



Universidad Autónoma de Querétaro

Facultad de Ingeniería

“Utilización de una red Device-Net como transferencia
de tecnología en una maquina inyectora de plástico.”

TESINA

Que como parte de los requisitos para obtener el título de:

Ingeniero en Automatización

Presentan:

Israel Barrón Balderas

José Carlos Sánchez Herrera

Dirigida por:

Dr. Edgar Alejandro Rivas Araiza

Querétaro, Querétaro., Agosto del 2010

La presente obra está bajo la licencia:
<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es>



CC BY-NC-ND 4.0 DEED

Atribución-NoComercial-SinDerivadas 4.0 Internacional

Usted es libre de:

Compartir — copiar y redistribuir el material en cualquier medio o formato

La licenciante no puede revocar estas libertades en tanto usted siga los términos de la licencia

Bajo los siguientes términos:



Atribución — Usted debe dar [crédito de manera adecuada](#), brindar un enlace a la licencia, e [indicar si se han realizado cambios](#). Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante.



NoComercial — Usted no puede hacer uso del material con [propósitos comerciales](#).



SinDerivadas — Si [remezcla, transforma o crea a partir](#) del material, no podrá distribuir el material modificado.

No hay restricciones adicionales — No puede aplicar términos legales ni [medidas tecnológicas](#) que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

Avisos:

No tiene que cumplir con la licencia para elementos del material en el dominio público o cuando su uso esté permitido por una [excepción o limitación](#) aplicable.

No se dan garantías. La licencia podría no darle todos los permisos que necesita para el uso que tenga previsto. Por ejemplo, otros derechos como [publicidad, privacidad, o derechos morales](#) pueden limitar la forma en que utilice el material.

I. INDICE GENERAL

I. INDICE GENERAL.....	1
II. INDICE DE FIGURAS.....	3
III. INDICE DE TABLAS	5
IV. RESUMEN.....	6
V. OBJETIVOS.....	8
VI. JUSTIFICACIÓN.....	9
VII. ANTECEDENTES.....	10
VIII.MARCO TEORICO	19
8.1 CAN	19
8.1.1 Frame de datos	19
8.1.2 Acceso a la red.....	19
8.1.3 Control de Errores.....	20
8.2 CAN y DeviceNet	20
8.3 DeviceNet.....	21
8.3.1 Características de DeviceNet	22
8.3.2 DeviceNet en el modelo OSI.	23
8.3.3 Modos de Comunicación	27
8.4 Topología	28
8.5 Detalles del Cable.....	30
8.6 Tap´s.....	31
8.6.1 Tap´s sellados	31
8.6.2 Tap´s abiertos.....	32
8.7 INTERFAZ 1770 KFD.....	33
8.8 SCANNER 1769-SDN.....	34
8.8.1 Características del modulo (Tabla 4.)	34
8.9 SENSOR 42EF-D1LDAK-F5	35
8.10 LIMIT SWITCH 802DN-AD5 (lever type).....	36
IX. DESARROLLO	37
9.1 Características y descripción de la máquina.	37
9.1.1 El principio básico de la maquina inyectora comprende las tres operaciones siguientes.	37
9.2 Armado de la red DeviceNet.	40
9.3 Configuración de la Interfaz 1770 KFD a través de RSLinx	43
9.4 Configuración del controlador del PLC a través de RSLinx.....	50
9.5 Configuración y ajustes principales de RSNetWorx para la red DeviceNet.....	56
9.6 Configuración y ajustes del Scanner con RSLogix 500.....	63
9.7 Monitoreo de la red utilizando el software LabVIEW	71

9.7.1 Configuración de RSLinx para poder establecer la comunicación entre la red y el PLC	83
9.7.2 Inicio del monitoreo de la red	86
X. CONCLUSIONES	88
XI. BIBLIOGRAFIA.....	89

II. INDICE DE FIGURAS

Figura 1. DeviceNet en el modelo OSI.....	23
Figura 2. Frame de datos CAN.....	25
Figura 3. Estructura en capas del protocolo CIP	26
Figura 4. Topología de DeviceNet	29
Figura 5. Configuración del cable	30
Figura 6. Detalles de cable	31
Figura 7. Tap Tee.....	32
Figura 8. Tap Multipuerto	32
Figura 9. Tap estilo abierto.....	32
Figura 10. Conexión del modulo en la red	33
Figura 11. Scanner 1769-SDN	34
Figura 12. Sensor ajustable y no ajustable	35
Figura 13. Maquina inyectora de plástico	38
Figura 14. Maquina inyectora de plástico con sus dispositivos	38
Figura 15. Esquemático de la red DeviceNet	39
Figura 16. Tap estilo abierto con conexión para la alimentación.....	40
Figura 17. Tap estilo abierto con configuración.....	40
Figura 18. Tap estilo abierto con resistencia terminal.....	41
Figura 19. Tap Tee para la derivación	41
Figura 20. Icono de RSLinx	43
Figura 21. Ventana "Configure Drivers"	43
Figura 22. Ventana para configurar controladores	44
Figura 23. Lista de los distintos tipos de controladores "Drivers"	44
Figura 24. Selección del driver a utilizar.....	45
Figura 25. Selección de la interfaz	45
Figura 26. Configuración del controlador 1770-KDF.....	46
Figura 27. Inicialización de la comunicación.....	47
Figura 28. Ventana para el nombre del controlador	47
Figura 29. Status del controlador.....	48
Figura 30. Ventana principal de RSLinx	49
Figura 31. RSWho	49
Figura 32. RSLinx	50
Figura 33. Ventana principal de RSLinx	50
Figura 34. Configurar controladores "Configure Drivers"	51
Figura 35. Lista de los tipos de controladores	51
Figura 36. Selección del driver a utilizar.....	52
Figura 37. Nombre del controlador	52
Figura 38. Ventana de configuración del controlador	53
Figura 39. Auto-Configure	53
Figura 40. Status del controlador.....	54
Figura 41. Opción RSWho	55
Figura 42. Ventana de RSWho	55
Figura 43. RSNetwox	56
Figura 44. Ventana principal de RSNetworx	56

Figura 45. Opción "Online"	57
Figura 46. Selección de la interfaz	58
Figura 47. Escaneado de la red	58
Figura 48. Dispositivos conectados en la red	59
Figura 49. Configuración del Scanner	59
Figura 50. Estado de la red	60
Figura 51. Ventana de la opción "Scanlist"	60
Figura 52. Ventana de la opción "Input" del scanner	61
Figura 53. Ventana de la opción "Summary"	61
Figura 54. RSLogix 500	63
Figura 55. Ventana principal de RSLogix 500	63
Figura 56. Opción "New"	64
Figura 57. Lista de controladores	64
Figura 58. Área de trabajo de RSLogix	65
Figura 59. Selección de "IO Configuration"	65
Figura 60. Ventana de configuración de E/S	66
Figura 61. Ventana "Read IO Config"	66
Figura 62. Scanner dado de alta como dispositivo de E/S	67
Figura 63. Selección de la opción "Download"	67
Figura 64. Ventana de "Save Program As.."	68
Figura 65. Ventana de descarga al driver configurado en RSLinx	68
Figura 66. Ventana del avance de la descarga del programa en el PLC	69
Figura 67. Ventana "Estar en línea o no con el PLC"	69
Figura 68. Ventana de comunicación entre el PLC y la PC	69
Figura 69. Ventana "Output"	70
Figura 70. Opción "Go offline"	70
Figura 71. Icono del programa LabVIEW	71
Figura 72. Ventana principal del software	71
Figura 73. Ventana "Front Panel"	72
Figura 74. Ventana "Block Diagram"	72
Figura 75. Colocación de texto en la ventana "Front Panel"	73
Figura 77. Colocación del texto con fondo de color	73
Figura 76. Ventana de "Tools"	73
Figura 78. Ventana del menú "Controls"	74
Figura 79. Colocación de un LED	74
Figura 80. Ventana "Front Panel" con los 5 LEDs	75
Figura 81. Colocación del nombre de la variable	75
Figura 82. Ventana "Block Diagram" con los iconos de los LEDs	76
Figura 83. Ventana de las funciones de comunicación	76
Figura 84. Ejemplo de la ventana de propiedades de una función de comunicación	77
Figura 85. Icono de una función de comunicación	77
Figura 86. Ventana "Block Diagram" con las funciones de comunicación	78
Figura 87. Conexiones virtuales entre las distintas funciones "DDE"	79
Figura 88. Opción "Service" de la función DDE Open	79
Figura 89. Opción "Project" de la función DDE Open	80
Figura 90. Opción "Item" de la función DDE Request	80
Figura 91. Herramienta "Decimal String to Number"	81

Figura 92. Herramienta "Number to Boolean Array"	82
Figura 93. Herramienta "Index Array"	82
Figura 94. Icono de RSLinx	83
Figura 95. Opción "DDE/OPC - Topic Configuration"	84
Figura 96. Ventana de la opción "Topic Configuration"	84
Figura 97. Opción para cargar el nuevo Topic en el PLC	85
Figura 98. Configuración del DDE/OPC Topic	85
Figura 99. Selección del driver a utilizar para el intercambio de datos.....	86
Figura 100. Monitoreo de la Red Activado	86

III. INDICE DE TABLAS

Tabla 1. Frame de datos.....	19
Tabla 2. Longitud y velocidades de la red.....	24
Tabla 3. Velocidad de transmisión en base al cable	29
Tabla 4. Descripción del Scanner 1769-SDN.....	34

IV. RESUMEN

El desarrollo del control distribuido que está formado por una gran variedad de campos va en paralelo con el de las comunicaciones. Esto está enfocado a diferentes niveles de abstracción sobre integración y producción de acuerdo a la filosofía de la "Computer Integrated Manufacturing" - CIM. Cada vez es más necesario disponer de dispositivos inteligentes para realizar el control o la supervisión remota. Un bus de campo transfiere información secuencial y serial por un número limitado de líneas o cables. Hay muchos tipos diferentes de buses en uso y muchos son altamente dependientes de las aplicaciones. En el presente trabajo se analizará el estado de avance en la tecnología de la comunicación de los buses de campo aplicados al control de procesos industriales.

En la industria actual se llevan a cabo múltiples procesos que si bien algunos pueden llevarse a cabo de manera independiente, existen muchos otros que necesitan de datos correspondientes al mismo sistema por lo que surge la necesidad de la comunicación entre los equipos y el operador. Inicialmente se llevaba a cabo esta comunicación por medio de señales analógicas como la presión, o incluso por medio de señales eléctricas de corriente (generalmente de 4-20mA).

Con la complejidad que adquirieron los procesos fue necesario el intercambio de una mayor cantidad de datos, lo cual hacía inviable la cantidad de elementos (como válvulas o cables). Con el objetivo de simplificar estas comunicaciones y de hacerlas más eficientes se aplicó la idea de los buses de datos, en donde toda la información del sistema viaja en un grupo de conductores en forma de señales digitales.

En principio la limitante de esta clase de comunicaciones se hallaba en la conversión de variables analógicas como presión o temperatura en una señal digital. Con el avance de la tecnología ya se encuentran disponibles en el mercado todos los instrumentos capaces de realizar esta tarea, por lo que la idea de un bus de datos se hace atractiva a la hora de controlar un proceso industrial complejo.

Por supuesto para que todos los instrumentos se pudieran “entender” entre sí, era necesario establecer un protocolo de comunicaciones. A final de cuentas se crearon muchos de ellos que llegaron a tener diferente éxito dependiendo de la aplicación en que fuera a usarse o hasta de la región geográfica.

En el presente trabajo implementaremos el uso de un protocolo de comunicación como medio de transmisión y monitoreo de los datos del estado de una maquina inyectora de plástico, esto nos permitirá hacer más funcional el sistema de cableado que actualmente utiliza la maquina ya que al utilizar el bus de campo reduciremos la cantidad de cables que utiliza la máquina y así será más fácil la detección de fallas en algún punto del proceso.

V. OBJETIVOS

En el presente trabajo de investigación se buscara explicar la definición y el funcionamiento de los distintos buses de campo que se utilizan para armar redes industriales que nos permitan manejar y monitorear fácilmente dispositivos de control tales como PLC's, DCS's, etc.

El objetivo principal es actualizar el sistema de comunicación de una maquina inyectora de plástico, esto como un ejemplo de la aplicación y ventajas que se pueden obtener al utilizar un bus de campo como protocolo de comunicación para el monitoreo y control de dicha maquina a través de un elemento de control tal como un PLC.

VI. JUSTIFICACIÓN

Se utilizara DeviceNet como medio de comunicación debido a que dicho bus de campo es una solución simple de comunicación en red que nos permite reducir el costo y el tiempo para cablear e instalar dispositivos de automatización industrial, al mismo tiempo que nos provee intercambiabilidad de componentes similares de distintos fabricantes. Es por eso que en el presente trabajo se buscara implementar dicho bus de campo para actualizar el sistema de comunicación de una maquina inyectora de plástico y también para poder así mismo reducir la cantidad de cables que necesita la máquina para su operación.

Además al implementar dicho bus de campo se verá reflejado un costo-beneficio, pues al ser un bus de campo solo incluye las capas: física, de enlace, y de aplicación del modelo OSI. Otra ventaja que se podrá obtener al utilizar DeviceNet es en el gasto que se tendrá que realizar ya que este es relativamente mínimo si consideramos que dicho bus es universalmente aceptado, que tiene la capacidad de aceptar la gran mayoría de los dispositivos de campo empleados en el proceso, además de que permite leer el estado de los dispositivos discretos conectados a la red.

Entre las aplicaciones que tiene DeviceNet y que podrían servir en el proyecto se encuentra que:

- Tiene la capacidad de reportar temperaturas.
- Leer la corriente de carga de un arrancador de motor.
- Cambiar la velocidad de un impulsor de un motor o contar el número de piezas (en nuestro proyecto serian envases) que han pasado en una banda transportadora durante el turno proceso actual o el proceso anterior (esto se puede hacer con sensores o bien con el uso de algún encoder).

VII. ANTECEDENTES

En el año de 1947 aparece el primer transistor, una revolución sin dudas, que marcaría un nuevo rumbo a la humanidad, prácticamente no existieron disciplinas que no se vieran influenciadas, y la industria no podría estar alejada de esto, es más, sería una de las más afectadas. En el año de 1972 se logró integrar varios transistores en lo que se denominó circuitos integrados, para luego dar paso a los microprocesadores, la curva de evolución hasta aquí avanzó lenta, pero muy firmemente; a partir de aquí la pendiente de la curva de evolución incrementó, y prácticamente no lo ha dejado de hacer.

Así es que la inclusión de microprocesadores en numerosos equipos de campo permitió su integración en redes, con importantes ventajas, tales como precisión, costo, diagnóstico, etc. Estos conceptos resultaron aplicables a controladores de un solo lazo, módulos remotos de I/O, transmisores inteligentes, PLCs, DCSs y otros equipos digitales de campo. Dado que no existían normas para la integración digital de estos equipos, cada proveedor desarrolló un protocolo propio, por lo que estas ventajas estuvieron limitadas a algunos equipos de un mismo proveedor.

Varios grupos han intentado generar e imponer una norma que permita la integración de equipos de distintos proveedores. Se pueden identificar dos corrientes en este sentido. En los Estados Unidos, la introducción en 1982 de los primeros transmisores inteligentes por parte de la firma Honeywell, seguida por otras compañías como Foxboro y Rosemount, incentivó a ISA a formar el comité SP50, cuyo objetivo es el diseño de un protocolo normalizado para la integración de transmisores inteligentes. En Europa, firmas como Siemens, Telemecanique, AEG, Klockner Moeller, y otras, concentraron sus esfuerzos en la creación de un bus digital para la integración de PLCs. Dos grupos se impusieron: la norma alemana Profibus, y la norma francesa FIP.

Puesto que el esfuerzo de los tres grupos está dirigido a desarrollar e imponer un bus de datos que permita la integración de equipos digitales de campo, se lo denominó genéricamente “Fielbus”.

El desarrollo de esta norma emergente no ha estado exento de marchas, contramarchas, e incluso una fuerte presión de proveedores que ya habían diseñado un Fieldbus, y deseaban disminuir sus futuros costos de diseño imponiéndolo.

El comité ISA SP50 fue el responsable de una tarea de normalización actualmente muy difundida: SP50.1 que normalizó la señal de 4 a 20 mA por el año 1975. El comité fue reconocado en 1984 para una nueva tarea, la normalización del Fieldbus. El objetivo era establecer una norma para el protocolo de comunicaciones digitales, serial y bidireccional para sensores y actuadores inteligentes montados en áreas del proceso. Un objetivo adicional era que el nuevo protocolo debía permitir el uso del cableado ya existente.

En forma simultánea la IEC empezó a trabajar en el desarrollo de un Fieldbus, con la comisión SC65WG6. Pronto ambos institutos unificaron sus esfuerzos, y muchos miembros de una comisión forman parte también de la otra.

Para su diseño, el Fieldbus fue dividido en distintos niveles según el modelo ISO/OSI. De los 7 niveles del modelo, el comité SP50 implementa solo tres: el nivel físico, el de enlace de datos y el de aplicación. Los demás niveles simplemente son omitidos. Además se agrega un nuevo nivel, el nivel de usuario, que es el que permite la intercambiabilidad de instrumentos. Este nivel consiste en una detallada descripción de la base de datos que identifica la función de un equipo, por medio de bloques funcionales como "entrada analógica", "lazo PID", y otros. Este concepto de octavo nivel o "capa de usuario" también es incluido en el desarrollo de ISP (Interoperable Systems Project). Cada nivel es estudiado por un subcomité, y aprobado por separado.

El desarrollo de Fieldbus SP50 resultó extremadamente lento, debido a la gran cantidad de usuarios y proveedores que participan en los comités con voz y voto.

La norma alemana Profibus y el grupo ISP: Profibus, o Process Fieldbus, es una norma alemana para Fieldbus, que contó con el aporte inicial de Bosch, Klockner-Moeller y Siemens. Ya existen más de 100 empresas que cuentan con productos ProfiBus. La definición del Profibus se basó esencialmente en la tecnología disponible: RS-485 para el nivel 1, IEC 955 para el nivel 2, y un subconjunto de MMS (Manufacturing Message

Specification) para el nivel 7. El protocolo IEC 955 especifica un modelo de control de acceso al medio denominado maestro volante (flying master), que es una combinación de los métodos de control de acceso al medio token bus y maestro-esclavo.

En este modelo, algunas estaciones son definidas como maestras, y otras como esclavas. Las estaciones maestras acceden al bus por medio de un esquema token bus. Mientras una estación maestra posee el testigo, realiza transacciones con sus esclavos según el esquema maestro-esclavo. Luego envía el testigo a la siguiente estación del anillo lógico.

Cuando Yokogawa, Siemens, Rosemount y Fisher fundan ISP el 15 de setiembre de 1992, buscaron tomar como base una tecnología existente y probada, para lograr una rápida implementación. Se seleccionó entonces la tecnología de Profibus, que fue integrada dentro del ISP.

Se cubrieron entonces las tres capas del modelo ISO/OSI que utiliza el fieldbus: la capa física, la de enlace, y la de aplicación. Sin embargo, el comité ISP pretende que el fieldbus permita la interoperabilidad de los equipos. En tal sentido, define como interoperabilidad la habilidad de sustituir un instrumento de un fabricante con uno equivalente de cualquier otro fabricante, sin degradar su rendimiento, ni el de la red, ni limitar su funcionalidad. Este nivel de detalle no es satisfecho por MMS, ya que para su implementación se requiere una definición desde el punto de vista de la estructura de los parámetros de configuración del dispositivo. Para ello se deberá definir el formato de la base de datos de un transmisor de presión diferencial, de un controlador, etc.

Esta funcionalidad es cubierta por el DDL (Drive & Diagnostic Link), que es un desarrollo de Rosemount, quien lo implementó en su protocolo HART.

El directorio del Proyecto ISP asigna una gran importancia al fieldbus, considerándolo como la próxima transición tecnológica de importancia en el control de procesos.

Hacia fines de 1993 el ISP contaba con un directorio con voz y voto formado por seis empresas (los cuatro miembros fundadores mas Foxboro y ABB), y más de un centenar de miembros adherentes, que tienen voz pero no voto, entre los que se destacan AEG

Modicon, Brooks, Eckardt, Endress+Hauser, Fuji, Hartmann & Braun, Magnetrol, Siebe y Valmet. Los miembros del ISP cuentan con experiencia proveniente de la comisión IEC/SP50, el Grupo de Usuarios HART y la Organización de Usuarios de Profibus. Esta experiencia, y la escasa cantidad de miembros con voto, han dotado a ISP de un alto dinamismo, esperándose que los primeros productos que cumplieron con ISP fueran lanzados al mercado en el año 1994.

Los Trabajos de implementación de Profibus y FIP dieron lugar al grupo SP50 de ISA que colabora con el IEC (International Electrotechnical Commission) en la elaboración de una norma internacional. Sus resultados pueden verse en la Norma IEC 1158. La arquitectura en esta norma también se basa en el modelo ISO/OSI con los mismos tres niveles de Profibus y FIP. ISA introduce un cuarto nivel (nivel de usuario), pero todavía no es aceptado por el IEC. En baja velocidad se admiten hasta 32 instrumentos por bus sin considerar la seguridad intrínseca, 12 instrumentos con alimentación por bus y sin seguridad intrínseca, y de 2 a 6 instrumentos con alimentación por bus y seguridad intrínseca. El nivel de usuario está destinado a facilitar el funcionamiento de los instrumentos de la red en conjunto, definiendo la estructura de la base de datos que residirá en cada instrumento de control o de medición. Lo que se busca es que distintos proveedores puedan construir sus productos con algoritmos predefinidos, requiriendo tan solo que la base de datos sea cargada para una configuración específica. También define el método para escalonar todos los bloques funcionales en base a la necesidad de tiempo crítico de cada bloque. El período de configuración de cada bloque queda determinado por la configuración de la base de datos. El nivel de usuario incorpora un lenguaje común de programación para todos los instrumentos de campo, proveyendo una terminal de programación y una metodología de configuración única para instrumentos de distintos proveedores.

El FF (fieldbus foundation) se caracteriza por su interoperabilidad multiproveedor y este es aceptado por la gran mayoría de los proveedores líderes en automatización de procesos y de fabricación de instrumentos.

La norma francesa FIP y el grupo WORLDFIP: FIP (Factory Instrumentation Protocol) es una norma de fieldbus desarrollada en Francia. Un club FIP se ocupa de difundir la aplicación de dicha norma a nivel internacional, fomentando el desarrollo de productos y brindando soporte técnico. El club FIP cuenta con más de 120 miembros de los más variados orígenes: Bélgica, Bulgaria, Inglaterra, Francia, Alemania, Italia, España, etc.

FIP se propone ser un sistema de gerenciamiento de una base de datos industrial, en tiempo real y distribuida, para esto se basa en el modelo PDC (Primary Domain Controller). De acuerdo a este modelo, los programas de aplicación que se ejecutan en las estaciones pueden cumplir tres funciones:

- **Productor de datos:** es un programa de aplicación responsable de la producción de un dato o variable. Por ejemplo, un transmisor es responsable de la producción de un dato que representa la variable censada, mientras que un controlador es responsable de la producción de un dato que representa el porcentaje de apertura de una válvula.
- **Consumidor de Datos:** es un programa de aplicación que requiere un dato o variable. Por ejemplo, un actuador requiere conocer el porcentaje de apertura al que debe estar la válvula, mientras que un controlador multilazo es consumidor de numerosas variables requeridas para cumplir sus funciones.
- **Distribuidor:** el distribuidor es responsable de la validez de la transferencia de datos, así como de que esta se produzca en los plazos de tiempo requeridos por el proceso.

FIP ofrece en el nivel de aplicación dos tipos de servicios: servicios de transferencia de mensajes (subconjunto de MMS) y Servicios periódicos/aperiódicos MPS.

En febrero de 1993 se funda el grupo WorldFIP, con el objetivo de promover a FIP como norma internacional de fieldbus. WorldFIP integra a miembros de FIP y nuevos miembros, contando actualmente con más de 150 miembros, incluyendo Honeywell, Elsag Bailey, Allen Bradley, Masoneilan Dresser, Square D, Yamatake-Honeywell, Cegelec, etc. Se basa

en la norma NFC 46-602 y NFC 46-607 y en el modelo ISO/OSI reducido a tres niveles, físico, de enlace y de aplicación.

WorldFIP Europe es un protocolo para operaciones de proceso, batch y de fabricación discreta. Entre las principales aplicaciones se incluyen las industrias automotriz, química, petroquímica, siderurgia, alimenticia y fabricación de papel. La simplicidad inherente del protocolo le ofrece al usuario una entrega garantizada de variables de tiempo crítico y le brinda la posibilidad de transferir archivos de datos en el mismo bus sin ninguna programación especial de las aplicaciones.

Algunos tipos de buses de campo que se han desarrollado son:

LonWorks, de Echelon Corp. es la única tecnología de control digital que soporta el uso simultáneo de varios medios de bajo costo, incluyendo par trenzado, línea eléctrica común, infrarrojo, radiofrecuencia y cable coaxial. Adecuado para aplicaciones en piso de planta, edificios comerciales, residencias privadas y control de instalaciones, el sistema LonWorks, que está basado en la tecnología Neuron de chips embebidos, puede ser configurado como un bus de dispositivo/sensor determinístico, par a par y maestro/esclavo, o como un flidbus de mayor nivel.

Utilizado en numerosas aplicaciones industriales de control, cuando se lo configura en una arquitectura de administración de red de tipo cliente/servidor, ofrece una capacidad de escalabilidad virtualmente ilimitada. El uso por parte del sistema de un importante volumen de componentes de bajo costo afecta favorablemente los costos de instalación.

ControlNet, de Allen-Bradley Co. fue introducido inicialmente como una red de control de alto funcionamiento y para propósitos generales. Por su capacidad inherente de datos, ControlNet puede soportar instrumentación inteligente, ofreciendo asimismo conectividad de proceso a sistemas de accionamiento de velocidad variable, MMI y DCS. Está destinado a aplicaciones de fabricación discreta de alta velocidad y procesos analógicos de tipo general.

