RISC-V Processor CircuiTi*k*Z Library

March 12, 2025

1 Components

1.1 Instruction Memory

a ← A   RD → rd
Instruction
Memory
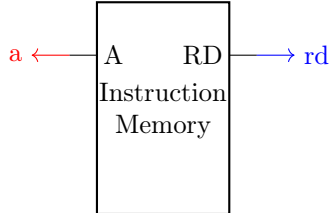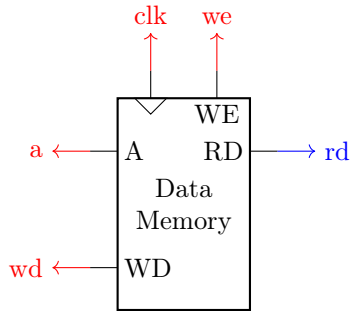
```
1  \begin{circuitikz}
2    \node[instrmem, align=center] (comp) {Instruction
        \\Memory};
3
4    \draw[->, red] (comp.a) -- ++(-.5, 0) node[left] {
        a};
5    \draw[->, blue] (comp.rd) -- ++(.5, 0) node[right]
        {rd};
6  \end{circuitikz}
```

1.2 Data Memory

clk   we

WE
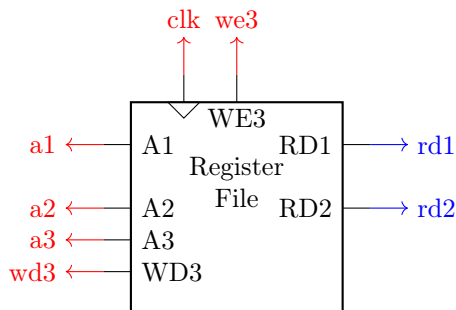a ← A   RD → rd
Data
Memory
wd ← WD

```
1  \begin{circuitikz}[]
2    \node[datamem, align=center] (comp) {Data\\Memory
        };
3    \draw[->, red] (comp.a) -- ++(-.5, 0) node[left]
        {a};
4    \draw[->, red] (comp.wd) -- ++(-.5, 0) node[left]
        {wd};
5    \draw[->, red] (comp.clk) -- ++(0, .5) node[above
        ] {clk};
6    \draw[->, red] (comp.we) -- ++(0, .5) node[above]
        {we};
7    \draw[->, blue] (comp.rd) -- ++(.5, 0) node[right
        ] {rd};
8  \end{circuitikz}
```

1.3 Register File

clk we3

WE3
a1 ← A1   RD1 → rd1
        Register
a2 ← A2   File RD2 → rd2
a3 ← A3
wd3 ← WD3
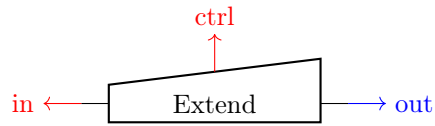
```
1    \begin{circuitikz}[]
2      \node[regfile, align=center] (comp) {
          Register\\File};
3      \draw[->, red] (comp.a1) -- ++(-.5, 0)
          node[left] {a1};
4      \draw[->, red] (comp.a2) -- ++(-.5, 0)
          node[left] {a2};
5      \draw[->, red] (comp.a3) -- ++(-.5, 0)
          node[left] {a3};
6      \draw[->, red] (comp.wd3) -- ++(-.5, 0)
          node[left] {wd3};
7
8      \draw[->, red] (comp.clk) -- ++(0, .5)
          node[above] {clk};
9      \draw[->, red] (comp.we3) -- ++(0, .5)
          node[above] {we3};
10     \draw[->, blue] (comp.rd1) -- ++(.5, 0)
          node[right] {rd1};
11     \draw[->, blue] (comp.rd2) -- ++(.5, 0)
          node[right] {rd2};
12   \end{circuitikz}
```
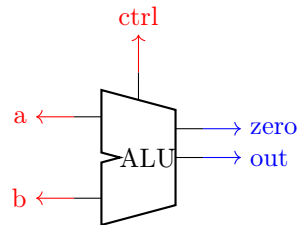
2

