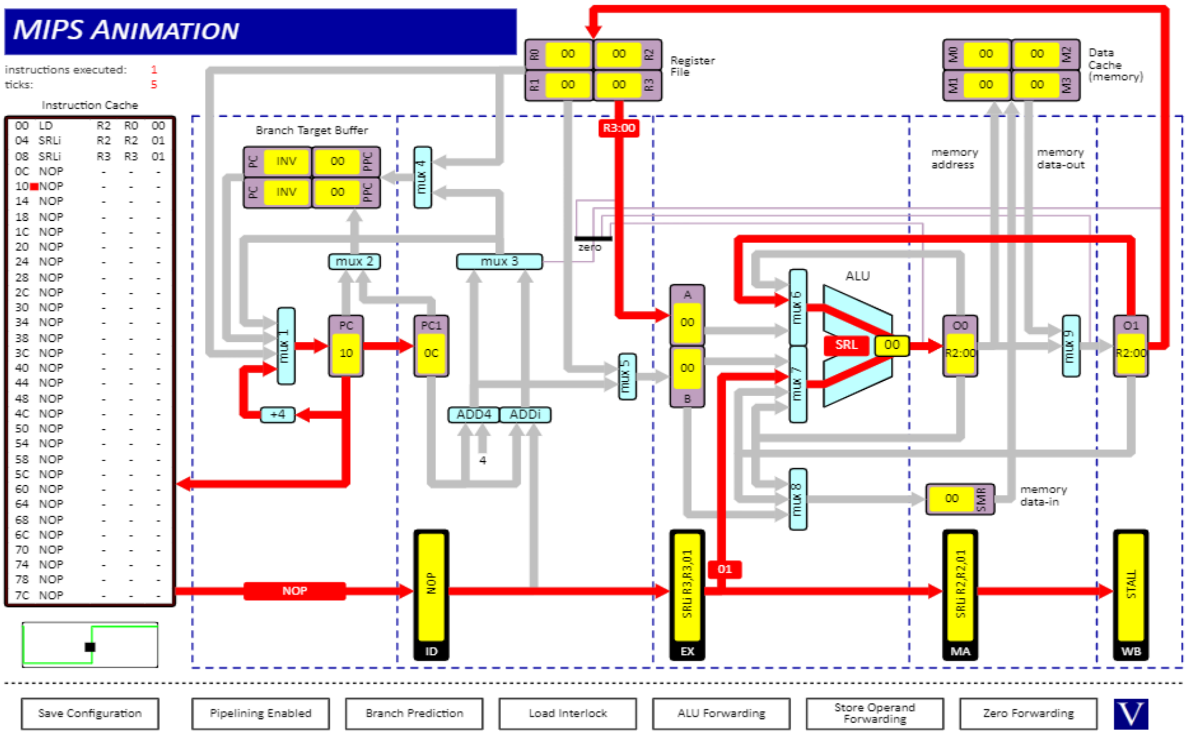
Computer Architecture Tutorial 4

Chloe Conneely 17323080

Q1.

a)



