

SISTEMA DE INSTRUMENTACIÓN VIRTUAL DE MUY BAJO COSTE APLICADO EN PRÁCTICAS DE ELECTRÓNICA Y CONTROL

De la Fuente Ruz, M.; Cano Martínez, J.M.; Casanova Peláez, P.;
Abarca Álvarez, A.; Abril Duro, J.

Universidad de Jaén
Dpto. Electrónica
Avda. Madrid 35 23071 Jaén

Tno: (953) 212434
FAX: (953) 212400

RESUMEN

Esta comunicación presenta la experiencia realizada en el Dpto. de Electrónica para la creación de varios módulos de prácticas en base a un sistema de Instrumentación Virtual creado para tal fin. Se pone de manifiesto la posibilidad de integrar estos sistemas en los laboratorios con un coste extremadamente reducido.

1. INTRODUCCIÓN

La mejora constante en la relación precio/prestaciones que en la última década han experimentado los ordenadores personales, y muy específicamente los denominados Compatibles PC's, ha conducido a una extensa implantación de estos equipos en infinidad de campos donde hasta hace unos años resultaba impensable su integración, sobre todo debido al coste que esto suponía.

Hoy en día resulta cada vez más frecuente encontrar en los laboratorios de formación e investigación, un equipamiento donde una de las piezas básicas es el ordenador personal. Esto se debe a que el computador resulta una herramienta extremadamente versátil, puede ser utilizada para realizar cálculos, simulaciones, procesar textos, almacenar y registrar información etc., y por otra parte un sofisticado instrumento de laboratorio o como equipo para control.

Hace bastante tiempo que el ordenador viene siendo utilizado, como "maestro" en sistemas programables de instrumentación, conectado a través de buses estándar "IEEE 488, RS232, VXI" u otros. Sin embargo la ampliación en la potencia de cálculo y visualización de los

ordenadores personales han permitido dar un paso más hasta llegar a los denominados Instrumentos Virtuales. Un Instrumento Virtual [1] es el conjunto formado por un programa *software* mas uno o varios instrumentos electrónicos conectados al computador, que permite controlar y visualizar en el ordenador la captura y transferencia de información desde o hacia los instrumentos. Debe entenderse como instrumentos, no solo los típicos equipos autónomos convencionales cuyo funcionamiento puede ser independiente del computador, sino que se han de considerar como tales las tarjetas electrónicas que se conectan dentro o fuera del computador y que permiten realizar funciones de captación, generación o procesamiento de señales. Hoy en día la idea de instrumento virtual va más allá del simple control, programación y presentación de las medidas realizadas por los instrumentos y/o tarjetas conectadas al ordenador, permitiendo con la interconexión de sistemas simples crear sofisticados instrumentos de control y verificación para tareas específicas.

