

# **ESCUELA POLITÉCNICA NACIONAL**

## **FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**

### **IMPLEMENTACIÓN DE PROTOTIPOS DE DRIVERS DE COMUNICACIÓN Y HOJAS DE DATOS ELECTRÓNICAS (EDS) DEVICENET DE BAJO COSTO PARA SENSORES INDUSTRIALES ANALÓGICOS Y DISCRETOS**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE  
INGENIERO EN “ELECTRÓNICA Y CONTROL”**

**MARÍA GABRIELA GAMBOA CHIRIBOGA**

**Email:maria.gamboa@epn.edu.ec**

**DIRECTOR: MSc. SILVANA DEL PILAR GAMBOA BENÍTEZ**

**Email:silvana.gamboa@epn.edu.ec**

**CODIRECTOR: MSc.PATRICIO IVÁN CHICO HIDALGO**

**Email:patricio.chico@epn.edu.ec**

**Quito, Noviembre 2018**

## **AVAL**

Certificamos que el presente trabajo fue desarrollado por María Gabriela Gamboa Chiriboga, bajo nuestra supervisión.

---

**MSc.SILVANA DEL PILAR GAMBOA BENÍTEZ**  
**DIRECTOR DEL TRABAJO DE TITULACIÓN**

---

**MSc.PATRICIO IVÁN CHICO HIDALGO**  
**CODIRECTOR DEL TRABAJO DE TITULACIÓN**

## DECLARACIÓN DE AUTORÍA

Yo, María Gabriela Gamboa Chiriboga, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

María Gabriela Gamboa Chiriboga

## **DEDICATORIA**

Mi tesis la dedico con todo mi amor a mi madre Ximena que con su perseverancia, apoyo y cariño durante toda mi vida han hecho posible que culmine con éxito mis estudios. Quiero que sepas que eres la mujer más fuerte que he conocido, me llena de orgullo ser tu hija y me siento muy bendecida y agradecida con Dios por haberme dado la mejor madre del mundo.

Gracias por ser mi apoyo cada día, por los esfuerzos impresionantes que tuviste que hacer y por seguir apoyándome en cada paso que doy.

Felicidades mami, ya somos Ingenieras.

## **AGRADECIMIENTO**

Primeramente, agradezco a mi madre Ximena que ha sido mi mejor amiga y ha permanecido a mi lado siempre.

A mi hermana María José que ha sido mi ejemplo a seguir y mi compañera de estudio.

A mi hermano Chris y su familia María José y Annalía con los que he compartido muchos momentos de felicidad.

A mi perrito Bruno que me llena de alegría y me acompañó durante mi vida universitaria.

# ÍNDICE DE CONTENIDO

AVAL.....	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN.....	VIII
ABSTRACT.....	IX
1. INTRODUCCIÓN.....	1
1.1 Objetivos.....	1
1.2 Alcance.....	2
1.3 Marco Teórico.....	3
Protocolo Industrial Común (CIP).....	3
Introducción a CIP.....	3
CIP Protocol Suite.....	4
Formato de Representación de Objetos CIP.....	6
Formato de Intercambio de Datos CIP.....	7
Comunicación Industrial DeviceNet.....	11
Estandarización de DeviceNet.....	12
Características Generales de la Red DeviceNet.....	13
Arquitectura de la Red DeviceNet.....	14
Herramientas de Software a Usarse.....	27
RSLogix5000.....	28
RSNetWorx para DeviceNet.....	28
RSLinx.....	29
USB-CAN Tool V2.02.....	30
FactoryTalk View Studio.....	31
2. METODOLOGÍA.....	32
2.1 Análisis de la Comunicación en Redes DeviceNet.....	32
Análisis de la Interacción Escáner/Sensor.....	34
Determinación de Formato de Trama de Datos.....	36
Análisis de Campo de Identificación.....	36
Análisis de Datos.....	39
Análisis de Formato de Intercambio de Datos.....	41

Hoja de Datos Electrónica (EDS).....	44
<b>2.2    Diseño e Implementación de Drivers de Comunicación DeviceNet .....</b>	<b>46</b>
Especificación de Componentes y Requerimientos de Implementación.....	46
Selección de Componentes .....	48
Módulo de Sensores CAN.....	48
Implementación de Drivers de Comunicación .....	51
Diseño e Implementación de Hardware .....	51
Desarrollo de Software.....	57
<b>2.3    Desarrollo de la Plataforma de Validación y Pruebas .....</b>	<b>58</b>
Módulo de Red Industrial DeviceNet .....	58
Software de Configuración.....	59
Configuración de la red DeviceNet.....	60
Configuración de RSLinx.....	60
Configuración de la Aplicación con RSLogix5000 .....	61
Configuración de la red con RSNetWorx.....	65
Mapeo de Datos con RSNetWorx.....	71
Aplicación para Prueba de Prototipos Implementados .....	77
Desarrollo del Interface de Operador con FactoryTalk View Studio.....	78
Interface de Operador Implementado .....	88
<b>3.    RESULTADOS Y DISCUSIÓN .....</b>	<b>92</b>
<b>3.1    Pruebas de Funcionamiento .....</b>	<b>92</b>
Pruebas de Hardware.....	92
Prueba de Calibración de Sensores .....	95
Pruebas de Repetibilidad y Resolución .....	97
Pruebas de Conectividad.....	101
<b>3.2    Monitoreo de la red DeviceNet con RSNetWorx .....</b>	<b>102</b>
<b>3.3    Monitoreo de Trama DeviceNet con USB-CAN Tool V2.02 .....</b>	<b>106</b>
Maestro/Esclavo Nodo de Nivel.....	107
Maestro/Esclavo Nodo Discreto .....	108
Maestro/Esclavo Nodo de Temperatura.....	109
<b>3.4    Monitoreo del HMI con FactoryTalk View Studio.....</b>	<b>109</b>
Funcionamiento del Nodo de Nivel.....	109
Funcionamiento del Nodo Discreto .....	111
Funcionamiento del Nodo de Temperatura.....	113
<b>4.    CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>115</b>

4.1 Conclusiones.....	115
4.2 Recomendaciones.....	116
5. REFERENCIAS BIBLIOGRÁFICAS .....	118
6. ANEXOS.....	120
ANEXO I.....	120
ANEXO II.....	121
ANEXO III.....	152
7. ORDEN DE EMPASTADO .....	158



## RESUMEN

El desarrollo de la industria mundial está marcado por la integración de las nuevas tecnologías de comunicación e información, pero esta integración de tecnología conlleva una inversión económica representativa. En el caso de la industria ecuatoriana, la integración de tecnología es limitada, sobre todo en la pequeña y mediana industria, pues la inversión económica inicial puede ser elevada. Esto puede traer como consecuencia el atraso de estas industrias ya que se privarán de los beneficios del uso de estas nuevas tecnologías.

Es por ello que este trabajo de titulación propone desarrollar un prototipo de driver de comunicación DeviceNet de bajo costo para sensores, así como las hojas de datos electrónicas necesarias para que estos drivers puedan ser configurados en aplicaciones computacionales de ingeniería comerciales. Se propone desarrollar los prototipos para redes DeviceNet justificado en que este tipo de redes se rige al estándar CIP (Common Industrial Protocol) en el que se basan otras redes como Ethernet/IP o ControlNet, y con las que DeviceNet es compatible.

Se desarrolla las placas electrónicas necesarias para la implementación de dos drivers para sensores analógicos y un driver de sensor discreto multivariable. Los drivers diseñados se probarán integrándolos a una red DeviceNet, con el objetivo de verificar la interoperabilidad del driver con equipos DeviceNet comerciales utilizando aplicaciones de software de configuración y monitoreo de redes DeviceNet tales como RSNETWORX, FACTORYTALK VIEW STUDIO, RSLOGIX 5000.

**PALABRAS CLAVE:** Driver de Comunicación DeviceNet, Estándar CIP (Common Industrial Protocol), Red de Dispositivos, Hojas de datos electrónicas.

## ABSTRACT

The development of the global industry is marked by the integration of new communication and information technologies, but this integration of technology entails a representative economic investment. In the case of the Ecuadorian industry, the integration of technology is limited, especially in the small and medium industry, since the initial economic investment can be high. This can result in the backwardness of these industries as they will be deprived of the benefits of using these new technologies.

Therefore this work proposes to develop a prototype of a low cost DeviceNet communication driver for sensors, as well as the electronic data sheets necessary for these drivers to be configured in commercial engineering computational applications. It is proposed to develop prototypes for justified DeviceNet networks in which this type of networks is governed by the CIP (Common Industrial Protocol) standard on which other networks such as Ethernet / IP or ControlNet are based, and with which DeviceNet is compatible.

It develops electronic boards necessary for the implementation of two drivers for analog sensors and a discrete multivariable sensor driver. The designed driver will be tested by integrating it into a DeviceNet network, with the objective of indicating the interoperability of the driver with commercial DeviceNet equipment using software configuration and monitoring applications of DeviceNet networks such as RSNETWORX, FACTORYTALK VIEW STUDIO, RSLOGIX 5000.

**KEYWORDS:** DeviceNet Communication Driver, CIP Standard (Common Industrial Protocol), Device network, Electronic Data Sheet.

# 1. INTRODUCCIÓN

En el sector industrial existen dispositivos comerciales de campo como son sensores, actuadores y controladores de lazo inteligentes provistos de diferentes protocolos de comunicación que facilitan la interacción entre dispositivos esclavos y maestros de distintas marcas comerciales. Estos dispositivos cumplen con estándares que garantizan su correcto funcionamiento y además brindan seguridad tanto a operarios como a los equipos que intervienen en un proceso industrial, pero su costo tiende a ser elevado y por tal motivo se restringen a ser usados en determinadas industrias con capacidad de inversión media-alta. Desafortunadamente, existen pocos dispositivos comerciales de campo con protocolos de comunicación como DeviceNet, Profibus, EtherNet/IP, entre otros enfocados a sectores con capacidad limitada de inversión, esto debido a que dentro del sector industrial se obedece más a una lógica de costo-beneficio.

Por otra parte, en el mercado se ofrecen componentes electrónicos comerciales de bajo costo, por lo que es posible desarrollar tarjetas electrónicas que en conjunto con herramientas computacionales permitan implementar drivers de comunicación de protocolos industriales como DeviceNet, Profibus, EtherNet/IP, entre otros, para dispositivos de campo como sensores analógicos y discretos que no posean interface de comunicación. Al ser tarjetas de bajo costo pueden ser adquiridas por industrias con capacidad limitada de inversión.

De esta forma el presente trabajo propone desarrollar un prototipo de driver de comunicación de bajo costo para sensores industriales analógicos y discretos para que puedan ser integrados a una red DeviceNet. Adicionalmente se propone la creación de las hojas electrónicas de datos (EDS) que se deben instalar en aplicaciones computacionales comerciales ya sea para la configuración inicial de estos drivers de comunicación dentro de una Red DeviceNet o para el monitoreo de estas redes durante operación normal.

## 1.1 Objetivos

El objetivo general de este trabajo es:

Implementar prototipos de drivers de comunicación y hojas de datos electrónicas (EDS) DeviceNet de bajo costo para sensores industriales

Los objetivos específicos de este trabajo son:

- Estudiar la base teórica y las normativas relacionadas con las redes DeviceNet, así como las características de la comunicación de sensores industriales a este tipo de redes.
- Determinar los formatos de las tramas de datos que deberán implementarse en los drivers de comunicación, así como el formato de las hojas de datos electrónicos EDS para software de configuración de redes DeviceNet.
- Establecer los requerimientos de hardware y de software para la implementación de drivers de comunicación DeviceNet para sensores industriales.
- Implementar el hardware correspondiente, la programación necesaria de los drivers de comunicación y los archivos EDS, de acuerdo a las características y requerimientos establecidos previamente.
- Validar el funcionamiento de los drivers de comunicación y los EDS desarrollados, integrándolos en una red industrial DeviceNet comercial.

## **1.2 Alcance**

El trabajo iniciará con un estudio de la base teórica y las normativas relacionadas con el CIP Protocol Suite (Common Industrial Protocol Suite), dentro del que se contemplan las redes DeviceNet. También se revisarán las características y requerimientos mínimos de la comunicación de sensores en redes de campo industriales. Luego se procede a la determinación de los formatos de las tramas de comunicación, así como las características de las hojas de datos electrónicos (EDS), para lo cual se analizará tramas de sensores comerciales conectados a una red DeviceNet, así como aplicaciones de software de configuración y monitoreo de redes DeviceNet. Tras este estudio se establecerá los requerimientos de hardware y de software para la implementación de los drivers. Posteriormente, se implementará el hardware necesario, se implementarán los EDSs, y se realizará la programación necesaria para generar las tramas DeviceNet, así como cumplir los requerimientos de comunicación establecidos previamente.

Finalmente se procederá a la validación del funcionamiento de los drivers y los EDSs desarrollados, integrándolos a una red industrial DeviceNet comercial. Para ello se implementará un sistema sencillo de monitoreo y control con componentes DeviceNet disponible en el Laboratorio de Redes Industriales. Con esto se busca comprobar la interoperabilidad de los drivers diseñados con equipos DeviceNet comerciales.

### 1.3 Marco Teórico

En este capítulo se presenta una revisión bibliográfica sobre el Protocolo Industrial Común (Common Industrial Protocol) y la comunicación industrial DeviceNet. Adicionalmente se describen los parámetros de las hojas de datos electrónicas, así como las características y requerimientos mínimos de la comunicación de sensores en redes de campo industriales.

#### Protocolo Industrial Común (CIP)

##### Introducción a CIP

El Protocolo Industrial Común, CIP, fue creado por ODVA y ControlNet International con el objetivo de desarrollar una capa de aplicación de comunicación, basada en el modelo de comunicación OSI (Open System Interconnection), en la que no sea necesario implementar una tecnología particular para interactuar con las capas más bajas de tal modelo, las cuales definen las características físicas para llevar a cabo la comunicación.

De esta manera se utiliza esta capa de aplicación común a las diferentes tecnologías de redes con la intención de facilitar la integración de las redes en los diferentes niveles de automatización y reducir los aislamientos que se presentan entre estos niveles dentro de una industria.

El Protocolo CIP al integrar diferentes redes bajo una capa de aplicación común que contempla las funciones de Aplicación, Presentación y Sesión del modelo OSI ha dado lugar a la creación de lo que se conoce como la “Familia de Redes CIP” o CIP Protocol Suite constituido por Ethernet/IP, ControlNet y DeviceNet y cuyos perfiles de comunicación se muestran en la Figura 1.1 [3], [4].

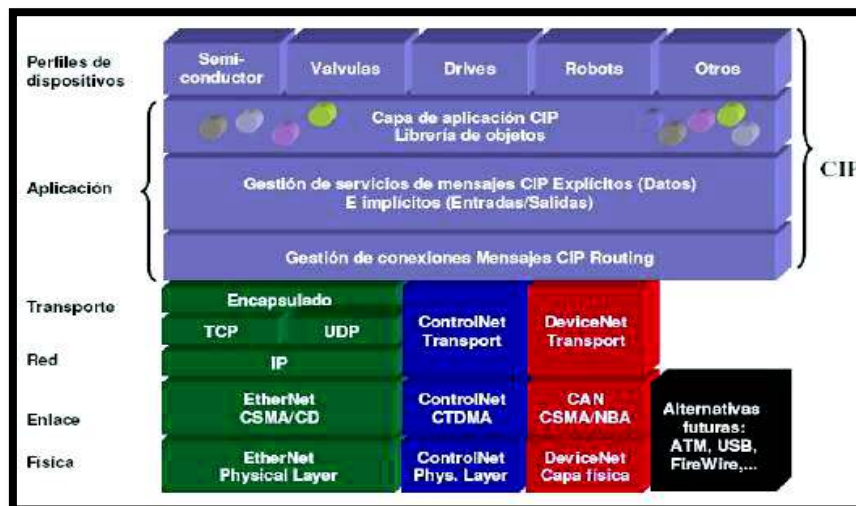


Figura 1.1 Familia de redes CIP [3], [4]

## CIP Protocol Suite

Las redes DeviceNet, ControlNet y EtherNet/IP definen su capa de aplicación basados en el protocolo CIP y por esta razón estas tres redes comparten características comunes, de allí que conforman el definido “CIP Protocol Suite” o “Familia de Redes CIP” y que se detallan a continuación.

**Red DeviceNet:** Esta red está basada en el modelo OSI estructurado en siete capas: capa física, capa de enlace de datos, capa de transporte, capa de presentación, capa de sesión y capa de aplicación. Su característica importante es que incorpora el protocolo CAN (Controller Area Network) en la capa de enlace de datos y física [2]. Este tipo de red está orientada a la integración de dispositivos de los niveles inferiores de la automatización tales como sensores, actuadores y controladores de lazo.

**Red ControlNet:** Es una red creada para habilitar la interacción e interoperabilidad entre controladores en sistemas de automatización, está orientada a ser utilizada en aplicaciones de tiempo real debido a que emplea alta velocidad para la transmisión de datos I/O, permite configurar mensajes y datos punto a punto (peer-to-peer) realizando estas acciones utilizando un mismo medio físico. ControlNet es una red determinística es decir es factible establecer los tiempos de transmisión de datos y al ser cíclica asegura que estos tiempos de transmisión sean constantes. Adicionalmente, este tipo de red permite un solo publicador en la red que hace uso de mensajes multicast en lugar de una comunicación punto a punto reduciendo así el tráfico de datos en la red [3].

**Red EtherNet/IP:** Ethernet/IP es la adaptación de CIP sobre la tecnología Ethernet emplea el estándar IEEE 802.3. Este tipo de red permite transportar grandes volúmenes de información y posee las bases de Ethernet tradicional en cuanto a capa física y enlace de datos y contempla a nivel de capa de red y transporte los protocolos del TCP/IP Protocol Suite [3].

ODVA y ControlNet International establecen las especificaciones para estas redes CIP determinando que cada red se va a componer de dos volúmenes:

- Volumen común para capas de aplicación
- Volumen específico para capas inferiores

A continuación, se detalla la organización de estas especificaciones CIP [3]:

La especificación EtherNet/IP consiste en:

- ✓ Volumen 1: Protocolo Industrial Común

- ✓ Volumen 2: Adaptación CIP a EtherNet/IP

La especificación DeviceNet consiste en:

- ✓ Volumen 1: Protocolo Industrial Común
- ✓ Volumen 3: Adaptación CIP a Devicenet

La especificación ControlNet consiste en:

- ✓ Volumen 1: Protocolo Industrial Común
- ✓ Volumen 4: Adaptación CIP a ControlNet

Las especificaciones de seguridad CIP:

- ✓ Volumen 5: Seguridad CIP

El volumen 1 describe los aspectos que son comunes a todas las adaptaciones de la red.

El volumen 2 es la adaptación de CIP para el transporte a TCP/IP y de UDP/IP sobre Ethernet.

El volumen 3 es la adaptación de CIP para DeviceNet, describe cómo CIP es adaptado a la capa de enlace de datos CAN.

El volumen 4 es la adaptación CIP para ControlNet, describe cómo CIP es adaptado a la capa de enlace de datos ControlNet.

El volumen 5 contiene todo lo referente a seguridad que debe incorporarse en las redes CIP.

La Tabla 1.1 muestra los volúmenes junto con las redes que han sido incorporadas por CIP.

**Tabla 1.1** Volúmenes de redes CIP [3]

Capas #5-#7	<b>CIP Volumen 1</b>		
Capas #1-#4	<b>CIP Volumen 2</b>	<b>CIP Volumen 3</b>	<b>CIP Volumen 4</b>
	EtherNet/IP	DeviceNet	ControlNet

Como se verá más adelante el volumen 1 y 3 son fundamentales para entender la adaptación de CIP a una red de campo DeviceNet.

## Formato de Representación de Objetos CIP

El protocolo CIP utiliza un modelado basado en objetos para describir la información, los servicios de comunicación disponibles y el comportamiento de un nodo CIP. De esta forma, CIP utiliza el modelado basado en objetos como una forma de representación abstracta de información y así definir las características de los dispositivos conectados a la red.

Los objetos CIP están estructurados en los siguientes elementos [3], [4], [5]:

- **Clase:** Objeto a través del cual se puede representar a un conjunto de componentes de sistemas con características comunes es decir del mismo tipo. Es así que se la puede considerar como una plantilla predefinida para la creación de objetos.
- **Instancia:** Objeto creado a partir de una clase con el cual se realizará una representación real de un objeto particular dentro de una clase.
- **Atributos:** Conjunto de datos que especifican las características y estados de una instancia perteneciente a una clase.

En la Figura 1.2 se representa el modelado de objetos de un nodo CIP.

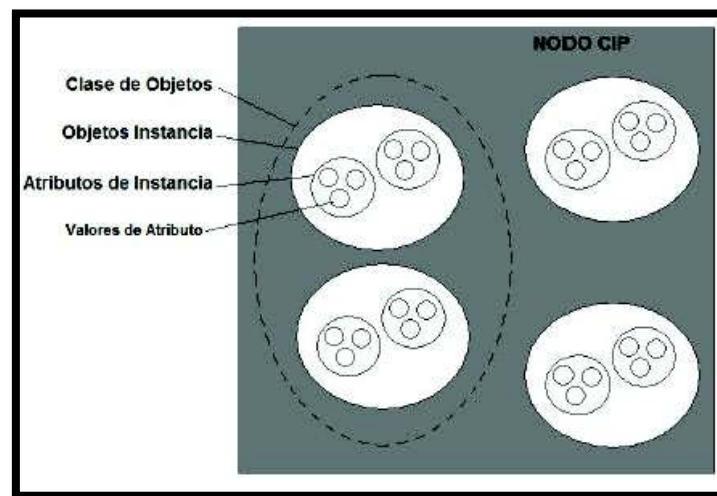


Figura 1.2 Modelo de Objetos Nodo CIP [3]

El protocolo CIP sigue un formato para direccionar los objetos que representan sus componentes: [3]

- **Dirección de Nodo** (Node Address)
- **Identificador de Clase** (Class Identifier o Class ID)
- **Identificador de Instancia** (Instance Identifier o Instance ID)



- **Identificador de Atributo** (Attribute Identifier o Attribute ID)
- **Código de Servicio** (Service Code)

A continuación, se realiza una descripción de cada objeto CIP.

**Dirección de Nodo:** Permite identificar a un nodo de la red CIP, en DeviceNet es denominado MAC-ID.

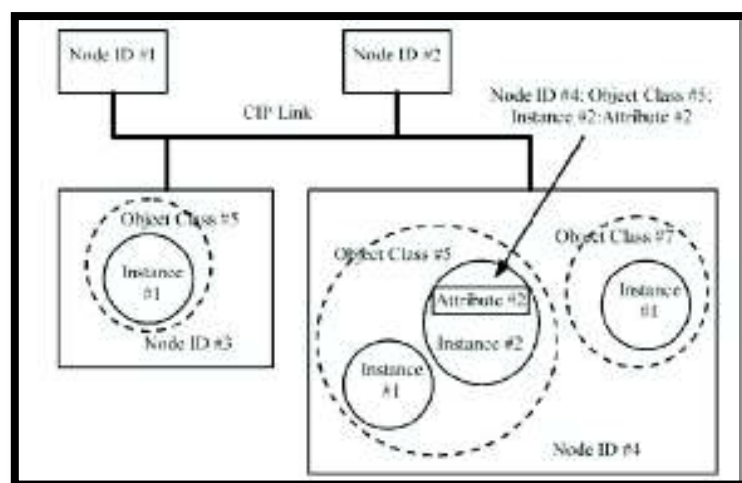
**Identificador de Clase:** Permite identificar a una clase de objetos que son accesibles desde la red.

**Identificador de Instancia:** Permite identificar a una instancia haciendo posible su diferenciación de otra instancia dentro de una misma clase.

**Identificador de Atributo:** Permite identificar a un atributo dentro de una clase.

**Código de Servicio:** Permite identificar una solicitud de acción para un objeto de instancia o clase.

En la Figura 1.3 se muestra los componentes de los objetos CIP

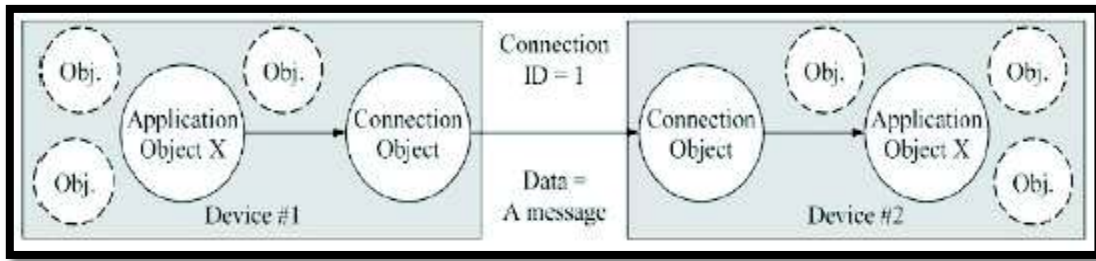


**Figura 1.3** Representación de Componentes de Objetos CIP [3]

### **Formato de Intercambio de Datos CIP**

CIP es propuesto como un protocolo de mensajería en el que se requiere primero establecer la conexión entre los nodos que intercambian información antes de comenzar con la transmisión de datos. Esta conexión que se realiza a través de los objetos definidos como de conexión permite la transmisión de información entre nodos y se identifica a través de un identificador numérico definido como “Conexión ID” o CID [3], [8].

En la Figura 1.4 se representa las conexiones para la transmisión de información entre diferentes nodos de una red.



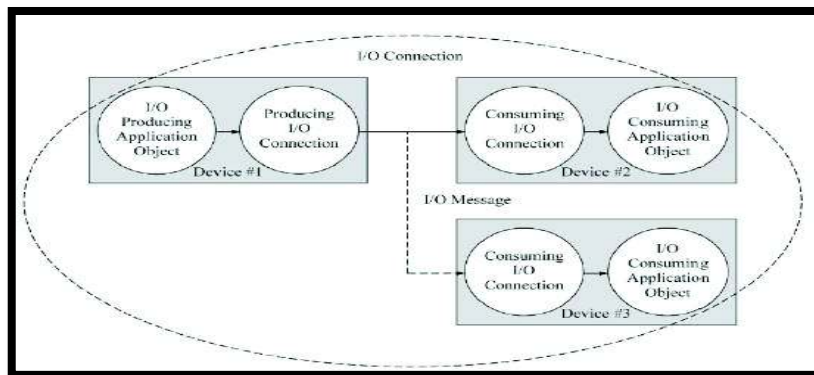
**Figura 1.4** Conexiones y Conexión ID CIP [3]

El formato al que se rige el CID depende del tipo de red. En el caso de una red DeviceNet implementa un formato en el que se rige al "IDENTIFIER" o identificador definido en el protocolo CAN, esto se debe a que DeviceNet implementa este protocolo en su capa de enlace de datos, esto será tratado en mayor detalle más adelante.

En lo referente al establecimiento mismo de la conexión, CIP lo propone de dos maneras [3], [4]:

- Conexiones de Mensajería I/O o Implícita.
- Conexiones de Mensajería Explícita.

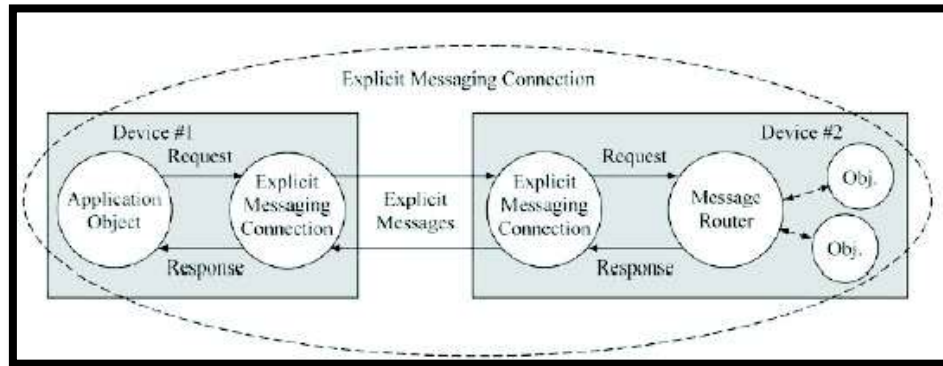
La Mensajería I/O o Implícita se emplean cuando se tiene una aplicación productor/consumidor que contienen datos I/O (entrada y salida). Son llamados de Mensajería Implícita ya que no requiere la información de dirección y/o servicio debido a que el nodo consumidor utiliza la información requerida basada en la Conexión ID. Utiliza un mensaje de tipo multicast. En la Figura 1.5 se muestran las conexiones para mensajería implícita utilizado por CIP.



**Figura 1.5** Conexión I/O CIP para Mensajería I/O multicast [3]

La mensajería explícita se emplea cuando se tiene una aplicación de tipo solicitud/encuesta y requiere la información de dirección y/o servicio que se solicita al dispositivo. Utiliza un mensaje de tipo punto a punto.

En la Figura 1.6 se muestran las conexiones para mensajería explícita utilizado por CIP.



**Figura 1.6** Conexión CIP para Mensajería Explícita punto a punto [3]

### **Configuración de Dispositivos CIP**

Al realizar la configuración de un dispositivo se puede manipular ciertos parámetros y características como la extensión del tamaño de variables que han sido preestablecidas por el fabricante del dispositivo y que deben modificarse acorde a los requerimientos de la red en que va a operar el dispositivo.

El protocolo CIP permite realizar la configuración de dispositivos que conforman la red por medio de tres métodos: [3]

- Hoja de Datos Impresa
- Objeto de Parámetro
- Hoja de Datos Electrónica

#### **a) Configuración por Hoja de Datos Impresa**

Esta herramienta de configuración proporciona la información de datos de servicio, clase, instancia y atributo para transmitir a un dispositivo. Para realizar esta configuración se debe manipular cada uno de los parámetros mencionados anteriormente y conocer el formato de envío y recepción de datos, es por esto que a pesar de obtener acceso a esta información este método no es el más adecuado pues si se cometen errores en el establecimiento de valores o formato no se lograría establecer la comunicación de los dispositivos.

## **b) Configuración por Objeto de Parámetro**

Este tipo de configuración si bien permite tener acceso a todos los parámetros de los datos del dispositivo impone un considerable tamaño de carga de información que resultaría ser excesiva en el caso de tener un dispositivo simple. Para realizar esta configuración se ve necesario que el usuario establezca valores en características que pueden no ser aceptadas por el dispositivo.

## **c) Configuración por Hoja de Datos Electrónica (EDS)**

Esta configuración permite obtener todos los parámetros a configurarse del dispositivo y adicionalmente se tiene acceso a la información de la conexión de I/O que corresponde a la extensión del tamaño de los datos de entrada y salida del dispositivo. La ventaja de este método es que no sobrecarga de información innecesaria al dispositivo y su implementación es sencilla ya que únicamente se puede cargar el archivo utilizando las herramientas disponibles en el software de configuración de red.

En el presente trabajo se utilizó la creación de un EDS debido a las ventajas que presenta y la facilidad de programación. Se describirá el procedimiento en la siguiente sección.

### **Hoja de Datos Electrónica (EDS)**

Un EDS es un archivo de texto escrito en ASCII. Puede ser generado en cualquier editor ASCII como es block de notas de Windows. Para realizar correctamente un EDS se debe seguir de manera ordenada un sistema de reglas y la sintaxis especificadas por CIP. De esta manera se puede llegar a realizar el EDS configurando la información más importante del dispositivo como son las conexiones I/O que soporta, valores de identificación del dispositivo asignados por el vendedor, tipo de datos que admite, entre otros.

En lo que respecta a la creación de un EDS son tres aspectos importantes que se deben tener en cuenta para crear un EDS [3]:

- El EDS se estructura por secciones.
- Al iniciar cada sección se debe escribir entre corchetes [ ] el nombre de la sección.
- Las secciones de "File" y "Device" son obligatorias para todo EDS, debido a que contiene la información que identifica al dispositivo.

A continuación, se detalla cada una de las secciones del EDS:

- [File]: Describe el contenido y la revisión del archivo.

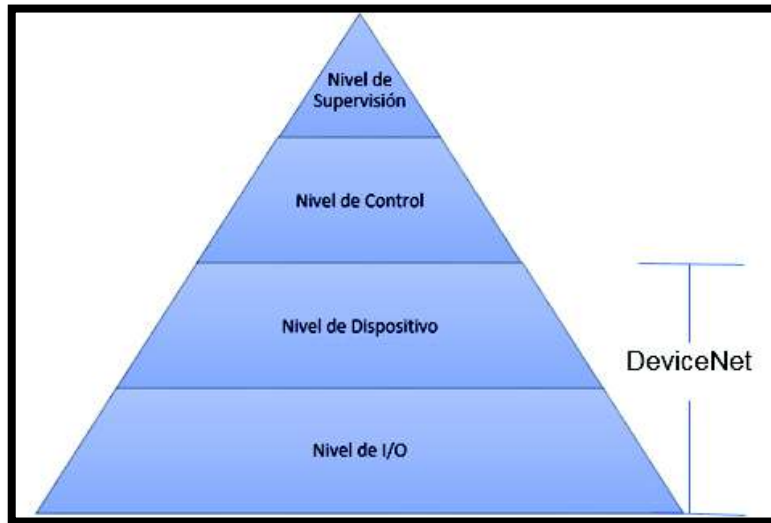
- [Device]: Es equivalente a la información de identificación del objeto y se utiliza para emparejar un EDS a un dispositivo.
- [Device Classification]: Describe con que red se puede conectar el dispositivo. Esta sección es opcional para DeviceNet, pero requerido para ControlNet y EtherNet/IP.
- [IO\_Info]: Describe los métodos de conexión y tamaños de E/S. Requerido para DeviceNet solamente.
- [Variant\_IO\_Info]: Describe múltiples configuraciones de datos IO\_Info. Requerido solo para DeviceNet.
- [ParamClass]: Describe atributos del nivel de clase del Objeto de Parámetro.
- [Params]: Identifica todos los parámetros de configuración en el dispositivo, sigue la definición del Objeto de Parámetro.
- [EnumPar]: Lista de enumeración de opciones de parámetro para presentar al usuario. Este es un método antiguo especificado solamente para DeviceNet.
- [Assembly]: Describe la estructura de los ítems de dato.
- [Groups]: Identifica a todos los grupos de parámetros en el dispositivo y nombre de grupo de listas y números de instancia del Objeto de Parámetro.
- [Connection Manager]: Describe las conexiones soportadas por el dispositivo. Utilizado típicamente en ControlNet y EtherNet/IP.
- [Port]: Describe los varios puertos de red que un dispositivo puede tener.
- [Modular]: Describe las estructuras modulares dentro de un dispositivo.
- [Capacity]: Especifica la capacidad de comunicación de los dispositivos ControlNet y EtherNet/IP.

## **Comunicación Industrial DeviceNet**

La red de campo DeviceNet fue desarrollada por la empresa Rockwell Automation en conjunto con la ODVA (Open DeviceNet Vendor Association).

DeviceNet es una red abierta que está implementada según el modelo de referencia OSI razón por la cual se encuentra estructurada de forma jerárquica en 7 capas, que se implementan sobre un bus industrial de tipo serial conectando los dispositivos que integrarán la red.

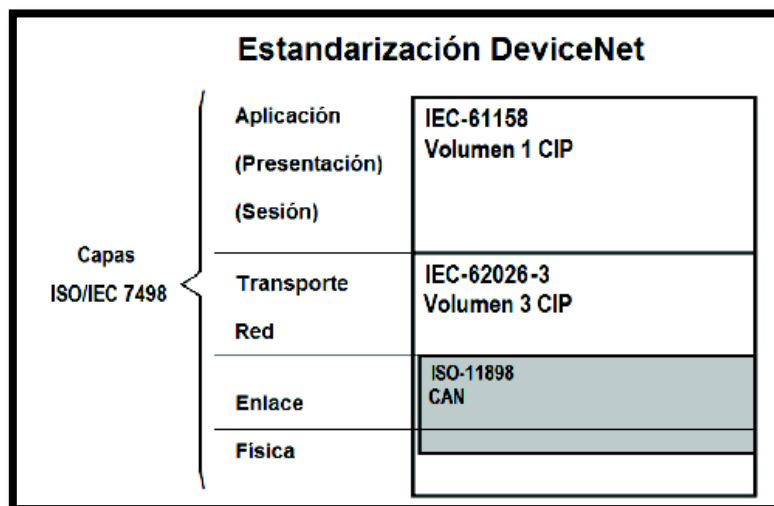
DeviceNet fue creado con la idea de ser un protocolo con funcionalidad media y de bajo costo para ser utilizado en la parte inferior de la arquitectura de un sistema de control industrial. Por este motivo una red DeviceNet conecta dispositivos como sensores, PLC, entre otros [3]. En la Figura 1.7 se muestran los niveles de la arquitectura de un sistema de control industrial y la localización de la red DeviceNet dentro de esta arquitectura.



**Figura 1.7** Arquitectura de un Sistema de Control Industrial

### **Estandarización de DeviceNet**

DeviceNet al basar su arquitectura en el modelo de referencia OSI (Open System Interconnection), estándar ISO/IEC 7498, su arquitectura queda definida en siete capas. En la Figura 1.8 se muestra el esquema de las normas que estandarizan a DeviceNet en cada una de estas capas.



**Figura 1.8** DeviceNet en Contexto de Estandarizaciones [3]

**a) ISO-11898**

El estándar ISO 11898 define el protocolo de comunicación CAN (Controller Area Network) consolidando así la capa de enlace y parcialmente la capa física. Este estándar define aspectos como temporización, sincronización de bit y codificación de la capa física.

**b) IEC 62026-3**

Es el estándar especificado para las capas 1 a 4 de la red DeviceNet. Como se ha mencionado antes el Volumen 3 de CIP hace una adaptación de su protocolo para incorporarlo a DeviceNet. Este estándar hace referencia a las especificaciones de los cables a usarse en una red DeviceNet. Es así, que se emplea este volumen con las especificaciones CAN de ISO 11898 completando la parte de la capa física que esta norma no define.

**c) IEC 61158**

Es la definición para buses de campo de CIP e incluye los volúmenes 1 y 4 de CIP. DeviceNet se compone de tres estándares internacionales siendo necesario para definir esta red el volumen 1 y 3 de CIP dado que el protocolo CAN está contenido en este último volumen. Este estándar hace referencia a la comunicación de datos digitales empleados para control distribuido en tiempo real y medición.

**Características Generales de la Red DeviceNet**

En la Tabla 1.2 se describen las principales características de una red DeviceNet [3]:

**Tabla 1.2** Principales Características de la red DeviceNet [3]

<b>Características</b>	<b>Descripción</b>
Topología	Línea Troncal/Derivaciones
Número de nodos	64
Longitud máxima de red	500m
Energización	24VDC sobre el mismo medio
Tasa de Transmisión	125,250 y 500 Kbps(según longitud de red)
Método de Acceso al Medio	CSMA/NBA(Arbitraje de Bus)
Paquetes de Datos	0-8 Bytes
Métodos de Comunicación	Productor/Consumidor (Strobe,Poll,COS,Cíclico)
Arquitectura de Control	Centralizada y Distribuida

## Arquitectura de la Red DeviceNet

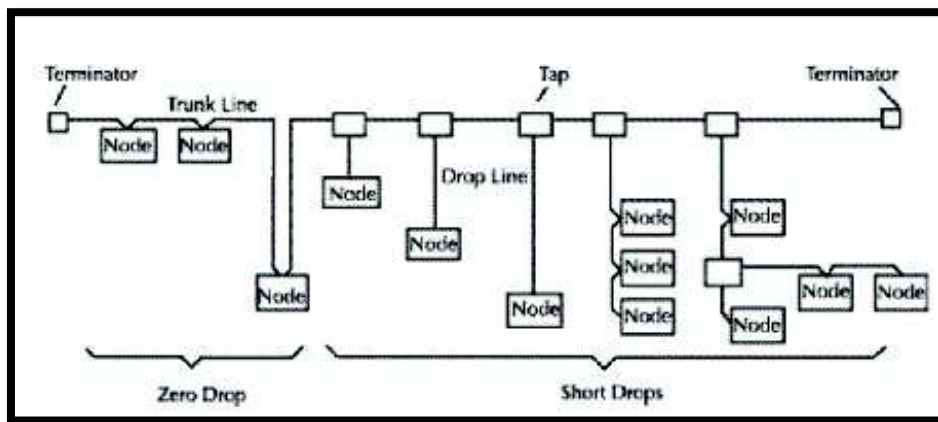
Como ya se mencionó en las secciones anteriores las capas físicas, enlace, red y transporte son definidas bajo el Volumen 3 de CIP. Las capas de aplicación formadas por sesión, presentación y aplicación son definidas en el Volumen 1 de CIP [3], [4].

### Capa Física DeviceNet [3]

#### Topología

Dado que DeviceNet implementa el protocolo CAN en su capa física, se hace referencia de este protocolo para describir características de DeviceNet. La topología que utiliza CAN es la de bus o línea troncal y en DeviceNet se define como Trunkline-Dropline (Línea Troncal con Derivaciones).

En la Figura 1.9 se evidencia la topología de una red DeviceNet.



