**FPU.h**

En este archivo “.h” define y configura la Unidad de Punto Flotante (FPU) y la Estructura de Control del Sistema (SCB) en un microcontrolador.

¿Qué es el SCB ?

El **System Control** **Block** (SCB) es un componente crucial en los microcontroladores basados en la arquitectura ARM Cortex-M. El SCB proporciona registros para controlar y monitorear varios aspectos del sistema, incluyendo la configuración del sistema, el manejo de interrupciones y la gestión de fallos.

¿Qué es la FPU?

La **Unidad de Punto Flotante (FPU)** es un componente del procesador que se encarga de realizar operaciones aritméticas en números de punto flotante (decimales). Esto incluye operaciones como suma, resta, multiplicación, división y cálculos más complejos como raíces cuadradas y funciones trigonométricas.

En el código se define la estructura para poder interactuar con todos los campos tanto de SCR como de FPU, como se muestra a continuación:

Definición de registros para la configuración del SCB a partir de estructuras:

Texto

Descripción generada automáticamente con confianza baja

Esto es lo que hace cada registro:

1. **CPUID (CPUID Base Register)**: Proporciona información sobre el procesador, como el implementador, la variante, la arquitectura, el número de parte y la revisión.
2. **ICSR (Interrupt Control and State Register)**: Controla y monitorea el estado de las interrupciones.
3. **VTOR (Vector Table Offset Register)**: Define la ubicación de la tabla de vectores de interrupción.
4. **AIRCR (Application Interrupt and Reset Control Register)**: Controla las interrupciones y los reinicios del sistema.
5. **SCR (System Control Register)**: Configura el comportamiento del sistema, como el modo de suspensión.
6. **CCR (Configuration Control Register)**: Configura varias opciones del sistema, como la alineación de datos y la gestión de excepciones.
7. **SHP (System Handlers Priority Registers)**: Define las prioridades de los manejadores del sistema.
8. **SHCSR (System Handler Control and State Register)**: Controla y monitorea el estado de los manejadores del sistema.
9. **CFSR (Configurable Fault Status Register)**: Proporciona información sobre los fallos configurables.
10. **HFSR (HardFault Status Register)**: Proporciona información sobre los fallos graves.
11. **DFSR (Debug Fault Status Register)**: Proporciona información sobre los fallos de depuración.
12. **MMFAR (MemManage Fault Address Register)**: Proporciona la dirección del fallo de gestión de memoria.
13. **BFAR (BusFault Address Register)**: Proporciona la dirección del fallo de bus.
14. **AFSR (Auxiliary Fault Status Register)**: Proporciona información adicional sobre los fallos.
15. **PFR, DFR, ADR, MMFR, ISAR**: Registros que proporcionan información sobre las características del procesador, depuración, características auxiliares, modelo de memoria y atributos del conjunto de instrucciones.
16. **, CPACR:** (Coprocessor Access Control Register) es un registro de 32 bits que controla el acceso a los coprocesadores en un procesador ARM, excepto los coprocesadores CP14 y CP15. Este registro es accesible solo en modos privilegiados y tiene una serie de campos que permiten configurar los derechos de acceso a los coprocesadores.

Despues de tener la definición de cada registro de control para la SBD vienen una parte de código donde se definen varias macros como se muestra en la siguiente imagen:

Texto

Descripción generada automáticamente

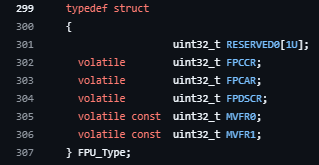
Cada macro se utiliza para indicar la posición para cada configuración, por ejemplo: las primeras dos macros definidas que son la posición y mascara para el campo “Implementer” para el registro CPUID del SCB. Entonces:

1. #define SCB\_CPUID\_IMPLEMENTER\_Pos 24U:
   * 24U: El número 24 seguido de U indica que es un número entero sin signo (unsigned).
   * SCB\_CPUID\_IMPLEMENTER\_Pos: Este define la posición del campo “IMPLEMENTER” dentro del registro CPUID. En este caso, el campo comienza en el bit 24 del registro.
2. #define SCB\_CPUID\_IMPLEMENTER\_Msk (0xFFUL <<SCB\_CPUID\_IMPLEMENTER\_Pos):
   * 0xFFUL: Este es un número hexadecimal que representa 255 en decimal. El sufijo UL indica que es un número entero sin signo de tipo largo (unsigned long).
   * << SCB\_CPUID\_IMPLEMENTER\_Pos: Este es el operador de desplazamiento a la izquierda (left shift). Desplaza el valor 0xFFUL 24 posiciones a la izquierda. Esto significa que el valor 0xFF (que es 11111111 en binario) se moverá 24 bits a la izquierda, llenando los bits menos significativos con ceros.

Entonces, (0xFFUL << SCB\_CPUID\_IMPLEMENTER\_Pos) se convierte en 0xFF000000UL. Esto crea una máscara que puede ser usada para aislar o manipular el campo “IMPLEMENTER” en el registro CPUID.

