

The Lab of Computer Organizationn

5-Basic logic component

Teacher: Xiaohong Jiang

Ta: Zhenyu Ye

Refer to Zonghui Wang' PPT

Contact Ta: 13732217324 / 517663

enyzy@163.com

Outline

- Experiment Purpose
- Experiment Task
- Basic Principle
- Operating Procedures

Experiment Purpose

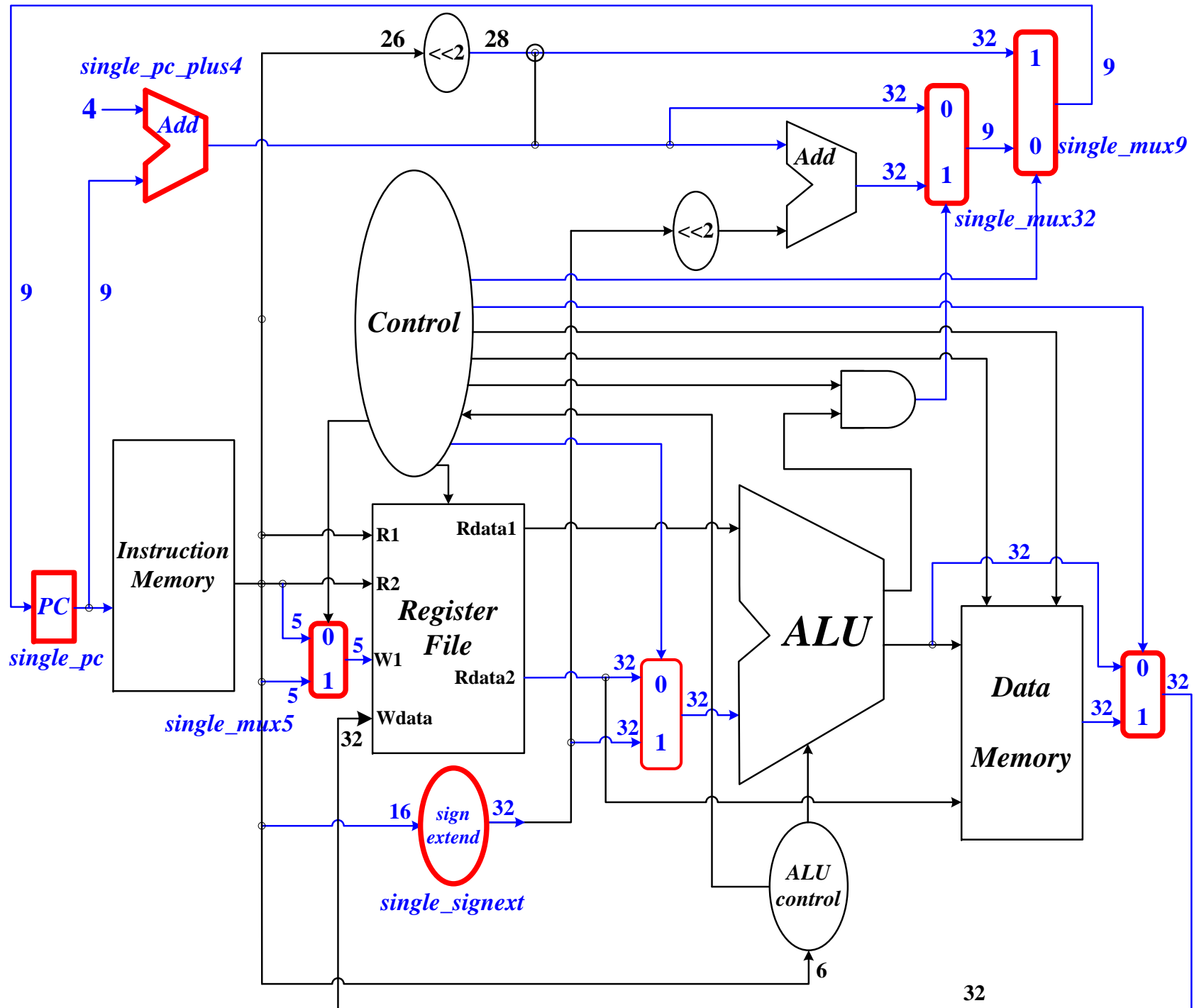
- Familiar with the design method of the basic logic components, and master the working principle and design method of the basic components.
- Master the design of the PC program counter and Implementation of $PC + 4$.
- Master the design of sign extension module.
- Master the design of a simple adder.

