TECNICATURA SUPERIOR EN TELECOMUNICACIONES

ELECTRÓNICA MICROCONTROLADA

Docentes: Ing. Jorge E. Morales, Téc. Gonzalo Vera.

**Título:** Estructura del Microcontrolador AVR® de 8 bits

**Grupo 1:**

❖ Birge, Adolfo Federico.

❖ Carunchio, Carlos Javier.

❖ Ferreyra, María Luciana.

❖ Gutiérrez, Emma Vilma.

❖ Merlo, Emmanuel.

❖ Romero, Gisela de Lourdes.

Estructura del Microcontrolador AVR® de 8 bits

Núcleo de CPU AVR

Tiempo de instrucción AVR

Unidad lógica aritmética AVR

Registros de propósito general AVR

Registro de pila AVR

Registro de estado de AVR

Memoria AVR

Optimización del código C en AVR

Memoria de lectura mientras se escribe AVR (gestor de arranque)

Puertos de E/S digitales AVR

Modos de suspensión de bajo consumo AVR

Registro de reducción de potencia periférico AVR

Fusibles AVR

Interfaces de programación AVR

Temporizador AVR Watchdog

AVR USART Introducción

Temporizador AVR Comparar registro doble búfer

Modos de funcionamiento de AVR ADC

Modo diferencial AVR ADC

Modo de reducción de ruido AVR ADC

Voltaje de referencia AVR ADC

Interfaz periférica serie AVR (SPI)

Controlador táctil periférico AVR (PTC)

Comparador interno AVR

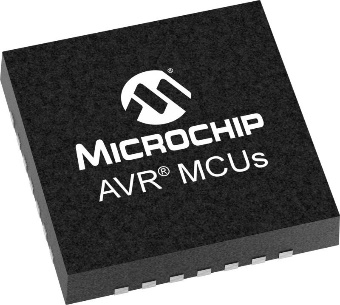
Detección de apagado AVR

Interrupciones AVR

Interrupciones externas AVR

Contador en tiempo real (RTC) de AVR

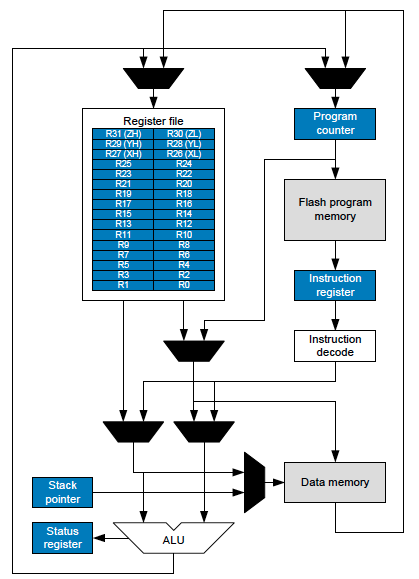
**Estructura del Microcontrolador AVR® de 8 bits**

**Los microcontroladores (MCU) AVR de 8 bits ofrecen una familia de dispositivos únicos y versátiles. Con la gran cantidad de periféricos incorporados, siempre es útil tener capacitación sobre cómo funcionan, cómo configurarlos y ejemplos de cómo usarlos. A continuación, detallaremos su arquitectura:**

**Núcleo de CPU AVR**

La función principal del núcleo de la Unidad Central de Procesamiento (CPU) AVR ® es garantizar la correcta ejecución del programa. Por lo tanto, la CPU debe poder acceder a las memorias, realizar cálculos, controlar periféricos y manejar interrupciones.

Diagrama de bloques de la arquitectura AVR



**Núcleo AVR**

Para maximizar el rendimiento y el paralelismo, el AVR utiliza una arquitectura Harvard con memorias y buses separados para programas y datos. Las instrucciones en la memoria del programa se ejecutan con canalización de un solo nivel. Mientras se ejecuta una instrucción, la siguiente instrucción se obtiene previamente de la memoria del programa. Este concepto permite ejecutar instrucciones en cada ciclo de reloj. La memoria del programa es una memoria Flash reprogramable en el sistema.

