

# Samkoon PLC





## Instrukcje programowania dla użytkownika PLC firmy Samkoon

### O niniejszej instrukcji

Niniejsza instrukcja obsługi opisuje funkcje programowania, które dostępne są w sterownikach PLC firmy Samkoon wszystkich serii. Dedykowana jest zarówno dla zaawansowanych jak i początkujących użytkowników. Serdecznie zapraszamy do zapoznania się z ofertą PLC firmy Samkoon, która reprezentuje jedno z największych osiągnięć firmy w dziedzinie swobodnie programowalnych sterowników. Wszystkie produkty Samkoon stanowią modułowy system automatyki o kompaktowych rozmiarach, idealny zarówno do prostych, jak i bardziej zaawansowanych zastosowań związanych z algorytmami logicznymi. System ten zapewnia wygodną obsługę (zarówno kontrolerów ruchu jak i paneli HMI) i jest przystosowany do współpracy w systemach sieciowych. Dzięki swojej kompaktowej budowie, atrakcyjnej cenie i wysokiej wydajności, sterowniki PLC firmy Samkoon stanowią idealne rozwiązanie dla wielu różnych aplikacji.

Ta instrukcja została starannie zweryfikowana pod względem zgodności opisu oprogramowania z rzeczywistością. Jednakże, nie można zagwarantować, że nie ma żadnych błędów ani nieprawidłowości. Wyklucza się wszelką odpowiedzialność i gwarancję co do pełnej dokładności zawartych informacji. Zawartość jest regularnie aktualizowana i uzupełniana. Wszelkie niezbędne poprawki wprowadzane będą w kolejnych wersjach. W celu uzyskania szczegółowych informacji należy zapraszać do kontaktu z działem technicznym.



## Spis treści:

OPERACJE LOGICZNE.....	6
PRZYKŁADY PROGRAMÓW:.....	7
LD/AND/OR.....	7
LDIM/ANDIM/ORIM.....	8
LDI/ANDI/ORI.....	9
LDIIM/ANDIIM/ORIIM.....	10
LDP/ANDP/ORP.....	11
LDF/ANDF/ORF.....	12
MEP.....	13
MEF.....	13
INV.....	14
OUT.....	15
OUTIM.....	16
SET.....	16
SETIM.....	17
RST.....	18
RSTIM.....	19
ALT.....	19
ALTP.....	20
INSTRUKCJE PORÓWNIANIA.....	21
LDW=/LDD=/LDF=.....	21
LDW <>/LDD <>/LDF <>.....	23
LDW >=/LDD >=/LDF >=.....	25
LDW <=/LDD <=/LDF <=.....	27
LDW>/LDD>/LDF>.....	29
LDW</LDD</LDF<.....	31
CMP/CMPD/CMPF.....	32
ZCP/ZCPD/ZCPF.....	34
INSTRUKCJE KONWERSJI.....	37
WTOD.....	37
DTOW.....	38
DTOF.....	39
BIN/BIND.....	40
BCD/BCDD.....	41
ROUND.....	42
TRUNC.....	43
INSTRUKCJE OPERACJI LOGICZNYCH.....	44
INVV/INVD.....	44
ANDW/ANDD.....	46
ORW/ORD.....	47
XORW/XORD.....	48
NEGW/NEGD.....	50
INSTRUKCJE TRANSFERU.....	52



MOV/MOVD/MOVB .....	52
MVBLK/MVDBLK .....	53
FMOV/FMOVD .....	55
SMOV .....	56
XCH/XCHD/XCHF .....	58
INSTRUKCJE OBLICZEŃ NA DANYCH ZMIENNOPRZECINKOWYCH .....	60
ADDF .....	60
SUBF .....	61
MULF .....	61
DIVF .....	62
SQRT .....	63
SIN .....	63
COS .....	64
TAN .....	65
LN .....	65
EXP .....	66
LOG .....	67
POW .....	67
ABSF .....	68
RAD/DEG .....	69
INSTRUKCJE NA DANYCH TYPU INTEGER .....	71
ADD/ADD .....	71
SUB/SUB .....	72
MUL/MULW/MULD .....	73
DIV/DIVW/DIVD .....	75
INC/INCD .....	76
DEC/DECD .....	77
FACT .....	78
ABSF ABS/ABSD .....	79
INSTRUKCJE TIMERA .....	81
TON .....	81
TONR .....	83
TOF .....	84
INSTRUKCJE LICZNIKA .....	86
CTU .....	86
CTD .....	87
CTUD .....	88
INSTRUKCJE REGULACJI PROGRAMU .....	90
FOR~NEXT .....	90
CJ~LBL .....	91
CJ .....	91
LBL .....	91
CALL .....	92
CALLM .....	93
STL/STLE/ST .....	96
STL .....	96
ST .....	96





RET.....	102
BREAK .....	104
FOR do NEXT .....	104
STL do STLE .....	104
IST .....	105
Powiązane rejestry .....	106
ISTNEXT.....	117
WDT .....	118
INSTRUKCJE PRZESUNIĘCIA .....	120
SHL/SHLD .....	120
SHR/SHRD .....	121
ROL/ROLD .....	123
ROR/RORD .....	124
SHLB.....	126
SHRB .....	127
INSTRUKCJE PRZERWANIA.....	129
ATCH .....	130
DTCH .....	131
EI.....	132
DI .....	132
INSTRUKCJE ZEGARA CZASU RZECZYWISTEGO .....	133
TRD .....	133
TWR .....	134
TRDS .....	135
TWRS .....	135
TSEC.....	136
RSEC.....	136
TCMP .....	137
TZCP .....	138
WKCMP .....	138
WKZCP .....	139
CKCMP .....	140
CKZCP .....	141
KOMUNIKACJA .....	143
MBUS.....	145
SEND .....	149
REV .....	151
CRC .....	153
CRCHECK.....	154
INSTRUKCJE IMPULSU .....	156
PLSF .....	158
PWM.....	160
PWMS .....	161
PLSY .....	163
PLSR/PLSA/EPLSR .....	165
PLSR/PLSA .....	166
EPLAR.....	167



PLSNEXT .....	169
PLSSTOP .....	171
DRVI/DRVA .....	172
POLYLINEF/POLYLINEI .....	174
LINEF/LINEI .....	183
ARCF/ARCI .....	185
BLOCK .....	187
PAUSE .....	189
ZRNR .....	190
JOG .....	197
FOLLOW .....	198
ECAM .....	199
Mechaniczny CAM i elektroniczny CAM .....	200
Clutch (sprzęgło) .....	214
Parametry stanu pracy .....	214
Aplikacja .....	214
EDRVI/EDRVA .....	217
INSTRUKCJE SZYBKIEGO ZLICZANIA .....	221
HCNT .....	221
EHCNT .....	225
HSCS .....	227
INSTRUKCJE NA CIĄGACH ZNAKÓW .....	230
I_S .....	231
DI_S .....	232
R_S .....	233
S_I .....	234
S_DI .....	235
S_R .....	237
INSTRUKCJE STEROWANIA PID .....	239
EPID .....	239
DODATEK .....	245
Rejestry funkcji specjalnych systemu .....	245



## OPERACJE LOGICZNE

### Uwagi

1. Instrukcje w tym rozdziale wykonują operacje logiczne na odpowiednich bitach.
2. Bity mają kilka typów, szczegóły zawarte są w tabeli poniżej.

### Rodzaje wejść/wyjść :

Typ rejestru	Opis	System liczbowy	zakres
X	Wejściowy bit PLC, niektóre z nich odpowiadają rzeczywistym stykom wejściowym, a pozostałe działają jak wewnętrzny rejestr bitowy.	ósemkowy	■ FGs_16MR/FGs_16MT 0~77 ■ inne 0~177
Y	Wyjściowy bit PLC, niektóre z nich odpowiadają rzeczywistym stykom wyjściowym, a pozostałe działają jak wewnętrzny rejestr bitów.	ósemkowy	■ FGs_16MR/FGs_16MT 0~77 ■ inne 0~177
M	Wewnętrzny rejestr bitów PLC.	dziesiętny	0~8223
T	Bit wyjściowy timera (licznika czasu), załączany, gdy timer osiągnie liczbę docelową, może być użyty do zresetowania timera.	dziesiętny	0~255
C	Bit wyjściowy licznika, ustawiony, gdy licznik osiągnie liczbę docelową, może być użyty do zresetowania licznika.	dziesiętny	0~255
S	Znacznik stanu systemu, używany w sterowaniu programem.	dziesiętny	0~999



## PRZYKŁADY PROGRAMÓW:

### LD/AND/OR

1. Ta instrukcja jest instrukcją uruchomienia styku **NO** (normalnie otwartego), ustawia określony bit jako styk wejściowy. Gdy wartość bitu wynosi 1, styk zwiera się; Gdy wartość bitu wynosi 0, styk rozwiera się.
2. Niniejsza instrukcja przedstawia styk  $\neg \mid$  w schemacie drabinkowym
3. W tabeli instrukcji, styki wejściowe można również połączyć przez **AND** (instrukcja połączenia szeregowego zestyku zwiernego) i **OR** (instrukcja połączenia równoległego zestyku zwiernego) (patrz przykład).

#### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Bit używany jako styk wejściowy NO	X/Y/M/T/C/S	-	Wartość logiczna

### Przykład

#### Lista instrukcji:

Network 000

```
LD   X000 // dodaj styk normalnie otwarty X000, styk zwiera się gdy X000 jest w pozycji ON
OR   X002 // dodaj styk normalnie otwarty X002 aby wykonać połączenie równoległe X000
AND  X001 // dodaj styk normalnie otwarty X001 aby wykonać połączenie szeregowe z X000
OUT  Y000 //dodaj Y000 jako cewkę wyjściową.
```


#### POP

#### Schemat drabinkowy:





## LDIM/ANDIM/ORIM

1. Ta instrukcja jest instrukcją uruchomienia styku priorytetowego szybkiej reakcji **NO** (normalnie otwartego), ustawia określony bit jako styk wejściowy. Gdy wartość bitu wynosi 1, styk zwiiera się bezzwłocznie; Gdy wartość bitu wynosi 0, styk otwiera się bezzwłocznie, niezależnie od odświeżenia okresu skanowania PLC.
2. Niniejsza instrukcja przedstawia styk  w schemacie drabinkowym.
3. W tabeli instrukcji, styki wejściowe można również połączyć przez **ANDIM** (instrukcja połączenia szeregowego styku zwiernego **NO** natychmiastowego działania) i **ORIM** (instrukcja połączenia równoległego styku zwiernego **NO** natychmiastowego działania) (patrz przykład).

### Ustawienia danych

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Bit używany jako styk wejściowy NO	X/Y/M/T/C/S	-	Wartość logiczna

### Przykład

#### Lista instrukcji:

Network 000

```
LDIM X000 // dodaj styk normalnie otwarty natychmiastowego działania X000
ORIM X002 // dodaj styk normalnie otwarty X002 aby wykonać połączenie równoległe z X000
ANDIM X001 // dodaj styk normalnie otwarty X001 aby wykonać połączenie szeregowe z X000
OUT Y000 //dodaj Y000 jako cewkę wyjściową.
```

POP

#### Schemat drabinkowy:





## LDI/ANDI/ORI

1. Ta instrukcja jest instrukcją uruchomienia styku **NC** (normalnie zamkniętego), ustawia określony bit jako styk wejściowy. Gdy wartość bitu wynosi 1, styk rozwiera się; Gdy wartość bitu wynosi 0, styk zwiiera się.
2. Niniejsza instrukcja przedstawia styk  $\neg|$  w schemacie drabinkowym.
3. W tabeli instrukcji, styki wejściowe można również połączyć przez **ANDI** (instrukcja połączenia szeregowego zestyku rozwiernego NC) i **ORI** (instrukcja połączenia równoległego zestyku rozwiernego) (patrz przykład).

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Bit używany jako styk wejściowy NC	X/Y/M/T/C/S	-	Wartość logiczna

### Przykład

#### Lista instrukcji:

Network 000

```
LDI X000 // dodaj styk normalnie zamknięty X000
ORI X002 // dodaj styk normalnie zamknięty X002 aby wykonać połączenie równoległe z X000
ANDI X001 // dodaj styk normalnie zamknięty X001 aby wykonać połączenie szeregowe z X000
OUT Y000 //dodaj Y000 jako cewkę wyjściową.
```

POP

#### Schemat drabinkowy:





## LDIIM/ANDIIM/ORIIM

1. Ta instrukcja jest instrukcją uruchomienia styku priorytetowego szybkiej reakcji **NC** (normalnie zamkniętego), ustawia określony bit jako styk wejściowy. Gdy wartość bitu wynosi 1, styk rozwiera się bezzwłocznie; Gdy wartość bitu wynosi 0, styk bezzwłocznie zwiiera się, niezależnie od odświeżenia okresu skanowania PLC.
2. Niniejsza instrukcja przedstawia styk  $\neg|$  w schemacie drabinkowym.
3. W tabeli instrukcji, styki wejściowe można również połączyć przez **ANDIIM** (instrukcja połączenia szeregowego priorytetowego styku rozwiernego szybkiej reakcji (NC) i **ORIIM** (instrukcja połączenia równoległego priorytetowego styku rozwiernego szybkiej reakcji (NC) (patrz przykład).

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Bit używany jako styk wejściowy szybkiej reakcji NC	X/Y/M/T/C/S	-	Wartość logiczna

### Przykład

#### Lista instrukcji:

Network 000

```

LDIIM      X000 // dodaj priorytetowy styk szybkiej reakcji normalnie zamknięty X000
ORIIM      X002 // dodaj priorytetowy styk szybkiej reakcji normalnie zamknięty X002 aby wykonać
połączenie równoległe X000
ANDIIM     X001 // dodaj priorytetowy styk szybkiej reakcji normalnie zamknięty X001 aby wykonać
połączenie szeregowe z X000
OUT        Y000 //dodaj Y000 jako cewkę wyjściową.
POP
  
```



Schemat drabinkowy:



## LDP/ANDP/ORP

1. Ta instrukcja jest instrukcją uruchomienia styku impulsowego reagującego na narastające zbocze sygnału, ustawia określony bit jako styk wejściowy. Styk zamyka się na okres skanowania tylko przy narastającym zboczu (OFF do ON) bitu.
2. Niniejsza instrukcja przedstawia styk  $\neg \uparrow$  w schemacie drabinkowym.
3. W tabeli instrukcji, styki wejściowe można również połączyć przez **ANDP** (instrukcja połączenia szeregowego styku impulsowy reagującego na narastające zbocze sygnału) i **ORP** (instrukcja połączenia równoległego styku impulsowego reagującego na narastające zbocze sygnału) (patrz przykład).

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Bit używany jako styk impulsowy reagujący na narastające zbocze sygnału	X/Y/M/T/C/S	-	Wartość logiczna

## Przykład

Lista instrukcji:

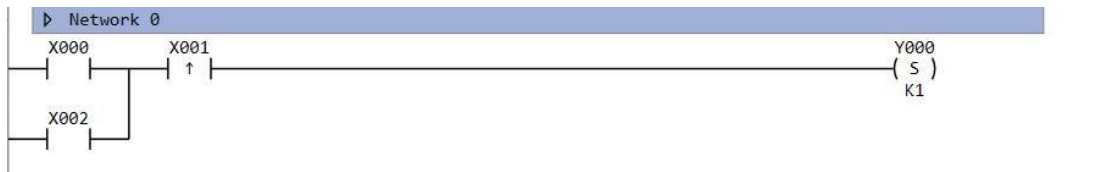
Network 000

```
LD  X000 // dodaj styk normalnie otwarty X000, cewka będzie zamknięta gdy X000 będzie w pozycji ON
OR  X002 // dodaj styk normalnie otwarty X002 aby wykonać połączenie równoległe z X000
ANDP X001 // dodaj styk impulsowy reagujący na narastające zbocze sygnału X001 aby wykonać połączenie szeregowe z X000
SET  Y000      K1 //ustaw Y000
POP
```





### Schemat drabinkowy:



## LDF/ANDF/ORF

1. Ta instrukcja jest instrukcją uruchomienia styku impulsowego reagującego na opadające zbocze sygnału, ustawia określony bit jako styk wejściowy. Styk zamyka się na okres skanowania tylko przy opadającym zboczu (ON do OFF) bitu.
2. Niniejsza instrukcja przedstawia styk  $\neg \downarrow$  w schemacie drabinkowym.
3. W tabeli instrukcji, styki wejściowe można również połączyć przez **ANDF** (instrukcja połączenia szeregowego styku impulsowy reagującego na opadające zbocze sygnału) i **ORF** (instrukcja połączenia równoległego styku impulsowego reagującego na opadające zbocze sygnału) (patrz przykład).

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Styk impulsowy reagujący na opadające zbocze sygnału	X/Y/M/T/C/S	-	Wartość logiczna

### Przykład

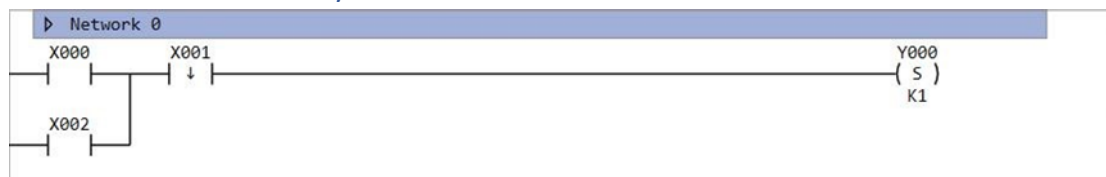
#### Lista instrukcji:

Network 000

```
LD   X000 // dodaj styk normalnie otwarty X000, cewka będzie zamknięta gdy X000 będzie w pozycji
      ON
OR   X002 // dodaj styk normalnie otwarty X002 aby wykonać połączenie równoległe z X000
ANDF X001 // dodaj styk impulsowy reagujący na opadające zbocze sygnału X001 aby wykonać
          połączenie szeregowe z X000
SET  Y000      K1 //ustaw Y000.
POP
```



Schemat drabinkowy:



## MEP

1. Ta instrukcja jest instrukcją włączania (stan przewodzenia) na okres skanowania przy narastającym zboczu (OFF do ON) operacji poprzedzającej. Ta instrukcja wyłącza (stan nieprzewodzący) w przypadkach innych niż zbocze narastające operacji poprzedzającej.
2. Niniejsza instrukcja przedstawia  $\text{---}\uparrow\text{---}$  w schemacie drabinkowym.

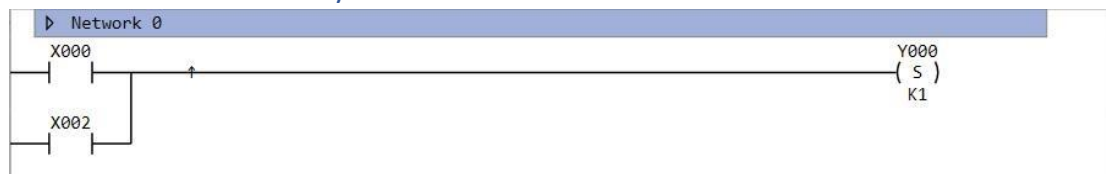
## Przykład

Lista instrukcji:

Network 000

```
LD   X000 // dodaj styk normalnie otwarty X000, cewka będzie zamknięta gdy X000 będzie w pozycji
      ON
OR   X002 // dodaj styk normalnie otwarty X002 aby wykonać połączenie równoległe z X000
MEP  // włącz przy narastającym zboczu będącej wynikiem operacji poprzedzającej
SET  Y000      K1 //ustaw Y000.
POP
```

Schemat drabinkowy:



## MEF

1. Ta instrukcja jest instrukcją włączania (stan przewodzenia) na okres skanowania przy opadającym zboczu (ON do OFF) operacji poprzedzającej. Ta instrukcja wyłącza (stan nieprzewodzący) w przypadkach innych niż zbocze opadające operacji poprzedzającej.
2. Niniejsza instrukcja przedstawia  $\text{---}\downarrow\text{---}$  w schemacie drabinkowym.



## Przykład

Lista instrukcji:


Network 000

```
LD   X000 // dodaj styk normalnie otwarty X000, cewka będzie zamknięta gdy X000 będzie w pozycji
      ON
OR   X002 // dodaj styk normalnie otwarty X002 aby wykonać połączenie równoległe z X000
MEP  // włącz przy narastającym zboczu będącym wynikiem operacji wyprzedzającej
SET  Y000      K1 //ustaw Y000.
POP
```

Schemat drabinkowy:



## INV

1. Ta instrukcja jest instrukcją odwracającą wynik operacji poprzedzającej.
2. Niniejsza instrukcja przedstawia  w schemacie drabinkowym.

## Przykład

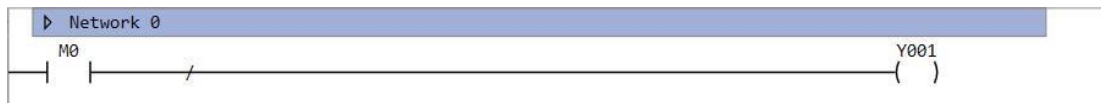
Lista instrukcji:

Network 000

```
LD   M000 // dodaj styk normalnie otwarty X000, cewka będzie zamknięta gdy X000 będzie w pozycji
      ON
INV  // odwróć wynik operacji
OUT  Y000 // włącz, gdy M000 jest wyłączony, wyłącz, gdy M000 jest włączone
POP
```



Schemat drabinkowy:



## OUT

1. To instrukcja wyprowadzania wyniku operacji do określonego bitu wyjściowego.
2. Niniejsza instrukcja przedstawia  $\text{---}(\text{---})\text{---}$  w schemacie drabinkowym.

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Bit używany jako styk wyjściowy	Y/M/S	-	Wartość logiczna

## Przykład

Lista instrukcji:

Network 000

LD X000 // dodaj styk normalnie otwarty X000,

OUT Y000 // włącz, gdy X000 jest włączone, wyłącz, gdy X000 jest wyłączony

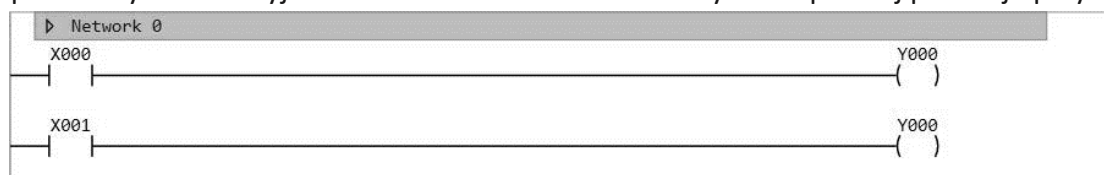
POP

Schemat drabinkowy:



## UWAGA

Nie dodawaj dwóch lub więcej wyjść OUT tego samego adresu bitowego w schemacie drabinkowym, w przeciwnym razie wyjście może zostać zablokowane. Rysunek poniżej pokazuje przykład **nieprawidłowej** instrukcji.





## OUTIM

1. To instrukcja wyprowadzania wyniku operacji do bitu wyjściowego, niezależnie od okresu skanowania PLC.
2. Niniejsza instrukcja przedstawia  $\text{---(I)---}$  w schemacie drabinkowym.

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Bit używany jako priorytetowy styk wyjściowy szybkiej reakcji	Y/M/S	-	Wartość logiczna

## Przykład

Lista instrukcji:

Network 000

LD X000 // dodaj styk normalnie otwarty X000,

OUTIM Y000 // włącz bezzwłocznie, gdy X000 jest włączone, wyłącz bezzwłocznie, gdy X000 jest wyłączony

POP

Schemat drabinkowy:



## SET

1. To instrukcja ustawiania (do wartości 1) określonego bitu i kolejnych.
2. Niniejsza instrukcja przedstawia  $\text{---(s)---}$  w schemacie drabinkowym.

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Adres pierwszego z ciągu bitów	Y/M/S	-	Wartość logiczna
(L)	Ustaw liczbę operacji	K/H	zależny od przypadku	32-bitowa liczba całkowita bez znaku



## Przykład

Lista instrukcji:

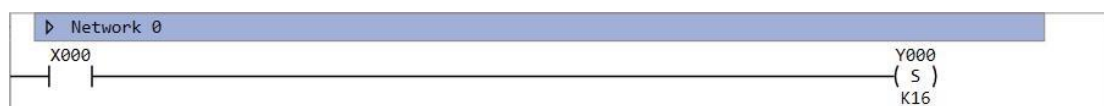
Network 000

LD X000 // dodaj styk normalnie otwarty X000,

SET Y00 K16 // ustaw bity od Y00 do Y015

POP

Schemat drabinkowy:



## SETIM

1. To instrukcja ustawiania (do wartości 1) określonego bitu i kolejnych, niezależnie od okresu skanowania PLC.
2. Niniejsza instrukcja przedstawia  $\text{---(S)}\text{---}$  w schemacie drabinkowym.

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Pierwszy z priorytetowych bitów szybkiej reakcji	Y/M/S	-	Wartość logiczna
(L)	Ustaw liczbę operacji	K/H	zależny od przypadku	32-bitowa liczba całkowita bez znaku

## Przykład

Lista instrukcji:

Network 000

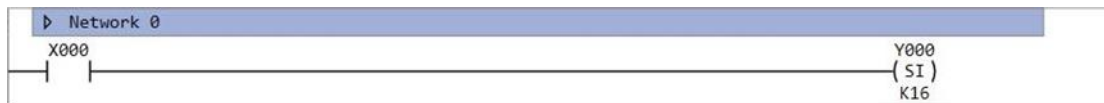
LD X000 // dodaj styk normalnie otwarty X000,

SETIM Y00 K16 // ustaw bezzwłocznie bity od Y000 do Y015

POP



### Schemat drabinkowy:



## RST

1. To instrukcja resetowania (do wartości 0) określonego bitu i kolejnych.
2. Niniejsza instrukcja przedstawia  $\text{—(R)—}$  w schemacie drabinkowym.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Adres pierwszego z ciągu bitów	Y/M/S	-	Wartość logiczna
(L)	Ustaw liczbę operacji	K/H	zależny od przypadku	32-bitowa liczba całkowita bez znaku

## Przykład

### Lista instrukcji:

Network 000

LD X000 // dodaj styk normalnie otwarty X000,

RST Y00 K16 // zresetuj bity od Y000 do Y015

POP

### Schemat drabinkowy:





## RSTIM

1. To instrukcja resetowania (do wartości 0) określonego bitu i kolejnych, niezależnie od okresu skanowania PLC.
2. Niniejsza instrukcja przedstawia  $\text{---(RI)---}$  w schemacie drabinkowym.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s)	Pierwszy z priorytetowych bitów szybkiej reakcji	Y/M/S	-	Wartość logiczna
(L)	Ustaw liczbę operacji	K/H	zależny od przypadku	32-bitowa liczba całkowita bez znaku

### Przykład

#### Lista instrukcji:

Network 000

LD X000 // dodaj styk normalnie otwarty X000,

RSTIM Y00 K16 // zresetuj bezzwłocznie bity od Y000 do Y015

POP

#### Schemat drabinkowy:



## ALT

Gdy ta instrukcja jest włączona, określony bit będzie odwracany (ON <-> OFF) w każdym cyklu skanowania PLC. Gdy to polecenie jest wyłączone, odwracanie jest zatrzymane.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenia		-	Wartość logiczna
(OUT)	Adres bitu do odwracania	Y/M/S	-	Wartość logiczna





## Przykład

Lista instrukcji:

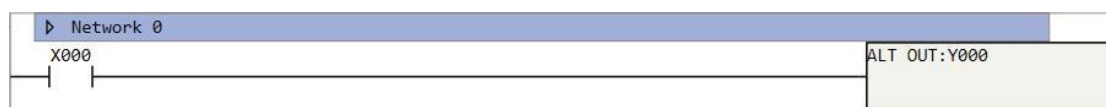
Network 000

LD X000 // dodaj styk normalnie otwarty X000,

ALT Y000 // Gdy X000 jest włączone, Y0 odwraca się w każdym czasie cyklu PLC

POP

Schemat drabinkowy:



## ALTP

Określony bit będzie odwracany (ON <-> OFF) **raz** przy narastającym zboczu sygnału bitu wejściowego.

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenia		-	Wartość logiczna
(OUT)	Bit do odwracania	Y/M/S	-	Wartość logiczna

## Przykład

Lista instrukcji:

Network 000

LD X000 // dodaj styk normalnie otwarty X000

ALTP Y000 // przy narastającym zboczu X000 Y0 odwróci się raz

POP

Schemat drabinkowy:





## INSTRUKCJE PORÓWNIANIA

### Uwagi

1. Instrukcje zamieszczone w tym rozdziale wykonują operacje porównywania wartości danych różnych typów.
2. Istnieją 3 typy danych: WORD, DWORD i FLOAT, szczegóły w tabeli poniżej.

Typ danych	Opis	Zakres
WORD	16-bitowa liczba całkowita ze znakiem	-32768~32767
DWORD	32-bitowa liczba całkowita ze znakiem	-2147483648~2147483647
FLOAT	Zmiennoprzecinkowe	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$

3. Istnieje kilka rodzajów rejestrów lub wejść, których można użyć do porównania.

Typ rejestru/wejście	Opis	System liczbowy	Zakres
K	Stała dziesiętna, może być liczbą całkowitą lub dziesiętną.	-	-
H	Kod szesnastkowy, używany do wypełnienia bajtów rejestru.	-	-
D	Wewnętrzny rejestr danych PLC.	Dziesiętny	0~8233
TV	Rejestr danych „timera”	Dziesiętny	0~255
CV	Rejestr danych licznika zliczającego zbrocze narastające.	Dziesiętny	0~255
AI	Rejestr danych wejścia analogowego.	Dziesiętny	0~31
AO	Rejestr danych wyjścia analogowego.	Dziesiętny	0~31
V/Z	Rejestr indeksu, który dodaje przesunięcie adresu. (Przykład użycia zamieszczony w pozycji LDW=/LDD=/LDF=)	Dziesiętny	0~7

### LDW=/LDD=/LDF=

1. Te instrukcje są instrukcjami porównawczymi EQ (ang. *equal to* - równa się), które porównują dwie wartości typu WORD/DWORD/FLOAT.

- ### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV32	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV32	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(s2)	Druga wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy

LDW<= D0 D1 // dokonaj porównania (mniejsze bądź równe) między D0 i D1  
ORW<= D4 D5 // dokonaj porównania (mniejsze bądź równe) między D4 i D5, połącz wynik równolegle  
AW<= D2 D3 // dokonaj porównania (mniejsze bądź równe) między D2 i D3, połącz wynik szeregowo



```

LDD<= D6 D8 // dokonaj porównania (mniejsze bądź równe) między D6(DWORD) i D8(DWORD)
ORD<= D14 D16 // dokonaj porównania (mniejsze bądź równe) między D14(DWORD) i D16(DWORD), połącz wynik
równolegle
AD<= D10 D12 dokonaj porównania (mniejsze bądź równe) D10 i D12, połącz wynik szeregowo
LDF<= D18 D20 // dokonaj porównania (mniejsze bądź równe) D18(FLOAT) i D20(FLOAT)
      ORF<= D26 D28 // dokonaj porównania (mniejsze bądź równe) D26(FLOAT) i D28(FLOAT), połącz
wynik szeregowo
      AF<= D22 D24 // dokonaj porównania (mniejsze bądź równe) D22(FLOAT) i D24(FLOAT), połącz wynik
równolegle
ORB // połącz równolegle
ORB // połącz równolegle
OUT Y000 //wyprowadź wyniki do wyjścia Y000
POP

```

Schemat drabinkowy:



## LDW <>/LDD <>/LDF <>

1. Te instrukcje są instrukcjami porównawczymi NE (ang. *not equal to* - nie równa się), które porównują dwie wartości typu WORD/DWORD/FLOAT.
2. Gdy pierwsza wartość jest nierówna drugiej, styk zamyka się, w przeciwnym razie styk rozwiera się.
3. Niniejsza instrukcja przedstawi  $\neg|W<>| \neg|D<>| \neg|F<>|$  w schemacie drabinkowym.
4. W tabeli instrukcji styki wejściowe można również podłączyć za pomocą [AW<>/AD<>/AF<>](#) (instrukcja połączenia szeregowego styków porównawczych NE) i [ORW<>/ORD<>/ORF<>](#) (instrukcja połączenia równoległego styków porównawczych NE) (patrz przykład).



Ustawienia danych:

## LDW<>

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem

## LDD<>

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV32	- 2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV32	- 2147483648~2147483647	32-bitowa liczba całkowita ze znakiem

## LDF<>

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(s2)	Druga wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy

## Przykład

Lista instrukcji:

Network 000

```
LDW<> D0 D1 // dokonaj porównania (nierówne) między D0 i D1
ORW<>      D4 D5 // dokonaj porównania (nierówne) między D4 i D5, połącz wynik równolegle
AW<>      D2 D3 // dokonaj porównania (nierówne) między D2 i D3, połącz wynik szeregowo
LDD<>      D6 D8 // dokonaj porównania (nierówne) między D6(DWORD) i D8(DWORD)
ORD<>      D14 D16 // dokonaj porównania (nierówne) między D14(DWORD) i D16(DWORD, połącz wynik równolegle
AD<>      D10 D12 dokonaj porównania (nierówne) D10 i D12, połącz wynik szeregowo
LDF<>      D18 D20 // dokonaj porównania (nierówne) D18(FLOAT) i D20(FLOAT)
ORF<>      D26 D28 // dokonaj porównania (nierówne) D26(FLOAT) i D28(FLOAT), połącz wynik szeregowo
AF<>      D22 D24 // dokonaj porównania (nierówne) D22(FLOAT) i D24(FLOAT), połącz wynik równolegle
ORB // połącz równolegle
```



ORB //połącz równolegle  
 OUT Y000 //wyprowadź wyniki do wyjścia Y000  
 POP

Schemat drabinkowy:



## LDW >=/LDD >=/LDF >=

1. Te instrukcje są instrukcjami porównawczymi GE (*ang. greater than or equal to*- większe bądź równe), które porównują dwie wartości typu WORD/DWORD/FLOAT.
2. Gdy pierwsza wartość jest większa lub równa drugiej, styk zamyka się, w przeciwnym razie styk otwiera się.
3. Niniejsza instrukcja przedstawia  $\text{---|W>=|---|D>=|---|F>=|}$  w schemacie drabinkowym.
4. W tabeli instrukcji styki wejściowe można również podłączyć za pomocą **AW>=/AD>=/AF>=** (instrukcja połączenia szeregowego styków porównawczych GE) i **ORW>=/ORD>=/ORF>=** (instrukcja połączenia równoległego styków porównawczych GE) (patrz przykład ).

Ustawienia danych:

## LDW>=

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem



## LDD>=

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV32	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV32	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem

## LDF>=

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(s2)	Druga wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy

## Przykład

### Lista instrukcji:

#### Network 000

```
LDW>= D0 D1 // dokonaj porównania (większe bądź równe) między D0 i D1
ORW>=      D4 D5 // dokonaj porównania (większe bądź równe) między D4 i D5, połącz wynik równolegle
AW>= D2 D3 // dokonaj porównania (większe bądź równe) między D2 i D3, połącz wynik szeregowo
LDD>= D6 D8 // dokonaj porównania (większe bądź równe) między D6(DWORD) i D8(DWORD)
ORD>= D14 D16 // dokonaj porównania (większe bądź równe) między D14(DWORD) i D16(DWORD), połącz wynik
równolegle
AD>= D10 D12 dokonaj porównania (większe bądź równe) D10 i D12, połącz wynik szeregowo
LDF>= D18 D20 // dokonaj porównania (większe bądź równe) D18(FLOAT) i D20(FLOAT)
ORF>= D26 D28 // dokonaj porównania (większe bądź równe) D26(FLOAT) i D28(FLOAT), połącz wynik szeregowo
AF>= D22 D24 // dokonaj porównania (większe bądź równe) D22(FLOAT) i D24(FLOAT), połącz wynik
równolegle
ORB // połącz równolegle
ORB // połącz równolegle
OUT Y000 //wyprowadź wyniki do wyjścia Y000
POP
```

### Schemat drabinkowy:



## LDW<=/LDD<=/LDF<=

- Te instrukcje są instrukcjami porównawczymi LE (*ang. less than or equal to*- mniejsze bądź równe), które porównują dwie wartości typu WORD/DWORD/FLOAT.
- Gdy pierwsza wartość jest mniejsza lub równa drugiej, styk zamyka się, w przeciwnym razie styk otwiera się.
- Niniejsza instrukcja przedstawia  $\neg W<=$   $\neg D<=$   $\neg F<=$  w schemacie drabinkowym.
- W tabeli instrukcji styki wejściowe można również podłączyć za pomocą **AW<=/AD<=/AF<=** (instrukcja połączenia szeregowego styków porównawczych LE) i **ORW<=/ORD<=/ORF<=** (instrukcja połączenia równoległego styków porównawczych LE) (patrz przykład ).

Ustawienia danych:

## LDW<=

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem

## LDD<=

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV32	- 2147483648~2147483647	32-bitowa liczba całkowita ze znakiem





(s2)	Dруга wartość do porównania	K/H/D/CV32	- 2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
------	-----------------------------	------------	----------------------------	---------------------------------------

## LDF<=

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(s2)	Dруга wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy

## Przykład

### Lista instrukcji:

#### Network 000

```
LDW<= D0 D1 // dokonaj porównania (mniejsze bądź równe) między D0 i D1
ORW<=      D4 D5 // dokonaj porównania (mniejsze bądź równe) między D4 i D5, połącz wynik równolegle
AW<= D2 D3 // dokonaj porównania (mniejsze bądź równe) między D2 i D3, połącz wynik szeregowo
LDD<= D6 D8 // dokonaj porównania (mniejsze bądź równe) między D6(DWORD) i D8(DWORD)
ORD<= D14 D16 // dokonaj porównania (mniejsze bądź równe) między D14(DWORD) i D16(DWORD), połącz wynik
równolegle
AD<= D10 D12 dokonaj porównania (mniejsze bądź równe) D10 i D12, połącz wynik szeregowo
LDF<= D18 D20 // dokonaj porównania (mniejsze bądź równe) D18(FLOAT) i D20(FLOAT)
ORF<= D26 D28 // dokonaj porównania (mniejsze bądź równe) D26(FLOAT) i D28(FLOAT), połącz wynik
szeregowo
AF<= D22 D24 // dokonaj porównania (mniejsze bądź równe) D22(FLOAT) i D24(FLOAT), połącz wynik
równolegle
ORB // połącz równolegle
ORB // połącz równolegle
OUT Y000 //wyprowadź wyniki do wyjścia Y000
POP
```



### Schemat drabinkowy:



## LDW>/LDD>/LDF>

1. Te instrukcje są instrukcjami porównawczymi GT (*ang. greater than* - większe od), które porównują dwie wartości typu WORD/DWORD/FLOAT.
2. Gdy pierwsza wartość jest większa, styk zamyka się, w przeciwnym razie styk otwiera się.
3. Niniejsza instrukcja przedstawia  $\text{---|W>|---|D>|---|F>|}$  w schemacie drabinkowym.
4. W tabeli instrukcji styki wejściowe można również podłączyć za pomocą [AW>/AD>/AF>](#) (instrukcja połączenia szeregowego styków porównawczych GT) i [ORW>/ORD>/ORF>](#) (instrukcja połączenia równoległego styków porównawczych GT) (patrz przykład).

### Ustawienia danych:

## LDW>

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem

## LDD>

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV32	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV32	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem



## LDF&gt;

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(s2)	Druga wartość do porównania	K/D/CV32	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy

## Przykład

## Lista instrukcji:

## Network 000

```
LDW> D0 D1 // dokonaj porównania (większe od) między D0 i D1
ORW> D4 D5 // dokonaj porównania (większe od) między D4 i D5, połącz wynik równolegle
AW> D2 D3 // dokonaj porównania (większe od) między D2 i D3, połącz wynik szeregowo
LDD> D6 D8 // dokonaj porównania (większe od) między D6(DWORD) i D8(DWORD)
ORD> D14 D16 // dokonaj porównania (większe od) między D14(DWORD) i D16(DWORD, połącz wynik równolegle
AD> D10 D12 dokonaj porównania (większe od) D10 i D12, połącz wynik szeregowo
LDF> D18 D20 // dokonaj porównania (większe od) D18(FLOAT) i D20(FLOAT)
ORF> D26 D28 // dokonaj porównania (większe od) D26(FLOAT) i D28(FLOAT), połącz wynik szeregowo
AF> D22 D24 // dokonaj porównania (większe od) D22(FLOAT) i D24(FLOAT), połącz wynik równolegle
ORB // połącz równolegle
ORB // połącz równolegle
OUT Y000 //wyprowadź wyniki do wyjścia Y000
POP
```

## Schemat drabinkowy:





## LDW</LDD</LDF<

1. Te instrukcje są instrukcjami porównawczymi LT (*ang. less than* - mniejsze od), które porównują dwie wartości typu WORD/DWORD/FLOAT.
2. Gdy pierwsza wartość jest mniejsza niż druga, styk zamyka się, w przeciwnym razie styk otwiera się.
3. Niniejsza instrukcja przedstawia  $\neg \text{W} < \neg \text{D} < \neg \text{F}$  w schemacie drabinkowym.
4. W tabeli instrukcji styki wejściowe można również podłączyć za pomocą [AW</AD</AF<](#) (instrukcja połączenia szeregowego styków porównawczych LT) i [ORW</ORD</ORF<](#) (instrukcja połączenia równoległego styków porównawczych LT) (patrz przykład ).

Ustawienia danych:

### LDW<

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem

### LDD<

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/H/D/CV32	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(s2)	Druga wartość do porównania	K/H/D/CV32	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem

### LDF<

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(s1)	Pierwsza wartość do porównania	K/D/CV32	$\pm 1.17549\text{e-}38\text{F} \sim \pm 3.40282\text{e}+38\text{F}$	zmiennoprzecinkowy
(s2)	Druga wartość do porównania	K/D/CV32	$\pm 1.17549\text{e-}38\text{F} \sim \pm 3.40282\text{e}+38\text{F}$	zmiennoprzecinkowy

### Przykład



### Lista instrukcji:

#### Network 000

```
LDW< D0 D1 // dokonaj porównania (mniejsze od) między D0 i D1
ORW< D4 D5 // dokonaj porównania (mniejsze od) między D4 i D5, połącz wynik równolegle
AW< D2 D3 // dokonaj porównania (mniejsze od) między D2 i D3, połącz wynik szeregowo
LDD< D6 D8 // dokonaj porównania (mniejsze od) między D6(DWORD) i D8(DWORD)
ORD< D14 D16 // dokonaj porównania (mniejsze od) między D14(DWORD) i D16(DWORD, połącz wynik równolegle
AD< D10 D12 dokonaj porównania (mniejsze od) D10 i D12, połącz wynik szeregowo
LDF< D18 D20 // dokonaj porównania (mniejsze od) D18(FLOAT) i D20(FLOAT)
ORF< D26 D28 // dokonaj porównania (mniejsze od) D26(FLOAT) i D28(FLOAT), połącz wynik szeregowo
AF< D22 D24 // dokonaj porównania (mniejsze od) D22(FLOAT) i D24(FLOAT), połącz wynik równolegle
ORB // połącz równolegle
ORB // połącz równolegle
OUT Y000 //wyprowadź wyniki do wyjścia Y000
POP
```

### Schemat drabinkowy:



## CMP/CMPD/CMPF

### Wprowadzenie do instrukcji

1. Poniższe instrukcje wyprowadzają wynik porównania dwóch wartości wejściowych do 3 kolejnych bitów wyjściowych. CMP jest używany dla wartości typu WORD, CMPD jest używany dla wartości typu DWORD, a CMPF jest używany dla wartości typu FLOAT.
2. Gdy pierwsza wartość wejściowa jest większa niż druga wartość wejściowa, pierwszy wyjściowy bit jest ustawiony na 1. Gdy pierwsza wartość wejściowa jest równa drugiej wartości wejściowej, drugi wyjściowy bit jest ustawiony na 1. Trzeci bit wyjściowy jest ustawiany na 1, gdy pierwsza wartość wejściowa jest mniejsza niż druga wartość wejściowa.



Ustawienia danych:

## CMP

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwsza wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(IN2)	Druga wartość do porównania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Pierwszy z wyjściowych bitów	Y/M/S	-	Wartość logiczna

## CMPD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwsza wartość do porównania	K/H/D/	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(IN2)	Druga wartość do porównania	K/H/D/	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Pierwszy z wyjściowych bitów	Y/M/S	-	Wartość logiczna

## CMPF

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwsza wartość do porównania	K/D/	$\pm 1.17549\text{e-}38\text{F} \sim \pm 3.40282\text{e+}38\text{F}$	zmiennoprzecinkowy



(IN2)	Druga wartość do porównania	K/D/	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Pierwszy z wyjściowych bitów	Y/M/S	-	Wartość logiczna

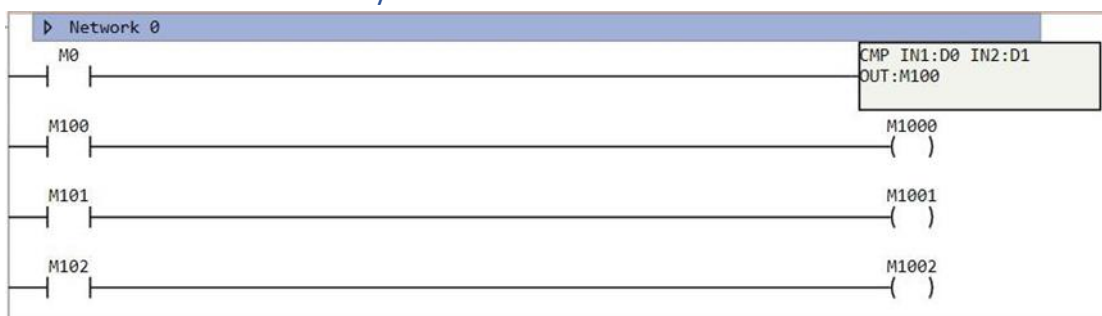
## Przykład

### Lista instrukcji:

Network 000

```
LD M0
CMP D0 D1 M100 //porównaj D0 i D1
POP
LD M100 //jeżeli D0>D1, ustaw M100 w pozycji ON
OUT M1000
POP
LD M101 //jeżeli D0=D1, ustaw M101 w pozycji ON
OUT M1001
POP
LD M102 // jeżeli D0<D1, ustaw M102 w pozycji ON
OUT M1002
POP
```

### Schemat drabinkowy:



## ZCP/ZCPD/ZCPF

1. Instrukcje te działają poprzez porównanie wartości porównywanej z przedziałem składającym się z 2 wartości granicznych i wyprowadzają wynik do 3 kolejnych wyjść. ZCP to wartość typu WORD, ZCPD to wartość typu DWORD, ZCPF to wartość typu FLOAT.
2. Gdy wartość porównania jest mniejsza niż dolna wartość graniczna, pierwszy z wyjściowych bitów jest ustawiany na 1; Gdy wartość porównania jest pomiędzy dolną wartością graniczną a górną wartością graniczną (nie większa niż dolna wartość graniczna i nie mniejsza niż górna wartość graniczna), drugi z wyjściowych bitów jest ustawiany na 1; Gdy wartość porównania jest większa niż górna wartość graniczna, trzeci z wyjściowych bitów jest ustawiany na 1.



## Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające lub wyłączające		0/1	Wartość logiczna

## ZCP

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Wartość dolnej granicy przedziału	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(IN2)	Wartość górnej granicy przedziału	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(IN3)	Wartość porównywana do porównania z przedziałem	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Pierwszy z wyjściowych bitów	Y/M/S	-	Wartość logiczna

## CMPD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Wartość dolnej granicy przedziału	K/H/D/	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(IN2)	Wartość górnej granicy przedziału	K/H/D/	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(IN3)	Wartość porównywana do porównania z przedziałem	K/H/D/	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Pierwszy z wyjściowych bitów	Y/M/S	-	Wartość logiczna

## CMPF

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
------------------	------	---------	--------	------------





(EN)	Wejście, które włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Wartość dolnej granicy przedziału	K/D/	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(IN2)	Wartość górnej granicy przedziału	K/D/	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(IN3)	Wartość porównywana do porównania z przedziałem	K/D/	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Pierwszy z wyjściowych bitów	Y/M/S	-	Wartość logiczna

## Przykład

### Lista instrukcji:

Network 000

LD M0

ZCP K100 K200 D0 Y000 //Gdy  $D0 < K100$ , Y000 jest w pozycji ON; Gdy  $K100 \leq D0 \leq K200$ , Y001 jest w pozycji ON; Gdy  $D0 > K200$ , Y002 jest w pozycji ON

POP

### Schemat drabinkowy:

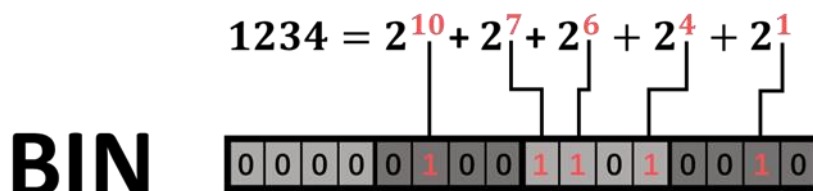




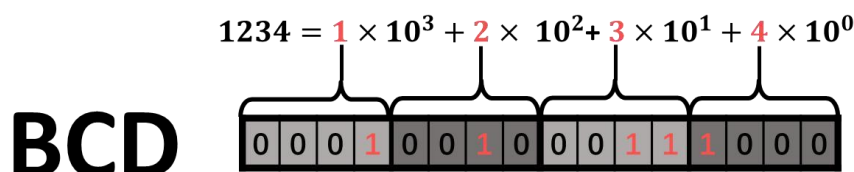
## INSTRUKCJE KONWERSJI

### Uwagi

1. Kod BIN jest kodem binarnym, rysunek poniżej pokazuje sposób kodowania.



2. Kod BCD jest zakodowanym binarnie kodem dziesiętnym, który konwertuje każdą cyfrę dziesiętną liczby na 4-bitowy kod binarny, a jego metodę kodowania pokazano na rysunku poniżej.



3. Mogą wystąpić błędy konwersji, a system posiada specjalne bity do rejestrowania tych błędów:

**M8168** - operacja nie powiedzie się, gdy dane DWORD są zbyt duże, aby można je było przekonwertować na WORD; marker zostanie ustawiony w pozycji ON.

**M8169** - konwersja BIN na BCD jest prawidłowa, jeśli kod BCD jest większy niż 0x9999; ten marker zostanie ustawiony w pozycji ON.

## WTOD

Ta instrukcja konwertuje dane WORD na dane DWORD i przechowuje przekonwertowane dane w określonych rejestrach D.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Dane WORD do przekonwertowania	K/H/D/CV/TV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr danych D do przechowywania	D	-	32-bitowa liczba całkowita ze znakiem



przekonwertowanych danych  
DWORD

## Przykład

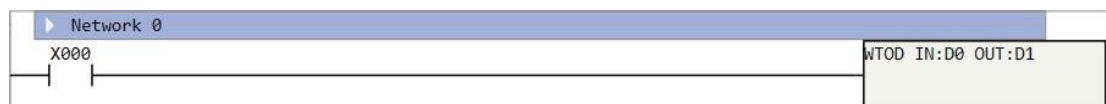
### Lista instrukcji:

Network 000

LD X000

WTOD D0 D1 //przekonwertuj dane WORD z D0 na dane typu DWORD i zapisz otrzymane dane w D1D2  
POP

### Schemat drabinkowy:



## DTOW

1. Ta instrukcja konwertuje dane DWORD na dane WORD i przechowuje przekonwertowane dane w określonych D-rejestrach.
2. Jeżeli dane typu DWORD są zbyt duże do konwersji, włącza się flaga przepełnienia rejestru bitowego **M8169**, a wyjściowy rejestr D nie wyświetli danych poprawnie.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Dane DWORD do przekonwertowania	K/H/D/	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	D-rejestr do przechowywania przekonwertowanych danych WORD	D	-	32-bitowa liczba całkowita ze znakiem

## Przykład

### Lista instrukcji:

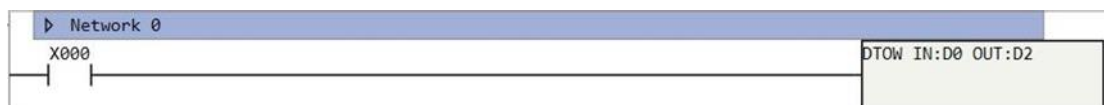
Network 000

LD X000



DTOW D0 D1 //przekonwertuj dane DWORD z D0D1 na dane typu WORD i zapisz otrzymane dane w D2  
POP

Schemat drabinkowy:



## DTOF

1. Ta instrukcja konwertuje dane typu DWORD na dane typu FLOAT i przechowuje przekonwertowane dane w określonych D-rejestrach.

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Dane DWORD do przekonwertowania	K/H/D/	$\pm 1.17549\text{e-}38\text{F} \sim \pm 3.40282\text{e+}38\text{F}$	zmiennoprzecinkowy
(OUT)	D-rejestr do przechowywania przekonwertowanych danych FLOAT	D	$\pm 1.17549\text{e-}38\text{F} \sim \pm 3.40282\text{e+}38\text{F}$	zmiennoprzecinkowy

## Przykład

Lista instrukcji:

Network 000

LD X000

DTOF D0 D1 //przekonwertuj dane DWORD z D0D1 na dane typu FLOAT i zapisz otrzymane dane w D2D3

POP

Schemat drabinkowy:





## BIN/BIND

1. Ta instrukcja konwertuje dane BCD do danych w formacie BIN i przechowuje przekonwertowane dane w określonych komórkach D. (szczegółowe informacje o danych BCD i BIN w podpunkcie uwagi). Instrukcja BIN jest używana dla danych 16-bitowych, BIND jest używana dla danych 32-bitowych.
2. Jeżeli konwersja jest niemożliwa, flaga przepełnienia rejestru bitowego **M8168** zostanie ustawiona w pozycji 1/ON.

Ustawienia danych:

### BIN

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	16-bitowe dane BCD do przekonwertowania	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0x9999	16-bitowy kod BCD
(OUT)	D-rejestr do przechowywania przekonwertowanych danych BIN	D/CV/TV/AO/V/Z		16-bitowy kod BIN

### BIND

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	32-bitowe dane BCD do przekonwertowania	K/H/D/CV	0x0000000~0x9999999	32-bitowy kod BCD
(OUT)	D-rejestr do przechowywania przekonwertowanych danych BIN	D/CV		32-bitowy kod BIN

### Przykład

Lista instrukcji:

Network 000

LD X000

BIN H0234 D0//przekonwertuj kod BCD 234 na dane typu BIN i zapisz otrzymane dane w D0



POP

Schemat drabinkowy:



## BCD/BCDD

1. Ta instrukcja konwertuje dane BIN na dane BCD i zapisuje przekonwertowane dane w określonych D-rejestrach. (szczegółowe informacje o danych BCD i BIN w podpunkcie uwagi). Instrukcja BCD jest używana dla danych 16-bitowych, BCDD jest używana dla danych 32-bitowych.
2. Jeżeli konwersja jest niemożliwa, flaga przepełnienia rejestru bitowego **M8168** zostanie ustawiona w pozycji 1/ON.

## BCD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	16-bitowe dane BIN do przekonwertowania	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod BIN
(OUT)	D-rejestr do przechowywania przekonwertowanych danych BIN	D/CV/TV/AO/V/Z		16-bitowy kod BCD

## BCDD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	32-bitowe dane BIN do przekonwertowania	K/H/D/CV	0x00000000~0xFFFFFFFF	32-bitowy kod BCD
(OUT)	D-rejestr do przechowywania przekonwertowanych danych BCD	D/CV		32-bitowy kod BCD



## Przykład

### Lista instrukcji:

Network 000

```
LD X000
BCD H0234      D1//przekonwertuj kod BIN Z D0 na dane typu BCD i przechowuj otrzymane dane w D1
POP
```

### Schemat drabinkowy:



## ROUND

1. Ta instrukcja zaokrągla dane typu FLOAT do danych typu DWORD i przechowuje przekonwertowane dane w określonym D-rejestrze. Dane FLOAT są zaokrąglane w dół, jeśli ułamkowa część danych FLOAT jest mniejsza niż 0,5; w przeciwnym razie jest zaokrąglana w górę.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Dane FLOAT do zaokrąglenia	K/D/	$\pm 1.17549\text{e-}38\text{F} \sim \pm 3.40282\text{e+}38\text{F}$	zmiennoprzecinkowy
(OUT)	D-rejestr do przechowywania zaokrąglonych danych DWORD	D	-	32-bitowa liczba całkowita ze znakiem

## Przykład

### Lista instrukcji:

Network 000

```
LD X000
ROUND K4.56      D0 //zaokrąglaj w górę K4.56 do K5 i przechowuj otrzymane dane w D0D1
POP
```



Schemat drabinkowy:



## TRUNC

1. Ta instrukcja obcina dziesiętną część danych w formacie FLOAT i konwertując je na dane typu DWORD; przechowuje przekonwertowane dane w określonym D-rejestrze.

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Dane FLOAT do obcięcia	K/D/	$\pm 1.17549\text{e-}38\text{F} \sim \pm 3.40282\text{e+}38\text{F}$	zmiennoprzecinkowy
(OUT)	D-rejestru do przechowywania zaokrąglonych danych DWORD	D	-	32-bitowa liczba całkowita ze znakiem

## Przykład

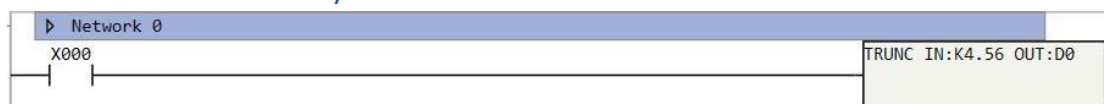
Lista instrukcji:

Network 000

LD X000

TRUNC      K4.56      D0 //zaokrąglaj w dół K4.56 do K4 i przechowuj otrzymane dane w D0D1  
POP

Schemat drabinkowy:







## INSTRUKCJE OPERACJI LOGICZNYCH

### Uwagi

Instrukcje w tym rozdziale wykonują logiczne operacje bitowe na danych WORD (16-bitowych) lub DWORD (32-bitowych). Operacje logiczne obejmują operację odwrócenia (**INV**), operację „i” (**AND**), operację „lub” (**OR**), operację alternatywy rozłącznej (**XOR**) i operację dopełnienia (**NEG**). **INV** i **NEG** działają na pojedynczym kodzie; **AND**, **OR**, **XOR** działają na dwóch kodach, szczegóły zawarte w tabelach poniżej.