Experiment Task

- Implementation of Single PC Module (perform Waveform Simulation).
- Implementation of Single Mux Module (FPGA verification).
- Implementation of Sign Extension Module (perform Waveform Simulation).
- Implementation of Simple Adder Module (perform Waveform Simulation).

Basic Principle

- Program Counter
- Mux
- Sign extension
- Simple adder


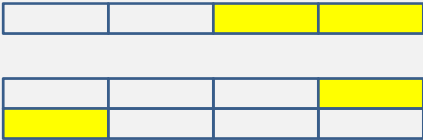



Program Counter

- Program counter PC, used to control the machine instructions' implementation order , is essentially the instruction address in the memory.
- According to the PC, the current CPU reads instructions to execute, and change the value of PC for each reading.
- Change of the value of PC has deal with the internal structure of memory:
 - If the memory data width is 8 bits, and instruction length is 32-bit: Read an instruction, $PC \leftarrow PC + 4$
 - If the memory data width is 32 bits, and instruction length is 32-bit: Read an instruction, $PC \leftarrow PC + 1$

Address Alignment

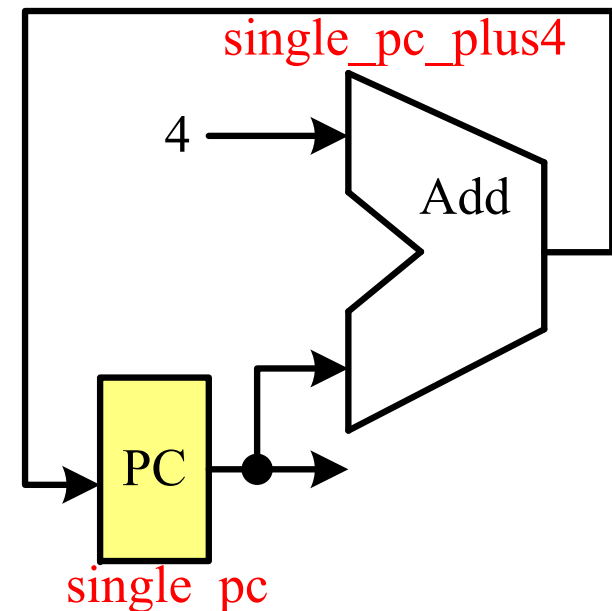
- Instructions memory: 32bit \times 512, address width[$\log_2 512$] = 9bits.
- Read and write analysis of 32bit memory:

	Address feature	Instruction addressing		Data addressing		legend
		read	write	read	write	
word (alignment)	Last 2 bits are 0	Read once	N.A.	Read once	Write once	
Half word	Last 1 bit is 0	N.A.	N.A.	Read 1~2 times/ compute Additionally	Write 1~2times/ Compute Additionally	
byte	Last bit is unsure	N.A.	N.A.	Read 1 time/ compute Additionally	Write 1 time/ Compute Additionally	

PC Verilog Code

```
module single_pc(clk, rst, i_pc, o_pc)
    input wire clk, rst;
    input wire [8:0] i_pc;
    output reg [8:0] o_pc;
    always @(posedge clk)
        assign o_pc =
endmodule
```

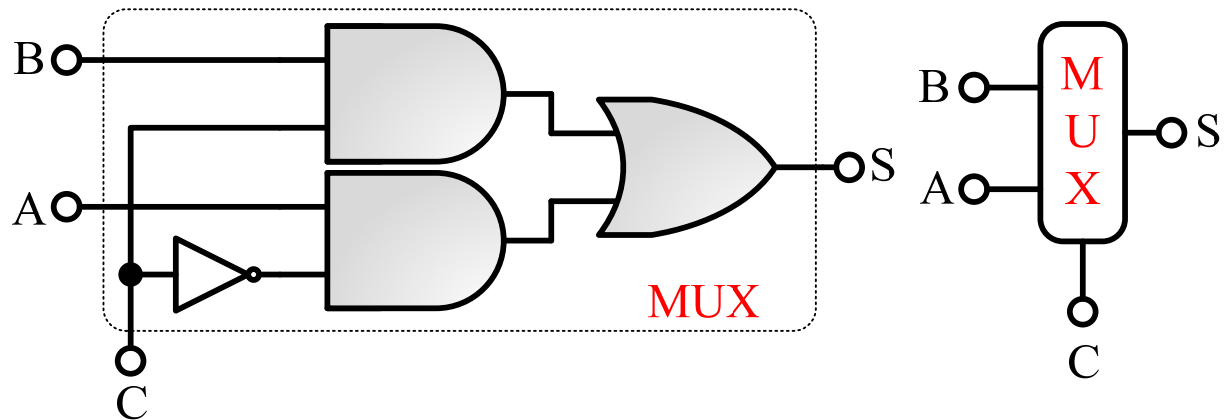
```
module single_pc_plus4(i_pc, o_pc);
    input wire [8:0] i_pc;
    output wire [8:0] o_pc;
    assign o_pc =
endmodule
```



