Cortex M0+

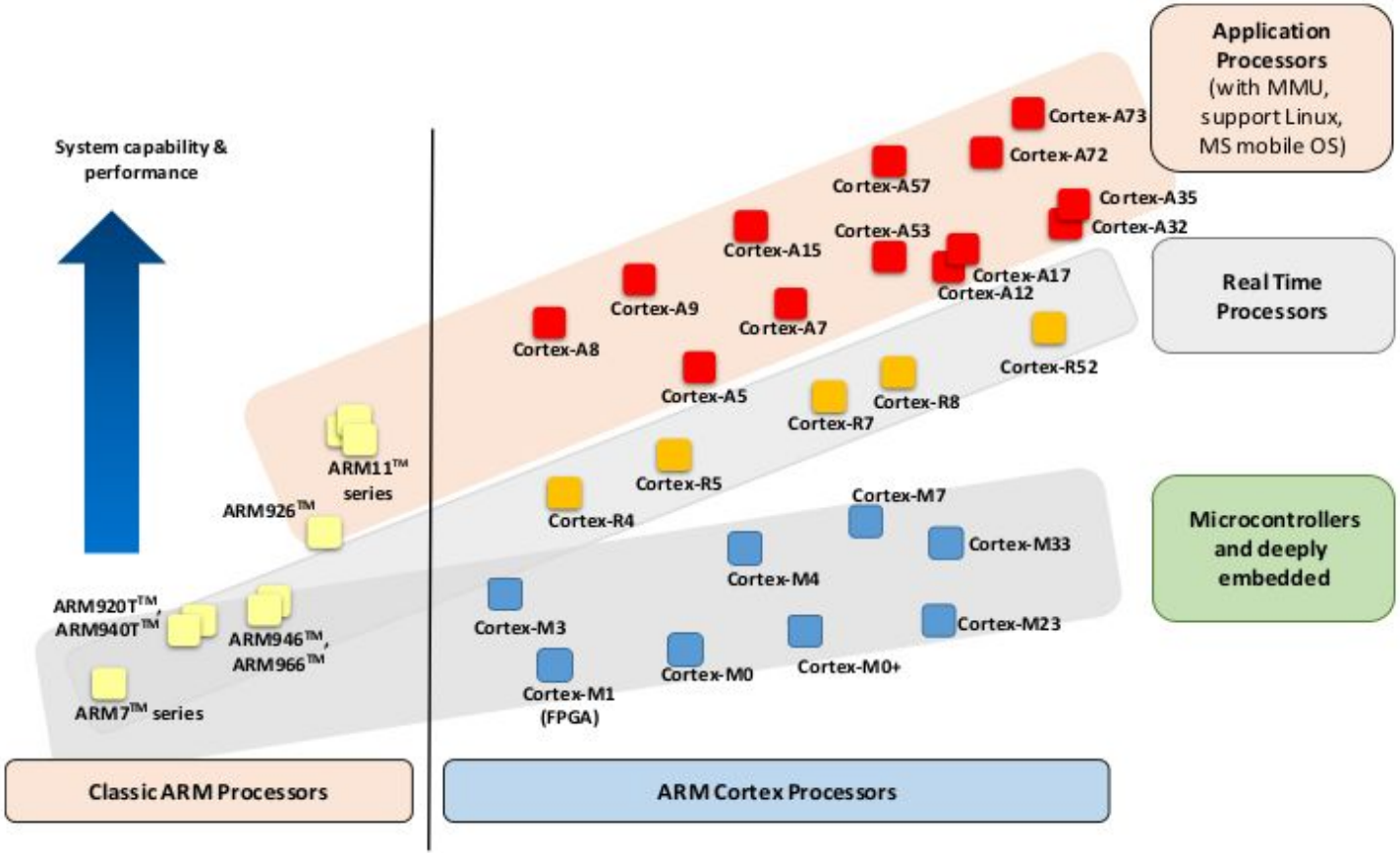


Figure 1: ARM processor family

Features

A low gate count processor that features:

—The ARMv6-M Thumb® instruction set.

—Thumb-2 technology

—Optionally, an ARMv6-M compliant 24-bit SysTick timer.

—A 32-bit hardware multiplier. This can be the standard single-cycle multiplier, or a 32-cycle multiplier that has a lower area and performance Implementation.

