UNIVERSIDAD PRIVADA DE TACNA

FACULTAD DE INGENIERIA

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA

TESIS:

“DISEÑO E IMPLEMENTACIÓN DE UN MÓDULO CON VARIADOR DE FRECUENCIA, PARA EL CONTROL ÓPTIMO DEL ARRANQUE Y LA VELOCIDAD DE UN MOTOR TRIFÁSICO EN UNA RED INDUSTRIAL DEVICENET PARA EL LABORATORIO DE CONTROL Y AUTOMATIZACIÓN”

PARA OPTAR:

EL TÍTULO PREFESIONAL DE INGENIERO ELECTRÓNICO

PRESENTADO POR:

BACH: Carlos Adrian ZAPANA VARGAS

BACH: Marlyn Yanira CAMPOS RIVEROS

Tacna – Perú

2018

UNIVERSIDAD PRIVADA DE TACNA

FACULTAD DE INGENIERÍA

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA

"DISEÑO E IMPLEMENTACIÓN DE UN MÓDULO CON VARIADOR DE FRECUENCIA, PARA EL CONTROL ÓPTIMO DEL ARRANQUE Y LA VELOCIDAD DE UN MOTOR TRIFÁSICO EN UNA RED INDUSTRIAL DEVICENET PARA EL LABORATORIO DE CONTROL Y AUTOMATIZACIÓN"

Tesis Sustentada y Aprobada el 14 de Diciembre del 2018; estando el Jurado Calificador Integrado por:

PRESIDENTE: ______________________

ING. ALFREDO ESTEBAN CALIZAYA CRUZ

SECRETARIO: ______________________

ING. CARLOS ARMANDO RODRIGUEZ SILVA

VOCAL: ______________________

ING. MARIA ELENA VILDOZO ZAMBRANO

ASESOR: ______________________

MAG. ANIBAL JUAN ESPINOZA ARANCIAGA

DECLARACION JURADA DE ORIGINALIDAD

Yo CARLOS ADRIAN ZAPANA VARGAS y MARLYN YANIRA CAMPOS RIVEROS en condición de egresados de la carrera Professional de Ingeniería Electrónica de La Universidad Privada de Tacna con grado académico de Bachiller, identificado con DNI: 43982338 y 45674879 respectivamente.

Declaro bajo juramento que:

1. Somos autores de la Tesis Titulada: "DISEÑO E IMPLEMENTACIÓN DE UN MÓDULO CON VARIADOR DE FRECUENCIA, PARA EL CONTROL ÓPTIMO DEL ARRANQUE Y LA VELOCIDAD DE UN MOTOR TRIFÁSICO EN UNA RED INDUSTRIAL DEVICENET PARA EL LABORATORIO DE CONTROL Y AUTOMATIZACIÓN".
2. La presente investigación de Tesis es original siendo resultados de nuestros esfuerzos y dedicación, la tesis que se da a conocer no ha sido copiada, para ello se ha respetado las normas intencionales de citas y referencias para las fuentes consultadas.
3. Declaro que es una Tesis Única, y no se ha presentado con anterioridad para lograr algún grado académico.
4. Soy consciente de que el hecho de no respetar los derechos del autor y hacer plagio, se expone a sanciones Universitarias y/o Legales, por lo que asumo toda responsabilidad y sanción.

De Identificarse falsificaciones, fraude, piratería, plagio o que la Tesis tenga antecedentes de las mismas características; asumo las responsabilidades y sanciones que se ameriten, responsabilizándome por todas las cargas legales que se deriven de ello sometiéndome a las normas establecidas y vigentes de la Universidad Privada De Tacna.

Tacna, Diciembre del 2018

Bach. Carlos Adrian Zapana Vargas

DNI 43982338

Bach. Marlyn Yanira Campos Riveros

DNI 45674879

DEDICATORIA

Dedico esta Tesis a mi Padre Walter Zapana por su sacrificio y voluntad para obtener una formación profesional, a mi madre Adriana Vargas por su constante guía, sabiduría y compresión en mi etapa de estudiante y a mi hermana Betty por su incondicional y desinteresado apoyo para la culminación de esta TESIS. Gracias.

Carlos Adrian ZAPANA VARGAS

DEDICATORIA

Papa Daniel y Mamá Yolita

Eres una mujer que simplemente me hace llenar de orgullo, te amo y no va haber manera de devolverte lo mucho que me has ofrecido desde que incluso no hubiera nacido. Esta tesis es un logro más que llevo a cabo, y sin lugar a dudas ha sido gracias a ti; no sé dónde me encontraría de no ser por tu constante ayuda, tu compañía y tu amor.

Mamá

Su afecto y cariño son los detonantes de mi felicidad, de mi esfuerzo, de mis ganas de buscar lo mejor para ustedes. Aun a su corta edad, me han enseñado y me siguen enseñando muchas cosas de esta vida.

Tahiel y Ethan

Fueron mi motivación más grande para concluir con éxito este proyecto de tesis.

Marlyn Yanira CAMPOS RIVEROS.

AGRADECIMIENTOS

La vida se encuentra llena de retos, y uno de ellos es la universidad, tras vernos dentro de ella, nos hemos dado cuenta que más allá de ser un reto, es una base no solo para nuestro entendimiento del campo en el que nos hemos visto inmersos, sino para lo que concierne a la vida y nuestro futuro.

Agradezco a mi institución y a mis maestros por sus esfuerzos y orientación para que finalmente pudiéramos graduarnos como felices profesionales.

RESUMEN

Este Proyecto desarrolla con más detalle un Control Remoto, Optimo Arranque y Velocidad de un Motor Asíncrono Trifásico. Estableciendo la Comunicación de un PLC Compact Logix 1769 – L23E y un Dispositivo Inteligente Variador POWERFLEX 70, por medio del protocolo de Comunicación DEVICENET. que muestran ser robustos en muchas aplicaciones para Dispositivos de Campo y son los más utilizados en la industria.

El proyecto comienza con el armado del módulo de Variador de Frecuencia, el cual nos permite tener un mejor acceso a las I/O y puerto de comunicación de nuestro Variador Powerflex 70, también nos permite tener una comunicación directa con nuestro equipo escáner DEVICENET 1769-SDN que va añadido al PLC, se adhiere el armado de un módulo de motor Trifásico de ½ HP.

Una vez teniendo los módulos tanto de nuestro variador como nuestro módulo del motor, se procede a enlazarlos mediante el módulo de Escáner para Redes DEVICENET, por medio de los programas de comunicación y programación propios del fabricante, RS LOGIX.

Habiendo realizado los cálculos para un correcto escalamiento se obtiene un óptimo funcionamiento de arranque y velocidad en nuestro motor Trifásico.

Finalmente podemos probar que la Automatización de equipos Industriales, Dispositivos Inteligentes, etc. Mediante Protocolos de Comunicación Industriales optimizan los dispositivos de campo y ayudan a la mejora de procesos.

Palabras Claves: DEVICENET, Redes Industriales, Protocolo de Comunicación.

ABSTRACT

This Project develops in more detail a Remote Control, optimal Starting and Speed of a Three-Phase Asynchronous Motor. Establishing the Communication of a Compact Logix PLC 1769 – L23E and a Smart Device POWERFLEX Variable 70, by means of the Communication Protocol DEVICENET. Which show to be robust in many applications for Field Devices and are the most used in the Industry.

The project Begins with the assembly of the Frequency Converter Module, which allows us to have better access to the I / O and communications port of our Powerflex 70 Inverter, it also allows us to have direct communications with our Scanner Equipment DEVICENET 1769 – SDN that is added to the PLC, adheres to the assembly of a ½ HP Three-Phase Motor Module.

Finally, we can prove that the Automation of Industrial equipment, smart devices, etc., through the Industrial Communication Protocol DEVICENET, optimizes field devices and helps to improve processes.

Keywords: DEVICENET, Industrial Networks, Communication Protocol.

INDICE DE CONTENIDOS

INDICE DE FIGURAS

:

INTRODDUCCIÓN

En la actualidad, debido al crecimiento mundial de la automatización de los procesos industriales, los controladores lógicos programables (PLCs) son ampliamente utilizados para la supervisión, control y automatización de dichos procesos.

La red de comunicación industrial DeviceNet es una red abierta e implementada según el modelo OSI. Las características que posee este estándar facilita el desarrollo de múltiples aplicaciones en diferentes procesos industriales.

En la presente tesis denominada**: "DISEÑO E IMPLEMENTACIÓN DE UN MÓDULO CON VARIADOR DE FRECUENCIA, PARA EL CONTROL ÓPTIMO DEL ARRANQUE Y LA VELOCIDAD DE UN MOTOR TRIFÁSICO EN UNA RED INDUSTRIAL DEVICENET PARA EL LABORATORIO DE CONTROL Y AUTOMATIZACIÓN"**, se propone realizar el diseño y la implementación de dos módulos de variador de frecuencia a través de un sistema de control abierto basado en el estándar DEVICENET, que controle el arranque y velocidad de un motor eléctrico asíncrono trifásico utilizando el PLC Allen Bradley Compact Logic 1769-L23, y un variador de frecuencia POWERFLEX 70 de Allen Bradley. Dichos módulos serán implementados para ser utilizados por los estudiantes de ingeniería electrónica; dichos módulos serán instalados en el Laboratorio de Automatización y control de la Universidad Privada de Tacna.

La presente tesis ha sido estructurada siguiendo los lineamientos que se encuentran en el reglamento de grados y títulos de la universidad privada de Tacna; ha sido desarrollado en cuatro capítulos: El Capítulo I denominado Planteamiento del Problema, desarrolla la parte de metodología de la investigación. Presenta además el marco teórico y los antecedentes del Proyecto a desarrollar. El Capítulo II denominado Marco Metodológico, presenta la metodología de la investigación seguida, el procedimiento seguido para el desarrollo de la Tesis como Proyecto así como el diseño y la implementación de los módulos de entrenamiento. El Capítulo III denominado Análisis de los resultados, se evalúan los resultados obtenidos y se verifican la operatividad de los dispositivos de la red industrial, así como la operatividad de los módulos de entrenamiento. Finalmente se presenta las conclusiones y las recomendaciones. Se termina con las referencias bibliográficas y los anexos del proyecto. Finalmente se muestra la Bibliografía utilizada y los anexos.

CAPÍTULO I
PLANTEAMIENTO DEL PROBLEMA

1.1 DESCRIPCION DEL PROBLEMA

El laboratorio de Automatización y Control de la Escuela Profesional de Ingeniería Electrónica cuenta con dispositivos denominados Controladores Lógicos Programables (PLCs), utilizados para el desarrollo de sus prácticas de laboratorio en las asignaturas que comprende el Área de Automatización y Control, una de las áreas de formación académica que brinda la carrera y que se encuentran en su Plan de Estudios vigente.

Por otro lado, el empleo de los sistemas de control distribuidos denominados redes de comunicaciones industriales han tomado mucho auge en la actualidad, con el propósito de optimizar los procesos industriales que se desarrollan en los centros de producción.

Cabe indicar que la zona sur del Perú, donde está incluida Tacna, es una zona minera por excelencia y donde se han establecido grandes Centro Mineros que producen cobre, plata y otros metales, cuyos procesos de refinación son complejos y requieren de tecnología de vanguardia para optimizar sus procesos.

El Laboratorio de Automatización y Control de la Escuela Profesional de Ingeniería Electrónica de la Universidad Privada de Tacna, cuenta con dispositivos de control, llámese sensores, actuadores, así como de estaciones de trabajo, controladores PLCs de diferentes fabricantes, sin embargo no cuentan con una red comunicación industrial basado DEVICENET.

El implementar nuevas tecnologías que faciliten el desarrollo de aplicaciones basado en el estándar DEVICENET dentro del laboratorio Automatización y Control de la Escuela Profesional de Ingeniería Electrónica, permitirá experimentar a nuestros estudiantes sobre el uso de redes industriales y a la vez desarrollar aplicaciones para el control distribuido de dichos procesos industriales.

Es de imperiosa necesidad que los estudiantes puedan desarrollar sus prácticas de laboratorio en ambientes de redes industriales, con la finalidad de complementar su formación académica teórica a través de redes de comunicación DEVICENET.

1.2 DEFINICION DEL PROBLEMA

El laboratorio de Automatización y Control de la Escuela Profesional de Ingeniería Electrónica de la Universidad Privada de Tacna, no cuenta con módulos de entrenamiento que empleen una red de comunicación industrial para controlar el arranque y la velocidad de un motor trifásico mediante variadores de frecuencia asistido por un Controlador Lógico Programable (PLC).

1.3 FORMULACION DEL PROBLEMA

El problema puede ser formulado con la siguiente pregunta de investigación:

¿Cómo el diseño y la implementación de un módulo de Variador de Frecuencia ajustable con comunicación DEVICENET, permite mejorar el control de arranque y velocidad de trabajo de motor trifásico en los procesos de producción en una empresa?

1.4 JUSTIFICACION DE LA INVESTIGACION

Debido al uso intensivo de los autómatas programables (PLC) en los procesos de producción, ello ha producido un cambio en la industria de la fabricación y las instalaciones industriales que forman parte del proceso. La automatización de los procesos industriales y el control a distancia o distribuido, pasan hoy en día por el desarrollo de las redes de comunicación industrial. Dichas redes de comunicación se han vuelto indispensables en un entorno de trabajo globalizado, donde el control y la supervisión de los procesos industriales de un Centro de Producción pueden ser monitoreados localmente o de moto remoto. DEVICENET es una de las redes de comunicaciones industriales más populares, porque se trata de una red de tecnología abierta basada en el modelo OSI.

Las redes DEVICENET permiten obtener un mejor control de elementos de que la conforman, tales como los dispositivos controladores de motor (variador de velocidad o variador de frecuencia, Estos dispositivos variadores de velocidad y de frecuencia, han ido evolucionado con el tiempo y se han convertido en equipos cada vez más robustos, con una mayor facilidad de programación, con un mejor método de control del motor eléctrico trifásico, y lo más importante, se han convertido en una herramienta que comunica al sistema de control todo lo ocurre en el motor. Con una comunicación DEVICENET, nos permite que el variador de frecuencia envíe toda la información recabada de sus mediciones como: voltaje, corriente, consumo energético, alarmas y fallas, entre otros, al sistema de control de la planta o del proceso, facilitando al operario del sistema, la operación en tiempo real.

1.5 OBJETIVOS

1.5.1 Objetivos generales

Diseñar e implementar un módulo de variador de frecuencia con comunicación DEVICENET, que permita automatizar procesos de producción y el estudio de la tecnología DEVICENET.

1.5.2 Objetivos específicos

- Desarrollar una lógica de control basado en el control de un variador de frecuencia para mejorar el conocimiento sobre una red con comunicación DEVICENET, existente en los diferentes tipos de industrias que se encuentran en nuestro país.
- Realizar esta aplicación con el variador POWERFLEX 70, comunicarlo mediante el módulo scanner DEVICENET modelo 1769 para Compact Logix, para poder realizar una óptima comunicación y control, obteniendo resultados exactos de cómo se comporta nuestro proceso y a la vez supervisarlo por medio de una interface (SCADA).

1.6 ANTECEDENTES DEL ESTUDIO

Nuestra región tiene como un eje principal para su desarrollo la explotación minera y procesamiento industrial de materias primas que allí se extraen.

Dichos centros mineros requieren un alto grado de automatización de sus plantas. En este sentido es de importancia conocer las posibilidades que el control automatizado ofrece para la industria, considerando este aspecto una necesidad inherente al desarrollo profesional, de los estudiantes de la escuela profesional de ingeniería electrónica, en la especialidad de control y automatización de procesos industriales.

A continuación se citará los trabajos desarrollados por Tesistas de algunas instituciones de educación superior latinoamericanas, que se han tomado en cuenta para desarrollar nuestro trabajo de tesis

1.6.1 Universidad Nacional San Agustín, Arequipa.

CANAZA Ch. (2017) presentó en Perú el trabajo de tesis para obtener el título de Ingeniero Electrónico:

> ***"Diseño e implementación de una ampliación de red industrial DEVICENET para el mejoramiento en la supervisión de los interruptores de parada de emergencia con cuerdas de seguridad (pullcords) en un sistema de fajas transportadoras para horno industrial de empresas del sector minero, año 2015"***

El trabajo de tesis presentado por **CANAZA Ch.** (2017), abordó como objetivo general:

> *"Diseñar e implementar una ampliación de red industrial DeviceNet que permita el mejoramiento de la supervisión del sistema de seguridad en un sistema de fajas transportadoras para empresas del sector minero."* ***(p. 7)***

1.6.2 Instituto Politécnico Nacional, México.

GARCÍA, J. (2013) presentó el trabajo de tesis para obtener el título de Ingeniero en Control y Automatización:

> **"IMPLEMENTAR UNA RED DEVICENET PARA LA AUTOMATIZACIÓN DE UN SISTEMA DE BANDAS TRANSPORTADORAS"**

El trabajo de tesis presentado por **GARCÍA, J. (2013)**, se resume en que:

> *"Estudia, diseña y realiza la puesta en marcha de una red DeviceNet para la automatización de un par de bandas transportadoras con el fin de tener un prototipo experimental."* ***(p.6)***

1.6.3 Universidad Austral, Chile.

MUÑOZ, J. (2007) presentó el trabajo de tesis para obtener el título de Ingeniero en Electrónica:

> **"ESTUDIO DE APLICACIÓN DE LOS ESTÁNDARES DEVICENET Y CONTROLNET DE COMUNICACIONES INDUSTRIALES COMO SOLUCIÓN DE RED DE CAMPO Y PROCESO EN UNA PLANTA INDUSTRIAL"**

El trabajo de tesis presentado por **MUÑOZ, J. (2007)**, se resume en que esta tesis:

> *"Estudia de manera profunda las redes de comunicación para aplicaciones industriales DeviceNet y ControlNet".* ***(p. 7)***

1.7 BASES TEORICAS

1.7.1 Automatización

La automatización se entiende como el acto de automatizar. Se define como la aplicación de dispositivos, equipos o procedimientos automáticos en el control de procesos de producción industrial, donde se efectúa la trasferencia de las tareas de producción, sin la participación del ser humano a un conjunto de elementos tecnológicos.

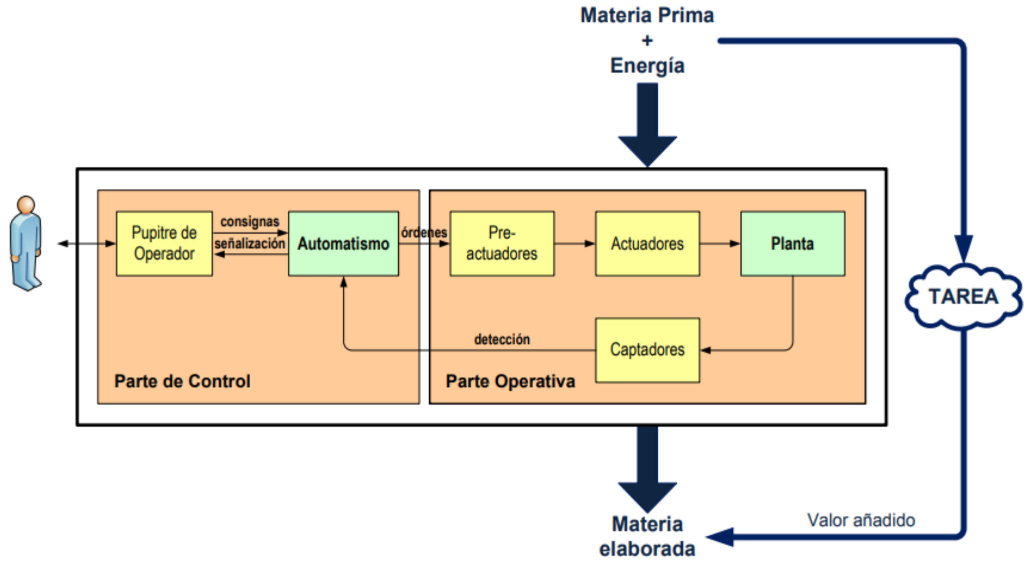


Figura 1: Estructura de un sistema automatizado

(Fuente: En "Tema 7: Introducción a la Automatización Industrial", por GARCIA, 2013, del sitio web: http://mapir.isa.uma.es/varevalo/teaching/automatica/pdfs/Tema%2007%20-%20Introducci%C3%B3n%20a%20la%20Automatizaci%C3%B3n%20Industrial%20v7_vicente.pdf

1.7.1.1 Objetivos de la automatización

Los objetivos principales que persigue la automatización aplicada en un proceso industrial, se ilustra a continuación:

- Optimizar la productividad de la empresa. Con la automatización se logra reducir los costos de producción y mejorar su calidad.
- Mejorar las condiciones laborales de seguridad y protección de los operadores en un centro de producción. Con la automatización se suprime los trabajos pesados y peligrosos, incrementando los niveles de seguridad.
- Simplificar la operatividad del proceso productivo. Con la automatización el recurso humano (operador del sistema) no requiere grandes volúmenes de información para la manipulación del proceso automatizado.
- Integrar los procesos de manufactura y producción. Con la automatización ambos procesos se encuentran en línea.
-

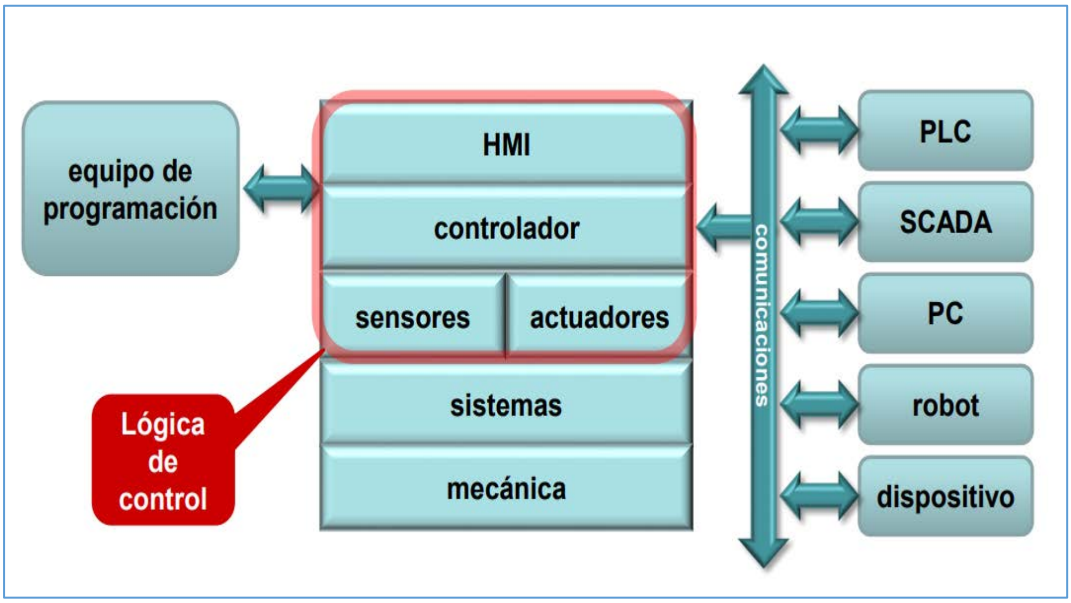


Figura 2: implantación del automatismo en un sistema de producción

(Fuente: En “Tema 7: Introducción a la Automatización Industrial”, por GARCIA, 2013, del sitio web: http://mapir.isa.uma.es/varevalo/teaching/automatica/pdfs/Tema%2007%20-%20Introducci%C3%B3n%20a%20la%20Automatizaci%C3%B3n%20Industrial%20v7_vicente.pdf

- Suministrar la Información en tiempo real del proceso. Con la automatización se puede supervisar los procesos de modo local o remoto “en línea”
- Facilitar la detección de averías y su reparación. Con la automatización, se utilizan dispositivos HMI (acrónimo de Human Machine Interface) que son empleados por los operadores para detectar fallas en el sistema.

1.7.1.2 Sistema automatizado

Un sistema de control donde se aplica la automatización se le denomina sistema automatizado y está compuesto de dos partes principales:

- Parte de mando
- Parte operativa

PARTE DE MANDO

La parte de mando es la componente inteligente de un sistema automatizado, que se encarga de establecer la comunicación entre los diferentes componentes del sistema a través del envío de instrucciones en determinadas direcciones pre-establecidas.

Dependiendo del nivel de automatización, la parte de mando puede establecer: **una etapa de supervisión y explotación** y **una etapa de control.** En la etapa de supervisión se emplean paneles de mando o una PC + SCADA. En la etapa de control se emplean autómatas programables, un PC + tarjeta de E/S, tarjetas de control electrónicas PICs, conocida como tecnología programada. Aunque aún se utilizan relés electromagnéticos, o módulos lógicos neumáticos (tecnología cableada).

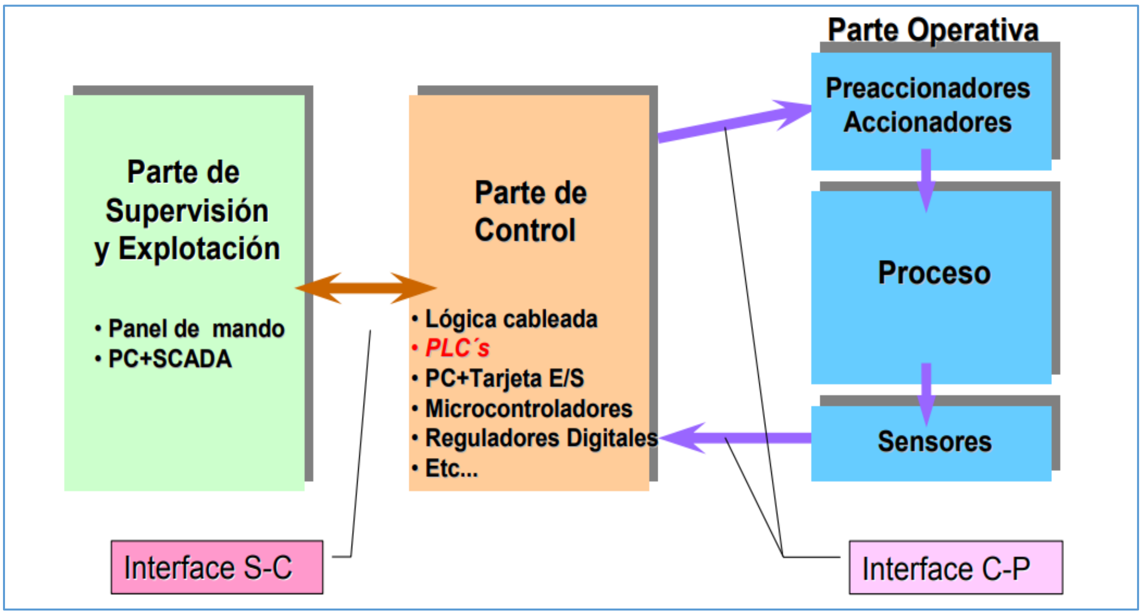


Figura 3: Componentes de un sistema automatizado

(Fuente: En "Sistema automatizado (PLCs)", por Felipe MATEOS, 2001, del sitio web: http://isa.uniovi.es/docencia/iea/teoria/plc.pdf

<u>PARTE OPERATIVA</u>

La parte operativa es la componente que actúa directamente sobre los componentes del sistema. Son los elementos de la parte operativa que hacen que el sistema realice la operación deseada en el proceso industrial. Los elementos que forman la parte operativa son los accionadores de las máquinas como motores, cilindros, compresores y los captadores como fotodiodos, transductores.

1.7.1.3 Tecnologías programadas

Los avances tecnológicos de los últimos años en el Área de la microelectrónica, como la integración a gran escala. Han permitido que los microprocesadores sean usados en la realización de automatismos, favoreciendo el desarrollo de las tecnologías programadas. Los equipos que se emplean en esta tecnología son:

- Los ordenadores personales o computadoras.
- Los autómatas programables.

La PC o computadora, como parte de mando de un sistema automatizado es utilizada en ambientes industriales por ser altamente flexible a modificaciones de proceso. Sin embargo, debido a su diseño orientado al usuario, resulta un elemento frágil para trabajar en entornos de líneas de producción. Un autómata programable industrial es un dispositivo de control robusto diseñado especialmente para trabajar en ambientes donde las condiciones de trabajo requieren niveles de seguridad,

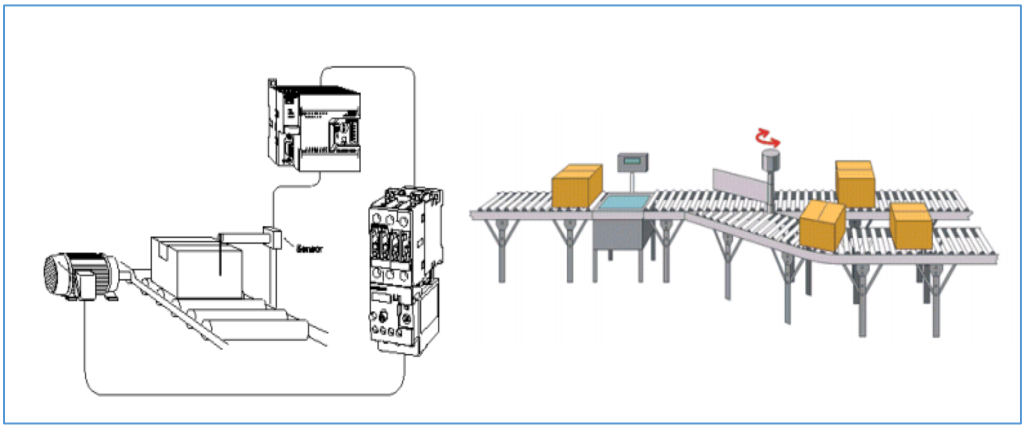


Figura 4: La tecnología programada en la industria

(Fuente: En "TEMA 1; AUTOMATIZACIÓN CON PLCs", por INFOPLC, 2010, del sitio web: http://www.infoplc.net/files/documentacion/automatas/infoPLC_net_1_Intro_ Automatas.pdf

1.7.2 Controlador lógico programable (PLC)

1.7.2.1 Definición de Autómata Programable (AP)

Según la Norma Europea IEC 61131 se define que un Autómata Programable (AP) es:

> *Un autómata programable (AP) es una máquina electrónica programable diseñada para ser utilizada en un entorno industrial, que utiliza una memoria programable para el almacenamiento interno de instrucciones orientada al usuario, para implantar soluciones específicas tales como funciones lógicas, secuencias, temporizaciones …., con el fin de controlar mediante entradas y salidas, digitales o analógicas diversos tipos de máquinas o procesos.* ***(MATEOS, 2001, p. 6)***

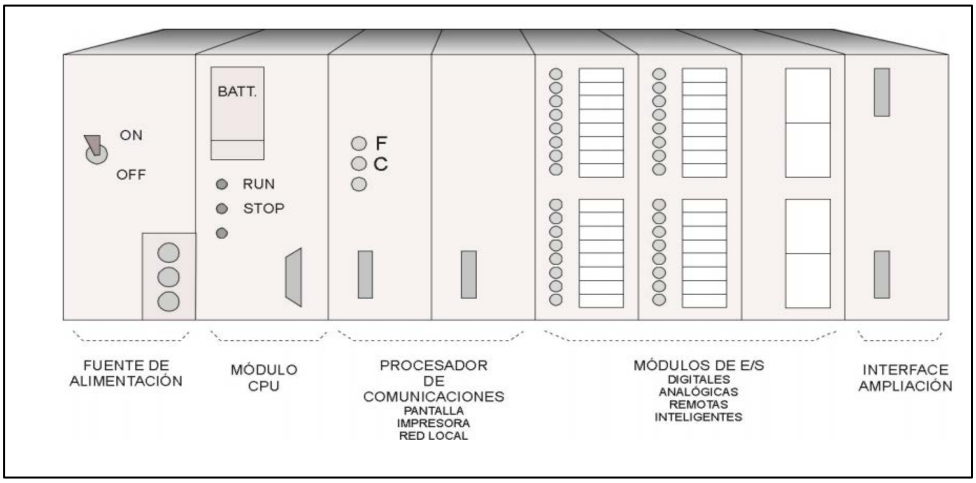


Figura 5: Componentes de un sistema automatizado

(Fuente: En "Sistema automatizado (PLCs)", por Felipe MATEOS, 2001, del sitio web: http://isa.uniovi.es/docencia/iea/teoria/plc.pdf

Considerando que un Controlador Lógico Programable (PLC) es un dispositivo electrónico de lógica digital, que puede ser programado por el usuario, destinado a controlar, supervisar máquinas o procesos lógicos y/o secuenciales, podemos afirmar que:

AUTÓMATA PROGRAMABLE (AP) = CONTROLADOR LÓGICO PROGRAMABLE (PLC)

1.7.2.2 Arquitectura de un PLC

El Controlador Lógico Programable (PLC) presenta la siguiente arquitectura estructura básica:

- La fuente de alimentación
- La Unidad de Procesamiento Central (CPU)
- El módulo de entradas digitales
- El módulo de salidas digitales
- El módulo de E/S analógicas
- Módulos especiales

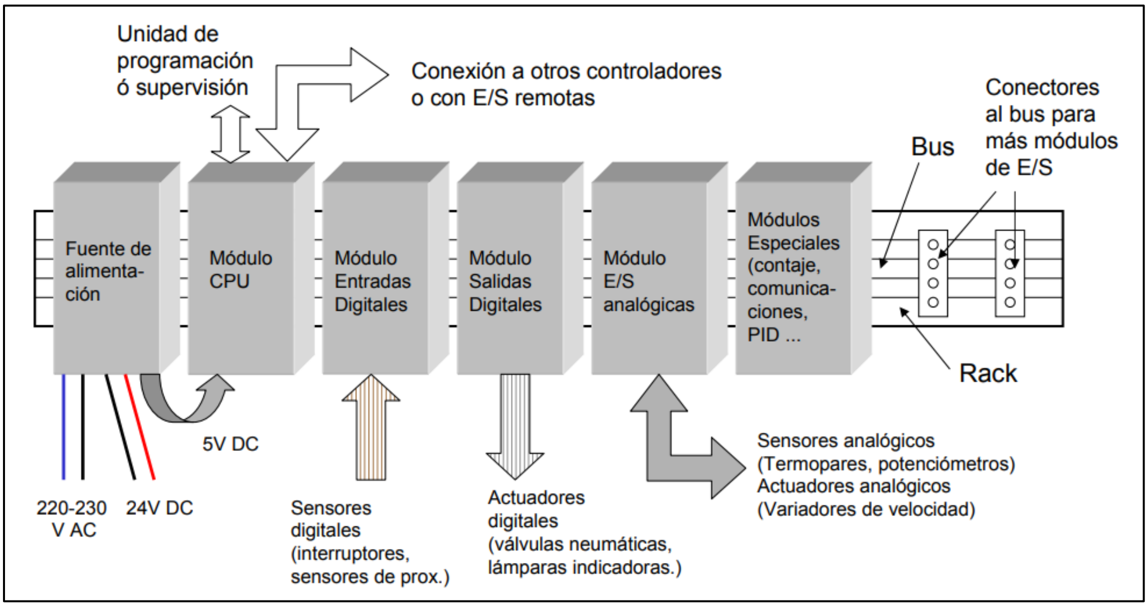

Unidad de programación ó supervisión
Conexión a otros controladores o con E/S remotas
Conectores al bus para más módulos de E/S
Bus
Fuente de alimentación
Módulo CPU
Módulo Entradas Digitales
Módulo Salidas Digitales
Módulo E/S analógicas
Módulos Especiales (contaje, comunicaciones, PID ...
Rack
5V DC
220-230 V AC
24V DC
Sensores digitales (interruptores, sensores de prox.)
Actuadores digitales (válvulas neumáticas, lámparas indicadoras.)
Sensores analógicos (Termopares, potenciómetros)
Actuadores analógicos (Variadores de velocidad)

Figura 6: Arquitectura de un PLC

(Fuente: En "Sistema automatizado (PLCs)", por Felipe MATEOS, 2001, del sitio web: http://isa.uniovi.es/docencia/iea/teoria/plc.pdf

1.7.2.3 Criterios de selección de un PLC

A continuación se presentan algunos criterios seguidos para la selección de un PLC, tanto criterios cualitativos como cuantitativos.

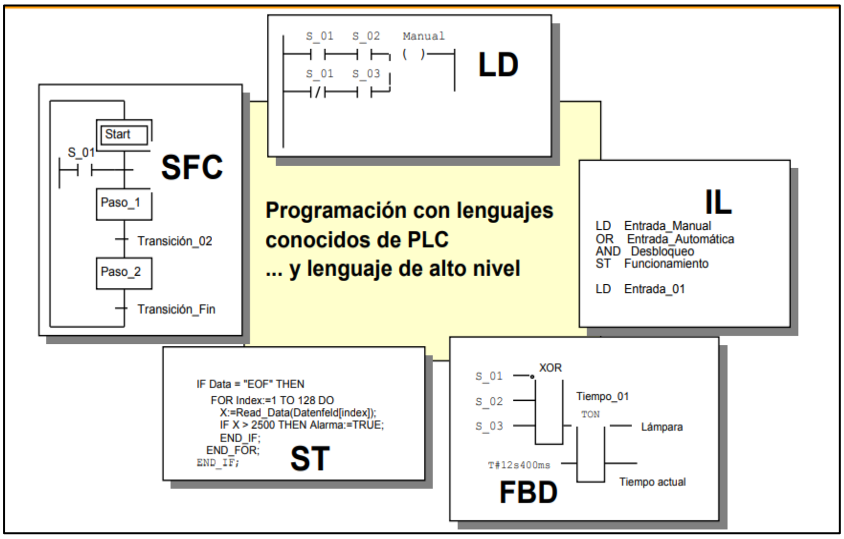
- Número de E/S a controlar, ya sean analógicas o digitales.
- Capacidad de la memoria de programa, acorde a la necesidad del proyecto a realizar y del proceso industrial a controlar.
- Potencia de las instrucciones, que está relacionado con la capacidad de procesamiento de la información en tiempo real.
- Posibilidad de conexión de periféricos, módulos especiales y comunicaciones del mismo fabricante.
- Compatibilidad con equipos de otras gamas.
- Normalización en planta, relacionado con la estandarización de los dispositivos de control.
- Fiabilidad del producto, relacionado con la robustez y confiabilidad que presta en la aplicación.
- Costo, el cual depende del fabricante y de la complejidad del proceso a automatizar.
- Previsión de repuestos. La existencia de un suministro asegurado de accesorios o equipos ayudará en la operatividad del sistema.

- Manuales de operación, configuración e instalación disponibles del producto en línea o físico.

1.7.2.4 Lenguajes de programación en un PLC

Los lenguajes de programación cumplen la función principal de establecer la comunicación entre la unidad de programación y el Controlador (PLC).

Es lenguajes de programación se clasifican en **visuales**: que emplean gráficos, y **escritos**: que usan conjunto de instrucciones, para la elaboración de un programa.



```
LD   Entrada_Manual
OR   Entrada_Automática
AND  Desbloqueo
ST   Funcionamiento

LD   Entrada_01
```

```
IF Data = "EOF" THEN
   FOR Index:=1 TO 128 DO
      X:=Read_Data(Datenfeld[index]);
      IF X > 2500 THEN Alarma:=TRUE;
      END_IF;
   END_FOR;
END_IF;
```

Figura 7: Lenguajes de Programación

(Fuente: En “Sistema automatizado (PLCs)”, por Felipe MATEOS, 2001, del sitio web: http://isa.uniovi.es/docencia/iea/teoria/plc.pdf

1.7.3 DEVICENET

1.7.3.1 Introducción

Siendo una red de comunicación industrial abierta basada en el modelo OSI, DEVICENET se puede definir como:

> *DeviceNet es una red de comunicación industrial de tipo serial que conecta a los controladores de una red con los dispositivos de entrada y salida. Es desarrollado por la empresa Rockwell Automation como un bus de comunicación abierto y hace su aparición en 1994. Actualmente está a cargo de la ODVA, acrónimo*

de la Open DeviceNet Vendor Association (Asociación de Vendedores DeviceNet Abierto). La ODVA cuenta con más de 300 empresas fabricantes de dispositivos asociadas. ***(CANAZA, 2017, p. 14)***

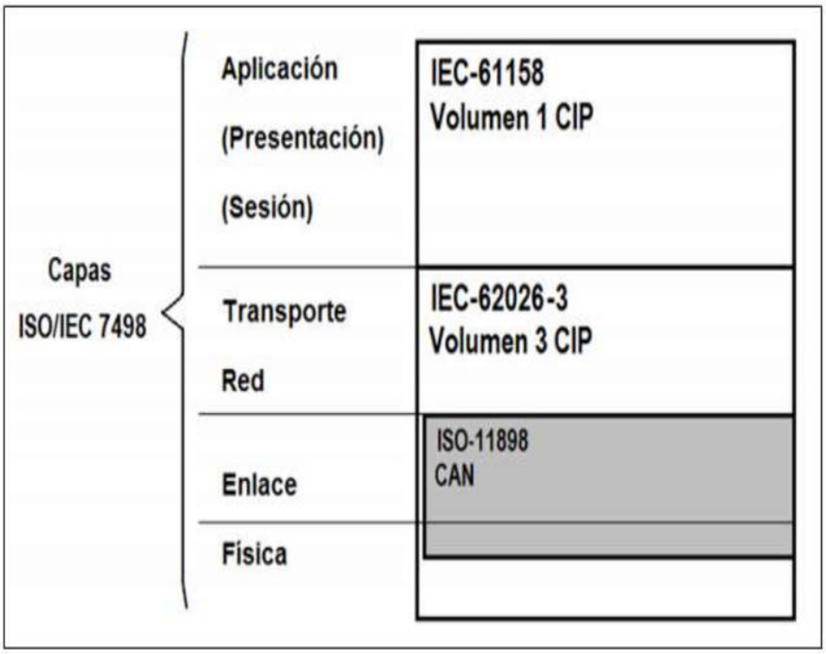


Figura 8: Representación de DeviceNet en el contexto de las estandarizaciones

(Fuente: "Repositorio de la Universidad San Agustin", Ch. CANAZA, 2017, del sitio web: http://repositorio.unsa.edu.pe/bitstream/handle/UNSA/3310/IEcavacj.pdf?sequence=1

1.7.2.2 Características de la red DEVICENET

Las características de posicionamiento de la red industrial DEVICENET indican:

La red DeviceNet se ubica, según la arquitectura NETLINX, en el nivel de dispositivo, es decir, en el nivel 1. Esto es representado en la figura siguiente. DeviceNet cumple su papel como la red de los dispositivos de bajo nivel, tales como sensores, botoneras y drives *entre otros. También conecta dispositivos de mayor nivel tales como PLC.* ***(CANAZA, 2017, p. 15)***

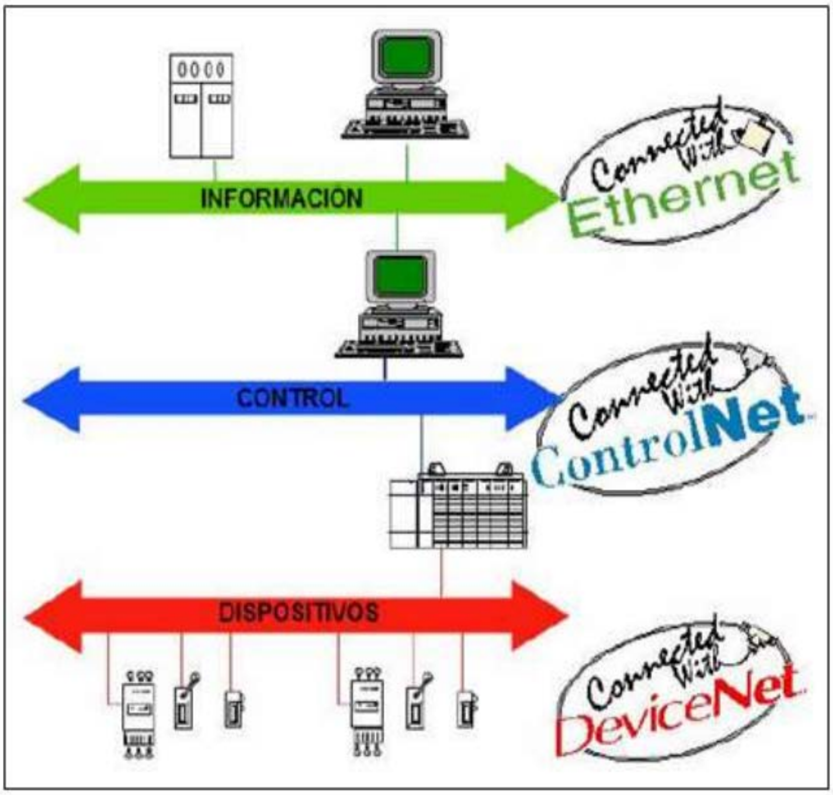


Figura 9: Arquitectura NETLINX

(Fuente: "Repositorio de la Universidad San Agustin", Ch. CANAZA, 2017, del sitio web: http://repositorio.unsa.edu.pe/bitstream/handle/UNSA/3310/IEcavacj.pdf?sequence=1

Las características de funcionamiento de la red industrial DEVICENET indican:

"El estándar de comunicación industrial DeviceNet está diseñado como un protocolo de funcionalidad media y de bajo costo para la conexión en red de sensores, actuadores y dispositivos asociados". ***(CANAZA, 2017, p. 15)***

La red DeviceNet permite utilizar hasta 64 nodos con una tasa de transmisión media de 125, 250 ó 500 kbp/s dependiendo de la longitud de la red. Los dispositivos pueden alimentarse a través del bus DeviceNet o disponer de su propia fuente de alimentación. ***(CANAZA, 2017, p. 16)***

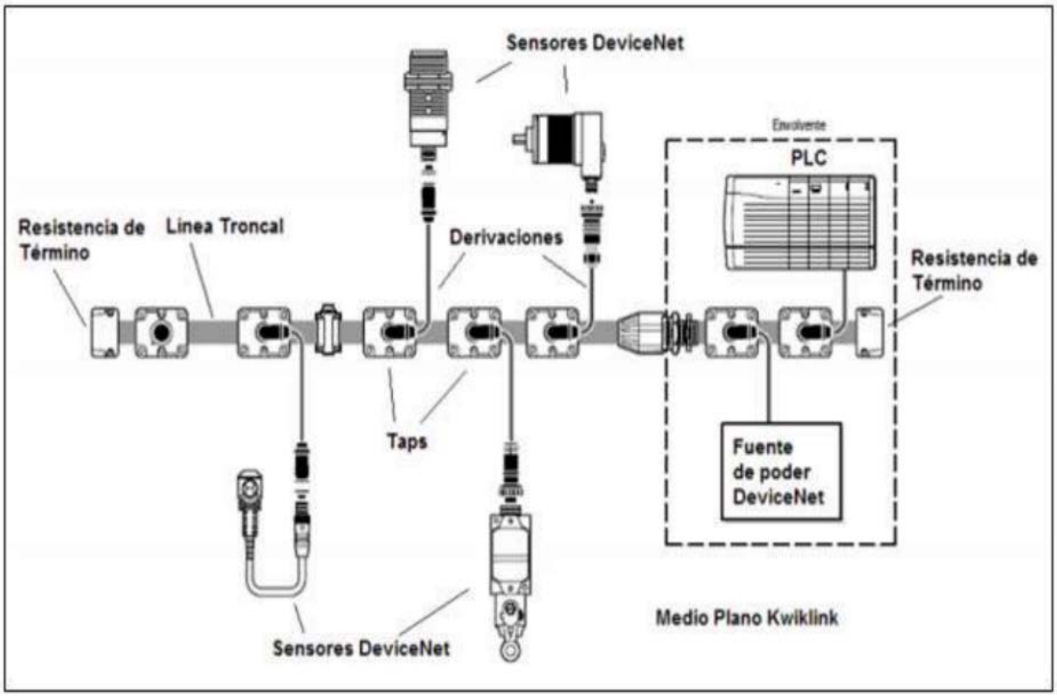


Figura 10: Configuración de red típica DeviceNet.

(Fuente: "Repositorio de la Universidad San Agustin", Ch. CANAZA, 2017, del sitio web: http://repositorio.unsa.edu.pe/bitstream/handle/UNSA/3310/IEcavacj.pdf?sequence=1

1.7.2.3 <u>Herramientas para la red DEVICENET</u>

Las herramientas para una red industrial DEVICENET se pueden dividir en tres grupos:

*"**Herramientas para la Capa Física**: Son herramientas en hardware y/o software que comprueban la integridad y cumplimiento de las especificaciones para la capa física y revisa la calidad en la transmisión de datos".* ***(CANAZA, 2017, p. 17)***

Herramientas de Configuración**: Son herramientas software que permiten comunicarse individualmente con dispositivos para monitoreo de datos y propósitos de configuración. Estos pueden ser desde simples software operativos sobre equipos manuales hasta poderosos paquetes de software para PC para la configuración de una red completa. La mayoría de las herramientas de configuración están basadas en los Electronic Data Sheets, EDS (Hojas de Datos Electrónicas). **(CANAZA, 2017, p. 17)

“**Herramientas de Monitoreo**: Por lo general son paquetes de software para PC que pueden capturar y mostrar las tramas CAN. Estas pueden mostrar tanto una trama CAN pura, así como su interpretación DeviceNet”. ***(CANAZA, 2017, p. 17)***

1.7.2.4 Arquitectura de campo de la red DEVICENET

La red industrial de campo DeviceNet, es una red de comunicación basada en el modelo de referencia ISO/OSI que utiliza niveles o capas. En la figura siguiente muestra la composición de la arquitectura de la red DeviceNet.

1.7.2.5 DeviceNet en el contexto de las estandarizaciones

ISO 11898**: El estándar ISO 11898 es la definición del protocolo de comunicación CAN (Controller Area Network), tanto de las versiones 2.A como 2.B del año 1991, desarrollado por la empresa BOSCH originalmente en 1984. Éste **implementa** una comunicación serial para dispositivos, y como se representa en la figura 5.1 anterior, define completamente una capa de enlace y parcialmente una capa física. Entorno a esta especificación Rockwell Automation desarrolló DeviceNet. **(MUÑOZ, 2007, p. 71)

IEC-62026-3: Esta norma del año 2000, es el estándar específico para las capas 1 a 4 de la red DeviceNet. Corresponde al Volumen 3 de CIP, en donde CIP es el acrónimo de Common Industrial Protocol (o Protocolo Industrial Comun). El Volumen 3 de CIP es la adaptación de este protocolo para DeviceNet. ***(MUÑOZ, 2007, p. 72)***

IEC 61158: Este estándar del año 2000, es la definición para los buses de campo. Esta norma incluye los Volúmenes 1 y 4 de CIP, en la forma de IEC 61158 type 2. El Volumen 1 de CIP especifica completamente las capas 5 a 7 del Modelo de Referencia OSI, como un protocolo común para cuatro redes **industriales**, que son las siguientes: • DeviceNet • ControlNet • Ethernet/IP • CompoNet. ***(MUÑOZ, 2007, p. 72)***

CAPÍTULO II
MARCO METODOLÓGICO

2.1 METODOLOGÍA

2.1.1 Tipos de Investigación

De acuerdo al estudio previsto, la implementación de módulos de entrenamiento de una red industrial, utilizando PLCs y variadores de frecuencia, basado en el estándar DEVICENET, y el desarrollo de una aplicación para el control de un motor trifásico, refleja un tipo de investigación descriptiva y cuasi - experimental y descriptiva.

2.1.2 Diseño de investigación

En el presente trabajo se utilizó en una primera etapa, la investigación descriptiva basada en la revisión bibliográfica de libros especializados, Tesis elaboradas en el Área de Automatización referidas a redes industriales y a la realización de módulos de entrenamientos, información revisada sobre sistema de control automatizada, manuales de operación, configuración e instalación de dispositivos de control del fabricante Rockwell INC, redes industriales basadas en el estándar DEVINET, las que se encuentran en la World Wide Web a través de direcciones URL. En una segunda etapa, se desarrolló una investigación de tipo experimental que permita desarrollar los módulos de entrenamientos basados en el estándar DEVICENET, orientada a resolver problemas de control de motores trifásicos asíncronos mediante variadores de frecuencia.

2.1.3 Hipótesis

“Con el diseño e implementación de un módulo de variador de frecuencia basado en comunicación DEVICENET, realizando la comunicación del autómata programable (PLC) al módulo variador de frecuencia, utilizando una lógica control, se demostrará un óptimo control de arranque y velocidad de trabajo de un motor trifásico, con lo cual permitirá al estudiante de Ingeniería Electrónica reforzar el aprendizaje sobre procesos de automatización y control, sobre el que está basado este módulo“

2.2. DISEÑO DEL MÓDULO VARIADOR DE FRECUENCIA

2.2.1 Descripción general del módulo

Se realizó el diseño de un módulo de variador de frecuencia, con el que se realiza un óptimo arranque y control de un motor trifásico, este módulo de frecuencia cuenta con comunicación DEVICENET. A través de un PLC se diseñó una lógica para su arranque y supervisión del motor, acompañado de un visualizador HMI realizado en el software FACTORY TALK VIEW.

Los elementos con que cuenta el módulo variador de frecuencia se muestran a continuación:

- Módulo variador Powerflex 70
- Switch principal
- Borneras de conexión
- Pulsadores nc y pilotos
- Cableado y conectores
- Plataforma del módulo.

Diagrama de bloques del sistema de control del módulo

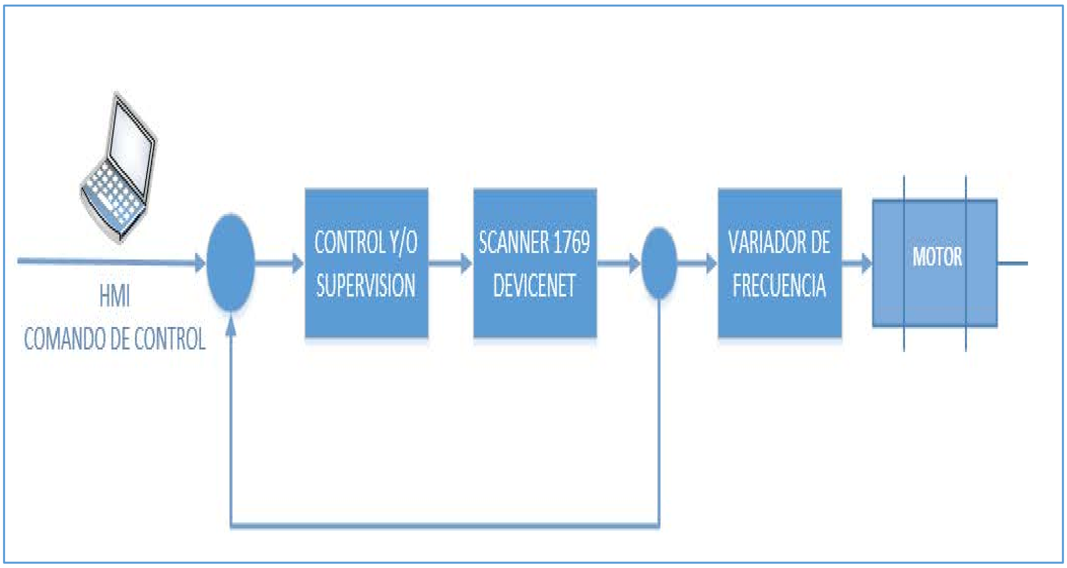


Figura 11: Diagrama de bloques de Nuestro Proceso

(Fuente: propia)

Diagrama esquemático el sistema de control del módulo

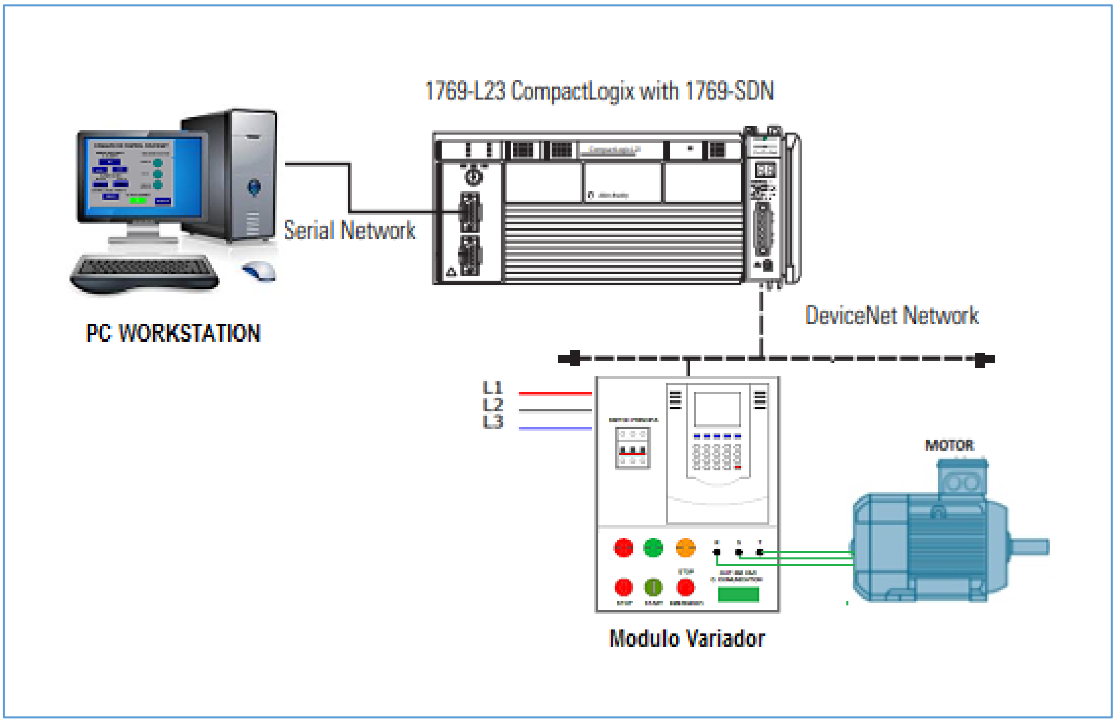


Figura 12: Diagrama esquemático de Nuestro Proceso

(Fuente: propia)

2.2.2 Diseño del soporte físico del módulo de entrenamiento

Para el diseño del soporte físico del módulo de entrenamiento de la red industrial DEVICENET, se tomó en cuenta los diseños existentes en el Laboratorio de Control y Automatización de la Universidad Privada de Tacna, con el fin de uniformizar o estandarizar los módulos que se van a implementar en dicho ambiente.

El Plano de la estructura física y del conexionado del soporte del módulo de entrenamiento fue desarrollado en AUTOCAD y validado por el responsable encargado del Laboratorio de Control y Automatización.

Para el diseño del plano de distribución, se tomó en cuenta las dimensiones de los dispositivos que iban a ser instalados.

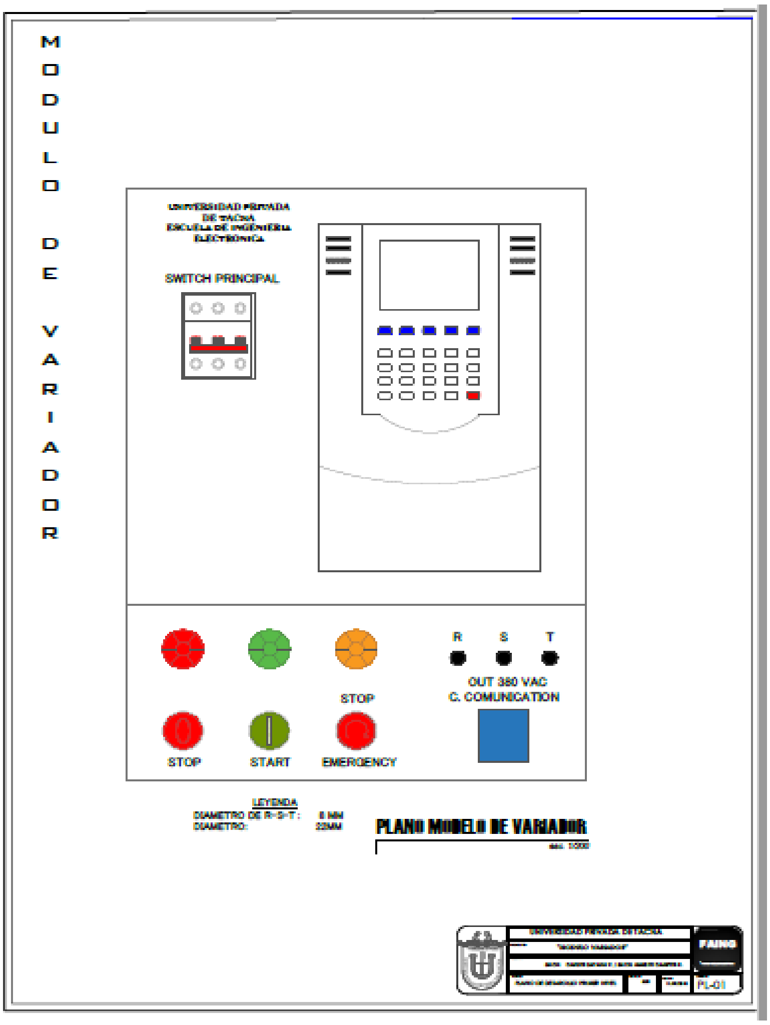


Figura 13: Plano final del módulo variador de frecuencia

(Fuente: propia)

Posteriormente, el diseño del soporte fue enviado para su fabricación de acuerdo al plano elaborado, con las mejoras sugeridas para optimizar el espacio empleado y tener una buena distribución de los dispositivos de control del módulo.

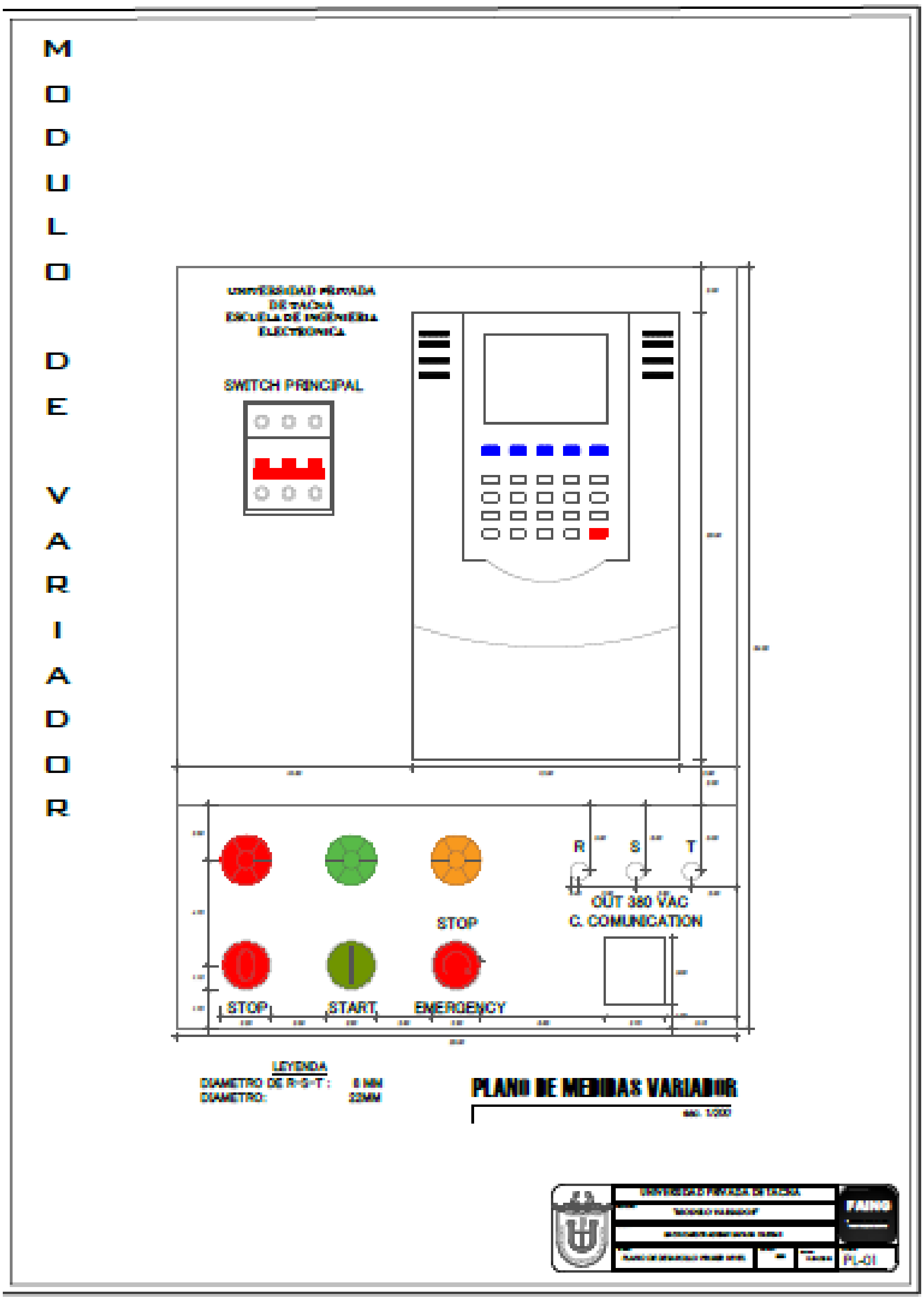


Figura 14: Plano final del módulo variador de frecuencia con medidas

(Fuente: propia)

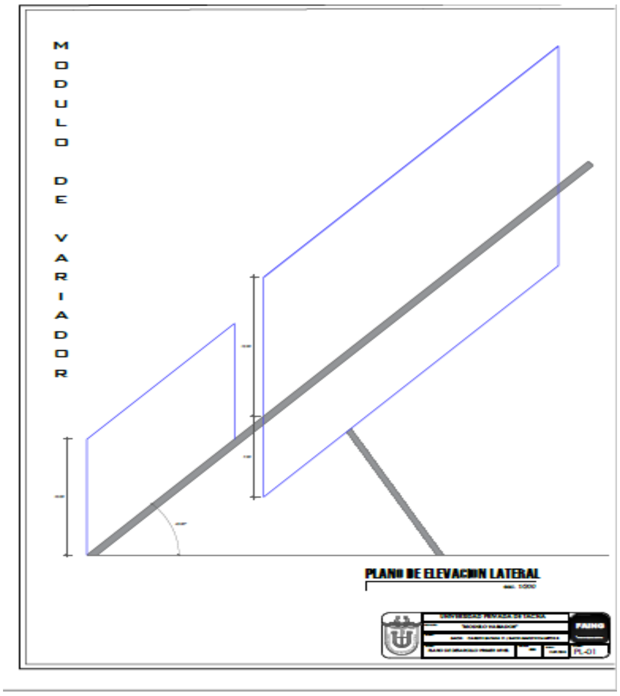


Figura 15: Plano de elevación final del módulo variador de frecuencia

(Fuente: propia)

2.2.3 Elementos del sistema de control

Módulo variador de frecuencia

El módulo Variador de Frecuencia consiste en un variador POWERFLEX 70, que está montado sobre una plataforma de metal, cuenta con pulsadores para su control local e indicadores de estados, con un switch principal para energizar el módulo, cableado interno para el sistema de potencia y también cableado para el control local.

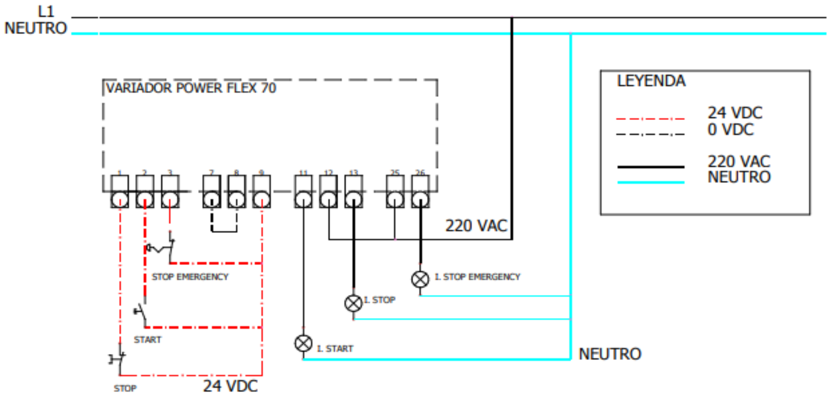


Figura 16: Diagrama de control

(Fuente propia)

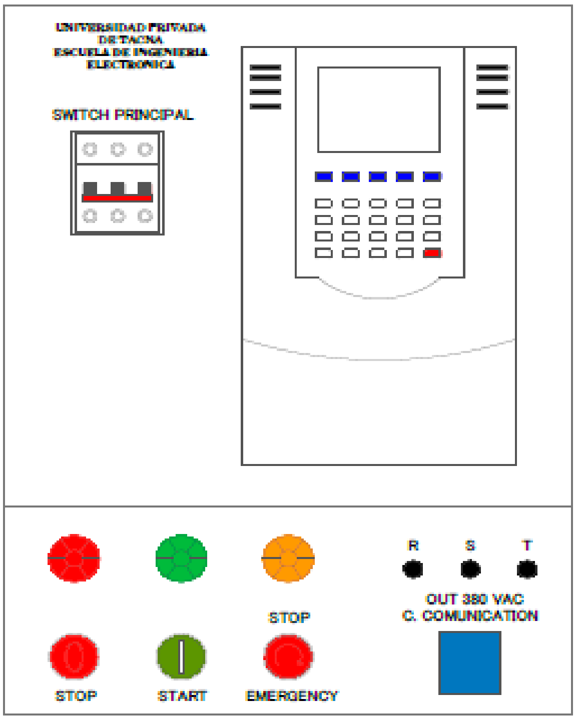


Figura 17: Diagrama esquemático

(Fuente propia)

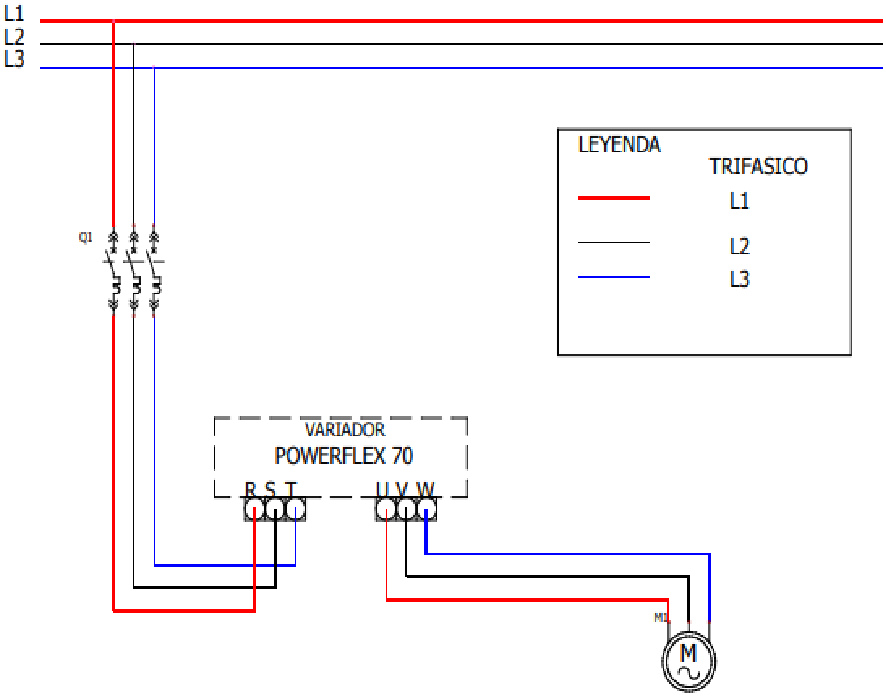


Figura 18: Diagrama de potencia

(Fuente propia)

PLC Compact Logic 1769-L23

El PLC 1769-L23, es un controlador compacto ideal para aplicaciones de control de baja complejidad, cuenta con la pre configuración de sus entradas y salidas y es limitado a un numero de módulos adicionales. Este controlador viene pre configurado con combinaciones para entradas y salidas digitales, analógicas y contadores de alta velocidad de E / S.

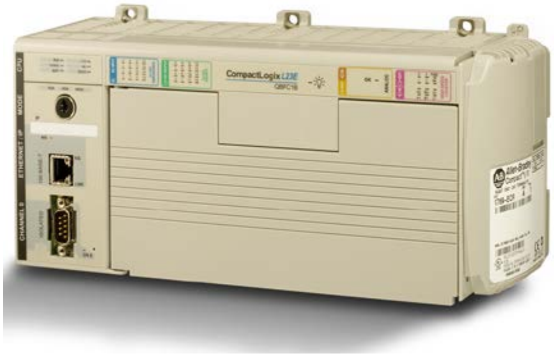


Figura 19: PLC Compact Logic 1769-L23

(Fuente "Van Meter Inc Shop", Van Meter Inc, 2017, extraído del sitio web: https://shop.vanmeterinc.com/1497216/Product/Allen-Bradley-1769-L23E)

Scanner 1769-SDN

El escáner usa las imágenes de datos de entrada y salida para transferir datos, estado e información de comandos entre el escáner y el controlador.

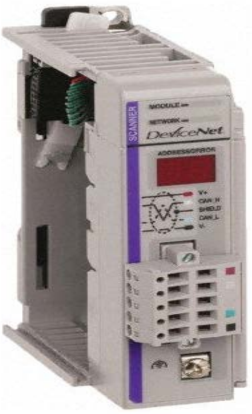


Figura 20: Scanner 1769-SDN

(Fuente "Octopart Shop", Octopart, 2017, extraído del sitio web: https://octopart.com/1769-sdn-allen+bradley-8017814)

Diagrama general del módulo variador d frecuencia

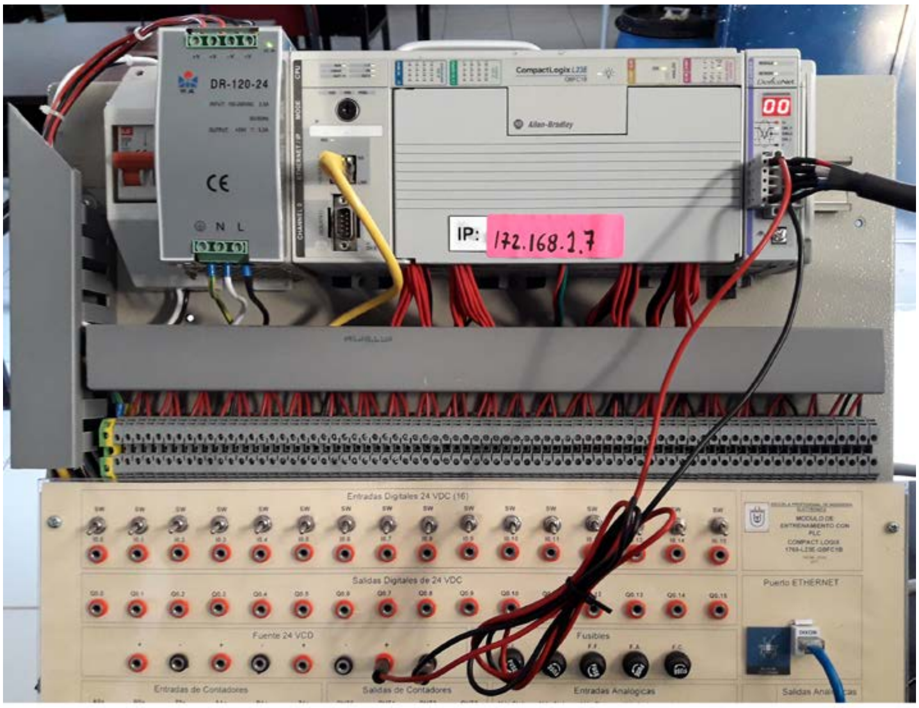


Figura 21: PLC Compact Logic 1769-L23 + SCANNER 1769-SDN

(Fuente propia)

2.3 IMPLEMENTACIÓN DEL MÓDULO VARIADOR DE FRECUENCIA

En total se implementaron dos módulos con variador de frecuencia, para el control de arranque y velocidad de un motor trifásico en una red de comunicación industrial DEVICENET, empleando el PLC Compact Logic 1769-L23 de Allen Bradley.

Los módulos implementados obedecen en su diseño a un sistema de control abierto los que permitirán el control de arranque y velocidad de un motor asíncrono trifásico mediante un variador de frecuencia POWERFLEX 70 de Rockwell Inc.

2.3.1 Guía de interconexión de componentes del módulo variador

1. Establecemos la interconexión de la Estación de Trabajo con el PLC 1769-L23, entonces se conecta el conector RJ-45 del cable Ethernet al puerto Ethernet (puerto superior) del controlador.

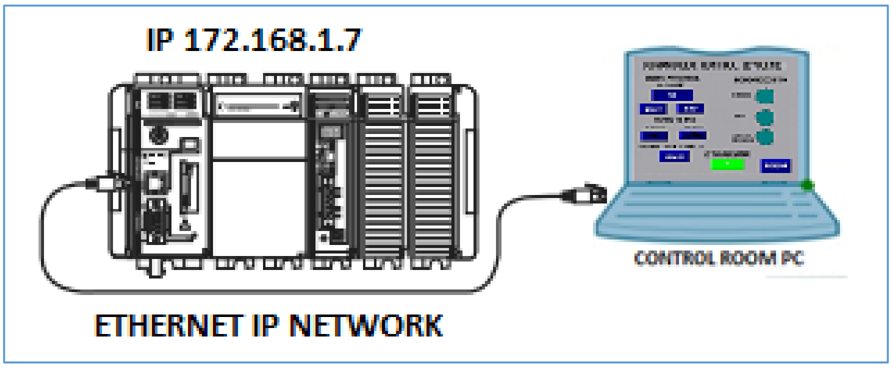


Figura 22: Comunicación Ethernet IP

(Fuente propia)

2. Instalación del módulo scanner 1769 - SDN en el controlador PLC1769-L23

2.1 <u>Descripción del módulo Scanner</u>

TABLA

Descripción del modulo scanner 1769

1	Palanca de bus (con función de enclavamiento	**6**	Tornillo de tierra
2A	Seguro superior para el riel DIN	**7A**	Conector mach de empalme DeviceNet
2B	Seguro inferior para el riel DIN	**7B**	Conector hembra DeviceNet extraíble
3A	Lengüeta superior para montaje en el panel	**8A**	Conector de bus móvil con pines hebra
3B	Lengüeta inferior para montaje en el panel	**8B**	Conector de bus con pines macho
4	Indicadores LED de estado de modulo y red	**9**	Etiqueta de la placa del fabricante
5	Pantallas numéricas de dirección y error		

(Tomada de "Módulo escáner Compact I/O DeviceNet", Rockwell, 2002, p.6 del sitio web: https://literature.rockwellautomation.com/idc/groups/ literature/documents/in/1769-in060_-es-p.pdf

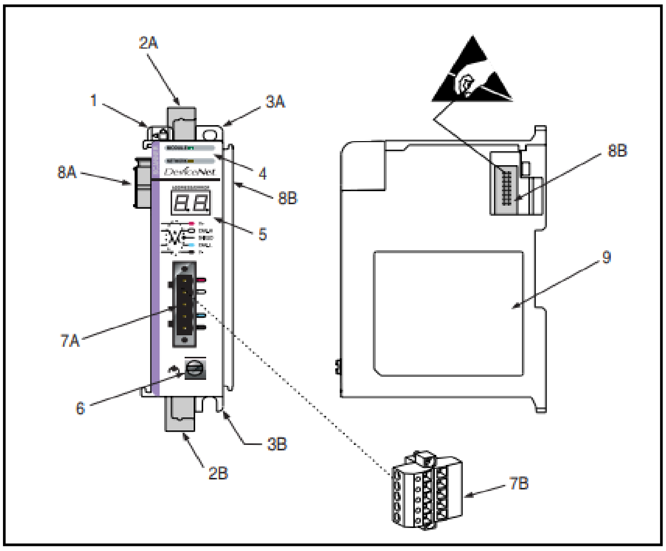


Figura 23: Módulo escáner Compact I/O DeviceNet

(Fuente: "Módulo escáner Compact I/O DeviceNet", Rockwell, 2002, del sitio web: ttps://literature.rockwellautomation.com/idc/groups/literature/ documents/in/1769-in060_-es-p.pdf)

2.2 Montaje en el riel DIN

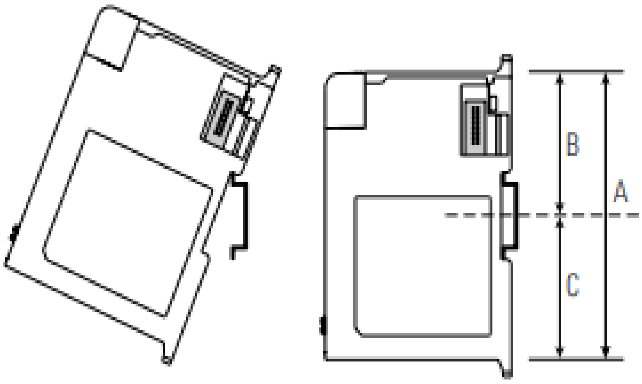


Figura 24: Montaje del módulo Scanner en el controlador 1769-L23

(Fuente: "Módulo escáner Compact I/O DeviceNet", Rockwell, 2002, del sitio web: ttps://literature.rockwellautomation.com/idc/groups/literature/ documents/in/1769-in060_-es-p.pdf)

2.3 Ensamblaje del módulo

El procedimiento de cómo ensamblar el sistema Compact I/O, se explica en el **manual de Rockwell "Módulo escáner Compact I/O DeviceNet"**

- *Desconecte la alimentación eléctrica.*

- *Verifique que la palanca del bus del módulo (A) esté en la posición desbloqueada (hasta el tope hacia la derecha).*
- *Use las ranuras de machihembrado superior e inferior (B) para fijar los módulos juntos.*
- *Mueva el módulo nuevamente a lo largo de las ranuras de machihembrado, hasta que los conectores del bus (C) queden alineados.*
- *Use sus dedos o un destornillador pequeño para empujar ligeramente la palanca del bus hacia atrás y dejar libre la lengüeta de posicionamiento (D).*
- *Mueva la palanca del bus del módulo hasta el tope hacia la izquierda (E) hasta que escuche un chasquido. Asegúrese de que haya quedado bien enclavada en el lugar debido*
- *Acople una tapa de terminación final (F) en el último módulo del sistema usando las ranuras de machihembrado como se hizo anteriormente.*
- *Fije la terminación de tapa final del bus (G).* ***(Rockwell, 2002, pp. 10-11)***

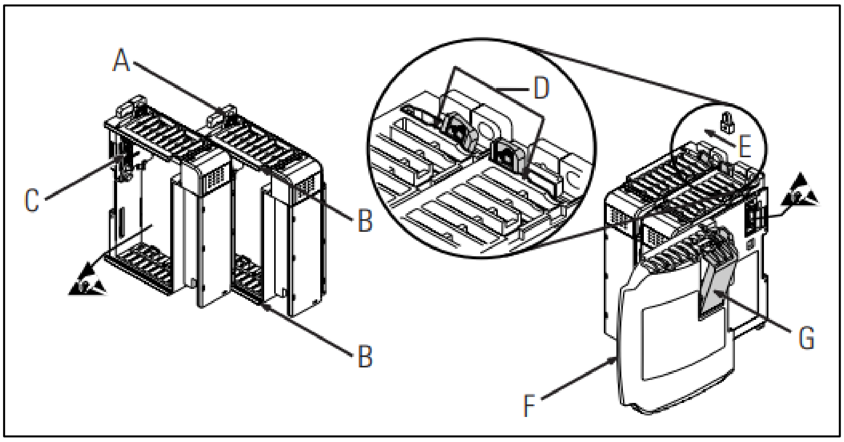


Figura 25: Ensamblaje del módulo

(Fuente: "Módulo escáner Compact I/O DeviceNet", Rockwell, 2002, del sitio web: ttps://literature.rockwellautomation.com/idc/groups/literature/ documents/in/1769-in060_-es-p.pdf)

3. Una vez instalado el módulo scanner en el controlador procedemos a las conexiones del cableado de campo.

   El procedimiento se explica en el **manual de Rockwell “Módulo escáner Compact I/O DeviceNet”**

- *Se procede a conectar el cable DeviceNet al conector extraíble.*
- *Insertar el conector hembra extraíble en el conector macho de empalme ubicado en el módulo escáner DeviceNet.*
- *Se procede a fijar el conector extraíble a la caja del escáner con los tornillos de montaje superior e inferior.* ***(Rockwell, 2002, p. 16)***

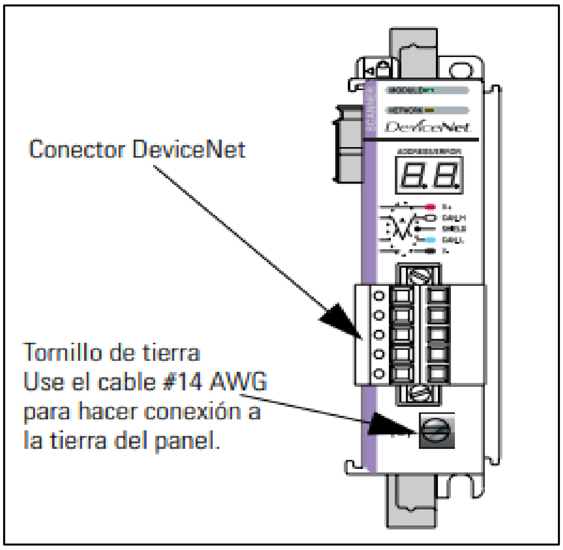


Figura 26: Cableado DEVICENET

(Fuente: “Módulo escáner Compact I/O DeviceNet”, Rockwell, 2002, del sitio web: ttps://literature.rockwellautomation.com/idc/groups/literature/ documents/in/1769-in060_-es-p.pdf)

4. Una vez realizado el cable de comunicación DeviceNet procedemos a conectarlos con el módulo de variador el puerto se encuentra en la parte frontal como se muestra en la figura siguiente:

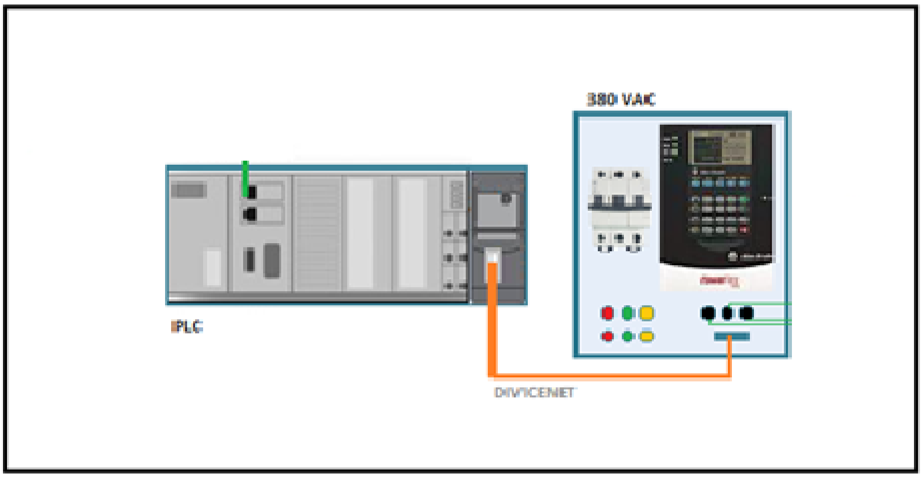


Figura 27: Conexión del cableado DeviceNet al Módulo Variador

(Fuente propia)

5. Como paso final, alimentamos al variador con voltaje trifásico de 380 vac y de la salida del variador se alimentara el motor.

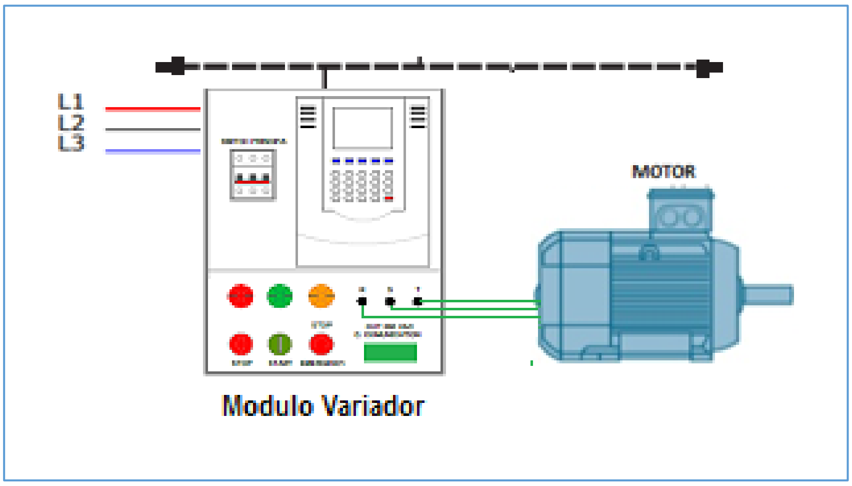


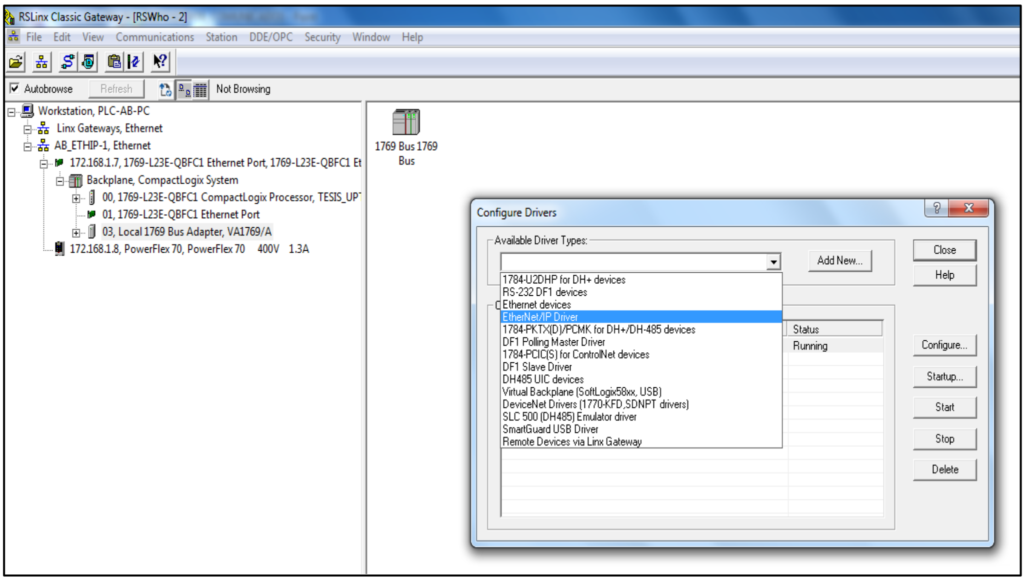
Figura 28: Conexión del motor trifásico al Módulo Variador

(Fuente propia)

2.3.2 Configuración del software de control

2.3.2.1 <u>Configurando la plataforma Rslinx</u>

- Empezamos abriendo el RSLINX CLASSIC, el cual nos permite establecer una comunicación de nuestra estación de trabajo (computador, Laptop, Estación Remota) y el autómata programable que en este caso es PLC Allen Bradley 1769-L23E-QBFC1B. Ahora seleccionamos el tipo de controlador para nuestra red seleccionada que nos permita acceder al PLC. Escogemos controlador Ethernet/Ip Driver, esto debido al que el PLC Allen Bradley 1769-L23E-QBFC1B cuenta con dos puertos de comunicación, Ethernet 10/100 y Serial RS 232 como se aprecia en la figura.



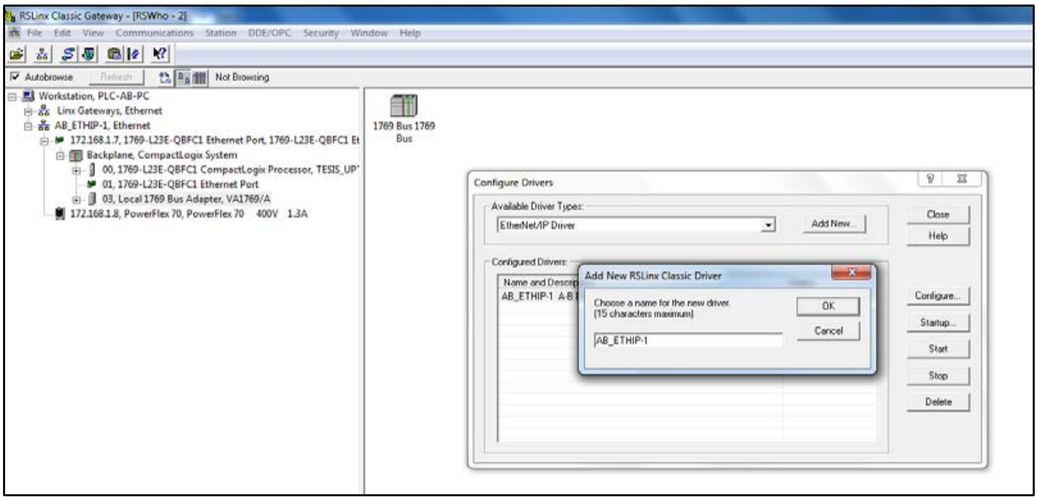
Workstation, PLC-AB-PC
Linx Gateways, Ethernet
AB_ETHIP-1, Ethernet
172.168.1.7, 1769-L23E-QBFC1 Ethernet Port, 1769-L23E-QBFC1 Et
Backplane, CompactLogix System
00, 1769-L23E-QBFC1 CompactLogix Processor, TESIS_UP
01, 1769-L23E-QBFC1 Ethernet Port
03, Local 1769 Bus Adapter, VA1769/A
172.168.1.8, PowerFlex 70, PowerFlex 70 400V 1.3A

1784-U2DHP for DH+ devices
RS-232 DF1 devices
Ethernet devices
EtherNet/IP Driver
1784-PKTX(D)/PCMK for DH+/DH-485 devices
DF1 Polling Master Driver
1784-PCIC(S) for ControlNet devices
DF1 Slave Driver
DH485 UIC devices
Virtual Backplane (SoftLogix58xx, USB)
DeviceNet Drivers (1770-KFD,SDNPT drivers)
SLC 500 (DH485) Emulator driver
SmartGuard USB Driver
Remote Devices via Linx Gateway

Figura 29: Selección de controlador

(Fuente propia)

- Escogemos el nombre para el controlador como se ve en la siguiente imagen.



Workstation, PLC-AB-PC
Linx Gateways, Ethernet
AB_ETHIP-1, Ethernet
172.168.1.7, 1769-L23E-QBFC1 Ethernet Port, 1769-L23E-QBFC1 Et
Backplane, CompactLogix System
00, 1769-L23E-QBFC1 CompactLogix Processor, TESIS_UP
01, 1769-L23E-QBFC1 Ethernet Port
03, Local 1769 Bus Adapter, VA1769/A
172.168.1.8, PowerFlex 70, PowerFlex 70 400V 1.3A

Available Driver Types: EtherNet/IP Driver

AB_ETHIP-1 A-B

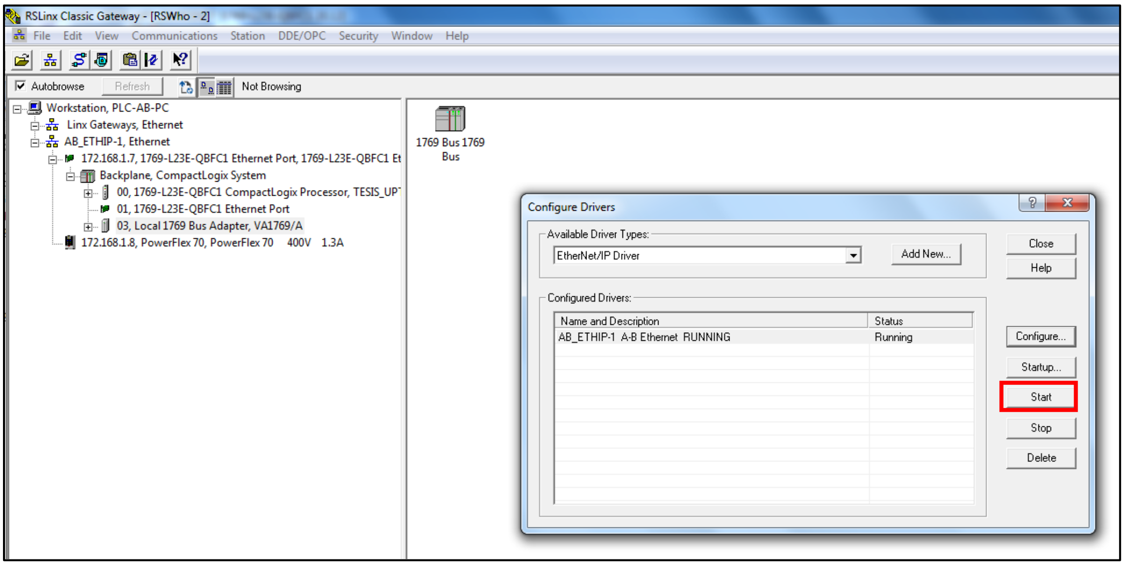
Choose a name for the new driver.
(15 characters maximum)

AB_ETHIP-1

Figura 30: Definiendo nombre del controlador

(Fuente propia)

- Establecido el nombre de nuestro controlador, procederemos a correr el drive para el reconocimiento del PLC y todos sus dispositivos conectados.



Workstation, PLC-AB-PC
Linx Gateways, Ethernet
AB_ETHIP-1, Ethernet
172.168.1.7, 1769-L23E-QBFC1 Ethernet Port, 1769-L23E-QBFC1 Et
Backplane, CompactLogix System
00, 1769-L23E-QBFC1 CompactLogix Processor, TESIS_UP
01, 1769-L23E-QBFC1 Ethernet Port
03, Local 1769 Bus Adapter, VA1769/A
172.168.1.8, PowerFlex 70, PowerFlex 70 400V 1.3A

Available Driver Types: EtherNet/IP Driver

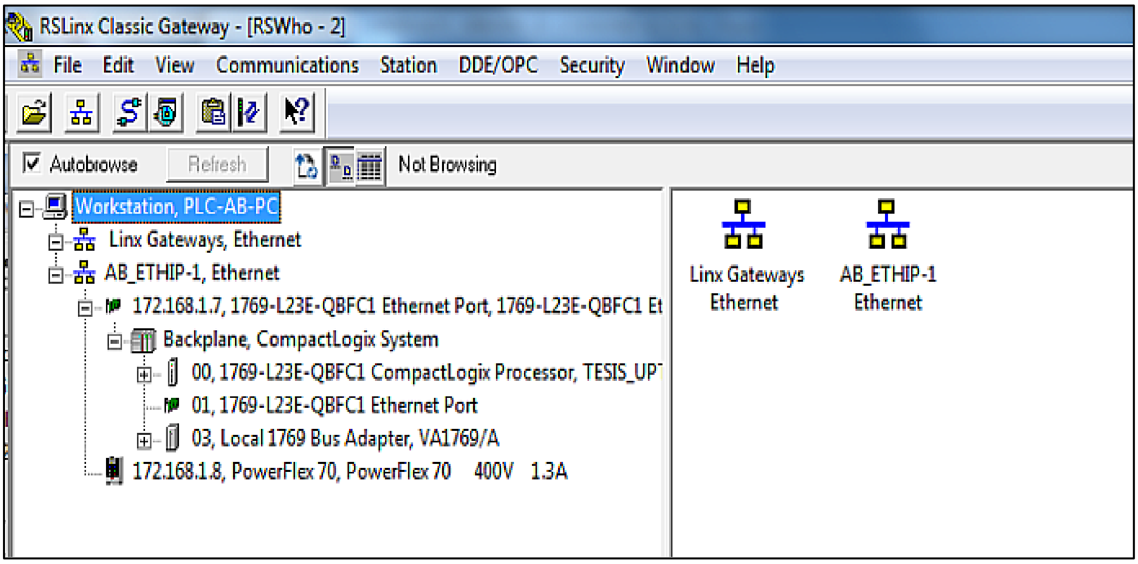
Configured Drivers:

Name and Description	Status
AB_ETHIP-1 A-B Ethernet RUNNING	Running

Figura 31: Reconocimiento del controlador

(Fuente propia)

- Verificamos con RSWho que este corriendo el controlador creado (AB-ETHIP-1, Ethernet), donde podremos observar y analizar las propiedades del PLC.


RSLinx Classic Gateway - [RSWho - 2]
File Edit View Communications Station DDE/OPC Security Window Help
☑ Autobrowse
Refresh
Not Browsing
Linx Gateways Ethernet
AB_ETHIP-1 Ethernet

Workstation, PLC-AB-PC
Linx Gateways, Ethernet
AB_ETHIP-1, Ethernet
172.168.1.7, 1769-L23E-QBFC1 Ethernet Port, 1769-L23E-QBFC1 Et
Backplane, CompactLogix System
00, 1769-L23E-QBFC1 CompactLogix Processor, TESIS_UP
01, 1769-L23E-QBFC1 Ethernet Port
03, Local 1769 Bus Adapter, VA1769/A
172.168.1.8, PowerFlex 70, PowerFlex 70 400V 1.3A

Figura 32: verificación del controlador creado AB-ETHIP-1, Ethernet

(Fuente propia)

2.3.2.2 Configurando la plataforma RSNETWORX

- Procedemos a abrir RSNETWORX, para poder configurar la red DeviceNet y así poder encontrar y configurar los dispositivos en su red rápidamente a través de esta sencilla interfaz. Procedemos a buscar los dispositivos asociados a la red DeviceNet con Network online.

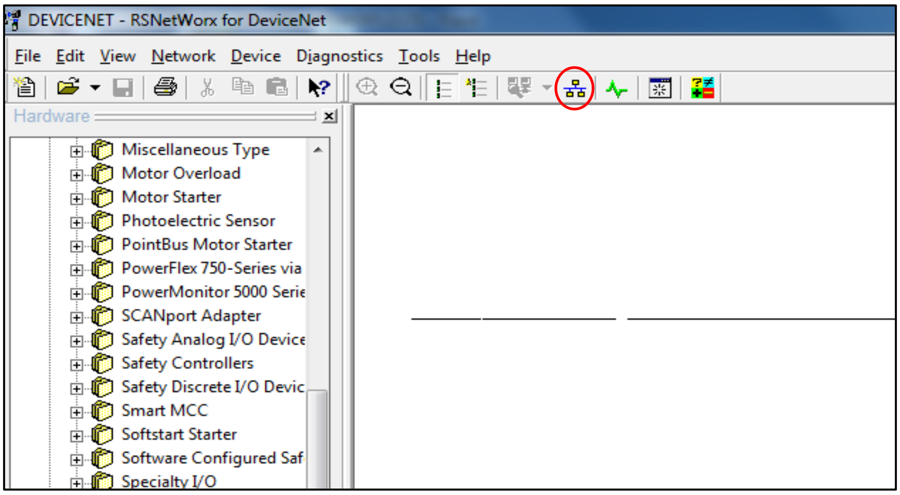


Figura 33: Escaneo de dispositivos en la red DeviceNet

(Fuente propia)

Con la configuración en línea nos permite:

- La habilidad de ver a todos los dispositivos que están disponibles y comunicándose en la red antes de hacer algún cambio.
- La habilidad de ver dispositivos que no se comunican más en la red.
- Escaneado los dispositivos en la red DevicNet, procederemos a cargar los valores y parámetros de los dispositivos encontrados.
- La habilidad de ver y resolver discordancias entre los dispositivos de la configuración y los físicamente instalados.
- Elegir entre cargar la configuración de algún dispositivo o descargar la configuración a alguno o todos los dispositivos.
- Capacidades de diagnóstico

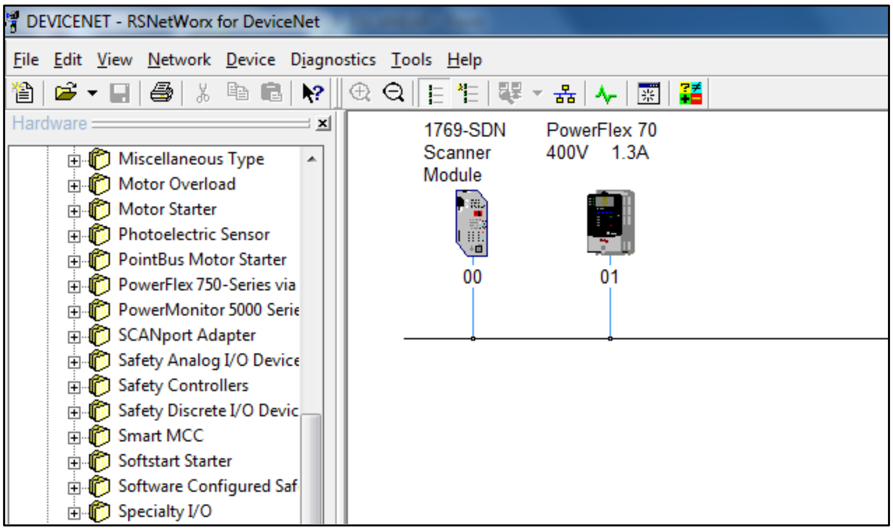


Figura 34: Dispositivos encontrado en la Red DeviceNet (Fuente propia)

- Procedemos a cargar los parámetros y datos de los dispositivos encontrados, seleccionamos el dispositivo en nuestro caso el variador Powerflex 70 y entramos a propiedades del variador.

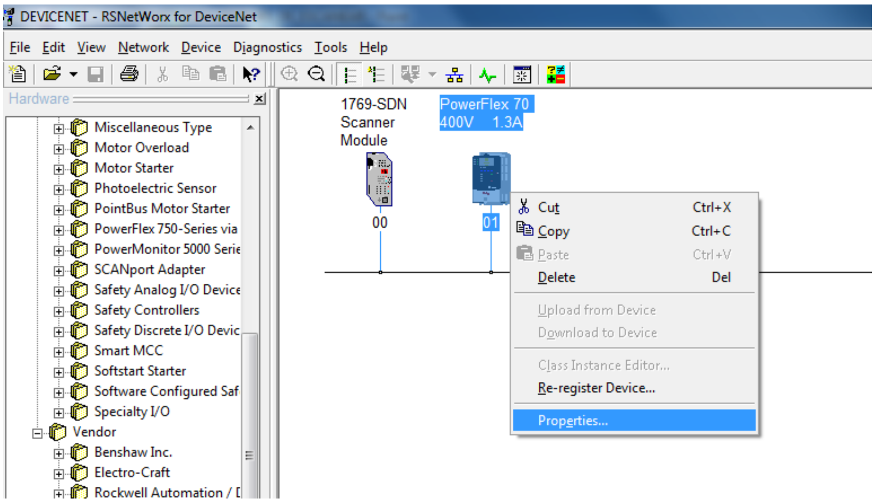
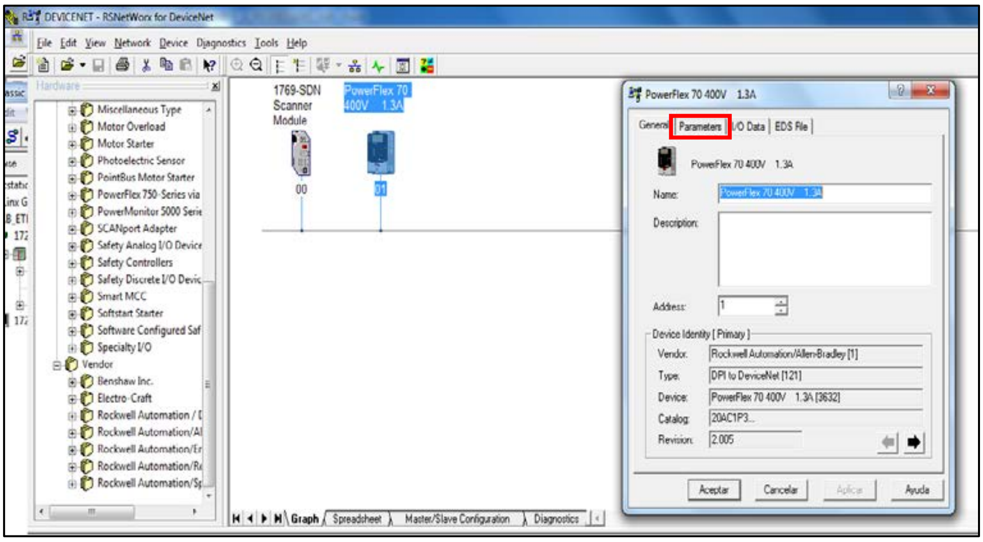


Figura 35: Carga de parámetro de dispositivos encontrado (Fuente propia)

- En propiedades del variador vamos a la pestaña de parámetros, aparece una ventana emergente que nos mostrara la opción de UP-LOAD y DOWN-LOAD, elegimos UP-LOAD, así se carga los valores y parámetros del variador.

Para crear y trabajar con una configuración de red en línea, es importante comprender las implicaciones de cargar y descargar.

- Carga (Uploading): Es el proceso de obtener los datos desde la red física y mostrarlos en el software de configuración.
- Descarga (Downloading): Es el proceso de enviar datos desde el software de configuración a la red física.



PowerFlex 70 400V 1.3A

Name: PowerFlex 70 400V 1.3A

Description:

Address: 1

Device Identity [Primary]

Vendor: Rockwell Automation/Allen-Bradley [1]

Type: DPI to DeviceNet [121]

Device: PowerFlex 70 400V 1.3A [3632]

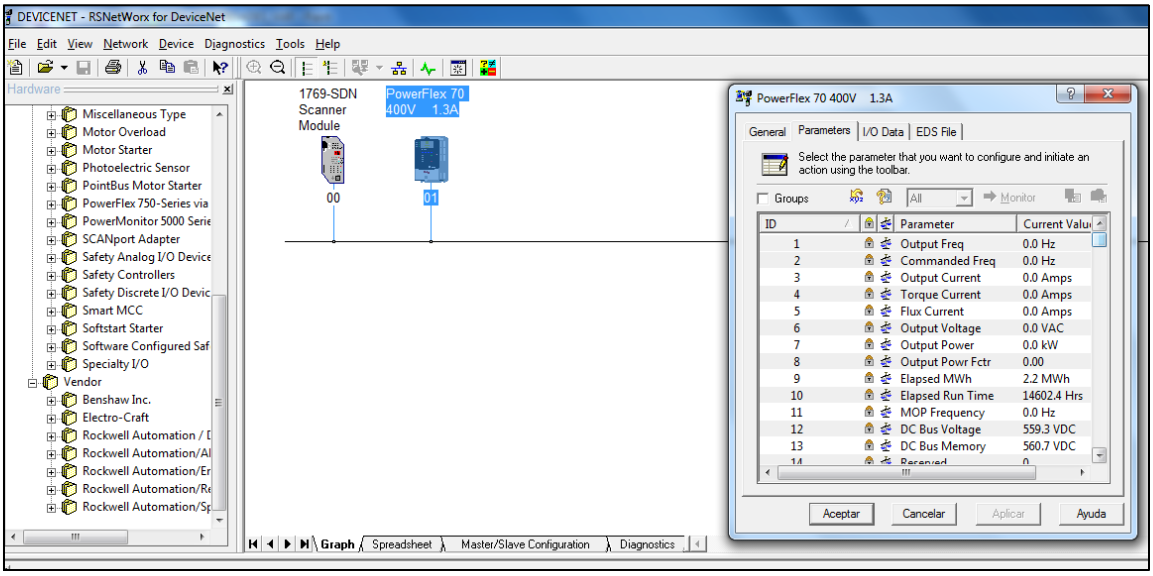
Catalog: 20AC1P3...

Revision: 2.005

Figura 36: Ventana con las Propiedades del Variador PowerFlex 70

(Fuente propia)

- Cargados los parámetros del Variador powerflex 70, aparecerá una lkista con todos los parámetros del variador PowerFlex 70 y también se define las entradas y salidas del dispositivo

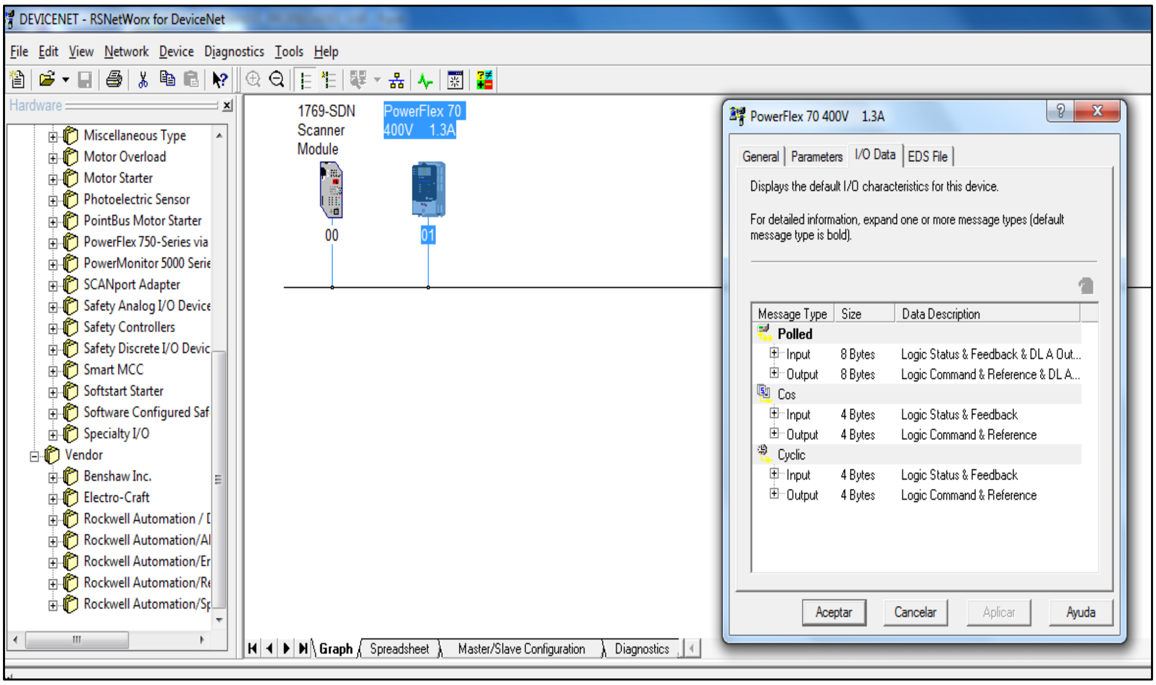


Select the parameter that you want to configure and initiate an action using the toolbar.

ID	Parameter	Current Valu
1	Output Freq	0.0 Hz
2	Commanded Freq	0.0 Hz
3	Output Current	0.0 Amps
4	Torque Current	0.0 Amps
5	Flux Current	0.0 Amps
6	Output Voltage	0.0 VAC
7	Output Power	0.0 kW
8	Output Powr Fctr	0.00
9	Elapsed MWh	2.2 MWh
10	Elapsed Run Time	14602.4 Hrs
11	MOP Frequency	0.0 Hz
12	DC Bus Voltage	559.3 VDC
13	DC Bus Memory	560.7 VDC
14	Reserved	0

Figura 37: parámetros cargados del variador PowerFlex 70

(Fuente propia)



Displays the default I/O characteristics for this device.

For detailed information, expand one or more message types (default message type is bold).

Message Type	Size	Data Description
Polled		
Input	8 Bytes	Logic Status & Feedback & DL A Out...
Output	8 Bytes	Logic Command & Reference & DL A...
Cos		
Input	4 Bytes	Logic Status & Feedback
Output	4 Bytes	Logic Command & Reference
Cyclic		
Input	4 Bytes	Logic Status & Feedback
Output	4 Bytes	Logic Command & Reference

Figura 38: Entradas y Salidas del Variador PowerFlex 70

(Fuente propia)

- Ahora procedemos hacer lo mismo con el MODULO SCANER 1769 –SDN, entramos a propiedades del módulo escáner.

