

**UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA**

**Carrera de Ingeniería Electrónica**



**Desarrollo de un panel para la reconfiguración  
del cableado en un módulo para control y  
supervisión**

Tesis para optar el título profesional de Ingeniero Electrónico

**Giancarlo Patiño Ortega**

**Código 201210160**

**Asesor**

Victor Murray

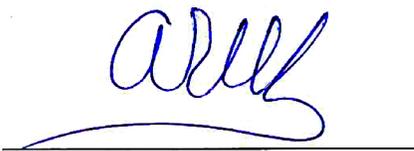
Lima - Perú

Abril 2019

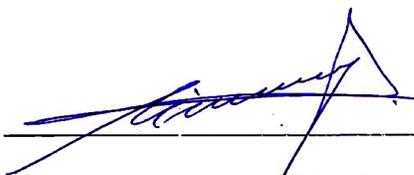
La tesis

**Desarrollo de un panel para la reconfiguración del cableado en un módulo para control y supervisión**

ha sido aprobada.

A handwritten signature in blue ink, appearing to read 'Arturo', written above a horizontal line.

Arturo Rojas

A handwritten signature in blue ink, appearing to read 'Victor Murray', written above a horizontal line.

Victor Murray

A handwritten signature in blue ink, appearing to read 'Sergio', written above a horizontal line.

Sergio Aranda

A handwritten signature in blue ink, appearing to read 'Elmer Ramirez', written above a horizontal line.

Elmer Ramirez

*Dedicatoria:*

*Con mucho cariño y en reciprocidad a todo el apoyo que he recibido de mis padres  
Geovanna y Edwin. Para ustedes el logro y sueño cumplido.*

*Agradecimientos:*

*A Dios por colocar cada situación en el momento preciso.*

*A mis padres y hermana quienes han sido el más grande apoyo durante esta etapa,  
alentando cada paso que doy.*

*A mi familia y amigos quienes de distintas formas obraron a bien de alcanzar una meta  
tan importante en la vida.*

*A mi asesor Victor Murray y profesores quienes compartieron conocimiento y  
experiencia vital en el desarrollo de mi vida profesional.*

# Índice general

	Pág.
<b>RESUMEN</b> . . . . .	<b>1</b>
<b>CAPÍTULO 1 INTRODUCCIÓN</b>	<b>2</b>
<b>CAPÍTULO 2 ANTECEDENTES</b>	<b>6</b>
2.1 Transmisores eléctricos . . . . .	7
2.1.1 Transmisores analógicos . . . . .	8
2.1.2 Transmisores digitales . . . . .	8
2.1.3 Transmisores híbridos . . . . .	8
2.1.4 Esquemas de conexión de los transmisores eléctricos . . . . .	9
2.2 Teoría de grafos . . . . .	11
2.2.1 Estructura de datos en la representación de grafos . . . . .	11
2.2.1.1 Estructura de lista . . . . .	11
2.2.1.2 Estructura matricial . . . . .	12
2.2.1.3 Breadth-first Search . . . . .	12
2.3 Arquitectura reconfigurable . . . . .	14
2.3.1 Dominios de aplicación . . . . .	14
2.3.2 Mecanismos de conmutación . . . . .	15
2.3.3 Cableado multipanel . . . . .	15
<b>CAPÍTULO 3 MÉTODOS</b>	<b>17</b>
3.1 Modelo de conexiones . . . . .	18

3.1.1	Análisis de señales de conexión . . . . .	18
3.1.1.1	Señales analógicas . . . . .	19
3.1.1.2	Señales digitales . . . . .	19
3.1.2	Definición del modelo de conexiones . . . . .	20
3.2	Hardware . . . . .	23
3.2.1	<i>Field Programmable Gate Array</i> . . . . .	24
3.2.2	Conexiones analógicas . . . . .	25
3.2.3	Conexiones digitales . . . . .	26
3.2.4	Esquema general de conexión . . . . .	27
3.3	Software . . . . .	28
3.4	Algoritmo implementado . . . . .	30
3.4.1	Camino más corto usando Grafos - algoritmo Dijkstra . . . . .	31
3.5	Comunicación entre computadora y FPGA . . . . .	36
<b>CAPÍTULO 4 RESULTADOS Y DISCUSIONES</b>		<b>38</b>
4.1	Implementación del Software . . . . .	38
4.1.1	Algoritmo Dijkstra . . . . .	39
4.1.2	Interfaz gráfica de usuario . . . . .	41
4.2	Implementación del Hardware . . . . .	43
4.3	Conexión con el módulo industrial para control y supervisión . . . . .	44
<b>CONCLUSIONES</b> . . . . .		<b>51</b>
<b>ANEXOS</b> . . . . .		<b>53</b>

# Índice de tablas

3.1	Matriz de conexión analógica general, se muestra las posibles conexiones entre las entradas analógicas indicando con una 'x' una conexión hacia conexiones de un controlador. . . . .	22
3.2	Bloque de conexión para cuatro canales analógicos basado en la Tabla 3.1.	23
3.3	Bloque de conexión reducido para cuatro canales analógicos. . . . .	23
3.4	Identificadores de cada node en el diagrama de Grafo. . . . .	34

# Índice de figuras

2.1	Conexión de un transmisor de dos hilos, el circuito en serie alimenta y a su vez comunica el instrumento con el receptor. . . . .	9
2.2	Conexión de un instrumento de tres hilos, una fuente externa alimenta el dispositivo pero comparte la referencia a tierra con el receptor. . . . .	10
2.3	Conexión de un instrumento de cuatro hilos, la alimentación está separada del protocolo de comunicación por corriente. . . . .	10
2.4	Descripción gráfica de la operación de Primera Búsqueda Amplia en un diagrama de Grafos [1]. (a) Diagrama de Grafos inicial con un nodo inicio y siete nodos camino. (b) Primera iteración que encuentra los nodos adyacentes. (c, d) Segunda iteración que encuentra los siguientes nodos adyacentes. (e, f, g) Tercera iteración en búsqueda del siguiente grupo de nodos adyacentes. (h, i) Última búsqueda para completar el diagrama. . .	13
3.1	Panel de conexión reconfigurable. Se muestra con cuadros verdes la ubicación de los bloques de conexión para señales analógicas. . . . .	21
3.2	FPGA elegido para la implementación en hardware. . . . .	25
3.3	Circuito para controlar una señal I/O analógica usando un relé controlado. . . . .	26
3.4	Circuito para controlar una señal I/O digital usando optoacopladores. . . . .	26
3.5	Diagrama de bloques para cada módulo de conexión, tanto señales de control del bloque de conexión y las conexiones al panel. . . . .	27
3.6	Conexión genera de la computadora hacia las tarjetas. (a) Computadora con software de controlador instalado. (b) Convertidor serial a I <sup>2</sup> C. (c) tarjetas necesarias para la conexión completa del módulo. . . . .	28

3.7	Interfaz gráfica de usuario GUI, replica exacta del módulo donde una caja marcada simboliza un bloque de conexión que se activa con los marcadores que presenta al lado de cada uno de ellas. . . . .	30
3.8	Se muestra un nodo fuente y uno destino, que tienen sistemas de nodos a manera diagrama de Grafos de dos caminos. . . . .	32
3.9	Diagrama de operación del algoritmo de Dijkstra implementado en los boques de conexión. . . . .	33
3.10	Flujo para la conexión y configuración de la tarjeta desde el software controlador. . . . .	37
4.1	Algoritmo Dijkstra encuentra el camino más corto entre el nodo origen 62 al nodo destino 40. . . . .	39
4.2	Algoritmo Dijkstra encuentra el camino más corto entre el nodo origen 62 al nodo destino 53. . . . .	40
4.3	Conexión analógica de un instrumento de dos hilos. . . . .	41
4.4	Conexión analógica de un instrumento de cuatro hilos. . . . .	42
4.5	Advertencia de conexión inválida no existente. . . . .	43
4.6	Bloque de conexión de señales analógicas implementado en hardware. . .	44
4.7	Comparación entre el módulo industrial y la integración con el bloque de conexión. . . . .	45
4.8	Conexión de tres transmisores hacia tres entradas analógicas desde la GUI.	46
4.9	Monitoreo de señales de entrada y salidas analógicas en el software Studio 5000. . . . .	47
4.10	Conexión válida errónea, la cual conecta un transmisor en dos puertos distintos. . . . .	50

## **RESUMEN**

Un panel modular es implementado para automatizar el proceso de recableado de un módulo industrial para control y supervisión. Su función principal es la configuración inicial de las conexiones necesarias entre instrumentos ubicados en una planta, equipos para implementar un proceso industrial a escala y controladores lógicos programables. El planteamiento se enfoca en la reconfiguración de hardware controlado por software a través de la interfaz gráfica de usuario diseñada para la tesis, con base en el algoritmo de Dijkstra para encontrar las conexiones a través de la representación de las señales analógicas y digitales en un grafo. Tanto el diseño del hardware como del software son partes esenciales en el desarrollo de la tesis, ambas se realizan a la medida del laboratorio de automatización y control en la UTEC, para finalmente conectar el bloque de modular de conexión con el panel del módulo industrial para control y supervisión y ejecutar una aplicación sencilla que demostró el correcto direccionamiento de las señales analógicas.

# Capítulo 1

## INTRODUCCIÓN

Ingeniería Electrónica en la Universidad de Ingeniería y Tecnología (UTEC) presenta la infraestructura a la vanguardia necesaria para la enseñanza y entrenamiento de los estudiantes en sistemas de control y automatización industriales, por ejemplo con el equipamiento del “Laboratorio de Automatización”. Este laboratorio cuenta con réplicas a escala de plantas industriales y módulos de supervisión industrial que permiten implementar redes industriales, sistemas SCADA completos, instrumentación industrial y sistemas avanzados de control<sup>1</sup>. De esta forma, el estudiante puede experimentar teórica como experimentalmente cada situación en la cual está envuelto un ingeniero encargado de la automatización de un proceso industrial real [2]. Actualmente el módulo industrial para control y supervisión se encuentra bajo proceso de patente [3] al igual que la planta industrial que posee los instrumentos como transmisores y actuadores necesarios [4].

Para hacer uso del laboratorio de automatización como ambiente de enseñanza se realizan dos pasos: Primero se requiere configurar los controladores lógicos programables (PLC por sus siglas en inglés) con el programa que el usuario diseña a través de su software de configuración interna, en este caso Studio 5000: el segundo paso a seguir es el cableado físico de los transmisores y actuadores a usar con las entradas o salidas respectivas de los PLCs o periferia descentralizada. El cableado tiene como objetivo transportar señales analógicas y digitales a través de cables sencillos punto a punto, este proceso de conexión en la industria no es recomendable o no se usa, pero por fines didácticos de laboratorio se realiza de acuerdo a criterios específicos de conexión según la estructura eléctrica interna de cada transmisor o actuador [5, 6].

---

<sup>1</sup>Programa Avanzado en Ingeniería del Diseño y Gestión de Proyectos de Automatización, julio 2017 <https://youtu.be/LZB3tkqKQ4c>. Última revisión 27-06-2018.

Cuando se empieza a usar el módulo para control y supervisión, durante las primeras sesiones o cursos de introducción al control y automatización, el usuario está sujeto al aprendizaje de buenas prácticas como el correcto uso del equipo en general y medidas de seguridad. Seguido de funciones básicas, conexiones, configuración de los equipos y administración de redes. Ya en cursos más avanzados la conexión de equipos, carga de programas y configuración de red pasan a segundo plano siendo tareas tácitas secundarias. De los tres procesos secundarios mencionados la conexión de equipos aun es un proceso manual, lo que significa invertir tiempo en recablear si en caso se requieren realizar dos configuraciones de módulo distintas, la manipulación constante de cables y si se quiere aprovechar la característica remota del software de los PLC se necesita la presencia de una persona que se encargue del cableado para el módulo para control y supervisión en específico.

La solución obvia radica en la puesta en marcha del módulo con todas las conexiones posibles de transmisores y actuadores manteniéndose todas fijas, con lo cual se pierde el dinamismo del módulo en poder conectar de distintas formas la planta con los PLCs, por lo tanto el fin didáctico desaparece por completo. Desde el punto de vista de la seguridad del equipo, no es apropiado tener dicha configuración ya que un programa con un proceso erróneo puede causar daños en los equipos.

La tesis plantea la implementación de un sistema para automatizar el proceso de conexión de los equipos involucrados entre sus entradas y salidas, para ello se desarrolla un panel modular de cableado reconfigurable controlado por software. El diseño del panel físico tiene como base el modelo de conexiones necesarias que se obtiene del análisis y síntesis de las señales analógicas y digitales. A partir del diseño se construye una tarjeta impresa que implementa en hardware el modelo analizado. A su vez se desarrolla un controlador por software para que una computadora pueda manejar las conexiones a voluntad del usuario. Tanto el módulo para control y supervisión como el panel se integran para tener finalmente el sistema completo de control y automatización con cableado

reconfigurable.

Las conexiones realizadas para cada proyecto implementado en el módulo para control y supervisión responden a distintas configuraciones de conexión, siguiendo criterios de acuerdo al protocolo o tipo de comunicación de los dispositivos en planta. Análogamente así como la evolución de los sistemas de control de la lógica fija a lógica reprogramable, las conexiones cableadas toman otro rumbo al permitir su adaptabilidad, versatilidad y automatización para realizar una configuración rápida de las conexiones, lo cual al mismo tiempo reduce la frecuencia de manipulación de cables y borneras.

El prototipo implementado cumple con conectar correctamente señales analógicas de los instrumentos en planta con los controladores, reconfigurándose de acuerdo al diseño planteado por un usuario a través de una interfaz gráfica (GUI) desde donde se administran las conexiones físicas virtualizadas. La comunicación entre el panel modular y la computadora es una conexión local USB.

El módulo de supervisión y control cuenta con tres controladores lógicos programables y un controlador de periferia descentralizada a modo de expansión de entradas y salidas en general, con un total de 28 entradas/salidas analógicas y 92 entradas/salidas digitales [3, 4]. A su vez, el módulo de instrumentación posee 17 instrumentos entre transmisores y actuadores, con un total de 15 entradas/salidas analógicas y 17 entradas/salidas digitales.

Para las conexiones analógicas hay alrededor de 800 posibles configuraciones, por ello se limita el número de conexiones analógicas con bloques de conexión de cuatro canales sin afectar la flexibilidad de conectar cualquier instrumento a cualquier controlador en simultáneo. El modelo de conexiones planteado busca que todos los transmisores y actuadores se comuniquen con los controladores a través del canal que esté disponible sin interrumpir la operación de otro canal ya conectado.

Dado que es una prueba de concepto, para demostrar el planteamiento de la tesis los métodos aplicados se enfocan en realizar todas las posibles conexiones para los cuatro canales analógicos de salida o entrada con un único bloque de conexión.

