



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA

**“ALMACENAMIENTO DE DATOS DE TEMPERATURA DE MOTOR BLDC PARA
GRAFICACIÓN Y ANÁLISIS EN DISPLAYS DISPONIBLES EN TARJETA AVR BUTTERFLY Y EN
TARJETA CONTROLADORA LPCXPRESSO”**

Fernando Adrián Rodríguez González ⁽¹⁾, Tamara Alexandra Samaniego Soto ⁽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)}

grodrigu@fiee.espol.edu.ec ⁽¹⁾, tsamanie@fiee.espol.edu.ec ⁽²⁾, cvaldiv@fiee.espol.edu.ec ⁽³⁾

Resumen

El siguiente proyecto de graduación expone el diseño e implementación de un control de temperatura utilizando las tarjetas LPCXpresso y AVR Butterfly para así demostrar una de las diversas aplicaciones que tienen estas tarjetas.

Entre las herramientas de hardware y software encontraremos el microcontrolador ATmega169 de la tarjeta AVR Butterfly, así como su respectiva configuración de puertos tanto de entrada como de salida, la pantalla LCD en la cual visualizaremos los datos de temperatura; la tarjeta LPCXpresso la cual nos ayudará con su convertidor análogo digital (ADC) a leer correctamente los datos de temperatura de tal forma que luego los podamos enviar por medio del protocolo de comunicación I2C. Además de los equipos antes mencionados también utilizaremos un sensor de temperatura LM35 el cual que estará midiendo la temperatura de un motor BLDC 42BLF01.

Finalmente los datos serán mostrados en el LCD que está disponible en la tarjeta AVR Butterfly.

Palabras Claves: Tarjeta LPCXpresso, Tarjeta AVR Butterfly, Sensor LM35, Protocolo I2C

Abstract

The following graduation project presents the design and implementation of temperature control using the LPCXpresso and AVR Butterfly cards to demonstrate one of the various applications that have these cards. Between the hardware and software tools you will find the ATmega169 microcontroller AVR Butterfly card, as well as their respective port configuration both input and output, the LCD screen where we will view the temperature data; the LPCXpresso card which will help us with its analog digital converter (ADC) to read the temperature data so that we can then send them through the I2C communication protocol. In addition to the equipment mentioned above an LM35 temperature sensor is used which will be measuring the temperature of a 42BLF01 BLDC motor.

Finally the data will be displayed on the LCD that is available on the AVR Butterfly card.

Keywords: LPCXpresso Target, AVR Butterfly Target, Sensor LM35, I2C protocol.

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA

1. Introducción

El uso de microcontroladores en nuestro entorno nos ayuda en varias aplicaciones en lo que respecta a lenguajes de alto nivel así como también a optimizar recursos de hardware.

El propósito de nuestro proyecto es dar a conocer la comunicación I2C entre la tarjeta de desarrollo LPCxpresso y AVR Butterfly

2. Limitación

La limitación que tenemos con nuestro proyecto es de tipo operacional debido a que siendo en el motor un BRUSHLESS (Motor sin escobillas) no puede ser forzado para aumentar su temperatura. Por tal motivo, las pruebas se las han realizado externamente sensando la temperatura de un cautín.

3. Herramientas de software

Para la realización de nuestro proyecto hicimos uso principalmente de 2 programas que los mencionamos y detallamos a continuación: AVR Studio 4 y LPCxpresso 4. Hay un tercer software que nos ayudó al momento de realizar pruebas con la tarjeta AVR Butterfly que es PROTEUS.

3.1. AVR Studio 4 [1]

AVR Studio 4 es un entorno de desarrollo integrado (IDE) para escribir y depurar aplicaciones AVR en el entorno de Windows. Soporta varias de las fases por las cuales se atraviesa al crear un nuevo producto basado en un microcontrolador AVR.

Este software ayuda al programador en el diseño, desarrollo, depuración y parte de la comprobación del proceso. El AVR Studio nos presta facilidad de manejo así como también una rápida ejecución al momento de simular cualquier proyecto.

AVR Studio 4 proporciona herramientas para la administración de proyectos, edición de archivo fuente, simulación del chip e interfaz para

emulación In-circuit para la poderosa familia RISC de microcontroladores AVR de 8 bits.



