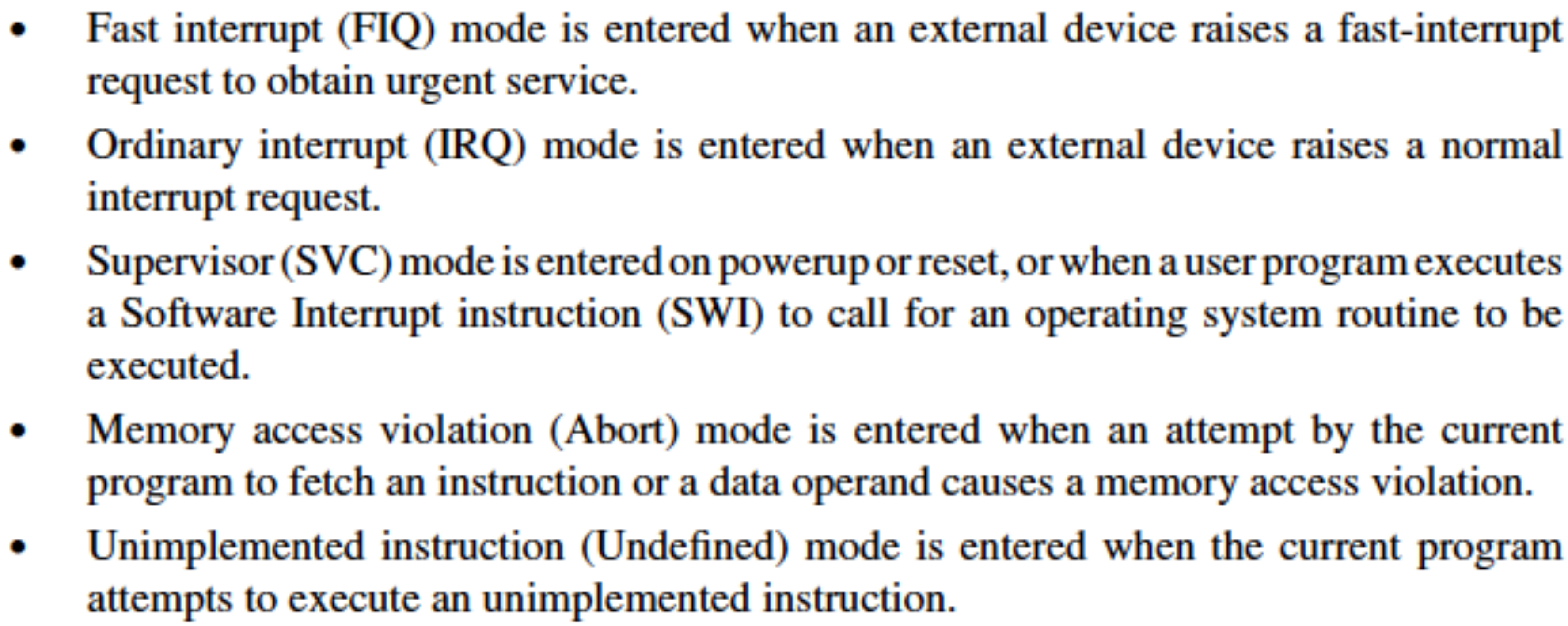
ARM Operating Modes

- Supports 7 Modes
  - System Mode
    - Privileged mode for exception handling
    - Uses same registers as user mode
    - Can only be entered from another exception
  - User Mode
    - For user applications
  - Five exceptions modes