MIPS ANIMATION

instructions executed: 1
ticks: 5

Instruction Cache

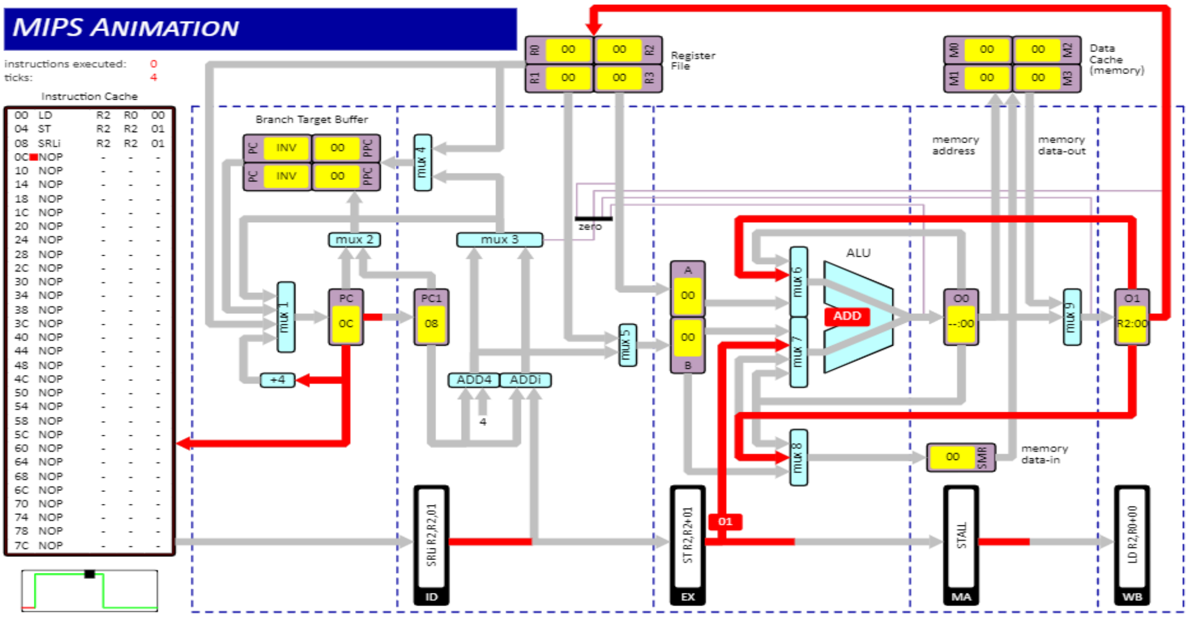
```
00  LD     R2  R0  00
04  SRLi   R2  R2  01
08  SRLi   R3  R3  01
0C  NOP    -   -   -
10  NOP    -   -   -
14  NOP    -   -   -
18  NOP    -   -   -
1C  NOP    -   -   -
20  NOP    -   -   -
24  NOP    -   -   -
28  NOP    -   -   -
2C  NOP    -   -   -
30  NOP    -   -   -
34  NOP    -   -   -
38  NOP    -   -   -
3C  NOP    -   -   -
40  NOP    -   -   -
44  NOP    -   -   -
48  NOP    -   -   -
4C  NOP    -   -   -
50  NOP    -   -   -
54  NOP    -   -   -
58  NOP    -   -   -
5C  NOP    -   -   -
60  NOP    -   -   -
64  NOP    -   -   -
68  NOP    -   -   -
6C  NOP    -   -   -
70  NOP    -   -   -
74  NOP    -   -   -
78  NOP    -   -   -
7C  NOP    -   -   -
```

LD R2, R0, 00

SRLi R2, R2, 01

SRLi R3, R3, 01

b)



