Congr. Int. en Ing. Electrónica. Mem. ELECTRO, Vol. 39, pp. 188-193, Oct 2017, Chihuahua, Chih. México
http://electro.itchihuahua.edu.mx/memorias_electro/MemoriaElectro2017.zip
ISSN 1405-2172

AUTOMATIZACIÓN DE CELDA DE MANUFACTURA UTILIZANDO LA METODOLOGIA DE GRAFCET Y GUIA GEMMA IMPLEMENTADA EN LABVIEW

Rodriguez Jose L, Yuridia Vega, Villegas Jose M., Rosel Manuel J., Pimentel Alex B.
Universidad Autónoma de Baja California
Escuela de Ciencias de la Ingeniería y Tecnología Dirección
Tel. (664) 676-8222
jlrodriguez@uabc.edu.mx, vegay@uabc.edu.mx, villegas_josemanuel@uabc.edu.mx, mrosel@uabc.edu.mx, alex.pimentel@uabc.edu.mx

RESUMEN.

Se presenta el uso de la metodología de Grafcet y la guía Gemma implementada en Labview para automatizar una estación de una celda de manufactura didáctica de llenado, tapado y rechazo de un producto. Los softwares existentes para programar controladores lógicos programables PLCs, ya cuentan con los modos de programación para Grafcet pero solamente se encuentran disponibles para ciertos modelos y el costo de las licencias es elevado. De la misma manera las interfaces de usuarios cuentan con sistemas operativos y herramientas que facilitan la programación de las aplicaciones comunes utilizadas en la automatización. En este trabajo se presenta la implementación del Grafcet utilizando el diagrama escalera, y el desarrollo de la interfaz gráfica utilizando la arquitectura de productor consumidor de Labview la cual permite el uso de otras herramientas como la conectividad a bases de datos para lograr la comunicación vertical con los diferentes niveles de la pirámide de automatización.

Palabras Clave: Grafcet, GEMMA, PLC, LabView

ABSTRACT.

We present the use of Grafcet methodology and the Gemma guide implemented in Labview to automate a station in a manufacturing cell of filling, capping and rejection of a product. Existing software to program PLCs programmable logic controllers already have programming modes for Grafcet but are only available for certain models and the cost of licenses is high. In the same way the user interfaces have the operating systems and the tools that facilitate the programming of the applications used in the automation. This work presents the implementation of Grafcet using the ladder diagram and the development of the graphical interface using the Labview producer consumer architecture which allows the use of other tools such as database connectivity to achieve vertical communication with the different levels of the automation pyramid.

Keywords: Grafcet, GEMMA, PLC, LabView

1. INTRODUCCIÓN

Debido a la alta competencia en el mercado y el progreso en las Tecnologías de la Información (TI), muchas compañías han implementado sistema ERP (Enterprise Resource Planning) y sistemas MES (Manufacturing Execution System) para aumentar su competitividad [1].

MES es el sistema el cual los sistemas de planeación del nivel superior y los sistemas control de bajo nivel en las industrias manufactureras puedan comunicarse sin ningún problema [2].
El software MES tradicional causa grandes dificultades en su desarrollo, largos periodos de desarrollo, altos costos, baja fiabilidad así como una mala capacidad de reconstrucción e integración.
En el desarrollo de software tradicional para un sistema MES existen los siguientes problemas:

1. En la fase de análisis de requerimientos, hay demasiadas personas involucradas, tales como expertos, consumidores, analizadores del sistema y desarrolladores que tienen diferentes conocimientos y diferentes grados de conocimiento sobre el sistema. Esta diversidad provoca una mala comunicación entre las personas con diferentes niveles de entendimiento del sistema causando diferentes demandas.
2. El modelo del sistema derivado del análisis del sistema, es deficiente en la estandarización, concepción y formalización, el cual es difícil de compartir, reusar y reducir la supervisión en el diseño del software.
3. El software desarrollado en módulos forman sistemas con alto acoplamiento, ocasionando dificultad para reconstruir, reusar, expansión e integración, el desarrollo de software toma largos periodos de tiempo, difícil y costoso.

