

INTEGRACIÓN DE DISPOSITIVOS Y REDES INDUSTRIALES SERIALES CON REDES TCP/IP USANDO JAVA

GERMÁN ANDRÉS RAMOS FUENTES

Ingeniero Electricista y Especialista en Automatización Industrial de la Universidad Nacional de Colombia. Profesor adscrito a la Facultad Tecnológica de la Universidad Distrital Francisco José de Caldas.
g_andresramos@eudoramail.com

Clasificación
Categoría Colciencias: 1

Fecha de recepción: 21-04-2003

Fecha de aceptación: 30-05-2003

Palabras claves: Integración de sistemas, comunicaciones industriales, redes industriales, Java.

Key words: System integration, industrial communications, industrial networks, Java.

1. Introducción

La Celda Manufactura Flexible de Experimental (CMFE) del Laboratorio de Mecatrónica de la Universidad Nacional de Colombia se constituye en un proyecto macro que pretende explorar los sistemas modernos de manufactura del entorno industrial. El proyecto de investigación lleva como título “Investigación Experimental en Automatización para Manufactura Flexible”; como parte de su desarrollo, específicamente en cuanto a los estudios adelantados en las áreas de automatización y comunicaciones industriales

y la puesta en funcionamiento de la CMFE, en el presente artículo se desarrolla el proceso de implementación de un sistema de integración y comunicación para la celda flexible, empleando redes de área local basadas en el protocolo TCP/IP.

En el documento se describen de manera breve los componentes de la celda experimental, con el propósito de enmarcar el problema de investigación; luego se aborda el diseño y estructura final del sistema de integración.

2. La Celda Flexible de Manufactura Experimental (CMFE)

El sistema se encuentra conformado por los siguientes dispositivos:

- Máquina experimental CNC: fresadora de 6 ejes
- Máquina de control de calidad CNC: digitalizador
- Módulo de dosificación y transporte CNC
- Manipulador Scara
- Red experimental RS485

Las máquinas CNC y el módulo CNC son manejados por controladores MX2000 de Superior Electric, mientras el robot Scara se maneja mediante una tarjeta de sensores y actuadores microcontrolada, desarrollada para tal fin. Estos dispositivos a su vez se encargan de establecer la comunicación de cada elemento en la red RS485. Cada módulo de la celda, además, está provisto de un PC asociado destinado al diseño y desarrollo individual de tareas.

2.1. Controlador MX2000



Figura 1. Controlador MX2000.

El controlador MX2000 es un sistema multiejes, capaz de controlar y sincronizar el movimiento de hasta ocho ejes en un sistema multitarea de hasta siete funciones simultáneas. El conjunto de instrucciones de programación en lenguaje Basic, la interacción de 350 Entradas/Salidas (E/

S) y los módulos controladores de motores tipo servo o paso a paso constituyen un sistema abierto, en cuanto a sus posibilidades de integración y compatibilidad, y adaptable para la implementación en control y automatización de procesos (Superior Electric, 2000: 12).

Las tareas del controlador se crean y programan mediante un computador personal (PC); desde él y mediante un protocolo de comunicación serial RS232 o RS485, los diferentes programas son enviados al dispositivo. Una vez cargados en la memoria del controlador, los programas pueden ser ejecutados de manera independiente y autónoma dependiendo de su robustez. El sistema también obedece a comandos enviados de manera individual al puerto serial del controlador; de esta forma pueden leerse las E/S, ejecutar programas o realizar movimientos en cada eje controlado (Superior Electric, 2000: 13,14).

Al habilitar la comunicación serial RS485, se posibilita la operación de los dispositivos en red, para lo cual es necesario asignar una dirección única a cada controlador. La interfaz RS485 es de cuatro hilos en comunicación *full duplex*.

• Software MX2000

Para facilitar la configuración y programación del controlador MX2000 la compañía Superior Electric proporciona una interfaz de software, igualmente llamada MX2000, que al instalarse en un PC permite configurar los ejes del controlador y diseñar las tareas a ejecutar en él. El software MX2000 realiza la compilación de los programas diseñados y carga el resultado vía RS232 en el controlador. Además, contiene un modo terminal, desde el cual se pueden enviar órdenes de manera directa para ser ejecutadas por el controlador.

2.2. Robot Scara

La celda flexible cuenta con un robot Scara IBM 7540 de cuatro ejes, con un alcance de 740 mm

y capacidad de 22,6 kg, diseñado especialmente para ensamble y manejo de material.