**Figura 1.9** Topología Línea-Troncal/Derivaciones de DeviceNet [3]

#### Velocidad y Longitud de la Red

Existe una relación inversamente proporcional entre la velocidad de transmisión y la longitud de la red. Por esta razón DeviceNet especifica 3 velocidades de operación, que se muestran en la Tabla 1.3 con sus respectivos rangos de longitud de red.

**Tabla 1.3** Tasas de Transmisión DeviceNet y Distancias Máximas del Medio Físico [3]

Tasa de Transmisión	Longitud Línea Troncal			Longitud Derivación	
	Cable Thick	Cable Thin	Cable Plano Kw/Klink	Máxima	Acumulada
125 kbps	500 m	100 m	420 m	6 m	156 m
250 kbps	250 m		200 m		78 m
500 kbps	100 m		75 m		39 m



## Especificaciones Medio Físico

**Cables:** DeviceNet utiliza 3 tipos de cables: Round-Thick, Round-Thin y Flat. Los cables tipo Round están formados por 5 hilos conductores (CAN\_H, CAN\_L, DRAIN, V-, V+) y los cables tipo Flat están formados por 4 hilos conductores (CAN\_H, CAN\_L, V-, V+).

Los hilos conductores de DeviceNet se muestran en la Figura 1.10.



**Figura 1.10** Cable tipo Round DeviceNet

En la Tabla 1.4 se detalla la designación para los hilos de los cables empleados en una red DeviceNet.

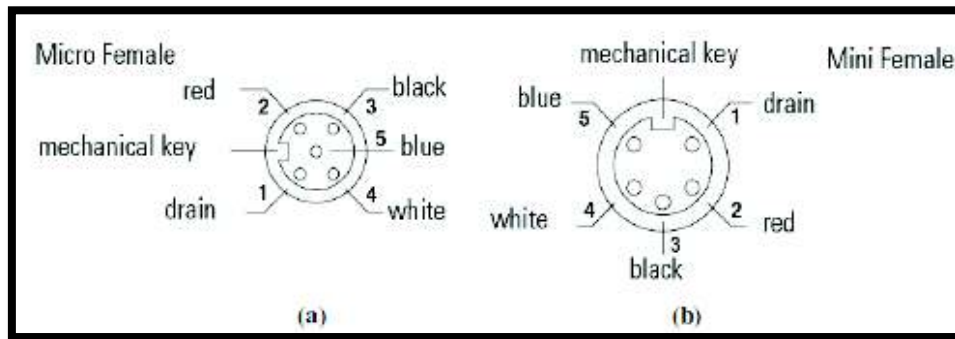
**Tabla 1.4** Hilos de cables DeviceNet [3]

Color del Alambre	Designación del Alambre	Uso
Blanco	CAN_H	Señal
Azul	CAN_L	Señal
Desnudo	Drain	Protección
Negro	V-	Energización
Rojo	V+	Energización

**Conectores:** DeviceNet utiliza dos tipos de conectores sellados y abiertos

- **Sellados:** Denominados así debido a que los hilos conductores en sus terminaciones tienen una carcasa tipo plug. Estos conectores son Mini-style y Micro-style.
  - ✓ **Mini-Style:** Utilizado para cables tipo Round -Thick y Round-Thin.
  - ✓ **Micro-Style:** Utilizado para cables tipo Round-Thin

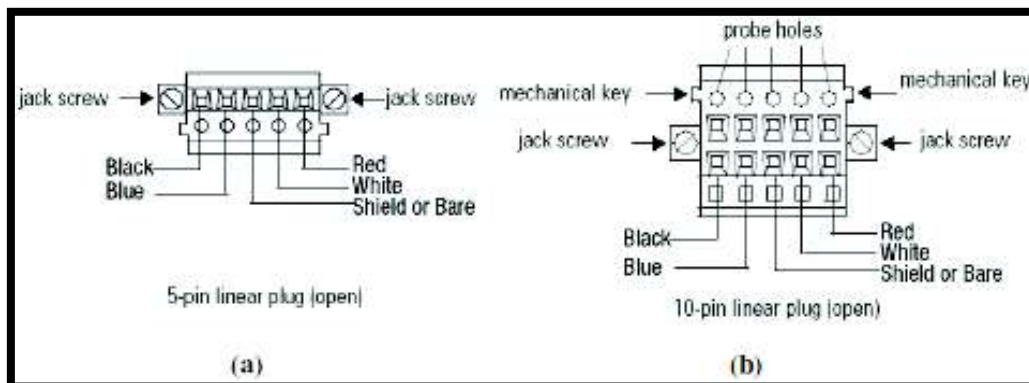
En la Figura 1.11 se muestran los conductores de tipo sellados.



**Figura 1.11** Conectores sellados DeviceNet (a)Micro-Style, y (b) Mini-Style [3]

- **Abiertos:** Denominados así debido a que los hilos conductores en sus terminaciones no tienen una carcasa. Los conectores a emplearse son Plug-IN y Fixed.
  - ✓ **Plug-In:** Utilizado para conectar los hilos conductores a un conector removible.
  - ✓ **Fixed:** Utilizado para conectar los hilos conductores a un terminal de tornillo fijo.

En la Figura 1.12 se muestran los conductores de tipo abierto.



**Figura 1.12** Tipos de conectores DeviceNet: (a) Conector abierto plug-in de 5 pines; (b) Conector abierto plug-in de 10 pines [3]

**Taps:** Son los puntos de la línea Troncal que tiene diferentes funciones como iniciar las derivaciones, entregar energía y conectar los nodos a la red. Entre los más importantes son T-Port, DeviceBox, PowerTap, DevicePort , entre otros.

En la Figura 1.13 se muestran los conectores de tipo Taps.



**Figura 1.13** Elementos para la Conexión sobre la Línea Troncal [3]

### **Capa de Enlace DeviceNet [3]**

#### **Método de Acceso al Medio**

DeviceNet implementa el protocolo CAN en su capa de enlace de datos. De esta forma queda definido el acceso al medio por CAN en donde el mecanismo de arbitraje para acceder al bus, ya sea para envío o recepción de datos, está definido por el método llamado CSMA/NBA (Carrier Sense Media Access/Not destructive Bit-wise Arbitration), en español se traduce como Acceso al Medio por Detección de Portadora/Arbitraje no Destructivo a Nivel de Bit y es el mecanismo que asegura que la información no se pierda debido a una colisión.

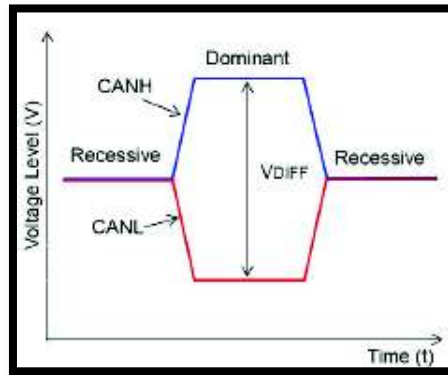
En este método de acceso al medio, cualquier nodo puede iniciar la transmisión de un mensaje siempre que el bus esté libre. Si dos nodos quieren transmitir simultáneamente el conflicto de acceso al bus se resuelve por el método de arbitraje de bit inteligente (bit-wise arbitration) usando el Identificador (Identifier) de la trama CAN.

#### **Niveles de Bus**

DeviceNet en la capa de enlace de datos utiliza los dos estados lógicos definidos en CAN “Recesivo” (1 lógico) y “Dominante” (0 lógico). El acceso al bus se determina a través de estos bits “Dominante” y “Recesivo”. El protocolo CAN realiza su transmisión de datos a través de dos señales denominadas CAN H (CAN High o CAN alto) y CAN L (CAN Low o CAN bajo) que forman un bus diferencial. El dispositivo puede seguir enviando siempre y cuando estos niveles sean iguales. Si un nivel “dominante” es detectado en el bus y llega la transmisión de un nivel “recesivo” entonces el dispositivo pierde el arbitraje y no puede enviar ningún bit más.

Estas dos señales se transmiten a través de dos conductores diferentes y se transmiten simultáneamente de un modo diferencial, es decir, portan voltajes invertidos disminuyendo el ruido.

En la Figura 1.14 se muestran los niveles de voltajes asignados en el protocolo CAN.



**Figura 1.14** Niveles de Voltajes CAN [3]

### Trama CAN

CAN define 4 tipos de tramas para la transmisión de datos y son las siguientes:

- **Trama de Dato** (Data Frame): Porta datos desde un transmisor a los receptores.
- **Trama Remota** (Remote Frame): Es transmitido por una unidad del bus para requerir la transmisión de la Trama de Dato con el mismo Identificador.
- **Trama de Error** (Error Frame): Es transmitido por cualquier unidad que haya detectado un error en el bus.
- **Trama de Sobrecarga** (Overload Frame): Es usado para proveer un retardo extra entre el precedente y la siguiente Trama de Dato o Trama Remota.

El protocolo DeviceNet utiliza sólo Trama de Datos (Data Frame) de CAN por tal motivo se lo detalla a continuación.

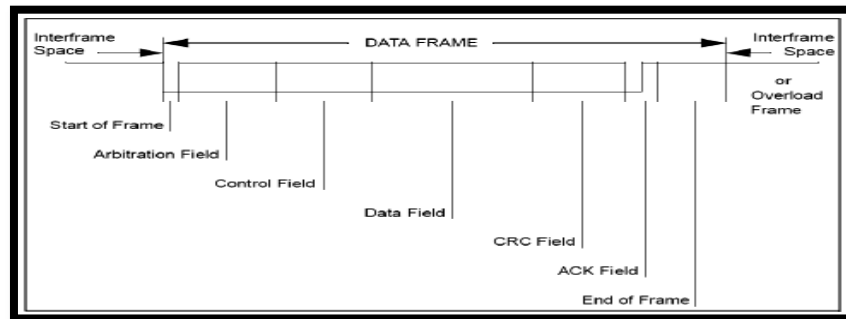
### Trama de Datos [2]

La Trama de Datos (Data Frame) contiene los datos a ser transferidos de un dispositivo a uno o varios receptores. Está conformado por 7 diferentes campos de bits:

- **Campo de Inicio de Comunicación** (Start of Frame)
- **Campo de Arbitraje de Trama** (Arbitration Frame)
- **Campo de Control** (Control Field)

- **Campo de Dato** (Data Field)
- **Campo de Chequeo Redundante de Errores** (CRC Field)
- **Campo de Chequeo de Errores** (ACK Field)
- **Campo de Fin de Comunicación** (End of Frame)

En la Figura 1.15 se muestran los campos que forman la trama de datos.



**Figura 1.15** Trama de Datos CAN [2]

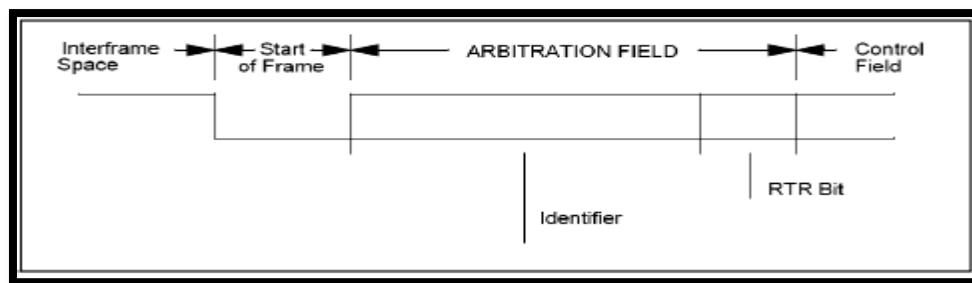
**a) Campo de Inicio de Comunicación (1 bit)**

El campo de inicio de comunicación indica el inicio de la transmisión, un nodo que forma parte de la red puede transmitir únicamente si el bus está desocupado, es decir depende del valor que se encuentre en este bit. Se inicia la transmisión con este bit como dominante (valor 0).

**b) Campo de Arbitraje de Trama (12 bits)**

El campo de arbitraje es el encargado de resolver por medio de arbitraje el conflicto para el acceso al medio. Está compuesto por dos campos Identificador o "Identifier" y "RTR-Bit".

En la Figura 1.16 se muestra el campo de arbitraje que forma parte de la trama de datos.



**Figura 1.16** Campo de Arbitraje de la Trama de Datos [2]

- **El Identificador**

El "Identifier" o Identificador, posee una longitud de 11 bits y como su nombre lo indica permite identificar el mensaje y la prioridad que tiene al momento de la transmisión de información.

- **El RTR Bit**

El RTR Bit (Remote transmisión Request Bit) permite diferenciar entre una trama de datos (valor 0) o una trama remota (valor 1).

**c) Campo de Control (6 bits)**

El campo de control permite conocer el número de bytes de datos que contiene el campo de datos que le sigue a continuación.

**d) Campo de Dato (8 bits)**

El campo de dato está formado por los datos a transferirse en el bus, puede transmitir hasta 8 bytes.

Los datos enviados en este campo varían dependiendo de la etapa de comunicación, y de las que se pueden identificar dos. La primera etapa consiste en que el dispositivo se encuentra preestableciendo la comunicación y la segunda etapa se presenta cuando la comunicación se estableció con éxito.

En la etapa de pre-establecimiento de comunicación el campo de datos inicialmente tiene el siguiente formato:

- **Código de Servicio:** Indica el valor numérico que representa la red a la que pertenece el dispositivo.
- **Class ID:** Indica el valor numérico que representa el identificador de clase.
- **Instancia ID:** Indica el valor numérico que representa el identificador de instancia dentro de una clase.
- **Elección de Asignación:** Indica el valor numérico que representa el tipo de mensaje que envía el dispositivo.
- **Asignación MAC-ID:** Indica el valor numérico que representa el dispositivo que envía el mensaje.

En la etapa de comunicación establecida, el campo de datos contiene el dato enviado por el dispositivo.

**e) Campo de Chequeo Redundante de Errores (16 bits)**

El campo de chequeo redundante de errores está formado por 16 bits, 15 de ellos definidos para el código de redundancia cíclica y 1 bit delimitador CRC.

**f) Campo de Chequeo de Errores (2 bits)**

El campo de chequeo de errores se compone de 2 bits el bit ACK y el delimitador ACK. El nodo transmisor envía este bit como recesivo al nodo receptor, y el nodo receptor envía este bit como dominante para que éste reporte si ha recibido el mensaje correctamente.

**g) Campo de Fin de Comunicación (7 bits)**

El campo de fin de comunicación está formado por 7 bits e indica que ha terminado la comunicación.

**Capa de Red y Transporte DeviceNet [3]**

**Identificador CAN**

El campo de Identificador utiliza 11 bits, permitiendo identificar así  $2^{11} = 2048$  mensaje.

El identificador de CAN o Conexión ID (Connection ID) está conformado por dos partes:

- MAC-ID (Media Access Control)
- Mensaje-ID (Message ID)

El MAC-ID está formado por 6 bits e indica la dirección de nodo sea de un dispositivo fuente o destino dentro de la red. La cantidad de nodos que son admitidos en DeviceNet es de 64.

El Mensaje-ID identifica el mensaje y no siempre es de la misma extensión ya que depende del grupo de mensaje al que corresponda. DeviceNet divide el rango de Conexión-ID en 4 grupos de mensajes.

En la Tabla 1.5 se muestra la distribución de los 4 grupos de mensajes definidos por DeviceNet.

**Tabla 1.5** Grupos de Mensajes DeviceNet [3]

Conexión ID = Identificador CAN (bits 10:0)										Tipo de Mensaje			
10	9	8	7	6	5	4	3	2	1	0			
0	Mensaje ID				Fuente MAC ID						Mensaje Grupo 1		
1	0	MAC ID					Mensaje ID					Mensaje Grupo 2	
1	1	Mensaje ID			Fuente MAC ID						Mensaje Grupo 3		
1	1	1	1	1	Mensaje ID						Mensaje Grupo 4		
1	1	1	1	1	1	1	x	x	x	x	Identificador CAN Inválido		

El Conexión-ID en DeviceNet está formado por Grupo de mensajes-ID, el Mensaje-ID dentro de este grupo y el MAC-ID del dispositivo. El MAC-ID puede contener la dirección de fuente o destino del dispositivo.

### Grupos de Mensajes DeviceNet

DeviceNet define 4 Grupos de Mensajes los cuales se describen a continuación.

#### a) Mensaje Grupo 1

Este grupo permite el intercambio de información para datos con alta prioridad. La prioridad para el mensaje se da primero por el Mensaje-ID y después por MAC-ID, es decir, si dos dispositivos quieren realizar la transmisión de datos simultáneamente entonces el dispositivo con más bajo Mensaje-ID gana el arbitraje. Si dos dispositivos envían el mismo Mensaje-ID al mismo tiempo al bus CAN entonces el dispositivo con menor MAC-ID gana el arbitraje.

#### b) Mensaje Grupo 2

Este grupo difiere del primero ya que está orientado al llamado “Predefined Master/Slave Connection Set” o Conjunto de Conexión Predefinida Maestro/Esclavo, es decir, es uno de los métodos que permiten establecer comunicación entre un dispositivo maestro y un dispositivo esclavo. La prioridad en este grupo se determina primero por MAC-ID y luego por Mensaje-ID.

#### c) Mensaje Grupo 3

Este grupo es similar al Grupo 1 con la diferencia que está destinado a realizar el intercambio de información para datos de baja prioridad. Este grupo se usa principalmente para Conexiones Explícitas Dinámicas que se caracterizan por llevar la información de dirección y servicio de un dispositivo. De los 7 mensajes posibles por dispositivo, 2 son



reservados a lo que es conocido como “Unconnected Message Manager” (UCMM) o Gestor de Mensajería No Conectada.

#### **d) Mensaje Grupo 4**

Los mensajes en este grupo al no incluir MAC-ID son utilizados como gestores de red.

#### **Establecimiento de la Conexión**

DeviceNet es un protocolo orientado a conexión, es decir que para realizar el intercambio de información primero tiene que establecer la conexión entre los dispositivos que forman la red para que se pueda transmitir y recibir datos. Para lograr esta conexión se determina un camino entre dos puntos externos y es posible realizarlo a través de los “Communication Objects” u Objetos de Comunicación.

Los Objetos de Comunicación inicialmente no están encendidos en los dispositivos ya que deben ser creados. Cuando un dispositivo es encendido por primera vez los únicos puertos que están disponibles son el Gestor de Mensajería No Conectada (UCMM) o el puerto Conjunto de Conexión Predefinida Maestro/Esclavo (Group 2 Only Unconnected Request Predefined Master/Slave Connection Set).

La comunicación a través del puerto UCMM para un dispositivo necesita de una alta capacidad de procesamiento para lograr cumplir con todos los requerimientos de comunicación es por este motivo que se creó el llamado Conjunto de Conexión Predefinida Maestro/Esclavo (Predefined Master/Slave Connection Set) que puede ser aplicado en dispositivos de bajas capacidades para lograr una comunicación simplificada y el intercambio de datos de entrada y salida.

En el presente trabajo se utiliza el Conjunto de Conexión Predefinida Maestro/Esclavo debido a sus ventajas entre las que se puede mencionar que es un método que utiliza menos recursos de CPU siendo ideal en dispositivos de baja capacidad de procesamiento como lo son sensores y actuadores, este se lo describe a continuación.

#### **Conjunto de Conexión Predefinida Maestro/Esclavo**

Este modelo llamado Conjunto de Conexión Predefinida Maestro/Esclavo (Predefined Master/Slave Connection Set) tiene una comunicación con estructura 1:n, esto quiere decir 1 dispositivo de control a “n” dispositivos de entrada y salida descentralizados . El “Master” o Maestro es el dispositivo central del sistema y los dispositivos descentralizados son “Slaves” o Esclavos.

Los Objetos de Conexión Predefinidos ocupan instancias de 1 a 5 en el Objeto de Conexión (Clase ID 0x05) [3], y separados en conexiones de mensajería Explícita y Conexiones de Mensajería I/O.

- Conexión de Mensajería Explícita (Explicit Messaging Connection):
  - ✓ Mensaje de Solicitud/Respuesta Explícita del Grupo 2 (Instancia ID 1)
- Conexiones de Mensajería I/O (I/O Messaging Connections):
  - ✓ Conexión E/S tipo Encuesta (Instancia ID 2)
  - ✓ Conexión E/S tipo Bit-Estroboscópico (Instancia ID 3)
  - ✓ Conexión E/S tipo Cambio de Estado o Cíclico (Instancia ID 4)
  - ✓ Conexión Multicast tipo Encuesta (Instancia ID 5)

En la Tabla 1.6 se muestra la distribución de conexiones ID para DeviceNet.

**Tabla 1.6** Distribución de las Conexiones ID definidas en DeviceNet del Conjunto de Conexión Predefinida Maestro/Esclavo [3]

Conexión ID = Identificador CAN (bits 10:0)											Descripción del Mensaje
10	9	8	7	6	5	4	3	2	1	0	
0	Grupo 1 Mensaje ID				Fuente MAC ID						Mensajes del Grupo 1
0	1	1	0	0	Fuente MAC ID						Respuesta E/S Multicast tipo Encuesta (Poll) del Esclavo.
0	1	1	0	1	Fuente MAC ID						Mensaje E/S tipo Cambio de Estado (COS) o Cíclico (Cyclic) del Esclavo.
0	1	1	1	0	Fuente MAC ID						Mensaje de Respuesta E/S tipo Bit-Estroboscópico (Bit-Strobe) del Esclavo.
0	1	1	1	1	Fuente MAC ID						Mensaje ACK de respuesta E/S tipo Encuesta o Cambio de Estado/Cíclico del Esclavo.

Continuación de la Tabla1.6.

1	0	MAC ID	Grupo 2 Mensaje ID			Mensajes del Grupo 2
1	0	Fuente MAC ID	0	0	0	Mensaje de Comando E/S tipo Bit-Estroboscópico del Maestro.
1	0	Fuente MAC ID	0	0	1	Grupo ID E/S Multicast tipo Encuesta del Maestro.
1	0	Destino MAC ID	0	1	0	Mensaje ACK tipo Cambio de Estado o Cíclico del Maestro.
1	0	Fuente MAC ID	0	1	1	Mensaje de Respuesta Explícita/Desconexión del Esclavo.
1	0	Destino MAC ID	1	0	0	Mensaje de Solicitud Explícita del Maestro.
1	0	Destino MAC ID	1	0	1	Mensaje de comando de E/S tipo Encuesta o Cambio de Estado/Cíclico del Maestro.
1	0	Destino MAC ID	1	1	0	Mensaje de Solicitud Explícito únicamente del Grupo 2 de desconexión.
1	0	Destino MAC ID	1	1	1	Mensaje de Chequeo de MAC ID Duplicada.

Como se evidencia en la tabla anterior la mayoría de los mensajes del Master contienen la dirección de destinatario Slave por ese motivo es importante que sólo un Master se comunique a cualquier Slave.

El Master asigna un Conjunto de Conexión Predefinida Maestro/Esclavo previniendo así la duplicación del CAN-ID.

Los servicios usados son Conexión de Asignación Maestro/Esclavo (Código de Servicio 0x4B) y Conjunto de Identificación de Lanzamiento Grupo 2 (Código de Servicio 0x4C).

La Tabla 1.7 muestra el formato de la conexión de asignación Maestro/Esclavo en donde se puede observar que los dos servicios utilizan la instancia 1 del Objeto DeviceNet (Clase ID 0x03)

**Tabla 1.7** Formato de Conexión de Asignación Maestro/Esclavo (Código de Servicio 0x4B [3])

Offset de Bytes \ Número de Bit	Número de Bit									
	7	6	5	4	3	2	1	0		
0	Frag [0]	XID	MAC ID [0x00]					Encabezado del Mensaje		
1	R/R [0]	Código de Servicio [0x4B]						Cuerpo del Mensaje		
2..5	Clase ID [0x03]									
	Instancia ID [0x01]									
	Elección de Asignación									
	0	0	Asignación MAC ID							

Cuando se utiliza el Conjunto de Conexión Predefinida Maestro/Esclavo para realizar la conexión de dispositivos el Mensaje de Asignación y la Instancia ID utiliza un formato de 8 bits cada uno respectivamente.

El campo de Elección de Asignación se utiliza para determinar el tipo de conexión predefinida.

La Tabla 1.8 muestra el formato del byte de elección de asignación.

**Tabla 1.8** Formato del byte Elección de Asignación [3]

Número de Bit	Descripción
0	Mensaje Explícito
1	Encuesta
2	Bit-Estroboscópico
3	Votación Multicast
4	Cambio de Estado
5	Cíclico
6	Ack Supresión
7	Reservado

## **Secuencia de Inicialización Maestro/Esclavo**

Una red DeviceNet con un maestro y un conjunto de esclavos se pone en marcha de acuerdo a la siguiente lógica.

- Todos los elementos que conforman la red inicializan su secuencia de autodetección e intentan mantenerse en línea.
- Cuando los esclavos se encuentran en línea van a esperar hasta que el Master realice su asignación.
- Estando en línea el Master intentará comunicarse con cada esclavo que esté configurado en su lista de escaneo siguiendo la secuencia de mensajes:
  - ✓ Intenta realizar la conexión de los esclavos utilizando el UCMM.
  - ✓ Si es exitoso, el maestro utiliza esta conexión para seguirse comunicándose con el esclavo.
  - ✓ Si no es exitosa esta conexión el master espera un tiempo de 1 segundo como mínimo para empezar la conexión nuevamente.
  - ✓ Si de manera consecutiva, por al menos 3 intentos, no es exitosa esta conexión, el Master intentará comunicarse con el esclavo utilizando el Conjunto de Conexión Predefinida Maestro/Esclavo
  - ✓ Si es exitosa, el maestro utiliza esta conexión para seguir comunicándose con el esclavo.
  - ✓ Si no es exitosa esta conexión el maestro espera un tiempo de 1 segundo como mínimo para empezar la conexión nuevamente utilizando el UCMM.
- Cuando el maestro ha logrado comunicarse con el esclavo, el esclavo es el encargado de realizar la verificación de esta conexión para a continuación iniciar la mensajería de datos de entrada y salida de los dispositivos.

## **Herramientas de Software a Usarse**

Para el desarrollo del presente trabajo se utiliza las siguientes herramientas de software:

- RSLogix5000
- RSNetWorx
- RSLinx

- USB-CAN Tool V2.02
- FactoryTalk View Studio

### **RSLogix5000**

Permite realizar la lógica de programación del PLC CompactLogix en diferentes lenguajes como son Ladder, FBD y SFC. Este software permite además realizar las siguientes acciones:

- “Download” y “Upload” del programa que se encuentre en una plataforma de control.
- Forzar entradas y salidas.
- Acceder a la configuración de los módulos conectados al sistema.
- Realiza el diagnóstico de fallas del procesador.
- Permite realizar la configuración de comunicación con el módulo PLC.
- Brinda un menú de ayuda para el usuario de las diferentes operaciones que se puedan necesitar al desarrollar una lógica de programación.



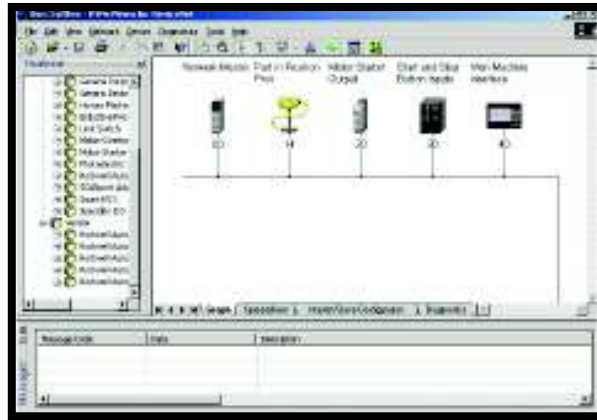
**Figura 1.17** Pantalla de inicio del programa RSLogix5000

### **RSNetWorx para DeviceNet**

Esta herramienta de software permite realizar el monitoreo y configuración de los dispositivos que forman la red DeviceNet. Además, permite la manipulación de los parámetros y características de los dispositivos de la red al proveer un acceso al usuario para configuración de la hoja electrónica de datos.

Las configuraciones que se pueden realizar en los dispositivos son:

- Asignación de nombre y descripción de un dispositivo.
- Direccionamiento de nodos.
- Auto detección de velocidad de transmisión de datos.
- Detección de fallos



**Figura 1.18** Pantalla de inicio del programa RSNetWorx para DeviceNet

La comunicación entre la PC con el programa RSNetWorx y la red DeviceNet se puede realizar por comunicación EtherNet/IP o añadiendo un interfaz de comunicación en la PC para ser configurado como un nodo más de la red.

El servidor de comunicación RSLinx será el encargado de realizar la conexión entre RSNetWorx y la red DeviceNet.

## **RSLinx**

Esta herramienta de software es un servidor de comunicación que permite realizar el enlace entre una aplicación del entorno de Windows y una red de la familia CIP. Entre sus principales funciones se puede mencionar las siguientes:

- Permite monitorear los estados de los dispositivos de la red.
- Proporciona una interfaz gráfica que facilita la navegación por la red.
- Permite realizar la configuración y monitoreo de dispositivos.



Figura 1.19 Pantalla de inicio del programa RSLinx

### USB-CAN Tool V2.02

Esta herramienta de software es un analizador de trama que permite al usuario conocer los datos que son intercambiados entre los dispositivos que forman una red.

A continuación, se nombran las acciones que se pueden realizar con este software:

- Permite el acceso a información como campo de identificación y tamaño de datos
- Permite configurar la velocidad de transmisión y el canal por el cual se obtendrán los datos.
- Posee diferentes modos de operación que permiten visualizar la información según requiera el usuario.
- Permite realizar el filtrado de información de los dispositivos basado en el campo de identificación.



Figura 1.20 Pantalla inicial del Software USB-CAN Tool [1]



## FactoryTalk View Studio

Esta herramienta de software permite realizar aplicaciones de interface operador-máquina (HMI) para computadoras personales como para PanelView's.

Esta herramienta posibilita al usuario realizar diferentes tipos de interfaces gráficas que permita la visualización de la aplicación en una o varias PCs (Site Edition Local y Site Edition Network) y en paneles touch (Machine Edition).

Este software permite establecer niveles de seguridad para autorizar o restringir el acceso a determinadas secciones de la interfaz de acuerdo al usuario.

Posee una amplia librería de instrumentos utilizados a nivel industrial. Además se puede realizar la migración de programas realizados en RSVIEW 32 y FactoryTalk View SE a FactoryTalk View ME y viceversa.

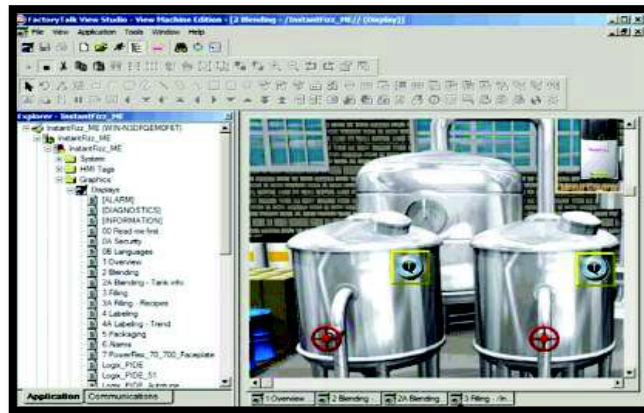


Figura 1.21 Pantalla inicial del Software FactoryTalk View Studio

## 2. METODOLOGÍA

En este trabajo se realiza una investigación aplicada ya que se detallará el proceso para el diseño de los drivers de comunicación para sensores analógicos y discretos para una red DeviceNet. Inicialmente se determina las características particulares de comunicación de dispositivos en la red DeviceNet basándose en el análisis del funcionamiento de un sensor comercial fotoeléctrico RightSight DeviceNet PHOTOSWITCH con modo de censado difuso y un PLC CompactLogix con módulo escáner DeviceNet 1769-SDN.

Tras el análisis se determina que para establecer la comunicación entre ellos es conveniente utilizar el Conjunto de Conexión Predefinida Maestro/Esclavo, tras lo cual se puede establecer los requerimientos de hardware y software para realizar la implementación de los drivers de comunicación para sensores analógicos y discretos a una red industrial DeviceNet.

### 2.1 Análisis de la Comunicación en Redes DeviceNet

Antes de establecer las condiciones y el formato de comunicación se inicia describiendo las características del escáner DeviceNet 1769-SDN integrado en el PLC CompactLogix y el sensor fotoeléctrico “RightSight DeviceNet PHOTOSWITCH”, que son los dispositivos en base a los cuales se analiza la comunicación DeviceNet.

#### Escáner DeviceNet 1769-SDN

El escáner es un módulo que puede conectarse a un PLC CompactLogix para facilitar la comunicación con dispositivos DeviceNet, pudiendo comunicarse hasta con 63 dispositivos. La fuente de alimentación eléctrica puede variar de 5 VDC a 24 VDC dependiendo estos límites del número de catálogo de la fuente. La velocidad y número de nodo vienen preestablecidos en valores de 125 kbps y 63 respectivamente. Tiene una especificación de distancia no mayor a cuatro módulos de la fuente de alimentación.

El módulo escáner 1769-SDN organiza sus entradas y salidas en dos formas: imagen de entrada e imagen de salida. Estas imágenes son espacios de memoria estructurados en forma de matrices de palabras que permiten la transferencia de datos, informar sobre los estados de la red con el escáner y realizar comandos entre el escáner y el PLC. Este módulo debe ser instalado y habilitado mediante configuración del PLC al que será conectado [2].

La imagen de entrada está formada por una matriz de 246 palabras y la imagen de salida está formada por una matriz de 182 palabras [2].

En las Tablas 2.1 y 2.2 se respresentan las imágenes de entrada y salidas del módulo escáner 1769-SDN.

**Tabla 2.1** Imagen de Entrada del Escáner 1769-SDN [2]

Palabra	Descripción	Tipo de Dato
0 a 65	Estructura de Estado	Matriz de 66 palabras
66 a 245	Entradas Esclavas DeviceNet	Matriz de 180 palabras

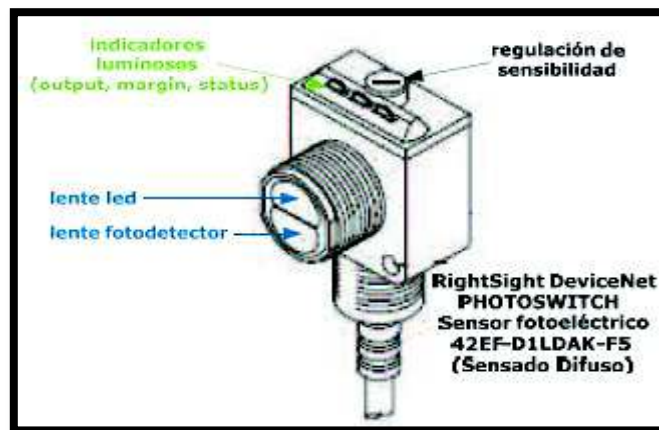
**Tabla 2.2** Imagen de Salida del Escáner 1769-SDN [2]

Palabra	Descripción	Tipo de Dato
0 a 1	Matriz de Comandos del Módulo	Matriz de 2 palabras
2 a 181	Salidas Esclavas DeviceNet	Matriz de 180 palabras

### Sensor Fotoeléctrico “RightSight DeviceNet PHOTOSWITCH”

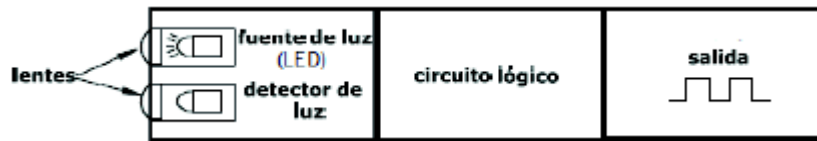
El sensor fotoeléctrico permite detectar la ausencia o presencia de un objeto debido a un cambio de luz, es decir, el objeto refleja el haz de luz emitido hacia la fuente y de esta manera es detectado por el fotodetector. Este sensor es de tipo infrarrojo y trabaja a una longitud de onda de 880nm [2].

En la Figura 2.1 se describen las partes del sensor fotoeléctrico “RightSight DeviceNet PHOTOSWITCH”



**Figura 2.1** Elementos del sensor fotoeléctrico [2]

El sensor está formado por 5 componentes: una fuente de luz, un detector de luz, dos lentes, un circuito lógico y una salida los mismos que se muestran en la Figura 2.2.



**Figura 2.2** Componentes del sensor fotoeléctrico [2]

El sensor realiza la transmisión de su dato empleando 1 byte, el mismo que será enviado al escáner utilizando el bit 0 de este byte y que corresponde a la detección del objeto.

En la Tabla 2.3 se muestra la distribución del byte de información utilizado por el sensor.

**Tabla 2.3** Registro del Byte de Información del Sensor [2]

Dato (Byte)	Valor de Bit	
	0	1
<b>Bit 0:</b> Salida Sensor	OFF	ON
<b>Bit 1:</b> Diagnostico	OK	Alarma
<b>Bit 2:</b> Margen de Diagnostico 1	OK	Margen Inestable
<b>Bit 3:</b> Margen de Diagnostico 2	OK	Margen Inestable
<b>Bit 4:</b> Salida de Movimiento	Movimiento	No Movimiento
<b>Bit 5:</b> Salida de Conta dor	Menos de lo Preestablecido	Preestablecido Alcanzado
<b>Bit 6:</b> No Utilizado		
<b>Bit 7:</b> No Utilizado		

### **Análisis de la Interacción Escáner/Sensor**

El sensor fotoeléctrico RightSight DeviceNet PHOTOSWITCH permitirá analizar los datos que envía al escáner para de esta manera determinar la forma en que se realiza la conexión entre ellos. Se configura el escáner y el sensor en el software RSNetWorx asignándoles una dirección de nodo y especificando una localidad de memoria en el escáner para que el sensor pueda ser reconocido dentro de la red DeviceNet.

Se conecta una PC con Software USB-CAN Tool V2.02 para monitorear las tramas en el bus de la red DeviceNet y obtener los datos de la misma. Los siguientes datos mostrados en la Tabla 2.4 se obtuvieron de la red sin conectar el sensor:

**Tabla 2.4** Datos de la red sin ningún elemento conectado

Frame ID	Tipo de Trama	Formato de Trama	Tamaño de Dato	Dato
0x0780	Dato	Estándar	0x04	30 4b 02 34
0x044E	Dato	Estándar	0x06	00 4b 03 01 01 00
0x0576	Dato	Estándar	0x06	00 4b 03 01 01 00
0x0780	Dato	Estándar	0x04	31 4b 02 34
0x0780	Dato	Estándar	0x04	30 4b 02 34
0x044E	Dato	Estándar	0x06	00 4b 03 01 01 00
0x0585	Dato	Estándar	0x06	00 4b 03 01 01 00

Seguidamente se procede a conectar el sensor obteniéndose los datos mostrados en la Tabla 2.5 mediante el analizador de trama:

**Tabla 2.5** Datos de la red con el sensor conectado

Frame ID	Tipo de Trama	Formato de Trama	Tamaño de Dato	Dato
0x044E	Dato	Estándar	0x06	00 4B 03 01 01 00
0x044F	Dato	Estándar	0x07	00 01 00 2B F5 0A 50
0x044E	Dato	Estándar	0x06	00 4B 03 01 01 00
0x044B	Dato	Estándar	0x04	00 CB 00
0x044C	Dato	Estándar	0x06	00 4B 03 01 10 00
0x044B	Dato	Estándar	0x03	00 CB 00
0x044C	Dato	Estándar	0x05	00 0E 01 01 01
0x044B	Dato	Estándar	0x04	00 8E 01 00
0x044C	Dato	Estándar	0x05	00 0E 01 01 02
0x044B	Dato	Estándar	0x04	00 8E 06 00
0x044C	Dato	Estándar	0x05	00 0E 01 01 03
0x044B	Dato	Estándar	0x04	00 8E 2C 00
0x044C	Dato	Estándar	0x05	00 0E 01 01 04
0x044B	Dato	Estándar	0x04	00 8E 01 06
0x044C	Dato	Estándar	0x06	00 10 05 01 0C 03
0x044B	Dato	Estándar	0x02	00 90

Continuación de la Tabla 2.5.

0x044C	Dato	Estándar	0x06	00 10 2B 01 01 10 00
0x044B	Dato	Estándar	0x04	00 90 10 00
0x044C	Dato	Estándar	0x07	00 10 05 04 11 01 00
0x044B	Dato	Estándar	0x04	00 90 08 00
0x044C	Dato	Estándar	0x05	00 0E 05 02 07
0x044B	Dato	Estándar	0x04	00 94 16 FF
0x044C	Dato	Estándar	0x05	00 0E 05 04 07
0x044B	Dato	Estándar	0x04	00 8E 01 00
0x044C	Dato	Estándar	0x05	00 0E 05 02 08
0x044B	Dato	Estándar	0x04	00 94 16 FF
0x044C	Dato	Estándar	0x05	00 0E 05 04 08
0x044B	Dato	Estándar	0x04	00 8E 00 00
0x044C	Dato	Estándar	0x07	00 10 05 04 09 FA 00
0x044B	Dato	Estándar	0x04	00 90 00 01

En la tabla anterior se puede evidenciar que el campo de identificación corresponde al rango de 400-5ff, el cual pertenece al Grupo de Mensajes 2 Conjunto de Conexión Predefinida Maestro/Esclavo (Group 2 Only Unconnected Request de Predefined Master/Slave Connection Set), por lo cual para el desarrollo de los drivers propuestos se usará este grupo de mensajes.