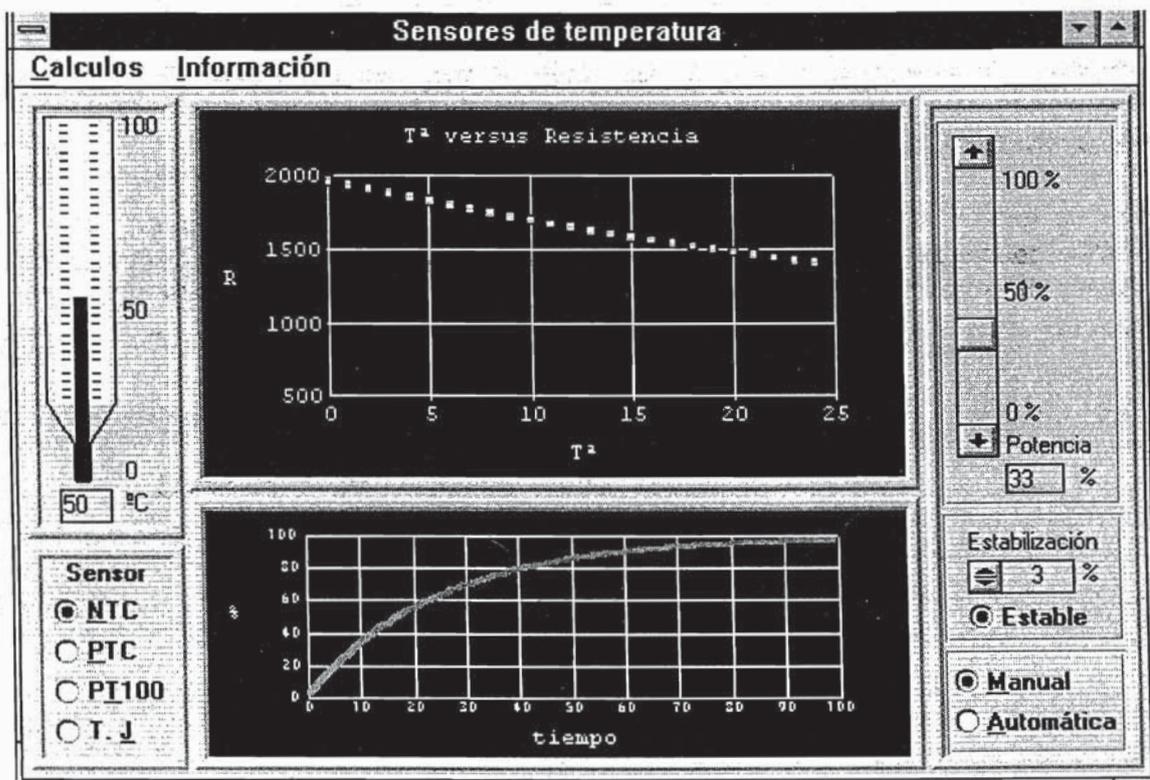


Figura 1. Aspecto de la pantalla gráfica de un instrumento virtual

Un instrumento virtual debe proveer una interfase con el usuario cómoda y de fácil manejo, con posibilidades gráficas (figura 1) y de cálculo potentes. Estas prestaciones determinan unos requisitos mínimos para el computador (Tarjeta gráfica en color, Coprocesador matemático, memoria etc.), así dentro de la gama de ordenadores personales compatibles se precisa como mínimo un computador con una CPU 386 + Coprocesador + Tarjeta Gráfica + Monitor Color + 4 (o más) Mb de RAM. En cuanto a los instrumentos conectados al computador coexisten tres filosofías, en muchos aspectos complementarias, que se pueden resumir en: Instrumentos autónomos conectados a través de un bus estándar "IEEE488, VXI etc", tarjetas insertadas en el los slots del PC y últimamente módulos no autónomos conectados al puerto paralelo o serie del ordenador. Dependiendo de la aplicación resultará más adecuado uno u otro tipo, si bien

se ha de tener en cuenta que el coste decrece muy bruscamente de una opción a otra. Así por ejemplo para adquirir una señal analógica lenta (<50 Hz BW) y con una precisión similar el coste puede ser: 200.000 pts para un instrumento autónomo, 40.000 pts para una tarjeta y unas 10.000 pts para un módulo conectable al puerto del ordenador. Esta comparación de precios pone de manifiesto que es posible disponer de un instrumento virtual a un precio asequible.

2. ELEMENTOS DE UN INSTRUMENTO VIRTUAL

Tal como se ha indicado, un instrumento virtual se compone de dos elementos fundamentales (figura 2) Software (Programa informático que gestiona toda transferencia de información entre el usuario y los instrumentos) y un Hardware (Conjunto de instrumentos autónomos, tarjetas y/o módulos conectados adecuadamente).

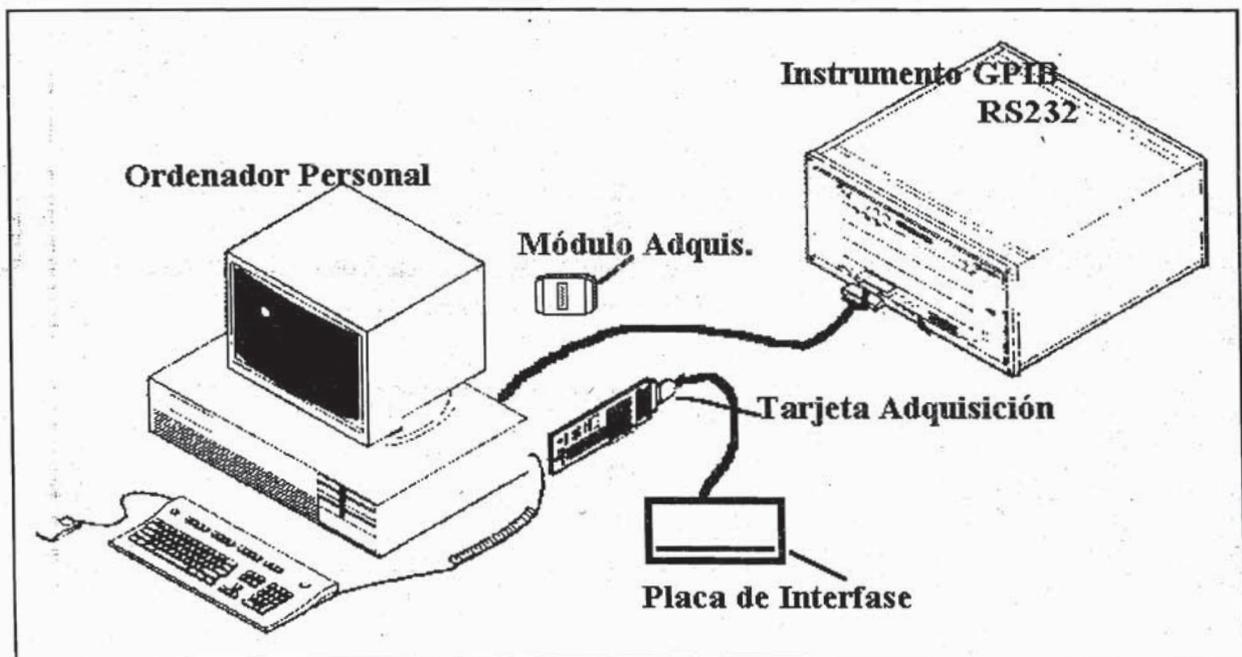


Figura 2. Elementos en un Sistema de Instrumentación Virtual

Software del Instrumento Virtual

Para el desarrollo del software que crea la interfase entre el usuario y el instrumento, se deben marcar unas pautas para elegir el camino apropiado. Más aún si tenemos en cuenta que toda la interfase no es otra cosa que un programa específico que puede desarrollarse con cualquier lenguaje de programación.

