

# IMPLEMENTACIÓN DE UN SERVIDOR OPC PARA EL CONTROL DE ARRANQUE Y PARO DE UN MOTOR TRIFÁSICO

*IMPLEMENTATION OF AN OPC SERVER TO CONTROL THE STARTING AND STOPPING OF A THREE-PHASE MOTOR*

**Julio César Martínez Gámez**

Universidad Tecnológica de Altamira  
*jmartinez@utaltamira.edu.mx*

**Laura Tayde Contreras Álvarez**

Tecnológico Nacional de México/Instituto Tecnológico de Ciudad Madero  
*ltaide@prodigy.net.mx*

**Javier Rossette García**

Universidad Tecnológica de Altamira  
*jrossete@utaltamira.edu.mx*

## Resumen

El OPC (OLE para Control de Procesos) es un estándar de comunicación en el campo del control y supervisión de procesos industriales, basado en una tecnología Microsoft, que ofrece una interfaz común para comunicación que permite que componentes de software individuales interactúen y compartan datos. Este artículo trata sobre la tecnología OPC, como una sencilla herramienta al estándar abierto de conectividad industrial más popular. En el Laboratorio Pesado de Mecatrónica de la UT de Altamira, se desarrolló un módulo OPC con el objetivo de realizar una interfaz gráfica que permita la comunicación e interoperabilidad entre diferentes fabricantes de software y hardware para el control de arranque de un motor trifásico. El módulo OPC puede ser utilizado en aplicaciones de automatización de control de procesos e instrumentación virtual.

**Palabras clave:** Automatización, control, tecnología OPC.

## Abstract

*The OPC (OLE for Process Control) is a communication standard in the field of control and supervision of industrial processes, based in a Microsoft technology,*

*that offers a common interface for communication which enables components of individual software to interact and share data. This article is about the OPC technology, as a simple tool to the open standard of the most popular industrial connectivity. In the Heavy Mechatronics Laboratory of the UT of Altamira, an OPC module was developed with the aim of creating a graphical interface that allows communication and interoperability between different software and hardware manufacturers for starting control of a triphase engine. The OPC module can be used in applications of automation of process control and virtual instrumentation.*

**Keywords:** *Automation, control, OPC technology*

## **1. Introducción**

Una de las actividades más importantes en las industrias modernas es el monitoreo y control de las variables asociadas a sus procesos de producción. En vista que las computadoras son herramientas capaces de almacenar, procesar y presentar información en forma atractiva y confiable, la tendencia en las industrias modernas es asociar sus procesos automatizados a programas que posean un ambiente en el cual el usuario pueda tener acceso para monitorear y modificar los distintos elementos que conforman su sistema de control (Wang L. y Tan K. C, 2006).

Entre las dificultades de las operaciones de control industrial, resalta la de compartir información tanto entre los distintos dispositivos de campo como con el resto de la empresa o instalación industrial, ya que, es muy posible que los dispositivos transmitan sus datos usando distintos protocolos, siendo generalmente estos incompatibles entre sí (Distefano M, 1999). Esta situación limita la operatividad entre equipos y aplicaciones de diferentes fabricantes, obligando a la utilización de medios de interfaz entre los distintos sistemas; esto, además de propiciar el encarecimiento de los mismos, crea una marcada dependencia, tanto con un determinado fabricante como con la tecnología utilizada en la planta o instalación (OPC FOUNDATION, 1998). Esta diversificación ha obligado a los desarrolladores de Sistemas de Control y Adquisición de Datos (SCADA) a incorporar diferentes tipos de controladores,

buscando incluir a diversos fabricantes (Distefano M, 1999). Todo esto ha evidenciado la necesidad de crear una norma de intercambio de datos a nivel de planta, desencadenando el surgimiento de la tecnología OLE (Object Linking and Embedding/Enlace e Inserción de Objetos), denominada OPC (OLE for Process Control / OLE para Control de Procesos).

El desarrollo de productos basados en el estándar OPC, así como en las metodologías actuales de diseño de software, garantizarán en un futuro cercano una alta compatibilidad entre plataformas tecnológicas en el área de la automatización industrial (Shimanuki Y. 1999), permitiendo a su vez, una independencia y generalización que liberará a las industrias de utilizar sólo la plataforma de programación propia del fabricante y supondrá un abaratamiento en la inversión, debido a que ahora, el usuario tendrá mayores alternativas para escoger los equipos con la tecnología que considere más conveniente a su proceso.

Como antecedente previo no se cuenta en la Universidad con ningún dispositivo OPC, es la primera vez que se desarrolla en la UT de Altamira este tipo de proyecto. Y como primer desarrollo, se realizará la comunicación con PLC'S MicroLogix 1500 de Allen-Bradley y Vision 130 de UNITRONICS y el siguiente software: RS Classic, RS Logix 500 pro y PanelBuilder32 de desarrollo de aplicaciones para terminales PanelView estándar; para el control (paro y arranque) de un motor trifásico. Este tipo de servidor OPC ayudará a los estudiantes de la carrera de TSU e Ingeniería en Mecatrónica a familiarizarse con la Tecnología OPC en la realización de sus proyectos de materias como Automatización y Control Industrial, Controladores Lógicos Programables (PLC'S), Instrumentación Virtual, Programación Visual.

