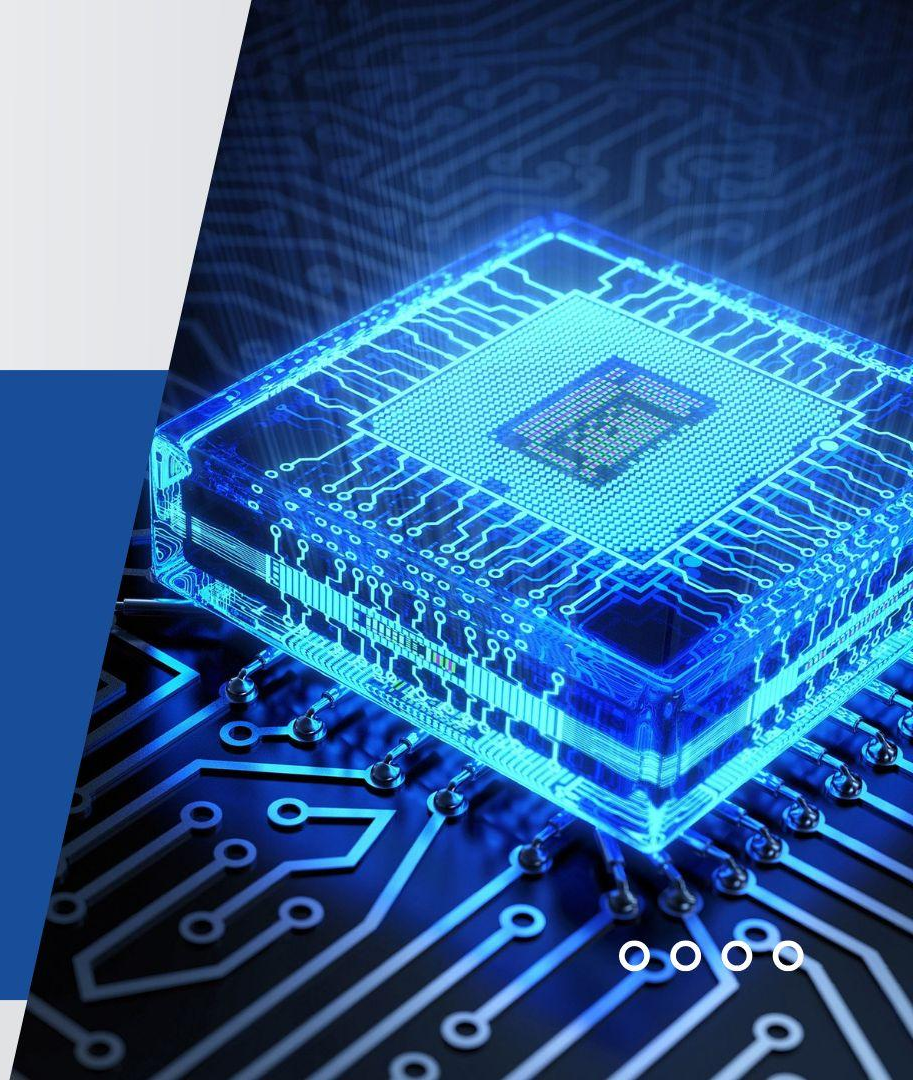
MERL

Accelerating Engineering Innovation

RISC-V Pipeline Core

MERL DSU

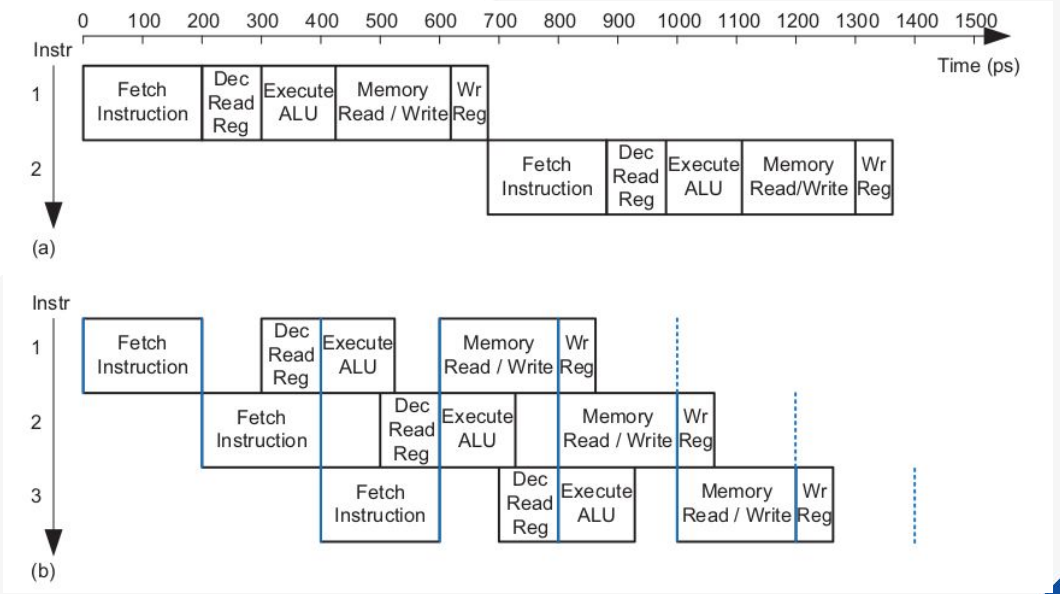
TABLE OF CONTENTS



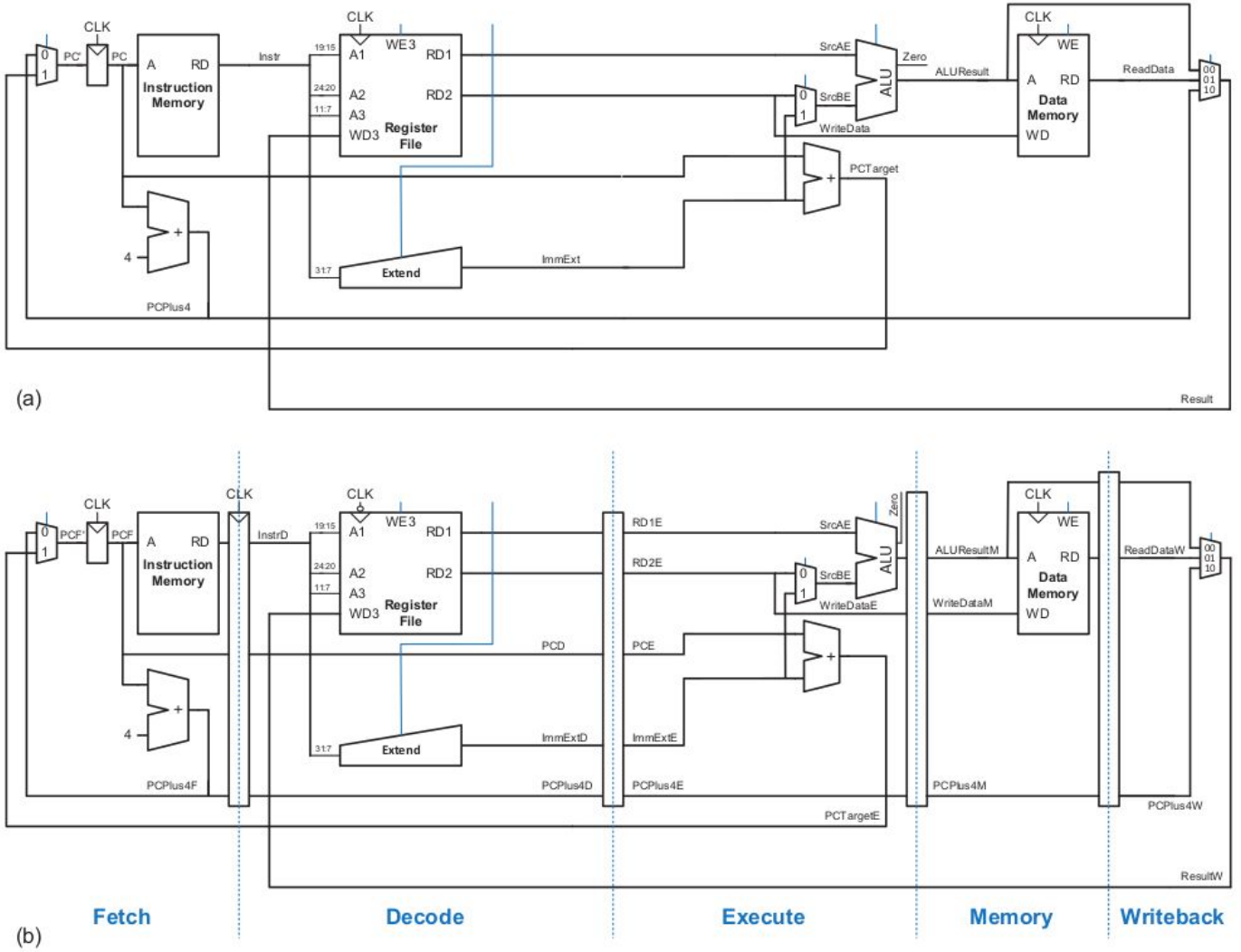
Overview of RISC-V Pipeline Architecture

Pipelining

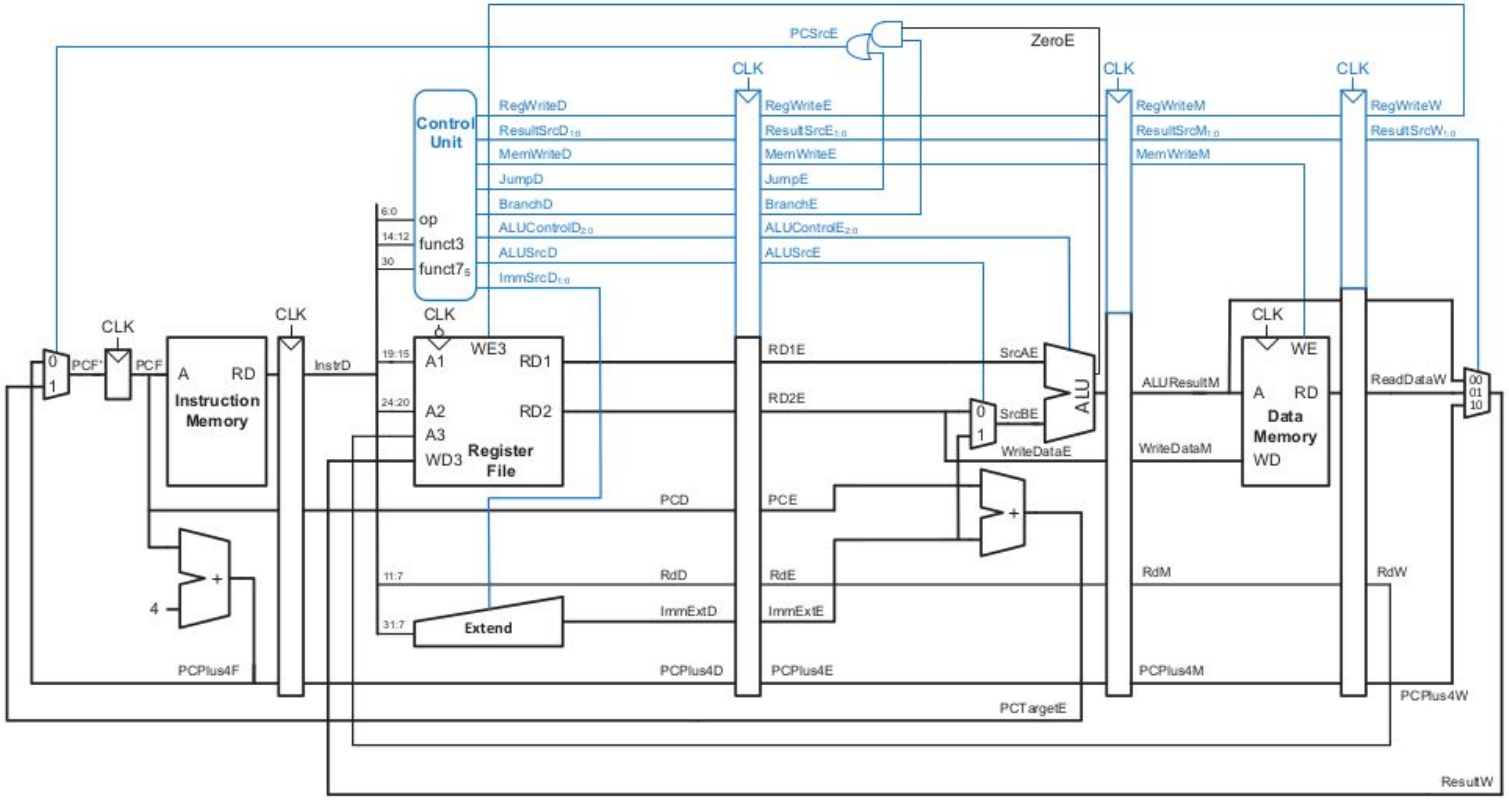
We design a pipelined processor by subdividing the single-cycle processor into five pipeline stages. Thus, five instructions can execute simultaneously, one in each stage. Because each stage has only one-fifth of the entire logic, the clock frequency is approximately five times faster.