Fig.3.1: Herramienta de software AVR Studio 4

3.2 LPCxpresso 4 [2]

LPCxpresso es una nueva plataforma de desarrollo de bajo costo disponible para NXP. Es compatible con microcontroladores basados en ARM LPC de NXP. La plataforma está compuesta por un IDE basado en eclipse y tiene una tarjeta de bajo costo que incluye un depurador JTAG.

Diseñado para la simplicidad y facilidad de uso, el LPCxpresso IDE proporcionará a los ingenieros de software una forma rápida y sencilla de desarrollar sus aplicaciones.

3.3. PROTEUS

Proteus es un software basado en aplicaciones de diseño en el cual también podemos realizar simulaciones. Consta de dos módulos: ARES e ISIS.

ARES.- es la herramienta de rutado de PROTES, se utiliza para la fabricación de placas de circuito impreso, esta herramienta puede ser utilizada de manera manual o dejar que el propio programa trace las pistas. Con este módulo también podemos tener una visualización en 3D del PCB que se ha diseñado.

ISIS.- es la herramienta en la cual podemos diseñar el circuito que deseamos con componentes muy variados, desde una simple resistencia hasta algún que otro microprocesador o microcontrolador, incluyendo fuentes de alimentación, generadores de señales y muchas otras prestaciones. Los diseños

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA

realizados en ISIS pueden ser simulados en tiempo real. Una de estas prestaciones es VSM, que es una extensión de la aplicación con la cual podemos simular, en tiempo real, todas las características de varias familias de microcontroladores, introduciendo nosotros mismos el programa que queramos que lleven a cabo. ISIS es el corazón del entorno integrado PROTEUS. Es mucho más que un simple programa de dibujo de esquemas electrónicos.



Fig. 3.3: Herramienta de software Proteus

4. Herramientas de hardware

Las partes tangibles de nuestro proyecto serán las tarjetas de desarrollo tanto de la familia de NXP como de la familia ATMEL. Estas son: LPCXpresso que tiene integrado un chip LPC1769 y la AVR Butterfly que está formado por un microcontrolador ATmega169PV y el sensor de temperatura LM35.

4.1. LPC 1769

La LPCXpresso es una tarjeta de evaluación completa para el desarrollo con microcontroladores de NXP.

Contiene:

- LPCXpresso IDE y "development tools".
- IDE basado en Eclipse.
- Ompiler y linker GNU.
- DGB debugger.
- LPCXpresso target board (stick).

El target board es un microcontrolador con todo lo necesario para encender y también es una herramienta que incluye un debugger y un programador.

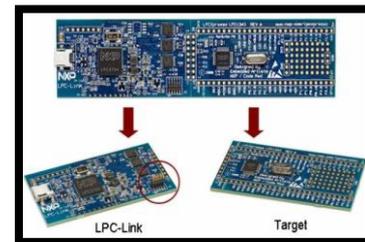


Fig. 4.1: Tarjeta LPCXpresso.

Del lado de la tarjeta están incluidos algunos periféricos básicos y se comercializan con diferentes tipos de microcontroladores. Para nuestro caso usaremos la LPC1769: ARM Cortex-M3, 512KB flash, 64KB SRAM, Ethernet, USB On the go.

4.2. AVR Butterfly [3]

El AVR Butterfly utiliza un microcontrolador AVR ATmega169PV, que combina la tecnología flash con el más avanzado y versátil microcontrolador de 8 bits disponible.

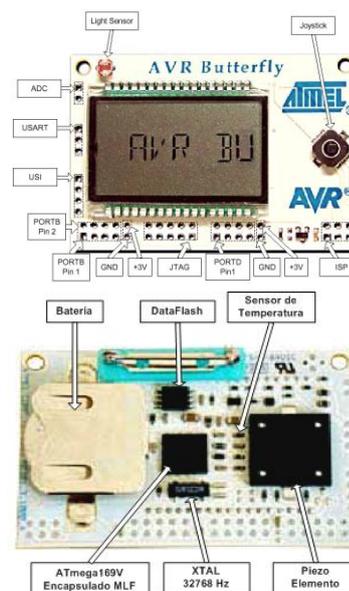


Fig. 4.2: Hardware disponible en el kit AVR Butterfly.



