**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**

MỤC LỤC

[MỤC LỤC 2](#_Toc131410454)

[DANH MỤC HÌNH ẢNH 3](#_Toc131410455)

[DANH MỤC BẢNG **Lỗi! Thẻ đánh dấu không được xác định.**](#_Toc131410456)

[Intel 8051 **Lỗi! Thẻ đánh dấu không được xác định.**](#_Toc131410457)

[Bài tập 2 **Lỗi! Thẻ đánh dấu không được xác định.**](#_Toc131410458)

# **DANH MỤC HÌNH ẢNH**

# **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 x RE**

*Trong đó :*

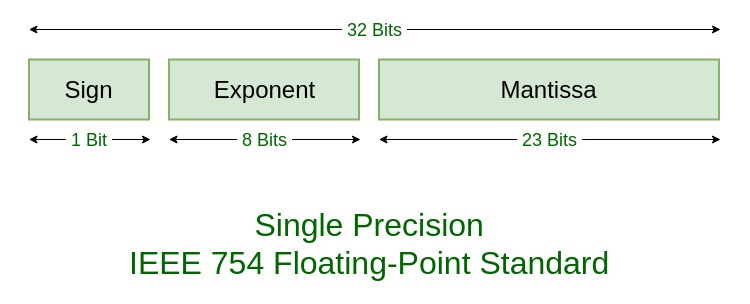
* M : Mantissa
* R : Radix
* E: Exponent

*Ví dụ:*

* 14.2268 (10) = 1.42268 x 101  = 1.42268 E+01
* 0.0154 (10) = 1.54 x 10-2  = 1.54 E-02

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

## **3. Floating-point binary format (Intel)**



*Trong đó:*

* Singed bit (1 bit) : 0 là dương, 1 là âm
* Bias exponent (characteristic): c = E + 2t-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

🡺 c = 6 + 28-1 -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 x 20
3. Normalize: 1100.011 x 20 =1.100011 x 23
4. Exponent: 3 + 28-1 -1 = 3 + 127 = 130 (10) = 1000 0010 (2)
5. Mantissa: 100011🡪 10001100000000000000000

🡺 Tổng hợp 3 phần : 0 10000010 10001100000000000000000

## **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) 🡺 c = 129 - 28-1 -1 = 129 – 127 = 2

3) Mantissa: 1.00001001100110011001101

4) De-normalize: 1.00001001100110011001101 x 22 = 100.001001100110011001101

5) Convert: 100 (2) = 4 (10) ; 001001100110011001101 (2)  0.15 (10)

🡺 Result = 4.15

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

**Sa.Ma.2Ea x Sb.Mb.2Eb = (Sa Sb) . (Ma x Mb). (2(Ea+Eb) – 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 (Mb) với mantissa của a (Ma), 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.