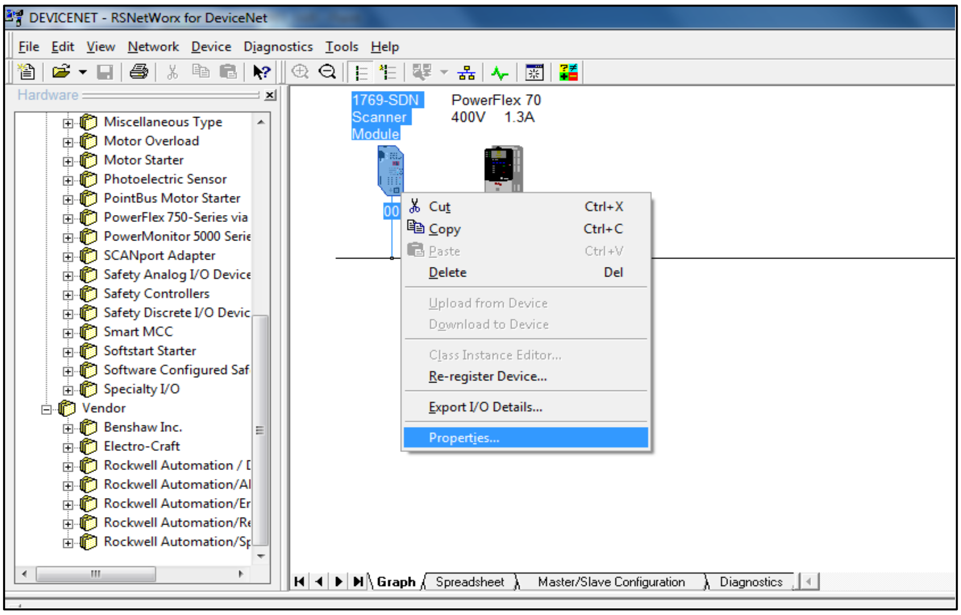


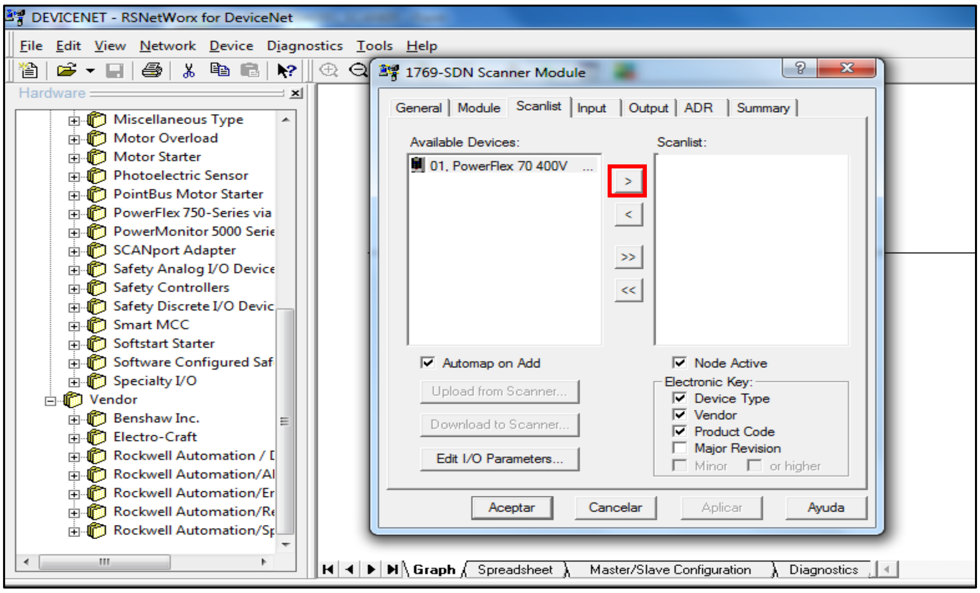
Figura 39: Ingresando a propiedades del SCANER 1769 –SDN

(Fuente propia)

En la pestaña ScanList, se observa el variador en la lista de dispositivos disponible, vamos a pasarla a la lista de escaneo como se ve en la figura.

La scanlist es la lista de dispositivos en la red con la cual el módulo escáner se comunicará. La lista de escán debe existir en un módulo escáner para que la comunicación exista entre los dispositivos y el procesador. La lista de escán provee al módulo de escáner con la siguiente información:

- Cuales dispositivos escanear.
- Como escanear a cada dispositivo.
- Qué tanto corresponde el dispositivo en la lista con el dispositivo en la red.
- Qué tan a menudo el dispositivo es escaneado.
- Donde los datos pueden ser encontrados en la memoria de cada dispositivo
- El tamaño de datos de entrada y de salida
- Donde los datos son mapeados en el módulo escáner con la finalidad que el controlador pueda efectuar la lectura correspondiente.



Available Devices: 01, PowerFlex 70 400V

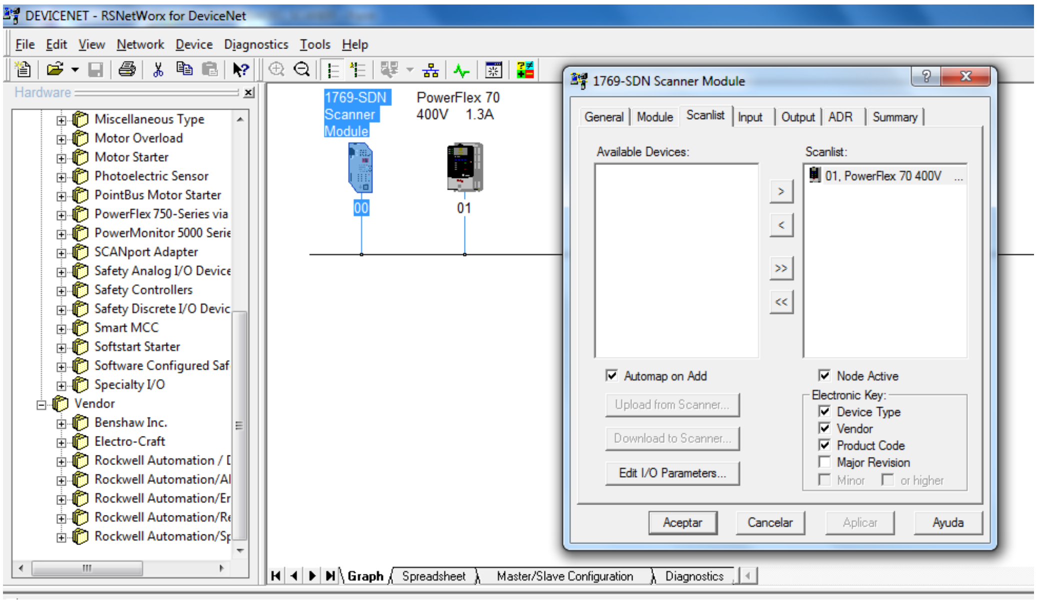
☑ Automap on Add

☑ Node Active

Electronic Key:
☑ Device Type
☑ Vendor
☑ Product Code
☐ Major Revision
☐ Minor ☐ or higher

Figura 40: Pasando Dispositivo PowerFlex 70 a lista de escaneo.

(Fuente propia)



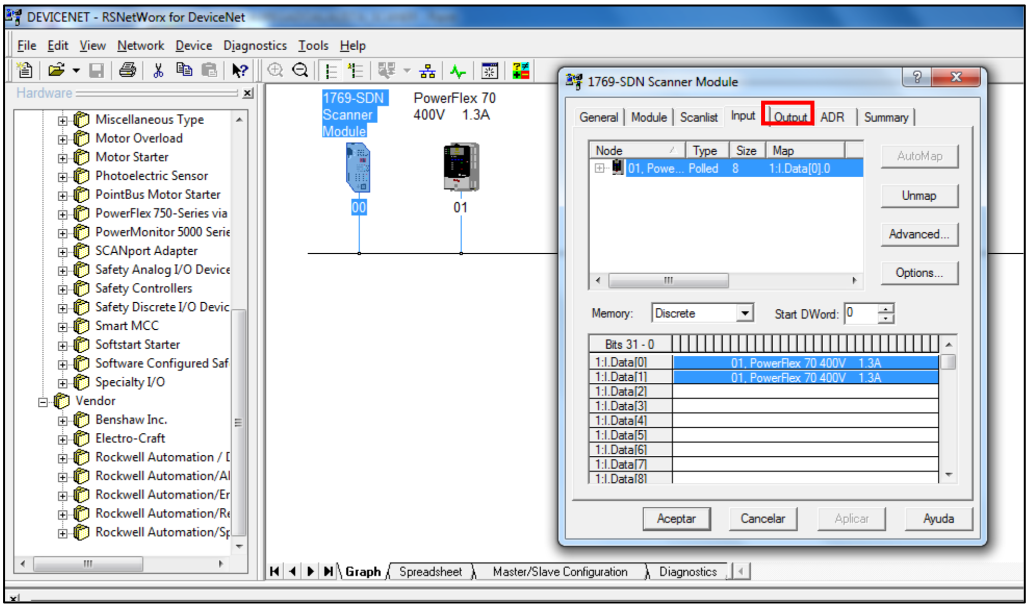
☑ Automap on Add

☑ Node Active

Electronic Key:
☑ Device Type
☑ Vendor
☑ Product Code
□ Major Revision
□ Minor □ or higher

Figura 41: Dispositivo PowerFlex 70 en la lista de escaneo.

(Fuente propia)



Node	Type	Size	Map
01, Powe...	Polled	8	1:I.Data[0].0

Memory: Discrete

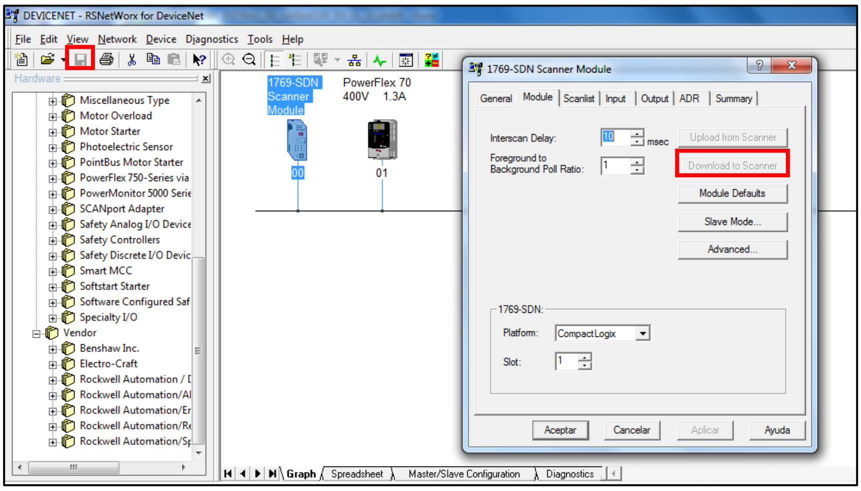
Start DWord: 0

Bits 31 - 0	
1:I.Data[0]	01, PowerFlex 70 400V 1.3A
1:I.Data[1]	01, PowerFlex 70 400V 1.3A
1:I.Data[2]	
1:I.Data[3]	
1:I.Data[4]	
1:I.Data[5]	
1:I.Data[6]	
1:I.Data[7]	
1:I.Data[8]	

Figura 42: Verificando las Entradas y Salidas en el SCANER 1769 –SDN

(Fuente propia)

- Verificado los parámetros y las entradas y salidas de nuestro variador en el SCANER 1769 –SDN, se procede a descargarlos para que queden grabados en la red creada y como paso final se guardara en un archivo.



1769-SDN Scanner Module

Interscan Delay: 10 msec

Foreground to Background Poll Ratio: 1

1769-SDN:

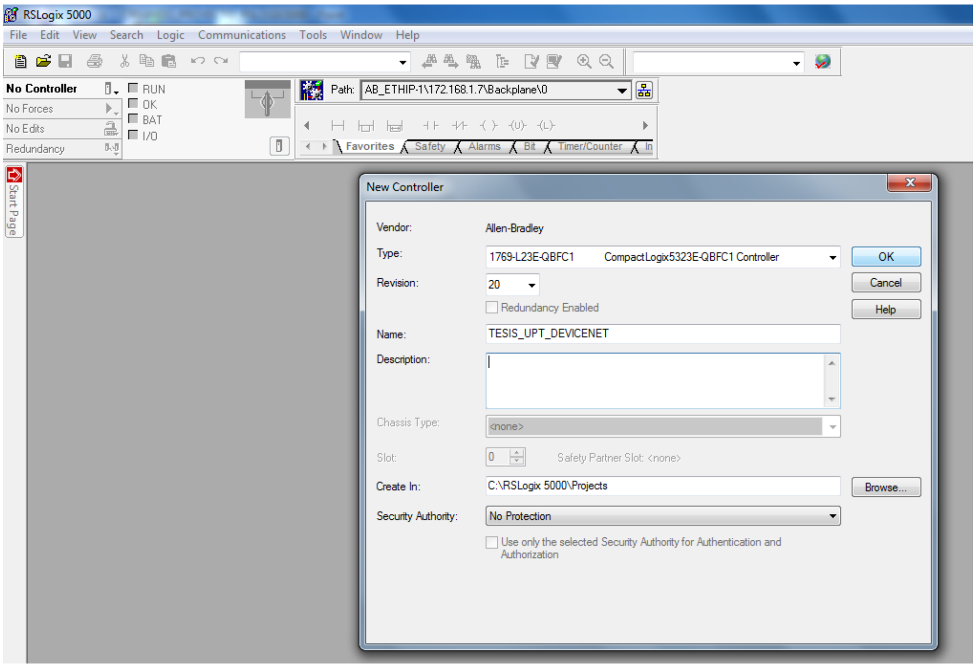
Platform: CompactLogix

Slot: 1

Figura 43: Descarga de parámetros y guardado del proyecto creado. (Fuente propia)

2.3.2.3 Configurando dispositivos de expansión en el RSLOGIX 5000

- El programa RSLogix 5000 nos permite realizar nuestro programa y rutinas para asignarle al PLC. El primer paso es elegir el modelo de PLC que tenemos, en nuestro caso el 1769-L23E-QBFC1B y le asignaremos la Revisión V20. (versión de software y firmware)



Path: AB_ETHIP-1\172.168.1.7\Backplane\0

New Controller

Vendor: Allen-Bradley

Type: 1769-L23E-QBFC1 CompactLogix5323E-QBFC1 Controller

Revision: 20

□ Redundancy Enabled

Name: TESIS_UPT_DEVICENET

Description:

Chassis Type: <none>

Slot: 0 Safety Partner Slot: <none>

Create In: C:\RSLogix 5000\Projects

Security Authority: No Protection

□ Use only the selected Security Authority for Authentication and Authorization

Figura 44: Selección del modelo del PLC y Nombre del Proyecto (Fuente propia)

- En la ventana del organizador controlador, vamos a la carpeta I/O Configuration, desplegamos y vamos a la carpeta Expansión I/O y agregamos Nuevo Módulo.

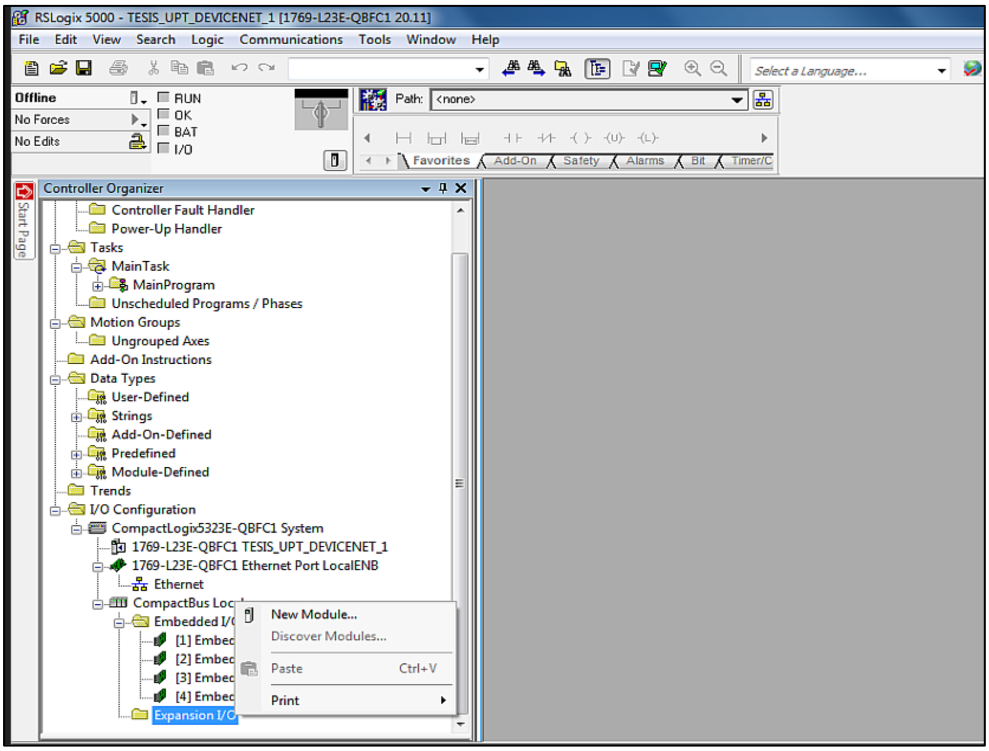
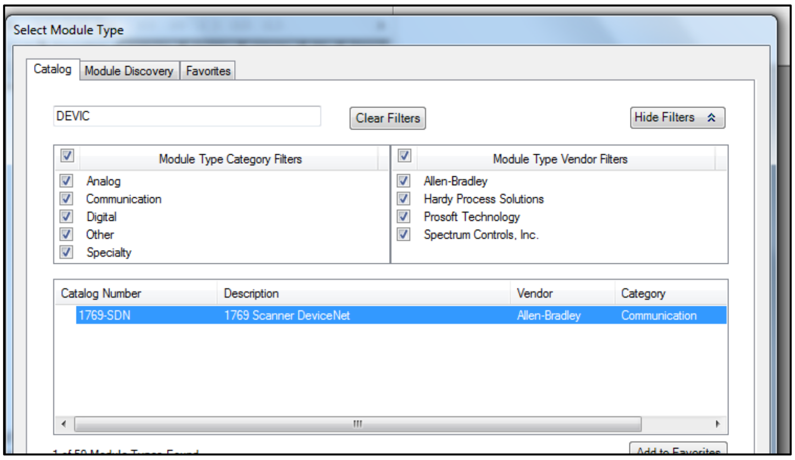


Figura 45: Agregando módulo SCANER 1769 –SDN.

(Fuente propia)

- En la ventana emergente tendremos que buscar el módulo SCANER 1769 –SDN, aceptamos y nos pedirá la revisión del módulo escáner, esta se encuentra en la etiqueta de información del equipo. En este caso es revisión 1, luego elegimos el nombre del módulo agregado.



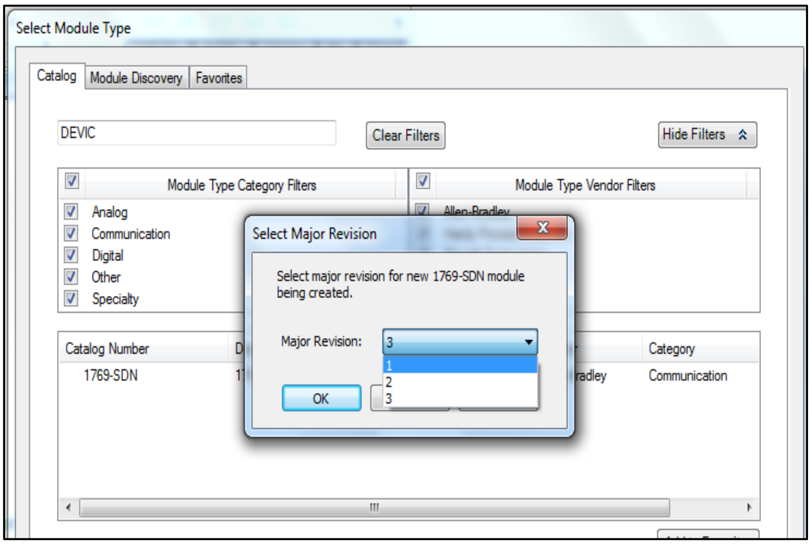
DEVIC

☑ Module Type Category Filters	☑ Module Type Vendor Filters
☑ Analog	☑ Allen-Bradley
☑ Communication	☑ Hardy Process Solutions
☑ Digital	☑ Prosoft Technology
☑ Other	☑ Spectrum Controls, Inc.
☑ Specialty	

Catalog Number	Description	Vendor	Category
1769-SDN	1769 Scanner DeviceNet	Allen-Bradley	Communication

Figura 46: Buscando en la Librería el Modulo SCANER 1769 –SDN

(Fuente propia)



Select Module Type

DEVIC

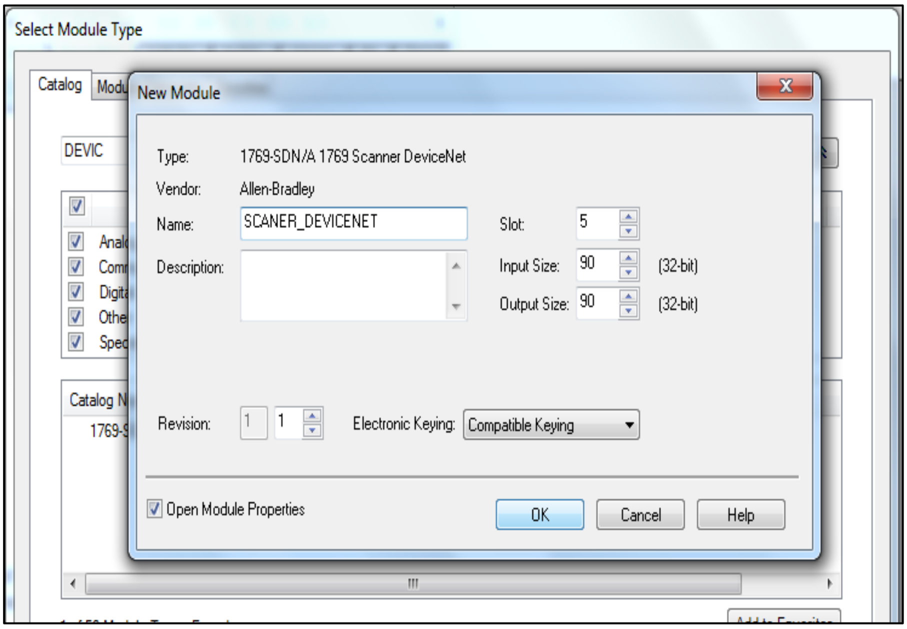
Select Major Revision

Select major revision for new 1769-SDN module being created.

Major Revision: 3

Catalog Number	Category
1769-SDN	Communication

Figura 47: Selección de la Revisión del módulo SCANER 1769 –SDN (Fuente propia)



Select Module Type

New Module

Type: 1769-SDN/A 1769 Scanner DeviceNet

Vendor: Allen-Bradley

Name: SCANER_DEVICENET

Description:

Slot: 5

Input Size: 90 (32-bit)

Output Size: 90 (32-bit)

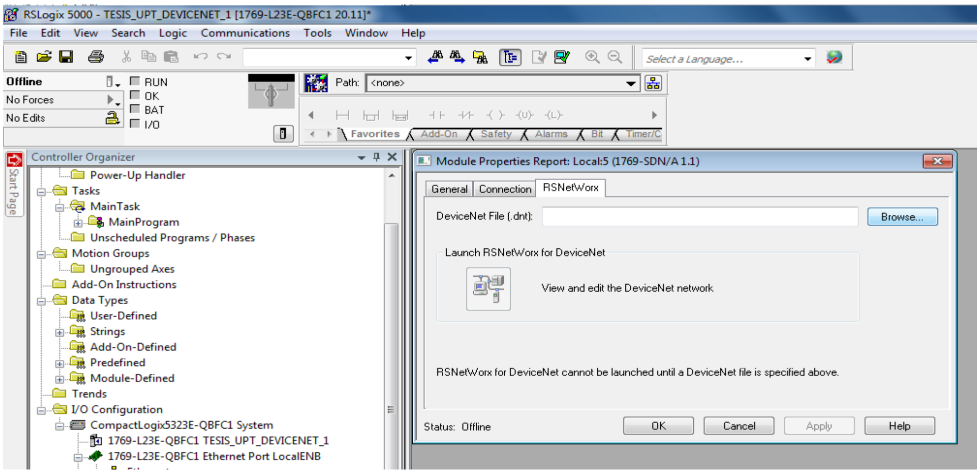
Revision: 1 1

Electronic Keying: Compatible Keying

☑ Open Module Properties

Figura 48: estableciendo el Nombre del módulo SCANER 1769 –SDN (Fuente propia)

- Como último paso tenemos que habilitar los parámetros que se guardaron en la red DeviceNet, buscamos el archivo guardado de la plataforma RSNETWORX.



RSLogix 5000 - TESIS_UPT_DEVICENET_1 [1769-L23E-QBFC1 20.11]*

Module Properties Report: Local:5 (1769-SDN/A 1.1)

DeviceNet File (.dnt):

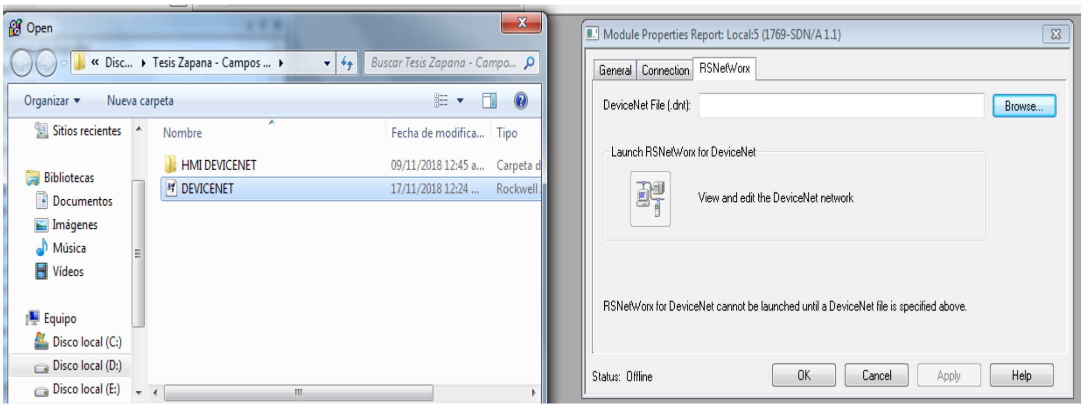
Launch RSNetWorx for DeviceNet

View and edit the DeviceNet network

RSNetWorx for DeviceNet cannot be launched until a DeviceNet file is specified above.

Status: Offline

Figura 49: búsqueda del archivo generado en la plataforma RSNETWORX (Fuente propia)



Nombre	Fecha de modifica...	Tipo
HMI DEVICENET	09/11/2018 12:45 a...	Carpeta d
DEVICENET	17/11/2018 12:24 ...	Rockwell

Module Properties Report: Local:5 (1769-SDN/A 1.1)

DeviceNet File (.dnt):

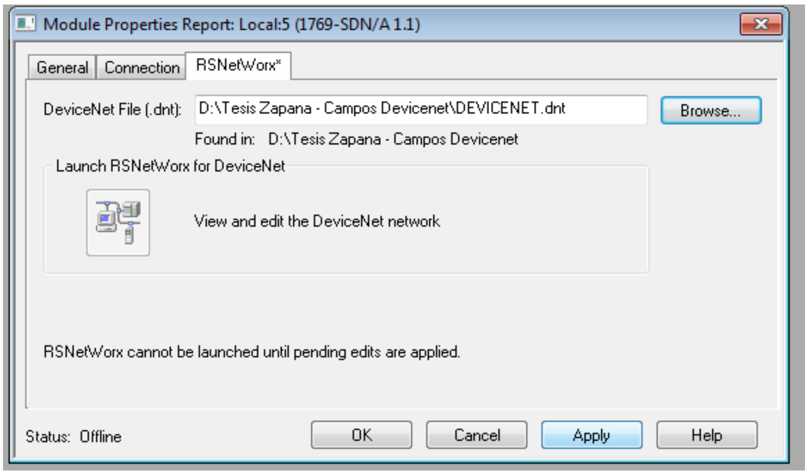
Launch RSNetWorx for DeviceNet

View and edit the DeviceNet network

RSNetWorx for DeviceNet cannot be launched until a DeviceNet file is specified above.

Status: Offline

Figura 50: Selección del Archivo generado en RSNETWORX (Fuente propia)



Module Properties Report: Local:5 (1769-SDN/A 1.1)

DeviceNet File (.dnt): D:\Tesis Zapana - Campos Devicenet\DEVICENET.dnt

Found in: D:\Tesis Zapana - Campos Devicenet

Launch RSNetWorx for DeviceNet

View and edit the DeviceNet network

RSNetWorx cannot be launched until pending edits are applied.

Status: Offline

Figura 51: aplicamos los cambios realizados y aceptamos. (Fuente propia)

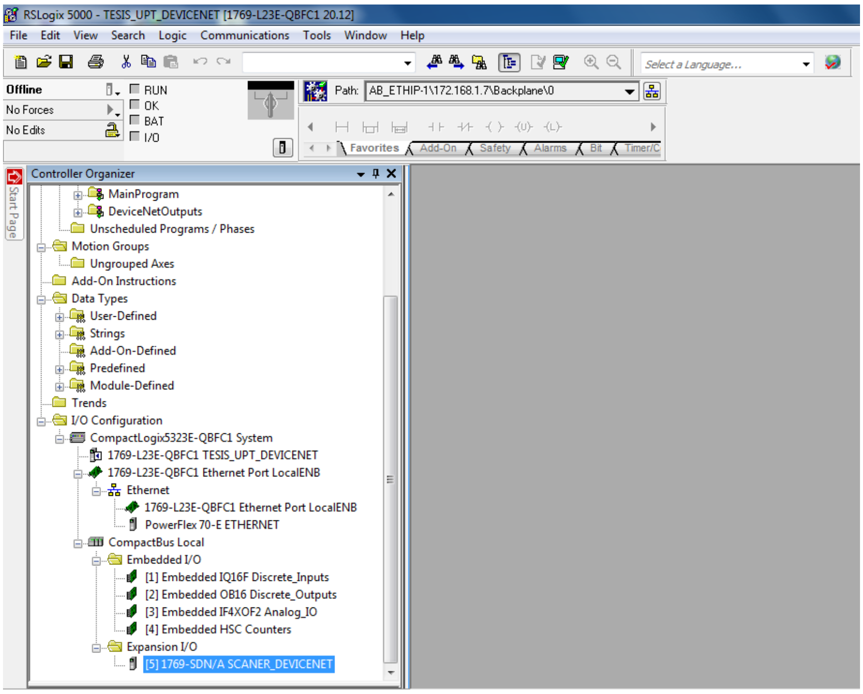
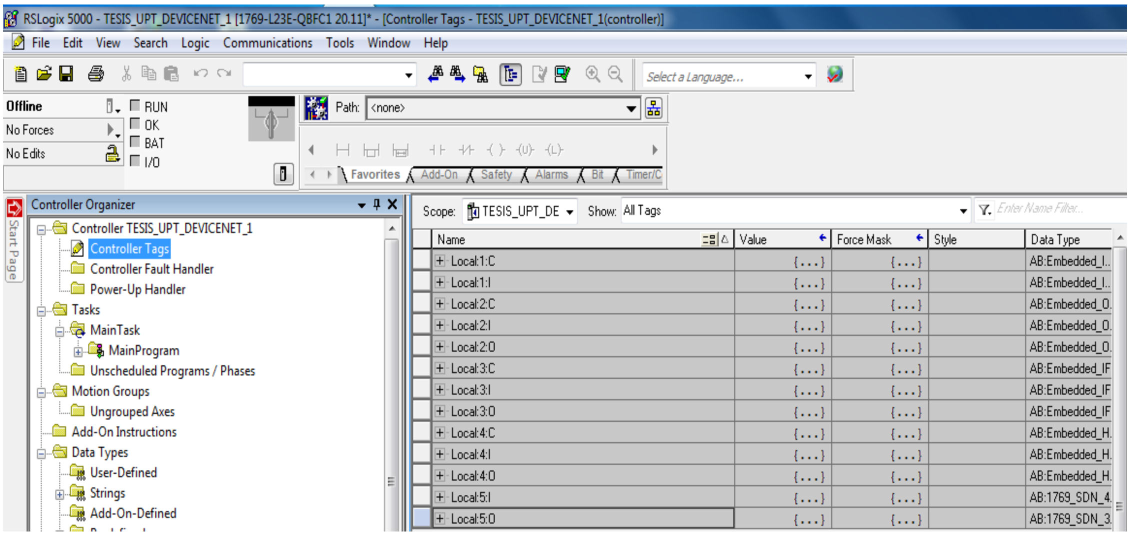

RSLogix 5000 - TESIS_UPT_DEVICENET [1769-L23E-QBFC1 20.12]
File Edit View Search Logic Communications Tools Window Help
Select a Language...
Offline
No Forces
No Edits
RUN
OK
BAT
I/O
Path: AB_ETHIP-1\172.168.1.7\Backplane\0
Favorites Add-On Safety Alarms Bit Timer/C
Start Page
Controller Organizer
MainProgram
DeviceNetOutputs
Unscheduled Programs / Phases
Motion Groups
Ungrouped Axes
Add-On Instructions
Data Types
User-Defined
Strings
Add-On-Defined
Predefined
Module-Defined
Trends
I/O Configuration
CompactLogix5323E-QBFC1 System
1769-L23E-QBFC1 TESIS_UPT_DEVICENET
1769-L23E-QBFC1 Ethernet Port LocalENB
Ethernet
1769-L23E-QBFC1 Ethernet Port LocalENB
PowerFlex 70-E ETHERNET
CompactBus Local
Embedded I/O
[1] Embedded IQ16F Discrete_Inputs
[2] Embedded OB16 Discrete_Outputs
[3] Embedded IF4XOF2 Analog_IO
[4] Embedded HSC Counters
Expansion I/O
[5] 1769-SDN/A SCANER_DEVICENET

Figura 52: Módulo agregado con las entradas y salidas declaradas de la Red creada en Devicenet.

(Fuente propia)

- Agregado el modulo del scanner con las entradas y salidas ya declaradas del variador, se verifica en el control de Tags las direcciones para la programación.


RSLogix 5000 - TESIS_UPT_DEVICENET_1 [1769-L23E-QBFC1 20.11]* - [Controller Tags - TESIS_UPT_DEVICENET_1(controller)]
File Edit View Search Logic Communications Tools Window Help
Select a Language...
Offline
No Forces
No Edits
RUN
OK
BAT
I/O
Path: <none>
Favorites Add-On Safety Alarms Bit Timer/C
Start Page
Controller Organizer
Controller TESIS_UPT_DEVICENET_1
Controller Tags
Controller Fault Handler
Power-Up Handler
Tasks
MainTask
MainProgram
Unscheduled Programs / Phases
Motion Groups
Ungrouped Axes
Add-On Instructions
Data Types
User-Defined
Strings
Add-On-Defined

Scope: TESIS_UPT_DE Show: All Tags

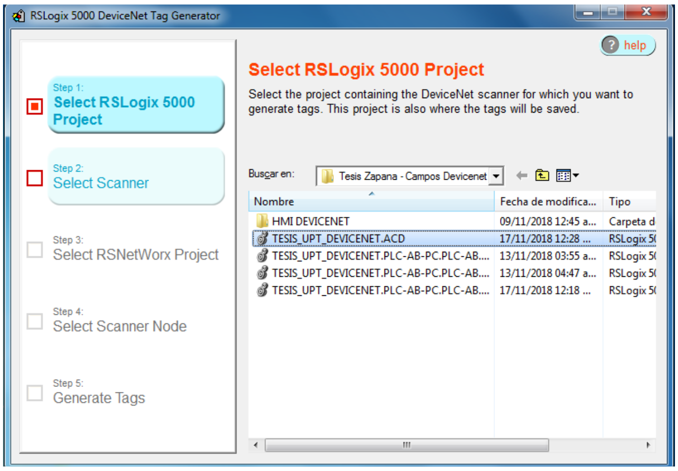
Name	Value	Force Mask	Style	Data Type
Local:1:C	{...}	{...}		AB:Embedded_I..
Local:1:I	{...}	{...}		AB:Embedded_I..
Local:2:C	{...}	{...}		AB:Embedded_O.
Local:2:I	{...}	{...}		AB:Embedded_O.
Local:2:O	{...}	{...}		AB:Embedded_O.
Local:3:C	{...}	{...}		AB:Embedded_IF
Local:3:I	{...}	{...}		AB:Embedded_IF
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 53: Entradas y Salidas del Variado Powerflex 70

(Fuente propia)

2.3.2.4 Configurando Plataforma DEVICETAG Generator de RSLOGIX 5000

Hasta ahora podemos leer y escribir para acceder a nuestros dispositivos, pero tener los datos en lenguaje nativo, no es la mejor manera de tratar con ellos. Para esto Rockwell tiene disponible una herramienta llamada DeviceNet Tag Generator disponible, esta plataforma traduce el lenguaje nativo y los agrupa en un leguaje más amigable al usuario, sus pasos se describen a continuación.



RSLogix 5000 DeviceNet Tag Generator

☒ Step 1: Select RSLogix 5000 Project

□ Step 2: Select Scanner

□ Step 3: Select RSNetWorx Project

□ Step 4: Select Scanner Node

□ Step 5: Generate Tags

Select RSLogix 5000 Project

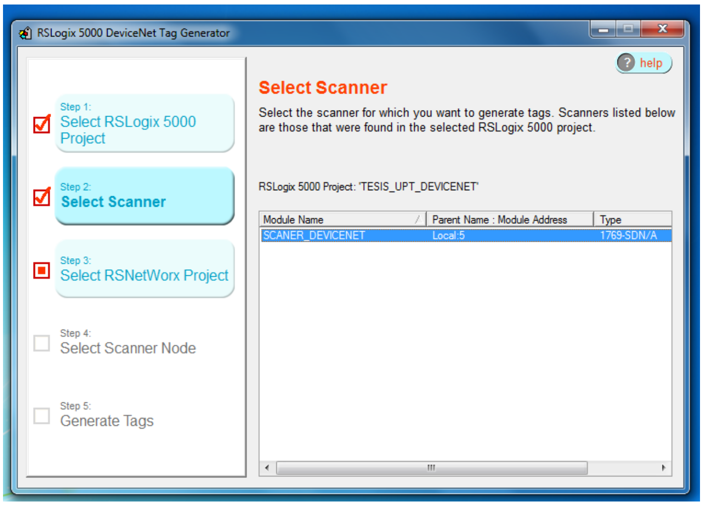
Select the project containing the DeviceNet scanner for which you want to generate tags. This project is also where the tags will be saved.

Buscar en: Tesis Zapana - Campos Devicenet

Nombre	Fecha de modifica...	Tipo
HMI DEVICENET	09/11/2018 12:45 a...	Carpeta d
TESIS_UPT_DEVICENET.ACD	17/11/2018 12:28 ...	RSLogix 5(
TESIS_UPT_DEVICENET.PLC-AB-PC.PLC-AB....	13/11/2018 03:55 a...	RSLogix 5(
TESIS_UPT_DEVICENET.PLC-AB-PC.PLC-AB....	13/11/2018 04:47 a...	RSLogix 5(
TESIS_UPT_DEVICENET.PLC-AB-PC.PLC-AB....	17/11/2018 12:18 ...	RSLogix 5(

Figura 54: Selección del proyecto con scanner Devicenet para generar los Tags.

(Fuente propia)



RSLogix 5000 DeviceNet Tag Generator

☑ Step 1: Select RSLogix 5000 Project

☑ Step 2: Select Scanner

☒ Step 3: Select RSNetWorx Project

□ Step 4: Select Scanner Node

□ Step 5: Generate Tags

Select Scanner

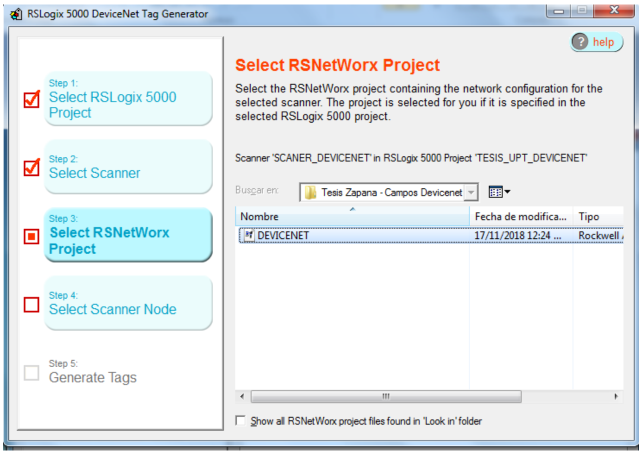
Select the scanner for which you want to generate tags. Scanners listed below are those that were found in the selected RSLogix 5000 project.

RSLogix 5000 Project: 'TESIS_UPT_DEVICENET'

Module Name	Parent Name : Module Address	Type
SCANER_DEVICENET	Local:5	1769-SDN/A

Figura 55: Selección del Scanner DeviceNet.

(Fuente propia)



Select RSNetWorx Project

Select the RSNetWorx project containing the network configuration for the selected scanner. The project is selected for you if it is specified in the selected RSLogix 5000 project.

Scanner 'SCANER_DEVICENET' in RSLogix 5000 Project 'TESIS_UPT_DEVICENET'

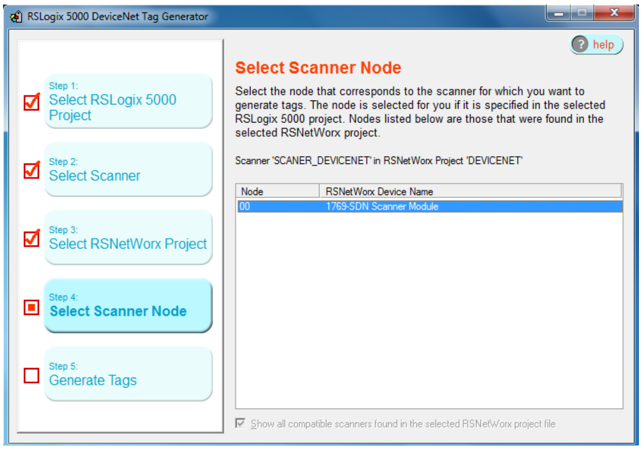
Buscar en: Tesis Zapana - Campos Devicenet

Nombre	Fecha de modifica...	Tipo
DEVICENET	17/11/2018 12:24 ...	Rockwell

□ Show all RSNetWorx project files found in 'Look in' folder

Figura 56: Selección del archivo DeviceNet creado en RsnetWorx.

(Fuente propia)



Select Scanner Node

Select the node that corresponds to the scanner for which you want to generate tags. The node is selected for you if it is specified in the selected RSLogix 5000 project. Nodes listed below are those that were found in the selected RSNetWorx project.

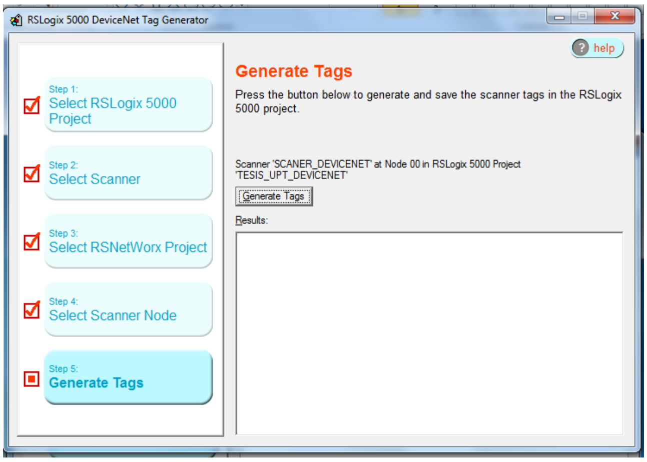
Scanner 'SCANER_DEVICENET' in RSNetWorx Project 'DEVICENET'

Node	RSNetWorx Device Name
00	1769-SDN Scanner Module

☑ Show all compatible scanners found in the selected RSNetWorx project file

Figura 57: Selección del nodo del Scanner.

(Fuente propia)



Generate Tags

Press the button below to generate and save the scanner tags in the RSLogix 5000 project.

Scanner 'SCANER_DEVICENET' at Node 00 in RSLogix 5000 Project 'TESIS_UPT_DEVICENET'

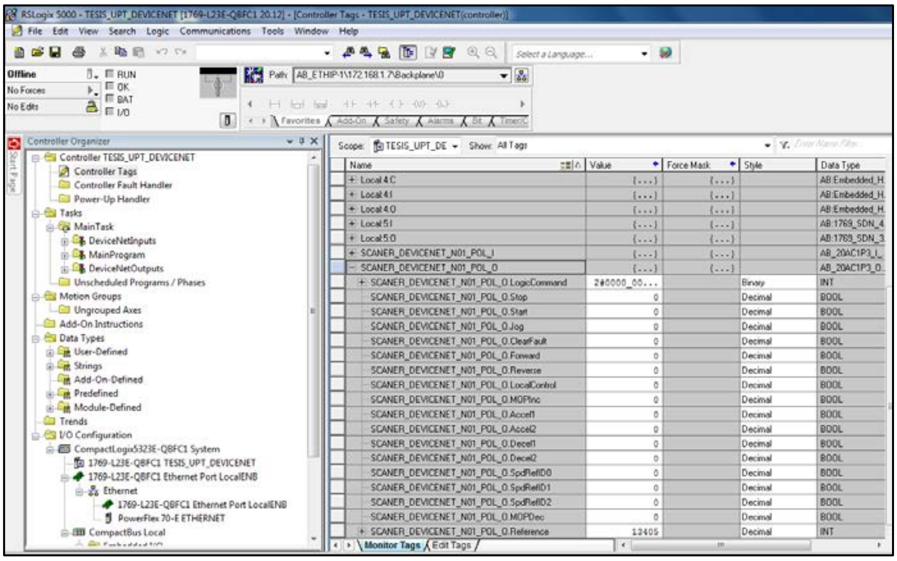
Generate Tags

Results:

Figura 58: Finalmente aplicamos en Generate Tags.

(Fuente propia)

2.3.2.5 Creación y configuración del programa y tags en la RSLOGIX 5000

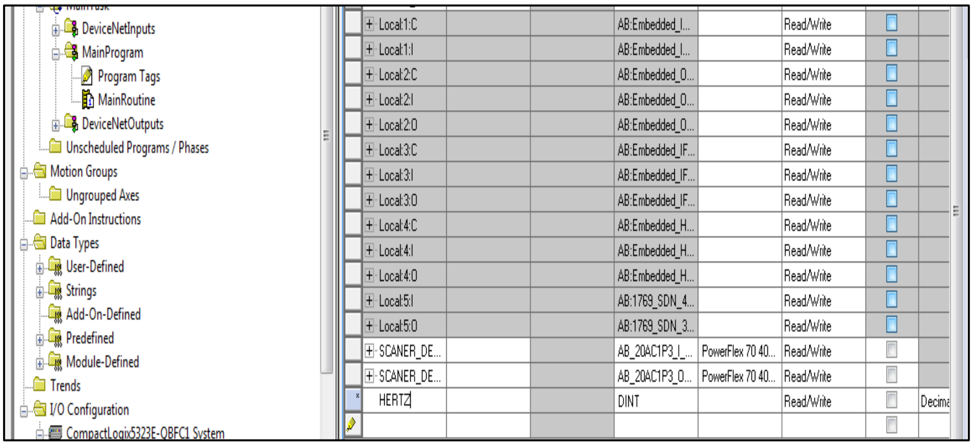

RSLogix 5000 - TESIS_UPT_DEVICENET [1769-L23E-QBFC1 20.12] - [Controller Tags - TESIS_UPT_DEVICENET(controller)]
File Edit View Search Logic Communications Tools Window Help
Offline
No Forces
No Edits
RUN
OK
BAT
I/O
Path: AB_ETHIP-1\172.168.1.7\Backplane\0
Favorites Add-On Safety Alarms Bit Timer/C
Controller Organizer
Controller TESIS_UPT_DEVICENET
Controller Tags
Controller Fault Handler
Power-Up Handler
Tasks
MainTask
DeviceNetInputs
MainProgram
DeviceNetOutputs
Unscheduled Programs / Phases
Motion Groups
Ungrouped Axes
Add-On Instructions
Data Types
User-Defined
Strings
Add-On-Defined
Predefined
Module-Defined
Trends
I/O Configuration
CompactLogix5323E-QBFC1 System
1769-L23E-QBFC1 TESIS_UPT_DEVICENET
1769-L23E-QBFC1 Ethernet Port LocalENB
Ethernet
1769-L23E-QBFC1 Ethernet Port LocalENB
PowerFlex 70-E ETHERNET
CompactBus Local
Scope: TESIS_UPT_DE
Show: All Tags
Monitor Tags
Edit Tags

Name	Value	Force Mask	Style	Data Type
+ 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
+ SCANER_DEVICENET_N01_POL_I	{...}	{...}		AB_20AC1P3_I_
− SCANER_DEVICENET_N01_POL_O	{...}	{...}		AB_20AC1P3_O.
+ SCANER_DEVICENET_N01_POL_O.LogicCommand	2#0000_00...		Binary	INT
SCANER_DEVICENET_N01_POL_O.Stop	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Start	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Jog	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.ClearFault	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Forward	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Reverse	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.LocalControl	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.MOPInc	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Accel1	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Accel2	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Decel1	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Decel2	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.SpdRefID0	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.SpdRefID1	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.SpdRefID2	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.MOPDec	0		Decimal	BOOL
+ SCANER_DEVICENET_N01_POL_O.Reference	12405		Decimal	INT

Figura 59: Etiquetas generados por Rslogix 5000 Tags Generatos Devicenet

(Fuente propia)

- Establecido los tags o etiquetas, se procede a crear el Tag HERTZ, con el cual nos permitirá hacer el escalamiento para nuestro Programa y Lógica de control.



+ Local:1:C			AB:Embedded_I...		Read/Write	□	
+ Local:1:I			AB:Embedded_I...		Read/Write	□	
+ Local:2:C			AB:Embedded_O...		Read/Write	□	
+ Local:2:I			AB:Embedded_O...		Read/Write	□	
+ Local:2:O			AB:Embedded_O...		Read/Write	□	
+ Local:3:C			AB:Embedded_IF...		Read/Write	□	
+ Local:3:I			AB:Embedded_IF...		Read/Write	□	
+ Local:3:O			AB:Embedded_IF...		Read/Write	□	
+ Local:4:C			AB:Embedded_H...		Read/Write	□	
+ Local:4:I			AB:Embedded_H...		Read/Write	□	
+ Local:4:O			AB:Embedded_H...		Read/Write	□	
+ Local:5:I			AB:1769_SDN_4...		Read/Write	□	
+ Local:5:O			AB:1769_SDN_3...		Read/Write	□	
+ SCANER_DE...			AB_20AC1P3_I_...	PowerFlex 70 40...	Read/Write	□	
+ SCANER_DE...			AB_20AC1P3_O...	PowerFlex 70 40...	Read/Write	□	
HERTZ			DINT		Read/Write	□	Decima
						□	

Figura 60: Creación de la etiqueta HERTZ.

(Fuente propia)

- La lógica que utilizaremos está basada en la ecuación de la pendiente, este nos permite un escalamiento.

$$y = mx + b$$

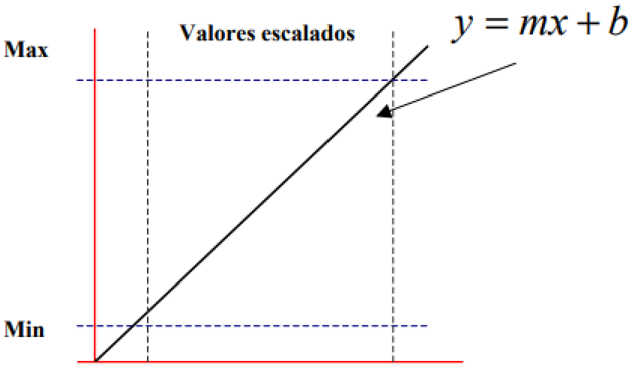


Figura 61: Valores escalados

(Fuente propia)

Valor escalado significara adecuar la variable a medir a su lectura máxima y mínima en el proceso con la salida o entrada análoga y su número de bits respectivos.

Donde:

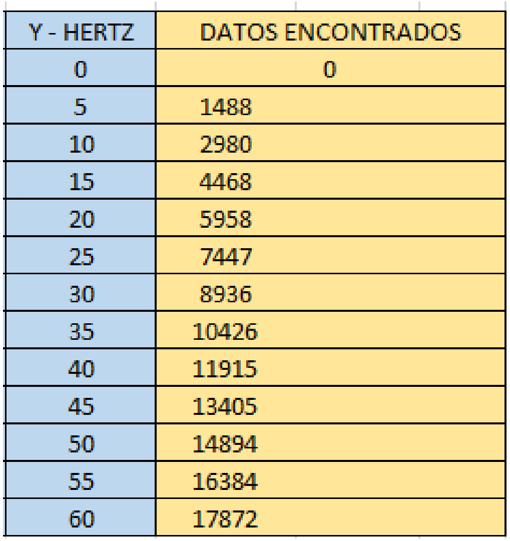
y = Salida escalada (0-60 Hz)

m = Pendiente (Rate)

x = Valor análogo de entrada (valores obtenidos del Variador)

b = Offset

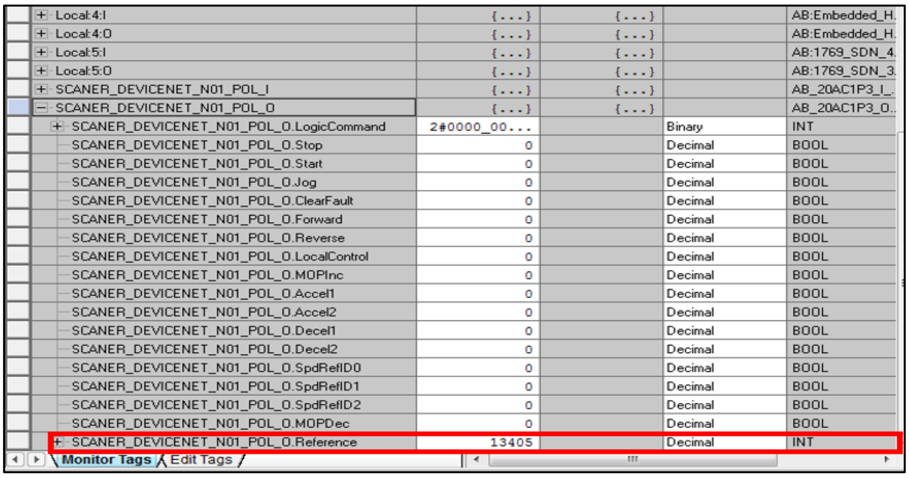
Obtennos los valores análogos de entrada de los datos del variador:



Y - HERTZ	DATOS ENCONTRADOS
0	0
5	1488
10	2980
15	4468
20	5958
25	7447
30	8936
35	10426
40	11915
45	13405
50	14894
55	16384
60	17872

Figura 62: Valores obtenidos manualmente del variador

(Fuente propia)



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.
SCANER_DEVICENET_N01_POL_I	{...}	{...}		AB_20AC1P3_I_.
SCANER_DEVICENET_N01_POL_O	{...}	{...}		AB_20AC1P3_O..
SCANER_DEVICENET_N01_POL_O.LogicCommand	2#0000_00...		Binary	INT
SCANER_DEVICENET_N01_POL_O.Stop	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Start	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Jog	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.ClearFault	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Forward	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Reverse	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.LocalControl	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.MOPInc	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Accel1	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Accel2	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Decel1	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Decel2	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.SpdRefID0	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.SpdRefID1	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.SpdRefID2	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.MOPDec	0		Decimal	BOOL
SCANER_DEVICENET_N01_POL_O.Reference	13405		Decimal	INT

Figura 63: Etiqueta de referencia de velociadad del variador

(Fuente propia)

Habiendo hallado los valores minimos y maximos de Y y X, Y los valores de RATE (pendiente) y OFFSET se calculan de la siguiente manera:

Y_2 (Escalado Max) = 60

Y_1 (Escalado Min) = 0

X_2 (Entrada Max) =17872

X_1 (Entrada Max) = 0

Pendiente $$RATE = \frac{EscaladoMax - EscaladoMin}{InputMax - InputMin}$$

$$Pendiente\ (RATE) = \frac{Y_2 - Y_1}{X_2 - X_1}$$

$$Pendiente\ (RATE) = \frac{60 - 0}{17872 - 0}$$

$$Pendiente\ (RATE) = 0.003357$$

$$OFFSET = EscaladoMin - (InputMin * Rate)$$

$$OFFSET = ESCALADO\ Min - (INPUT\ Min * RATE)$$

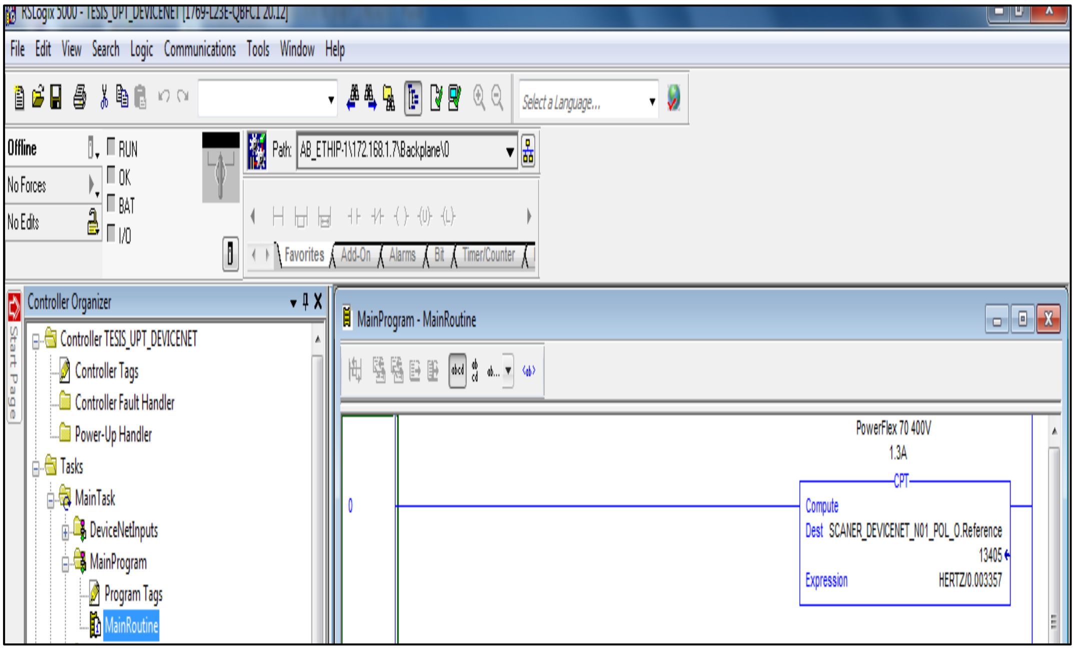
$$OFFSET = 0 - (0 * 0.003357)$$

$$OFFSET = 0$$

- En la capeta de desarrollo del programa Main Routine, utilizará la instrucción CPT, La instrucción CPT, es una instrucción que permite fórmulas matemáticas de hasta 255 caracteres complejas o sencillas en una sola instrucción. Cuando está habilitada, la instrucción CPT evalúa la expresión y pone el resultado en Dest.
  Expresión:

$$Expresion = \frac{HERTZ\ (TAG\ CREADO)}{RATE}$$

$$DEST. = \ EXPRESION$$



Path: AB_ETHIP-1\172.168.1.7\Backplane\0

Compute
Dest SCANER_DEVICENET_N01_POL_O.Reference
13405
Expression HERTZ/0.003357

Figura 64: Instrucción CPT.

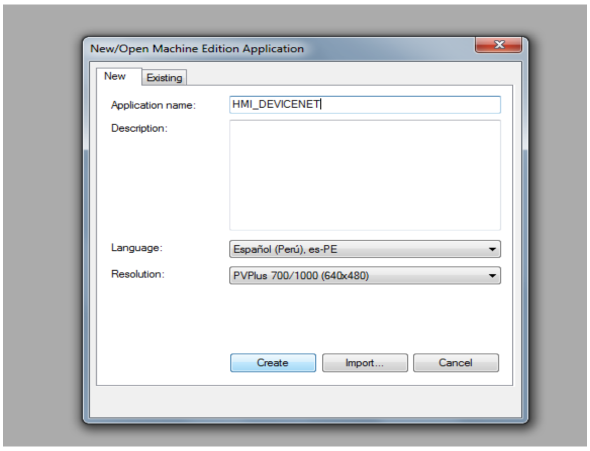
(Fuente propia)

2.3.2.6 Creando HMI en la plataforma FACTORYTALK VIEW STUDIO

El software FactoryTalk View es una aplicación de HMI versátil que proporciona una solución dedicada y potente para dispositivos de interfaz de operador a nivel de máquina. Como un elemento integral de la solución de visualización de Rockwell Automation, FactoryTalk View Machine Edition ofrece gráficos superiores, gestión de usuarios en tiempo de ejecución, cambio de idioma y un tiempo de puesta en servicio más rápido a través de un entorno de desarrollo común.

FactoryTalk View Machine Edition permite una interfaz de operador consistente en múltiples plataformas, en esta plataforma crearemos un HMI práctico para el fácil uso de personas.

- Creamos proyecto, nombramos el proyecto HMI_DEVICENET.



Application name: HMI_DEVICENET

Description:

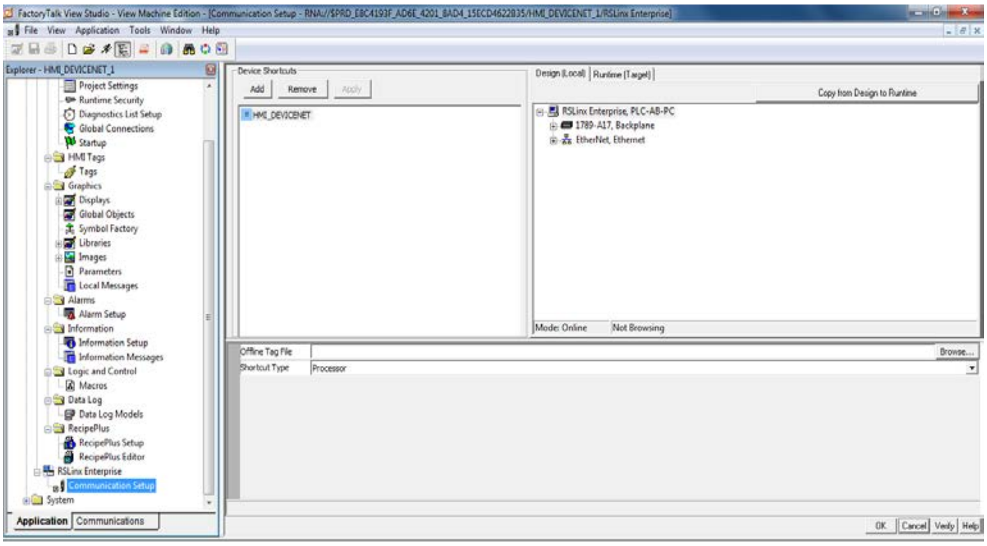
Language: Español (Perú), es-PE

Resolution: PVPlus 700/1000 (640x480)

Figura 65: Creación del proyecto en FactoryTalkView

(Fuente propia)

- En la ventana Explorer del proyecto, nos dirigimos a sección RSlinx Enterprise, como se aprecia en la figura. Aplicamos en Comunicatión Setup, aquí establecemos la comunicación del controlador y la plataforma FactoryTalkView, para poder obtener toda la data y etiquetas del RSlogix 5000.



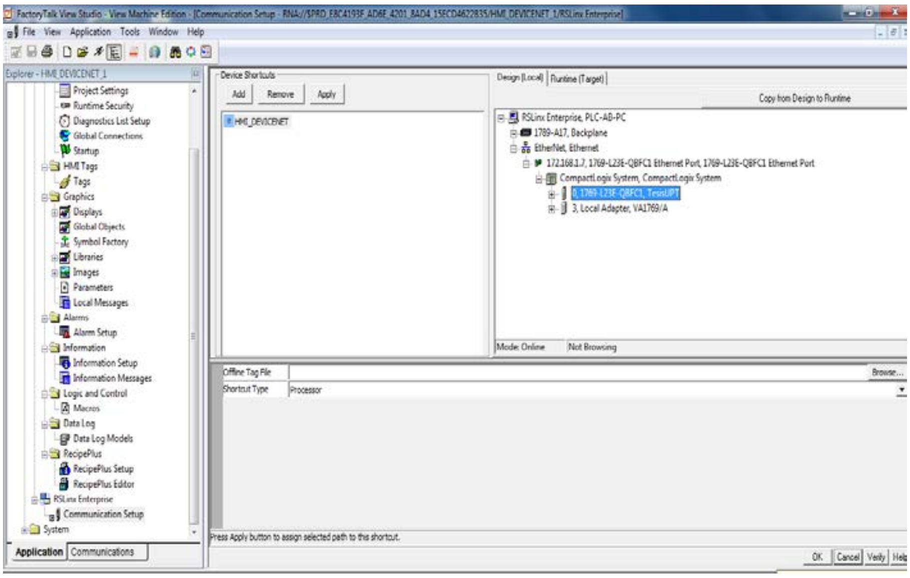
Mode: Online   Not Browsing

Offline Tag File

Shortcut Type: Processor

Figura 66: Estableciendo la comunicación con el PLC.

(Fuente propia)



HMI_DEVICENET

RSLinx Enterprise, PLC-AB-PC
1789-A17, Backplane
EtherNet, Ethernet
172.168.1.7, 1769-L23E-QBFC1 Ethernet Port, 1769-L23E-QBFC1 Ethernet Port
CompactLogix System, CompactLogix System
0, 1769-L23E-QBFC1, TesisUPT
3, Local Adapter, VA1769/A

Mode: Online   Not Browsing

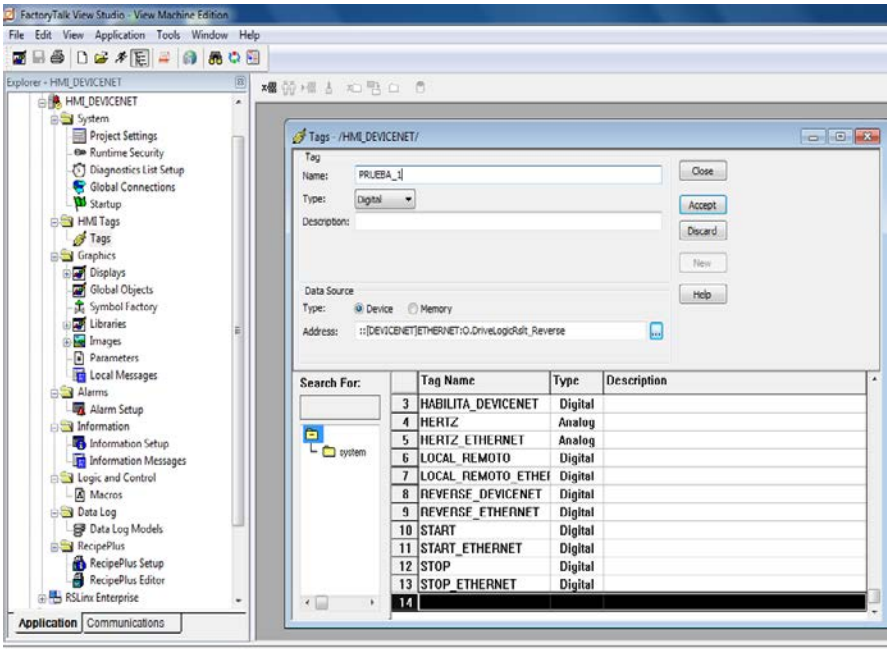
Offline Tag File:
Shortcut Type: Processor

Press Apply button to assign selected path to this shortcut.

Figura 67: Comunicación del controlador con la plataforma FactoryTalkView

(Fuente propia)

- En la carpeta HMI TAGS, empezamos a crear los Tags para crear nuestras plantillas del HMI. Y procedemos a direccionar los objetos con las etiquetas en RSlogix5000



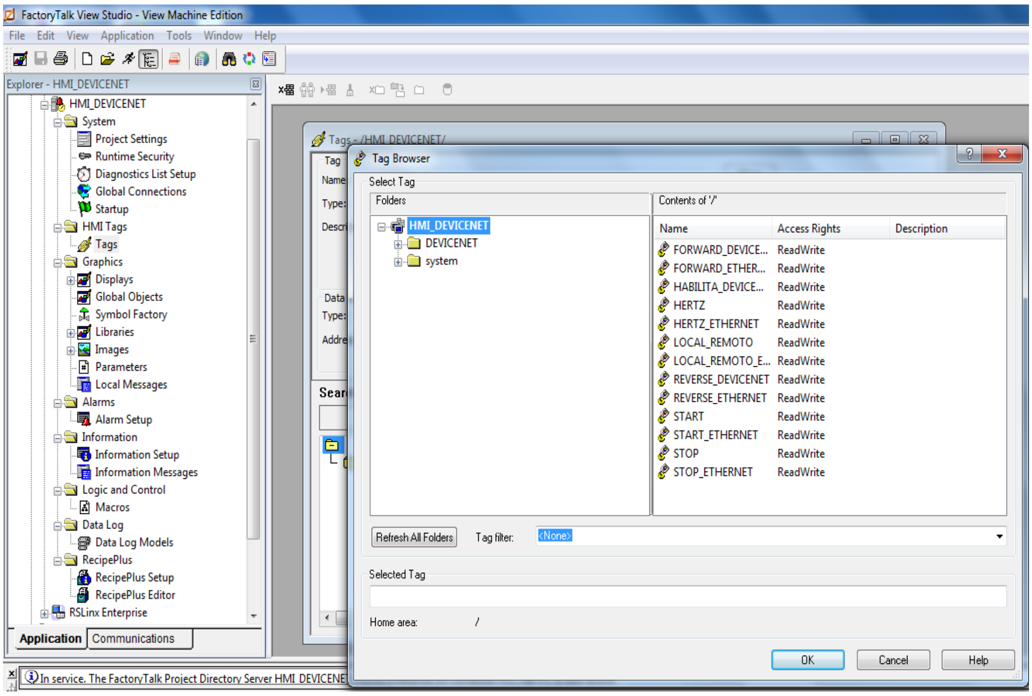
Tag
Name: PRUEBA_1
Type: Digital
Description:

Data Source
Type: ● Device ○ Memory
Address: ::[DEVICENET]ETHERNET:O.DriveLogicRslt_Reverse

	Tag Name	Type	Description
3	HABILITA_DEVICENET	Digital	
4	HERTZ	Analog	
5	HERTZ_ETHERNET	Analog	
6	LOCAL_REMOTO	Digital	
7	LOCAL_REMOTO_ETHEI	Digital	
8	REVERSE_DEVICENET	Digital	
9	REVERSE_ETHERNET	Digital	
10	START	Digital	
11	START_ETHERNET	Digital	
12	STOP	Digital	
13	STOP_ETHERNET	Digital	
14			

Figura 68: Creación de Tags

(Fuente propia)



Contents of '/'

Name	Access Rights	Description
FORWARD_DEVICE...	ReadWrite	
FORWARD_ETHER...	ReadWrite	
HABILITA_DEVICE...	ReadWrite	
HERTZ	ReadWrite	
HERTZ_ETHERNET	ReadWrite	
LOCAL_REMOTO	ReadWrite	
LOCAL_REMOTO_E...	ReadWrite	
REVERSE_DEVICENET	ReadWrite	
REVERSE_ETHERNET	ReadWrite	
START	ReadWrite	
START_ETHERNET	ReadWrite	
STOP	ReadWrite	
STOP_ETHERNET	ReadWrite	

Tag filter: <None>

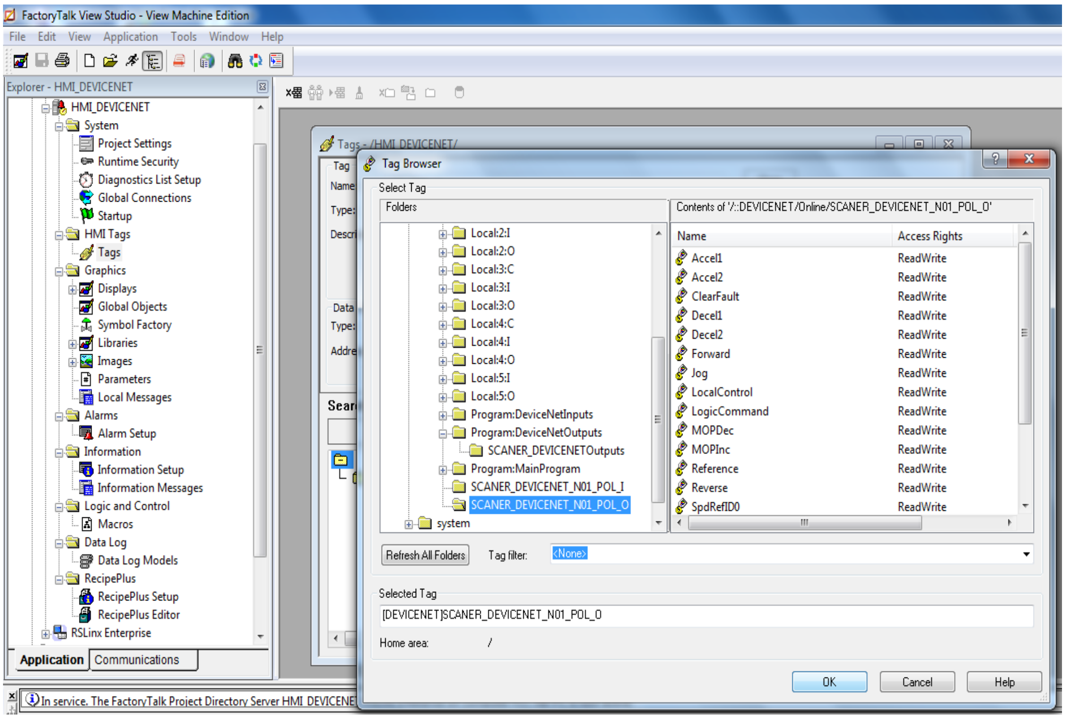
Selected Tag

Home area: /

In service. The FactoryTalk Project Directory Server HMI DEVICENE

Figura 69: Direccionamiento de los tags o etiquetas.

(Fuente propia)



Contents of '/::DEVICENET/Online/SCANER_DEVICENET_N01_POL_O'

Name	Access Rights
Accel1	ReadWrite
Accel2	ReadWrite
ClearFault	ReadWrite
Decel1	ReadWrite
Decel2	ReadWrite
Forward	ReadWrite
Jog	ReadWrite
LocalControl	ReadWrite
LogicCommand	ReadWrite
MOPDec	ReadWrite
MOPInc	ReadWrite
Reference	ReadWrite
Reverse	ReadWrite
SpdRefID0	ReadWrite

Tag filter: <None>

Selected Tag

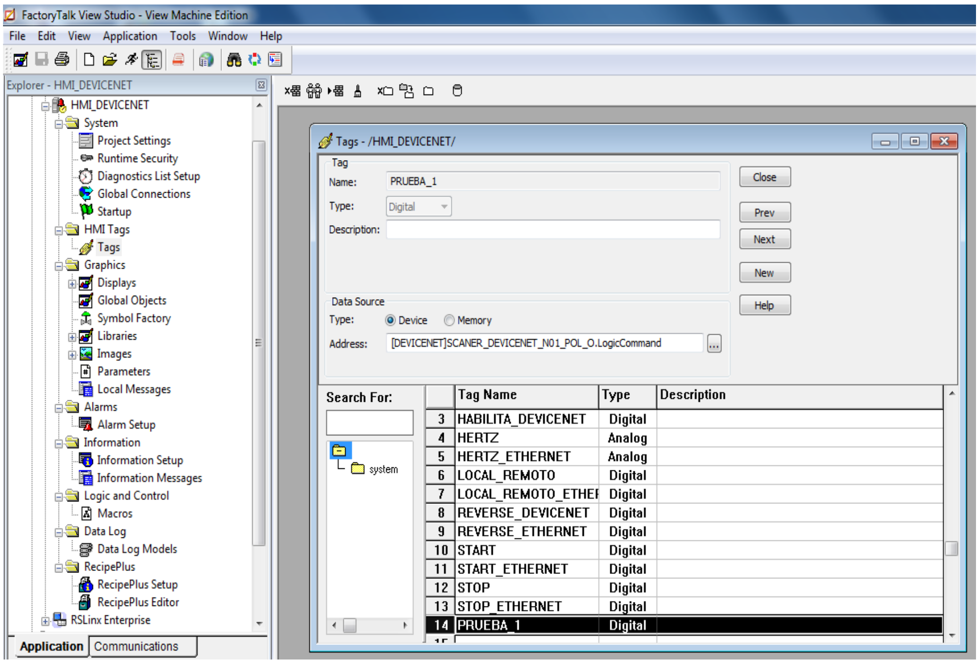
[DEVICENET]SCANER_DEVICENET_N01_POL_O

Home area: /

In service. The FactoryTalk Project Directory Server HMI DEVICENE

Figura 70: Asignando una dirección a un tag.

(Fuente propia)



Tag

Name: PRUEBA_1

Type: Digital

Description:

Data Source

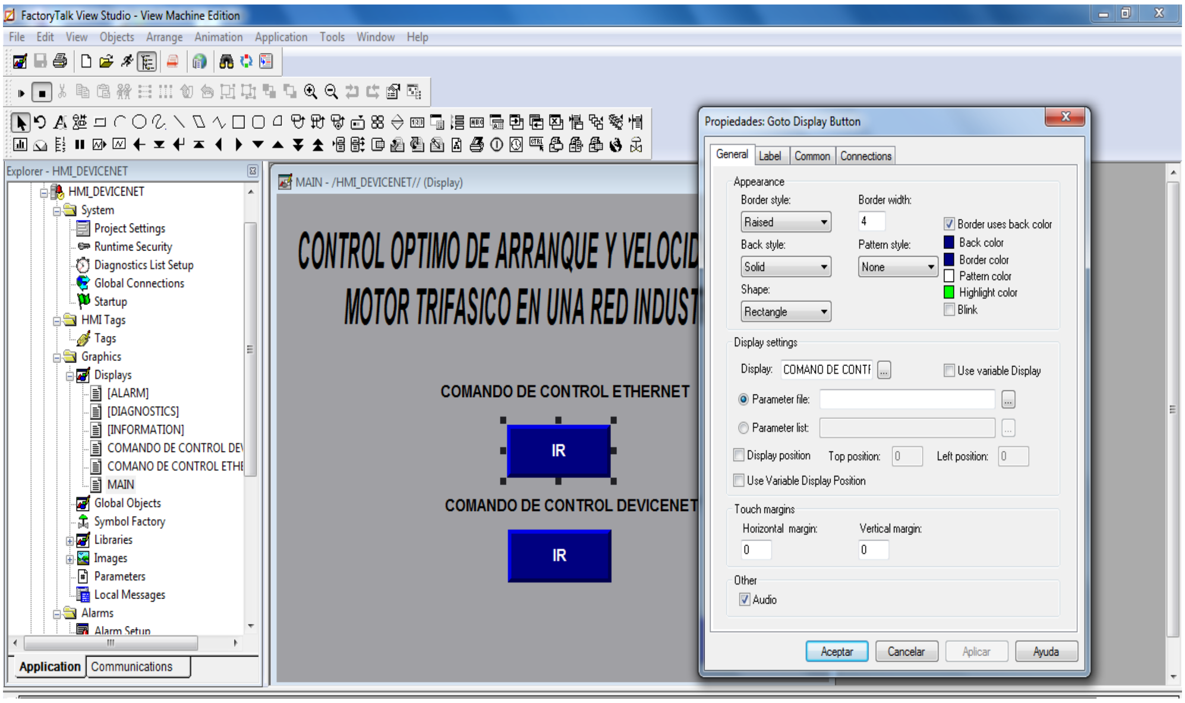
Type: ● Device ○ Memory

Address: [DEVICENET]SCANER_DEVICENET_N01_POL_O.LogicCommand

	Tag Name	Type	Description
3	HABILITA_DEVICENET	Digital	
4	HERTZ	Analog	
5	HERTZ_ETHERNET	Analog	
6	LOCAL_REMOTO	Digital	
7	LOCAL_REMOTO_ETHEI	Digital	
8	REVERSE_DEVICENET	Digital	
9	REVERSE_ETHERNET	Digital	
10	START	Digital	
11	START_ETHERNET	Digital	
12	STOP	Digital	
13	STOP_ETHERNET	Digital	
14	PRUEBA_1	Digital	

Figura 71: Creación de un Tag en la plataforma FactoryTalkView. (Fuente propia)

- Se procede a crear los objetos y armar la plantilla para los comandos de control del HMI.



Appearance

Border style: Raised

Border width: 4

Back style: Solid

Pattern style: None

Shape: Rectangle

☑ Border uses back color

Back color

Border color

Pattern color

Highlight color

□ Blink

Display settings

Display: COMANO DE CONTI

□ Use variable Display

● Parameter file:

○ Parameter list:

□ Display position Top position: 0 Left position: 0

□ Use Variable Display Position

Touch margins

Horizontal margin: 0

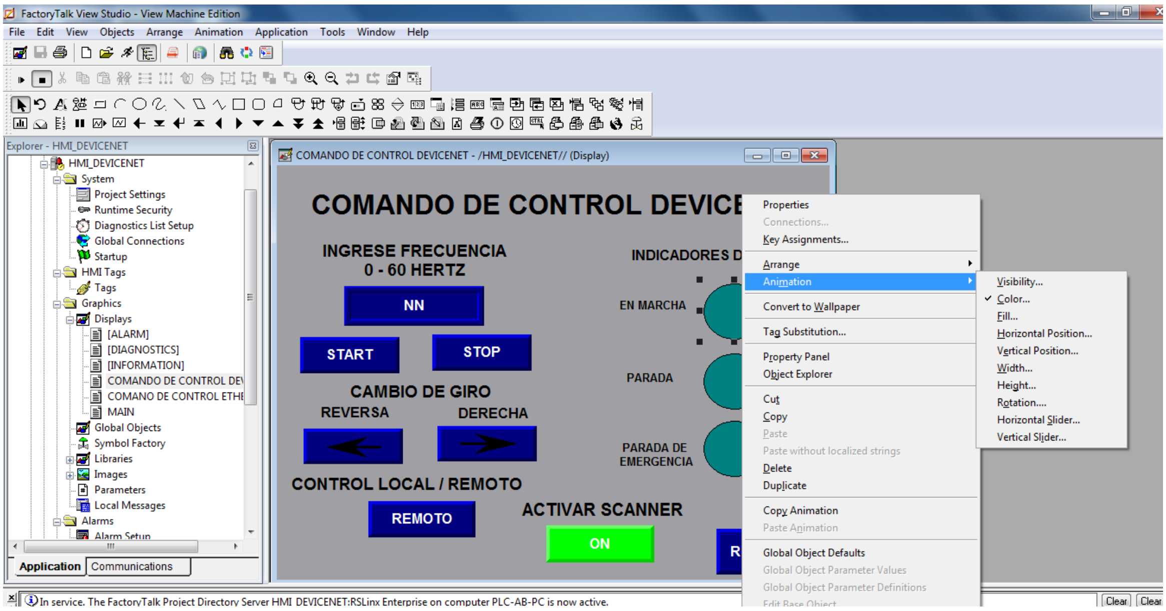
Vertical margin: 0

Other

☑ Audio

Figura 72: Configuración de los objetos creados en el Main Display (Fuente propia)

- Se atribuyen animaciones a los objetos que requieran para su cambio de estado según nuestra lógica.

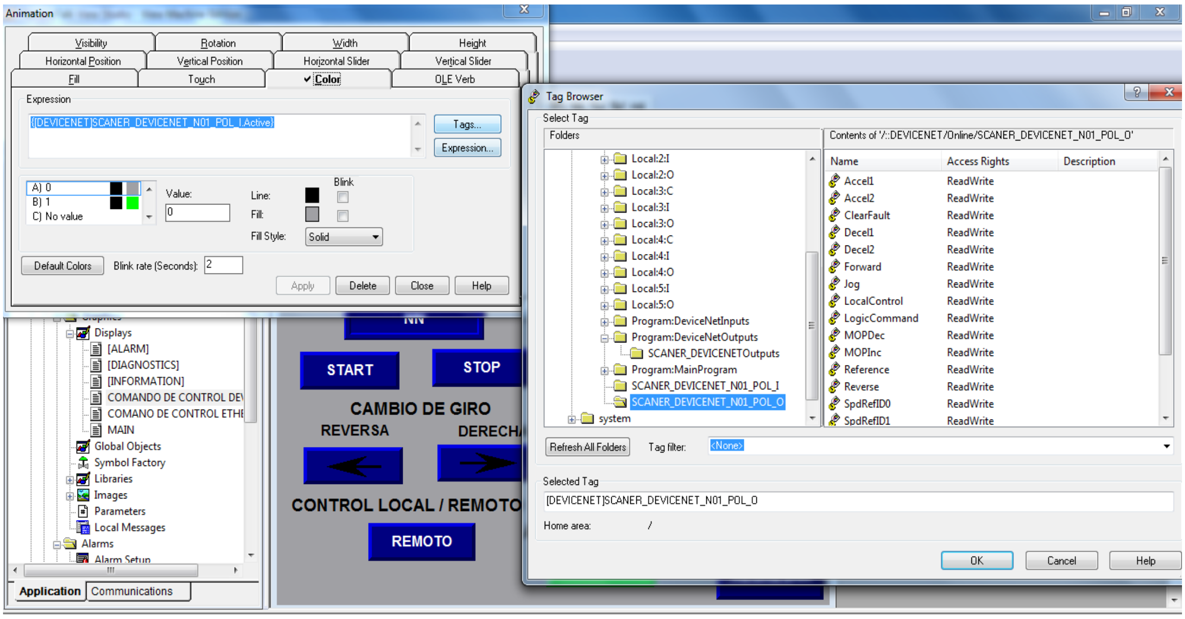


In service. The FactoryTalk Project Directory Server HMI DEVICENET:RSLinx Enterprise on computer PLC-AB-PC is now active.

Figura 73: Agregando animaciones a los objetos en el Display COMANDOS DE CONTROL DEVICENET.

(Fuente propia)

- Atribuidas las animaciones, se proceden a direccionar a las etiquetas designadas.



Expression

{[DEVICENET]SCANER_DEVICENET_N01_POL_I.Active}

A) 0
B) 1
C) No value

Value: 0

Line:

Fill:

Fill Style: Solid

Blink rate (Seconds): 2

Select Tag

Folders

Local:2:I
Local:2:O
Local:3:C
Local:3:I
Local:3:O
Local:4:C
Local:4:I
Local:4:O
Local:5:I
Local:5:O
Program:DeviceNetInputs
Program:DeviceNetOutputs
SCANER_DEVICENETOutputs
Program:MainProgram
SCANER_DEVICENET_N01_POL_I
SCANER_DEVICENET_N01_POL_O
system

Contents of '/::DEVICENET/Online/SCANER_DEVICENET_N01_POL_O'

Name	Access Rights	Description
Accel1	ReadWrite	
Accel2	ReadWrite	
ClearFault	ReadWrite	
Decel1	ReadWrite	
Decel2	ReadWrite	
Forward	ReadWrite	
Jog	ReadWrite	
LocalControl	ReadWrite	
LogicCommand	ReadWrite	
MOPDec	ReadWrite	
MOPInc	ReadWrite	
Reference	ReadWrite	
Reverse	ReadWrite	
SpdRefID0	ReadWrite	
SpdRefID1	ReadWrite	

Tag filter: <None>

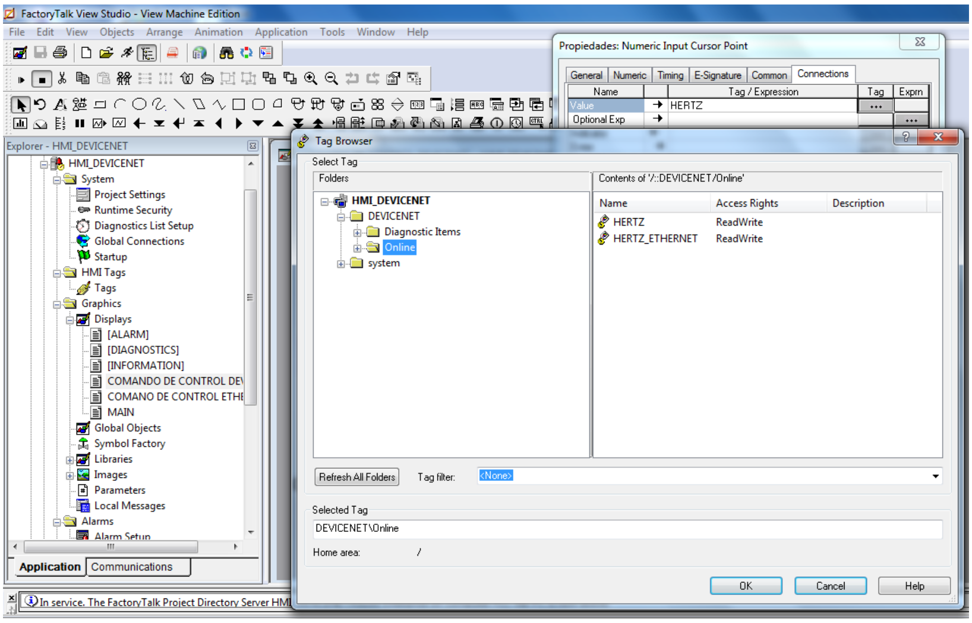
Selected Tag

[DEVICENET]SCANER_DEVICENET_N01_POL_O

Home area: /

Figura 74: Direccionamiento de los Objetos de Control.

(Fuente propia)



Name		Tag / Expression	Tag	Exprn
Value	→	HERTZ	...	
Optional Exp	→			...

Contents of '/::DEVICENET/Online'

Name	Access Rights	Description
HERTZ	ReadWrite	
HERTZ_ETHERNET	ReadWrite	

Tag filter: <None>

Selected Tag: DEVICENET\Online

Home area: /

Figura 75: Ejemplo del direccionamiento del tag HERTZ de RSlogix 5000.

(Fuente propia)

CAPITULO III
ANALISIS DE RESULTADOS

3.1 PRUEBAS DE OPERATIVIDAD DE LOS MÓDULOS DE ENTRENAMIENTO DE CONTROL DE VELOCIDAD DE MOTORES

Luego de que se siguieron los procedimientos establecidos por el fabricante Rockwell Inc para instalar los dispositivos de control para la implementación de la Red Industrial basada en el estándar DEVICENET, en esta etapa se procedió a la revisión y verificación de la operatividad de los actuadores, dispositivos de control, fuentes de alimentación, dispositivos de conmutación y de las máquinas rotativas (máquinas eléctricas). Se efectuaron los ajustes necesarios para la operatividad del módulo.

Una vez verificado la operatividad de los componentes del módulo de entrenamiento, se procedió a verificar la operatividad del módulo de variador de frecuencia utilizando la aplicación del control de velocidad de un motor asíncrono trifásico.

A continuación se muestran las imágenes de los módulos de variadores de frecuencia para el control de un motor síncrono trifásico, a los cuáles se le realizó las pruebas de operatividad de la aplicación en el Laboratorio de Automatización y Control de la Escuela Profesional de Ingeniería Electrónica de la Universidad Privada de Tacna.

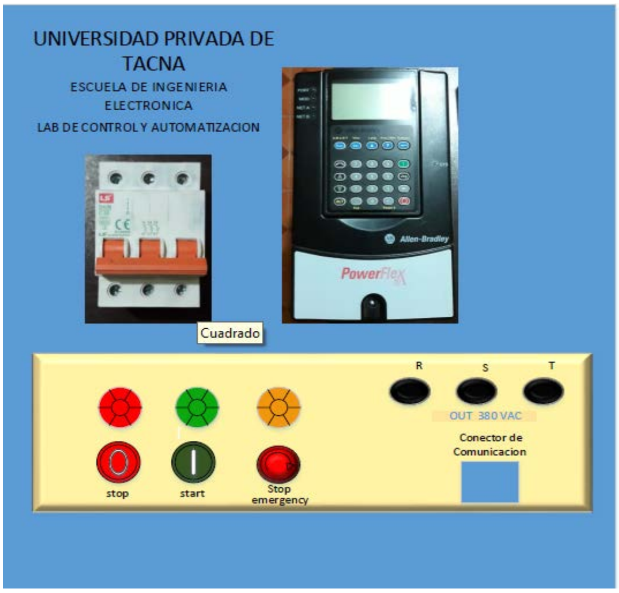

UNIVERSIDAD PRIVADA DE TACNA
ESCUELA DE INGENIERIA ELECTRONICA
LAB DE CONTROL Y AUTOMATIZACION
Allen-Bradley
PowerFlex
Cuadrado
R
S
T
OUT 380 VAC
Conector de Comunicacion
stop
start
Stop emergency

Figura 76: Módulo de variador de frecuencia implementado

(Fuente propia)

3.1.1 Ajustes efectuados para la operatividad del módulo

Se revisaron las conexiones internas para verificar la alimentación y control del variador, se recconectaron las borneras para evitar dañar los conectores inrternos del variador

Figura 77: Revisión de las conexiones internas del módulo

(Fuente propia)

Figura 78: Reconexión de las borneras del módulo del variador

(Fuente propia)

Se revisaron las conexiones y se programan los parámetros básicos para las pruebas de funcionamiento control local.

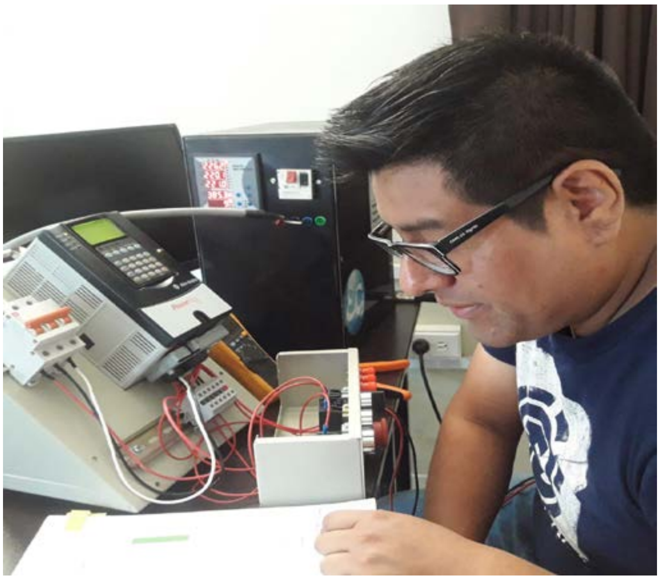


Figura 79: Ingreso de parámetros al variador

(Fuente propia)

Se realizó el reacondicionamiento del cableado para su mejor acondicionamiento en el módulo.

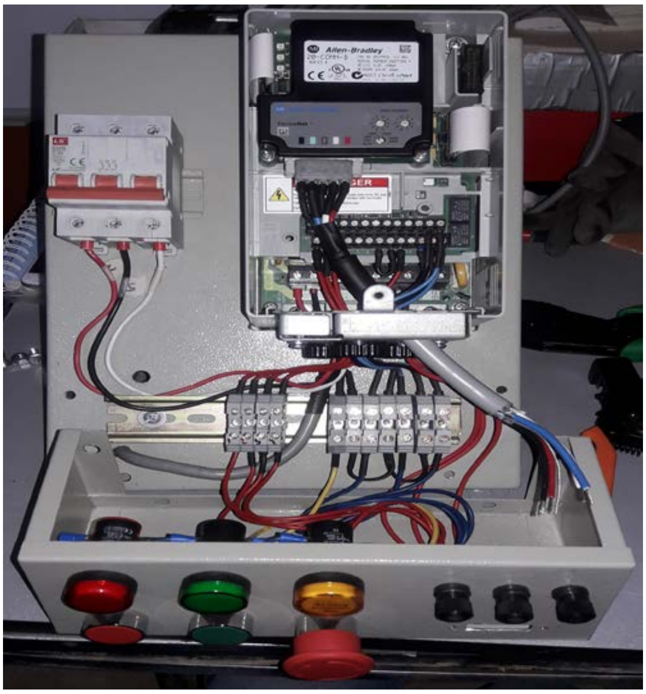


Figura 80: Reacondicionamiento del cableado del módulo

(Fuente propia)

Finalmente se procede a conectar al switch principal, comprobándose la operatividad del módulo variador de frecuencia.

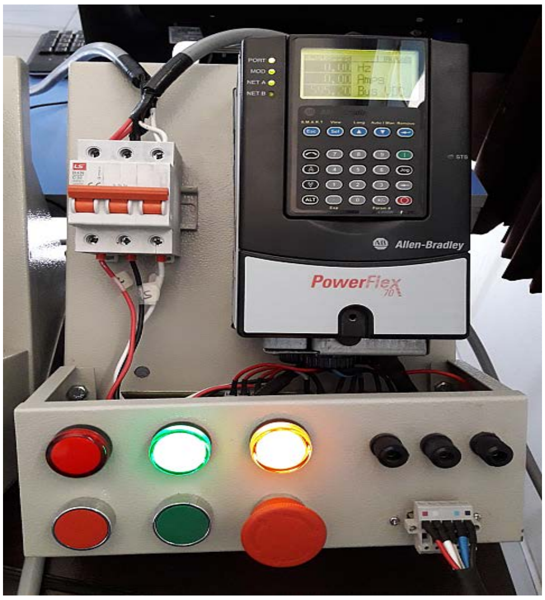


Figura 81: Módulo variador operativo

(Fuente propia)

3.2 EVALUACIÓN DE LA OPERATIVIDAD DE LOS MÓDULOS DE ENTRENAMIENTO

- Al realizar las pruebas de operatividad del variador de frecuencia PowerFlex 70 tuvimos inconvenientes con el accionamiento del motor trifásico, debido a que se podía visualizar que el PLC enviaba la señal al Variador de frecuencia, sin embargo el motor no giraba. Se pudo solucionar el inconveniente presentado, ya que se trataba de una omisión en la configuración en el Variador de frecuencia.
- En las primeras pruebas tal como se ve en las imágenes captadas, se pudo observar el accionamiento del motor trifásico, posteriormente se realizó la prueba de inversión del giro del motor y su posterior detención. Ya en pruebas posteriores se agregó las funciones de variación de la velocidad.
- Finalmente, después de realizar algunos modificaciones en la configuración de los dispositivos del módulo variador de frecuencia para verificar su operativdad, las pruebas se dieron con éxito, los equipos funcionaron de manera adecuada y se pudo comprobar la comunicación en la red industrial DEVICENET para el control de arranque y velocidad de un motor asíncrono trifásico mediante un variador de frecuencia.

CONCLUSIONES

- Se concretó el diseño y la implementación de los módulos de variadores de frecuencia utilizando el PLC ALLEN BRADLEY 1768 L23, el variador de frecuencia POWERFLEX 70, el scanner 1769 – SDN, de acuerdo al objetivo propuesto.

- Se verificó la operatividad del sistema de control que comprende el módulo y que opera dentro de la red de comunicación industrial que utiliza el estándar DEVICENET luego de realizar algunas modificaciones en la configuración de los dispositivos de control.

- Se verificó la operatividad de la red industrial DEVICENET implementada al desarrollar la aplicación que permite mejorar el control de arranque y velocidad de un motor asíncrono trifásico utilizando un variador de frecuencia. Las pruebas de operatividad se desarrollaron en el laboratorio de Control y Automatización de la Escuela Profesional de Ingeniería Electrónica de la Universidad Privada de Tacna, ubicada en el Campus Capanique 1.

RECOMENDACIONES

- Es necesario, que en el laboratorio de Control y Automatización de la carrera de ingeniería electrónica adecue sus instalaciones eléctricas para establecer una conexión eléctrica trifásica estabilizada y con protección de toma a tierra, como protección.

- Se requiere que previamente al desarrollo de un módulo de entrenamiento en redes industriales, es necesario familiarizarse con el manejo de los dispositivos, los protocolos de comunicación y los estándares que regulan la operatividad de las redes industriales.

- Es necesario que la operación y configuración de los módulos de variadores de frecuencia inicialmente sea realizado por el docente de clases, cuando los estudiantes de ingeniería electrónica desarrollen sus experiencias de laboratorio en redes industriales

- Es indispensable que, al momento de configurar los dispositivos de control, se deben seguir los procedimientos recomendados en los manuales técnicos de los fabricantes de dichos equipos, ya que contienen la información de cómo configurar los dispositivos de control, así como del actuador, que en nuestro caso fue el variador de frecuencia.

REFERENCIAS BIBLIOGRÁFICAS

1. **Canaza, Ch**. (2017) "Diseño e implementación de una ampliación de red industrial DEVICENET para el mejoramiento en la supervisión de los interruptores de parada de emergencia con cuerdas de seguridad (pullcords) en un sistema de fajas transportadoras para horno industrial de empresas del sector minero, año 2015", Arequipa, Perú.
   Del sitio web: http://repositorio.unsa.edu.pe/bitstream/handle/UNSA/3310/IEcavacj.pdf?sequence=1&isAllowed=y

2. **E-DUCATIVA CATEDU**. (2016). "*Máquinas eléctricas rotativas. Conceptos básicos".* 2018, de Plataforma e-ducativa aragonea
   Sitio web: http://e-ducativa.catedu.es/44700165/aula/archivos/repositorio/3000/3016/html/21_introduccin.html

3. **García, J. & Lorenzana, J. & Salinas, R.** (2013) "Implementar una red DEVICENET para la automatización de un sistema de bandas transportadoras" Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica México DF,
   Del sitio web: https://tesis.ipn.mx/jspui/bitstream/123456789/12778/1/IMPLEMENTAR%20UNA%20RED%20DEVICENET%20PARA%20LA%20AUTOMATIZACION%20DE%20UN%20SISTEMA%20DE%20BANDAS%20TRANSPORTADORAS.pdf

4. **García, J.** (2013). Tema 7: Introducción a la Automatización Industrial Departamento de Ingeniería de Sistemas y Automática, Universidad de Málaga, España.
   Del sitio web: http://mapir.isa.uma.es/varevalo/teaching/automatica/pdfs/Tema%2007%20-%20Introducci%C3%B3n%20a%20la%20Automatizaci%C3%B3n%20Industrial%20v7_vicente.pdf

5. **INFOPLC.** (2010) TEMA 1: Automatización con PLCs".
   Del sitio web: http://www.infoplc.net/files/documentacion/automatas/infoPLC_net_1_Intro_ Automatas.pdf

6. **Mateos, F.** (Nov. 2001) Estandarización con autómatas **programables**. Sistema automatizado. Universidad de Oviedo, España.
Del sitio web: http://isa.uniovi.es/docencia/iea/teoria/plc.pdf

7. **Rockwell Automation Inc**, (2018). FactoryTalk View Machine Edition User's Guide
Del sitio web: https://literature.rockwellautomation.com/idc/groups/literature/documents/um/viewme-um004_-en-e.pdf

8. **Rockwell Automation Inc. (**2014) CompactLogix Controllers Specifications,Technical Data
Del sitio web: https://www.elit.ee/docs/Automation/ALBR1768L43S_spec.pdf

9. **Rockwell Automation Inc. (**2009). 1769-SDN DeviceNet Scanner Module, User Manual.
Del sitio web: https://literature.rockwellautomation.com/idc/groups/literature/documents/um/1769-um009_-en-p.pdf

10. **Rockwell Automation Inc. (**2018). RSLinx Classic Getting Results Guide
Del sitio web: https://literature.rockwellautomation.com/idc/groups/literature/documents/gr/linx-gr001_-en-e.pdf

11. **Rockwell Automation Inc. (**2018). Getting Results with RSNetWorx for DeviceNet
Del sitio web: https://literature.rockwellautomation.com/idc/groups/literature/documents/gr/dnet-gr001_-en-e.pdf

12. **Rockwell Automation Inc. (**2018). LOGIX 5000 Controllers General Instructions Reference Manual
Del sitio web: https://literature.rockwellautomation.com/idc/groups/literature/documents/rm/1756-rm003_-en-p.pdf

13. **Rockwell Automation Inc. (**2018). PowerFlex 70 Adjustable Frequency AC Drives User Manual
Del sitio web: https://literature.rockwellautomation.com/idc/groups/literature/documents/um/20a-um001_-en-p.pdf

ANEXO:

DISPOSITIVOS DE CONTROL DE MODULO DE VARIADOR DE FRECUENCIA ALLEN-BRADLEY

1769-SDN DeviceNet Scanner Module

Allen-Bradley

Catalog Numbers 1769-SDN

User Manual

Allen-Bradley • Rockwell Software

Rockwell Automation

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING ⚠	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION ⚠	Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence
SHOCK HAZARD ⚠	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
BURN HAZARD ⚠	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

The information below summarizes the changes to this manual since the last printing.

We have included change bars as shown to the right of this paragraph to help you find new and updated information in this release of the manual. The table below lists the changes that have been made to this revision of the manual.

Topic	Page
Automatically Configure a DeviceNet Network	Chapter 4
Updated information	Index

Preface

Chapter 1

Overview

Chapter 2

Quick Start for Experienced Users

Chapter 3

Installation and Wiring

Chapter 4

Automatically Configure a DeviceNet Network

Chapter 5

Manually Configure the DeviceNet Network

Chapter 6

DeviceNet I/O Image

Chapter 7

Use the 1769-SDN Scanner Module with MicroLogix Controllers

Chapter 8

Troubleshooting

Appendix A

1769-SDN DeviceNet Class Codes

Appendix B

CompactLogix Backup on the DeviceNet Network

Glossary

Index

Read this preface to familiarize yourself with the rest of the manual.

About This Manual

This manual is a user manual for the Compact I/O 1769-SDN DeviceNet scanner module. It describes the procedures you use to install, program, and troubleshoot your scanner module. This manual:

- provides instructions on installing the scanner module.
- contains information about using the scanner module on the DeviceNet network.
- provides tips on troubleshooting the scanner module.
- contains application examples to show how the scanner module is used with various programmable controllers.

IMPORTANT

This manual focuses on the 1769-SDN scanner module with a MicroLogix 1500 control system on the DeviceNet network. Topics covered include using AutoScan, configuring, bridging, connecting, and controlling your DeviceNet network.

For information about using the 1769-SDN scanner module with a CompactLogix system, refer to DeviceNet Modules in Logix5000 Control Systems User Manual, publication DNET-UM004.

Who Should Use This Manual

Use this manual if you are responsible for designing, installing, programming, or troubleshooting control systems that use Rockwell Automation programmable controllers.

You should have a basic understanding of electrical circuitry and familiarity with relay logic. If you do not, obtain the proper training before using this product.

Conventions in This Manual

The following conventions are used throughout this manual:

- Bulleted lists such as this one provide information, not procedural steps.
- Numbered lists provide sequential steps or hierarchical information.

Additional Resources

The following documents contain additional information concerning Rockwell Automation products. Contact your local Rockwell Automation distributor to order hard copy publications. For electronic copies, go to http://literature.rockwellautomation.com.

Resource	Description
DeviceNet Modules in Logix5000 Control Systems User Manual, publication DNET-UM004	Describes configuring the CompactLogix controllers on the DeviceNet network.
ControlNet Modules in Logix5000 Control Systems User Manual, publication CNET-UM001	Describes configuring the CompactLogix controllers on the ControlNet network.
RSNetWorx for DeviceNet Getting Results Guide, publication DNET-GR001	Describes using RSNetWorx for DeviceNet software (catalog number 9357-DNETL3).
CompactLogix System User Manual, publication 1769-UM007	Describes planning, mounting, wiring, and troubleshooting your CompactLogix system. This manual focuses on the 1769-L20 and 1796-L30 CompactLogix controllers.
MicroLogix 1500 Programmable Controllers User Manual, publication 1764-UM001	Planning, mounting, wiring, and troubleshooting your MicroLogix 1500 system
Compact I/O Analog Modules User Manual, publication 1769-UM002	Installing, configuring, and using Compact I/O analog modules
DeviceNet Interface User Manual, publication 1761-UM005	How to install and use the DeviceNet Interface (catalog number 1761-NET-DNI)
DeviceNet Media Design and Installation Guide, publication DNET-UM072	DeviceNet network planning information
Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1	Grounding and wiring Allen-Bradley programmable controllers
National Electrical Code - Published by the National Fire Protection Association of Boston, MA.	Wire sizes and types for grounding electrical equipment

Overview

Introduction

This chapter provides an overview of communication between the CompactLogix and MicroLogix 1500 programmable controllers and DeviceNet devices via the 1769-SDN scanner module.

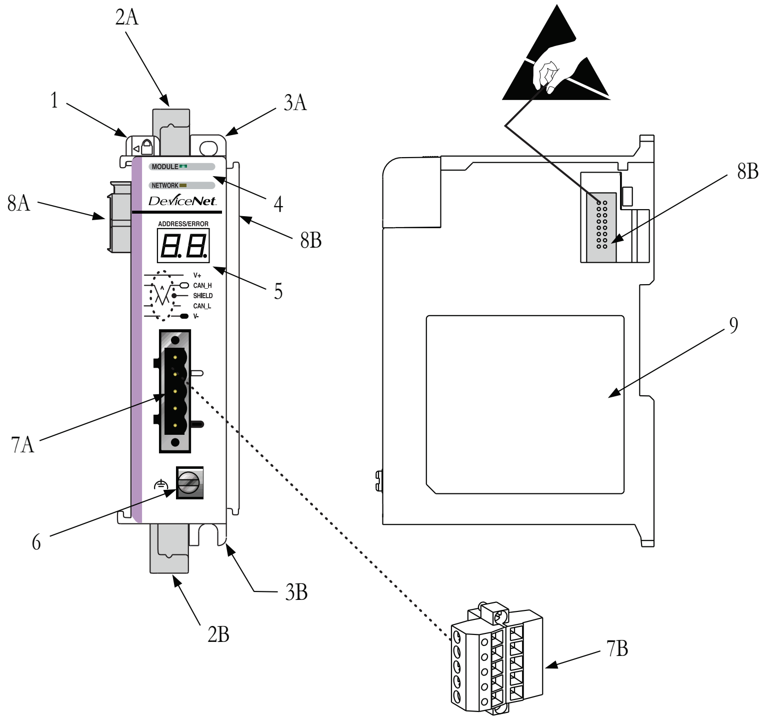
The configuration data tables and the RSNetWorx for DeviceNet dialog boxes used to configure the data tables are also described in this chapter. Before configuring your scanner, you must understand these items:

- Data exchange between the programmable controller and DeviceNet devices through the scanner
- User-configurable scanner module data tables
- Role of RSNetWorx for DeviceNet software

These topics are covered briefly in this chapter and in more detail throughout the rest of the manual.

Module Features

Use the following figure to identify the features of the scanner.



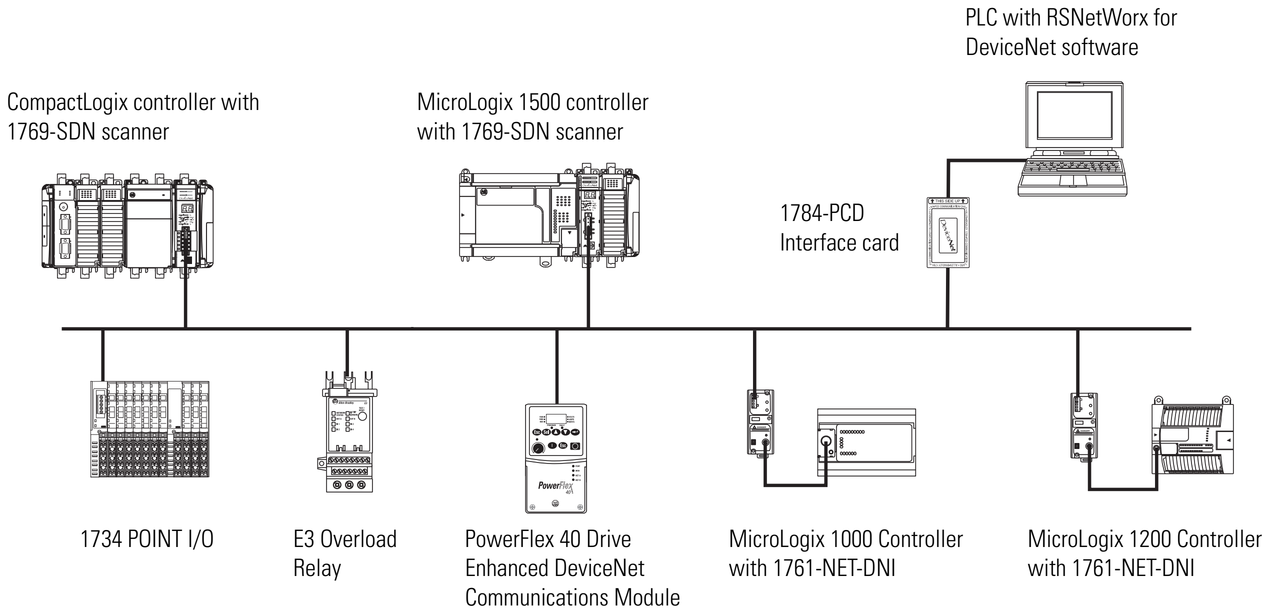
Module Features

Item	Description
1	Bus lever (with locking function)
2A	Upper DIN rail latch
2B	Lower DIN rail latch
3A	Upper panel mounting tab
3B	Lower panel mounting tab
4	Module and Network status LEDs
5	Address and Error numeric display
6	Grounding screw
7A	DeviceNet mating male receptacle
7B	Removable DeviceNet female connector
8A	Movable bus connector with female pins
8B	Bus connector with male pins
9	Nameplate label

Scanner Module Operation

In a typical configuration, the scanner module acts as an interface between DeviceNet devices and the programmable controller.

Device Network


PLC with RSNetWorx for DeviceNet software
CompactLogix controller with 1769-SDN scanner
MicroLogix 1500 controller with 1769-SDN scanner
1784-PCD Interface card
1734 POINT I/O
E3 Overload Relay
PowerFlex 40 Drive Enhanced DeviceNet Communications Module
MicroLogix 1000 Controller with 1761-NET-DNI
MicroLogix 1200 Controller with 1761-NET-DNI

The scanner module communicates with DeviceNet devices over the network to:

- Read inputs from slave devices
- Write outputs to slave devices
- Communicate with peer devices (messaging)
- Upload/download programs to a 1764-LRP based MicroLogix 1500 controller across a DeviceNet network

Communication with Your Slave Devices

The scanner module communicates with devices via strobe, poll, change of state, or cyclic I/O messages. It uses these messages to solicit data from or deliver data to each device. Data received from the devices, input data, is organized by the scanner module and made available to the controller. Data sent from your controller, output data, is organized in the scanner module and sent on to your devices.

- A strobe message is a multicast transfer of data that is 64 bits in length sent by the scanner module that initiates a response from each strobed slave device.

  The strobe devices respond with their data, which can be as much as 8 bytes of information. As a slave device, the scanner module does not support the strobe message.

- A poll message is a point-to-point transfer of data from 0...128 bytes sent by the scanner module to the slave device.

  The poll message also initiates a response from each poll slave. The slave device responds with its input data from 0...128 bytes.

- A change-of-state message is a transfer of data sent whenever a data change occurs.

  A user-configurable heartbeat rate allows devices to indicate proper operation during intervals between data changes.

- A cyclic message is a transfer of data sent at a specific user-configurable rate, such as every 50 ms.

IMPORTANT Throughout this document, input and output are defined from the controller's point of view. Output is data sent from the controller to a device. Input is data collected by the controller from a device.

In addition to I/O messaging, the scanner module also supports PCCC and CIP explicit messaging, defined later in this manual.

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

Module Command Array Bit Assignments

Output Word	Bit	Description	Behavior
0	0	Run	This bit controls when the module scans its mapped slave devices. When set (1), the scanner module will process I/O data as defined by its scanlist. The Fault and Disable Network command bits must be clear (0) to scan the network.
	1	Fault	When set, the scanner's I/O mode will be Halt; messaging will still operate. The fault bit is primarily used to artificially set the slave devices into a fault state due to some event or condition within the control program.
	2	Disable network	When set, the scanner module is functionally removed from the network.
	3	Reserved[1]	N/A
	4	Reset	Restarts access to the DeviceNet network.
	5…15	Reserved[1]	N/A
1	16…31	Reserved[1]	N/A

[1] Do not manipulate reserved bits. Doing so may interfere with future compatibility.

Input Data Image - CompactLogix

The input data image is transferred from the scanner module to the controller across the Compact I/O bus.

Word	Description	Data Type
0…89	DeviceNet slave inputs	90-DINT array

Output Data Image - CompactLogix

The output data image is transferred from the controller to the scanner module across the Compact I/O bus.

Word	Description	Data Type
0…89	DeviceNet slave outputs	90-DINT array

For additional information about the CompactLogix image structure, refer to the DeviceNet Modules in Logix5000 Control Systems User Manual, publication DNET-UM004.

RSNetWorx for DeviceNet Software as a Configuration Tool

RSNetWorx for DeviceNet software is used to configure the scanner's slave devices. This software tool connects to the scanner module over the DeviceNet network via an RS-232 interface (1770-KFD module) or PC card (1784-PCD or 1784-PCID).

We recommend RSNetworx for DeviceNet software, version 3.00 or later.