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA

Una de sus principales características en su diseño ya que tiene un bajo consumo de energía. Entre los recursos que nos brinda esta tarjeta, tenemos:

- Microcontrolador ATmega169PV.
- Pantalla LCD.
- Joystick.
- Cristal de 32KHz.
- Termistor.
- Bootloader, entre otros.

Es necesario contar con una batería de 3V para poner a alimentar esta tarjeta.

El ATmega169 es un Microcontrolador de 8 bits con arquitectura AVR RISC, este posee las siguientes características:

- Arquitectura RISC avanzada.
 - ❖ Conjunto de 130 instrucciones ejecutables en un solo ciclo de reloj.
 - ❖ 32 x 8 registros de trabajo de propósito general.
 - ❖ Rendimiento de hasta 16 MIPS a 16 MHz.
- Memoria no volátil para Programa y Datos.
 - ❖ Flash de 16 K bytes, auto-programable en el sistema.
 - ❖ Resistencia: 10 000 ciclos de Escritura/Borrado.
 - ❖ Sección Opcional para Código de Arranque con Lock Bits Independientes.
 - ❖ Programación en el Sistema a través del Programa de Arranque residente en el chip.
 - ❖ Operación Real de Lectura Durante la Escritura.
 - ❖ EEPROM de 512 bytes.
 - ❖ Resistencia: 100 000 ciclos de Escritura/Borrado.
 - ❖ SRAM Interna de 1 Kbyte.
 - ❖ Bloqueo de Programación para Seguridad del Software.
- Interfaz JTAG (conforme el Standard IEEE 1149.1)

- ❖ Capacidad de Boundary-Scan Acorde al Standard JTAG.
- ❖ Soporta Depuración On-chip.
- ❖ Programación de la FLASH, EEPROM, Fusibles y Lock Bits a través de la Interfaz JTAG.
- Características de los Periféricos.
 - ❖ 6 puertos de I/O de 8-bits y 1 de 5-bits.
 - ❖ Controlador de LCD de 4 x 25 segmentos.
 - ❖ Dos Temporizadores/Contadores de 8 bits con Preajustador (Prescaler) separado y Modo de comparación.
 - ❖ Un Temporizador/Contador de 16 bits con Preajustador (Prescaler) separado, Modo de Comparación y Modo de Captura.
 - ❖ Contador de Tiempo Real con Oscilador Separado.
 - ❖ Cuatro canales PWM.
 - ❖ Ocho canales ADC de 8 bits cada uno.
 - ❖ Interfaz Serial USART Programable.
 - ❖ Interfaz Serial SPI Maestro/Esclavo.
 - ❖ Interfaz Serial Universal con Detector de Condición de Inicio.
 - ❖ WatchDogTimer Programable con Oscilador Separado incluido en el chip.
 - ❖ Comparador Analógico.
 - ❖ Interrupción y Salida del Modo de Sleep por Cambio en Pin.
- Características especiales del microcontrolador.
 - ❖ Reset al Encendido y Detección Brown-Out Programable.
 - ❖ Oscilador Interno Calibrable.
 - ❖ Fuentes de Interrupción Internas y Externas.
 - ❖ Cinco modos de Sleep: Idle, ADC Noise Reduction, Power Save, Power-Down y Standby.
- Entradas/Salidas y Tipo de Encapsulado
 - ❖ 53 Líneas de I/O Programables.

- ❖ 64 patillas en el encapsulado TQFP y 64 pads (para montaje superficial) en el encapsulado QFN/MLF.
- Rangos de Velocidad
 - ❖ ATmega169V: 0–4 MHz a 1.8–5.5 V, – 8 MHz a 2.7–5.5 V
 - ❖ ATmega169: 0–8 MHz a 2.0–5.5 V, 0–16 MHz a 4.5–5.5 V
- Rango de Temperatura
 - ❖ Desde -40 ° C a 85 °C.
- Consumo de Energía
 - ❖ En el Modo Activo:
 - 1 MHz, 1.8 V: 350 uA.
 - 32 KHz, 1.8 V: 20 uA (incluyendo Oscilador).
 - 32 KHz, 1.8 V: 40 uA (incluyendo Oscilador y LCD)
 - ❖ En el Modo Power-Down:
 - 0.1uA a 1.8 V.

