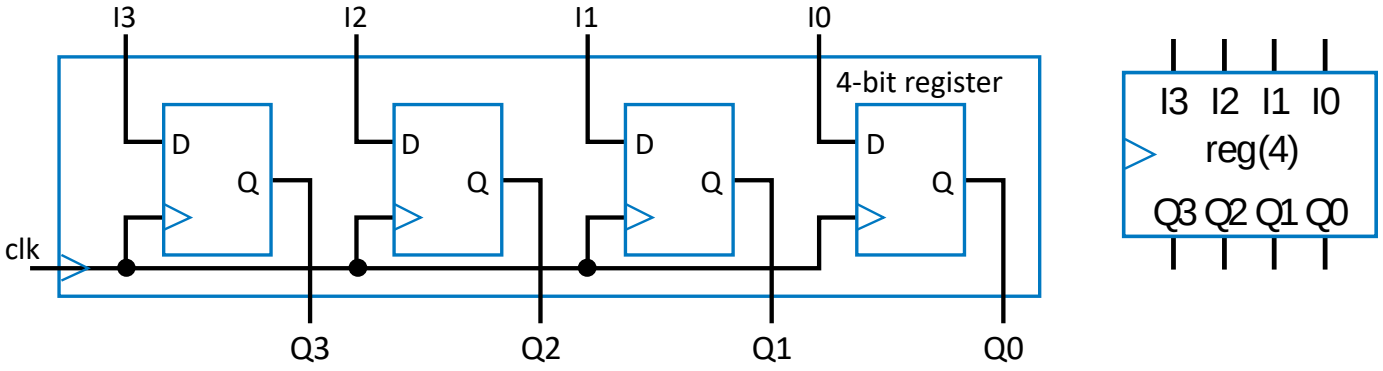
VE270 Final Exam Review
Register, Shifter, Memroy, PLD

Shi Li

2019.12.10

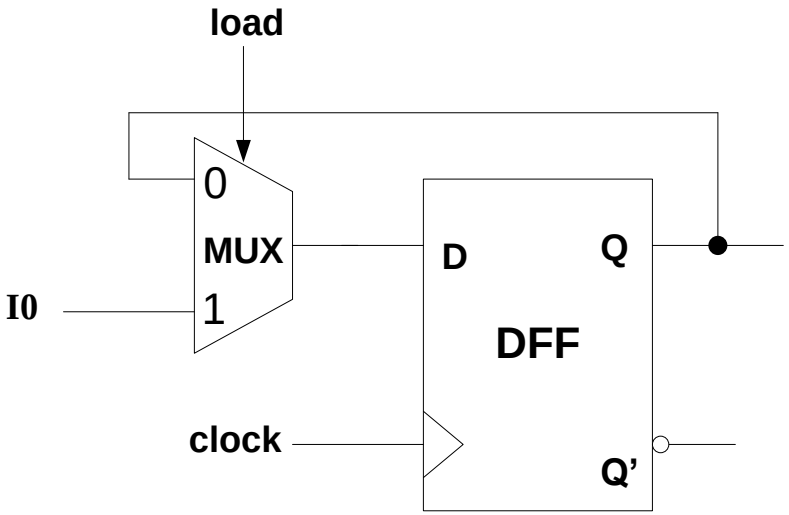
1. Register
Basic register

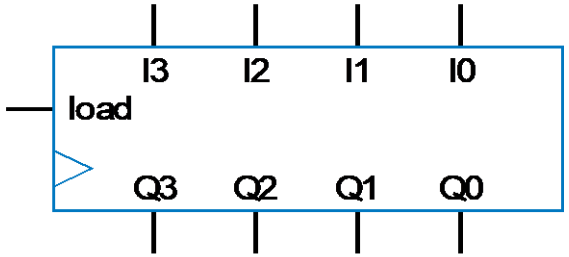
- Used to store data.



1. Register
Register with load

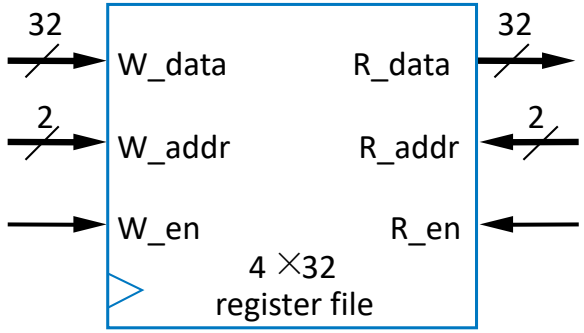
- Add a synchronous load signal.