La búsqueda de información se realiza con una vigencia no mayor a 15 años usando como palabras clave: reconfiguración de hardware, paneles adaptativos, bloques modulares y enrutamiento de señales. La información describe los objetivos, métodos usados y conclusiones en común entre los documentos investigados y el planteamiento de la tesis en conjunto con sus requerimientos. Los resultados obtenidos en el desarrollo de este trabajo de tesis han logrado ser publicados en la IEEE [7].

## Capítulo 2

### ANTECEDENTES

La realización de un panel de conexiones reconfigurable está basado en la lógica del hardware y el controlador necesario para que una computadora pueda comunicarse con dicho sistema y reconfigurarlo a través de código generado por el usuario. Cabe resaltar que los modos de conexión en el caso del hardware requieren cierto tipo de criterio de acuerdo al tipo de instrumento, las señales analógicas y digitales tienen una conexión directa en la forma como se conectan con el modelo de conexiones basado en la teoría de grafos. Por lo que también se debe parametrizar el tipo de algoritmo que ruteará todas las señales a través de la conmutación de *switches* en los paneles modulares reconfigurables.

El planteamiento para realizar un modelo de conexiones a través de redes se plantea en [8] basándose en tres tipos de topología: Mmalla de árboles, árbol de mallas y mallas Manhattan, se define teóricamente como realizar conexiones físicas para las topologías combinando mallas y árboles. Utilizando la teoría de grafos el sistema encuentra todos los posibles caminos que debe seguir una señal para conectarse de un punto a otro sin interferir en el camino de otra señal en la misma red de nodos. De la publicación se concluye que existe similitud en las propiedades de cada una de las redes por lo que son lo suficientemente inclusivas para montarlas en una red simple.

A nivel de implementación, en el laboratorio de investigación de la Fuerza Aérea de los Estados Unidos se desarrolló un sistema de cableado programable. El trabajo de investigación publicado [9] se enfoca en los conceptos requeridos para la implementación práctica que integre el sistema de cableado programable en cuestión, también se realiza la discusión sobre los retos que implican desarrollar el sistema que tangibilice los principios de interconexión de señales y su optimización a nivel de hardware utilizado. Su análisis

a profundidad de las consideraciones prácticas contempla dominios de la aplicación de los sistemas de cableado programable, qué tipos de señales puede y no puede conectar considerando condiciones ideales. Mecanismos de conmutación, *switches* o en síntesis relés de acuerdo al tipo de señales a transmitir y su calidad de conexión considerando el tamaño, forma y principio de funcionamiento de cada uno (escalas en milímetros y microsistemas [10]) haciendo referencia a que tan fiable es la conexión entre un punto A hasta un punto B usando 3 clases de conexión: cable fijo, cable dinámico [11, 12] y sistemas microelectromecánicos que transmiten paquetes enteros. Las tres versiones del hardware integrador exploran la adaptabilidad de switches microelectromecánicos en una estructura de cableado dentro de un panel, la arquitectura de conexiones entre nodos y la demostración en implementación real directa en paneles diseñados y fabricados como unidades de celdas con posibilidad de conectar señales digitales, alimentación, analógicas y microondas.

Para entender como realizar las conexiones en el panel propuesto, es requisito caracterizar los distintos tipos de señales que se usaron, tal cual las investigaciones [8, 9] diferencian la naturaleza de las señales ocupan una línea de comunicación a la vez, en este caso tanto las señales analógicas como digitales se comunican bajo un protocolo establecido según sus principios de conexión los que se traducen en transductores y transmisores eléctricos.

## **2.1 Transmisores eléctricos**

Llamados también transmisores electrónicos, su principio de funcionamiento está basado en la conversión de una variable física medible a una señal eléctrica, actuando como detectores de inductancia, transformadores diferenciales o resistencia variable (Puente de Wheatstone) [5] con una exactitud de 0,5 %.

### **2.1.1 Transmisores analógicos**

Transmiten el valor de una variable física que fue convertida a una señal eléctrica de acuerdo a diversos principios de funcionamiento del propio transmisor y del sistema elegido. Cada transmisor analógico puede generar la señal eléctrica debido a un cambio en la resistencia de los materiales que la componen. Un estándar para este tipo de conexión son las señales 4-20 mA que hacen referencia al valor 0 y 100 % respectivamente de la señal. Este estándar se usa debido a que su respuesta en tiempo es cinco veces menor en milisegundos respecto a un protocolo de comunicación sofisticado. La escala se configura en el dispositivo y en el controlador a usar, la gran mayoría de dispositivos tiene una escala determinada en unidades a medir respecto a la corriente generada por cada uno de ellos.

### **2.1.2 Transmisores digitales**

Debido al tipo de variable que se lee en estos transmisores, no requieren una conversión compleja a comparación de las señales analógicas, basta un umbral o condición específica para que un transmisor pueda enviar un cero o uno lógico. Con el avance de la tecnología la sofisticación ha incrementado a gran escala, permitiendo también usar este tipo de transmisión para enviar tramas de datos, trenes de impulsos que permiten enviar cierto tipo de información codificada al controlador, el cual es capaz de leer y descifrar el mensaje, desde el más simple como contar el número de impulsos digitales entrantes, hasta encontrar un patrón de bits con un dato embebido.

### **2.1.3 Transmisores híbridos**

HART es un tipo de protocolo punto a punto más usado en la industria debido a su aceptación por fabricantes del rubro, posicionándolo como el más comercial dentro de la automatización industrial [13] . El principio de funcionamiento está basado en

la modulación en frecuencia de señales digitales, donde la onda de señal analógica de 4-20mA se usa como una señal portadora y otra señal de +/- 0.5mA modulada en frecuencia lleva la información digital del transmisor al controlador usando dos frecuencias 1200Hz para representar un 1 lógico y 2200Hz para un 0 lógico. Ya que no requiere una reinstalación de componentes en hardware es el preferido para evolucionar una red entera hacia un protocolo más robusto que transmita mucha más información sobre el mismo canal de comunicación. Otros tipos de transmisores híbridos son de propiedad de fabricantes específicos, por lo que no es un estandar al momento de montar una red de control automático.

#### 2.1.4 Esquemas de conexión de los transmisores eléctricos

Conectar los dispositivos transmisores con las entradas y salidas de los controladores y periféricos responde a criterios eléctricos de alimentación y lectura/escritura de datos [5]. Cada tipo de transmisor usa cierto número de hilos, descritos a continuación.

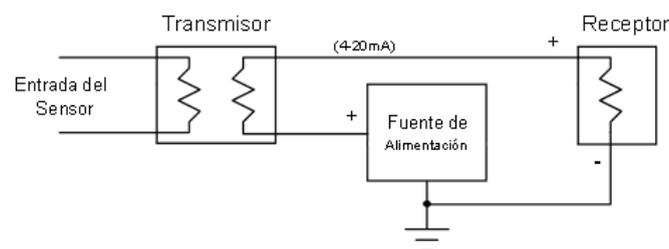


FIGURA 2.1: Conexión de un transmisor de dos hilos, el circuito en serie alimenta y a su vez comunica el instrumento con el receptor.

- Transmisor de dos hilos. Se conecta la fuente de alimentación directamente a la entrada del transmisor, su salida va conectada al controlador y se cierra el lazo llevando la conexión a la referencia en tierra como se observa en la figura 2.1. Su principal uso es la baja cantidad de potencia que necesita el transmisor para leer datos ya que la alimentación comparte espacio físico con la señal de salida, además del significativo ahorro de cables frente a otro dispositivo de tres o cuatro hilos.

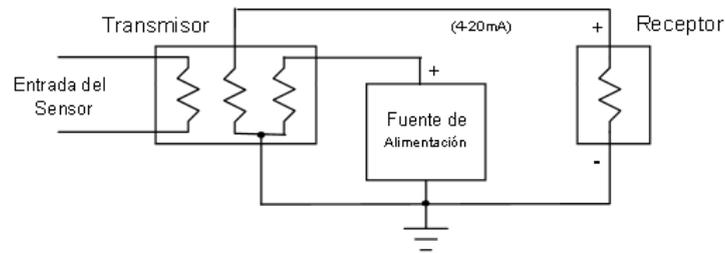


FIGURA 2.2: Conexión de un instrumento de tres hilos, una fuente externa alimenta el dispositivo pero comparte la referencia a tierra con el receptor.

- Transmisor de tres hilos. Cuando la potencia consumida por el transmisor no puede ser usada para la alimentación y la señal de salida, se utilizan tres hilos, donde uno alimenta el dispositivo, otro porta la señal de salida y el último cierra el lazo a tierra para ambas conexiones (ver figura 2.2).

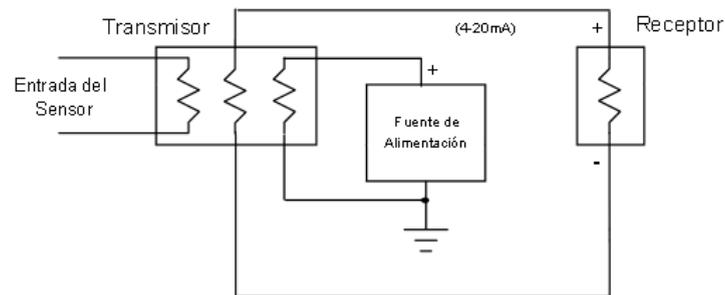


FIGURA 2.3: Conexión de un instrumento de cuatro hilos, la alimentación está separada del protocolo de comunicación por corriente.

- Transmisor de cuatro hilos. Para dispositivos que ya están completamente alimentados, se utiliza este tipo de transmisor, donde dos hilos son la alimentación, y los otros dos sobrantes se conectan directamente al circuito de transmisión de la señal. La diferencia entre los transmisores de tres y cuatro hilos es el aislamiento total entre la línea de alimentación y la de transmisión en un mismo dispositivo como se muestra en la figura 2.3.

## **2.2 Teoría de grafos**

T. Cormen presenta en su libro [6] algoritmos tanto para implementación en software como en hardware. La teoría de grafos es un método en el cual se representa un sistema con elementos llamados nodos y aristas. Sistemáticamente el buscar un nodo en el grafo significa atravesar cada arista del diagrama de grafos y haber visitado todos los nodos conectores. Por lo general su representación son líneas para las aristas y puntos de conexión para los nodos.

### **2.2.1 Estructura de datos en la representación de grafos**

Existen varias topologías y estrategias para buscar dos nodos y así poder optimizar el tiempo de búsqueda e implementación. A continuación se presenta los métodos más comunes y el que se usará como referencia.

#### **2.2.1.1. Estructura de lista**

- Lista de incidencia: Se representan con un vector de pares, donde cada par representa una arista del grafo.
- Lista de adyacencia: Visto desde cada nodo o vértice, cada uno tiene otros nodos adyacentes. Reduciendo de manera óptima el tiempo de búsqueda de un camino de un punto A a uno B. Se asocian los nodos del grafo en una lista que contenga todos los nodos adyacentes, terminando así como una lista de listas de incidencia para cada nodo dentro del grafo general.

### 2.2.1.2. Estructura matricial

- Matriz de incidencia: El grafo está representado por una matriz con una columna de aristas y la otra de nodos, donde se indica la conexión de los nodos a través de la arista.
- Matriz de adyacencia: El grafo es una matriz del tamaño de los nodos al cuadrado. Al igual que la lista, se conectan nodos y el valor de la arista define si están conectados o desconectados.

### 2.2.1.3. Breadth-first Search

Primera Búsqueda Amplia, es un algoritmo básico para buscar el arquetipo de un grafo. Otros algoritmos más complejos como el algoritmo de árbol de expansión mínimo de Prim o el algoritmo de camino corto simple de Dijkstra lo usan de referencia.

$$G = (V, E) \tag{2.1}$$

Se denomina como primera búsqueda amplia porque expande uniformemente la búsqueda de vértices, descubiertos y no descubiertos, a lo largo de toda la malla. Dado un grafo a partir de un nodo fuente "s" y (2.1).

Sistemáticamente se debe explorar las aristas "g" y cada vértice que se puede alcanzar desde el nodo "s", calculando la distancia y generando un "Breadth-first tree" (primer árbol de búsqueda) con raíz en el nodo "s".

En primera instancia se colocan como distancias infinitas con los nodos adyacentes al nodo "s" tal como se muestra en la figura 2.4a. La primera búsqueda ubica todos los nodos adyacentes al nodo "s" figura 2.4b. Luego recursivamente por cada nodo adyacente se repite el mismo proceso de ubicar nodos adyacentes no descubiertos como las figuras

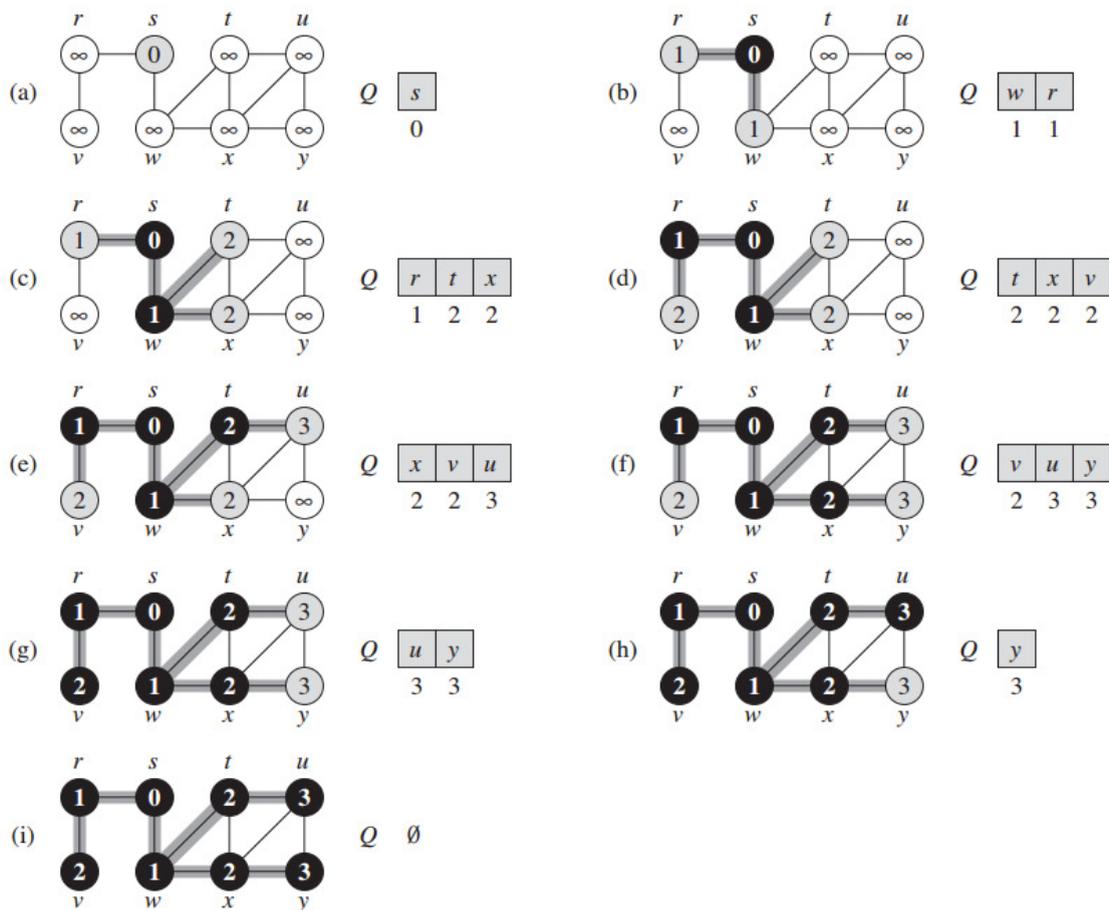


FIGURA 2.4: Descripción gráfica de la operación de Primera Búsqueda Amplia en un diagrama de Grafos [1]. (a) Diagrama de Grafos inicial con un nodo inicio y siete nodos camino. (b) Primera iteración que encuentra los nodos adyacentes. (c, d) Segunda iteración que encuentra los siguientes nodos adyacentes. (e, f, g) Tercera iteración en búsqueda del siguiente grupo de nodos adyacentes. (h, i) Última búsqueda para completar el diagrama.