El funcionamiento de la red es tanto determinístico como repetible; esto es, ControlNet garantiza el envío de los datos y no cambia el tiempo de transmisión si se agrega o se saca un dispositivo de la red. El resultado es un control más estricto en la mayoría de las aplicaciones.

HART, este protocolo provee comunicación digital bidireccional con dispositivos de campo inteligentes mientras conserva la compatibilidad y familiaridad de los tradicionales sistemas 4-20 mA. Su protocolo utiliza la norma Bell 200 que permite la superposición simultánea a niveles bajos de una señal de comunicaciones digital (el "1" lógico es representado por 1200 Hz, mientras el "0" lógico corresponde a 2200 Hz) en la parte superior de la señal analógica 4-20 mA.

Ya que HART fue diseñado para ampliar las comunicaciones con los instrumentos de medición y control que tradicionalmente se comunicaban con señales de 4-20 mA, es aplicable a todo tipo de industrias de proceso. Por tratarse de una superposición a un sistema existente, el HART ofrece una solución sin ningún riesgo para poder gozar de los beneficios que resultan de una comunicación más amplia con los dispositivos inteligentes.

SDS, de Honeywell es un protocolo de comunicaciones y entorno de control basado en CAN para automatización del piso de la planta. Permite el envío de mensajes entre sensores, actuadores, HMIs, controladores y otros dispositivos del piso de una planta. La arquitectura del sistema provee un entorno de comunicaciones en base al cual se pueden generar plataformas de control de alta velocidad, en tiempo real, centralizado o distribuido.

Según Honeywell, el atributo más significativo de SDS es la simplicidad. Su protocolo, fácil de entender, permite a los usuarios implementar fácilmente redes del mundo real y a los fabricantes de dispositivos desarrollar con mayor facilidad productos que soportan la red.

SERCOS, brinda a los usuarios de accionamientos y controles digitales los beneficios de la comunicación sincrónica con una resolución digital de 32 bits, aportando un medio de transmisión con fibras ópticas inmune al ruido, una extensa capacidad de diagnóstico y una buena respuesta en aplicaciones donde se exige control de movimiento multi-eje digital y

distribuido. Esta norma abierta tiene aprobación IEC-1491. Destinada a aplicaciones de control de movimiento, SERCOS está soportada por más de 30 proveedores de todo el mundo.

Genius, introducido en 1985 por GE Fanuc Automation, es un sistema de entradas/salidas distribuídas destinado a una amplia gama de aplicaciones desde la industria petroquímica y farmacéutica hasta el manipuleo de materiales.

El sistema se caracteriza por un control determinístico en tiempo real, mientras su tecnología de entradas/salidas distribuídas permite simplificar el cableado de control. La simplicidad del sistema y la aptitud de las E/S Genius permiten proveer automáticamente información de diagnóstico a través del cableado de campo permitiendo reducir en mucho el tiempo de ajuste, la eliminación de errores y la búsqueda de fallas, minimizando así la carga del sistema.

Seriplex, soportada por Seriplex Technology Organization, provee control determinístico de dispositivos tanto analógicos como digitales con el mismo bus. Su protocolo es transparente para el usuario y no hay reglas restrictivas para la topología del bus. Acepta configuraciones estrella, anillo y multigota en cualquier combinación, sin límites en el largo de las conexiones.

Los chips embebidos en esta red de pequeño tamaño, que no están basados en un microprocesador, permiten que los dispositivos de E/S y los bloques inteligentes conectados sean compactos y de bajo costo. La compactidad también hace posible que un solo cable Seriplex pueda reemplazar numerosos cables paralelos punto a punto, reduciendo y simplificando los costos de instalación y de puesta en marcha.

DeviceNet, es una red de dispositivos soportada por una organización de proveedores denominada ODVA. DeviceNet resuelve aplicaciones de control discreto en industria automotriz, alimenticia, manipuleo de materiales, máquinas herramienta, producción de pulpa y papel, y minería.

Al contar con una gran variedad de componentes, DeviceNet les brinda a los usuarios flexibilidad y una amplia base de soporte en aplicaciones específicas de producción, controlando los costos y maximizando la funcionalidad.

Interbus, La provisión de gran volumen de datos que requieren las E/S avanzadas es el atributo principal que define el InterBus, un bus de dispositivos popular en Europa y que ahora está ingresando en Norteamérica a través de una organización de usuarios/proveedores conocida como InterBus-S Club.

Este protocolo garantiza transmisiones seguras en base a su capacidad de verificación de errores. También incorpora extensas capacidades de diagnóstico y un protocolo de mensajes que permite enviar datos parametrizados y mensajes más complejos. Interbus está destinado a aplicaciones de fabricación discreta en la industria automotriz, gráfica y manipuleo de materiales.

AS-i, provisto por Siemens como un bus de red de bajo nivel, se utiliza en aplicaciones con sensores/actuadores binarios, por ejemplo en el procesamiento de productos alimenticios y operaciones de manipuleo de materiales. AS-i fue desarrollado por una asociación de fabricantes de sensores y actuadores para resolver requerimientos de conexión tanto de dispositivos binarios simples como de sistemas de control de mayor nivel.

Su tecnología de alta velocidad y bajo costo reemplaza el cableado desde los dispositivos hasta los paneles. La red usa un cable de dos conductores que transporta hacia los nodos tanto la señal de datos como la alimentación

ArcNet, es la columna vertebral de una red de área local de Contemporary Control Systems. Destinado a aplicaciones de control en tiempo real, este bus a nivel de dispositivos tiene aplicaciones en el control de procesos y en la automatización de fábricas y edificios.

El funcionamiento de ArcNet se basa en un protocolo de tipo token-passing, lo que les da a todos los nodos igual acceso a la red y elimina las posibles colisiones de transmisión en las redes ocupadas.

VIII. MARCO TEORICO

8.1 CAN

La red DeviceNet es una red basada en el protocolo CAN, lo que quiere decir que utiliza mensajes CAN para el intercambio de datos en la red.

El protocolo CAN es un protocolo de comunicación serial que describe los servicios de la capa 2 del modelo OSI/ISO (capa de enlace de datos). En esta capa, son definidos los diferentes tipos de mensajes (frames), la forma de detección de errores, validación y arbitraje de mensaje.

8.1.1 Frame de datos

Los datos en una red CAN son transmitidos a través de un frame de datos. Este tipo de frames es compuesto principalmente por un campo identificador de 11 bits (arbitraje field), y un campo de datos (data field), que puede contener hasta 8 bytes de datos. (Tabla 1.)

Identificador	8 bytes de datos							
11 bits	byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7

Tabla 1. Frame de datos

8.1.2 Acceso a la red

En una red CAN, cualquier elemento de la red puede intentar transmitir un frame para la red en un determinado instante. En algún caso cuando dos elementos intenten acceder a la red al mismo tiempo, solo se conseguirá transmitir el mensaje más prioritario. La prioridad de mensaje es definida por el identificador del frame CAN, cuanto menor sea el valor de este identificador, mayor será la prioridad del mensaje. El mensaje con identificador 0 (cero) corresponde al más prioritario.

8.1.3 Control de Errores

La especificación CAN define diversos mecanismos para control de errores, lo que la torna una red muy confiable y con un índice muy bajo de errores de transmisión. Cada dispositivo de la red debe ser capaz de identificar la ocurrencia de estos errores, e informar a los demás elementos que un error fue detectado.

Un dispositivo de la red CAN tiene contadores internos que son incrementados cada vez que un error de transmisión o recepción es detectado, y decrecido cuando un mensaje es enviado o recibido con suceso. Cuando ocurra una cantidad considerable de errores, el dispositivo podrá presentar los siguientes estados:

- Warning: cuando el contador pasa de un determinado límite de errores, el dispositivo entra en estado de warning lo que significa una elevada ocurrencia de errores.
- Error Passive: Cuando este valor sobre pasa un límite mayor entra en estado de error passive, en el cual para de actuar en la red al detectar que otro dispositivo envió un mensaje con error.
- Bus off: por último, tenemos el estado de bus off, en el cual el dispositivo no enviara o recibirá más mensajes de la red.

8.2 CAN y DeviceNet

Solamente la definición de cómo detectar errores, crear y transmitir un frame no son suficientes para definir un significado para los datos que son enviados a través de la red. Es necesario que tenga una especificación que indique como el identificador y los datos deben ser montados y como las informaciones deben ser cambiadas. De esta forma los elementos de red pueden interpretar correctamente los datos que son transmitidos. En este sentido, la

especificación DeviceNet define justamente como cambiar datos entre los equipamientos y como cada dispositivo debe interpretar estos datos.

8.3 DeviceNet

DeviceNet es un enlace de comunicación de bajo costo para conectar dispositivos industriales a una red y eliminar cableado costoso.

DeviceNet es un bus de campo económico para componentes industriales, tales como detectores de final de carrera, detectores ópticos, terminales de válvulas, convertidores de frecuencia, paneles de mando y muchos otros productos. También se utiliza para reducir el costoso cableado individual y para mejorar las funciones de diagnóstico específicas por componente.

Esta red se desarrolló para satisfacer las necesidades de la industria y así poder contar con un medio de comunicación que fuera aceptado universalmente y que tuviera la capacidad de aceptar la mayoría de los dispositivos empleados en el primer nivel de la planta. Generalmente utilizado en la integración de controladores industriales y dispositivos de entrada/salida (I/O); el protocolo sigue el modelo productor-consumidor, soporta varios modos de comunicación y posee prioridad entre mensajes.

Es un sistema que puede ser configurado para operar tanto en una arquitectura maestro-esclavo o en la arquitectura distribuida punto a punto. Además de esto, define dos tipos de mensajes, I/O (datos de proceso) y explicit (configuraron y parametrización). Tiene también mecanismos de detección de direcciones duplicadas y aislamiento de los nodos en caso de falla críticas.

8.3.1 Características de DeviceNet

DeviceNet es una de las redes que han adoptado el esquema productor/consumidor, lo que implica que la información producida por una sola fuente en la red alimenta en forma simultánea a todos los probables receptores y deja a éstos la decisión de escuchar o no el mensaje recibido. Los beneficios del esquema se reflejan en una mejor utilización del ancho de banda y la agilización en el tiempo de respuesta en la red.

La red permite que los dispositivos conectados a la misma se puedan enlazar como maestro/esclavo (Master/Slave), entre pares (Peer-to-Peer) y como productor/consumidor. Este último modo de comunicación es un modelo recientemente desarrollado para los protocolos de comunicación más sofisticados, como FF Fielbus y ControlNet. Múltiples nodos consumen en forma simultánea los mismos datos generados por un productor sencillo. Así, los nodos pueden sincronizarse con facilidad, y el ancho de banda es utilizado con mayor eficiencia. La relación maestro/esclavo es la más sencilla de entender, ya que el PLC o Scanner es el maestro y los dispositivos entrada/salida son los esclavos. El esclavo sólo habla cuando se le interroga, y únicamente hay un maestro por red.

Considérese que los nodos pueden ser eliminados o insertados sin necesidad de desconectar la energía a la red. También se dispone de fuentes de energía adicionales las cuales se pueden conectar en cualquier lugar de la red para tener un apoyo redundante. En algunos casos se requiere energizar determinados dispositivos que consumen grandes cantidades de energía de una fuente externa, tales como arrancadores de motores y válvulas solenoides. En estas situaciones se utilizan aisladores ópticos que mantienen la energía adicional aislada de la línea troncal.

A continuación, algunas de las principales características de DeviceNet:

- Número máximo de nodos por red: 64 en topología de bus con derivaciones.
- Distancia máxima: 100 m a 500 m. y hasta 6 km con repetidores, y en velocidades de 125, 250 y 500 Kbps.
- Emplea dos pares trenzados: control y alimentación, con alimentación en 24 VDC, con opción de redundancia.

- Transmisión basada en el modelo productor/consumidor con un empleo eficiente de ancho de banda y con mensajes desde 1 byte hasta largos ilimitados.
- Reemplazo Automático de nodos, no requiere de programación y elevado nivel de diagnósticos.

8.3.2 DeviceNet en el modelo OSI.

Al ser un bus de campo DeviceNet solo incluye las capas Física, Enlace y Aplicación. (Figura 1.)

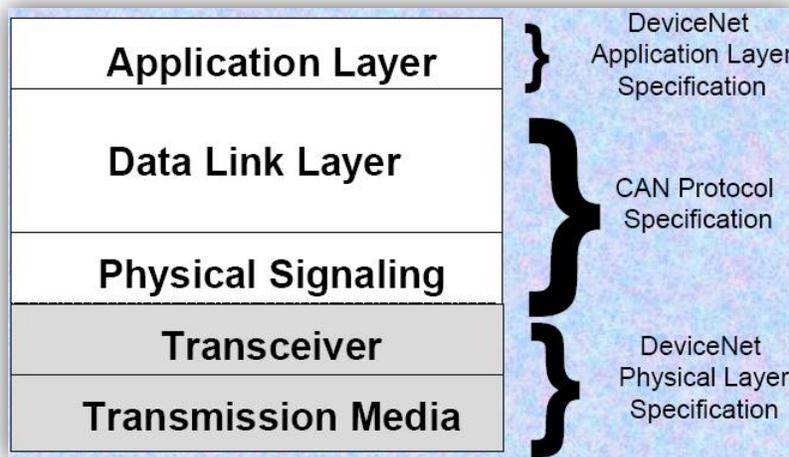


Figura 1. DeviceNet en el modelo OSI

8.3.2.1 Capa Física

DeviceNet utiliza la topología de red tronco/derivación que permite la fijación de la señal de alimentación siempre y cuando estén en el mismo cable. Esta alimentación provee una fuente conectada directamente a la red la cual a su vez suple a los transceivers CAN de los nodos y tiene las siguientes características:

- 24Vdc.
- Salida DC aislada de la entrada AC.
- Capacidad de corriente compatible con los equipamientos instalados.

El tamaño total de la red varía de acuerdo con la velocidad de transmisión utilizada, la cual se muestra en la Tabla 2.

Velocidad de transmisión	Tamaño de la red	Derivación	
		Máximo	Total
125 kbps	500 m	6 m	156 m
250 kbps	250 m		78 m
500 kbps	100 m		39 m

Tabla 2. Longitud y velocidades de la red

Para evitar reflexiones de señal en la línea, se recomienda la instalación de resistencias de terminación en las extremidades de la red, ya que la falta de estas puede provocar errores intermitentes. Estas resistencias deben poseer las siguientes características, conforme la especificación del protocolo:

- 121Ω
- 0,25W
- 1% de tolerancia

8.3.2.2 Capa de Enlace de Datos

La capa de enlace de datos de DeviceNet está definida por la especificación del CAN, el cual define dos estados posibles; dominante (nivel lógico 0) y recesivo (nivel lógico 1). Un nodo puede llevar a la red al estado dominante a transmitir alguna información. Así que solamente estará en estado recesivo si no hay nodos transmisores en estado dominante.

CAN utiliza el CSMA/NBA para acceder el medio físico. Esto significa que un nodo, antes de transmitir debe verificar si el espacio está libre para así evitar colisiones de datos. Si esta libre puede iniciar la transmisión del mensaje y de no ser así tendrá que esperar. Si más de un nodo quiere acceder a la red simultáneamente, un mecanismo basado en prioridad de

mensaje entrará en acción para decidir cuál de ellos tendrá prioridad sobre el otro. Este mecanismo no es destructivo, es decir, el mensaje es retenido para que no ocurra una colisión entre los mismos.

CAN define cuatro tipos de mensajes (data, remote, overload, error). De estos, DeviceNet utiliza apenas el frame de datos (data frame) y el frame de errores (error frame).

Los datos son intercambiados cuando se utiliza el frame de datos. La estructura de este frame se muestra en la Figura 2. Los errores son indicados a través del frame de errores. CAN posee una verificación y un confinamiento de errores bastante robusto. Esto garantiza que un nudo con problemas no afecte la comunicación en la red.

Interframe Space	1 bit	11 bits	1 bit	6 bits	0-8 bytes	15 bits	1 bit	1 bit	1 bit	7 bits	≥ 3 bits
	Start of Frame	Identifier	RTR bit	Control Field	Data Field	CRC Sequence	CRC Delimiter	ACK Slot	ACK Delimiter	End of Frame	Interframe Space

Figura 2. Frame de datos CAN

8.3.2.3 Capa de Transporte y Red

DeviceNet requiere que una conexión sea establecida antes de que haga el cambio de datos con el dispositivo. Para establecer esta conexión, cada nodo DeviceNet debe implementar el Unconnected Message Manager (UCMMO) o el Group 2 Unconnected Port. Estos dos mecanismos utilizan mensajes del tipo explicit para establecer la conexión que será utilizada para el cambio de datos entre un nodo y otro. Este cambio de datos utiliza mensajes del tipo I/O.

Los mensajes DeviceNet son clasificados en grupos, estos grupos definen funciones y prioridades específicas. Estos mensajes utilizan un campo identificador (11 bits) del frame de datos CAN para identificar únicamente cada uno de los mensajes, garantizando así el mecanismo de prioridades CAN.

Un nodo DeviceNet puede ser cliente, servidor o ambos. Además de esto, clientes y servidores pueden ser productores y/o consumidores de mensajes. En un típico nodo cliente, por ejemplo, su conexión producirá requisiciones y consumirá respuestas. Otras conexiones de clientes o servidores apenas consumirán mensajes. Es decir, el protocolo prevé diversas posibilidades de conexión entre los dispositivos. El protocolo dispone también de un recurso para detección de nodos con dirección (MAC ID) duplicada.

8.3.2.4 Capa de Aplicación – Protocolo CIP

DeviceNet utiliza el Common Industrial Protocol (CIP) en la capa de aplicación. Se trata de un protocolo estrictamente orientado a objetos, también utilizado por ControlNet y por Ethernet/IP. Es decir, que el medio físico es independiente de la capa de enlace de datos. La Figura 3. Presenta la estructura de este protocolo.

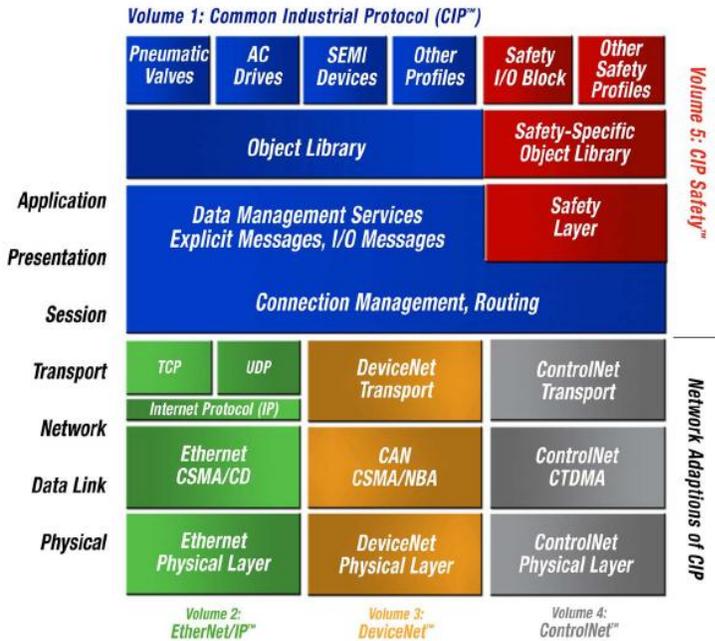


Figura 3. Estructura en capas del protocolo CIP

CIP tiene dos objetivos principales:

- Transporte de datos de control de los dispositivos de I/O.
- Transporte de informaciones de configuración y diagnóstico del sistema controlado.

Un nodo (maestro o esclavo) DeviceNet es entonces modelado por un conjunto de objetos CIP, los cuales encapsulan datos, servicios y a la vez determinan así su comportamiento.

Existen objetos obligatorios (todo dispositivo debe contener) y objetos opcionales. Objetos opcionales son aquellos que moldean el dispositivo conforme la categoría (llamado de perfil) a la que pertenecen, tales como: AC/DC drive, lector de código de barras o válvula neumática. Por ser diferentes, cada uno de estos contendrá un conjunto diferente de objetos.

8.3.3 Modos de Comunicación

El protocolo DeviceNet tiene dos tipos básicos de mensajes I/O y explicit. Cada uno de ellos es adecuado a un determinado tipo de dato:

- I/O: tipo de mensaje síncrono dedicado a desplazamiento de datos prioritarios entre un productor y uno o más consumidores. Se dividen de acuerdo con el método de cambio de datos. Los principales son:
 - ❖ Polled: método de comunicación en que el maestro envía un mensaje a cada uno de los esclavos de su lista (scan list). El esclavo recibe el mensaje y responde prontamente la solicitud del maestro. Este proceso es repetido hasta que todos sean consultados.
 - ❖ Bit-strobe: método de comunicación de donde el maestro envía a la red un mensaje conteniendo en 8 bytes de datos. Cada bit de estos 8 bytes representan un esclavo que es diseccionado y que responde de acuerdo con lo programado.

- ❖ Change of state: método de comunicación donde el cambio de datos entre maestro y esclavo ocurre cuando hay un cambio de valores monitoreados/controlados, en un cierto límite de tiempo. Cuando este límite es atingido. La configuración de esta variable de tiempo es hecha en el programa de configuración de red.
 - ❖ Cyclic: es otro método de comunicación muy parecido al anterior. La única diferencia es en la producción y consumo de mensajes. En este método, todos los cambios de datos ocurre en intervalos regulares de tiempo, independientemente de si hayan sido alterados o no. Este periodo también es ajustado en el software de configuración de red.
- Explicit: tipo de mensaje de uso general y no prioritario. Utilizado principalmente en tareas asíncronas tales como parametrización y configuración del equipamiento.

8.3.3.1 Conjunto de Conexión Predefinidas Maestro/Esclavo

DeviceNet emplea fundamentalmente un modelo de mensajes punto a punto. Sin embargo, es bastante común utilizar un esquema predefinido de comunicación basado en el mecanismo maestro/esclavo. Este esquema emplea un movimiento simplificado de mensajes del tipo I/O muy común en aplicaciones de control. La ventaja de este método esta en los requisitos necesarios para hacerlo.

8.4 Topología

DeviceNet usa la topología maestro/esclavo la cual consta de una línea Troncal y Derivaciones como se puede observar en la Figura 4.

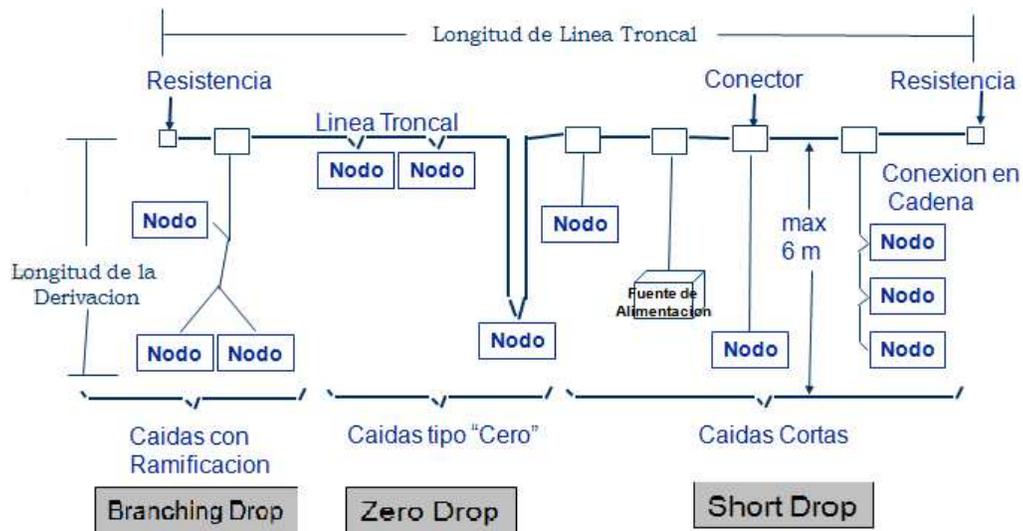


Figura 4. Topología de DeviceNet

La longitud de la línea troncal está definida en función de la velocidad de comunicación (Tabla 3.).

Velocidad de Transmisión de datos	Cable plano (Log. Máxima)	Cable grueso (Log. Máxima)	Cable delgado (Log. Máxima)
(Log. Máxima) 125 Kbps	420 mts	500 mts	100 mts
250 Kbps	200 mts	250 mts	100 mts
500 Kbps	75 mts	100 mts	100 mts

Tabla 3. Velocidad de transmisión en base al cable

La longitud acumulativa de la línea de derivación se refiere a la suma de todas las líneas de derivación, cable grueso o delgado, del sistema de cables. Esta suma no puede superar la longitud acumulativa máxima que se determina en función de la velocidad de transmisión de datos.

El recorrido máximo del cable desde cualquier dispositivo en una línea de derivación a la línea troncal es de 6 mts.

La distancia entre dos puntos no debe superar el recorrido máximo del cable.

8.5 Detalles del Cable

Para armar una red DeviceNet generalmente se utiliza un conductor de 5 hilos; 1 par para alimentación, 1 par para datos y un cable para blindaje. (Figura 5.)

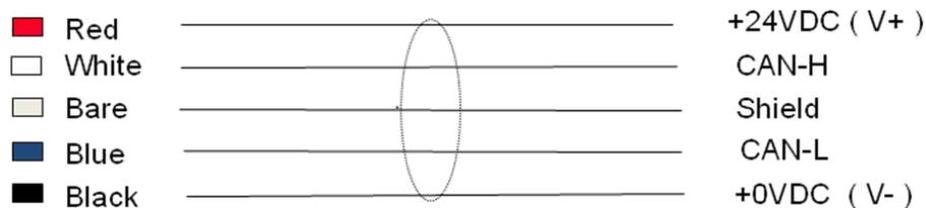


Figura 5. Configuración del cable

La capacidad máxima de suministro de corriente es de 8 amperes y 24 VDC en un solo cable.

Existen cuatro tipos de cable para armar la red.

