

ANÁLISIS Y DISEÑO DE INTERFAZ PARA CONEXIÓN DE PLC BASADO EN ARDUINO CON PLC ALLEN BRADLEY A TRAVÉS DE ETHERNET.

Sistema de interconexión entre PLC de diferentes arquitecturas

Por el estudiante José Mario Valencia Henao

Al PhD. Guillermo Roberto Solarte Martinez

Universidad Tecnológica de Pereira

Correo Institucional: sejoma989@utp.edu.co

Antes de comenzar con este proyecto quiero agradecer primero que todo a la gran Universidad Tecnológica de Pereira, y a todos sus colaboradores, juntos me ha brindado todas las herramientas y conocimientos necesarios para la Ingeniería de sistemas, pero sobre todo para la vida, por hacer de mi no solo un profesional, sino una persona consciente de los problemas sociales y culturales que afectan la región cafetera y de los utensilios necesarios para analizarlos y atacarlos.

Agradecimiento inmenso a mi director de proyecto, PhD. Guillermo Solarte, un gran profesional y una gran persona que he tenido la fortuna y la oportunidad de conocer desde mis inicios en la carrera, el cual ha generado en mí los deseos y los conocimientos necesarios para llevar a cabalidad la finalización de este proyecto.

A mi padre, el Ingeniero Mario Valencia, orgulloso egresado de la facultad de ingenierías en el programa Ingeniería Eléctrica de la Universidad Tecnológica de Pereira, cuya guía, apoyo y consejo ha estado para mí en los momentos más difíciles de mi vida y mi carrera, fue él quien me apoyó desde el inicio de este proyecto que se llamó Ingeniero de Sistemas, y sus conocimientos amplios en ciencias básicas, matemáticas y Automatización Industrial, fueron las bases iniciales de este proyecto de Investigación, gracias por ser siempre mi héroe.

A mi madre, Luz Stella Henao, un apoyo incondicional en todas las etapas de mi vida, el ángel que Dios puso para alumbrar mi camino y jamás permitir que desistiera de mis deseos de ser profesional. Gracias por ser la fiel representación de Dios en la tierra.

A mi familia, mis hermanos y mis amigos que han sabido apoyarme y entenderme en las buenas y malas, personas de gran valor que con su gran amor le han dado gran sentido a mi vida y motivación para terminar este proyecto

Apéndice

1.	Abstract	Page 3	
2.	Introducción	Pág 4	
3.	Marco Teorico	Pag 4	
4.	Formulación del problema	Pág 5	
5.	Glosario de términos	Pág 6	
6.	Desarrollo de la metodología	pág 7	
a.	<i>Etapa 1, Investigación y captación de información</i>	Pág 7	
	i.	<i>Actividad 1, Investigar documentación relacionada a la creación de una API, SOA e interfaz web</i>	Pag 8
	ii.	<i>Actividad 2, Investigar y conocer el funcionamiento interno de un dispositivo PLC Allen Bradley y de un PLC basado en arduino.</i>	Page 8
	iii.	<i>Actividad 3, Analizar cómo cada dispositivo PLC ejerce control sobre el sistema.</i>	Pág 9
b.	<i>Etapa 2, diseño y divulgación</i>	Page 10	
	i.	<i>Actividad 1, Diseñar una interfaz que permita administrar PLC Allen Bradley desde un PLC basado en arduino</i>	Page 10
		1. Tarea 1, Levantamiento de requerimientos.....	Page 11
		2. Tarea 2, Elaboración de casos de uso	Pag 12
		3. Tarea 3, Desarrollo de especificación	Page 13
7.	Conclusiones	Pág 15	
8.	Referencias	Pág 18	

Abstract

En las próximas páginas de este proyecto de investigación, el lector puede encontrar el planteamiento para una interfaz que permita la conexión entre equipos PLC de diferente fabricante y filosofía de lucro de software, por un lado tenemos al gigante de la automatización industrial Allen Bradley, que tiene bastante presencia en nuestra región, y cuenta con una suite de software privativo que incrementa su costo proporcionalmente con su funcionalidad, y por otro lado Industrial Shields, una empresa que ha surgido en la búsqueda por equipos de bajo costo, cobijada por el apogeo y el boom de la tecnología basada en código abierto, la cual ofrece equipos con características de soporte industriales, programables en plataformas Open Source; todo esto con el fin de aprovechar las infraestructuras actualmente instaladas basadas en el software con propietario para la captación de datos físicos y abrir las barreras de limitaciones de software para que los datos de producción puedan ser exportados y visualizados a gusto y nivel de los ejecutivos, sin pensar en adquirir software en el rango de las decenas de millones de pesos sino invirtiendo en conocimiento aprovechando las puertas que se abren al programar en plataformas gratuitas y abiertas a la comunidad de programadores multidisciplinariamente.

Palabras clave: *Automatización Industrial, Open Source, PLC, Arduino, SCADA, Interfaz OPC.*

ANÁLISIS Y DISEÑO DE INTERFAZ PARA CONEXIÓN DE PLC BASADO EN ARDUINO CON PLC ALLEN BRADLEY A TRAVÉS DE ETHERNET.

Introducción

En el mundo de la industria, es común encontrar dentro de las grandes fábricas y empresas, que muchos de sus procesos, desde los operacionales hasta los de producción, han sido tecnificados y estandarizados utilizando herramientas en el mercado como Computadores, Sistemas embebidos, lógica cableada y Controladores Lógicos Programables. Estos últimos, mundialmente conocidos como PLC, se han vuelto el pilar de la industrialización de procesos de producción, puesto que son computadoras sin periféricos E/S, optimizadas con mucha robustez para la transmisión y recepción de datos en ambientes hostiles y de mucho ruido en términos de señales, que brindan a quien lo usa, la seguridad y confiabilidad de los datos procesados que viajan desde la máquina a la estación de control, y viceversa.

