



**“UTILIZACION DE MICROCONTROLADORES ATMEL PARA LA REALIZACION DE HARDWARE DEMOSTRATIVO DE DIVERSOS TIPOS DE INTERRUPCIONES INTERNAS Y EXTERNAS DE LOS MICROCONTROLADORES ATMEL”
(Febrero 2012)**

Jorge Vega Rosero ⁽¹⁾, Giovanni Granados Ortiz ⁽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)}
jvega@fiec.espol.edu.ec ⁽¹⁾, ggranados@fiec.espol.edu.ec ⁽²⁾, cvaldiv@fiec.espol.edu.ec ⁽³⁾

Resumen

Este artículo tiene como objetivo entregar una guía práctica y material indispensable para todo estudiante que desee conocer sobre la Tecnología de los Microcontroladores Atmel.

Este proyecto se centra en la Utilización de las Interrupciones del Microcontrolador ATMEGA169 y su flexibilidad para ser implementadas en kit de evaluación AVR BUTTERFLY.

Aquí se tratan algunas de las interrupciones importantes y su implementación que serán de guía para futuros proyectos y mejores desarrollos.

Los problemas implementados aquí se realizaron en el programa AVR 4 Studio, tanto en Lenguaje ASEMBLER y en lenguaje C. Su demostración y simulación están para ser probados en el Programa PROTEUS, y su implementación física se presenta en un Protoboard para hacerlo muy didáctico.

Palabras claves: AVR Butterfly, Plataforma Interactiva, interrupciones Amega169.

Abstract

This article aims at providing a practical guide and essential equipment for all students who wish to learn about the technology of the Atmel microcontroller.

This project focuses on the use of the interrupts of the ATMEGA169 microcontroller and its flexibility to be implemented in AVR Butterfly evaluation kit.

Here are some of the significant interrupts and its implementation that will be a guide for future projects and best developments.

The problems implemented here were carried out in the program AVR4 Studio, both in Assembly Language and C Language. Its demo and simulation are to be tested in the PROTEUS Program, and its physical implementation is presented in a Breadboard to do so very didactic.

Keywords: AVR Butterfly, Platform Interactive, interrupts of the ATMEGA169.



I. INTRODUCCION

El desarrollo de herramientas para aprendizaje para cualquier rama de conocimiento ha sido motivo de continua evolución, en la Electrónica, nuevos desarrollos aparecen cada día, estos desarrollos ameritan una inmediata adaptabilidad forzada para estar en un nivel competitivo ante otras sociedades.

Esta guía ayuda a que nos encaminemos específicamente a desarrollos iniciales con los Microcontroladores Atmel. El aporte es dar a dar a conocer el potencial de estos Microcontroladores. Y que en menor tiempo posible el estudiante interesado en esta tecnología desarrolle su potencial creativo en base a esta.

Este artículo centra su área de interés en las Interrupciones del Microcontrolador ATmega 169 Fabricado por Atmel, y que lo encontramos embebido en el kit de desarrollo del AVR Butterfly.

Se utiliza dos software muy funcionales, los cuales son el Avr Studio 4 de la misma fabricante del Avr Butterfly la Cía. Atmel este software lo utilizamos para la programación en lenguaje C y ensamblador, además de el Proteus 7.0 el cual nos permite simular nuestro proyecto de una forma animada y verificar su funcionamiento.

II. IMPORTANCIA DE LAS INTERRUPCIONES

Las Interrupciones son un método del que disponen los Microcontroladores, para atender alguna circunstancia que requiera su inmediata atención. Al presentarse un pedido de interrupción el Microcontrolador da por terminado cualquier instrucción en curso, y hará un salto a una subrutina de Interrupción, una vez terminada esta subrutina, volverá a su labor anterior.

Si no existieran las Interrupciones el mundo como lo conocemos ahora no sería el mismo. Tendríamos que esperar que el Microcontrolador nos atiende a nosotros en nuestro requerimiento cuando a él le parezca conveniente, con las interrupciones el nos atiende cuando nosotros queremos y de manera inmediata. De ahí su gran importancia.

