D0-D16: Tim cac dau can tuong ung voi DAU RA M

D20-D36: Giá trị 16 đầu cân

D60 : Lưu tổng hiện tại

D62: Lưu tổng cài đặt

D80-D96:Chỉ số đầu cân sau khi lọc

D100-D116: Giá trị đầu cân sau khi lọc

D40-D56: Mảng lưu các đầu cân đã cộng (Arr)

Z0: Gia tri thay doi cua Arr(n) va Arr(n-1)

Z3: Đã cộng bao nhiêu giá trị (n-1)

Z6, Z7: Có bao nhiêu đầu cân có giá trị khác 0

Z5, Z4: Tim ra cac dau can ung voi M

Z1: Biến chạy các giá trị đầu cân (bien n)

Z2: Biến vị trí của các đầu cân (bien k)

Bước 1: Tạo 1 bảng có 2 cột liệt kê các đầu cân có giá trị #0 (So sánh D20-D36 với K0)

|  |  |
| --- | --- |
| D80 | D100 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

SAI

ĐÚNG

SAI

IF(Z1 >= Z2)

SAI

ĐÚNG

ĐÚNG

IF(Z1 >= Z2)

ĐÚNG

ĐÚNG

IF(D41 >= Z6)

IF(D60 > D62)

SAI

THOÁT

Gán D60 = D100

Gán Z1 = 0

Gán Z2 = 0

While (D62 # D60)

Nôm na là thế này. Tôi giả sử dãy số của bạn gồm các số từ a đến z. Và tổng cho trước là Tổng

So a với tổng

Nếu a > tổng thì so b với tổng...

Nếu a < tổng thì so a + b với tổng

Nếu a + b > tổng thì so a + c với tổng...

Nếu a + b < tổng thì so a + b + c với tổng

...

Nếu a + ... + o + z <> tổng thì so a + ... + p với tổng

...

Cho đến khi tìm được tập số thỏa điều kiện.

//////////////////////

///////////////////

///////////////////

Function SumF(Num As Double, ParamArray Args() As Variant) As String

Dim Data(), Cll As Range, n As Long, k As Long, Arr(), Total As Double

For i = LBound(Args) To UBound(Args)

For Each Cll In Args(i)

If Val(Cll.Value) <> 0 Then

n = n + 1

ReDim Preserve Data(1 To 2, 1 To n)

Data(1, n) = Val(Cll.Value)

Data(2, n) = Cll.Address(0, 0)

End If

Next

Next

ReDim Arr(1 To 1)

Arr(1) = 1

Total = Data(1, 1)

n = 1

k = 1

Do While Num <> Total

If Arr(1) = UBound(Data, 2) Then

SumF = "#N/A"

Exit Function

End If

If Total > Num Then

If k = UBound(Data, 2) Then

k = Arr(n - 1) + 1

Total = Total - Data(1, Arr(n)) - Data(1, Arr(n - 1)) + Data(1, k)

n = n - 1

Arr(n) = k

Else

k = k + 1

Total = Total - Data(1, Arr(n)) + Data(1, k)

Arr(n) = k

End If

Else

If k = UBound(Data, 2) Then

k = Arr(n - 1) + 1

Total = Total - Data(1, Arr(n)) - Data(1, Arr(n - 1)) + Data(1, k)

n = n - 1

Arr(n) = k

Else

k = k + 1

Total = Total + Data(1, k)

n = n + 1

ReDim Preserve Arr(1 To n)

Arr(n) = k

End If

End If

Loop

For k = 1 To n

SumF = SumF & "+" & Data(2, Arr(k))

Next

SumF = Replace(SumF, "+", "=", 1, 1)

