



## การจัดองค์การคอมพิวเตอร์

# W5.3 ซีพียูแอสกค์

31110321 Computer Organization

สำหรับนักศึกษาชั้นปีที่ 3 สาขาวิชาวิศวกรรมคอมพิวเตอร์

ทรงฤทธิ์ กิติศรีวรพันธุ์

songrit@npu.ac.th

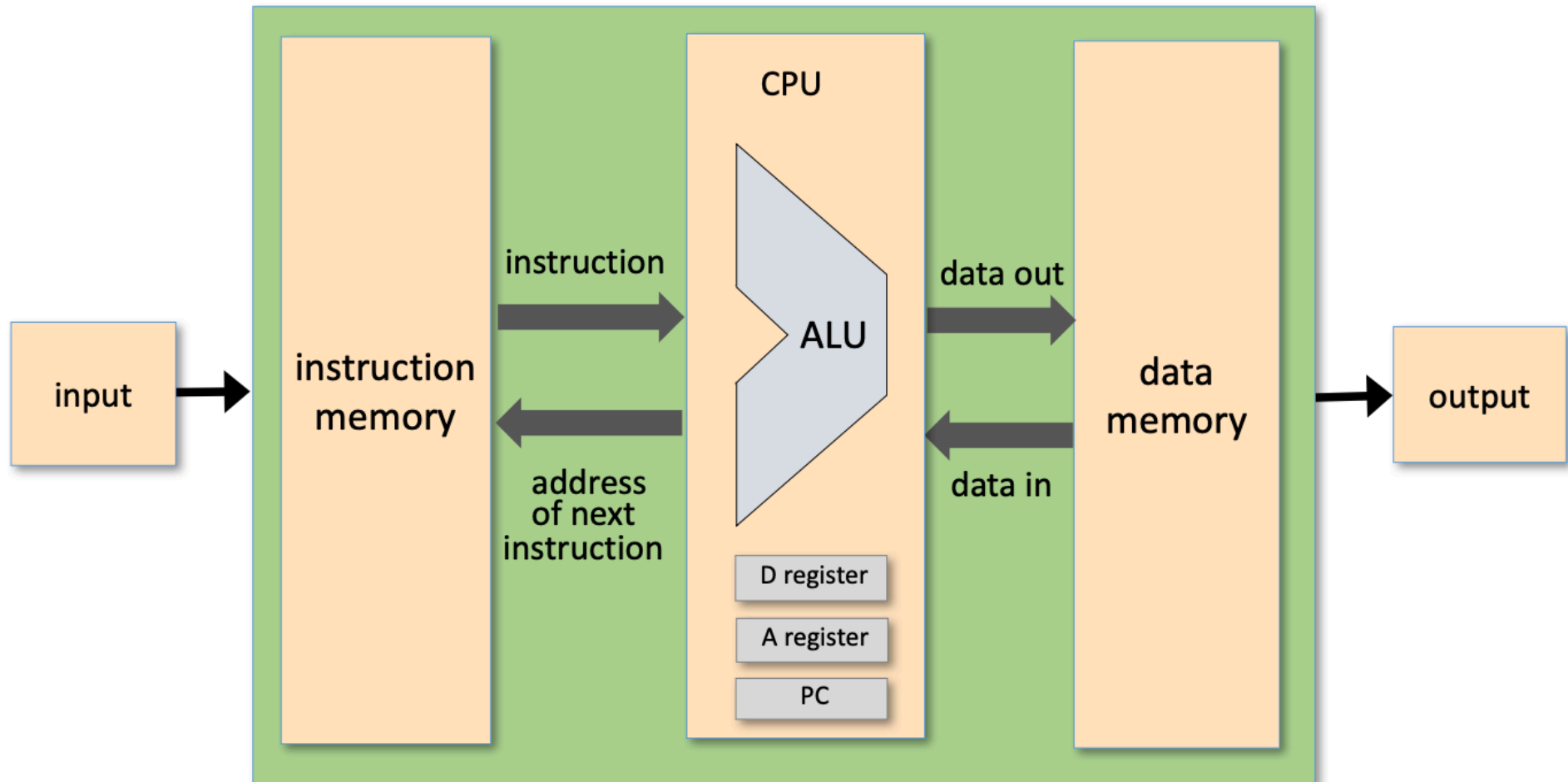
สาขาวิชาวิศวกรรมคอมพิวเตอร์  
มหาวิทยาลัยนครพนม

# Lecture plan

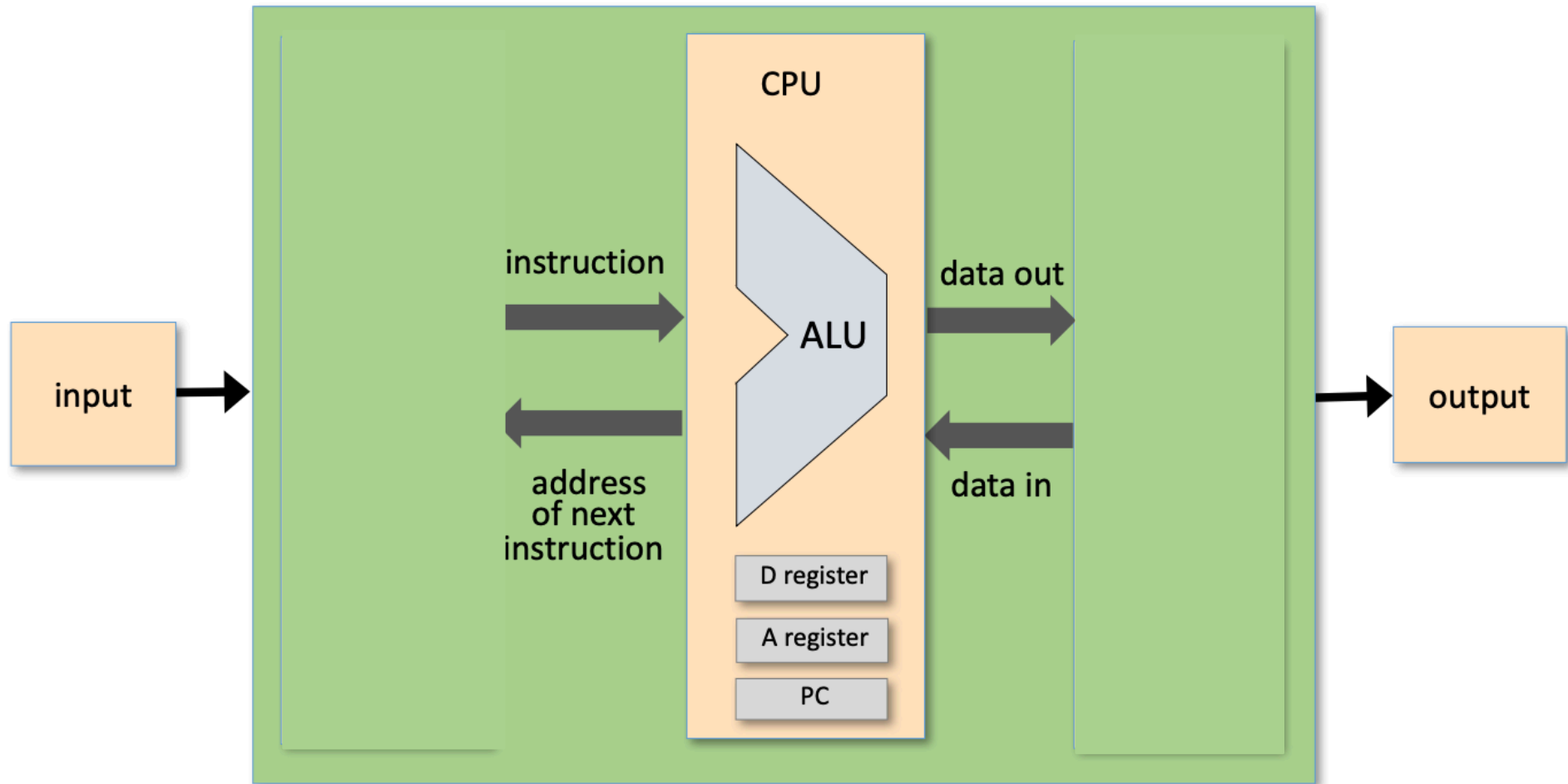
---

- 5.1 สถาปัตยกรรมพอนนอยมันน์
- 5.2 Fetch-Execute Cycle
- **5.3 ซีพียูแอสค**
- 5.4 แอสคคอมพิวเตอร
- 5.5 ภาพรวมโปรเจค 5

