Exercise 1: Given a pipelined processor with 5 stages and the following features:

- The Register File can be read and written in the same cycle.
- No forwarding technique is included.
- Branches are resolved at Decode. The next sequential instruction is fetched by default, and it is flushed if the branch is taken.

An application with the following features is executed:

- 20% of the instructions correspond to conditional branches, being 40% of them taken branches. There are not data dependences in branch instructions.

- 18% of the times, instruction I_{i+1} has a RAW dependency with instruction I_i (30% are load instructions).

- 6% of the times, instruction I_{i+2} has a RAW dependency with instruction I_i (30 % are load instructions), and in those cases there is never a dependency between I_{i+1} and I_{i+2}.

Obtain the following metrics:

a) CPI.

b) CPI if we add forwarding.

c) SPEED UP of b vs. a.

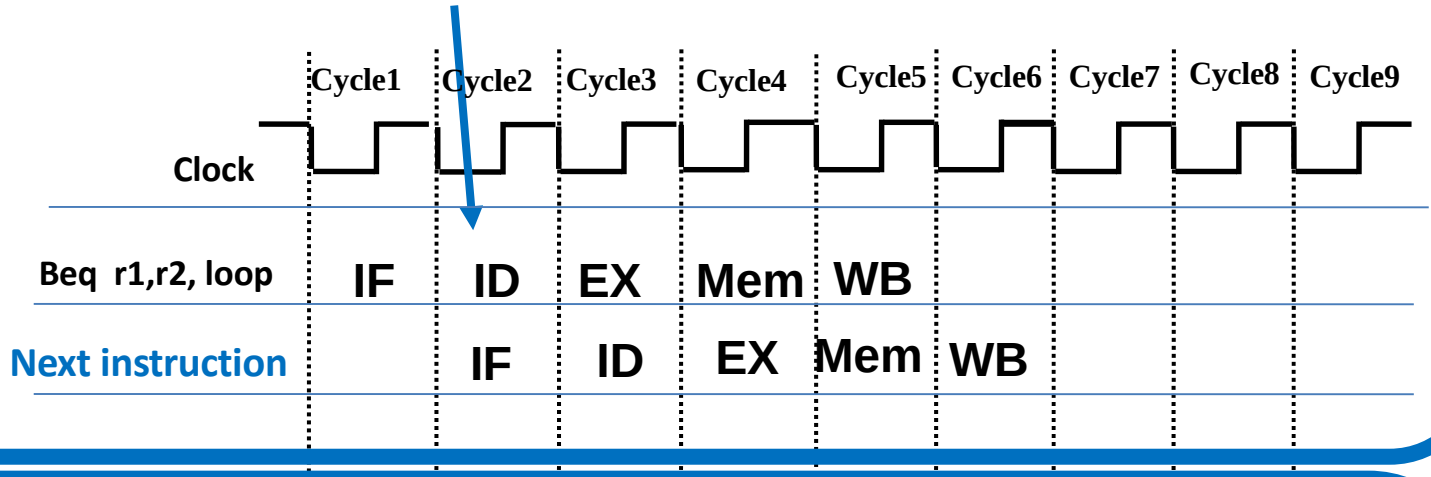
CPI = 1 + Penalty of branches + Penalty of RAW

- 20% of the instructions correspond to conditional branches, being 40% of them taken branches. There are not data dependences in branch instructions.

