

ECE 662

Machine Problem #2

Write microinstructions to implement ADD and CLR for all addressing modes. The HALT must also work. Test thoroughly! The Condition Codes must work properly.

You may NOT use the conditions ir3, ir2, ir1, or ir0 for this problem.

When you are done, make a copy of your microcode file and call it `specs.dat`. Be sure that you and your partner's names are in the opening comments in the file. Each person should submit their `specs.dat` file by uploading it into the Machine Problem 2 dropbox on Carmen by the due date. Do not submit the `details.1` file. Tests will be run on your `specs.dat` file to verify its correct operation. Keep a copy of the microcode file as you submitted it. Remember, your team must do its own work without collaboration with any other team (present or past).

Suggested approach:

1. After a fetch cycle, determine if the instruction is a
 - (a) Branch or special instruction,
 - (b) Single operand instruction, or
 - (c) Double operand instruction.
2. For any single operand instruction:
 - (a) Place the `dst` operand in a designated temporary register for all the addressing modes, and
 - (b) Place the effective address of the `dst`, except for addressing mode 0, in another designated temporary register.
3. For any double operand instruction:
 - (a) Place the `src` operand in a designated register, and
 - (b) Place the `dst` operand and effective address in temporary registers, as you did for single operand instructions.
4. Test for the instruction opcodes and execute the individual instructions. Note that the register addressing mode (0) is unique, when used for the `dst`, in that it does not involve a memory write.
5. Permit any addressing mode for the single and double operand instructions, but test only those allowed (pg. 6 of "Description of OSIAC 662"). This will simplify the programming.