4.3. Sensor de temperatura LM35 [4]

Dado que nuestro proyecto trata del sensamiento de temperatura de un motor BLDC, haremos uso del sensor de temperatura LM35; la cualidad de este sensor es que su medición es en °C con un factor de escala lineal de 10.0 mV/°C.

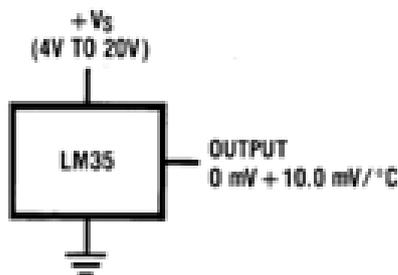


Fig. 4.3: Configuración del sensor de temperatura

5. Protocolo de comunicación.

El protocolo de comunicación que hemos utilizado para el desarrollo de nuestro proyecto es el I2c, del cual hablaremos a continuación.

5.1. Definición

Es un bus serial de comunicaciones compuesto de dos líneas: SDA y SCL. Este protocolo es muy utilizado en la actualidad en industrias principalmente para comunicar microcontroladores o microprocesadores con otros dispositivos que permitan este protocolo.

5.2. Funcionamiento del I2C

Como se menciona en la definición, este protocolo está compuesto por 2 líneas las cuales son drenador abierto por lo cual es necesaria la utilización de resistencias pull-up.

SDA: es la línea por la cual se transmitirán los datos hacia otro dispositivo.

SCL: es la señal del reloj.

Existen dos tipos de configuración para este protocolo:

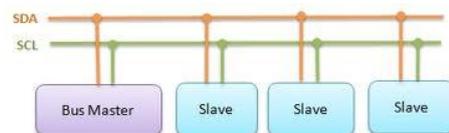


Fig. 5.2(a) Maestro – Esclavo

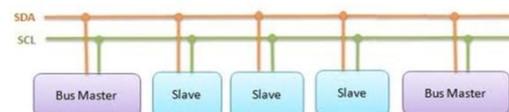


Fig. 5.2(b): Multimaestro

5.3. Transmisión de datos

Para realizar la transmisión hay que tener en cuenta que los bits de datos serán enviados por el SDA, para la transmisión de cada bit es necesario un pulso de reloj, es decir, que exista un cambio en la SCL. Para realizar un cambio de bit en la transmisión el estado de SCL debe ser en bajo.

En los gráficos siguientes mostraremos las condiciones y el esquema para la activación, transmisión y parada de datos del protocolo de comunicación I2C, así como también una breve explicación de los mismos.

Para iniciar la transmisión es necesaria la activación del bus, es decir, el envío del START, esto pondrá en alerta a todos los dispositivos esclavos que se encuentran conectados al bus

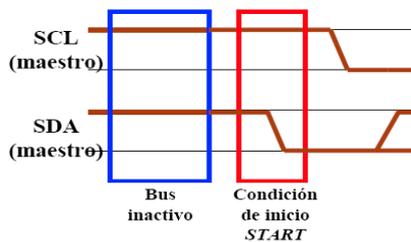


Fig. 5.3(a): Condición de START.

Luego de la señal de START el dispositivo envía una dirección, de esta manera se podrá identificar al esclavo con el cual se comunicará, así solamente este esclavo podrá contestar ante las señales del maestro.

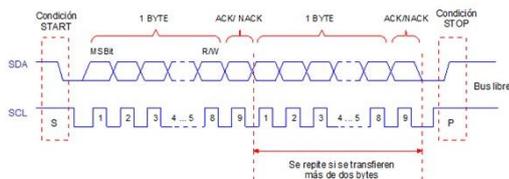


Fig. 5.3(b): Esquema de la transmisión de datos.

Para finalizar la transmisión es necesaria la condición de STOP la cual es de la siguiente manera:

SDA: para a estado alto mientras el SCL se encuentra en alto.

Recordemos que mientras la señal de SCL se encuentre en alto no hay transmisión de datos.