MIPS ANIMATION

instructions executed: 0
ticks: 4

Instruction Cache

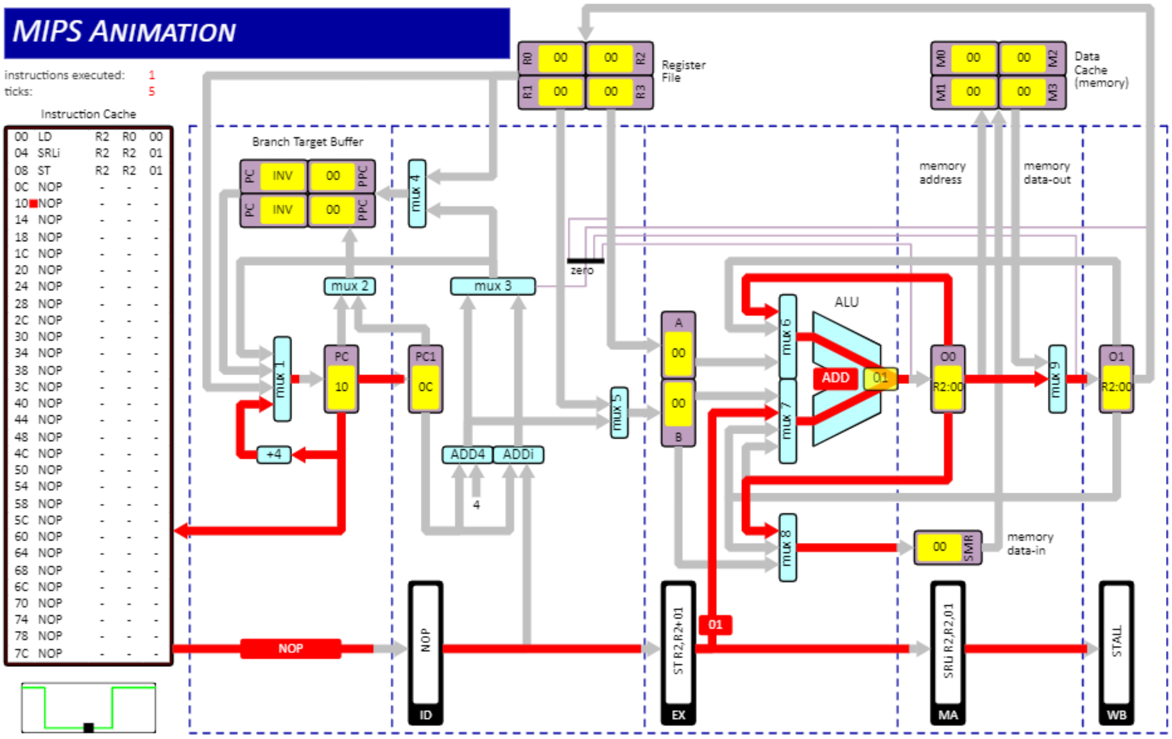
```
00  LD     R2  R0  00
04  ST     R2  R2  01
08  SRLi   R2  R2  01
0C  NOP    -   -   -
10  NOP    -   -   -
14  NOP    -   -   -
18  NOP    -   -   -
1C  NOP    -   -   -
20  NOP    -   -   -
24  NOP    -   -   -
28  NOP    -   -   -
2C  NOP    -   -   -
30  NOP    -   -   -
34  NOP    -   -   -
38  NOP    -   -   -
3C  NOP    -   -   -
40  NOP    -   -   -
44  NOP    -   -   -
48  NOP    -   -   -
4C  NOP    -   -   -
50  NOP    -   -   -
54  NOP    -   -   -
58  NOP    -   -   -
5C  NOP    -   -   -
60  NOP    -   -   -
64  NOP    -   -   -
68  NOP    -   -   -
6C  NOP    -   -   -
70  NOP    -   -   -
74  NOP    -   -   -
78  NOP    -   -   -
7C  NOP    -   -   -
```

LD R2, R0, 00

ST R2, R2, 01

SRLi R2, R2, 01

c)