ARM Operating Modes

Exception Types



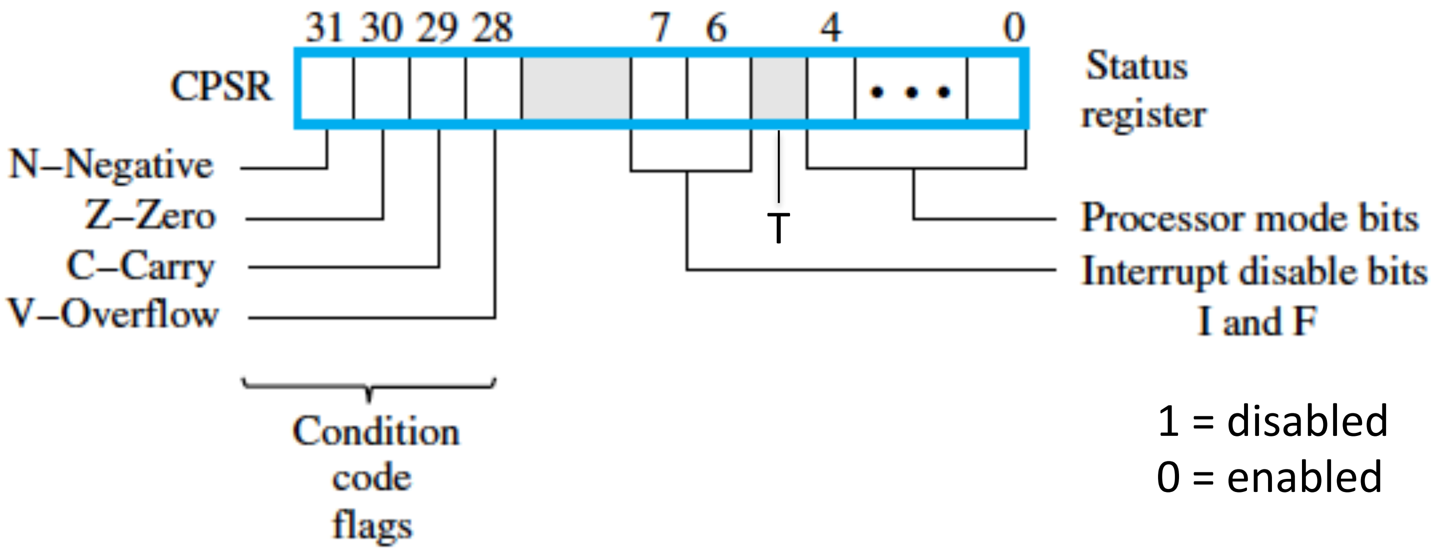
- Fast interrupt (FIQ) mode is entered when an external device raises a fast-interrupt request to obtain urgent service.
- Ordinary interrupt (IRQ) mode is entered when an external device raises a normal interrupt request.
- Supervisor (SVC) mode is entered on powerup or reset, or when a user program executes a Software Interrupt instruction (SWI) to call for an operating system routine to be executed.
- Memory access violation (Abort) mode is entered when an attempt by the current program to fetch an instruction or a data operand causes a memory access violation.
- Unimplemented instruction (Undefined) mode is entered when the current program attempts to execute an unimplemented instruction.

JOHNS HOPKINS UNIVERSITY | Engineering for Professionals

ARM Operating Modes

System Mode & exception modes are privileged modes.

