

Snow64 Operand Manager Details

- General Notes
 - This module determines the operands that are used in the EX stage.
 - Note that this module is the one that actually performs operand forwarding.
 - Also, it can produce a stall in some cases.
- Ports of Snow64OperandManager
 - This module is clocked.
 - Inputs:
 - An enable signal. This module will not do anything unless this enable signal is high.
 - This enable signal should only be high during the first clock cycle that an instruction is in the EX stage.
 - Perhaps an easy way to do this is to just check if the program counter value in the EX stage has changed? This might not work for a single-instruction infinite loop....
 - Whether or not to perform type casting of register B and whether or not to perform type casting of register C.
 - These are two separate signals.
 - Not all instructions actually need the type casting of both registers. Some instructions only need register B to be casted, and some instructions only need register C to be casted. Still others potentially need both B and C to be casted, and others do not need either register B or C to be casted (mostly those that depend only on register A, such as the control flow instructions).
 - The LAR file's read ports (including all metadata, such as the data type and scalar data offset)
 - The EX stage's past output (and whether or not it's valid)
 - Outputs:
 - The vector values of data LARs a, b, and c.
 - True scalar values of data LARs a, b, and c.
 - Whether or not the EX stage should stall on the current cycle.
- Functionality
 - Operand forwarding.
 - Operand forwarding is simply performed on whole vectors of data.
 - This is accomplished in part by simply injecting scalar results into a vector after performing a scalar operation, which has to be done during or before the write-back pipeline stage anyway.