Se presentan en este punto varias alternativas:

- i) Desarrollar completamente toda la interfase, generalmente gráfica. Esta opción resulta extremadamente costosa en tiempo y recursos. La creación del programa por medio de un lenguaje de alto nivel (C, Pascal, BASIC) resulta muy complicada sobre todo por la complejidad y poca transportabilidad de las pantallas gráficas esenciales en cualquier

instrumento virtual.

ii) Utilizar herramientas de desarrollo específicamente adaptadas al diseño de instrumentos virtuales, por ejemplo Labview de National Instruments. Por medio de este entorno de programación es posible diseñar un instrumento virtual con pocas nociones de programación y de manera muy eficiente. Esta opción no está exenta de inconvenientes. Por una parte el coste del paquete para el desarrollo es extremadamente alto (400.000 pts), y además es preciso pagar los derechos por cada instrumento virtual diseñado (licencia del runtime). Es necesario indicar también, que solo se pueden manejar instrumentos o tarjetas de los que se disponga un controlador (Driver) suministrado por el fabricante, esto limita el campo de aplicación notablemente, no solo por la falta de controladores para una gran parte de las tarjetas del mercado, sino porque no se dispone de herramientas para diseñar el controlador específico para la tarjeta o instrumento en cuestión. Por último decir que la velocidad de ejecución (proceso) del instrumento se ve muy mermada al tratarse de un programa pseudocompilado.

iii) Emplear un lenguaje de desarrollo a medio camino entre las opciones anteriores. Esto puede conseguirse utilizando un sistema operativo con una interfase de usuario gráfica (GUI) como Windows para los ordenadores PC y un lenguaje de programación Visual como Visual BASIC [2], Visual C u otro. La ventaja de esta alternativa frente a la opción (i), radica en la facilidad y simplicidad para crear la interfase gráfica, parte más compleja de cualquier Instrumento Virtual, además la transportabilidad del instrumento es total e inmediata " si funciona Windows en el equipo del usuario final funcionará el instrumento". El diseño de la interfase gráfica se puede ver muy simplificado si se dispone de controles de usuario (Custom Control) con posibilidades específicas (Gráficos, Botones especiales etc.), y además se disponen de todas las posibilidades que Windows incorpora (intercambio dinámico de datos, enlace de objetos) y todo el soporte para la impresión y manejo de datos del sistema. Esta opción por otro lado resulta mucho más económica, no solo debido al coste del entorno de programación (60.000 pts), sino además porque no es preciso pagar ninguna licencia (runtime) por los instrumentos finalmente desarrollados. Por último indicar que se trata de un sistema abierto, esto quiere decir que es posible desarrollar, por uno mismo, controladores específicos para la tarjeta o instrumento en cuestión sin ningún tipo de limitaciones (solo nuestros conocimientos de programación en C para Windows).

Hardware del Instrumento Virtual

La electrónica necesaria para el funcionamiento del instrumento virtual puede ser implementada a través de tres vías alternativas, que pueden en cualquier caso combinarse:

i) Utilizar un instrumento autónomos programables (osciloscopio, generador señales etc.) que disponga de algún tipo de comunicación con el ordenador personal (RS232, IEEE 488, VXI). Esta opción es la que mejores prestaciones permite obtener, pues la mayor parte del trabajo del Instrumento Virtual es realizado por el equipo en cuestión, y el software solo debe establecer la comunicación adecuada y programar las opciones necesarias del instrumento autónomo. Una simple tarjeta (IEEE 488, RS232 etc.) dentro del ordenador, junto con un controlador (Driver) genérico o específico para el instrumento, son las únicas piezas requeridas para la implementación. Como contrapartida de esta opción cabe indicar que los precios de los instrumentos programables son de 2 a 10 veces superiores a su equivalente no programable, y esto condiciona notablemente su utilización.

ii) Utilizar una tarjeta insertada [3] dentro de los slots de expansión del ordenador. Existen en el mercado multitud de tarjetas con posibilidades muy variadas (A/D, D/A, I/O, Contadores etc.), permitiendo una gran versatilidad a la hora de crear el instrumento, más aún si el diseño de la tarjeta es propio lo que permite adaptarse a las especificaciones concretas y en la mayor parte de los casos reducir notablemente el coste. A la hora de seleccionar la/s tarjeta/s es preciso prestar especial atención a las especificaciones de ésta/s. No se debe atender solo a especificaciones del tipo eléctrico/electrónico (precisión, resolución, número de canales, niveles de entrada/salida etc.) sino también a las del tipo software. Así por ejemplo, una tarjeta de adquisición de datos, que el fabricante indica que puede realizar la conversión en menos de 1 μ s puede ser totalmente inadecuada, si para el proceso de adquisición se precisa de la atención del microprocesador del ordenador personal, ya que éste no puede simultanear a tal velocidad las funciones de adquisición y atención al Instrumento Virtual. Se precisarían en estos casos tarjetas que dispongan de la posibilidad de acceso directo a memoria (DMA), suficientemente documentadas para poder realizar esta programación. El coste de este tipo de tarjetas es muy variado, resultando paradójico que en algunos casos suelen ser más caras que un instrumento convencional que realice las mismas funciones. Esta opción es muy utilizada en la actualidad ya que presenta una buena relación coste/prestaciones.