Por tal motivo la estandarización de las metodologías y estructuras de programación cobran importancia. Hoy en día se ha logrado la estandarización en las comunicaciones industriales entre los diferentes procesos donde los Sistemas MES se interconectan a través de OPC (OLE para el control de procesos) siendo un estándar aceptado universalmente que ofrece la habilidad de intercambio de datos entre los diferentes sistemas de automatización industrial en los diferentes niveles de la pirámide de automatización en la manufactura y procesos industriales [3].

En cuanto a la programación de controladores se cuenta con diferentes lenguajes de programación como el tradicional lenguaje de escalera, diagrama de bloques de funciones, lista de instrucciones, texto estructurado y SFC (Sequential Funtion Chart). Estas opciones de programación permiten que diferentes usuarios con diferentes disciplinas puedan hacer uso de los controladores que existen en el mercado pero esta misma variedad de lenguajes de programación se convierte en uno de los problemas descritos en desarrollo de software para la conectividad y aportación de datos para un sistema MES.

En este trabajo se presenta la metodología de automatización con Grafcet para automatizar una celda de manufactura didáctica de llenado tapado y rechazo de un producto y utilizando la guía GEMMA se construye una interfaz de usuario utilizando la estructura de productor consumidor basada en la plataforma de programación de Labview esto permitirá la escalabilidad, y la implementación de los módulos de comunicación para bases de datos que permitan compartir datos en línea para una posterior integración de la celda de manufactura a un Sistema MES.

2. CUERPO DEL ARTÍCULO

2.1. Desarrollo.

Para llevar a cabo el trabajo se utilizó la metodología de Grafcet descrita por Bacell en [4], en donde se identificaron las operaciones complejas del proceso como Indexado, Llenado, Pick & Place y Rechazo, en cada una de estas operaciones más complejas se determinaron las operaciones elementales que las representan, procediendo a dibujar el diagrama del Grafcet utilizando el editor SFCedit. Estos diagramas fueron implementados en diagrama escalera utilizando enclavamientos y Latch y Unlatch, organizándose cada uno de ellos en escaleras del programa a descargar en el PLC.

2.2. Grafcet y guía GEMMA.

Es un método de descripción de procesos independiente de la tecnología y del lenguaje de programación. Una vez obtenido el Grafcet este puede ser implementado en un sistema cableado o un sistema programable, al elegir el sistema programable este puede llevarse a cabo en diferentes lenguajes de programación que van desde los basados en texto hasta los lenguajes graficos. En 1977 se creó la guía GEMMA (guía de Marchas y Paradas) para el apoyo del Grafcet convirtiéndola en una herramienta de diseño y no solamente descriptiva. En 1988 fue estandarizado con la IEC 848, conociéndose en las plataformas de programación de controladores como SFC (Sequential Funtion Chart) [7].

El proceso a automatizar es una celda de manufactura que cuenta con un sistema de indexado que transporta botes a las estaciones de llenado, tapado y rechazo.

El sistema de indexado es operado por dos pistones lineales con sus respectivos finales de carrera identificándose a estos como A y C, se realiza el Grafcet de este proceso siguiendo las normas de Macroetapas se obtiene el grafico de la Figura 1.

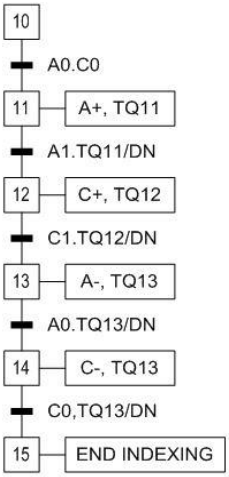

Figura 1. Grafcet de Macroetapa del indexado.

En la estación de llenado se cuenta con una tolva que almacena el producto que será dispensado en los botes a través de un pistón que abrirá la compuerta y permitirá el paso del producto por un tiempo, este será especificado por el usuario al elegir la opción de 100 grms o 200 grms. Al realizar el grafcet de este proceso se obtiene el grafico de la Figura 2.