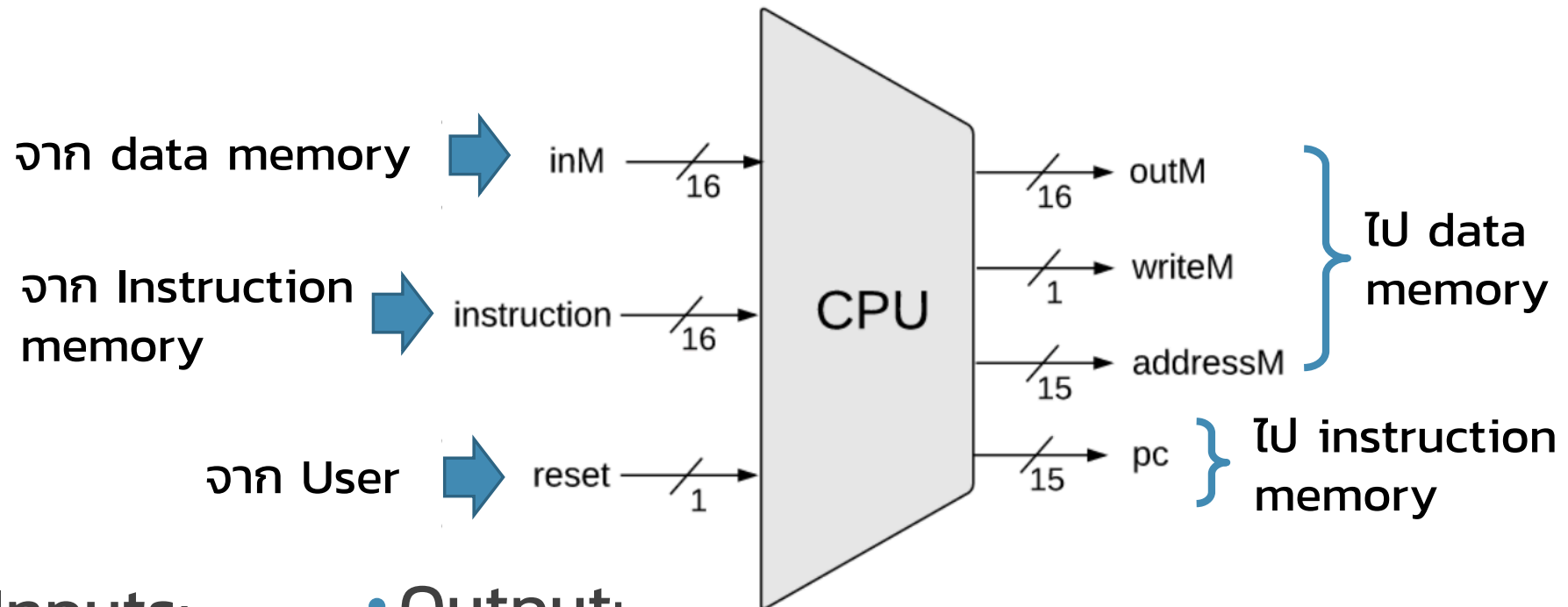
## Computer System



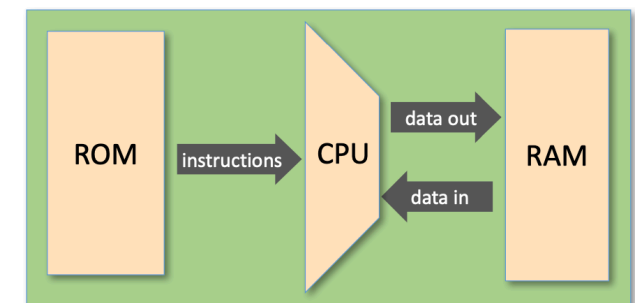
## Computer System



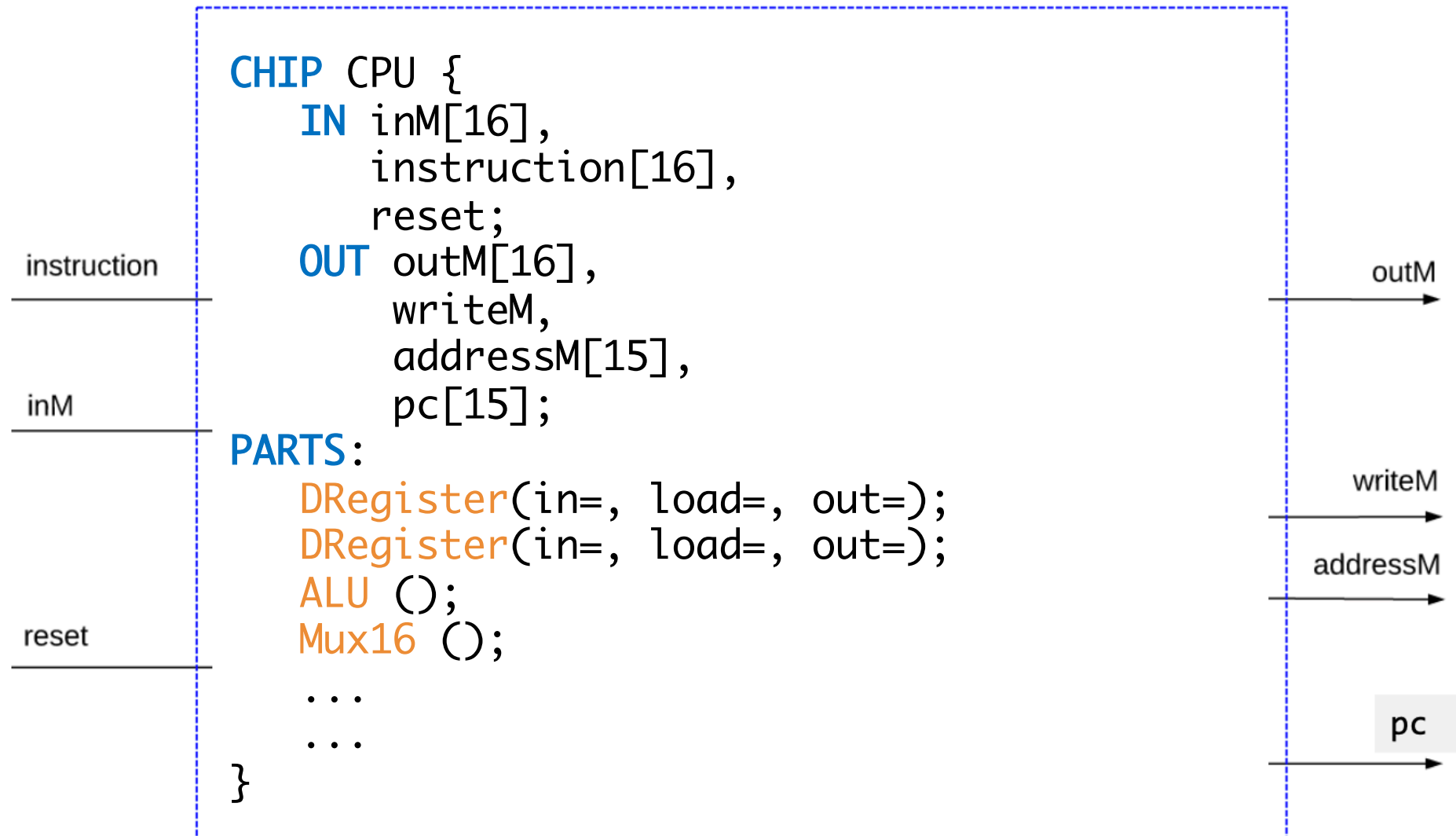
# Hack CPU Interface

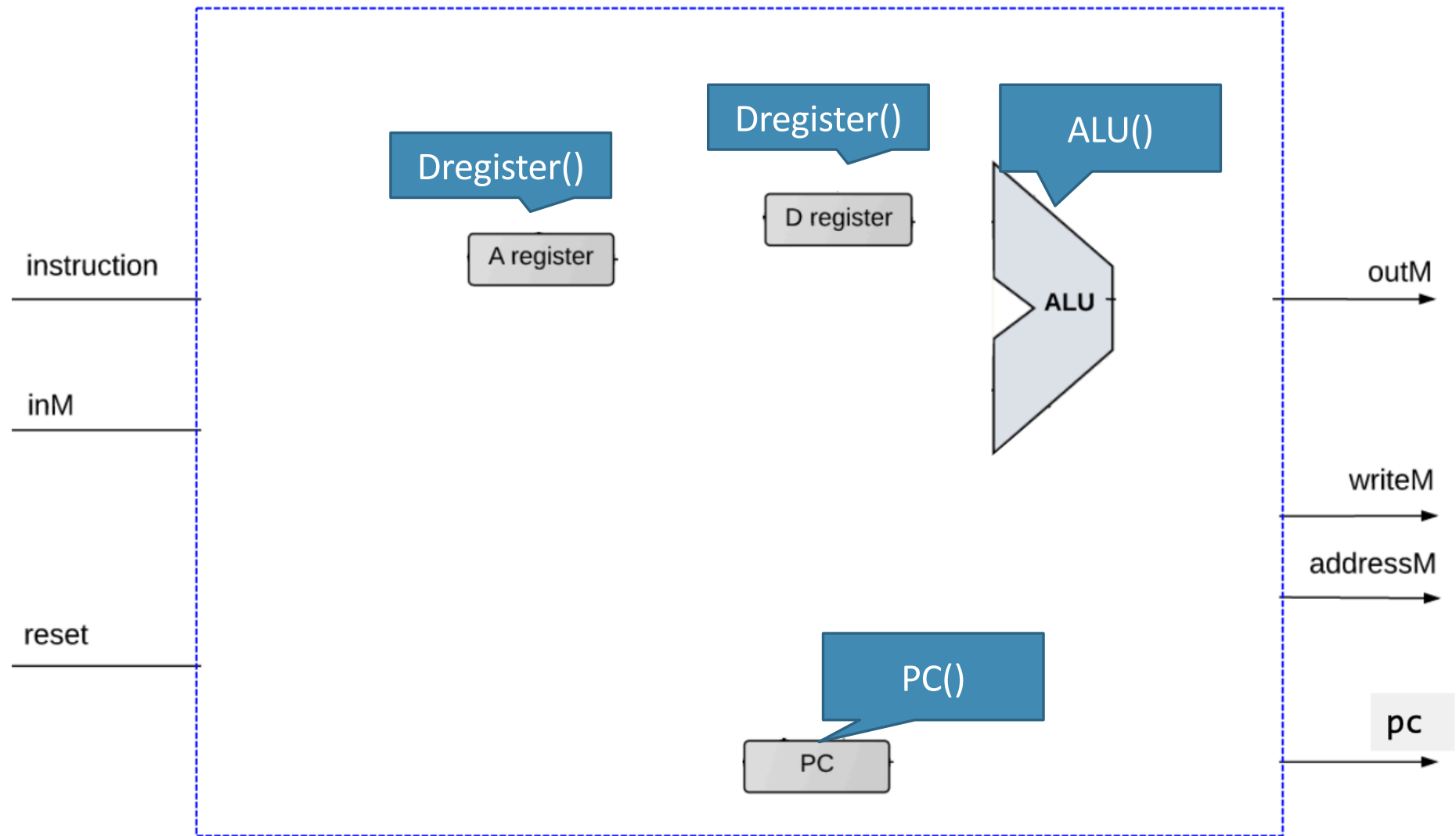


- Inputs:
  - Data value
  - Instruction
  - Reset bit
- Output:
  - Data value
  - Write mem ? (yes/no)
  - Memory addr.
  - Addr. of next.