End Function

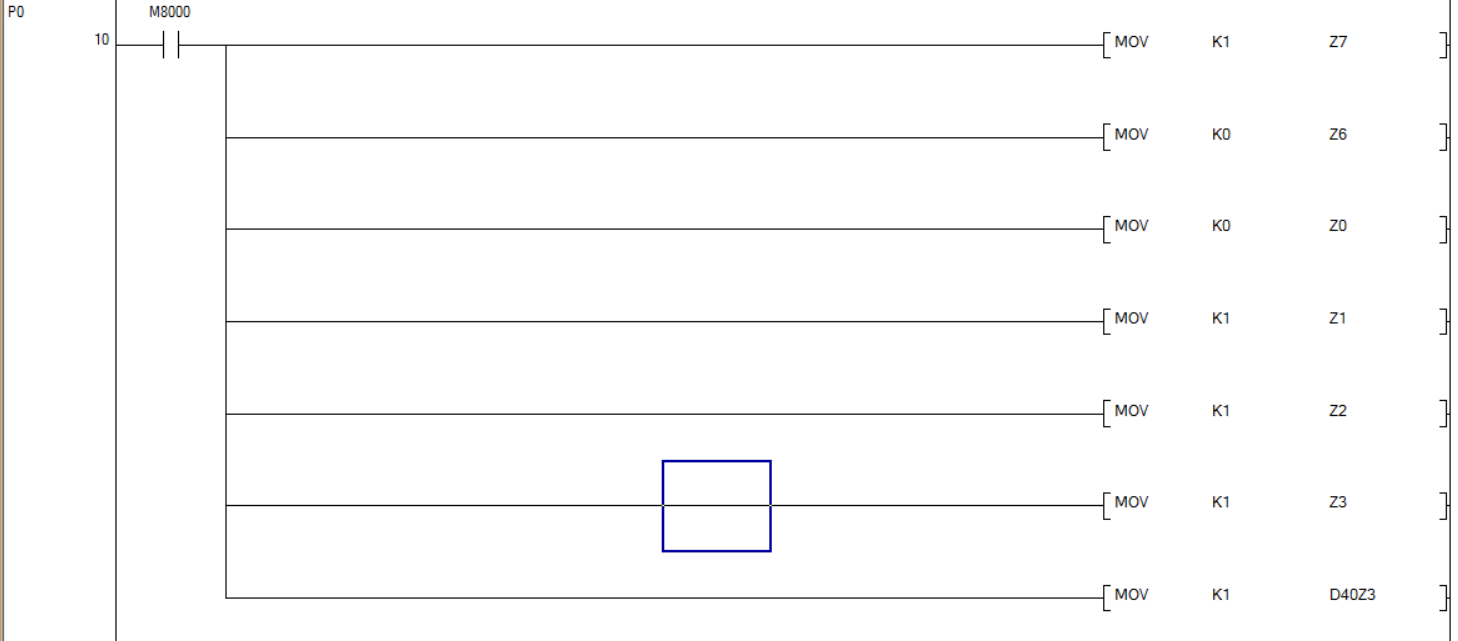
//////////////////

////////////////



Chương trình CHÍNH

Nhảy đến nhãn P0



Khởi tạo giá trị ban đầu cho biến k=1(Z2), n=1(Z1), Arr(1) (D40Z3)= 1, Z6(có bao nhiêu số khác 0), Z7 biến quét 16 giá trị, Z3(n-1).

