

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA KỸ THUẬT MÁY TÍNH



BÁO CÁO PROJECT

MÔN THIẾT KẾ LUẬN LÝ SỐ – CE118

| | |
|----------------------------------|--|
| HỌ VÀ TÊN: MSSV | Vòng Chí Cường 21521910 |
| HỌ VÀ TÊN: MSSV | Vũ Hoàng Tuấn 21521640 |
| LỚP: | CE118.N22.2 |

GIẢNG VIÊN HƯỚNG DẪN:

Trương Văn Cương

TP. HỒ CHÍ MINH – Tháng 7 năm 2023

1) Kiến thức tổng quát

1. Số dấu phẩy tĩnh (fixed point number)

Ví dụ : $1,24_{(10)} = 1.0011110101110000101$

Dấu phẩy tĩnh: Có một vị trí cố định ngăn cách giữa phần nguyên và phần lẻ.

2. Số dấu phẩy động (floating point number)

Số được biểu diễn dưới dạng: $X = M \times R^E$

Trong đó :

- M : Mantissa
- R : Radix
- E: Exponent

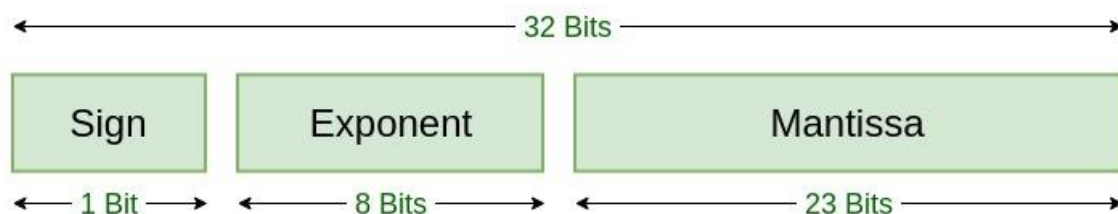
Ví dụ:

$$\diamond 14.2268_{(10)} = 1.42268 \times 10^1 = 1.42268 \text{ E}+01$$

$$\diamond 0.0154_{(10)} = 1.54 \times 10^{-2} = 1.54 \text{ E}-02$$

Vị trí dấu phẩy trong biểu diễn bình thường do phần bậc định ra trên phần định trị nên gọi là dấu phẩy động.

3. Floating-point binary format (Intel)



Single Precision
IEEE 754 Floating-Point Standard

Trong đó:

- Singed bit (1 bit) : 0 là dương, 1 là âm
- Bias exponent (characteristic): $c = E + 2^{t-1} - 1$
- E : là số mũ cần chứa
- t : là độ dài bit của trường Exponent

Ví dụ:

Ta cần chứa 6 bit mà phần Exponent có độ dài là 8 bit

$$\rightarrow c = 6 + 2^{8-1} - 1 = 6 + 127 = 133_{(10)} = 1000\ 0101_{(2)}$$

4. Chuyển Decimal sang Binary Floating-point

- Bước 1: Xác định dấu
- Bước 2: Chuyển đổi phần nguyên thành số nhị phân không dấu
- Bước 3: Chuyển đổi phần thập phân thành số nhị phân
- Bước 4: Chuẩn hóa giá trị bằng cách điều chỉnh phần mũ
- Bước 5: Thêm độ lệch vào phần mũ
- Bước 6: Chuyển đổi phần mũ có độ lệch thành số nhị phân không dấu
- Bước 7: Xác định các bit cuối cùng cho phần lưu giữa
- Bước 8: Tổng hợp tất cả lại

Ví dụ: Chuyển 1.375 sang 32-bit Floating-point

Convert to binary: $12.375_{(10)} = 1100.011_{(2)}$

- 1) Sign bit : 0
- 2) Add exponent part: $1100.011 = 1100.011 \times 2^0$
- 3) Normalize: $1100.011 \times 2^0 = 1.100011 \times 2^3$
- 4) Exponent: $3 + 2^{8-1} - 1 = 3 + 127 = 130_{(10)} = 1000\ 0010_{(2)}$
- 5) Mantissa: $100011 \rightarrow 100011000000000000000000$

\rightarrow Tổng hợp 3 phần : 0 10000010 100011000000000000000000

5. Chuyển Binary Floating-point sang Decimal

- Bước 1: Trích xuất bit dấu

- Bước 2: Trích xuất phần mũ
- Bước 3: Đồng bộ hóa phần mũ
- Bước 4: Chuyển đổi phần dạng thập phân
- Bước 5: Tính toán độ lớn của giá trị tổng quát
- Bước 6: Đưa vào biểu diễn của bit dấu.

Ví dụ: Chuyển 32-bit Floating-point 0 10000001 00001001100110011001101 sang Decimal

1) Sign bit : 0 → Số dương.

2) Exponent: $10000001_{(2)} = 129_{(10)} \rightarrow c = 129 - 2^{8-1} - 1 = 129 - 127 = 2$

3) Mantissa: 1.00001001100110011001101

4) De-normalize: $1.00001001100110011001101 \times 2^2 = 100.001001100110011001101$

5) Convert: $100_{(2)} = 4_{(10)} ; 001001100110011001101_{(2)} \approx 0.15_{(10)}$

→ Result = 4.15

6. Nhân hai số Floating point (Floating point Multiplication)

$$S_a.M_a.2^{E_a} \times S_b.M_b.2^{E_b} = (S_a \oplus S_b) . (M_a \times M_b). (2^{(E_a+E_b) - \text{bias}})$$

- Trích xuất dấu của kết quả từ hai bit dấu.
- Cộng hai phần mũ (E). Trừ đi thành phần độ lệch (bias) từ tổng này.
- Nhân mantissa của b (M_b) với mantissa của a (M_a), lưu ý các bit ẩn.
- Nếu bit MSB của tích 1' thì dịch kết quả sang phải 1 bit.
- Do đó, phần mũ được tăng lên 1 đơn vị do kết quả dịch sang phải 1 bit.

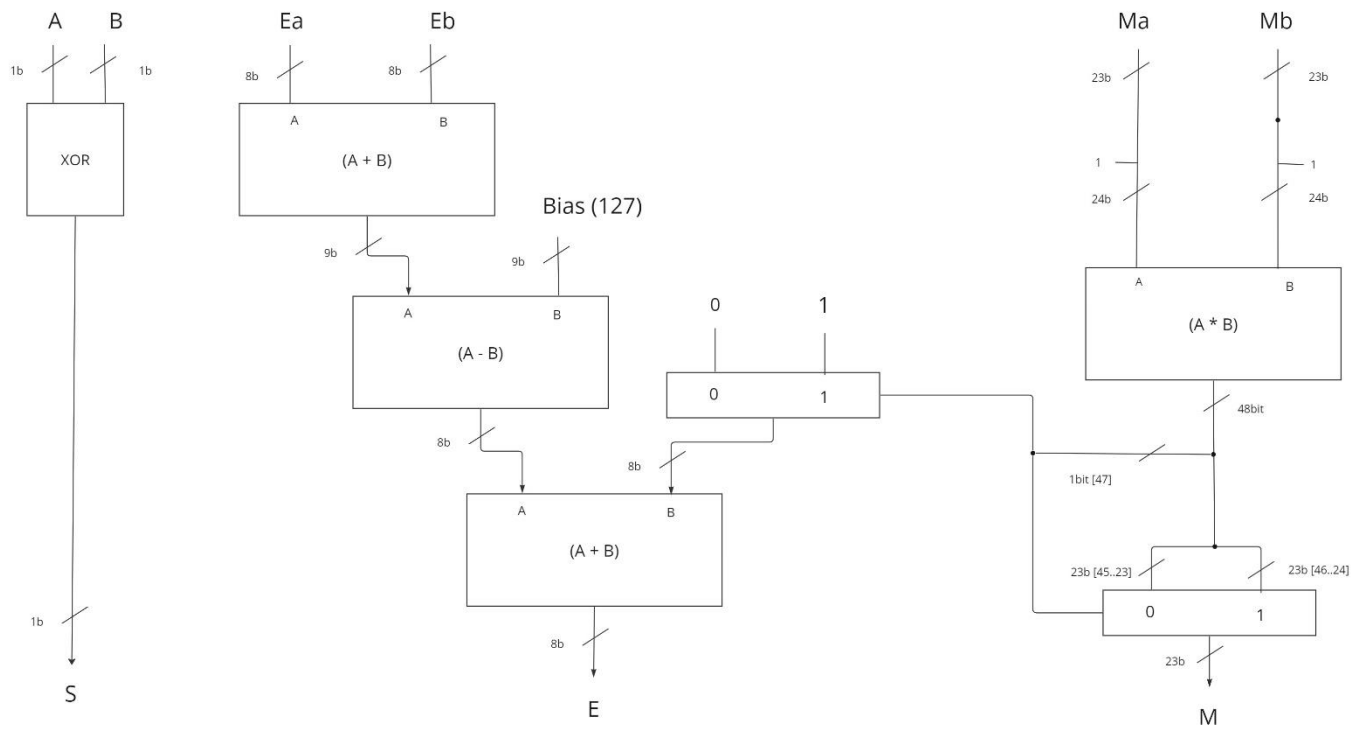
7. Chia hai số Floating Point (Floating point Division)

$$\frac{S_a \cdot M_a \cdot 2^{E_a}}{S_b \cdot M_b \cdot 2^{E_b}} = (S_a \oplus S_b) \cdot \left(\frac{M_a}{M_b}\right) \cdot (2^{(E_a - E_b) + \text{bias}})$$

