**CE118-Lab04\_05\_06**

**Thiết kế vi xử lý đơn giản**

**1. Lý thuyết**

Một vi xử lý đơn giản sẽ bao gồm 2 khối chính đó là Controller và Datapath.

• Controller có nhiệm vụ điều khiển đường đi của dữ liệu (điều khiển việc đọc ghi của Register File, …) và điều kiển việc thực hiện tính toán (điều khiển Opcode của khối ALU, …) trong khối Datapath.

• Datapath chứa các khối cần thiết để thực hiện việc tính toán (Register File, ALU,Bộ dịch,…) được điều khiển bởi Controller.

**2. Thực hành**

Sinh viên thực hiện thiết kế một vi xử lý đơn giản dùng để tính toán biểu thức sau:

**D3I3 + D2I2 – D1I1 + D0I0**

Trong đó:

• Dx là 4 ký số cuối của MSSV (Ví dụ 4 ký số cuối MSSV là 6789 thì D3 = 6, D2 = 7, D1 = 8, D0 = 9)

• Ix là 4 ký số được nhập lần lượt tại ngõ vào (Ix có 4 bit, **số Ix sẽ được sinh ngẫu nhiên lúc báo cáo**, sinh viên có thể sử dụng chức năng sinh số ngẫu nhiên trongphần mềm mô phỏng quartus để kiểm tra thiết kế.

Thực hiện thiết kế vi xử lý đơn giản để tính toán biểu thức: **0I3+5I2-5I1+0I0**

Bước 1: Xác định các bước giải quyết bài toán

|  |  |
| --- | --- |
| BƯỚC | CÔNG VIỆC |
| 1 | Data0 🡨 I0 |
| 2 | Data1 🡨 I1 |
| 3 | Data2 🡨 I2 |
| 4 | Data3 🡨 I3 |
| 5 | Temp 🡨 I2 << 2 |
| 6 | Data2 = Data2+Temp |
| 7 | Temp 🡨 I1 << 2 |
| 8 | Data1 = Data1+Temp |
| 9 | Sub 🡨 Data2 – Data1 |
| 10 | O 🡨 Sub |

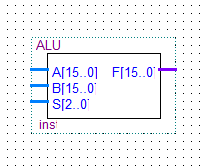
Bước 2: Xác định các khối cần thiết để thực hiện tính toán

a. Register File 8 thanh ghi 16 bit

b. ALU có các phép toán cần thiết cho yêu cầu bài toán (đã hướng dẫn ở Lab 3)

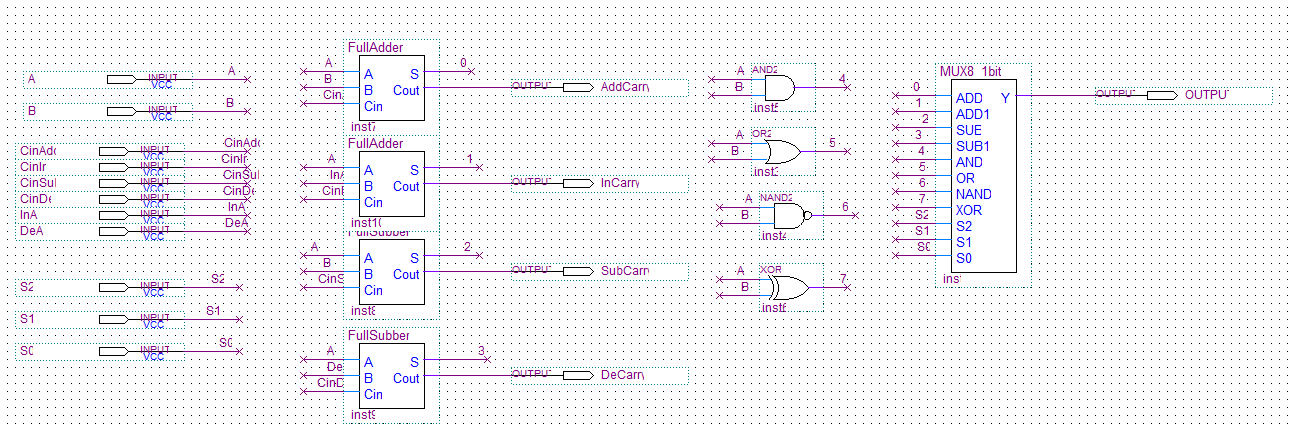
-Bên trong khối ALU gồm có khối AU để thực hiện các phép toán số học như cộng,cộng 1,trừ,trừ. Trong khi đó khối LU được sử dụng để tính toán các phép tính logic như AND,OR,NAND,XOR. Tín hiệu đầu vào S được dùng để lựa chức năng của khối