iii) Emplear sistemas no autónomos conectados a través de algún puerto del ordenador (Paralelo o Serie) [4]. Este tipo de sistemas es bastante novedoso, y permite, básicamente, las mismas funciones que las tarjetas de la opción (ii) si bien la conexión entre el ordenador y el sistema se realiza a través de un puerto estándar del ordenador. Se podría considerar como un sistema híbrido entre (i) y (ii). Una de las ventajas más significativas es su reducido coste, un sistema con 4 canales A/D, 1 D/A y 8 líneas I/O programables, puede estar por debajo de las 5000 pts. Por otra parte no se precisa abrir el ordenador para insertar dentro la tarjeta (origen de multitud de averías graves). El nivel de seguridad de estos sistemas es más alto debido a que entre el Bus del ordenador y las señales medidas existen dos interfases: el puerto y el sistema de adquisición. Como contrapartida cabe indicar que las posibilidades en velocidad de muestreo, precisión, etc. son limitadas y esto es importante a la hora de decidir su implantación.

3. SISTEMA DE INSTRUMENTACIÓN VIRTUAL LIPSOFT (SIV-LIP)

Tras este breve resumen de los aspectos básicos de un instrumento virtual estamos en condiciones de presentar el sistema de instrumentación virtual desarrollado en el Departamento de Electrónica de la Escuela Politécnica de Jaén, para su utilización en prácticas de instrumentación electrónica y control automático. El sistema presentado permite abordar un gran número de prácticas a un coste extremadamente reducido.

El diseño del SIV-LIP se ha realizado teniendo en cuenta las limitaciones de presupuesto. Por ello es por lo que se descartaron en el diseño las opciones triviales (Instrumentos, programables, y paquete de software Labview). La alternativa finalmente elegida ha sido utilizar un software con amplias posibilidades gráficas, junto con un hardware de bajo coste.

Software del SIV-LIP.

El entorno de funcionamiento para el SIV ha sido Windows 3.x de Microsoft. Este sistema permite una interfase gráfica de comunicación con el usuario (fundamental en los Instrumentos Virtuales), a la vez de una elevada potencia en la transferencia de datos entre distintas aplicaciones. Windows además es un sistema operativo conducido por sucesos (Events Driven) esto quiere decir que el flujo de control del programa se encaminará en atención al suceso que acaba de producirse. Así por ejemplo si un usuario pulsa un botón del ratón cuando se encuentra situado sobre un determinado control, se genera un evento, el programador debe crear el código para atender esto, por ejemplo activar un relé, cambiar una línea I/O etc. Como se aprecia esta forma de procesar la información resulta muy conveniente ya que al igual que sucede en los instrumentos convencionales, el usuario puede cambiar en cualquier momento la posición de los botones o ajustes y el instrumento ha de responder adecuadamente. Por otra parte es clara la tendencia hacia sistemas operativos de este tipo, siendo cada vez mayor el número de aplicaciones informáticas que funcionan bajo este sistema.

En cuanto al lenguaje de programación se ha optado por implementar los Instrumentos Virtuales por medio de Visual BASIC (figura 2), junto con controladores (Drivers) desarrollados, como controles de usuario (Custom Control) para el hardware del Instrumento Virtual. La elección de este lenguaje ha sido motivada por:

I.- Permite el diseño de la/s ventana/s de manera gráfica. Basta situar el control (botón, recuadro, etiqueta, etc.) sobre la ventana, dimensionarlo y asignarles propiedades (colores, efectos, etc.) y el control se encontrará listo para funcionar. Por tanto, y de modo similar a Labview el diseño de la interfase se realiza de manera gráfica.

II.- Según el fabricante es el método más rápido para desarrollar aplicaciones bajo Windows y sin necesidad de conocer todos los interiores de la programación para este sistema. Además las instrucciones del lenguaje son las mismas que cualquier BASIC estándar.

III.- Es posible diseñar controladores propios (Custom Control) y añadirlos sin demasiados problemas en el entorno de programación. Se disponen además de bastantes controladores específicos que realizan funciones complejas (gráficos, barras, indicadores etc.), pudiendo adquirir controles más sofisticados a un precio razonable.

Fases de diseño del Instrumento Virtual utilizando SIV-LIP

En primer lugar se debe concretar qué características ha de reunir el Instrumento Virtual, tipo de señales a medir y/o controlar, niveles, canales precisos, posibilidades de interacción con el usuario, etc. . Esta primera fase del desarrollo es fundamental, ya que determinará los elementos físicos, su interconexión y adaptación de señales.

Diseño del Software

El proceso de diseño del programa que creará y gestionará el Instrumento Virtual se puede concretar en las siguientes fases:

1ª- Se bosquejan sobre el papel los controles, paneles, botones, gráficos etc. para cada ventana,

que debe contener el Instrumento Virtual para realizar las funciones encomendadas.

2ª- Ya dentro del entorno de programación de Visual BASIC (figura 3), se van tomando de la caja de herramientas (Tool Box) cada uno de los controles. Se sitúa sobre la ventana y se dimensiona adecuadamente. Se le asignan propiedades iniciales (color, inscripción, efectos, valor inicial, etc.) dependiendo del tipo de control que se trate.