### **¿Qué es OPC?**

OPC es el método de conectividad de datos basado en los estándares más populares del mundo. OPC no es un protocolo, sino más bien un estándar para la conectividad de datos que se basa en una serie de especificaciones OPC, gestionadas por la OPC Foundation. Es utilizado para responder a uno de los

mayores retos de la industria de la automatización: comunicar dispositivos, controladores y/o aplicaciones, sin caer en los problemas habituales de las conexiones basadas en protocolos propietarios (figura 1).

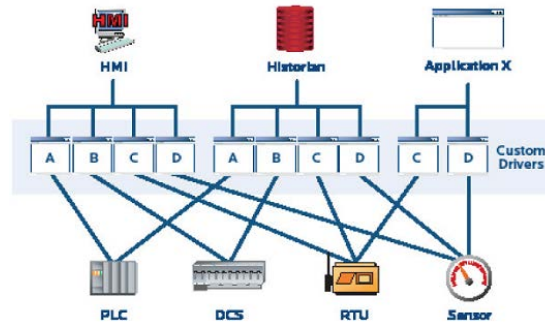


Figura 1 Problema de controladores personalizados.

Cualquier software que sea compatible con estas especificaciones, OPC proporciona a los usuarios e integradores, conectividad abierta e independiente, tanto del fabricante del dispositivo como del desarrollador de la aplicación cliente.

### **El OPC donde los protocolos propietarios fallan**

La clave del éxito de OPC en crear comunicaciones auténticamente independientes del fabricante estriba en que OPC, abstrae de los detalles de la implementación de fuentes de datos (PLC) y clientes de datos (HMI/SCADA), con lo que los datos se pueden intercambiar entre ellos sin que se conozcan los protocolos de comunicación nativos y de la organización interna de sus datos.

### **Comunicación del OPC (conceptualmente)**

Se puede representar como una capa de “abstracción” intermedia que se sitúa entre la fuente de datos y el cliente de datos, permitiéndoles intercambiar datos sin saber nada el uno del otro.

### **Funcionamiento del OPC (Funcionalmente)**

La “abstracción de dispositivo” OPC se consigue utilizando dos componentes OPC especializados llamados cliente OPC y servidor OPC. Cada uno de ellos es descrito en la siguiente sección. Es importante resaltar que el hecho de que la

fuentes de datos y el cliente de datos puedan comunicarse entre sí mediante OPC no significa que sus respectivos protocolos nativos dejen de ser necesarios o hayan sido reemplazados por OPC. Al contrario, estos protocolos y/o interfaces nativos siguen existiendo, pero sólo comunican con uno de los dos componentes del software OPC. Y son los componentes OPC los que intercambian información entre sí, cerrando así el círculo. La información puede viajar de la aplicación al dispositivo sin que estos tengan que hablar directamente entre sí.

### **Beneficios de utilizar conectividad OPC**

A primera vista, crear un driver propietario para dos componentes OPC (Cliente OPC y Servidor OPC), puede parecer que no sea importante, pero la experiencia ha demostrado lo contrario. Algunos de los beneficios clave de utilizar OPC, son:

- Una aplicación cliente OPC puede comunicarse libremente con cualquier servidor OPC visible en la red, sin la necesidad de utilizar ningún driver específico para la fuente de datos.
- Las aplicaciones cliente OPC pueden comunicarse con tantos servidores OPC como necesiten. No hay ninguna limitación inherente a OPC en el número de conexiones que se pueden establecer.
- Hoy en día hay un servidor OPC disponible para prácticamente todos los dispositivos nuevos o antiguos que existen en el mercado.
- Las fuentes de datos (hardware o software) que utilizan OPC pueden ser intercambiadas sin la necesidad de actualizar los drivers utilizados por cada aplicación que comunique con ellas mediante OPC. Sólo hay que mantener actualizado el Servidor OPC para esa fuente de datos.
- Los usuarios pueden elegir libremente los dispositivos, controladores y aplicaciones que mejor se ajusten a sus proyectos sin preocuparse del fabricante del que provienen o de si comunicarán entre sí.

### **Tipos de datos soporta OPC**

Los tipos de datos más comunes transferidos entre dispositivos, controladores y aplicaciones en automatización se pueden encuadrar en tres categorías:

- Datos de tiempo real.
- Datos históricos.
- Alarmas y Eventos.

### **¿Qué es un Servidor OPC?**

Un Servidor OPC es una aplicación de software. Un driver “estandarizado” desarrollado específicamente para cumplir con una o más especificaciones OPC. La palabra “Server” en “OPC Server” no hace referencia en absoluto al ordenador donde este software se estará ejecutando. Hace referencia a la relación con el Cliente OPC. Los Servidores OPC son conectores que se pueden asimilar a traductores entre el mundo OPC y los protocolos nativos de una fuente de datos. OPC es bidireccional, esto es, los Servidores OPC pueden leer y escribir en una fuente de datos. La relación Servidor OPC/Cliente OPC es de tipo maestro/esclavo, lo que significa que un Servidor OPC sólo transferirá datos a una fuente de datos si un cliente OPC, así se lo requiere.