2.4c-h hasta llegar al nodo final o destino figura 2.4i adyacentes al nodo adyacente “s” el cual es el nodo inicial.

Como se ve en la figura 2.4 la operación empieza con un vértice “s”, la primera iteración tiene como nodos adyacentes los vértices “w” y “r”. Luego de una segunda iteración de búsqueda se encuentran los nodos “t” y “x” como adyacentes al nodo 1 “w”, de la misma forma para el nodo “r”. La coloración varía de acuerdo a la iteración que se realiza siendo blanca como una arista no recorrida y un nodo no descubierto, gris

para un camino recorrido una sola vez y un nodo descubierto en la iteración inmediata y finalmente negro cuando una segunda iteración pasa por el mismo nodo que ya no tiene más adyacentes.

## **2.3 Arquitectura reconfigurable**

Todo sistema reconfigurable requiere una estructura tangible a nivel de hardware controlado por software, ya que por este medio se manejarán señales eléctricas de mayor potencia, que no deben mezclarse con señales de control de baja potencia. Para ello se usa una estructura que aisle ambos grupos de señales.

### **2.3.1 Dominios de aplicación**

Entre un punto A y otro B se espera una conexión eléctricamente perfecta para crear un sistema de cableado programable, cada conexión durante el diseño del sistema es eléctricamente perfecta al no tener resistencia o inductancia, cuando esta libre de acoplamientos o dispersión, es adaptable a distancias cortas y largas, todas estas razones refuerzan la idea que un modelo ideal no existe en el mundo real, para reducir errores debido a las conexiones, se debe mantener cada tipo de señal en su medio óptimo. Una señal en el dominio analógico debe considerar acoplaciones, atenuación, pérdidas de línea y dispersión. También se considera para este tipo de dominio una serie de subdominios como señales de baja intensidad y otras sujetas a amplificación, filtrado y procesos de retroalimentación que afectan directamente en el desempeño. Para el dominio digital que concierne la transmisión de voltaje discreto ha de considerarse las pérdidas de línea como el acoplamiento que se genere o pueda generar.

### **2.3.2 Mecanismos de conmutación**

El principio básico es un switch que conecta y desconecta señales, para conseguir un sistema adaptativo. De acuerdo al tipo de señales a enrutar y los dominios a cubrir. Desde relés electromecánicos hasta interruptores que usan principios de micro electro-mecánica (MEMS). Un relé puede conectar señales generando grandes conectores y espacio físico para contener todos los necesarios, más a pequeña escala el control es más complejo debido al tamaño de los actuadores, pero se ahorra en espacio y costo total.

### **2.3.3 Cableado multipanel**

A partir de la propuesta de modularidad entre bloques que implementen un sistema de cableado programable, reta al proyecto en respetar las conexiones que se realizan y no sobre escribir en ella. El paso posterior es buscar sistemas que implementen un enrutamiento de señales en módulos o bloques, por lo que existen sistemas de arquitecturas reconfigurables y que están en circulación comercial como los FPGA (Arreglo reconfigurable de compuertas, por sus siglas en Inglés) su arquitectura de conexión varía de acuerdo a la técnica usada por cada fabricante. Xilinx [12, 14] en sus modelos Spartan y Virtex utiliza bloques de conexión que sirven como relés lógicos conectando una señal en dos, tres o todas las direcciones posibles para enrutar los pines de conexión del FPGA al diseño realizado en la descripción de hardware de un proyecto a implementar.

La modularidad aplicable a la tesis es un concepto reciente, un ejemplo de implementación se dió durante el 2016, el proyecto ARA [15] de Google, planteó un modelo de celular con piezas intercambiables, donde cada puerto de conexión debía tener la capacidad de conectar genéricamente componentes al bloque central del dispositivo a través de líneas que transmitan datos y potencia. Dos años después Google dió por suspendido el proyecto debido a falta de enfoque de recursos en ese proyecto [16] del área ATAP [17]

(tecnología avanzada y proyectos, por sus siglas en inglés) por lo que se discontinuó el desarrollo del teléfono modular.

## Capítulo 3

### MÉTODOS

A partir del sistema de cableado reconfigurable [9] se analizó las posibles formas de conexión para un sistema de automatización con un panel de conexiones que consta de 16 entradas y 12 salidas analógicas para los controladores y 15 señales analógicas de entrada salida provenientes de los dispositivos en la planta de control. Tal es la cantidad de conexiones, que se pueden calcular hasta 1598 conexiones si se colocara un relé por cada conexión, lo que hace un sistema económicamente inviable de implementar. La primera posible solución se redujo a conectar solo uno de los controladores y las señales de campo, con lo que se obtuvo un total de 806 relés, aun siendo un diseño reducido no deja de ser un sistema inviable. Un segundo rediseño plantea la posibilidad de reducir aún más la malla hasta 266 relés, armando bloques genéricos con posibilidad de conexión a cuatro puertos, donde la restricción de conexión de una señal proveniente de campo permite enrutarse solo a tres puertos por bloque, a costa de parametrizar que solo ciertas señales se conecten a ciertos puertos mapeados. El mapeo de todas las señales a cablear para el módulo multipropósito, se subdivide en dos tipos de comunicación: Por un lado se tienen las señales analógicas con el principio de comunicación por variación de corriente entre 4 - 20 mA de acuerdo a dos criterios de conexión: dos y cuatro hilos. dentro del panel de conexiones existen en total 16 entradas analógicas o canales analógicos hacia los controladores, cada una tiene un par de puntos de conexión debido al circuito que requiere polo positivo y negativo por cada entrada, haciendo un total de 32 puntos en total. De la misma forma que las entradas, tiene doce salidas hacia afuera de los controladores, siguiendo el mismo principio de dos puntos de conexión por cada canal, haciendo un total de 24 conectores analógicos. Desde el punto de vista de la planta multipropósito se tienen en total diez salidas, la mitad de ellas son de cuatro hilos y las restantes de dos hilos, cinco

entradas analógicas con dos puntos de conexión por cada una para la polaridad en cada dispositivo.

Para las conexiones digitales, las salidas de los controladores suman en total 42. Las entradas hacia los controladores hacen un total de 46 puntos. Para completar todas las conexiones digitales, en planta se tienen 12 salidas y 5 entradas digitales. Cabe resaltar que cada conexión es independiente, por lo que los puntos de conexión coinciden con el número total de bornes de conexión ya sea entrada o salida.

### **3.1 Modelo de conexiones**

Establecer un modelo de conexión basado en el número de conexiones tanto analógicas como digitales permite conocer al sistema por completo, se empieza por el planteamiento de una conexión forzada de todas las entradas con todas las salidas sin excepción alguna, éste número de conexiones fija el máximo de conexiones correctas, conexiones sin sentido y conexiones inválidas. Debido a que no toda línea será correcta, el número se reduce a un modelo que describe todas las conexiones válidas, por lo cual se requiere hacer un análisis de cada tipo de señal según los transmisores presentes en la planta multipropósito de automatización y control.

#### **3.1.1 Análisis de señales de conexión**

Todas las señales que se generan de los transmisores tienen su propio principio de funcionamiento, sobre todo las señales analógicas que son continuas en el tiempo, se sabe que al codificar y digitalizarse, su manejo como dato binario permite su transporte a través de protocolos robustos de alta velocidad, pero la conversión requiere un alto costo computacional debido al hardware requerido para tal operación y además añade error numérico si en caso se necesitase volver a generar una señal continua a partir de los datos digitales de la misma.

### **3.1.1.1. Señales analógicas**

Según el principio de conexión de acuerdo al número de hilos, un transmisor analógico en planta con cuatro hilos usa un par para alimentación, el cual siempre está presente, y el otro par para la comunicación con el controlador a través de líneas o cables. Un transmisor de dos hilos funciona como un circuito en serie, utiliza un cable para su alimentación al polo positivo, otro para conectar el polo negativo con el polo positivo de la entrada del controlador y finalmente un último que cierra el circuito entre el polo negativo del controlador con la referencia a tierra, por lo tanto usa un total de tres líneas. Al expandir ambas configuraciones a un sistema automático, donde una línea conmuta entre conectada y desconectada, se requieren 32 líneas para conectar una salida de transmisor de cuatro hilos en planta hacia todas las posibles entradas de los PLCs y periferia descentralizada, lo que hace un total de 32 líneas x 5 transmisores = 160 líneas para una conexión total de las salidas de cuatro hilos de planta y las entradas a los controladores. En el caso de un transmisor de dos hilos usa 33 líneas para todas las posibles conexiones y 33 líneas x 5 transmisores = 165 líneas para conectar todos los transmisores de dos hilos a los PLCs y periferia descentralizada. En viceversa una de las salidas de los PLCs o periferia descentralizada usa un total de 10 líneas para conectarse a todos los actuadores, en total 10 líneas x 12 salidas = 120 líneas para una conexión total. Al sumar todos los datos del total de líneas calculadas, para conectar absolutamente todas las señales analógicas posibles, se requieren en total 445 líneas conmutables.

### **3.1.1.2. Señales digitales**

Debido a que las señales digitales son del tipo punto a punto, este facilita el diseño de un modelo de conexiones de la siguiente forma, una salida de planta tiene 46 líneas hacia la entrada de los controladores, haciendo un total de 46 líneas x 12 transmisores = 552 líneas para conexión total. Y en sentido contrario una salida de los controladores

puede conectar 5 líneas. Por lo tanto para conectar todas las entradas y salidas del módulo se requieren un total de 42 líneas x 5 salidas = 210 líneas conmutables. Un total de 762 líneas para una conexión digital absoluta.

### **3.1.2 Definición del modelo de conexiones**

Hasta este punto se consideran todos los casos de conexiones, significa implementar una estructura de 445 líneas conmutables, este escenario no es factible debido al tamaño de hardware que implica su implementación, además del alto costo de la electrónica que se ve envuelta. En base a la virtualización de varias redes o canales dentro de una sola red [8, 9] se plantea primero un modelo de cableado completo del sistema, para luego reducirlo a bloques modulares de conexión que cumplan la misma función de la conexión absoluta, con las líneas de conexión necesarias a fin de asegurar que todas o la mayoría de conexiones se puedan realizar y ninguna se sobreponga a otra a través de un mismo canal.

En el panel de conexiones del mismo módulo se pueden identificar las conexiones analógicas marcadas con cuadros de color verde en la figura 3.1. Tanto las entradas al controlador, como la periferia descentralizada tienen el mismo diseño para los canales de conexión, ocho puntos de conexión donde cada par de positivo y negativo es un canal de entrada hacia alguno de los controladores.

Para analizar el caso completo de las señales analógicas, se toma todos los canales de cualquier controlador, debido a que todas las líneas son bidireccionales no interesa si son entradas o salidas de planta a controladores o viceversa. Para explicar la situación actual, la tabla 3.1 representa las conexiones necesarias, de color rojo, en caso se quiera controlar absolutamente todas las líneas para poder conectar cualquier instrumento a cualquier canal analógico de cada controlador. Aquí se considera tanto la conexión en serie de instrumentos de dos hilos y conexión directa para aquellos instrumentos de cuatro hilos.

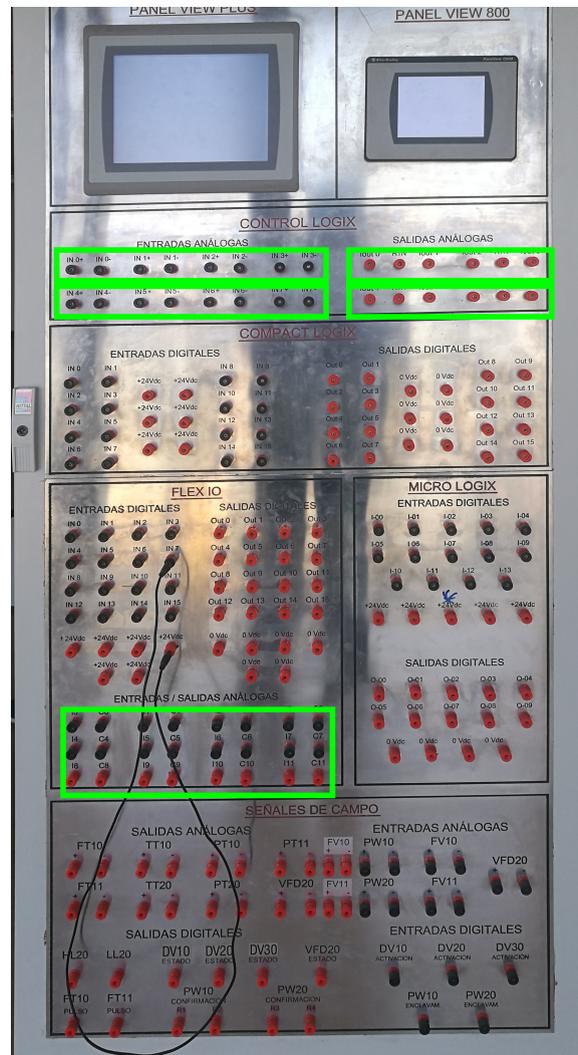


FIGURA 3.1: Panel de conexión reconfigurable. Se muestra con cuadros verdes la ubicación de los bloques de conexión para señales analógicas.

Para un solo bloque de conexión definido por solo cuatro canales (que responden como entrada o salida del módulo) se cuentan en total 69 conexiones calculadas a partir de la tabla 3.2 donde cada cuadro rojo representa una conexión o un cable, lógicamente este arreglo permite la conexión de cualquier instrumento a cualquier canal de entrada o salida disponible. Si por ejemplo se tienen ya cuatro instrumentos conectados a cada canal respectivamente, todas las demás conexiones no utilizadas quedan sobrando. Si se limitara a cada instrumento la conexión solo a tres posibles canales de cuatro, el número

TABLA 3.1: Matriz de conexión analógica general, se muestra las posibles conexiones entre las entradas analógicas indicando con una 'x' una conexión hacia conexiones de un controlador.