Operator	Opis	Przykład	
		Wejście	Wyjście
INV	Odwróć każdy bit kodu.	0xFFFF	0x0000
NEG	Odwróć każdy bit kodu i dodaj 1 do wyniku, aby uzyskać uzupełnienie, które jest negatywną formą kodu źródłowego.	0xFFFF	0x0001

Operator	Opis	Przykład		
		Wejście1	Wejście2	Wyjście
AND	Odpowiednie bity dwóch kodów są dodawane i wynik jest wyprowadzany. (Jeśli co najmniej jeden bit ma wartość 0, wynikiem operacji jest 0, w przeciwnym razie wynikiem jest 1).	0	0	0
		0	1	0
		1	0	0
		1	1	1
OR	Wykonaj operację „lub” na każdym odpowiednim bicie dwóch kodów i wynik wyjściowy. (Jeśli co najmniej jeden z dwóch bitów ma wartość 1, wynikiem operacji jest 1, w przeciwnym razie wynikiem jest 0).	0	0	0
		0	1	1
		1	0	1
		1	1	1
XOR	Wykonaj operację alternatywy rozłącznej na każdym odpowiednim bicie dwóch kodów i wynik wyjściowy. (Jeśli dwa bity są różne, wynikiem operacji jest 1, w przeciwnym razie wynikiem jest 0).	0	0	0
		0	1	1
		1	0	1
		1	1	0

### INVW/INVD



Instrukcje te wykonują operację odwrócenia na kodzie wejściowym i przechowują wynik w określonym rejestrze. **INVW** dotyczy danych WORD, **INVD** dotyczy danych DWORD. Szczegóły operacji odwracania znajdują się w uwagach w tym rozdziale.

Ustawienia danych:

## INVW

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Kod do wykonania operacji odwracania.	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod
(OUT)	D-rejestr do przechowywania kodu wyjściowego	D/CV/TV/AO/V/Z		16-bitowy kod

## INVD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Kod do wykonania operacji odwracania.	K/H/D/	0x00000000~0xFFFFFFFF	32-bitowy kod
(OUT)	D-rejestr do przechowywania kodu wyjściowego	D		32-bitowy kod

## Przykład

Lista instrukcji:

Network 000

LD X000

INVW D0 D1 // wykonaj operację odwrócenia na kodzie w D0 i zapisz wynik w D1

POP



Schemat drabinkowy:



## ANDW/ANDD

Instrukcje te wykonują operacje na kodach wejściowych i przechowują wynik w określonym rejestrze. ANDW dotyczy danych WORD, ANDD dotyczy danych DWORD. Szczegóły operacji zawarte są w punkcie [uwagi](#) na początku tego rozdziału.

Ustawienia danych:

### ANDW

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwszy kod do wykonania operacja AND	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod
(IN2)	Drugi kod do wykonania operacja AND	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod
(OUT)	D-rejestr do przechowywania kodu wyjściowego	D/CV/TV/AO/V/Z	-	16-bitowy kod

### ANDD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwszy kod do wykonania operacja AND	K/H/D/	0x00000000~0xFFFFFFFF	32-bitowy kod



(IN2)	Drugi kod do wykonania operacja AND	K/H/D	0x00000000~0xFFFFFFFF	32-bitowy kod
(OUT)	D-rejestr do przechowywania kodu wyjściowego	D	-	32-bitowy kod

## Przykład

### Lista instrukcji:

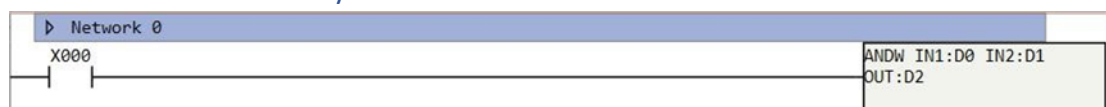
Network 000

LD X000

ANDW D0 D1 D2 // wykonaj operację AND na kodach w D0 i D1, a wynik zapisz w D2

POP

### Schemat drabinkowy:



## ORW/ORD

Instrukcje te wykonują operacje na kodach wejściowych i przechowują wynik w określonym rejestrze. **ORW** dotyczy danych WORD, **ORD** dotyczy danych DWORD. Szczegóły operacji zawarte są w punkcie [uwagi](#) na początku tego rozdziału.

### Ustawienia danych:

## ORW

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwszy kod do wykonania operacja OR	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod
(IN2)	Drugi kod do wykonania operacja OR	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod



(OUT)	D-rejestr do przechowywania kodu wyjściowego	D/CV/TV/AO/V/Z	-	16-bitowy kod
-------	--	----------------	---	---------------

## ORD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwszy kod do wykonania operacja OR	K/H/D/	0x00000000~0xFFFFFFFF	32-bitowy kod
(IN2)	Drugi kod do wykonania operacja OR	K/H/D	0x00000000~0xFFFFFFFF	32-bitowy kod
(OUT)	D-rejestr do przechowywania kodu wyjściowego	D	-	32-bitowy kod

## Przykład

Lista instrukcji:

Network 000

LD X000

ORW D0 D1 D2 // wykonaj operację OR na kodach w D0 i D1, a wynik zapisz w D2

POP

Schemat drabinkowy:



## XORW/XORD

Instrukcje te wykonują operacje **XOR** na kodach wejściowych i przechowują wynik w określonym rejestrze. **XORW** dotyczy danych WORD, **XORD** dotyczy danych DWORD. Szczegóły operacji **XOR** zawarte są w punkcie [uwagi](#) na początku tego rozdziału.



Ustawienia danych:

## XORW

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwszy kod do wykonania operacja XOR	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod
(IN2)	Drugi kod do wykonania operacja XOR	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod
(OUT)	D-rejestr do przechowywania kodu wyjściowego	D/CV/TV/AO/V/Z	-	16-bitowy kod

## XORD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwszy kod do wykonania operacja XOR	K/H/D/	0x00000000~0xFFFFFFFF	32-bitowy kod
(IN2)	Drugi kod do wykonania operacja XOR	K/H/D	0x00000000~0xFFFFFFFF	32-bitowy kod
(OUT)	D-rejestr do przechowywania kodu wyjściowego	D	-	32-bitowy kod

## Przykład

Lista instrukcji:

Network 000

LD X000

XORW D0 D1 D2 // wykonaj operację XOR na kodach w D0 i D1, a wynik zapisz w D2

POP



### Schemat drabinkowy:



## NEGW/NEGD

Instrukcje te wykonują operacje działania uzupełniającego na kodach wejściowych i przechowują wynik w określonym rejestrze. **NEGW** dotyczy danych WORD, **NEGD** dotyczy danych DWORD. Szczegóły operacji **NEG** zawarte są w punkcie [uwagi](#) na początku tego rozdziału.

### Ustawienia danych:

## NEGW

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwszy kod do wykonania operacja NEG	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod
(IN2)	Drugi kod do wykonania operacja NEG	K/H/D/CV/TV/AI/AO/V/Z	0x0000~0xFFFF	16-bitowy kod
(OUT)	D-rejestr do przechowywania kodu wyjściowego	D/CV/TV/AO/V/Z	-	16-bitowy kod

## NEGD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Pierwszy kod do wykonania operacja NEG	K/H/D/	0x00000000~0xFFFFFFFF	32-bitowy kod



(IN2)	Drugi kod do wykonania operacja NEG	K/H/D	0x00000000~0xFFFFFFFF	32-bitowy kod
(OUT)	D-rejestr do przechowywania kodu wyjściowego	D	-	32-bitowy kod

## Przykład

### Lista instrukcji:

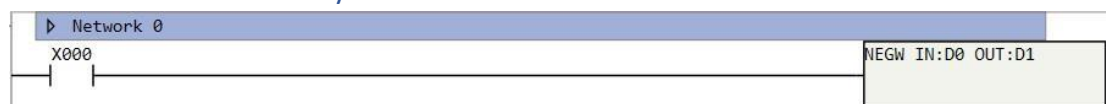
Network 000

LD X000

NEGW D0 D1 D2 // wykonaj operację NEG na kodzie w D0 , a wynik zapisz w D1

POP

### Schemat drabinkowy:







## INSTRUKCJE TRANSFERU

### Uwagi

Instrukcje w tym rozdziale przenoszą dane o określonej długości do określonych rejestrów. Długość danych może być różna, szczegóły w tabeli poniżej.

typ długości danych	Opis	Instrukcje odpowiadające
BIT	Minimalna jednostka danych	-
NIBBLE (półbajt)	4 bity	<b>SMOV</b>
BAJT	8 bitów	-
WORD	16 bitowe dane typu WORD	<b>MOV, MVBLK, FMOV, XCH</b>
DOUBLE-WORD	32 bitowe dane typu DWORD i FLOAT.	<b>MOVD/MOVF, MVDBLK, FMOVD, XCHD/XCHF</b>

## MOV/MOVD/MOVF

Instrukcje te kopiują dane z rejestrów źródłowych i przechowują je w rejestrach docelowych. **MOV** jest dla danych WORD, **MOVD** dla danych DWORD, **MOVF** dla danych FLOAT.

### Ustawienia danych:

## MOV

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Rejestr źródłowy.	D/CV/TV/AI/AO/V/Z	-	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy	D/CV/TV/AO/V/Z	-	16-bitowa liczba całkowita ze znakiem

## MOVD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
---------------------	------	---------	--------	------------



(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Rejestr źródłowy.	K/H/D/CV	-	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy	D/CV	-	32-bitowa liczba całkowita ze znakiem

## MOV

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Rejestr źródłowy.	K/D/AI/AO	-	zmiennoprzecinkowy
(OUT)	Rejestr docelowy	D	-	zmiennoprzecinkowy

## Przykład

### Lista instrukcji:

Network 000

LD X000

MOV D0 D1 // skopiuj dane WORD z D0 i zapisz w D1

POP

### Schemat drabinkowy:



## MVBLK/MVDBLK

Instrukcje te kopiują dane WORD/DOUBLE-WORD określonej długości z rejestrów źródłowych i przechowują je w rejestrach docelowych. **MVBLK** jest instrukcją dla danych WORD, **MVDBLK** dla danych DOUBLE-WORD.



Ustawienia danych:

## MVBLK

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(S)	Rejestr źródłowy.	D/CV/TV/AI/AO/V/Z	-	16-bitowy kod
(D)	Rejestr docelowy	D/CV/TV/AO/V/Z	-	16-bitowy kod
(N)	Ilość danych słownych do przesłania	K/H/D	1~1024	16-bitowa liczba całkowita bez znaku

## MVDBLK

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(S)	Rejestr źródłowy.	D/CV/TV/AI/AO/V/Z	-	32-bitowy kod
(D)	Rejestr docelowy	D/CV/TV/AO/V/Z	-	32-bitowy kod
(N)	Ilość danych DOUBLE-WORD do przesłania	K/H/D	1~1024	16-bitowa liczba całkowita bez znaku

## Przykład

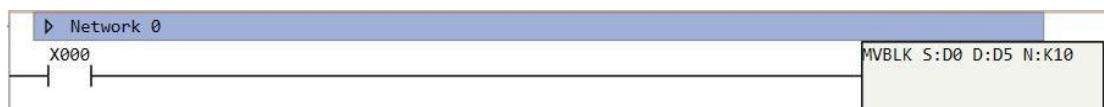
Lista instrukcji:

Network 000

```
LD      X000
MVBLK   D0    D5    K10 // skopiuj dane z D0-D9 i zapisz w D4-D14
POP
```



Schemat drabinkowy:



## FMOV/FMOVD

Instrukcje te kopiują dane WORD/DOUBLE-WORD z rejestrów źródłowych i przechowują je w wielu ciągłych rejestrach docelowych. **FMOV** jest instrukcją dla danych WORD, **FMOVD** dla danych DOUBLE-WORD.

Ustawienia danych:

### FMOV

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(S)	Rejestr źródłowy.	D/CV/TV/AI/AO/V/Z	-	16-bitowy kod
(T)	Rejestr docelowy	D/CV/TV/AO/V/Z	-	16-bitowy kod
(N)	Ilość danych słownych do przesłania	K/H/D	1~1024	16-bitowa liczba całkowita bez znaku

### FMOVD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(S)	Rejestr źródłowy.	K/H/D	-	32-bitowy kod
(T)	Rejestr docelowy	D	-	32-bitowy kod



(N)	Ilość danych DOUBLE-WORD do przesłania	K/H/D	1~8233	16-bitowa liczba całkowita bez znaku
-----	--	-------	--------	--------------------------------------

## Przykład

### Lista instrukcji:

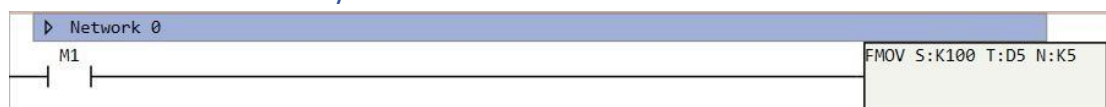
#### Network 000

LD X000

FMOV K100 D5 K5 // skopiuj K100 i zapisz w D5-D9

POP

### Schemat drabinkowy:



## SMOV

1. Ta instrukcja jest instrukcją przesunięcia cyfrowego, która dystrybuuje i łączy dane w jednostkach półbajtów (4 bity).
2. Ta instrukcja konwertuje kod binarny w rejestrze źródłowym (SV) i rejestrze docelowym (D) na kod BCD (0x0000 do 0x9999). Półbajty długości (SC), począwszy od (SS)-tego półbajtu, są przesyłane do rejestru docelowego (D) zaczynając od (DS)-tego półbajtu, konwertowane na kod binarny, a następnie zapisywane w rejestrze docelowym (D).

### Ustawienia danych:

## SMOV

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(SV)	Rejestr źródłowy.	K/H/D/CV/TV/AI/AO/ V/Z	-	16-bitowy kod BIN
(SS)	Początkowa pozycja półbajt przesunięcia	K/H	1~4	16-bitowa liczba całkowita bez znaku



(SC)	Ilość półbajtów do przesunięcia	K/H	1~4	16-bitowa liczba całkowita bez znaku
(DV)	Rejestr docelowy	D/CV/TV/AO/V/Z	-	16-bitowy kod BIN
(DS)	Bit docelowy	K/H	1~4	16-bitowa liczba całkowita bez znaku

Przykład:

Lista instrukcji:

Network 000

LDPM1

MOV K4321 D1 //nadpisz 4321 in D1

MOV K8888D2 //nadpisz 8888 in D2

POP

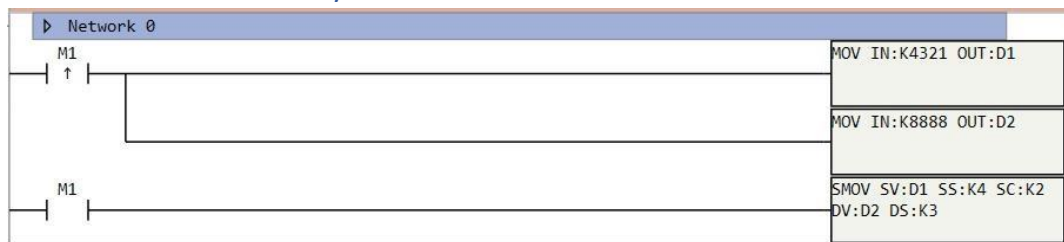
LD M1

SMOV D1 K4 K2 D2K3 // Gdy M1 jest w pozycji ON, D2 jest przechowywane w 8438

POP



## Schemat drabinkowy:



## XCH/XCHD/XCHF

Instrukcje te wymieniają dane z dwóch rejestrów. **XCH** jest dla danych WORD, **XCHD** dla danych DWORD, **XCHF** dla danych FLOAT.

## XCH

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(L)	Rejestr do wymiany.	D/CV/TV/AO/V/Z	-	16-bitowa liczba całkowita ze znakiem
(R)	Rejestr do wymiany.	D/CV/TV/AO/V/Z	-	16-bitowa liczba całkowita ze znakiem

## XCHD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(L)	Rejestr do wymiany.	D	-	32-bitowa liczba całkowita ze znakiem
(R)	Rejestr do wymiany.	D	-	32-bitowa liczba całkowita ze znakiem

## XCH

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna



(L)	Rejestr do wymiany.	D	-	zmiennoprzecinkowy
(R)	Rejestr do wymiany.	D	-	zmiennoprzecinkowy

## Przykład

### Lista instrukcji:

Network 000

LDP M7

XCH D5 D6 //Wymień dane D5 i D6

POP

### Schemat drabinkowy:







## INSTRUKCJE OBLICZEŃ NA DANYCH ZMIENNOPRZECINKOWYCH

### Uwagi

1. Rozdział zawiera instrukcje obliczeń na danych zmiennoprzecinkowych typu FLOAT. Obliczenia obejmują: **dodawanie, odejmowanie, mnożenie, dzielenie, pierwiastek, sinus, cosinus, tangens, logarytm, wykładnik, potęgę, wartość bezwzględną, arcus sinus, arcus cosinus, arcus tangens, kąt na radiany i radiany na kąt.**

2. Istnieją specjalne systemowe markery bitowe do rejestrowania błędów obliczeniowych.

**M8169:** W przypadku wystąpienia przepełnienia lub niedomiaru obliczeń.

**M8170:** Gdy wynik obliczeń jest ujemny lub dane wejściowe obliczeń są niedozwolone.

**M8171:** Gdy wynik obliczeń wynosi 0.

**M8172:** Gdy dzielnik lub dzielna wynosi 0.

### ADDF

1. Poniższa instrukcja dodaje jedno dane typu FLOAT (składnik sumy) do innych (składnik sumy) i zapisuje wynik w określonym rejestrze.

2. Ta instrukcja może włączyć markery **M8170** i **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Składnik sumy.	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(IN2)	Składnik sumy.	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

### Przykład

#### Lista instrukcji:

Network 000

```
LD X000
ADDF D0 D2 D4// D0D1+D2D3=D4D5
POP
```



Schemat drabinkowy:



## SUBF

1. Poniższa instrukcja odejmuje jedno dane typu FLOAT od innych i zapisuje wynik (różnicę) w określonym rejestrze.
2. Ta instrukcja może włączyć markery **M8170** i **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Odjemna.	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(IN2)	Odjemnik.	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

Lista instrukcji:

Network 000

LD X000

SUBFF D0 D2 D4// D0D1-D2D3=D4D5

POP

Schemat drabinkowy:



## MULF

1. Poniższa instrukcja mnoży jedno dane typu FLOAT (czynniki) z innymi (czynnikiem) i zapisuje wynik (iloczyn) w określonym rejestrze.
2. Ta instrukcja może włączyć markery **M8170** i **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
------------------	------	---------	--------	------------



(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Składowik mnożenia	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(IN2)	Składowik mnożenia	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

### Lista instrukcji:

Network 000

LD X000

MULF D0 D2 D4//  $D0D1 \times D2D3 = D4D5$

POP

### Schemat drabinkowy:



## DIVF

1. Ta instrukcja dzieli jedne dane typu FLOAT przez drugie i zapisuje wynik (różnicę) w określonym rejestrze.
2. Ta instrukcja może włączyć markery **M8170**, **M8171** i **M8172**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Dzielnia.	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(IN2)	Dzielnik.	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

### Lista instrukcji:

Network 000

LD X000

DIVF D0 D2 D4//  $D0D1 \div D2D3 = D4D5$

POP



## Schemat drabinkowy:



## SQRT

1. Jest to instrukcja obliczania pierwiastka kwadratowego z wejściowych danych typu FLOAT i przechowywania wyniku w określonym rejestrze.
2. Ta instrukcja może włączyć markery **M8170** i **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wartość początkowa do operacji pierwiastkowania	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

## Lista instrukcji:

Network 000

LD X000

SQRT K256 D0 //  $SQRT(256) = D0D1 = 16$ 

POP

## Schemat drabinkowy:



## SIN

1. Poniższa instrukcja oblicza sinus wejściowych danych typu FLOAT (kąt) i zapisuje wynik w określonym rejestrze. Jednostką wejściową jest stopień, użytkownik może również użyć instrukcji **DEG** do zamiany radianów na stopnie.
2. Ta instrukcja może włączyć markery **M8170** i **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
------------------	------	---------	--------	------------



(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Kąt wejściowy.	K/D/AI/AO	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

### Lista instrukcji:

Network 000

```
LD      X000
SIN     K30  D0// SIN(30°) = D0D1 = 0.5
```

POP

### Schemat drabinkowy:



## COS

- Poniższa instrukcja oblicza cosinus wejściowych danych typu FLOAT (kąt) i zapisuje wynik w określonym rejestrze. Jednostką wejściową jest stopień, użytkownik może również użyć instrukcji **DEG** do zamiany radianów na stopnie.
- Ta instrukcja może włączyć markery **M8170** i **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Kąt wejściowy.	K/D/AI/AO	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

### Lista instrukcji:

Network 000

```
LD      X000
COS     K60  D0// COS(60°) = D0D1 = 0.5
POP
```



## Schemat drabinkowy:



## TAN

1. Ta instrukcja oblicza tangens wejściowych danych typu FLOAT (kąt) i zapisuje wynik w określonym rejestrze. Jednostką wejściową jest stopień, użytkownik może również użyć instrukcji **DEG** do zamiany radianów na stopnie.
2. Ta instrukcja może włączyć markery **M8169**, **M8170** i **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Kąt wejściowy.	K/D/AI/AO	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

## Lista instrukcji:

Network 000

LD X000

TAN K45 D0// TAN(45°) = D0D1 = 1

POP

## Schemat drabinkowy:



## LN

1. Poniższa instrukcja oblicza logarytm naturalny (podstawa to stała naturalna, tj. w przybliżeniu równa 2,71828) wejściowych danych typu FLOAT i przechowuje wynik w określonym rejestrze.
2. Ta instrukcja może włączyć markery **M8169**, **M8170** i **M8171**.



Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wartość wejściowa – antylogarytm.	K/D/AI/AO	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

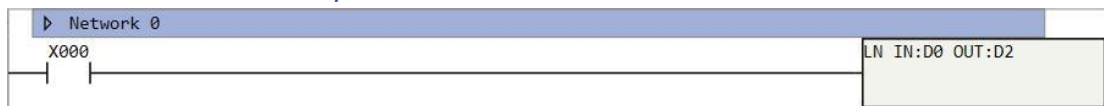
### Lista instrukcji:

Network 000

```
LD      X000
LN      D0      D2//LN(D0D1) = D2D3

POP
```

### Schemat drabinkowy:



## EXP

1. Jest to instrukcja wyznaczania wykładnika naturalnego funkcji wykładniczej  $e^x$  (ang. *natural exponential*) (podstawa to stała naturalna, tj. w przybliżeniu równa 2,71828) wejściowych danych typu FLOAT i przechowuje wynik w określonym rejestrze.
2. Ta instrukcja może włączyć marker **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wartość wejściowa.	K/D/AI/AO	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

### Lista instrukcji:

Network 000

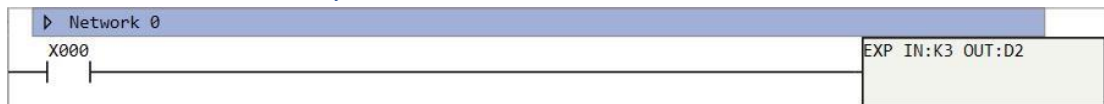
```
LD      X000
```



EXP K3 D2// EXP(3) = D2D3 = e^3

POP

Schemat drabinkowy:



## LOG

1. Jest to instrukcja wyznaczania logarytmu dziesiętnego (podstawa logarytmu jest równa 10) na wejściowych danych typu FLOAT i przechowuje wynik w określonym rejestrze.
2. Ta instrukcja może włączyć markery **M8169**, **M8170** i **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wartość wejściowa - antylogarytm.	K/D/AI/AO	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

Lista instrukcji:

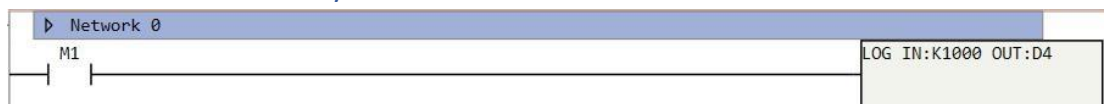
Network 000

LD X000

LOG K1000 D4 //LOG(K1000) = D4D5 = 3

POP

Schemat drabinkowy:



## POW

1. Poniższa instrukcja potęguje dane typu FLOAT pochodzące z dwóch wejść zmiennoprzecinkowych i przechowuje wynik w określonym rejestrze.





2. Ta instrukcja może włączyć marker **M8171**.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Podstawa	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(IN2)	Wykładnik potęgi	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

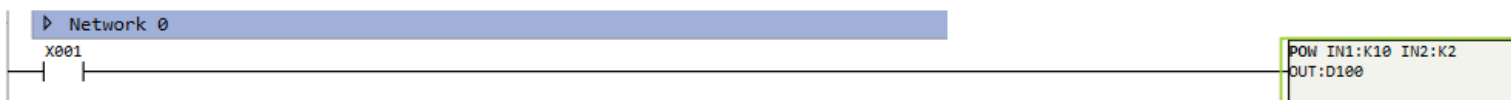
## Przykład

Lista instrukcji:

Network 000

```
LD    X001,
      POW    K10    K2    D100 //podnieś K10 do potęgi K2 i zapisz w D100
      POP
```

Schemat drabinkowy:



## ABSF

Poniższa instrukcja oblicza wartość bezwzględną na danych wejściowych typu FLOAT i zapisuje wynik we wskazanym rejestrze.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wejście do obliczenia wartości bezwzględnej.	K/D/	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

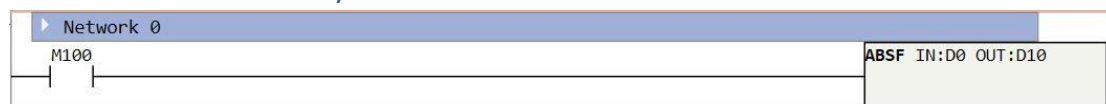


## Lista instrukcji:

Network 000

```
LD      M100
ABSF    D0    D10 //ABS(D0D1) = D10D11
POP
```

## Schemat drabinkowy:



## RAD/DEG

1. Instrukcja **RAD** konwertuje stopnie na radiany.
2. Instrukcja **DEG** konwertuje radiany na stopnie.

## Ustawienia danych

## RAD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wartość wejściowa (stopnie).	K/D/AI/AO	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## DEG

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wartość wejściowa (radiany).	K/D/AI/AO	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	zmiennoprzecinkowy
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	zmiennoprzecinkowy

## Przykład

## Lista instrukcji:

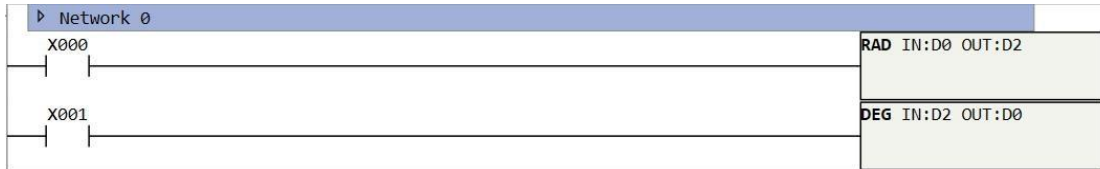
Network 000

```
LD      M100
RAD     D0    D2 //zamień stopnie z D0D1 na radiany w D2D3
```



```
LD      M100
DEG     D2    D0 //zamień radiany z D2D3 na stopnie w D0D1
POP
```

Schemat drabinkowy:





## INSTRUKCJE NA DANYCH TYPU INTEGER

1. Instrukcje w tym rozdziale wykonują obliczenia na danych typu integer (liczbach całkowitych). Obliczenia obejmują: **dodawanie, odejmowanie, mnożenie, dzielenie, inkrementację, dekrementację, silnię i wartość bezwzględną.**

2. Istnieją specjalne systemowe bity, rejestrujące błędy obliczeniowe:

**M8169:** Gdy w obliczeniach występuje przepełnienie lub niedopełnienie.

**M8170:** Gdy wynik obliczeń to minus lub dane wejściowe obliczeń to niedozwolone minusy.

**M8171:** Gdy wynik obliczenia wynosi 0.

**M8172:** Gdy dzielnik dzielenia wynosi 0.

### ADD/ADD

1. Poniższa instrukcja dodaje dane typu integer jednego wejścia (składnik sumy) do danych typu integer innego wejścia (składnik sumy) i zapisuje wynik w określonym rejestrze. Instrukcja **ADD** jest dla 16-bitowych liczb całkowitych ze znakiem, **ADD** jest dla 32-bitowych liczb całkowitych ze znakiem.

2. Ta instrukcja może włączyć markery **M8169**, **M8170** i **M8171**.

#### ADD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Składnik sumy.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(IN2)	Składnik sumy.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/TV/CV/AO/V/Z	-	16-bitowa liczba całkowita ze znakiem

#### ADD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna



(IN1)	Składnik sumy.	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(IN2)	Składnik sumy.	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV	-	32-bitowa liczba całkowita ze znakiem

## Przykład

### Lista instrukcji:

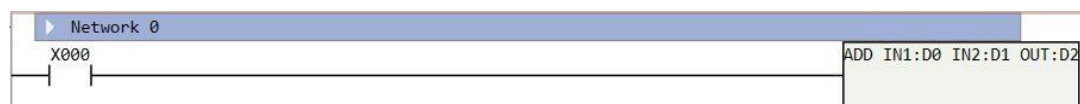
Network 000

LD X000

ADD D0 D1 D2// D0+D1=D2

POP

### Schemat drabinkowy:



## SUB/SUBD

1. Poniższa instrukcja odejmuje dane typu integer jednego wejścia od danych integer innego wejścia i zapisuje wynik (różnicę) w określonym rejestrze. Instrukcja **SUB** jest dla 16-bitowych liczb całkowitych ze znakiem, **SUBD** jest dla 32-bitowych liczb całkowitych ze znakiem.

2. Ta instrukcja może włączyć markery **M8169**, **M8170** i **M8171**.

## SUB

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Odjemna.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(IN2)	Odjemnik.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/TV/CV/AO/V/Z	-	16-bitowa liczba całkowita ze znakiem

## SUBD



Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Odjemna.	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(IN2)	Odjemnik.	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV	-	32-bitowa liczba całkowita ze znakiem

## Przykład

### Lista instrukcji:

Network 000

LD X000

SUB D0 D1 D2 //D0-D1=D2

POP

### Schemat drabinkowy:



## MUL/MULW/MULD

1. Instrukcje te mnożą dane typu integer jednego wejścia (czynnik 1) przez dane typu integer innego wejścia (czynnik 2) i przechowują iloczyn w określonym rejestrze. **MUL** jest używany do wejść w postaci 16-bitowych liczb całkowitych ze znakiem i wyjść w postaci 16-bitowych liczb całkowitych ze znakiem; **MULW** jest używany do wejść w postaci 16-bitowych liczb całkowitych ze znakiem i wyjść w postaci 32-bitowych liczb całkowitych ze znakiem; **MULD** jest używany do wejść w postaci 32-bitowych liczb całkowitych i wyjść w postaci 32-bitowych liczb całkowitych ze znakiem.

2. Ta instrukcja może włączyć markery **M8170** i **M8171**.

## MUL

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna



(IN1)	Składnik mnożenia	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(IN2)	Składnik mnożenia	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV	-	16-bitowa liczba całkowita ze znakiem

## MULW

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Składnik mnożenia	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(IN2)	Składnik mnożenia	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV	-	32-bitowa liczba całkowita ze znakiem

## MULD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Składnik mnożenia	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(IN2)	Składnik mnożenia	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV	-	32-bitowa liczba całkowita ze znakiem

## Przykład

Lista instrukcji:

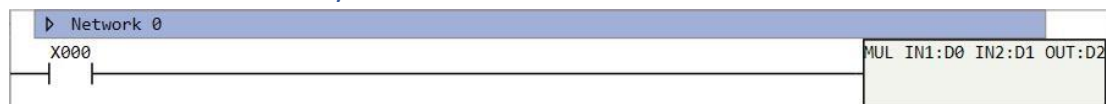
Network 000



LD X000

MUL D0 D1 D2  $//D0 \times D1 = D2$  D3POP

Schemat drabinkowy:



## DIV/DIVW/DIVD

- Instrukcje te dzielą dane liczbowe typu integer z jednego wejścia przez dane typu integer innego wejścia i przechowują wynik (różnicę) w określonym rejestrze. **DIV** jest używany do wejść w postaci 16-bitowych liczb całkowitych ze znakiem i wyjść w postaci 16-bitowych liczb całkowitych ze znakiem; **DIVW** jest używany do wejść w postaci 16-bitowych liczb całkowitych ze znakiem i wyjść w postaci 32-bitowych liczb całkowitych ze znakiem; **DIVD** jest używany do wejść w postaci 32-bitowych liczb całkowitych i wyjść w postaci 32-bitowych liczb całkowitych ze znakiem.
- Ta instrukcja może włączyć markery **M8170**, **M8171**.

## DIV

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Dzieln.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(IN2)	Dzielnik.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/TV/CV/AO/V/Z		16-bitowa liczba całkowita ze znakiem

## DIVW

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Dzieln.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem





(IN2)	Dzielnik.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV	-	32-bitowa liczba całkowita ze znakiem

## DIVD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN1)	Dzielnik.	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(IN2)	Dzielnik.	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV	-	32-bitowa liczba całkowita ze znakiem

## Przykład

### Lista instrukcji:

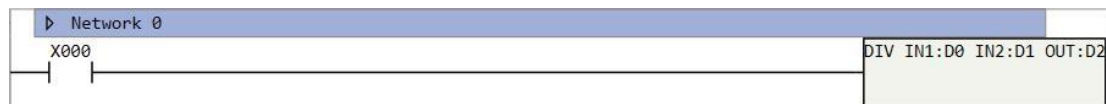
Network 000

LD X000

DIV D0 D1 D2 //D0÷D1=D2D3

POP

### Schemat drabinkowy:



## INC/INCD

- Poniższa instrukcja wykonuje operację inkrementacji (zwiększania) wejściowych danych typu integer i zapisuje wynik w określonym rejestrze. Instrukcja **INC** jest dla 16-bitowych liczb całkowitych ze znakiem, **INCD** jest dla 32-bitowych liczb całkowitych ze znakiem.
- Ta instrukcja może włączyć markery **M8169**, **M8170** i **M8171**.



## INC

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wejście do zwiększenia.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/TV/CV/AO/V/Z	-	16-bitowa liczba całkowita ze znakiem

## INCD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wejście do zwiększenia.	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV	-	32-bitowa liczba całkowita ze znakiem

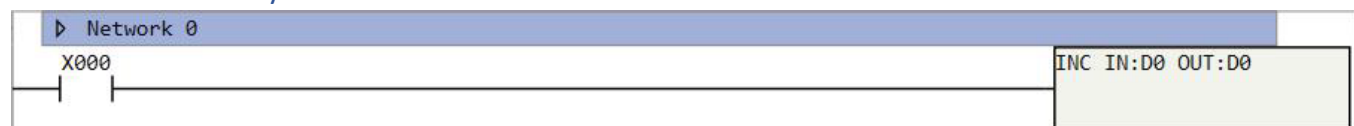
## Przykład

### Lista instrukcji:

Network 000

```
LD      X000
      INC  D0      D0 //D0 = D0 + 1
POP
```

### Schemat drabinkowy:



## DEC/DECD

- Poniższa instrukcja wykonuje operację dekrementacji (zmniejszania) z wejściowych danych typu integer i zapisuje wynik w określonym rejestrze. Instrukcja **DEC** jest dla 16-bitowych liczb całkowitych ze znakiem, **DECD** jest dla 32-bitowych liczb całkowitych ze znakiem.
- Ta instrukcja może włączyć markery **M8169**, **M8170** i **M8171**.



## DEC

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wejście do zmniejszenia.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/TV/CV/AO/V/Z	-	16-bitowa liczba całkowita ze znakiem

## DECD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wejście do zmniejszenia.	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV	-	32-bitowa liczba całkowita ze znakiem

## Przykład

### Lista instrukcji:

Network 000

```
LD      X000
      DEC  D0      D0 //D0 = D0 - 1
POP
```

### Schemat drabinkowy:



## FACT

1. Instrukcja ta oblicza silnię (!) z wejściowych danych typu integer i zapisuje wynik we wskazanym rejestrze.
2. Zakres wprowadzanych danych wynosi od 0 do 12.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
---------------------	------	---------	--------	------------



(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wejście do przeliczenia	K/H/D/TV/CV/AI/AO/V/Z	0~12	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D	-	16-bitowa liczba całkowita ze znakiem

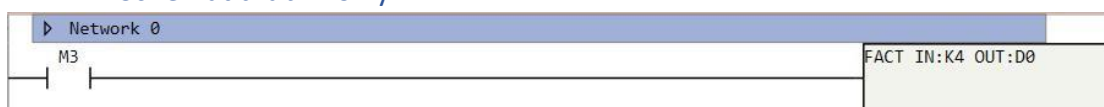
## Przykład

### Lista instrukcji:

Network 000

```
LD      X000
FACT K4  D0 //D0D1 = 4! = 24
POP
```

### Schemat drabinkowy:



## ABSF ABS/ABSD

Instrukcje te obliczają wartość bezwzględną danych wejściowych typu integer i przechowują wyniki w określonym rejestrze. ABS jest dla wejść w postaci 16-bitowych liczb całkowitych ze znakiem, ABSD jest dla wejść w postaci 32-bitowych liczb całkowitych ze znakiem.

### ABS

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wejście do obliczenia wartości bezwzględnej.	K/H/D/TV/CV/AI/AO/V/Z	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/TV/CV/AO/V/Z	-	16-bitowa liczba całkowita ze znakiem



## ABSD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(IN)	Wejście do obliczenia wartości bezwzględnej.	K/H/D/CV	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr wyjściowy przechowujący wynik.	D/CV		32-bitowa liczba całkowita ze znakiem

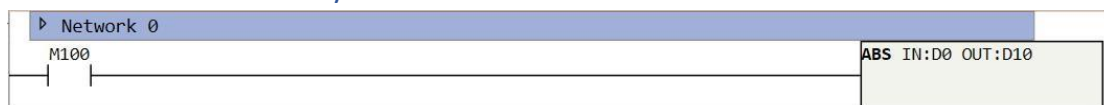
## Przykład

### Lista instrukcji:

Network 000

```
LD      M100
ABS     D0   D10 //ABS(D0D1) = D10D11
POP
```

### Schemat drabinkowy:





## INSTRUKCJE TIMERA

### Uwagi

1. Instrukcje w tym rozdziale zapisują odliczanie czasu w odpowiednim timerze (liczniku czasu) i ustawiają flagę bitową timera.
2. Istnieją 3 rodzaje timerów, które różnią się rozdzielczością. Rozdzielczość to minimalny przedział czasu, o który zwiększa się licznik czasu. Tabela poniżej pokazuje szczegóły.

Rozdzielczość	Maksymalny czas	Numer timera
100ms	6553.5s	T0~T199
10ms	655.35s	T200~T249
1ms	65.535s	T250~T255

W różnych licznikach czasu metody ich odświeżania są również różne.

#### Timer rozdzielczości 1ms:

Zegar jest odświeżany co 1 ms, w zależności od przerw systemowych i nie ma nic wspólnego z cyklem skanowania PLC ani przetwarzaniem programatora. Gdy okres skanowania jest dłuższy niż 1 ms, timer może być odświeżany wiele razy, a wartości timera (TV) i wyjścia timera (T) mogą być niespójne w jednym okresie skanowania.

#### Timer rozdzielczości 10ms:

Timer jest automatycznie odświeżany w każdym cyklu skanowania, dzięki czemu wartości timera (TV) i wyjścia timera (rejestr T) są spójne w cyklu skanowania.

#### Timer rozdzielczości 100ms:

Timer jest odświeżany tylko wtedy, gdy instrukcja timera jest przetwarzana, jeśli instrukcja nie jest przetwarzana lub przetwarzana więcej niż raz w jednym cyklu skanowania, taktowanie będzie niedokładne. Ten timer może być użyty **tylko raz** w programie.

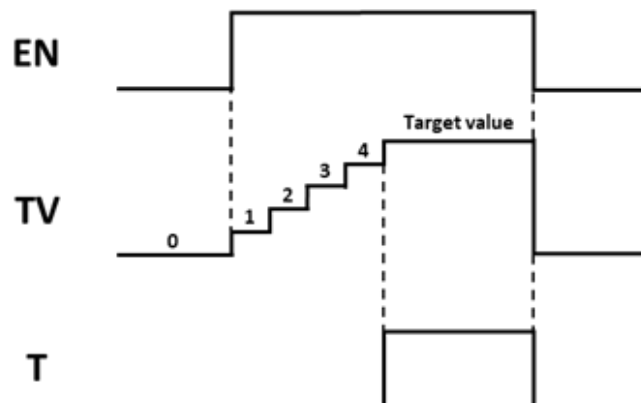
3. Różne typy instrukcji timera nie mogą współdzielić jednego timera. Dodatkowo zaleca się dodanie tylko jednej instrukcji timera w jednym programie.

## TON

1. Jest to instrukcja opóźnienia załączenia. Gdy polecenie jest włączane, timer rozpoczyna odliczanie. Gdy licznik osiągnie czas docelowy, flaga bitowa timera zostanie ustawiona w pozycji 1/ON; gdy instrukcja zostanie



wyłączona, licznik timera zostanie **wyzerowany**, a styk timera zostanie ustawiony w pozycji 0/OFF. Rysunek poniżej zawiera szczegóły.



2. Użytkownicy mogą użyć tego polecenia, aby liczyć czas dla pojedynczego obiektu.
3. Zresetowanie styku timera może również wyzerować odpowiedni licznik timera.

#### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(T)	Wybrany timer	TV		16-bitowa liczba całkowita bez znaku
(SV)	Czas odliczania timera.	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku

#### Przykład:

##### Lista instrukcji:

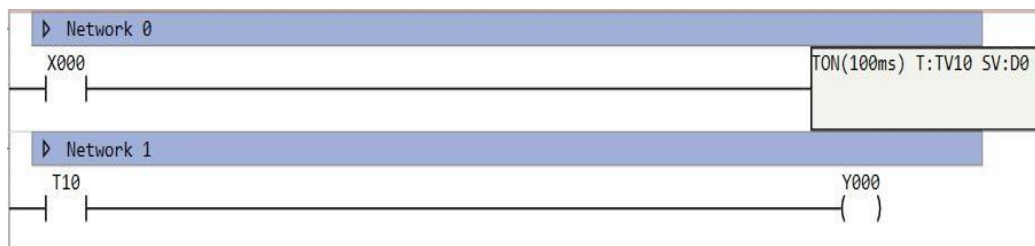
##### Network 000

```
LD      X100
TON TV10 D0 // gdy licznik timera TV10 osiągnie docelowy czas D0, T10 jest ustawiony na ON
```

##### Network 000

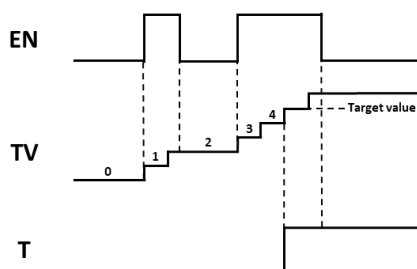
```
LD      T10
OUT Y000 // gdy bit Y000 jest włączony, działanie timera jest wyłączona.
POP
```

#### Schemat drabinkowy:



## TONR

1. Ta instrukcja jest instrukcją timera z opóźnieniem włączenia. Gdy polecenie jest włączane, timer uruchamia (kontynuuje) odliczanie. Styk wyjściowy timera zostanie ustawiony w pozycji 1/ON, gdy licznik timera osiągnie czas docelowy; Gdy instrukcja jest wyłączona, licznik czasu i odpowiadający mu styk wyjściowy będą **podtrzymywane**. Szczegóły przedstawiono na rysunku



2. Użytkownicy mogą użyć tego polecenia, aby liczyć czas dla kilku obiektów.
3. Zresetowanie styku timera może również wyzerować odpowiedni licznik timera.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(T)	Wybrany timer	TV		16-bitowa liczba całkowita bez znaku
(SV)	Czas odliczania timera.	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku

### Przykład:

#### Lista instrukcji:

#### Network 000

LD X000

TONR TV251 K100 // gdy licznik timera TV251 osiągnie docelowy czas K100, T251 jest ustawiony w pozycji ON

#### Network 000





LD T251

OUT Y000 // gdy bit Y000 jest włączony, polecenie odliczania timera jest wyłączone.

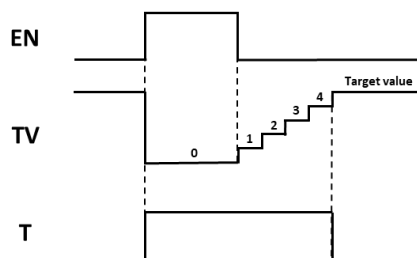
POP

Schemat drabinkowy:



## TOF

1. Ta instrukcja jest instrukcją timera opóźnienia wyłączenia. Gdy polecenie jest wyłączone, timer uruchamia się (kontynuuje) odliczanie. Styk wyjściowy timera zostanie ustawiony na OFF, gdy licznik timera osiągnie czas docelowy; Kiedy polecenie jest włączane, licznik timera zostanie wyzerowany, a styk wyjściowy timera zostanie ustawiony na ON. Szczegóły na rysunku poniżej.



2. Użytkownicy mogą użyć tego polecenia, aby liczyć czas dla jednego obiektu.
3. Zresetowanie styku timera może również wyzerować odpowiedni licznik timera.

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie.		0/1	Wartość logiczna
(T)	Wybrany timer	TV		16-bitowa liczba całkowita bez znaku
(SV)	Czas odliczania timera.	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku

Przykład:

Lista instrukcji:



### Network 000

LD X000

TOF TV251 K100 // gdy licznik timera TV251 osiągnie docelowy czas K100, T251 jest ustawiony na OFF

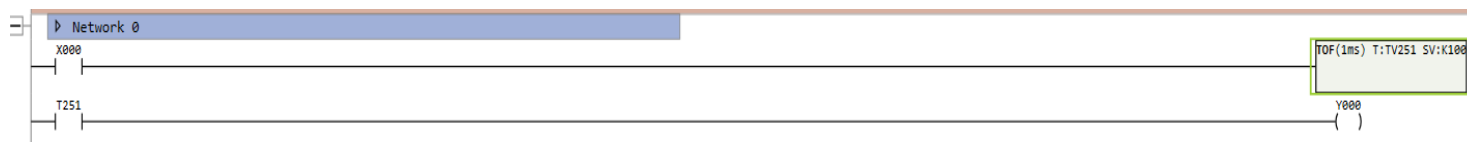
### Network 000

LD T251

OUT Y000 // gdy Y000 jest wyłączony, polecenie odliczania timera jest włączone.

POP

Schemat drabinkowy:





## INSTRUKCJE LICZNIKA

1. Instrukcje w tym rozdziale rejestrują dane z liczników zmiennych zboczy sygnału i wyjścia bitowe odpowiednich liczników, gdy zliczanie osiągnie liczbę docelową.
2. Istnieją trzy typy liczników, a ich bity danych i zastosowania są różne. Szczegóły przedstawiono w tabeli poniżej.

Typ licznika	Zakres zliczania	Numer licznika
16-bit	-32768~32767	C0~C199
32-bit	-2147483648~2147483647	C200~C249
32-bit (high-speed)	-2147483648~2147483647	C250~C255

Szybki licznik (*high-speed counter*) służy do zliczania szybkich impulsów.

3. Zaleca się dodanie jednego typu instrukcji licznika w jednym programie.

## CTU

1. Ta instrukcja zliczania w górę (liczba na liczniku wzrasta) przy każdym narastającym zboczu wejścia. Kiedy liczba osiągnie ustaloną liczbę docelową, odpowiedni bit wyjściowy zostanie ustawiony w pozycji wysokiej (ON).
2. Zresetowanie styku licznika spowoduje również wyczyszczenie liczby na liczniku.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(IN)	Wejście wyzwalające zliczanie.		0/1	Wartość logiczna
(T)	Wybrany licznik	CV		<b>■ C0~C199</b> 16-bitowa liczba całkowita ze znakiem <b>■ C200~C255</b> 32-bitowa liczba całkowita ze znakiem
(SV)	Docelowa wartość zliczania.	K/H/D	<b>■ C0~C199</b> -32768~32767 <b>■ C200~C255</b> -2147483648~2147483647	<b>■ C0~C199</b> 16-bitowa liczba całkowita ze znakiem <b>■ C200~C255</b> 32-bitowa liczba całkowita ze znakiem



### Przykład:

#### Lista instrukcji:

##### Network 000

LD X000

CTU CV0 D0 //zliczaj w górę, gdy licznik CV0 osiągnie liczbę docelową D0, C0 jest  
załączane

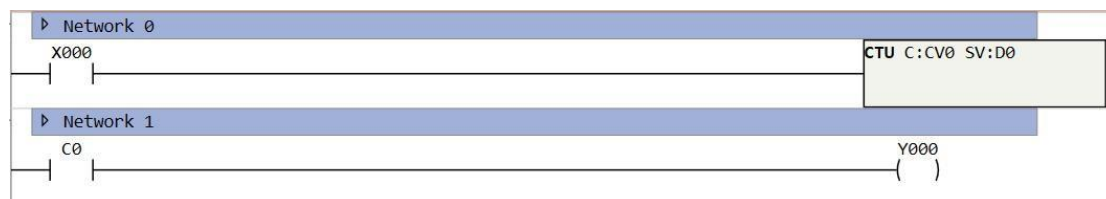
##### Network 000

LD C0

OUT Y000

POP

#### Schemat drabinkowy:



### CTD

1. Ta instrukcja zliczania w dół (liczba na liczniku maleje) przy każdym narastającym zboczu wejścia. Kiedy liczba osiągnie ustaloną liczbę docelową, odpowiedni bit wyjściowy zostanie ustawiony w pozycji wysokiej (ON).
2. Zresetowanie styku licznika spowoduje również wyczyszczenie liczby na liczniku.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(IN)	Wejście wyzwalające zliczanie.		0/1	Wartość logiczna
(T)	Wybrany licznik	CV		<b>■ C0~C199</b> 16-bitowa liczba całkowita ze znakiem <b>■ C200~C255</b> 32-bitowa liczba całkowita ze znakiem



(SV)	Docelowa wartość zliczania.	K/H/D	<b>■ C0~C199</b> -32768~32767 <b>■ C200~C255</b> -2147483648~2147483647	<b>■ C0~C199</b> 16-bitowa liczba całkowita ze znakiem <b>■ C200~C255</b> 32-bitowa liczba całkowita ze znakiem
------	-----------------------------	-------	--	---

### Przykład:

#### Lista instrukcji:

#### Network 000

LD X000

CTD CV0 D0 //zliczaj w dół, gdy licznik CV0 osiągnie liczbę docelową D0, C0 jest załączane

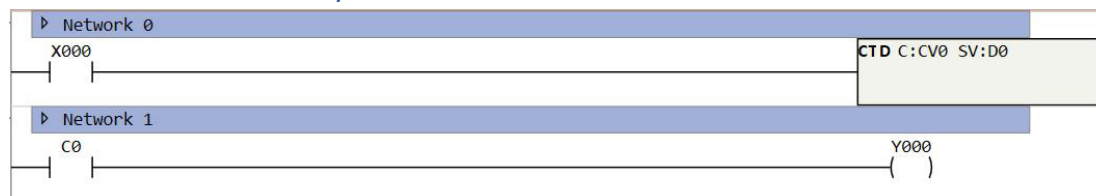
#### Network 000

LD C0

OUT Y000

POP

#### Schemat drabinkowy:



## CTUD

- Poniższa instrukcja zlicza w górę lub w dół przy każdym narastającym zboczach wejścia. Kierunek zliczania zależy od porównania między liczbą docelową a bieżącą liczbą. Kiedy liczba docelowa jest większa niż bieżąca liczba, licznik zlicza w górę, gdy liczba docelowa jest mniejsza niż bieżąca liczba, licznik zlicza w dół; gdy licznik osiągnie liczbę docelową, odpowiadające wyjście zostanie ustawione na ON.
- Zresetowanie styku licznika spowoduje również wyczyszczenie licznika.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(IN)	Wejście wyzwalające zliczanie.		0/1	Wartość logiczna



(T)	Wybrany licznik	CV		<b>■ C0~C199</b> 16-bitowa liczba całkowita ze znakiem <b>■ C200~C255</b> 32-bitowa liczba całkowita ze znakiem
(SV)	Docelowa wartość zliczania.	K/H/D	<b>■ C0~C199</b> -32768~32767 <b>■ C200~C255</b> -2147483648~2147483647	<b>■ C0~C199</b> 16-bitowa liczba całkowita ze znakiem <b>■ C200~C255</b> 32-bitowa liczba całkowita ze znakiem

### Przykład:

#### Lista instrukcji:

#### Network 000

LD X000

CTDUD CV0 D0 // Porównaj liczbę docelową z aktualną liczbą, licznik zliczy w górę lub w dół, gdy licznik CV0 osiągnie docelową liczbę D0, C0 włącza się

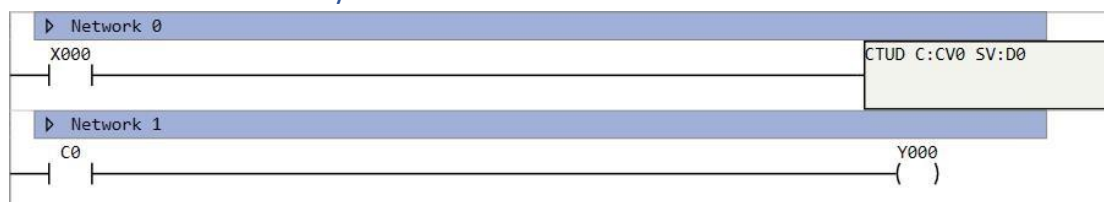
#### Network 000

LD C0

OUT Y000

POP

#### Schemat drabinkowy:





## INSTRUKCJE REGULACJI PROGRAMU

1. Instrukcje w tym rozdziale regulują etapy wykonywania programu sterującego. Zwykle programy są wykonywane sekwencyjnie a przy użyciu tych instrukcji wykonywanie programu może implementować funkcje, takie jak pętle, skoki, wywołania i podobne.
2. Projekt programu PLC można podzielić na trzy rodzaje: program główny, podprogram i blok funkcyjny. Program główny jest wejściem do wykonania - rozpoczęcie wykonywania zaczyna się od programu głównego; program podprogramu jest gałęzią wykonania - użytkownik może użyć instrukcji sterujących programem, aby go wywołać i wrócić do programu głównego; blok funkcyjny jest instrukcją zdefiniowaną przez użytkownika, którą użytkownik może napisać w języku C lub może zostać wywołana w programie jak inne instrukcje.
3. Rejestr S jest rejestrem bitowym stanu systemu, który oznacza stan systemu i działa w instrukcjach statusowych (STL/STLE/ST i IST).

## FOR~NEXT

1. Instrukcja **FOR** musi być używana w parze z instrukcją **NEXT**. Kiedy funkcja **FOR** jest włączona, segmenty programu między **FOR** i **NEXT** tworzą program pętli, który zostanie wykonany dla określonej ilości cykli, następnie po załączeniu się instrukcji **NEXT** program główny będzie kontynuowany. Gdy funkcja **FOR** jest wyłączona, część programu znajdująca się między **FOR** i **NEXT** zostanie pominięta.
2. Ilość cykli można modyfikować w trakcie działania programu. Gdy program ponownie wejdzie w pętlę, wykona instrukcje pętli dla zmodyfikowanej ilości cykli.
3. Nie zaleca się dodawania timera w programie pętli.
4. Maksymalna liczba pętli zagnieżdżonych wynosi 8.
5. Użytkownik może użyć instrukcji **CJ** lub **BREAK**, aby wyskoczyć z cyklu.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie		0/1	Wartość logiczna
(CNT)	Ilość cykli	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku

Przykład:

Lista instrukcji:



### Network 000

LDP X000

FOR K10 // Ilość pętli jest ustawiona na 10

### Network 000

LD X001

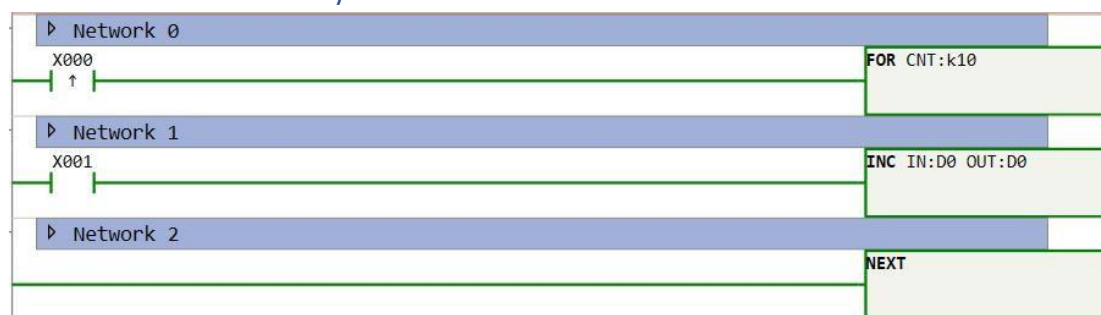
INC D0 D0 //D0 wzrasta w każdym cyklu

### Network 000

NEXT //koniec pętli

POP

### Schemat drabinkowy:



## CJ~LBL

1. Instrukcja CJ musi być używana w parze z instrukcją LBL. Gdy polecenie CJ jest włączone, program przeskoczy do określonej etykiety LBL; gdy polecenie CJ jest wyłączane, program będzie wykonywany w normalnej kolejności.
2. Instrukcja CJ i odpowiadająca jej etykieta LBL muszą znajdować się w tym samym programie (program główny lub podprogram). Nie jest możliwe przeskoczenie z programu głównego do podprogramu, ani wyskoczenie z podprogramu do programu głównego.

## CJ

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie		0/1	Wartość logiczna
(LBL)	Numer etykiety docelowej	K/H/	0~65535	16-bitowa liczba całkowita bez znaku

## LBL

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
---------------------	------	---------	--------	------------





(L)	Numer etykiety	K/H/	0~65535	16-bitowa liczba całkowita bez znaku
-----	----------------	------	---------	--------------------------------------

### Przykład:

#### Lista instrukcji:

##### Network 000

LD X000

CJ K0 //jeśli X000 jest w pozycji ON, przeskocz do LBL K0

##### Network 000

LD X001

ADD D0 D1 D2 // jeśli X000 jest w pozycji OFF, D2=D0+D1 gdy X001 jest w pozycji ON

##### Network 000

LBL K0 // Ustaw numer etykiety jako K0

#### Schemat drabinkowy:



## CALL

1. Gdy ta instrukcja jest włączona, wykonanie programu przeskoczy do początku ustawionego podprogramu ; Gdy ta instrukcja jest wyłączona, instrukcja podprogramu zostanie pominięta i kontynuowany będzie następny program.
2. Użytkownik może najpierw utworzyć podprogram, a także może dodać tę instrukcję, przenosząc podprogram z interfejsu projektu do interfejsu schematu drabinkowego.
3. Maksymalna liczba pętli zagnieżdżonych wynosi 8.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie		0/1	Wartość logiczna
(P)	Podprogram do wywołania.	-	-	-



### Przykład:

#### Lista instrukcji

Network 000

LD X000

CALL Sub1 //wywołaj podprogram "Sub1"

POP

#### Schemat drabinkowy:



## CALLM

1. Gdy to polecenie jest włączane, program wywoła określony blok funkcyjny; Gdy to polecenie jest wyłączane, ta instrukcja zostanie pominięta i kontynuowany będzie następny program.
2. Ta instrukcja powinna być wyzwalana sygnałem zbocza.
3. Szczegóły dotyczące bloku funkcyjnego znajdują się w części „Budowanie bloku funkcyjnego” w [dodatku](#).

