

# **Desarrollo e implementación de una interfaz en el Software Labview Ni, que permita el accionamiento, monitoreo y la puesta en marcha de los Variadores Sinamics G120 a través del protocolo de comunicación Modbus Rtu**

Karla Goya <sup>(1)</sup>, Holger Cevallos MSc. <sup>(2)</sup>  
Facultad de Ingeniería Eléctrica y Computación (FIEC) <sup>(1)(2)(3)</sup>  
Escuela Superior Politécnica del Litoral (ESPOL)  
Campus Gustavo Galindo, Km 30.5 vía Perimetral  
Apartado 09-01-5863. Guayaquil-Ecuador  
[kagagoya@espol.edu.ec](mailto:kagagoya@espol.edu.ec) <sup>(1)</sup>  
[hcevallos@fiiec.espol.edu.ec](mailto:hcevallos@fiiec.espol.edu.ec) <sup>(2)</sup>

## **Resumen**

*El presente proyecto surge según la necesidad que existe de que los estudiantes de la carrera de Ingeniería en Electricidad y electrónica y automatización industrial, se relacionen con los diferentes equipos, marcas y comunicación de red que existen en el sector industrial; y para apoyar al mejoramiento y fortalecimiento de la enseñanza mediante la práctica y el desarrollo tecnológico, el cual permite a los futuros ingenieros aprender sobre las tecnologías y procesos de control y automatización, que se usan actualmente.*

*El proyecto se enfoca en el desarrollo e implementación de una interfaz en el Software Labview Ni, que permita el accionamiento, monitoreo y la puesta en marcha de los variadores Sinamics G120 a través del protocolo de comunicación Modbus RTU el cual servirá para la utilización de los estudiantes en los diferentes laboratorios de control. El objetivo principal de nuestro proyecto constituye en desarrollar una interfaz en el software Labview Ni que sea muy amigable con el usuario y que por medio del protocolo de comunicación Modbus RTU se comunique con los variadores de velocidad Sinamics G120; con él con lo cual podemos acceder a todos sus registros y parámetros para poder ser modificados según nuestros requerimientos, y a su vez poder controlarlos y monitorearlos desde dicha interfaz.*

**Palabras Claves:** Interfaz, Modbus Rtu, Labview, Variadores, protocolo, Control

## **Abstract**

*This project arises according to the need of the students of engineering in electrical and electronics and industrial automation, relate with the different teams, brands and network communication that exist in the industrial sector; and to support the improvement and strengthening of education through practice and technological development, which allows future engineers to learn about technologies and processes of control and automation, which are currently used.*

*Project focuses on the development and implementation of an interface in the Ni Labview Software, allowing the drive monitoring and the implementation of the Sinamics G120 drives through the communication protocol Modbus RTU which will serve for the use of students in the different control laboratories. The main goal of our project is to develop an interface in the Labview software or be friendly with the user and that through the communication protocol Modbus RTU to communicate with Sinamics G120 drives; with it with which we can access all their records and parameters to be able to be modified according to our requirements, and at the same time be able to control and monitor them from that interface.*

**Keywords:** Interface, Modbus Rtu, Labview, Tia Portal, Data Source, Control



incluye el API NI-VISA para tener acceso a las características avanzadas del puerto serial y escribir códigos independientes de la interfaz que es compatible con cualquier interfaz que soporta al Virtual Instrument Software Architecture (VISA).



Figura 3. Interfaz RS485/422 de un puerto para USB

### 2.5. Panel básico operador (bop-2)

En el panel básico debemos hacer cambios en ciertos parámetros para poder dar paso a la comunicación Modbus. Configuración básica para la comunicación la dirección Modbus RTU del convertidor puede definirse mediante los interruptores de dirección de la Unidad De Control mediante el parámetro p2021.

Interruptor de dirección: Si se ha predeterminado una dirección válida por medio de los interruptores de dirección, siempre está activa esa dirección y el parámetro p2021 no se puede modificar.

