TinkerSim: A Tiny Processor That Can

Introduction

In this project, you will implement in C an instruction-level simulator that can take an object code that was produced in the previous assignment and executes it while simulating the processor.

Simulation

The simulation reads the 4 bytes at the address pointed to by the program counter (PC). These bytes contain the instruction to be simulated. A plausible design then parses the instruction and use the opcode field as an index into an array of functions, where each function implements an instruction.

Example

Consider the instruction:

and r_d, r_s, r_t

This can plausibly be done as follows:

Step 1: Looks up the opcode, jump through the function array into the function that simulates the add instructions.

Step 2: Read the values of r_s and $r_{t,}$ performs the and operation.

Step 3: Store the result in r_d

Step 4: Advance the program counter by 4 and continue.

The program counter is initialized to 0x1000. It should be incremented by 4 after every instruction, except for control instructions where jumps occur out of sequence. The simulation should terminate when the halt instruction is found.

Software Engineering and Logistics

The project may appear intimidating but once you understand the overall picture, it is a very mechanical implementation effort. Tenets of software engineering such as modularity in design, defensive programming, regression testing, and a project plan will put some order and reduce stress.

Happy coding, simulating, and testing.