Floating point instruction decoder test plan

1. Purpose

Floating point instruction decoder will be tested for functional correctness when provided with valid and invalid inputs.

2. Test environment

The module under test is Floating-Point Instruction Decoder. Testbench will provide the inputs (clk, reset, instr) and check the outputs.

3. Coverage

The sanity check will include

- A test case for every possible instruction (FLW, FSW, FADD.S, and etc.)
- A test case for every possible rounding mode (RNE, RTZ, RDN, and etc.)
- Reset asserted high.
- Invalid instructions
- Edge cases

4. Test cases

For each instruction

- Input: 32-bit encoded instruction, clk, reset set to zero
- Expected outputs: fp_op, register indices, control signals, and etc.

For reset

- Input: 32-bit encoded instruction, clk, reset set to one
- Expected outputs: fp_op and func set to None, everything else set to zero.

For invalid instructions

- Input: 32-bit encoded instruction that does not refer to any valid instructions, clk, reset set to zero
- Expected outputs: fp_op and func set to None, everything else set to zero.

5. Pass/Fail criteria

Pass if the outputted values match the expected values. Fail if they do not.