

Sistemas Temporizados implementados con microcontroladores Atmel, construcción de Plataforma básica para explicar el uso detallado del temporizador TIMER1

Rommel Chang Suarez ⁽¹⁾, Jefferson Moreno Briones ⁽²⁾, Carlos Valdivieso ⁽³⁾
Facultad de Ingeniería en Electricidad y Computación ⁽¹⁾ ⁽²⁾ ⁽³⁾
Escuela Superior Politécnica del Litoral - ESPOL ⁽¹⁾ ⁽²⁾ ⁽³⁾
Km 30.5 Vía Perimetral, Guayaquil, Ecuador ⁽¹⁾ ⁽²⁾ ⁽³⁾
romfchan@espol.edu.ec ⁽¹⁾, jeffmore@espol.edu.ec ⁽²⁾, cvaldiv@espol.edu.ec ⁽³⁾

Resumen

En el presente documento se exponen los principios básicos, el diseño y la implementación del TIMER/COUNTER1 del microcontrolador atmega169 de Atmel. La unidad del TIMER/COUNTER1 de 16 bits permite la correcta temporización (sincronización) para la ejecución del programa (administración de evento), generación de onda y medición temporizada de señales. La finalidad del presente proyecto es el desarrollo de una plataforma interactiva para la demostración del TIMER/COUNTER1, utilizado con los microcontroladores de Atmel, para ello se utilizarán varias herramientas de Software tales como el AVR Studio que es un entorno de desarrollo integrado (IDE) de Atmel para el desarrollo de aplicaciones basadas en microcontroladores AVR de 8-bits y un simulador de circuitos electrónicos como lo es el Proteus y para la implementación física se utilizará el Kit de desarrollo AVR butterfly.

Abstract

This paper presents the basic principles, design and implementation of the Timer/Counter1 from atmega169 microcontroller of Atmel. The 16-bit Timer/Counter1 unit allows the correct timing (synchronization) for program execution (event management), waveform generation and signal measurement. The purpose of this project is to develop an interactive platform for Timer/Counter1 implementations, used with Atmel microcontrollers. A software tool such as AVR Studio, which is an integrated development environment (IDE) by Atmel for developing applications based on 8-bit [AVR microcontroller](#) and an electronic circuit simulator such as the Proteus and the physical implementation will use with the AVR butterfly development Kit.

1. Introducción

El Timer 1 es un módulo temporizador/contador de 16 bits, que consiste en dos registros de 8 bits (TMR1H y TMR1L) que son de lectura y escritura. Este módulo incrementa su cuenta desde

0x000 hasta 0xFFFF y al desbordarse vuelve a 0x0000. Al presentarse el desborde la bandera de interrupción TMR1IF se pone a 1 y, si está habilitada, la interrupción se presenta.

Este módulo al igual que el [Timer 0](#) puede funcionar en modo temporizador y en modo contador. En modo temporizador el par de registros TMR1 se incrementa en cada ciclo de instrucción, este modo se selecciona poniendo a 0 el bit TMR1CS del registro T1CON. En modo contador el par de registros TMR1 se incrementa en cada flanco ascendente de una señal de reloj

externa, este modo se selecciona poniendo a 1 el bit TMR1CS del registro T1CON.

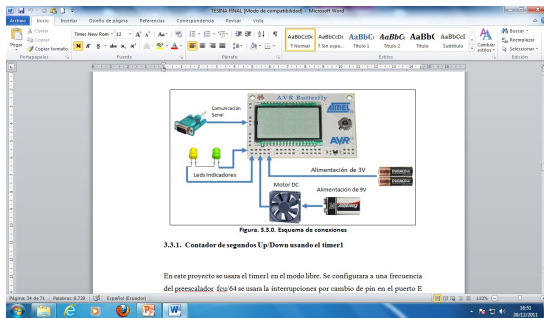
2. Características del Timer/counter1

- Diseño verdadero de 16 bit (p.ej. permite PWM de 16 bit).
- Dos Unidades Comparadoras independientes.
- Registros Comparadores de doble buffer.
- Una Unidad para Captura de Datos (Input Capture).
- Anulador de Ruido para la Captura de Datos.
- Timer Encerado por Coincidencia en Comparación (Auto Recarga).
- Fase Correcta del Modulador de Ancho de Pulso (PWM), libre de fallos.
- PWM de Periodo Variable.
- Generador de Frecuencia.
- Contador de Evento Externo.
- Cuatro Fuentes de Interrupción Independientes (TOV1, OCF1A, OCF1B e ICF1).