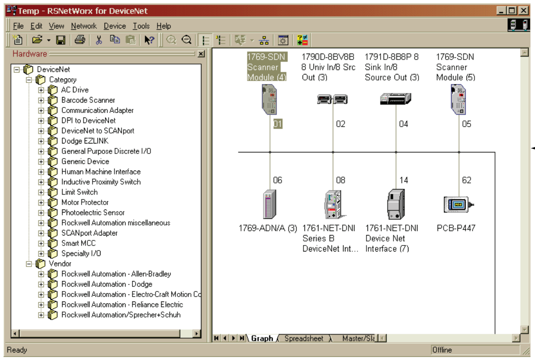
If your RSNetWorx configuration software does not include the required electronic data sheet (EDS) file, go to http://www.ab.com/networks/eds.

Register the new EDS file by using the EDS wizard in RSNetWorx for DeviceNet software. Access the wizard from the Tools menu. This configuration tool lets you to identify all of the DeviceNet devices and their locations in your system.

The controller must be in Program mode, or the scanner module in Idle mode (bit 0 of the Module Command Array = 0) for the scanner module to accept the configuration information.

RSNetWorx Configuration Dialog Map

The main **RSNetWorx** dialog.

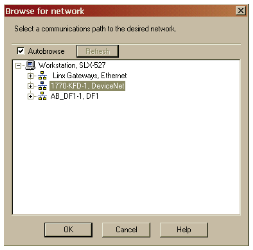

Temp - RSNetWorx for DeviceNet

Hardware

- DeviceNet
  - Category
    - AC Drive
    - Barcode Scanner
    - Communication Adapter
    - DPI to DeviceNet
    - DeviceNet to SCANport
    - Dodge EZLINK
    - General Purpose Discrete I/O
    - Generic Device
    - Human Machine Interface
    - Inductive Proximity Switch
    - Limit Switch
    - Motor Protector
    - Photoelectric Sensor
    - Rockwell Automation miscellaneous
    - SCANport Adapter
    - Smart MCC
    - Specialty I/O
  - Vendor
    - Rockwell Automation - Allen-Bradley
    - Rockwell Automation - Dodge
    - Rockwell Automation - Electro-Craft Motion Co
    - Rockwell Automation - Reliance Electric
    - Rockwell Automation/Sprecher+Schuh

Ready | Offline

File Edit View Network Device Tools Help
1769-SDN Scanner Module (4)
01
1790D-8BV8B 8 Univ In/8 Src Out (3)
02
1791D-8B8P 8 Sink In/8 Source Out (3)
04
1769-SDN Scanner Module (5)
05
06
1769-ADN/A (3)
08
1761-NET-DNI Series B DeviceNet Int...
14
1761-NET-DNI Device Net Interface (7)
62
PCB-P447
Graph
Spreadsheet
Master/Sl


Browse for network

Select a communications path to the desired network.

☑ Autobrowse

- Workstation, SLX-527
  - Linx Gateways, Ethernet
  - 1770-KFD-1, DeviceNet
  - AB_DF1-1, DF1

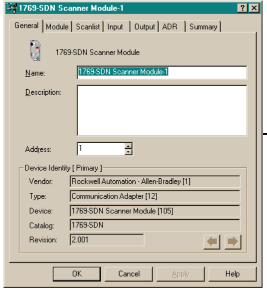
Click Online and select the driver to browse the network.

Double-click the 1769-SDN icon to access the 1769-SDN scanner module.

Click on the Scanlist tab to access the scanlist.

Move the device into the scanlist.

Click Download to Scanner to download the scanlist.


1769-SDN Scanner Module-1

1769-SDN Scanner Module

Name: 1769-SDN Scanner Module-1

Description:

Address: 1

Device Identity [Primary]

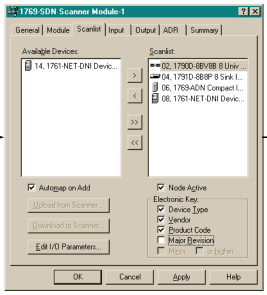
Vendor: Rockwell Automation - Allen-Bradley [1]

Type: Communication Adapter [12]

Device: 1769-SDN Scanner Module [105]

Catalog: 1769-SDN

Revision: 2.001


1769-SDN Scanner Module-1

Available Devices:

- 14, 1761-NET-DNI Devic...

Scanlist:

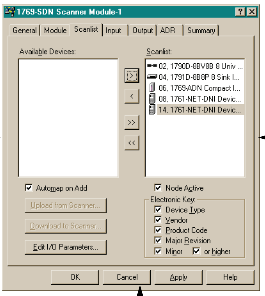
- 02, 1790D-8BV8B 8 Univ ...
- 04, 1791D-8B8P 8 Sink I...
- 06, 1769-ADN Compact I...
- 08, 1761-NET-DNI Devic...

☑ Automap on Add

☑ Node Active

Electronic Key:

☑ Device Type
☑ Vendor
☑ Product Code
☐ Major Revision
☐ Minor ☐ or higher


1769-SDN Scanner Module-1

Available Devices:

Scanlist:

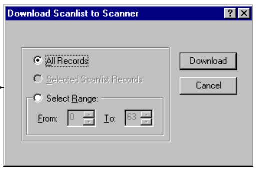
- 02, 1790D-8BV8B 8 Univ ...
- 04, 1791D-8B8P 8 Sink I...
- 06, 1769-ADN Compact I...
- 08, 1761-NET-DNI Devic...
- 14, 1761-NET-DNI Devic...

☑ Automap on Add

☑ Node Active

Electronic Key:

☑ Device Type
☑ Vendor
☑ Product Code
☑ Major Revision
☑ Minor ☑ or higher


Download Scanlist to Scanner

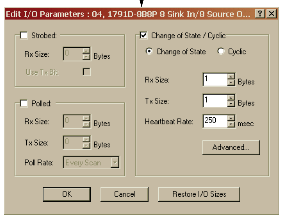
● All Records
○ Selected Scanlist Records
○ Select Range:

From: 0 To: 63

Double-click the device in the scanlist to edit a device's I/O parameters.

Click the Input tab and click the AutoMap button to automatically map input devices.

Click the Output tab and click the AutoMap button to automatically map output devices.


Edit I/O Parameters : 04, 1791D-8B8P 8 Sink In/8 Source O...

☐ Strobed:

Rx Size: 0 Bytes

Use Tx Bit: ☐

☐ Polled:

Rx Size: 0 Bytes

Tx Size: 0 Bytes

Poll Rate: Every Scan

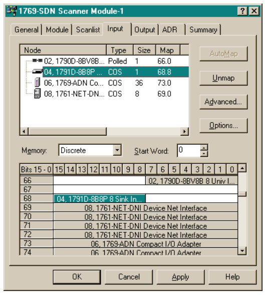
☑ Change of State / Cyclic

● Change of State ○ Cyclic

Rx Size: 1 Bytes

Tx Size: 1 Bytes

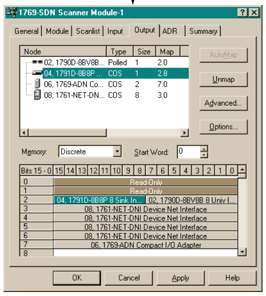
Heartbeat Rate: 250 msec


1769-SDN Scanner Module-1

Node	Type	Size	Map
02, 1790D-8BV8B...	Polled	1	66.0
04, 1791D-8B8P ...	COS	1	68.8
06, 1769-ADN Co...	COS	36	73.0
08, 1761-NET-DN...	COS	8	69.0

Memory: Discrete Start Word: 0

Bits 15 - 0	15 – 8	7 – 0
66		02, 1790D-8BV8B 8 Univ I...
67		
68	04, 1791D-8B8P 8 Sink In...	
69	08, 1761-NET-DNI Device Net Interface	
70	08, 1761-NET-DNI Device Net Interface	
71	08, 1761-NET-DNI Device Net Interface	
72	08, 1761-NET-DNI Device Net Interface	
73	06, 1769-ADN Compact I/O Adapter	
74	06, 1769-ADN Compact I/O Adapter	


1769-SDN Scanner Module-1

Node	Type	Size	Map
02, 1790D-8BV8B...	Polled	1	2.0
04, 1791D-8B8P ...	COS	1	2.8
06, 1769-ADN Co...	COS	2	7.0
08, 1761-NET-DN...	COS	8	3.0

Memory: Discrete Start Word: 0

Bits 15 - 0	15 – 8	7 – 0
0	Read-Only	
1	Read-Only	
2	04, 1791D-8B8P 8 Sink In...	02, 1790D-8BV8B 8 Univ I...
3	08, 1761-NET-DNI Device Net Interface	
4	08, 1761-NET-DNI Device Net Interface	
5	08, 1761-NET-DNI Device Net Interface	
6	08, 1761-NET-DNI Device Net Interface	
7	06, 1769-ADN Compact I/O Adapter	
8		

Quick Start for Experienced Users

Introduction

This chapter helps you get started using the 1769-SDN scanner module.

Topic	Page
Required Tools and Equipment	19
What You Need to Do	20

Procedures are based on the assumption that you have an understanding of Rockwell Automation controllers. You should understand electronic process control and be able to interpret the ladder logic instructions required to generate the electronic signals that control your application. Because it is a start-up guide for experienced users, this chapter does not contain detailed explanations about the procedures listed.

Required Tools and Equipment

Have the following tools and equipment ready:

- Personal computer
- Programmable controller: CompactLogix or MicroLogix 1500 system
- 1770-KFD RS-232 DeviceNet adapter or 1784-PCIDS, 1784-U2DN DeviceNet interface card
- For network communication: RSLinx software, version 2.30 or later
- For DeviceNet network configuration:
  - RSNetWorx for DeviceNet software, version 3.00 or later
- For ladder logic programming:
  - RSLogix 500 programming software, version 5.00.10 or later, or
  - RSLogix 5000 programming software, version 8.02 or later
- 1769-SDN scanner module
- Mounting hardware
- Screwdriver

What You Need to Do

Follow these steps to get started using the 1769-SDN scanner module.

1. Verify planned system configuration.

a. Ensure system power supply has sufficient current.

Maximum Current Draw

Module	5V DC	24V DC
1769-SDN	440 mA	0 mA

The scanner module cannot be located more than four modules away from the system power supply.

b. Verify that the DeviceNet network has adequate power.

DeviceNet Power Requirements

Module	DeviceNet Power Requirements
1769-SDN	N.E.C. Class 2 90 mA @ 11V DC, max. 110 mA @ 25V DC, max. 200 mA for 1.5 ms inrush

2. Remove power.

ATTENTION ⚠

Remove power before removing or inserting this module. When you remove or insert a module with power applied, an electrical arc may occur. An electrical arc can cause personal injury or property damage by:

- sending an erroneous signal to your system's field devices, causing unintended machine motion.
- causing an explosion in a hazardous environment.

Electrical arcing causes excessive wear to contacts on both the module and its mating connector and can lead to premature failure.

3. Assemble and mount the I/O bank.

   The scanner module can be attached to an adjacent controller, power supply, or I/O module. The scanner module can be panel or DIN-rail mounted. Modules can be assembled before or after mounting.

   Be sure to observe minimum spacing guidelines for adequate ventilation.

4. Ground the scanner module and complete the DeviceNet network wiring.

5. Apply power to the system.

6. Be sure that the programming software and equipment is ready.

7. Use RSLinx to configure drivers.

8. Use RSNetWorx for DeviceNet software to configure the 1769-SDN scanner module and the DeviceNet devices.

9. Use RSLogix software to create your project and ladder logic.

10. Start the system.
    a. Apply power.
    b. Download your program and put the controller into Run mode.
    c. status indicators turn on solid green.

11. Monitor the scanner module status to check if the scanner module is operating correctly.

    Module status is reported by the status indicators and numeric display on the front of the scanner module. The information is also stored in the scanner module's input data file, so these bits can be used in your control program to flag an error.

Notes:

Installation and Wiring

This chapter describes how to install and wire the 1769-SDN scanner module. This table describes what this chapter contains and where to find specific information.

Power Requirements

The scanner module receives power through the Compact I/O bus interface from the +5V DC system power supply.

Maximum Current Draw

Module	5V DC	24V DC
1769-SDN	440 mA	0 mA

The scanner module also draws power from the DeviceNet network.

DeviceNet Power Requirements

Module	DeviceNet Power Requirements
1769-SDN	N.E.C. Class 2 90 mA @ 11V DC, max. 110 mA @ 25V DC, max. 200 mA for 1.5 ms, inrush

General Considerations

The Compact I/O system is suitable for use in an industrial environment when installed in accordance with these instructions. Specifically, this equipment is intended for use in clean, dry environments (Pollution Degree 2[1]) and to circuits not exceeding Over Voltage Category II[2] (IEC 60664-1).[3]

Hazardous Location Considerations

This equipment is suitable for use in Class I, Division 2, Groups A, B, C, D or nonhazardous locations only. The following WARNING statement applies to use in hazardous locations.

WARNING ⚠

EXPLOSION HAZARD

Substitution of components may impair suitability for Class I, Division 2.

Do not replace components or disconnect equipment unless power has been switched off or the area is known to be nonhazardous.

Do not connect or disconnect components unless power has been switched off or the area is known to be nonhazardous.

This product must be installed in an enclosure.

All wiring must comply with N.E.C. article 501-4(b).

[1] Pollution Degree 2 is an environment where, normally, only nonconductive pollution occurs except that occasionally a temporary conductivity caused by condensation shall be expected.

[2] Over Voltage Category II is the load level section of the electrical distribution system. At this level, transient voltages are controlled and do not exceed the impulse voltage capability of the product's insulation.

[3] Pollution Degree 2 and Over Voltage Category II are International Electrotechnical Commission (IEC) designations.

Preventing Electrostatic Discharge

ATTENTION

Electrostatic discharge (ESD) can damage integrated circuits or semiconductors if you touch the bus connector pins. Follow these guidelines when you handle the module:

- Touch a grounded object to discharge static potential.
- Wear an approved wrist-strap grounding device.
- Do not touch the bus connector or connector pins.
- Do not touch circuit components inside the module.
- Use a static-safe work station, if available.
- Keep the module in its static-shield box when it is not in use.

Removing Power

ATTENTION

Remove power before removing or inserting this module. When you remove or insert a module with power applied, an electrical arc may occur. An electrical arc can cause personal injury or property damage by:

- sending an erroneous signal to your system's field devices, causing unintended machine motion.
- causing an explosion in a hazardous environment.

Electrical arcing causes excessive wear to contacts on both the module and its mating connector and can lead to premature failure.

Reducing Noise

We recommend installing this module in an industrial enclosure to reduce the effects of electrical interference. Group your modules to minimize adverse effects from radiated electrical noise and heat.

Protecting the Circuit Board from Contamination

The printed circuit boards of the modules must be protected from dirt, oil, moisture, and other airborne contaminants. We recommend installing the system in an enclosure suitable for the environment to protect these boards. The interior of the enclosure should be kept clean and the enclosure door should be kept closed whenever possible.

System Planning

Consider the following when planning your system:

- The scanner module can communicate with up to 63 DeviceNet devices.
- The scanner, as a master, can own up to 63 slave I/O nodes.
- The scanner module can simultaneously be a master and a slave owned by another DeviceNet master.
- A 1769-ECR right end cap or 1769-ECL left end cap is required to terminate the end of the Compact I/O bus.
- Each bank of Compact I/O modules must have its own power supply.

  A MicroLogix 1500 controller acts as the power supply for modules directly connected to it.

- A Compact I/O power supply, or MicroLogix 1500 base unit, has limits on the amount of +5V DC and +24V DC current it can supply to modules in its I/O bank.

  These limits depend on the catalog number of the power supply, for example, 1769-PA2. A bank of modules must not exceed the current limits of the I/O bank power supply or MicroLogix 1500 base unit.

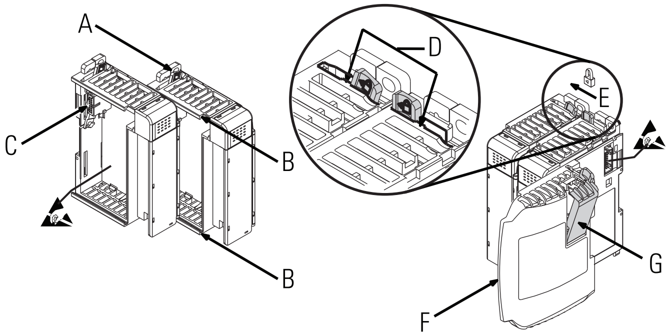
  Refer to the Compact 1769 Expansion I/O Power Supplies Installation Instructions, publication 1769-IN028, or the MicroLogix 1500 User Manual, publication 1764-UM001.

- The scanner module has a distance rating of four, therefore, the scanner module must be within four modules of the I/O bank's power supply.
- Determine the DeviceNet communication rate, based on standard DeviceNet considerations.
- Consider the number of words of I/O data the host controller supports.

For more information on planning your DeviceNet network, refer to the DeviceNet Media Design Installation Guide, publication DNET-UM072.

System Assembly

The scanner module can be attached to an adjacent controller, power supply, or I/O module. This procedure shows you how to assemble the Compact I/O system.



1. Disconnect power.

2. Check that the bus lever of the scanner module (A) is in the unlocked (fully right) position.

3. Use the upper and lower tongue-and-groove slots (B) to secure the modules together.

4. Move the scanner module back along the tongue-and-groove slots until the bus connectors (C) line up with each other.

5. Use your fingers or a small screwdriver to push the bus lever back slightly to clear the positioning tab (D).

6. Move the scanner module's bus lever fully to the left (E) until it clicks. Make sure it is locked firmly in place.

ATTENTION ⚠ When attaching I/O modules, it is very important that the bus connectors are securely locked together for a proper electrical connection.

7. Attach an end cap terminator (F) to the last module in the system by using the tongue-and-groove slots as before.

8. Lock the end cap bus terminator (G).

IMPORTANT A 1769-ECR or 1769-ECL right or left end cap must be used to terminate the end of the serial communication bus.

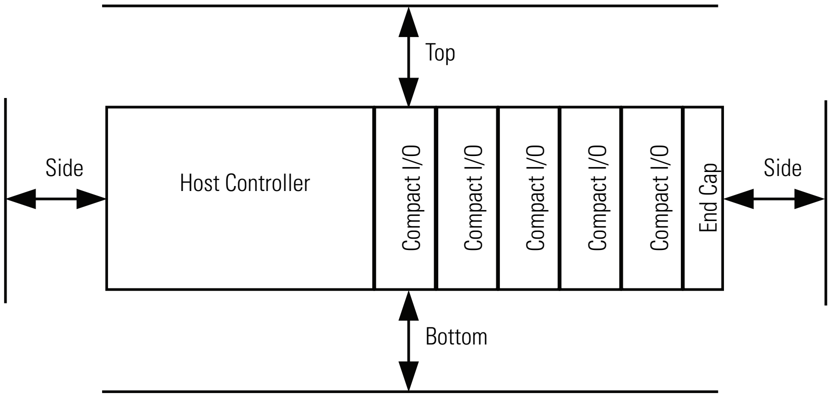
System Mounting

ATTENTION

During panel or DIN rail mounting of all devices, be sure that all debris (metal chips, wire strands) is prevented from falling into the module. Debris that falls into the module could cause damage at power up.

Minimum Spacing

Maintain spacing from enclosure walls, wireways, and adjacent equipment. Allow 50 mm (2 in.) of space on all sides for adequate ventilation, as shown below.



Allow at least 110 mm (4.33 in.) of enclosure depth to accommodate the scanner module and the DeviceNet connector.

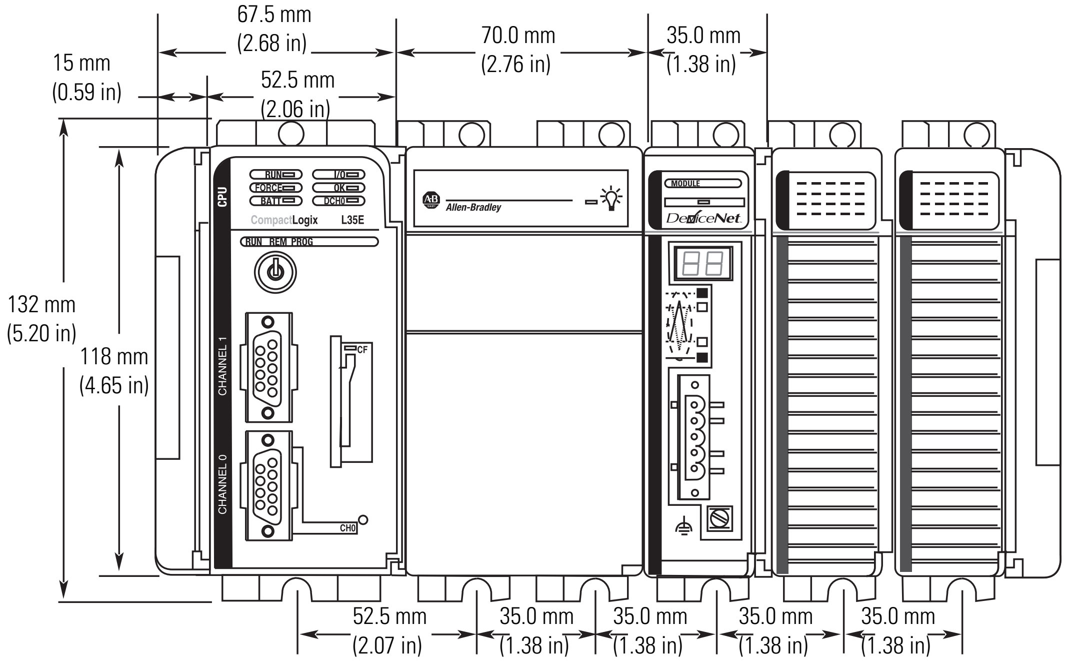
Panel Mounting

Mount the scanner module to a panel, using two screws per scanner module. Use M4 or #8 panhead screws. Mounting screws are required on every module.

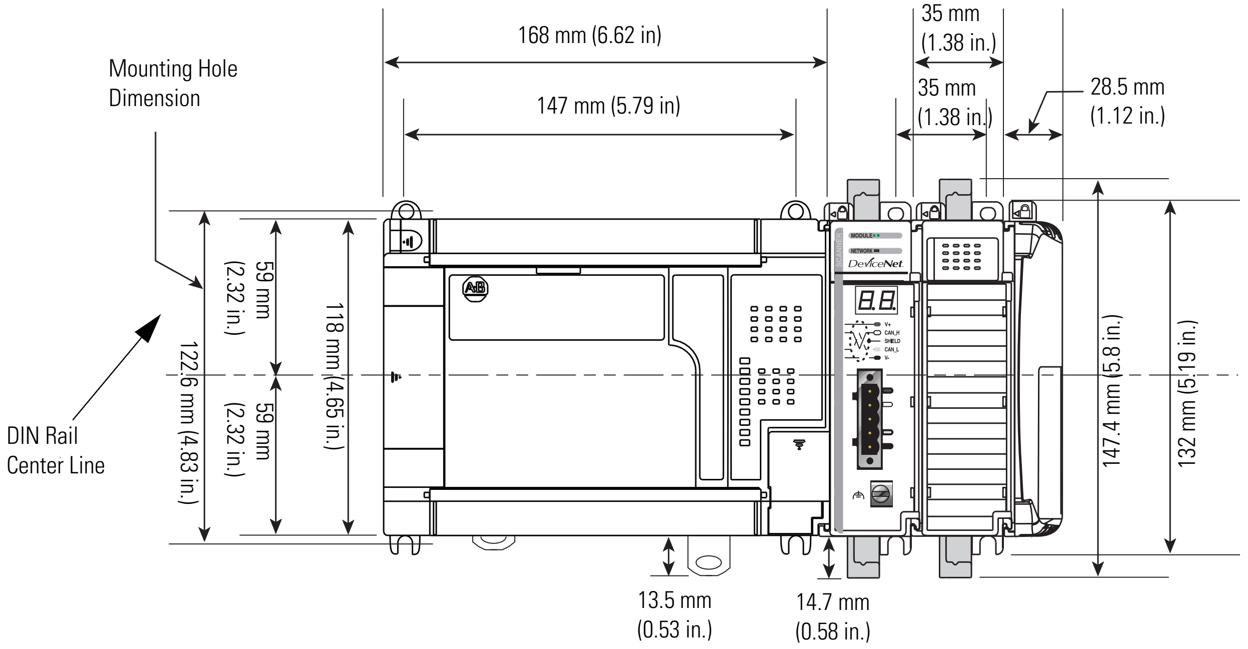
Panel Mounting Using the Dimensional Drawing

All dimensions are in millimeters (inches). Hole spacing tolerance: ±0.4 mm (0.016 in.).

Compact I/O System with L35E CompactLogix Controller and Power Supply



Compact I/O System with MicroLogix 1500 Base Unit and Processor



Panel Mounting Procedure Using Modules as a Template

This procedure lets you use the assembled modules as a template for drilling holes in the panel. Due to module mounting hole tolerance, it is important to follow these procedures.

1. On a clean work surface, assemble no more than three modules.
2. Using the assembled modules as a template, carefully mark the center of all module-mounting holes on the panel.
3. Return the assembled modules to the clean work surface, including any previously mounted modules.
4. Drill and tap the mounting holes for the recommended M4 or #8 screw.
5. Place the modules back on the panel, and check for proper hole alignment.
6. Attach the modules to the panel using the mounting screws.

   If mounting more modules, mount only the last one of this group and put the others aside. This reduces remounting time during drilling and tapping of the next group.

7. Repeat steps 1...6 for any remaining modules.

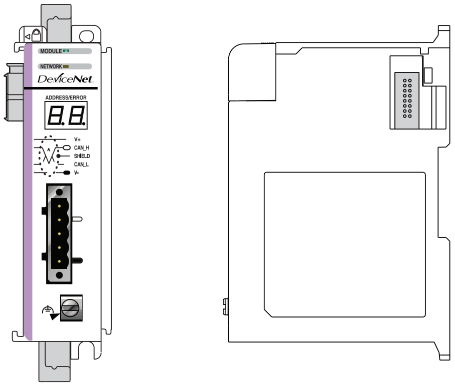
DIN Rail Mounting

The module can be mounted using the following DIN rails:

- 35 x 7.5 mm (EN 50022 - 35 x 7.5) or 35 x 15 mm (EN 50022 - 35 x 15).

Before mounting the module on a DIN rail, close the DIN rail latches. Press the DIN rail mounting area of the module against the DIN rail. The latches will momentarily open and lock into place. DIN rail mounting dimensions are shown below.

DIN rail mounting dimensions



Dimension	Height
A	118 mm (4.65 in.)
B	59 mm (2.325 in.)

Replace the Scanner Module within a System

The scanner module can be replaced while the system is mounted to a panel (or DIN rail) once power is removed.

1. Remove power.
2. Remove the DeviceNet cable from the scanner module by removing the DeviceNet connector.
3. Remove the upper and lower mounting screws from the scanner module (or open the DIN rail latches using a flat-blade screwdriver).
4. On the scanner module to be replaced and the right-side adjacent module (or end cap if the scanner module is the last module in the bank), move the bus levers to the right (unlock) to disconnect the scanner module from the adjacent modules.
5. Gently slide the disconnected scanner module forward.
6. If you feel excessive resistance, make sure that you disconnected the scanner module from the bus and that you removed both mounting screws (or opened the DIN latches).

   It may be necessary to rock the scanner module slightly from front to back to remove it, or, in a panel-mounted system, to loosen the screws of adjacent modules.
7. Before installing the replacement scanner, be sure that the bus lever on the right-side adjacent module is in the unlocked (fully right) position.
8. Slide the replacement scanner module into the open slot.
9. Connect the scanner module and modules together by locking (fully left) the bus levers on the replacement scanner module and the right-side adjacent module or end cap.
10. Replace the mounting screws (or snap the scanner module onto the DIN rail).
11. Replace the DeviceNet cable on the scanner module by attaching the connector to the scanner.
12. Restore the scanner module configuration using RSNetWorx for DeviceNet software.

IMPORTANT Be sure that the new module has the same node address and communication rate as the module that was replaced.

Field Wiring Connections

Follow these procedures to wire the scanner module.

Grounding the Scanner Module

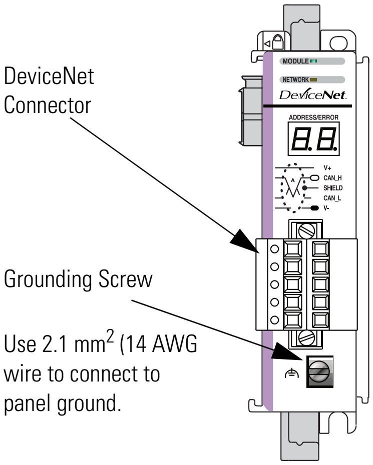
This product is intended to be mounted to a well-grounded mounting surface, such as a metal panel. Additional grounding connections from the scanner's mounting tabs or DIN rail (if used), are not required unless the mounting surface cannot be grounded.

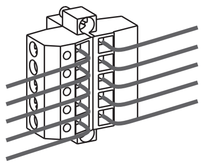
ATTENTION

The grounding screw on the front of the scanner module must be connected to a suitable ground source when operating in electrically noisy environments. Use a 2.1 mm^2 (14 AWG) wire to make this connection.

For additional information on grounding the scanner module, refer to Industrial Automation Wiring and Grounding Guidelines, Allen-Bradley publication 1770-4.1.

DeviceNet Wiring





Connect[1]	To
Red Wire	V+
White Wire	CAN High
Bare Wire	Shield
Blue Wire	CAN Low
Black Wire	V-

[1] DeviceNet cable colors are shown on the wiring label on the front of the scanner

1. Connect the DeviceNet cable to the removable connector as shown.
2. Insert the removable female connector into the mating male connector on the DeviceNet scanner module.
3. Screw the removable connector to the scanner module case with the upper and lower mounting screws to a torque of 0.6...0.7 Nm (5...6 in-lbs).

IMPORTANT	If the 1769-SDN scanner module is the first or last device connected to the DeviceNet network trunkline, be sure to add a termination resistor (120 Ω 1% ≥ ¼W resistor, Allen-Bradley part number 1485A-C2) across the Blue (CAN Low) and White (CAN High) wires.

Scanner Module Power-up

When power is applied via the Compact I/O bus, the scanner module goes through a self test sequence. Upon successful completion of the self test, the scanner module is ready to communicate.

The default scanner module settings are:

- Communication rate = 125 Kbps
- Node address = 63

Use your configuration software to change the communication rate and node address.

Notes:

Automatically Configure a DeviceNet Network

Introduction

This chapter provides a quick method for configuring a DeviceNet network. It uses the AutoScan feature to establish communication between the controller and your devices with minimal steps.

The DeviceNet AutoScan feature enables a scanner to automatically map a network of slave devices into its scanlist without the use of RSNetWorx for DeviceNet software. This greatly improves the ease of setting up a DeviceNet network, especially networks comprised of simple devices.

When the feature is enabled, a DeviceNet scanner continuously searches for devices on the network. Once a qualifying slave device is found, it is added to the scanner's scanlist and its I/O data is mapped into a predefined location in the scanner's I/O memory table based on the device's node address.

How AutoScan Operates

AutoScan is active when the feature is enabled and the scanner is in IDLE mode. When active, the scanner attempts to connect to each device not enabled in the scan list. The scanner only checks for devices with node addresses between 0 and 61, inclusive. The connections to these devices are made on a round robin basis.

When a device is found, the scanner gets the Produced and Consumed data sizes from the slave devices's Connection Object instance(s).

- If the Produced data size is greater than the configured I/O allocation size, the device is added to the scanlist with a produced size set equal to the I/O allocation size.

  When this happens, an I/O connection is made with the device. But it errors and error code #77 is displayed on the 1769-SDN for the device's node number.

- If the Consumed data size is greater than the configured I/O allocation size, then the node is rejected and not entered into the scan list.

However, you can change the I/O allocation size, as described in Configure I/O Allocation Size Via the User Program, to accommodate the device with the largest Produced and Consumed data sizes in your scanlist.

For qualifying nodes, the scanner enters the device into the scan list and attempts to allocate an I/O connection using one of the following communication format choices, in this particular order:

• Change Of State (COS)	EPR = 250ms
• Poll	EPR = 75ms
• Strobe	EPR = 75ms
• Cyclic	EPR = 500ms

EXAMPLE

If a photoeye was connected on a network that only supported strobed connections, the scanner does a couple of things.

- First, the scanner recognizes that a device exists for which memory was available for the node number with the configured allocation size on a network that was not currently mapped.

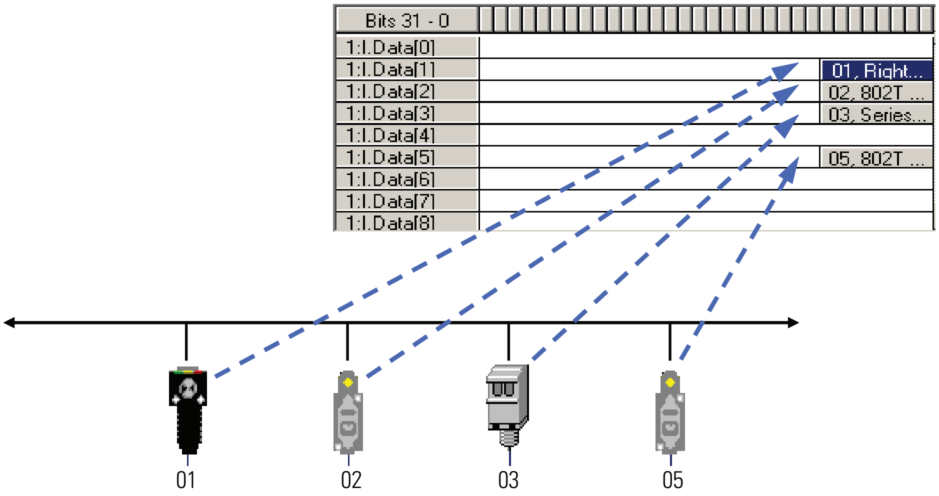
- Then, the scanner would attempt to initiate both COS and polled connections first, but the strobed connection would be selected as that is the only connection that the photoeye supported.

The input and output data is mapped into the scanner's I/O data table based on the device's node address and the configured fixed mapping size. The DINT-based formula is used with the CompactLogix controller for calculating the Input or Output data location is:

Input (Output) Offset = [(Node Address) x (Allocation Size)] / 4

EXAMPLE

When using the default fixed mapping size of 4 bytes, the input data for the devices shown in the example below is allocated in the 1769-SDN's input table as shown below. Notice node 1 is located in the data map at DINT location 1, node 2 at DINT location 2, and so on.



Bits 31 - 0	
1:I.Data[0]	
1:I.Data[1]	01, Right...
1:I.Data[2]	02, 802T ...
1:I.Data[3]	03, Series...
1:I.Data[4]	
1:I.Data[5]	05, 802T ...
1:I.Data[6]	
1:I.Data[7]	
1:I.Data[8]	

Notice that, in this example, node 4 is unused. However, the I/O memory slot remains allocated for it.

IMPORTANT

If you are using a MicroLogix 1500 controller with a 1769-SDN scanner, you must use the following WORD-based formula for calculating the Input or Output data location:

Input (Output) Offset = ([(Node Address) x (Allocation Size)] / 2) + Data Offset

In this formula the Data Offset = 66 for Input Offset and 2 for Output Offset.

The data offset value is used to account for scanners that have a fixed status field at the start of the input or output data, for example, the 1769-SDN scanner.

Determine If You Can Use AutoScan

Make sure your network meets the following requirements to use this chapter:

- The scanner's I/O allocation size is configured to accommodate the input and output data sizes of all devices on your DeviceNet network.

  The default AutoScan setting allocates a 4-byte entry in both the input and output memory maps in the scanner for each slave device detected on the network. This default size is chosen to accommodate the default Logix native data size of 32 bits, that is a DINT.

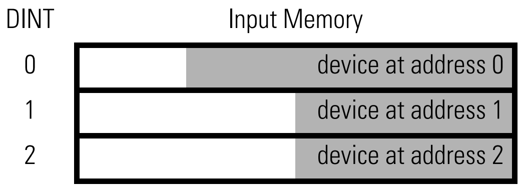
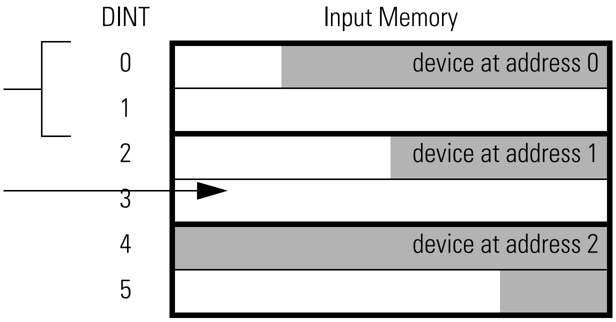
  If you use a device that sends more than 4 bytes of input or output data, for example, an E3 Solid State Overload Relay (catalog number 193-ECxx), you must change the I/O allocation size.

- You are using the CompactLogix 1769-SDN DeviceNet scanner with firmware revision 4.1 or greater.

If your network does not meet the requirements listed above, then use Chapter 5 and Chapter 7 to configure your network and control your devices.

How AutoScan Effects Your Network

As you use AutoScan, keep the following in mind:

Consideration	Description
AutoScan clears the current configuration.	With AutoScan, the scanner automatically sets up communication with the devices on your DeviceNet network. When you turn on the AutoScan option, the scanner removes any previous configuration that was done to the scanner.
AutoScan allocates a fixed memory size for each device.	At its default setting, AutoScan allocates 1 DINT of input memory and 1 DINT of output memory for each device on the DeviceNet network. The actual data for the device fills the portion that it needs and the rest remains unused.  DINT Input Memory 0 device at address 0 1 device at address 1 2 device at address 2
The bytes/node value defines how much memory for each address.	AutoScan lets you specify how much input and output memory to give to each address on your network. For example, if you specify 2 DINTs (8 bytes) per address, the scanner sets aside 2 DINTs for each address. The actual data for the device fills the portion that it needs and the rest remains unused.  DINT Input Memory 0 device at address 0 1 2 device at address 1 3 4 device at address 2 5

Consideration	Description
New devices are automatically available.	While the scanner is in idle mode, AutoScan continues to establish communication with devices that you connect to the network, as long as the device uses input data and output data sizes that fit in the scanner's I/O allocation size.
The Automatic Device Recovery (ADR) option is *not* available.	You have to use RSNetWorx for DeviceNet software to edit the configuration of the scanner to use the Automatic Device Recovery (ADR) option of a DeviceNet scanner. This turns off AutoScan.

Connect Each Device to the Network

As you connect your devices to the DeviceNet network, follow these guidelines:

1. Assign an address to each device. The following addresses are recommended but not required.

Give this address	To this device
0	scanner
1…61	your devices
62	hand held configurator, such as the DeviceNet Configuration Terminal, catalog number 193-DNCT
63	Leave open. Out of the box, a DeviceNet device is preset for address 63. Leaving address 63 open lets you get a new device on the network without conflicting with another device.

2. Connect the scanner and any network interface to the network.

By first connecting the scanner and/or network interface device to the network, you reduce the number of baud rate errors as you connect the rest of your devices:

- Scanners and network interface devices use a fixed baud rate.
- Sensors and similar DeviceNet devices use autobaud to set their baud rate. They wait for another device to communicate. Then they set their baud rate to the same baud rate as the other device.
- By first placing a scanner or network interface on the network, the other devices have a baud rate against which to set their baud rate.
- Initially, leave the baud rate of the scanner and network interface at the default setting of 125K bits/s. If you want to change the baud rate, wait until after you establish communication with all your devices at the default setting (125K).
- See Set the Node Address of a Device Via a DeviceNet Configuration Terminal on page 42 to set the scanner's DeviceNet address.

3. Connect the rest of your devices to the network one at a time.
   - Out of the box, a DeviceNet device is preset for address 63. Connect and set the devices one at a time to set the scanner's DeviceNet address. Otherwise the address conflicts may prevent communication with them.

   - If a device has a switch to set its baud rate, set the switch to autobaud, if available. Otherwise, set the device to the baud rate of the network.

   - After you change the address or baud rate of a device via a switch, cycle power to the device.

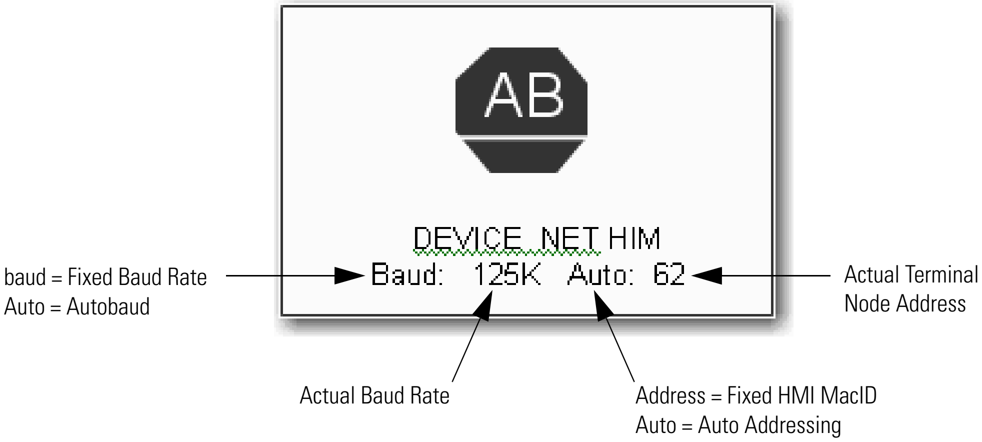
   - If a device has no switch or pushbutton for its address or baud rate, see Set the Baud Rate of a Device Via a DeviceNet Configuration Terminal below.

   - After you set the address of a device, check its network status indicator. Typically, a solid red indicator means an address conflict or problem with the baud rate.

Set the Baud Rate of a Device Via a DeviceNet Configuration Terminal

Follow these steps to set the baud rate for your DeviceNet network via the DeviceNet Configuration Terminal, catalog number 193-DNCT. For the rest of the chapter, the terminal is referred to as the 193-DNCT terminal.

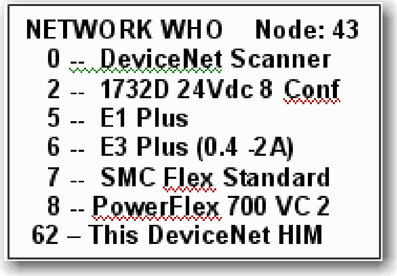
1. Connect the 193-DNCT terminal to the network. The following display appears for 10 seconds which shows the unit setup and the baud rate and node number values it has determined.



```
DEVICE NET HIM
Baud:  125K  Auto:  62
```

IMPORTANT The 193-DNCT terminal is shipped so that when it is placed on a DeviceNet network for the first time, it automatically sets its baud rate to that of the traffic on the network. The terminal uses Auto Addressing to assign itself an unused network node address.

After 10 seconds a **Network Who** screen similar to the example shown below should appear that displays all Nodes and associated devices on the network.



```
NETWORK WHO    Node: 43
  0 -- DeviceNet Scanner
  2 -- 1732D 24Vdc 8 Conf
  5 -- E1 Plus
  6 -- E3 Plus (0.4 -2A)
  7 -- SMC Flex Standard
  8 -- PowerFlex 700 VC 2
 62 – This DeviceNet HIM
```

If the **Network Who** screen does not appear as expected, then the 193-DNCT terminal may be set to **autobaud enabled** and is not able to determine a communication rate because no communication is occurring on the network at this time.

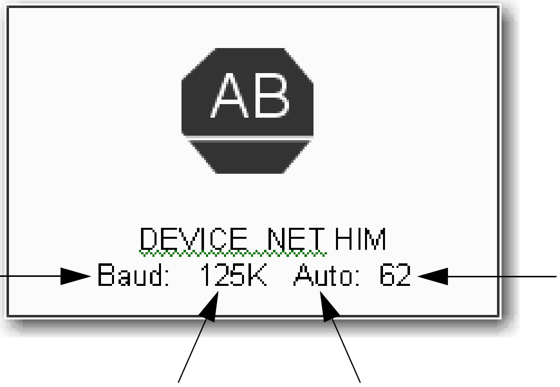
Follow these steps to disable the autobaud feature and set the baud rate.

1. On the 193-DNCT terminal keypad, press <ESC> ESC.
2. Select the **AutoBaud** option and use <Up Arrow> ▲ to select Disable.
3. Press <SEL> SEL to advance to the **BaudRate** option.
4. With the **BaudRate** option selected, use <Up Arrow> ▲ to select the appropriate baud rate.
5. Press <Enter> ↵ to complete the node commissioning function.
6. Press <ESC> ESC to exit setup.

Set the Node Address of a Device Via a DeviceNet Configuration Terminal

Follow these steps to set the node address of a device on your DeviceNet network via the 193-DNCT terminal.

1. Connect the 193-DNCT terminal to the network. The following display should appear for 10 seconds that shows the unit setup and the baud rate and node number values it has determined.



```
DEVICE NET HIM
Baud:  125K  Auto:  62
```

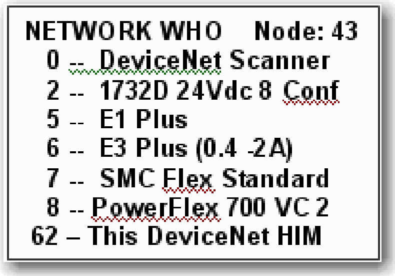
baud = Fixed Baud Rate
Auto = Autobaud

Actual Terminal
Node Address

Actual Baud Rate

Address = Fixed HMI MacID
Auto = Auto Addressing

After 10 seconds a Network Who screen similar to the example shown below should appear that displays all Nodes and associated devices on the network.



```
NETWORK WHO    Node: 43
  0 -- DeviceNet Scanner
  2 -- 1732D 24Vdc 8 Conf
  5 -- E1 Plus
  6 -- E3 Plus (0.4 -2A)
  7 -- SMC Flex Standard
  8 -- PowerFlex 700 VC 2
 62 – This DeviceNet HIM
```

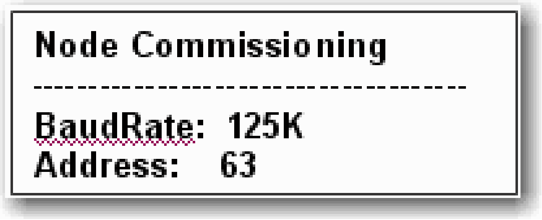
2. Use <Down Arrow> on the 193-DNCT terminal to scroll down through the list until the last entry for node 63 is highlighted, which should read 63 - No Product Name.

3. Press <Enter> to advance to the configuration screen.

4. Use <Down Arrow> to select the Tools option.

5. Press <Enter> ↵ and a screen appears with the **NodeComm** option highlighted.

6. Press <Enter> ↵ and the Node Commissioning screen appears as shown below with the BaudRate option highlighted.



```
Node Commissioning
------------------------------------
BaudRate:  125K
Address:    63
```

7. Because you do not need to change the baud rate, press <SEL> SEL to advance to the Address option and highlight it.

8. Press the numbers on the key pad for the node number that you plan to use for the device that was just installed on the DeviceNet network and press <Enter> ↵.

   When the address is changed an Apply Changes messages appears on the screen.

9. Press <SEL> SEL to highlight the **Apply Changes** message.

10. Press <Enter> ↵ to complete the node commissioning function.

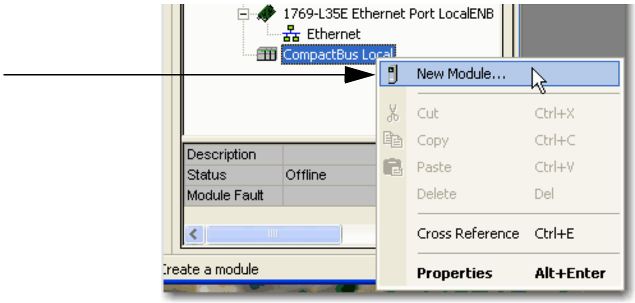
    After approximately two seconds, the 193-DNCT terminal re-initializes and in another 10 seconds, the 193-DNCT terminal again displays the Network Who screen. At this point, the new node appears in the table.

Add the Scanner to the RSLogix 5000 Project

Add the scanner to the I/O configuration of the controller to access the data of your network.

Add the Scanner to the I/O Configuration Folder

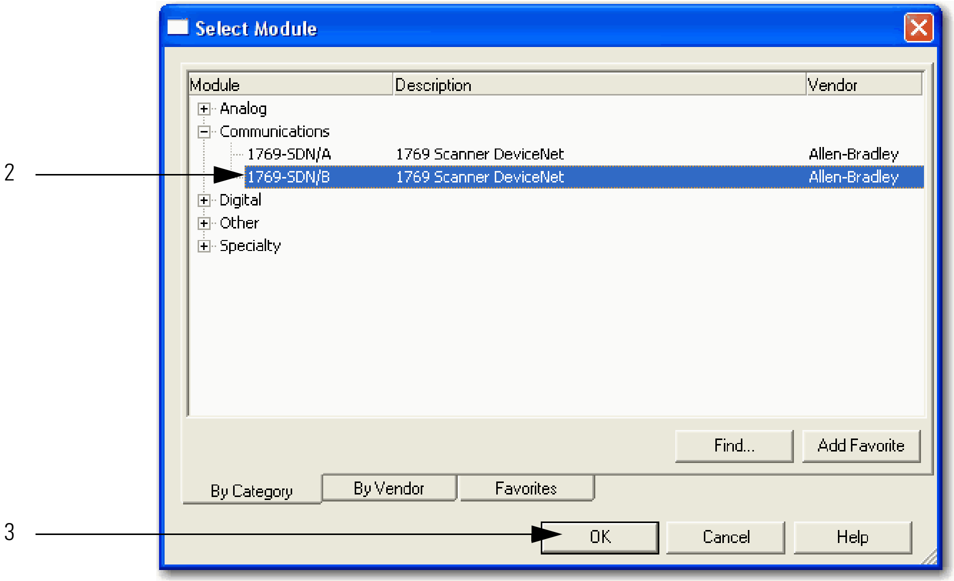
1. Right-click and choose New Module.



Description	
Status	Offline
Module Fault	

2. Choose the type of scanner.

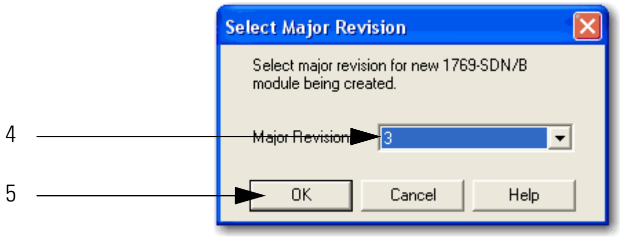
3. Click OK.



Module	Description	Vendor
⊞ Analog		
⊟ Communications		
1769-SDN/A	1769 Scanner DeviceNet	Allen-Bradley
1769-SDN/B	1769 Scanner DeviceNet	Allen-Bradley
⊞ Digital		
⊞ Other		
⊞ Specialty		

4. Select the Major revision of the scanner.

5. Click OK.



Select major revision for new 1769-SDN/B module being created.

Major Revision: 3

Define the Properties of the Scanner

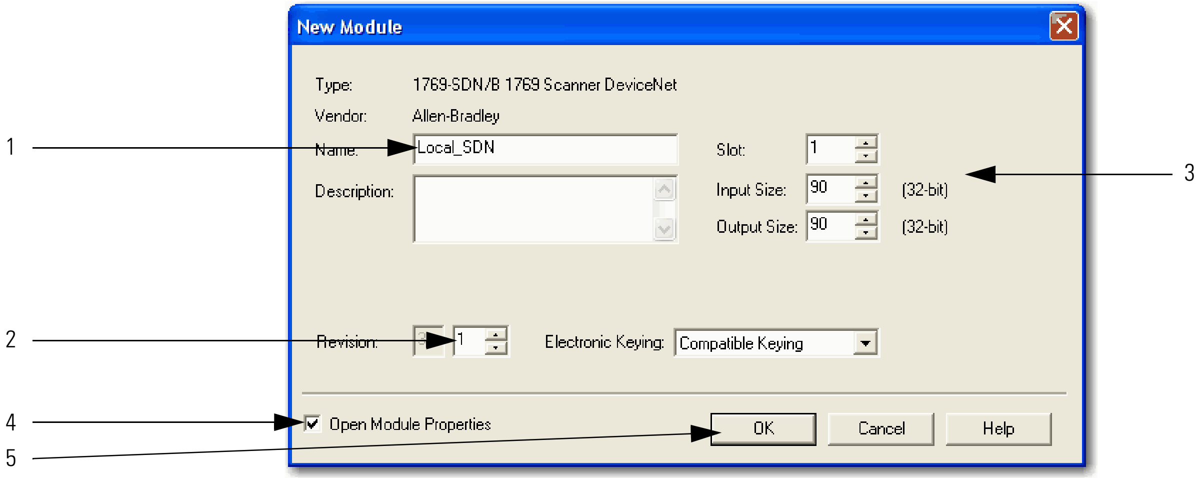
1. Name the scanner.

2. Set the scanner Minor Revision.

3. Choose the size of the input and output memory maps that the scanner will allocate for each device it detects on the network. Valid values range from 0 to 32 bytes per node.

4. If you need to make additional configuration changes, such as setting the Requested Packet Interval (RPI), leave the Open Module Properties box checked.

5. Click OK



Type: 1769-SDN/B 1769 Scanner DeviceNet

Vendor: Allen-Bradley

Name: Local_SDN

Description:

Slot: 1

Input Size: 90 (32-bit)

Output Size: 90 (32-bit)

Revision: 3 . 1

Electronic Keying: Compatible Keying

☑ Open Module Properties

6. When the Module Properties dialog appears, that is, if you left the Open Module Properties box checked, make additional required configuration changes.

Implement AutoScan

Make sure that the appropriate version of DeviceNet scanner is used to implement this feature. You must use the CompactLogix 1769-SDN DeviceNet scanner with firmware revision 4.1 or greater

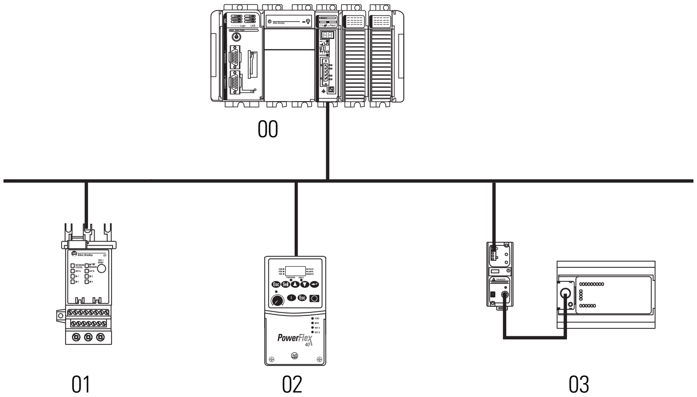
This section describes how to set up the feature and how it operates. Notice that explicit messaging is used for some of the steps. There are several ways that an explicit message can be sent on DeviceNet.

- A user ladder program
- External programming/configuration devices, such as the 193-DNCT terminal
- RSNetWorx for DeviceNet software

Since the purpose of the AutoScan feature is to eliminate the use of RSNetWorx for DeviceNet, instructions on how to send an explicit message via the class instance editor in RSNetWorx for DeviceNet are not covered in this document.

1. Set up the physical network. Make sure all devices are addressed appropriately, that is, there are no address conflicts, and are communicating at the same baud rate.

   The diagram below shows an example system using the 1769-SDN scanner.



   You can commission the node addresses via hardware switches on the device(s) or through the 193-DNCT terminal.

2. Set up I/O allocation size in the scanner.

   TIP This step is optional.

   The default AutoScan setting allocates a 4-byte entry in both the input and output memory maps in the scanner for each slave device detected on the network. This default size is chosen to accommodate the default Logix native data size of 32 bits (DINT). If that is adequate for the application, skip to step 3.

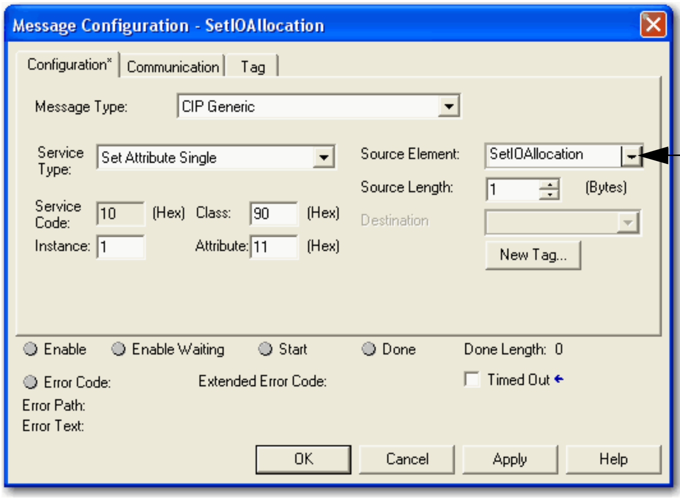
   For applications where the user would like to customize the I/O allocation size, the 4-byte allocation could be adjusted through an Explicit Message to the scanner using the SetAttributeSingle service. The entry allocation could be configured for 1 to 32 bytes per node. One way to set the allocation size is described in section Configure I/O Allocation Size Via the User Program.

The following table lists devices that most commonly use the AutoScan feature and their respective I/O allocation sizes.

Device	Bulletin Number	Input Allocation	Output Allocation
ArmorStart Distributed Motor Controller	280D/281D	1 byte	1 byte
193-E Electronic Overload Relays	193	8 bytes	1 byte
PowerFlex 40 Adjustable Frequency AC Drive	22B	4 bytes	4 bytes

Configure I/O Allocation Size Via the User Program

Use the parameters shown in the MSG message setup screen below to adjust the I/O allocation size. Make sure that the message is sent to the appropriate DeviceNet Scanner.



Message Configuration - SetIOAllocation

Configuration* | Communication | Tag

Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex) Class: 90 (Hex)

Instance: 1 Attribute: 11 (Hex)

Source Element: SetIOAllocation

Source Length: 1 (Bytes)

Destination

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:

This data tag should be configured as an SINT, and should contain the value of the desired per-node fixed mapping size (1 - 32)

IMPORTANT You can only change the I/O allocation size when:

- the scanner is in IDLE mode, and
- the AutoScan feature is disabled.

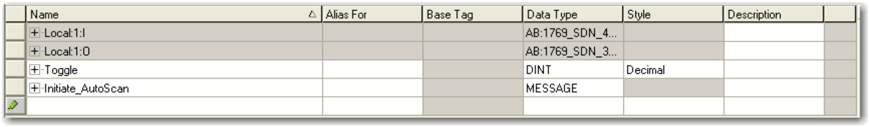
3. Execute an Explicit message to the scanner using the SetAttributeSingle service to enable AutoScan. There are multiple ways to send an explicit message on DeviceNet, including the following:
   - Initiate AutoScan Via the User Program
   - Initiate AutoScan via the 193-DNCT Terminal

Initiate AutoScan Via the User Program

Follow these steps to initiate AutoScan using a MSG instruction.

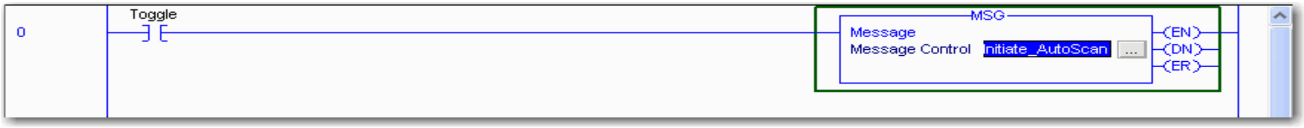
1. Verify that the scanner is in IDLE mode, that is bit 0 in the scanner control output word = 0, and that all slave nodes are connected to and communicating on the DeviceNet network.

2. In the RSLogix 5000 programming software Tag Editor, create new tags to initiate AutoScan via ladder programming as shown in the example below.



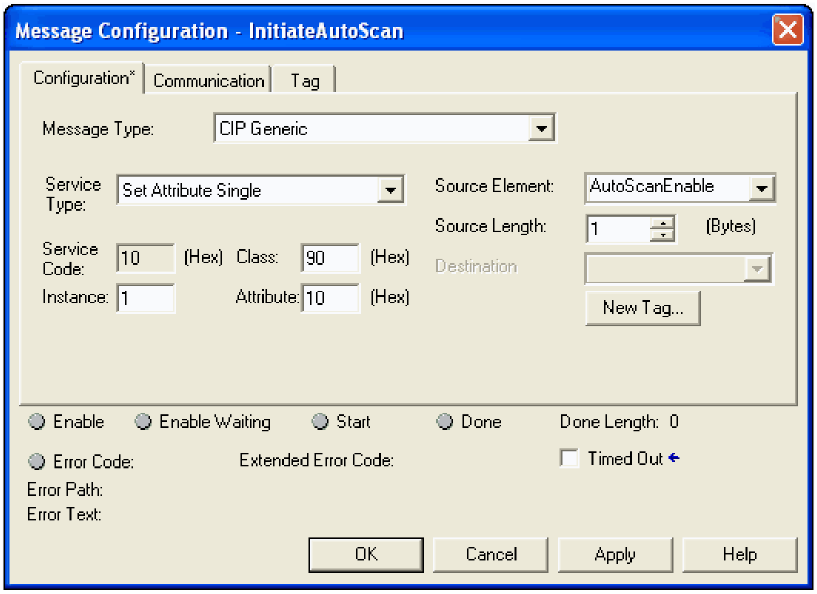
Name	Alias For	Base Tag	Data Type	Style	Description
+ Local:1:I			AB:1769_SDN_4...		
+ Local:1:O			AB:1769_SDN_3...		
+ Toggle			DINT	Decimal	
+ Initiate_AutoScan			MESSAGE		

3. Create a MSG instruction that uses the new tags.



Message
Message Control Initiate_AutoScan

4. Configure the MSG instruction to use the proper parameters.



Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex) Class: 90 (Hex)

Instance: 1 Attribute: 10 (Hex)

Source Element: AutoScanEnable

Source Length: 1 (Bytes)

Destination

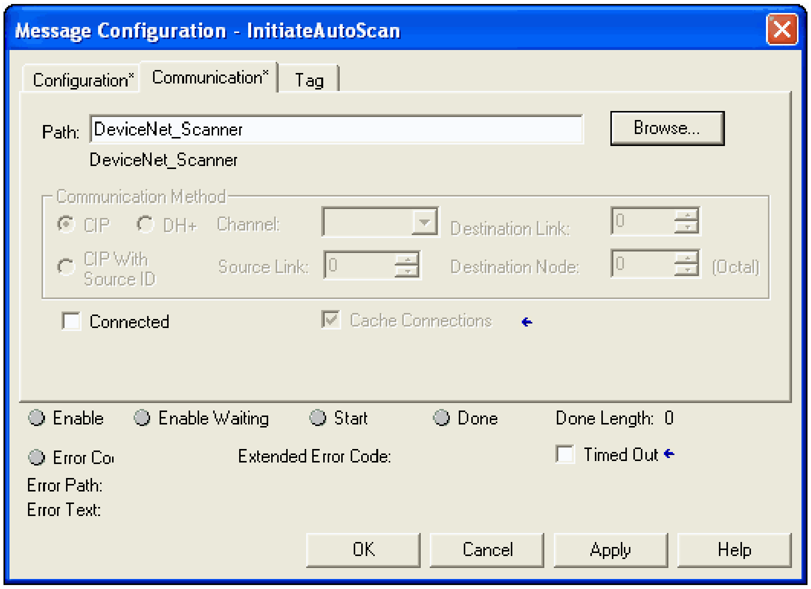
○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:

5. Set the path to your DeviceNet scanner.



Message Configuration - InitiateAutoScan

Configuration* | Communication* | Tag

Path: DeviceNet_Scanner

DeviceNet_Scanner

Communication Method

● CIP ○ DH+ Channel: Destination Link: 0

○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

□ Connected ☑ Cache Connections

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Co Extended Error Code: □ Timed Out

Error Path:

Error Text:

6. Return the scanner to Run mode, that is bit 0 in the scanner control output word = 1, and the scanner status display should display the scanner node address.

The previous example is from RSLogix 5000 programming software; refer to the appropriate user manuals to determine how to perform explicit messaging in other PLC platforms.

Initiate AutoScan via the 193-DNCT Terminal

Follow these steps to enable AutoScan using the 193-DNCT terminal, version 2.1 or higher.

1. Put your controller into Program mode, and verify that the scanner is in Idle mode (CommandRegister.Run = 0) and that all slave nodes are connected and communicating on the DeviceNet network.
2. Plug the 193-DNCT terminal into the DeviceNet network.
3. In the Network Who dialog, press <Up Arrow> ▲ to navigate to and select the first line (0 - 1769-SDN DeviceNet Scanner).
4. Press <Enter> ↵ to go to the scanner's configuration dialog.
5. Press <Down Arrow> ▼ to navigate to Scanner and press <Enter> ↵.
6. Press <Down Arrow> ▼ to navigate to AutoScan and press <Enter> ↵.

   The AutoScan Setup dialog appears.
7. If AutoScan is selected and set to Enable, press <Up Arrow> ▲ to change the setting to Disable.
8. Press <SEL> SEL to scroll down to Save. This disable the AutoScan feature.
9. Press <SEL> SEL to scroll down to Mapping.
10. Type in the number of bytes that you want to automap to each DeviceNet node, that is, 1…32. The default is 4 bytes.
11. Press <SEL> SEL to scroll down to Save, and press <Enter> ↵ to commit your byte size selections.

    The AutoScan feature is now configured with the byte size entered in step 9 and step 10. For the 1769-SDN scanner to remap the network with this new setting, the AutoScan feature must be re-enabled.

12. Change AutoScan setting to Enable and press <SEL> SEL to scroll down to Save.

IMPORTANT	When setting the AutoScan mode from Disable to Enable, the 1769-SDN scanner clears all existing configuration.

Notice that the 1769-SDN module status indicator flashes red and back to green, while the status display momentarily shows 72.

When the AutoScanning is complete, the status display blinks between 65 and the node address of the scanner.

13. On the AutoScan set-up screen, press either the <Up Arrow> ▲ or <Down Arrow> ▼

14. Press <SEL> SEL twice and then press <Enter> ↵ to save your changes.

15. Press <ESC> ESC three times to return to the Network Who screen on the 193-DNCT terminal.

If all the devices on the network are properly mapped, the scanner status display blinks back and forth between 80 and the node address of the scanner.

IMPORTANT	With 193-DNCT terminal, version 2.1 and higher, the active nodes line on the Network Who screen should show a number that exactly matches how many slave devices are on the DeviceNet network. If the number on the active nodes line doe not match the number of slave devices on the network, the cause typically is one of the following: • The consume I/O data size of a device is greater than the selected I/O allocation size. • The slave device is not on the network.

16. Put the controller into Run mode and verify that the scanner is in Run mode (CommandRegister.Run is 1) in the scanner control output word = 0.

The scanner status should display only the node address of the scanner. If it is flashing other codes as well, refer to Chapter 8, Troubleshooting for more information.

Additional Considerations Regarding AutoScan

The factory default setting for AutoScan is disabled for all products.

Make sure that input or output data memory size in the scanner is large enough to accommodate the size required based on the number of nodes on the network and the AutoScan I/O allocation size per node.

TIP

Nodes 0…61 inclusive are scanned and added to the scanlist if they are not already mapped and I/O space exists. For maximum capacity for slave devices, the scanner node address could be configured as node 62; node 63 could also be used but to avoid duplicate node address issues with a new device, this is not recommended.

IMPORTANT

According the ODVA DeviceNet Technical Overview, the maximum number of nodes on a DeviceNet network is 64. However, the normal useful number of slave nodes is 62 because:

- the DeviceNet scanner requires a node

and

- we recommend that you leave node 63 unused for devices that are added to the network in the future.

If a network interface is used on the network, and the above bullets are taken into consideration, then the useful number of nodes becomes 61.

EXAMPLE

If the I/O allocation size per node is configured for 16 bytes and there are 32 slave devices on the network (node addresses 1 - 32), AutoScan requires 16 bytes x 32 = 512 bytes (128 DINT) of I/O space in both the scanner's input and output table. Assuming it is a CompactLogix system, the maximum scanner input data table size is 90 DINT and 90 DINT for output. The required space exceeds what the 1769-SDN can support. The user would need to adjust the I/O allocation size or reduce the slave device count on the network to include all of the devices in the scanlist.

Devices outside of the scanner's allowable I/O image space will be rejected and will not be included in the scanlist.

See Scanlist on page 54 for an accurate maximum node count by allocation size.

Scanlist

Data Allocation (bytes)	Max Node
1	61
2	61
3	61
4	61
5	61
6	60
7	51
8	45
9	40
10	36
11	32
12	30
13	27
14	25
15	24
16	22
17	21
18	20
19	18
20	18
21	17
22	16
23	15
24	15
25	14
26	13
27	13
28	12
29	12
30	12
31	11
32	11

The AutoScan feature is automatically disabled in the scanner as soon as the scanner configuration is modified by RSNetWorx for DeviceNet software. For example, any manual changes to the scanlist using RSNetWorx for DeviceNet software disables the AutoScan feature in the scanner.

One new status code has been added to the Node Status list. This code is presented in the Node Status Table.

Status Code (Decimal)	Description of Status
65	AutoScan Active (Scanner only status)

When the scanner is in Run mode with AutoScan enabled, the scanner display alternates between 65 and the scanner node address.

When a scanner is transitioned from Run mode to Idle mode while AutoScan is enabled, it only scans the network for nodes that are not already in the scanlist. However, while in Idle mode, an AutoScan DISABLE to ENABLE transition causes the scanner to erase the existing scanlist and scan for all nodes on the network.

The AutoScan feature checks for the Quick Connect setting in each slave device and enables Quick Connect in the scanner if it is enabled in the slave devices.

Type of Connection that the Scanner Sets Up

The type of update (connection) that the scanner sets up with each device depends on the device. The scanner chooses the first connection type that the device supports in this order:

1. change-of-state (COS)
2. polled
3. strobed
4. cyclic at 1000 ms

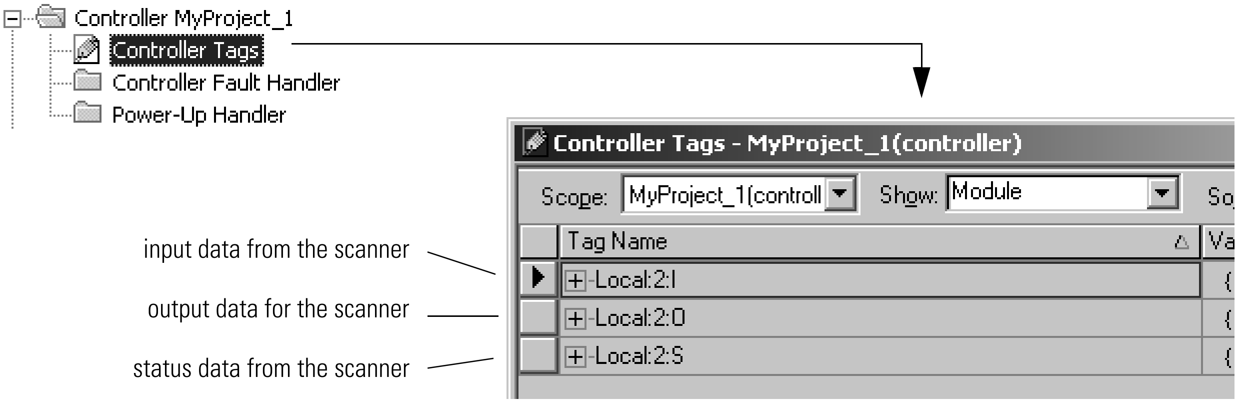
The scanner tries to set up a change-of-state connection. If the device doesn’t support change-of-state, then the scanner tries to set up a polled connection, etc. The type of connection that the scanner sets up may *not* be the default for the device.

As an option, you can allocate more memory for each device.

Consideration:	Description:
The scanner sets-up communication with any device that fits within the allocated memory size.	The scanner automatically sets up communication with those devices that fit within the memory allocated for each address. • For example, if you allocate 2 DINTs (8 bytes) per address, the scanner sets up communication with any device that sends or receives 0 - 8 bytes of data. • The scanner adds as many device as it can until it runs out of memory. • If you give too much memory to each address, you may not have enough memory for all your devices.
The scanner skips devices that are too large.	If a device needs more memory than is allocated, the scanner skips it and *does not* set up communication with it. **IMPORTANT**: If the Produced data size is greater than the configured I/O allocation size, the device is added to the scanlist with a produced size set equal to the I/O allocation size. When this happens, an I/O connection is made with the device. But it errors and error code #77 is displayed on the 1769-SDN for the device’s node number. For example, if you specify 2 DINTs (8 bytes) per address but a device sends 9 bytes, the scanner *does not* add the device to the scan list.
Manually editing the scan list turns off AutoScan.	If you use RSNetWorx for DeviceNet software to edit the configuration of the scanner, the scanner turns off AutoScan. *Do not* turn it back on or you will clear the configuration that you just entered. For example, if you use RSNetWorx for DeviceNet software to manually add a device to the scan list, the scanner turns off AutoScan. If turn on AutoScan again, the scanner clears out the current configuration and starts over.

Access Device Data

When you add the scanner to the I/O configuration of the controller, RSLogix 5000 software automatically creates a set of tags for the input, output, and status data of the network:


Controller MyProject_1
Controller Tags
Controller Fault Handler
Power-Up Handler
Controller Tags - MyProject_1(controller)
Scope: MyProject_1(controll
Show: Module
So
Tag Name
Va
+-Local:2:I
+-Local:2:O
+-Local:2:S
input data from the scanner
output data for the scanner
status data from the scanner

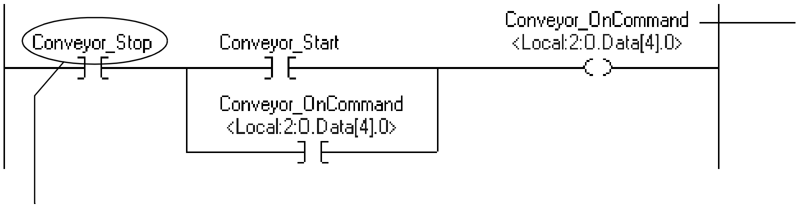
The tags for your DeviceNet data follow this format:

location	:type	.Data	IO_offset	▨ .bit

▨ = Optional

Where	Is		
location	location of the scanner in the system		
	If you have the 1769-SDN scanner in a		Then location is
		local chassis	Local:slot_number_of_scanner
		remote chassis	adapter:slot_number_of_scanner where: adapter is the name of the EtherNet/IP or ControlNet module in the remote chassis.
type	type of data:		
	Where	Is	
	input from a device	I	
	output to a device	O	
IO_offset_address	address of the device on the DeviceNet network (based on 4 bytes per node)		
bit	specific bit within the data of the device		

While you can use the input and output tags of the scanner directly in your logic, it is a lot easier to use alias tags.



As an option, create tags that describe each device without pointing them to the actual addresses of the devices. Later, convert the tags to aliases for the data of the devices.

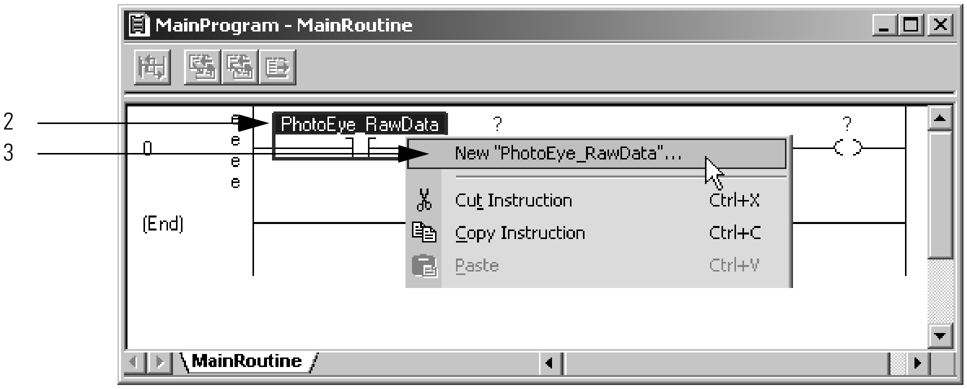
alias tag – a tag that represents another tag

- Both tags share the same data.
- When the data changes, both tags change.
- An alias tag provides a descriptive name for data, such as DeviceNet input or output data.
- If the location of the data changes, simply point the alias tag lets to the new location without editing your logic.

1. Enter your logic.

2. Type a descriptive tag name for the DeviceNet data.

3. Right-click the tag name and choose New…

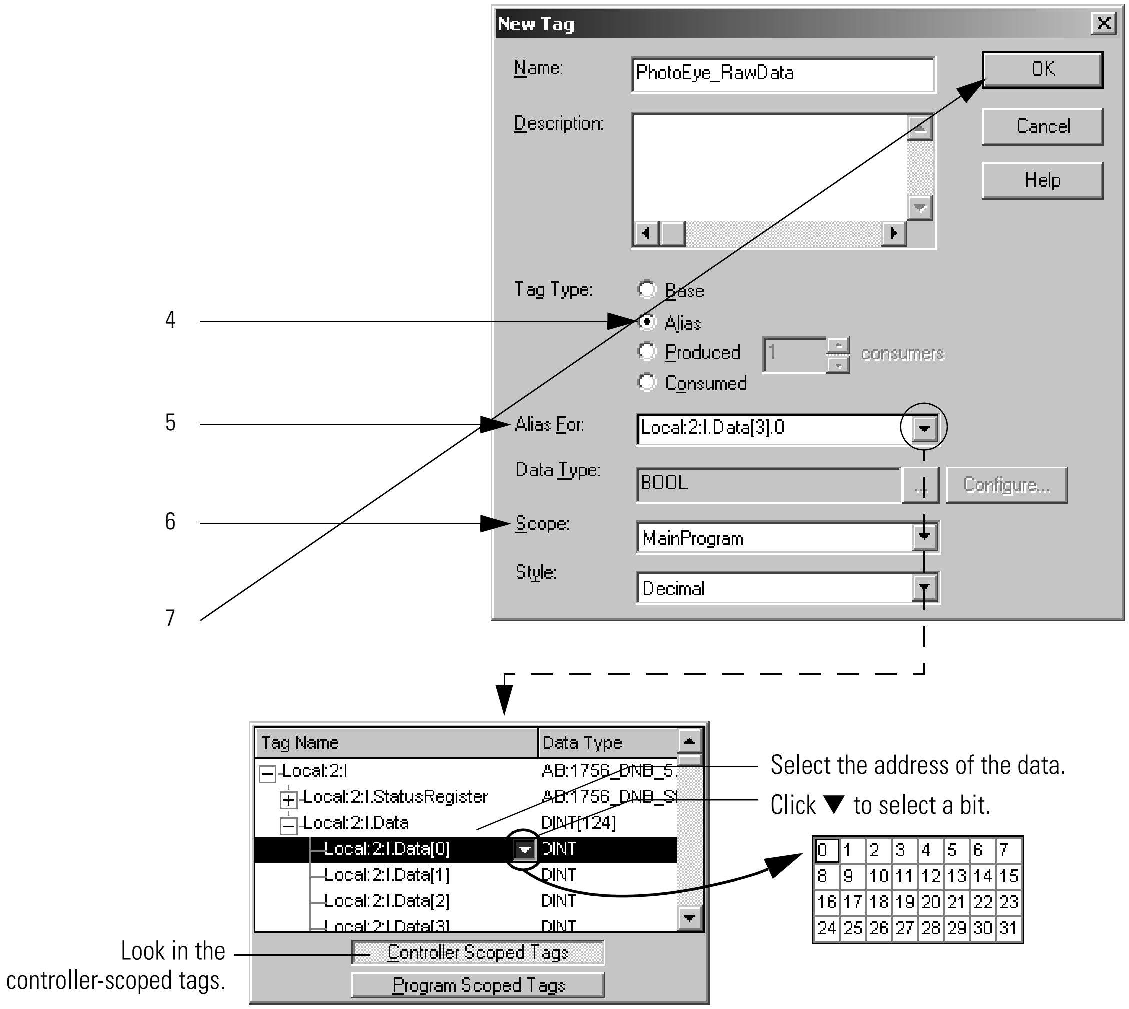


4. Select the Alias button.

5. Select the tag that this alias tag represents.

6. Select the scope for the alias tag.

7. Choose OK.



Name: PhotoEye_RawData

Description:

Tag Type: ○ Base ● Alias ○ Produced 1 consumers ○ Consumed

Alias For: Local:2:I.Data[3].0

Data Type: BOOL

Scope: MainProgram

Style: Decimal

Tag Name	Data Type
⊟-Local:2:I	AB:1756_DNB_5:
⊞-Local:2:I.StatusRegister	AB:1756_DNB_St
⊟-Local:2:I.Data	DINT[124]
Local:2:I.Data[0]	DINT
Local:2:I.Data[1]	DINT
Local:2:I.Data[2]	DINT
Local:2:I.Data[3]	DINT

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31

Select the address of the data.

Click ▼ to select a bit.

Look in the controller-scoped tags.

Put the Scanner in Run Mode

Follow these steps to run the DeviceNet network.

1. Place the controller in Run/Remote Run mode.

2. Set the following bit of the output structure for the scanner.

If you want to	Set this bit	To:
run the network	…O.CommandRegister.Run	1
not run the network (idle mode)	…O.CommandRegister.Run	0

3. Check to see if the scanner is in Run mode, by making sure an 80 status code does not appear on the scanner's display.

Manually Configure the DeviceNet Network

Introduction

This chapter describes how to configure the DeviceNet network using RSLinx and RSNetWorx for DeviceNet software. This table describes what this chapter contains and where to find specific information.

Topic	Page
Software Versions	61
Install the Software	62
Use RSLinx Software to Configure Your DeviceNet Driver	62
Use RSNetWorx for DeviceNet Software to Configure the 1769-SDN Scanlist	64

Software Versions

This table lists the software and revision level required to operate with the 1769-SDN scanner module.

Function	Software	Version
Communication	RSLinx	2.30 or later
DeviceNet configuration	RSNetWorx for DeviceNet	3.00 or later
Ladder logic programming software	RSLogix 500	5.00.10 or later
	RSLogix 5000	8.02 or later

You will also need one of the following interfaces to use your computer to communicate with the DeviceNet network:

- 1770-KFD RS-232 DeviceNet adapter
- 1784-PCD or 1784-PCID DeviceNet interface card

If your RSNetWorx for DeviceNet software does not include the required Electronic Data Sheet (EDS) file, go to http://www.ab.com/networks/eds.

Register the new EDS file using the EDS Wizard in RSNetWorx for DeviceNet software. Access the wizard from the Tools menu.

Install the Software

Follow these steps to install RSLinx and RSNetWorx for DeviceNet software

1. Insert the software CD-ROM in the drive.

   The CD-ROM supports Windows Autorun. If you have Autorun configured, the installation will automatically start when you insert the CD-ROM in your drive. If you do not have Autorun configured, perform steps 2 and 3.

2. From the Windows Start menu, choose Run.

3. Browse for the Setup program on the CD-ROM and open it**.**

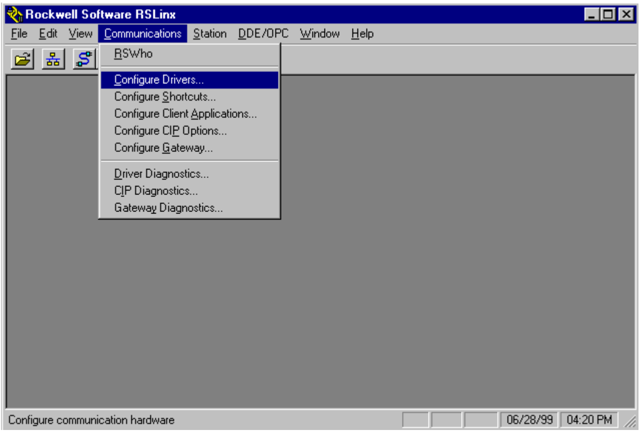
4. Follow the prompts that appear as you install the software.

After software installation is complete, you will use RSLinx software to configure your DeviceNet driver and RSNetWorx for DeviceNet software to configure the network.

Use RSLinx Software to Configure Your DeviceNet Driver

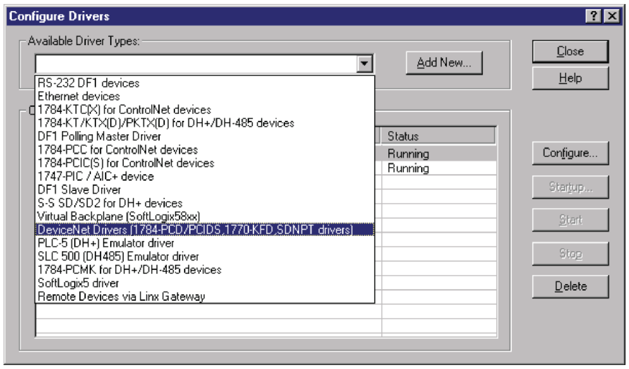
Follow these steps to configure the DeviceNet driver.

1. Start RSLinx software.



2. From the Communications menu, choose Configure Drivers.

3. From the Available Driver Types pull-down menu, choose Add/New.



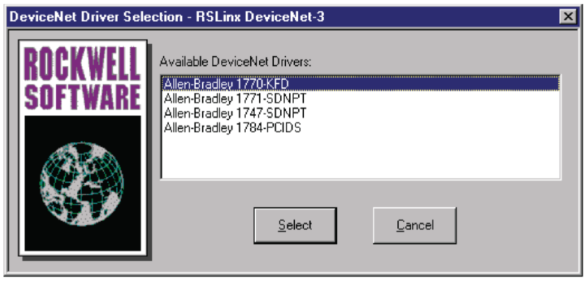
Configure Drivers

Available Driver Types:

- RS-232 DF1 devices
- Ethernet devices
- 1784-KTC(X) for ControlNet devices
- 1784-KT/KTX(D)/PKTX(D) for DH+/DH-485 devices
- DF1 Polling Master Driver
- 1784-PCC for ControlNet devices
- 1784-PCIC(S) for ControlNet devices
- 1747-PIC / AIC+ device
- DF1 Slave Driver
- S-S SD/SD2 for DH+ devices
- Virtual Backplane (SoftLogix58xx)
- DeviceNet Drivers (1784-PCD/PCIDS,1770-KFD,SDNPT drivers)
- PLC-5 (DH+) Emulator driver
- SLC 500 (DH485) Emulator driver
- 1784-PCMK for DH+/DH-485 devices
- SoftLogix5 driver
- Remote Devices via Linx Gateway

Status
Running
Running

4. .Select your driver.

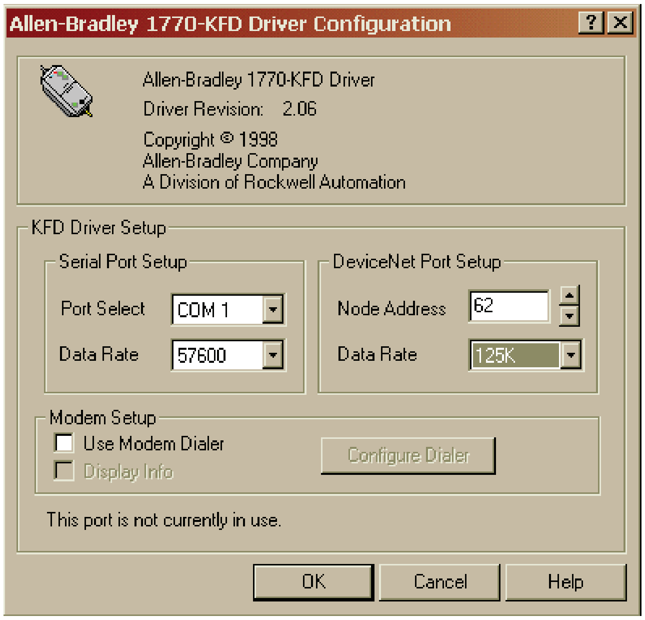


DeviceNet Driver Selection - RSLinx DeviceNet-3

Available DeviceNet Drivers:

- Allen-Bradley 1770-KFD
- Allen-Bradley 1771-SDNPT
- Allen-Bradley 1747-SDNPT
- Allen-Bradley 1784-PCIDS

5. Your driver setup will depend on your particular system setup (COM port, communication rate, node address). Choose the appropriate settings for your system.



Allen-Bradley 1770-KFD Driver Configuration

Allen-Bradley 1770-KFD Driver
Driver Revision: 2.06
Copyright © 1998
Allen-Bradley Company
A Division of Rockwell Automation

KFD Driver Setup

Serial Port Setup
Port Select: COM 1
Data Rate: 57600

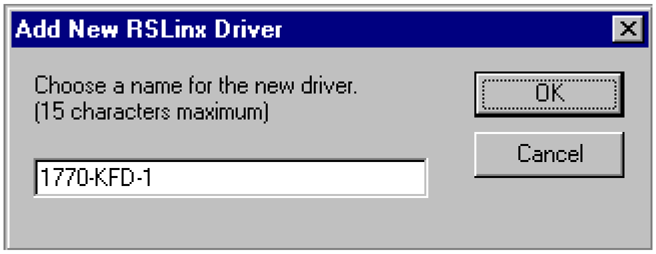
DeviceNet Port Setup
Node Address: 62
Data Rate: 125K

Modem Setup
□ Use Modem Dialer
□ Display Info

This port is not currently in use.

The software will take a few seconds to configure the driver.

6. .Click OK.



Add New RSLinx Driver

Choose a name for the new driver.
(15 characters maximum)

1770-KFD-1

7. Click OK to use the default driver name.

8. Minimize RSLinx software.

You will use the driver you just configured to browse and configure the network with RSNetWorx for DeviceNet software.

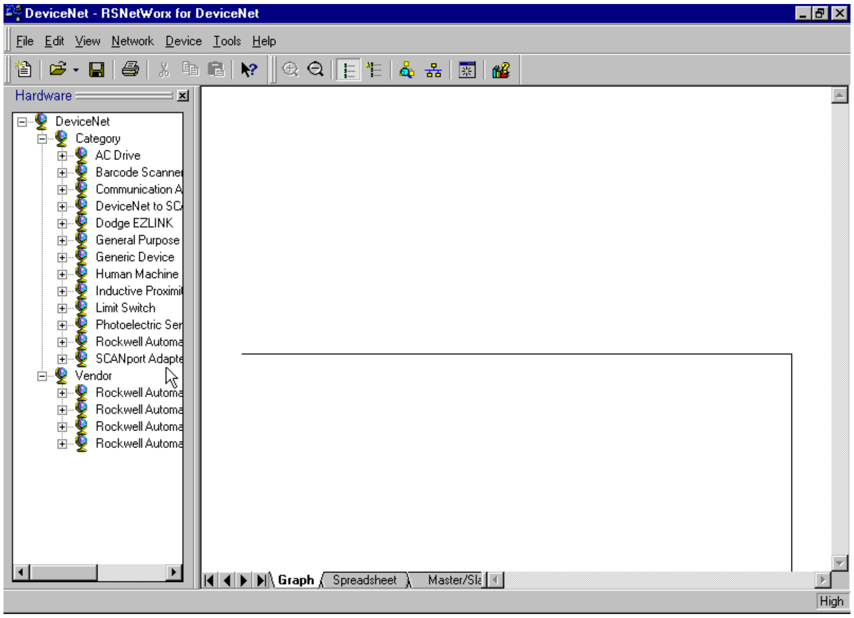
Use RSNetWorx for DeviceNet Software to Configure the 1769-SDN Scanlist

This manual assumes a certain level of familiarity with RSNetWorx for DeviceNet software. For detailed information on RSNetWorx for DeviceNet software, refer to your software's documentation.

Set Up an Online Connection

Follow the procedure below to set up an online connection to the DeviceNet network using the 1770-KFD driver.

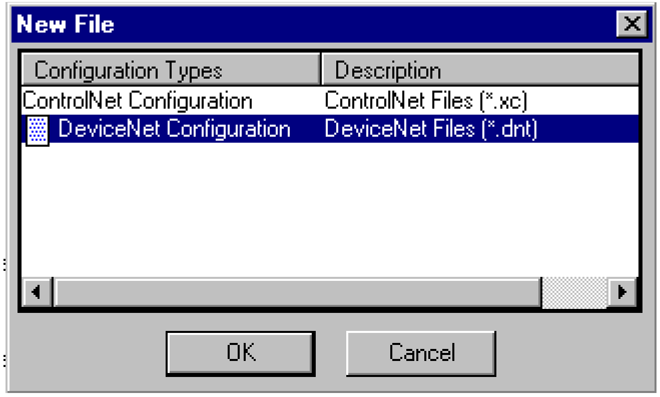
1. Start RSNetWorx for DeviceNet software.



DeviceNet - RSNetWorx for DeviceNet

2. From the File menu, choose New.

   If you have ControlNet configured on your system you may see this dialog. Otherwise, proceed to step 4.

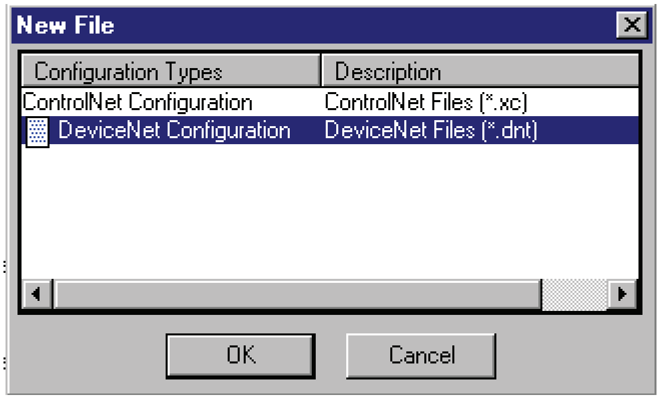


Configuration Types	Description
ControlNet Configuration	ControlNet Files (*.xc)
DeviceNet Configuration	DeviceNet Files (*.dnt)

3. Select DeviceNet Configuration and click OK.

4. Click the Online button on the toolbar.

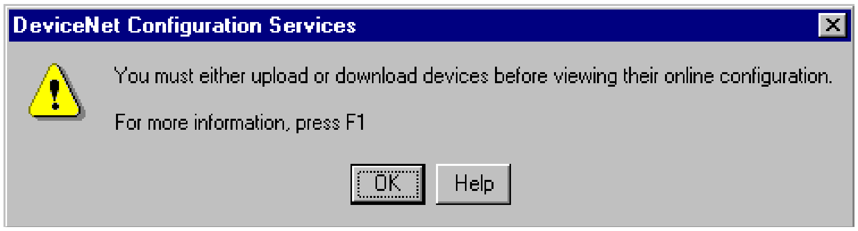
   A list of the available networks will appear. Your list may be different from that shown below, depending upon the drivers you have configured on your system.



Configuration Types	Description
ControlNet Configuration	ControlNet Files (*.xc)
DeviceNet Configuration	DeviceNet Files (*.dnt)

5. Select your DeviceNet driver and click OK.

   You will be prompted to upload or download devices before going online.

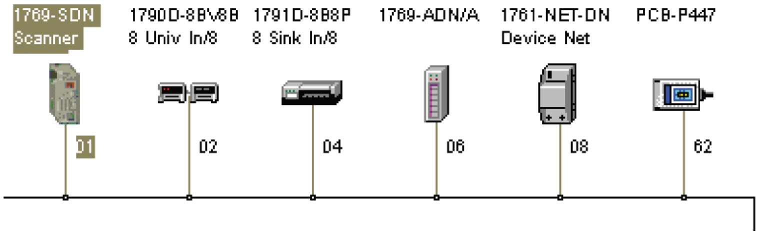


You must either upload or download devices before viewing their online configuration.

For more information, press F1

6. Click OK to upload the devices and go online.

RSNetWorx for DeviceNet software begins browsing for network devices. When the software is done browsing, your network is displayed.



RSNetWorx for DeviceNet software performs a single pass browse when you go online or choose the browse feature. The software will poll for devices one time and display the results. If a node that was online later goes offline, there will be no live indication in RSNetWorx for DeviceNet software. You must manually perform a browse to detect the missing node.

Click the button to manually perform the browse.

If RSNetWorx for DeviceNet software fails to find a device, check the physical connection to the device. If the physical connection is intact, verify that the device's communication rate is the same as the communication rate of the DeviceNet driver.

If your RSNetWorx for DeviceNet software does not include the required electronic data sheet (EDS) file, go to http://www.ab.com/networks/eds.

Use the EDS Wizard in RSNetWorx for DeviceNet software to register the new EDS file. Access the wizard from the Tools menu.

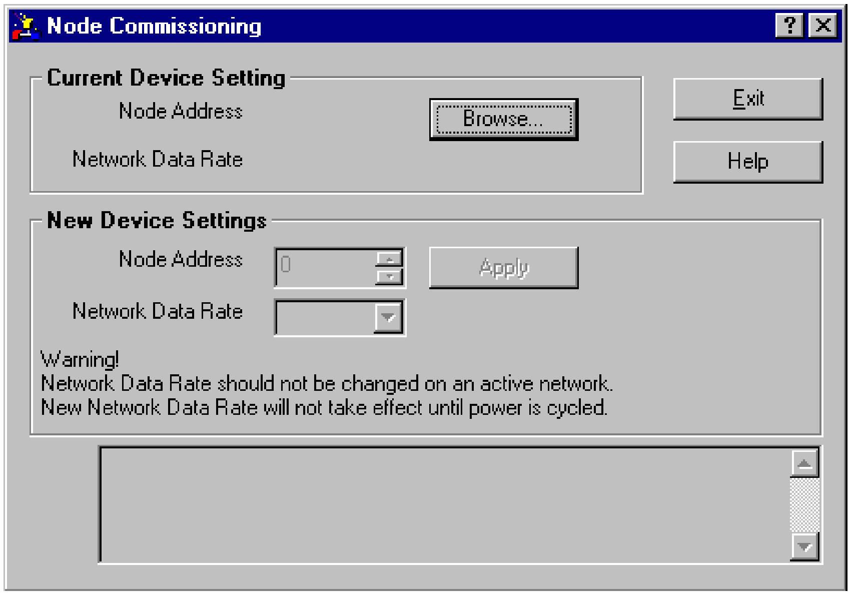
Set the Node Address

Once the network browse is complete, the node addresses appear to the right of their icons.

Use this procedure to change the node address of devices on the network (such as a photoeye). You can also change the network communication rate (baud rate) of some devices. Depending upon the device, you may need to cycle power for communication rate changes to take effect.

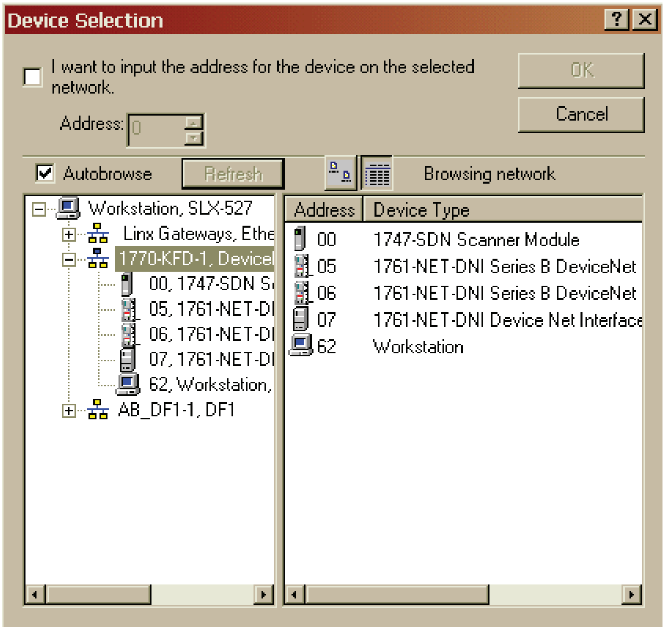
The controller must be in Program mode, or the scanner module in Idle mode (bit 0 of the Module Command Array = 0), for the scanner module to accept the configuration information.

1. From the Tools menu, choose Node Commissioning.



Warning!
Network Data Rate should not be changed on an active network.
New Network Data Rate will not take effect until power is cycled.

2. Click Browse.



Device Selection

□ I want to input the address for the device on the selected network.

Address: 0

☑ Autobrowse

Browsing network

Workstation, SLX-527
Linx Gateways, Ethe
1770-KFD-1, Devicel
00, 1747-SDN S
05, 1761-NET-DI
06, 1761-NET-DI
07, 1761-NET-DI
62, Workstation,
AB_DF1-1, DF1

Address	Device Type
00	1747-SDN Scanner Module
05	1761-NET-DNI Series B DeviceNet
06	1761-NET-DNI Series B DeviceNet
07	1761-NET-DNI Device Net Interface
62	Workstation

3. Select the DeviceNet network.

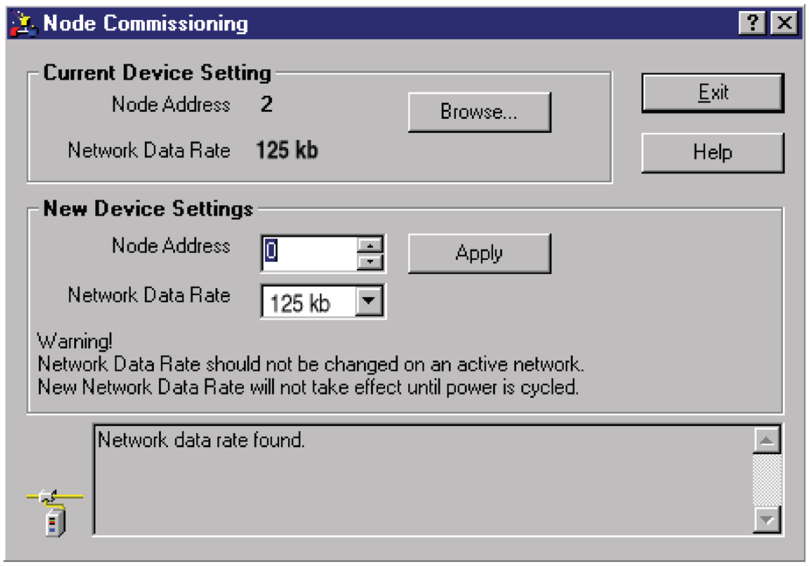
The devices on the network will appear in the right panel of the dialog.

4. Select the device you are commissioning in the right panel and click OK.

   You will see the Node Commissioning dialog with the current settings for your scanner module.

   The default scanner module settings are as follows.

   - Communication rate = 125 Kbps
   - Node address = 63



Node Commissioning

Current Device Setting

Node Address 2

Network Data Rate 125 kb

New Device Settings

Node Address 0

Network Data Rate 125 kb

Warning!
Network Data Rate should not be changed on an active network.
New Network Data Rate will not take effect until power is cycled.

Network data rate found.

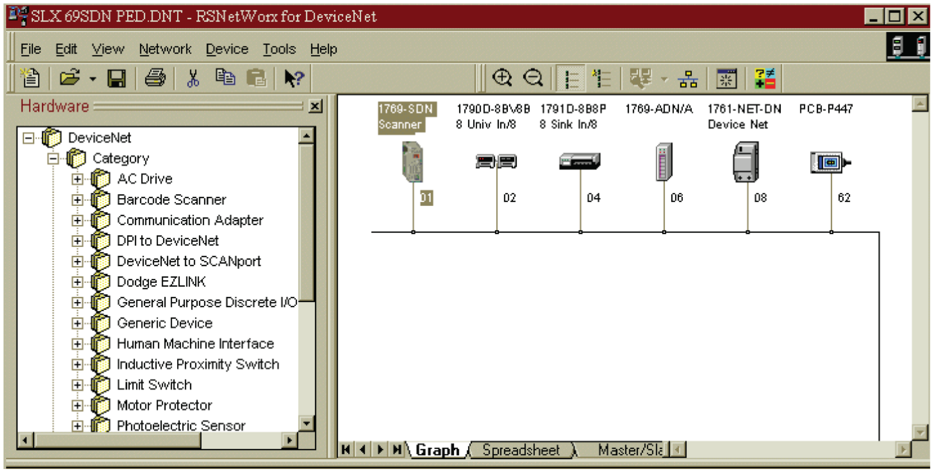
5. In the Node Address box, enter the new node address.

6. Click Apply and Exit the dialog.

Configure the I/O Devices

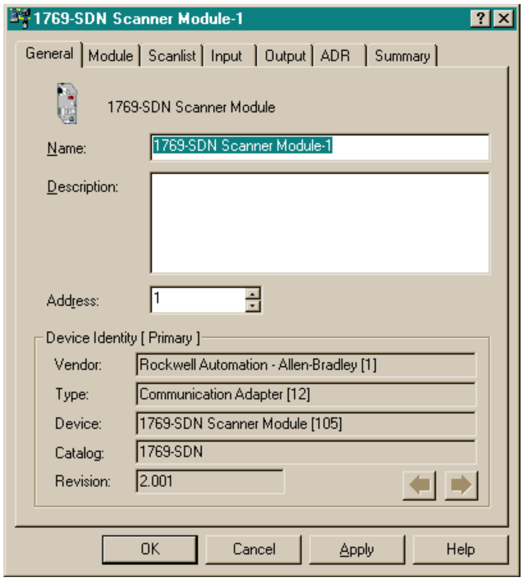
The information in this section will focus on the scanner module's configuration and parameters that you can change by using RSNetWorx for DeviceNet software.

This dialog shows how a DeviceNet network is shown within RSNetWorx for DeviceNet software. As you can see, node one is a 1769-SDN scanner. Double-click on the scanner module to view or modify the scanner's parameters.



General Tab

This dialog shows the properties that RSNetWorx for DeviceNet software displays for the 1769-SDN scanner module.



1769-SDN Scanner Module

Name: 1769-SDN Scanner Module-1

Description:

Address: 1

Device Identity [Primary]

Vendor: Rockwell Automation - Allen-Bradley [1]

Type: Communication Adapter [12]

Device: 1769-SDN Scanner Module [105]

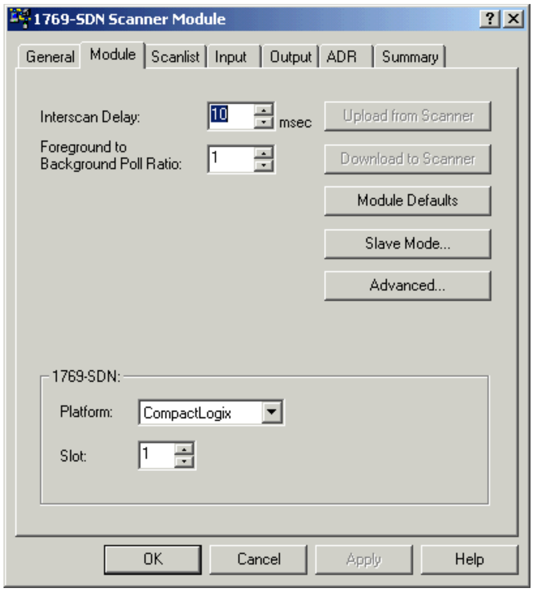
Catalog: 1769-SDN

Revision: 2.001

Available items are accessed through a series of tabs located across the top of the dialog. The General tab is the default tab and provides information relative to the module.

If you ever need product support, you will need to provide the revision number when you contact Rockwell Automation Technical Support.

Module Tab



Interscan Delay: 10 msec

Foreground to Background Poll Ratio: 1

1769-SDN:

Platform: CompactLogix

Slot: 1

Interscan Delay

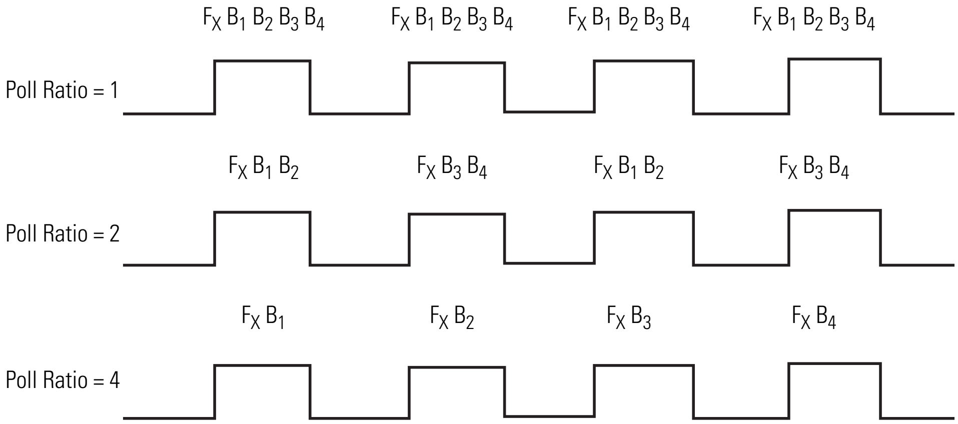
This parameter defines the delay time the scanner module uses between scans of the DeviceNet network. If you have slave devices configured for polled behavior in the 1769-SDN scanner module scanlist, Interscan Delay (ISD) defines the amount of time the scanner module waits between DeviceNet I/O updates.

Increasing the ISD time causes a longer network scan, which adversely affects overall input-to-output performance. However, the increase allows lower priority messages to get more network access. These lower priority messages include those used to do network browsing and configuration upload/download functions. So, if these network functions are sluggish on your system, increase the ISD time.

Foreground to Background Poll Ratio

Devices set for polled behavior can be polled on every I/O scan (foreground) or they can be polled less frequently (background). Setting a device for foreground or background behavior is done when you configure each device in the scanner's input section.

Foreground/Background Polling Behavior



Each waveform shows polling frequency for the given Foreground to Background Poll Ratio.

Where:

F_X = x foreground slave devices.

B_1 = 1st background slave device.

B_2 = 2nd background slave device.

B_3 = 3rd background slave device.

B_4 = 4th background slave device.

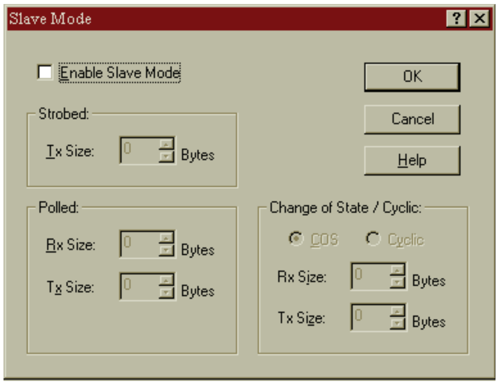
Module Defaults

Clicking Module Defaults sets Interscan Delay and Foreground to Background Poll Ratio to the following values:

- Interscan Delay: 10 ms
- Foreground to Background Poll Ratio: 1

Slave Mode

Enable Slave mode only if you want to use this scanner module as a slave. Clicking Enable Slave Mode opens this dialog.



□ Enable Slave Mode

Strobed:
Tx Size: 0 Bytes

Polled:
Rx Size: 0 Bytes
Tx Size: 0 Bytes

Change of State / Cyclic:
● COS ○ Cyclic
Rx Size: 0 Bytes
Tx Size: 0 Bytes

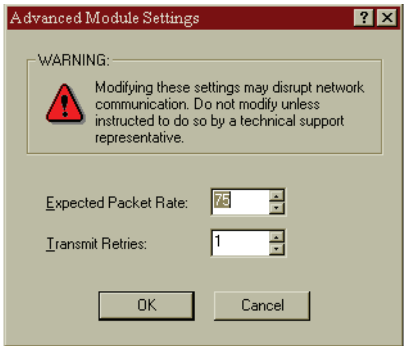
This parameter enables the slave functionality of the scanner, and lets you set the I/O behavior, with the amount of data to be exchanged. The Slave mode settings are described in this table.

Slave Mode Settings

Connection Format	Description
Strobed	Not supported by the 1769-SDN scanner module.
Polled	Values entered into Rx and Tx parameters define how much data will be exchanged over the polled connection that owns the scanner's slave I/O. Maximum size is 128 bytes.
Change of State / Cyclic	Values entered into Rx and Tx parameters define how much data will be exchanged over the change of state or cyclic connection that owns the scanner's slave I/O. Maximum size is 128 bytes.

Advanced Module Settings

Clicking Advanced opens this dialog.



Advanced Module Settings

WARNING:

Modifying these settings may disrupt network communication. Do not modify unless instructed to do so by a technical support representative.

Expected Packet Rate: 75

Transmit Retries: 1

Expected Packet Rate (EPR)

When the scanner module opens a polled or strobed I/O connection, it sets a maximum timeout (expected packet rate) with the device. If the device does not receive a packet from the scanner module within four times the EPR value, the slave device drops the connection. If the scanner module does not receive a packet from the slave within four times the EPR value, it drops the connection and periodically attempts to open a new connection.

When a connection is dropped, status bits in the scanner module identify that the slave is not online. Slave behavior when a connection is dropped is a function of the slave device. If the slave is an I/O device, the outputs will be cleared, held at last state, or set to a fault condition (refer to the slave device's documentation for actual I/O behavior when a connection is dropped).

The EPR default value is 75 ms.

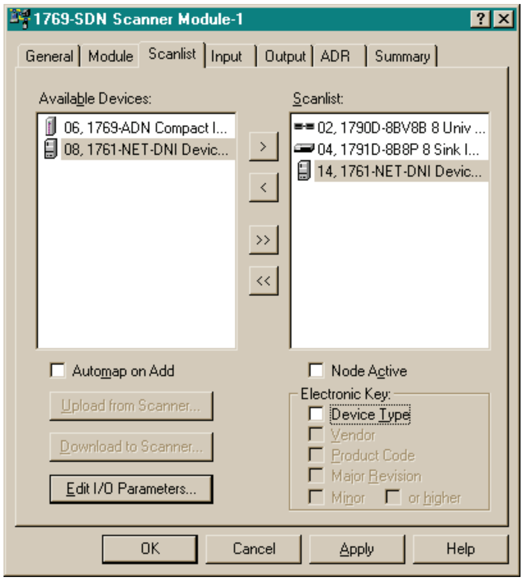
IMPORTANT	Changing the EPR number should be done carefully because it effects how long it takes the scanner module to detect a missing device.

Transmit Retries

Transmit Retries specifies the number of times the scanner module attempts to retransmit a change of state or cyclic message that has not been acknowledged by the slave device. The connection is not necessarily dropped as a result of reaching the retry limit.

Scanlist Tab

The Scanlist defines which devices on the DeviceNet network are owned by the scanner.



Available Devices:

- 06, 1769-ADN Compact I...
- 08, 1761-NET-DNI Devic...

Scanlist:

- 02, 1790D-8BV8B 8 Univ ...
- 04, 1791D-8B8P 8 Sink I...
- 14, 1761-NET-DNI Devic...

☐ Automap on Add

☐ Node Active

Electronic Key:

☐ Device Type
☐ Vendor
☐ Product Code
☐ Major Revision
☐ Minor ☐ or higher

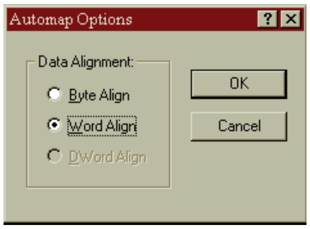
Available Devices

These are the devices on the network that have the ability to be slave I/O devices. Slave-capable devices do not have to be used as slave I/O by a scanner. They may alternately be used as slave I/O by another scanner module on the same network, or they may have dual functionality. An example of a dual function device is the 1761-NET-DNI module. The DNI can be used as slave I/O, or as a communication interface for PLCs or other DF1 full-duplex devices.

Scanlist

These devices have been assigned to be slave I/O to this scanner. A slave device on the DeviceNet network can be owned by only one master at a time. Devices in the scanlist are configured using the input and output tabs.

Automap on Add



Automap Options

Data Alignment:

○ Byte Align
● Word Align
○ DWord Align

Automap allows a slave's I/O to be automatically mapped into the scanner's input or output image tables when the slave device is added to the scanlist. do not check this box if you intend to map a slave device into a particular input or output memory location.

If you do select Automap, you will be prompted to choose how the data should be aligned in the scanner.

Edit I/O Parameters

These parameters will vary depending upon the slave device. Information on configurable parameters is usually provided in the device's documentation.