Realmente si no existieran las interrupciones la programación se realizaría de manera secuencial, y para atender algún requerimiento, este debería esperar su turno. Por ejemplo si tuviéramos una alarma de incendios, para un edificio de 10 pisos, tendríamos que consultar a cada sensor uno a uno su

estatus, si se genera una señal de alarma en el piso 10, el Microcontrolador Primero tendría que revisar los otros 9 pisos antes de atender el número 10. Ocasionando un retraso de decenas de microsegundos, algo que para un incendio no es apreciable, pos igual el Microcontrolador Funcionara eficientemente. Pero hay otras situaciones en que la respuesta inmediata es fundamental y los microsegundos son esenciales. Como en la detección de fallas eléctricas, o los nuevos sistemas de inyección y combustión de automóviles electrónicos.

A. CARACTERISTICAS DE LAS INTERRUPCIONES EN EL ATMEGA 169

El Microcontrolador Atmega dispone de 23 Interrupciones, no todas disponibles para ser utilizadas en el Avr Butterfly debido a que sus puertos están comprometidos con la pantalla LCD y Joystick. Estas Interrupciones son las Siguientes:

1. PIN EXTERNO RESET
2. INTERRUPCION EXTERNA INTO
3. INTERRUPCION EXTERNA POR CAMBIO EN PIN
4. INTERRUPCION EXTERNA POR CAMBIO EN PIN
5. TIMER/COUNTER2 COMPARACION CON VALOR DEFINIDO
6. TIMER/COUNTER2 POR DESBORDE
7. TIMER/COUNTER1 COMPARACION CON VALOR DEFINIDO
8. TIMER/COUNTER1 COMPARACION CON VALOR DEFINIDO EN A
9. TIMER/COUNTER1 COMPARACION CON VALOR DEFINIDO EN B
10. TIMER/COUNTER1 POR DESBORDE
11. TIMER/COUNTER0 COMPARACION CON VALOR DEFINIDO
12. TIMER/COUNTER0 POR DESBORDE
13. SPI TRANSMISION SERIAL COMPLETA
14. USART, RX COMPLETA
15. USART, REGISTRO DE DATO VACIA
16. USART, TX COMPLETA
17. USI CONDICION DE INICIO
18. USI DESBORDE
19. COMPARADOR ANALOGICO
20. ADC, CONVERSION COMPLETA
21. EEPROM LISTA
22. ALMACENAMIENTO EN MEMORIA DE PORGRAMA LISTA
23. LCD INICIO DE FRAME.

B. CONECTORES DEL AVR BUTTERFLY

Algunos de los pines de I/O del micro-controlador ATmega169 están disponibles en los conectores Del AVR Butterfly. Estos conectores son para comunicación, programación y entrada al ADC del ATmega169. En la Fig 1. se muestra los conectores del AVR Butterfly.

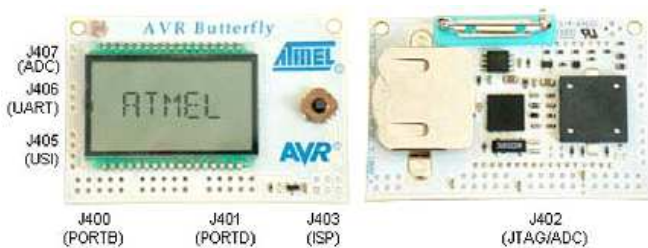


Fig. 1. Conectores del AVR Butterfly para acceso a periféricos

III. RECURSOS DE HARDWARE Y SOFTWARE DEL MICROCONTROLADOR ATMEL AVR

INTRODUCCION

ATMEL fabrica los micro-controladores de la familia AVR, esta nueva tecnología proporciona todos los beneficios habituales de arquitectura RISC (Reduce dinstruction set computer) y memoria flash reprogramable eléctricamente. La característica que los identifica a estos micro-controladores de ATMEL es la memoria flash y eeprom que incorpora. AVR compite con varias familias de micro-controladores bien establecidas en el mercado, tales como 8051 de Intel, 68HC11 de Motorola y la familia PIC de Microchip. La firma también produce y vende varios subproductos de la popular familia 8051 con la diferencia de que están basados en la memoria flash.

Adicionalmente, ATMEL también proporciona en línea el entorno software (AVR estudio) que permite editar, ensamblar y simular el código fuente. Una vez ensamblado y depurado el código fuente del programa, se transferirá el código máquina a la memoria flash del micro-controlador para esto se debe disponer de otro entorno de desarrollo para programar en forma serial o paralelo la memoria flash.

A continuación se va a tratar sobre los recursos de hardware y software disponibles en la familia de microcontroladores de ATMEL, tales como Avr Studio 4, WinAvr, Avr Butterfly, las principales características del hardware y el firmware, además

del Proteus que es una herramienta de software muy necesaria para la simulación de los proyectos.

