SIMPLE RISC PROCESSOR
FLOATING-POINT UNIT
v.1.0
5 may 2011

1. Requirements

Floating-point execution unit processes all floating-point instructions from the Simple RISC Processor ISA, namely ADDF and SUBF instructions.
The operands and result floating-point format complains with the floating-point data format defined in the Simple RISC Processor ISA.
Note: extreme cases (overflow and underflow) are not taken into account.

2. Microarchitecture

The microarchitecture of the floating-point execution unit for the Simple RISC Processor is sketched in Figure 1. It reflects the main steps of the floating-point addition algorithm:

1. operand alignment (the operand with the smaller exponent is right shifted with a number of positions equal to the difference between exponents);
2. operand complement, according to operands signs and the instruction type (addition or subtraction);
3. addition (the result may be one bit larger);
4. result mantissa complement, if the result is negative;
5. result mantissa normalisation (the mantissa is left shifted so that it's most significant nonzero bit (the implicit bit) is at the left of the msb, and the exponent is increased or decreased accordingly);

Prior to step 2 the implicit bit for both operands must be explicitly inserted into its proper place.
The normalization step mantissa shifting and exponent correction are done with a number that represents the distance between the most significant nonzero bit of the addition result (computed through a priority encoder) and the msb-1 bit of the addition result.

3. Timing

The microarchitecture must be split into several pipeline stages (three or four) in order to sustain the desired high clock frequency of the Simple RISC Processor.

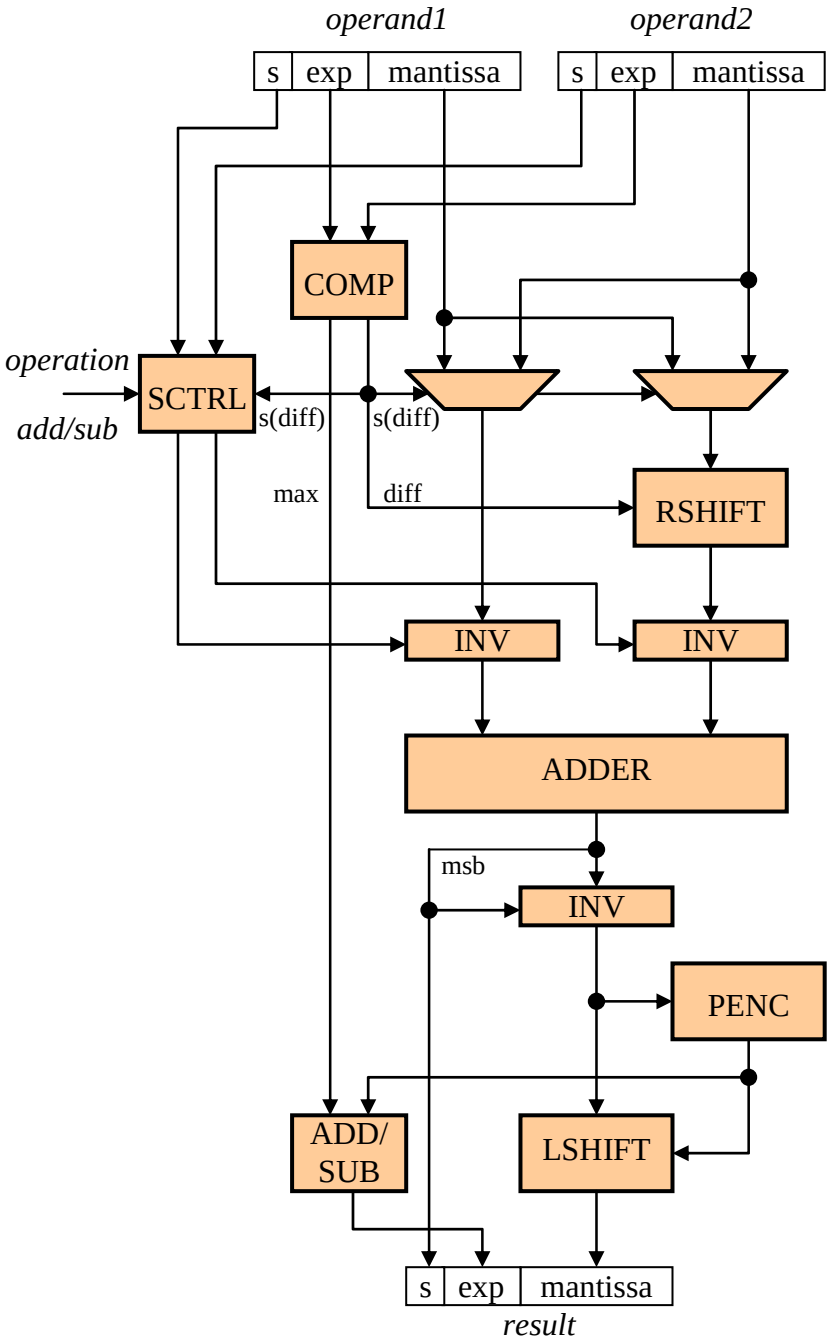

Figure 1
Simple RISC Processor floating-poit execution unit block diagram.
diff - exponent difference, s(diff) - the sign of exponent difference (indicates the greatest operand), max - the greatest exponent, msb - the sign of the addition result.