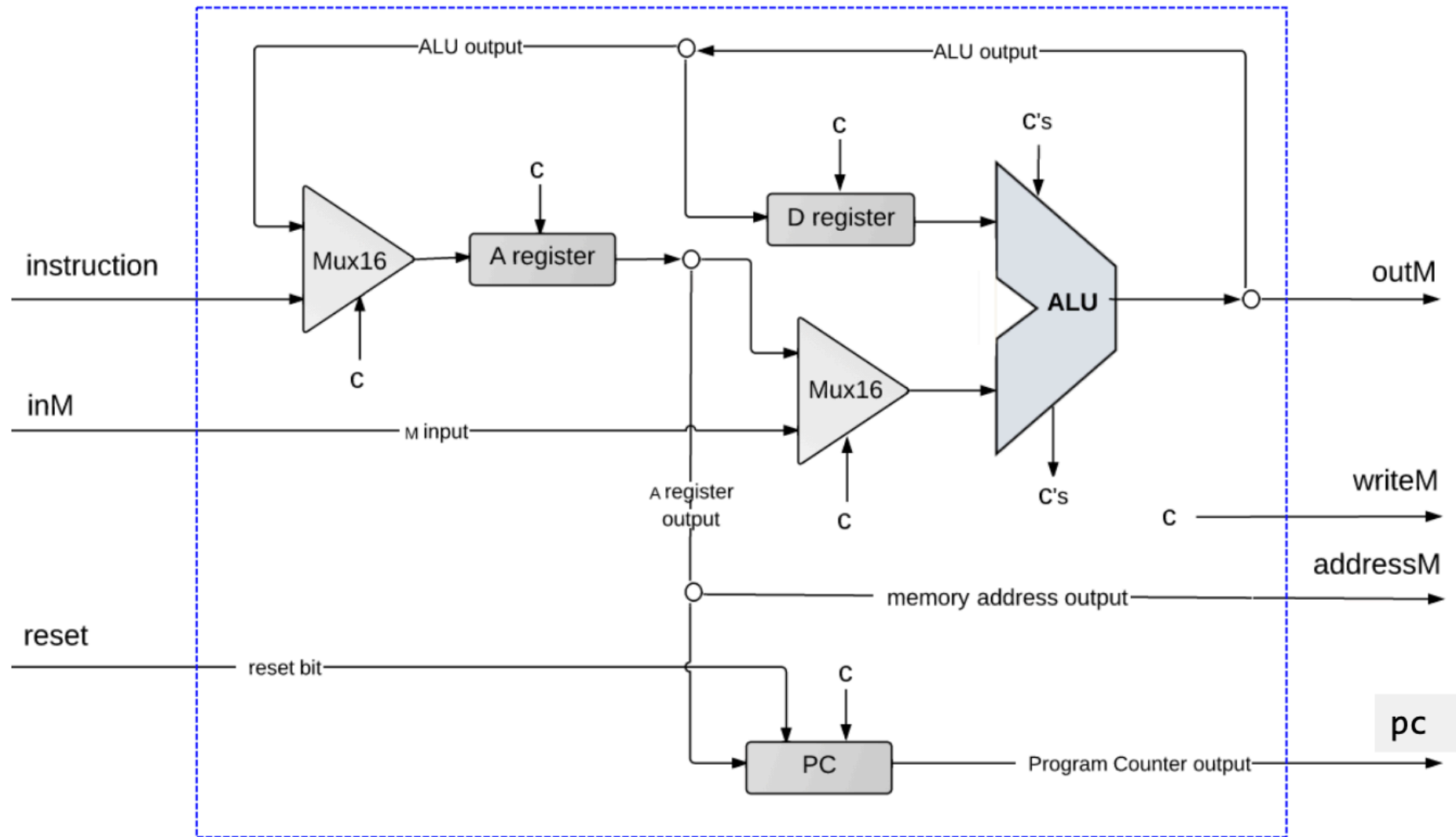


# CPU interface



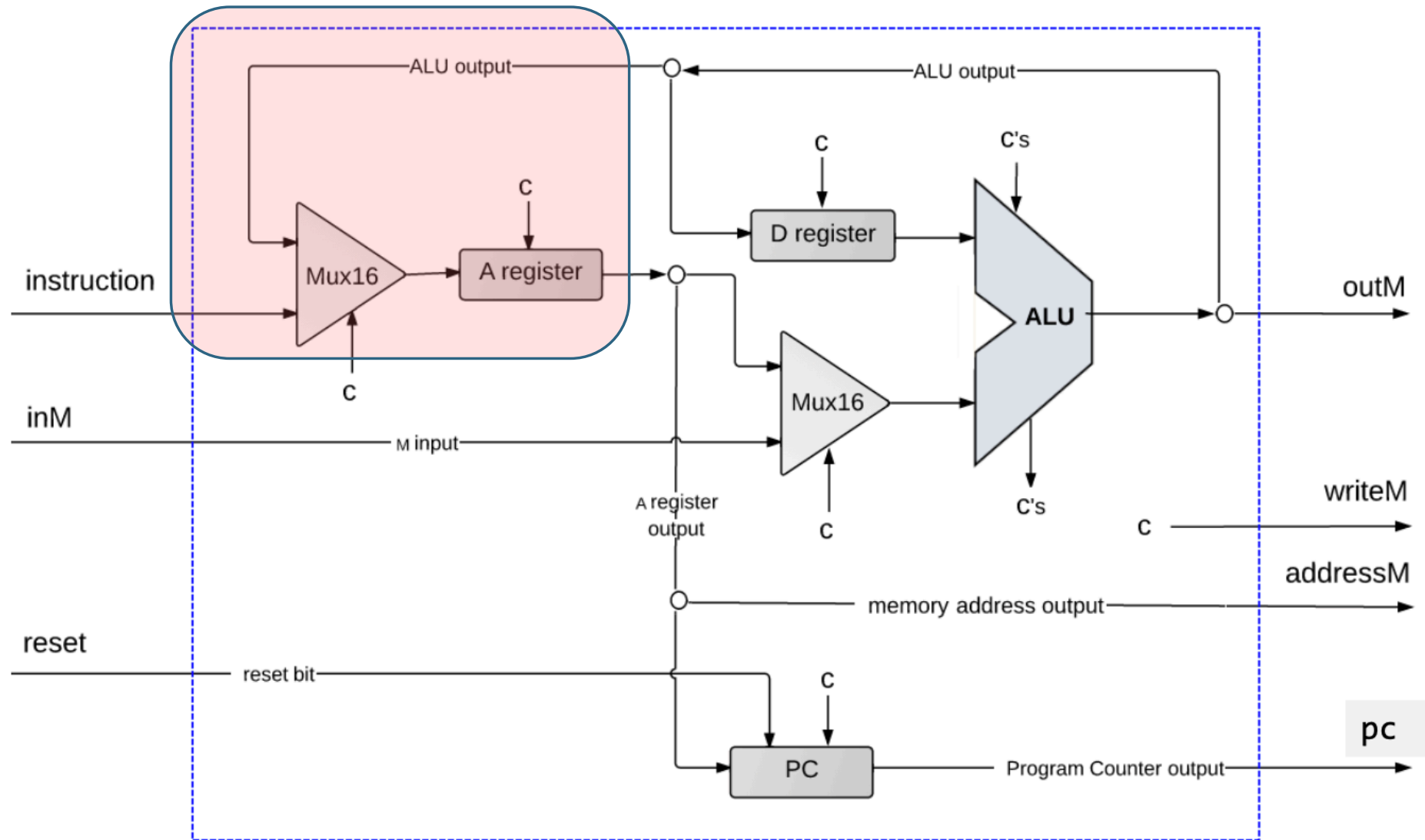


# C คือ control bit



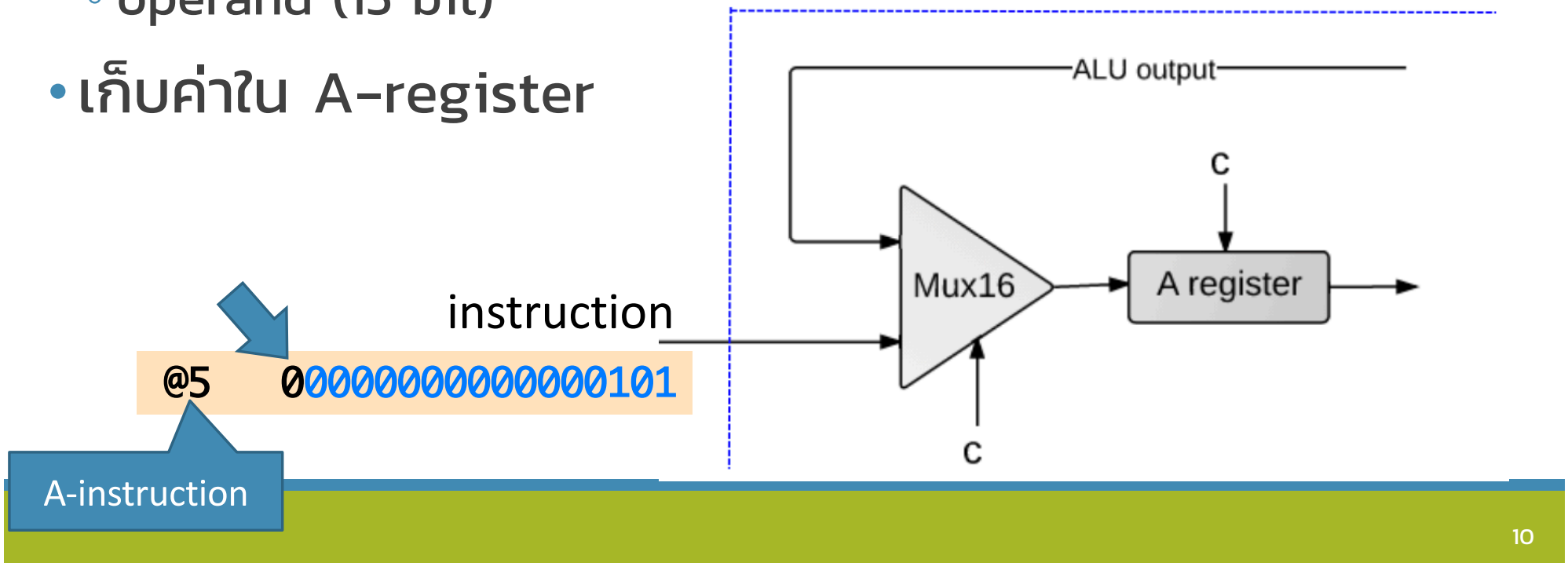


# CPU : instruction handling



# CPU : A-instruction

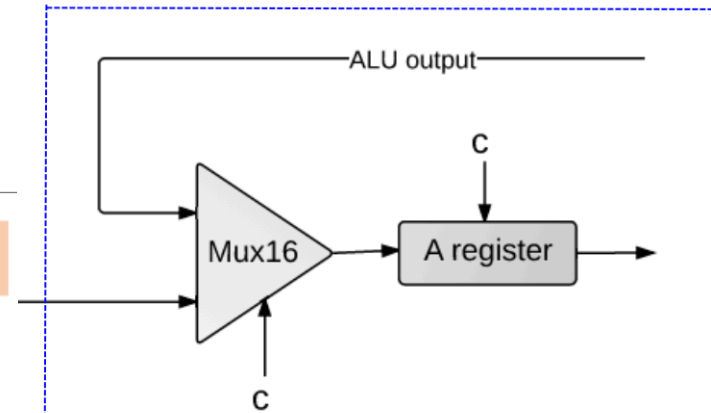
- เริ่มต้นด้วยบิต 0
- หลังจากถอด mnemonic เป็น **instruction code**
- แบ่ง instruction เป็น 2
  - ส่วน opcode (1บิต)
  - operand (15 bit)
- เก็บค่าใน A-register



# Sample HDL

@5 0000000000000101

CHIP CPU {



PARTS:

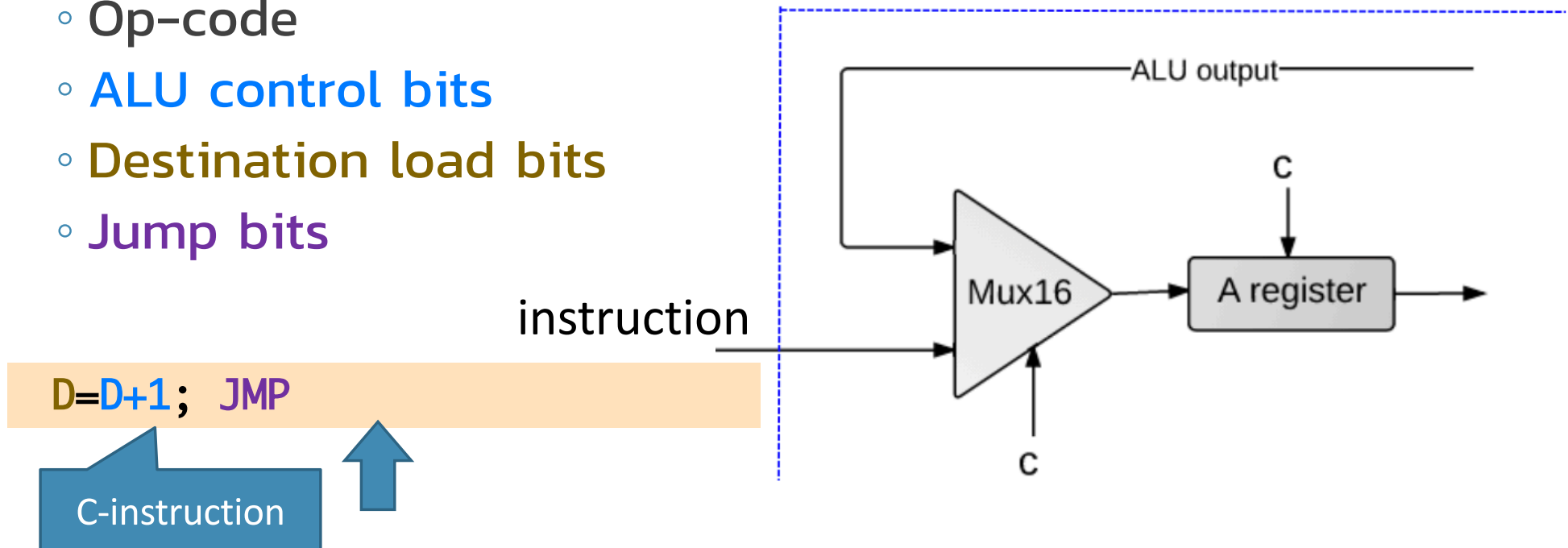
```
Not(in=instruction[15],out=ni);
Mux16(a=outtM,b=instruction,sel=ni,out=i);
Or(a=ni,b=instruction[5],out=intoA);
ARegister(in=i,load=intoA,out=A,out[0..14]=addressM);
...
...
```

}

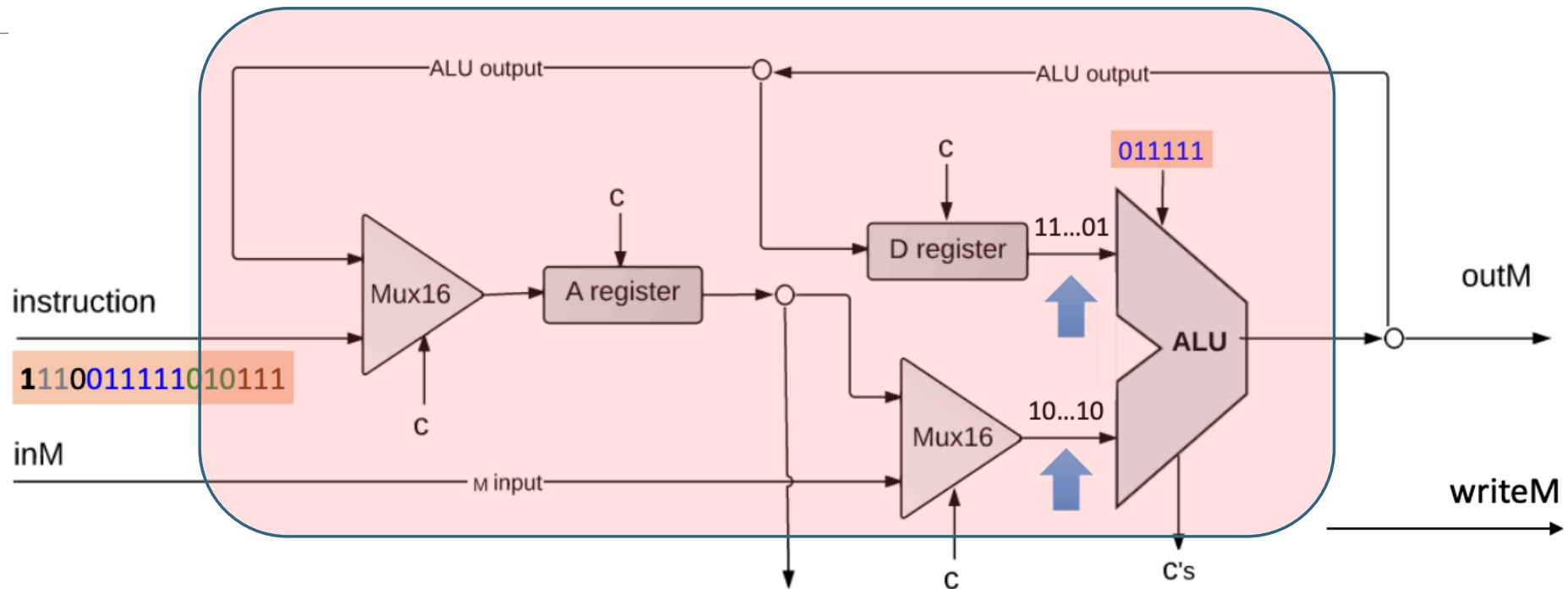
- Handle A-instruction

# CPU : C-instruction

- เริ่มต้นด้วย bit 1
- แปลง mnemonic เป็น instruction code
- แบ่ง instruction เป็น 4 ส่วน
  - Op-code
  - ALU control bits
  - Destination load bits
  - Jump bits

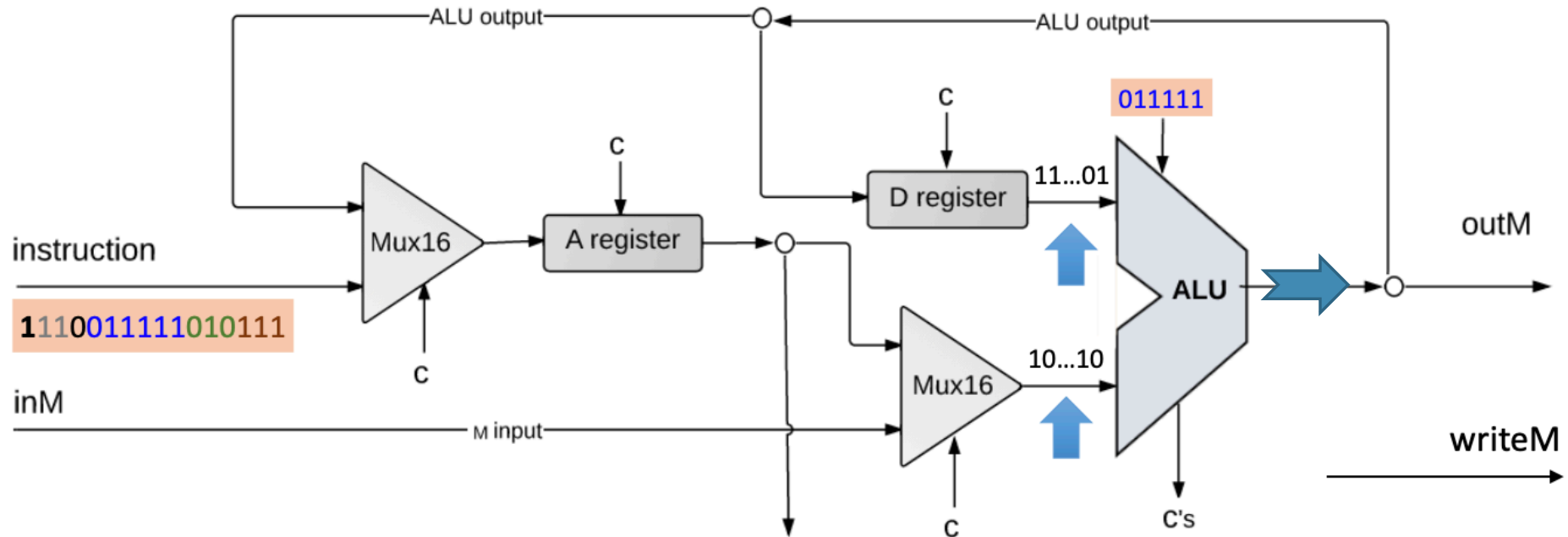


# CPU : C-instruction



- ALU data input:
  - Input 1 : הַרְח D-register
  - Input 2: הַרְח
    - A-register
    - data memory
- ALU control bit:
  - Control bit
  - הַרְח instruction

