para sistemas
SCADA mediante una tarjeta de adquisición de datos aplicando
Datsocket con fines didácticos**

AUTOR:

Rodríguez Ortiz, Cristian Eduardo

Trabajo de Titulación previo a la obtención del título de
INGENIERO EN TELECOMUNICACIONES

TUTOR:

M. Sc. Zamora Cedeño, Néstor Armando

Guayaquil, Ecuador

3 de Marzo del 2020



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

CERTIFICACIÓN

Certificamos que el presente trabajo fue realizado en su totalidad por el Sr.
Rodríguez Ortiz, Cristian Eduardo como requerimiento para la obtención
del título de **INGENIERO EN TELECOMUNICACIONES**.

TUTOR

M. Sc. Zamora Cedeño, Néstor Armando

DIRECTOR DE CARRERA

M. Sc. Heras Sánchez, Miguel Armando

Guayaquil, 3 de marzo del 2020



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

DECLARACION DE RESPONSABILIDAD

Yo, **Rodríguez Ortíz, Cristian Eduardo**

DECLARO QUE:

El trabajo de titulación: **“Diseño e implementación de un algoritmo de control para sistemas SCADA mediante una tarjeta de adquisición de datos aplicando Datasocket con fines didácticos”**, previo a la obtención del Título de **Ingeniero en Telecomunicaciones**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, 3 de marzo del 2020

EL AUTOR

RODRÍGUEZ ORTÍZ, CRISTIAN EDUARDO



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

AUTORIZACIÓN

Yo, **Rodríguez Ortiz, Cristian Eduardo**

Autorizo a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: **“Diseño e implementación de un algoritmo de control para sistemas SCADA mediante una tarjeta de adquisición de datos aplicando Datasocket con fines didácticos”**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

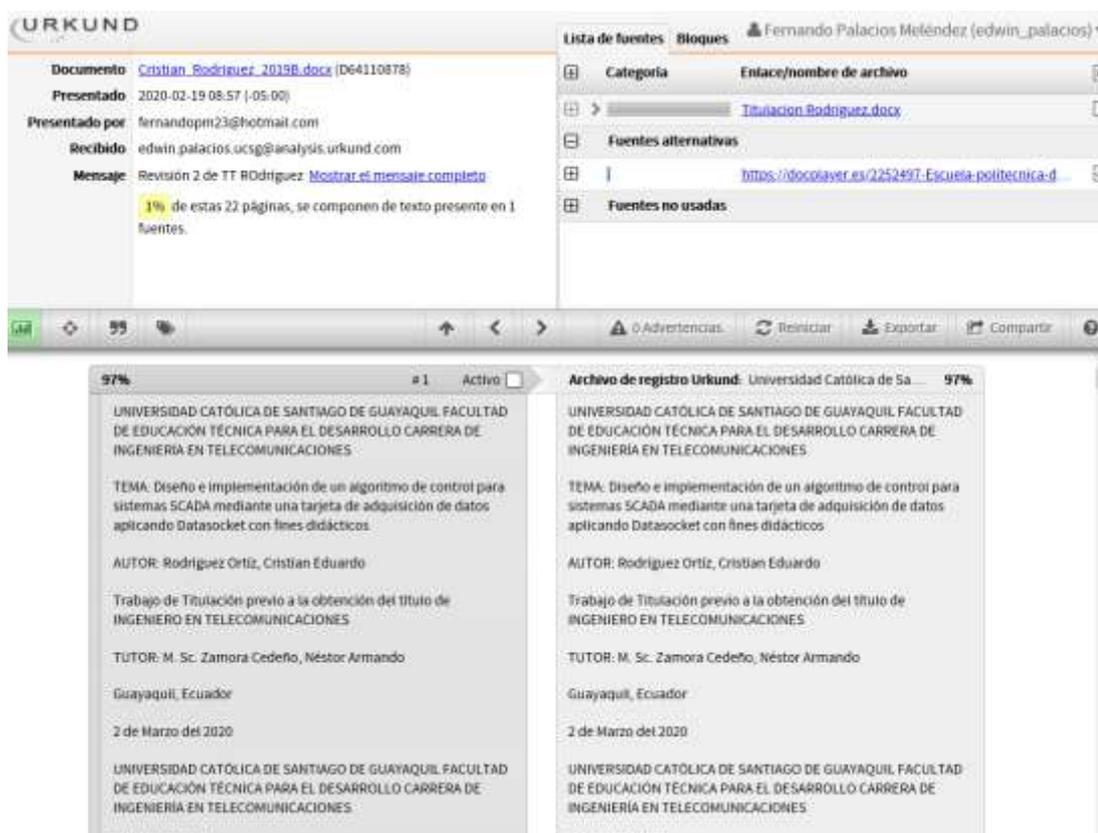
Guayaquil, 3 de marzo del 2020

EL AUTOR

RODRÍGUEZ ORTÍZ, CRISTIAN EDUARDO

REPORTE DE URKUND

Informe del Trabajo de Titulación de la Carrera de Ingeniería en TELECOMUNICACIONES, con 1% de coincidencias perteneciente al estudiante, CRISTIAN EDUARDO RODRÍGUEZ ORTÍZ.



The screenshot displays the URKUND interface. On the left, a document summary is shown for 'Cristian_Rodriguez_2019B.docx' (ID: D64110378), presented on 2020-02-19 08:57 (-05:00) by fernandopm23@hotmail.com to edwin.palacios.ucsg@analysis.orkund.com. The message indicates a revision of a thesis and that 1% of the 22 pages consist of text from 1 source.

On the right, the 'Lista de fuentes' (List of sources) is visible, showing a table with columns for 'Categoría' and 'Enlace/nombre de archivo'. One source is listed: 'Titulacion.Rodriguez.docx'. Below this, there are sections for 'Fuentes alternativas' (with a URL: https://docplayer.es/2252497-Escuela-politecnica-d) and 'Fuentes no usadas'.

At the bottom, two side-by-side document preview windows are shown, both displaying 97% similarity. The text in these windows is identical and reads: 'UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO CARRERA DE INGENIERÍA EN TELECOMUNICACIONES', 'TEMA: Diseño e implementación de un algoritmo de control para sistemas SCADA mediante una tarjeta de adquisición de datos aplicando Datasocket con fines didácticos', 'AUTOR: Rodriguez Ortiz, Cristian Eduardo', 'Trabajo de Titulación previo a la obtención del título de INGENIERO EN TELECOMUNICACIONES', 'TUTOR: M. Sc. Zamora Cedeño, Néstor Armando', 'Guayaquil, Ecuador', '2 de Marzo del 2020', and 'UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO CARRERA DE INGENIERÍA EN TELECOMUNICACIONES'.

Atte.

M. Sc. Néstor Armando Zamora Cedeño

TUTOR TRABAJO DE TITULACIÓN

DOCENTE TITULAR AUXILIAR – TIEMPO COMPLETO

DEDICATORIA

Quiero dedicar este trabajo a mis padres ya que fueron las personas quienes me brindaron su confianza y su apoyo para poder llegar a este punto crucial de mi carrera, a Dios ya que es un pilar fundamental en mi familia y de gran apoyo en mi toma de decisiones.

EL AUTOR

RODRÍGUEZ ORTÍZ, CRISTIAN EDUARDO

AGRADECIMIENTO

Agradezco de antemano a Dios y a mis padres quienes con sus consejos y recomendaciones me guiaron en el buen camino del estudio.

A mis amigos quienes me brindan siempre su colaboración y consejos.

A mi tutor el Ing. Néstor Zamora quien con sus consejos me pudo ayudar con el desarrollo de este proyecto

EL AUTOR

RODRÍGUEZ ORTÍZ, CRISTIAN EDUARDO



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TRIBUNAL DE SUSTENTACIÓN

f. _____

M. Sc. ROMERO PAZ, MANUEL DE JESUS
DECANO

f. _____

M. Sc. PALACIOS MELÉNDEZ, EDWIN FERNANDO
COORDINADOR DEL ÁREA

f. _____

Ms. Sc. ROMERO ROSERO, CARLOS BOLIVAR
OPONENTE

Índice General

CAPÍTULO 1: DESCRIPCIÓN GENERAL DEL TRABAJO DE TITULACIÓN 2	
1.1. Introducción.....	2
1.2. Antecedentes.	3
1.3. Definición del Problema.....	3
1.4. Justificación del Problema.....	4
1.5. Objetivos de la investigación.	4
1.5.1. Objetivo General.....	4
1.5.2. Objetivos Específicos.	4
1.6. Hipótesis.....	5
1.7. Metodología de Investigación.....	5
CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA	6
2.1. Descripción general de los sistemas de control industrial.	6
2.2. Descripción general de SCADA, DCS y PLC.	7
2.3. Sistemas SCADA	10
2.3.1. Instrumentación de campo.	12
2.3.2. PLC y/o RTU.....	14
2.3.3. Redes de comunicaciones remotas.....	16
2.3.4. Software SCADA Host.....	18
2.4. Requisitos para SCADA	20
2.5. Desafíos de los sistemas SCADA.	21
2.6. Transferencia de datos y comunicación en sistemas SCADA.....	22
2.7. Topologías de comunicaciones en SCADA.....	23
2.8. Tipos de transmisiones en SCADA.	24
2.9. Arquitecturas SCADA.....	26
2.9.1. Primera generación de sistemas SCADA monolíticos.	26
2.9.2. Segunda generación de sistemas SCADA distribuidos.	27
CAPÍTULO 3: DISEÑO, IMPLEMENTACIÓN Y RESULTADOS.....	30

3.1.	Descripción de la herramienta de simulación de instrumentos virtuales.	30
3.2.	Programación Equipo Emisor	31
3.2.1.	Creación de un Nuevo VI para insertar la programación para el Emisor	31
3.2.2.	División de pantallas de Diagrama de Bloque y Panel Frontal del Emisor	32
3.2.3.	Conector Datasocket en LabVIEW	32
3.2.4.	Creación de Switch en LabVIEW	33
3.2.5.	Creación de lazo While en LabVIEW para el Emisor	34
3.3.	Programación Equipo Receptor	37
3.3.1.	Creación de un Nuevo VI para insertar la programación para el Receptor	37
3.3.2.	División de pantallas de Diagrama de Bloque y Panel Frontal de Receptor	37
3.3.3.	Conector Datasocket en LabVIEW	38
3.3.4.	Creación de Switch en LabVIEW	39
3.3.5.	Creación de lazo While en LabVIEW para el Receptor	40
3.3.6.	Demultiplexación de información enviada por Emisor	40
3.4.	Activación de Datasocket en Emisor y Receptor	41
3.5.	Conexión de tarjeta de adquisición de datos DAQ	41
3.6.	Configuración de tarjeta DAQ	41
3.7.	Implementación de tarjeta DAQ a programación de receptor	44
3.8.	Configuración de entrada analógica a la programación	45
3.9.	Armado de circuito con tarjeta DAQ	46
	Conclusiones	48
	Recomendaciones	49
	Glosario de Términos	50
	REFERENCIAS BIBLOGRAFIAS	51

Índice de Figuras

Capítulo 2:

Figura 2. 1: Ejemplo de un ICS para un sistema de utilidad total del sitio.	6
Figura 2. 2: Arquitectura de un sistema de control distribuido (DCS).	8
Figura 2. 3: Componentes de un PLC.....	9
Figura 2. 4: Descripción general del sistema SCADA.....	11
Figura 2. 5: Evolución de los microcontroladores.	12
Figura 2. 6: Diagrama de un sistema de control lógico programable (PLC).	14
Figura 2. 7: Diagrama de unidades de telemetría remotas (RTU).	15
Figura 2. 8: Red de comunicación del sistema de control de supervisión y adquisición de datos (SCADA).....	16
Figura 2. 9: Plataforma SCADA Host.....	19
Figura 2. 10: Protocolos utilizados para la comunicación en las subestaciones.	23
Figura 2. 11: Diagrama del esquema de topologías de comunicaciones SCADA.	24
Figura 2. 12: Tipos de transmisiones en serie y paralelo para transferencia de datos digitales.	25
Figura 2. 13: Explicación gráfica de la transmisión en serie y paralelo para transferencia de datos digitales.	25
Figura 2. 14: Configuración correspondiente a la primera generación de sistemas SCADAs.....	27
Figura 2. 15: Configuración correspondiente a la segunda generación de sistemas SCADAs.....	29

Capítulo 3:

Figura 3. 1: Portada de programa LabVIEW versión 2018	31
Figura 3. 2: Creación de un nuevo VI para insertar la programación para el Emisor.....	31
Figura 3. 3: División de pantallas de Diagrama de Bloque y Panel Frontal del Emisor.....	32

Figura 3. 4: Menú del conector Datasocket en LabVIEW	33
Figura 3. 5: Conector Datasocket en LabVIEW	33
Figura 3. 6: Creación de Switch en LabVIEW	34
Figura 3. 7: Creación de Switch en LabVIEW	34
Figura 3. 8: Creación de lazo While en LabVIEW	35
Figura 3. 9: Creación de lazo While en LabVIEW	35
Figura 3. 10: Creación de lazo While en LabVIEW	35
Figura 3. 11: Creación de lazo While en LabVIEW	36
Figura 3. 12: Creación de bloque para su ejecución después de detener ciclo While.....	36
Figura 3. 13: Creación de un nuevo VI para insertar la programación para el Emisor.....	37
Figura 3. 14: División de pantallas de Diagrama de Bloque y Panel Frontal del receptor.....	37
Figura 3. 15: Conector Datasocket en LabVIEW	38
Figura 3. 16: Conector Datasocket en LabVIEW	38
Figura 3. 17: Creación de Switch en LabVIEW	39
Figura 3. 18: Creación de Switch en LabVIEW	39
Figura 3. 19: Creación de lazo While en LabVIEW para el Receptor.....	40
Figura 3. 20: Demultiplexación de información enviada por Emisor	40
Figura 3. 21: Activación de Datasocket en Emisor y Receptor.	41
Figura 3. 22: Conexión de tarjeta de adquisición de datos DAQ	41
Figura 3. 23: Configuración de tarjeta DAQ.	42
Figura 3. 24: Configuración de tarjeta DAQ.	42
Figura 3. 25: Configuración de tarjeta DAQ.	43
Figura 3. 26: Configuración de tarjeta DAQ.	43
Figura 3. 27: Implementación de tarjeta DAQ a programación de receptor.	44
Figura 3. 28: Implementación de tarjeta DAQ a programación de receptor.	44
Figura 3. 29: Configuración de entra analógica a la programación.....	45
Figura 3. 30: Configuración de entra analógica a la programación.....	45
Figura 3. 31: Configuración de entra analógica a la programación.....	46
Figura 3. 32: Configuración de entra analógica a la programación.....	46
Figura 3. 33: Armado de circuito con tarjeta DAQ.....	47

Índice de Tablas

Capítulo 2:

Tabla 2. 1: Ventajas y desventajas de los instrumentos de control.....	13
--	----

Resumen

En el presente trabajo se diseña e implementa una programación de un algoritmo de control para sistemas SCADA en la cual va a permitir interactuar de manera analógica y digital los conocimientos adquiridos en el salón de clase sobre el uso del simulador LabVIEW y tarjetas de adquisición de datos, para esto se crea dos programaciones llamadas emisor y receptor, en el emisor va a tener la programación designada para poder enviar la información de las interacciones que tiene permitido modificar, el receptor va captar la información por medio del Datasocket y poder enviar la información al circuito armado el cual va a estar conectado por medio de la tarjeta DAQ, en el que se valida tanto por medio de programación y por el circuito armado la información enviada por el emisor. La programación creada en el presente trabajo va a servir como material de apoyo para la formación de profesionales en la carrera Ingeniería en Telecomunicaciones

Palabras claves: SCADA, LabVIEW, DAQ, Analógico, Digital, Datasocket

Abstract

In the present work, a programming of a control algorithm for SCADA systems is designed and implemented in which it will allow to interact in analogue and digital way the knowledge acquired in the classroom about the use of the LabVIEW simulator and data acquisition cards For this purpose, two programs called sender and receiver are created, in the sender it will have the designated programming to be able to send the information of the interactions that it is allowed to modify, the receiver will capture the information through the Datasocket and be able to send the information to armed circuit which will be connected by means of the DAQ card, in which the information sent by the transmitter is validated both by means of programming and by the armed circuit. The programming created in this work will serve as support material for the training of professionals in the Telecommunications Engineering degree.

Keywords: SCADA, LabVIEW, DAQ, Analogical, Digital, Datasocket

CAPÍTULO 1: DESCRIPCIÓN GENERAL DEL TRABAJO DE TITULACIÓN

1.1. Introducción.

A medida que las capacidades técnicas de las computadoras, los sistemas operativos y las redes mejoraron, la gestión organizacional impulsó un mayor conocimiento del estado en tiempo real de las operaciones remotas de la planta. Además, en organizaciones con una serie de operaciones geográficamente separadas, la adquisición remota de datos, el control y el mantenimiento se hicieron cada vez más atractivos desde el punto de vista de la gestión y los costos.

Para (Siguenza Riofrio, 2019; Tandel et al., 2017) estas capacidades se conocen colectivamente como sistemas de control de supervisión y adquisición de datos (*Supervisory Control and Data Acquisition, SCADA*) en otras palabras es un sistema que permite controlar, supervisar y tener la adquisición de datos. Como su nombre indica, no es un sistema de control total, sino que se centra en el nivel de supervisión. Como tal, es un paquete de software puro que se coloca en la parte superior del hardware con el que está conectado, en general a través de controladores lógicos programables (*Programmable Logic Controllers, PLC*) u otros módulos de hardware comercial.

Los sistemas SCADA analizan datos en tiempo real para monitorear y controlar el correcto funcionamiento de los procesos de control. Los sistemas SCADA son componentes vitales de las infraestructuras críticas de la mayoría de las naciones, por ejemplo, controlan tuberías, sistemas de agua y transporte, servicios públicos, refinerías, plantas químicas y una amplia variedad de operaciones de fabricación. SCADA proporciona a la gerencia datos en tiempo real sobre las operaciones de producción; implementa paradigmas de control más eficientes; mejora la seguridad de la planta y el personal y reduce los costos de operación. (Aghenta & Iqbal, 2019)

Estos beneficios son posibles gracias al uso de hardware y software estándar en los sistemas SCADA combinados con protocolos de comunicación mejorados y una mayor conectividad a redes externas, incluida Internet. Sin embargo, estos beneficios se obtienen al precio de una mayor vulnerabilidad a ataques o acciones erróneas de una variedad de fuentes externas e internas.

