

Mech-SuperVi/Server: Programa para la supervisión de procesos mecánicos

Autores:

Alberto Villalonga Jaén¹. alberto.villalonga@umcc.cu

Ramón Quiza Sardiñas². ramon.quiza@umcc.cu

Yarens Cruz Hernández³. yarens.cruz@umcc.cu

¹ Alberto Villalonga Jaén. Graduado de Ingeniería Automática. Master en Ingeniería Asistida por Computadora. Profesor Asistente del Departamento de Mecánica de la Universidad de Matanzas. Miembro del Grupo de Investigación de Fabricación Avanzada y Sostenible.

² Ramón Quiza Sardiñas. Graduado de Ingeniería Mecánica. Doctor en Ciencias Técnicas. Profesor Titular del Departamento de Mecánica de la Universidad de Matanzas. Coordinador del Grupo de Investigación de Fabricación Avanzada y Sostenible.

³ Yarens Cruz Hernández. Graduado de Ingeniería Automática. Profesor Instructor del Departamento de Informática de la Universidad de Matanzas.

Resumen

Mech-SuperVi/Server forma parte de un paquete que permite el monitoreo de señales de procesos y sistemas mecánicos, basado en tecnologías de placa reducida. Mech-SuperVi/Server realiza las funciones de servidor, tomando las señales de los dispositivos de monitoreo y almacenándolas. La comunicación con los dispositivos de medición y de consulta se realiza a través de la suite de protocolos TCP/IP. Adicionalmente, el programa puede visualizar los valores medidos y los históricos y servir de interfaz para la toma de decisiones. Mech-SuperVi/Server dispone de una interfaz gráfica de usuario para realizar las principales tareas de la aplicación. La aplicación es multiplataforma y se distribuye bajo licencia GPL (versión 3).

Palabras claves: Monitoreo, Qt, Arquitectura cliente-servidor, Procesos mecánicos.

Abstract

Mech-SuperVi / Server is part of a package that allows the monitoring of process signals and mechanical systems, based on s. Mech-SuperVi / Server performs the server functions, taking the signals from the monitoring devices and storing them. Communication with the measurement and query devices is done through the TCP / IP suite of protocols. In addition, the program can display measured and historical values and serve as an interface for decision making. Mech-SuperVi / Server has a graphical user interface to perform the main tasks of the application. The application is cross-platform and distributed under GPL license (version 3).

Keywords: Monitoring, Qt, Client-Server architecture, Mechanical process

Índice:

Introducción	3
Materiales y métodos:	3
Resultados:	7
Conclusiones:	9
Referencias Bibliográficas:	9

Introducción:

Los sistemas de monitoreo son de vital importancia para el funcionamiento de cualquier proceso productivo pues nos brinda la posibilidad de comprender de una manera detallada su dinámica de trabajo lo que posibilita que se realice corrección sobre sus principales parámetros de ajuste con la meta de lograr un funcionamiento más óptimo y eficiente.

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 & Tan, 2006).

Casi en su totalidad todos los sistemas de monitoreo que se utilizan en nuestro país son foráneos lo que trae consigo una gran inversión inicial debido a sus altos precios en el mercado. El Grupo de Fabricación Avanzada de la Universidad de Matanzas entre sus líneas de investigación se encuentra la automatización, el monitoreo y control de procesos mecánicos. Dentro de esta línea surgió como proyecto la creación de una arquitectura flexible y de bajo costo para el monitoreo de sistemas y procesos mecánicos. En la figura 1 se muestra la arquitectura propuesta para dicho proyecto (Villalonga et al, 2016).

Como parte fundamental de esta arquitectura se puede destacar la función del servidor que es el encargado de recibir la información captada por el ordenador de placa reducida (SBC), procesarla y presentarla a los usuarios del sistema. Para ello se desarrolló una aplicación de código abierto y multiplataforma que es el objeto de este trabajo.

Materiales y métodos:

La arquitectura de software propuesta para el sistema de monitoreo, ver figura 2, está basada en una arquitectura cliente servidor en la cual el software que se ejecuta en el ordenador de placa reducida realiza la función de cliente y Mech-SuperVi/Server ejecutada en una PC de servidor. El procedimiento empleado para su implementación está acorde al Proceso Unificado Racional (RUP) (Jacobson et al, 2000).

