

Elección de un modelo para el control remoto de instrumentos de medida

Susana Pérez Mantilla¹, José Ángel Amador Fundora²

¹Instituto Politécnico Superior José Antonio Echevarría. Ingeniero. susana.pm@electronica.cujae.edu.cu

²Instituto Politécnico Superior José Antonio Echevarría. Máster en Ciencias. amador@electronica.cujae.edu.cu

RESUMEN

El campo de la instrumentación controlada por ordenador está experimentando un fuerte desarrollo en los últimos años lo que está conduciendo a una enorme variedad de tecnologías físicas de comunicación (GPIB¹, RS232², USB³, Ethernet), de arquitecturas de instrumentos (VXI⁴, PXI⁵, LXI⁶), así como una amalgama de capas de software (VISA⁷, IVI⁸), lenguajes de comunicación (SCPI⁹) y entornos completos de desarrollo de sistemas de control de instrumentación (LabView, MatLab). La selección de un modo para el control remoto de instrumentos de Radio constituye un problema a resolver a la hora de llevar a cabo el proceso de automatización en un laboratorio. Sin embargo, realizar las mediciones de forma automática permite la adquisición, el almacenamiento y procesamiento de datos de forma más eficiente.

Palabras claves: arquitecturas de instrumentos, lenguajes de comunicación, sistemas de control de instrumentación, tecnologías físicas de comunicación.

Choosing a remote control model for measuring instruments

ABSTRACT

The field of computer controlled instrumentation is experiencing strong growth in recent years which is leading to a huge variety of physical technologies of communication (GPIB, RS232, USB, Ethernet) instruments architectures (VXI, PXI, LXI) as well as an amalgamation of software layers (VISA, IVI) communication languages (SCPI) and complete development environment for instrumentation control systems (LabView, MatLab). The selection of a mode for remote radio instruments is a problem to be solved when carrying out the process in a laboratory automation. However, doing measurements automatically enables the acquisition, storage and processing of data in a more efficient way.

¹ Siglas del término en inglés para "General Purpose Interface Bus".

² Siglas del término en inglés para "Recommended Standard 232", conocido como Electronic Industries Alliance RS232C.

³ Siglas del término en inglés para "Universal Serial Bus".

⁴ Siglas del término en inglés para "Virtualization eXtensions for Instrumentation".

⁵ Siglas del término en inglés para "PCI eXtensions for Instrumentation".

⁶ Siglas del término en inglés para "LAN eXtensions for Instrumentation".

⁷ Siglas del término en inglés para "Virtual Instrument Standard Architecture".

⁸ Fundación IVI (en inglés, Interchangeable Virtual Instrument) promueve el empleo de especificaciones para la programación de instrumentación de medida.

⁹ Siglas del término en inglés para "Standard Commands for Programmable Instrumentation".

Key words: communication languages, communication physical technologies, instrument architecture, instrumentation control systems

INTRODUCCIÓN

La medición de variables es una actividad vital en cualquier rama de la ciencia. En muchos casos nos servimos de Sistemas de Adquisición de Datos (SAD) para obtener información de un determinado proceso.

En la actualidad, el método más ampliamente utilizado de registro de datos es con un data logger (o registrador sin papel). Los registradores de datos son instrumentos autónomos que obtienen señales de medida, las convierten en datos digitales y almacenan los datos internamente. Muchos registradores de datos, incorporada la muestra, dan la posibilidad de transferir los datos a un PC para su análisis fuera de línea, almacenamiento permanente o de generación de informes.

El uso de ordenadores con los datos de registro se inicia casi al mismo tiempo que se introduce el PC. Sin embargo, se necesitaban hardware y software especial para llevar a cabo su función. Hewlett-Packard, un fabricante de instrumentos de prueba y de adquisición de datos, desarrolló Hewlett-Packard Interface Bus (HPIB) para superar la lentitud de los primeros PCs. A medida que más fabricantes de instrumentos aprobaban el uso del HPIB, pronto se convirtió en el bus de interfaz estándar (GPIB). Posteriormente fue adoptado como un estándar IEEE, concretamente el IEEE-488. Cuando creció la potencia de procesamiento en los dispositivos de medición basados en PC, también lo hicieron sus capacidades. Los datos registrados utilizan ahora software y procesadores de alta capacidad, disco duro, pantalla y E/S, obteniendo características que van más allá del registro de datos. Los registradores de datos modernos ofrecen visualización en tiempo real, análisis en línea, funciones definidas por el usuario, terabytes de almacenamiento de datos, conectividad e interoperabilidad de red.