Al finalizar el sistema de indexado en estado 15, utilizando un sensor óptico se detecta la presencia del bote en la posición uno del sistema de indexado, utilizando la función BSL (Bit Shift Left) se introduce el valor del sensor en el registro de corrimiento, de esta manera se conocerá si existe presencia de bote en cada una de las estaciones al finalizar el ciclo de indexado.

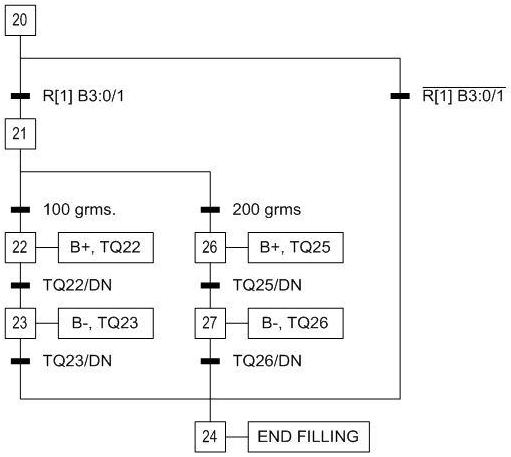

Figura 2. Grafcet de Macroetapa de Estación de llenado.

En la estación de tapado se realiza el pick&place de la tapadera del bote a través de una ventosa, en este proceso no se cuentan con los inicios y finales de carrera de los actuadores optándose por el uso de temporizadores. Al realizar el Grafcet de este proceso se obtiene el grafico de la Figura 3.

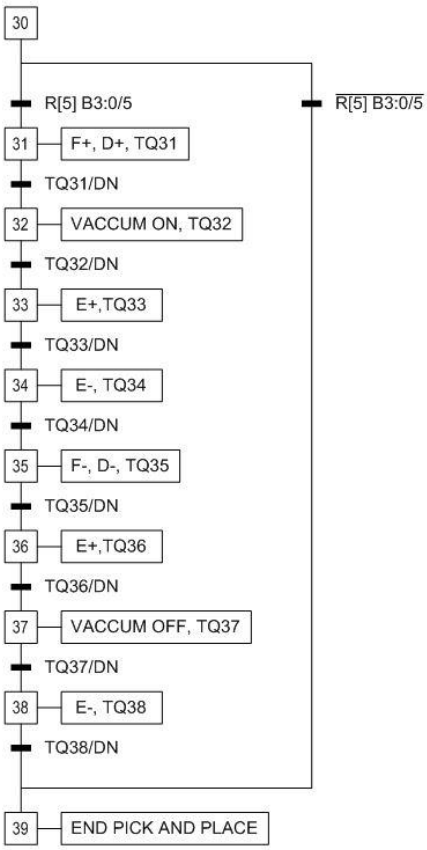

Figura 3. Grafcet de Macroetapa de Estación de tapado

En la estación de rechazo, un sensor detecta la presencia de la tapadera del bote si esta se encuentra presente no realizara el proceso de rechazo, que consiste en abrir una compuerta activada por el actuador H. Al realizar el Grafcet de este proceso se obtiene el grafico de la Figura 3.

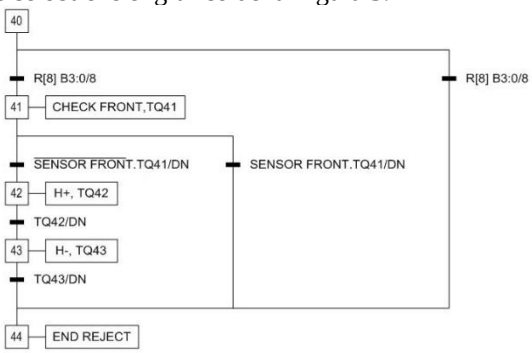

Figura 4. Grafcet de Macroetapa de Estación de rechazo.

En cada una de las estaciones se hace uso de un salto hacia adelante en función de la información que guardan los bits de la palabra B3:0 del registro de corrimiento, la cual determina la presencia de bote en cada posición del sistema de indexado.