Pipelining

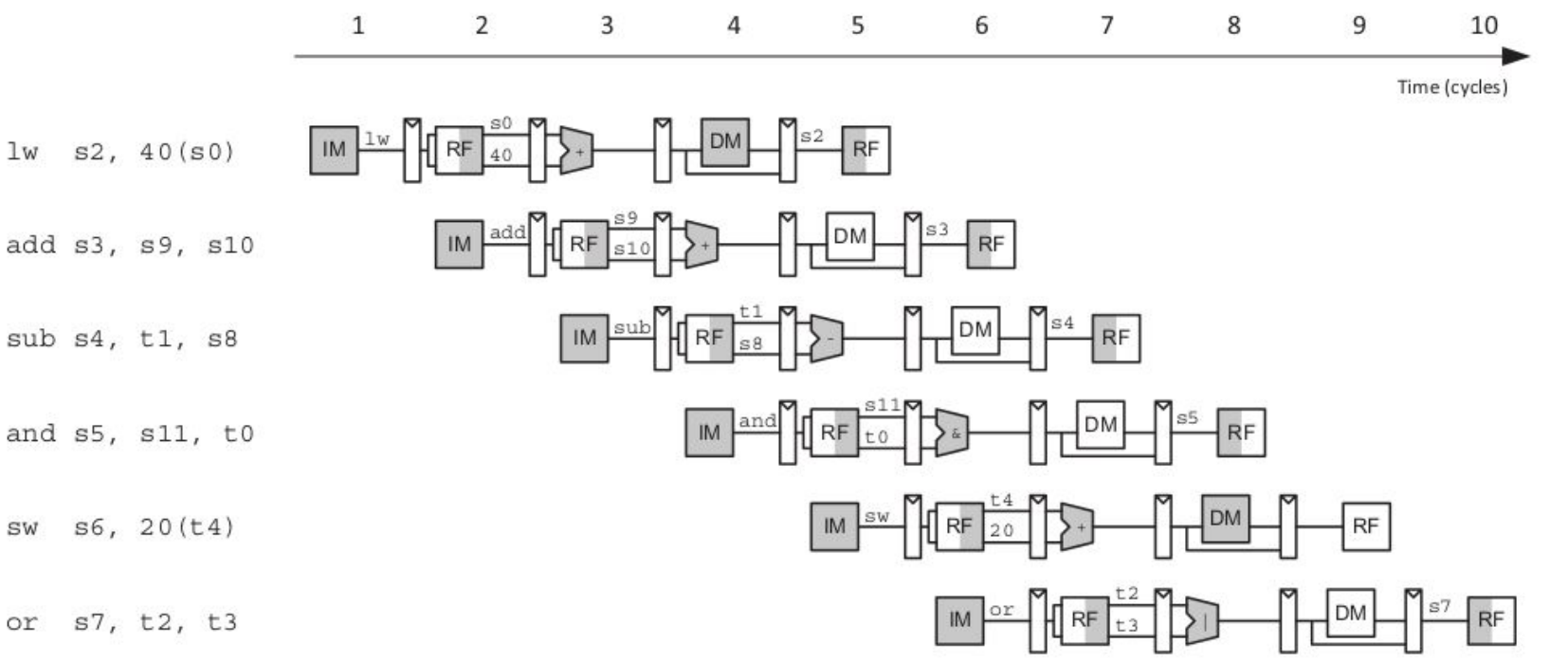

CLK
PC
PC
A
RD
Instruction Memory
Instr
19:15
24:20
11:7
A1
A2
A3
WD3
WE3
RD1
RD2
Register File
SrcAE
Zero
ALU
ALUResult
SrcBE
WriteData
CLK
WE
A
RD
Data Memory
WD
ReadData
00
01
10
PCTarget
4
ImmExt
31:7
Extend
PCPlus4
Result
(a)
CLK
PCF'
PCF
A
RD
Instruction Memory
InstrD
19:15
24:20
11:7
A1
A2
A3
WD3
WE3
RD1
RD2
Register File
RD1E
RD2E
SrcAE
Zero
ALU
ALUResultM
SrcBE
WriteDataE
WriteDataM
CLK
WE
A
RD
Data Memory
WD
ReadDataW
00
01
10
PCD
PCE
4
ImmExtD
ImmExtE
31:7
Extend
PCPlus4F
PCPlus4D
PCPlus4E
PCPlus4M
PCPlus4W
PCTargetE
ResultW
Fetch
Decode
Execute
Memory
Writeback
(b)

Pipeline Datapath



Implementation of Fetch Cycle

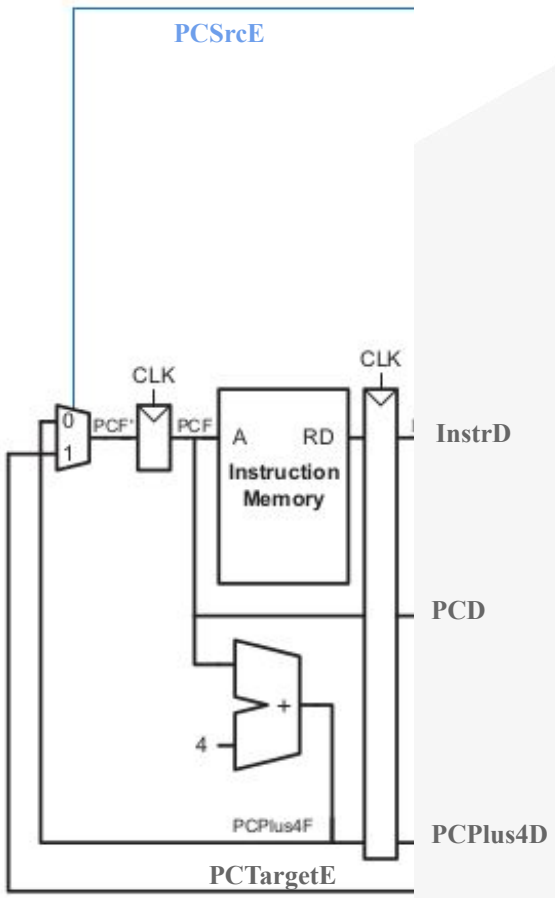
Abstract View of Pipelining



Fetch Cycle Datapath

Modules to be Integrated:

1) PC Mux
2) Program Counter
3) Adder
4) Instruction Memory
5) Fetch Stage Registers