If the branch is not taken

20% x 60%

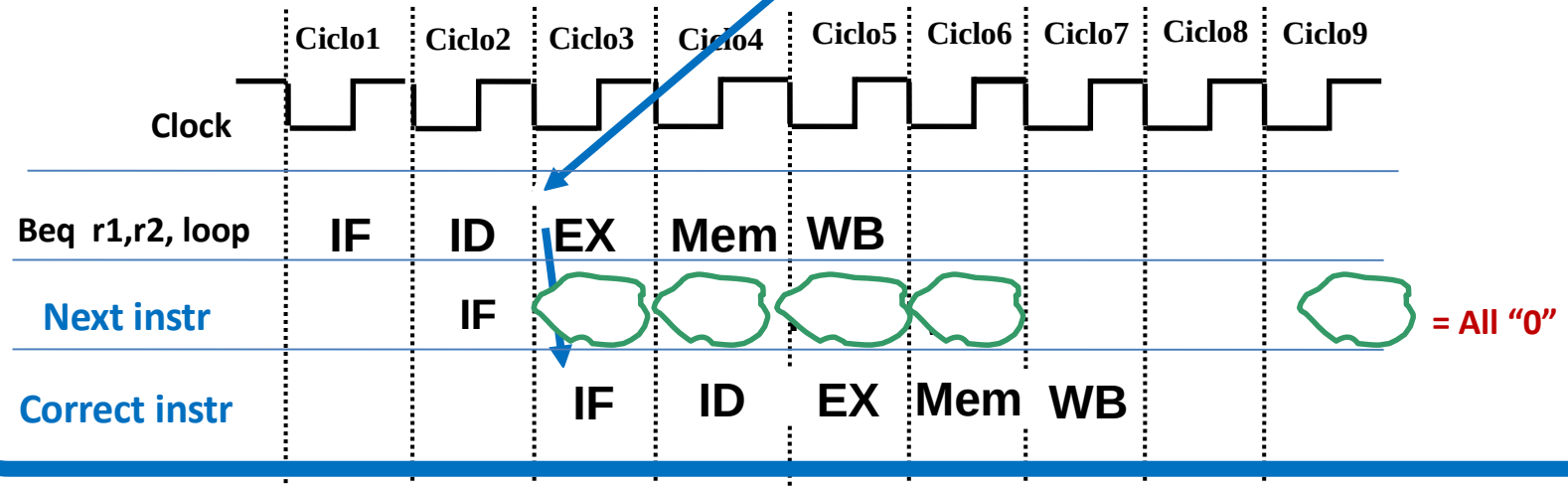
The **branch target address** and the **condition** evaluation are performed in **ID stage**



If the branch is taken

20% x 40%

The new adrress is stored in PC on the falling edge of the clock that starts the **Ex stage**

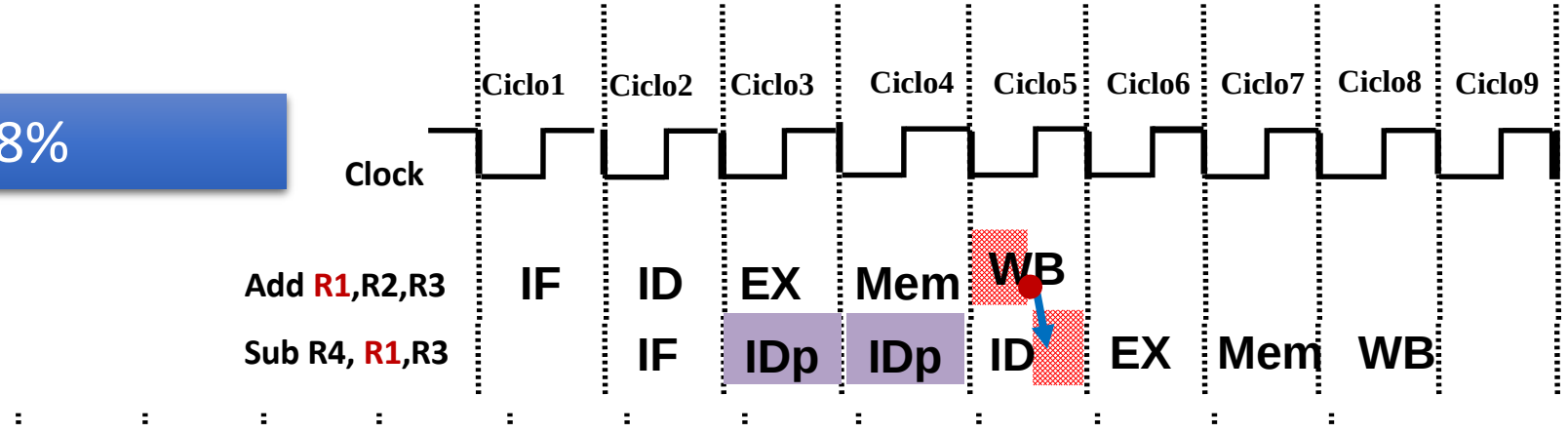


Penalty of branches = 20%x40%x1

RAW. No forwarding.

