

SISTEMA DE ADQUISICIÓN DE DATOS INTELIGENTE PARA UN REACTOR DE FUSIÓN.

Alberto Sánchez Florido

Ingeniería en Electrónica Industrial y Automática
EPSEVG, Departamento Ingeniería Electrónica UPC
Av. Víctor Balaguer s/n, 08800 Vilanova i la Geltrú

1. Resumen

En este trabajo, el punto inicial de partida es el de un hardware que se encarga de recolectar información sobre radiaciones de fusión nuclear. Dicho hardware consta de bolómetros y una tarjeta de adaptación de nivel de voltaje y amplificación de señal.

Debido a la necesidad de variar la ganancia en cada ensayo determinado con tal de obtener unos resultados de lectura de radiaciones óptimos y comprensibles, se debe seleccionar una de cuatro ganancias para los amplificadores.

El bolómetro es el instrumento a partir del cual se obtiene la información de la radiación electromagnética. A partir de aquí, interesa acondicionar la señal del bolómetro con parámetros ajustables, es por eso que se diseña una placa de adaptación y amplificación.

En total existen 16 líneas de adaptación y amplificación. En una de ellas nos encontramos con un amplificador diferencial INA13 seguido de un amplificador de instrumentación INA128.

La función del INA137 es la de adaptación de nivel, pues realmente atenúa para poder trabajar con un rango dinámico que se verá directamente afectado por la ganancia del INA12, y en este caso en particular es de gran interés evitar la saturación de cada una de las líneas.

Se tratará de implementar un control de ganancia remota e inteligente, puesto que el sistema actual dispone de un sistema físico y manual para la configuración de las cuatro diferentes ganancias, además se incorporará un 'offset' o un nivel de voltaje añadido en los amplificadores con tal de exprimir al máximo el rango dinámico del sistema de adquisición de datos o SAD.

Dicho control se gestionará mediante un microcontrolador dispuesto en una plataforma de desarrollo llamada Netduino Plus 2.

Como característica adicional, y aprovechando el nuevo hardware a instalar, una vez estudiada la normativa TEDS (una hoja de datos electrónica implementada sobre el propio sistema) se reproducirá una versión similar en la propia tarjeta nueva. Esto permite la lectura y escritura no solo de las ganancias de los amplificadores, sino también de parámetros destacables e importantes del hardware diseñado que se deben tener en cuenta en momentos determinados de los ensayos.

El sistema anteriormente descrito se encuentra actualmente instalado en Madrid, concretamente en el CIEMAT, donde se llevan a cabo las pruebas.

Una vez diseñado todo el hardware y software, éste quedará disponible para sustituir al sistema actual y potenciar las posibilidades del mismo.

2. Introducción

El sistema viene controlado por una placa de desarrollo de nombre comercial Netduino Plus 2. Esta placa controladora se puede programar en lenguaje C# a través de Microsoft Visual Studio. Además cuenta con una bahía micro-SD por la cual se instalará una tarjeta con los datos del TEDS del sistema en cuestión para poder acceder a ella remotamente. Paralelamente se encargará de la gestión de las ganancias programables de la placa de adaptación y amplificación.

3. Programa de control

La idea principal es poder utilizar la Netduino Plus 2 para controlar remotamente todas las ganancias de los amplificadores INA128 y a la vez almacenar y consultar el TEDS propio de la placa. Para ello, se ha almacenado en una tarjeta Micro-SD el contenido básico de algunos parámetros de la placa en formato .txt, como pueden ser las ganancias (inicialmente en un valor por defecto) y otros parámetros propios de un TEDS básico.

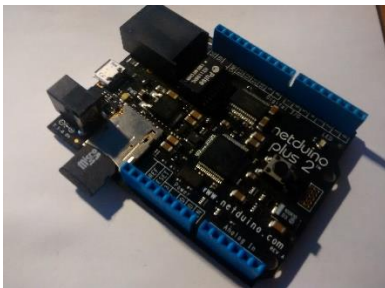


Figura 1. Netduino Plus 2 y la tarjeta micro-SD.