**Registros**

El archivo de registro de acceso rápido contiene 32 registros de trabajo de propósito general de 8 bits con un solo tiempo de acceso de ciclo de reloj. Seis de los 32 registros se pueden utilizar como tres punteros de registro de direcciones indirectas de 16 bits para el direccionamiento del espacio de datos, lo que permite cálculos de direcciones eficientes. Uno de estos punteros de dirección también se puede utilizar como puntero de dirección para tablas de búsqueda en la memoria de programa Flash. Estos registros de funciones adicionales son los registros X, Y y Z de 16 bits.

**Unidad Aritmética Lógica (ALU)**

La ALU admite operaciones aritméticas y lógicas entre registros o entre una constante y un registro. El tiempo de acceso de ciclo de reloj único permite operaciones de ALU de ciclo único. En una operación ALU típica, se emiten dos operandos desde el archivo de registro, se ejecuta la operación y el resultado se almacena nuevamente en el archivo de registro en un ciclo de reloj. Las operaciones de registro único también se pueden ejecutar en la ALU. Después de una operación aritmética, el registro de estado se actualiza para reflejar información sobre el resultado de la operación. El flujo del programa lo proporcionan las instrucciones de salto y llamada condicionales e incondicionales, capaces de abordar directamente todo el espacio de direcciones. La mayoría de las instrucciones AVR tienen un solo formato de palabra de 16 bits. Cada dirección de memoria de programa contiene una instrucción de 16 o 32 bits.

**Memoria**

Los espacios de memoria en la arquitectura AVR son todos mapas de memoria lineales y regulares. El espacio de la memoria flash del programa se divide en dos secciones, la sección del programa de arranque y la sección del programa de aplicación. Ambas secciones tienen bits de bloqueo dedicados para protección contra escritura y lectura/escritura. La instrucción Store Program Memory (SPM) que escribe en la sección de memoria Flash de la aplicación debe residir en la sección Boot Program.

Durante las interrupciones y las llamadas a subrutinas, el Contador de programa (PC) de la dirección de retorno se almacena en la pila. La pila se asigna efectivamente en la SRAM de datos generales y, en consecuencia, el tamaño de la pila solo está limitado por el tamaño total de la SRAM y el uso de la SRAM.

Todos los programas de usuario deben inicializar el puntero de pila (SP) en la rutina de reinicio (antes de que se ejecuten las subrutinas o interrupciones). El SP es accesible para lectura/escritura en el espacio de E/S. Se puede acceder fácilmente a la SRAM de datos a través de los cinco modos de direccionamiento diferentes admitidos en la arquitectura AVR.

El espacio de memoria de E/S contiene 64 direcciones para funciones periféricas de la CPU como registros de control, interfaz periférica en serie (SPI) y otras funciones de E/S. Se puede acceder a la memoria de E/S directamente o como ubicaciones de espacio de datos que siguen a las del archivo de registro, 0x20 - 0x5F. Además, este dispositivo ha ampliado el espacio de E/S de 0x60 - 0xFF en SRAM.

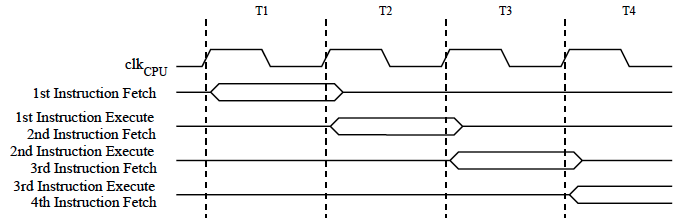
**Interrupciones**

Un módulo de interrupción flexible tiene sus registros de control en el espacio de E/S con un bit de habilitación de interrupción global adicional en el registro de estado. Todas las interrupciones tienen un vector de interrupción separado en la tabla de vectores de interrupción. Las interrupciones tienen prioridad de acuerdo con su posición en el vector de interrupción. Cuanto menor sea la dirección del vector de interrupción, mayor será la prioridad.