Para (Toker et al., 2005) los sistemas SCADA usan principalmente control de bucle abierto debido a la comunicación menos confiable. Permite el monitoreo y control remoto de una variedad de dispositivos industriales tan diversos como bombas de agua y gas, interruptores de seguimiento y señales de tráfico.

1.2. Antecedentes.

Desde el desarrollo del sistema SCADA hasta su uso en la industria, hay tres roles asignados a una persona asociada con el SCADA. El primer rol es Desarrollador SCADA (*SCADA Developer*), que desarrolla la lógica central del procesamiento de datos y el mecanismo de control y comunicación. El segundo rol es del Ingeniero SCADA (*SCADA Engineer*), que desarrolla la SCADA HMI para el usuario, de acuerdo con las necesidades y requisitos del sistema SCADA en el lado del usuario y el propósito para el que se utiliza.

El tercer rol es del Operador SCADA (*SCADA Operator*), que monitorea y controla los procesos; reconoce alertas y eventos en los dispositivos remotos utilizando la HMI desarrollada por SCADA Engineer. El SCADA HMI es fácil de utilizar debido a los esfuerzos realizados por los ingenieros de SCADA, que escriben la lógica de visualización detrás del HMI SCADA HMI utilizando las bibliotecas principales desarrolladas por el desarrollador de SCADA.

1.3. Definición del Problema.

Uno de los principales inconvenientes es el desconocimiento de usos de herramientas para aplicaciones usando las tarjetas de adquisición de datos.

¿Qué aplicación se podría implementar para el uso de las tarjetas de adquisición de datos en el entorno de las telecomunicaciones?

Se pudo realizar una aplicación usando el programa Datasocket para poder generar una comunicación por medio de LAN, WAN o Wi-Fi, con la ayuda de simuladores como material de apoyo como LabVIEW.

1.4. Justificación del Problema.

La mayoría de las instituciones nos dan a conocer programas para realizar implementaciones de un sinnúmero de aplicaciones, teniendo en cuenta el uso de para nuevas presentaciones a las siguientes generaciones.

Toda implementación que se realice con fines didácticos es de gran ayuda aporte para los estudiantes. Debido a esto se va a proceder un diseño e implementación de un algoritmo de control usando el sistema SCADA donde se empleara un diseño de redes de comunicaciones para diferentes aplicaciones, no solo será útil para estudiantes de la carrera de Electrónica, sino también se aplica en las Telecomunicaciones en relación a la transmisión de datos que utiliza SCADA.

1.5. Objetivos de la investigación.

1.5.1. Objetivo General.

Realizar el diseño e implementación de un algoritmo de control para sistemas SCADA mediante una tarjeta de adquisición de datos aplicando Datasocket con fines didácticos

1.5.2. Objetivos Específicos.

- Describir los fundamentos teóricos de protocolos y arquitecturas de los sistemas SCADA.
- Explicar el funcionamiento de la plataforma de simulación mediante instrumentación virtual – LabVIEW.
- Diseñar e Implementar una rutina de aprendizaje para el uso de Datasocket con LabVIEW para los estudiantes en ingeniería en Telecomunicaciones con fines didácticos.

1.6. Hipótesis.

Al momento de utilizar LabVIEW como herramienta de apoyo para los estudiantes de ingeniería en Telecomunicaciones en la Universidad Católica de Santiago de Guayaquil van a poder captar de manera más precisa y práctica los problemas propuestos por los docentes en clases.

1.7. Metodología de Investigación.

En la actualidad existen diferentes definiciones relacionadas a los métodos, técnicas, y diseños metodológicos de investigación. Las investigaciones en general son: cuantitativas, cualitativas y de corte mixto (cualitativo-cuantitativo).

Los diseños de investigación se clasifican de acuerdo con el contexto de la investigación. Por este motivo, el presente trabajo de titulación emplea un enfoque cuantitativo basado en los diseños descriptivos, explicativos y exploratorios mediante el método de la simulación – experimentación.

CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA

2.1. Descripción general de los sistemas de control industrial.

Para (Manso, 2016) el sistema de control industrial (*Industrial Control System, ICS*) es un término general que abarca varios tipos de sistemas de control (véase la figura 2.1), incluidos los sistemas (SCADA) que permiten tener el control, supervisión y adquisición de datos, los sistemas (*Distributed Control Systems, DCS*) de control distribuido y otras configuraciones de sistemas de control más pequeños, como los controladores lógicos programables (*Programmable Logic Controllers, PLC*) a menudo utilizados en los sectores industriales e infraestructuras críticas.

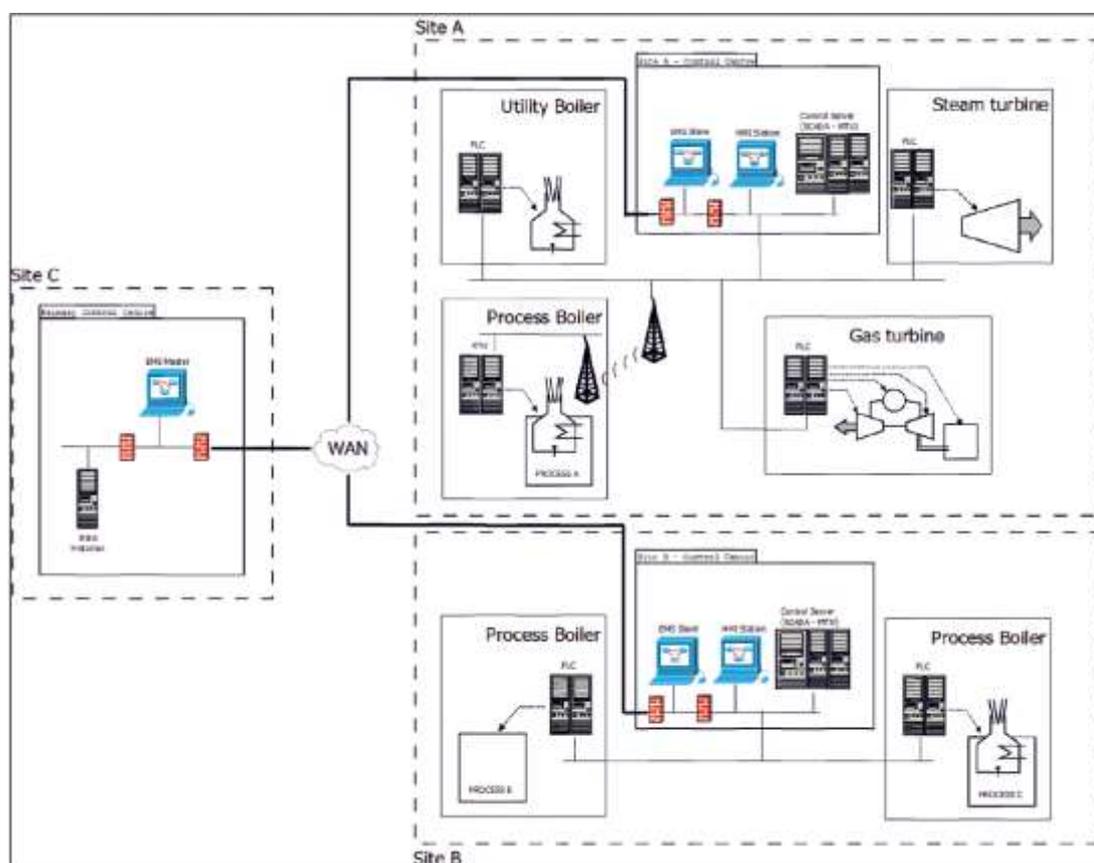


Figura 2. 1: Ejemplo de un ICS para un sistema de utilidad total del sitio.

Fuente: (Manso, 2016)

Los ICS se utilizan típicamente en industrias como electricidad, agua, petróleo y gas, química, transporte, farmacéutica, pulpa y papel, alimentos y bebidas, y fabricación discreta (por ejemplo, automotriz, aeroespacial y bienes duraderos). Estos sistemas de control son críticos a la operación de las

infraestructuras complejas que a menudo son sistemas altamente interconectados y mutuamente dependientes. (Johnson, 2016)

Por ejemplo, en Estados Unidos es importante tener en cuenta que aproximadamente el 90% de las infraestructuras complejas son de propiedad y operación privadas. Las agencias federales también operan muchos de los procesos industriales mencionados anteriormente; otros ejemplos incluyen control de tránsito aéreo y manejo de materiales (por ejemplo, manejo de correo del Servicio Postal).

2.2. Descripción general de SCADA, DCS y PLC.

Los sistemas SCADA son sistemas altamente distribuidos que se utilizan para controlar activos dispersos geográficamente, a menudo dispersos en miles de km², donde la adquisición y el control centralizados de datos son críticos para la operación del sistema. Se utilizan en sistemas de distribución como la distribución de agua y los sistemas de recolección de aguas residuales, tuberías de petróleo y gas, redes de energía eléctrica y sistemas de transporte ferroviario.

Un centro de control SCADA realiza monitoreo y control centralizado para sitios de campo a través de redes de comunicaciones de larga distancia, incluyendo alarmas de monitoreo y procesamiento de datos de estado. Según la información recibida de las estaciones remotas, los comandos de supervisión automatizados o controlados por el operador pueden enviarse a los dispositivos de control de la estación remota, que a menudo se denominan dispositivos de campo. Los dispositivos de campo controlan operaciones locales como abrir y cerrar válvulas e interruptores, recolectar datos de los sistemas de sensores y monitorear el entorno local para detectar condiciones de alarma.

Los DCS generalmente se llegan a dar uso para poder controlar desarrollos industriales como las refinerías de petróleo y gas. Los DCS se dan a conocer como una arquitectura que permite controlar el nivel de supervisión

que supervisa diferentes subsistemas añadidos responsables de controlar detalladamente un proceso fijado.

Para (Mehta & Reddy, 2015) los procesos tan grandes, el DCS comercial es normalmente el sistema de control elegido. La figura 2.2 muestra la arquitectura más común de los sistemas de control distribuido en el mundo de hoy.

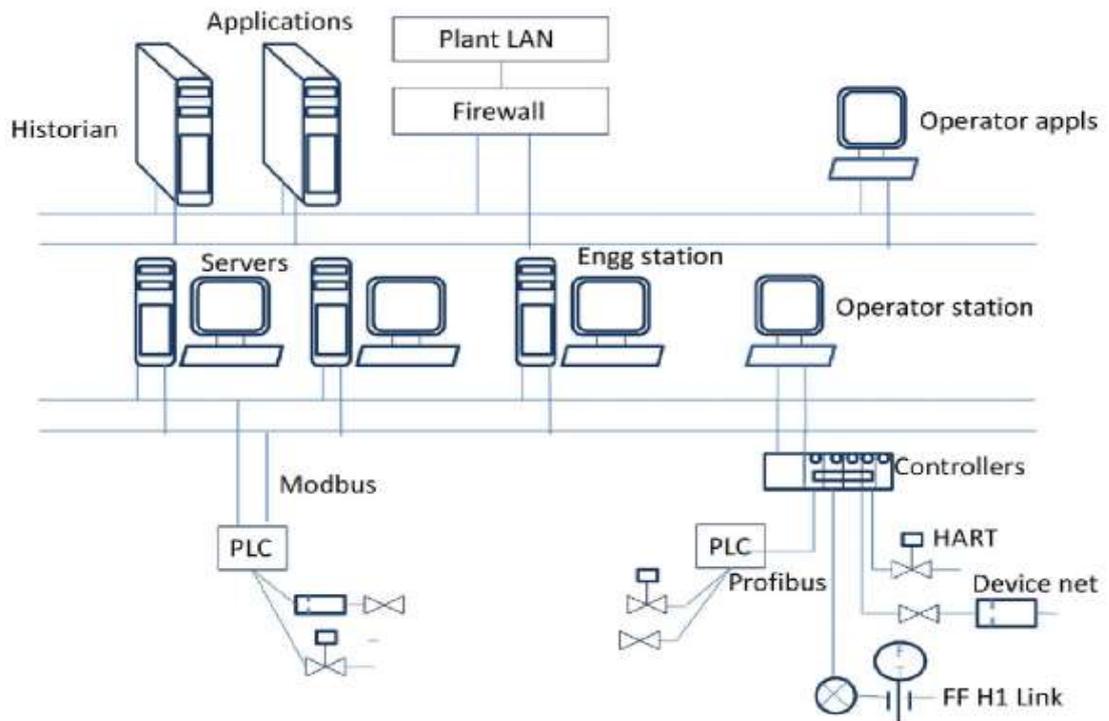


Figura 2. 2: Arquitectura de un sistema de control distribuido (DCS).

Fuente: (Mehta & Reddy, 2015)

El control del producto y el proceso generalmente se logra mediante la implementación de retroalimentación o bucles de control de avance mediante el cual lo fundamental es tener en cuenta las condiciones del producto y los procesos que sostienen alrededor de un área de ajuste deseado. Los controladores programables específicos (PLC) permiten tener una tolerancia del proceso alrededor de un área de ajuste detallado.

Los PLC por lo general son dispositivos que tienen un estado sólido basados en computadoras que les permite controlar otros equipos y procesos industriales, a menudo los PLC son unos de los principales componentes para las configuraciones de sistemas de control de menor tamaño que se da uso

para dar un control regulatorio de los procesos discuto, como controles de sopladores de hollín de platas de energía y líneas de ensamblaje de automóviles y controles de sopladores de hollín de plantas de energía. Los PLC se utilizan ampliamente en casi todos los procesos industriales. La figura 2.3 muestra los diferentes componentes de un PLC.

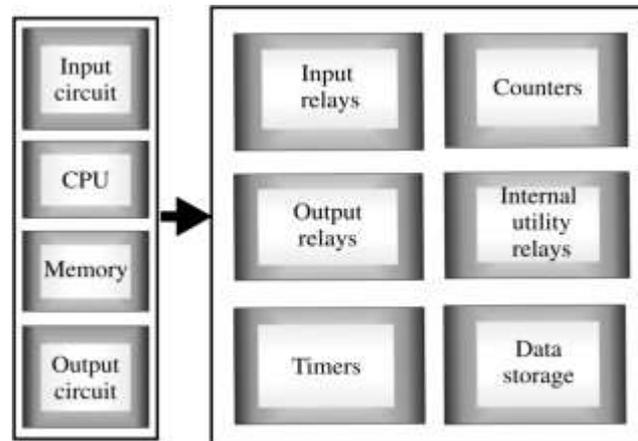


Figura 2. 3: Componentes de un PLC.
Fuente: (Greeff & Ghoshal, 2004)

Según (Fraser, 2001) las industrias de fabricación basadas en procesos suelen utilizar dos procesos principales:

- 1) Procesos de fabricación continua: se ejecutan continuamente, a menudo con transiciones para hacer diferentes grados de un producto. Los procesos típicos de fabricación continua incluyen el flujo de combustible o vapor en una planta de energía, el petróleo en una refinería y la destilación en una planta química.
- 2) Procesos de fabricación por lotes: tienen pasos de procesamiento distintos, realizados en una cantidad de material. Hay un paso inicial y final distinto para un proceso por lotes con la posibilidad de operaciones breves de estado estable durante los pasos intermedios.

Las industrias de fabricación discreta suelen realizar una serie de pasos en un solo dispositivo para crear el producto final. El ensamblaje electrónico y mecánico de piezas y el mecanizado de piezas son ejemplos típicos de este tipo de industria. Tanto las industrias basadas en procesos como las discretas utilizan los mismos tipos de sistemas de control, sensores y redes. Algunas instalaciones son un híbrido de fabricación discreta y basada en procesos.

Si bien los sistemas de control utilizados en las industrias de distribución y fabricación son muy similares en operación, son diferentes en algunos aspectos. Una de las principales diferencias es que los subsistemas controlados por DCS o PLC generalmente se encuentran dentro de una fábrica más confinada o un área centrada en la planta, en comparación con los sitios de campo SCADA geográficamente dispersos. Las comunicaciones DCS y PLC generalmente se realizan utilizando tecnologías de red de área local (LAN) que generalmente son más confiables y de alta velocidad en comparación con los sistemas de comunicación de larga distancia utilizados por los sistemas SCADA.

De hecho, los sistemas SCADA están diseñados específicamente para manejar los desafíos de comunicación a larga distancia, como los retrasos y la pérdida de datos que representan los diversos medios de comunicación utilizados. Los sistemas DCS y PLC generalmente emplean mayores grados de control de circuito cerrado que los sistemas SCADA porque el control de los procesos industriales suele ser más complicado que el control de supervisión de los procesos de distribución. Estas diferencias pueden considerarse sutiles para el alcance de este documento, que se centra en la integración de la seguridad de la tecnología de la información (TI) en estos sistemas.

2.3. Sistemas SCADA

SCADA es una abreviatura de las palabras Control de supervisión y adquisición de datos. Los sistemas SCADA están compuestos por redes, dispositivos electrónicos, sensores de medición, dispositivos de conmutación, etc. La estructura del sistema permite controlar y gestionar procesos local o remotamente, analizar los datos, generar alarmas y enviarlos a diferentes ubicaciones, etc. El término SCADA a menudo se usa para definir el sistema que está geográficamente disperso.

Históricamente, se han producido productos SCADA que son genéricos con un enfoque de "un zapato para todos" en varios mercados. A medida que SCADA ha madurado para proporcionar soluciones específicas a mercados

SCADA específicos, ha proporcionado soluciones para sistemas SCADA de red de área amplia que dependen de enlaces de comunicación tenues. Estos tipos de sistemas SCADA se utilizan ampliamente en todo el mercado de petróleo y gas debido al hecho de que los activos se distribuyen en grandes áreas geográficas.

Mirando la estructura general de un sistema SCADA (véase la figura 2.4), hay cuatro niveles distintos dentro de SCADA, estos son;

- 1) Instrumentación de campo,
- 2) PLC y/o RTU.
- 3) Redes de comunicaciones.
- 4) Software host SCADA.