Controlador		Entradas analógicas																
		0+	0-	1+	1-	2+	2-	3+	3-	4+	4-	5+	5-	6+	6-	7+	7-	
Transmisores en planta	FT10	+	X		X		X		X		X		X		X		X	
		-		X		X		X		X		X		X		X		X
	FT11	+	X		X		X		X		X		X		X		X	
		-		X		X		X		X		X		X		X		X
	VFD20	+	X		X		X		X		X		X		X		X	
		-		X		X		X		X		X		X		X		X
	FV10	+	X		X		X		X		X		X		X		X	
		-		X		X		X		X		X		X		X		X
	FV11	+	X		X		X		X		X		X		X		X	
		-		X		X		X		X		X		X		X		X
	TT10	+																
		-	X		X		X		X		X		X		X		X	
	TT20	+																
		-	X		X		X		X		X		X		X		X	
	PT10	+																
		-	X		X		X		X		X		X		X		X	
	PT20	+																
		-	X		X		X		X		X		X		X		X	
	PT11	+																
		-	X		X		X		X		X		X		X		X	
0V			X		X		X		X		X		X		X		X	

de conexiones sobrantes se reduce pero a costa de perder también la posibilidad de conectar un instrumento a un canal. La máxima reducción sin afectar el funcionamiento del hardware se logra cuando cada instrumento se puede conectar a solo dos canales más la añadidura de una línea auxiliar múltiple que puede ser usada por alguno de los transmisores que no tienen conexión directa a dos canales. La tabla 3.3 muestra que la reducción de conexiones realizada permite aun conectar por completo cuatro instrumentos bajo las limitaciones de que un mismo instrumento tiene conexión directa a dos canales, solo un instrumento con principio de conexión de 2 hilos puede tomar posesión de la línea auxiliar en caso la conexión directa apunte a un puerto ya ocupado.

TABLA 3.2: Bloque de conexión para cuatro canales analógicos basado en la Tabla 3.1.

Controlador		Entradas analógicas								
		0+	0-	1+	1-	2+	2-	3+	3-	
Transmisores en planta	FT10	+	X		X		X		X	
		-		X		X		X		X
	FT11	+	X		X		X		X	
		-		X		X		X		X
	VFD20	+	X		X		X		X	
		-		X		X		X		X
	FV10	+	X		X		X		X	
		-		X		X		X		X
	FV11	+	X		X		X		X	
		-		X		X		X		X
	TT10	+								
		-	X		X		X		X	
	TT20	+								
		-	X		X		X		X	
	PT10	+								
		-	X		X		X		X	
	PT20	+								
		-	X		X		X		X	
PT11	+									
	-	X		X		X		X		
0V			X		X		X		X	

TABLA 3.3: Bloque de conexión reducido para cuatro canales analógicos.

Controlador		Entradas analógicas								
		0+	0-	1+	1-	2+	2-	3+	3-	multi
Transmisores en planta	FT10	+				X		X		
		-					X		X	
	FT11	+			X				X	
		-				X				X
	VFD20	+			X		X			
		-				X		X		
	FV10	+	X						X	
		-		X						X
	FV11	+	X				X			
		-		X				X		
	TT10	+								
		-	X		X					
	TT20	+							X	X
		-								
	PT10	+					X			X
		-								
	PT20	+					X			X
		-								
PT11	+									
	-	X							X	
Multi		X		X		X		X		
0V			X		X		X		X	

### 3.2 Hardware

Con la tabla 3.3 se define el número total de conexiones de entrada y salida necesarias, tanto para los controladores como los instrumentos de control, debido al tipo de señal con la que cada instrumento entrega o recibe información no se puede usar el mismo método para conectar una señal analógica como una señal digital. Para tal caso el modelo de conexiones se limita a usar 38 relés para conectar las señales analógicas y se usará un Arreglo de Compuertas Programables (FPGA por sus siglas en inglés) para el control de cada uno de los relés. En el caso de conexiones digitales se usará el mismo FPGA pero lo que se reconfigurará son las salidas y entradas para que puedan leer un valor binario y puedan escribirlo a la salida o viceversa de acuerdo a donde se conectan en el panel de conexiones del módulo para control y supervisión.

También el diseño de hardware involucra la conexión desde la salida de una computadora que ejecuta el software controlador, a través de una conexión serial que debe convertir los datos transmitidos al protocolo de circuito inter integrado (I<sup>2</sup>C por sus siglas en inglés) a través del cual se comunican los bloques de conexión modulares y cómo cada uno de los bloques está conectado con el módulo para control y supervisión tanto a las entradas y salidas de los controladores, como la conexión con los instrumentos ubicados en la planta para control y supervisión.

### **3.2.1 *Field Programmable Gate Array***

Con el modelo de conexiones definido se seleccionan los componentes a usar para cada bloque de conexión modular. La pieza fundamental en cada bloque es el FPGA Spartan3 s200 que controlará la activación independiente de 38 salidas, el FPGA fue elegido para este trabajo debido a que se tiene ya un esquema de conexión implementable y además posee 200 pines I/O. Cada una de estas 38 señales digitales accionan un relé, por lo que se usan 38 relés, el relé es activado por un electroimán interno que abre o cierra una línea de transmisión que es por donde pasaran cada una de las señales de 24V con una corriente variable entre 4mA y 20mA, el relé tiene en serie una resistencia de 110 Ohm para evitar que cuando cada terminal tenga un voltaje constante de alimentación que accione el electroimán este actúe como una línea de conexión sin resistencia y dañe directamente el FPGA.

Internamente el FPGA se divide en dos bloques principales: El primero es un módulo para comunicaciones I<sup>2</sup>C, es quien establece la comunicación entre la computadora que controla todo el hardware, es decir, todos los bloques de conexión conectados al mismo bus I<sup>2</sup>C. Cada bloque de conexión modular mantiene una dirección de esclavo fija predeterminada. El segundo bloque es el controlador de las salidas digitales, cada

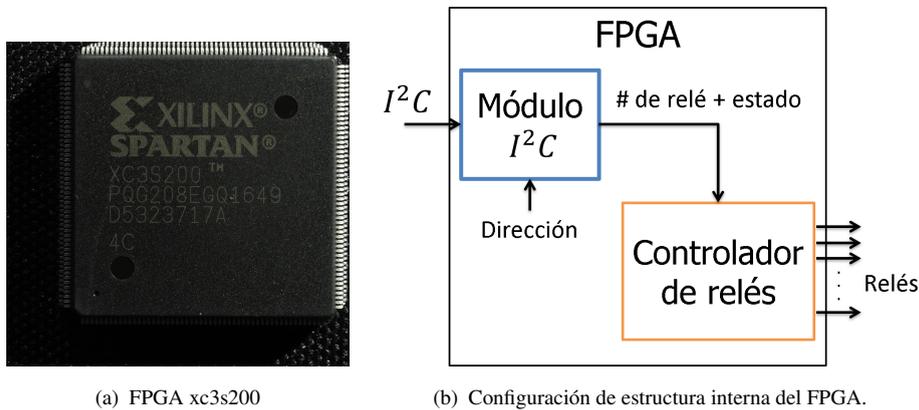


FIGURA 3.2: FPGA elegido para la implementación en hardware.

salida esta conectada a un relé por lo que cada vez que la computadora envía una actualización a través del protocolo I<sup>2</sup>C especificando que relé de los 38 debe cambiar de estado entre conectado o desconectado.

### 3.2.2 Conexiones analógicas

La señal continua analógica transporta información que se puede manipular de distintas formas, la opción de menos costo computacional es usar líneas conmutables a través de relés que a medida que se activen o desactiven generen un camino para que la señal pase de un punto a otro en una red. La otra opción es digitalizar el valor analógico a través de convertidores analógico digitales, lo cual permite manipular la información con más control y versatilidad a costa de hardware requerido y errores debido a como funcionan los convertidores.

Como primera versión se decide usar el sistema de conexión por relés, debido a que esta configuración permite la bidireccionalidad de las señales y no requiere hardware especial, tan solo evitar que en un nivel de voltaje constante el inductor interno del relé se convierta en un cable, la opción que se ajusta es colocar una resistencia en serie para que el circuito resistencia - inductor a lo largo del tiempo se convierta en solo una resistencia

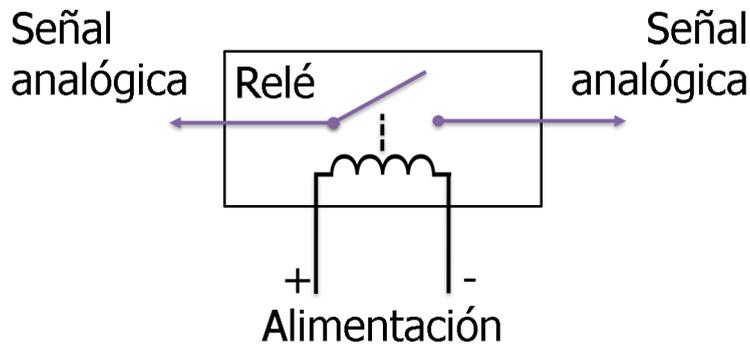


FIGURA 3.3: Circuito para controlar una señal I/O analógica usando un relé controlado.

para el FPGA. El modo de funcionamiento de cada relé es sencillo ya que se activa o desactiva solo si el software solicitó su conexión, la alimentación es quien controla cuales de los relés se conectan o no, ya que solo si están alimentados conducen, como se ve en la figura 3.3 el relé separa circuitos de potencia y control.

### 3.2.3 Conexiones digitales

Dada la simplicidad de los valores lógicos de una línea digital que se representa como un sistema binario, además el número de entradas y salidas, de los controladores y señales de planta, no supera el número de pines de un sistema embebido estandar, se puede considerar usar lógica integrada.

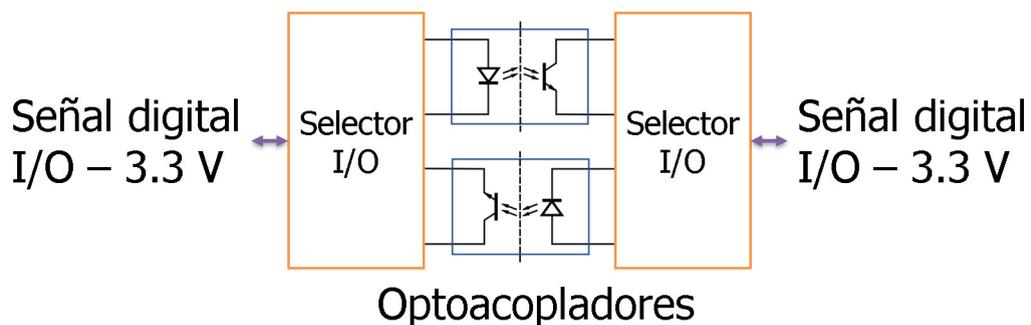


FIGURA 3.4: Circuito para controlar una señal I/O digital usando optoacopladores.

Aprovechando que se usa un FPGA con tablas de valores de verdad reconfigurables, solo basta leer cada entrada y reproducirla a una salida y viceversa. Para este proceso se propone el circuito de la figura 3.4 donde los selectores intercambian entre los estados de salida o entrada según sea el caso, los optoacopladores leen el dato bajo una configuración y escriben un dato según el valor de los selectores.

### 3.2.4 Esquema general de conexión

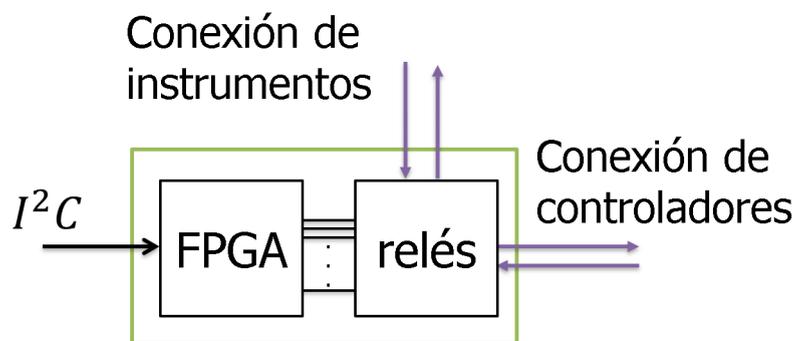


FIGURA 3.5: Diagrama de bloques para cada módulo de conexión, tanto señales de control del bloque de conexión y las conexiones al panel.

Se mencionó que cada tarjeta posee un FPGA X3CS200 y 38 relés, a esto se agregan los circuitos necesarios para la programación del FPGA y alimentación de toda la tarjeta. Visto este conjunto de elementos como un todo se convierte en un bloque de conexiones modular, cada bloque modular se conecta con el módulo para control y supervisión de la siguiente forma: todos los instrumentos están siempre conectados a todas las tarjetas ya que se debe mantener la posibilidad de conectar cualquier instrumento. Un bloque modular a parte de la conexión a instrumentos también tiene la conexión a cuatro canales analógicos por lo que se completa la posibilidad de poder conectar cualquier instrumento a cualquier controlador bajo la limitación que por cada bloque modular un instrumento se limita a conectarse solo a dos entradas. Como se observa en la figura 3.5 las conexiones externas hacia los instrumentos o hacia el módulo son bidireccionales, ya que la

señal no se procesa sino solo se direcciona, de esta forma se evita el uso innecesario de convertidores analógico digitales.

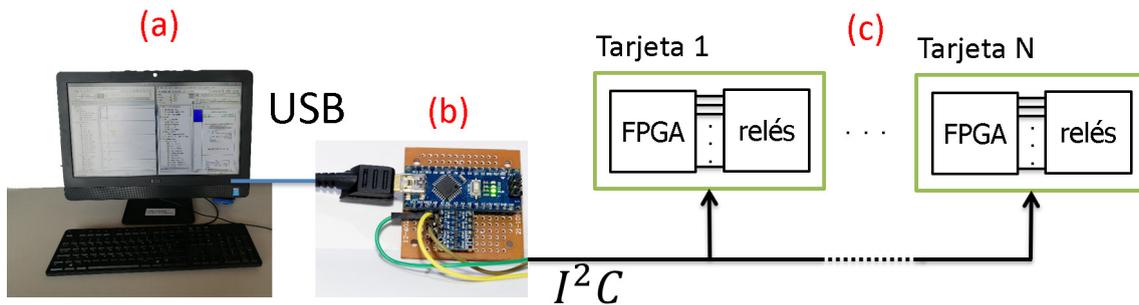


FIGURA 3.6: Conexión general de la computadora hacia las tarjetas. (a) Computadora con software de controlador instalado. (b) Convertidor serial a I<sup>2</sup>C. (c) tarjetas necesarias para la conexión completa del módulo.