### **¿Qué es un Cliente OPC?**

Un Cliente OPC es una pieza de software creada para comunicarse con Servidores OPC. Utiliza mensajería definida por una especificación concreta de la OPC Foundation.

### **Quehacer de un Cliente OPC**

- Conceptualmente. Un Cliente OPC representa un destino de datos. Inician y controlan la comunicación con Servidores OPC, basados en las peticiones recibidas desde la aplicación en la que están embebidos. Los Clientes OPC traducen las peticiones de comunicación provenientes de una aplicación dada en la petición OPC equivalente y la envían al Servidor OPC adecuado para que la procese.
- Técnicamente, los Clientes OPC son módulos de software utilizados por una aplicación que le permite comunicarse con cualquier Servidor OPC compatible visible en la red. Típicamente, los Clientes OPC están

embebidos en aplicaciones como HMI'S, SCADA'S, graficadores, generadores de informes, convirtiéndolos en aplicaciones compatibles OPC.

### **Comunicación de un Cliente OPC con múltiples dispositivos (Servidores OPC) de forma simultánea**

Es importante recalcar que los Clientes OPC, por su diseño, sólo son capaces de comunicar con Servidores OPC, no con los dispositivos finales. Esta aclaración es necesaria porque los Clientes OPC deben ser independientes de protocolos, puesto que de otra forma se repetirá el error del dispositivo-driver del pasado. Los Clientes OPC pueden comunicarse de forma simultánea con múltiples Servidores OPC. Esto significa que un Cliente OPC puede leer y escribir datos desde y hacia múltiples dispositivos (fuentes de datos), a través de sus respectivos Servidores OPC.

## **2. Métodos**

El tipo de investigación desarrollada para este proyecto es descriptiva. Para la elaboración del módulo OPC fue necesario contar con un gabinete industrial metálico para equipo eléctrico, en donde se colocaron los PLC'S MicroLogix 1500 de Allen-Bradley y el OPLC Vision 130 de UNITRONICS, éste último tiene la versatilidad de poseer un panel de operación gráfica, así como también el Panel View 600 de Allen-Bradley. Además, se utilizaron los siguientes softwares: RS Classic, RS Logix 500 PRO y PanelBuilder32 de desarrollo de aplicaciones para terminales PanelView estándar. Dichos softwares permiten la vinculación y la comunicación para el desarrollo del sistema OPC que controla el arranque y paro de un motor trifásico.

## **3. Resultados**

### **Realización de la programación en el módulo OPC:**

- **Configuración del software RSLinx:** El primer paso consiste en configurar el software RSLinx y el driver DF1, esto sirve para realizar una conexión punto a punto (figura 2).

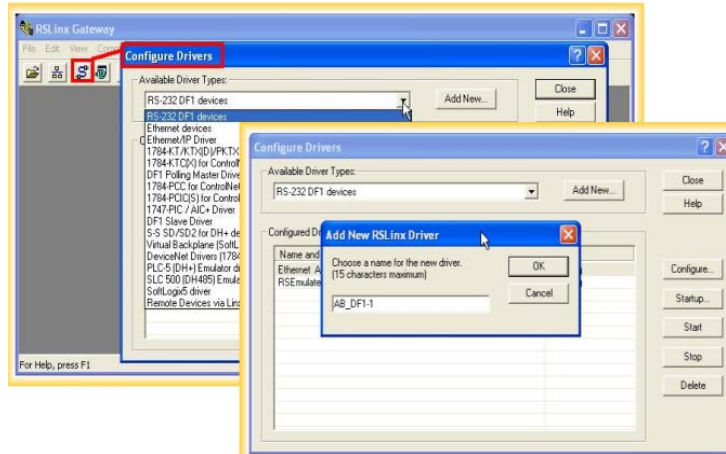


Figura 2 Pantalla que muestra como configurar el driver DF1 del software RSLinx.

Se observa que el software detecta el PanelView 600, y con esto el dispositivo está preparado para ejecutar el PanelBuilder 32 y cargarlo, antes se tiene que poner el panel en modo de transferencia, accediendo al modo de configuración, de lo contrario se ejecutará la aplicación que tiene precargada (figura 3).

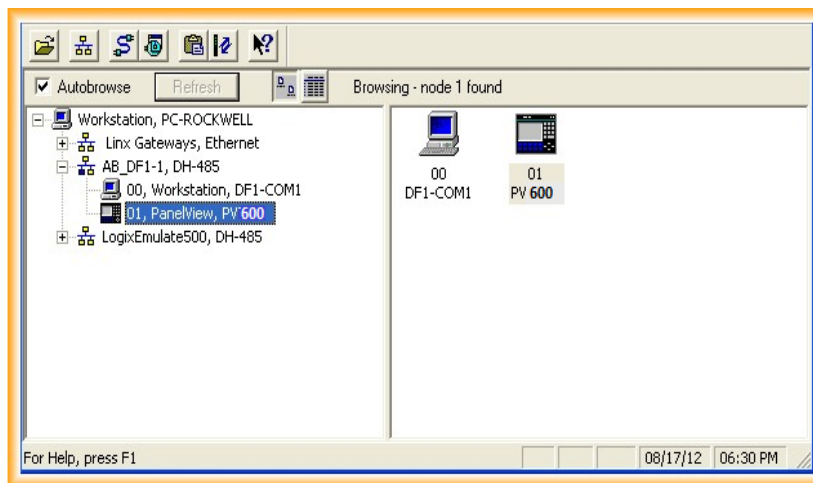


Figura 3 Detección del PanelView.

