Lab 1 Content

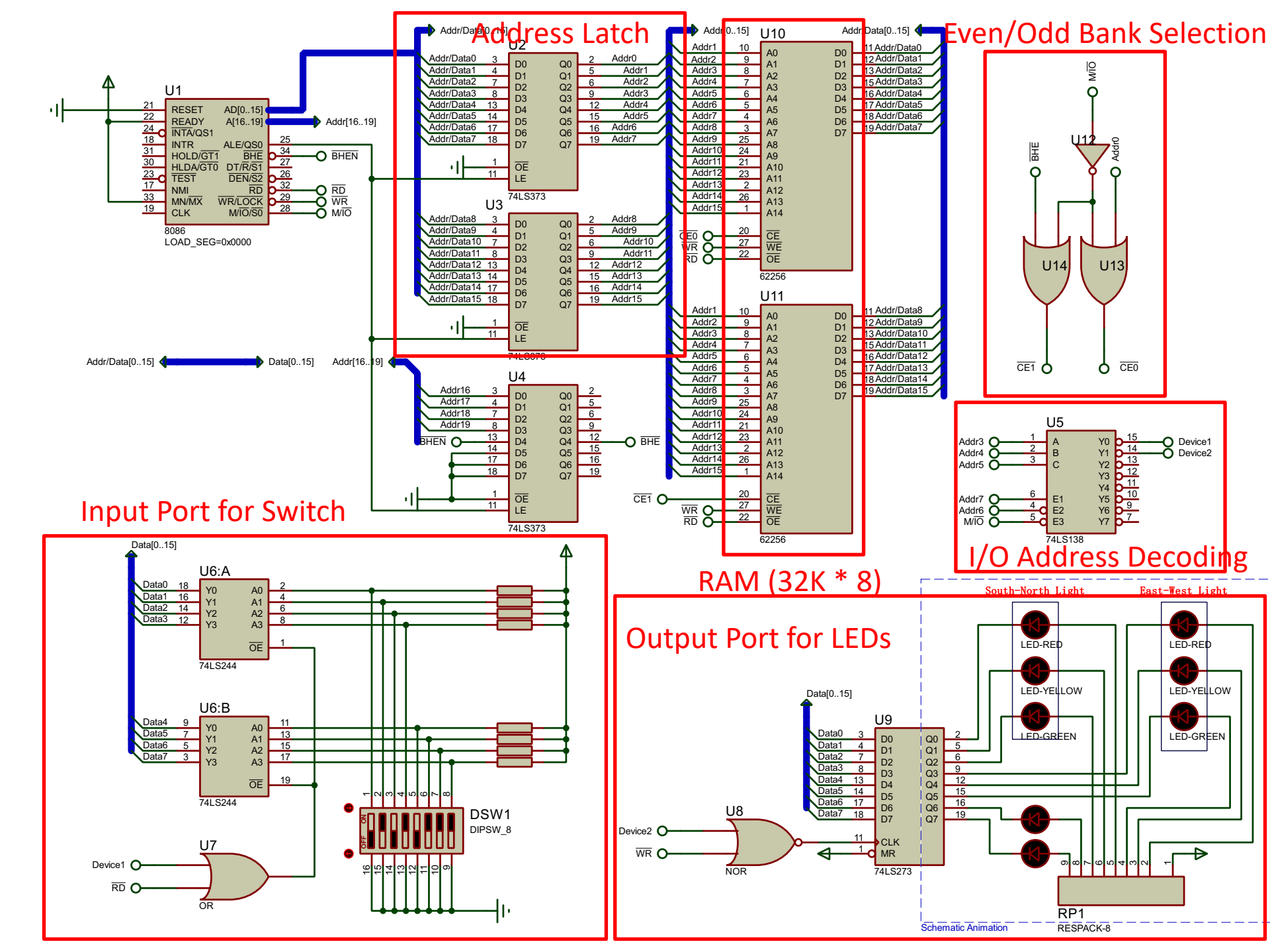
- Install and run Proteus simulation software
  - Instructions:
    https://www.jianshu.com/p/21ad26e0d579
  - Download from jbox:
    https://jbox.sjtu.edu.cn/l/TnaRjS
- Master the basic I/O operations in 8086
- Address decoding: how to derive the I/O port number given the address decoding circuitry