La fabricación y programación de controladores lógicos programables, o PLCs, ha sido comandada por los grandes fabricantes internacionales, Siemens en Europa, Allen Bradley en estados unidos, Omron en Japón, Delta en China, entre otros, desarrollan junto con sus equipos, software que se encarga de gestionar los datos y la comunicación entre estos, por

consiguiendo cada fabricante desarrolla software que es ajustado a las características de cada dispositivo, sin embargo, un sistema basado en PLCs, es un sistema cerrado, hermético, que poco contacto tiene con el mundo exterior, solo lo que el fabricante y las normativas lo permitan, en conclusión, un desarrollo con un PLC es muy ajustado en términos de escalabilidad y alcance a las limitantes de cada fabricante, comparado a los grandes costos que acarrea.

Marco teórico

Hoy en día el software libre ha llegado a lugares impensables, y, es en este punto en donde un fabricante de PLCs se propuso a instalar placas arduino en estos controladores lógicos, desde el 2006, abriendo paso a un nuevo mundo de automatizaciones industriales basadas en internet de las cosas, este es un gran aliciente para los profesionales en sistemas que tenemos afinidad o interés con el mundo de la automatización, no sólo para desarrollar nuevos sistemas automatizados bajo la filosofía del software libre, sino también para migrar los sistemas actualmente instalados y aumentar las funcionalidades ya obtenidas, con las ventajas de la programación en plataformas abiertas.

Por años, el sector industrial ha automatizado operaciones usando dispositivos PLC, en latinoamérica el fabricante líder ha sido Allen bradley, con tecnología de punta, y equipos compactos

de mediana gama, como el Micrologix 1100, pero costoso, alrededor de 4 millones por equipo, y software como el RS Logix 500 que comienza desde los 15 millones de pesos en adelante por estación, es decir, de manera local, lo que hace que la automatización de cualquier proceso signifique una mediana o gran inversión, y pensar en un sistema distribuido con varias estaciones hace que sea una inversión monumental.

Formulación del problema

En nuestra región, debido al nublado concepto cultural que se ve en la inversión en tecnología, puesto que el retorno a capital en algunos casos puede tomar décadas, son contadas las empresas que cuentan con sistemas de automatización industrial estandarizados a nivel mundial, importados e instalados por empresas como G&L Ingenieros, que llevan gran trayectoria en la instalación y acoplamiento de sistemas automatizados, pero que por consiguiente suponen grandes inversiones en capital para quien los contrate, no sólo para adquirir estos costosos equipos, también para suplir los costos de mano de obra tecnificada. El inconveniente principal a la hora de pensar en migrar sistemas completos que ya cumplen de manera óptima el objetivo que fueron diseñados, a software libre, es el costo, puesto que los directivos ya han aprobado los presupuestos, y las fábricas poseen equipos e infraestructura de alguno(s) de los fabricantes ya mencionados anteriormente y que llamaremos “cerrados”, y por esto, plantear un cambio en el cerebro (PLC) del proyecto, significaría idear un proyecto

nuevo con costos que ya fueron asumidos y ejecutados, para solucionar un problema que ya fue corregido; ningún directivo en sus 5 sentidos estaría dispuesto a re - asumir costos innecesarios.

Arduino, por qué se usó?

Costos: Un PLC basado en arduino puede estar alrededor del millón de pesos (\$1'000.000), sin embargo su programación se puede simular con la facilidad de acceder a una placa de arduino Mega, por \$40.000 y agregar una placa de conexión ethernet por \$25.000, comparar a los costos de un PLC Allen Bradley Micrologix 1100 alrededor de cuatro millones de pesos (\$4'000.000) hace que sea la opción más directa al hablar de presupuestos. No hay mucho que decir del software de programación gratuito Arduino IDE, que está disponible para ser descargado desde la red, mientras que la suma de los licenciamientos de Rockwell Ronda los treinta millones de pesos (\$30'000.000).

Programación y conexión transparente: Solo es requerida una conexión a través de un cable serial de datos desde la placa Arduino Mega, con salida a puerto USB para su alimentación de energía y programación desde un PC, se configura el software Arduino IDE con 3 clicks al seleccionar el modelo de la tarjeta, y la placa está lista para recibir órdenes. Los dispositivos de salida como diodos led o motores que funcionan a un máximo de 12 V, son muy baratos y fáciles de conseguir en los almacenes del centro de la ciudad.

¿Que es el microframework Flask y por que fue elegido?

Microframework es el término que ha sido usado para referirse a Frameworks minimalistas de aplicaciones web, en contraste con Frameworks Full-Stack. Un microframework carece de la mayoría de funcionalidades que es común esperar en un Full-Stack, (como lo es Django) tales como manejo de cuentas, autenticación, autorizaciones, roles, permisos, abstracción de Bases de datos a través de un mapeo de objetos relacionales, motores de plantillas web, entre otras.

Las ventajas de tener un microframework se hacen mayores cuando la tarea de programar se vuelve simple, al ser escalable, Flask también permite añadir módulos, servicios o librerías, puesto que por defecto no está atado a ningún motor de plantillas o bases de datos, inclusive deja el manejo de excepciones a gusto del programador.

Glosario de términos

PLC.

Para efectos prácticos, se resume que un PLC es una computadora robusta y confiable, cuyos periféricos de entrada son sensores y de salida son actuadores, diseñada para estar en ambientes hostiles en términos de señales, y transmitir datos con un nivel de confiabilidad muy alto, pueden ser programadas en el sitio de trabajo o remotamente según sea su arquitectura, se pueden encontrar de muchas marcas, tipos y especificaciones, y por lo general el software que utiliza debe

ser adquirido a la misma empresa que lo fabrica.

