



Almacenamiento de datos de Vibraciones de Motor BLDC para graficación y análisis en displays disponibles en tarjeta AVR Butterfly y en Tarjeta Controladora LPCXpresso (Diciembre 2012)

Edison Figueroa⁽¹⁾, Jorge Saavedra⁽²⁾, Carlos Valdivieso⁽³⁾
Facultad de Ingeniería en Electricidad y Computación⁽¹⁾⁽²⁾⁽³⁾
Escuela Superior Politécnica del Litoral (ESPOL)⁽¹⁾⁽²⁾⁽³⁾
Campus Gustavo Galindo, Km 30.5 Vía Perimetral, Apartado 09-01-5863. Guayaquil, Ecuador⁽¹⁾⁽²⁾⁽³⁾
edijafig@espol.edu.ec⁽¹⁾, jopasaav@espol.edu.ec⁽²⁾, cvaldiv@fiee.espol.edu.ec⁽³⁾

Resumen

Este artículo tiene como objetivo mostrar una aplicación que hemos desarrollado basada en el uso de dos tipos de microcontroladores avanzados, esta será implementar un sensor de impacto y sonido para un motor BLDC.

Para la realización del proyecto se utilizan dos microcontroladores: la tarjeta LPCXPRESSO 1769 de 32 bits y el Kit de desarrollo AVR Butterfly, que fueron programados con ayuda de dos herramientas de programación llamadas LPCXpresso 4 y AVR Studio 4, además de la utilización de la pantalla LCD disponible en tarjeta AVR Butterfly y presentación de mensajes de operación, adicional a esto se complementó un sistema de alarma.

Palabras claves: AVR Butterfly, Plataforma Interactiva, Comunicación UART.

Abstract

This article aims to show an application based on the use of two type of advanced microcontrollers, it will implement a sound impact sensor form BLDC motor.

In order to accomplish our goal, we are using two platforms, LPCXpresso 1769 microcontroller card and 32-bit AVR Butterfly development kit, which were programmed using two programming tools called LPCXpresso 4 and AVR Studio 4 to use the LCD available in Butterfly AVR card for message display operation, this is supplemented by additional alarm system.

Keywords: AVR Butterfly, LPCXpresso

I. INTRODUCTION

El proyecto consiste en la comunicación entre dos dispositivos, la tarjeta LPC1769 y el kit de desarrollo AVR Butterfly, haciendo uso del módulo SSP en modo de operación SPI, para analizar el ruido generado por un motor BLDC. Se emplearon los programas LPCXpresso y AVR Studio para la programación de los dispositivos, en conjunto con el lenguaje de programación C.

En la implementación del proyecto, la tarjeta LPC1769 toma la función de emisor o maestro, y se encarga de enviar los datos obtenidos por un sensor de impacto y sonido, asimismo, si existe un dato que sea causa de alerta se activará un contador que al llegar al número 5 activará una un grupo de LED's que funcionarán como alarma del suceso.

La recepción de los datos enviados por la tarjeta LPC1769, la realiza el kit de desarrollo AVR Butterfly, tomando el nombre de receptor o esclavo. Este kit analiza los datos, bajo el mismo criterio que el emisor. Además en la LCD del mismo, se visualizarán mensajes del estado del motor. Para efectos de prueba, el sensor puede ser calibrado de acuerdo a las necesidades del usuario.

II. COMUNICACIÓN SPI

SPI es un estándar que utiliza un bus de cuatro líneas y permite la transmisión sincrónica de datos entre un dispositivo maestro y uno o varios dispositivos esclavos. Es implementado por un módulo de hardware llamado SSP, este módulo permite la comunicación serial síncrona entre dos o más dispositivos a una alta velocidad y es razonablemente fácil de implementar, además que permite una transferencia de datos no limitada por bloques de 8 bits.