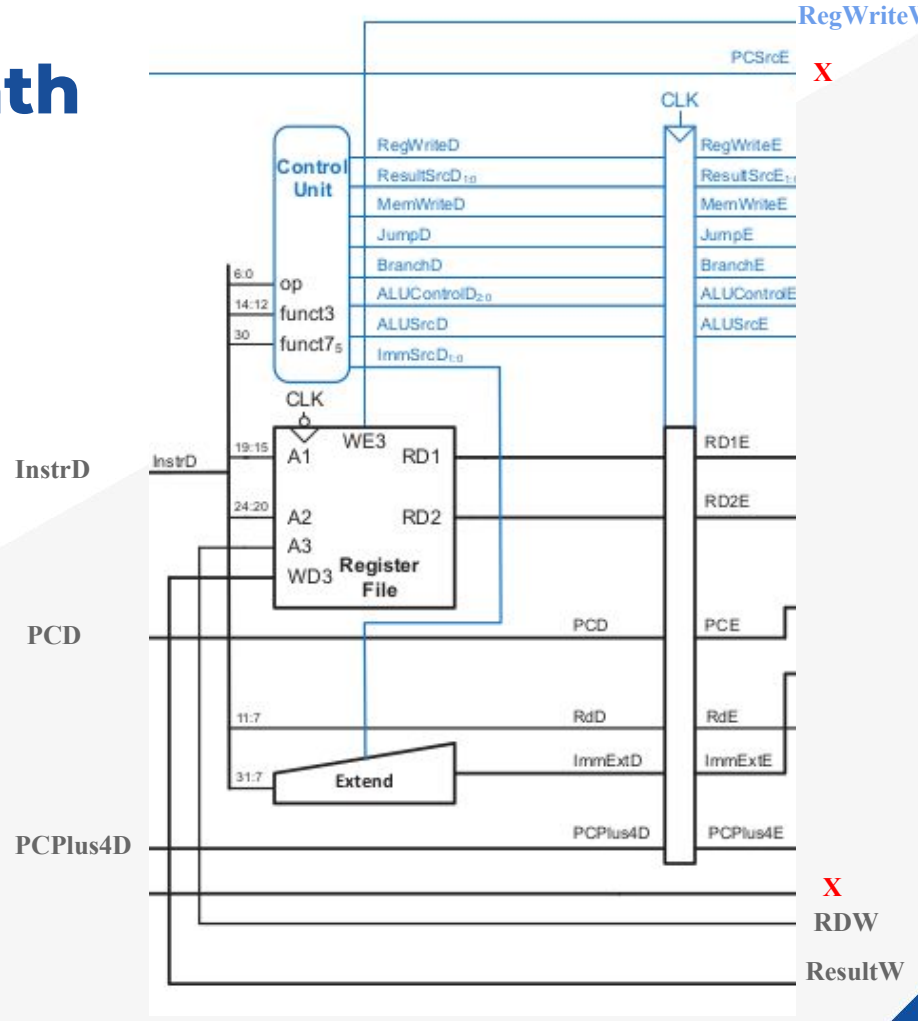


Implementation of Decode Cycle

Decode Cycle Datapath

Modules to be Integrated:

1) Control Unit
2) Register File
3) Extender
4) Decode Stage Registers

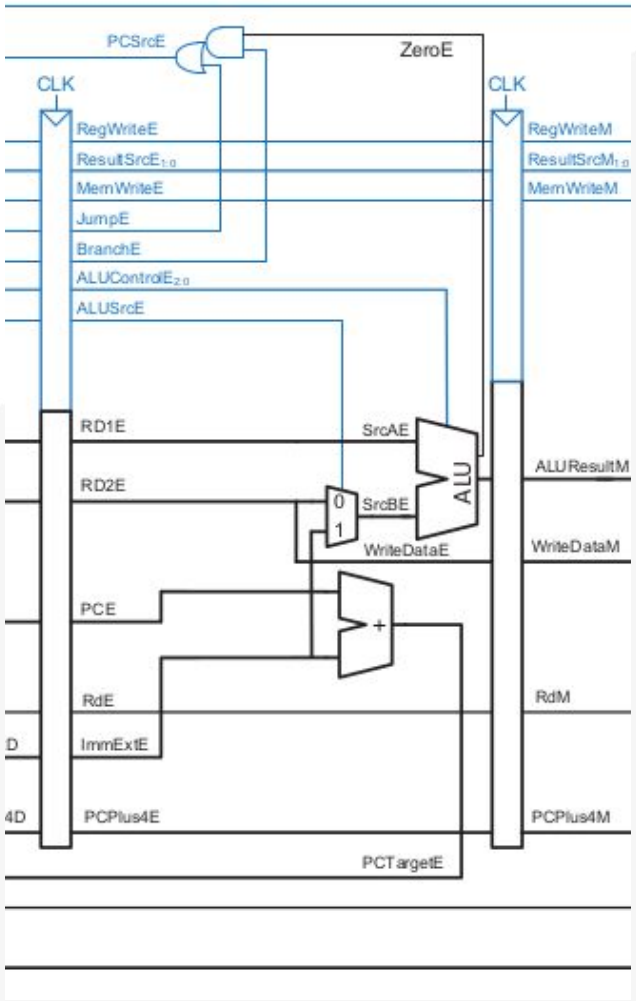


Implementation of Execute Cycle

Execute Cycle Datapath

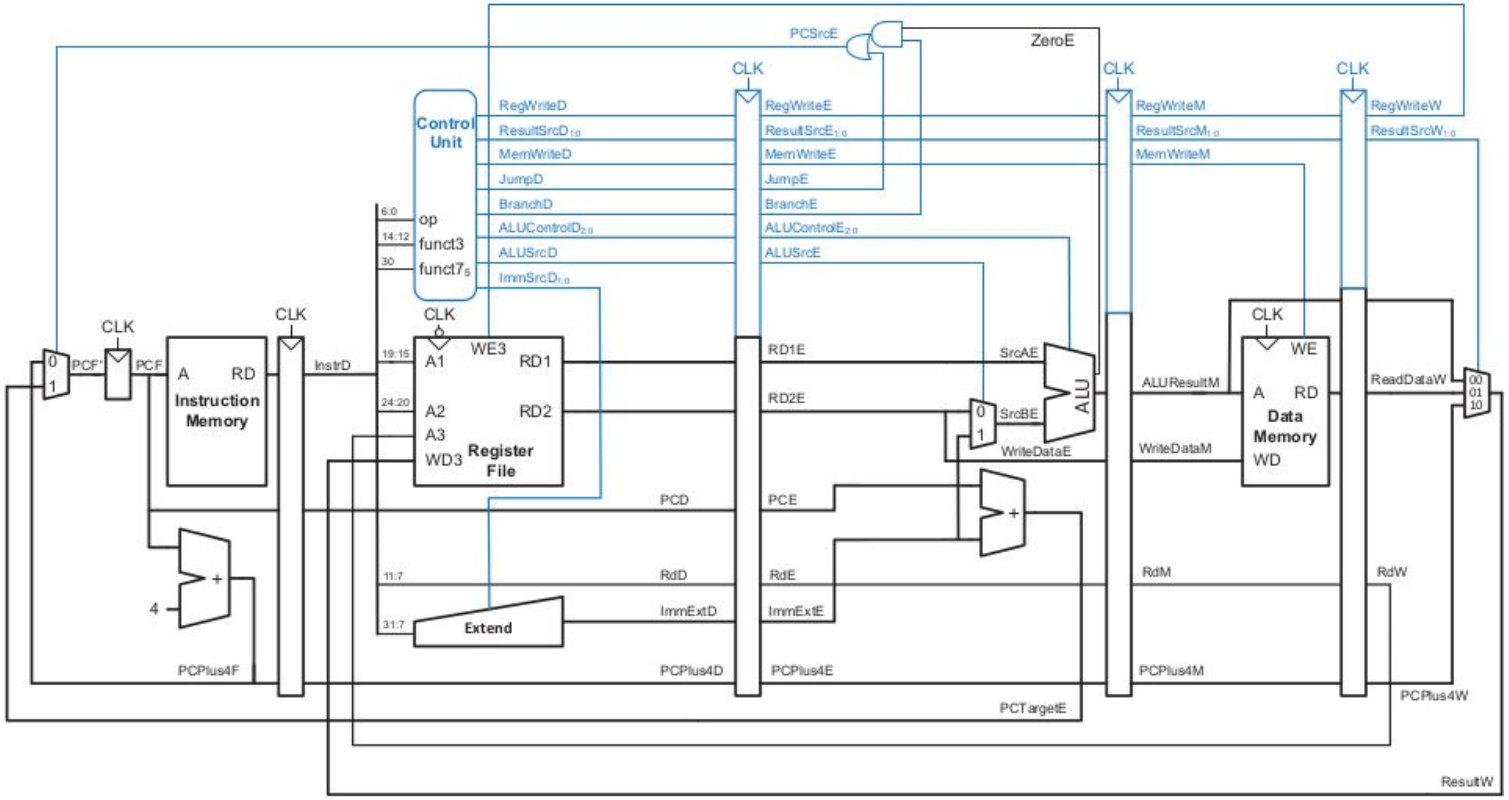
Modules to be Integrated:

1) AND Gate
2) Mux
3) Adder
4) ALU
5) Execute Stage Registers



Implementation of Memory Cycle

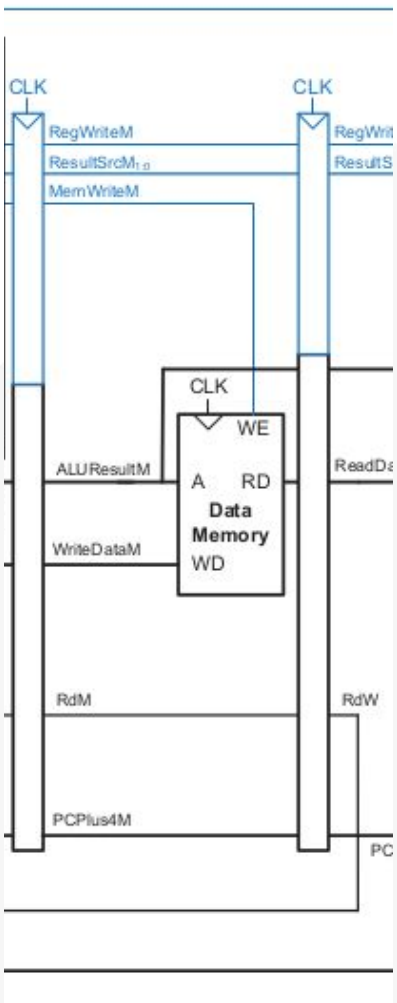
Pipeline Datapath



Memory Cycle Datapath

Modules to be Integrated:

1) Data Memory
2) Memory Stage Registers

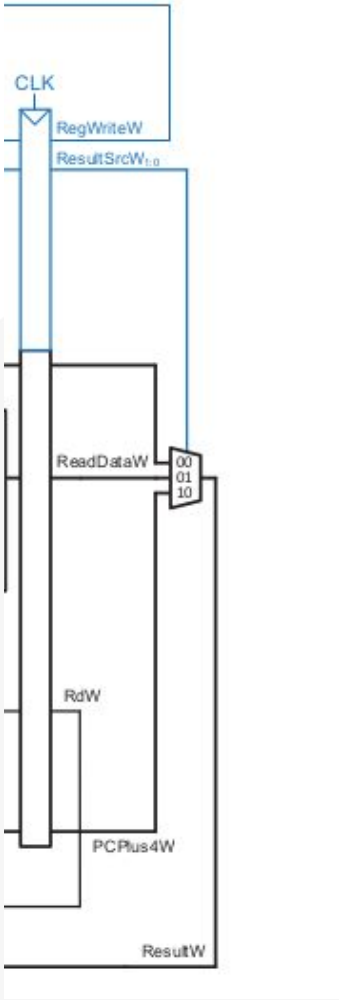


Implementation of Write Back Cycle

Write Back Cycle Datapath

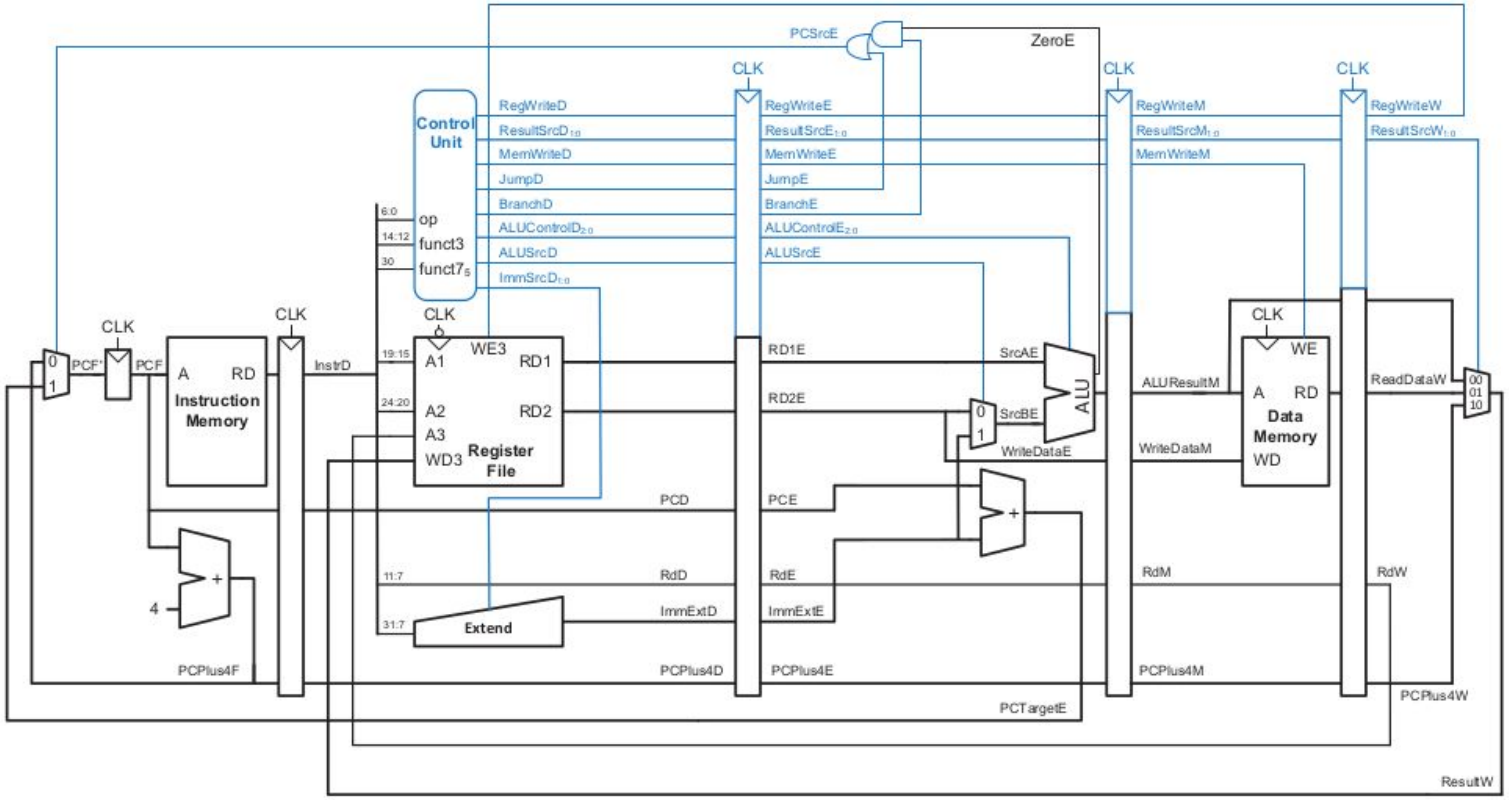
Modules to be Integrated:

1) Mux



Implementation of Pipeline Top

Pipeline Datapath



Pipeline Hazards

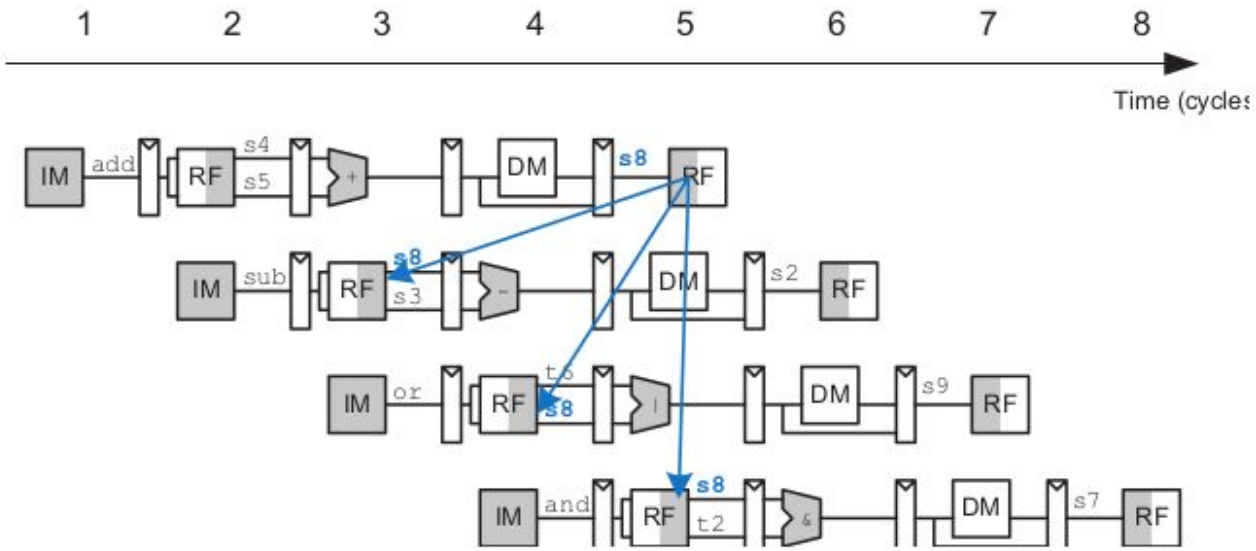
Pipeline Hazard

- ***Structural Hazard***
  1. Hardware does not support the execution of instruction in same clock cycle.
  2. Without having Two memories RISC-V pipelining architecture will have structural hazard.

- ***Data Hazard***
  1. Data to be executed is not available.
  2. May occur when pipeline is stalled.
  3. Solve by using **forwarding** or **bypassing** technique.

Data Hazard In Pipelining

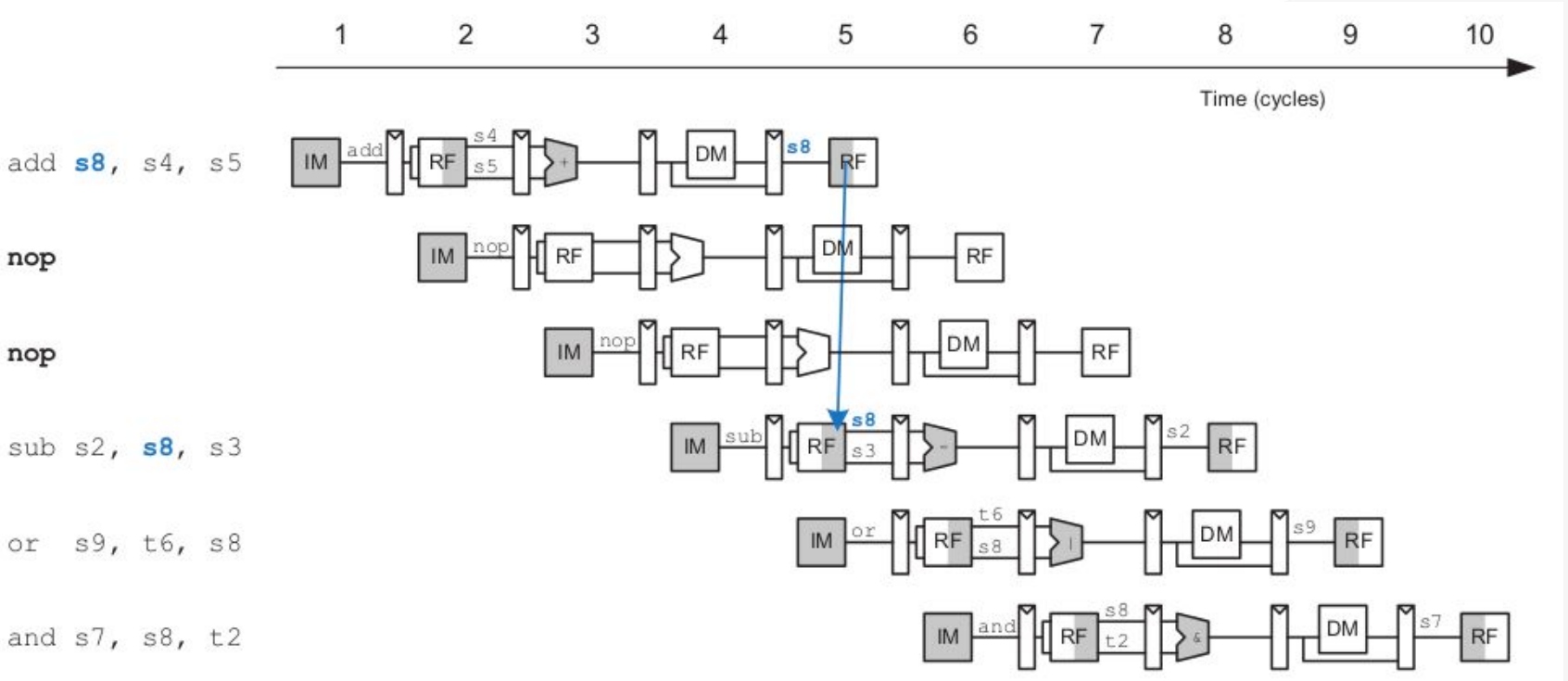
```
add s8, s4, s5
sub s2, s8, s3
or  s9, t6, s8
and s7, s8, t2
```