## **Determinación de Formato de Trama de Datos**

### **Análisis de Campo de Identificación**

Como se explicó anteriormente el método para establecer la comunicación entre Maestro/Esclavo es a través del Conjunto de Conexión Predefinida Maestro/Esclavo, por lo cual el mismo se detalla a continuación.

En la Tabla 2.6 se describe el campo de identificación de DeviceNet.

**Tabla 2.6** Definición del Campo Identificación DeviceNet CAN [3]

Conexión ID = Identificador CAN (bits 10:0)										Rango Hexadecimal	Tipo de Mensaje	
10	9	8	7	6	5	4	3	2	1			0
0	Mensaje ID				Fuente MAC ID						000-3FF	Mensaje Grupo 1
1	0	MAC ID					Mensaje ID				400-5FF	Mensaje Grupo 2
1	1	Mensaje ID			Fuente MAC ID						600-7BF	Mensaje Grupo 3
1	1	1	1	1	Mensaje ID						7C0-7EF	Mensaje Grupo 4
1	1	1	1	1	1	1	x	x	x	x	7F0-7FF	Identificador CAN Inválido

Como se observa en la Tabla 2.5 existen los siguientes campos de identificación pertenecientes a este grupo, los cuales se detallan en la Tabla 2.7.

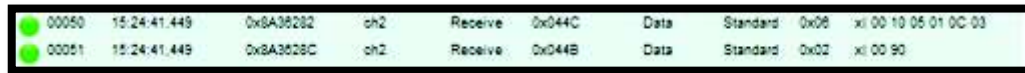
**Tabla 2.7** Campo de Identificación perteneciente al Mensaje de Solicitud Explícito del Grupo 2 de desconexión y Chequeo MAC ID Duplicada.

Frame ID [Hexadecimal]	Frame ID [Binario]	Grupo de Mensaje	MAC ID	Número de Mensaje	Interpretación del Mensaje
0x044E	10001001110	10	001001	110	Mensaje de Solicitud Explícito únicamente del Grupo 2 de desconexión.
0x044F	10001001111	10	001001	111	Mensaje de Chequeo de MAC ID Duplicada.

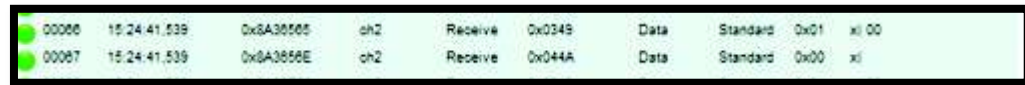
El campo de identificación de la Tabla 2.5 indica que el maestro (escáner) está intentando establecer la conexión con el esclavo (sensor).

Una vez identificado el campo de identificación perteneciente al Grupo 2 Conjunto de Conexión Predefinida Maestro/Esclavo enviado por el maestro, se muestran 4 campos de identificación, de los cuales dos se presentan mientras se establece la comunicación y los dos restantes cuando ya se ha establecido la comunicación entre el maestro y el esclavo.

En las Figuras 2.3 y 2.4 se muestran los campos de identificación para la comunicación maestro/esclavo.



**Figura 2.3** Campo de Identificación pertenecientes al Grupo 2 (Estableciendo Comunicación Maestro/Esclavo)



**Figura 2.4** Campo de Identificación pertenecientes al Grupo 2 (Comunicación establecida Maestro/Esclavo)

Con los datos obtenidos es posible tener información sobre la forma en que se realiza la comunicación Maestro/Esclavo.

En la Tabla 2.8 y 2.9 se detalla el análisis de estos campos de identificación de acuerdo a la distribución de conexiones ID que define DeviceNet para Grupo 2 Conjunto de Conexión Predefinida Maestro/Esclavo.

**Tabla 2.8** Campo de Identificación pertenecientes al Mensaje Grupo 2

Frame ID [Hexadecimal]	Frame ID [Binario]	Grupo de Mensaje	MAC ID	Número de Mensaje	Interpretación del Mensaje
0x044C	10001001100	10	001001	100	Mensaje de Solicitud Explícita del Maestro.
0x044B	10001001011	10	001001	011	Mensaje de Respuesta Explícita/Desconexión del Esclavo.
0x044A	10001001010	10	001001	010	Mensaje ACK tipo Cambio de Estado o Cíclico del Maestro.



**Tabla 2.9** Campo de Identificación perteneciente al Mensaje Grupo 1

Frame ID [Hexadecimal]	Frame ID [Binario]	Grupo de Mensaje	Número de Mensaje	MAC ID	Interpretación del Mensaje
0x0349	01101001001	0	1101	001001	Mensaje E/S tipo Cambio de Estado (COS) o Cíclico (Cyclic) del Esclavo.

### Análisis de Datos

En la Tabla 2.10 se detalla el análisis de los datos que son enviados entre Maestro/Esclavo para establecer la comunicación y que fueron capturados mediante el analizador de trama.

**Tabla 2.10** Formato de Datos para la Comunicación Maestro/Esclavo

Número de Estado	Envío de Mensaje Maestro/Esclavo	Dato	Descripción Datos
1	Maestro	00 4B 03 01 01 00	Solicitud del maestro para objetos DeviceNet y respuesta del esclavo para indicar su presencia en la red.
	Esclavo	00 01 00 2B F5 0A 50	
2	Maestro	00 4B 03 01 01 00	Solicitud del maestro de información establecida por el vendedor sobre el dispositivo que está en la red (Número de Identificación, tipo de dispositivo DeviceNet, Código de Producto, Revisión, Estatus, Número Serial) y respuesta del esclavo
	Esclavo	00 CB 00	
3	Maestro	00 4B 03 01 10 00	Solicitud del maestro de información establecida por el vendedor sobre el dispositivo que está en la red (Número de Identificación, tipo de dispositivo DeviceNet, Código de Producto, Revisión, Estatus, Número Serial) y respuesta del esclavo
	Esclavo	00 CB 00	
4	Maestro	00 0E 01 01 01	Solicitud del maestro de información establecida por el vendedor sobre el dispositivo que está en la red (Número de Identificación, tipo de dispositivo DeviceNet, Código de Producto, Revisión, Estatus, Número Serial) y respuesta del esclavo
	Esclavo	00 8E 01 00	
5	Maestro	00 0E 01 01 02	Solicitud del maestro de información establecida por el vendedor sobre el dispositivo que está en la red (Número de Identificación, tipo de dispositivo DeviceNet, Código de Producto, Revisión, Estatus, Número Serial) y respuesta del esclavo
	Esclavo	00 8E 06 00	
6	Maestro	00 0E 01 01 03	Solicitud del maestro de información establecida por el vendedor sobre el dispositivo que está en la red (Número de Identificación, tipo de dispositivo DeviceNet, Código de Producto, Revisión, Estatus, Número Serial) y respuesta del esclavo
	Esclavo	00 8E 2C 00	

Continuación de la Tabla 2.10.

7	Maestro	00 0E 01 01 04	de cada parámetro solicitado.	
	Esclavo	00 8E 01 06		
8	Maestro	00 10 05 01 0C 03		
	Esclavo	00 90		
9	Maestro	00 10 2B 01 01 10 00		
	Esclavo	00 90 10 00		
10	Maestro	00 10 05 04 11 01 00		
	Esclavo	00 90 08 00		
11	Maestro	00 0E 05 02 07		Solicitud del maestro para obtener la información del valor del dato de entrada tipo encuesta y respuesta del esclavo del dato solicitado.
	Esclavo	00 94 16 FF		
12	Maestro	00 0E 05 04 07	Solicitud del maestro para obtener la información del valor del dato de entrada tipo cambio de estado y respuesta del esclavo del dato solicitado.	
	Esclavo	00 8E 01 00		
13	Maestro	00 0E 05 02 08	Solicitud del maestro para obtener la información del valor del dato de salida tipo encuesta y respuesta del esclavo del dato solicitado.	
	Esclavo	00 94 16 FF		

Continuación de la Tabla 2.10.

14	Maestro	00 0E 05 04 08	Solicitud del maestro para obtener la información del valor del dato de salida tipo cambio de estado y respuesta del esclavo del dato solicitado.
	Esclavo	00 8E 00 00	
15	Maestro	00 10 05 04 09 FA 00	Solicitud del maestro para información de tasa de paquete esperado y respuesta del esclavo a la información solicitada.
	Esclavo	00 90 00 01	

### **Análisis de Formato de Intercambio de Datos**

De acuerdo a la secuencia con la que se presentan las tramas y sus correspondientes campos de identificación se establece un formato de comunicación Maestro/Esclavo. Este formato de comunicación consta de dos etapas fundamentales que son el “Pre-Establecimiento de la Comunicación” y el “Intercambio de Datos”.

El “Pre-Establecimiento de la Comunicación” se presenta únicamente cuando se realiza la energización o reconexión del dispositivo a la red.

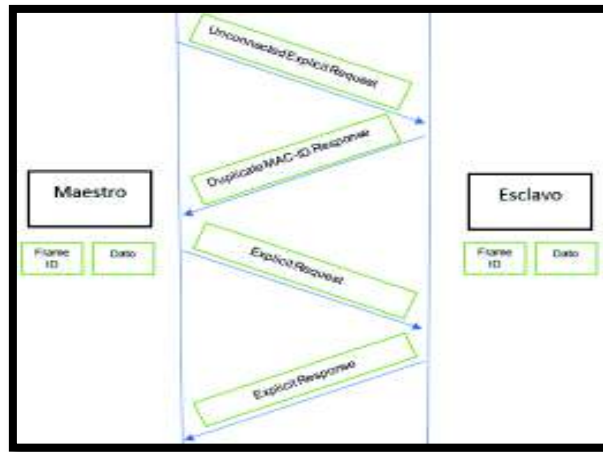
#### **Pre-Establecimiento de la Comunicación**

Esta primera etapa permite realizar la inicialización de la comunicación Maestro/Esclavo, es decir, establece la conexión entre ellos sin que el dispositivo esclavo envíe aún datos relacionados al proceso industrial.

- El Maestro envía una solicitud por Mensaje de Solicitud Explícito del Grupo 2 de desconexión (Group 2 Only Unconnected Explicit Request Message).
- El Esclavo indica al Maestro que inicie con el mensaje de chequeo de duplicación de MAC ID (Duplicate MAC ID Check Message).
- El Maestro envía una solicitud de mensajes explícito (Master’s Explicit Request Messages).

- El Esclavo envía una respuesta de mensaje explícito (Slave's Explicit/Unconnected Response Messages).

Se repetirá seguidamente los mensajes de solicitud y respuesta explícita entre Maestro/Esclavo hasta establecer la comunicación entre ellos como se muestra en la Figura 2.5.



**Figura 2.5** Formato de Establecimiento de Comunicación Maestro/Esclavo

Realizadas diferentes pruebas aplicadas al sensor en la red DeviceNet se determina que son necesarios 15 estados para establecer la conexión Maestro/Esclavo. Un estado está formado por una solicitud de comunicación por parte del maestro y la respuesta enviada por el esclavo.

La Tabla 2.11 muestra los 15 estados para establecer la comunicación Maestro/Esclavo.

**Tabla 2.11** Estados para Establecer la Conexión Maestro/Esclavo

Número de Estado	Frame ID	Dato
1	0x044E	00 4B 03 01 01 00
	0x044F	00 01 00 2B F5 0A 50
2	0x044E	00 4B 03 01 01 00
	0x044B	00 CB 00
3	0x044C	00 4B 03 01 10 00
	0x044B	00 CB 00
4	0x044C	00 0E 01 01 01
	0x044B	00 8E 01 00

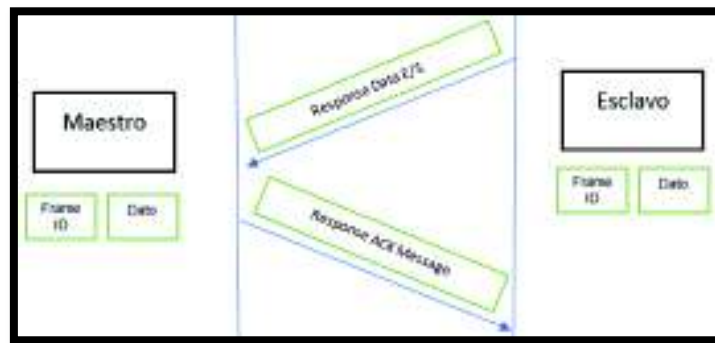
Continuación de la Tabla 2.11.

5	0x044C	00 0E 01 01 02
	0x044B	00 8E 06 00
6	0x044C	00 0E 01 01 03
	0x044B	00 8E 2C 00
7	0x044C	00 0E 01 01 04
	0x044B	00 8E 01 06
8	0x044C	00 10 05 01 0C 03
	0x044B	00 90
9	0x044C	00 10 2B 01 01 10 00
	0x044B	00 90 10 00
10	0x044C	00 10 05 04 11 01 00
	0x044B	00 90 08 00
11	0x044C	00 0E 05 02 07
	0x044B	00 94 16 FF
12	0x044C	00 0E 05 04 07
	0x044B	00 8E 01 00
13	0x044C	00 0E 05 02 08
	0x044B	00 94 16 FF
14	0x044C	00 0E 05 04 08
	0x044B	00 8E 00 00
15	0x044C	00 10 05 04 09 FA 00
	0x044B	00 90 00 01

### Intercambio de Datos

En esta segunda etapa ya se ha establecido la conexión Maestro/Esclavo y el dispositivo esclavo está habilitado para enviar datos relacionados con el proceso industrial al maestro, a través del procedimiento que se detalla a continuación y se representa en la Figura 2.6.

- El esclavo envía un mensaje indicando el tipo de dato (Slave's I/O COS or Cyclic Message).
- El maestro responde con un mensaje de reconocimiento del tipo de dato enviado por el esclavo (Master's COS or Cyclic Acknowledge Message).



**Figura 2.6** Formato de Comunicación Establecida Maestro/Esclavo

La Tabla 2.10 indica que a partir del estado 11 la información del valor de los datos de entrada y salida del sensor es necesaria, razón por la cual se debe conocer la forma de modificar estos valores. En el capítulo anterior se mencionó diferentes métodos de configuración de parámetros de dispositivos en una red DeviceNet, por lo que se detalla la modificación de la hoja de datos electrónica (EDS) a continuación.

### Hoja de Datos Electrónica (EDS)

La hoja de datos electrónica permite conocer toda la información de un dispositivo conectado a una red CIP como lo es DeviceNet, entre los cuales se destacan número de identificación, tipo de dispositivo, código de producto, revisión, status, número serial y la extensión del tamaño de los datos de entrada y salida que corresponde a uno de los objetivos importantes en este trabajo.

Como se mencionó en el capítulo anterior los campos obligatorios en un EDS son:

- File
- Device

El campo "File" contiene el nombre del dispositivo, fecha de creación del EDS, dirección web para acceder al EDS en línea del dispositivo, entre otros.

El campo "Device" contiene la información que el vendedor del dispositivo le asigna para identificarlo. Esta información es única para cada dispositivo, es decir, los parámetros especificados en esta sección no pueden ser iguales para dos dispositivos. Los parámetros especificados son vendedor, tipo de dispositivo, revisión, código y nombre del producto.

La Figura 2.7 muestra los datos establecido en un dispositivo por el vendedor.

```

[File] DescText = "RightSight 42EF Photoelectric Sensor E
      CreateDate      = 07-30-2005;
      CreateTime      = 12:00:00;
      ModDate         = 10-18-2005;
      ModTime         = 11:40:00;
      Revision        = 2.01;
      HomeURL         = "http://www.ab.com/networks/eds/

[Device] VendCode      = 0001;
         VendName     = "Rockwell Automation/Allen-Bradley";
         ProdType     = 6;
         ProdTypeStr  = "Photoelectric Sensor";
         ProdCode     = 44;
         MajRev       = 1;
         MinRev       = 4;
         ProdName     = "RightSight Standard Diffuse";
         Catalog      = "42EF-D1LDAK-F5";
         Icon         = "42EF PhotoSensor.ico";

```

**Figura 2.7** Parámetros Fundamentales de un EDS

En adición a estos parámetros se encuentra [IO\_Info] en el que se indica el tipo de dato admitido por el dispositivo y el número de bytes que son utilizados por el sensor.

En este caso el sensor fotoeléctrico “ RightSight Standard Diffuse” admite un 1 byte de entrada y puede ser de tipo COS (Change of State) o Estroboscópico (Strobe) como se muestra en la Figura 2.8.

```

[IO_Info]
  I_cosetPathNotSupported = 1;
  Default                 = 0x0004;
  PollInfo                = 0x0001, 0, 0;
  StrobeInfo              = 0x0002, 1, 0;
  COSInfo                 = 0x0004, 1, 0;
  Input1 =
    1,
    6,
    0x0006,
    "Sensor",
    6 "20 04 24 01 30 03"

```

**Figura 2.8** Parámetro IO\_Info Sensor Fotoeléctrico comercial “RightSight Standard Difuse”

Para la modificación de los datos de entrada del sensor se modifican las siguientes secciones del EDS:

- IO\_Info
- Variant\_IO\_Info

Es necesaria la modificación del parámetro “IO\_Info” para especificar el valor de los bytes de entrada que son necesarios para que sean admitidos los datos de un dispositivo.

Al realizar una modificación del parámetro anterior se hace necesario el implementar una nueva sección en el EDS y es “Variant\_IO\_Info” con el cual se habilita la modificación del parámetro “IO\_Info”.

De no utilizar este nuevo parámetro, aunque se modifique el valor del parámetro “IO\_Info” el dispositivo no admitirá el cambio realizado.

Se muestra en la Figura 2.9 el valor de “IO\_Info” y el formato de “Variant\_IO\_Info”.

```

Input1 =
3,
6,
0x0006,
"Sensor",
6, "20 04 24 01 30 03",
"The Format for the COS and Strobe input byte follows: \n"
"DB7 Not Used \n"
"DB6 Not Used \n"
"DB5 counter output \n"
"DB4 Motion Output \n"
"DB3 Margin Diagnostic 3x \n"
"DB2 Margin Diagnostic 2x \n"
"DB1 Diagnostic \n"
"DB0 Sensor Output \n"
"see specific parameter for more information.";

[variant_io_info]
input1 =
1,
6,
0x0006,
"Sensor",
6, "20 04 24 01 30 03",

```

Figura 2.9 Modificación del Parámetro IO\_Info y Variant\_IO\_Info

## 2.2 Diseño e Implementación de Drivers de Comunicación DeviceNet

### Especificación de Componentes y Requerimientos de Implementación

Establecidas los requerimientos para la comunicación entre dispositivos a una red DeviceNet se determina el hardware y software necesarios a implementar, así como las herramientas para verificar su funcionamiento. Los elementos requeridos de hardware se muestran en la Tabla 2.12 y su forma de implementación se muestra en la Figura 2.10.

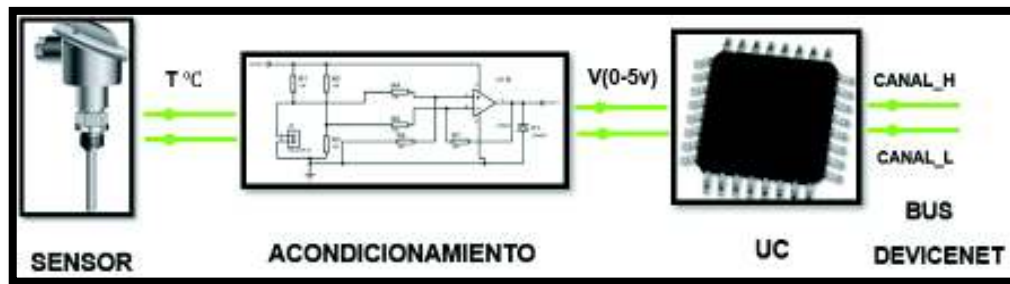
Tabla 2.12 Descripción de Elementos de Hardware

Elemento	Descripción	Características
Microcontrolador	El microcontrolador deberá permitir la programación de la trama de comunicación.	Procesar el protocolo CAN y manejar el voltaje diferencial de este protocolo
Sensores	Los sensores a utilizarse deben ser de tipo analógico y discreto.	Sensores analógicos y discretos con acondicionamiento de voltaje.



Continuación de la Tabla 2.12.

Circuitos de Acondicionamiento	de Tendrán los acondicionamientos para los sensores utilizados, así como los componentes para tener una conexión directa con el bus de la red.	Placas con borneras para conexión de los drivers al bus de datos DeviceNet y acondicionamientos de voltaje los sensores analógicos y discretos.
Red DeviceNet	La red debe tener dispositivos comerciales y los drivers implementados para verificar la interoperabilidad entre ellos.	Dispositivos comerciales de la red: el sensor fotoeléctrico con modo de censado difuso RightSight DeviceNet PHOTOSWITCH y un PLC CompactLogix con módulo escáner DeviceNet 1769-SDN



**Figura 2.10** Implementación de los componentes de Hardware

Los elementos requeridos de software se muestran en la Tabla 2.13.

**Tabla 2.13** Descripción de Elementos de Software

Software	Descripción
Analizador de Trama	Esta herramienta computacional debe permitir visualizar los datos que son enviados por los drivers y los dispositivos comerciales a través del bus de datos de la red.

Continuación de la Tabla 2.13.

Servidor de Comunicación	Permitirá el enlace entre la aplicación y la red DeviceNet.
Herramienta de Programación de Microcontroladores	Para realizar la programación de los microcontroladores
Herramienta de Programación de PLC	Para realizar la programación del controlador lógico programable (PLC).
Monitoreo de Red	Para configurar los drivers y dispositivos comerciales de la red DeviceNet.

### **Selección de Componentes**

En el laboratorio de Redes Industriales de la Escuela Politécnica Nacional se dispone de un módulo con sensores con protocolo CAN, los cuales son revisados para verificar si cumplen las especificaciones previamente establecidas y ser adaptados como parte del desarrollo de los prototipos de drivers de comunicación propuestos.

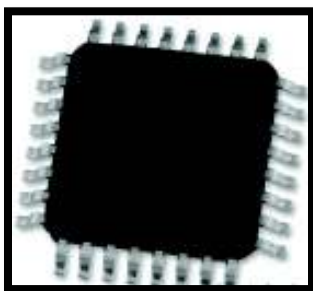
Tras la revisión se verifica que es factible reutilizar las placas con los microcontroladores pero los acondicionamientos de cada sensor requieren implementarse en nuevas placas ya que es necesario incluir los bornes de conexión hacia el bus de la red DeviceNet.

#### **Módulo de Sensores CAN**

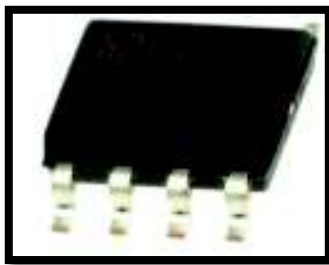
El módulo cuenta con tres nodos CAN:

- Nodo de Nivel
- Nodo Discreto de Humo, Presencia y Luz
- Nodo de Temperatura

Todos los nodos están formados por un microcontrolador AT90CAN32 el cual procesa tramas de datos CAN y un transceiver CAN L9616 el cual permite que el microcontrolador trabaje con el voltaje diferencial para este protocolo. En las Figuras 2.11 y 2.12 se muestran los dispositivos mencionados.



**Figura 2.11** Microcontrolador AT90CAN32 [16]



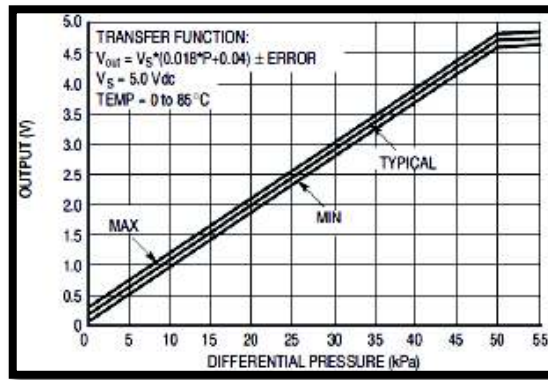
**Figura 2.12** Tranceiver CAN L9616 [11]

### **Nodo de Nivel**

El nodo de nivel consta de un sensor de presión diferencial MPX5050DP, el mismo que tiene una salida de voltaje acondicionada de cero a cinco voltios, este sensor se muestra en la Figura 2.13.



**Figura 2.13** Sensor de presión diferencial MPX5050DP [12]



**Figura 2.14** Salida vs presión diferencial del sensor MPX5050DP [12]

### Nodo Discreto

El nodo discreto consta de un sensor de luz LDR (Light Diode Resistance), un sensor de presencia y un sensor de humo. Cada uno de estos sensores emplea un byte para enviar su información.



**Figura 2.15** Sensor de luz LDR [1]



**Figura 2.16** Sensores de Presencia y Humo [15]

Los sensores de Humo y Presencia envían un 0L correspondiente a un voltaje de 0 VDC y 1L correspondiente a un voltaje de 5 VDC al momento de accionarse.

El sensor de Luz LDR realiza una comparación de voltaje, con su voltaje mínimo representa un 0L y con un voltaje máximo de 5 VDC representa un 1L.

## Nodo de Temperatura

El sensor utilizado es SIEMENS PT100 de tres hilos con rango de medición de -50 a +250 °C .



Figura 2.17 Sensor de Temperatura PT100 [15]

## Implementación de Drivers de Comunicación


### Diseño e Implementación de Hardware

#### Cable de Interface

El cable utilizado en una red DeviceNet puede ser de tipo redondo (Round) que está formado por cinco hilos conductores o plano (Flat) que está formado por cuatro hilos conductores.

El cable de tipo redondo puede ser utilizado para la línea troncal como para líneas de derivación de la red, por este motivo se utiliza un cable tipo redondo empleado en sistemas contra incendios de uso industrial de 4 hilos de 18-22 AWG para realizar la expansión de la red DeviceNet.

Tabla 2.14 Descripción del cable de 4 Hilos

Cable de 4 Hilos 18 AWG	Número de Hilos	Designación del Hilo	Descripción
	2	V+	Alimentación de 24 VDC
		V-	
	2	Canal H	Canales del bus de datos.
		Canal L	

En la Figura 2.18 se visualiza la red DeviceNet formada por dispositivos comerciales y los prototipos implementados.



**Figura 2.18** Expansión de la red DeviceNet

### **Placas Diseñadas**

Se realizó el diseño de tres placas que cumplan las condiciones necesarias para poder conectarlas al bus de datos de la red DeviceNet. Cada una de las placas considera los acondicionamientos necesarios para cada sensor y las borneras para conexión a la red DeviceNet. Los acondicionamientos fueron replicados a los existentes en el módulo de sensores CAN.

Como se mencionó en el capítulo anterior el bus provee una alimentación de 24V DC por lo que se ve necesario el empleo de un regulador de voltaje ya que los sensores tienen alimentación de 5V DC (luz y nivel) y de 12V DC (humo y presencia).

También se consideró que se tiene que enviar la información al canal H y L por lo que las placas están acondicionadas con salidas para estos canales.

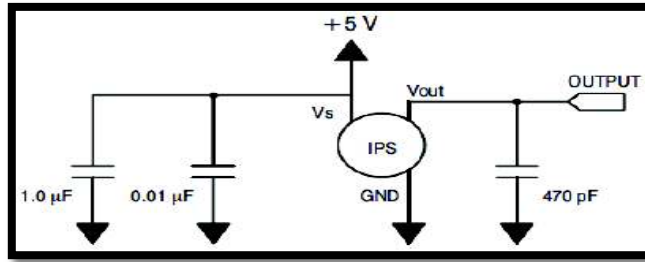
Las placas diseñadas son:

- Placa Nodo de Nivel
- Placa Nodo Discreto de Humo, Presencia y Luz
- Placa Nodo de Temperatura

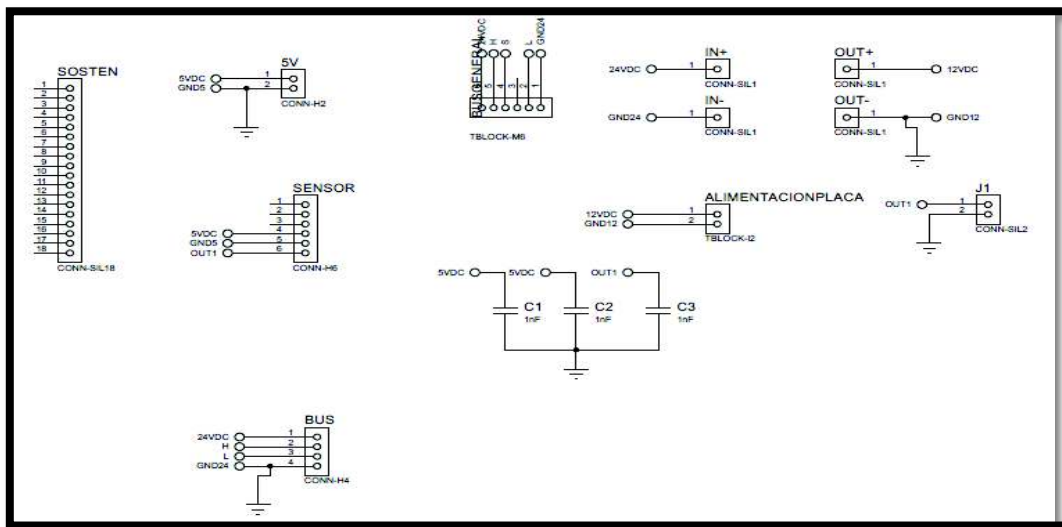
### **Nodo de Nivel**

En el caso de la placa diseñada se considera que disponga de borneras de conexión para el bus de datos de la red DeviceNet y el acondicionamiento para el sensor de presión diferencial MPX5050DP, este último fue implementado con capacitores para desacoplamiento y filtro a la salida con valores ya establecidos por el fabricante.

El diseño de la placa con el acondicionamiento mencionado se muestra a continuación en las Figuras 2.19 y 2.20.



**Figura 2.19** Acondicionamiento del Sensor Presión Diferencial MPX5050DP



**Figura 2.20** Diseño de placa Nodo de Nivel

### Nodo Discreto

La placa diseñada posee las borneras de conexión para el bus de datos de la red DeviceNet y para el acondicionamiento de los tres sensores discretos, los cuales serán implementados usando optoacopladores 4N25 como se muestran en las Figuras 2.21 y 2.22, para acondicionar los voltajes de 24 VDC del bus CAN a los 12 VDC de las placas electrónicas y viceversa. En las ecuaciones 2.1 y 2.2 se muestran los cálculos de las resistencias utilizadas para el acondicionamiento del nodo discreto.

$$R_{ENTRADA} = \frac{V}{I} = \frac{12V - 1.3V}{10mA} = 1070 \text{ ohmios} \cong 1k\text{ohmio}$$

**Ecuación 2.1** Cálculo de valor de resistencia a la entrada del optoacoplador 4N25

Donde:

V= Representa la diferencia entre el voltaje de alimentación y voltaje de caída del led del optoacoplador 4N25.

I= Representa la corriente de entrada del optoacoplador 4N25.

$$R_{SALIDA} = \frac{V}{I} = \frac{5}{50mA} \cong 100 \text{ ohmios}$$

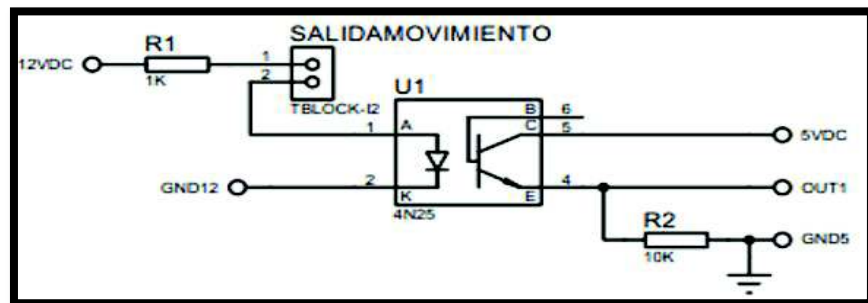
**Ecuación 2.2** Cálculo de valor de resistencia a la salida del optoacoplador 4N25

Donde:

V= Representa el voltaje de operación de optoacoplador 4N25 a 25°C.

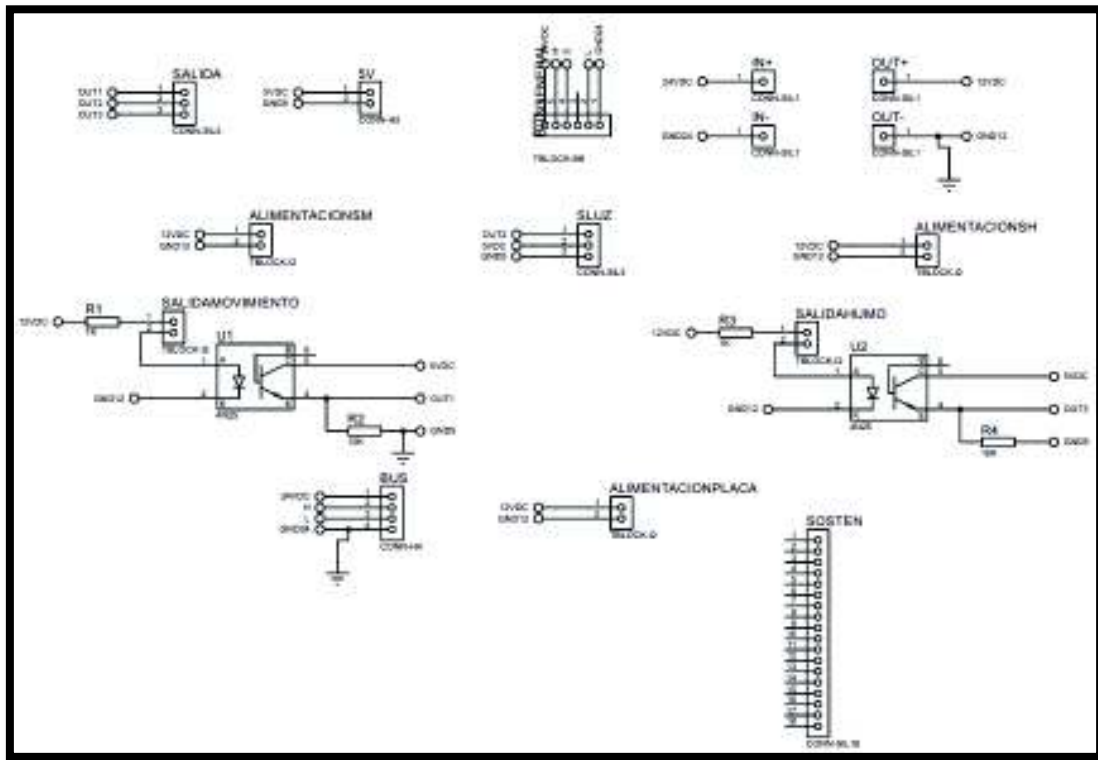
I= Representa la corriente de salida del optoacoplador 4N25.

Con los cálculos realizados se observa que se puede emplear una resistencia de mayor o igual a 100 ohmios para garantizar que no se exceda la máxima corriente de salida del optoacoplador por lo que se utiliza una resistencia de 10kohmios.



**Figura 2.21** Acondicionamiento de optoacopladores 4N25





**Figura 2.22** Diseño de placa Nodo Discreto

### Nodo de Temperatura

Se considera en la placa diseñada las borneras de conexión para el bus de datos de la red DeviceNet y para el acondicionamiento de sensor de temperatura SIEMENS PT100, en el caso del acondicionamiento se utilizó un puente de wheatstone con un amplificador operacional LM324 como se muestra en la Figuras 2.23 y 2.24.

Se establece los valores de las resistencias para poder conocer el voltaje a la salida del puente de wheatstone utilizando la ecuación 2.3.

$$V_s = \left( \frac{R_2}{R_2 + R_1} - \frac{R_4}{R_4 + R_3} \right) \times V = \left( \frac{1000K}{1000K + 1K} - \frac{100}{100 + 1K} \right) \times 9 = 8.17$$

### Ecuación 2.3

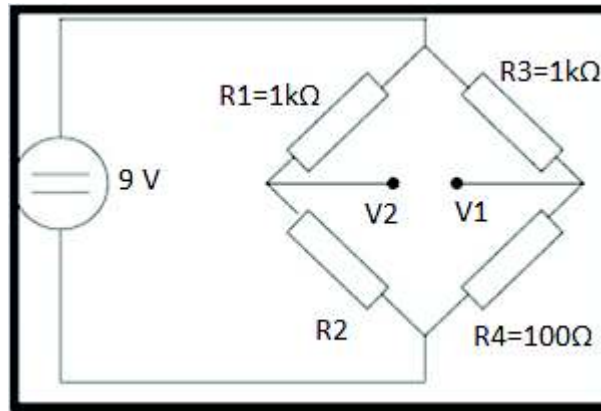
 Acondicionamiento para Puente de Wheatstone

Donde:

V= Representa el voltaje de alimentación.

Ri = Representa las resistencias que forman el puente de wheatstone.

Vs = Representa el voltaje de salida del puente de wheatstone.



**Figura 2.23** Acondicionamiento de Sensor de Temperatura PT100

Se establece los valores de las resistencias  $R1=R3= 1\text{Kohmio}$  y  $R2=R4=1.95\text{Kohmios}$  del amplificador operacional LM324 para poder conocer el voltaje a la salida utilizando la ecuacion 2.4.

$$V_{out} = \left(\frac{R2}{R1}\right) \times (V2 - V1) = \left(\frac{1k}{1.95k}\right) \times (V2 - V1) = 0.51 \times (V2 - V1)$$

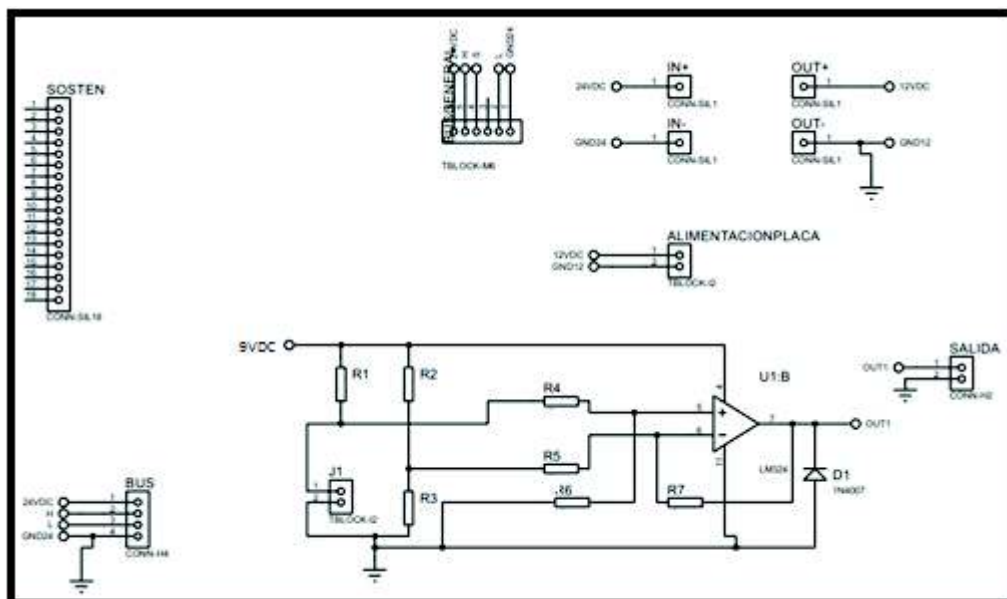
**Ecuación 2.4** Acondicionamiento para Amplificador operacional LM324

Donde:

$V2, V1=$  Representan los voltajes de los ramales del puente de wheatstone.

$R_i=$  Representa las resistencias de acoplamiento para el amplificador operacional.

$V_{out}=$  Representa el voltaje de salida del amplificador operacional.



**Figura 2.24** Diseño de placa Nodo de Temperatura

## **Desarrollo de Software**

### **Diagrama de Flujo del Microcontrolador**

A partir del análisis de la comunicación en redes DeviceNet, se debe tener en cuenta que la comunicación entre nodos en una red DeviceNet consta de dos etapas, la primera en la que se preestablece la comunicación Maestro/Esclavo configurando los 15 estados de comunicación determinados anteriormente y que se detallan en la Tabla 2.11 y la segunda cuando se ha establecido la comunicación, con este objeto se programará el microcontrolador en cada prototipo de driver de comunicación para que cumpla con este formato y secuencia de datos.

En el desarrollo del programa del microcontrolador se vio necesario implementar un registro que maneja el protocolo CAN que es el “objeto de mensaje” o “MOB”. Este registro permite manejar una trama CAN, es decir, permite realizar la descripción de un mensaje CAN como un objeto en donde los mensajes a enviar y/o recibir son predefinidos para disminuir el trabajo del software.

Existen diferentes tipos de objetos de mensaje tales como: modo desactivado, modo de transmisión y modo de recepción.

Los objetos de mensaje utilizados son el modo de transmisión y recepción en el que se envía la información de campo de identificación y datos de conexión de cada prototipo implementado.

Una vez enviada esta información y establecida la comunicación Maestro/Esclavo se realiza el envío del dato por el esclavo y se espera una confirmación de mensaje recibido del maestro.

El software en el que se desarrolla la programación del microcontrolador es AVRStudio 7 de ATMEL y el lenguaje de programación que se utilizará es lenguaje C.