**ALU**

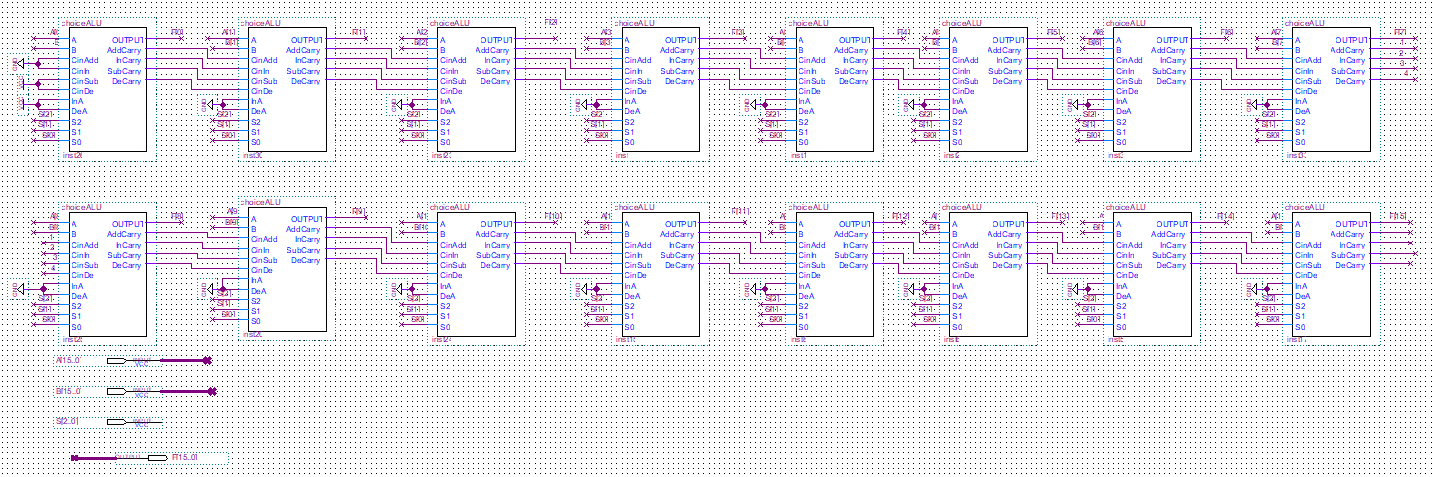
****

|  |  |
| --- | --- |
| Thao tác | Mã |
| Cộng | 000 |
| Cộng 1 | 001 |
| Trừ | 010 |
| Trừ 1 | 011 |
| AND | 100 |
| OR | 101 |
| NAND | 110 |
| XOR | 111 |