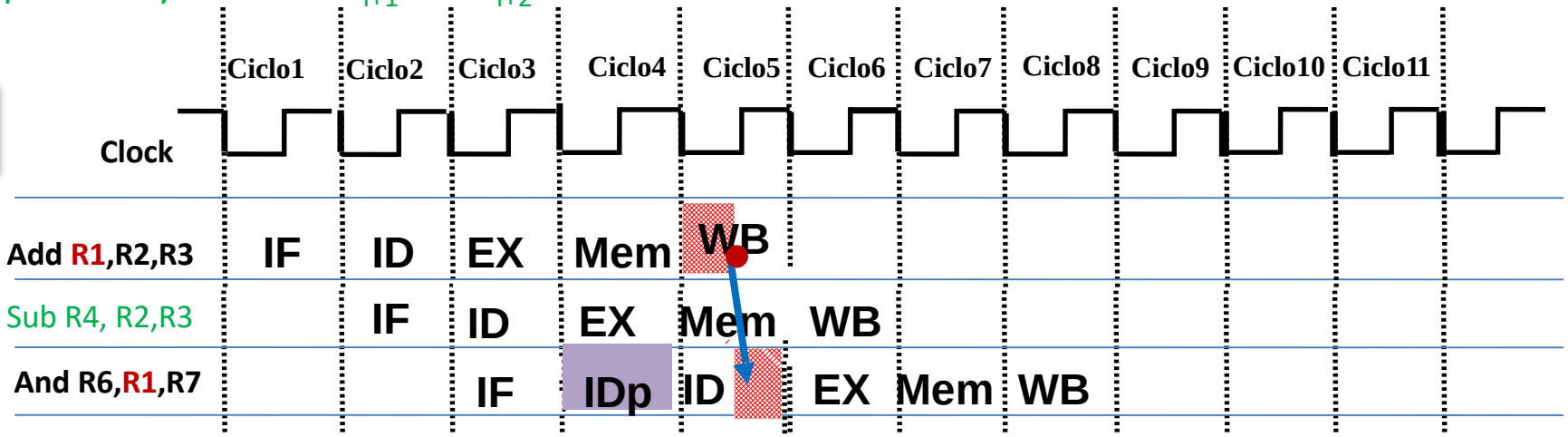
- 18% of the times, instruction I_{i+1} has a RAW dependency with instruction I_i (30% are load instructions).

18%



- 6% of the times, instruction I_{i+2} has a RAW dependency with instruction I_i (30 % are load instructions), and in those cases there is never a dependency between I_{i+1} and I_{i+2}.