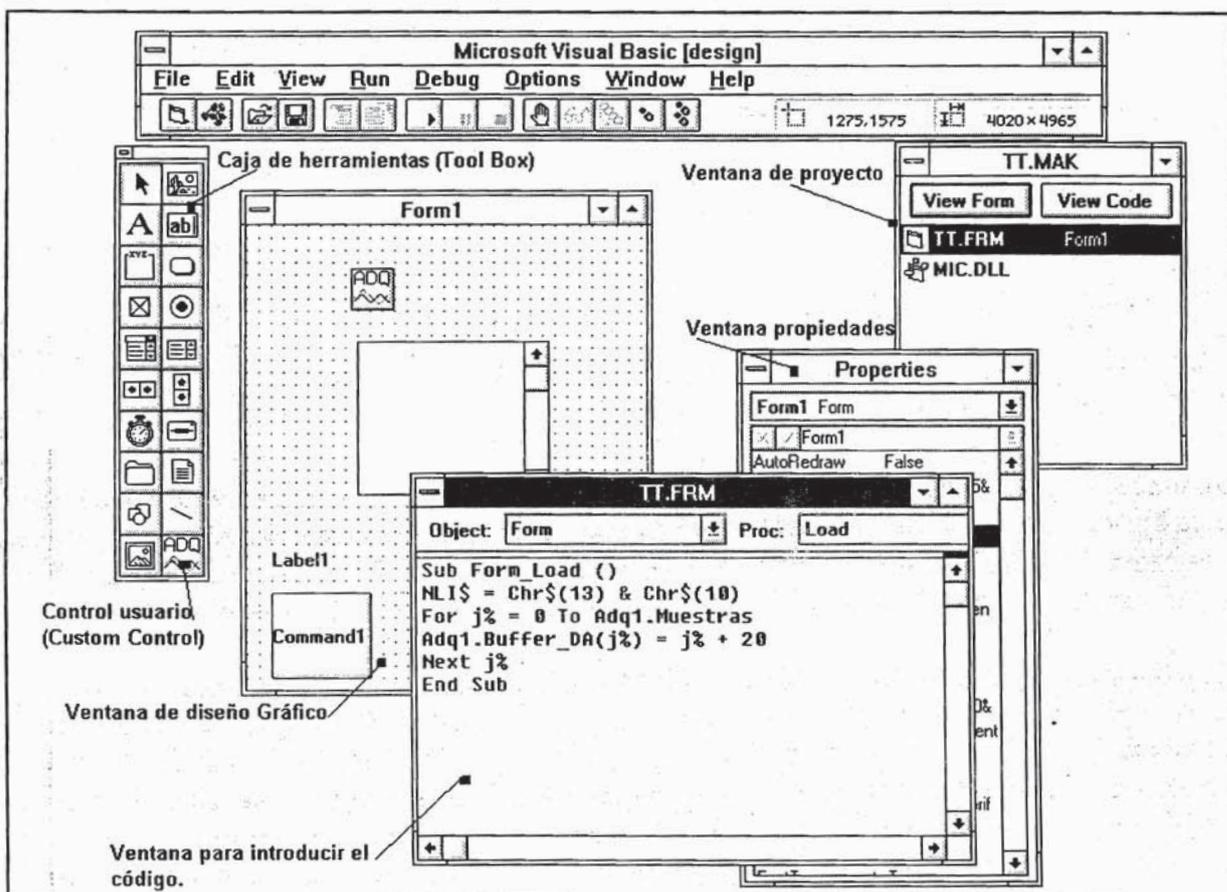


Figura 3. Entorno de desarrollo de Visual BASIC

3ª- Tras el diseño gráfico se procede a introducir el código que hará funcionar al Instrumento Virtual. El diseño del código se realiza según la filosofía de un sistema operativo controlado por eventos (Events Driven) como es Windows. Así si tenemos sobre el panel un conmutador de dos posiciones y el usuario lo activa (hace click con el ratón) tendremos que decidir qué hacer. Para ello y desde el entorno de desarrollo de Visual BASIC, se abre la ventana para insertar las instrucciones requeridas para atender a ese suceso. Se va creando el código correspondiente para cada suceso que deba atender la interfase del Instrumento, teniendo en cuenta que desde el mismo código pueden desencadenarse otros sucesos en otros controles. Se crean simultáneamente todas las rutinas para realizar operaciones específicas de cálculo u otras. Estas rutinas se deberían diseñar según un esquema de arriba abajo (Top-Down) y pueden ser invocadas desde cualquier gestión de sucesos. Se pueden crear rutinas para realizar el filtrado de los datos, su escalado, cálculos complejos (FFT u otro) etc. No existe limitación en cuanto a las posibilidades de trabajo, teniendo en cuenta que a este nivel se trata de un lenguaje de programación convencional.

4ª- Prueba y verificación. Esta es otra característica notable de Visual BASIC, ya que permite de manera cómoda seguir las secuencias y acciones del programa, a la vez que se pueden examinar los valores de las variables por medio de su depurador incorporado (Debugger). Tal y como en cualquier lenguaje, se ha de verificar el comportamiento del software, si bien en este tipo de programación resulta más complejo porque a priori no se pueden establecer las acciones que puede realizar el usuario, y se debe, por tanto, prevenir de manera eficiente cualquier tipo de acción incorrecta.