Dado que los sistemas de adquisición de datos basados en PC estarán siempre conectados al PC, las mediciones pueden visualizar en el monitor del PC en tiempo real. Los resultados son visibles al instante. Algunos entornos de programación, tales como LabView y MatLab, permiten a los usuarios crear interfaces personalizadas para controlar el dispositivo de medición y mostrar los resultados.

En este trabajo se analizarán las características de algunos de estos componentes con el fin de seleccionar el modelo más adecuado para llevar a cabo el proceso de control automático de equipos de instrumentación de medida en un laboratorio. Para el caso de la interfaz de comunicación se elegirá una entre GPIB o Ethernet y en dependencia de esta selección el lenguaje de comandos a emplear. Para el caso del entorno de programación se analizarán MatLab y LabView.

1. INSTRUMENTOS

El nivel más bajo del modelo lo constituye los instrumentos del entorno que se controla. Estos pueden ser:

- Fuentes de CD
- Multímetros digitales
- Generadores de funciones

- Osciloscopios
- Medidores de potencia
- Generadores de RF
- Analizadores de espectros
- Analizadores de redes, entre otros

Estos equipos pueden ser contralados a través de diversos buses como son: GPIB, RS232, USB y Ethernet. Para el control por GPIB los instrumentos están dotados con una tarjeta hardware de conexión al bus GPIB. Para su control disponen de un software interno de control que interpreta los mensajes que recibe por el bus GPIB e interacciona con el firmware propio del equipo. El "Parser" o intérprete es el thread o hebra de gestión del intercambio de mensajes por el bus GPIB.

2. INTERFACES DE COMUNICACIÓN

Por más de 20 años, científicos e ingenieros han utilizado ampliamente el estándar IEEE 488, bus de interfaz de propósito general (GPIB) para automatizar sistemas de instrumentación. A medida que la tecnología de computadoras entra al contexto de pruebas y mediciones, y buses como USB, Ethernet e IEEE 1394 son considerados para la conectividad en instrumentos, y surgen las preguntas acerca del futuro de GPIB como el bus preferido para el control de instrumentos.

2.1 Ethernet

Recientemente, los fabricantes de instrumentos han empezado a incluir Ethernet como una interfaz de comunicación alternativa. A pesar de que Ethernet es nuevo en las aplicaciones de control, es una tecnología madura que es ampliamente utilizada para sistemas de medición. Con más de 100 millones de computadoras alrededor del mundo con capacidad Ethernet, el argumento de utilizarlo como una solución para el control de instrumentos tiene mucho peso.

Las aplicaciones de control de instrumentos con Ethernet pueden tomar ventajas de las características únicas del bus, las cuales incluyen control remoto, facilidad para compartir los instrumentos entre usuarios, y una fácil publicación de los datos obtenidos. Por otra parte, los usuarios toman ventaja de las extensas redes Ethernet existentes en sus compañías y laboratorios. Sin embargo, esta ventaja podría ser un problema en algunas compañías ya que requerirá la intervención de administradores de red en las aplicaciones tradicionales de ingeniería.

Otros factores a considerar al examinar Ethernet para el control de instrumentos, son la velocidad de transferencia, determinismo y seguridad. La mayoría de las redes Ethernet de hoy en día son del tipo 10BaseT o 100BaseTX, las cuales transmiten datos a 10 Mb/s ó 100 Mb/s respectivamente. Sin embargo, estas tasas de transferencia son teóricas debido a factores como: tráfico en la red, retrasos e ineficiencia en la transferencia de datos. Por otra parte, debido a la incertidumbre en las tasas de transferencia, no se puede asegurar un determinismo en la comunicación a través de Ethernet. Finalmente, los usuarios que requieran integridad y privacidad en los datos deberán tomar medidas de precaución especiales.