**-ALU 1bit**



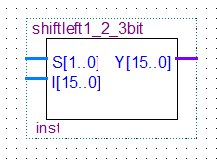
**-ALU 16bit**



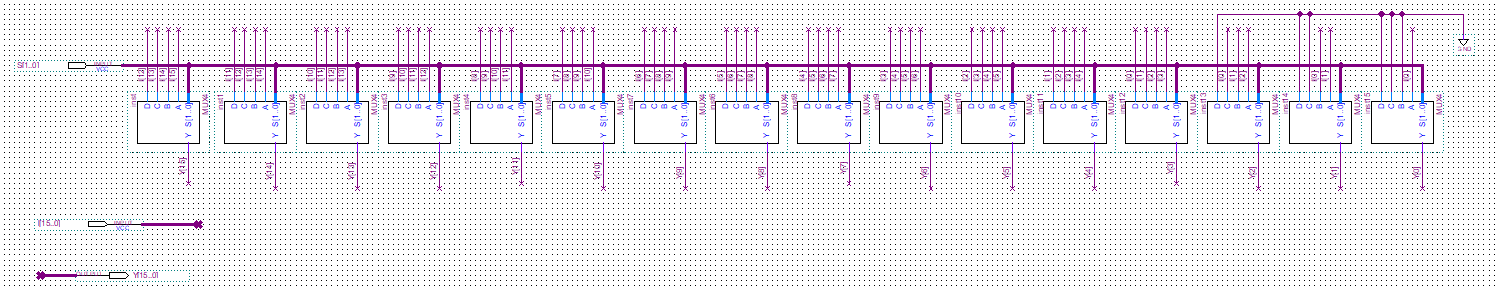
c. Bộ dịch trái: Sử dụng Mux và ghép nối dây phù hợp để tạo ra bộ dịch trái

-Khối ShiftLeft dùng để thực hiện thao tác dịch bit.Khối ShiftLeft được ghép nối 16 bộ MUX4\_1 và sử dụng tín hiệu S để chọn số bit cần dịch từ 0 đến 3

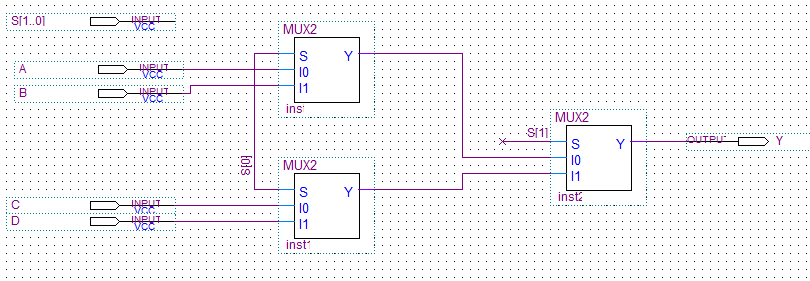
**SHIFT LEFT**

****

|  |  |
| --- | --- |
| Thao tác | Mã |
| <<0 | 00 |
| <<1 | 01 |
| <<2 | 10 |
| <<3 | 11 |