—Low power sleep-mode entry

Features

—The system interface supports either little-endian or byte invariant big-endian data accesses.

—The ability to have deterministic, fixed-latency, interrupt handling.

—Load/store-multiples and multicycle-multiplies that can be abandoned and restarted to facilitate rapid interrupt handling.

—C Application Binary Interface compliant exception model.

Features

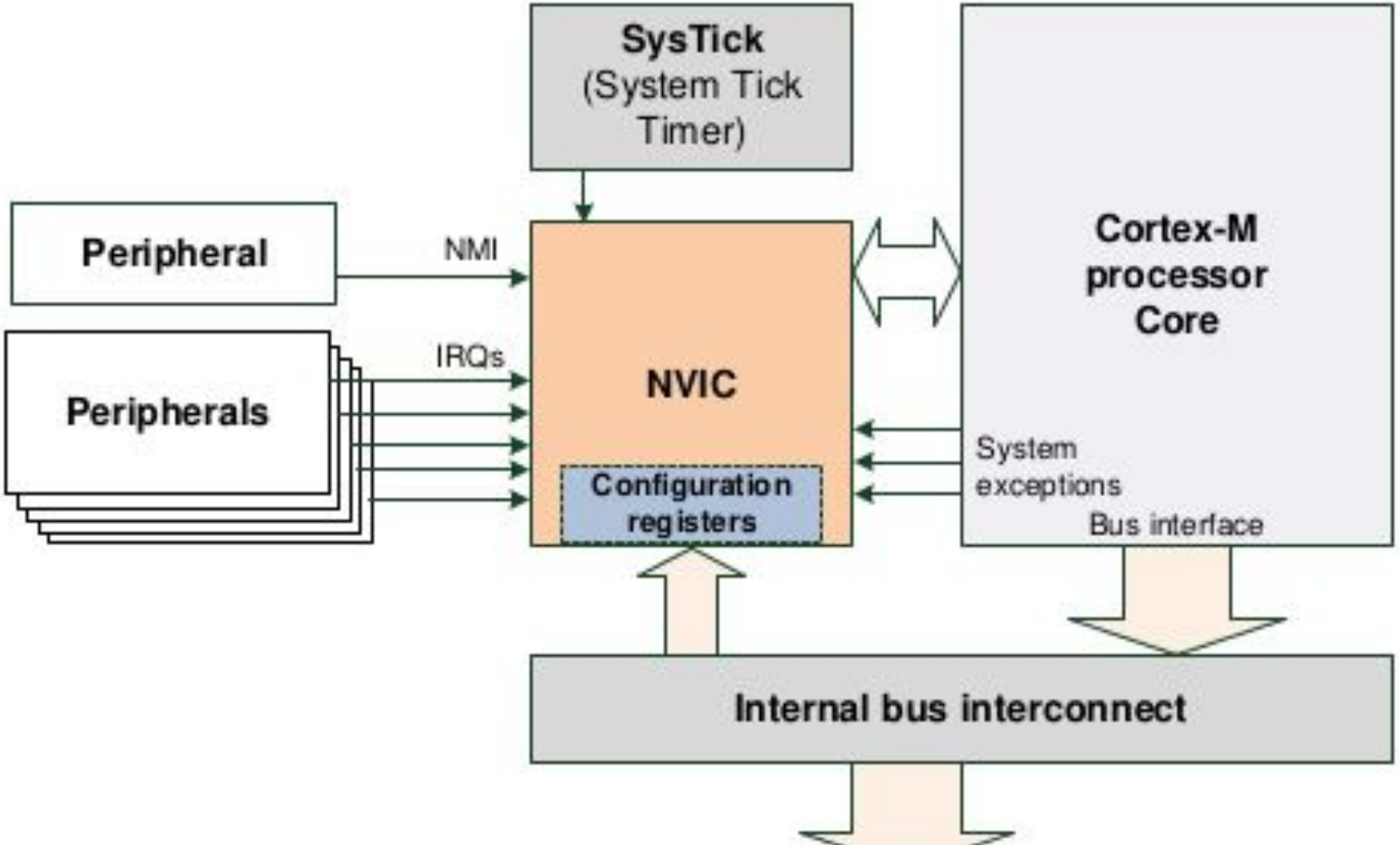
NVIC that features:

—1, 2, 4, 8, 16, 24, or 32 external interrupt inputs, each with four levels of priority

—dedicated Non-Maskable Interrupt (NMI) input

—support for both level-sensitive and pulse-sensitive interrupt lines

—optional Wake-up Interrupt Controller (WIC), providing ultra-low power sleep mode support.


