Design of a Pipelined 3-Element Adder

Lab 7 P1 + P2

Introduction

10/16/2012

Our focus of this discussion is design and paper submission.
Verilog coding and simulation is not discussed.

So skip pages 3, 4, 8, & 9 of the 12-page document.

($R) <= ($Z) + ($Y) + ($X) if there is no overflow.

4-bit Reg IDs because it is a

16 x 16 IFRF

Read PORTS

Write Port

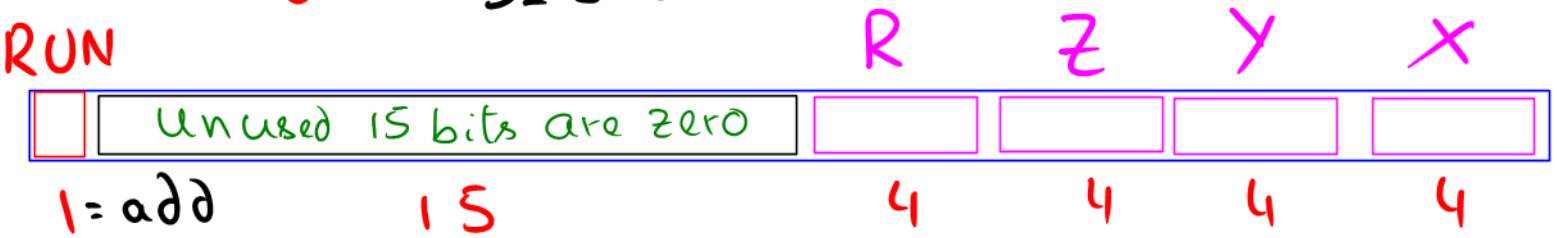
Initial contents of the Register File

Register id		Content	Register id		Content
Register 00	0	0001h	Register 08	8	0100h
Register 01	1	0002h	Register 09	9	0200h
Register 02	2	0004h	Register 10	A	0400h
Register 03	3	0008h	Register 11	B	0800h
Register 04	4	0010h	Register 12	C	1000h
Register 05	5	0020h	Register 13	D	2000h
Register 06	6	0040h	Register 14	E	FFF8h
Register 07	7	0080h	Register 15	F	FFFFh

$0 is like any other register.