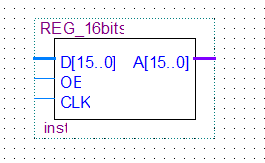


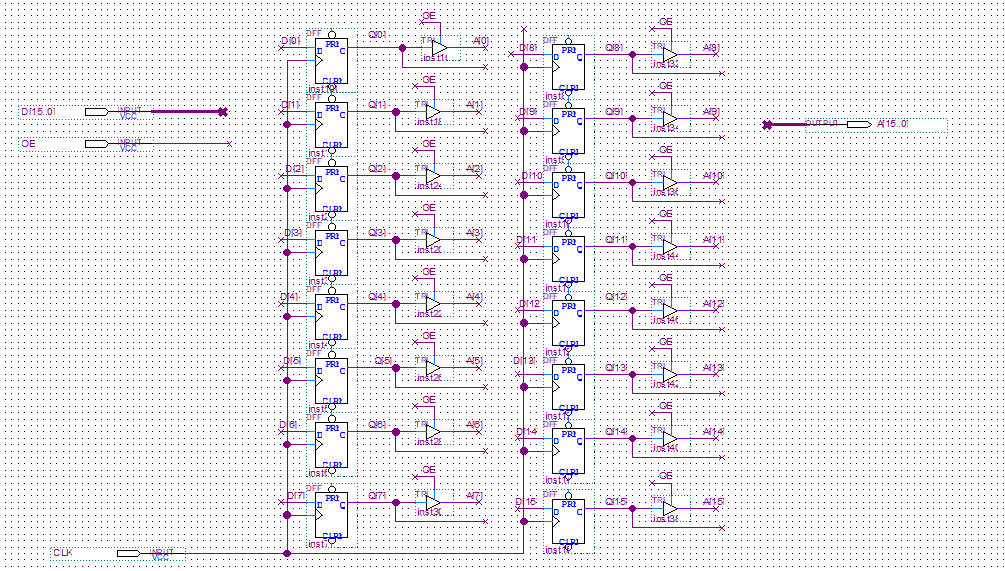
**MUX4\_1**



d. Thanh ghi 16 bit

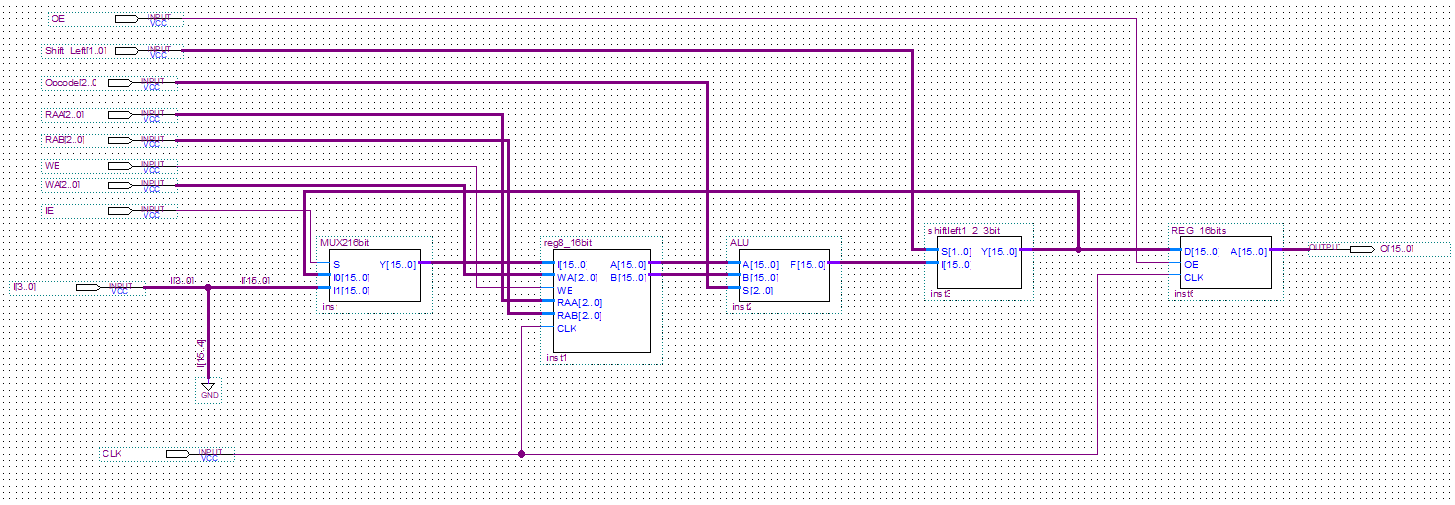
-Ghép nhiều thanh ghi 1 bit lại với nhau để được thanh ghi 16 bit sử dụng tín hiệu OE (output enable) để cho phép/không cho phép xuất kết quả (sử dụng tri-state)