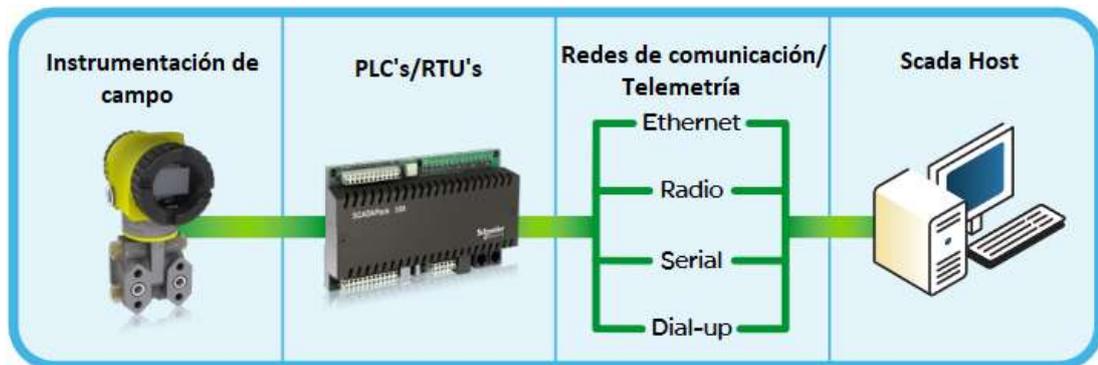


Figura 2. 4: Descripción general del sistema SCADA.
Elaborado por: Autor.

Más adelante, se analizan cada uno de estos niveles en detalle, describiendo su función, cómo SCADA ha cambiado en los últimos 30 años y el impacto de los requisitos de seguridad y el cumplimiento normativo en la operación del sistema SCADA. Un sistema heredado consiste en un host central o maestro, una o más unidades de recopilación y control de datos, también conocidas como unidad terminal remota (*RTU*), adicional tiene una colección de software definido y personalizado dándole uso para el monitoreo y control elementos de datos de campo ubicados de forma remota. La figura 2.5 muestra la estructura típica de un SCADA moderno para la transmisión y distribución eléctrica.

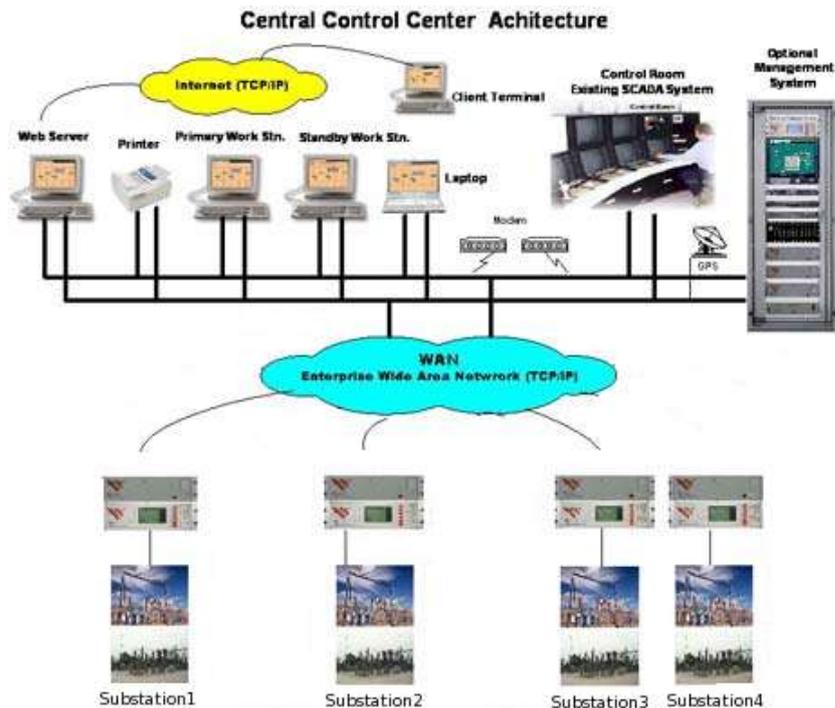


Figura 2. 5: Evolución de los microcontroladores.

Fuente: (Greeff & Ghoshal, 2004)

2.3.1. Instrumentación de campo.

“No se puede controlar lo que no se mide” es un viejo adagio, lo que significa que la instrumentación es un componente clave de un sistema de control seguro y optimizado. Tradicionalmente, las bombas y sus valores operativos correspondientes se habrían controlado manualmente, es decir, un operador arrancar/parar las bombas localmente y las válvulas se habrían abierto/cerrado manualmente. Poco a poco, estos instrumentos se habrían equipado con sensores de retroalimentación, como interruptores de límite, que proporcionan conectividad para estos dispositivos con cable en un PLC o RTU local, para así poder transmitir datos al software SCADA host.

Aunque el técnico de instrumentación de hoy requiere más conocimiento técnico y la capacidad de diseñar, instalar y mantener equipos, que, en el pasado, esto se ve mitigado por el costo reducido en la automatización de procesos y las mayores habilidades técnicas que posee el personal. Hoy en día, la mayoría de los dispositivos de campo, como las válvulas, han sido equipados con actuadores, lo que permite que un PLC o RTU controle el dispositivo en lugar de depender de la manipulación manual. Esta capacidad

significa que el sistema de control puede reaccionar más rápidamente para optimizar la producción o el apagado bajo eventos anormales. La tabla 2.1 se muestra las ventajas y desventajas relacionadas a los instrumentos de control.

Tabla 2. 1: Ventajas y desventajas de los instrumentos de control.

	Instrumentación prematura	Sensores de retroalimentación	Añadir actuadores
Ventaja	La instalación es rentable	Vista central	Control central
Desventaja	Sigue siendo caro de operar	Caro para operar	Mayores requisitos técnicos

Elaborado por: Autor.

En términos de cumplimiento normativo, la instrumentación para la industria del petróleo y el gas ha tenido que cumplir con las clasificaciones de clase, división y grupo peligrosos. El requisito es que el instrumento debe estar diseñado para la ubicación o área en la que se ha colocado, por ejemplo. Un entorno donde se conoce la existencia de vapores explosivos durante condiciones normales de funcionamiento, o durante condiciones anormales.

En muchos casos, el instrumento también debe funcionar en entornos hostiles. Muchos tipos de instrumentación están diseñados para temperaturas extremas de frío y calor. Si la instrumentación no está diseñada para estas temperaturas, se requiere un entorno artificial dentro de un gabinete o algún tipo de edificio. Esto tiene un costo adicional no solo en el diseño inicial sino también para el mantenimiento continuo.

La instrumentación también debe cumplir con cualquier estándar de compatibilidad electromagnética (*Electromagnetic Compatibility, EMC*) que pueda existir, para asegurar que un dispositivo eléctrico no tenga efectos indeseables en su entorno u otros dispositivos eléctricos dentro de su entorno.

2.3.2. PLC y/o RTU.

Los controladores lógicos programables (PLC) y las unidades de telemetría remota (RTU) solían ser dispositivos claramente diferentes, pero con el tiempo ahora son casi lo mismo. Esto ha sido una convergencia de tecnología a medida que los fabricantes de estos dispositivos ampliaron sus capacidades para satisfacer las demandas del mercado. La figura 2.6 muestra el diagrama de conexiones de un sistema de control lógico programable conocido como PLC.

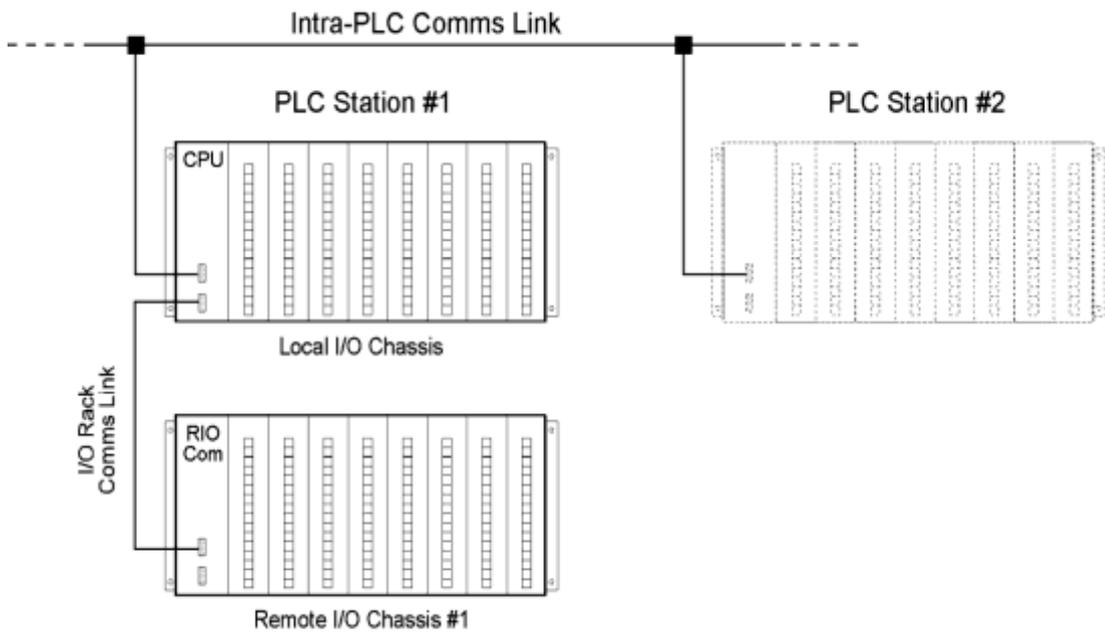


Figura 2. 6: Diagrama de un sistema de control lógico programable (PLC).
Fuente: (Portal, 2018)

Si se retrocede 30 años, una RTU era una caja de telemetría "tonta" para conectar instrumentos de campo. La RTU "retransmitiría" los datos de los instrumentos al SCADA host sin ningún procesamiento o control, pero tenía interfaces de comunicación o telemetría bien desarrolladas. En la década de 1990, la programación de control se agregó a la RTU para que funcionara más como un PLC. Los PLC, por otro lado, siempre podían hacer el programa de control, pero carecían de interfaces de comunicación y capacidad de registro de datos, que fueron incorporados en cierta medida durante la última década.

Un desarrollo adicional de los dispositivos en el campo es ofrecer una aplicación específica que pueda incorporar una serie de instrumentos y dispositivos RTU/PLC, incorporando conjuntos de tecnología para

proporcionar un enfoque "estándar" para los requisitos de procesos comunes, por ejemplo, producción de pozos de gas que incluye elementos de monitoreo, medición de flujo y control que se extenderían como un activo en SCADA host. La figura 2.7 muestra el diagrama esquemático de unidades de telemetría remotas (RTU).

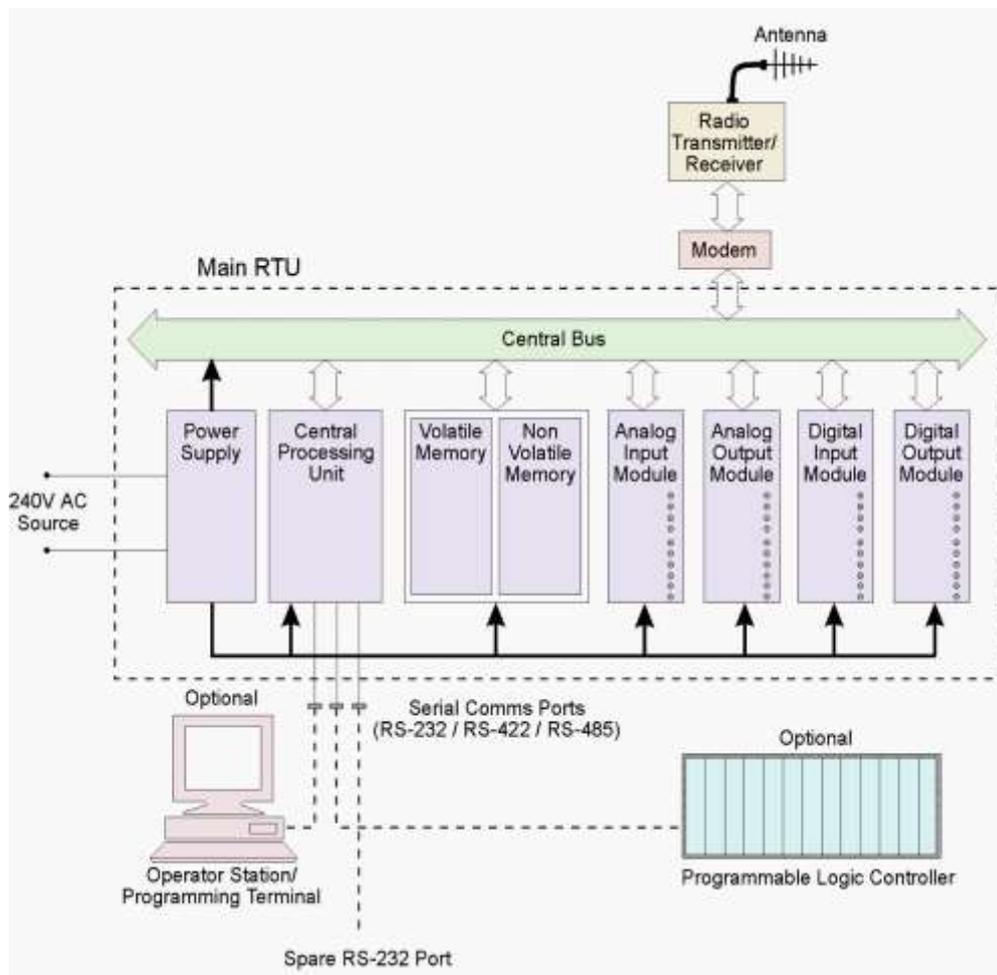


Figura 2. 7: Diagrama de unidades de telemetría remotas (RTU).
Fuente: (Portal, 2018)

En términos de cumplimiento normativo y medioambiental, los PLC y RTU tienen el mismo tipo de requisitos que la instrumentación, ya que operan en el mismo entorno. Sin embargo, los PLC tradicionalmente no han sido tan respetuosos con el medio ambiente como las RTU. Esto se debe principalmente al hecho de que los PLC fueron diseñados para operar en áreas, como pisos de fábricas, donde el ambiente ya estaba condicionado hasta cierto punto.

2.3.3. Redes de comunicaciones remotas.

La red de comunicación remota es necesaria para transmitir datos desde RTU/PLC remotos, que se encuentran en el campo, al host SCADA ubicado en el centro de control central. Con los activos distribuidos en una gran área geográfica, la comunicación es el pegamento o la parte de enlace de un sistema SCADA y es esencial para su funcionamiento. Qué tan bien un sistema SCADA puede gestionar la comunicación con activos remotos es fundamental para el éxito del sistema SCADA.

La figura 2.8 muestra una típica red Ethernet remota utilizando sistemas SCADA.

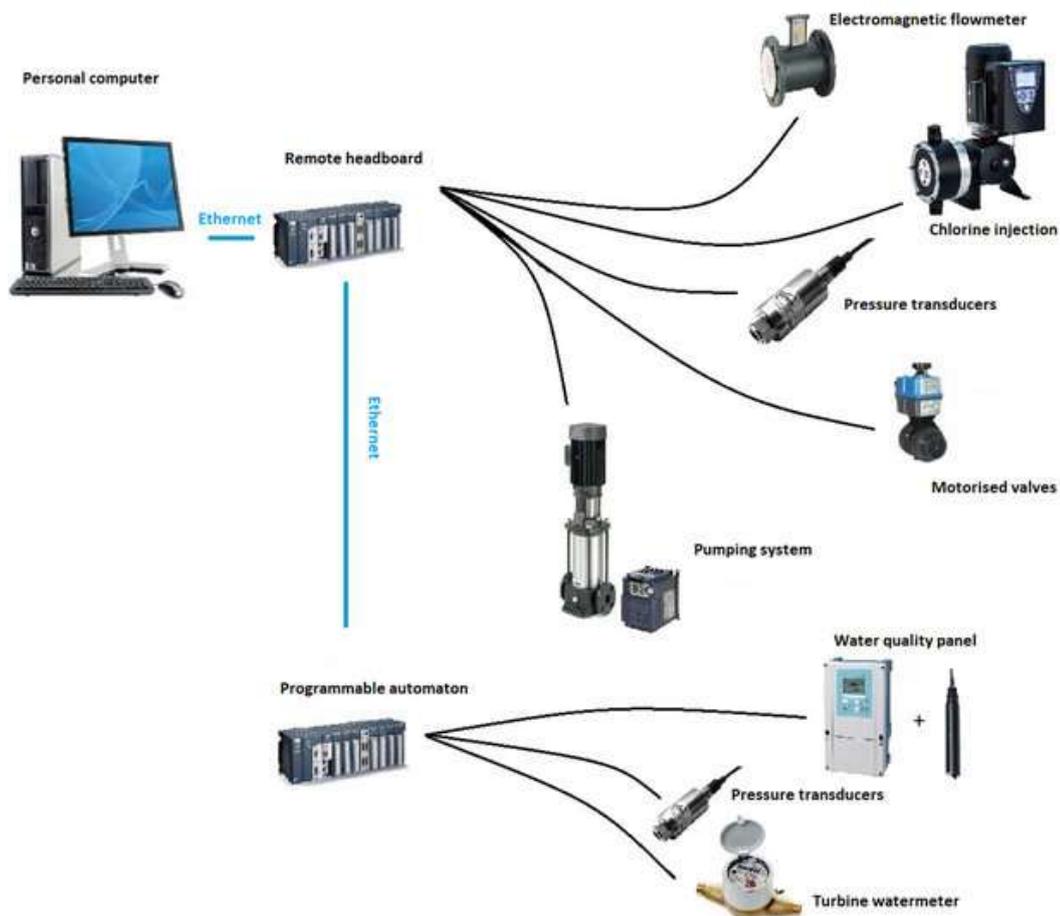


Figura 2. 8: Red de comunicación del sistema de control de supervisión y adquisición de datos (SCADA).

Fuente: (Solé-Torres et al., 2019)

Hace veinte años, la red de comunicaciones habría sido alquilada por líneas o módems de acceso telefónico que eran muy costosos de instalar y mantener, pero en los últimos 10 – 15 años muchos usuarios han cambiado a

comunicaciones por radio o satélite para reducir costos y eliminar el cableado problemático. Más recientemente, se han puesto a disposición otros tipos de comunicación que incluyen comunicaciones celulares y dispositivos de radio mejorados que pueden soportar mayores velocidades de comunicación y mejores diagnósticos. Sin embargo, el hecho de que estos tipos de medios de comunicación aún sean propensos a fallar es un problema importante para los sistemas SCADA modernos y distribuidos.