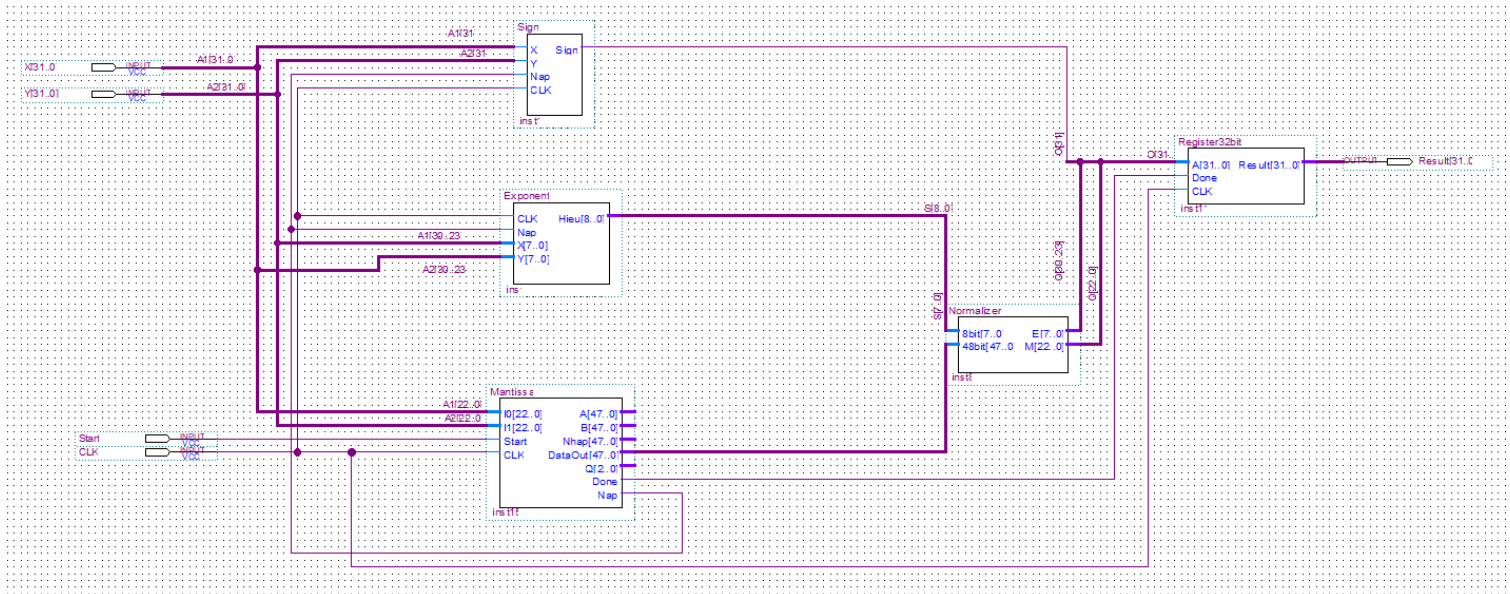
- Trích xuất dấu của kết quả từ hai bit dấu.
- Tìm giá trị tuyệt đối của hiệu hai số mũ (E). Cộng hiệu của E với bias.
- Chia mantissa của a (M_a) cho mantissa của b (M_b) và xem xét các bit ẩn.
- Nếu có một số 0 dẫn đầu thì chuẩn hóa kết quả bằng cách dịch trái.
- Do việc chuẩn hóa, số mũ được giảm đi theo số lần dịch trái.

2) Sơ đồ mạch

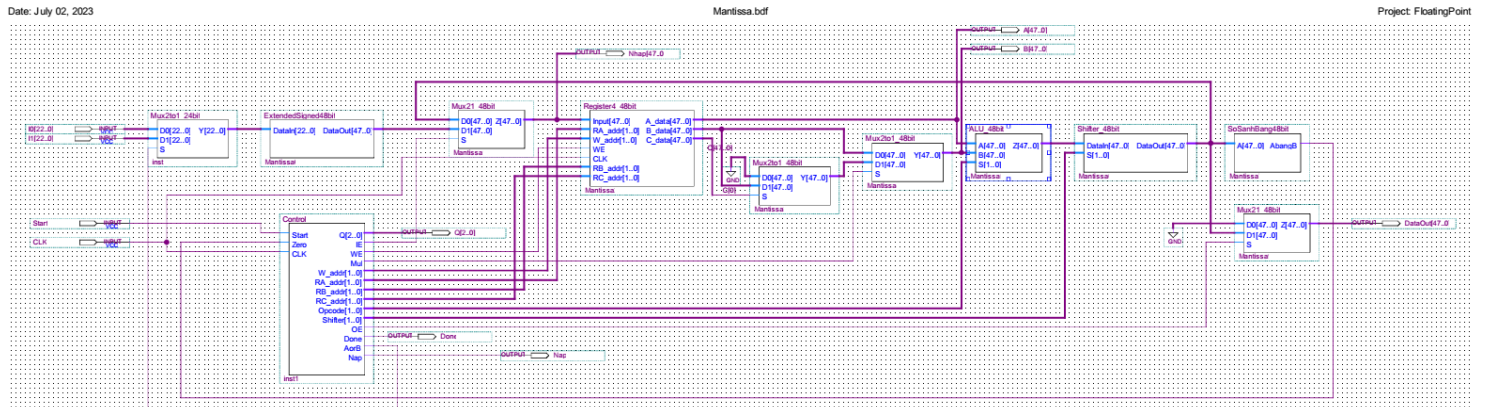
Floating point Multiplication



Mạch nhân floating point:



Mantissa:



1. Bảng ALU

| Opcode (S ₁ S ₀) | Chức năng |
|---|-----------|
| 00 | ADD |
| 01 | SUB |
| 10 | A + 1 |
| 11 | AND |

2. Bảng Shifter

| S ₁ S ₀ | Chức năng |
|-------------------------------|-------------------------|
| 00 | Không dịch |
| 01 | Dịch trái 1 bit |
| 10 | Dịch phải luận lý 1 bit |
| 11 | Output = 0 |

3. Phần Control

Bảng trạng thái – thực thi:

| TTHT | TTKT | Các biến trong khối đường dữ liệu |
|------|--|-----------------------------------|
| | Điều kiện, trạng thái | Điều kiện, thực thi |
| S0 | Start = 0, S0 Start = 1, S1 | , Done = 0 , Outport = 0 |
| S1 | S2 | Multipler = Import |
| S2 | S3 | Multiplicand = Import |
| S3 | S4 | Result = 0 |
| S4 | S5 | Result = Result + Multiplicand |
| S5 | S6 | Multiplicand << 1 |
| S6 | Zero = 0 (Multipler ≠ 0), S4 Zero = 1 (Multipler = 0), S7 | Multipler >> 1 |
| S7 | S0 | , Done = 1 , Outport = Result |

Bảng chuyển đổi:

| | TTHT | TTKT | | | | D-FF | | | |
|-----------|--|--|-----|-----|-----|--|-----|-----|-----|
| | Q₂Q₁Q₀ | Q₂⁺Q₁⁺Q₀⁺ | | | | D₂D₁D₀ | | | |
| | | Start, Zero | | | | Start, Zero | | | |
| | | 00 | 01 | 10 | 11 | 00 | 01 | 10 | 11 |
| S0 | 000 | 000 | 000 | 001 | 001 | 000 | 000 | 001 | 001 |
| S1 | 001 | 010 | 010 | 010 | 010 | 010 | 010 | 010 | 010 |
| S2 | 010 | 011 | 011 | 011 | 011 | 011 | 011 | 011 | 011 |
| S3 | 011 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| S4 | 100 | 101 | 101 | 101 | 101 | 101 | 101 | 101 | 101 |
| S5 | 101 | 110 | 110 | 110 | 110 | 110 | 110 | 110 | 110 |
| S6 | 110 | 100 | 111 | 100 | 111 | 100 | 111 | 100 | 111 |
| S7 | 111 | 000 | 000 | 000 | 000 | 000 | 000 | 000 | 000 |

