

Desarrollo de un escalímetro multicanal con conexión USB para supervisión y registro del flujo de neutrones en los conductos de irradiación del reactor RP-10

Oscar Baltuano*

Dirección de Investigación y Desarrollo, Instituto Peruano de Energía Nuclear,
Av. Canadá 1470, Lima 41, Perú

Resumen

Los experimentos que son realizados en la sala experimental del reactor RP-10, en muchos casos, requieren la supervisión del flujo relativo del haz de neutrones para normalizar los resultados obtenidos durante la experimentación. Esto debido a las variaciones normales en la potencia del reactor durante su operación. Un contador temporizador automático multicanal (MCS) para el registro de estas variaciones es muy conveniente durante el tiempo que dure el experimento. Un equipo comercial es muy costoso y estaría siendo subutilizado si se destinara a este trabajo. Dispositivos electrónicos modernos, muy económicos permiten desarrollar soluciones adaptadas a este requerimiento a un costo muy bajo. Se presenta los resultados del desarrollo de un MCS compacto destinado a cubrir esta necesidad.

Abstract

Normally, experiments that are performed at the RP-10 reactor hall require to measure the relative neutron flux in order to normalize results obtained during the experiment. This is necessary because of the reactor power changes during the operation period. An automatic multichannel counter timer or scaler (MCS) is convenient for recording to register these neutron flux changes during the experimental period. Commercial equipment is very expensive and would be under-utilized if used to perform this task. Modern electronic devices are cheaper and allow to develop solutions adapted to this requirement while maintaining a low cost. Thus, in this study the results of development of a compact MCS designed to meet the stated requirement are presented.

1. Introducción

Durante el funcionamiento rutinario del reactor RP-10 son frecuentes las variaciones de la potencia de operación, dentro de ciertos límites de tolerancia. Estas fluctuaciones de potencia se reflejan en las variaciones correspondientes del flujo de neutrones en los conductos de irradiación. En determinados experimentos es necesario mantener este flujo constante, y en caso de no ser posible, se hace imprescindible conocer estas variaciones de forma relativa durante el tiempo que dure el experimento con la finalidad de normalizar los resultados obtenidos.

La medición de la variación relativa del flujo se realiza generalmente usando detectores de fisión (U-235) conocidos como monitores de haz colocados directamente en el haz de neutrones. La cadena de instrumentación similar a las utilizadas en otras experiencias. El número de cuentas durante un tiempo

prefijado es registrado en un contador-temporizador o escalímetro. Actualmente, en la sala experimental del RP-10 solo se cuenta con equipos que funcionan en modo stand-alone y no poseen ningún tipo de conexión a una PC, razón por la cual, el trabajo de registro de las variaciones del flujo tiene que hacerse en forma manual.

Por esta razón, se hace necesario disponer de un escalímetro que, fijado su tiempo de medición sea redispensible en forma automática y tenga la capacidad de almacenar en cada período de medición, el número de cuentas obtenidas; es decir, con capacidad multicanal. El resultado final es un diagrama de cuentas versus tiempo que representa las variaciones relativas en el flujo de neutrones en el punto de medición.

* Correspondencia autor: obaltuano@ipen.gob.pe

Este tipo de instrumentos son conocidos como Multichannel Scaler (MCS) y tienen una concepción de diseño digital relativamente sencilla, siempre en cuando no se requieran tiempos de adquisición y respuesta o velocidades de funcionamiento muy elevadas. Por esta razón, se optó por desarrollar un MCS compacto, de bajo costo y adaptado a las necesidades específicas de medición del flujo relativo de neutrones en los conductos de irradiación.

2. Diseño electrónico

2.1 Consideraciones de diseño

El MCS requerido debe tener la capacidad de funcionar en modo stand-alone o conectado a una PC en modo esclavo. Actualmente, el estándar usado para la conexión de periféricos a una PC es el puerto USB, por lo cual se decidió utilizar este tipo de conexión para la configuración, transferencia de datos y operación en modo esclavo del MCS.

El ancho de la ventana de tiempo de medición debe ser configurable entre un rango amplio de valores con un mínimo de 1 segundo y un máximo de 30 minutos. El número de canales depende del tiempo total de medición deseado y del ancho de la ventana de medición (*TimeWindow*).

$$\#canales = \frac{T_{medición}}{TimeWindow}$$

A diferencia del modo stand-alone en el cual el número de canales está limitado por la capacidad de memoria disponible en el MCS; en modo esclavo, el número de canales está limitado únicamente por la capacidad de almacenamiento de la PC.