2.2 GPIB

GPIB es un estándar de conexión que permite la comunicación de un ordenador con instrumentos electrónicos de medida. Las siglas corresponden a General Purpose Interface Bus, pero a pesar de este nombre, fue diseñado específicamente para la conexión de instrumentos de medida. [1]

El bus de transmisión de datos de GPIB es de 8 bits en paralelo, y lógica negativa con niveles TTL estándar (cierto si el voltaje es ≤ 0.8 V y falso si el voltaje es ≥ 2.0 V). Los cables y conectores tienen el aspecto típico mostrado en la Figura 1. Están apantallados y permiten velocidades de transferencia de 1 MB/s, aunque existen versiones que llegan hasta los 8 MB/s.

Los conectores tienen dos lados de conexión (macho y hembra) permitiendo diversas estructuras topológicas (bus, estrella y combinaciones). Los hay de dos tipos: americano (24 pines) y europeo (IEC-625.1, 25 pines).



Figura 1: Cable GPIB

Para que el bus GPIB alcance la velocidad de transmisión para la que fue diseñado (hasta 8 MB/s), deben cumplirse los siguientes requisitos:

- Puede haber un máximo de 15 dispositivos conectados al bus, y al menos dos tercios de ellos deben estar encendidos.
- La separación máxima entre dos dispositivos es 4 m, y la separación promedio en toda la red debe ser menor de 2 m (National Instruments comercializa un extensor de fibra óptica (GPIB 140 y 140/2) que permite alcanzar una longitud de hasta 2 km).
- La longitud total de la red no debe exceder los 20 m.

Un sistema típico constará de un ordenador con una tarjeta controladora GPIB, más los instrumentos (compatibles con IEEE 488, obviamente). Existen tarjetas GPIB para prácticamente todos los ordenadores presentes en el mercado (PC, Macintosh, estaciones Sun, Silicon Graphics, DEC Alpha, HP RS/6000, etc.). En el caso concreto del PC, las controladoras GPIB pueden conectarse al bus ISA, PCI, PCMCIA (portátiles), USB, Ethernet, Firewire, y los puertos serie y paralelo. Existen asimismo adaptadores para los estándares de comunicación RS-232 y RS-485.

Muchos fabricantes ofrecen adaptadores que permiten conectar instrumentos GPIB a otros como Ethernet, operando de una forma más o menos transparente para el programador. Asimismo, el

estándar VISA (Virtual Instrument Standard Architecture) ha simplificado considerablemente el desarrollo de software al hacer bastante transparente el bus físico de conexión.

2.3 GPIB o Ethernet

Ninguna de las tecnologías actuales ha conseguido una penetración apreciable en el sector del control de instrumentos de medida, donde GPIB sigue siendo el estándar dominante. Esto se debe al hecho de que ninguno de los competidores se haya impuesto como predominante; la rapidez con la que cambia la tecnología PC, en contraste con la de los instrumentos de medida, que tienen un tiempo de vida más largo y el elevado coste de dichos instrumentos, que les confiere una gran inercia para actualizarse. Por otra parte, posee la ventaja de brindarle seguridad global al sistema, y, como factor psicológico en contra de Ethernet el que los ingenieros tuvieran que recurrir a los administradores de red para montar su sistema. Otro factor determinante es la latencia, una conexión GPIB tiene menos demora que una conexión LAN, por ejemplo, una solicitud de identificación (*IDN?) es más rápida en GPIB. Esto es debido a que el tiempo que toma una conexión LAN depende de muchos factores como la infraestructura de la red y la velocidad de las interfaces.

3. FORMATOS DE INTERCAMBIO DE DATOS

La funcionalidad del estándar GPIB ha evolucionado a lo largo del tiempo y se encuentra descrito en las siguientes especificaciones:

- IEEE 488.1 (1975): Especificación que define las características de nivel físico (mecánico y eléctrico), así como sus características funcionales básicas.
- IEEE 488.2 (1987): Especificación que define las configuraciones mínimas, los comandos y formatos de datos básicos y comunes a todos los equipos, el manejo de errores y los protocolos que se siguen en las comunicaciones.
- SCPI (Standard Commands for Programmable Instrumentation): Especificación construida sobre el estándar IEEE 488.2 que define una estructura de comandos estándar aceptados por múltiples instrumentos de muchos fabricantes como: Agilent Technologies, CEC (Capital Equipment Corporation), IOTech hardware, Keithley, MCC (Measurement Computing Corporation) y NI (National Instruments).
- VISA: Especificación definida por Agilent y National Instrument. Es una biblioteca que puede ser usada para desarrollar aplicaciones y drivers de I/O de forma que software de diferentes empresas puedan trabajar conjuntamente sobre el mismo sistema y que pueden ser instalados en conjunción con drivers VXI plug&play utilizando simultáneamente varios medios de comunicación (GPIB, VXI, RS232, LAN, etc) y en aplicaciones desarrolladas con diferentes lenguajes (C, C++, VisualBasic, etc.).