Biểu thức luận lý sau khi rút gọn :

$$D2 = (Q2 \cdot (|Q1 + |Q0)) + (|Q2 \cdot Q1 \cdot Q0)$$

$$D1 = ((Q1 \cdot |Q0) \cdot (|Q2 + Zero)) + (|Q1 \cdot Q0)$$

$$D0 = ((|Q1 \cdot |Q0) \cdot (Start + Q2)) + ((Q1 \cdot |Q0) \cdot (|Q2 + Zero))$$

Bảng ngõ ra điều khiển

| TTHT | Q ₂ Q ₁ Q ₀ | IE | WE | Mul | W_addr | A | B | C | Opcode | Shifter | OE | AorB | Nạp |
|------|--|----|----|-----|-------------------|----|----|----|----------|------------------|----|------|-----|
| S0 | 000 | 0 | 0 | 0 | XX | XX | XX | XX | XX | XX | 0 | X | 0 |
| S1 | 001 | 1 | 1 | X | Multiplier 00 | XX | XX | XX | XX | XX | 0 | 1 | 1 |
| S2 | 010 | 1 | 1 | X | Multiplicand 01 | XX | XX | XX | XX | XX | 0 | 0 | 0 |
| S3 | 011 | 0 | 1 | 0 | Result 10 | 00 | 00 | XX | Sub 01 | 00 | 0 | X | 0 |
| S4 | 100 | 0 | 1 | 1 | 10 | 10 | 01 | 00 | ADD 00 | 00 | 0 | X | 0 |
| S5 | 101 | 0 | 1 | 0 | 01 | 01 | 01 | XX | AND 11 | Shift Left 01 | 0 | X | 0 |
| S6 | 110 | 0 | 1 | 0 | 00 | 00 | 00 | XX | AND 11 | Shift Right 10 | 0 | X | 0 |
| S7 | 111 | 0 | 0 | 0 | X | 10 | 10 | XX | AND 11 | 00 | 1 | X | 0 |

Biểu thức luận lý sau khi rút gọn :

Opcode[1] = Q2 · (Q0 + Q1)

Opcode[0] = Q0 + Q1

Shifter[1] = (Q1 · |Q0)

Shifter[0] = (|Q1 · Q0)

OE = Q2 · Q1 · Q0

IE = Q2’Q1’Q0 + Q2’Q1Q0’

WE = Q2Q1’ + Q2’Q0 + Q1Q0’

Mul = Q2Q1’Q0’

W_addr[1] = Q1’Q0’ + Q1Q0

W_addr[0] = Q2Q0 + Q2’Q0’

RA_addr[1] = Q1’Q0’ + Q2Q1Q0

RA_addr[0] = Q1’Q0

RB_addr[1] = Q2Q1Q0

RB_addr[0] = Q1’

RC_addr[1] = 0

$RC_addr[0] = 0$

$A \text{ or } B = Q_1'$

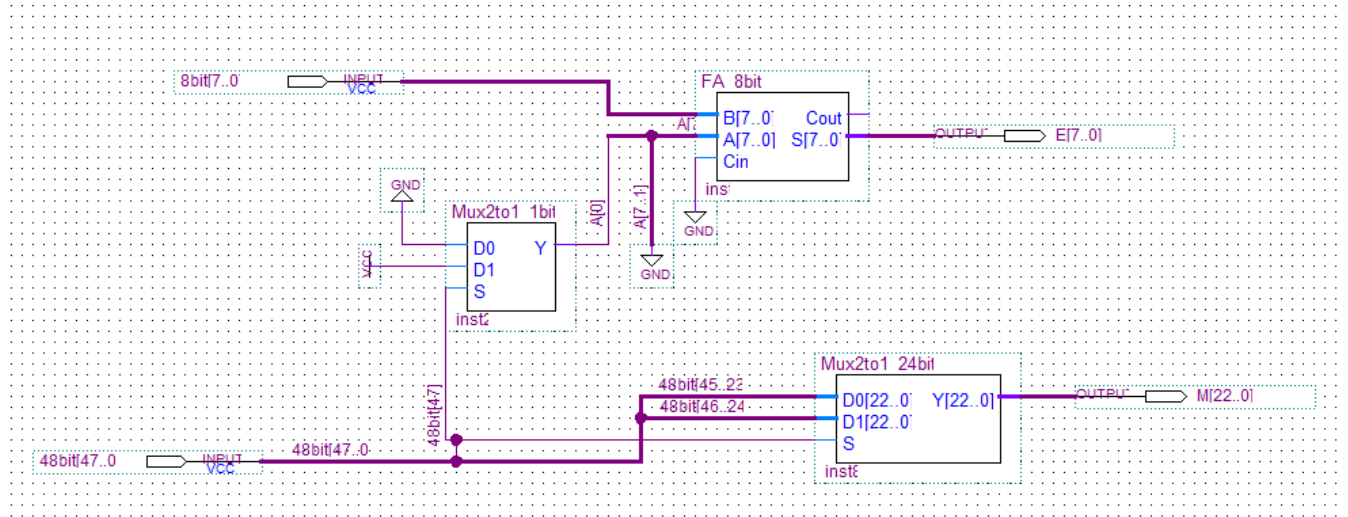
$Nap = Q_2'Q_1'Q_0$

Bảng ngõ ra trạng thái:

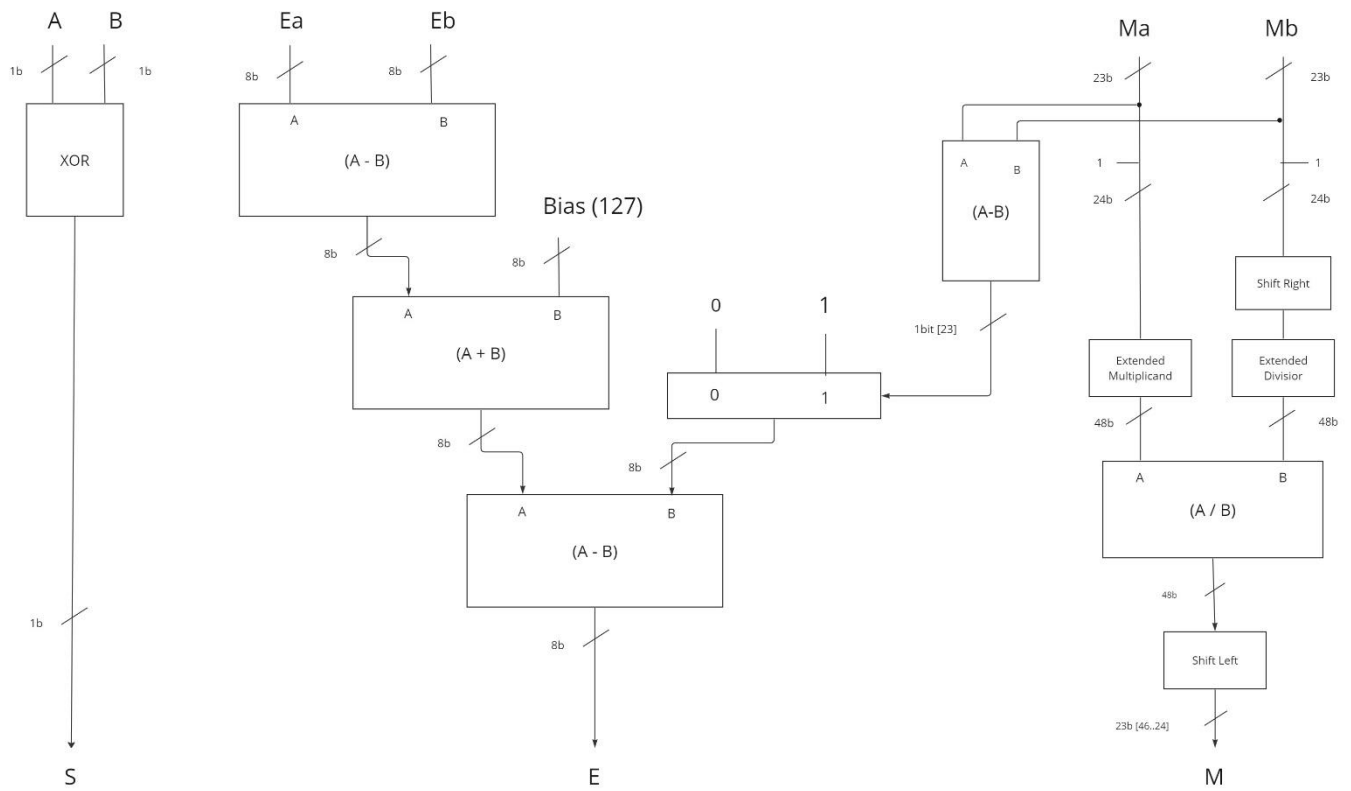
| TTHT | $Q_2Q_1Q_0$ | Done |
|-------------|-------------------------------|-------------|
| S0 | 000 | 0 |
| S1 | 001 | 0 |
| S2 | 010 | 0 |
| S3 | 011 | 0 |
| S4 | 100 | 0 |
| S5 | 101 | 0 |
| S6 | 110 | 0 |
| S7 | 111 | 1 |