AVR STUDIO 4

AVR Studio es un Entorno de Desarrollo Integrado (IDE) para escribir y depurar aplicaciones AVR en el entorno de Windows 9x/Me/NT/2000/XP/7. Ver Fig.3.

AVR Studio 4 soporta varias de las fases por las cuales se atraviesa al crear un nuevo producto basado en un micro-controlador AVR. Las fases típicas son:

- La definición del producto. El producto que debe crearse se define basándose en el conocimiento de la tarea que se quiere resolver y la entrada que tendrá en el mercado.

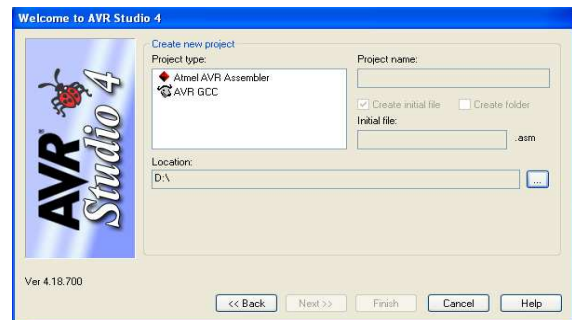


Fig.III. Entorno de AVR STUDIO 3

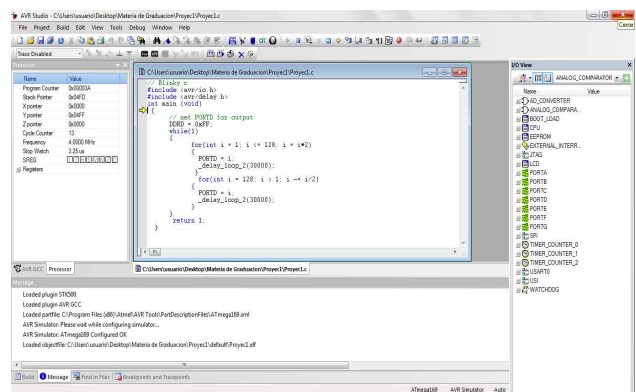


Fig. 4. Ventana Principal IDE

- La especificación formal. Se define una especificación formal para el producto.
- Asignación de la tarea a un equipo. A un equipo del proyecto, que consiste de una o más personas, se le asigna la tarea de crear el producto basándose en la especificación formal.

- El equipo del proyecto pasa por la secuencia normal de diseño, desarrollo, depuración, comprobación, planificación de producción, prueba y embarque.

Como se dijo anteriormente, el AVR Studio es un Entorno de Desarrollo Integrado (IDE). Éste tiene una arquitectura modular completamente nueva, que incluso permite interactuar con software de otros fabricantes.

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 micro-controladores AVR de 8 bits.

AVR Studio 4 consiste de muchas ventanas y sub-módulos. Cada ventana apoya a las partes del trabajo que se intenta emprender. En la Fig. 4. se puede apreciar las ventanas principales del IDE.

WINAVR

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.

El programador vía puerto serie mprog.exe, que permite transferir el programa compilado, que se encuentra en un archivo llamado main.hex, a la memoria flash del micro-controlador utilizando únicamente un cable de tres líneas.

En pocas palabras WinAVR es un conjunto de herramientas de desarrollo para micro-controladores RISC AVR de Atmel, basado en software de código abierto y compilado para funcionar en la plataforma Microsoft Windows. WinAVR incluye las siguientes herramientas:

- Avr-gcc, el compilador de línea de comandos para C y C++.
- Avr-libc, la librería del compilador que es indispensable para avr-gcc.
- Avr-as, el ensamblador.
- Avrdude, la interfaz para programación.
- Avarice, la interfaz para JTAG ICE.
- Avr-gdb, el depurador.
- Programmers Notepad, el editor.
- MFile, generador de archivo makefile.

AVR BUTTERFLY

El Kit AVR Butterfly Fig. 2. se diseñó para demostrar los beneficios y las características importantes de los micro-controladores ATMEL. El AVR Butterfly utiliza el micro-controlador AVR ATmega169V, que combina la Tecnología Flash con el más avanzado y versátil micro-controlador de 8 bits disponible.