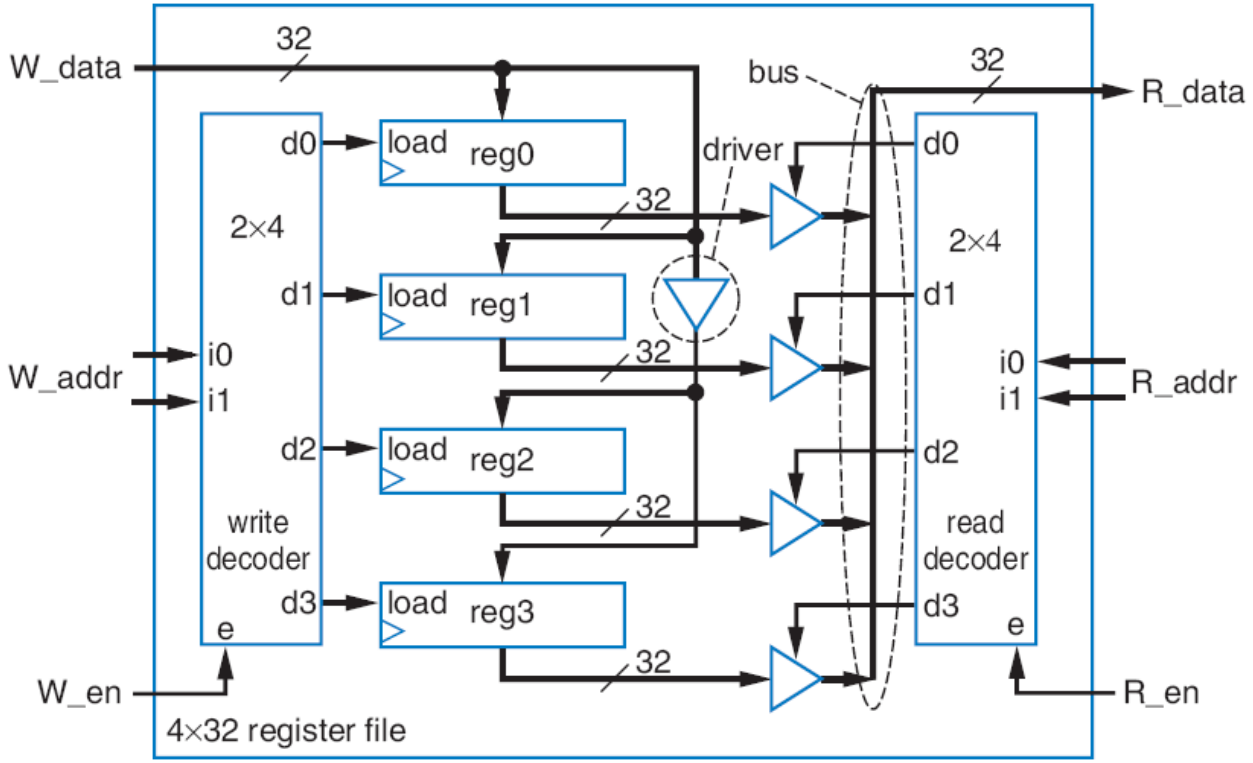




1. Register
Register file

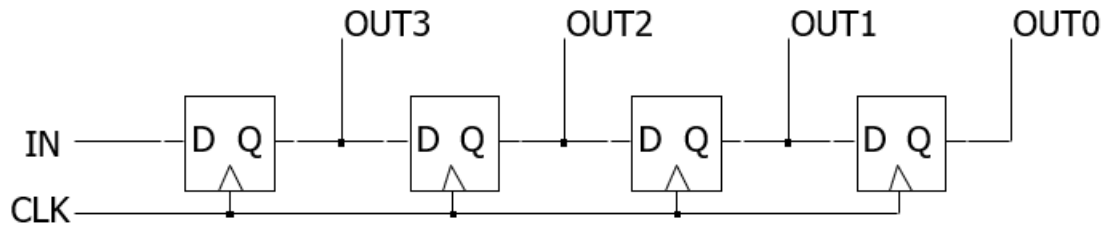
- Used to store a lot of data.



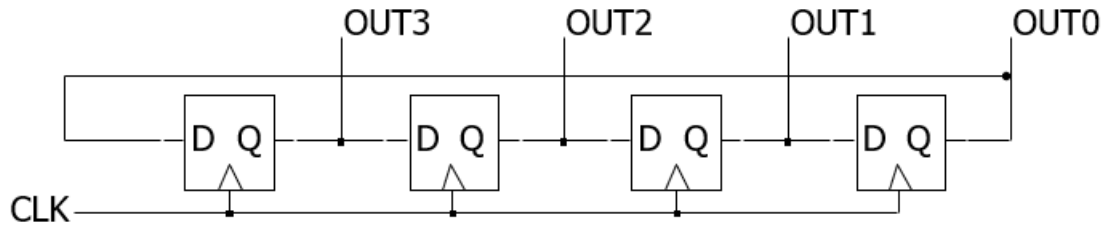


1. Register
Shift register (not shifter) & Rotate register

- Shift or rotate, every clock cycle.



	IN	OUT(3:0)
Initial value:	0	0110
rising edge:	0	0011
rising edge:	0	0001
rising edge:	0	0000
rising edge:	1	1000
rising edge:	0	0100

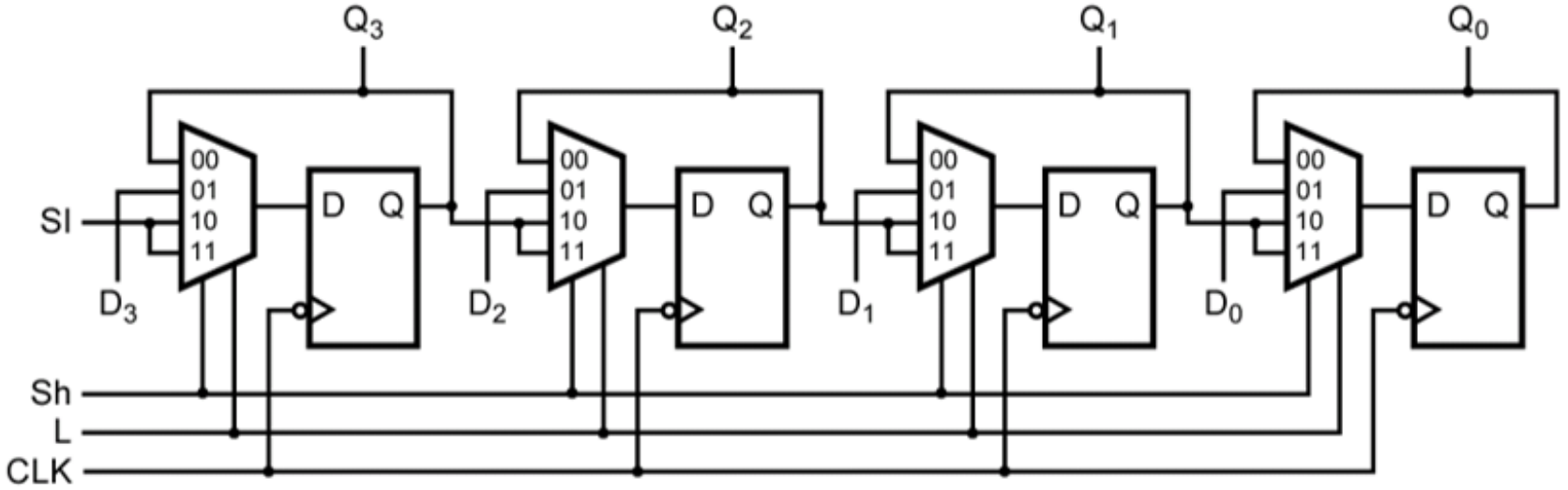


	OUT(3:0)
Initial value:	0110
rising edge:	0011
rising edge:	1001
rising edge:	1100
rising edge:	0110
rising edge:	0011

1. Register
Universal shift register

- Shift or rotate, every clock cycle.
- Exercise: Write Verilog code for it.

Inputs		Action
Sh (Shift)	L (Load)	
0	0	no change
0	1	load
1	X	Shift Right



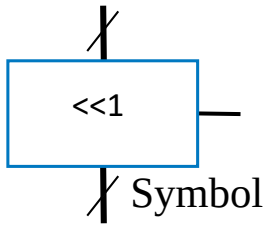
1. Register
Exercise Solution

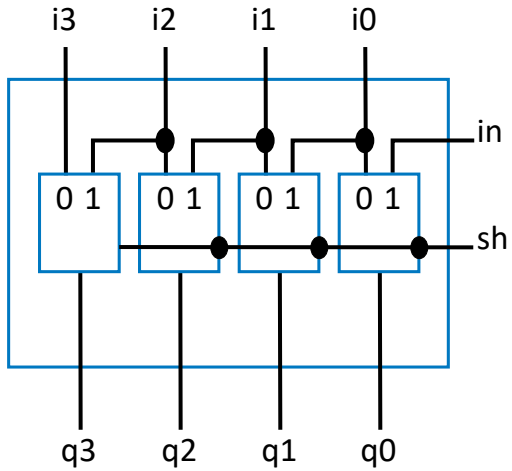
```
module Universal_Shift_Reg (D, Q, SI, Sh, L, Clk);
  input  SI, Sh, L, Clk;
  input  [3:0] D;
  output [3:0] Q;
  reg    [3:0] Q;

  always @ (posedge clock)
  begin
    if (Sh == 1)
    begin
      Q[2:0] <= Q[3:1];
      Q[3]   <= SI;
    end
    else if (L == 1)
      Q <= D;
  end
endmodule
```