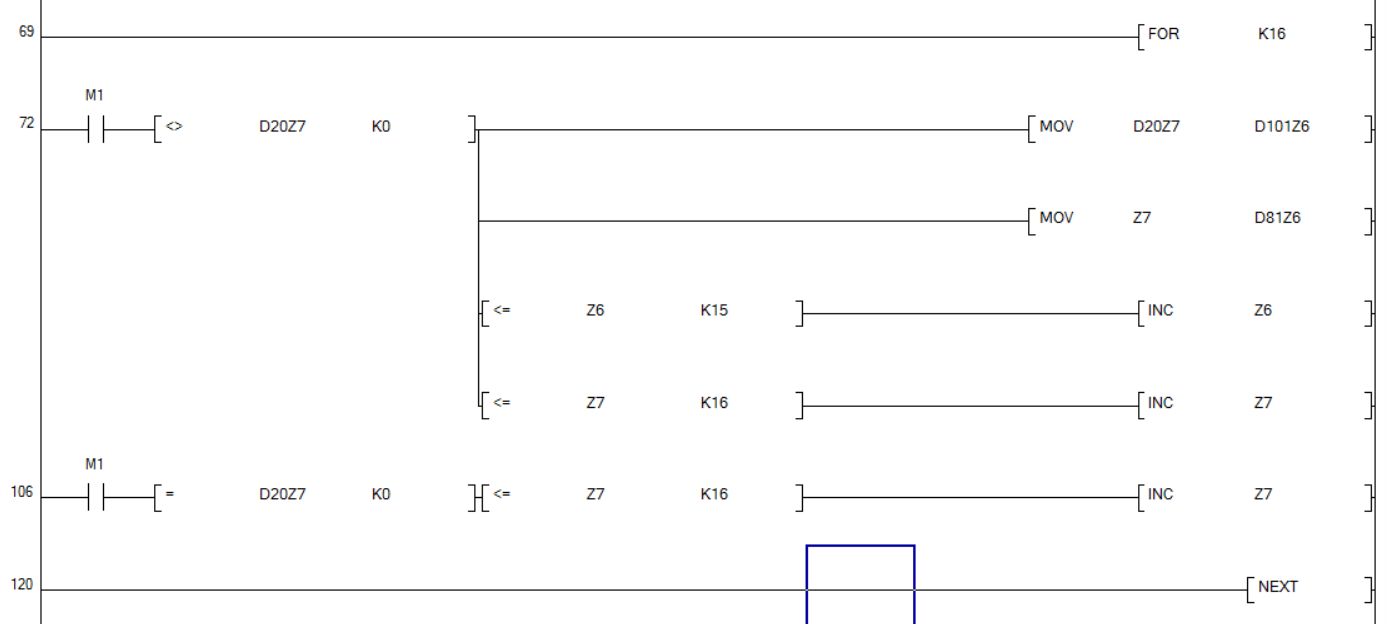


Z5 quét các giá trị có trong Arr và xuất ra M31-M46

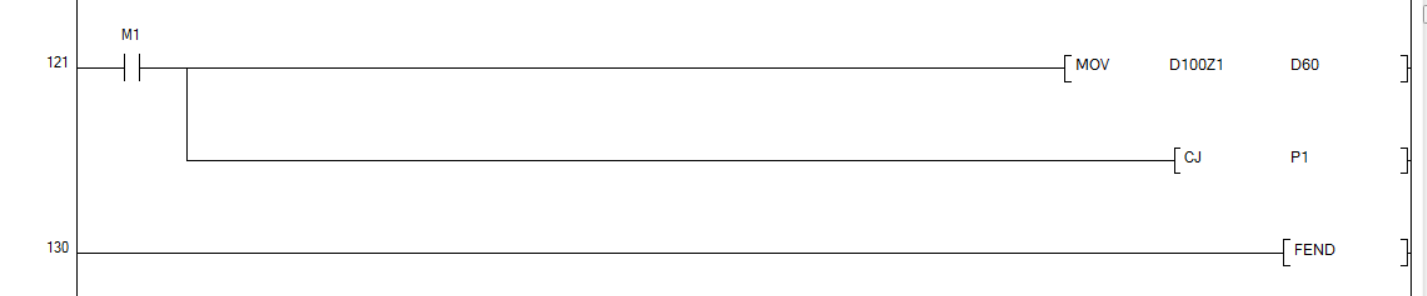
Z4 biến chạy để xuất M31-M46

Xóa tất cả các biến M30-M46

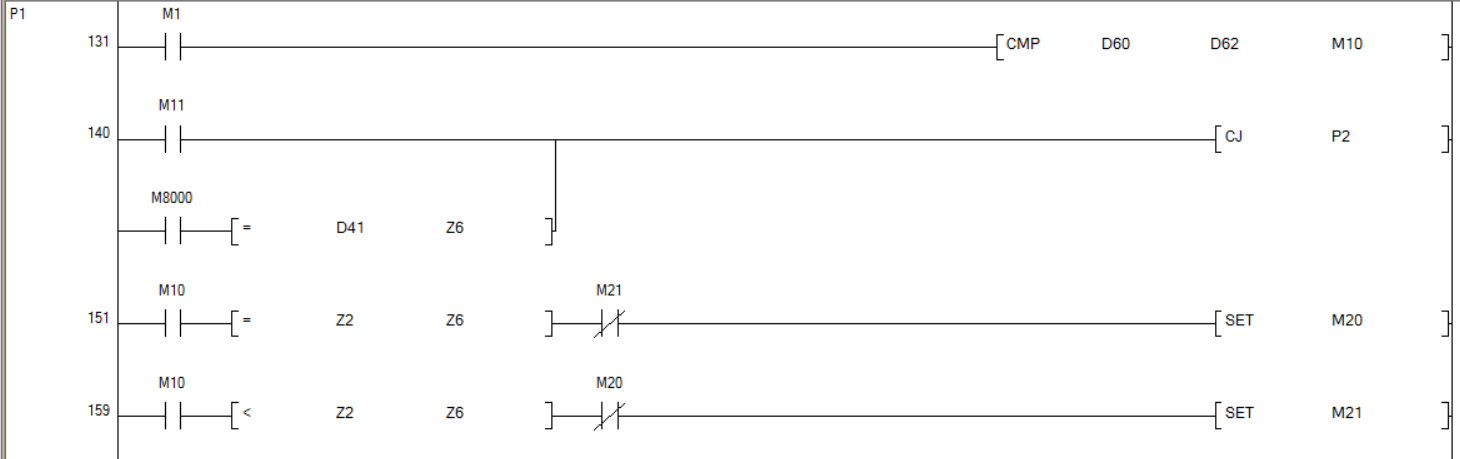
Xóa giá trị D2



Quét tất cả 16 đầu cân và tìm ra các giá trị # 0 Lưu vào bảng bắt đầu từ D101 và vị trí thực tế của các đầu cân vào bảng bắt đầu từ D81