Para ejecutar el PanelBuilder32, File-> Upload, aparece una ventana donde se puede configurar el directorio para la aplicación y especificar el tipo de comunicación (figura 4).



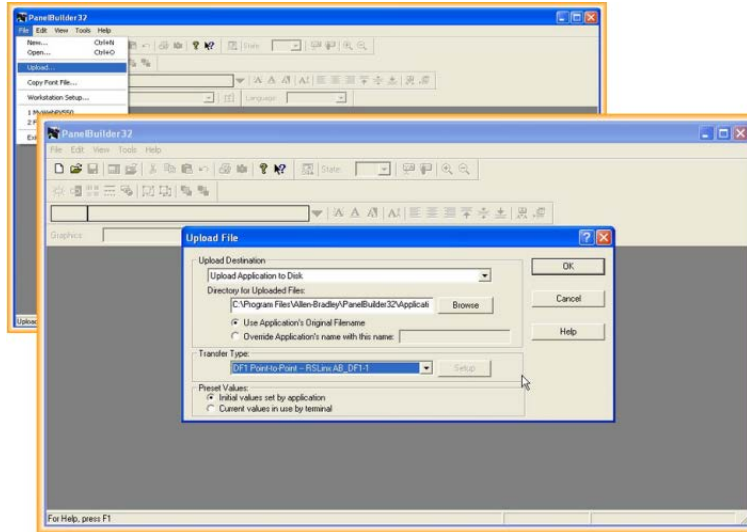


Figura 4 Ejecución del PanelView.

Una vez especificado el tipo de comunicación, se procede a pulsar OK y la transferencia se inicia. Con este último paso, se realiza el “Backup” (respaldo) del proyecto (figura 5).

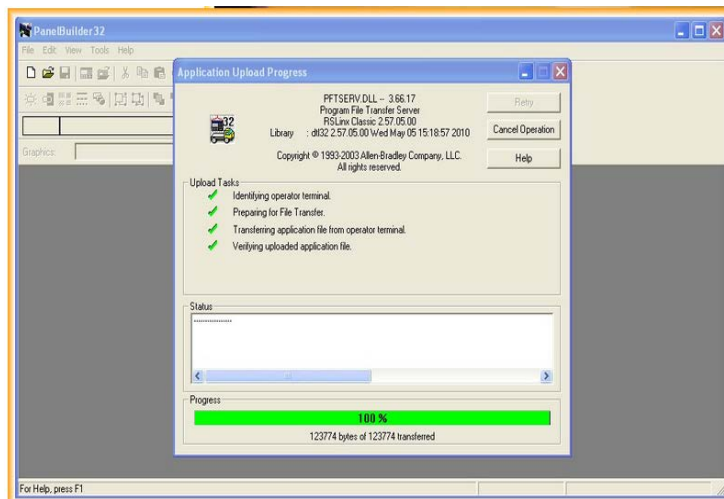


Figura 5 Finalización del PanelView.

- **Comunicación de PanelView 600 AB con PLC Allen Bradley Micrologix 1500.** Una vez establecida la comunicación con el PanelView, se realiza la comunicación con el PLC Allen Bradley Micrologix 1500; de la misma forma que con el PanelView a través del software RSLinx (figura 6).

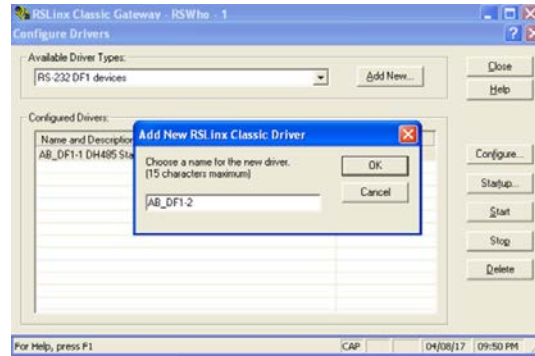


Figura 6 Comunicación del PLC Allen Bradley Micrologix 1500.

Posteriormente, se configura el puerto COM a utilizar (en este caso el puerto 3). Sólo dar clic en el botón de “auto-configurar”, el cual se encargará de realizar la correspondiente configuración en base a los parámetros autogenerados y después se hace click en el botón “Ok” (figura 7).

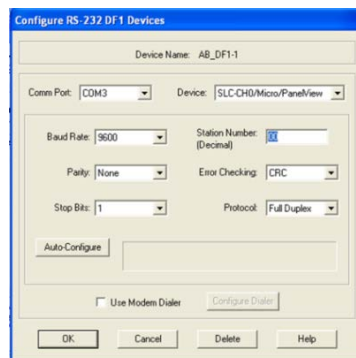


Figura 7 Configuración del puerto COM del PLC Allen Bradley Micrologix 1500.