Instr. format

32 bits




```
Add $4, $3, $2, $1 ; Translation: 80004321 (Hex)
```

Dependent instruction

can only get help from

the senior in 

Senior is in a position to help a junior only when the Senior is in 

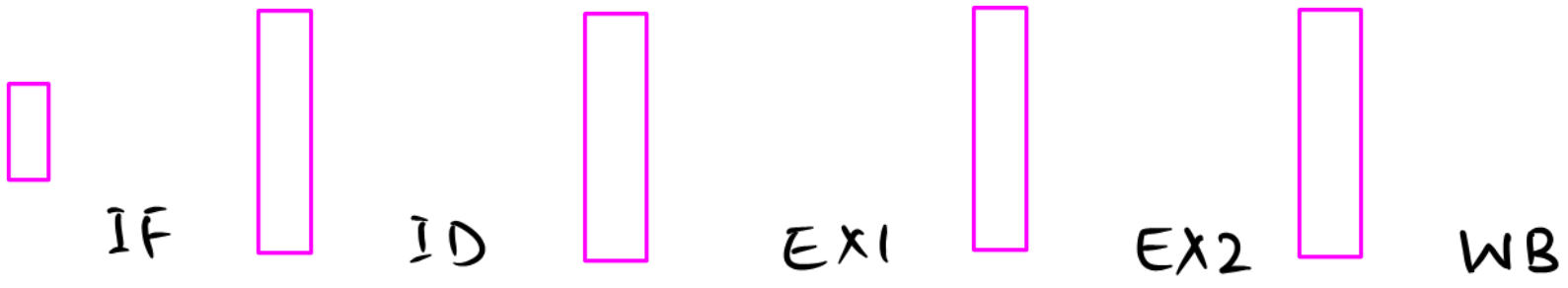
Dependent junior can only

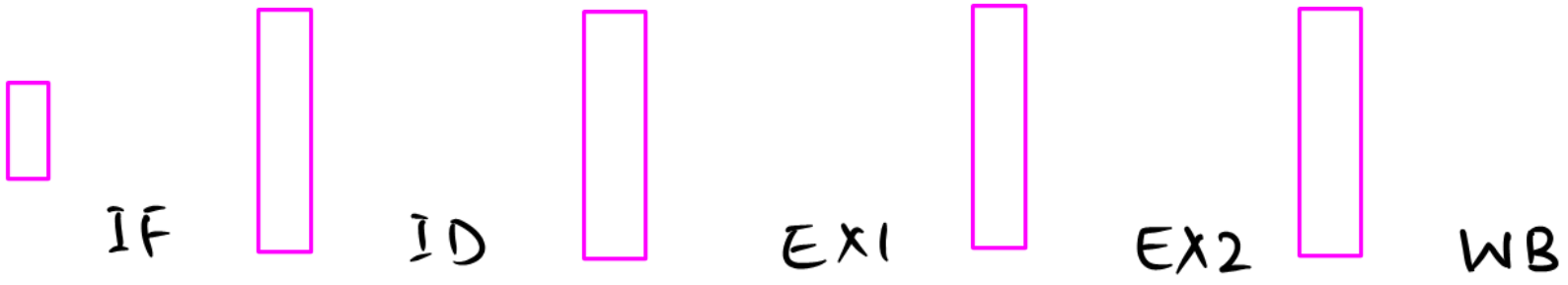
Stall in 

(a) Text book 5-stage pipeline

(b) This Special pipeline

What if there is only one Source register (like in Lab 7 P3)? Can the dependent Instr. Stall in EX1?