Gán giá trị đầu cân đầu tiên vào biến Tính Tổng D60 và nhảy đến nhãn P1



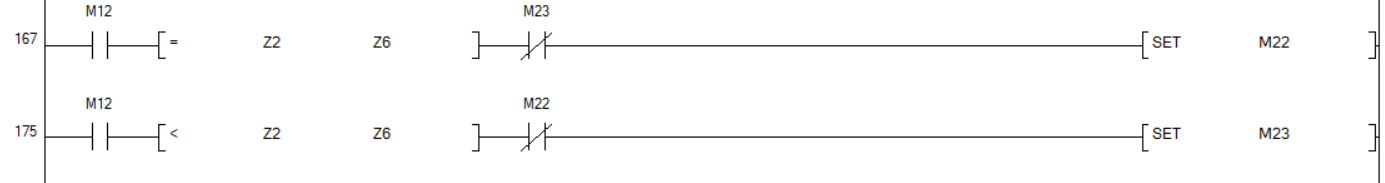
So sánh D60(Giá trị Tổng hiện tại) và D62(Giá trị Tổng cài đặt)

Nếu D60 > D62 thì M10 ON; D60 = D62 thì M11 ON; D60 < D62 thì M12 ON

Vị trí đầu tiên trong bảng VỊ TRÍ THỰC TẾ của đầu cân có giá trị #0 bằng VỊ TRÍ ĐẦU CÂN CUỐI CÙNG thì nhảy đến nhãn P2. Hoặc M11 ON (tức là D60 = D62) thì nhảy đến nhãn P2.

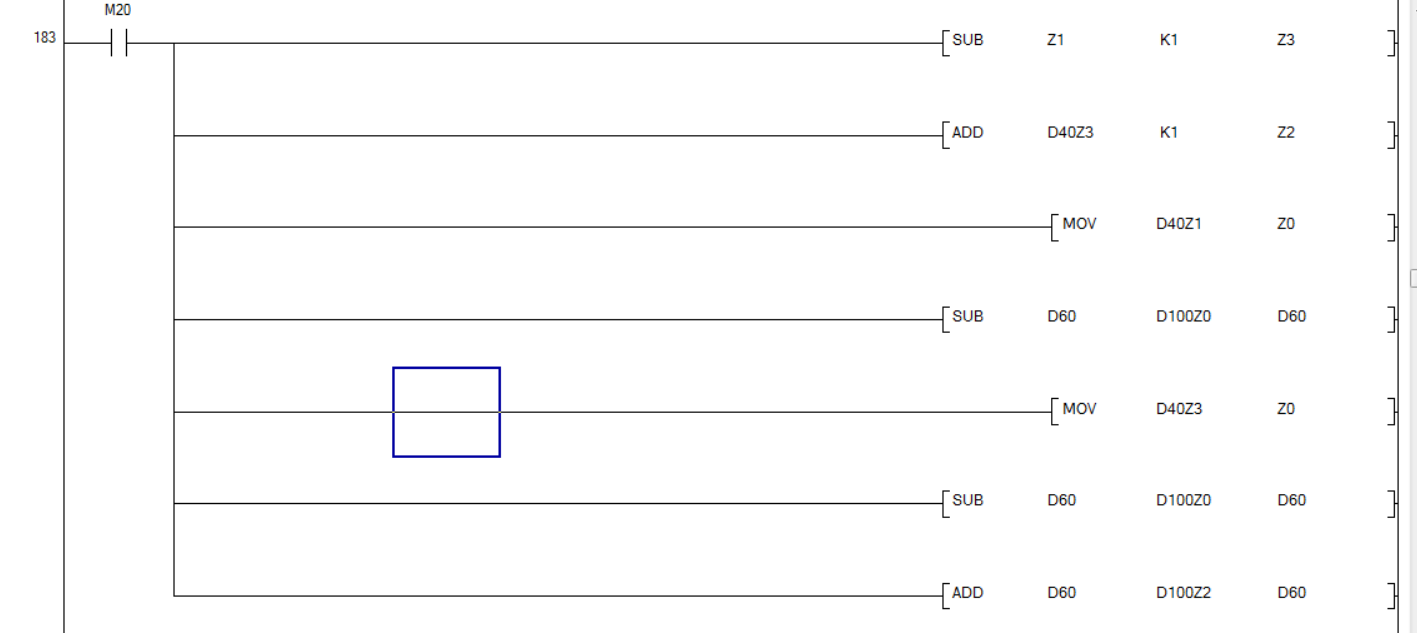
Nến M10 ON (tức là D60 > D62) và biến Z2(biến n) = Z6(Có bao nhiêu đầu cân #0) SET M20.

Nến M10 ON (tức là D60 > D62) và biến Z2(biến n) < Z6(Có bao nhiêu đầu cân #0) SET M21



Nến M12 ON (tức là D60 < D62) và biến Z2(biến n) = Z6(Có bao nhiêu đầu cân #0) SET M22.

Nến M12 ON (tức là D60 < D62) và biến Z2(biến n) < Z6(Có bao nhiêu đầu cân #0) SET M23.