# CPU : C-instruction

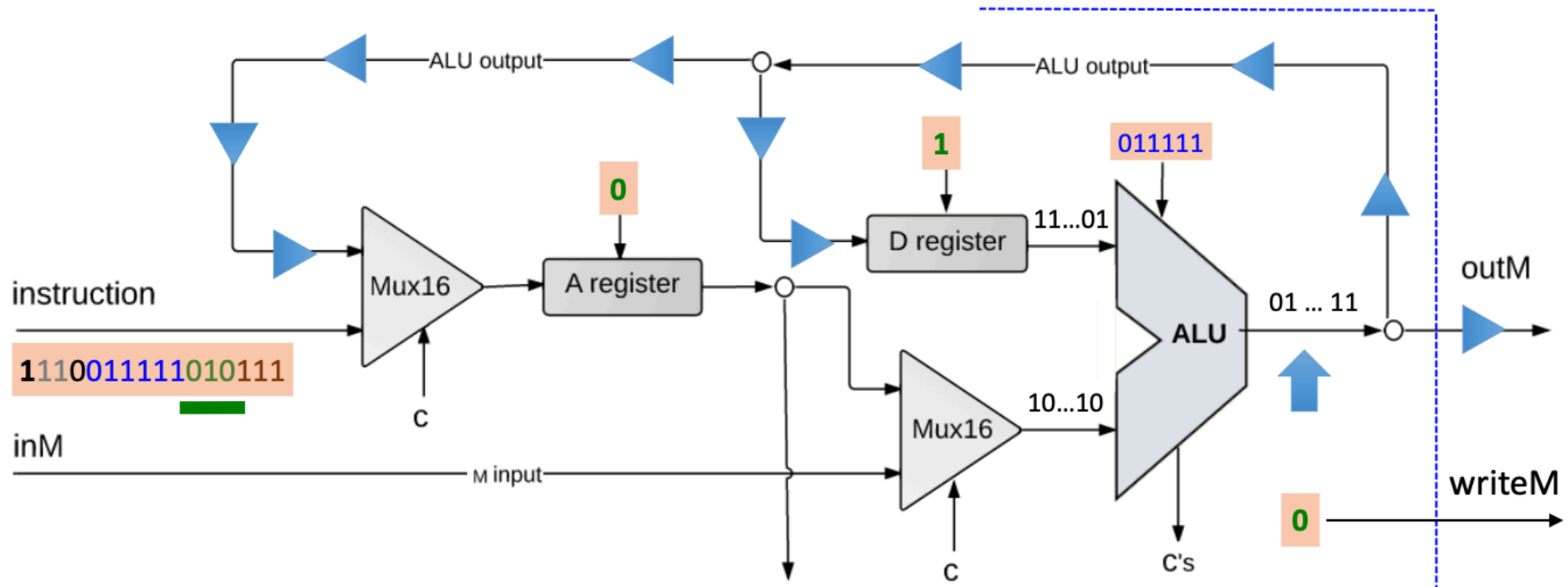


- Output 3 ที่ พร้อมกัน

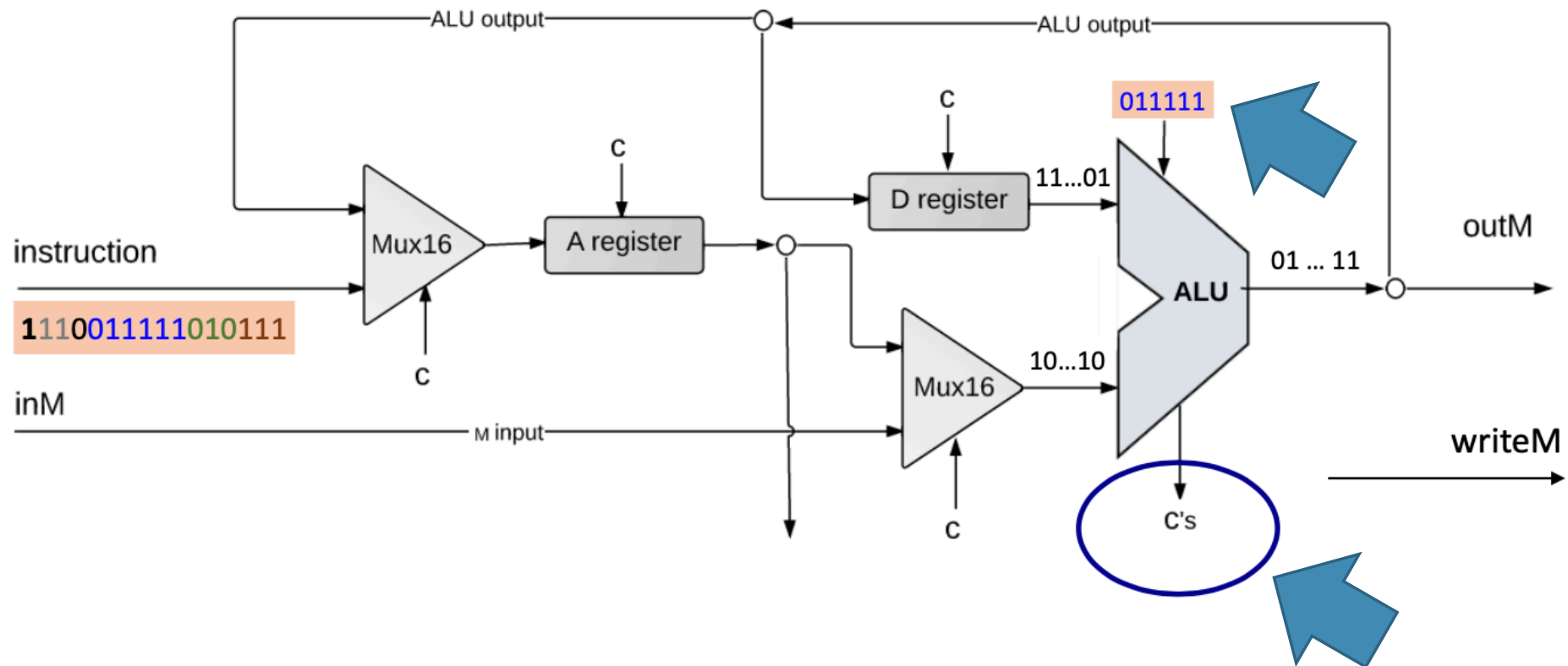
- D register
- A register
- M register

ส่ง output ออกพร้อมกัน  
Fan-out

# CPU : C-instruction

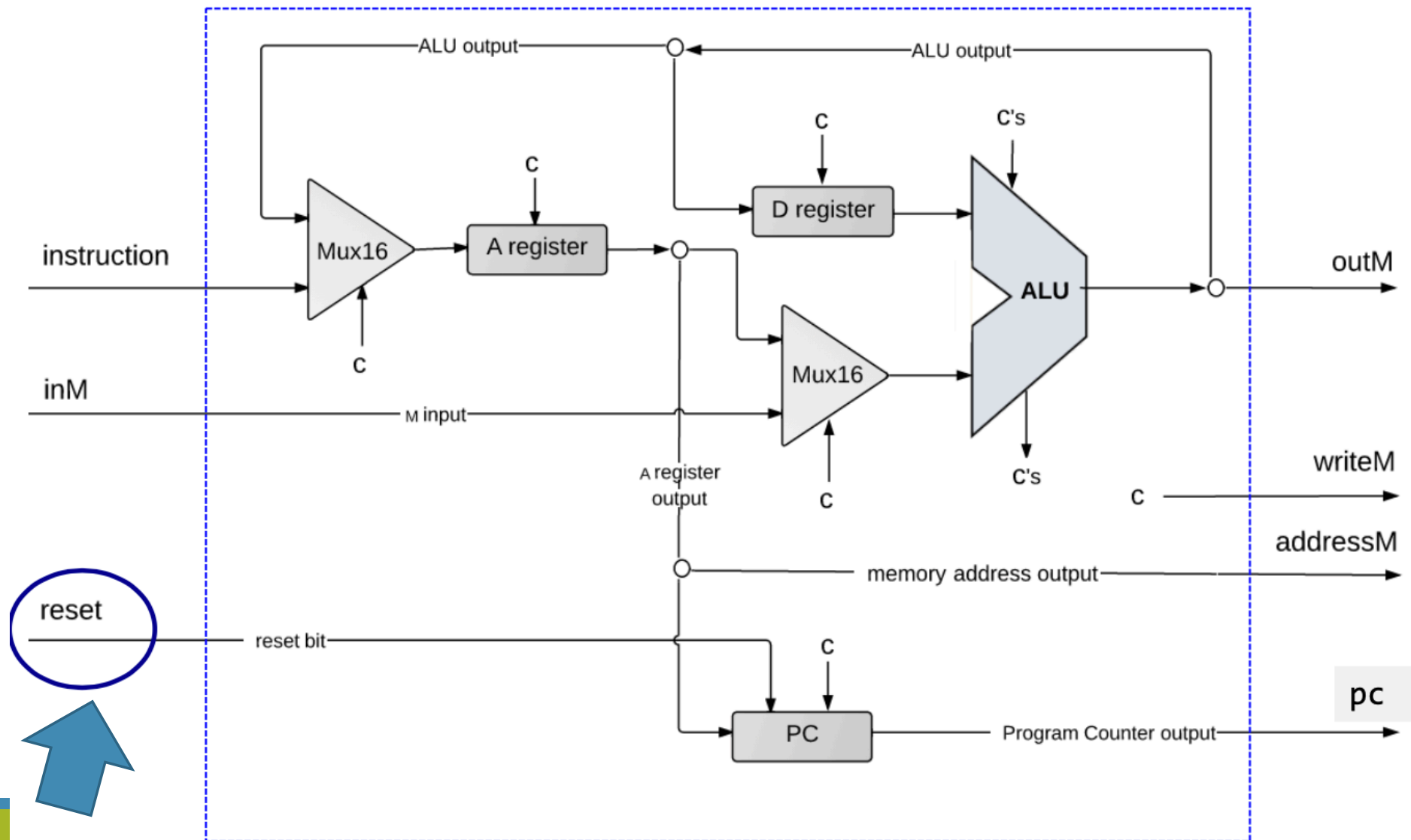