Líneas de control y datos de SPI

- SCK: El dispositivo maestro provee una señal de reloj para la sincronización, ésta controla cuando los datos pueden ser enviados y cuando deben ser leídos, a esta señal se le da el nombre de SCK. Como SPI es síncrono, los datos se envían por cada pulso de reloj y la frecuencia del mismo puede variar sin interrumpir la transmisión de los datos, la velocidad de transmisión cambiará al mismo tiempo que varíe la frecuencia. En SPI solo el dispositivo maestro puede controlar la señal de reloj SCK, ningún dato se transferirá a menos que la señal de reloj no esté presente.

- SS: Usualmente una señal selectora controlará cuando un dispositivo esclavo debe permanecer en la transmisión. Esta señal debe usarse cuando existe uno o más dispositivos

esclavos en el sistema pero puede ser opcional cuando solamente existe uno. Esta señal es conocida como SS y significa selector de esclavo (Slave Select). Con esta señal el dispositivo maestro le indica al esclavo que desea iniciar la transmisión SPI, para esto el maestro baja la señal SS a cero voltios, como resultado el dispositivo esclavo se activa; mientras que a niveles de voltaje superior el dispositivo va a estar inactivo.

- MOSI: Esta línea está definida como salida desde el dispositivo maestro y entrada del dispositivo esclavo. Los datos son transmitidos en una sola dirección desde el maestro hacia el esclavo.

- MISO: Esta línea está definida como entrada desde el dispositivo maestro y salida del dispositivo esclavo. Los datos son transmitidos en dirección opuesta al MOSI, es decir desde el esclavo hacia el maestro.

III. SENSOR DE IMPACTO Y SONIDO

Un sensor es un dispositivo diseñado para detectar variaciones de una magnitud física y transformarla en señales útiles capaces de cuantificar y manipular. El sensor, visto en la Figura 1, que utilizamos tiene un rango de detección de sonidos de hasta 3 metros de distancia ajustable por un potenciómetro, que en conjunto con un micrófono, el cual recepta los sonidos, emitirá una señal digital con duración de tiempo igual a la del sonido en la terminal de salida del sensor. El voltaje de alimentación es de 5 voltios.

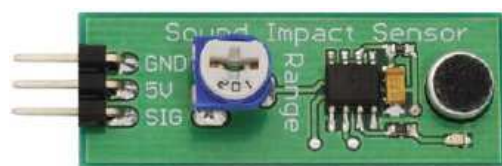


Figura 1 Sensor de Impacto de Sonido [8]

IV. TARJETA LPC1769

La tarjeta LPC1769 consta de dos partes: el microcontrolador LPC1769 y un depurador JTAG, como se puede observar en la Figura 2. La porción de depuración JTAG de la tarjeta LPC se llama LPC-Link. La LPC-Link está conformada por una cabecera de 10 pines JTAG, y se conecta con otros dispositivos a través de la interfaz USB.



Figura 2 Tarjeta LPC1769 [9]

El microcontrolador consta de 100 pines los cuales están distribuidos en su contorno, esta tarjeta cuenta con 3 puertos configurables (GPIO0, GPIO1, GPIO2) como entradas o salidas de un sistema según sea el caso.

V. KIT DE DESARROLLO AVR BUTTERFLY

El kit de desarrollo AVR Butterfly está diseñado para demostrar los beneficios y características claves de los microcontroladores ATMEL, utiliza el microcontrolador ATmega169 y posee partes y aplicaciones que lo distinguen de la tarjeta LPC1769, tales como un display LCD para mostrar información, reproducción de música, sensor de temperatura y un joystick. El kit se alimenta con tres voltios y posee varios conectores para comunicación y programación, como se puede ver en la Figura 3.

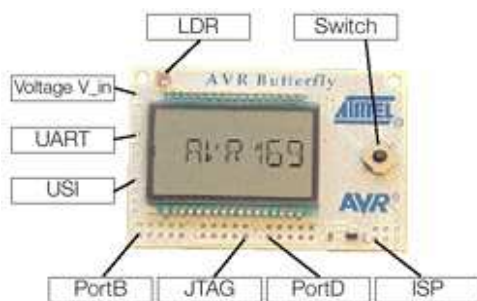


Figura 3 Kit de Desarrollo AVR Butterfly [2]

El ATmega169 es un microcontrolador de 8 bits con arquitectura AVR RISC, y es compatible con un completo conjunto de programas y elementos de desarrollo, como compiladores de lenguaje de programación C, ensambladores de macro, simuladores de programas, emuladores de circuito y kits de evaluación.