En la Figura 2.25 se muestra el diagrama de flujo del programa realizado en cada microcontrolador y el código correspondiente se encuentra en el Anexo I.

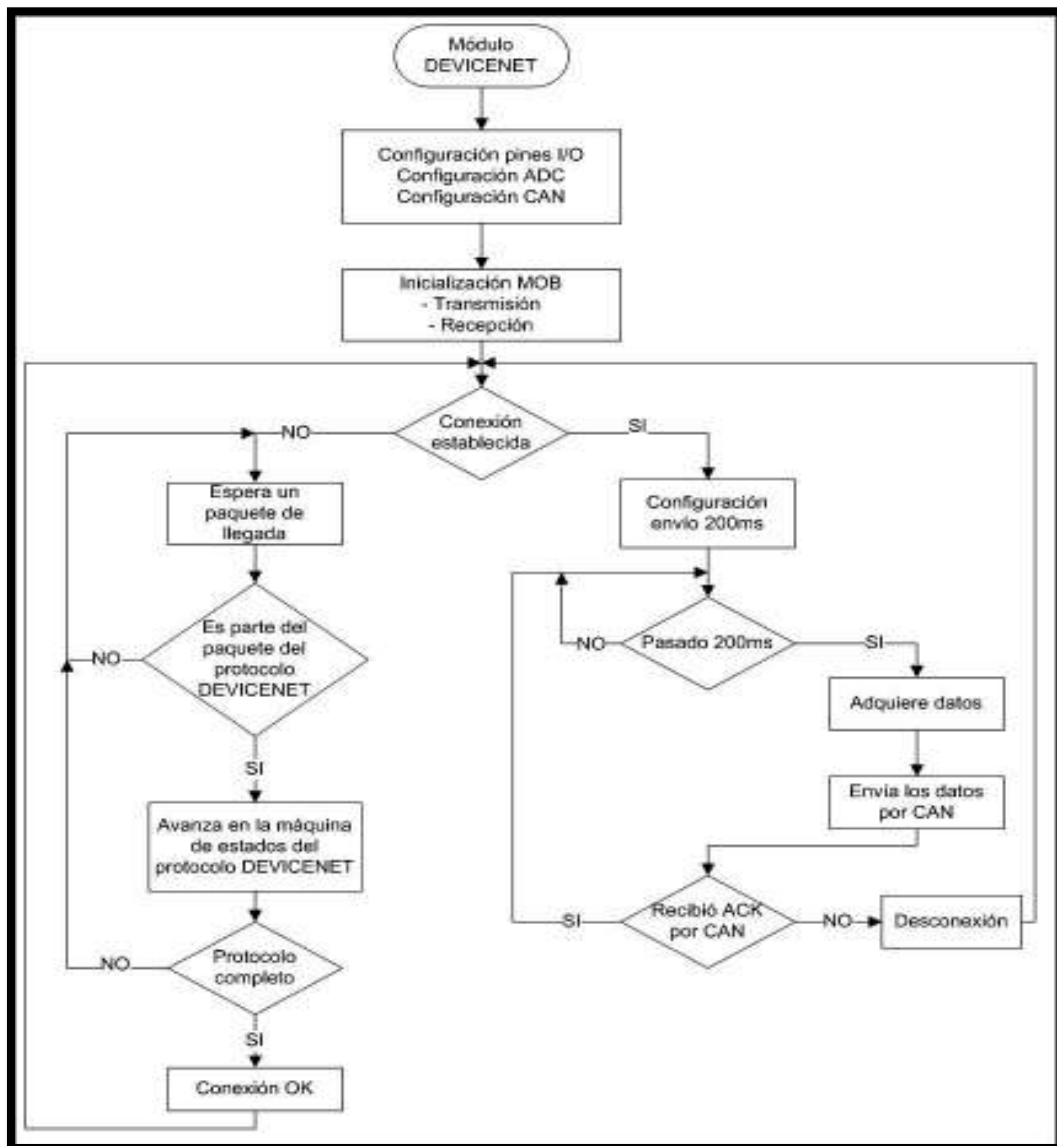


Figura 2.25 Diagrama de Flujo del Microcontrolador

## 2.3 Desarrollo de la Plataforma de Validación y Pruebas

Se ve necesario el utilizar una red DeviceNet para integrar los prototipos de los drivers diseñados que al interactuar con dispositivos comerciales permitirán verificar su interoperabilidad por lo que se describe a continuación los componentes de software y hardware y software a utilizar con este fin.

### Módulo de Red Industrial DeviceNet

La red DeviceNet tiene los siguientes componentes:

- PLCs CompactLogix con módulo scanner DeviceNet 1769-SDN: Estos dos elementos forman una unidad, y corresponde al dispositivo maestro de la red.

- Sensor fotoeléctrico RightSight DeviceNet PHOTOSWITCH con modo de censado difuso: Es un dispositivo esclavo.
- PanelView 300 DeviceNet: Dispositivo esclavo con datos de entrada y salida.
- PowerFlex700 DeviceNet (interfaz 20-COMM-D): Variador de velocidad, es un dispositivo esclavo.
- Interfaz RS232/DeviceNet: Permite tener la conexión del computador personal a la red, este dispositivo no forma parte de los esclavos de la red.
- DNIs DeviceNet Interface: Permite la comunicación entre dispositivos con protocolo RS-232.
- Base para Columna Luminosa: Dispositivo esclavo.



**Figura 2.26** Módulo de Red DeviceNet

El PLC con módulo escáner es el maestro y los restantes dispositivos pueden ser configurados como esclavos en la red.

### **Software de Configuración**

- **RSLinx:** Software que permite realizar la comunicación entre la red DeviceNet y la aplicación desarrollada en el computador personal.
- **RSLogix5000:** Software en el que se desarrolla la programación del controlador lógico programable (PLC).

- **RSNetWorx:** Software en el que se realiza la configuración de todos los elementos de la red DeviceNet e integración de las hojas de datos electrónicas (EDS).
- **USB-CAN Tool V2.02:** Software que permite ver el campo de identificación y datos de los dispositivos de la red.

## Configuración de la red DeviceNet

### Configuración de RSLinx

Se realiza la configuración en RSLinx para comunicar al PLC con el software RSLogix 5000 en el cual se desarrolla la lógica de programación para la aplicación.

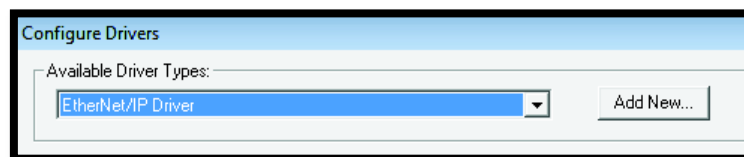
Se abre el software RSLinx que se encuentra en el menú Inicio y se realiza su configuración siguiendo los siguientes pasos:

- En la barra de herramientas se presiona el botón “Configure Drivers”.



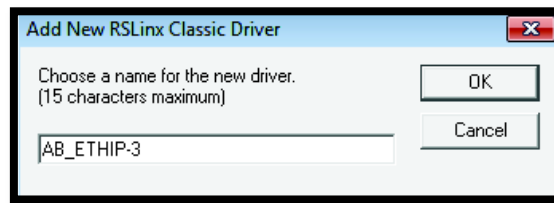
**Figura 2.27** Configuración de Controladores

- Aparece una ventana “Configure Drivers” se selecciona el tipo de controlador “Ethernet/IP Driver” y se presiona “Add New”.



**Figura 2.28** Selección del tipo de Controlador

- Se despliega un cuadro de diálogo “Add New RSLinx Driver” sin realizar modificaciones se presiona “OK”.



**Figura 2.29** Asignación del nombre del Controlador

- Se despliega una ventana en donde se asigna la dirección IP del computador, se presiona “Apply”.



**Figura 2.30** Configuración de la dirección IP del computador

- En la ventana de “Configure Drivers” aparece el controlador seleccionado anteriormente.



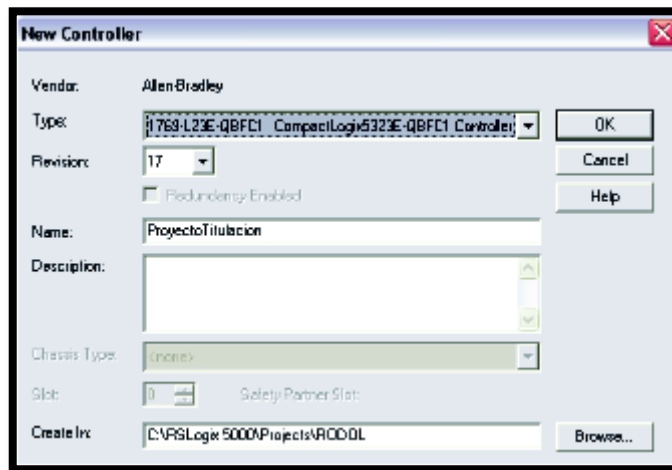
**Figura 2.31** Verificación del Controlador

### **Configuración de la Aplicación con RSLogix5000**

En el software RSLogix 5000 se establece la comunicación con el PLC y el software RSNetWorx en el que se diseña la red implementada.

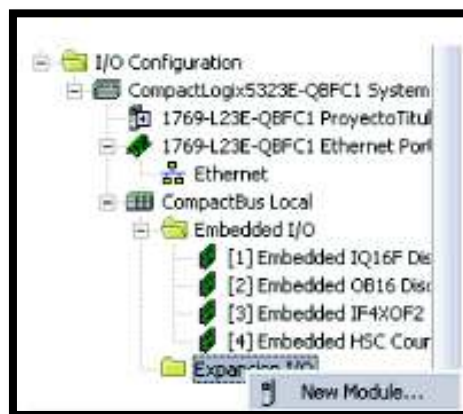
El PLC tiene que reconocer al módulo escáner y esto es posible asignando un slot en su chasis. Se utiliza el software RSLogix5000 para desarrollar la aplicación.

En RSLogix5000 en el menú “File” se selecciona “Create New” aparece una ventana en la cual se selecciona el controlador a utilizarse.



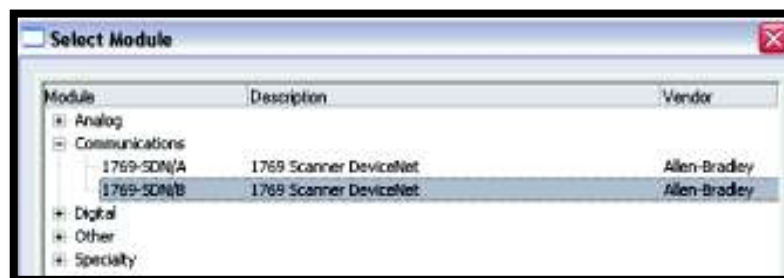
**Figura 2.32** Selección del Controlador

En la parte izquierda de la pantalla se encuentra “I/O Configuration”, seleccionamos “Expansion I/O” damos clic derecho y se presiona “Módulo Nuevo”.



**Figura 2.33** Agregar módulo nuevo

Se despliega una ventana en la que se selecciona “Select Module “ y en “Communication” se elige el módulo 1769-SDN serie B y se presiona “OK”



**Figura 2.34** Selección del módulo escáner

Aparece un cuadro de diálogo “New Module” en el cual se ingresa el nombre para el módulo y se acepta “OK”.

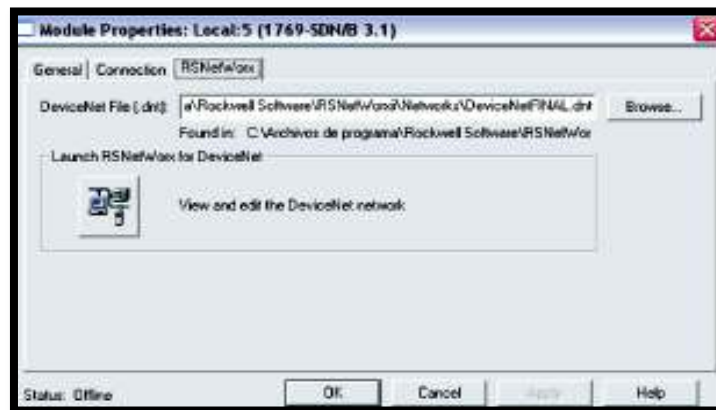




**Figura 2.35** Asignación de nombre al módulo escáner

Ya que se ha configurado al módulo escáner para que pueda utilizarse en la aplicación se procede a enrutar el archivo creado en RSNetWorx.

Se presenta una ventana “Module Properties”, en la pestaña “RSNetWorx” se busca la ruta del archivo en el que se configuró la red DeviceNet. Una vez seleccionado el archivo se presiona “OK”.



**Figura 2.36** Vinculación de RSLogix5000 con RSNetWorx

Se procede a crear la aplicación para lo cual se debe considerar que los tags que corresponden al módulo escáner son: Local:5:I entradas y Local:5:O salidas, ya que en estas localidades se encontrarán los datos enviados o recibidos de los diferentes dispositivos conectados a la red DeviceNet. La localidad 5 se debe a que el módulo escáner ha sido instalado en el slot 6 del PLC CompactLogix.

Es posible visualizar los tags correspondientes al escáner en el submenú desplegado a la izquierda de la pantalla inicial de RSLogix5000 en la opción “Controller Tags”.

Name	Alias For	Base Tag	Data Type
Local 1:C			AB:Embedded_IQ...
Local 1:I			AB:Embedded_IQ...
Local 2:C			AB:Embedded_O...
Local 2:I			AB:Embedded_O...
Local 2:O			AB:Embedded_O...
Local 3:C			AB:Embedded_IF...
Local 3:I			AB:Embedded_IF...
Local 3:O			AB:Embedded_IF...
Local 4:C			AB:Embedded_H...
Local 4:I			AB:Embedded_H...
Local 4:O			AB:Embedded_H...
Local 5:I			AB:1769_SDN_4
Local 5:O			AB:1769_SDN_3...

**Figura 2.37** Tags correspondientes al Módulo Escáner

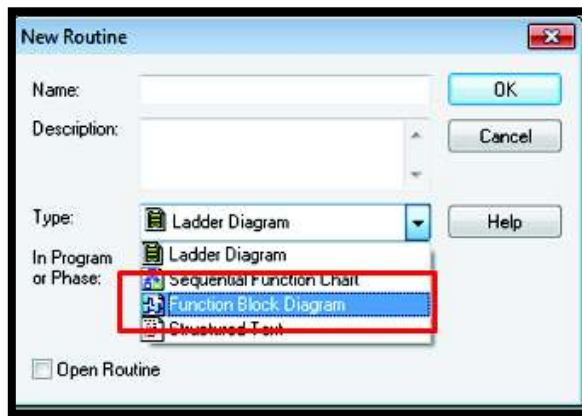
Asignado el módulo escáner en el software RSLogix5000 se desarrolla la lógica de programación en el lenguaje de programación FBD (Function Block Diagram).

Al empezar un proyecto se crean automáticamente archivos o fichas llamadas rutinas, se ubica “MainRoutine” que se encuentra en el submenú de la izquierda de la pantalla, con clic derecho se despliega un submenú en el que se selecciona “New Program”.



**Figura 2.38** Creación de una nueva rutina de Programación

Se despliega una ventana en la que se escribe el nombre de la rutina y se elige el tipo de rutina (Ladder,FDB,SFC) seguidamente se presiona “OK “.



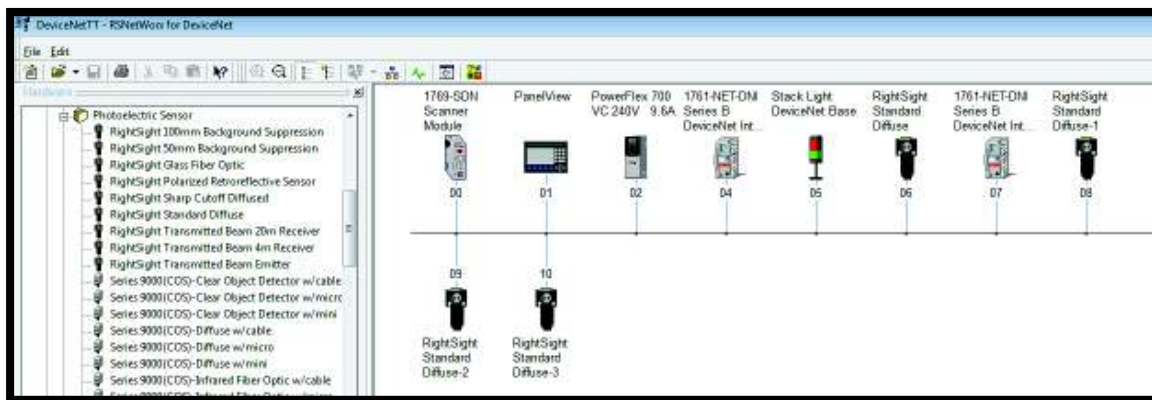
**Figura 2.39** Selección del tipo de lenguaje de Programación

Se desarrolla la lógica descrita anteriormente en el lenguaje FBD. Terminada la lógica de programación se procede a realizar la descarga del proyecto en el PLC.

### Configuración de la red con RSNetWorx

El diseño de la red es implementado con el software RSNetWorx para DeviceNet.

Se puede asignar la dirección en el dispositivo haciendo doble clic sobre el mismo y digitando a continuación la dirección que se le desea asignar. Es posible asignar las direcciones de los dispositivos del valor 0 al 63 como se muestra en la Figura 2.40.



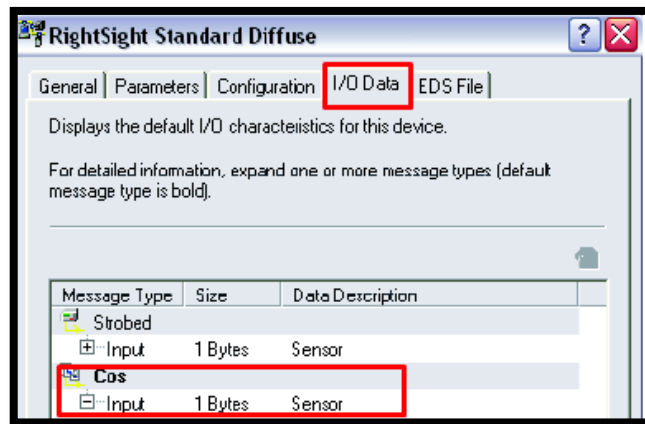
**Figura 2.40** Representación de la red DeviceNet y los nodos CAN en el software RSNetWorx

Como se observa en la Figura anterior cada prototipo se representa como un sensor “RightSight Standard Difuse” asignando la dirección a cada uno de ellos de la siguiente manera:

- Nodo de Temperatura dirección “08”
- Nodo Discreto dirección “09”

- Nodo de Nivel dirección “10”

Como se vio en las características del sensor fotoeléctrico comercial “RightSight Standard Diffuse” envía 1 byte de dato de entrada al igual que los nodos de nivel y temperatura. En RSNetWorx se puede obtener esta información dando doble clic sobre el dispositivo en la ventana que se despliega se selecciona la pestaña “I/O Data”.

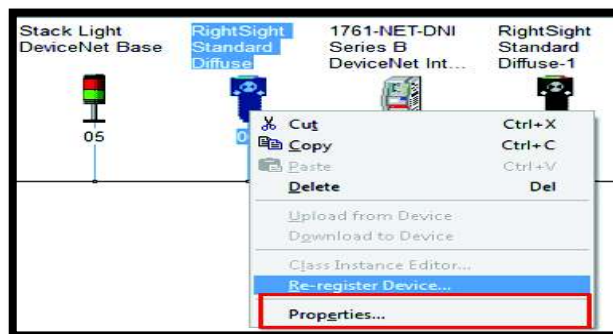


**Figura 2.41** I/O Data del sensor RightSight Standard Diffuse

El nodo de nivel al estar formado por tres sensores (luz, presencia y humo) requiere del envío de 3 bytes de datos de entrada por lo que se ve necesario modificar el EDS del sensor comercial para adaptarlo a las características de este nodo.

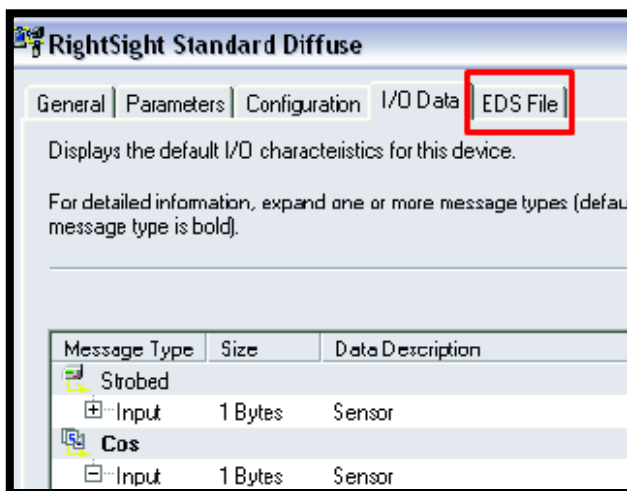
Se accede al EDS del sensor fotoeléctrico comercial “RightSight Standard Difuse” a través del software RSNetWorx siguiendo los siguientes pasos:

- Damos un clic con el botón derecho del mouse y se despliega un submenú, en el que seleccionamos la opción “Properties”.



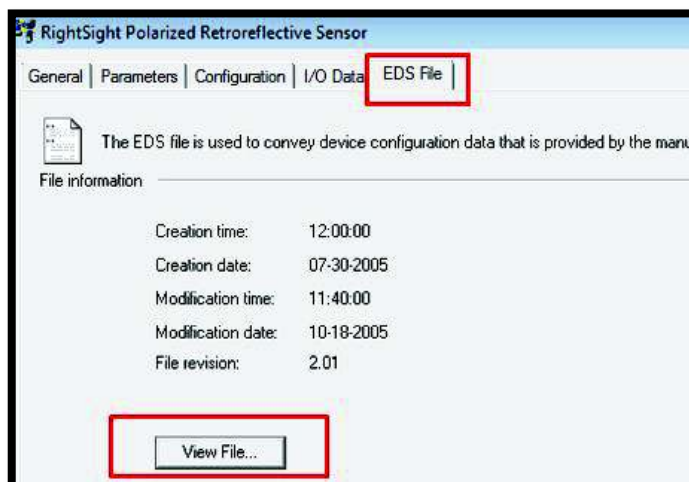
**Figura 2.42** Acceso al EDS del Sensor

- Se despliega una ventana, con 5 pestañas entre ellas “EDS File” la seleccionamos.



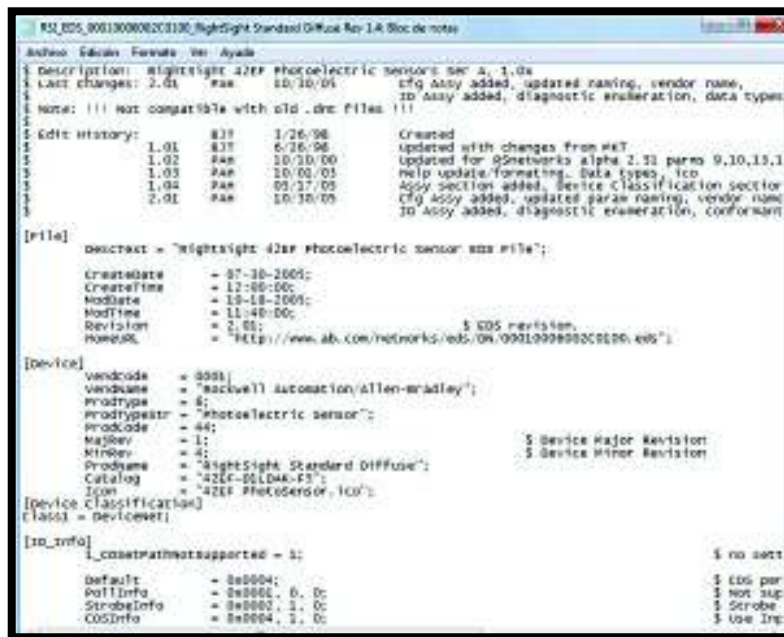
**Figura 2.43** Selección del EDS File del Sensor

- Hacemos clic sobre “View File” y se despliega el EDS del sensor en un archivo de block de notas.



**Figura 2.44** Visualización del EDS del Sensor

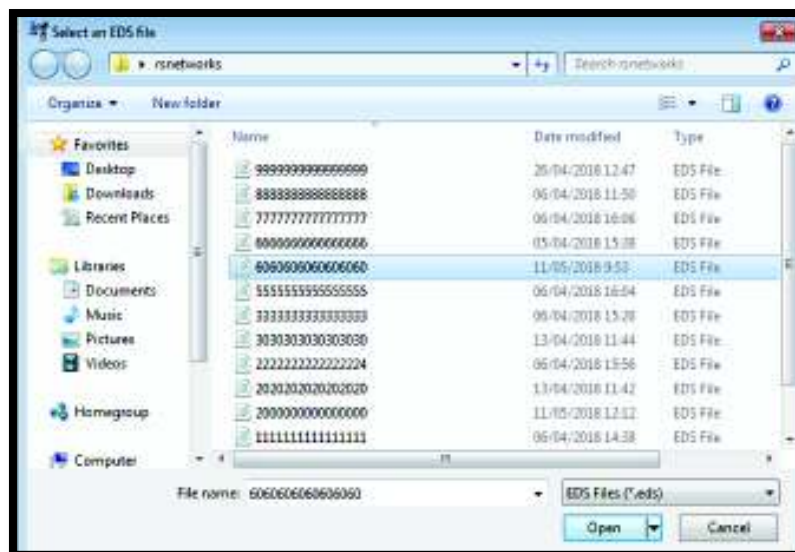
En la Figura 2.45 se muestra en detalle el contenido del archivo EDS.



**Figura 2.45** EDS Sensor Fotoeléctrico comercial “RightSight Standard Difuse”

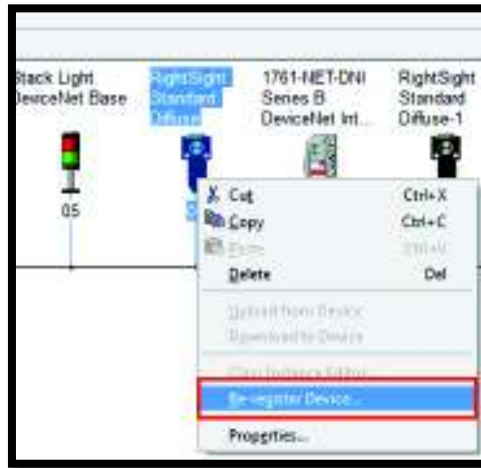
Como se puede apreciar en la Figura 2.45 se encuentra toda la información referente al sensor. Se realiza las modificaciones en el parámetro de entradas y salidas del sensor, las mismas que fueron establecidas previamente en secciones anteriores.

Realizados todos los cambios se creará un nuevo EDS, se copia el contenido del EDS modificado a un nuevo bloc de notas y se le asigna un nombre, el mismo que debe tener un formato de 16 números enteros y con extensión “.eds”.



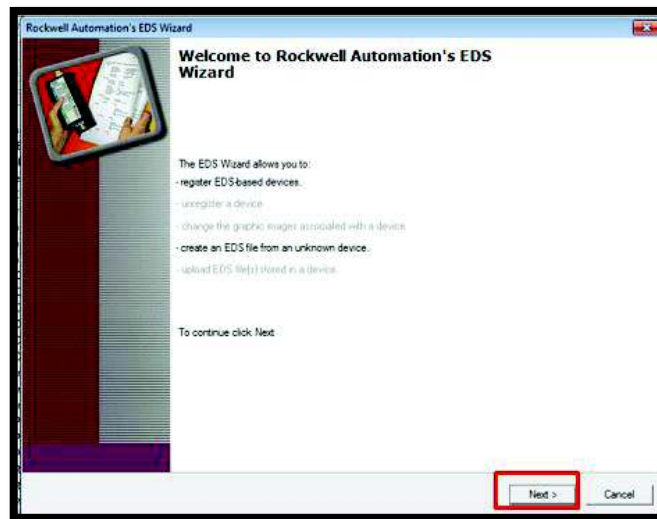
**Figura 2.46** Asignación de nombre al EDS

- Para realizar el cambio de EDS en el dispositivo a través del software RSNetWorx, se hace clic derecho y se despliega un submenú en el que se selecciona “Re-register Device”.



**Figura 2.47** Registro del nuevo EDS

- Se despliega un cuadro de diálogo en la que seleccionamos “Next”.



**Figura 2.48** Inicio de carga del archivo EDS

- Se despliega una ventana en la que seleccionamos “Register an EDS files” y presionamos “Next”.



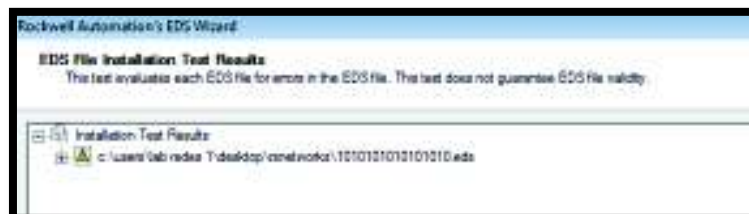
**Figura 2.49** Registro de un archivo EDS

- A continuación, seleccionamos en “Browse” el archivo que se creó en block de notas con la información configurada anteriormente y presionamos “Next”.



**Figura 2.50** Selección de la ruta de archivo EDS

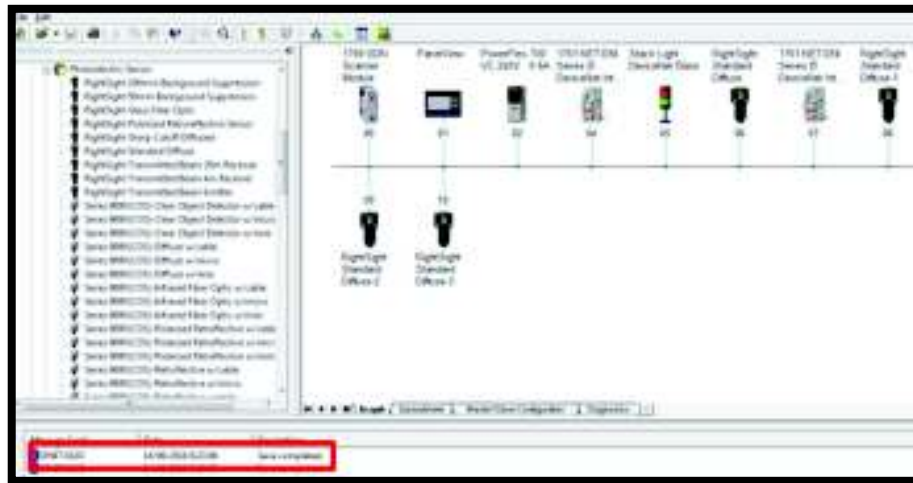
- Se presiona “Next” en las siguientes ventanas que se despliegan.



**Figura 2.51** Finalización de la carga del archivo EDS

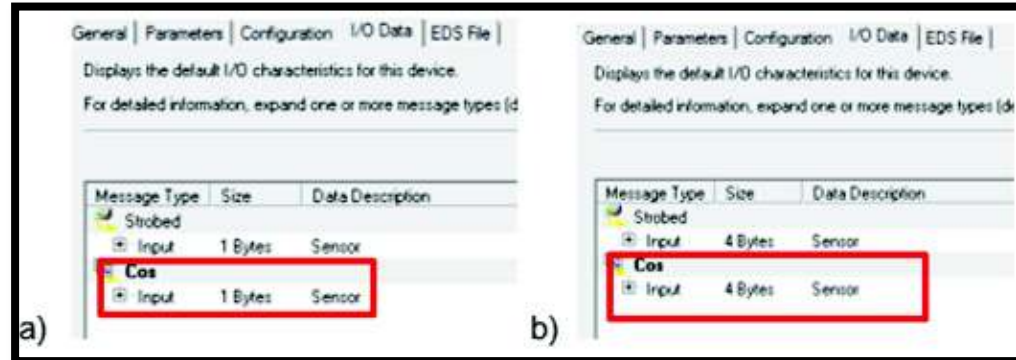


- Para guardar los cambios realizados en el sensor con su nueva EDS se cierra RSNNetWorx y se lo vuelve a abrir.



**Figura 2.52** Finalización de la configuración del EDS del sensor

Para verificar el cambio realizado en el EDS del sensor se puede hacer doble clic sobre el mismo, se despliega una ventana y en la pestaña “I/O Data” se muestra que ya posee hasta 4 bytes para datos de entrada.



**Figura 2.53** Parámetro I/O Data a) Sensor Comercial b) Sensor Configurado

### Mapeo de Datos con RSNNetWorx

Se realiza en el software RSNNetWorx el mapeo de datos para asignar a cada uno de los elementos que intercambian datos en la red a una localidad de memoria del módulo escáner.

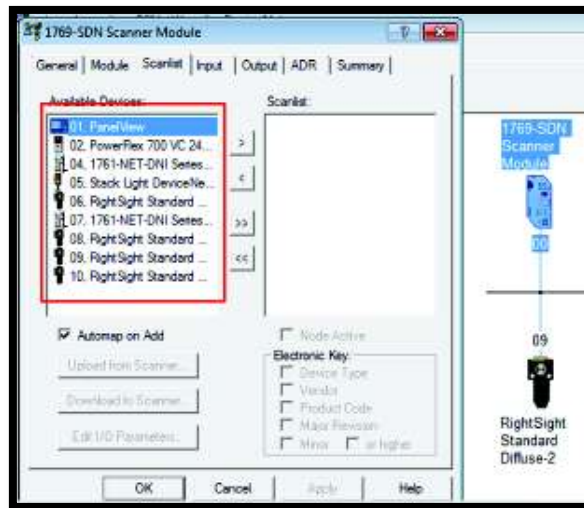
Los dispositivos que serán mapeados son el PanelView, PowerFlex y los 4 sensores “RightSight Standard Difuse”, de los cuales 3 corresponden a los prototipos.

Para mapear estos dispositivos se presiona el botón en “línea” y aparecerá un cuadro de diálogo que se acepta.



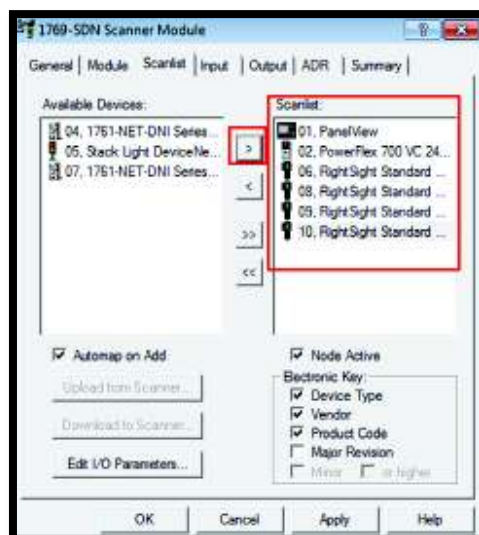
**Figura 2.54** Acceso de RSNetWorx a la red

Se inicia de esta manera un barrido de todos los nodos de la red. Para ingresar a las localidades de memoria del escáner se hace doble clic sobre la figura y se ingresa a la pestaña “Scanlist”.



**Figura 2.55** Ingreso a Scanlist

En esta ventana aparecen todos los dispositivos de la red y se agrega uno a uno los dispositivos antes mencionados.



**Figura 2.56** Dispositivos a ser mapeados

Finalizada la selección de los dispositivos se ingresa el valor de entradas y salidas de cada uno.

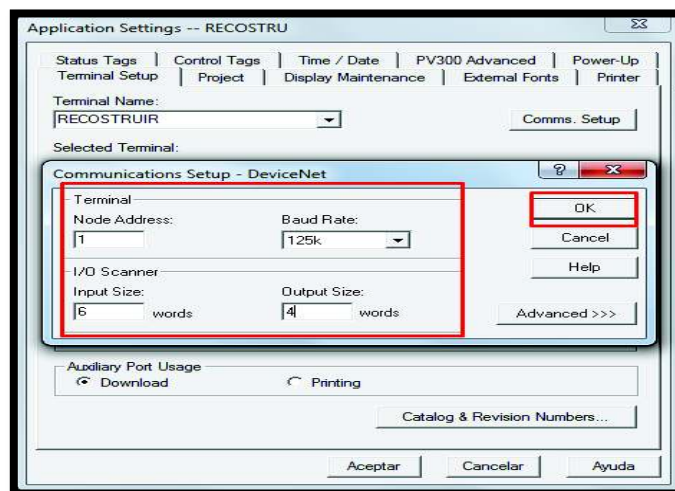
Los datos de entrada y salida del PanelView deben ser configurados siempre por el usuario. La programación del PanelView se puede realizar con el software PanelBuilder32, el mismo que permite realizar la configuración de las diferentes pantallas a mostrarse, creación de tags, configuración de teclas de función, entre otros.

El PanelView utilizado posee una programación previa a la cual se le hicieron modificaciones para adaptarla a la nueva aplicación desarrollada. Se eliminó una alarma y un reloj que fueron configurados en las pantallas de la aplicación anterior. La configuración de entradas y salidas de este dispositivo se establece con los siguientes tags.

	Tag Name	Data Type	Description	Address	Initial Value
1	arranca_PV	Bit	Arranca el motor	I:0/2	0
2	Conteo_PLC	Unsigned Inte	Muestra el dato de conteo del PLC	O:2	0
3	Giro_PV	Bit	Ingresa el sentido d giro desde el PanelView	I:0/0	1
4	ModoFun_PV	Bit	Ingresa el modo de funcionamiento	I:0/1	0
5	NumObjeto_PV	Unsigned Inte	Numero maximo de objetos a ser contados	I:4	0
6	paro_PV	Bit	Orden de paro al motor desde el PanelView	I:0/3	0
7	PLC_velocidad	Unsigned Inte	Muestra la velocidad en el PanelView	O:0	0
8	set_velocidad	Unsigned Inte	Setea velocidad desde PanelView	I:2	0

**Figura 2.57** Tags de PanelView

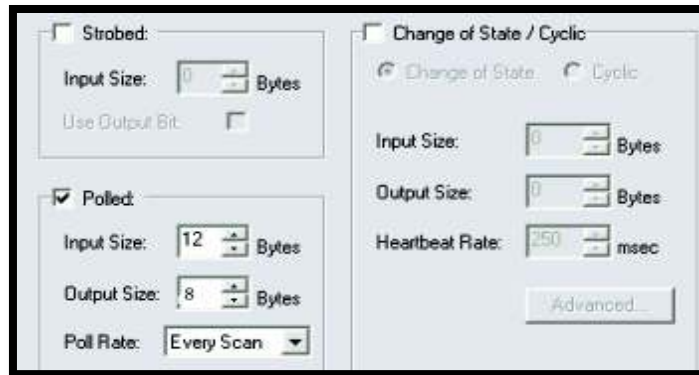
En la parte izquierda de la pantalla del programa PanelBuilder32 se muestra las pantallas creadas de la aplicación. En esta sección se encuentra la opción para realizar la configuración de comunicación a la cual accedemos para configurar las entradas y salidas asignadas al PanelView.



**Figura 2.58** Configuración de Comunicación PanelBuilder32

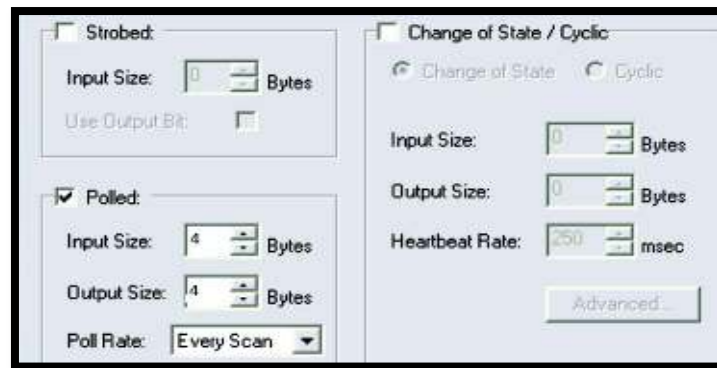
En esta configuración se asigna la dirección de nodo que va a tener el PanelView en la red, la misma que se utilizará al momento de realizar la configuración en el software RSNetWorx.

Se procede a realizar el mapeo de datos en el software RSNetWorx del PanelView, estableciendo como datos de entrada 12 bytes y datos de salida 8 bytes.



**Figura 2.59** Configuración de Parámetros E/S PanelView

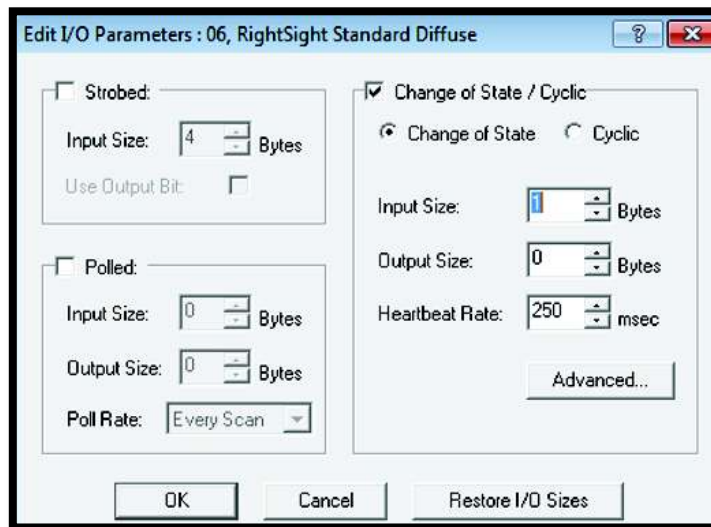
El mapeo de datos del PowerFlex no requiere especificar los datos de entrada y salida pues se encuentran predefinidos.