- Grueso: 12.2 mm diámetro, para línea troncal. Soporta corrientes 8 amperes. (Figura 6.)
- Delgado: 6.9 mm diámetro, es más flexible. Ideal para líneas de derivación. Soporta 3 amperes de corriente. (Figura 6.)
- Plano: Este cable no tiene longitudes predeterminadas y es posible poner conexiones donde se necesiten.
- De derivación (KwikLink): Cable de derivación de 4 conductores, no blindado.

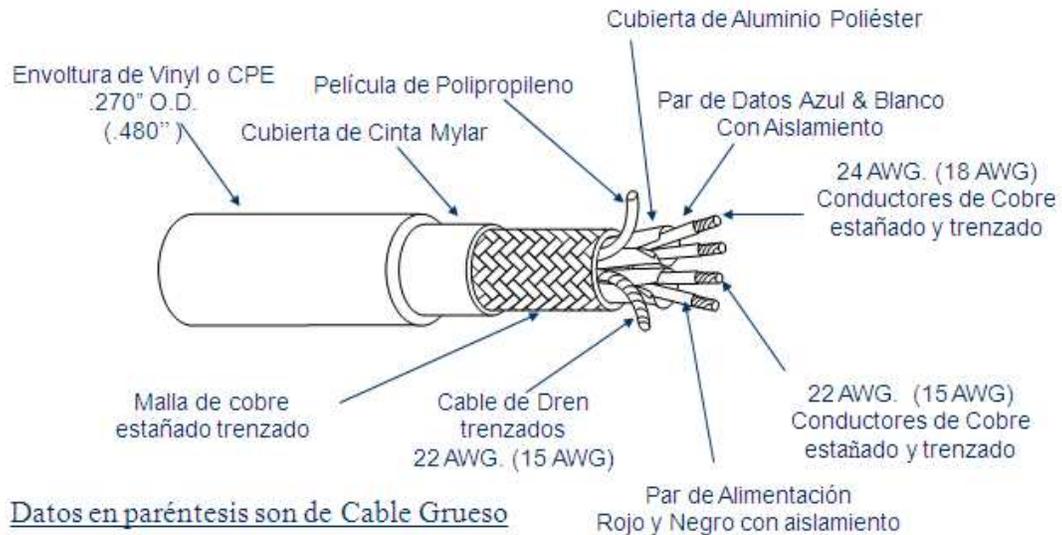


Figura 6. Detalles de cable

8.6 Tap's

Para poder hacer las conexiones entre cables de DeviceNet es necesario utilizar conectores Tap's sellados o bien estilo abierto.

8.6.1 Tap's sellados

Este tipo de Tap's generalmente se utilizan en la instalación de la línea troncal o para segmentos de la misma.

Existen dos tipos: Conectores Micro o Mini, esto es dependiendo del dispositivo.

También hay dos tipos de conectores: Tee (Figura 7) y Multipuerto. (Figura 8)

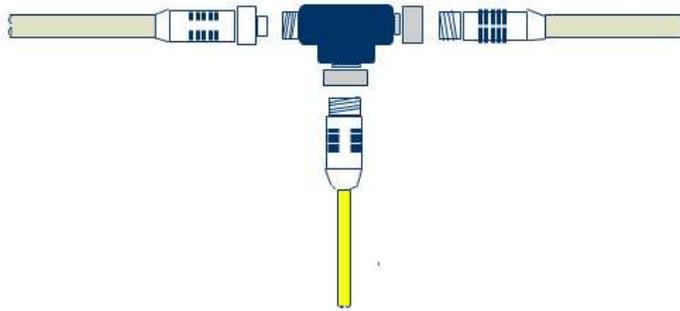


Figura 7. Tap Tee

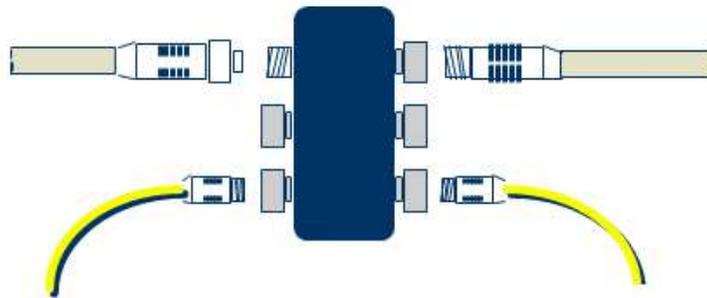


Figura 8. Tap Multipuerto

8.6.2 Tap's abiertos

Este tipo de conectores se utilizan al principio o al final de la línea tronca, también sirven para el conector de la alimentación de la línea. (Figura 9.)

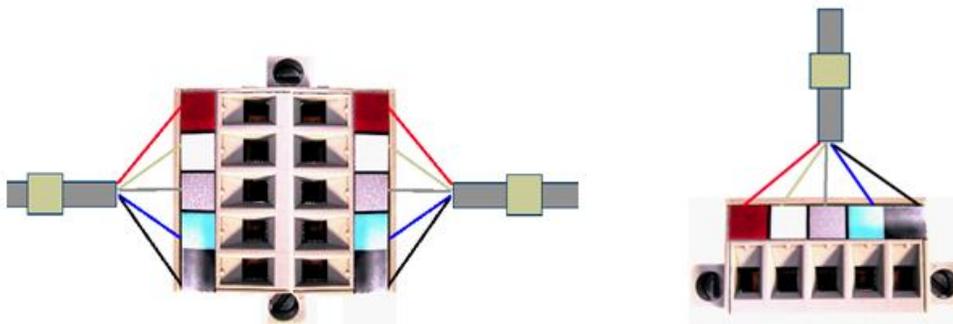


Figura 9. Tap estilo abierto

8.7 INTERFAZ 1770 KFD

El módulo 1770-KFD es una interfaz que se comunica a un ordenador por medio del puerto serial RS-232 para proveer el acceso a DeviceNet network. (Figura 10.)

Este módulo de interfaz permite comunicarse:

- Punto a Punto: El ordenador se comunica directamente con el dispositivo (esclavo) vía módulo 1770-KFD para conectarse por nodo o para cambiar los parámetros de configuración.
- A través de DeviceNet network: Esto permite realizar la transmisión de datos, gestión y diagnóstico de redes locales.

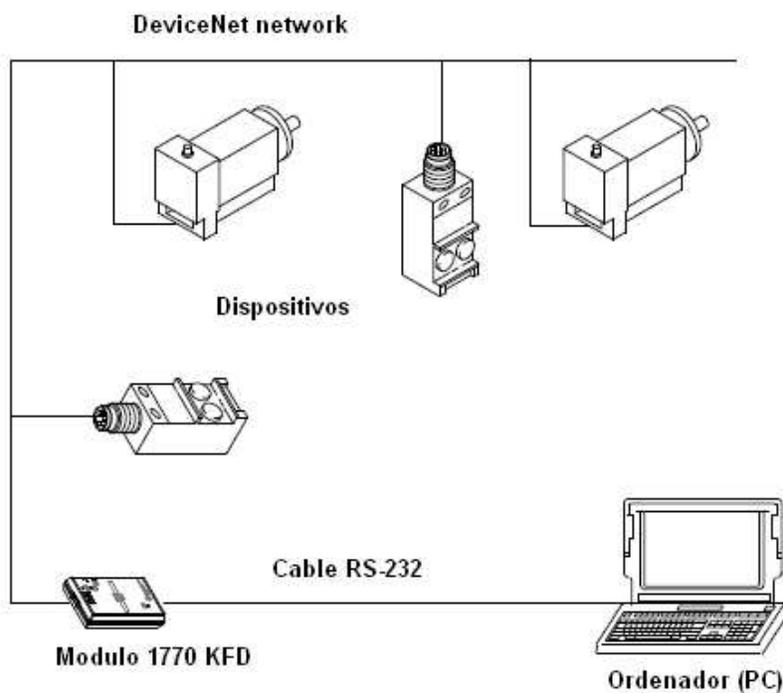


Figura 10. Conexión del módulo en la red

8.8 SCANNER 1769-SDN

El escáner 1769-SDN actúa como interface entre los dispositivos DeviceNet y el PLC.
(Figura 11.)

El escáner se comunica con los dispositivos DeviceNet mediante la red para:

- Leer entradas de dispositivos esclavos.
- Escribir salidas a dispositivos esclavos.
- Comunicarse con dispositivos esclavos (transmisión de mensajes).
- Cargar/descargar programas a un controlador MicroLogix a través de una red DeviceNet.

8.8.1 Características del módulo (Tabla 4.)

Ítem	Descripción
1	palanca de bus (con enclavamiento)
2A	seguro superior para el riel DIN
2B	seguro inferior para el riel DIN
3A	lengüeta superior para montaje en panel
3B	lengüeta inferior de montaje en panel
4	Indicadores LED de estado del módulo y de la red
5	pantalla numérica de dirección y error
6	tornillo de tierra
7A	conector macho de acoplamiento DeviceNet
7B	conector hembra DeviceNet extraíble
8A	conector de bus móvil con pines hembra
8B	conector de bus con pines macho
9	etiqueta de la placa del fabricante

Tabla 4. Descripción del Scanner 1769-SDN

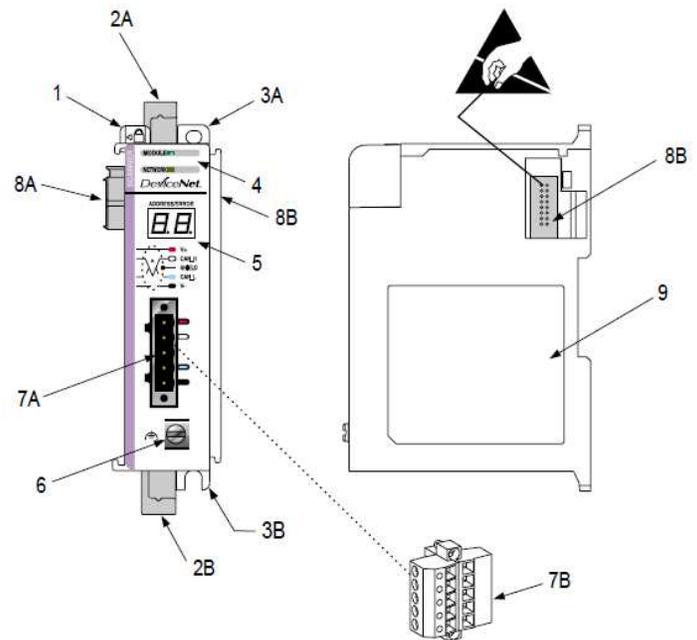


Figura 11. Scanner 1769-SDN

8.9 SENSOR 42EF-D1LDAK-F5

Los sensores fotoeléctricos RightSight DeviceNet hacen interface directamente con esta red industrial sin necesidad de bloques de E/S adicionales, ni adaptadores. Existen dos tipos de sensores Ajustable y No Ajustable (Figura 12)

Características

- Envoltente RightSight compacto
- Capacidad para resistir proyecciones de agua de 1200 psi (8270 kPa)
- Interface directa a la red DeviceNet
- Protocolos de estroboscopio y COS
- Temporizadores de conexión y desconexión/mono impulso
- Contador ajustable con salida
- Detección de movimiento ajustable
- Diagnósticos de umbral de doble margen

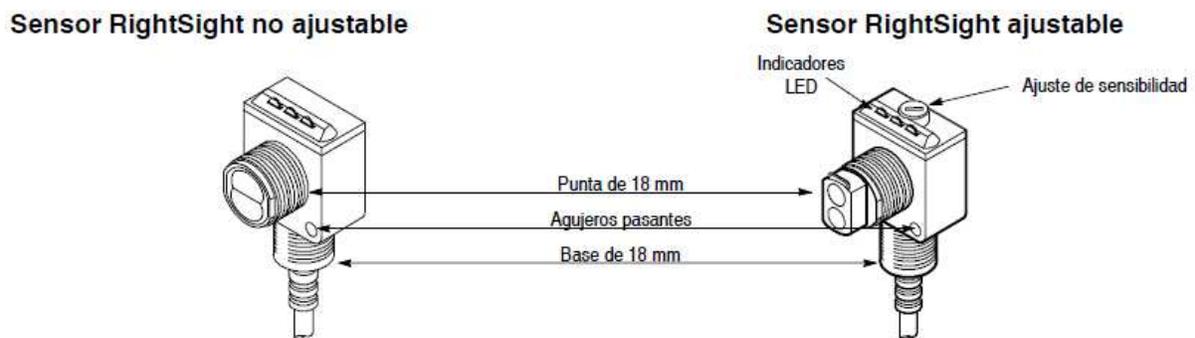


Figura 12. Sensor ajustable y no ajustable

8.10 LIMIT SWITCH 802DN-AD5 (lever type)

Este interruptor funciona por medio de una palanca que se sujeta a un árbol moleteado que se extiende desde que hace un ligero contacto. Este a su vez puede ser equipado con diferentes palancas, dependiendo del uso.

Este limit switch está construido dentro de las características y beneficios de DeviceNet, esto es para atender las tres principales necesidades:

- Aumento del flujo de información
- Una forma fácil y rápida de cambiar el dispositivo
- Reducción del tiempo de parada mediante el uso de diagnóstico avanzado.

Características

- Conexión directa a la red DeviceNet
- Autobaud
- Salidas duales con distintos ángulos de operación
- Aprender y ejecutar nuevos ángulos
- Salidas programables para N.O y N.C.
- Contadores configurables y reset a la salida.

IX. DESARROLLO

9.1 Características y descripción de la máquina.

El propósito de la maquina inyectora de plástico es ser capaz de suministrar la materia prima requerida por el usuario al molde, el cual debe de tener un sistema de enfriamiento apropiado para que el producto se encuentre en buen estado y no pierda sus propiedades y especificaciones indicadas. Los sistemas que componen a la maquina son: Sistema hidráulico, térmico, mecánico, de enfriamiento y de control.

Cuando se aplica calor a un material termoplástico para fundirlo se dice que se plastifica. El material ya fundido o plastificado por calor puede hacer se fluir mediante presión y llenar un molde donde el material solidifica y toma forma del molde. Este proceso se le nombra moldeo por inyección.

9.1.1 El principio básico de la maquina inyectora comprende las tres operaciones siguientes.

I.- Elevar la temperatura del plástico a un punto donde pueda fluir bajo la aplicación de presión. Normalmente esto se hace calentando los gránulos sólidos del material hasta formar una masa fundida con una viscosidad y temperatura uniforme. Actualmente, esto se hace dentro del barril de la máquina.

II.- Permitir la solidificación del material e el molde cerrado. En esta etapa el material fundido ya plastificado, se transfiere a la parte inferior del cañón o sea a la boquilla, que inyecta hacia los varios canales del molde hasta llegar a las cavidades donde toma la forma del producto final.

III.- Apertura del molde para la extracción de la pieza. Esto se hace después de mantener el material bajo presión dentro del molde y una vez que el calor es removido para permitir solidificar el material en la forma deseada.

A continuación se presenta el esquemático de una maquina inyectora de plástico (Figura 13.)

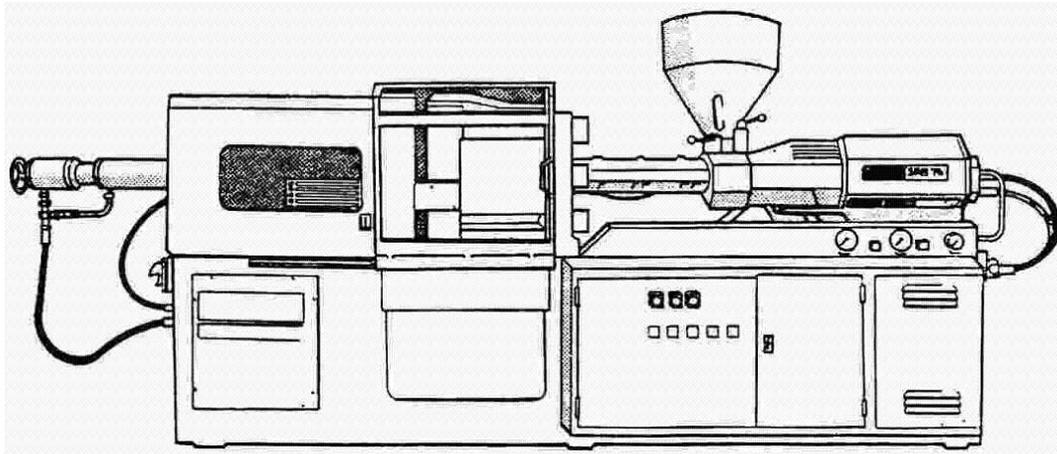


Figura 13. Maquina inyectora de plástico

Como podemos observar en la figura anterior esta máquina no cuenta con una red de monitoreo; El propósito de este trabajo es acondicionar una red DeviceNet a la maquina inyectora.

A continuación un esquemático en donde se representara la ubicación de los componentes de la red. (Figura 14.)

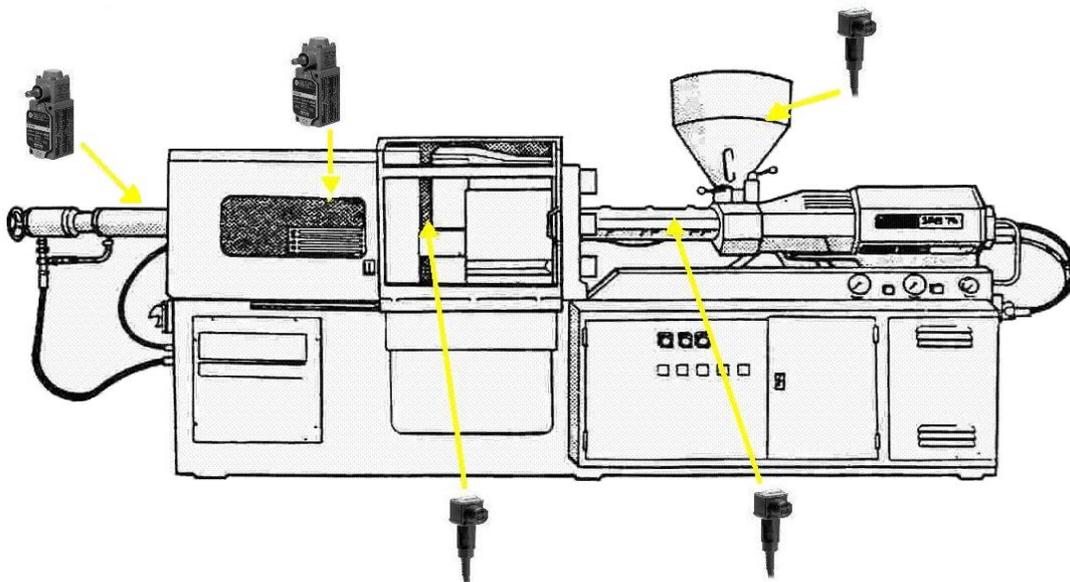


Figura 14. Maquina inyectora de plástico con sus dispositivos

Los elementos utilizados en esta red son:

- 3 Sensores Fotoeléctricos 42EF-D1LDAK-F5
- 2 Limit Switch 802DN-AD5
- 5 mts de cable troncal
- 6 mts de cable de derivación
- 1 PLC Micrologix 1500
- 1 Modulo 1770 KFD
- 1 Fuente de Alimentación para DeviceNet
- 1 Scanner 1769 SDN
- Conectores Tap's

A continuación un esquemático del armado de la red. (Figura 15.)

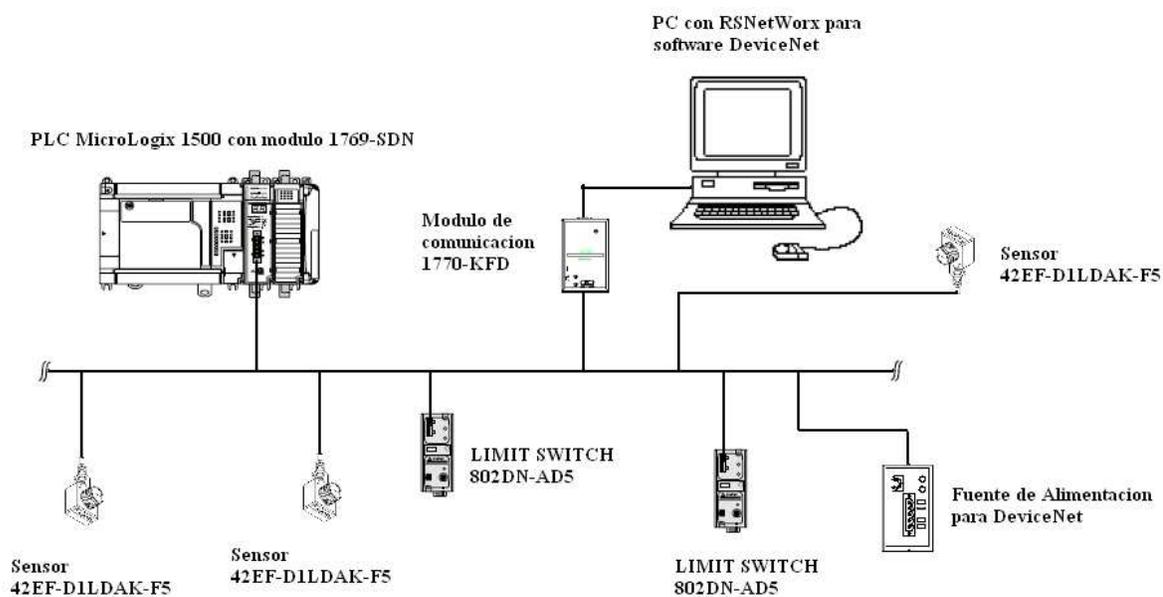


Figura 15. Esquemático de la red DeviceNet

9.2 Armado de la red DeviceNet.

Lo primero que se necesita hacer es la conexión física entre la PC y el modulo de comunicación 1770-KFD por medio del puerto RS 232, se debe tener mucho cuidado al hacer esta conexión pues al momento de conectar podemos doblar los pines de conector y esto puede afectar la comunicación entre el modulo y la PC.

Ahora procederemos a conectar el Power Tap, este es un dispositivo el cual va conectado a la fuente de alimentación y al modulo 1770-KFD. Para poder conectar el Power Tap a la fuente de alimentación es necesario ponerle un conector Tap estilo abierto, en la Figura 16. se muestra la configuración de la conexión.

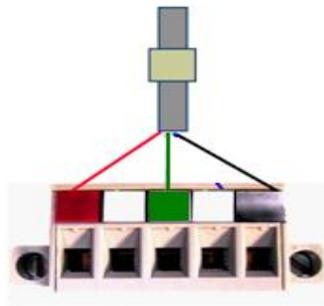


Figura 16. Tap estilo abierto con conexión para la alimentación

Este dispositivo también va conectado al modulo 1770-KFD ya que por medio de este es como se comunica la red a la PC, para esta conexión se necesita un Tap estilo abierto.

Los Tap's estilo abierto ya traen la configuración en código de colores como se muestra en la Figura 17.

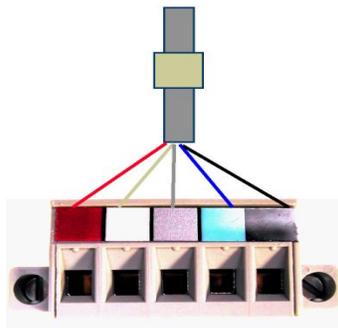


Figura 17. Tap estilo abierto con configuración

Es importante destacar que el Power Tap ya trae un tramo de cable para que de ahí se comiencen a hacer las conexiones de la red troncal, también trae otro tramo de cable al cual se le conecta una resistencia de 120 Ω . (Figura 18.)

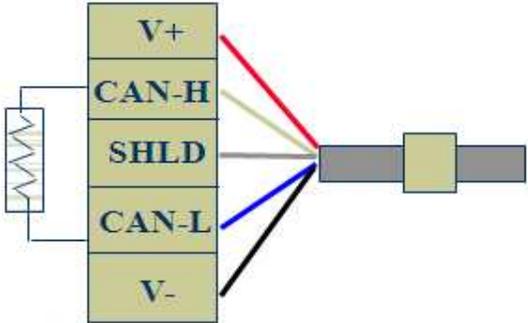


Figura 18. Tap estilo abierto con resistencia terminal

Hecho todo lo anterior se procede a conectar un Tap Tee para que de este se pueda hacer una derivación y a la vez sea la continuación de la línea Troncal. La derivación va al primer Limit Switch 802DN-AD5. (Figura 19.)

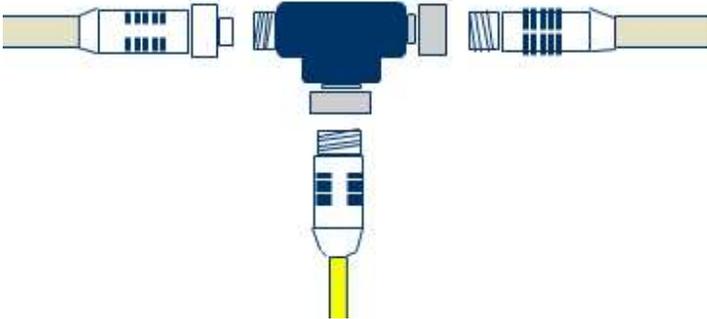


Figura 19. Tap Tee para la derivación

Al terminar de conectar el Limit Switch a la derivación que pertenece, se continúa con la conexión del otro tramo de la línea Troncal al Tap Tee, esto es para hacer que la línea troncal se extienda los más cerca de donde va a ir ubicada la próxima derivación.

Ya que la línea troncal esté bastante cerca de donde va a salir la derivación colocamos otro Tap Tee para poder hacer la derivación que va dirigida al próximo Limit Switch 802DN-AD5, y se continua poniendo más cable para la line troncal.

El procedimiento de colocación de Tap's Tee es repetitivo, pues para cada derivación se ocupara un Tap.

De la misma manera se extiende la red hasta estar lo más cerca de donde se va a colocar el siguiente dispositivo que es un sensor fotoeléctrico 42EF-D1LDAK-F5; Colocamos un Tap Tee para la derivación de la línea troncal.