3. Diseño del Hardware

El kit AVR Butterfly se diseñó para demostrar los beneficios y las características más importantes de los microcontroladores.

El AVR Butterfly utiliza el microcontrolador AVR ATmega169V, que combina la Tecnología Flash con el más avanzado y versátil microcontrolador de 8 bits disponible. [3]



Los siguientes recursos están disponibles en el Kit AVR Butterfly: Microcontrolador 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.
- Acceso externo al canal 1 del ADC del ATmega169, para lectura de voltaje en el rango de 0 a 5 V.
- Interfaz USI, para una interfaz adicional de comunicación.

El AVR Butterfly está proyectado para el desarrollo de aplicaciones con el ATmega169 y además puede usarse como un módulo dentro de otros productos. [7]

4. Software



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.

AVR Studio apoya al diseñador en el diseño, desarrollo, depuración y parte de la comprobación del proceso. AVR Studio es actualizado continuamente y está disponible para descargarlo desde www.atmel.com

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. [6]

5. Descripción de proyectos.

En este ítem vamos a describir los proyectos realizados con el Kit Avr Butterfly con respecto al tema del módulo TIMER/COUNTER1 y los diferentes modos de uso del TIMER1.

Además se mostrara la simulación en Proteus de las salidas más importantes del proyecto.

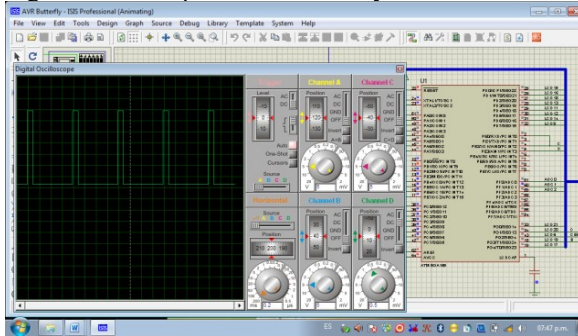
5.1 Contador de segundos Up/Down

En este proyecto se usara el timer1 en el modo libre. Se configurara a una frecuencia del preescalador $f_{cu}/64$ se usara la interrupciones por cambio de pin en el puertoE donde esta conectados los botones del joystick derecha e izquierda para indicar si cuenta ascendente o descendente.

Se da uso de la LCD para mostrar el conteo de los segundos y se muestra la rutina para inicializar la pantalla LCD incorporada en el Kit Avr Butterfly El programa inicia y necesita de los botones derecha del joystick para contador up y el botón izquierda para el contador down. [1]

5.2 Uso del timer1 modo CTC

En el siguiente proyecto se estudiara el timer 1 en el modo de comparación doble usando los registros de comparación OCR1A y OCR1B.



Cuando los registros del timer 1 lleguen a la comparación B entonces apagamos un led en el puerto C y en la comparación del registro A lo encendemos usaremos unos tiempos de comparación para producir las siguiente onda con un clock de $f_{cu}/1024$ [5].

5.3 Reloj a diferentes frecuencias Este proyecto se desarrolló para crear una fuente de reloj basado en el timer/counter1 en modo libre (ciclo de trabajo 50%). Usaremos la interrupción por sobrepaso del timer1 para generar nuestra onda cuadrada.

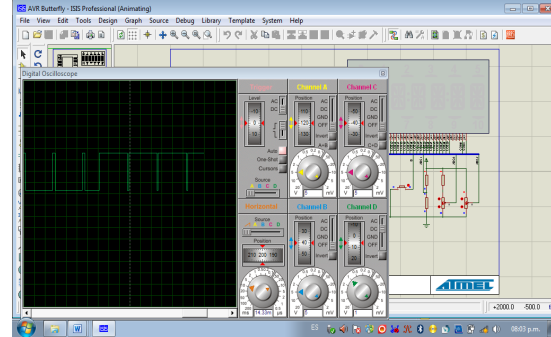
Se utiliza la interrupción por cambio de pin para cambiar la frecuencia del reloj para esto usamos los botones del joystick y la señal de salida la activamos en el pin 6 del puertoC .