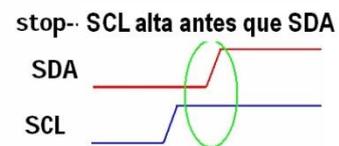


Fig. 5.3(c): Condición de STOP

6. Proyecto terminado y funcionando

Durante la realización de nuestro proyecto, hicimos uso de la comunicación I2C entre las tarjetas AVR Butterfly y LPCXpresso. La LPCXpresso la utilizamos como transmisor, es decir, que está configurada como master y hemos hecho uso para recibir los datos o sea configurada como esclavo la tarjeta AVR Butterfly, los datos que serán transmitidos se podrán visualizar la LCD de esta tarjeta.

A continuación mostraremos mediante diagramas de bloques y flujo la configuración de nuestro proyecto.

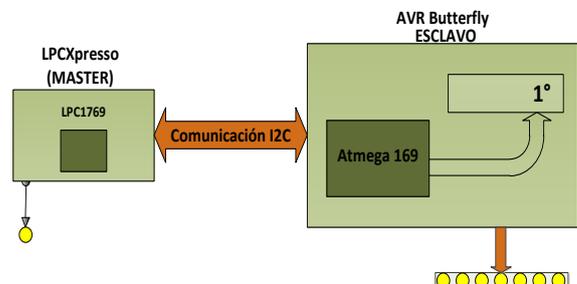


Fig. 6.1: Diagrama de bloques

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA

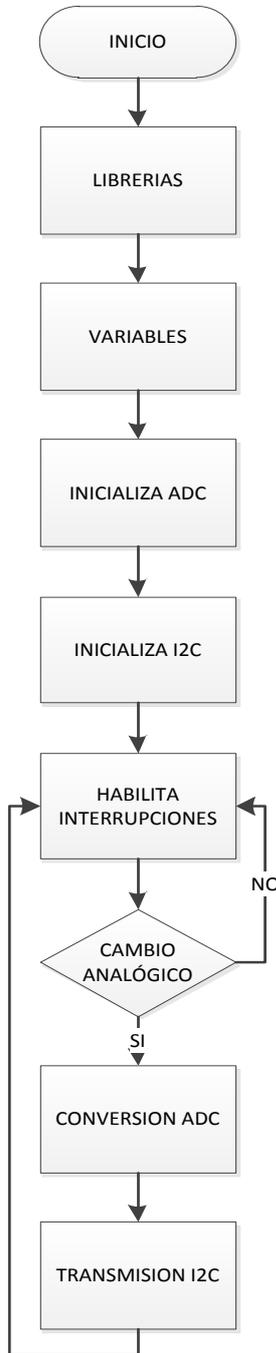


Fig. 6.2: Diagrama de flujo del master (LPCXpresso)

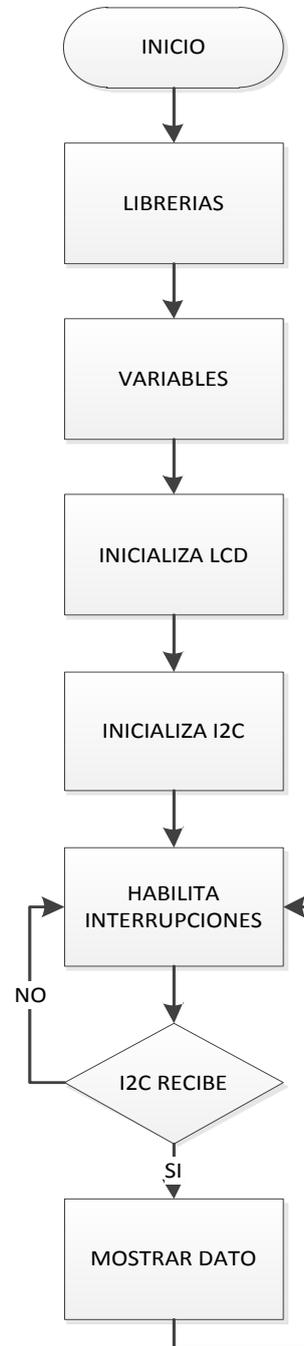


Fig. 6.3: Diagrama de flujo del esclavo (AVR Butterfky)

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA

En esta figura se muestra el proyecto ya funcionando. Como podemos apreciar, tenemos: Las tarjetas AVR Butterfly, la LPCXpresso el sensor, led el cual nos indica cuando termino la transmisión, entre otros.

