

TARJETA BASADA EN EL MICROCONTROLADOR ATMEL AVR ATMEGA-323

M. BARRÓN RUIZ¹, J. MARTÍNEZ PÉREZ²

¹Dpto de Ingeniería de Sistemas y Automática. Avenida de Otaola 29, EUITI de Eibar, UPV-EHU. 20600-Eibar. España. Email: ispbarum@sb.ehu.es

²Dpto de Electrónica. Instituto de Enseñanza Secundaria Inventor Cosme García. Logroño. España. Email: jdisen@openbank.es

Creemos que la familia de microcontroladores (MCU) AVR de ATMEL, aparecida en 1996, ofrece ventajas sobre las familias clásicas: '68, '51, Z8, PIC,... Por este motivo nos hemos propuesto el desarrollo de una tarjeta basada en AVR, que permita a nuestros estudiantes explorar las características de estos MCUs y que les facilite la realización de sus Proyectos Fin de Carrera. Esperamos que las siguientes líneas sirvan de guía a los docentes que se encuentran evaluando las distintas opciones disponibles con el objetivo de ofrecer lo mejor a sus alumnos.

1. Introducción

Los siguientes apartados describen una tarjeta destinada a la enseñanza de los microcontroladores (MCUs) basada en un ATMEL ATmega323 con capacidad para autoprogramar su memoria FLASH y EEPROM utilizando el puerto serie de un PC.

2. Características de la familia de microcontroladores AVR

Generalmente las máquinas RISC requieren mayor tamaño de código que las máquinas CISC para realizar una misma tarea. Para salvar este inconveniente los AVR han sido dotados de una arquitectura RISC mejorada [1, 3] que consiste en un MCU RISC que ejecuta una instrucción por cada pulso de reloj, al que se le han añadido un mayor número de instrucciones para reducir el tamaño de código e incrementar aún más su velocidad. Las instrucciones añadidas no penalizan ni las prestaciones ni el consumo propio de los RISC.

Los lenguajes de alto nivel se han convertido en el método de programación estándar para los MCUs debido a la disminución del tiempo de desarrollo y a la simplificación del mantenimiento de los programas. El lenguaje C es probablemente el lenguaje de alto nivel más utilizado con los MCUs. Los lenguajes de alto nivel han tenido tradicionalmente una desventaja con respecto al lenguaje ensamblador: el aumento del tamaño del código. La cooperación entre el equipo de diseño de los AVR y el equipo de diseño de su compilador C [2] ha producido una arquitectura muy bien adaptada al empleo de lenguajes de alto nivel, que facilita el diseño de compiladores C muy eficientes.

Además de los dos aspectos anteriores, las características más importantes de los AVR son:

- Arquitectura Harvard (espacios separados para memoria de código y de datos).
- Ancho de palabra de código de 16 bits. Ancho de palabra de datos de 8 bits.
- Ejecución de instrucciones en un solo pulso de reloj.
- Amplia familia con más de 20 miembros y encapsulados desde 8 hasta 64 patillas.
- Las patillas de I/O admiten corrientes de sumidero de hasta 20 mA.
- Todos los AVR, salvo los nuevos FPSLIC, almacenan el programa en FLASH.
- Memoria FLASH desde 1 hasta 128 KBytes (la arquitectura admite hasta 8 MB).
- Todos los AVR disponen internamente de 32 registros (acumuladores) de 8-bits.
- Todos los AVR, salvo ATtiny11 y ATtiny28, poseen EEPROM (desde 64 B hasta 4 KB).
- La memoria RAM varía desde 0 en los más pequeños, hasta 4 KB.
- Todos los AVR tienen lógica interna de RESET. No necesitan componentes externos.
- Todos los AVR disponen de temporizador *Watchdog*.
- Todos los AVR se pueden programar y/o borrar mediante *interface* serie ISP.
- Todos los AVR de más de 20 patillas permiten programación paralela.
- Oscilador interno que hace innecesario un cristal externo y libera dos pines.
- Arquitectura optimizada para el uso de lenguajes de alto-nivel.
- Fácil programación en ensamblador gracias a su acertado conjunto de instrucciones.
- Compilador C (GNU), Ensamblador y simulador (ATMEL) gratuitos.
- Compiladores C (Iar, Codevision, ImageCraft, ...), Pascal (AVRco) y Basic (Bascom).