SCADA.

(Supervisory Control And Data Acquisition traducido a Supervisión, Control y Adquisición de Datos) es un estilo de software para ordenadores que permite controlar y supervisar procesos industriales a distancia. Facilita retroalimentación en tiempo real con los dispositivos de campo (sensores y actuadores), y controla el proceso automáticamente. Provee de toda la información que se genera en el proceso productivo (supervisión, control calidad, control de producción, almacenamiento de datos, etc.) y permite su gestión e intervención.

Interfaz.

Para la definición de una interface, el portal sistemas.com (Anonimo) usa las siguientes palabras: “Definimos genéricamente en el mundo de la electrónica a la Interfaz como todo puerto que nos permite enviar y recibir señales desde un componente a otro, teniendo entonces distintas formas de realizar este envío dispuestas por las Especificaciones Técnicas de cada equipo, o bien mediante el establecimiento de distintos estándares que permiten la comunicación.

Automatización industrial.

La automatización industrial, que viene del griego antiguo auto, ‘guiado por uno mismo’ es el uso de sistemas o elementos computarizados y electromecánicos para fines industriales. va mucho más allá que un mero sistema de control, la instrumentación industrial abarca los

sensores, los transmisores de campo, los sistemas de control y supervisión, los sistemas de transmisión y recolección de datos y las aplicaciones de software en tiempo real para supervisar y controlar las operaciones de plantas o procesos industriales.

Pirámide de la automatización.

Es un esquema que representa las capas básicas que relacionan todos los agentes que actúan en un proceso de automatización industrial, puede relacionarse con el modelo de la capa OSI, ya que en la base se encuentran los dispositivos físicos, y a medida que se escala en la pirámide, se encuentran los drivers, las redes, las interfaces y las herramientas que permiten la visualización, monitoreo, control y gestión de los datos e información. La pirámide de automatización planteada por la empresa mexicana de automatización SSIGSA DISEÑO, (2016) mostrada en Figura 1, Gráfico de la pirámide de automatización básica de 4 niveles da una visión más gráfica.

Figura 1
Gráfico de la pirámide de automatización básica de 4 niveles



Fuente: <http://ssigsa.com/images/piramide.png>

(Consulta 20 de noviembre 2018)

Tomado de: SUMINISTROS, SERVICIOS E INGENIERÍA INTEGRAL DEL GOLFO S.A. DE C.V.

Desarrollo de la metodología

Es imprescindible saber cómo dar y recibir órdenes del PLC basado en arduino, y también conocer sobre los protocolos que se utilizan actualmente como interfaces de comunicación compatibles con el dispositivo cerrado AB, a fin de permitir que las entradas de este. que es el dispositivo que se encuentra instalado en las fábricas, sean las salidas del PLC arduino, y a su vez las salidas de AB. sean entradas para el PLC basado en placas de software libre.

Con la información recolectada sobre el funcionamiento de los dos dispositivos, y las tecnologías actuales en cuanto a los protocolos compatibles de interconexión, se sientan las bases para una 2a etapa, la cual encierra los conocimientos adquiridos y plantea un sistema en el cual los datos que son recolectados en campo, por los periféricos conectados al PLC AB. pueden ser procesados, guardados, organizados y mostrados a través de Arduino, con una interfaz desarrollada en software libre.

Etapa 1: Investigación y captación de información.

En esta etapa se realiza la disposición y recolección de la información necesaria para el planteamiento de este proyecto, que

abarca desde la conceptualización de herramientas y arquitecturas de actualidad para la creación de interfaces multiplataforma, hasta el reconocimiento de los ítems necesarios para el entendimiento tanto del PLC Allen Bradley como del M-Duino. Se divide en 3 actividades principales, la cuales se describen detalladamente a continuación:

Actividad 1, Investigar documentación relacionada a la creación de una API, SOA e interfaz web:

Al comenzar el proyecto no había mucha claridad a la hora de la etiqueta que se le daría al tipo de arquitectura del software que se diseña para encargarse de permitir una conexión multiplataforma, en este caso entre RSLogix 500 y Arduino IDE, sin embargo con el paso del tiempo, las horas de estudio y recolección de datos en el curso, fue posible descubrir otro estándar de comunicación que ha sido desarrollado en el campo del control y supervisión de procesos industriales, llamado OPC (OLE for Process Control) Según la fundación OPC (2018), OPC es un protocolo basado en una tecnología Microsoft, que ofrece una interfaz común para comunicación la cual a su vez permite que componentes de software individuales interactúen y compartan datos, permitiendo que un dispositivo de hardware del nivel de planta de producción actúe como servidor de cualquier aplicación que se programe para leer/escribir cualquier variable que este ofrezca. OPC está basado en OLE, software de tipo cliente pesado de Microsoft que permite crear un archivo principal, al cual se puede hacer un referencia de datos donde todo

cambio posterior en el archivo principal se refleja en el documento donde se hizo referencia. A pesar de que los grandes fabricantes como Allen Bradley han incluido OPC en sus productos, y que hace posible la extracción de datos desde el dispositivo Micrologix 1100 a una hoja en excel como lo veremos más adelante, se hace necesaria la creación de un servidor programado como OPC que sea externo a la Suite de Rockwell Automation.

OPC plantea la interconexión de todos los dispositivos a través del protocolo ethernet, por lo que se plantea la creación de un servidor local en el micro framework de programación para python flask, el cual ha tomado gran fuerza en los últimos tiempos para la creación de API en python, ya ha sido trabajado y probado en conjunto con Arduino en un proyecto para la materia cliente servidor y es un excelente recurso a la hora de proyectos que tienen propuesto un servidor de características ágiles y livianas.

