Microprocessors

The MIPS Architecture
(Floating Point Instruction Set)
Mar 26th, 2002
The Notion of Coprocessor

A coprocessor is a separate processor that is intimately connected to the main processor and performs separate functions.

Originally was a real separate chip
- FPT coprocessor on MIPS
- 8087 coprocessor (fpt) on original PC

Now is usually on the same chip
- True of both the 80x86, Pentium and MIPS
Coprocessing on MIPS

The original idea was a general one

- Up to three coprocessors
- Specialized functions
- Well defined interface to processor
Coprocessor Instruction Set

Load word to coprocessor
Store word from coprocessor
Move to coprocessor
Move from coprocessor
Move control to coprocessor
Move control from coprocessor
Coprocessor operation
Branch on coprocessor true
Branch on coprocessor false
Coprocesors on the MIPS

There are only two of them

- System coprocessor
  - Specialized coprocessor instructions
- Floating-point coprocessor
  - Floating-point instructions
Floating-Point Formats

Two formats

- 32-bit
- 64-bit

Both these are IEEE formats

- Referring to the IEEE-754 standard
- We will deal with details of these formats and the operations required by this standard later.
Basic FPT Operations

Load and/store from/to the FPA registers
Moves between FPA and CPU registers
Computational operations
  • Add, subtract, multiply, divide
  • Convert to/from integer
Floating-point comparisons
FP Register Set

32 x 32-bit registers, FPR0-FPR31

- Each register can hold 32-bit fpt value
- Or two adjacent registers can hold 64-bit value (pair must be even/odd, e.g. FPR4/5)

Control/Status register

Implementation/Revision register
Control/Status Register

Condition bit from compare instructions
Exception indication bits
Trap enable bits
Sticky exception bits
Rounding mode
Rounding Modes

Round to nearest even
  • This is the normal default mode
Round/truncate towards zero
Round down towards minus infinity
Round up towards plus infinity
  • Last two can be used for interval arithmetic
Unbiased Rounding

Round to nearest exact number

For 0.5 case, round so that result is even

- Example for integers
- 2.5 rounds to 2
- 3.5 rounds to 4

This avoids bias in rounding
Traps

If traps are enabled
- Overflow causes a system trap

If traps are not enabled
- Generate infinities
- Infinities have “normal” arithmetic
  - $(+\text{infinity}) + (+\text{infinity}) \rightarrow (+\text{infinity})$
  - $(+\text{infinity}) + (-\text{infinity}) \rightarrow (\text{not-a-number})$
  - $1.0 / (-0.0) \rightarrow (-\text{infinity})$
FPT Load/Store

Load/Store from memory
- Single precision
  - Double precision requires two loads/stores
- Control word
- Revision control register (load only)

Register moves
- To/from CPU registers and FP registers
- To/from CPU registers and FP control reg
FPT Computational Instructions

Floating add/subtract/multiply/divide
Absolute value
Register move (between fpt regs)
Negate
Convert single to double and vice versa
Convert to integer format
FPT Comparison Instructions

Compare

- Compares 2 single or double fpt values

Possible tests are all combinations

- Normal comparisons
- Unordered/ordered comparisons
  - In unordered/ordered comparisons, all comparisons with NaN are false.
  - For example, unordered or less than or equal
    - If either operand is NaN result is False
    - Otherwise result is normal LE