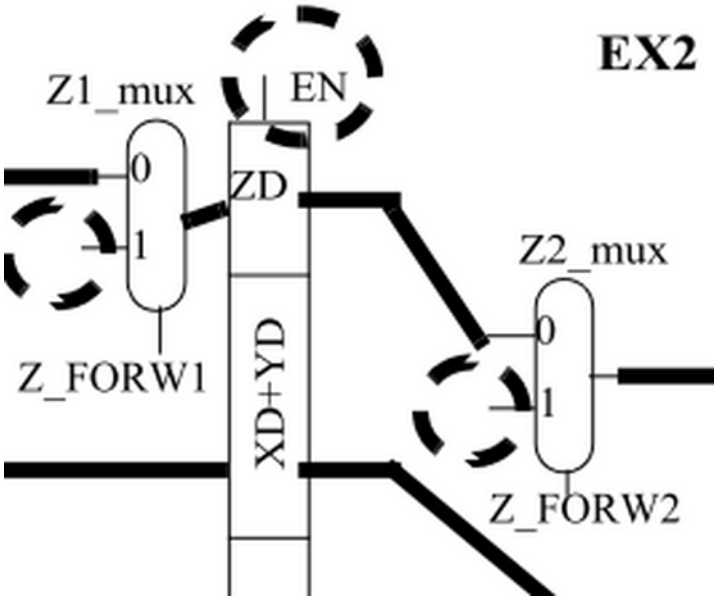


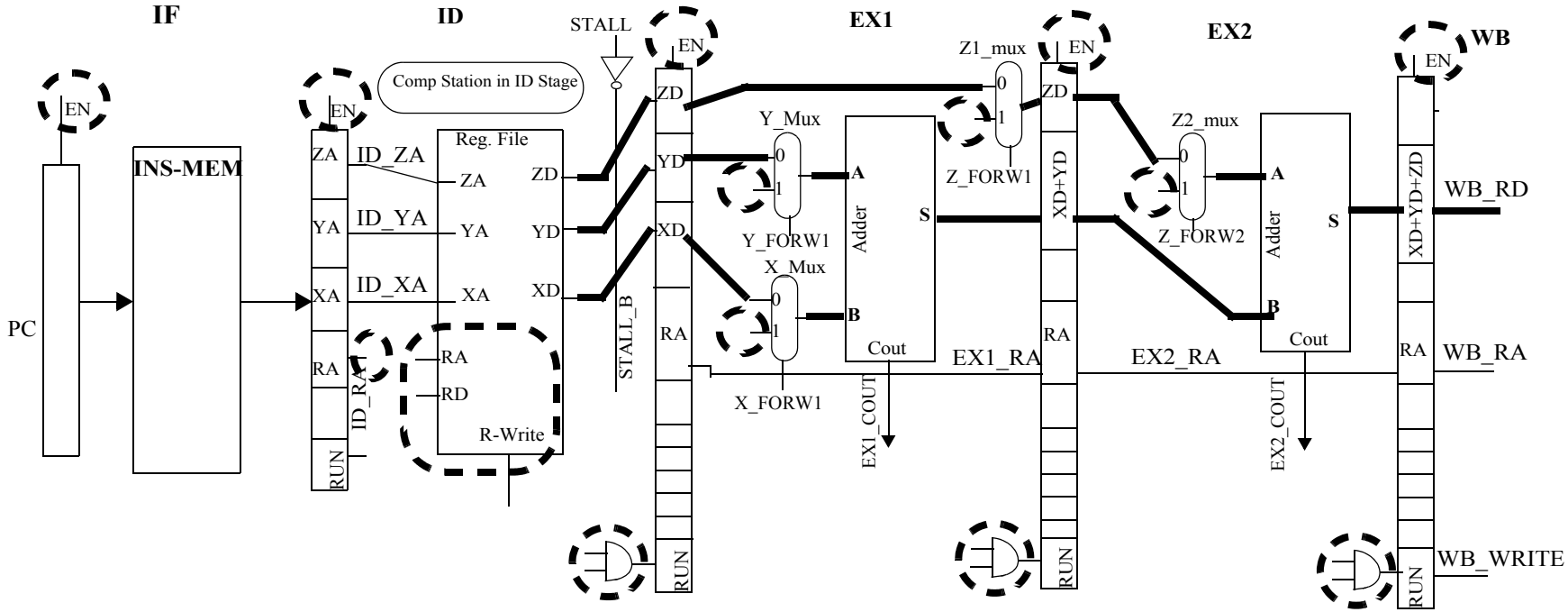


Question from Lab 7 P3

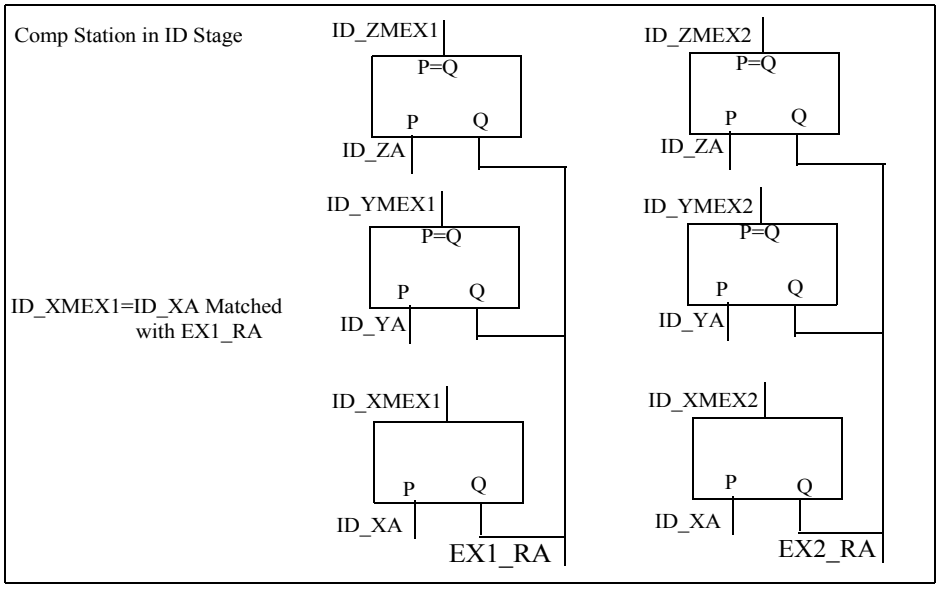
_____ (Unlike/Like) the MIPS 5 stage pipeline, where the instructions _____ (have only one source register / can have two source registers), here the instructions _____ (have only one source register / can have two source registers). Hence it _____ (is / isn't) possible to stall the dependent instruction in EX1 stage instead of the ID stage.