Mech-SuperVi/Server fue desarrollado íntegramente en C++ en el IDE multiplataforma Qt. La aplicación es la encargada de la recepción, el procesamiento, la presentación y el almacenamiento de la información enviada por la Raspberry-Pi. En la figura 3 se muestra la ventana principal de la aplicación.

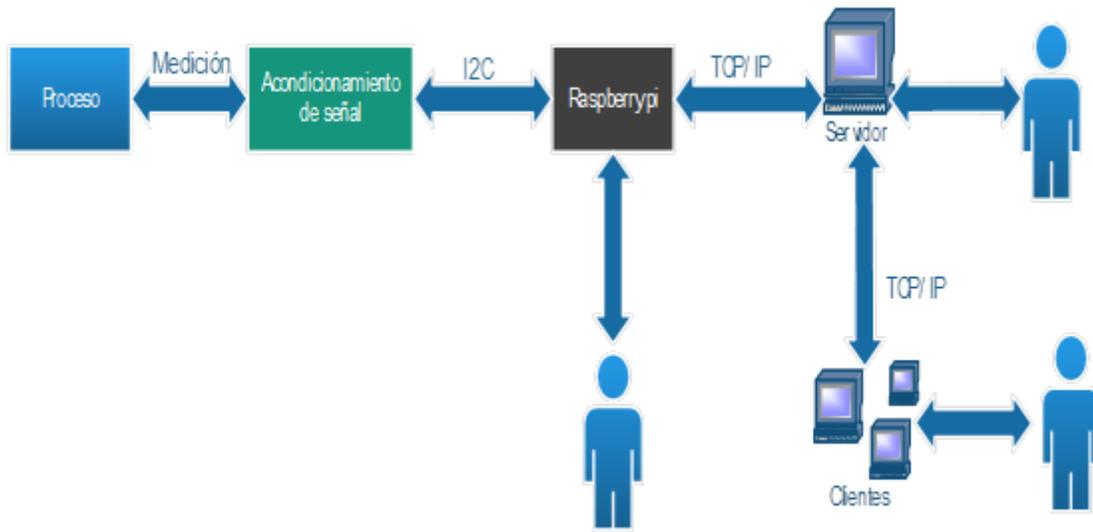


Figura 1. Arquitectura hardware de sistema de monitoreo para procesos mecánicos.

Para la comunicación con la SBC se consta con una interfaz de red, la cual utiliza el protocolo TCP/IP para garantizar el establecimiento de las conexiones y el intercambio confiable de datos. Sin embargo debido a que TCP/IP pertenece a las capas de transporte (TCP) y red (IP) del modelo OSI de la ISO(12) y no es responsable de darle el formato deseado por una determinada aplicación a su campo de datos ni de la lógica en el intercambio de los mismos, para ello se hace necesario la creación de un protocolo sobre la capa de aplicación de dicho modelo que se adapte a las necesidades del sistema, por lo que se implementó una trama de datos personalizada, a la cual se le realiza el chequeo de errores a través de una comprobación de redundancia cíclica (CRC); debido también a que TCP/IP no posee seguridad intrínseca, se brinda la opción para entornos donde la seguridad informática

es de vital importancia la transmisión de la información de manera encriptada, mediante el uso del Protocolo de Capa de Conexión Segura (SSL).

Como se puede observar en la figura 2 la aplicación está formada por cuatro módulos fundamentales: procesamiento de datos, visualización, modelación y configuración remota.

