

HERRAMIENTA CLIENTE/SERVIDOR DESTINADA A LA ENSEÑANZA SEMIPRESENCIAL DEL LENGUAJE DE CONTROL DE INSTRUMENTACIÓN SCPI

M. SAMPLON, J. MUR, S. ARTAL.

Departamento de Ingeniería Eléctrica. Escuela Universitaria de Ingeniería Técnica Industrial de Zaragoza (EUITIZ). Universidad de Zaragoza. España.

El estándar SCPI establece un lenguaje homogéneo de mandatos para el control de instrumentación de amplia difusión. La necesidad de disponer de instrumentos capaces de responder a este lenguaje, generalmente de gama media-alta, dificulta su incorporación práctica en un curso de instrumentación. En este documento se presenta una metodología de enseñanza del mismo a través de TCP/IP que permite afrontarla con un número pequeño de instrumentos además de dar opción a su enseñanza semipresencial.

1. Introducción

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 (GPIB, RS232, USB, Ethernet), de arquitecturas de instrumentos (VXI, PXI, LXI), así como una amalgama de capas de software (VISA, IVI), lenguajes de comunicación (SCPI) y entornos completos de desarrollo de sistemas de control de instrumentación (LabVIEW, LabWindows) [1]. Este panorama tecnológico resulta apabullante para quien se enfrenta por primera vez a él. Sin embargo diferentes motivos, como la implantación de protocolos de calidad en el proceso productivo entre otros, apuntan a la importancia que tiene para un ingeniero una formación específica en este campo. Es por ello que cada vez está siendo más habitual en las asignaturas destinadas a la metrología, y con mayor énfasis en las dedicadas a la metrología de magnitudes eléctricas, la inclusión dentro de sus temarios de sistemas destinados a la realización de medidas de forma automatizada con apoyo de un ordenador, y así se ha hecho dentro de la asignatura de Electrometría, obligatoria (troncal) de tercer curso de la titulación de Ingeniero Técnico Industrial, especialidad de Electricidad, de la Escuela Universitaria de Ingeniería Técnica Industrial de Zaragoza.

Con el objeto de dotar al estudiante de una formación suficiente en este campo que le permita poner en marcha de forma autónoma un sistema de este tipo, y atendiendo al tiempo limitado que se dispone para ello, de entre las diferentes aproximaciones tecnológicas se ha optado por mostrar al estudiante la forma de resolver el problema de la comunicación con los instrumentos usando un lenguaje de programación de alto nivel y el lenguaje SCPI [2]. La comunicación con los dispositivos se realiza fundamentalmente a través de una interface GPIB junto con una biblioteca de llamadas compatibles con la NI-488.2. En caso de emplear la conexión RS232 se emplean las funciones estándar del sistema operativo.

El lenguaje SCPI nace ante la necesidad de estandarizar la comunicación con los dispositivos de forma que instrumentos con funcionalidades similares respondan a los mismos mandatos independientemente de su tipología y fabricante. En este sentido, una fuente de alimentación con capacidad de monitorizar la tensión continua que está generando y un multímetro digital obedecerán al mismo mensaje de solicitud de medida de tensión DC, lo que facilita y acelera la puesta a punto del sistema de medida. Los mensajes SCPI se codifican como cadenas de caracteres ASCII en un formato legible para el programador.

Se ha evitado el uso de VISA porque, pese a presentar una interface homogénea con respecto a los canales de comunicación, incorpora una complejidad estructural añadida. Resulta más interesante que el estudiante entienda la comunicación utilizando los recursos básicos antes de hacerle afrontar una nueva capa de abstracción. Por unas consideraciones similares tampoco se ha empleado IVI, que además implica la necesidad de disponer de drivers específicos para cada dispositivo. También se ha huido de los entornos completos de desarrollo dado que adolecen de su propia complejidad intrínseca, constituyen software privativo, son muy caros, máxime considerando la necesidad de disponer de varias licencias para dotar a un laboratorio de prácticas, resultan excesivos para sistemas sencillos, que por otra parte suelen ser los que se necesitan en la mayor parte de las ocasiones, y además crearían la tendencia en el estudiante a optar por su uso en su ejercicio profesional en detrimento por desconocimiento de soluciones basadas en software libre que conllevarían globalmente un desembolso económico mucho menor.

