Computer System Design Lab
Design Experiment 5b: Computer System Design

Indian Institute of Technology Tirupati

Nov 5, 2019

1 Objective and Problem Statement:

The objective of this design experiment is to integrate all the hardware components such as memories: RAM and ROM, I/O devices: keyboard and display (default devices are the host system's I/O devices in which the simulator is run) and the processor to design a working computer system.

A computer system consists of multiple components which interact with each other to execute an instruction or a set of instructions. The design methodology that has been adopted in this course is modular in nature. All the components has been designed and tested individually and signed off as a working component. To finally realize a working computer system all these components need to be integrated and tested step by step.

* here the term *test* or *testing* indicate the simulation based verification.

2 Design and Verification

Following are the suggested steps to design and test the complete computer system.

1. Computer System Design: This is a step of integration of all the components one after another.

   (a) The ROM and RAM modules to be integrated with that of the processor module. The ROM module is for instruction and the RAM module is for data. The respective signals such as data bus/instruction bus, address bus, and read/write signals need to be connected accordingly. A new top-level CHIP need to be created for the computer system design. The name of the top CHIP be given uniquely. The direction of each signal be taken care off for functionally correct connection.

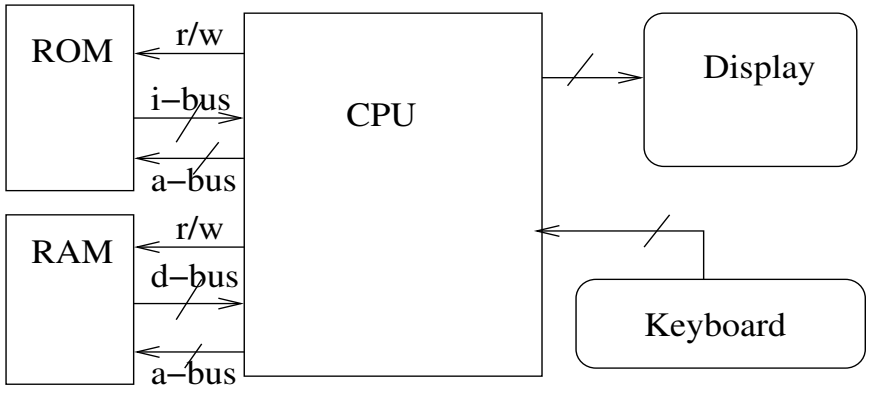

Figure 1: Block diagram of 16-bit computer system.

(b) Next, the I/O devices be connected. The connection of I/O devices: Display Screen and Keyboard is default in the simulator. The default devices are the screen and keyboard of host system.

Recheck: Kindly ensure that a reset to program counter has been implemented.

2. Design Verification: At least two type of verifications need to be performed. The first type of verification requires to verify the execution of all the 28 instructions. To carry out this particular verification, **prepare a test bench file which contains all the instructions of this mystical processor** and simulate it by first loading the program into ROM and then performing step-wise execution. As the instructions get executed the **content of the registers and data memory be recorded in a log file**.

   Second, a **four different programs** be prepared to test each of the components separately, instruction memory be tested with branch instructions as well as a branch instruction which jump to location out of range. Similarly, the data memory (RAM) be tested with large size array and stack. The display and keyboard be tested with separate program. You may reuse some of the program which has been written as part of assembly program exercise.

3 Experimental Flow

1. Write a top CHIP module that integrates all the required modules: memories and processor.
2. Load the program into your ROM (not the simulator's ROM). The loading of the program to ROM can be done via test bench file in a similar way it was done while testing the ROM chip. To facilitate this an access to the ROM be provided from the primary inputs.
3. Once the program is loaded, reset the program counter (PC) to point to the first instruction and start execution from the beginning.
4. As the program executes record the value of registers and final output of the program.
5. Cross check your final as well as intermediate output with the expected output. If it does not matches there could be two possibilities 1) the hardware design in wrong, 2) the program written is wrong. Cross check the correctness of your program by running on the bare simulator. If the it is correct then go for cross checking your BHDL code of top CHIP.

4 Tools:

- Language: The Nand2Tetris HDL and TSL (test scripting language)
  Refer: Appendix A and B of text book.
- Tools: Hardware Simulator of Nand2Tetris.
  `https://www.nand2tetris.org/software`
- Machine and OS: x86_64 machines with any distribution of Linux (Ubuntu or CentOS).

5 Reporting and Evaluation

Prepare a report which contains the high level view of your BHDL module organisation, block diagram of the design, the test programs and corresponding log files also be included in the report. A final reported be submitted at completion of last experiment.

The evaluation will be based on execution of correct design and execution of all the five test programs. The execution of the first program which consists of all the instuction will carry maximum weightage. Out of remaining four programs, the program which test ROM and RAM will be given next highest weightage. The program that test screen and keyboard be given next points.