5ª- Creación de la ayuda. La ayuda en línea, es una más de las posibilidades que permite Windows, y por supuesto Visual BASIC. Para ello se deben crear ficheros de texto con una estructura correcta (normas para la creación de ayudas) y compilarlos por medio del compilador de ayudas. Se ha de incluir en cada control una clave (predefinida en la compilación) para que automáticamente aparezca la ayuda referida a ese aspecto, con solo pulsar una tecla.

6ª- Por último se ha de crear un disco de instalación automático para la aplicación final. La creación de este disco de Setup resulta cómoda si se utiliza otra herramienta de Visual BASIC. Este programa selecciona y comprime los archivos de la aplicación y genera los discos para la instalación de la aplicación dentro del entorno Windows.

Diseño del Hardware

Se han desarrollado un conjunto de módulos que conectados a través del puerto paralelo del ordenador permiten la adquisición y generación de señales, tanto analógicas como digitales. Los elementos diseñados son 5, cuyas características básicas pueden resumirse como sigue:

I. Módulo LS-ADQ. Contiene 4 canales de conversión A/D y 1 canal D/A ambos de 8 bits, unipolares y con un rango entre 0..2,5V de entrada. Además dispone de 8 líneas de entrada/salida digital que aceptan niveles TTL o CMOS a 5 Voltios.

II. Módulo LS-TIM. Contiene dos temporizadores/contadores programables de 6 décadas 0..999999 que aceptan niveles TTL o CMOS a 5V, y frecuencias de entrada hasta 2Mhz. Además dispone de 8 líneas de entrada/salida digital que aceptan niveles TTL o CMOS a 5V.

III. Módulo LS-IO. Este contiene 16 líneas de entrada salida digital programables (línea a línea) que acepta niveles TTL o CMOS a 5V.

IV. Módulo LS-AD12. Contiene 8 canales(multiplexados) A/D y 1 canal D/A ambos de 12 bits, unipolares y que aceptan niveles de entrada entre 0..4096.

V. Placa de Interfase. Se trata de una placa de propósito general que permite la conexión y preadaptación y protección de las señales, desde el proyecto (práctica) hasta los sistemas 1..4. Esta placa dispone de una fuente de alimentación simétrica $\pm 5V$ y de una zona para situar una placa de prototipos (Board) donde el usuario puede montar la práctica a desarrollar. La conexión entre esta placa y los sistemas 1..4 se realiza por medio de un cable plano.

Los sistemas 1..4 comparten las siguientes características comunes:

- Se conectan al puerto del ordenador por medio de un conector Sub-D de 25 pines macho. En su otro extremo disponen de un conector similar hembra para conectar más módulos o en el último módulo la impresora, si la hubiera.

- Cada módulo es direccionable, para ello basta con cerrar los conmutadores apropiados en la placa de expansión. Se pueden conectar unidades mezcladas de cada tipo simultáneamente. Así es posible conectar hasta 8 tipo 1, 2 tipo 2, 4 tipo 3 y 3 tipo 4, todos ellos en un mismo puerto paralelo y seguir utilizando sin problemas la impresora.

- Las dimensiones de los módulos es muy reducida (6x5x2 cm), tal como se aprecia en la figura 4.

- La conexión de las señales se realiza a través de un conector para cable plano de 20 hilos.

- Cada módulo dispone de un controlador (Driver) tipo .SYS para DOS y su correspondiente control de usuario (Custom Control) para Visual BASIC. La comunicación con los controladores se realiza por medio de instrucciones de alto nivel.

- El coste de cualquiera de ellos es muy reducido, y siempre inferior a 7000 pts.

El diseño de los módulos se ha realizado utilizando dispositivos de muy bajo consumo y que disponen de comunicación por medio de un bus serie estándar (I2C, SPI, MicroWire etc.). Se ha creado a la vez un protocolo específico y patentado para la comunicación a través del puerto paralelo.

4. MÓDULO DE PRÁCTICAS.

No se ha de olvidar que el objetivo final de todo el trabajo es la elaboración de unos módulos de prácticas para las asignaturas de Instrumentación Electrónica (Sensores y Acondicionadores) y Servomecanismos (Control por computador). El diseño del sistema de instrumentación virtual ha estado condicionado a este objetivo primordial, si bien la utilización del SIV-LIP puede ser multidisciplinar.

Módulos para las prácticas de Instrumentación.

Como ejemplo se van a incluir dos de los módulos desarrollados para esta asignatura, que está orientada hacia el estudio de elementos sensores y acondicionadores de señal.

La primera fase en el desarrollo de los módulos es definir claramente qué objetivos formativos se desean cubrir. Atendiendo a estos objetivos se elaboró un guión de trabajo y se fueron ensayando cómo se podría transmitir al alumno más claramente cada uno de los aspectos importantes. En función de esto se diseñó un guión más específico para la práctica. A partir de aquí sólo queda sustituir la instrumentación convencional (osciloscopio, multímetro, generador, etc.) por el instrumento virtual.

Atendiendo al guión establecido se eligen los módulos de adquisición, su interconexión y se bosquejan las pantallas del instrumento virtual. Siguiendo las fases de diseño software ya

comentadas se procede a completar la aplicación. Por último se crea el manual de prácticas definitivo.

En la filosofía seguida para las prácticas, el alumno no sólo ha de manejar la aplicación y realizar las medidas, además debe realizar los montajes de los dispositivos (sensores y/o acondicionadores en cuestión). Esto ralentiza bastante el desarrollo, pero creemos que con ello el alumno fija los conocimientos teóricos sobre la materia.