3.1 SCPI

A pesar de los estándares IEEE 488.1 y 488.2, existía libertad para que cada fabricante eligiera los comandos de sus instrumentos. La norma SCPI aparece en 1991 para conseguir una estandarización de los comandos de control y el formato de los datos de los instrumentos. El objetivo es que,

independientemente del fabricante, equipos que tienen la misma funcionalidad respondan de igual forma a un conjunto estándar de comandos. [2]

La norma SCPI reduce los costes de desarrollo y mantenimiento de programas de control de sistemas de instrumentación para pruebas automáticas. Esto se consigue ya que:

- facilita el aprendizaje y uso de los comandos y los datos
- facilita el desarrollo y mantenimiento de los programas
- posibilita la sustitución de equipos con los mínimos cambios de software.

Para conseguir estos objetivos la norma establece la sintaxis y los formatos de los mensajes para que instrumentos con la misma funcionalidad o instrumentos del mismo tipo utilicen los mismos comandos. Por ejemplo, los comandos para medir una frecuencia utilizando frecuencímetros de distintos fabricantes serán los mismos. Además, la medida de la frecuencia con otro instrumento que lo permita, por ejemplo un osciloscopio digital o un multímetro, también utilizarán los mismos comandos.

Estructura de comandos SCPI

SCPI especifica un "lenguaje" que tiene por objeto que se utiliza para el control de instrumentos de prueba. Proporciona una sintaxis común, la estructura de mando y formato de intercambio de datos.

Los comandos SCPI se escriben como texto ASCII, y tienen una estructura jerárquica por niveles, separados por dos puntos. Cuando se escribe, las tres primeras letras se escriben en mayúsculas, ya que son los que definen la acción.

HORizontal:MAIn:SCALe <valor>

La estructura de comando SCPI se organiza para dar una topología de árbol con ramas que sale de un tronco central o zona de origen. Los comandos SCPI relacionados se agrupan en ramas. Cada palabra clave en el comando se llama un nodo. El primer nodo se denomina el nodo raíz. Como se muestra a continuación:

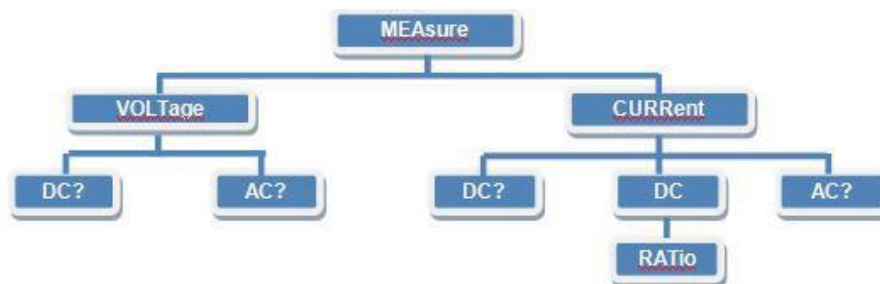


Figura 2: Diagrama de nivel de comandos SCPI

SCPI no define el enlace de comunicaciones, a pesar de que fue diseñado originalmente con el uso GPIB en mente. Sin embargo, en estos días con una variedad mucho más grande de interfaces y enlaces

disponibles para el control de instrumentación es ampliamente utilizado con otras normas incluyendo RS-232, Ethernet, LXI, USB, PXI, etc.

Tipos de comandos SCPI

Dentro del conjunto de comandos SCPI, hay dos tipos de comandos que pueden ser emitidos:

- **Configurar el funcionamiento:** una operación de conjunto que ordena al instrumento ajustarse de una forma particular. Por ejemplo, podría pedir a una fuente de alimentación que esté ajustado a la tensión, límite de corriente dada, etc.
- **Funcionamiento:** consulta problemas de funcionamiento, el instrumento debe responder con una lectura. Las operaciones de consulta se terminan con un signo de interrogación "?".