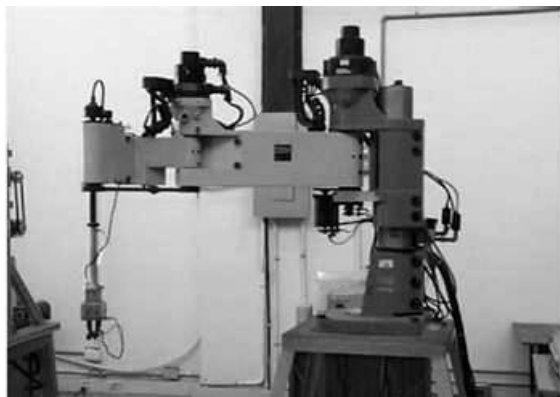


Figura 2. Robot Scara.

El manejo y comunicación con el robot Scara se realiza por medio de una tarjeta microcontrolada de sensores actuadores. El módulo de comunicación asíncrona cuenta con conexión RS485 para operación en red, o RS232 para permitir la conexión directa a un PC (Silva, 1995: 91).

Para el manejo del robot se dispone de un conjunto codificado de instrucciones. No obstante, el dispositivo controlador tiene como desventaja el no poder almacenar los programas desarrollados para el robot. Por esto, las tareas deben ser ejecutadas directamente desde el PC, o enviadas a través de la red RS485 (Silva, 1995: 95).

2.3. Red experimental RS485

La red se encuentra constituida por un bus común de conexión RS485 *full duplex* (conocido como RS485 de cuatro hilos) en conexión *daisy chain*, funcionando bajo un sistema maestro-esclavo. El nombre completo del estándar RS485 es TIA/EIA-485-A (Perrin, 1999: 1). El maestro, en este caso el PC servidor, se encarga de manejar el sistema y de establecer la comunicación con los equipos esclavos en la red, que son los controladores MX2000 y la tarjeta de sensores actuadores (Silva, 1995: 81).

2.4. Tarjeta conversora RS232/RS485

Este dispositivo hace posible la conexión del PC maestro por medio del puerto serial estándar RS232 a la red RS485; realiza la conversión de nivel de señal RS232 a RS485 y es configurable *Half Duplex* o *Full Duplex* (Silva, 1995: 102).

3. El Sistema de integración CMFE

El uso de ordenadores y redes corporativas en el sector industrial es cada vez más difundido, al establecerse más relaciones con los procesos de manufactura, producción, e involucrarlos con actividades administrativas y de planeación de la empresa. A nivel operativo, esta circunstancia significa intercambios eficientes de información entre las actividades administrativas, de planeación y de proceso. A nivel físico y funcional, este intercambio implica la interconexión de las diferentes redes que transmiten la información empresarial: la red de ordenadores del nivel administrativo y de planeación (red corporativa) y la red industrial (de las plantas de producción).

Las redes corporativas más empleadas en la actualidad son las basadas en el protocolo TCP/IP (Transmission Control Protocol/Internet Protocol); en ellas la distribución de la información se realiza mediante los diferentes servidores instalados en ellas: servidores de datos, archivos, aplicaciones y servidores web. Empleando una o varias estaciones-servidor en planta que tengan acceso a las dos redes, puede lograrse la difusión de la información hacia la red corporativa, y desde esta hacia la red industrial.

Considerando el creciente uso de Internet como herramienta de acceso y supervisión remota, tener un sistema en planta basado en la arquitectura TCP/IP se convierte en un paso siguiente muy cercano. También contribuyen favorablemente las características y ventajas que JAVA aporta a la integración, al constituirse en una herramienta independiente de la plataforma y desarrollada para trabajar en ambiente de redes de ordenadores e Internet (Sun Microsystems, Java Tutorial);

aunque su utilización en ambientes industriales es reciente, se trata de una interesante opción de experimentación en tales ambientes.

Con las anteriores consideraciones, el objetivo, en términos operativos, es obtener un sistema que comunique la red experimental RS-485 de la celda de manufactura con una red de área local basada en Ethernet y el protocolo TCP/IP, al igual que conseguir comunicación vía Internet. Desde el punto de vista funcional, el objetivo es ofrecer un sistema de integración entre las dos redes (administración-planta), ofreciendo una interfaz de usuario adecuada que permita el intercambio de información entre la celda de manufactura y medios computacionales externos ubicados en la red de área local TCP/IP e Internet (Ramos, 2003: 33).