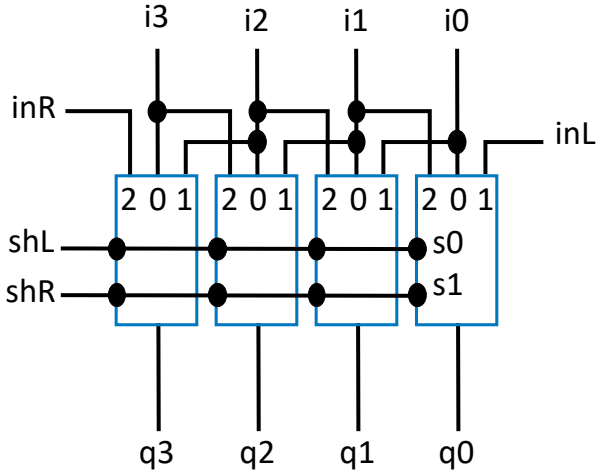
2. Shifter (not shift register)
Basic shifter

- Combinational component, not sequential! (do not depend on Clk)
- Shift left once = multiply by 2
- Shift right once = divide by 2





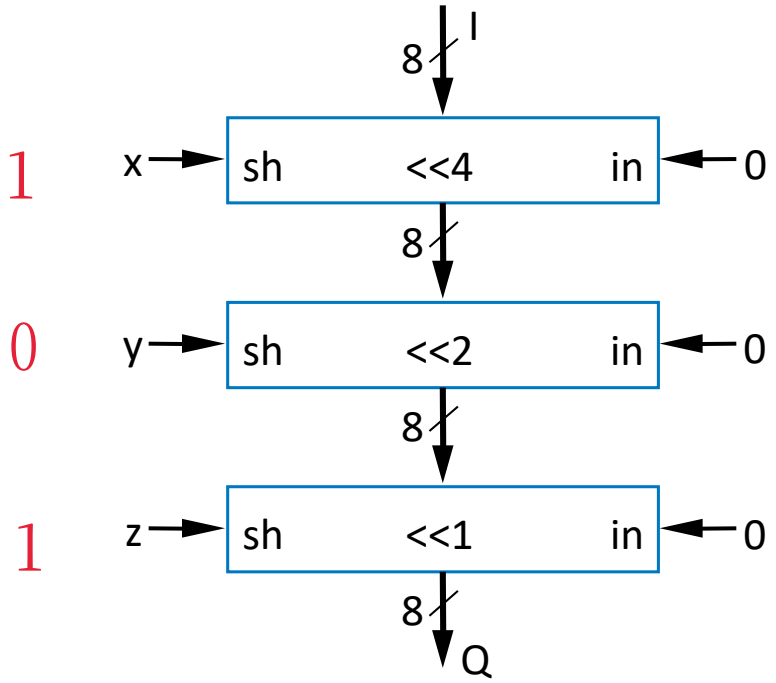
Shifter with left shift or no shift



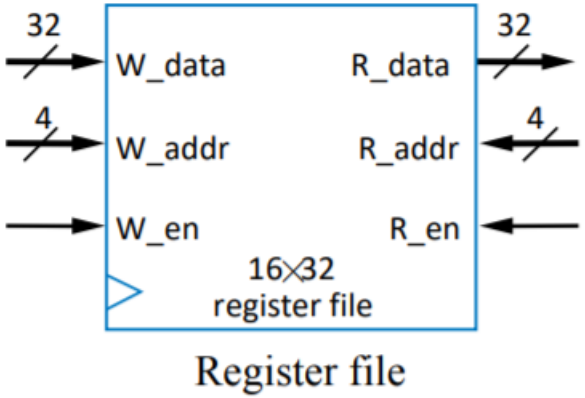
Shifter with left shift, right shift, and no shift

2. Shifter (not shift register)
Bigger shifter

- How to use bigger shifter to shift by any amount?
- Suppose the input is 11111111, how to set xyz so that the output will be 11100000?

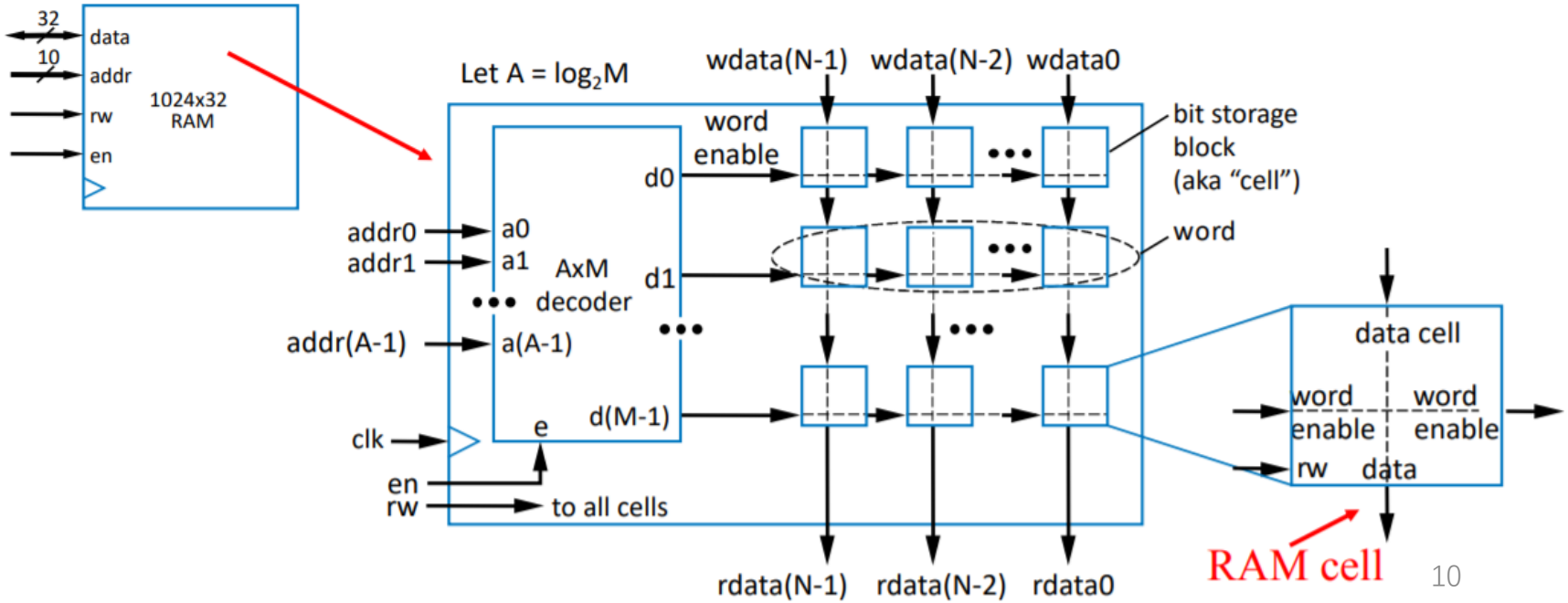


3. Memory Concept



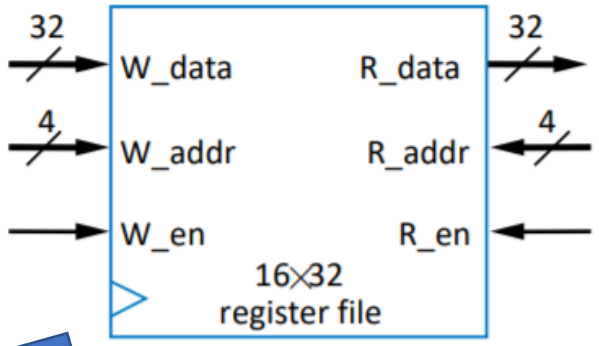
Register file

- Random Access Memory (RAM)
  - Similarity to register file: read/write by address
  - Difference to register file: RAM is larger, slower, implemented on a separate device

