

Computer Architecture Lab5

Lab Assignment 5

Jianyu Huang (UT EID: **jh57266**)

jianyu@cs.utexas.edu

General Idea

1. Save the MAR into the temporary register VA
2. Load the MAR with the address of the PTE of the page containing the VA:
$$\text{MAR}[15:8] \leftarrow \text{PTBR}[15:8]$$
$$\text{MAR}[7:0] \leftarrow \text{LSHF}(\text{VA}[15:9], 1)$$

i.e., $\text{MAR} \leftarrow \text{PTBR} + (2 \times \text{page_number})$.
3. Read the PTE of the page containing the VA into the MDR
4. Check for a protection exception/a page fault
5. Set the reference bit of the PTE. If the pending access is a write, set the modified bit of the PTE
6. Write the PTE back to memory
7. Load the physical address into the MAR:
$$\text{MAR}[13:9] \leftarrow \text{PFN}$$
$$\text{MAR}[8:0] \leftarrow \text{VA}[8:0]$$

i.e. $\text{MAR} \leftarrow \text{PFN} + \text{BIP}$
8. If the operation is a write, load the MDR with the source register

State Diagram Modifications

Interruption/Exception Handler State Diagram, RTI State Diagram, Virtual to Physical Address Translation State Diagram are on Page 3.

The changes of the original state diagrams are shown on Page 4.

Data Path Modifications and New Control Signals

The changes of the original data path and new control signals are shown on Page 5.

I mainly add the following Control Signals comparing to Lab 4:

V2P/1: mark a virtual to physical address translation, set in State 53, 51, 52, 61, 62.

LD.VA/1: Load VA from MAR if it is assert.

LD.J/1: Load J from TEMP_J if it is assert, which means to come back to the original state after virtual to physical address translation.

GatePTBR/1: Load PTE address onto the bus

GateVA/1: Load calculated physical address onto the bus

Microsequencer Modifications

The changes of the original microsequencer are shown on Page 6. I try to make the changes to microsequencer as little as possible and avoid adding new mux on top of the original microsequencer so that the critical path will not be added.

When the [E] bit is set to 1, which means there is an exception detected in the previous state, the next state address will be set to State 49 by the microsequencer, which is the entry of exception handler. When COND2 is set to 1 and [I] is one, which means there is an interruption in the previous instruction, the program will go to State 49, too. If neither of the above two cases are true and [V2P] bit is set to 1, State 54 will be the next state, since [V2P] means the program needs a Virtual to Physical Address Translation. In all other cases, the microsequencer will be the same as normal LC3b microsequencer like Lab3.

I only add another control signal to the same mux and some combinatorial logic to J[4], so the critical path isn't affected too much.

Technical Details

1. When to reset [E]

Reset_E should be in the same state of [E],

If Resetting E is put on State 49, then there is an issue here: because we first evaluate microsequencer to get the address of the next state before E is reset to 0, we will go to State 49 twice.

2. When to reset [I]

[I] should be reset in the end of interruption handler.

State 43 we need to check whether it is INTV/EXCV. If we have both INTV/EXCV not equal to 0, the only way we can check is through checking [I], so we cannot reset [I] before this state. However, if we can reset EXCV/INTV each time we get out State 43, we may not need to do this way. We only need to check whether EXCV/INTV is 0 or not.

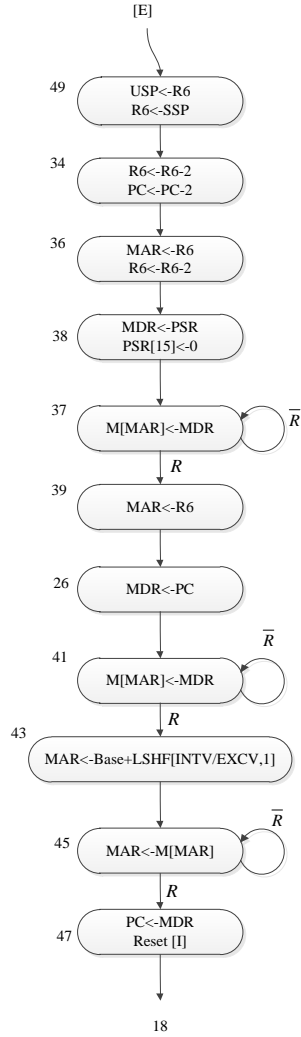
3. COND2 is set in State 53 instead of State 18/19

Because only in State 53, State 33 is the next State. We have to choose State 33 or State 49 in the state which has COND2.

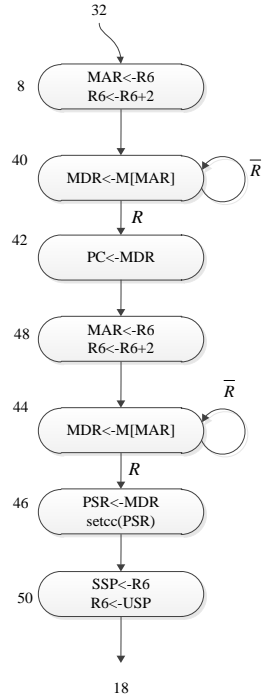
4. Check [E] in extra new State 60/61 instead of State 23/24 like Lab4

In the Virtual to Physical Address Translation microinstructions, MDR is changed, so MDR in State 16/17 will be consistent with MDR in State 23/24 if we add check [E] in State 23/24.