Example Code

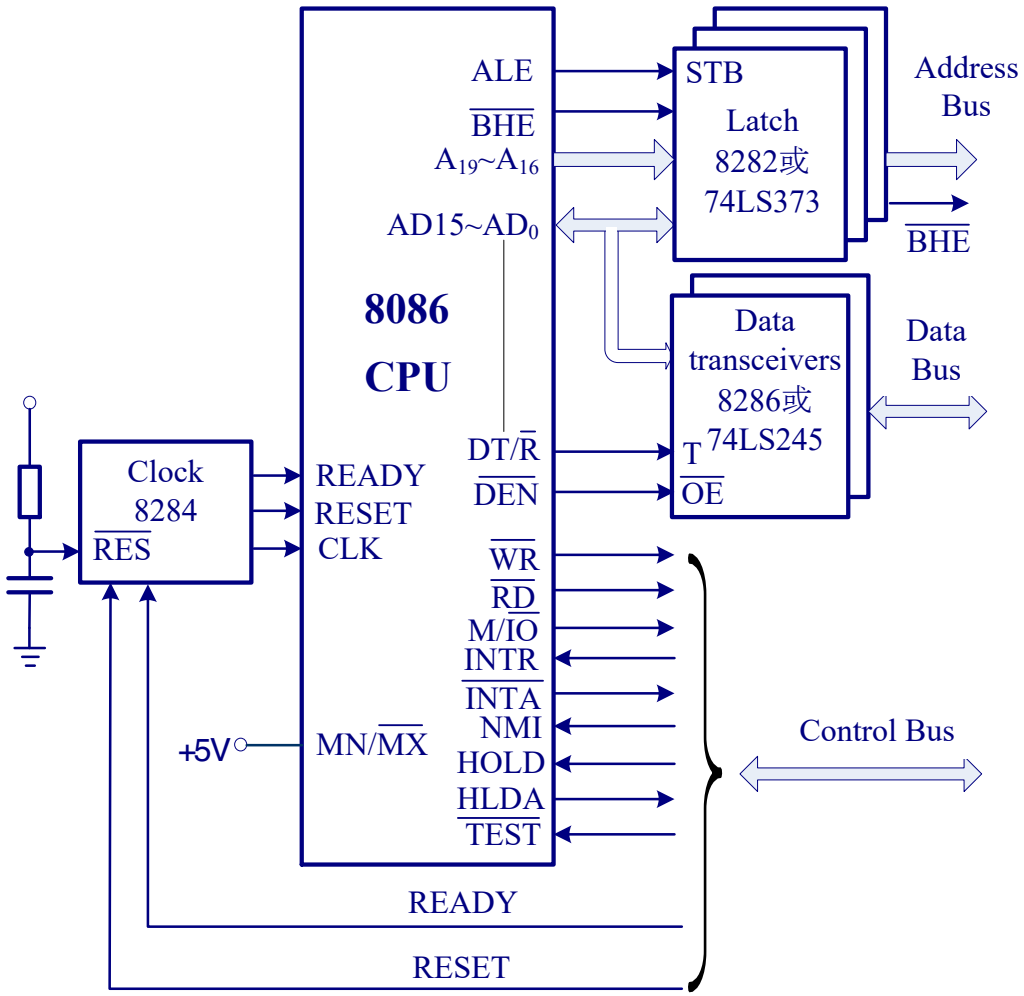
- The example code reads from the Device1 port
- Invert the value
- Write the new value to Device2

- Try running the code and change the switch status (Device1)