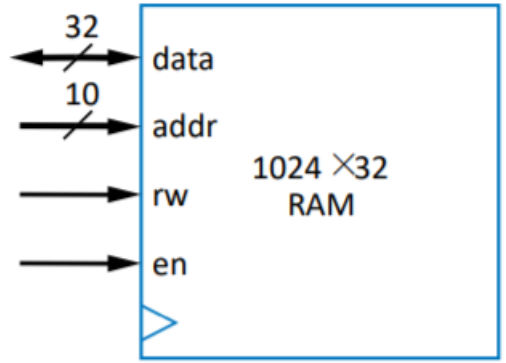


3. Memory
Concept: Random Access Memory (RAM)

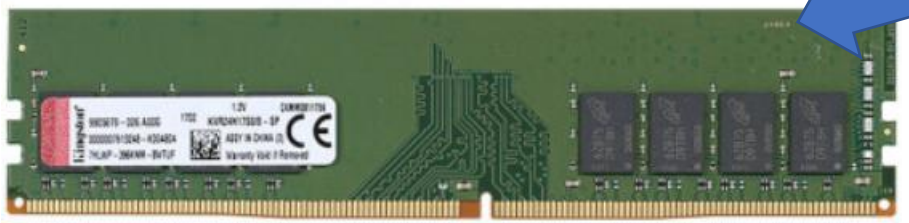
- Similarity to register file: read/write by address
- Difference to register file: RAM is larger, slower, implemented on a separate device



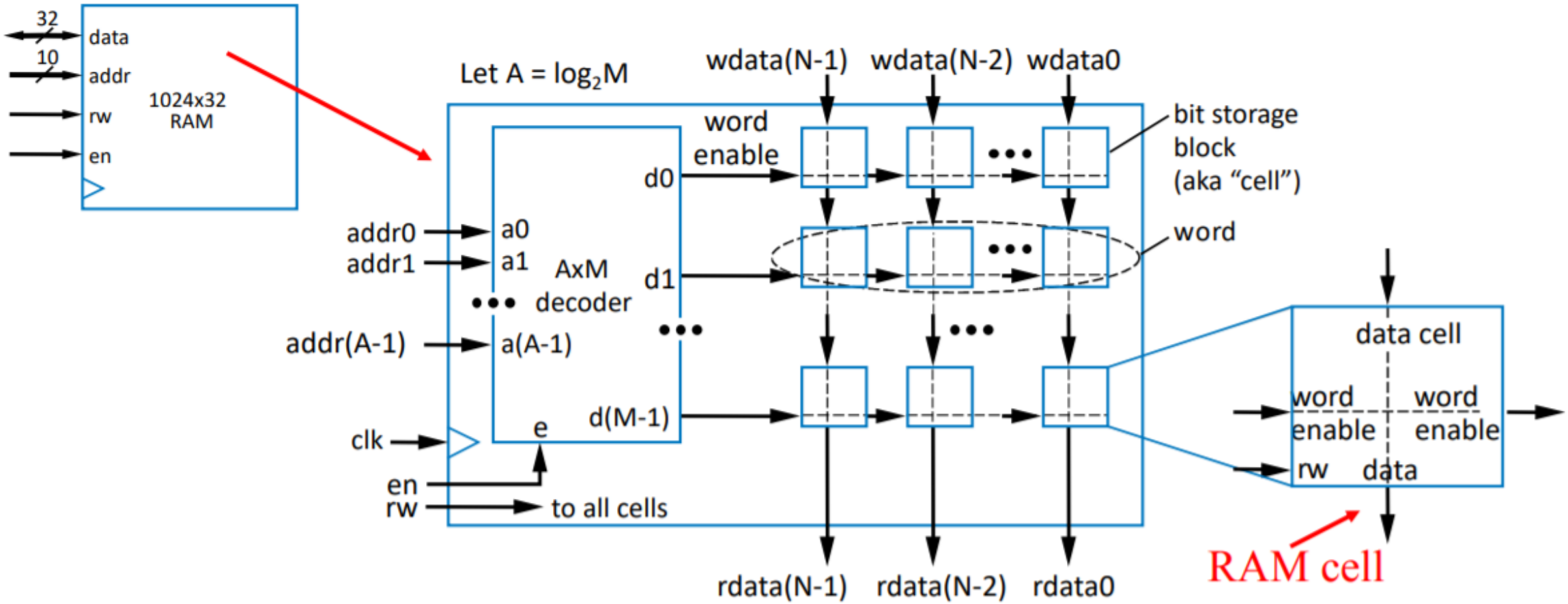
Register file



RAM block symbol

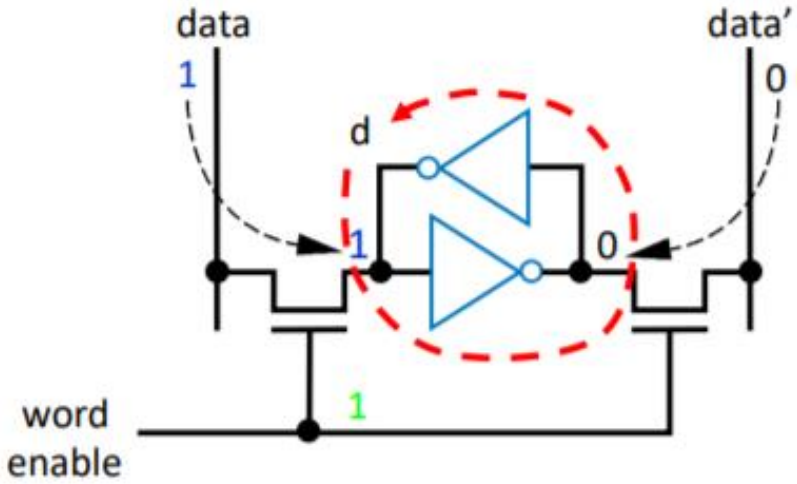


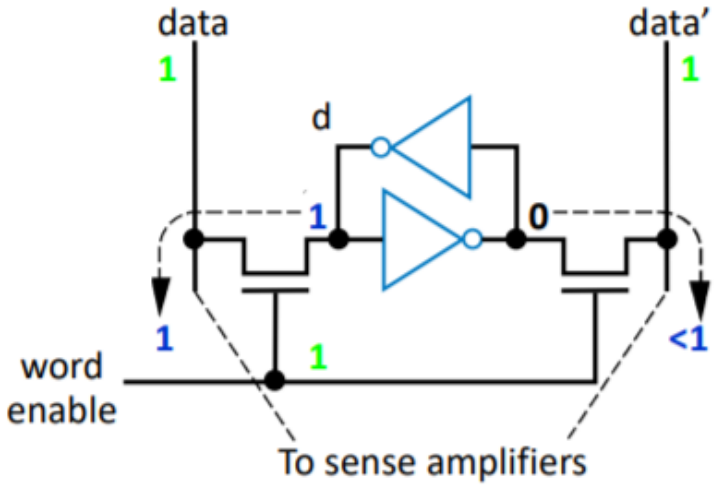
3. Memory
Concept: Random Access Memory (RAM)