4. Diseño general

Esta etapa del desarrollo del proyecto se ejecutó a partir de la definición de tres criterios de diseño relacionados entre sí:

- Bajo costo
- Mínima utilización de hardware
- Utilización intensiva de software

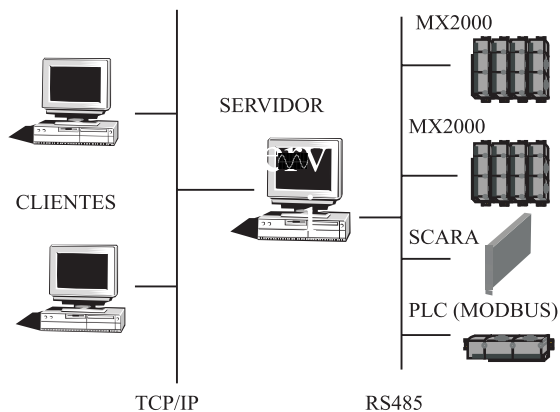


Figura 3. Sistema de integración.

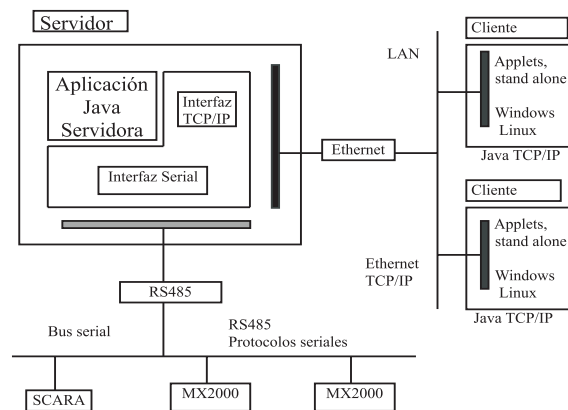


Figura 4. Diseño general.

Con el diseño, se busca heredar la confiabilidad y eficiencia de los estándares de comunicación empleados, al igual que mantener una arquitectura abierta y estándar.

En términos generales (Figuras 3 y 4) el diseño propone un servidor o nodo local, el cual representa la estación que maneja la información en planta (por ejemplo, el PC maestro en la celda de manufactura). Dicha estación se encuentra conectada a dos redes: a) la red en planta (red RS485 en la celda de manufactura), mediante la tarjeta conversor RS232/RS485 para la conexión al puerto serial estándar RS232 del PC; b) la red TCP/IP, mediante una tarjeta de red convencional.

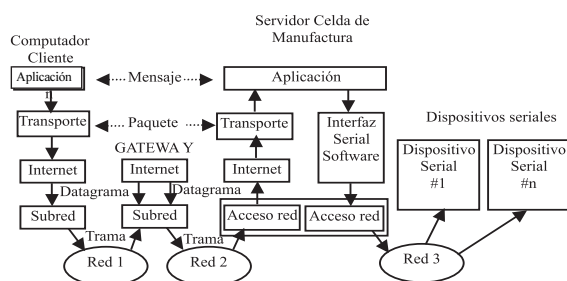


Figura 5. Flujo de mensajes.

La Figura 5 muestra el flujo de mensajes y su relación con el modelo de referencia Internet (*Internet Reference Model*) (Cisco, 1998: A-4). Una estación externa conectada a la red TCP/IP mediante una petición cliente-servidor, en donde la estación local hace el papel de servidor, solicita abrir un canal bidireccional entre el cliente o estación externa y uno de los dispositivos locales a cargo de la estación local.

Toda la información así enviada y recibida, es tomada por la estación local mediante su conexión TCP/IP. Interiormente, es dirigida hacia su puerto serial si la comunicación va de la estación externa hacia el dispositivo local a cargo; la información es tomada del puerto serial y enviada a la red TCP/IP si la comunicación va del dispositivo local a cargo hacia la estación externa. De cualquier forma, se mantiene una configuración maestro-esclavo entre la estación externa y el dispositivo local a cargo, con el PC maestro como intermediario, principal gestor y supervisor de toda comunicación.

La comunicación puede lograrse con cualquier dispositivo local; ello depende de la adecuada dirección que se indique desde la estación externa, que debe hacerse concordar con el estipulado en la red en planta (red RS485).