Actividad 2, Investigar y conocer el funcionamiento interno de un dispositivo PLC Allen Bradley y de un PLC basado en arduino.

PLC Allen Bradley.

El micrologix 1100 ha sido diseñado para monitorización remota y para aplicaciones que requieran bastante disponibilidad de memoria, pero requiere pocos puertos de entradas y salidas. Posee un socket Ethernet/IP de 10/100 Mbps para mensajería P2P (Peer to peer) y cobertura de conectividad para toda la familia de los productos PLC de la familia Allen Bradley

bajo las limitantes del software producido por Rockwell Automation.

El punto de quiebre está en los costos de adquisición tanto del hardware, que oscila alrededor de US\$1,300, mil trescientos dólares, alrededor de cuatro millones de pesos (\$4'000.000), como del software que se requiere para el aprovechamiento de esta herramienta, es toda una suite de utilidades, RsLogix 500, RSLinx RSEmulate, RsView, los cuales con todas las características y funcionalidades pueden tener un costo de Alrededor treinta y ocho millones quinientos mil pesos (\$38'000.000).

PLC Industrial Shields.

En este caso se toma la referencia del M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS, el cual se puede apreciar seguido a este párrafo en Figura 5 Imagen de M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS. Está especialmente diseñado para su uso en un entorno profesional y cuenta con 13 entradas y 8 salidas (dispositivo de similares características al AB. que cuenta con 10 entradas y 6 salidas), de las cuales 7 entradas y 3 salidas son digitales configurables por software, cuenta con puertos de conexión para comunicación tipo Ethernet, USB tipo B, puerto I2C que ofrece la posibilidad de expandirse con 127 módulos y gobernar hasta 6604 E/S en modo maestro esclavo, además de módulos adicionales de sensores, o de expansión de la misma marca. También cuenta con dos puertos seriales, un RS-232 y un puerto HALF/FULL Duplex RS-485. Soporta protocolos TCP/IP, Modbus TCP y Modbus RTU y tiene la capacidad de

comportarse como un Web server, lo que le ofrece una gran flexibilidad y control.

Actividad 3, Analizar cómo cada dispositivo PLC ejerce control sobre el sistema.

PLC Allen Bradley.

Para tomar control del sistema, después de que todas las conexiones físicas han sido instaladas, el PLC Allen Bradley debe ser configurado en la plataforma RSLinx, la cual permite la comunicación entre los dispositivos de control a través del protocolo Ethernet.

Luego de que se selecciona la comunicación entre la tarjeta del controlador del cual se van a extraer los datos el PLC y el computador o la pantalla táctil con la que se va a descargar el programa, se procede a descargar el programa lógico en lenguaje ladder que ha sido programado en la plataforma RSLogix 500. El PLC recibe las órdenes a ejecutar en este programa a través del cambio de modo de Offline a Descarga, y una vez se vuelva a cambiar el modo a Online de inmediato comienza a ejecutar los datos de las órdenes de control que se indique en el código de lógica de contactos Ladder, sobre los actuadores del sistema, y recoger los datos que le entregan los sensores para su posterior análisis en el SCADA. En este punto es opcional el uso del software RSEmulate, debido a que se puede o no simular el proceso con el PLC, ya depende de la experticia y la planeación de quien ejecuta el proyecto.

Después de que el sistema ya esta probado y funcional, el control pasa al SCADA que se haya elegido, en la suite de Rockwell se

ofrece RSView, sin embargo en este punto cualquier SCADA que tenga compatibilidad con este equipo puede tomar el mando. El SCADA se encarga de las ejecuciones de órdenes bien sea para la activación de máquinas, para el funcionamiento de una electroválvula, etc. Esto depende la naturaleza del proyecto. Paralelo a ejecutar las órdenes, el dispositivo se encuentra recibiendo datos en tiempo real de los sensores, si el proyecto cuenta con ellos, (por ejemplo un flujómetro que se encarga de la medición de flujo de determinado líquido que pasa por una área en determinado tiempo) el SCADA también según su programación estará almacenando información en determinada Base de Datos, para luego proceder a analizar los datos y generar informes gerenciales, de producción, administrativos, etc.

PLC Industrial Shields.

La primer gran diferencia que nos topamos al trabajar con estos dispositivos, en relación a los tradicionales, es que su lenguaje de programación es Arduino IDE, cuya adquisición es gratuita y tiene una estructura bastante simple, que divide la ejecución en dos bloques principales, `setup()` y `loop()`, la primer función que se ejecuta en el programa y solo una vez, es `setup()` donde se definen las variables `pinmode` (qué pines digitales son de entrada o salida) y se inicia la comunicación serial. `Loop()` contiene todo el código que será ejecutado en cascada, donde se configurará el comportamiento de las entradas y las salidas