3. Memory
Concept: Static RAM (SRAM)

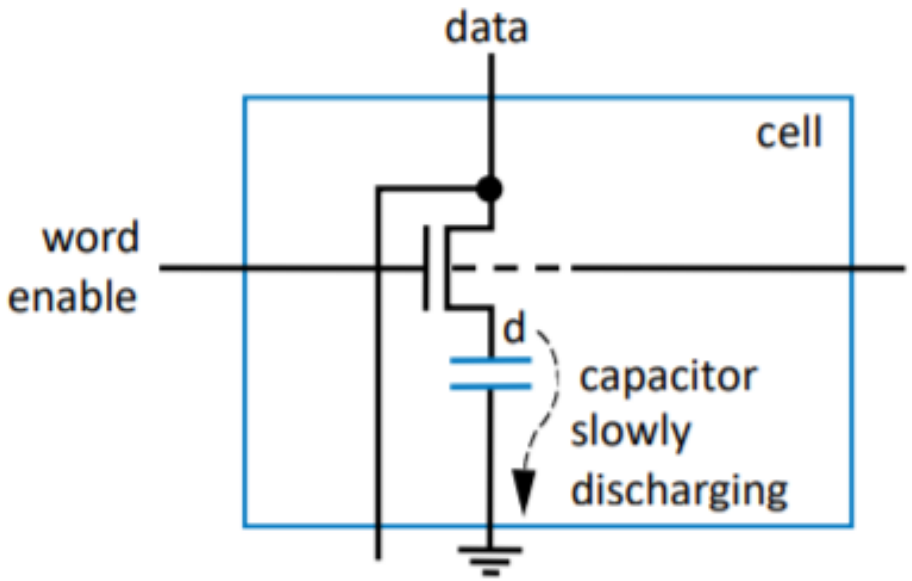
- 6 transitors (Why?)
- You need to understand
  - How to write a cell?
  - How to read a cell?





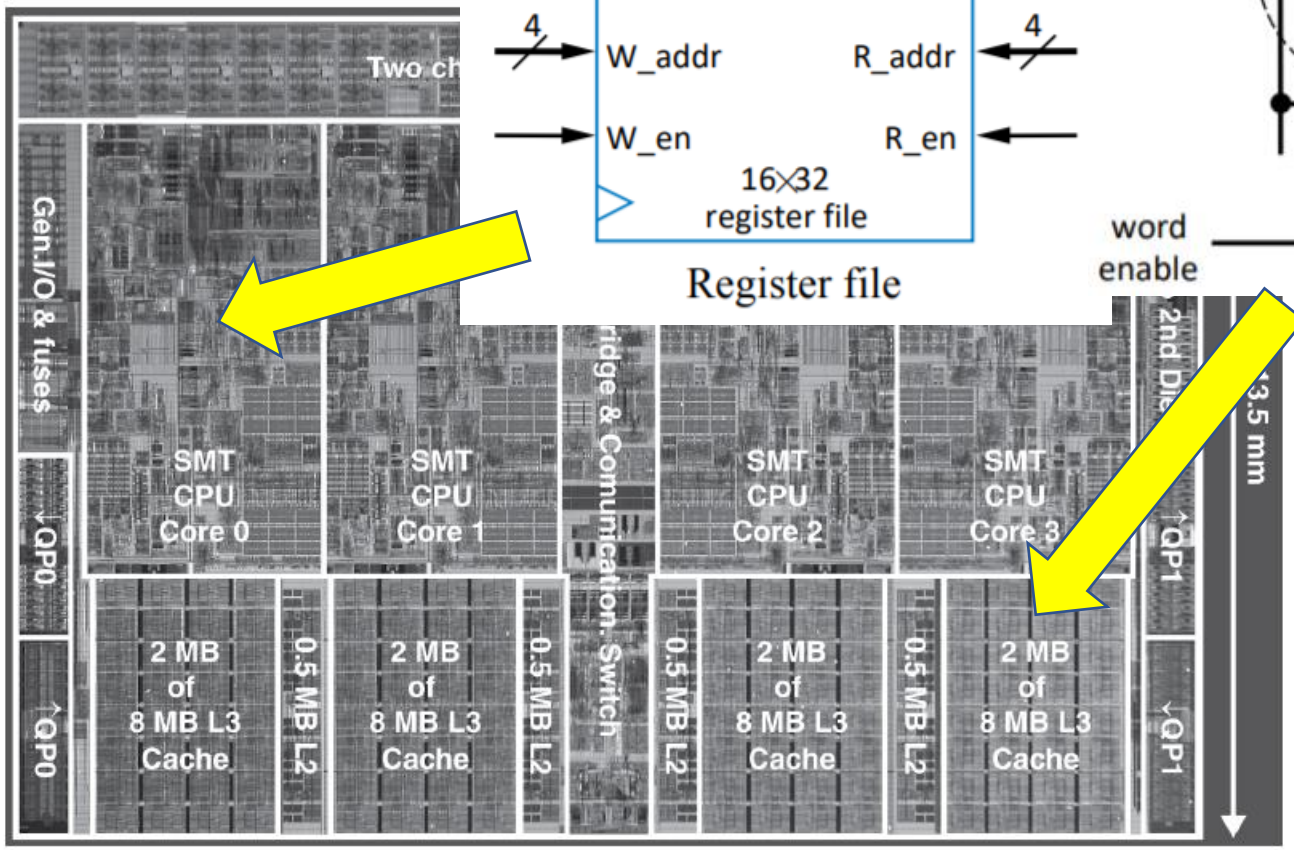
3. Memory
Concept: Dynamic RAM (DRAM)

- 1 transistor and 1 capacitor
- Why use DRAM? Cheaper, larger.