$Done = Q_2 \cdot Q_1 \cdot Q_0$

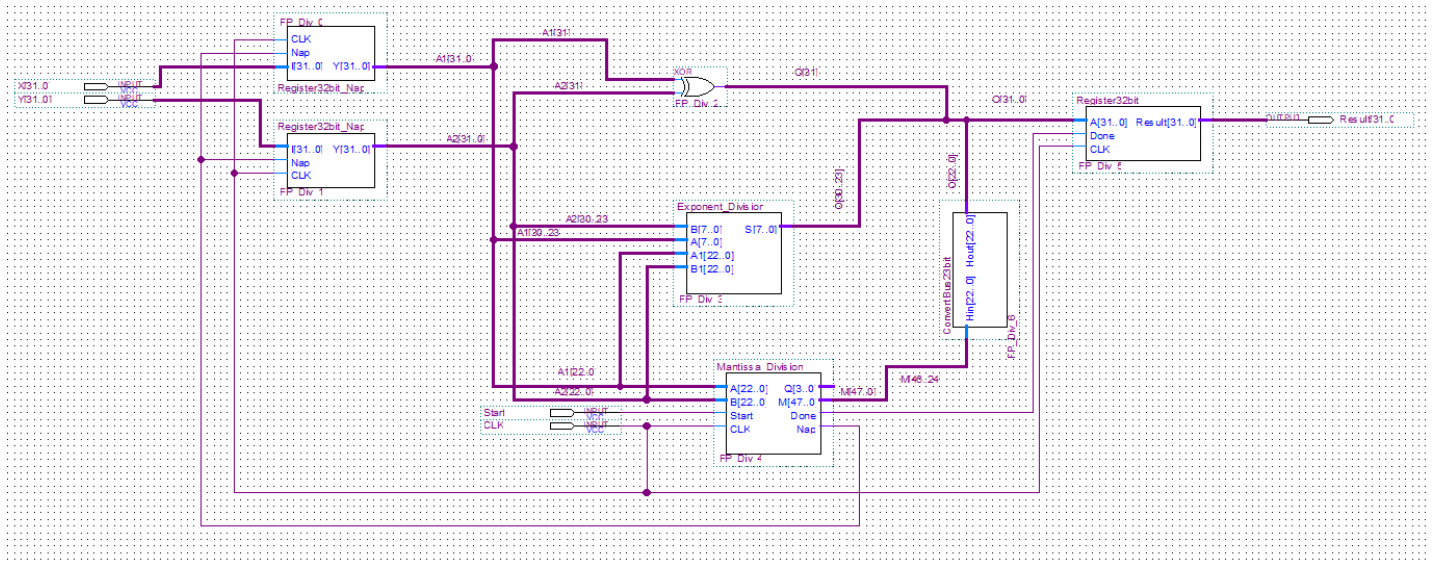
4. Normalizer



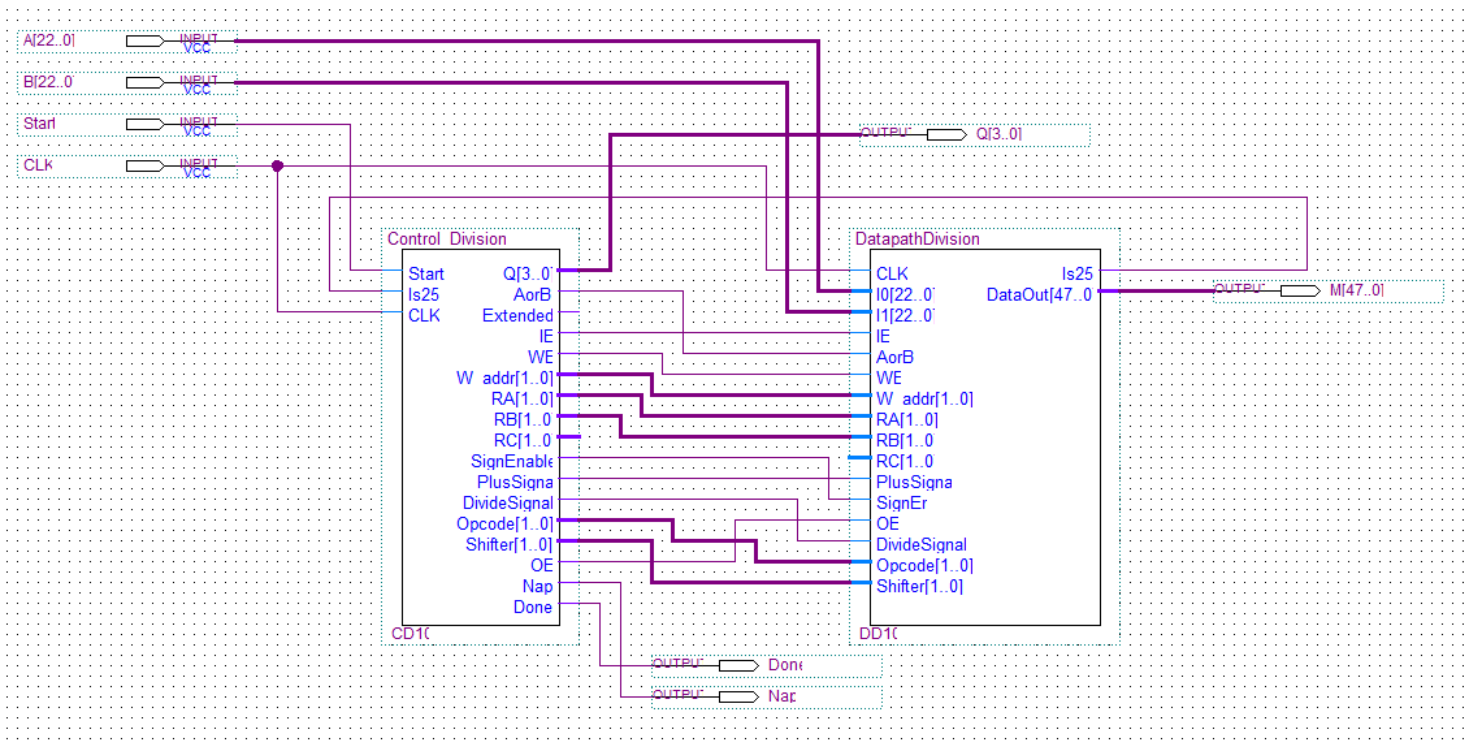
Floating point Division



Mạch chia floating point

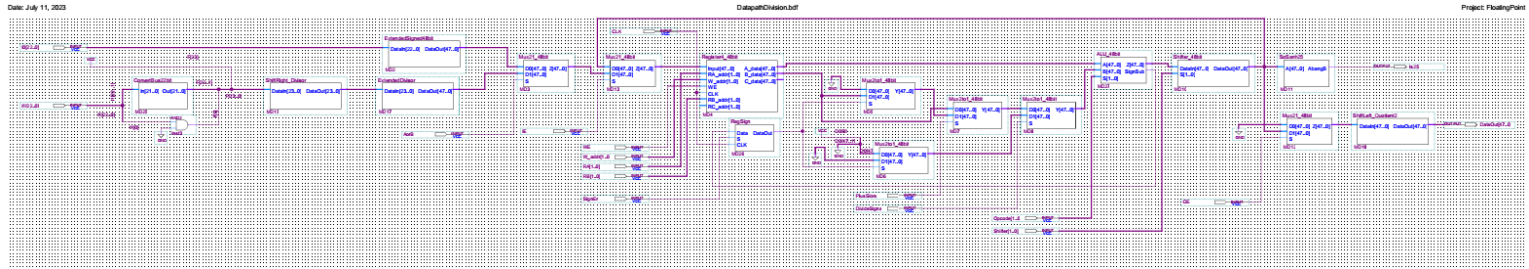


Mantissa:



Datapath:

Date: July 11, 2023



1. Bảng ALU

| Opcode (S_1S_0) | Chức năng |
|-------------------------------------|------------------|
| 00 | ADD |
| 01 | SUB |
| 10 | $A + 1$ |
| 11 | AND |

2. Bảng Shifter

| S_1S_0 | Chức năng |
|----------------------------|-------------------------|
| 00 | Không dịch |
| 01 | Dịch trái 1 bit |
| 10 | Dịch phải luận lý 1 bit |
| 11 | Output = 0 |

3. Phần Control

Bảng trạng thái – thực thi:

| TTHT | TTKT | Các biến trong khối đường dữ liệu |
|------|---|--|
| | Điều kiện, trạng thái | Điều kiện, thực thi |
| S0 | Start = 0, S0 Start = 1, S1 | , Done = 0 , Outport = 0 |
| S1 | S2 | Remainder = Import |
| S2 | S3 | Divisor = Inport |
| S3 | S4 | Quotient = 0 |
| S4 | S5 | i = 0 |
| S5 | S6 | Remainder = Remainder – Divisor |
| S6 | S7 | Quotient << 1 |
| S7 | S8 | Ramainder[47] = 0, Quotient += 1 Ramainder[47] = 1, Quotient += 0 |
| S8 | S9 | Ramainder[47] = 0, Ramainder += 0 Ramainder[47] = 1, Ramainder += Divisor |
| S9 | S10 | Divisor >> 1 |
| S10 | Is25 = 0 (i ≠ 25), S5 Is25 = 1 (i = 25), S11 | i = i + 1 |
| S11 | S0 | , Done = 1 , Outport = Quotient |

Bảng chuyển đổi:

| | TTHT | TTKT | | | | D-FF | | | |
|------------|---|---|------|------|------|---|------|------|------|
| | Q₃Q₂Q₁Q₀ | Q₃⁺Q₂⁺Q₁⁺Q₀⁺ | | | | D₃D₂D₁D₀ | | | |
| | | Start, Is25 | | | | Start, Is25 | | | |
| | | 00 | 01 | 10 | 11 | 00 | 01 | 10 | 11 |
| S0 | 0000 | 0000 | 0000 | 0001 | 0001 | 0000 | 0000 | 0001 | 0001 |
| S1 | 0001 | 0010 | 0010 | 0010 | 0010 | 0010 | 0010 | 0010 | 0010 |
| S2 | 0010 | 0011 | 0011 | 0011 | 0011 | 0011 | 0011 | 0011 | 0011 |
| S3 | 0011 | 0100 | 0100 | 0100 | 0100 | 0100 | 0100 | 0100 | 0100 |
| S4 | 0100 | 0101 | 0101 | 0101 | 0101 | 0101 | 0101 | 0101 | 0101 |
| S5 | 0101 | 0110 | 0110 | 0110 | 0110 | 0110 | 0110 | 0110 | 0110 |
| S6 | 0110 | 0111 | 0111 | 0111 | 0111 | 0111 | 0111 | 0111 | 0111 |
| S7 | 0111 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 |
| S8 | 1000 | 1001 | 1001 | 1001 | 1001 | 1001 | 1001 | 1001 | 1001 |
| S9 | 1001 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 | 1010 |
| S10 | 1010 | 0101 | 1011 | 0101 | 1011 | 0101 | 1011 | 0101 | 1011 |
| S11 | 1011 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |
| S12 | 1100 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |
| S13 | 1101 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |
| S14 | 1110 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |
| S15 | 1111 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |

Biểu thức luận lý sau khi rút gọn :

$$D3 = ((Q3 \cdot |Q2) \cdot ((Is25 \cdot |Q0) + |Q1)) + (|Q3 \cdot Q2 \cdot Q1 \cdot Q0)$$

$$D2 = ((|Q3 \cdot Q2) \cdot (|Q1 + |Q0)) + (|Is25 \cdot Q3 \cdot |Q2 \cdot Q1 \cdot |Q0) + (|Q3 \cdot |Q2 \cdot Q1 \cdot Q0)$$

$$D1 = (|Q3 \cdot ((|Q1 \cdot Q0) + (Q1 \cdot |Q0))) + (|Q2 \cdot ((|Q1 \cdot Q0) + (Is25 \cdot Q1 \cdot |Q0)))$$

$$D0 = ((|Q2 \cdot |Q0) \cdot (Q1 + Q3 + \text{Start})) + ((|Q3 \cdot |Q0) \cdot (Q1 + (Q2 \cdot |Q1))))$$

Bảng ngõ ra điều khiển :

| | AorB | IE | WE | W_addr | RA | RB | Sign Enable | Plus Signal | Divide Signal | Opcode | Shifter | OE | Nạp |
|------------|------|----|----|----------------|----|----|----------------|----------------|------------------|------------|------------------|----|-----|
| S0 | X | 0 | 0 | XX | XX | XX | 0 | X | X | XX | XX | 0 | 1 |
| S1 | 0 | 1 | 1 | Remainder 00 | XX | XX | 0 | X | X | XX | XX | 0 | 0 |
| S2 | 1 | 1 | 1 | Divisor 01 | XX | XX | 0 | X | X | XX | XX | 0 | 0 |
| S3 | X | 0 | 1 | Quotient 10 | 00 | 00 | 0 | 0 | 0 | Sub 01 | 00 | 0 | 0 |
| S4 | X | 0 | 1 | i 11 | 00 | 00 | 0 | 0 | 0 | Sub 01 | 00 | 0 | 0 |
| S5 | X | 0 | 1 | 00 | 00 | 01 | 1 | 0 | 0 | Sub 01 | 00 | 0 | 0 |
| S6 | X | 0 | 1 | 10 | 10 | 10 | 0 | 0 | 0 | AND 11 | Shift Left 01 | 0 | 0 |
| S7 | X | 0 | 1 | 10 | 10 | XX | 0 | X | 1 | ADD 00 | 00 | 0 | 0 |
| S8 | X | 0 | 1 | 00 | 00 | 01 | 0 | 1 | 0 | ADD 00 | 00 | 0 | 0 |
| S9 | X | 0 | 1 | 01 | 01 | 01 | 0 | 0 | 0 | AND 11 | Shift Right 10 | 0 | 0 |
| S10 | X | 0 | 1 | 11 | 11 | XX | 0 | 0 | 0 | ADD 1 10 | 00 | 0 | 0 |
| S11 | X | 0 | 0 | 10 | 10 | 10 | 0 | 0 | 0 | AND 11 | 00 | 1 | 0 |

Biểu thức luận lý sau khi rút gọn :

$$AorB = Q1$$

$$IE = (|Q3 \cdot |Q2) \cdot ((Q1 \cdot |Q0) + (|Q1 \cdot Q0))$$

$$WE = (|Q3 \cdot Q1) + (Q3 \cdot |Q0) + (|Q1 \cdot Q0) + Q2$$

$$W_addr[1] = (Q1 \cdot (Q0 + Q3)) + (Q2 \cdot |Q0)$$

$$W_addr[0] = (|Q1 \cdot ((Q3 \cdot Q0) + (|Q3 \cdot |Q0))) + (|Q2 \cdot Q1 \cdot |Q0)$$