SysTick
(System Tick
Timer)
Peripheral
NMI
Peripherals
IRQs
NVIC
Configuration
registers
Cortex-M
processor
Core
System
exceptions
Bus interface
Internal bus interconnect

Programmers Model

1. Privileged Mode
   a. Access to all part of the memories
2. Unprivileged access level
   a. Access to restricted parts of the memories
3.

Operation Modes and States

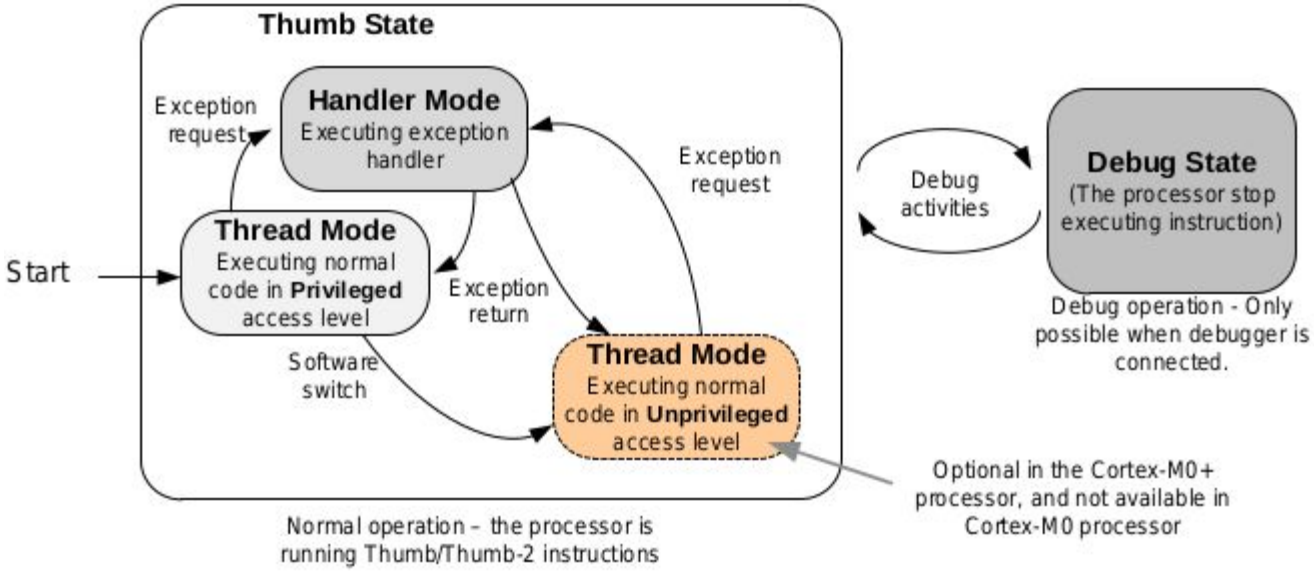
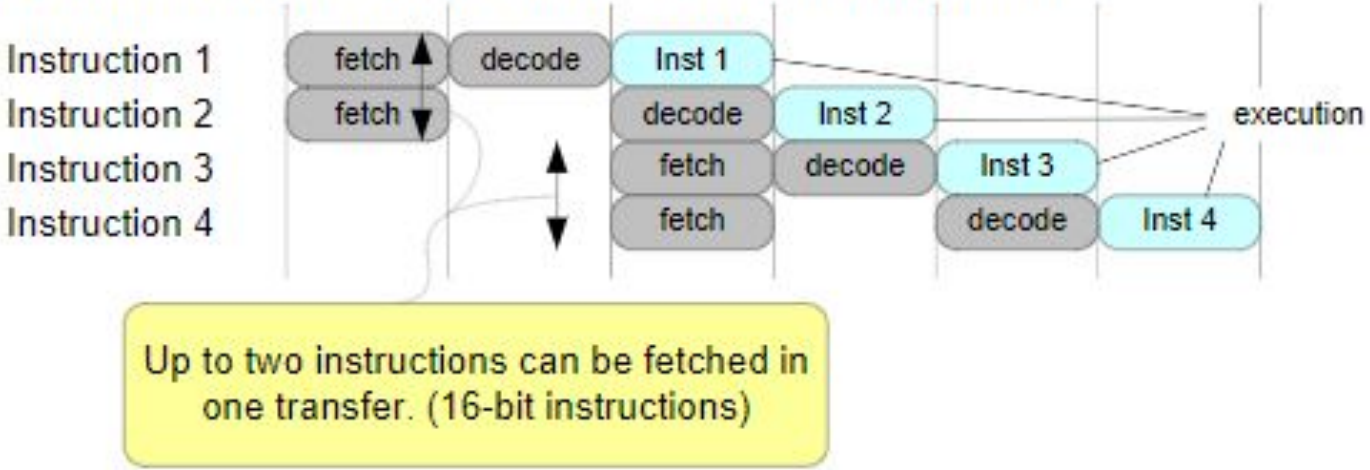
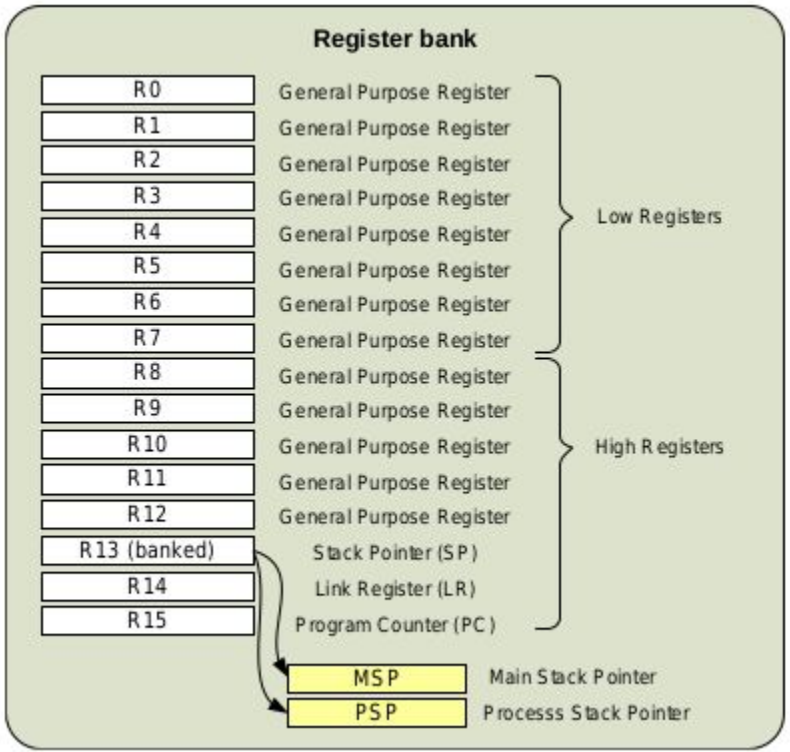
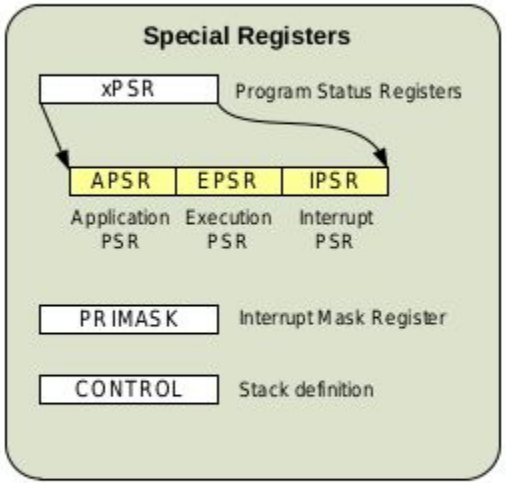