Además del control de la ganancia, se ha añadido un offset de -5V a los INA128 para poder aprovechar al máximo el rango dinámico del SAD.

En un principio se pensó en un archivo con formato XML para almacenar los datos del TEDS en la tarjeta SD, pero más tarde se rediseñó en un formato

propio con tal de reducir el número de bytes de información, ya que el protocolo utilizado (TCP/IP) da problemas si nos encontramos con datos mayores a 536 bytes debido al propio protocolo.

A priori se ha analizado el sistema actual. La parte del control de las ganancias consta de unos micro-interruptores cuya desventaja reside en tener que estar físicamente en el lugar de la placa para poder cambiar las ganancias.

El software utilizado para la edición y creación de las PCB, así como la creación de librerías de los componentes requeridos se llama Altium Designer.

4. Hardware

En este trabajo también se realizó el aprendizaje de la norma IEEE Std. 1451.4. En este apartado se puede consultar el hardware utilizado, como por ejemplo una memoria EEPROM DS2431 que cuenta con el protocolo One-Wire.



Figura 2. Acelerómetro 4508-B003 compatible con TEDS.



Figura 3. NI SC-2350

El SC-2350 de National Instruments es un adaptador blindado para módulos SCC de acondicionamiento de señales que tiene conectores configurables para sensores inteligentes TEDS Clase II. El NI SC-2350, el cual se cablea directamente a dispositivos DAQ de la Serie M y la Serie E, puede ser energizado ya sea directamente desde el dispositivo DAQ o con una fuente de alimentación externa.

Por la parte de adquisición del bolómetro, es interesante modificar lo menos posible la placa de adaptación y amplificación, por lo que únicamente se han creado pistas en los pines 5 de los INA128, que corresponden a las tensiones de referencia, para poder así introducir el offset de -5V.

Estas pistas se llevarán a la placa secundaria donde se instalará la Netduino Plus 2, y allí mediante un regulador de tensión se proporcionará la tensión de -5V.

La placa secundaria contiene el circuito necesario para el control de las ganancias.

En un principio resultaba inviable llevar 64 pines digitales a los conectores J para crear un circuito que uniera las diferentes combinaciones de ganancias, por lo que se llegó a la conclusión que lo

mejor era implementar un circuito compuesto de registros de desplazamientos y multiplexores. De esta manera el número de pines se reduce a tan solo 3.

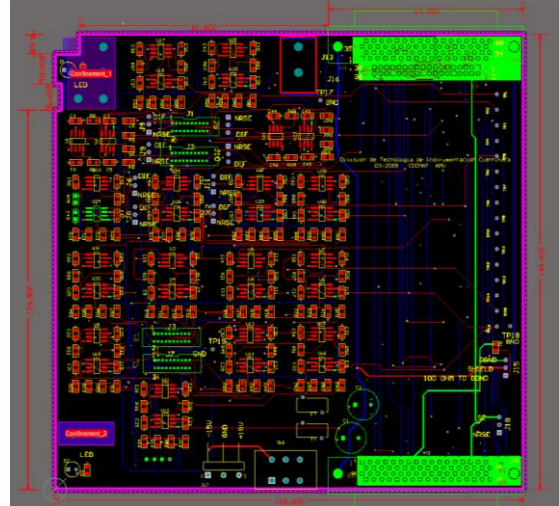


Figura 4. Hardware de la tarjeta de adaptación de nivel y amplificación de señal.

5. Software

```
public void ReadAllMem(byte[] data)
{
    OneWireBus.TouchReset();
    //OneWireBus.WriteByte(Command.SkipROM); //skip ROM
    OneWireBus.WriteByte(Command.ReadScratchPad); // Read Memory
    OneWireBus.WriteByte(0x00); //Read Offset 0000h
    OneWireBus.WriteByte(0x00);

    for (int i = 0; i < 128; i++) //whole mem is 144
    {
        data[i] = (byte)OneWireBus.ReadByte();
    }
}

public byte[][] RetrieveROM()
{
    System.Collections.ArrayList devices = new System.Collections.ArrayList();
    OneWireBus.TouchReset();
    devices = OneWireBus.FindAllDevices();
    byte[][] ROM = new byte[devices.Count][];
    for (int i = 0; i < ROM.Length; i++) ROM[i] = new byte[8];
    for (int d = 0; d < devices.Count; d++)
    {
        for (int i = 0; i < 8; i++)
        {
            ROM[d][i] = ((byte[])devices[d])[i];
            if (ROM[d][0] != 45) //sino es un DS2431, l'expulso de la llista
            {
                ROM[d][0] = 0;
                d--;
                break;
            }
        }
    }
}
return ROM;
}
```