Tabla 1.- Procedimiento para el ajuste de la dirección del bus de campo

| Parámetro | Descripción   |
|-----------|---|
| P0015=21  | Macro Unidad de accionamiento<br>Selección de la configuración de E/S   |
| P2030=2   | Selección de protocolo bus de campo<br>2: Modbus  |
| P2020     | Velocidad de transferencia bus de campo<br>Para la comunicación se pueden ajustar velocidades de transferencia de 4800 bits/s ... 187500 bits/s; el ajuste de fábrica es = 19200 bits/s |
|           | Modbus Timing<br>Índice 0: tiempo máximo de procesamiento esclavo-telegrama:<br>Tiempo máximo que puede transcurrir antes de que el esclavo   |

|       |   |
|-------|---|
| P2024 | envíe respuesta al maestro.<br>Índice 1: Tiempo de retardo de caracteres: Tiempo de retardo de caracteres: Retardo máximo admisible entre los distintos caracteres dentro de un frame de Modbus. (Tiempo de procesamiento estándar de Modbus para 1,5 bytes).<br>Índice 2: tiempo de pausa entre telegramas: Retardo máximo admisible entre telegramas Modbus. (Tiempo de procesamiento estándar de Modbus para 3,5 bytes). |
| P2029 | Estadística de errores de bus de campo<br>Indicación de los errores de recepción en la interfaz del bus de campo.   |
| P2040 | Tiempo de vigilancia de datos de proceso<br>Determina el tiempo que debe transcurrir para que se genere una alarma si no se transmiten datos de proceso.<br>Este tiempo debe ajustarse en función del número de esclavos y de la velocidad de transferencia ajustada en el bus (ajuste de fábrica = 100 ms).  |

### 3. Estructura de Comunicación Sinamics G120

Parte esencial de nuestro trabajo de investigación es establecer la comunicación Modbus con el software LabView 2014 y para esto debemos tener en cuenta la estructura de comunicación que permiten los variadores de velocidad Sinamics G120, según la unidad de control que hemos escogido la CU240B-2 la cual nos permite trabajar con comunicación Modbus RTU en la cual estableceremos la configuración de la comunicación y todos los requerimientos que este exige para poder lograr tener acceso a los registros de lectura y escritura que este posee

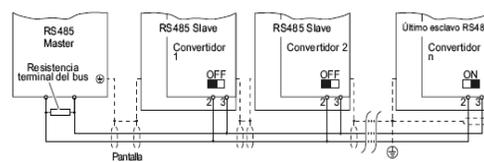


Figura 4. Red de comunicación a través de RS-485

Se debe conectar la resistencia terminal del bus para la primera y la última estación. Podemos consultar la posición de la resistencia terminal del bus en la figura. Se puede retirar uno o varios esclavos del bus desenchufando el conector de bus sin que se interrumpa la comunicación para las otras estaciones, pero no el primero ni el último. Durante el funcionamiento con bus, la primera y la última estación del bus deben recibir tensión continuamente.

### Interfaz en serie

La comunicación con Modbus RTU se realiza a través de la interfaz RS485, con un máximo de 247 esclavos. La longitud máxima del cable es de 1200 m (3281 pies).

## 5. Solución Propuesta

Diseñaremos una interfaz gráfica en el software Labview Ni la cual debe ser muy amigable al usuario, que por medio del protocolo de comunicación Modbus RTU pueda acceder a los registros y diferentes parámetros del variador Sinamics G120 el cual permite la visualización, monitoreo y modificación de diferentes parámetros para administrar diversas condiciones de funcionamiento en la puesta en marcha de los variadores de velocidad Sinamics G120.

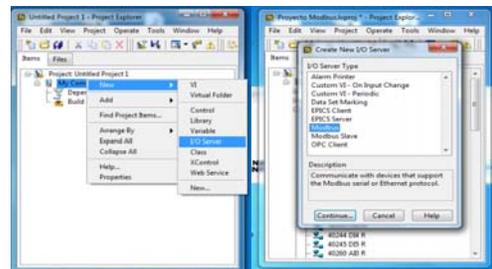
El desarrollo de este proyecto nos permitirá poner en práctica todos los conocimientos adquiridos a lo largo de nuestra carrera, como es el uso del software Labview que aparte de educativo también puede dar grandes soluciones a diferentes problemas y necesidades y sobretodo poder demostrar que por el avance y las tecnologías que existen en los proveedores de equipos eléctricos en el campo industrial ahora es mucho más factible la interacción humano – máquina.