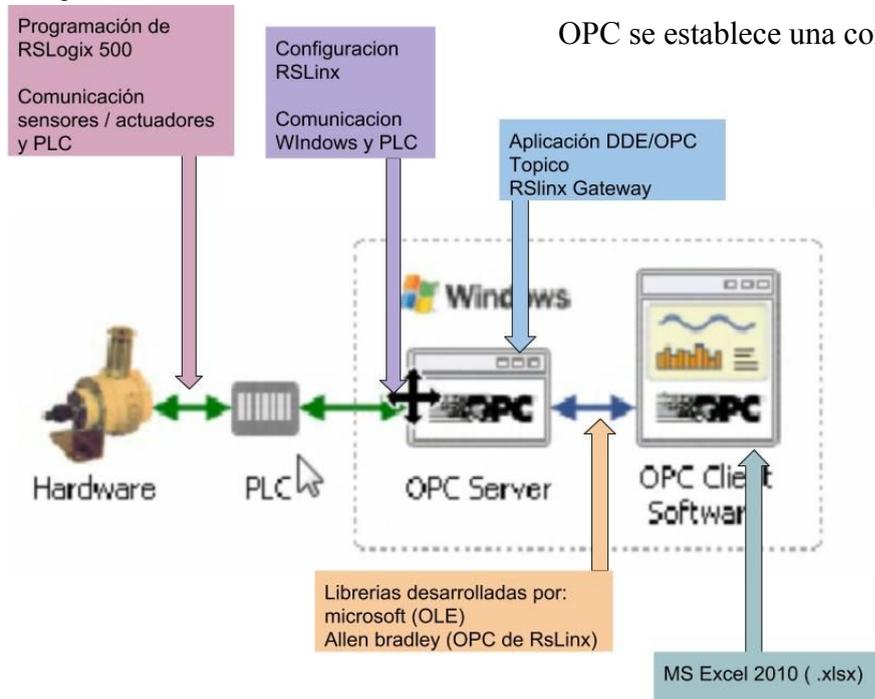
La segunda gran diferencia tiene que ver con el uso de los protocolos de comunicación de manera libre para el intercambio de datos físicos entre diferentes PLC, los cuales son: el protocolo I2C, protocolo propiedad de arduino el cual funciona sólo bajo sus dispositivos y la arquitectura maestro - esclavo, protocolo Serial, el cual es el más usado en la búsqueda de estabilidad, confiabilidad y seguridad en los datos sacrificando un poco la velocidad y rapidez entre ellos y, solo entre dispositivos M-Duinos, protocolo Ethernet.

Etapa 2: Diseño y divulgación.

Actividad 1, Diseñar una interfaz que permita administrar PLC Allen Bradley desde un PLC basado en arduino

Esta actividad fue planeada teniendo en cuenta los conceptos básicos de análisis y desarrollo de una interfaz, sin embargo con el desarrollo de la investigación, se descubrió el estándar OPC el cual surgió en 1995 con la unión de los grandes fabricantes de hardware y software industriales, que junto con Microsoft desarrollaron la primera versión de lo que sería el estándar mundial de protocolos de comunicación e interoperabilidad entre dispositivos de automatismos industriales. Es por esta razón que se decide planear la interfaz en base a esta arquitectura, la cual tiene su primer bosquejo graficado seguido en Figura 2 Bosquejo del sistema en base a un gráfico de arquitectura OPC.

Figura 2
Bosquejo del sistema en base a un gráfico de arquitectura OPC



Fuente:

https://drive.google.com/open?id=1mr3pUbPC7998LHjRuIBAXPeIAP69Hmny_XvYYuwPqv8

Tarea 1, Levantamiento de requerimientos:

La interfaz a desarrollar debe tener un módulo Acceso_Datos_Sincronos que permita leer, escribir y monitorizar variables que contienen datos de proceso actuales en el PLC Allen Bradley Micrologix 1100. Su principal uso será el transmitir datos de tiempo-real entre PLC AB. Y PLC M-Duino 21 I/Os plus, y otro dispositivos de control en un PC pantalla cliente. Se debe poder seleccionar

explícitamente las variables o tópicos (elementos OPC) que quiera leer, escribir o monitorizar en el servidor desde el PLC AB, bien sea a través de la interfaz o usando el Anexo D, Configurar un tópico OPC/DDE en RsLinx para obtener datos de un controlador Logix 500 y agregarlo a una hoja de cálculo de excel, al final de este documento. A través de la interfaz OPC se establece una conexión al servidor

M-Duino creando un objeto OPCServer, donde los datos de salida del Allen Bradley son las entradas de la interfaz, y contiene los métodos navegar_direcciones y buscar_dato OPC y sus propiedades, como el tipo de datos, o los permisos de acceso, los cuales se guardaran en un diccionario de datos .

Se debe poder escoger entre 2 modos, el primero monitorizar los cambios de los valores por el M-Duino, a fin de que si este detecta un cambio que supere límites establecidos, reaccione con cierto tipo de alerta o mensaje y gestione este problema ofreciendo el instante de muestreo

(timestamp) y la calidad a los datos enviados. La calidad especifica si el dato es correcto (bueno), no disponible (malo) o desconocido (dudoso).

El segundo modo, una visualización en pantalla cíclica de los datos en busca de cambios para la supervisión en tiempo real, con una demora de actualización que no debe ser mayor a 2 segundos. Este tiempo de actualización es utilizado en el servidor para comprobar los valores cíclicamente en busca de cambios. A cada ciclo, el servidor envía sólo los valores cambiados al usuario.