Khi SET M20. Tính giá trị Z3(n-1) = Z1(n) -1

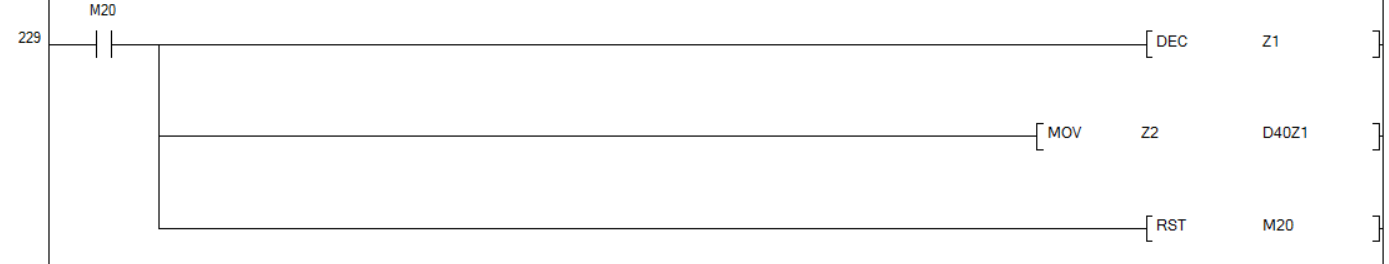
Gán Z2 (biến k) = Arr(n-1) (tức D40Z3) + 1

Z0 là biến chạy để lấy Khối lượng trong BẢNG từ D101đến D100Z6

Tìm giá trị Arr(n) = D40Z1

Tìm giá trị Arr(n-1) = D40Z3

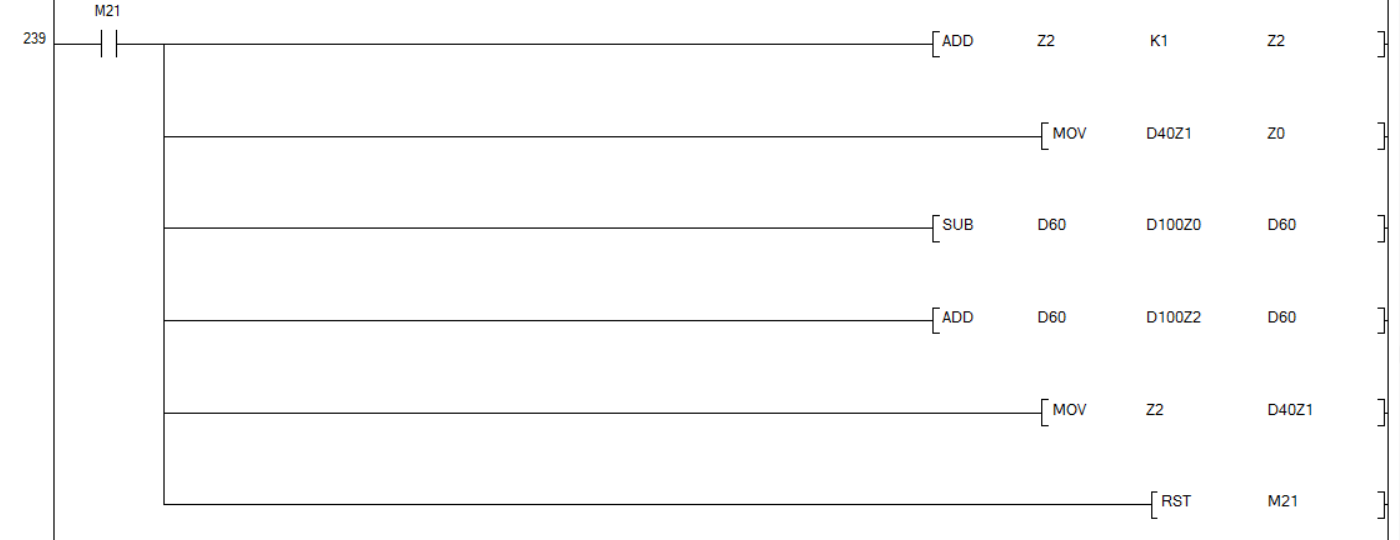
D60 = D60 – Khối lượng ở vị trí Arr(n) - Khối lượng ở vị trí Arr(n-1) + Khối lượng ở vị trí k 🡺 D60 = D60 – D100Z0(Khi Z0 = D40Z1) - D100Z0(Khi Z0 = D40Z3) + D100Z2



Giảm Z1 (n) xuống 1 đơn vị

Gán lại giá trị Arr(n) = Z2 (k)

RST M20.



Khi SET M21.