Una vez realizadas las macroetapas de las estaciones, se procede realizar el grafcet principal tomando las consideraciones de la guía GEMMA como el tipo de coordinación. En la coordinación horizontal las tareas descritas en los grafcets de las macroetapas son realizadas de manera independientes y después son enlazadas a través de condiciones en sus transiciones. En la coordinación piramidal o jerarquizada, se realiza un grafcet principal que mandara a llamar a cada una de las macroetapas. Para este trabajo se realizó la coordinación piramidal, realizado el Grafcet principal mostrado en la Figura 5. Como se observa en esta figura, al terminar el indexado deberán iniciar de manera simultánea todas las estaciones de trabajo por lo tanto se utiliza la divergencia y convergencia en Y.

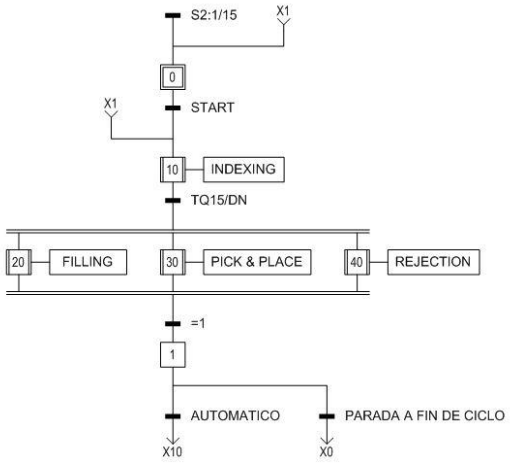

Figura 5. Grafcet principal de tipo piramidal o jerarquizado.

El diseño de la interfaz de usuario es desarrollado en función de los estados definidos en la guía GEMMA para este proceso se implementaron dos ciclos parada que cambian del Paro en estado inicial (A1) al estado de Producción Normal (F1) y viceversa. Un ciclo es a través de la Demanda de Paro a Fin de Ciclo (A2) y el segundo ciclo es a través del paro de Emergencia (D1), como se muestra en la figura 6.

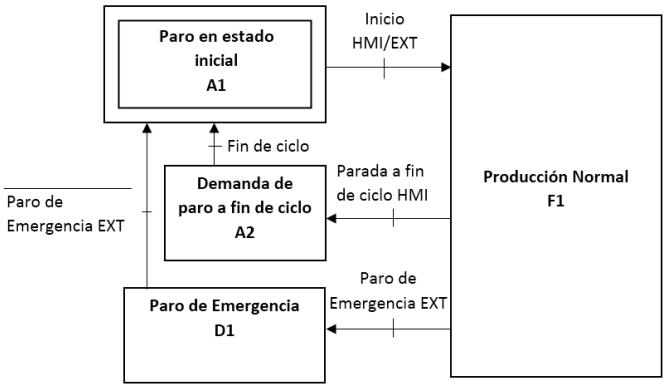

Figura 6. Estados de la guía GEMMA

Las transiciones que provocan el cambio de estado en nuestra guía GEMMA son el inicio, la activación del paro de emergencia Exterior, la desactivación del paro de emergencia exterior, la parada a fin de ciclo y el fin de ciclo que corresponden a la finalización de las macroetapas, el funcionamiento de estas transiciones ya están implementadas en el grafcet principal de la figura 5, podrán estar de manera física en la botonera de la estación o podrán estar en el HMI o interfaz de usuario desarrollado en la LabView.

2.3. Diagrama Escalera.

El diagrama escalera del proyecto deberá estar organizado en diferentes archivos de programa o diagramas de escalera, como se muestra en la figura 5 donde cada macroetapa deberá tener su diagrama escalera de manera independiente, se deberá tener un archivo de programa de Inicialización la cual deberá ser ejecutada en cada ciclo de encendido del controlador o cuando este cambie de modo stop a modo run. Para el rastreo de fallas deberá tenerse un diagrama de escalera para las Salidas y utilizando lógica combinacional los estados activaran las salidas del proceso las cuales deberán ser únicas, es decir solamente una salida energizada para cada dirección de salida.