Por último, en la figura 8 se muestra cómo se ejecuta el PLC Allen Bradley Micrologix 1500.

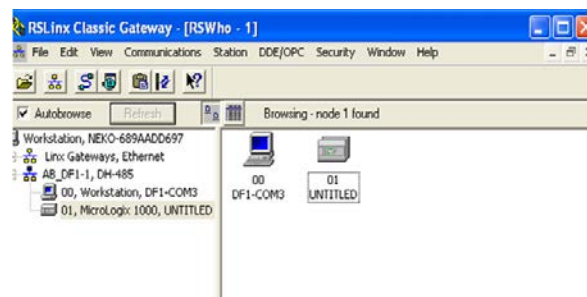


Figura 8 Ejecución del PLC Allen Bradley Micrologix 1500.

- **Se configura el PanelBuilder32.** Y se selecciona “crear una nueva aplicación” (figura 9).

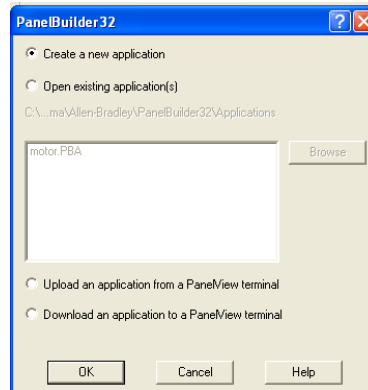


Figura 9 PanelBuilder32.

Se establecen los parámetros acordes al PanelView a trabajar. Y se muestra en pantalla cómo se puede programar el control del arranque y paro de un motor trifásico (figura 10).

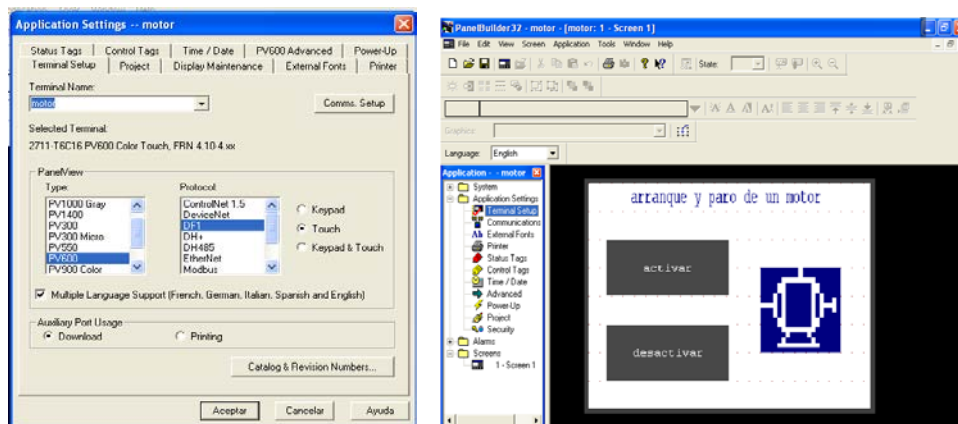


Figura 10 PanelBuilder32.

Una vez creada la pantalla principal del programa, sólo basta editar las etiquetas para comunicar el PanelView con el PLC, por lo cual se procede a abrir el RSLogix 500 PRO y se crea el programa que se cargará al PLC; esto no resulta difícil ya que se tiene comunicación a través del software RSLinx (figura 11).

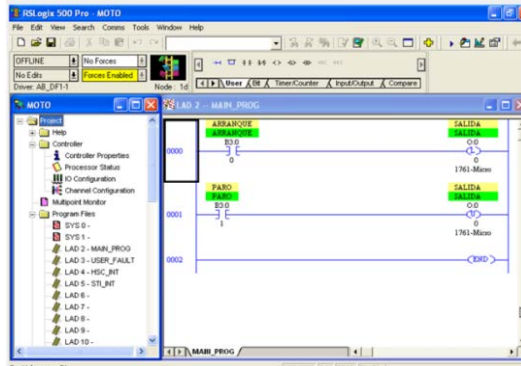


Figura 11 Programa para controlar el arranque y paro de un motor trifásico.

Después de cargar el programa al PLC, se importan los “Tags” (etiquetas), es decir; se realiza la declaración de variables mediante la siguiente ruta: tools/database/ASCII EXPORT/CSV. Y se asegura que la casilla “add/symbol Desc.” esté marcada y se le da aceptar (figura 12).

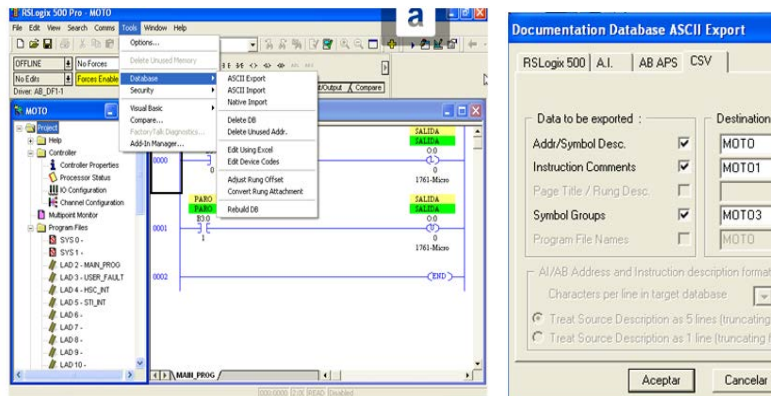


Figura 12 Importación de Tags.