**Figura 2.60** Configuración de Parámetros E/S PowerFlex

Se detalla a continuación el mapeo del sensor fotoeléctrico “RightSight Standard Difuse” y los tres prototipos implementados.

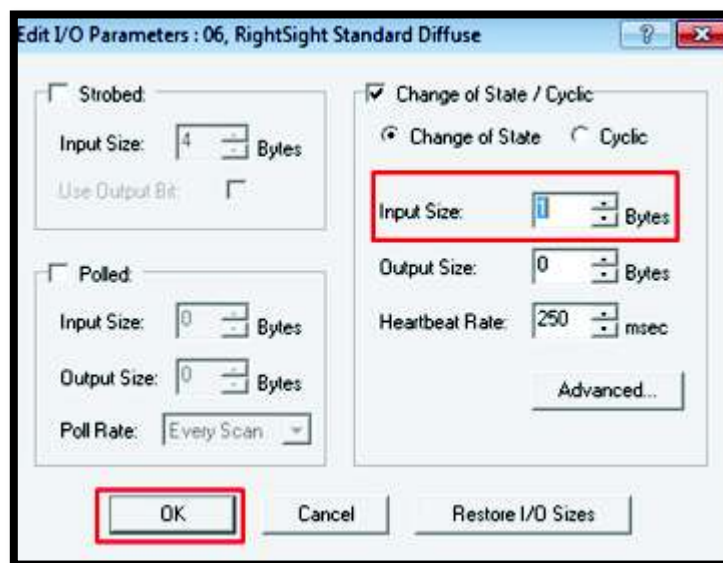
- En el sensor comercial fotoeléctrico “RightSight Standard Difuse” se habilita la opción “Edit I/O Parameters” la seleccionamos.



**Figura 2.61** Edición de los datos de entrada del sensor

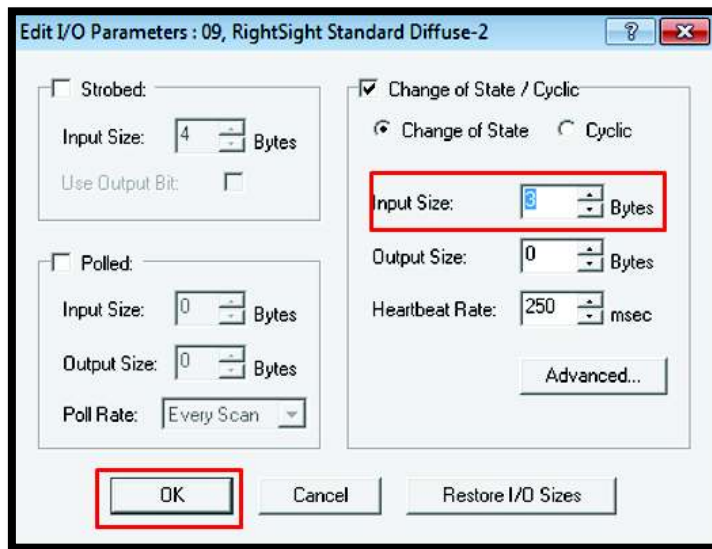
Aparece una ventana donde se puede modificar el tipo de mensaje y el número de entradas y salidas del sensor. Por defecto está habilitado el tipo de mensaje “COS/Cyclic” por lo que se procede a cambiar el número de entradas.

- El sensor comercial y los nodos de nivel y temperatura poseen 1 byte de datos de entrada por lo que se configura en “Input” este valor.



**Figura 2.62** Asignación de 1 byte de entrada del Sensor Comercial

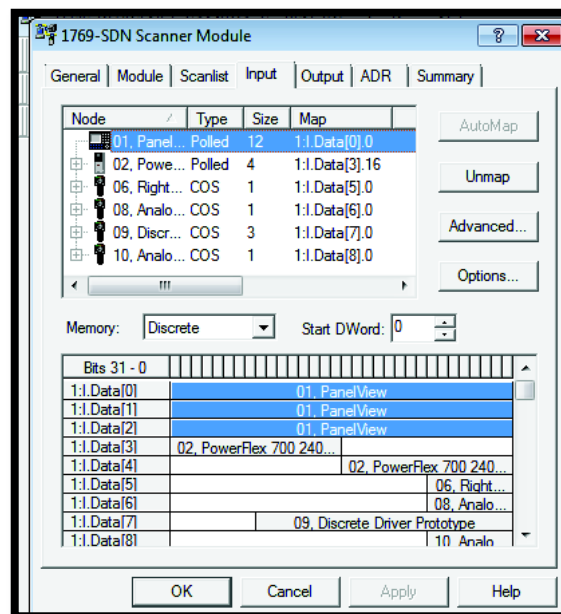
- El mapeo del nodo discreto se realiza de forma similar con una modificación en el campo de “Input” colocamos el valor de 3 bytes.



**Figura 2.63** Asignación de 3 bytes de entrada del nodo Discreto

Una vez realizada la edición de datos de entrada y salida de los dispositivos, se procede a asignar el espacio de memoria en el que se quiere almacenar el o los datos de información para esto nos dirigimos a la pestaña “Input”, seleccionamos el dispositivo que deseamos asignar a la memoria del escáner.

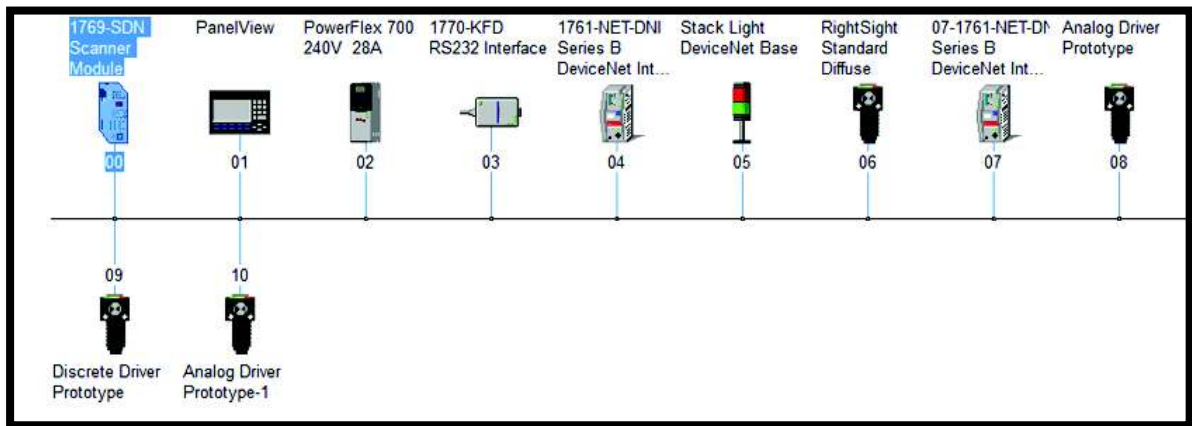
En esta ventana aparece la opción de “Start DWord” la misma que permite ajustar el lugar donde se desea iniciar el mapeo del dato del dispositivo. Una vez seleccionado el lugar de memoria en el escáner se presiona “Automap”.



**Figura 2.64** Asignación de Espacios de Memoria del Escáner

Como se observa en la Figura 2.64 se ha modificado los nombres de los prototipos implementados, esta modificación es posible haciendo doble clic sobre el nombre del dispositivo.

Una vez asignados todos los dispositivos a la memoria del escáner se aplican los cambios seleccionando “Apply” y se presiona “OK” de esta manera queda configurado el escáner de la red DeviceNet.



**Figura 2.65** Diseño Final de la Red DeviceNet con Implementación de Prototipos

### **Aplicación para Prueba de Prototipos Implementados**

Se crea una aplicación con el objetivo de evidenciar el funcionamiento de los sensores implementados junto con los dispositivos comerciales en la red DeviceNet. En la aplicación se desarrolla un proceso para colocar fertilizante en plantas.

En el PanelView se realizará la configuración para mover la banda transportadora, como es sentido de giro, velocidad, arranque y paro del motor que la comanda. Así como también el usuario podrá indicar la cantidad de plantas que desea tener con fertilizante y seleccionar el modo de operación manual y automático.

El modo manual se realizará con la activación del sensor de presencia y el arranque comandado desde el PanelView.

El modo automático se realizará con la activación del sensor de presencia.

Se dispone de un sensor de luz el mismo que con su activación abre una válvula para verter sobre la planta el fertilizante.

El sensor de nivel permite advertir al usuario el nivel bajo y alto del tanque. La alarma de nivel bajo se acciona desde los 5 cm de nivel de líquido y la alarma de nivel alto se activa desde los 25 cm de nivel de líquido.

El tanque se llenará a partir de un nivel de líquido de 4 cm hasta alcanzar los 27 cm con la apertura y cierre de una válvula que se encuentra en la parte superior.

Al final de la banda transportadora se encuentre el sensor comercial de DeviceNet que realiza el conteo del número de plantas que ha sido configurado por el usuario en el PanelView.

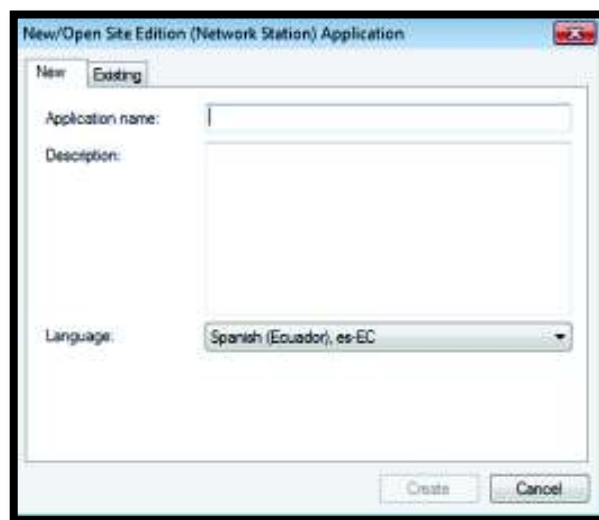
En este proceso se encuentra un sensor de temperatura que en conjunto con el Set Point de temperatura establecido por el usuario en el HMI mandará a accionar un ventilador o un calefactor.

La activación del sensor de humo hace que todo el proceso se pare automáticamente. De igual manera el proceso se detiene si es pulsado en el HMI el botón de paro de emergencia.

El sentido de giro hacia atrás de la banda será posible solo si es accionado desde el HMI para dar mantenimiento a la banda. Este proceso al igual que el paro de emergencia puede realizar solo el usuario que ha sido configurado como supervisor en el HMI.

### **Desarrollo del Interface de Operador con FactoryTalk View Studio**

Se realiza el HMI en el software FactoryTalk View Studio a continuación se detalla su elaboración. Se abre el software FactoryTalk View Studio que se encuentra en Inicio Rockwell Software. Se despliega una ventana inicial en la cual se puede crear una nueva aplicación o abrir una aplicación creada anteriormente. Se crea un nuevo proyecto al cual se le asigna un nombre en "Application Name" y seleccionamos "Create".



**Figura 2.66** Pantalla para crear una nueva aplicación en FactoryTalk View Studio

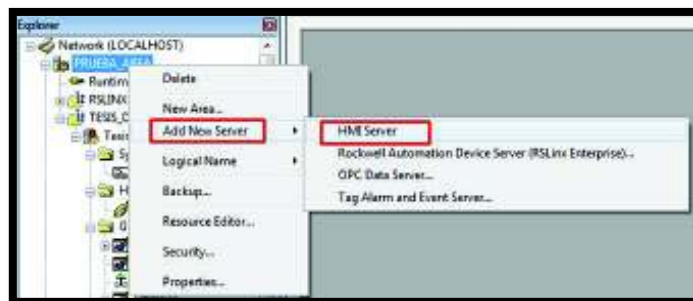
En la Figura 2.67 se muestra el submenú de la pantalla inicial en la que se encuentran diferentes objetos para implementarlos en el HMI.





**Figura 2.67** Submenú de la Pantalla inicial de FactoryTalk View Studio

Se tiene que establecer la comunicación entre el PLC y el HMI desarrollado, para hacerlo en la parte izquierda de la pantalla damos un clic sobre el nombre que se ha asignado en el proyecto, se abre un menú en que seleccionamos “Add New Server” y “HMI Server”.



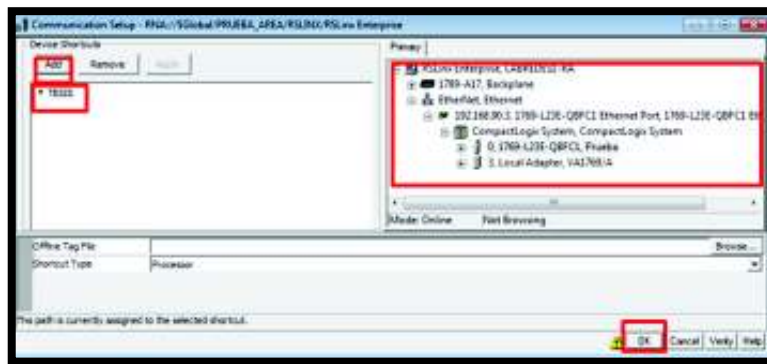
**Figura 2.68** Establecimiento de la comunicación PLC y HMI

Se ingresa a una nueva pantalla en la que seleccionamos “Add” para añadir una nueva comunicación a la que asignamos un nombre. En el lado derecho de esta pantalla se busca el módulo de PLC con el que se realizará la comunicación.



**Figura 2.69** Selección del Módulo de Comunicación

Seguidamente, Hacemos clic en “OK” de esta manera se copian los tags que se encuentran en el programa cargado del PLC para poder usarlos en el HMI.



**Figura 2.70** Adquisición de tags del Módulo PLC

Una vez realizado este procedimiento aparece una ventana en la cual seleccionamos “SI” y se puede dar inicio a la elaboración del HMI.



**Figura 2.71** Finalización de la configuración de Comunicación

A continuación, se detalla la creación de pantallas, botones, tags y la forma para agregar seguridad, animaciones e imágenes a los elementos que forman parte del HMI.

### Creación de Pantallas

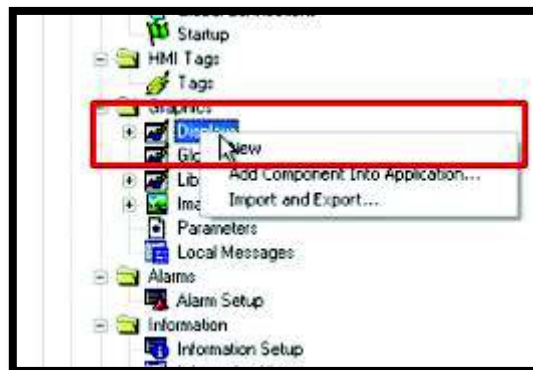
Para crear una pantalla nueva en el lado izquierdo del menú desplegado de FactoryTalk View Studio buscamos “Graphics” damos un clic y se despliega un submenú que contiene Displays, Libraries, Images, entre otros.

En la Figura 2.72 se muestra diferentes herramientas para implementarse en la creación del HMI.



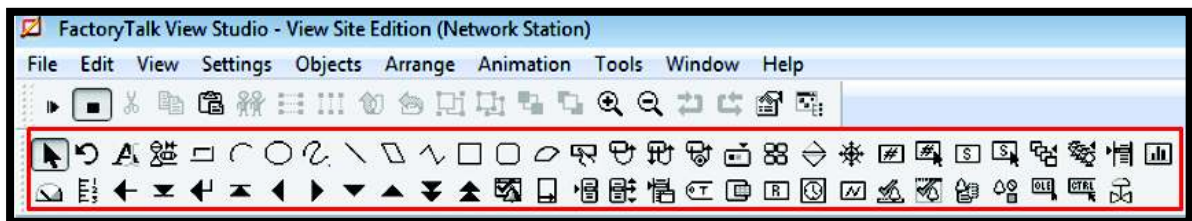
**Figura 2.72** Submenú de opciones para elaboración HMI

Seleccionamos “Displays” con un clic derecho se despliega un submenú en el que seleccionamos “New” y se creará una nueva ventana.



**Figura 2.73** Creación de una nueva ventana

En la parte superior de la pantalla de FactoryTalk se encuentran herramientas que van a permitir insertar texto, imágenes, botones, entre otros.



**Figura 2.74** Barra de Herramientas FactoryTalk

## Insertar Imágenes y Texto

Para introducir texto en el HMI en la barra de Herramientas seleccionamos “Text” y en la pantalla creada marcamos una sección en la que queremos ubicar el texto. Se puede modificar tipo y tamaño de fuente, dar formato al texto, etc.



Figura 2.75 Insertar Texto

Para introducir texto en el HMI en la barra de Herramientas seleccionamos “Image” y en la pantalla creada marcamos una sección en la que queremos ubicar la imagen. Aparece una ventana en la que se selecciona “Add” la que permite que busquemos la dirección en la que se encuentra la imagen que se va a insertar.

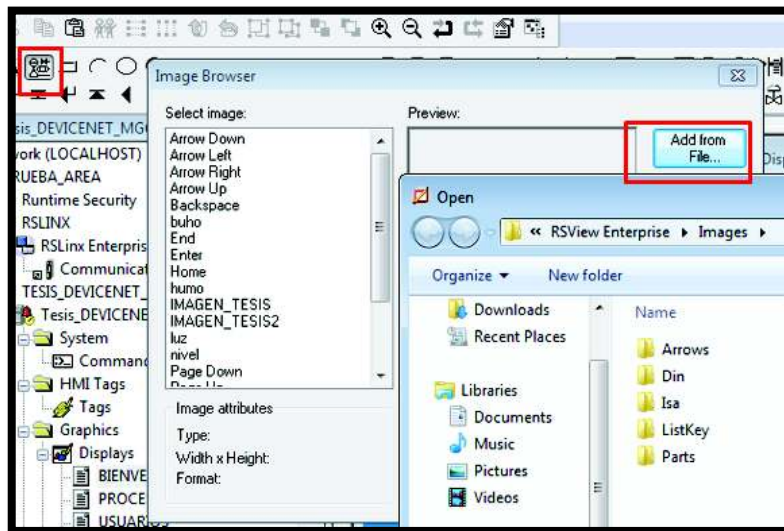


Figura 2.76 Insertar Imagen

## Creación de Botones

Para la aplicación se utilizarán dos tipos de botones “Momentary Push Button” y “Manteined Push Button”. “Momentary Push Button” va a mantener su estado siempre que se lo

mantenga presionado y “Maintened Push Button” basta con pulsarlo una vez para mantener su estado y solo cambiará si se pulsa nuevamente.



**Figura 2.77** Tipos de Botones

### Creación de Tags

En el menú del lado izquierdo de la pantalla se encuentra la opción “HMI Tags” damos un clic y se despliega una ventana de configuración.

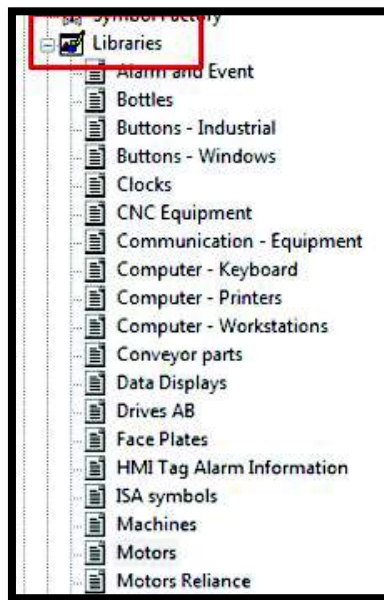


**Figura 2.78** Creación de Tags

En esta ventana podemos configurar la variable especificando nombre, tipo y dirección en el caso de ser una variable que se encuentre en el PLC.

### Selección de Animaciones

Las animaciones se pueden dar a cualquier objeto del HMI. FactoryTalk cuenta con una amplia variedad de objetos que intervienen en procesos industriales, se los puede encontrar en “Libraries”.



**Figura 2.79** Librería de Objetos

Para asignar la animación se da clic derecho sobre el objeto, se despliega un menú en el que se selecciona “Animation” y aparece una lista de efectos como color, movimiento, visibilidad, entre otros que se le puede otorgar al objeto.

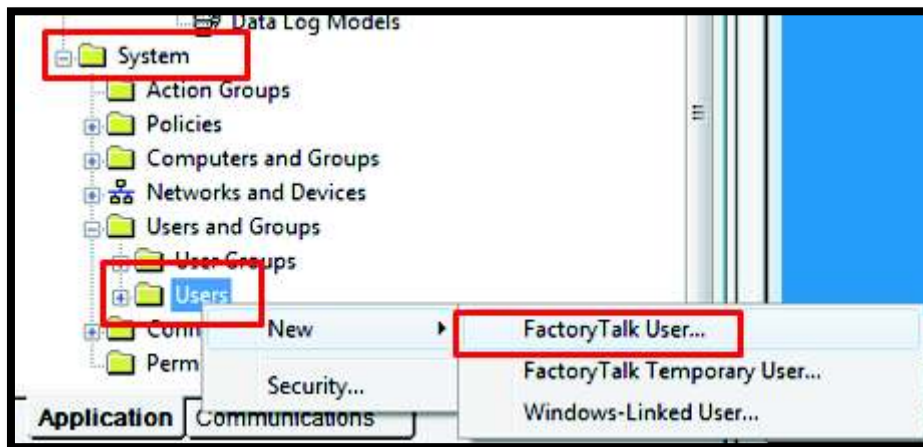


**Figura 2.80** Animaciones para un Objeto

### Asignación de Seguridad

Se puede asignar seguridad para restringir el acceso a determinadas ventanas del HMI configurando usuarios y grupos.

En el menú de la izquierda de la pantalla se encuentra “System” buscamos dentro de esta sección “User”. Dando un clic derecho seleccionamos “New” y “User”.



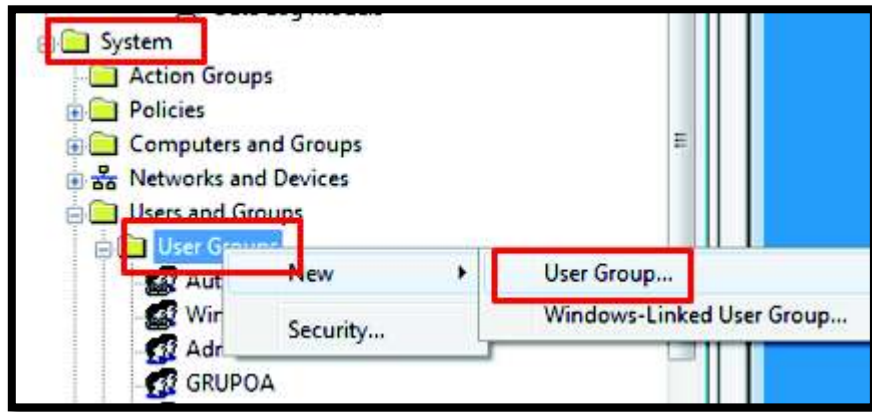
**Figura 2.81** Asignación de Seguridad

Aparece una ventana en la que se escribe el nombre del usuario y se selecciona “Password never expires” para tener una seguridad por niveles, seguidamente se escribe la contraseña de acceso.



**Figura 2.82** Creación de Usuarios

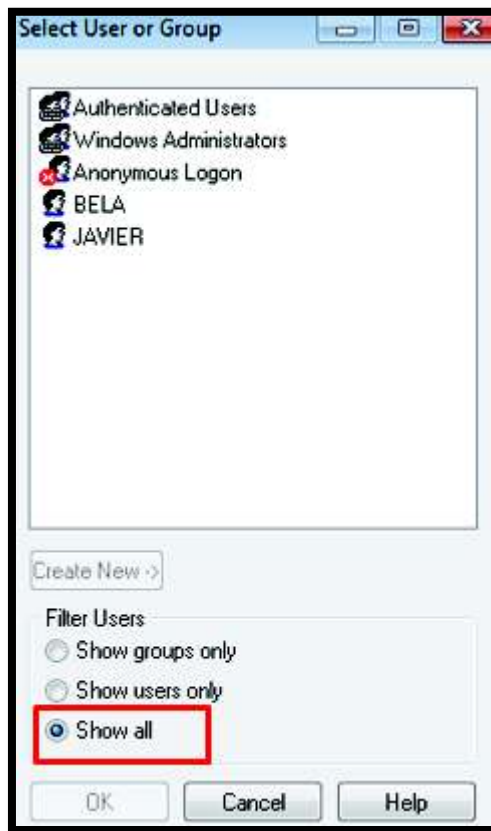
La creación de grupos se realiza en la sección “User Group”, Dando un clic derecho seleccionamos “New” y “User Group”.



**Figura 2.83** Creación de Grupos

Aparece una ventana en la que se escribe el nombre del grupo y se selecciona “Add”, se presenta un submenú en el que se selecciona “Show all” para que se visualicen todos los usuarios creados.

Se seleccionan todos los usuarios que van a pertenecer al grupo y se presiona “OK”.



**Figura 2.84** Asignación de Usuarios a Grupos

En el menú desplegado a la izquierda de la pantalla seleccionamos “Runtime Security”, se despliega una ventana en la que seleccionamos “Security Accounts” para agregar los



grupos de seguridad creados, seleccionamos los grupos y los agregamos con la opción “Add”.



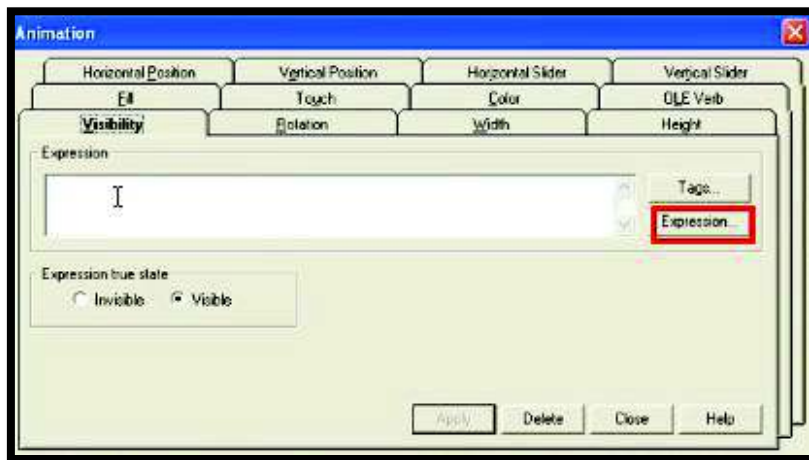
**Figura 2.85** Selección de Grupos de Seguridad

Seguidamente se selecciona el nivel de seguridad, por medio de realizar una selección de códigos de seguridad establecidos por letras para tener acceso a las diferentes pantallas del HMI.



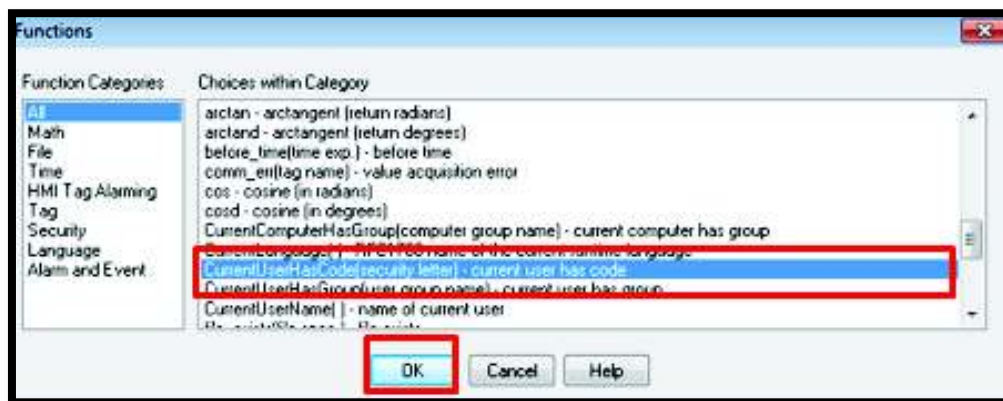
**Figura 2.86** Selección de Códigos de Seguridad

El objeto que va a tener seguridad de acuerdo al nivel debe ser configurado dando un clic derecho sobre el mismo en “Animation” seleccionamos “Visibility” y se despliega una ventana en la que seleccionamos “Expression”.



**Figura 2.87** Configuración de Seguridad de un Objeto

Se despliega una ventana en la que se selecciona “Function” y se selecciona “Security Letter”



**Figura 2.88** Selección de la función de Seguridad por Niveles

Esta función permite especificar, digitando la letra, que nivel va a tener acceso a este objeto.

En la pantalla del HMI se debe establecer dos botones para el ingreso del usuario a la aplicación “Login” y “Logout” que se encuentran en la barra de herramientas, no es necesario realizar ningún tipo de programación adicional pues ya se encuentran preconfigurados.

### Interface de Operador Implementado

El HMI implementado estará formado por 5 ventanas:

- Bienvenida
- Usuarios

- Proceso
- Visualización de Estados de Sensores
- Visualización del Proceso

### Ventana de Bienvenida

Esta ventana se muestra para dar inicio al HMI contiene un botón que permite acceder a la siguiente ventana “Usuarios”.



Figura 2.89 Ventana de Bienvenida HMI

### Ventana de Usuarios

Esta ventana mostrará los botones que dan acceso a las diferentes ventanas del HMI de acuerdo usuario, el mismo que puede cerrar su sesión en cualquier momento habilitando así el ingreso de otro usuario configurado previamente en la aplicación.

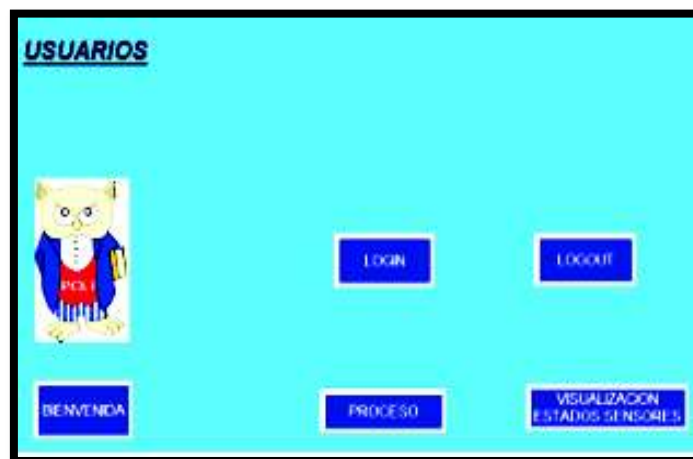


Figura 2.90 Ventana de Usuario HMI

## Ventana de Proceso

Esta ventana permite realizar el control y monitoreo de los sensores que intervienen en el proceso. En esta ventana el usuario puede visualizar los parámetros configurados en el PanelView tales como: objetos contados, velocidad del motor y el accionamiento para el modo de operación “manual”.

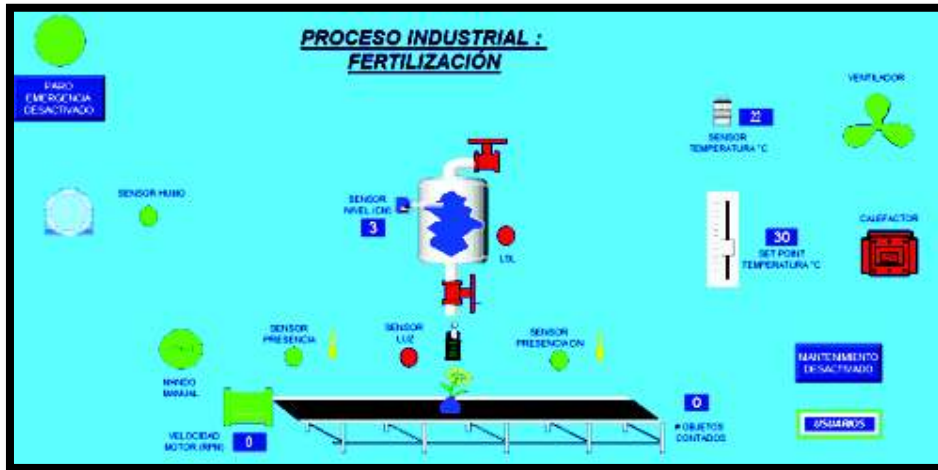


Figura 2.91 Ventana de Proceso HMI

## Ventana de Visualización de Estados de Sensores

Esta ventana permite visualizar el estado de los sensores que intervienen en el proceso. En la parte superior izquierda se muestran los estados de los sensores discretos y en la parte superior derecha e inferior izquierda se muestran los estados de los sensores analógicos de nivel y temperatura respectivamente. El estado del sensor comercial DeviceNet se muestra en la parte inferior derecha de la pantalla.

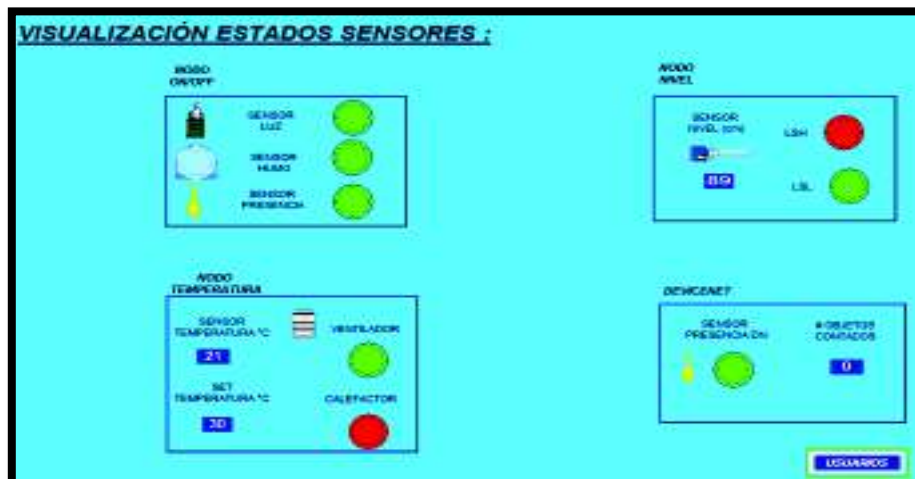
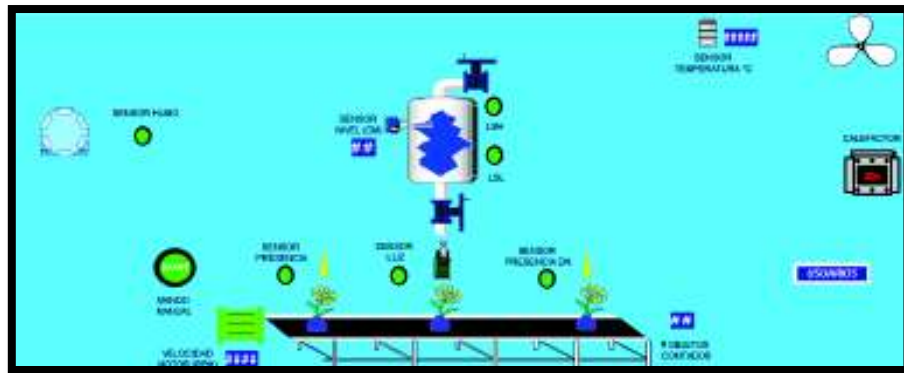


Figura 2.92 Ventana de Visualización de Estados de Sensores HMI

## Ventana de Visualización del Proceso

Esta ventana si bien permite visualizar el proceso no posibilita al usuario a realizar ningún tipo de control, es decir, no puede realizar el ingreso del set point de temperatura, accionar el paro de emergencia o dar mantenimiento a la banda.



**Figura 2.93** Ventana de Visualización de Proceso HMI

### 3. RESULTADOS Y DISCUSIÓN

Los prototipos diseñados en el Capítulo 2 son puestos a prueba haciéndolos trabajar conjuntamente con los dispositivos comerciales de una red industrial DeviceNet. Con el objetivo de observar su correcto funcionamiento se utiliza diferentes softwares de configuración y monitoreo de redes DeviceNet.

#### 3.1 Pruebas de Funcionamiento

En esta sección se detalla las pruebas realizadas para la validación del funcionamiento de los prototipos desarrollados.

##### Pruebas de Hardware

##### Integridad Física

Esta prueba permite establecer la integridad de los prototipos desarrollados, además para garantizar la confiabilidad de las pruebas, estas pruebas se realizan con los equipos de la red DeviceNet comercial a ser usada durante las pruebas.

En esta prueba se verifica que no existan roturas, montaje flojo y falta de soporte. La Tabla 3.1 muestra los parámetros considerados para llevar a cabo la prueba de integridad física.

**Tabla 3.1** Verificación de Integridad Física de la red DeviceNet

Descripción	Ok		Comentario
	Si	No	
Verificar externamente que no existan golpes, roturas, montaje flojo y falta de soporte en los dispositivos comerciales DeviceNet.	X		Se verifica que el PLC CompactLogix con módulo escáner, el sensor fotoeléctrico “RightSight DeviceNet PHOTOSWITCH”, PanelView, PowerFlex, Interfaz RS232/DeviceNet y DNIs se encuentra en buenas condiciones.

Continuación de la Tabla 3.1.

Verificar externamente que no existan golpes, roturas, montaje flojo y falta de soporte en los prototipos implementados.	X		Se verifica que las tarjetas electrónicas de los prototipos se encuentran en buenas condiciones.
Verificar que el módulo permita expansiones de la red DeviceNet.	X		Se verifica que la red permite realizar expansiones para incrementar dispositivos en la línea de derivaciones.

### Conexionado

Esta prueba verifica que la longitud permitida para derivaciones de la red sea la correcta de acuerdo a la velocidad de transmisión de datos.

Las Tablas 3.2 y 3.3 muestran los parámetros considerados para llevar a cabo la prueba de conexionado.

**Tabla 3.2** Determinación de la Longitud de la Línea de Derivación de la red DeviceNet

Velocidad de Transmisión de Datos	Longitud Acumulada de la Línea de Derivación
125 Kbps	156 m
250 Kbps	78 m
500 kbps	39 m

**Tabla 3.3** Verificación de la Longitud de la Línea de Derivación de la red DeviceNet

Descripción	Ok		Comentario
	Si	No	
Velocidad de transmisión 125 Kbps y Longitud Acumulada de la Línea de Derivación 156 m	X		La longitud acumulada de la red es 10 metros, la cual es menor al valor permitido para la velocidad de transmisión establecido.

### Voltajes de Alimentación

Esta prueba permite verificar que todos los elementos de la red DeviceNet prototipos y dispositivos comerciales se enciendan al momento de activar el interruptor correspondiente.

La Tabla 3.4 muestra los parámetros considerados para llevar a cabo la prueba de voltajes de alimentación.

**Tabla 3.4** Verificación de Voltajes de Alimentación

Descripción	Ok		Comentario
	Si	No	
Verificar que todos los dispositivos comerciales DeviceNet como los prototipos implementados se enciendan al accionar el interruptor de 24 VDC.	X		
Verificar que el variador de velocidad (PowerFlex) y el motor se enciendan al accionar el interruptor de 220 VAC.	X		
Verificar los voltajes de los sensores incorporados en los prototipos para su funcionamiento.	X		Nodo de Nivel y Temperatura: 9 VDC  Nodo Discreto: 12 VDC

### Señales de Entrada y Salida

Se verifica que todas las entradas y salidas analógicas y discretas estén funcionando y se podrá visualizar en el programa del PLC y en la interfaz de operador. La Tabla 3.5 muestra los parámetros considerados para llevar a cabo la prueba de señales de entrada y salida.



**Tabla 3.5** Verificación de Señales de Entrada y Salida

Descripción	Ok		Comentario
	Si	No	
Entradas Discretas	X		Prototipos implementados: dato de sensores de luz, presencia y humo. Dispositivos comerciales DeviceNet: PanelView
Salidas Discretas	X		Dispositivos comerciales DeviceNet: PanelView, PowerFlex
Entradas Analógicas	X		Prototipos implementados: dato de sensores de nivel y temperatura Dispositivos comerciales DeviceNet: PowerFlex
Salidas Analógicas	X		Dispositivos comerciales DeviceNet: PowerFlex, PanelView

Todas las señales de entrada y salida tanto analógicas como discretas se activaron y desactivaron en el programa del PLC.

## Prueba de Calibración de Sensores

### Sensor de Nivel

La calibración del sensor de nivel se realizó por medio de la ecuación de una recta tomando valores de voltaje de salida y los centímetros medidos en el tanque.

La Figura 3.1 y la Tabla 3.6 muestran los parámetros considerados para determinar la calibración realizada con el sensor de nivel.

Se realiza el cálculo del error promedio para verifica la correcta calibración de los sensores empleando la ecuación 3.1.

$$Error = \frac{|\Delta x|}{Xo} \times 100\%$$

**Ecuación 3.1** Cálculo del Error Porcentual

Donde:

$\Delta x$ =Representa la variación de la variable medida.

$X_o$ = Representa el valor real de la variable medida.