Electronic Key

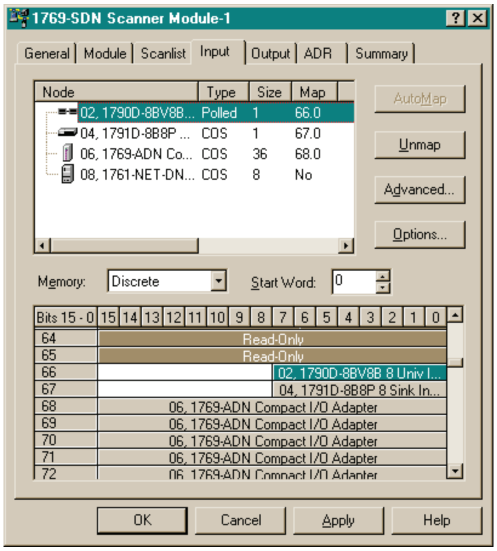
The electronic key is used to make sure that a particular slave device always matches the intended device when the scanner module initiates a connection to it. When one of the devices in the Scanlist section is highlighted, these boxes may be checked to indicate to what extent the key parameters must match the actual device on the network. A match of just Device Type can be selected or the additional parameters of Vendor ID, Product Code, Major Revision, and Minor Revision may be incrementally added.

For Major and Minor Revisions, you can select or higher to indicate that either an exact match, or some higher revision value of the revision is required.

If the scanner module detects a mismatch with any of the key parameters checked, an AutoVerify failure will occur for that slave device and the scanner module will not continue the connection allocation process.

Input Tab

The input tab dialog lets you to define how data from all of the scanner's slave devices are mapped into the input image of the controller.



Node	Type	Size	Map
02, 1790D-8BV8B...	Polled	1	66.0
04, 1791D-8B8P ...	COS	1	67.0
06, 1769-ADN Co...	COS	36	68.0
08, 1761-NET-DN...	COS	8	No

Memory: Discrete

Start Word: 0

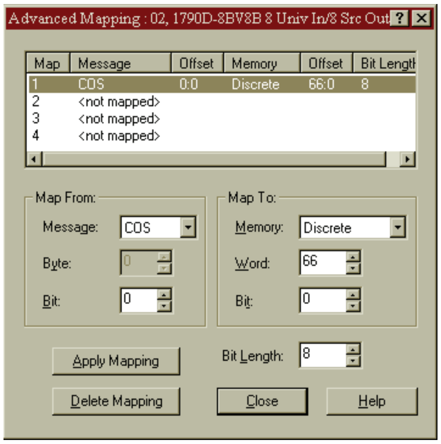
Bits 15 - 0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
64	Read-Only															
65	Read-Only															
66									02, 1790D-8BV8B 8 Univ I...							
67									04, 1791D-8B8P 8 Sink In...							
68	06, 1769-ADN Compact I/O Adapter															
69	06, 1769-ADN Compact I/O Adapter															
70	06, 1769-ADN Compact I/O Adapter															
71	06, 1769-ADN Compact I/O Adapter															
72	06, 1769-ADN Compact I/O Adapter															

In this example, four input devices are mapped into the scanner's input image. The dialog shows each device's:

- node number.
- catalog number.
- connection type used between the scanner module and the slave device (strobed, polled, cyclic, or change of state).
- amount of data that will be exchanged (in bytes).
- word within the controller's input image where the data will appear.

Advanced

The scanner module allows each node’s slave data to be broken up (segmented) into separate input image areas. This capability lets you group data from multiple slave nodes into contiguous memory locations. This is simply an ease-of-use feature. For maximum performance, do not segment a slave device’s data.


Advanced Mapping : 02, 1790D-8BV8B 8 Univ In/8 Src Out

Map	Message	Offset	Memory	Offset	Bit Length
1	COS	0:0	Discrete	66:0	8
2	<not mapped>				
3	<not mapped>				
4	<not mapped>				

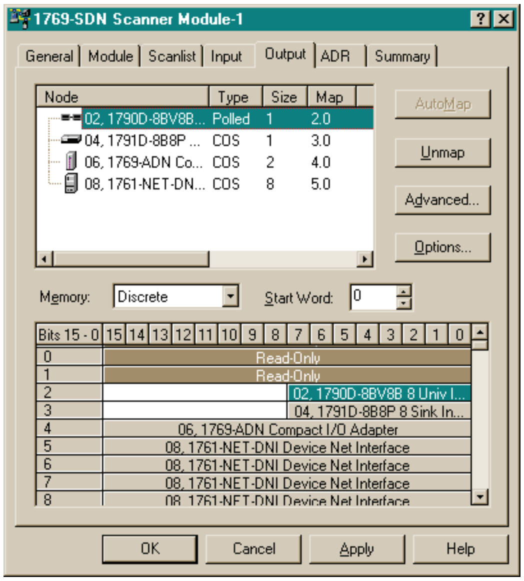
Map From:
Message: COS
Byte: 0
Bit: 0

Map To:
Memory: Discrete
Word: 66
Bit: 0

Bit Length: 8

Output Tab

The output tab dialog lets you define how data from the scanner module is mapped to the outputs of the slave devices.


1769-SDN Scanner Module-1

Node	Type	Size	Map
02, 1790D-8BV8B...	Polled	1	2.0
04, 1791D-8B8P ...	COS	1	3.0
06, 1769-ADN Co...	COS	2	4.0
08, 1761-NET-DN...	COS	8	5.0

Memory: Discrete
Start Word: 0

Bits 15 - 0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Read-Only															
1	Read-Only															
2									02, 1790D-8BV8B 8 Univ I...							
3									04, 1791D-8B8P 8 Sink In...							
4	06, 1769-ADN Compact I/O Adapter															
5	08, 1761-NET-DNI Device Net Interface															
6	08, 1761-NET-DNI Device Net Interface															
7	08, 1761-NET-DNI Device Net Interface															
8	08, 1761-NET-DNI Device Net Interface															

In this example, four output devices are mapped into the scanner's output image. The dialog box shows each device's:

- node number.
- catalog number.
- connection type used between the scanner module and the slave device (strobed, polled, cyclic, or change of state).
- amount of data that will be exchanged (in bytes).
- word within the controller's output image.

Auto Device Replacement (ADR) Tab

The Auto Device Replacement feature automates the replacement of a failed slave device on a DeviceNet network by returning it to the prior level of operation. This includes Configuration Recovery (CR) and Auto-address Recovery (AAR).

CR allows a slave device to be removed from the network and replaced with an identical slave device configured at the same communication rate and node address as the device being replaced.

ADR allows a slave device to be removed from the network and replaced with another identical slave device that is residing on the network at node address 63 and is not in the scan list. The replacement device will have its node address automatically changed to that of the device being replaced. Depending on the level of revision keying, it may be possible for the node address of the replacement device to be changed but not brought online due to a revision-keying mismatch.

The 1769-SDN scanner module can handle any number of device failures simultaneously, however, the AAR feature will be disabled for devices that have the same electronic key. The CR feature will remain active. In this case, you must use the Node Commissioning tool that ships with RSNetWorx for DeviceNet software to change the node's address.

If the electronic key of the replacement device differs only by revision, the CR feature may not work. Before replacing any device, you should make sure that the configuration parameters of the replacement device are compatible with the existing device.

IMPORTANT This behavior may introduce a possible safety hazard for devices such as drives. Use the full electronic key, including revision. You must be using a 1769-SDN scanner module and RSNetWorx for DeviceNet software to enable electronic key revision checking.

Important Considerations

Keep the following in mind when using the ADR feature with RSNetWorx for DeviceNet software:

- ADR is intended for use in single-master environments.
- The CR and AAR features can be used only with devices that are in the scanlist of the scanner.
- The AAR feature can be enabled for a device only if the CR feature is also enabled.
- The CR feature cannot be enabled for devices that have no writable parameters (for example, a device that is entirely configured using hardware switches).

  RSNetWorx for DeviceNet software will notify you of devices that have no writable parameters.

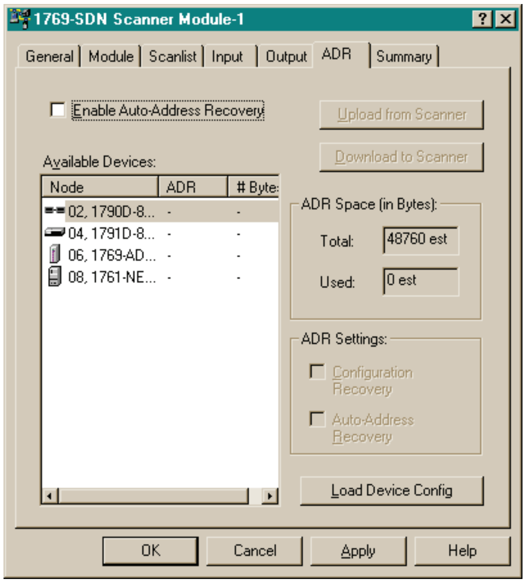
- The AAR feature will not work for devices that do not support changing the node address over the network.

Configure ADR Parameters

On the ADR tab, you can configure the ADR parameters for the 1769-SDN scanner module. Using the controls on this tab, you can select ADR parameters and enable or disable this functionality either globally or on a device-specific basis.

Configure each device on the network. If online, make sure that the configuration is synchronized with the online devices.

1. On the ADR tab, select the device you want to configure to use ADR.



☐ Enable Auto-Address Recovery

Available Devices:

Node	ADR	# Byte:
02, 1790D-8...	-	-
04, 1791D-8...	-	-
06, 1769-AD...	-	-
08, 1761-NE...	-	-

ADR Space (in Bytes):

Total: 48760 est

Used: 0 est

ADR Settings:

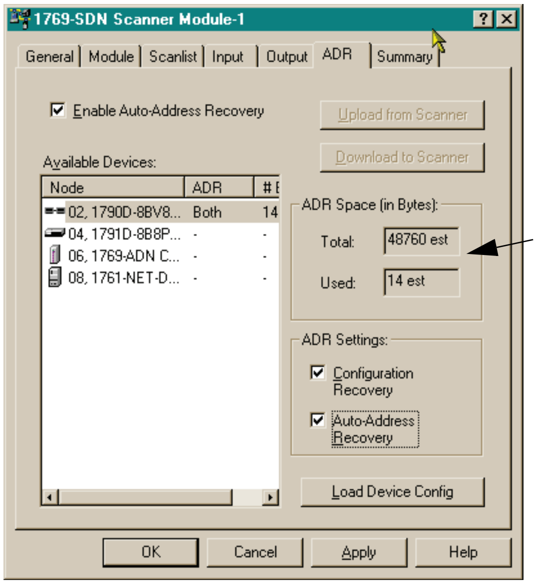
☐ Configuration Recovery

☐ Auto-Address Recovery

2. Click Load Device Config.

   This will load the configuration of each device selected into the scanner module configuration software. If you are online, but did not upload the network, the configuration retrieved for the devices may not be what is actually used online.

3. Click the Configuration Recovery checkbox.



☑ Enable Auto-Address Recovery

Available Devices:

Node	ADR	#
02, 1790D-8BV8...	Both	14
04, 1791D-8B8P...	-	-
06, 1769-ADN C...	-	-
08, 1761-NET-D...	-	-

ADR Space (in Bytes):

Total: 48760 est

Used: 14 est

ADR Settings:

☑ Configuration Recovery

☑ Auto-Address Recovery

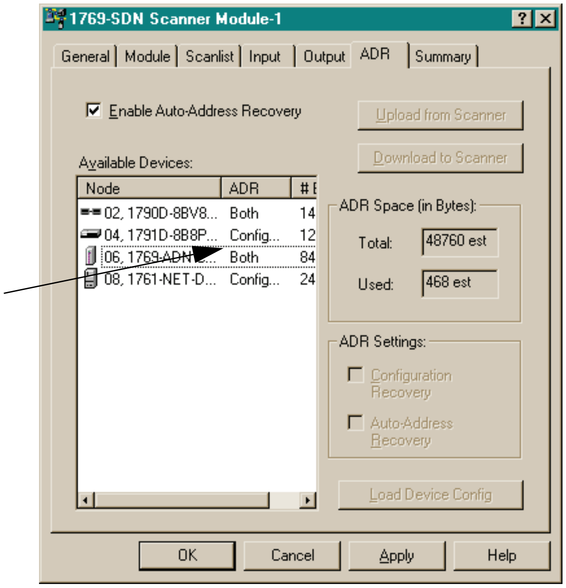
Notice the ADR space (in bytes) is displayed for the module you are updating.

4. Click the Auto-Address Recovery checkbox if desired.

   The Enable Auto-Address Recovery checkbox needs to be checked in order for AAR to work.

5. Repeat steps 1…4 for each desired module.

   Notice that Both indicates Configuration Recovery and Auto-Address Recovery are enabled while Config indicates that only Configuration Recovery is enabled.



☑ Enable Auto-Address Recovery

Available Devices:

Node	ADR	#
02, 1790D-8BV8...	Both	14
04, 1791D-8B8P...	Config...	12
06, 1769-ADN ...	Both	84
08, 1761-NET-D...	Config...	24

ADR Space (in Bytes):

Total: 48760 est

Used: 468 est

ADR Settings:

☐ Configuration Recovery

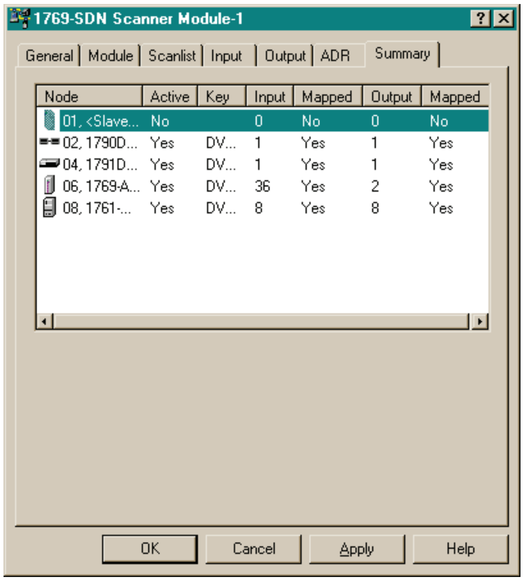
☐ Auto-Address Recovery

6. When finished, click Download to Scanner.

7. Click Apply or OK.

Summary Tab

The summary tab provides a condensed picture of the scanner's configuration: which nodes are mapped, if they are active, and receive and transmit information.

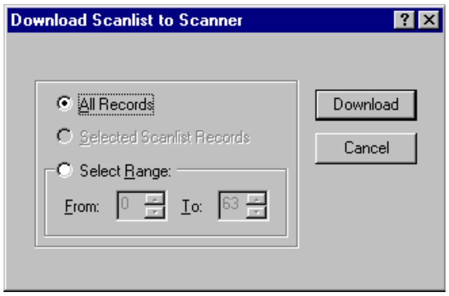


Node	Active	Key	Input	Mapped	Output	Mapped
01, <Slave...	No		0	No	0	No
02, 1790D...	Yes	DV...	1	Yes	1	Yes
04, 1791D...	Yes	DV...	1	Yes	1	Yes
06, 1769-A...	Yes	DV...	36	Yes	2	Yes
08, 1761-...	Yes	DV...	8	Yes	8	Yes

Download and Save Your Configuration

Follow these steps to download and save your configuration.

1. Click the Scanlist tab and then Download to Scanner.



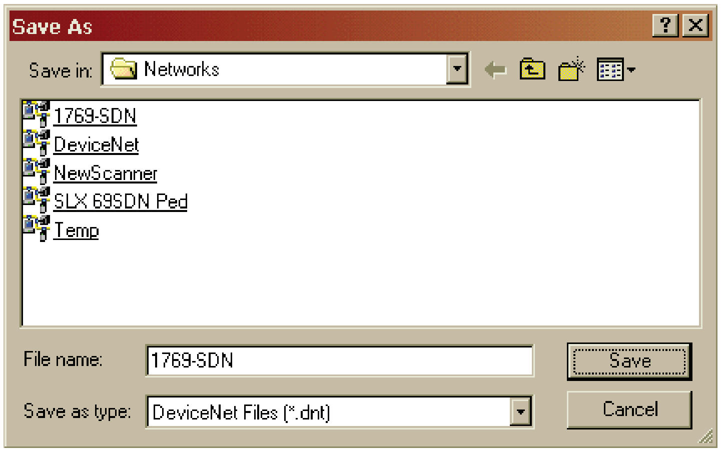
● All Records
○ Selected Scanlist Records
○ Select Range:
From: 0 To: 63

2. Click All Records.

3. Click Download to download the configuration to the scanner.

4. Click OK to complete the DeviceNet scanner module configuration.

5. From the File menu, choose Save As.



File name: 1769-SDN

Save as type: DeviceNet Files (*.dnt)

6. Save the configuration to a DeviceNet file.

7. Close RSNetWorx for DeviceNet software.

Notes:

DeviceNet I/O Image

Introduction

This chapter describes the input and output data structure. This table describes what this chapter contains and where to find specific information.

1769-SDN Input Structure

The 1769-SDN scanner module’s input image is configurable up to 246 words. The input image is broken into two primary components, the status area and the input data area.

Input Image

Name	Size	Word Offset
Status Structure	66-words (fixed)	0…65
Input Data Image	180-word array (configurable)	66…245

MicroLogix 1500 Status Structure

The first area of the MicroLogix 1500 controller input image is the Status Structure. The status words are described in more detail in the following sections.

Description	Words	Data Type
Scan counter	0 and 1	2 words
Device failure array	2…5	64-bit array
Autoverify failure array	6…9	64-bit array
Slave device idle array	10…13	64-bit array
Active node array	14…17	64-bit array
Reserved[1]	18…19	4-byte array
Scanner module status	20 and 21	4-byte array
Reserved array[1]	22…31	20-byte array
Device status array	32…63	64-byte array
Module status register	64 and 65	2 words

[1] Do not manipulate reserved bits. Doing so may interfere with future compatibility.

Scan Counter

This 32-bit unsigned value is incremented each time the DeviceNet network is scanned. The value will automatically roll over to zero and continue counting.

Device Failure Array

These 64 bits provide status information for use in your control program. For each slave device owned by the scanner, you should monitor the respective bit within these four words. If a slave device faults, the bit that corresponds to that node address will be set (1). If a slave device is not faulted or returns from a faulted state to an operating state, the corresponding bit for that node will be cleared (0).

Slave Device Status Information

Input Word	Bit 0...15	Description
2[1]	Node 0…15	Bit ON (1) = Slave node faulted Bit OFF (0) = Slave node not faulted
3[2]	Node 16…31	
4[3]	Node 32…47	
5[4]	Node 48…63	

[1] Bit 0 corresponds to Node 0, Bit 1 corresponds to Node 1 … Bit 15 corresponds to Node 15.

[2] Bit 0 corresponds to Node 16, Bit 1 corresponds to Node 17 … Bit 15 corresponds to Node 31.

[3] Bit 0 corresponds to Node 32, Bit 1 corresponds to Node 33 … Bit 15 corresponds to Node 47.

[4] Bit 0 corresponds to Node 48, Bit 1 corresponds to Node 49 … Bit 15 corresponds to Node 63.

Autoverify Failure Array

These four words allow the control program to monitor if a slave's Device Key and Size matches the Device Key and Size in the scanner. A bit value of 0 (OFF) represents a configuration match, a bit value of 1 (ON) represents a mismatch.

ATTENTION ⚠

A value of 0 does not indicate that the slave has been brought online or is functional, only that there is a configuration match between the slave and the scanner.

Slave Device Configuration Comparison to Scanner Module Configuration

Input Word	Bit 0…15	Description
6[1]	Node 0…15	Bit ON (1) = Slave node mismatch Bit OFF (0) = Slave node match
7[2]	Node 16…31	
8[3]	Node 32…47	
9[4]	Node 48…63	

[1] Bit 0 corresponds to Node 0, Bit 1 corresponds to Node 1 … Bit 15 corresponds to Node 15.

[2] Bit 0 corresponds to Node 16, Bit 1 corresponds to Node 17 … Bit 15 corresponds to Node 31.

[3] Bit 0 corresponds to Node 32, Bit 1 corresponds to Node 33 … Bit 15 corresponds to Node 47.

[4] Bit 0 corresponds to Node 48, Bit 1 corresponds to Node 49 … Bit 15 corresponds to Node 63.

Slave Device Idle Array

The slave device idle array contains 64 bits of data. Each bit indicates the state of a slave device. A bit value of 0 represents a Non-idle mode or that the slave is not present in the scanlist. A bit value of 1 represents Idle mode where the slave node is present in the scanlist.

Slave Device Status

Input Word	Bit 0…15	Description
10[1]	Node 0...15	Bit ON (1) = Idle Bit OFF (0) = Non-idle
11[2]	Node 16…31	
12[3]	Node 32…47	
13[4]	Node 48…63	

[1] Bit 0 corresponds to Node 0, Bit 1 corresponds to Node 1 … Bit 15 corresponds to Node 15.

[2] Bit 0 corresponds to Node 16, Bit 1 corresponds to Node 17 … Bit 15 corresponds to Node 31.

[3] Bit 0 corresponds to Node 32, Bit 1 corresponds to Node 33 … Bit 15 corresponds to Node 47.

[4] Bit 0 corresponds to Node 48, Bit 1 corresponds to Node 49 … Bit 15 corresponds to Node 63.

Active Node Array

The Active Node array contains 64 bits of data that represents each slave node's status. A slave node is considered active when it is present and enabled in the master's scanlist. A bit value of 0 means Not Active, a bit value of 1 means Active.

Slave Node Status

Input Word	Bit 0...15	Description
14[1]	Node 0…15	Bit ON (1) = Active node Bit OFF (0) = Inactive node
15[2]	Node 16…31	
16[3]	Node 32…47	
17[4]	Node 48…63	

[1] Bit 0 corresponds to Node 0, Bit 1 corresponds to Node 1 … Bit 15 corresponds to Node 15.

[2] Bit 0 corresponds to Node 16, Bit 1 corresponds to Node 17 … Bit 15 corresponds to Node 31.

[3] Bit 0 corresponds to Node 32, Bit 1 corresponds to Node 33 … Bit 15 corresponds to Node 47.

[4] Bit 0 corresponds to Node 48, Bit 1 corresponds to Node 49 … Bit 15 corresponds to Node 63.

Scanner Module Status

The Scanner Device Status is a 4-byte array.

Scanner Device Status

Input Word	Bit	Description
20	0…7 (lower byte)	Scanner Address in BCD
	8…15 (upper byte)	Scanner Status in BCD
21	0…7 (lower byte)	Slave Device Address in BCD
	8…15 (upper byte)	Slave Device Status in BCD

Reserved Array

Reserved Array

Input Word	Description[1]
22...31	Always 0

[1] Do not manipulate reserved bits. Doing so may interfere with future compatibility.

Device Status Array

The Device Status Array is a 64-byte array containing the information shown in this table. Each byte indicates the status code of the scanner's master and the slave devices.

Device Status

Input Word	Bit	Description
32	0…7 (lower byte)	Node 0 DeviceNet status
	8…15 (upper byte)	Node 1 DeviceNet status
33	0…7 (lower byte)	Node 2 DeviceNet status
	8…15 (upper byte)	Node 3 DeviceNet status
...	...	...
63	0…7 (lower byte)	Node 62 DeviceNet status
	8…15 (upper byte)	Node 63 DeviceNet status

Module Status Register

The scanner module supports a 32-bit Module Status Register.

Input Word	Bit	Description	Operation
64	0	Run	Bit ON (1) = scanning I/O Bit OFF (0) = halted
	1	Fault	Bit ON (1) = faulted Bit OFF (0) = not faulted
	2	Network disable	Bit ON (1) = disabled Bit OFF (0) = not disabled
	3	Device failure	Bit ON (1) = failure Bit OFF (0) = no failure
	4	Autoverify failure	
	5	Communication failure	
	6	Duplicate node failure	
	7	DeviceNet power detect	Bit ON (1) = power Bit OFF (0) = no power
	8...15	Reserved[1]	N/A
65	0...15	Reserved[1]	N/A

[1] Do not manipulate reserved bits. Doing so may interfere with future compatibility.

CompactLogix Status Structure

The second area of the CompactLogix controller input image is the Status Structure. The status words are described in more detail in the following sections.

Description	Data Type
I.Status.Scan Counter	2 words
I.Status.Device Failure Register	64-bit array
I.Status.Autoverify Failure Register	64-bit array
I.Status.Device Idle Register	64-bit array
I.Status.Active Node Register	64-bit array
I.Status.Status Display	8 bytes
I.Status.Scanner Address	1 byte
I.Status.Scanner Status	1 byte
I.Status.Scrolling Device Address	1 byte
I.Status.Scrolling Device Status	1 byte
I.Status.Device Status	64 bytes

Scan Counter

This 32-bit unsigned value is incremented each time the DeviceNet network is scanned. The value will automatically roll over to zero and continue counting.

Device Failure Register

These 64 bits provide status information for use in your control program. For each slave device owned by the scanner, you should monitor the respective bit within these four words. If a slave device faults, the bit that corresponds to that node address will be set (1). If a slave device is not faulted or returns from a faulted state to an operating state, the corresponding bit for that node will be cleared (0).

Slave Device Status Information

Input Word	Bit 0...15	Description
2[1]	Node 0…15	Bit ON (1) = Slave node faulted Bit OFF (0) = Slave node not faulted
3[2]	Node 16…31	
4[3]	Node 32…47	
5[4]	Node 48…63	

[1] Bit 0 corresponds to Node 0, Bit 1 corresponds to Node 1 … Bit 15 corresponds to Node 15.

[2] Bit 0 corresponds to Node 16, Bit 1 corresponds to Node 17 … Bit 15 corresponds to Node 31.

[3] Bit 0 corresponds to Node 32, Bit 1 corresponds to Node 33 … Bit 15 corresponds to Node 47.

[4] Bit 0 corresponds to Node 48, Bit 1 corresponds to Node 49 … Bit 15 corresponds to Node 63.

Autoverify Failure Register

These four words allow the control program to monitor if a slave's Device Key and Size matches the Device Key and Size in the scanner. A bit value of 0 (OFF) represents a configuration match, a bit value of 1 (ON) represents a mismatch.

ATTENTION ⚠

A value of 0 does not indicate that the slave has been brought online or is functional, only that there is a configuration match between the slave and the scanner.

Slave Device Configuration Comparison to Scanner Module Configuration

Input Word	Bit 0...15	Description
6[1]	Node 0…15	Bit ON (1) = Slave node mismatch Bit OFF (0) = Slave node match
7[2]	Node 16…31	
8[3]	Node 32…47	
9[4]	Node 48…63	

[1] Bit 0 corresponds to Node 0, Bit 1 corresponds to Node 1 … Bit 15 corresponds to Node 15.

[2] Bit 0 corresponds to Node 16, Bit 1 corresponds to Node 17 … Bit 15 corresponds to Node 31.

[3] Bit 0 corresponds to Node 32, Bit 1 corresponds to Node 33 … Bit 15 corresponds to Node 47.

[4] Bit 0 corresponds to Node 48, Bit 1 corresponds to Node 49 … Bit 15 corresponds to Node 63.

Device Idle Register

The slave device idle array contains 64 bits of data. Each bit indicates the state of a slave device. A bit value of 0 represents a Non-idle mode or that the slave is not present in the scanlist. A bit value of 1 represents Idle mode where the slave node is present in the scanlist.

Slave Device Status

Input Word	Bit 0...15	Description
10[1]	Node 0...15	Bit ON (1) = Idle Bit OFF (0) = Non-idle
11[2]	Node 16…31	
12[3]	Node 32…47	
13[4]	Node 48…63	

[1] Bit 0 corresponds to Node 0, Bit 1 corresponds to Node 1 … Bit 15 corresponds to Node 15.

[2] Bit 0 corresponds to Node 16, Bit 1 corresponds to Node 17 … Bit 15 corresponds to Node 31.

[3] Bit 0 corresponds to Node 32, Bit 1 corresponds to Node 33 … Bit 15 corresponds to Node 47.

[4] Bit 0 corresponds to Node 48, Bit 1 corresponds to Node 49 … Bit 15 corresponds to Node 63.

Active Node Register

The Active Node array contains 64 bits of data that represents each slave node’s status. A slave node is considered active when it is present and enabled in the master’s scanlist. A bit value of 0 means Not Active, a bit value of 1 means Active.

Slave Node Status

Input Word	Bit 0...15	Description
14[1]	Node 0…15	Bit ON (1) = Active node Bit OFF (0) = Inactive node
15[2]	Node 16…31	
16[3]	Node 32…47	
17[4]	Node 48…63	

[1] Bit 0 corresponds to Node 0, Bit 1 corresponds to Node 1 … Bit 15 corresponds to Node 15.

[2] Bit 0 corresponds to Node 16, Bit 1 corresponds to Node 17 … Bit 15 corresponds to Node 31.

[3] Bit 0 corresponds to Node 32, Bit 1 corresponds to Node 33 … Bit 15 corresponds to Node 47.

[4] Bit 0 corresponds to Node 48, Bit 1 corresponds to Node 49 … Bit 15 corresponds to Node 63.

Status Display

The Status Display is a 4-byte array of the 1769-SDN display.

Scanner Address

The Scanner Address is the scanner’s DeviceNet node number.

Scanner Status

The Scanner Status is the scanner’s DeviceNet status.

Scrolling Device Address

The Scrolling Device Address is the DeviceNet address and status of node with errors. The data updates once per second, scrolling through all nodes with errors.

Scrolling Device Status

The Scrolling Device Status is the status of the node with errors. The data updates once per second, scrolling through all nodes with errors.

Device Status

The Device Status is the status of each DeviceNet node, indexed by node number into a 64-byte array.

CompactLogix Status Register

The third area of the CompactLogix controller input image is the Status Register. The status words are described in more detail in the following sections.

Description	Data Type
I.StatusRegister.Run	bit
I.StatusRegister.Fault	
I.StatusRegister.DisableNetwork	
I.StatusRegister.DeviceFailure	
I.StatusRegister.Autoverify	
I.StatusRegister.CommFailure	
I.StatusRegister.DupNodeFail	
I.StatusRegister.DnetPowerDetect	

Run

This bit displays whether the controller is in Idle or Run module.

- 0 = Idle
- 1 = Run

Fault

This bit displays whether the network is faulted.

- 0 = Network is not faulted
- 1 = Network is faulted

Disable Network

This bit displays whether the network is disabled.

- 0 = Network is not disabled
- 1 = Network is disabled

Device Failure

This bit displays whether any device failures have occurred on the network.

- 0 = No device failures exist
- 1 = Device failure exists (examine the status structure for causes)

Autoverify Failure

This bit displays whether the scanner has failed to initialize any devices on the network.

- 0 = The scanner has initialized all devices
- 1 = The scanner has failed to initialize at least one device

Comm Failure

This bit displays whether a communication failure exists on the network.

- 0 = No communication failures exist
- 1 = A communication failure exists

Dup Node Failure

This bit displays whether a network failure exists because of a duplicate node address.

- 0 = No failure exists
- 1 = Failure exists due to duplicate node address

Dnet Power Detect

This bit displays whether a DeviceNet power failure exists on the network.

- 0 = No Devicenet power failure exists on the network
- 1 = DeviceNet power failure exists

CompactLogix Command Register

The first area of the CompactLogix controller output image is the Command Register. The status words are described in more detail in the following sections.

Description	Data Type
O.CommandRegister.Run	bit
O.CommandRegister.Fault	
O.CommandRegister.DisableNetwork	
O.CommandRegister.HaltScanner	
O.CommandRegister.Reset	

Run

This bit changes the DeviceNet network between Idle and Run module.

- 0 = Idle
- 1 = Run

Fault

This bit forces the scanner into a faulted condition.

- 0 = Network is not faulted
- 1 = Network is faulted

Disable Network

This bit disables the DeviceNet network.

- 0 = Network is not disabled
- 1 = Network is disabled

Halt Scanner

This bit halts the scanner.

- 0 = Scanner is not halted
- 1 = Scanner is halted

Reset

This bit resets the scanner.

- 0 = The scanner has not been reset
- 1 = The scanner has been reset.

Do not leave this bit = 1. If you do not change the bit back to 0, the scanner continuously resets.

Input Data Image

The Input Data Image is described on page 15 for the MicroLogix 1500 controller and on page 16 for the CompactLogix controller.

1769-SDN Output Structure

The Output Structure is described on page 15 for the MicroLogix 1500 controller and on page 16 for the CompactLogix controller.

Chapter 7

Use the 1769-SDN Scanner Module with MicroLogix Controllers

Introduction

This chapter contains an example in which the 1769-SDN scanner module is used with a MicroLogix 1500 controller. This table describes what this chapter contains and where to find specific information.

Topic	Page
MicroLogix 1500 Controllers	103
RSLogix 500 Programming Software I/O Configuration	104
Backplane Messaging	111
Program Upload and Download	112
Configure a Local DeviceNet Message	113

MicroLogix 1500 Controllers

The MicroLogix 1500 programmable controller has two different processors that are compatible with the 1769-SDN scanner module. The 1764-LSP and 1764-LRP processors can use the scanner module as a DeviceNet master and own DeviceNet slave devices. This allows either processor to communicate with intelligent devices like drives, scales, and starters, or use the DeviceNet network to expand the processor's I/O capacity.

The 1764-LRP processor allows messaging functionality over the DeviceNet network, so non-I/O data may be exchanged. Multiple 1769-SDN scanner modules may be used in a 1764-LRP system. However, only the first two scanners can be used for messaging.

The amount of power that modules draw from the processor or expansion power supply, and the amount of data that the processor can support will determine how many can be used. Configuration tools are available from http://www.ab.com/micrologix to determine if an application can be supported

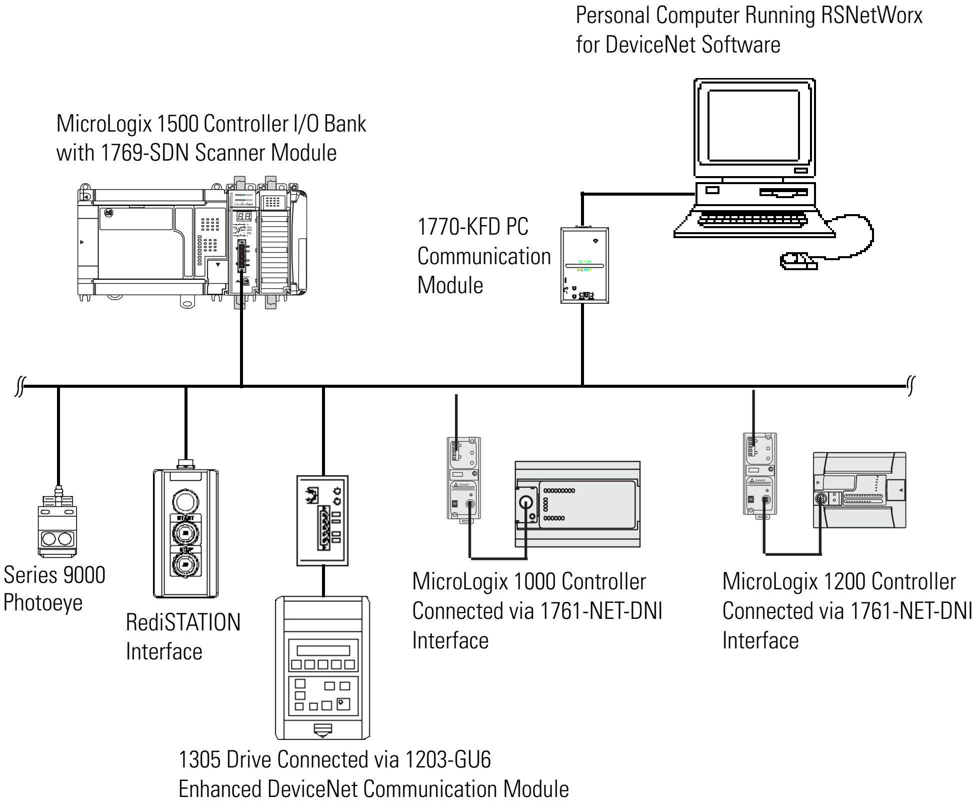
In addition to electrical limitations, data space limitations also exist. The maximum size of the input and output images for each module in the system is 250 input and 250 output words of data.

RSLogix 500 Programming Software I/O Configuration

One of the advanced features of RSLogix 500 software is the ability to have the programming software establish a communication connection with the controller and read which I/O modules are attached to the controller. This capability significantly reduces the effort involved in configuring a system.

This section illustrates how to determine which I/O modules are attached to the controller, and manually configure the modules. We will then configure the scanner. An example network is shown below.

DeviceNet Network

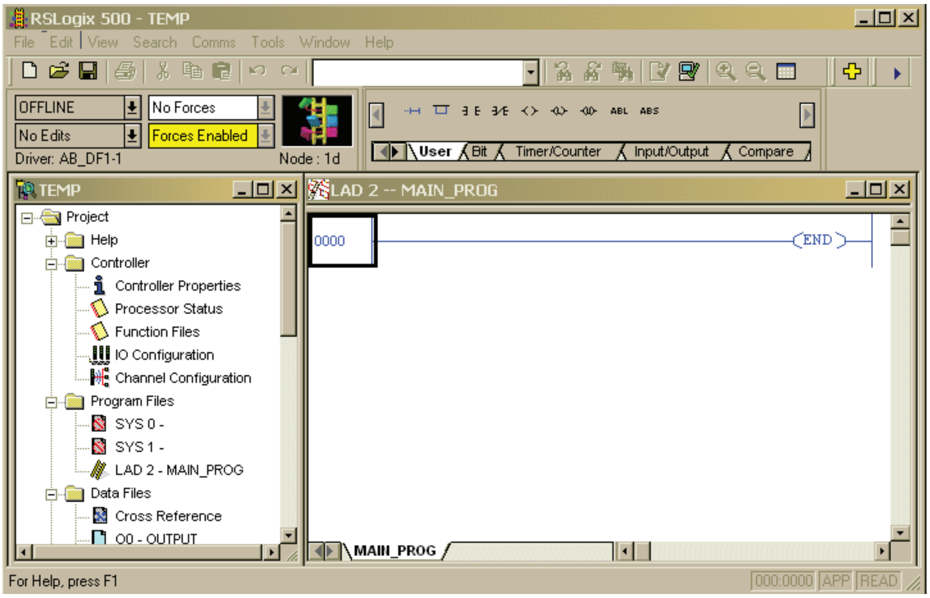

Personal Computer Running RSNetWorx for DeviceNet Software
MicroLogix 1500 Controller I/O Bank with 1769-SDN Scanner Module
1770-KFD PC Communication Module
Series 9000 Photoeye
RediSTATION Interface
MicroLogix 1000 Controller Connected via 1761-NET-DNI Interface
MicroLogix 1200 Controller Connected via 1761-NET-DNI Interface
1305 Drive Connected via 1203-GU6 Enhanced DeviceNet Communication Module

Start the Project

Follow these steps to begin your project.

1. Open RSLogix 500 software.
2. Choose File ⇒New ⇒MicroLogix 1500 LRP series C.

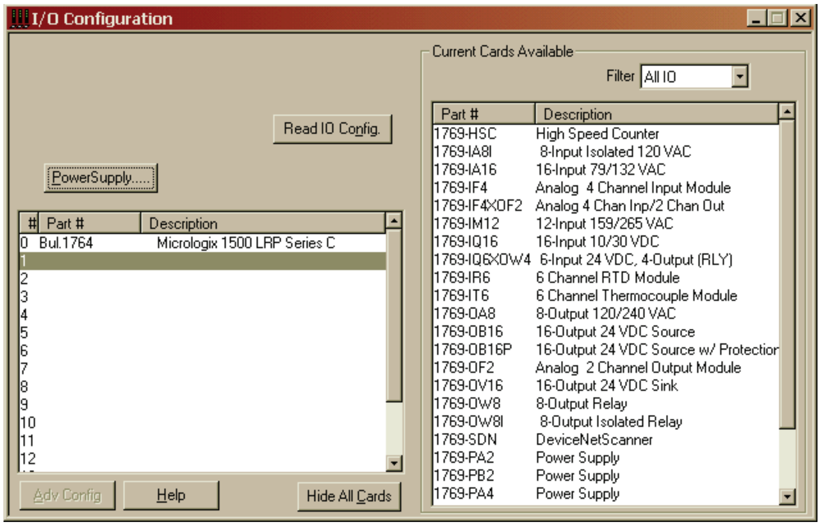
   This dialog should match what you see on your computer.



In this example the name for this application is TEMP. In the TEMP dialog, you see everything associated with the application. Within the Controller folder you will see I/O Configuration.

3. Double-click I/O Configuration.

I/O Configuration Screen



Filter: All IO

#	Part #	Description
0	Bul.1764	Micrologix 1500 LRP Series C
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		

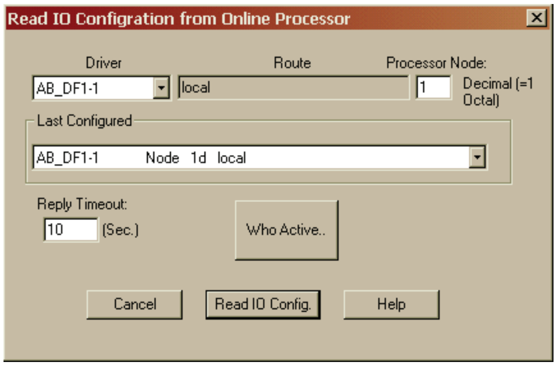
Part #	Description
1769-HSC	High Speed Counter
1769-IA8I	8-Input Isolated 120 VAC
1769-IA16	16-Input 79/132 VAC
1769-IF4	Analog 4 Channel Input Module
1769-IF4XOF2	Analog 4 Chan Inp/2 Chan Out
1769-IM12	12-Input 159/265 VAC
1769-IQ16	16-Input 10/30 VDC
1769-IQ6XOW4	6-Input 24 VDC, 4-Output (RLY)
1769-IR6	6 Channel RTD Module
1769-IT6	6 Channel Thermocouple Module
1769-OA8	8-Output 120/240 VAC
1769-OB16	16-Output 24 VDC Source
1769-OB16P	16-Output 24 VDC Source w/ Protectior
1769-OF2	Analog 2 Channel Output Module
1769-OV16	16-Output 24 VDC Sink
1769-OW8	8-Output Relay
1769-OW8I	8-Output Isolated Relay
1769-SDN	DeviceNetScanner
1769-PA2	Power Supply
1769-PB2	Power Supply
1769-PA4	Power Supply

The I/O configuration dialog shows each of the controller I/O slots. For the MicroLogix 1500 controller, slot 0 contains the embedded I/O that is part of the MicroLogix 1500 base unit. Slots 1...16 are for Compact I/O expansion modules (referred to as local I/O, because they are physically attached to the controller). Slots 9...16 are available using only a series C processor with a series B base unit.

Click Read I/O Config to have RSLogix 500 software read the controller's local I/O and configure the slots automatically.

Drag the appropriate modules from the available list (right pane) to the appropriate slot on the left to manually configure the controller I/O. Modules must be contiguous (with no open slots) from 1...16.

Read I/O Configuration



Read IO Configration from Online Processor

Driver: AB_DF1-1
Route: local
Processor Node: 1 Decimal (=1 Octal)

Last Configured: AB_DF1-1 Node 1d local

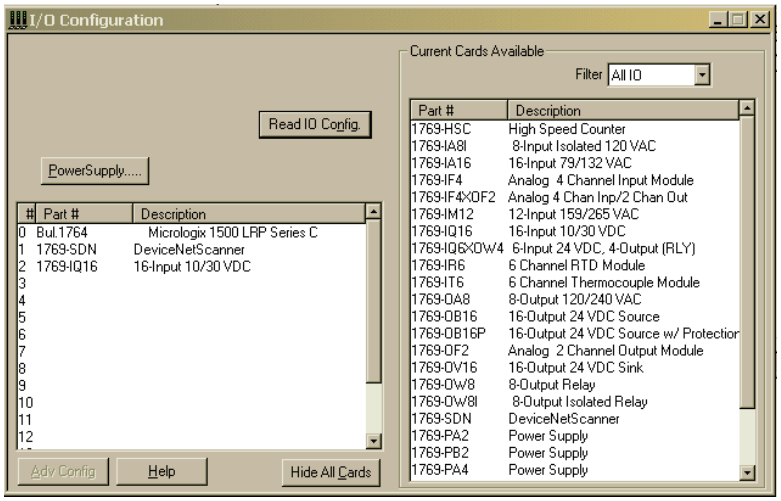
Reply Timeout: 10 (Sec.)

Next you see a communication dialog that lets you select a communication path by using RSLinx software to connect to the MicroLogix controller.

If you have previously connected to a controller, the communication driver that you used before will be the active driver. This dialog provides the ability to change the driver or perform a Who Active across a network to locate the specific MicroLogix controller.

If the driver and path are correct, click Read I/O Config.

Installed I/O



I/O Configuration

#	Part #	Description
0	Bul.1764	Micrologix 1500 LRP Series C
1	1769-SDN	DeviceNetScanner
2	1769-IQ16	16-Input 10/30 VDC
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		

Current Cards Available

Filter: All IO

Part #	Description
1769-HSC	High Speed Counter
1769-IA8I	8-Input Isolated 120 VAC
1769-IA16	16-Input 79/132 VAC
1769-IF4	Analog 4 Channel Input Module
1769-IF4XOF2	Analog 4 Chan Inp/2 Chan Out
1769-IM12	12-Input 159/265 VAC
1769-IQ16	16-Input 10/30 VDC
1769-IQ6XOW4	6-Input 24 VDC, 4-Output (RLY)
1769-IR6	6 Channel RTD Module
1769-IT6	6 Channel Thermocouple Module
1769-OA8	8-Output 120/240 VAC
1769-OB16	16-Output 24 VDC Source
1769-OB16P	16-Output 24 VDC Source w/ Protection
1769-OF2	Analog 2 Channel Output Module
1769-OV16	16-Output 24 VDC Sink
1769-OW8	8-Output Relay
1769-OW8I	8-Output Isolated Relay
1769-SDN	DeviceNetScanner
1769-PA2	Power Supply
1769-PB2	Power Supply
1769-PA4	Power Supply

RSLogix 500 software displays all of the I/O modules that are attached to the MicroLogix controller.

In this example, there is a 1769-SDN scanner module at slot 1, and a 16-point digital input module in slot 2.

1769-SDN Scanner Module Configuration

Double-click the module to configure a specific module. Double-click the module in slot 1 to configure the scanner module in this example.

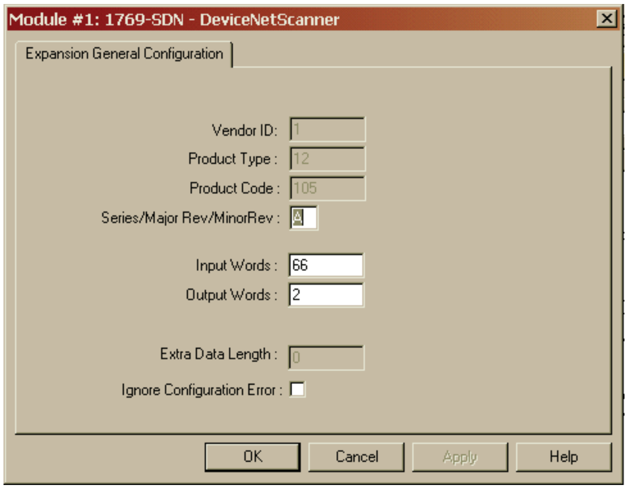
Input Words

This is the number of input words the controller assigns to the module. The first 66 words (0...65) are required by the scanner module for status. DeviceNet slave input data words start at slot word 66. You can have a maximum of 180 input words for DeviceNet slave devices (maximum slot amount for 1769-SDN scanner module inputs = 246).

Output Words

This is the number of output words the controller assigns to the module. The first two words (0 and 1) are required by the scanner module for status. DeviceNet slave output data words start at slot word 2. You can have a maximum of 180 output words for DeviceNet slave devices (maximum slot amount for scanner module outputs = 182).

Changing the 1769-SDN Configuration



Module #1: 1769-SDN - DeviceNetScanner

Expansion General Configuration

Vendor ID: 1

Product Type : 12

Product Code : 105

Series/Major Rev/MinorRev : A

Input Words : 66

Output Words : 2

Extra Data Length : 0

Ignore Configuration Error : □

You can change (add or remove) the amount of data the controller has assigned to the scanner module in the expansion module configuration dialog. From within RSLogix 500 software, I/O configuration, open the 1769-SDN scanner module and change the input or output words as needed. Save the program and download to the controller.

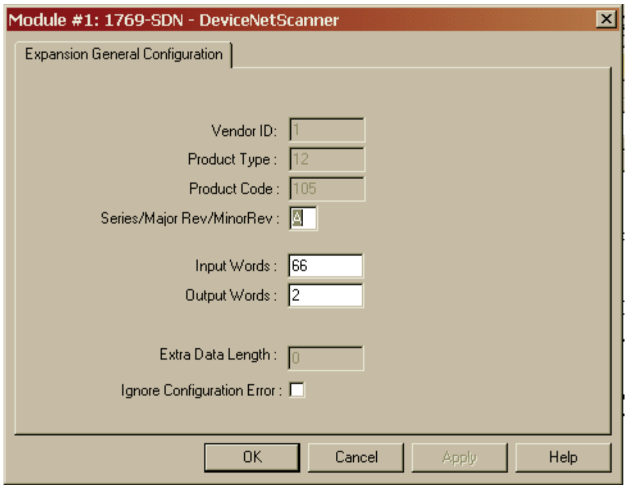
Reducing the number of words for either inputs or outputs will require a change in the 1769-SDN scanner module's scanlist, which is done using RSNetWorx for DeviceNet software.

IMPORTANT

We recommend that you not reduce the number of words assigned to a 1769-SDN scanner module's scanlist once a system is operational. Changing the number of words may cause addressing problems on the controller side, and mapping changes on the DeviceNet network.

Adding words to an existing system is relatively easy to do, because it doesn't affect existing addresses or mapping. Add the number of additional words that are needed in the module (using the above example), and change the scanlist using RSNetWorx for DeviceNet software.

The configuration dialog for the scanner module is shown below.


Module #1: 1769-SDN - DeviceNetScanner

Expansion General Configuration

Vendor ID: 1
Product Type : 12
Product Code : 105
Series/Major Rev/MinorRev : A

Input Words : 66
Output Words : 2

Extra Data Length : 0
Ignore Configuration Error : □

Ignore Configuration Error

Checking (enabling) the Ignore Configuration error checkbox instructs the module to ignore I/O size mismatches. If this is checked and the input/output scanlist configured by RSNetWorx for DeviceNet software (DeviceNet side of the 1769-SDN scanner module) does not match the amount of I/O data assigned by the controller (0...180 words), the module will not generate an error.

If this checkbox is not checked, the number of data words on the controller side must match the number of words configured by RSNetWorx for DeviceNet software. The default condition is unchecked (report an error on mismatch).

Series/Major Rev/Minor Rev

You will need this information if you contact Rockwell Automation Technical Support.

Backplane Messaging

The MicroLogix 1500 1764-LRP processor and the 1769-SDN scanner module also support backplane messaging. This new level of functionality allows the processor to read (get) or write (set) data to other devices on the DeviceNet network. This is also referred to as Explicit Messaging.

You can use two different types of messages to exchange information with the DeviceNet device. The type of message used is determined by the destination device. You can generate a PCCC message or a CIP message.

PCCC Messaging

Programmable controller communications commands (PCCC) provides point to point and master/slave communication between devices. PCCC is an open protocol that is built into all Allen-Bradley controllers, and many other Allen-Bradley and third-party products.

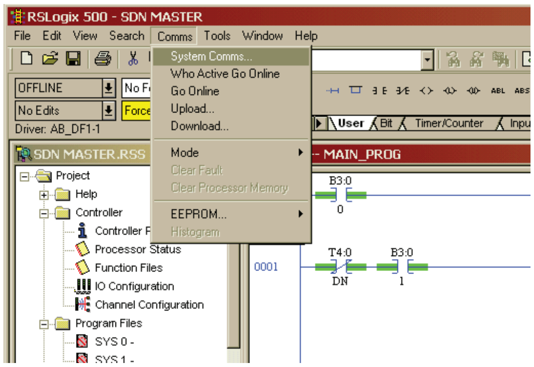
PCCC messaging has been used for many years on DH-485, DH+, and Ethernet networks, and for point-to-point communication between Allen-Bradley controllers. PCCC messaging lets you upload or download programs over the DeviceNet network, and allows users to message across DeviceNet networks, just like they did using DH-485 or DH+ networks. There are a number of devices that support PCCC messaging over DeviceNet, including the 1761-NET-DNI (DNI) interface, 1203-GU6 interface, and RSLinx software. If the DeviceNet network has DNI interfaces, either device can initiate a PCCC message.

Program Upload and Download

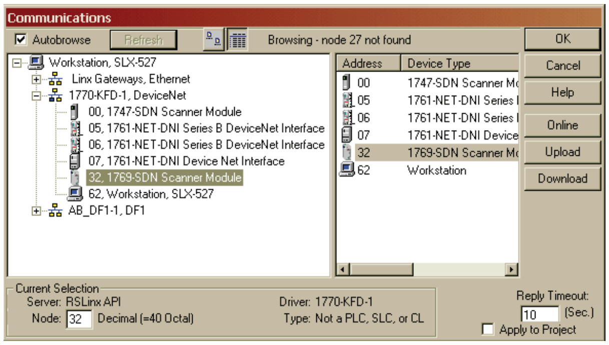
Before performing a program upload or download through the scanner, be sure that the module is properly installed in the system, and that a terminator is present at the end of the Compact I/O expansion bus.

IMPORTANT DeviceNet networks can operate at 125 Kbps, 250 Kbps, or 500 Kbps. Depending on network size and communication activity, performing program upload or download operations while the network is controlling an application may impact control system performance. It is up to you to know and understand how upload and download will impact operations.

Choose Comms ⇒System Comms to upload or download programs with RSLogix 500 software.



System Comms will generate an RSLinx software dialog similar to the example below.


Communications
☑ Autobrowse
Refresh
Browsing - node 27 not found
Workstation, SLX-527
Linx Gateways, Ethernet
1770-KFD-1, DeviceNet
00, 1747-SDN Scanner Module
05, 1761-NET-DNI Series B DeviceNet Interface
06, 1761-NET-DNI Series B DeviceNet Interface
07, 1761-NET-DNI Device Net Interface
32, 1769-SDN Scanner Module
62, Workstation, SLX-527
AB_DF1-1, DF1
OK
Cancel
Help
Online
Upload
Download

Address	Device Type
00	1747-SDN Scanner Mo
05	1761-NET-DNI Series I
06	1761-NET-DNI Series I
07	1761-NET-DNI Device
32	1769-SDN Scanner Mo
62	Workstation

Current Selection
Server: RSLinx API
Node: 32 Decimal (=40 Octal)
Driver: 1770-KFD-1
Type: Not a PLC, SLC, or CL

Reply Timeout: 10 (Sec.)
□ Apply to Project

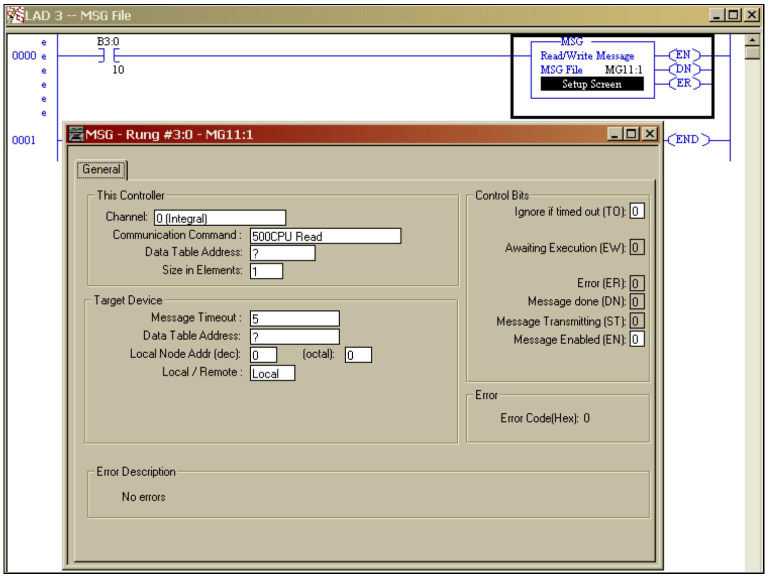
In this example, the DeviceNet interface is a 1770-KFD module. Selecting the 1770-KFD driver will show the devices on the DeviceNet network.

In this example, upload or download can be performed with the devices at nodes 5, 6, 7, and 32. Node 32 is a 1769-SDN module. Simply highlight the 1769-SDN module and click either Upload or Download on the right side of the screen.

Configure a Local DeviceNet Message

This section describes how to configure a local message by using the scanner module and a MicroLogix 1500 1764-LRP processor.

Message Setup Dialog



This Controller
Channel: 0 (Integral)
Communication Command : 500CPU Read
Data Table Address: ?
Size in Elements: 1

Target Device
Message Timeout : 5
Data Table Address: ?
Local Node Addr (dec): 0 (octal): 0
Local / Remote : Local

Control Bits
Ignore if timed out (TO): 0
Awaiting Execution (EW): 0
Error (ER): 0
Message done (DN): 0
Message Transmitting (ST): 0
Message Enabled (EN): 0

Error
Error Code(Hex): 0

Error Description
No errors

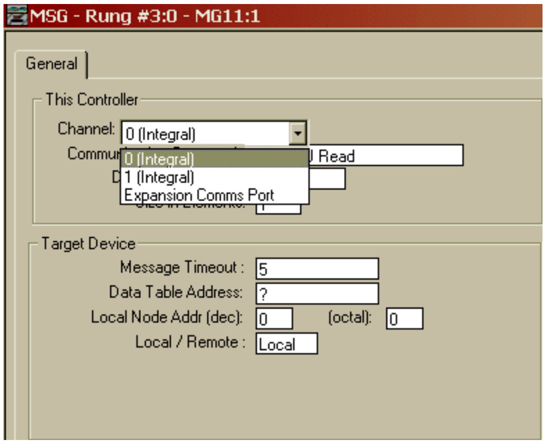
Rung 0 shows a standard RSLogix 500 message (MSG) instruction preceded by conditional logic.

Access the Message Setup dialog by double-clicking the Setup Screen.

The RSLogix 500 Message Setup dialog appears. This dialog is used to set up or monitor message parameters for this controller, target device, and control bits. Descriptions of each of these sections follow.

Channel Parameter

The 1764-LRP controller supports three different pathways for messaging. Channels 0 and 1 are RS-232 ports and are functionally identical to MicroLogix 1200 and MicroLogix 1500 1764-LSP controllers. The 1764-LRP controller also supports backplane communication through the Expansion Communication Port (ECP) as illustrated below.



Channel: 0 (Integral)

Commun... ...U Read

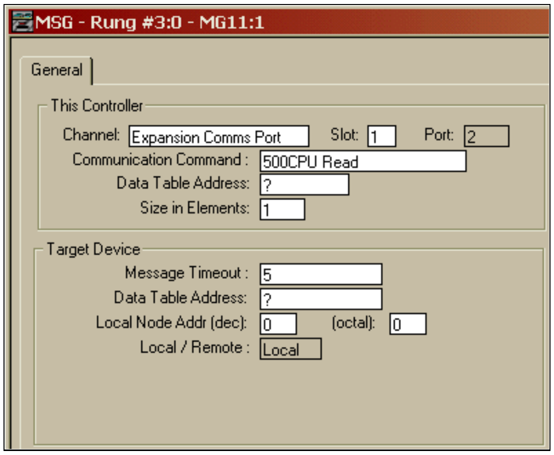
Message Timeout : 5

Data Table Address: ?

Local Node Addr (dec): 0 (octal): 0

Local / Remote : Local

When ECP is chosen, you are able to select which slot position (1...16) the scanner module resides in. The 1764-LRP processor can support up to two 1769-SDN scanner modules with full messaging functionality.



Channel: Expansion Comms Port Slot: 1 Port: 2

Communication Command : 500CPU Read

Data Table Address: ?

Size in Elements: 1

Message Timeout : 5

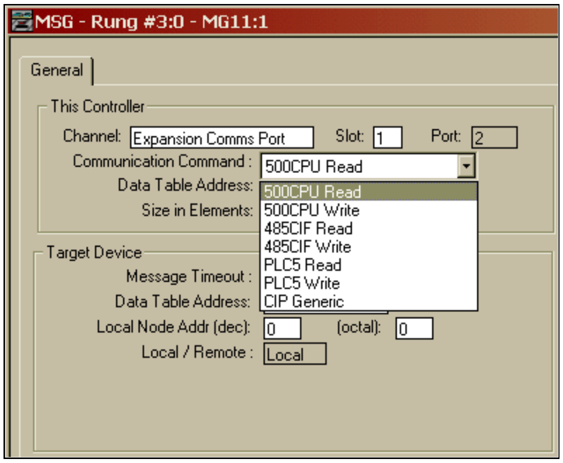
Data Table Address: ?

Local Node Addr (dec): 0 (octal): 0

Local / Remote : Local

You can use multiple 1769-SDN scanner modules in a MicroLogix 1500 system, but you can message only through the first two. Any other 1769-SDN scanner module can be used only for I/O scanning.

Communication Command



MSG - Rung #3:0 - MG11:1

General

This Controller

Channel: Expansion Comms Port Slot: 1 Port: 2

Communication Command : 500CPU Read

Data Table Address:

Size in Elements:

Target Device

Message Timeout :

Data Table Address:

Local Node Addr (dec): 0 (octal): 0

Local / Remote : Local

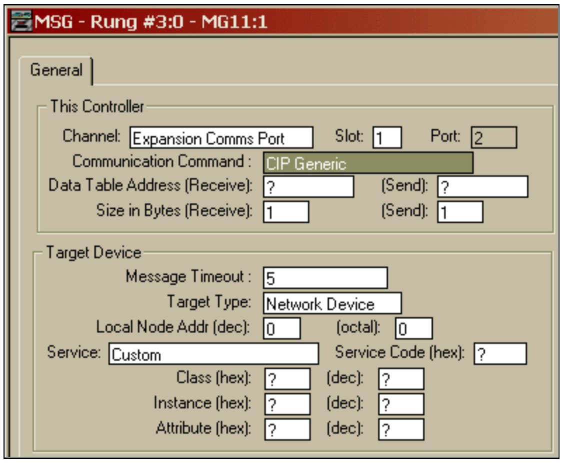
The 1764-LRP controller supports the six standard types of communication commands (same as all other MicroLogix 1200 and 1500 controllers) and CIP Generic. When any of these six standard commands are chosen, you can initiate standard messages to destination devices connected to DeviceNet products that support PCCC messaging (including MicroLogix and SLC controllers using 1761-NET-DNI interfaces, and other MicroLogix 1500 controllers using 1769-SDN scanner modules). You can initiate reads, writes, program upload/download and online monitoring across DeviceNet. This is functionally identical to DH-485 and DH+ networking.

CIP Generic

Control & Information Protocol (CIP) is a newer and more versatile protocol than PCCC. It is an open protocol that is supported by newer Allen-Bradley controllers and third-party products.

CIP messaging is the native messaging format for the DeviceNet network. All DeviceNet devices are compliant with CIP messaging. The MicroLogix 1500 1764-LRP processor (series C) has an enhanced message instruction that provides simple, easy to use CIP messaging.

Selecting CIP Generic configures the message instruction to communicate with DeviceNet devices that do not support PCCC messaging. When CIP Generic is chosen, you will notice that a number of message parameters change and many new ones become available depending upon the service selected.



MSG - Rung #3:0 - MG11:1

This Controller

Channel: Expansion Comms Port Slot: 1 Port: 2

Communication Command : CIP Generic

Data Table Address (Receive): ? (Send): ?

Size in Bytes (Receive): 1 (Send): 1

Target Device

Message Timeout : 5

Target Type: Network Device

Local Node Addr (dec): 0 (octal): 0

Service: Custom Service Code (hex): ?

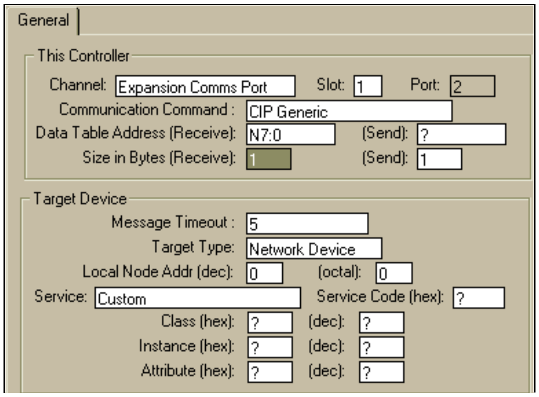
Class (hex): ? (dec): ?

Instance (hex): ? (dec): ?

Attribute (hex): ? (dec): ?

Data Table Address (receive and send)

This value identifies the data file location within the 1764-LRP controller that will receive data from the DeviceNet device, and/or the starting data file location that will be sent to the destination DeviceNet device.



This Controller
Channel: Expansion Comms Port Slot: 1 Port: 2
Communication Command : CIP Generic
Data Table Address (Receive): N7:0 (Send): ?
Size in Bytes (Receive): 1 (Send): 1

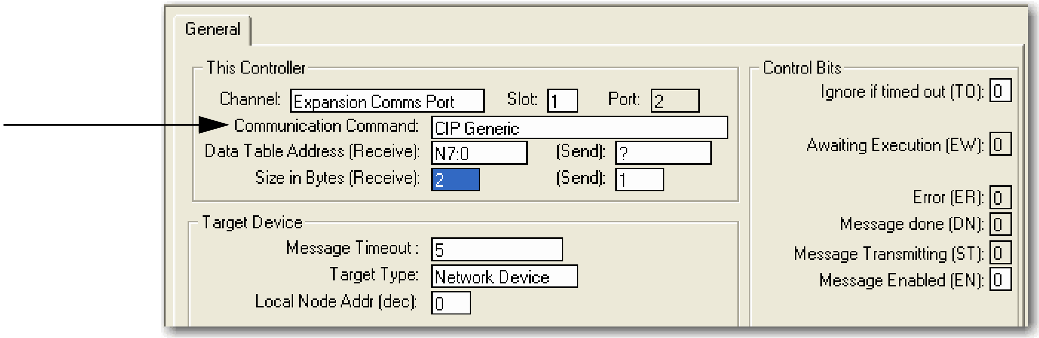
Target Device
Message Timeout : 5
Target Type: Network Device
Local Node Addr (dec): 0 (octal): 0
Service: Custom Service Code (hex): ?
Class (hex): ? (dec): ?
Instance (hex): ? (dec): ?
Attribute (hex): ? (dec): ?

Size in Bytes (receive and send)

Since all data transmitted on the DeviceNet network is byte based, you must enter the number of bytes that will be received and sent. You must make sure that enough memory is available in the destination device. Word elements within 1764-LRP controllers contain 2 bytes each. These include Bit and Integer data files. Long word and Floating point elements contain 4 bytes each.

For receive, the size in bytes entered must be greater than or equal to the number of bytes that the DeviceNet device will return. DeviceNet devices return a fixed number of bytes, depending on the class and service. If more data is returned than expected, the message will be in error and no data will be written. If less data is returned than expected, the data will be written and the remainder of the bytes will be filled with zeros.

In the example dialog shown below, N7:0 will receive 2 bytes (1 word) of data.

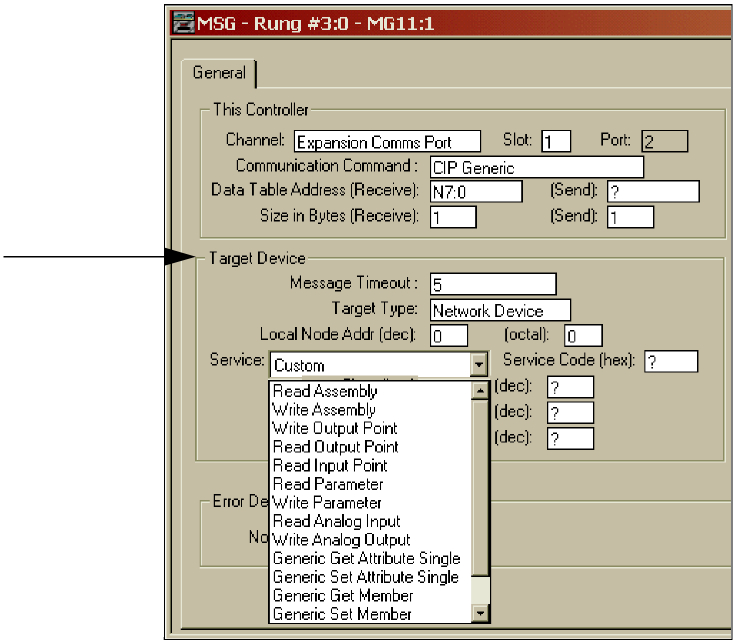


This Controller
Channel: Expansion Comms Port Slot: 1 Port: 2
Communication Command: CIP Generic
Data Table Address (Receive): N7:0 (Send): ?
Size in Bytes (Receive): 2 (Send): 1

Target Device
Message Timeout : 5
Target Type: Network Device
Local Node Addr (dec): 0

Control Bits
Ignore if timed out (TO): 0
Awaiting Execution (EW): 0
Error (ER): 0
Message done (DN): 0
Message Transmitting (ST): 0
Message Enabled (EN): 0

Target Device



MSG - Rung #3:0 - MG11:1

General

This Controller

Channel: Expansion Comms Port Slot: 1 Port: 2

Communication Command : CIP Generic

Data Table Address (Receive): N7:0 (Send): ?

Size in Bytes (Receive): 1 (Send): 1

Target Device

Message Timeout : 5

Target Type: Network Device

Local Node Addr (dec): 0 (octal): 0

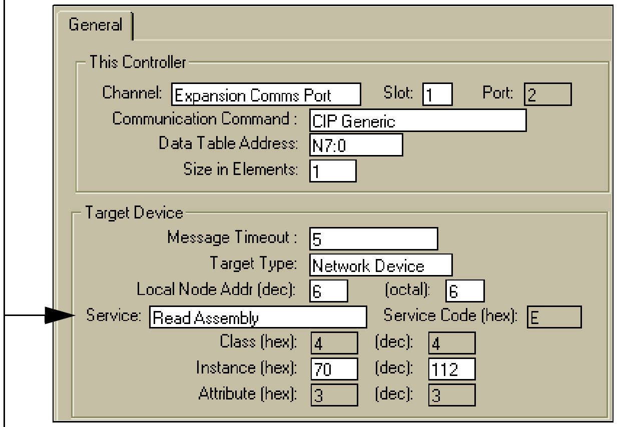
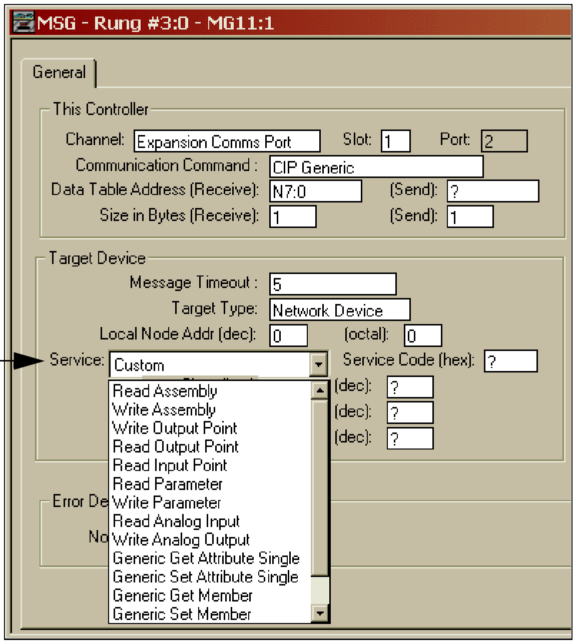
Service: Custom Service Code (hex): ?

(dec): ?

(dec): ?

(dec): ?

Field	Definition
Message Timeout	Message timeout is specified in seconds. If the target does not respond within this time period, the message instruction will generate a specific error. The amount of time that is acceptable should be based on application requirements and network capacity/loading.
Target Type	You can select either Module or Network Device. If you need to message to a device on the DeviceNet network, select Network Device. If you need to message to a DeviceNet parameter on the scanner, select Module. This allows the control program access to module parameters. Many of the module parameters are not editable, and some can be edited only when the module is in Idle mode.
Local Node Address	This is the target device's DeviceNet node number.

Field	Definition
Service	The DeviceNet network uses services to provide specific messaging functions. A number of standard services with their corresponding parameters have been preconfigured for ease of use.  General This Controller Channel: Expansion Comms Port Slot: 1 Port: 2 Communication Command : CIP Generic Data Table Address: N7:0 Size in Elements: 1 Target Device Message Timeout : 5 Target Type: Network Device Local Node Addr (dec): 6 (octal): 6 Service: Read Assembly Service Code (hex): E Class (hex): 4 (dec): 4 Instance (hex): 70 (dec): 112 Attribute (hex): 3 (dec): 3 If you need to use a service that is not available, select one of the generic services. The generic service lets you enter specific service code parameters. Information on what services a target device supports is usually provided in the device's documentation.  MSG - Rung #3:0 - MG11:1 General This Controller Channel: Expansion Comms Port Slot: 1 Port: 2 Communication Command : CIP Generic Data Table Address (Receive): N7:0 (Send): ? Size in Bytes (Receive): 1 (Send): 1 Target Device Message Timeout : 5 Target Type: Network Device Local Node Addr (dec): 0 (octal): 0 Service: Custom Service Code (hex): ? (dec): ? (dec): ? (dec): ? Read Assembly Write Assembly Write Output Point Read Output Point Read Input Point Read Parameter Write Parameter Read Analog Input Write Analog Output Generic Get Attribute Single Generic Set Attribute Single Generic Get Member Generic Set Member Error De No

MSG Instruction Error Codes

When the processor detects an error during the transfer of Expansion I/O Communication Module message data, the processor sets the ER bit and writes an error code at MG file sub-element #18 that you can monitor from your programming software.

1769-SDN Module Error Code

Error Code	Description
E0H	Expansion I/O Communication Module Error

The processor also writes general status information related to this error at the high byte of MG file sub-element #22 that can be monitored from your program.

Error Status Information

General Status	Description
01H	Illegal or unsupported service parameter
02H	Resource unavailable
04H	Segment type error in IOI
07H	Connection lost
08H	Service not supported
09H	Invalid attribute value
0BH	Already in requested mode/state
0CH	Object state conflict
0EH	Attribute not setable
10H	Device state conflict
11H	Reply data too large
13H	Not enough data
14H	Attribute not supported
15H	Too much data
16H	Object does not exist
19H	Store operation failure
20H	Invalid parameter
28H	Invalid member ID

Troubleshooting

Introduction

The 1769-SDN scanner module's front panel status indicators aid in troubleshooting the module. The diagnostics provided by these indicators are described in this chapter.

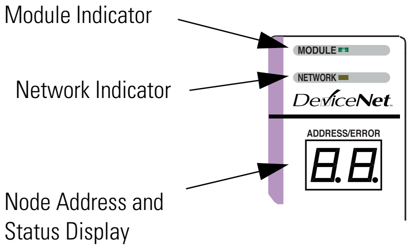
Status Indicators

The first step in troubleshooting is to observe the 1769-SDN scanner module's status indicators and seven-segment numeric displays. The indicators function as follows:

- The bi-color (green/red) Module status indicator shows whether the scanner module has power and is functioning properly.
- The bi-color (green/red) Network status indicator provides information about the DeviceNet channel communication link.
- The numeric display shows node address and status display information.

  Status information precedes the node address.

This table summarizes the meanings of the status indicators and numeric codes.



Troubleshooting the Status Indicators and Numeric Display

Indicator	Color/Status	Description
Module	Off	No power applied to module. Apply power.
	Flashing Green	No bus master (MicroLogix or CompactLogix controller) present. Verify module connectors are properly seated. If they are, cycle power to the controller. If this does not correct the problem, replace the controller. If replacing the controller does not correct the problem, replace the scanner.
	Solid Green	Normal operation.
	Flashing Red	• Recoverable Fault - Memory has been erased or is being programmed. Complete the flash update or start a new update. • Bad firmware in module. Replace firmware.
	Solid Red	Unrecoverable fault. Verify module connectors are properly seated. If they are, verify that bus terminator/end cap is installed. Cycle power. If still faulted, replace the module.

Troubleshooting the Status Indicators and Numeric Display

Indicator	Color/Status	Description
Network	Off	No module power, no network power, or communications are not occurring between the module and the DeviceNet network. (This may be an acceptable condition.) Verify module has power. Check that the DeviceNet cable is securely connected and the DeviceNet network is powered. Verify that network power is adequate (11…5V DC).
	Flashing Green	Device is operational. There are no connections established with any of the network devices. If the module is supposed to be controlling DeviceNet slaves, configure the module's scanlist.
	Solid Green	Normal operation. Scanlist is configured. Module is not in Idle mode.
	Flashing Red	One or more of the devices that the scanner module is communicating with is in a timed out state. Monitor the status display, or the module's status field to determine which slave device is offline.
	Solid Red	Critical network failure. Duplicate DeviceNet node address detected. Reset module. Change module's node address or change conflicting device's node address. If failure continues, replace module.
Seven-segment Numeric Display	Node Address and Status Display	Indicates diagnostic information about the status of the module. • When the numeric display is showing 0…63, it is indicating the scanner's DeviceNet node address. • When it shows 67…69, it indicates backup status. • When it shows 70…99, it indicates an error code for the displayed node address. • When it flashes alternating numbers, one is the error code (70…99), and the other is the node number (0…63) that has generated the error.

Error Codes

The following table describes the error codes indicated by the seven-segment numeric display.

Device Status

Code (decimal)	Name	Description
67	Backup Mode	Scanner in Backup Scanner mode.
68	No Backup Scanner	No backup scanner module present. Install and configure a backup scanner, if needed.
69	Backup Scanner Invalid CRC	The configuration of the primary and the secondary controller does not match. Either one or both of the configurations is incorrect. Make sure that the same configuration is downloaded to both scanners.
70	Duplicate Node	Controller has Failed Duplicate Node Address Check. The node address selected is already in use. Change the module's or conflicting device's network address (node number) to an available one.
71	Illegal Scanlist Data	Illegal data in scanlist. Reconfigure the scanlist table and remove any illegal data.
72	Slave Timeout	One of the module's slave devices has stopped communicating. Inspect the module's slave devices and verify the DeviceNet connections.
73	Electronic Key Mismatch	The slave device Vendor ID key parameter does not match the slave's configuration in the module's scanlist. Make sure that the device at the flashing node address matches the desired electronic key (vendor, product code, product type)
75	No Messages Received	No network traffic received by the scanner. 10 seconds have elapsed and no network traffic for the module or for any other device have been received by the module. Verify the scanlist is correctly configured to scan slave devices. Verify DeviceNet network connections.
76	No Message For Scanner	No direct network traffic for the scanner module detected. 10 seconds elapsed and no DeviceNet input being screened by the module has been received.
77	Slave Data Size Mismatch	The data being received from the slave device does not match the configuration in the scanlist. Either reconfigure the slave device, or change the module's scanlist to match the slave device.
78	No Such Device	Slave device in scanlist does not exist. Either add the device to the DeviceNet network, or delete the device's entry in the scanlist.
79	Transmit Failure	The module has failed to transmit a message. Make sure that the module is connected to a valid network. Check for disconnected cables.

Device Status

Code (decimal)	Name	Description
80	In Idle Mode	Module is in Idle mode. Put the controller into RUN mode and enable the RUN bit in the Module Command Array. See page 16.
81	Scanner Faulted	The scanner module has stopped producing and consuming I/O data. This condition does not affect the scanner's system or messaging modes. Check the FAULT value in the Module Command Array. See page 16.
82	Fragmentation Error	Error detected in sequence of fragmented I/O messages from device. Check scanlist table entry for slave device to make sure that input and output data lengths are correct. Check slave device configuration.
83	Slave Init Error	Slave device is returning error responses when the module attempts to communicate with it. Check slave device's configuration. Reboot slave device.
84	Not Yet Initialized	Module has not completed its initial attempt to establish communication with its slaves.
85	Receive Buffer Overflow	Data size returned is larger than expected. Configure the slave device for a smaller data size.
86	Device Went Idle	Device is producing idle state. Check the device configuration and slave node status.
89	Auto Device Replacement (ADR) Error	Slave device responded with an error to the initialization data sent to it by the scanner; or the configuration table in the scanner's flash memory is not valid for a slave node. Try the ADR download again. If it still fails, try clearing the ADR flash by downloading an empty ADR configuration to the scanner module and then try the ADR configuration again.
90	Disabled Network	DeviceNet Port is disabled. Check for the DISABLE being set in the Module Command Array. See page 16.
91	Bus Off	Bus off condition detected on integral DeviceNet port. Check the DeviceNet connections and physical media integrity. Check system for failed slave devices or other possible sources of network interference. Check the Baud Rate.
92	No DeviceNet Power	No network power detected on DeviceNet port. Provide network power. Make sure the module drop cable is providing the proper power to the DeviceNet port.

Device Status

Code (decimal)	Name	Description
95	FLASH Update	Flash Update In Progress **IMPORTANT**: Do not disconnect the module from the network while a FLASH update is in progress.
98	Firmware Corrupted	Firmware is corrupted. Reflash module firmware. Do not power cycle the module. Doing so may cause the module to become inoperable. If the problem persists contact Rockwell Automation Technical Support.
99	Hard Fault	Cycle power. Reflash module firmware. Contact Rockwell Automation Technical Support.

1769-SDN DeviceNet Class Codes

Introduction

This appendix contains the most commonly used class codes for the 1769-SDN DeviceNet scanner module. They are shown in the following tables.

DeviceNet Object

Name	Class	Instance	Attribute	Data Size	Access
MAC ID	0x03	0x01	0x01	1 byte	Get/Set[1]
Baud Rate	0x03	0x01	0x02	1 byte	Get/Set
Bus Off Counter	0x03	0x01	0x04	1 byte	Get

[1] The set also causes a reset.

Identity Object

Name	Class	Instance	Attribute	Data Size	Access
Vendor ID	0x01	0x01	0x01	2 bytes	Get
Device Type	0x01	0x01	0x02	2 bytes	Get
Product Code	0x01	0x01	0x03	2 bytes	Get
Revision	0x01	0x01	0x04	2 bytes	Get
Status	0x01	0x01	0x05	2 bytes	Get
Serial Number	0x01	0x01	0x06	4 bytes	Get
Product Name	0x01	0x01	0x07	9 bytes	Get

Notes:

CompactLogix Backup on the DeviceNet Network

Introduction

This appendix offers a solution to back up your CompactLogix controller on the DeviceNet network. CompactLogix system backup on the DeviceNet network is a simple, low-cost, back-up system most effective when used in smaller applications that require fast switchovers from a primary to a secondary controller.

This back-up solution:

- minimizes downtime in case of controller failure when the same program is used in both controllers.

- mitigates the risk of changes adversely affecting the application (use old, proven program in one controller and new, untested program in the other controller). If the new untested program causes a problem, a forced switchover can be made to the older proven program without downloading the program again.

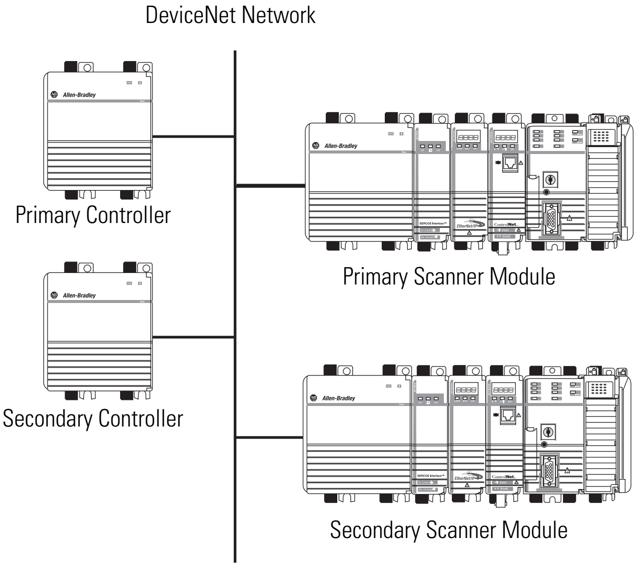
The CompactLogix backup on the DeviceNet network solution takes advantage of Shared DeviceNet Mastership of Slave I/O Devices technology. Typically, only a single DeviceNet master exists for any particular slave. With Shared DeviceNet Mastership, two masters can exist. Heartbeat communication between primary and secondary controllers determines which scanner module is the master and which scanner module remains in Standby mode.

How the Backup Works

The following figure shows an example backup system. In the backup system, the following occurs:

- Both controllers/scanners simultaneously receive all inputs.
- Both controllers execute in parallel but are not synchronized.
- Only the primary controllers send output data to the I/O devices. A virtual switch in the 1769-SDN scanner module is used to switch outputs between primary and secondary controllers.
- After failure or forced switchover, outputs are automatically switched by the 1769-SDN scanner module from the primary controller to secondary. When the switch occurs, the secondary controller becomes the primary controller.

  The switchover occurs so quickly that the I/O devices do not timeout; these devices are unaware that redundant controllers/scanners exist and are unaware of the switchover.



Backup System Requirements

The CompactLogix backup on DeviceNet solution requires that you use the following:

- RSLogix 5000 programming software, version 10 or later
- Two CompactLogix controllers, firmware revision 10.*x* or later
- Two 1769-SDN scanner modules, firmware revision 3.*x* or later

Additional requirements are as follows:

- When setting up the DeviceNet network, you must set the primary and secondary 1769-SDN scanner modules to the same node address and reserve the next node address.

  We recommend that you set the primary and secondary 1769-SDN scanner module node addresses to 0 and reserve node 1. However, you can use any successive node numbers (for example, 30 and 31).

- All I/O and operator interfaces that required backup must be on the DeviceNet network.

- The scanlists in the two DeviceNet scanner modules must be identical.

Configure the Backup System

Follow these steps to configure a CompactLogix backup system on the DeviceNet network. Some of these steps are described in greater detail in the rest of the appendix.

1. Install all I/O and operator interfaces that you need to back up on the DeviceNet network.

   We recommend that you reserve node addresses 0 and 1 for the two 1769-SDN scanner modules used with the CompactLogix controllers within a backup system. If you do not use 0 and 1, make sure you reserve two consecutive numbers for the 1769-SDN scanner modules when you install I/O and other devices on the DeviceNet network.

2. Connect a CompactLogix controller with a 1769-SDN scanner module to the DeviceNet network.

3. Set the scanner module node address to 0 (or the lower of the two node addresses reserved for the CompactLogix controller backup system).

4. Apply power to the controller and the network.

5. Use RSNetWorx for DeviceNet software to download the network's scanlist to the 1769-SDN scanner module.

   You can either use a scanlist from a new configuration or from a previously-used configuration. If the scanlist is a new configuration, we recommend that you save it to a new project for later use.

6. Use RSLogix 5000 programming software to download the appropriate user program to the CompactLogix controller.

   The program should contain the explicit messages that enable the backup feature for this controller and scanner module. The messages are described in the Develop the CompactLogix Backup Application section beginning on page 134.

7. Put the controller into Run mode.

8. Either disable power to the controller or disconnect the scanner module from the DeviceNet network.

   This controller will be the secondary controller.

9. Connect the other CompactLogix controller with a 1769-SDN scanner module on the network.

10. Set the node address to 0.

11. Apply power to the controller and scanner module.

12. Use RSNetWorx for DeviceNet software to download the same scanlist used in step 5.

    It may be necessary to browse the network again before downloading the scanlist. This second browsing of the network allows RSNetWorx for DeviceNet software to establish communication to the new scanner module at the same node number as the previous scanner.

13. Use RSLogix 5000 programming software to download the user program to the second CompactLogix controller as you did in step 6.

    Typically, you download the same user program to the second CompactLogix controller as you did to the first. However, unlike the scanlists, the user programs in the controllers do not have to be identical.

14. Put the controller into Run mode.

    This controller is now ready to use and is the primary controller.

15. Reapply power to the secondary controller and/or reconnect the secondary scanner module to the DeviceNet subnet.

This completes the backup process. For more detailed information on some of the steps listed previously, see the next section.

Develop the CompactLogix Backup Application

The CompactLogix backup is enabled from an RSLogix 5000 programming software user program with a few simple ladder rungs. The following rungs are used in the CompactLogix backup:

- Backup Heartbeat Configuration Rungs - required
- Reading Backup State Rung - optional
- Reading Backup Status - optional

Backup Heartbeat Configuration Rungs

The first, and most critical, step is to set the backup heartbeat constant in the DeviceNet scanner. The heartbeat constant enables the backup feature and determines the switchover time (2 x heartbeat).

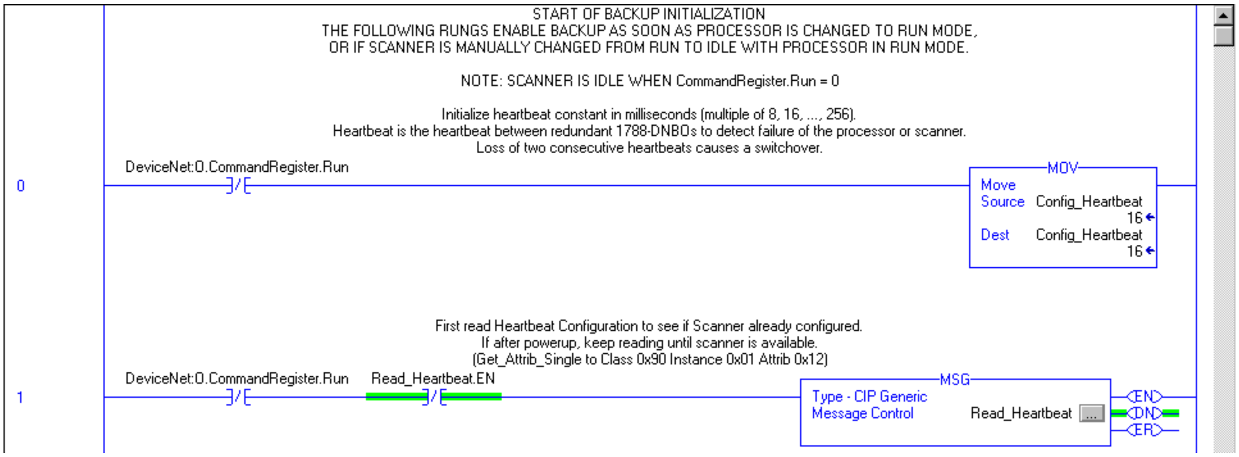
By default, the heartbeat is zero; this default value disables the Backup mode. Your user program must set the heartbeat to a non-zero value to enable backup.

The heartbeat occurs in multiples of 8 ms (8, 16, 24). We recommend a value of 16...48 ms for most applications. The recommended heartbeat times result in switchover times of 32...96 ms. These times do not include controller scan delays.

IMPORTANT If multiples of 8 are not used for the requested heartbeat, then the DeviceNet scanner module uses the next higher supported heartbeat value that can be read from the scanner. For example, if you set the heartbeat to 10, the scanner module uses a 16 ms heartbeat.

Setting the Heartbeat Constant

You can set the heartbeat constant with five rungs of ladder logic. This figure shows rungs 0 and 1 and the message setup used in rung 1. The message in rung 1 uses the INT data type.



START OF BACKUP INITIALIZATION
THE FOLLOWING RUNGS ENABLE BACKUP AS SOON AS PROCESSOR IS CHANGED TO RUN MODE,
OR IF SCANNER IS MANUALLY CHANGED FROM RUN TO IDLE WITH PROCESSOR IN RUN MODE.

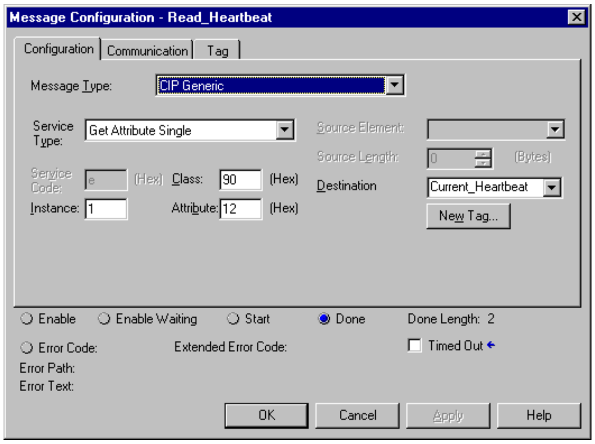
NOTE: SCANNER IS IDLE WHEN CommandRegister.Run = 0

Initialize heartbeat constant in milliseconds (multiple of 8, 16, ..., 256).
Heartbeat is the heartbeat between redundant 1788-DNBOs to detect failure of the processor or scanner.
Loss of two consecutive heartbeats causes a switchover.

First read Heartbeat Configuration to see if Scanner already configured.
If after powerup, keep reading until scanner is available.
(Get_Attrib_Single to Class 0x90 Instance 0x01 Attrib 0x12)

1
DeviceNet:O.CommandRegister.Run
Read_Heartbeat.EN
MSG
Type - CIP Generic
Message Control Read_Heartbeat ...
EN
DN
ER

Rung 1 Message Configuration and Communication tabs.



Message Configuration - Read_Heartbeat

Message Type: CIP Generic

Service Type: Get Attribute Single

Service Code: e (Hex) Class: 90 (Hex)

Instance: 1 Attribute: 12 (Hex)

Source Element:

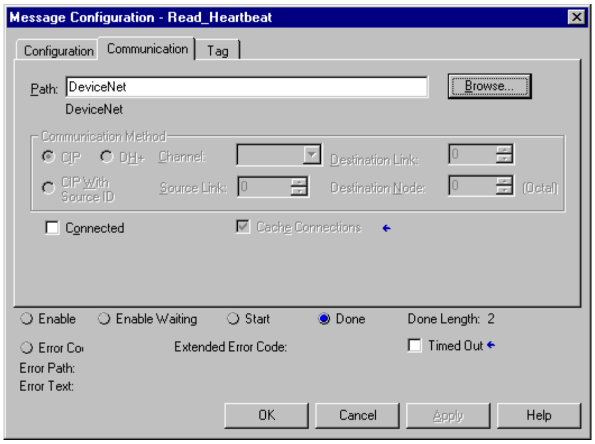
Source Length: 0 (Bytes)

Destination Current_Heartbeat

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 2

○ Error Code: Extended Error Code: □ Timed Out

Error Path:
Error Text:



Message Configuration - Read_Heartbeat

Path: DeviceNet

DeviceNet

Communication Method

● CIP ○ DH+ Channel: Destination Link: 0

○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

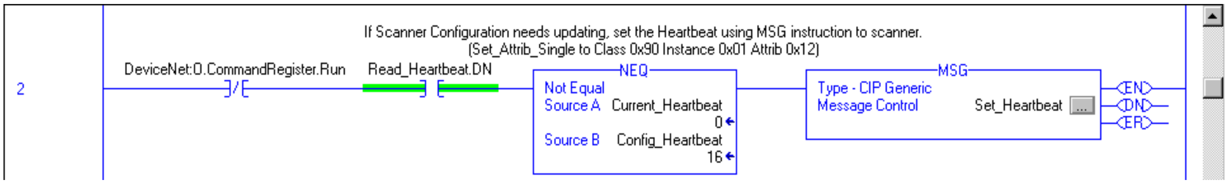
□ Connected ☑ Cache Connections

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 2

○ Error Co Extended Error Code: □ Timed Out

Error Path:
Error Text:

This figure shows rung 2 and the message setup used on it. The message in rung 2 uses the INT data type.



2

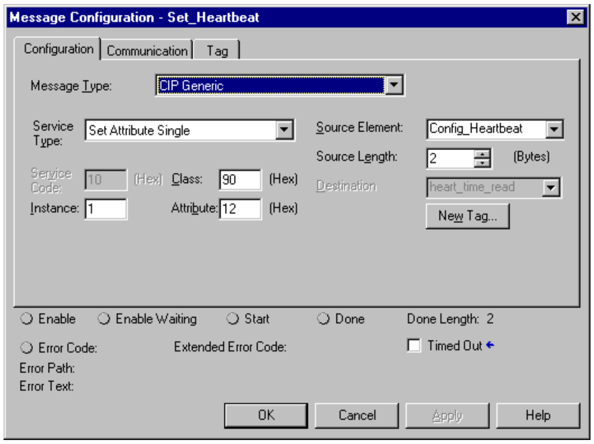
If Scanner Configuration needs updating, set the Heartbeat using MSG instruction to scanner.
(Set_Attrib_Single to Class 0x90 Instance 0x01 Attrib 0x12)

DeviceNet:0.CommandRegister.Run]/[Read_Heartbeat.DN] [

NEQ
Not Equal
Source A Current_Heartbeat
0
Source B Config_Heartbeat
16

MSG
Type - CIP Generic
Message Control Set_Heartbeat ...

Rung 2 Message Configuration and Communication tabs.



Message Configuration - Set_Heartbeat

Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex)

Class: 90 (Hex)

Instance: 1

Attribute: 12 (Hex)

Source Element: Config_Heartbeat

Source Length: 2 (Bytes)

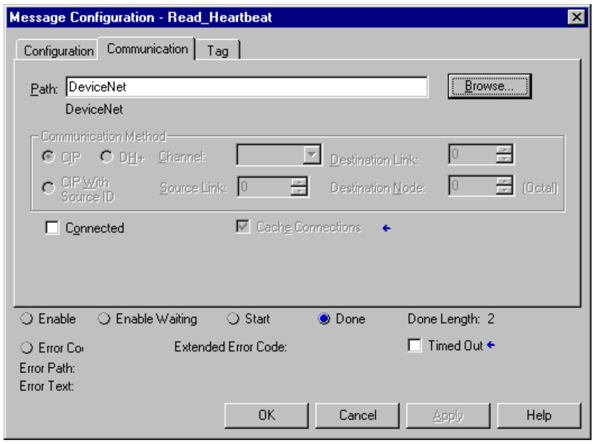
Destination: heart_time_read

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 2

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:



Message Configuration - Read_Heartbeat

Path: DeviceNet

DeviceNet

Communication Method

● CIP ○ DH+ Channel: Destination Link: 0

○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

□ Connected ☑ Cache Connections

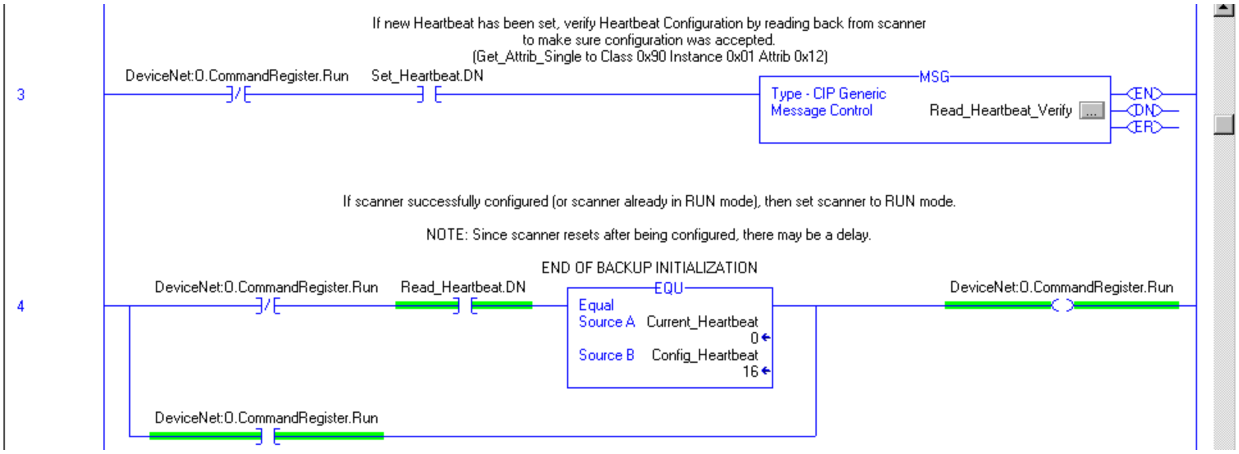
○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 2

○ Error Co Extended Error Code: □ Timed Out

Error Path:

Error Text:

This figure shows rungs 3 and 4 and the message setup used on it. The message in rung 3 uses the INT data type.



3

If new Heartbeat has been set, verify Heartbeat Configuration by reading back from scanner to make sure configuration was accepted. (Get_Attrib_Single to Class 0x90 Instance 0x01 Attrib 0x12)

4

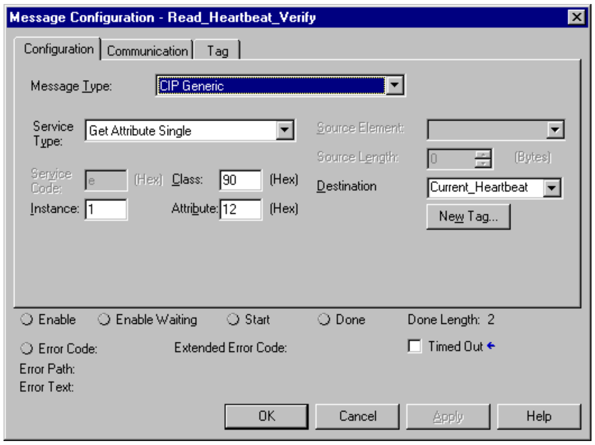
If scanner successfully configured (or scanner already in RUN mode), then set scanner to RUN mode.

NOTE: Since scanner resets after being configured, there may be a delay.

END OF BACKUP INITIALIZATION

DeviceNet:O.CommandRegister.Run
Read_Heartbeat.DN
EQU
Equal
Source A Current_Heartbeat
0
Source B Config_Heartbeat
16
DeviceNet:O.CommandRegister.Run
DeviceNet:O.CommandRegister.Run

Rung 3 Message Configuration and Communication tabs.



Message Configuration - Read_Heartbeat_Verify

Configuration | Communication | Tag

Message Type: CIP Generic

Service Type: Get Attribute Single

Service Code: e (Hex)

Class: 90 (Hex)

Instance: 1

Attribute: 12 (Hex)

Source Element:

Source Length: 0 (Bytes)

Destination: Current_Heartbeat

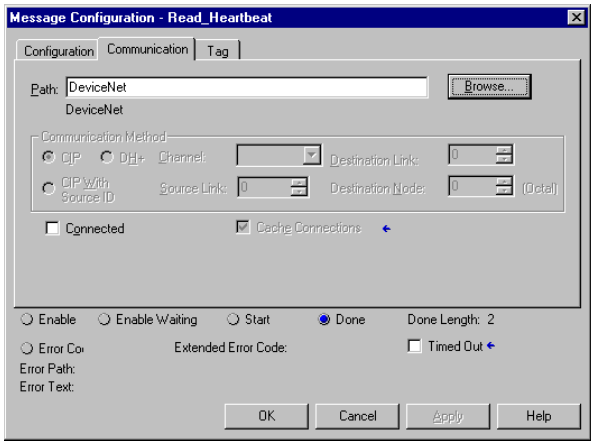
New Tag...

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 2

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:



Message Configuration - Read_Heartbeat

Configuration | Communication | Tag

Path: DeviceNet

DeviceNet

Communication Method

● CIP ○ DH+ Channel:

Destination Link: 0

○ CIP With Source ID

Source Link: 0

Destination Node: 0 (Octal)

□ Connected ☑ Cache Connections

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 2

○ Error Co Extended Error Code: □ Timed Out

Error Path:

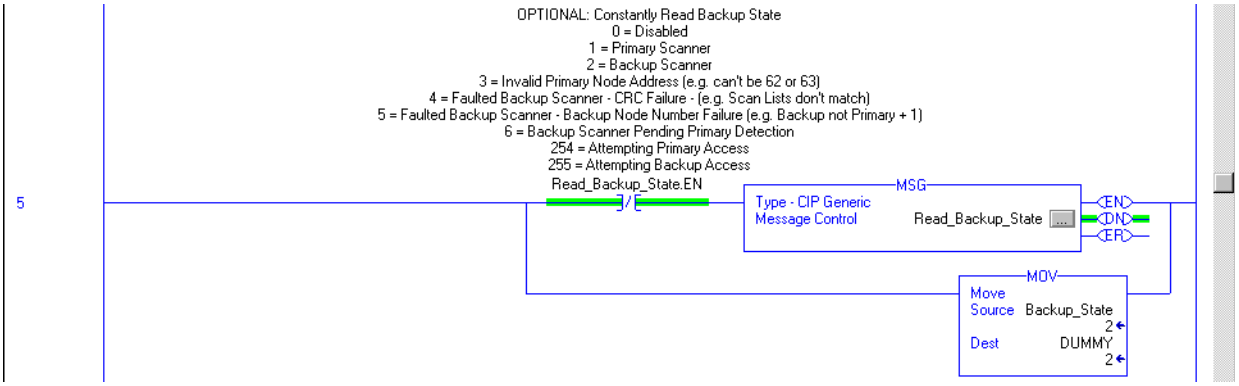
Error Text:

This completes the required portion of ladder logic to enable the CompactLogix backup on the DeviceNet network. The following sections describe how to use additional ladder logic to read backup state and status. However, these sections are not required to complete the backup solution.

Reading Backup State Rung

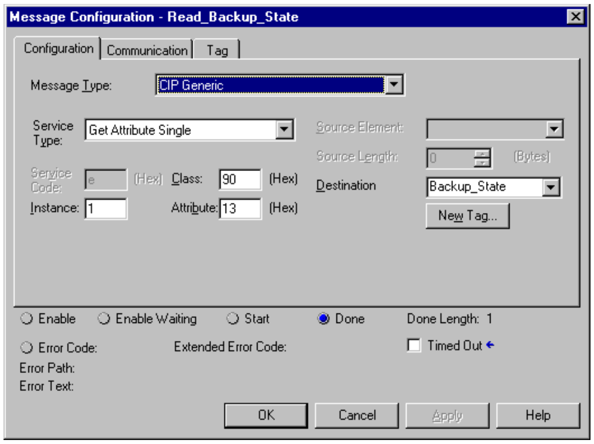
You can read the backup state of the DeviceNet scanner module with a single rung of ladder logic. The backup state is useful for debugging or more sophisticated backup schemes. The message in this rung uses the SINT data type.

This figure shows the rung you can use to read the backup state.



OPTIONAL: Constantly Read Backup State
0 = Disabled
1 = Primary Scanner
2 = Backup Scanner
3 = Invalid Primary Node Address (e.g. can't be 62 or 63)
4 = Faulted Backup Scanner - CRC Failure - (e.g. Scan Lists don't match)
5 = Faulted Backup Scanner - Backup Node Number Failure (e.g. Backup not Primary + 1)
6 = Backup Scanner Pending Primary Detection
254 = Attempting Primary Access
255 = Attempting Backup Access

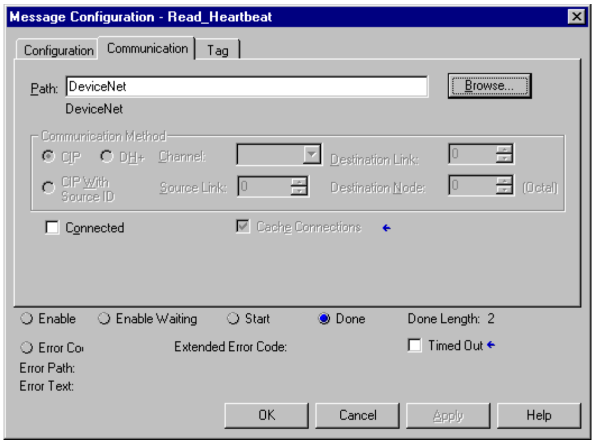
Rung 5 Message Configuration and Communication tabs.



Message Configuration - Read_Backup_State

Message Type: CIP Generic
Service Type: Get Attribute Single
Source Element:
Source Length: 0 (Bytes)
Service Code: e (Hex)
Class: 90 (Hex)
Destination: Backup_State
Instance: 1
Attribute: 13 (Hex)

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 1
○ Error Code: Extended Error Code: □ Timed Out
Error Path:
Error Text:



Message Configuration - Read_Heartbeat

Path: DeviceNet
DeviceNet

Communication Method
● CIP ○ DH+ Channel: Destination Link: 0
○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)
□ Connected ☑ Cache Connections

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 2
○ Error Co Extended Error Code: □ Timed Out
Error Path:
Error Text:

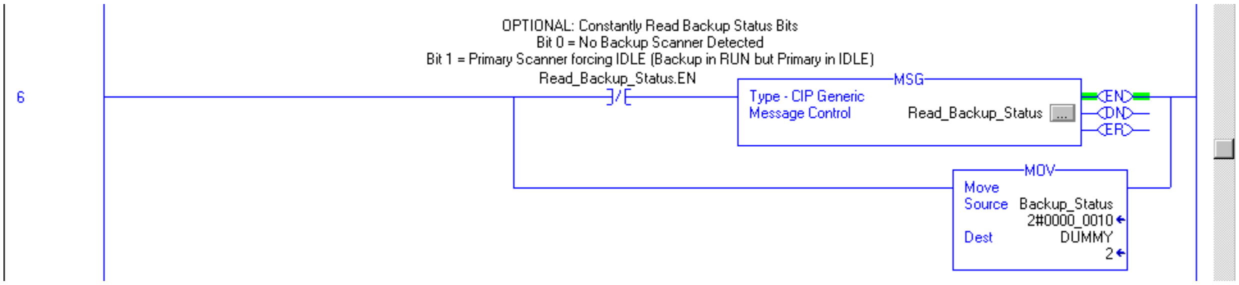
This table describes the possible values this message may return when reading the backup state of the DeviceNet scanner.

If the message reads this value	The backup state of the DeviceNet scanner module is
0	Disabled
1	Primary scanner
2	Backup scanner
3	Invalid primary node address (for example, the node address cannot be 62 or 63)
4	Faulted backup scanner module - CRC failure (for example, the scanlists in the scanners do not match)
5	Faulted backup scanner module - backup node number failure (for example, the backup scanner module is not using a node number = the primary node number + 1)
6	Backup scanner module pending primary detection
254	Attempting primary access
67	Scanner in Backup Scanner mode
68	No backup scanner module present
69	Invalid backup scanner module CRC
255	Attempting backup access

Reading Backup Status

You can read the backup status of the DeviceNet scanner module with a single rung of ladder logic. The backup state is useful for debugging or more sophisticated backup schemes. The message in this rung uses the SINT data type.

This figure shows the rung you can use to read the backup state.



6

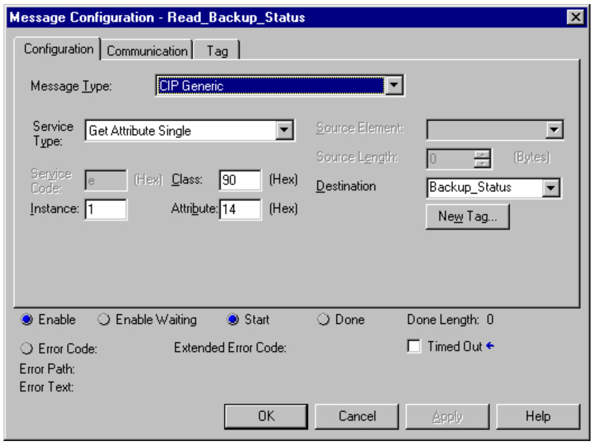
OPTIONAL: Constantly Read Backup Status Bits
Bit 0 = No Backup Scanner Detected
Bit 1 = Primary Scanner forcing IDLE (Backup in RUN but Primary in IDLE)

Read_Backup_Status.EN]/[

MSG
Type - CIP Generic
Message Control Read_Backup_Status ...
EN
DN
ER

MOV
Move
Source Backup_Status
2#0000_0010
Dest DUMMY
2

Rung 6 Message Configuration and Communication tabs.



Message Configuration - Read_Backup_Status

Configuration | Communication | Tag

Message Type: CIP Generic

Service Type: Get Attribute Single

Service Code: e (Hex) Class: 90 (Hex)

Instance: 1 Attribute: 14 (Hex)

Source Element:

Source Length: 0 (Bytes)

Destination Backup_Status

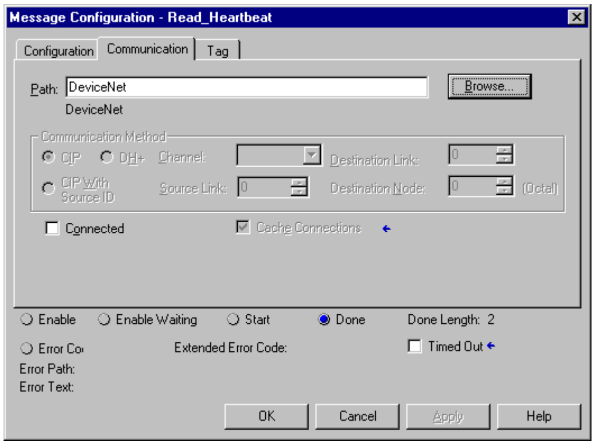
New Tag...

● Enable ○ Enable Waiting ● Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:



Message Configuration - Read_Heartbeat

Configuration | Communication | Tag

Path: DeviceNet

DeviceNet

Communication Method

● CIP ○ DH+ Channel: Destination Link: 0

○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

□ Connected ☑ Cache Connections

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 2

○ Error Co Extended Error Code: □ Timed Out

Error Path:

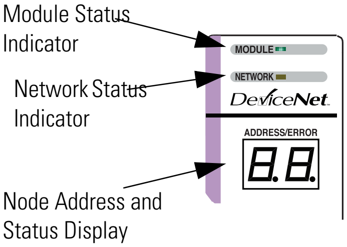
Error Text:

This table describes the possible values this message may return when reading the backup status of the DeviceNet scanner module.

If the message reads this value	The backup state of the DeviceNet scanner module is
0	No backup scanner module detected
1	Primary scanner module forcing Idle (backup in Run mode but primary in Idle mode)
3	Primary scanner in Run mode detects backup scanner in Idle mode

Using Indicators to Check Status

The 1769-SDN scanner module's status indicators provide useful information (for example, determining which scanner module is primary) about backup scanner module status.



Module Status Indicator

This table lists the indicators to monitor when checking backup status.

Module Status Indicator Behavior

If this indicator	Exhibits this behavior	This condition exists
Module status (MS)	Flashing red	• Recoverable Fault - Memory has been erased or is being programmed. • Bad firmware in module.

Node Address and Status Display

The seven-segment numeric display shows the following information about the primary and secondary scanner modules in either Run or Idle modes.

TIP If error codes other than those shown in these tables appear in the display, refer to Chapter 8, Troubleshooting, for more information on error codes.

Primary Scanner Module With Valid Backup Scanner Module

Run Mode	**Primary**	**Secondary**
Node Number	00	01 or 00
Status	00 (Run mode)	67 (Backup mode enabled) or 00 (Run mode)
Primary or Secondary	P - (Primary)	S - (Secondary)
Idle Mode		
	Primary	**Secondary**
Node Number	00	01 or 00
Status	80 (Idle mode)	67 (Backup mode enabled) or 80 (Idle mode)
Primary or Secondary	P - (Primary)	S - (Secondary)

Primary Scanner Module Without Valid Backup Scanner Module

Run Mode	**Primary**	**Secondary**
Node Number	00 or 01	01 or 00
Status	68 (No backup scanner present) or 00 (Run mode)	Undefined error or 69 (Backup scanner invalid CRC)
Primary or Secondary	P - (Primary)	--
Idle Mode		
	Primary	**Secondary**
Node Number	00 or 01	01 or 00
Status	68 (No backup scanner present) or 80 (Idle mode)	Undefined error or 69 (Backup scanner invalid CRC)
Primary or Backup	P - (Primary)	--

Lost Primary Scanner Module With Valid Backup Scanner Module

Run Mode	**Primary**	**Secondary**
Node Number	00	01 or 00
Status	Undefined error	68 (No backup scanner present) or 00 (Run mode)
Primary or Secondary	--	P - (Primary)
Idle Mode		
	Primary	**Secondary**
Node Number	00	00 or 01
Status	Undefined error	68 (No backup scanner present) or 80 (Idle mode)
Primary or Secondary	--	P - (Primary)

CRC Error

Run Mode	**Primary**	**Secondary**
Node Number	00 or 01	00 or 01
Status	68 (No backup scanner present) or 00 (Run mode)	69 (Backup scanner invalid CRC)
Primary or Secondary	P - (Primary)	--
Idle Mode		
	Primary	**Secondary**
Node Number	00 or 01	00 or 01
Status	68 (No backup scanner present) or 80 (Idle mode)	69 (Backup scanner invalid CRC)
Primary or Secondary	P - (Primary)	--

Lost Primary Scanner Module Without Valid Backup Scanner Module

Run Mode	**Primary**	**Secondary**
Node Number	Undefined error	Undefined error
Status	--	--
Primary or Secondary	--	--

Development and Debugging Tips

When you implement the CompactLogix system backup on the DeviceNet network, we recommend that you consider the following development and debugging tips:

- Develop and debug the entire application with only the primary controller and scanner module present. When the application is totally verified, then download the program and exact same scanlist to the secondary controller, without the primary controller present. Verify that the secondary is also functioning properly, and then both primary and secondary can be added to the network at the same time.