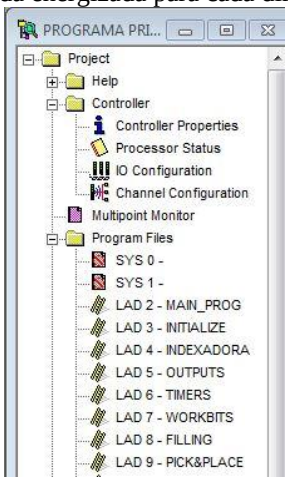

Figura 7. Organización de los archivos del programa.

Los estados del Grafcet pueden ser implementados de dos maneras, utilizando circuitos de retención o utilizando salidas Latch/Unlatch, como se muestra en la figura 8

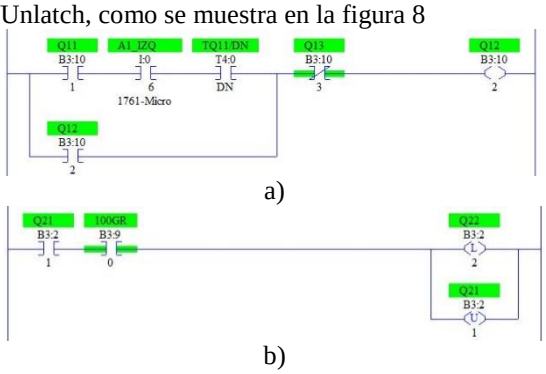

Figura 8. a) Circuitos de retención, b) salidas Latch y Unlatch

2.4. Interfaz de usuario en Labview.

La conectividad OPC es un estándar de comunicación basado en la tecnología OLE proveída por Microsoft, y desarrollada y mantenida por la fundación OPC que consiste en una coalición de compañías de la industria manufactureras donde Rockwell Automation es miembro. El objetivo de este grupo es de proveer un mecanismo de comunicación industrial estándar entre los dispositivos de piso y las aplicaciones cliente. Existen varios proveedores que ofrecen la conectividad OPC, uno de ellos es RSLinx Classic el cual es un servidor OPC compatible, con las interfaces necesarias para una aplicación OPC cliente que permite acceder a datos de forma consistente de un servidor OPC, además RSLinx Classic provee varios formatos como DDE (Dynamic Data Exchange) [6].

Para este trabajo se utilizó el OPC server de RSLinx, después de haberlo configurado se procedió a configurar un cliente del servidor OPC en un proyecto de Labview, a través de la librería creada en el proyecto se accede al enlace OPC1 para enlazar las transiciones programadas en el PLC, como se puede observar en la figura 9.

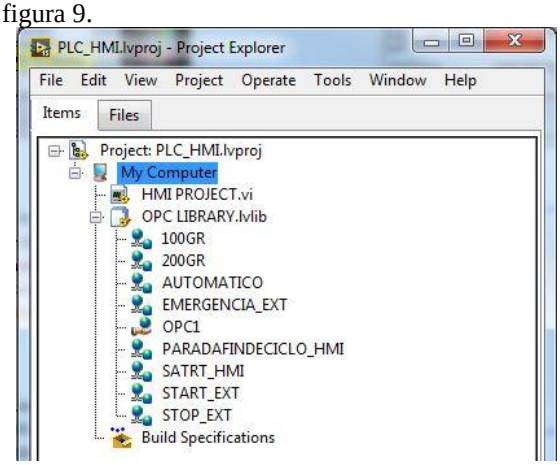

Figura 9. Configuración del Cliente OPC.

Una vez agregadas las variables al proyecto se realiza el VI, con el nombre HMI PROJECT, definiendo el panel frontal como se muestra en la figura 10.