Solution of Data Hazards

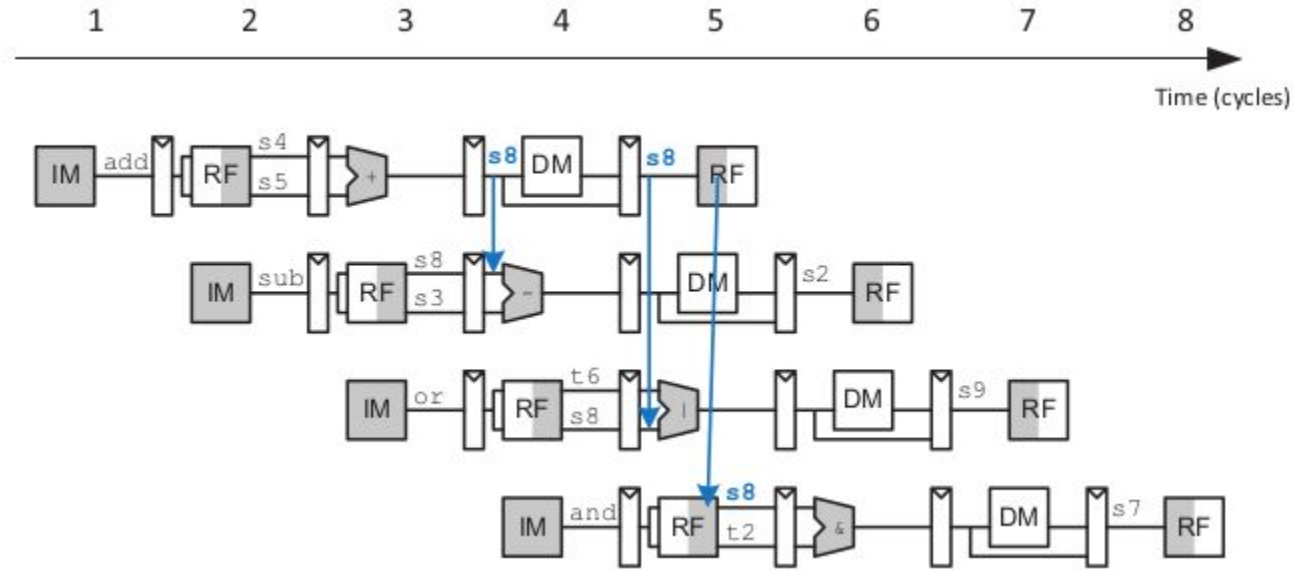
- ***Solving Data Hazards with nops***
- ***Solving Data Hazard with Forwarding / Bypassing***

Using Nops

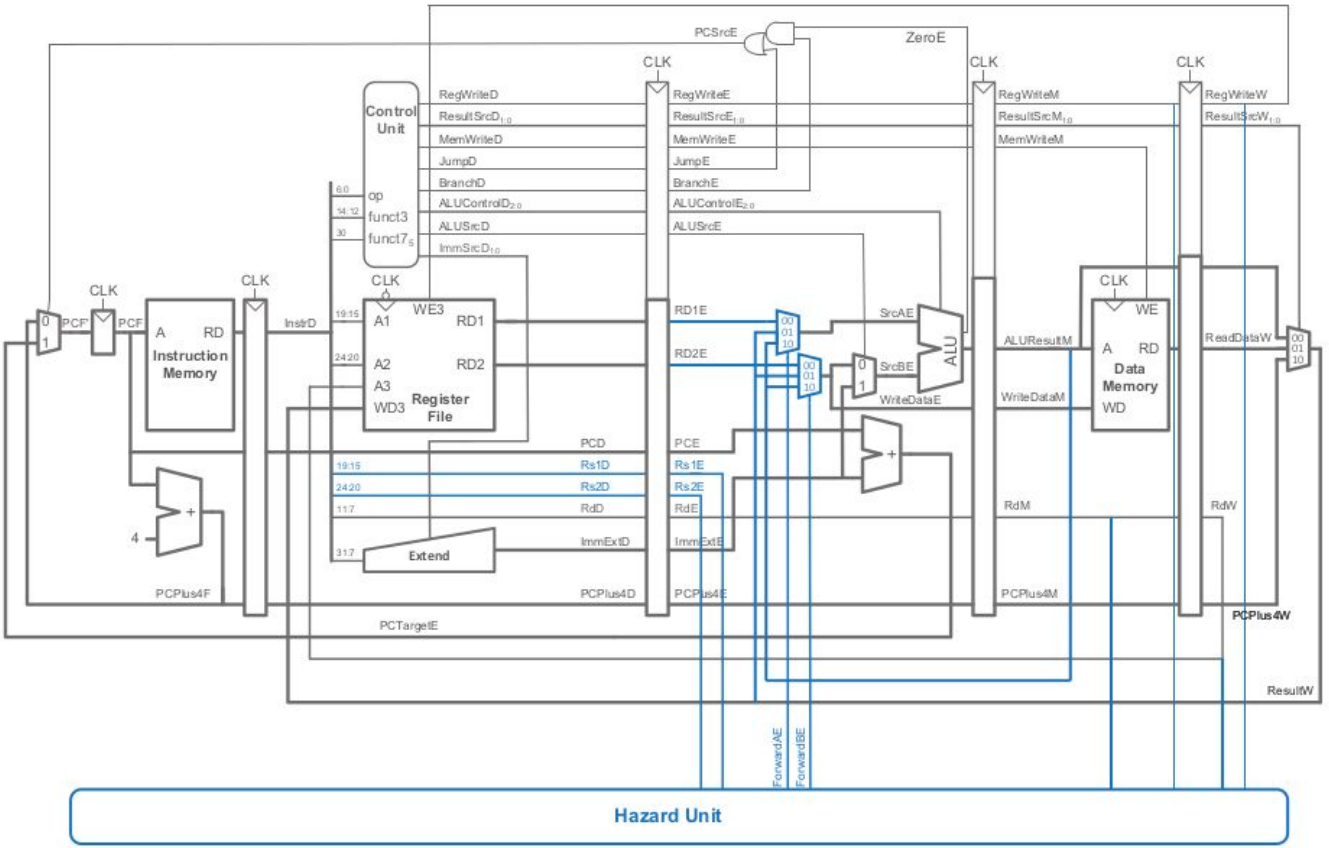

1
2
3
4
5
6
7
8
9
10
Time (cycles)
add s8, s4, s5
nop
nop
sub s2, s8, s3
or s9, t6, s8
and s7, s8, t2
IM
add
RF
s4
s5
+
DM
s8
RF
IM
nop
RF
DM
RF
IM
nop
RF
DM
RF
IM
sub
RF
s8
s3
-
DM
s2
RF
IM
or
RF
t6
s8
|
DM
s9
RF
IM
and
RF
s8
t2
&
DM
s7
RF