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie		0/1	Wartość logiczna
(F)	Blok funkcyjny do wywołania.	-	-	-
(P1)	Parametr funkcji.	-	-	Dowolny
(P2)	Parametr funkcji.	-	-	Dowolny
(P3)	Parametr funkcji.	-	-	Dowolny
(P4)	Parametr funkcji.	-	-	Dowolny

### Przykład:

#### Funkcja sortowania:

Function block:

// Zainicjuj dane

void Data\_Init(WORD data, WORD size)

{

short i = 0 ;

for ( i = 0 ; i < \*size ; i++)

data[i] = (i ^ 0x4321); //wykonaj instrukcję XOR na każdym z danych 0x4321



```

}
// sortuj dane od najmniejszych do największych
void Data_Sort(WORD data, WORD size)
{
short i = 0;
short j = 0;
short temp = 0;
for ( i = 0 ; i < *size ; i++)
{
temp = data[i];
for (j = i ; j > 0 && data[j-1] > temp ; j--)
data[j] = data[j-1]; // kontynuuj działania w pętli
data[j] = temp;
}
}

```

#### Lista instrukcji

LDP M0 // wywołaj funkcję Data\_Init(zainicjuj dane) raz, gdy M0 jest włączone, D[0]=0x4321, D[1]=0x4320, D[2]=0x4323, D[3]=0x4322, D[4]=0x4325, D[5]=0x4324, D[6]=0x4327

MOV K7 D100 //ustaw długość danych D100 na 7

CALLM Data\_Init D0 D100 // wywołaj funkcję Data\_Init(zainicjuj dane), dane=D0, rozmiar=D100

POP

LDP M1 //wywołaj funkcję Data\_Sort (sortuj dane), gdy M1 jest włączone, D[0]=0x4320, D[1]=0x4321, D[2]=0x4322, D[3]=0x4323, D[4]=0x4324, D[5]=0x4325, D[6]=0x4327

CALLM Data\_Sort D0 D100 // wywołaj Data\_Sort(sortuj dane), dane=D0, rozmiar=D100

POP

LDW< D1 D2

AW< D2 D3

AW< D3 D4

AW< D4 D5

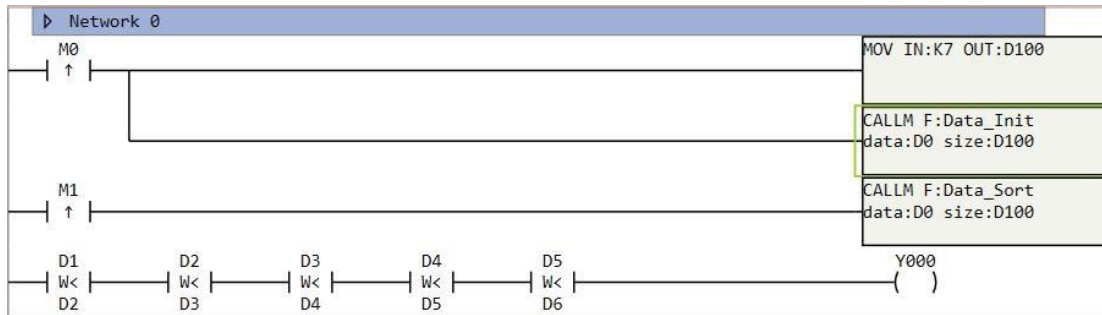
AW< D5 D6

OUT Y000 //jeżeli D1<D2<D3<D4<D5<D6, Y000 jest włączone

POP



### Schemat drabinkowy:



### Funkcja obliczania odchylenia standardowego

Blok funkcyjny:

// oblicz średnią

**void Average**(FLOAT in, WORD size, FLOAT out)

```
{
*out = 0.0;
short i = 0;
for (; i < *size; i++)
*out += in[i];
*out /= *size;
}
```

// oblicz odchylenie standardowe

[Array(in, size)]

**void StandardDeviation**(FLOAT in, WORD size, FLOAT out)

```
{
float ave = 0.0;
Average(in, size, &ave);
short i = 0;
*out = 0.0;
for (; i < *size; i++)
*out += (in[i] - ave) * (in[i] - ave);
*out /= *size;
*out = sqrt(*out);
}
```

### Lista instrukcji

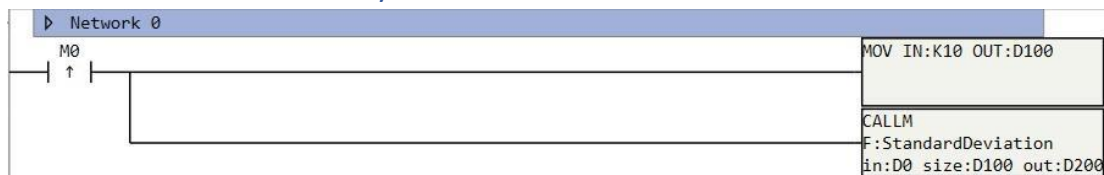
LDP M0

MOV K10 D100 //ustaw długość danych D100 jako 10

CALLM StandardDeviation D0 D100 D200 // wywołaj funkcję StandardDeviation (odchylenie standardowe), wejście=D0, rozmiar=D100, wyjście=D200



Schemat drabinkowy:



## STL/STLE/ST

1. Te instrukcje są instrukcjami programowania krokowego, **STL** rozpoczyna program krokowy, **STLE** kończy program krokowy, **ST** włącza. Dodatkowo zresetowanie cewki ustawiającej (**S**) za pomocą instrukcji **RST** może wyłączyć program krokowy.
2. Gdy użyjesz **RST** do wyłączenia programu krokowego, wszystkie wyjścia (np. styk wyjściowy, przekaźnik czasowy, licznik, wyjście impulsowe) w podprogramie zostaną zresetowane lub wyczyszczone.
3. **STL** musi być używany razem ze **STLE**, a podprogram między **STL** i **STLE** nie może być zagnieżdżony.
4. Ustawienie odpowiedniego przekaźnika **S** za pomocą instrukcji **SET** może również włączać program krokowy. Różnica polega na tym, że przy użyciu **SET**, ustawiony przekaźnik **S** zostanie automatycznie zresetowany w następnym okresie skanowania; Gdy używasz **ST**, ustawiony przekaźnik **S** będzie podtrzymywany.

## STL

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie		0/1	Wartość logiczna
(L)	Adres cewki ustawiającej program krokowy	S	0/1	Wartość logiczna

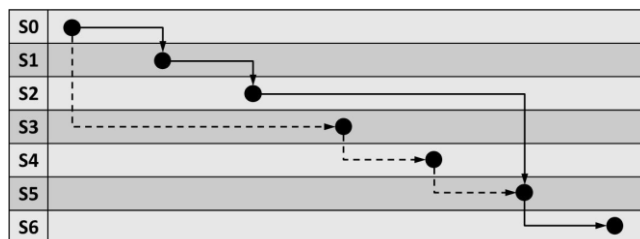
## ST

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie		0/1	Wartość logiczna
(S)	Adres docelowego styku programu krokowego do włączenia	S	0/1	Wartość logiczna

Przykłady:

### Wybór gałęzi.

W tym przykładzie użytkownik może wybrać jedną lub obie gałęzie programu do wykonania. Poniżej zamieszczono szczegóły.



## Lista instrukcji:

## Network 000

```
LD    X000
SET   S0      K1 //przejdź do programu krokowego S0 gdy X000 jest włączone
POP
```

## Network 001

```
STL   S0 //utwórz program krokowy S0
LD    M8151
OUT   Y000
POP
LD    X000
SET   S1      K1 //przejdź do programu krokowego S1 gdy X000 jest włączone
POP
LD    X003
SET   S3      K1 //przejdź do programu krokowego S3 gdy X003 jest włączone
POP
STLE  //zakończ program krokowy S0
```

## Network 002

```
STL   S1 //utwórz program krokowy S1
LD    M8151
OUT   Y001
POP
LD    X001
SET   S2      K1 //przejdź do programu krokowego S2 gdy włączony jest X001
POP
STLE  //zakończ program krokowy S1
```

## Network 003

```
STL   S2 //utwórz program krokowy S2
LD    M8151
OUT   Y002
POP
LD    X002
SET   S5      K1 //przejdź do programu krokowego S5 gdy włączony jest X002
POP
STLE  //zakończ program krokowy S2
```

## Network 004



```
STL    S3 //utwórz program krokowy S3
LD      M8151
OUT     Y003
        POP
LD      X004
SET     S4      K1 //przejdź do programu krokowego S4 gdy włączony jest X004
        POP
        STLE //zakończ program krokowy S3
```

#### Network 005

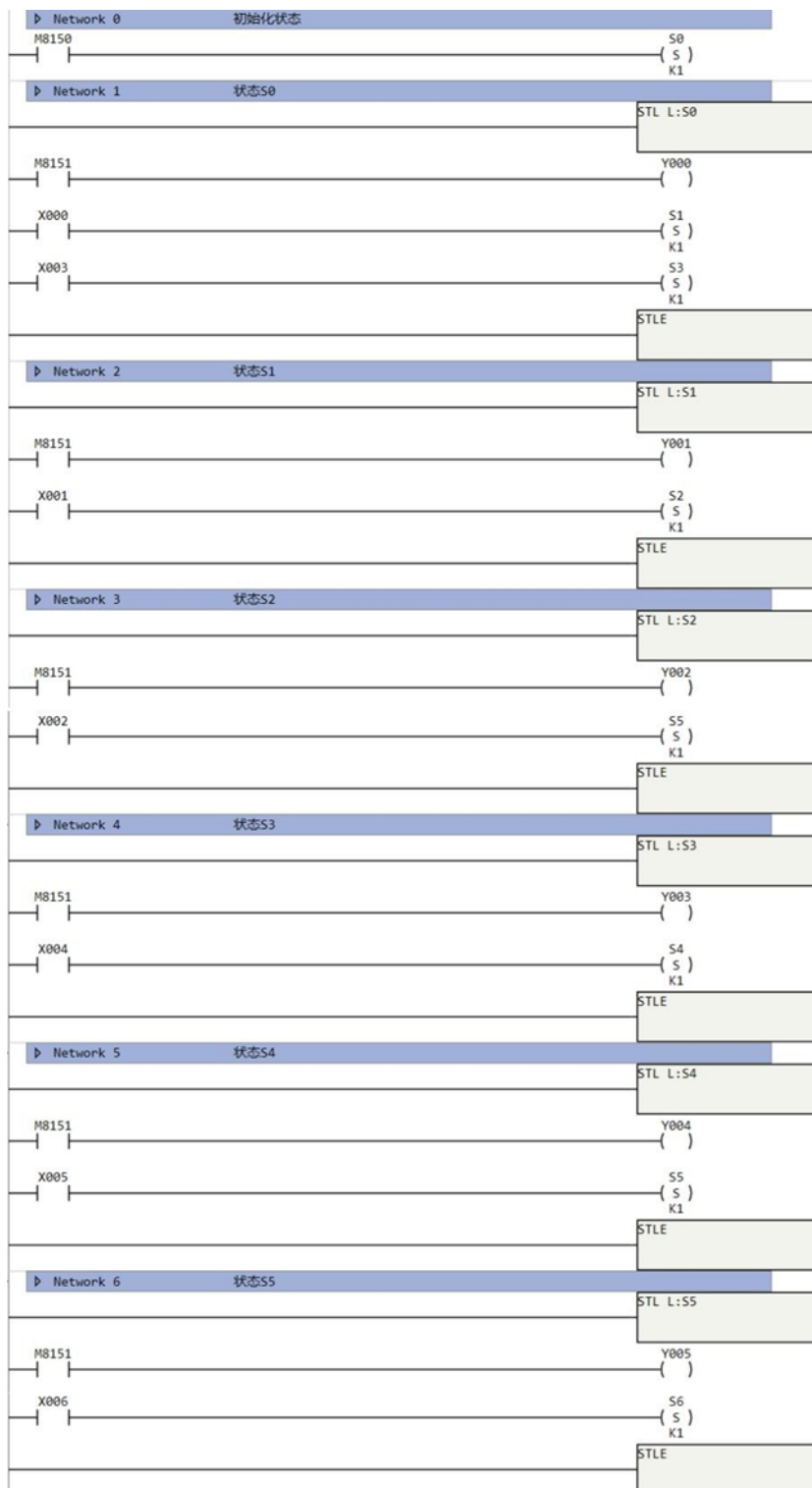
```
STL    S4 //utwórz program krokowy S4
LD      M8151
OUT     Y004
        POP
LD      X005
SET     S5      K1 //przejdź do programu krokowego S5 gdy włączony jest X005
        POP
        STLE //zakończ program krokowy S4
```

#### Network 006

```
STL    S5 //utwórz program krokowy S5
LD      M8151
OUT     Y005
        POP
LD      X006
SET     S6      K1 //przejdź do programu krokowego S6 gdy włączony jest X006
        POP
        STLE //zakończ program krokowy S5
        POP
```



## Schemat drabinkowy:

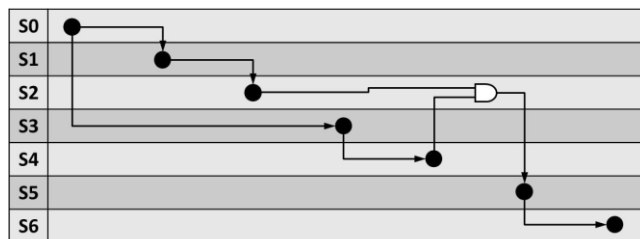






### Wybór gałęzi.

W tym przykładzie dwie gałęzie drabiny są równoległe, obie są wykonywane. Poniżej zamieszczono szczegóły.



### Lista instrukcji:

#### Network 000

```
LD    X000
SET   S0    K1 //przejdź do programu krokowego S0 gdy włączane jest X000
POP
```

#### Network 001

```
STL   S0 //utwórz program krokowy S0
LD    M8151
OUT   Y000
POP
LD    X000
SET   S1    K1 //przejdź do programu krokowego S1 gdy włączane jest X000
SET   S3    K1 //przejdź do programu krokowego S3 gdy włączane jest X003
POP
```

```
STLE  //zakończ program krokowy S0
```

#### Network 002

```
STL   S1 //utwórz program krokowy S1
LD    M8151
OUT   Y001
POP
LD    X001
SET   S2    K1 //przejdź do programu krokowego S2 gdy włączane jest X001
POP
```

```
STLE  //zakończ program krokowy S1
```

#### Network 003

```
STL   S2 //utwórz program krokowy S2
LD    M8151
OUT   Y002
POP
```

```
STLE  //zakończ program krokowy S2
```

#### Network 004

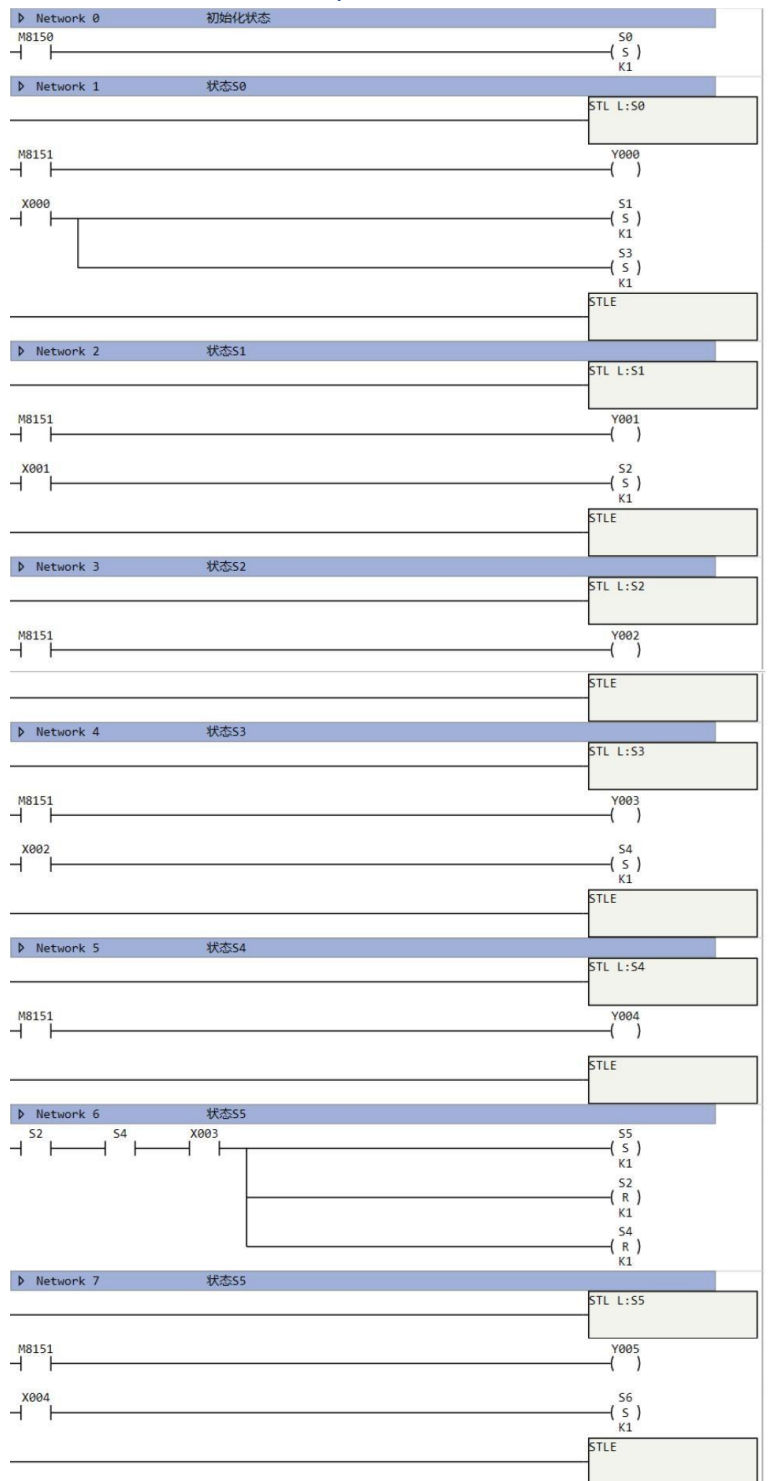
```
STL   S3 //utwórz program krokowy S3
LD    M8151
OUT   Y003
POP
LD    X002
```



```
SET    S4    K1 //przejdź do programu krokowego S4 gdy włączane jest X002
      POP
STLE   //zakończ program krokowy S3
Network 005
STL    S4 //utwórz program krokowy S4
LD     M8151
OUT    Y004
      POP
STLE   //zakończ program krokowy S4
Network 006
LD     S2
AND    S4
AND    X003
SET    S5    K1 //przejdź do programu krokowego S5 gdy włączane jest X003
RST    S2    K1 //S2 musi być resetowane ręcznie
RST    S4    K1 //S4 musi być resetowane ręcznie
      POP
101
Network 007
STL    S5 //utwórz program krokowy S5
LD     M8151
OUT    Y005
      POP
LD     X004
SET    S6    K1 //przejdź do programu krokowego S6 gdy włączane jest X004
      POP
STLE   //zakończ program krokowy S5
```



### Schemat drabinkowy:



## RET

1. Ta instrukcja jest używana w podprogramie i nie jest możliwe użycie jej w programie głównym.



2. Gdy to polecenie jest włączane, podprogram nie będzie już wykonywany, ale powróci do programu głównego i będzie kontynuował wykonywanie od miejsca wywołania podprogramu.

#### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie		0/1	Wartość logiczna

#### Przykład:

##### Program główny:

##### Tabela instrukcji:

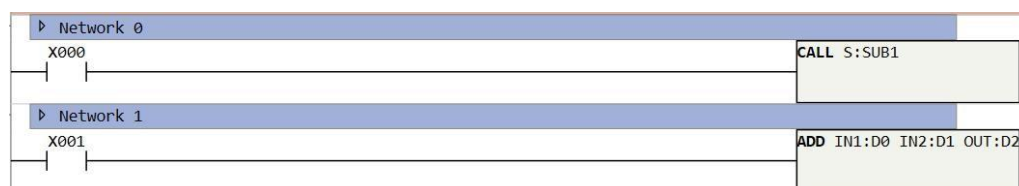
##### Network 000

```
LD X000
CALL SUB1 //wywołaj podprogram "SUB1"
```

##### Network 001

```
LD X001
ADD D0 D1 D2 //D2=D0+D1 gdy włączany jest X001
POP
```

##### Schemat drabinkowy:



##### Podprogram:

##### Tabela instrukcji:

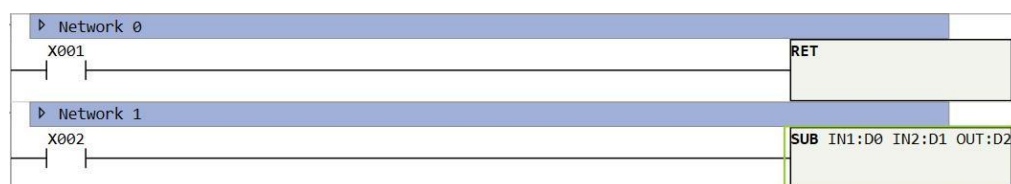
##### Network 000

```
LD X001
RET //powrót do programu głównego przy załączeniu X001
```

##### Network 001

```
LD X002
SUB D0 D1 D2 //D2=D0-D1 gdy X001 jest wyłączany i włączany jest X002
```

##### Schemat drabinkowy:





## BREAK

Włączona instrukcja **BREAK**, jeśli znajduje się w części pętli (**FOR** do **NEXT**), wymusi wyskoczenie z tej pętli i będzie kontynuowany program po **NEXT**; W programie krokowym (**STL** do **STLE**), wyskoczy z tego programu i będzie kontynuował program po **STLE**. Jeśli znajduje się zarówno w części z pętlą, jak i w programie krokowym, wyskoczy z części z pętlą.

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie		0/1	Wartość logiczna

Przykłady:

## FOR do NEXT

Network 000

LD X000

FOR K100 //rozpocznij pętlę, zapętlaj 100 razy

Network 001

LD X001

INC D0 D0 // D0 wzrasta w każdej pętli

Network 002

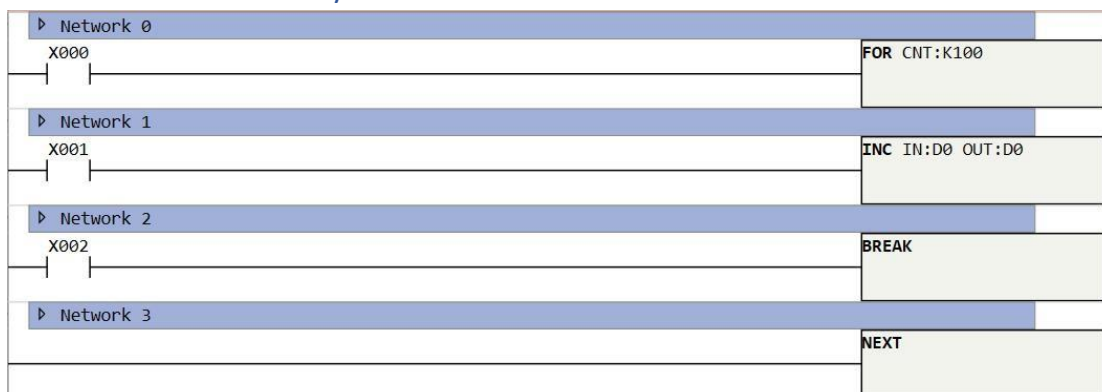
LD X002

BREAK // wyskocz z pętli, gdy X002 jest włączony

Network 003

NEXT //zakończ pętlę

Schemat drabinkowy:



## STL do STLE

Network 000

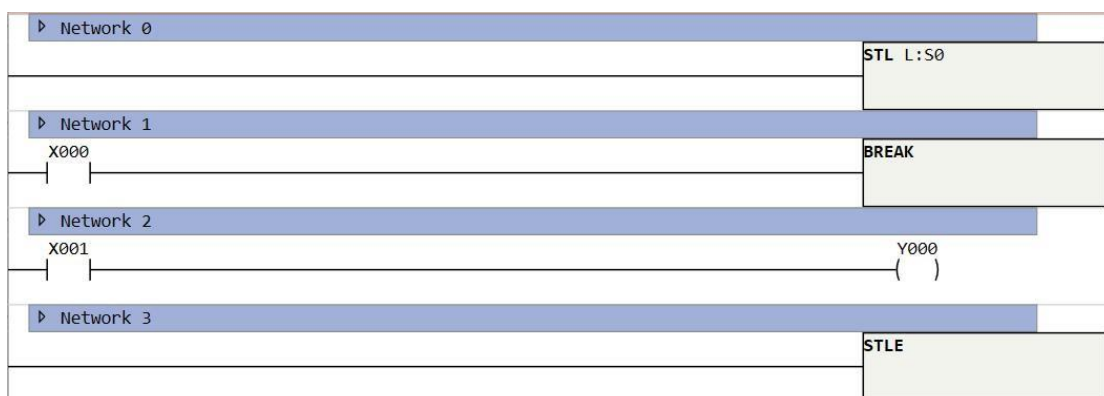
STL S0 // utwórz program krokowy

Network 001



LD X000  
 BREAK //wyskocz z programu krokowego, gdy X000 jest włączany  
 Network 002  
 LD X001  
 OUT Y0 //kontynuuj tę instrukcję (jeśli nie będzie wyskoku)  
 Network 003  
 STLE //zakończ program krokowy

Schemat drabinkowy:



## IST

1. Ta instrukcja automatycznie kontroluje stan początkowy i rejestry specjalne w programie krokowym.
2. Kolejnych 8 bitów do sterowania programem krokowym rozpoczyna się od wprowadzenia wejściowego bitu (IN). Bit ten działa przy narastającym zboczu, w efekcie czego następują: przejście do trybu ręcznego (IN), przejście do trybu powrotu do zera (IN+1), przejście do trybu pojedynczego kroku (IN+2), przełączenie na tryb pojedynczego cyklu (IN+3), przełączenie na tryb automatyczny (IN+4), start bez powrotu do zera (IN+5), start (IN+6) i zatrzymanie (IN+7).
3. Ta instrukcja jest powiązana z niektórymi rejestrami specjalnymi (rejestrami danych i bitami), (szczegóły w tabeli). Nie używaj tych rejestrów jak zwykłych w programie.

Rejestr		Opis	Dostęp do odczytu (O) i zapisu (Z)
D	D8048	Zapisz bieżący tryb programu drabinkowego (0 oznacza niewłączony, 1 tryb ręczny, 2 tryb powrotu do punktu początkowego, 3 tryb pojedynczego kroku, 4 tryb pojedynczego cyklu, 5 tryb automatyczny).	O
M	M8020	Transfer STL wyłączony.	O/Z
	M8021	Rozpoczęcie się transferu.	O/Z



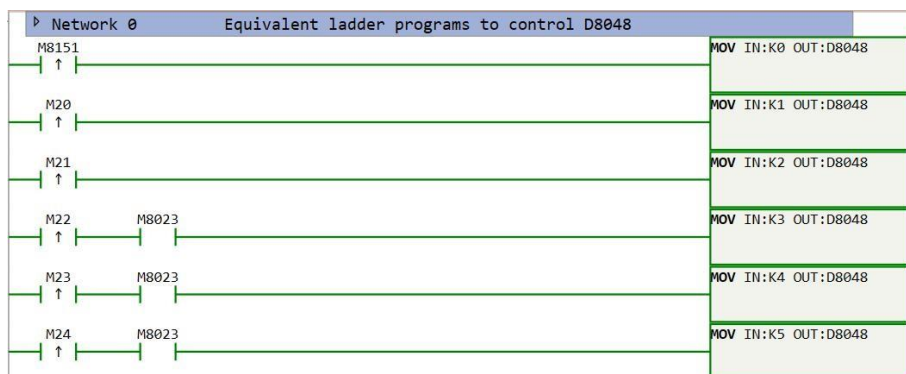
	<b>M8022</b>	Impuls (sygnał) startu IST.	O
	<b>M8023</b>	Powrót do zera zakończony.	O
	<b>M8024</b>	Stan zerowy.	O/Z
	<b>M8025</b>	Wszystkie resety wyjść są wyłączone.	O/Z
	<b>M8026</b>	STL włączony.	O
	<b>M8027</b>	Włącz monitorowanie STL.	O/Z
<b>S</b>	S0	Cewka ustawiająca trybu ręcznego.	O/Z
	S1	Cewka ustawiająca trybu powrotu do stanu początkowego.	O/Z
	S2	Cewka ustawiająca trybów automatycznych (w tym tryb jednego kroku, tryb jednego cyklu, tryb automatyczny).	O/Z

## Powiązane rejestry

Szczegóły dotyczące wyjść specjalnych i wyjść ustawiających (S0 do S2), automatycznie kontrolowanych przez instrukcję **IST**, pokazane są w przykładowym schemacie drabinkowym poniżej, główny bit sterujący jest ustawiony jako M20 - nie może być zaprogramowany.

## D8048

**D8048** jest powiązany z bieżącym trybem działania programu krokowego (w tym trybem ręcznym, trybem powrotu do punktu początkowego, trybem pojedynczego kroku, trybem pojedynczego cyklu i trybem automatycznym). Jego wartość początkowa wynosi 0 i zmienia się przy narastającym zboczu określonych sygnałów. Tryb pojedynczego kroku, tryb pojedynczego cyklu i tryb automatyczny są sklasyfikowane jako tryby automatyczne, i te 3 tryby są przesyłane tylko wtedy, gdy **M8023** jest włączony (powrót do punktu początkowego został zakończony). Równoznaczny program drabinkowy zamieszczony został na rysunku poniżej.



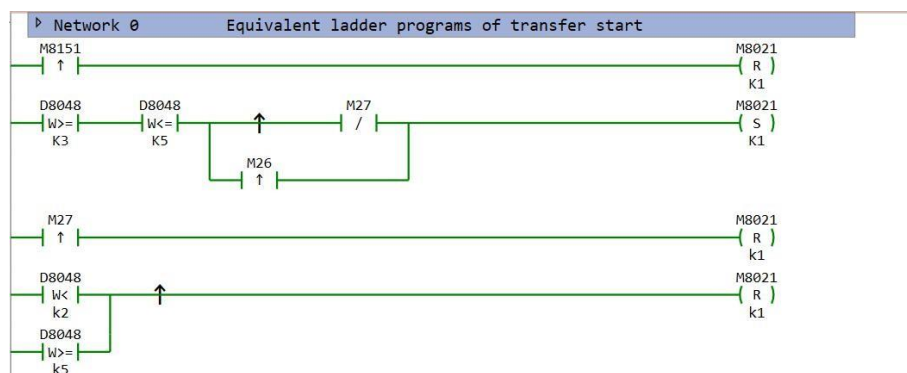
## M8020



Kiedy **M8020** jest włączony, przesyłanie każdego stanu jest wyłączone.

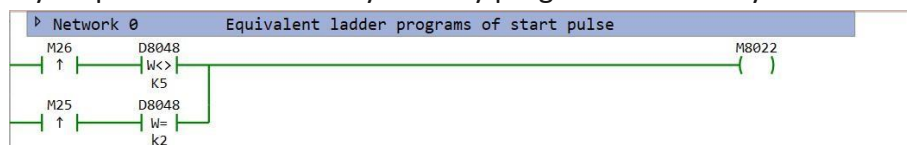
## M8021

**M8021** działa tylko w trybach automatycznych; gdy włącza się program krokowy (zbocze narastające IN+5), marker ten włącza się; gdy program krokowy zatrzymuje się (zbocze narastające IN+5), marker ten jest wyłączany; gdy tryb zostanie przełączony na tryb ręczny lub tryb powrotu do zera, marker zostanie wyłączony. Poniżej rysunek przykładowego programu w schemacie drabinkowym.



## M8022

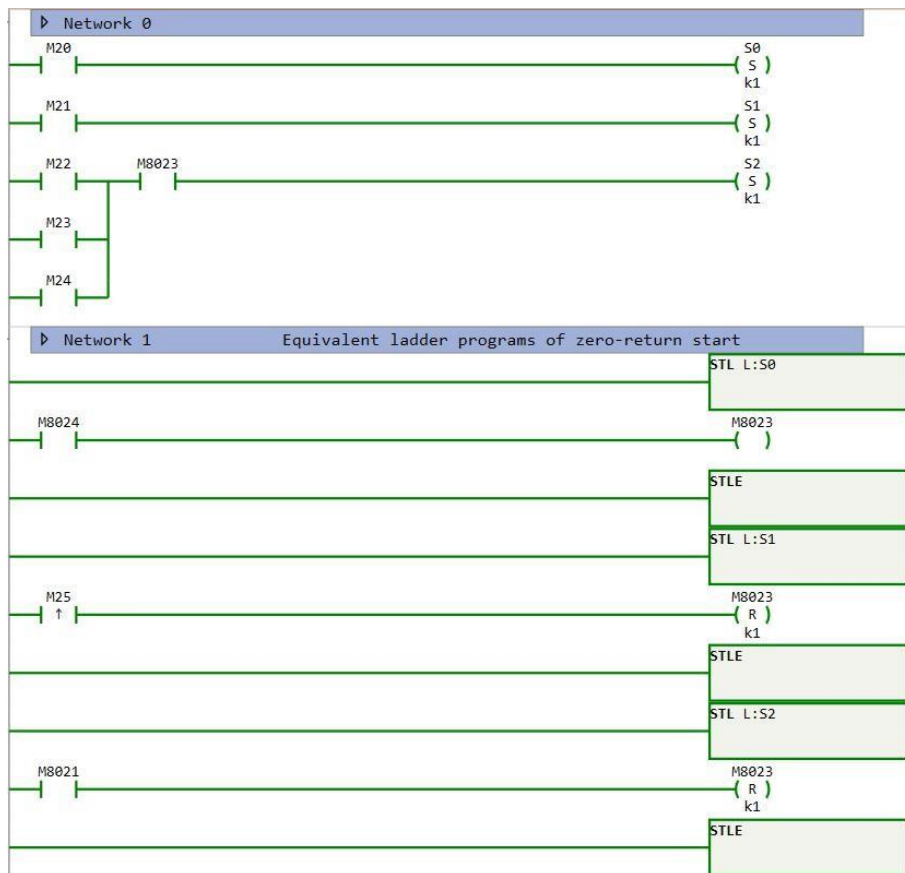
**M8022** wysyła impuls podczas operacji startu w trybie automatycznym lub startu operacji z powrotem do zera w trybie powrotu do zera. Przykładowy program drabinkowy zamieszczony został na rysunku poniżej.



## M8023

**M8023** włącza się tylko w trybie ręcznym po zakończeniu powrotu do zera. Przykładowy program drabinkowy zamieszczony został na rysunku poniżej.





## M8024

**M8024** jest markerem powrotu do zera; wykrywa, czy maszyna powraca do punktu zerowego (musi być przypisany przez użytkownika), po zakończeniu powrotu do zera marker ten włącza się, w przeciwnym razie pozostaje wyłączony.

## M8025

Gdy tryb jest przełączany pomiędzy trybem ręcznym, trybem powrotu do zera i trybem automatycznym, wszystkie wyjścia - cewki (Y) i przekaźniki (S) są resetowane, jeżeli maszyna nie znajduje się w punkcie zerowym. Jeżeli **M8025** został wcześniej ustawiony w pozycji ON, resetowane są tylko przekaźniki (S).

## M8026

W sytuacji kiedy którekolwiek z przekaźników (S) jest włączony, **M8026** włącza się. Gdy żaden z przekaźników (S) nie jest włączony, **M8026** jest wyłączony.

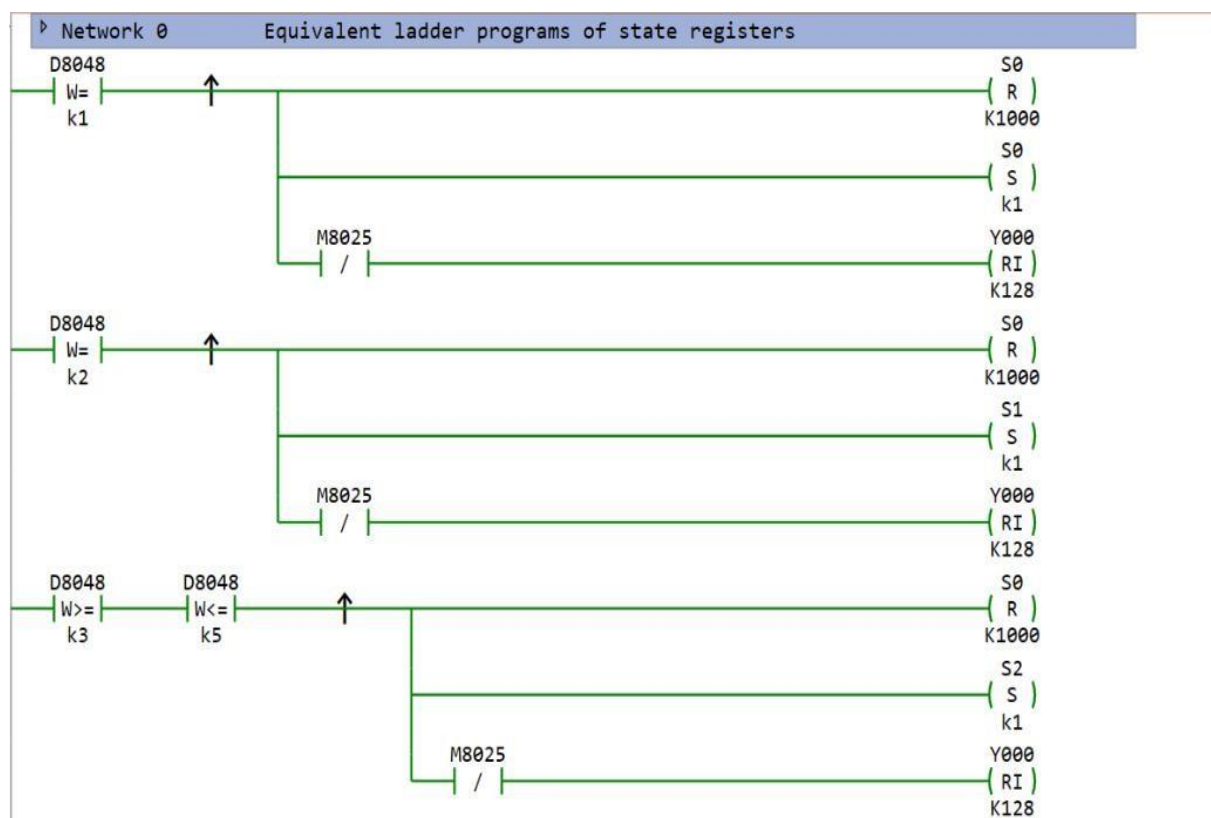
## M8027

Gdy **M8027** jest włączony, zapisuje w **D8040** do **D8047** sekwencyjnie od małego do dużego numeru przekaźników, które operują programem krokowym.



## Przełączniki S0, S1 i S2

**S0, S1 i S2** są w programie odpowiednio wyjściami ustawiającymi: trybu ręcznego, trybu powrotu do zera i trybu automatycznego. Po wybraniu jednego trybu, odpowiednia cewka załączająca S zostanie ustawiona na 1, a pozostałe zostaną zresetowane, w tym samym czasie wszystkie wyjściowe cewki Y zostaną natychmiast zresetowane; użytkownik może również ustawić **M8025** w pozycji wysokiej, aby anulować resetowanie wyjść Y. Przykład programu drabinkowego został zamieszczony poniżej.



Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Włącza lub wyłącza polecenie		0/1	Wartość logiczna
(IN)	Styk ustawiający program	X/M	0/1	Wartość logiczna

### Uwagi:

1. Użyj tej instrukcji na początku programu głównego i włącz ją za pomocą **M8151** (włączony podczas pracy PLC).
2. Niniejszej instrukcji można użyć tylko raz w projekcie.

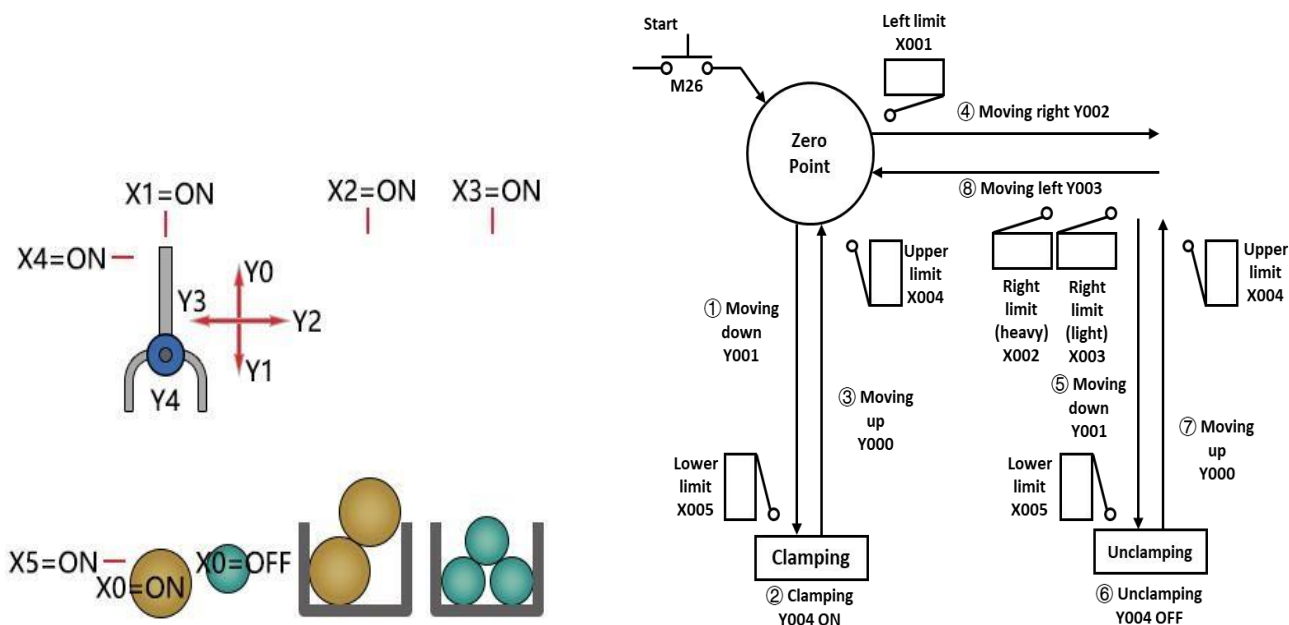


### Przykład

Zadanie polega na rozdzielaniu kulek o różnych ciężarach, mechanizm i przebieg pracy przedstawiono na rysunku poniżej. Mechaniczne ramię jest poruszane w górę za pomocą wyjścia Y000, w dół przez wyjście Y001, w prawo przez wyjście Y002, w lewo przez wyjście Y003; wejście X000 służy do rozróżnia wagi kul (ON dla ciężkich i OFF dla lekkich), wejście X001 to lewy limit ruchu poziomego, X002 to prawy limit ruchu ciężkich kul, X003 to prawy limit ruchu lekkich kulek, X004 to górna granica ruchu pionowego, X005 to dolna granica ruchu pionowego.

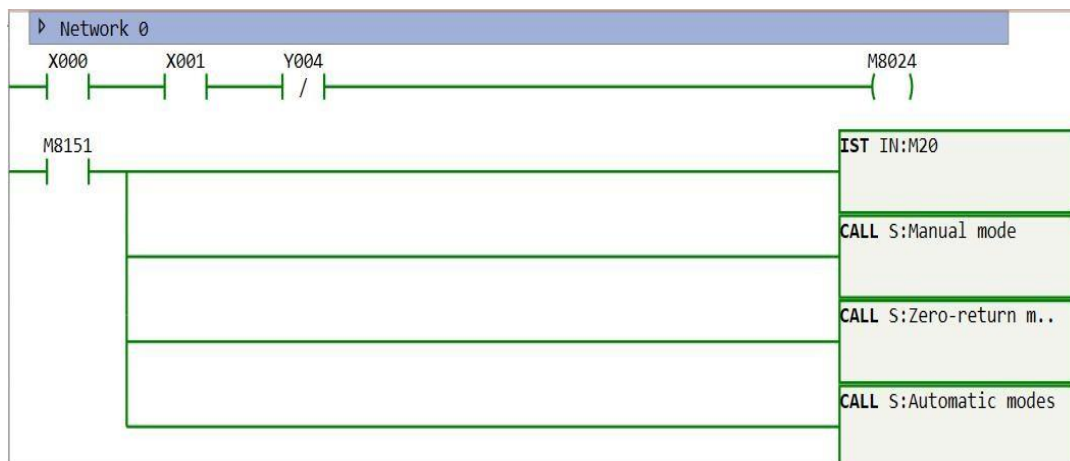
Proces pracy jest następujący:

1. Przesuń w dół do dolnej granicy od punktu zerowego.
2. Zaciśnij jedną kulę i oceń wagę.
3. Przejdź do górnej granicy.
4. Przesuń do prawej granicy ciężkich/lekkich kul, jeśli X000 jest WŁ./WYŁ.
5. Przejdź w dół do dolnego limitu.
6. Odepnij kulę.
7. Przesuń do górnej granicy.
8. Przesuń do lewej granicy.

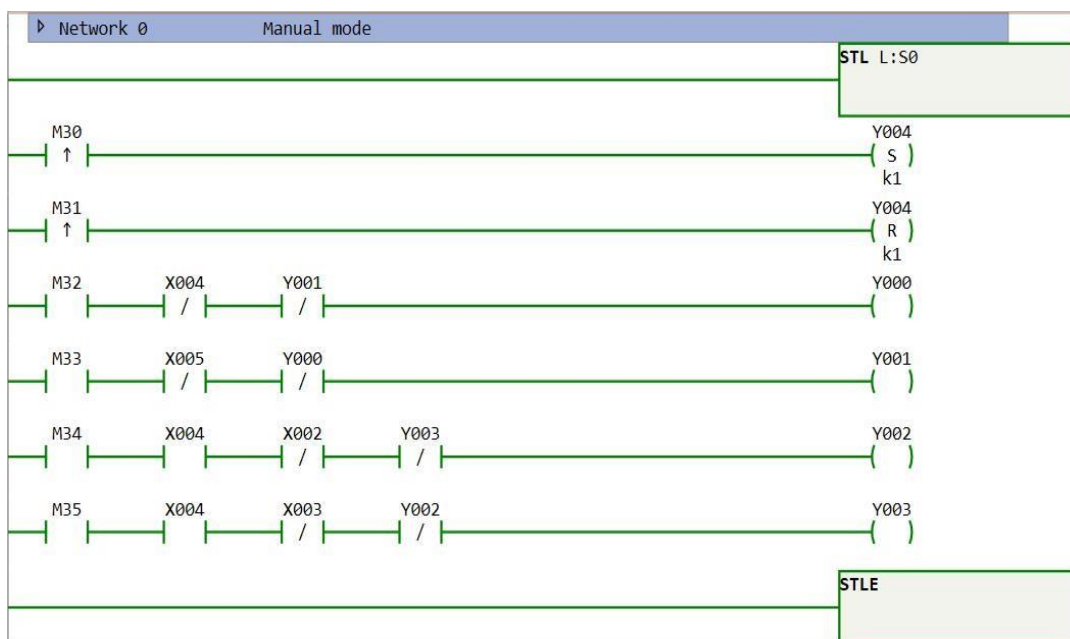


Program drabinkowy przedstawia się następująco:

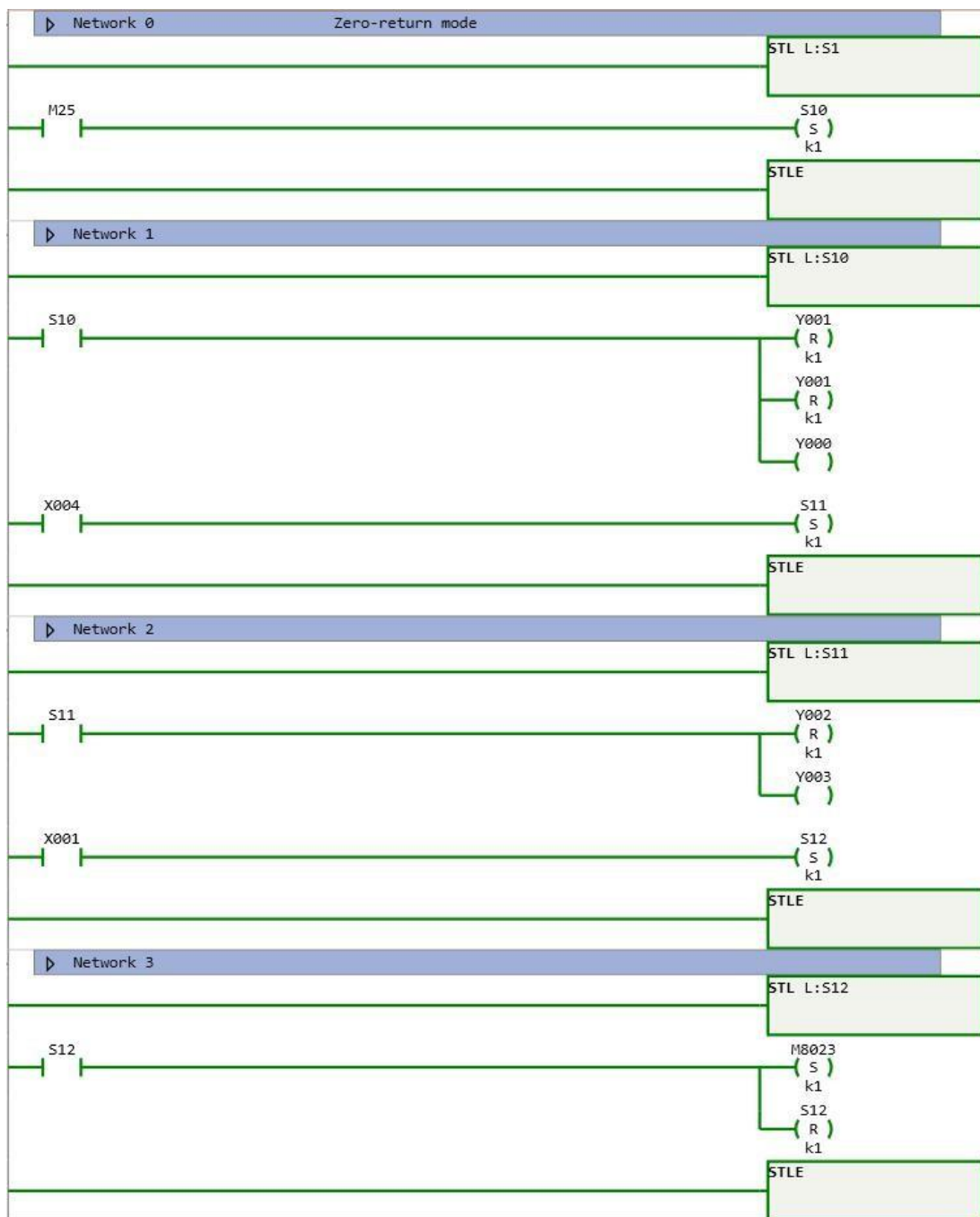
Program główny ustawia warunek powrotu do zera **M8024**, włącza instrukcję **IST** i wywołuje podprogram trzech rodzajów trybów (tryb ręczny, tryb powrotu do zera i tryb automatyczny).



Program trybu ręcznego posiada wejście do ręcznego sterowania wyjściem. Numer zastosowanej cewki ustawiającej to **S0**.



Program trybu powrotu do zera przesuwa ramię robota do punktu zerowego poprzez ustawienie **M25**. Numer zastosowanej cewki ustawiającej to **S1**.



Tryb pojedynczego kroku, tryb pojedynczego cyklu i tryb automatyczny współdzielą jeden podprogram, a numer zastosowanej cewki ustawiającej to **S2**.

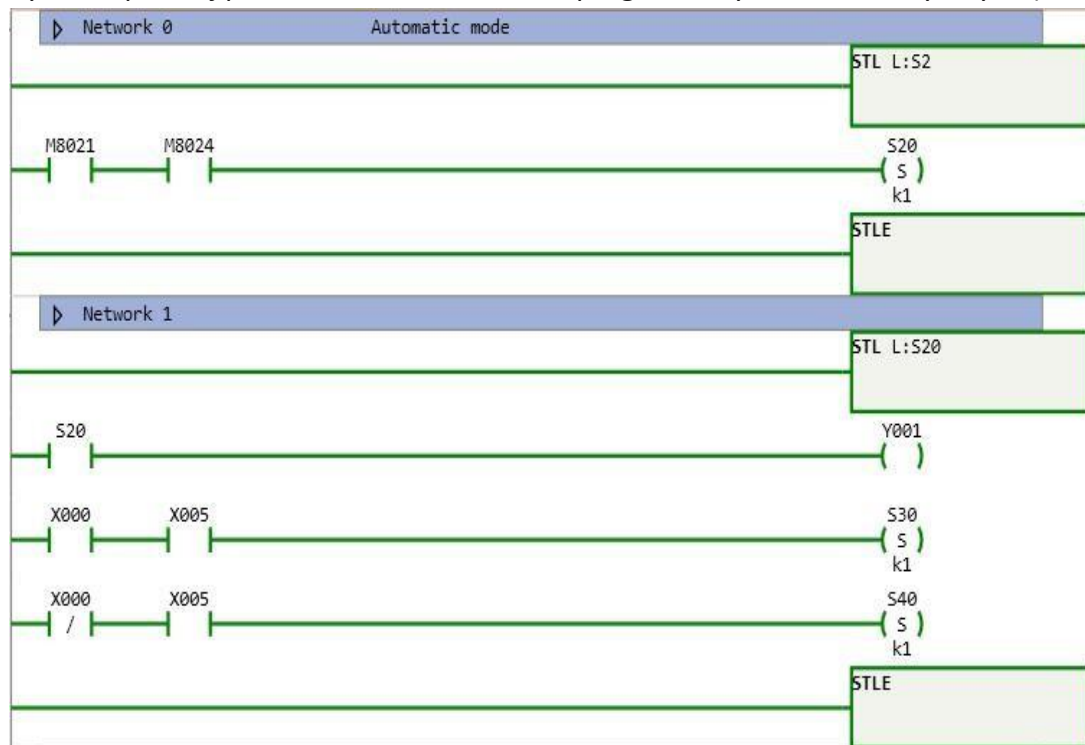
W trybie pojedynczego kroku proces programu przebiega krok po kroku, poprzez narastające zbocze **M26** (program nie wykona następnego kroku, dopóki stan **M25** nie zmieni się z niskiego na wysoki).

W trybie pojedynczego cyklu proces programu wykonuje cały cykl i nie będzie wykonywany następny cykl, dopóki nie pojawi się zbocze narastające **M26**.

W trybie automatycznym przebieg programu uruchamia się automatycznie.



Rysunek poniżej przedstawia uruchomienie programu trybów automatycznych (S2->S20)



Rysunek poniżej przedstawia rozgałęziony program przenoszenia ciężkich kul (S30->S31->S32);



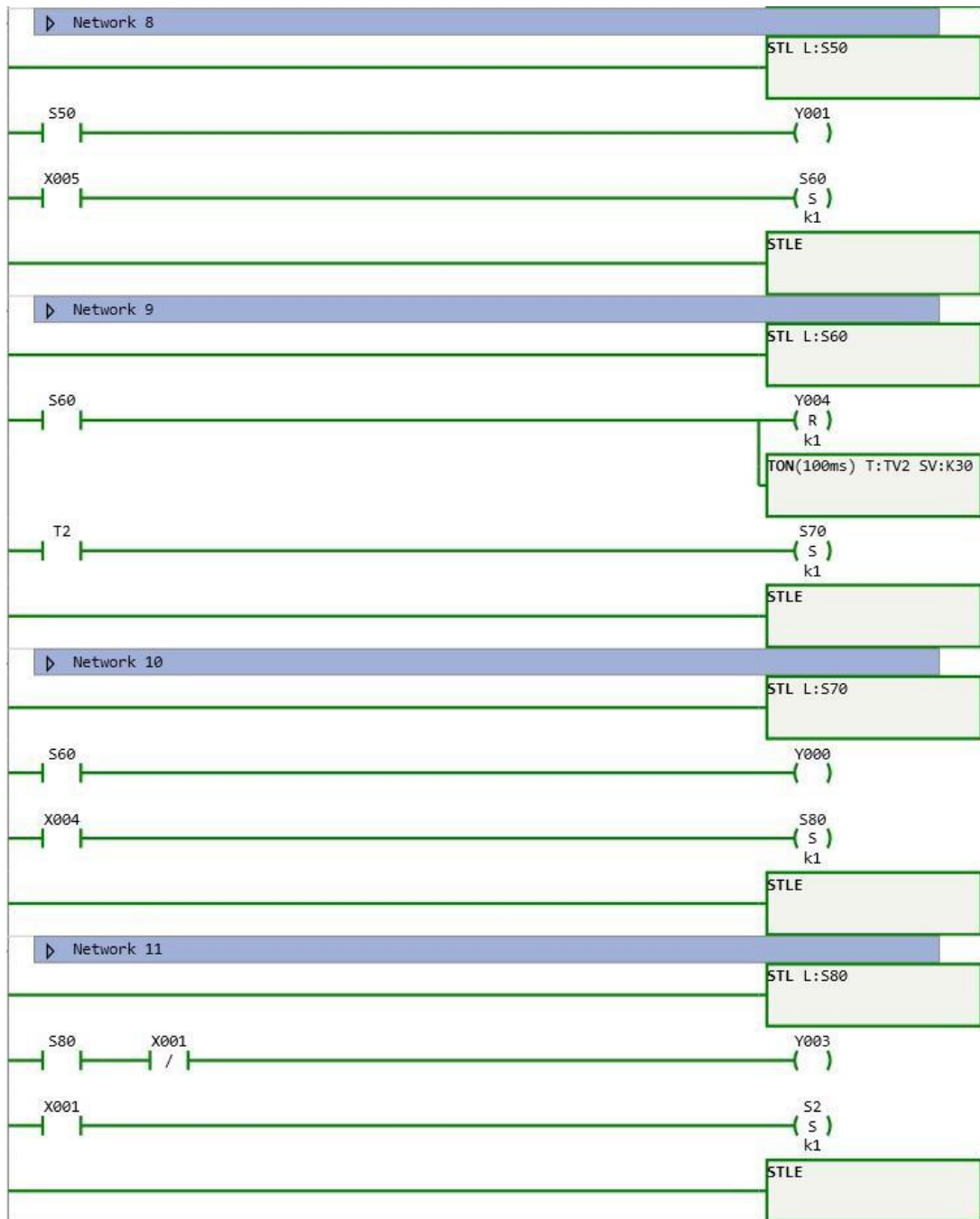
Rysunek poniżej przedstawia rozgałęziony program przenoszenia kul lekkich (S40->S41->S42);







Rysunek poniżej przedstawia program odkładania kuli i powrotu do stanu początkowego (S50->S60->S70->S2).