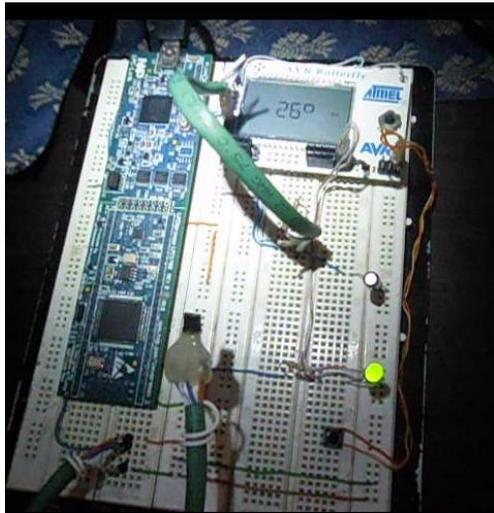


Fig. 6.4: Proyecto funcionando.

CONCLUSIONES

- 1) El protocolo I2C es muy útil para transmisión de datos entre dispositivos así sean de diferente clase, en este caso hemos realizado esta comunicación entre las tarjetas AVR BUTTERFLY y LPCXpresso con gran éxito; su comprobación se efectuó tomando datos de temperatura mediante un sensor LM35 a un motor BLDC y visualizándolos en una LCD propia de la tarjeta AVR BUTTERFLY. Algo muy importante es recalcar que estas tarjetas cuentan con microcontroladores que tienen módulos internos para la transmisión y recepción por I2C.
- 2) La sincronización en la comunicación de las tarjetas por medio del protocolo I2C fue algo muy necesario dado que se nos presentaron algunos inconvenientes debido a las velocidades con las que trabajan cada una de

ellas, dicho problema fue muy bien superado gracias a diferentes validaciones que le permitía tanto al maestro como al esclavo cuando era transmisión y cuando se podía enviar otro dato.

- 3) Fue de gran ayuda el que la tarjeta LPCXpresso tenga incluido en su software ejemplos, ya que estos fueron de gran ayuda aprender a utilizarla de una mejor manera así como también en la realización de nuestro proyecto.
- 4) La característica de controlador LCD, del ATmega169, permite abaratar costos en la implementación de aplicaciones que necesitan despliegue de información a través de LCD. Esta característica permite controlar pantallas de cristal líquido de bajo costo, que por ser básicas no poseen ni driver interno, ni interfaces de comunicación como la mayoría de los costosos módulos LCD.

RECOMENDACIONES

- 1) No apoyar el Kit AVR Butterfly en superficies conductoras tales como metal, líquidos, etc., puesto que podrían causar daños en el mismo.
- 2) Se recomienda colocar espadines hembra o macho en los espacios destinados para conexiones externas de este kit en lugar de colocar cables directamente ya que esto podría causar un corto circuito.
- 3) Obtener información suficiente sobre los microcontroladores a usar en este caso AVR Butterfly y LPCXpresso para conocer la correcta distribución y funcionamiento de cada uno de los pines.
- 4) Se recomienda revisar el datasheet de todos los componentes para así revisar la configuración y conexión de todos sus pines para evitar cualquier daño irreversible en sus componentes.



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA

BIBLIOGRAFIA

[1] AVR Butterfly Evaluation kit – User Guide

<http://www.atmel.com/Images/doc4271.pdf>

Nombre del autor: ATMEL

Fecha de consulta: Marzo 2012

[2] LPCxpresso –Ics – NXP Semiconductors

<http://ics.nxp.com/lpcxpresso/>

Nombre del autor: NXP Semiconductors

Fecha de consulta: Marzo 2012

[3] Características del kit AVR Butterfly en español

<http://www.espe.edu.ec/repositorio/T-ESPE-014271.pdf>

<http://www.datasheetcatalog.org/datasheet/atmel/2514S.pdf>

Nombre del autor: ATMEL Corporation 2003

Fecha de consulta: Marzo 2012

[4] LM35 – Sensor de temperatura de precisión en grados centígrados.

<http://electronica.webcindario.com/componentes/lm35.htm>

Nombre del autor: Carlos Díaz

Fecha de consulta: Abril 2012