Para este tipo de proyectos donde se están realizando dos tareas a diferentes velocidades es necesario definir dos While Loops, el primero estará realizando la tarea de leer los cambios de valor de las variables que ocurren en el servidor OPC y el segundo While Loop atenderá los eventos que ocurren en el panel frontal. Debido a esta necesidad se escoge el patrón de diseño de productor consumidor de datos y productor consumidor de eventos respectivamente

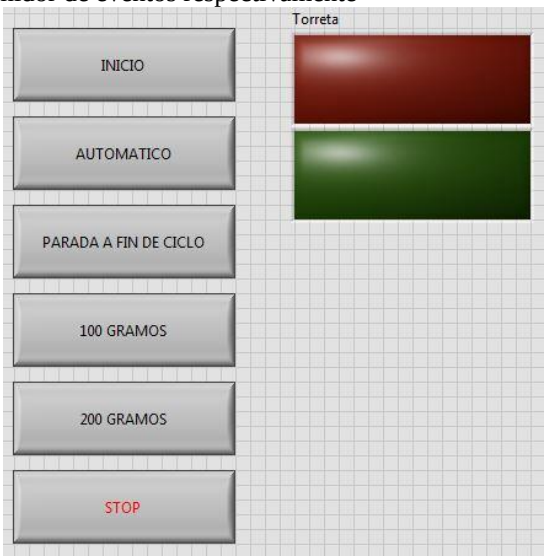

Figura 10. Panel frontal.

Para el productor consumidor de datos que atenderá a los cambios de las variables del PLC se realiza el código del productor como se muestra en la figura 11. Donde se están detectando los flancos de subida, utilizando la operación mayor que y un feedback node, con esto se evita el duplicado de las acciones almacenadas en la cola.

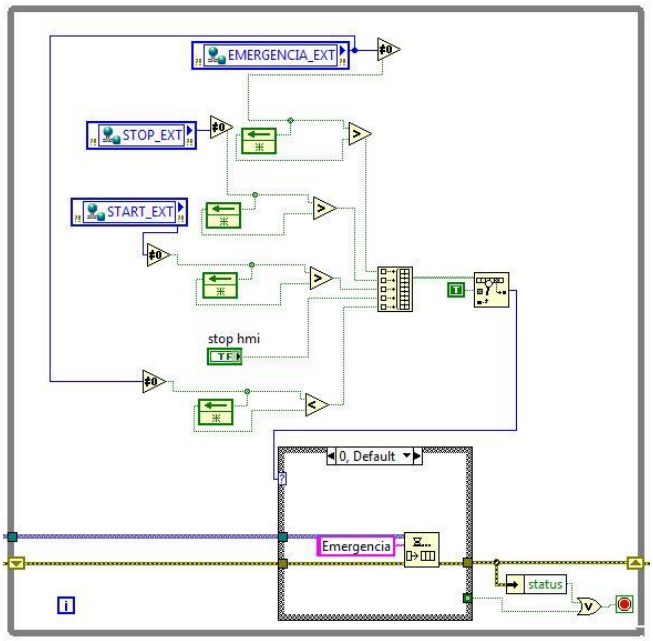

Figura 11. Código del productor de datos

En el productor consumidor de eventos, se configura un evento para cada control booleano del panel frontal y en el respectivo case del evento se ingresa en la cola la acción a ejecutar por el consumidor como se muestra en la figura 12.

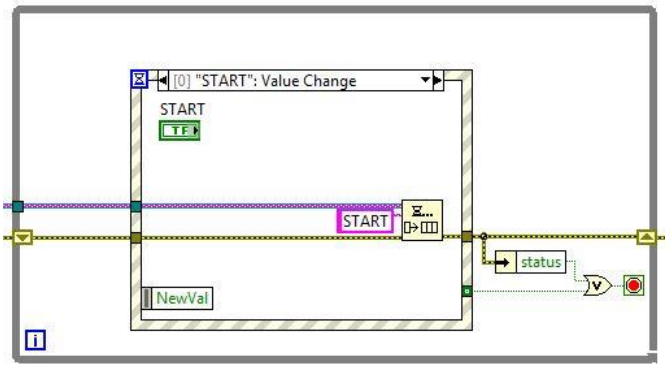

Figura 12. Código del productor de eventos.

El consumidor de las acciones del productor de eventos deberá escribir en la variables configuradas en el OPC server, las cuales serán transiciones de algunos de las macroetapas programadas en el PLC. Debido a los tiempos de actualización de las variables del servidor OPC es más lento que los ciclos de ejecución del consumidor de eventos es necesario que después de cambiar el valor de una variable se tenga un tiempo de retardo para asegurar que el cambio se realice, como se muestra en la figura 13.