## ISTNEXT

Niniejsza instrukcja działa w trybie pojedynczego kroku i trybie pojedynczego cyklu. Gdy funkcja **ISTNEXT** jest włączona, można przenieść się do następnego kroku bez zakończenia aktualnego kroku.

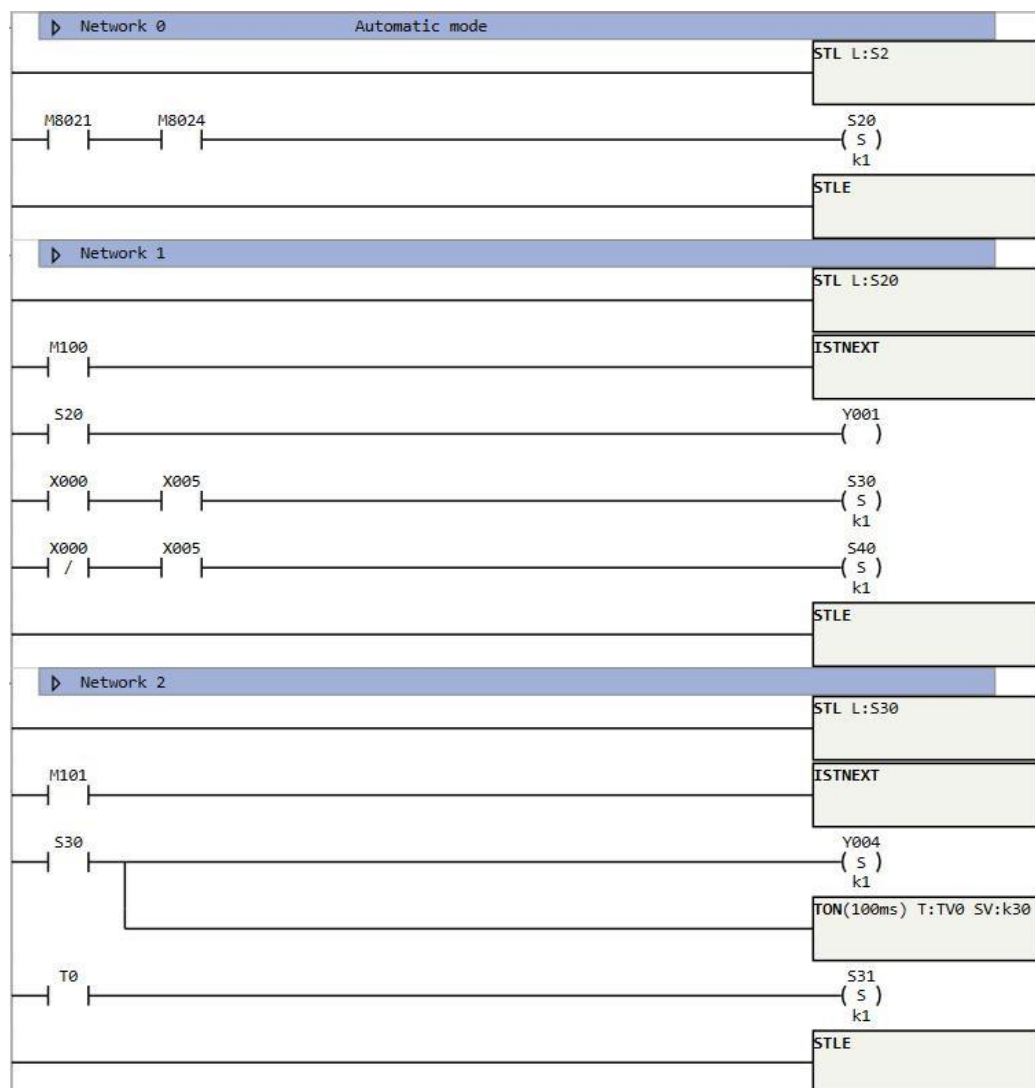
**M8020** zostanie zresetowany po włączeniu tej instrukcji, aby mógł przejść do następnego kroku.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna

### Przykład

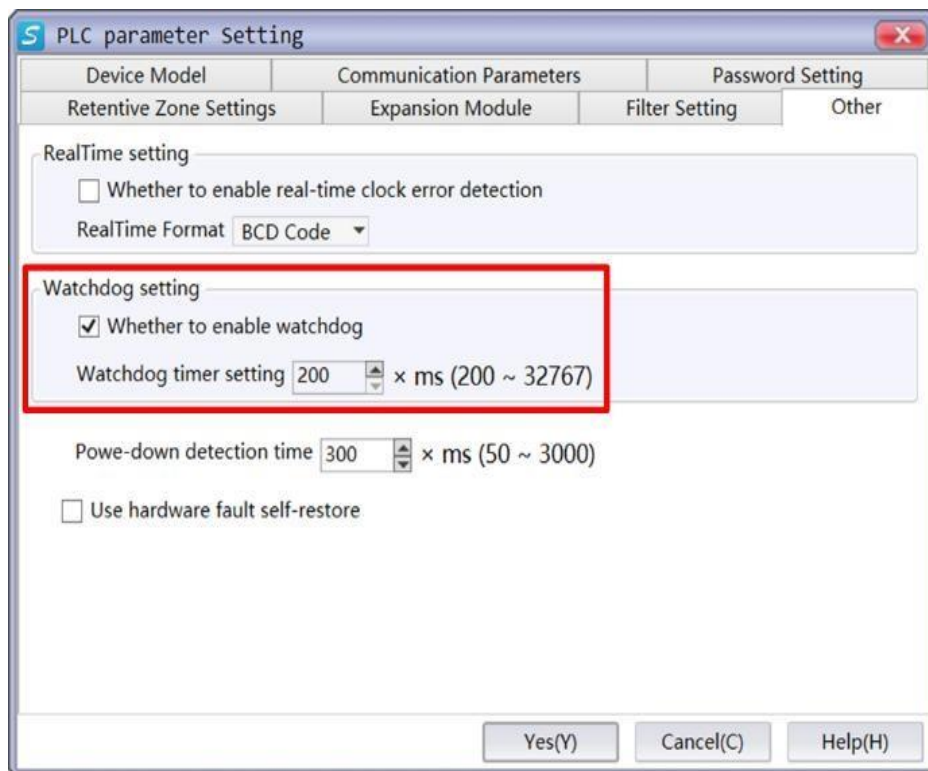
Odnosząc się do [przykładu](#) trybu automatycznego w **IST**, Rysunek poniżej pokazuje przypadek, w którym zastosowano **ISTNEXT**.





## WDT

1. Ta instrukcja resetuje timer watchdoga w programie.
2. Użytkownik może rozpocząć korzystanie z funkcji watchdog i ustawić wartość docelową timera watchdoga w ustawieniach, jak pokazano na rysunku poniżej. Watchdog będzie resetowany automatycznie w każdym okresie skanowania, ale można go również zresetować ręcznie za pomocą tej instrukcji. Jeśli czas zegara kontrolnego przekroczy czas docelowy, PLC zatrzyma się i zgłosi kod błędu 11 w **D8176**.
3. Przekroczenie czasu zegara watchdoga jest zawsze spowodowane nieskończoną pętlą w programie.



### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna

### Lista i instrukcji:

#### Network 000

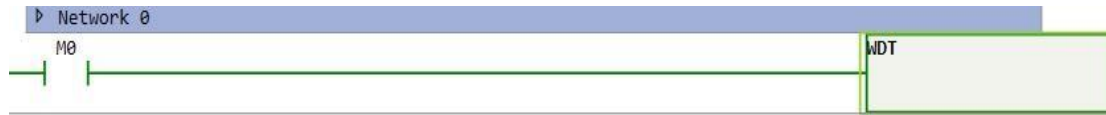
LD M0

WDT // zresetuj timer watchdoga przy zboczu narastającym M0



POP

Schemat drabinkowy:





## INSTRUKCJE PRZESUNIĘCIA

1. Instrukcje w tym rozdziale wykonują operacje przesuwania na rejestrach danych i kolejnych rejestrach bitowych.

2. Istnieją systemowe rejestry markery, które rejestrują stan wyniku przesunięcia:

**M8166:** Kiedy ostatni bit operacji przesunięcia wynosi 1, ten marker jest włączony.

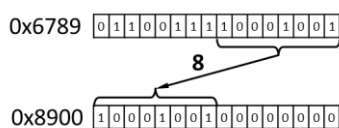
**M8167:** Gdy długość operacji przesuwania wynosi 0, ten marker jest włączony.

## SHL/SHLD

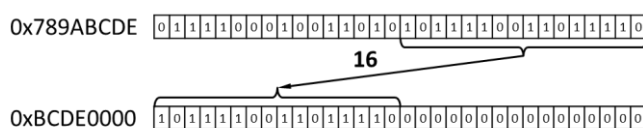
1. Przesuwają dane wejściowe w lewo o (n) bitów i wyprowadzają wynik do określonych rejestrów (rysunek poniżej). **SHL** dotyczy danych typu **WORD**, **SHLD** dotyczy danych typu **DWORD**.

2. Ta instrukcja może ustawić **M8166** i **M8167** w pozycji ON.

### SHL



### SHLD



## SHL

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(IN1)	Rejestr źródłowy do przesunięcia w lewo.	K/H/D/TV/CV/AI/AO/V/Z	-	Kod 16-bitowy
(IN2)	Ilość bitów do przesunięcia w lewo.	K/H/D	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy do przechowywania wyniku	D/TV/CV/AO/V/Z	-	Kod 16-bitowy

## SHLD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
---------------------	------	---------	--------	------------



(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(IN1)	Rejestr źródłowy do przesunięcia w lewo.	K/H/D/TV/CV/AI/AO/V/Z	-	Kod 32-bitowy
(IN2)	Ilość bitów do przesunięcia w lewo.	K/H/D	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy do przechowywania wyniku	D/TV/CV/AO/V/Z	-	Kod 32-bitowy

Przykład

Lista instrukcji

Network 000

LD X000

SHL D0 K1 D1 //przesuń dane D0 w lewo o 1-bit i zachowaj otrzymane dane w D1

POP

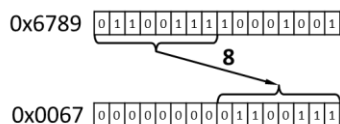
Schemat drabinkowy:



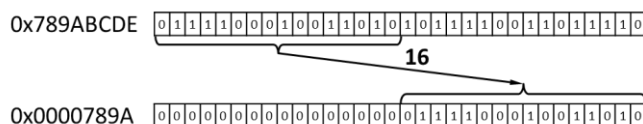
## SHR/SHRD

1. Przesuwają dane wejściowe w prawo o (n) bitów i wyprowadzają wynik do określonych rejestrów (rysunek poniżej). **SHR** dotyczy danych typu **WORD**, **SHRD** dotyczy danych typu **DWORD**.
2. Ta instrukcja może ustawić **M8166** i **M8167** w pozycji ON.

### SHR



### SHRD





## SHR

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(IN1)	Rejestr źródłowy do przesunięcia w prawo.	K/H/D/TV/CV/AI/AO/V/Z	-	Kod 16-bitowy
(IN2)	Ilość bitów do przesunięcia w prawo.	K/H/D	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy do przechowywania wyniku	D/TV/CV/AO/V/Z	-	Kod 16-bitowy

## SHRD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(IN1)	Rejestr źródłowy do przesunięcia w prawo.	K/H/D/TV/CV/AI/AO/V/Z	-	Kod 32-bitowy
(IN2)	Ilość bitów do przesunięcia w prawo.	K/H/D	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy do przechowywania wyniku	D/TV/CV/AO/V/Z	-	Kod 32-bitowy

### Przykład

### Lista instrukcji

### Network 000

LD X000

SHLR D0 K1 D1 //przesuń dane D0 w prawo o 1-bit i zachowaj otrzymane dane w D1

POP



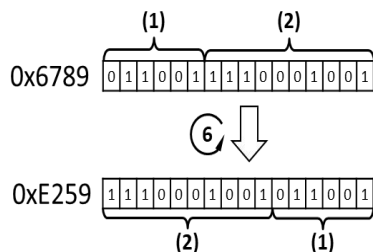
Schemat drabinkowy:



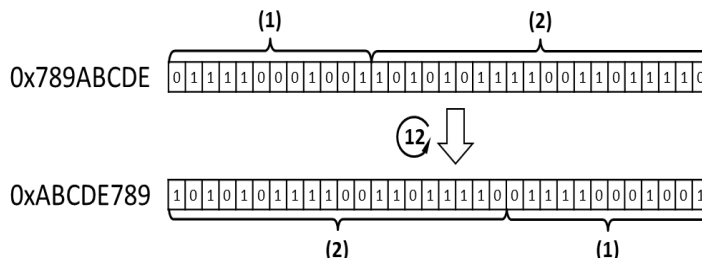
## ROL/ROLD

1. Instrukcje te przemieniają dane wejściowe w lewą stronę o (n) bitów i wyprowadzają wynik do określonych rejestrów (rysunek poniżej). **ROL** dotyczy danych typu **WORD**, **ROLD** dotyczy danych typu **DWORD**.
2. Ta instrukcja może ustawić **M8166** i **M8167** w pozycji ON.

### ROL



### ROLD



## ROL

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(IN1)	Rejestr źródłowy do wymiany w lewą stronę.	K/H/D/TV/CV/AI/AO/V/Z	-	Kod 16-bitowy
(IN2)	Ilość bitów do wymiany w lewą stronę.	K/H/D	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy do przechowywania wyniku	D/TV/CV/AO/V/Z	-	Kod 16-bitowy





## ROL

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(IN1)	Rejestr źródłowy do wymiany w lewą stronę.	K/H/D/TV/CV/AI/AO/V/Z	-	Kod 32-bitowy
(IN2)	Ilość bitów do wymiany w lewą stronę.	K/H/D	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy do przechowywania wyniku	D/TV/CV/AO/V/Z	-	Kod 32-bitowy

Przykład

Lista instrukcji

Network 000

LD X000

ROL D0 K1 D1 //przemień dane D0 w lewo o 1-bit i zachowaj wynik w D1

POP

Schemat drabinkowy:

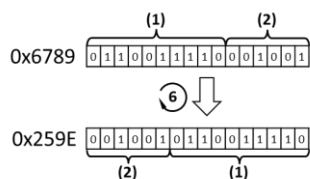


## ROR/RORD

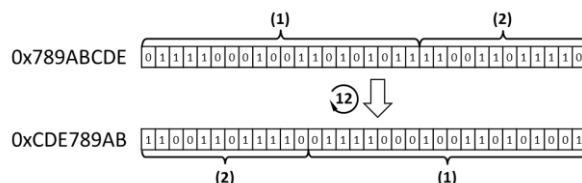
- Instrukcje te przemieniają dane wejściowe w prawą stronę o (n) bitów i wyprowadzają wynik do określonych rejestrów (rysunek poniżej). **ROR** dotyczy danych typu **WORD**, **RORD** dotyczy danych typu **DWORD**.
- Ta instrukcja może ustawić **M8166** i **M8167** w pozycji ON.



## ROR



## RORD



## ROR

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(IN1)	Rejestr źródłowy do wymiany w prawą stronę.	K/H/D/TV/CV/AI/AO/V/Z	-	Kod 16-bitowy
(IN2)	Ilość bitów do wymiany w prawą stronę.	K/H/D	-32768~32767	16-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy do przechowywania wyniku	D/TV/CV/AO/V/Z	-	Kod 16-bitowy

## RORD

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(IN1)	Rejestr źródłowy do wymiany w prawą stronę.	K/H/D/TV/CV/AI/AO/V/Z	-	Kod 32-bitowy
(IN2)	Ilość bitów do wymiany w prawą stronę.	K/H/D	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem
(OUT)	Rejestr docelowy do przechowywania wyniku	D/TV/CV/AO/V/Z	-	Kod 32-bitowy

Przykład

Lista instrukcji

Network 000

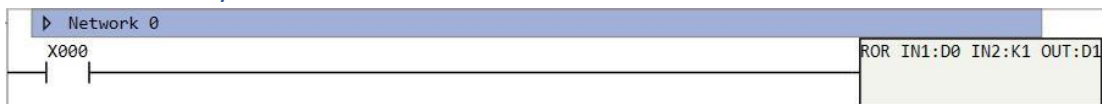


LD X000

ROR D0 K1 D1 //przemień dane D0 w prawo o 1-bit i zachowaj wynik w D1

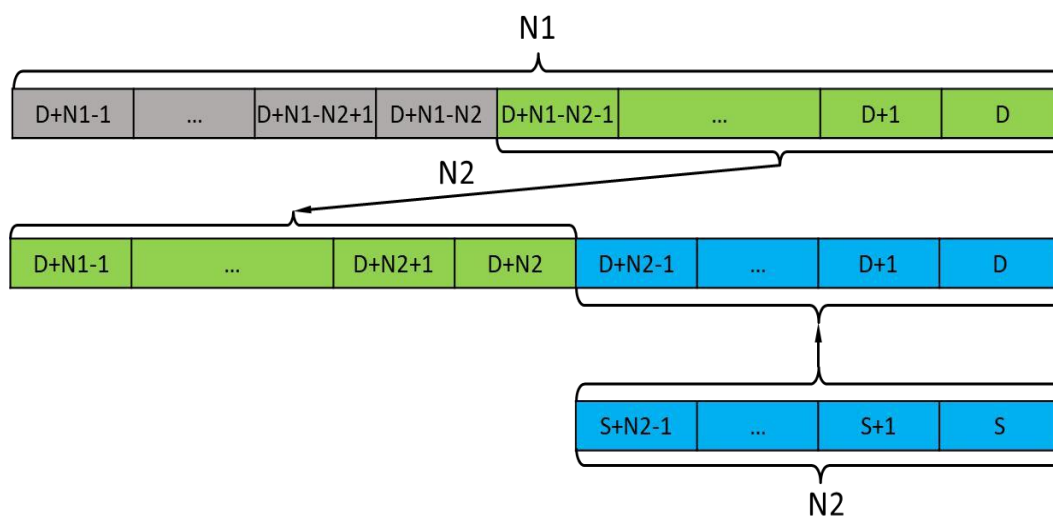
POP

Schemat drabinkowy:



## SHLB

1. Ta instrukcja przesuną ciąg kolejnych danych bitowych (D) długości (N1) w lewo o długość (N2) i zapełnia powstałe puste miejsca ciągiem bitów źródłowych (S), (szczegóły patrz rysunek)
2. Ta instrukcja może ustawić **M8166** i **M8167** w pozycji ON.



Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(S)	Pierwszy ze źródłowych bitów do wypełnienia pustego miejsca	X/Y/M/S/T/C	-	Wartość logiczna
(D)	Pierwszy z bitów do przesunięcia w lewo	Y/M/S/T/C	-	Wartość logiczna



(N1)	Długość ciągu źródłowego.	K/H/D	-32768~32767	16-bitowa liczba całkowita ze znakiem
(N2)	Długość przesunięcia.	K/H/D	-32768~32767	16-bitowa liczba całkowita ze znakiem

### Przykład

### Lista instrukcji

### Network 000

LD X000

SHLB M0 Y0 K16 K1 //przesuń bity Y0~Y15 o 1 w lewo, i zapełnij miejsce od Y0 danymi z M0

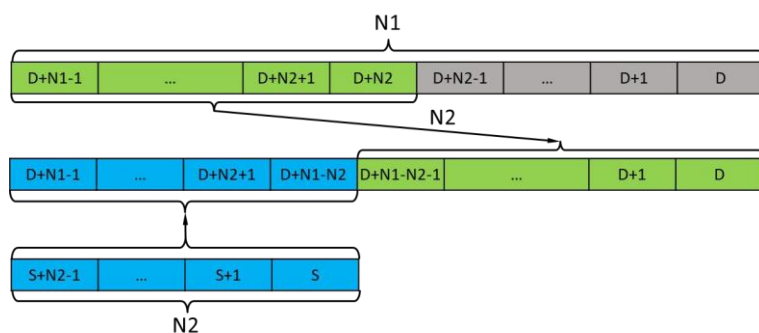
POP

### Schemat drabinkowy:



## SHRB

1. Ta instrukcja przesuwa ciąg kolejnych danych bitowych (D) o długości (N1) w prawo o długość (N2) i zapełnia powstałe puste miejsca ciągiem bitów źródłowych (S), (szczegóły patrz rysunek)
2. Ta instrukcja może ustawić **M8166** i **M8167** w pozycji ON.



Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna



(S)	Pierwszy ze źródłowych bitów do wypełnienia pustego miejsca	X/Y/M/S/T/C	-	Wartość logiczna
(D)	Pierwszy z bitów do przesunięcia w prawo	Y/M/S/T/C	-	Wartość logiczna
(N1)	Długość ciągu źródłowego.	K/H/D	-32768~32767	16-bitowa liczba całkowita ze znakiem
(N2)	Długość przesunięcia.	K/H/D	-32768~32767	16-bitowa liczba całkowita ze znakiem

Przykład

Lista instrukcji

Network 000

LD X000

SHRB M0 Y0 K16 K1 //przesuń bity Y0~Y15 o 1 w prawo, i zapełnij miejsce od Y0 danymi z M0

POP

Schemat drabinkowy:





## INSTRUKCJE PRZERWANIA

1. Przerwanie jest funkcją procesora PLC do obsługi szczególnych przypadków, takich jak błąd, alarm. Przerwanie może być wywołane przez wybrane czynniki przerwania, użytkownik może powiązać te czynniki przerwania z programem obsługi przerwania (podprogramem), tak aby odpowiedni program obsługi przerwania został wykonany, w przypadku wystąpienia czynnika przerwania.
2. Pierwszeństwo przerwań: Gdy dwa przerwania o różnych priorytetach zostaną wyzwolone jednocześnie, w pierwszej kolejności obsłużone zostanie przerwanie o wyższym priorytecie. Jeśli przerwanie o wyższym priorytecie zostanie wyzwolone, gdy obsługiwane jest przerwanie o niższym priorytecie, obsługa przerwania o niższym priorytecie zostanie wstrzymana do czasu zakończenia obsługi przerwania o wyższym priorytecie.
3. Tabela poniżej przedstawia tablicę wektorów adresów przerwań dla przerwań zewnętrznych. Przerwania zmiany zbocza i przerwania czasowe mają niższy priorytet niż inne. Nie należy wyzwać zewnętrznego przerwania zbyt często, w przeciwnym razie niektóre przerwania mogą nie być obsługiwane poprawnie.

	Numer przerwania	Czynnik przerywający
Dla wszystkich serii	K0	Narastające zbocze wejścia X000
	K1	Opadające zbocze wejścia X000
	K2	Zmiana sygnału zbocza na wejściu X000
	K3	Narastające zbocze wejścia X001
	K4	Opadające zbocze wejścia X001
	K5	Zmiana sygnału zbocza na wejściu X001
	K6	Timer przerwań 0 osiąga czas docelowy (czas docelowy można ustawić za pomocą specjalnego rejestru systemowego <b>D8173</b> , zakres wynosi 1 ~ 32767 ms, rozdzielczość wynosi 1 ms)
	K7	Timer przerwań 1 osiąga czas docelowy (czas docelowy można ustawić za pomocą specjalnego rejestru systemowego <b>D8174</b> , zakres wynosi 1 ~ 32767 ms, rozdzielczość wynosi 1 ms)
Dla serii FGm	K8	Wysyłanie impulsu Y000 zakończone.
	K9	Wysyłanie impulsu Y001 zakończone.
	K10	Wysyłanie impulsu Y002 zakończone.
	K11	Wysyłanie impulsu Y003 zakończone.
	K12	Wysyłanie impulsu Y004 zakończone.
	K13	Wysyłanie impulsu Y005 zakończone.
	K14	Wysyłanie impulsu Y006 zakończone.
	K15	Wysyłanie impulsu Y007 zakończone.



K16	Wysyłanie impulsu Y010 zakończone.
K17	Wysyłanie impulsu Y011 zakończone.
K18	Licznik CV235 zlicza do liczby docelowej
K19	Licznik CV236 zlicza do liczby docelowej
K20	Licznik CV237 zlicza do liczby docelowej
K21	Licznik CV238 zlicza do liczby docelowej
K22	Licznik CV239 zlicza do liczby docelowej
K23	Licznik CV240 zlicza do liczby docelowej
K24	Licznik CV241 zlicza do liczby docelowej
K25	Licznik CV242 zlicza do liczby docelowej
K26	Licznik CV243 zlicza do liczby docelowej
K27	Licznik CV244 zlicza do liczby docelowej
K28	Licznik CV245 zlicza do liczby docelowej
K29	Licznik CV246 zlicza do liczby docelowej
K30	Narastające zbocze wejścia X002
K31	Opadające zbocze wejścia X002
K32	Zmiana sygnału zbocza na wejściu X002
K33	Narastające zbocze wejścia X003
K34	Opadające zbocze wejścia X003
K35	Zmiana sygnału zbocza na wejściu X003
K36	Narastające zbocze wejścia X004
K37	Opadające zbocze wejścia X004
K38	Zmiana sygnału zbocza na wejściu X004
K39	Narastające zbocze wejścia X005
K40	Opadające zbocze wejścia X005
K41	Zmiana sygnału zbocza na wejściu X005

## ATCH

Ta instrukcja łączy czynnik przerwania z programem obsługi przerwania. Odpowiedni podprogram zostanie wykonany, gdy wystąpi czynnik przerwania, jeśli polecenie przerwania jest włączone.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
---------------------	------	---------	--------	------------



(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(EV)	Podprogram do wywołania	-	-	wskaźnik 32-bitowy
(INT)	Numer czynnika przerwania	K	0~41	16-bitowa liczba całkowita bez znaku

Przykład

Lista instrukcji:

Network 000

LDP M0

ATCH INTR\_0 K0 // połącz czynnik przerwania K0 z programem podprogramu INTR\_0

EI // udziel zgody na przerwanie

Network 001

LDP M1

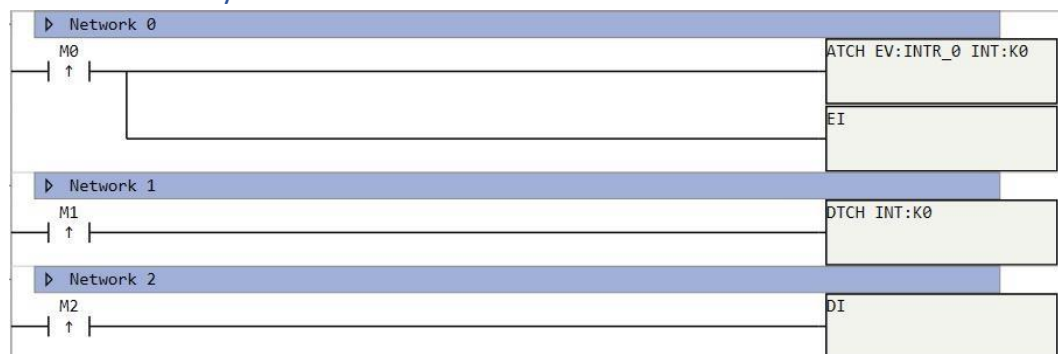
DTCH K0 // odłącz czynnik przerwania K0 od dołączonego programu przerwania

Network 002

LDP M2

DI // wyłącz przerwanie

Schemat drabinkowy:



## DTCH

1. Ta instrukcja odłącza czynnik przerwania od dołączonego programu obsługi przerwania.
2. W konsekwencji rozkazu **DTCH**, zliczanie timera przerwania 0 i timera przerwania 1 zostanie zresetowane, odpowiedni numer docelowy w **D8713** i **D8174** również zostanie zresetowany.

Przykład

Patrz [przykład](#) ATCH.





## EI

1. Ta instrukcja umożliwia wykonanie programu obsługi przerwania; program obsługi przerwania nie zostanie wykonany, jeśli EI nie jest włączone.
2. Kiedy PLC przechodzi w stan RUN, przerwanie jest domyślnie wyłączone.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna

### Przykład

Patrz [przykład](#) ATCH.

## DI

Ta instrukcja blokuje wykonywanie programu przerwania do momentu włączenia instrukcji EI.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna

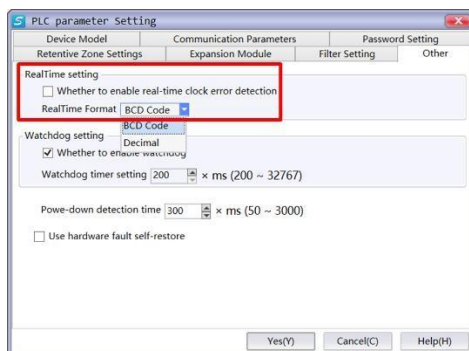
### Przykład

Patrz [przykład](#) ATCH.



## INSTRUKCJE ZEGARA CZASU RZECZYWISTEGO

1. Instrukcje w tym rozdziale wykonują operacje odczytu, zapisu i porównywania danych zegara, a zegar systemowy PLC nazywa się RTC (ang. *real-time clock* -zegar czasu rzeczywistego).
2. Dane RTC w PLC składają się z 8 części: rok, miesiąc, dzień, godzina, minuta, sekunda, część rezerwowa i tydzień. Dane RTC są przechowywane w formacie **BCD**.
3. Użytkownik może ustawić format danych zegara (patrz rysunek poniżej). Format danych zegara można wybrać za pomocą kodu BCD lub kodu BIN (BIN i BCD patrz uwagi do instrukcji konwersji). Dane RTC samego PLC są formatowane za pomocą BCD, a ustawienie nie ma na to wpływu



4. Tabela poniżej pokazuje szczegóły odczytu danych zegarowych (dla zapisu danych tygodnia: 0 to niedziela, 1 to poniedziałek, 2 to wtorek, 3 to środa, 4 to czwartek, 5 to piątek, 6 to sobota).

Numer rejestru danych	Opis	Zakres	
		BCD	BIN
0	Rok	0x2000~0x2099	2000~2099
1	Miesiąc	0x0001~0x0012	1~12
2	Dzień	0x0001~0x0031	1~31
3	Godzina	0x0000~0x0023	0~23
4	Minuta	0x0000~0x0059	0~59
5	Sekunda	0x0000~0x0059	0~59
6	Zarezerwowany	-	-
7	Dzień tygodnia	0x0000~0x0006	0~6

## TRD

Ta instrukcja odczytuje dane zegara czasu rzeczywistego **RTC** i zapisuje je w określonych 8 kolejnych komórkach danych D, gdy funkcja jest włączona (szczegóły na [początku](#) rozdziału).

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
---------------------	------	---------	--------	------------



(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(T)	Adres pierwszej komórki D do przechowywania danych	D	-	-

#### Lista instrukcji:

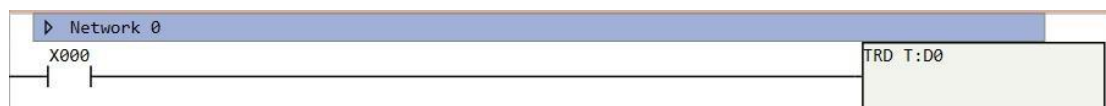
#### Network 000

LD X000

TRD D0 //odczytaj dane z RTC i zapisz D0~D7

POP

#### Schemat drabinkowy:



## TWR

1. Ta instrukcja wpisuje do PLC dane RTC z określonych 8 kolejnych elementów danych D, gdy jest włączona
2. Ta instrukcja powinna być wyzwalana sygnałem zbocza.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(T)	Rejestr D do przechowywania danych	D	-	-

#### Przykład

#### Lista instrukcji:

#### Network 000

LD X000

TWR D0 //zapisz dane RTC z D0~D7 w PLC

POP

#### Schemat drabinkowy:





## TRDS

1. Ta instrukcja odczytuje dane RTC, oblicza ile sekund upłynęło od 2000/1/1 00:00:00 i zapisuje drugą liczbę w określonym rejestrze danych D (double-word).
2. Ta instrukcja powinna być wyzwalana sygnałem zbocza.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(T)	Rejestr D do przechowywania ilości sekund	D	0~ 4294967295	32-bitowa liczba całkowita bez znaku

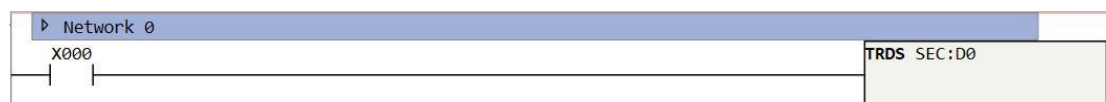
### Przykład

#### Lista instrukcji:

#### Network 000

```
LD    X000
TDRS  D0 //zapisz liczbę przeminętych sekund w D0D1
POP
```

#### Schemat drabinkowy:



## TWRS

1. Na podstawie liczby sekund, które upłynęły od czasu 2000/1/1 00:00:00 zapisanego w określonym rejestrze danych D, instrukcja ta oblicza dane zegara i wpisuje je do PLC.
2. Ta instrukcja powinna być wyzwalana sygnałem zbocza.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(T)	Rejestr danych D do przechowywania ilości sekund	D	0~ 4294967295	32-bitowa liczba całkowita bez znaku



## TSEC

Ta instrukcja odczytuje dane zegara z określonego ciągu 8 kolejnych komórek D, oblicza ile sekund upłynęło od 2000/1/1 00:00:00 i zapisuje liczbę sekund w określonych D (double-word).

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(T)	Adres pierwszej komórki D przechowującej dane zegara	D	-	-
(SEC)	Rejestr D do zapisania otrzymanego wyniku	D	0~ 4294967295	32-bitowa liczba całkowita bez znaku

Przykład

Lista instrukcji:

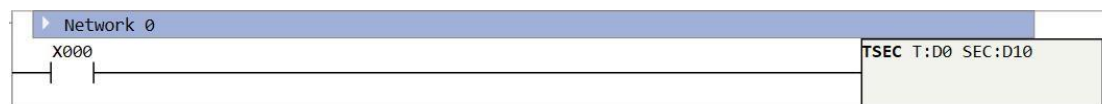
### Network 000

LD X000

TSEC D0 D10 //przekształć dane z D0~D7 na liczbę sekund i zapisz w D10

POP

Schemat drabinkowy:



## RSEC

Ta instrukcja odczytuje liczbę sekund z określonego rejestru D, oblicza dane zegara, na podstawie określonych sekund po 2000/1/1 00:00:00 i przechowuje dane zegara w określonym ciągu 8 kolejnych komórek D.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(SEC)	Rejestr D przechowujący dane zegara	D	0~ 4294967295	32-bitowa liczba całkowita bez znaku
(T)	Adres pierwszej komórki D do zapisania otrzymanego wyniku	D	-	-



## Przykład

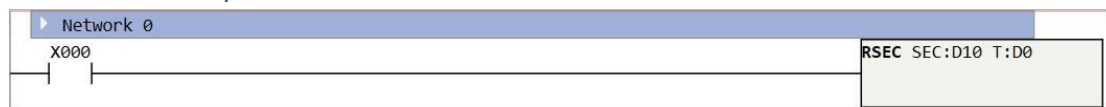
## Lista instrukcji:

## Network 000

LD X000

RSEC D10 D0 //zamień sekundy z D10 w dane zegara i zapisz w D0~D7

## Schemat drabinkowy:



## TCMP

Ta instrukcja odczytuje dane zegara RTC z PLC, porównuje je z danymi zegara przechowywanymi w określonym ciągu 8 komórek D i wyprowadza wynik do określonych bitów wyjściowych. Jeśli czas wejściowy jest wcześniejszy niż RTC, pierwszy z wyjściowych bitów jest włączany; Jeśli czas wejściowy jest taki sam jak RTC, drugi z wyjściowych bitów jest włączany; Jeśli czas wejściowy jest późniejszy niż RTC, trzeci wyjściowy bit jest włączany.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(T)	Adres pierwszej komórki D przechowywującej dane zegara	D	-	-
(OUT)	Adres pierwszej komórki do danych wyjściowych	Y/M/S	0/1	Wartość logiczna

## Przykład

## Lista instrukcji:

## Network 000

LD X000

TCMP D0 Y0 //porównaj czas z D0~D7 z RTC i przełącz wynik do wyjścia Y0~Y2

## Schemat drabinkowy:





## TZCP

1. Ta instrukcja odczytuje dane zegara RTC z PLC , porównuje je z przedziałem czasowym składającym się z danymi startowymi zegara i końcowymi, następnie wysyła wynik do określonych adresów bitowych. Jeśli czas RTC jest wcześniejszy niż przedział czasowy, pierwszy z wyjściowych bitów jest włączony; Jeśli czas RTC mieści się w ramach czasowych, drugi z wyjściowych bitów jest włączany; Jeśli czas RTC jest po przedziale czasowym, trzeci z bitów jest włączany.
2. Godzina zakończenia musi być późniejsza niż godzina rozpoczęcia. Na przykład ramy czasowe takie jak [2020/1/1 12:00:00 środa, 2019/1/1 12:00:00 wtorek] są niedozwolone.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(S)	Adres pierwszej z komórek D przechowujących startowe dane zegara	D	-	-
(E)	Adres pierwszej z komórek D przechowujących końcowe dane zegara	D	-	-
(OUT)	Adres pierwszej komórki do danych wyjściowych	Y/M/S	0/1	Wartość logiczna

### Przykład

#### Lista instrukcji:

#### Network 000

LD X000

TCMP D0 D10 Y0 //porównaj dane zegara RTC z przedziałem czasowym od D0~D7 do D10~D17, wynik prześlij do Y0~Y2

POP

#### Schemat drabinkowy:



## WKCMP

1. Porównuje dane RTC z określonymi danymi zegara, nie zważając na dane roku, miesiąca i dnia.



2. Ta instrukcja odczytuje dane RTC (bez roku, miesiąca i dnia), porównuje je z danymi czasu i tygodnia przechowywanymi w określonych 5 kolejnych adresach D-rejestrów (ostatnie 5 słów całej danych zegara) i wyprowadza wynik do określonej rejestry bitowe. Jeśli wprowadzony czas i tydzień są wcześniejsze niż czas RTC, pierwszy z wyjściowych rejestrów bitowych jest włączany; Jeśli wprowadzony czas i tydzień są takie same dla czas RTC, drugi z wyjściowych rejestrów bitowych jest włączany; Jeśli wprowadzony czas i tydzień są późniejsze niż czas RTC, trzeci z wyjściowych rejestrów bitowych jest włączany.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(T)	Adres pierwszej komórki D przechowującej dane godziny i dnia tygodnia	D	-	-
(OUT)	Adres pierwszej komórki do danych wyjściowych	Y/M/S	0/1	Wartość logiczna

Przykład:

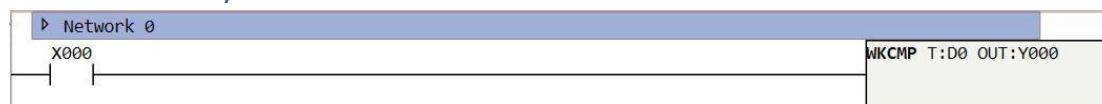
Lista instrukcji:

#### Network 000

LD X000

WKZCMP D0 Y0 //porównaj dane godziny i dnia tygodnia D0~D4 z danymi zegara RTC i wyślij wynik do Y0~Y2  
POP

Schemat drabinkowy:



## WKZCP

1. Porównuje dane zegara RTC z określonym przedziałem czasowym, nie zważając na dane roku, miesiąca i dnia.
2. Ta instrukcja odczytuje dane RTC (bez roku, miesiąca i dnia), porównuje je z przedziałem czasowym składającym się z zegara początkowego (bez roku, miesiąca i dnia) i zegara końcowego (bez roku, miesiąca i dnia) i wyprowadzają wynik do określonych bitów. Jeśli odczyt czasu zegara RTC jest przed ramą czasową, pierwszy z wyjściowych bitów jest włączany; Jeśli czas RTC mieści się w przedziale czasowym, drugi z wyjściowych bitów jest włączany; Jeśli czas RTC jest po przedziale czasowym, trzeci z wyjściowych bitów jest włączany.
3. Dane wejściowe zegara początkowe i końcowe (bez roku, miesiąca i dnia) są przechowywane w 5 kolejnych komórkach rejestru D, użytkownik może również odczytać z czwartej komórki rejestru D danych zegara.
4. Zegar końcowy przedziału czasowego musi być późniejszy niż zegar początkowy przedziału czasowego. Na przykład rama czasowa taka jak [23:59:59 sobota, 00:00:00 niedziela] jest niedozwolona





Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(S)	Adres pierwszej komórki D początku przedziału czasowego (bez roku, miesiąca i dnia)	D	-	-
(E)	Adres pierwszej komórki D końca przedziału czasowego (bez roku, miesiąca i dnia)	D	-	-
(OUT)	Adres pierwszej komórki do danych wyjściowych	Y/M/S	0/1	Wartość logiczna

Przykład:

Lista instrukcji:

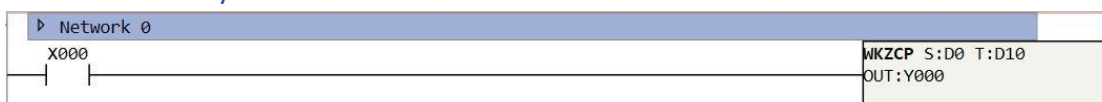
### Network 000

LD X000

WKZCP D0 Y0 //porównaj dane (bez roku, miesiąca i dnia) z zegara RTC z D0~D4 z danymi zegara RTC w D10~D14 i wyprowadź wynik do Y0~Y2

POP

Schemat drabinkowy:



## CKCMP

1. Ta instrukcja porównuje dane zegara RTC z określonymi danymi czasu, które obejmują tylko godzinę, minutę i sekundę.
2. Ta instrukcja odczytuje dane RTC, porównuje je z danymi czasowymi przechowywanymi w określonym ciągu 3 komórek rejestru D (od czwartego do szóstego słowa całych danych zegara) i wyprowadza wynik do określonych wyjść bitowych. Jeśli czas odczytany jest wcześniejszy niż czas RTC, pierwszy z wyjściowych bitów jest włączany; Jeśli czas odczytany jest taki sam jak czas RTC, z wyjściowych bitów jest włączany; Jeśli czas odczytany jest późniejszy niż czas RTC, trzeci z wyjściowych bitów jest włączany.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna



(T)	Adres pierwszej komórki D przechowującej dane zegara	D	-	-
(OUT)	Adres pierwszej komórki do danych wyjściowych	Y/M/S	0/1	Wartość logiczna

Przykład:

Lista instrukcji:

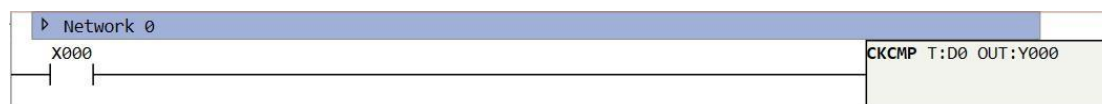
### Network 000

LD X000

CKCMP D0 Y0 //porównaj czas D0~D2 z czasem zegara RTC i wyprowadź wynik do Y0~Y2

POP

Schemat drabinkowy:



## CKZCP

1. Porównuje dane zegara RTC z określonym przedziałem czasowym, które obejmują tylko godzinę, minutę i sekundę.
2. Ta instrukcja odczytuje dane RTC (które obejmują tylko godzinę, minutę i sekundę), porównuje je z przedziałem czasowym składającym się z zegara początkowego (które obejmują tylko godzinę, minutę i sekundę) i zegara końcowego (które obejmują tylko godzinę, minutę i sekundę) i wyprowadzają wynik do określonych bitów. Jeśli odczyt czasu zegara RTC jest przed ramą czasową, pierwszy z wyjściowych bitów jest włączany; Jeśli czas RTC mieści się w przedziale czasowym, drugi z wyjściowych bitów jest włączany; Jeśli czas RTC jest po przedziale czasowym, trzeci z wyjściowych bitów jest włączany.
3. Dane wejściowe zegara początkowe i końcowe (które obejmują tylko godzinę, minutę i sekundę) są przechowywane w 3 kolejnych komórkach rejestru D, użytkownik może również odczytać z 4-6 komórki rejestru D danych zegara.
4. Zegar końcowy przedziału czasowego musi być późniejszy niż zegar początkowy przedziału czasowego. Na przykład rama czasowa taka jak [23:59:59, 00:00:00] jest niedozwolona

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście, które włącza lub wyłącza polecenie		0/1	Wartość logiczna
(S)	Adres pierwszej komórki D początku przedziału czasowego (bez roku, miesiąca i dnia)	D	-	-



(E)	Adres pierwszej komórki D końca przedziału czasowego (bez roku, miesiąca i dnia)	D	-	-
(OUT)	Adres pierwszej komórki do danych wyjściowych	Y/M/S	0/1	Wartość logiczna

Przykład:

Lista instrukcji:

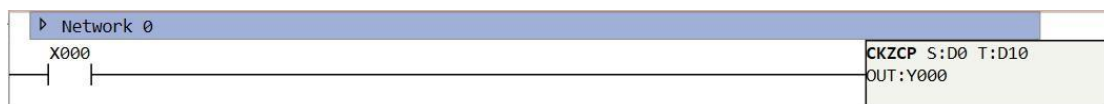
Network 000

LD X000

CKZCP D0 D10 Y0 //porównaj dane (godziny, minuty i sekundy) z zegara RTC z D0~D2 z danymi zegara RTC w D10~D12 i wyprowadź wynik do Y0~Y2

POP

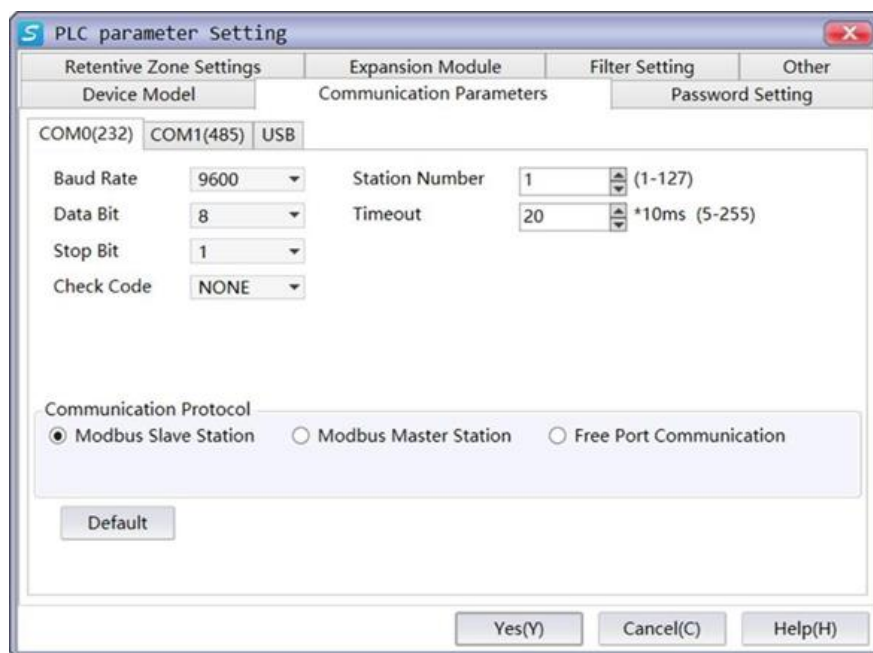
Schemat drabinkowy:



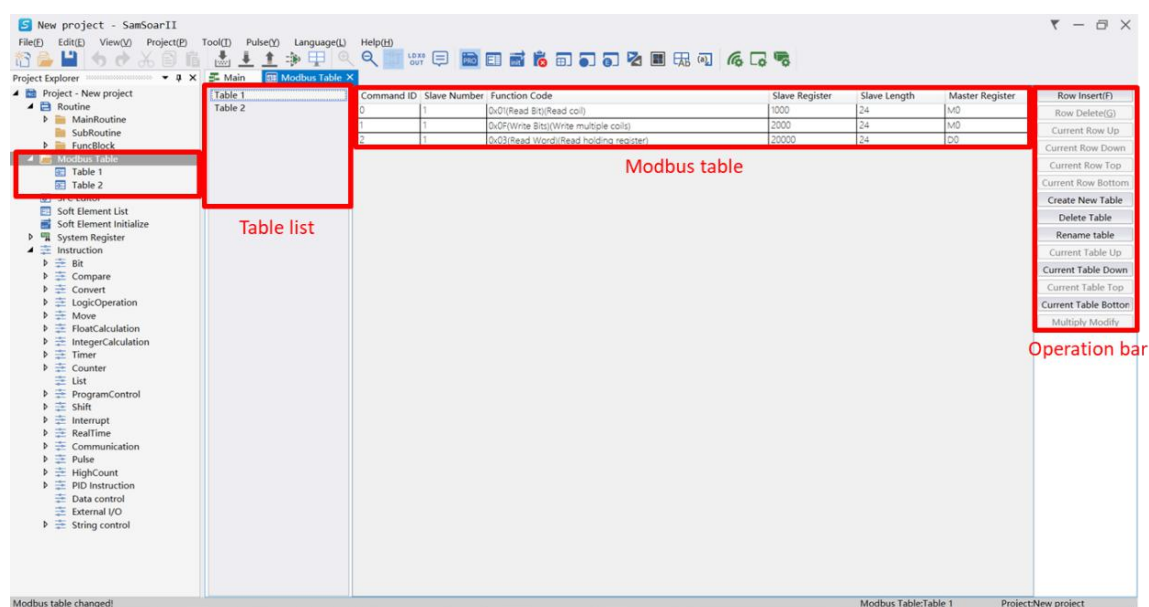


## KOMUNIKACJA

Samkoon PLC obsługuje protokół komunikacyjny **Modbus**, który można zastosować do portów szeregowych, takich jak **RS232**, **RS485** i **USB**; **Modbus** obsługuje maksymalnie 127 stacji slave (podrzędnych). Użytkownik może ustawić parametry komunikacji w ustawieniach parametrów PLC, jak pokazano na rysunku poniżej.



W celu korzystania z komunikacji Modbus, użytkownik musi najpierw utworzyć tabelę Modbus przed użyciem instrukcji komunikacji. Jak pokazano na rysunkach poniżej, użytkownik może wejść do tabeli Modbus z eksploratora projektu i edytować tabelę za pomocą paska operacji.





Row Insert(F)	Wstaw nowy wiersz poniżej
Row Delete(G)	Usuń wybrany wiersz
Current Row Up	Przesuń wybrany wiersz o jeden wiersz wyżej
Current Row Down	Przesuń wybrany wiersz o jeden wiersz niżej
Current Row Top	Przesuń wybrany wiersz na początek tabeli
Current Row Bottom	Przesuń wybrany wiersz na koniec tabeli
Create New Table	Utwórz nową tabelę Modbus
Delete Table	Usuń bieżącą tabelę Modbus
Rename table	Zmień nazwę bieżącej tabeli Modbus
Current Table Up	Przesuń bieżącą tabelę Modbus o 1 pozycję w górę
Current Table Down	Przesuń bieżącą tabelę Modbus o 1 pozycję w dół
Current Table Top	Przesuń bieżącą tabelę Modbus na początek listy
Current Table Bottom	Przesuń bieżącą tabelę Modbus na koniec listy
Multiply Modify	Modyfikuj wiele wybranych wierszy

Tabele Modbus obsługują kody funkcji 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07, 0x08, 0x0f i 0x10, ich funkcje i obsługiwane typy rejestrów zawarte są w tabeli poniżej.

Kod funkcji	Funkcja	Obsługiwane rejestry
0x01	Odczyt wyjść binarnych	X/Y/M/S/T/C
0x02	Odczyt wejść binarnych	X
0x03	Odczytaj przytrzymany rejestr	D/TV/CV/CV32/AI/AO
0x04	Odczyt rejestru typu holding	-
0x05	Zapis pojedynczego wyjścia binarnego	Y/M/S/T/C
0x06	Zapis jednego rejestru	D/TV/CV/CV32/AI/AO
0x0f	Zapis wielu wyjść binarnych	Y/M/S/T/C
0x10	Zapis wielu rejestrów	D/TV/CV/CV32/AI/AO

Poprzez ustawienie rejestru slave, długości rejestru slave i rejestru master (nadrzędnego) w tabeli Modbus, dane o określonej długości mogą być odczytywane z rejestrów slave i przechowywane w rejestrze master lub kopiowane z rejestrów master i zapisywane w rejestrach slave. Adres rejestru slave jest oznaczony numerem przesunięcia, Tabela poniżej pokazuje numer przesunięcia różnych typów rejestrów.

Typ rejestru	Zakres	Typ danych	Odczyt kod funkcji	Napisz kod funkcji	Numer offsetu
--------------	--------	------------	--------------------	--------------------	---------------



X	0~127	bit	0x0x/0x02	0x05	0
EX	0~511	bit	0x01/0x02	0x05	512
Y	0~127	bit	0x01	0x05/0x0f	10000
EY	0~511	bit	0x01	0x05/0x0f	10512
M	0~8191	bit	0x01	0x05/0x0f	30000
S	0~999	bit	0x01	0x05/0x0f	50000
T	0~255	bit	0x01	0x05/0x0f	60768
C	0~255	bit	0x01	0x05/0x0f	60512
D	0~8191	WORD	0x03/0x04	0x0x6/0x10	40000
CV	0~199	WORD	0x0x3	0x0x6/0x10	60000
TV	0~255	WORD	0x03	0x0x6/0x10	60256
CV32	200~255	DWORD	0x03	0x0x6/0x10	61024
AI	0~31	WORD	0x03	0x0x6/0x10	20000
EAI	0~79	WORD	0x03	0x0x6/0x10	20100
AO	0~31	WORD	0x03	0x0x6/0x10	20512
EAO	0~79	WORD	0x03	0x0x6/0x10	20612

**EX, EY, EAI, EAO** to rejestry modułu rozszerzenia. Dla rejestru typu CV32, każde 2 kolejne numery przesunięcia tworzą dane 32-bitowe, na przykład: numer przesunięcia 61024 jest młodszym słowem CV200, numer przesunięcia 61025 jest starszym słowem CV200.

## MBUS

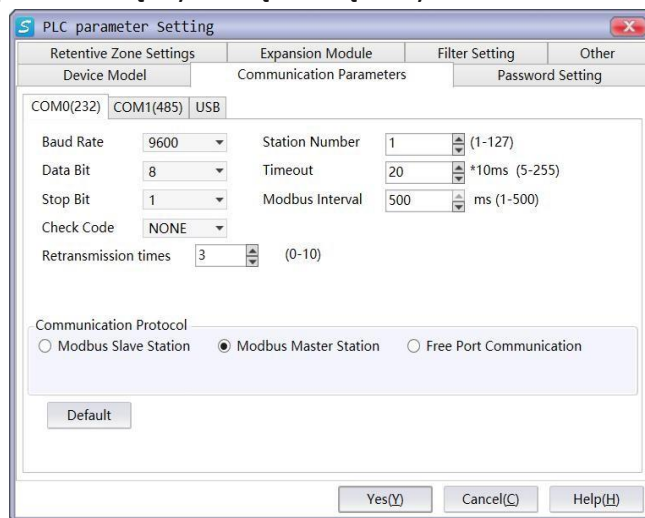
1. Ta instrukcja jest używana do komunikowania się stacji master (nadrzędnej) ze stacją slave (podrzedną) za pomocą tabeli Modbus .

2. Przed użyciem tej instrukcji użytkownik musi ustawić parametry komunikacyjne portu szeregowego, jak pokazano na rysunku poniżej.

- Numer stacji: numer stacji master lub slave, urządzenia zawarte w tej samej sieci komunikacyjnej muszą mieć ustawione różne numery stacji.
- Szybkość transmisji (baud rate): częstotliwość sygnału danych, jednostką jest bod na sekundę, można ustawić na 1200, 2400, 4800, 9600, 19200, 38400, 57600 lub 115200. Szybkość transmisji urządzenia master i slave musi być taka sama.
- Bit danych (data bit): długość sygnału danych każdej ramki danych.
- Bit stopu (stop bit): długość sygnału stopu, oznacza koniec jednej ramki danych.



- Bit kontrolny (check bit): bit, który znajduje się za bitem danych, może być ustawiony na zero, parzystość lub nieparzystość. Jeśli żaden, nie ma bitu kontrolnego; Jeśli parzysty, bit kontrolny wynosi 1, jeśli jest parzysta liczba wysokiego poziomu sygnału w bicie danych; Jeśli nieparzysty, bit kontrolny wynosi 1, jeśli istnieje liczba nieparzysta na wysokim poziomie sygnałów w bicie danych.
- Czasy przekroczenia limitu czasu i retransmisji: jeśli urządzenie slave nie zwróci wiadomości dla ustawień limit czasu, urządzenie master ponownie wyśle dane, zresetuje limit czasu i zaczeka na wiadomość zwrotną od urządzenia slave. Jeśli osiągną ustawione czasy retransmisji, master zgłosi kod błędu 0x10 w określonym rejestrze danych.
- Interwał Modbus: odstęp czasu między każdą ramką danych.



3. Ta instrukcja przypisuje rejestr D do przechowywania informacji komunikacji, które rejestrują status i błąd komunikacji Modbus. Tabela poniżej pokazuje kod informacji i odpowiadające mu znaczenie. Ta instrukcja przypisuje również rejestr D do przechowywania identyfikatora bieżącego polecenia z tabeli Modbus, będzie się on zwiększał lub zapętlął, gdy wykonywane jest odpowiednie polecenie, użytkownik może również zapisać go, aby kontrolować, które polecenie ma zostać wykonane.