**Figura 5.** Diagrama general de conexiones y equipos usados en el proyecto.

## 4. Comunicación Entre la Unidad de Control del Variador Sinamics G120 y el Software Labview Ni

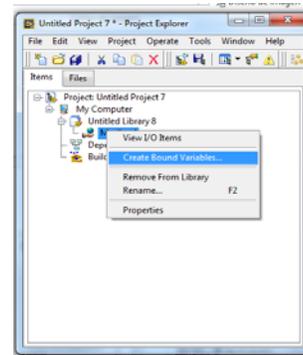
Para establecer la Comunicación Labview – Variador, tenemos que configurar al maestro y al esclavo, empezaremos con el maestro en este caso LabView, Creamos un programa nuevo en Labview al cual le llamaremos proyecto Modbus. En el cual crearemos todas las variables que permite la asignación de los registros Modbus a los parámetros de la unidad de control del variador. En *My Computer*, damos click en *New* donde escogeremos la opción *I/O Server*, donde elegiremos que tipo de I/O server queremos usar, para nuestro caso escogemos *Modbus*.



**Figura 6.** Crear un Nuevo I/O server.

Configurar Modbus I/O Server, es importante escoger el puerto serial adecuado, eso dependerá del puerto serial de la cada PC.

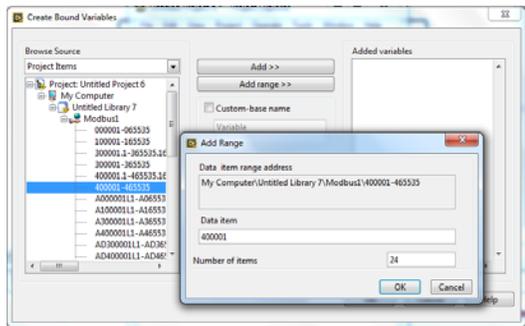
Luego de crear el servidor Modbus, damos click y aparecerá una opción que dice *Create Bound Variables*, en donde crearemos todas las variables que permitirán la asignación de registros de Modbus a los parámetros de la unidad de control del variador.



**Figura 7.** Creación de variables

Al escoger la opción de la que hablamos aparece la siguiente pantalla mostrada en la figura 4.7, donde damos click en *Project/ My computer/ library/ Modbus*

donde luego aparece las direcciones asignadas de la tabla de datos de referencia de Modbus. En la cual escogeremos los registros de retención sin signo de 16 bits enteros, de lectura o escritura representados por 400001–465535. Luego escogemos la opción *Add range* y donde dice *Numbers of ítems*, escribimos el número de variables que queremos crear.



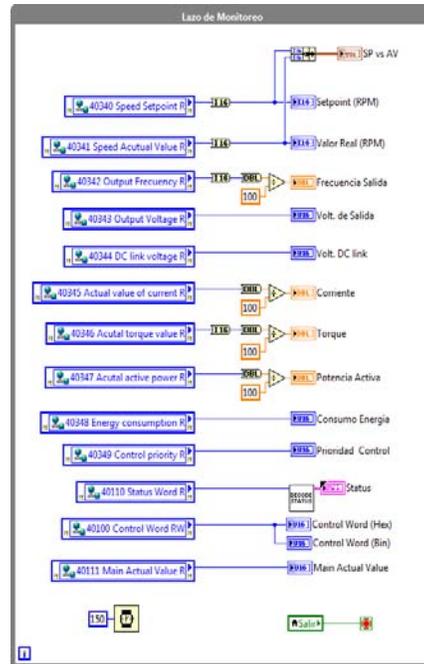
**Figura 8.** Creación de las variables de Modbus

Después de crear todas las variables necesarias para establecer correctamente la comunicación, agregamos un *Vi*, en donde creamos un lazo de monitoreo en el cual vamos a llamar a todas las variables de lectura que creamos anteriormente desde una *shared variable*.



**Figura 9.** Creación de las Shared Variables.

Hemos creado el lazo de monitoreo con todas las variables necesarias como podemos observar en la figura 9 y en la figura 10 podemos observar la prueba de lectura con valores escalables los cuales están dado por el fabricante del convertidor.

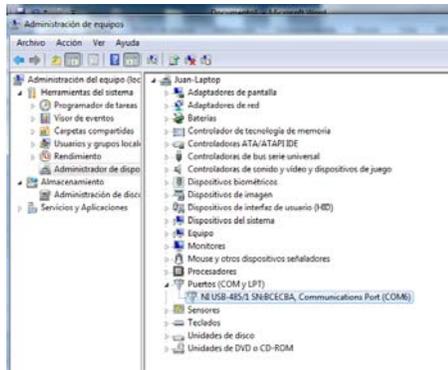


**Figura 10.** Lectura de variables con valores escalados.

Después de haber creado los lazos de monitoreo para poder confirmar que hemos logrado establecer la comunicación Variador – Labview se deben seguir los siguientes pasos en la pc, en la conexión de los cables y en los conectores seriales para conseguir el correcto funcionamiento del programa.