Una vez que concluye el paso anterior que utiliza el software del PLC RSLogix500, se realiza el mismo procedimiento para el PanelBuilder32. Cuando está abierto el Tag editor, se puede importar todos los Tags File -> Import Tags. Dependiendo del formato que se haya seleccionado en la exportación, debe ser el mismo formato en el PanelBuilder (figura 13).

Una vez importados los Tags, basta vincularlos con el botón de arranque y paro creados en la pantalla HMI. Posteriormente, se activa el botón de arranque dando doble click sobre él y éste desplegará una ventana donde

se selecciona el botón que dice Edit Tag, se da click y listo. En este punto, se puede editar el Tag agregado. Se realiza el mismo procedimiento para el botón de paro (figura 14).

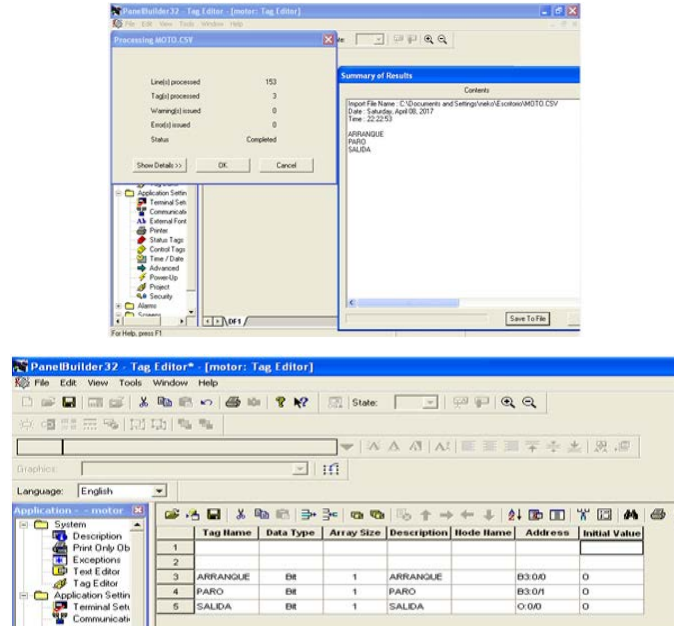


Figura 13 Importación de tags en PanelBuilder

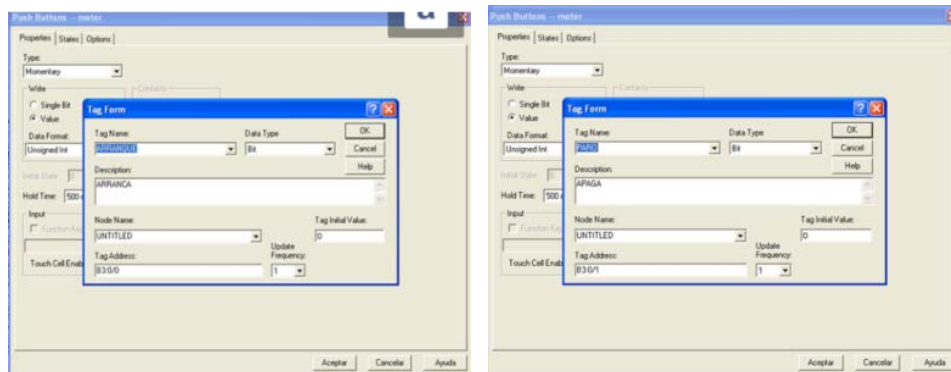


Figura 14 Activación de botones de arranque y paro de un motor trifásico.

Por último, se carga el programa al PanelView600 y se comprueba que exista comunicación y que el programa funcione de forma correcta. Si se detiene el PLC, se comprueba el fallo que nos muestra en el PanelView "684 Required network node not found".

## 4. Análisis y discusión

Para obtener los datos de las variables de entrada necesarias para el diagnóstico proveniente del servidor OPC se desarrolló una herramienta gráfica que permite la extracción de datos del servidor OPC en cuestión. La figura 15, muestra la interfaz desarrollada para conectarse a un servidor OPC y extraer sus datos.