El paso anterior se va a repetir otras dos veces más, ya que como se mencionó anteriormente se van a utilizar 3 sensores fotoeléctricos.

Para poder terminar y cerrar la línea troncal es necesario colocar otra derivación con un Tap Tee, al final de este derivación se debe colocar una resistencia de 120Ω en un Tap estilo abierto, esto es necesario ya que de esta forma se indica el final de la red DeviceNet.

9.3 Configuración de la Interfaz 1770 KFD a través de RSLinx

RSLinx es el software que permite establecer una comunicación entre la PC y la interfaz que se utilizara. Para esto lo primero que se debe hacer es ingresar al software RSLinx dando doble clic sobre el icono correspondiente al programa (Figura 20.).



Figura 20. Icono de RSLinx

Una vez dentro del programa, este mostrara la ventana principal del software en la que se despliegan un menú de opciones, para iniciar la configuración del driver (o controlador) se debe dar un clic sobre el icono "Configure Drivers". (Figura 21.)

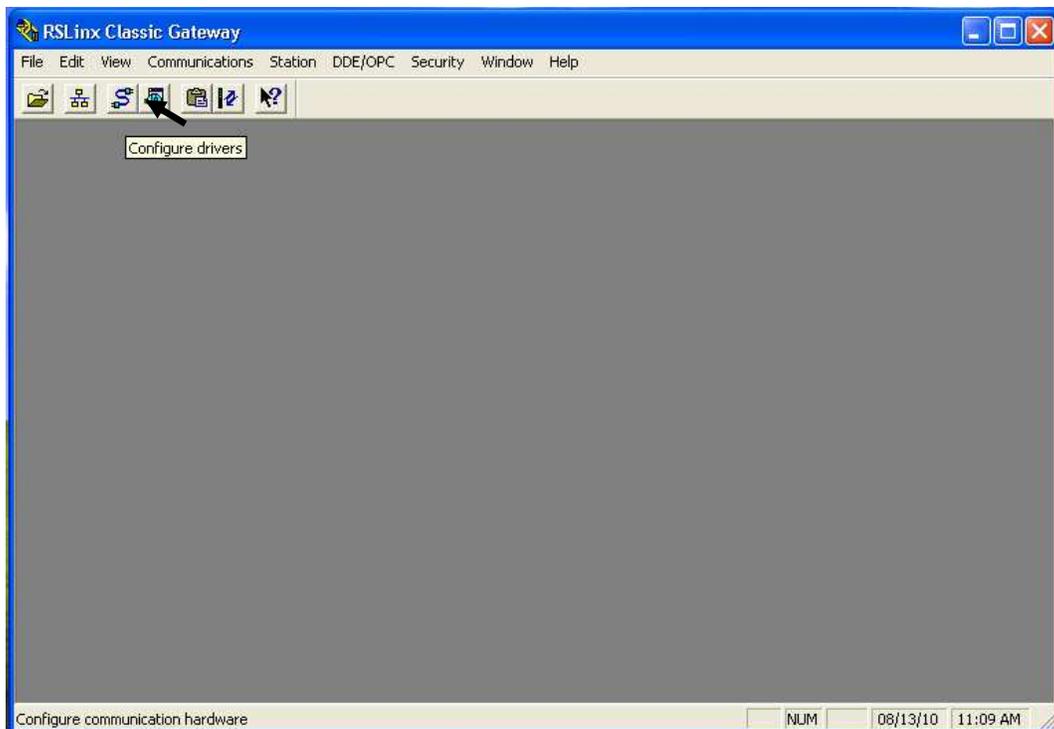


Figura 21. Ventana "Configure Drivers"

Enseguida se arrojará una nueva ventana (Figura 22.) en la cual se podrá generar el driver que utilizará la interfaz de la red.

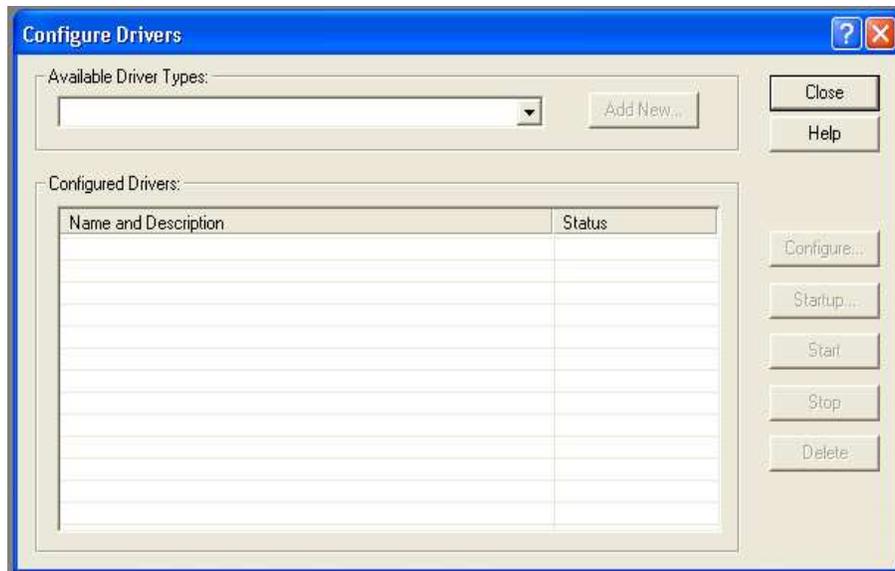


Figura 22. Ventana para configurar controladores

Para crear el controlador se da un clic sobre la opción “Available Driver Types”; la cual sirve para agregar el controlador (driver) que utilizará, ya que en esta opción se despliega una lista de los diferentes tipos de drivers que se pueden configurar con RSLinx.(Figura 23)

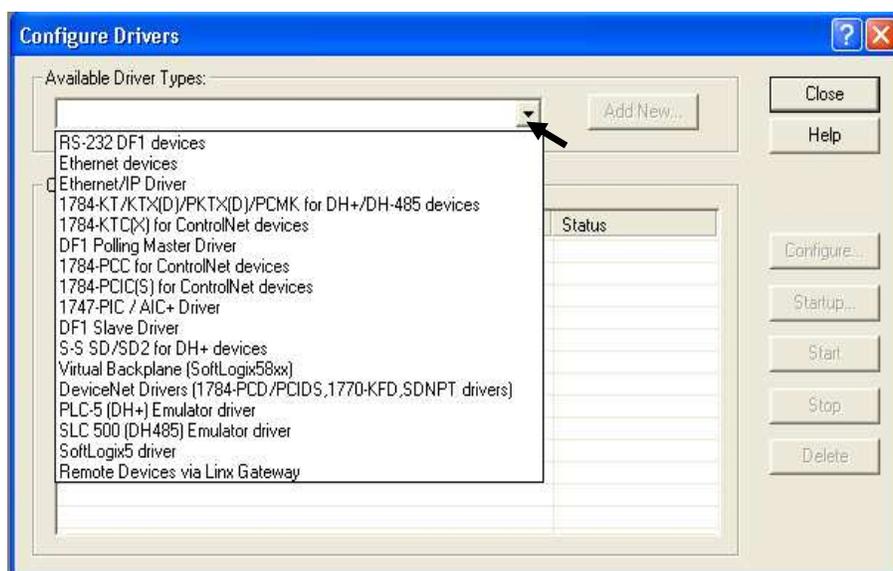


Figura 23. Lista de los distintos tipos de controladores "Drivers"

Para este caso se utilizara la opción de “DeviceNet Drivers” (ya que esta es la que incluye el driver de la interfaz a utilizar), y después de seleccionar esta opción se debe dar un clic sobre “Add New”. (Figura 24.)

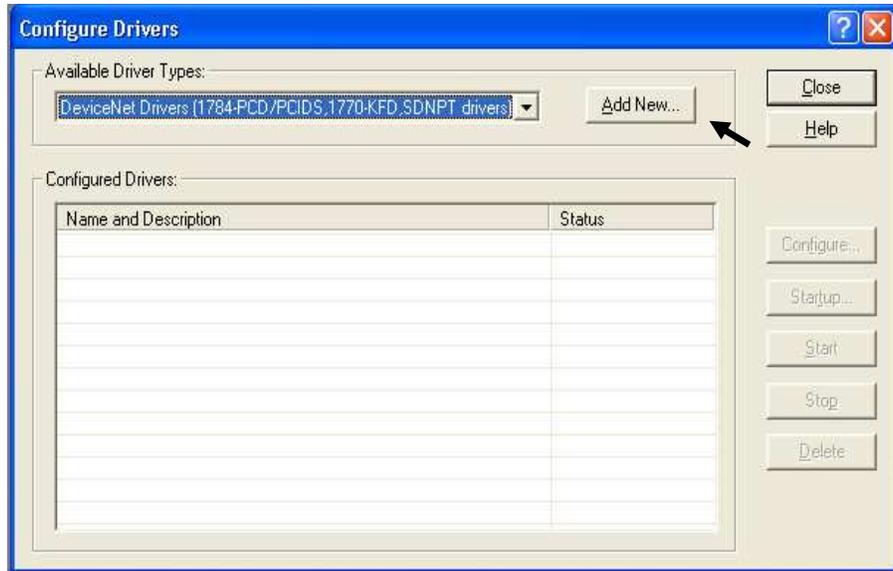


Figura 24. Selección del driver a utilizar

Después de dar clic sobre la opción “Add New” se mostrara una nueva ventana en la cual se debe de seleccionar el tipo de interfaz a utilizar para la red DeviceNet que se construirá, en este caso seleccionaremos la opción “Allen-Bradley 1770-KFD” y en seguida dar clic sobre el botón “Select”. (Figura 25.)



Figura 25. Selección de la interfaz

Ahora el programa arrojará una nueva ventana (Figura 26.) en la cual debemos de elegir las siguientes características:

- Serial Port Setup
 - ❖ Port Select (número de puerto serial por el cual se hará la comunicación), en este caso se utilizará el COM 1 de la PC.
 - ❖ Data Rate (velocidad de transmisión de los datos), se trabajará con la de 9600.

- DeviceNet Port Setup
 - ❖ Node Address (el número de nodo que utilizará la interfaz dentro de la red DeviceNet), en este caso lo colocaremos como el nodo número 62.
 - ❖ Data Rate (velocidad de transmisión de los datos dentro de la red), se colocará la de 125 K.

Después de ingresar los parámetros necesarios solo se debe dar un clic sobre el botón “OK”.



Figura 26. Configuración del controlador 1770-KDF

Después de haber presionado la opción “OK” aparecerá en pantalla la ventana que se muestra en la Figura 27., la cual aparecerá parpadeando unos cuantos segundos.



Figura 27. Inicialización de la comunicación

Si la conexión física entre la interfaz y la PC está bien, aparecerá una nueva ventana en la cual se debe colocar el nombre del controlador (Figura 28.), se dejara el nombre que en automático arroja y ahora se debe de dar un clic sobre la opción “OK”.



Figura 28. Ventana para el nombre del controlador

Enseguida aparecerá de nuevo la ventana “Configure Drivers”, si todo quedo bien entonces el controlador que se configuro aparecerá en la tabla “Configure Drivers” con el nombre que se le coloco, la velocidad de transmisión y el “Status” del mismo (Figura 29.).

Si el controlador que se creó aparece con el Status de “Runnig” eso significa que el controlador ya está funcionando y que ya se puede establecer una comunicación entre la Interfaz y la PC, y por tanto ya está lista la interfaz para poder a iniciar la configuración de la red DeviceNet dentro de RSNetworks lo cual se explicara más adelante.

Para finalizar dar clic sobre la opción “Close” de la Figura 29.

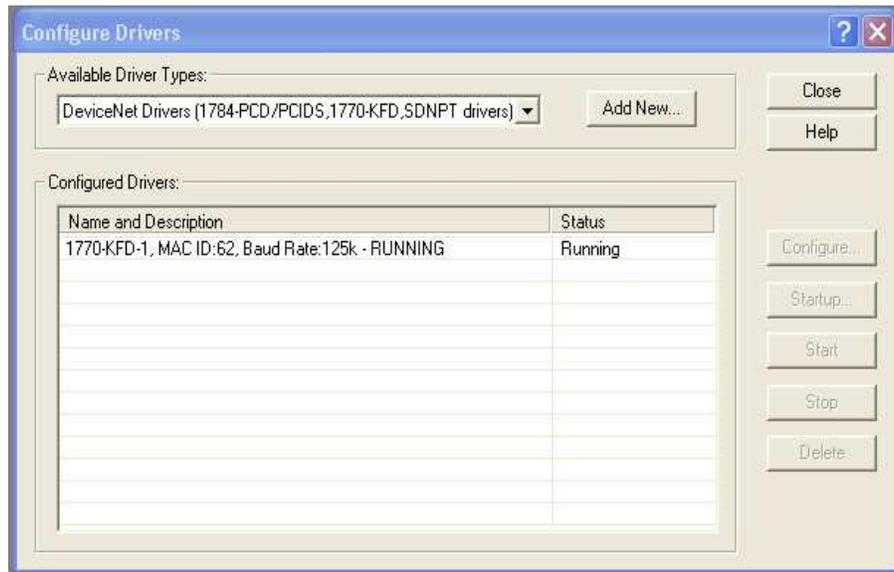


Figura 29. Status del controlador

Si se desea se puede comprobar que la comunicación entre la Interfaz y la PC está establecida, para ello en la venta principal de RSLinx se debe dar un clic sobre la opción RSWho (Figura 30.), dicha opción arrojará una nueva ventana en la cual se muestran los controladores que están activos en este caso deberá aparecer la interfaz que se configuro anteriormente (Figura 31.), si es así entonces todo está bien y se procede a cerrar la ventana del programa RSLinx.

(Nota. Se debe cerrar el programa pero no se debe de apagar, de lo contrario el controlador se deshabilitara).



Figura 30. Ventana principal de RSLinx

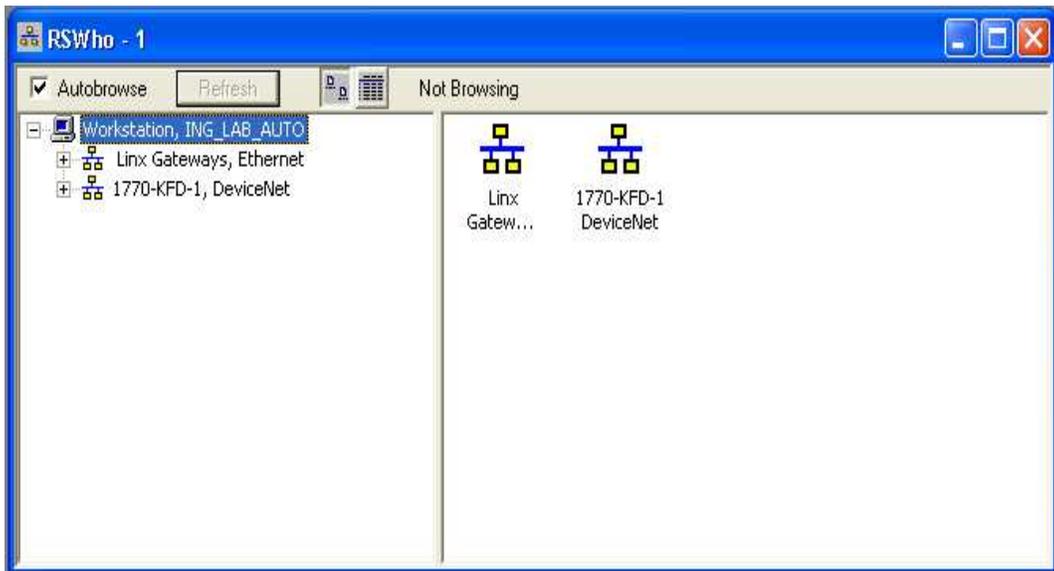


Figura 31. RSWho

9.4 Configuración del controlador del PLC a través de RSLinx

RSLinx es el software que permite establecer una comunicación entre la PC y la interfaz que se utilizara. Para esto lo primero que se debe hacer es ingresar al software RSLinx dando doble clic sobre el icono correspondiente al programa. (Figura 32.)



Figura 32. RSLinx

Una vez dentro del programa, este nos arrojará la ventana principal del software en la que se despliegan un menú de opciones, para iniciar la configuración del driver (o controlador) se debe dar un clic sobre el icono “Configure Drivers”. (Figura 33.)

Enseguida se desplegará una nueva ventana (Figura 34.) en la cual se podrá generar el driver que utilizará el PLC.



Figura 33. Ventana principal de RSLinx

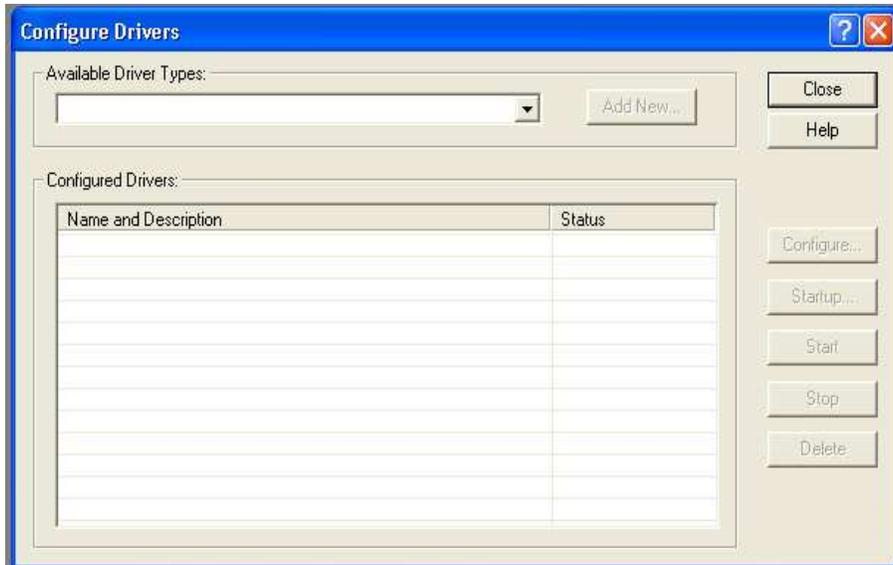


Figura 34. Configurar controladores "Configure Drivers"

Para crear el controlador se debe dar un clic sobre la opción “Available Driver Types”; la cual sirve para agregar el controlador (driver) que se va a utilizar, ya que en esta opción se despliega una lista de los diferentes tipos de drivers que se pueden configurar con RSLinx. (Figura 35.)

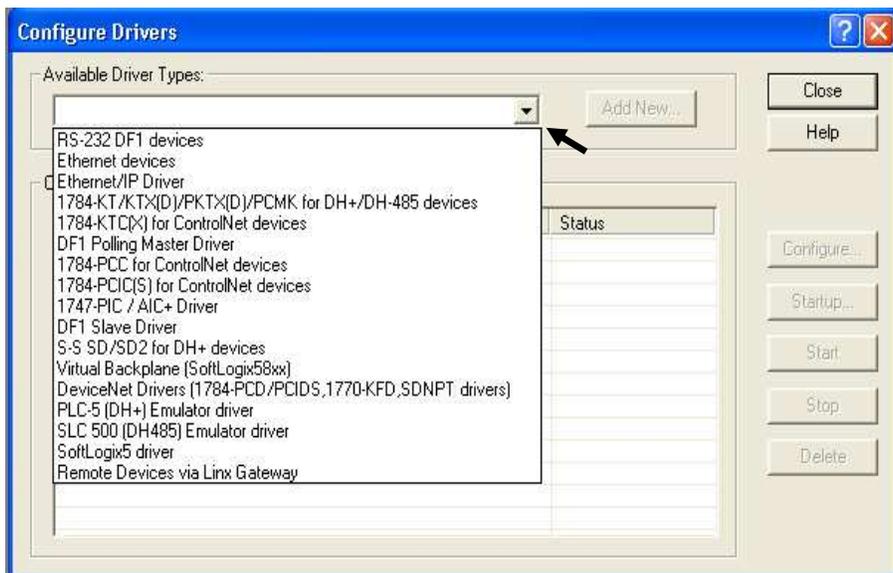


Figura 35. Lista de los tipos de controladores

De la lista hay que seleccionar la opción de RS-232 DF1 ya que esta es el tipo de comunicación que utiliza el PLC, y después de seleccionar esta opción se debe dar un clic sobre “Add New”. (Figura 36.)

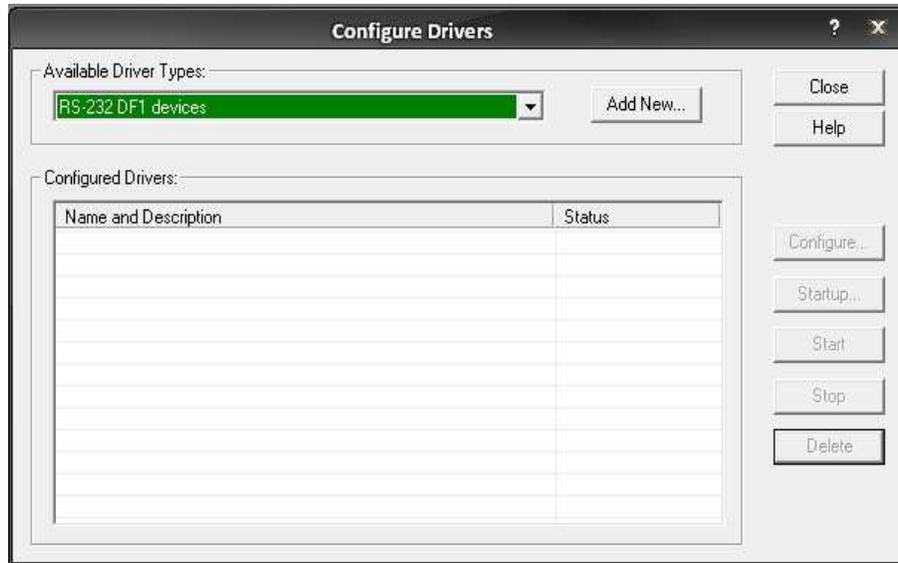


Figura 36. Selección del driver a utilizar

Al presionar el botón de “Add New” aparecerá una opción en la cual se debe colocar el nombre que tendrá el controlador, en este caso dejaremos el que da en automático el programa “AB_DF1-1” (Figura 37.), dar clic sobre el botón “OK”.

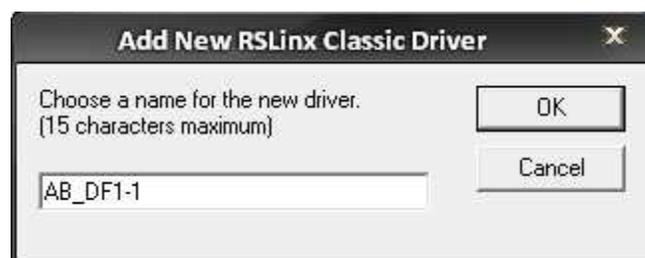


Figura 37. Nombre del controlador

Aparecerá una nueva ventana (Figura 38.) en la cual se configura las características del controlador (driver):



Figura 38. Ventana de configuración del controlador

- Comm Port (número de puerto por el cual se hará la comunicación), en este caso se trabajara con el COM1 de la PC.
- Device (el tipo de controlador), colocar la opción de PLC-CH0

Enseguida dar clic en la opción de “Auto-Configure” de la ventana de la Figura 39 para que se realice de forma automática la configuración del controlador.

Si la conexión física entre la PC y el PLC está bien elaborada aparecerá el mensaje “Autoconfiguration Succesfull” (Figura 39.) si es así entonces proceder a dar clic en la opción “OK”.

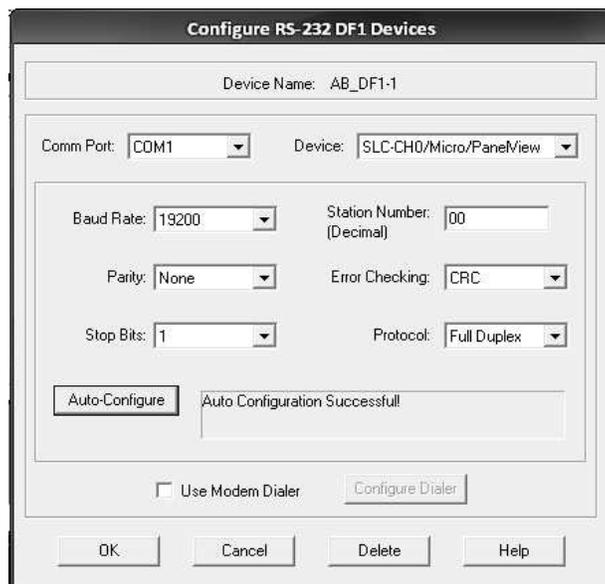


Figura 39. Auto-Configure

Después de dar clic en la opción “OK” el programa regresara a la ventana “Configure Drivers”, en dicha ventana en la opción de “Configured Drivers” el nombre del controlador que fue creado con el “Status” de “Runnig” (Figura 40.) dar clic en “Close”.

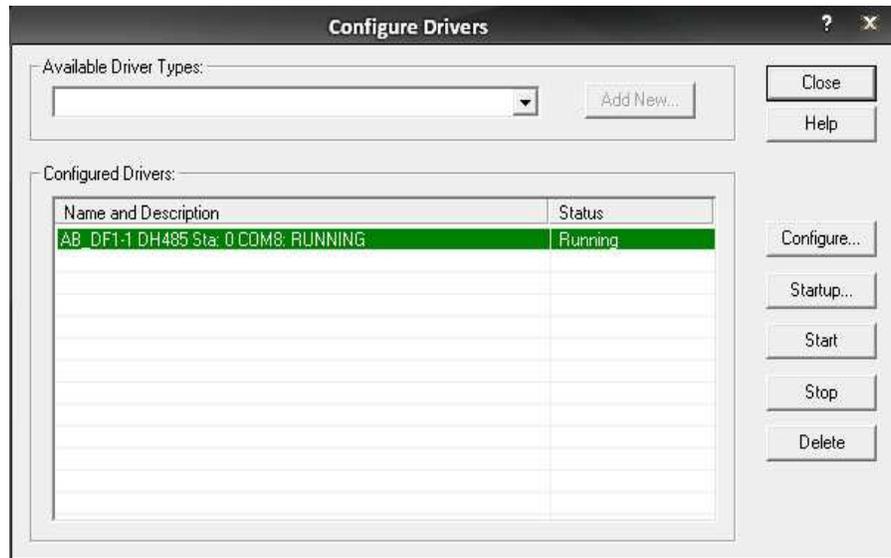


Figura 40. Status del controlador

Si se desea se puede comprobar que la comunicación entre el PLC y la PC está establecida, para ello en la venta principal de RSLinx dar un clic sobre la opción RSWho (Figura 41.), dicha opción nos arrojará una nueva ventana en la cual se muestran los controladores que están activos en este caso deberá aparecer el controlador que configuramos anteriormente (Figura 42.), si es así entonces todo está bien y se procede a cerrar la ventana del programa RSLinx.