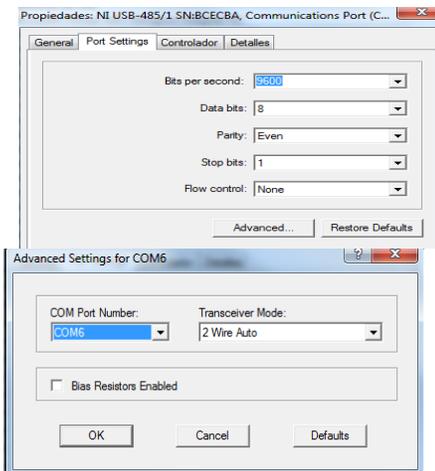
Con el computador apagado, conectamos el cable conversor de USB a RS-485, y luego encendemos el computador. Es muy importante conectar el cable antes de encender el computador, caso contrario el cable podría no ser reconocido adecuadamente. Adicionalmente trabajamos con el conector serial DB 9 hembra el cual conectamos por un lado al conector RS-485 y al Variador de Velocidad que tiene una regleta de bornes asignadas.

A continuación se debe configurar el puerto en Windows, para ello hacemos abrimos el Administrador de Equipos de Windows (clic derecho en Equipo>>Administrar...)



**Figura 11.** Pasos de configuración para la comunicación

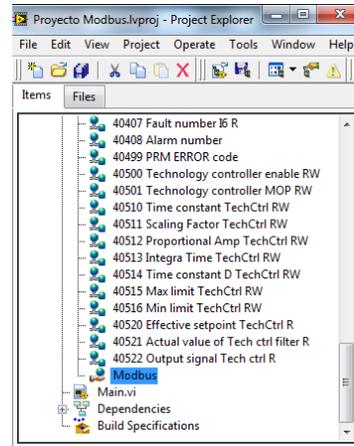
En el Administrador de Equipos seleccionamos Administrador de Dispositivos y navegamos hasta Puertos (COM y LPT) y hacemos doble clic en el dispositivo NI USB485/1. En la pestaña de Port Settings nos aseguramos que la configuración sea la siguiente.



**Figura 12.** Pasos de configuración para la comunicación

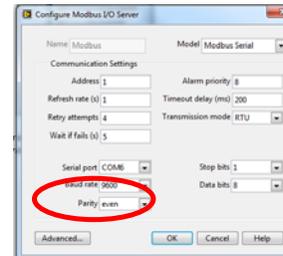
Es importante que el puerto en el que se configure sea el COM6. Si por alguna razón este puerto no está disponible, seleccionar otro puerto y se debe cambiar el puerto por defecto usado en el programa como se indica más adelante. En el modo de transmisión debemos escoger 2 Wire Auto ya que esta representa la conexión que hemos realizado en los pines de los conectores involucrados. Hacemos clic en aceptar para cerrar todas las ventanas abiertas.

Abrimos el proyecto Proyecto Modbus.lvproj. en el proyecto, expandimos Librería IO y damos clic derecho en Modbus.



**Figura 13.** Pasos de configuración en LabView

En propiedades nos aseguramos que la configuración esté como se indica a continuación, y en la configuración de Serial Port seleccionamos el puerto configurado anteriormente. Luego damos clic en OK para cerrar esa ventana.

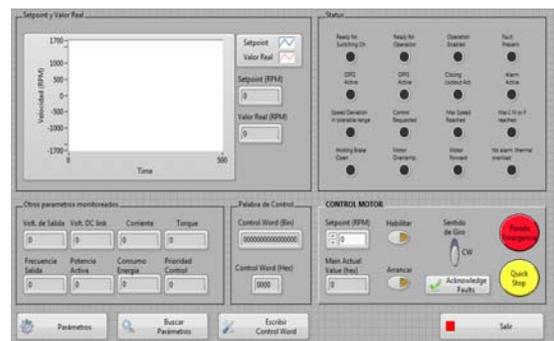


**Figura 14.** Configuración Modbus

Damos doble clic en Main.vi para abrir el programa principal corremos el programa y verificamos si existe la comunicación.

## 6. Diseño de Interfaz para usuario en Labview.

Al iniciar el programa tenemos el siguiente panel frontal: el cual describiremos a continuación.



