

# ARM System on a chip Architecture

FIUSAC. EIME.

Electrónica 5

# Sistema I/O

- ARM maneja periféricos de entrada salida y dispositivos mapeados en memoria con soporte de interrupción. Estas instrucciones aparecen en el mapa de la memoria y pueden ser escritos-leídos usando las mismas instrucciones (load-store) que cualquier otra localidad de memoria.
- Los periféricos pueden llamar la atención del procesador cuando se hace una solicitud de interrupción:
  - IRQ: interrupción normal, la mayoría de las interrupciones se hacen acá
  - FIQ: interrupción rápida, esta interrupción actúa más rápido porque tiene mayor prioridad
- Ambas interrupciones son enmascarables.
- Algunos sistemas pueden incluir acceso directo a memoria (DMA) para manejar mejor el tráfico de gran ancho de banda en los I/O

# Excepciones

- Son usadas para manejar eventos inesperados, los cuales suceden durante la ejecución del programa
- Se pueden considerar en 3 grupos:
  - Excepciones generadas como un efecto directo de ejecución de una instrucción: interrupciones de software, instrucciones indefinidas y prefetch aborts (instrucciones inválidas)
  - Excepciones generadas como un efecto apartado de una instrucción: data aborts (fallo en memoria durante la carga o guardado de datos de acceso)
  - Excepciones generadas externamente: Reset, I/O y FIO

- ARM soporta un rango de interrupciones, llamadas de supervisor, etc.
- La forma en que se atienden de forma general es:
  - El estado actual se guarda copiando el contenido de PC en R14\_exc y el contenido de CPSR en SPSR\_exc
  - El modo de operación del procesador se cambia al modo de excepción
  - PC se fuerza a un valor entre 00 y 1C, el valor específico dependerá del tipo de excepción
- El valor de PC se fuerza a una localidad de memoria dada por la dirección del vector de interrupción, normalmente contiene el salto a una subrutina donde se encuentra cómo manejar la interrupción. El portador de interrupción usará R13\_exc, el cual normalmente se inicializó previamente a un punto dedicado en la pila de la memoria, con el fin de guardar los contenidos de los registros
- Cuando se termina de ejecutar la interrupción, todos los valores regresan a la normalidad antes de que se procediera al cambio.

# Entrada de excepción

- Cuando ocurre una excepción, ARM completa la instrucción actual (menos con Reset, las cuales provocan que se atiendan inmediatamente), y luego atiende a la excepción.
- El procesador actúa con la siguiente secuencia de acciones:
  - Cambia al modo de operación correspondiente a la excepción
  - Salva la dirección de la siguiente instrucción en R14 del nuevo modo de operación
  - Salva el valor de CPSR en el SPSR del nuevo modo
  - Deshabilita IRQs, seteando el bit 7 en CPSR y, si la interrupción es FIQ, deshabilita las siguientes FIQs seteando el bit 6 en CPSR
  - Fuerza a PC a empezar a ejecutar la instrucción localizada según el vector de interrupción

# Tabla del vector de interrupción

| EXCEPCIÓN                      | MODOS | DIRECCIÓN DE VECTOR |
|--------------------------------|-------|---------------------|
| Reset                          | SVC   | 0x00000000          |
| Instrucción indefinida         | UND   | 0x00000004          |
| Interrupción de software (SWI) | SVC   | 0x00000008          |
| Prefetch abort                 | Abort | 0x0000000C          |
| Data abort                     | Abort | 0x00000010          |
| IRQ                            | IRQ   | 0x00000018          |
| FIQ                            | FIQ   | 0x0000001C          |

# Vector de Interrupción

- Normalmente, este vector contiene un Branch a una rutina específica, aunque el código para FIQ puede empezar inmediatamente una vez se haya ocupado la dirección del vector más alta.
- Los registros de banco en cada modo de privilegio, son usados para mantener la dirección de retorno u la del puntero de pila.
- FIQ tiene registros privados adicionales para dar un mejor funcionamiento para evitar salvar registros de modo usuario, los cuales normalmente ya fueron guardados

# Retorno de excepción

- Una vez fue atendida la excepción, se debe reestablecer el código y el estado del usuario tal y como estaba antes de que se trabajara, para lo cual:
  - Cualquier registro de usuario modificado debe reestablecerse con los datos colocados en la pila
  - CPSR se reestablece según la información almacenada en SPSR
  - PC debe cambiar al dato con la instrucción a ejecutar en el programa principal.