Algunos comandos del instrumento pueden actuar como un conjunto y operación de consulta. Un ejemplo puede ser un CAL? comando que podría iniciar una rutina de auto-calibración y a continuación, devolver los resultados de la calibración, aunque esto sería dependiente del instrumento.

4. ENTORNOS DE PROGRAMACIÓN

La primera alternativa de programa que se tiene en cuenta (y, en ocasiones, la más práctica), es el uso de software propietario desarrollado por los mismos fabricantes del instrumento. Por ejemplo, el programa gratuito Intuilink de Agilent Technologies, o el programa de pago Wavestar para los osciloscopios de Tektronix Inc.

La ventaja evidente de estos programas es que pueden ser empleados nada más conectar los instrumentos, y proporcionan ya hechas las funciones más comunes que uno puede desear realizar, sin necesidad de programar. Las desventajas son también claras: por tratarse de software cerrado, sólo puede ser usado para la tarea para la que fue diseñado, y además son imposibles de integrar con otros programas.

4.1 LABVIEW de National Instruments

LabView está orientado a programación gráfica en vez de al desarrollo de código C. NI (National Instruments) mantiene la compatibilidad con otros compiladores de propósito general, como las distintas suites de Microsoft Visual Studio (Visual Basic, Visual C++, .NET, etc), a través de su producto Measurement Studio. Los aspectos más destacables de este producto son:

- Bibliotecas completas para la comunicación entre dispositivos (puerto serie, paralelo, GPIB, TCP/IP, etc).
- Especial facilidad para el desarrollo de interfaces gráficas adaptadas a los instrumentos de medida (dispone de elementos para mostrar formas de onda, conmutadores, potenciómetros, etc)
- Soporte para los drivers IVI de numerosos instrumentos (extensión .fp).

- Posibilidad de ofrecer el programa final desarrollado mediante una aplicación de instalación.

4.2 MATLAB con Instrument Control Toolbox

MatLab, que fue en origen un conjunto de rutinas para manipulación de matrices, ha evolucionado con el tiempo para convertirse en un entorno de programación de propósito general con gran potencia matemática y aplicabilidad en muchos ámbitos de la ciencia y la ingeniería, gracias a sus módulos de extensión (toolboxes) de procesamiento de señales, control, ecuaciones diferenciales, y un largo etcétera. En muchos casos las toolboxes representan el estado del arte en programación numérica en sus respectivos campos. [4]

Recientemente, los creadores de MatLab, Mathworks, emprendieron una línea clara de ampliación de su mercado hacia la conexión de hardware con el PC, distribuyendo toolboxes para el control de tarjetas de adquisición, generación de código y emulación de DSPs, xPC (control remoto de PCs para operación en tiempo real), adquisición de imágenes, etc. La primera versión de la toolbox específica de control de instrumentos de medida apareció en 2001, y la versión disponible en la actualidad está suficientemente depurada para ser competitiva con productos más maduros como CVI. Existe también una toolbox de adquisición de datos con soporte para tarjetas de entrada y salida de señales analógicas y digitales, dentro de esta misma línea de productos.

La toolbox ha sido diseñada de forma que el acceso a los instrumentos de medida se realiza mediante objetos, que pueden ser de dos clases: interfaz o de dispositivo.

Mientras que el primer método equivale a la programación con comandos SCPI, el segundo enfoque permite explotar toda la potencia de IVI. Para ello es preciso disponer del driver de MatLab correspondiente (extensión .mdd), obtenible en la página de Mathworks o bien usar una utilidad llamada makemid para convertir (o “envolver”) el driver IVI ya instalado en el sistema. La utilidad midedit permite ver y modificar todos los objetos definidos en los aparatos, sus propiedades, y las funciones ejecutables.

La selección del lenguaje de programación, MatLab, se basa en las prestaciones que ofrece este software, ya que este cuenta con una aplicación que se especializa en el control de instrumentos y su reconocimiento en esta área es global.

