University of Toronto, Faculty of Applied Science and Engineering
Department of Electrical and Computer Engineering

ECE 243S – Computer Organization – 2012

Simple Multi-Cycle Processor

Overview

This implementation of a simple multi-cycle processor consists of the datapath, control (FSM), and various inputs and outputs. The processor supports only fixed-length 8-bit instructions and data, and can only access memory one byte at a time. Overall, 10 instructions are implemented in hardware.

Functional Description

The inputs and outputs include:

Inputs:

- reset – clears the values of all registers and control signals, resets the condition flags, and resets the control FSM (KEY[0])
- clock – all writes and cycle transitions happen on the positive edge of the clock (KEY[1])
- HEXsel[1..0] – selects signals to be shown on the hex displays. For the DE2, use the multicycle.v as the top level module in Quartus II (which is the default). The hex displays will show either the four processor registers or the PC, IR and memory output registers, depending on the value of HEXsel. For the DE1, you may use the multicycle.bdf as the top level module in Quartus II. This reduces the number of registers that can be shown at once to two registers. This is due to the DE1 only having four hex displays. (SW[17..16])

Outputs:

- PCWrite – LEDR[17]
- AddrSel – LEDR[16]
- MemRead – LEDR[15]
- MemWrite – LEDR[14]
- IRLoad – LEDR[13]
- R1Sel – LEDR[12]
- MDRLoad – LEDR[11]
- R1R2Load – LEDR[10]
- ALU1 – LEDR[9]
- ALU2[2..0] – LEDR[8..6]
- ALUOp[2..0] – LEDR[5..3]
- ALUOutWrite – LEDR[2]
- RFWrite – LEDR[1]
- RegIn – LEDR[0]

- FlagWrite – LEDG[7]
- N (negative) – LEDG[1]
- Z (zero) – LEDG[0]
- Register Display
    - Displays 8-bit values of the four processor registers, PC, IR and memory output on HEX7 – HEX0
    - Each 8-bit value takes 2 of the hex displays, so not all registers can be shown at once
    - Note: as mentioned previously, different top level module may be selected depending on which board you are using, DE2 or DE1.

Instructions

- Addition (add)
- Subtraction (sub)
- NAND (nand)
- Shifting (shift)
- OR with Immediate (ori)
- Load (load)
- Store (store)
- Branch If Positive Zero (bpz)
- Branch If Zero (bz)
- Branch If Not Zero (bnz)