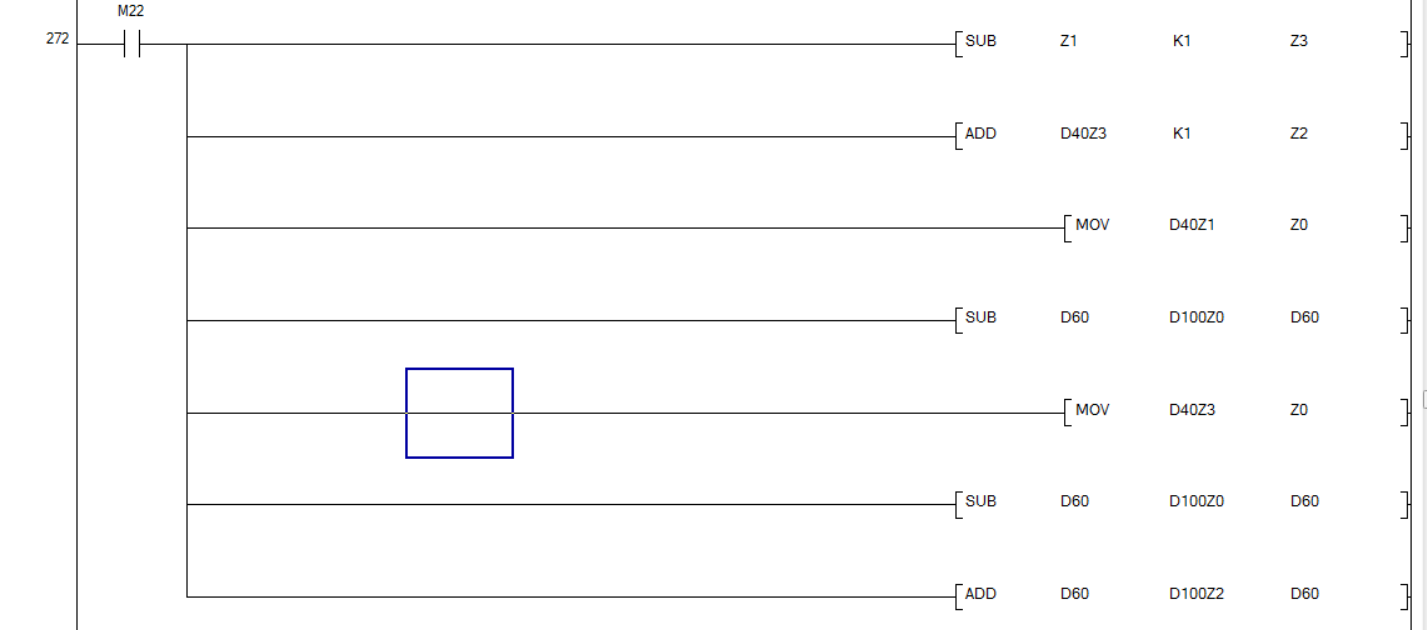
Tăng Z2(k) lên 1 đơn vị

D60 = D60 - Khối lượng ở vị trí Arr(n) + Khối lượng ở vị trí k

* D60 = D60 - D100Z0(Khi Z0 = D40Z1) + D100Z2

Gán lại Arr(n) (tức D40Z1) = Z2 (k)

RST M21.



Khi SET M22. Tính giá trị Z3( tức n-1) = Z1(n) -1

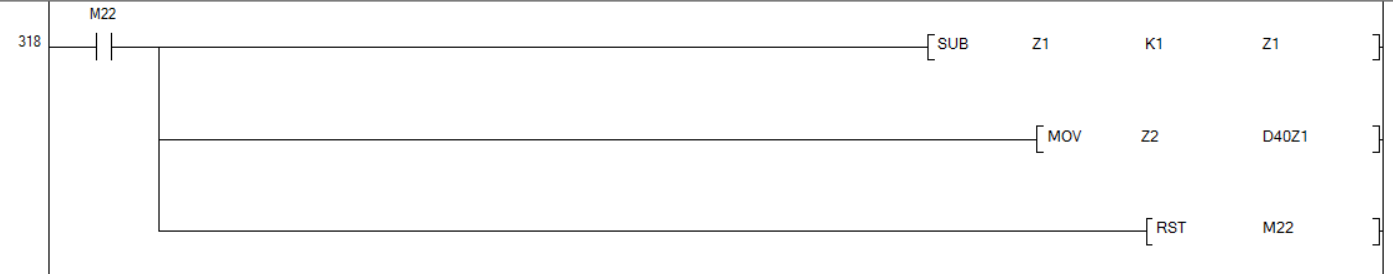
Gán Z2 (biến k) = Arr(n-1) (tức D40Z3) + 1