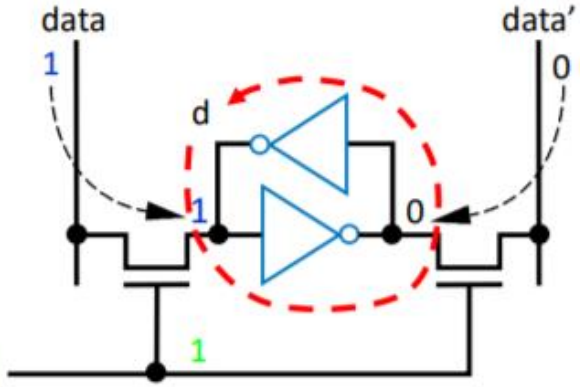


Example: Intel Nehalem 4-core processor

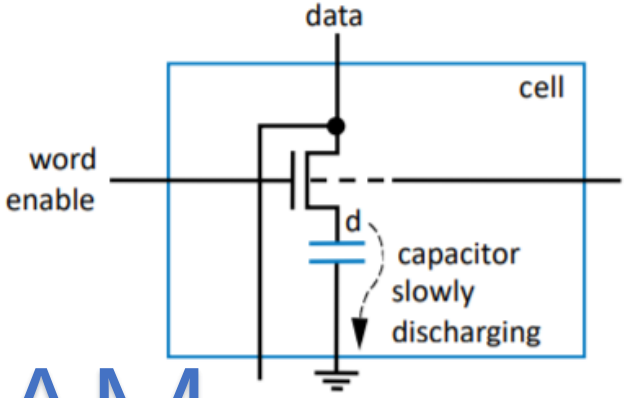
Intel Nehalem 4-core p



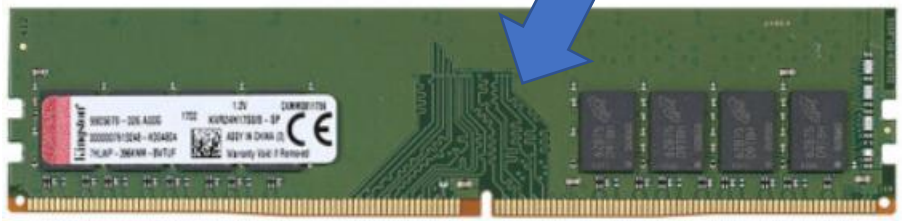
Register file



SRAM



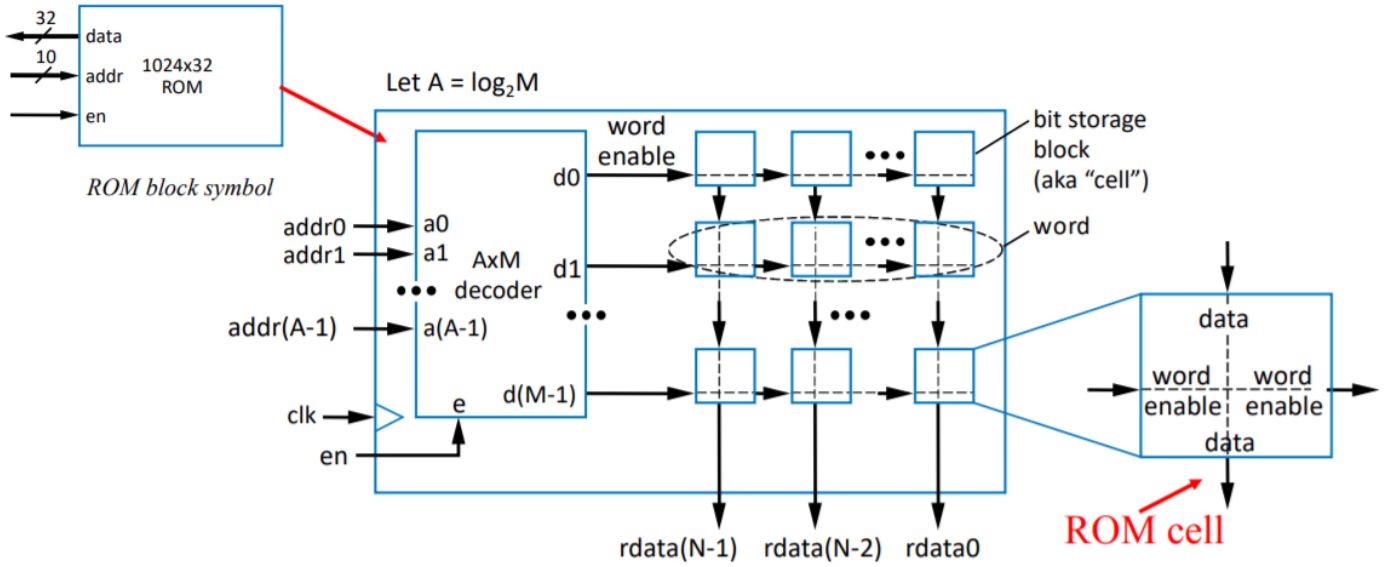
DRAM



Per core: 32KB L1 I-cache, 32KB L1 D-cache, 512KB L2 cache

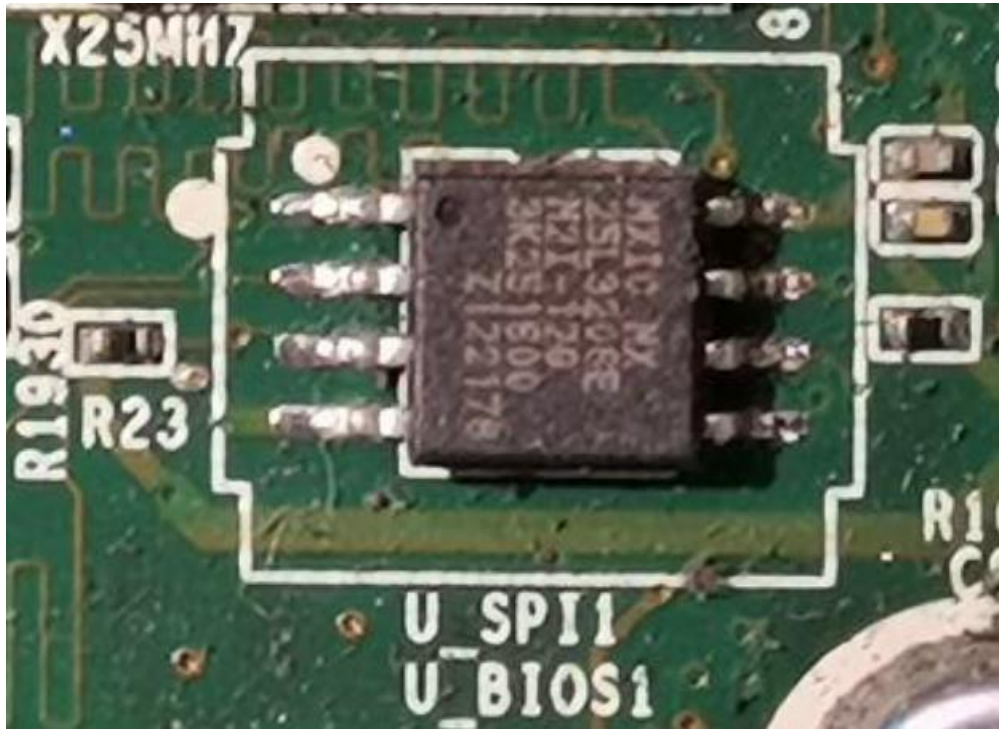
3. Memory
Concept: Read-Only Memory (ROM)

- Memory that can only be read, but cannot be written
- So… Why use ROM?
  - Small
  - Non-volatile
  - Fast
  - Low power
- Structure: Similar to RAM
- Type: EPROM, EEPROM…



3. Memory
Concept: Read-Only Memory (ROM)

- Where can you find ROM in your computer?
- Eg. Basic Input & Output System (BIOS) program is stored in ROM on your motherboad.