El Kit AVR Butterfly expone las siguientes características principales:

- ✓ La arquitectura AVR en general y la ATmega169 en particular.
- ✓ Diseño de bajo consumo de energía.
- ✓ El encapsulado tipo MLF.
- ✓ Periféricos:
 - Controlador LCD.
 - Memorias:
 - Flash, EEPROM, SRAM.
 - DataFlash externa.
- ✓ Interfaces de comunicación:
 - UART, SPI, USI.
- ✓ Métodos de programación
 - Self-Programming/Bootloader, SPI, Paralelo, JTAG.
- ✓ Convertidor Analógico Digital (ADC).
- ✓ Timers/Counters:
 - Contador de Tiempo Real (RTC).
 - Modulación de Ancho de Pulso (PWM).

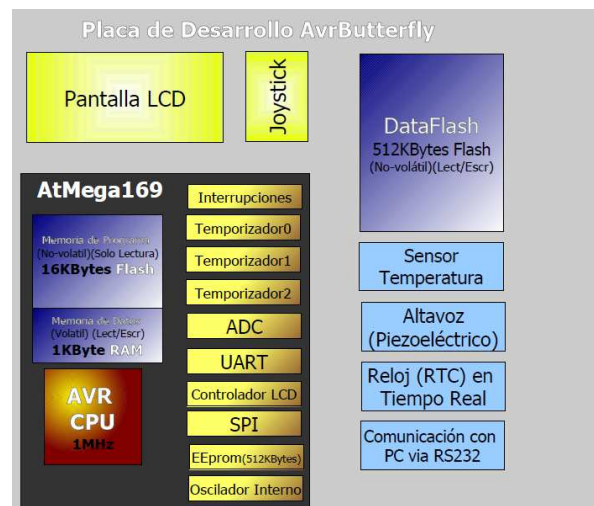


Fig. 6. Diagrama de bloques Avr Butterfly

El AVR Butterfly está proyectado para el desarrollo de aplicaciones con el ATmega169 y además puede usarse como un módulo dentro de otros productos.



HARDWARE DISPONIBLE

Los siguientes recursos están disponibles en el Kit AVR Butterfly, tal como se muestra en el diagrama de bloques de la Fig. 6:

- Micro-controlador ATmega169V (en encapsulado tipo MLF).
- Pantalla tipo vidrio LCD de 120 segmentos, para demostrar las capacidades del controlador de LCD incluido dentro del ATmega169.
- Joystick de cinco direcciones, incluida la presión en el centro.
- Altavoz piezoeléctrico, para reproducir sonidos.
- Cristal de 32 KHz para el RTC.
- Memoria DataFlash de 4 Mbit, para el almacenar datos.
- Convertidor de nivel RS-232 e interfaz USART, para comunicarse con unidades fuera del Kit sin la necesidad de hardware adicional.
- Termistor de Coeficiente de Temperatura Negativo (NTC), para sensar y medir temperatura.
- Resistencia Dependiente de Luz (LDR), para sensar y medir intensidad luminosa.
- Acceso externo al canal 1 del ADC del ATmega169, para lectura de voltaje en el rango de 0 a 5 V.
- Emulación JTAG, para depuración.
- Interfaz USI, para una interfaz adicional de comunicación.
- Terminales externas para conectores tipo Header, para el acceso a periféricos.
- Batería de 3 V tipo botón (600mAh), para proveer de energía y permitir el funcionamiento del AVR Butterfly.
- Bootloader, para programación mediante la PC sin hardware especial.
- Aplicación demostrativa pre-programada.
- Compatibilidad con el Entorno de Desarrollo AVR Studio 4.

FIRMWARE INCLUIDO

El AVR Butterfly viene con una aplicación pre-programada. Esta sección presentará una revisión de los elementos de esta aplicación.

Los siguientes bloques vienen preprogramados en el AVR Butterfly:

- ✓ Código Cargador de Arranque (BootloaderCode).
- ✓ Código de la Aplicación.
 - ❖ Máquina de Estados
 - ❖ Funciones incluidas:
 - Nombre-etiqueta.

- Reloj (fecha).
 - Mediciones de temperatura.
 - Mediciones de luz.
 - Lecturas de voltaje.
 - Reproducción de tonadas/melodías.
 - Ahorro de energía automático.
 - Ajuste de contraste del LCD.
- ❖ Más funciones podrán ser agregadas después, como por ejemplo:
- Calculadora.
 - Función de recordatorio.
 - Alarma (alarmas diarias, temporizadores para la cocina, etc.).
 - Reproducción de melodías y visualización del texto (función de Karaoke).
 - Con la DataFlash de 4 Mb el usuario podrá almacenar una cantidad grande de datos.