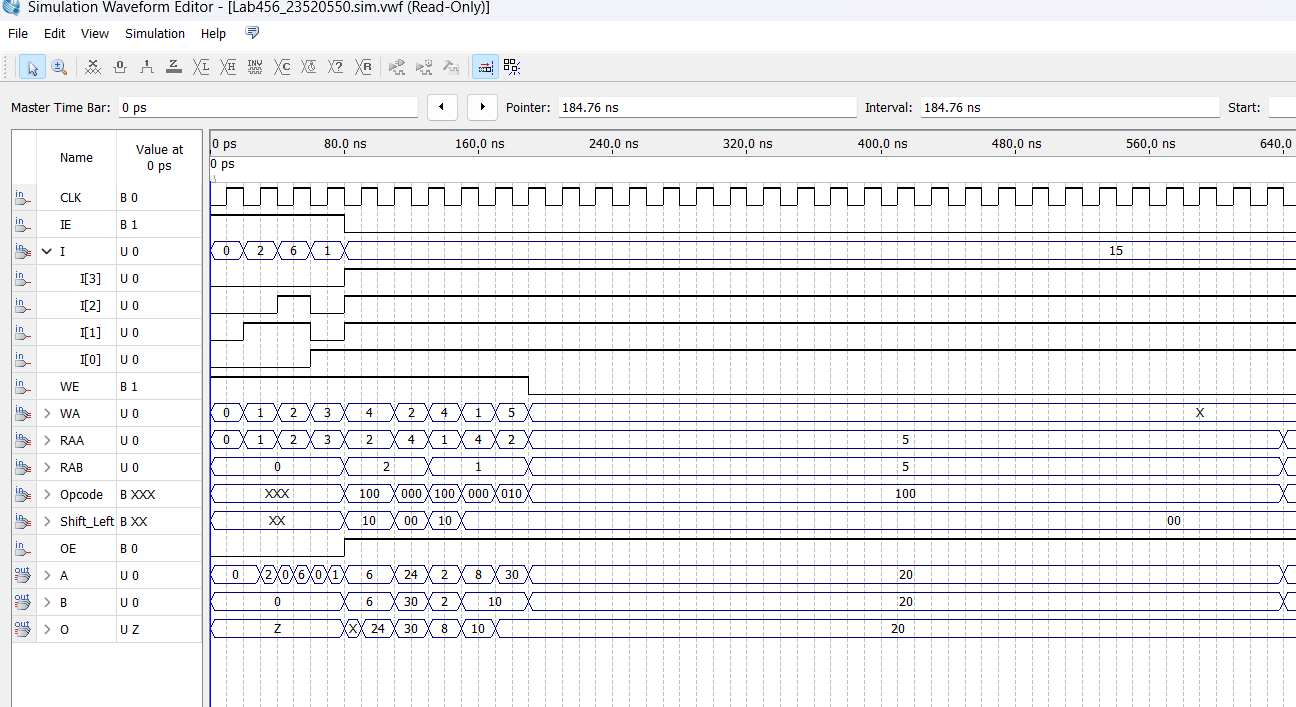
e. Datapath

-Thực hiện ghép nối các khối cần thiết của bài toán lại để có được Datapath



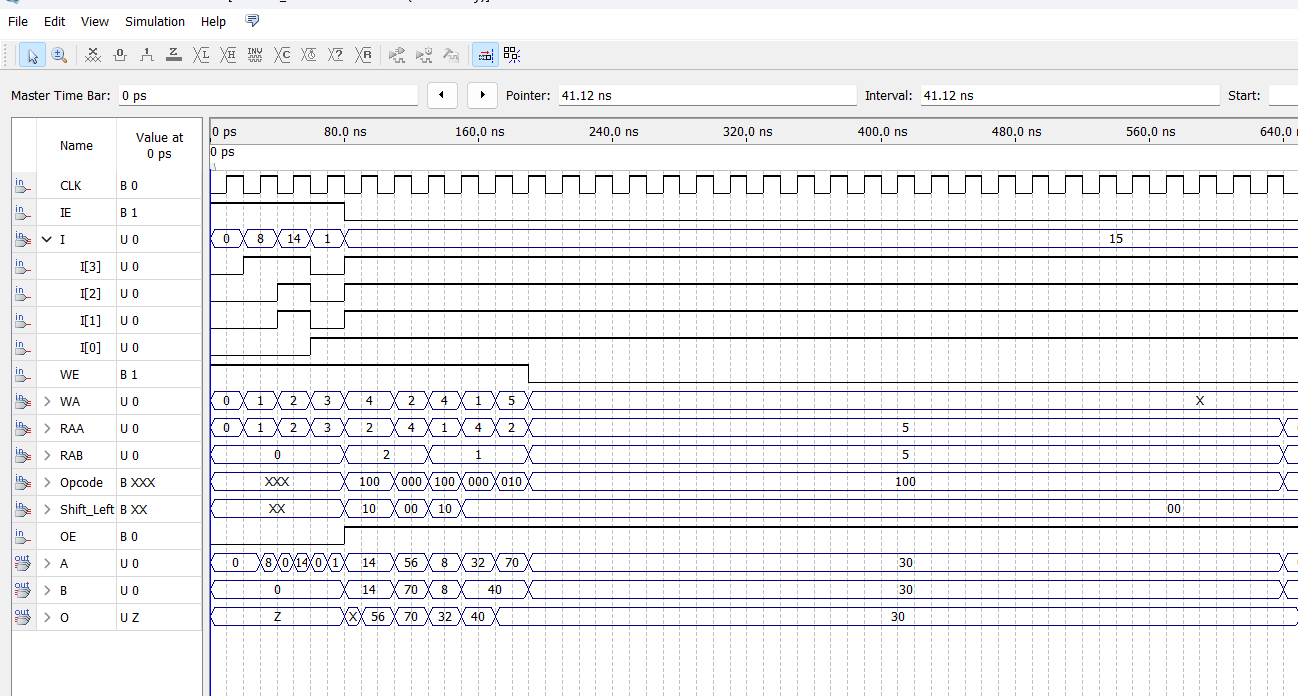
-Thực hiện thử WaveForm với giá trị I3=1,I2=6,I1=2,I0=0