Se realiza las pruebas con el sensor obteniéndose los siguientes datos:

**Tabla 3.6** Cálculo del Error de Medición del Sensor de Nivel

Altura en el Tanque (cm)	Altura Nodo de Nivel (cm)	Error Porcentual (%)
3	2	3.33
6	5	1.66
9	9	0.00
12	12	0.00
15	15	0.00
18	18	0.00
21	21	0.00

Se obtiene un error entre 3.33% y 0% que es aceptable considerando la calibración de los instrumentos de medición.

### **Sensor de Temperatura**

La calibración del sensor de temperatura se realizó por medio de la ecuación de una recta tomando valores de voltaje de salida y el valor en grados centígrados entregado por una termocupla.

La Figura 3.2 y la Tabla 3.7 muestran los parámetros considerados para determinar la calibración realizada con el sensor de temperatura. Se realiza las pruebas con el sensor obteniéndose los siguientes datos:

**Tabla 3.7** Cálculo del Error de Medición del Sensor de Temperatura

Temperatura Termocupla (°C)	Temperatura Nodo de Temperatura (°C)	Error Porcentual (%)
25	24	4.00
35	34	2.85
45	44	2.22
55	54	1.81

Continuación de la Tabla 3.7.

65	64	1.53
75	74	1.33
85	84	1.17

Se obtiene un error entre 4% y 1.17 % que es aceptable considerando la calibración de los instrumentos de medición.

### Pruebas de Repetibilidad y Resolución

El objetivo es evaluar la repetibilidad y resolución de los sensores analógicos implementados en los prototipos. A continuación, se definen estos conceptos.

**Repetibilidad:** Capacidad que tiene un instrumento de medir valores idénticos aplicando las mismas condiciones a una variable física [7]. Se utiliza la fórmula de la desviación estándar como se muestra en la ecuación 3.2.

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (x_i - \bar{x})^2}{N - 1}}$$

#### Ecuación 3.2 Cálculo de Repetibilidad

Donde:

$\sigma$ =Representa la desviación estándar.

$X_i$ = Representa el valor del conjunto de datos usados para la obtención de la desviación estándar.

$X_o$ = Representa el valor promedio entre todos los datos.

$N$ = Representa el número de datos.

**Resolución:** Incremento mínimo de una variable física que efectúa un cambio representativo en la medida del instrumento [7]. Se utiliza la fórmula de resolución como se muestra en la ecuación 3.3.

$$Resolución = \frac{\Delta X}{2^n - 1}$$

#### Ecuación 3.3 Cálculo de Resolución

Donde:

$\Delta X$ =Representa la variación de la variable medida.

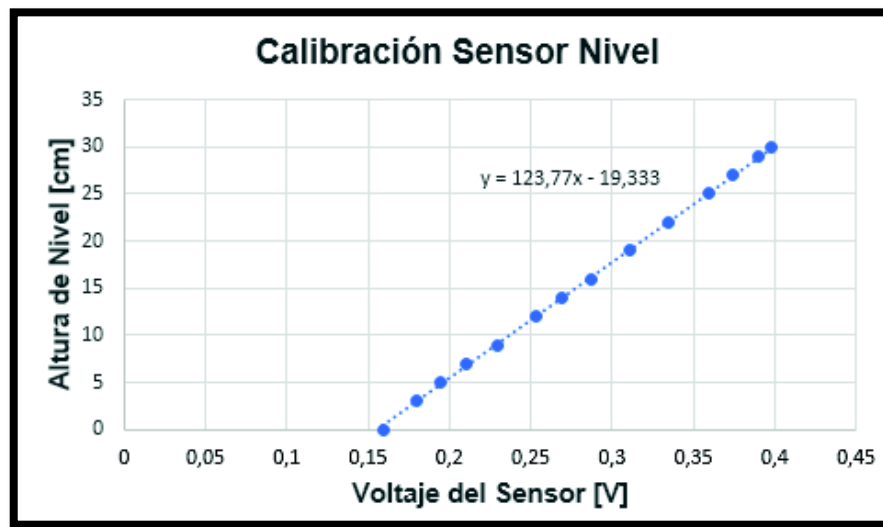
n=Representa el valor del conjunto de datos

### Repetibilidad Nodo de Nivel

La variación del nivel de agua del tanque no representa una variación muy grande a la salida de voltaje del sensor la misma que se encuentra entre un valor de 0.088-4.813 voltios para una altura de 600 cm de nivel de líquido aproximadamente.

Por esta razón las muestras fueron tomadas cada 30 segundos con un voltaje aplicado de 0.2 voltios debido a que la variación entre 0-30 cm del nivel de líquido en el tanque presenta una variación en voltaje de 0.159-0.398 voltios aproximadamente.

El valor esperado para el voltaje aplicado de 0.2 voltios es de 7 cm como se muestra en la Figura 3.1.



**Figura 3.1** Calibración del Sensor de Nivel

La Tabla 3.8 muestra los valores obtenidos por el sensor de nivel para verificar su repetibilidad.

**Tabla 3.8** Repetibilidad del Sensor de Nivel

Número de Muestra	Nivel de Líquido (cm)
1	7
2	6

Continuación de la Tabla 3.8.

3	7
4	7
5	7
6	7
7	7
8	7
9	6
10	7

Con los datos obtenidos se puede ver que el sensor posee una alta repetitividad ya que en el intervalo de  $x = \bar{x} \pm \sigma$  están el 42% de los datos.

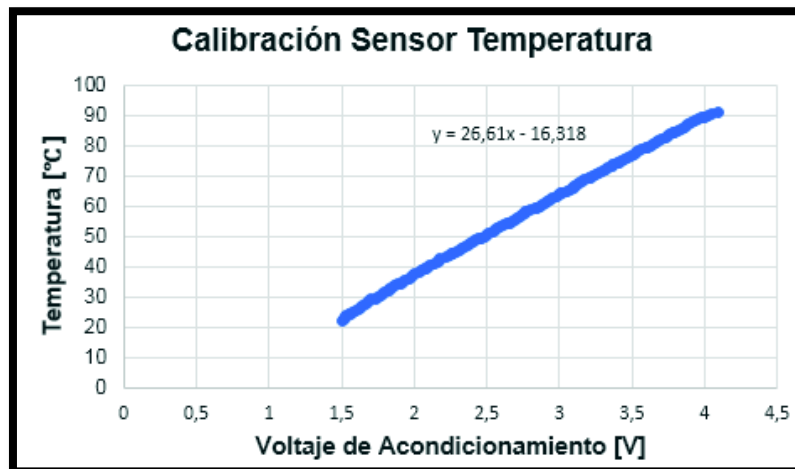
$$\sigma = \sqrt{\frac{\sum_{i=1}^N (x_i - \bar{x})^2}{N - 1}} = \sqrt{\frac{1.6}{9}} = 0.42 = 42\%$$

**Ecuación 3.4** Repetibilidad sensor de Nivel

### Repetibilidad Nodo de Temperatura

Estas muestras fueron tomadas cada 30 segundos con un voltaje aplicado de 2.5 voltios debido a que la variación entre 0-100 °C de temperatura se tiene una variación en voltaje de 0-5 voltios aproximadamente.

El valor esperado para el voltaje aplicado de 2.5 voltios es de 50 °C como se muestra en la Figura 3.2.



**Figura 3.2** Calibración del Sensor de Temperatura

La Tabla 3.9 muestra los valores obtenidos por el sensor de temperatura para verificar su repetibilidad.

**Tabla 3.9** Repetibilidad del Sensor de Temperatura

Número de Muestra	Temperatura (°C)
1	51
2	50
3	50
4	50
5	51
6	50
7	50
8	50
9	50
10	50

Con los datos obtenidos se puede ver que el sensor posee una alta repetitividad ya que en el intervalo de  $x = \bar{x} \pm \sigma$  están el 42% de los datos.

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (x_i - \bar{x})^2}{N - 1}} = \sqrt{\frac{1.6}{9}} = 0.42 = 42\%$$

**Ecuación 3.5** Repetibilidad sensor de Temperatura

### Resolución Nodo de Nivel

La Tabla 3.10 muestra los parámetros considerados para determinar la resolución del sensor de nivel.

**Tabla 3.10** Resolución del Sensor de Nivel

Nivel (cm)	Voltaje (V)	Número de Bits	Resolución (Voltios/Bit)	Resolución (cm/Bit)
0-30	0.159-0.398	10	0,000233	0,029325

### Resolución Nodo de Temperatura

La Tabla 3.11 muestra los parámetros considerados para determinar la resolución del sensor de temperatura.

**Tabla 3.11** Resolución del Sensor de Temperatura

Temperatura (°C)	Voltaje (V)	Número de Bits	Resolución (Voltios/Bit)	Resolución (cm/Bit)
0-100	0-5	10	0,004887	0,097751

## Pruebas de Conectividad

### Configuración de Direccionamiento IP del PLC

Configurar la dirección del PLC de acuerdo a la siguiente tabla. La Tabla 3.12 muestra los parámetros considerados para llevar a cabo la configuración de direccionamiento del PLC.

**Tabla 3.12** Configurar la Dirección del PLC

Descripción	Ok		Comentario
	Si	No	
Dirección IP: 192.168.90.3	X		.
Máscara de Subred: 255.255.255.0	X		

### Conectividad

Se verifica la conectividad entre el PLC y el HMI por lo cual tienen que encontrarse en red. La Tabla 3.13 muestra los parámetros considerados para llevar a cabo la prueba de conectividad.

**Tabla 3.13** Verificación de Conectividad PLC y HMI

Descripción	Ok		Comentario
	Si	No	
Verificar que el PLC se encuentre en red asignada.	X		.
Verificar que la máquina donde se elaboró el HMI se encuentre configurada en la red del PLC.	X		Dirección IP: 192.168.90.10 Máscara de Subred: 255.255.255.0

Continuación de la Tabla 3.13.

Verificar conectividad entre la máquina donde se elaboró el HMI y el PLC hagan ping.	X		
--	---	--	--

La conectividad de dispositivos DeviceNet se realizan en el numeral 3.2.

Se verifica la correcta configuración del escáner considerando los parámetros especificados en la Tabla 3.14.

**Tabla 3.14** Verificación de Configuración del módulo Escáner

Descripción	Ok		Comentario
	Si	No	
Verificar la lista de elementos configurados en el escáner en RSNetWorx.	X		
Verificar la velocidad de intercambio de datos en la red(baudios), dirección de nodo, serie, revisión del escáner.	X		Dirección IP: 192.168.90.10 Máscara de Subred: 255.255.255.0
Verificar que al apagar y encender la fuente de alimentación de 24 VDC del escáner la pantalla no muestre códigos de error.	X		
Verificar que el led indicador de estado "OK" se encuentre de color verde permanente.	X		

### 3.2 Monitoreo de la red DeviceNet con RSNetWorx

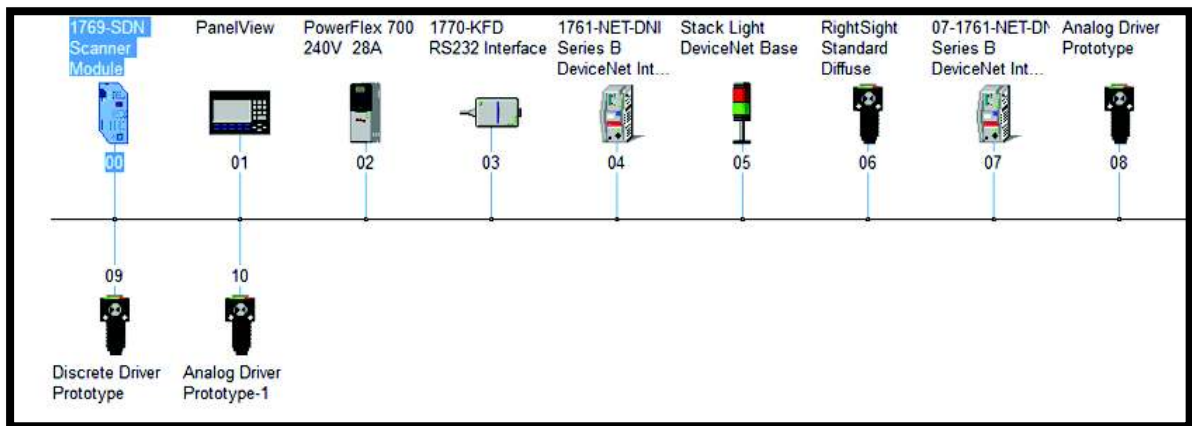
La red DeviceNet con todos los elementos comerciales y los prototipos implementados se muestran a continuación en la Tabla 3.15 y en la Figura 3.3.



**Tabla 3.15** Dispositivos de la Red DeviceNet

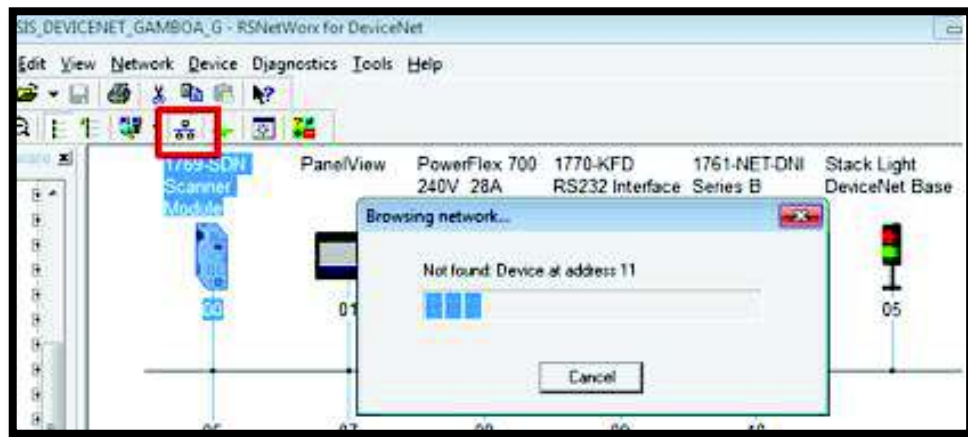
Dispositivo	Descripción de Dispositivo
Sensor fotoeléctrico RightSight DeviceNet PHOTOSWITCH con modo de censado difuso	Dispositivos comerciales
PanelView 300 DeviceNet	
PowerFlex700 DeviceNet	
Interfaz RS232/DeviceNet	
DNIs DeviceNet Interface	
Base para Columna Luminosa	
Prototipos de Drivers Analógicos DNIs DeviceNet Interface	Dispositivos no comerciales: Prototipos de Drivers de Comunicación.
Prototipo de Driver Discreto	

La Figura 3.3 muestra la red DeviceNet con los prototipos implementados.



**Figura 3.3** Red DeviceNet con Prototipos Implementados

Una vez establecida la red se selecciona el botón en línea para realizar el reconocimiento de cada uno de los elementos de la red.



**Figura 3.4** Reconocimiento de los dispositivos de la red

En el software RSNetWorx existen algunos indicadores para los diferentes estados que pueden presentar los dispositivos que conforman la red entre los cuales se tiene “Mismatch”, “Missing” y “Unknow”.

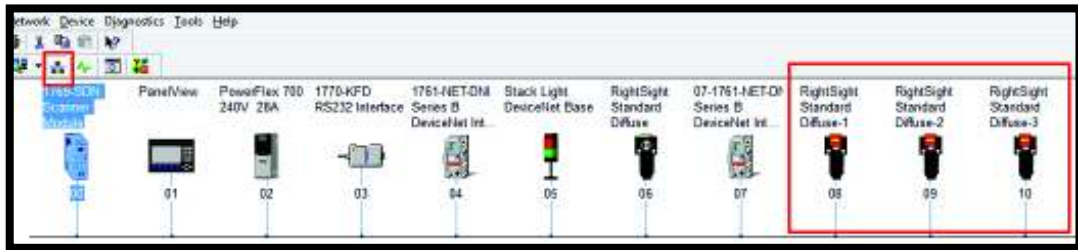
- **Mismatch:** La información de identidad del dispositivo físico no es la misma que la del dispositivo que fue configurado en la red.
- **Missing:** El dispositivo que se configuró en la red no está en línea o no está conectado a la red.
- **Unknow:** El dispositivo que se ha configurado en la red existe en el archivo de configuración de la red, pero el software no lo ha identificado mientras realiza la búsqueda.

En la Figura 3.5 se muestran todos los estados posibles para un dispositivo en el software RSNetWorx.

No icon	Match	The <u>identity</u> information for the device shown in the network configuration is the same as the identity information of the physical device.
	Mismatch	The <u>identity</u> information for the device shown in the network configuration is not the same as the identity information of the physical device.
	Missing	The device shown in the network configuration either is offline or it is not connected to the network.
	Unknown	The device shown in the network configuration exists in the offline configuration file, but the software has not yet identified the device in the browse sequence.
	SNN Error	The Safety Network Number in the safety device is either invalid or does not match the Safety Network Number for the device in the offline project. <b>Tip: This icon only appears in online mode.</b>

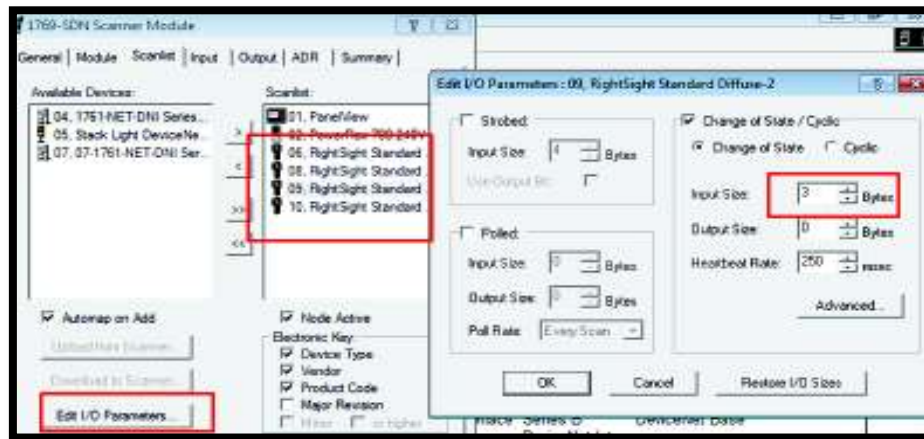
**Figura 3.5** Estados para los dispositivos de la Red

Los prototipos que fueron incluidos en la red DeviceNet presentan el indicativo “Missing”, si bien se realizó el cambio en la hoja de datos electrónica (EDS) del sensor para incorporar los prototipos a la red, la configuración del software no puede ser modificada por el usuario.



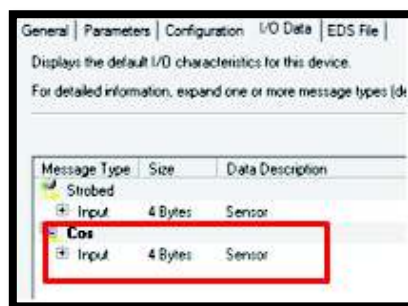
**Figura 3.6** Indicativo de los Prototipo Implementados

A pesar de tener este indicativo es posible realizar la configuración en el escáner de los parámetros I/O.



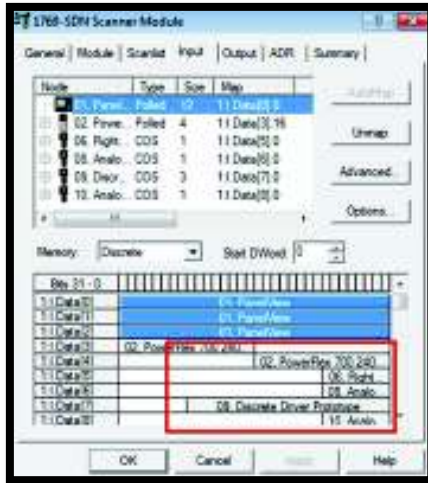
**Figura 3.7** Configuración de parámetros I/O de los Prototipos Implementados

Como se muestra en la figura anterior los parámetros de datos de entrada de los prototipos es configurable por el usuario debido a la modificación realizada al EDS explicada en el capítulo anterior de este trabajo.



**Figura 3.8** Hoja de Datos Electrónica (EDS) Modificada

Los espacios de memoria del escáner para ubicar los datos de los prototipos se muestran a continuación en la Figura 3.9.



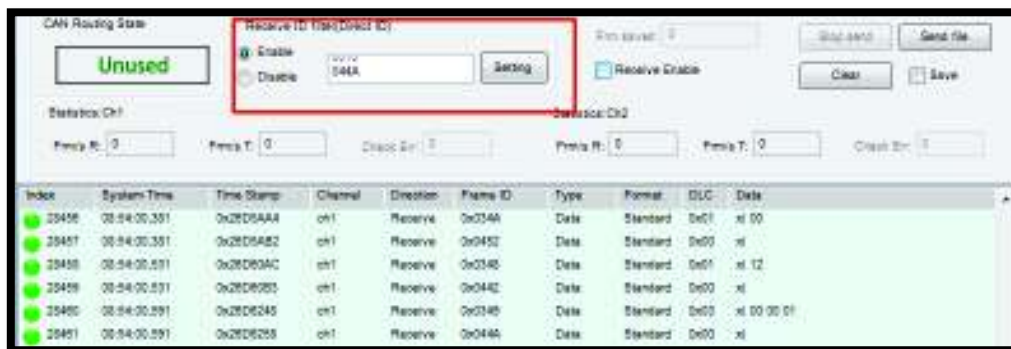
**Figura 3.9** Configuración de mapeo de datos de los Prototipos Implementados

Como se puede evidenciar en la Figura 3.9 los espacios de memoria han sido ubicados de acuerdo a las configuraciones realizadas previamente de los datos de entrada y salida de cada uno de los dispositivos.

Cada localidad de memoria del escáner está formado por un “DWord”, es decir, en una localidad de memoria se puede almacenar hasta 4 bytes de información.

### 3.3 Monitoreo de Trama DeviceNet con USB-CAN Tool V2.02

El reconocimiento de los prototipos por el escáner DeviceNet se muestra a continuación donde se muestra el campo de identificación y los datos enviados entre Maestro/Esclavo. Se utiliza la herramienta de “Receive ID filter(Direct ID)” del software para filtrar solo la información de los prototipos.



**Figura 3.10** Información de la comunicación Maestro/Esclavo de los Prototipos Implementados

La Figura 3.11 y las Tablas 3.16 y 3.17 muestran la verificación de los campos de identificación Maestro/Esclavo de los prototipos implementados para el nodo de nivel.

### Maestro/Esclavo Nodo de Nivel

●	28462	08:54:00,591	0x26D62B9	ch1	Receive	0x034A	Data	Standard	0x01	x  00
●	28463	08:54:00,591	0x26D62BF	ch1	Receive	0x0452	Data	Standard	0x00	x

**Figura 3.11** Información Nodo de Nivel

**Tabla 3.16** Descripción Frame ID Nodo de Nivel (Esclavo)

<b>Maestro/Esclavo</b>	Esclavo
<b>Frame ID Hexadecimal</b>	0x034A
<b>Frame ID Binario</b>	01101001010
<b>Grupo de Mensaje</b>	0
<b>Número de Mensaje</b>	1101
<b>MAC ID</b>	001010
<b>Descripción del Mensaje</b>	Mensaje E/S tipo Cambio de Estado o Cíclico del Esclavo.

**Tabla 3.17** Descripción Frame ID Nodo de Nivel (Maestro)

<b>Maestro/Esclavo</b>	Maestro
<b>Frame ID Hexadecimal</b>	0x0452
<b>Frame ID Binario</b>	10001010010
<b>Grupo de Mensaje</b>	10
<b>MAC ID</b>	001010
<b>Número de Mensaje</b>	010
<b>Descripción del Mensaje</b>	Mensaje ACK tipo Cambio de Estado o Cíclico del Maestro.

La Figura 3.12 y las Tablas 3.18 y 3.19 muestran la verificación de los campos de identificación Maestro/Esclavo de los prototipos implementados para el nodo discreto.

## Maestro/Esclavo Nodo Discreto

●	28460	08:54:00,591	0x26D6248	ch1	Receive	0x0349	Data	Standard	0x03	x  00 00 01
●	28461	08:54:00,591	0x26D6258	ch1	Receive	0x044A	Data	Standard	0x00	x

**Figura 3.12** Información Nodo Discreto

**Tabla 3.18** Descripción Frame ID Nodo Discreto (Esclavo)

<b>Maestro/Esclavo</b>	Esclavo
<b>Frame ID Hexadecimal</b>	0x0349
<b>Frame ID Binario</b>	01101001001
<b>Grupo de Mensaje</b>	0
<b>Número de Mensaje</b>	1101
<b>MAC ID</b>	001001
<b>Descripción del Mensaje</b>	Mensaje E/S tipo Cambio de Estado o Cíclico del Esclavo.

**Tabla 3.19** Descripción Frame ID Nodo Discreto (Maestro)

<b>Maestro/Esclavo</b>	Maestro
<b>Frame ID Hexadecimal</b>	0x044A
<b>Frame ID Binario</b>	10001001010
<b>Grupo de Mensaje</b>	10
<b>MAC ID</b>	001001
<b>Número de Mensaje</b>	010
<b>Descripción del Mensaje</b>	Mensaje ACK tipo Cambio de Estado o Cíclico del Maestro.

La Figura 3.13 y las Tablas 3.20 y 3.21 muestran la verificación de los campos de identificación Maestro/Esclavo de los prototipos implementados para el nodo de temperatura.

## Maestro/Esclavo Nodo de Temperatura

●	28458	08:54:00,531	0x28D80AC	ch1	Receive	0x0348	Data	Standard	0x01	x  12
●	28459	08:54:00,531	0x28D80B3	ch1	Receive	0x0442	Data	Standard	0x00	x

**Figura 3.13** Información Nodo de Temperatura

**Tabla 3.20** Descripción Frame ID Nodo de Temperatura (Esclavo)

<b>Maestro/Esclavo</b>	Esclavo
<b>Frame ID Hexadecimal</b>	0x0348
<b>Frame ID Binario</b>	01101001000
<b>Grupo de Mensaje</b>	0
<b>Número de Mensaje</b>	1101
<b>MAC ID</b>	001000
<b>Descripción del Mensaje</b>	Mensaje E/S tipo Cambio de Estado o Cíclico del Esclavo.

**Tabla 3.21** Descripción Frame ID Nodo de Temperatura (Maestro)

<b>Maestro/Esclavo</b>	Maestro
<b>Frame ID Hexadecimal</b>	0x0442
<b>Frame ID Binario</b>	10001000010
<b>Grupo de Mensaje</b>	10
<b>MAC ID</b>	001000
<b>Número de Mensaje</b>	010
<b>Descripción del Mensaje</b>	Mensaje ACK tipo Cambio de Estado o Cíclico del Maestro.

### 3.4 Monitoreo del HMI con FactoryTalk View Studio

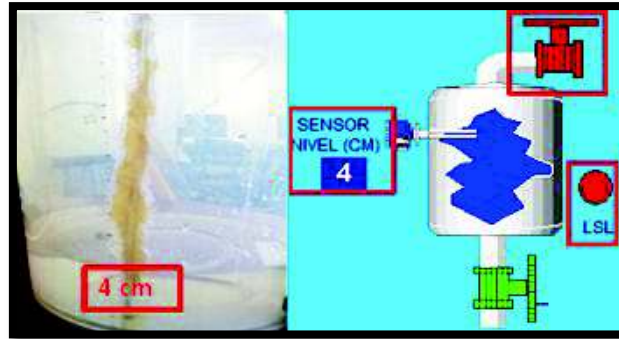
Se realiza las pruebas de funcionamiento comparando las pruebas físicas hechas a los sensores de los prototipos y el HMI diseñado en el software FactoryTalk View Studio.

#### Funcionamiento del Nodo de Nivel

En las siguientes figuras se verifica la calibración que se realizó del sensor de nivel para diferentes alturas de líquido.

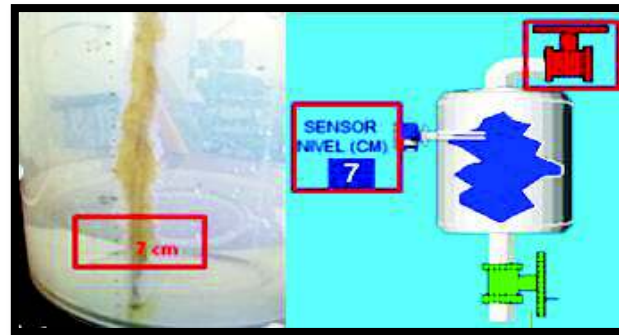
El sensor de nivel permite advertir al usuario el nivel bajo y alto del líquido en el tanque. La alarma de nivel bajo se acciona desde los 5 cm de nivel de líquido y la alarma de nivel alto se activa desde los 25 cm de nivel de líquido. El tanque se llenará a partir de un nivel de líquido de 4 cm hasta alcanzar los 27 cm con la apertura y cierre de una válvula que se encuentra en la parte superior.

**Sensor de Nivel (4cm):** Se activa la alarma de nivel bajo y se realiza la apertura de la válvula para llenar el tanque.



**Figura 3.14** Sensor de Nivel 4cm

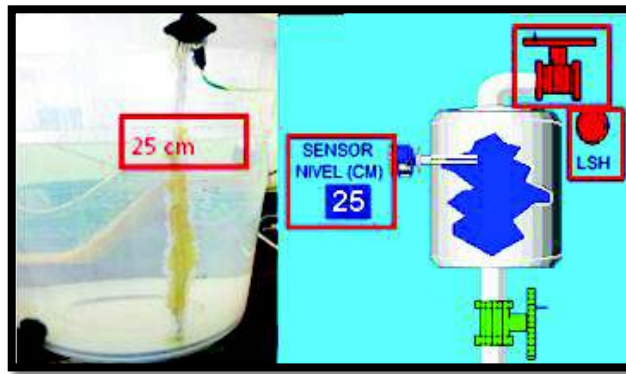
**Sensor de Nivel (7cm):** Se desactiva la alarma de nivel bajo y se mantiene abierta la válvula para llenar el tanque.



**Figura 3.15** Sensor de Nivel 7cm

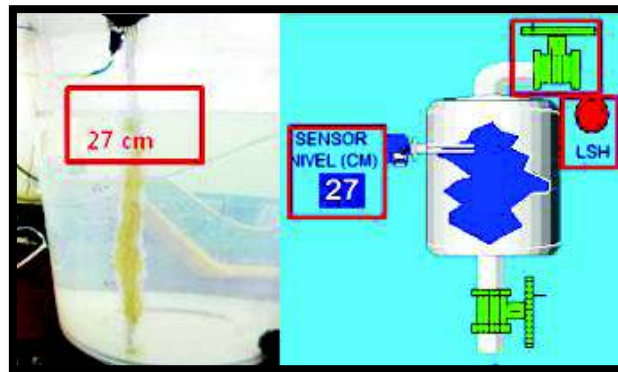
**Sensor de Nivel (25cm):** Se activa la alarma de nivel alto y se mantiene abierta la válvula para llenar el tanque.





**Figura 3.16** Sensor de Nivel 25 cm

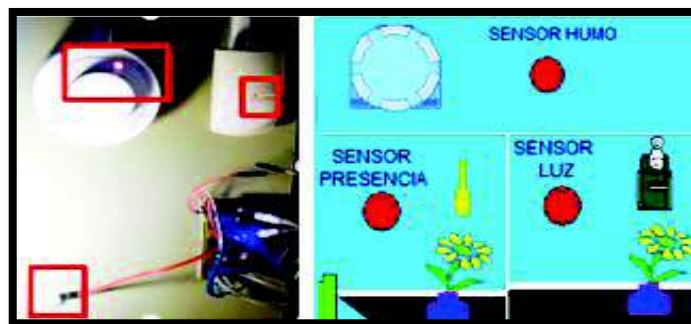
**Sensor de Nivel (27cm):** Se mantiene activada la alarma de nivel alto y se realiza el cierre de la válvula para llenar el tanque.



**Figura 3.17** Sensor de Nivel 27 cm

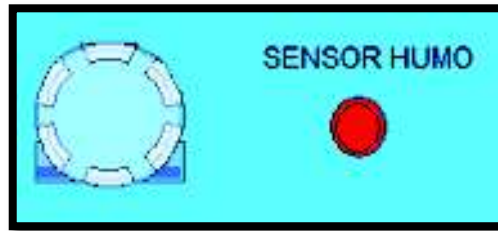
### **Funcionamiento del Nodo Discreto**

Se realiza el accionamiento de los sensores humo, presencia y luz que conforman el nodo discreto.



**Figura 3.18** Activación de los sensores del Nodo Discreto

**Sensor de Humo:** La activación de este sensor detiene todo el proceso.



**Figura 3.19** Sensor de Humo

**Sensor de Presencia:** La activación de este sensor inicia el proceso en modo manual y automático.



**Figura 3.20** Sensor de Presencia

**Sensor de Luz:** La activación de este sensor inicia la apertura de la válvula para colocar el fertilizante en la planta.



**Figura 3.21** Sensor de Luz

## Funcionamiento del Nodo de Temperatura

Para las pruebas de este sensor se utiliza una termocupla para validar la calibración del sensor de temperatura PT100.

**Sensor de Temperatura:** Este sensor en conjunto con el set point de temperatura, establecido por el usuario, permite realizar el accionamiento de un ventilador y un calefactor. Se tiene una histéresis de 4°C, es decir 2°C sobre el valor del set point de temperatura no se acciona el ventilador y 2°C bajo el valor del set point de temperatura no se acciona el calefactor.

**Sensor de Temperatura (22°C):** El set de temperatura se establece en 30°C, al ser superior a la temperatura del sensor se activa el calefactor.

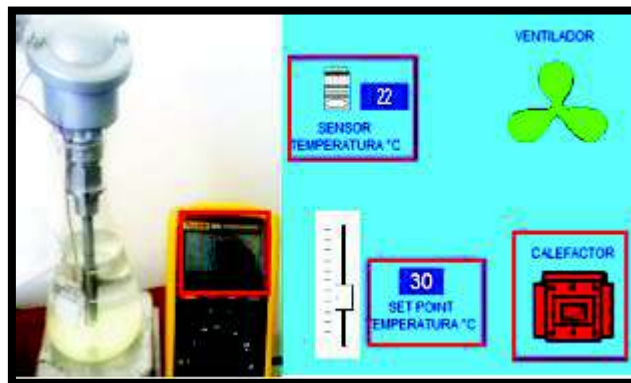


Figura 3.22 Sensor de Temperatura a 22°C

**Sensor de Temperatura (28°C):** El set de temperatura se establece en 30°C, al ser 2°C inferior al valor del set point de temperatura no se acciona el calefactor.

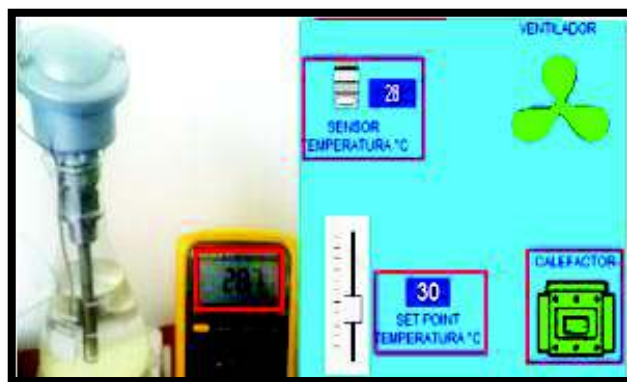
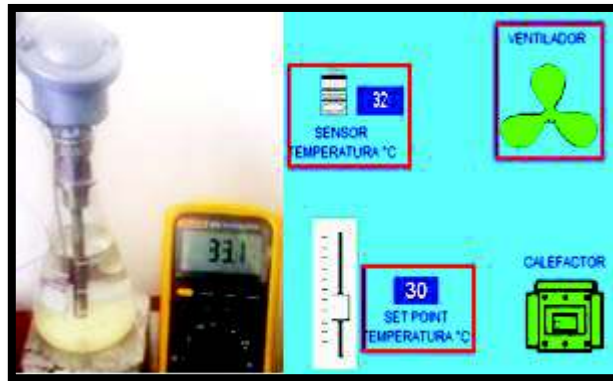


Figura 3.23 Sensor de Temperatura a 28°C

**Sensor de Temperatura (32°C):** El set de temperatura se establece en 30°C, al ser 2°C superior al valor del set point de temperatura no se acciona el ventilador.



**Figura 3.24** Sensor de Temperatura a 32°C

**Sensor de Temperatura (33°C):** El set de temperatura se establece en 30°C, al ser inferior a la temperatura del sensor se activa el ventilador.



**Figura 3.25** Sensor de Temperatura a 33°C

## 4. CONCLUSIONES Y RECOMENDACIONES

### 4.1 Conclusiones

- Es factible implementar drivers de comunicación para una red DeviceNet empleando microcontroladores comerciales que permiten abaratar los costos de sensores con capacidad de comunicación y que pueden ser utilizados en un sistema de control industrial.
- La comunicación Maestro/Esclavo en una red DeviceNet es posible implementando las características de comunicación que deben tener los dispositivos esclavos a través de un lenguaje de programación que sea apropiado para definir el formato, estructura y sintaxis expuestas en las especificaciones del protocolo CIP para ser reconocidos por el dispositivo maestro de la red.
- Durante el desarrollo de este proyecto se estableció las características que el protocolo industrial DeviceNet utiliza del protocolo CAN que forma parte de su capa de enlace de datos para realizar la comunicación entre dispositivos maestro/esclavo tales como grupo de mensaje, MAC ID y número de mensaje que son componentes fundamentales que conforman el campo de identificación.
- Se determinó que existen dos maneras de realizar la comunicación Maestro/Esclavo, el primero es el método de Gestionador de Mensajería No Conectada el cual no se utilizó debido a que requiere una alta capacidad de procesamiento de software para el dispositivo, que en caso de ser dispositivos simples como sensores no justifica su utilización. El segundo método es el Conjunto de Conexión Predefinida Maestro/Esclavo, el cual fue el método empleado para establecer los parámetros que permiten realizar la conexión de comunicación maestro/esclavo sin realizar una carga excesiva a las capacidades de los dispositivos con baja capacidad de procesamiento.
- Las hojas de datos electrónicas (EDS) al tener toda la información sobre las características de los componentes tales como identificación, tipo de dispositivo DeviceNet, Código de Producto, Revisión, Status, Número Serial entre otros, y funciones de un dispositivo electrónico permiten a su vez realizar la modificación de parámetros de acuerdo a los requerimientos de comunicación del dispositivo, entre los más importantes la extensión del campo de datos para entrada y salida. Estas modificaciones son factibles de realizar siguiendo el formato establecido para un

EDS otorgado por el protocolo CIP y adicionando funciones que van a permitir reconfigurar el dispositivo con los cambios efectuados.

- El formato del campo de datos en la trama de comunicación de los prototipos varía dependiendo del estado en el que se encuentre el dispositivo durante el proceso de comunicación, si se encuentra estableciendo comunicación o si ya se ha comunicado con el maestro. En el primer caso el esclavo envía como dato la información que se ha configurado en el EDS y en el segundo caso el esclavo envía el dato analógico o discreto proporcional a la medición según corresponda.
- El correcto funcionamiento de los dispositivos que conforman una red industrial depende de un adecuado manejo de los requerimientos tanto de hardware como de software que son empleados en cada dispositivo. En lo referente al hardware en el caso de los prototipos de comunicación se vio necesario implementar voltajes adecuados para los niveles CAN y el Bus diferencial DeviceNet.
- La comunicación de los prototipos en la red DeviceNet requiere de un correcto establecimiento del formato de la trama y de la especificación del tipo y grupo de mensajes para garantizar la correcta comunicación entre los dispositivos y dentro de uno de los aspectos importantes está la prioridad que se le dé a la información enviada por los dispositivos.
- Existen aplicaciones de software industriales que permiten realizar la configuración y monitoreo de redes DeviceNet, las mismas que permiten verificar el correcto funcionamiento de los dispositivos que la conforman. El software que hizo posible desarrollar el proyecto es RSNetWorx para DeviceNet, el mismo que permite incorporar un EDS de un dispositivo del que se han reconfigurado sus parámetros para adaptarlo a un sistema de control.

## **4.2 Recomendaciones**

- Revisar las condiciones físicas, conexionado y voltajes de alimentación de los dispositivos comerciales y de los prototipos implementados de la red DeviceNet para garantizar su correcto funcionamiento.
- Verificar que se cuente con la instalación del software a utilizarse en el desarrollo de la aplicación tales como: RSNetWorx, RSlinx, RSLogix 5000, FactoryTalk View Studio para poder configurar los dispositivos que forman la red DeviceNet.

- Se recomienda revisar cada una de las especificaciones de los protocolos de comunicación empleados como es DeviceNet y CAN para poder comprender de mejor manera el formato y sintaxis del campo de identificación de esta manera se garantiza el éxito en el desarrollo de prototipos como los implementados.
- Se recomienda revisar otros tipos de redes que forman parte del protocolo CIP tales como ControlNet y Ethernet que utilizan distintos protocolos de comunicación en los diferentes niveles de producción e intégralos a la red DeviceNet para realizar la expansión de la misma y que cumpla con las especificaciones de velocidad de transmisión de datos, longitud de línea troncal como línea de derivaciones, entre otros.
- Se recomienda implementar los requerimientos de comunicación Maestro/Esclavo desarrollados en este trabajo con dispositivos que tengan datos de entrada como datos de salida para determinar si es necesario incorporar nuevos requerimientos para realizar la comunicación de estos dispositivos con la red DeviceNet.
- Se recomienda utilizar conectores y cables comerciales especificados para ser empleados en una red DeviceNet para realizar su expansión evitando desconexiones durante la comunicación del Maestro con los dispositivos Esclavos.