VI. MOTOR BLDC

Un motor BLDC sin escobillas es una máquina síncrona con la frecuencia de alimentación, capaz de desarrollar altos torques en forma transitoria para oponerse a todo esfuerzo que trate de sacarla de sincronismo. Hoy en día el motor BLDC es

utilizado en muchos sectores desde implementaciones de forma capsular, es decir micro robots quirúrgicos, hasta aplicaciones de gran escala industrial y automotriz

El motor BLDC (Figura 4) cuenta con las ventajas de una mayor duración dado que no existe rozamiento en las escobillas, que a su vez se traduce en una emisión mucho menor de interferencias electromagnéticas. Otra de las ventajas, es el mayor control sobre la velocidad del motor y su posición. A la vez, permite introducir un mayor número de elementos sensores en el motor, por lo que se pueden realizar controles, no sólo de velocidad y sentido, sino también de errores entre otros parámetros.

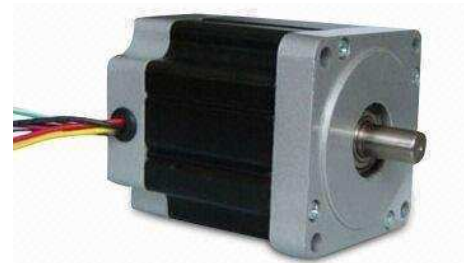


Figura 4 Motor BLDC [10]

Estas notables ventajas, tales como la inexistente caída de tensión (y pérdida de energía) entre las escobillas y las delgas de un colector, el bajo ruido generado durante el funcionamiento o las bajas vibraciones mecánicas, logradas mediante un desarrollo más cuidado, permiten una mejor eficiencia del sistema en general, una mayor fiabilidad de duración y un mejor rendimiento energético [11].

VII. HERRAMIENTAS DE DESARROLLO

LPCxpresso

El LPCxpresso es un programa para evaluación con microcontroladores LPC de NXP, el cual incluye todas las herramientas necesarias para desarrollar una alta calidad de soluciones de manera efectiva en tiempo y costo. Consiste de un editor de texto para editar código fuente, un compilador y enlazador para transformar el código fuente en un programa ejecutable y un depurador.

También cuenta con la última versión de la cadena de herramientas GNU estándar con una biblioteca propia optimizado en lenguaje de programación C. Además el IDE LPCxpresso puede construir un ejecutable de cualquier tamaño con la optimización del código completo.

AVR Studio 4

AVR Studio es un Entorno de Desarrollo Integrado (IDE) para escribir y depurar aplicaciones AVR en el sistema operativo de Windows, tiene una arquitectura modular que permite interactuar con programas de otros fabricantes y soporta todas las herramientas de ATMEL que apoyan a la arquitectura AVR 8 bits.

Para la compilación del código hecho en el AVR Studio se utiliza la distribución WinAvr, que es una recopilación de programas de software libre diseñados para facilitar las tareas de programación y desarrollo de los microcontroladores AVR. Dicha distribución WinAvr incorpora además del compilador GCC de consola, un editor de texto especialmente diseñado para ayudar al programador y hacer el código más legible mediante su resaltado con colores.

VIII. PROYECTO TERMINADO

El proyecto consiste en la transmisión de datos recibidos de un sensor de impacto y sonido en la tarjeta LPC1769 hacia el kit de desarrollo AVR Butterfly, el cual contiene un microcontrolador ATmega169 que está programada para recibir los datos. La LCD mostrará por cada dato recibido un mensaje que nos indica el estado del motor, los mensajes que se muestran son: "NORMAL" y "ALERTA".

En el estado de "NORMAL" se reflejará un comportamiento adecuado del motor; mientras que cuando ocurra lo contrario, inmediatamente se mostrará el mensaje "ALERTA" y en un display de 7 segmentos comenzará un contador que aumenta si el ruido producido por el motor no cesa, en caso de llegar al número 5, entonces se activarán unos leds en forma de alarma.