PROTEUS

Proteus es una aplicación CAD, compuesta de tres módulos:

- ISIS (Intelligentschematic input system): Es el módulo de captura de esquemas.
- VSM (Virtual sistemmodelling): Es el módulo de simulación incluyendo Prospice.
- ARES (Advancedroutingmodelling): Es el modulo para la realización de circuitos impresos (PCB).

El modulo ISIS es un programa que nos permite dibujar sobre un área de trabajo un circuito que posteriormente podremos simular. [6]

En la manipulación del software casi siempre existirán varias opciones para un mismo fin, normalmente podremos optar por seguir un menú, acceder a un icono o trabajar en el teclado.

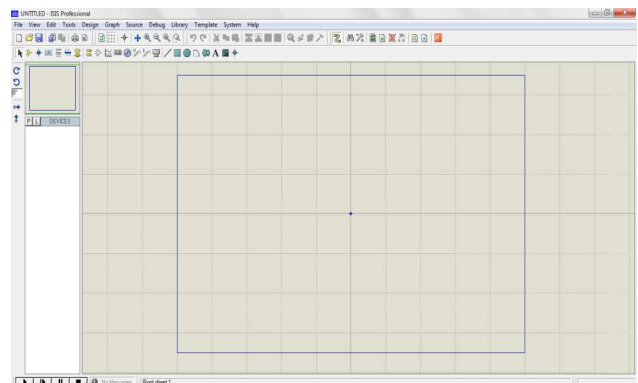


Fig. 7. Ventana de Trabajo de Proteus

El menú permite acceder a la mayor parte de opciones del programa, sin embargo algunas solo están disponibles en los iconos de las barras de herramientas. Las barras de herramientas son varias y se pueden colocar en cualquier parte de la pantalla. El área de trabajo es donde realizaremos nuestros circuitos, ver Fig. 7. [7]

Al abrir Proteus, lo primero que necesitamos es extraer los componentes que se van a utilizar en el circuito, para lo que debemos usar la barra de herramientas de componentes. Para acceder a una librería hay que presionar P que se encuentra en el extremo izquierdo, en las librerías son donde se encuentran los componentes y a partir de aquí se empieza a seleccionar los que necesitamos. Luego de armado nuestro circuito se lo guarda y se lo puede simular.

IV. PLATAFORMAS UTILIZADAS PARA DESARROLLAR EL PROYECTO

INTRODUCCIÓN

Para el desarrollo de los proyectos básicos y mostrar su funcionalidad se armo sobre un protoboard una estación de trabajo, en donde el Avr Butterfly será nuestro controlador y el LCD nuestro visualizador de los eventos además, se implemento varios circuitos adicionales, como teclados exteriores y circuitos básicos como Driver de corriente para Encender Leds que ameritan mayor energía.

ESTACION DE TRABAJO EDUCATIVA PARA PRUEBAS EXPERIMENTALES

Es una placa de uso genérico reutilizable, o semipermanente usado para construir prototipos de circuitos electrónicos con o sin soldadura. Normalmente se utilizan para la realización de pruebas experimentales. Además de los protoboard plásticos, libres de soldadura, también existen en el mercado otros modelos de placas de prueba.

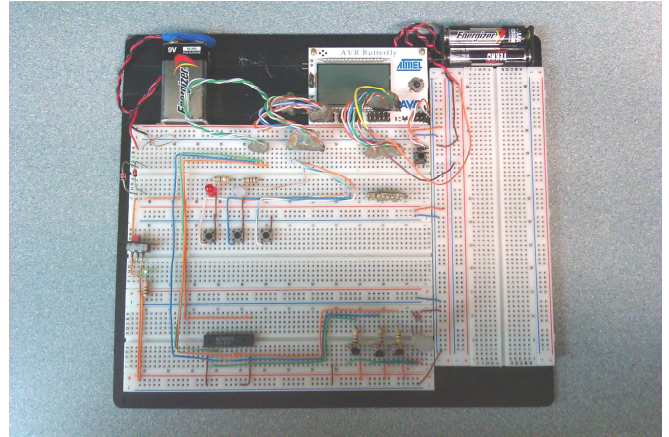


Fig. 7. Plataforma Interactiva

La tableta experimental, es una herramienta que nos permite interconectar elementos electrónicos, ya sean resistencias, capacidades, semiconductores, etc., sin la necesidad de soldar los componentes. Está lleno de orificios metalizados –con contactos de presión- en los cuales se insertan las componentes del circuito a ensamblar.