MIPS ANIMATION

instructions executed: 1
ticks: 5

Instruction Cache

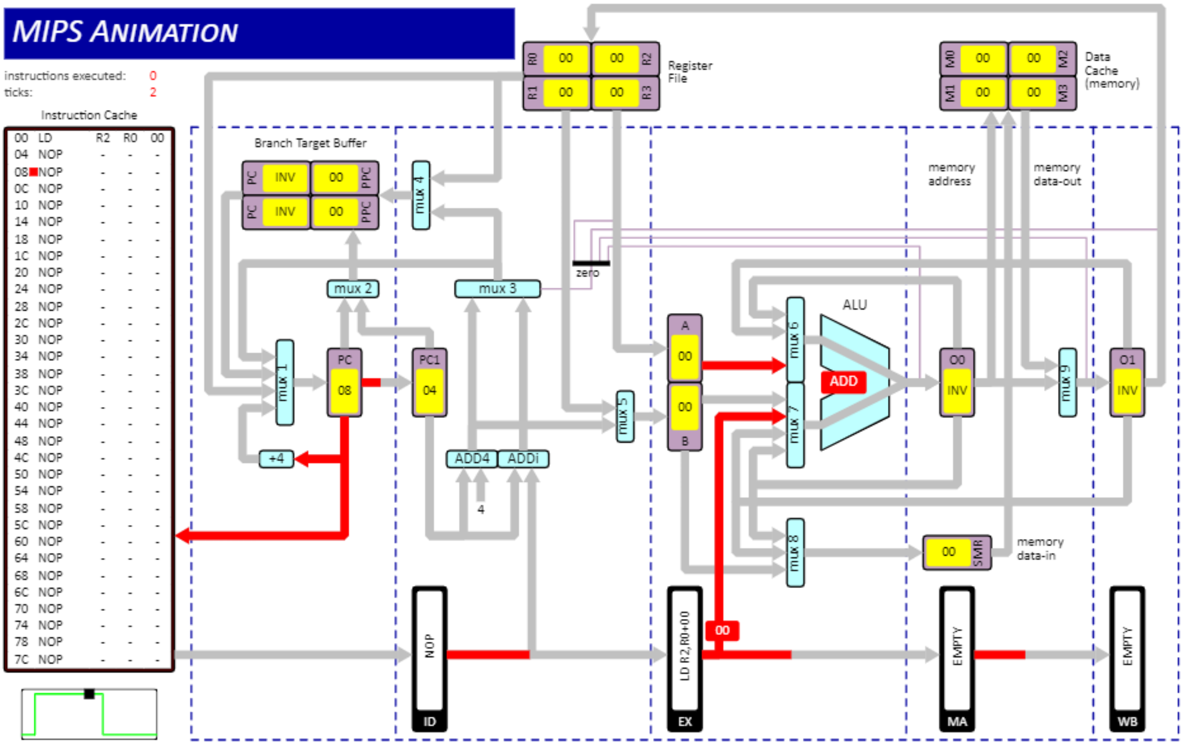
00	LD	R2	R0	00
04	SRLi	R2	R2	01
08	ST	R2	R2	01
0C	NOP	-	-	-
10	NOP	-	-	-
14	NOP	-	-	-
18	NOP	-	-	-
1C	NOP	-	-	-
20	NOP	-	-	-
24	NOP	-	-	-
28	NOP	-	-	-
2C	NOP	-	-	-
30	NOP	-	-	-
34	NOP	-	-	-
38	NOP	-	-	-
3C	NOP	-	-	-
40	NOP	-	-	-
44	NOP	-	-	-
48	NOP	-	-	-
4C	NOP	-	-	-
50	NOP	-	-	-
54	NOP	-	-	-
58	NOP	-	-	-
5C	NOP	-	-	-
60	NOP	-	-	-
64	NOP	-	-	-
68	NOP	-	-	-
6C	NOP	-	-	-
70	NOP	-	-	-
74	NOP	-	-	-
78	NOP	-	-	-
7C	NOP	-	-	-

LD R2, R0, 00

SRLi R2, R2, 01

ST R2, R2, 01

d)