Mux

- Multiplexer output is the selected one from multiple input values by a control signal.
- CPU design uses these selectors:
 - 1-bit, 2-bit, 5-bit, 9-bit, 32-bit

C	S
0	A
1	B

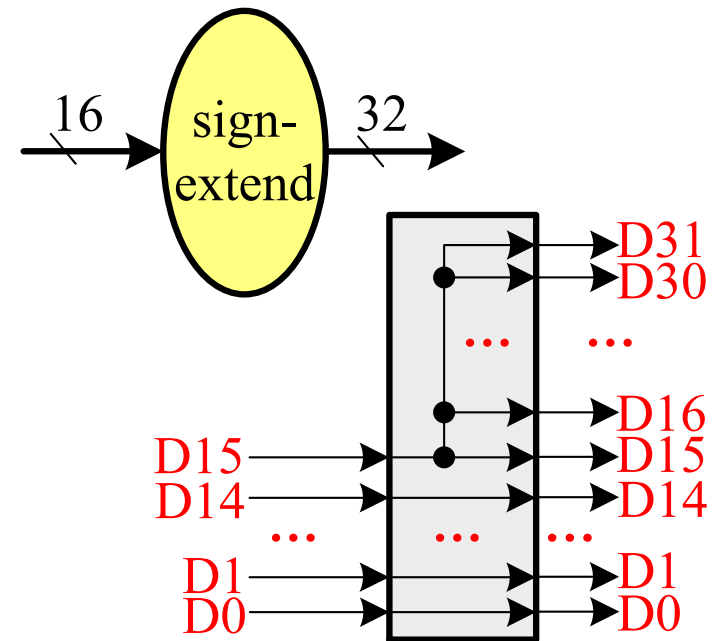


Mux Verilog Code

```
module single_mux (A, B, C, S);  
    paramter N = 1;    // N: 1, 2, 5, 9, 32  
    input  wire [N-1:0] A, B;  
    input  wire          C;  
    output wire [N-1:0] S;  
    assign S =  
ndmodule
```

Sign Extension

- By complement rules, sign extension: If the 16th digit is 1, the extended bit is 1, otherwise 0.
- It is mainly used in the CPU design for extending the 16-bit input data to 32-bit data as output



Sign Extension Verilog Code

```
module single_signext(i_16, o_32);  
    input wire [15:0] i_16;  
    output reg [31:0] o_32;  
    always @()  
        o_32 <=  
endmodule
```

Simple Adder Verilog Code

```
module single_add (i_op1, i_op2, o_out);  
    parameter N = 32;  
    input  wire[N-1:0] i_op1, i_op2;  
    output wire[N-1:0] o_out;  
    assign o_out =  
endmodule
```

Experiment Content

- PC: Perform Waveform Simulation.
- Mux: FPGA Verification:
8 switches divided into two groups of 4, with a button for selecting control: display the four digits selected with a digital tube and four LED.
- Sign Extension: Perform Waveform Simulation.
- Single Adder: Perform Waveform Simulation.

Precaution

- Simulation for PC module

```
module pc_top(clk, rst, i_pc, o_pc)
```

```
...
```

```
    single_pc M1(clk, rst, i_pc, o_pc);
```

```
    single_pc_plus4 M2(o_pc, i_pc);
```

```
end module
```

```
perform waveform simulation for pc_top.
```

- 16th bit of Sign Extension inputs should include 0 and 1;