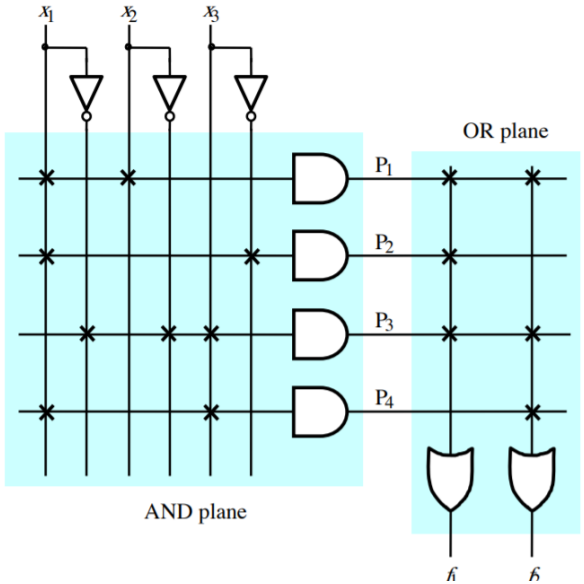


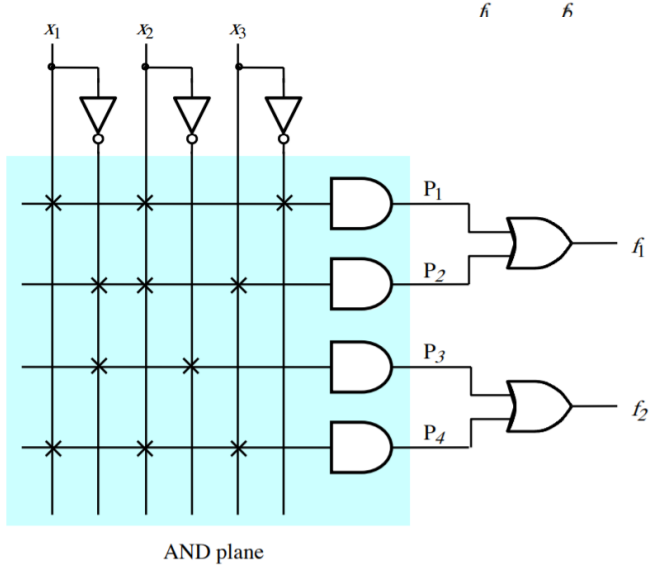
(The ROM in the picture is actually in one of the computers in JI…)

4. Programmable Logic Devices (PLD) Concept

- Simple Simple programmable logic devices (SPLD)
  - Programmable logic array (PLA)
  - Programmable array logic (PAL)
- Complex programmable logic array (CPLD)
- Field-programmable gate array (FPGA)

4. Programmable Logic Devices (PLD) Concept: PLA & PAL



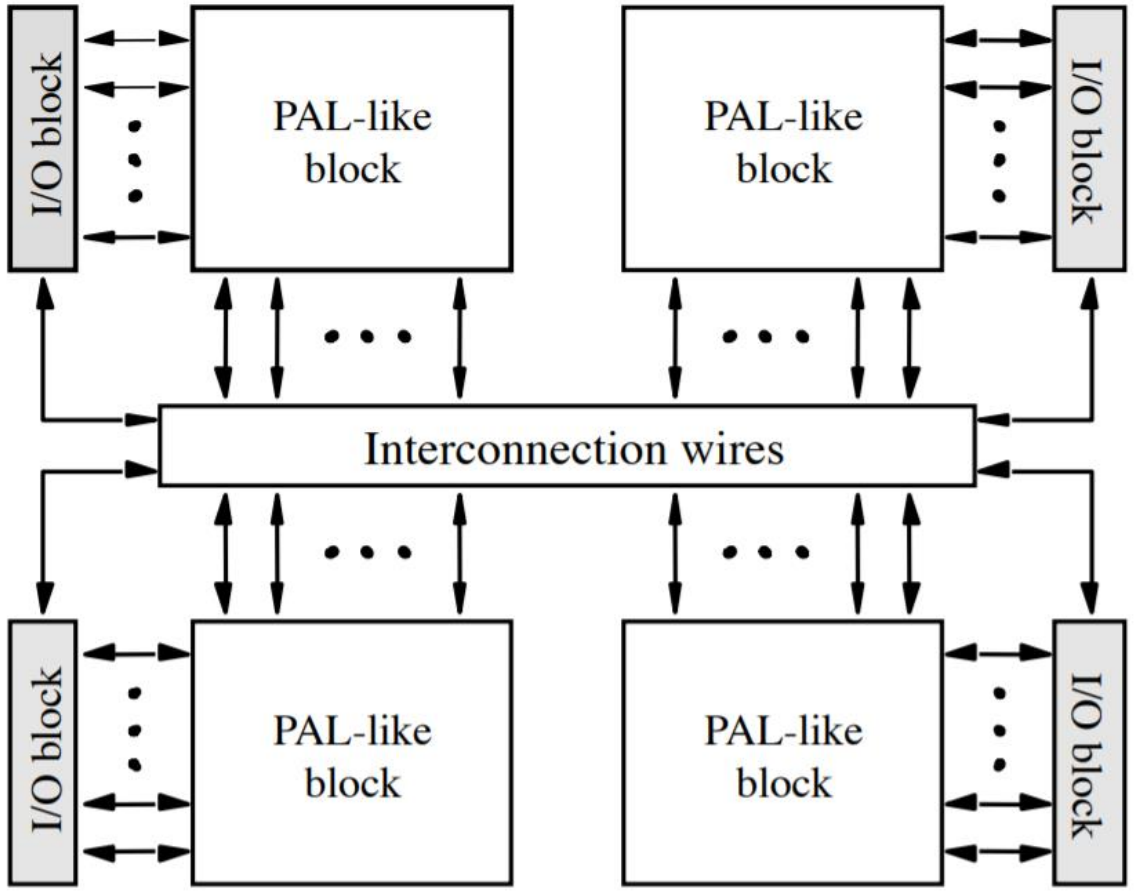


BASIS FOR COMPARISON	PLA	PAL
Stands for	Programmable Logic Array	Programmable Array Logic
Construction	Programmable array of AND and OR gates.	Programmable array of AND gates and fixed array of OR gates.
Availability	Less prolific	More readily available
Flexibility	Provides more programming flexibility.	Offers less flexibility, but more likely used.
Cost	Expensive	Intermediate cost
Number of functions	Large number of functions can be implemented.	Provides the limited number of functions.
Speed	Slow	High