Práctica: Sensores de temperatura.

Una de las variables físicas que con mayor frecuencia se ha de medir en procesos industriales de control es la temperatura. Existen distintos tipos de sensores de temperatura entre los que cabe destacar las resistencias NTC, PTC y RTD, y los termopares. A continuación se expone una experiencia práctica sobre algunas de las características de estos elementos sensores resistivos.

Objetivos de la práctica.

Se pretende obtener la gráfica resistencia-temperatura de un sensor de temperatura determinado, tomando como referencia una temperatura en base a un elemento sensor patrón. En función de los resultados obtenidos mediante gráficas en el instrumento virtual (figura. 1), se podrá estudiar el comportamiento del dispositivo sensor, comprobando el grado de linealidad del mismo, además de experimentar diseños que mejoren o linealicen la respuesta térmica del sensor.

Desarrollo.

Se realizará un sencillo "pozo térmico" construido con 8 resistencias de potencia que formarán dos cuadrados superpuestos de cuatro elementos cada uno. Se introducirá en el dispositivo bajo test y el sensor patrón de referencia procurando que ambos estén lo más juntos posible.

Para producir un crecimiento continuo de temperatura se aplicará una tensión máxima fija a las resistencias del "pozo térmico". Para un aumento escalonado se aplicarán distintos niveles de tensión.

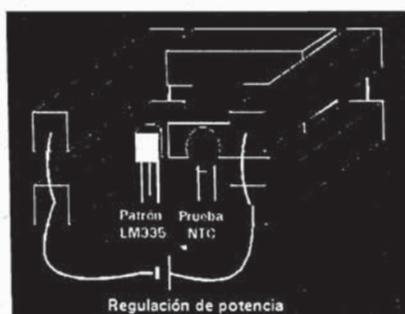


Figura 4. Pozo térmico.

Práctica: Galgas extensiométricas.

Existen distintos métodos sensores de fuerza: reluctivo, piezoelectrico, resistivo, etc., de los cuales cabe destacar el resistivo basado en el uso de las galgas extensiométricas, cuyo principio de funcionamiento es la variación de su resistencia ante una elongación o deformación provocada por compresión o tensión. Dada la buena relación calidad/precio de estos dispositivos, y la posibilidad de aplicarlos en medición de fenómenos físicos que conllevan una deformación (par, presión, peso, tensión, etc.), es uno de los tipos de sensores más extendidos.

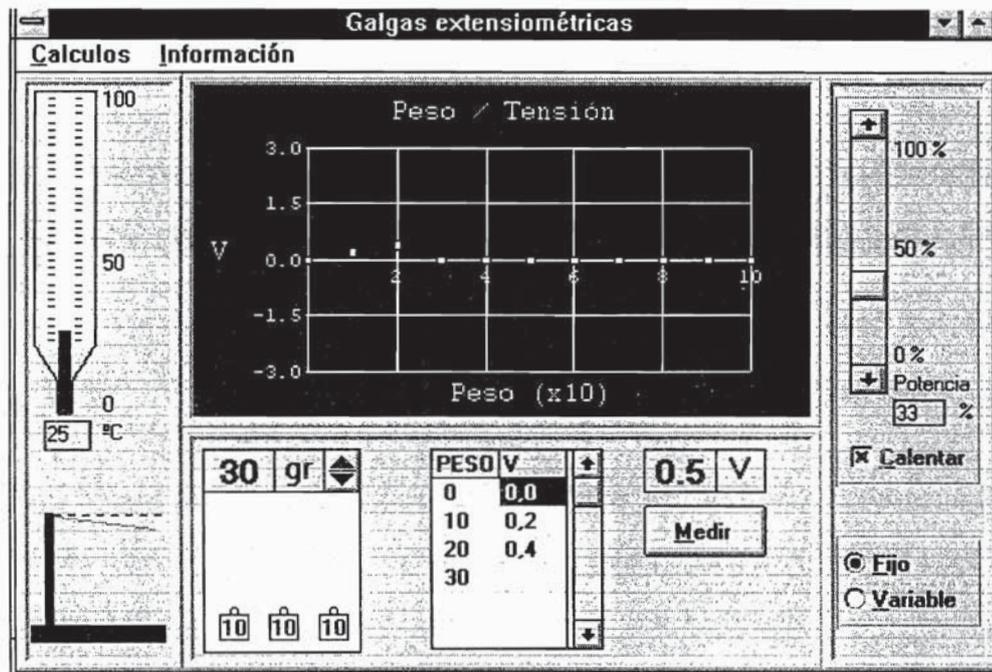


Figura 5. Panel del instrumento virtual.

Como la variación de la resistencia de una galga es pequeña, se recurre al uso del puente de Wheatstone y amplificador de instrumentación para la medición de una tensión proporcional al cambio de resistencia [5].

Objetivos de la práctica.

Se trata de estudiar el comportamiento y operación con galgas extensiométricas mediante su correcta ubicación y aplicación en dispositivos mecánicos básicos, determinación de la deformación unitaria y sensibilidad para distintas configuraciones de los sensores en el puente, así como los efectos de temperatura y su compensación, cuyos resultados se visualizarán mediante tablas de valores y gráficas en el instrumento virtual.

Desarrollo.

Se instalarán adecuadamente las distintas galgas extensiométricas (2+2) en la cara superior e inferior de una barra elástica de metal [6].