Al mismo tiempo que el medio de comunicación cambió, también lo hicieron los protocolos. Los protocolos son lenguajes electrónicos que los PLC y RTU utilizan para intercambiar datos, ya sea con otros PLC y RTU o plataformas SCADA Host. Como un desarrollo adicional, muchos fabricantes se inclinaron por un solo protocolo, MODBUS, pero agregaron elementos propietarios para cumplir con los requisitos específicos de funcionalidad.

Por ejemplo, para la industria del petróleo y el gas, hay varias variantes de MODBUS, que incluyen, entre otras, MODBUS ASCII, MODBUS RTU, Enron MODBUS y MODBUS/TCP. Esto proporcionó un estándar de comunicación para la recuperación de flujo o datos de proceso de una RTU o PLC particular.

Este desarrollo incremental en el uso de las variantes del protocolo MODBUS fue visto como una mejora, pero todavía vinculaba a un cliente con un fabricante en particular, que es el caso en la actualidad. Un buen ejemplo es cómo un host SCADA recupera los datos de flujo históricos de una RTU/PLC. Sin embargo, el avance del software SCADA Host, y en algunos casos el intercambio de lenguajes de protocolo ha significado que muchos de los problemas con elementos propietarios se han resuelto aún más.

En los últimos años, han aparecido protocolos verdaderamente no propietarios, como el protocolo de red distribuida (*Distributed Network Protocol, DNP*). Estos protocolos se han creado independientemente de cualquier fabricante y son más un estándar de la industria; muchas personas

y fabricantes se han suscrito a estos protocolos y han contribuido a su desarrollo.

2.3.4. Software SCADA Host.

Tradicionalmente, el software SCADA Host ha sido el mecanismo para ver pantallas gráficas, alarmas y tendencias. El control desde el Host SCADA solo estuvo disponible cuando se desarrollaron elementos de control para instrumentos remotos. Estos sistemas estaban aislados del mundo exterior y eran dominio de operadores, técnicos e ingenieros. Su responsabilidad era monitorear, mantener y diseñar procesos y elementos SCADA. Con los avances en la tecnología de la información (TI), este ya no es el caso. Muchas partes interesadas diferentes ahora requieren acceso en tiempo real a los datos que genera el software SCADA Host.

En consecuencia, existe un impulso para que el host SCADA sea una entidad empresarial que proporcione datos a varios usuarios y procesos diferentes. Esto ha alentado al desarrollo de software SCADA Host a adoptar estándares y mecanismos para admitir la interfaz con estos sistemas. También significa que TI, tradicionalmente separada de los sistemas SCADA, ahora está involucrada en ayudar a mantener las redes, la interfaz de la base de datos y el acceso de los usuarios a los datos, tal como se muestra en la figura 2.9.

Muchos de los productos SCADA Host iniciales se desarrollaron específicamente para el entorno de fabricación donde un sistema SCADA residía dentro de un solo edificio o complejo, y no poseía muchas de las funciones de comunicación de telemetría requeridas por los sistemas SCADA para activos distribuidos geográficamente.

Estos tipos de SCADA Hosts de primera generación a menudo requerían un PLC o RTU híbrido, llamado conductor frontal (*Front-End Driver, FED*) o procesador frontal (*Front-End Processor, FEP*), para ser utilizado para manejar las comunicaciones con dispositivos remotos. Esto dio lugar a una serie de desventajas, ya que requirió una programación especializada,

externa a la plataforma SCADA Host, y creó un cuello de botella en las comunicaciones. Aunque múltiples dispositivos FED o FEP resolvieron algo de esto, hubo costos adicionales y dificultades para crearlos y mantenerlos debido a su naturaleza especializada.

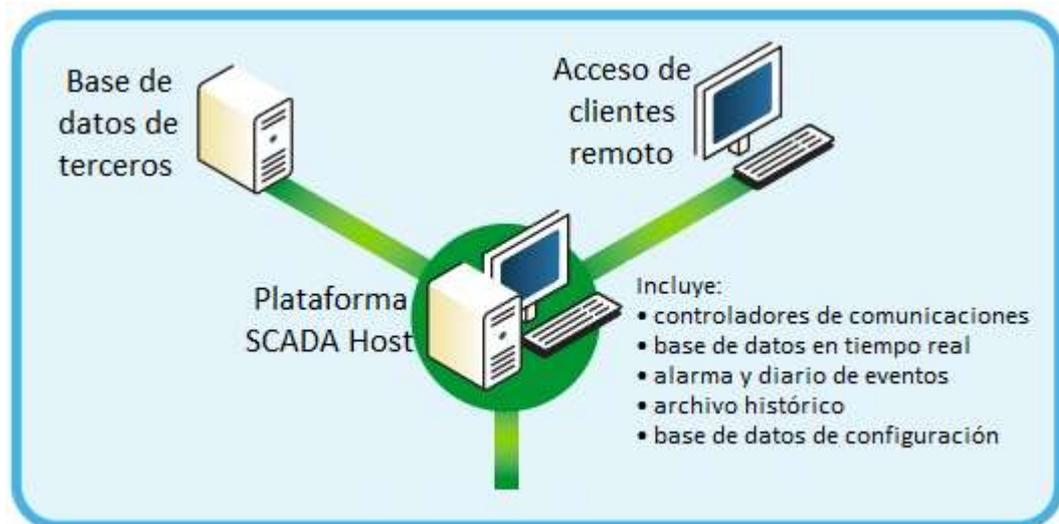


Figura 2. 9: Plataforma SCADA Host.
Fuente: (Riyan, 2016)

El software SCADA moderno que encapsula la funcionalidad de telemetría ya no requiere este tipo de PLC híbridos para las comunicaciones. Ahora usan programas de software llamados "controladores" que están integrados en el host SCADA. Los controladores de software contienen los diferentes tipos de protocolos para comunicarse con dispositivos remotos como RTU y PLC.

A medida que se desarrolló la tecnología, las plataformas de software SCADA Host pudieron aprovechar muchas características nuevas. Estos incluyeron el desarrollo de bases de datos integrales diseñadas específicamente para los requisitos del software SCADA Host, pudiendo manejar miles de cambios por segundo, para sistemas realmente grandes, y así cumplir con la interfaz de base de datos estándar, como la conectividad de base de datos abierta (*Open DataBase Connectivity, ODBC*) y la vinculación e integración de objetos para bases de datos (*Object Linking and Embedding for DataBases, OLE DB*).

Estos estándares son necesarios para que las bases de datos de terceros puedan acceder a los datos del software SCADA Host. El acceso remoto del cliente al SCADA Host es otra tecnología que ha permitido a los usuarios operar y monitorear los sistemas SCADA mientras se mueven entre o en otras ubicaciones. Hay un impulso hacia la seguridad operacional para los sistemas SCADA Host dentro de la industria del petróleo y el gas. Las regulaciones de administración de la sala de control analizan el software SCADA Host y cómo funciona en términos de operaciones, mantenimiento y administración. También cubre el grado de integración del sistema SCADA y su uso de arquitectura y estándares abiertos.

2.4. Requisitos para SCADA

Las condiciones eléctricas anormales en el sistema de energía son detectadas y manejadas por relés de protección. La capacidad de restaurar correctamente el sistema de energía a la operación normal depende en gran medida tanto de la configuración de los relés como de la comprensión de la topología del sistema de energía. (Djokic et al, 2005)

Los usuarios finales y las empresas de servicios eléctricos están cada vez más preocupados por la calidad y la fiabilidad del sistema eléctrico y todas las perturbaciones disminuyen la calidad total de la electricidad. La automatización ha reemplazado al personal de mantenimiento local en la subestación y la automatización también ha aumentado la confiabilidad de la distribución, pero no puede realizar controles y operaciones sin la correcta parametrización y configuración de las aplicaciones y dispositivos. Para manejar estas cosas, el personal de mantenimiento en la sala de control aún debe tener una buena percepción del sistema de energía.

Como resultado de los desarrollos tecnológicos, se pueden definir los requisitos para el monitoreo y control. Los cambios en el sistema deben ser rastreados y monitoreados muy de cerca, de manera confiable y los eventos y consecuencias son necesarios para analizarlos automáticamente para que puedan reaccionar a las posibilidades del sistema rápidamente. Esto crea presión para los proveedores de aplicaciones y dispositivos.

Hoy en día, la situación en la subestación es bastante desafiante, especialmente para los integradores de sistemas debido a una amplia gama de dispositivos y proveedores diferentes. Esto crea la necesidad de desarrollar controles y sistemas de monitoreo continuamente.

2.5. Desafíos de los sistemas SCADA.

Una topología del sistema de potencia se define, como el estado del interruptor automático. La topología del sistema se actualiza automáticamente mediante el funcionamiento de los relés de protección o, en algunos casos, se realiza manualmente. No importa cuál sea la convención, es posible que los datos topológicos no reflejen el estado de conmutación real de los dispositivos de alimentación en el campo. Si el valor de medición es malo, podría causar un error notable en el largo período de tiempo. La medición incorrecta de la energía causa errores en los informes diarios, mensuales y anuales.

Los operadores también pueden tener dificultades para tomar decisiones para restaurar partes del sistema de energía que pueden haber sido desconectadas automáticamente por los dispositivos electrónicos inteligentes (*Intelligent Electronic Devices, IED*), especialmente si los operadores no tienen una forma clara de verificar las condiciones del sistema o la topología relacionada.

El problema con SCADA no es la falta de datos. Hay más datos disponibles de los que se utilizan, pero el problema es cómo usarlos racionalmente. Las RTU antiguas no están diseñadas para rastrear las mediciones analógicas en tiempo real. Solo se implementa el informe por excepción cuando se superan ciertos umbrales. (Somestad et al., 2010)

Durante fallas eléctricas en la red, por ejemplo, falla de cortocircuito, las señales analógicas cambian muy rápidamente, y solo un seguimiento más cercano puede proporcionar información sobre las relaciones de fase y los cambios dinámicos en las magnitudes de la señal. Si las fallas son necesarias para detectar con RTU, se deben usar los registradores de perturbaciones

externas. Si se utilizan nuevos IED, ofrece más oportunidades para reaccionar ante las fallas y obtener información sólida.

Un problema importante es también la incompatibilidad de los dispositivos y la inexistencia de un estándar relevante que unifique las formas de datos producidos por la variedad de dispositivos. La información de los dispositivos difiere entre los protocolos y, por esa razón, no es simple hacer que los dispositivos se comuniquen entre sí. En el futuro, un nuevo estándar de comunicación de subestación, IEC 61850, puede facilitar este problema.

2.6. Transferencia de datos y comunicación en sistemas SCADA.

Esta sección trata algunos principios de la transferencia de datos. La primera parte presenta en breve transmisión en serie y en paralelo. Estas técnicas todavía están en uso, pero podría decirse que especialmente la transmisión en serie pura ya es una técnica anticuada en la subestación porque las necesidades y demandas de comunicación han aumentado. La transmisión en serie no puede transferir suficientes datos tan rápido como se requiere. (Fovino, 2014)

Se han desarrollado muchos protocolos de comunicación diferentes en las últimas décadas. Las diferentes soluciones y dispositivos establecen diferentes demandas para las señales y la comunicación. La figura 2.10 muestra algunos protocolos que pueden utilizarse para la comunicación en los sistemas de automatización de transmisión y distribución eléctrica.

De la figura 2.10 se tiene que SU es la unidad de estación, PU es la unidad de protección, CU es la unidad de control, C/P es la unidad de control/protección, CT/VT es el transformador de instrumentos, ComU, es la unidad de comunicación, BS es la estación de bus, PB es el bus de procesos, y SG es la subestación de control.

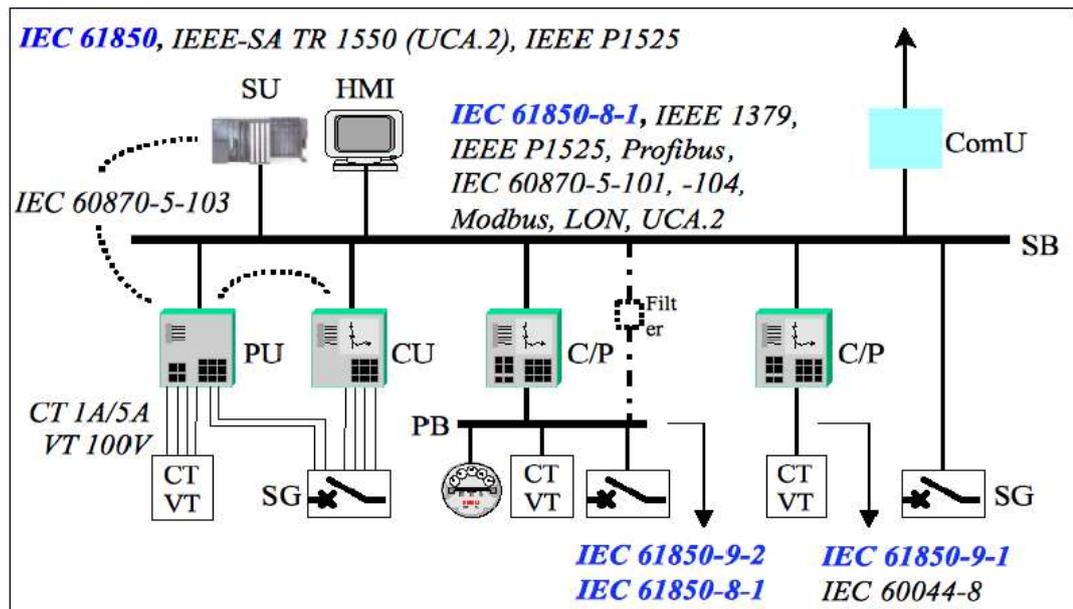


Figura 2. 10: Protocolos utilizados para la comunicación en las subestaciones.
Fuente: (Arghira et al., 2011)

2.7. Topologías de comunicaciones en SCADA.

La red de comunicación se puede construir de varias maneras. Algunas de las estructuras comunes utilizadas en el sistema SCADA se representan de manera simplificada en la figura 2.11. Las arquitecturas de comunicación son diferentes según la implementación.

La figura 2.11 muestra cuatro tipos de arquitectura utilizados, donde se detalla a continuación los tipos de arquitectura:

- Punto a punto: funcionalmente que no tiene mayor dificultad, sin embargo, los gastos aumentan ya que es necesario tener canales individuales para cada conexión.
- En serie: Este tipo de configuración permite reducir la cantidad de canales utilizados, pero al momento de compartir canales baja la eficiencia y complejidad de las operaciones SCADA.
- Estrella, en serie y multipunto: Este tipo de configuraciones necesitan un canal por dispositivo lo que disminuye su eficiencia, pero va a tener una mayor complejidad del sistema.

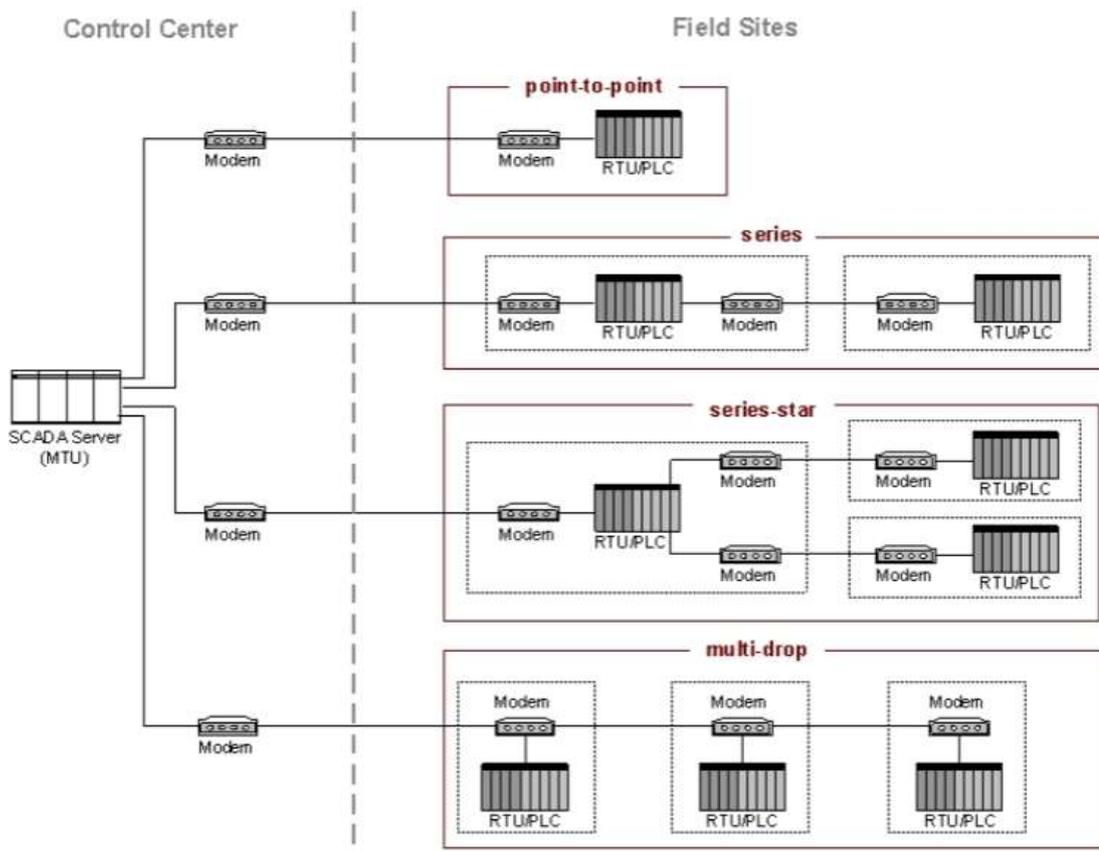


Figura 2. 11: Diagrama del esquema de topologías de comunicaciones SCADA.

Fuente: (Arghira et al., 2011)

2.8. Tipos de transmisiones en SCADA.

La solución tradicional en la transferencia de datos digitales es la transmisión en serie o en paralelo. En la comunicación en serie, los datos se envían un bit a la vez desde el transmisor al receptor y en la comunicación en paralelo, varios bits de datos o bytes completos se transfieren simultáneamente a través de varios canales paralelos. La figura 2.12 muestra la diferencia entre las transferencias de datos en serie y paralelo.