Kod	Opis
0x01	Niedozwolony kod funkcji
0x02	Niedozwolony adres
0x03	Niedozwolone dane
0x04	Operacja slave nie powiodła się
0x05	Przetwarzanie poleceń
0x06	Stacja Slave jest zajęta
0x07	Niedozwolony typ danych
0x08	Nielegalne polecenie ID
0x09	Niedozwolone hasło
0x10	Przekroczono limit czasu komunikacji
0x12	Błąd ustawień stacji Master/Slave



0x13	Numer Slave jest taki sam jak numer Master
0x14	Adres rejestru przekracza limit
0x15	Wykonanie polecenia nie powiodło się
0x18	Błąd otrzymanej ramki informacyjnej (błąd długości, błąd kontroli CRC)
0x20	Parametr jest niemodyfikowalny
0x21	Parametr jest niemodyfikowalny podczas pracy
0x22	Parametr jest chroniony hasłem

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające		0/1	Wartość logiczna
(COM)	Wybrany port szeregowy do komunikacji.	-	-	-
(TBL)	Wybrana tabela Modbus.	-	-	-
(WR)	Rejestr danych do przechowywania kodu informacji o komunikacji.	D		Kod binarny
(WR_ID)	Rejestr danych do przechowywania bieżącego identyfikatora polecenia z tabeli Modbus.	D		16-bitowa liczba bez znaku

1. Ustawienie limitu czasu powinno być dostosowane do szybkości transmisji, jeśli szybkość transmisji jest niska, wolny przepływ danych może wymagać dłuższego czasu oczekiwania. Dodatkowo, jeśli rozmiar danych do przesłania jest duży, transmisja może również wymagać dłuższego czasu oczekiwania.
2. Gdy do komunikacji Modbus używany jest port szeregowy, nie można go używać do innych celów takie jak pobieranie/wysyłanie, monitorowanie i komunikacja na wolnym porcie.
3. Jeśli jedno urządzenie slave komunikacji Modbus zostanie odłączone, urządzenie master będzie nadal odwoływać się do niego i spowoduje to opóźnienie w komunikacji. Aby uzyskać szybszą reakcję, użytkownik może ustawić krótszy czas oczekiwania i krótszy czas retransmisji.

### Przykład

W tym przykładzie zastosowano 3 tabele Modbus, a pojedyncza tabela jest przeznaczona dla jednej stacji slave. Komunikacja będzie przebiegać w pętli między 3 stacjami slave, a użytkownik może również wybrać, z którą stacją slave ma się komunikować, wpisując numer podrzędny w **D1234**. Tabela Modbusa:





Slave 1	Command ID	Slave Numb	Function Code	Slave Register	Slave Length	Master Register
Slave 2	0	1	0x03(Read Word)(Read holding register)	60200	1	D0
Slave 3	1	1	0x01(Read Bit)(Read coil)	30000	10	M0
	2	1	0x03(Read Word)(Read holding register)	40000	10	D100
Slave 1	Command ID	Slave Numb	Function Code	Slave Register	Slave Length	Master Register
Slave 2	0	2	0x03(Read Word)(Read holding register)	60200	1	D200
Slave 3	1	2	0x01(Read Bit)(Read coil)	30000	10	M200
	2	2	0x03(Read Word)(Read holding register)	40000	10	D300
Slave 1	Command ID	Slave Numb	Function Code	Slave Register	Slave Length	Master Register
Slave 2	0	3	0x03(Read Word)(Read holding register)	60200	1	D400
Slave 3	1	3	0x01(Read Bit)(Read coil)	30000	10	M200
	2	3	0x03(Read Word)(Read holding register)	40000	10	D500

## Lista instrukcji:

## Network 000

LDP M8183

MPS

AW= D1234 K1

AW= D5001 K2

MOV K2 D1234 //przejdź do slave 2 po zakończeniu komunikacji slave 1

MOV K0 D5003 //zainicjuj polecenie ID dla slave 2

MRD

AW= D1234 K2

AW= D5003 K2

MOV K3 D1234 //przejdź do slave 3 po zakończeniu komunikacji slave 2

MOV K0 D5005 //zainicjuj polecenie ID dla 3

MRD

AW= D1234 K3

AW= D5005 K2

MOV K3 D1234 //idź do slave 1 po zakończeniu komunikacji slave 3

MOV K0 D5001 //zainicjuj polecenie ID dla slave 1

MPP

RST M8183 K1

POP

LD M6000

MPS

AW= D1234 K1

MBUS K1 Slave 1 D5000 D5001 //zezwól na komunikację Modbus dla slave 1

MRD

AW= D1234 K1

MBUS K1 Slave 2 D5002 D5003 // zezwól na komunikację Modbus dla slave 2

MRD

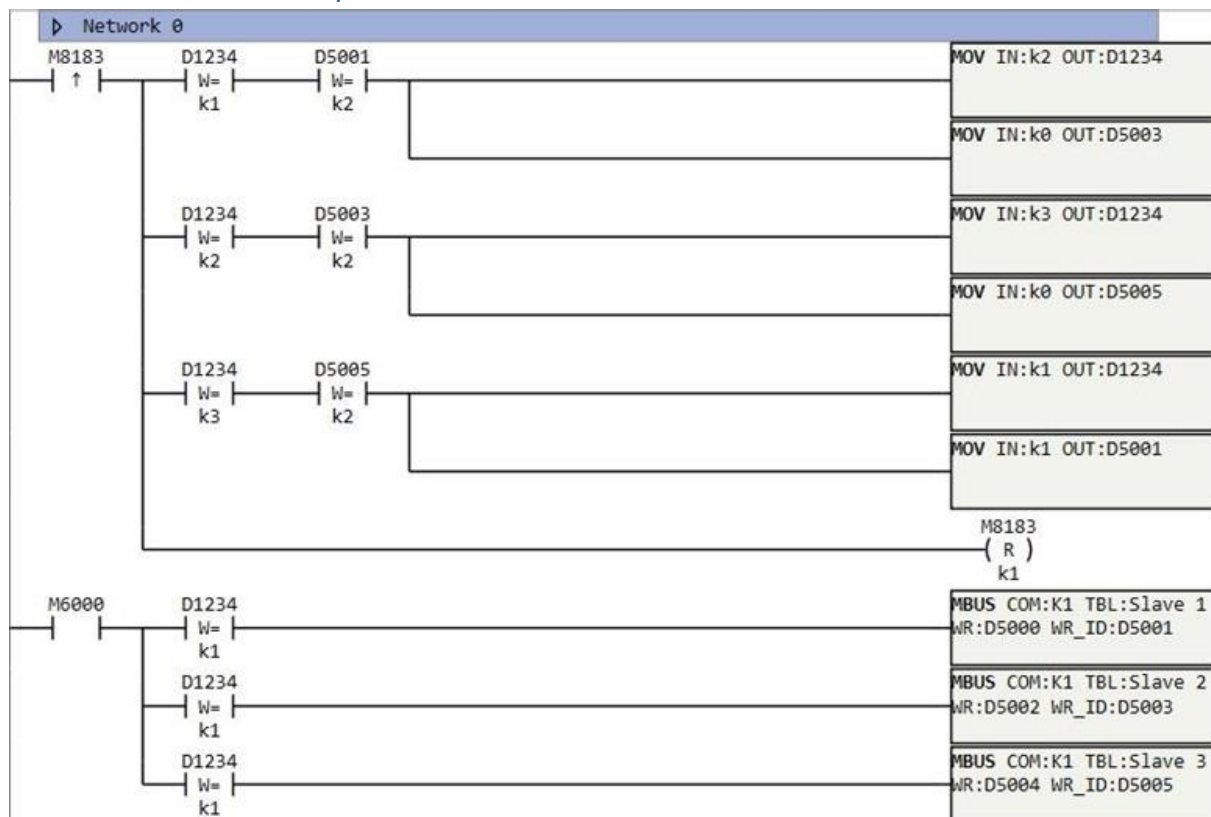
AW= D1234 K1

MBUS K1 Slave 3 D5004 D5005 // zezwól na komunikację Modbus dla slave 3

POP



## Schemat drabinkowy:



## SEND

1. Niniejsza instrukcja dotyczy komunikacji na wolnym porcie, przesyła dane w podanych Rejestrach D (ADDR) o określonej długości (LEN) do określonego portu szeregowego (COM).
2. Istnieją systemowe rejestry specjalne **M**, które rejestrują stan transmisji odpowiednich portów szeregowych (zamieszczone w tabeli poniżej).
3. Istnieje dodatkowy parametr różniący się od komunikacji Modbus: bit bufora. Gdy bit bufora jest wybierany za pomocą 8, transmisja odczytuje tylko niski bajt rejestrów D; Gdy bufor jest wybrany za pomocą 16, transmisja odczytuje i zapisuje wysoki bajt i niski bajt Rejestru D. Użytkownik może ustawić ten parametr w ustawieniach parametrów komunikacji.

Rejestr	Opis
<b>M8176</b>	Włącza się, gdy port szeregowy COM0 przesyła dane.
<b>M8177</b>	Włącza się, gdy port szeregowy COM1 przesyła dane.
<b>M8180</b>	Włącza się po zakończeniu transmisji portu szeregowego COM0.
<b>M8181</b>	Włącza się po zakończeniu transmisji portu szeregowego COM1.



Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające		0/1	Wartość logiczna
(COM)	Wybrany port szeregowy do wysyłania danych.	-	-	-
(ADDR)	Pierwsza komórka rejestru danych do odczytu	D	-	-
(LEN)	Długość danych do odczytania i wysłania	K/H/D	0~512	16-bitowa liczba całkowita bez znaku

1. Użytkownik musi wybrać protokół komunikacyjny jako wolny port komunikacji w ustawieniach parametrów komunikacji.
2. Gdy port szeregowy jest używany do komunikacji z wolnym portem, nie można go używać do innych celów takie jak pobieranie/wysyłanie, monitorowanie i komunikacja Modbus.
3. Modyfikacja parametrów komunikacji zaczyna obowiązywać po ponownym pobraniu.
4. Komunikacja na wolnym porcie jest półdupleksowa, więc instrukcje SEND i REV nie mogą działać w tym samym momencie.

#### Przykład

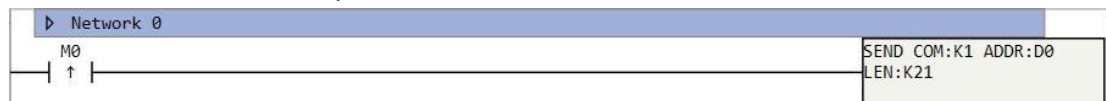
#### Lista instrukcji:

#### Network 000

LDP M0

SEND K1 D0 K21 //wybierz COM1, pierwszy bit danych to D0, długość danych = 21;  
 //jeśli bufor jest 8-bitowy, wszystkie młodsze bajty w D0~D20, 21-bajt w całości zostaną wysłane  
 //jeśli bufor jest 16-bitowy, wszystkie dane w D0~D10 i młodszy bajt D11,  
 21-bajt w całości zostaną wysłane  
 //SEND jest wyzwalany przez narastające zbocze, jeśli jest włączony przez normalne  
 zamkniętą cewkę, SEND wykona się podczas każdego okresu skanowania

#### Schemat drabinkowy:



#### Przykład

#### Lista instrukcji:

#### Network 000

LD M0

MPS

MVBLK D2000 D1 D0 //przenieść blok danych o długości D0 z D2000 do D1

MPP



ANDI M8180 // wyłączyć opcję SEND po zakończeniu transmisji

MEP

SEND K1 D1 D0 // wyślij dane do COM0, gdy M0 jest włączone

Network 001

LD M8180

REV K0 D1001 D1000 // włączyć REV po zakończeniu transmisji

Network 001

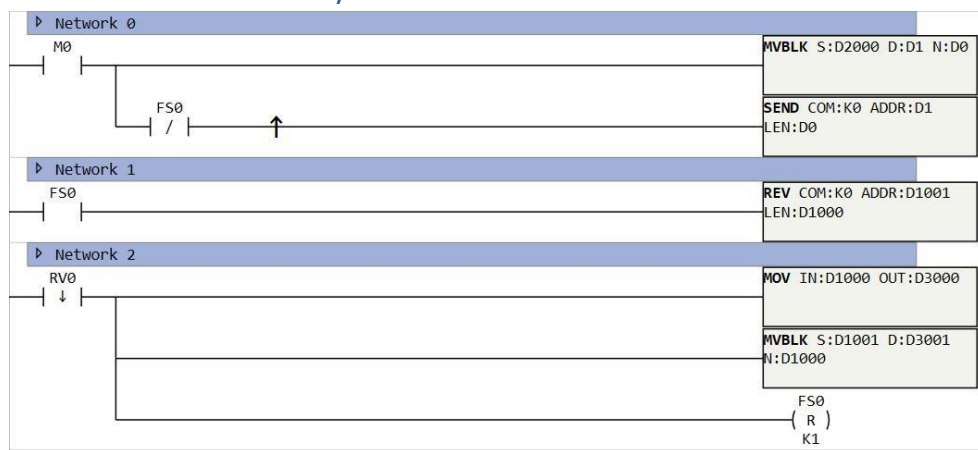
LDF M8178 // zamknij opadającym zboczem po zakończeniu odbioru

MOV D1000 D3000 // przenieś długość danych z D1000 do D3000

MVBLK D1001 D3001 D1000 // przenieś blok danych o długości D1000 z D1001 do D3001

RST M8180 K1 // zresetuj M8180 po zakończeniu transmisji

Schemat drabinkowy:



## REV

1. Ta instrukcja jest stosowana do komunikacji na wolnym porcie, odbiera dane z określonego portu szeregowego (COM), zapisuje dane do określonych rejestrów D (ADDR) i zapisuje długość danych do określonego rejestru D (LEN).
2. Istnieją systemowe rejestry specjalne **M**, które rejestrują stan transmisji odpowiednich portów szeregowych (zamieszczone w tabeli poniżej).
3. Istnieje dodatkowy parametr różniący się od komunikacji Modbus: bit bufora. Gdy bit bufora jest wybierany za pomocą 8, transmisja odczytuje tylko niski bajt rejestrów D; Gdy bufor jest wybrany za pomocą 16, transmisja odczytuje i zapisuje wysoki bajt i niski bajt Rejestru D. Użytkownik może ustawić ten parametr w ustawieniach parametrów komunikacji.

Rejestr	Opis
<b>M8178</b>	Włącza się, gdy port szeregowy COM0 odbiera dane.



<b>M8179</b>	Włącza się, gdy port szeregowy COM1 odbiera dane.
<b>M8182</b>	Włącza się po zakończeniu odbioru danych portu szeregowego COM0.
<b>M8183</b>	Włącza się po zakończeniu odbioru danych portu szeregowego COM1.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające		0/1	Wartość logiczna
(COM)	Wybrany port szeregowy do odbierania danych.	-	-	-
(ADDR)	Pierwsza komórka rejestru danych do odczytu	D	-	-
(LEN)	Długość danych do odebrania i zapisania	D	-	-

1. Użytkownik musi wybrać protokół komunikacyjny jako wolny port komunikacji w ustawieniach parametrów komunikacji.
2. Gdy port szeregowy jest używany do komunikacji z wolnym portem, nie można go używać do innych celów takie jak pobieranie/wysyłanie, monitorowanie i komunikacja Modbus.
3. Modyfikacja parametrów komunikacji zaczyna obowiązywać po ponownym pobraniu.
4. Komunikacja na wolnym porcie jest półduplexowa, więc instrukcje SEND i REV nie mogą działać w tym samym momencie.

#### Przykład

#### Lista instrukcji:

#### Network 000

LDP M0

REV K1 D0 D1000 //wybierzCOM1, pierwszy bit to D0, długość danych = D1000;

Założmy, że COM1 odbiera dane: 0x10 0x11 0x12 0x13 0x14

//jeśli bufer jest 8-bitowy, D1000=5, D0=0x0010, D1=0x0011,

D2=0x0012, D3=0x0013, D4=0x0014

//jeśli bufer jest 16-bitowy, D1000=5, D0=0x1110, D1=0x1312,

D2=0x0014

#### Schemat drabinkowy:



#### Network 000

LD M0

REV K0 D1 D0 //odbierz dane z COM0 gdy włączany jest M0 is ON i zapisz dane w D1

#### Network 001

LDP M8182



SEND K0 D1001 D1000 //włącz **SEND** po zakończeniu odbierania danych

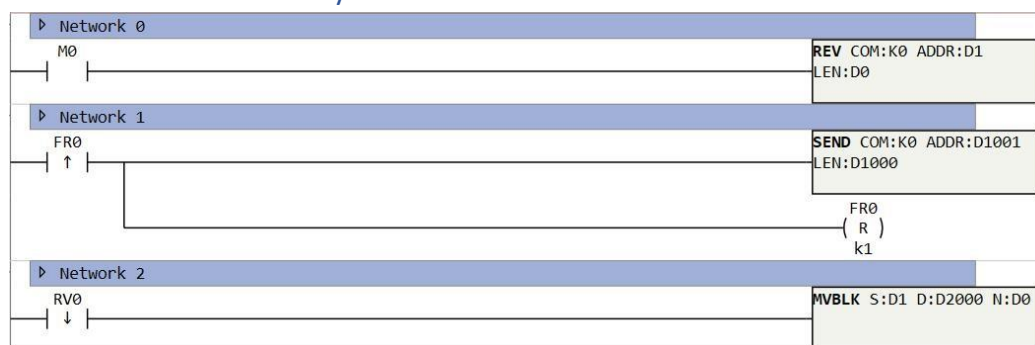
RST M8182 K1 //zresetuj automatycznie M8182

Network 001

LDF M8178 //zamknij opadającym zboczem po zakończeniu odbioru

MVBLK D1 D2000 D0 //przenieść zapisane dane odbioru do D2000

#### Schemat drabinkowy:



## CRC

1. Ta instrukcja oblicza sumę kontrolną w funkcji CRC (MODE) z określoną długością danych (LEN) przechowywanych w określonych rejestrach danych (DATA).
2. Jednostką długości danych jest bajt, dane muszą być zapisane w D-rejestrach. Należy zwrócić uwagę, że wysoki bajt (high byte) Rejestru D znajduje się po prawej stronie, niski (low byte) po lewej.
3. Wygenerowana suma kontrolna jest również nazywana bitem kontrolnym i będzie przechowywana ściśle po danych pierwotnych, te 2 części razem tworzą kod CRC (cykliczny kod nadmiarowy).
4. Tryby CRC zawarte są w tabeli.

tryb CRC	Numer	Wielomian	Długość kodu
CRC-16	K0	$x^{16}+x^{15}+x^{12}+1$	2 bajty
CRC-12	K1	$x^{12}+x^{11}+x^3+x^2+x+1$	2 bajty
CRC-8	K2	$x^8+x^2+x+1$	1 bajt
CRC-CCITT	K3	$x^{16}+x^{12}+x^5+1$	2 bajty
CRC-32	K4	$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$	4 bajty
CRC-32C	K5	$x^{32}+x^{28}+x^{27}+x^{26}+x^{25}+x^{23}+x^{22}+x^{20}+x^{19}+x^{18}+x^{14}+x^{13}+x^{11}+x^{10}+x^9+x^8+x^6+1$	4 bajty

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
---------------------	------	---------	--------	------------



(EN)	Wejście załączające/wyłączające		0/1	Wartość logiczna
(DATA)	Rejestry danych przechowują dane początkowe.	D	-	kod binarny
(LEN)	Długość danych pierwotnych.	K/D	-	16-bitowa liczba całkowita bez znaku
(MODE)	Tryb CRC do obliczania sumy kontrolnej.	D	0~5	16-bitowa liczba całkowita bez znaku

### Przykład

#### Lista instrukcji:

#### Network 000

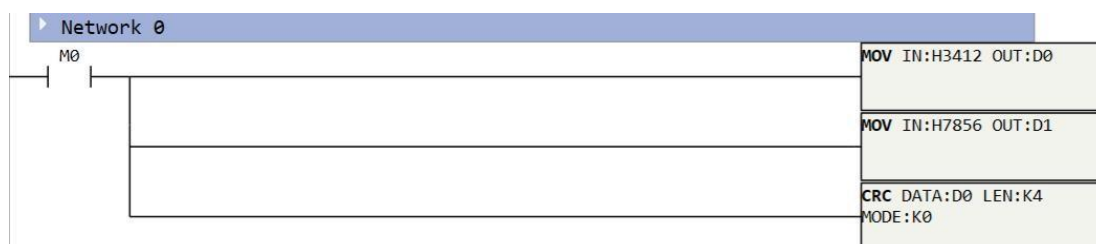
LD M0

MOV H3412 D0

MOV H7856 D1

CRC D0 K4 K0 //wybierz CRC-16, długość danych to 4 bajty, oblicz CRC z 0x12345678 i zapisz wynik 0x107B w D2

#### Schemat drabinkowy:



## CRCHECK

1. Ta instrukcja sprawdza kod CRC w określonym trybie CRC (MODE) i wysyła wynik do określonego rejestru bitowego (RES), pierwotne dane po określonej długości (LEN) zaczynają się od określonego rejestru danych (DATA).
2. Kod CRC musi być przechowywany w rejestrach D. Należy zwrócić uwagę, że wysoki bajt (high byte) Rejestru D znajduje się po prawej stronie, niski (low byte) po lewej.
3. Bit wyjściowy jest włączony, gdy kontrola powiodła się, jest wyłączony, gdy kontrola nie została zaliczona.
4. Tryby CRC zawarte są w tabeli:

tryb CRC	Numer	Wielomian	Długość kodu
CRC-16	K0	$x^{16}+x^{15}+x^{12}+1$	2 bajty
CRC-12	K1	$x^{12}+x^{11}+x^3+x^2+x+1$	2 bajty
CRC-8	K2	$x^8+x^2+x+1$	1 bajt



CRC-CCITT	K3	$x^{16}+x^{12}+x^5+1$	2 bajty
CRC-32	K4	$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$	4 bajty
CRC-32C	K5	$x^{32}+x^{28}+x^{27}+x^{26}+x^{25}+x^{23}+x^{22}+x^{20}+x^{19}+x^{18}+x^{14}+x^{13}+x^{11}+x^{10}+x^9+x^8+x^6+1$	4 bajty

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające		0/1	Wartość logiczna
(DATA)	Rejestry danych przechowują dane początkowe.	D	-	kod binarny
(LEN)	Długość danych pierwotnych.	K/D	-	16-bitowa liczba całkowita bez znaku
(MODE)	Tryb CRC do obliczania sumy kontrolnej.	D	0~5	16-bitowa liczba całkowita bez znaku

Przykład

Lista instrukcji:

Network 000

LD M0

MOV H3412 D0

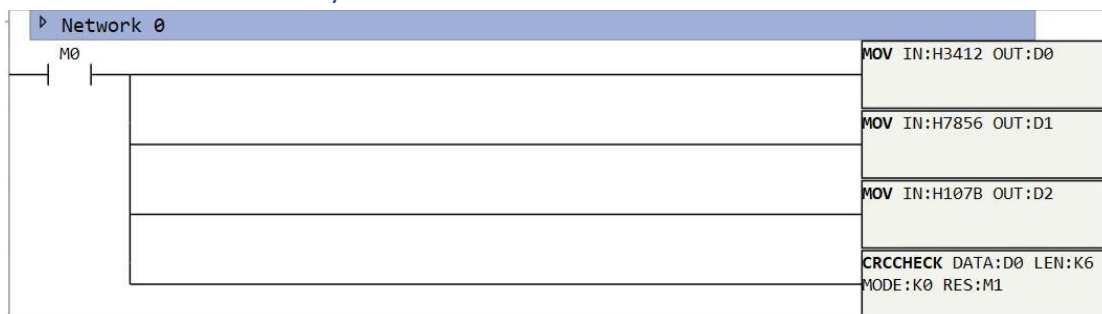
MOV H7856 D1

MOV H107B D1

CRCHECK D0 K6 K0 M1 //dane pierwotne to 0x12345678, suma kontrolna to 0x7B10,  
kontrola jest zaliczona

POP

Schemat drabinkowy:







## INSTRUKCJE IMPULSU

1. Jak dotąd, dla istniejących serii, **FGs\_16MT/FGs\_32MT/FGRB\_C8X8T** posiada **2 szybkie wyjścia impulsowe** (Y0-Y1), a najwyższa częstotliwość na wyjściu impulsowym wynosi 200 kHz; **FGs\_64MT/FGRS\_C8X8T/FGRE\_C8X8T** posiada 4 szybkie wyjścia impulsowe (Y0-Y3), a najwyższa częstotliwość na wyjściu impulsowym wynosi 200 kHz; **FGm\_64MT** posiada 10 szybkich wyjść impulsowych (Y0-Y7, Y10-Y11), a najwyższa częstotliwość na wyjściach impulsowych wynosi 500kHz. Sterownik PLC typu przekaźnikowego nie obsługuje szybkich impulsów.
2. Dane szybkiego wyjścia impulsowego mogą być zliczane tylko za pomocą szybkiego licznika, ale nie licznika wewnętrznego, nie można go również zliczać przez wyzwalanie zboczem punktów Y.
3. W przypadku małego obciążenia, czas otwarcia tranzystora w stanie OFF będzie dłuższy. Gdy w takiej sytuacji potrzebny jest krótki czas, zalecane jest dodać rezystancję obciążenia aby zwiększyć prąd obciążenia.
4. Rejestry zliczania zbiorczego impulsów (D8140~D8158) są ważnymi rejestrami. Są one czytelne i zapisywalne, gdy zostaną zapisane z nową wartością, liczenie będzie kontynuowane na podstawie nowej zapisanej wartości.
5. Wyprowadzany jest czas szybkiego impulsu, wartość w rejestrach zliczania impulsów jest dyskretna i zmienna. Kiedy użytkownicy muszą porównać wartość, można użyć instrukcji mniejszej lub większej niż, ale nie instrukcji równa się.
6. Kiedy punkt wyjściowy jest używany do szybkiego wyjścia impulsowego, nie może być używany jako wspólny punkt wyjściowy, więc nie będzie reagował na operację WŁ. lub WYŁ.
7. Jeśli częstotliwość wyjścia impulsowego jest większa niż 200k, lepiej dodać rezystor podciągający w porcie wyjściowym (zaleca się wybrać rezystancję 1k, gdy napięcie podciągające wynosi 24V).
8. Istnieją specjalne rejestry systemowe, które przechowują informacje o wykonywaniu instrukcji. Użytkownik może modyfikować niektóre z nich. Dla wszystkich instrukcji impulsowych istnieją rejestry rejestrujące kierunek impulsu, liczbę impulsów, przekroczenie licznika impulsów i stan wyjścia impulsowego (szczegóły w tabeli poniżej). Dla wielostopniowych instrukcji impulsowych (np. PLSR, POLYLINE) istnieją rejestry rejestrujące aktualny numer segmentu i numer błędu segmentu (szczegóły w tabeli poniżej). Dla instrukcji interpolacji (np. POLYLINE, LINE, ARC) istnieją rejestry rejestrujące stan wyjściowy (szczegóły w tabeli poniżej).



	Rejestr kierunku impulsu	Rejestr flagi przepełnienia licznika	Rejestr stanu wyjścia impulsowego	Licznik akumulacji impulsów	Port wyjścia impulsowego
Typ danych	Wartość logiczna	Wartość logiczna	Wartość logiczna	32-bitowa liczba całkowita ze znakiem	-
Opis	Pokaż kierunek impulsu odpowiedniego portu wyjściowego, 1 dla dodatniego, 0 dla ujemnego	Pokaż, czy odpowiedni licznik jest przepełniony, 1 dla przepełnienia, 0 dla braku	Pokaż, czy odpowiedni port wyjściowy wysła impuls, 1 dla wyjścia impulsów, 0 dla braku	Zapisz, ile impulsów wysła odpowiedni port wyjściowy	Odpowiadający port wyjścia impulsowego
Rejestr	M_8102 (o)	M_8118 (o)	M_8134 (o)	D_8140 (o/z)	Y000
	M_8103 (o)	M_8119 (o)	M_8135 (o)	D_8142 (o/z)	Y001
	M_8104 (o)	M_8120 (o)	M_8136 (o)	D_8144 (o/z)	Y002
	M_8105 (o)	M_8121 (o)	M_8137 (o)	D_8142 (o/z)	Y003
	M_8106 (o)	M_8122 (o)	M_8138 (o)	D_8144 (o/z)	Y004
	M_8107 (o)	M_8123 (o)	M_8139 (o)	D_8146 (o/z)	Y005
	M_8108 (o)	M_8124 (o)	M_8140 (o)	D_8148 (o/z)	Y006
	M_8109 (o)	M_8125 (o)	M_8141 (o)	D_8150 (o/z)	Y007
	M_8110 (o)	M_8126 (o)	M_8142 (o)	D_8152 (o/z)	Y010
	M_8111 (o)	M_8127 (o)	M_8143 (o)	D_8154 (o/z)	Y011

	Numer rejestru segmentu błędu	Numer rejestru bieżącego segmentu	Port wyjścia impulsowego
Typ danych	16-bitowa liczba całkowita bez znaku	16-bitowa liczba całkowita bez znaku	-



Opis	Pokazuje, w którym segmencie ścieżki wystąpił błąd.	Wyświetla, w którym segmencie ścieżki znajduje się bieżący punkt ruchu	Odpowiadający port wyjścia impulsowego
Rejestr	D_8108 (o)	D_8124 (o)	Y000
	D_8109 (o)	D_8125 (o)	Y001
	D_8110 (o)	D_8126 (o)	Y002
	D_8111 (o)	D_8127 (o)	Y003
	D_8112 (o)	D_8128 (o)	Y004
	D_8113 (o)	D_8129 (o)	Y005
	D_8114 (o)	D_8130 (o)	Y006
	D_8115 (o)	D_8131 (o)	Y007
	D_8116 (o)	D_8132 (o)	Y010
	D_8117 (o)	D_8133 (o)	Y011

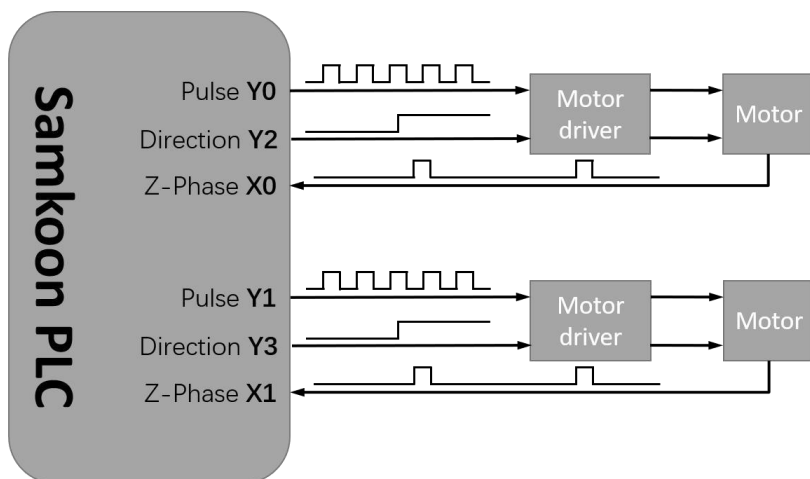
	Rejestr stanu danych wyjściowych platformy interpolacji
Typ danych	Wartość logiczna
Opis	Wyświetl, czy odpowiednia platforma wysyła, 1 wysyła, 0 nie wysyła.
Rejestr	M_8144 (1-wszy) (o)
	M_8145 (2-gi) (o)
	M_8146 (3-ci) (o)
	M_8147 (4-ty) (o)
	M_8148 (5-ty) (o)

## PLSF

1. Instrukcja PLSF to instrukcja zmiennej częstotliwości na wyjściach impulsowych PLC; nie ma gładkiego przejścia podczas transformacji częstotliwości.



2. Gdy instrukcja jest włączona, impuls rozpocznie się z ustawioną częstotliwością, a częstotliwość można regulować w czasie rzeczywistym, pulsowanie pracuje do momentu wyłączenia funkcji.
- Gdy częstotliwość jest ustawiona na wartość większą niż maksymalna, będzie działać jako maksymalna.
3. Niniejsza instrukcja nie obejmuje wyjścia portu kierunkowego pulsowania, jeśli istnieje potrzeba wyjścia kierunkowego, użytkownik może zdefiniować dowolny port wyjściowy jako wyjście kierunkowe. Pokazane na rysunku poniżej, Y2, Y3 i faza Z są definiowane przez użytkownika.



Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(F)	Częstotliwość impulsu, jednostką jest [Hz].	K/H/D	■ FGs/FGRB/FGRE/FGRS 0~200000 ■ FGm 0~500000	32-bitowa liczba całkowita bez znaku
(OUT)	Wyjście impulsu.	Y	-	Wartość logiczna

1. Parametrem częstotliwości impulsów PLSF jest 32-bitowa liczba całkowita, która zajmuje dwa rejestry D.
2. Ta instrukcja wyprowadza impuls bez kierunku, więc licznik akumulacji impulsów (D8140-D8158) może liczyć tylko w kierunku dodatnim.

#### Przykład

#### Lista Instrukcji

##### Network 000

LDP M8151 // włączone, gdy PLC pracuje, musi być wyzwalane zboczem narastającym, aby uniknąć powtarzania zadania

MOVD K400 D0 //pierwotna częstotliwość

##### Network 001

LDP M1

MOVD K900 D0 // zmienna częstotliwość zapisana w D0D1

**Network 002**

LDP M2

MOVD K650 D0 // zmienna częstotliwość zapisana w D0D1

**Network 003**

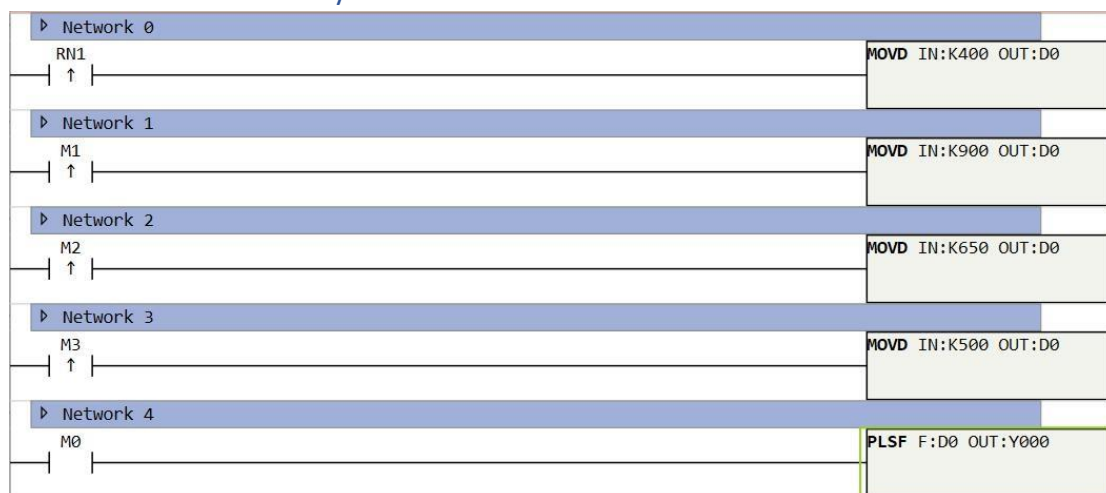
LDP M3

MOVD K500 D0 // zmienna częstotliwość zapisana w D0D1

**Network 004**

LD M0

PLSF D0 Y000 // gdy M0 jest włączone, wyjście Y0 wysyła impuls z częstotliwością zapisaną w D0D1 i zatrzymuje wysyłanie impulsów, gdy M0 jest wyłączone.

**Schemat drabinkowy:****PWM**

1. Instrukcja PWM to instrukcja zmiany częstotliwości impulsów i zmiany współczynnika wypełnienia; nie ma gładkiego przejścia podczas transformacji częstotliwości.
2. Gdy instrukcja jest włączona, impulsy zostaną wysyłane z ustawioną częstotliwością i współczynnikiem wypełnienia, częstotliwość (frequency) i cykl pracy (duty cycle) można regulować w czasie rzeczywistym, pulsowanie zatrzymuje się, aż instrukcja zostanie wyłączona.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające instrukcję		0/1	Wartość logiczna



(F)	Częstotliwość impulsu, jednostką jest [Hz].	K/H/D	<b>■ FGs/FGRB/FGRE/FGRS</b> 0~200000 <b>■ FGm</b> 0~500000	32-bitowa liczba całkowita bez znaku
(DC)	Cykl pracy impulsu, współczynnik zajmowany przez szerokość sygnału w cyklu impulsu.	K/H/D	0~100	8-bitowa liczba całkowita bez znaku
(OUT)	Wyjście impulsu.	Y	-	Wartość logiczna

1. Parametr częstotliwości impulsów PWM to 32-bitowa liczba całkowita, która zajmuje dwa rejestry D.
2. Ta instrukcja wyprowadza impuls bez kierunku, więc licznik akumulacji impulsów (D8140-D8158) może liczyć tylko w kierunku dodatnim.
3. Zniekształcenie częstotliwości pojawi się, gdy impuls jest bardzo szybki, lepiej nie ustawiać zbyt wysokiej częstotliwości impulsu, gdy istnieje potrzeba precyzyjnej kontroli szerokości impulsu.

#### Przykład

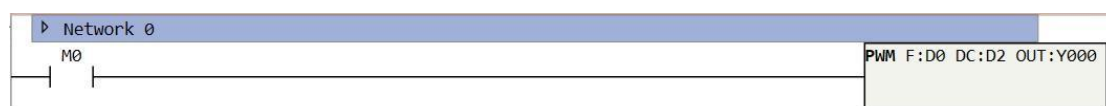
#### Lista Instrukcji

#### Network 000

LD M0

PWM D0 D2 Y000 // Kiedy M0 jest włączone, wyjście Y0 wysyła impuls z częstotliwością zapisaną w D0D1 i współczynnikiem wypełnienia zapisanym w D2D3

Schemat drabinkowy:



## PWMS

1. Instrukcja **PWMS** to instrukcja wyprowadzania impulsu ze zmiennym okresem impulsu i zmienną szerokością impulsu.
2. Gdy instrukcja jest włączona, impuls będą wysyłane z ustawionym okresem i szerokością impulsu, okres i szerokość impulsu można regulować w czasie rzeczywistym, pulsowanie pracuje do momentu wyłączenia funkcji.
3. Jednostki czasu tej instrukcji są wybieralne: milisekundy i mikrosekundy. Ponieważ przy dużej prędkości impulsu może wystąpić zniekształcenie kształtu fali, system automatycznie ograniczy szerokość impulsu po wybraniu jednostki czasu w postaci mikrosekund, na przykład, gdy szerokość impulsu jest ustawiona na 1us, system dostosuje ją do 2us, aby zachować kształt fali.



4. Ta instrukcja udostępnia rejestr **M8062** do sekwencji wyjściowej poziomu transformacji, ten rejestr decyduje o wyprowadzeniu poziomu wysokiego lub niskiego na początku w cyklu impulsowym.

Gdy **M8062** jest włączony, najpierw poziom skuteczny, gdy **M8062** jest wyłączony, najpierw poziom nieefektywny. Należy zauważyć, że **M8062** należy ustawić przed włączeniem funkcji **PWMS**, zmiana stanu **M8062** podczas pracy nie przyniesie efektu.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(U)	Jednostka czasu parametru okresu impulsu i parametru szerokości impulsu, 1 dla milisekundy i 0 dla mikrosekundy.	K/H	0~1	16-bitowa liczba całkowita bez znaku
(P)	Okres impulsu, jednostką jest milisekunda lub mikrosekunda.	K/H/D	<ul style="list-style-type: none"> <li><b>milisekunda</b> 2~65535</li> <li><b>mikrosekunda</b> 5~65535</li> </ul>	16-bitowa liczba całkowita bez znaku
(DP)	Szerokość impulsu, jednostką jest milisekunda lub mikrosekunda	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku
(OUT)	Wyjście impulsu	Y	-	Wartość logiczna

1. Parametr cyklu impulsu i parametr szerokości impulsu PWMS to 16-bitowa liczba całkowita, która zajmuje tylko jeden rejestr D.
2. Ta instrukcja wyprowadza impuls bez kierunku, więc licznik akumulacji impulsów (D8140-D8158) może liczyć tylko w kierunku dodatnim.
3. Zniekształcenie częstotliwości pojawi się, gdy impuls jest bardzo szybki, lepiej nie ustawiać zbyt wysokiej częstotliwości impulsu, gdy istnieje potrzeba precyzyjnej kontroli szerokości impulsu.

#### Przykład

#### Lista instrukcji

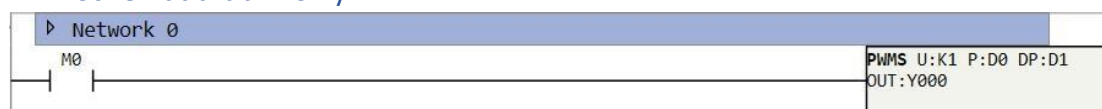
#### Network 000

LD M0

PWMS K0 D0 D1 Y000 // Kiedy M0 jest włączone, Y0 wysyła impuls z cyklem impulsu (ms) zapisanym w D0 i szerokością impulsu (ms) zapisaną w D1

POP

#### Schemat drabinkowy





## PLSY

1. Ta instrukcja wyprowadza określoną ilość impulsów o zmiennej częstotliwości.
2. Gdy instrukcja PLSY jest włączona, impulsy będą wysyłane z ustawioną częstotliwością, a częstotliwość można regulować w czasie rzeczywistym, pulsowanie pracuje do momentu wyłączenia funkcji. Gdy liczba wyprowadzanych impulsów osiągnie docelową liczbę impulsów, pulsowanie zostanie zatrzymane.
3. Niniejsza instrukcja nie obejmuje wyjścia portu kierunkowego pulsowania, jeśli istnieje potrzeba wyjścia kierunkowego, użytkownik może zdefiniować dowolny port wyjściowy jako port kierunkowy. (Patrz instrukcja [PLSF](#))

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające instrukcję		0/1	Wartość logiczna
(F)	Częstotliwość impulsu, jednostką jest [Hz].	K/H/D	<b>■ FGs/FGRB/FGRE/FGRS</b> 0~200000 <b>■ FGm</b> 0~500000	32-bitowa liczba całkowita bez znaku
(P)	Docelowa liczba impulsów.	K/H/D	0~4294967296	32-bitowa liczba całkowita bez znaku
(OUT)	Wyjście impulsu.	Y	-	Wartość logiczna

1. Parametr częstotliwości i docelowa liczba impulsu PLSY są 32-bitowymi liczbami całkowitymi, które zajmują dwa rejestry D.
2. Ta instrukcja wyprowadza impuls bez kierunku, więc licznik akumulacji impulsów (D8140-D8158) może liczyć tylko w kierunku dodatnim.

### Przykład

### Lista instrukcji

#### Network 000

LDP M8151 //włączone, gdy PLC działa, musi być wyzwalone zboczem narastającym, aby uniknąć powtarzania zadania  
 MOVD K400 D0 //pierwotna częstotliwość

#### Network 001

169  
 LDP M1  
 MOVD K900 D0 //zmienna częstotliwość zapisana w D0D1



**Network 002**

LDP M2

MOVD K650 D0 //zmienna częstotliwość zapisana w D0D1

**Network 003**

LDP M3

MOVD K500 D0 //zmienna częstotliwość zapisana w D0D1

**Network 004**

LD M0

PLSY D0 D2 Y000 //gdy M0 jest włączone, Y0 wysła impuls z częstotliwością zapisaną w D0D1 i przestaje, gdy wartość w D2D3 osiągnie docelową liczbę impulsów, częstotliwość zmienia się.

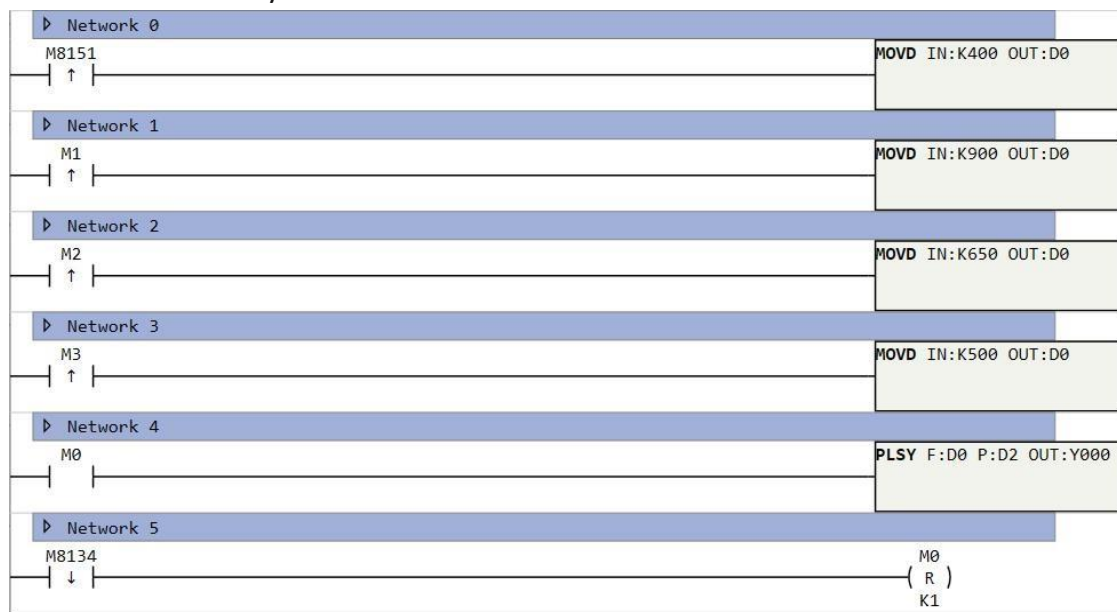
**Network 005**

LDF M8134

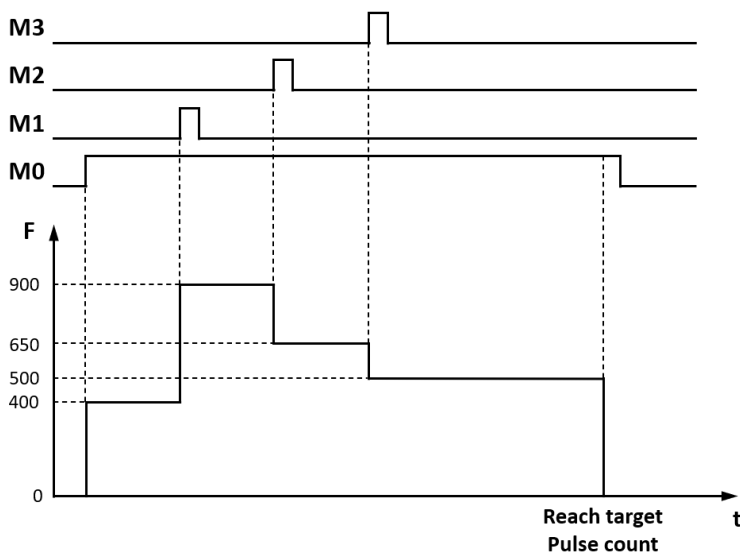
RST M0 K1 //po zatrzymaniu impulsu zresetuj M0, aby wyłączyć funkcję PLSF

POP

Schemat drabinkowy:



Wykres sekwencji:



## PLSR/PLSA/EPLSR

1. **PLSR**, **PLSA** i **EPLSR** są instrukcjami wielostopniowych wyjść impulsowych, wysyłają impuls z ustawioną częstotliwością i docelową liczbą impulsów, a parametry można regulować dynamicznie. Docelowa liczba impulsów **PLSR** jest ustawiona względnie, docelowy impuls **PLSA** jest ustawiony bezwzględnie, **EPLSR** może ustawić czas przyspieszania/zwalniania każdego segmentu osobno.
2. Kiedy **PLSR/PLSA/EPLSR** jest włączone, impuls rozpocznie się z ustawioną częstotliwością i przejdzie do następnego segmentu, gdy osiągnie docelową ilość impulsów, przejście zostanie wykonane w czasie przyspieszania/zwalniania. Wszystkie te parametry pracy są przechowywane w części obszaru rejestru D, zaczynającej się od pierwszego adresu Dn. Rysunek poniżej pokazuje przypadek, gdy pierwszy adres to D0.
3. Gdy docelowa liczba impulsów jednego segmentu jest ustawiona na <1, wyprowadzanie impulsów będzie stałe w segmencie, aż odpowiednia instrukcja **PLSNEXT** zostanie włączona. Kiedy częstotliwość i docelowa ilość impulsów jednego segmentu jest ustawiona na 0, segment będzie segmentem końcowym, wykonywanie impulsów zostanie zatrzymane w tym segmencie niezależnie od kolejnych D-rejestrów.
4. Dla pojedynczego segmentu całości, ustawia się częstotliwość początkową segmentu jako  $F_s$  (Hz), następnie częstotliwość końcową segmentu jako  $F_e$  (Hz), sumę czasów przyspieszania i zwalniania jako  $T$  (s), ilość impulsów tego segmentu jako  $N$ . Parametry te powinny spełniać następujący warunek:

$$(F_s + F_e)T \leq 2N$$

5. Dla **PLSR** o kierunku wyjścia decyduje znak częstotliwości lub docelowa ilość impulsów w pierwszym segmencie (jeżeli jeden z nich jest ujemny, kierunek wyjścia jest ujemny) i nie zmienia się on w kolejnych segmentach. W przypadku **PLSA** wyjście kierunku jest ustalane na podstawie docelowej liczby impulsów w każdym segmencie i będzie się zmieniać automatycznie. W przypadku **EPLSR** wyjście kierunku również zmienia się automatycznie zgodnie z rzeczywistą liczbą impulsów.
6. **M8068** to rejestr polaryzacji wyjścia impulsowego, gdy jest włączony, wszystkie wyjścia kierunku impulsowego zostaną odwrócone. **M8069** jest rejestrem typu krzywej, gdy jest włączony, wysyłanie impulsów przyspiesza i zwalnia jako typ krzywej S, w innym przypadku jako typ krzywej T.

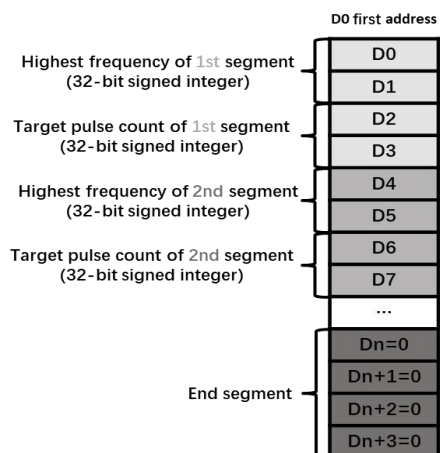


7. Parametry częstotliwości i docelowej liczby impulsów zapisane w rejestrze D mogą być modyfikowane podczas pracy, ale modyfikacja jest dozwolona tylko w segmentach następujących po bieżącym.

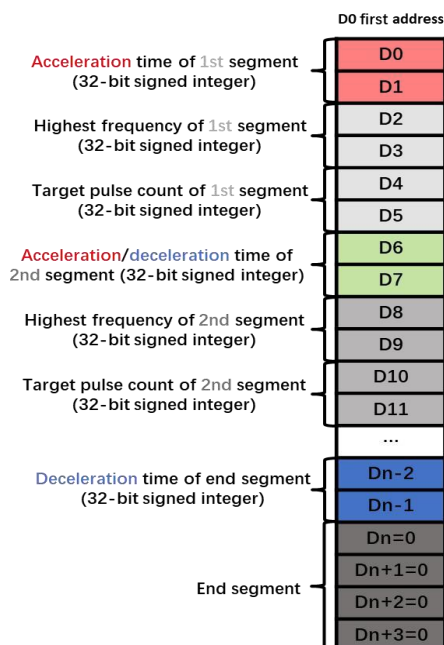
### Ustawienia danych:

**PLSR/PLSA** i **EPLSR** tworzą fragment obszaru rejestrów D do przechowywania parametrów segmentów. Jak pokazano na rysunku poniżej, dla **PLSR/PLSA** każdy z czterech rejestrów D przechowuje parametry jednego segmentu; W przypadku **EPLSE** każde sześć rejestrów D przechowuje parametry jednego segmentu.

#### PLSR/PLSA



#### EPLSR



#### PLSR/PLSA

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające instrukcję		0/1	Wartość logiczna
(D)	Pierwszy adres parametrów segmentów.	D	-	32-bitowy wskaźnik
(T)	Czas przyspieszania/zwalniania podczas tranzytu częstotliwości.	K/D	0~65535	16-bitowa liczba całkowita bez znaku
(DIR)	Port, który wysyła kierunek impulsu.	Y	-	Wartość logiczna
(OUT)	Port wysyłający impuls.	Y	-	Wartość logiczna



## EPLAR

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające instrukcję		0/1	Wartość logiczna
(D)	Pierwszy adres parametrów segmentów.	D	-	32-bitowy wskaźnik
(DIR)	Port, który wysyła kierunek impulsu.	Y	-	Wartość logiczna
(OUT)	Port wysyłający impuls.	Y	-	Wartość logiczna

### Uwagi

1. Parametr częstotliwości impulsów PLSR/PLSA/EPLSR to 32-bitowa liczba całkowita, która zajmuje dwa rejestry D.
2. Ta instrukcja wyprowadza impuls z kierunkiem, dzięki czemu licznik impulsów (D8140-D8158) może liczyć w obu kierunkach.
3. Wysyłanie impulsów zatrzymuje się w segmencie z częstotliwością 0 i liczbą impulsów docelowych 0, zwróć na to uwagę podczas modyfikowania parametrów podczas pracy.
4. Liczniki akumulacji impulsów liczą się tylko wtedy, gdy instrukcja impulsowa jest włączona, zliczanie zostanie wyczyszczone po wyłączeniu instrukcji.

### Przykłady:

1. Za pomocą instrukcji MOVD ustaw parametry PLSR:

### Lista instrukcji

#### Network 000

LDP M8151

```

MOVD    K100000    D0 //ustawić częstotliwość impulsu 1-go segmentu na D0D1
MOVD    K200000    D2 //ustaw docelową ilość impulsów 1-go segmentu na D2D3
MOVD    K200000    D4 //ustawić częstotliwość impulsu 2-go segmentu na D4D5
MOVD    K300000    D6 //ustawić częstotliwość impulsu 1-go segmentu na D6D7
FMOVD   K0         D8 K2 //segment końcowy, ustaw 0 na D8-D10

```

#### Network 001

LD M0

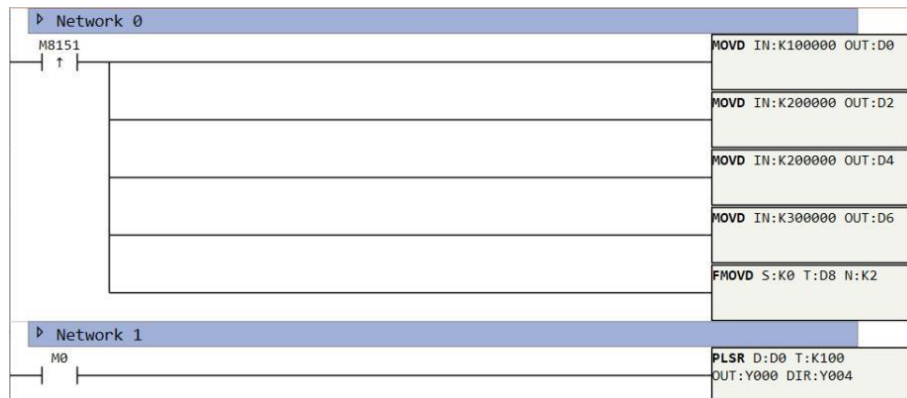
```

PLSR    D0    K100    Y000    Y004 //PLSR zaczyna działać, gdy M0 jest włączone

```



### Schemat drabinkowy:



### 2. Użyj bloku funkcyjnego do ustawienia parametrów PLSR:

Znajdź instrukcję **PLSR** w eksploratorze projektu, klikając ścieżkę: Project -> instruction -> Pulse -> PLSR, kliknij prawym przyciskiem myszy PLSR, będzie możliwe ustawienie nazwy funkcji i blok funkcji. Utwórz tę funkcję, a na schemacie drabinkowym pojawi się blok instrukcji **CALLM** (przykład na rysunkach poniżej).

Params Function

New function name: SetPLSRParams\_2

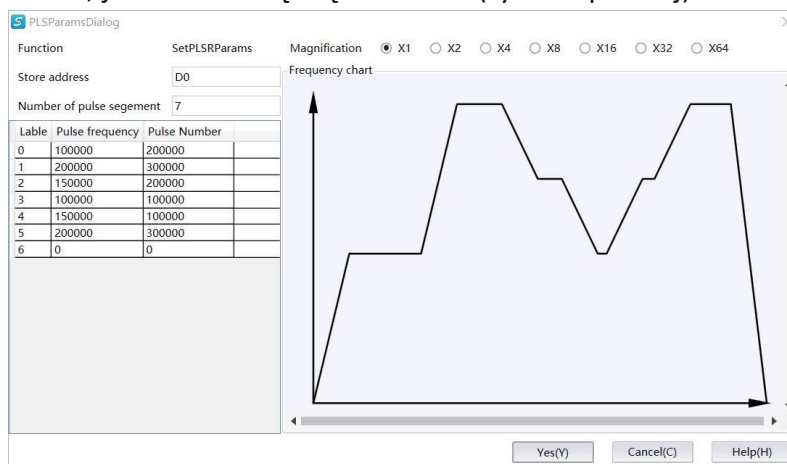
Funcblock of function: 脉冲参数设置

Yes(Y) Cancel(C)

```

CALLM F:SetPLSRParams
in:D0
  
```

Kliknij dwukrotnie blok instrukcji **CALLM**, możliwe jest wpisanie wartości w tabeli, aby ustawić parametry. Wykres częstotliwości pokazuje pokrótce, jak zmienia się częstotliwość (rysunki poniżej).





```

1  [PLSRParams]
2  void SetPLSRParams (DWORD in)
3  {
4      in[0] = 100000;The highest frequency of pulse segment 1
5      in[1] = 200000;The pulse number of pulse segment 1
6      in[2] = 200000;The highest frequency of pulse segment 2
7      in[3] = 300000;The pulse number of pulse segment 2
8      in[4] = 150000;The highest frequency of pulse segment 3
9      in[5] = 200000;The pulse number of pulse segment 3
10     in[6] = 100000;The highest frequency of pulse segment 4
11     in[7] = 100000;The pulse number of pulse segment 4
12     in[8] = 150000;The highest frequency of pulse segment 5
13     in[9] = 100000;The pulse number of pulse segment 5
14     in[10] = 200000;The highest frequency of pulse segment 6
15     in[11] = 300000;The pulse number of pulse segment 6
16     in[12] = 0;The highest frequency of pulse segment 7
17     in[13] = 0;The pulse number of pulse segment 7
18 }

```

### Network 000

```

LD      M8151
CALLM   SetDPLSRParams    D0 //ustawić parametry przez funkcje
MOVD    K100    D100 //ustaw czas przyspieszania/zwalniania

```

### Network 001

```

LD      M0
PLSR     D0      D100 Y000 Y004 //PLSR zaczyna działać, gdy M0 jest włączone

```

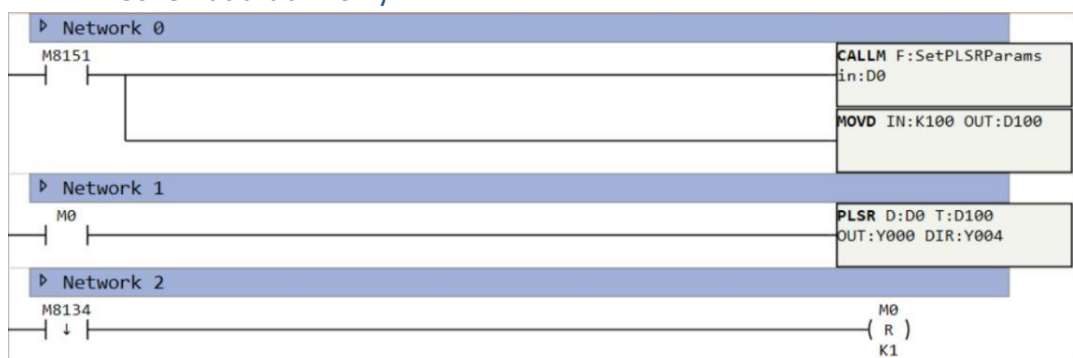
### Network 002

```

LDF      M8134 //koniec wysyłania impulsu
RST      M0     K1 //zresetuj M0
POP

```

Schemat drabinkowy:



## PLSNEXT

Ta instrukcja przełącza bieżący segment wielostopniowego wyjścia impulsowego na następny segment. Gdy instrukcja jest włączona, a wysyłanie impulsów ma stabilną częstotliwość, wysyłanie impulsów natychmiast przyspiesza/zwalnia do częstotliwości następnego segmentu.



Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(OUT)	Port wysyłający impuls.	Y	-	Wartość logiczna

1. Ta instrukcja nie działa, gdy pulsowanie przyspiesza lub zwalnia. Gdy następny segment jest segmentem końcowym, a częstotliwość impulsu jest stabilna, włączenie tej instrukcji spowoduje natychmiastowe zatrzymanie impulsu.
2. Ta instrukcja może być używana tylko dla **PLSR/EPLSR**.
3. Ta instrukcja musi być włączona z wyzwalaniem zboczem, w przeciwnym razie impuls będzie w nieładzie.

#### Przykład

#### Lista instrukcji

##### Network 000

```
LD M8151
MOVD K100000 D0 // ustaw częstotliwość 1-go segmentu
MOVD HFFFFFFF D2 // ustaw liczbę impulsów docelowych pierwszego segmentu na maksimum, poczekaj na PLSNEXT
MOVD K200000 D4 // ustaw częstotliwość 2-go segmentu
MOVD K9999999 D6 // ustawić docelową ilość impulsów 2-go segmentu
FMOVD K0 D8 K2 //koniec segmentu
MOVD K100 D100 //ustaw czas przyspieszania/zwalniania
```

##### Network 001

```
LD M0
PLSR D0 D100 Y000 Y010 // PLSR zaczyna działać, gdy M0 jest włączone
```

##### Network 002

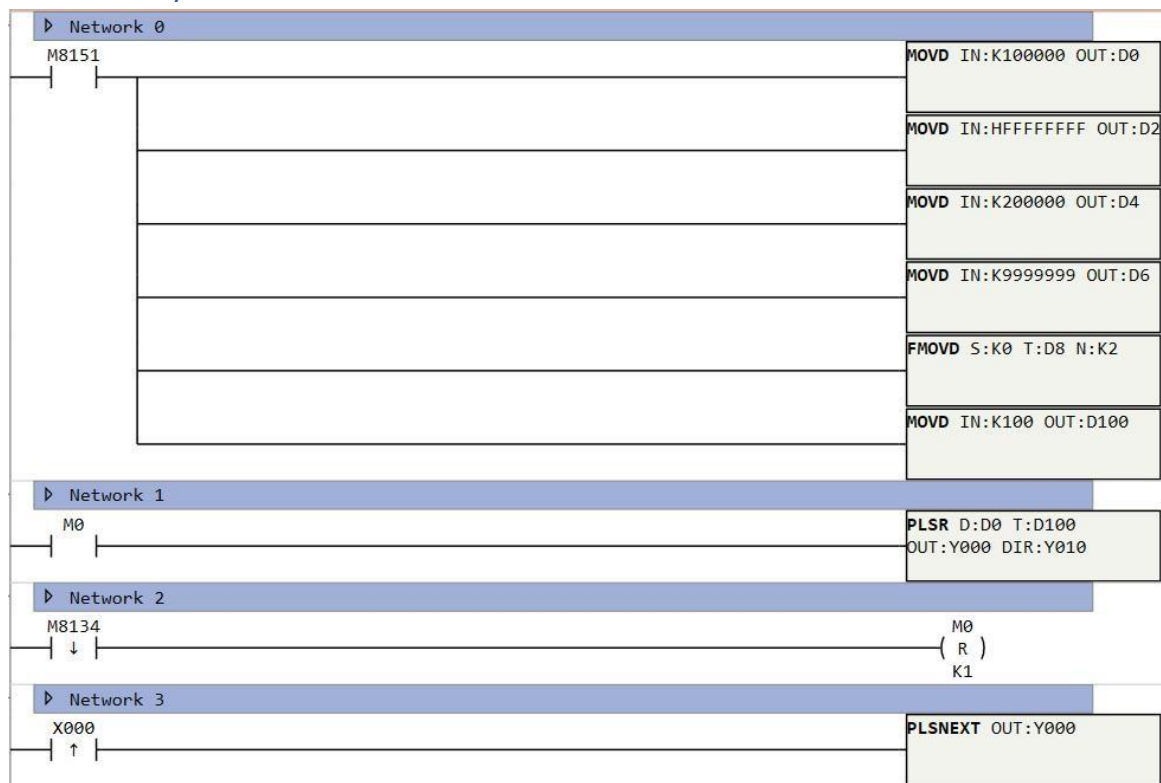
```
LDF M8134 // wysyłanie impulsów zakończone
RST M0 K1 //zresetuj M0
```

##### Network 003

```
LDP X000 //użyj narastającego zbocza X000 do wyzwalania
PLSNEXT Y000 //impuls przełączania Y000
```



## Schemat drabinkowy:



## PLSSTOP

Gdy ta instrukcja jest włączona, odpowiedni port wyjścia impulsowego przestanie wysyłać impulsy zgodnie z wybranym trybem zatrzymania. Jeśli tryb zatrzymania to zatrzymanie natychmiastowe, wysyłanie impulsów zostanie natychmiast zatrzymane bez czasu zwalniania; Jeśli tryb zatrzymania tryb zwalniania do zatrzymania, wysyłanie impulsów zostanie zatrzymane w ustawionym czasie zwalniania.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające instrukcję		0/1	Wartość logiczna
(MODE)	Tryb zatrzymania wyjścia impulsowego, 0 dla natychmiastowego zatrzymania, 1 dla zwolnienia do zatrzymania	K	0~1	8-bitowa liczba całkowita bez znaku
(TIME)	Czas zwalniania do zatrzymania.	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku
(OUT)	Port wysyłający impuls.	Y	-	Wartość logiczna

Ta instrukcja musi być załączana sygnałem zbocza.





Przykład

Lista instrukcji

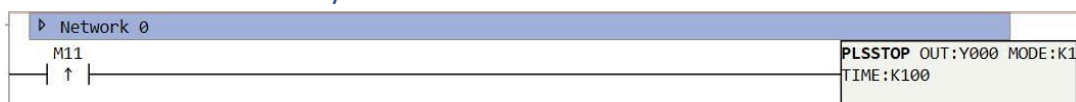
### Network 000

LDP M11 //załącz narastającym zboczem M11

PLSSTOP Y000 K1 K100 //wyjście impulsowe Y0 zatrzymuje się po 100ms

POP

Schemat drabinkowy:



## DRVI/DRVA

- DRVI/DRVA** wyprowadza impuls z ustawioną częstotliwością, ustawioną docelową liczbą impulsów, ustawionym czasem przyspieszania i zwalniania, **DRVI** jest ustawiony z względną docelową liczbą impulsów, a **DRVA** jest ustawiony z bezwzględną docelową liczbą impulsów. Gdy instrukcja jest włączona, impuls zaczyna przyspieszać do ustawionej częstotliwości w ustawionym czasie przyspieszania i zwalnia do 0 w ustawionym czasie zwalniania i osiąga zadaną liczbę impulsów docelowych. Gdy instrukcja jest wyłączona, puls natychmiast się zatrzyma.
- Te instrukcje mogą być kontrolowane przez specjalny rejestr funkcji **M8068**, gdy **M8068** jest włączony, kierunek wyjścia portu zostanie odwrócony.
- Ta instrukcja może być ustawiona w trybie krzywej za pomocą specjalnego rejestru funkcji **M8069**, gdy **M8069** jest włączony, krzywa częstotliwości ma kształt litery S, gdy jest wyłączona, krzywa częstotliwości ma kształt litery T.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(F)	Częstotliwość impulsu, jednostką jest [Hz].	K/H/D	■ <b>FGs/FGRB/FGRE/FGRS</b> 0~200000 ■ <b>FGm</b> 0~500000	32-bitowa liczba całkowita bez znaku
(P)	Docelowa liczba impulsów.	K/H/D	0~4294967296	32-bitowa liczba całkowita bez znaku
(A)	Czas przyspieszania.	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku
(T)	Czas zwalniania.	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku
(OUT)	Port wysyłający impuls.	Y	-	Wartość logiczna



(DIR)	Port, który wysyła kierunek impulsu.	Y	-	Wartość logiczna
-------	--------------------------------------	---	---	------------------

Ta instrukcja wyprowadza impuls z kierunkiem, dzięki czemu licznik impulsów (**D8140-D8158**) może zliczać w obu kierunkach.

[Przykład](#)

[Lista instrukcji](#)

### Network 000

LDP M8151

MOVD K100000 D0 //docelowa ilość impulsów w D0D1

MOVD K50000 D2 //częstotliwość impulsów D2D3

MOV K100 D4 //czas przyspieszania w D4

MOV K200 D5 //czas zwalniania D5

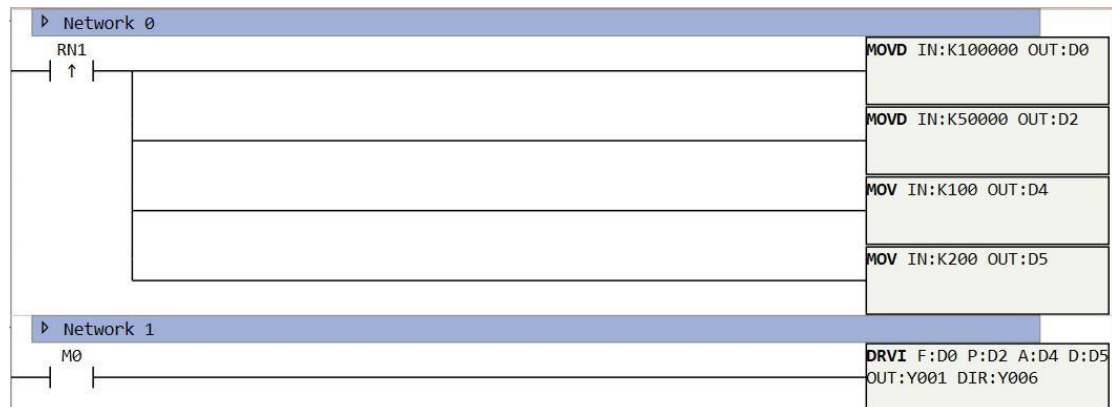
### Network 001

LD M0

DRVI D0 D2 D4 D5 Y1 Y6 //impuls zaczyna być wysyłany, gdy M0 jest włączone

POP

[Schemat drabinkowy:](#)





## POLYLINEF/POLYLINEI

1. Instrukcje **POLYLINEI/POLYLINEF** różnią się typami parametrów (integer i float), odpowiednio używają pls (liczba impulsów) i mm (milimetry) jako jednostek.
2. Te instrukcje wysyłają impulsy do sterowania ruchem, którego ścieżka jest połączona liniami i łukami. Częstotliwość impulsu płynnie przechodzi przez ścieżki, gdy dwa segmenty ścieżki są połączone współliniowo lub stycznie, w przeciwnym razie impuls zwalnia lub przyspiesza w pobliżu połączeń. Konfiguracja linii i łuków dotyczy wprowadzenia instrukcji **LINEI/LINEF** oraz **ARCI/ARCF**.
3. Parametry **POLYLINE** mogą być mapowane do rejestrów D w celu kontroli dynamicznej, użytkownik może modyfikować wartości w rejestrach D w celu modyfikacji parametrów ścieżki, dołączając wartość do rejestrów D w celu dodania nowych segmentów. Dodatkowo dopuszczalna jest modyfikacja mapowanych wartości prędkości w rejestrach D podczas ruchu, dzięki czemu prędkość może być dostosowywana dynamicznie. Jeśli użytkownik przełączy tryb mapowania na „Tylko mapowanie do rejestrów D”, nie ma potrzeby wprowadzania żadnych wartości parametrów w instrukcji, tor ruchu zostanie całkowicie określony przez przypisaną sekwencję danych w D-rejestrach.

### Ustawienie systemowe polilinii:

W przypadku korzystania z którejkolwiek instrukcji interpolacji (POLYLINEI/POLYLINEF, LINEI/LINEF, ARCF i BLOCK itp.) w pierwszej kolejności ustawiona być musi platforma interpolacji („platform system”). Tworzy się układ ruchów z 2 osiami, na których opiera się cała interpolacja.

### Krótkie wprowadzenie

1. Kolejność kliknięć ustawień systemu to: Main menu -> Pulse -> Pulse platform.
2. Do wyboru jest **5 platform ruchów interpolowanych**, które zaczynają obowiązywać po włączeniu polecenia.
3. Istnieją dwa rodzaje jednostek do wyboru: mm (milimetry) i pls (liczba pls). Milimetr jest bardziej zalecany, ponieważ jednostka milimetra jest bardziej realistyczna, mniej podatna na błędy i łatwa do przeliczenia. Wybór



jednostek wpływa tylko na ustawienia platformy interpolacji, rzeczywista jednostka ruchu zależy od rodzaju instrukcji.

4. Przetwarzanie przekroczenia jest stosowane razem z ograniczeniem osi X/Y. System zaalarmuje, gdy ruch następnego segmentu przekroczy limit, jeśli wybrano „Ignorowane”, przekroczenie zostanie zignorowane; Jeśli wybrano „Zwalnianie i zatrzymanie”, ruch zostanie zatrzymany w punkcie końcowym ostatniego segmentu w ramach limitu.

5. Port wyjścia impulsowego musi być ustawiony z szybkim portem wyjściowym impulsowym ([patrz pierwsza uwaga dotycząca instrukcji impulsowych](#)), port wyjściowy kierunku można ustawić z dowolnym portem Y.

Wszystkie porty wyjścia impulsowego i porty wyjścia kierunkowego włączonej tabeli ruchów interpolowanych nie mogą powtarzać się.

6. Ekwiwalent impulsu (czasami nazywany również długością skoku) odnosi się do odległości, jaką siłownik porusza się w odpowiedzi na pojedynczy impuls lub sygnał sterujący. Aby uzyskać dokładny ekwiwalent impulsu, należy wziąć pod uwagę różne czynniki, takie jak podział silnika, elektroniczne przełożenie skrzyni biegów, mechaniczne przełożenie skrzyni biegów i tak dalej. Ekwiwalent impulsu to parametr podstawowy, dokładne jego określenie pozwala na wiarygodne powiązanie współrzędnych zestawu z rzeczywistą miarą przemieszczenia siłownika. Jest to szczególnie ważne w przypadku zastosowań, gdzie wymagana jest wysoka precyzja pozycjonowania, takich jak w automatyka przemysłowa, robotyka, maszyny CNC itp.

7. Pozycja graniczna jest teoretyczną pozycją graniczną platformy interpolacji, należy ją ustawić zgodnie z rzeczywistą sytuacją. Gdy ruch przekroczy granicę, system będzie przetwarzał zgodnie z wybranym trybem przetwarzania przekroczenia.

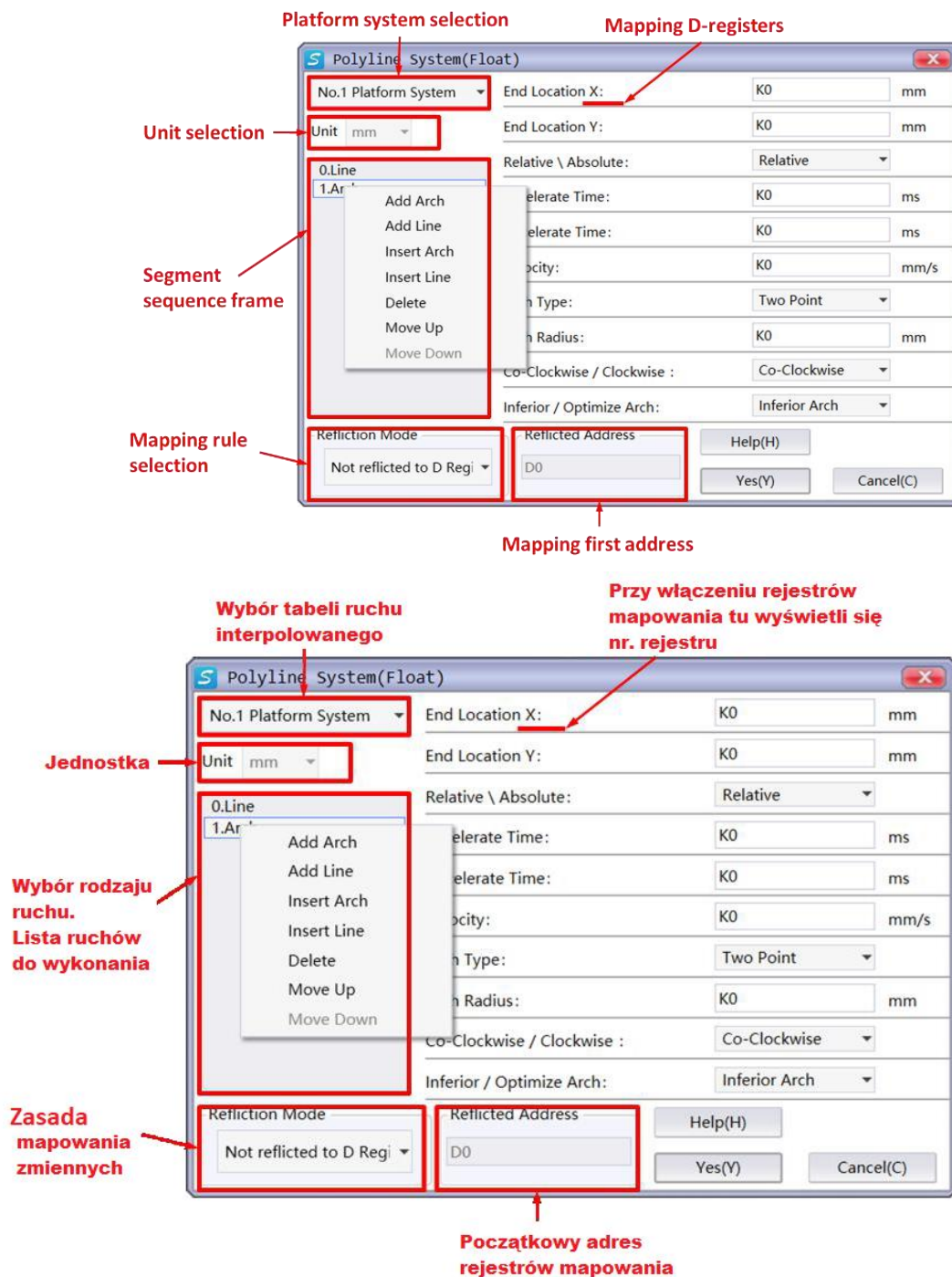
8. Luz przekładni oznacza luz, który powstaje, gdy wał napędowy cofa się i napotyka szczelinę (zawsze spowodowaną ścieraniem i montażem konstrukcji mechanicznej). System skompensuje luz w kilku cyklach interpolacji, kiedy odpowiednia oś jest odwrócona. Zwykle nie ma potrzeby ustawiania tego parametru w strojeniu zgrubnym, najbardziej przydaje się w strojeniu dokładnym. Tego parametru nie powinno się ustawiać zbyt wysoko, ponieważ może to prowadzić do utraty precyzji i dokładności ruchu, lepiej wyregulować konstrukcję mechaniczną, gdy luz jest duży. Dodatkowo parametr luzu nie ma większego znaczenia w przypadku sytuacji z dużą prędkością.

### Uwagi

1. Ekwiwalent impulsów osi X i osi Y powinien być ustawiony tak, aby był jak najbardziej zgodny z rzeczywistym ruchem; jeśli nie jest to możliwe, jako jednostkę systemową należy wybrać pls (liczba impulsów), aby uniknąć problemów z konwersją.
2. Gdy ekwiwalent impulsu osi X i osi Y jest ustawiony inaczej, wszystkie obliczenia jednostki liczby impulsów bez związku z określoną osią (takie jak obliczenie prędkości) będą oparte na ekwiwalencie impulsu osi X.
3. Wszystkie porty wyjścia impulsowego i porty wyjścia kierunkowego włączonej platformy nie mogą się powtarzać.



## Wprowadzenie do operacji



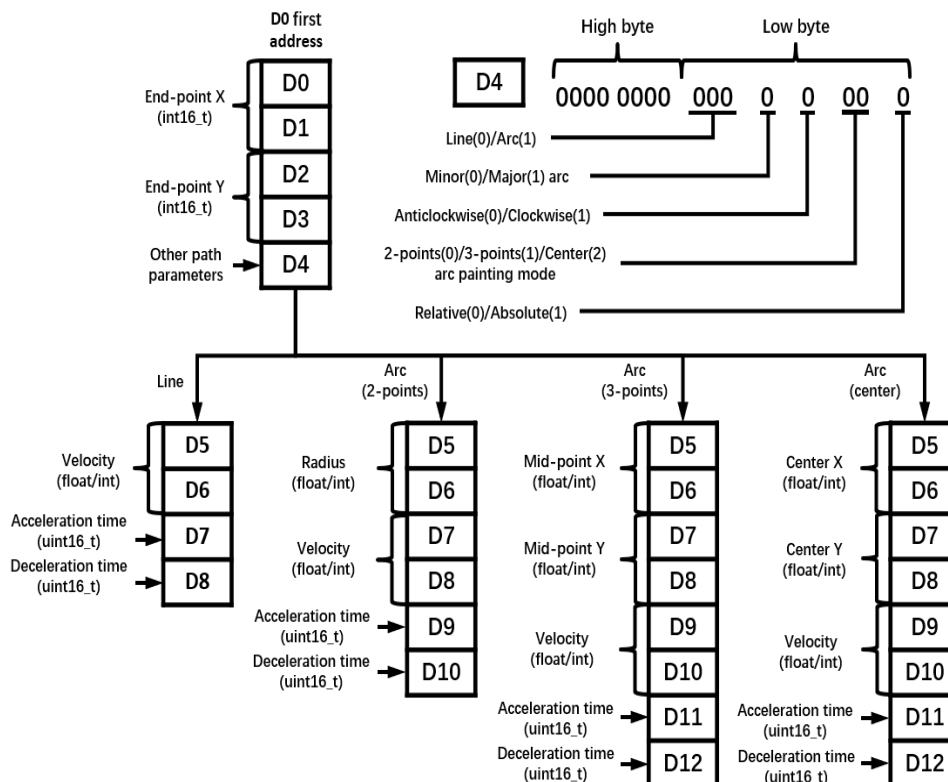
1. Po pierwsze, platforma ruchów interpolowanych „platform system” musi być wybrana i włączona, w przeciwnym razie instrukcja nie może działać normalnie.
2. Użytkownik może wybrać mm (milimetry) lub pls (liczbę impulsów) jako jednostkę parametru, odpowiednio zgodnie z instrukcją POLYLINEF i POLYLINEI. **POLYLINEF** jest bardziej zalecane, ponieważ jednostka milimetrowa jest bardziej realistyczna, mniej podatna na błędy i łatwa do konwertowania.



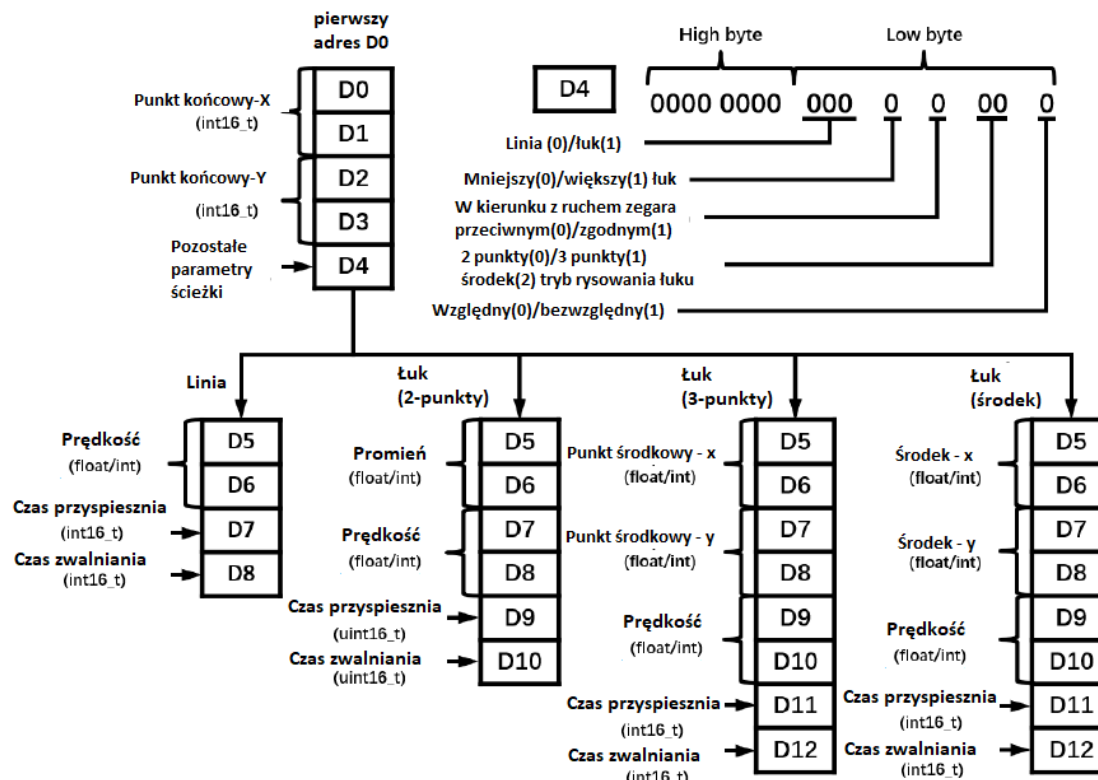
3. Następnie należy kliknąć prawym przyciskiem myszy listę ruchów do wykonania po lewej stronie, aby wybrać operację. Do wyboru są operacje dodawania, wstawiania, usuwania i przenoszenia. (Konfiguracja linii i łuków odnosi się do wprowadzenia **LINEI/LINEF** i **ARCI/ARCF**).

4. Parametry położenia i prędkości można wprowadzać za pomocą danych typu K lub D. W sytuacji, gdy parametr prędkości jest wprowadzany za pomocą danych typu D, zmiana wartości odpowiadającego rejestru D może również zmieniać prędkość w czasie rzeczywistym. Dodatkowo prędkość nie zmieniałaby się natychmiast podczas przyspieszania lub zwalniania, ale po przyspieszaniu lub zwalnianiu. Należy zauważyć, że przy stosowaniu instrukcji typu MOV do przypisania rejestrów D dla parametrów ścieżki, o wyborze **MOVD** i **MOVF** decyduje **POLYLINE** typ instrukcji (**POLYLINEI**/**POLYLINEF**). **POLYLINEI** jest instrukcją używającą parametrów integer, której jednostką jest liczba impulsów, więc wartość musi być przypisana przez **MOVD**. **POLYLINEF** to instrukcja wykorzystująca parametry float, których jednostką jest milimetr, więc wartość musi być przypisana przez **MOVF**. A jednostką parametru przyspieszania i zwalniania jest milisekunda, jest to 16-bitowa liczba całkowita niezależnie od typu instrukcji, powinna być przypisana przez **MOV**.

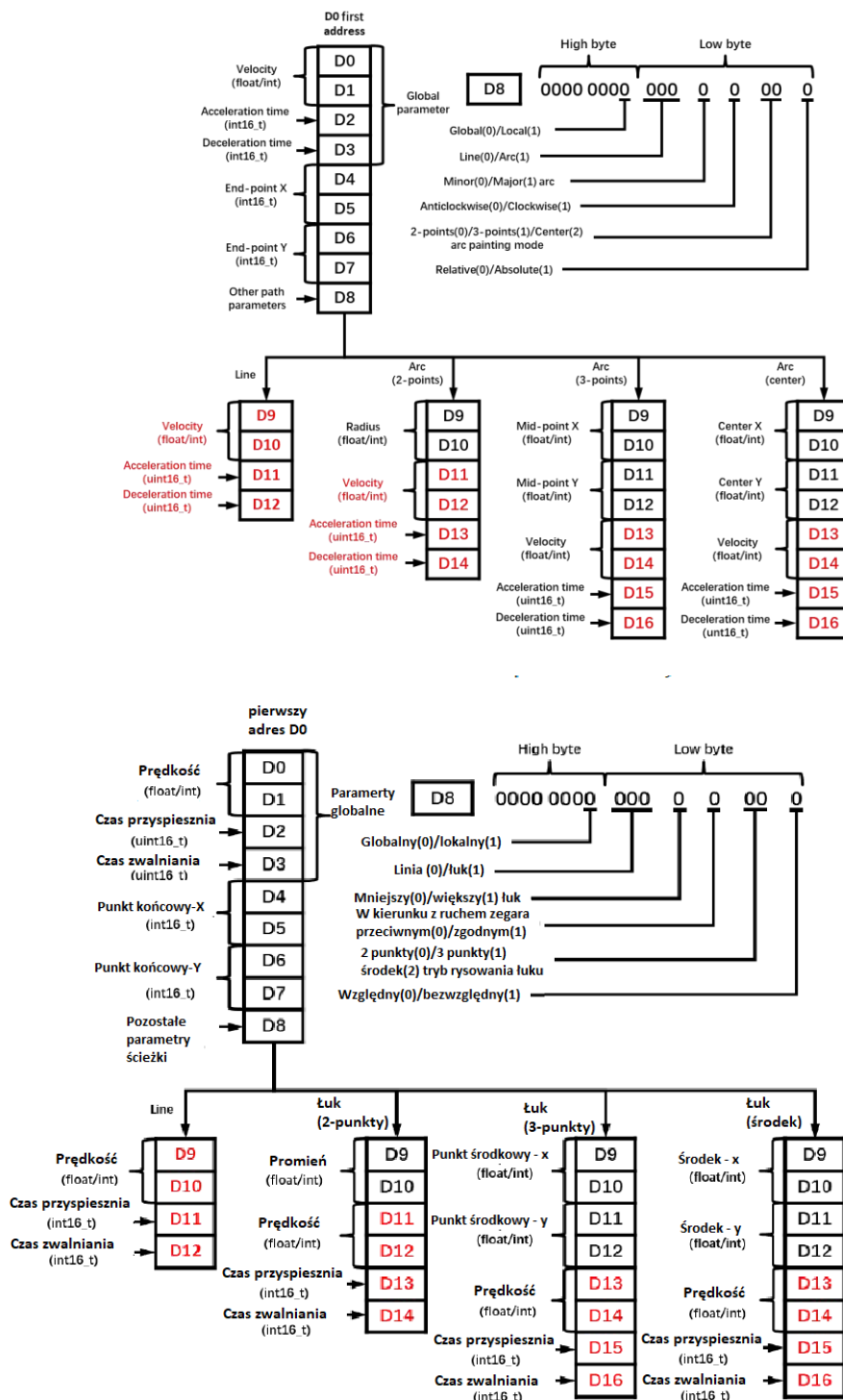
5. Tryb „Mapping to D-registers” oznacza, że w czasie, gdy instrukcja jest włączona, wszystkie parametry ustawień (w tym położenie, prędkość) zostaną zapisane w zmapowanych rejestrach D w określonej kolejności. Reguła mapowania odnosi się do rysunku poniżej, również zmapowany adres rejestrów D pojawi się przed polem wprowadzania, gdy użytkownik wybierze ten tryb. Należy zauważyć, że wykonywanie segmentów interpolacji jest procesem jednokierunkowym. Segment następujący po wykonanym będzie przetwarzany i nie ma możliwości jego modyfikowania. Modyfikacja jest dozwolona tylko 2 lub więcej segmentów po tym, który jest wykonywany, w przeciwnym razie pojawi się potencjalny błąd, który doprowadzi do zamieszania, a nawet błędu w działaniu. Ta funkcja ma zastosowanie tylko wtedy, gdy potrzebne są niewielkie modyfikacje ścieżki.





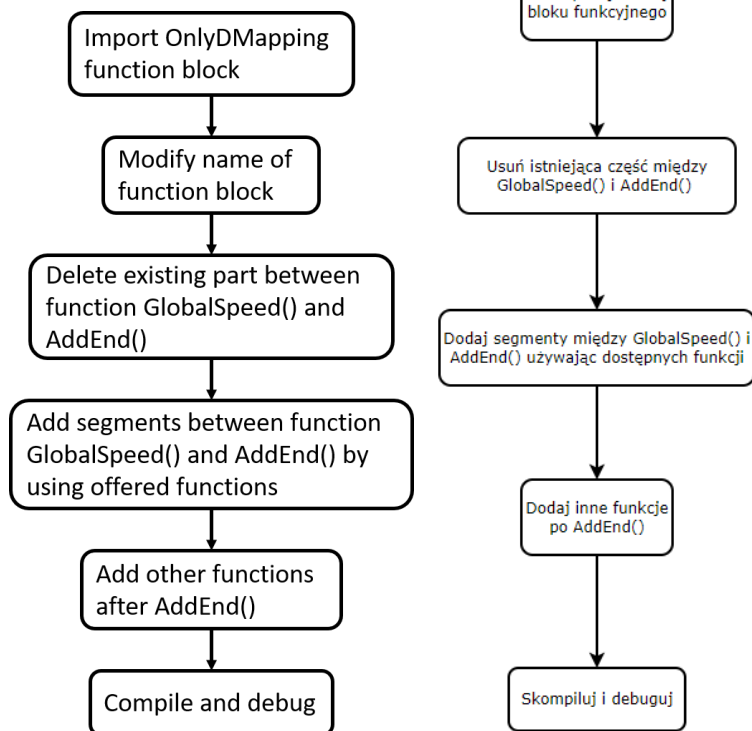
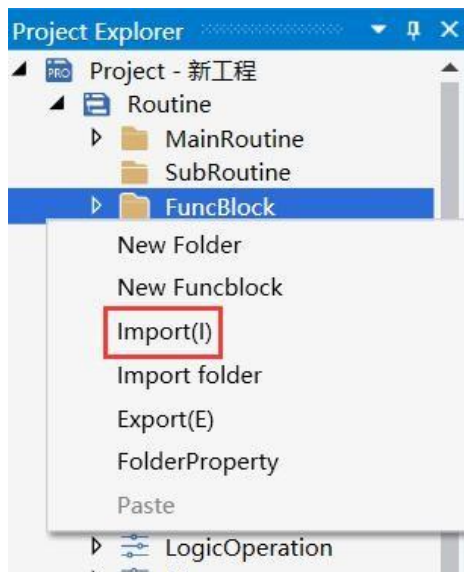


6. Gdy wybrany jest tryb „Tylko mapowanie do rejestrów D”, ścieżka będzie kontrolowana tylko przez wartości w zmapowanych rejestrach D, a wartości wprowadzone w instrukcji nie mają żadnego efektu w tym trybie. Reguła mapowania odnosi się do rysunku poniżej. W odróżnieniu od trybu „Mapping to D-registers” istnieje blok rejestrów D, który zapisuje globalne parametry prędkości. Jeśli bit kontroli trybu prędkości segmentów w innych parametrach jest ustawiony na 1, prędkość tego segmentu będzie kontrolowana przez lokalne parametry prędkości (zaznaczone na rysunku kolorem czerwonym), w przeciwnym razie będzie kontrolowana przez globalne parametry prędkości. Ta funkcja ma zastosowanie, gdy wymagana jest wsadowe modyfikowanie prędkości ścieżki.



7. W trybie „Only mapping to D registers” („Tylko mapowanie do rejestrów D”) używanie instrukcji typu MOV do przypisania wartości do zmapowanych rejestrów D będzie kłopotliwe i powodujące błędy. Do tej czynności przeznaczone są szablony funkcji „OnlyDMappingI” i „OnlyDMappingF”. Użytkownicy mogą importować i używać bloków funkcyjnych, patrz rysunek poniżej.





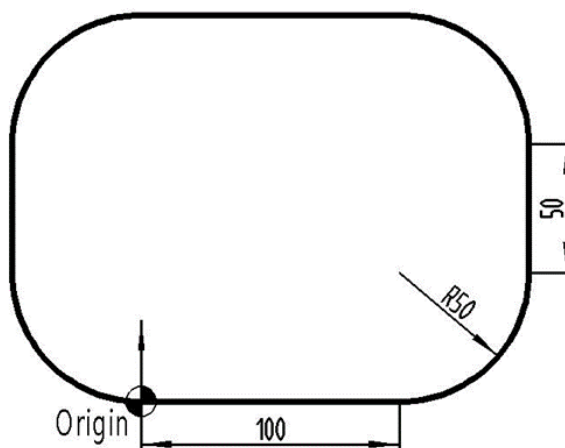
### Uwaga

1. To instrukcja wyprowadzania **impulsu z kierunkiem**, dzięki czemu licznik impulsów (D8140-D8158) może zliczać zarówno w kierunku dodatnim, jak i ujemnym.
2. Czasy przyspieszania i zwalniania nie będą egzekwowane, jeśli ustawiona wartość będzie zbyt mała.
3. Gdy ekwiwalenty impulsów osi X i Y są ustawione inaczej w platformie interpolacji, domyślnie prędkość rzeczywista będzie obliczana z ekwiwalentu impulsów osi X.
4. Rejestr **M8134** może być wyświetlany tylko wtedy, gdy wyprowadzany jest sygnał z Y0. Jeśli istnieje potrzeba uzyskania stanu pracy systemu interpolacji, użyj rejestru **M8144**, który pokazuje ON, gdy jakkolwiek oś tabeli ruchów interpolowanych wysła impuls.

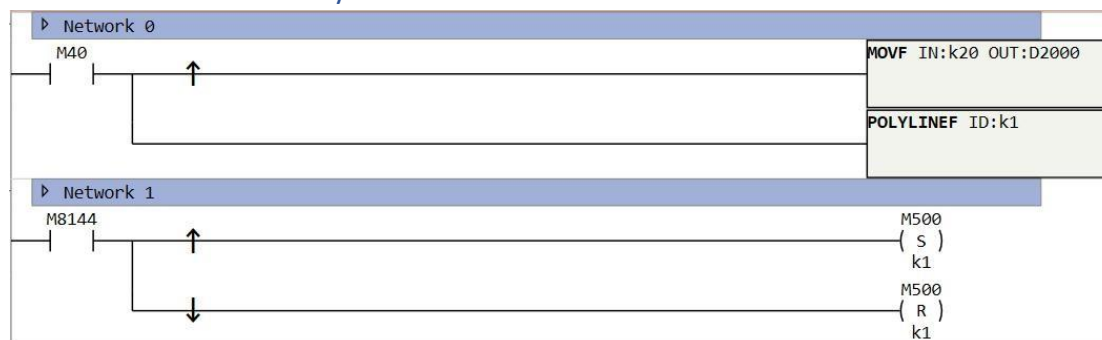


### Przykład

Można użyć tej instrukcji do narysowania wykresu jak na rysunku poniżej, prędkość interpolacji linii wynosi 100mm/s i jest regulowana, prędkość interpolacji łuku wynosi 50mm/s i jest stała.



### Schemat drabinkowy:



#### 1. Brak mapowania na D-rejestry

Dodaj kolejno linie i łuki oraz ustaw ich parametry pozycji. Dla prędkości, linia jest ustawiana za pomocą danych typu D i przypisana wartość za pomocą **MOV**/**MOV**F zgodnie z typem instrukcji, łuk jest ustawiany za pomocą danych typu K - K50.

#### 2. Mapowanie do D-rejestrów

Dodaj kolejno linie i łuki, ustaw pierwszy adres mapowania D-rejestrów i ustaw pozycję ich parametrów. Dostosuj wartość w odpowiednim rejestrze mapowania, aby dostosować prędkość podczas działania interpolacji.

#### 3. Mapowanie tylko do rejestrów D

Zaimportuj blok funkcyjny **OnlyDMappingF** i ustaw jako następujący kod. Dodatkowo wyzwoł blok funkcyjny raz, aby przypisać wartość przed włączeniem instrukcji interpolacji.

```
ParaAddr = GlobalSpeedF(ParaAddr, 100, 200, 200);
ParaAddr = AddLineF(ParaAddr, 100, 0, 0);
ParaAddr = Add2ArcF_S(ParaAddr, 50, 50, 50, 0, 0, 0, 50, 200, 200);
ParaAddr = AddLineF(ParaAddr, 0, 100, 0);
ParaAddr = Add2ArcF_S(ParaAddr, -50, 50, 50, 0, 0, 0, 50, 200, 200);
ParaAddr = AddLineF(ParaAddr, -100, 0, 0);
ParaAddr = Add2ArcF_S(ParaAddr, -50, -50, 50, 0, 0, 0, 50, 200, 200);
```



```
ParaAddr = AddLineF(ParaAddr, 0, -100, 0);  
ParaAddr = Add2ArcF_S(ParaAddr, 50, -50, 50, 0, 0, 0, 50, 200, 200);  
    ParaAddr = AddEndF(ParaAddr);
```



## LINEF/LINEI

### Wprowadzenie do instrukcji

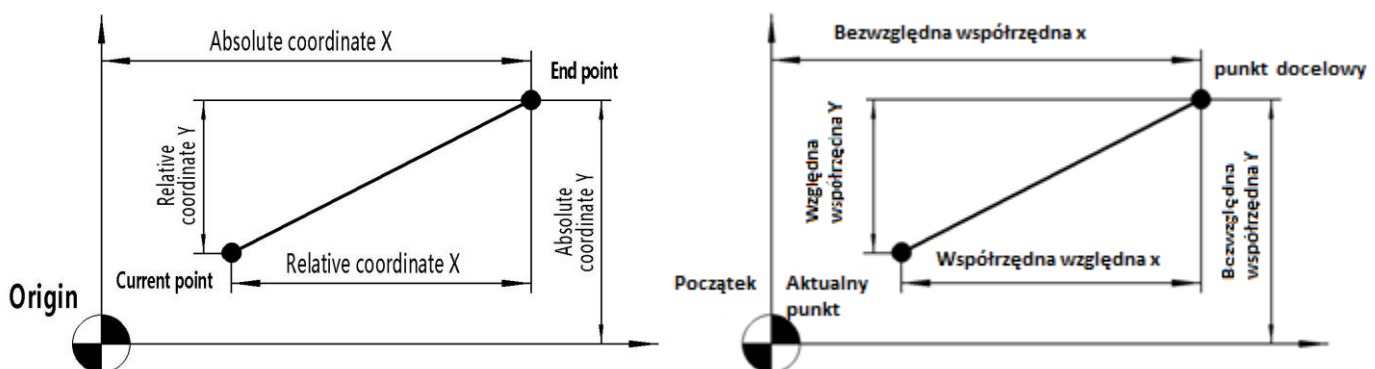
1. Instrukcje **LINEI/LINEF** różnią się typami parametrów (integer i float), odpowiednio używają pls (liczba impulsów) i mm (milimetry) jako jednostek.
2. Instrukcje te określają ścieżkę w linii prostej od bieżącego punktu do punktu końcowego w określonej platformie interpolacji.
3. Relacje długości toru linii (**L**), prędkości (**V**), czasu przyspieszania (**T<sub>ac</sub>**) i czasu hamowania (**T<sub>dc</sub>**) powinny spełniać poniższy warunek (wartość tych parametrów zostanie dostosowana automatycznie, jeśli ich relacje nie spełniają warunku):

$$0.5 \times (T_{ac} + T_{dc}) \times V \leq L$$

Jeśli parametry zostały ustawione zbyt nieracjonalnie, aby można je było regulować automatycznie, instrukcja może przestać wysyłać impulsy po sprawdzeniu błędu. Użytkownicy mogą modyfikować parametry odpowiadające kodowi błędu zapisanemu w rejestrze **D8176**.

### Wprowadzenie do operacji

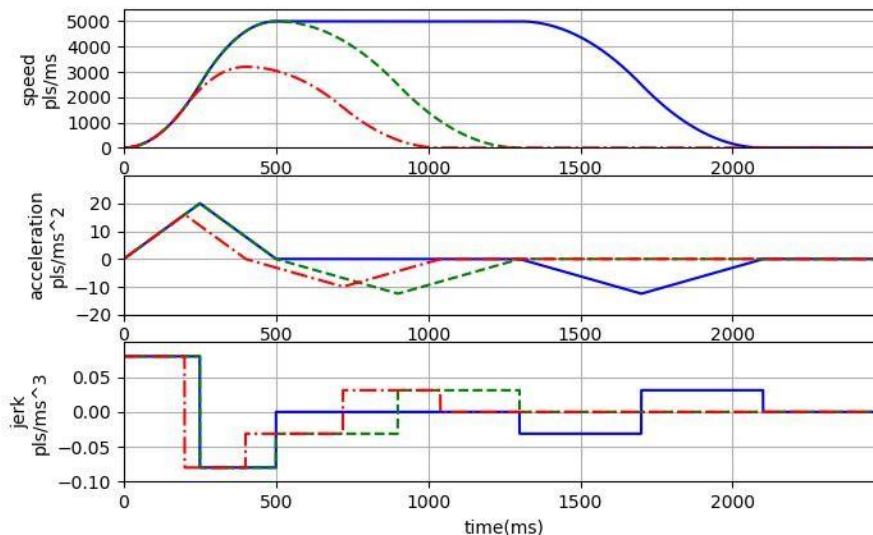
1. Najpierw należy wybrać i włączyć numer polilinii (patrz [ustawienia](#) tabeli ruchów interpolacji), w przeciwnym razie instrukcja nie może działać normalnie.
2. Użytkownik może wybrać mm (milimetry) lub pls (liczbę impulsów) jako jednostkę parametru, odpowiednio odpowiadającą instrukcji LINEF i LINEI. Bardziej zalecane są jednostki LINEF, ponieważ jednostka milimetrowa jest bardziej realistyczna, mniej podatna na błędy i łatwa do przeliczenia.
3. Punkt końcowy X i punkt końcowy Y to współrzędne położenia końcowego linii ścieżki, parametry te można wprowadzić za pomocą danych typu K lub D. O wyborze MOVD i MOVF do przypisania wartości parametrów decyduje typ instrukcji LINE (LINEI/LINEF).
4. Gdy wybrany jest tryb współrzędnych względnych, współrzędna punktu końcowego jest oparta na przesunięciu od bieżącego punktu. Gdy wybrany jest tryb współrzędnych bezwzględnych, współrzędna punktu końcowego jest oparta na przesunięciu od punktu początkowego. Te 2 tryby odnoszą się do rysunku poniżej.



5. Parametry czasu przyspieszania i zwalniania można ustawić za pomocą danych typu K lub D. W przypadku ustawienia z danymi typu D, podczas inicjalizacji instrukcja odczyta wartość z rejestrów D jako czas przyspieszania i



zwalniania. Parametry czasu przyspieszania i zwalniania są 16-bitowymi liczbami całkowitymi, muszą być przypisane instrukcją **MOV**. Ograniczony okresem interpolacji (4 ms) algorytmu, przeskakiwanie częstotliwości będzie zbyt nagłe, aby przyspieszyć i zwalniać skutecznie, jeśli czas został ustawiony za mały. Przyspieszenie i zwalnianie w interpolacji to asymetryczne przyspieszenie i opóźnienie w kształcie litery S, szczegóły dotyczące kształtu przedstawiono na rysunku.



6. Parametr prędkości oznacza prędkość wypadkową osi X i osi Y, parametr ten można ustawić za pomocą danych typu K lub D. W przypadku ustawień danych typu K prędkość ścieżki nie będzie ustawialna; Jeśli zostanie ustawione dane typu D, prędkość może być regulowana przez wartość w odpowiednim rejestrze D. Również wybór MOVD i MOVF do przypisania wartości parametrów prędkości jest ustawiana z instrukcją typu LINE (LINEI/LINEF). Jeśli ekwiwalenty impulsów osi X i Y są ustawione inaczej, system ustawi domyślnie prędkość na podstawie impulsu osi X. Gdy czas przyspieszania, czas zwalniania i parametry prędkości są ustawione nieprawidłowo, system dostosuje te parametry, aby były prawidłowe. Korekta odnosi się do rysunku powyżej (niebieska linia oznacza przypadek normalny, zielona linia oznacza przypadek graniczny, czerwona linia oznaczająca przypadek dostosowany).

#### Uwaga

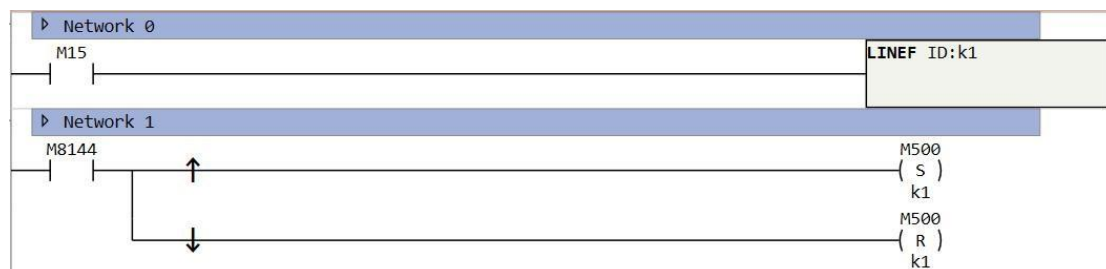
1. Ta instrukcja wyprowadza impuls kierunku, dzięki czemu licznik impulsów (D8140-D8158) może zliczać w obu kierunkach.
2. Przyspieszanie/zwalnianie nie działa, gdy jest ustawiona zbyt mała wartość.
3. Gdy jednostka jest ustawiona to pls, jednostką prędkości jest pls/s. Jeśli ekwiwalent impulsu osi X i osi Y są ustawione inaczej, prędkość wypadkowa jest obliczana domyślnie na podstawie ekwiwalentu impulsu osi X.
4. Rejestry specjalne **M8134** do **M8153** pokazują tylko, czy dany port wyjścia impulsowego wysyła impulsy, jeśli istnieje potrzeba wykrycia, czy interpolacja jest uruchomiona, użytkownik powinien użyć marker **M8144** do **M8148**, który pokazuje, czy odpowiednia platforma interpolacji „platform system” jest uruchomiona.



### Przykład

Narysuj linię od bieżącego punktu do punktu końcowego, gdy rozpoczyna się interpolacja, ustawiany jest **M500**, po zakończeniu interpolacji **M500** jest resetowana. Platforma ruchów interpolowanych „platform system” ustawia port Y0 jako port wyjściowy impulsów osi X, ustawia Y1 jako port wyjściowy impulsów osi Y.

### Schemat drabinkowy:



## ARCF/ARCI

### Wprowadzenie do instrukcji

1. Instrukcje **ARCI/ARCF** różnią się typami parametrów (integer i float), odpowiednio używają pls (liczba impulsów) i mm (milimetry) jako jednostek.
2. Te instrukcje uzupełniają ścieżkę łuku od bieżącego punktu do punktu końcowego w określonej platformie interpolacji.
3. Relacje długości toru linii (**L**), prędkości (**V**), czasu przyspieszania (**T<sub>ac</sub>**) i czasu hamowania (**T<sub>dc</sub>**) powinny spełniać poniższy warunek (wartość tych parametrów zostanie dostosowana automatycznie, jeśli ich relacje nie spełniają warunku):

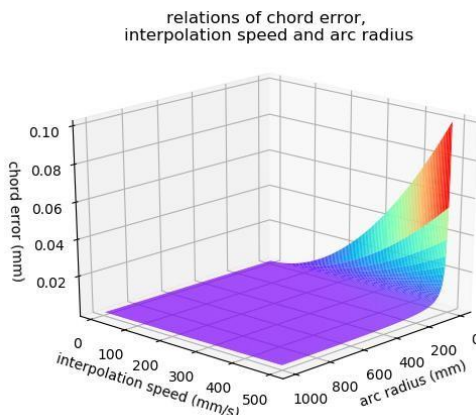
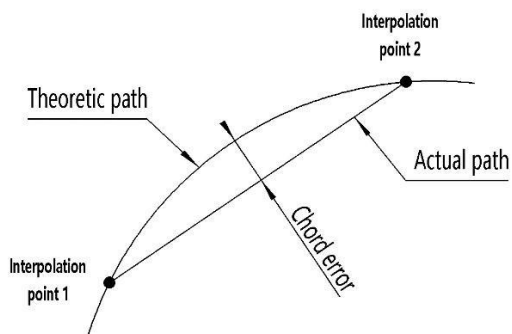
$$0.5 \times (T_{ac} + T_{de}) \times V \leq L$$

Jeśli parametry zostały ustawione zbyt nieracjonalnie, aby można je było regulować automatycznie, instrukcja może przestać wysyłać impulsy po sprawdzeniu błędu. Użytkownicy mogą modyfikować parametry odpowiadające kodowi błędu zapisanemu w rejestrze **D8176**.

4. W trybie malowania łukiem 2-punktowym należy zwrócić uwagę podczas ustawiania parametrów, że długość od punktu końcowego do punktu bieżącego musi być mniejsza niż promień łuku, w przeciwnym razie system uzna długość za średnicę i przeprofiluje ścieżkę łuku.
5. W trybie malowania łukiem 3-punktowym należy pamiętać, że podczas ustawiania parametrów punkt końcowy, punkt środkowy i punkt bieżący nie mogą być współliniowe.
6. W trybie malowania łukiem ze współrzędnymi środka należy zwrócić uwagę podczas ustawiania parametrów, że punkt środkowy musi znajdować się na środku prostopadłej linii łączącej punkt końcowy z punktem bieżącym.
7. Ścieżka interpolacji łuku składa się ze ścieżki mikrolinii, wystąpi błąd cięciwy (*chord error*) między ścieżką rzeczywistą a ścieżką teoretyczną (rysunek poniżej). Kiedy parametr promienia łuku jest mały, a parametr prędkości interpolacji duży, nawet jeśli parametry są prawidłowe, błąd cięciwy nadal nie może zostać zignorowany. Relacje błędu cięciwy, prędkości interpolacji i promienia łuku odnoszą się do rysunku po prawej, błąd cięciwy



będzie widoczny przy dużej prędkości i małym promieniu. System obniży prędkość, aby uniknąć dużego błędu cięwiwy, a także nie powinno być łuku o zbyt małym promieniu w rzeczywistym użyciu.



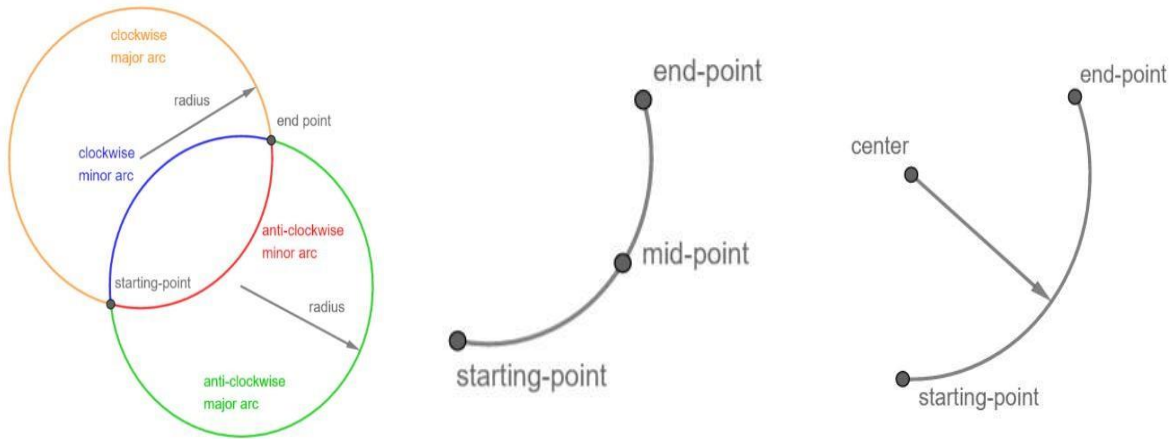
### Wprowadzenie do operacji

1. Najpierw należy wybrać i włączyć numer polilinii (patrz ustawienia tabeli ruchów interpolowanych „platform system”), w przeciwnym razie instrukcja nie może działać normalnie.
2. Użytkownik może wybrać mm (milimetry) lub pls (liczbę impulsów) jako jednostkę parametru, odpowiadającą odpowiednio instrukcji LINEF i LINEI. Bardziej zaleca się użycie LINEF, ponieważ jednostka milimetrowa jest bardziej realistyczna, mniej podatna na błędy i łatwa do przeliczenia.
3. Punkt końcowy X i punkt końcowy Y to współrzędne położenia końca łuku, parametry te można wprowadzić za pomocą danych typu K lub D. O wyborze MOVD i MOVF do przypisania wartości parametrów decyduje typ instrukcji LINE (LINEI/LINEF).
4. Gdy wybrany jest tryb współrzędnych względnych, współrzędna punktu końcowego jest oparta na przesunięciu od bieżącego punktu. Gdy wybrany jest tryb współrzędnych bezwzględnych, współrzędna punktu końcowego jest oparta na przesunięciu od punktu początkowego.
5. Parametry czasu przyspieszania i zwalniania można ustawić za pomocą danych typu K lub D. W przypadku ustawienia z danymi typu D, podczas inicjalizacji instrukcja odczyta wartość z rejestrów D jako czas przyspieszania i zwalniania. Parametry czasu przyspieszania i zwalniania są 16-bitowymi liczbami całkowitymi, muszą być przypisane instrukcją MOV. Ograniczony okresem interpolacji (4 ms) algorytmu, przeskakiwanie częstotliwości będzie zbyt nagłe, aby przyspieszyć i zwalniać skutecznie, jeśli czas został ustawiony za mały. Przyspieszenie i zwalnianie w interpolacji to asymetryczne przyspieszenie i opóźnienie w kształcie litery S.
6. Parametr prędkości oznacza prędkość wypadkową osi X i osi Y, parametr ten można ustawić za pomocą danych typu K lub D. W przypadku ustawień danych typu K prędkość ścieżki nie będzie ustawialna; Jeśli zostanie ustawione dane typu D, prędkość może być regulowana przez wartość w odpowiednim rejestrze D. Również wybór MOVD i MOVF do przypisania wartości parametrów prędkości jest ustawiana z instrukcją typu ARC (ARCI/ARCF). Jeśli ekwiwalenty impulsów osi X i Y są ustawione inaczej, system ustawi domyślnie prędkość na podstawie impulsu osi X. Gdy czas przyspieszania, czas zwalniania i parametry prędkości są ustawione nieprawidłowo, system dostosuje te parametry, aby były prawidłowe.
7. Tryby rysowania łuku obejmują tryb 2-punktowy, tryb 3-punktowy i tryb współrzędnych środka, te tryby używają różnych parametrów. W trybie 2-punktowym użytkownik musi potwierdzić, którą ścieżkę przejść między bieżącym punktem a punktem końcowym, ustalając kierunek obrotu (zgodnie z ruchem wskazówek zegara lub przeciwnie do





ruchu wskazówek zegara) i długość łuku (mniejszy lub większy łuk). W trybie 3-punktowym użytkownik musi tylko potwierdzić punkt środkowy między bieżącym punktem a punktem końcowym. W trybie współrzędnych środka użytkownik musi tylko potwierdzić środek łuku.