Figura 5. Software diseñado para la memoria DS2431.

Por la parte del aprendizaje de la norma IEEE Std. 1451.4. se ha desarrollado un programa para la escritura y lectura de la memoria DS2431 mediante la Netduino Plus 2.

En la figura 5 se puede consultar un extracto del código diseñado.

Por la parte del software de control de las ganancias, se puede consultar un extracto de código en la figura 6.

```

public static void StartListening()
{
    while (true)
    {
        if (newdata)
        {
            UpdateAllAmplifiers();
            newdata = false;
        }
        using (Socket clientSocket = listeningSocket.Accept())
        {
            led.Write(true);

            IPEndPoint clientIP = clientSocket.RemoteEndPoint as IPEndPoint;
            Debug.Print("Received request from " + clientIP.ToString());
            clientSocket.Receive(receive);
            if (CompareChars(System.Text.Encoding.UTF8.GetChars(receive), Commands.Read ))
            {
                message = ReadConsumed();
                message2=AddBytesToRead(message);
                clientSocket.Send(Encoding.UTF8.GetBytes(message2));
            }
            if (CompareChars(System.Text.Encoding.UTF8.GetChars(receive), Commands.Write))
            {
                clientSocket.Receive(buffer);
                StreamWriter sw = new StreamWriter(path);
                char[] save = System.Text.Encoding.UTF8.GetChars(buffer);
                sw.Write(save);
                Thread.Sleep(300);
                sw.Close();
                newdata = true;
            }
            clientSocket.Close();
            led.Write(false);
        }
    }
}

```

Figura 6. Software diseñado para la comunicación entre servidor (Netduino Plus 2) y cliente (LabVIEW).

Continuando con el software, en la figura 7 se puede consultar un fragmento correspondiente al software para el control de la ganancia en la propia tarjeta de adaptación de nivel y control mediante la Netduino Plus 2.

```

public void selectAmplifierAndGain(Amplifier s, Gain g)
{
    int ser = (int)g;
    SerValue = (uint)(ser << (((int)s) * 2));
}

public void selectAmplifierAndGain(int s, int g)
{
    int ser = (int)g;
    SerValue = (uint)(ser << (((int)s) * 2));
}

private void RefreshSelection(uint SerValue)
{
    cnt = 31;
    while (cnt >= 0)
    {
        int bit = (int)SerValue % 2;
        rckport.Write(false);
        sckport.Write(false);
        if (bit == 1)
        {
            serport.Write(true);
        }
        else
        {
            serport.Write(false);
        }
        sckport.Write(true);
        rckport.Write(true);
        SerValue = SerValue / 2;
        cnt--;
    }
}

```

Figura 7. Software de control de ganancia.

5. Solución adoptada

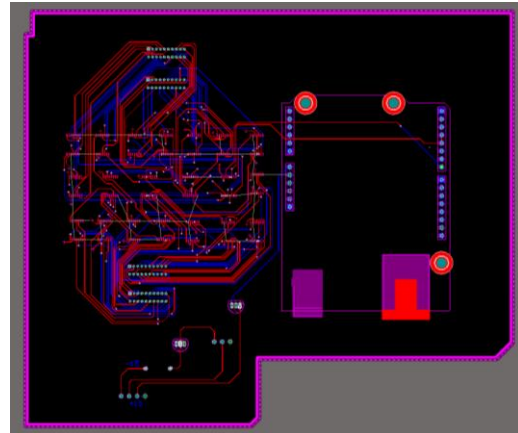


Figura 8. Placa secundaria diseñada, donde se instala la Netduino y los dispositivos de desplazamiento y multiplexación de puertos, además del offset de -5V.