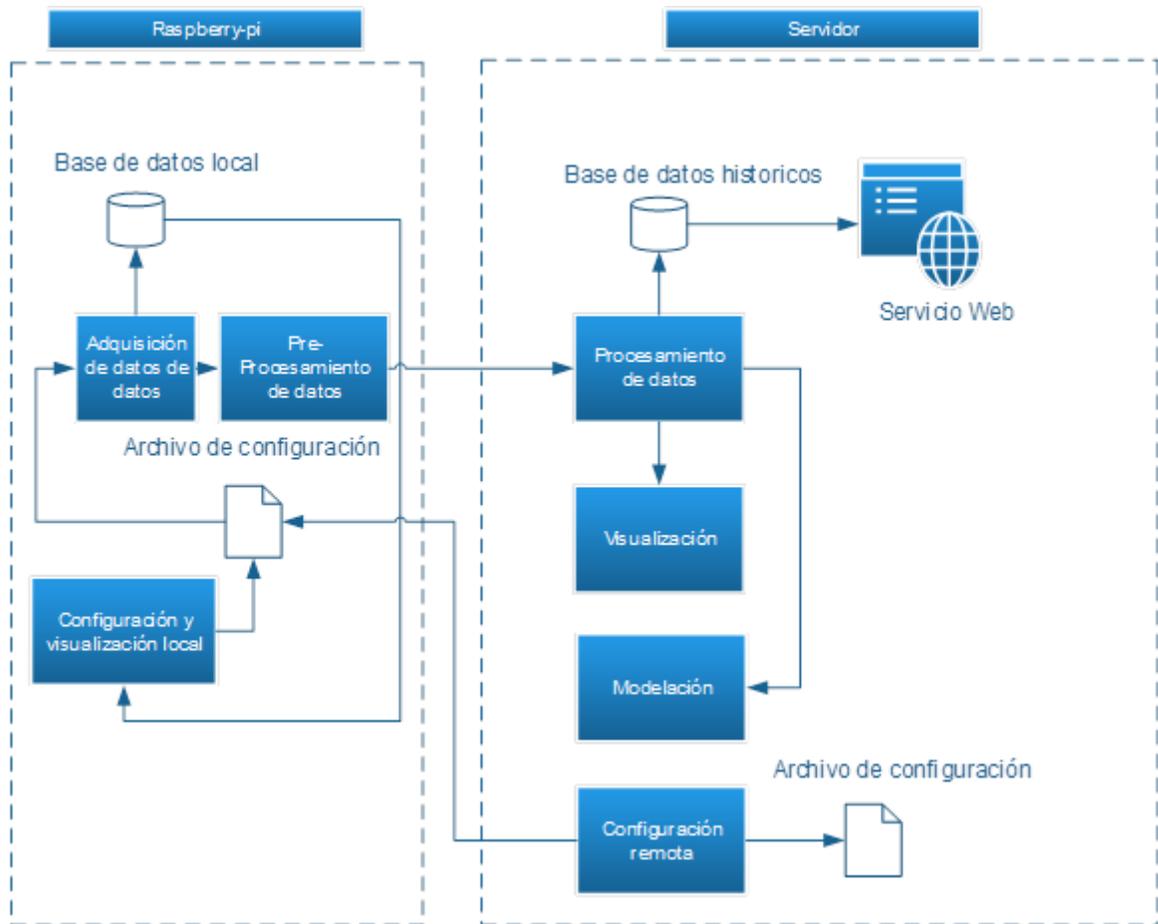


Figura 2. Arquitectura de software cliente-servidor del sistema de supervisión de procesos mecánicos.

El procesamiento de datos viene formado por una serie de herramientas que permiten tanto el filtrado de la señal como la obtención de rasgos basados fundamentalmente en estadígrafos. Apoyada en la utilización de la Librería Científica GNU (GSL) se encapsulan funcionalidades como la transformada rápida de Fourier, la transformada discreta de wavelet, y estadígrafos

como: raíz cuadrática media, pico máximo, kourtosis, desviación estándar, media, cálculo de errores, entre otros.