Access to CPSR is allowed so I and F bits can be changed



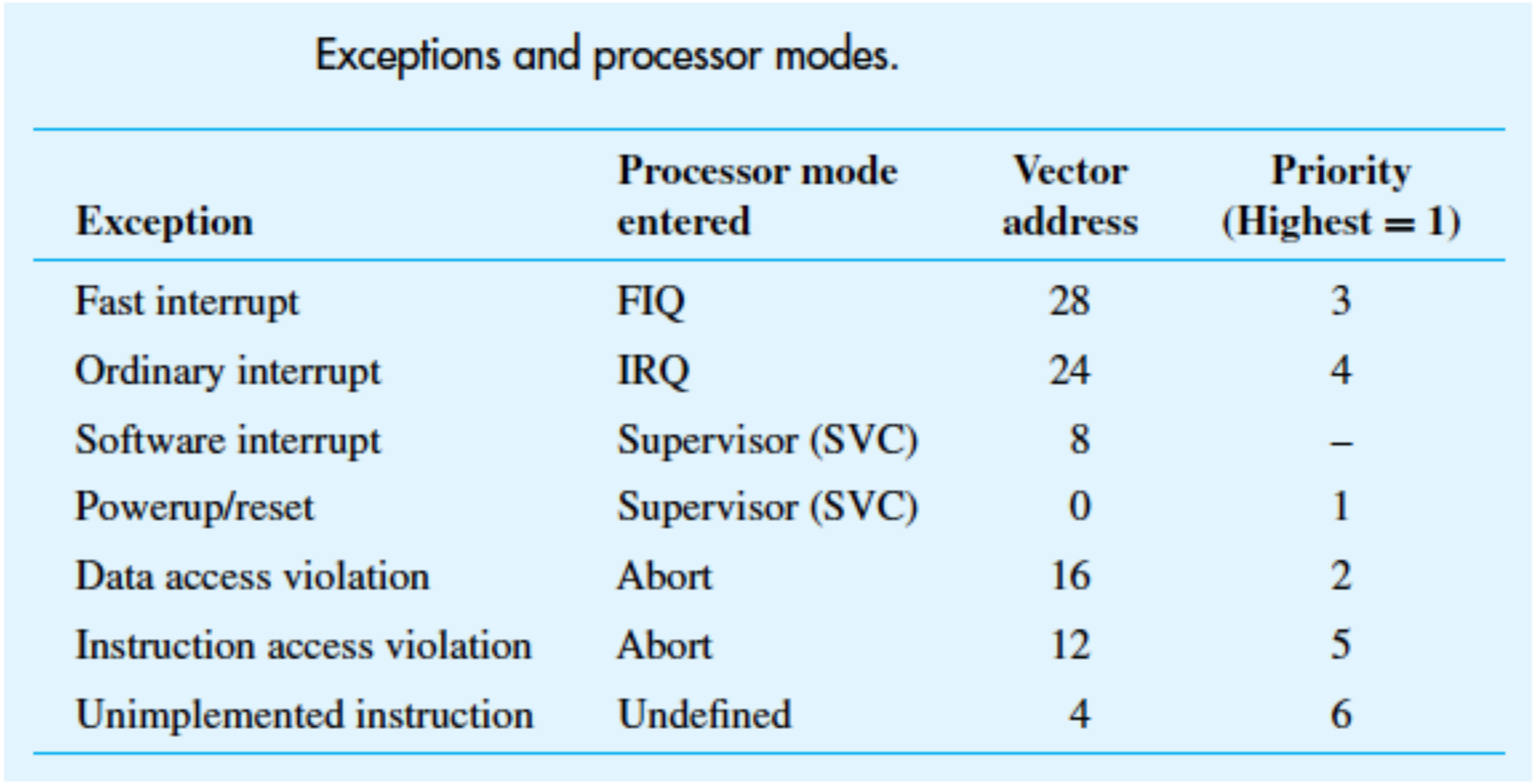
1 = disabled

0 = enabled

User mode is unprivileged, so instructions that change CPSR are not available.

ARM Operating Modes

- Exception Handling
  - Vector table in low memory maps exception code



Exceptions and processor modes.

Exception	Processor mode entered	Vector address	Priority (Highest = 1)
Fast interrupt	FIQ	28	3
Ordinary interrupt	IRQ	24	4
Software interrupt	Supervisor (SVC)	8	–
Powerup/reset	Supervisor (SVC)	0	1
Data access violation	Abort	16	2
Instruction access violation	Abort	12	5
Unimplemented instruction	Undefined	4	6

ARM Operating Modes

- Banked Registers
  - Exceptions switch from user mode to one of 5 exception modes
  - Extra (banked) registers are substituted for some of the 16 normal registers used in System or User mode
  - Replaced registers are left unchanged (no need to save)
  - There is a set of banked registers for each exception mode

- *"Banked"* registers are used in exception handling

ARM Operating Modes

Accessible registers in different modes of the ARM processor.