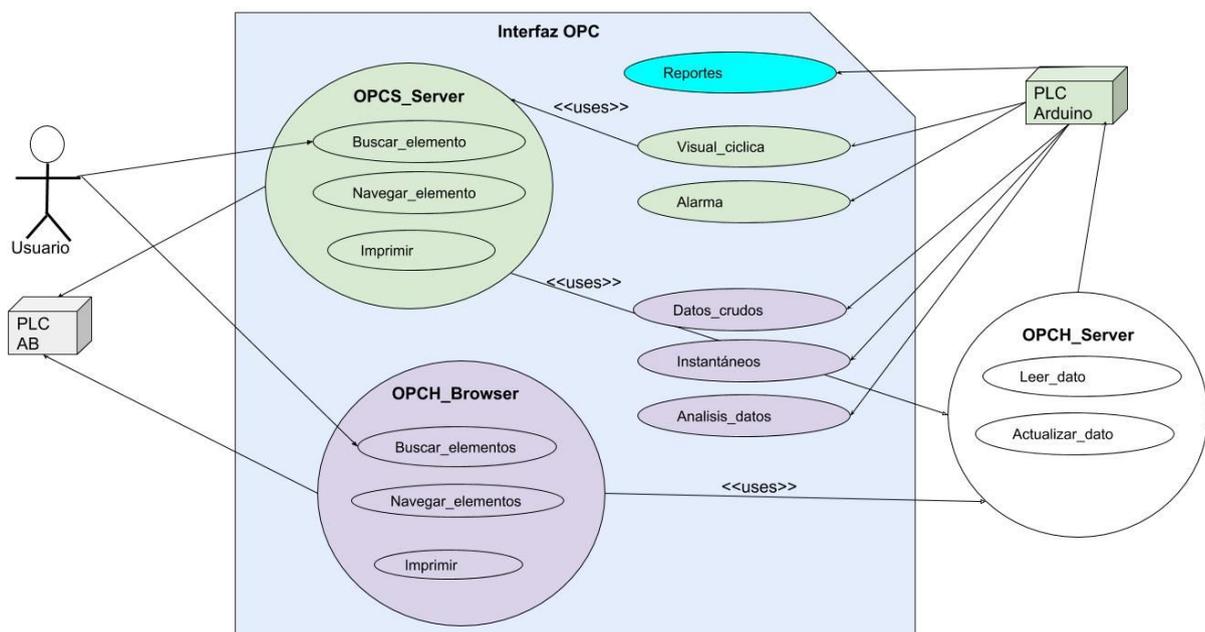
Otro módulo que se propone es Acceso_Datos_Historicos, que debe permitir el acceso a datos ya almacenados. Los archivos históricos se deben recuperan de manera uniforme, desde el sistema de registros simple del Micrologix, hacia la interfaz OPC, como se puede apreciar al final del documento en Anexo D.

Tarea 2, Elaboración de casos de uso

La elaboración de los casos de uso gráficamente apreciada a través de Figura 3

Diagrama UML de los casos de uso, se hace de acuerdo al levantamiento de requerimientos hecho en la tarea anterior y se ve a continuación.

Figura 3
Diagrama UML de los casos de uso



Fuente:

<https://drive.google.com/open?id=1Soi45kSRXP0P9KLJWjysPHs231Gzm85Zhv8zNmQTA9A>

Autor

Tarea 3, Desarrollo de especificación

Para la especificación del sistema se plantea Tabla 1 Documentación de los actores que interactúan en los casos de uso, donde se aprecian los actores que interactúan más adelante en Tabla 2 Lista de drivers arquitectónicos.

Tabla 1
Documentación de los actores que interactúan en los casos de uso

Actor	Caso de uso en los que participa	Tipo	Descripción
Usuario	2	Primario	Cliente que manipula el sistema
PLC Allen Bradley Micrologix 1100	2	Secundario	PLC que funciona como esclavo y se encarga de recoger los datos de campo
PLC M.Duino 21 I/O	7	Secundario	PLC que funciona como maestro y recoge los datos para administrarlos, analizarlos y mostrarlos

Tabla 2
Lista de drivers arquitectónicos

Tipo de driver	Id	Descripción del driver	Prioridad
Atributo de Calidad Disponibilidad	AC1	El sistema debe estar disponible un 98% del tiempo de uso	5
Caso de Uso	CU1	El sistema debe permitir buscar una variable con datos actuales almacenada en el PLC Micrologix	5
Caso de Uso	CU2	El sistema debe permitir la navegación hacia el elemento buscado en Micrologix a través de las herramientas OPC	5
Caso de Uso	CU3	El sistema debe permitir la impresión por pantalla de los datos seleccionados	5
Caso de Uso	CU4	El servidor PLC M-duino debe leer la información que se trae desde Micrologix	5
Caso de Uso	CU5	El sistema debe poder analizar datos históricos recibidos y entregar estadísticas	4
Caso de Uso	CU6	El sistema debe poder mostrar una búsqueda instantánea de datos históricos en determinado periodo de tiempo	3

Caso de Uso	CU7	El sistema debe permitir la visualización gráfica de datos cíclicos actualizados cada 2 segundos	3
Caso de Uso	CU8	El sistema debe permitir la configuración de alarmas bajo límites establecidos	3
Restricción	R1	El sistema no debe permitir la búsqueda de dos tipos de dato a la vez	4
Atributo de calidad Desempeño	AC2	El sistema no debe tardar más de 1 minuto en hacer una búsqueda de un conjunto de datos históricos	3
Atributo de calidad Desempeño	AC3	El sistema no debe tomar más de 2 segundos para actualizar los datos en tiempo real	3
Atributo de calidad Usabilidad	AC4	El sistema debe ser de facil manipulacion para el usuario final	5
Restricción	R2	La interfaz debe estar programada en python 3, usando la librería OPENPYXL para la comunicación con Excel, y SOCKET para la comunicación con Arduino	4

Conclusiones

Finalizado el análisis y el diseño de la interfaz para la interconexión y transmisión de datos entre diferentes tecnologías de PLC, lo primero que se trae a colación es el planteamiento de la interfaz, puesto que al principio de este proyecto, se tenían ideas muy nubladas sobre las tecnologías que usa la industria de la automatización. Uno de los retos más interesantes fue la formulación del proyecto, debido a que se tenía desconocimiento sobre el tipo de interfaz que debía usarse, API, SOA, y otros modelos fueron ideados, y tratados de llevar a la conceptualización, sin embargo, fue solo hasta el desarrollo completo del proyecto que se pudo encontrar el estilo de arquitectura que se ha usado en esta misma industria hace unos años, OPC, que es ahora un estándar mundial para todos los grandes fabricantes de equipos de automatización y que tiene bases muy sólidas desde 1995, es lo que ha permitido la interoperabilidad entre equipos y software de diferentes fabricantes.