Z0 là biến chạy để lấy Khối lượng trong BẢNG từ D101đến D100Z6

Tìm giá trị Arr(n) = D40Z1

Tìm giá trị Arr(n-1) = D40Z3

D60 = D60 – Khối lượng ở vị trí Arr(n) - Khối lượng ở vị trí Arr(n-1) + Khối lượng ở vị trí k 🡺 D60 = D60 – D100Z0(Khi Z0 = D40Z1) - D100Z0(Khi Z0 = D40Z3) + D100Z2



Giảm Z1 (n) xuống 1 đơn vị

Gán lại Arr(n) (tức D40Z1) = Z2 (k)



Tăng Z2 (k) lên 1 đơn vị

D60 = D60 + Khối lượng ở vị trí k

🡺 D60 = D60 + D100Z2

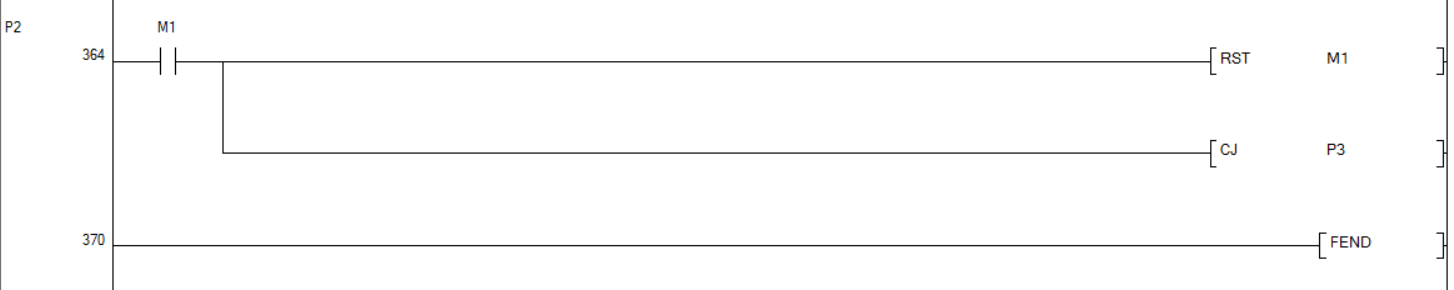
Tăng Z1 (n) lên 1 đơn vị

Gán Arr(n) (tức D40Z1) = Z2 (k)

RST M23.