(Nota. Se debe cerrar el programa pero no se debe de apagar, de lo contrario el controlador se deshabilitara).



Figura 41. Opción RSWho

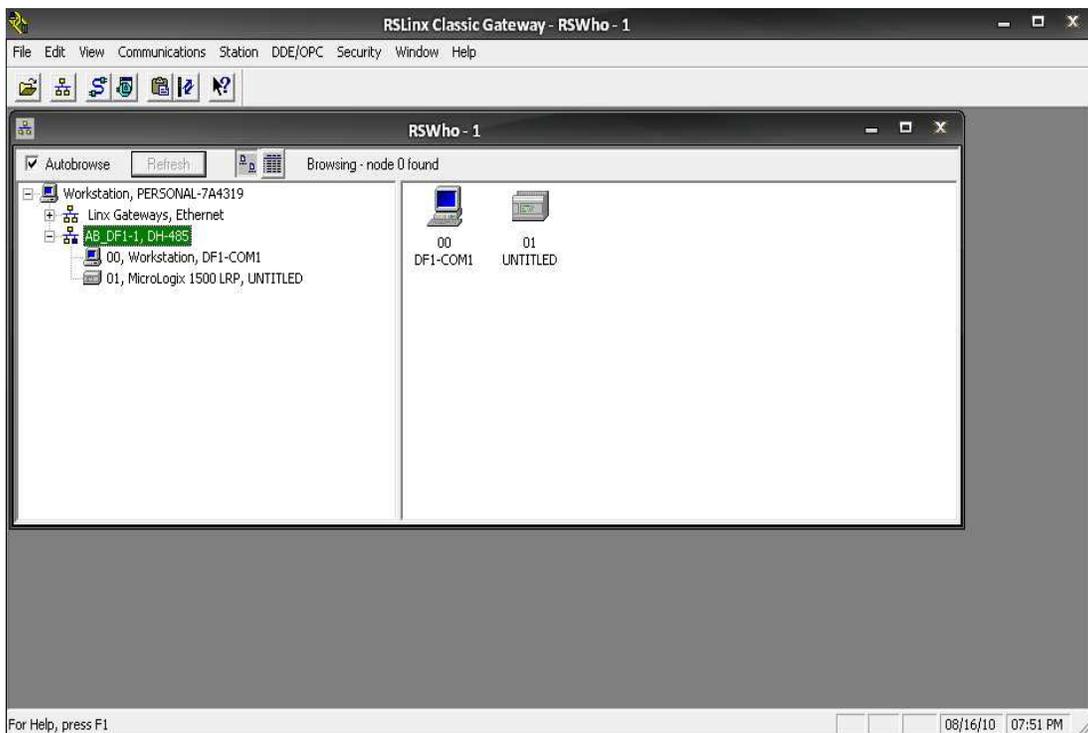


Figura 42. Ventana de RSWho

9.5 Configuración y ajustes principales de RSNetWorx para la red DeviceNet

RSNetWorx es el software que permite configurar los dispositivos que estén conectados a la red, así mismo este programa es el que permite definir que dispositivo será el “maestro” y cuales dispositivos serán los “esclavos” en la red. Para iniciar hay que hacer doble clic en el icono del software RSNetWorx. (Figura 43.)



Figura 43. RSNetworx

Ya que se ingrese al programa se mostrara en pantalla la ventana principal, en la que se despliega el menú de opciones (Figura 44.), en la parte derecha de dicha ventana es en la cual se colocaran los dispositivos que formaran la red.

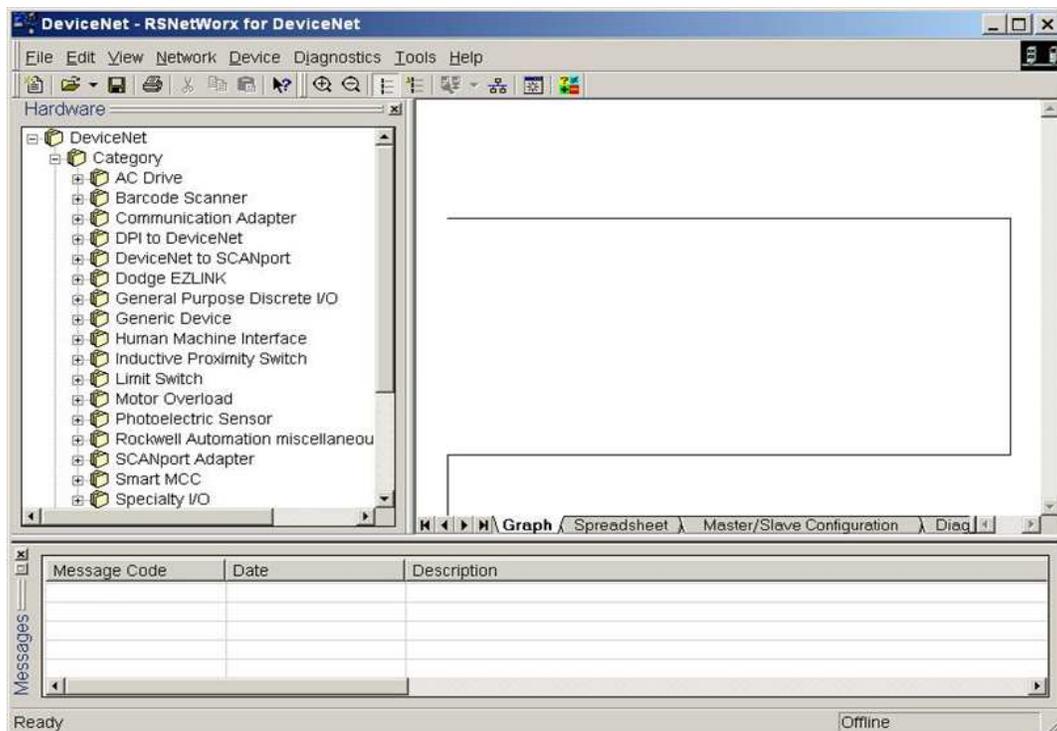


Figura 44. Ventana principal de RSNetworx

Para comenzar se debe estar en línea con la interfaz configurada en la Sección 6.3, dar clic en la opción “Online” la cual se encuentra en la barra de herramientas del programa (Figura 45.) o bien con la tecla F10 y dar clic en la opción “Aceptar”.

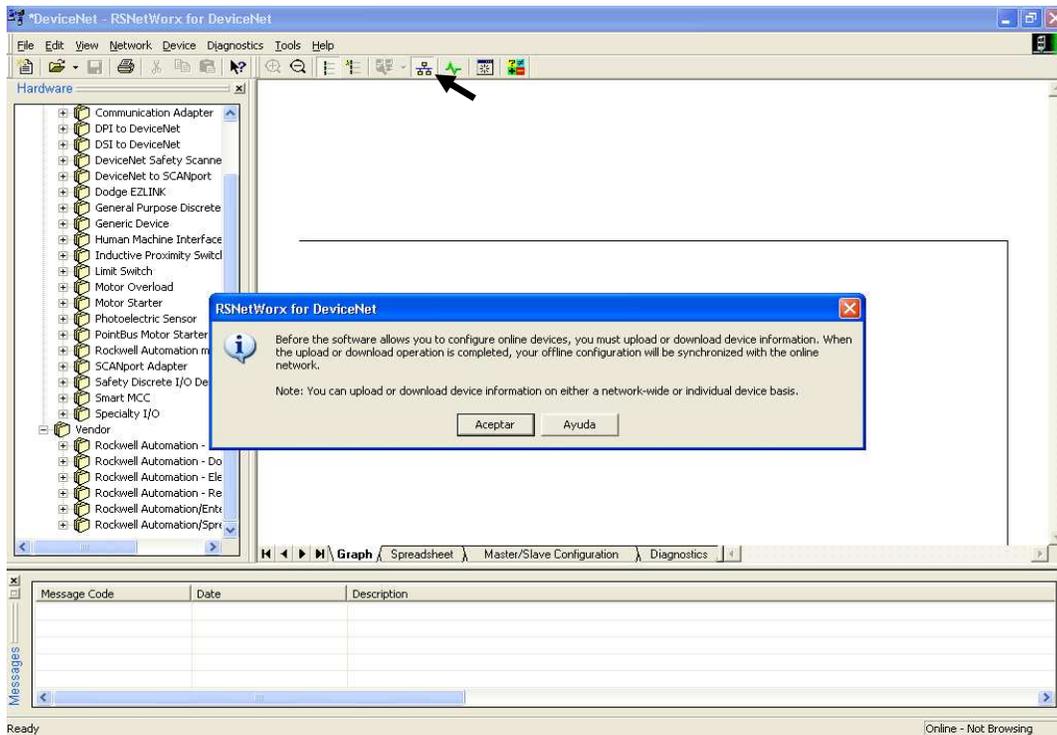


Figura 45. Opción "Online"

Aparecerá una nueva ventana en la cual se debe dar doble clic en el nombre del controlador que fue configurado anteriormente en RSLinx “1770-KFD-1”, y después dar clic en el botón “OK”. (Figura 46.)

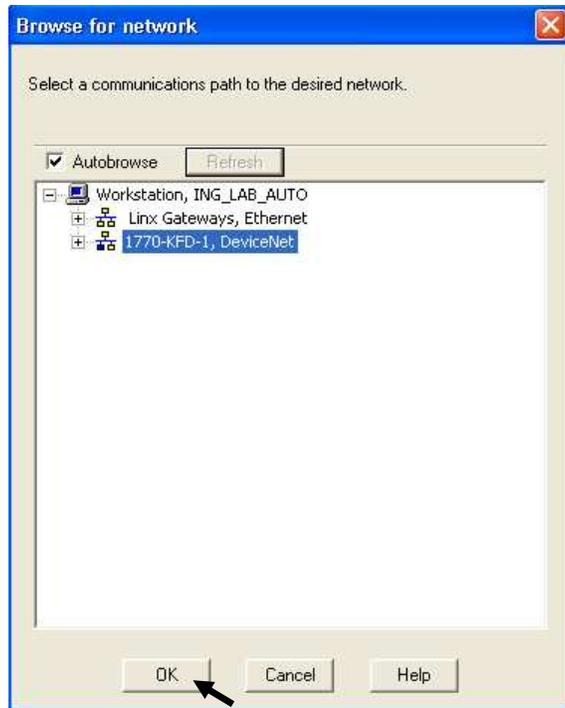


Figura 46. Selección de la interfaz

Aparecerá en pantalla una nueva ventana en la cual se debe dar clic en la opción “Aceptar”, enseguida el programa comenzara a través de la interfaz a buscar de forma automático los dispositivos que se encuentran conectados. (Figura 47.)



Figura 47. Escaneado de la red

Se mostraran en pantalla los dispositivos que están conectados a la red. (Figura 48.)

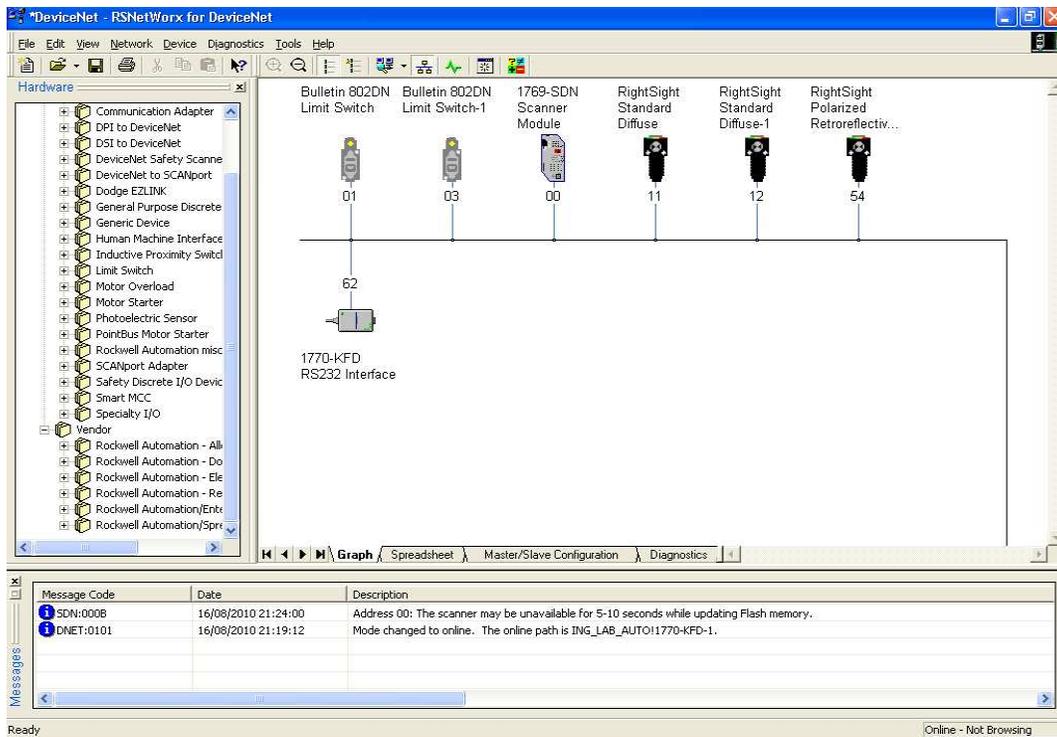


Figura 48. Dispositivos conectados en la red

Ahora se procederá a configurar el scanner como el “Maestro” de la red, dando doble clic sobre el icono del scanner, se mostrara la ventana de propiedades del dispositivo, aparecerá otra ventana, dar clic en la opción “Upload”. (Figura 49.)



Figura 49. Configuración del Scanner

El scanner comenzara a hacer una identificación de los dispositivos que se encuentran conectados físicamente en la red. (Figura 50.)



Figura 50. Estado de la red

Terminado el escaneo, dar clic en la opción “Scanlist”, dar clic en el botón “>>” para que los dispositivos conectados a la red se registren en el “Scanlist” del scanner. (Figura 51.)

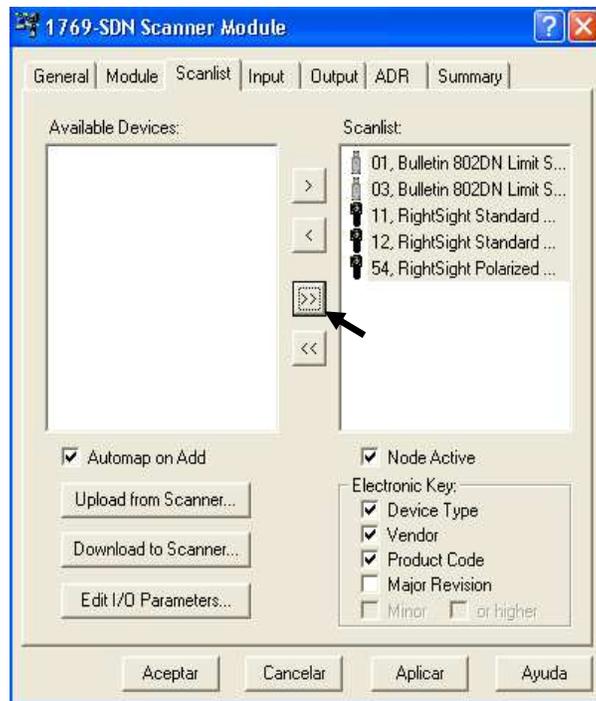


Figura 51. Ventana de la opción "Scanlist"

En la misma ventana de propiedades se puede consultar las direcciones que utilizaran los dispositivos en el Scanner y posteriormente en el PLC para esto dar un clic sobre la opción de “Input”. (Figura 52.)

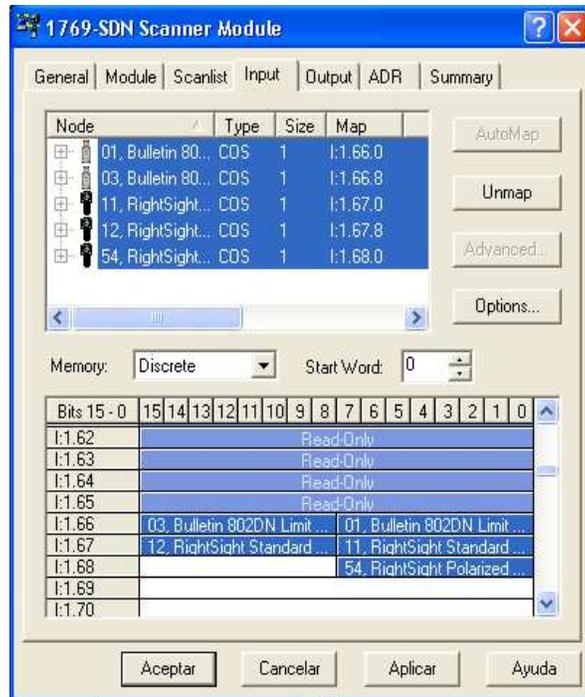


Figura 52. Ventana de la opción "Input" del scanner

Otra función muy útil de la ventana de propiedades del scanner es que también permite ver si todos los dispositivos están activados, dar un clic sobre la opción "Summary" (Figura 53.) en la ventana de propiedades del scanner, dar clic en la opción "Aplicar" y enseguida proceder a cerrar la ventana de propiedades del scanner.

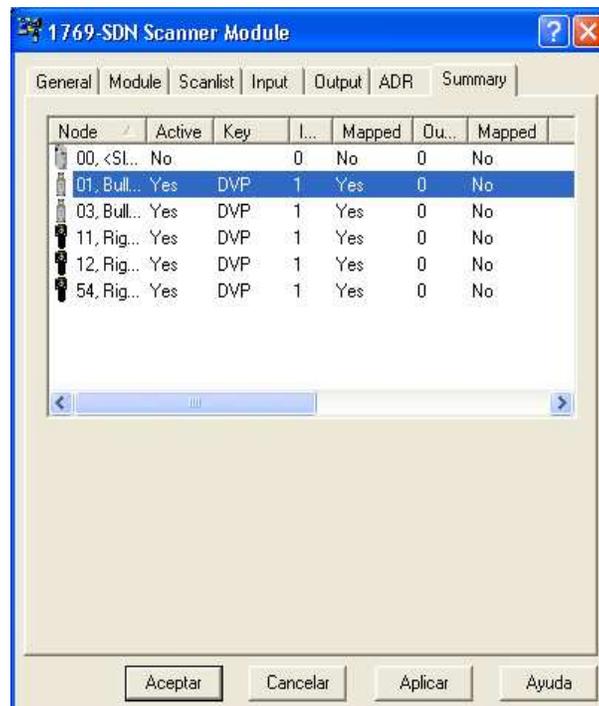


Figura 53. Ventana de la opción "Summary"

Si todo fue realizado con éxito el scanner físicamente tendrá el código “00” lo cual confirma que la red está bien configurada. Pero en caso de que el scanner marque el código “80” debemos de proceder a encenderlo, la manera de hacer esto se explica más adelante en la unidad 4 y 5, ya que se requerirá usar un PLC para poder encender el scanner.

Antes de cerrar el programa debemos guardar la configuración de la red, ingresamos a la opción “File” y después seleccionamos “Save As”, colocar un nombre a la configuración y dar clic en “Aceptar”. Posteriormente se puede proceder a cerrar el programa.

9.6 Configuración y ajustes del Scanner con RSLogix 500

RSLogix es el software que permite encender el scanner que se utilizara en la red DeviceNet (este procedimiento solo aplicara en caso de que el Scanner este apagado es decir cuando este marque “80”).

Una vez que se activa el controlador del PLC tal como se indic  en la Secci n 6.4 se podr  activar el Scanner “1769 SDN” que se utiliza en la red DeviceNet. Abrir el programa RSLogix (Figura 54.) dando doble clic sobre el icono del programa.



Figura 54. RSLogix 500

Una vez que se ingresa al programa este desplegara la ventana principal, la cual contiene un men  de opciones. (Figura 55.)

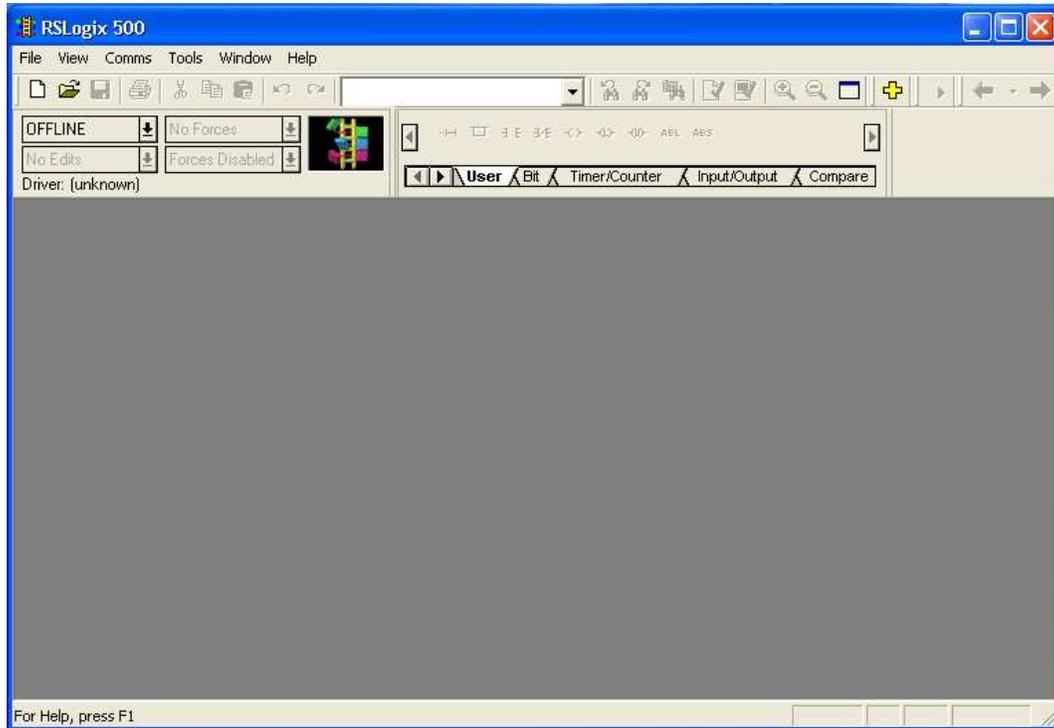


Figura 55. Ventana principal de RSLogix 500

Para iniciar en la ventana principal elegir la opción “New” (Figura 56.) la cual desplegara una lista en la cual se muestran los distintos tipos de PLC que se pueden programar con este software, de la lista seleccionar “Micrologix 1500 LRP Series B”. (Figura 57.)

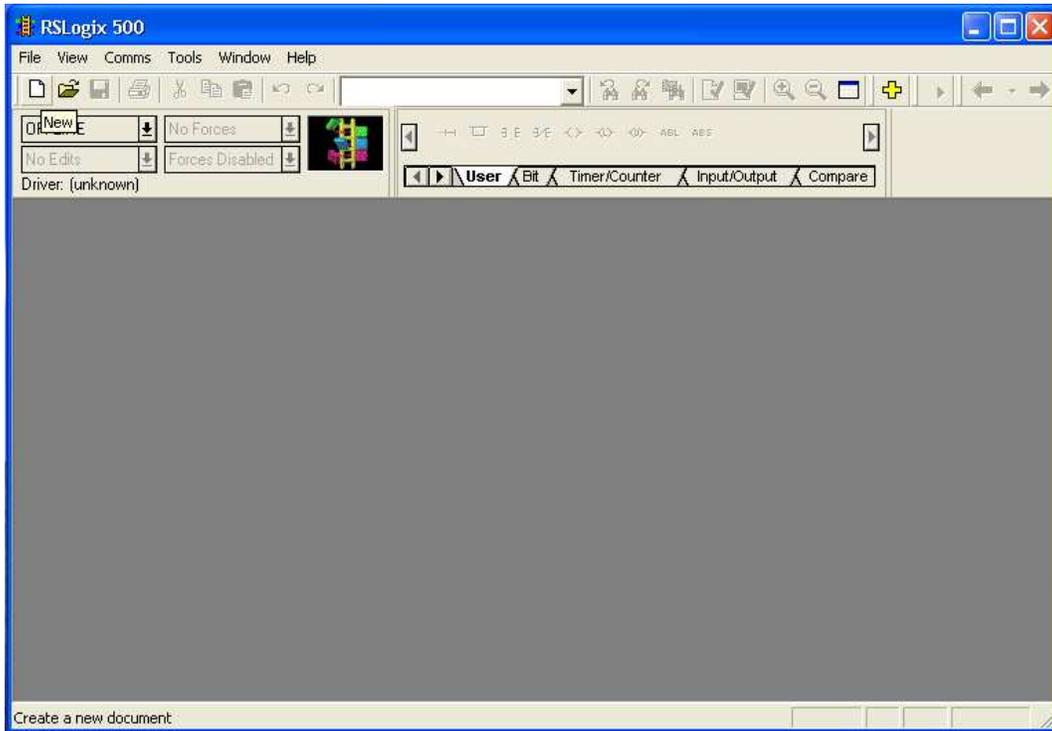


Figura 56. Opción "New"

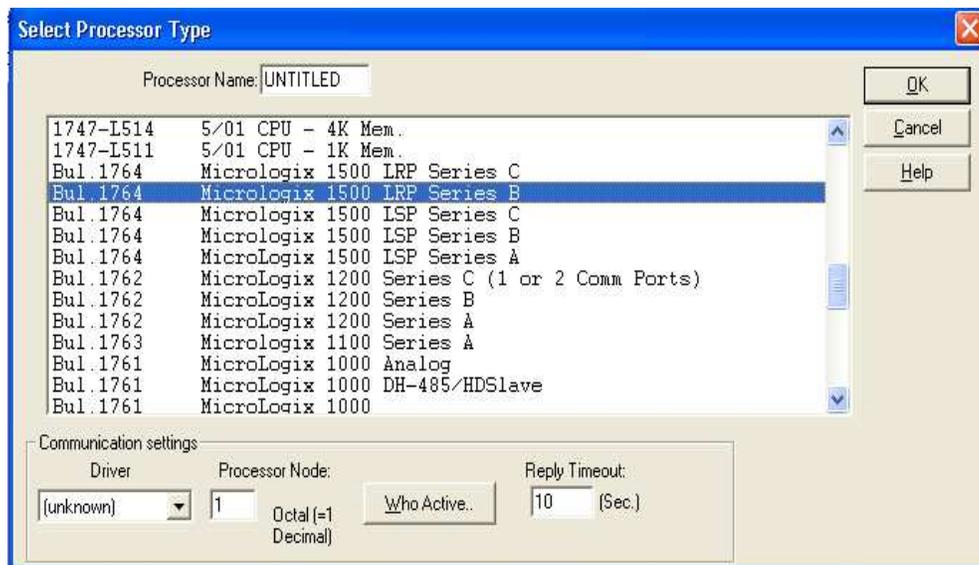


Figura 57. Lista de controladores

Una vez que se seleccionó el PLC, se mostrara una nueva ventana, en esta ventana es en la cual se puede programar al PLC, checar módulos conectados, etc. (Figura 58.)