1.4 Extend Unit

```
1   \begin{circuitikz}[]
2     \node[extend, align=center] (comp) {Extend
        };
3     \draw[->, red] (comp.in) -- ++(-.5, 0)
        node[left] {in};
4     \draw[->, red] (comp.ctrl) -- ++(0, .5)
        node[above] {ctrl};
5     \draw[->, blue] (comp.out) -- ++(.5, 0)
        node[right] {out};
6   \end{circuitikz}
```
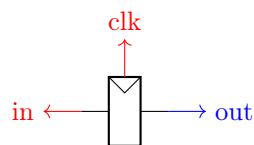
1.5 Arithmetic Logic Unit

```
1   \begin{circuitikz}[]
2     \node[alu, align=center] (comp) {ALU};
3     \draw[->, red] (comp.a) -- ++(-.5, 0) node[left] {a};
4     \draw[->, red] (comp.b) -- ++(-.5, 0) node[left] {b};
5     \draw[->, red] (comp.ctrl) -- ++(0, .5) node[above] {
        ctrl};
6     \draw[->, blue] (comp.out) -- ++(.5, 0) node[right] {
        out};
7     \draw[->, blue] (comp.zero) -- ++(.5, 0) node[right]
        {zero};
8   \end{circuitikz}
```
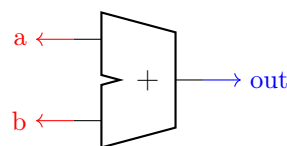
1.6 Register

```
1   \begin{circuitikz}[]
2     \node[reg, align=center] (comp) {};
3     \draw[->, red] (comp.in) -- ++(-.5, 0) node[left] {in};
4     \draw[->, red] (comp.clk) -- ++(0, .5) node[above] {clk
        };
5     %\draw[->, red] (comp.en) -- ++(0, -.5) node[below] {en
        };
6     \draw[->, blue] (comp.out) -- ++(.5, 0) node[right] {out
        };
7   \end{circuitikz}
```
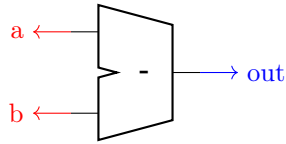
1.7 Adder

```
1   \begin{circuitikz}[]
2     \node[adder, align=center] (comp) {};
3     \draw[->, red] (comp.a) -- ++(-.5, 0) node[left] {a};
4     \draw[->, red] (comp.b) -- ++(-.5, 0) node[left] {b};
5     \draw[->, blue] (comp.out) -- ++(.5, 0) node[right] {
        out};
6   \end{circuitikz}
```
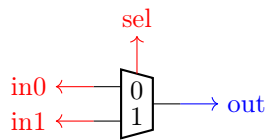
3

1.8   Subtractor

```
1    \begin{circuitikz}[]
2      \node[sub, align=center] (comp) {};
3      \draw[->, red] (comp.a) -- ++(-.5, 0) node[left] {a};
4      \draw[->, red] (comp.b) -- ++(-.5, 0) node[left] {b};
5      \draw[->, blue] (comp.out) -- ++(.5, 0) node[right] {
           out};
6    \end{circuitikz}
```

1.9   Multiplexer

```
1    \begin{circuitikz}[]
2      \node[mux, align=center] (comp) {};
3      \draw[->, red] (comp.in0) -- ++(-.5, 0) node[left] {in
           0};
4      \draw[->, red] (comp.in1) -- ++(-.5, 0) node[left] {in
           1};
5      \draw[->, red] (comp.sel) -- ++(0, .5) node[above] {sel
           };
6      \draw[->, blue] (comp.out) -- ++(.5, 0) node[right] {
           out};
7    \end{circuitikz}
```

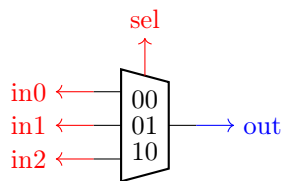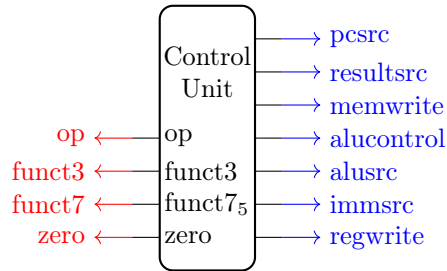## 1.10   Multiplexer with 3 inputs

```
1    \begin{circuitikz}[]
2      \node[3mux, align=center] (comp) {};
3      \draw[->, red] (comp.in0) -- ++(-.5, 0) node[left] {in
           0};
4      \draw[->, red] (comp.in1) -- ++(-.5, 0) node[left] {in
           1};
5      \draw[->, red] (comp.in2) -- ++(-.5, 0) node[left] {in
           2};
6      \draw[->, red] (comp.sel) -- ++(0, .5) node[above] {
           sel};
7      \draw[->, blue] (comp.out) -- ++(.5, 0) node[right] {
           out};
8    \end{circuitikz}
```