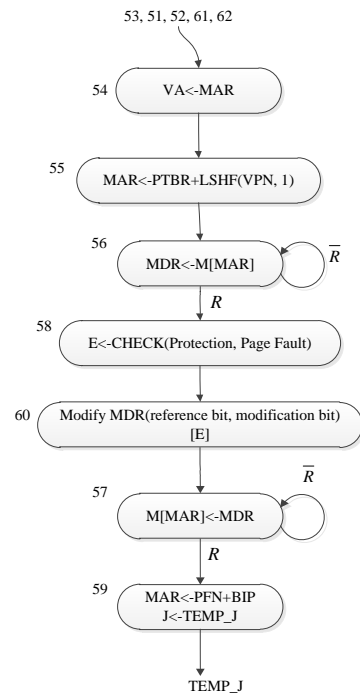
Interruption/Exception Handler



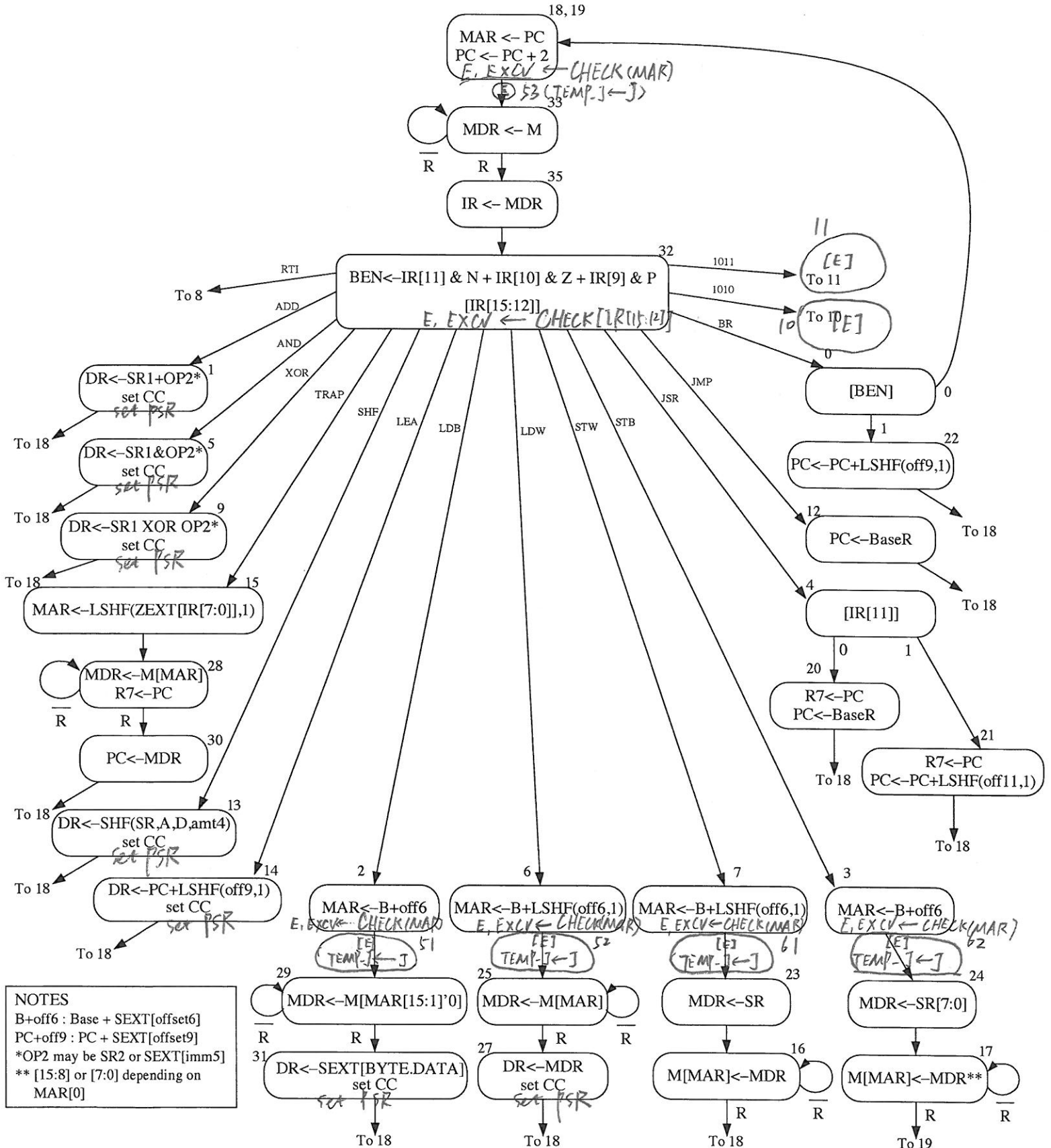
RTI



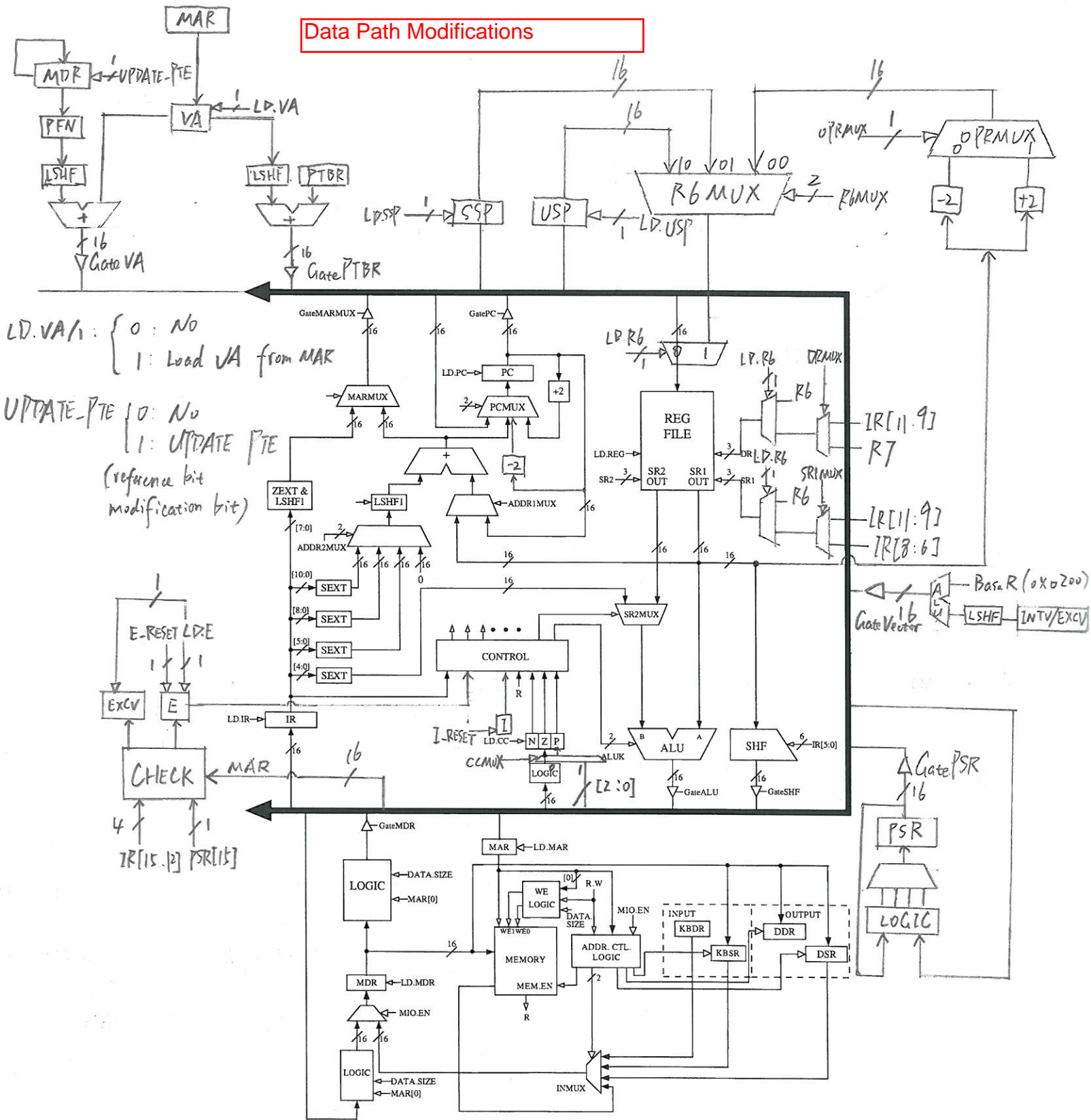
Virtual to Physical Translation



State Diagram Modifications



Data Path Modifications



Control Signals:

LD.R6/1: { 0: Normal operations
1: DR=R6 or SRI=R6

R6 MUX: { 00: select R6-z/R6+z

OPR MUX { 0: select R6-z
1: select R6+z

LD.SSP/1: { 0: No
1: LOAD

LD.USP/1: { 0: No
1: LOAD

CC MUX: { 0: Normal CC
1: CC from PSR[2:0]

PSR MUX: { 00: BUS
01: PSR[15] ← 0
10: change CC in PSR

PC MUX/2: { 00: PC+2
01: BUS
10: ADDR
11: PC-Z

L-RESET: { 0: NO
1: RESET

E-RESET: { 0: NO
1: RESET

Microsequencer Modifications