Para la implementación del proyecto se utilizaron los siguientes materiales:

- 1 Tarjeta LPC1769 (Figura 5a).
- 1 kit de desarrollo AVR Butterfly (Figura 5b).
- 1 Sensor de impacto y sonido 29132 (Figura 5c).
- 1 Cable Adaptador USB (Figura 5d).
- 3 LED'S Bicolor (Figura 5e).
- 1 Display de 7 segmentos (Figura 5f).
- 1 Batería de 3 V (Figura 5g).
- 1 Fuente de 5 V.
- Resistencias de 330 Ω .

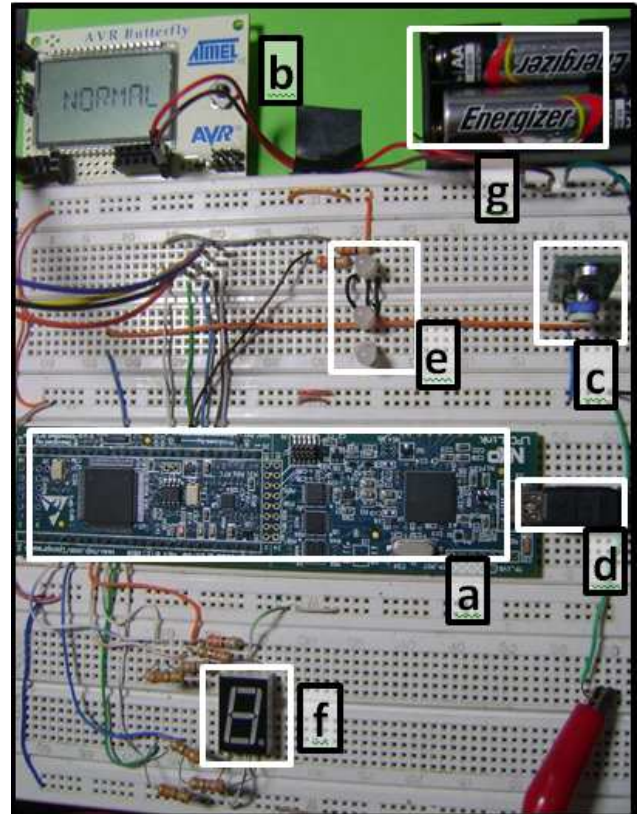


Figura 5 Proyecto en Estado Normal

En la Figura 6 se muestra el diagrama de bloques correspondiente al proyecto de presente tesina, como podemos observar consta de 3 bloques importantes. En el primer bloque tenemos al sensor de Impacto y Sonido el cual detecta los ruidos generados por el motor y actúa como entrada para el segundo bloque que consta de una tarjeta LPC1769 que toma la función de maestro o emisor, y con la cual existe la comunicación SPI con el tercer bloque, representado por el kit de desarrollo AVR Butterfly que actúa como receptor o esclavo. Para la visualización al usuario de una alarma tenemos los bloques: LCD, Display y LED's.