A pesar de las limitaciones de la Netduino, con esta configuración se obtiene el mismo resultado que con 64 pines digitales. Los registros de desplazamiento reciben la palabra digital desde un pin digital mediante software. Estos registros de desplazamiento colocan los bits de la palabra digital a través de sus pines de manera que se obtiene más pines de salida. Una vez están los bits correspondientes en cada pin de los registros, los multiplexores los reciben en sus pines de selección de salida.

La ganancia depende entonces, de la palabra digital que reciben los registros de desplazamiento.

Los multiplexores analógicos cierran el circuito con la resistencia deseada, de manera similar a colocar un jumper entre dos extremos.

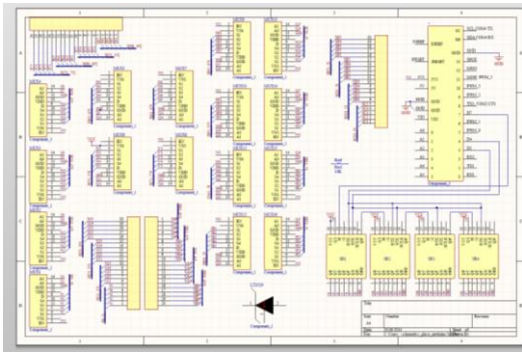


Figura 9. Esquema eléctrico de la placa secundaria diseñada.

El regulador de tensión LT1029 se alimenta a -15V y junto con una resistencia de 10kΩ es capaz de generar una tensión de salida de -5V.

En el programa diseñado en LabVIEW, cuando pulsamos sobre READ, se establece una comunicación TCP con la Netduino a través de un cable Ethernet conectado al PC o a cualquier nodo de la red donde se pueda alcanzar la IP configurada al conector RJ45 de la Netduino.

Se puede observar en la figura 46 las diferentes pestañas del recuadro blanco que corresponden a los 16 canales de amplificación para cada uno de los INA128.

La Netduino recibe el comando READ y envía toda la información del archivo .txt al programa LabVIEW que a su vez muestra la información recogida en el panel frontal.

El programa también permite realizar cambios sobre la información, ya sea en los campos del TEDS como en las ganancias. Dichos cambios se reflejan en la Netduino y ésta se encarga de actualizar todas las ganancias de la placa automáticamente según la información introducida en el panel frontal.

Antes de poder modificar la información, se debe entrar en modo edición pulsando sobre el botón EDIT. Una vez pulsado, se puede aplicar los cambios sobre el canal en el que nos encontramos según la disposición de las pestañas pulsando sobre el botón APPLY CHANGES.

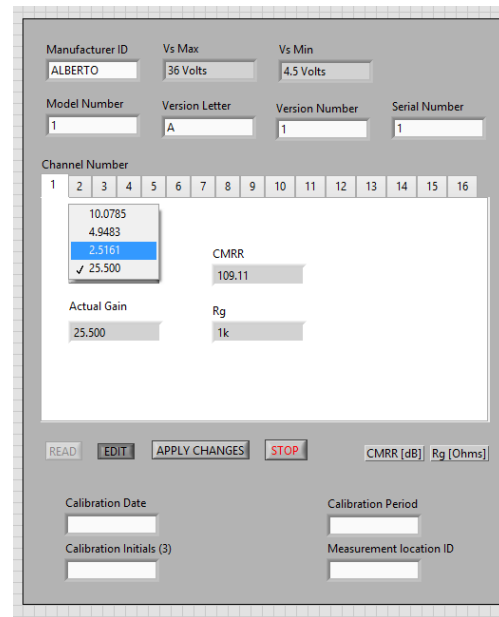


Figura 10. Modo edición del panel frontal donde podremos modificar parámetros como las ganancias.

6. Conclusiones

Todos los resultados obtenidos han sido los esperados en mayor o menor medida.