Finalmente la conexión de todas las tarjetas montadas en el bus I<sup>2</sup>C se conectan a la computadora principal la cual ejecuta el software necesario para realizar las conexiones desde un programa diseñado específicamente para esta ampliación. La computadora envía toda la información a través de USB el cual es un protocolo serial, por lo que se usa una interfaz estándar que convierte el protocolo serial en protocolo I<sup>2</sup>C como se ve en la figura 3.6 en el ítem b. Cabe resaltar que la interfaz no toma parte en el procesamiento de la señal, ya que solo es un nexo para la interconexión de ambos protocolos.

### 3.3 Software

En paralelo se tiene la programación interna que asegura el correcto funcionamiento del hardware, el cual debe trabajar de manera óptima de acuerdo a las especificaciones requeridas. El controlador se divide en dos partes principales: la primera y vital es el algoritmo a usar, en este se tiene la lógica de conexión de cada uno de los instrumentos, para ello se utiliza la teoría de Grafos. Grafos implementa una red de nodos donde cada nodo tiene uno o más nodos de adyacencia, de esta red se determina un nodo fuente de donde se empezará a buscar camino hacia un nodo final; la segunda parte hace referencia

a como la computadora se comunica con cada una de las tarjetas para configurar cada una de las conexiones en cada bloque de conexión. Como sugiere la figura 3.6 se tiene una interfaz que convierte protocolos, ya que un es un sistema embebido estandar tan solo se configura parámetros generales de cada protocolo para que su conversión esté configurada de acuerdo a como cada uno de los dispositivos funciona.

Para esta tesis se implementó también una interfaz gráfica de usuario (GUI por sus siglas en inglés). Su función es replicar el panel de conexiones del módulo industrial para control y supervisión pero de una forma virtual, este proceso de conexión es parecido a la manera física de conectar los instrumentos con los controladores pero visto en una pantalla de computadora, incluso en la interfaz gráfica se deben considerar los principios básicos de conexión de acuerdo al tipo de instrumento que se tiene (número de hilos por dispositivo) ya que el algoritmo implementado limita las conexiones inválidas y no permite que se realicen para proteger los bloques de conexión modular como el equipo de automatización industrial.

La imagen 3.7 muestra la interfaz gráfica para conectar las señales analógicas del módulo industrial para control y supervisión, en ella el usuario conecta virtualmente los instrumentos a los controladores bajo los parámetros de cada bloque de conexión. El modo de funcionamiento del software es intuitivo desde que las conexiones se realizan tal cual se ejecutarían en el módulo real, para iniciar las conexiones, al menos una de las casillas de verificación, ver figura 3.7(a), por cada bloque de conexión debe estar activada, ésto define que hay un bloque conectado en esos puertos, el cual resalta con un cuadro de color magenta ver figura 3.7(b), luego se hace un clic en el punto donde se quiere empezar una conexión y se suelta el cursor donde se desea terminar la conexión, en caso se requiere hacer una desconexión el botón en la figura 3.7(c) desconectará todo el módulo industrial. Ésta interfaz tiene detrás el algoritmo, descrito a continuación, que relaciona las conexiones del módulo en un grafo, el cual permite o limita conexiones entre el módulo industrial para control y supervisión y los instrumentos de planta.

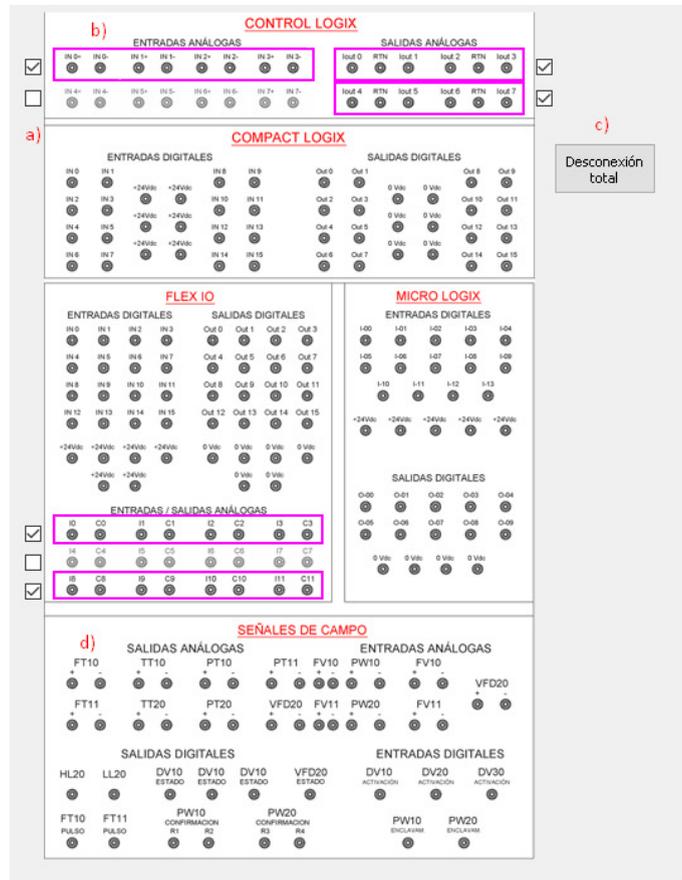


FIGURA 3.7: Interfaz gráfica de usuario GUI, replica exacta del módulo donde una caja marcada simboliza un bloque de conexión que se activa con los marcadores que presenta al lado de cada uno de ellas.

### 3.4 Algoritmo implementado

Aquí es donde se transforma el modelo de conexiones definido en un modelo de nodos y aristas conocido como diagrama de Grafos, para el modelo de conexiones de la presente tesis se diseñó el Grafo específico de acuerdo a la matriz 3.3 que ya tiene las conexiones reducidas. Para esta matriz se tienen la cantidad de nodos como de conexiones se tengan: primero los nodos evidentes los cuales vienen a ser cada uno de los puntos de conexión entre dos terminales (los cuadros en rojo) pero cada uno de estos nodos posee también aristas las cuales interconectan los nodos, por lo tanto toda entrada o salida tanto

de los puntos de conexión hacia los controladores o los instrumentos también cuentan como un nodo.

### **3.4.1 Camino más corto usando Grafos - algoritmo Dijkstra**

Para conocer el cómo se conectará una señal desde un nodo fuente hacia un nodo destino se establece un camino por el cual atravesará. El algoritmo elegido para hallar el camino más corto para esta aplicación es Dijkstra, debido a que su implementación involucra el uso de varias iteraciones donde se analiza nodo a nodo el camino por el que recorre la menor distancia. Una vez ubicados toda la lista de nodos a usar, se guardan los datos de los nodos involucrados en un arreglo. Como es lógico, cuando una señal ocupa uno o varios nodos en específico, hasta que no se desconecte esta señal, los nodos no pueden reusarse para conectar otras señales, por lo cual un nodo siempre tendrá dos estados: ocupado, si en caso ya esta siendo usado, y disponible, si es que este nodo no ha sido usado y está disponible para conectar otra señal.

Desde que la conexión entre un terminal de instrumento y uno de controlador tiene en la mayoría de casos solo un nodo intermedio, esto define el mínimo necesario para conectar una señal analógica. La figura cambia con los instrumentos de dos hilos, si el instrumento no tiene una conexión directa a uno de los canales que conecta al controlador, precisará usar la línea auxiliar la cual invierte más nodos, pero cumple el objetivo de poder conectar cualquier instrumento con cualquier controlador o viceversa.

Como ejemplo si se tienen dos caminos en una implementación, el algoritmo de Dijkstra determinará cual es el camino más corto, tomando de ejemplo los nodos de la figura 3.8 una misma señal tiene dos caminos para elegir, si tomar el nodo 1 o tomar el nodo 2 y 3, de acuerdo al algoritmo el número de saltos entre nodos para la primera opción es uno y para la segunda es dos, por lo tanto la respuesta es que se usará el nodo 1 para establecer la conexión desde el nodo fuente hasta el nodo destino. Este mismo ejemplo es el

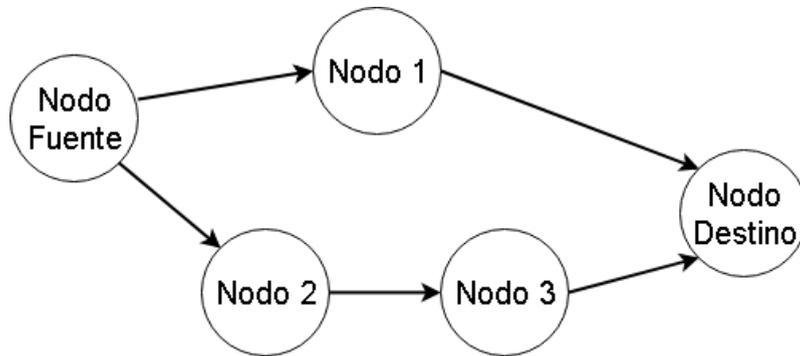


FIGURA 3.8: Se muestra un nodo fuente y uno destino, que tienen sistemas de nodos a manera diagrama de Grafos de dos caminos.

caso de las conexiones donde el nodo fuente y destino representan los puntos de conexión del panel de conexiones del módulo para control y supervisión y algún instrumento y los nodos intermedios son los puntos de conexión o relés implementados ya en la tarjeta.

Computacionalmente las aristas son adyacencias, entonces un Grafo que tiene tres nodos se representa por una matriz de 3x3. En el caso de las conexiones analógicas se tienen 69 nodos, por lo que la matriz de adyacencias es de 69x69 donde el valor 1 en la posición (M; N) representa que ambos nodos son adyacentes con una arista de peso unitario.

El algoritmo implementado busca en todos los nodos adyacentes hasta ubicar la menor distancia desde el nodo fuente hasta el nodo destino para finalmente devolver la lista de nodos que componen el camino más corto para que la señal sea transmitida a través de ellos. el diagrama de flujo de la figura 3.9 se repite el proceso de actualizar el nodo actual e ir al siguiente nodo más cercano hasta ubicar así todos los nodos que se usarán de camino. el máximo número de veces que se iterará depende directamente de la cantidad de nodos. En caso no se encuentre una conexión posible entre dos nodos el sistema simplemente termina la iteración y devuelve el valor nulo que indica que no es posible realizar dicha conexión. Por lo tanto el algoritmo Dijkstra con la matriz de adyacencia son quienes limitan las conexiones posibles, solo si existe una arista entre

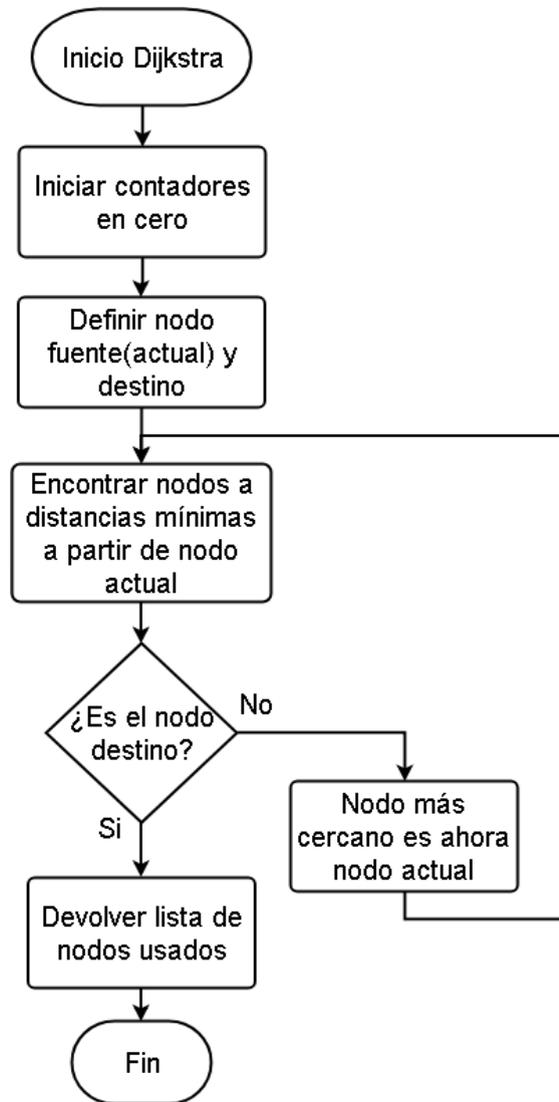


FIGURA 3.9: Diagrama de operación del algoritmo de Dijkstra implementado en los boques de conexión.

dos nodos pueden ser conectados, en caso contrario solo se advierte al usuario sobre una posible conexión errónea o inválida.

No es posible forzar conexiones, debido a que el algoritmo responde solo a la matriz de adyacencia y solo devuelve un resultado válido si la conexión cumple con los requisitos de criterio de conexión y existe la posibilidad dentro de la matriz. Esta es la limitación por software que portege al módulo para control y supervisión de fallas debido

a conexiones corruptas.

El núcleo del algoritmo es el grafo que relaciona entradas y salidas a través de nodos, a este grafo representado por una matriz de adyacencias se aplica el algoritmo Dijkstra para encontrar los caminos más cortos requeridos para realizar una conexión punto a punto. Cada nodo está identificado con un número único, dicho identificador es usado para saber que nodos se usa en una conexión, que nodos se conectan con otros y a partir de ello obtener el camino más corto como un listado de nodos usados.

TABLA 3.4: Identificadores de cada node en el diagrama de Grafo.

Control Logix			Entradas analógicas									
Nodos	Instrumento	Polaridad	60	61	62	63	64	65	66	67	68	
			0+	0-	1+	1-	2+	2-	3+	3-	multi	
Transmisores en planta	38	FT10	+					22		2		
	39		-					6		32		
	40	FT11	+			35				19		
	41		-			26				9		
	42	VFD20	+			5		29				
	43		-			16		12				
	44	FV10	+	37						34		
	45		-		11						8	
	46	FV11	+	31				21				
	47		-		15				1			
	48	TT10	+									
	49		-	28		25						
	50	TT20	+									
	51		-							18	10	
	52	PT10	+									
	53		-					36			24	
	54	PT20	+									
	55		-			20					7	
	56	PT11	+									
	57		-	4							33	
58	Multi		14		0		30		17			
59	0V			3		13		23		27		

A su vez cada nodo representa una conexión física en la tarjeta, como se observa en la tabla 3.4 cada punto de conexión tiene un nodo identificador asociado, los nodos del

0 al 37 marcados de amarillo en la tabla 3.4 son cada uno de los relés en un solo bloque de conexión. Los nodos del 38 al 57 representan la conexión hacia los instrumentos de la planta industrial. Los nodos del 60 al 67 son las conexiones hacia los canales analógicos de los controladores. Se incluyen también nodos especiales: los nodos 58 y 68 representan las líneas auxiliares requeridas para hacer un puente a otro canal analógico en caso un instrumento se quede sin conexión directa a un canal analógico y el nodo 59 es la conexión a la referencia a tierra para cerrar el circuito en serie a través del cual se conectan los instrumentos de dos hilos.