MIPS ANIMATION

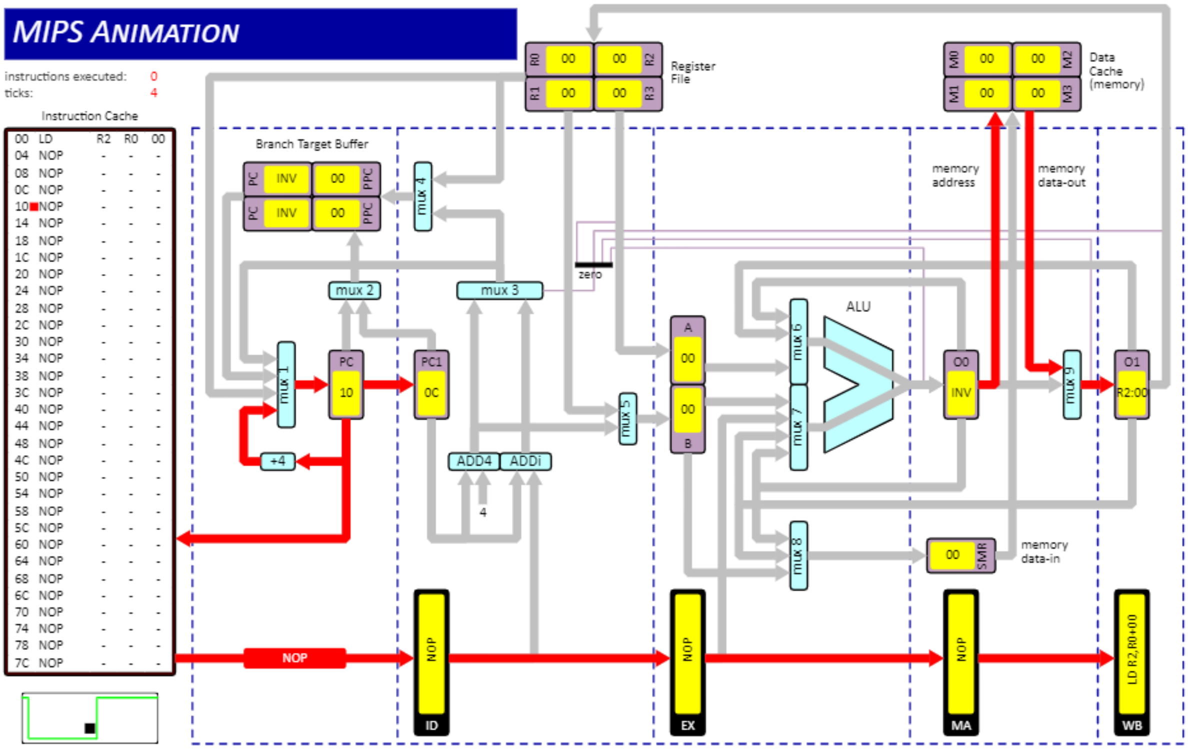
instructions executed: 0
ticks: 2

Instruction Cache

```
00 LD   R2 R0 00
04 NOP  -  -  -
08 NOP  -  -  -
0C NOP  -  -  -
10 NOP  -  -  -
14 NOP  -  -  -
18 NOP  -  -  -
1C NOP  -  -  -
20 NOP  -  -  -
24 NOP  -  -  -
28 NOP  -  -  -
2C NOP  -  -  -
30 NOP  -  -  -
34 NOP  -  -  -
38 NOP  -  -  -
3C NOP  -  -  -
40 NOP  -  -  -
44 NOP  -  -  -
48 NOP  -  -  -
4C NOP  -  -  -
50 NOP  -  -  -
54 NOP  -  -  -
58 NOP  -  -  -
5C NOP  -  -  -
60 NOP  -  -  -
64 NOP  -  -  -
68 NOP  -  -  -
6C NOP  -  -  -
70 NOP  -  -  -
74 NOP  -  -  -
78 NOP  -  -  -
7C NOP  -  -  -
```

LD R2, R0, 00

e)