Seteamos el valor del registro TCNT1 para empezar en el momento justo para el periodo de reloj. [5]

5.4 Control PWM de intensidad de luz

Las señales de frecuencia y de ciclo de trabajo

variados tienen una amplia gama de aplicaciones en automatización. Un ejemplo típico es un circuito de control de potencia. Refiérase a la siguiente figura. Si un cero lógico (0) indica un interruptor abierto y un uno lógico (1) indica un interruptor cerrado, la potencia eléctrica que se transmite a los consumidores será directamente proporcional a la duración del pulso. Esta relación se le denomina Ciclo de Trabajo.



En este proyecto aprendemos a configurar el timer1 en modo pwm y dependiendo del registro de control OCR1A variamos el valor promedio de la salida. Un cambio de frecuencia se lo realiza por medio de la interrupción por cambio de pin y el otro cambio por la variación del pin 2 del puerto E y se muestra en la pantalla LCD el porcentaje de la salida. [2]

6. Conclusiones

-Conocer y entender un nuevo lenguaje de programación fue uno de los puntos más importantes de este proyecto. De gran aporte para la vida profesional ya que nos da la pauta para ingresar en mas familias de microcontroladores.

-Mediante el desarrollo de nuestro proyecto pudimos destacar una de las principales características del Timer1 que actúa temporizador/contador ascendente parecido al TMR0, pero con algunas peculiaridades que lo hacen muy interesante a la hora de incluir temporizaciones en nuestros programas. La primera de ellas, es que se trata de un contador de 16 bits cuyo valor se almacena en dos registros de 8 bits, en ambos registros se pueden leer y escribir su valor durante la ejecución del programa.

-Cuando el Timer1 está habilitado, el valor de esos registros se incrementan desde 0000h a FFFFh y una vez que llega a su máximo valor empieza otra vez desde 0 avisándonos de ello por medio de una



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERIA ELECTRICA Y COMPUTACION



bandera.

Si está activa la interrupción por desbordamiento del Timer 1 al desbordarse el contador, el programa entra en la función de tratamiento a la interrupción por desbordamiento del Timer1.

7. Recomendaciones

-No conectar cables directamente en los espacios para conexiones externas del Kit, ya que podrían causar cortocircuito; en su lugar, colocar Headers fijos, en ocasiones hay que añadir una fuente externa de 3V para poder visualizar los datos en la LCD cuando se gasta la batería interna del KI AVR Butterfly.

- Al momento de codificar software en lenguaje C, es recomendable segmentar el código fuente en funciones especializadas, esto quiere decir que cada función realice una sola tarea específica; de este modo se podrán utilizar las mismas funciones en otras aplicaciones.

- Es preciso y necesario recomendar el uso del Kit AVR Butterfly, simultáneamente con la Guía de Prácticas de Laboratorio, en la cátedra de Microcontroladores.

8. Referencias

- [1] PARDUE, Joe, C Programming for Microcontrollers, tomo 1, 1ra Edición, Editorial Smiley Micros, Knoxville-Tennessee Octubre del 2005.
- [2] MANN, Richard, How to Program an 8-bit Microcontroller Using C Language, disponible en: www.atmel.com, 23 de junio del 2004.
- [3] RODLAND, Arild, Novice's Guide to AVR Development, disponible en: www.atmel.com, 2 de febrero del 2004.
- [4] AVRProg User Guide, disponible en: www.atmel.com. Fecha de consulta 11/03/2011
- [5] 8-bit AVR Microcontroller with 16K Bytes In-System Programmable Flash ATmega169V Atmega169 Rev A to E, disponible en: www.atmel.com, Fecha de consulta 10/23/2011.
- [6] Introduction to the Atmel AVR Butterfly, disponible en: www.atmel.com, Fecha de consulta 11/11/201.
- [7] AVR Butterfly Evaluation Kit User Guide, disponible en: www.atmel.com/products/AVR/butterfly, Fecha de consulta 09/02/201.