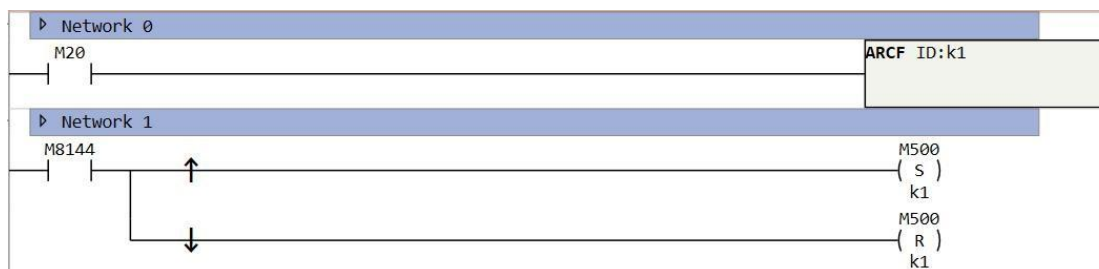
### Uwaga

1. W trybie rysowania łuku z 2 punktów długość od punktu bieżącego do punktu końcowego musi być mniejsza niż promień łuku.(rys 1)
2. W trybie rysowania łuku z 3 punktów punkt końcowy, punkt środkowy i punkt bieżący nie mogą być współliniowe.(rys 2)
3. W trybie malowania łuku za pomocą współrzędnych środka środek musi znajdować się w środkowej prostopadłości linii łączącej punkt końcowy z bieżącym.(rys 3)
4. Punkt końcowy nie może pokrywać się z bieżącym punktem.

### Przykład

Narysuj mniejszy łuk w kierunku przeciwnym do ruchu wskazówek zegara od bieżącego punktu do punktu końcowego o promieniu 100 mm, gdy rozpoczyna się interpolacja, ustawiana jest M500, gdy interpolacja jest zakończona, M500 jest resetowana. Platforma interpolacji „platform system” ustawia port Y0 jako port wyjściowy impulsów osi X, ustawia Y1 jako port wyjściowy impulsów osi Y.

### Schemat drabinkowy:



## BLOCK





### Wprowadzenie do instrukcji

1. Ta instrukcja steruje impulsem do rysowania zaimportowanej grafiki w określonej platformie interpolacji.
2. Do tej pory ta instrukcja obsługuje tylko importowaną grafikę z programu AutoCAD, który składa się z linii i łuków, dodatkowo obsługuje tylko format pliku DXF.

### Wprowadzenie do operacji

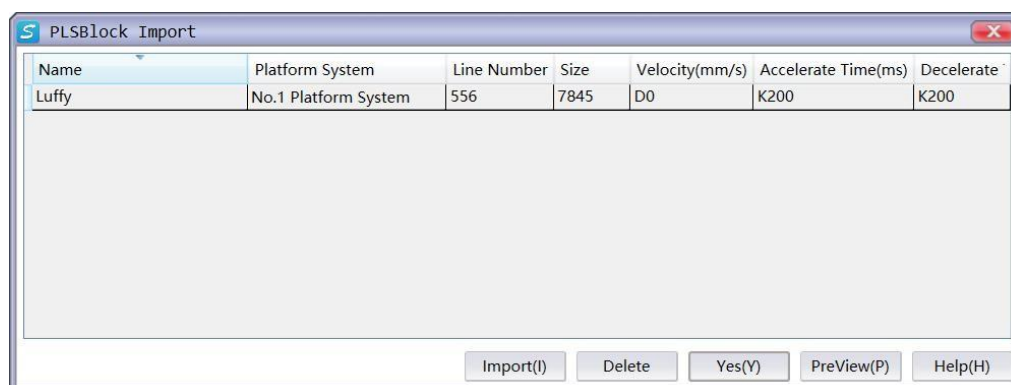
1. Po pierwsze, używając oprogramowania AutoCAD do rysowania grafiki za pomocą linii i łuków, należy zwrócić uwagę na to, że użytkownik powinien potwierdzić rozmiar platformy ruchów interpolowanych „platform system”, aby namalowany wykres nie przekroczył granicy platformy interpolacji. Wszystkie importowane grafiki są malowane ze współrzędnymi bezwzględными, więc punkt początkowy w tabeli ruchów interpolowanych „platform system” musi pokrywać się ze współrzędnymi punktu początkowego w programie AutoCAD, ustawić jednostkę rozmiaru na milimetr i sprawdzić, czy wszystkie segmenty są potrzebne do narysowania grafiki, i tu zaleca się zapisanie wykresu jako 2014 DXF.
2. Ustaw i włącz platformę ruchów interpolowanych „platform system” .
3. Importuj wykres klikając: main menu -> pulse -> import graph. (szczegóły na rysunku)
4. Po zakończeniu importu wybierz platformę ruchów interpolowanych „platform system”, w której znajduje się wykres i ustaw parametr prędkości.
5. Dodaj instrukcję **BLOCK** do wykresu drabinkowego, wybierz, który wykres malować i ustaw flagę bitową. Podczas procesu rysowania niektórych wykresów nie można ukończyć jednym pociągnięciem, między tymi częściami wykresu potrzebne są linie przejściowe. Te linie przejściowe, których nie trzeba rysować są liniami fikcyjnymi (linie kropkowane); linie, które trzeba malować są liniami rzeczywistymi (linie ciągłe). Flaga bitowa służy do rozróżnienia między liniami rzeczywistymi a liniami fikcyjnymi, pokazuje 1 w liniach rzeczywistych i 0 w liniach fikcyjnych.

### Uwaga

1. Czasy przyspieszania i zwalniania nie będą miały znaczenia, jeśli ustawiona wartość będzie zbyt mała.
2. Podczas procesu malowania parametr prędkości zostanie dostosowany automatycznie, jeśli jest niedozwolony, prędkość zostanie dopasowana zgodnie z kształtami ścieżki wykresu.
3. Rozmiar importowanego pliku wykresu nie może być zbyt duży. W przeciwnym razie nie można pobrać wykresu.

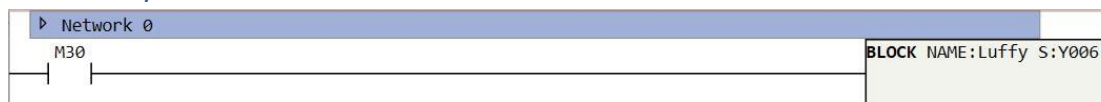
### Przykład

Narysuj wykres jak na rysunku poniżej, zaimportuj plik Luffy.dxf, ustaw flagę bitową Y6, ustaw i włącz platformę interpolacji, ustaw prędkość D0, ustaw czas przyspieszania K200, ustaw zwalnianie K200.





Schemat drabinkowy:



## PAUSE

### Wprowadzenie do instrukcji

1. Ta instrukcja może być używana tylko dla funkcji **POLYLINE** i **BLOCK**.
2. Niniejsza instrukcja obowiązuje podczas procesu rysowania **POLILINE** i **BLOK**. Wysyłanie impulsów odpowiedniej platformy ruchów interpolowanych „platform system” zostanie zatrzymane, gdy **PAUSE** jest włączone i będzie kontynuowane, gdy **PAUSE** jest wyłączone, **PAUSE** nie wpływa na ścieżki rysowania grafiki.

### Wprowadzenie do operacji

1. W pierwszej kolejności ustaw odpowiednie parametry instrukcji **POLYLINE** lub **BLOCK**.
2. Ustaw platformę ruchów interpolacji instrukcji **PAUSE** zgodnie z ustawieniami platformy ruchów interpolacji instrukcji **POLYLINE** lub **BLOCK**.
3. Włącz instrukcję **POLYLINE/BLOCK**, aby ustawić rozpoczęcie rysowania grafiki. Włącz instrukcję **PAUSE**, gdy potrzebna jest przerwa w rysowaniu, wtedy rysowanie zwolni i zatrzyma się, czas zwalniania jest zgodny z czasem zwalniania ustawionym w instrukcji **POLYLINE/BLOCK**. Wyłącz instrukcję **PAUSE**, gdy potrzebna jest kontynuacja rysowania.

### Uwaga

1. Pauza nie zawsze reaguje natychmiast, **PAUSE** nie rozpocznie się podczas przyspieszania/zwalniania rysowania, ale gdy prędkość się ustabilizuje.

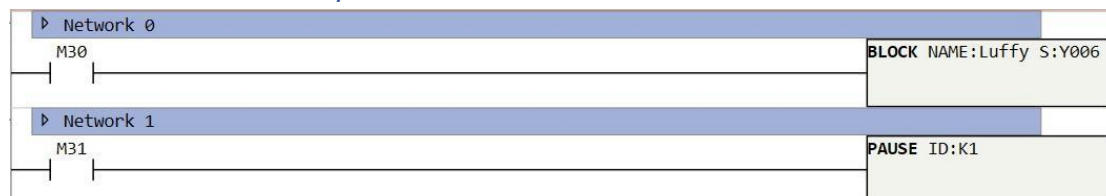


2. Podczas pauzy nie wyłączaj odpowiedniej instrukcji interpolacji, w przeciwnym razie, gdy instrukcja interpolacji jest włączona, rysowanie wykresu rozpocznie się od początku, ale nie od miejsca, w którym się zatrzymało.

3. Po pauzie instrukcja interpolacji nadal będzie zajmowała porty impulsowe i kierunkowe. Stan zatrzymania po pauzie jest niemożliwy i nie da się go usunąć.

### Przykład

#### Schemat drabinkowy:



## ZRNR

### Wprowadzenie do instrukcji

Ta instrukcja jest **instrukcją bazowania** (z opcją włączenia indeksu), stosowana aby napęd powrócił do punktu zerowego. Gdy ta instrukcja jest włączona, napęd porusza się z ujemnym kierunkiem powracając do punktu bazowego, a impulsy kroku będą generowane aż do momentu napotkania czujnika bazującego. Gdy ta instrukcja jest wyłączona, sygnał kroku zostanie natychmiastowo zatrzymany.

### Ustawienia danych

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/wyłączające		0/1	Wartość logiczna
(RS)	Prędkość bazowania.	K/H/D	-2147483648 do 2147483647	32-bitowa liczba całkowita ze znakiem
(CS)	Prędkość pełzania.	K/H/D	-2147483648 do 2147483647	32-bitowa liczba całkowita ze znakiem
(A)	Czas przyspieszenia wysyłania impulsów.	K/H/D	0 do 65535	16-bitowa liczba całkowita bez znaku
(D)	Czas zwalniania wysyłania impulsów.	K/H/D	0 do 65535	16-bitowa liczba całkowita bez znaku
(OS)	Wejście czujnika bazującego	X/M	-	Wartość logiczna
(PL)	Wejście czujnika limitu dodatniego	X/M	-	Wartość logiczna
(NL)	Wejście czujnika limitu ujemnego	X/M	-	Wartość logiczna
(OUT)	Wyjście sygnału częstotliwości (PUL)	Y	-	Wartość logiczna
(DIR)	Wyjście sygnału kierunku (DIR)	Y	-	Wartość logiczna

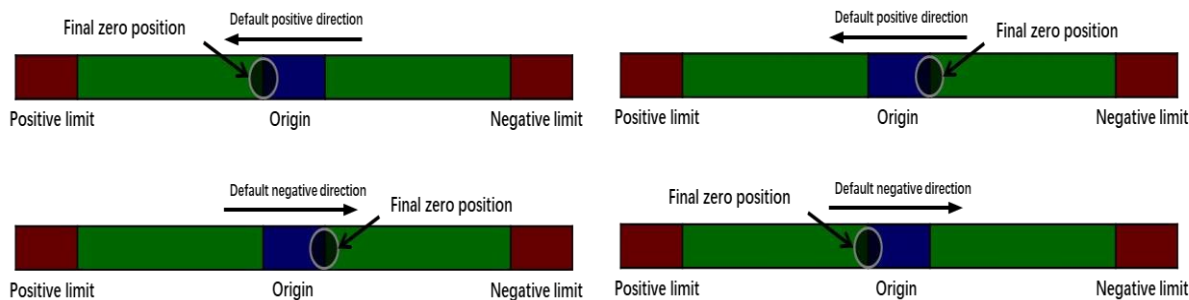


## Wprowadzenie

1. Prędkość powrotu i prędkość pełzania można wprowadzić za pomocą 32-bitowej wartości typu K/D/H(hex). Ustawienie obu parametrów prędkości na wartość ujemną spowoduje zmianę domyślnego kierunku powrotu do zera, dodatkowo zmieni się również położenie sygnału zerowego.
2. Porty wejściowe sygnału zerowego, dodatniego/ujemnego sygnału granicznego i sygnału indeksu powinny być wprowadzane za pomocą portów, które łączą się z odpowiednim czujnikiem/krańcówką (port wejściowy sygnału można wprowadzić za pomocą portu X lub rejestru M), razem z nimi dodatni/ujemny sygnał graniczny i sygnał indeksu są opcjonalne.
3. Polaryzacja sygnału zerowego i dodatniego/ujemnego sygnału granicznego powinna być określona przez rzeczywistą polaryzację sygnału odpowiedniego czujnika (normalnie otwarty lub normalnie zamknięty).
4. W większości przypadków (pozycja początkowa znajduje się poza obszarem sygnału zerowego), domyślny kierunek powrotu do zera decyduje o kierunku ruchu napędu od pozycji początkowej do zera.
5. Istnieją 2 rodzaje trybu powrotu do zera: tryb sygnału bliskiego punktu i tryb zerowego zbocza dodatniego; ścieżka powrotu do zera i pozycja 2 trybów są różne (patrz schematyczny wykres powrotu do zera).
6. Tryb powrotu do zera i domyślny kierunek powrotu do zera wspólnie decydują o ostatecznej pozycji zera (po której stronie zera znajduje się obszar sygnału).

## Uwagi

1. Po użyciu tej instrukcji do zakończenia operacji powrotu do bazy, liczniki impulsów odpowiednich portów wyjściowych zostaną automatycznie zresetowane do wartości początkowej.
2. Tryb filtrowania czujników podłączonych do portów wejściowych X powinien być wyłączony, w przeciwnym razie wystąpi duży błąd na końcowym położeniu bazowym.
3. Gdy sygnały portów pulsujących są wysokie, kierunek ruchu napędu (obroty silnika) jest dodatni, a odpowiadająca granica jest dodatnia; w przeciwnym razie kierunek jest kierunkiem ujemnym, odpowiadająca granica jest granicą ujemną.
4. Podczas powrotu do bazy, wyjścia kierunku pulsowania są sterowane automatycznie przez PLC zgodnie z rzeczywistą pozycją napędu, nie ma potrzeby ręcznego sterowania (aby upewnić się, że na wyjścia kierunku pulsowania mają wpływ inne instrukcje).
5. Jeśli domyślny kierunek powrotu do bazy jest wybrany jako dodatni, w większości przypadków (pozycja początkowa znajduje się poza obszarem sygnału zerowego) kierunek pulsowania zaczyna się od stanu wysokiego, napęd porusza się dodatnio; Jeśli domyślny kierunek powrotu do zera jest wybrany jako ujemny, jest odwrotnie. Użytkownik może wybrać odpowiedni domyślny kierunek powrotu do bazy w zależności od aktualnego przypadku (np. w której pozycji początkowej obszaru znajduje się początek powrotu do bazy).
6. Użytkownik powinien potwierdzić ostateczną pozycję bazową przed użyciem. Rysunek poniżej pokazuje końcową pozycję bazową w trybie sygnału bliskiego punktu. Końcowe położenie zerowe znajdzie się po stronie obszaru sygnału zerowego w pobliżu granicy dodatniej, jeśli domyślny kierunek powrotu do bazy jest dodatni, lub w pobliżu granicy ujemnej, jeśli domyślny kierunek powrotu do zera jest ujemny.



7. Dodatni/ujemny sygnał graniczny i sygnał indeksu są opcjonalne; jeśli żaden z 2 sygnałów granicznych nie jest wybrany, powrót do zera z dowolnej pozycji nie może być zrealizowany (ścieżki powrotu do zera, które odnoszą się do niewybranych sygnałów granicznych, nie będą realizowane, inne ścieżki niezwiązane z niewybranymi sygnałami granicznymi mogą być nadal realizowane). Użytkownik może wybrać sygnał indeksu według potrzeb (impulsowanie indeksu serwomechanizmu można podłączyć do PLC. Należy zauważyć, że sygnał musi być na poziomie 24 V, jeśli nie jest należy przekształcić go, w przeciwnym razie PLC nie rozpozna sygnału).

8. Jeśli napęd zatrzymuje się podczas wchodzenia w obszar dodatniego/ujemnego limitu podczas powrotu do zera, a komunikat o błędzie pokazuje „utrata sygnału zerowego”, sprawdź, czy czujnik sygnału zerowego działa poprawnie. Po rozwiązaniu problemu powrót do zera może nadal działać po włączeniu instrukcji. Jeśli napęd zatrzyma się podczas powrotu do zera, a komunikat o błędzie pokazuje „odległość przyspieszania i zwalniania jest zbyt długa”, oznacza to, że napęd osiągnął zasięg, a nawet przekroczył obszar sygnału granicznego, sprawdź pozycję stołu warsztatowego i zmniejsz czas przyspieszania i zwalniania.

9. Jeśli napęd nie może zwolnić do prędkości pełzania w obszarze sygnału zerowego, należy odpowiednio rozszerzyć obszar sygnału zerowego lub skrócić czas przyspieszania i zwalniania, aby zapewnić wystarczającą powierzchnię do spowolnienia napędu do prędkości pełzającej.

10. Aby zapewnić bezpieczeństwo, upewnij się, że obszar sygnału limitu jest wystarczająco szeroki, aby napęd mógł zwolnić do 0.

11. Aby zapewnić bezpieczeństwo, zaleca się wybranie polaryzacji sygnału dodatniego/ujemnego jako normalnie zamkniętego, aby uniknąć utraty skuteczności sygnału granicznego.

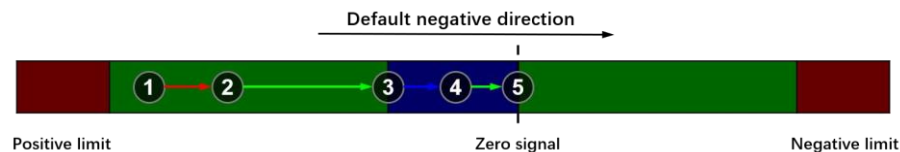
12. Na błąd powrotu mogą mieć wpływ 3 czynniki: końcowa prędkość ruchu po osiągnięciu pozycji zerowej (jeśli prędkość jest mniejsza, błąd jest mniejszy; okres skanowania PLC (jeśli okres skanowania jest krótszy, błąd jest mniejszy), bezwładność napędu (jeśli bezwładność jest mniejsza, błąd jest mniejszy).

### Schematy powrotu do zera

W przypadku ujemnego domyślnego kierunku powrotu do zera, poniższe schematyczne wykresy pokazują ruch napędu podczas powrotu do zera w 2 trybach (Na wykresach kółko w liczbie oznacza punkt podczas powrotu do zera, czerwona strzałka oznacza ruch przyspieszony, zielona strzałka oznacza ruch jednostajny, niebieska strzałka oznacza ruch spowolniony).

1. Pozycja początkowa znajduje się pomiędzy obszarem sygnału dodatniego limitu a obszarem sygnału zerowego.

- Tryb **sygnału bliskiego punktu** (Near-point signal mode), jak pokazano na rysunku:

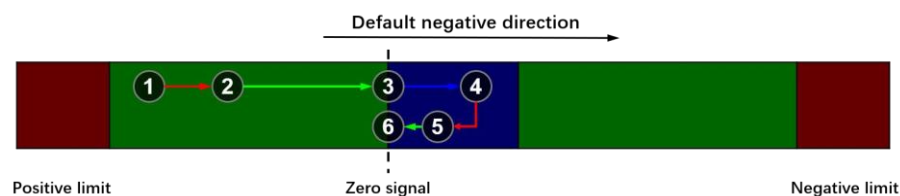


①→②: Napęd przyspiesza do prędkości powrotu w kierunku ujemnym.

③→④: Napęd wchodzi do obszaru sygnału zerowego i zwalnia do prędkości pełzania.

⑤: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.

- Tryb **narastającego zbocza punktu zero** (Zero positive-edge mode), jak pokazano na rysunku:



①→②: Napęd przyspiesza do prędkości powracania w kierunku ujemnym.

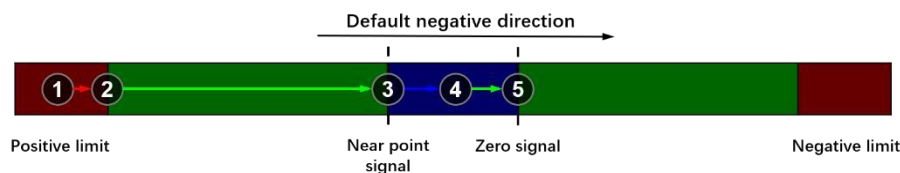
③→④: Napęd wchodzi w obszar sygnału zerowego i zwalnia do 0.

④→⑤: Napęd przyspiesza do prędkości pełzania w kierunku przeciwnym.

⑥: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.

## 2. Pozycja startowa zlokalizowana w obszarze sygnału limitu dodatniego.

- Tryb **sygnału bliskiego punktu** (Near-point signal mode), jak pokazano na rysunku:



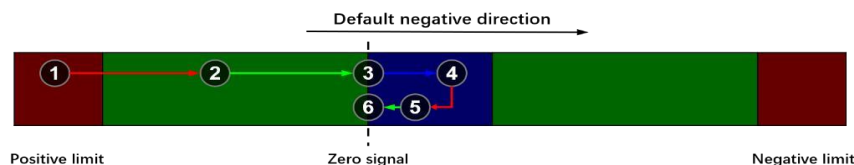
①→②: Napęd przyspiesza do prędkości powracania w kierunku ujemnym.

③→④: Napęd wchodzi w obszar sygnału zerowego i zwalnia do prędkości pełzania.

⑤: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.

- Tryb **narastającego zbocza punktu zero** (Zero positive-edge mode), jak pokazano na rysunku:





①→②: Napęd przyspiesza do prędkości powracania w kierunku ujemnym.

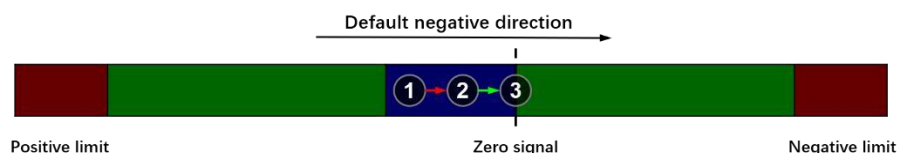
③→④: Napęd wchodzi w obszar sygnału zerowego i zwalnia do 0.

⑤→⑥: Napęd przyspiesza do prędkości pełzania w kierunku przeciwnym.

⑥: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.

### 3. Pozycja startowa znajduje się w obszarze sygnału zerowego

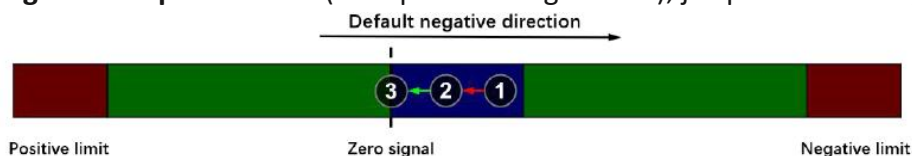
- Tryb **sygnału bliskiego punktu** (Near-point signal mode), jak pokazano na rysunku:



①→②: Napęd przyspiesza do prędkości pełzania w kierunku ujemnym.

③: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.

- Tryb **narastającego zbocza punktu zero** (Zero positive-edge mode), jak pokazano na rysunku:

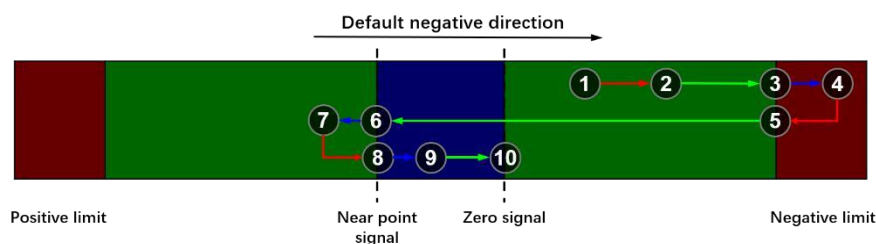


①→②: Napęd przyspiesza do prędkości powracania w kierunku ujemnym.

③: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.

### 4. Pozycja początkowa znajduje się pomiędzy obszarem sygnału zerowego a obszarem sygnału ujemnego limitu.

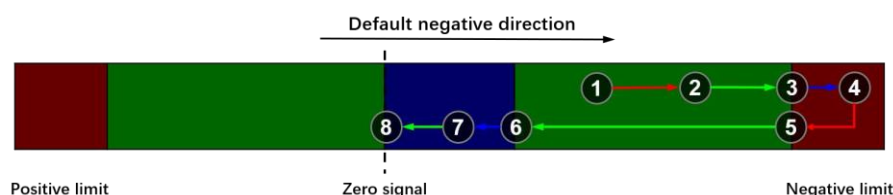
- Tryb **sygnału bliskiego punktu** (Near-point signal mode), jak pokazano na rysunku:





- ①→②: Napęd przyspiesza do prędkości powracania w kierunku ujemnym.
- ③→④: Napęd wchodzi w obszar limitu ujemnego i zwalnia do 0.
- ④→⑤: Napęd przyspiesza do prędkości powracania w kierunku dodatnim.
- ⑥→⑦: Napęd opuszcza obszar sygnału zerowego i zwalnia do 0.
- ⑦→⑧: Napęd przyspiesza do prędkości powracania w kierunku ujemnym.
- ⑧→⑨: Napęd wchodzi w obszar sygnału zerowego i zwalnia do prędkości pełzania w kierunku ujemnym.
- ⑩: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.

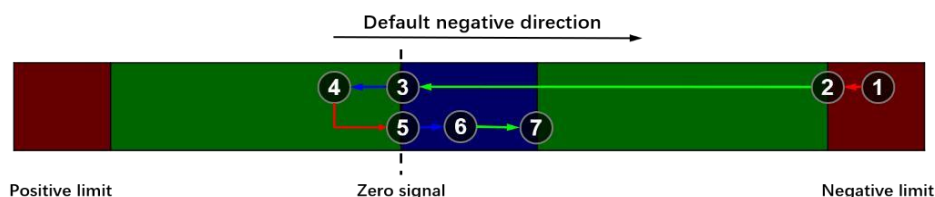
- Tryb **narastającego zbocza punktu zero** (Zero positive-edge mode), jak pokazano na rysunku:



- ①→②: Napęd przyspiesza do prędkości powracania w kierunku ujemnym.
- ③→④: Napęd wchodzi w obszar limitu ujemnego i zwalnia do 0.
- ④→⑤: Napęd przyspiesza do prędkości powracania w kierunku przeciwnym.
- ⑥→⑦: Napęd wchodzi w obszar sygnału zerowego i zwalnia do prędkości pełzania.
- ⑧: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.

5. Pozycja początkowa znajduje się w obszarze sygnału limitu dodatniego.

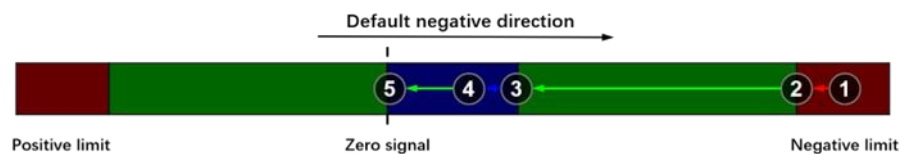
- Tryb **sygnału bliskiego punktu** (Near-point signal mode), jak pokazano na rysunku:



- ①→②: Napęd przyspiesza do prędkości powracania w kierunku dodatnim.
- ③→④: Napęd opuszcza obszar sygnału zerowego i zwalnia do 0.
- ④→⑤: Napęd przyspiesza do prędkości powracania w kierunku ujemnym.
- ⑤→⑥: Napęd wchodzi w obszar sygnału zerowego i zwalnia do prędkości pełzania w kierunku ujemnym
- ⑦: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.

- Tryb **narastającego zbocza punktu zero** (Zero positive-edge mode), jak pokazano na rysunku:





①→②: Napęd przyspiesza do prędkości powracania w kierunku dodatnim.

③→④: Napęd wchodzi w obszar sygnału zerowego i zwalnia do prędkości pełzania w kierunku dodatnim.

⑤: Napęd zatrzymuje się obok obszaru sygnału zerowego; jeśli wybrano sygnał indeksu, napęd zatrzymuje się przy sygnale indeksu.



## JOG

### Wprowadzenie do instrukcji

1. Ta instrukcja może generować impuls JOG z kierunkiem poprzez ustawienie częstotliwości i czasu przyspieszania/zwalniania. Gdy ta instrukcja jest włączona, impuls przyspiesza do określonej częstotliwości przy jednostajnym przyspieszeniu, gdy ta instrukcja jest wyłączona, impuls nie zatrzymuje się natychmiast, ale jednostajnie zwalnia do 0.
2. Ta instrukcja może automatycznie ocenić znak częstotliwości i dostosować kierunek generowania impulsów punktów wyjściowych Y. Na przykład, gdy częstotliwość wynosi 100 kHz, kierunek impulsów jest dodatni, gdy częstotliwość wynosi -100 kHz, kierunek impulsów jest ujemny; nie ma potrzeby ręcznej zmiany kierunku pulsowania.
3. Ta instrukcja zapewnia symetryczne (taki sam czas przyspieszania i zwalniania) sterowanie położeniem w kształcie litery T i w kształcie litery S; będzie to kształt litery S, gdy bit kontrolny znajdujący się w rejestrze funkcji specjalnej **D8069** wynosi 1, oraz kształt litery T, gdy bit ma wartość 0.

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające		0/1	Wartość logiczna
(F)	Częstotliwość impulsu, jednostką jest [Hz].	K/H/D	■ <b>FGs/FGRB/FGRE/FGRS</b> 0~200000 ■ <b>FGm</b> 0~500000	32-bitowa liczba całkowita bez znaku
(T)	Czas przyspieszania/ zwalniania podczas tranzytu częstotliwości.	K/D	0~65535	16-bitowa liczba całkowita bez znaku
(DIR)	Port, który wysyła kierunek impulsu.	Y	-	Wartość logiczna
(OUT)	Port wysyłający impuls.	Y	-	Wartość logiczna

### Uwaga

Ta instrukcja wyprowadza impuls z kierunkiem, dzięki czemu licznik impulsów (**D8140-D8158**) może zliczać w obu kierunkach.

### Przykład

#### Tabela instrukcji:

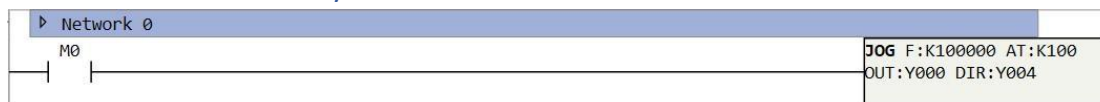


## Network 000

LD M0

JOG K100000 K100 Y0 Y4 //JOG zaczyna wysyłać impulsy, gdy M0 jest włączone  
POP

Schemat drabinkowy:



## FOLLOW

## Wprowadzenie do instrukcji

Ta instrukcja wyprowadza impuls, który zależy od zmian wartości w odpowiednim rejestrze (np. licznika). Częstotliwość impulsów zależy od tego, jak szybko zmienia się wartość, ilość impulsów zależy od tego, jak bardzo zmienia się wartość.

## Wprowadzenie do operacji

1. Najpierw (jak w przykładzie drabinkowym poniżej) ustaw pierwszy adres o wartości **D0**, punkt wyjściowy **Y0** i punkt kierunkowy **Y12**.
2. Wartość w adresie **D0** (wartość 32-bajtowa) jest wartością docelową, z której wynika impuls, w tym przykładzie wartością docelową jest **CV235**. Należy zauważyć, że jeśli wartość zmienia się zbyt szybko, odpowiadająca jej częstotliwość może być poza limitem. Dodatkowo o wartości impulsu decyduje tylko zmiana wartości docelowej, pierwotna wartość przed włączeniem instrukcji nie ma znaczenia.
3. Wartość w adresie **D2** jest parametrem wydajności, wartość jest ograniczona od 0 do 100. Gdy parametr wydajności jest duży, ruch będzie ściśle dążył do celu z krótkim opóźnieniem, postępowanie będzie ścisłe. Gdy parametr wydajności jest mały, śledzenie będzie powolne, nie tak rygorystyczne, ale bardziej elastyczne, bardziej odpowiednie do reakcji na szok. Jeśli nie ma potrzeby tak ścisłego naśladowania, zaleca się ustawienie tego parametru na 50.
4. Wartość w adresie **D3** jest parametrem współczynnika kompensacji sprężenia wyprzedzającego, wartość jest ograniczona od 0 do 100. Ponieważ śledzenie rzeczywistych ruchów jest historyczne; aby zachować dokładność pozycjonowania podczas ruchu, można dodać odpowiednią kompensację wyprzedzającą. Jeśli nie ma potrzeby dużej dokładności pozycjonowania, zaleca się ustawienie tego parametru na 0.
5. Wartość w adresie **D4** jest mnożnikiem, wariacja następującej wartości zwiększa się mnożąc przez ten współczynnik. Wartość w adresie **D5** jest współczynnikiem dzielenia, zmiana docelowej wartości zmniejsza się poprzez podzielenie tego współczynnika. Te 2 współczynniki można ustawić od 1 do 100. Wynik po mnożeniu i dzieleniu jest wartością impulsów, a wynik poza ekstremum będzie traktowany jako ekstremum.

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające		0/1	Wartość logiczna



(ARG)	Adres pierwszej komórki zawierającej parametry.	D	-	32-bitowy wskaźnik
(DIR)	Port, który wysyła kierunek impulsu.	Y	-	Wartość logiczna
(OUT)	Port wysyłający impuls.	Y	-	Wartość logiczna

### Uwaga

1. Ta instrukcja wyprowadza impuls z kierunkiem, dzięki czemu licznik impulsów (**D8140-D8158**) może zliczać w obu kierunkach.
2. Wartość docelowa nie powinna gwałtownie zmieniać się podczas śledzenia, w przeciwnym razie częstotliwość pulsu może przekroczyć maksymalny limit częstotliwości.

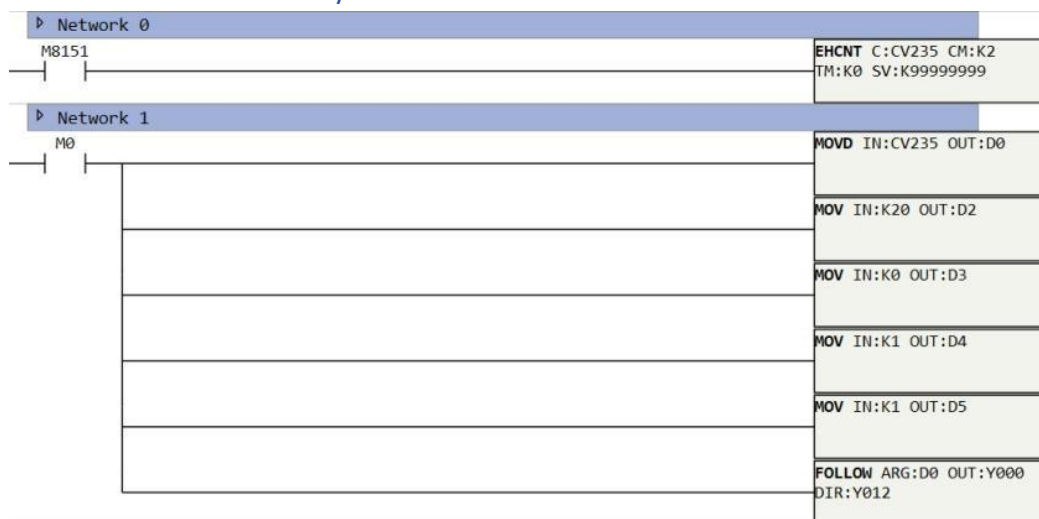
### Network 000

```
LD M8151
EHCNT CV235 K2 K0 K9999999 //włączyć licznik podczas pracy
```

### Network 001

```
LD M0
MOVD CV235 D0 //ustawić instrukcję podążania za licznikiem CV235
MOV K20 D2 //ustawić parametr wydajności
MOV K0 D3 //ustawić parametr współczynnika kompensacji wyprzedzającej
MOV K1 D4 //ustawić współczynnik mnożenia
MOV K1 D5 //ustawić współczynnik podziału
FOLLOW D0 Y000 Y012 //włącz polecenie FOLLOW, gdy M0 jest włączane
POP
```

### Schemat drabinkowy:



## ECAM

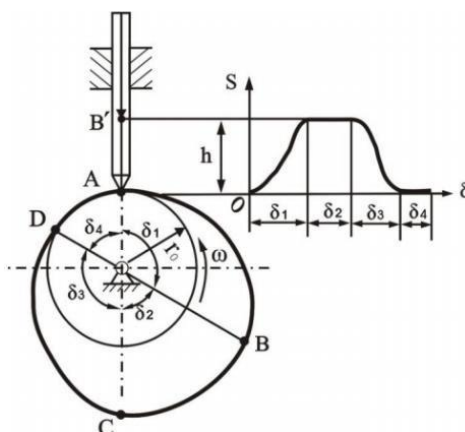
### Wprowadzenie



W przemyśle **cięcia materiałów** tradycyjne metody wykorzystują głównie zatrzymywanie przy cięciu i zatrzymywanie przy skrawaniu; metody te są nieefektywne i mogą nie spełniać wymagań procesu obróbki materiału. W branżach opakowaniowych i poligraficznych, które wymagają przejścia faz w czasie rzeczywistym, tradycyjne metody wykorzystują głównie mechaniczne CAM do realizacji.

## Mechaniczny CAM i elektroniczny CAM

Rysunek poniżej pokazuje, jak działa mechaniczne urządzenie do cięcia CAM; komponent, który napędza obrót CAM, to oś wejściowa, komponent, który popychany jest przez CAM, to oś wyjściowa. Każda pozycja osi wejściowej odpowiada jednej określonej pozycji osi wyjściowej, o odpowiadającej relacji decyduje profil CAM, który można wyrazić zależnością funkcyjną między położeniem osi wejściowej i osi wyjściowej. W rzeczywistym czasie pracy nie wszystkie części profilu CAM odgrywają kluczową rolę, tylko niektóre punkty i segmenty profilu CAM pracują w procesie. Poprzez wykorzystywanie tych punktów i odcinków, w połączeniu z odpowiednimi modelami matematycznymi, można zbudować zależności funkcyjne pomiędzy położeniem wału głównego i wału napędzanego. Na podstawie relacji funkcji położenie osi wyjściowej można obliczyć z położenia osi wejściowej, dzięki czemu PLC może wysyłać impulsy do sterowania osią wyjściową w celu przesunięcia do odpowiedniej pozycji, tak jak robi to profil CAM - tak działa elektroniczny CAM.



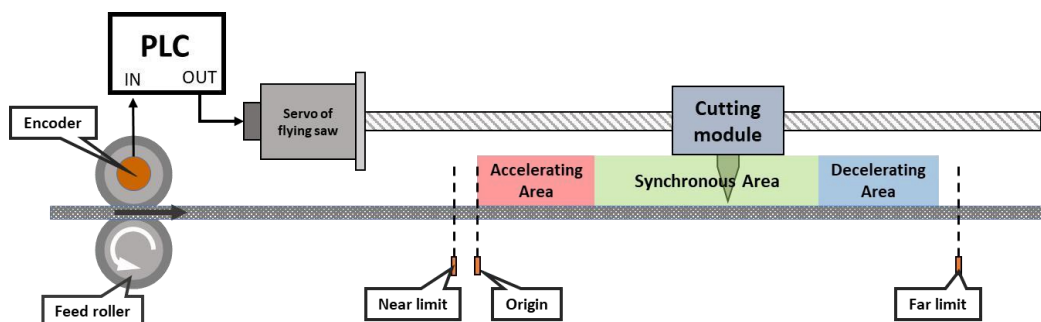
W porównaniu z mechanicznym CAM, elektroniczny CAM jest bardziej elastyczny, wygodniejszy w modyfikacji profilu i tańszy w utrzymaniu. Obecnie instrukcja ECAM ma 3 tryby cięcia w locie: tryb latającej piły (**flying saw**), tryb latających nożyc gilotynowych (**flying shear**) i **uniwersalny tryb CAM**. Wśród nich tryb latającej piły i tryb latających nożyc to 2 specyficzne scenariusze zastosowań uniwersalnego CAM, głównie do cięcia profili, cięcia papieru o stałej długości i innych. Dla tych 3 trybów instrukcja **ECAM** zapewnia interaktywny interfejs graficzny do ustawiania parametrów, dodatkowo użytkownik może przeglądać i dostosowywać krzywą ruchu CAM.

### Krótkie wprowadzenie do funkcji latającej piły

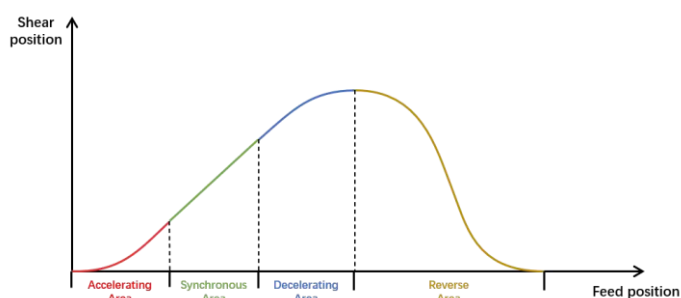
Tryb latającej piły (**flying saw mode**) w ECAM steruje napędzającym ruchem narzędzia tnącego, aby podążać za osią wejściową. Narzędzie tnące tnije materiał w obszarze synchronicznego w ruchu, a następnie zwalnia do 0 i wraca natychmiast do punktu początkowego, aby wykonać następny ruch cięcia. Ten rodzaj ruchu skrawającego może utrzymać ciągłość osi wejściowej i poprawić wydajność. Jak pokazano na rysunku poniżej, moduły latającej piły składają się z napędu posuwu osi wejściowej, sprzężenia zwrotnego położenia osi wejściowej, liniowego podążania osi wyjściowej i modułu tnącego. Zazwyczaj serwo lub przetwornik stanowi napęd posuwu osi wejściowej, enkoder



stanowi sprzężenie zwrotne położenia osi wejściowej, które przekazuje sprzężenie zwrotne aktualnej pozycji i prędkości osi wejściowej. Serwo ze śrubą pociągową stanowi liniowe podążanie za osią wyjściową, która napędza ruch modułu tnącego. Moduł tnący napędzany głównie ciśnieniem hydraulicznym, układem ciśnieniowym lub serwonapędem. W porównaniu ze wszczęciem programu latającej piły do układu serwo, użycie PLC do zrealizowania tego programu będzie łatwiejsze do modyfikacji. Instrukcja ECAM sprawia, że sterowanie latającą piłą jest bardziej wygodne i wydajne.



Główna część napędowa jest napędzana serwomechanizmem lub przetwornikiem, napędza rolkę do podawania. Pozycja i prędkość osi wejściowej są sprzężone zwrotnie z enkoderem, który łączy się z wejściem PLC. Serwo latającej piły otrzymuje sygnały impulsowe z PLC ustawione przez funkcję latającej piły i ustawia moduł tnący w pozycji znajdującej się na osi wyjściowej odpowiadającej krzywej CAM, następnie moduł tnący tnije zgodnie ze znakami w obszarze synchronicznym. Punkt początkowy służy do resetowania modułu tnącego, bliski i daleki limit służą do ochrony ograniczenia. Latająca piła musi skonfigurować długość cięcia i automatycznie generować krzywą latającej piły zgodnie z parametrami krzywej, parametrami mechanicznymi i tak dalej. Krzywa składa się z 4 segmentów: obszar przyspieszania, obszar synchroniczny, obszar zwalniania i obszar wsteczny. Te 4 segmenty prezentowane są na rysunku poniżej.

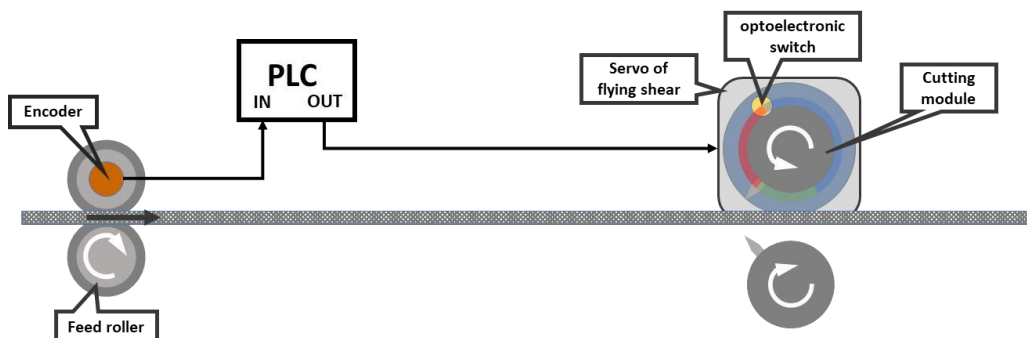


Na rysunku oś pozioma oznacza położenie osi wejściowej – podawania materiału, oś pionowa oznacza położenie osi wyjściowej – narzędzia tnącego. Długość materiału równa sumie odległości, jaką oś wejściowa przechodzi w 4 obszarze, długość obszaru synchronicznego jest równa prędkości synchronicznej (takiej samej jak prędkość osi wejściowej w stabilnym działaniu) mnoży się przez czas skrawania (musi być większy niż rzeczywisty użyty czas skrawania). Należy zauważyć, że przyspieszanie i zwalnianie będzie bardziej stabilne, jeśli obszar przyspieszania i zwalniania jest dłuższy. Gdy długość materiału jest stała, dłuższy obszar przyspieszania i zwalniania spowoduje skrócenie obszaru synchronicznego, a obszar synchroniczny musi być wystarczająco długi, aby sprostać potrzebom cięcia.

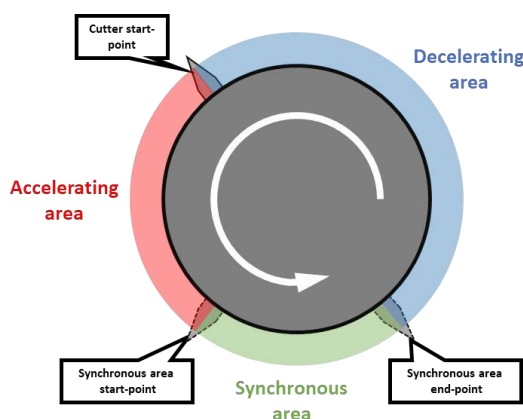


### Krótkie wprowadzenie funkcji noża obrotowego (flying shear)

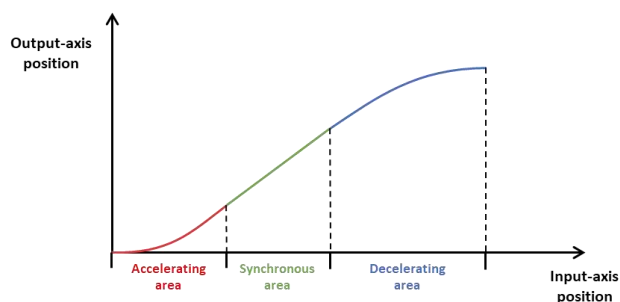
Podobnie jak latająca piła, obracające się nożyce gilotyny utrzymują synchronizację osi wyjściowej i wejściowej podczas cięcia, aby cięcie było ciągłe i wydajne. Różny jest tryb ruchu osi wyjściowej: dla latającej piły (*flying saw*) jest to **ruch liniowy**, dla obrotowego noża (*flying shear*) jest to **ruch obrotowy**. Jak pokazano na rysunku niżej, moduł obrotowego noża składa się z napędu posuwu osi wejściowej, sprzężenia zwrotnego położenia osi wejściowej oraz koła tnącego. Zwykle koło tnące jest napędzane przez serwomechanizm.



Jak pokazano na rysunku, obszar ruchu latających nożyc gilotyny jest podzielony na 3 części: obszar przyspieszania (*accelerating area*), obszar synchroniczny (*synchronous area*) i obszar zwalniania (*decelerating area*). Ponieważ ruch walcowania jest ruchem obrotowym, nie ma obszaru wstecznego.



Aby wygenerować krzywą latających nożyc, użytkownik musi skonfigurować długość cięcia, kąt obszaru przyspieszania, kąt obszaru synchronicznego, liczbę ostrzy i tak dalej. Krzywa przedstawiona na rysunku poniżej.



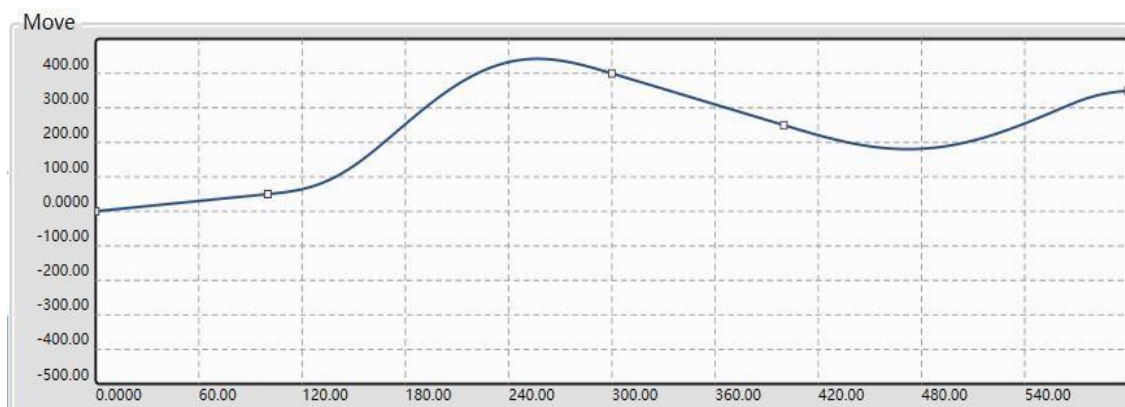
Na rysunku oś pozioma oznacza położenie osi wejściowej, oś pionowa oznacza kątowne położenie osi wyjściowej.





## Krótkie wprowadzenie do ogólnego CAM

Latająca piła i latające nożyce (*flying saw / flying shear*) to powszechne zastosowanie CAM. Dla mądrzejszego i bardziej elastycznego użytkowania istnieje ogólny CAM, który może generować krzywą ruchu za pomocą kluczowych połączeń. Jak pokazano na rysunku poniżej, użytkownik może ustawić dowolne docelowe połączenia (składające się ze współrzędnych X/Y i nachylenia), system wygeneruje krzywą ruchu CAM.



## Ustawienia danych

Parametry ECAM można podzielić na 2 części: jedna część to parametry ogólne (*general parameters*), a druga to parametry w różnych trybach. ECAM nie uruchomi się, jeśli parametry są ustawione nieprawidłowo, a rejestr kodów błędów **D8176** wyświetli odpowiedni kod błędu **46**. Możliwe przypadki nieprawidłowych parametrów zamieszczone są w [adnotacjach](#).

### Uwagi:

- INT32 oznacza 32-bitową liczbę całkowitą ze znakiem, UINT32 oznacza 32-bitową liczbę całkowitą bez znaku, FP32 oznacza 32-bitową liczbę zmiennoprzecinkową, INT16 oznacza 16-bitową liczbę całkowitą ze znakiem, UINT16 oznacza 16-bitową liczbę całkowitą bez znaku, bool oznacza bit flagi.
- **R** oznacza tylko do odczytu, **W** oznacza tylko do zapisu, **R/W** oznacza do odczytu i zapisu.
- X, Y to rejestr wejścia/wyjścia, D to rejestr podwójnych słów, K to stała, CV to rejestr szybkiego licznika, M to wewnętrzny rejestr bitów.

## Konfiguracja portu i parametry ogólne



### General parameter

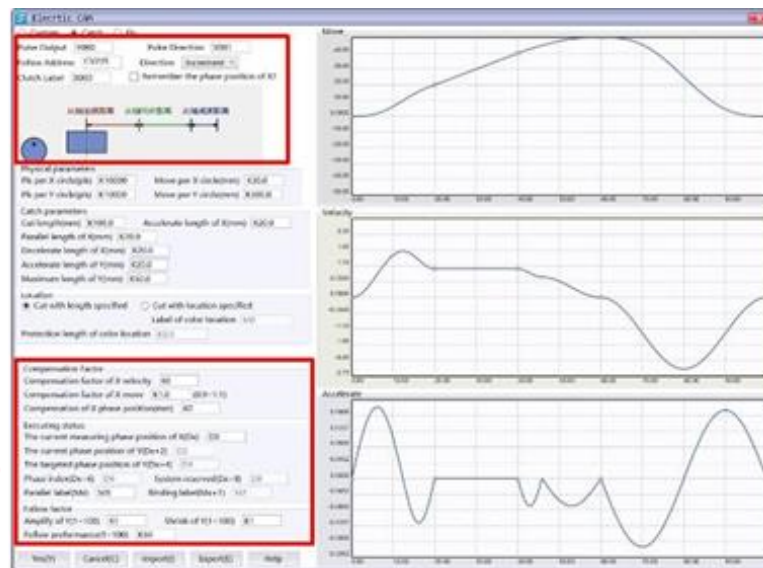


Tabela poniżej zawiera krótkie informacje o odpowiednich parametrach. Wśród parametrów parametry ogólne są podzielone na 3 typy: parametr stanu pracy, współczynnik śledzenia i współczynnik kompensacji.

	Typ parametru	Nazwa parametru	Typ danych	Ope rand	Domyślny	Zakres	Jed nos tka
Konfiguracja portu	Powiązane z impulsem	Wyjście impulsowe		Y			
		Kierunek impulsu		Y			
	Powiązane z podążaniem (follow)	Cel podążania		D/C V			
		Kierunek liczenia			Inkrementalny		
	związany ze sprzęgłem (clutch)	Bit flagi sprzęgła		X/M			
	Inne	Faza zapamiętywania osi wprowadzania			No		
Parametr ogólny	Parametr stanu pracy	Zmierzona pozycja osi wejściowej	INT32 W/R (Do odczytu podczas pracy, do zapisu po wyłączeniu)	D			pls
		Aktualna pozycja osi wejściowej	INT32 W/R	D			pls
		Docelowa pozycja osi wejściowej	INT32 W/R	D			pls
		Bieżący numer segmentu	UNIT32 W/R	D			



		System zarezerwowany	INT32 W/R	D			
		Bit flagi synchronizacji	bool W/R	M			
		Wiążący bit flagowy	bool W/R	M			
	Następujący czynnik	Współczynnik mnożenia	INT32 W	D/K	1	1~100	
		Współczynnik dzielenia	INT32 W	D/K	1	1~100	
		Wydajność do podążania	INT32 W	D/K	50	1~100	
	Współczynnik kompensacji	Współczynnik kompensacji prędkości w osi wejściowej	FP32 W		0		
		Współczynnik kompensacji przemieszczenia osi wejściowej	FP23 W		1	0.9~1.1	
		Współczynnik kompensacji fazy osi wejściowej	FP32 W		0		mm

- **Pulse output:**

Port Y PLC, który wysyła impuls.

- **Pulse direction:**

Port Y PLC, który wysyła kierunek impulsów.

- **Address of following target:**

Rejestr, który rejestruje pozycję fazową osi wejściowej, można ustawić za pomocą rejestru D lub CV.

- **Counting direction:**

Kierunek liczenia celu podążania, może być ustawiony przyrostowo lub malejąco

- **Clutch flag-bit:**

Bit flagowy sterujący sprzęgłem można ustawić za pomocą rejestru X lub M (obsługuje port X karty rozszerzeń IO, nie obsługuje portu X rozszerzonego modułu, dodatkowo port X musi być odporny na zakłócenia przez określony czas filtrowania).

- **Remembering phase position of input-axis:**

Czy zapamiętać pozycję fazy osi wejściowej, jeśli zapamiętać, bieg zostanie wznowiony od pozycji fazy, w której ECAM jest wyłączony; Jeśli nie, uruchamianie zostanie uruchomione ponownie z punktu początkowego.

- **Measured phase position of input-axis:**

Pozycja fazy, którą oś wejściowa lokalizuje w jednym cyklu, przeliczana na początku nowego cyklu. W przypadku latającej piły/nożyc pozycja fazowa oznacza liczbę impulsów sprzężenia zwrotnego odpowiedniego enkodera



podczas jednego cięcia długości; W przypadku ogólnego CAM pozycja fazowa oznacza liczbę impulsów odpowiedniego enkodera od pierwszego połączenia kluczowego do ostatniego.

- **Current output-axis phase position:**

Pozycja fazy, którą oś wyjściowa lokalizuje w jednym cyklu, przeliczana na początku nowego cyklu. W przypadku latającej piły i ogólnego CAM pozycja fazowa może być taka sama w różnych obszarach (segmentach), powinna działać z bieżącym numerem segmentu; W przypadku latającego ścinania pozycja fazowa oznacza liczbę impulsów sprzężenia zwrotnego odpowiedniego enkodera podczas długości jednego polecenia cięcia.

- **Target output-axis phase position:**

Pozycja fazowa osi wyjściowej, której odpowiada oś wejściowa na krzywej CAM; jest to cel, za którym podąża oś wyjściowa.

- **Current segment number:**

Bieżący numer segmentu całego cyklu pracy.

- **System reserved:**

Zarezerwowane rejestry systemu rejestrującego dane zajmują trzy rejestry D.

- **Synchronization flag-bit:**

Bit flagowy, który pokazuje, czy oś wyjściowa znajduje się w obszarze synchronizacji, 1- tak, 0 - nie.