V. DESCRIPCIÓN DEL PROYECTO

INTRODUCCIÓN

A continuación vamos a enunciar los ejercicios que serán probados en la plataforma interactiva educativa, la cual va a permitir demostrar el funcionamiento de las Interrupciones, no todas las Interrupciones serán expuestas, sin embargo son las más utilizada para iniciar proyectos mas elaborados.

ESPECIFICACIÓN

PROYECTO 1

Elaborar un Programa en lenguaje C que habilite el Joystick del ArvButterfly, al presionar cualquier opción del Joystick este deberá Transmitir el nombre de la Opción escogida por medio de transmisión Usart. Por lo tanto si se presiona el Joystick hacia arriba deberá transmitir la palabra ARRIBA. y de esta manera para las otras 4 opciones restantes.

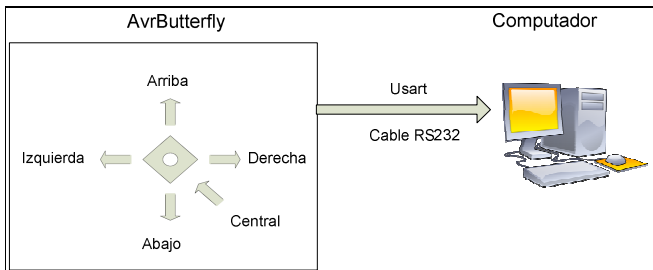


Fig. 8. Esquema de Proyecto 1

OBJETIVO 1

Manejo de sentencias en Asembler, configuración de Joystick del AVR BUTTERFLY, por medio de interrupciones, Utilización del Modulo Usart, para enviar dato referente a la Opción escogida por medio del Joystick.

PROYECTO 2

Elaborar un controlador continuo para Led Tricolor, por medio de PWM, este debe cambiar de color en forma tenue y aleatoria.

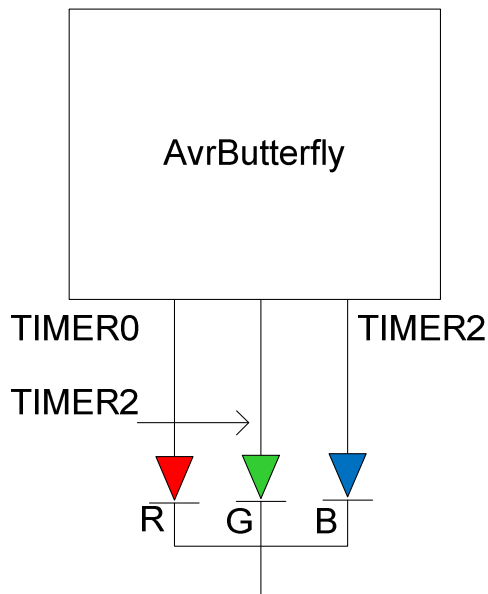


Fig. 9. Esquema de Proyecto 2

OBJETIVO 2

Mostrar el uso de las Interrupciones para generar PWM, y así poder generar una gama de colores a partir de un LED RGB.

PROYECTO 3

Elaborar en lenguaje C un programa que sea para apertura de una puerta electrónica, al ingresar la clave correcta se activara un led que simula la apertura de la puerta. Se debe implementar tres botones, correspondientes a los números 1,2,3.

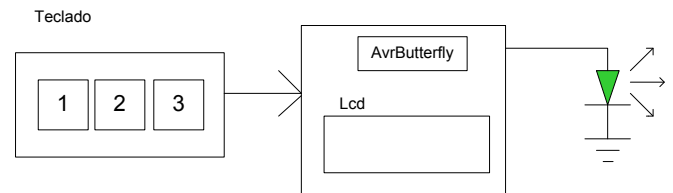


Fig. 10. Esquema de Proyecto 3

OBJETIVO 3

Utilización de interrupción por cambio en los Pórticos para el ingreso de números.

Ingreso de datos en Memoria SRAM.

Se debe ingresar 5 valores por medio del teclado numérico, al coincidir con la clave guardada previamente en la Memoria EEPROM del AvrButterfly deberá activarse el Led que representa la apertura de la cerradura Electrónica.

PROYECTO 4

Elaborar un programa en lenguaje C, que active un ventilador cuando la lectura del ADC0 ha alcanzado el valor Máximo decimal de 200. Y apagar el ventilador cuando el ADC0 ha alcanzado el valor mínimo decimal de 100, utilizar el NTC integrado del AvrButterfly. Ver grafica 3-19.