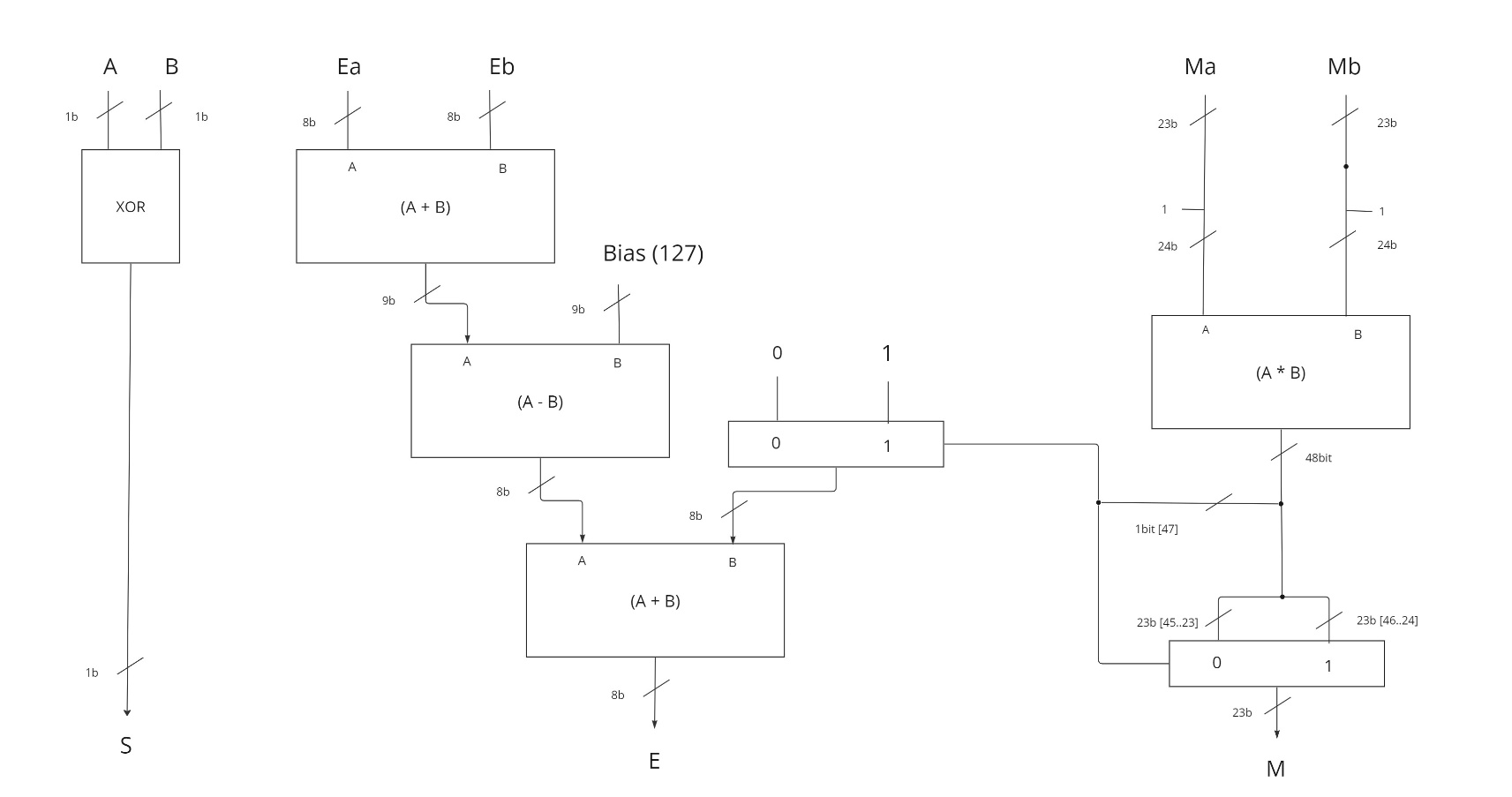
## **7. Chia hai số Floating Point (Floating point Division)**

**= (Sa Sb) . (). (2(Ea-Eb) + 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 (Ma) cho mantissa của b (Mb) 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**



Ảnh có chứa biểu đồ, hàng, văn bản, Song song

Mô tả được tạo tự động***Mạch nhân floating point:***

***Ảnh có chứa hàng, biểu đồ, văn bản, Sơ đồ

Mô tả được tạo tự độngMantissa:***

1. ***Bảng ALU***

|  |  |
| --- | --- |
| **Opcode (S1S0)** | **Chức năng** |
| 00 | ADD |
| 01 | SUB |
| 10 | A + 1 |
| 11 | AND |

1. ***Bảng Shifter***

|  |  |
| --- | --- |
| **S1S0** | **Chức năng** |
| 00 | Không dịch |
| 01 | Dịch trái 1 bit |
| 10 | Dịch phải luận lý 1 bit |
| 11 | Output = 0 |

1. ***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 · (|Q1 + |Q0) ) + (|Q2 · Q1 · Q0)