**Figura 15.** Interfaz Gráfica del programa.

**Setpoint y Valor Real:** En este lugar se grafican el set point junto con el valor real enviado por el variador. Además se muestran los valores numéricos de dichos valores. El signo de la velocidad indica la dirección de giro.

**Status:** Se monitorea la palabra de control de Status del variador, que envía la información mostrada en el panel frontal. Referirse al anexo A para conocer más de la palabra Status.

**Otros parámetros monitoreados.** Se monitorea información enviada por el variador. Esta información está adecuadamente escalada según las unidades mostradas.

**Palabra de Control:** Se muestra el valor de la palabra de control

**Control Motor:** Desde los controles indicados se realiza el control del motor, como por ejemplo cambio de setpoint, arrancar el motor, parada de emergencia, invertir giro y demás.

Se tiene además botones que sirven para configurar el sistema.

**Parámetros:** Abre una ventana de configuración de los parámetros más comunes

**Buscar Parámetros:** Abre una ventana donde se pueden buscar y leer/escribir todos los parámetros asequibles a través de Modbus.

**Escribir Control Word (Avanzado):** Permite escribir directamente un valor en la palabra de control.

## 7. Conclusiones

1. Se concluye que mediante el diseño de la interfaz propuesta, hemos conseguido tener acceso a los registros de control Modbus que existen en el Variador de Velocidad Sinamics G120, los cuales podemos monitorear y controlar desde la interfaz que hemos diseñado. La cual nos permite emular y de cierta forma mejorar al software de programación creado por el fabricante.

2. El diseño y la implementación de este interfaz fue desarrollado de manera práctica y didáctica para labores de enseñanza y capacitación de los estudiantes, dejando además conocer la diversidad de funciones y aplicaciones que poseen los dispositivos.

3. Fue de suma importancia la revisión y utilización de los manuales de instrucciones de servicio emitidos por el fabricante, ya que contienen la información necesaria y correcta sobre conexiones, comunicación, funcionamiento y parámetros bajo los cuales los Variadores Sinamics desarrollan de forma correcta su funcionamiento.

4. La utilización del software LabVIEW permitió una fácil programación y simulación de los diferentes

aspectos necesarios para permitir la comunicación, control y monitoreo de los elementos y dispositivos necesarios para la puesta en marcha de los Variadores Sinamics G120.

5. Ya que las librerías que permiten la comunicación Modbus RTU son creadas según los registros de Modbus de la unidad de control de los variadores Sinamics, podemos concluir que al asignar las mismas librerías para otro tipo de variador ya sea de otra potencia, este seguirá teniendo la misma comunicación. De la misma forma para poder tener acceso a los registros Modbus de otros fabricantes, debemos cambiar las librerías existentes, por las que están dadas en los registros Modbus de otras marcas.

## 8. Agradecimientos

A mi Director de Proyecto el MSc. Holger Cevallos Ulloa por su dedicado apoyo. A todos los docentes que pudieron compartir su valioso conocimiento. A mis grandes amigos que hicieron muy divertida esta larga carrera.

## 9. Referencias

- [1] Wikipedia, "Variadores de Velocidad", [http://es.wikipedia.org/wiki/Variador\\_de\\_velocidad](http://es.wikipedia.org/wiki/Variador_de_velocidad), 2014.
- [2] Desarrollo Web, "Protocolos de comunicación", <http://www.desarrolloweb.com/articulos/1617.php>, 2014.
- [3] Siemens, Motores trifásicos eléctricos de baja tensión. [http://www.swe.siemens.com/spain/web/es/industry/drive\\_tech/Documents/catalogo\\_abreviadoMotoresBajatenion.pdf](http://www.swe.siemens.com/spain/web/es/industry/drive_tech/Documents/catalogo_abreviadoMotoresBajatenion.pdf), 2014.
- [4] LabView 2014, "Modulo LabView Data logging and Supervisory Control (DSC)", <http://www.ni.com/labview/labviewdsc/esa>, 2014.
- [5] Siemens Sinamics, "Sinamics G120 Standart Inverters" <http://www.enelva.rs/siemens/katalog/Kontrola%20kretanja/Sinamics%20G120.pdf>, 2014.
- [6] Siemens Automation, "Lista de Parámetros\_CU240B-E" [https://cache.automation.siemens.com/dnl/Tk0NTE5AAAA\\_23132008\\_HB/CU240S\\_23132008\\_COMP\\_es.pdf](https://cache.automation.siemens.com/dnl/Tk0NTE5AAAA_23132008_HB/CU240S_23132008_COMP_es.pdf), 2014.