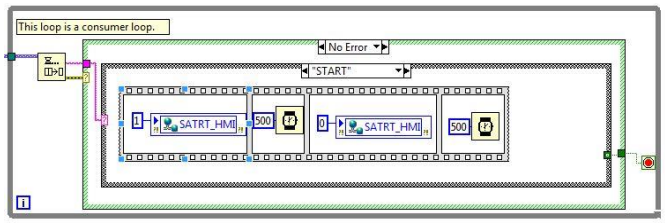

Figura 13. Código del consumidor de eventos.

2.5. Resultados

Al implementar el código derivado de los gráficos funcionales para cada uno de los procesos permite al programador realizar una depuración de manera independiente de cada una de las tareas a realizar. Es necesario contar en cada uno de los actuadores se tengan los inicios y finales de carrea permitiendo transiciones más fiables en el cambio de estados evitar colisiones en los actuadores de las estaciones. Al utilizar la metodología de diseño expuesta en este trabajo se obtuvo un software de fácil escalamiento, ya que las tareas se encuentran organizadas de manera independiente en los archivos del programa. El uso de la metodología del Grafcet permitió lograr un funcionamiento de cada una de las estaciones sin tener que hacer modificaciones o depuraciones a prueba y error. La comunicación OPC a través de RSlinx fue satisfactoria y la característica del servidor de utilizar la base de datos generada por el archivo del programa con extensión RSS permite la utilización de los símbolos utilizados en el software de programación del PLC, acelerando los tiempos de programación.

El uso del patrón de diseño productor consumidor y las herramientas de comunicación permiten el añadir funciones para alimentar cualquier base de datos a través de ODBC (Object DataBase Connectivity).

2.6. Conclusiones.

El uso de la metodología del Grafcet y la Guía Gemma como herramienta de diseño para automatizar procesos industriales genera un estándar en la generación de código para distintos controladores utilizados en la industria y la academia.

Al tomar las consideraciones de la guía Gemma como el tipo de coordinación entre macroetapas, los estados de funcionamiento, los estados de paro y los estados de paro de emergencia permiten al profesional de la automatización acotar proyectos y valorizar el trabajo a realizar.

Los avances tecnológicos en el área de computación, la aparición de sistemas operativos más robustos como Windows Embedded, permiten que tabletas electrónicas, pantallas táctiles o televisores con computadoras miniaturas (Compute Stick) puedan correr aplicaciones como Labview o sus ejecutables convirtiéndolos en Unidades Terminales Remotas de bajo costo de un sistema SCADA que alimentan a servidores de Base de Datos para y lograr la comunicación vertical con los diferentes niveles de la pirámide de automatización.

2.7. Referencias.

[1] Chang Jianto & Qiu Yuanying. (2013). Integrated system of ERP and MES Based on DHNN Evaluation. 2013 Fourth International Conference on Digital Manufacturing & Automation. 709-712.
[2] Kenji Mizuoka & Masanobu Koga. (2010). MDA Development of Manufacturing Execution System Based on Automatic Code Generation. SICE Annual Conference 2010, 3103-3106.
[3] W. Mahnke, S.H. Leitner, and M. Damm. OPC Unified Architecture. SpringerLink: Springer e-Books. Springer Berlin Heidelberg, 2009.
[4] J. Balcells, J. L. Romeral. Automatas Programables. Alfaomega, 1998.
[5] R. Piedrafita Moreno. Ingeniería de la Automatización Industrial. Alfaomega, 2004.
[6] A. Rodriguez Penin. Sistemas Scada. Alfaomega 2013.
[7] M.L. Alvarez, A. Burgos, I. Sarachaga, E. Estévez, M. Marcos, GEMMA based approach for generating PLCopen Automation projects, IFAC Proceedings Volumes, Volume 45, Issue 4, 2012, Pages 230-235, ISSN 1474-6670, http://dx.doi.org/10.3182/20120403-3-DE-3010.00067.