MIPS ANIMATION

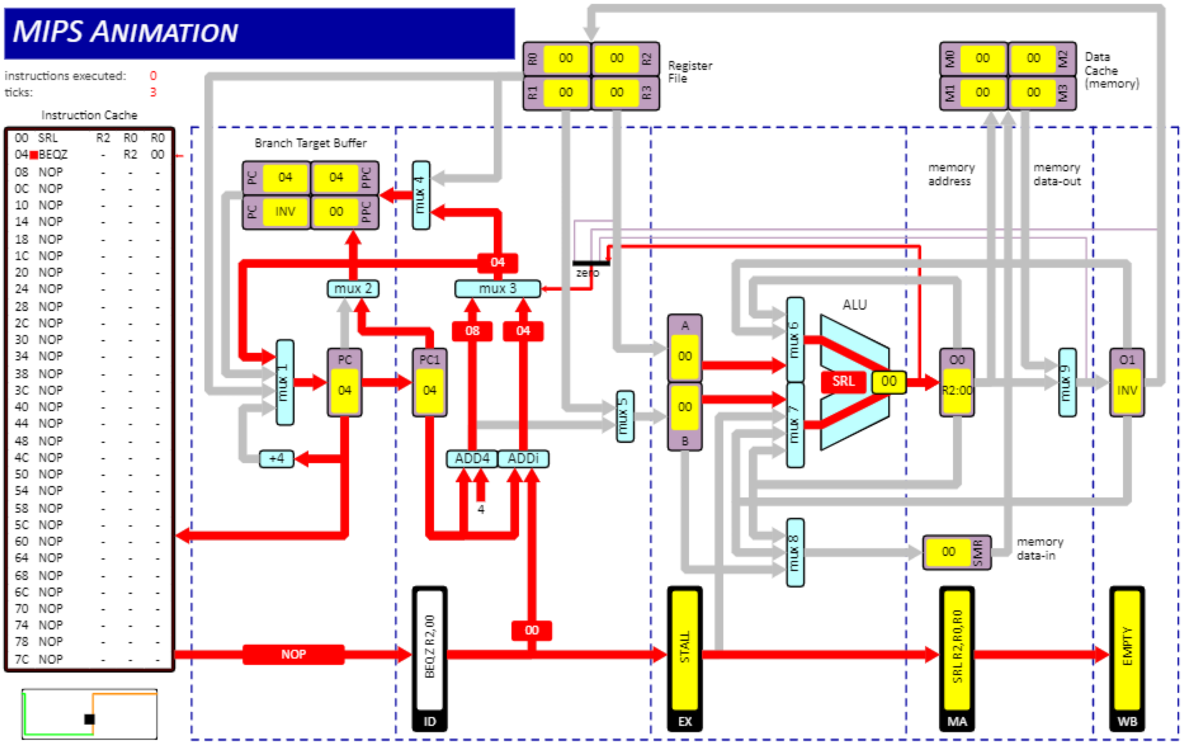
instructions executed: 0
ticks: 4

Instruction Cache

```
00 LD   R2 R0 00
04 NOP  -  -  -
08 NOP  -  -  -
0C NOP  -  -  -
10 NOP  -  -  -
14 NOP  -  -  -
18 NOP  -  -  -
1C NOP  -  -  -
20 NOP  -  -  -
24 NOP  -  -  -
28 NOP  -  -  -
2C NOP  -  -  -
30 NOP  -  -  -
34 NOP  -  -  -
38 NOP  -  -  -
3C NOP  -  -  -
40 NOP  -  -  -
44 NOP  -  -  -
48 NOP  -  -  -
4C NOP  -  -  -
50 NOP  -  -  -
54 NOP  -  -  -
58 NOP  -  -  -
5C NOP  -  -  -
60 NOP  -  -  -
64 NOP  -  -  -
68 NOP  -  -  -
6C NOP  -  -  -
70 NOP  -  -  -
74 NOP  -  -  -
78 NOP  -  -  -
7C NOP  -  -  -
```

LD R2, R0, 00

f)



MIPS ANIMATION

instructions executed: 0
ticks: 3