Se puede medir la deformación unitaria de una sola galga mediante la aplicación de una

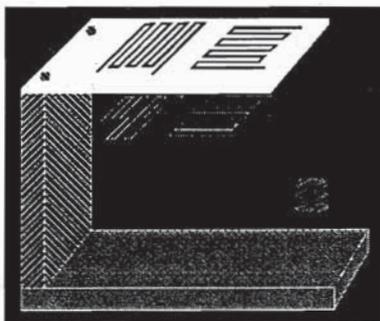


Figura 6. Disposición de las galgas extensiométricas sobre el soporte.

fuerza, además de comprobar los efectos de tensión (alargamiento superior), compresión (acortamiento inferior), efectos de temperatura, sensibilidad, etc. [5] para distintas disposiciones:

- A) Conexión de una galga a un brazo del puente.
- B) Conexión de otra galga (del mismo lado) en el brazo contiguo.
- C) Disposición de dos galgas de cada lado en brazos opuestos del puente.
- D) Conexión de las cuatro galgas dos a dos, en brazos opuestos del puente.

Práctica: Control automático de nivel.

El campo de utilización del Sistema de Instrumentación Virtual presentado es multidisciplinar. Un ejemplo más de su aplicación ha sido la creación de un módulo de prácticas para la asignatura de Servomecanismos (Regulación Automática). En esta práctica se realiza el control en tiempo real de un proceso físico. El proceso controlado está formado por dos depósitos interconectados, una bomba accionada por un motor y un sensor de nivel.

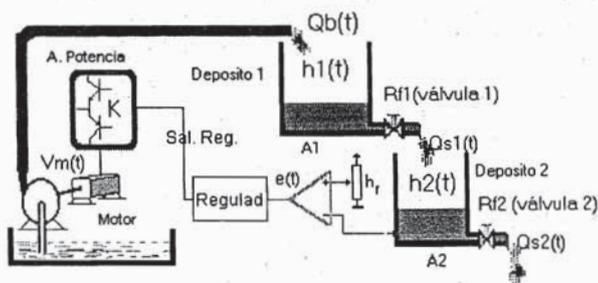


Figura 7. Esquema del proceso controlado

Objetivos de la práctica.

El alumno debe caracterizar los elementos que componen el sistema por medio de ensayos de respuesta temporal. Tras la caracterización el alumno debe diseñar un regulador apropiado para conseguir un comportamiento óptimo del lazo de control.

Desarrollo.

El alumno caracteriza el sistema aplicando a través del generador (C. D/A), señales patrones (escalón) y registrando (figura 8) mediante los C. A/D la evolución de las señales.

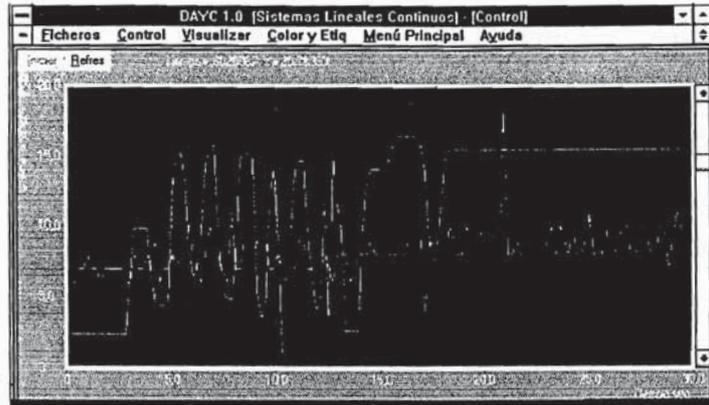


Figura 8. Aspecto de la ventana de registro de señales

Con los registros en pantalla obtiene la función de transferencia que mejor aproxima a cada elemento. Posteriormente y ayudado por el resto de utilidades de la aplicación (Lugar Raíces, Bode, etc.) diseña teóricamente un regulador apropiado. Se le informa al computador el regulador que debe utilizar y se pone a funcionar el sistema en lazo cerrado, registrando la evolución de las señales.

5. CONCLUSIONES

El trabajo presentado ha tratado de mostrar que resulta posible compatibilizar la utilización de tecnologías actuales y los escasos presupuestos disponibles para la formación. A nuestro entender no se precisa de sofisticados y costosos sistemas (Software y Hardware) para conseguir introducir los instrumentos virtuales y sus aplicaciones dentro de la formación universitaria. A veces, una buena dosis de imaginación y muchas horas de trabajo permiten conseguir lo que cientos de millones de pesetas no podrían.

6. BIBLIOGRAFÍA.

- [1] National Instruments. "Software advances in measurements and instrumentation". National Instruments seminar 1992.
- [2] Visual BASIC. Manuales de la versión 3.0 profesional. Microsoft Corp.
- [3] Data Translation. "Product Handbook". Data Translation 1994.
- [4] LipSoft. Manuales del usuario de sistemas de adquisición. Dpto. Electrónica . U. Jaén.
- [5] Robert F. Coughlin, Frederick F. Driscoll. "Amplificadores operacionales y circuitos integrados lineales". Prentice Hall. 1991.
- [6] Stephen. E. Derenzo. "INTERFACING: A laboratory approach using de microcomputer for instrumentation, data analysis, and control". Prentice-Hall. 1990.