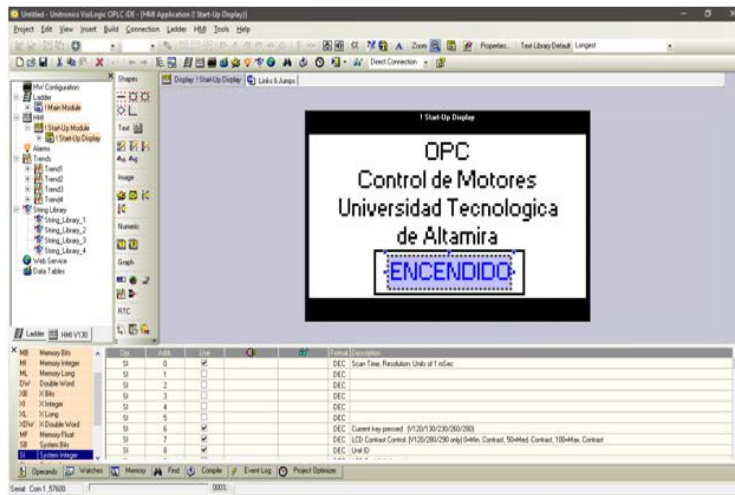


Figura 15 Interfaz gráfica del PLC Vision 130 de UNITRONICS.

### Programación del OPLC Vision 130 de UNITRONICS

Al ejecutar el programa se selecciona el dispositivo OPLC Vision 130 de UNITRONICS (figura 16).



Figura 16 PLC Vision 130 de UNITRONICS.

Posteriormente, se elige el modelo de entradas y salidas, para después dar click en OK. En la primera pantalla se realiza la interfaz gráfica del OPLC Vision 130 de UNITRONICS (figura 17).

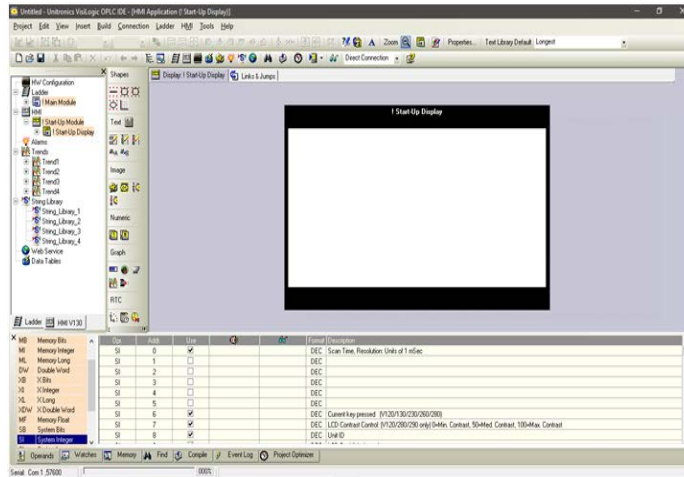


Figura 17 Diseño de interfaz gráfica del PLC Vision 130 de UNITRONICS.

Después, se utiliza la herramienta Text para colocar un cuadro con la siguiente información: “OPC Control de Motores, Universidad Tecnológica de Altamira”. Luego se utiliza la herramienta Binary Text para aplicarlo como indicador (figura 18).



Figura 18 Programación de interfaz gráfica del PLC Vision 130 de UNITRONICS.

Como siguiente paso, se realiza la programación en escalera. Y se coloca un contacto cerrado y una memoria, y se configura (figura 19). Y finalmente, se carga el programa (figura 20).

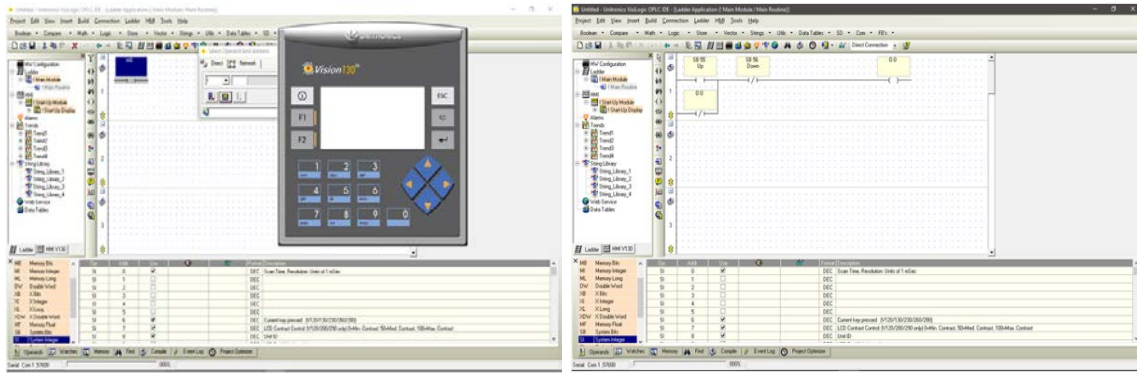


Figura 19 Programación en escalera del PLC Vision 130.

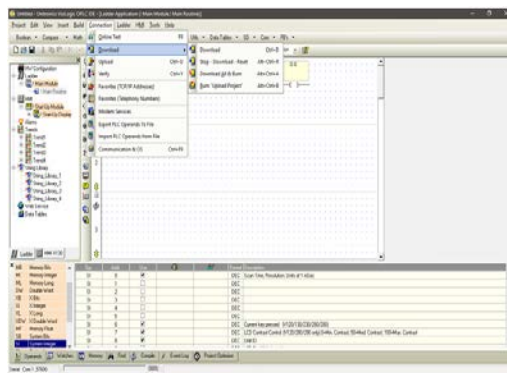


Figura 20 Carga del programa de arranque y paro de motor con PLC Vision 130.