El modo de funcionamiento de la tabla es el siguiente: se toma como ejemplo de conexión válido la conexión del polo positivo del FT10, representado en el grafo como el nodo 38, con el polo positivo del canal analógico 2, nodo 64 en el grafo. El análisis empieza por el nodo de inicio, suponiendo que la fuente es el nodo 38 se analiza todos los posibles caminos que puede tomar, según su adyacencia los nodos más próximos son el 22 y el 2, ambos se guardan en un arreglo para pasar a la siguiente iteración. Ahora el análisis se hace en los nodos que ya se encontraron, el nodo 22 tiene adyacencia al 64 directamente, ya que es el nodo destino, el algoritmo se detiene y devuelve el camino seguido el cual es 38, 22, 64 y con esos valores ya se puede configurar el hardware. Cuando se tienen dos nodos que no tienen adyacencia alguna, significa que se realizó una conexión inválida por lo que se muestra un mensaje de advertencia y se evita la conexión porque es posible que la conexión no exista o si existe dañe el sistema. Otro caso particular es cuando un nodo ya ha sido ocupado para una conexión, en dicho caso el algoritmo Dijkstra no incluye el nodo en la lista de posibles nodos adyacentes para analizarlos en las iteraciones buscando la siguiente opción con otro nodo que esté libre.

Los casos especiales donde se usa la línea auxiliar que conecta los instrumentos de dos hilos se da cuando un nodo destino ya está ocupado por otro instrumento, ya que los instrumentos con este principio de conexión solo tienen un relé directo. Bajo la premisa mencionada, si por ejemplo se quiere conectar el transmisor de temperatura TT20

la primera opción por ser la distancia más corta es usar el nodo 18, pero si otro instrumento ya ocupa el canal analógico 3 entonces la distancia más corta será usar el camino con el nodo 51 que es la conexión del instrumento, el nodo 10 que conecta el instrumento con la línea auxiliar y para terminar la conexión alguno de los nodos de otros canales como son el 14, 0 o 30. Cabe resaltar que la línea auxiliar se ocupa con solo un instrumento a la vez.

### **3.5 Comunicación entre computadora y FPGA**

Luego que el algoritmo se ejecuta y puede validar que efectivamente la conexión es correcta, el paso siguiente es actualizar este valor o valores en la tarjeta que corresponde conmutando los relés necesarios. Ya que la computadora envía los datos ya procesados a través de un protocolo serial hacia la interfase, la programación de ésta última no se considera dentro del algoritmo, solo transmite datos mas no los modifica, por lo cual es ciega al sistema siendo solo una pieza convertidora. Del otro lado ya luego de la conversión a I<sup>2</sup>C la tarjeta con la dirección a la que se escribe recibe el dato y actualiza el nuevo estado del relé o los relés necesarios.

De manera general cada vez que se inicia el programa el serial abre la comunicación en el puerto serial donde está la interfase y a partir de este punto itera el número de veces que el usuario ejecuta una acción en la GUI. El diagrama de flujo de la figura 3.10 explica este proceso, cabe resaltar que Dijkstra es un proceso ya explicado, no hace parte de la comunicación pero es a partir de este dato que se realiza la validación de una conexión correcta en el marco de limitar las conexiones y proteger el módulo industrial para control y supervisión desde el alcance del software controlador.

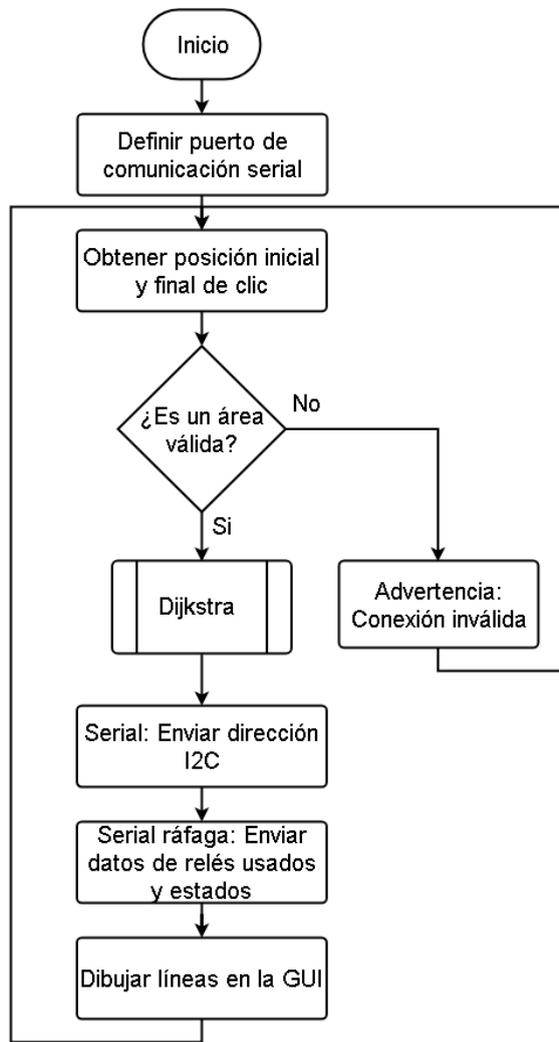


FIGURA 3.10: Flujo para la conexión y configuración de la tarjeta desde el software controlador.

## Capítulo 4

### RESULTADOS Y DISCUSIONES

La implementación de la prueba de concepto inicial se realizó colocando uno de los bloques de conexión de señales analógicas al módulo industrial para control y supervisión, específicamente en los canales de entrada 0, 1, 2 y 3 del PLC Control Logix. La forma de probar el funcionamiento del bloque de conexión es leer parámetros provenientes de la conexión analógica de los distintos dispositivos en la computadora, a través de un programa diseñado en el software Studio 5000. Este programa lee los valores en los canales mencionados y muestra en pantalla lo obtenido de cada transmisor. Para comparar se realizan dos configuraciones de cableado distintas, la primera que es la conexión directa usando cables punto a punto y la segunda que incluye al bloque de conexiones montado y funcionando.

La sección se divide en los resultados parciales de cada parte de la tesis por separado, el funcionamiento del algoritmo Dijkstra y la interfaz gráfica de usuario que corresponden al software. La tarjeta con el montaje de los relés y el FPGA XC3S200 de control que corresponden al hardware. Ambas partes luego se juntan en un solo sistema para analizar el funcionamiento del bloque de conexión con el módulo.

#### 4.1 Implementación del Software

Los resultados del algoritmo se relacionan con la interfaz gráfica, pero ambas partes trabajan en dos dominios distintos, el fin del algoritmo es realizar la ejecución lógica, mientras la interfaz gráfica permite la interacción intuitiva del software con el usuario final.

### 4.1.1 Algoritmo Dijkstra

Para comprobar que realmente los nodos origen y destino se conectan a través del camino más corto, se muestra una iteración a partir del cual se obtiene una lista de nodos que incluye un nodo origen, nodos analizados y un último nodo destino. La realización solo involucra el algoritmo por lo que hasta este punto aún no se consideran la comunicación con el hardware. Los número de nodo está asignado según la tabla 3.4 de la metodología.

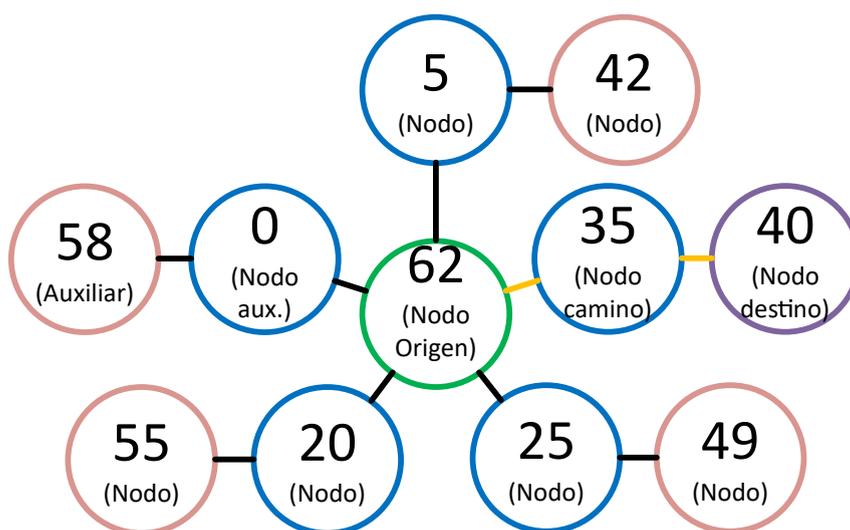


FIGURA 4.1: Algoritmo Dijkstra encuentra el camino más corto entre el nodo origen 62 al nodo destino 40.

El objetivo de la implementación es conectar el nodo origen 62 con el nodo 40, los cuales corresponden a una entrada en el controlador Control Logix y la salida del flujómetro FT11. La figura 4.1 muestra los nodos analizados a partir del nodo origen 62, siendo los círculos de color azul los posibles caminos de la primera iteración que puede tomar la señal para conectar los nodos origen y destino, la segunda iteración busca el nodo adyacente de los nodos descubiertos en la primera iteración, de color rojo tenue se muestran los nodos de la segunda iteración y de color morado el nodo destino encontrado, por lo tanto el camino más corto que elige el algoritmo es la lista de nodos 62, 35 y 40 marcados con una línea amarilla, donde el nodo 35 se asocia directamente al relé identificado con el mismo número en la implementación en hardware.

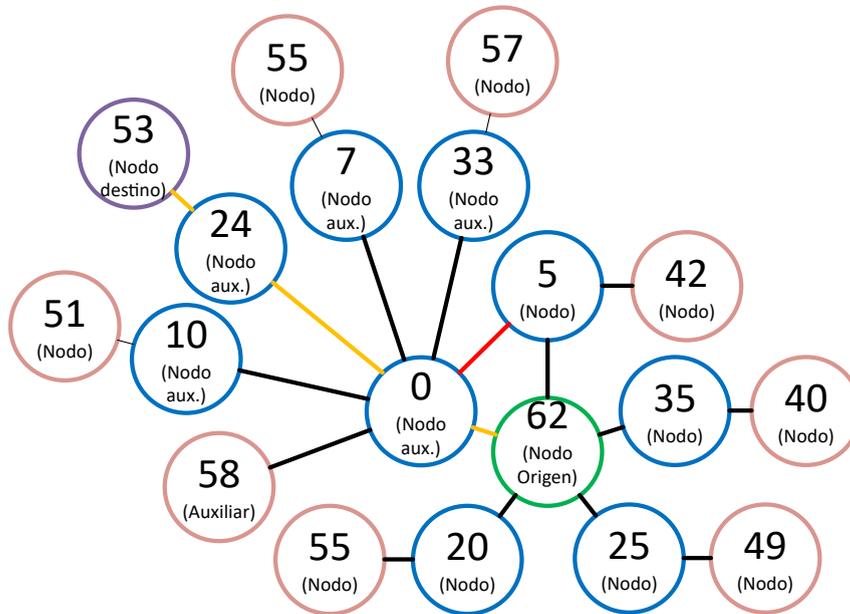


FIGURA 4.2: Algoritmo Dijkstra encuentra el camino más corto entre el nodo origen 62 al nodo destino 53.

En el caso especial para demostrar el funcionamiento de las líneas auxiliares en el algoritmo, se utilizó una conexión indirecta la cual para alcanzar una conexión del nodo origen 62 (como en el primer resultado mostrado de la figura 4.1) con el nodo destino 53, se debe conectar a través de la línea auxiliar. La figura 4.2 muestra gráficamente el mapeo de como se encontró el camino más corto, la primera iteración busca los nodos adyacentes al nodo origen, la segunda iteración busca los nodos adyacentes a los nodos previamente encontrados, debido a la presencia de nodos que representan una conexión o terminal directa al controlador, no se continúa por esa ramificación y es ahí donde acaba su camino. En el grafo el nodo 0 esta conectado al nodo 5, ya que fue analizado previamente solo se omite este nodo y se prosigue con los demás adyacentes al nodo 0 el cual solo tiene conexión con otros nodos de la línea auxiliar. Después de dos iteraciones mas se ubica el nodo destino conectado al nodo 24, por lo tanto el camino más corto determinado por el algoritmo son los nodos 62, 0, 24 y 53 marcados con una línea de color amarillo.

Tanto el grafo definido como el algoritmo no se muestran directamente al usuario por lo que también se incluye la prueba de conexión de dos terminales analógicos en el

módulo, a través de una interfaz gráfica de usuario.

#### 4.1.2 Interfaz gráfica de usuario

La Interfaz programada en el entorno de desarrollo integrado "Qt creator" une el algoritmo general del sistema con la experiencia virtual de uso del módulo industrial para control y supervisión, además muestra al usuario el estado actual de su proyecto y si se está cometiendo algún error durante la ejecución de la conexión de los instrumentos a los controladores. Las pruebas que se realizaron consisten en realizar conexiones válidas e inválidas.

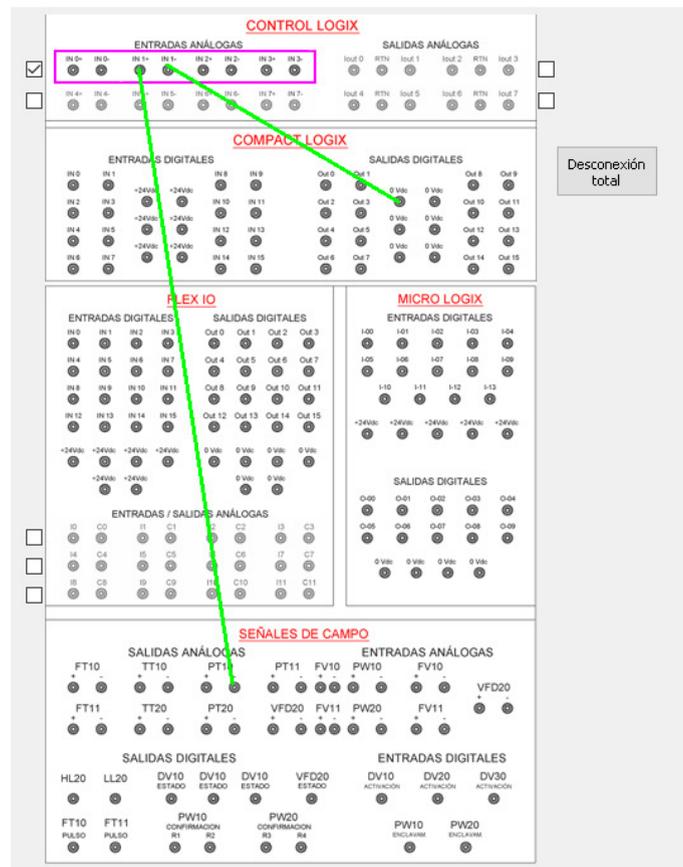


FIGURA 4.3: Conexión analógica de un instrumento de dos hilos.