- No configuration parameters are entered from RSNetworx for DeviceNet software or RSLogix 5000 programming software to enable backup. All configuration occurs in the user program. The entire application (except for a few ladder rungs) can be developed without knowledge that the application will have a backup controller and scanner module.

- Local I/O still works when this solution is used, but the local I/O is not backed up.

- Switchover time depends on the user configurable heartbeat. Typically, switchover occurs after two heartbeats, plus the time required to complete network polling.

- The I/O during switchover is not bumpless. Since the programs and I/O updates are not synchronized, it is possible for the secondary controller to be either slightly faster or slower than the primary.

  For example, if output changes during a switchover, the fact that the primary and secondary controllers are not synchronized because the output to momentarily switch between an older and newer value. If you configure the switchover time slower than the program scan and I/O update, the secondary lags behind the primary and eliminates this.

- State variables, such as counters or timers, are not synchronized. The user program must synchronize the primary and secondary controllers, typically over an EtherNet/IP or ControlNet network link between controllers. If the outputs are dependent on a state variable, the lack of synchronization can also cause a bumpy switchover.

- As with all backup and redundancy systems, the I/O must change at a slower rate than the switchover time. If the inputs change faster than the switchover, the change of state is lost.

- Either the user program or user action determine the primary controller. In its simplest mode, the first scanner module to turn on or become available on the DeviceNet network is the primary.

- Unlike some backup systems (such as PLC-5 controller systems), the primary controller still maintains control of the I/O, and switchover does not occur if the primary controller is set to Program/Idle mode. The secondary 1769-SDN scanner module also indicates that it is in Idle mode.

- By default, a switchover will not occur if the default fault routine or user fault routine is executed in the primary controller. However, the user fault routine can force a switchover if desired.

- If an operator interface is on the DeviceNet network, it can work without knowledge of which controller is primary or secondary.

- Online edits are not automatically performed on both primary and secondary controllers, since no synchronization exists between primary and secondary controllers. Once an online edit occurs on the primary controller, the primary and secondary will have different programs.

- CompactLogix system backup on the DeviceNet network is not hot backup. Hot backup implies complete synchronization of program, program variables, and I/O. Also, I/O switchover is completely bumpless with hot backup.

Notes:

Auto-Address Recovery (AAR)

Auto-Address Recovery (AAR) allows a slave device to be removed from the network and replaced with another identical slave device that is residing on the network at node address 63 and not in the scan list. The replacement device will have its node address automatically changed to that of the device being replaced. Depending on the level of revision keying, it may be possible for the node address of the replacement device to be changed but not brought online due to a revision-keying mismatch.

Auto Device Replacement (ADR)

The Auto Device Replacement feature automates the replacement of a failed slave device on a DeviceNet network by configuring the new device to the prior level of operation. This includes Configuration Recovery (CR) and Auto-Address Recovery (AAR).

Bridge

The scanner module's support of explicit message transfer.

Change of State (COS)

A type of I/O data communication. The scanner module can send and receive data with slave devices that have the change of state feature. Data is sent whenever a data change occurs, or at the predefined heartbeat interval.

Configuration Recovery (CR)

Configuration Recovery (CR) allows a slave device to be removed from the network and replaced with an identical slave device configured at the same communication rate and Node Address as the device being replaced.

Controller

The programmable controller, for example CompactLogix or MicroLogix 1500.

Cyclic

A type of I/O data communication. The scanner module can send and receive data with slave devices that have the cyclic feature. Data is sent at a user-configurable rate.

Dual Mode

The scanner module is in Dual mode when it serves as a master to one or more slaves and as a slave to another master simultaneously.

Electronic Data Sheet (EDS)

A vendor-supplied template that specifies how information is displayed as well as what is an appropriate entry (value).

Explicit Messaging

A type of messaging used for lower priority tasks, such as configuration and data monitoring.

Heartbeat Rate

Devices that are configured for change of state data will send data at this rate if no data change occurs.

Host Platform

The computer on which the application software is run.

I/O

Input and output modules

Input Data

Data produced by a DeviceNet device and collected by the scanner module for the host platform to read.

MAC ID

The network address of a DeviceNet node. Also referred to as node address.

Multicast

Used when describing a strobe message.

Network

The DeviceNet network or the RSNetWorx for DeviceNet software representation of the network.

Node

Hardware that is assigned a single address on the network. Also referred to as a device.

Offline

When the host platform is not communicating on the network.

Online

When the host platform is configured and enabled to communicate on the network.

Output Data

Data produced by the host platform that is written to the scanner module's memory. This data is sent by the scanner module to DeviceNet devices.

Point-to-Point

Used when describing a poll message. The message solicits a response from a single, specified device on the network.

Polled

A type of input/output-data communication. A polled message solicits a response from a single, specified device on the network (a point-to-point transfer of data).

Record

The node address and channel-specific memory assigned in the scanner module's nonvolatile storage for a node in the scanlist.

Rx

Abbreviation for receive

Scanlist

The list of devices (nodes) with which the scanner module is configured to exchange I/O data.

Scanner

The function of the 1769-SDN scanner module to support the exchange of I/O with slave modules.

Slave Mode

The 1769-SDN scanner module is in slave mode when it is placed in another DeviceNet master's scanlist as a slave device.

Strobed

A type of I/O data communication. A strobed message solicits a response from each strobed device (a multicast transfer). It is a 64-bit message that contains one bit for each device on the network.

Tx

Abbreviation for transmit

How Are We Doing?

Your comments on our technical publications will help us serve you better in the future.
Thank you for taking the time to provide us feedback.

You can complete this form and mail (or fax) it back to us or email us at RADocumentComments@ra.rockwell.com.

Pub. Title/Type 1769-SDN DeviceNet Scanner Module

Cat. No. *1769-SDN* Pub. No. 1769-UM009E-EN-P Pub. Date August 2009 Part No.

Please complete the sections below. Where applicable, rank the feature (1=needs improvement, 2=satisfactory, and 3=outstanding).

Overall Usefulness	1 2 3	How can we make this publication more useful for you?
Completeness (all necessary information is provided)	1 2 3	Can we add more information to help you? procedure/step illustration feature example guideline other explanation definition
Technical Accuracy (all provided information is correct)	1 2 3	Can we be more accurate? text illustration
Clarity (all provided information is easy to understand)	1 2 3	How can we make things clearer?
Other Comments		You can add additional comments on the back of this form.

Your Name

Your Title/Function

Location/Phone

Would you like us to contact you regarding your comments?

___No, there is no need to contact me

___Yes, please call me

___Yes, please email me at ________________________

___Yes, please contact me via ________________________

Return this form to: Rockwell Automation Technical Communications, 1 Allen-Bradley Dr., Mayfield Hts., OH 44124-9705

Email: RADocumentComments@ra.rockwell.com

PLEASE FASTEN HERE (DO NOT STAPLE)

Other Comments

PLEASE FOLD HERE

PLEASE REMOVE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 18235 CLEVELAND OH

POSTAGE WILL BE PAID BY THE ADDRESSEE

Rockwell Automation

1 ALLEN-BRADLEY DR
MAYFIELD HEIGHTS OH 44124-9705

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://support.rockwellautomation.com, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration, and troubleshooting, we offer TechConnect support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://support.rockwellautomation.com.

Installation Assistance

If you experience a problem within the first 24 hours of installation, please review the information that's contained in this manual. You can also contact a special Customer Support number for initial help in getting your product up and running.

United States	1.440.646.3434 Monday – Friday, EST
Outside United States	Please contact your local Rockwell Automation representative for any technical support issues.

New Product Satisfaction Return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

United States	Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for the return procedure.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

User Manual

Allen-Bradley

DeviceNet Network Configuration

1756 ControlLogix, 1756 GuardLogix, 1769 CompactLogix, 1769 Compact GuardLogix, 1789 SoftLogix, Studio 5000 Logix Emulate

Allen-Bradley • Rockwell Software

Rockwell Automation

Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid-state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

⚠	**WARNING:** Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
⚠	**ATTENTION:** Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence
⚡	**SHOCK HAZARD:** Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
♨	**BURN HAZARD:** Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.

This manual contains new and updated information. Changes throughout this revision are marked by change bars, as shown to the right of this paragraph.

New and Updated Information

This table contains the changes made to this revision.

Topic	Page
Updated supported controller models.	Cover

Notes:

Preface

Chapter 1
DeviceNet Overview

Chapter 2
Connect a Computer to the DeviceNet Network

Chapter 3
Connect Devices to the Network

Chapter 4
Configure the Network Offline

Chapter 5

Configure the Network Online

Chapter 6

Automatically Configure a DeviceNet Network

Chapter 7

Control a Device

Chapter 8

Interlock and Share Inputs

Chapter 9

Communicate with a PanelView Standard Terminal

Chapter 10

Communicate with a FactoryTalk View Project

Chapter 11

Tune the Performance of a DeviceNet Network

Chapter 12

Automate the Replacement of a Failed Device

Appendix A

Map the Memory Location with Advanced Mapping

Index

Notes:

This manual describes how you can use DeviceNet modules with your Logix5000 controller and communicate with various devices on the DeviceNet network.

You should use this manual if you program applications that use DeviceNet with one of these Logix5000 controllers:

- 1756 ControlLogix controllers
- 1768 CompactLogix controllers
- 1769 CompactLogix controllers
- 1789 SoftLogix 5800 controllers
- PowerFlex 700S with DriveLogix controllers

You should also understand the following:

- Networking concepts
- RSNetWorx for DeviceNet software
- RSLogix 5000 programming software
- RSLinx Classic communication software

Additional Resources

For more information on the products included in this publication, use the publications listed in this table.

Resource	Description
DeviceNet Modules Installation Instructions, publication DNET-IN001	Describes how to install and set up 1756-DNB, 1769-ADN, and 1769-SDN DeviceNet modules.
DeviceNet Media Design Installation Guide, publication DNET-UM072	Describes how to design, install, and troubleshoot a DeviceNet cable system.
Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001M	Links to a collection of programming manuals that describe how you can use procedures that are common to all Logix5000 controller projects.

Network Configuration

The following chapters describe how to set up a DeviceNet network:

- Chapter 2—Connect a Computer to the DeviceNet Network
- Chapter 3—Connect Devices to the Network
- Chapter 4—Configure the Network Offline
- Chapter 5—Configure the Network Online

You **are not required to** complete all tasks in each chapter in the **exact order** presented to set up your DeviceNet application. For example, you can configure your network offline before you connect a computer to the network.

However, there are **some requirements** related to the order in which you complete tasks. For example, you must complete the tasks in chapters 2 and 3 before you can configure the network online.

The following table describes optional and required conditions to consider when determining the order in which you plan to complete tasks in your DeviceNet application.

Network Configuration Tasks

Task	Optional Conditions	Required Conditions
Connect a computer to the network	• Can be completed before or after connecting devices to the network • Can be completed before or after configuring the network **offline**	Must be completed before configuring the network **online**
Connect devices to the network	• Can be completed before or after connecting a computer to the network • Can be completed before or after configuring the network **offline**	Must be completed before configuring the network **online**
Configure the network offline	• Can be completed before or after connecting a computer to the network • Can be completed before or after connecting devices to the network • Can be completed before configuring the network online	None
Configure the network online	Can be completed without creating a network configuration file offline	• Computer must be connected to the network before configuring the network online • Devices must be connected to the network before configuring the network online

DeviceNet Overview

The Logix5000 family of controllers operates with many DeviceNet communication modules. This chapter describes each communication module and the preliminary tasks you must complete before your configure and program the DeviceNet network.

DeviceNet communication modules share these features:

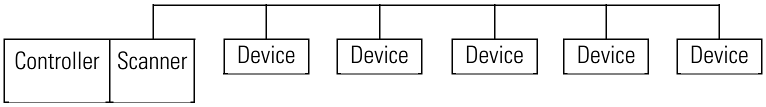
- Interface via cabling systems using either round or flat media that provide both power and communication
- Use network protocols
- Require no network scheduling
- Support messaging, produced/consumed data, and distributed I/O

Choose a Single Network or Subnets

You can organize the devices on the network in a single network or several, smaller distributed networks known as subnets.

Single Network

When you use a single network, you place all your devices on a single DeviceNet network and connect the controller directly to the network via a scanner. The following graphic shows a single network.



Advantages to Using a Single Network

There are advantages to using a single network for your DeviceNet application:

- The overall cost to install the network is lower than using subnets.
- You need to manage only a single network.
- The Logix5000 controller is local to the DeviceNet scanner. For example, with a single network in a ControlLogix application, the 1756-L64 controller is in the same ControlLogix chassis as the 1756-DNB scanner.

Disadvantages to Using a Single Network

There are disadvantages to using a single network for your DeviceNet application:

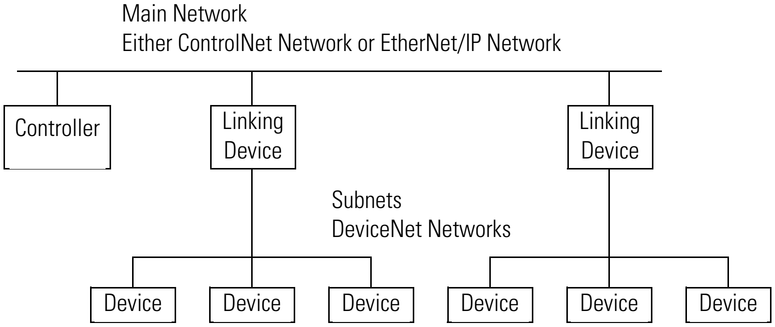
- The network must use shorter distances from one end to another.
- The more devices on the network, the slower the overall performance of the network.
- Your network may have more power supply requirements than can be handled by one network
- A single network can contain only up to 64 nodes

Subnets

A subnet configuration is a main network that is connected to distributed subnets using a scanner, or linking device. In this option, you must install a ControlNet network or EtherNet/IP network, also known as a backbone, that connects to distributed subnets using a linking device.

For example, if you choose an EtherNet/IP network backbone, you must use 1788-EN2DN linking devices to connect the subnets.

The following graphic shows a subnet network.



Advantages to Using Subnets

There are advantages to using subnets for your DeviceNet application:

- Typically, there are shorter runs on subnets, which allow a faster communication rate for the DeviceNet network.
- With fewer devices on each subnet, the overall performance of the network is faster.
- There are simpler power requirements.

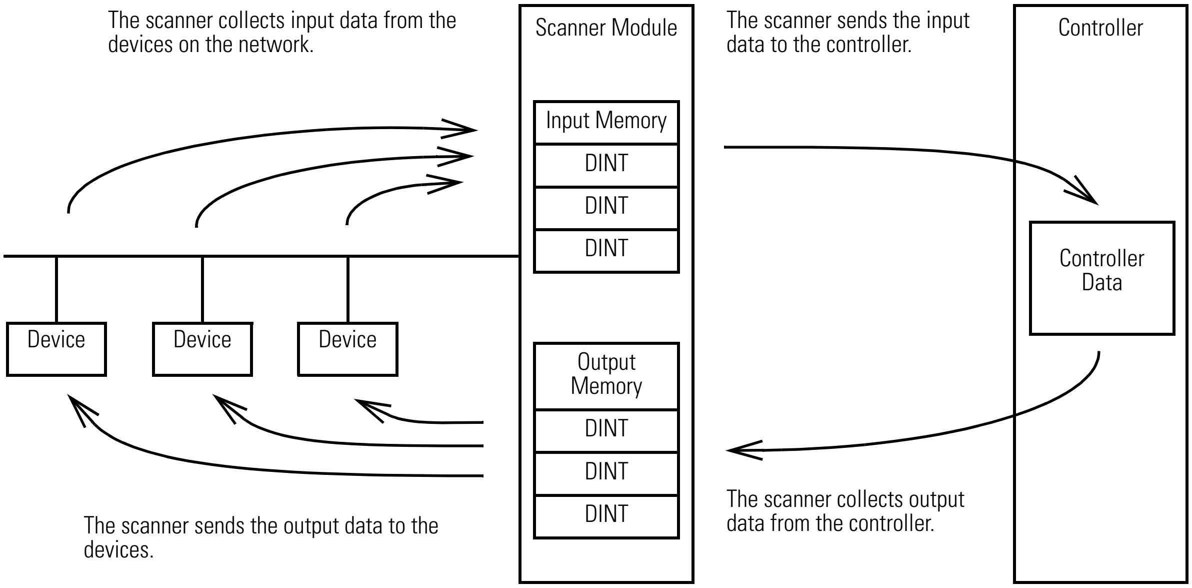
Disadvantages to Using Subnets

There are disadvantages to using subnets for your DeviceNet application:

- The overall cost to install the network is higher than using a single network.
- You must manage multiple networks.
- The Logix5000 controller is remote from the linking device. For example, with subnets in a 1768 CompactLogix application, a 1768-L45 controller is remote from the 1788-CN2DN linking device.

Choose a Scanner

The DeviceNet scanner connects a Logix5000 controller to the devices on a DeviceNet network. The following graphic shows how a scanner exchanges data between a controller and devices on the DeviceNet network.


The scanner collects input data from the devices on the network.
Scanner Module
Input Memory
DINT
DINT
DINT
Output Memory
DINT
DINT
DINT
The scanner sends the input data to the controller.
Controller
Controller Data
Device
Device
Device
The scanner sends the output data to the devices.
The scanner collects output data from the controller.

The following table describes how to choose a scanner.

If you are using	And	Use this scanner
Single network	1768 or 1769 CompactLogix controller	CompactLogix 1769-SDN modules
	ControlLogix controller	ControlLogix 1756-DNB modules
	DriveLogix controller	1788-DNBO DeviceNet daughtercard
	SoftLogix 5800 controller	1784-PCIDS card
Subnets	EtherNet/IP main network	EtherNet/IP to DeviceNet Linking Device 1788-EN2DN
	ControlNet main network	ControlNet to DeviceNet Linking Device 1788-CN2DN

Bridge Across Networks

Logix5000 controllers can usually communicate with devices on other networks with no additional configuration or programming. A bridge connects two networks.

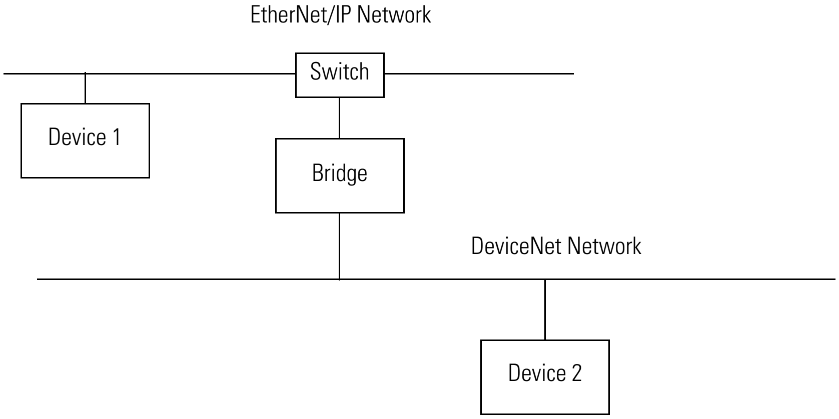
IMPORTANT You cannot bridge from a device on a DeviceNet network to a device on a ControlNet nor EtherNet/IP network. You can only bridge from devices on ControlNet or EtherNet/IP networks to devices on DeviceNet networks.

Refer to table Bridging Across Networks on page 17 for more information.

The bridge is one of the following:

- A single device with communication ports for two different networks, such as a 1788-EN2DN linking device
- A separate communication device in the same chassis

For example, the bridge device shown in the following graphic is connected to both EtherNet/IP and DeviceNet networks. Device 1 on an EtherNet/IP network can communicate with Device 2 on a DeviceNet network through the bridge.



The following table describes how communication can bridge the networks.

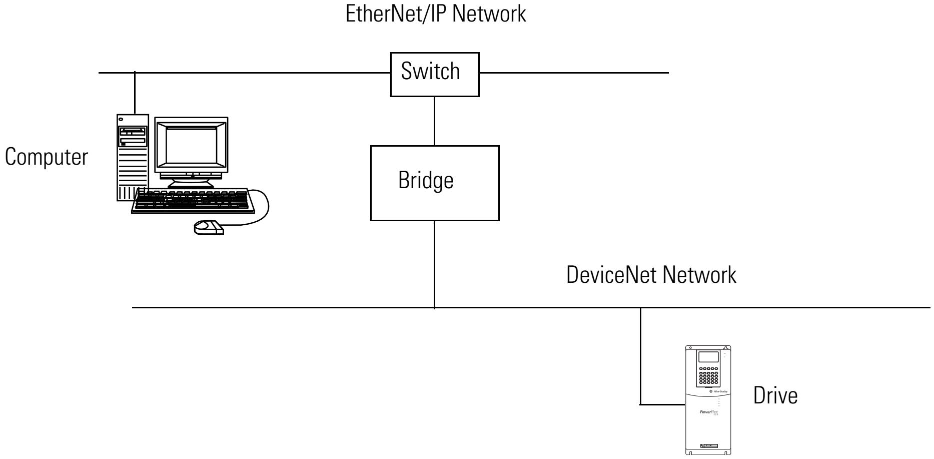
Bridging Across Networks

A device on this network	Can access a device on this network			
	EtherNet/IP	**ControlNet**	**DeviceNet**	**RS-232**[2]
EtherNet/IP	yes	yes	yes	yes
ControlNet	yes	yes	yes	yes
DeviceNet	no	no	yes	no
RS-232	yes	yes[1]	yes	yes

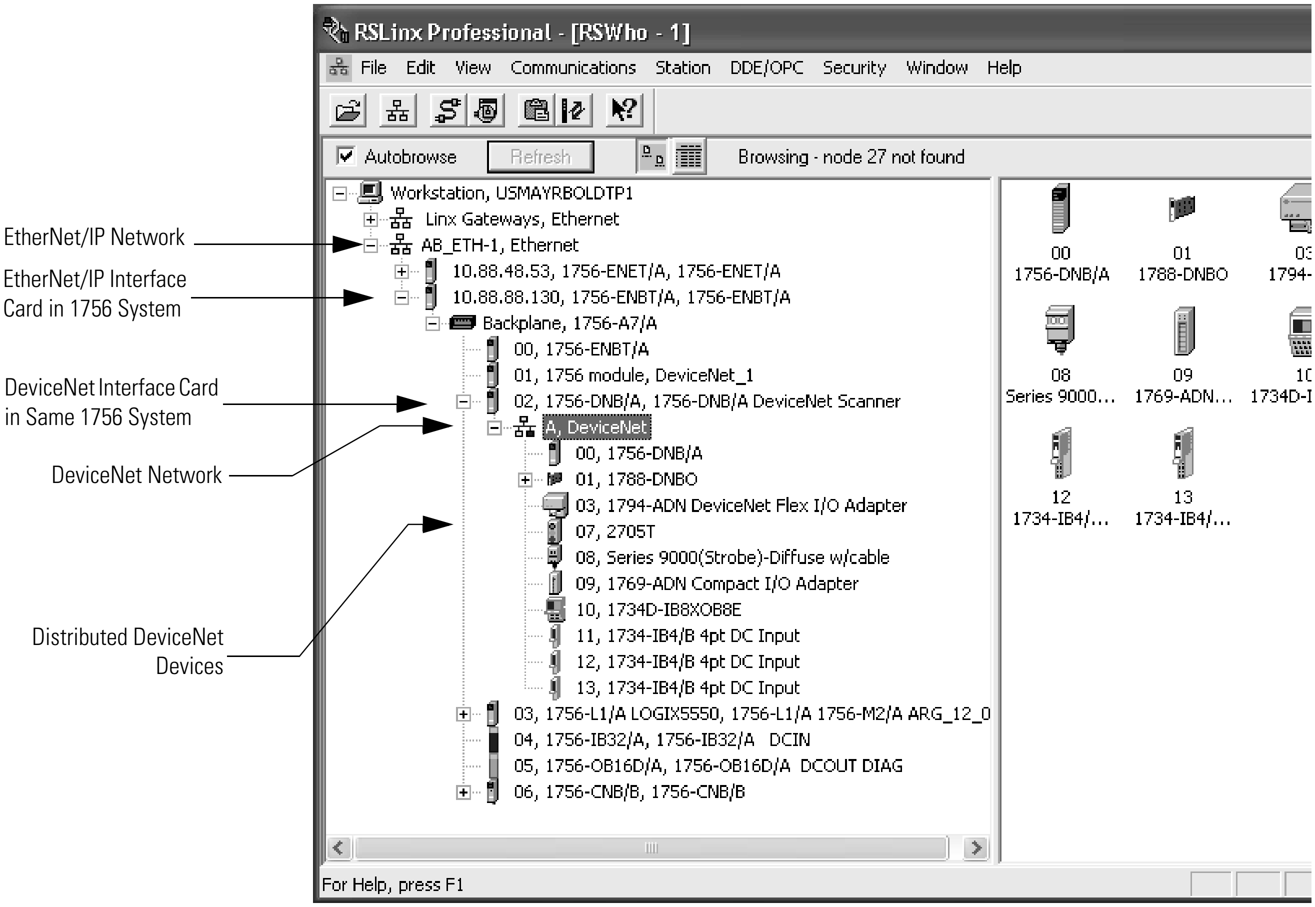
[1] To use RSNetWorx for ControlNet software to configure and schedule a ControlNet network, we recommend that you complete one of the following tasks.
- Connect to an EtherNet/IP network and bridge to the ControlNet network.
- Use a 1784-U2CN interface device to connect directly to the ControlNet network.

[2] Typically, this is a point-to-point connection between a Logix5000 controller and another device, such as a PanelView Plus operator terminal.

In this example, a computer configures a drive on a DeviceNet network. The workstation bridges an EtherNet/IP network to reach the drive.



In this example, the RSLinx communication software window shows how the DeviceNet bridge links to the EtherNet/IP network.



☑ Autobrowse

Browsing - node 27 not found

```
Workstation, USMAYRBOLDTP1
  Linx Gateways, Ethernet
  AB_ETH-1, Ethernet
    10.88.48.53, 1756-ENET/A, 1756-ENET/A
    10.88.88.130, 1756-ENBT/A, 1756-ENBT/A
      Backplane, 1756-A7/A
        00, 1756-ENBT/A
        01, 1756 module, DeviceNet_1
        02, 1756-DNB/A, 1756-DNB/A DeviceNet Scanner
          A, DeviceNet
            00, 1756-DNB/A
            01, 1788-DNBO
            03, 1794-ADN DeviceNet Flex I/O Adapter
            07, 2705T
            08, Series 9000(Strobe)-Diffuse w/cable
            09, 1769-ADN Compact I/O Adapter
            10, 1734D-IB8XOB8E
            11, 1734-IB4/B 4pt DC Input
            12, 1734-IB4/B 4pt DC Input
            13, 1734-IB4/B 4pt DC Input
        03, 1756-L1/A LOGIX5550, 1756-L1/A 1756-M2/A ARG_12_0
        04, 1756-IB32/A, 1756-IB32/A   DCIN
        05, 1756-OB16D/A, 1756-OB16D/A  DCOUT DIAG
        06, 1756-CNB/B, 1756-CNB/B
```

For Help, press F1

Choose a Baud Rate for the Network

You must choose a baud rate for the DeviceNet network. There are three rates available for the network:

- 125 kbps—This is the default baud rate for a DeviceNet network. It is the easiest baud rate to use and is usually sufficient.
- 250 kbps
- 500 kbps

The following table describes the most common methods to set a baud rate.

Method	Description
Autobaud feature	At powerup, the device automatically sets its baud rate to the baud rate of the first device it hears on the network. The baud rate remains set until the device powers up again. The network requires at least one device with a fixed baud rate so the autobaud devices have something against which to set. Typically, scanners and network interfaces have a fixed baud rate.
Switches or push button on the device	Some devices have switches or push buttons that set the baud rate as follows: • The device reads the switch setting at powerup. • Typically, the switch lets you select either autobaud or a fixed baud rate, that is 125 Kbps, 250 Kbps, or 500 Kbps • If you change the switch setting, you must cycle power to the device before the change takes effect. There are exceptions. For example, the 1756-DNB module has a push button that only lets you set the baud rate if the module is disconnected from the network or network power is off. Once you change the baud rate, the module automatically resets to the new baud rate.
Software	Some devices require a programming device to set its address. For example, you can use the computer and the DeviceNet node commissioning tool to set the baud rate of a device. The node commissioning tool is available in either of the following methods: • Automatically when you install RSNetWorx for DeviceNet software • As a separate application on the RSLogix 5000 programming software CD, revision 13.0 or later

The length of the trunkline and type of cable determines which baud rates you can use.

Baud Rate	Maximum Distance			Cumulative Drop Line Length
	Flat Cable	Thick Cable	Thin Cable	
125K bit/s	420m (1378 ft)	500m (1640 ft)	100m (328 ft)	156 m (512 ft)
250K bit/s	200m (656 ft)	250m (820 ft)	100m (328 ft)	78m (256 ft)
500K bit/s	75m (246 ft)	100m (328 ft)	100m (328 ft)	39m (128 ft)

If you change the baud rate of the network, make sure that all devices change to the new baud rate. Mixed baud rates produce communication errors.

Complete the following steps to set the baud rate for the network.

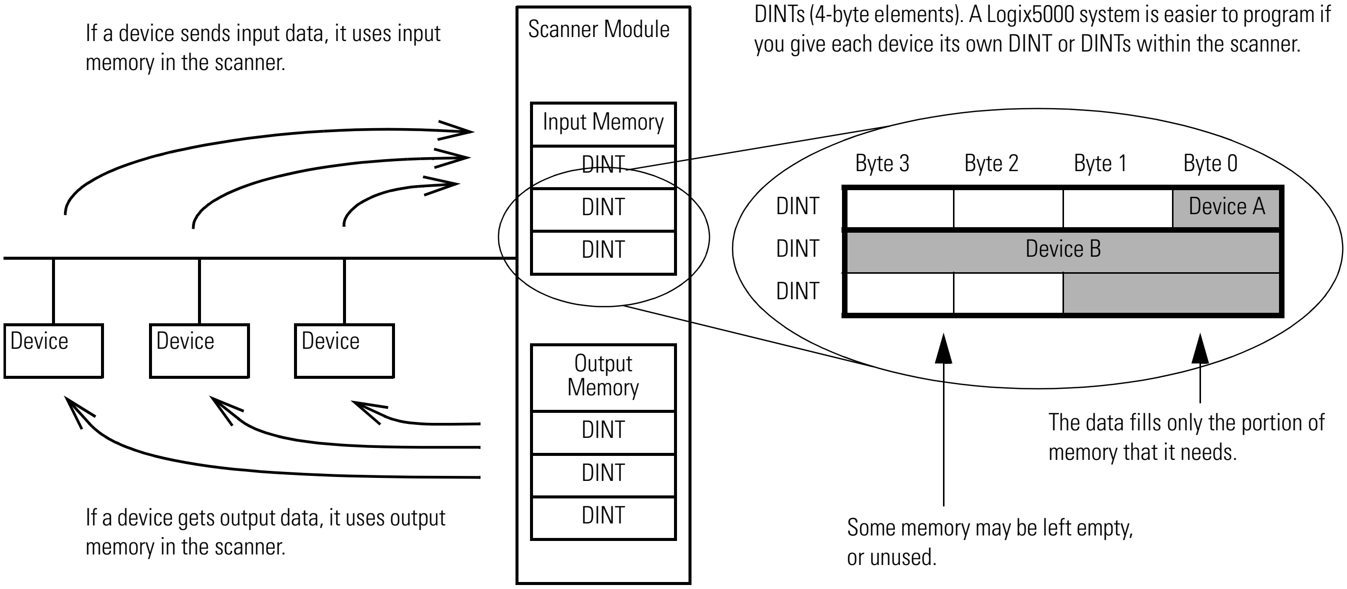
1. Connect the network interface to the network and set its baud rate.
2. Connect the scanner to the network and set its baud rate.
3. For each device that has **only** fixed baud rates (**no** autobaud), set the baud rate and connect it to the network.

4. Connect the remaining devices to the network and enable autobaud for each of them.

If a device	Then
has a switch to enable autobaud	1. Set the switch to autobaud. 2. Connect the device to the network.
does **not** have a switch to enable autobaud	1. Connect the device to the network. 2. Use RSNetWorx for DeviceNet software to enable autobaud.

Calculate Scanner Memory Requirements

A Logix5000 scanner has fixed sections of memory for the network's input and output data. Each device on the network requires either scanner input or output memory. Devices that send and receive data need both input and output memory.



If a device sends input data, it uses input memory in the scanner.

The memory of a Logix5000 scanner is organized as an array of DINTs (4-byte elements). A Logix5000 system is easier to program if you give each device its own DINT or DINTs within the scanner.

The data fills only the portion of memory that it needs.

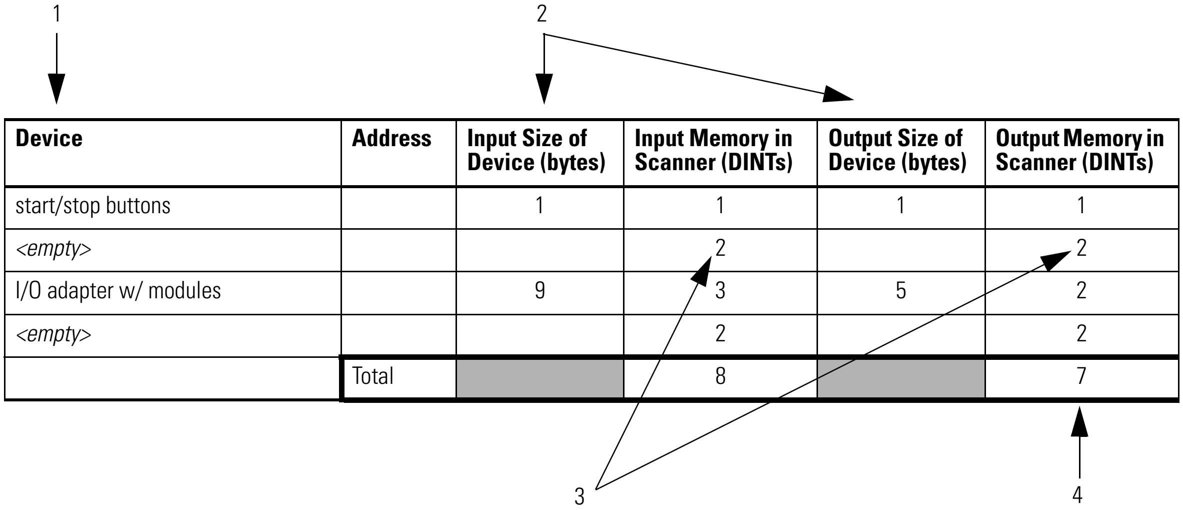
If a device gets output data, it uses output memory in the scanner.

Some memory may be left empty, or unused.

To make sure the network is within limits, calculate the amount of input and output memory that the scanner needs. This information is useful when you configure the scanner.

Complete the following steps to calculate how much scanner memory you need.

1. List the devices on the network.
2. Record how many bytes each device sends to your control system as input data and gets from your control system as output data.
3. Convert the input and output sizes to DINTs where DINTs = (bytes/4) rounded up to an integer.
4. Determine the total memory that you need in your scanner.



Device	**Address**	**Input Size of Device (bytes)**	**Input Memory in Scanner (DINTs)**	**Output Size of Device (bytes)**	**Output Memory in Scanner (DINTs)**
start/stop buttons		1	1	1	1
<empty>			2		2
I/O adapter w/ modules		9	3	5	2
<empty>			2		2
	Total		8		7

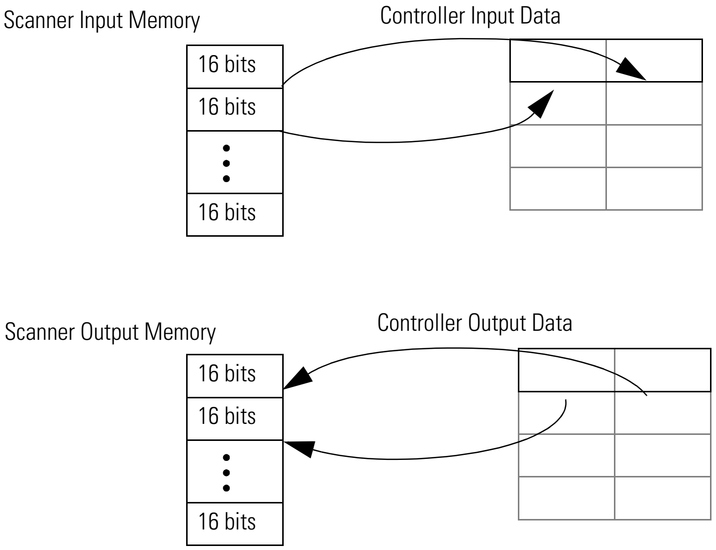
Check the I/O Limits of the Scanner

Once you tally the input and output data for the network, make sure it is within the limits of the scanner. If they exceed the limits, use multiple scanners.

Scanner	**Maximum input data (DINTs)**	**Maximum output data (DINTs)**
1756-DNB	124	123
1769-SDN	90	90
1788-CN2DN	124	123
1788-EN2DN	124	123
1788-DNBO	124	123

When Using a SoftLogix 5800 Controller

The 1784-PCIDS universal PCI scanner card organizes its input and output memory in 16-bit increments. When you access the data in the controller, the data is packed into 32-bit increments (DINTs). This means that two 16-bit scanner data locations will fit into a single 32-bit controller data location.



A Logix5000 system is easier to program if you give each device its own DINT or DINTs within the controller. To accomplish this with a 1784-PCIDS scanner, allocate memory in 4-byte increments. This may result in some 16-bit words being left unused.

Assign an Address to Each Device

To communicate on the DeviceNet network, each device requires its own address. In general, a device can use any address in the range of 0…63. However, we recommend that you follow the guidelines in the following table.

Give this device	This address	Notes
Scanner	0	If you have multiple scanners, give them the lowest addresses in sequence (0, 1…).
Any device on the network except the scanner	1…61	• Give the lower addresses to devices with 15 bytes or more of input or output data. • Gaps between addresses are OK and have no effect on system performance. If you are uncertain of the final lay-out of your system, leave gaps between addresses. This gives you some flexibility as you develop your system.
Computer interface to the network	62	If you connect a computer directly to the DeviceNet network, use address 62 for the computer. • Many computer interface devices use this address as their default. • The 1784-U2DN device can connect a computer directly to a DeviceNet network.
No device	63	Always leave address 63 open. Out of the box, most DeviceNet devices are preset for address 63. • Some devices have no switches or push button to set the address. They require software, such as RSNetWorx for DeviceNet software to change the address. This means that you must first place it on the network at its preset address of 63 before you can change the address. • If another device is already using address 63, there will be an address conflict and you won't be able to communicate with the newly connected device. • Leaving address 63 open makes it possible to configure a new device. • The auto-address recovery feature also requires address 63 to be open.

Refer to Chapter 3 on page 29 for more information on how to assign an address to each device.

Required Software

You must use the correct software with the DeviceNet application.

- To connect your computer to the DeviceNet network, use RSLinx communication software.

- To program the Logix5000 controller, use RSLogix 5000 programming software.

- To the configure the DeviceNet network, use RSNetWorx for DeviceNet software.

Notes:

Connect a Computer to the DeviceNet Network

This chapter shows how to connect a computer to the network. After you physically connect a computer to the network, you must configure a driver in RSLinx communication software to communicate over the network.

After you connect a computer to the network and configure a driver in RSLinx communication software, you can complete these tasks:

- Configure the devices on the network
- Configure network parameters
- Upload, download, monitor, and program projects for Logix5000 controllers

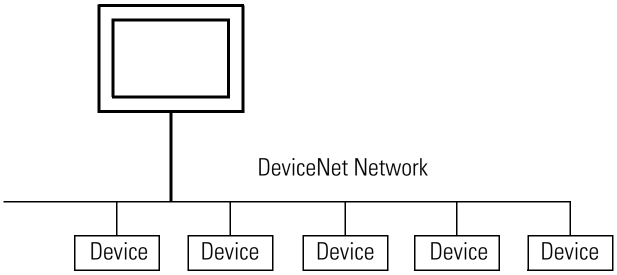
Some networks let you bridge to other networks in your system. This lets you connect to one network and access devices or controllers on other networks.

Connection Options

To access the DeviceNet network, do one of the following:

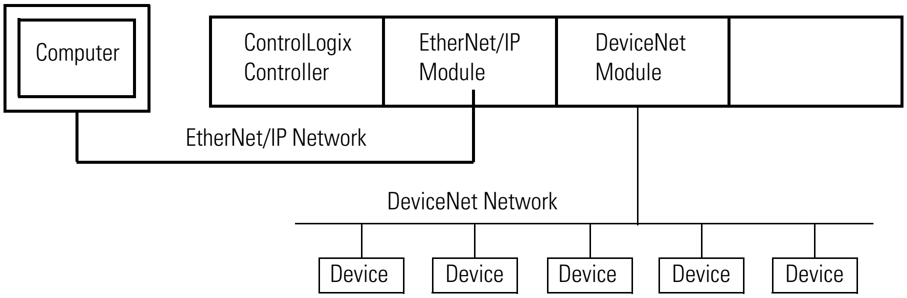
- Connect directly to the network via the 1784-U2DN interface device. If you connect directly to a DeviceNet network, you can access only the devices on that network. If you use this method, refer to Set Up the DeviceNet Driver on page 27.

  The following graphic shows a computer connected directly to a DeviceNet network.



- Connect to a different network and bridge to the desired DeviceNet network. This requires **no** additional programming.

  The following graphic shows a computer connected to a DeviceNet network through an EtherNet/IP network used with a ControlLogix system.



For more information about installing modules on the DeviceNet network, refer to the Rockwell Automation Literature Library at the following address:

http://www.rockwellautomation.com/literature/

To find the installation publications specific to your module, search by the module's catalog number.

Set Up the DeviceNet Driver

The requirements for setting up the DeviceNet driver depend on your version of RSLinx Classic software.

RSLinx Classic Software Version	Action
2.50 or earlier	Upgrade to version 2.51 or later to use the 1784-U2DN interface device.
2.51, 2.52, or 2.53	Proceed to Obtain the Driver for the Interface Device on page 27.
2.54 or later	The 1784-U2DN interface device driver is already installed on the computer. Proceed to Verify that the Driver Works on page 27.

Obtain the Driver for the Interface Device

Follow these steps to download and install the device driver for the 1784-U2DN interface device.

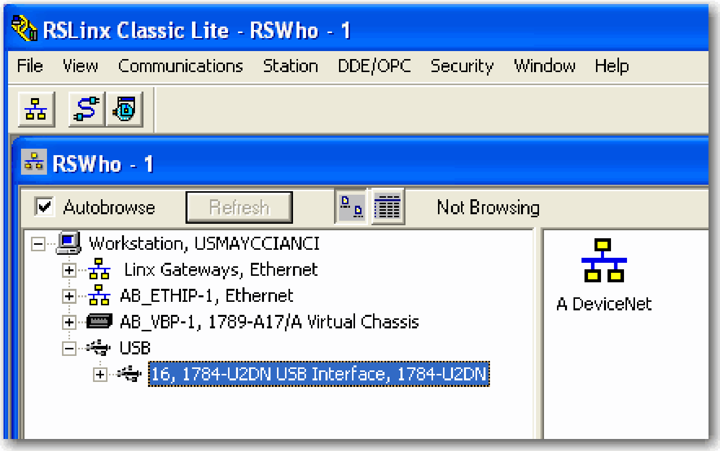
1. Visit http://www.rockwellautomation.com/knowledgebase/.

2. Open tech note ID 53280 and follow the instructions to install the driver.

Verify that the Driver Works

Complete the following steps to verify that the driver for the 1784-U2DN interface device works.

1. Open RSLinx Classic software.
2. Click the Browse button and verify that the 1784-U2DN interface appears under USB.



Workstation, USMAYCCIANCI
Linx Gateways, Ethernet
AB_ETHIP-1, Ethernet
AB_VBP-1, 1789-A17/A Virtual Chassis
USB
16, 1784-U2DN USB Interface, 1784-U2DN

Notes:

Chapter 3

Connect Devices to the Network

This chapter describes how to connect a device to the network and set the device's address so it can communicate on the DeviceNet network.

Before You Begin

Be sure that all communication modules have been installed. Also, have a list of the devices that you are putting on the network and an address for each of them. The following table shows an example list of devices.

Device	Address	Input Size of Device (bytes)	Input Memory in Scanner (DINTs)	Output Size of Device (bytes)	Output Memory in Scanner (DINTs)
scanner	0	n/a	n/a	n/a	n/a
PanelView terminal	3	128	32	128	32
<empty>			2		2
I/O adapter w/ modules	5	9	3	5	2
<empty>			2		2
drive	7	4	1	4	1
<empty>			2		2
photoeye	9	1	1	0	0
computer interface	62	n/a	n/a	n/a	n/a
	63				
	Total		43		41

Set the Node Address of a Device

You can use the following options to set the node address of DeviceNet devices. However, not all options apply to every DeviceNet device.

For example, you can use all three options with the 1756-DNB ControlLogix DeviceNet scanner, but you can use only the second and third methods with the 1769-SDN Compact I/O DeviceNet scanner.

- Set Node Address via Hardware Mechanism
- Set Node Address via Software
- Set Node Address via DeviceNet Node Commissioning Tool

All DeviceNet devices ship with their node addresses set to 63. To avoid duplicate node number conditions on the network, you should change the node address for each device to a unique number as you add it to the network.

Give this address	To this device
0	Scanner
1…61	Devices
62	Computer interface to the network, such as a 1784-U2DN device
63	None Out of the box, a DeviceNet communication module is preset for address 63. Leaving address 63 open lets you get a new device on the network without conflicting with another device.

For more information about setting the node address of DeviceNet devices, refer to the Rockwell Automation Literature Library at the following address:

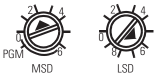
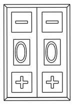
http://www.rockwellautomation.com/literature/

To find the publications specific to your module, search by the module's catalog number.

Set Node Address via Hardware Mechanism

Many DeviceNet devices have a hardware mechanism that you can use to set the node address. If a device has a hardware mechanism to set the node address, use that mechanism.

The following table describes the two most common hardware mechanisms.

Mechanism	Graphic	Description
Rotary switch	 2 4 0 PGM 6 MSD 2 4 0 8 6 LSD	You turn the dials of rotary switches to specific numbers that represent the device's node address. You usually need a small flathead screwdriver to turn the switches. A device reads the switches when you power it up. If you change the address, you should cycle power to that device for the change to take effect.
Push-wheel switch		You set the numbers on the push-wheel switch to specific numbers that represent the device's node address.

IMPORTANT As long as a device's hardware mechanism sets the node address to 63 or lower, you cannot change the node address with RSNetWorx for DeviceNet software or the DeviceNet node commissioning tool.

Make sure each device's node address set by a hardware mechanism matches the node address used in your RSNetWorx for DeviceNet software network configuration file, as described on page 40.

However, if you set a device's hardware mechanism to a number **higher than 63**, you can use RSNetWorx for DeviceNet software or the DeviceNet node commissioning tool to set the node address.

IMPORTANT You must cycle power to the module for node changes set through hardware to take effect.

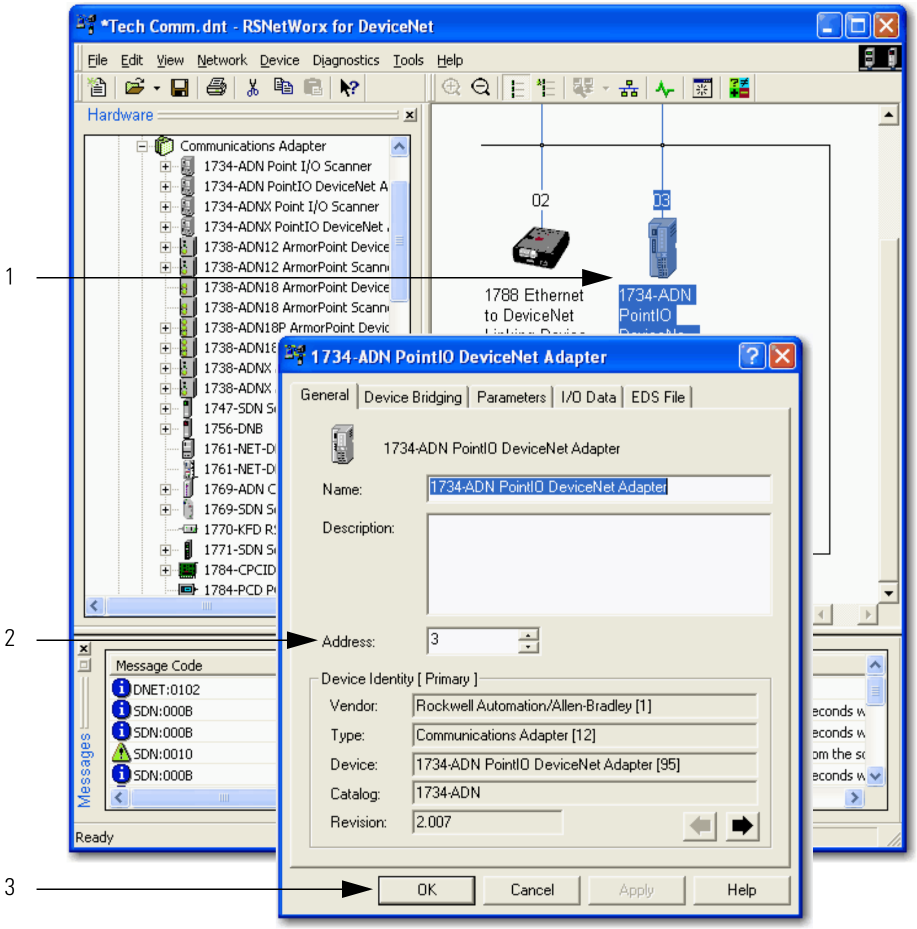
Set Node Address via Software

You can set a device's node address in RSNetWorx for DeviceNet software.

IMPORTANT	Setting a device's node address is only one task used when configuring a device with RSNetWorx for DeviceNet software. For complete information on how to configure all parameters with RSNetWorx for DeviceNet software, including setting a device's node address, refer to Chapter 4 on page 37.

Complete the following steps to set a device's node address in your network configuration file. This example uses an AC drive.

1. Double-click the device.
2. Enter the DeviceNet address for the device.
3. Click OK to close the configuration window.


1
2
3
*Tech Comm.dnt - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
Hardware
Communications Adapter
1734-ADN Point I/O Scanner
1734-ADN PointIO DeviceNet A
1734-ADNX Point I/O Scanner
1734-ADNX PointIO DeviceNet
1738-ADN12 ArmorPoint Device
1738-ADN12 ArmorPoint Scann
1738-ADN18 ArmorPoint Device
1738-ADN18 ArmorPoint Scann
1738-ADN18P ArmorPoint Devic
1738-ADN18
1738-ADNX
1738-ADNX
1747-SDN S
1756-DNB
1761-NET-D
1761-NET-D
1769-ADN C
1769-SDN S
1770-KFD R
1771-SDN S
1784-CPCID
1784-PCD P
02
1788 Ethernet to DeviceNet Linking Device
03
1734-ADN PointIO DeviceNe
Messages
Message Code
DNET:0102
SDN:000B
SDN:000B
SDN:0010
SDN:000B
econds w
econds w
om the sc
econds w
Ready
General | Device Bridging | Parameters | I/O Data | EDS File
OK Cancel Apply Help

1734-ADN PointIO DeviceNet Adapter

1734-ADN PointIO DeviceNet Adapter

Name: 1734-ADN PointIO DeviceNet Adapter

Description:

Address: 3

Device Identity [Primary]

Vendor: Rockwell Automation/Allen-Bradley [1]

Type: Communications Adapter [12]

Device: 1734-ADN PointIO DeviceNet Adapter [95]

Catalog: 1734-ADN

Revision: 2.007

IMPORTANT	The node address assigned by RSNetWorx for DeviceNet software only takes effect if the device **does not** have a hardware mechanism to set the node address or if the device has its hardware mechanism set to a number higher than 63.

Set Node Address via DeviceNet Node Commissioning Tool

You can use the DeviceNet node commissioning tool available in RSNetWorx for DeviceNet software to set the node addresses of devices on the DeviceNet network. Remember the following when you decide to use the DeviceNet node commissioning tool.

- You can only use this tool with a DeviceNet network that is online.
- You can only use this tool as you add new devices to the network that either do not have hardware mechanisms to set their node address or have their hardware mechanism set to a number higher than 63.

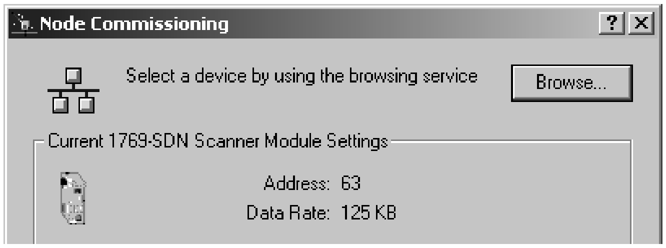
  If you add a device to the network with a hardware mechanism setting the node address to 63 or lower, this tool does not change the device's node address.
- You should complete the tasks described in this section each time a new device is added to the network.
- If you add more than one device to the online network before using the DeviceNet node commissioning tool, you will experience duplicate node address conflicts on the network because all new devices initially use node address 63.
- Keep track of the node addresses you set with the DeviceNet node commissioning tool and verify they match the device's configuration in the RSNetWorx for DeviceNet software configuration file.

Complete the following steps to use the DeviceNet node commissioning tool to set a device's node address. This example uses the 1769-SDN Compact I/O DeviceNet Scanner module.

1. Verify that the network is online.
2. Connect a device to the DeviceNet network.
3. Choose Start>Programs>Rockwell Software>RSNetWorx for DeviceNet>DeviceNet node commissioning tool.

   or

   From the Tools pull-down menu in RSNetWorx for DeviceNet software, choose Node Commissioning.
4. Click Browse.



Node Commissioning

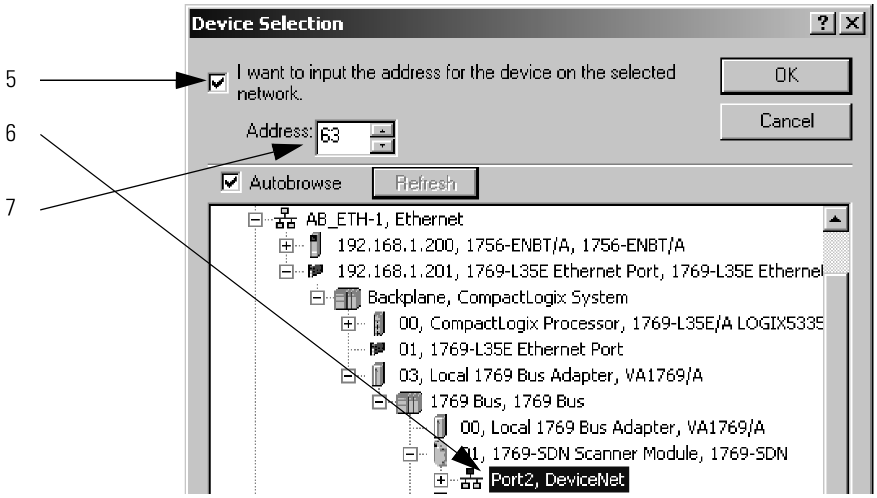
Select a device by using the browsing service

Current 1769-SDN Scanner Module Settings

Address: 63

Data Rate: 125 KB

5. On the Device Selection dialog box, check the 'I want to input the address for the device on the selected network' box.
6. Browse to the DeviceNet network.
7. Type the current address for the device. Out of the box, devices use address 63.
8. Click OK.



Device Selection

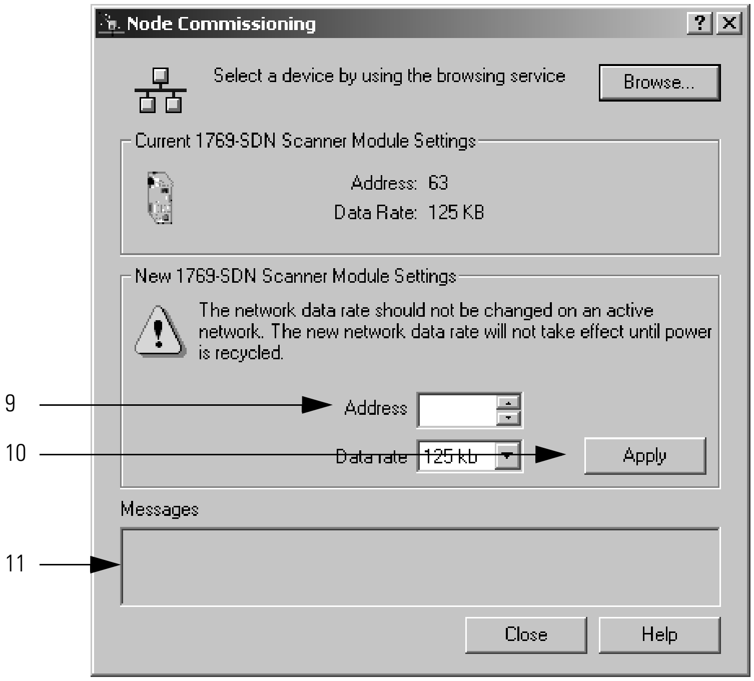
☑ I want to input the address for the device on the selected network.

Address: 63

☑ Autobrowse

```
AB_ETH-1, Ethernet
  192.168.1.200, 1756-ENBT/A, 1756-ENBT/A
  192.168.1.201, 1769-L35E Ethernet Port, 1769-L35E Ethernet
    Backplane, CompactLogix System
      00, CompactLogix Processor, 1769-L35E/A LOGIX5335
      01, 1769-L35E Ethernet Port
      03, Local 1769 Bus Adapter, VA1769/A
        1769 Bus, 1769 Bus
          00, Local 1769 Bus Adapter, VA1769/A
          01, 1769-SDN Scanner Module, 1769-SDN
            Port2, DeviceNet
```

9. When you return to the Node Commissioning dialog box, enter the new address for the device.
10. Click Apply.
11. Look for confirmation.



Node Commissioning

Select a device by using the browsing service

Current 1769-SDN Scanner Module Settings

Address: 63

Data Rate: 125 KB

New 1769-SDN Scanner Module Settings

The network data rate should not be changed on an active network. The new network data rate will not take effect until power is recycled.

Address

Data rate 125 kb

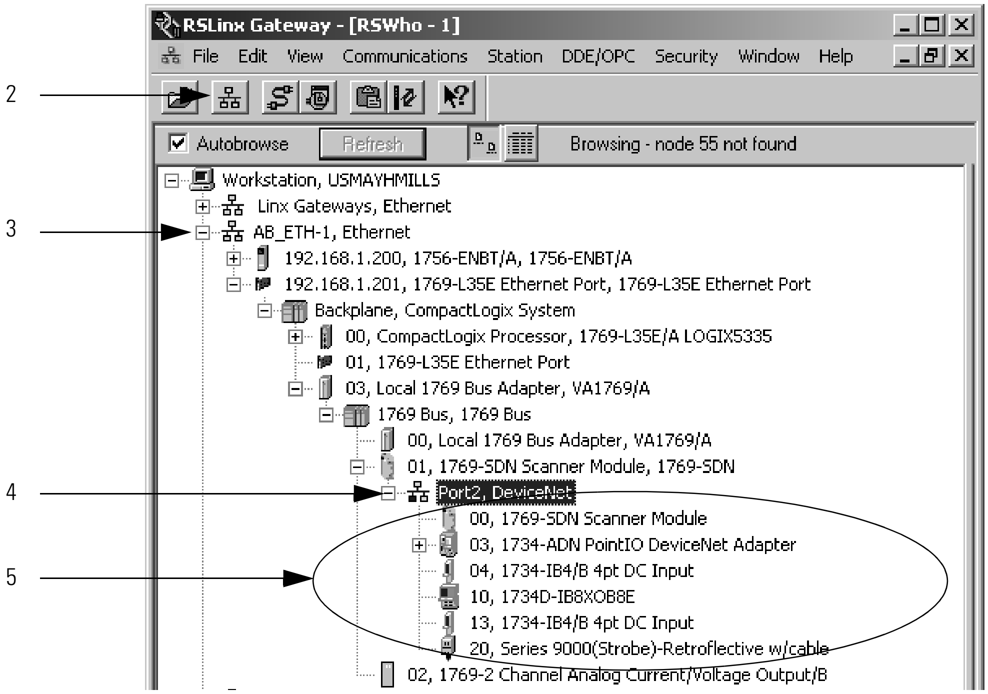
Messages

12. If you need to use the DeviceNet node commissioning tool to set another device's node address, return to step 2 on page 34 and repeat the process.

Make Sure Your Devices Are on Your Network

Once you have assigned a node address to each device, make sure that the devices are communicating on the network. Complete these steps to make sure your devices are on the network.

1. Start RSLinx communication software.
2. Go online.
3. Expand a driver that lets you access the DeviceNet network.
4. Browse to the DeviceNet network.
5. Make sure you see all the devices that are connected to the DeviceNet network.


RSLinx Gateway - [RSWho - 1]
File Edit View Communications Station DDE/OPC Security Window Help
Autobrowse
Refresh
2
3
4
5

```
Browsing - node 55 not found

Workstation, USMAYHMILLS
  Linx Gateways, Ethernet
  AB_ETH-1, Ethernet
    192.168.1.200, 1756-ENBT/A, 1756-ENBT/A
    192.168.1.201, 1769-L35E Ethernet Port, 1769-L35E Ethernet Port
      Backplane, CompactLogix System
        00, CompactLogix Processor, 1769-L35E/A LOGIX5335
        01, 1769-L35E Ethernet Port
        03, Local 1769 Bus Adapter, VA1769/A
          1769 Bus, 1769 Bus
            00, Local 1769 Bus Adapter, VA1769/A
            01, 1769-SDN Scanner Module, 1769-SDN
              Port2, DeviceNet
                00, 1769-SDN Scanner Module
                03, 1734-ADN PointIO DeviceNet Adapter
                04, 1734-IB4/B 4pt DC Input
                10, 1734D-IB8XOB8E
                13, 1734-IB4/B 4pt DC Input
                20, Series 9000(Strobe)-Retroflective w/cable
            02, 1769-2 Channel Analog Current/Voltage Output/B
```

Chapter 4

Configure the Network Offline

This chapter describes how to configure the network offline with RSNetWorx for DeviceNet software.

Before You Begin

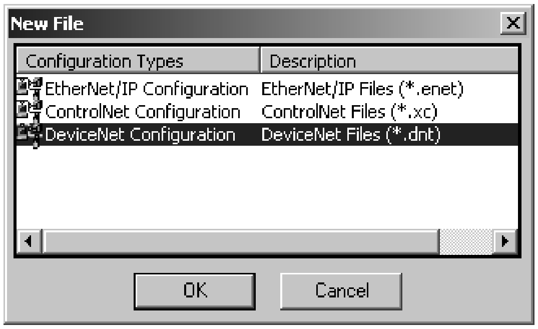
Before you configure the DeviceNet network, make sure you have a list of the devices that you put on the DeviceNet network and, at minimum, the address for each. The following table shows an example list of devices.

Device	Address	Input Size of Device (Bytes)	Input Memory in Scanner (DINTs)	Output Size of Device (Bytes)	Output Memory in Scanner (DINTs)
scanner	0	n/a	n/a	n/a	n/a
PanelView terminal	3	128	32	128	32
<empty>			2		2
I/O adapter w/ modules	5	9	3	5	2
<empty>			2		2
drive	7	4	1	4	1
<empty>			2		2
photoeye	9	1	1	0	0
computer interface	62	n/a	n/a	n/a	n/a
	63				
	Total		43		41

Create a File for the Network

Complete the following steps to create a DeviceNet configuration file.

1. Start RSNetWorx for DeviceNet software.
2. Create a file.
3. Select DeviceNet Configuration.



Configuration Types	Description
EtherNet/IP Configuration	EtherNet/IP Files (*.enet)
ControlNet Configuration	ControlNet Files (*.xc)
DeviceNet Configuration	DeviceNet Files (*.dnt)

4. Click OK.
5. Save the file.

   Make sure you give the file a name that identifies this specific DeviceNet network.

Create Your Network in RSNetWorx for DeviceNet Software

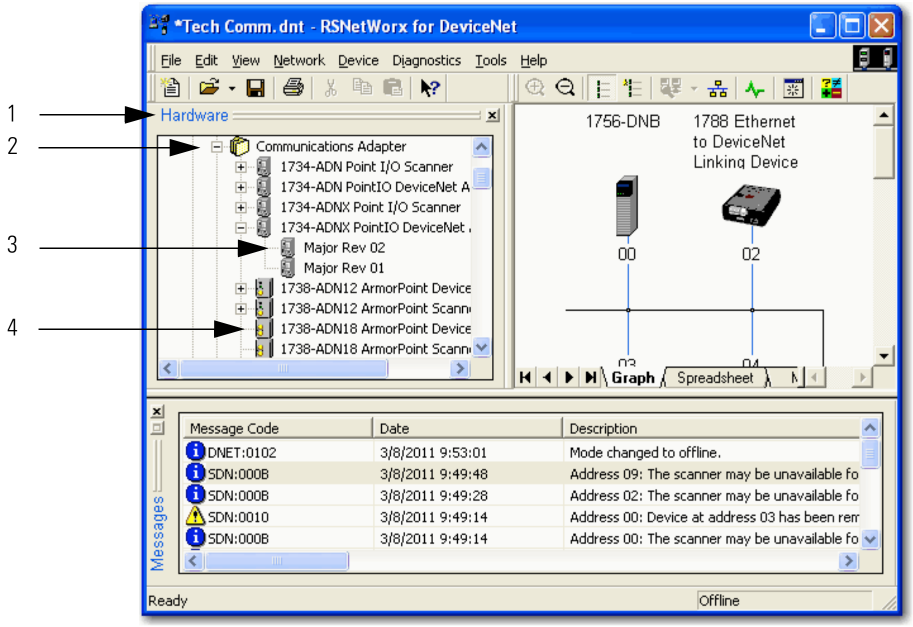
Before you configure a DeviceNet communication module in RSNetWorx for DeviceNet software, you must add it to the network configuration file.

The finished picture **should match** the collection of devices that are or will be physically connected to the DeviceNet network. If the network configuration file you create offline does not match the physical collection of devices on the network, you may experience issues when you go online with your project.

Complete the following steps to add each device to network configuration file.

1. Browse the hardware list for the device.
2. If there is a [+] sign next to the device, click the sign to expand the choices in that section.
3. Double-click the major revision of the device.

   We recommend that the major revision of all devices added to the offline network match the devices that will be connected to the online network.
4. For a device without a list of major revisions, that is, no [+] or [-] sign, double-click the device.


1
2
3
4
*Tech Comm.dnt - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
Hardware
Communications Adapter
1734-ADN Point I/O Scanner
1734-ADN PointIO DeviceNet A
1734-ADNX Point I/O Scanner
1734-ADNX PointIO DeviceNet
Major Rev 02
Major Rev 01
1738-ADN12 ArmorPoint Device
1738-ADN12 ArmorPoint Scann
1738-ADN18 ArmorPoint Device
1738-ADN18 ArmorPoint Scann
1756-DNB
1788 Ethernet to DeviceNet Linking Device
00
02
03
04
Graph
Spreadsheet
Messages
Ready
Offline

Message Code	Date	Description
DNET:0102	3/8/2011 9:53:01	Mode changed to offline.
SDN:000B	3/8/2011 9:49:48	Address 09: The scanner may be unavailable fo
SDN:000B	3/8/2011 9:49:28	Address 02: The scanner may be unavailable fo
SDN:0010	3/8/2011 9:49:14	Address 00: Device at address 03 has been rem
SDN:000B	3/8/2011 9:49:14	Address 00: The scanner may be unavailable fo

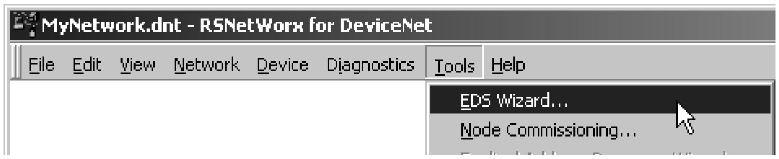
If the hardware list **does not** show a device, then RSNetWorx for DeviceNet software requires the EDS file for the device.

To add an EDS file, follow these steps.

1. To see if an EDS file is available, go to the following site:

   http://www.rockwellautomation.com/resources/eds/

2. Use the EDS wizard of RSNetWorx for DeviceNet software to register the file and see it.


MyNetwork.dnt - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
EDS Wizard...
Node Commissioning...

Configure Each Device

After adding devices to the network configuration file, as described in Create Your Network in RSNetWorx for DeviceNet Software on page 39, you configure parameters for each device to define the modules' behavior.

IMPORTANT	You can configure most devices as you add them to the network configuration file or you can add all the devices and then configure them. Typically, you add a network scanner to the network first. In this case, we recommend that you add all devices to the network configuration file before configuring the scanner. Multiple parameters that need to be configured in the scanner's configuration, for example, building a scan list, require you to choose from devices on the network. Refer to Configure the Scanner on page 44 for more information.

Complete these tasks when configuring DeviceNet communication modules:

- Specify a Device Node Address
- Configure Device Parameters

Specify a Device Node Address

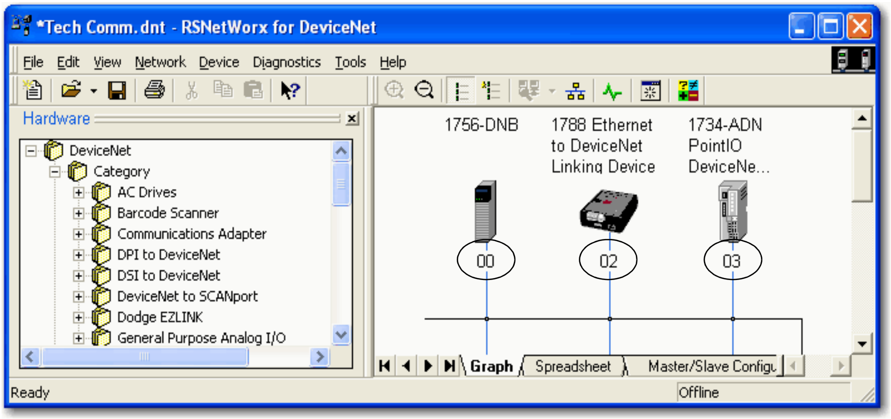
The following options are available to set a device's node address:

- Hardware mechanism, as described on page 30
- RSNetWorx for DeviceNet software, as described in this chapter
- DeviceNet node commissioning tool, as described on page 33

All DeviceNet devices ship with their node addresses set to 63. To avoid duplicate node number conditions on the network, you should change the node address for each device to unique numbers.

Give this address	To this device
0	Scanner
1…61	Your devices
62	Computer interface to the network, such as a 1784-U2DN device
63	None Out of the box, a DeviceNet communication module is preset for address 63. Leaving address 63 open lets you get a new device on the network without conflicting with another device.

When you create your network in RSNetWorx for DeviceNet software, as described on page 39, devices are automatically assigned node addresses based on the order in which they were added to the network. The number appears below the device's graphic on the screen as shown below.



As you create the network, consider the following:

- If you used a hardware mechanism to assign a node address for a device, that number takes precedence over any number you assign in RSNetWorx for DeviceNet software.

  Make sure the numbers assigned by the hardware mechanism and in your configuration file are the same for each device. For example, if the node address for a 1756-DNB ControlLogix DeviceNet Scanner is set to 2 via a hardware mechanism, but in the RSNetWorx for DeviceNet software configuration file, the node address is 0, you need to change the address in the software to 2.

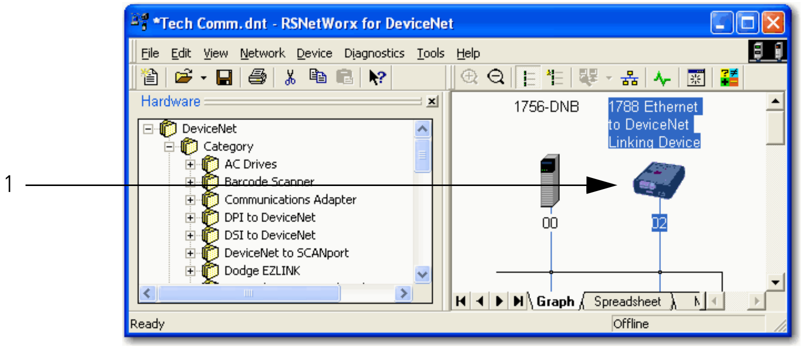
- The node addresses that are automatically assigned as you add devices to the configuration file do not take effect when the project is offline.

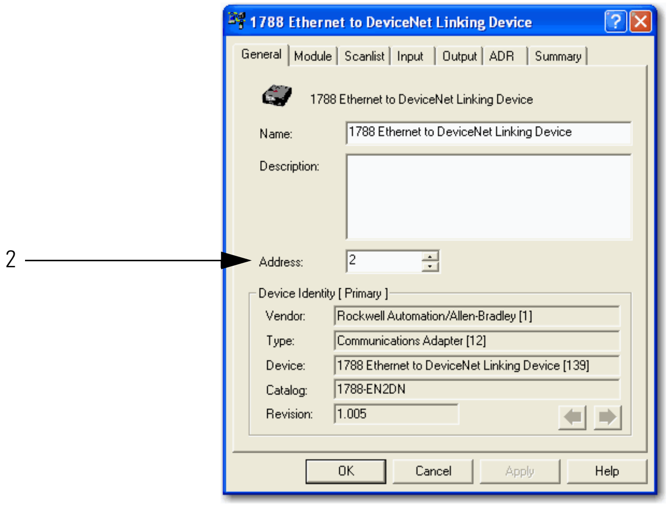
- For devices that do not have hardware mechanisms, the node number assigned in the network configuration file takes effect when you download the project to the DeviceNet network, as described on page 55.

Change a Device Node Address

You may need to assign a device's node address that is different from the number automatically assigned when the device is added to the configuration file. Complete the following steps to assign a device a specific node address.

1. Double-click the device.
2. Enter the node address for the device.





1788 Ethernet to DeviceNet Linking Device

Name: 1788 Ethernet to DeviceNet Linking Device

Description:

Address: 2

Device Identity [Primary]

Vendor: Rockwell Automation/Allen-Bradley [1]

Type: Communications Adapter [12]

Device: 1788 Ethernet to DeviceNet Linking Device [139]

Catalog: 1788-EN2DN

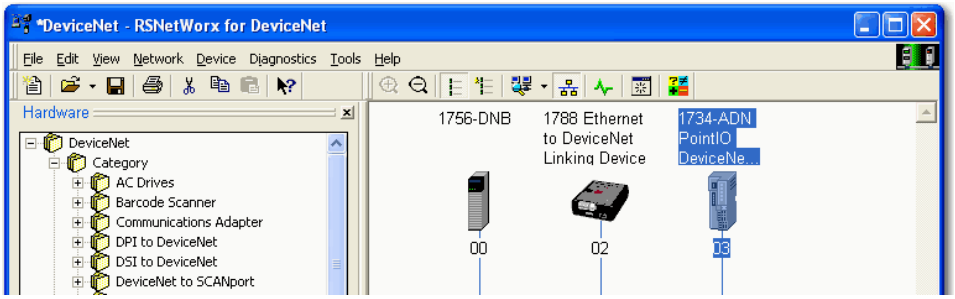
Revision: 1.005

3. Click OK.

Configure Device Parameters

Complete the following steps to configure device parameters.

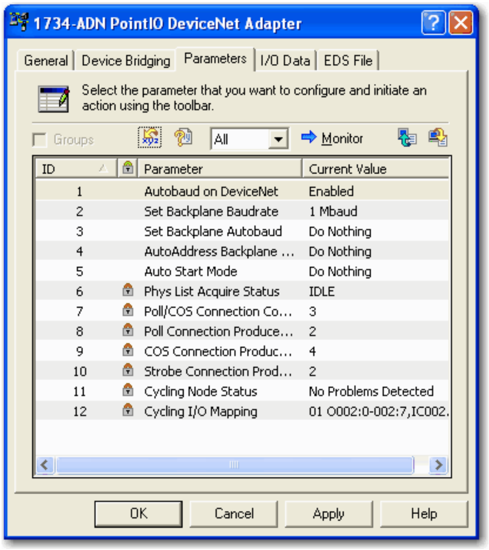
1. Double-click the device to display the configuration dialog box.


*DeviceNet - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
Hardware
DeviceNet
Category
AC Drives
Barcode Scanner
Communications Adapter
DPI to DeviceNet
DSI to DeviceNet
DeviceNet to SCANport
1756-DNB
1788 Ethernet to DeviceNet Linking Device
1734-ADN PointIO DeviceNe...
00
02
03

2. Click the appropriate tab.
3. Set a parameter to the desired new value.

   Typically, there are two methods to set a parameter:

   - Choose a parameter from a pull-down menu
   - Type a new value



Select the parameter that you want to configure and initiate an action using the toolbar.

ID		Parameter	Current Value
1		Autobaud on DeviceNet	Enabled
2		Set Backplane Baudrate	1 Mbaud
3		Set Backplane Autobaud	Do Nothing
4		AutoAddress Backplane ...	Do Nothing
5		Auto Start Mode	Do Nothing
6	🔒	Phys List Acquire Status	IDLE
7	🔒	Poll/COS Connection Co...	3
8	🔒	Poll Connection Produce...	2
9	🔒	COS Connection Produc...	4
10	🔒	Strobe Connection Prod...	2
11	🔒	Cycling Node Status	No Problems Detected
12	🔒	Cycling I/O Mapping	01 O002:0-002:7,IC002.

4. Click Apply to apply the change and leave the configuration dialog box open.

   or

   Click OK to apply the change and close the configuration dialog box.

Configure the Scanner

A DeviceNet scanner manages input and output data for a controller. The scanner receives input data from I/O devices, organizes the information into scanner data tables, and sends the input data to the controller when the controller requests it. In addition, when the scanner receives output data from the controller, it sends the data to the I/O devices.

A DeviceNet scanner is the only device that can be used as a master on a DeviceNet network. When there is only one scanner on a network, it is the master for that network by default. When there are multiple scanners on the same network, each device can have only one scanner designated as its master, which is the scanner that controls its outputs.

You must configure the scanner to define how it communicates with other devices on the DeviceNet network. You must complete the following tasks to configure the scanner to communicate with the devices on the network:

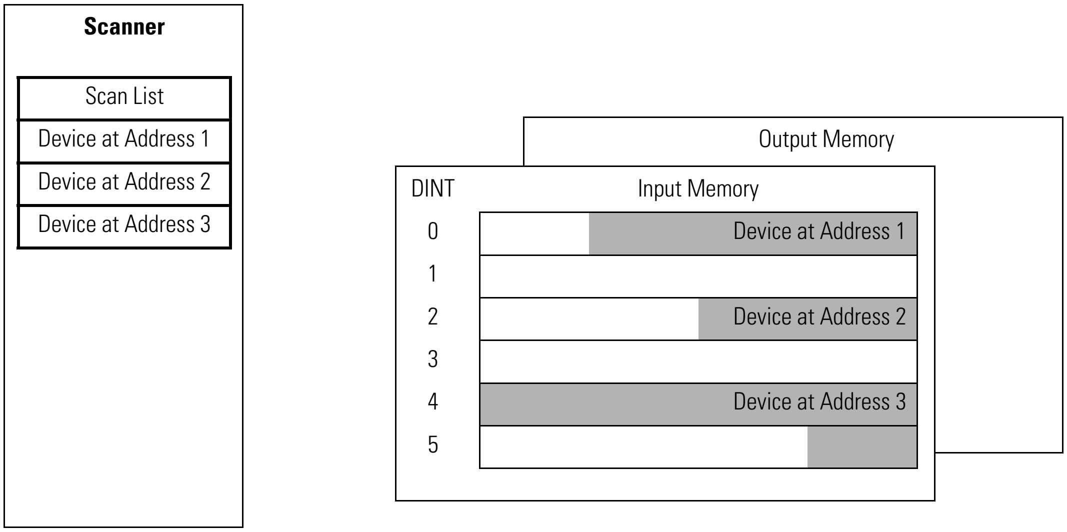
- Build the Scan List
- Set the Alignment Option
- Manually Assign Each Device to a Memory Location

IMPORTANT Make sure you configure the scanner **after** you add all devices to the network. When you add a device to the network, it automatically appears in the list of available devices for the scanner's scan list.

If you build a scan list and then add another device to the network, it will not be included in the scan list until you repeat the steps in the next section.

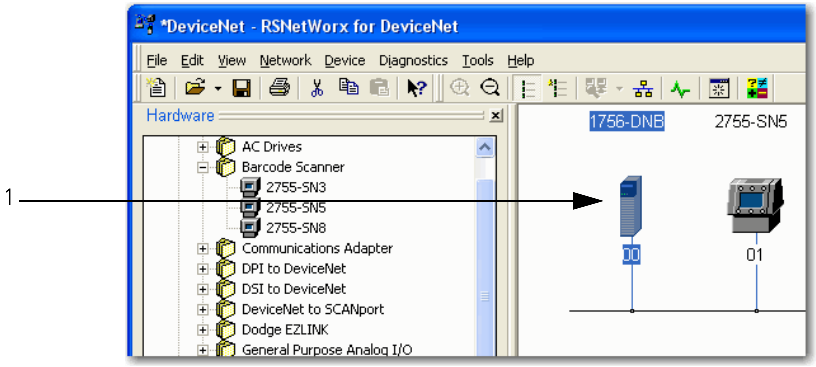
Build the Scan List

A scan list is a list of devices with which the scanner communicates. For each device in the scanner's scan list, the scanner sets aside input or output memory for the data of the device.



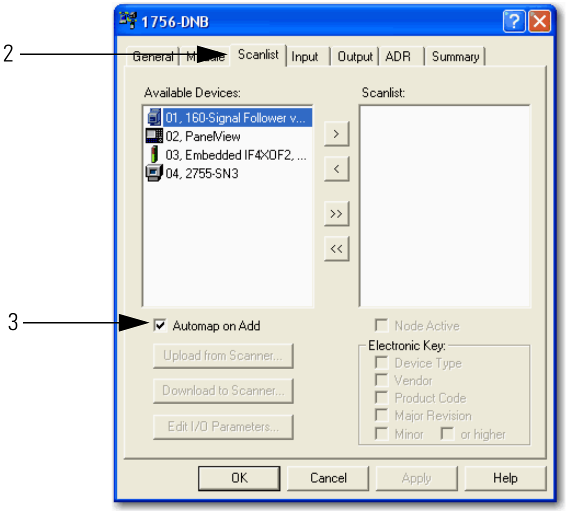
Complete the following steps to build a scan list.

1. Double-click the scanner to display configuration dialog box.



2. Click the Scanlist tab.

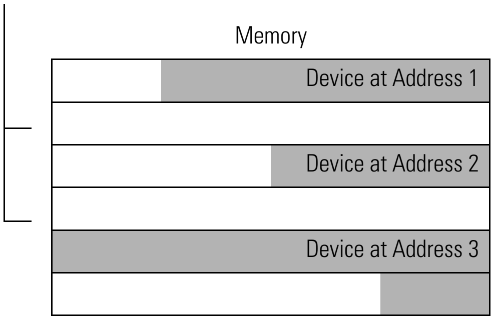
3. Clear or check Automap on Add.



RSNetWorx for DeviceNet software can automatically assign the memory location for each device.

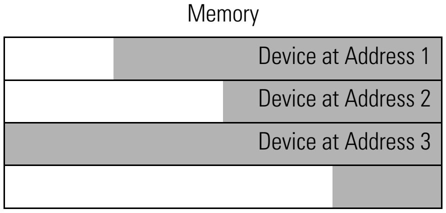
a. If you want to leave gaps between devices in the memory as shown below, clear the box.

Leave Gaps Between Devices

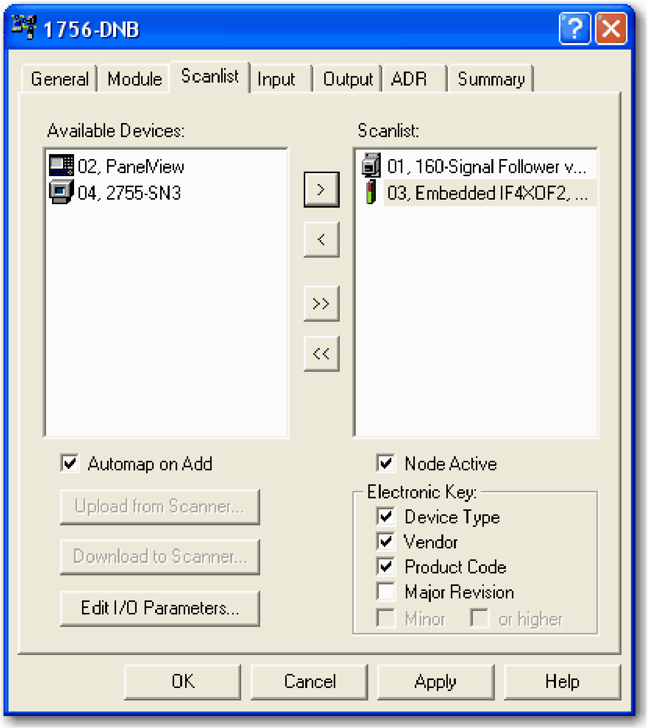


b. If you want to place devices in sequential DINT's as shown below, leave the box checked. When you check the box, the software automatically assigns a memory location for each device as you add it to the scan list.

Place Devices in Sequential DINTs



4. Move devices from the Available Devices column to the Scanlist column.



Available Devices:

- 02, PanelView
- 04, 2755-SN3

Scanlist:

- 01, 160-Signal Follower v...
- 03, Embedded IF4XOF2, ...

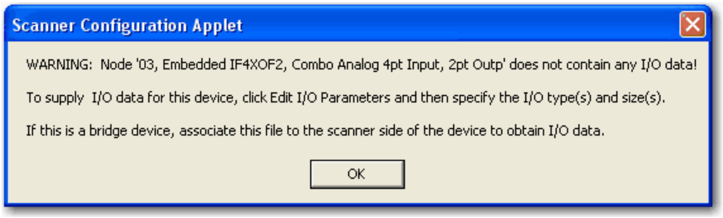
☑ Automap on Add

☑ Node Active

Electronic Key:

☑ Device Type
☑ Vendor
☑ Product Code
☐ Major Revision
☐ Minor ☐ or higher

5. If you get the following warning for a device, see Set the I/O Parameters of a Device on page 167.



WARNING: Node '03, Embedded IF4XOF2, Combo Analog 4pt Input, 2pt Outp' does not contain any I/O data!

To supply I/O data for this device, click Edit I/O Parameters and then specify the I/O type(s) and size(s).

If this is a bridge device, associate this file to the scanner side of the device to obtain I/O data.

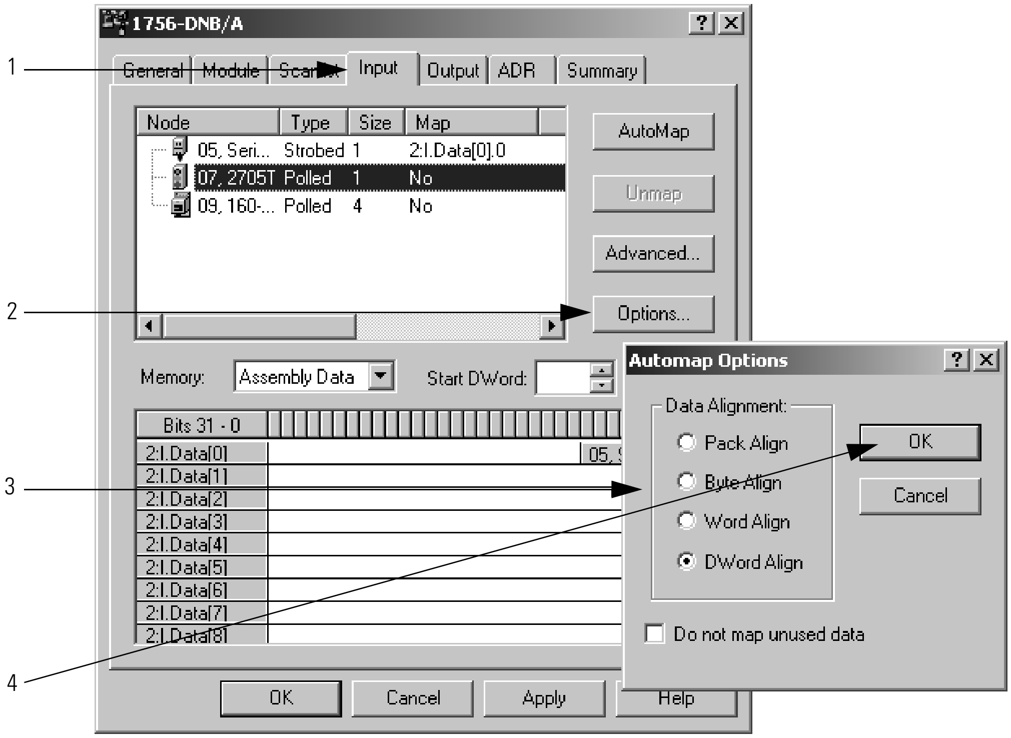
Set the Alignment Option

Choose a data alignment option to map the I/O data so that it is aligned on a boundary, such as a byte, word, or double-word, or efficiently grouped without alignment in the input or output memory map. To map I/O data so it is grouped without alignment, click the Pack Align option.

IMPORTANT	The alignment option you choose applies to both the input and output maps.

Complete the following steps to select an alignment option.

1. Click the Input tab.
2. Click Options.
3. Click the desired data alignment.
4. Click OK to close the Automap Options dialog box.



Node	Type	Size	Map
05, Seri...	Strobed	1	2:I.Data[0].0
07, 2705T	Polled	1	No
09, 160-...	Polled	4	No

Memory: Assembly Data

Start DWord:

Bits 31 - 0	
2:I.Data[0]	05, !
2:I.Data[1]	
2:I.Data[2]	
2:I.Data[3]	
2:I.Data[4]	
2:I.Data[5]	
2:I.Data[6]	
2:I.Data[7]	
2:I.Data[8]	

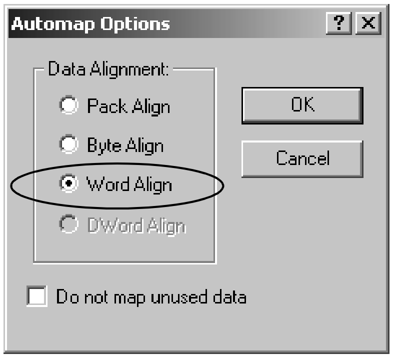
Data Alignment:

○ Pack Align
○ Byte Align
○ Word Align
● DWord Align

□ Do not map unused data

SoftLogix 5800 Controller

The SoftLogix 5800 scanner 1784-PCIDS organizes its input and output memory in 16-bit words. For that scanner, click the Word Align option.



Data Alignment:
○ Pack Align
○ Byte Align
● Word Align
○ DWord Align

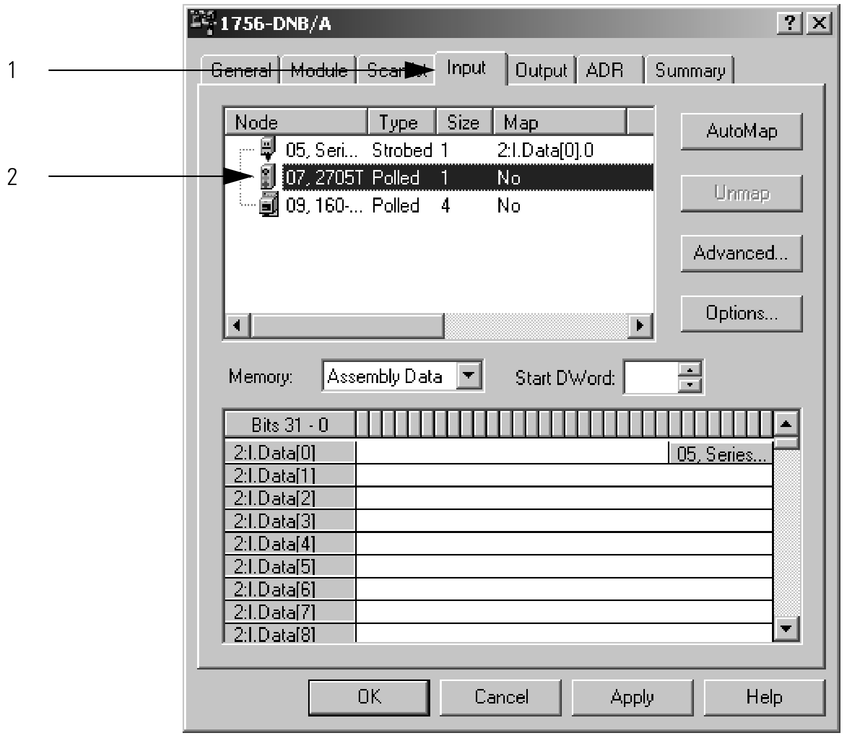
□ Do not map unused data

Manually Assign Each Device to a Memory Location

You can manually assign locations for device data.

IMPORTANT	If you configured the software to automatically assign memory locations as devices are added, as described on page 45, skip this section.

1. Click the Input tab.
2. Select the device.



Node	Type	Size	Map
05, Seri...	Strobed	1	2:I.Data[0].0
07, 2705T	Polled	1	No
09, 160-...	Polled	4	No

Memory: Assembly Data

Start DWord:

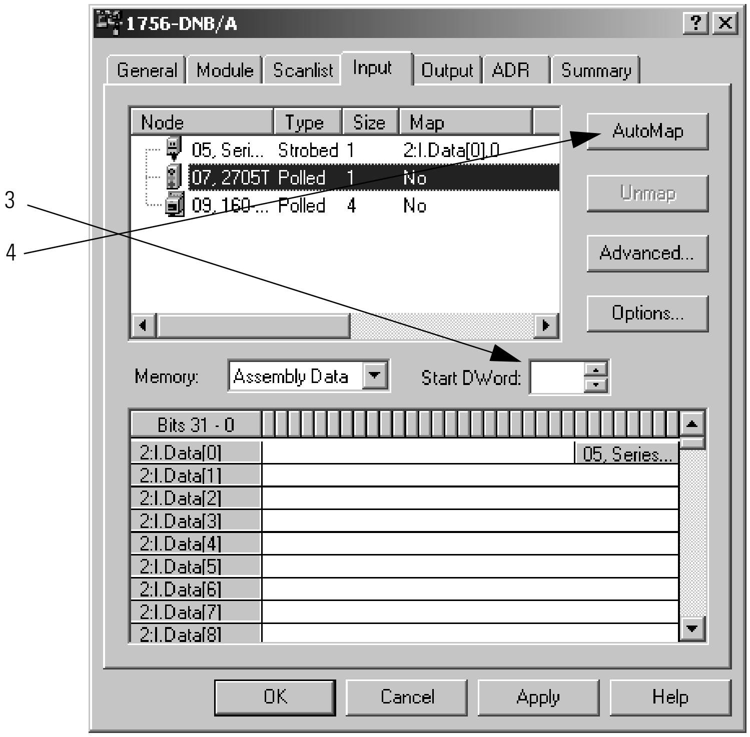
Bits 31 - 0	
2:I.Data[0]	05, Series...
2:I.Data[1]	
2:I.Data[2]	
2:I.Data[3]	
2:I.Data[4]	
2:I.Data[5]	
2:I.Data[6]	
2:I.Data[7]	
2:I.Data[8]	

3. In the Start DWord field, enter the element number to which you want to assign the data.

   This is the starting point for the data. Larger data sizes wrap to several elements. For example, to start the data in . . . Data[3], enter 3 in the Start DWord box.

4. Click Automap.

   An entry for the device appears in the input array.



Node	Type	Size	Map
05, Seri...	Strobed	1	2:I.Data[0].0
07, 2705T	Polled	1	No
09, 160-...	Polled	4	No

Memory: Assembly Data

Start DWord:

Bits 31 - 0	
2:I.Data[0]	05, Series...
2:I.Data[1]	
2:I.Data[2]	
2:I.Data[3]	
2:I.Data[4]	
2:I.Data[5]	
2:I.Data[6]	
2:I.Data[7]	
2:I.Data[8]	

5. Click the Output tab and repeat step 2 through step 4.

6. Click OK to complete the scanner configuration.

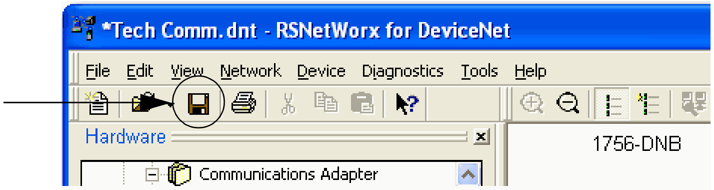
Sometimes, a specific input or output value may end up as the upper bytes of a DINT in the scanner.

Instance 70 Data Format (Basic Speed Control Input Assembly)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0						Running1		Faulted
1								
2	Speed Actual RPM (Low Byte)							
3	Speed Actual RPM (High Byte)							

To make your programming easier, use advanced mapping to re-map the value to its own memory location. For more information, see Map the Memory Location with Advanced Mapping on page 183.

Save the Configuration File

After you configure each device on the network, including the scanner, save the file.


*Tech Comm.dnt - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
Hardware
Communications Adapter
1756-DNB

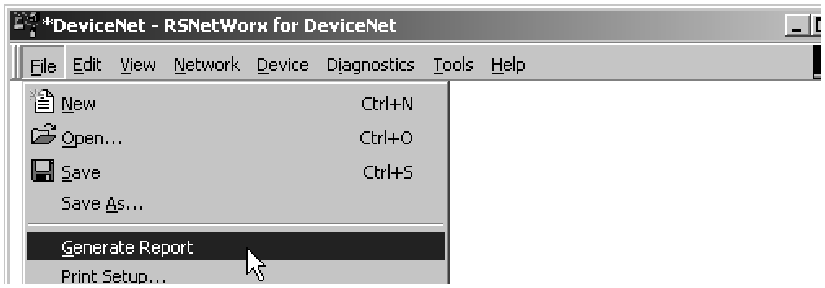
Generate an RSNetWorx for DeviceNet Report

An RSNetWorx for DeviceNet report shows these items:

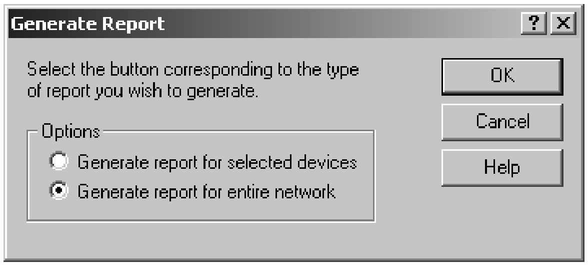
- Devices on the network
- Memory addresses of those devices in the scanner
- Device configurations

The report is a useful reference when you program your system. Complete the following steps to generate a report.

1. From the File menu in RSNetWorx for DeviceNet software, choose Generate Report.


*DeviceNet - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
New Ctrl+N
Open... Ctrl+O
Save Ctrl+S
Save As...
Generate Report
Print Setup...

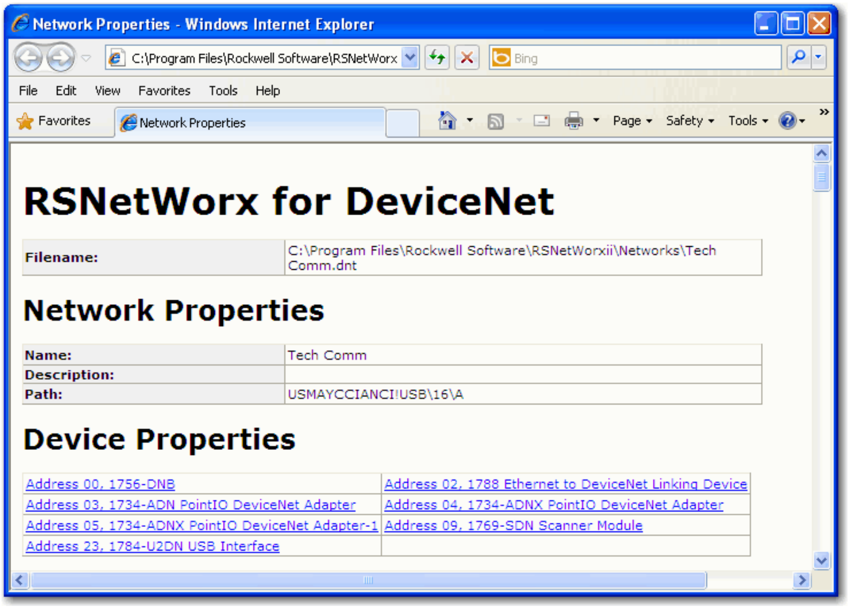
2. Click Generate report for entire network.



Select the button corresponding to the type of report you wish to generate.

Options
○ Generate report for selected devices
● Generate report for entire network

The report appears in your web browser.


Network Properties - Windows Internet Explorer
C:\Program Files\Rockwell Software\RSNetWorx
Bing
File Edit View Favorites Tools Help
Favorites
Network Properties
Page Safety Tools

RSNetWorx for DeviceNet

Filename:	C:\Program Files\Rockwell Software\RSNetWorxii\Networks\Tech Comm.dnt

Network Properties

Name:	Tech Comm
Description:	
Path:	USMAYCCIANCI!USB\16\A

Device Properties

Address 00, 1756-DNB	Address 02, 1788 Ethernet to DeviceNet Linking Device
Address 03, 1734-ADN PointIO DeviceNet Adapter	Address 04, 1734-ADNX PointIO DeviceNet Adapter
Address 05, 1734-ADNX PointIO DeviceNet Adapter-1	Address 09, 1769-SDN Scanner Module
Address 23, 1784-U2DN USB Interface	

Go Online to Your Network

When you are finished configuring the network configuration file offline, you may be ready to go online and download the network configuration file to the network.

IMPORTANT Before you can go online and download the network configuration file to the DeviceNet network, you must have completed the following tasks:

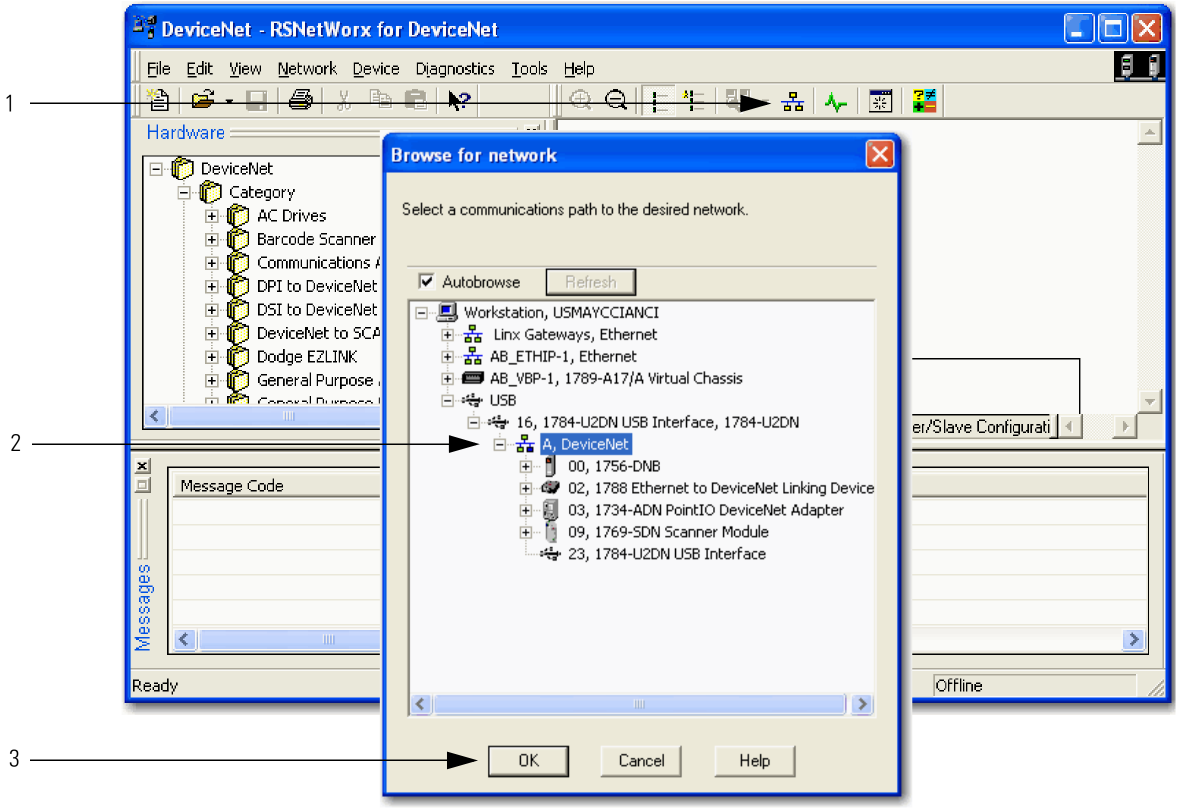
- Connect a computer to the network, as described in Chapter 2
- Connect all required devices to the network, as described in Chapter 3

When you go online to a DeviceNet network, RSNetWorx for DeviceNet software browses the network once and shows the devices on the network. Remember the following when you go online:

- RSNetWorx for DeviceNet software does not read (upload) or change (download) the parameters of any of the devices.
- The picture you see remains static. It does not show any changes since the last browse.
- The network configuration file configured offline must match the picture created when RSNetWorx for DeviceNet software browses the network.

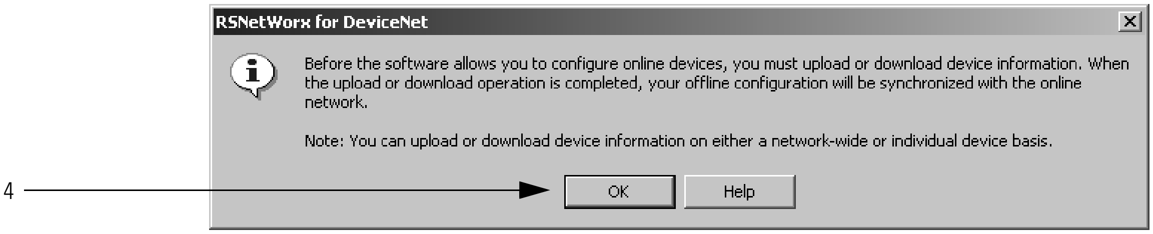
Complete the following steps to go online.

1. Click the Online button.
2. Select the DeviceNet network.
3. Click OK.


1
2
3
DeviceNet - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
Hardware
DeviceNet
Category
AC Drives
Barcode Scanner
Communications A
DPI to DeviceNet
DSI to DeviceNet
DeviceNet to SCA
Dodge EZLINK
General Purpose
Messages
Message Code
er/Slave Configurati
Ready
Offline
Browse for network
☑ Autobrowse
Refresh
Workstation, USMAYCCIANCI
Linx Gateways, Ethernet
AB_ETHIP-1, Ethernet
AB_VBP-1, 1789-A17/A Virtual Chassis
USB
16, 1784-U2DN USB Interface, 1784-U2DN
A, DeviceNet
00, 1756-DNB
02, 1788 Ethernet to DeviceNet Linking Device
03, 1734-ADN PointIO DeviceNet Adapter
09, 1769-SDN Scanner Module
23, 1784-U2DN USB Interface
OK
Cancel
Help

Select a communications path to the desired network.

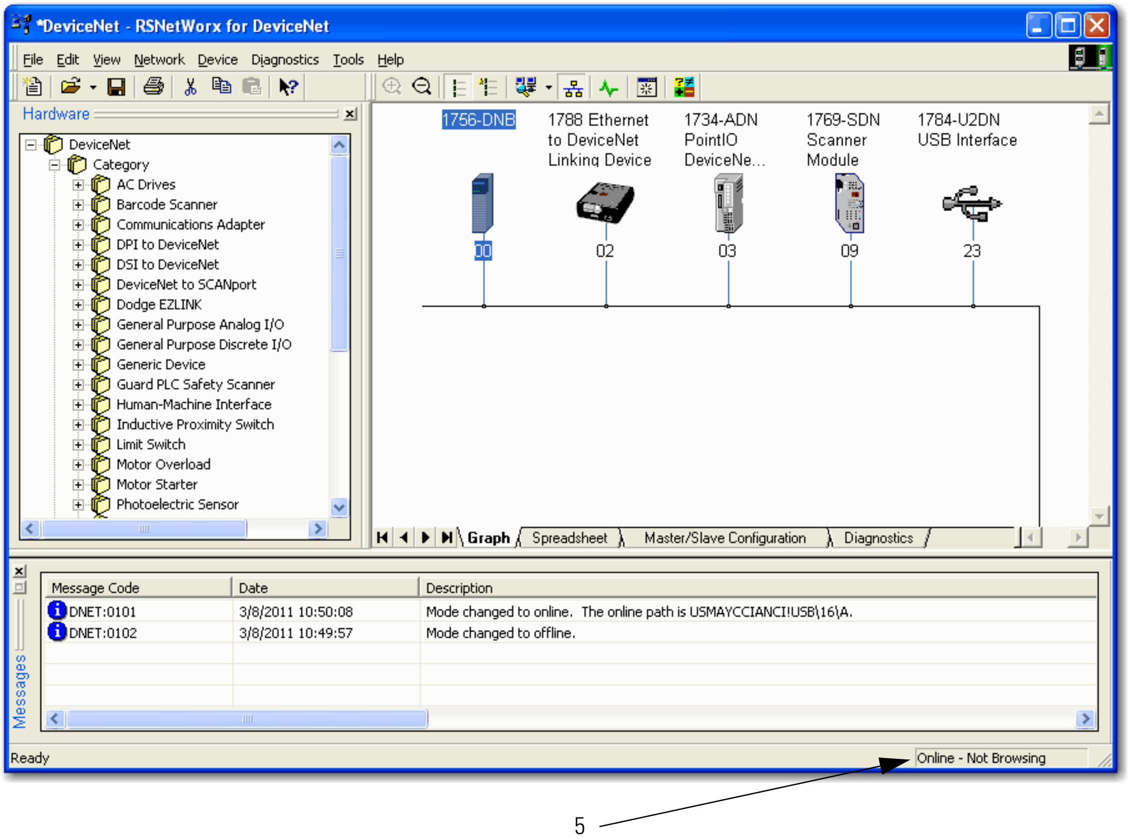
4. When the pop-up message appears, click OK.



Before the software allows you to configure online devices, you must upload or download device information. When the upload or download operation is completed, your offline configuration will be synchronized with the online network.

Note: You can upload or download device information on either a network-wide or individual device basis.

5. Verify that you are online.



Message Code	Date	Description
DNET:0101	3/8/2011 10:50:08	Mode changed to online. The online path is USMAYCCIANCI!USB\16\A.
DNET:0102	3/8/2011 10:49:57	Mode changed to offline.

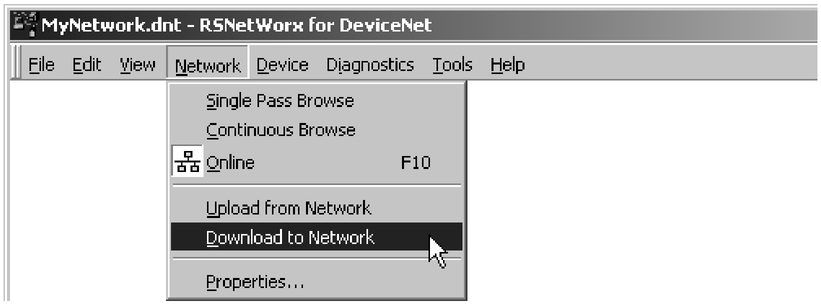
Download Configuration to Your Network

After you go online with the network configuration file you created while offline, you can download the configuration to the network.

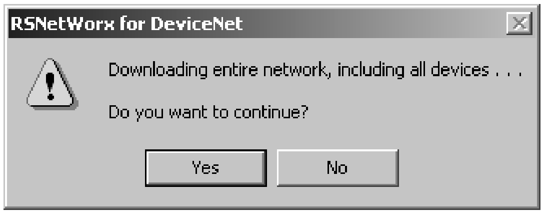
IMPORTANT	Before you download configuration to the network, make sure the scanner is in Idle mode. To put the scanner in Idle mode, do one of the following: • Place the controller in program/remote program mode. • Turn off the …O.CommandRegister.Run bit of the scanner.

Complete the following steps to download configuration to the DeviceNet network.

1. From the File menu in Network>Download to Network.



2. When prompted, click Yes to download the entire network.


RSNetWorx for DeviceNet

Downloading entire network, including all devices . . .

Do you want to continue?

Notes:

Chapter 5

Configure the Network Online

This chapter explains how to configure the network online with RSNetWorx for DeviceNet software.

Configuring the network online reduces the number of configuration tasks you must complete compared to configuring the network offline. Configuring the network online has these advantages:

- Devices on the network automatically appear in your network configuration file as soon as you go online. You do not need to add the devices to the network configuration file.

- The network configuration file automatically matches the physical setup of devices on the network as well as the major and minor revisions of the online devices.

- The configuration is guaranteed to match the major and minor revisions of the online devices.

- You can easily upload device configurations to your network configuration file, make changes to the configuration parameters, and download them to the device.

Before You Begin

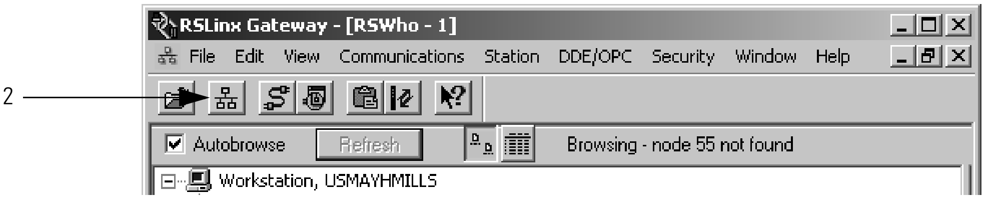
Before you configure the network, make sure you have a list of the devices that are on the network and, at minimum, the node address for each of them. The following table shows an example list of devices.

Device	Address	Input Size of Device (Bytes)	Input Memory in Scanner (DINTs)	Output Size of Device (Bytes)	Output Memory in Scanner (DINTs)
scanner	0	n/a	n/a	n/a	n/a
PanelView terminal	3	128	32	128	32
<empty>			2		2
I/O adapter w/ modules	5	9	3	5	2
<empty>			2		2
drive	7	4	1	4	1
<empty>			2		2
photoeye	9	1	1	0	0
computer interface	62	n/a	n/a	n/a	n/a
	63				
	Total		43		41

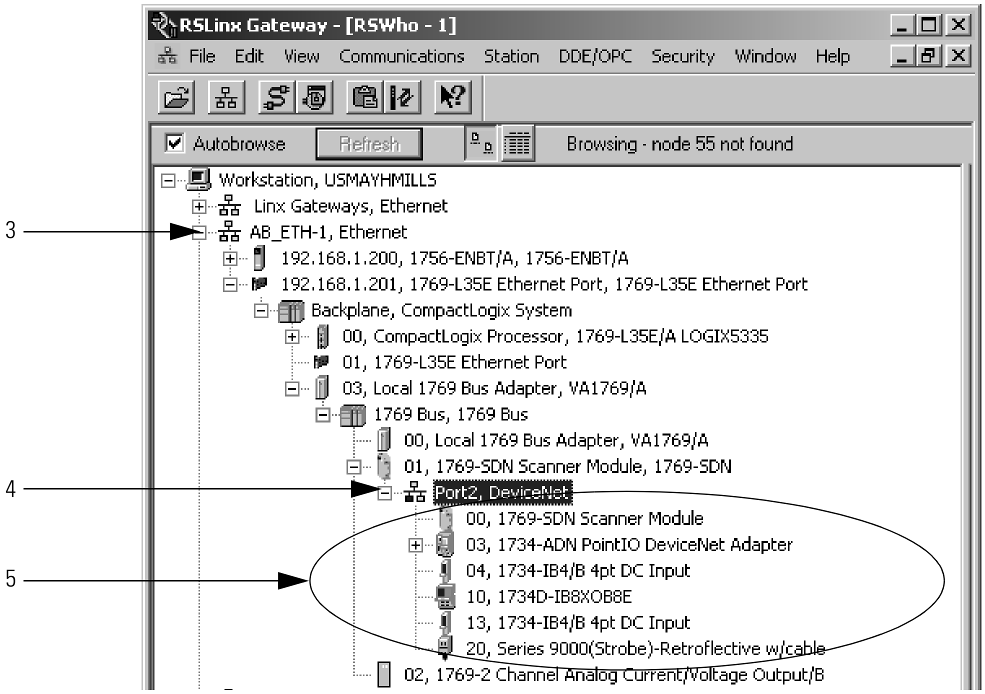
Verify Communication Between the Computer and Devices

To configure your network online, your computer must be able to communicate with each device on the DeviceNet network. Use RSLinx communication software to verify that you can communicate with all the devices.

