ARM Exceptions, Interrupts, Vector table

Exception

Exception - halt the normal sequential execution of instructions

Exception handlers

- handling errors
- handling interrupts
- other events generated by external system
- improves system performance

Exception

Exception

- Data abort
- Fast interrupt request
- Interrupt request
- Prefetch abort
- Software interrupt
- Reset
- Undefined instruction

Exception Handling

An exception is any condition that needs to halt the normal sequential execution of instructions

Examples

- ARM core is reset
- Failed instruction fetch or memory access failure
- encountering undefined instruction
- Software interrupt instruction execution
- external interrupt is raised

Exception handling is the method of processing these exceptions.

Exception Handling

- Most exceptions have associated an software exception handler
- Software exception handler - a software routine that executes when an exception occurs
- A data abort exception will have a data abort handler.
- The handler first determines the cause of the exception and then services the exception.

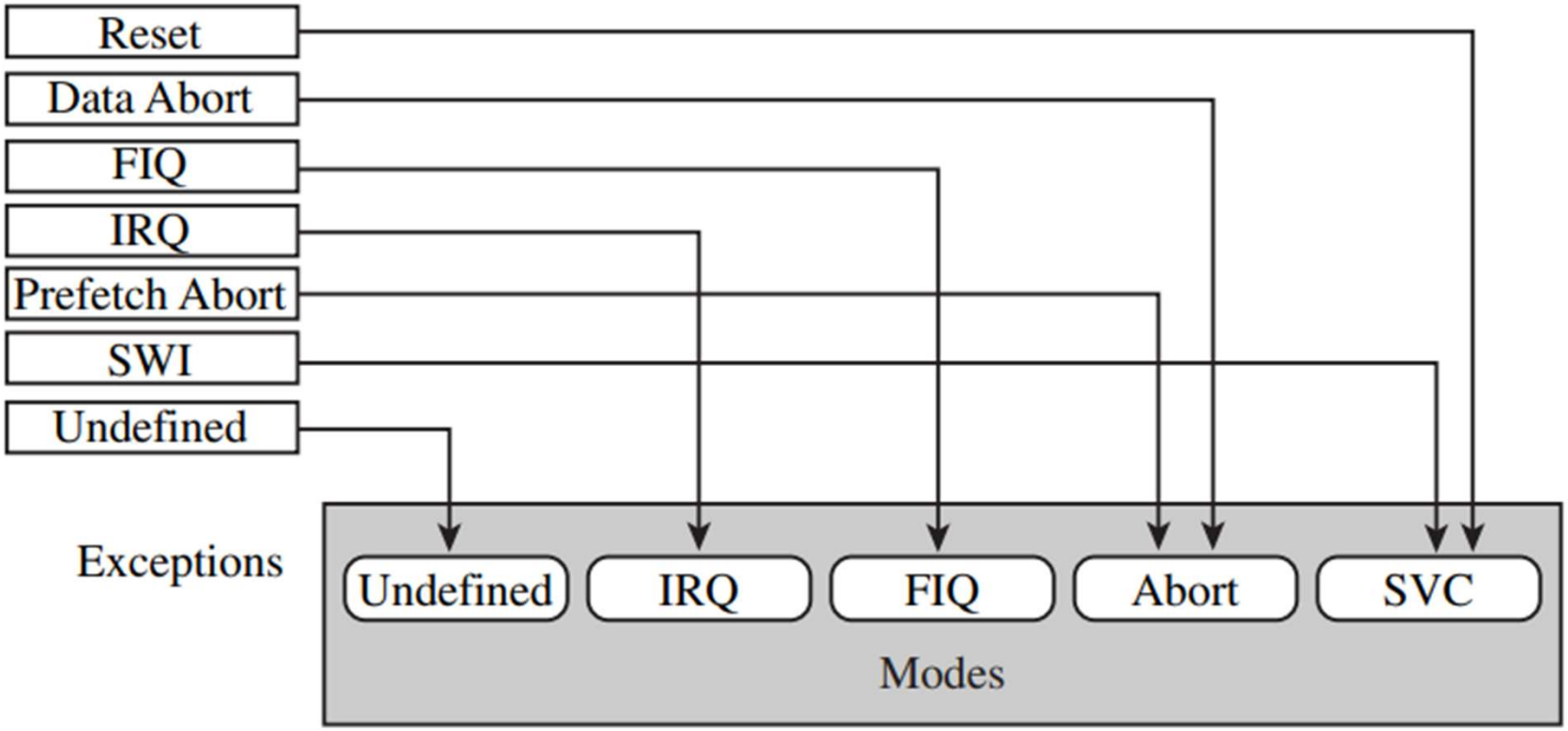
ARM processor exceptions and modes

- Each exception causes the core to enter a specific mode.
- Any ARM processor modes can be entered manually by changing the cpsr
- When an exception causes a mode change, the core automatically,
  - saves the cpsr to the spsr of the exception mode
  - saves the pc to the lr of the exception mode
  - sets the cpsr to the exception mode
  - sets pc to the address of the exception handler

ARM processor exceptions and modes

ARM processor exceptions and associated modes.

Exception	Mode	Main purpose
Fast Interrupt Request	*FIQ*	fast interrupt request handling
Interrupt Request	*IRQ*	interrupt request handling
SWI and Reset	*SVC*	protected mode for operating systems
Prefetch Abort and Data Abort	*abort*	virtual memory and/or memory protection handling
Undefined Instruction	*undefined*	software emulation of hardware coprocessors



Exceptions and associated modes.