La comunicación paralela se utiliza en distancias cortas, generalmente dentro de las computadoras. La transmisión paralela es una forma rápida de transferir datos, pero a larga distancia necesita más cables, lo que hace que el manejo de errores sea más desafiante y, por lo tanto, los costos aumentan. Mientras que la transmisión en serie se utiliza generalmente para transferir datos entre dos sistemas informáticos o sistemas donde la distancia de transferencia es larga.

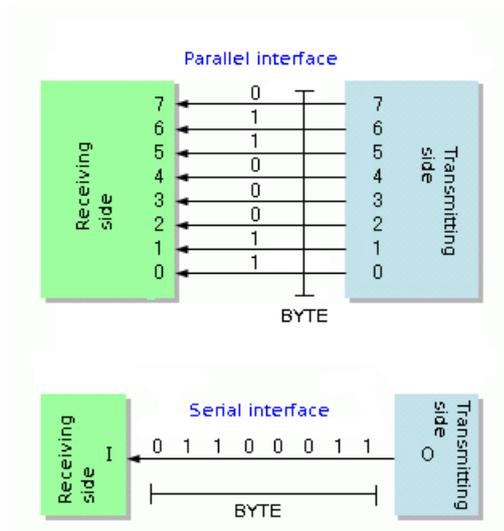


Figura 2. 12: Tipos de transmisiones en serie y paralelo para transferencia de datos digitales.

Fuente: (Arghira et al., 2011)

Los protocolos, por ejemplo, SPA, utilizado en la automatización de subestaciones, se basan en un modo serie. La transmisión en serie puede ser síncrona o asíncrona. En la transmisión síncrona, los grupos de bits se combinan en tramas y las tramas se envían continuamente, con o sin datos, al receptor.

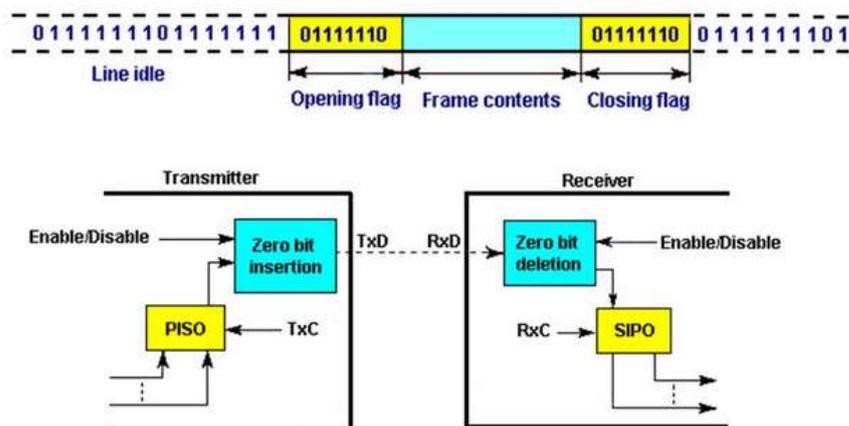


Figura 2. 13: Explicación gráfica de la transmisión en serie y paralelo para transferencia de datos digitales.

Fuente: (Thomas & McDonald, 2017)

En la transmisión asíncrona, los grupos de bits se envían como unidades independientes con bandera de apertura/cierre y sin sincronización de enlace de datos, para permitir espacios de tamaño arbitrario entre tramas.

Sin embargo, los bits de inicio/parada mantienen la sincronización física del nivel de bits una vez detectados. La figura 2.13 aclara la situación de las transmisiones síncronas y asíncronas.

2.9. Arquitecturas SCADA.

La evolución de los sistemas SCADA están a la par con la sofisticación y crecimiento de la tecnología informática actual. Se procede a detallar las generaciones de sistemas SCADA

2.9.1. Primera generación de sistemas SCADA monolíticos.

Cuando se desarrollaron por primera vez los sistemas SCADA, el concepto de informática en general se centró en los sistemas "mainframe". Las redes eran generalmente inexistentes, y cada sistema centralizado estaba solo. Como resultado, los sistemas SCADA eran sistemas independientes prácticamente sin conectividad a otros sistemas. Las redes de área amplia (*Wide Area Networks, WAN*) que se implementaron para comunicarse con unidades terminales remotas (RTU) se diseñaron con un único propósito en mente: comunicarse con las RTU en el campo y nada más. Además, los protocolos WAN en uso hoy en día eran en gran medida desconocidos en ese momento.

Los protocolos de comunicación utilizados en los sistemas SCADA han sido diseñados por empresas proveedoras de equipos RTU y, por lo general, eran propietarios. Además, estos protocolos generalmente eran muy "ajustados", y no admitían prácticamente ninguna funcionalidad más allá de los puntos de escaneo y control requeridos de manera interna en el equipo remoto. Al mismo tiempo, esto no es posible combinar diferentes tipos de tráfico de información en las transmisiones RTU en la red.

La conectividad a la estación maestra del sistema SCADA era restringida por los propietarios a través de las empresas que venden estos sistemas. Por ejemplo, para conectar la estación maestra SCADA se requiere utilizar a nivel de bus, controladores (adaptadores) licenciados conectado en la parte

posterior de la Unidad Central de Procesamiento (*Central Processing Unit, CPU*).

Para la primera generación de SCADA la redundancia fue posible a través del empleo de sistemas mainframe proveídos de igual manera. Por ejemplo, el primero se denomina “primario” y el otro se llama de “respaldo”, cuya conexión es nivel de bus. De tal manera, que una de las funciones importantes del sistema fue el monitoreo del “primario” y este se encargaba siempre de la espera que existía detección de fallos. Es decir, que el procesamiento fue bajo o nulo en los sistemas de operación en espera. La figura 2.14 muestra una arquitectura típica SCADA de primera generación.

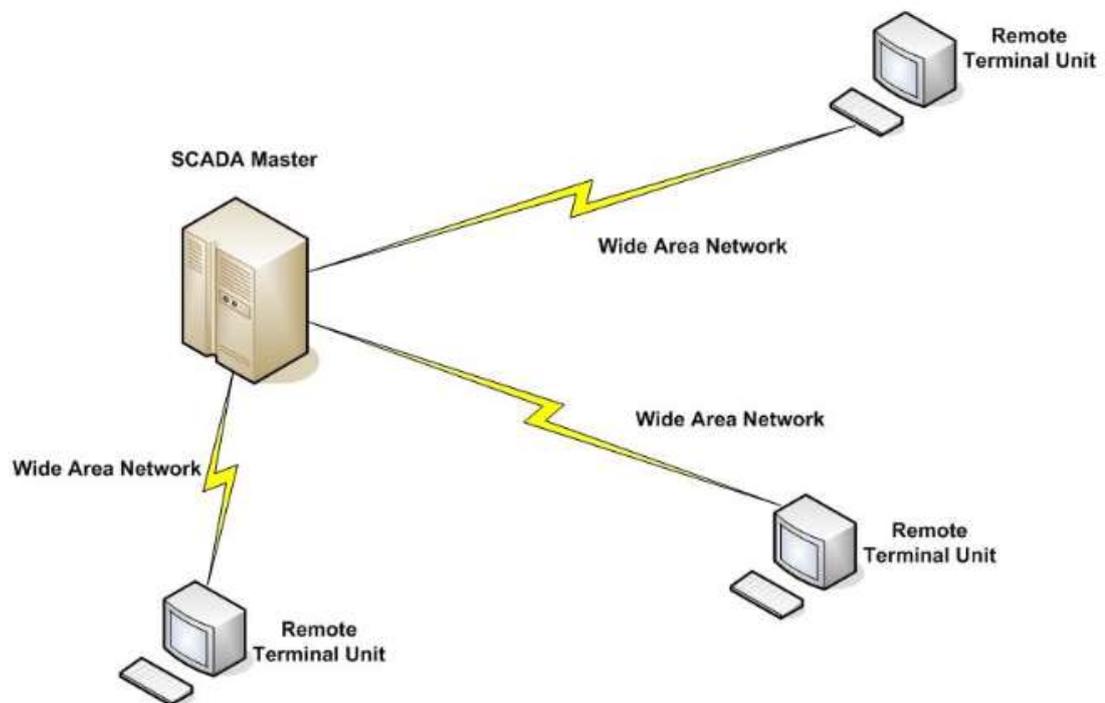


Figura 2. 14: Configuración correspondiente a la primera generación de sistemas SCADA.

Fuente: (Thomas & McDonald, 2017)

:

2.9.2. Segunda generación de sistemas SCADA distribuidos.

La segunda generación de este sistema fue SCADA distribuido que se benefició de las mejoras en la minimización del sistema y también aprovechaba el empleo de una red de área local (*Local Area Networks, LAN*) para distribuir el procesamiento mediante el uso de múltiples sistemas. Por ejemplo, para el caso de múltiples estaciones, ellas tenían funciones o tareas definidas. Estas tareas, por ejemplo, eran de la conexión a una red de área

local (LAN) y transmitían información entre ellos en tiempo real. Generalmente, este tipo de estaciones eran un tipo minicomputadora, es decir, mucho más pequeñas y de bajo costo con relación a los de primera generación.

Determinadas estaciones de trabajos distribuidas son utilizadas como dispositivo de procesamiento de las comunicaciones entre dispositivos RTU. En otras palabras, estas estaciones se comportan como una interfaz de operación, conocida como interfaz hombre-máquina (*Human-Machine Interface, HMI*). Otras fueron utilizadas con propósitos de procesamiento computacional o como plataforma de base de datos.

Estos sistemas múltiples de distribución en SCADA suministraban mayor potencia de procesamiento de información que estaba disponible en un único procesador. El tipo de redes que permitía sistemas característicos eran basadas en redes de área local y con la desventaja que no podían desplegarse más allá del entorno local.

Algunos de los protocolos LAN que se utilizaron eran de naturaleza patentada, donde el proveedor creó su propio protocolo de red o versión de este en lugar de retirar uno existente del estante. Esto permitió a un proveedor optimizar su protocolo LAN para el tráfico en tiempo real, pero limitó (o eliminó efectivamente) la conexión de red de otros proveedores a la LAN SCADA.

La figura 2.15 muestra la configuración de la arquitectura típica de segunda generación en sistemas SCADAs. Además, del incremento de potencia en el procesamiento computacional distribuido también se pueden apreciar mejoras a nivel de repetición (redundancia) y seguridad del sistema en su conjunto. Es decir, que una estación de trabajo HMI podría fallar la comunicación con otra estación dentro de la red mediante operadores de aplicaciones LabVIEW.

Como ya se explicó anteriormente la segunda generación de los sistemas SCADAs no permiten conexiones locales en redes de área amplia

(Wide Area Network, WAN). Las WAN's sirven para transmitir información de datos en dispositivos RTU tal como ocurrió en la primera generación de SCADA, la segunda generación de sistemas SCADA también se limitó al hardware, software y dispositivos periféricos que fueron provistos o a los menos seleccionados por el proveedor.

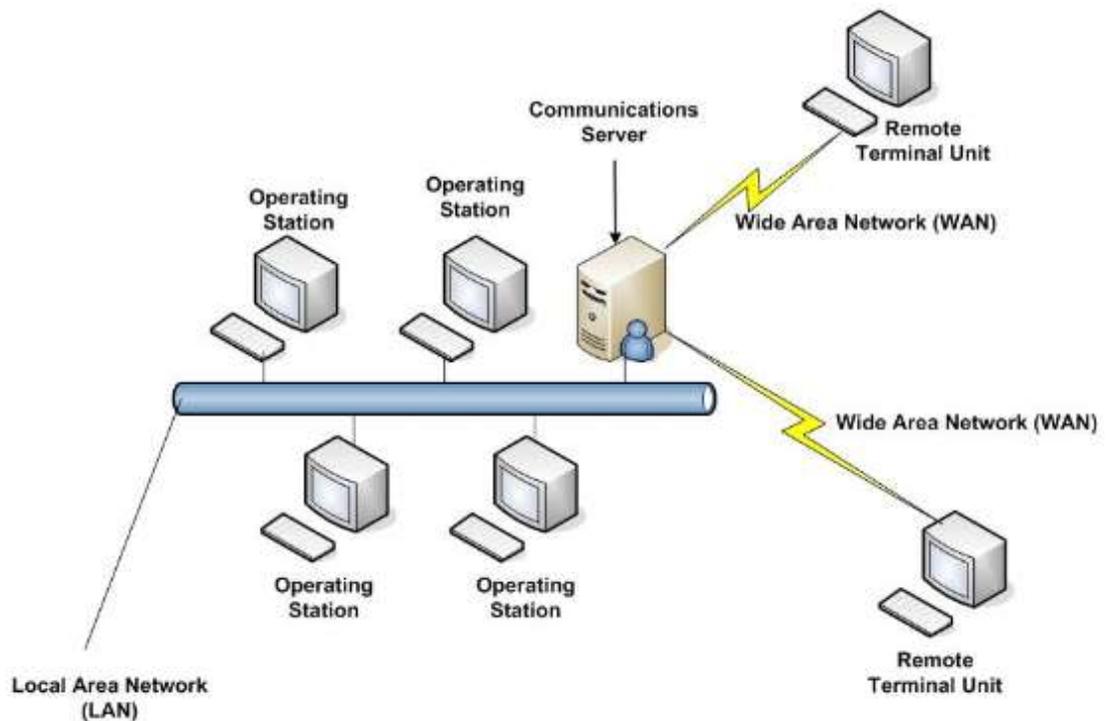


Figura 2. 15: Configuración correspondiente a la segunda generación de sistemas SCADAs.

Fuente: (Thomas & McDonald, 2017)

CAPÍTULO 3: DISEÑO, IMPLEMENTACIÓN Y RESULTADOS

En este capítulo se procede a explicar de manera detallada la programación realizada en el presente trabajo con el simulador LabVIEW. Se realiza la programación para la transmisión de información por medio del Datasocket al receptor, el receptor va a tener conectado el circuito con la tarjeta DAQ en la cual se va a poder validar los cambios o información realizada por la programación del emisor y adicional se observa las mismas acciones realizadas en el circuito. El presente capítulo se basa en la obtención de los objetivos específicos descritos en el capítulo 1.

En el capítulo 2 del presente trabajo de titulación se procedió a describir y con revisión bibliográfica de manera detallada los protocolos y arquitecturas aplicadas al sistema SCADA,

3.1. Descripción de la herramienta de simulación de instrumentos virtuales.

LabVIEW desarrollado por la empresa National Instrument es una plataforma de simulación que utiliza instrumentos virtuales para el modelamiento de sistemas en tiempo real y no real. Dispone de múltiples instrumentos virtuales en sus librerías que asemejan a equipos reales. En la figura 3.1 se observa la portada del programa LabVIEW versión 2018 donde se realizará la programación del emisor y receptor del trabajo presentado.

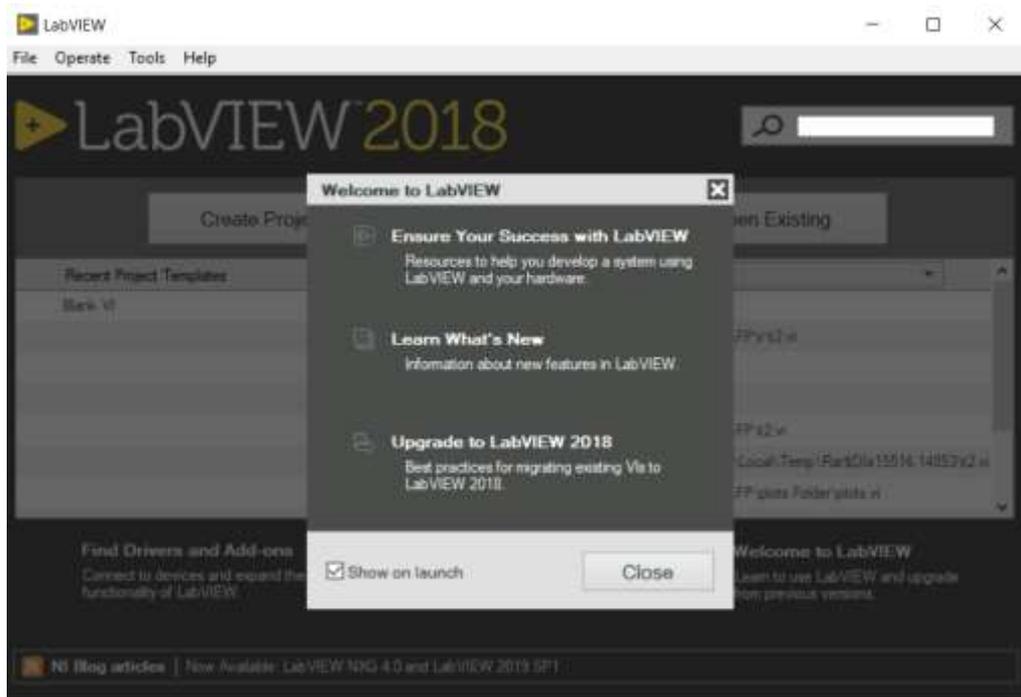


Figura 3. 1: Portada de programa LabVIEW versión 2018
Elaborado por: Autor.

3.2. Programación Equipo Emisor

3.2.1. Creación de un Nuevo VI para insertar la programación para el Emisor

Como se observa en la figura 3.2 se procede a crear un nuevo proyecto donde se va a desplegar dos ventanas, donde se procederá a crear la programación designada para el Emisor

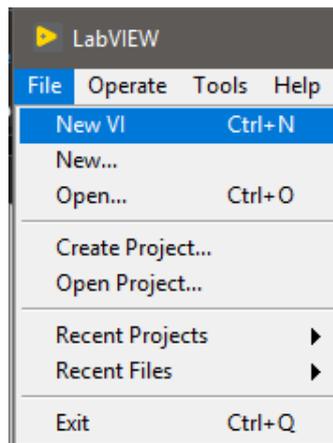


Figura 3. 2: Creación de un nuevo VI para insertar la programación para el Emisor
Elaborado por: Autor.

3.2.2. División de pantallas de Diagrama de Bloque y Panel Frontal del Emisor

Se procede a dividir las pantallas presionando en el teclado Ctrl+T para poder separar el Diagrama de Bloque que es donde se va a realizar la programación interna y el Panel Frontal nos permite diseñar y crear la parte visualizada por el usuario.

