ARM Overview

- Advanced RISC Machines (ARM)
    - Employs RISC-style architecture with some CISC-style features
    - 32-bit registers and addresses
    - Byte-addressable memory with 8-bit bytes, 16-bit halfwords and 32-bit words.
    - Enforces memory alignment for words and half words.
    - Supports both big-endian and little-endian memory storage order

ARM Overview

- RISC features
    - Fixed length 32-bit instructions
    - Load/store architecture
    - All arithmetic and logic instructions use only register operands

ARM Overview

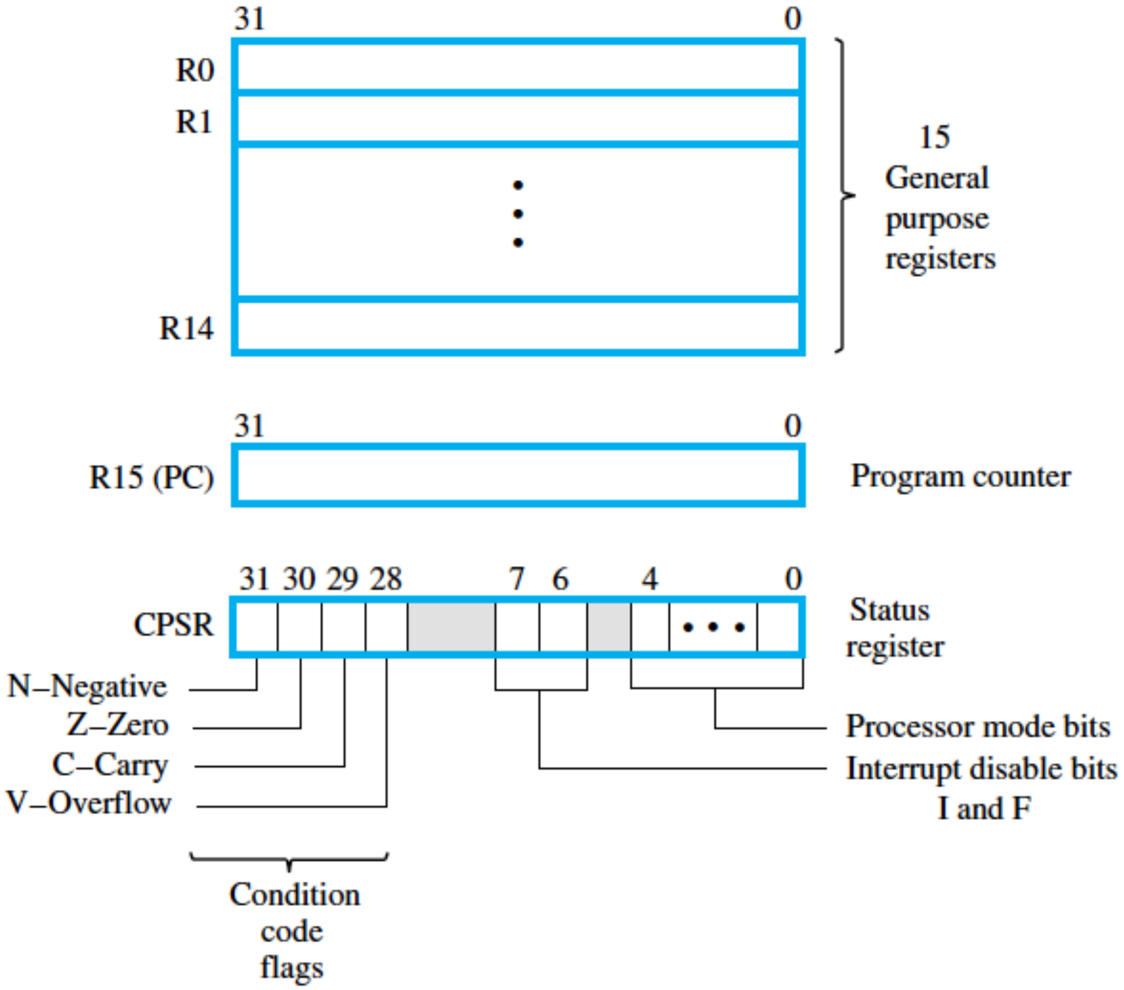
- CISC features
  - Supports autoincrement, autodecrement & PC-relative addressing modes
  - Condition codes (N,Z,V,C) are used for branching & conditional execution
  - Multiple registers can be loaded from or stored into a block of consecutive memory words

ARM Overview

- Unusual features
  - All instructions are conditionally executed
    - conditions specified by 4-bit field within each instruction
    - Can be set to always execute
    - Permits shorter routines than RISC machines using many branch instructions
  - No divide or explicit shift instructions
    - Division must be done in software
    - Immediate or register operands can be shifted by a prescribed amount before being used

ARM Overview

- 16 registers (R0 – R15)



ARM Overview

- R15 used as the program counter (PC)
- R14 is the subroutine linkage register
- R13 is the stack pointer
- "Banked registers" facilitate context switches
  - Additional copies of R0 – R14
  - Reduces the need to save and restore registers.

ARM Overview

- ARM addressing modes
  - Indexed addressing mode
  - Register mode
  - Immediate mode
  - Indirect mode
  - Absolute mode
  - Auto-increment & auto-decrement
  - PC relative mode

ARM Overview

- Other ARM features
  - Employs memory mapped I/O
    - Direct programmed controlled
    - Interrupt driven
  - Supports coprocessors
    - Floating point
    - Signal and video coprocessors
    - Exchange data between CPU and coprocessor registers
    - Memory transfers to/from coprocessor registers

ARM Overview

- Supports THUMB ISA
  - Used in low-cost & low-power embedded systems
    - Mobile devices such as phones
    - 16-bit machine instructions (reduced code space)
      - Fetched and expanded to 32 bits at run time
  - T bit within CPSR indicates when in Thumb mode
    - Programs can contain a mix of Thumb and standard routines.
    - Many Thumb instructions use a two-operand format
    - Conditional execution applies to all standard instructions but only to branches in Thumb set