El hecho de disponer del protocolo One-Wire para la implementación de un TEDS facilita enormemente la labor de la instalación de cualquier dispositivo que soporte este método, ya que tan solo se requiere de una línea de datos y alimentación compartida.

Por otra parte, el protocolo escogido en la escritura y lectura del TEDS en la Netduino es aceptable para una transmisión de datos de tamaño reducido. No obstante, si se desea transmitir información más extensa, ya sea en forma XML o JSON se debe

fracionar en diferentes paquetes, ya que el protocolo TCP/IP tan solo soporta 536 bytes de manera remota.

Al final se decidió por adoptar un formato de texto similar a una versión simplificada del formato JSON, y de esta manera simplificar el envío y captura de datos en un solo paquete que se traduce en una sola función en LabVIEW.

El siguiente paso de este proyecto será montar todo el sistema en un entorno de ensayo real, hecho que dará fruto a nuevos proyectos y colaboraciones.

En cuanto al sistema Netduino, cabe destacar que es una plataforma excelente que cuenta con un entorno de programación a mi parecer exquisito, con una infinidad de opciones y configuraciones, además de puntos de parada en código y ventana de inmediato donde el usuario es capaz de evaluar código en tiempo real mientras el programa permanece en pausa. Quizás el único hincapié destacable en este tipo de plataformas de desarrollo como es la Netduino es la imposibilidad en ocasiones del aprovechamiento de todos los recursos que puede ofrecer este sistema, ya que en el caso de este proyecto la placa ofrecía muchos más pines y protocolos de comunicación que no se ha encontrado la obligación de utilizar pero que están ahí y supone un coste de fabricación. Si el sistema se dedica exclusivamente a este tipo de tareas se puede suponer que no se exprime todo el potencial, no obstante, al ser un sistema de fabricación en gran escala por parte del fabricante su precio se ve ajustado.

Cuando se trabaja con señales analógicas cuyas mínimas modificaciones pueden comprometer toda la cadena de medida, es muy importante tener siempre presente el

tema del ruido eléctrico. Este proyecto tuvo ciertos problemas a la hora de poner en marcha la fabricación de las placas diseñadas por la cantidad excesiva de planos y masas para evitar ruidos.

7. Aspectos a mejorar.

El primer aspecto a mejorar hace referencia a la implementación de la señal de offset hacia los amplificadores, ya que se puede dar el caso donde cada amplificador requiera un offset diferente. En nuestro sistema el offset se comparte en los 16 canales, hecho que no puede ser del todo conveniente. Por tanto una alternativa es, mediante otro control lógico que pude realizar la propia Netduino, realizar una multiplexación parecida a la que se realiza con las ganancias, pero esta vez con las tensiones de referencia, disponiendo así de 16 tensiones de referencia independientes para cada canal.

El otro aspecto hace referencia el TEDS implementado. Si bien es cierto que se le puede dar una funcionalidad añadida a la parte del TEDS ofreciendo más parámetros de interés que pueden ser de gran utilidad a la hora de la consulta de los datos.

8. Bibliografía

(Del Rio J. et al 2011). Standards-based Plug & Work for Instruments in Ocean Observing Systems” IEEE Journal of Oceanic Engineering. (ISSN 0364-9059)

(J del Río et al, 2011) “IEEE 1451 HTTP Server Implementation for Marine Data” Fourth International Workshop on Marine Technology. MarTech 2011

(K C Lee et al 2004) IEEE-1451-Based Smart Module for In-Vehicle Networking Systems of Intelligent Vehicles. K C Lee, M H Kim, S Lee, H H Lee

(Arcidiacono, R. et al, 2005) Feasibility study of a XML-based software environment to manage data acquisition hardware devices

(Jose Antonio Soriano García, 2013) Desarrollo de un sistema de Adquisición de datos inteligente conforme a la norma IEEE Std. 1451.4

www.ni.com

<http://netduino.com/>

<http://www.maximintegrated.com/en/products/1-wire/flash/overview/index.cfm>

<http://msdn.microsoft.com/en-us/library/ee435793.aspx>