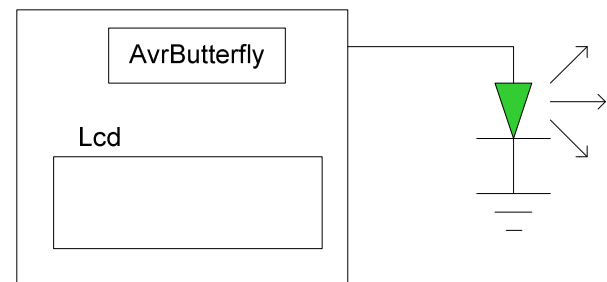


Fig. 11. Esquema de Proyecto 4

OBJETIVO 4

Configurar el convertidor analógico digital.
Realizar mediciones mediante el sensor TNC intergrado del AvrButterfly.
Activar un led, que representa el Encendido de un Ventilador.

PROYECTO 5

El circuito debe tener dos botoneras externas, para cada jugador, una tercera botonera que será la que da inicio al juego, además, el circuito constara de dos Led, estos indican el turno de cada jugador.
Cuando el jugador uno presiona el botón en su turno se genera un valor aleatorio y este es mostrado en el lcd.
Luego de este se dará paso para que se presione el botón del jugador 2.
Luego de esto será nuevamente el turno del jugador 1, y luego del jugador 2.
Ganara el que al sumar los valores de los dos dados obtenga el mayor puntaje.

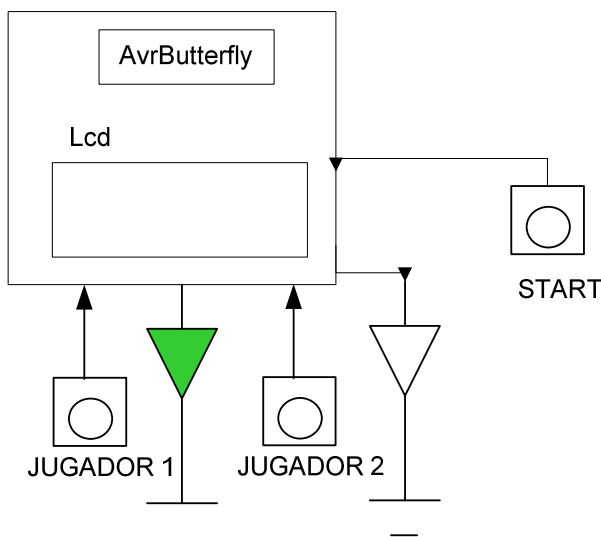


Fig. 11. Esquema de Proyecto 5

OBJETIVO 5.

Utilizar las interrupciones por cambio en los pórtilos para habilitación de botoneras externas.
Utilización de librería LCD, para mostrar valores aleatorios.

VI. CONCLUSIONES

1. Se implementaron 5 proyectos Didácticos mostrando el potencial del AvrBUtterfly, estos proyectos son orientados a mostrar la versatilidad de las Interrupciones Generales de las Microcontroladores ATMEL.
2. En las Líneas de comandos tanto en lenguaje Asembler como en el Leguaje C se describen la forma en que se habilitan las interrupciones, no se utilizaron todas de las 23 interrupciones que posee el Atmega169. Sin embargo con las aplicadas se deja abierto un camino para expandirse a las demás Interrupciones.
3. Con el uso del Microcontroladores AVR como nueva tecnología, se expanden las posibilidades de desarrollos más elaborados, y junto con la tecnología de Microchip se tiene más opciones a escoger, para futuros desarrollos. En donde el precio, y más que nada la disponibilidad de los elementos es fundamental para terminar cualquier proyecto.