El segundo factor a discutir tiene que ver con los límites y el alcance que tuvo el proyecto en su formulación, si bien este proyecto nace de un problema real que se presentó en una empresa, y su componente principal PLC Micrologix 1100 fue el mismo que se pensó para este proyecto, se dificulta demasiado el trabajo en base a este equipo, pues su acceso es costoso, y para el tiempo que el proyecto estaba

tomando forma, se habían roto las relaciones comerciales y ya no se posee acceso al equipo ni al software que se necesita para su programación y funcionamiento, aunque este inconveniente tuvo solución por medio de una versión demo por 7 Días, se pudo constatar lo cerrada que es la arquitectura de la empresa Rockwell Automation, y los altos costos que puede generar el acceso completo de las herramientas que permiten utilizar un recurso como este, así como la complicada labor de cotizar los costos. El distribuidor oficial, Melexa, ubicado en el sector industrial la popa, nos brindó la información de costes que se puede apreciar en el desarrollo de la metodología, sin embargo, fue tras una espera de alrededor mes y medio que fue posible tener una reunión con el encargado para el Eje cafetero, debido a la complejidad que exige nuestro proyecto en torno a la programación y el conocimiento del equipo. Esto también da una visión de la espera que puede generar plantear un proyecto con una tecnología como la de Allen Bradley.

Otro hecho que ha generado un contraste ha sido la elección del protocolo a utilizar para la comunicación de los equipos, si bien Ethernet ha sido un protocolo que ha tomado mucha fuerza en los últimos años para el desarrollo de redes industriales de velocidad mas rapida para la transmisión de datos, y esto ha tocado la puerta del mundo de la automatización, con nuevos equipos que incorporan puertos Ethernet, se desconocía totalmente el estado del arte de este sector de la industrial, y a través de

esta investigación se ha podido constatar que a pesar del avance de nuevos protocolos, los protocolos Serial son aún hoy en día los más usados para la transmisión estable de datos entre equipos o entre PLCs y sensores y actuadores. RS232 y RS485 son aun protocolos vigentes en redes de automatización industrial, pues garantizan la fiabilidad y estabilidad de los datos sin importar los ruidosos o toscos ambientes donde se encuentren instalados, y son los más usados cuando lo que se busca es poca caída de paquetes en la transmisión de datos críticos, como puede serlo cualquier proceso que involucre recursos de una empresa. Es por esto que se encontró mucha documentación sobre la comunicación entre Arduino y el software de Rockwell, sin embargo las librerías que se ofrecían solamente establecen comunicación a través de protocolos seriales, y este es el motivo por el cual tuvo que ser añadida una herramienta como Excel, ya que por la colaboración de Microsoft con la industria de la automatización, el protocolo OLE permite la interacción entre productos Microsoft y Allen Bradley. Se pudo constatar lo poderosa que es esta herramienta de ofimática, sin embargo hace que el planteamiento de este proyecto resulte en muchas rutas adicionales para los datos, lo que puede retrasar bastante el tiempo de respuesta y reacción del sistema, en comparación con una arquitectura con conexión por protocolos seriales.

Como se puede observar en los anexos, Industrial Shields es solo una más de las empresas que ha incursionado en el mundo del desarrollo libre para elementos de

automatización de procesos industriales, ya que es un nuevo mercado emergente en el que se han sumado varios fabricantes, como lo son Controllino, Kunbus, entre otras. La escogencia de esta empresa fue basada en la tecnología de las placas bases que usan, puesto que por conocimiento y experimentación propia para proyectos de otras materias y que fueron incluidos en esta investigación, se posee una experiencia previa y un conocimiento sobre el funcionamiento y la programación de Arduino más allá de las placas basadas en Raspberrypi de la industria alemana. Esto no quiere decir que Industrial Shields sea la compañía que tenga que usarse al pensar en un proyecto como estos, porque como se pudo descubrir a medida que este proyecto avanzó, trabajar con una placa Raspberry puede tener más ventajas a la hora de buscar interoperabilidad entre los sistemas, al ser una placa que funciona bajo sistemas operativos Linux y que tiene más capacidad de procesamiento, y son muchas las opciones que ambas empresas ofrecen en cuanto a productos y servicios. Lo que diferencia grandemente a estas dos empresas es el modelo de arquitectura para la construcción de los módulos PLC, Industrial Shields ofrece sus dispositivos compactos, listos para ser programados y conectados a una fuente de energía eléctrica, por otro lado Kunbus plantea soluciones con equipos modulares de 3 partes, Módulo Central, que controla las operaciones, un módulo de entradas y salidas, que es la expansión que permite conectar IOs, y modulo fuente de poder que entrega la energía necesaria para el funcionamiento del sistema de control. Estos 3 son los elementos de hardware que se necesitan para crear un sistema que se

programe y conecte a los sensores. Las connotaciones que trae esta diferencia son dos, en relación a capacidad y precio. Por un lado, como es bien sabido por las personas que integran sistemas de control, los PLC cuya arquitectura es modular son equipos que por lo general superan por mucho en capacidad de procesamiento de procesos y entradas salidas, a dispositivos compactos. Esto resulta en la segunda connotación que tiene que ver con que un sistema con PLC modulares resulta más costoso que trabajar con Compactos, por lo que su escogencia debería depender del tamaño del proyecto que se quiere abordar.