# คำสั่ง ALU : Output





# CPU : control



# CPU : control

---

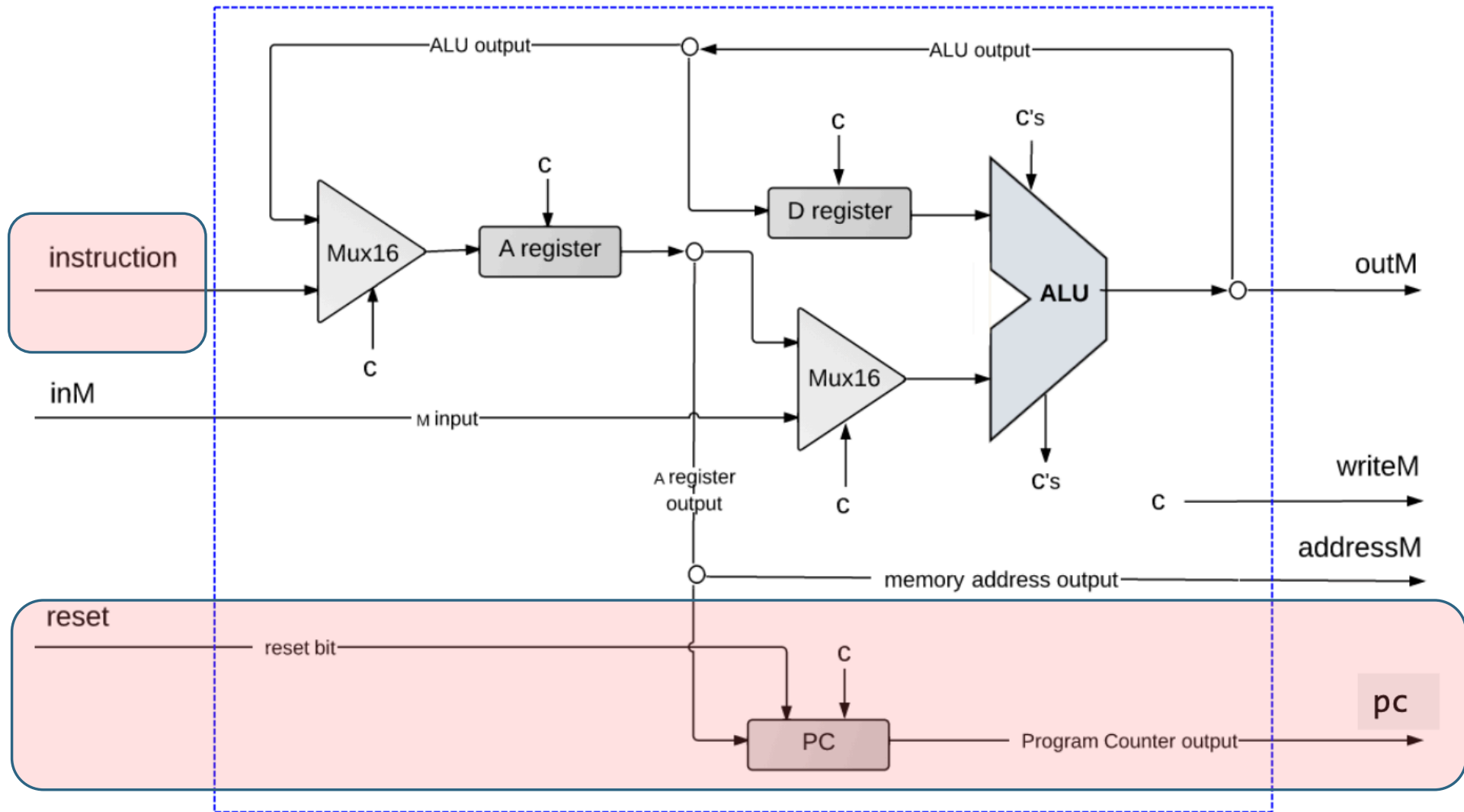
- คอมพิวเตอร์กำลังรันโปรแกรม
- เมื่อกด reset ทำให้โปรแกรมเริ่มต้นรันใหม่



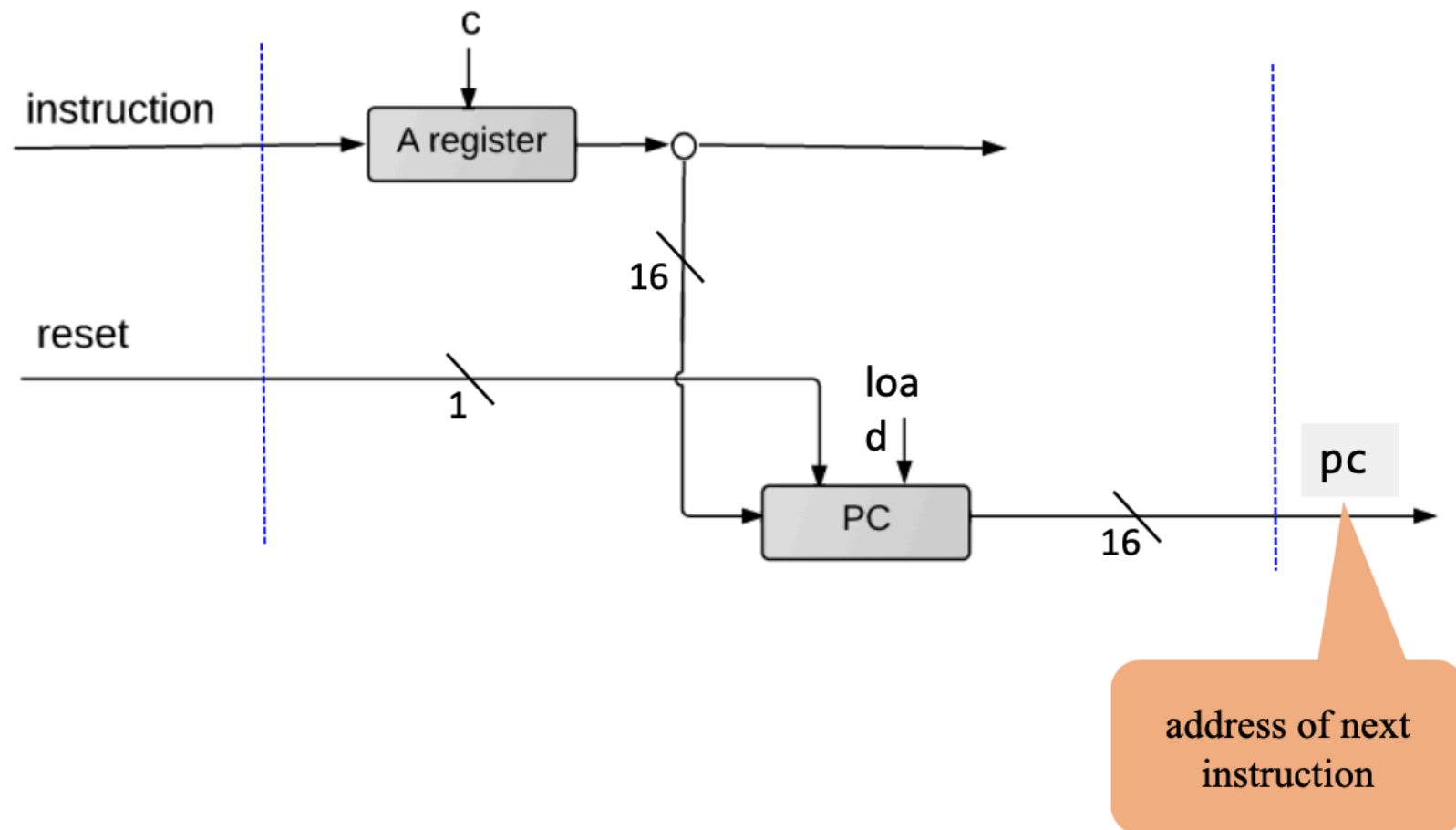
computer exterior  
(well, kind of)