VII. RECOMENDACIONES

1. Al inicio se hizo complicado, pues se relaciona la tecnología de Microchip con la ATMEL, cosa que se hizo algo confuso, el Atmega169 tiene muchas más set de Instrucciones, que al inicio las instrucciones diversas del Atmega pueden parecer las mismas, incluso pueden llegar hacer pensar al programador que es un hecho que su código está bien. Sin embargo, el aplicar las Instrucciones en assembler, hay que tener muy en cuenta la dirección de memoria de los Registros a Utilizar, como conclusión en esto les aseguro que primero hay que ver que registro se va a trabajar antes de aplicar cualquier instrucción y no al revés.
2. Complementando la recomendación descrita anterior, puedo decir que incluso el propio código expuesto en el datasheet del Atmega, NO FUNCIONA, pues al tratar de aplicarlo el compilador muestra errores. Como recomendación hay que darse cuenta que los registros del Atmega169 están en dos bancos, un banco va desde la dirección (0xFF) hasta la dirección (0x60). Por ejemplo en este banco se deben utilizar las Instrucción STS para escribir datos. Sin embargo esta misma instrucción no funciona para el Banco que va desde la Dirección 0x3F (0x5F) al 0x00 (0x20), aquí se debe utilizar la Instrucción OUT.



ESCUELA SUPERIOR POLITECNICA DEL LITORAL

FACULTAD DE INGENIERIA ELECTRICA Y COMPUTACION



3. En los Programas en lenguaje C, para las Instrucciones se utiliza SIGNAL(Vector) sin embargo modernamente se utiliza ISR(vector). igualmente Funciona.
4. Se utiliza el AvrStuido 4 para el desarrollo y para grabar el AvrButterfly, sin embargo este mismo Proceso de Grabado por medio de RS232, no Funciona en la Versión 5 del AvrStudio, esto se debe a que le falta un complemento el AvrProg, este se baja de internet y debe ser incluido en las Herramientas. Con esto se consigue trabajar bien en el AvrStudio 5.

[7] Fundación Wikimedia, Inc, Intel 8051, http://es.wikipedia.org/wiki/Intel_8051 , fecha de consulta diciembre 2011

[8] Frino, Luis, ASPECTOS BASICOS DEL PROGRAMA PROTEUS, <http://www.frino.com.ar/teus.htm> , fecha de consulta diciembre 2011

[9] Landen, AVR Butterfly Quick Start User Guide, http://elmicro.com/files/atmel/atavrbfly_quickstart.pdf , fecha de consulta diciembre 2011

[10] Doxygen, Global manipulation of the interrupt flag, http://www.nongnu.org/avr-libc/user-manual/group__avr__interrupts.html , fecha de consulta diciembre 2011

[11] Mikroontroller, ATMEGA128: AD-Wandlung - Interrupt, <http://www.mikrocontroller.net/topic/38318> , fecha de consulta diciembre 2011

[12] Pudn, Downloads Codes Google Pudn, http://search.pudn.com/search_read_en.asp?keyword=AVR+adc , consulta diciembre 2011

[13] Pudn, ADC.c, http://read.pudn.com/downloads90/sourcecode/embed/343367/task/task1/ADC.c_.htm , fecha de consulta diciembre 2011

[14] DISQUS, ADC, <http://jmlab.com/adc/adc.html> , fecha de consulta diciembre 2011

[15] Pardue, Joe, Butterfly Alternate Pin Uses, http://www.smileymicros.com/download/Butterfly%20Alternate%20Pin%20Uses.pdf?&MMN_position=62:62 , fecha de consulta diciembre 2011

[16] AVR Freaks, Temperature sensor, <http://www.avrfreaks.net/index.php?name=PNphpBB2&file=viewtopic&p=819824> , fecha de consulta diciembre 2011

VIII. REFERENCIAS

[1] Quiceno Manrique, Andrés Felipe, Estructura general de un programa en C para sistemas embebidos, <http://trucoselectronicayprogramacion.blogspot.com/2010/11/estructura-general-de-un-programa-en-c.html>, Fecha de consulta noviembre 2011.

[2] AVR Tutorials, AVR Microcontroller Stack and Stack Pointer, <http://www.avr-tutorials.com/general/avr-microcontroller-stack-operation-and-stack-pointer>, fecha de consulta noviembre 2011.

[3] Academic, Gama de color, <http://www.esacademic.com/dic.nsf/eswiki/516331> , fecha de consulta noviembre 2011

[4] DISQUS, Interrupciones, <http://jmlab.com/interrupciones/interrupciones.html>, fecha de consulta noviembre del 2011

[5] Escuela Superior Politécnica de ALCOY, COMPARATIVA DE MICROCONTROLADORES ACTUALES, <http://server-die.alc.upv.es/assignaturas/LSED/2002-03/Micros/downloads/trabajo.pdf>, fecha de consulta noviembre del 2011

[6] Domínguez Arellano, Rufino, Interrupciones, <http://ael.110mb.com/informatica/Int8051.pdf> , fecha de consulta diciembre del 2011.