4.1.1

Branch = 0 # there is no branch instruction here so the value is 0
Mux in the middle = ALU # the mux box in the middle is connected to ALU box
ALU operation = AND # the operation is AND operation
MemWrite = 0 # there is no data in the registers so the MemWrite and MemRead are 0.
MemRead = 0 # same
Mux in the bottom = second input from registers # for read and store
RegWrite = 1 # the signal is true after the output written to the register

4.1.2

Registers # instruction data and values are stored and loaded in this registers
Instruction memory # instructions are stored in the instruction memory
Program counter # the instructions use pc
ALU # ALU perform the AND operation

4.1.3

Produce outputs but not used:
ALU # it produced a zero signal but the AND instruction is not a branch
instruction so it is not used.
Produce no outputs:
Data Memory # there is no data read or stored in this instruction so there is no
output for this block

4.7.1

The 0-15 bits is 0000 0000 0001 0100
The sign-extend is 0000 0000 0000 0000 0000 0000 0001 0100
In the jump instruction, the 26 bits expanded to 28 bits with left shift
So the shift left 2 is 0001 1000 1000 0000 0000 0101 0000

4.7.2

In ALU instruction the input takes the lower 6 bits(0-5) so it is 010100
The main instruction is the first 6 bits(26-31) so it is 101011
The instruction is sw so 00 is input to ALU control.

4.7.3

The new PC address will be PC+4
PC->Add->Mux->jump Mux->PC

4.7.4

For main Mux, since sw is not branch instruction, the output is PC+4
For Mux before the Write register, the output is instruction[20-16] or [15-11] so it is 00010 or 0000
For Mux before ALU, the output is instruction [15-0] so it is 0000 0000 0001 0100 = 20

4.7.5
ALU = -3 and 20
Add = PC and 4
Add(ALU result) = PC+4 and 20x4 = 80

4.7.6
Read Register 1 = instruction[25-21] = 00011
Read Register 2 = instruction[20-16] = 00010
Write Register = instruction[20-16] or [15-11] = 00010 or 00000
Write Data = 0