D1 = ((Q1 · |Q0) · (|Q2 + Zero)) + (|Q1 · Q0)

D0 = ( (|Q1 · |Q0) · (Start + Q2) ) + ( (Q1 · |Q0) · (|Q2 + Zero) )

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TTHT** | **Q2Q1Q0** | **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 | Multipler | 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 |

**Bảng ngõ ra điều khiển**

**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

AorB = Q1’

Nap = Q2’Q1’Q0

**Bảng ngõ ra trạng thái:**

|  |  |  |
| --- | --- | --- |
| **TTHT** | **Q2Q1Q0** | **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 = Q2 · Q1 · Q0

1. ***Ảnh có chứa văn bản, biểu đồ, hàng, số

   Mô tả được tạo tự độngNormalizer***

## **Floating point Division**

Ảnh có chứa biểu đồ, bản phác thảo, Kế hoạch, Bản vẽ kỹ thuật

Mô tả được tạo tự động

***Ảnh có chứa biểu đồ, văn bản, Kế hoạch, hàng

Mô tả được tạo tự độngMạch chia floating point***

***Ảnh có chứa văn bản, biểu đồ, Kế hoạch, hàng

Mô tả được tạo tự độngMantissa:***

Ảnh có chứa ảnh chụp màn hình, hàng, văn bản, Phông chữ

Mô tả được tạo tự động***Datapath:***

1. ***Bảng ALU***

|  |  |
| --- | --- |
| **Opcode (S1S0)** | **Chức năng** |
| 00 | ADD |
| 01 | SUB |
| 10 | A + 1 |
| 11 | AND |

1. ***Bảng Shifter***

|  |  |
| --- | --- |
| **S1S0** | **Chức năng** |
| 00 | Không dịch |
| 01 | Dịch trái 1 bit |
| 10 | Dịch phải luận lý 1 bit |
| 11 | Output = 0 |

1. ***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 · |Q2) · ( (Is25 · |Q0) + |Q1) ) + ( |Q3 · Q2 · Q1 · Q0 )

D2 = ((|Q3 · Q2) · (|Q1 + |Q0)) + (|Is25 · Q3 · |Q2 · Q1 · |Q0) + (|Q3 · |Q2 · Q1 · Q0)

D1 = ( |Q3 · ( (|Q1 · Q0) + (Q1 · |Q0) ) ) + ( |Q2 · ( (|Q1 · Q0) + (Is25 · Q1 · |Q0) ) )

D0 = ( (|Q2 · |Q0) · (Q1 + Q3 + Start) ) + ( (|Q3 · |Q0) · ( Q1 + (Q2 · |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 · |Q2) · ( (Q1 · |Q0) + (|Q1 · Q0) )

WE = (|Q3 · Q1) + (Q3 · |Q0) + (|Q1 · Q0) + Q2

W\_addr[1] = (Q1 · (Q0 + Q3)) + (Q2 · |Q0)

W\_addr[0] = ( |Q1 · ( (Q3 · Q0) + (|Q3 · |Q0) ) ) + (|Q2 · Q1 · |Q0)

RA[1] = Q1 · (Q2 + Q3)

RA[0] = (|Q2 · Q1 · |Q0) + (Q3 · |Q1 · Q0)

RB[1] = Q1 · (|Q0 + Q3)

RB[0] = |Q1 · (Q3 + Q0)

SignEn = |Q3 . Q2 . |Q1. Q0

Plus Signal = (Q3 · |Q1 · |Q0)

Divide Signal = (Q2 · Q1 · Q0)

Opcode[1] = (Q1 · |Q0) + (Q3 · Q0)

Opcode[0] = (|Q3 · (|Q0 + |Q1)) + (|Q2 · Q0)

Shifter[1] = (Q3 · |Q1 · Q0)

Shifter[0] = (|Q3 · Q1 · |Q0)