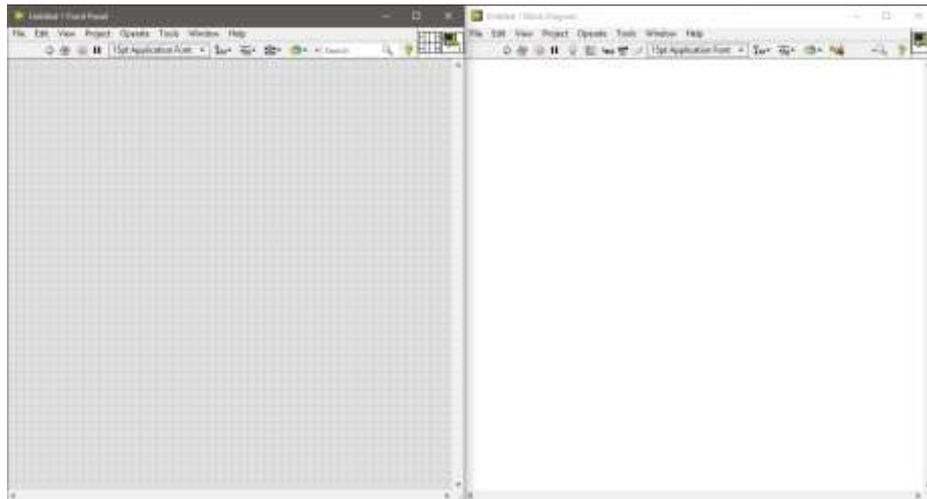


Figura 3. 3: División de pantallas de Diagrama de Bloque y Panel Frontal del Emisor

Elaborado por: Autor.

3.2.3. Conector Datasocket en LabVIEW

En la ventana bloque de diagrama clic derecho sobre la sección en blanco se desplegará una ventana emergente, seleccionar Data comunicación y Datasocket.

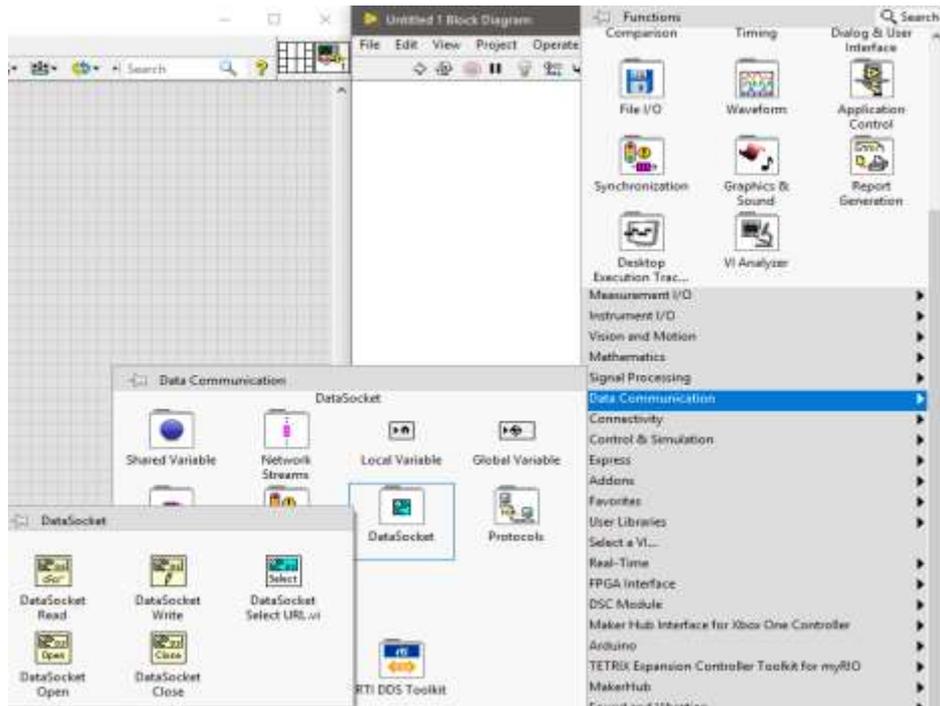


Figura 3. 4: Menú del conector Datasocket en LabVIEW
Elaborado por: Autor.

Se procede a seleccionar la herramienta de Datasocket Open donde se configura la dirección en la cual se realiza la conexión entre las dos computadoras.

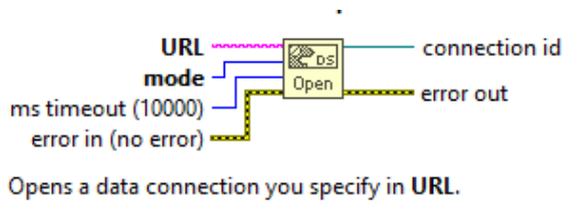


Figura 3. 5: Conector Datasocket en LabVIEW
Elaborado por: Autor.

Se procede la creación de una dirección donde se almacenará el Datasocket y se transmitirá la información entre los dos VI con el nombre `dstp://localhost/información` y se configura el modo si es de escritura.

3.2.4. Creación de Switch en LabVIEW

Crear los Switch para el envío de datos desde un computador a otro para controlar el valor. Clic derecho en el panel de control y seleccionar `Control>>Modernos>Push botón`

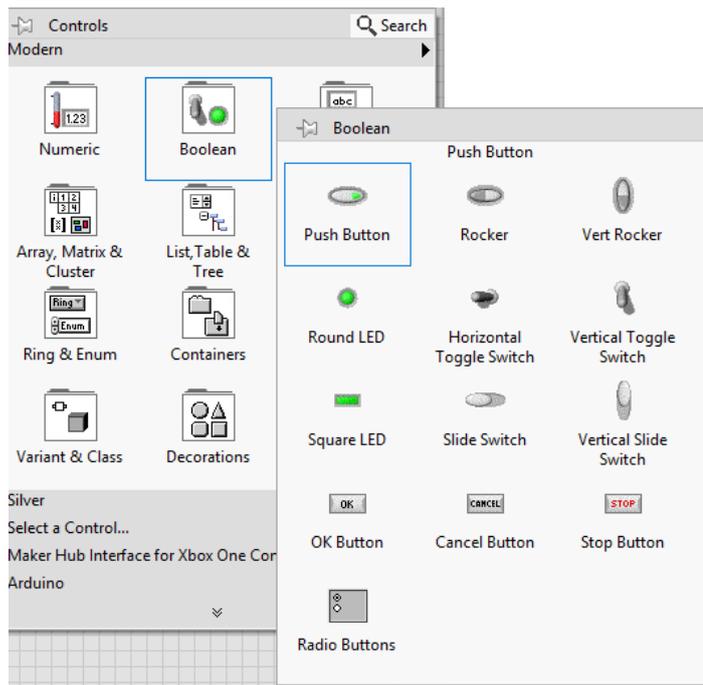


Figura 3. 6: Creación de Switch en LabVIEW
Elaborado por: Autor.

Como se muestra en la figura Crear 8 botones con el nombre botón 0 hasta botón 7.

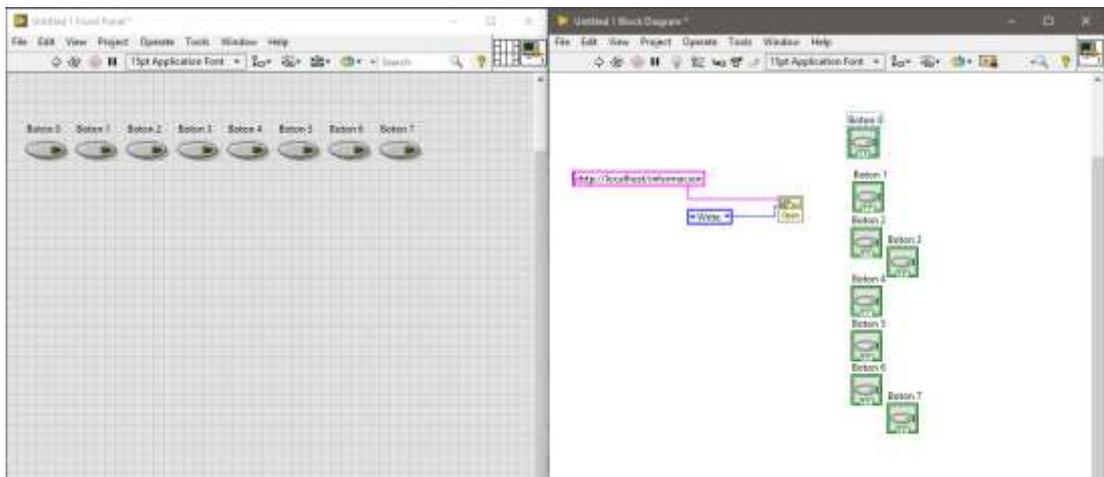


Figura 3. 7: Creación de Switch en LabVIEW
Elaborado por: Autor.

3.2.5. Creación de lazo While en LabVIEW para el Emisor.

Se procede a agregar un lazo de repetición While donde se ejecutará el programa.

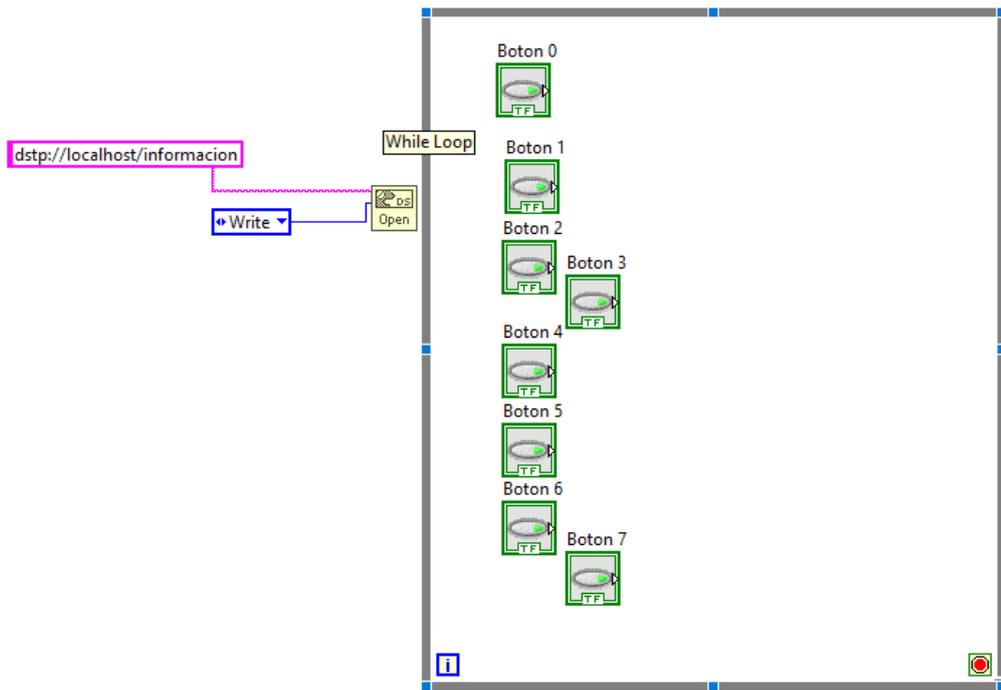


Figura 3. 8: Creación de lazo While en LabVIEW
Elaborado por: Autor.

Una vez agregado el lazo While se procede a incluir un bloque de escritura del Datasocket donde se enviará el dato.

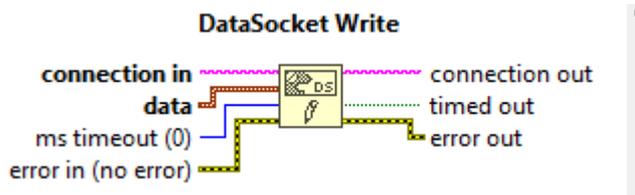


Figura 3. 9: Creación de lazo While en LabVIEW
Elaborado por: Autor.

Después se procede a adicionar un bloque que permita la combinación de señales mediante una misma línea.

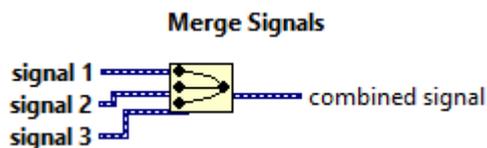


Figura 3. 10: Creación de lazo While en LabVIEW
Elaborado por: Autor.

Se procede a agregar la señal binada y enviada por el Datasocket de escritura.

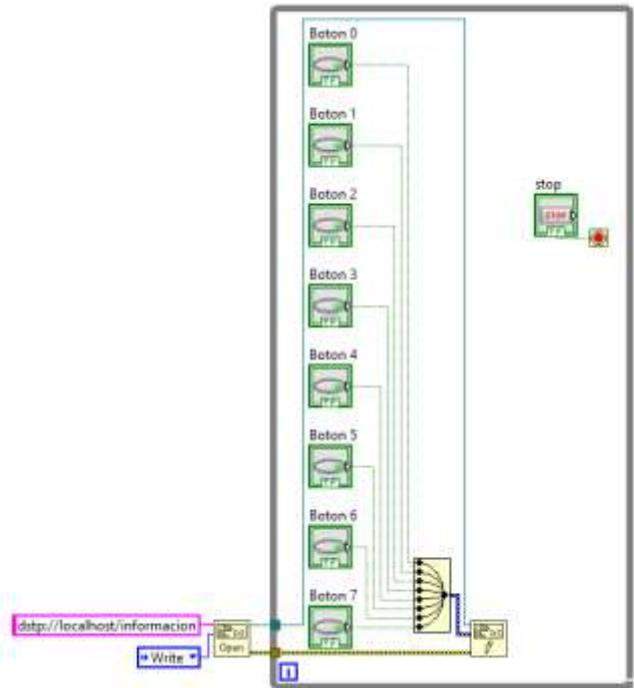


Figura 3. 11: Creación de lazo While en LabVIEW
Elaborado por: Autor.

Por último, se procede a agregar un bloque que se ejecute luego de detener el ciclo While que cierre el Datasocket.

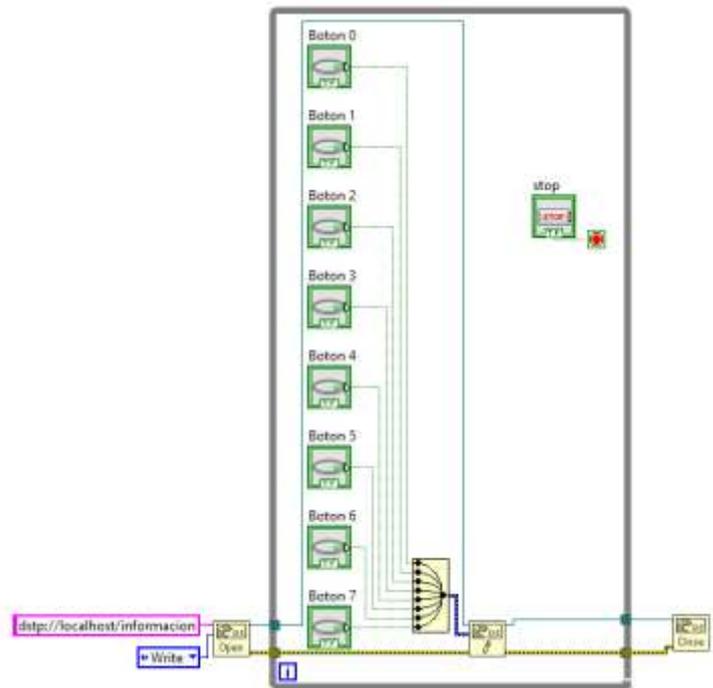


Figura 3. 12: Creación de bloque para su ejecución después de detener ciclo While.
Elaborado por: Autor.

3.3. Programación Equipo Receptor

3.3.1. Creación de un Nuevo VI para insertar la programación para el Receptor

Se crea un nuevo Vi que sirve para la recepción de los datos del Datasocket en otro computador donde se procederá a colocar el Datasocket en modo lectura con la misma dirección del Datasocket

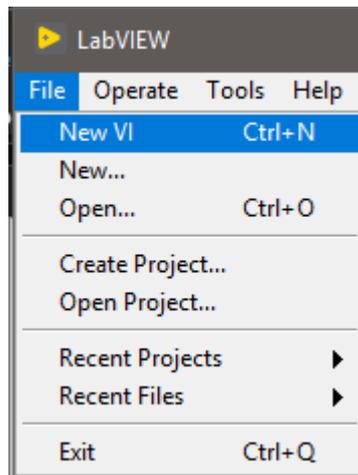


Figura 3. 13: Creación de un nuevo VI para insertar la programación para el Emisor
Elaborado por: Autor.

3.3.2. División de pantallas de Diagrama de Bloque y Panel Frontal de Receptor.

Se procederá a dividir las pantallas presionando en el teclado **Ctrl+T** para poder separar el Diagrama de Bloque que es donde se realizará la programación interna y el Panel Frontal nos permite diseñar y crear la parte visualizada por el usuario.

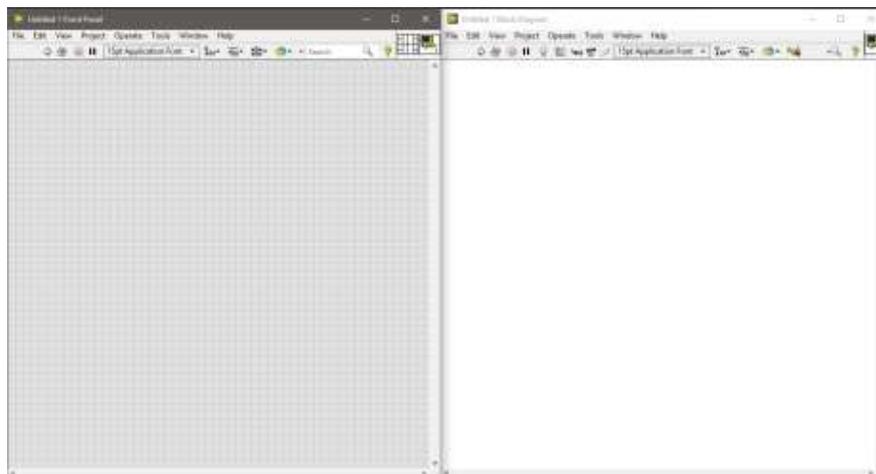


Figura 3. 14: División de pantallas de Diagrama de Bloque y Panel Frontal del receptor.

Elaborado por: Autor.

3.3.3. Conector Datasocket en LabVIEW

En la ventana bloque de diagrama click derecho sobre la sección en blanco se desplegará una ventana emergente, seleccionar Data comunicación y Datasocket.