Figure 3. Pipeline stages in the Cortex-M0 processor



Register Bank





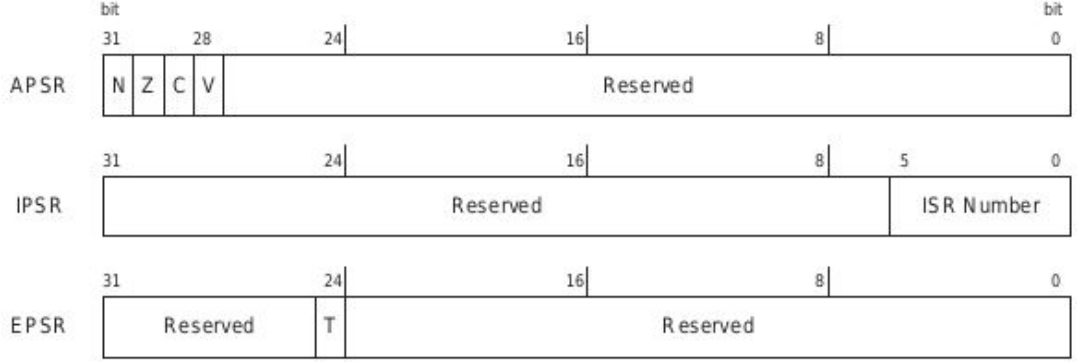
Register Bank

- R0-R12
  - Registers R0-R12 are for general uses.
- R13, Stack Pointer
  - R13 is the Stack Pointer. It is used for accessing the stack memory via PUSH and POP operations.
  - The Main Stack Pointer (MSP, or SP_main in ARM documentation) is the default Stack Pointer after reset, and is used when running exception handlers.
  - The Process Stack Pointer (PSP, or SP_process in ARM documentation) can only be used in Thread mode (when not handling exceptions).

Register Bank

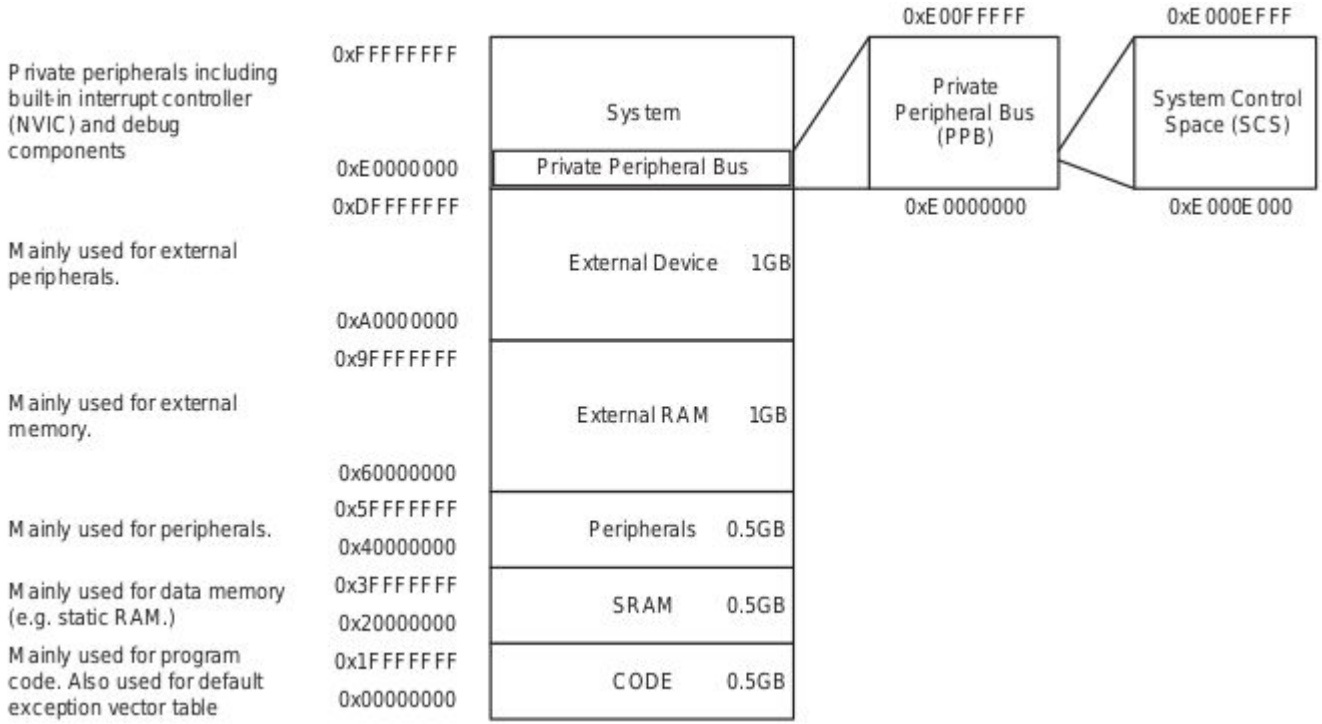
- R14, Link Register
  - R14 is the Link Register (LR). The LR is used for storing the return address of a subroutine or function call.
- R15, Program Counter
  - R15 is the PC. It is readable and writeable. A read returns the current instruction address plus four (this is caused by the pipeline nature of the design).
- PRIMASK Interrupt Mask Special Register
  - The PRIMASK register is a 1-bit wide interrupt mask register. When set, it blocks all interrupts apart from the Non-Maskable Interrupt (NMI) and the HardFault exception.
- CONTROL Special Register
  - The stack pointer selection is determined by the processor mode as well as the configuration of the CONTROL register