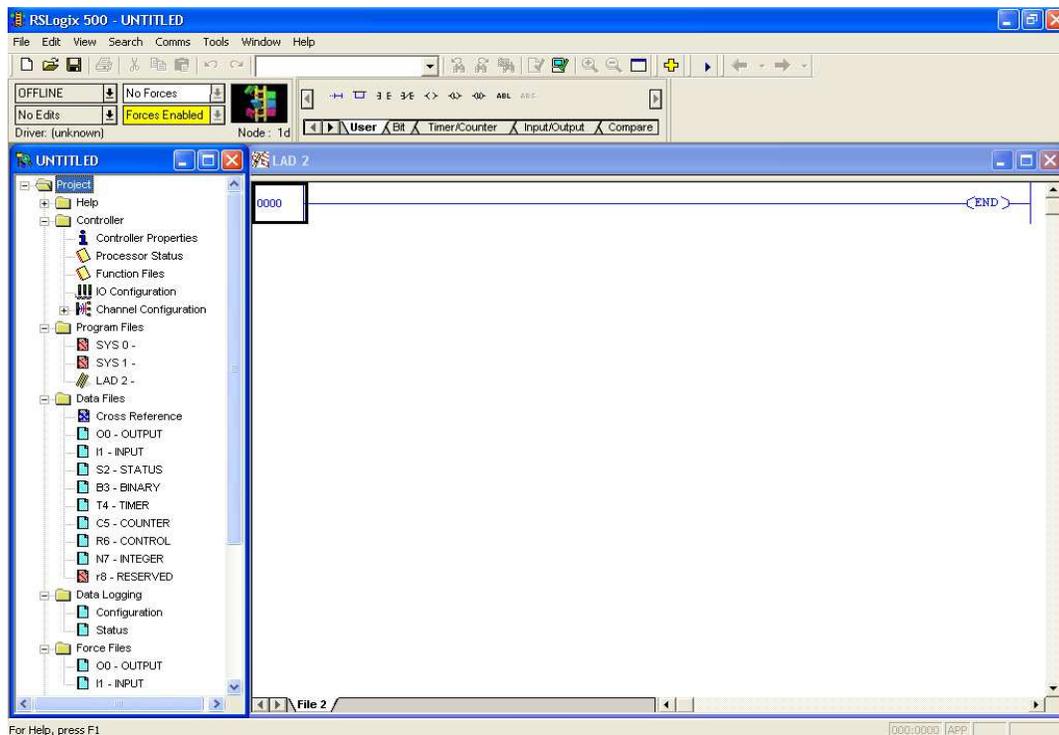


Figura 58. Área de trabajo de RSLogix

De la ventana "UNTITLED" seleccionar la opción "IO Configuration" (Figura 59.), ya que en esta opción es en la cual se le podrá indicar al PLC una instrucción para que reconozca los módulos que tenga conectados físicamente.

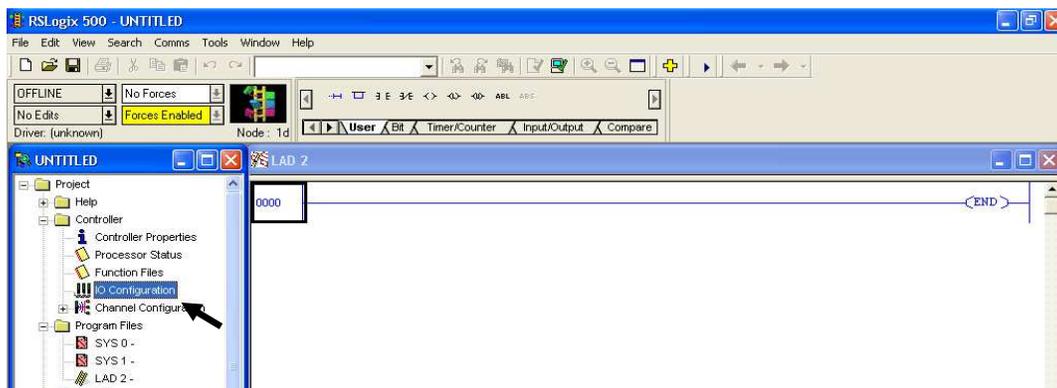


Figura 59. Selección de "IO Configuration"

Al seleccionar esta opción aparecerá otra ventana la cual muestra los dispositivos de E/S (IO) que están conectados a la PC. (Figura 60.)

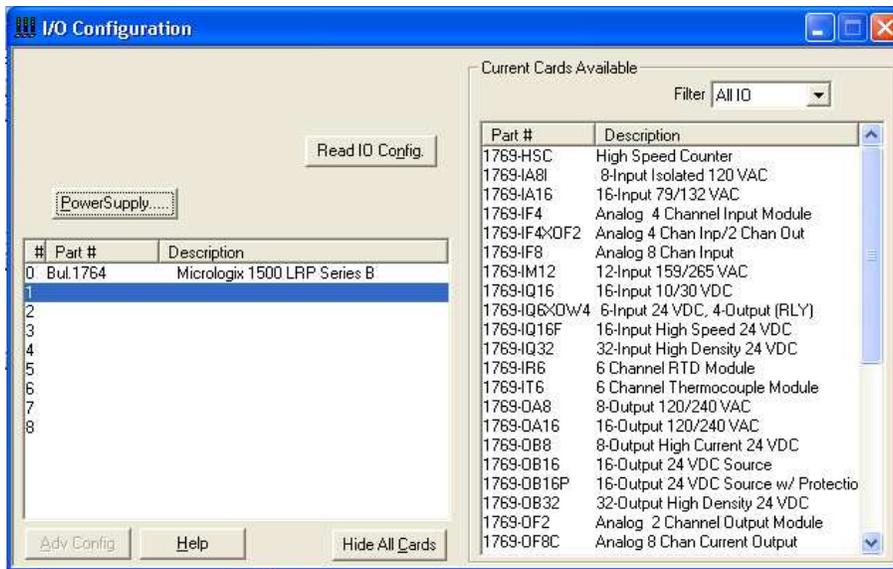


Figura 60. Ventana de configuración de E/S

Como se observa en la parte izquierda de la ventana de la Figura 60. El único dispositivo de E/S que reconoce es el PLC, para aumentar el Scanner como dispositivo de E/S hay presionar la opción "Read IO Configuration" de la ventana que se muestra en la Figura 60, al presionar esa opción aparecerá una nueva ventana (Figura 61.) la cual solicita el "Driver" al cual está conectado tanto el Scanner como el PLC, elegir el controlador que se configuro en la Sección 6.4 y presionar la opción "Read IO Config."

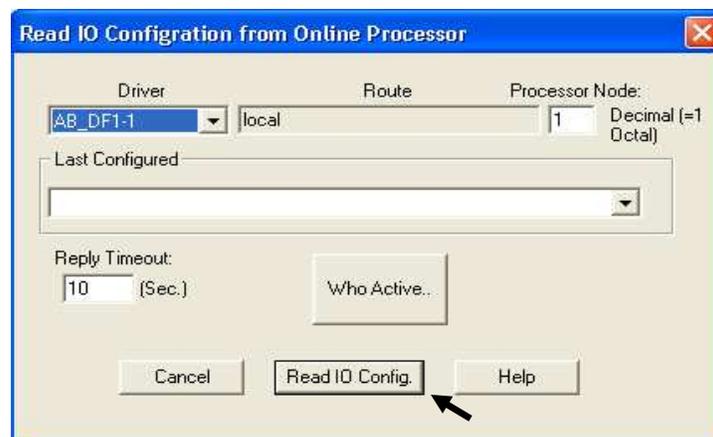


Figura 61. Ventana "Read IO Config"

Después de dar clic en la opción “Read IO Config.” cerrar esa ventana y en automático se mostrara de nuevo la ventana “IO Configuration” la cual ya mostrara en la parte izquierda de la ventana al “Scanner 1769 SDN” como dispositivo de E/S dado de alta (Figura 62.), enseguida cerrar la ventana.

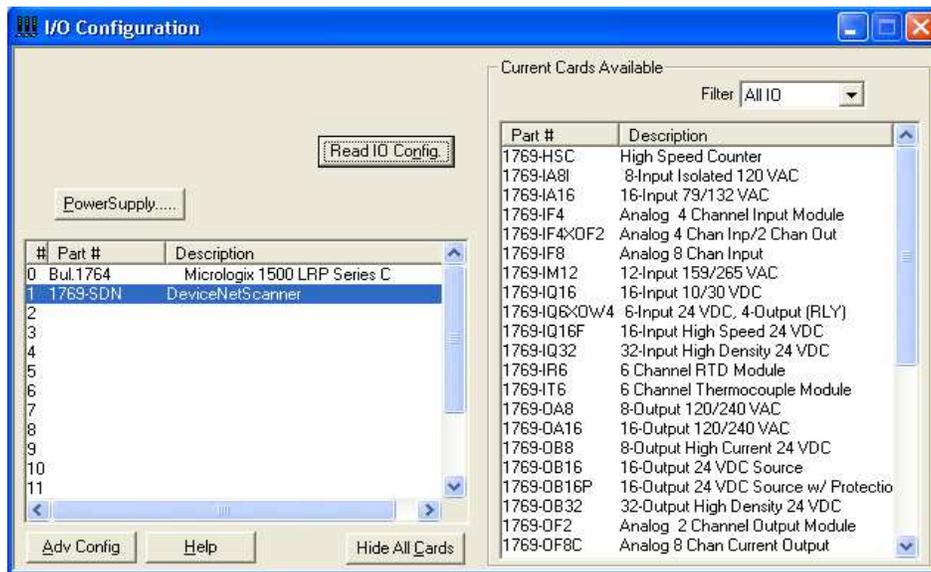


Figura 62. Scanner dado de alta como dispositivo de E/S

Se mostrara en pantalla de nuevo el área de trabajo del programa, ahora hay que conectarse al PLC, dar clic en la parte superior izquierda y seleccionar la opción “Download”. (Figura 63.)

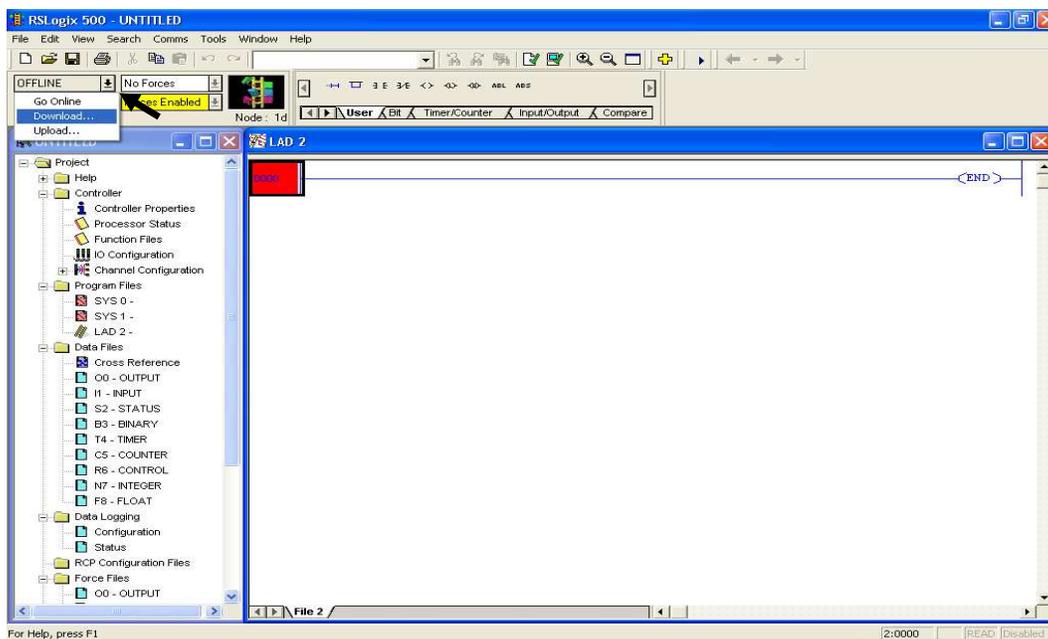


Figura 63. Selección de la opción "Download"

Al seleccionar la opción “Download” el software pedirá guardar los cambios realizados en el programa, se colocara el nombre “Red” y dar clic en la opción “Guardar”. (Figura 64.)

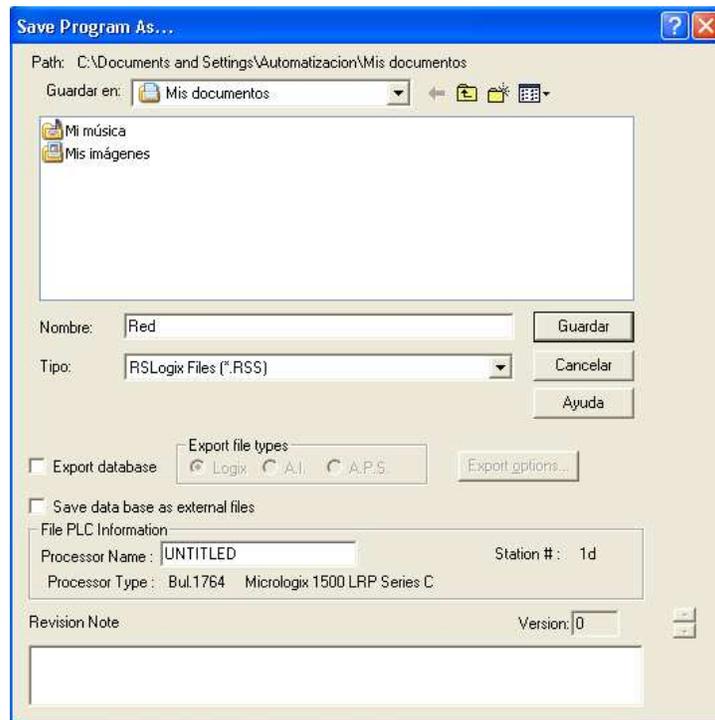


Figura 64. Ventana de "Save Program As.."

Después de guardar el programa se mostrara una nueva ventana la cual preguntara si se desea descargar el programa al PLC que se encuentra conectado al driver configurado en RSLinx, dar clic en “Sí” (Figura 65.), al dar clic en dicha opción se mostrara una ventana temporal en la cual se mostrara el avance de la descarga del programa al PLC. (Figura 66.)

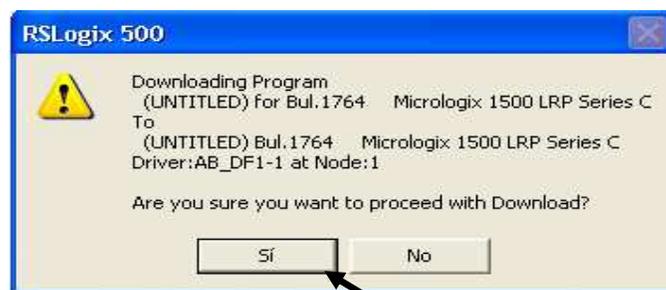


Figura 65. Ventana de descarga al driver configurado en RSLinx

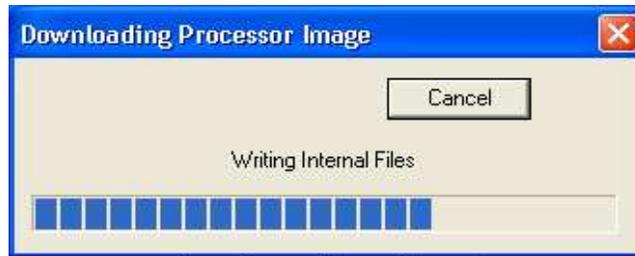


Figura 66. Ventana del avance de la descarga del programa en el PLC

Una vez que la barra de proceso de descarga se llena aparecerá en pantalla una ventana que pregunta si se desea estar en línea con el PLC, dar clic en “Sí”. (Figura 67.)



Figura 67. Ventana "Estar en línea o no con el PLC"

Al estar en línea con el PLC en la parte superior izquierda aparecerá en color verde fosforescente la opción “REMOTE RUN” (Figura 68.) lo cual indica que la PC ya esta comunicándose con el PLC, por lo cual ya se podrá activar el scanner de la red.

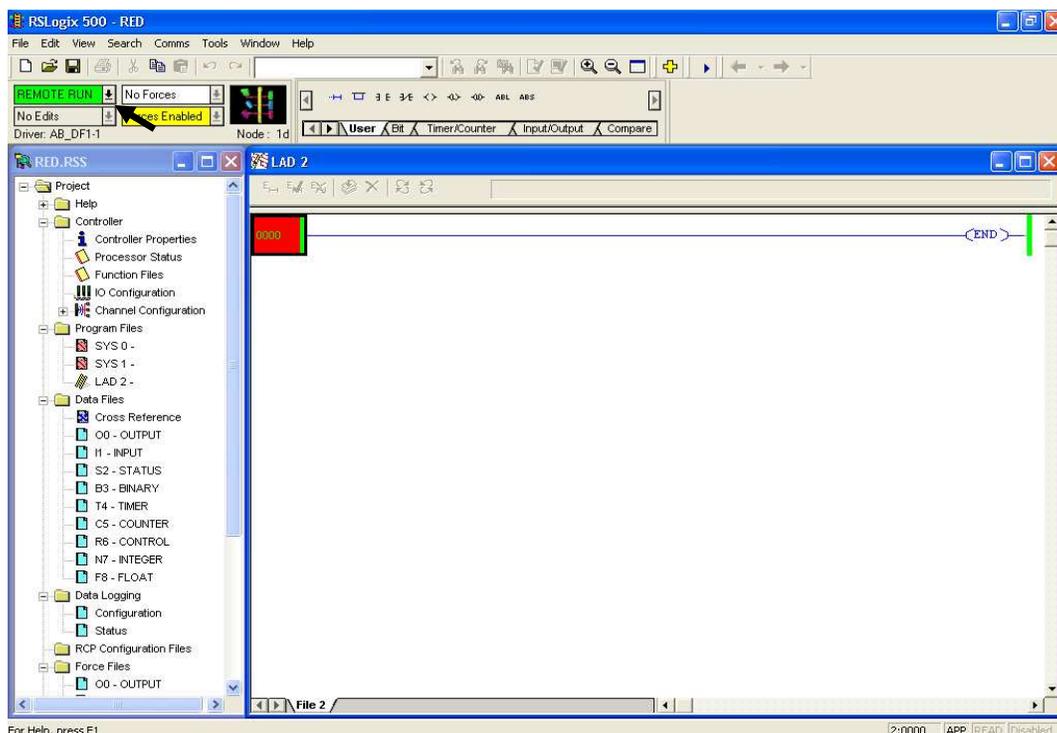


Figura 68. Ventana de comunicación entre el PLC y la PC

Ahora se procede a encender el scanner, en la lista “RED.RSS” que aparece en la parte izquierda de la ventana, dar clic en “Data Files” y después dar doble clic sobre la opción “Input”, esto desplegara una ventana en la cual se muestran las salidas (output) tanto del PLC como del Scanner, para encender el scanner solo se debe de colocar un “1” en el primer bit de la palabra O: 1.0. (Figura 69.)

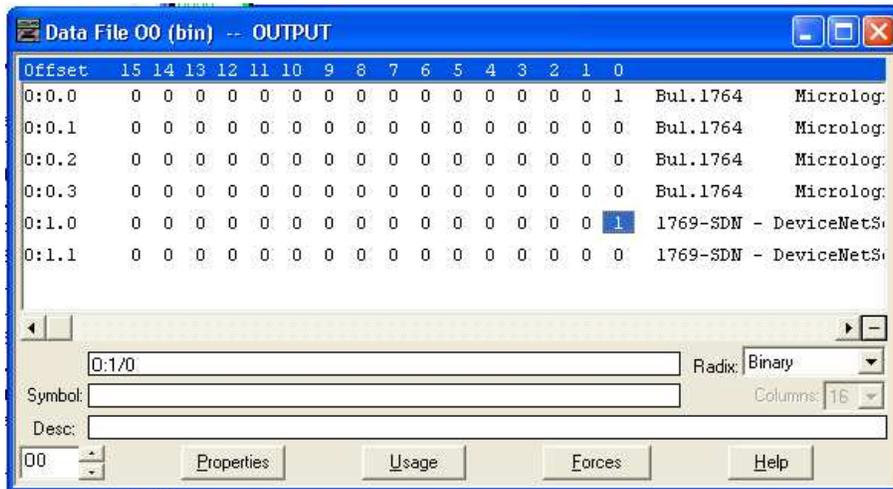


Figura 69. Ventana "Output"

Al realizar lo anterior el scanner quedara encendido y por tanto la red ya no tendrá ningún error, proceder a cerrar la ventana “Output” de la Figura 69, enseguida dar clic en la opción “Go offline” la cual se encuentra en la parte superior izquierda (Figura 70.), aparecerá una ventana la cual preguntara si se desea guardar los cambios, dar clic en la opción “Sí”. Enseguida ya se puede cerrar el programa RSLogix 500.

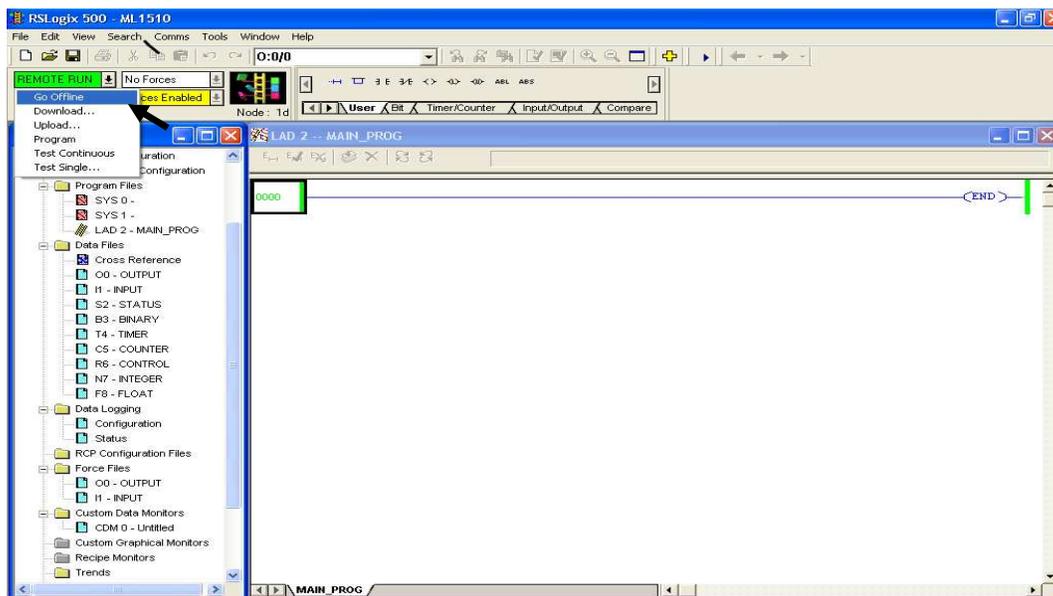


Figura 70. Opción "Go offline"

9.7 Monitoreo de la red utilizando el software LabVIEW

LabVIEW es el software que permite realizar el monitoreo del estado de los dispositivos que conforman la red. Para iniciar dar doble clic sobre el icono del programa (Figura 71).



Figura 71. Icono del programa LabVIEW

Aparecerá la ventana de opciones del programa (Figura 72). Para poder crear el proyecto se debe dar un clic sobre la opción “Blank VI”.