Using Forwarding / Bypassing

```
add s8, s4, s5
sub s2, s8, s3
or  s9, t6, s8
and s7, s8, t2
```



Updated Pipeline Top Architecture



Implementation of Hazard Unit

Condition Table

Mux control	Source	Explanation
ForwardA = 00	ID/EX	The first ALU operand comes from the register file.
ForwardA = 10	EX/MEM	The first ALU operand is forwarded from the prior ALU result.
ForwardA = 01	MEM/WB	The first ALU operand is forwarded from data memory or an earlier ALU result.
ForwardB = 00	ID/EX	The second ALU operand comes from the register file.
ForwardB = 10	EX/MEM	The second ALU operand is forwarded from the prior ALU result.
ForwardB = 01	MEM/WB	The second ALU operand is forwarded from data memory or an earlier ALU result.

Condition for Data Hazard

Memory Stage

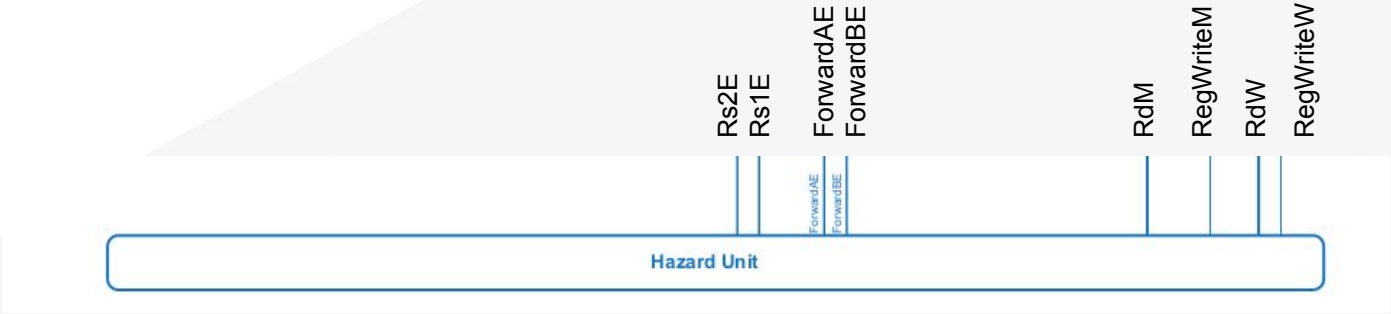
if (RegWriteM and (RdM != 0) and (RdM == Rs1E))
ForwardAE = 10

if (RegWriteM and (RdM != 0) and (RdM == Rs2E))
ForwardBE = 10

WriteBack Stage

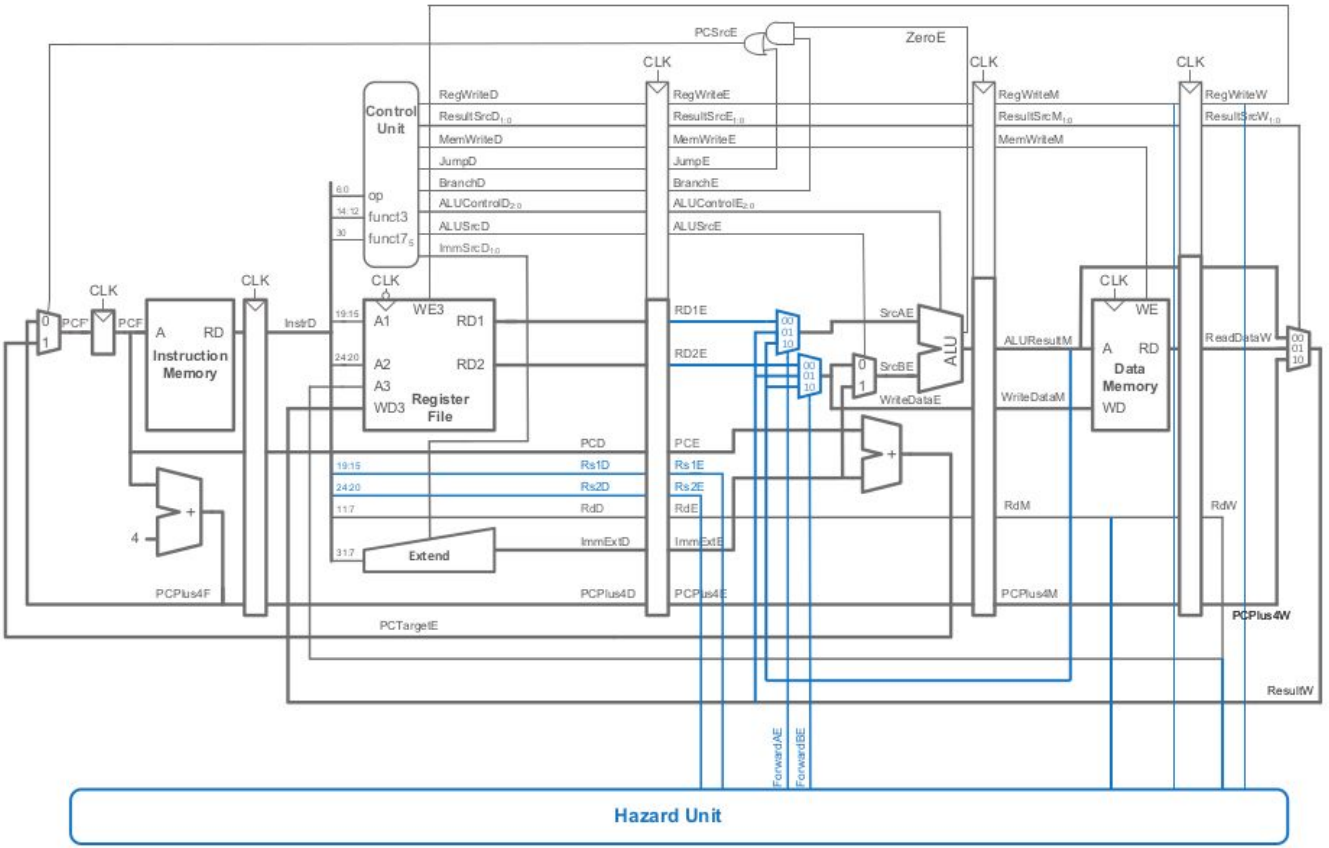
if (RegWriteW and (RdW != 0) and (RdW == Rs1E))
ForwardAE = 01

if (RegWriteW and (RdW != 0) and (RdW == Rs2E))
ForwardBE = 01



Implementation of Pipeline Top II

Updated Pipeline Top Architecture



Thank You