Vector Table

Vector Table - a table of addresses that the ARM core branches to when an exception is raised.

- B<address> - This branch instruction provides a branch relative from the pc.
- LDR pc, [pc, offset] - This load register instruction loads the handler address from memory to the pc.
- The address is an absolute 32 bit address stored close to the vector table.
- Loading this absolute literal value results in a slight delay in branching to a specific handler due to the extra memory access.
- Branch to any address in memory.

Vector Table

- LDR pc, [pc, hash-0xff0] - loads a specific interrupt service routine address from address 0xfffff030 to PC
- This specific instruction is only used when a vector interrupt controller is present.
- MOV pc, hash immediate - copies an immediate value into the pc

Vector Table

Vector table and processor modes.

Exception	Mode	Vector table offset
Reset	SVC	+0x00
Undefined Instruction	UND	+0x04
Software Interrupt (SWI)	SVC	+0x08
Prefetch Abort	ABT	+0x0c
Data Abort	ABT	+0x10
Not assigned	—	+0x14
IRQ	IRQ	+0x18
FIQ	FIQ	+0x1c

Exception priorities

- Simultaneously occurring exceptions - priority mechanism
- Highest priority exception - reset
- data abort
- lowest priority - software interrupt, undefined instruction exception
- Interrupts can be disabled by setting I and F bits in the cpsr

Exception priorities

Reset Handler -

- initializes the system, including setting up memory and caches
- External interrupt sources should be initialized before enabling IRQ and FIQ interrupts
- Set up stack pointers for all processor modes

- during the first few instructions it is assumed that no exceptions or interrupts will occur
- code should be designed to avoid SWI, undefined instructions, memory access that may abort

Data abort exception

- Data abort exceptions - occur when the memory controller or MMU indicates that an invalid address is accessed
- When the current code attempts to read or write to memory without the correct access permissions
- An FIQ exception can be raised within a data abort handler since FIQ exceptions are not disabled
- When an FIQ is completely serviced, control is returned back to data abort

Data abort exception

FIQ - occurs when an external peripheral sets the FIQ pin

FIQ - highest priority

Exception priority levels.

Exceptions	Priority	*I* bit	*F* bit
Reset	1	1	1
Data Abort	2	1	—
Fast Interrupt Request	3	1	1
Interrupt Request	4	1	—
Prefetch Abort	5	1	—
Software Interrupt	6	1	—
Undefined Instruction	6	1	—

Data abort exception

- core disables both IRQ and FIQ once it enters into FIQ handler
- no external source can interrupt the processor unless the IRQ and FIQ exceptions are reenabled by software

IRQ exception

- IRQ exception occurs when an external peripheral sets the pin
- Second-highest priority exception
- IRQ handler will be entered if neither an FIQ exception nor a data abort exception occurs
- On entry to IRQ handler, IRQ exceptions are disabled until the current interrupt source has been cleared

Prefetch abort

- A prefetch abort exception occurs when an attempt to fetch an instruction results in a memory fault.
- This exception is raised when the instruction is in the execute stage of the pipeline and if none of the higher exceptions have been raised
- On entry to the handler, IRQ exceptions will be disabled, FIQ will remain unchanged

SWI exception

- SWI exception occurs when the SWI instruction is executed and none of the other higher-priority exceptions have been flagged
- On entry to the handler, cpsr will be set to supervisor mode
- if the system uses nested swi calls, link reg r14 and spsr must be stored away before branching to the nested SWI

Undefined exception

- An undefined instruction exception occurs when an instruction not in the ARM or Thumb instruction set reaches the execute stage of the pipeline and none other exceptions are flagged.
- The ARM processor asks the coprocessors if they can handle this as a coprocessor instruction.
- If none of the coprocessors claims the instruction, an undefined instruction exception is raised
- SWI and Undefined instruction - same priority
- they cannot occur at the same time

Link Register Offsets

- When an exception occurs, the link register is set to a specific address based on the current pc.
- A list of useful addresses for the different exceptions

Exception	Address	Use
Reset	—	*lr* is not defined on a Reset
Data Abort	*lr* − 8	points to the instruction that caused the Data Abort exception
FIQ	*lr* − 4	return address from the FIQ handler
IRQ	*lr* − 4	return address from the IRQ handler
Prefetch Abort	*lr* − 4	points to the instruction that caused the Prefetch Abort exception
SWI	*lr*	points to the next instruction after the SWI instruction
Undefined Instruction	*lr*	points to the next instruction after the undefined instruction

Interrupts

Two Types -

- An exception raised by an external peripheral- IRQ, FIQ
- SWI instruction

Both suspend the normal flow of program execution

Assigning Interrupts

- A system designer decides which hardware peripheral can produce which interrupt request.
- An interrupt controller connects multiple external interrupts to one of the two ARM interrupt requests.
- Interrupt controller can be programmed to allow an external interrupt source to cause either an IRQ or FIQ exception.
- Software interrupts are normally reserved to call privileged os routines.
- SWI instruction - used to change a program running in user mode to a privileged mode
- Interrupt requests are assigned for general purpose interrupts (timer interrupt)
- FIQ are reserved for a single interrupt source that requires a fast response time