CSC 350 Project 2 Mid-Project Report

Spring 2019

Group Members

Trevor Harness	Akash Charitar	Alwien Dippenaar	Erik Yu
V00867541	V00875728	V00849850	V00865663
tharness@uvic.ca	akashcharitar@uvic.ca	alwiend@uvic.ca	hanxiaoyu@uvic.ca

The three main deliverables of our project are the emulator itself, the demo programs, and the report. This mid-project report will cover the progress, problems, and proposed changes if required for each component.

Progress

The emulator consists of the user interface, the state elements, the emulation stages, and user feedback. The interface is complete, allowing invocation of the emulator from the command line and specification of an assembled program to run. The state elements, including RAM, ROM, and registers, have been implemented. The memory system of the Intel 4004 CPU is complex: the full RAM configuration is organized into eight banks each containing four chips that are themselves comprised of four registers, so for the purposes of this project programs are emulated with a single bank of memory containing a single chip. This totals 80 accessible memory locations of 4 bits each. The emulation stages are the fetch, decode, and execute stages. The instructions currently implemented are:

- LDM
- FIM
- SRC
- LD
- ADD
- XCH
- NOP
- FIN
- CLB
- WRM
- WR# (# = {0, 1, 2, 3})
- ISZ
- INC

This list is enough to successfully emulate our first demo program. To terminate execution of a program, we decided to use the KBP instruction. Since we need a way to tell the emulator that a program is finished and KBP is only used very infrequently (and never for our demos), this instruction will

immediately cause the emulator to halt. Upon completion of a program, feedback is provided through the dumping of register and memory contents.

To date, we have completed one demo program. This demo places the value 0xF into every available cell of RAM. The code demonstrates that our emulator is capable of register loads, writing to memory, and conditional jumps.

The structure of the report has been written. It will introduce the project, give an overview of the Intel 4004 architecture, explain the design of the emulator, and show demo assembly and the results of emulating it.

Problems

The emulator implementation has not been much trouble. Besides adding more instructions, the next major piece of work is implementing a two-stage pipeline as per the original proposal comments. This task has been assigned and work will continue throughout the week.

There have not been any significant problems with creating demo programs, thanks to freely available online assemblers and emulators for the 4004. We are using these utilities to test the code itself and verify the behavior of our emulator.

Because the emulator, demo programs, and project report are interdependent, it has been difficult to synchronize work. For example, the design of the emulator cannot be explained until it has been designed, and the results of emulating a program cannot be shown until both the program and the emulator have been written. We are using Slack to communicate, but the nature of the project has made it difficult to convey what is needed from other members and when.

Proposed Changes

We intend to maintain the scope of the project in terms of deliverables and specifically the emulation of assembled 4004 assembly code. However, since this is not a production emulator and in the interest of balancing time, we propose restricting the breadth of the project to the emulation of our demo programs. With this change, our emulator will still demonstrate its capabilities on the variety of instructions in the demo programs but will not implement every instruction of the 4004.