A continuación, mostrare una lista de cada campo de configuración de registro para el SBD y su funcionamiento:

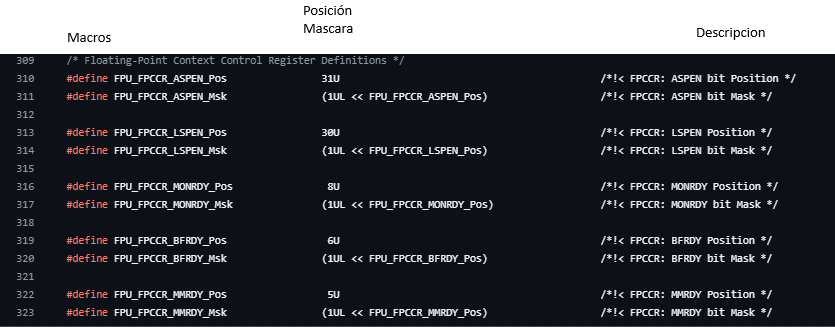
Definición de registros para la configuración del FPU a partir de estructuras:



Esto es lo que hace cada registro:

1. **FPCCR (Floating-Point Context Control Register)**:
   * **Función**: Controla el contexto de la FPU, incluyendo la preservación del estado y la disponibilidad de la FPU.
   * **Bits Importantes**:
     + ASPEN (bit 31): Habilita la preservación automática del estado de la FPU.
     + LSPEN (bit 30): Habilita la preservación perezosa del estado de la FPU.
     + MONRDY (bit 8): Indica si el monitor de depuración está listo.
     + BFRDY (bit 6): Indica si la FPU está lista para manejar fallos de bus.
     + MMRDY (bit 5): Indica si la FPU está lista para manejar fallos de gestión de memoria.
     + HFRDY (bit 4): Indica si la FPU está lista para manejar fallos graves.
     + THREAD (bit 3): Indica el modo del procesador (hilo o handler).
     + USER (bit 1): Indica el nivel de privilegio (usuario o privilegiado).
     + LSPACT (bit 0): Indica si la preservación perezosa del estado está activa.
2. **FPCAR (Floating-Point Context Address Register)**:
   * **Función**: Almacena la dirección del contexto de la FPU.
   * **Bits Importantes**:
     + ADDRESS (bits 3-31): Dirección del contexto de la FPU.
3. **FPDSCR (Floating-Point Default Status Control Register)**:
   * **Función**: Controla el estado predeterminado de la FPU.
   * **Bits Importantes**:
     + AHP (bit 26): Controla el redondeo de precisión alternativa.
     + DN (bit 25): Controla el modo de desnormalización.
4. **MVFR0 y MVFR1 (Media and FP Feature Registers)**:
   * **Función**: Proporcionan información sobre las características de la FPU y los medios.
   * **Bits Importantes**: Estos registros son de solo lectura y contienen información sobre las capacidades de la FPU, como el soporte para operaciones de punto flotante de simple y doble precisión.

La FPU de igual manera tiene macros de posición y mascara como el SCB, donde cada macro es para la configuración de la FPU como se muestra en la siguiente imagen:



Al final del programa FPU.h viene la siguiente funcion:

Texto

Descripción generada automáticamente

Esta funcion sirve para determinar el tipo de Unidad de Punto Flotante (FPU) presente en el sistema:

Se declara la variable mvfro de 32 bits sin signo para almacenar el valor del registro MVFRO de la FPU, el MVFR0 (Media and VFP Feature Register 0) es un registro de solo lectura que proporciona información sobre las características de la Unidad de Punto Flotante (FPU) y la arquitectura SIMD (Single Instruction, Multiple Data) avanzada en el procesador

**Campos del MVFR0**

Algunos de los campos importantes del registro MVFR0 incluyen:

* **SP (Single Precision)**: Indica si la FPU soporta operaciones de precisión simple.
* **DP (Double Precision)**: Indica si la FPU soporta operaciones de precisión doble.
* **(Divide)**: Indica si la FPU soporta operaciones de división en hardware.
* **SR (Square Root)**: Indica si la FPU soporta operaciones de raíz cuadrada en hardware

**En el programa FPU.c**

La condición para determinar el tipo de FPU:

|  |
| --- |
| if ((mvfr0 & (FPU\_MVFR0\_Single\_precision\_Msk | FPU\_MVFR0\_Double\_precision\_Msk)) == 0x020U)  {  return 1U; /\* Single precision FPU \*/  }  else  {  return 0U; /\* No FPU \*/  } |

* Se aplica una máscara al valor de mvfr0 para aislar los bits que indican el soporte de precisión simple y doble de la FPU.
* == 0x020U: Se compara el resultado con 0x020U (valor que indica que solo se soporta precisión simple).
* return 1U;: Si la condición es verdadera, la función devuelve 1, indicando que la FPU soporta precisión simple.
* return 0U;: Si la condición es falsa, la función devuelve 0, indicando que no hay FPU.

En resumen, esta función verifica si la FPU del sistema soporta precisión simple y devuelve 1 si es así, o 0 si no hay FPU.

* **Condicional de preprocesador**: Verifica si la FPU está presente (\_\_FPU\_PRESENT == 1) y si se va a utilizar (\_\_FPU\_USED == 1).
* SCB->CPACR: Accede al registro CPACR (Coprocessor Access Control Register) del SCB.
* |= ((3UL << 10\*2)|(3UL << 11\*2)): Configura los bits correspondientes a los coprocesadores CP10 y CP11 para darles acceso completo (Full Access).
  + 3UL: Es un valor de 2 bits (11 en binario) que indica acceso completo.
  + << 10\*2 y << 11\*2: Desplaza el valor 3UL a las posiciones correspondientes a CP10 y CP11 en el registro CPACR.