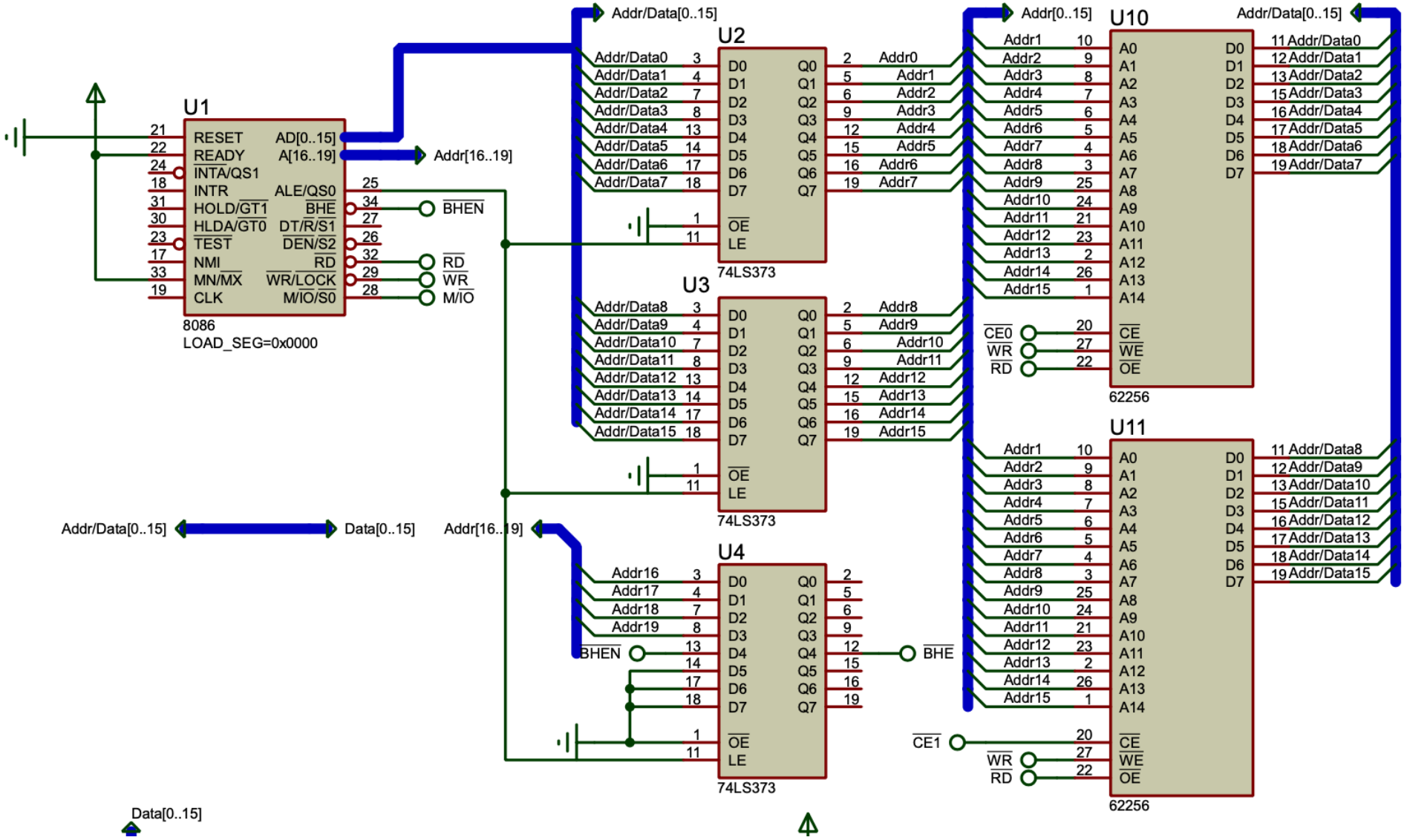
```
.MODEL SMALL
  .DATA
  .STACK 64
  .CODE
Device1   EQU  80h
Device2   EQU  88h
main proc far
Again: IN  AL,Device1
          NOT AL
          OUT Device2,AL
          JMP Again
main endp
END     main
```


Address Latch
Even/Odd Bank Selection
Input Port for Switch
RAM (32K * 8)
I/O Address Decoding
Output Port for LEDs
U1
8086
LOAD_SEG=0x0000
U2
U3
U4
74LS373
U10
U11
62256
U12
U13
U14
U5
74LS138
U6:A
U6:B
74LS244
U7
OR
DSW1
DIPSW_8
U8
NOR
U9
74LS273
South-North Light
East-West Light
LED-RED
LED-YELLOW
LED-GREEN
RP1
RESPACK-8
Schematic Animation