$$RA[1] = Q1 \cdot (Q2 + Q3)$$

$$RA[0] = (|Q2 \cdot Q1 \cdot |Q0) + (Q3 \cdot |Q1 \cdot Q0)$$

$$RB[1] = Q1 \cdot (|Q0 + Q3)$$

$$RB[0] = |Q1 \cdot (Q3 + Q0)$$

$$SignEn = |Q3 \cdot Q2 \cdot |Q1 \cdot Q0$$

$$Plus\ Signal = (Q3 \cdot |Q1 \cdot |Q0)$$

$$Divide\ Signal = (Q2 \cdot Q1 \cdot Q0)$$

$$Opcode[1] = (Q1 \cdot |Q0) + (Q3 \cdot Q0)$$

$$Opcode[0] = (|Q3 \cdot (|Q0 + |Q1)) + (|Q2 \cdot Q0)$$

$$Shifter[1] = (Q3 \cdot |Q1 \cdot Q0)$$

$$Shifter[0] = (|Q3 \cdot Q1 \cdot |Q0)$$

$$OE = (Q3 \cdot Q1 \cdot Q0)$$

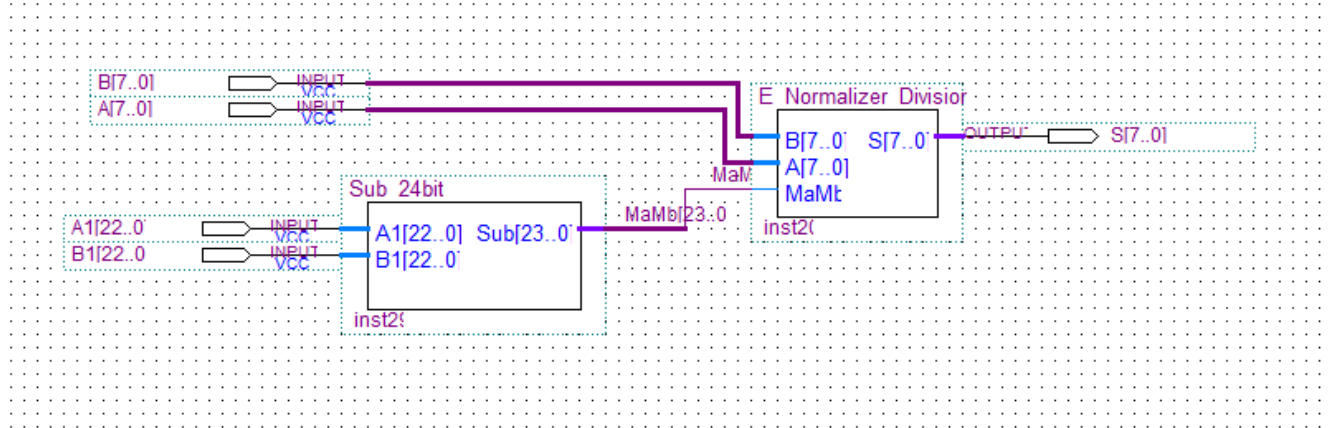
$$Nap = (|Q3 \cdot |Q2 \cdot |Q1 \cdot |Q0)$$

Bảng ngõ ra trạng thái:

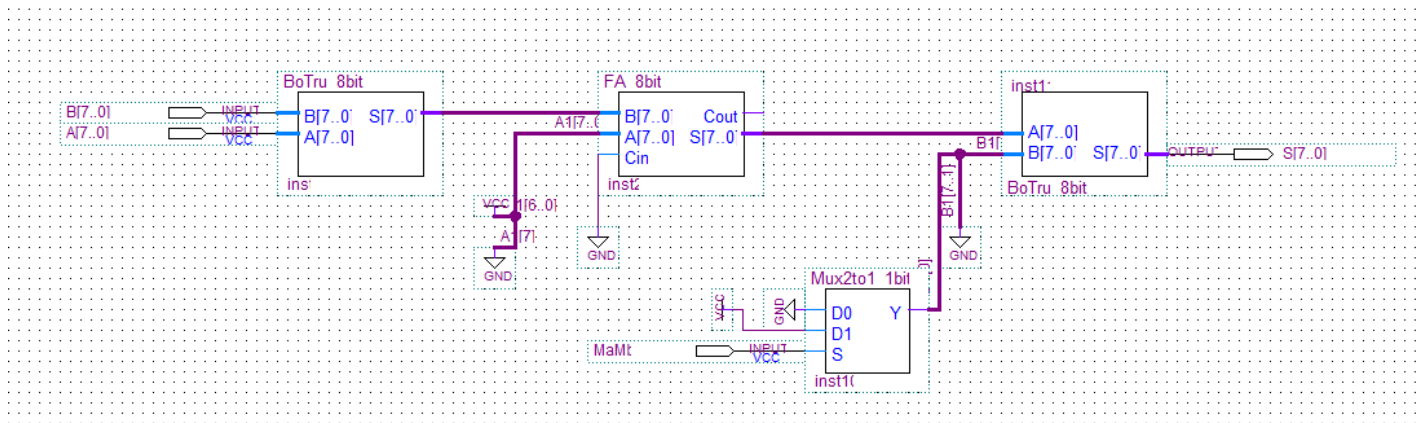
| TTHT | Q₂Q₁Q₀ | Done |
|-------------|--|-------------|
| S0 | 0000 | 0 |
| S1 | 0001 | 0 |
| S2 | 0010 | 0 |
| S3 | 0011 | 0 |
| S4 | 0100 | 0 |
| S5 | 0101 | 0 |
| S6 | 0110 | 0 |
| S7 | 0111 | 0 |
| S8 | 1000 | 0 |
| S9 | 1001 | 0 |
| S10 | 1010 | 0 |
| S11 | 1011 | 1 |
| S12 | 1100 | X |
| S13 | 1101 | X |
| S14 | 1110 | X |
| S15 | 1111 | X |

$$\text{Done} = (Q_3 \cdot Q_1 \cdot Q_0)$$

4. Exponent



5. Normalier



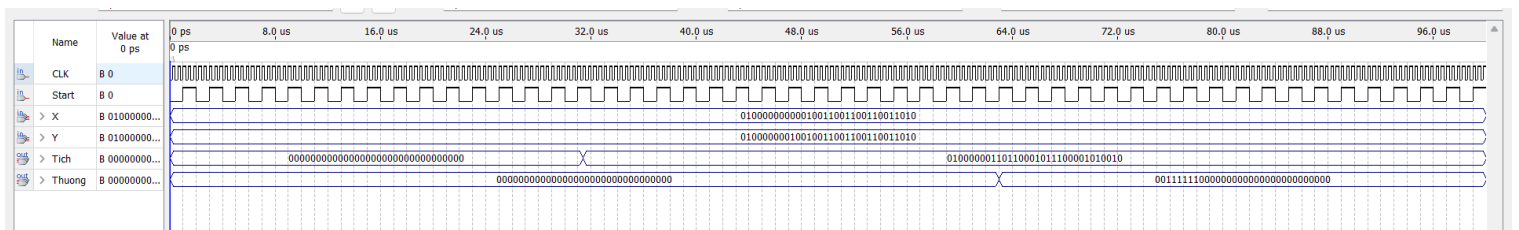
3) Test case

Test case 1 :

| Giá trị | Decimal | Binary | | |
|------------------|---------|--------|----------|--------------------------|
| | | S | E | M |
| X | 2.15 | 0 | 10000000 | 00010011001100110011010 |
| Y | 3.15 | 0 | 10000000 | 10010011001100110011010 |
| Product | 6.7725 | 0 | 10000001 | 10110001011100001010010 |
| Test case | 6.7725 | 0 | 10000001 | 10110001011100001010010 |
| Quotient | 0.6825 | 0 | 01111110 | 01011101011100001010010 |
| Test case | 0.5 | 0 | 01111110 | 000000000000000000000000 |

Sai số nhân = 0

Sai số chia = 0.1825

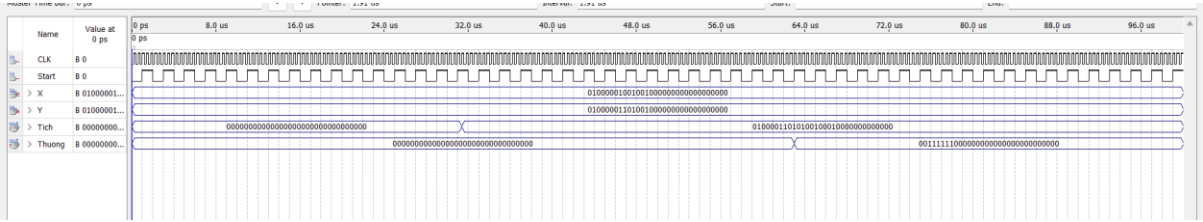


Test case 2:

| Giá trị | Decimal | Binary | | |
|-----------|---------|--------|----------|--------------------------|
| | | S | E | M |
| X | 10.25 | 0 | 10000010 | 010010000000000000000000 |
| Y | 20.5 | 0 | 10000011 | 010010000000000000000000 |
| Product | 210.125 | 0 | 10000110 | 101001000100000000000000 |
| Test case | 210.125 | 0 | 10000110 | 101001000100000000000000 |
| Quotient | 0.5 | 0 | 01111110 | 000000000000000000000000 |
| Test case | 0.5 | 0 | 01111110 | 000000000000000000000000 |