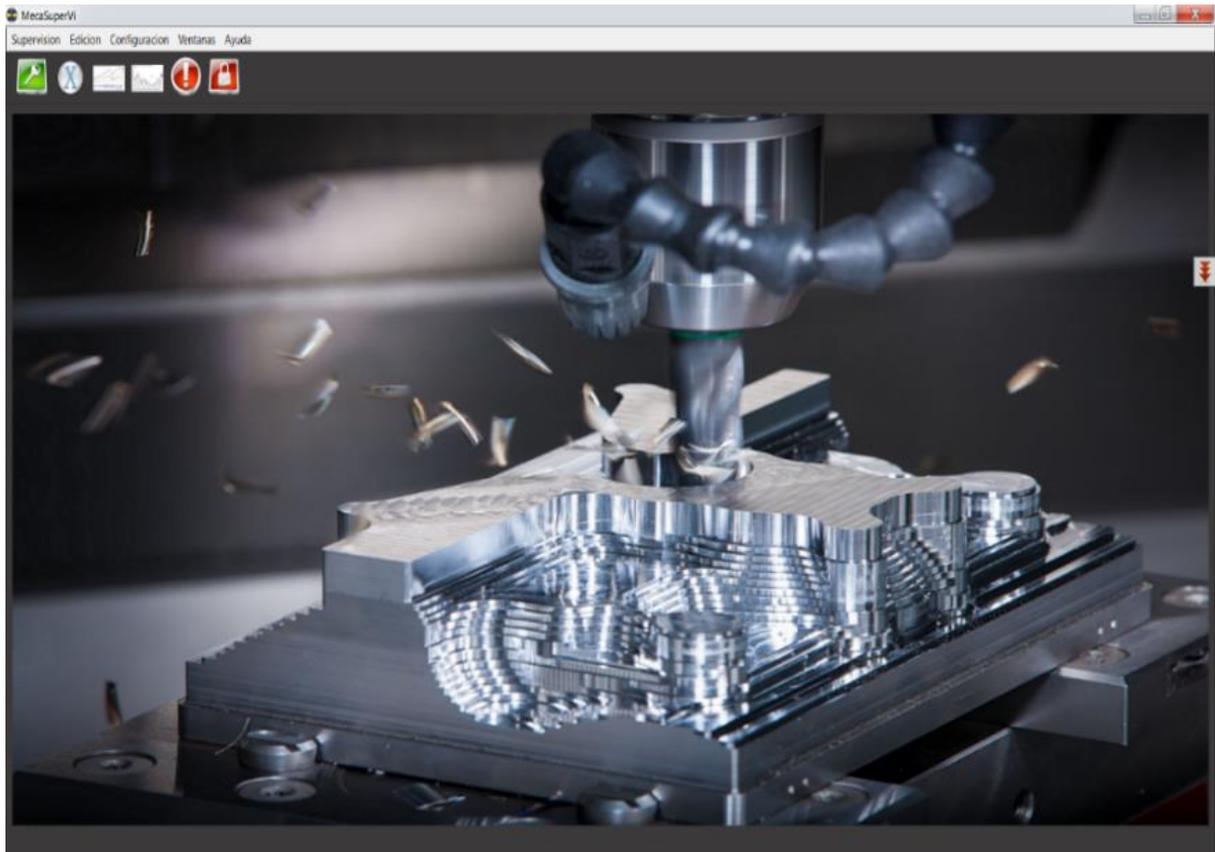


Figura 3. Ventana principal de Mech-SuperVi/Server.

El módulo de visualización se encarga fundamentalmente de realizar la presentación de la información obtenida del proceso para ello se basa en la presentación de datos a través de ventanas de tendencias, históricos y tablas de variables. Para graficar los datos se hizo uso de la biblioteca Qwt, la cual se distribuye bajo la licencia Pública General de GNU.

El módulo de modelación permite la configuración y la obtención de modelos del proceso a través del uso de regresiones matemáticas como el método de los mínimos cuadrados ordinario o a través del uso de redes neuronales artificiales tipo perceptron multicapa. Consta de una interfaz gráfica para la configuración de los parámetros de ambos métodos.

El módulo de configuración presenta una interfaz gráfica que brinda la posibilidad de configurar las variables del proceso que se van a medir con todos sus parámetros como son: tipo de señal de entrada, rango de medición, rango de alarma, entre otros. En la figura 4 se muestra una captura de la ventana de este módulo.