El sistema accede entonces desde la red TCP/IP externa a dos dispositivos: los controladores MX2000 y la tarjeta de sensores actuadores, ubicados y conectados en la red RS485. Cada uno de estos dispositivos tiene su propia co-

dificación y lenguaje de instrucciones, y debe ser implementado en software.

A nivel de aplicación, se reconocen dos partes primordiales: a) una aplicación servidora alojada en el PC maestro, que gestiona el establecimiento y administra la comunicación; b) las aplicaciones cliente, que realizan la petición de conexión y operan sobre los dispositivos de la celda. De esta manera, mediante la utilización de una red TCP/IP existente, la red RS485 y el empleo de la tarjeta de conversión RS232/RS485, una estación externa puede acceder al manejo de cada dispositivo de la celda flexible de manufactura (Ramos, 2003: 34).

5. Acceso a Internet

La extensión hacia una estación externa conectada con Internet se hace posible mediante un servidor que provea acceso a Internet para la red TCP/IP existente; por ejemplo, un servidor *proxy* que permita acceder al PC maestro en forma segura.

El canal de comunicación vía Internet pretende aprovechar las ventajas que la tecnología web puede aportar. La base para alcanzar este fin es el diseño de un sitio web y la configuración del PC maestro como un servidor web. Sobre ella se incrustarán *applets* (una de las ventajas de Java) con las mismas funciones de las aplicaciones creadas incluidas en el contenido de las páginas web; ello dará toda la funcionalidad para la conexión de la celda de manufactura en una intranet o en la red mundial.

6. Resultados obtenidos

A continuación se describen los resultados obtenidos, desde el estado inicial de la CMFE hasta la implementación del sistema de integración y comunicación. Luego se hace referencia a las funciones de cada una de las partes que componen el sistema.

6.1. Estado inicial

Inicialmente, la CMFE era manejada mediante un servidor programado en Visual Basic; este manejaba únicamente la celda en la red RS485 o RS232, no soportaba multitarea y cada operación conjunta de la celda debía ser programada en lenguaje *Visual Basic*, teniéndose que generar el protocolo de cada máquina durante la programación de cada nueva tarea de la celda.

El servidor contiene una interfaz para manejar de manera individual el robot Scara y los controladores MX2000. Debido a que la CMFE se encuentra en desarrollo, existe la necesidad de trabajar individualmente cada máquina para la ejecución y mejoramiento de procesos de mecanizado; teniendo en cuenta que el servidor existente solo permitía a un usuario trabajar en una máquina a la vez (perdiendo su utilidad), se retornó al trabajo con un PC en cada máquina en comunicación RS232.

6.2. Implementación del sistema de integración CMFE

Al implementar el sistema de integración desarrollado, un único servidor conectado a la red RS485 y la red TCP/IP permite manejar en conjunto la CMFE, al igual que establecer sesiones individuales en cada máquina. Todas las conexiones y sesiones se realizan desde la red TCP/IP y permiten el acceso remoto de los usuarios, para la operación, supervisión o control de las máquinas o de la CMFE en general.

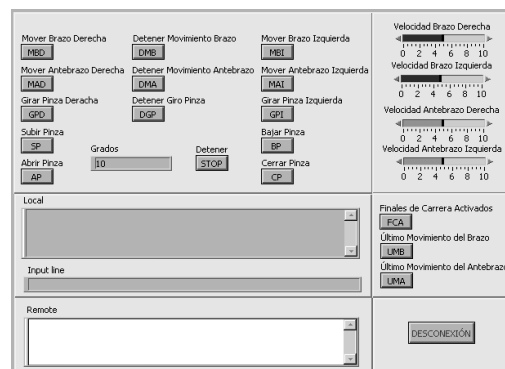
Dos aplicaciones fueron desarrolladas: el servidor CMFE (núcleo central) y los clientes, que se conectan a este núcleo y envían las peticiones de los usuarios para que luego sean dirigidas por servidor a los dispositivos de la celda. Para mostrar el desempeño de las comunicaciones se crearon tres interfaces de usuario: una para el controlador MX2000, una para el robot Scara y otra para el manejo de la CMFE en conjunto.