5. MODELO DE CONTROL REMOTO PROPUESTO

A partir de las características analizadas de cada capa del modelo, el sistema propuesto será el siguiente (Figura3):

- PC: Computadora con tarjeta GPIB o en lugar de esta se usa un conversor USB/GPIB.
- MatLab: Programa interactivo que ofrece un lenguaje de programación de alto nivel y unos conjuntos de bibliotecas matemáticas y gráficas muy extensas.
- Instrumentation Control ToolBox: Conjunto de funciones MatLab que permite gestionar los driver GPIB o Visa, y establecer comunicación con los equipos.

- Driver VISA: Estándar de driver que ofrece un conjunto de bibliotecas estandarizadas que permiten integrar equipos conectados por diferentes medios de comunicación.
- Driver GPIB (SCPI): El computador interactúa con el bus GPIB a través de una tarjeta de control hardware que resuelve y atiende los dos protocolos anteriores. El propio fabricante ofrece una interfaz software implementada por un conjunto de funciones que permiten el acceso de los programas a la funcionalidad del bus.
- Nivel Operativo (IEEE 488.2): El protocolo operativo básico dentro del que se encuadra el intercambio de información, datos e instrucciones básicas de control.
- Nivel Físico (IEEE 488.1): La comunicación entre los equipos se basa en un bus físico, compuesto por un conjunto de líneas con niveles lógicos bien definidos y con protocolos de comunicación basados en los estados lógicos de las líneas.
- Equipo: Instrumento del entorno que se controla.

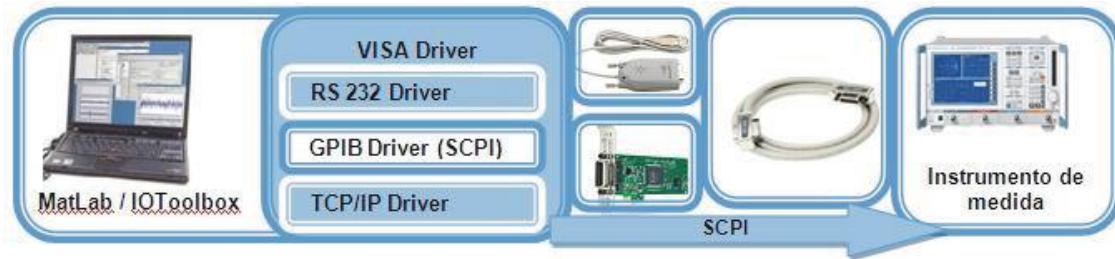


Figura 3: Componentes del modelo de control mediante GPIB.

CONCLUSIONES

El desarrollo de las tecnologías encaminadas a la adquisición de datos conlleva a la creación de mecanismos que faciliten este proceso, como lo es el control remoto de instrumentos de medida a través de la PC. Como se abordó en este trabajo existen diversas formas de realizar este control, pero la selección del mecanismo propuesto está basada en las características de un laboratorio que necesite, sobre todas las cosas, seguridad y robustez. Con este propósito operan GPIB y MatLab, no mejores que sus competidores en otras cuestiones, pero predominantes en estas dos. Este modelo permite interconectar varios equipos que se encuentren en una misma habitación y responde a un estándar aceptado por la mayoría de los fabricantes (cualquier instrumento “entenderá” los comandos). Además posee una velocidad de intercambio de datos suficientemente alta para que la transferencia de los paquetes de datos sea inapreciable para el usuario. Dado que los sistemas de adquisición de datos basados en PC estarán siempre conectados al PC, las mediciones se pueden visualizar en el monitor del PC en tiempo real, es decir los resultados son visibles al instante. Algunos entornos de programación permiten a los usuarios crear interfaces personalizadas para controlar el dispositivo de medición y mostrar los resultados.

REFERENCIAS BIBLIOGRÁFICAS

1. “¿Qué es GPIB?”, disponible en: <http://www.ni.com>. (Referencia en Internet)
2. “SCPI”, disponible en http://www.radio-electronics.com/info/t_and_m/scpi-standard-commands-for-programmable-instrumentation, 2010. (Referencia en Internet)
3. “Instrument Control Toolbox”, MatLab.
4. “Ventajas del registro de datos basado en PC”, disponible en <http://www.todoproductividad.blogspot.com>, 2010. (Referencia en Internet)
- 5.C.A.: “Instrumentación Electrónica”, Universidad de Oviedo.
- 6.SECO GRANJA, FERNANDO: “Conexión de Instrumentos de Medida con GPIB”.