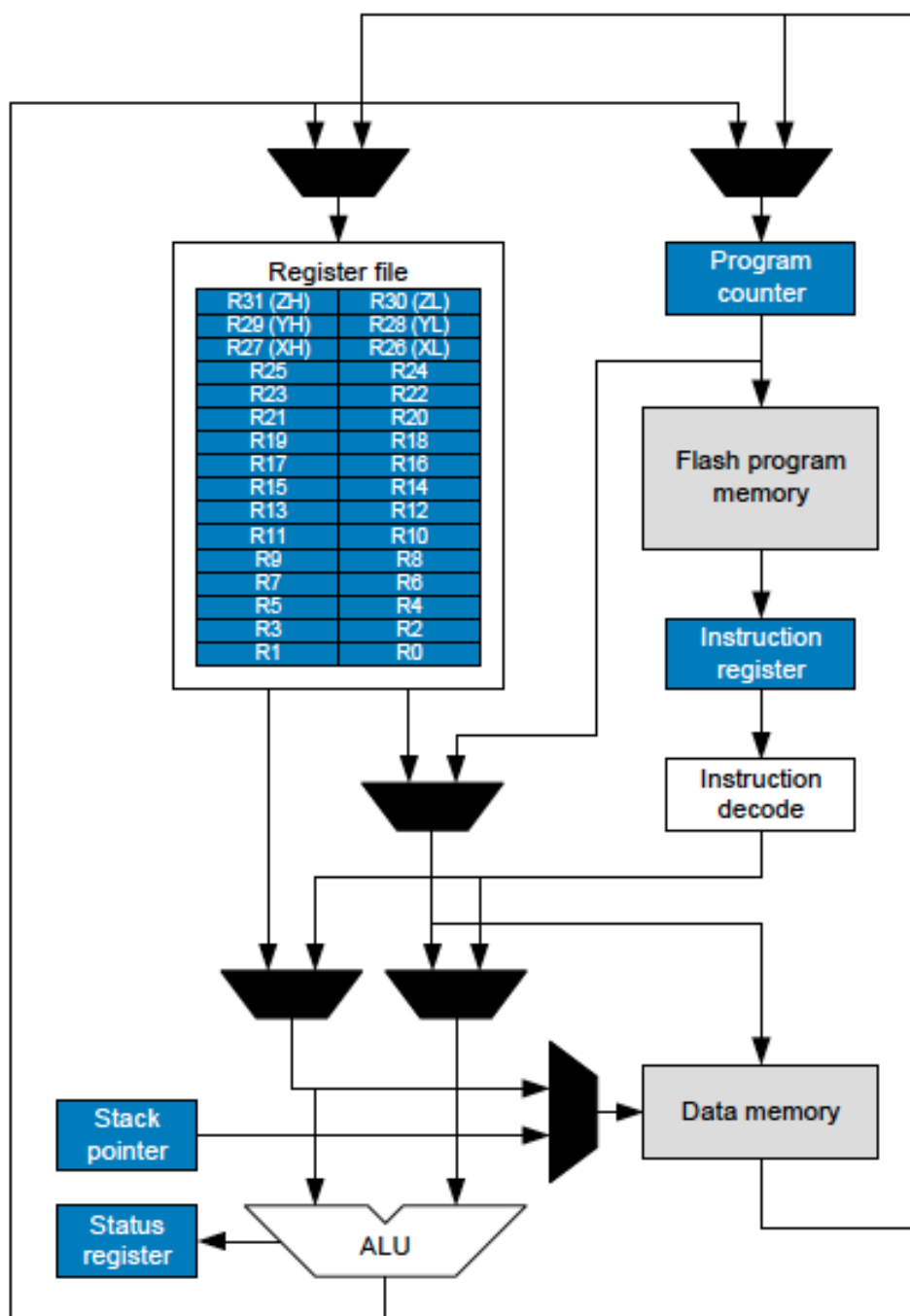


# Núcleo AVR® de 8 bits

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



(/local--files/8avr:avrcore/avrcore.png)

## 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 ( $/8_{avr:instime}$ ). 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 ( $/8_{avr:gpr}$ ) 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 ( $/8_{avr:alu}$ ) 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 (/8avr:stack) . 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 (/8avr:int) 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 (/8avr:status) . 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.