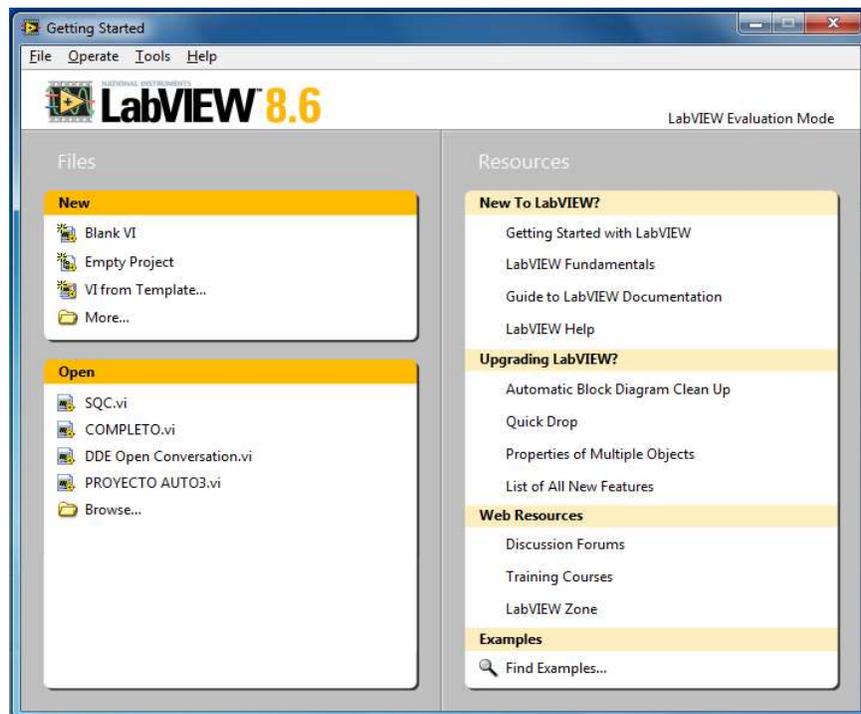


Figura 72. Ventana principal del software

Aparecerán dos ventanas en las cuales se desarrollara el proyecto la primer ventana “Front Panel” (Figura 73) es en la cual se colocara la visualización de los componentes de la red, la segunda ventana “Block Diagram” (Figura 74) sirve para hacer las conexiones virtuales entre los dispositivos de la red y el programa, es importante tomar en cuenta que cada modificación que se realice en la ventana “Front Panel” se reflejara de forma automática en la ventana “Block Diagram”.

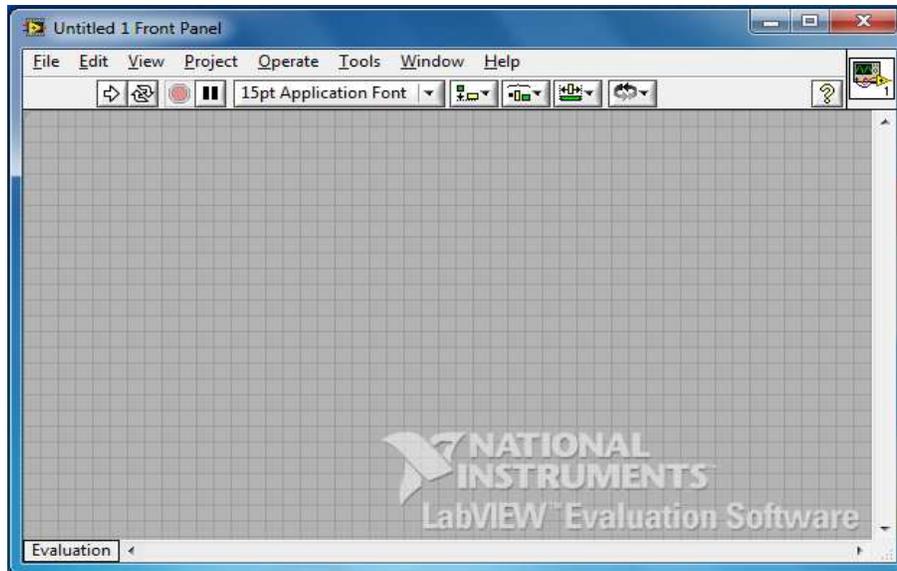


Figura 73. Ventana "Front Panel"

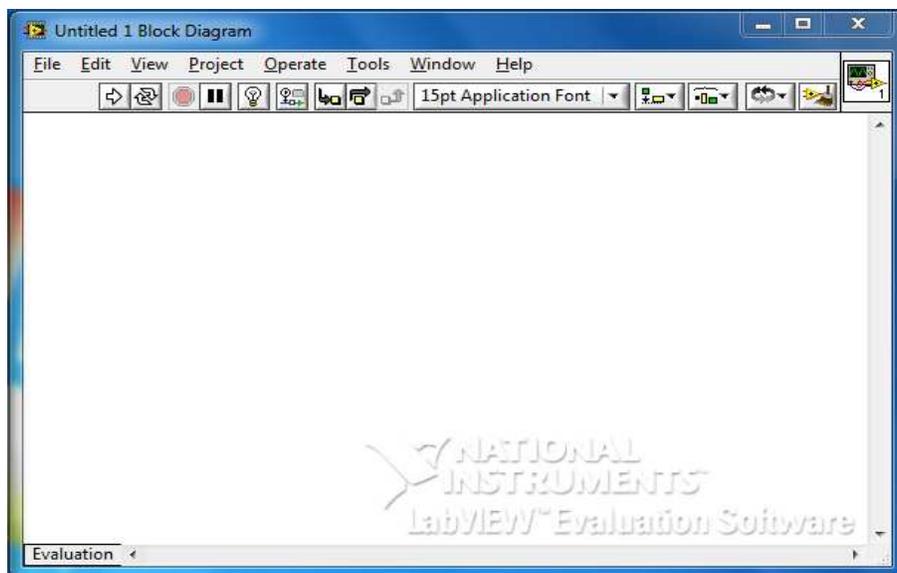


Figura 74. Ventana "Block Diagram"

En la ventana “Front Panel” en cualquier parte de la zona gris de la ventana dar doble clic del botón izquierdo del mouse para poder escribir algún texto sobre esta ventana, en este caso colocar “MONITOREO DE LA RED DEVICENET” (Figura 75).

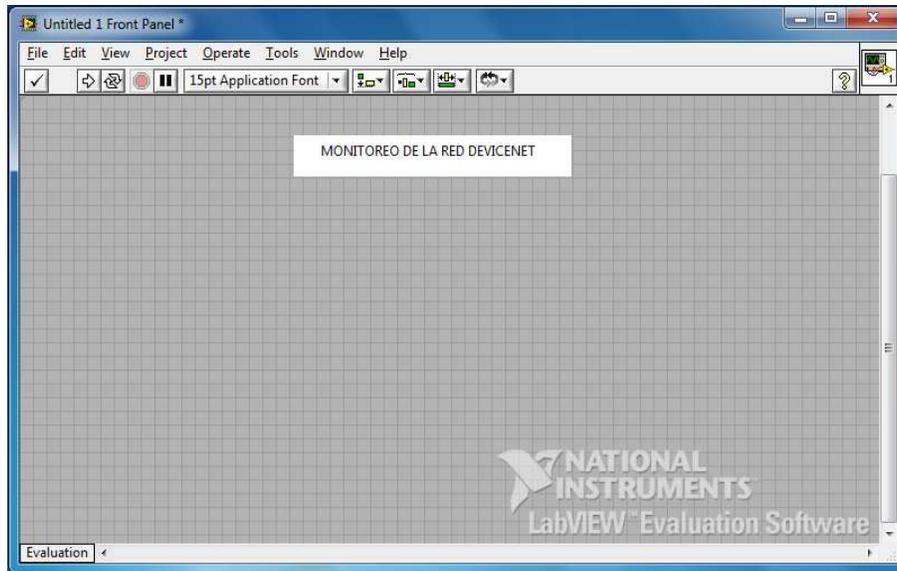


Figura 75. Colocación de texto en la ventana "Front Panel"

Si se desea colocar características como color del texto o color del fondo del texto en la barra de herramientas del programa dar clic sobre la opción "View" lo cual desplegará una lista, seleccionar "Tools Palette" lo cual arrojará una pequeña ventana (Figura 76) la cual incluye opciones que nos permiten mejorar la apariencia del texto colocado anteriormente (Figura 77).



Figura 76. Ventana de "Tools"

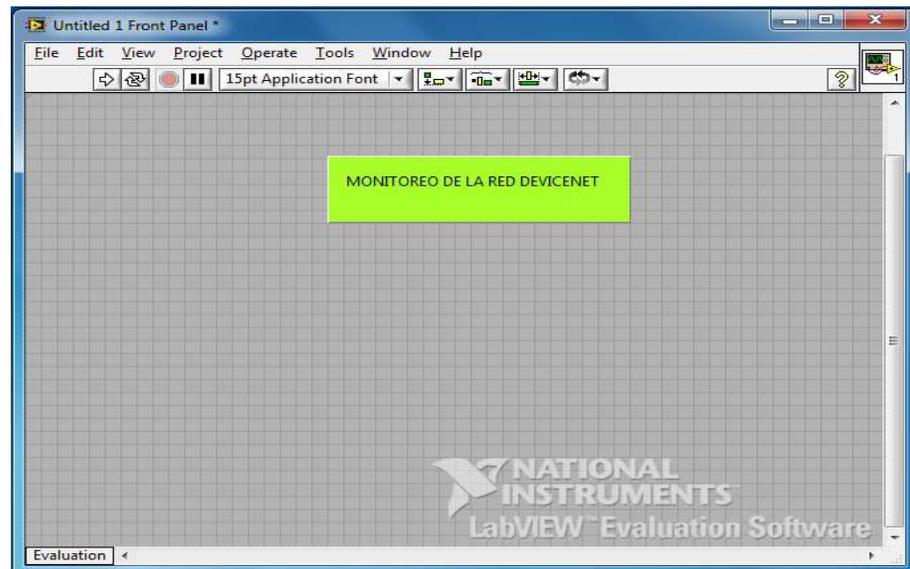


Figura 77. Colocación del texto con fondo de color

Posicionarse en la ventana “Front Panel” y dar un clic con el botón derecho del mouse lo cual desplegara la ventana “Controls” (Figura 78) la cual contiene opciones de botones, leds, etiquetas, gráficos, indicadores, etc.

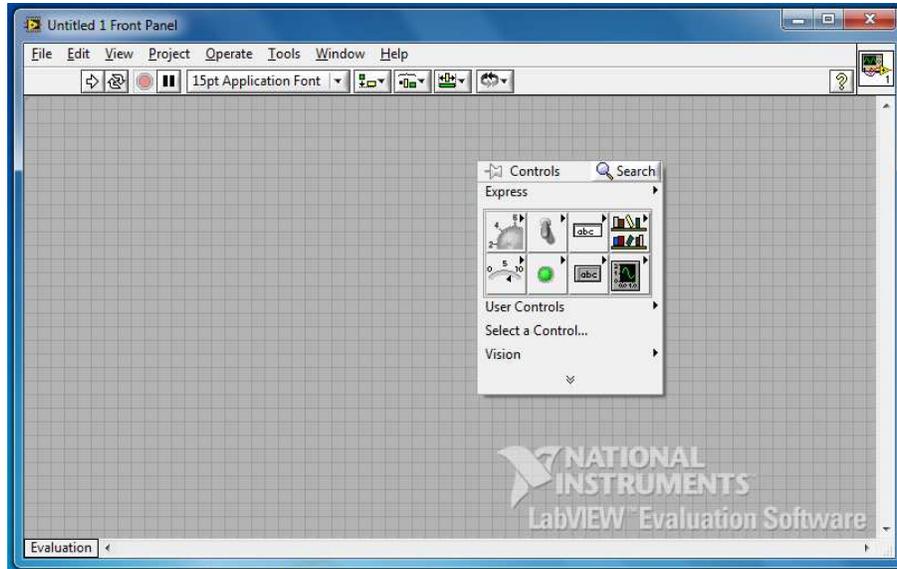


Figura 78. Ventana del menú "Controls"

En el proyecto se estará monitoreando el estado de los 5 sensores que conforman la red así que esto se realizara utilizando leds. Para colocar los leds en la ventana “Controls” dar clic sobre la opción “LEDs – Square LED” y arrastrarlo hasta la zona gris de la ventana “Front Panel” (Figura 79).

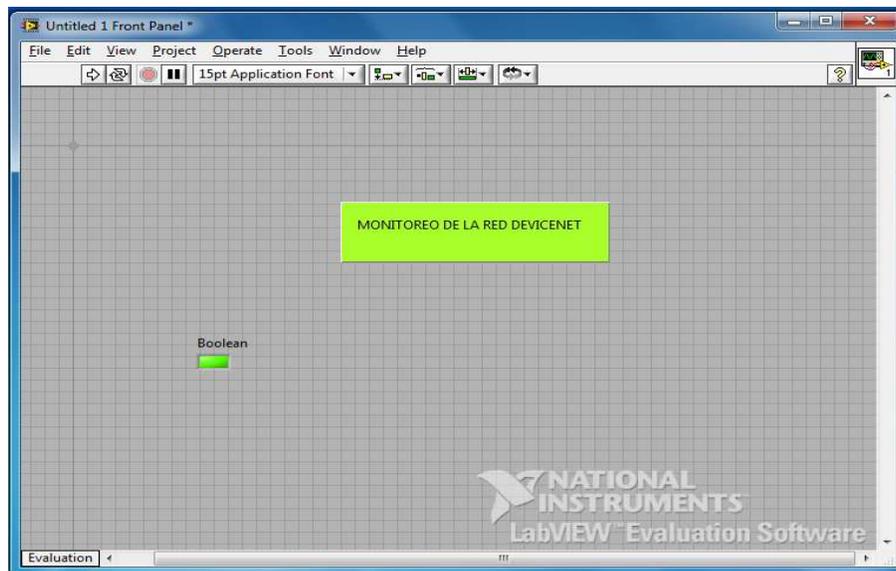


Figura 79. Colocación de un LED

Debido a que la red está conformada por 5 sensores entonces la operación anterior debe repetirse 4 veces más, es decir se deben de colocar 5 LEDs en la ventana “Front Panel” (Figura 80).

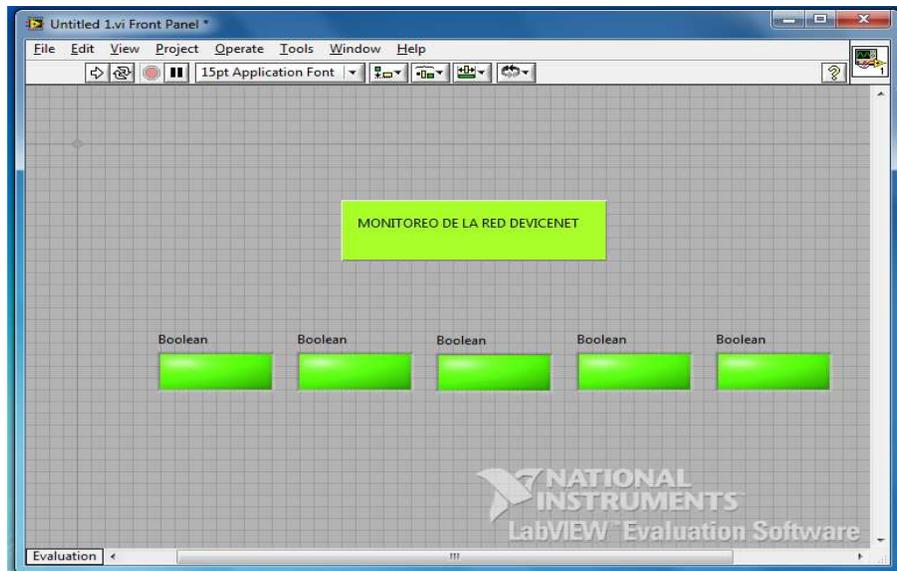


Figura 80. Ventana "Front Panel" con los 5 LEDs

Como se puede ver en la Figura 80 los LEDs que fueron colocados tienen el nombre “Boolean” para poder facilitar el monitoreo de los elementos de la red se le debe cambiar dicho nombre, dar doble clic sobre el nombre actual de los diferentes LEDs y colocar “Sensor 1”, “Sensor 2”, “Sensor 3”, “Limit Switch 1” y “Limit Switch 2” respectivamente. (Figura 81)

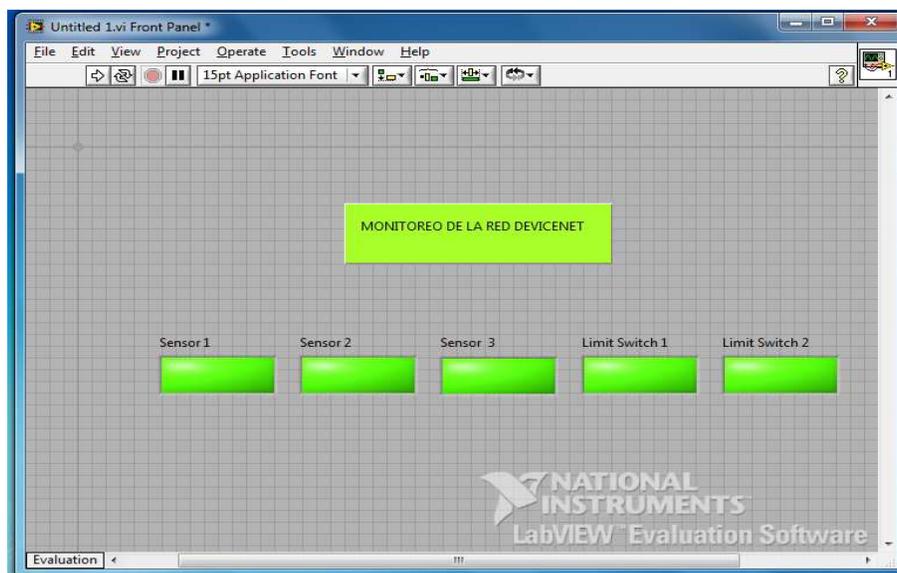


Figura 81. Colocación del nombre de la variable

Ahora ir a la ventana “Block Diagram” la cual aparecerá con 5 iconos los cuales representan los sensores que fueron agregados anteriormente en la ventana “Front Panel” (Figura 82).

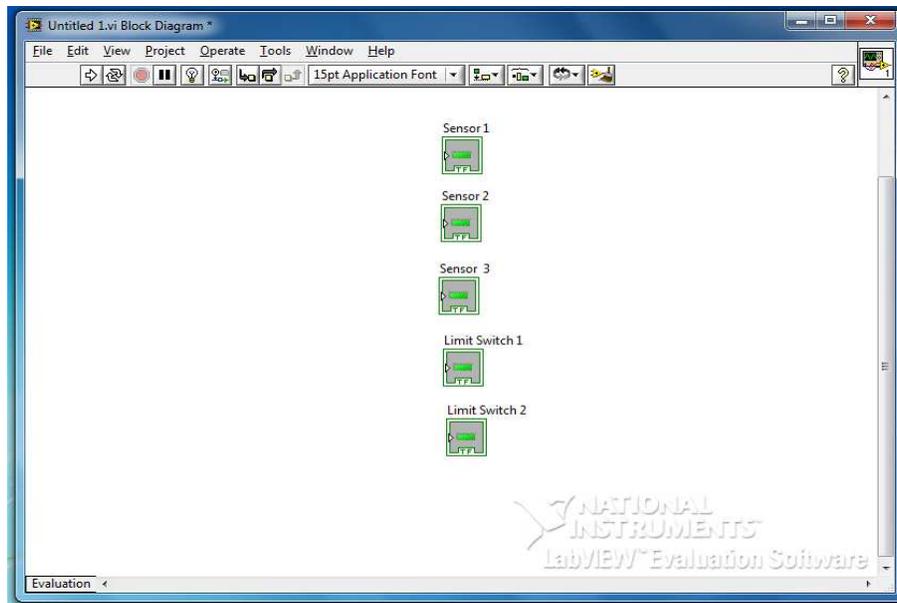


Figura 82. Ventana "Block Diagram" con los iconos de los LEDs

Se debe proceder a configurar las conexiones virtuales necesarias para que el monitoreo de la red pueda funcionar. Para poder incluir las funciones de comunicación que se utilizaran se debe ingresar a la ruta “C:\Archivos de Programa\National Instruments\LabVIEW 8.6\vi.lib\platform” y seleccionar la biblioteca “dde.llb”, aparecerá una pequeña ventana la cual contiene las diferentes funciones de comunicación que contiene el software (Figura 83).

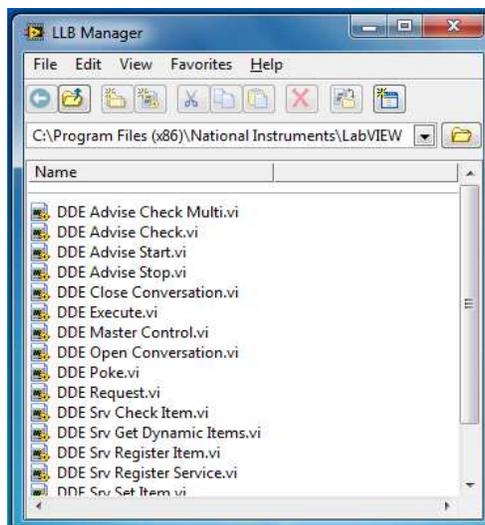


Figura 83. Ventana de las funciones de comunicación

De la ventana de la Figura 83 se utilizarán las funciones “DDE Open Conversation”, “DDE Request” y “DDE Close Conversation”. Para colocar dichas funciones en la ventana “Block Diagram” se debe seguir el siguiente procedimiento:

- Dar doble clic sobre la función que se insertara, aparecerá una nueva ventana (Figura 84) de esta seleccionar con el mouse el icono correspondiente a la función (Figura 85) y sin soltar el botón del mouse arrastrar dicho icono hasta la ventana “Block Diagram” del programa.

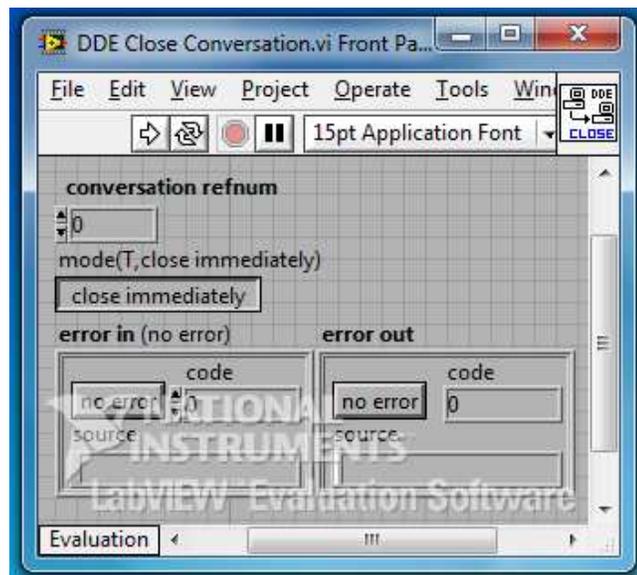


Figura 84. Ejemplo de la ventana de propiedades de una función de comunicación

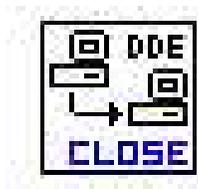


Figura 85. Icono de una función de comunicación

El procedimiento mencionado anteriormente se utilizara para insertar:

- 1 - DDE Open Conversation
- 5 - DDE Request
- 1 - DDE Close conversation

Una vez que se colocaron todas las funciones de comunicación en la ventana “Block Diagram” (Figura 86) se debe proceder a configurar dichas funciones.

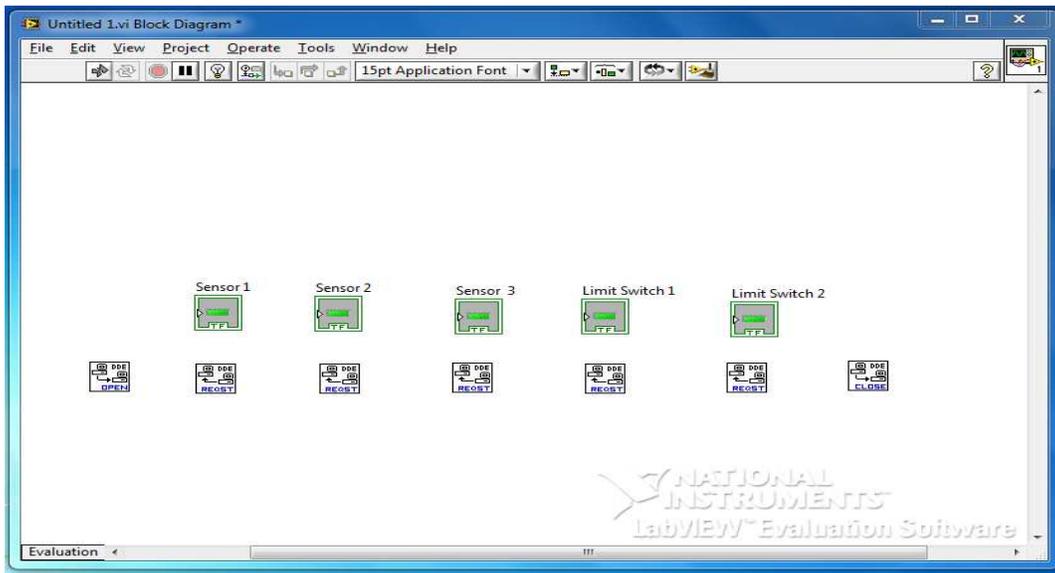


Figura 86. Ventana "Block Diagram" con las funciones de comunicación

Cada una de las funciones contiene las opciones:

- **“error out”**. Se encuentra en la parte inferior del lado derecho del icono de cada función y se utiliza una línea amarilla para hacer la conexión de este parámetro.
- **“conversation refnum in”**. Se encuentra en la parte superior derecha del lado izquierdo del icono de la función “DDE Request” y se utiliza una línea azul para hacer la conexión de este parámetro con la opción “conversation refnum out” de la función de comunicación siguiente.
- **“conversation refnum”**. Se encuentra en la parte superior de lado derecho de la función “DDE Open” la cual debe conectarse con la opción “conversation refnum in” del primer “DDE Request” y en la parte superior izquierda de la función “DDE Close” la cual debe estar enlazada con la opción “conversation refnum out” del quinto “DDE Request” utilizado. Esta opción también se enlaza con una línea color azul.
- **“conversation refnum out”**. Se encuentra en la parte superior derecha de la función “DDE Request”, esta opción debe conectarse con la opción

“conversation refnum” de la función “DDE Close”. (esto solo aplica en el caso de la función “DDE Request” que se encuentre junto a la función “DDE Close”).