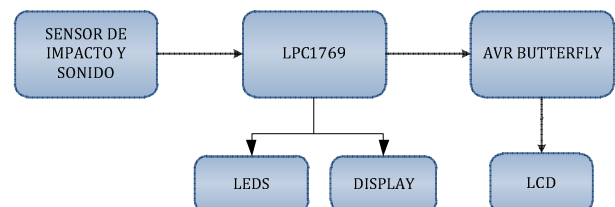


Figura 6 Diagrama de Bloques Proyecto



IX. CONCLUSIONES

1. Este proyecto demuestra la comunicación serial sincrónica SPI entre microcontroladores de distintos fabricantes, que se dio satisfactoriamente gracias al estudio de los manuales de usuarios de cada tarjeta y a la realización de ejercicios con éstos.
2. Si interpretamos todo lo descrito en los capítulos tres y cuatro comprobamos que para el sistema la distancia a la que se encuentra el motor del sensor, es un factor importante, debido a que el sensor puede ser calibrado por el usuario, y es necesario tener en cuenta el ruido generado por otros agentes para que no influya en el funcionamiento del sensor, tampoco sobrepasar el rango de los tres metros de cobertura del mismo.
3. Este proyecto fue desarrollado con lenguaje de programación "C", utilizando la herramienta AVR Studio 4 para programar el microcontrolador ATmega169 del Kit AVR Butterfly, y el programa LPCXpresso para la tarjeta LPC1769. Las librerías del LPCXpresso fueron de gran ayuda ya que sirvieron de ejemplo y guía a la hora de programar, teniendo en cuenta que no se tuvo un programa para simular en funcionamiento de la implementación.

X. RECOMENDACIONES

1. Tener en cuenta que los dispositivos son de uso delicado y aunque hay que tener cuidado de no dañarlo, también hay que asegurarse de que se estén haciendo bien las conexiones para que el funcionamiento sea el correcto. Para asegurar que la transmisión entre los dispositivos se esté dando es necesario que los cables estén bien conectados, para lo cual sugerimos comprar cable UTP de mayor grosor para evitar rupturas internas dentro de los orificios del protoboard que produzcan una falla en la comunicación.
2. Es necesario que los cables que son conectados a la computadora estén funcionando correctamente, tal es el caso del cable de conexión USB que energiza y ayuda a la programación de la tarjeta LPC1769, asimismo con la comunicación USART entre el kit de desarrollo AVR Butterfly y la computadora. [6]

3. La importancia de unir los niveles de referencia de la tarjeta LPC1769, el kit de desarrollo AVR Butterfly y del sensor fue clave esencial para lograr la meta debido a que aunque los tres son energizados de distintas fuentes y tienen que tener una conexión común para que funcionen correctamente.

XI. REFERENCIAS

- [1]. Padmaraja Yedamale, Microchip Technology Inc. – AN885: Brushless DC (BLDC) Motor Fundamentals Fecha de consulta: 06/03/12.
- [2]. Martin THOMAS, AVR-Projects, http://gandalf.arubi.uni-kl.de/avr_projects/
Fecha de consulta: 08/03/12.
- [3]. ATMEL, ATMEGA169 datasheet, www.atmel.com/dyn/resources/prod_documents/doc2514.pdf
Fecha de consulta: 09/03/12.
- [4]. NXP Semiconductors, Cortex M3, Manual de Usuario 10470 http://www.nxp.com/documents/user_manual/UM10470.pdf
Fecha de consulta: 21/02/12.
- [5]. Topic 18c: Serial Peripheral Interface <http://ww1.microchip.com/downloads/en/DeviceDoc/70243b.pdf>
Fecha de Consulta: 09/03/12.
- [6]. Richard Leonel Guerrero Jumbo, KIT DE DESARROLLO AVR BUTTERFLY, DESARROLLO DE GUÍA DE PRÁCTICAS DE LABORATORIO Y TUTORIALES.
Fecha de Consulta: 09/03/12.
- [7] Causas de vibración de los motores de inducción de 2polos mediante análisis espectral



www.motors-electrics.com/pdf/Vibraciones.pdf

Fecha de Consulta: 09/03/12.

[8] Parallax, Sound Impact Sensor Documentation

<https://www.parallax.com/Portals/0/Downloads/docs/prod/sens/29132-SoundImpactSesnor-v1.0.pdf>

[9] Alan Kharsansky, Introducci_on a LPCXpresso y repaso del lenguaje C, Seminario de Sistemas Embebidos, Agosto 2011

http://laboratorios.fi.uba.ar/lse/seminario/material-2011/Sistemas_Embebidos-2011_2doC-Intro_a_LPCXpresso_y_repaso_lenguaje_C-Kharsansky.pdf

[10] Osmetec BLDC Motor

http://www.osmtec.com/bldc_motor_57bl.htm

[11] Mario Sacco, Motores Brushless o BLDC (Métodos de Accionamiento), Abril 2012

<http://www.neoteo.com/motores-brushless-bldc>