Los ya mencionados criterios de conexión son probados en la interfaz gráfica, la prueba se basa en conectar un instrumento analógico de dos hilos, una conexión en serie, como se observa en la figura 4.3 se conecta el transmisor de presión PT10 al canal analógico 1 del Control Logix, se detecta como conexión válida.

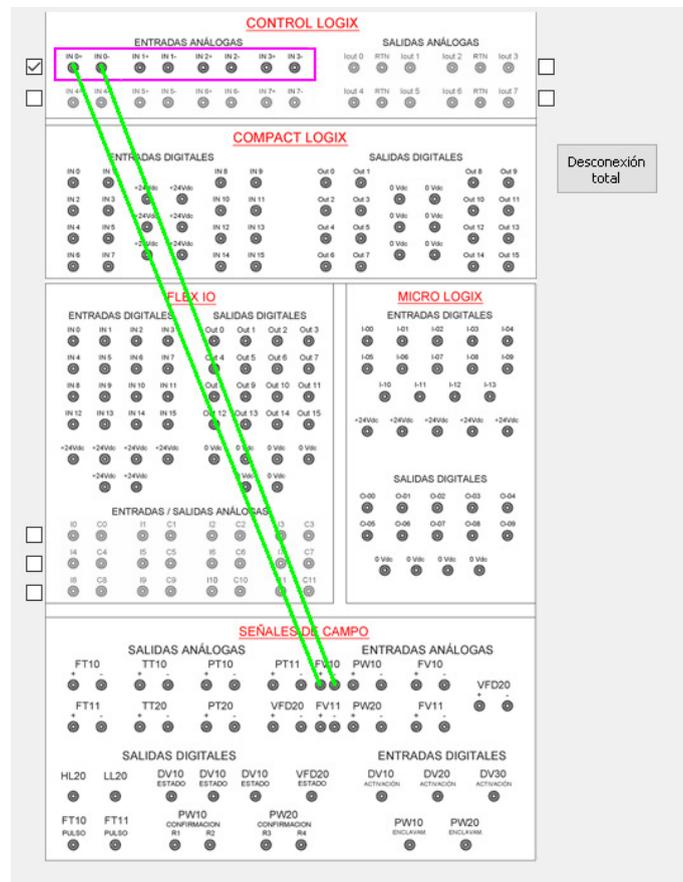


FIGURA 4.4: Conexión analógica de un instrumento de cuatro hilos.

Para el caso de la válvula proporcional FV10 conectada al canal analógico 0, se establecen ambas conexiones, polos positivo y negativo, el resultado es una conexión válida entre ambos nodos o terminales, como se muestra en la figura 4.4.

En caso se ejecute una conexión inválida, inexistente o que no cumple con los límites establecidos de que no todos los instrumentos se pueden conectar a todos los canales para un solo bloque de conexión, aparece una advertencia indicando la invalidez de esta conexión, la cual no se plasmará en la conmutación de los relés respectivos para su

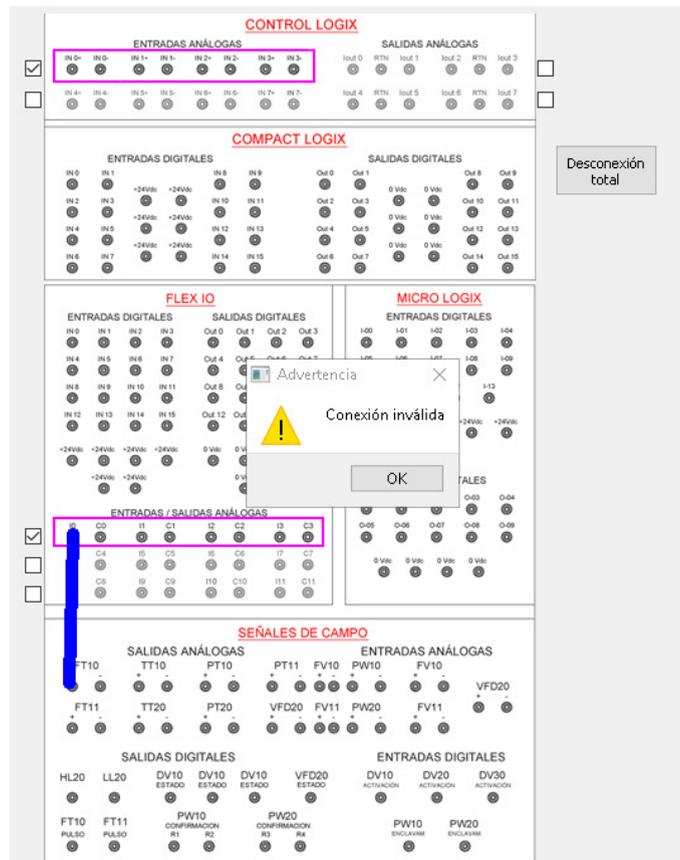


FIGURA 4.5: Advertencia de conexión inválida no existente.

direccionamiento. Tal es el caso mostrado en la figura 4.5 donde se requiere conectar el flujómetro FT10 al canal 0 del FLEX I/O de la periferia descentralizada, según la tabla 3.4 no existe un camino para que ambos nodos se conecten, por lo que se advierte al usuario que esta conexión no esta disponible.

## 4.2 Implementación del Hardware

Dado que ya se establecieron los relés necesarios correctamente identificados con según la tabla 3.4 se procedió a la implementación de la tarjeta la cual incluye el FPGA para el control de los relés independientes.

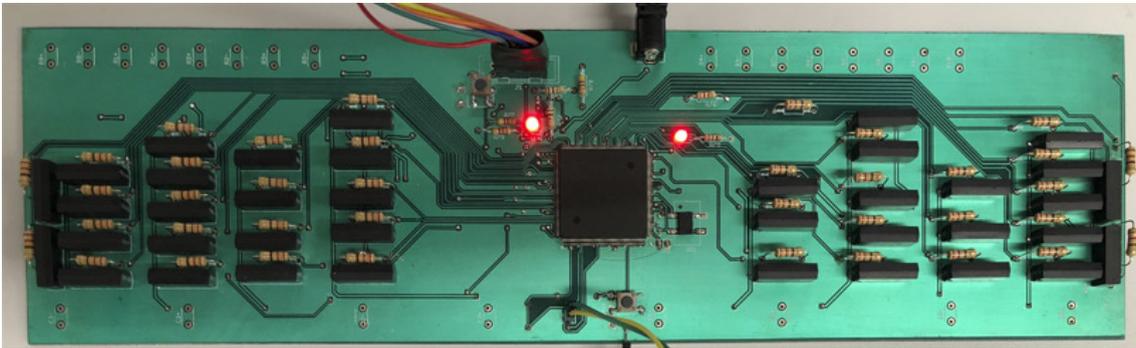


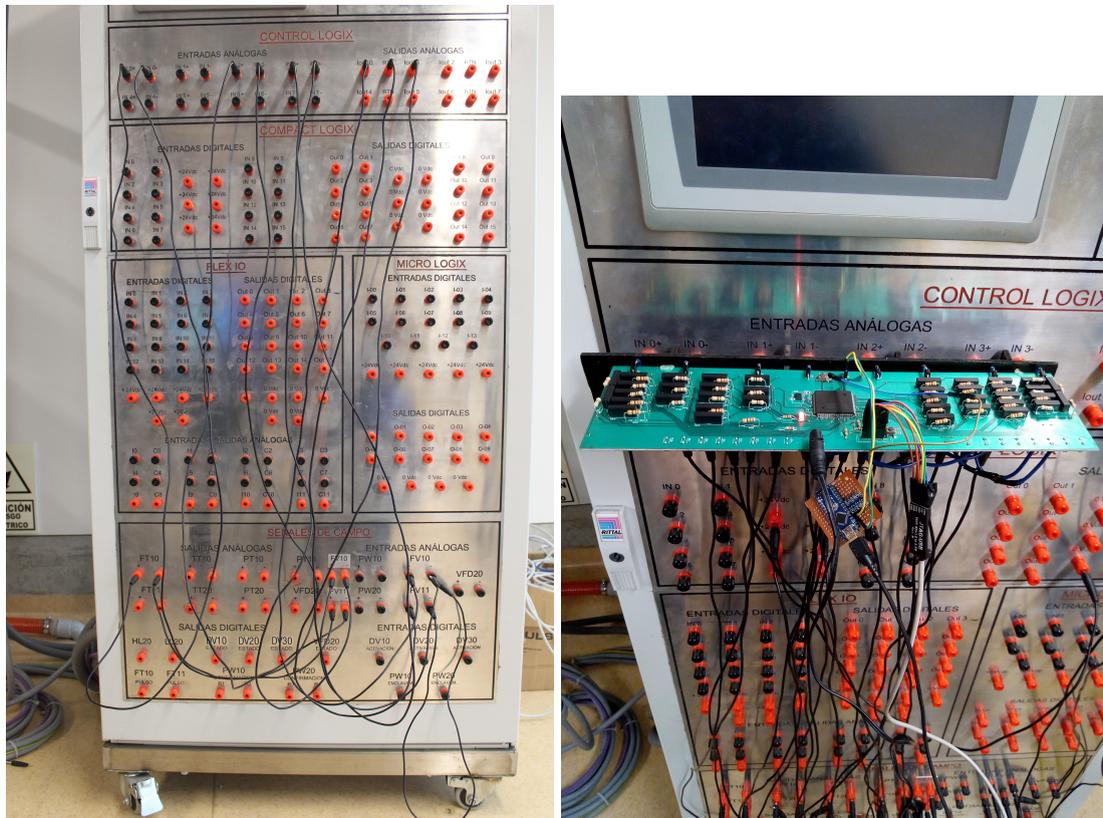
FIGURA 4.6: Bloque de conexión de señales analógicas implementado en hardware.

Internamente el FPGA está configurado con dos bloques como se ve en la figura 3.2(b) los cuales en total utilizan el 4 % de tablas de verdad de 4 entradas (4LUT por sus siglas en inglés) los cuales son recursos internos del FPGA, y usa el 29 % de buffers de entrada y salida (IOB por sus siglas en inglés) siendo un total de 38 buffers de salida para los 38 relés y 4 para la comunicación I<sup>2</sup>C ya que son señales que conmutan entre entradas y salidas.

### 4.3 Conexión con el módulo industrial para control y supervisión

Para finalizar con los resultados obtenidos se integró el módulo industrial para control y supervisión con el panel de reconfiguración de cableado, para su realización se creó un programa en el software Studio 5000 el cual lee las señales analógicas de los cuatro primeros canales en el controlador Control Logix.

Como se ve en la figura 4.7 la comparación entre el cableado estándar y la conexión modular, donde 4.7a es una conexión fija y requiere el recableado manual para poder leer un mismo dato en los diferentes canales, mientras 4.7b ofrece la posibilidad de recablear automáticamente entre 160 combinaciones, dato calculado a partir de la matriz de adyacencia según la tabla 3.4. En la experimentación se conectó directamente el bloque de conexiones en los cuatro primeros canales del controlador Control Logix, de lo cual se



(a) Módulo cableado.

(b) Bloque de conexión.

FIGURA 4.7: Comparación entre el módulo industrial y la integración con el bloque de conexión.

obtiene la tabla de validación que efectivamente una señal es conectada automáticamente y no tiene diferencia entre la conexión directa usando cables.

Por lo tanto comparando la configuración cableada manualmente de la figura 4.7a y la captura de pantalla de la interfaz de usuario de la figura 4.8 se comprueba que la didáctica de cableado no se pierde, lo que asegura que el fin con el que se desarrolló el módulo industrial para control y supervisión aún permanece intacta incluso con las limitaciones observadas en el planteamiento del modelo de conexiones.

Para finalizar la conexión realizada tanto manualmente como con el bloque de conexiones, no afecta en el funcionamiento del módulo industrial multiproósito ya que no existen caídas de tensión en las señales enrutadas, las señales analógicas no se interfieren

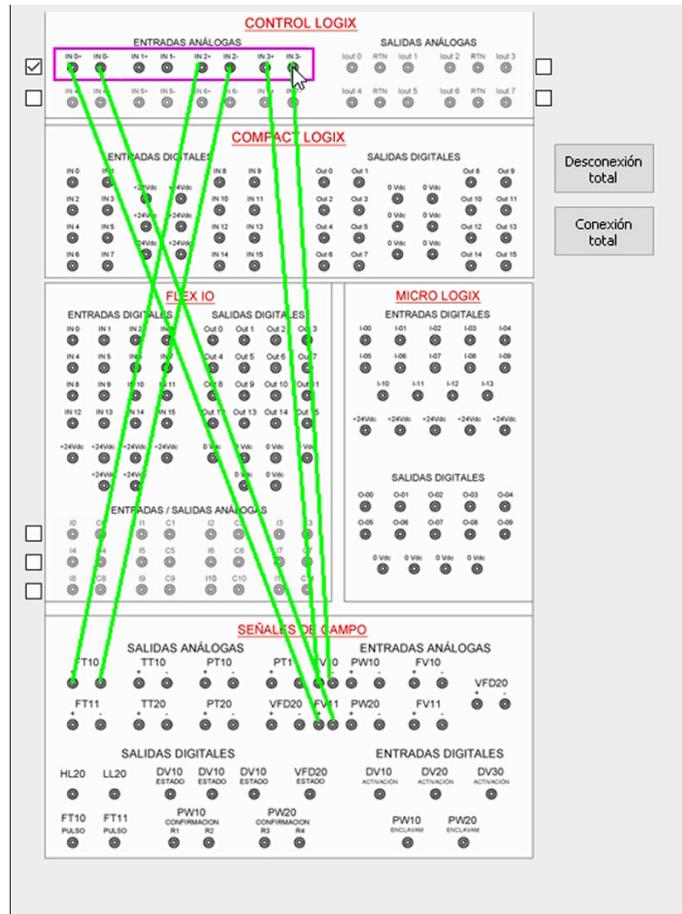


FIGURA 4.8: Conexión de tres transmisores hacia tres entradas analógicas desde la GUI.

por lo que están libres de ruido y sin modificación alguna. Lo antes mencionado se comprueba al comparar los resultados que se ven en la lectura de parámetros en el software Studio5000 en la figura 4.9 donde el canal 0 muestra el valor de posición de la válvula proporcional FV11, el canal 1 permanece desconectado, el canal 2 muestra valores del flujómetro electromagnético FT10 y el canal 3 muestra la posición de la otra válvula proporcional FV10. Tanto con el bloque de conexiones como con el cableado directo se obtienen los mismos resultados a nivel de transmisión de datos desde la planta industrial al módulo para control y supervisión.

