Intel IA-32 Architecture

- Evolved from the x86 architecture
  - Uses 32-bit memory addresses
  - 32-bit registers
    - Supports 8-bit, 16-bit, 32-bit and 64-bit operands
  - Byte-addressable memory
    - Allows unaligned memory accesses
    - Employs little-endian memory storage order
  - Presents a CISC architecture to programmer
    - Breaks down CISC instructions into micro-ops
    - Micro-ops execute internally like RISC instructions

Intel IA-32 Architecture

- Memory Organization
  - Early x86 machines defined “word” as 16 bits
  - IA-32 “double-word” is a 32-bit item
  - “Quad-word” is a 64-bit item
  - Uses 32-bit memory addresses
  - Earlier x86 machines used 20-bit addresses
    - Derived from 16-bit segment registers and an offset
    - For code, stack and four data segments
  - IA-32 architecture supports flat address space
    - Maps all segments to a common 4 GB space
    - Shared by code, stack and data areas

Intel IA-32 Architecture

- IA-32 instructions vary in length
  - Unlike fixed-length instructions on RISC processors
  - Instruction encoding is complex
- Only 8 general purpose register
  - Can be treated as 8-bit or 16-bit parts
  - Maintains compatibility with older x86 instructions
- Supports CISC instructions
  - These do the work of an entire routine on RISC system
    - Such as string search and manipulation
  - Arithmetic/logic instructions can use memory operands
    - Not a load/store architecture

Intel IA-32 Architecture

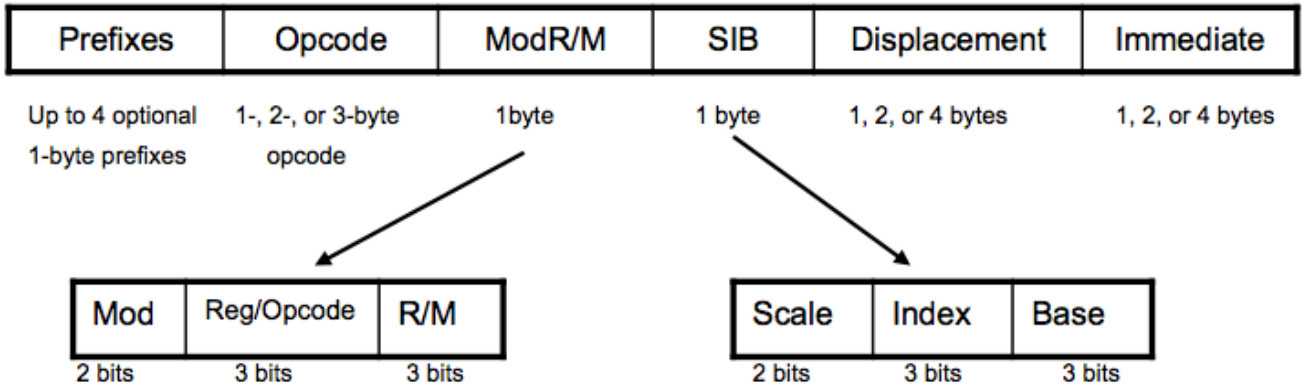
- Floating point unit uses separate set of 8 registers
  - Registers are organized as a stack ST(0) – ST(7)
  - Internal 80-bit extended precision float format is used
  - Converts IEEE 754 floats on the way to/from memory
- Supports SIMD vector type operations
  - MMX for integer operands
    - Multi-Media Extensions
  - SSE for floating point operands
    - Streaming SIMD Extensions

Intel IA-32 Architecture

- Performs I/O using special instructions
  - IN reads from an I/O device port
  - OUT sends data to an I/O device port
  - I/O ports are mapped to separate I/O space
  - Memory-mapped I/O can also be used
    - Using the normal memory access instructions
      - Like the only option for I/O on RISC systems

Intel IA-32 Architecture

- IA-32 Machine Instruction format is complex
  - Vary in size from 1 to 14 bytes
  - Complicates prefetching instructions



  - Opcode may be 1, 2 or 3 bytes
  - Other fields are optional and depend on the instruction type and the addressing mode used