El Servidor CMFE se programó en Java y se activó para recibir peticiones de conexión en un puerto TCP definido; una vez se acepta una conexión se crea una clase para el manejo del nuevo cliente, quedando en espera de otras solicitudes de conexión. El manejo simultáneo de diferentes conexiones se realiza mediante la implementación multitarea, el uso de *threads* y *sockets*.

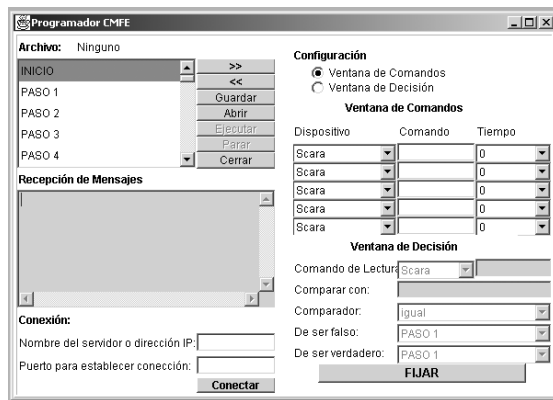
Los clientes fueron programados en Java y funcionan a la vez como aplicaciones *stand alone* o como *applets* dentro de páginas web. No obstante, ellos pueden programarse en cualquier lenguaje que provea manejo TCP/IP; para demostrar esta característica se desarrollaron clientes para el controlador MX2000 y el robot Scara en Visual Basic y LabView. La siguiente figura muestra tres diferentes interfaces de usuario implementadas.



a)



b)



c)

Figura 6. Interfaces de usuario: a) Interfaz Visual Basic para MX2000; b) interfaz LabView para robot Scara; c) interfaz Java para la programación de la CMFE en conjunto.

Las aplicaciones cliente desarrolladas en Java pueden correrse en todas las plataformas para las cuales existe la máquina virtual de Java (JVM), tales como Windows, Unix, Linux, Macintosh, Solaris (Sun Microsystems, Java Tutorial). Incrustadas en páginas web como *applets* ellas pueden solicitarse en una Intranet o vía Internet; para ello el servidor CMFE debe correrse en el mismo servidor web que da acceso a estas páginas.

El uso de la tecnología web y la gran integración de ordenadores demostrada en la red mundial son la base para sustentar el amplio campo de acción del sistema de integración diseñado para la CMFE.

6.3. Aplicación servidora - servidor CMFE

La aplicación servidora es el eje central del sistema de integración; por medio de ella se gestionan las peticiones de conexión de las aplicaciones cliente y se transmiten los mensajes, desde estas hasta los dispositivos seriales (red RS485) y, en sentido contrario, sus respuestas.

Dos interfaces de software y hardware bien definidas son manejadas. De una parte, la red de protocolo TCP/IP e interfaz Ethernet (tarjeta de

red Ethernet convencional), de otra, la red de bus serial RS485, con los protocolos distintivos de cada dispositivo (mediante el puerto RS232).

Las funciones de la aplicación servidora son:

- Atender las peticiones de conexión de los clientes TCP/IP
- Manejar la comunicación TCP/IP
- Mantener la conexión entre clientes TCP/IP y dispositivos seriales y realizar desconexiones, en caso de ser necesario
- Dirigir los mensajes de las aplicaciones cliente hacia los dispositivos seriales
- Estar en capacidad de intercambiar mensajes entre el cliente TCP/IP y el servidor o PC maestro
- Manejar el puerto serie del servidor o PC maestro
- Manejar los protocolos de comunicación de los dispositivos seriales: controlador MX2000 y robot Scara (entre más protocolos se abarque más posibilidades de aplicación tendrá el sistema de integración)
- Proveer un modo de envío de programas hacia los controladores MX2000
- Tener un archivo de las sesiones de trabajo efectuadas

6.7. Aplicaciones cliente

Cada aplicación cliente contiene el protocolo específico del dispositivo para el cual está diseñada; ella se encarga de elaborar los mensajes y tramas que se enviarán a través de la red TCP/IP y de ofrecer una interfaz que permita al usuario alejarse del manejo del protocolo.

Una de las funciones de la interfaz de usuario es proveerle las opciones suficientes para que a partir de la información ingresada puedan construirse mensajes y peticiones completas dirigidas a los dispositivos.