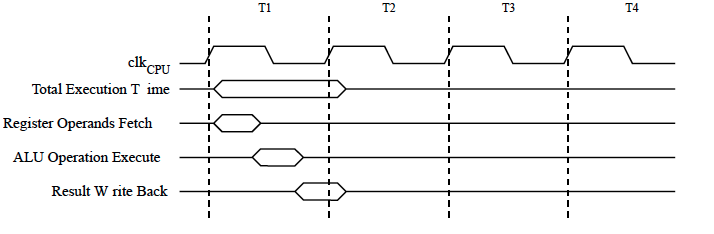
**Tiempo de instrucción AVR**

La unidad central de procesamiento (CPU) AVR es impulsada por el reloj de la CPU, generado directamente desde la fuente de reloj seleccionada para el chip. No se utiliza ninguna división de reloj interna. La arquitectura de Harvard y el concepto de archivo de registro de acceso rápido permiten obtener y ejecutar instrucciones en paralelo. Este es el concepto básico de canalización para obtener hasta 1 MIPS por MHz con los resultados únicos correspondientes para funciones por costo, funciones por relojes y funciones por unidad de potencia.

**Obtención de instrucciones en paralelo y ejecución de instrucciones**

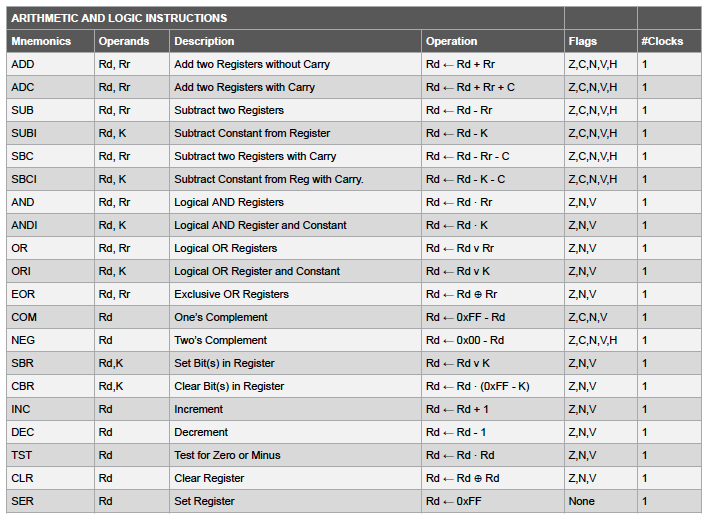
 En un solo ciclo de reloj, se ejecuta una operación de unidad lógica aritmética (ALU) utilizando dos operandos de registro y el resultado se almacena nuevamente en el registro de destino.

**Operación ALU de ciclo único**

****

**Unidad lógica aritmética AVR**

La Unidad Aritmética Lógica (ALU) AVR® de alto rendimiento opera en conexión directa con los 32 registros de trabajo de propósito general. Dentro de un solo ciclo de reloj, se ejecutan operaciones aritméticas entre registros de Propósito General o entre un registro y un inmediato. Las operaciones ALU se dividen en tres categorías principales: funciones aritméticas, lógicas y de bits. Algunas implementaciones de la arquitectura también proporcionan un poderoso multiplicador que admite tanto la multiplicación con signo/sin signo como el formato fraccionario.