2. Dificultades para la impartición de formación práctica

La formación en programación de ordenadores en un lenguaje de alto nivel viene generalmente cubierta por sus propias asignaturas en el plan de estudios de una titulación de ingeniería -en nuestro caso existe una específica en el primer año- por lo que la asignatura de Electrometría se centra entre otros aspectos en la formación en el lenguaje SCPI. En lo que sigue denominaremos como *guión SCPI* al conjunto ordenado de mandatos SCPI que se envía a los instrumentos para la realización de una tarea específica. El lenguaje SCPI no es un lenguaje de programación y en ese sentido no incorpora mandatos de control de flujo de ejecución de ahí que se opte por la denominación *guión* antes que *programa*.

La inclusión de docencia práctica sobre este lenguaje de comunicación presenta, no obstante, dos problemas principales:

- El estándar define un lenguaje que en definitiva se materializa mediante cadenas de caracteres. La forma habitual y más flexible, aunque no única, de envío y recepción de esas cadenas hacia y desde el dispositivo consiste en la redacción de un programa de ordenador en un lenguaje de alto nivel, típicamente C ó C++. Una práctica de SCPI pasaría por la elaboración por parte del alumno de dicho programa. Aún partiendo de plantillas de código C esto podría convertir la práctica destinada a SCPI en una práctica de una asignatura de lenguajes de programación de ordenadores antes que de Electrometría. Las dificultades que pudiera tener el estudiante con el entorno de desarrollo o con el código C, le desviaría del verdadero propósito de la práctica y apantallarían el sentido de control de instrumentación que incorpora el SCPI.
- Dada la relativa complejidad del estándar normalmente está implementado en dispositivos de gama media-alta, de forma que resulta prohibitivo equipar siete puestos de prácticas (los usuales en nuestros laboratorio) con este tipo de instrumentos.

Ambos problemas pueden ser abordados mediante simulaciones de instrumentos mediante software [3], si bien esta metodología no puede compararse al uso de dispositivos reales. Por otra parte resulta significativo que durante la elaboración de un guión SCPI el estudiante dedica la mayor parte del tiempo al diseño, consulta de manuales y edición del mismo, siendo muy inferior el tiempo dedicado a su ejecución. Dado que es en este último estadio donde realmente se emplea el instrumento, es innecesario que disponga de él durante todo el tiempo de prácticas. Es por ello que una forma alternativa de solventar las dificultades indicadas consiste en disponer de un sistema de multiplexación de los instrumentos entre los estudiantes de forma que éstos elaboren el guión SCPI de forma independiente en un editor de textos y únicamente se conecten a los instrumentos en el momento de ejecutarlo para comprobar su funcionalidad. Es durante la ejecución del guión y sólo entonces cuando el estudiante dispone del sistema de instrumentación de forma exclusiva.

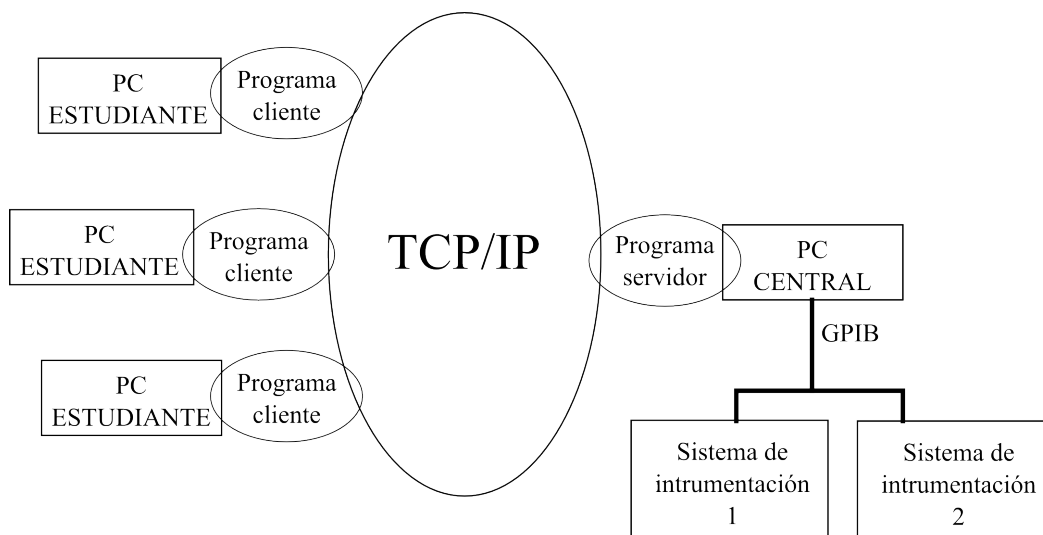


Figura 1. Estructura del sistema.