Existe la implementación de tres clientes, cada uno representado en una clase:

- MX2000 Cliente: para los controladores MX2000
- ScaraApplet: para el robot Scara
- Programador CMFE: para la programación de la CMFE en conjunto
- ModbusConnection: para comunicar esclavos de protocolo modbus (para expansiones futuras)

6.7.1. Petición de conexión

Luego de correr los programas debe realizarse la petición de comunicación. Todos los clientes poseen una interfaz de conexión que permite ingresar el nombre o dirección IP del servidor al que se desea conectar, y el puerto en el cual ese servidor atiende las peticiones de conexión. Los datos deben ser conocidos por el usuario; además, para realizar la conexión el servidor debe encontrarse en estado activo.

6.7.2. Applets y aplicaciones *stand alone*

Todas las aplicaciones clientes pueden funcionar como tales, o incrustarse dentro de páginas web como *applets*. No obstante, además de la definición, existe una diferencia importante entre los dos modos de implementación de los clientes:

Una aplicación corre de manera autónoma en el ordenador o *host* en donde se encuentra alojada (de allí su nombre *stand alone*), puede utilizar los recursos ofrecidos por el sistema operativo sin mayores restricciones, leer o guardar información en disco, etc. Sin embargo, para la ejecución de una aplicación Java de este tipo debe tenerse el archivo compilado y los componentes que este utiliza, almacenados en el PC.

Por el contrario, un *applet* es una aplicación que, al igual que la página web en donde se encuentra alojada, viaja a través de la red al ser solicitada. Por lo tanto, el cliente simplemente debe disponer de un navegador que soporte Java instalado

en su ordenador. El *applet* se ejecuta en el ordenador cliente y utiliza sus recursos, pero de manera limitada; por ejemplo, no puede guardar en disco ni conectarse con otro ordenador diferente de aquel desde el cual se descargó (Eck, 2002); pero a pesar de estas restricciones, su funcionalidad es suficiente para la aplicación aquí descrita.

6.7.3. Cliente MX2000

Existen varias funciones importantes que cumple la aplicación cliente MX2000:

- Realizar la petición de conexión y gestionarla exitosamente.
- Proveer el protocolo para comunicarse con el controlador MX2000.
- Proporcionar una interfaz de usuario fácil de manejar.
- Proporcionar un medio para el envío de programas al controlador

El propósito no es reemplazar el software MX-2000; por el contrario, este continúa siendo parte fundamental en la creación y compilación de los programas del controlador. El objetivo es proveer una herramienta que opere desde la red TCP/IP, capaz de manejar directamente el controlador y de proveer un medio para cargarle los programas previamente compilados con el software MX2000. El desarrollo de los programas puede realizarse automáticamente desde algún software CAD.



6.7.4. Cliente Robot Scara

Las funciones que provee la aplicación cliente para el robot Scara son:

- Realizar la petición de conexión y gestionarla exitosamente
- Proveer el protocolo para comunicarse con el robot Scara
- Proporcionar una interfaz de usuario fácil de manejar: suministrar los medios para controlar los diferentes movimientos del robot, así como la posibilidad de preguntar por desplazamientos realizados y finales de carrera activados

6.8. Tecnología web

Para obtener los beneficios de la tecnología web es suficiente que la aplicación servidora corra en un ordenador que provea funciones de servidor web, lo cual es proporcionado por sistemas operativos bien conocidos como Windows, Unix y Linux, o mediante aplicaciones que, instaladas en ordenadores corrientes, proveen las funciones de servidores web.

Los *applets* clientes son cargados por ordenadores en la red con navegadores comunes que soportan Java; a su vez, estos se conectan con la aplicación servidora. Para ello necesitan conocer el puerto por el cual se atienden las conexiones; una vez aceptada la petición, podrán acceder a los dispositivos seriales.

Los servidores web utilizan el puerto 80 para atender las peticiones de entrada a las páginas que solicitan los navegadores, mientras que la aplicación servidora recibe peticiones de conexión de las aplicaciones clientes (*applets*) en un puerto distinto. A cada conexión aceptada se le asigna un nuevo puerto, por el cual se realizarán las transacciones con el dispositivo serial. Esta es la forma como el ordenador puede mantener múltiples conexiones simultáneas con diferentes aplicaciones. La aplicación servidora atiende y mantiene de igual manera conexiones con aplicaciones clientes y *applets* clientes.