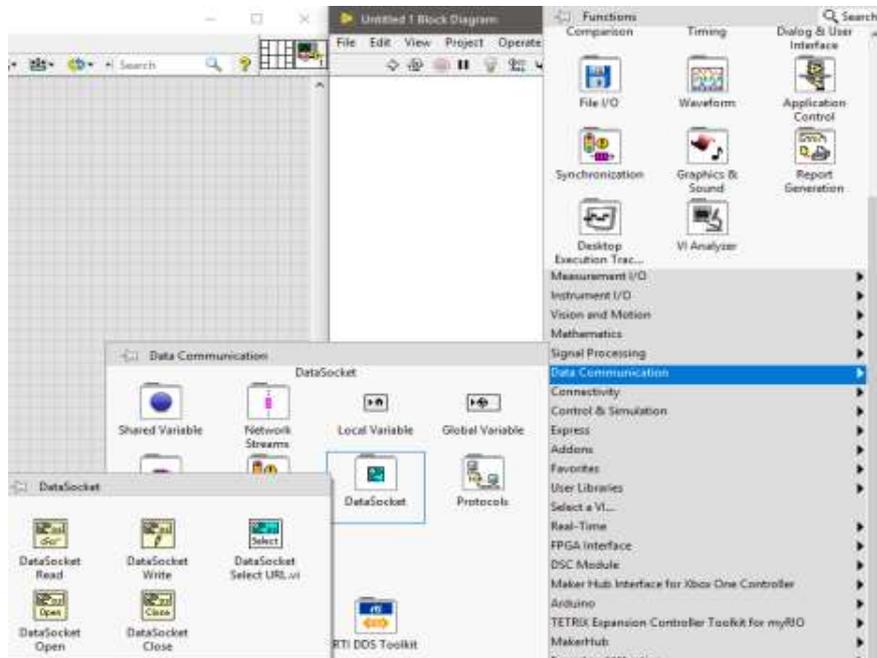


Figura 3. 15: Conector Datasocket en LabVIEW
Elaborado por: Autor.

Se procede a seleccionar la herramienta de Datasocket Open donde se configura la dirección donde se realizará la conexión entre las dos computadoras.

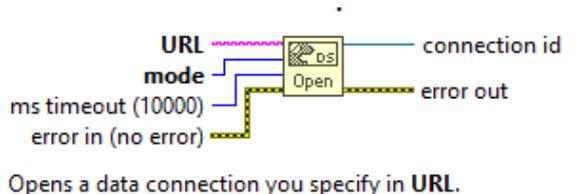


Figura 3. 16: Conector Datasocket en LabVIEW
Elaborado por: Autor.

Se procede a crear una dirección donde se almacenará el Datasocket y se transmitirá la información entre los dos VI con el nombre `dstp://localhost/información` configuramos el modo si es de escritura.

3.3.4. Creación de Switch en LabVIEW

Crear los Switch para el envío de datos desde un computador a otro para controlar el valor. Clic derecho en el panel de control y seleccionar Control>>Modernos>Push botón

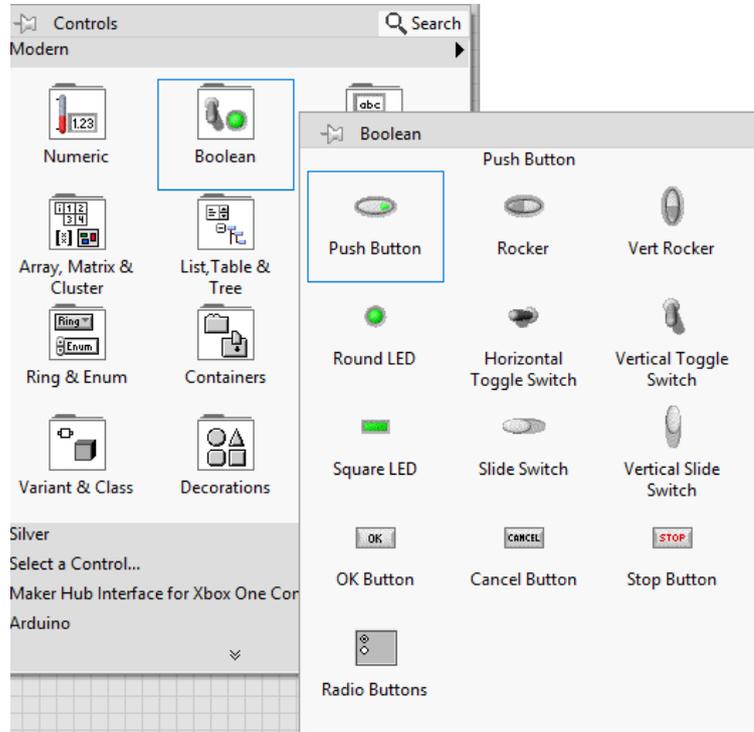


Figura 3. 17: Creación de Switch en LabVIEW
Elaborado por: Autor.

Como se muestra en la figura 3.18 se crearon 8 botones con el nombre botón 0 hasta botón 7.

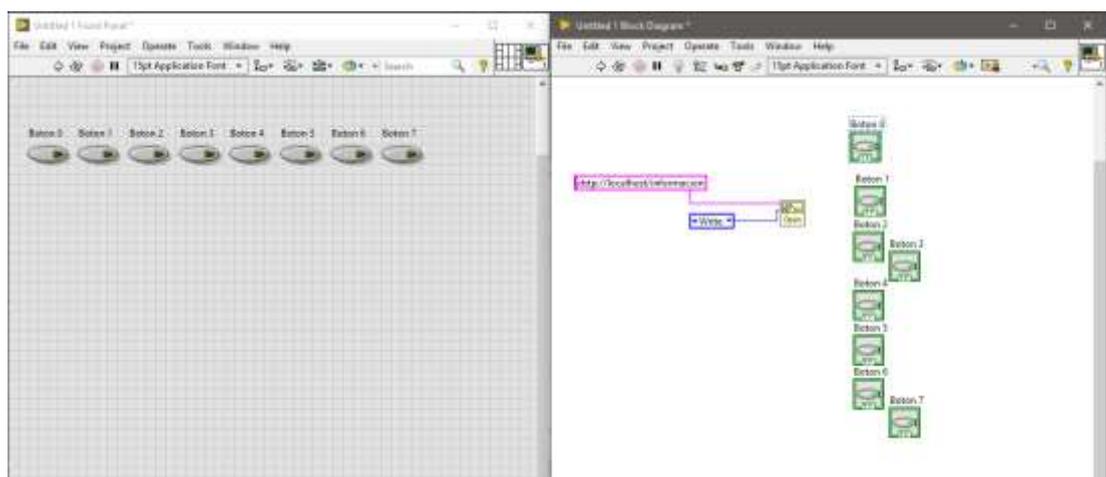


Figura 3. 18: Creación de Switch en LabVIEW
Elaborado por: Autor.

3.3.5. Creación de lazo While en LabVIEW para el Receptor

Se procede a crear un bucle While y utilizar la herramienta de lectura del Datasocket en esta herramienta es necesario saber qué tipo de dato se recibirá al computador en este caso se crea ocho datos booleanos convertidos en una misma señal.

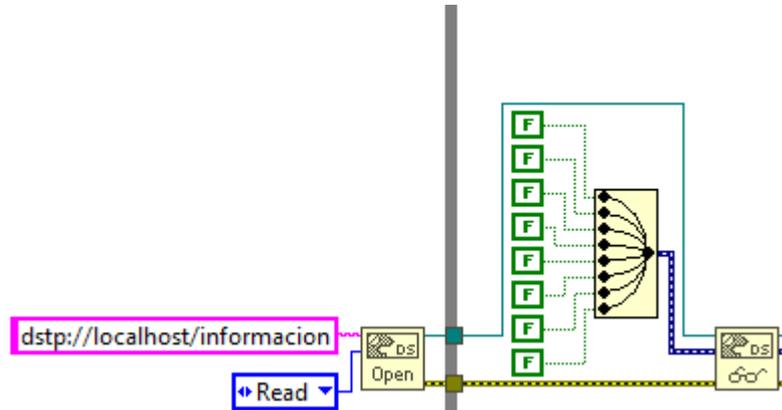


Figura 3. 19: Creación de lazo While en LabVIEW para el Receptor
Elaborado por: Autor.

3.3.6. Demultiplexación de información enviada por Emisor.

Al momento de recibir el dato es importante volver a demultiplexar la señal y convertirlas en datos únicos para esto se creará también indicadores donde se visualizará la recepción de datos.

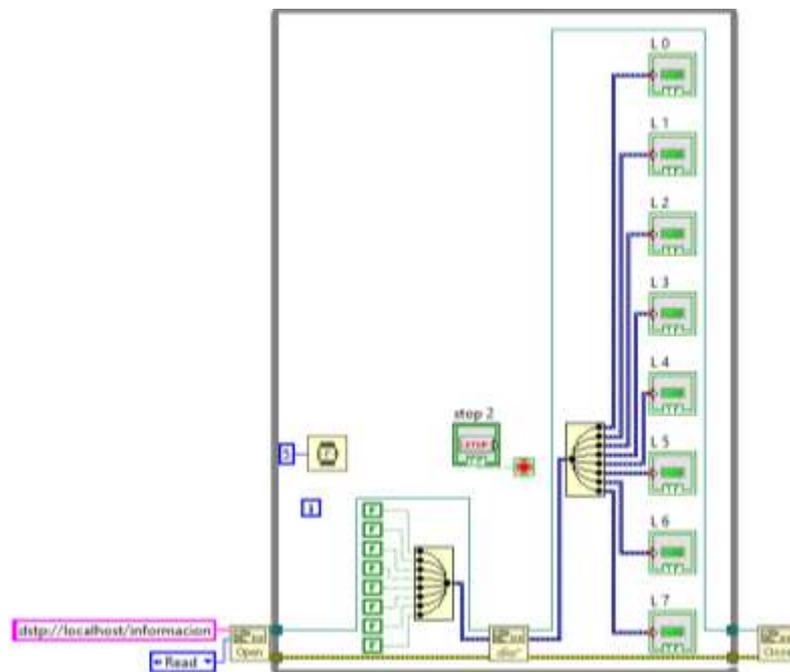


Figura 3. 20: Demultiplexación de información enviada por Emisor
Elaborado por: Autor

3.4. Activación de Datasocket en Emisor y Receptor

Se procede a activar el servidor Datasocket en ambas máquinas para poder visualizar la conexión establecida entre las dos computadoras.

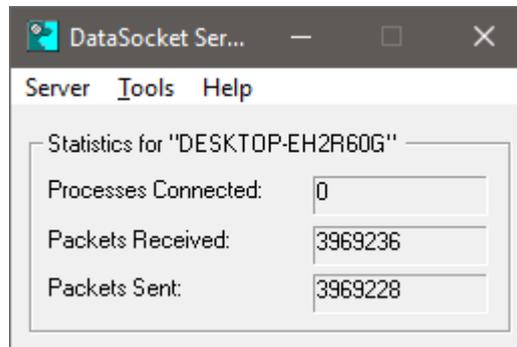


Figura 3. 21: Activación de Datasocket en Emisor y Receptor.
Elaborado por: Autor

3.5. Conexión de tarjeta de adquisición de datos DAQ.

En el receptor se procede a conectar el dispositivo de adquisición y generación de datos. En el bloque de diagrama seleccionar Measurement i/o >> DaQmx >> Daq Asist.



Figura 3. 22: Conexión de tarjeta de adquisición de datos DAQ
Elaborado por: Autor

3.6. Configuración de tarjeta DAQ

Se procede a crear el recurso para enviar el bit de salida a la tarjeta de adquisición de datos, tal como se muestra en la figura 3.23.

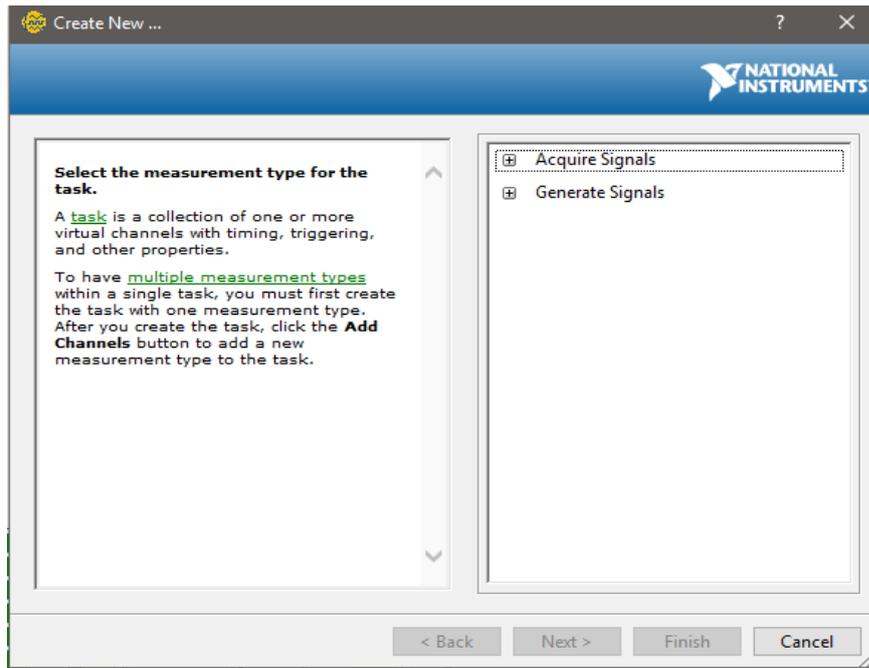


Figura 3. 23: Configuración de tarjeta DAQ.
Elaborado por: Autor

En la figura 3.24 se observa el procedimiento para seleccionar un dato digital de salida.

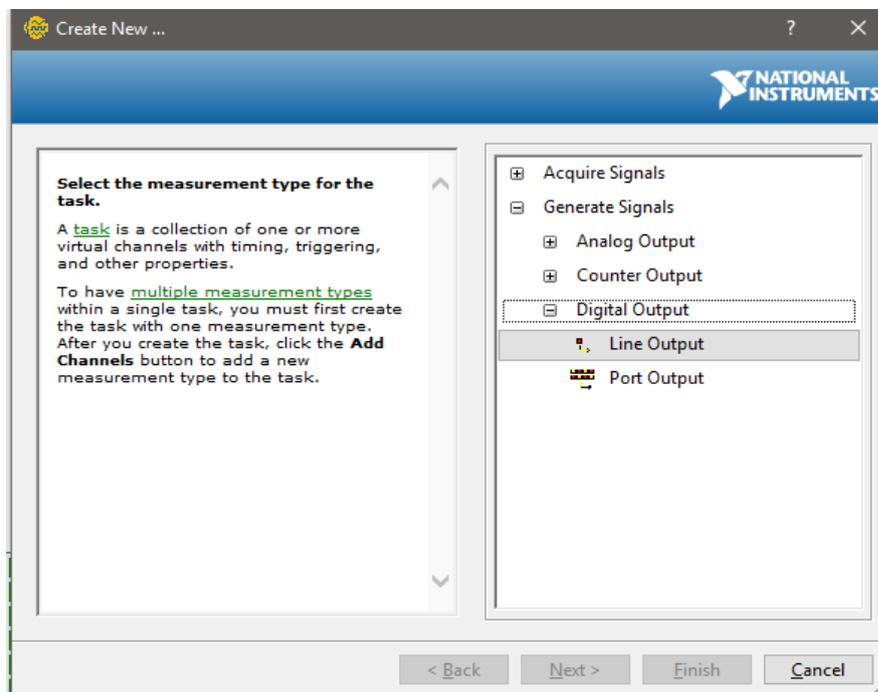


Figura 3. 24: Configuración de tarjeta DAQ.
Elaborado por: Autor

En la figura 3.25 se observa el procedimiento para seleccionar los bits de salida

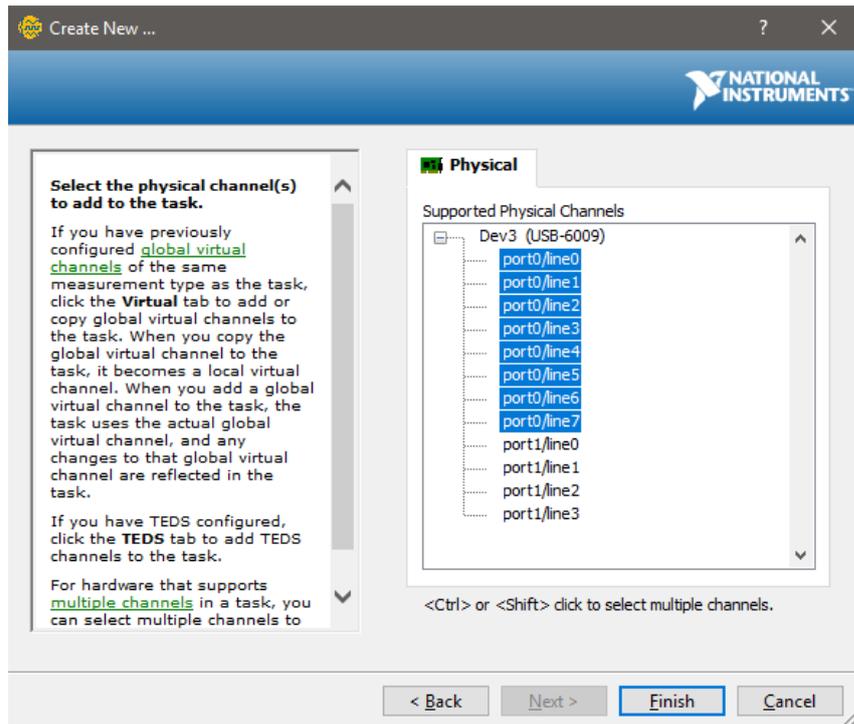


Figura 3. 25: Configuración de tarjeta DAQ.
Elaborado por: Autor

La figura 3.26 muestra la configuración de la lógica de los bits digitales de salida.