2.2 Ingeniería básica

Se seleccionó un microcontrolador (μ C) PIC18F2550 [1] el cual posee, entre otras características, un puerto USB 2.0 Este dispositivo contiene todo el hardware necesario para implementar un MCS, sin requerir hardware adicional. El firmware o programa de control del microcontrolador ha sido desarrollado para realizar todas las tareas de control, temporización y comunicación. Únicamente ha sido necesario recurrir a unos pocos LED's, pulsadores, algunas resistencias y condensadores adicionales.

2.3 Firmware

El programa para el funcionamiento del microcontrolador se escribió en lenguaje C usando el compilador MPLAB C18 [2] de Microchip y el entorno de desarrollo MPLAB IDE.

Debido a que el dispositivo posee todo el hardware necesario para la aplicación no ha sido necesario programar ningún periférico externo (Figura 1).

El controlador está configurado para operar a una frecuencia de 48 MHz, a partir de un cristal de 20 MHz. Esto se logra a través de un PLL interno que genera la frecuencia de 48 MHz necesaria para operar el puerto USB 2.0 La velocidad interna del CPU es $\frac{1}{4}$ de la frecuencia del cristal: 12 MIPS (millones de instrucciones por segundo). El TIMER1, un contador interno de 16 bits recibe de entrada esta frecuencia, luego de pasar un por un divisor de frecuencia entre 8 (1.5 MHz) El TIMER1 está configurado para proporcionar una interrupción de hardware cada 1/25 segundo. Un registro interno de 16 bits (*CounterTics*) realiza el conteo de los impulsos anteriores para generar la temporización o tiempo de medición. Teniendo en cuenta el valor máximo de este contador interno: 65535; el tamaño máximo de la ventana de tiempo será:

$$TimeWindow = \frac{65535}{25} \approx 2600 \text{ segundos.}$$

Considerando este valor, se ha fijado, por software, un valor máximo de 1800 s (30 minutos) para la ventana de tiempo. El valor mínimo también determinado por software es de 1 segundo.

El TIMER3 interno es otro contador de 16 bits que ha sido configurado para realizar el conteo de los pulsos de entrada provenientes del detector. El valor máximo (65535) impuesto por el tamaño del contador ha sido superado utilizando además un registro interno de 16 bits (TMR3_UP), lo cual permite aumentar la capacidad de conteo a 32 bits o $4,295 \times 10^9$ cuentas aproximadamente.

El funcionamiento del controlador se basa en una máquina de estados que realiza las tareas de conteo, temporización y comunicación con la PC de forma coordinada y secuencial. Durante la operación, al finalizar la ventana de tiempo, los 4 bytes (32 bits) conteniendo el valor de conteo de ese tiempo son

almacenados en la memoria EEPROM del controlador y simultáneamente son enviados a la PC por el puerto USB.

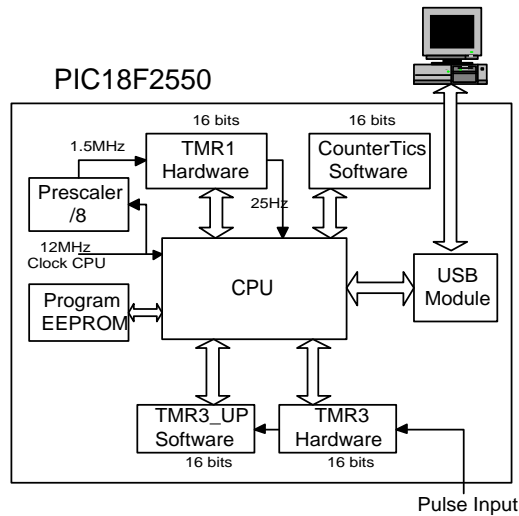


Figura 1. Esquema simplificado del hardware interno utilizado del microcontrolador.

El PIC18F2550 posee una memoria de programa de 32KB; de los cuales, 16KB se reservan para el programa de control y el stack de comunicación USB, los otros 16KB de la memoria son utilizados para almacenar los valores de contaje. Cada valor tiene una extensión de 4 bytes, por lo cual la capacidad interna del controlador es de 4196 canales. Este valor satisface perfectamente cualquier requerimiento actual y permite que el dispositivo trabaje en modo stand-alone. Durante la operación en modo esclavo esta limitación no existe por cuanto los valores son transferidos y almacenados directamente en la PC conforme se van generando.