reset

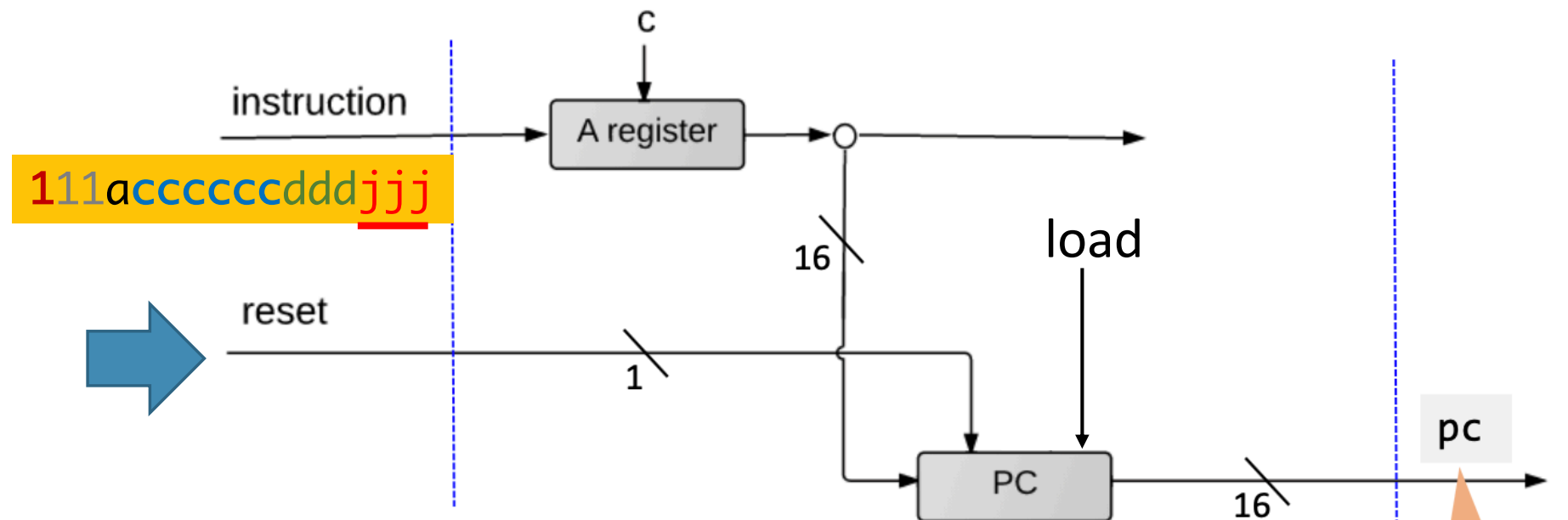
# CPU : instruction handling



# CPU : control



# CPU : control



## PC operation (implementation)

if (reset==1) PC = 0

else

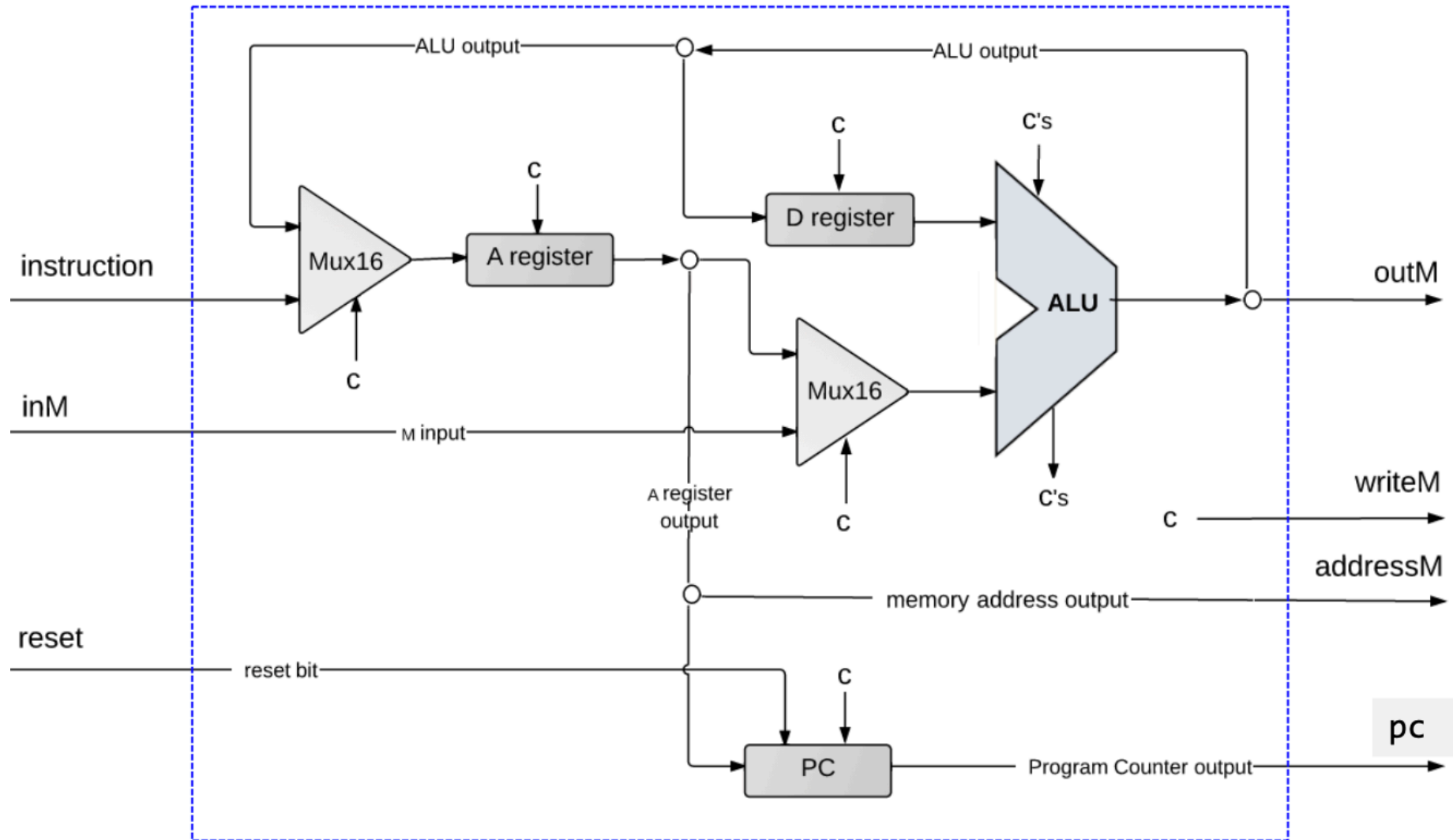
// in the course of handling the current instruction:

$load = f(\text{jump bits, ALU control outputs})$

if (load == 1) PC = A // jump

else PC++ // next instruction

# CPU : complete !!



# Coming up: W5.4 Hack Computer

---

แฮกคอมพิวเตอร์