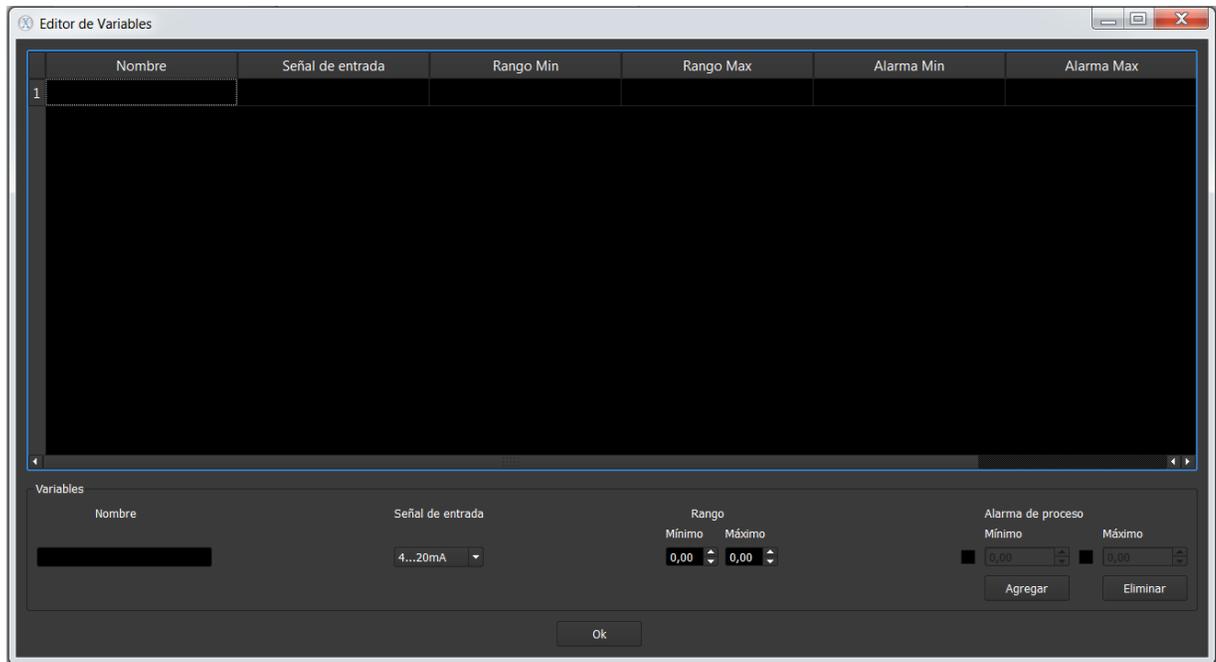


Figura 4. Ventana módulo de configuración.

Resultados:

El sistema se validó a través de la utilización de pruebas de caja negra las cuales se realizan partiendo de los requisitos funcionales del sistema, sin tener en cuenta su estructura interna. Las pruebas consisten en aplicar un conjunto de juegos de datos bajo determinadas condiciones y observar las salidas que se obtienen, para determinar si la funcionalidad que se prueba opera correctamente.

Para la prueba, el sistema de monitoreo se conectó a un proceso de torneado de semiacabado, en el cual se midieron las componentes de la fuerza de corte axial y tangencial. Para ello, se conectaron dos puentes de Wheastone formados por cuatro extensómetros como se muestra en la figura 5.

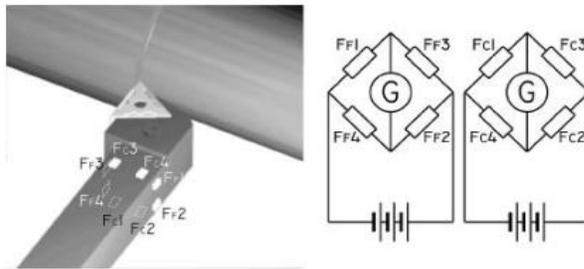


Figura 5. Puentes de Wheastone para la medición de fuerza.

En las figuras 6 y 7 se observan las capturas de las pantallas de los gráficos generados para la fuerza de corte axial y tangencial respectivamente. De manera general la aplicación tuvo un comportamiento estable puesto que las mediciones y las gráficas se comportaron de acorde a los valores esperados y no se reportaron pérdidas de datos ni de conexión.