xPSR, Combined Program Status Register



- ➔ The APSR contains the ALU flags: N (negative flag), Z (zero flag), C (carry or borrow flag), and V (overflow flag). These bits are at the top 4 bits of the APSR.
- ➔ The IPSR contains the current executing ISR (Interrupt Service Routine) number. Each exception on the Cortex-M0/M0+ processor has a unique associated ISR number
- ➔ The EPSR on the Cortex-M0/M0þ processor contains the T bit which indicates that the processor is in the Thumb state.

Memory System



Exceptions and Interrupts

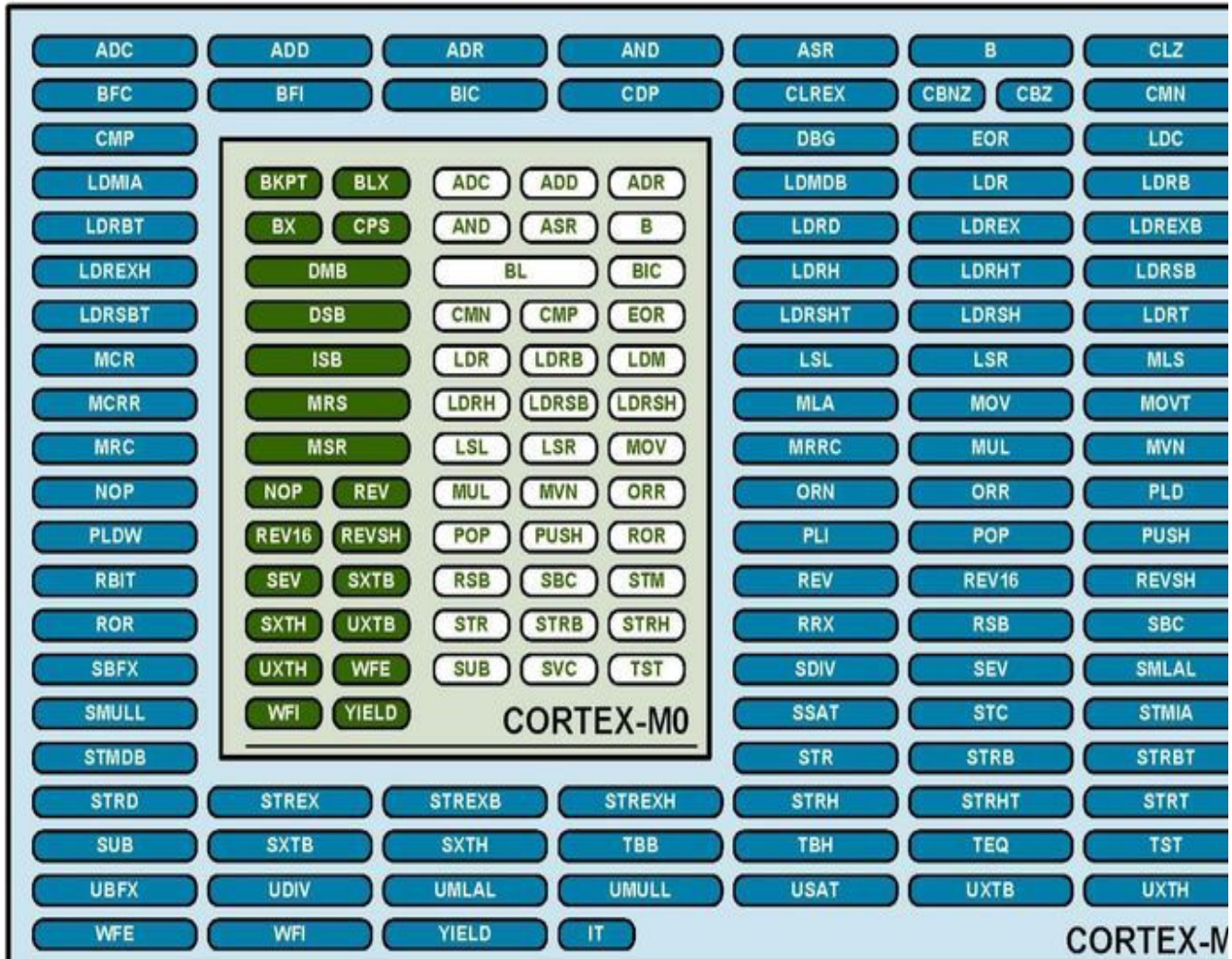
Exception type	Exception number	Description
Reset	1	Power on reset or system reset.
NMI	2	Non-Maskable interrupt—highest priority exception that cannot be disabled. For safety critical events.
HardFault	3	For fault handling—activated when a system error is detected.
SVCall	11	Supervisor call—activated when SVC instruction is executed. Primarily for OS applications.
PendSV	14	Pendable service (system) call—activate by writing to an interrupt control and status register. Primarily for OS applications.
SysTick	15	System Tick timer exception – typically used by an OS for a regular system tick exception. The system tick timer (SysTick) is an optional[a] timer unit inside the Cortex®-M processor.
IRQ0 to IRQ31[b]	16–47	Interrupts—can be from external sources or from on-chip peripherals.