6.9. Hardware y software

Las especificaciones de hardware y software del sistema de integración se dividen en dos partes estructurales:

6.9.1. Servidor CMFE

La aplicación servidora actualmente necesita sistema operativo Windows, debido a que el manejo del puerto serial no viene implementado en la versión de Java conocida (JDK1.3.1), con la cual se desarrolló el proyecto. Para cumplir con esta función, la compañía Sun Microsystems provee un paquete llamado Javacomm, del cual existen actualmente versiones para Windows 32 bits y Solaris. El Servidor CMFE fue probado con el paquete para Windows; él debe tener instalada la máquina virtual de Java en la cual se recomienda la distribución JSDK2.

Como se mencionó, para aprovechar las ventajas de la tecnología web, se hace necesario que el mismo ordenador que aloja la aplicación servidora provea funciones de servidor web. Esto se logra desde el propio sistema operativo, o mediante aplicaciones que, instaladas en el ordenador, realizan estas funciones. Se recomienda que el ordenador, usado como servidor sea un Pentium III o superior, lo cual es aconsejable por razones de velocidad y desempeño, especialmente si se utiliza como servidor web a la vez. Para la comunicación con la red TCP/IP se requiere de una tarjeta de red Ethernet 10/100 convencional.

La interfaz serial debe proveer el estándar RS485 con conexión de cuatro hilos, y lo más indicado es que sea opto-aislada. Algunas cuentan con resistencia de terminación, que puede ser beneficiosa de acuerdo con la ubicación del servidor en la red RS485 (Perrin, 1999: 5). Cuando se tiene un conversor RS232/RS485 se recomienda que su alimentación se realice desde el mismo puerto serial.

6.9.2. Clientes CMFE

Las aplicaciones clientes pueden correrse en cualquier sistema operativo para el cual exista una máquina virtual de Java. Esto muestra el alto grado de integración que se adquiere al utilizar este sistema de comunicación para la CMFE, empleando Java en su desarrollo.

Utilizando *applets*, no es necesario tener instaladas las aplicaciones clientes en el ordenador cliente. Además, los navegadores que soportan Java generalmente se instalan junto con una versión de la JVM, en cuyo caso no es necesaria su instalación separada. En todo caso debe revisarse que la versión sea la correspondiente al JSDK2.

Cualquier arquitectura compatible con la Máquina Virtual de Java es una opción en este caso. Un ordenador con un desempeño similar a un PC con procesador Pentium II de 300 mHz y 64 Mb de RAM o superior, será suficiente. Para el manejo de red TCP/IP se puede utilizar un adaptador de red Ethernet 10/100 común.

7. Proyección de resultados

Aunque el sistema de comunicación e integración se implementó en la CMFE del laboratorio de Mecatrónica de la Universidad Nacional de Colombia, el sistema puede ser adap-

tado para sistemas de condiciones similares. Esto depende de la implementación de protocolos de comunicaciones de los dispositivos a conectar; entre más protocolos se consideren, más posibilidades de aplicación tendrá el sistema de integración; así, el ideal es trabajar con protocolos estándares para incrementar el nivel de integración. Una vez establecido el sistema de integración, pueden agregarse las utilidades que las bases de datos proporcionan; el sistema también puede ser usado en funciones de supervisión.

tado para sistemas de condiciones similares. Esto depende de la implementación de protocolos de comunicaciones de los dispositivos a conectar; entre más protocolos se consideren, más posibilidades de aplicación tendrá el sistema de integración; así, el ideal es trabajar con protocolos estándares para incrementar el nivel de integración. Una vez establecido el sistema de integración, pueden agregarse las utilidades que las bases de datos proporcionan; el sistema también puede ser usado en funciones de supervisión.

Para la programación de una celda de manufactura fue implementada una aplicación sencilla utilizando el concepto de diagrama de flujo; no obstante, se demostró que el sistema de integración es suficientemente robusto y eficiente como para ser empleado con técnicas más avanzadas de manufactura, en general para el manejo global de los dispositivos seriales involucrados.

Las aplicaciones creadas por el usuario no necesariamente deben ser desarrolladas en Java; se realizaron pruebas con clientes diseñados en Visual Basic y LabView, demostrando así que la funcionalidad de las aplicaciones es el objetivo final y el sistema de integración el intermediario y manejador de las comunicaciones entre las dos redes.

8. Conclusiones

La utilización de plataformas y arquitecturas abiertas es fundamental para lograr un alto grado de integración; esta es la elección más adecuada si se piensa en futuras expansiones y la actualización continua de procesos. Las plataformas y arquitecturas deben tener en cuenta los dispo-

sitivos involucrados en el proceso, los protocolos de comunicación y los medios físicos para lograrla y los lenguajes de programación, entre los elementos más importantes.