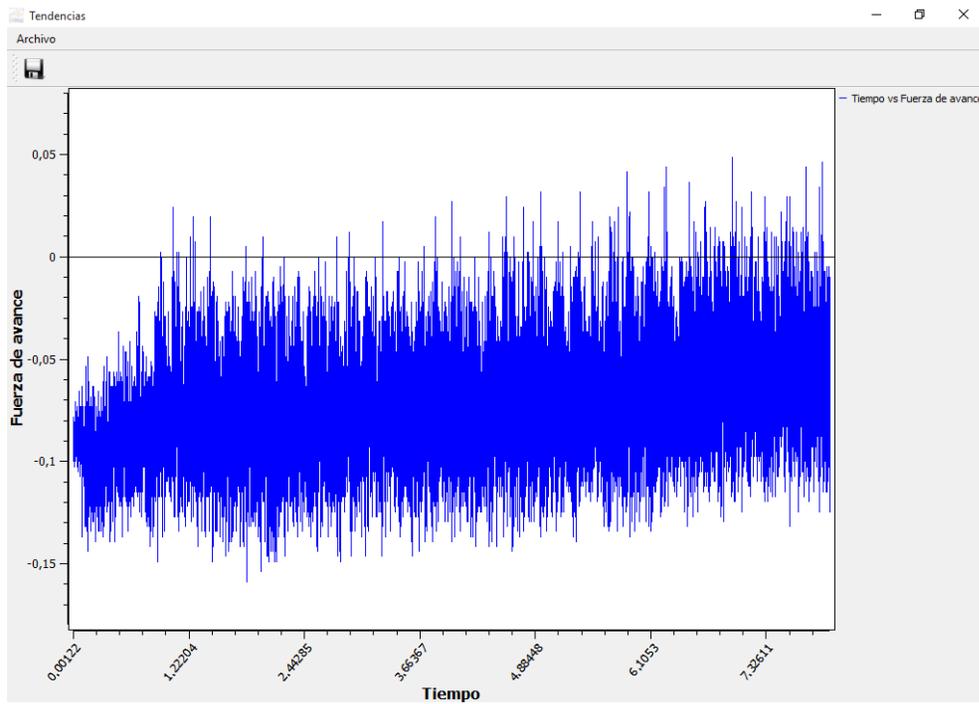


Figura 6. Fuerza de corte axial

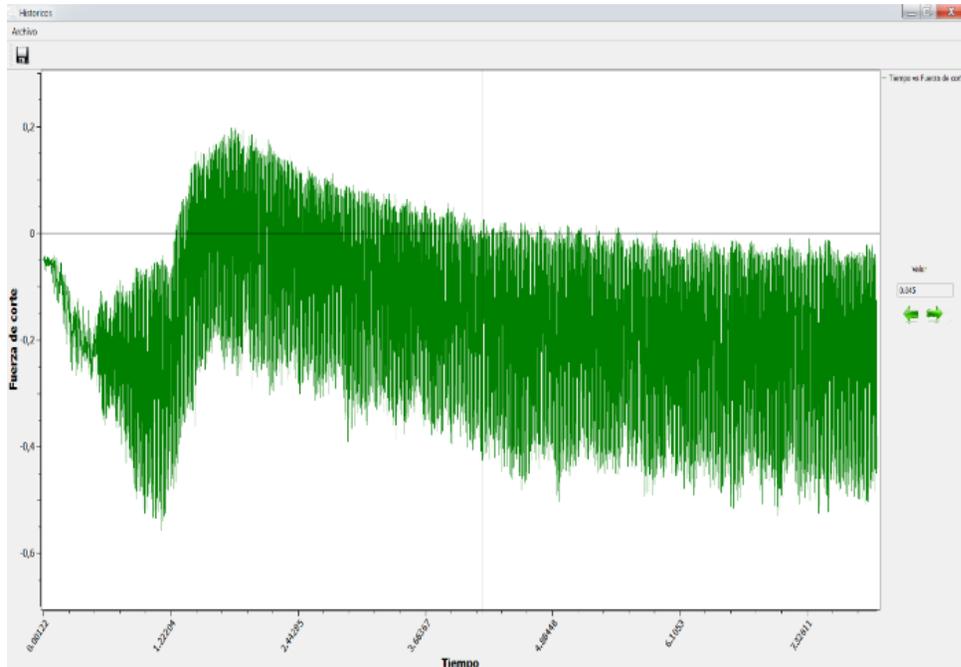


Figura 7. Fuerza de corte tangencial

Conclusiones:

Se realizó el diseño de una aplicación que cumple las funciones de servidor en un sistema de monitoreo de proceso mecánicos que permite la visualización y el almacenamiento de las variables del proceso, brinda la posibilidad de obtener modelos matemáticos que nos permiten realizar el análisis de la dinámica del proceso, su estabilidad y la optimización de sus parámetros principales, además de análisis offline para la detección de fallos, mediante el uso de otras herramientas de software, permitiendo así reducir los tiempos de paradas por roturas y que los mantenimientos sean más cortos y eficientes.

Referencias Bibliográficas:

Wang L. y Tan K. C. (2006). “Modern Industrial Automation Software Design” (en ingles), John Wiley & Sons, Inc., Hoboken, New Jersey, USA.

Villalonga, A., Rodriguez, G., Quiza, R., Haber, E. “Arquitectura flexible y de bajo costo para sistema de monitoreo de procesos mecánicos”, presentado al XVII Convección Científica de Ingeniería y Arquitectura, La Habana Cuba, 2016, ISBN 978-959-261-533-5.

Jacobson I, Booch G, Rumbaugh J. “El Proceso Unificado de Desarrollo de Software”.
Madrid, Pearson Educación; 2000.