En la figura 21 muestra algunas fotografías del diseño del módulo OPC, para el paro y arranque de un motor trifásico.



Figura 21 Construcción de equipo OPC de paro y arranque de motor trifásico.

## 5. Conclusiones

El contar con un servidor OPC, permitió construir una aplicación de control para el arranque y paro de un motor trifásico de una manera fácil y rápida, haciendo



que no importe el tipo de hardware ni el formato o medio físico en que se transmiten los datos, con OPC se tiene una gran flexibilidad a la hora de producir sus propios sistemas SCADA, si en algún momento se desea hacer una renovación tecnológica o una ampliación, no se está sujeto a tener que adquirir equipos industriales del mismo fabricante de sus equipos anteriores.

La aplicación desarrollada para el control, arranque y paro de un motor trifásico es muy fácil configurar e instalar, sin excesivas exigencias de hardware, con interfaces amigables y de arquitectura abierta, capaz de crecer y adaptarse a las necesidades cambiantes del propio OPC. Si se usa un servidor OPC, los sistemas SCADA pueden ser diseñados a la medida de las necesidades de los usuarios.

Se verificó la capacidad de operación del software en el servidor OPC, teniendo éxito en las pruebas de arranque y paro de motor trifásico.

Las experiencias y pruebas realizadas demuestran que este tipo de proyectos puede ser de gran valor tanto en ambientes industriales reales como en el área educativa.

Se pretende utilizar este servidor OPC en el Laboratorio Pesado de Mecatrónica, como apoyo en las materias de Automatización y Control, PLC'S e Instrumentación Virtual; en donde los alumnos realizan el estudio de programación de PLC'S y programación gráfica en LabView de National Instruments con el dispositivo CompactRIO (plataforma que tiene una variedad de controladores embebidos con dos objetivos de procesamiento:

- Un procesador en tiempo real para comunicación y procesamiento de señales.
- FPGA programable por el usuario para implementar control de alta velocidad y temporización y disparo personalizados directamente en hardware).

## **5. Pares Revisores**

### **Revisor 1**

Nombre: MC. Óscar Martínez Hernández  
Institución: Instituto de Estudios Superiores de Tamaulipas

Cédula Profesional: 6666064  
Área de conocimiento: Ingeniería  
Correo electrónico: oscar.martinez@iest.edu.mx  
Teléfono: 833 235 6159

## **Revisor 2**

Nombre: Dr. José Genaro González Hernández  
Institución: Tecnológico Nacional de México/Instituto Tecnológico de Ciudad Madero  
Cédula Profesional: 8713470  
Área de conocimiento: Metodología de la Educación  
Correo electrónico: Jgenaro.gonzalez@itcm.edu.mx  
Teléfono: 833 388 5403

## **7. Bibliografía y Referencias**

- [1] Darek Kominek, P. Eng. Alberta, Canadá (2009). Matrikon OPC. OPC: ¿De qué se trata, y cómo funciona? "Guía para entender la Tecnología OPC". Fecha de consulta: 22 de noviembre de 2017: <http://www.MatrikonOPC.es>.
- [2] Wang L. y Tan K. C. (2006). "Modern Industrial Automation Software Design". John Wiley & Sons, Inc., Hoboken, New Jersey, USA.
- [3] Distefano M. (1999). "Comunicaciones en Entornos Industriales". Facultad de Ingeniería. Universidad Nacional de Cuyo, Argentina.
- [4] Villegas, Ángel, Herrera, Isaac, Gómez, Gerardo, Rodríguez, José, Lugo, Edgar, Pacheco, José (2008). Aplicación para el monitoreo y control de procesos industriales basada en el estándar de comunicaciones OPC. Revista Ingeniería UC: <http://www.redalyc.org/articulo.oa?id=70712293002>.
- [5] Wolfgang Mahnke, Stefan-Helmut Leitner (2009). Arquitectura OPC unificada. La norma futura para la modelización de las comunicaciones y la información en la automatización. Revista ABB: [http://www.infoplc.net/files/documentacion/comunicaciones/infoplc\\_net\\_guia\\_para\\_entender\\_la\\_tecnologia\\_opc.pdf](http://www.infoplc.net/files/documentacion/comunicaciones/infoplc_net_guia_para_entender_la_tecnologia_opc.pdf).
- [6] Shimanuki Y. (1999). "OLE for process control (OPC) for new industrial automation systems". Proceedings of the IEEE International Conference on

Systems, Man, and Cybernetics (SMC 99), Vol. 6, pp. 1048-1050. Tokio, Japón.

- [7] [http://support.elmark.com.pl/unitronics/pdf/vision/V130/v130-33-ra22\\_instl-guide\\_12-07.pdf](http://support.elmark.com.pl/unitronics/pdf/vision/V130/v130-33-ra22_instl-guide_12-07.pdf)
- [8] [http://literature.rockwellautomation.com/idc/groups/literature/documents/gr/2711-gr003\\_-es-p.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/gr/2711-gr003_-es-p.pdf)
- [9] <http://www.etitudela.com/entrenadorcomunicaciones/downloads/labviewintroduccionopcserver.pdf>.