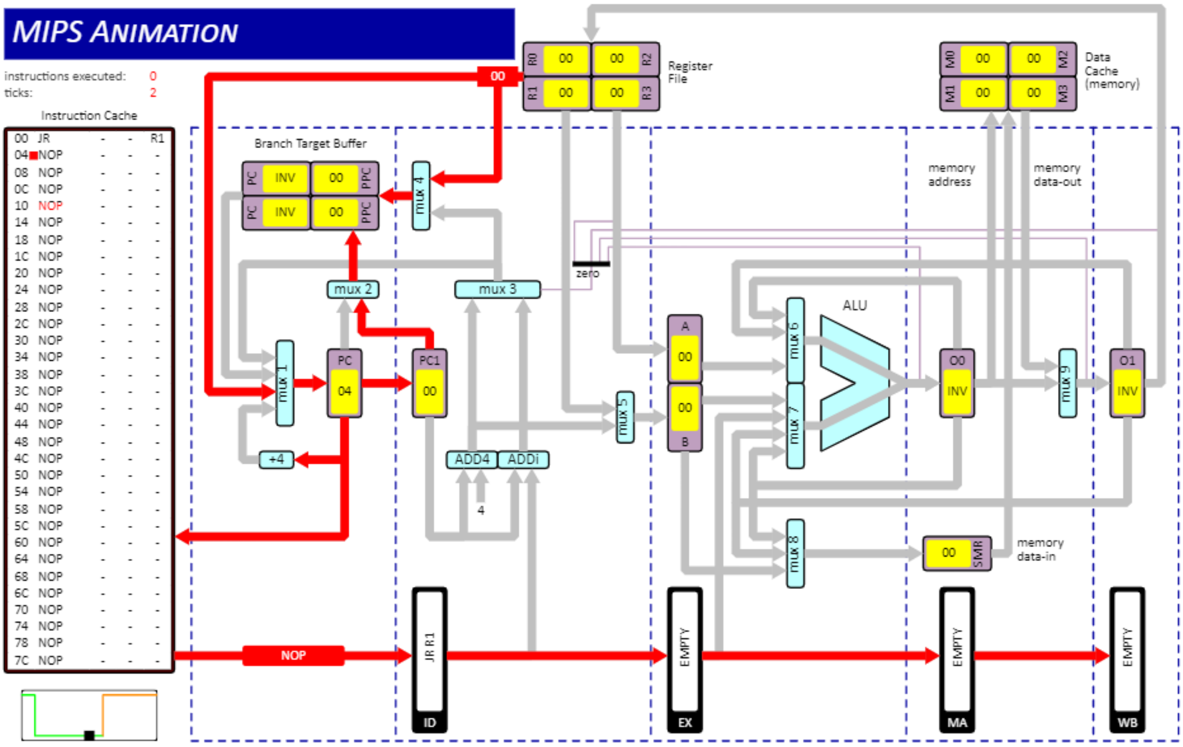
Instruction Cache

```
00  SRL    R2  R0  R0
04  BEQZ   -   R2  00
08  NOP    -   -   -
0C  NOP    -   -   -
10  NOP    -   -   -
14  NOP    -   -   -
18  NOP    -   -   -
1C  NOP    -   -   -
20  NOP    -   -   -
24  NOP    -   -   -
28  NOP    -   -   -
2C  NOP    -   -   -
30  NOP    -   -   -
34  NOP    -   -   -
38  NOP    -   -   -
3C  NOP    -   -   -
40  NOP    -   -   -
44  NOP    -   -   -
48  NOP    -   -   -
4C  NOP    -   -   -
50  NOP    -   -   -
54  NOP    -   -   -
58  NOP    -   -   -
5C  NOP    -   -   -
60  NOP    -   -   -
64  NOP    -   -   -
68  NOP    -   -   -
6C  NOP    -   -   -
70  NOP    -   -   -
74  NOP    -   -   -
78  NOP    -   -   -
7C  NOP    -   -   -
```

