CS 251 — LECTURE 9

Bartosz Antczak *Instructor: Stephen Mann* *January 31, 2017*

9.1 Operations on Numbers in Floating-Point Representation

Floating-Point Addition

Consider $9.54 \times 10^2 + 6.83 \times 10^1$ (assume we can only round to two digits). To add these two numbers:

1. Match the exponents ($9.54 \times 10^2 + .683 \times 10^2$)
2. Add significands, with sign: 10.223×10^2
3. Normalize: 1.0223×10^3
4. Check for exponent overflow/underflow
5. Round: 1.02×10^3

Floating-Point Multiplication

Consider $(9.54 \times 10^2) \times (6.83 \times 10^1)$ (assume we can only round to two digits). To add these two numbers:

1. Add exponents: $2 + 1 = 3$
2. Multiply significands: $9.54 \times 6.83 = 65.1582$
3. Normalize: 6.51582×10^4
4. Check for exponent overflow/underflow
5. Round: 6.52×10^4
6. Set sign

9.1.1 Accuracy of Floating-Point Numbers

The biggest problem with accuracy is a round-off error (e.g., using a calculator to disprove Fermat's last theorem). The result of an operation cannot be represented precisely, which means that the result must be rounded. **In this class, we'll round 1/2 up**.
For $n-$bit accuracy, we need to keep $n + 2$ bits during the computation.

9.2 Single-Cycle Processor Implementation

We will implement small subsets of MIPS operations, such as `lw`, `sw`, and `add`.

Instruction Format

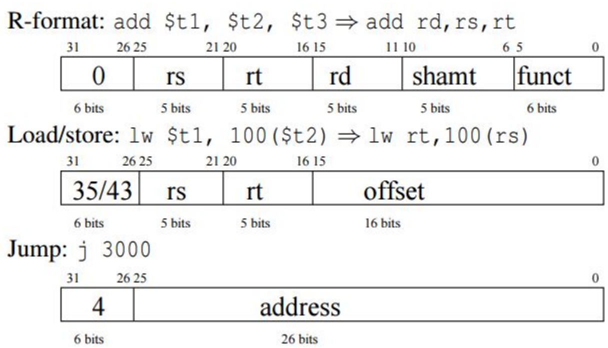

Figure 9.1: The 32-bit layout for each respective MIPS instruction. Courtesy of Prof. Mann's slides.