Enlazar conforme a lo descrito anteriormente cada una de las distintas funciones “DDE” utilizadas (Figura 87).

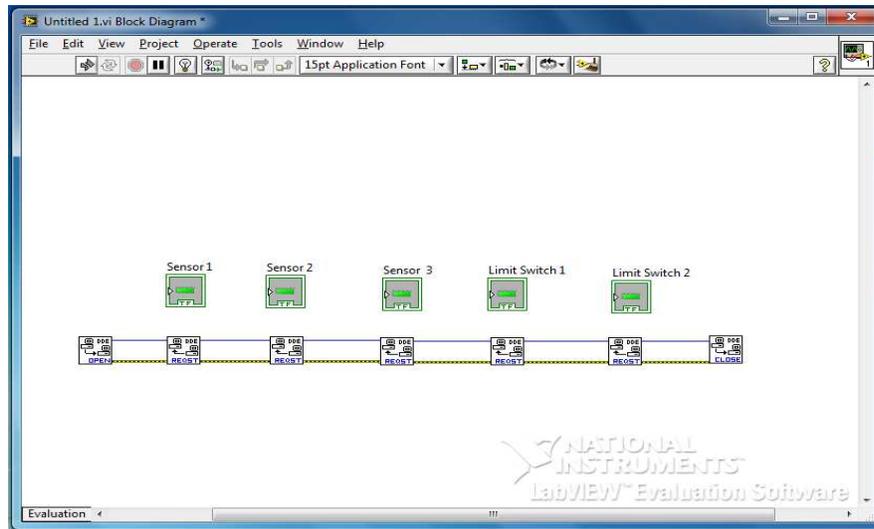


Figura 87. Conexiones virtuales entre las distintas funciones "DDE"

Ahora se realizaran las conexiones necesarias en cada una de las funciones:

➤ **DDE Open**

❖ Opción “Service”

Esta opción se encuentra en la parte superior del lado izquierdo de la función, colocar el puntero del ratón (mouse) en dicha opción y dar un clic en el botón derecho del ratón (mouse), seleccionar “Create” y después “Constant”, aparecerá un pequeño cuadro de color rosa en donde se debe escribir “RSLinx” ya que este es el programa que nos permite hacer la comunicación. (Figura 88)

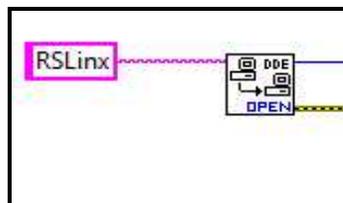


Figura 88. Opción "Service" de la función DDE Open

❖ Opción "Project"

Esta opción se encuentra en la parte superior del lado izquierdo de la función, colocar el puntero del ratón (mouse) en dicha opción y dar un clic en el botón derecho del ratón (mouse), seleccionar "Create" y después "Constant", aparecerá un pequeño cuadro de color rosa en donde se debe escribir el nombre del programa que se creó en el software RSLogix 500 (proceso descrito en la unidad 9), en el caso del presente trabajo escribir "Red". (Figura 89)

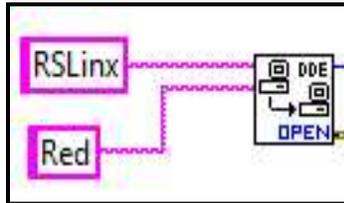


Figura 89. Opción "Project" de la función DDE Open

➤ **DDE Request**

❖ Opción "Item"

Esta opción se encuentra en la parte central del lado izquierdo de la función, colocar el puntero del ratón (mouse) en dicha opción y dar un clic en el botón derecho del ratón (mouse), seleccionar "Create" y después "Constant", aparecerá un pequeño cuadro de color rosa en donde se debe de escribir la dirección de la entrada o salidas de la red que se estarán monitoreando, por ejemplo: para el primer sensor tendríamos que colocar "I1:66/0" y así sucesivamente con los demás sensores según la dirección que tengan cada uno de ellos dentro del PLC. (Figura 90)

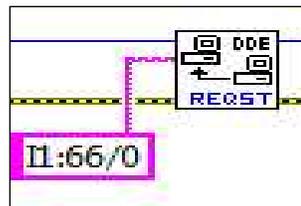


Figura 90. Opción "Item" de la función DDE Request

❖ Opción “Data”

Esta opción se localiza en la parte central del lado derecho de la función, esta opción requiere de otros elementos para poder funcionar, los elementos que se requieren son:

1. *Decimal String to Number*

En cualquier parte de la ventana dar un clic derecho, se desplegará una lista de opciones en la cual se debe seleccionar “Programming”, enseguida “String”, de la ventana que aparecerá seleccionar “String/ Number Conversion” y enseguida seleccionar “Decimal String to number”, aparecerá un pequeño icono el cual debe colocarse cerca de la función request.

Ahora conectar la opción “String” de dicha herramienta con la opción “Data” de la función “Request”. (Figura 91)

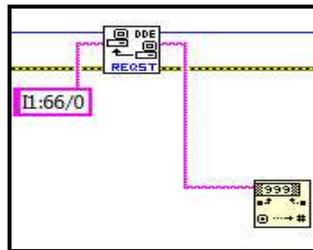


Figura 91. Herramienta "Decimal String to Number"

2. *Number to Boolean Array*

En cualquier parte de la ventana dar un clic derecho, se desplegará una lista de opciones en la cual se debe seleccionar “Programming”, en la ventana que aparecerá seleccionar “Boolean” lo cual desplegará una nueva ventana en la cual se debe dar clic sobre la opción “Number to Boolean Array”, esto colocará un nuevo icono en la ventana, dicho icono se debe colocar cerca de la función “Request”. Ahora se debe conectar la opción “Number” de esta herramienta con la opción “Number” de la herramienta “Decimal String to Number”. (Figura 92)

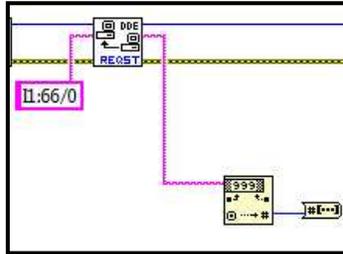


Figura 92. Herramienta "Number to Boolean Array"

3. Index Array

En cualquier parte de la ventana dar un clic derecho, se desplegará una lista de opciones en la cual se debe seleccionar "Programming", en la ventana que aparecerá ahora se debe de seleccionar "Array", en la nueva ventana que se arroja se debe seleccionar "Index Array", esto colocará un nuevo icono en la ventana, dicho icono se debe colocar cerca de la función "Request". Enseguida se debe conectar la opción "Array" de esta herramienta con la opción "Boolean Array" de la herramienta "Number to Boolean Array".

En la opción "Index" de esta herramienta, dar clic derecho en el ratón (Mouse), seleccionar "Numeric Palette" y de la ventana que aparecerá dar clic en la opción "Numeric Constant" y colocar un "0" en el recuadro azul que aparecerá.

Ahora se debe conectar la opción "Element" de esta herramienta con el icono del sensor que será monitoreado. (Figura 93)

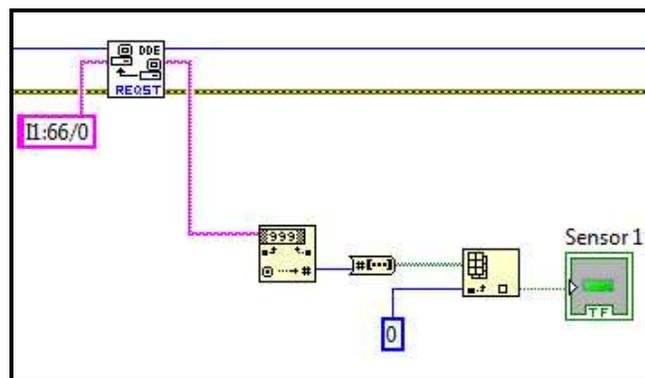


Figura 93. Herramienta "Index Array"

Nota: Los pasos mencionados anteriormente se deben de repetir 5 veces ya que son 5 los elementos que se están monitoreando, cuidando de colocar la dirección correcta de cada sensor es decir el número de "Input" configurado anteriormente en el software "RS NetWorx" para cada uno de los sensores.

9.7.1 Configuración de RSLinx para poder establecer la comunicación entre la red y el PLC

RSLinx es el software que nos permite establecer una comunicación tipo “DDE” entre la PC y el PLC. Sin cerrar software “LabView” lo primero que debemos hacer es ingresar al software RSLinx el cual debe estar habilitado con el “Driver” configurado para el PLC (visto anteriormente) para ingresar a este dar un clic sobre el icono del software el cual se debe encontrar en la PC. (Figura 94).

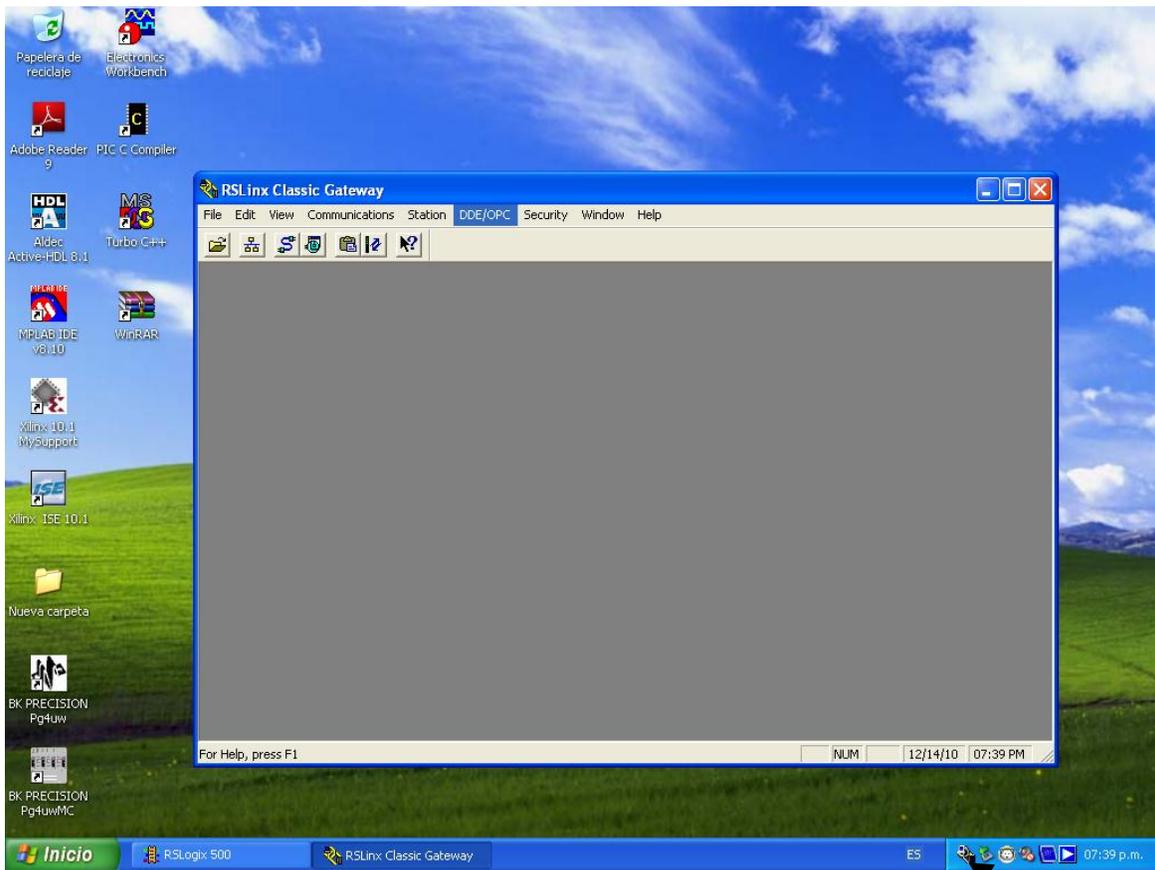


Figura 94. Icono de RSLinx

Se mostrara en pantalla la ventana principal del software en este caso con un clic elegiremos la opción “DDE/OPC”, se desplegara una pequeña ventana con una lista de opciones se debe seleccionar “Topic Configuration”. (Figura 95)

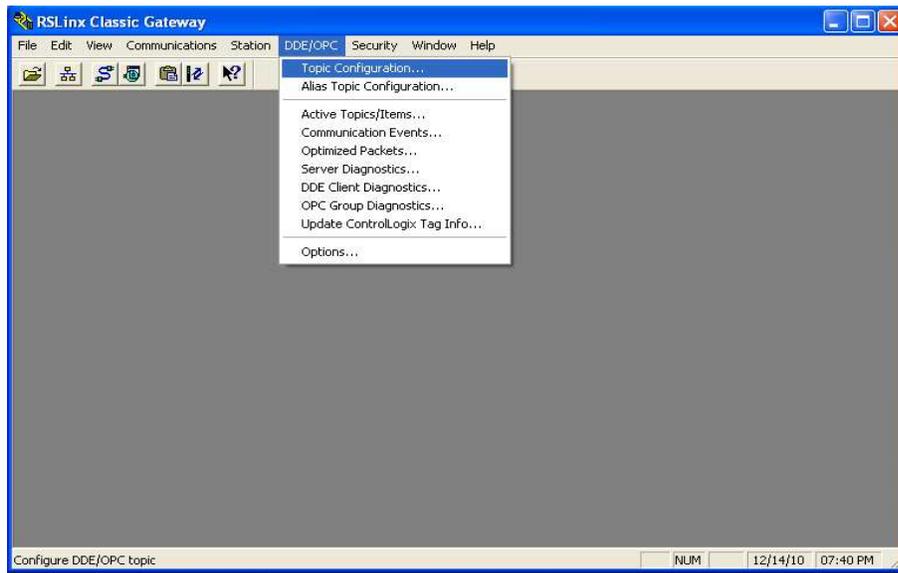


Figura 95. Opción "DDE/OPC - Topic Configuration"

El software nos mostrara una nueva ventana (Figura 96) en la cual se puede configurar el driver que comunica al PLC con la PC (driver AB_DF1-1) para que este funcione correctamente en el intercambio dinámico de datos (“DDE”) que se utilizará.

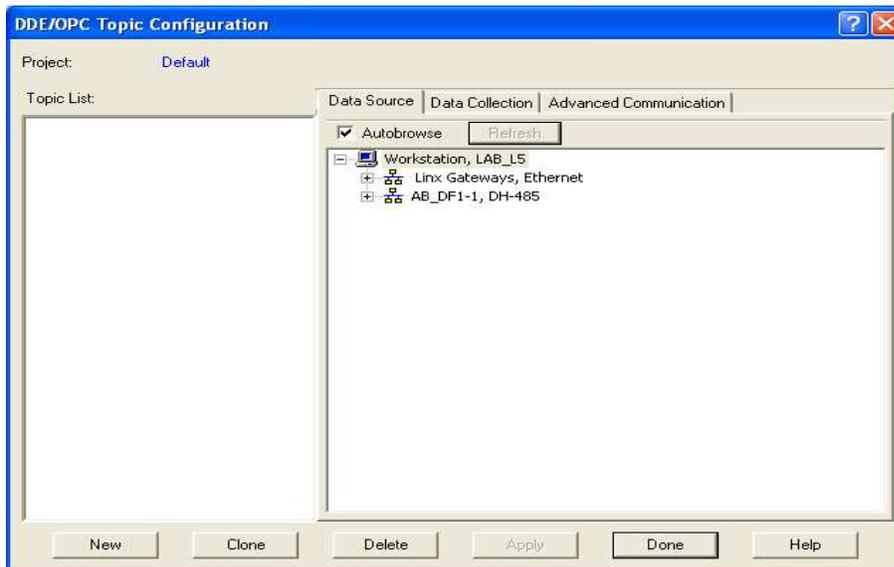


Figura 96. Ventana de la opción "Topic Configuration"

Enseguida se debe agregar un “Topic List”, dar clic en la opción “New” de la ventana mostrada anteriormente, en la parte izquierda de la ventana aparecerá el nombre “New

Topic”, dar doble clic sobre este y colocar el nombre del programa que se creó anteriormente en el software “RS Logix 500” en este caso colocar “Red” (Figura 97), presionar la tecla “Enter” y en la pequeña ventana que aparecerá dar clic sobre la opción “sí” (Figura 98) y presionar la opción “Apply”.

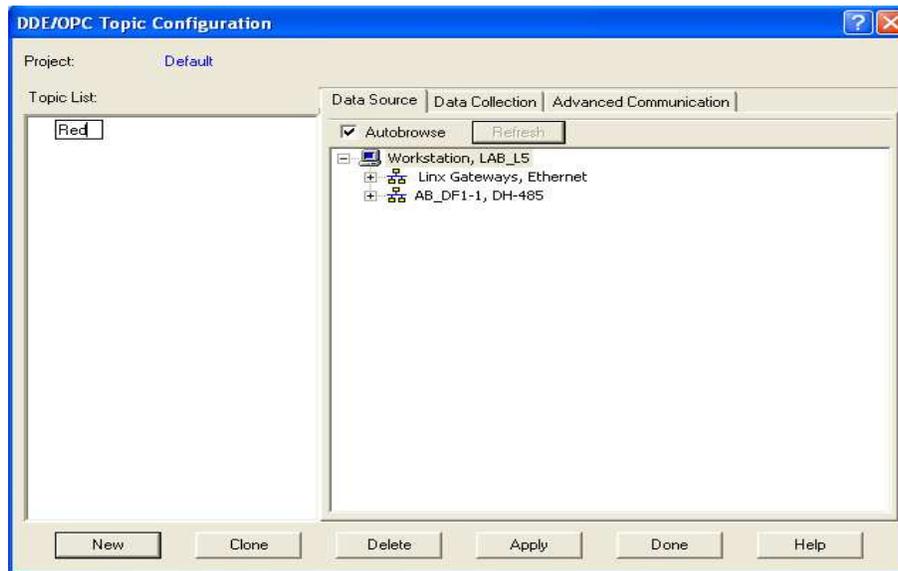


Figura 98. Configuración del DDE/OPC Topic

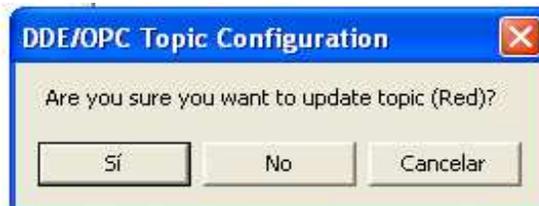


Figura 97. Opción para cargar el nuevo Topic en el PLC

Para continuar, dar clic sobre la opción “Advanced Communication” que se encuentra en la parte superior derecha de la ventana de la figura 97, se mostrara una ventana (Figura 99) en la cual se debe dar clic sobre la opción “Communications Driver” y seleccionar el driver creado en capítulos anteriores para la comunicación del PLC con la PC, en este caso seria “AB_DF1-1”, enseguida presionar la opción “Done” y de esta forma ya está lista la configuración para el intercambio dinámico de datos que se utilizara.

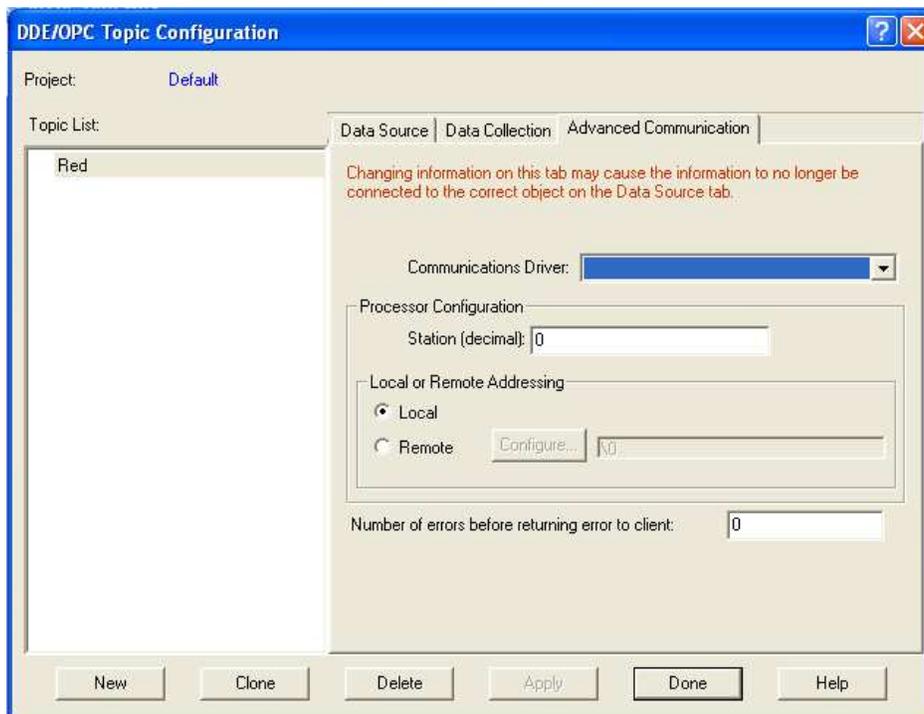


Figura 99. Selección del driver a utilizar para el intercambio de datos

9.7.2 Inicio del monitoreo de la red

Ahora se debe regresar a la ventana del software “LabView” y se debe dar clic en la opción “Run Continuously”  , si todo esta correcto en la pantalla del software se estará mostrando el estado de los distintos sensores conectados a la red (estado alto en color verde fosforescente y estado bajo en color verde fuerte) como se muestra en la Figura 100.

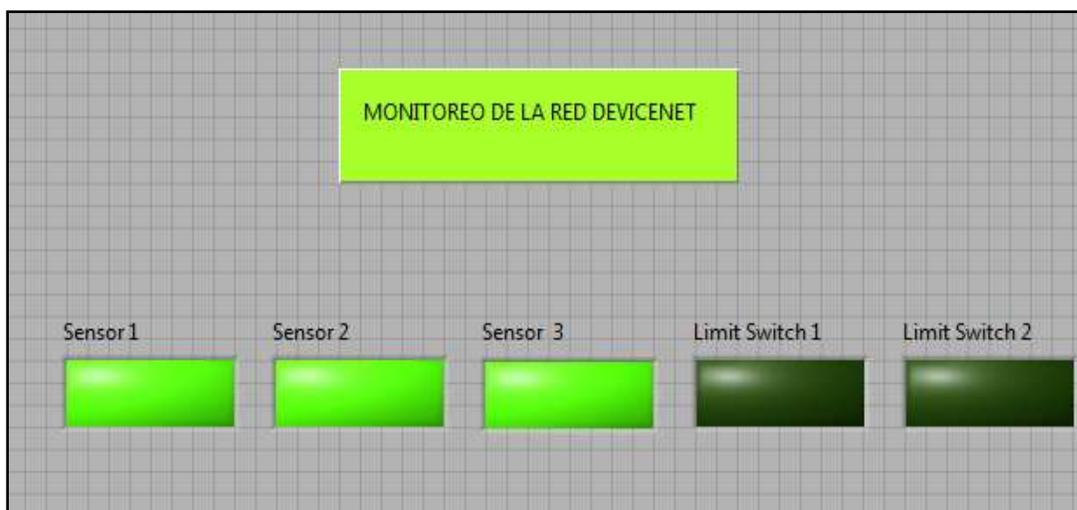


Figura 100. Monitoreo de la Red Activado

En caso de que se desee pausar o terminar el monitoreo solo se debe dar clic sobre la opción “Pause”  o “Abort Execution”  .

X. CONCLUSIONES

El uso de la tecnología como medio de transmisión nos permite manipular y controlar de manera más rápida y eficiente el control de un proceso, para ser más precisos en el caso de nuestro trabajo pudimos observar que los buses de campo nos ayudan a reducir costos y cantidades de material utilizado (por ejemplo cables) en cierto tipo de instalaciones industriales en lo que respecta a su sistema de control.

El trabajo resulto ser todo un éxito ya que si se lograron alcanzar los objetivos que se habían planteado al inicio del mismo, así mismo se pudo aprender más sobre el concepto y aplicación de uno de los buses de campo de mayor uso en el mundo del control automático.

El mundo del control automático cada día gana más territorio en el entorno industrial por lo cual es muy importante el adaptar cada uno de los procesos que aún se hacen de manera manual para así poder reducir costos y tiempos de operación, esto no solo beneficiara a las industrias sino también a los operadores ya que su trabajo lo podrán realizar de una manera más sencilla y segura.

XI. BIBLIOGRAFIA

- 1) <http://sistemas-uvg.blogcindario.com/2010/05/00003-modelo-osi.html>
- 2) <http://www.scribd.com/doc/13980265/CAPITULO-7-Capa-de-Enlace-de-Datos>
- 3) <http://www.emb.cl/electroindustria/articulo.mv?xid=117&rank=1>
- 4) <http://www.ab.com/en/epub/catalogs/12762/2181376/214372/1768364/3404052/tab3.html>
- 5) <http://www.isa.uniovi.es/docencia/redes/Apuntes/tema9.pdf>
- 6) <http://www.instrumentacionycontrol.net/es/curso-supervision-procesos-por-computadora/271-los-buses-de-campo-directo-al-grano.html>
- 7) <http://www.scribd.com/doc/16280900/Buses-de-Campo>
- 8) DeviceNet Specification, Volume I, Release 1.3, Open
- 9) DeviceNet Vendor Association Inc., December 1995
- 10) Tindell K, "Calculating CAN response times", University of York, England 1995.
- 11) Mc Farlane Andy, "Tutorial: Fieldbus review", Sensor Review, Vol 17, Num 3, 1997.
- 12) Sirgo J.A., "Redes locales en entornos industriales: Buses de campo", Universidad de Oviedo, 1997.
- 13) Quezada J., "Bus CAN: Estado de buses industriales y aplicaciones" Escuela Técnica Superior de Ingenieros Industriales", 1999.
- 14) www.fieldbus.org, Fieldbus Organization.
- 15) www.bosh.de/KB/can, Página de la Bosh dedicada a CAN.
- 16) www.Kvaser.se. .KingDom CAN applications.