3. Características del microcontrolador ATmega323

A la hora de elegir un MCU destinado a la enseñanza parece lógico optar por alguno que disponga de la gama más completa de periféricos, del mayor número de formas de programación y que ofrezca facilidades para la depuración del software. El dispositivo ATmega323 elegido para nuestra tarjeta de prácticas (también acepta el ATmega163) además de las características generales de la familia AVR descritas en el apartado anterior, posee numerosas características propias lo que permite la experimentación con todos los periféricos actuales de esta familia. Los siguientes datos proporcionan una rápida descripción del chip:

- Arquitectura RISC avanzada [3] con 130 instrucciones.
- Encapsulado PDIP de 40 patillas o TQFP de 44 patillas.
- Frecuencia de operación entre 0 y 8 MHz (entre 0 y 4 MHz para el ATmega323L).
- Tensión de trabajo entre 4,0 y 5,5 V. (2,7 y 5,5V si ATmega323L).
- Oscilador interno RC calibrado a 1 MHz.
- 32 KB de memoria FLASH, 2 KB de SRAM y 1 KB de EEPROM.
- 32 líneas de I/O programables individualmente.
- Multiplicación hardware en 2 pulsos de reloj.

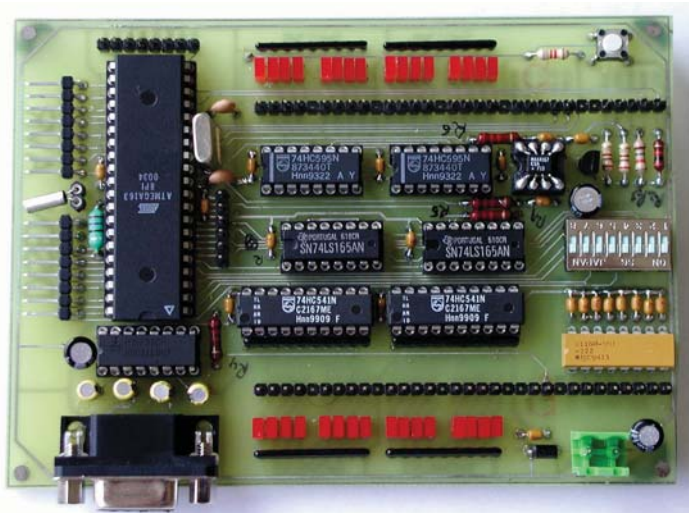


Figura 1: Tarjeta ATmega323.

- Programación paralela, serie ISP, JTAG y autoprogramación.
- Soporte para depuración mediante JTAG.
- Dos *Timers/Counters* de 8 bits con sendos *prescaler* y modo comparación.
- *Timer/counter* de 16 bits con su *prescaler* y modos captura y comparación.
- Contador de tiempo real con oscilador separado optimizado para 32.768 Hz.
- Convertidor A/D de ocho canales y 10 bits de resolución.
- Cuatro canales PWM..
- Comparador analógico.
- *Interfaces* serie: USART, I²C y SPI.
- *Watchdog* programable con su propio oscilador.
- *Power-on Reset* y detector programable de *Brown-out*.
- 19 fuentes de interrupción, 3 externas y 16 de los periféricos internos.
- Seis modos de trabajo para consumo reducido de potencia.
- Ensamblador, simulador y depurador JTAG suministrado por ATMEL sin costo alguno.

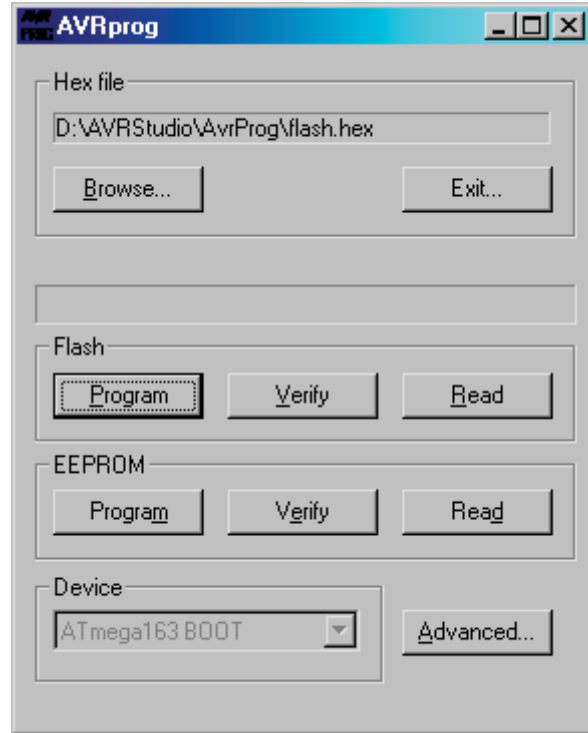


Figura 2: Programa AVRProg.