3. Sistema cliente/servidor. Requerimientos.

El sistema de multiplexación puede construirse mediante un par de aplicaciones cliente/servidor comunicándose a través del protocolo TCP/IP, tal como recoge la figura 1. El estudiante se sienta ante un PC en el que se ejecuta la aplicación cliente, encargada de enviar el guión SCPI a la aplicación servidor que se ejecuta en el PC central que a su vez tiene conectados varios *sistemas de instrumentación*. La comunicación se lleva a cabo mediante TCP/IP lo que permite establecer el enlace a través de una LAN o de Internet. Por sistema de instrumentación entendemos uno o varios instrumentos que realizan conjuntamente un experimento. Un sistema de instrumentación viene identificado mediante un nombre específico o *etiqueta*. Dentro de cada sistema de instrumentación, cada instrumento viene igualmente identificado por una etiqueta.

3.1. Aplicación cliente.

El estudiante emplea el programa cliente para seleccionar el sistema de instrumentación sobre el que quiere actuar y para la edición del guión. Éste está compuesto de mandatos SCPI y metamandatos que dan un cierto control sobre la ejecución. El texto se interpreta línea a línea. En concreto cada línea del guión consiste en un mandato al sistema de instrumentación o bien un metamando. En el primer caso, la línea comienza con la etiqueta que identifica el instrumento destinatario del mandato seguido, tras uno o varios espacios en blanco, del mandato SCPI en sí. Los metamandatos comienzan con el carácter '#', lo que permite identificarlos.

Mediante metamandatos se pueden incorporar estructuras de control como bucles y evaluación de condiciones. Sin embargo implica una complejidad añadida a la interfaz que confunde al estudiante y lo distrae del objetivo de revisar/experimentar con los mandatos SCPI. La aplicación que se ha desarrollado tiene un objetivo docente, y se ha primado ese criterio frente a la aplicabilidad práctica o la optimización del control de instrumentación, aunque pueda usarse para ello. Por ello únicamente se han incorporado tres metamandatos:

- **Pausa.** Fuerza una pausa en la ejecución con propósitos de depuración. La pausa dura un máximo de 30 segundos para evitar que el estudiante bloquee accidentalmente el uso del sistema de instrumentación. La sintaxis es: #PAUSA
- **Comentario.** Fuerza a que el resto de la línea se ignore. Permitiendo la inclusión de comentarios en el guión. La sintaxis es: ###
- **Lectura de datos desde instrumento.** Cuando un mandato SCPI realiza una pregunta a un instrumento (por ejemplo le pregunta el valor de la tensión que está midiendo), el dispositivo coloca la respuesta en su buffer de salida, pero no la envía a través del canal de comunicación

hasta que se hace la llamada correspondiente a la biblioteca que gestiona la comunicación a través de la interface correspondiente (por ejemplo, *ibrd* para el uso de la interface GPIB mediante la biblioteca NI-488.2). Este metamandato fuerza la realización de esa llamada. Dado que la solicitud de datos se hace a un instrumento concreto, el metamandato va precedido de la etiqueta identificativa del instrumento. La sintaxis es en este caso: `ETIQUETA_INSTRUMENTO #LEE_DATOS`

La aplicación cliente tiene las siguientes funciones:

- Es un editor de texto ASCII básico para la escritura del guión.
- Gestiona el lado cliente de la conexión TCP/IP
- Cuando el usuario ordena la ejecución del guión realiza las siguientes acciones:
 1. Establece la conexión con el servidor.
 2. Identifica al usuario.
 3. Solicita al servidor el sistema de instrumentación que quiere usar.
 4. Si el servidor indica que el sistema está ocupado, informa al usuario, espera un intervalo de tiempo aleatorio (de unos pocos segundos) y vuelve a realizar la solicitud. Este paso se reitera hasta tener acceso al sistema de instrumentación.
 5. Analiza la primera línea de código. Si está en blanco o es un comentario la ignora. Si es el metamandato de pausa, la ejecuta. En cualquier otro caso envía al servidor un mensaje compuesto por la etiqueta del sistema de instrumentación, la etiqueta del instrumento y el mandato SCPI.
 6. Recibe la respuesta del servidor, normalmente un mensaje de confirmación de la ejecución sin interés para el usuario. En caso de que la respuesta sea el resultado de una petición de lectura de datos, se muestra la respuesta en pantalla.
 7. Se repiten los pasos 5 y 6 hasta procesar todas las líneas del guión.
 8. Cierra la conexión con el servidor.

3.2. Aplicación servidor.

La aplicación servidor tiene las siguientes funciones:

- Registra los sistemas de instrumentación que tiene activos.
- Gestiona el lado servidor de la conexión TCP/IP. Mantiene un registro de las conexiones activas.
- Asigna un sistema de instrumentación concreto a un usuario en caso de que esté libre. En caso contrario envía un mensaje indicando que el sistema está ocupado.
- Reinicia los instrumentos antes y después de la ejecución del guión.
- Actúa de pasarela entre los mensajes que llegan a través del protocolo TCP/IP y la red de instrumentación (típicamente GPIB), encaminándolos al sistema de instrumentación e instrumento adecuado.
- Tras la ejecución de cada mandato SCPI envía siempre una respuesta al cliente. Esta respuesta consistirá en un mensaje de confirmación de la ejecución o bien la información que se ha solicitado a un instrumento a través del metamandato `#LEE_DATOS`

4. Detalles de implementación

Ambas aplicaciones se han escrito en lenguaje C++ y buscando herramientas libres para su desarrollo. Bajo Windows se ha empleado el IDE wxDev-C++ y para Linux el IDE Anjuta. En ambos casos el compilador ha sido GNU GCC. Con el fin de mejorar la portabilidad de código entre Windows y Linux se ha empleado la biblioteca wxWidgets [4].

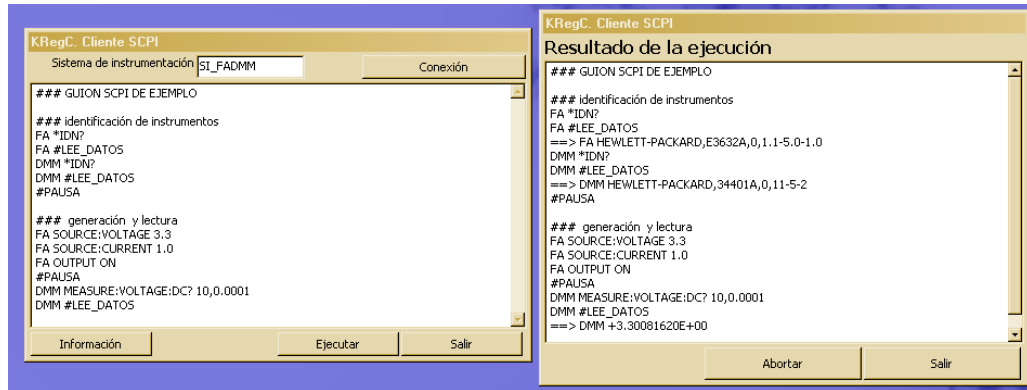


Figura 2. Aplicación cliente. Ventana de edición del guión y ventana de resultados de la ejecución

La principal dificultad en la programación de la aplicación cliente estriba en la interface de usuario y puede resolverse con facilidad mediante wxWidgets, que incorpora una clase que es un editor de textos en sí mismo así como una clase para el lado cliente de una comunicación TCP/IP.