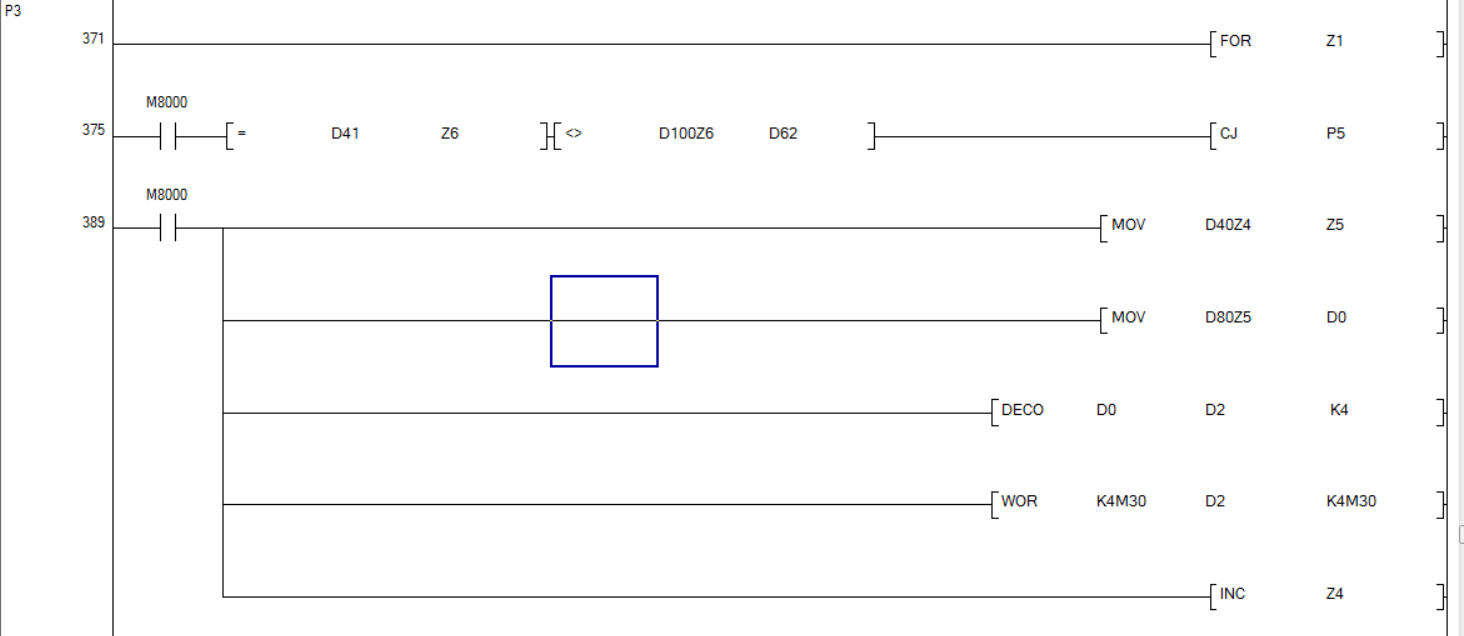
Khi D60 # D62 thì LẶP LẠI NHÃN P1



Khi D60 = D60 thì nhảy đến NHÃN P2

RST M1

Nhảy đến NHÃN P3.

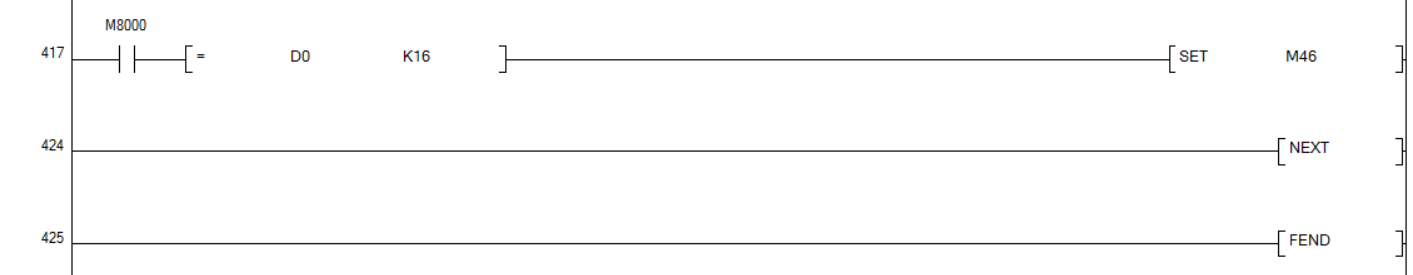


Chạy hàm FOR Z1 (n) lần

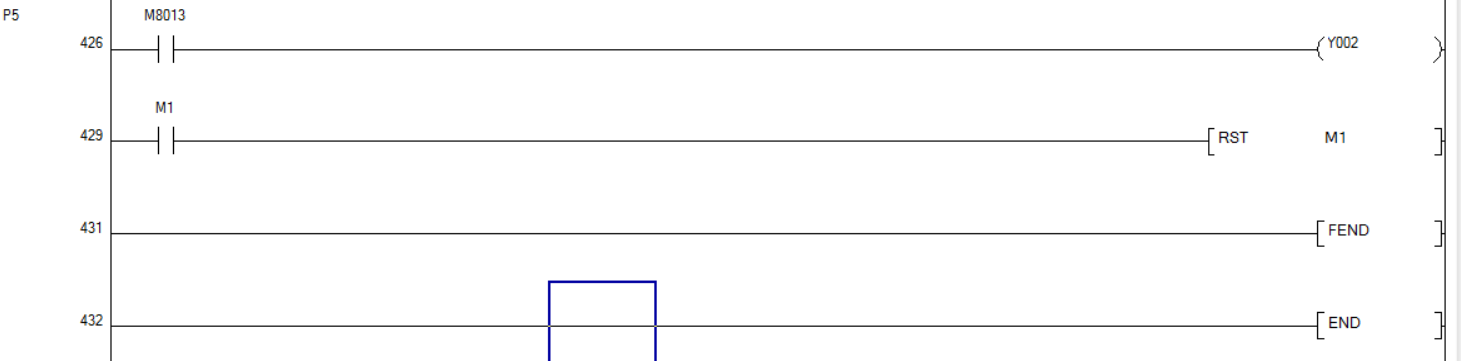
Khi D41 = Z6: Giá trị đầu tiên Arr(1) tức D41 = Số đầu cân có giá trị #0 và Giá trị #0 cuối cùng phải KHÁC D60 (Thuộc trường hợp không có số nào phù hợp) thì nhảy đến NHÃN P5 để THOÁT

Z5 biến chạy để lấy vị trí khối lượng đầu cân đã cộng (D80Z5)

Lấy giá trị D80Z5 giải mã để xuất đầu ra tương ứng, D2 là vùng nhớ trung gia để xử lý



Do hàm WOR chỉ xử lý được từ M30>> M45 nên khi D0 = 16 thì ta SET M46 (Trường hợp đầu cân ở vị trí thứ 16 được cộng vào D60).



Khi đã quét tất cả các đầu cân mà không có giá trị phù hợp thì nhảy đến NHÃN P5 để THOÁT và quay lại VÒNG QUÉT tiếp theo. Kết thúc 1 lần xử lý.