Senni Tan 400196392 T02

1) op0 + op1 · I[31] · (¬I[30]) · I[29] · (¬I[24])
2) I[31] · I[30] · I[29] · I[28] · I[27] · (¬I[26]) · (¬I[25]) · (¬I[24]) · (¬I[23]) · (¬I[22]) · (¬I[21])
3) LDUR X2, [X1, #23]
    a. Reg2Loc: X
       Branch: 0
       MemRead: 1
       MemtoReg: 1
       ALUop: 00
       MemWrite: 0
       ALUSrc: 1
       RegWrite: 1
    b. Read Register: 00001
       Write Register: 00010
    c. 0x 0000 0000 0000 0017
    d. Data Memory, Registers, Instruction Memory, Sign-extend, The Mux and The ALUSrc behind it, ALU result.
    e. Branch unit produces no output for this instruction; no resources produce an output that is not used.
    f. Program Counter + 4
4) 4.4
    a. 4.4.1 LDUR
    b. 4.4.2 LDUR, STUR
    c. 4.4.3 STUR, CBZ
5) 4.12
    a. 4.12.1
       No new function block needed, for swap operation, we need to load and store data so no new function block is needed.
    b. 4.12.2
       No modifications needed. The swap can use the original instructions for load and store.
    c. 4.12.3
       No new data paths needed. The swap can use the original instructions for load and store, therefore it can use the original data path, no new data path is needed.
    d. 4.12.4
       No new control signals are needed. It can use those signals that already exists.
6) 4.16

a. 4.16.1
   Pipelined: 350ps
   Non-pipelining: 250 + 350 + 150 + 300 + 200 = 1250ps
b. 4.16.2
   Pipelined: 5 * 350 = 1750ps
   Non-pipelining: 1250ps
c. 4.16.3
   I will split up the longest stage: ID 350ps
   The new longest stage will be: MEM 300ps
d. 4.16.4
   LDUR + STUR = 20% + 15% = 35%
e. 4.16.5
   LDUR + ALU/Logic = 20% + 45% = 65%

7) 4.20

```
ADDI X1, X2, #5
NOP
NOP
NOP
ADD X3, X1, X2
ADDI X4, X1, #15
ADD X5, X3, X2
```