La aplicación servidor se ha estructurado en tres capas:

1. Una capa encargada de la comunicación TCP/IP, además de la interface de usuario que en este caso es trivialmente sencilla. Esta capa se ha resuelto empleando las clases correspondientes de la biblioteca wxWidgets.
2. Una capa encargada del encaminamiento de mensajes recibidos a través del protocolo TCP/IP a los sistemas de instrumentación. Esta capa se ha resuelto mediante una clase gestora de los sistemas de instrumentación cuyas funciones únicamente son las de registro de sistemas activos y encaminamiento de mensajes recibidos al sistema de instrumentación correspondiente. Ante la recepción de un mensaje esta capa controla si el sistema destinatario existe, pero no se ocupa de si está actualmente en uso o no.
3. Una capa encargada de representar al sistema de instrumentación en sí. Cada sistema de instrumentación se incorpora al servidor mediante la escritura de una clase que lo representa y que deriva de una clase virtual que establece la interface de programación y ofrece los servicios básicos. Esta clase tiene los siguientes cometidos:
 - Establece el estado de libre/ocupado del sistema de instrumentación. Lo asigna a un usuario que lo solicite en caso de estar libre y en ese proceso inicializa el sistema de instrumentación.
 - Recibe todos los mensajes destinados al sistema de instrumentación pero considera únicamente los mensajes que le llegan del usuario asignado en ese momento. Analiza el mensaje y envía el mandato SCPI al instrumento correspondiente. Lee la respuesta del instrumento si la hay y la devuelve hacia arriba en la estructura en capas para ser enviada de vuelta al cliente a través de la conexión TCP/IP.

La figura 2 muestra a la aplicación cliente ejecutando un guión sobre un sistema de instrumentación identificado con la etiqueta SI_FADMM y que está constituido por una fuente de alimentación, identificada con la etiqueta FA, y un multímetro digital, identificado con la etiqueta DMM. La salida de la fuente de alimentación está conectada a la entrada de medida de tensión del multímetro. El guión solicita las cadenas de identificación de ambos instrumentos según el estándar IEEE 488.2, configura la fuente de alimentación para generar una salida de 3.3V y solicita al multímetro que mida esa tensión. Una vez elaborado el guión SCPI en la ventana de edición, los resultados de la ejecución aparecen en una segunda ventana. En la figura se muestran ambas.

5. Un caso real

Describiendo un caso real, durante el curso 2006/2007 y dentro de la asignatura Electrometría que se cita en la introducción, los condicionantes de partida eran los siguientes:

- El número de estudiantes matriculados era de 58. Su conocimiento de lenguajes de programación se ceñía en términos generales, a un nivel básico de Pascal, que se impartía en la asignatura Fundamentos de Informática. Por otra parte no se detectaba un gran interés por la programación en general entre los estudiantes.
- El número de horas asignado para prácticas de laboratorio con SCPI era de cuatro, articuladas en dos sesiones de dos horas cada una. De forma previa se programaron dos horas de revisión del estándar, que se impartieron en formato de clase magistral y sin apoyo de instrumentación.
- La instrumentación de que se disponía susceptible de ser controlada mediante SCPI era la siguiente:
 - 1 fuente de alimentación Agilent/Hewlett-Packard E3632A.
 - 2 multímetros digitales (DMM) Agilent/Hewlett-Packard 34401A.
 - 1 generador de señales Sony/Tektronix AFG310.

Si bien se disponía de algunos otros instrumentos con capacidad de ser controlados por ordenador, ninguno de ellos obedecía al estándar por lo que no se consideraron.

Para una primera sesión de prácticas se plantearon tres sistemas constituidos cada uno de ellos por un único instrumento: Una fuente de alimentación con una carga resistiva conectada, un multímetro con una referencia de tensión AC y de corriente DC conectadas y un generador de señales conectado a un osciloscopio. El osciloscopio tenía propósito de permitir a los estudiante visualizar la señal que estaban generando y no se controlaba por ordenador. A los estudiantes se les solicitaba que realizasen pequeñas tareas sencillas, solubles en la mayoría de los casos con un único mandato SCPI, con objeto de familiarizarse con su uso y los manuales de los instrumentos. A modo de ejemplo la lista de tareas para realizar con el multímetro fue la siguiente:

- Obtener la cadena de identificación del instrumento.
- Configurar el dispositivo para medida de frecuencia.
- Configurar el dispositivo para medida de tensión AC en la escala de más sensibilidad.
- Configurar el filtro lento de señal.
- Realizar una medida de I DC mediante MEASURE.
- Realizar una medida de I DC mediante READ?.
- Realizar una medida de I DC mediante FETCH?.
- Realizar una medida de I DC mediante una señal de trigger proveniente del ordenador.
- Realizar una medida de VAC en el rango de menor sensibilidad.
- Realizar una medida de VAC en el rango de mayor sensibilidad.
- Realizar una medida de frecuencia.
- Hacer una inicialización del instrumento.
- Hacer que el DMM realice una serie de medidas de VAC durante un intervalo y obtener valores máximo, mínimo y promedio.