Address Data Bus Decoupling

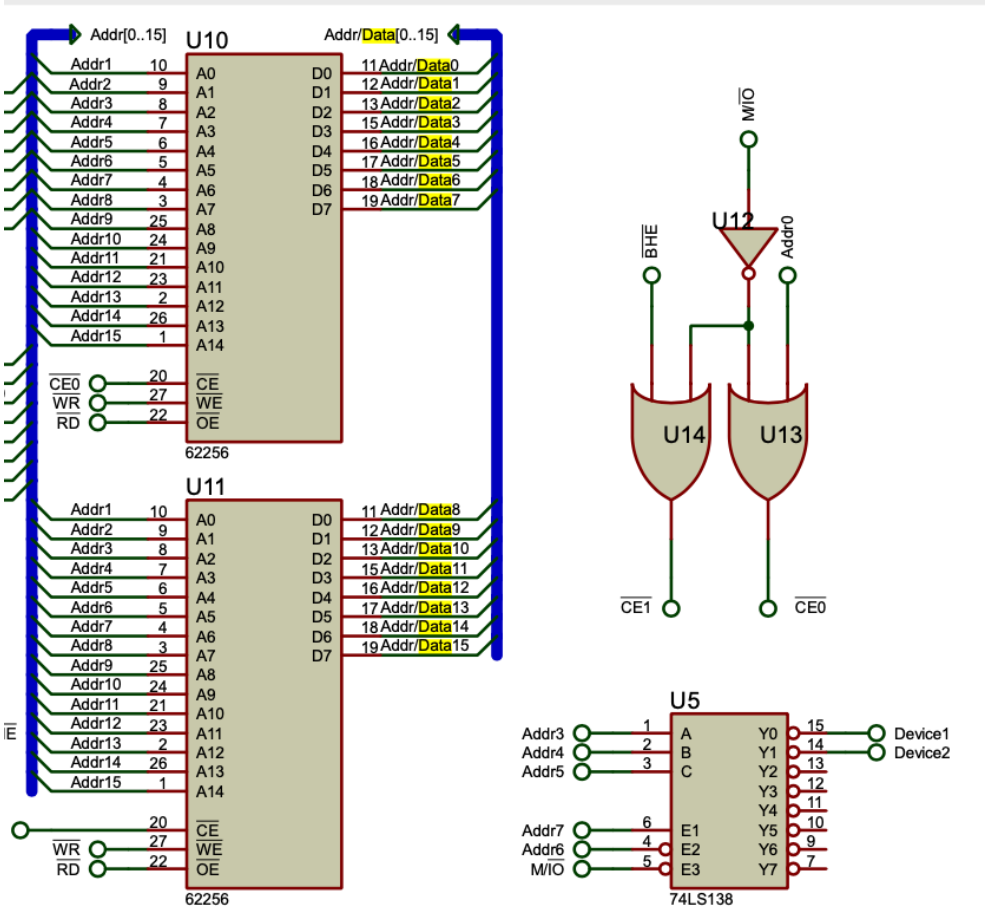


Address Data Bus Decoupling



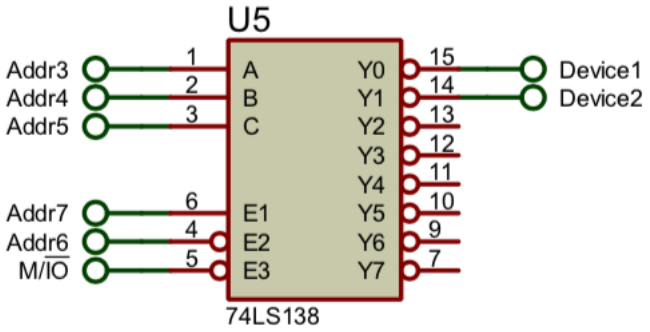
Even/Odd Bank Selection

- When is CE1 and CE0 effective?
  - CE1 == 0 needs BHE == 0 and M/IO == 1
  - CE0 == 0 needs Addr0 == 0 and M/IO == 1



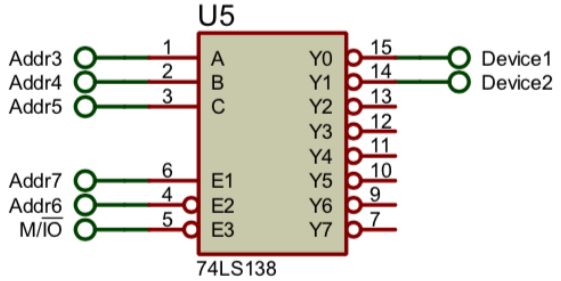
I/O Address Decoding

- What addresses would generate the effective Device1 and Device2 signals?