OE = (Q3 · Q1 · Q0)

Nap = (|Q3 · |Q2 · |Q1 · |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 |

Done = (Q3 · Q1 · Q0)

1. Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, hàng

   Mô tả được tạo tự động***Exponent***
2. Ảnh có chứa văn bản, biểu đồ, hàng, ảnh chụp màn hình

   Mô tả được tạo tự động***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 | 00000000000000000000000 |

Sai số nhân = 0

Sai số chia = 0.1825

Ảnh có chứa văn bản, ảnh chụp màn hình, hàng, số

Mô tả được tạo tự động

***Test case 2:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Giá trị | Decimal | Binary | | |
| **S** | **E** | **M** |
| X | 10.25 | 0 | 10000010 | 01001000000000000000000 |
| Y | 20.5 | 0 | 10000011 | 01001000000000000000000 |
| Product | 210.125 | 0 | 10000110 | 10100100010000000000000 |
| Test case | 210.125 | 0 | 10000110 | 10100100010000000000000 |
| Quotient | 0.5 | 0 | 01111110 | 00000000000000000000000 |
| Test case | 0.5 | 0 | 01111110 | 00000000000000000000000 |

Sai số nhân = 0

Sai số chia = 0

Ảnh có chứa văn bản, hàng, số, Phông chữ

Mô tả được tạo tự động

***Test case 3:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Giá trị | Decimal | Binary | | |
| **S** | **E** | **M** |
| X | 30 | 0 | 10000011 | 11100000000000000000000 |
| Y | 3.3 | 0 | 10000000 | 10100110011001100110011 |
| Product | 99 | 0 | 10000101 | 10001100000000000000000 |
| Test case | 98.9999 | 0 | 10000101 | 10001011111111111111111 |
| Quotient | 9.0909 | 0 | 10000010 | 00100010111010001010100 |
| Test case | 8 | 0 | 10000010 | 00000000000000000000000 |

Sai số nhân = 10-4

Sai số chia = 1.0909

Ảnh có chứa văn bản, hàng, số, Phông chữ

Mô tả được tạo tự động

***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 | 01011010100000010110010 |
| Test case | 43.156951 | 0 | 10000100 | 01011001010000010111000 |
| Quotient | 1.293698 | 0 | 01111111 | 01001011001011111100101 |
| Test case | 1.25 | 0 | 01111111 | 01000000000000000000000 |

Sai số nhân = 0.156227

Sai số chia = 0.043698

Ảnh có chứa văn bản, hàng, số, Phông chữ

Mô tả được tạo tự động

***Test case 5:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Giá trị | Decimal | Binary | | |
| **S** | **E** | **M** |
| X | 1.25 | 0 | 01111111 | 01000000000000000000000 |
| Y | 1.4 | 0 | 01111111 | 01100110011001100110011 |
| Product | 1.75 | 0 | 01111111 | 11000000000000000000000 |
| Test case | 1.749999 | 0 | 01111111 | 10111111111111111111111 |
| Quotient | 0.892857 | 0 | 01111110 | 11001001001001001000111 |
| Test case | 0.5 | 0 | 01111110 | 00000000000000000000000 |

Sai số nhân = 10-6

Ảnh có chứa văn bản, hàng, số, Phông chữ

Mô tả được tạo tự độngSai 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 | 10010001111010111000011 |
| Product | -66.71872 | 1 | 10000101 | 00001010110111111111100 |
| Test case | −66.718719 | 1 | 10000101 | 00001010110111111111100 |
| Quotient | -0.422929 | 1 | 01111101 | 10110001000101000100110 |
| Test case | −0.25 | 1 | 01111101 | 00000000000000000000000 |

Sai số nhân = -10-6

Ảnh có chứa văn bản, hàng, số, Phông chữ

Mô tả được tạo tự độngSai 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 | 10000000000000000000000 |
| 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

