ECTE432 – Single Cycle MIPS Processor

The MIPS processor implemented was developed using a HDL language called Verilog. The component for a single cycle MIPS processor includes:

* PC – Program Counter is a flip flop whereby the component is triggered at the positive edge developed in Verilog. The output is an updated PC+4 or the branch address. Both input and output is 32 bits as the ISA in MIPS is 32 bits wide.
* Instruction Memory – Inputs the address value from the PC and fetches the instruction to operate. The output is concatenated in Verilog so it outputs directly the register addresses with 5 bits, immediate value 16bits, opcode and the function in 6 bits.
* Sign Extend - Converts the 16 bit immediate value from the instruction memory into a 32 bit value by assigning the first 16 bits to all 1s and the last 16 bits to the immediate value.
* Control – Inputs the opcode that determines the control lines for the muxes.
* ALU – Perform arithmetic calculations for the inputs of A and B. Selecting the operation using a control signal from the ALU control.
* ALU Control – Determines the operation to be carried out into the ALU.
* Data Memory – Stores and Reads the address outputted from the ALU and write data from the output of register file reg2.

All the above components were all created in Verilog files and instantiated into one top level MIPS file. Whereby each components is interconnected together using wires in Verilog. Therefore, in the Simulink model there is only one black box that contains the top level entity of the entire single cycle processor with outputs to determine the output using a the wavescope in Simulink.

The outputs that are made available includes the inputs for ALU of a and b to determine it is reading and executing the correct inputs. ALU output is included to demonstrate that the operation of the ALU is correctly working with the functionality of the type of instruction. The PC (Program Counter) to justify that it is updating (PC+4) and branching at the correct instances when branch and zero is both 1. The read register values in the register block to determine that the processor is reading the correct address and data. The Control signals of the regdst mux that determines that it is selecting the correct register it is writing to the correct address e.g. When it is an R-type instruction regdst is 1 to select the reegister address of rd at the bit locations of [15-11]. The alusrc control line to select the input for the alu b input to distinguish between an R-type and I-type instruction e.g. For an I type instruction the immediate value is used instead of the read reg value of rt. The branch control line to justify it is branching when the of ALU a and b are equal. Lastly, the regwrite control signal to demonstrate that the register file is writing back at correct address value.

For the simulation parameters the simulation time required is 100-200 to function the entire operation of the program from start to finish including the loops. To ensure that the black box is working double click on the black box and in the pop up make sure that the Simulation mode is in “ISE Simulator”. Other information required is to include all the outputs from the black box into the wavescope for debugging purposes.