Fall 2022 CEG-4110-10 CEG-6110-10

Introduction to Software Engineering

Class Project Overview

# Project Title:

* Intellec 8 Mod 80 Emulator

# Team Name:

* Class Example

# Team Members:

1. Me,
2. Myself,
3. I.

# Project Synopsis:

Emulate an Intel 8080 microprocess to execute the Monitor software in Intellec 8 Mod 80 Intel development system.

# Background:

In the early 1970’s Intel corporation built a development system for the 8-bit 8008 microprocessor called the Intellec 8. Intel went on to design and released the more powerful 8080 microprocessor. Intel then updated the development system with a new processor board and program to support the development of new systems containing Intel 8080 microprocessors. The “Monitor” program was the only software that was installed in the system. The “Monitor” software was not an operating system but a small set of driver assembly level subroutines.

# Concept of Operation:

The Intellec 8 Mod 80 emulator will provide an environment for the execution of the Monitor software that resides in EPROMs installed in the original hardware.

The Intellec 8 Mod 80 hardware environment included:

* 4 K Random Access Memory (RAM),
* 2 K EPROM (8 x 256 bytes)
* Intel 8080 Microprocessor
* 31 Front panel switches to control the operation of hardware, including:
  + Control
    - RESET
    - INT
    - DEP AT HLT
    - DEP
    - STEP/CONT
  + Mode
    - WAIT
    - SRCH-WAIT
    - MEM ACCESS
    - I/O ACCESS
    - SENSE
  + Address Control
    - LOAD
    - INCR
    - DECR
    - LOAD PASS
  + Address/Data (Bits 15-8)
    - MEM ADDRESS HIGH / I/O ADDRESS / SENSE DATA
  + Address/Instruction/Data (Bits 7-0)
    - MEM ADDRESS LOW / INT INST / DATA / PASS COUNT
* Front panel display is composed of 48 LED’s
* Serial Interface to a teletype printer with paper tape reader and punch

The emulator will run in a terminal window on a host processor as the main user interface to control the emulator. The emulator commands and responses will be displayed in the terminal window. The input and output of the Monitor program will also be performed in the terminal window using a different prompt for emulator versus Monitor program output.

The emulator will also accept inputs and outputs to and from the terminal and transfer that I/O to and from the emulated software. The emulator will also emulate the memory configuration of the Intellec 8 Mod 80 hardware.

# Estimation of Effort:

## Task Description (TD)

Requirements, Design, Code & Unit Test (CUT), Integration and Test (I&T) anticipated components that make up the Intellec 8 Mod 80 Computer Software Configuration Item (CSCI). The following identifies the anticipated components:

1. Keyboard input processing
   1. Emulator control
   2. Front panel input (switches)
   3. Emulated teletype input
2. Display output processing
   1. Emulator status
   2. Front panel output (LEDs)
   3. Emulated teletype output
3. Memory emulation
   1. RAM
   2. EPROM (Including loading of the monitor program)
4. Processor register emulation
5. Instruction emulation:
   1. 245 Instructions (10 possible instructions not implemented)

Perform verification and validation testing for the emulator.

## Basis Of Estimate (BOE)

|  |  |  |  |
| --- | --- | --- | --- |
| Component | SLOC | Multiplication Factor | Total SLOC |
| Keyboard Input |  |  |  |
| Emulator Control | 100 | 1 | 100 |
| Front Panel Switches | 10 | 31 | 310 |
| Teletype Input | 20 | 1 | 20 |
| Display Output |  |  |  |
| Emulator Status | 50 | 1 | 50 |
| Front Panel LEDs | 10 | 48 | 480 |
| Teletype Output | 40 | 1 | 40 |
| Memory |  |  |  |
| RAM | 50 | 1 | 50 |
| EPROM | 50 | 1 | 50 |
| Registers | 50 | 1 | 50 |
| Instructions | 245 | 10 | 2450 |
|  |  | Total Code | 3600 |
| Productivity Factor | 20 SLOC/Hour | 180 hours |  |

## Assumptions:

1. 20 SLOC per hour productivity factor.
2. Unit Test code count is included in the estimate, but the time to develop unit test is included in the productivity factor.
3. Integration and Test time is included in the 20 SLOC per hour productivity factor.
4. Minimal design documentation is anticipated and included in the 20 SLOC estimate. The following identifies potential design documentation:
   1. UML diagrams showing code the anticipated design
   2. Interface definitions
   3. State machine diagram
5. The code and any documentation will be reviewed only by the developer.
6. The code will be developed and tested on a MacBookPro using Xcode Integrated Development Environment.

# Extra Information