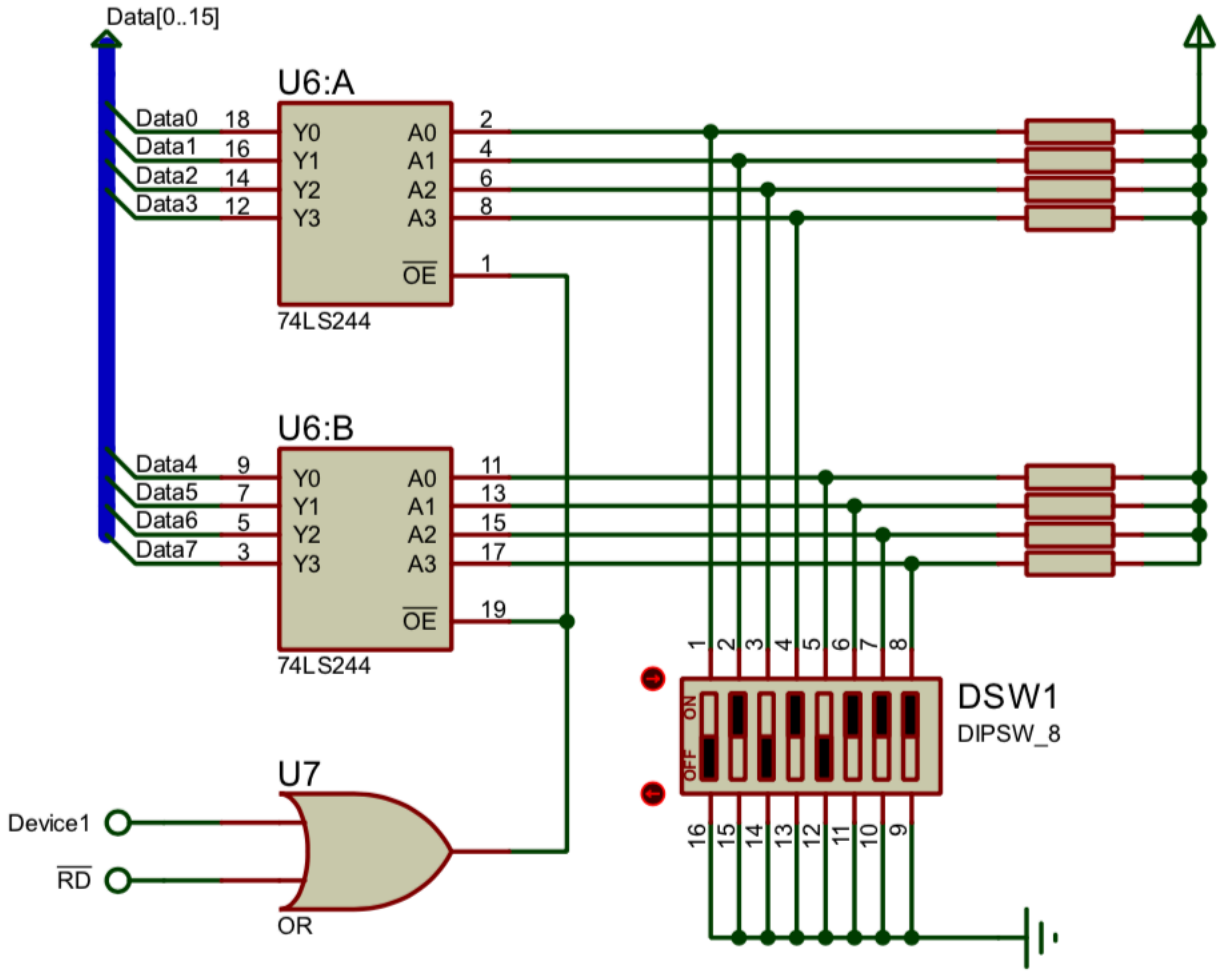
I/O Address Decoding

- When Device1 is activated
  - M/IO = 0, Addr7 = 1, Addr6 = 0, Addr5-3 = 000
  - Port number 80H meets this requirement, but Device1 has more aliases
  - Linear selective decoding，部分译码
- When Device2 is activated
  - M/IO = 0, Addr7 = 1, Addr6 = 0, Addr5-3 = 001
  - Port number 88H meets this requirement, but Device2 has more aliases



Input Port Design

- How does the CPU reads from the switch?