## 5. REFERENCIAS BIBLIOGRÁFICAS

- [1] A. A. Jaramillo Espinoza, «Diseño y construcción de una red de sensores CAN controlados y supervisados a nivel de planta mediante el protocolo Ethernet industrial», 2016.
- [2] V. C. Wilson Rodolfo y N. Nacimba Juan Carlos, «Diseño e implementación de un módulo didáctico para una red DeviceNet», 2010.
- [3] V. Schiffer, «Common Industrial Protocol (CIP™) and the Family of CIP Networks», en Industrial Communication Technology Handbook, Second Edition, vol. 20141360, CRC Press, 2014, pp. 1-100.
- [4] «DeviceNet Protocol Overview», Real Time Automation, 18-06-2014. [En línea]. Disponible en: <https://www.rtaautomation.com/technologies/devicenet/>. [Accedido: 29-may-2018].
- [5] «CIP – The Common Industrial Protocol», Real Time Automation, 21-nov-2017. [En línea]. Disponible en: <https://www.rtaautomation.com/blog/cip-common-industrial-protocol/>. [Accedido: 30-may-2018].
- [6] «The Common Industrial Protocol (CIP) and the Family of CIP Networks» ODVA», Real Time Automation, 26-feb-2016. [En línea]. Disponible en: <https://www.odva.org/Publication-Download/ID/18/The-Common-Industrial-Protocol-CIP-and-the-Family-of-CIP-Networks>. [Accedido: 30-may-2018].
- [7] P. Cartte, «CARACTERÍSTICAS-DE-LOS-INSTRUMENTOS-DE-MEDICION», 11-enero-2016.
- [8] J. Mosso y S. Chaudari, «CIP, El Protocolo hace la Diferencia, Tecnología y Estudio de Caso», en *CIP, Protocolo, Tecnología y Estudio*, Chile, 18-feb-2014.
- [9] M. Collantes y A. López, «Protocolos y Seguridad de red en infraestructuras SCI», vol. 2, España, p. 38, 16-mar-2016.
- [10] M. Andreoni y F. Galdeano, «ETHERNET-ETHERNET/IP», Universidad Nacional de San Juan, San Juan, 19-jun-2015.
- [11] K. Henn, «HIGH SPEED CAN BUS TRANSCEIVER». nov-2000.
- [12] «Integrated Silicon Pressure Sensor On-Chip Signal Conditioned, Temperature Compensated and Calibrated». mar-2010.



- [13] R. Automation, "Módulo escáner Compact I/O DeviceNet". may-2002.
- [14] R. Automation, "PowerFlex 700 AC Drives – Frames 0...10". jul-2014.
- [15] "Sensores– HOGARTECNO – Fibaro en Argentina y Uruguay". [En línea]. Disponible en: <https://www.hogartecno.com/sensor-de-humo/>. [Accedido: 03-sep-2018].
- [16] A. Corporation, "8-bit Microcontroller with 32/64K Bytes of ISP Flash and CAN Controller". sep-2005.

## **6. ANEXOS**

### **ANEXO I**

#### **PROGRAMACIÓN DE MICROCONTROLADORES**

El código implementado en los microcontroladores se encuentra en formato digital.

## ANEXO II

### MANUAL DE USUARIO PARA MÓDULO DE RED DEVICENET

#### CONFIGURACIÓN DEL SERVIDOR DE COMUNICACIÓN RSLINX

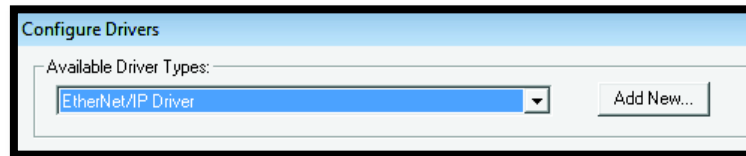
Se abre el software RSLinx que se encuentra en el menú Inicio. Para realizar su configuración se realizan los siguientes pasos:

- En la barra de herramientas se presiona el botón “Configure Drivers”.



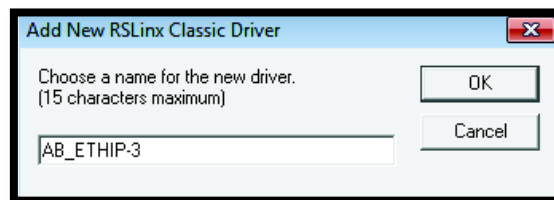
**Figura II.1** Configuración de Controladores

- Aparece una ventana “Configure Drivers” se selecciona el tipo de controlador “Ethernet/IP Driver” y se presiona “Add New”.



**Figura II.2** Selección del tipo de Controlador

- Se despliega un cuadro de diálogo “Add New RSLinx Classic Driver” sin realizar modificaciones se presiona “OK”.



**Figura II.3** Asignación del nombre del Controlador

- Se despliega una ventana en donde se asigna la dirección IP del computador y seguidamente se presiona “Apply”.



**Figura II.4** Configuración de la dirección IP del computador

- En la ventana de “Configure Drivers” aparece el controlador seleccionado anteriormente.

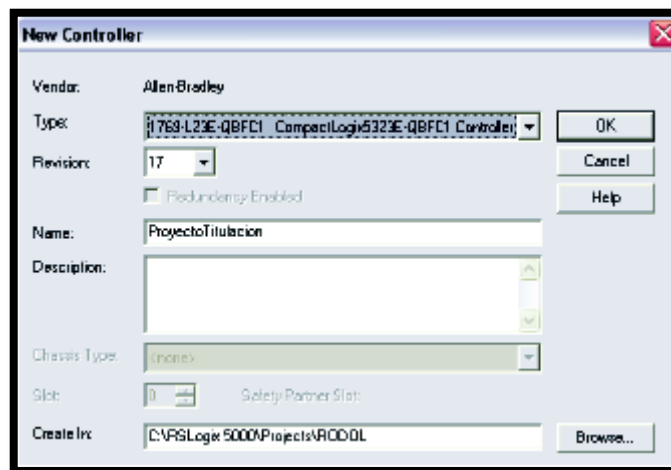


**Figura II.5** Verificación del Controlador

## CONFIGURACIÓN DE LA APLICACIÓN CON RSLOGIX5000

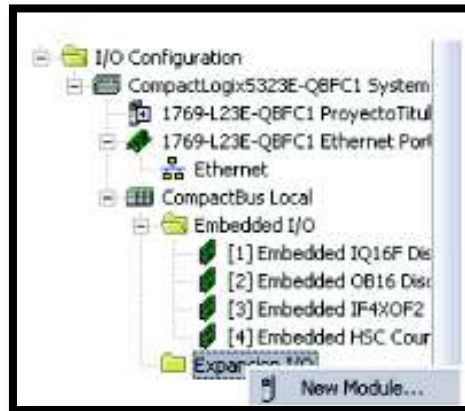
El PLC tiene que reconocer al módulo escáner y esto es posible asignando un slot en su chasis. Se utiliza el software RSLogix5000 para desarrollar la aplicación.

En RSLogix5000 en el menú “File” se selecciona “Create New” aparece una ventana en la cual se selecciona el controlador a utilizarse.



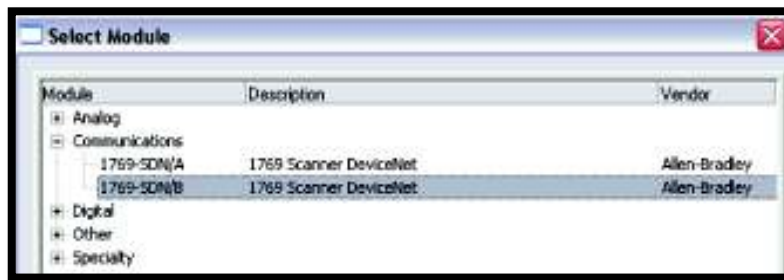
**Figura II.6** Selección del Controlador

En la parte izquierda de la pantalla se encuentra “I/O Configuration”. Se localiza el sistema CompactLogix y seleccionamos “Expansion I/O” damos clic derecho y presionamos “New Module”.



**Figura II.7** Agregar módulo nuevo

Se despliega una ventana en la que se selecciona “Select Module “ y en “Communication” se elige el módulo 1769-SDN serie B y se presiona “OK”



**Figura II.8** Selección del módulo escáner

Aparece un cuadro de diálogo “New Module” en el cual se ingresa el nombre para el módulo y se acepta “OK”.



**Figura II.9** Asignación de nombre al módulo escáner

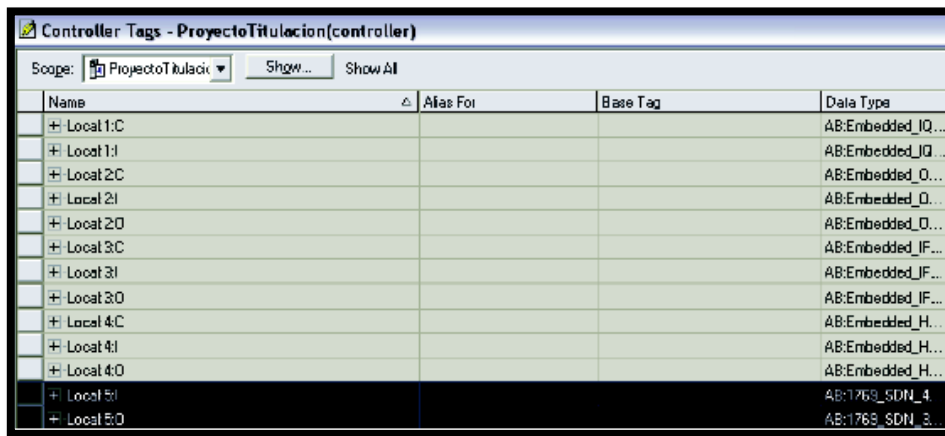
Ya que se ha configurado al módulo escáner para que pueda utilizarse en la aplicación se procede a enrutar el archivo creado en RSNetWorx.

Se presenta una ventana “Module Properties”, en la pestaña “RSNetWorx” se busca la ruta del archivo en el que se configuró la red DeviceNet. Una vez seleccionado el archivo se presiona “OK”.



**Figura II.10** Vinculación de RSLogix5000 con RSNetWorx

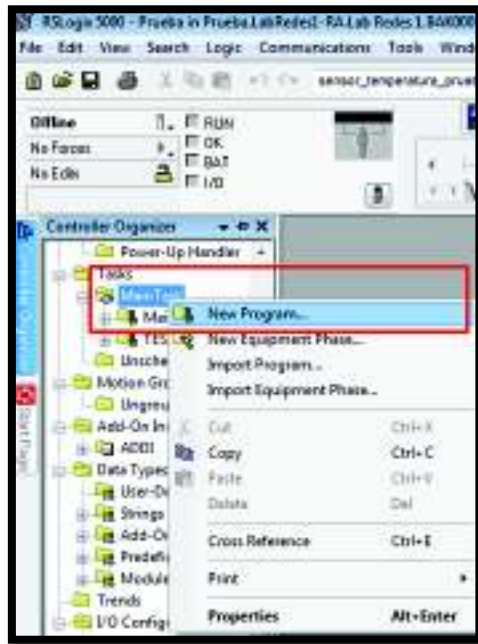
Se procede a crear la aplicación para lo cual se debe considerar que los tags que corresponden al módulo escáner son: Local:5:I entradas y Local:5:O salidas. Es posible visualizarlas en la ventana de la izquierda en “Controller Tags”.



**Figura II.11** Tags correspondientes al Módulo Escáner

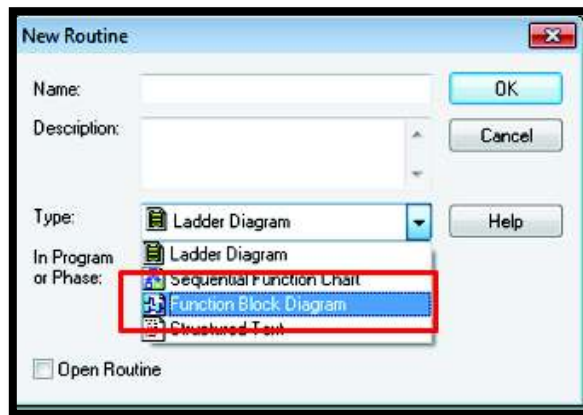
Asignado el módulo escáner en el software RSLogix5000 se desarrolla la lógica de programación en el lenguaje de programación FBD (Function Block Diagram).

Al empezar un proyecto se crean automáticamente archivos o fichas llamadas rutinas, se ubica “MainRoutine” que se encuentra en el submenú de la izquierda de la pantalla, con clic derecho se despliega un submenú en el que se selecciona “New Program”.



**Figura II.12** Creación de una nueva rutina de Programación

Se despliega una ventana en la que se escribe el nombre de la rutina y se elige el tipo de rutina (Ladder,FDB,SFC), seguidamente se presiona “OK “.



**Figura II.13** Selección del tipo de lenguaje de Programación

Se desarrolla la lógica descrita anteriormente en el lenguaje FBD. Terminada la lógica de programación se procede a realizar la descarga del proyecto en el PLC.

## CONFIGURACIÓN DE LA RED CON RSNETWORK

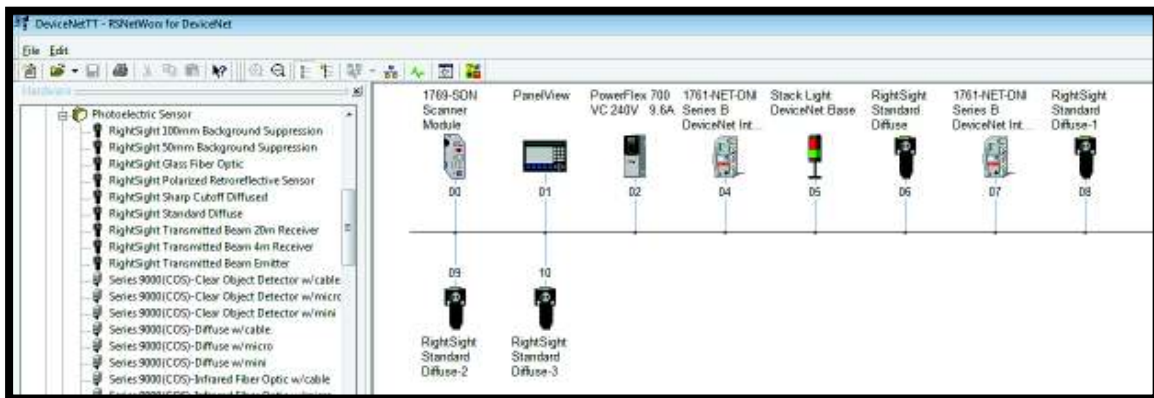
Se abre el software RSNetWorx que se encuentra en la carpeta de programas Rockwell Software.

Se configura cada dispositivo con la dirección con la que va a ser identificado en la red.

Se puede asignar la dirección en el dispositivo haciendo doble clic sobre el mismo y digitando a continuación la dirección que se le desea asignar.

La red DeviceNet tiene los siguientes componentes:

- PLCs MicroLogix con módulo scanner DeviceNet 1769-SDN: Estos dos elementos forman una unidad, y corresponde al dispositivo maestro de la red.
- 4 Sensores fotoeléctricos RightSight DeviceNet PHOTOSWITCH con modo de censado difuso: Dispositivos esclavos, 3 de estos sensores corresponden a los prototipo implementados.
- PanelView 300 DeviceNet: Dispositivo esclavo con datos de entrada y salida.
- PowerFlex700 DeviceNet (interfaz 20-COMM-D): Variador de velocidad, dispositivo esclavo.
- Interfaz RS232/DeviceNet: Permite tener la conexión del computador personal a la red, este dispositivo no forma parte de los esclavos de la red.
- DNIs DeviceNet Interface: Permite la comunicación entre dispositivos con protocolo RS-232.
- Base para Columna Luminosa. Dispositivo esclavo.



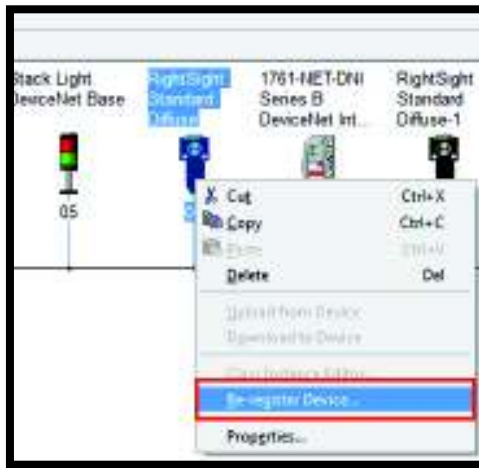
**Figura II.14** Representación de la red DeviceNet en el software RSNetWorx

### **Configuración de la Hoja de Datos Electrónica (EDS) en el software RSNetWorx**

Para implementar una EDS en un dispositivo a través del software RSNetWorx se debe realizar los siguientes pasos:

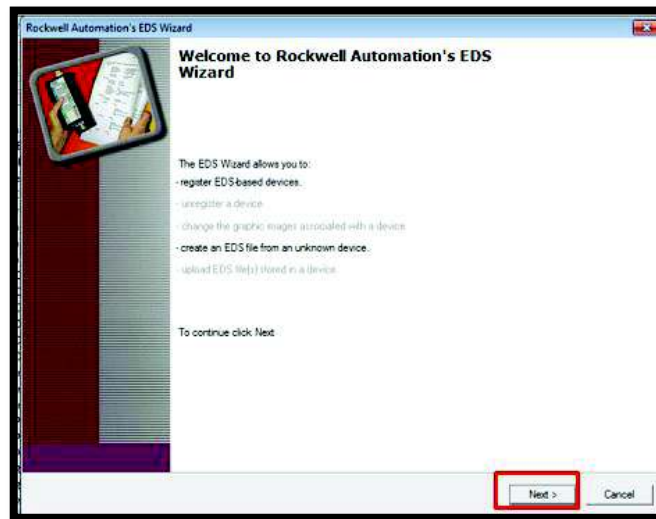
- Se hace clic derecho sobre el dispositivo y se despliega un submenú en el que se selecciona “Re-register Device”.





**Figura II.15** Registro del nuevo EDS

- Se despliega un cuadro de diálogo en la que seleccionamos “Next”.



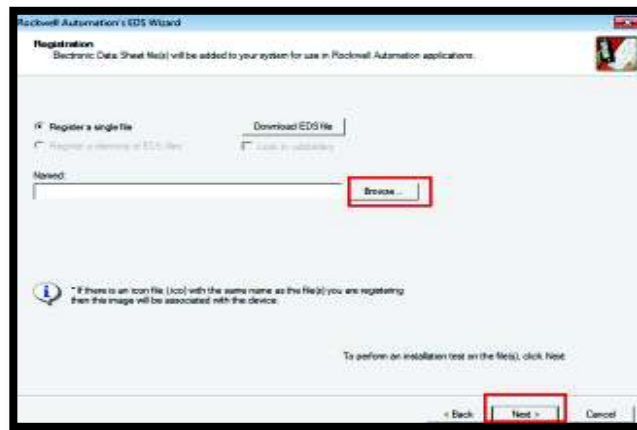
**Figura II.16** Inicio de carga del archivo EDS

- Se despliega una ventana en la que seleccionamos “Register an EDS files” y presionamos “Next”.



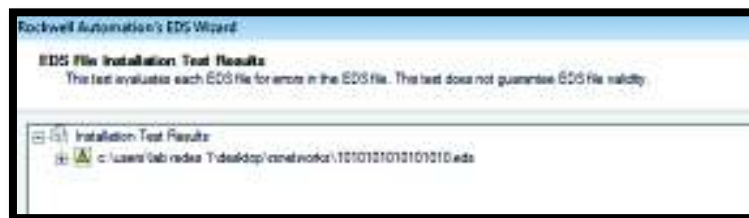
**Figura II.17** Registro de un archivo EDS

- A continuación, seleccionamos en “Browse” el archivo que se desea implementar y se presiona “Next”.



**Figura II.18** Selección de la ruta de archivo EDS

- Se presiona “Next” en las siguientes ventanas que se despliegan.



**Figura II.19** Finalización de la carga del archivo EDS

- Para guardar los cambios realizados en el sensor con su nueva EDS se cierra RSNetWorx y se lo vuelve abrir.



**Figura II.20** Finalización de la configuración del EDS del sensor

## MAPEO DE DATOS CON RSNETWORX

Se realiza en el software RSNetWorx el mapeo de datos para asignar a cada uno de los elementos que intercambian datos en la red una localidad de memoria del módulo escáner.

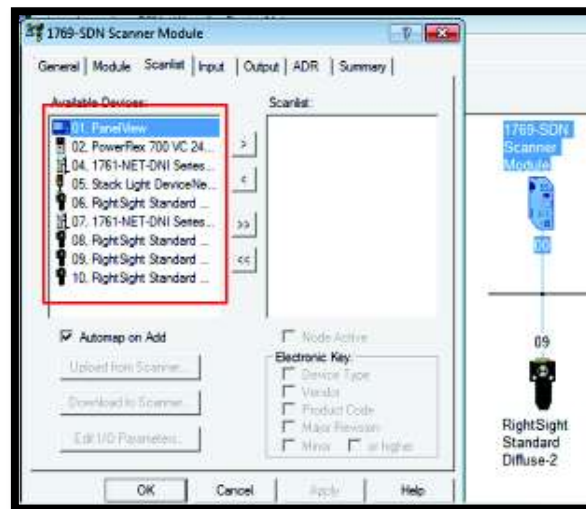
Los dispositivos que serán mapeados son el PanelView, PowerFlex y el sensor “RightSight Standard Difuse” y los 3 prototipos implementados.

Para mapear estos dispositivos se presiona el botón en “línea” y aparecerá un cuadro de diálogo que se acepta.



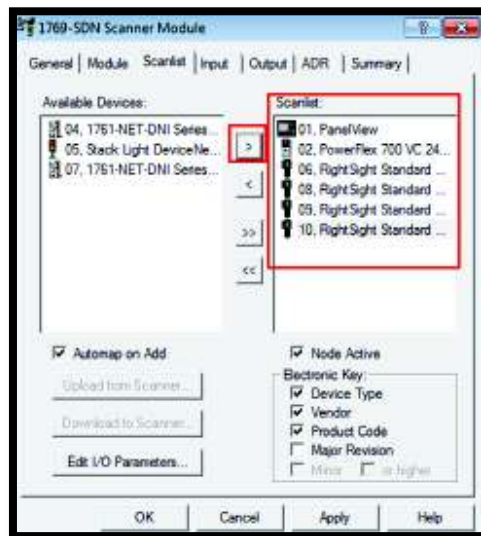
**Figura II.21** Acceso de RSNetWorx a la red

Se inicia de esta manera un barrido de todos los nodos de la red. Para ingresar a las localidades de memoria del escáner se hace doble clic sobre la figura y se ingresa a la pestaña “Scanlist”.



**Figura II.22** Ingreso a Scanlist

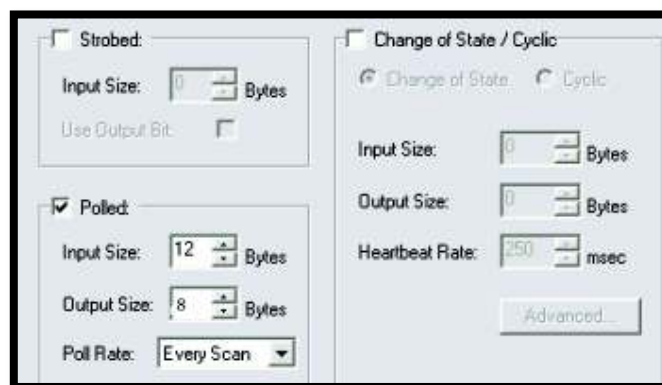
En esta ventana aparecen todos los dispositivos de la red y se agrega uno a uno los dispositivos antes mencionados.



**Figura II.23** Dispositivos a ser Mapeados

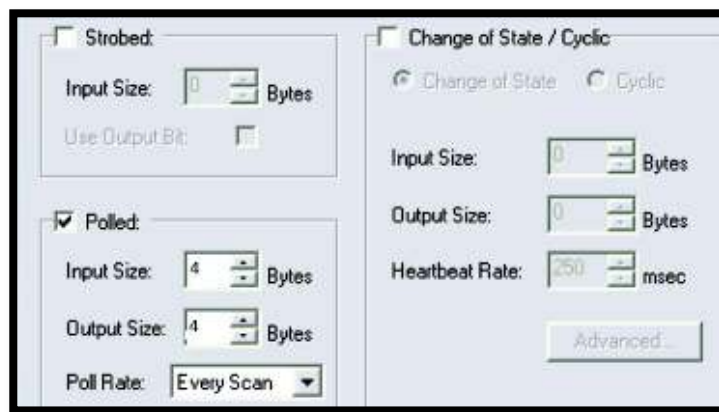
Finalizada la selección de los dispositivos se ingresa el valor de entradas y salidas de cada uno. Los datos de entrada y salida del PanelView deben ser configurados siempre por el usuario.

Se procede a realizar el mapeo de datos en el software RSNNetWorx del PanelView, estableciendo como datos de entrada 12 bytes y datos de salida 8 bytes.



**Figura II.24** Configuración de Parámetros E/S PanelView

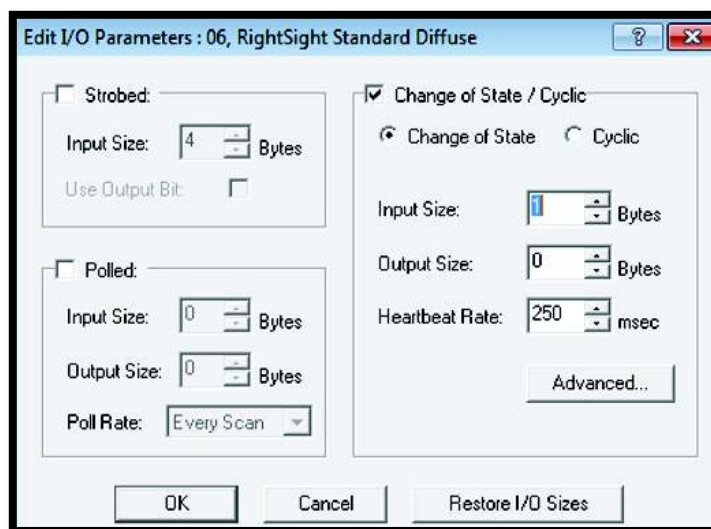
El mapeo de datos del PowerFlex no requiere especificar los datos de entrada y salida pues se encuentran predefinidos.



**Figura II.25** Configuración de Parámetros E/S PowerFlex

Se detalla a continuación el mapeo del sensor “RightSight Standard Difuse” y los 3 prototipos implementados.

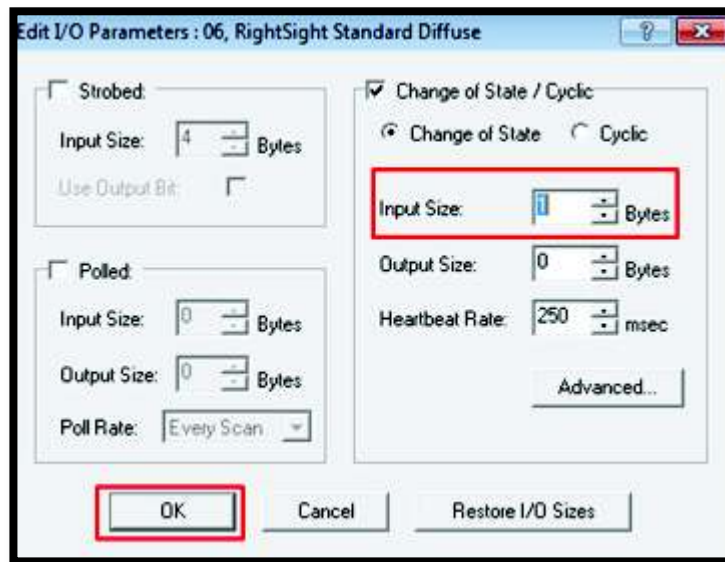
En el sensor comercial fotoeléctrico “RightSight Standard Difuse”. Se habilita la opción “Edit I/O Parameters” la seleccionamos.



**Figura II.26** Edición de los datos de entrada del sensor

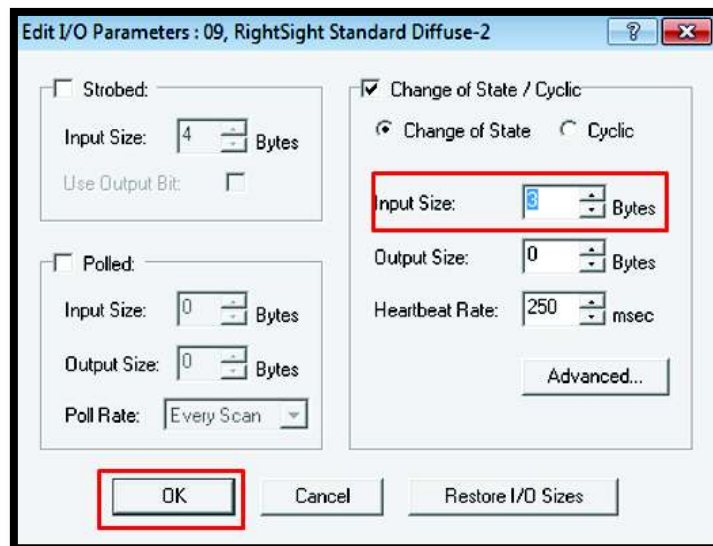
Aparece una ventana donde se puede modificar el tipo de mensaje y el número de entradas y salidas del sensor. Por defecto está habilitado el tipo de mensaje “COS/Cyclic” por lo que se procede a cambiar el número de entradas.

- El sensor comercial y los nodos de nivel y temperatura poseen 1 byte de datos de entrada por lo que se configura en “Input” este valor.



**Figura II.27** Asignación de 1 byte de entrada del Sensor Comercial

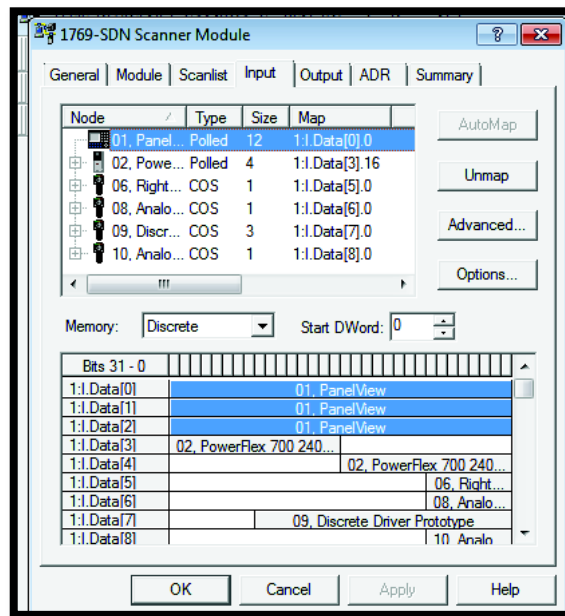
- El mapeo del nodo discreto se realiza de forma similar con una modificación en el campo de “Input” colocamos el valor de 3 bytes.



**Figura II.28** Asignación de 3 bytes de entrada del nodo discreto

Una vez realizada la edición de datos de entrada y salida de los dispositivos, se procede a asignar el espacio de memoria en el que se quiere almacenar el o los datos de información para esto nos dirigimos a la pestaña “Input”, seleccionamos el dispositivo que deseamos asignar a la memoria del escáner.

En esta ventana aparece la opción de “Start DWord” la misma que permite ajustar el lugar donde se desea iniciar el mapeo del dato del dispositivo. Una vez seleccionado el lugar de memoria en el escáner se presiona “Automap”.



**Figura II.29** Asignación de Espacios de Memoria del Escáner

Asignados todos los dispositivos a la memoria del escáner se aplican los datos “Apply” y se presiona “OK”.

## **DESARROLLO DEL INTERFACE DE OPERADOR CON FACTORYTALK VIEW STUDIO**

Se crea una aplicación con el objetivo de evidenciar el funcionamiento de los sensores implementados junto con los dispositivos comerciales en la red DeviceNet. En la aplicación se desarrolla un proceso para colocar fertilizante en plantas.

En el PanelView se realizará la configuración para mover la banda transportadora, como es sentido de giro, velocidad, arranque y paro del motor que la comanda. Así como también el usuario podrá indicar la cantidad de plantas que desea tener con fertilizante y seleccionar el modo de operación manual y automático.

El modo manual se realizará con la activación del sensor de presencia y el arranque comandado desde el PanelView.

El modo automático se realizará con la activación del sensor de presencia.

Se dispone de un sensor de luz el mismo que con su activación abre una válvula para verter sobre la planta el fertilizante.

El sensor de nivel permite advertir al usuario el nivel bajo y alto del tanque. La alarma de nivel bajo se acciona desde los 5 cm de nivel de líquido y la alarma de nivel alto se activa desde los 25 cm de nivel de líquido.

El tanque se llenará a partir de un nivel de líquido de 4 cm hasta alcanzar los 27 cm con la apertura y cierre de una válvula que se encuentra en la parte superior.

Al final de la banda transportadora se encuentre el sensor comercial de DeviceNet que realiza el conteo del número de plantas que ha sido configurado por el usuario en el PanelView.

En este proceso se encuentra un sensor de temperatura que en conjunto con el Set Point de temperatura establecido por el usuario en el HMI mandará a accionar un ventilador o un calefactor.

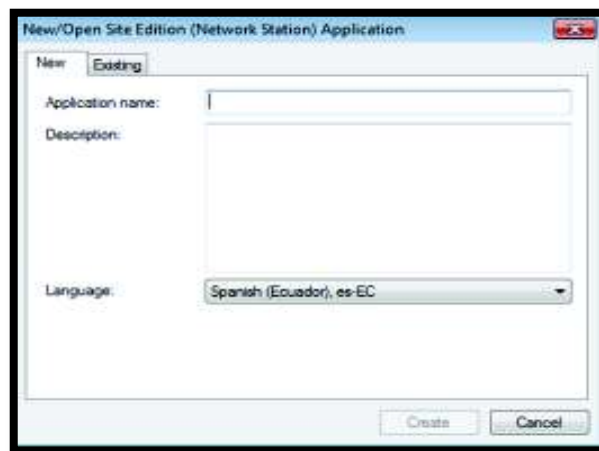
La activación del sensor de humo hace que todo el proceso se pare automáticamente. De igual manera el proceso se detiene si es pulsado en el HMI el botón de paro de emergencia.

El sentido de giro hacia atrás será posible solo si es accionado desde el HMI para dar mantenimiento a la banda. Este proceso puede realizar solo el usuario que ha sido configurado como supervisor en el HMI.

Se realiza el HMI en el software FactoryTalk View Studio a continuación se detalla su elaboración.

Se abre el software FactoryTalk View Studio que se encuentra en el menú Inicio en la carpeta de programas de Rockwell Software.

Se despliega una ventana inicial en la cual se puede crear una nueva aplicación o abrir una aplicación ya creada anteriormente. Se crea un nuevo proyecto al cual se le asigna un nombre en "Application Name" y seleccionamos "Create".



**Figura II.30** Pantalla para crear una nueva aplicación en FactoryTalk View Studio

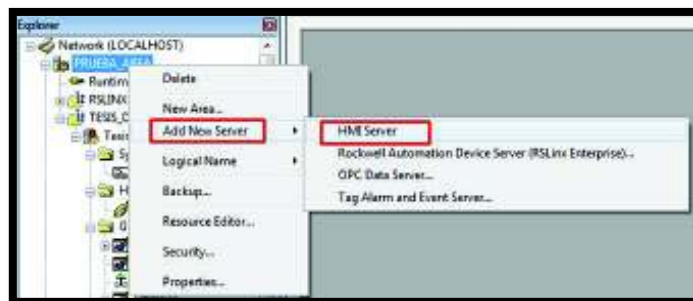
En la Figura I.31 se muestra el submenú de la pantalla inicial en la que se encuentran diferentes objetos para implementarlos en el HMI.





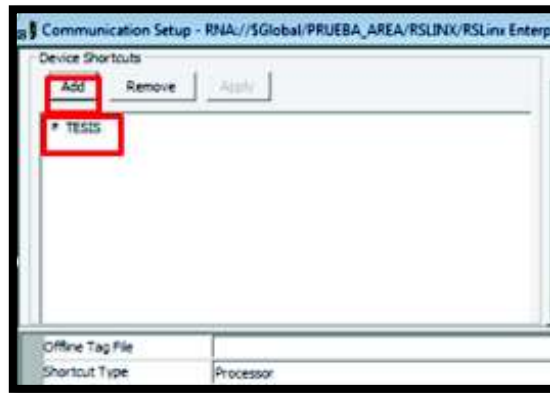
**Figura II.31** Submenú de la Pantalla inicial de FactoryTalk View Studio

Se tiene que establecer la comunicación entre el PLC y el HMI desarrollado, para hacerlo en la parte izquierda de la pantalla damos un clic sobre el nombre que se ha asignado en el proyecto, se abre un menú en que seleccionamos “Add New Server” y “HMI Server”.



**Figura II.32** Establecimiento de la comunicación PLC y HMI

Se ingresa a una nueva pantalla en la que seleccionamos “Add” para añadir una nueva comunicación a la que asignamos un nombre. En el lado derecho de esta pantalla se busca el módulo de PLC con el que se realizará la comunicación.



**Figura II.33** Selección del Módulo de Comunicación

Seguidamente, hacemos clic en “OK” de esta manera se copian los tags que se encuentran en el programa cargado del PLC para poder usarlos en el HMI.



**Figura II.34** Adquisición de tags del Módulo PLC

Una vez realizado este procedimiento aparece una ventana en la cual seleccionamos “SI” y se puede dar inicio a la elaboración del HMI.

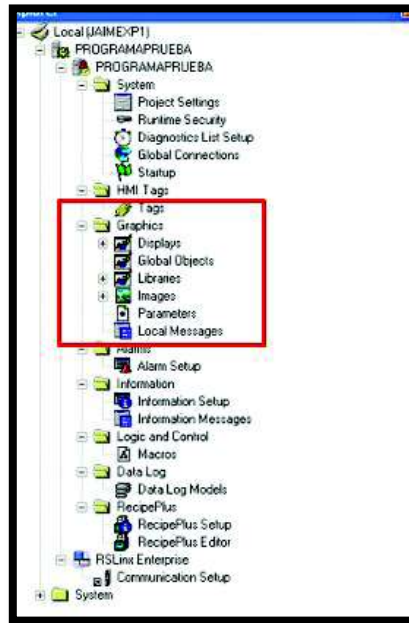


**Figura II.35** Finalización de la configuración de Comunicación

A continuación, se detalla la creación de pantallas, botones, tags y la forma para agregar seguridad, animaciones e imágenes a los elementos que forman parte del HMI.

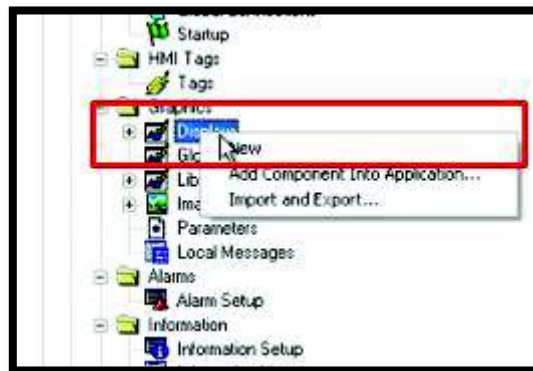
## Creación de Pantallas

Para crear una pantalla nueva en el lado izquierdo del menú desplegado de FactoryTalk View Studio buscamos “Graphics” damos un clic y se despliega un submenú que contiene Displays, Libraries, Images, entre otros.



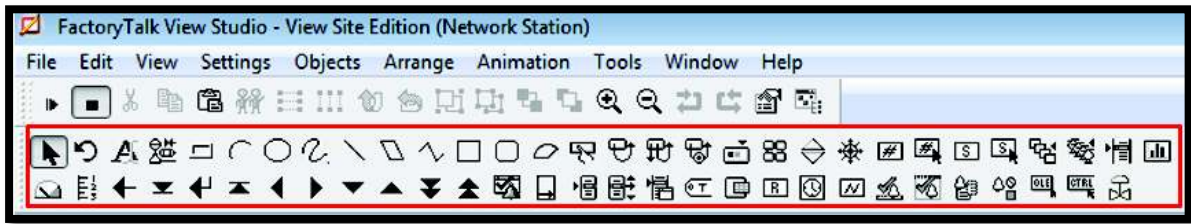
**Figura II.36** Submenú de opciones para elaboración HMI

Seleccionamos “Displays” con un clic derecho se despliega un submenú en el que seleccionamos “New” y se creará una nueva ventana.