Otro punto de discusión para este sistema es el hecho del planteamiento inicial utilizando otro dispositivo PLC Arduino, pues se pudo constatar que sería innecesario agregar otro PLC Arduino para dar órdenes a un sistema básico que solo cuente con 1 PLC Micrologix, ya que con el desarrollo de una interfaz que integre los elementos vistos anteriormente, como lo son la conexión OPC a un sistema en python a través de excel, es suficiente el poder de procesamiento que se necesita para generar informes, alarmas y órdenes. El agregamiento uso de un dispositivo arduino de control sería imperante en el momento que se desee adecuar un sistema que cuente con varios PLC conectados en red. En este caso sí traería muchas ventajas la implementación de un equipo que se dedique solo a dar órdenes y administrar los datos recibidos, basado en filosofía de software libre. Este supuesto se hace después de constatar lo poderoso que es el equipo Micrologix 1100, en cuanto a capacidad de procesamiento.

Junto a la conclusión anterior se tiene la nueva concepción de que a la hora de proponer nuevos proyectos, se busque seleccionar solo equipos de programación libre, sin la necesidad de agregar dispositivos que se dediquen solo al control de las entradas y salidas y otro al control del sistema completo, pues los dispositivos industriales de arduino cumplen con toda la normativa requerida en términos de seguridad y rendimiento Europeo, y que una adecuación como la vista en todo el desarrollo del procedimiento sería óptima en el momento que solo se quiera adecuar un sistema que ya está funcional, explícitamente como el objetivo general de este proyecto lo indica.

Referencias

- Allen, Brett. (2016). Video de Youtube: Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel. Disponible en <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s>
- ARDUINO. (s.f). ¿ Qué es arduino?. Recuperado de <https://arduino.cl/que-es-arduino/>
- Automation Networks. (s.f). RSLinx Classic and enterprise software, Recuperado de: <http://automation-networks.es/glossary/rslnx-classic-and-enterprise-software>
- CTIN. Centro de Tecnología e Innovación. (2013). Que es un PLC ?. Ciudad de Mexico, recuperado de <http://www.ctinmx.com/que-es-un-plc/>

- GL Ingenieros. (s.f). Automatización y Calidad de la Potencia. Recuperado de <http://www.glingenieros.com.co/calidad-de-energia/>
- Industrial Shields. (s.f). Quienes somos, Controlador PLC Industrial basado en Arduino, de Industrial Shields. [Consulta 30 agosto 2018], Recuperado de: https://www.industrialshields.com/es_ES/about-us
- Industrial Shields. (2016). Manual de Familia M-Duino (Págs 10-16). [Consulta 28 de Octubre 2018]. Recuperado de: https://www.plcarduino.com/wp-content/uploads/2017/05/23.03.16_M-duino-Family-PLC.pdf
- Ing. Cesar Augusto. (Usuario Steemit: @autinf). (2018). La pirámide de automatización. Recuperado de: <https://steemit.com/spanish/@autinf/la-piramide-de-automatizacion>
- International Standard. (2007). IEC 61131-2 Tercera edición, Programmable controllers – Part 2: Equipment requirements and tests. Recuperado de https://webstore.iec.ch/p-preview/info_iec61131-2%7Bed3.0%7Den.pdf
- KUNBUS. (s.f). Meet the revolution PI products - Industrial Raspberry PI. (Consulta 20 de octubre 2018). Recuperado de <https://revolution.kunbus.com/revolution-pi-series/>
- KUNBUS. (s.f). Meet the revolution RevPi Connect base module - Industrial Raspberry PI, products. Consulta 20 de octubre 2018. Recuperado de <https://revolution.kunbus.com/revpi-connect/>
- Marsellach, Francesc. FMJ Ingenieros (2012). Introducción a la automatización. Recuperado de <https://www.fmjingenieros.com/servicios/automatizacion-industrial/introduccion-a-la-automatizacion>
- Open Source Organization. (s.f). The Open Source Definition | Open Source Initiative. Recuperado de <http://opensource.org>
- Openmote. (s.f) Open Hardware for the industry of internet of things. [Consulta 18 noviembre 2018], Recuperado de <http://www.openmote.com/>
- OPC Foundation. (2018). What is OPC?. Recuperado de: <https://opcfoundation.org/about/what-is-opc/>
- Prieto, Albert. Industrial Shields. (2016) Controlador PLC Industrial basado en Arduino, de Industrial Shield, Presentación modelo propio industria 4.0. Recuperado de <http://www.automaticeinstrumentacion.com/es/downloads2/industria-l-shields-albert-prieto.compressed.pdf>
- Rockwell Automation. (s.f). RS Logix 500 Solutions from Rockwell Software, Recuperado de <http://www.rockwellautomation.com/rockwellsoftware/products/rslogix500.page>
- Rockwell Automation. (s.f). Sistemas de controlador logico programable Micrologix 1100, Recuperado de

<http://ab.rockwellautomation.com/es/Programmable-Controllers/MicroLogix-1100#overview>

Rockwell Automation. Supersedes Publication (2005). ML 1100 series B Profile 1763-PP001A-EN-P – July 2005 En Publication 1763-PP001B-EN-P – 2007— (Eds.) Disponible en https://literature.rockwellautomation.com/idc/groups/literature/documents/pp/1763-pp001_-en-p.pdf

SSIGSA. (s.f). Automatización y control: Integrando y desarrollando tecnología. Recuperado de <http://ssigsa.com/auto.php#>

Valencia H., Jose V. (Autor). (2018). Diseño y aplicación de un sistema bajo arquitectura cliente servidor para la activación de riego automatizado en Arduino a través de Twitter, página 5 y 6, Recuperado de https://www.researchgate.net/publication/328174883_Diseño_y_aplicación_de_un_sistema_bajo_arquitectura_cliente_servidor_para_la_activación_de_riego_automatizado_en_Arduino_a_traves_de_Twitter