Q 1.7 2 of 2 to 1 muxes ⇒ 1 of 3 to 1 mux





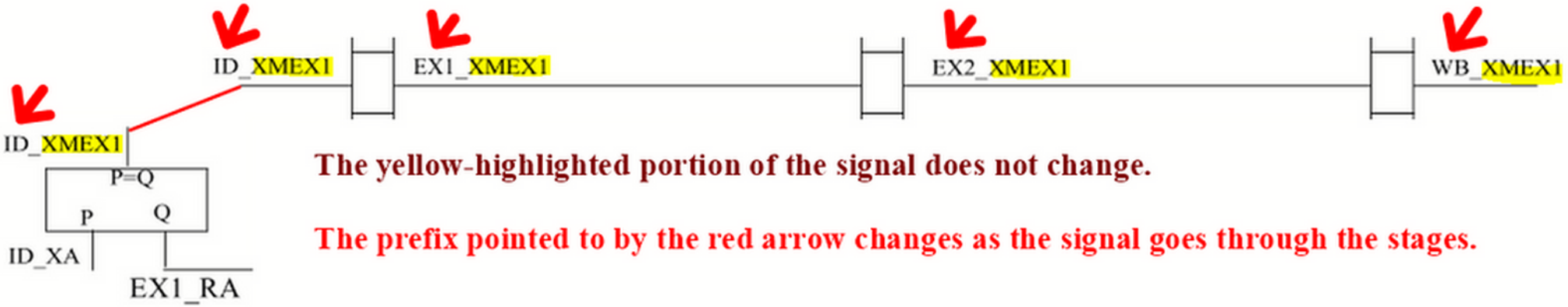
1. Complete all missing connections marked in dotted lines ▬ ▬ ▬
   Also complete the RA(Result Addreee) connection in ID stage (ID_RA).
2. Complete all five enable (EN) controls on the pipeline registers (including PC).
3. Complete the forwarding paths into the four forwarding muxes.
4. Complete logic to inject bubble into the next stage if the current instruction is being stalled or being flushed.
5. Draw the logic on a separate oage for generating **STALL, X_FORW1,Y_FORW1, Z_FORW1, Z_FORW2**.



Pinelined 3-element Adder
Block Diagram
LAB 7 Part1

Fig. 1

ee457_lab7_P1_comparator_inference_signal_naming_convention.pdf



The yellow-highlighted portion of the signal does not change.

The prefix pointed to by the red arrow changes as the signal goes through the stages.