El proyecto de integración de la Celda de Manufactura Flexible Experimental con una red de protocolo TCP/IP se constituye en una aplicación de la filosofía que así se describe; para cumplirla, en el desarrollo del proyecto se escogió Java como lenguaje de programación.

Las ventajas de este lenguaje en la programación de aplicaciones que trabajan en red fue comprobada. La seguridad del código, la robustez en el manejo de excepciones, el manejo multihilos y su portabilidad fueron características de gran utilidad, generando importantes beneficios en el desempeño del sistema de comunicaciones implementado. Así mismo, al poder emplear las mismas aplicaciones como *applets* se obtienen todos los beneficios de la tecnología web, sin rediseñar ni modificar el código. La utilización del estándar EIA-RS485 en la red experimental de la CMFE facilitó la integración de las dos redes, lográndose el propósito de implementar arquitecturas y estándares abiertos.

Los resultados del proyecto se constituyen en una base sobre la cual pueden implementarse más funciones para el mejor desempeño de la CMFE. Para ampliar el campo de acción del sistema basta con estudiar la estructura de las aplicaciones cliente y servidora. Las interfaces de usuario implementadas cubren las necesidades de trabajo de cada dispositivo. A su vez, se deja toda la documentación disponible para la implementación de nuevas interfaces para la comunicación con otros dispositivos que se adicionen a la CMFE.

Al proveer de un medio para el envío de programas al controlador MX2000 se aporta una ventaja en cuanto a portabilidad y eficiencia en el manejo global de la CMFE. En general, utilizando este sistema de comunicación se está

en capacidad de proporcionar acceso remoto desde una red TCP/IP a todo tipo de dispositivo serial del cual se disponga de protocolo. Para esto, la programación orientada a objetos proporciona las ventajas de una fácil incrustación de nuevos protocolos.

Por medio de este trabajo se llegó a la solución de un problema comúnmente encontrado en la automatización de procesos industriales. Se abordó el problema de las comunicaciones industriales encontrando una solución de bajo costo basada en el uso intensivo de software en comparación con el hardware requerido. El uso de tecnologías abiertas posibilita las futuras expansiones, y el acercamiento al uso de la tecnología web abre nuevas posibilidades en el entorno industrial.

El sistema aquí descrito puede ser usado como base de un sistema de supervisión o como controlador global de las tareas de los dispositivos seriales conectados; las aplicaciones pueden ser desarrolladas en cualquier lenguaje de programación que maneje *sockets* y TCP/IP.

El sistema de comunicación desarrollado se constituye en una base para la creación de aplicaciones adicionales, dejando a futuros diseñadores un proceso transparente de obtención de resultados en cuanto a integración de redes. Las aplicaciones podrán ser complementadas pensando en la funcionalidad de la celda y el mejoramiento de su eficiencia y efectividad, obteniendo programas que permitan el manejo global del proceso de manufactura utilizando técnicas conocidas como Redes de Petri, Cadenas de Markov o nuevas teorías y técnicas que pueden generar futuros proyectos de investigación (Ramos, 2003: 60).



REFERENCIAS BIBLIOGRÁFICAS

- ECK, David J. *Introduction to Programming Using Java Version 4.0*, Julio 2002, <http://math.hws.edu/javanotes>
- Manual CISCO, Appendix A, Understanding-TCP/IP, Cisco Corp.
- PERRIN, Bob. The Art and Science of RS-485, Feature Article: Circuit Cellar online, www.circuitcellar.com
- RAMOS, G. A. (2003). *Integración y Comunicación de Celda Flexible de Manufactura Experimental con una Red de Área Local Basada en el Protocolo TCP/IP*. Universidad Nacional de Colombia, Bogotá.
- SILVA, William, C. y ORJUELA G. (1995). *Red Industrial con Interfaz RS-485 para Prototipo CMF*. Universidad Distrital F.JC. Facultad Tecnológica. Bogotá.
Sun Microsystems. Java Tutorial. <http://java.sun.com/docs/books/tutorial/>
- SUPERIOR ELECTRIC (2000). *Instructions for Superior Electric Slo-Syn Model MX-2000 Programmable Multi-Axis Motion Controller*.