0x1+5x6-5x2+0x0=20



-Thực hiện thử WaveForm với giá trị I3=1,I2=14,I1=8,I0=0

0x1+5x14-5x8+0x0=30



f. Controller: Thiết kế máy trạng thái để điều kiển việc hoạt động các khối trong Datapath.

**CONTROLLER**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Q3** | **Q2** | **Q1** | **Q0** | **IE** | **WE** | **WA** | **RAA** | **RAB** | **Opcode** | **Shift** | **OE** |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 011 | xxx | xxx | xxx | xx | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 010 | xxx | xxx | xxx | xx | 0 |
| 2 | 0 | 0 | 1 | 0 | 1 | 1 | 001 | xxx | xxx | xxx | xx | 0 |
| 3 | 0 | 0 | 1 | 1 | 1 | 1 | 000 | xxx | xxx | xxx | xx | 0 |
| 4 | 0 | 1 | 0 | 0 | 0 | 1 | 100 | 010 | 010 | 100 | 10 | 0 |
| 5 | 0 | 1 | 0 | 1 | 0 | 1 | 010 | 100 | 010 | 000 | 00 | 0 |
| 6 | 0 | 1 | 1 | 0 | 0 | 1 | 100 | 001 | 001 | 100 | 10 | 0 |
| 7 | 0 | 1 | 1 | 1 | 0 | 1 | 001 | 100 | 001 | 000 | 00 | 0 |
| 8 | 1 | 0 | 0 | 0 | 0 | 1 | 101 | 010 | 001 | 010 | 00 | 0 |
| 9 | 1 | 0 | 0 | 1 | 0 | 0 | xxx | 101 | 101 | 100 | 00 | 1 |
| 10 | 1 | 0 | 1 | 0 | 0 | 0 | xxx | 101 | 101 | 100 | 00 | 1 |
| 11 | 1 | 0 | 1 | 1 | 0 | 0 | xxx | 101 | 101 | 100 | 00 | 1 |
| 12 | 1 | 1 | 0 | 0 | 0 | 0 | xxx | 101 | 101 | 100 | 00 | 1 |
| 13 | 1 | 1 | 0 | 1 | 0 | 0 | xxx | 101 | 101 | 100 | 00 | 1 |
| 14 | 1 | 1 | 1 | 0 | 0 | 0 | xxx | 101 | 101 | 100 | 00 | 1 |
| 15 | 1 | 1 | 1 | 1 | 0 | 0 | xxx | 101 | 101 | 100 | 00 | 1 |

g. Thiết kế bộ vi xử lý cho bài toán: Thực hiện ghép nối Controller và Datapath với nhau, sau đó mô phỏng kiểm tra hoạt động.