2.4 Comunicación USB

La conexión de un dispositivo embebido a una PC a través del puerto USB es un proceso complejo que involucra un diseño cuidadoso del firmware del periférico, el software de control en la PC (host) y los drivers necesarios para establecer la comunicación. Sin embargo, un método sencillo para realizar esta conexión es la conversión de una interfaz RS-232 a USB [3]. Este método simple pero eficaz implica ciertas limitaciones en la comunicación, por lo cual su implementación requiere un análisis de los requerimientos de la aplicación.

Hay diferentes formas de convertir una interfaz RS-232 a USB, cada una de ellas implica un distinto nivel de habilidad. La más simple es emular la interfaz RS-232 sobre el puerto USB. Una ventaja de este método es que la aplicación en la PC reconoce la conexión USB como un puerto RS-232 y por lo tanto no se requieren cambios en el software de la PC. Otra ventaja es que este método aprovecha los controladores de Microsoft que ya están incluidos desde Windows 98 SE en adelante, haciendo innecesario el desarrollo de un driver.

La especificación CDC (Communications Device Class) de USB [4] define varios modelos de comunicación, incluyendo la emulación serial. Microchip proporciona para su familia de microcontroladores con interfaz USB, un firmware USB CDC que contiene los descriptores para un dispositivo emulador RS-232 USB y todos los manejadores requeridos específicos de la clase. De esta manera, el diseñador no debe preocuparse en crear descriptores o manejadores de funciones para escribir los requerimientos específicos de clase, facilitando el desarrollo de la aplicación.

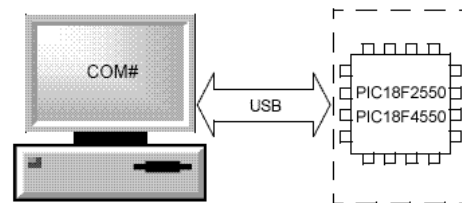


Figura 2. Emulación en PC de un puerto RS-232 para comunicación con dispositivos USB.

En este caso específico, se ha utilizado el ejemplo “USB Device - CDC Basic Demo” contenido en la librería “Microchip USB Firmware Framework v.2.3” [5]; el cual, ha sido modificado para esta aplicación en particular.

2.5 Programa de control en la PC

En el lado de la PC se ha desarrollado un programa de control escrito en Fortran 95. Para la comunicación se utilizó la librería SciComm de MicroGlyph [6] que proporciona acceso a las API's de comunicación RS-232 de Windows. También ha sido de utilidad el conjunto de librerías gráficas de Winteracter para Windows.

MS Windows no posee un archivo .inf estándar para el controlador CDC; sin embargo, los controladores son parte de la instalación de Windows. Solo es necesario proveer un archivo .inf cuando un dispositivo CDC es conectado inicialmente a un sistema Windows. Archivos .inf de ejemplo se encuentran en [5]. Para utilizarlos, deben ser modificados para reflejar los VID (Vendor ID) y PID (Product ID) específicos con los que ha sido grabado el firmware en el dispositivo. Si es necesario también se puede modificar la cadena correspondiente para cambiar el texto en la identificación del dispositivo vista por el usuario en el manejador de dispositivos de Windows.

La estructura del programa de control es bastante simple: recibe del MCS los datos correspondientes al número de cuentas y el valor de tiempo de medición correspondiente. Ambos datos son almacenados en dos vectores (arrays), los cuales a su vez son visualizados en forma de un gráfico de dispersión X-Y.



Figura 3. Imagen mostrando el protoboard (círculo rojo) montado con el módulo MCS conectado a una computadora portátil y a la cadena de instrumentación.

El programa también envía al MCS, comandos para el inicio/parada de conteo y la operación de reset que borra la información tanto en la PC como en el MCS. La asignación del valor de la ventana de tiempo *TimeWindow* se hace igualmente desde el

programa de control y puede ser modificado en cualquier momento. Finalmente, se adicionaron opciones para hacer la depuración de la comunicación y el guardado de la información en un archivo en el disco.

3. Operación, Resultados y Discusión

El módulo se ha puesto en operación tanto en pruebas de laboratorio con un generador de pulsos, como en mediciones del flujo de neutrones en los conductos de irradiación del reactor RP-10 con un monitor de haz y la cadena de instrumentación correspondiente (Figura 3).