1. Start RSLinx communication software.
2. Click the Online button.



3. Expand a driver that lets you access the DeviceNet network.
4. Select the DeviceNet network.
5. Make sure you see all the devices that are connected to the DeviceNet network.


RSLinx Gateway - [RSWho - 1]
File Edit View Communications Station DDE/OPC Security Window Help
☑ Autobrowse
Refresh
3
4
5

Browsing - node 55 not found

```
Workstation, USMAYHMILLS
  Linx Gateways, Ethernet
  AB_ETH-1, Ethernet
    192.168.1.200, 1756-ENBT/A, 1756-ENBT/A
    192.168.1.201, 1769-L35E Ethernet Port, 1769-L35E Ethernet Port
      Backplane, CompactLogix System
        00, CompactLogix Processor, 1769-L35E/A LOGIX5335
        01, 1769-L35E Ethernet Port
        03, Local 1769 Bus Adapter, VA1769/A
          1769 Bus, 1769 Bus
            00, Local 1769 Bus Adapter, VA1769/A
            01, 1769-SDN Scanner Module, 1769-SDN
              Port2, DeviceNet
                00, 1769-SDN Scanner Module
                03, 1734-ADN PointIO DeviceNet Adapter
                04, 1734-IB4/B 4pt DC Input
                10, 1734D-IB8XOB8E
                13, 1734-IB4/B 4pt DC Input
                20, Series 9000(Strobe)-Retroflective w/cable
            02, 1769-2 Channel Analog Current/Voltage Output/B
```

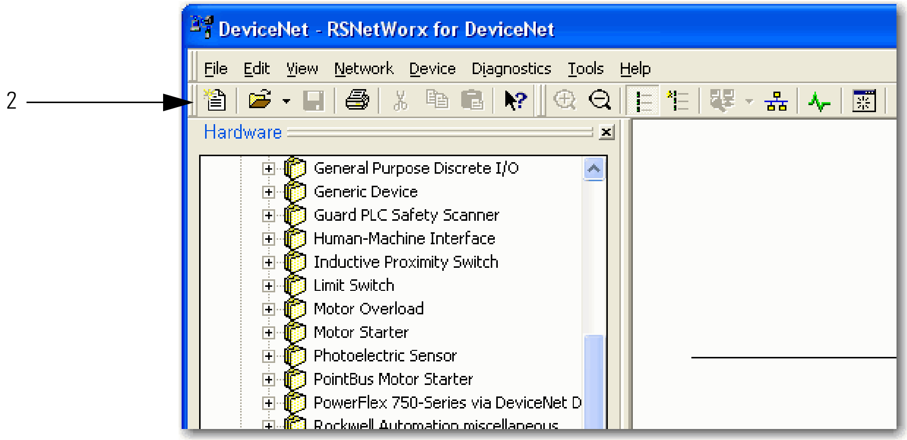
IMPORTANT	If you cannot view the network, verify that your computer is connected to the network. Refer to Chapter 2 on page 25 for more information on how to connect the computer to the network. If you can view the network but cannot see all the devices that should be on the network, verify the devices are connected to the network. Refer to Chapter 3 on page 29 for more information on how to connect the devices to the network.

Create a New File for the Network

Before you go online, you must create a new network configuration file.

Complete the following steps to create a DeviceNet configuration file.

1. Start RSNetWorx for DeviceNet software.
2. Create a file.


2
DeviceNet - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
Hardware
General Purpose Discrete I/O
Generic Device
Guard PLC Safety Scanner
Human-Machine Interface
Inductive Proximity Switch
Limit Switch
Motor Overload
Motor Starter
Photoelectric Sensor
PointBus Motor Starter
PowerFlex 750-Series via DeviceNet D
Rockwell Automation miscellaneous

Go Online to Your Network

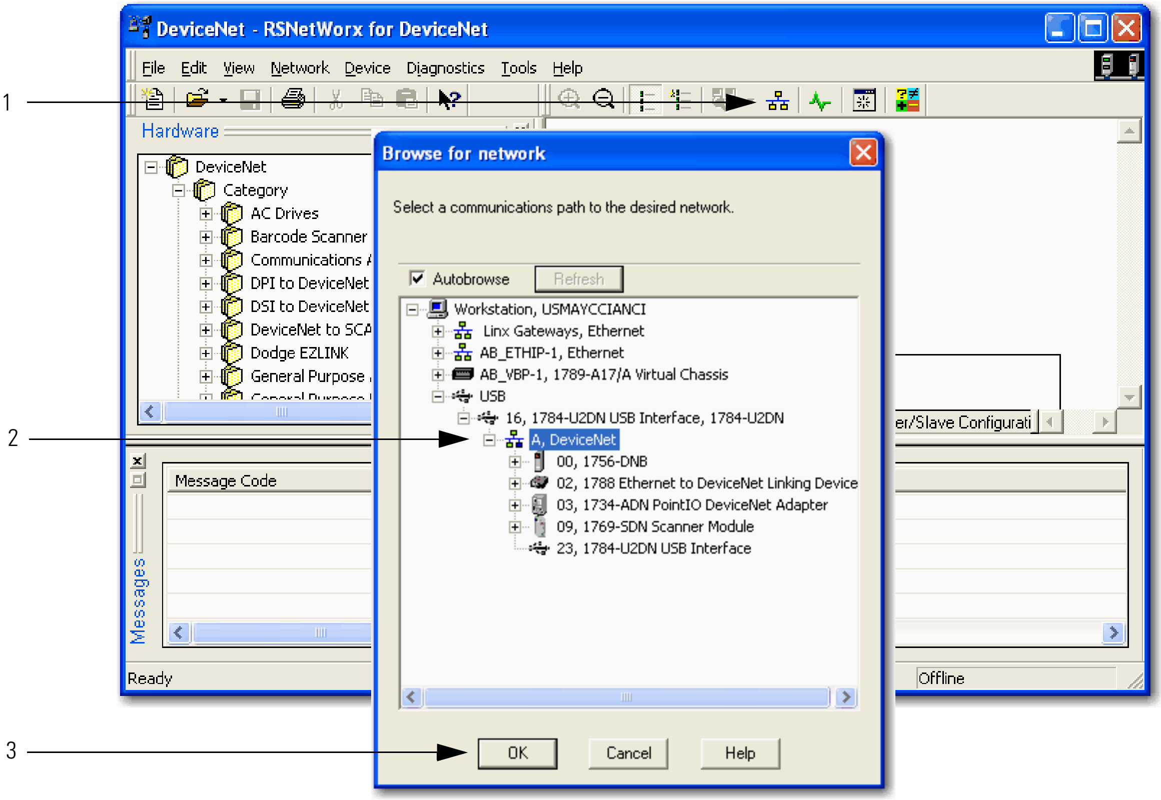
When you go online, RSNetWorx for DeviceNet software browses the network once and shows the devices currently on the network in the new network configuration file.

Keep in mind the following when you go online:

- RSNetWorx for DeviceNet software does not read (upload) or change (download) the parameters of any of the devices on the network.
- The picture that results from browsing remains static. It does not show any changes since the last browse.

Complete the following steps to go online.

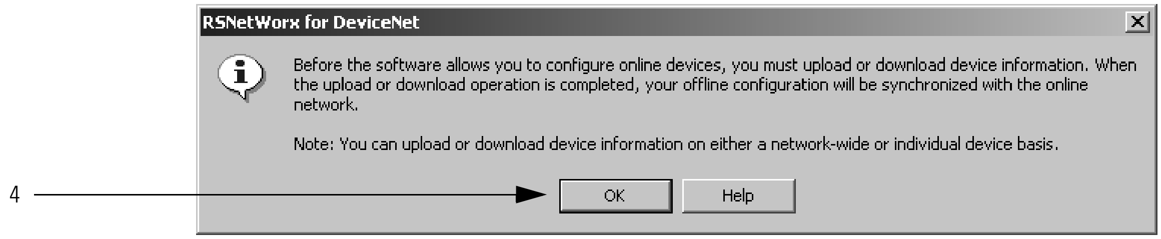
1. Click the Online button.
2. Select the DeviceNet network.
3. Click OK.



Select a communications path to the desired network.

1
2
3
DeviceNet - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
Hardware
DeviceNet
Category
AC Drives
Barcode Scanner
Communications A
DPI to DeviceNet
DSI to DeviceNet
DeviceNet to SCA
Dodge EZLINK
General Purpose
Messages
Message Code
Ready
Offline
er/Slave Configurati
Browse for network
☑ Autobrowse
Refresh
Workstation, USMAYCCIANCI
Linx Gateways, Ethernet
AB_ETHIP-1, Ethernet
AB_VBP-1, 1789-A17/A Virtual Chassis
USB
16, 1784-U2DN USB Interface, 1784-U2DN
A, DeviceNet
00, 1756-DNB
02, 1788 Ethernet to DeviceNet Linking Device
03, 1734-ADN PointIO DeviceNet Adapter
09, 1769-SDN Scanner Module
23, 1784-U2DN USB Interface
OK
Cancel
Help

4. When the pop-up message appears, click OK.

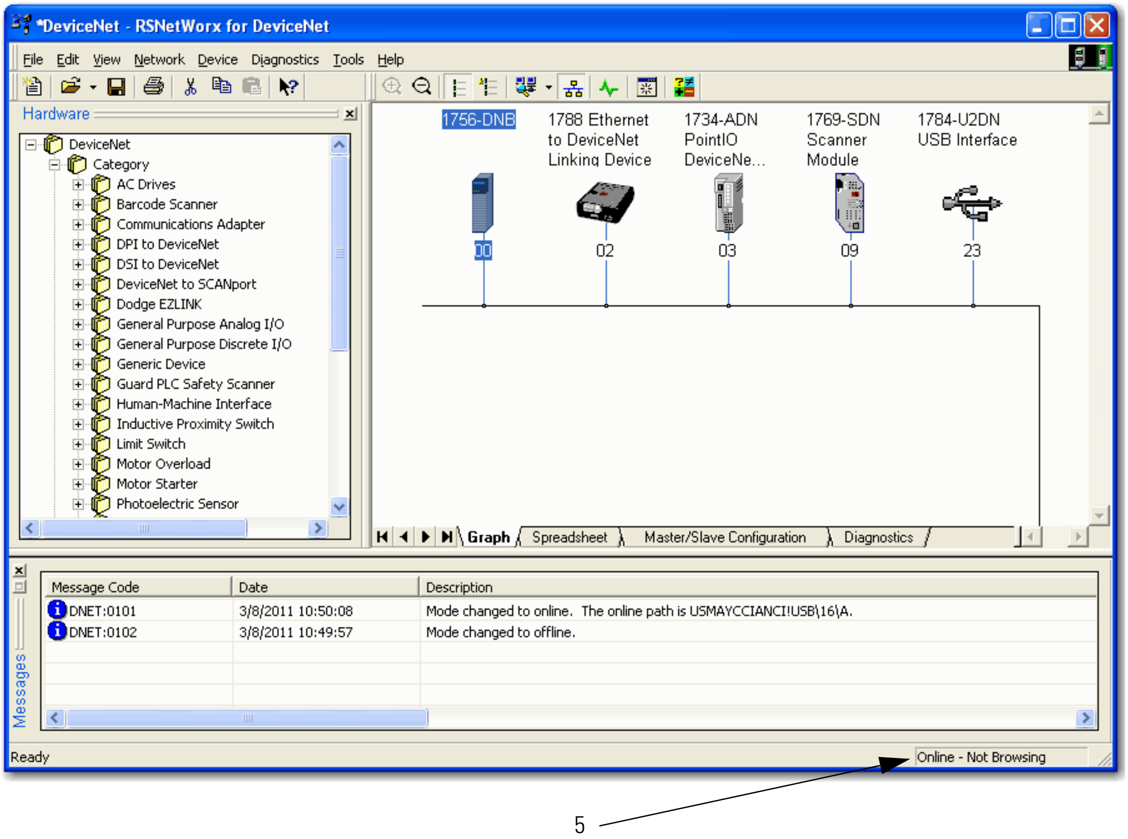


RSNetWorx for DeviceNet

Before the software allows you to configure online devices, you must upload or download device information. When the upload or download operation is completed, your offline configuration will be synchronized with the online network.

Note: You can upload or download device information on either a network-wide or individual device basis.

5. Verify that you are online.



Message Code	Date	Description
DNET:0101	3/8/2011 10:50:08	Mode changed to online. The online path is USMAYCCIANCI!USB\16\A.
DNET:0102	3/8/2011 10:49:57	Mode changed to offline.

Configure Each Device

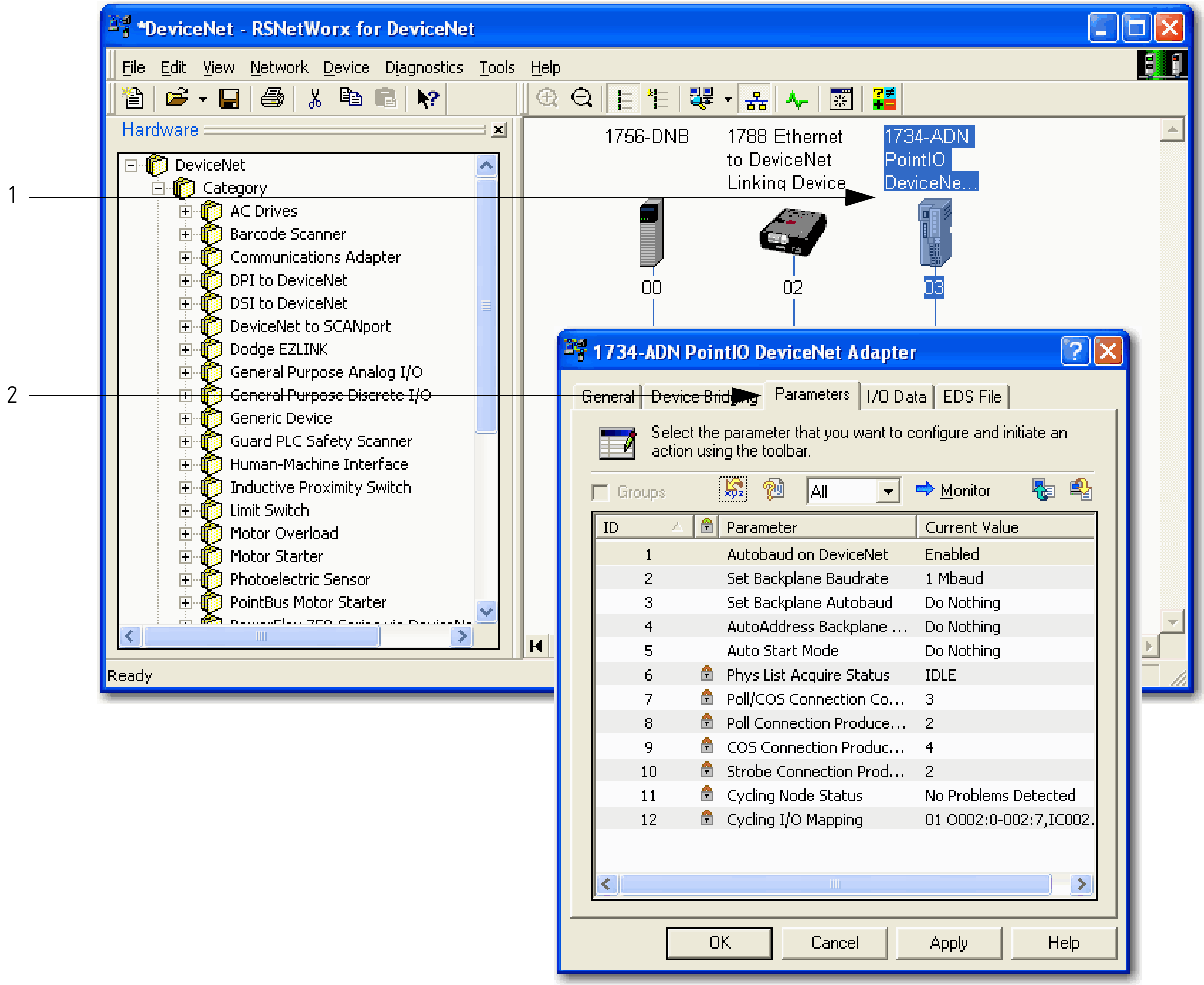
Once the devices on the DeviceNet network appear in the network configuration file, complete these tasks to change the configuration for a device:

- Upload the Configuration of a Device
- Change and Download Device Configuration

Upload the Configuration of a Device

When you configure the network online, the devices on the network have parameters configured. Complete the following steps to upload configuration from a device to the network configuration file.

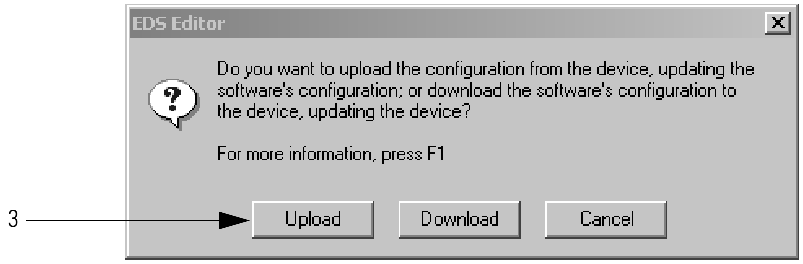
1. Double-click the device to open the configuration dialog box.
2. Click the Parameters tab.



Select the parameter that you want to configure and initiate an action using the toolbar.

ID	Parameter	Current Value
1	Autobaud on DeviceNet	Enabled
2	Set Backplane Baudrate	1 Mbaud
3	Set Backplane Autobaud	Do Nothing
4	AutoAddress Backplane ...	Do Nothing
5	Auto Start Mode	Do Nothing
6	Phys List Acquire Status	IDLE
7	Poll/COS Connection Co...	3
8	Poll Connection Produce...	2
9	COS Connection Produc...	4
10	Strobe Connection Prod...	2
11	Cycling Node Status	No Problems Detected
12	Cycling I/O Mapping	01 O002:0-002:7,IC002.

3. When prompted, upload the configuration from the device to the network configuration file.


EDS Editor

Do you want to upload the configuration from the device, updating the software's configuration; or download the software's configuration to the device, updating the device?

For more information, press F1

Change and Download Device Configuration

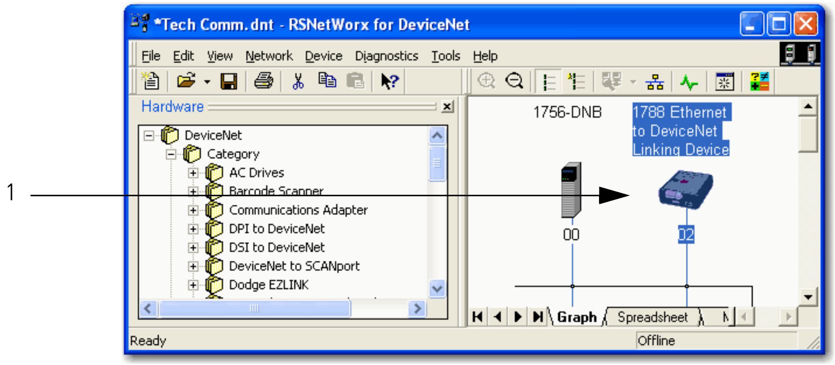
After you upload a device's configuration to the network configuration file, you can make changes to the configuration and download it.

Complete the following steps to change and download new configuration parameters.

1. Double-click the device to open the configuration dialog box.

   or

   If the device configuration has already been uploaded and the configuration dialog box is open, go to step 2.



   The configuration dialog box appears.

2. Click the appropriate tab.

3. Set a parameter to the desired new value.

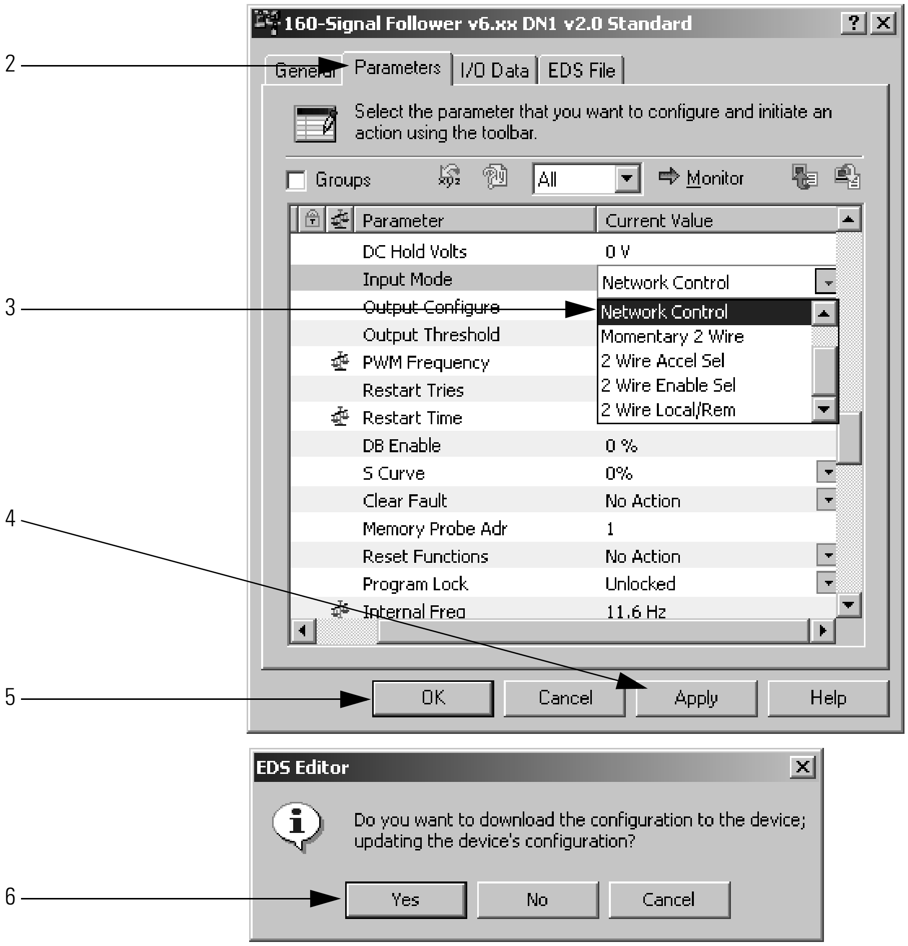
   Typically, there are two methods to change a parameter:

   - Choose a parameter from a pull-down menu
   - Type a new value

4. Apply the changes.

5. Click OK to close the dialog box.

6. When prompted, download the changes.



Select the parameter that you want to configure and initiate an action using the toolbar.

Parameter	Current Value
DC Hold Volts	0 V
Input Mode	Network Control
Output Configure	
Output Threshold	
PWM Frequency	
Restart Tries	
Restart Time	
DB Enable	0 %
S Curve	0%
Clear Fault	No Action
Memory Probe Adr	1
Reset Functions	No Action
Program Lock	Unlocked
Internal Freq	11.6 Hz

Network Control
Momentary 2 Wire
2 Wire Accel Sel
2 Wire Enable Sel
2 Wire Local/Rem

Do you want to download the configuration to the device; updating the device's configuration?

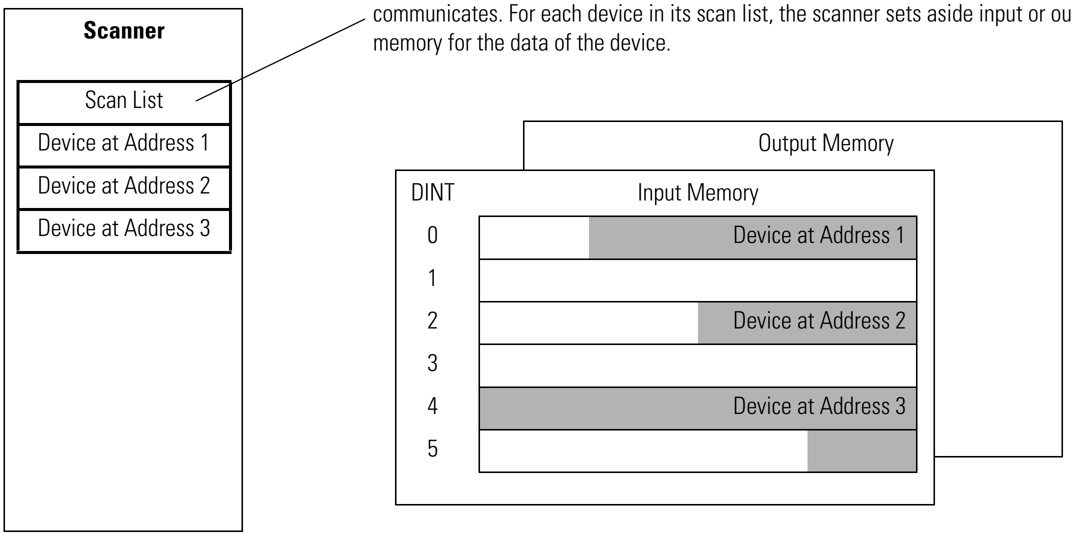
Configure the Scanner

A DeviceNet scanner manages input and output data for a controller. The scanner receives input data from I/O devices, organizes the information into scanner data tables, and sends the input data to the controller when the controller requests it. In addition, when the scanner receives output data from the controller, it sends the data to the I/O devices.

A DeviceNet scanner is the only device that can be used as a master on a DeviceNet network. When there is only one scanner on a network, it is the master for that network by default. When there are multiple scanners on the same network, each device can have only one scanner designated as its master, which is the scanner that controls its outputs.

You must configure the scanner to define how it communicates with other devices on the DeviceNet network. When you are configuring the network online, complete the following tasks to configure the scanner:

- Upload the Current Scanner Configuration
- Define the Scanner Properties
- Build the Scan List
- Set the Alignment Option

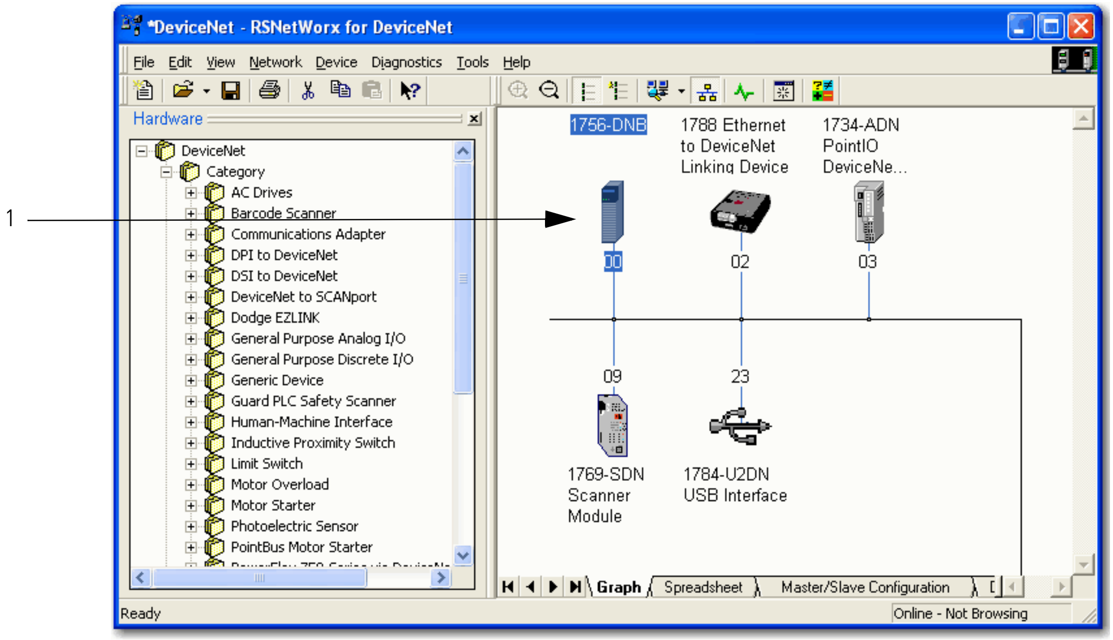


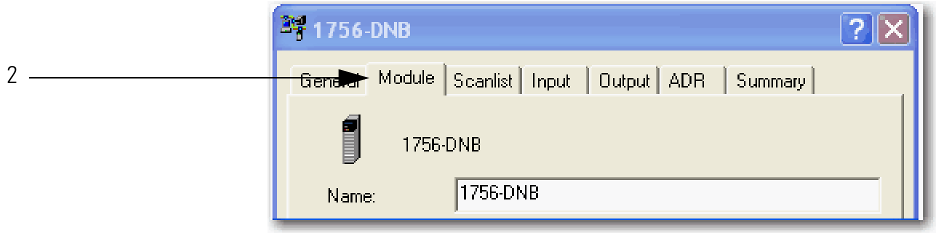
Scan list—A list in the scanner that identifies the devices with which the scanner communicates. For each device in its scan list, the scanner sets aside input or output memory for the data of the device.

Upload the Current Scanner Configuration

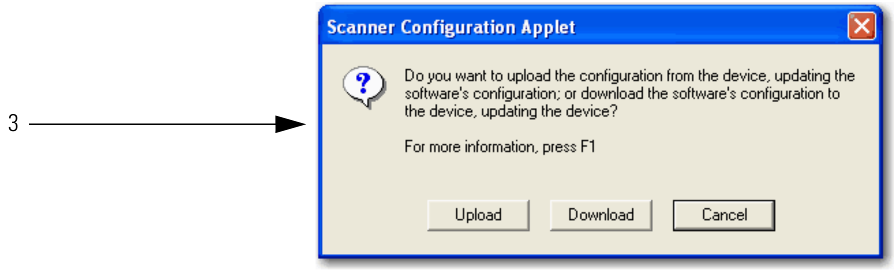
Complete the following steps to upload the current scanner configuration.

1. Double-click the scanner to open the configuration dialog box.
2. Click the Module tab.
3. When prompted, upload the configuration from the scanner.





Name: 1756-DNB



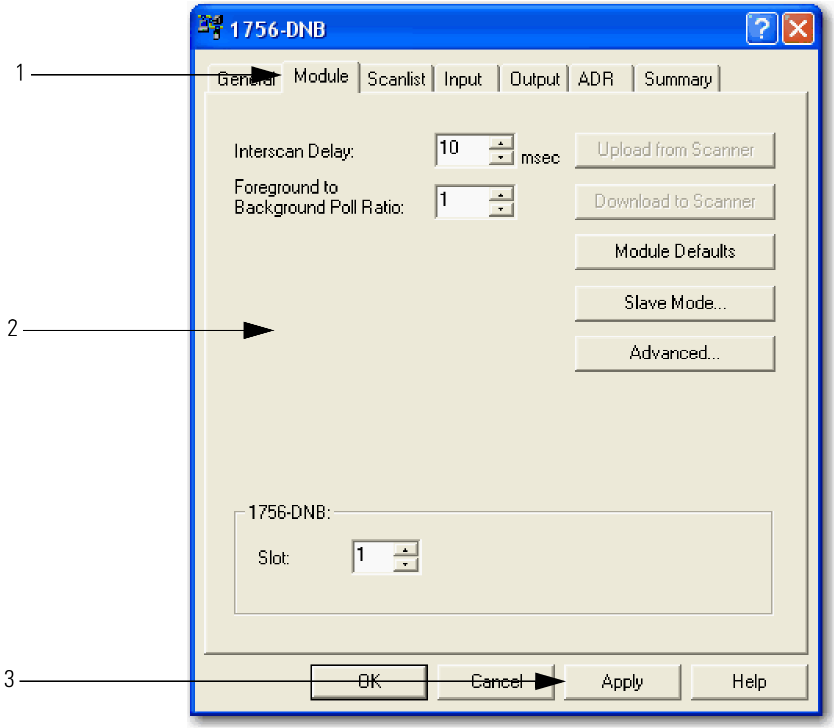
Do you want to upload the configuration from the device, updating the software's configuration; or download the software's configuration to the device, updating the device?

For more information, press F1

Define the Scanner Properties

Complete the following steps to change the scanner properties, if necessary.

1. Click the Module tab.
2. Make the necessary changes.
3. Click Apply to make the changes.
4. When a message prompts you to indicate whether to download your changes to the scanner, click No to continue configuring the scanner on additional tabs.



Interscan Delay: 10 msec

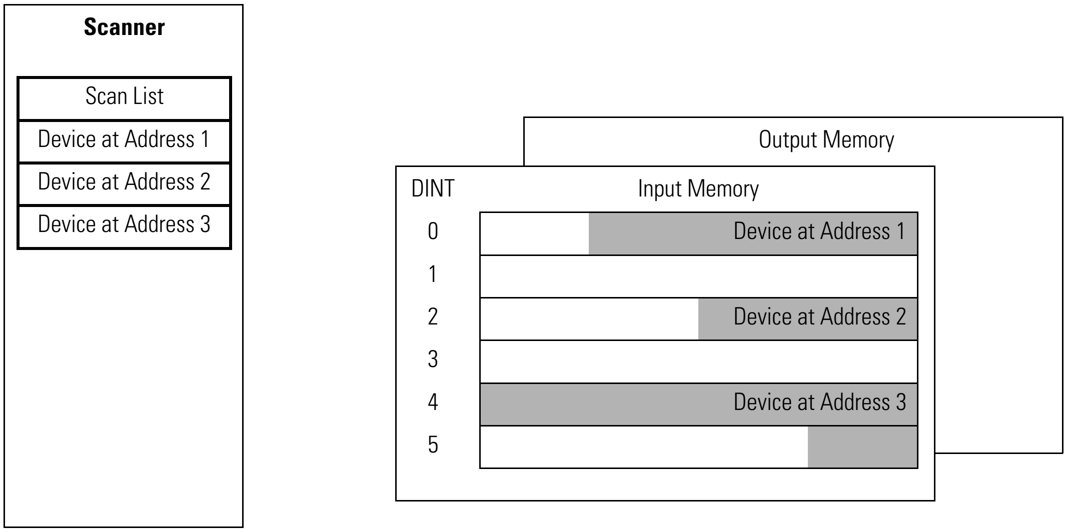
Foreground to Background Poll Ratio: 1

1756-DNB:

Slot: 1

Build the Scan List

A scan list is a list of devices with which the scanner communicates. For each device in the scanner's scan list, the scanner sets aside input or output memory for the data of the device.

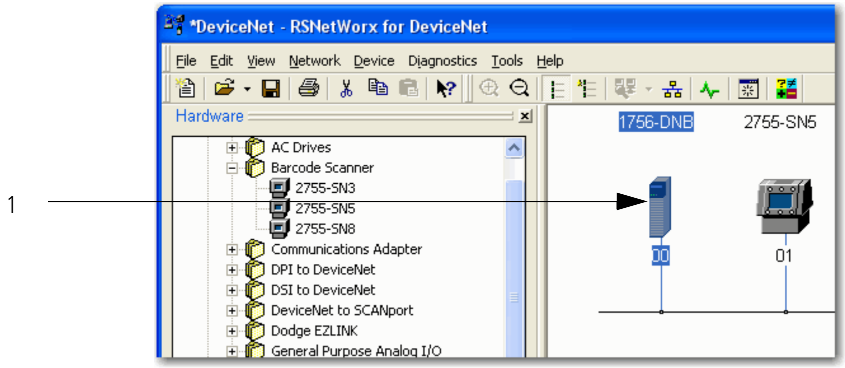


Complete the following steps to build a scan list.

1. Double-click the scanner to open the configuration dialog box.

   or

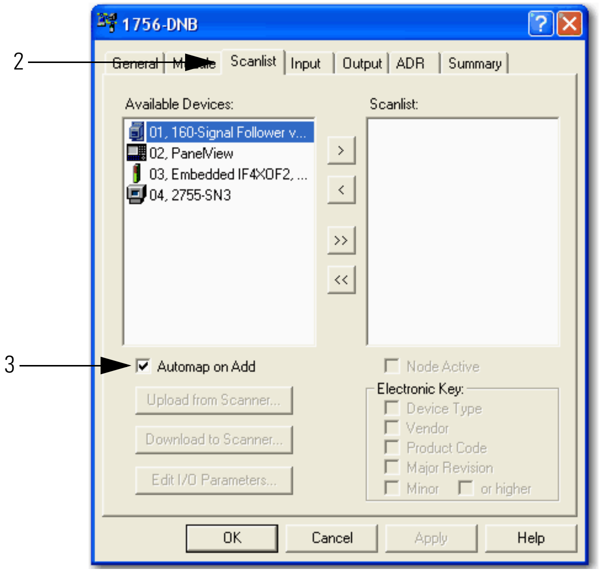
   If the scanner configuration has already been uploaded and the configuration dialog box is open, go to step 2.



2. Click the Scanlist tab.

   The devices on the network appear in the Available Devices column.

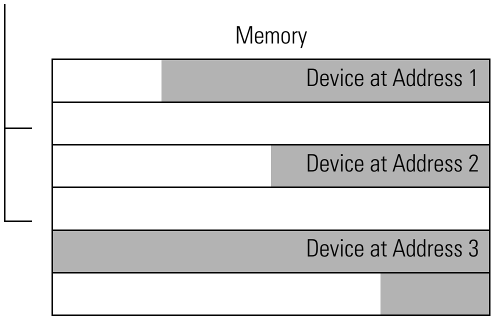
3. Clear or check Automap on Add.



RSNetWorx for DeviceNet software can automatically assign the memory location for each device.

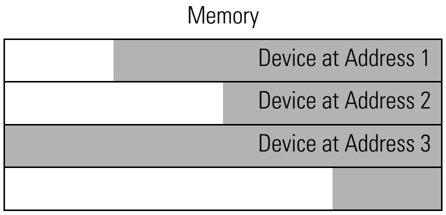
a. If you want to leave gaps between devices in the memory, as shown below, clear the box.

Leave Gaps Between Devices

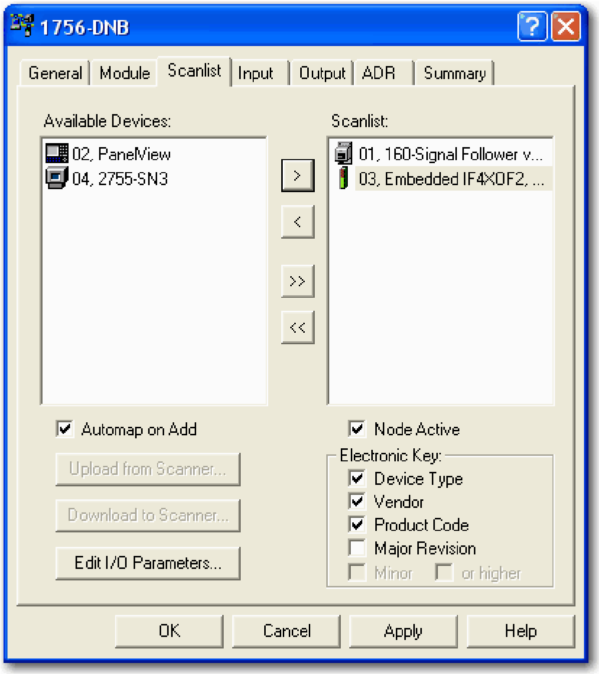


b. If you want to place devices in sequential DINT's, as shown below, leave the box checked. When you check the box, the software automatically assigns a memory location for each device as you add it to the scan list.

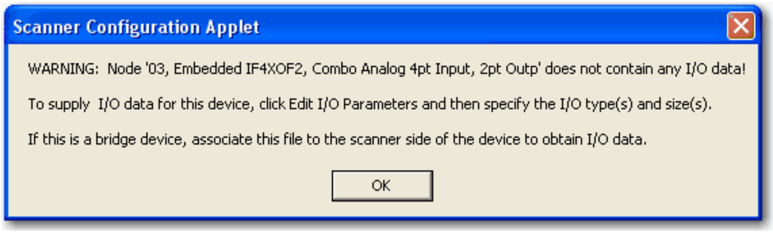
Place Devices in Sequential DINTs



4. Move devices from the Available Devices column to the Scanlist column.



If you get the following warning for a device, see Set the I/O Parameters of a Device on page 167.



WARNING: Node '03, Embedded IF4XOF2, Combo Analog 4pt Input, 2pt Outp' does not contain any I/O data!

To supply I/O data for this device, click Edit I/O Parameters and then specify the I/O type(s) and size(s).

If this is a bridge device, associate this file to the scanner side of the device to obtain I/O data.

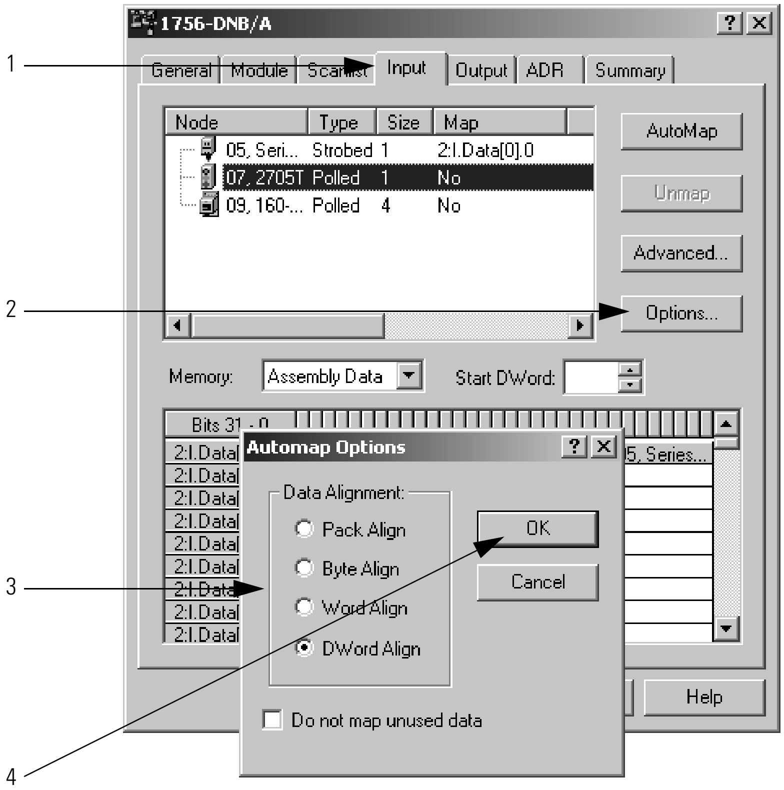
Set the Alignment Option

Use the alignment option to map the I/O data so that it is aligned on a boundary, such as a byte, word, or double-word, or efficiently grouped without alignment in the input or output memory map. To map I/O data so it is grouped without alignment, click the Pack Align option.

IMPORTANT	The alignment option you choose applies to both the input and output maps.

Complete the following steps to select an alignment option.

1. Click the Input tab.
2. Click Options.
3. Click the desired data alignment.
4. Click OK.



Node	Type	Size	Map
05, Seri...	Strobed	1	2:I.Data[0].0
07, 2705T	Polled	1	No
09, 160-...	Polled	4	No

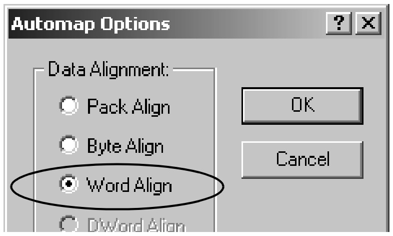
Memory: Assembly Data
Start DWord:

Data Alignment:
○ Pack Align
○ Byte Align
○ Word Align
● DWord Align

□ Do not map unused data

SoftLogix 5800 Controller

In SoftLogix 5800 applications, the 1784-PCIDS scanner organizes its input and output memory in 16-bit words. For that scanner, click Word Align.



Data Alignment:

○ Pack Align
○ Byte Align
● Word Align
○ DWord Align

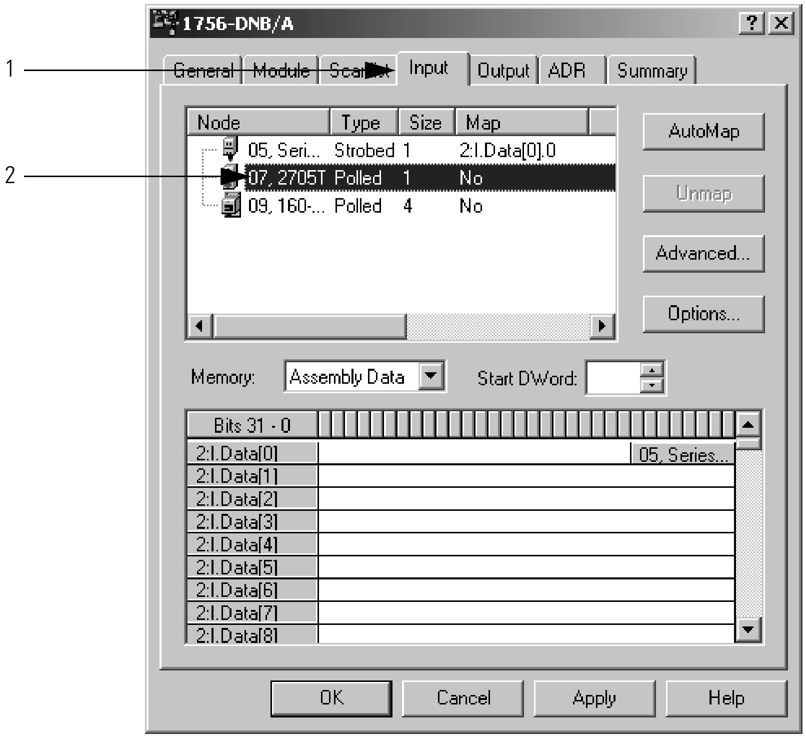
Manually Assign Each Device to a Memory Location

You can manually assign locations for device data.

IMPORTANT	If you configured the software to automatically assign memory locations as devices are added, as described on page 70, skip this section.

Complete the following steps to manually assign each device to a memory location.

1. Click the Input tab.
2. Select the device.



Node	Type	Size	Map
05, Seri...	Strobed	1	2:I.Data[0].0
07, 2705T	Polled	1	No
09, 160-...	Polled	4	No

Memory: Assembly Data
Start DWord:

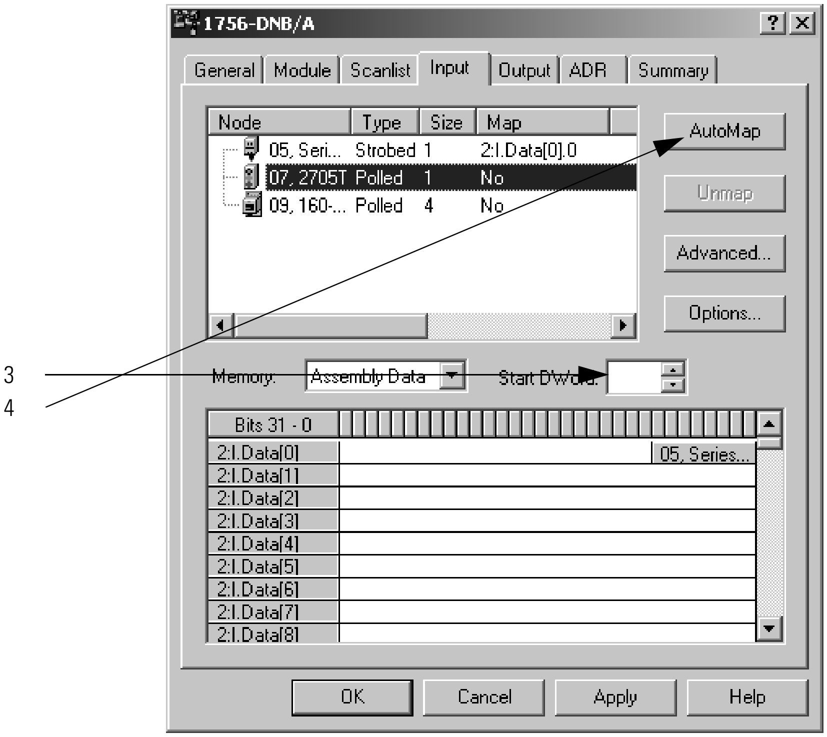
Bits 31 - 0	
2:I.Data[0]	05, Series...
2:I.Data[1]	
2:I.Data[2]	
2:I.Data[3]	
2:I.Data[4]	
2:I.Data[5]	
2:I.Data[6]	
2:I.Data[7]	
2:I.Data[8]	

3. In the Start DWord field, enter the element number to which you want to assign the data.

   This is the starting point for the data. Larger data sizes wrap to several elements. For example, to start the data in . . . Data[3], type 3 in the Start DWord box.

4. Click Automap.

   An entry for the device shows up in the input array.



Node	Type	Size	Map
05, Seri...	Strobed	1	2:I.Data[0].0
07, 2705T	Polled	1	No
09, 160-...	Polled	4	No

Memory: Assembly Data Start DWord:

Bits 31 - 0	
2:I.Data[0]	05, Series...
2:I.Data[1]	
2:I.Data[2]	
2:I.Data[3]	
2:I.Data[4]	
2:I.Data[5]	
2:I.Data[6]	
2:I.Data[7]	
2:I.Data[8]	

5. Click the Output tab and repeat step 2 through step 4.

6. Click OK to complete scanner configuration.

Sometimes, a specific input or output value may end up as the upper bytes of a DINT in the scanner.

Instance 70 Data Format (Basic Speed Control Input Assembly)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0						Running1		Faulted
1								
2	Speed Actual RPM (Low Byte)							
3	Speed Actual RPM (High Byte)							

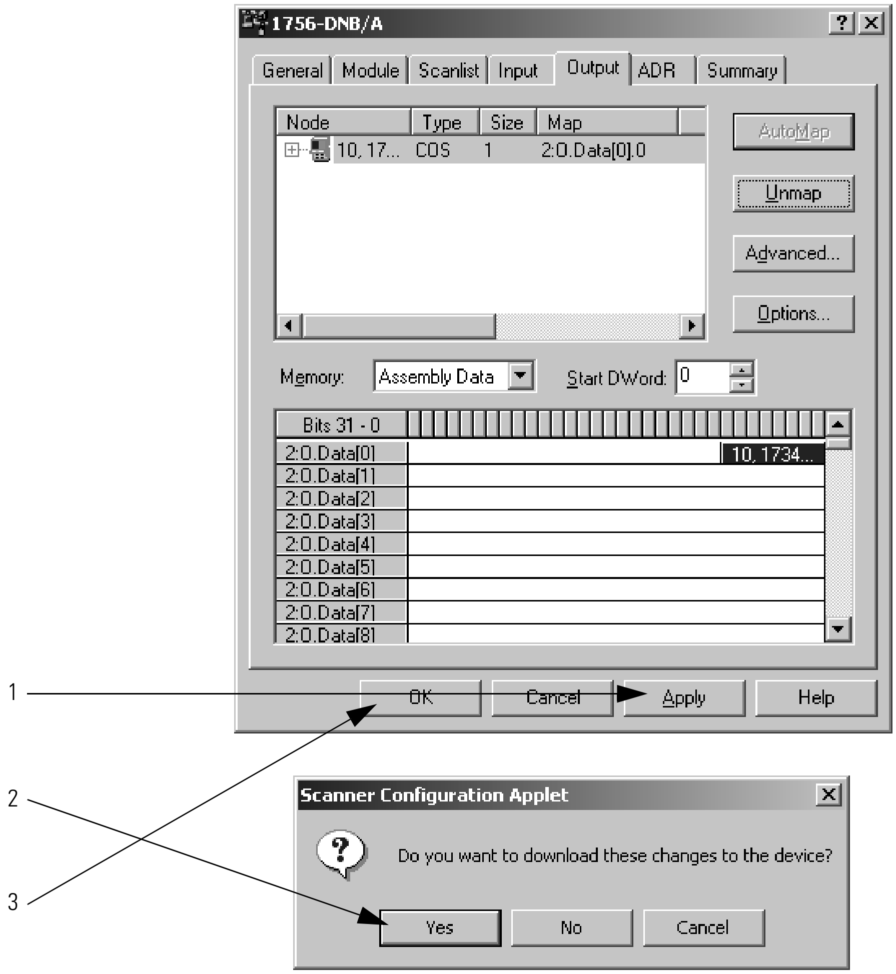
To make your programming easier, use advanced mapping to re-map the value to its own memory location. For more information, see Map the Memory Location with Advanced Mapping on page 183.

Download the Configuration to the Scanner

IMPORTANT	Make sure the scanner is in Idle mode. Complete one of the following tasks to put the scanner in Idle mode: • Turn off the …O.CommandRegister.Run bit of the scanner. • Place the controller in program/remote program mode.

Complete the following steps to download configuration to the scanner.

1. Apply the changes.
2. When the Scanner Configuration Applet warning appears, click Yes to download the changes to the device.
3. Click OK.



Node	Type	Size	Map
10, 17...	COS	1	2:0.Data[0].0

Memory: Assembly Data

Start DWord: 0

Bits 31 - 0	
2:0.Data[0]	10, 1734...
2:0.Data[1]	
2:0.Data[2]	
2:0.Data[3]	
2:0.Data[4]	
2:0.Data[5]	
2:0.Data[6]	
2:0.Data[7]	
2:0.Data[8]	

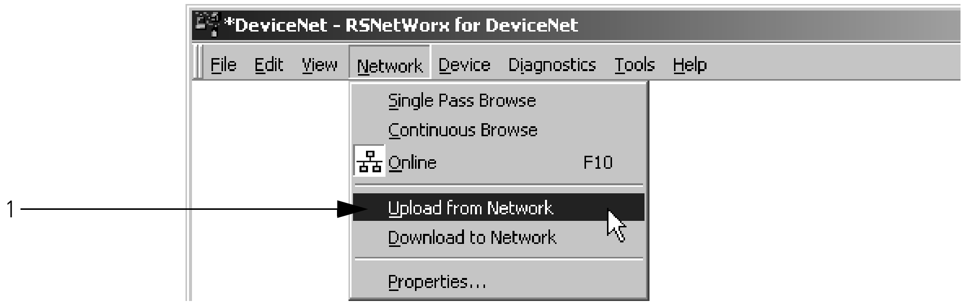
Do you want to download these changes to the device?

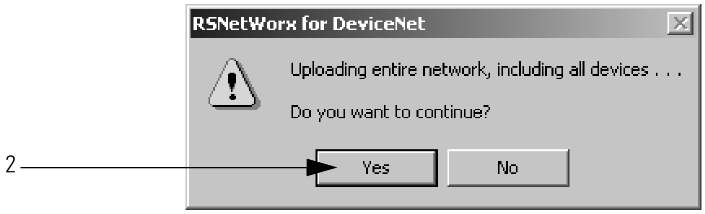
Upload and Save the Configuration File

Once you finish configuring the devices on the network, upload the entire network and save the file. This stores the configuration of each device in an offline file.

Complete the following steps to upload and save the configuration file.

1. From the File menu in RSNetWorx for DeviceNet software, choose Upload from Network.
2. When the warning appears, click Yes to upload the entire network.
3. Save the file.





Uploading entire network, including all devices . . .

Do you want to continue?

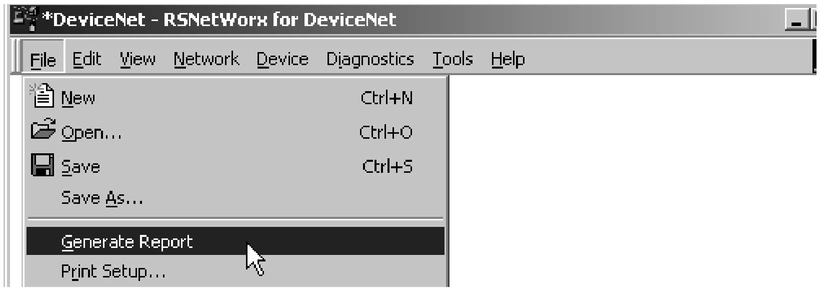
Generate an RSNetWorx for DeviceNet Report

An RSNetWorx for DeviceNet software report shows these items:

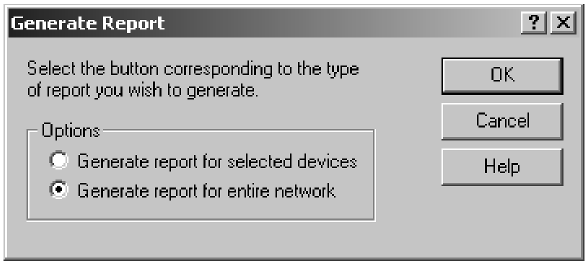
- Devices on the network
- Memory addresses of devices in the scanner
- Device configurations

The report is a useful reference when you program your system. Complete the following steps to generate a report.

1. From the File menu in RSNetWorx for DeviceNet software, choose Generate Report.


*DeviceNet - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
New Ctrl+N
Open... Ctrl+O
Save Ctrl+S
Save As...
Generate Report
Print Setup...

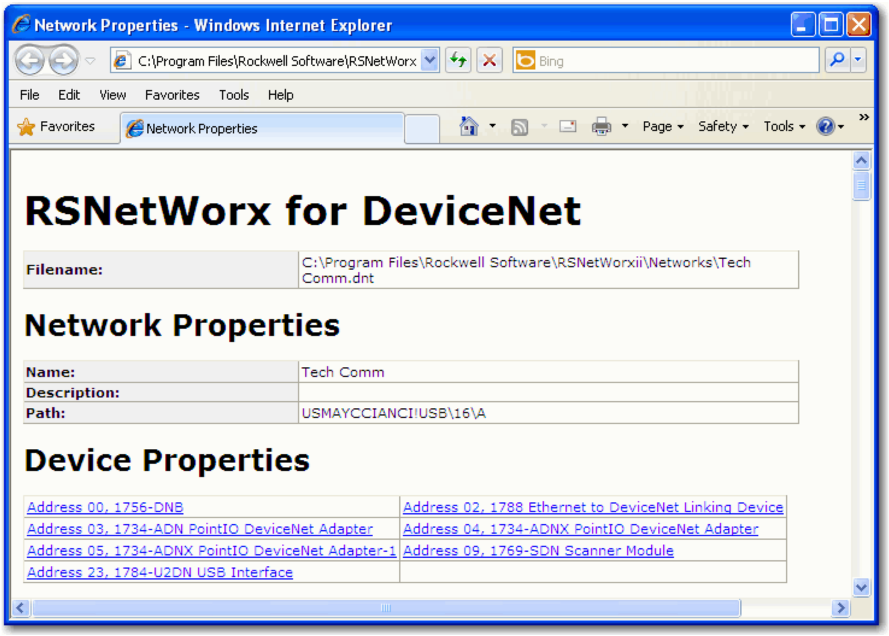
2. Click Generate report for entire network and click OK.



Select the button corresponding to the type of report you wish to generate.

Options
○ Generate report for selected devices
● Generate report for entire network

The report appears in your web browser.


Network Properties - Windows Internet Explorer
C:\Program Files\Rockwell Software\RSNetWorx
Bing
File Edit View Favorites Tools Help
Favorites
Network Properties
Page Safety Tools

RSNetWorx for DeviceNet

Filename:	C:\Program Files\Rockwell Software\RSNetWorxii\Networks\Tech Comm.dnt

Network Properties

Name:	Tech Comm
Description:	
Path:	USMAYCCIANCI!USB\16\A

Device Properties

Address 00, 1756-DNB	Address 02, 1788 Ethernet to DeviceNet Linking Device
Address 03, 1734-ADN PointIO DeviceNet Adapter	Address 04, 1734-ADNX PointIO DeviceNet Adapter
Address 05, 1734-ADNX PointIO DeviceNet Adapter-1	Address 09, 1769-SDN Scanner Module
Address 23, 1784-U2DN USB Interface	

Notes:

Chapter 6

Automatically Configure a DeviceNet Network

This chapter provides a quick method for configuring a DeviceNet network. It uses the AutoScan feature to establish communication between the controller and the devices on the DeviceNet network with minimal steps.

The DeviceNet AutoScan feature enables a scanner to automatically map a network of slave devices into its scan list without the use of RSNetWorx for DeviceNet software. This greatly improves the ease of setting up a DeviceNet network, especially networks comprised of simple devices.

When the feature is enabled, a DeviceNet scanner continuously searches for devices on the network. Once a qualifying slave device is found, it is added to the scanner's scan list and its I/O data is mapped into a predefined location in the scanner's I/O memory table based on the device's node address.

How AutoScan Operates

IMPORTANT AutoScan works only with 1756-DNB and 1769-SDN modules in Logix controller applications.

AutoScan is active when the feature is enabled and the scanner is in Idle mode. When active, the scanner attempts to connect to each device not enabled in the scan list. The scanner only checks for devices with node addresses between 0 and 61, inclusive. The connections to these devices are made on a round-robin basis.

When a device is found, the scanner gets the produced and consumed data sizes from the slave device's Connection Object instances.

- If the produced data size is greater than the configured I/O allocation size, the device is added to the scan list with a produced size set equal to the I/O allocation size.

  When this happens, an I/O connection is made with the device, but an error occurs and error code #77 appears on the 1769-SDN for the device's node number.

- If the consumed data size is greater than the configured I/O allocation size, then the node is rejected and not entered into the scan list.

However, you can change the I/O allocation size, as described in Configure I/O Allocation Size Via the User Program, to accommodate the device with the largest produced and consumed data sizes in your scan list.

For qualifying nodes, the scanner enters the device into the scan list and attempts to allocate an I/O connection using one of the following communication format choices in this particular order:

- Change Of State (COS) EPR = 250ms
- Poll EPR = 75ms
- Strobe EPR = 75ms
- Cyclic EPR = 500ms

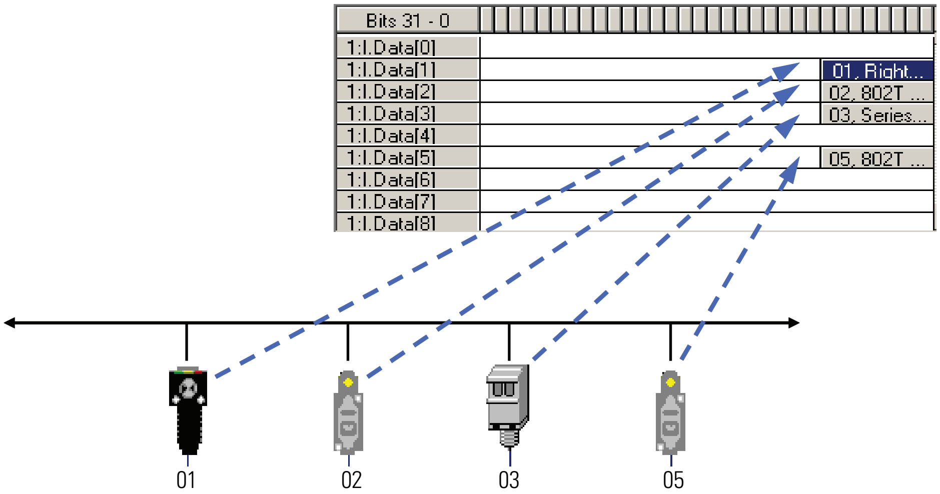
EXAMPLE If a photoeye is connected on a network that only supported strobed connections, the scanner executes the following tasks:

- The scanner recognizes that a device exists for which memory is available for the node number with the configured allocation size on a network that is not currently mapped.
- The scanner attempts to initiate both COS and polled connections first, but the strobed connection is selected as that is the only connection that the photoeye supported.

The input and output data is mapped into the scanner's I/O data table based on the device's node address and the configured fixed mapping size. The DINT-based formula that is used with the CompactLogix controller for calculating the input or output data location is as follows:

Input (Output) Offset = [(Node Address) x (Allocation Size)] / 4

EXAMPLE When using the default fixed mapping size of 4 bytes, the input data for the devices shown in the example below is allocated in the 1769-SDN's input table as shown below. Notice node 1 is in the data map at DINT location 1, node 2 at DINT location 2, and so on.



Bits 31 - 0	
1:I.Data[0]	
1:I.Data[1]	01, Right...
1:I.Data[2]	02, 802T ...
1:I.Data[3]	03, Series...
1:I.Data[4]	
1:I.Data[5]	05, 802T ...
1:I.Data[6]	
1:I.Data[7]	
1:I.Data[8]	

Notice that, in this example, node 4 is unused. However, the I/O memory slot remains allocated for it.

IMPORTANT If you are using a MicroLogix 1500 controller with a 1769-SDN scanner, you must use the following WORD-based formula for calculating the input or output data location:

Input (Output) Offset = ([(Node Address) x (Allocation Size)] / 2) + Data Offset

In this formula the Data Offset = 66 for Input Offset and 2 for Output Offset.

The data offset value is used to account for scanners that have a fixed status field at the start of the input or output data, such as the 1769-SDN scanner.

Determine If You Can Use AutoScan

Make sure your network meets these requirements to use this chapter:

- You have completed all the tasks required to do the following:
  - Connect a computer and devices to the network.
  - Create a network configuration file.
  - Go online.
  - Download the configuration file to the network.

  Refer to Chapter 2 through Chapter 5 to complete the tasks listed above.

- Your DeviceNet scanner must support the AutoScan feature. For more information, refer to your firmware release notes.

- Your application uses RSLogix 5000 programming software, version 13 or later.

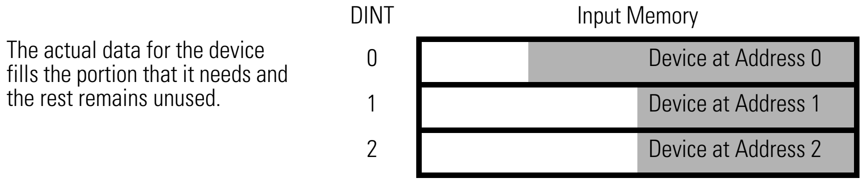
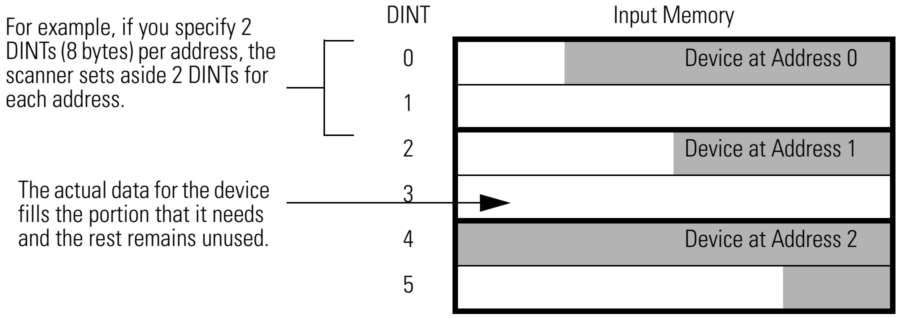
- The scanner's I/O allocation size is configured to accommodate the input and output data sizes of all devices on your DeviceNet network.

  The default AutoScan setting allocates a 4-byte entry in both the input and output memory maps in the scanner for each slave device detected on the network. This default size is chosen to accommodate the default Logix native data size of 32 bits, that is a DINT.

  If you use a device that sends more than 4 bytes of input or output data, such as an E3 Solid State Overload Relay (catalog number 193-EC*xx*), you must change the I/O allocation size.

How AutoScan Affects Your Network

As you use AutoScan, keep in mind the considerations described in the following table.

Consideration	Description
AutoScan clears the current configuration.	With AutoScan, the scanner automatically sets up communication with the devices on your DeviceNet network. When you turn on the AutoScan option, the scanner removes any previous configuration that was done to the scanner.
AutoScan allocates a fixed memory size for each device.	At its default setting, AutoScan allocates 1 DINT of input memory and 1 DINT of output memory for each device on the DeviceNet network.  The actual data for the device fills the portion that it needs and the rest remains unused. DINT Input Memory 0 1 2 Device at Address 0 Device at Address 1 Device at Address 2
The bytes/node value defines how much memory for each address.	AutoScan lets you specify how much input and output memory to give to each address on your network.  For example, if you specify 2 DINTs (8 bytes) per address, the scanner sets aside 2 DINTs for each address. The actual data for the device fills the portion that it needs and the rest remains unused. DINT Input Memory 0 1 2 3 4 5 Device at Address 0 Device at Address 1 Device at Address 2
New devices are automatically available.	While the scanner is in Idle mode, AutoScan continues to establish communication with devices that you connect to the network, as long as the devices use input data and output data sizes that fit in the scanner's I/O allocation size.
The Automatic Device Recovery (ADR) option is not available.	You have to use RSNetWorx for DeviceNet software to edit the configuration of the scanner to use the Automatic Device Recovery (ADR) option of a DeviceNet scanner. This turns off AutoScan.

Install the DeviceNet Node Commissioning Tool

Use the DeviceNet node commissioning tool to set a device's node address and baud rate when that device does not have a hardware mechanism to do so.

You can skip this step if either of the following conditions apply:

- All your devices have hardware mechanisms to set a node address and baud rate. In this case, you do not need the tool.
- You already have the tool installed.

Follow these steps to install the node commissioning tool.

1. On the RSLogix 5000 software CD, find the following folder where language is the language of your software:

   language\Tools\Node Commissioning Tool

   For example, for software in English, open the ENU folder.
2. Follow the instructions in the readmefirst file.

Connect Devices to the Network

When you use the AutoScan functionality, you should do the following:

- Install and configure the scanner and any network interface devices on the network first.
- Install other devices on the network once the scanner and network interface devices are on the network.

Install a Scanner or Network Interface Devices

Complete the following steps to install a scanner or network interface device on the DeviceNet network.

1. Connect the scanner and any network interface devices to the network.
2. Set a node address for the scanner and any network interface devices.

   Out of the box, a DeviceNet device is preset for node address 63. To avoid address conflicts, connect and configure the devices one at a time. Otherwise, the address conflicts may prevent communication.

The following addresses are recommended but not required.

Give this address	To this device
0	Scanner.
1…61	Your devices.
62	Computer interface to the network, such as a 1784-U2DN or 1788-PCIDS device.
63	Leave open. Out of the box, a DeviceNet communication module is preset for address 63. Leaving address 63 open lets you get a new device on the network without conflicting with another device.

- Refer to Set the Node Address of a Device on page 30 for more information on several options you can use to set the node address.
- Refer to Set the Node Address and Baud Rate with the DeviceNet Node Commissioning Tool on page 87 for more information on how to use just the DeviceNet node commissioning tool.

3. Set a baud rate for the scanner and any network interface devices.

When setting baud rates, consider the following:

- If you set the baud rate on the scanner or network interface device before you install other devices on the network, you reduce the number of baud rate errors.
- Scanners and network interface devices use a fixed baud rate.
- Sensors and similar DeviceNet communication modules use autobaud to set their baud rate. They wait for another device to communicate. Then they set their baud rate to the same baud rate as the other device.
- By first placing a scanner or network interface device on the network, the other device has a network baud rate against which to set its baud rate.
- Initially, leave the baud rate of the scanner and network interface at the default setting of 125KBps. If you want to change the baud rate, wait until after you establish communication with all your devices at the default setting (125K).
- Refer to Set the Node Address and Baud Rate with the DeviceNet Node Commissioning Tool on page 87 for more information.

Install Other DeviceNet Devices

Complete the following steps to install other devices on the DeviceNet network.

1. Connect the rest of your devices to the network one at a time.
2. Set a node address for each device after you add it to the network.

   Out of the box, a DeviceNet device is preset for node address 63. To avoid address conflicts, connect and configure the devices one at a time. Otherwise, the address conflicts may prevent communication.

   The following addresses are recommended but not required.

Give this address	To this device
0	Scanner
1…61	Your devices
62	Computer interface to the network, such as a 1784-U2DN or 1788-PCIDS device
63	Leave open. Out of the box, a DeviceNet communication module is preset for address 63. Leaving address 63 open lets you get a new device on the network without conflicting with another device.

   - Refer to Set the Node Address of a Device on page 30 for more information on how to use any of several options to set the node address.
   - Refer to Set the Node Address and Baud Rate with the DeviceNet Node Commissioning Tool on page 87 for more information on how to use the DeviceNet node commissioning tool.

3. Set a baud rate for each device after you add it to the network.

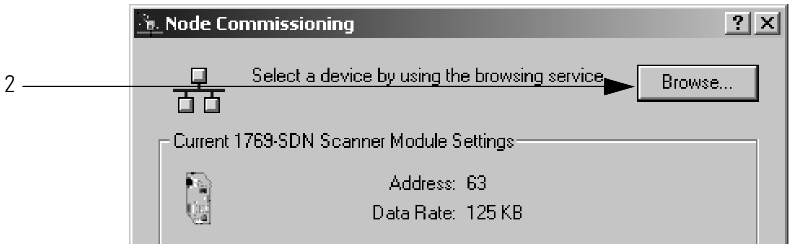
   When setting baud rates, consider the following:

   - Sensors and similar DeviceNet communication modules use autobaud to set their baud rate. They wait for another device to communicate. Then they set their baud rate to the same baud rate as the other device.
   - If a device has a hardware mechanism to set its baud rate, set it to autobaud, if available. Otherwise, set the device to the baud rate of the network.
   - After you change the address or baud rate of a device via a switch, cycle power to the device.
   - If a device has no hardware mechanism to set its address or baud rate, Refer to Set the Node Address and Baud Rate with the DeviceNet Node Commissioning Tool on page 87.
   - After you set the address of a device, check its network status indicator. Typically, a solid red indicator signifies an address conflict or problem with the baud rate.

Set the Node Address and Baud Rate with the DeviceNet Node Commissioning Tool

Complete the following steps to set a node address and baud rate with the DeviceNet node commissioning tool.

1. Start the node commissioning tool.
2. Click Browse.



Node Commissioning

Select a device by using the browsing service

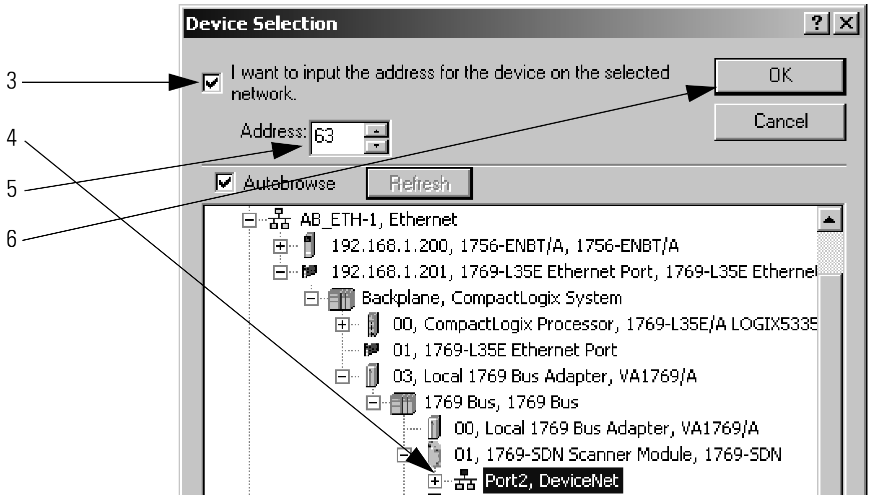
Current 1769-SDN Scanner Module Settings

Address: 63

Data Rate: 125 KB

3. Check I want to input the address for the device on the selected network.
4. Select the DeviceNet network.
5. Enter the current address for the device.

   Out of the box, a device uses address 63.
6. Click OK.



Device Selection

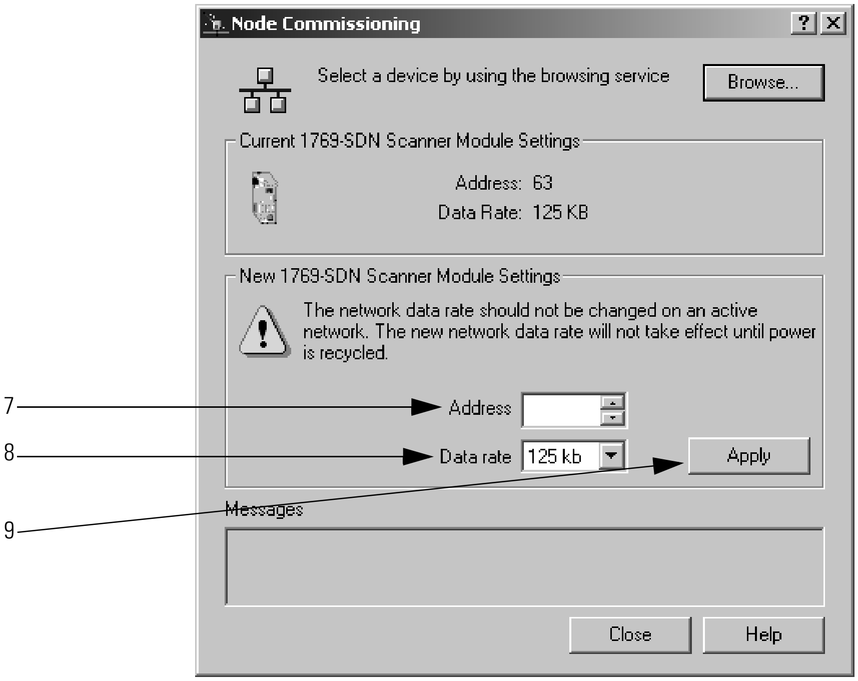
☑ I want to input the address for the device on the selected network.

Address: 63

☑ Autobrowse

```
AB_ETH-1, Ethernet
   192.168.1.200, 1756-ENBT/A, 1756-ENBT/A
   192.168.1.201, 1769-L35E Ethernet Port, 1769-L35E Ethernel
      Backplane, CompactLogix System
         00, CompactLogix Processor, 1769-L35E/A LOGIX5335
         01, 1769-L35E Ethernet Port
         03, Local 1769 Bus Adapter, VA1769/A
            1769 Bus, 1769 Bus
               00, Local 1769 Bus Adapter, VA1769/A
               01, 1769-SDN Scanner Module, 1769-SDN
                  Port2, DeviceNet
```

7. Enter the new address for the device.
8. Choose the baud rate for the device.
9. Click Apply.



Node Commissioning

Select a device by using the browsing service

Current 1769-SDN Scanner Module Settings

Address: 63
Data Rate: 125 KB

New 1769-SDN Scanner Module Settings

The network data rate should not be changed on an active network. The new network data rate will not take effect until power is recycled.

Address
Data rate 125 kb

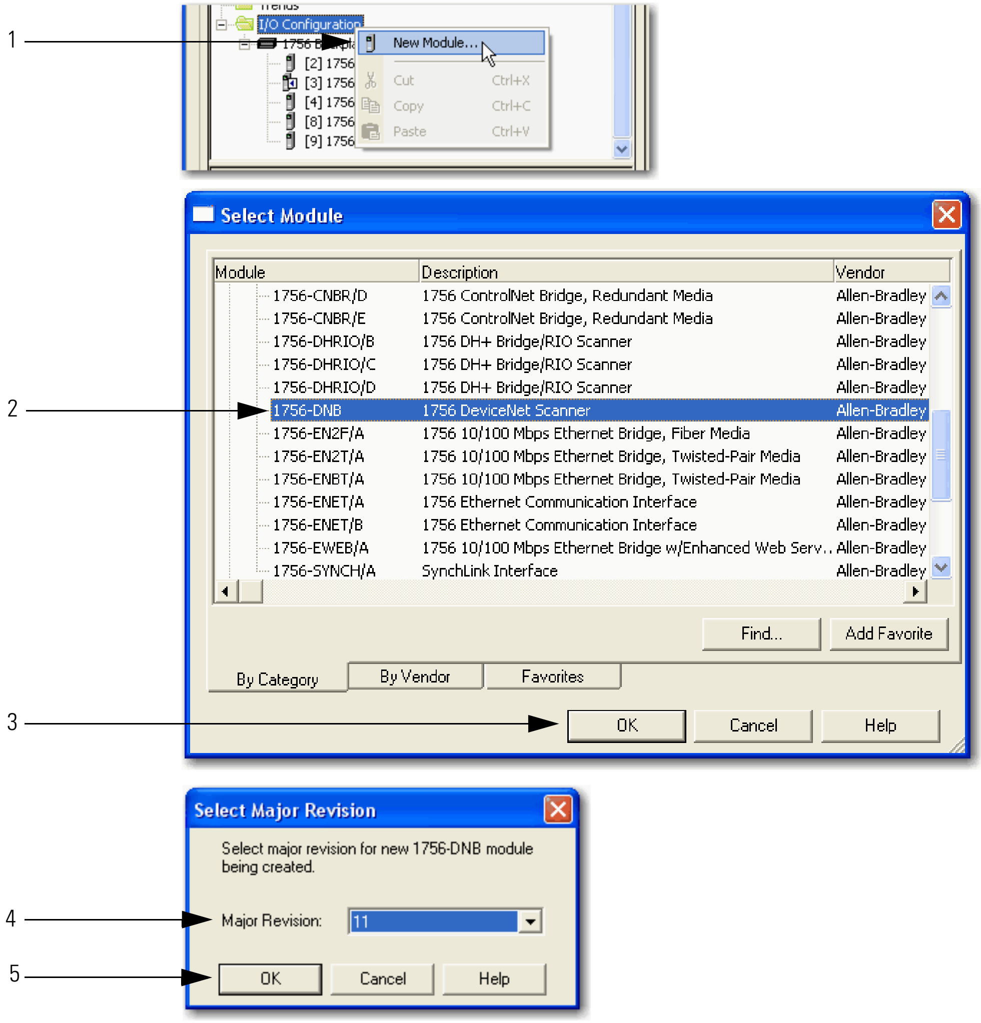
Messages

Add the Scanner to the RSLogix 5000 Project

To access the data of the network, add the scanner to the I/O configuration of the controller.

Add the Scanner to the I/O Configuration Folder

1. Right-click the I/O Configuration folder and choose New Module.
2. Select the type of scanner.
3. Click OK.
4. From the Major Revision pull-down menu, choose a scanner revision number.
5. Click OK.



Module	Description	Vendor
1756-CNBR/D	1756 ControlNet Bridge, Redundant Media	Allen-Bradley
1756-CNBR/E	1756 ControlNet Bridge, Redundant Media	Allen-Bradley
1756-DHRIO/B	1756 DH+ Bridge/RIO Scanner	Allen-Bradley
1756-DHRIO/C	1756 DH+ Bridge/RIO Scanner	Allen-Bradley
1756-DHRIO/D	1756 DH+ Bridge/RIO Scanner	Allen-Bradley
1756-DNB	1756 DeviceNet Scanner	Allen-Bradley
1756-EN2F/A	1756 10/100 Mbps Ethernet Bridge, Fiber Media	Allen-Bradley
1756-EN2T/A	1756 10/100 Mbps Ethernet Bridge, Twisted-Pair Media	Allen-Bradley
1756-ENBT/A	1756 10/100 Mbps Ethernet Bridge, Twisted-Pair Media	Allen-Bradley
1756-ENET/A	1756 Ethernet Communication Interface	Allen-Bradley
1756-ENET/B	1756 Ethernet Communication Interface	Allen-Bradley
1756-EWEB/A	1756 10/100 Mbps Ethernet Bridge w/Enhanced Web Serv...	Allen-Bradley
1756-SYNCH/A	SynchLink Interface	Allen-Bradley

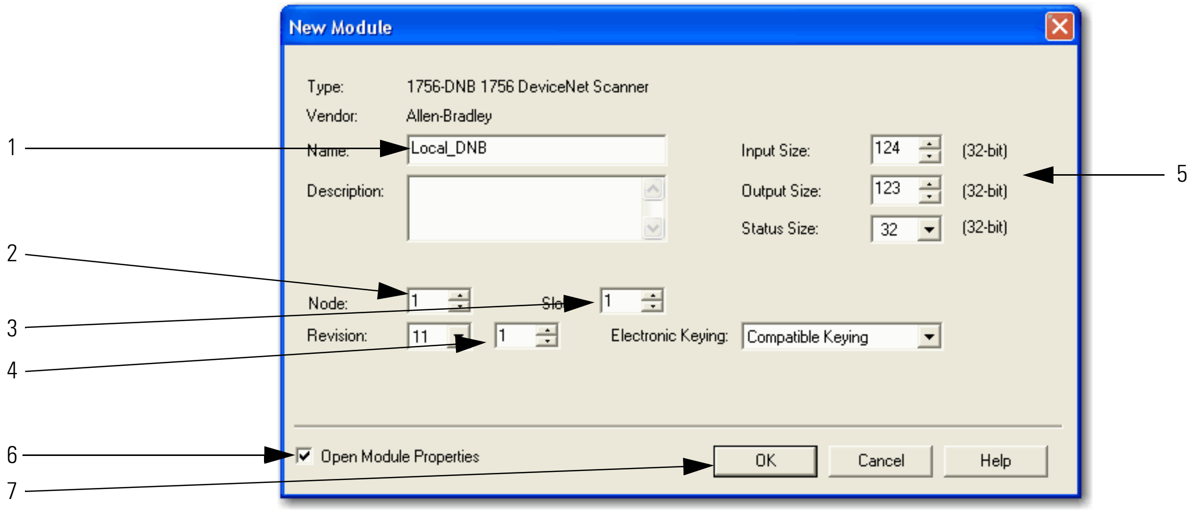
Select major revision for new 1756-DNB module being created.

Major Revision: 11

Define the Properties of the Scanner

1. Type a name for the scanner.
2. Enter the scanner node number.
3. Enter the scanner slot number.
4. Enter the scanner minor revision.
5. Enter the size of the input and output memory maps that the scanner will allocate for each device it detects on the network.

   Valid values range from 0…32 bytes per node.
6. If you need to make additional configuration changes, such as setting the Requested Packet Interval (RPI), check Open Module Properties.
7. Click OK.



Type: 1756-DNB 1756 DeviceNet Scanner

Vendor: Allen-Bradley

Name: Local_DNB

Description:

Input Size: 124 (32-bit)

Output Size: 123 (32-bit)

Status Size: 32 (32-bit)

Node: 1

Slo: 1

Revision: 11 1

Electronic Keying: Compatible Keying

☑ Open Module Properties

8. If the Module Properties dialog box appears, make any additional configuration changes.

Enable AutoScan with RSLogix 5000 Software

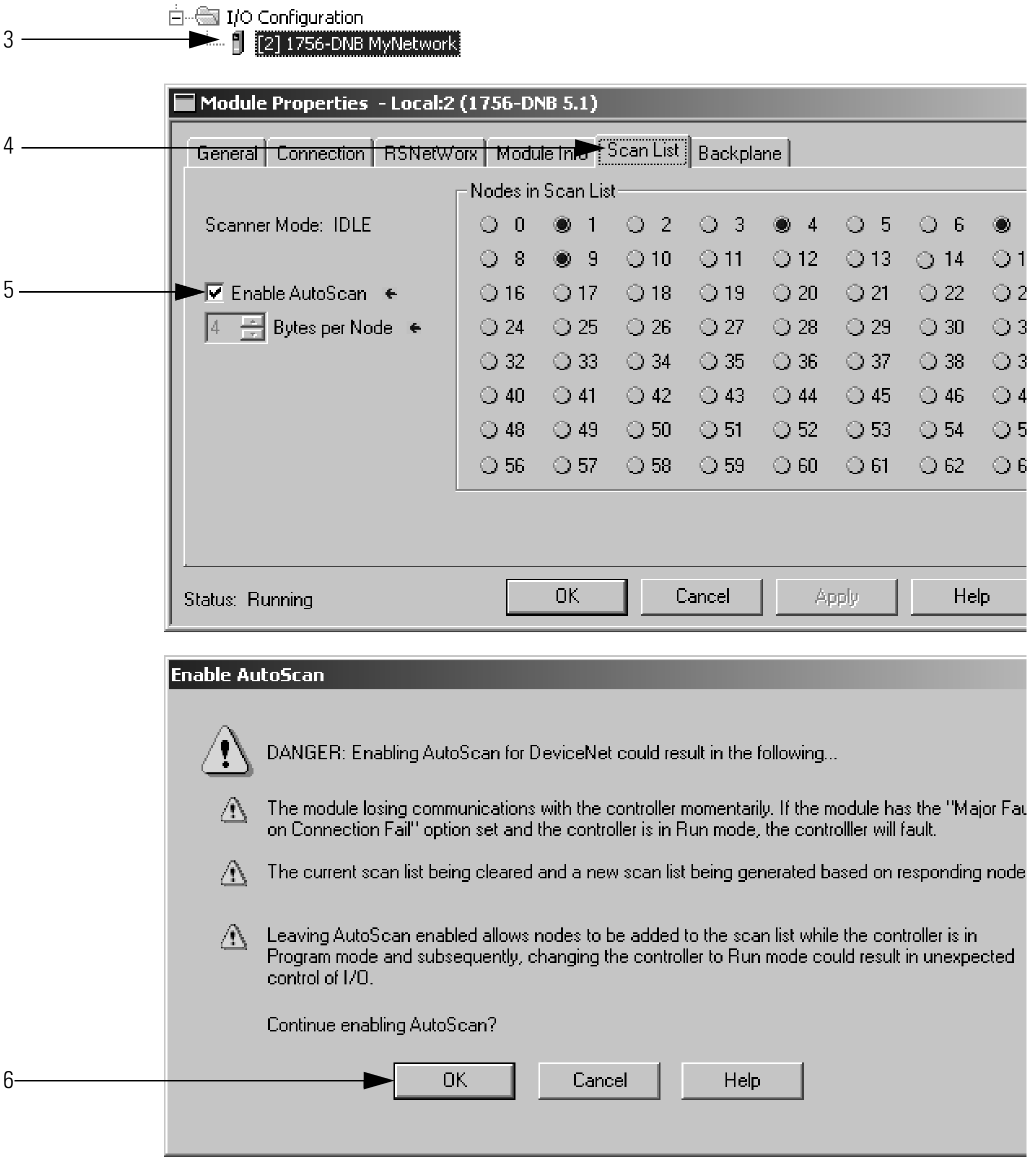
To enable AutoScan with RSLogix 5000 software, follow these steps.

1. Save changes to your RSLogix 5000 project.
2. Download the project to the Logix5000 controller.

IMPORTANT	In the following steps, you clear any existing configuration from the scanner and reconfigure it to communicate with the devices on the network. In the controller, this may change the tag addresses of the devices. If you have already programmed your logic, make sure that it still addresses the correct data.

3. Double-click the scanner in the Controller Organizer to access its properties.
4. Click the Scan List tab.

   A blue dot in the Nodes in Scan List section indicates a device that the scanner now controls.
5. Check Enable AutoScan.
6. When the Enable AutoScan warning appears, click OK.



Module Properties - Local:2 (1756-DNB 5.1)

Scanner Mode: IDLE

☑ Enable AutoScan

4 Bytes per Node

Nodes in Scan List

○ 0	● 1	○ 2	○ 3	● 4	○ 5	○ 6	●
○ 8	● 9	○ 10	○ 11	○ 12	○ 13	○ 14	○ 1
○ 16	○ 17	○ 18	○ 19	○ 20	○ 21	○ 22	○ 2
○ 24	○ 25	○ 26	○ 27	○ 28	○ 29	○ 30	○ 3
○ 32	○ 33	○ 34	○ 35	○ 36	○ 37	○ 38	○ 3
○ 40	○ 41	○ 42	○ 43	○ 44	○ 45	○ 46	○ 4
○ 48	○ 49	○ 50	○ 51	○ 52	○ 53	○ 54	○ 5
○ 56	○ 57	○ 58	○ 59	○ 60	○ 61	○ 62	○ 6

Status: Running

Enable AutoScan

DANGER: Enabling AutoScan for DeviceNet could result in the following...

The module losing communications with the controller momentarily. If the module has the "Major Fau on Connection Fail" option set and the controller is in Run mode, the controlller will fault.

The current scan list being cleared and a new scan list being generated based on responding node

Leaving AutoScan enabled allows nodes to be added to the scan list while the controller is in Program mode and subsequently, changing the controller to Run mode could result in unexpected control of I/O.

Continue enabling AutoScan?

Initiate AutoScan via the User Program

The DeviceNet AutoScan feature enables a scanner to automatically map a network of slave devices into its scan list without the use of RSNetWorx for DeviceNet software. This greatly improves the ease of setting up a DeviceNet network, especially networks comprised of simple devices.

When the feature is enabled, a DeviceNet scanner continuously searches for devices on the network. Once a qualifying slave device is found, it is added to the scanner's scan list and, based on the device's node address, its I/O data is mapped into a predefined location in the scanner's I/O memory table.

Implementing AutoScan

To implement this feature, make sure that the appropriate version of DeviceNet scanner is used. See page 97 for the list of compatible products supporting this feature.

This section describes how to set up the feature and how it operates. Notice that explicit messaging is used for some of the steps. An explicit message can be sent on a DeviceNet network in the following ways:

- A user ladder program
- External programming/configuration devices, such as the hand-held DeviceNet Configuration Terminal, catalog number 193-DNCT
- RSNetWorx for DeviceNet software

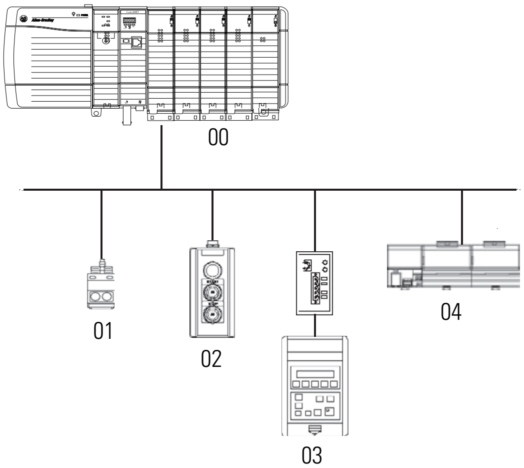
Since the purpose of the feature is to eliminate the use of RSNetWorx for DeviceNet software, instructions on how to send an explicit message via the class instance editor in the software are not covered in this document.

To implement the feature, follow these steps.

1. Set up the physical network.

   Make sure all devices are addressed appropriately. For example, be sure there are no address conflicts and devices are communicating at the same baud rate.

   The diagram below shows an example system using the 1756-DNB scanner.



   You can commission the node addresses via hardware switches on the devices or through other DeviceNet configurators, such as the hand-held DeviceNet Configuration Terminal. For more information on how to set up the DeviceNet Configuration Terminal's node address, see the DeviceNet Configuration Terminal User Manual, publication 193-UM009A-EN-P.

2. Set up I/O allocation size in the scanner.

   TIP This step is optional.

   The default AutoScan setting allocates a 4-byte entry in both the input and output memory maps in the scanner for each slave device detected on the network. This default size is chosen to accommodate the default Logix native data size of 32 bits (DINT). If that is adequate for the application, go to step 3.

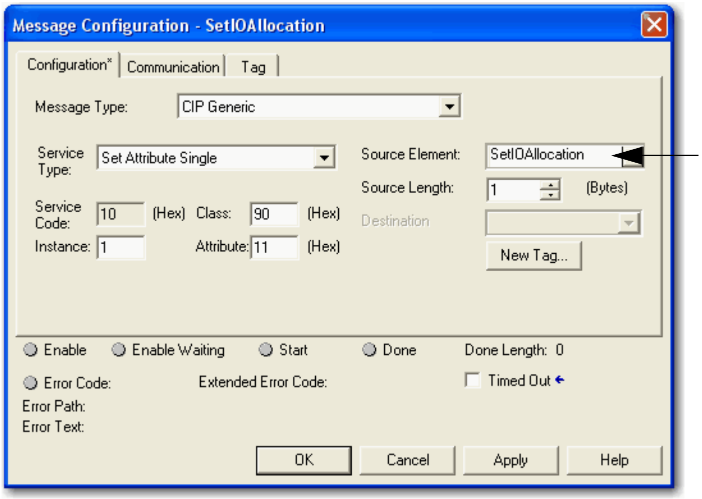
   For applications where you want to customize the I/O allocation size, the 4-byte allocation can be adjusted through an explicit message to the scanner using the SetAttributeSingle service. The entry allocation can be configured for 1 to 32 bytes per node.

   Configure the allocation size using one of these methods:

   - Configure I/O Allocation Size Via the User Program
   - Configure I/O Allocation Via a DeviceView Configurator

Configure I/O Allocation Size Via the User Program

Use the parameters shown in the Message Configuration dialog box below to adjust the I/O allocation size. Make sure that the message is sent to the appropriate DeviceNet scanner.



Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex) Class: 90 (Hex)

Instance: 1 Attribute: 11 (Hex)

Source Element: SetIOAllocation

Source Length: 1 (Bytes)

Destination

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:

This data tag should be configured as an SINT, and should contain the value of the desired per-node fixed mapping size (1 - 32).

Configure I/O Allocation Via a DeviceView Configurator

Rockwell Automation offers the hand-held DeviceNet Configuration Terminal, catalog number 193-DNCT, to configure individual devices on a DeviceNet network.

To configure the I/O allocation size, attach a configurator device on the network and send an explicit message to the scanner by using the parameters below. Send the desired allocation size (1…32 bytes) to the attribute below to configure the per-node I/O allocation.

Field	Value
Service Code	10 Hex
Class	90 Hex
Instance	1
Attribute	11 Hex

For more information on how to use the DeviceNet Configuration Terminal, refer to these publications:

- 193-DNCT DeviceNet Configuration Terminal Quick Reference, publication 193-QR002A-EN-P
- DeviceNet Programming Terminal User Manual, publication 193-UM009A-EN-P

IMPORTANT	You can change the I/O allocation size only when the scanner is in Idle mode, and the AutoScan feature is disabled.

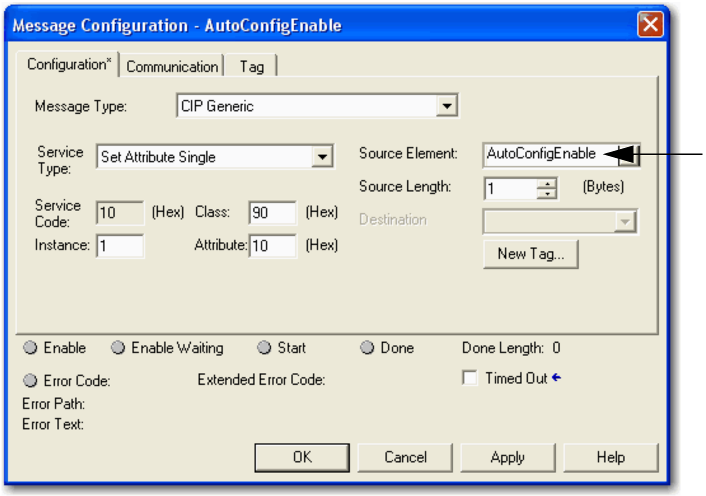
3. Enable AutoScan.

   This is accomplished by executing an explicit message to the scanner by using the SetAttributeSingle service. As mentioned before, there are multiple ways to send an explicit message on DeviceNet, including the following:

   - Initiate AutoScan Via the User Program
   - Initiate AutoScan via the DeviceView Configurator

Initiate AutoScan Via the User Program

To enable AutoScan by using the MSG instruction, use the parameters shown below and make sure that the message is sent to the appropriate DeviceNet scanner. The figure shown below is from RSLogix 5000 software. Refer to the appropriate user manuals to determine how to perform explicit messaging in other PLC platforms.



Message Configuration - AutoConfigEnable

Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex) Class: 90 (Hex)

Instance: 1 Attribute: 10 (Hex)

Source Element: AutoConfigEnable

Source Length: 1 (Bytes)

Destination

Done Length: 0

Error Code:

Extended Error Code:

Error Path:

Error Text:

This data tag should be configured as a SINT. Upon execution of the MSG, AutoScan is:
Enabled if tag = 1
Disabled if tag = 0

Initiate AutoScan via the DeviceView Configurator

To enable AutoScan by using a DeviceNet configurator, attach the device on the network and send an explicit message to the scanner using the parameters below. Send a 1 to that attribute to enable the feature, and 0 to disable.

Field	Value
Service Code	10 Hex
Class	90 Hex
Instance	1
Attribute	11 Hex

For more information on how to use the DeviceNet Configuration Terminal, refer to these publications:

- 193-DNCT DeviceNet Configuration Terminal Quick Reference, publication 193-QR002A-EN-P
- DeviceNet Programming Terminal User Manual, publication 193-UM009A-EN-P.

IMPORTANT You can change the I/O allocation size only when the scanner is in Idle mode, and the AutoScan feature is disabled.

Once the feature is enabled, the scanner scans the network to populate and configure the scan list automatically.

4. Put scanner in RUN mode to begin system operation.

Additional Considerations Regarding AutoScan

The factory default setting for AutoScan is disabled for all products.

Make sure that input or output data memory size in the scanner is large enough to accommodate the size required based on the number of nodes on the network and the AutoScan I/O allocation size per node.

EXAMPLE If the I/O allocation size per node is configured for 16 bytes and there are 32 slave devices on the network (node addresses 1…32), AutoScan requires 16 bytes x 32 = 512 bytes (128 DINT) of I/O space in both the scanner's input and output table. Assuming it is a ControlLogix system, the maximum scanner input data table size is 124 DINT and 123 DINT for output. The required space exceeds what the 1756-DNB can support. You would need to adjust the I/O allocation size or reduce the slave device count on the network to include all of the devices in the scan list.

⚠ **ATTENTION:** Devices outside of the scanner's allowable I/O image space will be rejected and will not be included in the scan list.

The AutoScan feature is automatically disabled in the scanner as soon as a scanner property is modified by RSNetWorx for DeviceNet software. For example, any manual changes to the scan list using RSNetWorx for DeviceNet software disables the AutoScan feature in the scanner.

One new status code has been added to the Node Status list. This code is presented in the Node Status Table.

Status Code (Decimal)	Description of Status
65	AutoScan Active (Scanner only status)

When the scanner is in Run mode with AutoScan enabled, the scanner display alternates between 65 and the scanner node address.

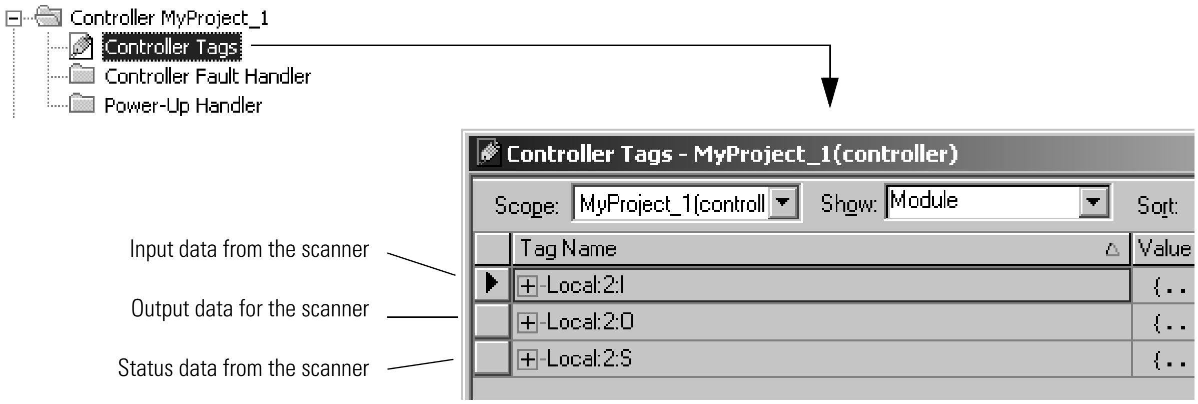
When a scanner is transitioned from Run mode to Idle mode while AutoScan is enabled, it only scans the network for nodes that are not already in the scan list. However, while in Idle mode, an AutoScan DISABLE to ENABLE transition causes the scanner to erase the existing scan list and scan for all nodes on the network.

The AutoScan feature enables AAR (Auto-Address Recovery) for each of the configured slave devices.

The AutoScan feature checks for the Quick Connect setting in each slave device and enables Quick Connect in the scanner if it is enabled in the slave devices.

Access Device Data

When you add the scanner to the I/O configuration of the controller, RSLogix 5000 software automatically creates a set of tags for the input, output, and status data of the network.



Controller MyProject_1
Controller Tags
Controller Fault Handler
Power-Up Handler

Controller Tags - MyProject_1(controller)

Scope: MyProject_1(controll Show: Module Sort:

Tag Name	Value
+-Local:2:I	{..
+-Local:2:O	{..
+-Local:2:S	{..

Input data from the scanner

Output data for the scanner

Status data from the scanner

The tags for your DeviceNet data follow this format.

location	:type	.Data	[dnet_address]	.bit

▨ = Optional

Where	Is		
location	location of the scanner in the system		
	If you have this scanner	In a	Then location is
	ControlLogix 1756-DNB	local chassis	Local:slot_number_of_scanner
		remote chassis	adapter:slot_number_of_scanner where: adapter is the name of the EtherNet/IP or ControlNet module in the remote chassis.
type	type of data:		
	Where	Is	
	input from a device	I	
	output to a device	O	
dnet_address	address of the device on the DeviceNet network (based on only 4 bytes per node)		
bit	specific bit within the data of the device		

While you can use the input and output tags of the scanner directly in your logic, it is a lot easier to use alias tags. Alias tags can be used whether you use AutoScan or not to configure the scanner.

Put the Scanner in Run Mode

To run the DeviceNet network, follow these steps.

1. Place the controller in Run or Remote Run mode.
2. Set the following bit of the output structure for the scanner.

To put the scanner in Run mode, turn on this bit.

If you want to	The set this bit	To
Run the network	…O.CommandRegister.Run	1
Not run the network (Idle mode)	…O.CommandRegister.Run	0
Fault the network	…O.CommandRegister.Fault	1
Not fault the network	…O.CommandRegister.Fault	0
Disable the network	…O.CommandRegister.DisableNetwork	1
Enable the network	…O.CommandRegister.DisableNetwork	0
Halt the scanner (ceases all operation)	…O.CommandRegister.HaltScanner	1
Unhalt the scanner	…O.CommandRegister.HaltScanner	0
Reset the scanner	…O.CommandRegister.Reset	1
Resume operation after a reset	…O.CommandRegister.Reset	0

3. Check the scanner for Run mode.

If you have this scanner	Then this indicator	Displays
ControlLogix 1756-DNB	4-character display	RUN
CompactLogix 1769-SDN	2-character display	its node number when in Run mode

Additional Information About AutoScan

Type of Connection that the Scanner Sets Up

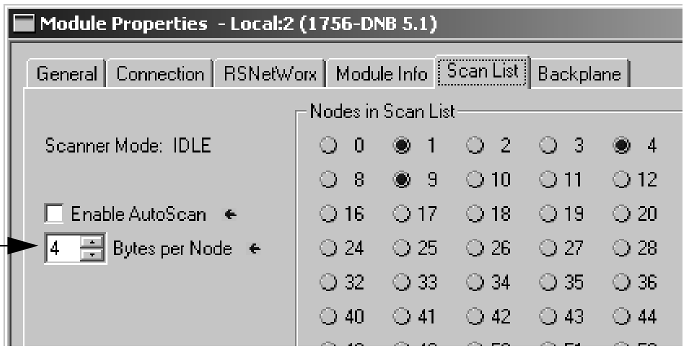
The type of update (connection) that the scanner sets up with each device depends on the device. The scanner chooses the first connection type that the device supports in this order:

1. Change-of-state (COS)
2. Polled
3. Strobed
4. Cyclic at 1000 ms

The scanner tries to set up a change-of-state connection. If the device does not support change-of-state, then the scanner tries to set up a polled connection and so on. The type of connection that the scanner sets up may not be the default for the device.

Allocating More Memory for Each Device

The AutoScan feature is easiest to use if you leave it set to 1 DINT (4 bytes) of input memory and output memory for each address.



Scanner Mode: IDLE

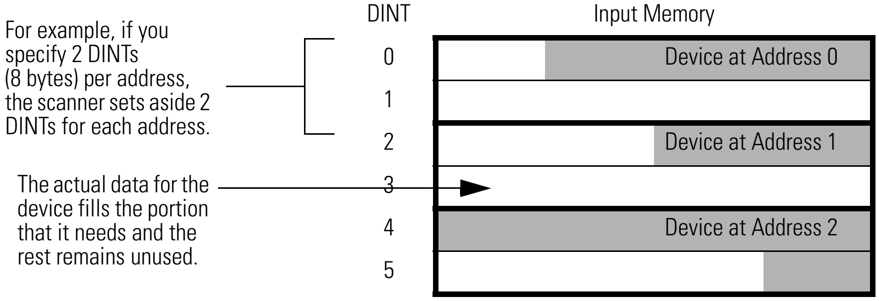
☐ Enable AutoScan

4 Bytes per Node

○ 0	● 1	○ 2	○ 3	● 4
○ 8	● 9	○ 10	○ 11	○ 12
○ 16	○ 17	○ 18	○ 19	○ 20
○ 24	○ 25	○ 26	○ 27	○ 28
○ 32	○ 33	○ 34	○ 35	○ 36
○ 40	○ 41	○ 42	○ 43	○ 44

Number of bytes of input memory and output memory to allocate in for each address

As an option, you can allocate more memory for each device.

Consideration	Description
The bytes/node value defines the amount of memory for each address.	AutoScan lets you specify how much input and output memory to give to each address on the network.  For example, if you specify 2 DINTs (8 bytes) per address, the scanner sets aside 2 DINTs for each address. The actual data for the device fills the portion that it needs and the rest remains unused. DINT Input Memory 0 1 2 3 4 5 Device at Address 0 Device at Address 1 Device at Address 2
The scanner sets-up communication with any device that fits within the allocated memory size.	The scanner automatically sets up communication with those devices that fit within the memory allocated for each address. • For example, if you allocate 2 DINTs (8 bytes) per address, the scanner sets up communication with any device that sends or receives 1…8 bytes of data. • The scanner adds as many device as it can until it runs out of memory. • If you give too much memory to each address, you may not have enough memory for all your devices.
The scanner skips devices that are too large.	If a device needs more memory than is allocated, the scanner skips it and does not set up communication with it. For example, if you specify 2 DINTs (8 bytes) per address but a device sends 9 bytes, the scanner does not add the device to the scan list.
Manually editing the scan list turns off AutoScan.	If you use RSNetWorx for DeviceNet software to edit the configuration of the scanner, the scanner turns off AutoScan. Do not turn it back on or you will clear the configuration that you just entered. For example, if you use RSNetWorx for DeviceNet software to manually add a device to the scan list, the scanner turns off AutoScan. If turn on AutoScan again, the scanner clears it current configuration and starts over.

Control a Device

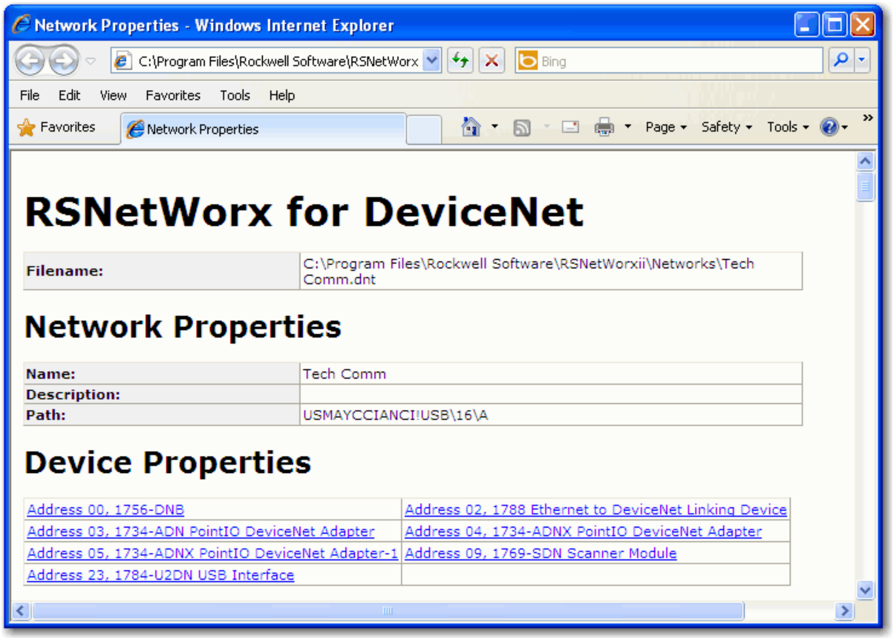
Use this chapter to develop the logic that examines and controls your devices.

Before You Begin

Before you use this chapter, get the following information:

- RSNetWorx Report for the Network
- Data Map for Each of Your Devices

RSNetWorx Report for the Network


Network Properties - Windows Internet Explorer
C:\Program Files\Rockwell Software\RSNetWorx
Bing
File Edit View Favorites Tools Help
Favorites
Network Properties
Page Safety Tools

RSNetWorx for DeviceNet

Filename:	C:\Program Files\Rockwell Software\RSNetWorxii\Networks\Tech Comm.dnt

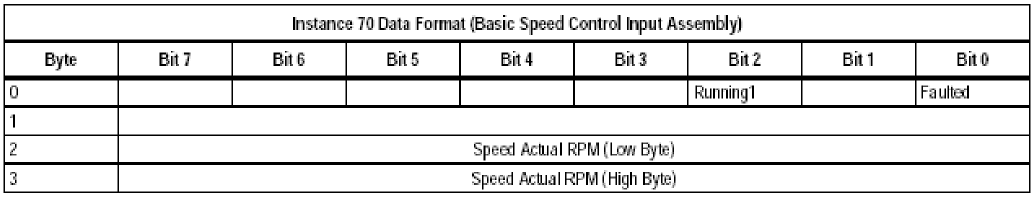
Network Properties

Name:	Tech Comm
Description:	
Path:	USMAYCCIANCI!USB\16\A

Device Properties

Address 00, 1756-DNB	Address 02, 1788 Ethernet to DeviceNet Linking Device
Address 03, 1734-ADN PointIO DeviceNet Adapter	Address 04, 1734-ADNX PointIO DeviceNet Adapter
Address 05, 1734-ADNX PointIO DeviceNet Adapter-1	Address 09, 1769-SDN Scanner Module
Address 23, 1784-U2DN USB Interface	

Data Map for Each of Your Devices



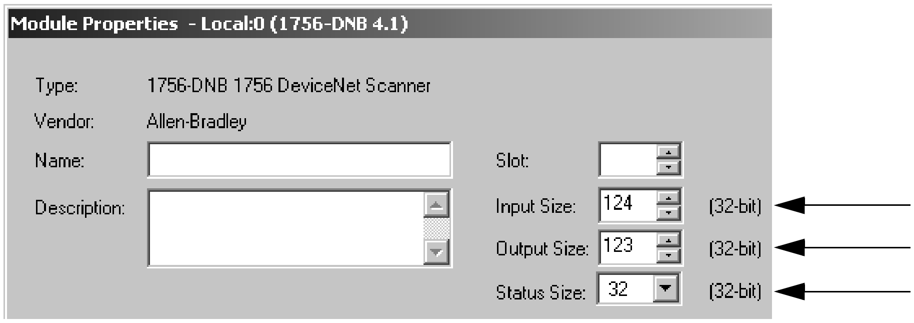
Instance 70 Data Format (Basic Speed Control Input Assembly)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0						Running1		Faulted
1								
2	Speed Actual RPM (Low Byte)							
3	Speed Actual RPM (High Byte)							

Add the Scanner to the Controller's I/O Configuration

To access the data of the network, add the scanner to the controller's I/O configuration in RSLogix 5000 programming software. However, you may need to conserve bandwidth on the EtherNet/IP or ControlNet network.

Conserve EtherNet/IP or ControlNet Network Bandwidth

The default configuration of the scanner gives you the maximum amount of input, output, and status data, as shown in the following graphic.



Module Properties - Local:0 (1756-DNB 4.1)

Type: 1756-DNB 1756 DeviceNet Scanner
Vendor: Allen-Bradley
Name:
Description:
Slot:
Input Size: 124 (32-bit)
Output Size: 123 (32-bit)
Status Size: 32 (32-bit)

If the scanner communicates with the controller via an EtherNet/IP or ControlNet network and you need to conserve bandwidth over that network, consider reducing the input, output, or status sizes.

- Set the input and output sizes = the number of input and output DINTs in the scanner that actually store device data.
- If you are **not** going to use all the status information, set the status size to the minimum required.

EXAMPLE	If you want to use **only** the ASCII representation of scanner status/display, set the status size to 10. If you also want to read the status code of the scanner, set the status size to 11.

See Table 1 - Set the Status Size for a Scanner on page 106 for more information on how to change the status size of a scanner from the default values.