**0I3+5I2-5I1+0I0**

IE=Q3'Q2'

WE=Q3' + Q2'Q1' Q0’

WA[2]= Q2Q0' + Q3

WA[1]= Q3'Q2'Q1' + Q1'Q0

WA[0]= Q2'Q0' + Q2Q1Q0

RAA[2]= Q0 + Q3Q1+Q3Q2

RAA[1]= Q3'Q1'Q0' + Q2'Q1'Q0'

RAA[0]= Q3Q0+Q3Q2+Q1Q0’

RAB[2]=Q3Q0+Q3Q1+Q3Q2

RAB[1]= Q3'Q1'

RAB[0]= Q1 + Q3

Opcode[2]= Q2Q0' + Q3Q0 + Q3Q1

Opcode[1]= Q2'Q1'Q0'

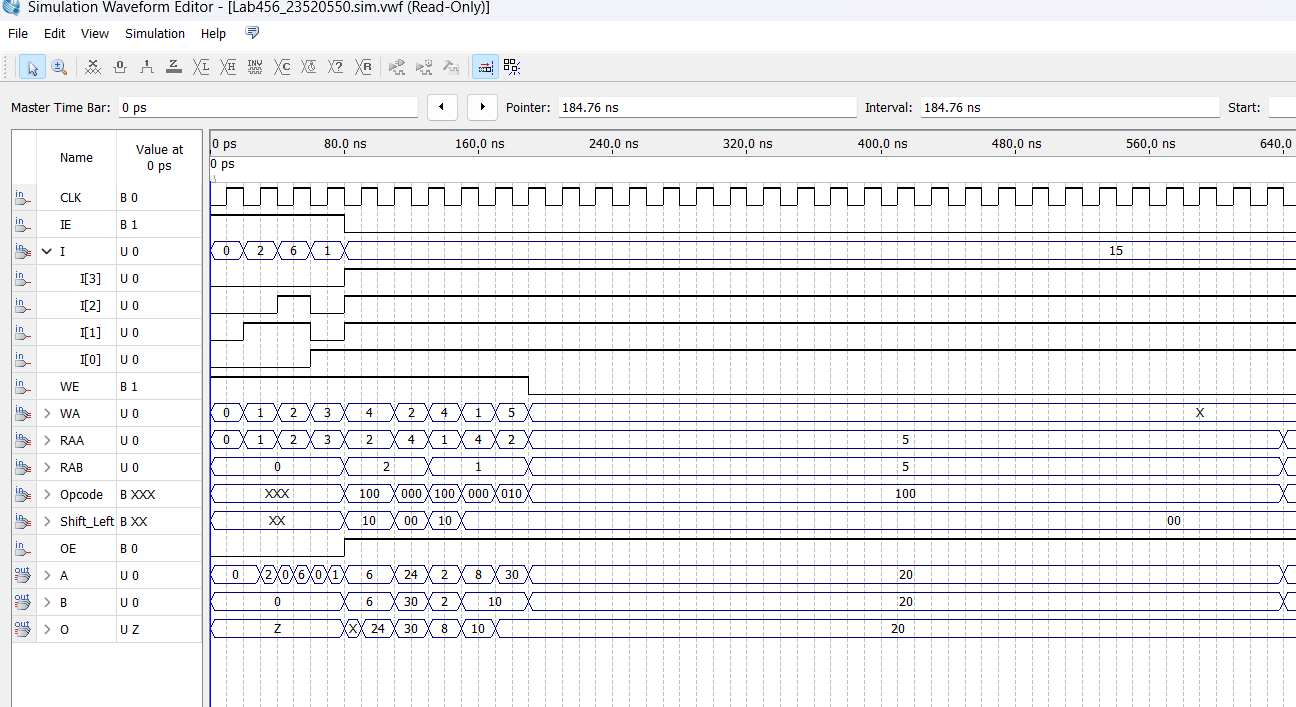
Opcode[0]=0

Shift[1]= Q3'Q0'

Shift[0]=0

OE= Q3Q0 + Q3Q1 + Q3Q2

0x1+5x6-5x2+0x0=20



0x1+5x14-5x8+0x0=30