4. Hardware de la tarjeta controladora ATmega323

Se trata de una tarjeta, véase figura 1, que contiene nueve chips: el microcontrolador ATmega323; un MAX-232 para la conversión entre niveles TTL \leftrightarrow RS-232; el doble AOI MAX-4167, con entradas y salidas *rail to rail*, con salidas de alta capacidad de conducción de corriente para amplificar dos señales analógicas; dos registros de desplazamiento SIPO de 8 bits 74HC595 para obtener 16 salidas digitales monitorizadas por 16 LEDs; dos *octal buffer driver* 74HC541 para monitorizar 16 entradas digitales mediante LEDs, y dos registros de desplazamiento PISO de 8 bits 74HC165 para leer las 16 entradas digitales.

La tarjeta incluye dos cristales de cuarzo de 7,3728 MHz y 32,768 KHz. El primer cristal lo utiliza el MCU, y el segundo, junto con el *timer* T2, permite realizar un reloj en tiempo real. El MCU lee las 16 entradas y escribe en las 16 salidas utilizando el bus serie SPI (*Serial Peripheral Interface*). Se ha procurado dejar libre el mayor número de hilos de IO del ATmega323 y proporcionarle entradas y salidas adicionales monitorizadas con LEDs para facilitar la realización de prácticas y aumentar las posibilidades de conexión con el exterior.

5. Software de la tarjeta controladora ATmega323

El ATmega323 es un dispositivo de 16 Kword (32 KB) de memoria FLASH dividida en dos secciones: la sección de aplicación y la sección de autoprogramación (*Boot Loader section*). El tamaño de la sección de autoprogramación puede configurarse mediante fusibles a 256, 512, 1024 o 2048 *words*. El MCU dispone de una instrucción SPM (*Store Program Memory*)

para escribir datos en la memoria FLASH. La instrucción SPM solo puede ejecutarse desde la sección de autoprogramación. El usuario puede escribir un programa que permita la re-escritura del área de aplicación de la memoria FLASH del MCU y ubicarlo en la sección de autoprogramación. Este programa llamado *Boot-Loader* debe encargarse además de la gestión de las comunicaciones con un PC. El programa *Boot-Loader* puede comunicarse con el PC utilizando dos hilos cualesquiera, o bien su bus SPI, o su conector JTAG o la USART.

En la tarjeta descrita se ha utilizado la USART para comunicarse con el PC, y se ha creado un programa *Boot-Loader* de 240 words que permite la lectura, borrado y escritura de la memoria FLASH y EEPROM de la MCU, y que utiliza las dos últimas posiciones de la EEPROM para guardar el número de borrados de la memoria FLASH, ya que ATMEL sólo garantiza 1000 ciclos de borrado y escritura para la misma. De esta forma se puede tener un control sobre el uso de la tarjeta y decidir el momento en que debe cambiarse el MCU, o destinarlo a otro tipo de aplicación que no utilice la característica de reprogramación.

En el PC se necesita un programa capaz de dialogar con el *Boot-Loader* para leer el contenido de la memoria de la tarjeta ATmega323 o para proporcionar el código y/o los datos con los que reprogramar las memorias FLASH y/o EEPROM. Para nuestra tarjeta se3 ha utilizado como *interface* de usuario el programa para Windows **AVRProg** [4], disponible de forma gratuita en www.atmel.com, véase figura 2. Este programa puede obtenerse de forma individual o integrado dentro del software **AVR Studio** el cual incluye ensamblador, simulador y depurador de código fuente.

Al conectar la alimentación a la tarjeta ATmega323, o después de un RESET, el control del programa se transfiere al *Boot-Loader*. Este programa examina el estado de la patilla SCL (PC.0). Si esta patilla se encuentra a nivel alto cede el control al programa del usuario residente en la sección de aplicación, pero si se encuentra a nivel bajo dialoga con el PC para permitir la lectura programación y verificación del contenido de la memoria FLASH y EEPROM del MCU. La programación de la memoria FLASH comienza siempre con el borrado de la sección de aplicación pero manteniendo inalterado el contenido de la sección de autoprogramación, es decir el propio *Boot-Loader*.