Sai số nhân = 0

Sai số chia = 0

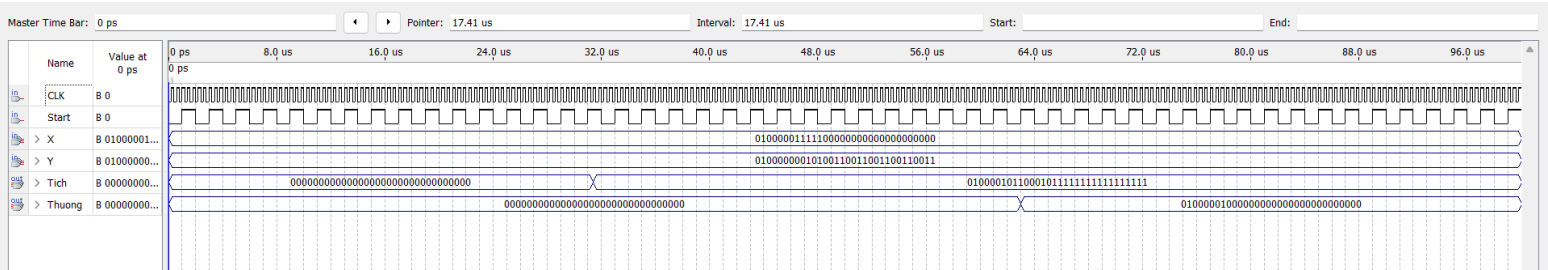


Test case 3:

| Giá trị | Decimal | Binary | | |
|-----------|---------|--------|----------|--------------------------|
| | | S | E | M |
| X | 30 | 0 | 10000011 | 111000000000000000000000 |
| Y | 3.3 | 0 | 10000000 | 10100110011001100110011 |
| Product | 99 | 0 | 10000101 | 100011000000000000000000 |
| Test case | 98.9999 | 0 | 10000101 | 100010111111111111111111 |
| Quotient | 9.0909 | 0 | 10000010 | 00100010111010001010100 |
| Test case | 8 | 0 | 10000010 | 000000000000000000000000 |

Sai số nhân = 10⁻⁴

Sai số chia = 1.0909

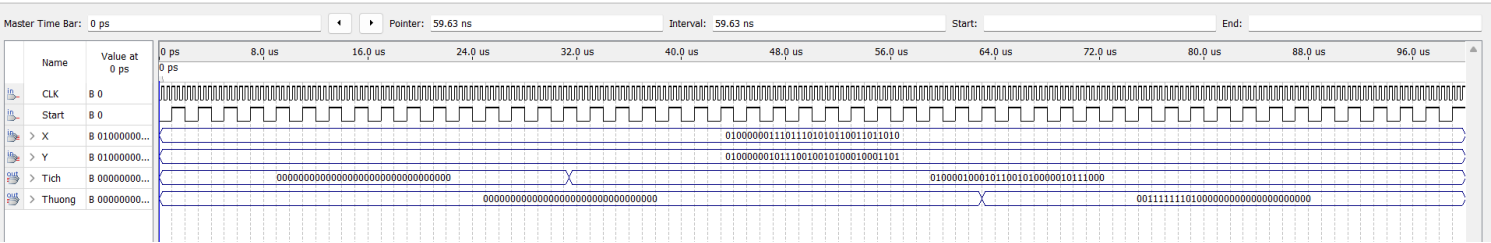


Test case 4:

| Giá trị | Decimal | Binary | | |
|-----------|-----------|--------|----------|--------------------------|
| | | S | E | M |
| X | 7.4586 | 0 | 10000001 | 11011101010110011011010 |
| Y | 5.7862 | 0 | 10000001 | 01110010010100010001101 |
| Product | 43.313178 | 0 | 10000100 | 010110101000000010110010 |
| Test case | 43.156951 | 0 | 10000100 | 010110010100000010111000 |
| Quotient | 1.293698 | 0 | 01111111 | 010010110010111111100101 |
| Test case | 1.25 | 0 | 01111111 | 010000000000000000000000 |

Sai số nhân = 0.156227

Sai số chia = 0.043698

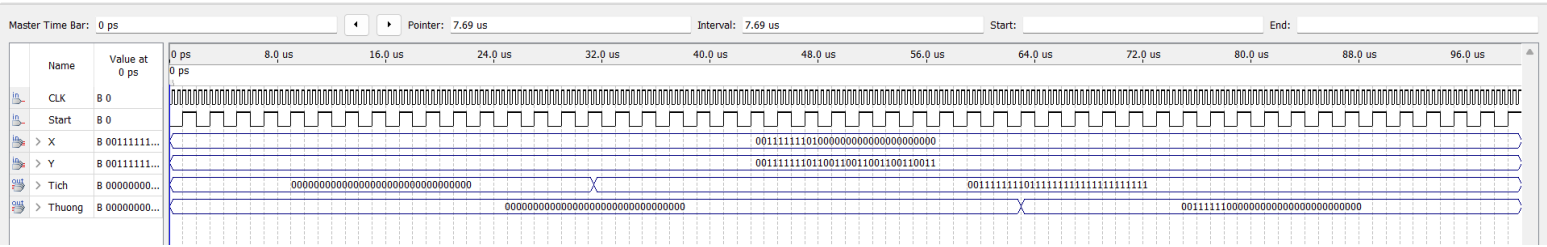


Test case 5:

| Giá trị | Decimal | Binary | | |
|-----------|----------|--------|----------|--------------------------|
| | | S | E | M |
| X | 1.25 | 0 | 01111111 | 010000000000000000000000 |
| Y | 1.4 | 0 | 01111111 | 01100110011001100110011 |
| Product | 1.75 | 0 | 01111111 | 110000000000000000000000 |
| Test case | 1.749999 | 0 | 01111111 | 101111111111111111111111 |
| Quotient | 0.892857 | 0 | 01111110 | 11001001001001001000111 |
| Test case | 0.5 | 0 | 01111110 | 000000000000000000000000 |

Sai số nhân = 10⁻⁶

Sai số chia = 0.392857

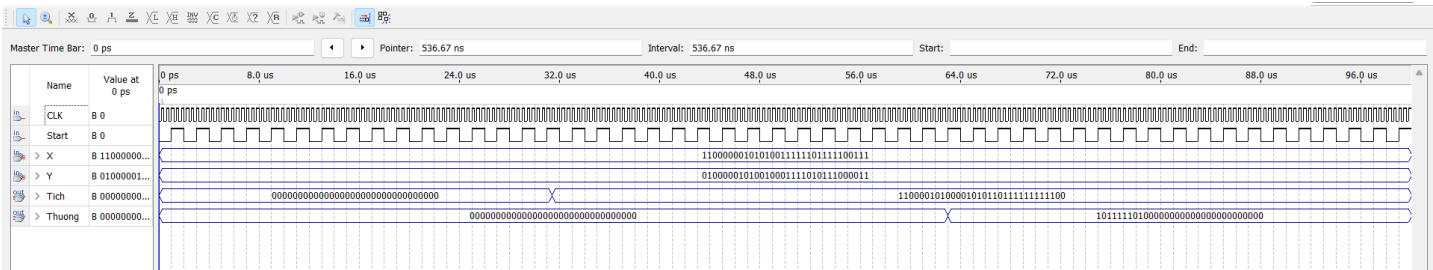


Test case 6:

| Giá trị | Decimal | Binary | | |
|-----------|------------|--------|----------|--------------------------|
| | | S | E | M |
| X | -5.312 | 1 | 10000001 | 01010011111101111100111 |
| Y | 12.56 | 0 | 10000010 | 100100011111010111000011 |
| Product | -66.71872 | 1 | 10000101 | 000010101101111111111100 |
| Test case | -66.718719 | 1 | 10000101 | 000010101101111111111100 |
| Quotient | -0.422929 | 1 | 01111101 | 10110001000101000100110 |
| Test case | -0.25 | 1 | 01111101 | 000000000000000000000000 |

Sai số nhân = -10⁻⁶

Sai số chia = -0.172929

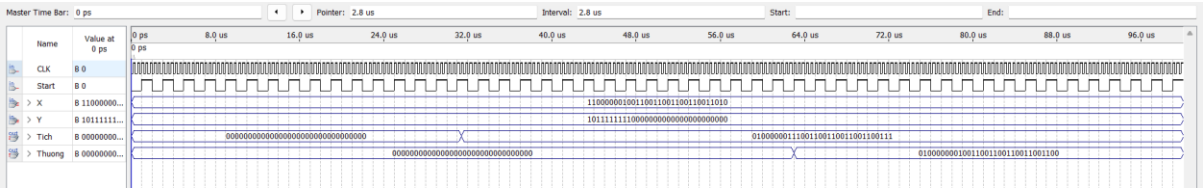


Test case 7:

| Giá trị | Decimal | Binary | | |
|-----------|----------|--------|----------|--------------------------|
| | | S | E | M |
| X | -4.8 | 1 | 10000001 | 00110011001100110011010 |
| Y | -1.5 | 1 | 01111111 | 100000000000000000000000 |
| Product | 7.2 | 0 | 10000001 | 11001100110011001100110 |
| Test case | 7.2 | 0 | 10000001 | 11001100110011001100111 |
| Quotient | 3.2 | 0 | 10000000 | 10011001100110011001101 |
| Test case | 3.199999 | 0 | 10000000 | 10011001100110011001100 |