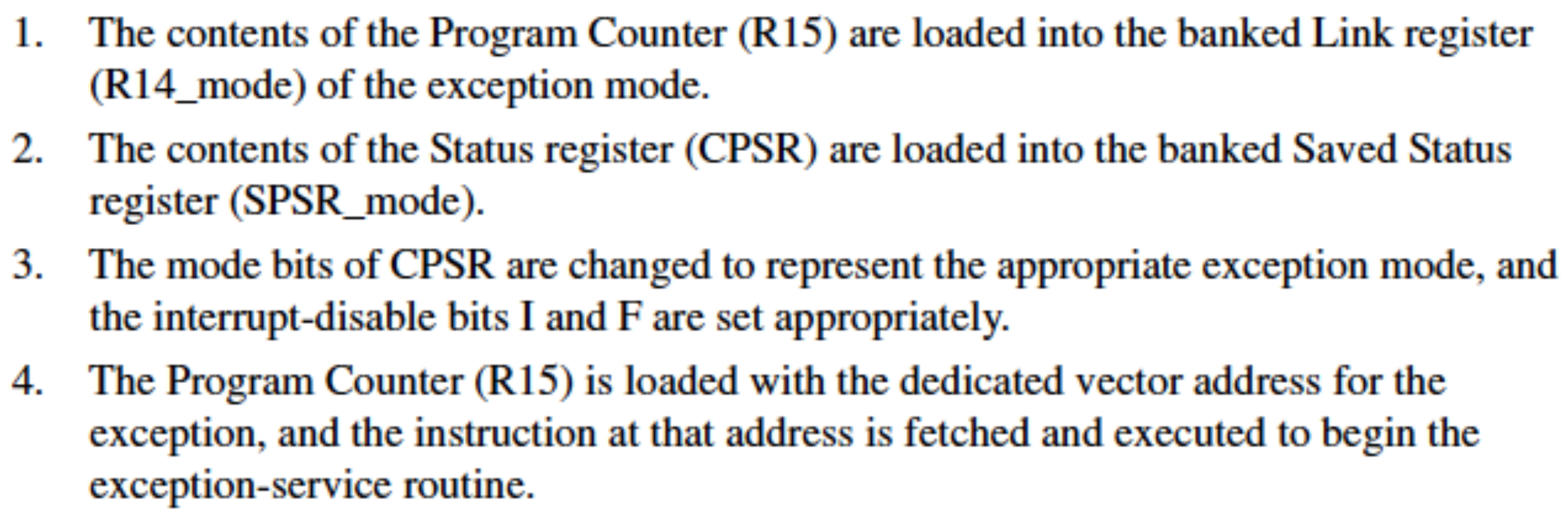
User/System	FIQ	IRQ	Supervisor	Abort	Undefined
R0	R0	R0	R0	R0	R0
R1	R1	R1	R1	R1	R1
R2	R2	R2	R2	R2	R2
R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7
R8	R8_fiq	R8	R8	R8	R8
R9	R9_fiq	R9	R9	R9	R9
R10	R10_fiq	R10	R10	R10	R10
R11	R11_fiq	R11	R11	R11	R11
R12	R12_fiq	R12	R12	R12	R12
R13	R13_fiq	R13_irq	R13_svc	R13_abt	R13_und
R14	R14_fiq	R14_irq	R14_svc	R14_abt	R14_und
R15	R15	R15	R15	R15	R15

Processor status register

CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
	SPSR_fiq	SPSR_irq	SPSR_svc	SPSR_abt	SPSR_und

