Advanced Microprocessors

RISC & DSP
RISC Processors

RISC stands for Reduced Instruction Set Computer

Compared to CISC
- Simpler
- Faster
Why RISC?

- Complex instructions – even a trend to adding direct support for HLL operations!
- Many addressing modes, microprogrammed
- Long (10-12 CPU cycles) and variable execution times

- e.g. Late 1970s VAX: 304 instructions, 16 addressing modes, 11 instruction lengths!
Why RISC?

- Found that 20% of instructions executed for 80% of the time
- Simplify ➞ Speed up
- Catalysts:
  - Progress of optimising compiler technology
  - Faster, cheaper memory
  - ASIC technology instead of silicon fab equipment
RISC Properties

- Single word standard length of all instructions
  - Usually 32 bit to allow sufficient addressing of 3-operand instructions
- Small number of instructions, perhaps $\leq 128$
- Small number of addressing modes, perhaps $\leq 4$
- All of these properties
  - Simplify decoding and control logic
  - Encourage hardwired ALUs - faster
RISC Properties

- Single cycle execution for all (or at least 80%) instructions
  - Fast
  - Facilitates pipelining
- Memory access by load & store instructions only
  - Minimise number of instructions that must access memory during execution
  - Encourages single cycle instructions
RISC Properties

- All operations except load and store are register-to-register
- Relatively large number (at least 32) of general-purpose CPU registers
  - Sufficient for storing intermediate results
- Both properties minimise memory access
- Hardwired control unit
  - Faster than microprogrammed
Current RISC Designs

- Most current high-performance processors are RISC
  - Alpha
  - ARM
  - MIPS
  - PowerPC
  - SPARC
  - not Intel
- Most are 32 bit
- Most have 64+ registers
- Most have dual primary cache (fetch both branches)
- Most use instruction pipelining
- Many are 2-issue superscalar (two execution units)
Are Intel Processors CISC or RISC?

- Both – they are CISC with an underlying RISC core

See

DSP = Digital Signal Processor
Digital Signal Processing

- Sample one or more physical signals (A/D)
- Operate on the digital representation of the signal(s)
- Output a physical representation of the result(s) (D/A)

- Do this repeatedly, rapidly, and to meet hard timing constraints
Typical Applications

- Low cost embedded systems
  - Modem, mobile phone, HDD, automotive control
  - Price sensitive, high volume
- High-performance applications
  - Radar/sonar/seismic imaging, voice recognition
  - Often (DSP) multiprocessor
- Computer-based multimedia
  - Music synthesis, hi-fi audio, speech/audio/video compression and decompression
DSP vs. Analogue Circuits

DSP implementations may have advantages:

- Predictable, repeatable behaviour
  - Insensitive to environment – temperature, etc.
  - Insensitive to component tolerances
- Reprogrammability
- Component size advantages
  - e.g. 100μF capacitor larger than 10pF
DSP Metrics

- **Sample rate**
  - Sample rates required by applications range from < 1Hz to > 10 GHz

- **Clock rate**
  - Typically 100 MHz or faster

- **Numeric representation**
  - Floating point – larger dynamic range, easier to program
  - Fixed point – faster, but more complicated programming
Common DSP Features

- Fast multiply & accumulate (MAC) operation
- Multiple-access (per cycle) memory architecture
- Specialized addressing modes
- Specialized execution control
- Instruction pipelining
- On-chip peripherals for I/O
Single-cycle MAC

- Multiply and accumulate (MAC) operation in one instruction cycle
  - Most DSP algorithms (filtering, transforms, etc.) are multiply-intensive, involving "sum-of-products" terms
  - DSPs provide multiplier and accumulator hardware in the processor data path
  - Provide extra bits in the accumulator to guard against overflow of the accumulated result
Multiple-access Memory Architecture

- Ability to complete several memory accesses in a single cycle
  - Needed for high performance numerically-intensive algorithms
  - Multi-port on-chip memory
  - Multiple on-chip busses
  - Fetch instruction and operand simultaneously - Harvard architecture (separate instruction and data memory)
Specialized Addressing

- Dedicated address generators
- Once addressing registers loaded, address generation operates in the background
- Specialized addressing modes
  - Register indirect with post-increment to support operations on sequential data
  - Circular addressing to support operations on data buffers
  - Bit-reversed addressing to support FFT algorithm
Specialized Execution Control

- Many DSP algorithms involve looping
  - Supported by “repeat” instruction that allows looping without using cycles for testing the loop counter or jumping to the loop start
- May have fast context-switch, often using windowed registers
- May have low-latency interrupts
On-chip Peripherals

- All have on-chip serial and/or parallel interfaces
  - May support off-chip A/D and D/A
  - Fast synchronous serial (SPI) expansion is common
- All have a DMA mechanism
- Most have programmable timers
- Some have on-chip A/D and D/A
  - e.g. DSPs designed for modems, mobile phones
**DSP Embodiments**

- **Most are single-chip processors**
  - Package typically a QFP

- **DSP cores**
  - DSP vendor licenses the DSP design
  - DSP is fabricated on the same die as other circuitry to make a DSP-based ASIC
  - Alternative: customizable DSP
  - Expensive $\Rightarrow$ high volume only
DSP Embodiments

- Multi-chip modules
  - MCMs comprise multiple dies in a single package
  - e.g. TI has a MCM containing 2 DSPs and 128k SRAM

- Multi-processor ICs
  - e.g. Zilog and Motorola both have parts containing a microcontroller and a DSP
References


To retrieve PDFs, edit the links as per the following example (case sensitive):