Part 2

EX2 WB

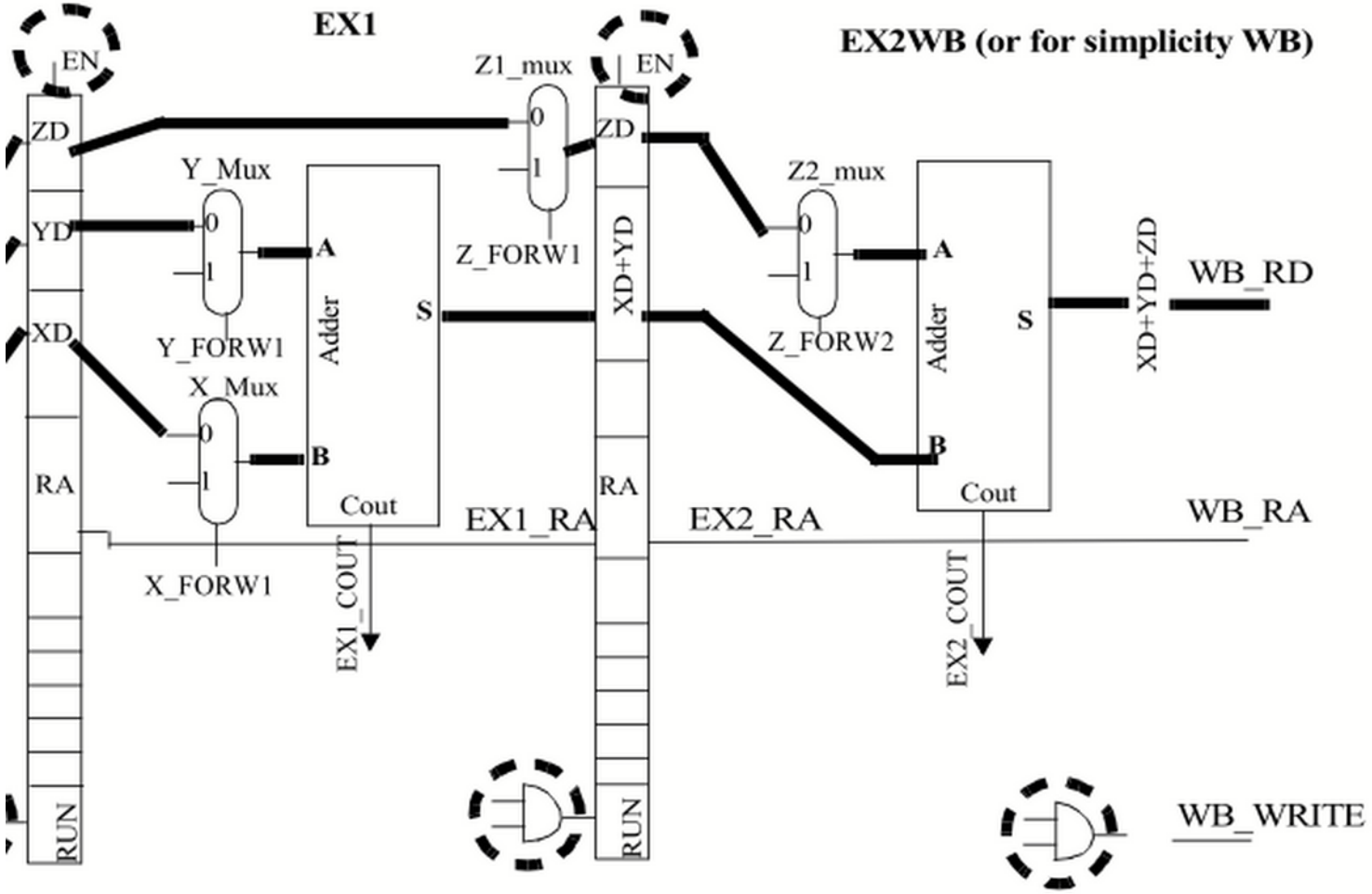
Combined into

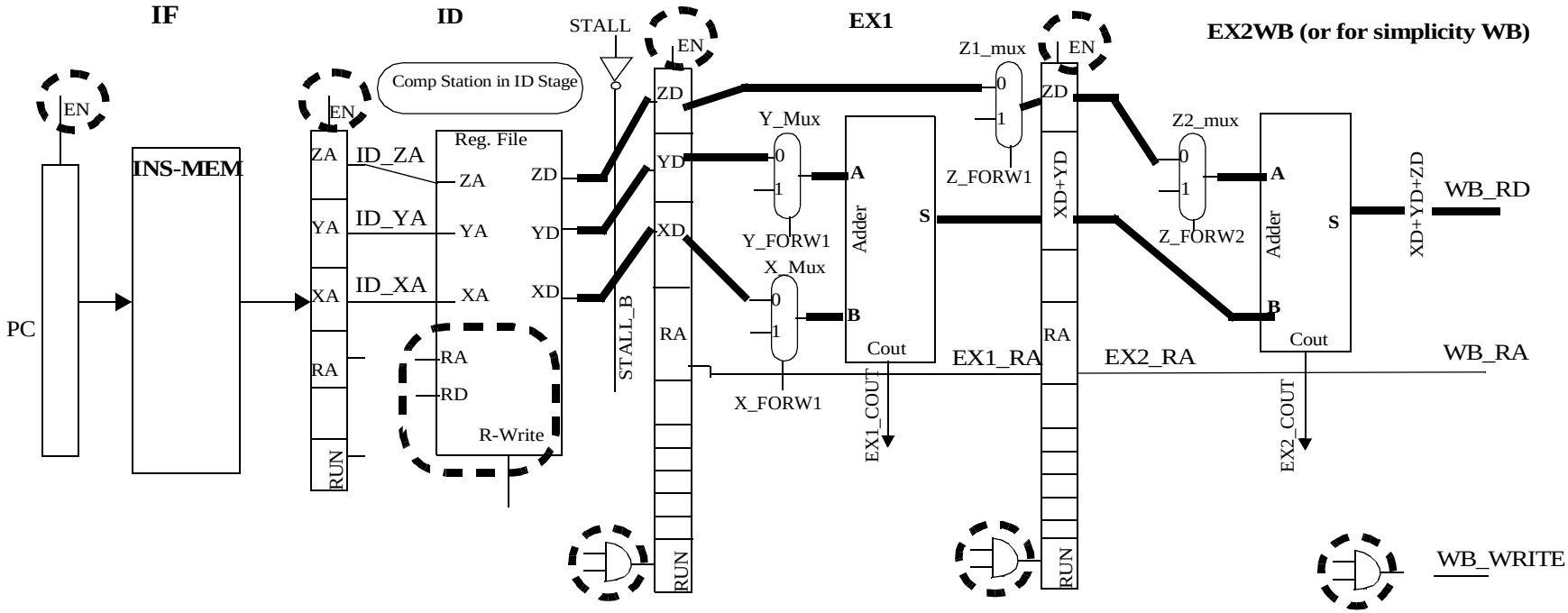
EX2 WB

We provided an incomplete block diagram for this part on the last page. We just removed the EX2/WB stage register (of the Part 1) but did not fix anything else. You please remove items which are not needed and complete the rest of this block diagram.

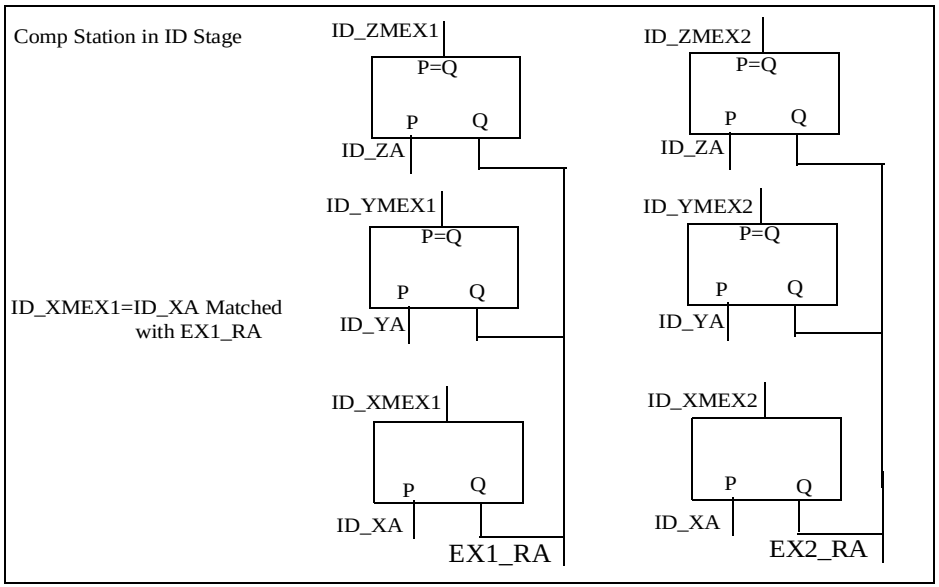
Though we are not doing timing design, let us apply this simple rule regarding helping (forwarding) towards the end of the clock. The register file is internally forwarding and we assume that the clock is wide enough for the instruction in EX2WB stage to perform the original EX2 operation of adding Z, checking to see if there was an overflow, and writing into the register file and forwarding the result data (write data) at the end of the clock to the instruction in the ID stage.

Question 2.1 If that is the case, the instruction in EX2WB stage should not have any difficulty to help the instruction in EX1 towards the end of the clock for the _____ (X / Y / Z) register as the recipient of the help does not have to perform any addition operation on this data.


EX1
EX2WB (or for simplicity WB)
EN
ZD
YD
XD
RA
RUN
Y_Mux
X_Mux
Y_FORW1
X_FORW1
Adder
A
B
S
Cout
EX1_COUT
Z1_mux
Z_FORW1
XD+YD
EX1_RA
Z2_mux
Z_FORW2
EX2_RA
EX2_COUT
XD+YD+ZD
WB_RD
WB_RA
WB_WRITE



This figure is basically the same as the Fig. 1 for part 1 on page 2 except that the EX2/WB stage register was removed as we are merging the EX2 and WB stages into one EX2WB stage.
You need to remove items which are not necessary and comple the rest..
You need to generate any STALL and/or FORW (forwadng) signals on this page itself.



Pinelined 3-element Adder
Block Diagram
LAB 7 Part2 (with EX2 and WB merged)

Fig. 2