En la segunda sesión de prácticas se propuso a los estudiantes la resolución de algunos problemas de automatización de medidas más complejos que implicasen el usos de sistemas de instrumentación formados por más de un dispositivo. En concreto uno de los sistemas que se propuso estaba constituido por un generador de señal que excitaba a un condensador. Un multímetro se hallaba conectado en serie con el condensador para medir la corriente circulante por el mismo. Un segundo multímetro estaba conectado en paralelo con el condensador para medir su caída de tensión. Al estudiante se le solicitaba que generase un guión SCPI que al ejecutarse devolviese la información suficiente para poder determinar la capacidad del condensador bajo excitación sinusoidal a diferentes frecuencias

El laboratorio se organizó de la siguiente manera. El PC que actuaba de servidor disponía normalmente de dos sistemas de instrumentación conectados. Una cámara de vídeo reproducía los frontales de los instrumentos en la pantalla de proyección de forma que los estudiantes más alejados tuvieran acceso a ellos durante el proceso de depuración del guión. A los estudiantes se les entregaba al inicio el conjunto de tareas que debían realizar de forma automatizada así como los manuales de los instrumentos para su consulta.

El software descrito se utilizó en cuatro turnos de prácticas con una duración de cuatro horas por turno, contabilizando un total de 16 horas de uso además de las empleadas en el examen final de la asignatura. Durante estas sesiones llegaron a estar conectados hasta 9 ordenadores simultáneamente. La aplicación cliente contaba de un medidor de tiempo en espera, siendo éste el tiempo que el programa servidor mantenía retenido el guión enviado a la espera de que se liberara el sistema de instrumentación al que se dirigía. Dado que cuando se puso en marcha la aplicación, aún se encontraba en fase de pruebas no se pueden aportar datos estadísticos, pero era inferior en todos los casos a 8 minutos frente a las dos horas que duraban cada una de la sesiones.

Se probó asimismo la viabilidad de montar el sistema en el despacho del profesor y permitir a los estudiantes conectarse de forma remota. En este caso el estudiante no disponía de la información de los frontales de los instrumentos pero sí de comunicación con el profesor y soporte por parte de éste a través de un sistema de chat.

6. Conclusiones

Se ha diseñado un sistema orientado a la docencia del estándar SCPI a través de un enlace TCP/IP lo que permite la optimización de recursos (con un pequeño número de instrumentos programables se puede atender a un grupo numeroso de estudiantes) así como soporte para enseñanza semipresencial.

El sistema hace hincapié en el uso de los mandatos SCPI y permite al estudiante centrarse en su utilización evitando la necesidad de programar a bajo nivel el protocolo de comunicación.

El sistema ha demostrado su funcionalidad en pruebas de docencia reales, constituyendo a su vez un sencillo ejemplo de control de instrumentación a distancia y en modo compartido, en donde quedan patentes sus ventajas.

7. Agradecimientos

El presente proyecto se ha desarrollado bajo el Programa de Enseñanza Semipresencial, sección “utilización de herramientas TIC especializadas en metodologías activas” del año 2007 de la Universidad de Zaragoza.

Referencias

- [1] Keithley. *Understanding new developments in data acquisition, measurement and control*. 1ª edición
- [2] SCPI Consortium. SCPI-99. <http://www.scpiconsortium.org/SCPI-99.pdf>.
- [3] Zorzano L., Zorzano A., Zorzano J., y Vicuña J, *Sistema de aprendizaje de programación de instrumentos con comandos SCPI*. TAAE '06 VII Congreso de Tecnologías Aplicadas a la Enseñanza de la Electrónica. Madrid, julio 2006. CD-ROM de Actas del Congreso. ISBN: 84-689-9590-8
- [4] Smart J., Hock K. *Cross-Platform GUI Programming with wxWidgets*. Ed. Prentice Hall, 2006. ISBN: 0131473816