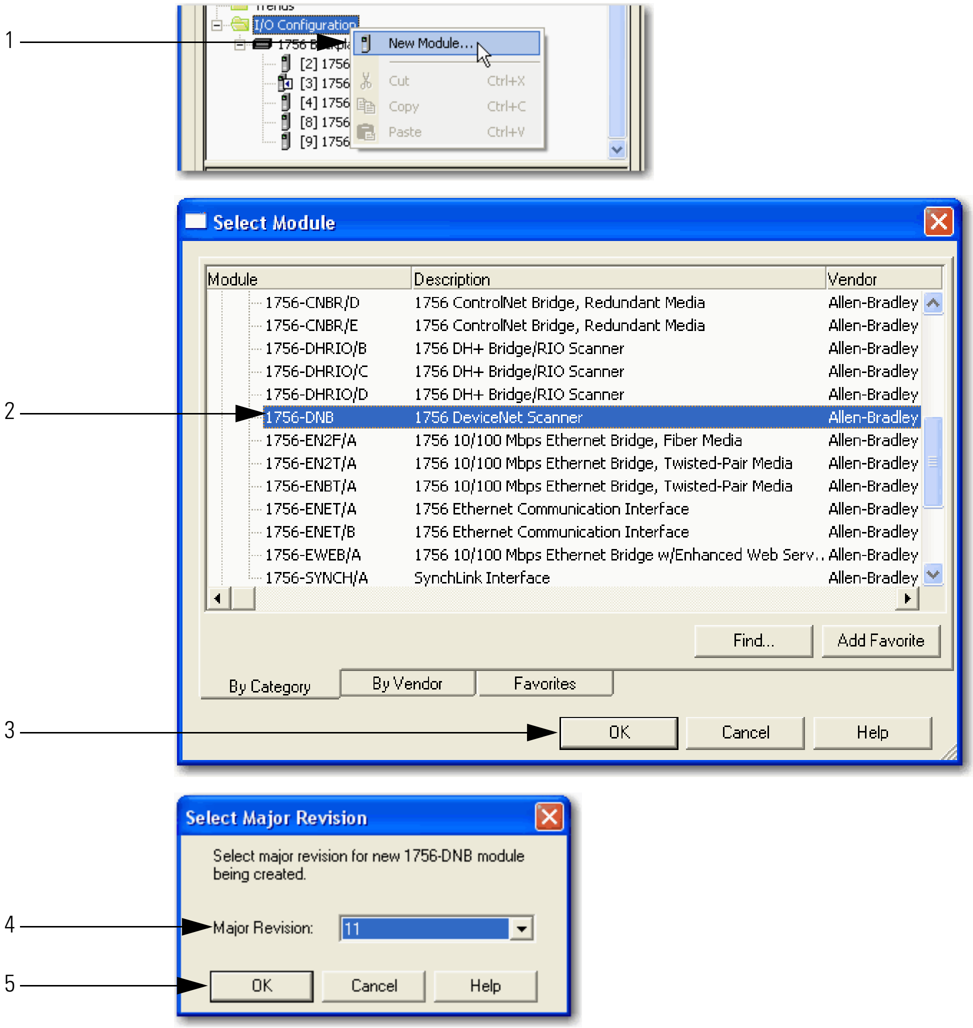
Table 1 - Set the Status Size for a Scanner

If you want the following information	Set the status size (DINTs) to the following value	This setting gives you the following parameter values.	
		Member	**Data Type**
Count of I/O scans	10	ScanCounter	DINT
Indication that a device has failed: • There is 1 bit for each address on the DeviceNet network (0 - 63). • The position of a bit = address of a device. • If a bit = 1, then the device at that address has failed.		DeviceFailureRegister	SINT[8]
Indication that the data size of a device does not match the amount of memory allocated for the device in the scanner: • There is 1 bit for each address on the DeviceNet network (0 - 63). • The position of a bit = address of a device. • If a bit = 1, then their is a mismatch with that address.		AutoverifyFailureRegister	SINT[8]
Indication that a device is idle: • There is 1 bit for each address on the DeviceNet network (0 - 63). • The position of a bit = address of a device. • If a bit = 1, then the device at that address is idle.		DeviceIdleRegister	SINT[8]
Indication that a device is online: • There is 1 bit for each address on the DeviceNet network (0 - 63). • The position of a bit = address of a device. • If a bit = 1, then the device at that address is online.		ActiveNodeRegister	SINT[8]
ASCII representation of scanner status/display		StatusDisplay	SINT[4]
Address of the scanner	11	ScannerAddress	SINT
Status code of scanner		ScannerStatus	SINT
Address with an error: • Scrolls through the addresses with errors • ScrollingDeviceStatus member shows the status code		ScrollingDeviceAddress	SINT
Status code of an address with an error: • Scrolls through addresses with errors • ScrollingDeviceAddress member shows the address		ScrollingDeviceStatus	SINT
Possible future expansion of the structure—5 DINTs	16		
Status code of lower 32 devices—1 byte per device	24	DeviceStatus	SINT[32]
Status code of all devices—1 byte per device	32	DeviceStatus	SINT[64]

Add the Scanner to the I/O Configuration Folder

Complete the following steps to add the scanner to the I/O configuration file.

1. Right-click and choose New Module.
2. Select the type of scanner.
3. Click OK.
4. From the Major Revision pull-down menu, choose a major revision number for the scanner.
5. Click OK.



Module	Description	Vendor
1756-CNBR/D	1756 ControlNet Bridge, Redundant Media	Allen-Bradley
1756-CNBR/E	1756 ControlNet Bridge, Redundant Media	Allen-Bradley
1756-DHRIO/B	1756 DH+ Bridge/RIO Scanner	Allen-Bradley
1756-DHRIO/C	1756 DH+ Bridge/RIO Scanner	Allen-Bradley
1756-DHRIO/D	1756 DH+ Bridge/RIO Scanner	Allen-Bradley
1756-DNB	1756 DeviceNet Scanner	Allen-Bradley
1756-EN2F/A	1756 10/100 Mbps Ethernet Bridge, Fiber Media	Allen-Bradley
1756-EN2T/A	1756 10/100 Mbps Ethernet Bridge, Twisted-Pair Media	Allen-Bradley
1756-ENBT/A	1756 10/100 Mbps Ethernet Bridge, Twisted-Pair Media	Allen-Bradley
1756-ENET/A	1756 Ethernet Communication Interface	Allen-Bradley
1756-ENET/B	1756 Ethernet Communication Interface	Allen-Bradley
1756-EWEB/A	1756 10/100 Mbps Ethernet Bridge w/Enhanced Web Serv...	Allen-Bradley
1756-SYNCH/A	SynchLink Interface	Allen-Bradley

Select major revision for new 1756-DNB module being created.

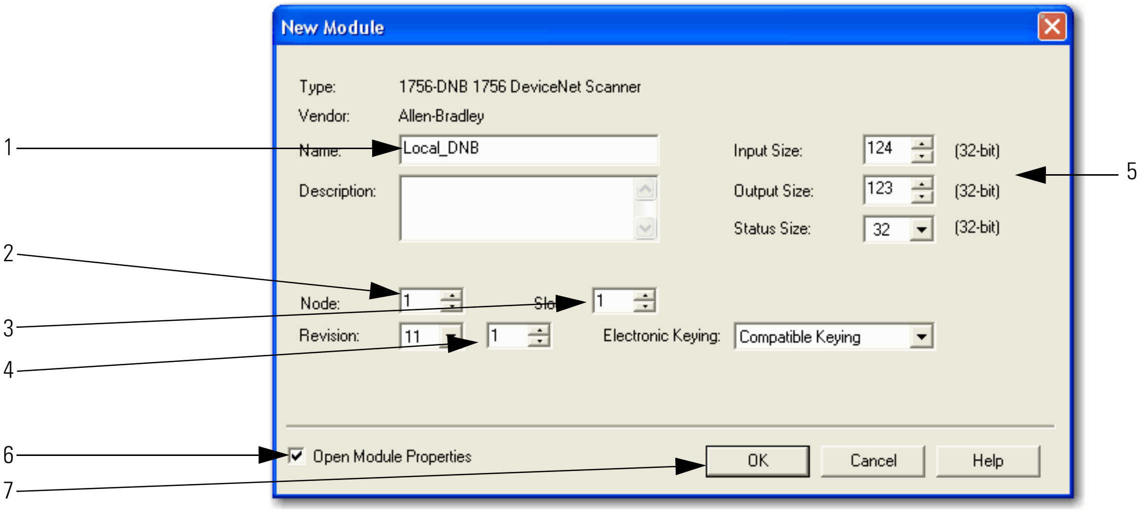
Major Revision: 11

Configure the Scanner

Complete the following steps to configure the scanner.

1. Type a name for the scanner.
2. Enter a node number.
3. Enter the slot number.
4. Enter the minor revision.
5. Enter the size of the input and output memory maps that the scanner will allocate for each device it detects on the network.

   Valid values range from 0…32 bytes per node.
6. If you need to make additional configuration changes, such as setting the requested packet interval (RPI), check Open Module Properties.
7. Click OK.



New Module

Type: 1756-DNB 1756 DeviceNet Scanner

Vendor: Allen-Bradley

Name: Local_DNB

Description:

Input Size: 124 (32-bit)

Output Size: 123 (32-bit)

Status Size: 32 (32-bit)

Node: 1

Slot: 1

Revision: 11 1

Electronic Keying: Compatible Keying

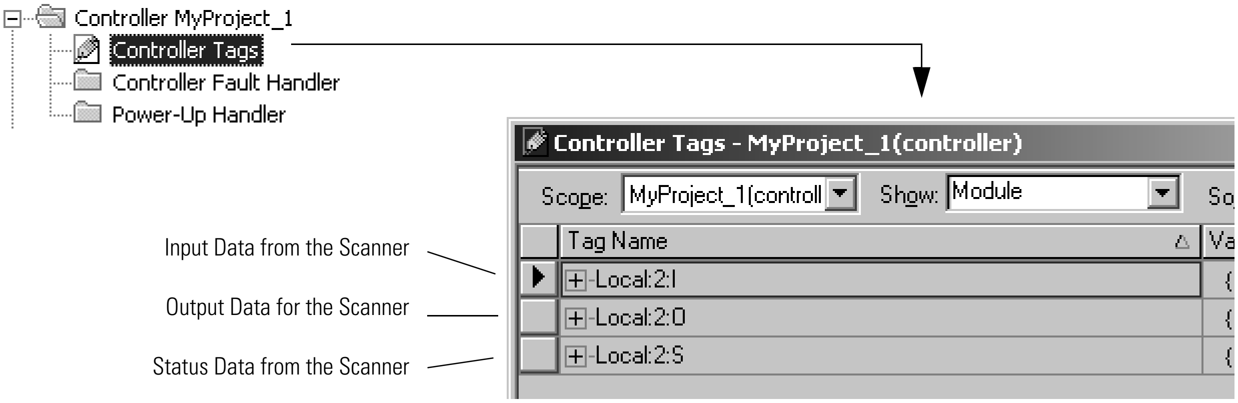
☑ Open Module Properties

8. If the Module Properties dialog box appears, make additional configuration changes.

   You can change scanner configuration on the following tabs:
   - General
   - Connection
   - RSNetWorx

Determine the Address of DeviceNet Data

When you add the scanner to the I/O configuration of the controller, RSLogix 5000 programming software automatically creates a set of tags for the input, output, and status data of the network.



Input Data from the Scanner

Output Data for the Scanner

Status Data from the Scanner

The tags for your DeviceNet data follow this format.

The scanner memory uses this format...

slot	type	.Data	[element]	.bit

which is this tag in the controller.

location	:type	.Data	[element]	.bit

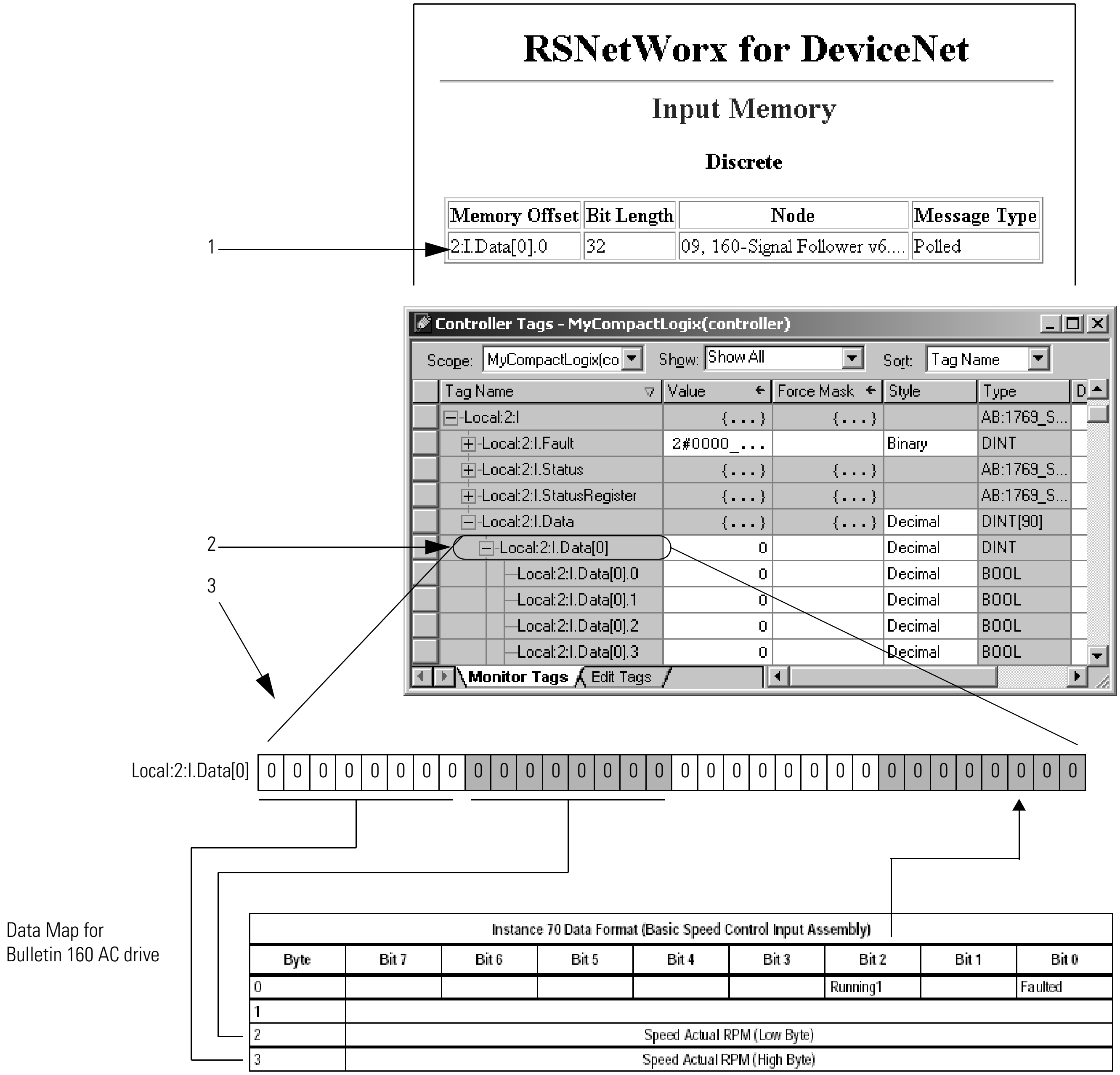
▨ = Optional

Where	Is	
Slot	The slot number of the scanner	
Location	**If you have this scanner**	**Then location is**
	Local ControlLogix 1756-DNB	Local:slot_number_of_scanner
	Remote ControlLogix 1756-DNB	name_of_remote_bridge:slot_number_of_scanner
	CompactLogix 1769-SDN	Local:slot_number_of_scanner
	SoftLogix 5800 1784-PCIDS	Local:slot_number_of_scanner
	Linking Device 1788-EN2DN or 1788-CN2DN	The name of the linking device in the I/O configuration of the controller
Type	**If the data is**	**Then type is**
	Input from a device	I
	Output to a device	O
	The status of the network	S
Element	A specific DINT (DWord, 32-bit integer) within the array	
Bit	A specific bit within an integer	

Complete the following steps to determine the tag name, or address, for DeviceNet data.

1. On the RSNetWorx report for the network, find the memory address for the input or output data of the device.
2. Find the corresponding tag in the controller-scoped tags of the controller.
3. Find the required data within the controller tag.

   Use the data map for the device as a reference.



RSNetWorx for DeviceNet

Input Memory

Discrete

Memory Offset	Bit Length	Node	Message Type
2:I.Data[0].0	32	09, 160-Signal Follower v6....	Polled

Controller Tags - MyCompactLogix(controller)

Scope: MyCompactLogix(co Show: Show All Sort: Tag Name

Tag Name	Value	Force Mask	Style	Type
−Local:2:I	{...}	{...}		AB:1769_S...
+Local:2:I.Fault	2#0000_...		Binary	DINT
+Local:2:I.Status	{...}	{...}		AB:1769_S...
+Local:2:I.StatusRegister	{...}	{...}		AB:1769_S...
−Local:2:I.Data	{...}	{...}	Decimal	DINT[90]
−Local:2:I.Data[0]	0		Decimal	DINT
Local:2:I.Data[0].0	0		Decimal	BOOL
Local:2:I.Data[0].1	0		Decimal	BOOL
Local:2:I.Data[0].2	0		Decimal	BOOL
Local:2:I.Data[0].3	0		Decimal	BOOL

Monitor Tags / Edit Tags

Local:2:I.Data[0] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Instance 70 Data Format (Basic Speed Control Input Assembly)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0						Running1		Faulted
1								
2	Speed Actual RPM (Low Byte)							
3	Speed Actual RPM (High Byte)							

SoftLogix 5800 Controller

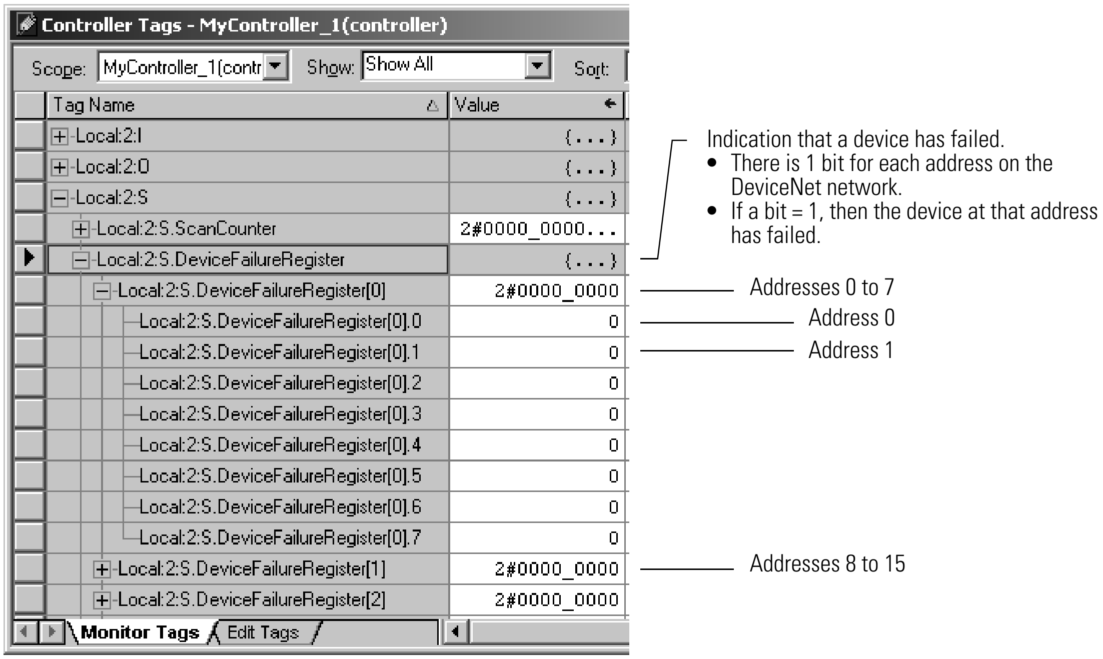
The SoftLogix 5800 scanner 1784-PCIDS organizes input and output memory in 16-bit words. It uses address format word.bit.

Where	Is
Word	INT (16-bit integer) with the memory of the scanner
Bit	A specific bit within an integer

While you can use the input and output tags of the scanner directly in your logic, it is easier to use alias tags.

Determine If a Device Has Failed

If a DeviceNet communication device stops communicating, such as because of a device failure, the tag for the device stays at its last value. To make sure that your input data is valid, we recommend that you buffer the input data and examine the device failure register.



Controller Tags - MyController_1(controller)

Scope: MyController_1(contr Show: Show All Sort:

Tag Name	Value
⊞-Local:2:I	{...}
⊞-Local:2:O	{...}
⊟-Local:2:S	{...}
⊞-Local:2:S.ScanCounter	2#0000_0000...
⊟-Local:2:S.DeviceFailureRegister	{...}
⊟-Local:2:S.DeviceFailureRegister[0]	2#0000_0000
Local:2:S.DeviceFailureRegister[0].0	0
Local:2:S.DeviceFailureRegister[0].1	0
Local:2:S.DeviceFailureRegister[0].2	0
Local:2:S.DeviceFailureRegister[0].3	0
Local:2:S.DeviceFailureRegister[0].4	0
Local:2:S.DeviceFailureRegister[0].5	0
Local:2:S.DeviceFailureRegister[0].6	0
Local:2:S.DeviceFailureRegister[0].7	0
⊞-Local:2:S.DeviceFailureRegister[1]	2#0000_0000
⊞-Local:2:S.DeviceFailureRegister[2]	2#0000_0000

Indication that a device has failed.

- There is 1 bit for each address on the DeviceNet network.
- If a bit = 1, then the device at that address has failed.

Addresses 0 to 7

Address 0

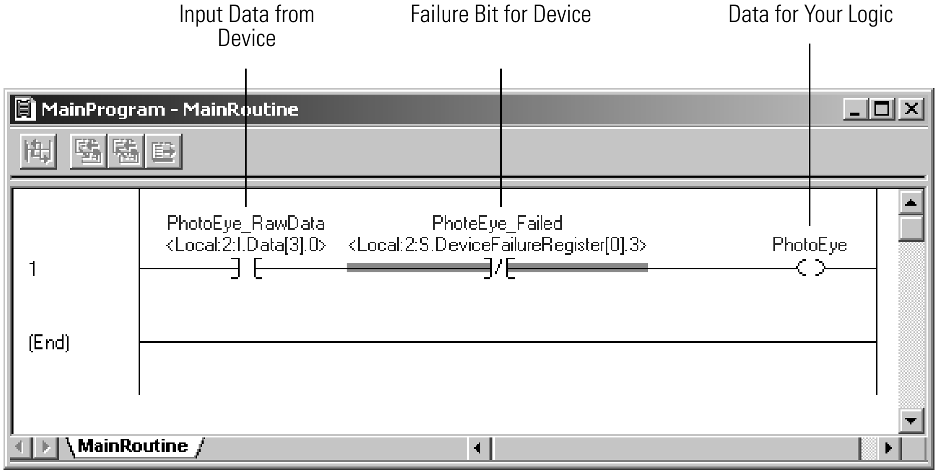
Address 1

Addresses 8 to 15

On every scan of the controller, execute logic similar to the following:

```
If PhotoEye_RawData = 1 and PhotoEye_Failed = 0 then
    PhoteEye = 1
Otherwise PhoteEye = 0
```

Use the PhotoEye tag in the rest of your logic (not PhotoEye_RawData).



Input Data from Device

Failure Bit for Device

Data for Your Logic

Place the Scanner in Run Mode

Complete the following steps to run the DeviceNet network.

1. Set the following bit of the output structure for the scanner.

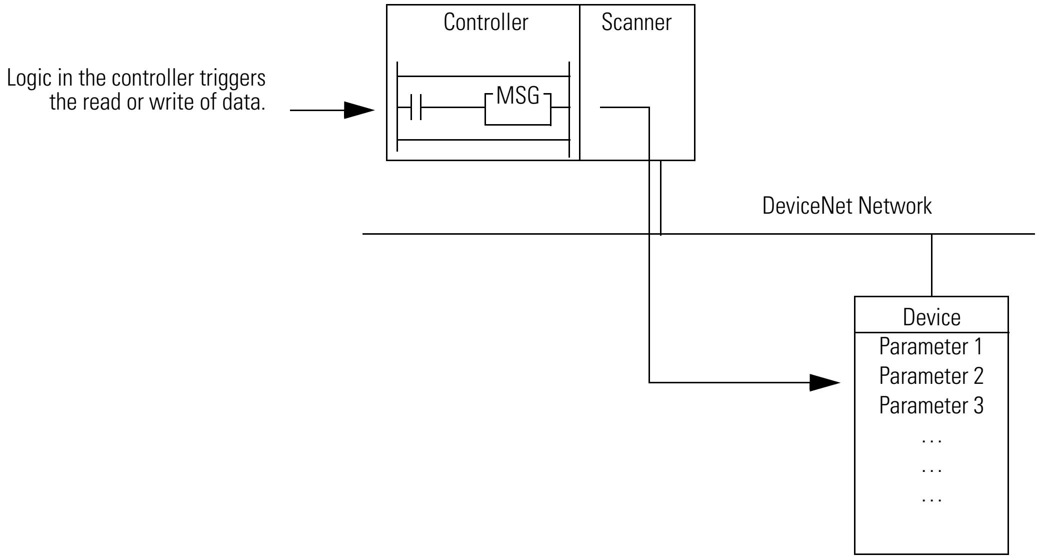
To put the scanner in Run mode, turn on this bit.

If you want to	Set this bit	To
Run the network	…O.CommandRegister.Run	1
Not run the network (idle mode)	…O.CommandRegister.Run	0
Fault the network	…O.CommandRegister.Fault	1
Not fault the network	…O.CommandRegister.Fault	0
Disable the network	…O.CommandRegister.DisableNetwork	1
Enable the network	…O.CommandRegister.DisableNetwork	0
Halt the scanner (ceases all operation)	…O.CommandRegister.HaltScanner	1
Unhalt the scanner	…O.CommandRegister.HaltScanner	0
Reset the scanner	…O.CommandRegister.Reset	1
Resume operation after a reset	…O.CommandRegister.Reset	0

2. Place the controller in Run or Remote Run mode.

When to Use a MSG Instruction

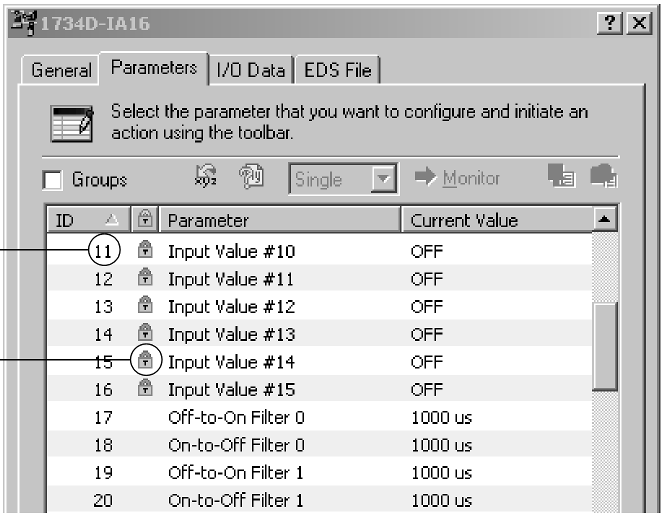
If you want to set or get a parameter based on conditions in your logic, use a Message (MSG) instruction in ladder logic to access the parameter.



Some parameters **do not** require ongoing updates. For example, initializing configuration parameters may occur only when the controller goes to Run mode. By using a MSG instruction for those parameters, you save bandwidth on the DeviceNet network for more critical or ongoing data.

Determine the Parameter Number to Access

In RSNetWorx for DeviceNet software, determine the parameter number that you want to access. Some parameters are locked, as shown in the following graphic.



Select the parameter that you want to configure and initiate an action using the toolbar.

ID	🔒	Parameter	Current Value
11	🔒	Input Value #10	OFF
12	🔒	Input Value #11	OFF
13	🔒	Input Value #12	OFF
14	🔒	Input Value #13	OFF
15	🔒	Input Value #14	OFF
16	🔒	Input Value #15	OFF
17		Off-to-On Filter 0	1000 us
18		On-to-Off Filter 0	1000 us
19		Off-to-On Filter 1	1000 us
20		On-to-Off Filter 1	1000 us

Parameter Number

This symbol means you can only get the value of the parameter (read-only).

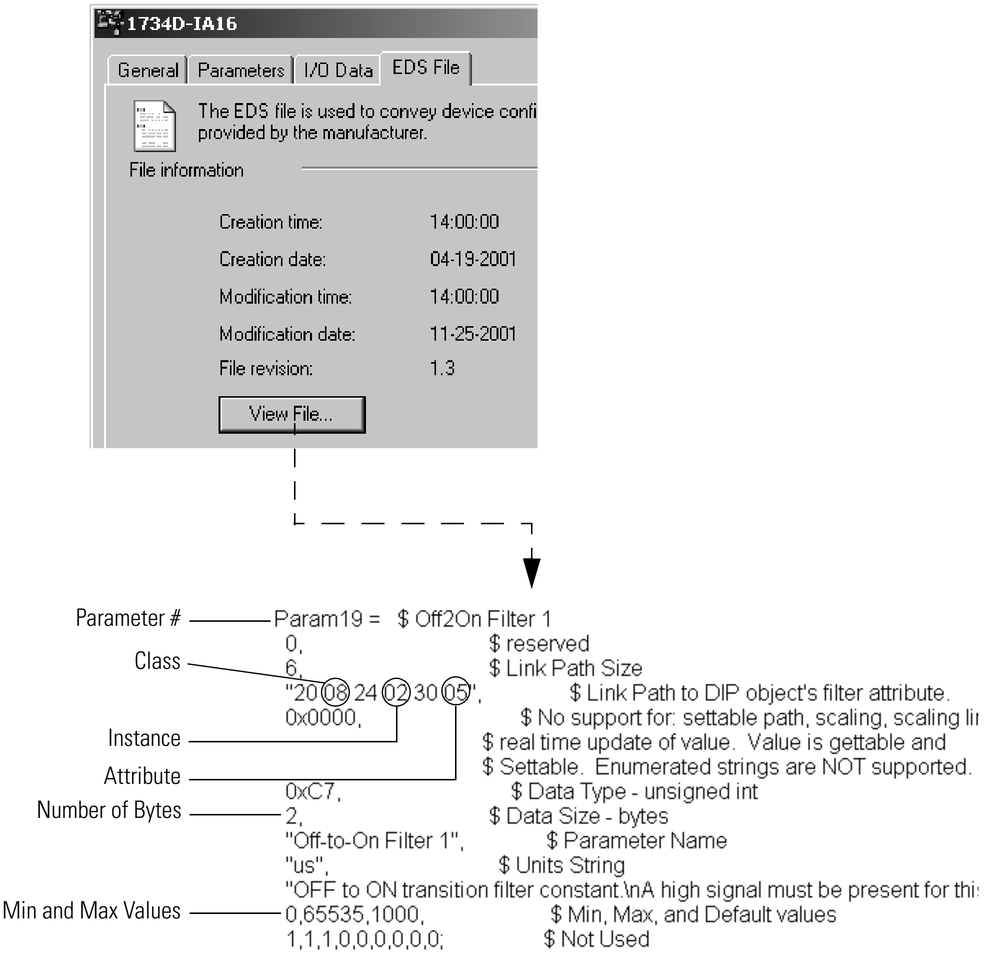
Determine the Configuration of the Parameter

To get or set a parameter, you must find the following information about it:

- Class # (hex)
- Instance # (hex)
- Attribute # (hex
- Number of bytes (size)
- Minimum value
- Maximum value
- Decimal places

Some devices assume a specific number of decimal places in a value.

In addition to the documentation for the device, the EDS file may also give you the required information.



1734D-IA16

The EDS file is used to convey device confi
provided by the manufacturer.

File information

Creation time:	14:00:00
Creation date:	04-19-2001
Modification time:	14:00:00
Modification date:	11-25-2001
File revision:	1.3

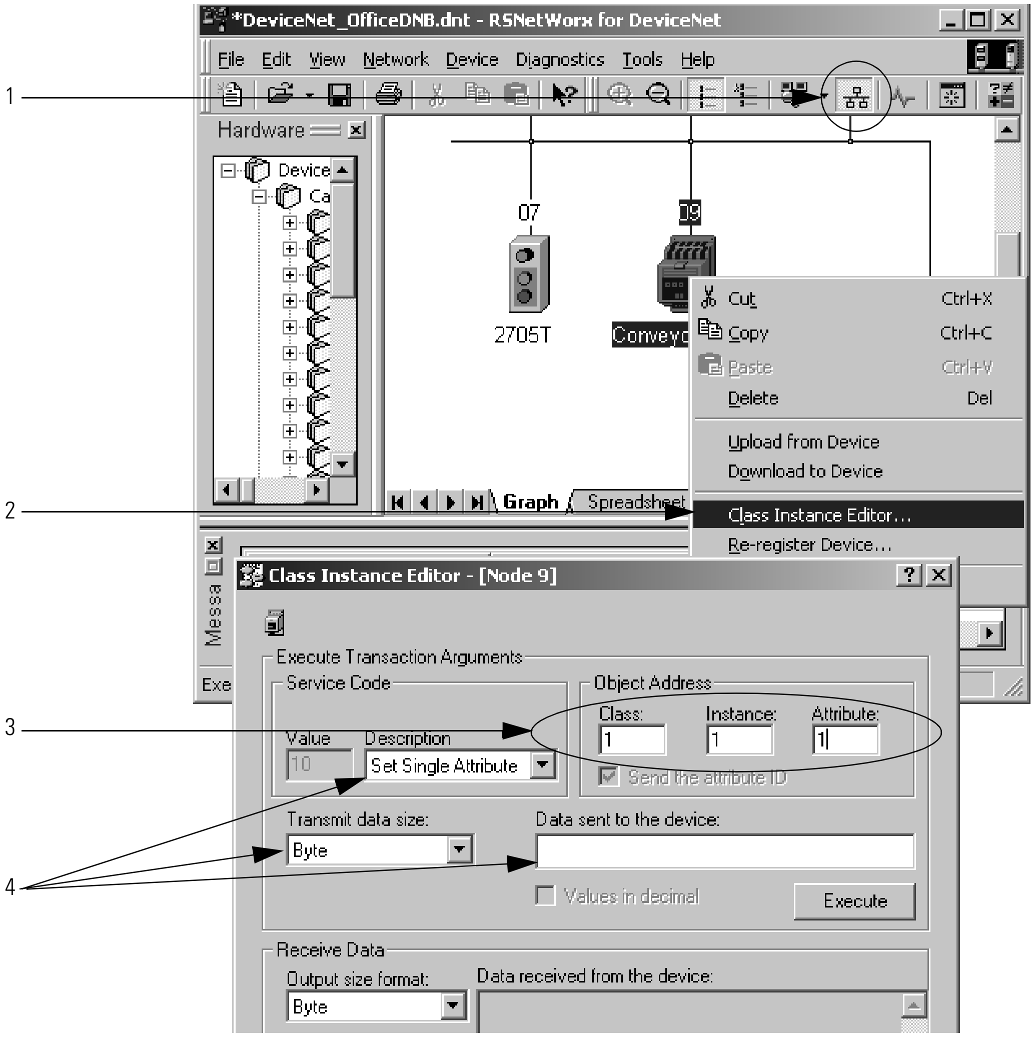
```
Param19 =  $ Off2On Filter 1
    0,                      $ reserved
    6,                      $ Link Path Size
    "20 08 24 02 30 05",            $ Link Path to DIP object's filter attribute.
    0x0000,                  $ No support for: settable path, scaling, scaling li
                          $ real time update of value.  Value is gettable and
                          $ Settable.  Enumerated strings are NOT supported.
    0xC7,                      $ Data Type - unsigned int
    2,                      $ Data Size - bytes
    "Off-to-On Filter 1",             $ Parameter Name
    "us",                    $ Units String
    "OFF to ON transition filter constant.\nA high signal must be present for thi
    0,65535,1000,               $ Min, Max, and Default values
    1,1,1,0,0,0,0,0,0;              $ Not Used
```

Test the Parameter

A simple way to make sure that you have the correct configuration for a parameter, such as data size or values, is to use the Class Instance editor in RSNetWorx for DeviceNet software.

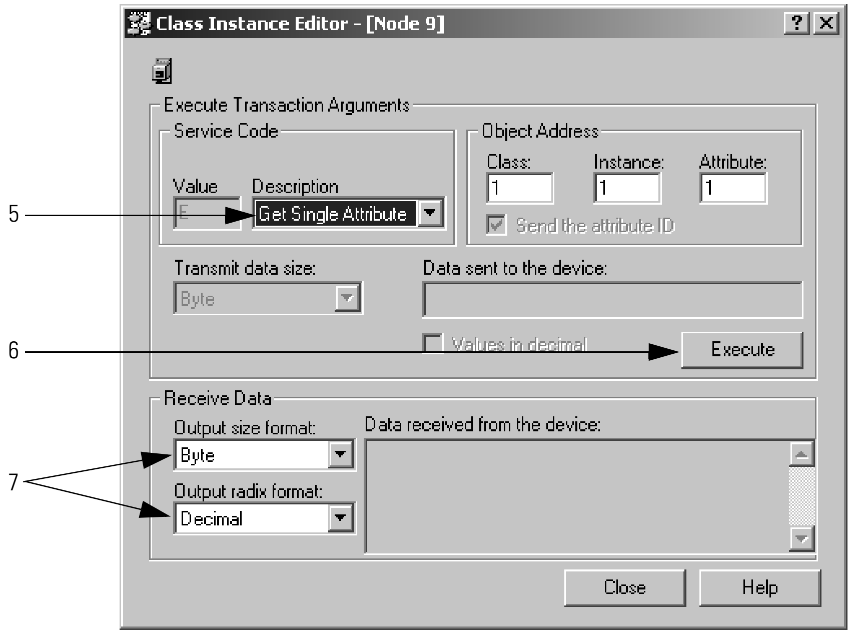
Complete the following steps to test the parameter.

1. In RSNetWorx for DeviceNet software, go online to the DeviceNet network.
2. Right-click the device and choose Class Instance Editor.
3. Type the class, instance, and attribute for the parameter.
4. Change the parameter.
   a. Choose Set Single Attribute.
   b. Choose the number of bytes.
   c. Type the new value in hexadecimal format.



Value: 10
Description: Set Single Attribute
Class: 1
Instance: 1
Attribute: 1
Transmit data size: Byte
Output size format: Byte

5. From the Description pull-down menu, choose Get Single Attribute to read the parameter.
6. Click Execute.
7. To change how output data is displayed, choose the size and format.


Class Instance Editor - [Node 9]

Execute Transaction Arguments

Service Code

Value: E

Description: Get Single Attribute

Object Address

Class: 1

Instance: 1

Attribute: 1

☑ Send the attribute ID

Transmit data size: Byte

Data sent to the device:

☐ Values in decimal

Receive Data

Output size format: Byte

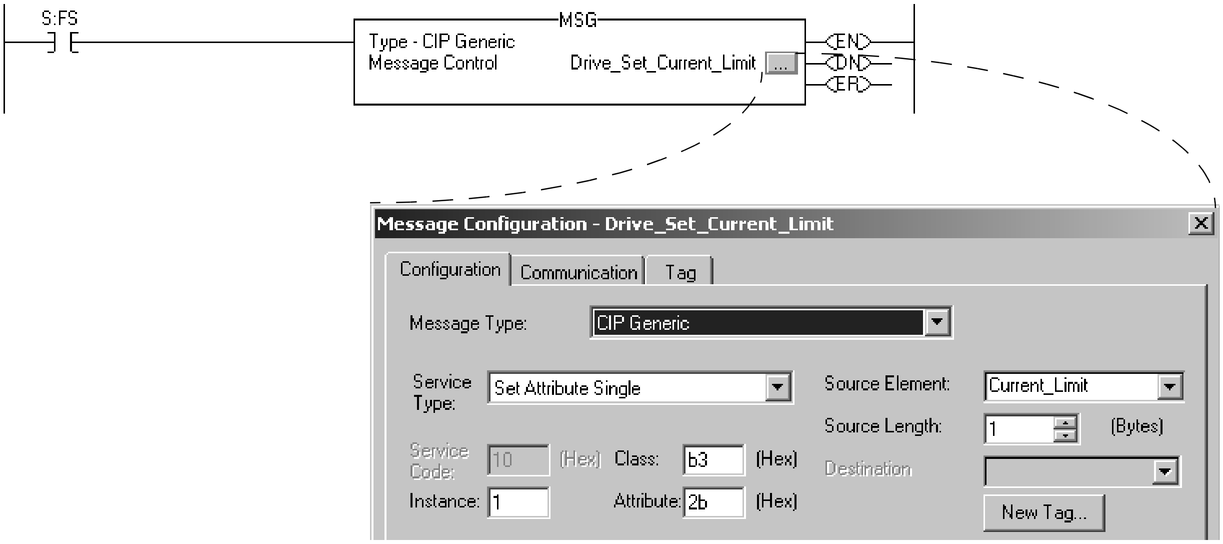
Output radix format: Decimal

Data received from the device:

Enter Message Logic

To access the parameter of a device (get or set the parameter), configure the MSG instruction as CIP Generic.

Change the current limit of the drive.


S:FS

MSG

Type - CIP Generic

Message Control: Drive_Set_Current_Limit

Message Configuration - Drive_Set_Current_Limit

Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex)

Class: b3 (Hex)

Instance: 1

Attribute: 2b (Hex)

Source Element: Current_Limit

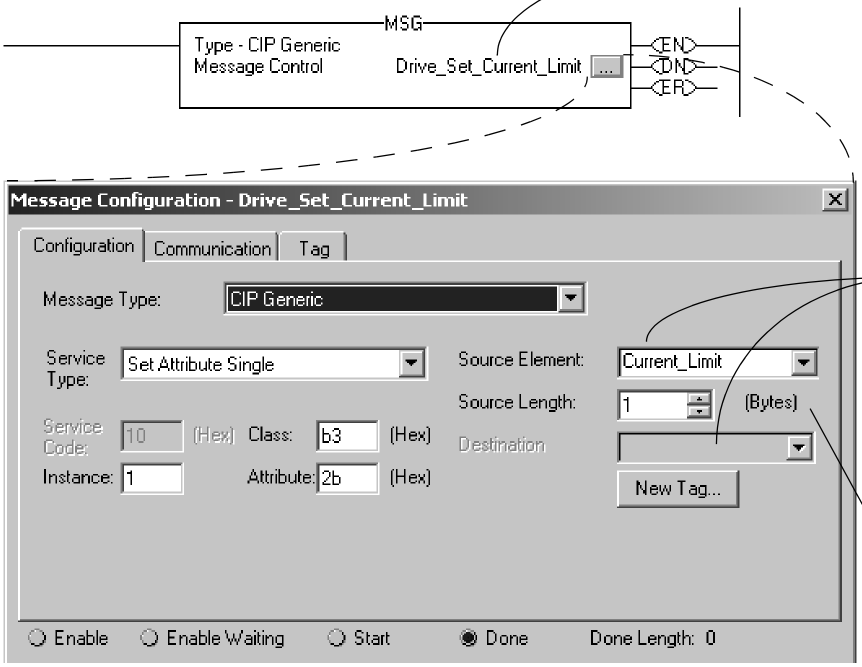
Source Length: 1 (Bytes)

Destination

You must complete the following tasks to configure the MSG instruction:

- Define the Source or Destination Data
- Enter and Configure the MSG Instruction
- Set the Communication Path

Define the Source or Destination Data



Message Configuration - Drive_Set_Current_Limit

Configuration | Communication | Tag

Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex) Class: b3 (Hex)

Instance: 1 Attribute: 2b (Hex)

Source Element: Current_Limit

Source Length: 1 (Bytes)

Destination:

New Tag...

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 0

Tag that controls the instruction.

- Scope—Controller.
- Data type—MESSAGE.
- The tag **cannot** be part of an array or a user-defined data type.

Source or destination for the data that the instruction sets or gets.

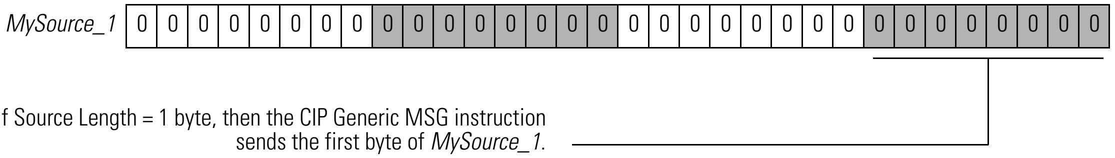
- Scope—Controller.
- Data type—In general, use the DINT data type, even when you set or get less than 4 bytes.
- Value—Make sure the source value stays within the minimum and maximum values for the parameter that you are setting.

Number of bytes (only if setting a value).

In general, follow these guidelines:

- Use the DINT data type for the source or destination tag, even when you set or get less than 4 bytes.
- Make sure the source value stays within the minimum and maximum values for the parameter that you are setting.

When setting a value, the CIP Generic MSG instruction takes only the specified number of bits from the source tag.



MySource_1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

For example, if Source Length = 1 byte, then the CIP Generic MSG instruction sends the first byte of *MySource_1*.

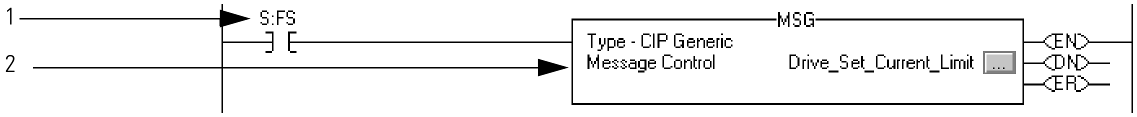
To increase the efficiency of your logic, minimize the use of SINT or INT data types. Whenever possible, use the DINT data type for integers.

- A Logix5000 controller typically compares or manipulates values as 32-bit values (DINTs or REALs).
- The controller typically converts a SINT or INT value to a DINT or REAL value before it uses the value.
- If the destination is a SINT or INT tag, the controller typically converts the value back to a SINT or INT value.
- The conversion to or from SINTs or INTs occurs automatically with no extra programming. However, it takes extra execution time and memory.

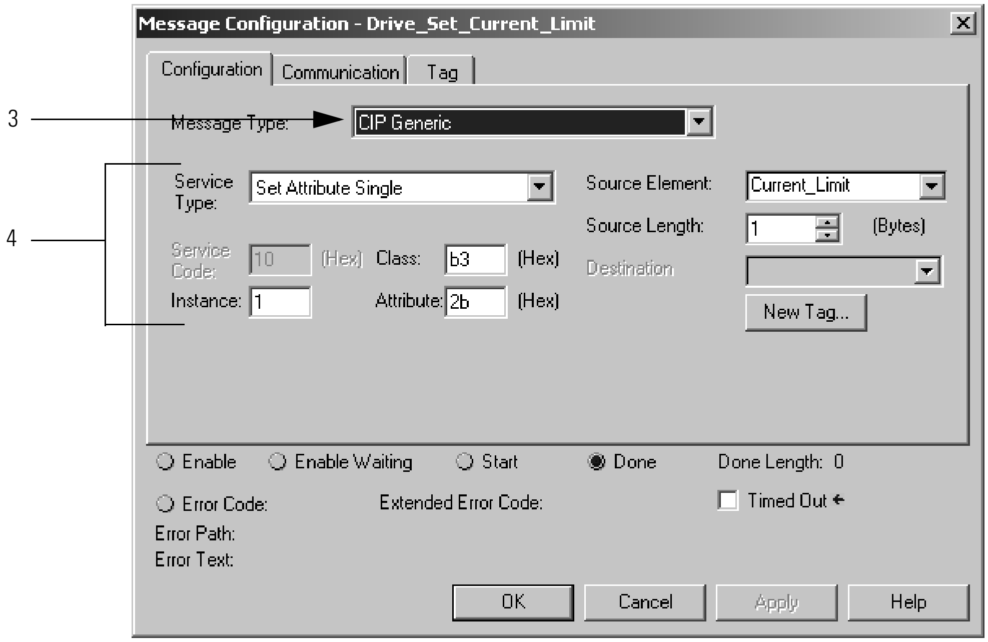
Enter and Configure the MSG Instruction

Complete the following steps to enter and configure the Message (MSG) instruction.

1. Enter the condition for the data transfer.
2. Enter the MSG instruction.



3. Choose CIP Generic.
4. To change a parameter:
   - Choose Set Attribute Single.
   - Choose the tag that has the new value.
   - Enter the number of bytes.
   - Type the class, instance, and attribute for the parameter in hexadecimal format.



Message Configuration - Drive_Set_Current_Limit

Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex)

Class: b3 (Hex)

Instance: 1

Attribute: 2b (Hex)

Source Element: Current_Limit

Source Length: 1 (Bytes)

Destination

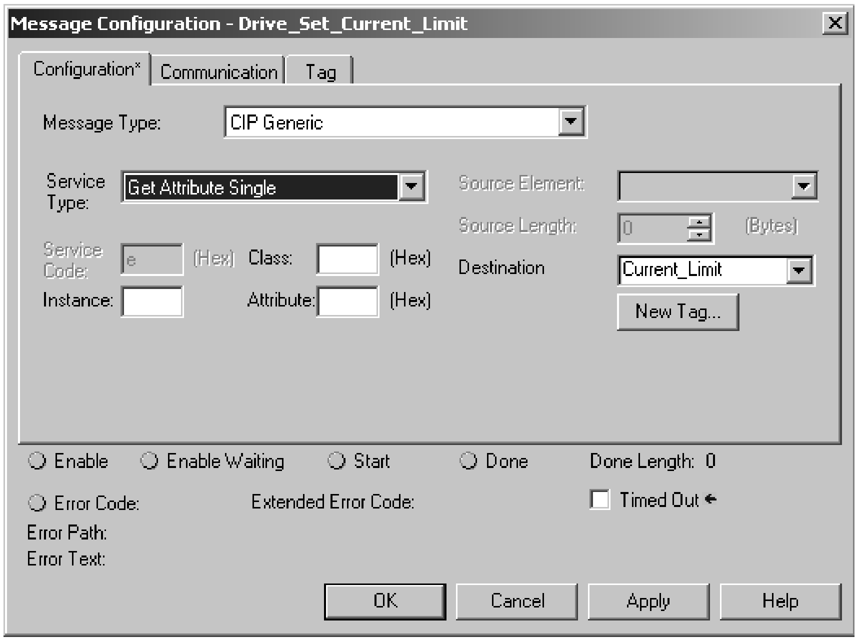
○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:

5. To read a parameter:
   - Choose Get Attribute Single from the Service Type pull-down menu.
   - Choose the tag to store the value from the Destination pull-down menu.
   - Type the class, instance, and attribute for the parameter in hexadecimal format.



Message Configuration - Drive_Set_Current_Limit

Message Type: CIP Generic

Service Type: Get Attribute Single

Service Code: e (Hex)

Class: (Hex)

Instance:

Attribute: (Hex)

Source Length: 0 (Bytes)

Destination: Current_Limit

Done Length: 0

Set the Communication Path

The communication path specifies the route to the device. A communication path follows this format:

scanner_name,2,device_address

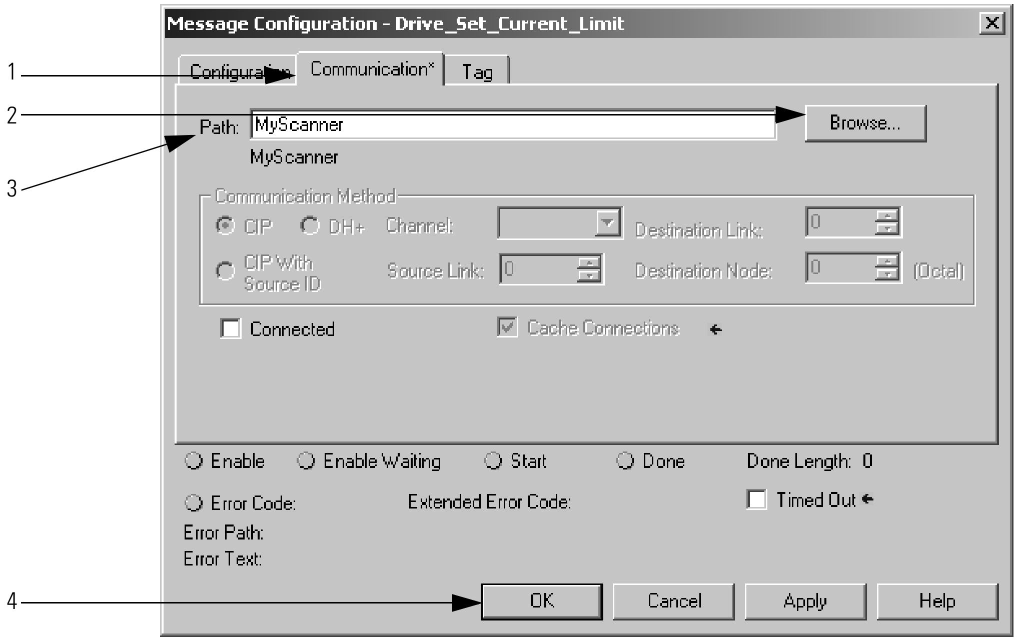
Where	Is
scanner_name	The name of the scanner in the I/O Configuration folder of the controller.
device_address	The address of the device on the DeviceNet network.

EXAMPLE If the name of the scanner is MyScanner and the device is at address 3, the path is as follows:

MyScanner,2,3

Complete the following steps to set the communication path.

1. Click the Communication tab.
2. Click the Browse button and select the scanner.
3. Type the rest of the path.
4. Click OK.



Path: MyScanner
MyScanner

● CIP ○ DH+ ○ CIP With Source ID

□ Connected ☑ Cache Connections

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:
Error Text:

Typically, a CIP Generic MSG instruction requires no connection to transfer its data.

This type of message	Using this communication method	Uses a connection
CIP data table read or write	CIP	Yes
PLC2, PLC3, PLC5, or SLC (all types)	CIP	No
	CIP with Source ID	No
	DH+	Yes
CIP Generic	CIP	Your choice[1]
Block-transfer read or write	N/A	Yes

[1] You can connect CIP generic messages, but for most applications we recommend you leave CIP Generic messages unconnected.

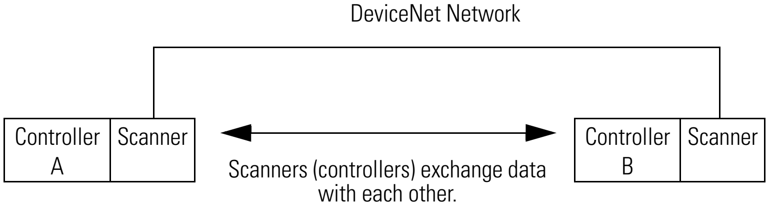
For more information on programming MSG instructions, see the Logix5000 Controller General Instructions Reference Manual, publication 1756-RM003.

Interlock and Share Inputs

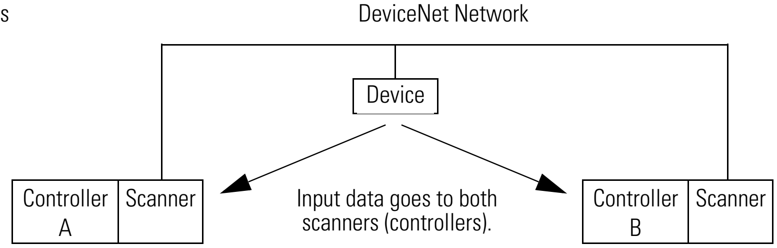
The chapter describes how to interlock and share inputs over a DeviceNet network.

Topic	Page
Interlock	124
Share Inputs	129

Interlocking



Sharing Inputs



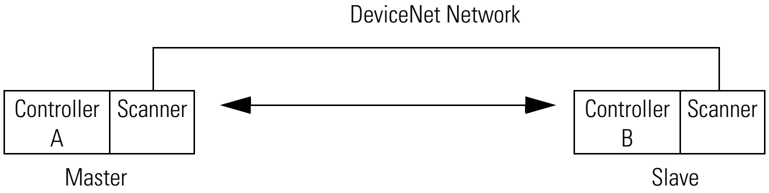
Interlock

To set up an interlock between two controllers over a DeviceNet network, complete the following tasks:

- Choose a Master Controller
- Determine How Much Data to Exchange
- Enable Slave Mode for the Slave Scanner
- Map the Slave Mode Data
- Add the Slave to the Master Scanner's Scan List
- Map the Data of the Slave
- Place Both Scanners In Run Mode

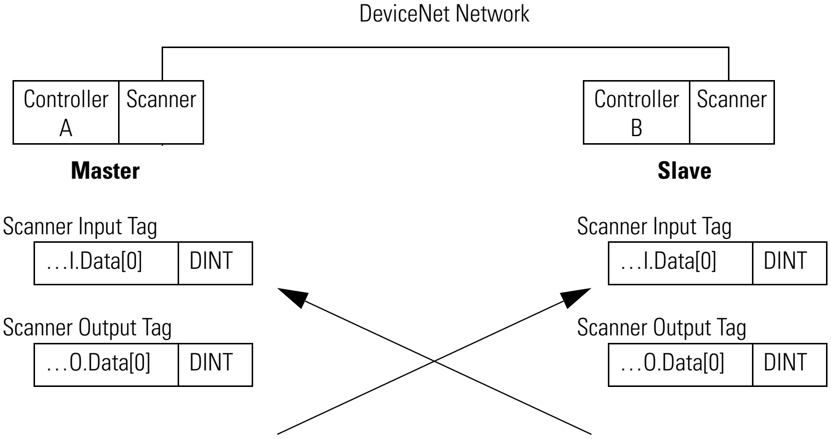
Choose a Master Controller

To interlock, choose a controller to serve as the master. The other controller becomes a slave to the master. This defines the relationship between the controllers. The scanners of each controller still scan and control their own devices, if desired.



Determine How Much Data to Exchange

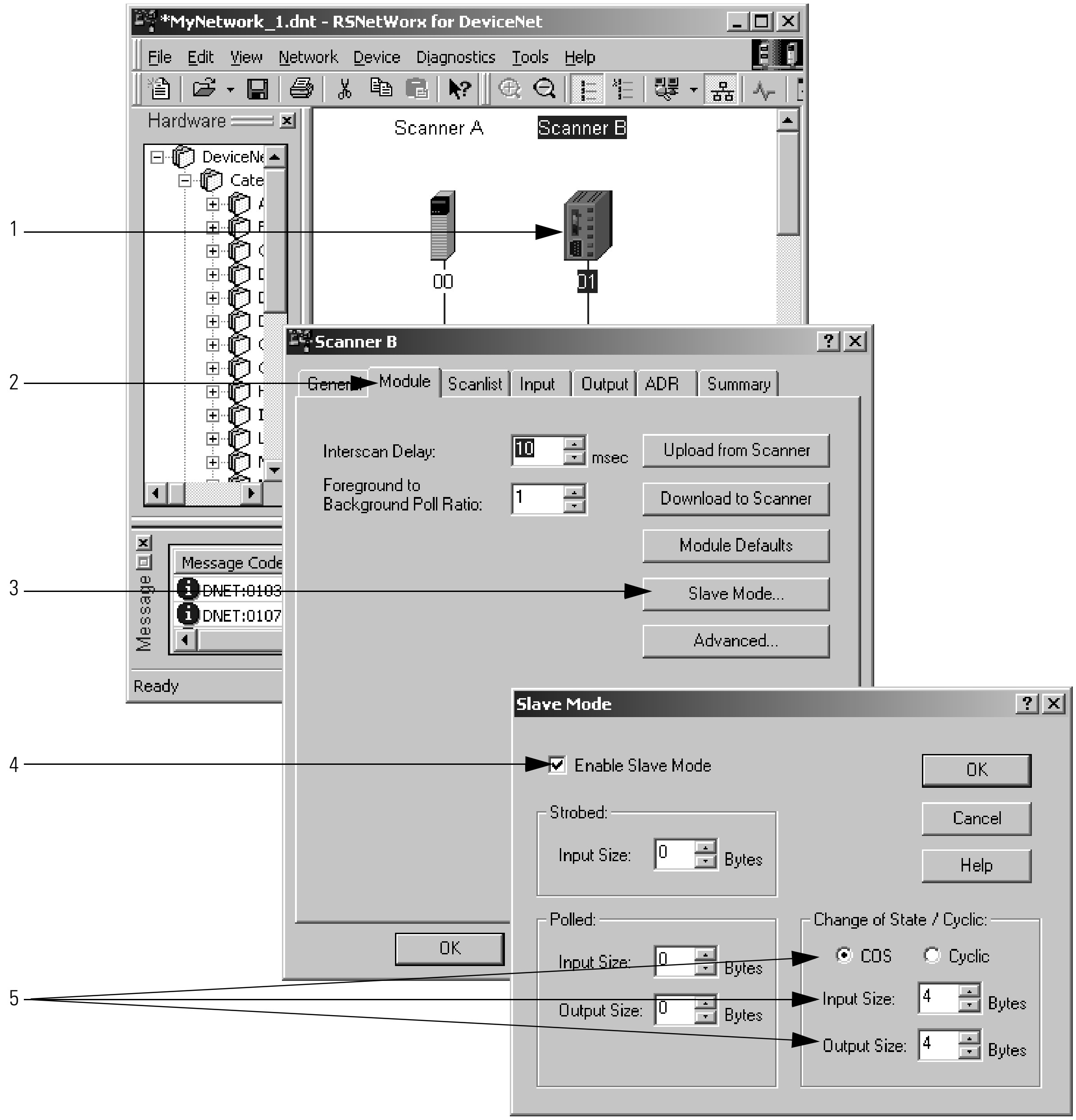
Before you configure the scanners for the interlock, determine how much data you want to exchange between the controllers.



Enable Slave Mode for the Slave Scanner

Complete the following steps to enable Slave mode.

1. In RSNetWorx for DeviceNet software, double-click the slave scanner to open its properties.
2. Click the Module tab.
3. Click Slave Mode.
4. Check the Enable Slave Mode check box.
5. Define the I/O parameters.



DNET:0103
DNET:0107

Interscan Delay: 10 msec

Foreground to Background Poll Ratio: 1

☑ Enable Slave Mode

Strobed:
Input Size: 0 Bytes

Polled:
Input Size: 0 Bytes
Output Size: 0 Bytes

Change of State / Cyclic:
● COS ○ Cyclic
Input Size: 4 Bytes
Output Size: 4 Bytes

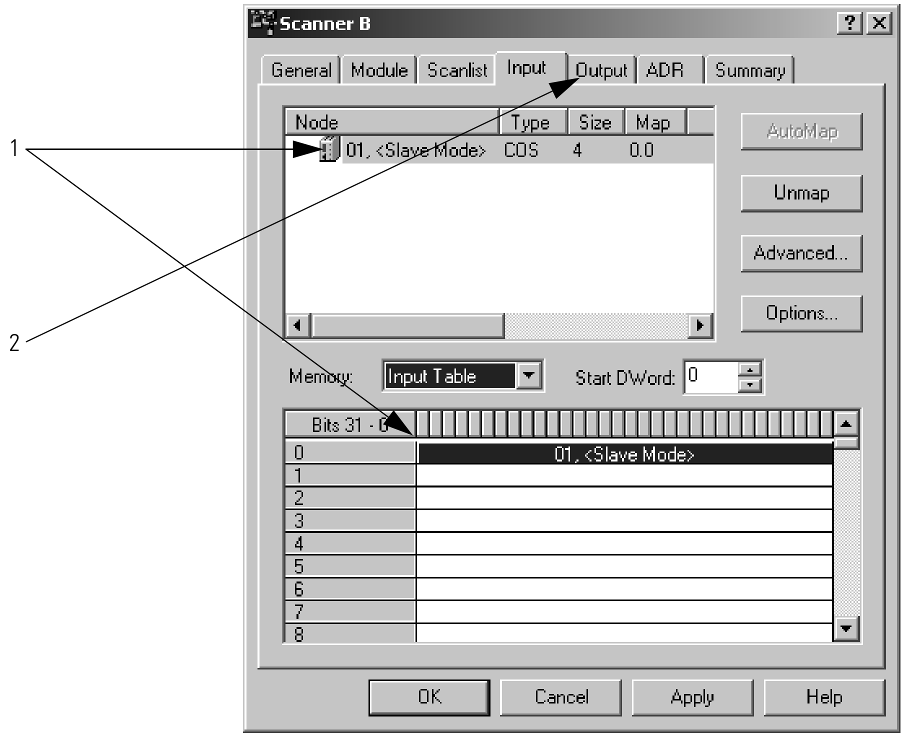
Map the Slave Mode Data

Complete the following steps to map Slave mode data.

1. Map the Slave mode data to the input memory of the slave scanner.

   This is the data that the scanner (controller) gets from the master.

2. Repeat for the data that the slave scanner (controller) sends to the master.



Node	Type	Size	Map
01, <Slave Mode>	COS	4	0.0

Memory: Input Table

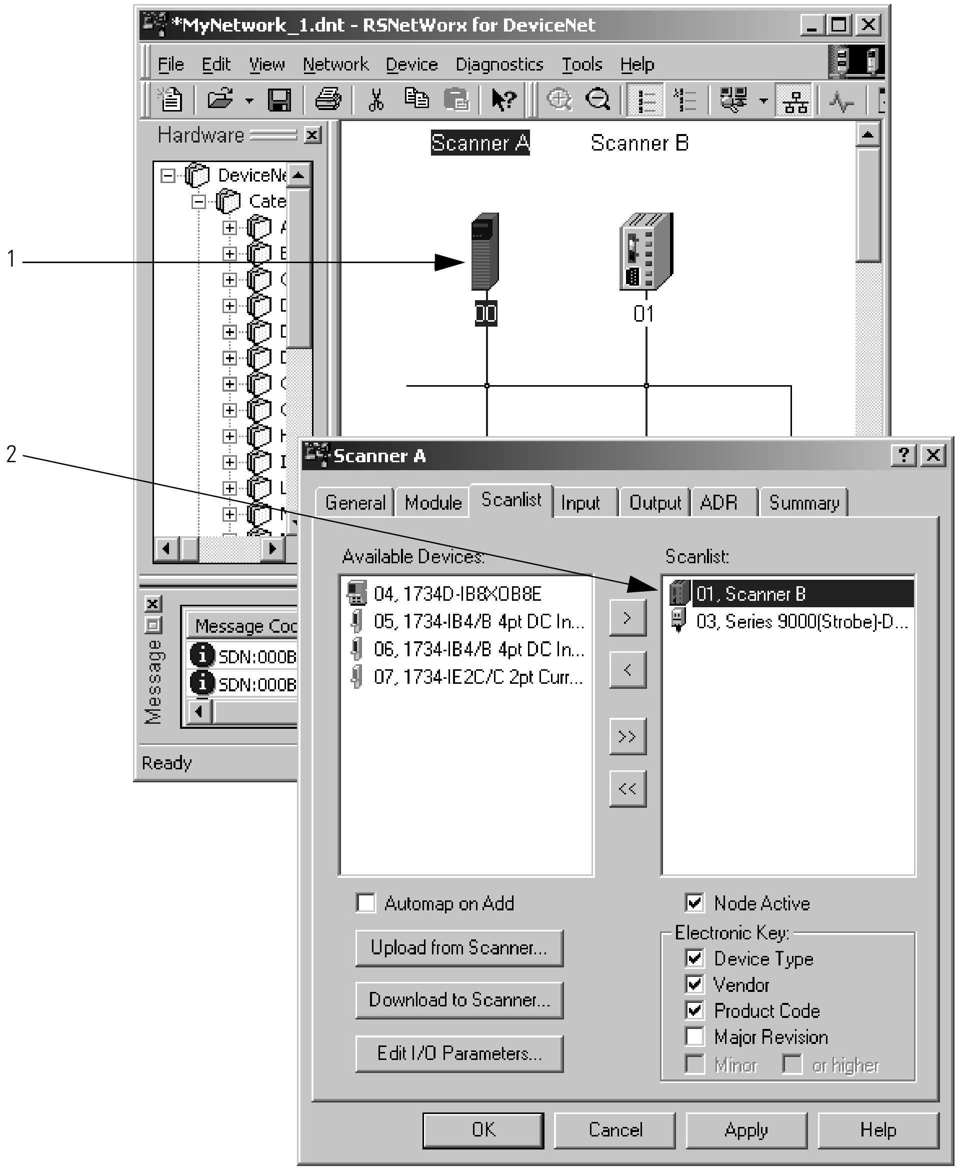
Start DWord: 0

Bits 31 - 0	
0	01, <Slave Mode>
1	
2	
3	
4	
5	
6	
7	
8	

Add the Slave to the Master Scanner’s Scan List

Complete the following steps to add the slave to the master’s scan list.

1. In RSNetWorx for DeviceNet software, double-click the master scanner to open its properties.
2. Add the slave to the scan list.



Available Devices:
- 04, 1734D-IB8XOB8E
- 05, 1734-IB4/B 4pt DC In...
- 06, 1734-IB4/B 4pt DC In...
- 07, 1734-IE2C/C 2pt Curr...

Scanlist:
- 01, Scanner B
- 03, Series 9000(Strobe)-D...

□ Automap on Add

☑ Node Active

Electronic Key:
- ☑ Device Type
- ☑ Vendor
- ☑ Product Code
- □ Major Revision
- □ Minor □ or higher

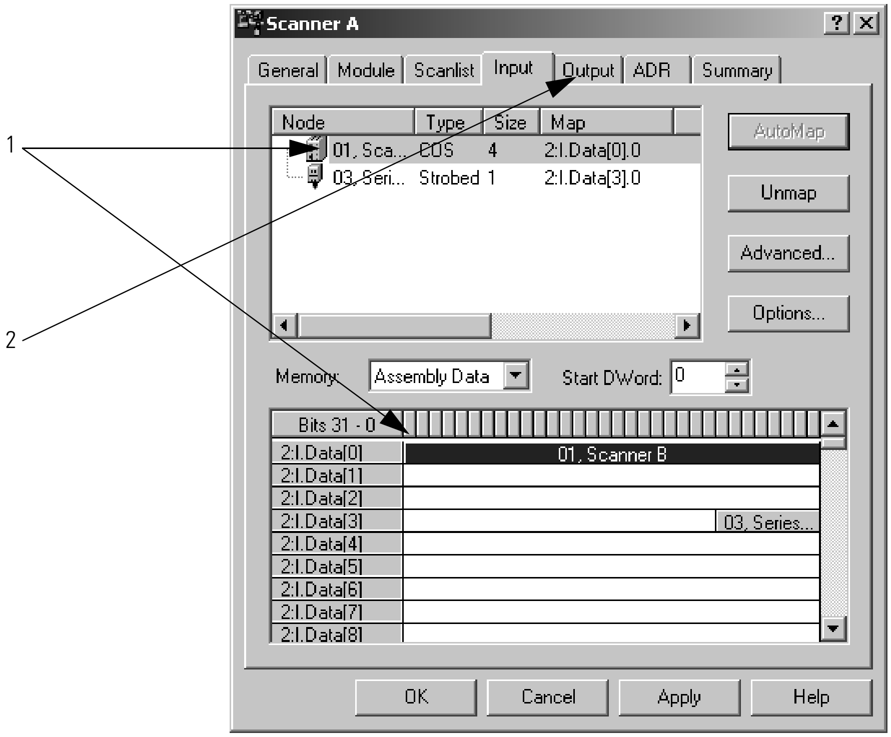
Map the Data of the Slave

Complete the following steps to map the data.

1. Map the slave scanner to the input memory of the master scanner.

   This is the data that the scanner (controller) gets from the slave.

2. Repeat for the data that the master scanner (controller) sends to the slave.



Node	Type	Size	Map
01, Sca...	COS	4	2:I.Data[0].0
03, Seri...	Strobed	1	2:I.Data[3].0

Memory: Assembly Data

Start DWord: 0

Bits 31 - 0	
2:I.Data[0]	01, Scanner B
2:I.Data[1]	
2:I.Data[2]	
2:I.Data[3]	03, Series...
2:I.Data[4]	
2:I.Data[5]	
2:I.Data[6]	
2:I.Data[7]	
2:I.Data[8]	

Place Both Scanners In Run Mode

To exchange data, place both scanners in Run mode. Refer to Place the Scanner in Run Mode on page 113 for more information on placing both scanners in Run mode.

Share Inputs

To let multiple scanners (controllers) consume input data from the same input device, complete the tasks in this section.

Add the Input to the First Scanner

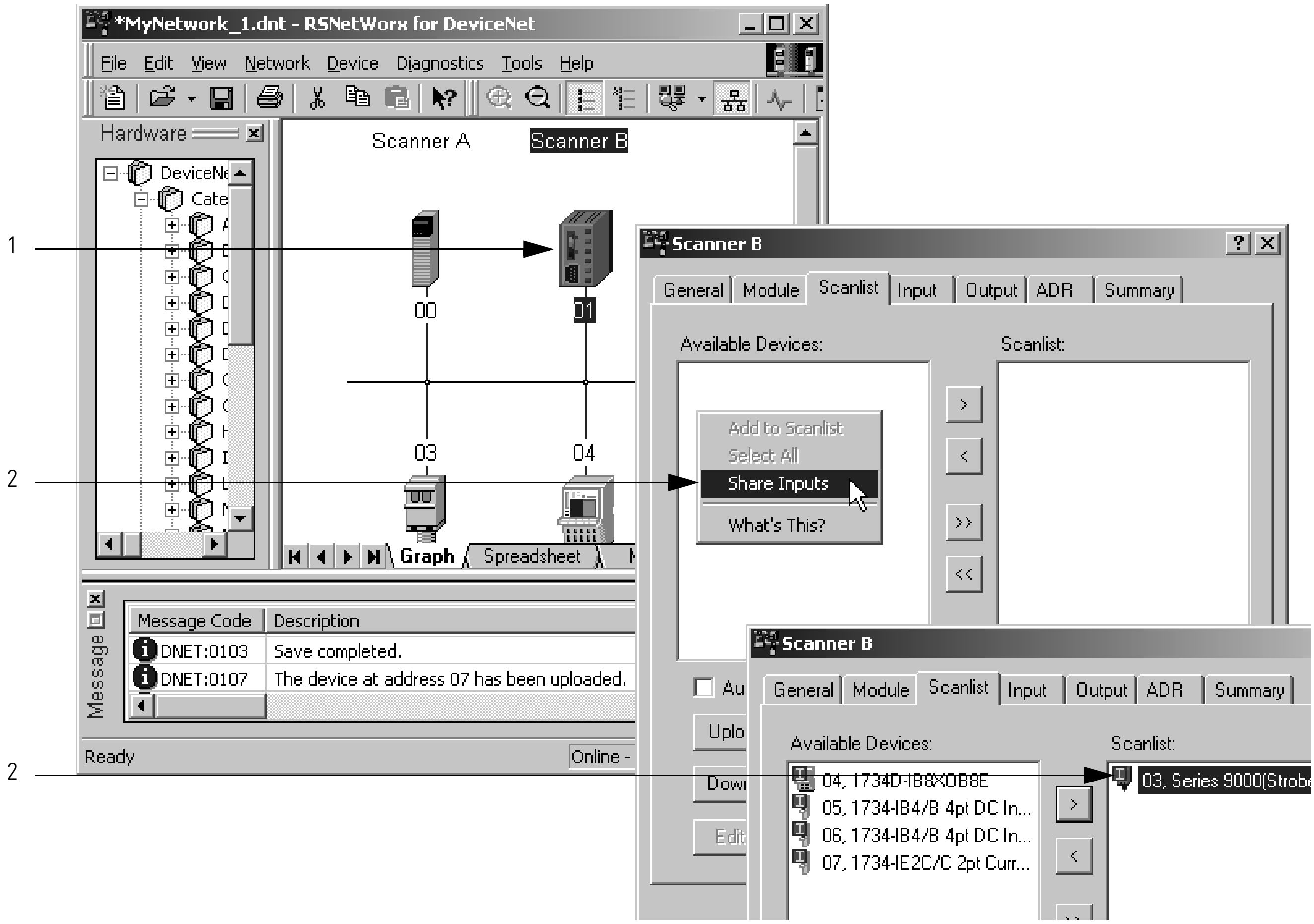
Establish communication between the input and one of the scanners. Use one of the following sections to establish communication:

- Configure the Network Offline on page 37
- Configure the Network Online on page 57

Add the Input to the Second Scanner

Complete the following steps to add the input to the second scanner.

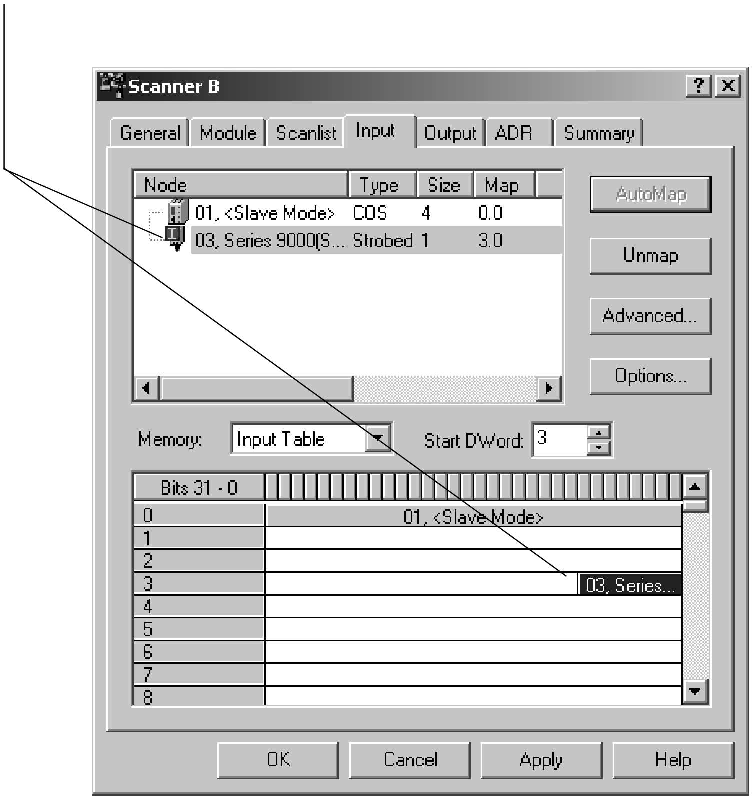
1. In RSNetWorx for DeviceNet software, display the scan list for the second scanner.
2. In the Available Devices list, right-click and choose Shared Inputs.
3. Add the input to the scan list.


1
2
2
*MyNetwork_1.dnt - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
Hardware
DeviceNe
Cate
Scanner A
Scanner B
00
01
03
04
Graph
Spreadsheet
Scanner B
General Module Scanlist Input Output ADR Summary
Available Devices:
Scanlist:
Add to Scanlist
Select All
Share Inputs
What's This?
Scanner B
General Module Scanlist Input Output ADR Summary
Available Devices:
Scanlist:
04, 1734D-IB8XOB8E
05, 1734-IB4/B 4pt DC In...
06, 1734-IB4/B 4pt DC In...
07, 1734-IE2C/C 2pt Curr...
03, Series 9000(Strob
Ready
Online -

Message Code	Description
DNET:0103	Save completed.
DNET:0107	The device at address 07 has been uploaded.

Map the Input Data in the Second Scanner

Map the input data to the input memory of the second scanner.



Node	Type	Size	Map
01, <Slave Mode>	COS	4	0.0
03, Series 9000(S...	Strobed	1	3.0

Memory: Input Table

Start DWord: 3

Bits 31 - 0	
0	01, <Slave Mode>
1	
2	
3	03, Series...
4	
5	
6	
7	
8	

Communicate with a PanelView Standard Terminal

This chapter describes how to configure and program communication with a PanelView Standard terminal on a DeviceNet network.

Choose Data Types

For the tags in the PanelView terminal, use the data types described in the following table as a starting point.

If the object on the PanelView screen reads or writes	Then use this data type	Which uses this many bits in the PanelView terminal
Single bit	Bit	1
Integer	Unsigned integer	16

Data types, such as signed integer and float, also work with Logix5000 controllers. However, they require additional configuration and programming.

Choose a Communication Method

You have three options to send data to and from a PanelView terminal.

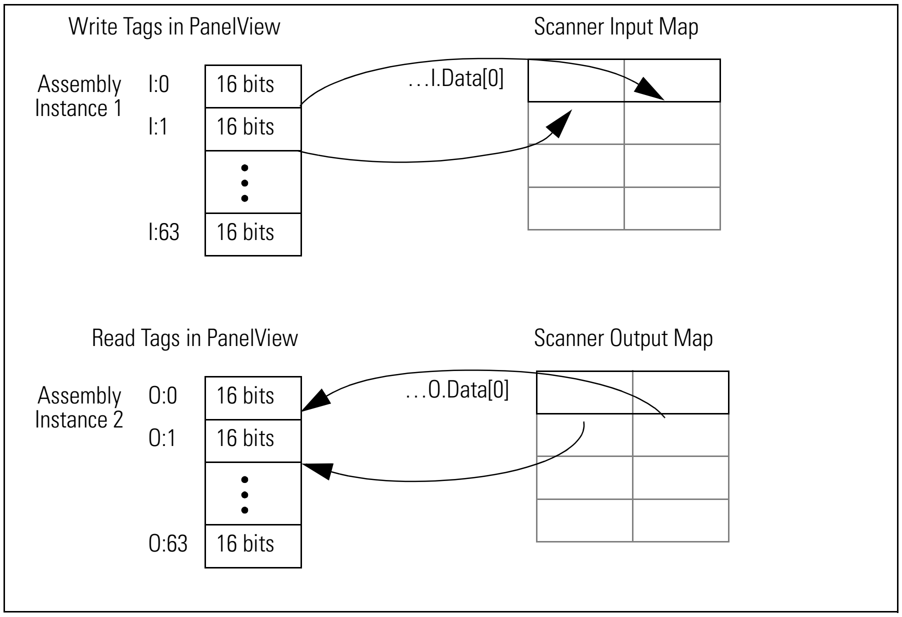
If you want to	Use this method	Considerations
Communicate with the PanelView terminal using the regular I/O communication of the DeviceNet network	I/O slave	• Easiest to use—requires no additional programming. • Use this as your first choice. • Higher priority on the network than explicit server and explicit client updates.
Communicate with the PanelView terminal based on conditions in your logic	Explicit server	• Provides additional data when you use up the I/O slave assemblies. • Lower priority on the network than I/O slave updates.
Use the PanelView terminal to get or set a parameter of a device on your DeviceNet network (not a controller)	Explicit client	• **Does not** use the controller or scanner. • Lower priority on the network than I/O slave updates.

I/O Slave Communication

I/O Slave

Scanner polls PanelView Terminal for I/O data.

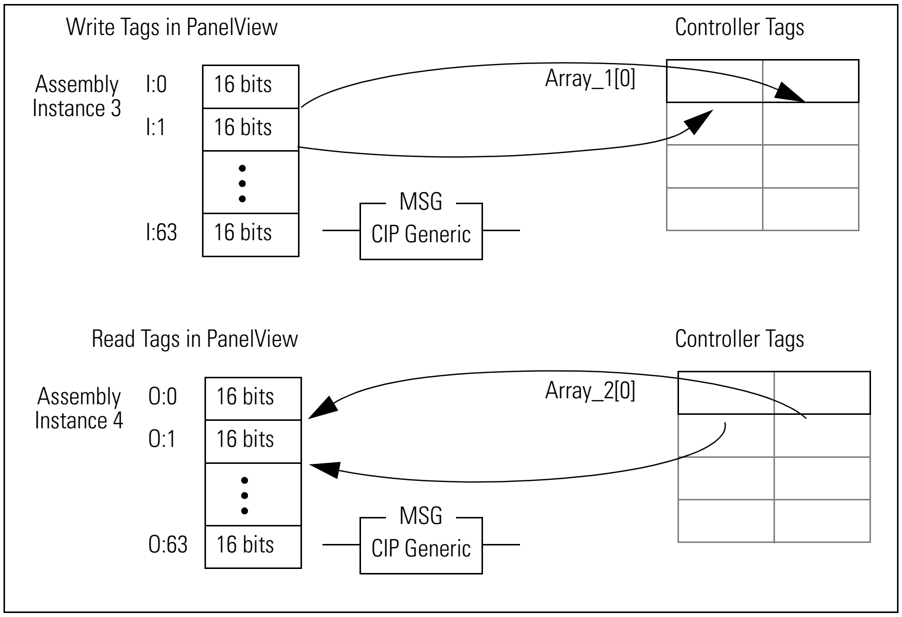
- You define the input and output sizes up to 64 words.
- Assembly instance 1 gives input data to the controller.
- Assembly instance 2 gets output data from the controller.



Explicit Server Communication

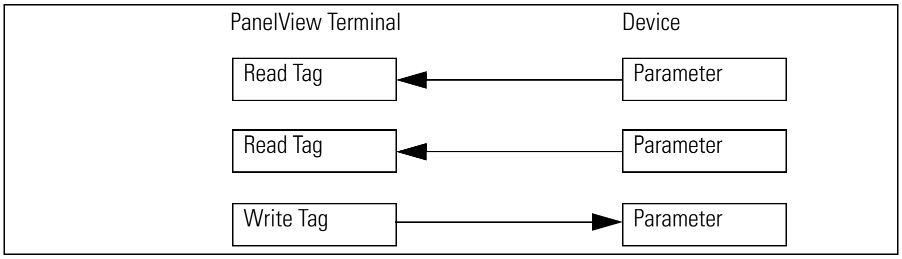
Explicit Server
Controller executes a MSG instruction that gets or sets data in the PanelView terminal.

- 14 assembly instances are available for explicit - server transfers.
- Instance #s are 3…16.
- You define an instance as either input data (I) or output data (O), but not both.
- Each instance provides 64 words of either input or output data for the terminal.



Explicit Client Communication

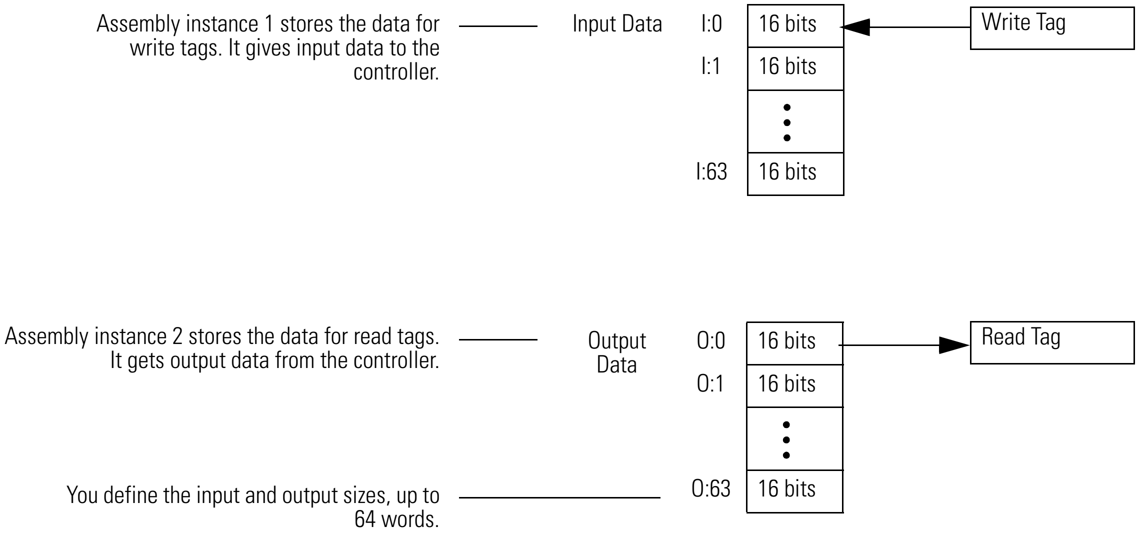
Explicit Client
PanelView terminal sets or gets data in another device on a tag-by-tag basis.



Plan and Configure I/O Slave Tags

Like the other DeviceNet communication modules, I/O slave tags use space in the input and output maps of the scanner. The scanner gets or sets the data on each scan of the DeviceNet network.

A PanelView terminal gives you two blocks of 16-bit words (assembly instances) for I/O slave tags.

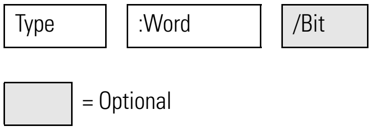

Assembly instance 1 stores the data for write tags. It gives input data to the controller.

Assembly instance 2 stores the data for read tags. It gets output data from the controller.

You define the input and output sizes, up to 64 words.

Use a Word/Bit Format for Each Tag

Each I/O slave tag requires a specific address in the corresponding assembly instance. A tag address uses the following format:

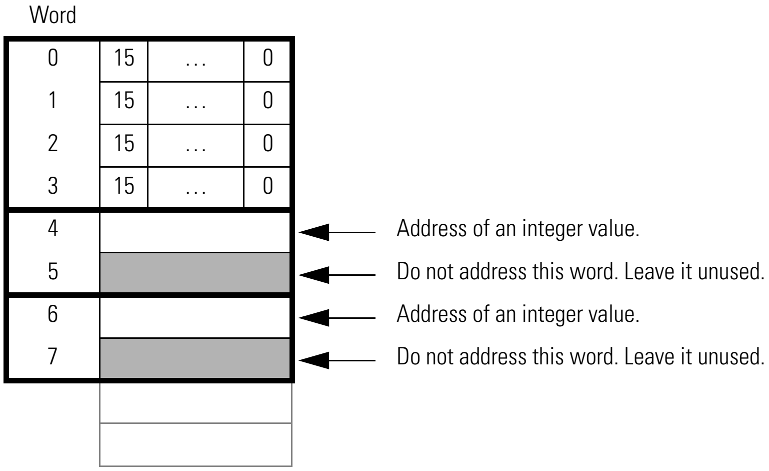


Where	Is	
Type	Type of tag	
	If the tag is a	**Then use**
	Write tag (sends input data to the controller)	I
	Read tag (gets output data from the controller)	O
Word	Specific 16-bit word within the assembly	
Bit	Specific bit within Word (0…15)	

For Integers, Skip Every Other Word

Logix5000 controllers use 32-bit integers (DINTs). To make your programming easier, lay out your PanelView tags as follows.

1. For bit-level tags, set aside an even number of words.
2. For each integer, set aside two words. Start each integer on an even word. This lets each integer map to its own element in the scanner/controller.



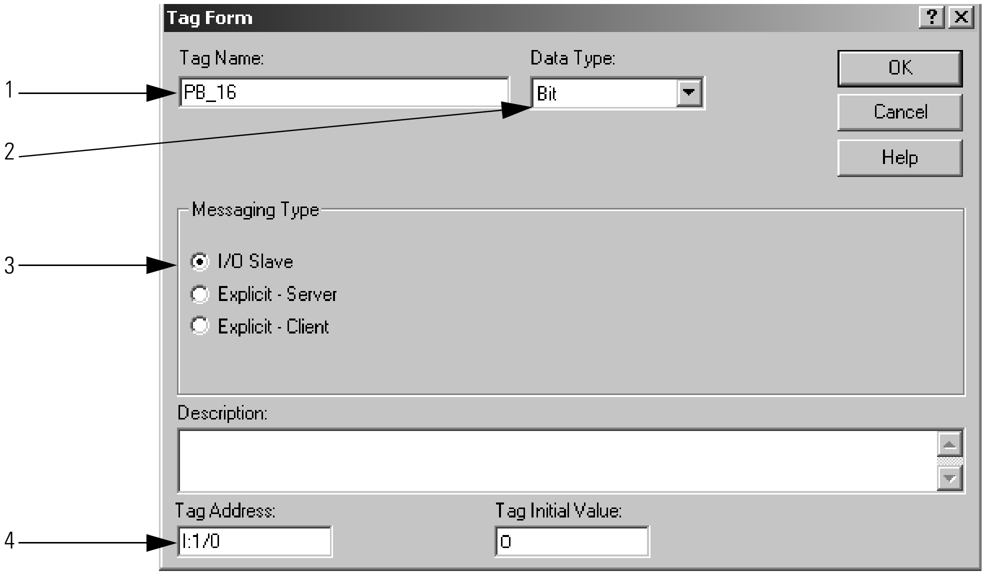
Word			
0	15	...	0
1	15	...	0
2	15	...	0
3	15	...	0
4			
5	▨		
6			
7	▨		

Word 4: Address of an integer value.
Word 5: Do not address this word. Leave it unused.
Word 6: Address of an integer value.
Word 7: Do not address this word. Leave it unused.

Configure an I/O Slave Tag

Complete the following steps to configure an I/O slave tag.

1. Type a descriptive name for the tag.
2. Choose the data type for the tag.
3. Let the scanner update the data.
4. Assign an address for the tag within the input or output assembly.



Tag Name: PB_16
Data Type: Bit
Messaging Type
● I/O Slave
○ Explicit - Server
○ Explicit - Client
Description:
Tag Address: I:1/0
Tag Initial Value: 0

Set Up the Terminal on Your Network

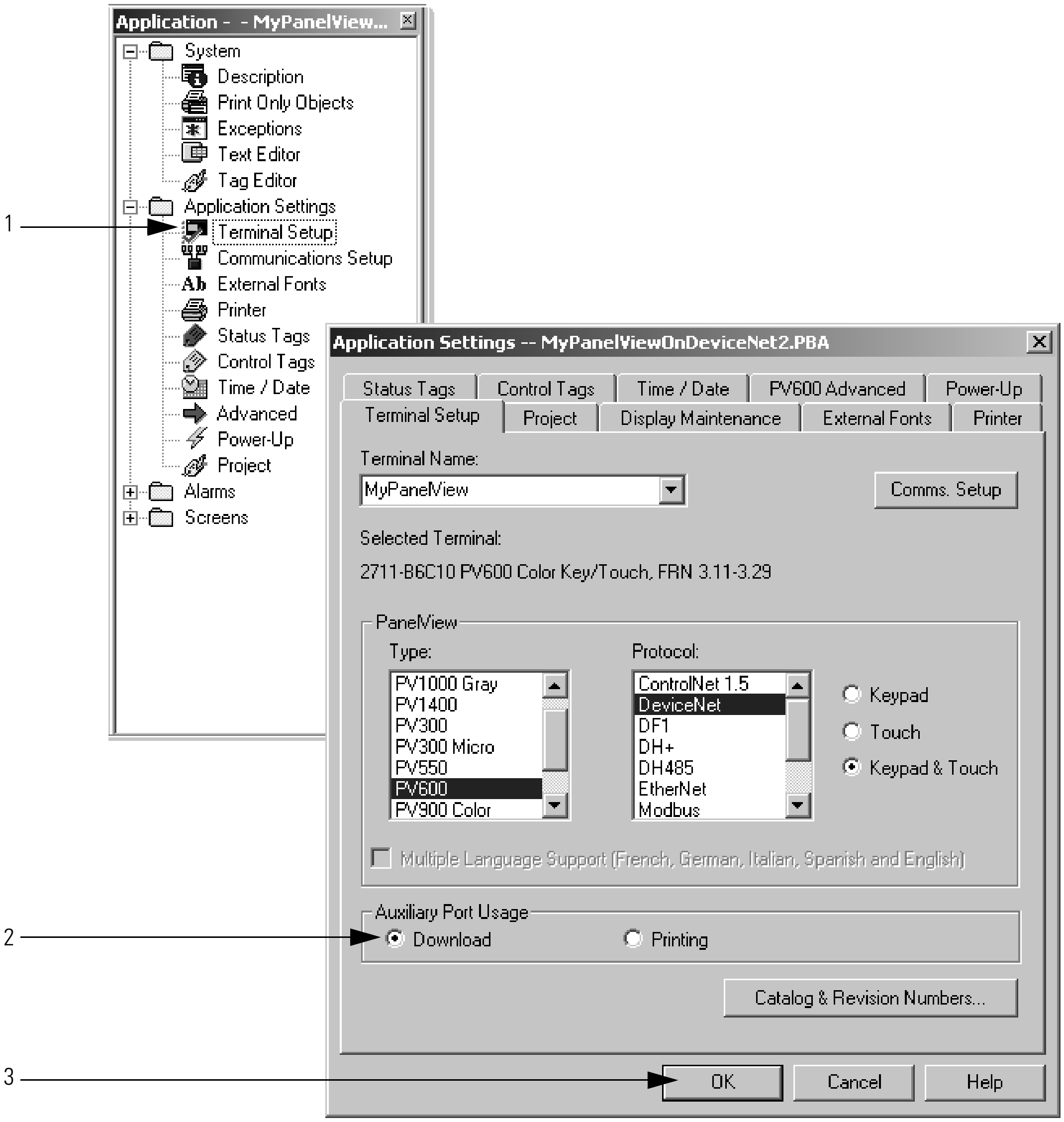
You must complete the following tasks in PanelBuilder32 software to configure a PanelView terminal for communication on a DeviceNet network:

- Set the Protocol
- Set the Node Address and I/O Sizes

Set the Protocol

Complete the following steps to set the protocol.

1. Double-click Terminal Setup.
2. Choose the auxiliary port usage.
3. Click OK.



Application Settings -- MyPanelViewOnDeviceNet2.PBA

Terminal Name: MyPanelView

Selected Terminal:

2711-B6C10 PV600 Color Key/Touch, FRN 3.11-3.29

PanelView Type: PV600

Protocol: DeviceNet

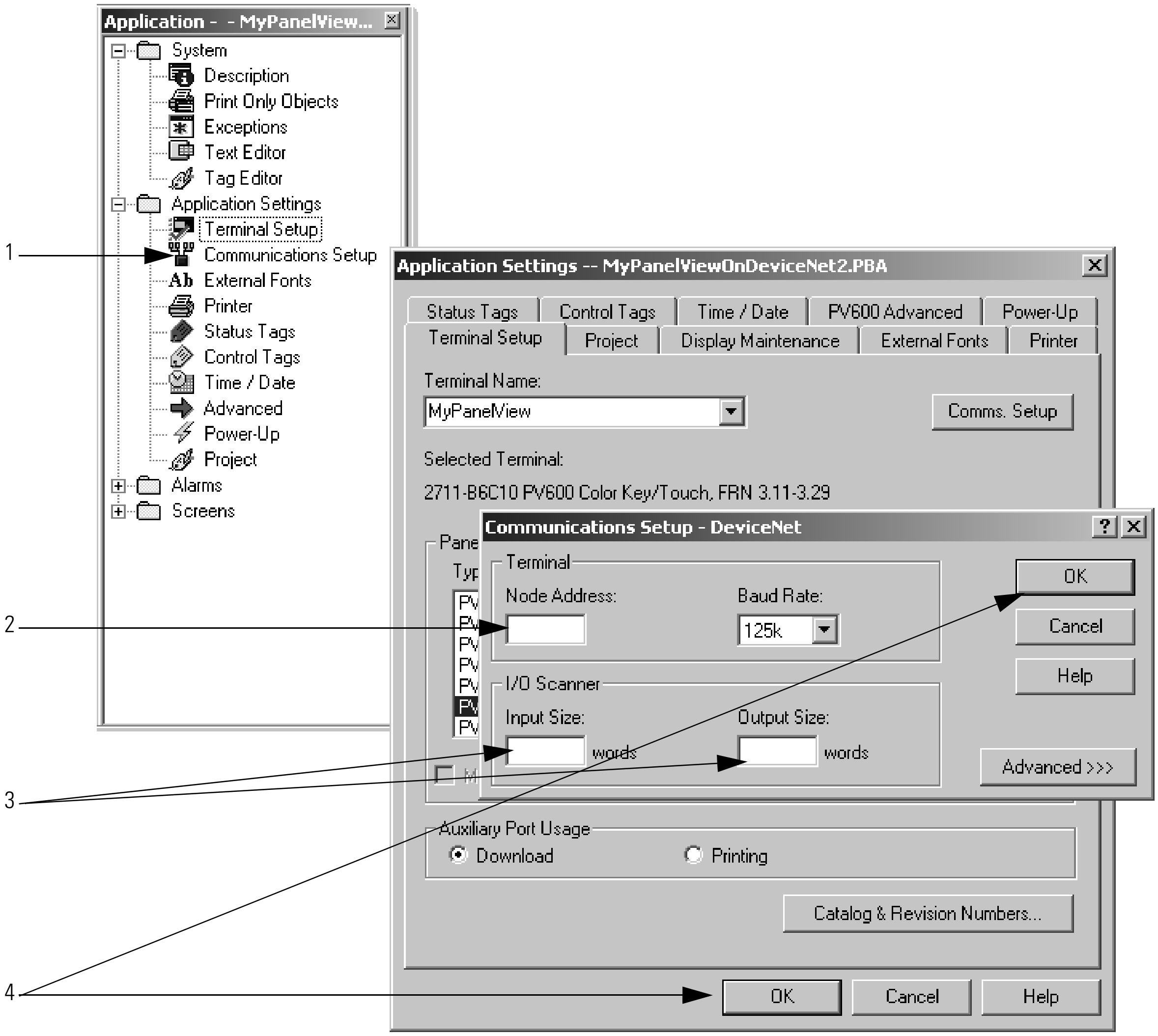
● Keypad & Touch

Auxiliary Port Usage: ● Download

Set the Node Address and I/O Sizes

Complete the following steps to set the node address and I/O sizes.

1. Double-click Communication Setup.
2. Type the address of the PanelView terminal.
3. Type the number of input words and output words that you will use (64 maximum each).
4. Click OK to close the dialog boxes.



Configure the Scanner to Update I/O Slave Tags

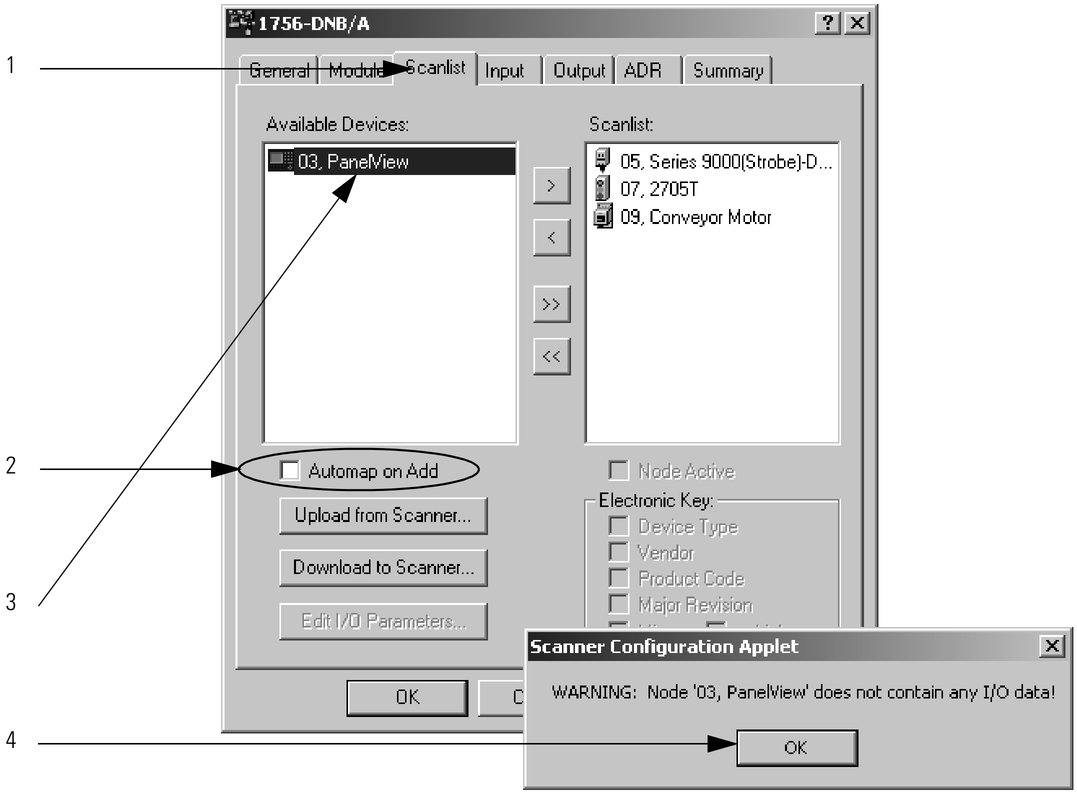
Complete the following tasks to access I/O slave tags and map the data to the input and output maps of the scanner:

- Add the Terminal to the Scan List
- Edit I/O Parameters
- Map Input and Output Data

Add the Terminal to the Scan List

Complete the following steps to add the terminal to the scanlist.

1. Click the Scanlist tab.
2. Clear Automap on Add.
3. Add the terminal to the scanlist.
4. Click OK.

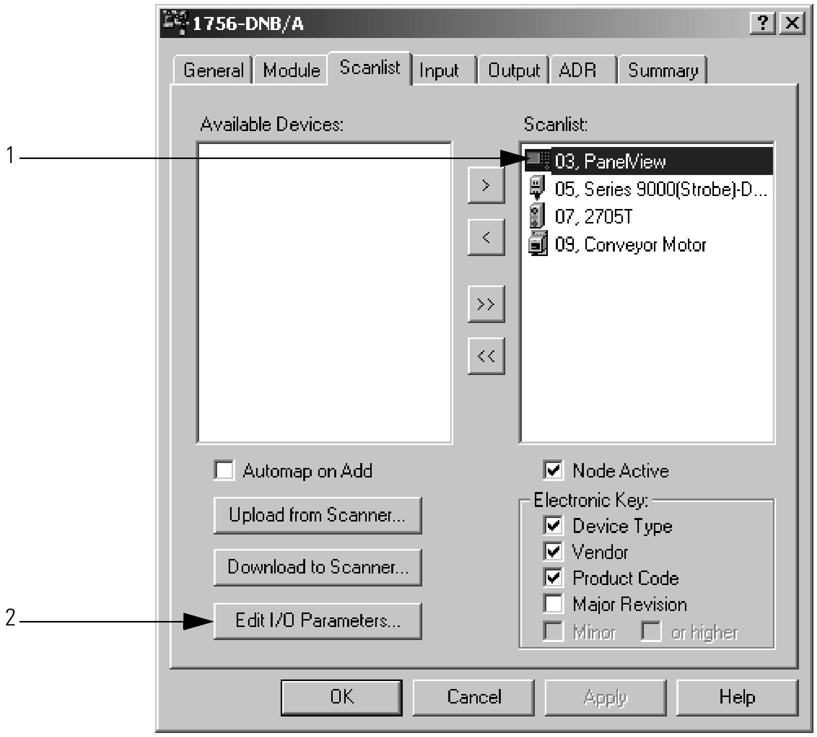


WARNING: Node '03, PanelView' does not contain any I/O data!

Edit I/O Parameters

Complete the following steps to edit I/O parameters.

1. Select the terminals.
2. Click Edit I/O Parameters.
3. Enter the input and output sizes in bytes. Make sure each number is two times the number you entered in the communication set-up of the terminal (1 word = 2 bytes).
4. Click OK.



1756-DNB/A

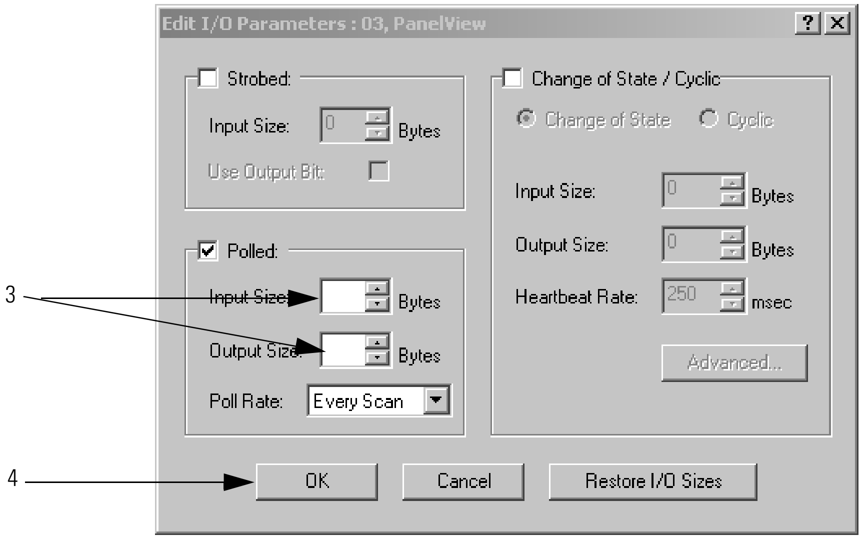
Available Devices:

Scanlist:
03, PanelView
05, Series 9000(Strobe)-D...
07, 2705T
09, Conveyor Motor

□ Automap on Add

☑ Node Active

Electronic Key:
☑ Device Type
☑ Vendor
☑ Product Code
□ Major Revision
□ Minor □ or higher



Edit I/O Parameters : 03, PanelView

□ Strobed:
Input Size: 0 Bytes
Use Output Bit: □

☑ Polled:
Input Size: Bytes
Output Size: Bytes
Poll Rate: Every Scan

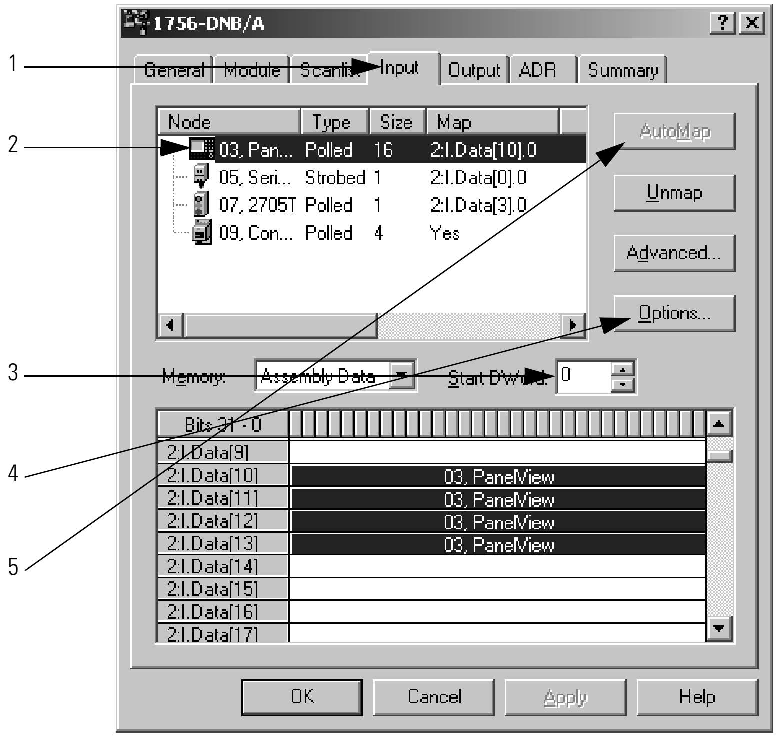
□ Change of State / Cyclic
● Change of State ○ Cyclic
Input Size: 0 Bytes
Output Size: 0 Bytes
Heartbeat Rate: 250 msec

Map Input and Output Data

Complete the following steps to map input and output data.

1. Click the Input tab.
2. Select the terminal.
3. Enter the starting element for the data in the input array.
4. Set the alignment option (typically DWord align).
5. Click AutoMap.

   An entry for the device shows up in the input array.
6. Click the Output tab and repeat steps 2 through 5.



Node	Type	Size	Map
03, Pan...	Polled	16	2:I.Data[10].0
05, Seri...	Strobed	1	2:I.Data[0].0
07, 2705T	Polled	1	2:I.Data[3].0
09, Con...	Polled	4	Yes

Memory: Assembly Data

Start DWord: 0

Bits 31 - 0	
2:I.Data[9]	
2:I.Data[10]	03, PanelView
2:I.Data[11]	03, PanelView
2:I.Data[12]	03, PanelView
2:I.Data[13]	03, PanelView
2:I.Data[14]	
2:I.Data[15]	
2:I.Data[16]	
2:I.Data[17]	

Address I/O Slave Tags in the RSLogix 5000 Programming Software Project

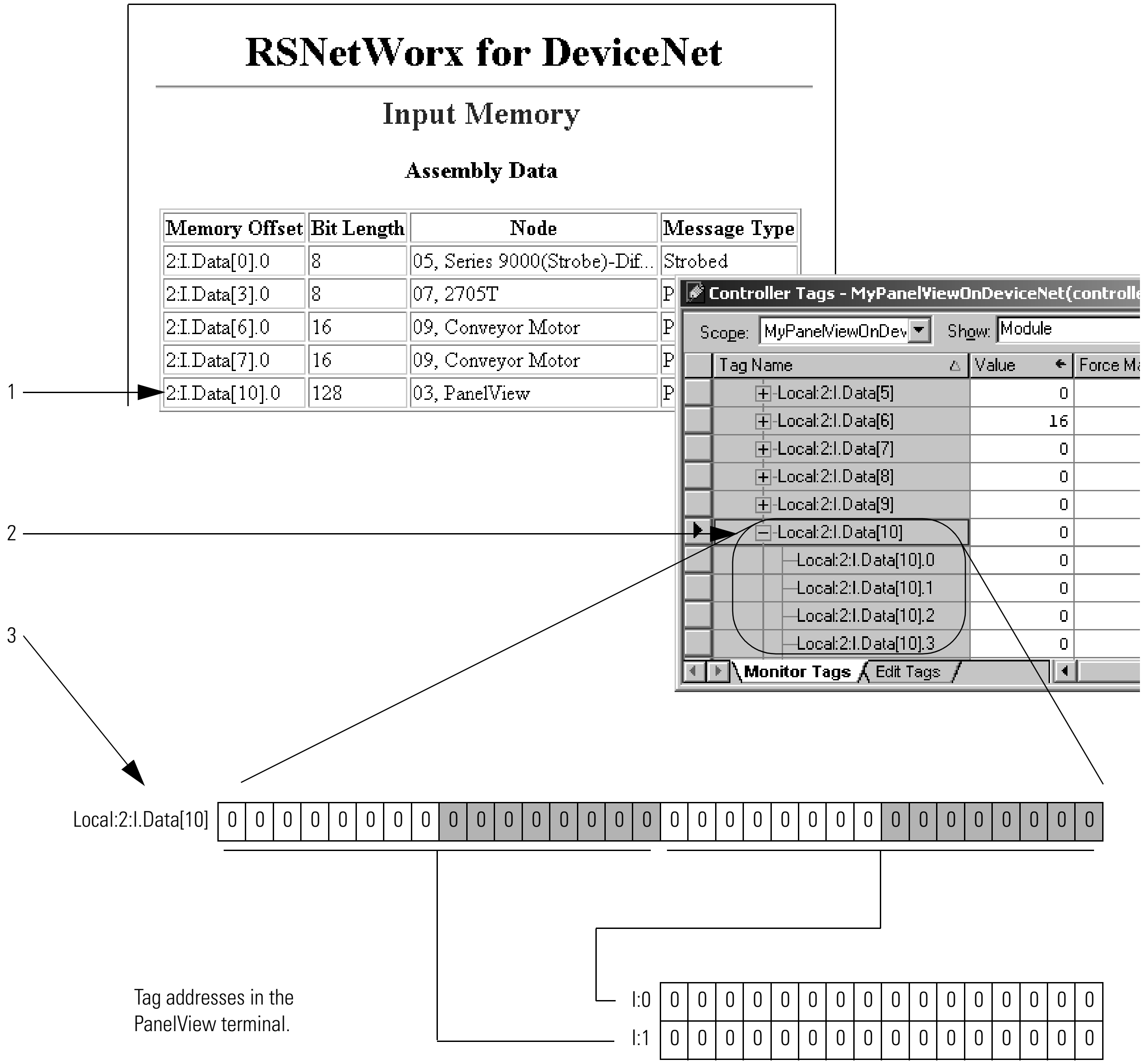
You must get the following information to find the data for an I/O slave tag in your RSLogix 5000 programming software project:

- RSNetWorx for DeviceNet report for the network
- Address for the tag in the PanelView terminal

Complete the following steps to get the information described previously.

1. On the report for the network, find the memory address for the PanelView terminal.
2. Find the corresponding tag in the controller-scoped tags of the controller.
3. Find the data within the controller tag.

   Use the tag address as a reference.



RSNetWorx for DeviceNet

Input Memory

Assembly Data

Memory Offset	Bit Length	Node	Message Type
2:I.Data[0].0	8	05, Series 9000(Strobe)-Dif...	Strobed
2:I.Data[3].0	8	07, 2705T	P
2:I.Data[6].0	16	09, Conveyor Motor	P
2:I.Data[7].0	16	09, Conveyor Motor	P
2:I.Data[10].0	128	03, PanelView	P

Controller Tags - MyPanelViewOnDeviceNet(controll

Scope: MyPanelViewOnDev Show: Module

Tag Name	Value	Force Ma
+Local:2:I.Data[5]	0	
+Local:2:I.Data[6]	16	
+Local:2:I.Data[7]	0	
+Local:2:I.Data[8]	0	
+Local:2:I.Data[9]	0	
−Local:2:I.Data[10]	0	
Local:2:I.Data[10].0	0	
Local:2:I.Data[10].1	0	
Local:2:I.Data[10].2	0	
Local:2:I.Data[10].3	0	

Local:2:I.Data[10] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

I:0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

I:1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Tag addresses in the PanelView terminal.

DeviceNet tags use the format described below.