**Figura II.37** Creación de una nueva ventana

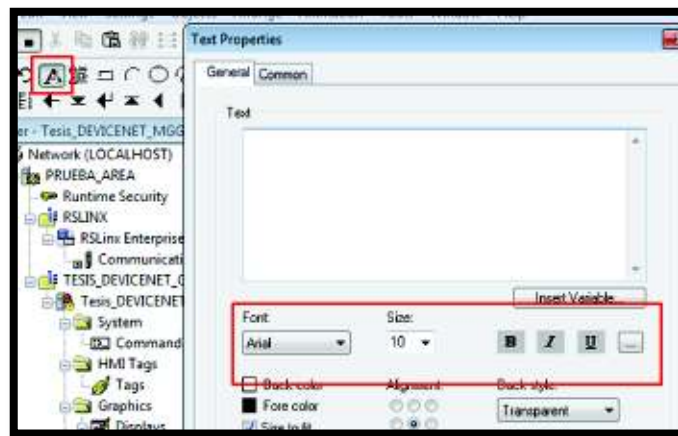
En la parte superior de la pantalla de FactoryTalk se encuentran herramientas que van a permitir insertar texto, imágenes, botones, entre otros.



**Figura II.38** Barra de Herramientas FactoryTalk

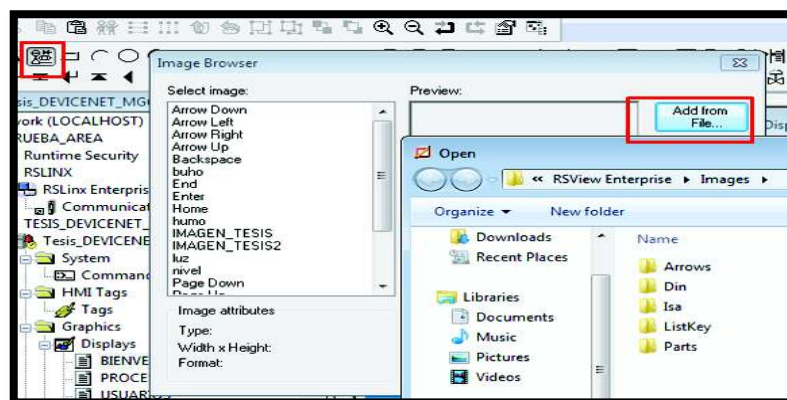
### Insertar Imágenes y Texto

Para introducir texto en el HMI en la barra de Herramientas seleccionamos “Text” y en la pantalla creada marcamos la sección en la que queremos ubicar el texto. Se puede modificar tipo y tamaño de fuente, dar formato al texto, etc.



**Figura II.39** Insertar Texto

Para introducir imágenes en el HMI en la barra de Herramientas seleccionamos “Image” y en la pantalla creada marcamos una sección en la que queremos ubicar la imagen. Aparece una ventana en la que se selecciona “Add” la que permite que busquemos la dirección en la que se encuentra la imagen que se va a insertar.



**Figura II.40** Insertar Imagen

## Creación de Botones

Para la aplicación se utilizarán dos tipos de botones “Momentary Push Button” y “Manteined Push Button”. “Momentary Push Button” va a mantener su estado siempre que se lo mantenga presionado y “Manteined Push Button” basta con pulsarlo una vez para mantener su estado y solo cambiará si se pulsa nuevamente.



Figura II.41 Tipos de Botones

## Creación de Tags

En el menú del lado izquierdo de la pantalla se encuentra la opción “HMI Tags” damos un clic y se despliega una ventana de configuración.

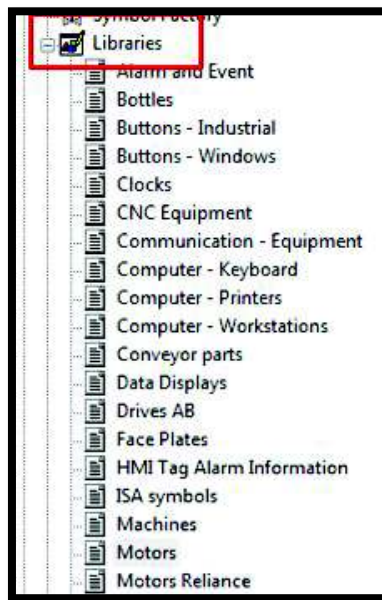


Figura II.42 Creación de Tags

En esta ventana podemos configurar la variable especificando nombre, tipo y dirección en el caso de ser una variable que se encuentre en el PLC.

## Selección de Animaciones

Las animaciones se pueden dar a cualquier objeto del HMI. FactoryTalk cuenta con una amplia variedad de objetos que intervienen en procesos industriales y se los puede encontrar en “Libraries”.



**Figura II.43** Librería de Objetos

Para asignar la animación se da clic derecho sobre el objeto, se despliega un menú en el que se selecciona “Animation” y aparece una lista de efectos como color, movimiento, visibilidad, entre otros que se le puede otorgar al objeto.

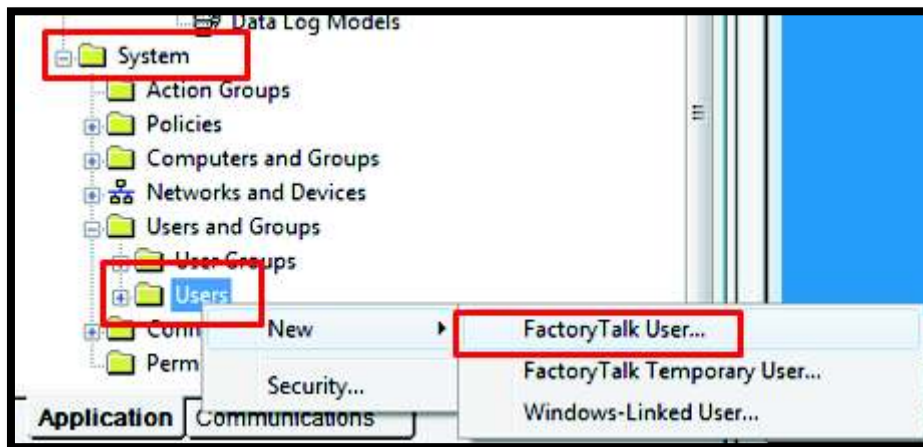


**Figura II.44** Animaciones para un Objeto

### Asignación de Seguridad

Se puede asignar seguridad para restringir el acceso a determinadas ventanas del HMI configurando usuarios y grupos.

En el menú de la izquierda de la pantalla se encuentra “System” buscamos dentro de esta sección “User”. Dando un clic derecho seleccionamos “New” y “User”.



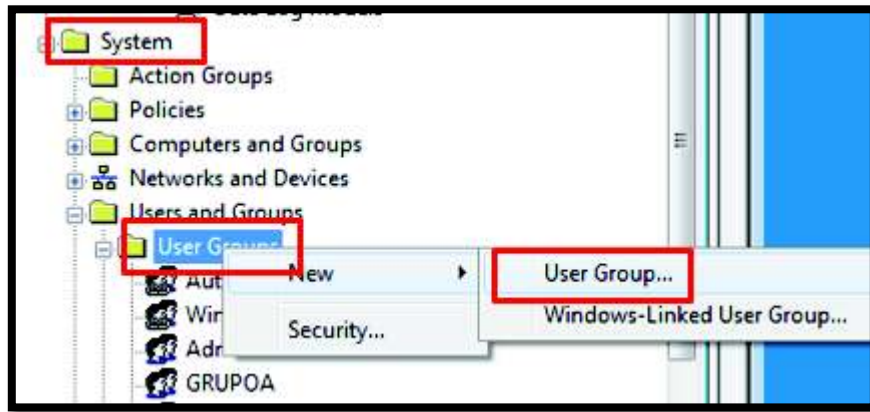
**Figura II.45** Asignación de Seguridad

Aparece una ventana en la que se escribe el nombre del usuario y se selecciona “Password never expires” para tener una seguridad por niveles, seguidamente se escribe la contraseña de acceso.



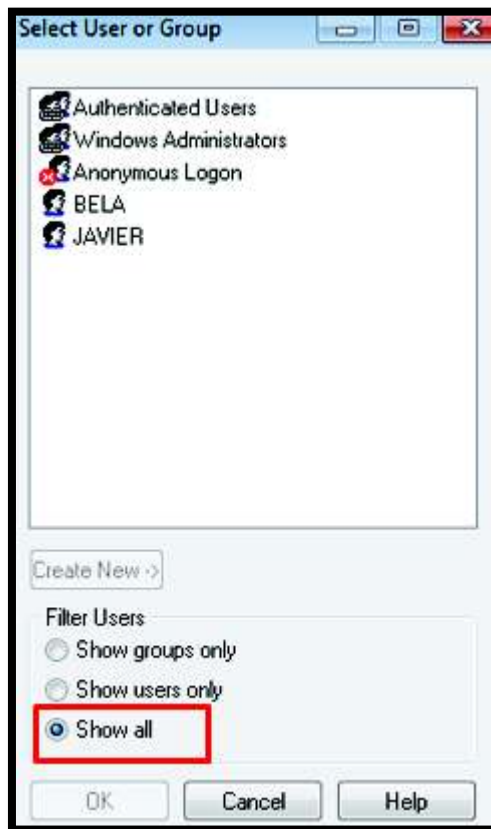
**Figura II.46** Creación de Usuarios

La creación de grupos se realiza en la sección “User Group”, Dando un clic derecho seleccionamos “New” y “User Group”.



**Figura II.47** Creación de Grupos

Aparece una ventana en la que se escribe el nombre del grupo y se selecciona “Add”, se presenta un submenú en el que se selecciona “Show all” para que se visualicen todos los usuarios creados. Se seleccionan todos los usuarios que van a pertenecer al grupo y se presiona “OK”.



**Figura II.48** Asignación de Usuarios a Grupos

En el menú desplegado a la izquierda de la pantalla seleccionamos “Runtime Security”, se despliega una ventana en la que seleccionamos “Security Accounts” para agregar los



grupos de seguridad creados, seleccionamos los grupos y los agregamos con la opción “Add”.



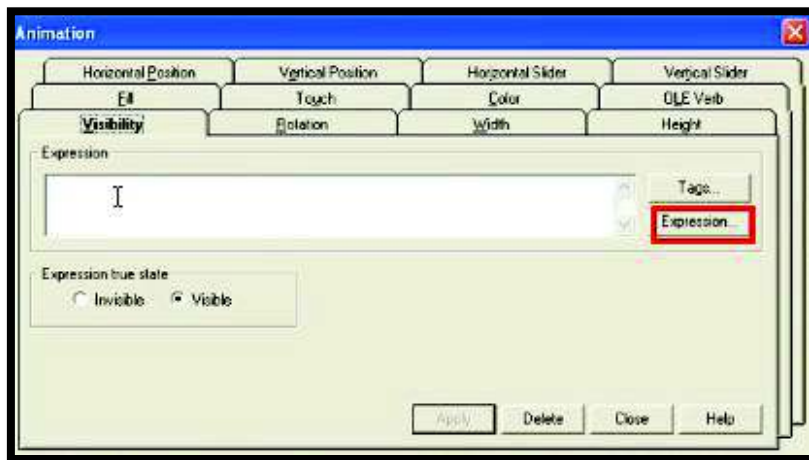
**Figura II.49** Selección de Grupos de Seguridad

Seguidamente se selecciona el nivel de seguridad, por medio de asignación de códigos de seguridad establecidos por letras para tener acceso a las diferentes pantallas del HMI.



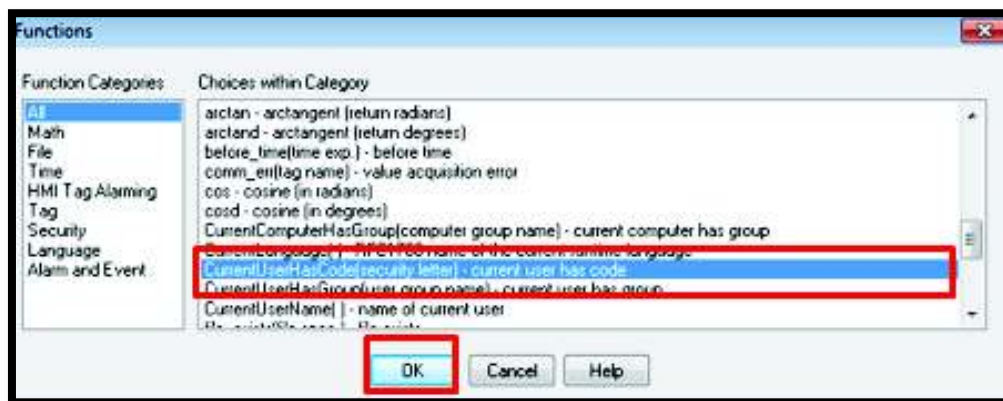
**Figura II.50** Selección de Códigos de Seguridad

El objeto que va a tener seguridad de acuerdo al nivel debe ser configurado dando un clic derecho sobre el mismo en “Animation” seleccionamos “Visibility” y se despliega una ventana en la que seleccionamos “Expression”.



**Figura II.51** Configuración de Seguridad de un Objeto

Se despliega una ventana en la que se selecciona “Function” y se selecciona “Security Letter”



**Figura II.52** Selección de la función de Seguridad por Niveles

Esta función permite especificar, digitando la letra, que nivel va a tener acceso a este objeto.

En la pantalla del HMI se debe establecer dos botones para el ingreso del usuario a la aplicación “Login” y “Logout” y se encuentran en la barra de herramientas, no es necesario realizar ningún tipo de programación adicional pues ya se encuentran preconfigurados.

## INTERFACE DE OPERADOR IMPLEMENTADO

El HMI desarrollado en este proyecto consta de 5 ventanas y describe el proceso de colocación de fertilizante en plantas.

- Bienvenida
- Usuarios

- Proceso
- Visualización de Estados de Sensores
- Visualización del Proceso

A continuación, se describe cada una de las ventadas creadas en el HMI.

### **Ventana de Bienvenida**

Esta ventana se muestra para dar inicio al HMI contiene un botón que permite acceder a la siguiente ventana “Usuarios”.



**Figura II.53** Ventana de Bienvenida HMI

### **Ventana de Usuarios**

En esta ventana se habilita y deshabilita el ingreso de un usuario. Esta ventana mostrará los botones que dan acceso a las diferentes ventanas de acuerdo al usuario.

Se crearon dos usuarios:

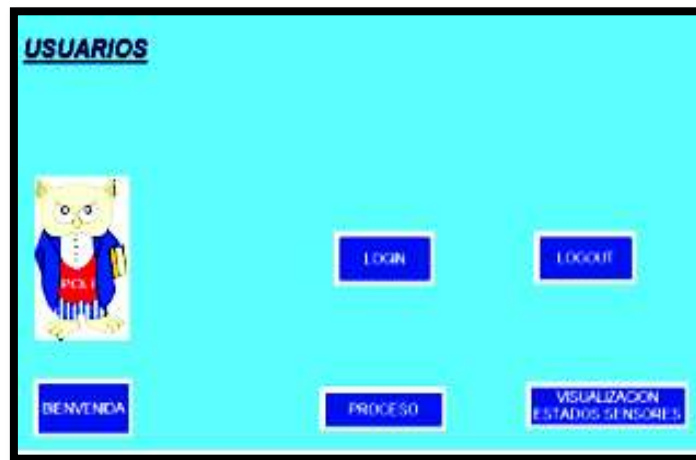
**USUARIO UNO:** JAVIER

**CLAVE:** JAVIER JAVIER

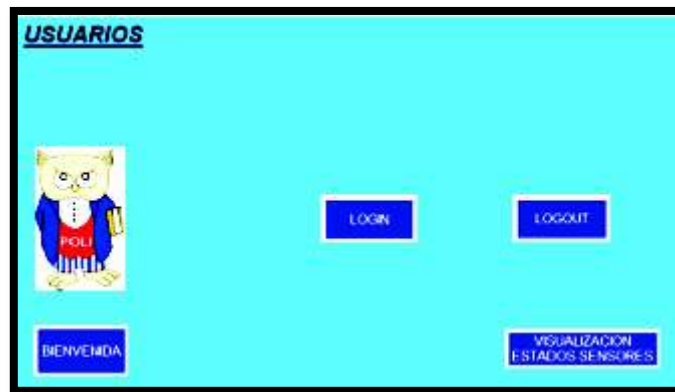
**USUARIO DOS:** BELA

**CLAVE:** BELA BELA

El usuario “JAVIER” tiene acceso a todas las ventanas del HMI y el usuario “BELA” tiene acceso a la ventana visualización de estado de sensores y visualización de proceso.



**Figura II.54** Ventana de Usuario “Javier” HMI



**Figura II.55** Ventana de Usuario “Bela” HMI

### **Ventana de Proceso**

Esta ventana permite realizar el control y monitoreo de los sensores que intervienen en el proceso.

**Sensor de Temperatura:** El usuario puede modificar el set point de temperatura. De este dependerá el accionamiento de un ventilador o de un calefactor. El lazo de histéresis de temperatura es de 4 grados.

**Sensor de Nivel:** Permite advertir al usuario el nivel bajo y alto del tanque. El tanque se llenará de acuerdo a la activación de estas alarmas con la apertura y cierre de una válvula.

**Sensor de Luz:** Su activación inicia el movimiento de la banda transportadora en modo manual o automático.

**Sensor de Humo:** Su activación detiene a todo el proceso.

**Sensor de Presencia:** Su activación abre una válvula para verter sobre la planta el fertilizante.

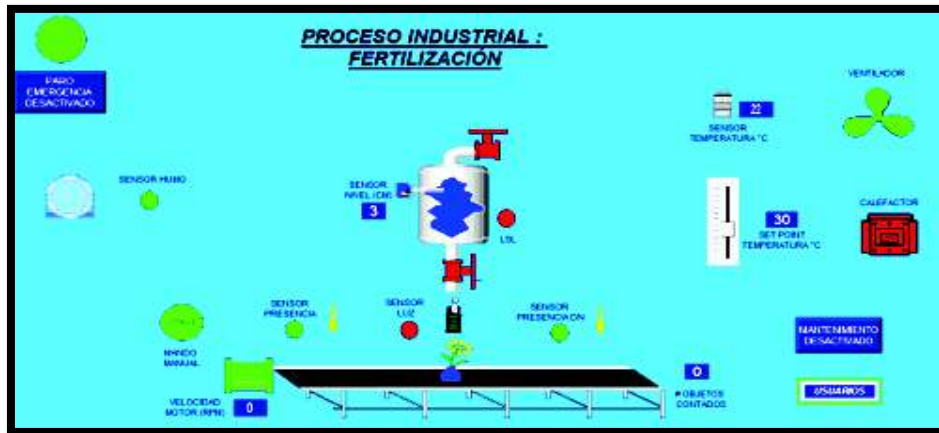


Figura II.56 Ventana de Proceso HMI

### ACCIONES ADICIONALES

**Mantenimiento:** Esta acción se realiza con la activación del sensor de presencia y pulsando la función activar y giro hacia atrás del PanelView.

**Paro de Emergencia:** Esta acción detiene el proceso hasta que se lo vuelva a presionar.

### Ventana de Visualización de Estado de Sensores

Esta ventana fue creada para el usuario "BELA" permitiendo visualizar los estados de los sensores.



Figura II.57 Ventana de Visualización de Estados de Sensores HMI

## Ventana de Visualización del Proceso

Esta ventana fue creada para usuario "BELA" permitiendo visualizar el proceso permitiendo establecer a este usuario el set point de temperatura. No se permite a este usuario realizar paro de emergencia ni la acción de mantenimiento.

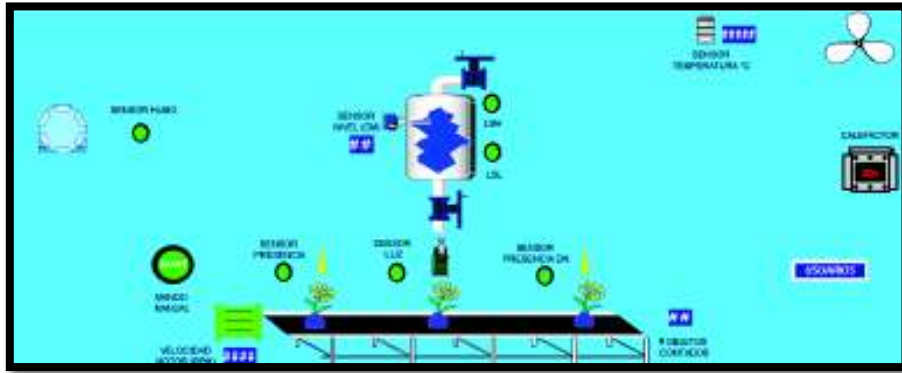


Figura II.58 Ventana de Visualización de Proceso HMI

## Acciones del PanelView

El PanelView permite realizar la configuración para mover la banda transportadora, como es sentido de giro, velocidad, arranque y paro del motor que la comanda. Así como también el usuario podrá indicar la cantidad de plantas que desea tener con fertilizante y seleccionar el modo de operación manual y automático.

Para navegar por las pantallas del PanelView se va seleccionando las teclas de función que va solicitando en pantalla.

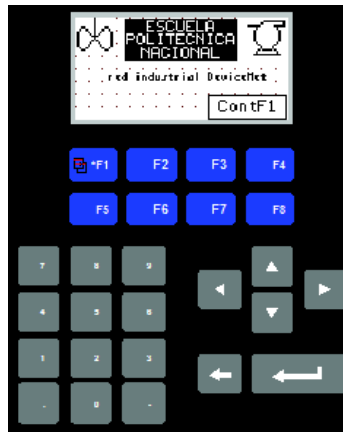
El PanelView consta de 7 pantallas las cuales son Bienvenida, Creadores, Menú Principal, Seguridad, Parámetros, Modo Operación y Funcionamiento.



Figura II.59 Ventanas del PanelView

A continuación, se describe la función de cada apantalla del PanelView.

**Pantalla de Bienvenida:** Es la primera pantalla que se muestra el energizar el PanelView.



**Figura II.60** Pantalla de Bienvenida PanelView

**Pantalla de Creadores:** Esta pantalla muestra a la creadora de la aplicación.



**Figura II.61** Pantalla de Creadores PanelView

**Pantalla de Menú Principal:** En esta pantalla se despliegan las opciones para ir a las siguientes pantallas seguridad, parámetros, modo de operación.



**Figura II.62** Pantalla de Menú Principal PanelView

**Pantalla de Seguridad:** Permite configurar las claves de los usuarios, las mismas que solo pueden ser modificadas por el supervisor.



**Figura II.63** Pantalla de Seguridad PanelView

Para ingresar a la aplicación se debe ingresar la clave de “Supervisor”. Las claves de usuarios “Charles” y “Mantenimiento” permiten modificar al supervisor sus respectivas claves.

Properties Operators Secured Screens Info							
	Name	Password	Config Access	Modify Others	Visible	Changeable	Use Existing
1	Javier	101980	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	Mantenimien	101981	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	Supervisor	101982	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	Operator 4		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	Operator 5		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	Operator 6		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Figura II.64** Claves de Usuarios PanelView

**Pantalla de Parámetros:** Permite configurar el sentido de giro del motor y su velocidad en rpm.



**Figura II.65** Pantalla de Parámetros PanelView



**Pantalla de Modo Operación:** Permite configurar el modo de operación manual o automático.



**Figura II.66** Pantalla de Modo de Operación

**Pantalla de Funcionamiento:** Permite establecer el número de objetos a contar por el sensor fotoeléctrico RightSight DeviceNet PHOTOSWITCH y permite arrancar o detener al motor. Además, se muestra la velocidad y el número de objetos contados por el sensor.




**Figura II.67** Pantalla de Funcionamiento PanelView

## ANEXO III

### HOJAS TÉCNICAS DE ELEMENTOS UTILIZADOS

#### MICROCONTROLADOR CAN AT90CAN32

<p><b>Features</b></p> <ul style="list-style-type: none"><li>• High-performance, Low-power AVR<sup>®</sup> 8-bit Microcontroller</li><li>• Advanced RISC Architecture<ul style="list-style-type: none"><li>– 133 Powerful Instructions – Most Single Clock Cycle Execution</li><li>– 32 x 8 General Purpose Working Registers + Peripheral Control Registers</li><li>– Fully Static Operation</li><li>– Up to 16 MIPS Throughput at 16 MHz</li><li>– On-chip 2-cycle Multiplier</li></ul></li><li>• Non-volatile Program and Data Memories<ul style="list-style-type: none"><li>– 32K/64K/128K Bytes of In-System Reprogrammable Flash (AT90CAN32/64/128)<ul style="list-style-type: none"><li>• Endurance: 10,000 Write/Erase Cycles</li></ul></li><li>– Optional Boot Code Section with Independent Lock Bits<ul style="list-style-type: none"><li>• Selectable Boot Size: 1K Bytes, 2K Bytes, 4K Bytes or 8K Bytes</li><li>• In-System Programming by On-Chip Boot Program (CAN, UART ...)</li><li>• True Read-While-Write Operation</li></ul></li><li>– 1K/2K/4K Bytes EEPROM (Endurance: 100,000 Write/Erase Cycles) (AT90CAN32/64/128)</li><li>– 2K/4K/8K Bytes Internal SRAM (AT90CAN32/64/128)</li><li>– Up to 64K Bytes Optional External Memory Space</li><li>– Programming Lock for Software Security</li></ul></li><li>• JTAG (IEEE Std. 1149.1 Compliant) Interface<ul style="list-style-type: none"><li>– Boundary-scan Capabilities According to the JTAG Standard</li><li>– Programming Flash (Hardware ISP), EEPROM, Lock &amp; Fuse Bits</li><li>– Extensive On-chip Debug Support</li></ul></li><li>• CAN Controller 2.0A &amp; 2.0B - ISO 16845 Certified <sup>(1)</sup><ul style="list-style-type: none"><li>– 15 Full Message Objects with Separate Identifier Tags and Masks</li><li>– Transmit, Receive, Automatic Reply and Frame Buffer Receive Modes</li><li>– 1Mbits/s Maximum Transfer Rate at 8 MHz</li><li>– Time stamping, TTC &amp; Listening Mode (Spying or Autobaud)</li></ul></li><li>• Peripheral Features<ul style="list-style-type: none"><li>– Programmable Watchdog Timer with On-chip Oscillator</li><li>– 8-bit Synchronous Timer/Counter-0<ul style="list-style-type: none"><li>• 10-bit Prescaler</li><li>• External Event Counter</li><li>• Output Compare or 8-bit PWM Output</li></ul></li><li>– 8-bit Asynchronous Timer/Counter-2<ul style="list-style-type: none"><li>• 10-bit Prescaler</li><li>• External Event Counter</li><li>• Output Compare or 8-Bit PWM Output</li><li>• 32KHz Oscillator for RTC Operation</li></ul></li><li>– Dual 16-bit Synchronous Timer/Counter-1 &amp; 3<ul style="list-style-type: none"><li>• 10-bit Prescaler</li><li>• Input Capture with Noise Canceler</li><li>• External Event Counter</li><li>• 3-Output Compare or 16-Bit PWM Output</li><li>• Output Compare Modulation</li></ul></li><li>– 8-channel, 10-bit SAR ADC<ul style="list-style-type: none"><li>• 8 Single-ended Channels</li><li>• 7 Differential Channels</li><li>• 2 Differential Channels With Programmable Gain at 1x, 10x, or 200x</li></ul></li><li>– On-chip Analog Comparator</li><li>– Byte-oriented Two-wire Serial Interface</li><li>– Dual Programmable Serial USART</li><li>– Master/Slave SPI Serial Interface<ul style="list-style-type: none"><li>• Programming Flash (Hardware ISP)</li></ul></li></ul></li><li>• Special Microcontroller Features<ul style="list-style-type: none"><li>– Power-on Reset and Programmable Brown-out Detection</li><li>– Internal Calibrated RC Oscillator</li><li>– 8 External Interrupt Sources</li><li>– 5 Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down &amp; Standby</li><li>– Software Selectable Clock Frequency</li><li>– Global Pull-up Disable</li></ul></li><li>• I/O and Packages<ul style="list-style-type: none"><li>– 53 Programmable I/O Lines</li><li>– 64-lead TQFP and 64-lead QFN</li></ul></li><li>• Operating Voltage: 2.7 - 5.5V</li><li>• Operating Temperature: Automotive (-40 °C to +125 °C)</li><li>• Maximum Frequency: 8 MHz at 2.7V, 16 MHz at 1.5V</li></ul> <p>Note: 1. See details on Section 10.4.3 on page 241.</p>	
	<p>8-bit AVR<sup>®</sup> Microcontroller with 32K/64K/128K Bytes of ISP Flash and CAN Controller</p>
	<p>AT90CAN32 AT90CAN64 AT90CAN128</p>
	<p>Automotive</p>
	<p>Summary</p>

# SENSOR DE PRESIÓN DIFERENCIAL MPX5050DP

## Integrated Silicon Pressure Sensor On-Chip Signal Conditioned, Temperature Compensated and Calibrated

The MPX5050/MPXV5050G series piezoresistive transducer is a state-of-the-art monolithic silicon pressure sensor designed for a wide range of applications, but particularly those employing a microcontroller or microprocessor with A/D inputs. This patented, single element transducer combines advanced micromachining techniques, thin-film metalization, and bipolar processing to provide an accurate, high level analog output signal that is proportional to the applied pressure.

### Features

- 2.5% Maximum Error over 0° to 85°C
- Ideally suited for Microprocessor or Microcontroller-Based Systems
- Temperature Compensated Over -40° to +125°C
- Patented Silicon Shear Stress Strain Gauge
- Durable Epoxy Unibody Element
- Easy-to-Use Chip Carrier Option

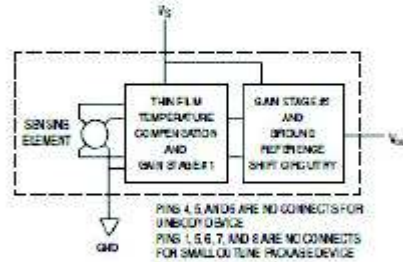
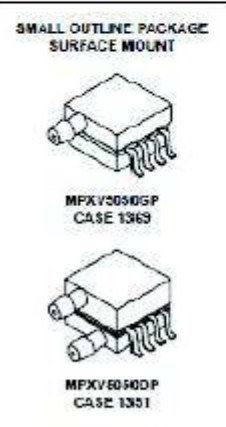


Figure 1. Fully Integrated Pressure Sensor Schematic



PIN NUMBER			
1	N/C	5	N/C
2	V <sub>s</sub>	6	N/C
3	Gnd	7	N/C
4	V <sub>out</sub>	8	N/C

NOTE: Pins 1, 5, 6, 7, and 8 are internal device connections. Do not connect to external circuitry or

## MPX5050 MPXV5050G SERIES

Monaco's Preferred Device

**INTEGRATED  
PRESSURE SENSOR**  
0 to 50 kPa (0 to 7.25 psi)  
0.2 to 4.7 Volts Output

### UNIBODY PACKAGE



MPX5050D  
CASE 867



MPX5050GP  
CASE 867B




MPX5050DP  
CASE 867C

PIN NUMBER			
1	V <sub>s</sub>	4	N/C
2	Gnd	5	N/C
3	V <sub>s</sub>	6	N/C

NOTE: Pins 4, 5, and 6 are internal device connections. Do not connect

## SENSOR DE MOVIMIENTO SWAN SQUAD

### Sensor de Movimiento SWAN SQUAD



Compensación de temperatura. Diseño compacto apto para aplicaciones domiciliarias.

Contador de pulsos variable 1-2 y 3. Ajuste de sensibilidad. Inmunidad ambiental. Altura de montaje entre 1.8 -2.4 mts. (Sin necesidad de ajuste). Led de función.

Método de detección: Quad (pir cuatro elementos)- Alimentación: 8.2 a 16V.  
Consumo; reposo: 14Ma, disparo: 22Ma con led, Disparo: 19 Ma sin led.  
Selección de antimascota por jumper: 15Kg.-25 Kg. Tiempo de alarma: 2 seg.  
Salida de alarma: NC. Llave tamper: NC. Led indicador: encendido de alarma.  
Temperatura de trabajo: -20C° a 60C°. Rechazo a RF: 30V/mt. Medidas: 92mm x 59mm x 37mm. Peso: 40 gr.

Modo de detección Quad (4 elementos).

## SENSOR DE HUMO SYSTEM SENSOR




### Serie 100 Detectores de humo fotoeléctricos

Especificaciones Generales	
<b>Voltaje de servicio</b> Según base	<b>Construcción</b> Termoplástico retardante de llama
<b>Corriente en estado pasivo</b> 85 µAmp	<b>Temperatura</b> Mod. 2151: 0°C a 49°C (32°F a 120°F) / Mod. 2151T: 0°C a 38°C (32°F a 100°F)
<b>Sensibilidad</b> 3% ± 7%/ft. fotoeléctrico	<b>Rango de velocidad de aire, listado UL</b> Fotoeléctrico: 0 - 15.2 m/s ( 0 - 3000 fpm )
<b>Peso</b> 102 gr (3.6 oz)	<b>Rango de humedad</b> 10% - 93% sin condensación
<b>Dimensiones</b> Altura: 42mm (1.66") Diámetro sin base: 104 mm (4.1") Diámetro con base: 155 mm (6.1")	<b>Espacio de detección</b> Sobre techos lisos (según NFPA 72) el espaciado cada 9,14 metros (30 pies) se puede tomar como guía. Para otro espaciado entre detectores se debe considerar la altura y tipo de techo, movimiento del aire y otros requerimientos de respuesta.

**Selección de Tipo de Bases y adaptador.**  
 Nota: Las bases de la serie 400 de 6 pulgadas, modelos B401B, B401BR, B402B, B404B y B406B se pueden utilizar con detectores de la serie 100 en aplicaciones de remodelación, mediante el anillo adaptador F110

Modelo de Base	Tipo de Circuito	Resistencia Limitadora de Corriente	Voltaje Nominal	Tipo de Contacto	Consumo de Corriente en Alarma
B110LP/B401B	2 hilos*	No	-	12 / 24 VCD	10 - 130**
B110RLP/B401R*/B401BR	2 hilos*	Si	-	24 VCD	10 - 62
B112LP/B402B	4 hilos	Si	Forma A y C	24 VCD	17 - 36
B114LP/B404B	4 hilos	Si	Forma A y C + A supervisor	120 VCA	75 mA (CA) max.
B116LP/B406B	2 hilos*	No	Forma C	24 VCD	20 - 100**
B401†	2 hilos*	No	-	12 / 24 VCD	10 - 130**

\*Funcionalidad compatible solo con paneles listados    \*\*Limitada por el panel de control    †Base tipo europea del mismo diámetro que el detector

Tipo de contactos del relé: Resistivo o Inductivo (60% factor de potencia) carga

Forma A: 2.0 a 30VCA/CD

Forma C: 0.6A a 110VCD, 2.0A a 30VCD

1.0A a 125VCA, 2.0A a 30VCA

## SENSOR DE TEMPERATURA PT100 SIEMENS

### SONDA TEMPERATURA PT 100, PT 1000



El rango de utilización de las sondas de temperatura depende del tipo de construcción que se fabrique y del tipo de sensor de temperatura que se utilice. Como información general se puede considerar que una sonda de temperatura pt100 con salida directa a cable puede trabajar entre -50°C y 250°C en función del tipo de cable y de las medidas de la vaina de protección. En el caso de las sondas de temperatura con cabezal de conexiones el rango más habitual abarca desde -50°C a los 500°C. Es conveniente indicar siempre la temperatura de trabajo a efectos de elegir la construcción y los materiales más adecuados. Como orientación, a continuación se muestran las construcciones más habituales, aunque en muchos casos estas construcciones vienen detalladas por los propios planos y especificaciones del cliente.

## PLCS MICROLOGIX CON MÓDULO ESCANER DEVICENET 1769-SDN

### 1769-SDN Scanner Module Data Tables

The scanner module uses input and output data images to transfer data, status, and command information between the scanner module and the MicroLogix controller to manage the flow of data between your controller and network devices.

#### Input Data Image - MicroLogix 1500

The input data image is transferred from the scanner module to the controller across the Compact I/O bus.

Word	Description	Data Type
0...65	Status structure	66-word array
66...245	DeviceNet slave inputs	180-word array

See [Chapter 6](#) for definitions of the Status structure.


#### Output Data Image - MicroLogix 1500

The output data image is transferred from the controller to the scanner module across the Compact I/O bus.

Word	Description	Data Type
0 and 1	Module command array	2-word array
2...181	DeviceNet slave outputs	180-word array

## POWERFLEX700 DEVICENET

### PowerFlex 700 AC Drives



#### Features

- Includes power ratings:
  - 200...240V: 0.37...66 kW / 0.5...100 Hp / 2.2...260 A
  - 380...480V: 0.37...132 kW / 0.5...200 Hp / 1.1...260 A
  - 500...600V: 1...150 Hp / 1.7...144 A
  - 690V: 45...132 kW / 52...142 A
- Includes IP20 NEMA/UL Type 1, Open/Flange, IP54 NEMA 12.
- Provides V/Hz control, sensorless vector control, and vector control with FORCE™ Technology
- Offers CIP™ (Common Industrial Protocol) communications including DeviceNet™, ControlNet™, and EtherNet/IP™
- Meets demands for space, flexibility, and reliability
- Provides excellent torque production, tight torque regulation, and fast update times of torque inputs
- Offers a control cassette which houses all control, I/O, and encoder options and is interchangeable with all PowerFlex® 700 drive ratings.
- Offers configuration and programming via optional integral LCD keypad, remote keypad, or software
- Offers Premier Integration with PowerFlex® AC drives which uses the Logix control platform to simplify parameter and tag programming and reduce development time
- Our industry expertise in [Drives Systems Solutions](#) can help you maximize machine availability and minimize project risk.

## OPTOACOPLADOR 4N25

<b>FAIRCHILD</b> SEMICONDUCTOR®		<b>GENERAL PURPOSE 6-PIN PHOTOTRANSISTOR OPTOCOUPLEDERS</b>					
4N25 4N37	4N26 H11A1	4N27 H11A2	4N28 H11A3	4N35 H11A4	4N36 H11A5		
ELECTRICAL CHARACTERISTICS (T <sub>A</sub> = 25°C unless otherwise specified)							
<b>INDIVIDUAL COMPONENT CHARACTERISTICS</b>							
Parameter	Test Conditions	Symbol	Min	Typ*	Max	Unit	
<b>EMITTER</b>							
Input Forward Voltage	(I <sub>F</sub> = 10 mA)	V <sub>F</sub>		1.18	1.25	V	
Reverse Leakage Current	(V <sub>R</sub> = 5.0 V)	I <sub>R</sub>		0.081	10	µA	
<b>DETECTOR</b>							
Collector-Emitter Breakdown Voltage	(I <sub>C</sub> = 10 mA, I <sub>E</sub> = 0)	BV <sub>CE</sub>	30	100		V	
Collector-Base Breakdown Voltage	(I <sub>C</sub> = 100 µA, I <sub>E</sub> = 0)	BV <sub>CB</sub>	70	120		V	
Emitter-Collector Breakdown Voltage	(I <sub>E</sub> = 100 µA, I <sub>C</sub> = 0)	BV <sub>EC</sub>	7	10		V	
Collector-Emitter Dark Current	(V <sub>CE</sub> = 10 V, I <sub>E</sub> = 0)	I <sub>CE0</sub>		1	50	nA	
Collector-Base Dark Current	(V <sub>CB</sub> = 10 V)	I <sub>CB0</sub>			20	nA	
Capacitance	(V <sub>CE</sub> = 5 V, f = 1.0 kHz)	C <sub>in</sub>		8	15	pF	

## AMPLIFICADOR OPERACIONAL LM324

### LM324, LM324A, LM324E, LM224, LM2902, LM2902E, LM2902V, NCV2902

**MAXIMUM RATINGS** ( $T_A = +25^\circ\text{C}$ , unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltages Single Supply Split Supplies	$V_{CC}$ $V_{CC}, V_{EE}$	32 $\pm 16$	Vdc
Input Differential Voltage Range (Note 1)	$V_{IDR}$	$\pm 32$	Vdc
Input Common Mode Voltage Range	$V_{ICR}$	-0.3 to 32	Vdc
Output Short Circuit Duration	$t_{SC}$	Continuous	
Junction Temperature	$T_J$	150	$^\circ\text{C}$
Thermal Resistance, Junction-to-Air (Note 2)	Case 646 Case 751A Case 948G	$R_{\theta JA}$ 118 156 190	$^\circ\text{C/W}$
Storage Temperature Range	$T_{stg}$	-65 to +150	$^\circ\text{C}$
Operating Ambient Temperature Range	$T_A$	LM224 LM324, LM324A, LM324E LM2902, LM2902E LM2902V, NCV2902 (Note 3) -25 to +85 0 to +70 -40 to +105 -40 to +125	$^\circ\text{C}$

Stresses exceeding those listed in the Maximum Ratings table may damage the device. If any of these limits are exceeded, device functionality should not be assumed, damage may occur and reliability may be affected.

1. Split Power Supplies.
2. All  $R_{\theta JA}$  measurements made on evaluation board with 1 oz. copper traces of minimum pad size. All device outputs were active.
3. NCV2902 is qualified for automotive use.

#### ESD RATINGS

Rating	HBM	MM	Unit
ESD Protection at any Pin (Human Body Model – HBM, Machine Model – MM)			
NCV2902 (Note 3)	2000	200	V
LM324E, LM2902E	2000	200	V
LM324DG/DR2G, LM2902DG/DR2G	200	100	V
All Other Devices	2000	200	V

## 7. ORDEN DE EMPASTADO