Con una frecuencia de entrada fija de 12 MHz la variación máxima obtenida en la medición ha sido de ± 30 cuentas por segundo. Esto equivale a un error de menos de $2,5 \times 10^{-6}$, valor que satisface ampliamente los requerimientos de la aplicación. Este error tiende a disminuir conforme disminuye la frecuencia de entrada debido a que los retardos producidos en el microcontrolador para atender las operaciones de hardware varían en el rango de algunas decenas de nanosegundos, esto gracias a la alta frecuencia de operación del CPU. Estos retardos del microcontrolador son menos significativos conforme la frecuencia de entrada es menor, y, a frecuencias por debajo de 1 MHz el error puede considerarse, para fines prácticos, cero.

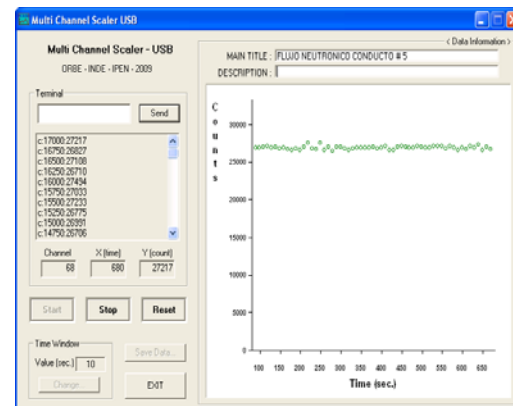


Figura 4. Front-End del programa de control mostrando el resultado obtenido a 7.5 MW con mediciones cada 10 segundos en el conducto de irradiación N° 5.

En razón a que las mediciones del flujo de neutrones no superan tasas de pulsos de 10^6 debido al tiempo muerto de los detectores. También la característica intrínsecamente

aleatoria de los procesos nucleares, que producen variaciones estadísticas en las mediciones nucleares, hacen que los errores antes mencionados no sean significativos para estas aplicaciones.

En la Figura 4 se observa el resultado de la medición relativa del flujo de neutrones en el conducto de irradiación N° 5 del reactor RP-10 a una potencia nominal de 7,5 MW, con intervalos de 10 segundos, para lo cual se utilizó un monitor de haz de fisión de U-235. Se observan ligeras fluctuaciones en el flujo de neutrones.

4. Conclusiones

Los dispositivos electrónicos modernos tales como los microcontroladores brindan la posibilidad de desarrollar instrumentación específica a un costo muy bajo comparado con los equipos comerciales.

Encontrar el punto de equilibrio entre utilidad y costo debe ser parte fundamental en el diseño y desarrollo de un producto. En el caso del MCS descrito, se han considerado únicamente las características mínimas indispensables de operación para obtener una buena relación beneficio/costo. Aún así, la inclusión de nuevas funcionalidades es una tarea sencilla debido a su versatilidad y escalabilidad.

El uso de dispositivos embebidos conectados a PC's a través de un puerto USB, debería ser el estándar de desarrollo de nuevos productos debido a la amplia utilización de este tipo de interfaces, razón por la que se ha convertido en el estándar actual de la industria.

Si bien es cierto, el uso de interfaces USB involucra un grado de dificultad mayor que otro tipo de conexiones, la emulación serial proporciona un camino de fácil migración para desarrollos ya terminados, pasando de RS-232 a USB. Del lado de la PC, se requiere una mínima modificación del software. Del lado del dispositivo embebido, la familia de microcontroladores PIC18F2455/2550/4455/4550 proporciona una actualización simple del hardware desde las familias de dispositivos PIC16C745/765 y PIC18FXX2.

5. Bibliografía

[1] Microchip Inc., PIC18F2455/2550/4555/4550 Data Sheet. 28/40/44-Pin, High Performance, Enhanced Flash, USB Microcontrollers with Nano Watt Technology. USA. 2007.

[2]Microchip, MPLAB C18 C Compiler User's Guide. USA, 2005.

[3]Rawin Rojvanit, Migrating Applications to USB from RS-232 UART with Minimal Impact on PC Software. Microchip, USA, 2004.

[4] USB Implementers Forum Inc., "USB Class Definitions for Communication Devices Version 1.1", 1999.

[5]Microchip Inc., Microchip USB Device Firmware Framework User's Guide. USA. 2008.

[6]MicroGlyph Systems, SciComm Serial Communication Library Version 7.0 <http://www.microglyph.com/>