Sai số nhân = 0

Sai số chia = 10⁻⁶

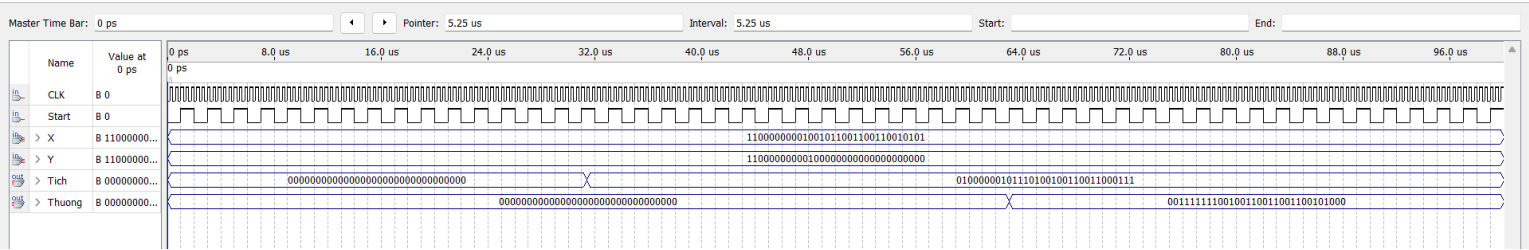


Test case 8:

| Giá trị | Decimal | Binary | | |
|-----------|-----------|--------|----------|--------------------------|
| | | S | E | M |
| X | -2.587499 | 1 | 10000000 | 01001011001100110010101 |
| Y | -2.25 | 1 | 10000000 | 001000000000000000000000 |
| Product | 5.821887 | 0 | 10000001 | 01110100100110011100110 |
| Test case | 5.821872 | 0 | 10000001 | 01110100100110011000111 |
| Quotient | 1.149999 | 0 | 01111111 | 00100110011001100101011 |
| Test case | 1.149998 | 0 | 01111111 | 001001100110011001010100 |

Sai số nhân = 1.5x10⁻⁵

Sai số chia = 10⁻⁶

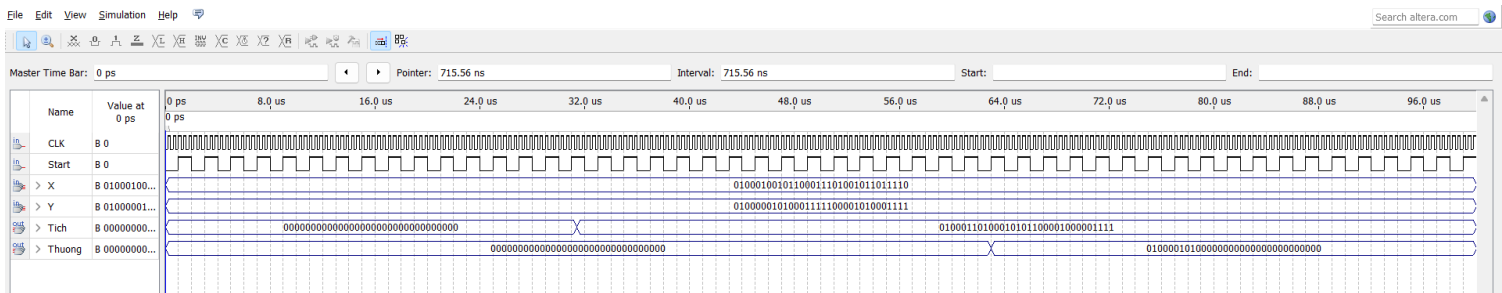


Test case 9:

| Giá trị | Decimal | Binary | | |
|-----------|--------------|--------|----------|-------------------------|
| | | S | E | M |
| X | 1422.58965 | 0 | 10001001 | 01100011101001011011110 |
| Y | 12.485 | 0 | 10000010 | 10001111100001010001111 |
| Product | 17761.03178 | 0 | 10001101 | 00010101100001000010000 |
| Test case | 17761.029296 | 0 | 10001101 | 00010101100001000001111 |
| Quotient | 113.943904 | 0 | 10000101 | 11000111110001101000111 |
| Test case | 64 | 0 | 10000101 | 00000000000000000000000 |

Sai số nhân = 0.02484

Sai số chia = 49.943904

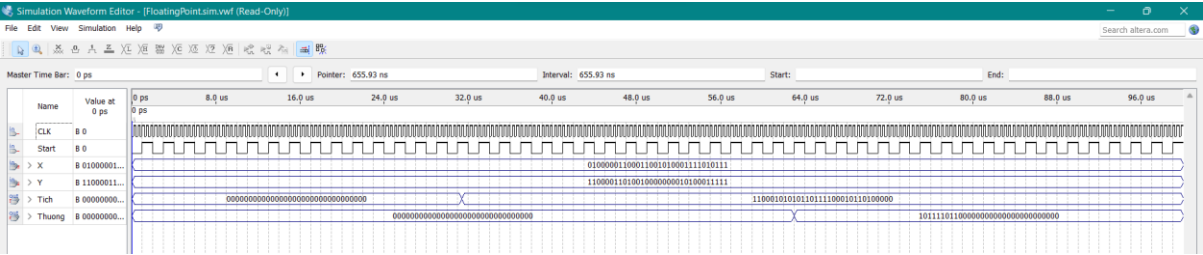


Test case 10:

| Giá trị | Decimal | Binary | | |
|-----------|-------------|--------|----------|--------------------------|
| | | S | E | M |
| X | 17.58 | 0 | 10000011 | 00011001010001111010111 |
| Y | -200.02 | 1 | 10000110 | 10010000000010100011111 |
| Product | -3156.3156 | 1 | 10001010 | 10001010100010100001101 |
| Test case | -3516.35156 | 1 | 10001010 | 101101111100010110100000 |
| Quotient | -0.087891 | 1 | 01111011 | 011010000000000000110010 |
| Test case | -0.0625 | 1 | 01111011 | 000000000000000000000000 |

Sai số nhân = 0

Sai số chia = -0.025391

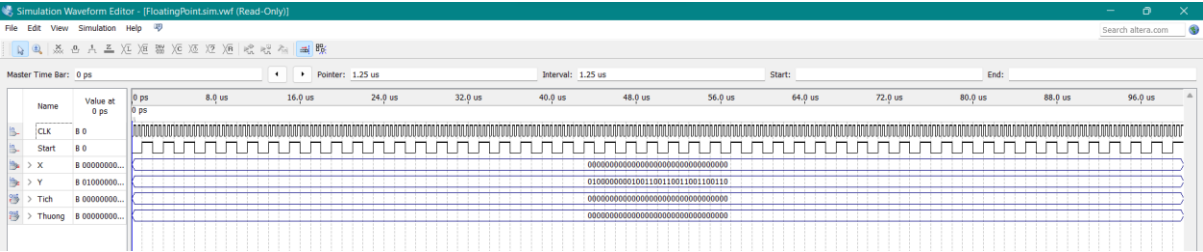


Test case 11:

| Giá trị | Decimal | Binary | | |
|-----------|---------|--------|----------|--------------------------|
| | | S | E | M |
| X | 0 | 0 | 00000000 | 000000000000000000000000 |
| Y | 2.6 | 0 | 10000000 | 01001100110011001100110 |
| Product | 0 | 0 | 00000000 | 000000000000000000000000 |
| Test case | 0 | 0 | 00000000 | 000000000000000000000000 |
| Quotient | 0 | 0 | 00000000 | 000000000000000000000000 |
| Test case | 0 | 0 | 00000000 | 000000000000000000000000 |

Sai số nhân = 0

Sai số chia = 0



Test case 12:

| Giá trị | Decimal | Binary | | |
|-----------|---------|--------|----------|------------------------------|
| | | S | E | M |
| X | 2.6 | 0 | 10000000 | 01001100110011001100110 |
| Y | 0 | 0 | 00000000 | 000000000000000000000000 |
| Product | 0 | 0 | 00000000 | 000000000000000000000000 |
| Test case | 0 | 0 | 00000000 | 000000000000000000000000 |
| Quotient | x | x | xxxxxxxx | xxxxxxxxxxxxxxxxxxxxxxxxxxxx |
| Test case | 0 | 0 | 00000000 | 000000000000000000000000 |

Ta mặc định là trường hợp chia cho 0 thì đầu ra thương sẽ bằng 0 (Ngõ vào Start sẽ không được bật nên không bao giờ chuyển sang trạng thái S1 nếu $Y = 0$).

Sai số nhân = 0

Sai số chia = 0