Scanner Memory Format	Tag in Controller
slot:type.Data[element].bit	location:type.Data[element].bit

Where	Is		
Location	Location of the scanner in the system		
	If you have this scanner	**Then location is**	
	ControlLogix 1756-DNB	**In a**	**Location is**
		local chassis	Local:slot_number_of_scanner
		remote chassis	adapter:slot_number_of_scanner where: adapter is the name of the EtherNet/IP or ControlNet module in the remote chassis.
	CompactLogix 1769-SDN	Local:slot_number_of_scanner	
	SoftLogix 5800 1784-PCIDS		
	EtherNet/IP to DeviceNet Linking Device (1788-EN2DN)	The name of the scanner in the I/O configuration of the controller	
	ControlNet to DeviceNet Linking Device (1788-CN2DN)		
Type	Type of data:		
	Where	**Is**	
	Input from a device	I	
	Output to a device	O	
	Status of the network	S	
Element	A specific DINT (DWord, 32-bit integer) within the array		
Bit	A specific bit within an integer		

SoftLogix 5800 Controller

The SoftLogix 5800 scanner 1784-PCIDS organizes input and output memory in 16-bit words. It uses the following address format.

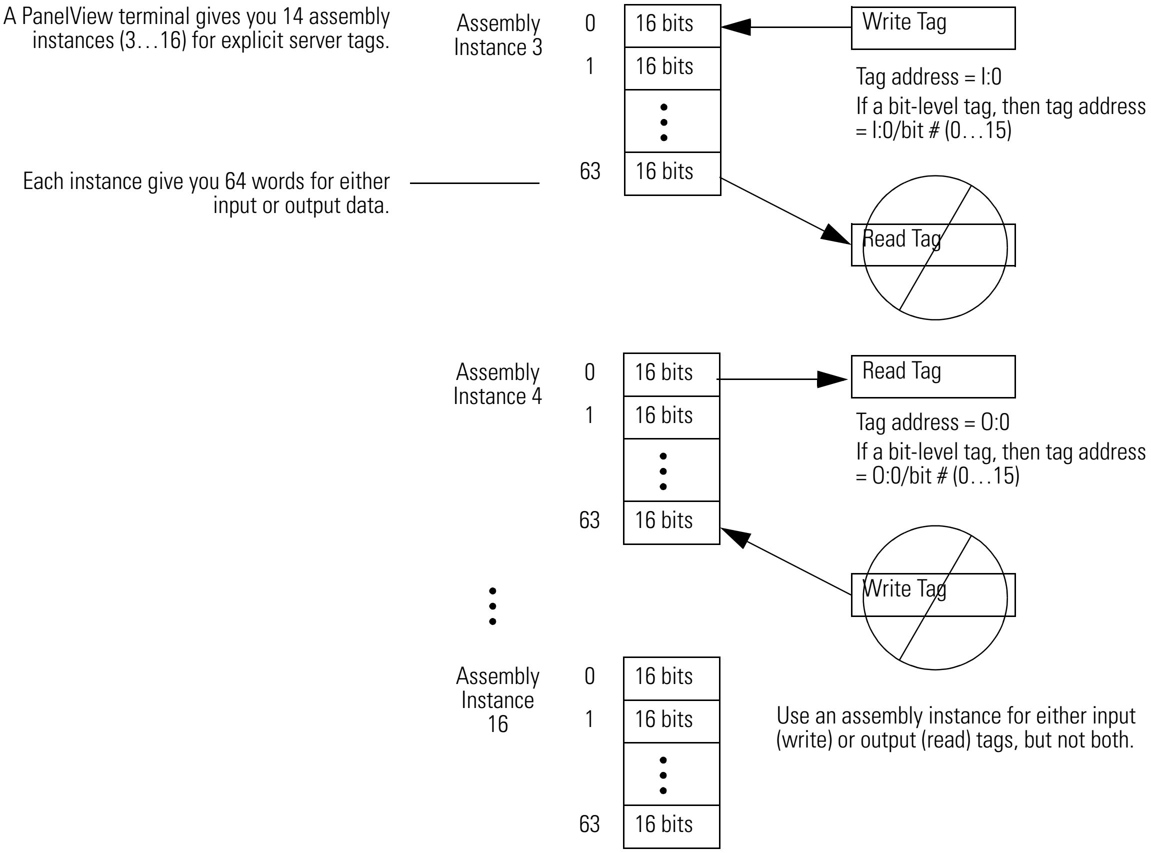
word.bit

Where	Is
Word	INT (16-bit integer) with the memory of the scanner
Bit	A specific bit within an integer

Plan and Configure Explicit Server Tags

Explicit server tags are similar to I/O tags except that the controller initiates the communication with the terminal. Explicit server tags **do not** show up on the input and output maps of the scanner.

Assign Assembly Instances


A PanelView terminal gives you 14 assembly instances (3…16) for explicit server tags.
Each instance give you 64 words for either input or output data.
Assembly Instance 3
0
1
63
16 bits
Write Tag
Tag address = I:0
If a bit-level tag, then tag address = I:0/bit # (0…15)
Read Tag
Assembly Instance 4
Read Tag
Tag address = O:0
If a bit-level tag, then tag address = O:0/bit # (0…15)
Write Tag
Assembly Instance 16
Use an assembly instance for either input (write) or output (read) tags, but not both.

Determine how you will use each assembly instance.

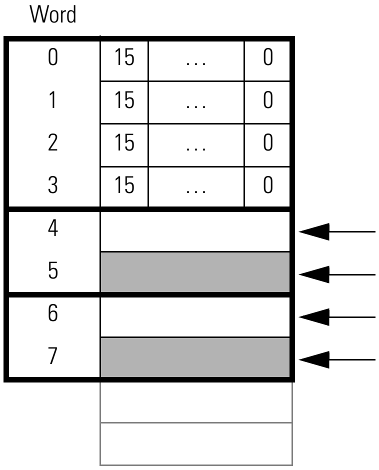
Instance Number	Input (Write) or Output (Read)	Instance Number	Input (Write) or Output (Read)
1	input	9	
2	output	10	
3		11	
4		12	
5		13	
6		14	
7		15	
8		16	

For Integers, Skip Every Other Word

Logix5000 controllers use 32-bit integers (DINTs). Complete the following steps to lay out your PanelView tags in a method that makes programming easier.

1. For bit-level tags, set aside an even number of words.
2. For each integer, set aside 2 words.

   Start each integer on an even word. This method lets each integer map to its own element in the scanner/controller.



Address of an integer value.

Do not address this word. Leave it unused.

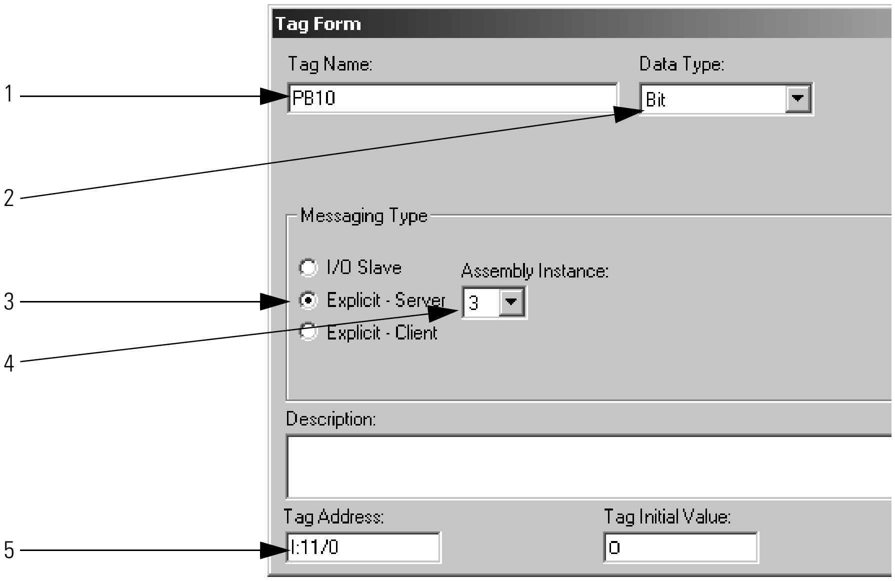
Address of an integer value.

Do not address this word. Leave it unused.

Configure an Explicit Server Tag

Complete the following steps to configure an Explicit-Server tag.

1. Type a descriptive name for the tag.
2. Choose the data type for the tag.
3. Let the controller initiate the update.
4. Choose the assembly instance for the tag.
5. Assign an address for the tag within the assembly instance.
   - Write tag = I:word/bit
   - Read tag = O:word/bit



Tag Form

Tag Name: PB10

Data Type: Bit

Messaging Type

○ I/O Slave
● Explicit - Server
○ Explicit - Client

Assembly Instance: 3

Description:

Tag Address: I:11/0

Tag Initial Value: 0

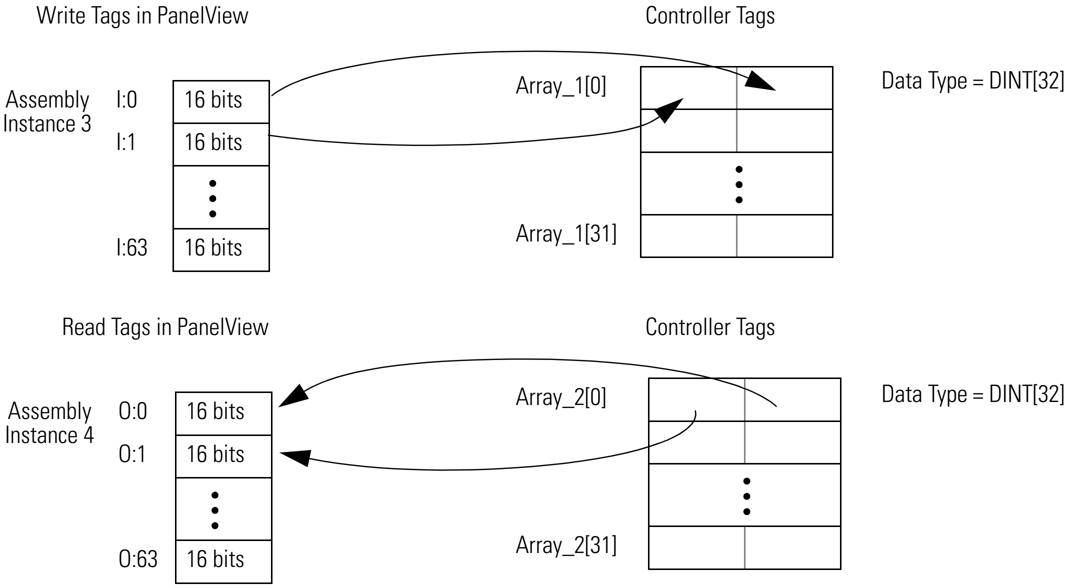
Program the Controller to Get/Set Explicit Server Tags

Complete the following tasks to let the controller read or write data from or to an Explicit-Server tag:

- Create an Array for the Assembly Instance
- Enter and Configure the MSG Instruction
- Set the Communication Path

Create an Array for the Assembly Instance

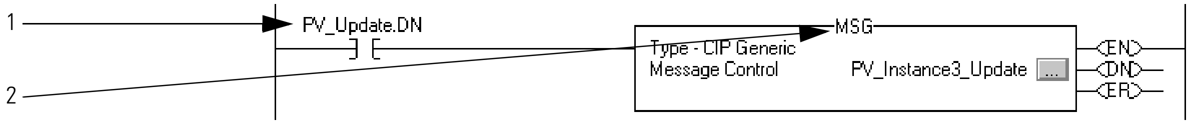
For each assembly instance that you use for explicit server tags, create an array in the RSLogix 5000 project for the data.



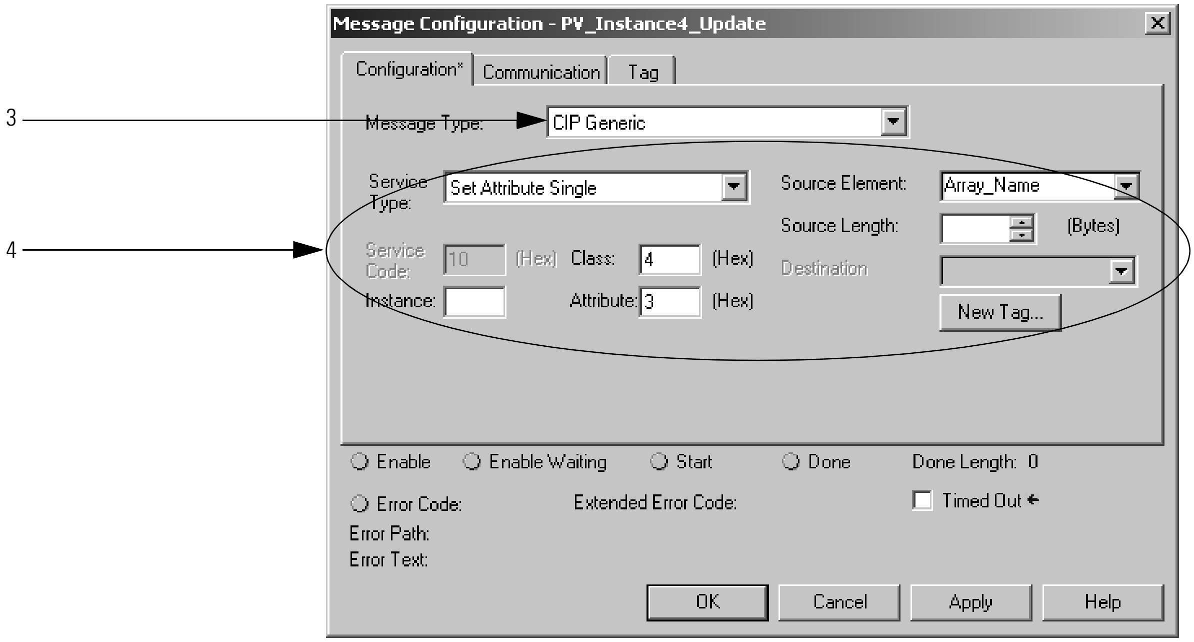
Enter and Configure the MSG Instruction

Complete the following steps to enter and configure the MSG instruction.

1. Enter a condition for the data transfer, such as the DN bit of a timer.
2. Enter a MSG instruction.



3. Select CIP Generic.
4. Complete the configuration to send output data.
   a. From the Service Type pull-down menu, choose Set Attribute Single.
   b. From the Source Element pull-down menu, choose the array that has the data.
   c. In the Source Length field, enter the number of bytes that you have addressed in the PanelView instance (words x 2).
   d. In the Class field, type 4.
   e. In the Instance field, type the assembly instance of the data in the PanelView terminal. Convert it to hexadecimal format.
   f. In the Attribute field, type 3.



Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex)

Class: 4 (Hex)

Instance:

Attribute: 3 (Hex)

Source Element: Array_Name

Source Length: (Bytes)

Destination:

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

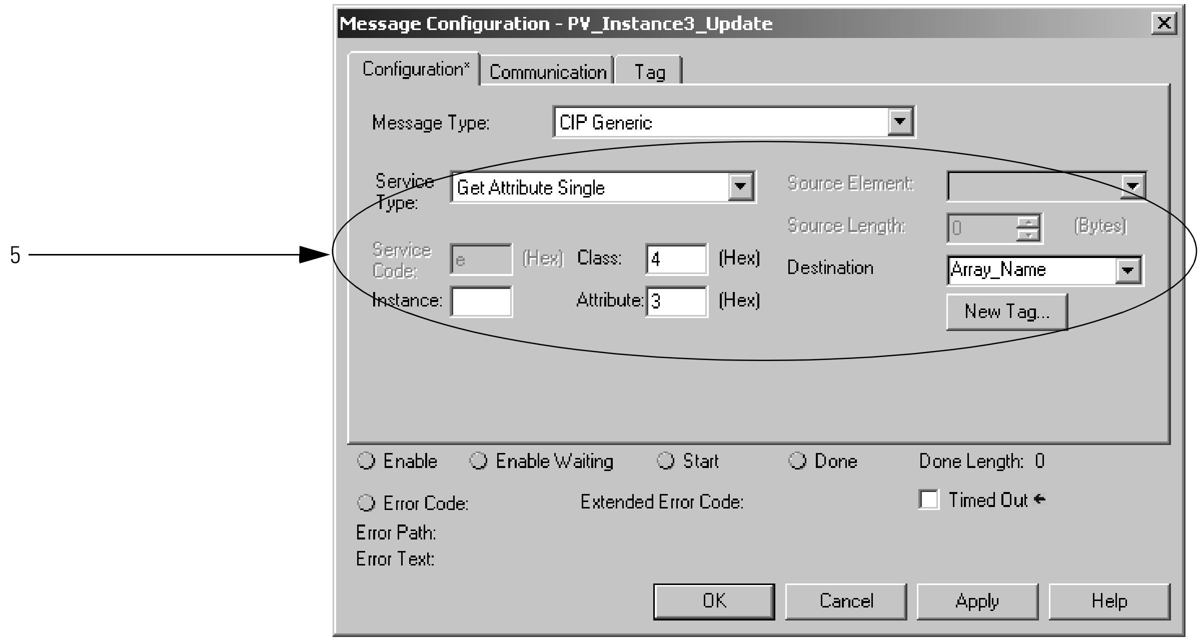
○ Error Code: Extended Error Code: □ Timed Out ←

Error Path:

Error Text:

5. Complete the configuration to get input data.
   a. From the Service Type pull-down menu, choose Get Attribute Single.
   b. From the Destination pull-down menu, choose the array to store the data.
   c. In the Class field, type 4.
   d. In the Instance field, type the assembly instance of the data in the PanelView terminal.

      Convert it to hexadecimal format.
   e. In the Attribute field, type 3.


Message Configuration - PV_Instance3_Update

Message Type: CIP Generic

Service Type: Get Attribute Single

Service Code: e (Hex)

Class: 4 (Hex)

Instance:

Attribute: 3 (Hex)

Source Element:

Source Length: 0 (Bytes)

Destination: Array_Name

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out ←

Error Path:

Error Text:

Set the Communication Path

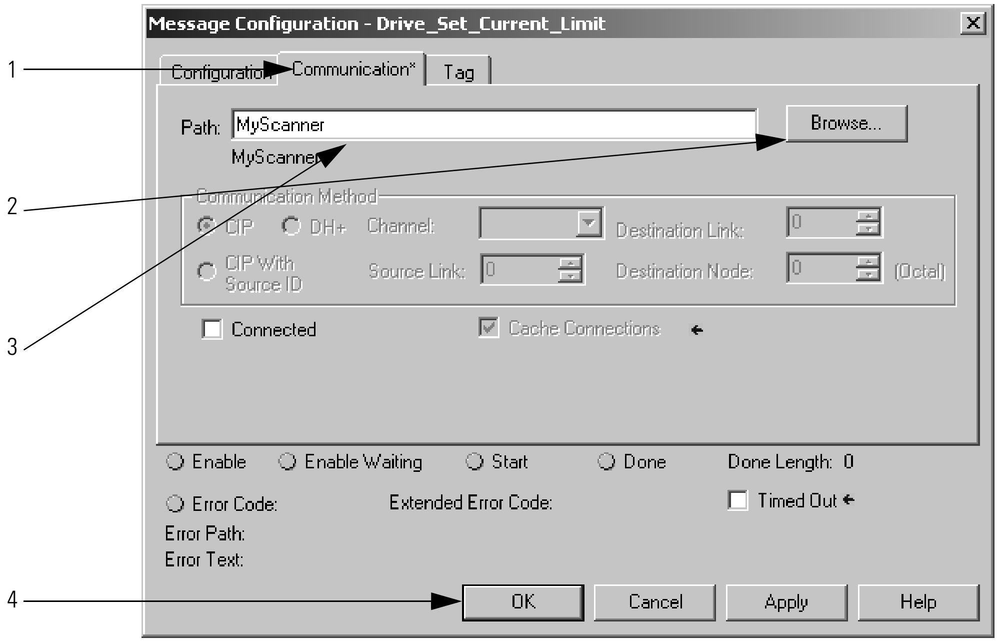
The communication path specifies the route to the PanelView terminal. A communication path uses the following format:

scanner_name,2,device_address

Where	Is
scanner_name	The name of the scanner in the I/O Configuration folder of the controller.
device_address	The address of the device on the DeviceNet network.

Complete the following steps to set the communication path.

1. Click the Communication tab.
2. Click the Browse button and select the scanner.
3. Type the rest of the path.
4. Click OK.



Message Configuration - Drive_Set_Current_Limit

Path: MyScanner

MyScanner

Communication Method

● CIP ○ DH+ Channel: Destination Link: 0

○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

□ Connected ☑ Cache Connections

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

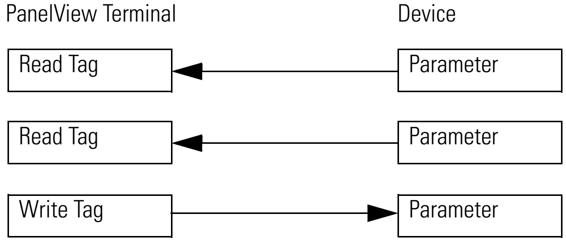
Error Path:

Error Text:

For more information on programming MSG instructions, see the Logix5000 Controller General Instructions Reference Manual, publication 1756-RM003.

Configure Explicit Client Tags

Use an Explicit Client tag to let the PanelView terminal get or set a parameter of another device on the DeviceNet network.



An Explicit Client tag **does not** do the following:

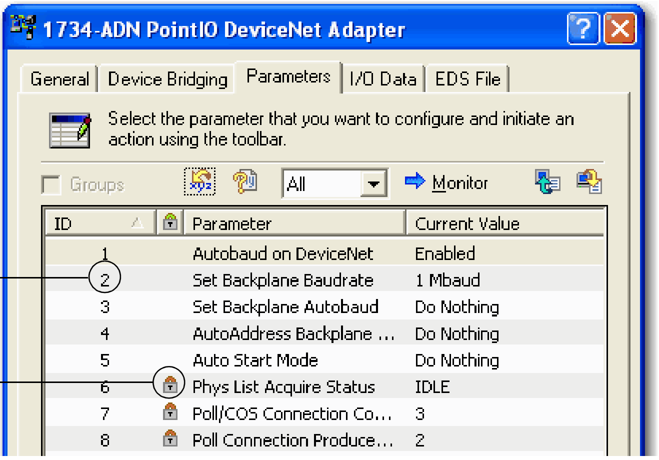
- Show up on the input or output map of the scanner
- Involve the controller
- Use an address in an assembly instance of the PanelView terminal

Complete the following tasks to configure Explicit Client tags:

- Determine the Parameter Number to Access
- Determine the Configuration of the Parameter
- Configure an Explicit Client Tag

Determine the Parameter Number to Access

Use RSNetWorx for DeviceNet software to determine the parameter number that you want to access. Some parameters are read-only and are shown with a lock symbol.



Select the parameter that you want to configure and initiate an action using the toolbar.

ID	🔒	Parameter	Current Value
1		Autobaud on DeviceNet	Enabled
2		Set Backplane Baudrate	1 Mbaud
3		Set Backplane Autobaud	Do Nothing
4		AutoAddress Backplane ...	Do Nothing
5		Auto Start Mode	Do Nothing
6	🔒	Phys List Acquire Status	IDLE
7	🔒	Poll/COS Connection Co...	3
8	🔒	Poll Connection Produce...	2

Parameter Number

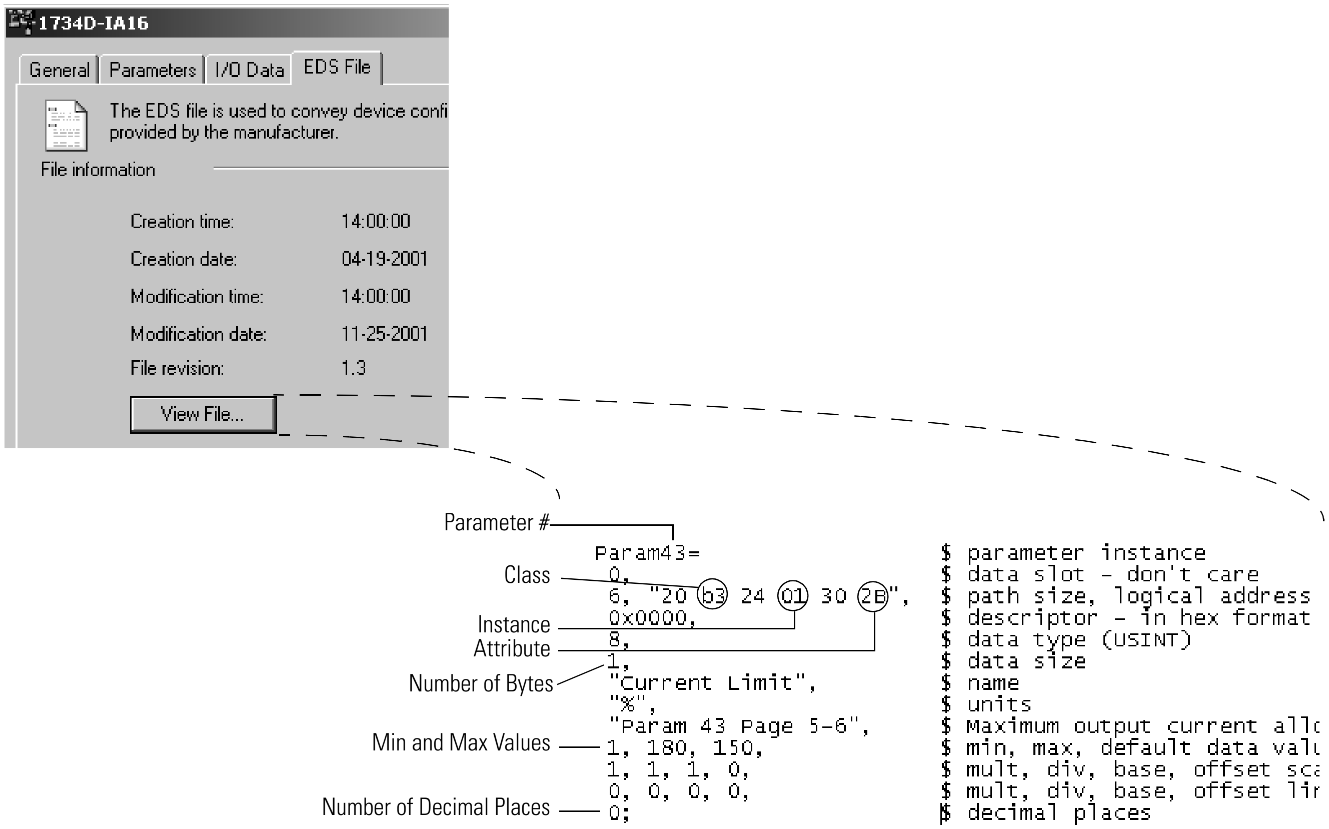
This symbol means you can get only the value of the parameter (read-only.)

Determine the Configuration of the Parameter

Find the information about the parameters listed in the following table to get or set a parameter.

Item	Value
Class # (hex)	
Instance # (hex)	
Attribute # (hex)	
Number of bytes (size)	
Minimum value	
Maximum value	
Decimal places Some devices assume a specific number of decimal places in a value.	

In addition to the documentation for the device, the EDS file may also give you the required information.



The EDS file is used to convey device confi provided by the manufacturer.

File information

Creation time:	14:00:00
Creation date:	04-19-2001
Modification time:	14:00:00
Modification date:	11-25-2001
File revision:	1.3

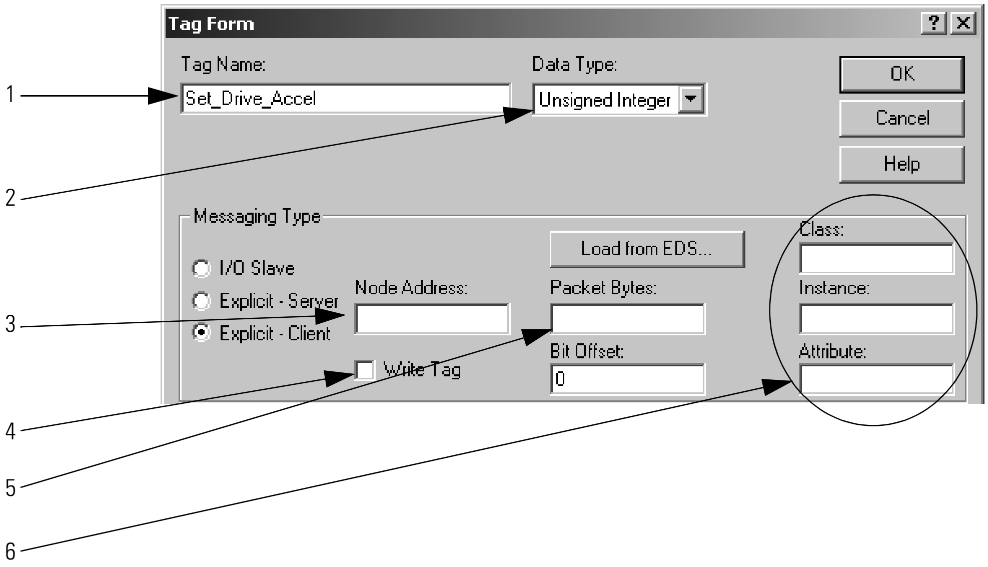
```
Param43=                                $ parameter instance
 0,                                     $ data slot - don't care
 6, "20 b3 24 01 30 2B",                $ path size, logical address
 0x0000,                                $ descriptor - in hex format
 8,                                     $ data type (USINT)
 1,                                     $ data size
 "Current Limit",                       $ name
 "%",                                   $ units
 "Param 43 Page 5-6",                   $ Maximum output current allo
 1, 180, 150,                           $ min, max, default data valu
 1, 1, 1, 0,                            $ mult, div, base, offset sca
 0, 0, 0, 0,                            $ mult, div, base, offset lin
 0;                                     $ decimal places
```

Configure an Explicit Client Tag

Complete the following steps to configure an Explicit Client tag.

1. Type a descriptive name for the tag.
2. Choose the data type for the tag.

   Let the PanelView terminal initiate the update.
3. Type the address of the device.
4. If the PanelView terminal sets the parameter, check the Write Tag box.
5. Type the number of bytes in the parameter.
6. Type the class, instance, and attribute numbers for the parameter.



Tag Name: Set_Drive_Accel

Data Type: Unsigned Integer

Messaging Type

○ I/O Slave

○ Explicit - Server

● Explicit - Client

Node Address:

□ Write Tag

Packet Bytes:

Bit Offset: 0

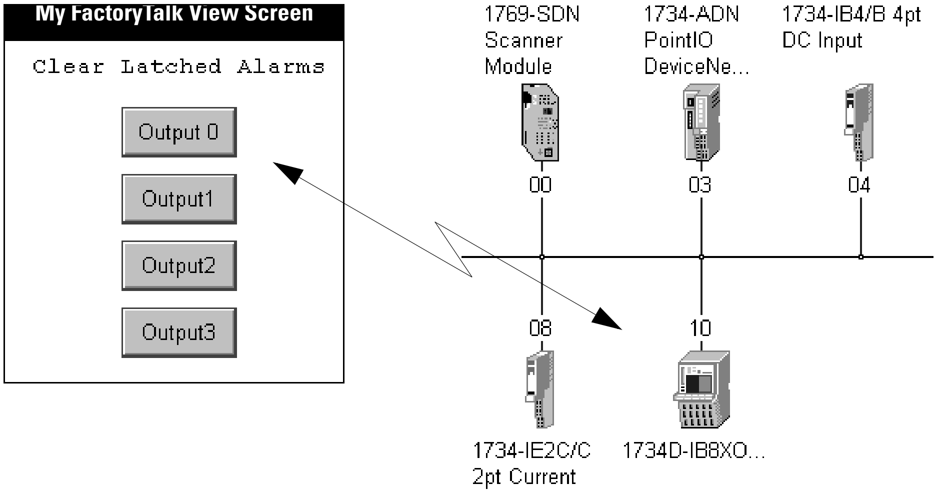
Class:

Instance:

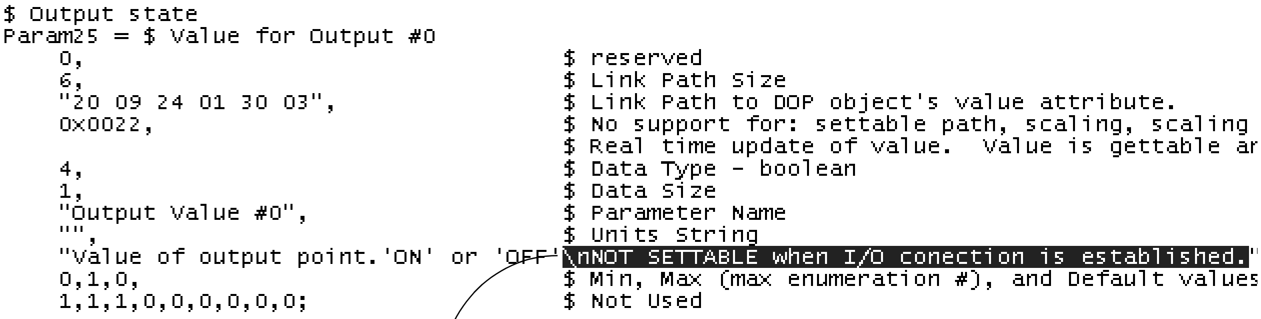
Attribute:

Communicate with a FactoryTalk View Project

This chapter describes how use a FactoryTalk View project to get or set a parameter of a DeviceNet communication module.



IMPORTANT Once you add a device to the scan list of a scanner, HMI software such as FactoryTalk View **cannot** write to (set) **some** parameters.



```
$ Output state
Param25 = $ Value for Output #0
    0,                                   $ reserved
    6,                                   $ Link Path Size
    "20 09 24 01 30 03",                 $ Link Path to DOP object's value attribute.
    0x0022,                              $ No support for: settable path, scaling, scaling
                                         $ Real time update of value.  Value is gettable ar
    4,                                   $ Data Type - boolean
    1,                                   $ Data Size
    "Output Value #0",                   $ Parameter Name
    "",                                  $ Units String
    "Value of output point.'ON' or 'OFF'\nNOT SETTABLE when I/O conection is established."
    0,1,0,                               $ Min, Max (max enumeration #), and Default values
    1,1,1,0,0,0,0,0,0;                   $ Not Used
```

Once this device is in the scan list of the scanner, a FactoryTalk View project cannot set this parameter.

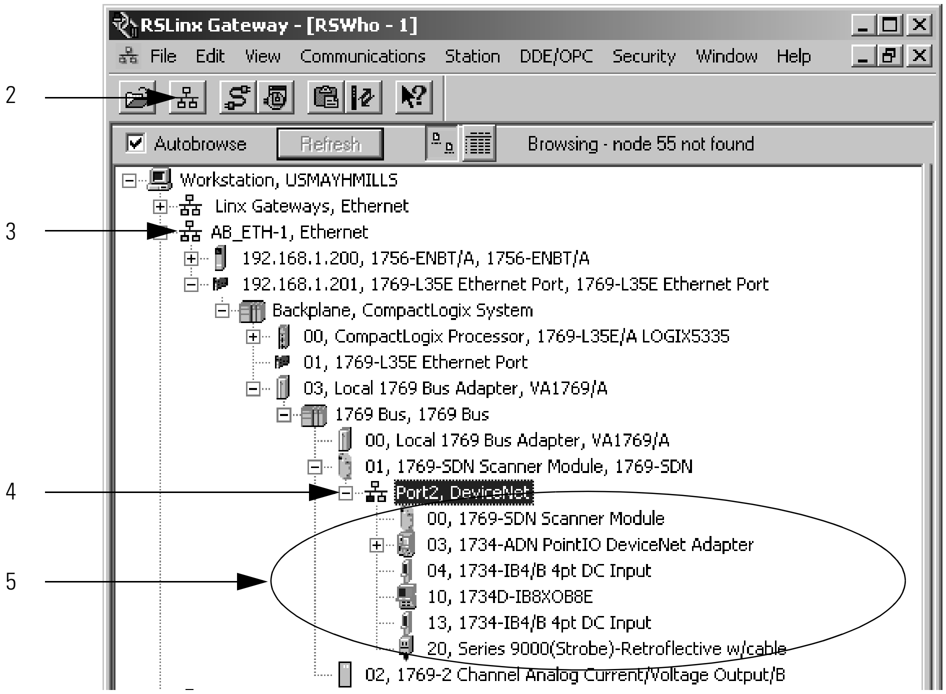
To access the DeviceNet network, either connect the computer with the FactoryTalk View application to any of the following networks:

- Same DeviceNet network as the desired device
- EtherNet/IP or ControlNet network and bridge communication to the DeviceNet network
  - Avoid bridging through a 1768 or 1769 CompactLogix controller, or DriveLogix controller. They have limited resources for bridging.
  - For the controllers mentioned in the previous bullet, use the I/O tags in the controller, if possible.

Before You Begin

Before you use this chapter, make sure that you can see all your devices on the DeviceNet network. Complete the following steps to see your DeviceNet network.

1. Start RSLinx communication software.
2. Browse the network.
3. Expand a driver that lets you access the DeviceNet network.
4. Select the DeviceNet network.
5. Verify that you see all the devices that are connected to the DeviceNet network.


2
3
4
5
RSLinx Gateway - [RSWho - 1]
File Edit View Communications Station DDE/OPC Security Window Help
☑ Autobrowse
Refresh

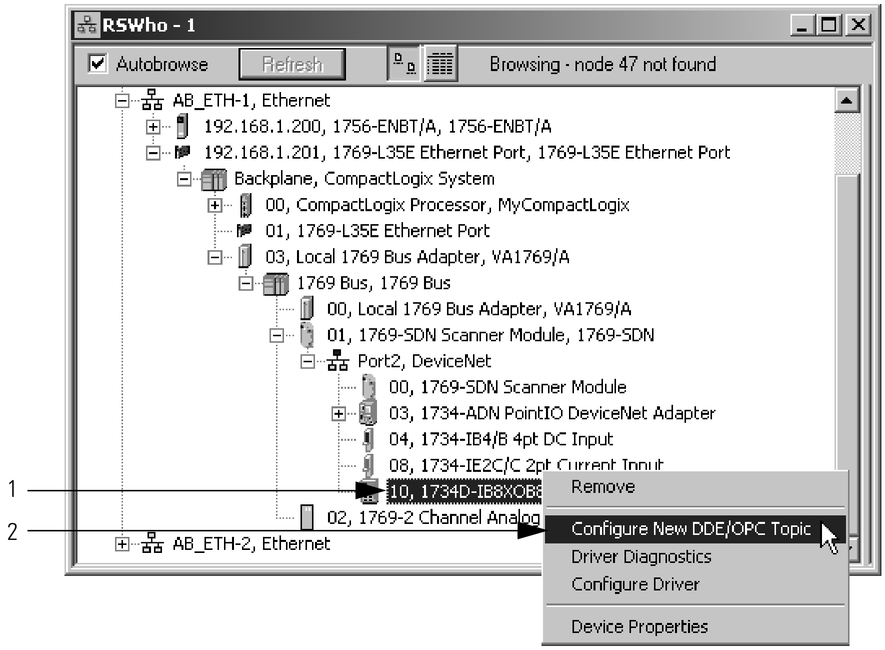
Browsing - node 55 not found

```
Workstation, USMAYHMILLS
  Linx Gateways, Ethernet
  AB_ETH-1, Ethernet
    192.168.1.200, 1756-ENBT/A, 1756-ENBT/A
    192.168.1.201, 1769-L35E Ethernet Port, 1769-L35E Ethernet Port
      Backplane, CompactLogix System
        00, CompactLogix Processor, 1769-L35E/A LOGIX5335
        01, 1769-L35E Ethernet Port
        03, Local 1769 Bus Adapter, VA1769/A
          1769 Bus, 1769 Bus
            00, Local 1769 Bus Adapter, VA1769/A
            01, 1769-SDN Scanner Module, 1769-SDN
              Port2, DeviceNet
                00, 1769-SDN Scanner Module
                03, 1734-ADN PointIO DeviceNet Adapter
                04, 1734-IB4/B 4pt DC Input
                10, 1734D-IB8XOB8E
                13, 1734-IB4/B 4pt DC Input
                20, Series 9000(Strobe)-Retroflective w/cable
            02, 1769-2 Channel Analog Current/Voltage Output/B
```

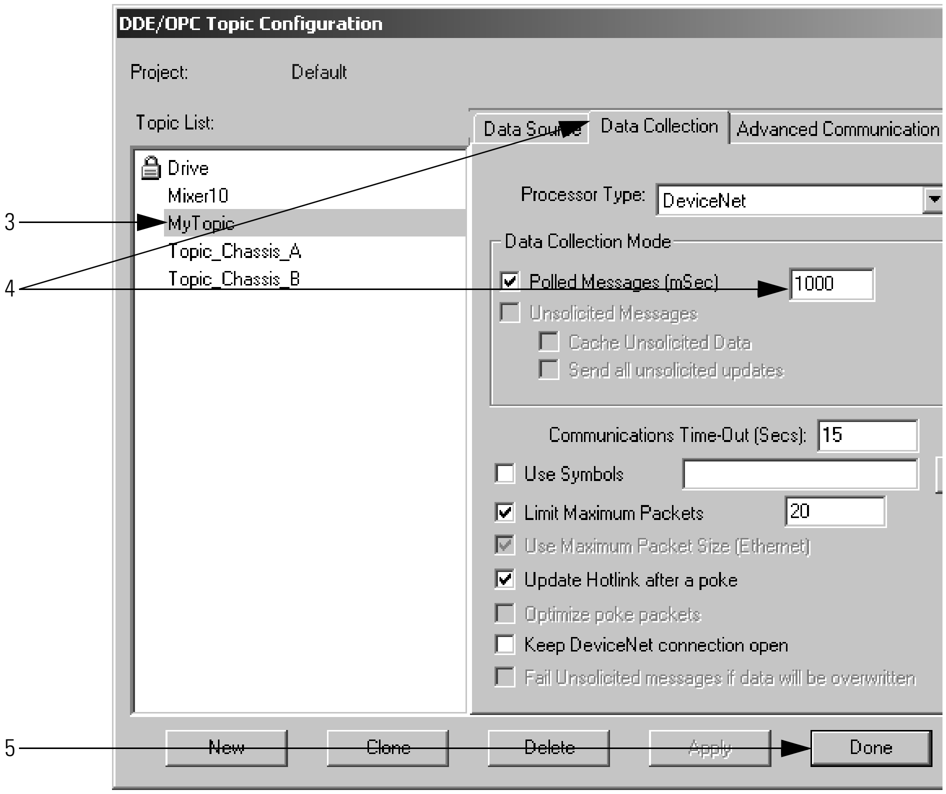
Create a Topic for the Device

Use RSLinx communication software to create a topic for the DeviceNet communication module that you want to access. Complete the following steps to create a topic for the device.

1. In RSLinx communication software, browse to the device that you want to access.
2. Right-click the device and choose Configure New DDE/OPC Topic.


RSWho - 1
☑ Autobrowse
Refresh
Browsing - node 47 not found
AB_ETH-1, Ethernet
192.168.1.200, 1756-ENBT/A, 1756-ENBT/A
192.168.1.201, 1769-L35E Ethernet Port, 1769-L35E Ethernet Port
Backplane, CompactLogix System
00, CompactLogix Processor, MyCompactLogix
01, 1769-L35E Ethernet Port
03, Local 1769 Bus Adapter, VA1769/A
1769 Bus, 1769 Bus
00, Local 1769 Bus Adapter, VA1769/A
01, 1769-SDN Scanner Module, 1769-SDN
Port2, DeviceNet
00, 1769-SDN Scanner Module
03, 1734-ADN PointIO DeviceNet Adapter
04, 1734-IB4/B 4pt DC Input
08, 1734-IE2C/C 2pt Current Input
10, 1734D-IB8XOB8
02, 1769-2 Channel Analog
AB_ETH-2, Ethernet
Remove
Configure New DDE/OPC Topic
Driver Diagnostics
Configure Driver
Device Properties
1
2

3. Type a name for the topic.
4. To change how often RSLinx communication software updates the tag, click the Data Collection tab and type a new poll period.
5. Click Done.
6. When prompted, click Yes to update the topic.



DDE/OPC Topic Configuration

Project: Default

Topic List:
- Drive
- Mixer10
- MyTopic
- Topic_Chassis_A
- Topic_Chassis_B

Processor Type: DeviceNet

Data Collection Mode

☑ Polled Messages (mSec) 1000

☐ Unsolicited Messages

☐ Cache Unsolicited Data

☐ Send all unsolicited updates

Communications Time-Out (Secs): 15

☐ Use Symbols

☑ Limit Maximum Packets 20

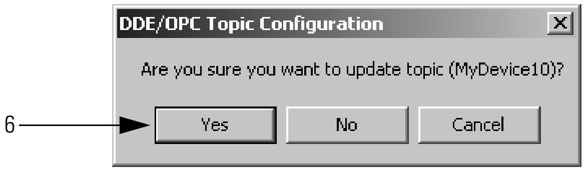
☑ Use Maximum Packet Size (Ethernet)

☑ Update Hotlink after a poke

☐ Optimize poke packets

☐ Keep DeviceNet connection open

☐ Fail Unsolicited messages if data will be overwritten



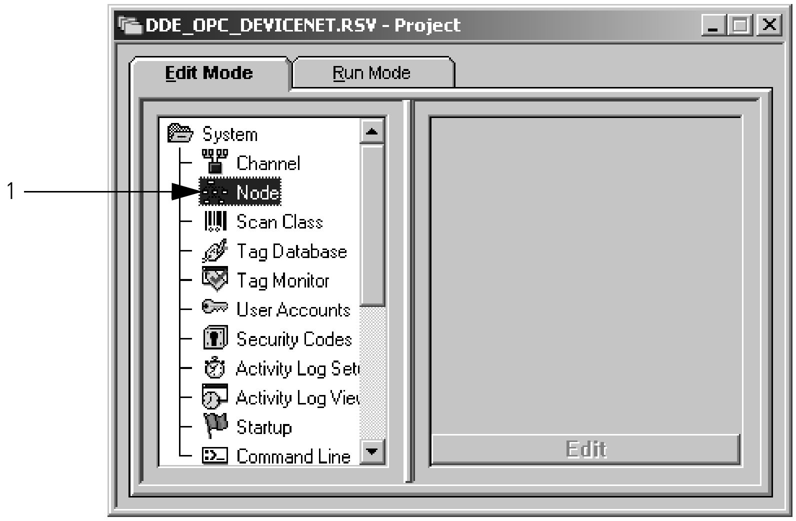
DDE/OPC Topic Configuration

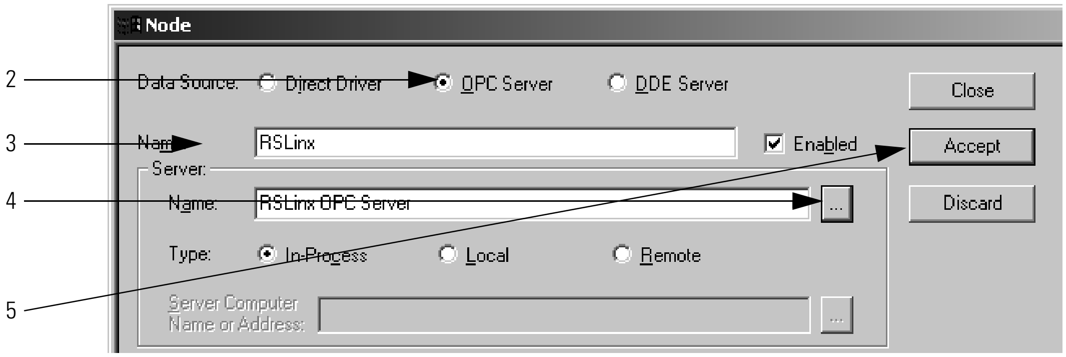
Are you sure you want to update topic (MyDevice10)?

Create a Node

In the FactoryTalk View project, create a node for your RSLinx topics. Complete the following steps to create a node.

1. Open the list of nodes for the project.
2. Choose OPC Server.
3. Type a name for the node.
4. Select RSLinx.
5. Click Accept.





Data Source: ○ Direct Driver ● OPC Server ○ DDE Server

Name: RSLinx ☑ Enabled

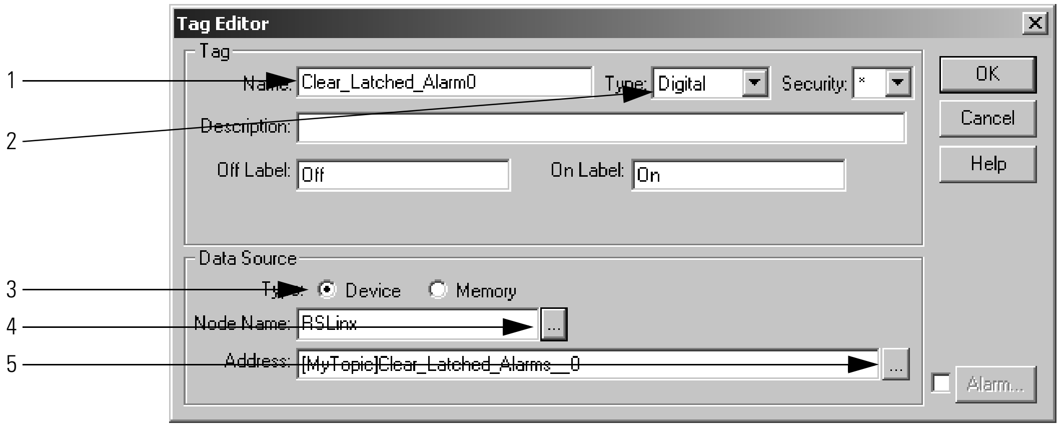
Name: RSLinx OPC Server

Type: ● In-Process ○ Local ○ Remote

Create a Tag for the Parameter

Complete the following steps to create a tag for the parameter in FactoryTalk View software.

1. Type the name of the tag.
2. Choose the type of tag.
3. Choose Device.
4. Select the node that contains the topic for the device.
5. Open the address browser.
6. Browse to the offline list of tags for the topic, that is, device.
7. Select the parameter and click OK.



Tag

Name: Clear_Latched_Alarm0 Type: Digital Security: *

Description:

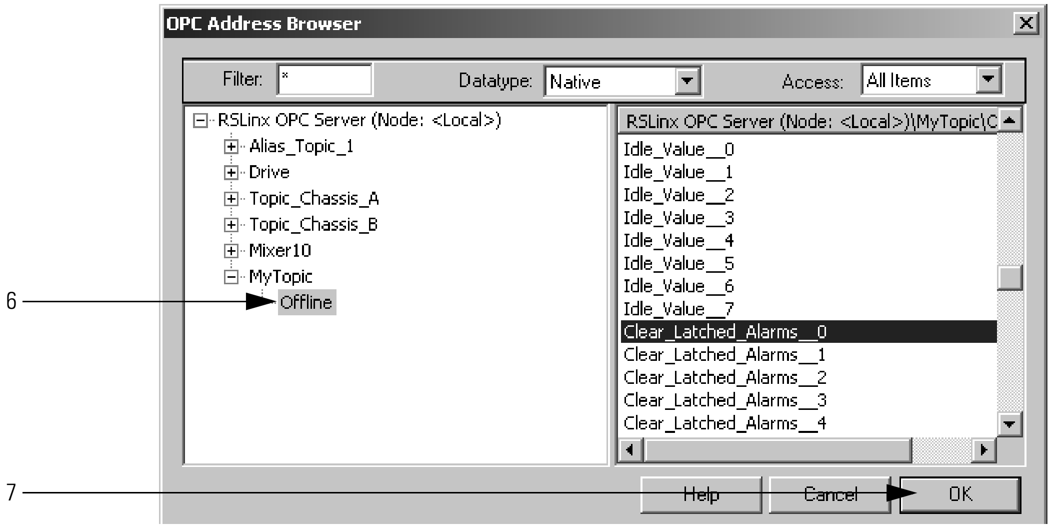
Off Label: Off On Label: On

Data Source

Type: ● Device ○ Memory

Node Name: RSLinx

Address: [MyTopic]Clear_Latched_Alarms__0



Filter: * Datatype: Native Access: All Items

RSLinx OPC Server (Node: <Local>)
- Alias_Topic_1
- Drive
- Topic_Chassis_A
- Topic_Chassis_B
- Mixer10
- MyTopic
  - Offline

RSLinx OPC Server (Node: <Local>)\MyTopic\C

Idle_Value__0
Idle_Value__1
Idle_Value__2
Idle_Value__3
Idle_Value__4
Idle_Value__5
Idle_Value__6
Idle_Value__7
Clear_Latched_Alarms__0
Clear_Latched_Alarms__1
Clear_Latched_Alarms__2
Clear_Latched_Alarms__3
Clear_Latched_Alarms__4

Notes:

Chapter 11

Tune the Performance of a DeviceNet Network

This chapter describes how to improve the performance of the network.

Topic	Page
Factors that Affect Performance	162
Change the Configuration of Your Network	165

As you configure and program the network, use the default settings whenever possible. Once the network is running, determine if you need to improve performance.

To improve the performance of the network, consider the information in the following table.

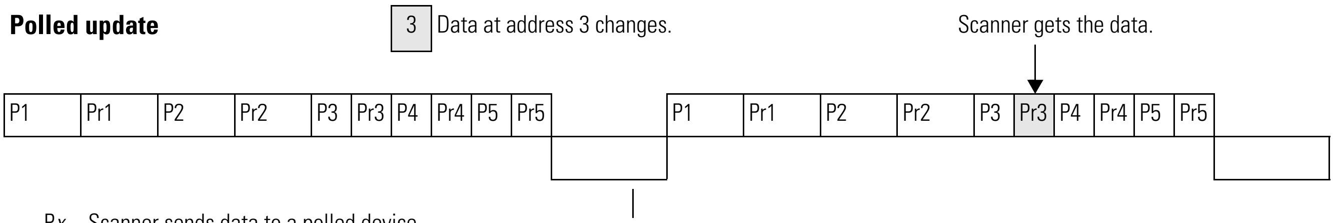
If	Then
A specific device requires a faster update	Change the I/O parameters of the device to change of state (COS).
An analog device does **either** of the following: • Changes slower than the scan cycle • Requires a repeatable update period, such as for PID calculations	Change the I/O parameters of the device to cyclic.
Multiple devices are input only **and** I/O parameters are currently set to polled with an input size less than or equal to 8 bytes	For each of those devices, change their I/O parameters to strobed.
Two or more devices send or receive large amounts of data, such as the PanelView operator terminal	• For each of those devices, set their I/O parameters to polled with a poll rate = background. • For the scanner, set the poll ratio = 2. Increase the poll ratio if needed.
Communication intermittently stops (status code 78) with a device that sends or receives large amounts of data, such as the PanelView operator terminal, **and** has the I/O parameters currently set to polled	Increase the interscan delay.

Factors that Affect Performance

The following example shows how different I/O or network parameters affect the performance of the network.

Scan Cycle

Polled update

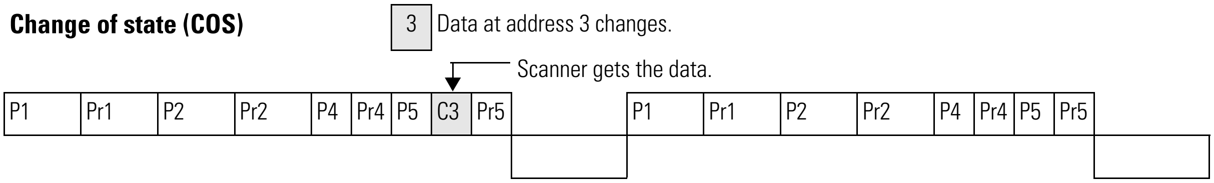


P*x*—Scanner sends data to a polled device.
Pr*x*—Polled device sends input data to scanner.
x is the address of a device.

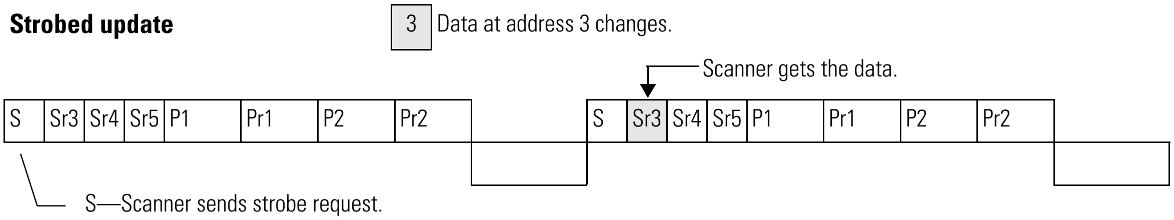
Interscan delay

- Scanner continues to get polled data from slower responders or larger devices.
- Communication with software, such as RSLinx and RSNetWorx occurs, for uploading, downloading, and browsing, for example.

Change of state (COS)

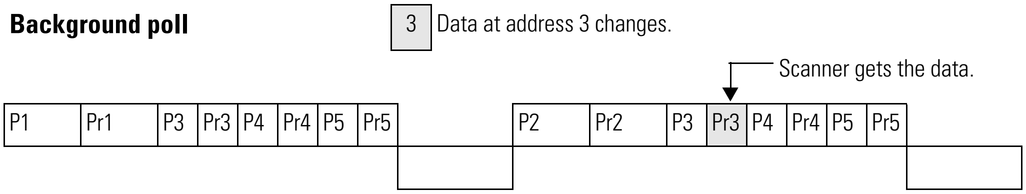


Strobed update

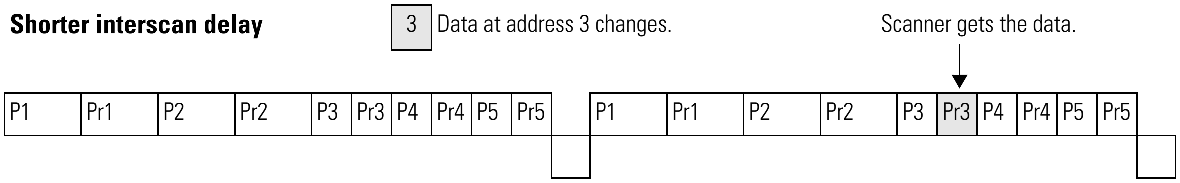


S—Scanner sends strobe request.
Sr*x*—Strobed devices send input data to scanner.
x is the address of a device.

Background poll



Shorter interscan delay



I/O Parameters of Each Device

The type of connection (message) that you configure for a device determines when data transfers between the device and the scanner. Consider the following points when you configure the type of connection:

- Each device has a default connection type. This is a good starting point.
- Some devices may not offer all connection (message) types.

The following table describes the different types of connections (messages) that you can configure for a device.

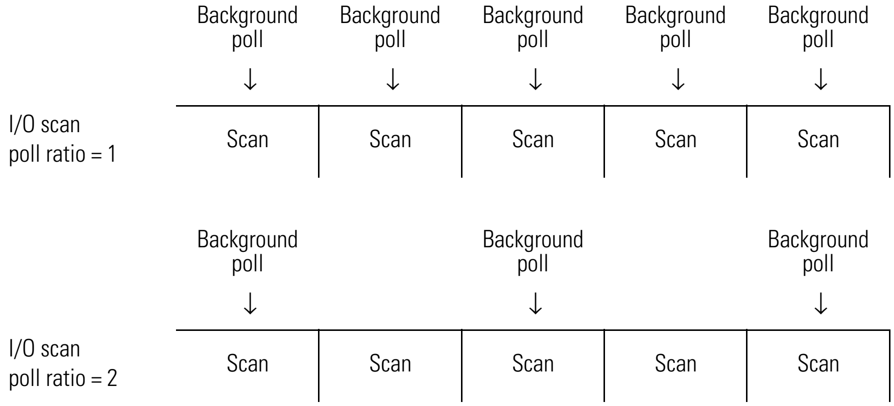
Connection (Message) Type	Description
Cyclic	Data transfers at the period that you specify. The default range is 48…32,000 milliseconds.
Change of state (COS)	Both the scanner and the device sends data whenever the data changes. You also specify a heartbeat period for the connection. • If the data does not change within the heartbeat period, the scanner or device sends the data at the end of the period. • This lets both the scanner and device know that the other is still operational.
Strobed	The scanner sends a single strobed request to solicit data from the strobed devices. • The request is 64-bits long (1 bit for each node). • In response to the request, each device that is configured for a strobed connection sends its data up to 8 bytes.
Polled	A point-to-point data transfer that occurs every I/O scan or as a ratio of the I/O scan (background). • At the specified poll rate (every scan or background), the scanner sends data to a polled device up to 255 bytes. The data is either output data for the device or a request for input data from the device. • If the polled device gets a request for input data, it sends its input data up to 255 bytes.

Background Poll

The foreground to background poll ratio lets you adjust how often the scanner polls certain devices for their data. In general, use the default values. Change them only if you need to tune the performance of your system.

Parameter	Description	Default Setting
Poll rate	• Applies to a device with a polled connection. • Defines whether the scanner polls the device every I/O scan (foreground) or as a ratio of the I/O scan (background).	Every scan
Foreground to background poll ratio	• Applies to devices with a polled connection that is configured for a background poll rate. • Determines how often the devices are polled. • By default, the scanner performs background polls every scan (poll ratio = 1).	1

The following diagram shows the effect of a change to the poll ratio.



IMPORTANT

When using a foreground to background poll ratio other than 1, the total network time-out value of the EPR may need to be increased so the background devices do not time out.

The expected packet rate (EPR) defaults to 75, which is then multiplied by 4 ms to get a 300 ms timeout for a polled/strobed I/O connection.

Interscan Delay

The interscan delay determines how long the scanner waits before it starts another I/O scan. Follow these guidelines:

- In general, leave the interscan delay at its default value. Change it only if you need to tune the performance of your system.
- Keep the interscan delay ≥ 5 ms. Otherwise, you may have trouble accessing the network.

Parameter	Description	Default Setting
Interscan delay	• Last segment of the I/O scan. • Starts after the last packet of the poll to the last node in the scanner's scan list. • Provides time for larger devices and slower responders to return their polled data. • Provides time for software, such as RSLinx and RSNetWorx, to access the network for uploading, downloading, and browsing, for example. • Scanner waits for the interscan delay before it strobes or polls devices again. • A shorter interscan delay may improve the update time of strobed or polled data.	10 ms

Change the Configuration of Your Network

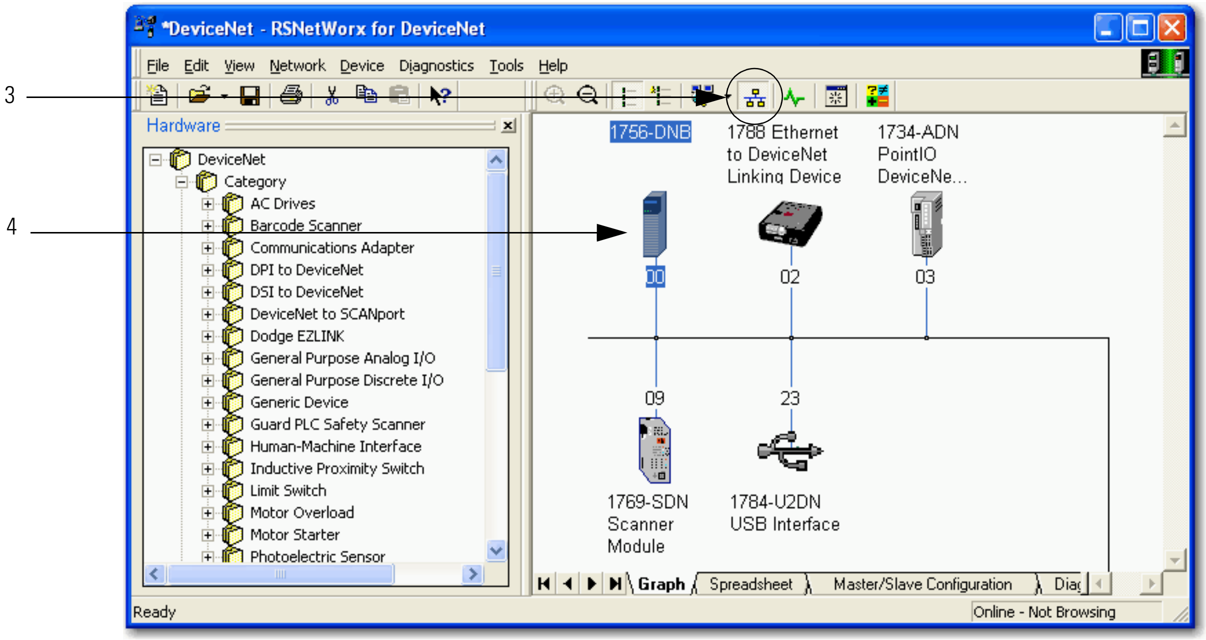
Complete these tasks to change the configuration of the network with RSNetWorx for DeviceNet software:

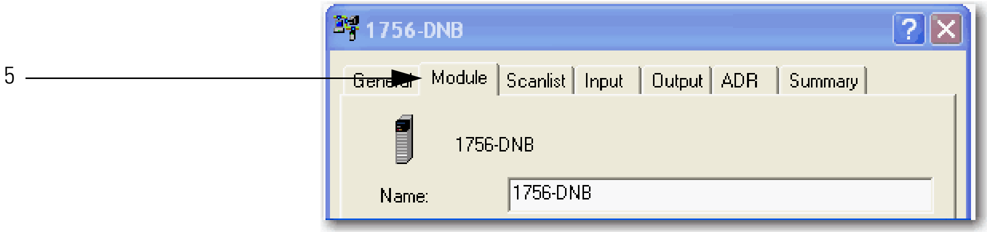
- Upload the Current Configuration of the Scanner
- Set the Interscan Delay and Poll Ratio
- Set the I/O Parameters of a Device
- Download the Configuration to the Scanner
- Save the Configuration File

Upload the Current Configuration of the Scanner

Complete these steps to upload the current configuration.

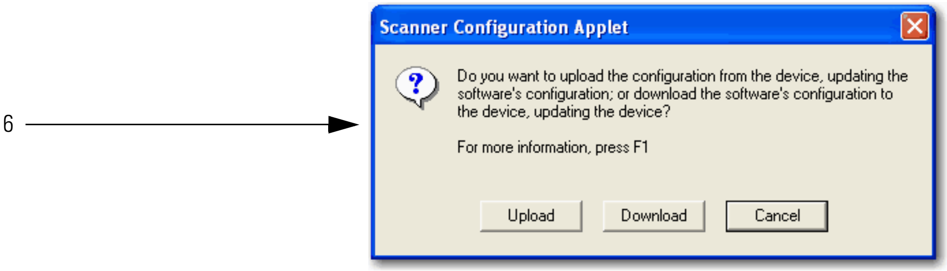
1. Start RSNetWorx for DeviceNet software.
2. If necessary, open the file for the network.
3. Go online.
4. Double-click the scanner.
5. Click the Module tab.
6. Click Upload from Scanner.
7. When prompted, upload the configuration from the scanner.





1756-DNB

Name: 1756-DNB



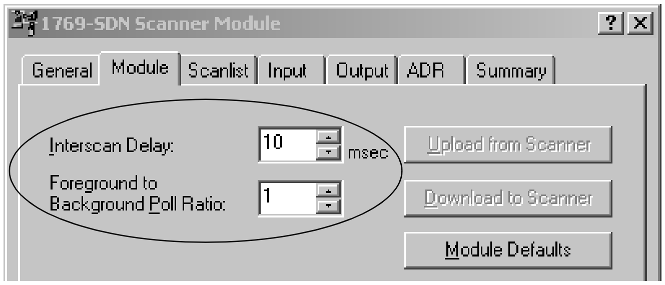
Do you want to upload the configuration from the device, updating the software's configuration; or download the software's configuration to the device, updating the device?

For more information, press F1

Set the Interscan Delay and Poll Ratio

Change the parameters shown below if needed.

- For information on changing the Interscan Delay parameter, refer to page 165.
- For information on changing the Foreground to Background Poll Ratio parameter, refer to page 164.



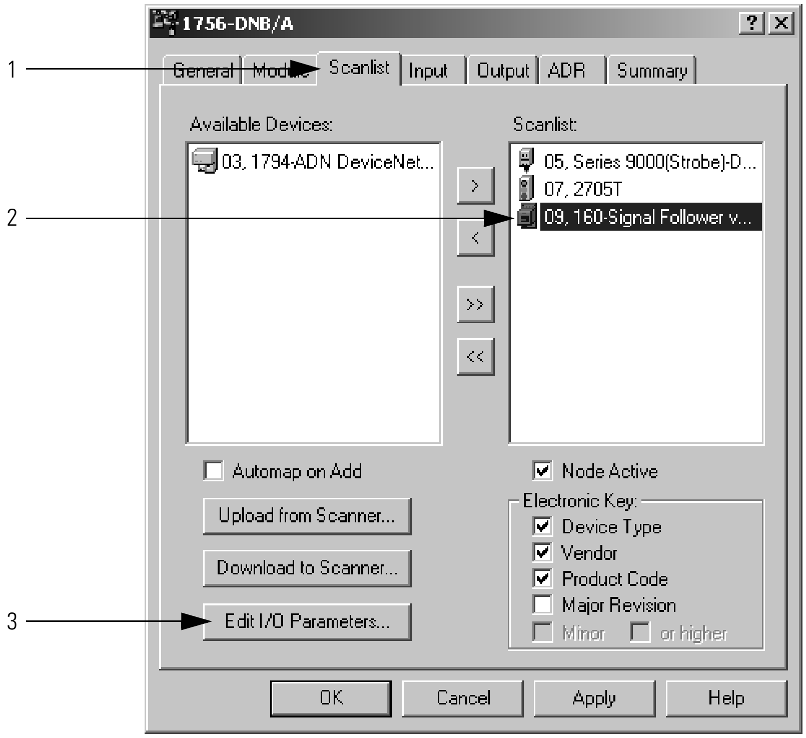
Interscan Delay: 10 msec

Foreground to Background Poll Ratio: 1

Set the I/O Parameters of a Device

Complete these steps to set the I/O parameters of a device.

1. Click the Scanlist tab.
2. Select the device.
3. Click Edit I/O Parameters to display the Edit I/O Parameters dialog box.



Available Devices:
03, 1794-ADN DeviceNet...

Scanlist:
05, Series 9000(Strobe)-D...
07, 2705T
09, 160-Signal Follower v...

□ Automap on Add

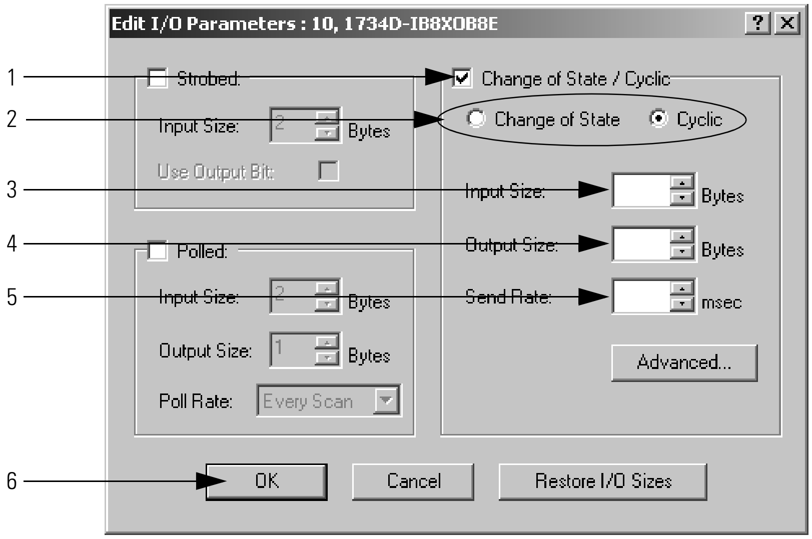
☑ Node Active

Electronic Key:
☑ Device Type
☑ Vendor
☑ Product Code
□ Major Revision
□ Minor □ or higher

Change of State or Cyclic Transfer

Complete these steps to configure the I/O parameters for the Change of State setting.

1. Check Change of State/Cyclic checkbox.
2. Click the Change of State or Cyclic option.
3. Enter the number of bytes that the devices sends to the controller.
4. Enter the number of bytes that the controller sends to the device.
5. For a cyclic update, enter the period of the update.
6. Click OK.



Edit I/O Parameters : 10, 1734D-IB8XOB8E

□ Strobed:
Input Size: 2 Bytes
Use Output Bit: □

□ Polled:
Input Size: 2 Bytes
Output Size: 1 Bytes
Poll Rate: Every Scan

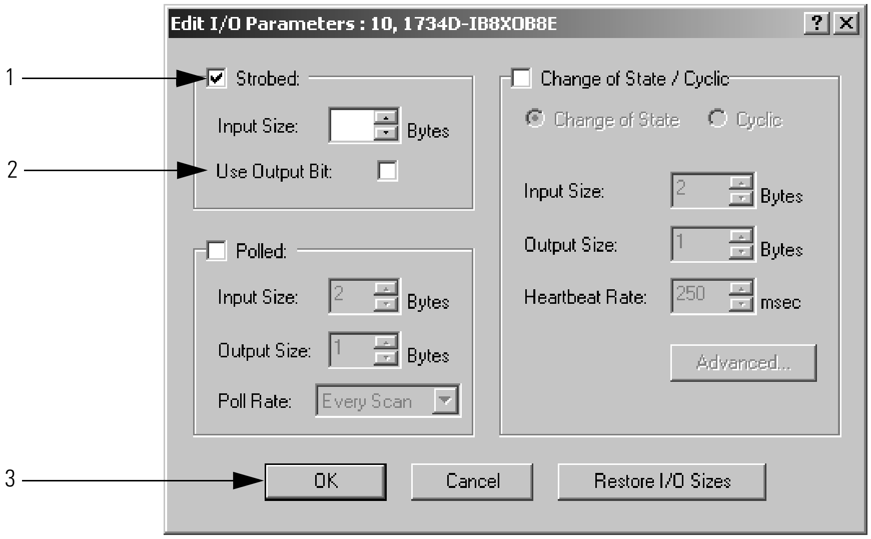
☑ Change of State / Cyclic
○ Change of State ● Cyclic
Input Size: Bytes
Output Size: Bytes
Send Rate: msec

Strobed Transfer

Complete these steps to configure the I/O parameters for the Strobed Transfer setting.

1. Check Strobed.
2. If the single bit being sent to the strobed device needs to be accessed by the Logix controller, check Use Output Bit.

   This lets you map the bit into the I/O data being transferred with the controller.
3. Enter the number of bytes that the device sends to the controller.
4. Click OK.



Edit I/O Parameters : 10, 1734D-IB8XOB8E

☑ Strobed:
Input Size: Bytes
Use Output Bit: □

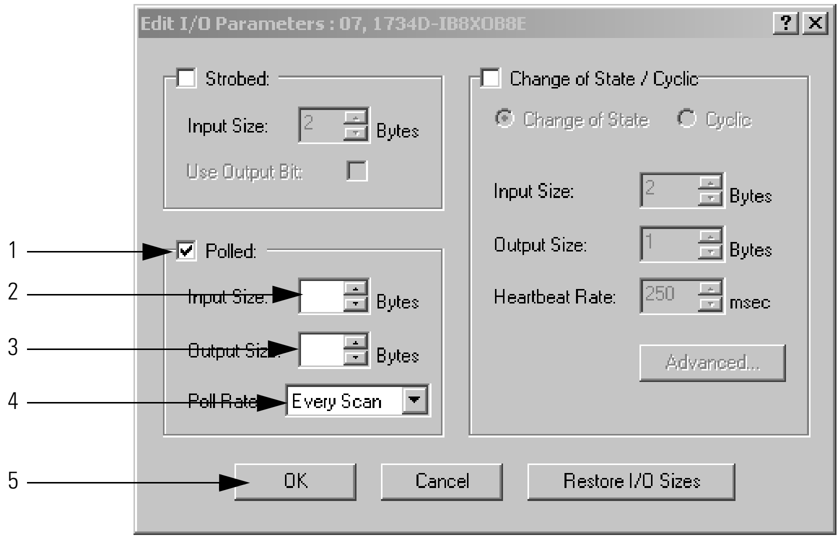
□ Polled:
Input Size: 2 Bytes
Output Size: 1 Bytes
Poll Rate: Every Scan

□ Change of State / Cyclic
● Change of State ○ Cyclic
Input Size: 2 Bytes
Output Size: 1 Bytes
Heartbeat Rate: 250 msec

Polled Transfer

Complete these steps to configure the I/O parameters for the Polled setting.

1. Check Polled.
2. Enter the number of bytes that the device sends to the controller.
3. Enter the number of bytes that the controller sends to the device.
4. Choose whether to poll the device every scan or in the background.
5. Click OK.



Edit I/O Parameters : 07, 1734D-IB8XOB8E

☐ Strobed:
Input Size: 2 Bytes
Use Output Bit: ☐

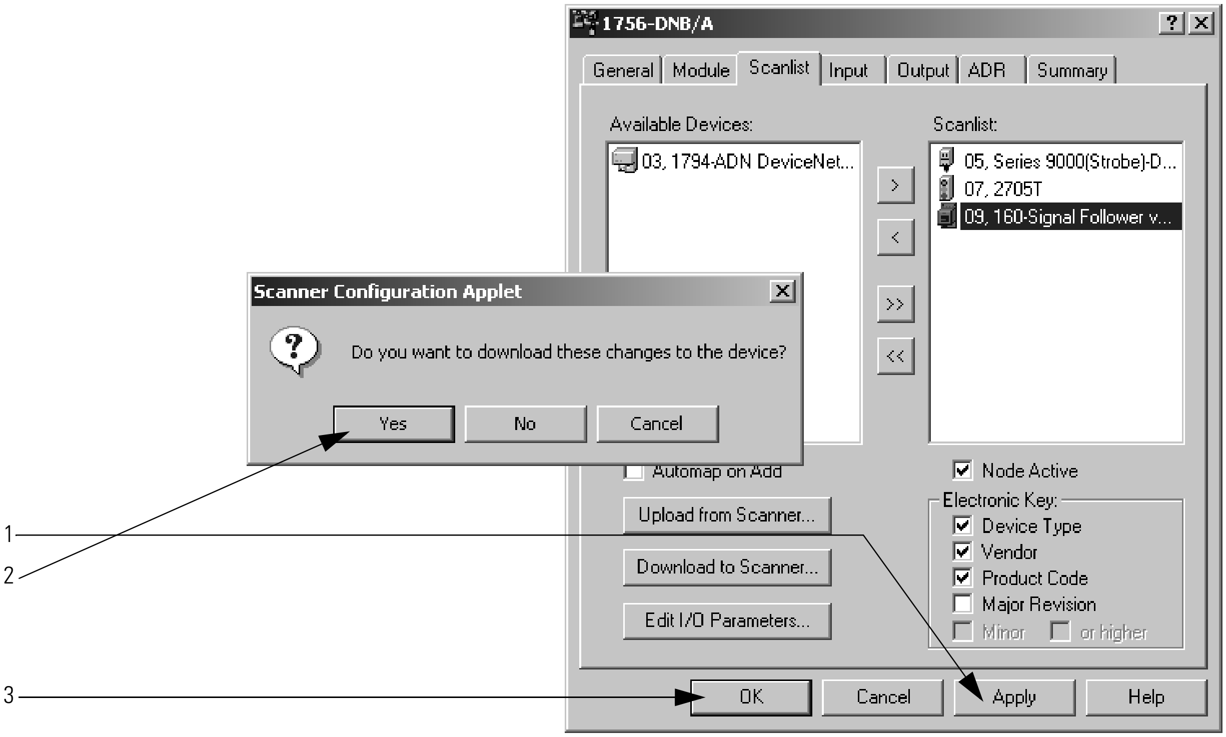
☑ Polled:
Input Size: Bytes
Output Size: Bytes
Poll Rate: Every Scan

☐ Change of State / Cyclic
● Change of State ○ Cyclic
Input Size: 2 Bytes
Output Size: 1 Bytes
Heartbeat Rate: 250 msec

Download the Configuration to the Scanner

Complete these steps to download the configuration to the scanner.

1. Click Apply.
2. When prompted, click Yes to download the changes.
3. Click OK.



1756-DNB/A

Available Devices:
03, 1794-ADN DeviceNet...

Scanlist:
05, Series 9000(Strobe)-D...
07, 2705T
09, 160-Signal Follower v...

□ Automap on Add

☑ Node Active

Electronic Key:
☑ Device Type
☑ Vendor
☑ Product Code
□ Major Revision
□ Minor □ or higher

Scanner Configuration Applet

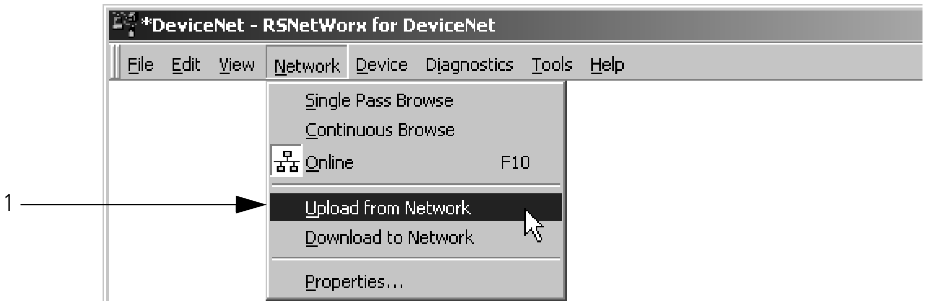
Do you want to download these changes to the device?

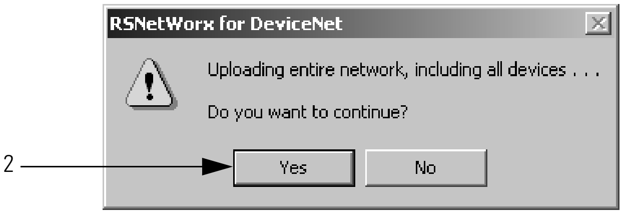
Save the Configuration File

After you make a change to the network, upload the entire network and save the file. This makes sure that the offline configuration file matches the network.

Complete these steps to save the configuration file.

1. From the Network menu, choose Upload from Network.
2. When prompted, click Yes to upload the entire network.
3. Save the file.





Uploading entire network, including all devices . . .

Do you want to continue?

Automate the Replacement of a Failed Device

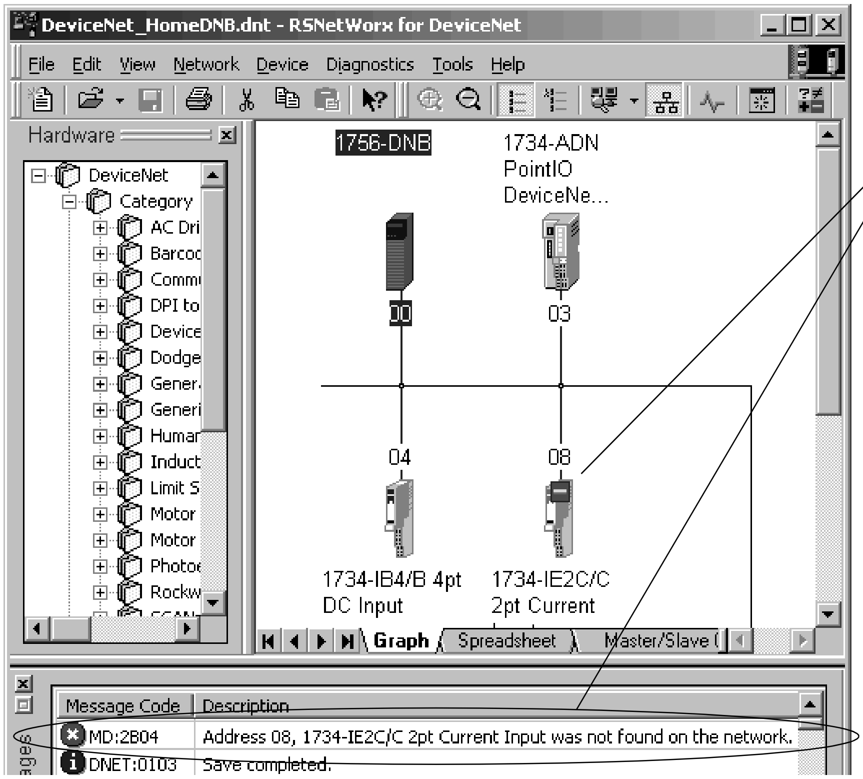
This chapter describes how to reduce the time it takes to replace a failed device.

Topic	Page
Automatic Device Recovery	173
Set Up Automatic Device Recovery	175

Automatic Device Recovery

To reduce system downtime if a device fails, use the automatic device recovery (ADR) option. With ADR, you **do not** have to use any software tools to get a replacement device configured and online.

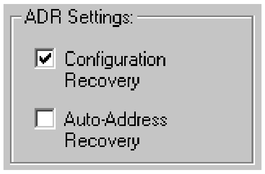
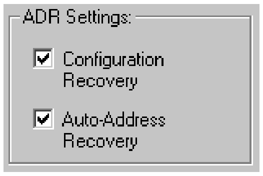
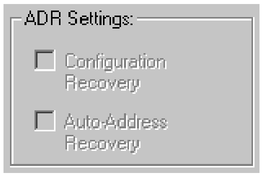
IMPORTANT Some devices **do not** support ADR.



Message Code	Description
MD:2B04	Address 08, 1734-IE2C/C 2pt Current Input was not found on the network.
DNET:0103	Save completed.

With ADR, the scanner automatically configures a replacement part. If the address of the device is set via software, the scanner also sets the address of the replacement device.

You configure ADR on a device-by-device basis. You can set up the following ADR settings for each device.

If you want to	And	Then select this ADR option for the device
Automatically configure a replacement device that matches the electronic key of a failed device	Manually change the address of the replacement device	 ADR Settings: ☑ Configuration Recovery ☐ Auto-Address Recovery
	Automatically set the address of the replacement device to the default address (63) of the failed device via software	 ADR Settings: ☑ Configuration Recovery ☑ Auto-Address Recovery
Manually configure a replacement device	→	 ADR Settings: ☐ Configuration Recovery ☐ Auto-Address Recovery

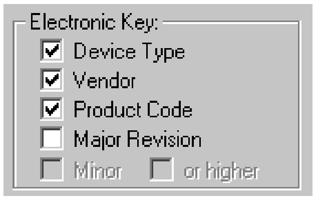
ATTENTION: If a DeviceNet network has more than one scanner, enable Auto-Address Recovery for only one scanner. If more than one scanner is configured for Auto-Address Recovery, there is no way to determine which scanner will recognize a newly-inserted device on the DeviceNet network.

Set Up Automatic Device Recovery

Complete these tasks to set up ADR for a device:

- Choose an Electronic Key Level for a Device
- Update Your Network Configuration File
- Define the Electronic Key
- Enable Auto-Address Recovery for the Scanner
- Set the ADR Settings for the Device
- Download the Changes to the Scanner
- Upload and Save the Configuration File

Choose an Electronic Key Level for a Device



Electronic Key:

☑ Device Type

☑ Vendor

☑ Product Code

☐ Major Revision

☐ Minor ☐ or higher

Use the electronic key options to define how closely a replacement device must match a failed device before the scanner applies ADR. The scanner configures/addresses only a device that meets the checkbox items that are checked in the electronic key of the failed device.

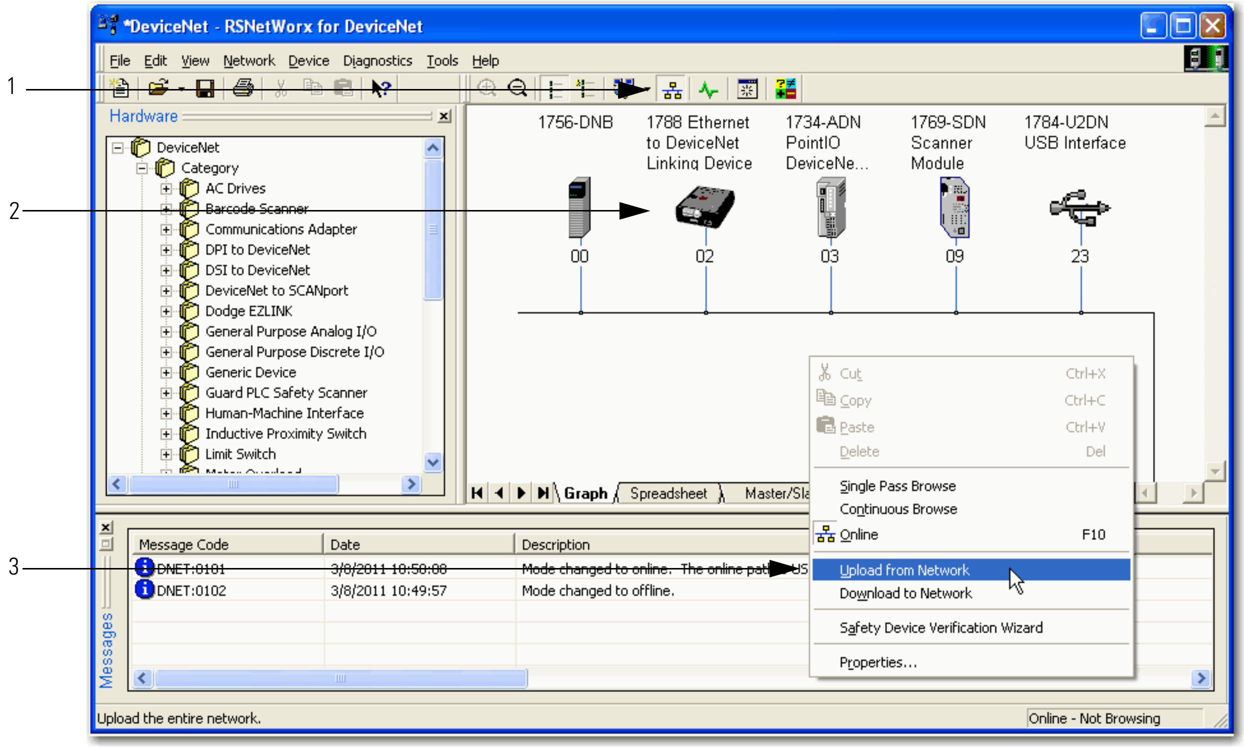
If multiple devices with the same electronic key fail at the same time, the scanner disables auto-address recovery for those devices. This prevents the scanner from changing the address of the wrong device.

Update Your Network Configuration File

When you set up ADR for a device, RSNetWorx for DeviceNet software reads the configuration for the device from the configuration file and stores it in the scanner. Before you set up ADR for a device, make sure the configuration file is up-to-date.

Complete these steps to update the network file.

1. Go online.
2. Configure the device.
3. Right-click and upload the entire network.
4. Save the network configuration.

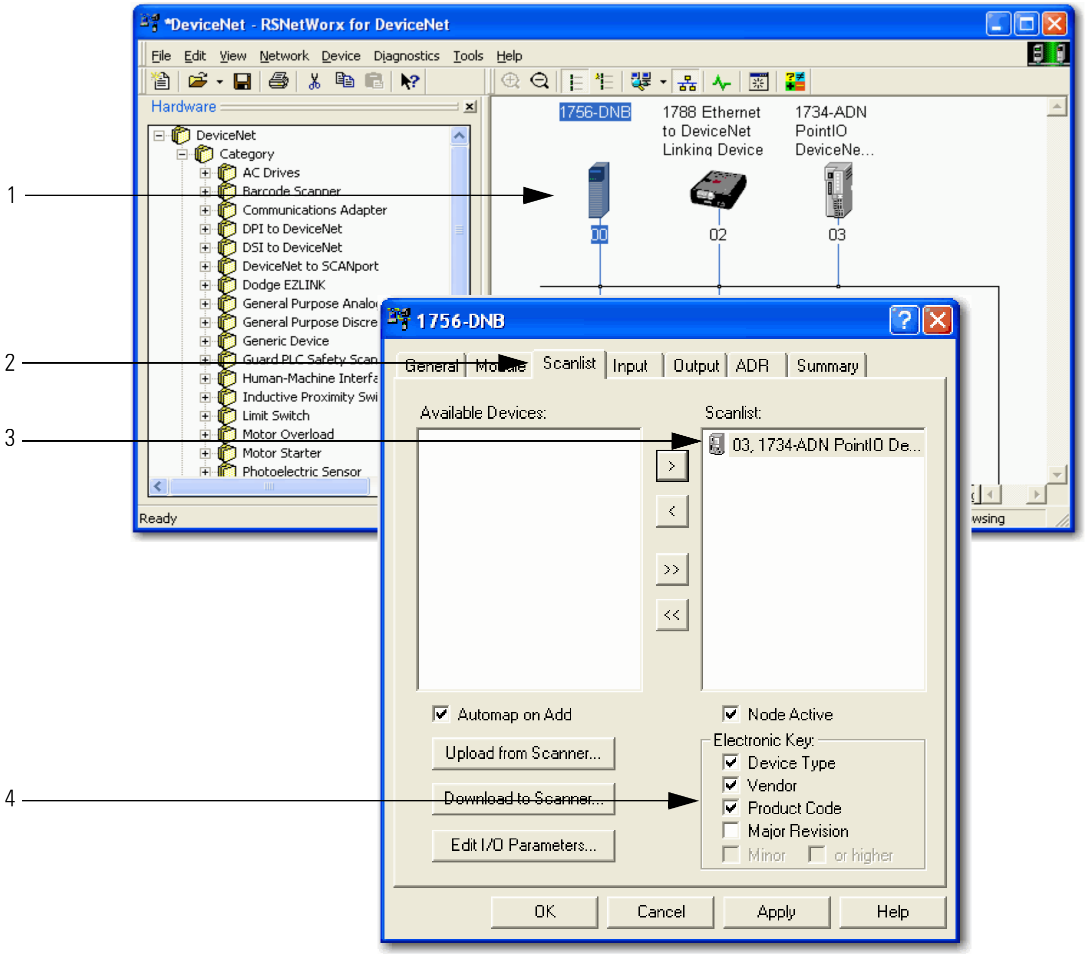


Message Code	Date	Description
DNET:0101	3/8/2011 10:50:08	Mode changed to online. The online pat...US
DNET:0102	3/8/2011 10:49:57	Mode changed to offline.

Define the Electronic Key

Complete the following steps to define the electronic key.

1. Double-click the scanner.
2. Click the Scanlist tab.
3. Select the device.
4. Check the items that must match before a replacement device receives the configuration/address of the selected device.



Scanlist: 03, 1734-ADN PointIO De...

☑ Automap on Add

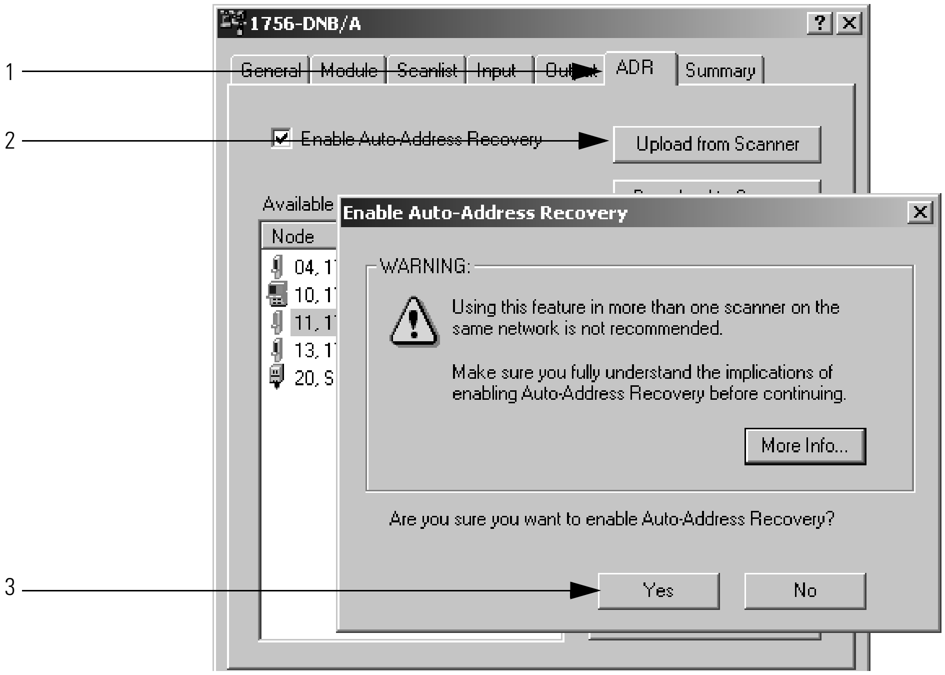
☑ Node Active

Electronic Key:
☑ Device Type
☑ Vendor
☑ Product Code
☐ Major Revision
☐ Minor ☐ or higher

Enable Auto-Address Recovery for the Scanner

Complete these steps to enable Auto-Address Recovery.

1. Click the ADR tab.
2. Make sure Enable Auto-Address Recovery is checked.
3. When prompted, click Yes to enable Auto-Address Recovery.



Using this feature in more than one scanner on the same network is not recommended.

Make sure you fully understand the implications of enabling Auto-Address Recovery before continuing.

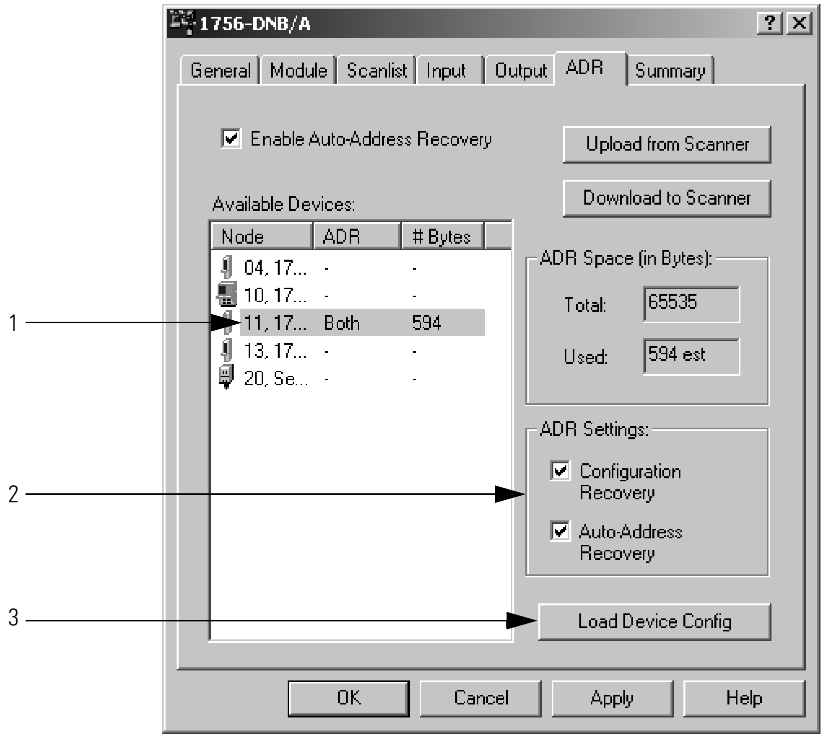
Are you sure you want to enable Auto-Address Recovery?

Set the ADR Settings for the Device

Complete these steps to set the ADR settings for the device.

1. Select the device.
2. Select the ADR settings for the device.
3. Read the configuration data of the device into the ADR configuration of the RSNetWorx project.

IMPORTANT	Make sure you upload all changes made to a device online into the RSNetWorx project before you click the Load Device Config button.



☑ Enable Auto-Address Recovery

Available Devices:

Node	ADR	# Bytes
04, 17...	-	-
10, 17...	-	-
11, 17...	Both	594
13, 17...	-	-
20, Se...	-	-

ADR Space (in Bytes):

Total: 65535

Used: 594 est

ADR Settings:

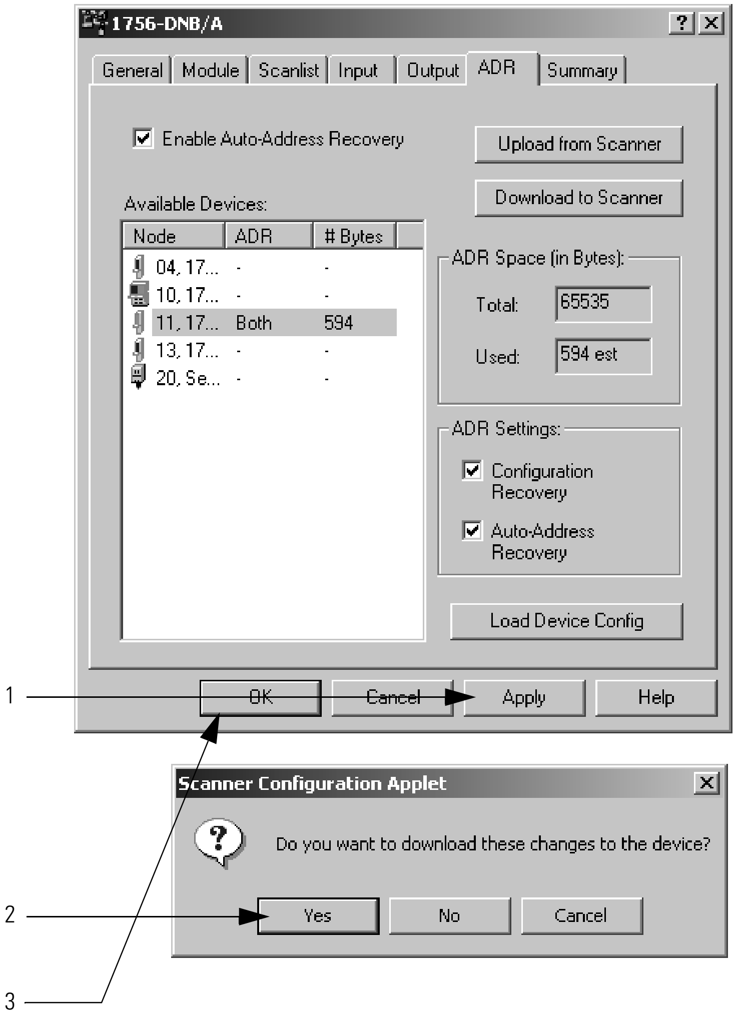
☑ Configuration Recovery

☑ Auto-Address Recovery

Download the Changes to the Scanner

Complete these steps to download changes to the scanner.

1. Click Apply.
2. When prompted, click Yes to download the changes.
3. Click OK.



1756-DNB/A

☑ Enable Auto-Address Recovery

Available Devices:

Node	ADR	# Bytes
04, 17...	-	-
10, 17...	-	-
11, 17...	Both	594
13, 17...	-	-
20, Se...	-	-

ADR Space (in Bytes):

Total: 65535

Used: 594 est

ADR Settings:

☑ Configuration Recovery

☑ Auto-Address Recovery

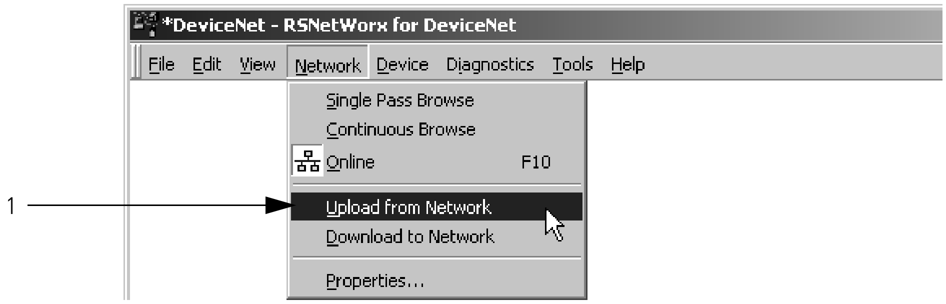
Scanner Configuration Applet

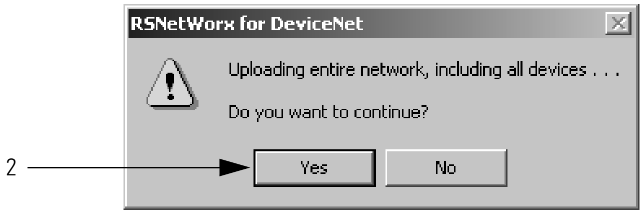
Do you want to download these changes to the device?

Upload and Save the Configuration File

Complete these steps to upload and save the configuration file.

1. Choose Network>Upload from Network.
2. When prompted, click Yes to upload the entire network.
3. Save the file.





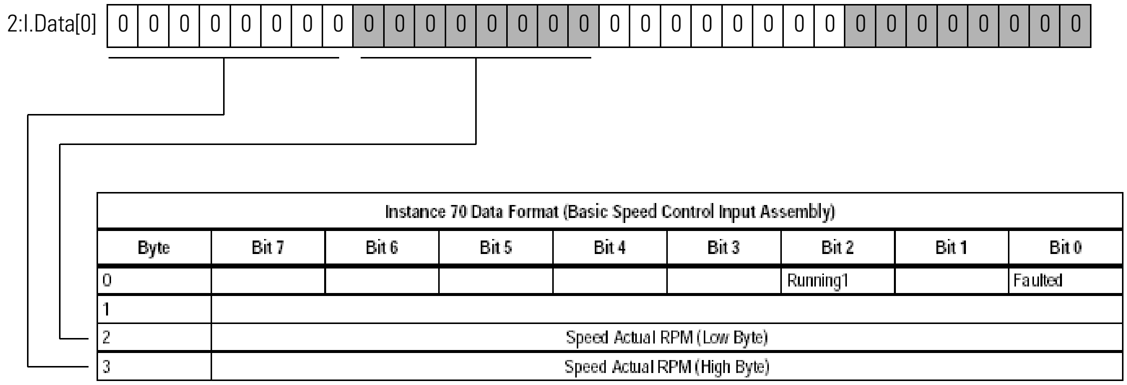
Uploading entire network, including all devices . . .

Do you want to continue?

Notes:

Map the Memory Location with Advanced Mapping

Sometimes, an input or output value for a device may end up encapsulated within a larger tag. For example, a speed value may end up as the upper 16 bits of a DINT element in the scanner. To access the value, you would have to use additional programming.



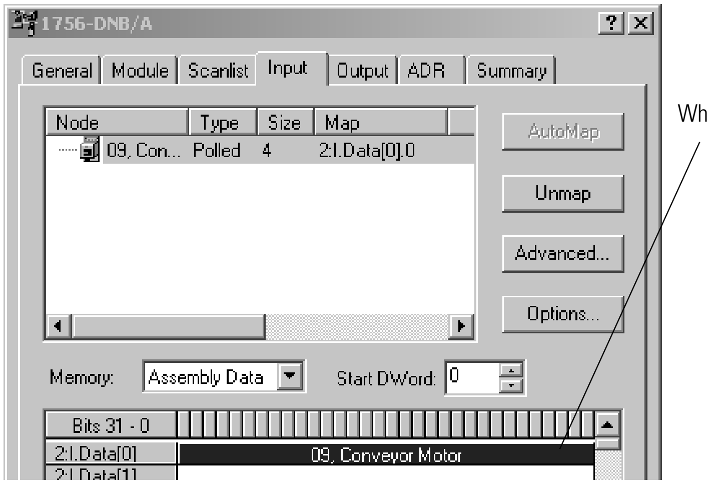
2:I.Data[0]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Data map for
Bulletin 160
AC drive

Instance 70 Data Format (Basic Speed Control Input Assembly)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0						Running1		Faulted
1								
2	Speed Actual RPM (Low Byte)							
3	Speed Actual RPM (High Byte)							

To make your programming easier, re-map the value to its own tag within the data array of the scanner. This lets you access the value without additional programming.



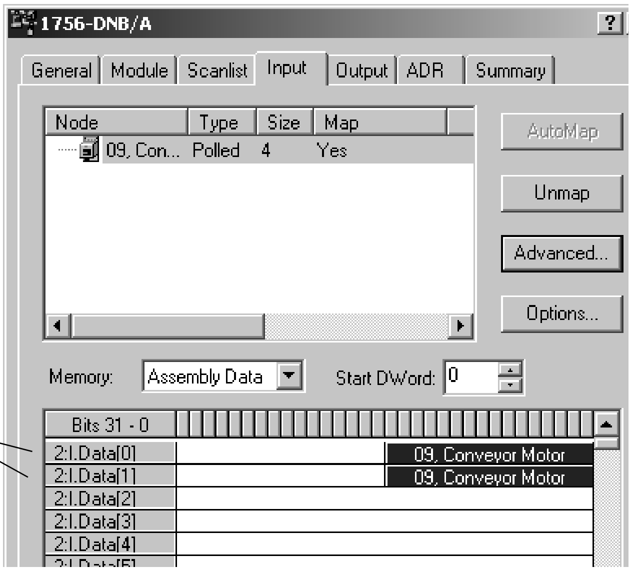
Node	Type	Size	Map
09, Con...	Polled	4	2:I.Data[0].0

Memory: Assembly Data

Start DWord: 0

Bits 31 - 0	
2:I.Data[0]	09, Conveyor Motor
2:I.Data[1]	

When you use AutoMap, all of the data for a device ends up packed together.



Node	Type	Size	Map
09, Con...	Polled	4	Yes

Memory: Assembly Data

Start DWord: 0

Bits 31 - 0	
2:I.Data[0]	09, Conveyor Motor
2:I.Data[1]	09, Conveyor Motor
2:I.Data[2]	
2:I.Data[3]	
2:I.Data[4]	
2:I.Data[5]	

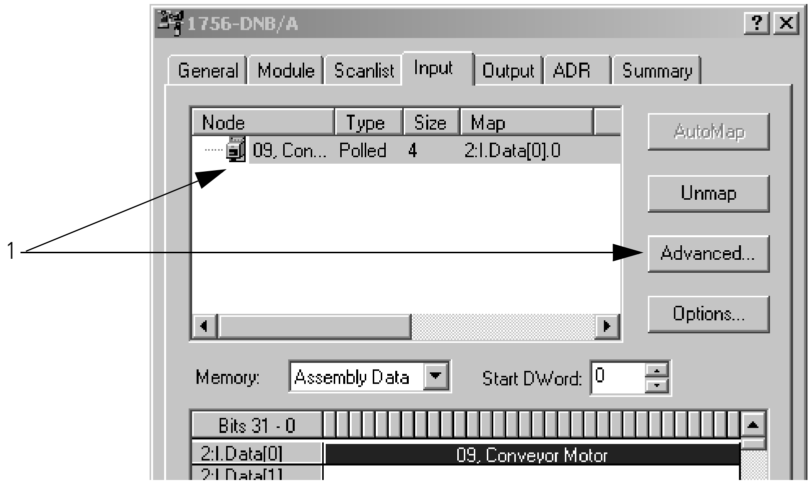
Advanced mapping lets you unpack the data into several map entries. In this example, the upper 16 bits of the original map entry are now in an individual tag.

Give a Value Its Own Memory Location

Complete these steps to give a value its own memory location in the input or output memory of the scanner.

1. Select the device and click the Advanced button.
2. For the first map entry, specify the first bit of the data.
   a. Choose a connection type.
   b. Enter the starting byte of the data.
   c. Enter the starting bit of the data.
3. Specify the map location for the data.
   a. Choose the element number in the map.
   b. Enter the starting bit.
   c. Enter the number of bits.

4. Click Apply Mapping.

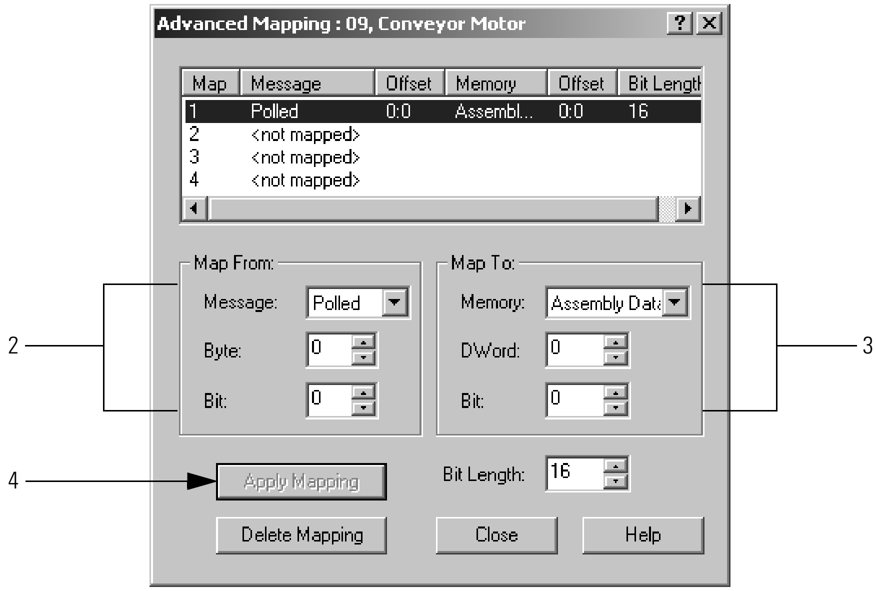


Node	Type	Size	Map
09, Con...	Polled	4	2:I.Data[0].0

Memory: Assembly Data

Start DWord: 0

Bits 31 - 0	
2:I.Data[0]	09, Conveyor Motor
2:I.Data[1]	



Map	Message	Offset	Memory	Offset	Bit Lengtl
1	Polled	0:0	Assembl...	0:0	16
2	<not mapped>				
3	<not mapped>				
4	<not mapped>				

Map From:

Message: Polled

Byte: 0

Bit: 0

Map To:

Memory: Assembly Datı

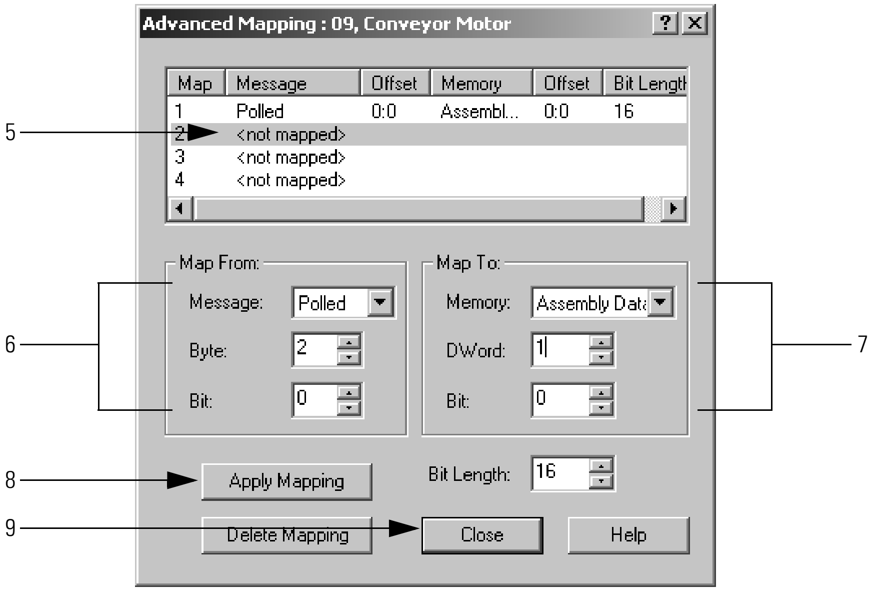
DWord: 0

Bit: 0

Bit Length: 16

5. Select the next map number.
6. Specify the first bit of the data for the next map entry for this device.
   a. Choose a connection type.
   b. Enter the starting byte of the data.
   c. Enter the starting bit of the data.
7. Specify the map location for the data.
   a. Choose the element number in the map.
   b. Enter the starting bit.
   c. Enter the number of bits.
8. Click Apply Mapping.

9. Click Close when you are done.



Advanced Mapping : 09, Conveyor Motor

Map	Message	Offset	Memory	Offset	Bit Length
1	Polled	0:0	Assembl...	0:0	16
2	<not mapped>				
3	<not mapped>				
4	<not mapped>				

Map From:

Message: Polled

Byte: 2

Bit: 0

Map To:

Memory: Assembly Data

DWord: 1

Bit: 0

Bit Length: 16

Notes:

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://www.rockwellautomation.com/support/, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration, and troubleshooting, we offer TechConnect support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/support/.

Installation Assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

United States or Canada	1.440.646.3434
Outside United States or Canada	Use the Worldwide Locator at http://www.rockwellautomation.com/support/americas/phone_en.html, or contact your local Rockwell Automation representative.

New Product Satisfaction Return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

United States	Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for the return procedure.

Documentation Feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete this form, publication RA-DU002, available at http://www.rockwellautomation.com/literature/.

Rockwell Otomasyon Ticaret A.Ş., Kar Plaza İş Merkezi E Blok Kat:6 34752 İçerenköy, İstanbul, Tel: +90 (216) 5698400

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication DNET-UM004C-EN-P - June 2016
Supersedes Publication DNET-UM004B-EN-P - March 2011