6. Conclusiones

Se ha presentado una tarjeta basada en un moderno MCU y se han descrito sus características y las herramientas necesarias para el desarrollo de aplicaciones. La sencillez de la tarjeta, el costo cero para el software de desarrollo y para el equipo de programación, así como la riqueza de periféricos y a la potencia del ATmega-323 son algunos de sus aspectos más interesantes.

Referencias

- [1] Alf-Egil Bogen, Vegard Wollan, "AVR Enhanced RISC Microcontrollers", ATMEL Corp.
- [2] Gaute Myklebust, "The AVR Microcontroller and C Compiler Co-Design", 3rd European Microprocessor and Microcontroller Seminar, Heathrow, UK, November 6, 1996, 164-170.
- [3] "8-bit AVR Microcontroller with 32K Bytes of In-System Programmable Flash , ATmega323 ATmega323L" Data Sheet, Rev. 1457C-07/2001, ATMEL Corporation.
- [4] "AVRProg User Guide" ", Rev. 1021A-A-01/1998, ATMEL Corporation.

CURSO PRÁCTICO DE ELECTRÓNICA CON EL EQUIPO ELECTRONICS-LAB

M. BARRÓN¹, J. MARTÍNEZ², J. L. ALONSO¹, J. M. MELEIRO¹,

¹Dpto de Ingeniería de Sistemas y Automática. Avenida de Otaola 29, EUITI de Eibar, UPV-EHU. 20600-Eibar. España. Email: ispbarum@sb.ehu.es

²Dpto de Electrónica. Instituto de Enseñanza Secundaria Inventor Cosme García. Logroño. España. Email: jdisen@openbank.es

Se presenta aquí un Curso Práctico de Electrónica que cubre el estudio de gran parte de los circuitos analógicos y digitales más comúnmente utilizados. El curso exige el montaje real de los circuitos en las superficies disponibles a tal efecto en el equipo Electronics-Lab. Este equipo se presentó en TAAE 2000 [1] donde fue galardonado con el “Premio al Mejor Equipo Hardware”; unos días antes había conseguido un segundo premio en el Design2K Contest organizado por la revista americana Circuit Cellar y la empresa PHILIPS. Esperamos que el material elaborado sea de utilidad para todos aquellos que piensan que en el proceso de enseñanza-aprendizaje de una materia como la Electrónica debe fomentarse el trabajo directo con los componentes electrónicos evitando el actual abuso de la simulación.

1. Introducción

Electronics-Lab, véase figura 1, es un equipo electrónico basado en un microcontrolador (MCU) que se conecta al puerto serie de un PC para aprovechar las mejores características de los MCUs y de los PCs. Puede encontrarse una descripción detallada del hardware y software del Electronics-Lab en la referencia [2]. Los MCUs son dispositivos ricos en entradas y salidas de naturaleza digital y analógica y están bien dotados para relacionarse con el mundo físico, mientras que los PCs con sus enormes capacidades de cálculo y de visualización gráfica, pueden dibujar en tiempo real las señales capturadas por las entradas del MCU y presentar el interface de usuario, véase figura 2, para que éste pueda actuar sobre las salidas del MCU. En definitiva, la conjunción del equipo Electronics-Lab con el PC, junto al software que corre en ambos equipos proporciona casi toda la instrumentación necesaria en las prácticas de Electrónica como puede verse a continuación:

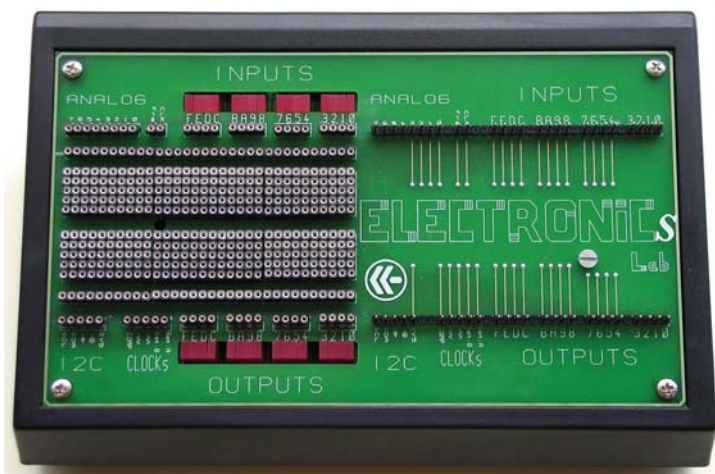


Figura 1: Equipo Electronics-Lab.