**Registros de propósito general AVR**

La estructura del archivo de registro AVR® está optimizada para el conjunto de instrucciones de la computadora del conjunto de instrucciones reducido mejorado (RISC) del AVR. Para lograr el rendimiento y la flexibilidad requeridos, el archivo de registro admite los siguientes esquemas de E/S:

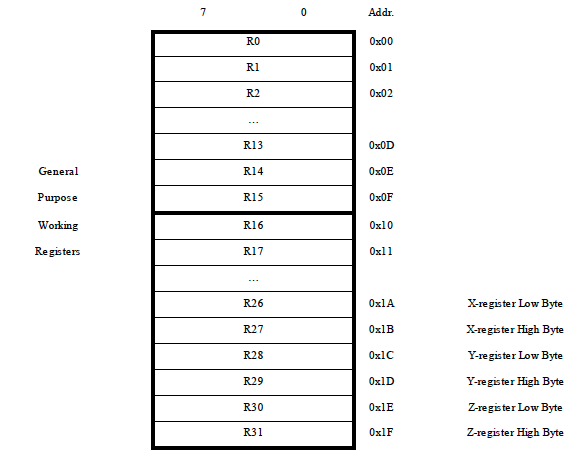
Un operando de salida de 8 bits y una entrada de resultado de 8 bits.

Dos operandos de salida de 8 bits y una entrada de resultado de 8 bits.

Dos operandos de salida de 8 bits y una entrada de resultado de 16 bits.

Un operando de salida de 16 bits y una entrada de resultado de 16 bits.

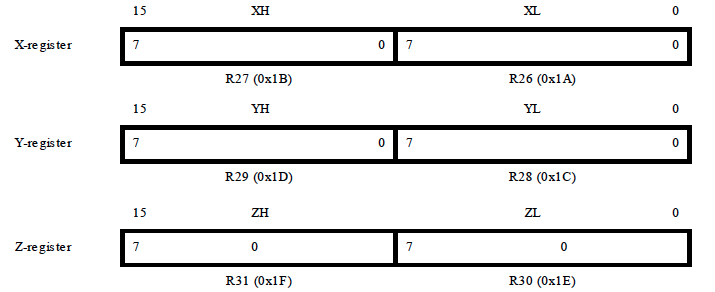
**Conjunto de instrucciones ALU**

Registros de trabajo de propósito general de la CPU AVR

La mayoría de las instrucciones que operan en el archivo de registro tienen acceso directo a todos los registros y la mayoría de ellas son instrucciones de un solo ciclo. A cada registro también se le asigna una dirección de memoria de datos, asignándolos directamente a las primeras 32 ubicaciones del espacio de datos del usuario. Aunque no se implementa físicamente como ubicaciones SRAM, esta organización de memoria proporciona una gran flexibilidad en el acceso a los registros, ya que los registros de puntero X, Y y Z se pueden configurar para indexar cualquier registro en el archivo.

El registro X, el registro Y y el registro Z

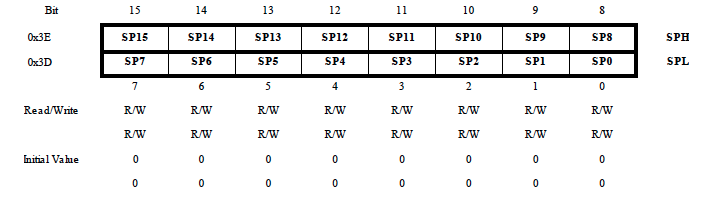
Los registros R26 a R31 tienen algunas funciones adicionales a su uso de propósito general. Estos registros son punteros de dirección de 16 bits para el direccionamiento indirecto del espacio de datos. Los tres registros de direcciones indirectas (X, Y y Z) se definen como se describe en la figura.



**Registro de pila AVR**

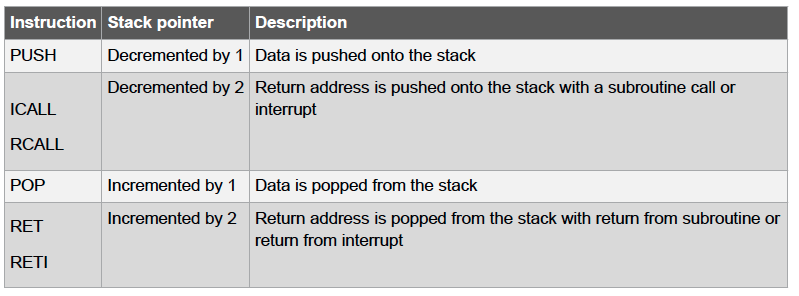
La pila se utiliza principalmente para almacenar datos temporales, variables locales y direcciones de retorno después de interrupciones y llamadas a subrutinas. Se implementa como un crecimiento de ubicaciones de memoria superiores a inferiores. El registro del puntero de pila siempre apunta a la parte superior de la pila; apunta al área de la pila SRAM de datos donde se encuentran las pilas de subrutinas e interrupciones.