6%

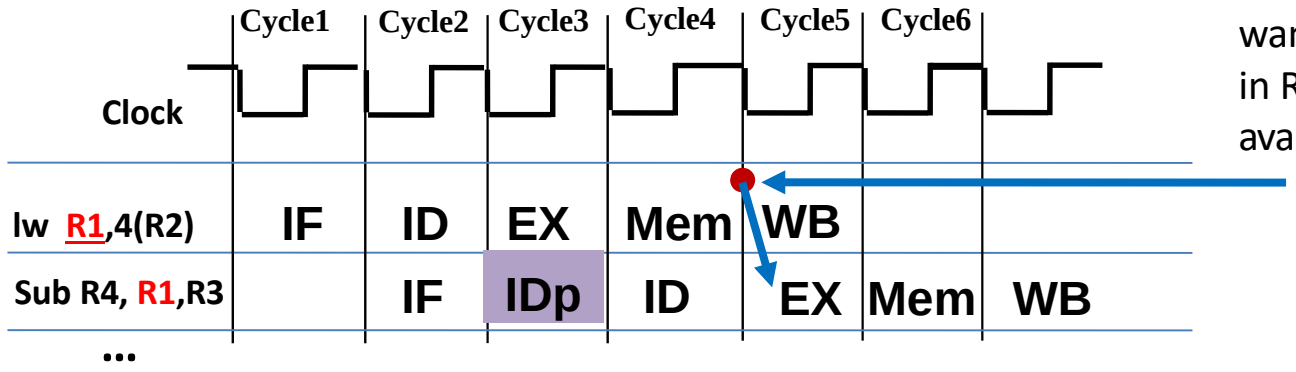


Penalty of RAW = 18%x2 + 6%x1

If we add forwarding: only LOAD instructions can cause stalls.

- 18% of the times, instruction I_{i+1} has a RAW dependency with instruction I_i (30% are load instructions).

18%x30%

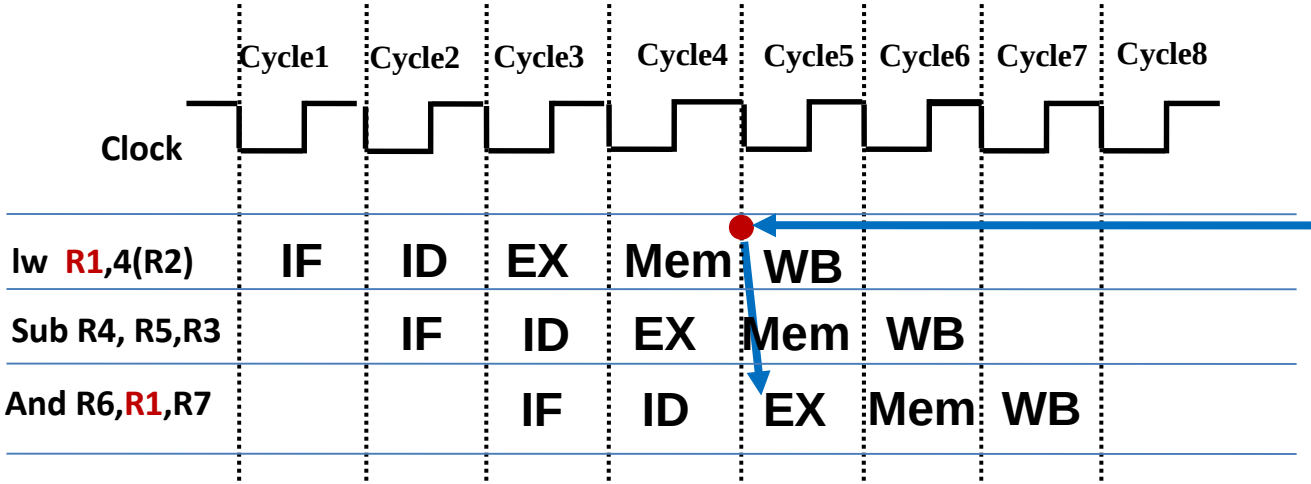


The value we want to store in R1 is available

Penalty of RAW = 18%x30%x1

- 6% of the times, instruction I_{i+2} has a RAW dependency with instruction I_i (30 % are load instructions), and in those cases there is never a dependency between I_{i+1} and I_{i+2}.

6%x30%



The value that we want to store in R1 is available

Obtain the following metrics:

CPI = 1 + Penalty of branches + Penalty of RAW

Penalty of branches = 20%x40%x1

Penalty of RAWa = 18%x2 + 6%x1

Penalty of RAWb = 18%x30%x1

a) CPI.

CPI = 1 + 0,08 + 0,42 = 1,5

b) CPI if we add forwarding.

CPI = 1 + 0,08 + 0,054 = 1,134

c) SPEEDUP of b vs. a.

a) Speedup = CPIb / CPIa = 1,32