Para el desarrollo de la primera tarjeta impresa base para un solo bloque de conexión modular, se utilizó el FPGA Spartan XC3S200 debido a que su conexión eléctrica

Name	Value	Name	Value
+ Local:4:C	{...}	+ Local:4:C	{...}
+ Local:4:I	{...}	+ Local:4:I	{...}
+ Local:5:C	{...}	+ Local:5:C	{...}
+ Local:5:I	{...}	+ Local:5:I	{...}
+ Local:5:O	{...}	+ Local:5:O	{...}
Canal0	20.210571	Canal0	20.218933
Canal1	-3.625	Canal1	-3.625
Canal2	0.0046...	Canal2	-0.006...
Canal3	14.610626	Canal3	14.615845
Canal0_HART	0.0	Canal0_HART	0.0
Canal1_HART	0.0	Canal1_HART	0.0
Canal2_HART	0.0	Canal2_HART	0.0
Canal3_HART	0.0	Canal3_HART	0.0
FV10	15.0	FV10	15.0
FV11	20.0	FV11	20.0

(a) Módulo directamente cableado.

(b) Usando el bloque de conexión.

FIGURA 4.9: Monitoreo de señales de entrada y salidas analógicas en el software Studio 5000.

está basada en el trabajo de investigación previo[9], el cual usa el mismo modelo de FPGA para la acción de conmutar relés que redireccionan señales, el análisis de este sistema escapa de los objetivos de la tesis. Además el FPGA XC3S200 cuenta con 141 pines digitales de propósito general de entrada y salida (GPIO por sus siglas en inglés) los cuales se programaron para controlar cada relé independientemente. Desde el punto de vista costo respecto a beneficio no es la solución óptima por tres razones: La primera es que la potencia consumida por el FPGA es de 0.13W, dato obtenido del software Xilinx XPower analyzer<sup>1</sup> teniendo sistemas más sencillos que consumen menos potencia como Circuitos integrados de aplicación específica (ASIC por sus siglas en inglés) o microcontroladores. La segunda razón es que el FPGA tiene un tiempo de respuesta en los GPIO de 1 ms como máximo desde que la señal va desde uno de los LUTs hasta la salida en alguno de

<sup>1</sup>Herramienta para la estimación de potencia usada en un FPGA. Última revisión 01-07-2018 [https://www.xilinx.com/support/documentation/sw\\_manuals/xilinx11/xpa\\_c\\_overview.htm](https://www.xilinx.com/support/documentation/sw_manuals/xilinx11/xpa_c_overview.htm)

sus pines, tiempo que no es justificable ya que el módulo industrial para control y supervisión no requiere alta velocidad de conmutación, sino una sola conmutación en la etapa de configuración para luego ejecutar el programa diseñado en los controladores de éste. Finalmente se tiene la opción de usar un microcontrolador con salidas suficientes para cubrir el total de señales de control por cada relé y pines necesarios para las comunicaciones, o un serializador que expande el número de salidas de un microcontrolador o FPGA con salidas limitadas por su tamaño. Cabe resaltar que la conexión entre los pines de salida del FPGA controlan directamente cada relé, el circuito comprende una resistencia en serie con el relé para reducir la corriente controlada por el FPGA y no exeder el límite de 10mA por pin según su hoja de datos. Se calculó el valor mínimo de voltaje que requiere un relé para activarse con un valor de 2.2V continuos y 220 Ohm en la resistencia en serie, ya que de acuerdo a la hoja de datos del fabricante del relé, éste ofrece una resistencia interna de 500 Ohms. Al hacer la prueba por cada relé se observó que no todos alcanzaban el nivel de voltaje mínimo para activarse, siendo en algunos casos la resistencia de 220 Ohm demasiado alta, por lo que se rediseñó el sistema con resistencias de 110Ohm.

Actualmente el algoritmo se ejecuta en el software controlador, es decir el algoritmo Dijkstra que busca el camino más corto para llegar de un nodo a otro, es procesado por la computadora, para luego el resultado de los nodos que deben ser conectados se transmiten al FPGA para que direcciona las señales correspondientes. No es un proceso complejo, según los resultados obtenidos el software se ejecuta en aproximadamente 4 milisegundos y no llega a ocupar el 1 % de recursos de una computadora personal estándar. Desde el punto de vista de optimización, todo el procesamiento puede ejecutarse en el FPGA, el cual, por su estructura de LUT el procesamiento del camino más corto se puede hacer en el FPGA aprovechando los recursos que hasta ahora están suspendidos.

Se vió que al conectar un solo bloque en el módulo industrial para control y supervisión no todos los instrumentos se pueden conectar a todos los canales disponibles para ese bloque específico. La limitación se produce al reducir el número de relés en la red

debido al tamaño de hardware que significa su implementación como tarjeta electrónica. Aun con esta limitación, el sistema tiene la capacidad de cubrir la conexión de cuatro instrumentos, número máximo de conexiones que soporta cada bloque, de esta forma se compensa la limitación y se aprovecha la posibilidad de conexiones variadas.

Durante la ejecución de la configuración en la interfaz gráfica de usuario se puede observar una configuración que el sistema marca como válida y permite su conexión aun-que esté errónea. Debido al principio de conexión de los instrumentos de cuatro hilos, solo se conectan dos cables punto a punto los cuales corresponden a conectar terminales positivos y negativos respectivamente entre el controlador a usar y el instrumento a conectar. La figura cambia cuando uno de los terminales del instrumento se conecta en el terminal positivo de un bloque de conexión, pero el par negativo puede conectarse en el terminal negativo de otro bloque de conexión del mismo controlador o incluso en otro controlador.

Por ejemplo si se quiere leer la salida analógica de la válvula FV10 conectado al quinto canal del controlador Control Logix como se muestra en la figura 4.10, el polo positivo del flujómetro está en el conector de misma polaridad en canal correspondiente, mas el polo negativo está conectado al polo negativo de otro canal, lo que producirá que no se lea dato alguno en el puerto deseado o se tenga un dato erróneo.

Para la implementación de la tesis usando un solo bloque de conexión como prueba de concepto, el cable programador para el FPGA y la interfaz del bus I<sup>2</sup>C se invirtió un aproximado de 200 dólares americanos. Por lo tanto, un sistema completo que integre todas las conexiones analógicas tendrá un costo total de aproximadamente US\$1400, siendo aún un precio inviable para la implementación total como parte del presente trabajo de tesis. Sin embargo, otra forma de realizar automáticamente las conexiones sin perder la didáctica del bloque sería implementar una especie de brazo robótico que sea capaz de mover cada conexión cuando se requiera, incrementando así los costos generales para un proyecto de reconexión automática en un módulo para control y supervisión. Muy aparte de la diferencia de precio de implementación, el factor tiempo de implementación y

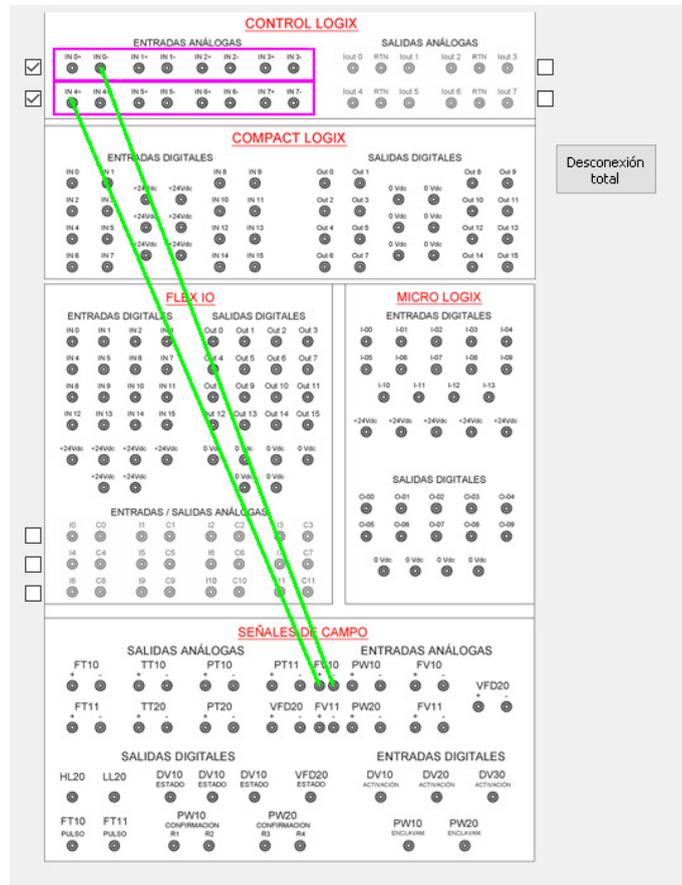


FIGURA 4.10: Conexión válida errónea, la cual conecta un transmisor en dos puertos distintos.

complejidad de uso de equipos es un punto a favor, ya que a comparación de un sistema mecánico el funcionamiento es sencillo al intercambiar líneas conmutando relés.

La continuación de este trabajo, con su respectiva implementación total para todo el módulo sería el inicio de lograr ofrecer laboratorios remotos de hardware. Sin embargo, esto debería ser complementado con un trabajo de diseño de software en temas de estabilidad de conexiones remotas y de seguridad en redes.

## CONCLUSIONES

Se logró implementar el panel de reconfiguración de cableado automático con hasta un máximo de cuatro conexiones por bloque, permitiendo la conexión absoluta con siete módulos iguales repartidos en el tablero de conexiones del módulo industrial para control y supervisión. A partir del mapeo inicial de conexiones y el modelo reducido planteado para un solo bloque de conexión, se redujo en 55 % la cantidad de relés usados, lo que permitió la implementación de una tarjeta más pequeña con la cantidad mencionada de relés, asegurando la conexión de cualquier instrumento analógico con cualquier controlador del mismo tipo de señal en sentido bidireccional, ya que el modelo funciona tanto para entradas como salidas.

Del modelo de conexiones se diseñó una tarjeta que alberga 38 relés conectados a las salidas del FPGA Spartan XC3S200 a cada una de sus salidas, una resistencia en serie que evita una línea cerrada la que se traduce en un cortocircuito entre la alimentación de una salida del FPGA y la referencia a tierra. Posterior a ello se implementó la tarjeta con su respectiva interfase de comunicación la cual traduce el protocolo serial a protocolo I2C para integrar en una sola red la computadora principal y las tarjeta que hace de bloque de conexión en el módulo industrial para control y supervisión.

Toda limitación de conexión está definida por el grafo definido por software, en el que se incluye el algoritmo Dijkstra para determinar el camino más corto a través del cual una señal será direccionada. Se incluye al desarrollo del software la interfaz gráfica

de usuario, siendo esta interfaz el nexo por el cual el usuario realiza una conexión virtual que se procesa y traduce en una conexión física en el bloque de conexiones.

Se demostró que el cableado automático usando el bloque de conexiones en el módulo industrial para control y supervisión no altera la señal transmitida desde un instrumento a los controladores o viceversa, por lo que el sistema es ciego a este hardware lo que evita suprimir la dinamicidad con la que el módulo se utiliza. La reducción de tiempo en el recableado se disminuye permitiendo trabajar con distintas configuraciones de cableado por proyecto implementado.

Como trabajo futuro se debe plantear un sistema de prueba rápido para asegurar que los relés están funcionando correctamente, actualmente la cantidad posible de combinaciones es de  $2^{38} = 8.8\text{M}$  por lo que toma demasiado tiempo probar el funcionamiento de todas las combinaciones posibles.

## ANEXOS

Los archivos utilizados para la prueba del bloque de conexión son:

- Hardware: Descripción de hardware a implementar en el FPGA:
  - main.vhd
  - i2c\_slave.vhd
  - panel\_reconfig.ucf
  - main.bit
- Controlador - Software.zip: incluye los archivos SerialGUI.cpp y SerialGUI.h para crear el entorno en un compilador C++.
- Programa de prueba - Studio5000: Para ser ejecutado en el software Studio5000, el cual escribe valores analógicos en las salidas de los controladores y lee datos provenientes de cualquier transmisor en planta.

se encuentran adjuntos en la versión digital de la tesis siguiendo el link a continuación:

<https://drive.google.com/open?id=1whzd6Ncfc5A5bo4Ch5uyFAwMOrHMDxUc>

## REFERENCIAS BIBLIOGRÁFICAS

- [1] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*. the MIT Press, 2009.
- [2] V. Murray, A. Bejarano, and C. Matsuno, “An open coworking space to allow engineering students to develop innovative competences: Utec garage,” in *Proc. IEEE 8th Int. Conf. Engineering Education (ICEED)*, Dec. 2016, pp. 109–114.
- [3] L. Castillo, A. Rojas, and V. Murray, “Módulo industrial multipropósito para control y supervisión,” Perú patent 522-2017 DIN, Mar. 28, 2017.
- [4] ———, “Planta industrial multipropósito para control e instrumentación,” Perú patent 523-2017 DIN, Mar. 28, 2017.
- [5] A. Creus, *Instrumentación Industrial*, 8th ed., Alfaomega, Ed. Marcobo, 2010.
- [6] J. Calderón and Y. Sánchez, *MEDICIONES E INSTRUMENTACIÓN*, 2004.
- [7] G. Patino and V. Murray, “Dynamic reconfigurable wiring panel for a multipurpose industrial module,” pp. 1–4, Nov 2018.
- [8] A. DeHon, “Unifying mesh- and tree-based programmable interconnect,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 10, pp. 1051–1065, Oct. 2004.
- [9] V. Murray, M. Pattichis, D. Llamocca, and J. Lyke, “Field-programmable wiring systems,” *Proceedings of the IEEE*, vol. 103, no. 7, pp. 1159–1180, Jul. 2015.

- [10] J.-E. Wong, J. H. Lang, and M. A. Schmidt, "An electrostatically-actuated MEMS switch for power applications," in *Proc. IEEE Thirteenth Annual Int. Conf. Micro Electro Mechanical Systems (Cat. No.00CH36308)*, Jan. 2000, pp. 633–638.
- [11] J. Li and C.-K. Cheng, "Routability improvement using dynamic interconnect architecture," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 6, no. 3, pp. 498–501, Sep. 1998.
- [12] M. J. Alexander and G. Robins, "New performance-driven FPGA routing algorithms," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 15, no. 12, pp. 1505–1517, Dec. 1996.
- [13] F. Group, "Hart - digital transformation for analog instruments," <https://fieldcommgroup.org/technologies/hart>, 2018, Último acceso: 2018-07-16. [Online]. Available: <https://fieldcommgroup.org/technologies/hart>
- [14] Tutorial-Reports.com, "Fpga routing tutorial," <http://www.tutorial-reports.com/computer-science/fpga/routing.php>, 2013, Último acceso: 2018-07-16.
- [15] G. ATAP, "Advanced technologies and projects," <https://atap.google.com/>, 2016, Último acceso: 2018-07-16.
- [16] T. Verge, "Ara project," [https://www.wired.com/2016/05/project-ara-lives-googles-modular-phone-is-ready/?mbid=social\\_twitter](https://www.wired.com/2016/05/project-ara-lives-googles-modular-phone-is-ready/?mbid=social_twitter), 2016, Último acceso: 2018-07-16.
- [17] D. Alba, "Death of project ara shows google is all grown up," <https://www.wired.com/2016/09/death-project-ara-shows-google-grown-up/>, 2016, Último acceso: 2018-07-16.