Ảnh có chứa văn bản, ảnh chụp màn hình, hàng, số

Mô tả được tạo tự độngSai 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 | 00100000000000000000000 |
| 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 | 00100110011001100101000 |

Sai số nhân = 1.5x10-5

Sai số chia = 10-6

Ảnh có chứa văn bản, hàng, ảnh chụp màn hình, số

Mô tả được tạo tự động***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

Ảnh có chứa văn bản, ảnh chụp màn hình, hàng, số

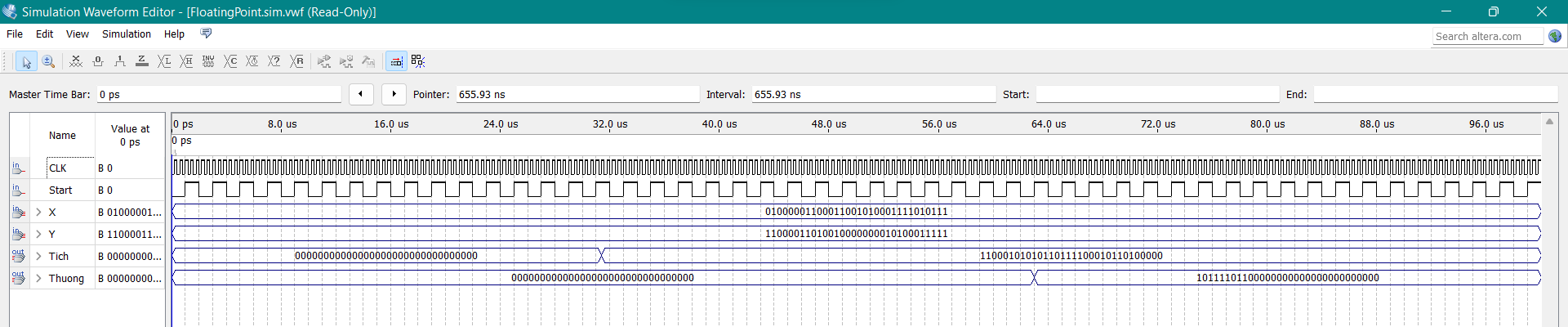
Mô tả được tạo tự độngSai số chia = 49.943904

***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 | 10110111100010110100000 |
| Quotient | -0.087891 | 1 | 01111011 | 01101000000000000110010 |
| Test case | −0.0625 | 1 | 01111011 | 00000000000000000000000 |

Sai số nhân = 0

Sai số chia = -0.025391

***Test case 11:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Giá trị | Decimal | Binary | | |
| **S** | **E** | **M** |
| X | 0 | 0 | 00000000 | 00000000000000000000000 |
| Y | 2.6 | 0 | 10000000 | 01001100110011001100110 |
| Product | 0 | 0 | 00000000 | 00000000000000000000000 |
| Test case | 0 | 0 | 00000000 | 00000000000000000000000 |
| Quotient | 0 | 0 | 00000000 | 00000000000000000000000 |
| Test case | 0 | 0 | 00000000 | 00000000000000000000000 |

Sai số nhân = 0

Sai số chia = 0

Ảnh có chứa văn bản, hàng, số, Phông chữ

Mô tả được tạo tự động***Test case 12:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Giá trị | Decimal | Binary | | |
| **S** | **E** | **M** |
| X | 2.6 | 0 | 10000000 | 01001100110011001100110 |
| Y | 0 | 0 | 00000000 | 00000000000000000000000 |
| Product | 0 | 0 | 00000000 | 00000000000000000000000 |
| Test case | 0 | 0 | 00000000 | 00000000000000000000000 |
| Quotient | x | x | xxxxxxxx | xxxxxxxxxxxxxxxxxxxxxxx |
| Test case | 0 | 0 | 00000000 | 00000000000000000000000 |

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

Ảnh có chứa văn bản, hàng, số, Phông chữ

Mô tả được tạo tự động