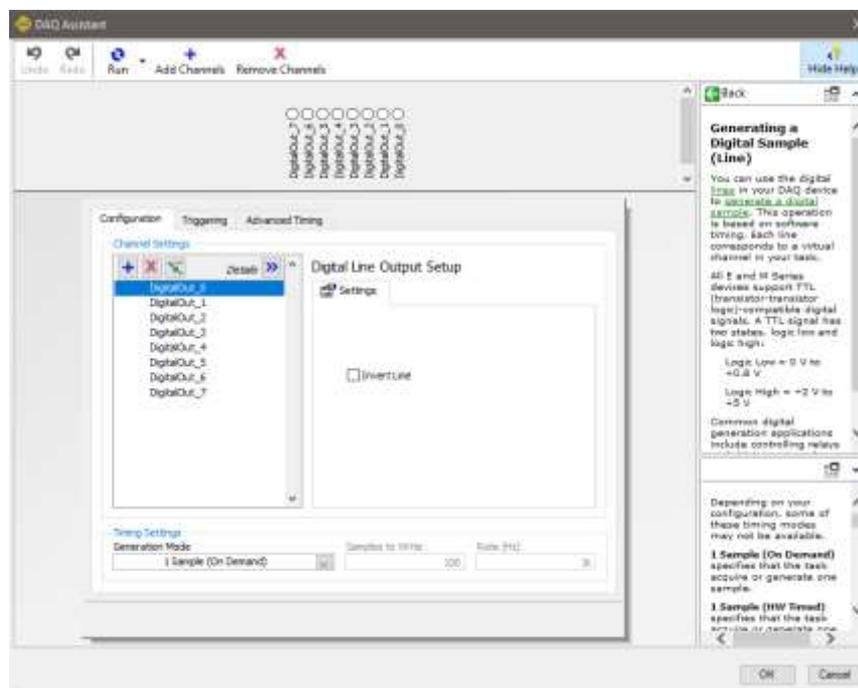


Figura 3. 26: Configuración de tarjeta DAQ.
Elaborado por: Autor

3.7. Implementación de tarjeta DAQ a programación de receptor.

Una vez configurada la tarjeta DAQ se procede a conectar el pin que se recibe del Datasocket al pin de la DAQ.

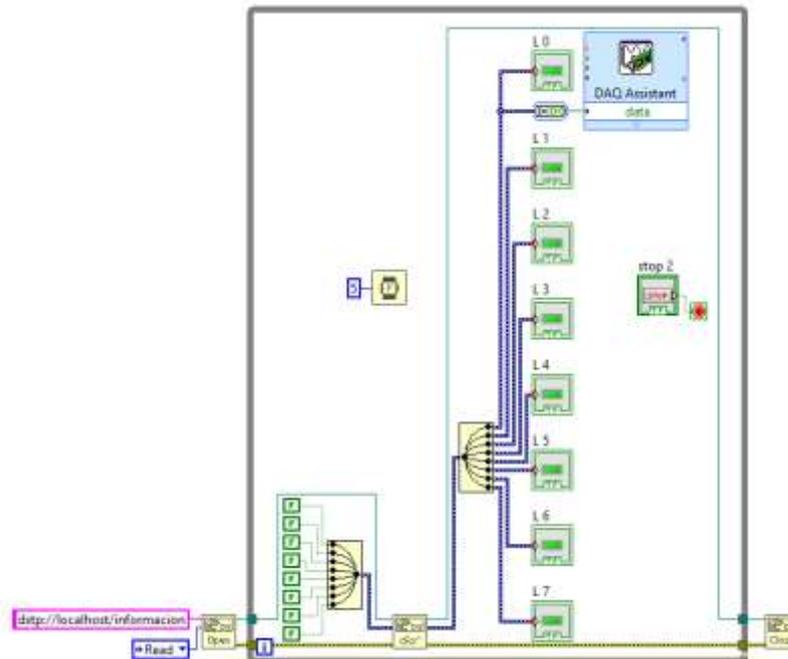


Figura 3. 27: Implementación de tarjeta DAQ a programación de receptor.

Elaborado por: Autor

Luego se procede a conectar cada pin con un canal de la DAQ.

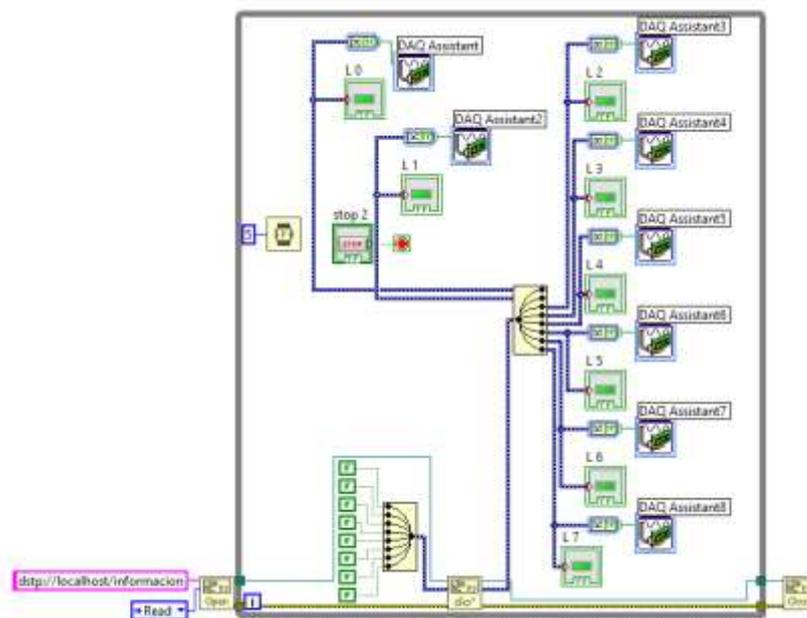


Figura 3. 28: Implementación de tarjeta DAQ a programación de receptor.

Elaborado por: Autor

3.8. Configuración de entrada analógica a la programación.

En el VI llamado receptor se procede a agregar las variables que enviara el receptor.

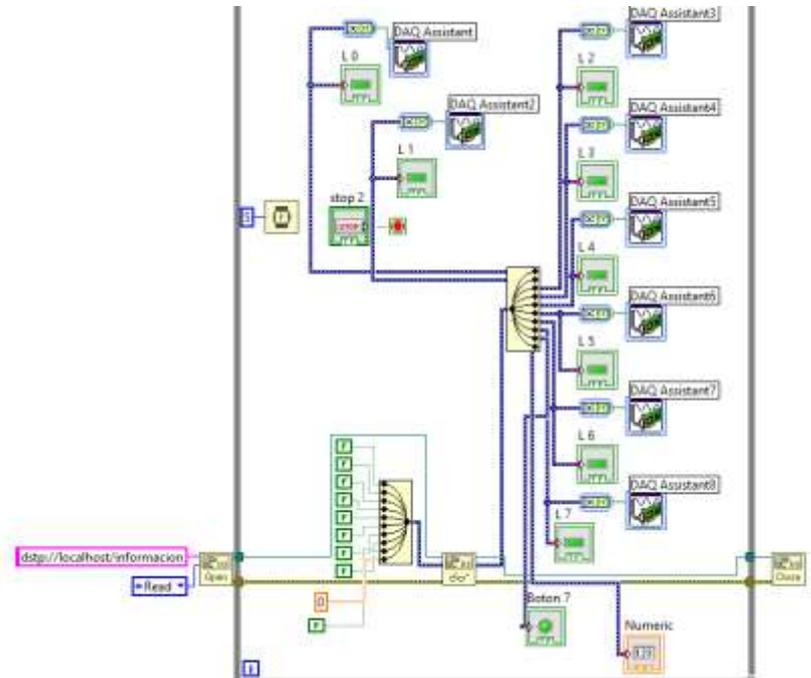


Figura 3. 29: Configuración de entra analógica a la programación.
Elaborado por: Autor

Se procede a crear una lógica con la condición de cuando este en alto se mostrará el valor de la tarjeta de adquisición de datos mediante la conversión analógica a digital y cuando este en falso un valor simulando la señal de forma virtual. La figura 3.30 muestra cómo crear la entrada analógica.

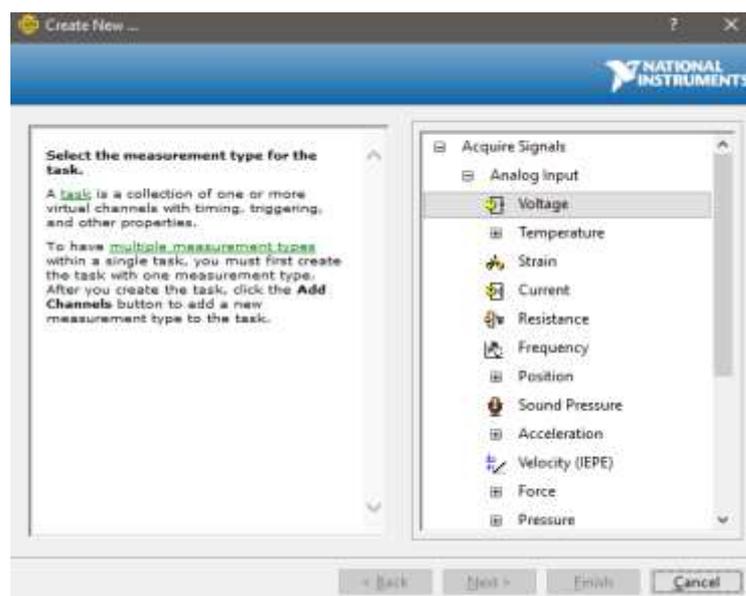


Figura 3. 30: Configuración de entra analógica a la programación.
Elaborado por: Autor



Figura 3. 33: Armado de circuito con tarjeta DAQ.
Elaborado por: Autor

Conclusiones.

- Se realizó la demostración de una simulación en el programa LabView usando el sistema SCADA, para así proceder a realizar un diseño para fines didácticos.

- En base del estudio del uso de LabVIEW en conjunto con la tarjeta DAQ, se realiza un diseño e implementación de una aplicación enfocada en una rutina de aprendizaje usando un Datasocket, así llegando a permitir manipular el envío y recepción de datos de manera analógica a digital y viceversa, por medio de la tarjeta DAQ.

- A partir de la investigación realizada en el presente trabajo se concluye que el uso de herramientas de apoyo como LabVIEW, tarjeta DAQ y Datasocket va dirigido para los estudiantes como para los docentes para la carrera de Ingeniería en Telecomunicaciones, ya que en el trabajo presentado se validó que al usar simuladores y equipos físicos como la tarjeta DAQ nos permiten captar y comprender de manera visual y práctica lo aprendido en el salón de clase y poderlo aplicar en el laboratorio de Telecomunicaciones de la Universidad Católica de Santiago de Guayaquil.

Recomendaciones.

- Se recomienda tener en cuenta el uso de herramientas de apoyo como la tarjeta DAQ ya que permite realizar varias funciones con un solo dispositivo
- Realizar programación en diferentes simuladores como Matlab donde nos permita utilizar varias funciones de la tarjeta DAQ para diferentes proyectos.
- Se recomienda tener en cuenta las versiones del simulador donde se realizó la programación para que no haya conflicto al momento de ejecutar el programa.
- Se recomienda antes de ejecutar el programa en los dos equipos revisar la IP a la cual se están conectando para así poder establecer su conexión sin inconvenientes.

Glosario de Términos

SCADA: Supervisa, controla y adquisición de datos.

ICS: Sistema de control Industrial.

PLC: Controladores lógicos Programables

DSC: Sistemas de control distribuido

LAN: Red de área local.

TI: Tecnología de la información.

RTU: Unidad terminal remota.

EMC: Compatibilidad electromagnética.

DNP: Protocolo de red distribuido.

FED: Controlador Frontal.

FEP: Procesador Frontal.

ODBC: Conectividad de base de datos abierta.

OLE DB: Vinculación e incrustación de objetos para bases de datos

IED: Dispositivos electrónicos inteligentes.

CPU: Unidad Central de procesamiento.

WAN: Redes de área amplia.

REFERENCIAS BIBLOGRAFIAS

- Aghenta, L. O., & Iqbal, M. T. (2019). Development of an IoT Based Open Source SCADA System for PV System Monitoring. *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, 1–4. <https://doi.org/10.1109/CCECE.2019.8861827>
- Arghira, N., Hossu, D., Fagarasan, I., Iliescu, S., & Costianu, D. (2011). Modern SCADA philosophy in power system operation—A survey. *U.P.B. Sci. Bull*, 73(2).
- Fovino, I. N. (2014). SCADA System Cyber Security. En K. Markantonakis & K. Mayes (Eds.), *Secure Smart Embedded Devices, Platforms and Applications* (pp. 451–471). Springer. https://doi.org/10.1007/978-1-4614-7915-4_20
- Fraser, R. E. (2001). *Process measurement and control: Introduction to sensors, communication, adjustment, and control*. Prentice Hall.
- Greeff, G., & Ghoshal, R. (Eds.). (2004). System hierarchies and components. En *Practical E-Manufacturing and Supply Chain Management* (pp. 26–65). Elsevier. <https://doi.org/10.1016/B978-075066272-7/50006-3>
- Johnson, L. (2016). Security Component Fundamentals for Assessment. En *Security Controls Evaluation, Testing, and Assessment Handbook* (pp. 531–627). Elsevier. <https://doi.org/10.1016/B978-0-12-802324-2.00011-7>
- Manso, O. (2016). Security Analysis of the Electronic Management System for a Total Site Utility System. En *Computer Aided Chemical Engineering* (Vol. 38, pp. 37–42). Elsevier. <https://doi.org/10.1016/B978-0-444-63428-3.50011-4>
- Mehta, B. R., & Reddy, Y. J. (2015). Distributed control system. En *Industrial Process Automation Systems* (pp. 75–133). Elsevier. <https://doi.org/10.1016/B978-0-12-800939-0.00006-1>

- Portal, E.-E. E. (2018, abril 25). Five Terms You MUST Be Familiar With: SCADA, DCS, PLC, RTU and Smart Instrument | EEP. *EEP - Electrical Engineering Portal*. <https://electrical-engineering-portal.com/scada-dcs-plc-rtu-smart-instrument>
- Riyan, W. (2016, septiembre 25). Read n Write: SCADA Host Software. *Read n Write*. <http://riyansblog.blogspot.com/2016/09/scada-host-software.html>
- Siguenza Riofrio, J. R. (2019). *Diseño e implementación de aplicación práctica de un control SCADA para controles eléctricos usando LabVIEW y DAQ*. [Trabajo de Titulación, Universidad Católica de Santiago de Guayaquil]. <http://repositorio.ucsg.edu.ec/handle/3317/13387>
- Solé-Torres, Duran-Ros, Arbat, Pujol, Ramírez de Cartagena, & Puig-Bargués. (2019). Assessment of Field Water Uniformity Distribution in a Microirrigation System using a SCADA System. *Water*, 11(7), 1346. <https://doi.org/10.3390/w11071346>
- Sommestad, T., Ericsson, G. N., & Nordlander, J. (2010). *SCADA System Cyber Security—A Comparison of Standards*. 5590215. <http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-32242>
- Tandel, M., Joshi, U., Golhani, A., Tandel, M., Joshi, U., & Golhani, A. (2017). Scripting engine for SCADA HMI. *2017 2nd International Conference for Convergence in Technology (I2CT)*, 492–496. <https://doi.org/10.1109/I2CT.2017.8226178>
- Thomas, M. S., & McDonald, J. D. (2017). *Power System SCADA and Smart Grids*. https://nls.lids.org.uk/welcome.html?ark:/81055/vdc_100052741861.0x000001
- Toker, O., Cheded, L., & Fan, R. (2005). Internet-based SCADA: A new approach using Java and XML. *Computing and Control Engineering*, 16(5), 22–26. <https://doi.org/10.1049/cce:20050504>



DECLARACIÓN Y AUTORIZACIÓN

Yo, **Rodríguez Ortiz, Cristian Eduardo** con C.C: # 095031407-0 autor del Trabajo de Titulación: **Diseño e implementación de un algoritmo de control para sistemas SCADA mediante una tarjeta de adquisición de datos aplicando Datasocket con fines didácticos** previo a la obtención del título de **INGENIERO EN TELECOMUNICACIONES** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 2 de marzo del 2020

f. _____

Nombre: Rodríguez Ortiz, Cristian Eduardo

C.C: 095031407-0



Presidencia
de la República
del Ecuador



Plan Nacional
de Ciencia, Tecnología,
Innovación y Saberes



SENESCYT
Secretaría Nacional de Educación Superior,
Ciencia, Tecnología e Innovación

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TÍTULO Y SUBTÍTULO:	Diseño e implementación de un algoritmo de control para sistemas SCADA mediante una tarjeta de adquisición de datos aplicando Datasocket con fines didácticos.	
AUTOR(ES)	Rodríguez Ortiz, Cristian Eduardo	
REVISOR(ES)/TUTOR(ES)	M. Sc. Zamora Cedeño, Néstor Armando	
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil	
FACULTAD:	Facultad de Educación Técnica para el Desarrollo	
CARRERA:	Ingeniería en Telecomunicaciones	
TÍTULO OBTENIDO:	Ingeniero en Telecomunicaciones	
FECHA DE PUBLICACIÓN:	2 de Marzo del 2020	No. DE PÁGINAS: 68
ÁREAS TEMÁTICAS:	Sistemas de Comunicaciones, Instrumentación Virtual.	
PALABRAS CLAVES/ KEYWORDS:	SCADA, LabVIEW, DAQ, Analógico, Digital, Datasocket	
RESUMEN/ABSTRACT:	<p>En el presente trabajo se diseña e implementa una programación de un algoritmo de control para sistemas SCADA en la cual va a permitir interactuar de manera analógica y digital los conocimientos adquiridos en el salón de clase sobre el uso del simulador LabVIEW y tarjetas de adquisición de datos, para esto se crea dos programaciones llamadas emisor y receptor, en el emisor va a tener la programación designada para poder enviar la información de las interacciones que tiene permitido modificar, el receptor va captar la información por medio del Datasocket y poder enviar la información al circuito armado el cual va a estar conectado por medio de la tarjeta DAQ, en el que se valida tanto por medio de programación y por el circuito armado la información enviada por el emisor. La programación creada en el presente trabajo va a servir como material de apoyo para la formación de profesionales en la carrera Ingeniería en Telecomunicaciones.</p>	
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO
CONTACTO CON AUTOR/ES:	Teléfono: +593-989393394	E-mail: reduardo-06@hotmail.com
CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE	Nombre: Palacios Meléndez, Edwin Fernando	
	Teléfono: +593-9-67608298	
	E-mail: edwin.palacios@cu.ucsg.edu.ec	
SECCIÓN PARA USO DE BIBLIOTECA		
No. DE REGISTRO (en base a datos):		
No. DE CLASIFICACIÓN:		
DIRECCIÓN URL (tesis en la web):		