SRL R2, R0, R0

BEQZ R2, 00

g)



MIPS ANIMATION

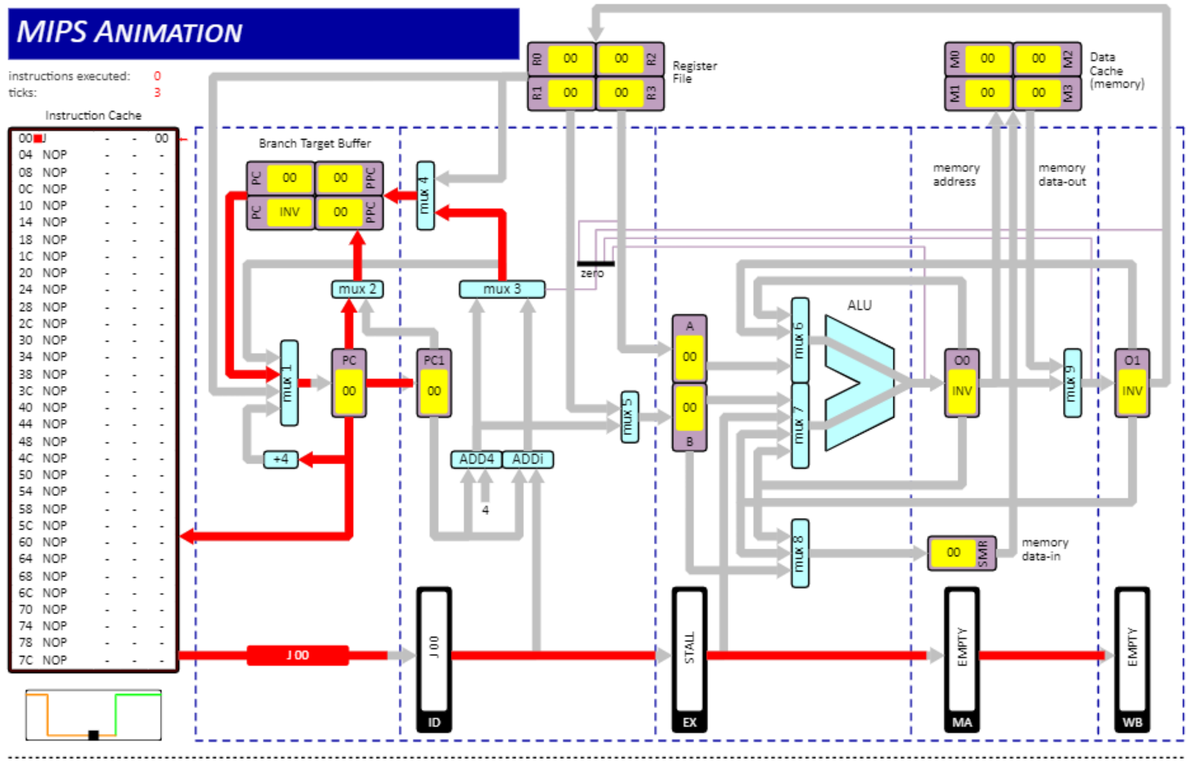
instructions executed: 0
ticks: 2

Instruction Cache

```
00  JR     -   -   R1
04  NOP    -   -   -
08  NOP    -   -   -
0C  NOP    -   -   -
10  NOP    -   -   -
14  NOP    -   -   -
18  NOP    -   -   -
1C  NOP    -   -   -
20  NOP    -   -   -
24  NOP    -   -   -
28  NOP    -   -   -
2C  NOP    -   -   -
30  NOP    -   -   -
34  NOP    -   -   -
38  NOP    -   -   -
3C  NOP    -   -   -
40  NOP    -   -   -
44  NOP    -   -   -
48  NOP    -   -   -
4C  NOP    -   -   -
50  NOP    -   -   -
54  NOP    -   -   -
58  NOP    -   -   -
5C  NOP    -   -   -
60  NOP    -   -   -
64  NOP    -   -   -
68  NOP    -   -   -
6C  NOP    -   -   -
70  NOP    -   -   -
74  NOP    -   -   -
78  NOP    -   -   -
7C  NOP    -   -   -
```

JR R1

h)



```
00 J    -  -  00
04 NOP  -  -  -
08 NOP  -  -  -
0C NOP  -  -  -
10 NOP  -  -  -
14 NOP  -  -  -
18 NOP  -  -  -
1C NOP  -  -  -
20 NOP  -  -  -
24 NOP  -  -  -
28 NOP  -  -  -
2C NOP  -  -  -
30 NOP  -  -  -
34 NOP  -  -  -
38 NOP  -  -  -
3C NOP  -  -  -
40 NOP  -  -  -
44 NOP  -  -  -
48 NOP  -  -  -
4C NOP  -  -  -
50 NOP  -  -  -
54 NOP  -  -  -
58 NOP  -  -  -
5C NOP  -  -  -
60 NOP  -  -  -
64 NOP  -  -  -
68 NOP  -  -  -
6C NOP  -  -  -
70 NOP  -  -  -
74 NOP  -  -  -
78 NOP  -  -  -
7C NOP  -  -  -
```

J 00

Q2.

(i)

R1 result = 0x15

Clock cycles = 10

(ii)

R1 result = 0x15

Clock cycles = 18

(iii)

R1 result = 0x06

Clock cycles = 10

In (i) the instruction is executed and the intermediate result is passed back to the ALU through the immediate registers for the next instruction while also being sent to the register file. In (ii) pipeline stalls occur because the next instruction is waiting for the answer from the previous instruction due to data dependency. In (iii) there is no data dependency or immediate registers to store results from previous instructions so the instructions occur as independent operations. The instructions keep reading values from the registers that are not yet updated.

Q3.

(i)

Instructions executed: 38

Clock cycles: 50

The pipeline takes the first four clock cycles to execute the 1st instruction. After this, 8 pipeline stalls occur. These stalls occur mainly in relation to all calls of the LD instruction (3 times) as this is a memory access instruction. Stalls also occur for SLLi (2 times), ANDi (2 times) and J (1 time).

(ii)

Instructions executed: 38

Clock cycles: 53

The pipeline takes the first four clock cycles to execute the 1st instruction. 10 pipeline stalls occur. These stalls occur mainly in relation to all calls of the LD instruction (4 times) as this is a memory access instruction. Stalls also occur for all calls of SLLi (4 times), one call of ANDi and one call of J. The lack of branch prediction is the cause for an increase in pipeline stalls, especially in relation to ANDi.

(iii)

Instructions executed: 38

Clock cycles: 46

The pipeline takes the first four clock cycles to execute the 1st instruction. 4 pipeline stalls occur. The stalls occur mainly in relation to calls of ANDi (2 times), SRLi (1 time) and J (1 time). The stalls for ANDi relate to the following instruction BEQZ checking the value of R2 immediately after ANDi operates on R2, mean BEQZ must wait for the result to be stored in R2. The stalls are fewer because having SLL occur first allows for the value of R2 to be updated for SRL as SLL operates on R3.