Max Sanchez

Annemarie Bell

Y-86 Report

The goal behind this project was to complete and create a microprocessor that goes through the 3 stages of a microprocessor, Fetch/Decode, Execute and Memory. Below is a description of the following designs for each of the pieces in the unit and what they are meant to do.

Caption: these are the main functions we use for the assembly

The Fetch Unit:

The goal behind the fetch unit is to take the instruction that has been based of the binary conversion of the c code and create a PC output based on the logic.

- The fetch unit grabs up to 6 bytes of address using the PC
- Updates the PC based on the current instruction:

Current instruction	Byte size
Halt	1
Nop	1
Rrmov	2
irmove	6
Rmov	6
Mrmov	6
OPL	2
Jxx	5

- The PC is then updated by the number of Bytes used in the instruction
    - PC <- PC + len(INS) {In our case we are aware that the lengths can be 1,2,5,6}
- In our case the logism shows icode, ifun, rA,rB, Val c, Val P and new PC

The Register File:

- For the register we are creating a 8x32 register, this includes:
    - 2 read and write ports
    - 2 decoders
    - And tri state buffers
- The register is going to take in values that are affected by the table below:
    - RA: rrmovl, rmovl, and opl
    - RB: removal, irmovl, rmmovl, opl

The ALU:

The ALU is the arithmetic portion of the processor, it takes the function if an OPL or Jump function and follows the logical order of said function to produce the data that is wanted.

Data Memory:

The data memory portion of this circuit holds the memory and updates it according to the instructions given to it. The memory will be stored or written back to the location depending on the instruction and what it follows.

New PC:

This is the function when the PC is completely updated and the cycle of the processor for that clock is complete, when this is finished the processor moves on to the next set of instructions.