1.11   Single-Cycle Control Unit
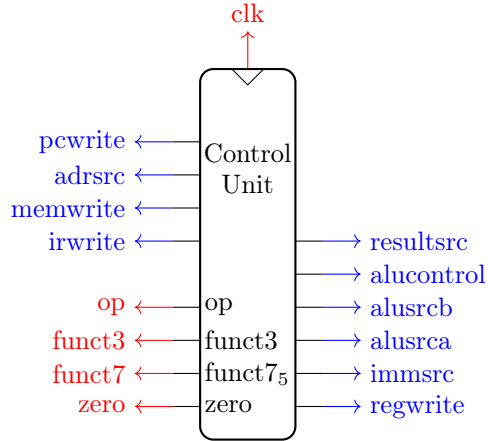
```
1   \begin{circuitikz}[]
2       \node[ctrlunitsc, align=center] (comp) {
            Control\\Unit};
3       \draw[->, red] (comp.op) -- ++(-.5, 0)
            node[left] {op};
4       \draw[->, red] (comp.funct3) -- ++(-.5,
            0) node[left] {funct3};
5       \draw[->, red] (comp.funct7) -- ++(-.5,
            0) node[left] {funct7};
6       \draw[->, red] (comp.zero) -- ++(-.5, 0)
            node[left] {zero};
7
8       \draw[->, blue] (comp.pcsrc) -- ++(.5, 0)
            node[right] {pcsrc};
9       \draw[->, blue] (comp.resultsrc) --
            ++(.5, 0) node[right] {resultsrc};
10      \draw[->, blue] (comp.memwrite) -- ++(.5,
            0) node[right] {memwrite};
11      \draw[->, blue] (comp.alucontrol) --
            ++(.5, 0) node[right] {alucontrol};
12      \draw[->, blue] (comp.alusrc) -- ++(.5,
            0) node[right] {alusrc};
13      \draw[->, blue] (comp.immsrc) -- ++(.5,
            0) node[right] {immsrc};
14      \draw[->, blue] (comp.regwrite) -- ++(.5,
            0) node[right] {regwrite};
15  \end{circuitikz}
```