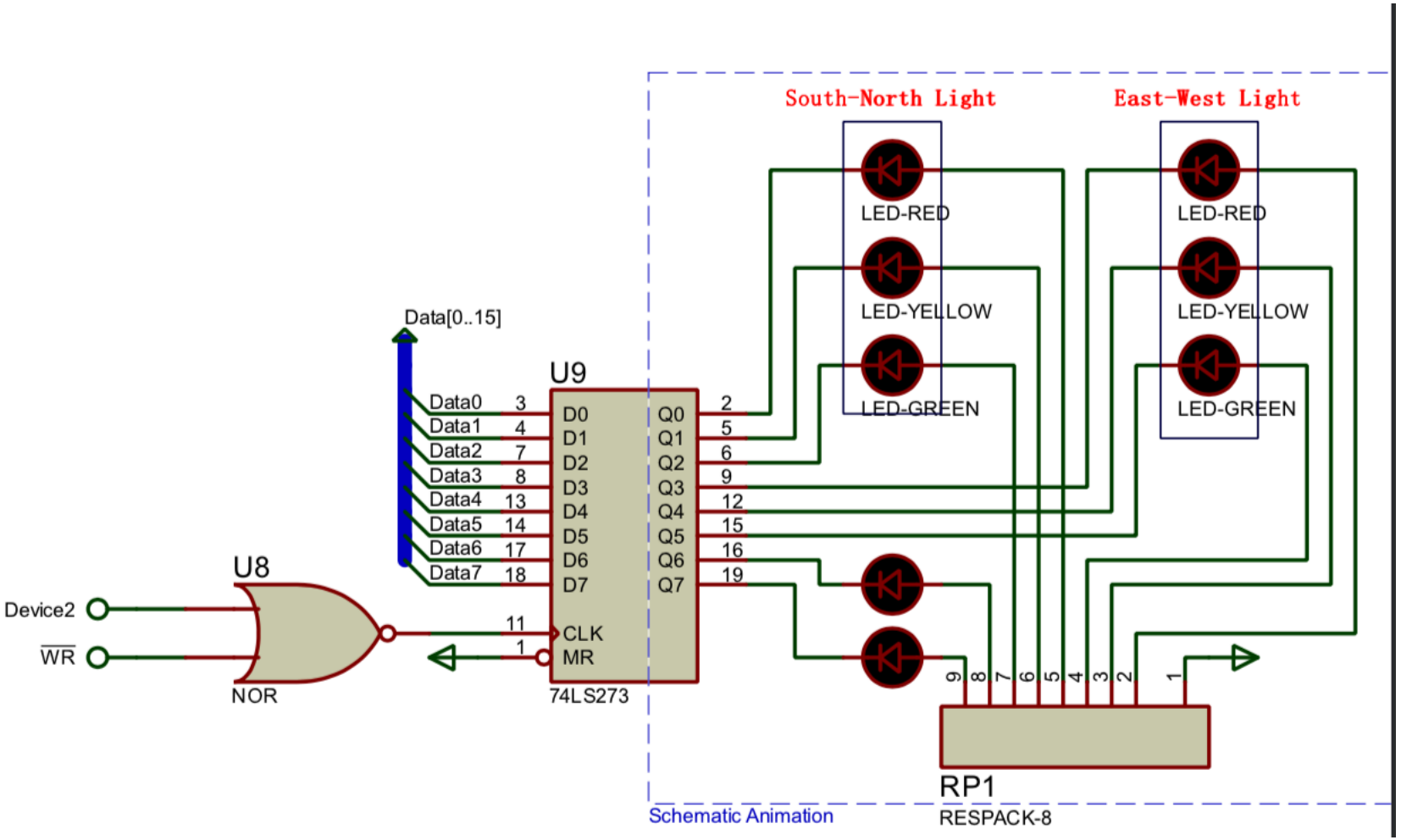


思考题

- How does the CPU reads from the switch?
  - Through executing “IN AL, 80H” instruction
    - Device1 is active
    - The status of the switch is connected to the data bus D7-D0
    - CPU reads D7-D0 to AL
    - If the switch is ON (OFF), the corresponding bit is 0 (1)

Output Port Design

- How does the CPU control the LED light?

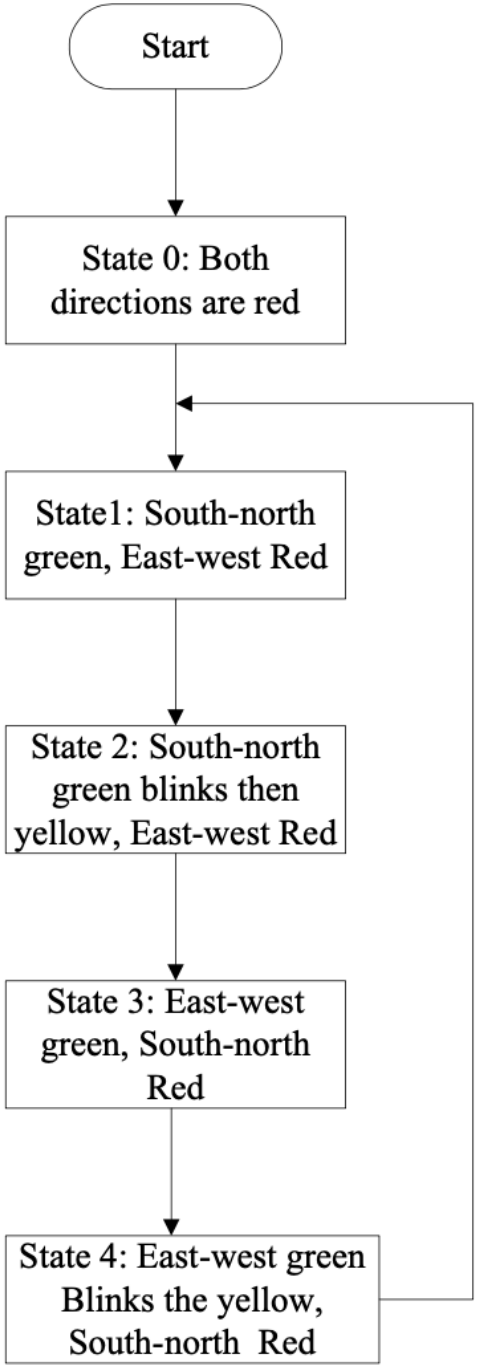


Output Port Design

- How does the CPU control the LED light?
  - Through executing "OUT 88H, AL" instruction
    - Device2 is active
    - The value of AL is put on the data bus D7-D0
    - The latch (74LS273) records the values on D7-D0 and uses it to drive the eight LEDs
    - Value 0 turns the LED on, and value 1 turns the LED off
    - If the switch is ON (OFF), the corresponding bit is 0 (1)
    - Since we use the latch to store the 8-bit value, the status of LEDs do not change until a new value is written

Requirement for This Programming Lab

- Write an assembly program to control the LEDs with three lights (red, green, and yellow), which mimics the traffic light



nt for This Programming Lab

embly program to control the
hree lights (red, green, and yellow),

The state of port 273

State	Meaning	The state of 273 D7---D0
State 0	Both directions are red	××110110 36H
State 1	South-north green, east-west red	××110011 33H
State 2	South-north green blinks then yellow, east-west red	South-north green blinks (on and off), east-west red:

Requirement II

- How should we change the I/O address decoding circuitry to
  - Device1 corresponds to 90H
  - Device2 corresponds to A0H