Puntero de pila



El AVR Stack Pointer se implementa como dos registros de 8 bits en el espacio de E/S. El número de bits realmente utilizados depende de la implementación.

Conjunto de instrucciones de pila

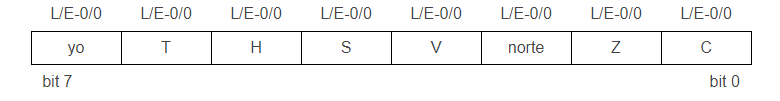


Un comando PUSH de pila disminuirá el puntero de pila. El programa debe definir la pila en la SRAM de datos antes de que se ejecuten las llamadas a subrutinas o se habiliten las interrupciones. El valor inicial del Puntero de pila es igual a la última dirección de la SRAM interna y el Puntero de pila debe configurarse para que apunte por encima del inicio de la SRAM.

**Registro de estado de AVR**

El registro de estado contiene información sobre el resultado de la última instrucción aritmética ejecutada. Esta información se puede utilizar para alterar el flujo del programa con el fin de realizar operaciones condicionales. El registro de estado se actualiza después de todas las operaciones de la unidad lógica aritmética (ALU). En muchos casos, esto eliminará la necesidad de usar las instrucciones de comparación dedicadas, lo que dará como resultado un código más rápido y compacto.

ESTADO: Registro de estado



**bit 7**

I: Habilitación de interrupción global: el bit de habilitación de interrupción global debe establecerse para que se habiliten las interrupciones. El control de habilitación de interrupción individual se realiza entonces en registros de control separados. Si se borra el registro de Habilitación de interrupción global, ninguna de las interrupciones se habilita independientemente de la configuración de habilitación de interrupción individual. El bit I (bit 7) se borra por hardware después de que se ha producido una interrupción y se establece mediante la instrucción RETI (Return from Interrupt) para habilitar las interrupciones subsiguientes. La aplicación también puede establecer y borrar el bit I con las instrucciones Set Global Interrupt Flag (SEI) y Clear Global Interrupt Flag (CLI), como se describe en la referencia del conjunto de instrucciones.

**bit 6**

T: almacenamiento de copia: las instrucciones de copia de bits, carga de bits (BLD) y almacenamiento de bits (BST), utilizan el bit T como fuente o destino para el bit operado. La instrucción BST puede copiar un bit de un registro en el archivo de registro en T y la instrucción BLD puede copiar un bit en T en un bit en un registro en el archivo de registro.

**bit 5**

H: Indicador de medio acarreo: El indicador de medio acarreo, H, indica medio acarreo en algunas operaciones aritméticas. Es útil en la aritmética decimal de código binario (BCD).

**bit 4**

S: Bandera de signo, S = N xor V: El bit S es siempre un exclusivo o entre la Bandera negativa y la Bandera de desbordamiento del complemento a dos.

**bit 3**

V: Indicador de desbordamiento de complemento a dos: El indicador de desbordamiento de complemento a dos, V, es compatible con la aritmética de complemento a dos.

**bit 2**

N: Bandera Negativa: La Bandera Negativa, N, indica un resultado negativo en una operación aritmética o lógica.

**bit 1**

Z: Indicador cero: El indicador cero, Z, indica un resultado cero en una operación aritmética o lógica.

**bit 0**

C: Indicador de acarreo: El indicador de acarreo, C, indica un acarreo en una operación aritmética o lógica.