Exceptions and Interrupts

- Exceptions are events that cause changes to program control: when an exception occurred,
- Instead of continuing program execution, the processor suspends the current executing task
- and executes a part of the program code called the exception handler.
- After the exception handler is completed, it will then resume the normal program execution.

Nested Vectored Interrupt Controller

- Flexible Interrupt Management
  - In the Cortex-M processors, each external interrupt can be enabled, disabled, and can have its pending status set or clear by software.
- Nested Interrupt Support
  - When an exception occurs such as an external interrupt, the NVIC will compare the priority of this exception to the current level.
  - If the new exception has a higher priority, the current running task will be suspended
  - Some of the registers will be stored on to the stack memory and the processor will start executing the exception handler of the new exception. This process is called “preemption.”
- Vectored Exception Entry
  - When an exception occurs, the processor will need to locate the starting point of the corresponding exception handler.
- Interrupt Masking


ADC
ADD
ADR
AND
ASR
B
CLZ
BFC
BFI
BIC
CDP
CLREX
CBNZ
CBZ
CMN
CMP
DBG
EOR
LDC
LDMIA
LDMDB
LDR
LDRB
LDRBT
LDRD
LDREX
LDREXB
LDREXH
LDRH
LDRHT
LDRSB
LDRSBT
LDRSHT
LDRSH
LDRT
MCR
LSL
LSR
MLS
MCRR
MLA
MOV
MOVT
MRC
MRRC
MUL
MVN
NOP
ORN
ORR
PLD
PLDW
PLI
POP
PUSH
RBIT
REV
REV16
REVSH
ROR
RRX
RSB
SBC
SBFX
SDIV
SEV
SMLAL
SMULL
SSAT
STC
STMIA
STMDB
STR
STRB
STRBT
STRD
STREX
STREXB
STREXH
STRH
STRHT
STRT
SUB
SXTB
SXTH
TBB
TBH
TEQ
TST
UBFX
UDIV
UMLAL
UMULL
USAT
UXTB
UXTH
WFE
WFI
YIELD
IT
BKPT
BLX
ADC
ADD
ADR
BX
CPS
AND
ASR
B
DMB
BL
BIC
DSB
CMN
CMP
EOR
ISB
LDR
LDRB
LDM
MRS
LDRH
LDRSB
LDRSH
MSR
LSL
LSR
MOV
NOP
REV
MUL
MVN
ORR
REV16
REVSH
POP
PUSH
ROR
SEV
SXTB
RSB
SBC
STM
SXTH
UXTB
STR
STRB
STRH
UXTH
WFE
SUB
SVC
TST
WFI
YIELD
CORTEX-M0
CORTEX-M