Reference: techdifferences.com/difference-between-pla-and-pal.html

4. Programmable Logic Devices (PLD) Concept: CPLD

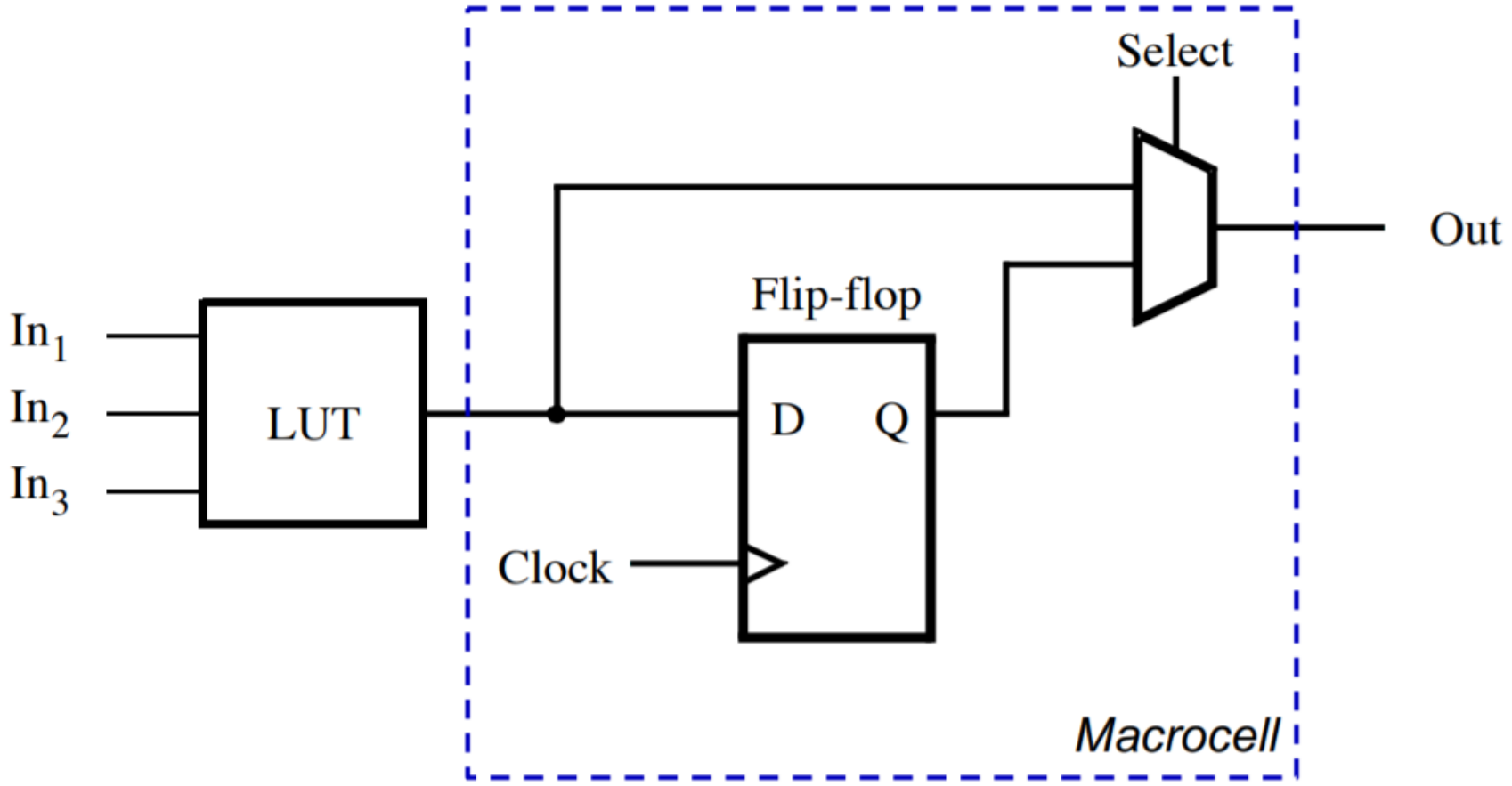
- More complex than PLA or PLA
- Provide more I/O



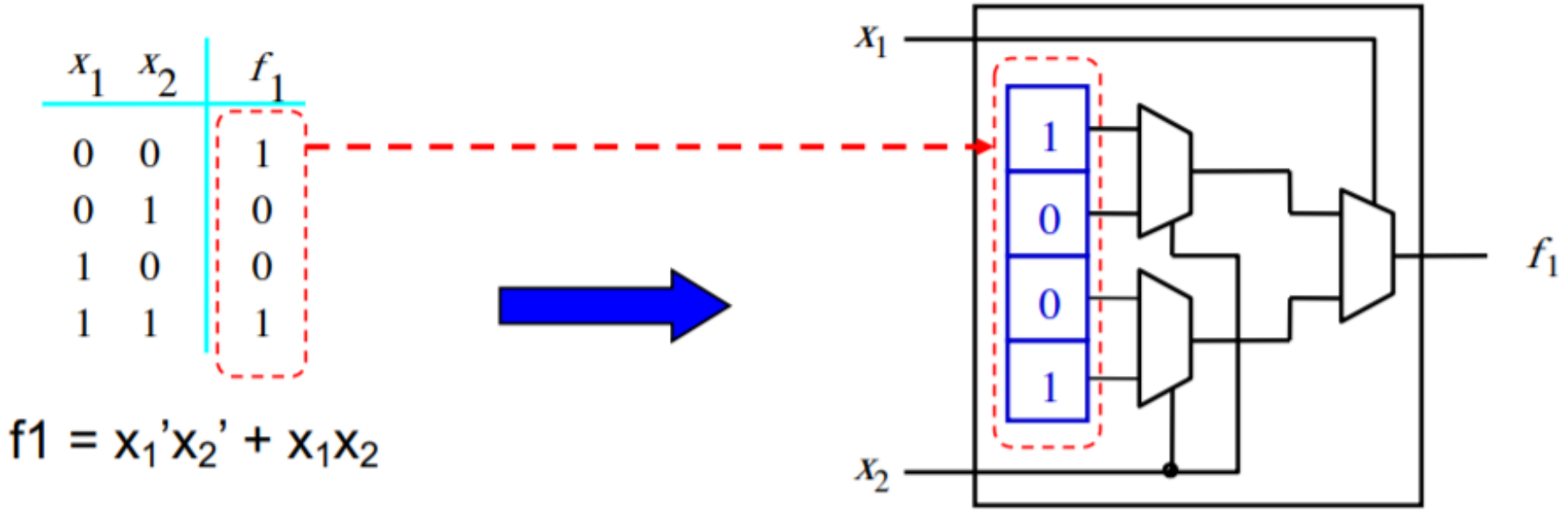
4. Programmable Logic Devices (PLD) Concept: FPGA

- Reprogrammable
  - SRAM-based (volatile, what we use in lab)
  - EEPROM-based (Flash-based)
- One-time programmable (OTP)
  - Anti-fuse-based FPGA
    - Anti-fuse: apply high-voltage/temperature: insulating->conducting
  - EPROM-based

4. Programmable Logic Devices (PLD)
Configurable Logic Block (CLB) in FPGA



4. Programmable Logic Devices (PLD) Look-Up Table (LUT) in FPGA



Thanks for your support!
Good luck and see you VE370 :)