ARM Operating Modes

- Actions taken when exception occurs

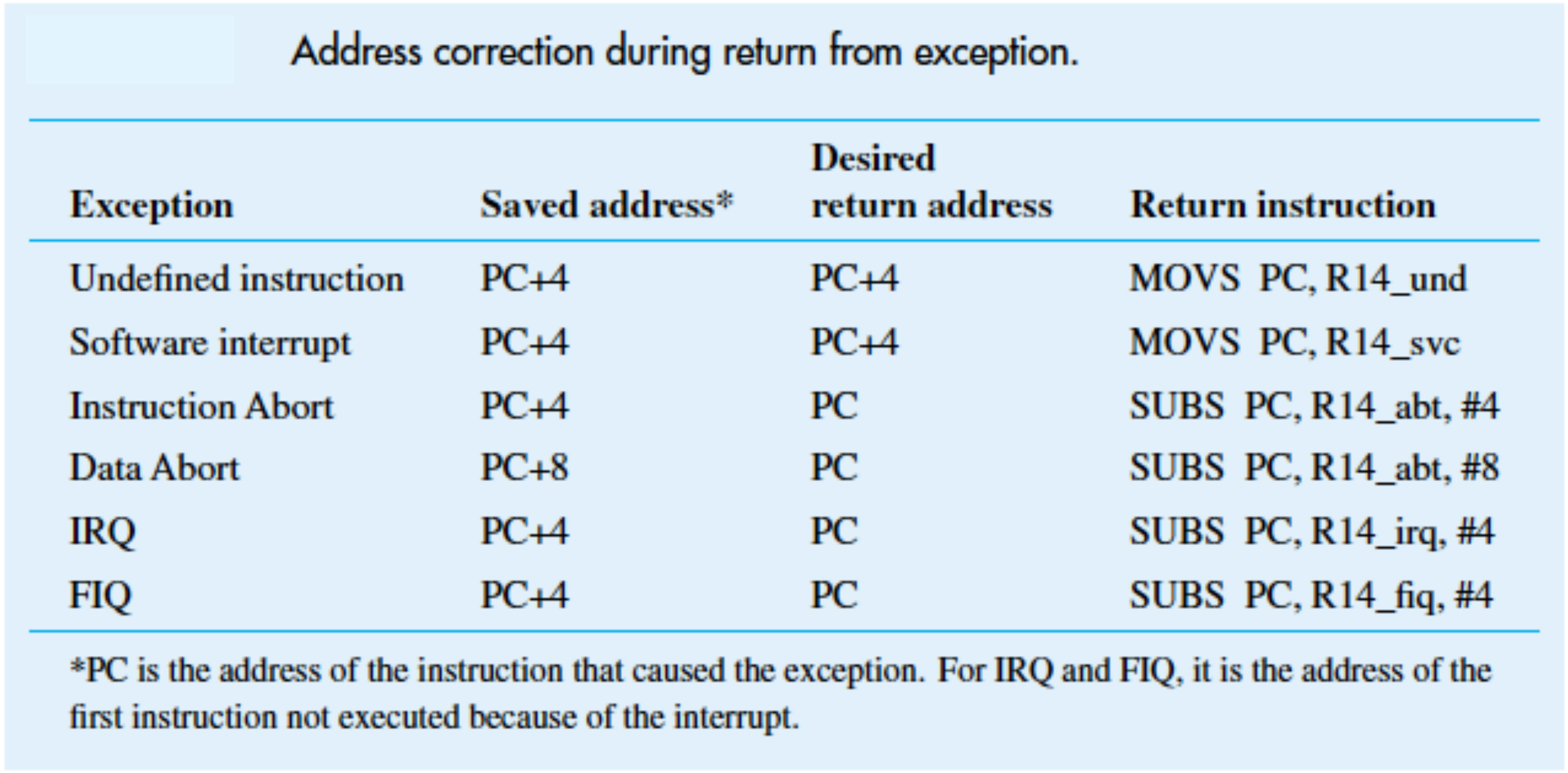


1. The contents of the Program Counter (R15) are loaded into the banked Link register (R14_mode) of the exception mode.
2. The contents of the Status register (CPSR) are loaded into the banked Saved Status register (SPSR_mode).
3. The mode bits of CPSR are changed to represent the appropriate exception mode, and the interrupt-disable bits I and F are set appropriately.
4. The Program Counter (R15) is loaded with the dedicated vector address for the exception, and the instruction at that address is fetched and executed to begin the exception-service routine.

ARM Operating Modes

- Return from exception
    - Operating System initializes R13_mode to point to top of stack area for each exception mode
    - Return from an exception handler is performed by copying the mode link register (R14_mode) into the program counter and copying the SPSR-mode register into the CPSR
        - Example: SUBS PC,R14_irq,#4
            - Sets PC = R14_irq – 4
            - The S suffix means copy SPSR_irq into CPSR
    - MOVS PC,R14_svc   returns from software interrupt

ARM Operating Modes



Address correction during return from exception.

Exception	Saved address*	Desired return address	Return instruction
Undefined instruction	PC+4	PC+4	MOVS PC, R14_und
Software interrupt	PC+4	PC+4	MOVS PC, R14_svc
Instruction Abort	PC+4	PC	SUBS PC, R14_abt, #4
Data Abort	PC+8	PC	SUBS PC, R14_abt, #8
IRQ	PC+4	PC	SUBS PC, R14_irq, #4
FIQ	PC+4	PC	SUBS PC, R14_fiq, #4

*PC is the address of the instruction that caused the exception. For IRQ and FIQ, it is the address of the first instruction not executed because of the interrupt.