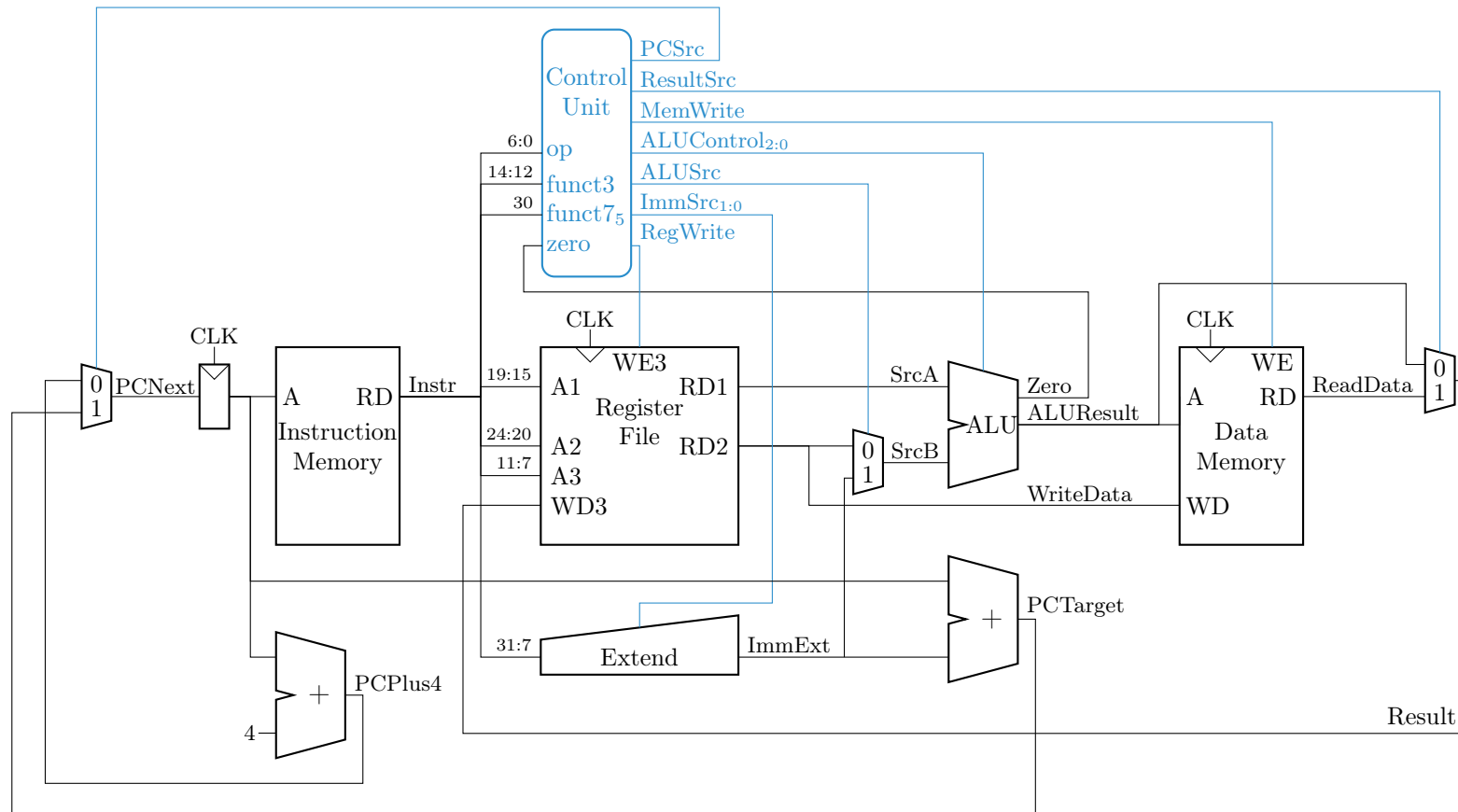## 1.12 Multi-Cycle Control Unit

```
1    \begin{circuitikz}[]
2      \node[ctrlunitmc, align=center] (comp)
           {Control\\Unit};
3      \draw[->, red] (comp.op) -- ++(-.5, 0)
           node[left] {op};
4      \draw[->, red] (comp.funct3) --
           ++(-.5, 0) node[left] {funct3};
5      \draw[->, red] (comp.funct7) --
           ++(-.5, 0) node[left] {funct7};
6      \draw[->, red] (comp.zero) -- ++(-.5,
           0) node[left] {zero};
7      \draw[->, red] (comp.clk) -- ++(0,.5)
           node[above] {clk};
8
9      \draw[->, blue] (comp.resultsrc) --
           ++(.5, 0) node[right] {resultsrc};
10     \draw[->, blue] (comp.memwrite) --
           ++(-.5, 0) node[left] {memwrite};
11     \draw[->, blue] (comp.alucontrol) --
           ++(.5, 0) node[right] {alucontrol
           };
12     \draw[->, blue] (comp.alusrca) --
           ++(.5, 0) node[right] {alusrca};
13     \draw[->, blue] (comp.alusrcb) --
           ++(.5, 0) node[right] {alusrcb};
14     \draw[->, blue] (comp.immsrc) --
           ++(.5, 0) node[right] {immsrc};
15     \draw[->, blue] (comp.regwrite) --
           ++(.5, 0) node[right] {regwrite};
16     \draw[->, blue] (comp.irwrite) --
           ++(-.5, 0) node[left] {irwrite};
17     \draw[->, blue] (comp.adrsrc) --
           ++(-.5, 0) node[left] {adrsrc};
18     \draw[->, blue] (comp.pcwrite) --
           ++(-.5, 0) node[left] {pcwrite};
19   \end{circuitikz}
```

2 Single-Cycle RISC-V Processor

3 Multi-Cycle RISC-V Processor