# Retorno de una excepción

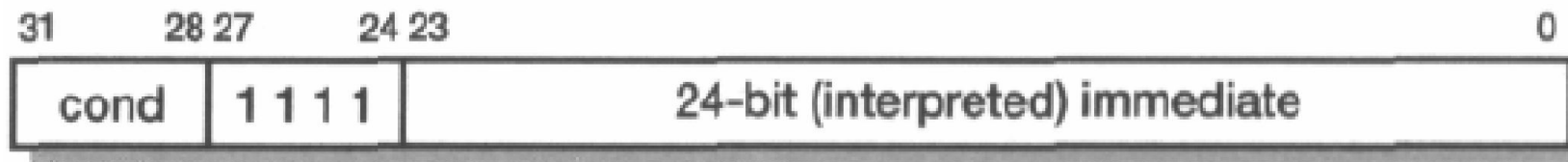
- Una de estas direcciones de retorno fueron almacenados en R14 cuando la información se almacenó en la pila.
  - Para retorno de SWI o instrucción indefinida:
    - `MOVS pc, r14,`
  - Para retorno de una IRQ, FIQ o prefetch abort:
    - `SUBS pc, r14, #4`
  - Para retorno de data abort para recuperar data Access:
    - `SUBS pc, r14, #8`
- El modificador “s” significa la forma especial de una instrucción cuando el registro destino es PC.
- Note cómo una instrucción de retorno incorpora un ajuste a la dirección de retorno cuando es necesario:
  - IRQ y FIQ deben retornar una instrucción antes en orden a ejecutar la instrucción que fue usurpada por la solicitud de excepción
  - Prefetch abort debe retornar una instrucción antes para ejecutar la instrucción que causó el error en el fallo de memoria cuando se solicite con anterioridad.
  - Data abort debe regresar 2 instrucciones antes para tratar de ejecutar la instrucción de transferencia, la cual fue la instrucción antes de que fuera usurpada la entrada de excepción.

# Prioridades en excepciones

- Varias excepciones pueden solicitarse al mismo tiempo, por lo tanto, es necesario definir el orden de prioridad para determinar el orden en el que éstas serán atendidas. En ARM el orden es:
  - RESET
  - Data abort
  - FIQ
  - IRQ
  - Prefetch abort
  - SWI, instrucción indefinida (incluye coprocesador ausente)

# Interrupción de software (SWI)

- Es usada para llamadas al sistema operativo y a menudo se llama llamada de supervisor.
- Cuando sucede coloca el procesador en el modo supervisor e inicia a ejecutar instrucciones de la dirección 0x08. Esta es un área de memoria protegida y es posible construir un sistema operativo, el cual es protegido de usuarios maliciosos.



**Figure 5.5** Software interrupt binary encoding.

- El campo donde están los 24 bits inmediatos no tienen influencia en la operación de la instrucción, pero es interpretado por el código del sistema
- Si la condición sucede cuando se pasó la entrada al modo supervisor usando el estándar de entrada a excepción de ARM, el procesador hará las siguientes acciones:
  - Guardar la dirección de la instrucción después de SWI en r14\_svc
  - Guardar el CPSR en SPSR\_cvc
  - Ingresar el modo supervisor y deshabilitar IRQs (pero no FIQs) colocando el CPSR [4:0] a 10011 y CPSR [7] tol.
  - Colocar el PC a 08 e iniciar a ejecutar las instrucciones colocadas acá
- Para retornar a la instrucción después de SWI la rutina del sistema no solo debe copiar r14\_svc de vuelta en PC, además debe restaurar CPSR de SPSR\_svc. Esto requiere el uso de una de las formas especiales de la instrucción de procesamiento de datos.

# Instrucciones de procesamiento de datos

- Son usadas para modificar

# Organización ARM pipeline 3 etapas

- Sus principales componentes son:
  - El registro banco, almacena el estado del procesador. Tiene 2 puertos de lectura y 1 de escritura los cuales pueden ser usados por cualquier registro, y adicional leer y escribir del R15
  - Barrel shifter, el cual puede intercambiar o rotar un operando por cualquier número de bits
  - ALU, desempeña funciones aritméticas y lógicas requeridas por el set de instrucciones
  - Registro de dirección e incrementador, el cual selecciona y mantiene toda las direcciones de memoria y genera direcciones secuenciales cuando se requiere.
  - Registros de datos, los cuales mantienen que pasan de o hacia la memoria
  - Decodificador de instrucciones y control lógico asociado.

# Pipeline de 3 etapas

- Fetch: la instrucción es traída de la memoria y colocada en la instrucción pipeline
- Decode: la instrucción es decodificada y el control de señales de datapath se preparan para el siguiente ciclo. En esta etapa, la instrucción posee la lógica de decodificación pero no el datapath
- Execute: la instrucción posee el datapath, el registro de banco es leído y el operando es rotado, el resultado de la ALU se genera y escribe en el registro destino.
- En cualquier momento, instrucciones diferentes pueden ocupar cada una de estas etapas, porque es capaz de realizar operaciones diferentes.