- **Binding flag-bit:**

Bit flagowy, który kontroluje i pokazuje, czy oś wyjściowa jest powiązana z osią wejściowo-wyjściową, 1- tak, 0 - nie.

- **Multiplication coefficient:**

Współczynnik proporcjonalnie wzmacniający położenie fazowe osi wyjściowej.

- **Division coefficient:**

Współczynnik zmniejszający proporcjonalnie pozycję fazową osi wyjściowej.

- **Following performance:**

Parametr regulujący sztywność nadążania, im większy parametr, tym bliższe będzie nadążanie.

- **Input-axis velocity compensation factor:**

Współczynnik kompensacji, który kompensuje błąd prędkości spowodowany opóźnieniem czasowym obliczeń. System ustawi ten współczynnik automatycznie.

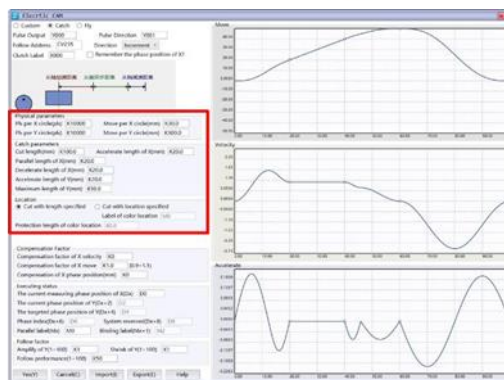
- **Input-axis displacement compensation factor:**

Współczynnik kompensacji, który kompensuje skumulowany błąd przemieszczenia spowodowany pozostałością algorytmu lub przesunięciem parametrów mechanicznych. Można go ustawić jako domyślny w trybie stałego znaku (fixed mark).

- **Input-axis phase compensation factor:**

Współczynnik kompensacji, który kompensuje wyprzedzenie lub opóźnienie pozycji fazowej osi wyjściowej. Może być również używany do regulacji pozycji cięcia.

## Parametr latającej piły



- **Output-axis synchronous angle:**



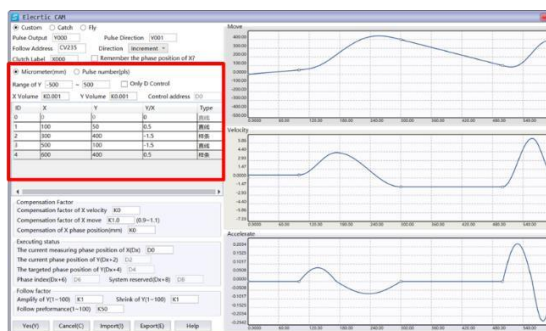
Kąt obszaru przyspieszania osi wyjściowej.

- **Output-axis accelerating angle:**

Kąt całego obszaru ruchu osi wyjściowej.

## Parametr ogólny CAM

General CAM  
parameter



Rysunek powyżej przedstawia interfejs ogólnej konfiguracji parametrów CAM, Tabela przedstawia krótkie informacje o odpowiednich parametrach.

	Typ parametru	Nazwa parametru	Typ danych	Operand	Domyślny	Zakres	Jednostka
Parametry ogólne CAM	Parametry mechaniczne	Jednostka					
		Zakres osi wyjściowej					
		Ekwiwalent impulsu osi wejściowej	FP32 W	D/K	0.01	>0	mm/pls
		Ekwiwalent impulsu osi wyjściowej	FP32 W	D/K	0.01	>0	mm/pls
	Kluczowy parametr złącza	Współrzędna osi wejściowej		D/K			
		Współrzędna osi wyjściowej		D/K			
		docelowe nachylenie		D/K			
		typ segmentu					
	Inne	Tryb sterowania					
		Adres początkowy rejestrów kontrolnych					

- **Unit:**

Jednostka współrzędnych połączenia klucowego, może być ustawiona za pomocą mm (milimetrów) lub pls (liczba impulsów).

- **Ranged of output-axis:**

Zakres wyświetlania osi wyjściowej w interfejsie krzywej CAM.



- **Pulse equivalent of input-axis:**

Odległość, na jaką oś wejściowa porusza się w ciągu jednego impulsu.

- **Pulse equivalent of output-axis:**

Odległość, na jaką oś wyjściowa porusza się w ciągu jednego impulsu.

- **Input-axis coordinate:**

Współrzędna X złącza kluczowego na krzywej CAM.

- **Output-axis coordinate:**

Współrzędna Y połączenia kluczowego na krzywej CAM.

- **Input-axis coordinate:**

Współrzędna X złącza kluczowego na krzywej CAM.

- **Target slope:**

Nachylenie w kluczowym połączeniu na krzywej CAM.

- **Segment type:**

Typ krzywej segmentu, można ustawić za pomocą linii prostej lub splajnu.

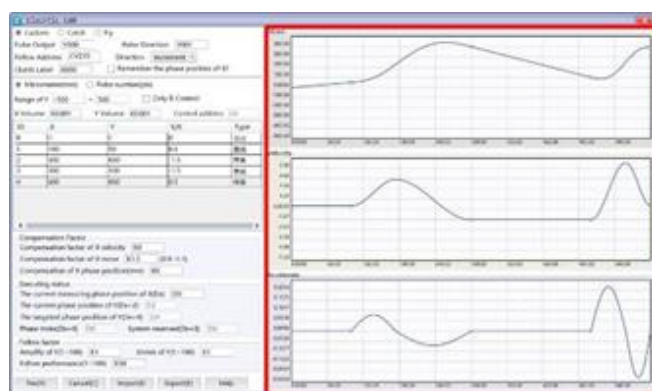
- **Control mode:**

Czy mapować parametry krzywej do rejestrów D.

- **Start address of control registers:**

Po wybraniu trybu sterowania użytkownik może ustawić adres startowy mapowanych rejestrów D.

## Obszar podglądu krzywej



Curve preview area

Jak pokazano na rysunku powyżej, oś X oznacza przemieszczenie osi wejściowej, oś Y na tych wykresach oznacza kolejno przemieszczenie, prędkość, przyspieszenie osi wyjściowej. Użytkownik może wprowadzić parametry i przeciągnąć złącze w obszarze podglądu, aby dostosować krzywą. Krzywa powinna być ustawiona płynnie, unikać nadmiernej prędkości i przyspieszenia.

## Adnotacje

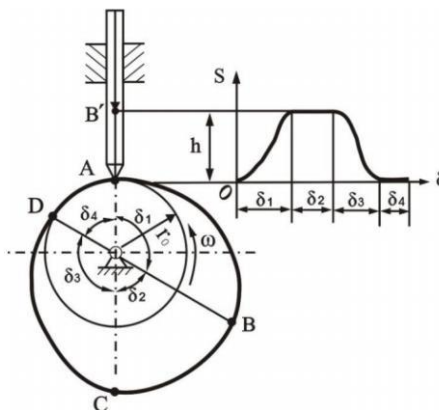
### Fixed-mark mode - Tryb znacznika

Cykl ruchu rozpoczyna się od punktu początkowego po wykryciu kolorowego znacznika (bit flagi znacznika koloru jest włączony). Jeśli odległość między dwoma kolorowymi znacznikami jest większa niż zlecona długość cięcia, rzeczywistą długością cięcia będzie odległość między znakami; Jeśli odległość między kolorowymi znacznikami jest



równa zadanej długości cięcia, rzeczywista długość cięcia będzie poleconą odległością cięcia; Jeśli odległość między kolorowymi znacznikami jest mniejsza niż zadana odległość cięcia, wykrycie drugiego kolorowego znacznika nie przyniesie efektu, ale będzie działać, dopóki odległość między następnym znacznikiem a znacznikiem początkowym nie będzie większa niż odległość zadana cięcia, w tym przypadku może być jeden lub więcej nieprawidłowych sygnałów kolorowych znaczników między 2 ważnymi kolorowymi znacznikami.

### Racjonalność ustawienia parametrów



Wszystkie ustawienia parametrów nie mogą być poza zakresem danych, powinny również spełniać określone zależności, w przeciwnym razie pojawi się raport błędu z kodem błędu 46 w **D8176**. Dla latającej piły definiujemy długość obszaru przyspieszania osi wejściowej jako **A<sub>1</sub>**, długość obszaru synchronicznego osi wejściowej jako **S**, długość obszaru zwalniania osi wejściowej jako **D<sub>1</sub>**, długość cięcia porządkujemy jako **L<sub>1</sub>**, odległość osi wyjściowej obszar przyspieszania jako **A<sub>2</sub>**, długość graniczna osi wyjściowej jako **L<sub>2</sub>**, a parametry te powinny spełniać następujące zależności:

$$A_1 + S + D_1 < L_1$$

$$A_2 + S < L_2$$

Dla latającego obrotowego narzędzia tnącego (*flying shear*) definiujemy kąt pola przyspieszenia osi wyjściowej jako **α**, kąt pola synchronicznego osi wyjściowej jako **β**, liczbę nożyc jako **K**, a parametry te powinny spełniać następujące zależności:

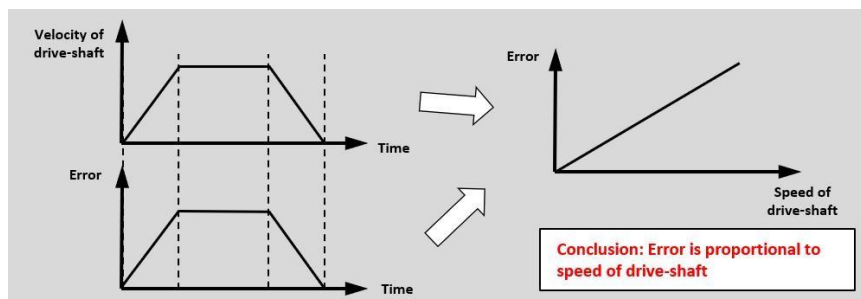
$$\alpha + \beta < (360^\circ / K)$$

### Trzy rodzaje czynników kompensacji błędów:

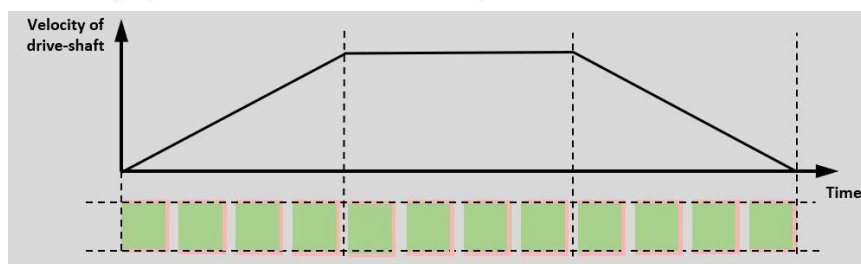
- Współczynnik kompensacji prędkości w osi wejściowej

**Opis:** ten współczynnik kompensacji jest używany do kompensacji błędu proporcjonalnie do prędkości osi wejściowej. Podczas przyspieszania, jeśli prędkość jest większa, błąd jest większy; Podczas zwalniania jest odwrotnie; Podczas ruchu jednostajnego błąd pozostaje stabilny. Zależność między błędem a prędkością przedstawia rysunek:





Error symptom shows in finished workpieces



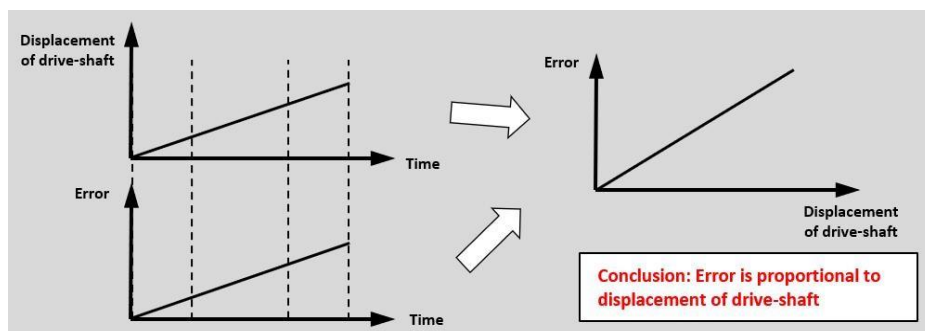
**Przyczyna:** ze względu na właściwość ECAM : błąd wynika z opóźnienia spowodowanego procesem operacyjnym.

**Rozwiązanie:** program poda odpowiedni algorytm czasu opóźnienia  $T$ , wraz z prędkością wejściowej osi  $F$ , można obliczyć kompensację  $\Delta$  równą  $T \cdot F$ . Dodaj kompensację do pozycji osi wejściowej, która była mierzona za każdym razem, aby można było skompensować błąd. Należy zauważyć, że znaczenie współczynnika kompensacji prędkości w osi wejściowej to regulacja przesunięcia  $T$ ; w większości przypadków można go ustawić na 0 i w razie potrzeby nieznacznie skorygować.

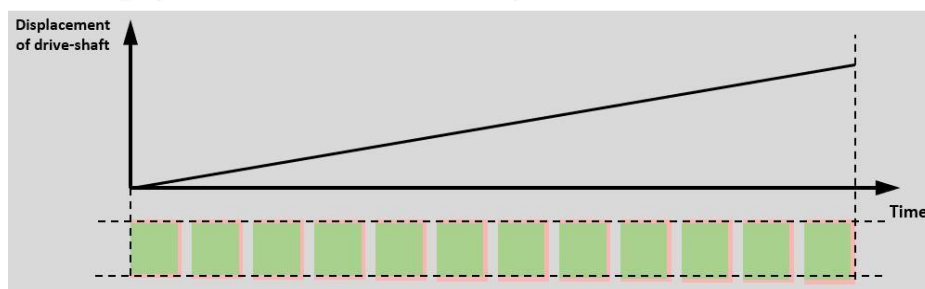
- **Współczynnik kompensacji przemieszczenia osi wejściowej**

**Opis:** ten współczynnik kompensacji jest używany do kompensacji błędu proporcjonalnie do przesunięcia osi wejściowej. Błąd będzie wzrastał wraz z ruchem osi wejściowej. Związek między błędem a przemieszczeniem przedstawia rysunek poniżej:





#### Error symptom shows in finished workpieces



**Przyczyna:** istnieją różnice między rzeczywistą strukturą mechaniczną a modelem teoretycznym.

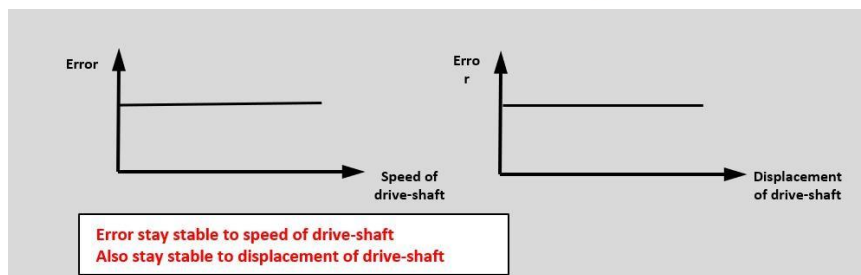
Np. długość wyprzedzenia  $L$  podczas okrążenia toczącego się koła jest równa iloczynowi średnicy koła toczącego się  $R$  i  $\pi$ . Choć  $\pi$  jest liczbą niewymierną, w rzeczywistości w obliczeniach  $\pi$  będzie używane jako wartość przybliżona. Jeśli  $\pi$  zostanie zaokrąglone do 3,14159, a  $D$  wynosi 100 mm, wystąpi błąd 0,000265358979 mm w każdej rundzie i zostanie on skumulowany do 2,65368979 mm w 10000 cyklach.

**Rozwiązanie:** Ponieważ błąd jest proporcjonalny do całej odległości ruchu, możemy uzyskać proporcjonalną zależność  $K$  w bieżącym procesie i kompensację ( $K \cdot L$ ) w każdym okrążeniu, aby zrekompensować błąd. Należy zauważyć, że w trybie o stałej długości współczynnik jest bardzo bliski 1; W trybie stałego znaku współczynnik wynosi domyślnie 1 i można go zignorować, jeśli nie ma takiej potrzeby.

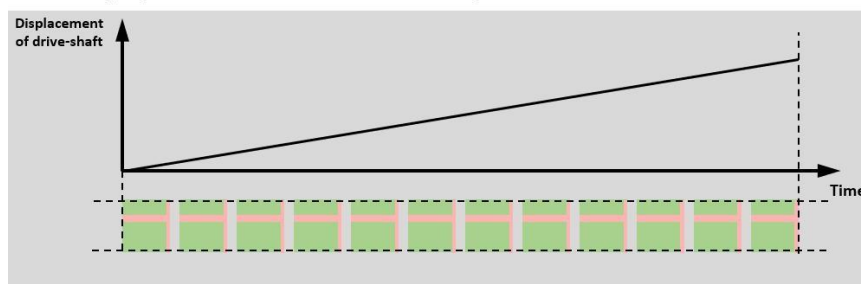
#### Współczynnik kompensacji fazy osi wejściowej

**Opis:** ten współczynnik kompensacji jest używany do kompensacji błędów proporcjonalnie do fazy osi wejściowej.

Błąd nie ma związku z prędkością i przemieszczeniem osi wejściowej, wynika z przesunięcia fazowego w pierwszym skrawaniu. Symptom błędu odnosi się do rysunku:

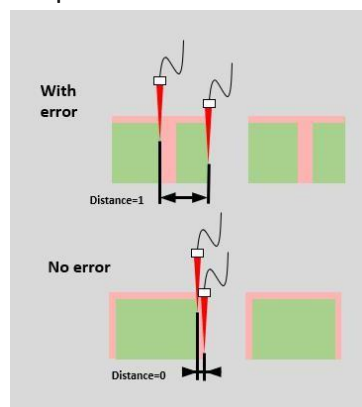


#### Error symptom shows in finished workpieces



**Przyczyna:** Istnieje odchylenie fazy między osią wejściową a osią wyjściową, zawsze ma to miejsce przy pierwszym uruchomieniu.

**Rozwiązanie:** Dodaj początkową kompensację fazy w bieżącym procesie osi wejściowej, aby skompensować błąd. Jeśli współczynnik kompensacji jest odchyleniem w pierwszym skrawaniu: jeśli jest dodatni, cięcie będzie przyspieszone, jeśli jest ujemny, cięcie będzie opóźnione.



### Ustawianie parametrów ogólnego ECAM

Parametry ogólnego ECAM powinny spełniać pewne określone zasady i mogą być dostosowywane automatycznie, jeśli są niezgodne:

1. Jeżeli kierunek zliczania wartości docelowej osi wejściowej jest inkrementalny, współrzędna osi wejściowej bieżącego segmentu musi być większa niż współrzędna osi wejściowej ostatniego segmentu.
2. Jeśli współrzędna Y dwóch sąsiednich złączy jest taka sama, ich nachylenie zostanie automatycznie ustawione na 0.
3. Jeśli typem krzywej bieżącego odcinka jest linia prosta, współrzędne nachylenia i osi wyjściowej muszą być dopasowane, w przeciwnym razie zostaną dostosowane automatycznie.



4. Jeśli typem krzywej bieżącego segmentu jest linia prosta, a typem krzywej ostatniego segmentu jest splajn, nachylenie ostatniego segmentu zostanie dostosowane tak, aby było takie samo jak w bieżącym segmencie, aby zapewnić płynne przejście.

## Clutch (sprzęgło)

Bit flagowy sprzęgła może być ustawiony jako rejestr X lub M. Gdy bit flagi sprzęgła jest ustawiony, powiązanie między osią wejściową i wyjściową zostanie zerwane bez wyłączania instrukcji, w ten sposób ruch osi wyjściowej może być kontrolowany przez inne instrukcje impulsu. Gdy bit flagi sprzęgła zostanie zresetowany, praca będzie kontynuowana od fazy przerywania wiązania (jeśli była zapamiętana) osi wejściowej i osi wyjściowej.

Należy zauważyć, że przy włączonym sprzęgle, jeśli nie jest to szczególnie konieczne, zaleca się utrzymywanie osi wejściowej w bezruchu, gdy sprzęgło jest włączone, w przeciwnym razie długość pierwszego cięcia materiału po zresetowaniu sprzęgła będzie miała odpowiednie przesunięcie. Jeśli oś wyjściowa jest kontrolowana przez inną instrukcję impulsową, reset bitu flagi musi nastąpić po wyłączeniu instrukcji, w przeciwnym razie powiązanie między osią wejściową a osią wyjściową nie zostanie przywrócone, jeśli pozycja osi wyjściowej ulegnie odchyleniu od sprzęgła WŁ. do sprzęgła WYŁ., nastąpi odchylenie całej jazdy osi wyjściowej, dla latającej piły może to spowodować najechanie. Jeśli wyłączysz instrukcję, gdy sprzęgło jest włączone bez zapamiętania fazy osi wejściowej, oś wejściowa wznowi bieg od początku, co może również spowodować odchylenie całego ruchu osi wyjściowej.

## Parametry stanu pracy

Parametry pracy są przechowywane w kolejnych rejestrach D (double-WORD) i rejestrach bitowych M. Jeśli istnieje potrzeba zapisania pozycji pracy, gdy instrukcja jest wyłączona, użytkownik może розміścić te rejestry w strefie referencyjnej, tak aby ECAM mógł automatycznie wznowić pracę od pozycji zapisanej w tych rejestrach, a nie od początku. Użytkownik może również zresetować te rejestry na schemacie drabinkowym, aby ponownie uruchomić ECAM od początku bez zapamiętywania fazy wejściowej osi.

Dodatkowo, te rejestry są tylko do odczytu, gdy instrukcja ECAM jest włączona, do odczytu i zapisu, gdy instrukcja ECAM jest wyłączona, więc operacja resetowania może zostać wykonana tylko wtedy, gdy instrukcja ECAM jest wyłączona.

## Aplikacja

Ten rozdział przedstawia głównie kroki korzystania z instrukcji **ECAM**:

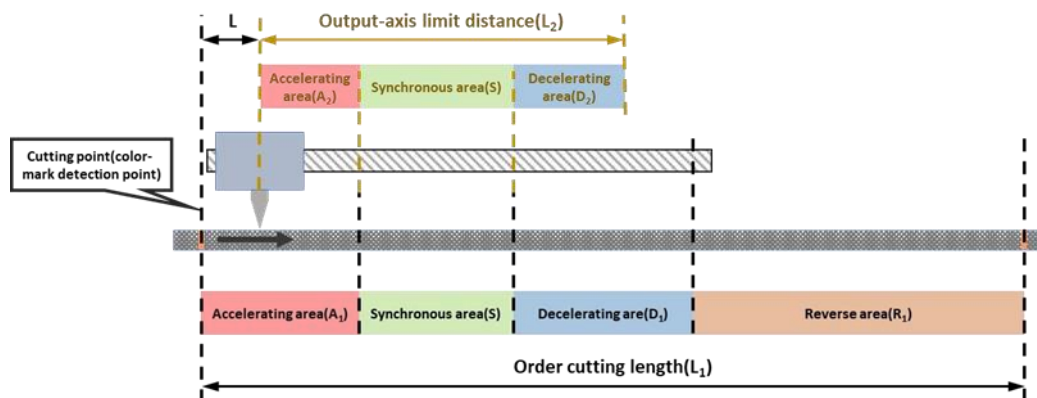
1. Sprawdź, czy port impulsowy PLC jest sprawny, serwomechanizm działa prawidłowo i upewnij się, że okablowanie jest prawidłowe.
2. Skonfiguruj port pulsujący i adres wartości docelowej. Potwierdź, że wartość docelowa ruchu do przodu osi wejściowej jest zwiększana lub zmniejszana. Skonfiguruj bit flagi sprzęgła, podłączając sygnał sprzęgła do portu X lub mapując do rejestru M. Potwierdź, czy konieczne jest zapamiętanie bieżącej fazy osi wejściowej i wyjściowej, jeśli to konieczne, parametry stanu pracy powinny być ustawione na strefę referencyjną, aby instrukcja ECAM mogła zostać uruchomiona ponownie od ostatniej pozycji, w której zatrzymuje się po wyłączeniu zasilania.



3. Skonfiguruj parametry mechaniczne ECAM, takie jak liczba impulsów i odległość do przodu każdej tury. Parametry te muszą być jak najdokładniejsze, w przeciwnym razie mogą mieć wpływ na długość cięcia lub obszar synchroniczny.
4. Wybierz tryb stałego znaku trybu o stałej długości zgodnie z rzeczywistym przypadkiem i ustaw parametry latającej piły lub obrotowego noża:

### Latająca piła

Zakładając, że ustawianie parametrów jest racjonalne, celem ustawiania parametrów jest zapewnienie stabilnego przyspieszenia modułu tnącego i więcej miejsca na obszar synchroniczny do zakończenia cięcia. Rysunek poniżej pokazuje parametry latającej piły.



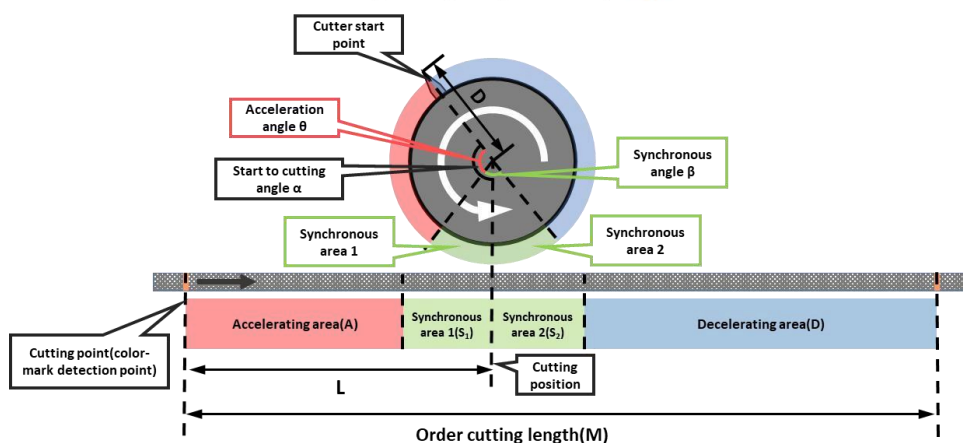
Ustaw odległość ruchu osi wejściowej w obszarze przyspieszania jako  $A_1$ , odległość ruchu osi wyjściowej w obszarze przyspieszania jako  $A_2$ , odległość między pozycją osi wejściowej a pozycją osi wyjściowej w początku układu współrzędnych jako  $L$  oraz  $L = |A_1 - A_2|$ . Przy  $L$  punkt skrawania może być ustawiony (zalecane jest ustawienie  $A_1$  powyżej  $A_2$ , w przeciwnym razie oś wyjściowa będzie zwalniać w obszarze przyspieszania, co nie jest stabilne i energooszczędne).

Jeśli wybrano tryb stałej długości, zgodnie z długością  $L$ , użytkownik powinien upewnić się, że materiał jest wystarczająco długi, aby sprostać pierwszemu cięciu.

Jeśli wybrany jest tryb stałego znaku, gdy oś wyjściowa jest w punkcie początkowym, jeśli kolorowy znak znajduje się w punkcie wykrywania, zaznaczony punkt będzie punktem cięcia. Użytkownik może również ustawić pewne przesunięcie na znaku w celu przesunięcia punktu cięcia zgodnie z rzeczywistym przypadkiem.

- **Latające nożyce**

Zakładając, że ustawienie parametrów jest racjonalne, celem ustawienia parametrów jest zapewnienie stabilnego przyspieszenia toczącego się koła i więcej miejsca na obszar synchroniczny do zakończenia cięcia. Rysunek poniżej pokazuje parametry cięcia w locie nożem obrotowym.



Na rysunku  $\alpha$  jest to kąt między pozycją początkową ostrza a pozycją cięcia,  $\beta$  jest kątem obszaru synchronicznego osi wyjściowej,  $\theta$  jest kątem obszaru przyspieszania osi wyjściowej.  $M$  jest wyznaczoną długością skrawania,  $A$  jest długością obszaru przyspieszania osi wejściowej,  $S_1$  jest długością obszaru synchronicznego osi wejściowej przed pozycją cięcia,  $L$  jest odległością między początkiem osi wejściowej a pozycją cięcia.  $D$  jest średnicą ostrza,  $i$  jest liczbą frezów. Parametry te powinny odpowiadać następującemu wzorowi:

- $A = M\theta / (360^\circ / i)$
- $S_1 = ((\alpha - \theta) / 360^\circ) \pi D, (\alpha > \theta)$
- $L = A + S_1$
- $\theta + \beta < (360^\circ / i), (\theta < \alpha - \theta < \beta)$

W przypadku wybrania trybu o stałej długości należy ustawić odpowiednie  $\beta$  i  $\theta$  zgodnie ze wzorem (d), aby zapewnić stabilne przyspieszenie. Uzyskaj  $L$  zgodnie ze wzorem (a)(b)(c), aby użytkownik mógł upewnić się, że materiał jest wystarczająco długi, aby sprostać pierwszemu cięciu. Jeśli wybrany jest tryb stałego znaku, punkt wykrywania znaku jest zawsze ustawiany w punkcie cięcia. Użytkownik może również ustawić pewne przesunięcie na znaku w celu przesunięcia punktu cięcia zgodnie z rzeczywistym przypadkiem.

5. Ustaw odpowiedni współczynnik zgodnie z aktualnym przypadkiem i potrzebą, szczegóły odnoszą się do instrukcji FOLLOW.

6. Napisz program ręczny, aby sterować ruchem osi wejścia/wyjścia i zresetować oś wyjściową do początku.

7. Wykonaj jazdę próbną po zakończeniu ustawiania, obserwuj synchronizację prędkości w obszarze synchronicznym. Jeśli błąd synchronizacji jest mały, można rozpocząć cięcie próbne, jeśli błąd synchronizacji jest duży, użytkownicy muszą sprawdzić, czy ustawienia parametrów są prawidłowe.

8. Po próbnym cięciu długość cięcia może odbiegać od ustawionej, błąd może być spowodowany niezgodnością parametrów mechanicznych z rzeczywistym stanem. Użytkownik może mierzyć bieżącą długość podczas okrążeń o określonej liczbie, obliczać i resetować parametry mechaniczne.

### Uwagi

- Modyfikacja parametrów nie działa podczas wykonywania instrukcji, ale po jej ponownym uruchomieniu (włączenie po wyłączeniu).
- Jeśli parametry zostaną ustawione nieracjonalnie, po włączeniu instrukcja nie zostanie wykonana, ale zgłosi błąd. Użytkownik musi sprawdzić i zmodyfikować te parametry i ponownie uruchomić instrukcję.



3. Użytkownik może użyć flagi obszaru synchronicznego do wyzwolenia cięcia i użyć sygnału flagi opadającego zbocza obszaru synchronicznego do zliczania.
4. Lepiej jest utrzymywać stałe przyspieszanie/zwalnianie, w przeciwnym razie może to powodować wibracje konstrukcji mechanicznej i wpływać na pomiar enkodera, a nawet błąd przekroczenia z powodu nadmiernej prędkości.
5. Zerowy powrót latającej piły i latających nożyc gilotyny jest inny, zerowy powrót latających nożyc wymaga wybrania ścieżki powrotu (zgodnie z ruchem wskazówek zegara lub przeciwnie do ruchu wskazówek zegara), aby uniknąć kolizji.
6. Nie czyść ani nie modyfikuj rejestru wartości docelowej podczas wykonywania instrukcji, w przeciwnym razie najprawdopodobniej spowoduje to błąd przekroczenia z powodu nagłej zmiany wartości docelowej osi wejściowej.
7. Wykrywanie kolorowych znaczników nie zadziała, gdy sprzęgło jest włączone. Zanim sprzęgło zostanie wyłączone, wszystkie instrukcje sterujące osią wyjściową muszą zostać wyłączone.
8. W przypadku konieczności ponownego uruchomienia krzywej CAM i zapamiętanie ustawionej fazy wejściowej osi jest ustawione, użytkownik może wyłączyć instrukcję ECAM, wyczyścić wszystkie parametry stanu pracy i ponownie ją uruchomić.

## EDRVI/EDRVA

### Wprowadzenie do instrukcji

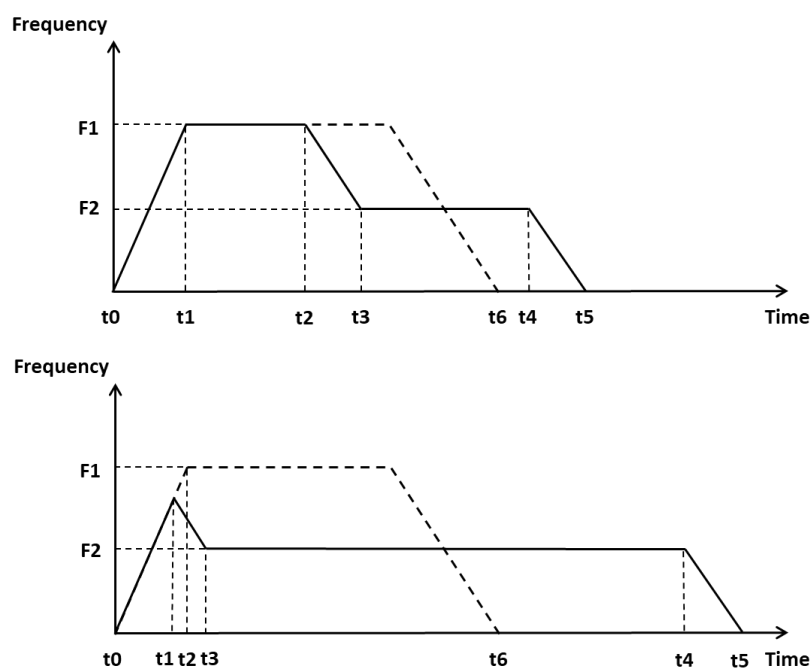
1. Te instrukcje wysyłają impuls z regulowaną częstotliwością i docelową ilością impulsów.  
W przypadku EDRVI regulacja częstotliwości i docelowej liczby impulsów jest względna; W przypadku EDRVA regulacja częstotliwości i docelowej liczby impulsów jest bezwzględna.
2. Gdy instrukcja jest włączona, przyspieszy do ustawionej częstotliwości w ustawionym czasie przyspieszania, zwolni do 0 w ustawionym czasie zwalniania i osiągnie ustawioną docelową liczbę impulsów w procesie wyprowadzania. Gdy instrukcja jest wyłączona, wysyłanie impulsów zostanie natychmiast zatrzymane.  
Jeśli użytkownik zmodyfikuje częstotliwość i liczbę impulsów docelowych podczas wysyłania, wysyłanie dostosuje się do zmodyfikowanej wartości przy przyspieszaniu/zwalnianiu.
4. Ta instrukcja może być kontrolowana przez specjalny rejestr funkcji M8068, gdy M8068 jest włączony, wyjście portu kierunkowego odwróci się.
5. Ta instrukcja może być ustawiona w trybie krzywej za pomocą specjalnego rejestru funkcji M8069, gdy M8069 jest włączony, krzywa częstotliwości ma kształt litery S, gdy jest wyłączona, krzywa częstotliwości ma kształt litery T.
6. Gdy docelowy numer impulsu jest ustawiony na 0x7FFFFFFF, impuls będzie wysyłany w kierunku dodatnim; Gdy numer impulsu docelowego jest ustawiony na 0x80000000, impuls będzie wysyłany w kierunku ujemnym. Jeśli instrukcja zaczyna się od tych 2 rodzajów ustawienia liczby impulsów docelowych, regulacja docelowej liczby impulsów podczas wysyłania nie przyniesie żadnego efektu.
7. Gdy częstotliwość jest ustawiona na 0 podczas wysyłania, wysyłanie impulsów zostanie natychmiast zatrzymane bez zwalniania i będzie kontynuowane z przyspieszeniem, gdy częstotliwość zostanie ustawiona na nową wartość różną od zera.



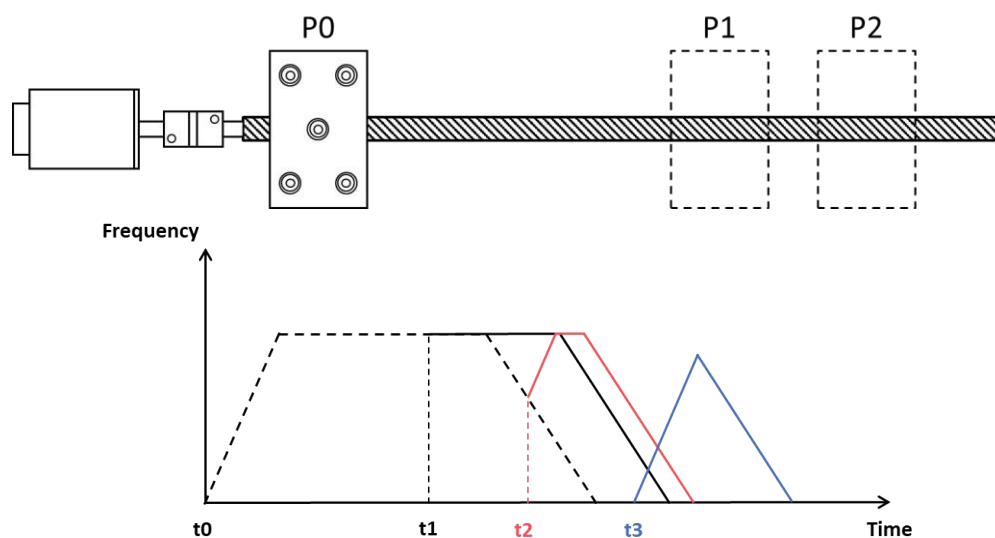


## Wprowadzenie do operacji

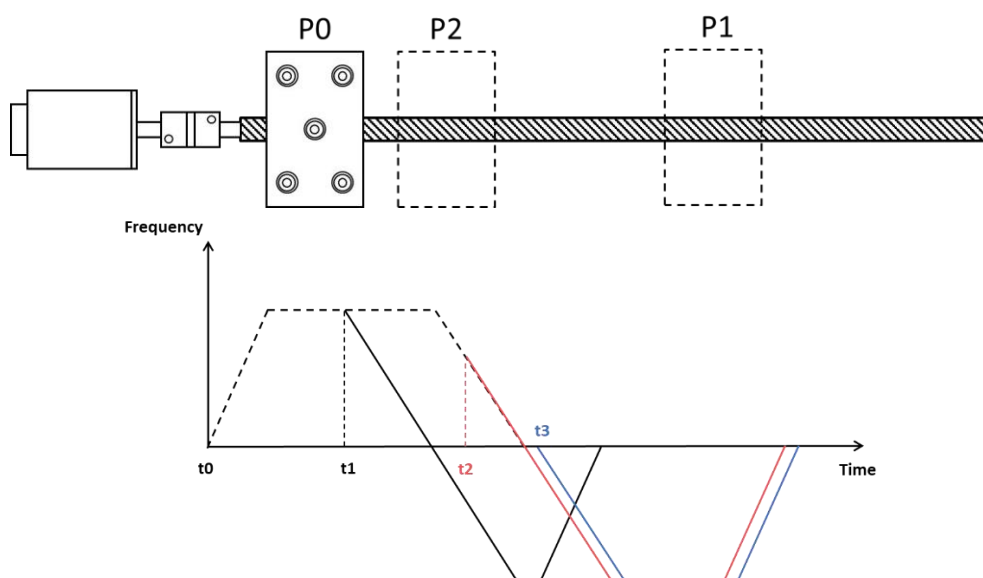
1. Ustaw częstotliwość, docelową ilość impulsów, czas przyspieszania i czas zwalniania, a następnie włącz instrukcję. Jeśli nie wykonujesz regulacji, ta instrukcja działa jak EDRV. Jeśli dostosujesz częstotliwość lub parametr docelowej ilości impulsów zapisany w odpowiednich rejestrach D, w następnym okresie, gdy wykryje zmianę parametrów, system przeliczy krzywą częstotliwości, dodatkowo regulacja docelowej ilości impulsów może spowodować odwrócenie silnika, szczegóły zostaną wprowadzone w późniejszych częściach. Dopasowanie działa tak długo, jak długo instrukcja jest włączona, pomimo zakończenia wydawania impulsów. Dodatkowo regulacja czasu przyspieszania/zwalniania podczas wysyłania impulsu nie ma żadnego efektu.
2. Na rysunkach poniżej linia przerywana to krzywa bez regulacji, linia ciągła to krzywa z regulacją. Na rysunku pierwszym częstotliwość jest dostosowywana w czasie  $t_2$  od  $F_1$  do  $F_2$ , gdy częstotliwość jest stabilna, na rysunku drugim częstotliwość jest dostosowywana do  $F_2$  w czasie  $t_1$ , gdy częstotliwość przyspiesza. W obu przypadkach wysyłanie impulsów nadal osiąga docelową ilość impulsów bez regulacji.



Regulacja liczby impulsów docelowych służy do regulacji pozycji. Na pierwszym rysunku poniżej  $P_0$  to początek,  $P_1$  to pozycja docelowa bez regulacji docelowej liczby impulsów,  $P_2$  to pozycja docelowa z dostosowaną docelową liczbą impulsów, a  $P_2$  jest po  $P_1$ . System dostosuje częstotliwość, aby osiągnąć nową docelową liczbę impulsów. Rysunek drugi pokazuje 3 rodzaje przypadków (czarna linia oznacza regulację podczas stabilnej częstotliwości, czerwona linia oznacza regulację podczas zwalniania, niebieska linia oznacza regulację po zakończeniu wysyłania impulsu).



Rysunki poniżej pokazują przypadek, w którym P2 jest przed P1.



#### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające		0/1	Wartość logiczna
(F)	Regulowana częstotliwość wysyłania impulsów.	K/H/D	<ul style="list-style-type: none"> <li>■ FGs/FGRB/FGRE/FGRS</li> <li>0~200000</li> <li>■ FGm</li> </ul>	32-bitowa liczba całkowita bez znaku





			0~500000	
(P)	Docelowa liczba impulsów wyjścia impulsowego, która jest regulowana	K/H/D	0~2147483647	32-bitowa liczba całkowita bez znaku
(A)	Czas przyspieszenia wyjścia impulsowego	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku
(D)	Czas zwalniania wyjścia impulsowego	K/H/D	0~65535	16-bitowa liczba całkowita bez znaku
(DIR)	Port wyprowadzający kierunek impulsu	Y	-	Wartość logiczna
(OUT)	Port wysyłający impuls.	Y	-	Wartość logiczna

Ta instrukcja wyprowadza impuls z kierunkiem, dzięki czemu licznik impulsów (**D8140-D8158**) może zliczać w obu kierunkach.



## INSTRUKCJE SZYBKIEGO ZLICZANIA

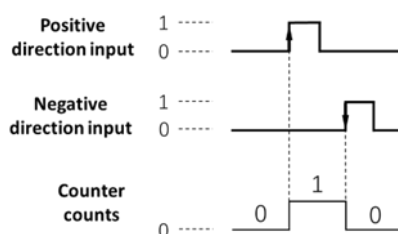
### Uwagi

1) Instrukcje szybkiego zliczania są przeznaczone do zliczania szybkich impulsów, a do szybkiego zliczania impulsów używane są specjalne liczniki **CV** (CV235 do CV255).

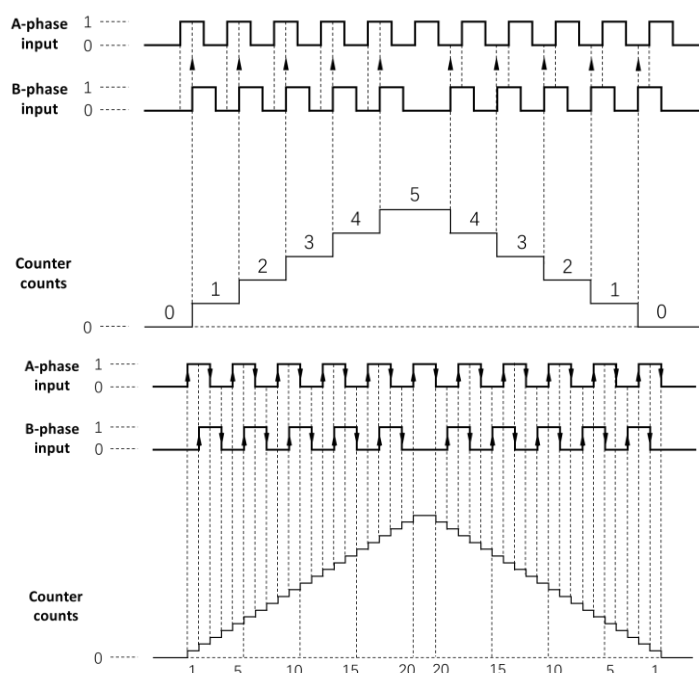
2) istnieją trzy rodzaje trybów liczenia dla szybkich instrukcji: liczenie jednokierunkowe, liczenie dwukierunkowe i liczenie fazowe **AB**:

- Zliczanie jednokierunkowe zlicza impulsy jednego wejścia w jednym kierunku (dodatnim lub ujemnym).
- Zliczanie dwukierunkowe zlicza impulsy z dwóch wejść w dwóch kierunkach (dodatnim i ujemnym).

Szczegóły odnoszą się do rysunku:



- Zliczanie fazowe AB (obsługa enkoderów) zlicza impulsy stanów A i B silnika, dostępne są 2 tryby zliczania AB: tryb zliczania w logice x2 i tryb zliczania w logice x4. Szczegóły dotyczące 2 trybów znajdują się na rysunkach poniżej:



## HCNT



## Wprowadzenie do instrukcji

1. Ta instrukcja obsługuje większość 4 portów wejścia impulsowego, więc seria FGm nie obsługuje tej instrukcji. W przypadku serii, które obsługują tylko 2 szybkie porty impulsowe (FGs\_16MT/FGs\_32MT/FGRB\_C8X8T), tylko X0 i X1 mogą być używane do szybkiego zliczania.
2. Licznikom szybkim przypisano w niniejszej instrukcji określony tryb zliczania, szczegóły patrz tabela poniżej (w tabeli **U** oznacza zliczanie w górę, **N** oznacza zliczanie w dół, **A** oznacza stan **A**, **B** oznacza stan **B**).

	Rejestr szybkiego licznika	Port szybkiego wejścia impulsowego			
		X000	X001	X002	X003
Liczenie jednokierunkowe	CV235	U			
	CV236		U		
	CV237			U	
	CV238				U
Liczenie dwukierunkowe	CV246	U	D		
	CV248			U	D
Zliczanie fazowe AB	CV251	A	B		
	CV252			A	B

3. Zakres zliczania wynosi od -2 147 483 648 do 2 147 483 647, gdy zliczanie jest poza zakresem, wystąpi przepełnienie lub niedopełnienie.
4. Tryb zliczania fazowego AB - zliczanie jest kontrolowane przez rejestry funkcji specjalnych M8051 i M8052, które sterują trybem zliczania CV251 i CV252. Gdy bit kontrolny wynosi 0, jest tryb zliczania w logice x2; Gdy bit kontrolny ma wartość 1, jest to tryb zliczania w logice x4.

## Ustawienia danych

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(C)	Adres wybranego rejestru liczników.	CV	-	32-bitowy wskaźnik
(SV)	Docelowy numer odpowiedniego licznika.	K/H/D	- 2147483648~2147483647	32-bitowa liczba całkowita ze znakiem



### Uwagi

1. Podczas podłączania wejścia do portu, w celu poprawy przeciwdziałania zakłóceniom, zaleca się dodanie linii ekranujących na linii wejściowej i podłączenie linii ekranującej do uziemienia.
2. Gdy używany jest jeden szybki licznik, port wejściowy, który zajmuje, nie może być używany do innego licznika, nie można go również zastosować do wspólnego użytku wejściowego.
3. Tryb zliczania faz AB należy ustawić przed uruchomieniem instrukcji, modyfikacja podczas pracy nie ma znaczenia.

### Przykłady

#### 1. Zliczanie jednokierunkowe

#### Tabela instrukcji

#### Network 000

LD M0

HCNT CV235 K1000 //gdy M0 jest Wł., CV235 zaczyna zliczać impulsy wejścia X0, gdy liczba impulsów osiągnie 1000, C235 zostanie załączone

#### Network 001

LD C235

OUT Y000 //gdy liczba impulsów osiągnie 1000, Y0 będzie włączone

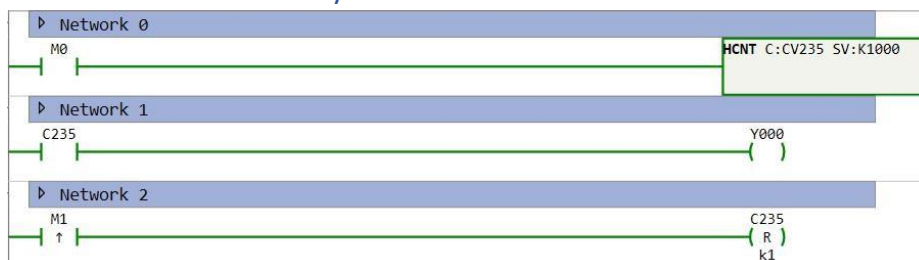
#### Network 002

LDP M1

RST C235 K1 //gdy M1 jest Wł., zresetuj C235 i CV235

POP

#### Schemat drabinkowy



#### 2. Zliczanie dwukierunkowe

#### Tabela instrukcji

#### Network 000

LD M0

HCNT CV246 K1000 //gdy M0 jest Wł., CV246 zaczyna zliczać impulsy dodatnie wejścia X0, gdy liczba impulsów osiągnie 1000, C246 zostanie załączone

#### Network 001

LD C246



OUTY000 // gdy liczba impulsów osiągnie 1000, Y0 będzie włączane

#### Network 002

LDP M1

RST C246 K1 // gdy M1 jest Wł., zresetuj C246 i CV246

POP

#### Schemat drabinkowy



### 3. Tryb zliczania fazowego AB

#### Tabela instrukcji

#### Network 001

LDP M8150

SET M8051 // ustawić tryb zliczania na tryb zliczania w logice x2 podczas pracy PLC

#### Network 001

LD M0

HCNT CV251 K1000 // gdy M0 jest włączone, CV251 zaczyna zliczać impulsy X0 (faza A) i

Wejście X1 (faza B), gdy liczba impulsów osiągnie 1000, C251 zostanie załączone

#### Network 002

LD C251

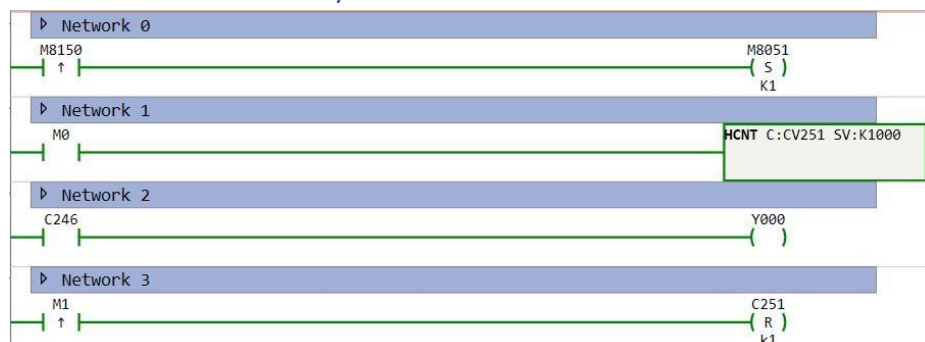
OUT Y000 // gdy liczba impulsów osiągnie 1000, Y0 będzie Wł.

#### Network 003

LDP M1

RST C251 K1 // gdy M1 jest Wł., zresetuj C251 i CV251

#### Schemat drabinkowy





## EHCNT

### Wprowadzenie do instrukcji

1. Ta instrukcja może być zastosowana tylko do PLC serii **FGm**.
2. Istnieją 4 rodzaje trybów liczenia dla tej instrukcji:
  - Zliczanie jednokierunkowe: użyj jednego portu wejściowego i możesz liczyć tylko w kierunku dodatnim.
  - Zliczanie dwukierunkowe: użyj dwóch portów wejściowych, jeden jest zliczany w kierunku do góry, a drugi w kierunku do dołu.
  - Zliczanie fazowe AB (obsługa enkoderów) - użyj dwóch portów wejściowych, jednego dla fazy A, a drugiego dla fazy B, można ustawić tryb zliczania w logice x2 i tryb zliczania w logice x4.
  - Zliczanie impulsów i kierunku: użyj dwóch portów wejściowych, jeden to wejście impulsowe, a drugi to wejście kierunku impulsów. Gdy wejście kierunku impulsów jest włączone, zliczanie odbywa się w kierunku dodatnim, w przeciwnym razie w kierunku ujemnym.
3. Istnieją 2 rodzaje trybów wyzwalania licznika startu i zerowania:
  - Samoczynne wyzwalanie: po wybraniu tego trybu licznik rozpocznie zliczanie po włączeniu i zeruje licznik oraz bit flagi zliczania, gdy liczba zliczeń osiągnie liczbę docelową.
  - Wyzwalanie zewnętrzne: ten tryb ma do wyboru dwa zestawy portów wejściowych (X25/X26 i X27/X32). Gdy wybrano X25/X26, użytkownik może użyć X25 do uruchomienia licznika i użyć X26 do zresetowania licznika; Gdy wybrano X27/X32, użytkownik może użyć X27 do uruchomienia licznika i użyć X32 do zresetowania licznika.
4. Ta instrukcja obsługuje większość 12 portów wejściowych (od X0 do X7 i od X10 do X13). Jak pokazano w tabeli poniżej, w 4 trybach zliczania zależności między portami wejściowymi a szybkimi licznikami są różne.
5. Zakres zliczania wynosi od -2 147 483 648 do 2 147 483 647, gdy zliczanie jest poza zakresem, nastąpi przepełnienie lub niedopełnienie.

	Rejestr szybkiego licznika	Port szybkiego wejścia impulsowego											
		X00	X01	X02	X03	X04	X05	X06	X07	X10	X11	X12	X13
Liczenie jednokierunkowe	CV235	U											
	CV236		U										
	CV237			U									
	CV238				U								
	CV239					U							
	CV240						U						
	CV241							U					



	CV242								U				
	CV243									U			
	CV244										U		
	CV245											U	
	CV246												U
Liczenie dwukierunkowe	CV235	U	D										
	CV237			U	D								
	CV239					U	D						
	CV241							U	D				
	CV243									U	D		
	CV245											U	D
Zliczanie fazowe AB	CV235	A	B										
	CV237			A	B								
	CV239					A	B						
	CV241							A	B				
	CV243									A	B		
	CV245											A	B
Impulsowanie i kierunek	CV235	PLS	DIR										
	CV237			PLS	DIR								
	CV239					PLS	DIR						
	CV241							PLS	DIR				
	CV243									PLS	DIR		
	CV245											PLS	DIR

Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
---------------------	------	---------	--------	------------



(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(CM)	Tryb zliczania, <b>0</b> dla jednokierunkowego, <b>1</b> dla dwukierunkowego, <b>2</b> dla logiki x2 zliczania fazowego AB, <b>3</b> dla logiki x4 zliczania fazowego AB, <b>4</b> dla impulsowania i kierunku.	K	0~4	16-bitowa liczba całkowita bez znaku
(TM)	Tryb wyzwalania licznika, <b>0</b> dla samoczynnego wyzwalania; <b>2</b> dla zewnętrznego wyzwalania zestawu <b>1</b> (X25/X26), <b>3</b> dla zewnętrznego wyzwalania zestawu <b>2</b> (X27/X32).	K	0~3	16-bitowa liczba całkowita bez znaku
(SV)	Docelowy numer odpowiadającego licznika.	K/H/D	-2147483648~2,147483647	32-bitowa liczba całkowita ze znakiem

#### Uwagi:

1. Podczas podłączania wejścia do portu, w celu poprawy przeciwdziałania zakłóceniom, zaleca się dodanie taśmy ekranujących na linii wejściowej i podłączenie linii ekranującej do uziemienia.
2. Gdy używany jest jeden szybki licznik, port wejściowy, który zajmuje, nie może być używany do innego licznika, nie można go również zastosować do powszechnego wejścia.
3. Reakcja na zliczenie licznika do liczby docelowej jest natychmiastowa, nie ma na nią wpływu okres skanowania.

## HSCS

#### Wprowadzenie do instrukcji

1. Ta instrukcja służy do wyzwalania zdarzenia przerwania, gdy zliczanie szybkiego licznika osiągnie liczbę docelową.
2. Sterownik PLC serii **FGm** nie obsługuje tej instrukcji.
3. Ta instrukcja jest używana razem z HCNT, działa tylko wtedy, gdy włączony jest HCNT na odpowiednim szybkim liczniku.
4. Przerwania tej instrukcji muszą być wyzwalane w takiej kolejności, jeśli jedno przerwanie zostanie pominięte, kolejne nie będą wyzwalane tylko wtedy, gdy użytkownik wyzeruje licznik.
5. Zdarzenie przerwania zostanie obsłużone po uruchomieniu, nie ma to związku okresem skanowania PLC.





### Ustawienia danych:

**High Count Compare**

Counter(CV32): CV251      Invoke Mode: Single

Cycle Count: D0

Table Count: 3      Counter Mode: Absolute

ID	Target Value	Interruption
0	K1000	SET M10
1	K2000	RST M10
2	K3000	t1

☒ Target Value  
☐ Constance ☒ Increment ☐ Decrement  
☒ K ☐ D 3000  
 Inc/Dec Step: 2

☒ Interruption  
 t1  
 Modify arguments

Operations  
 Added Number: 3  
 Add      Multiply Add  
 Modify      Remove  
 Move Up      Move Down

Yes(Y)      Cancel(C)      Help

#### 1. Licznik:

Można ustawić za pomocą licznika jednofazowego (CV235 do CV238) i licznika fazy AB (CV251 i CV252) (jak dotąd nie obsługuje licznika dwufazowego).

#### 2. Tryb wyzwalania:

Można ustawić w trybie pojedynczym lub trybie cyklicznym. W trybie pojedynczym, po wyzwoleniu ostatniego przerwania, licznik będzie kontynuował zliczanie, przerwy nie będą ponownie wyzwalane tylko wtedy, gdy użytkownik zresetuje licznik; W trybie cyklicznym po wyzwoleniu ostatniej przerwy licznik zeruje się automatycznie i rozpocznie odliczanie z dodaniem liczby cykli o 1, dodatkowo przerwy mogą być wyzwalane od pierwszej.

#### 3. Liczba cykli:

Rejestry D, które rejestrują, ile cykli wyzwalają przerwy.

#### 4. Tryb liczenia:

Można ustawić w trybie bezwzględny lub względny. W trybie bezwzględnym liczba docelowa oznacza rzeczywistą ilość impulsów wyjściowych; W trybie względnym liczba docelowa oznacza różnicę rzeczywistej ilości impulsów od ostatniego celu do aktualnej.

#### 5. Numer docelowy:

Można ustawić za pomocą danych typu K lub danych typu D, użytkownik może dodawać, przenosić, modyfikować i usuwać numery docelowe za pomocą operacji na interfejsie.

#### 6. Przerwa:

Można ustawić na 4 sposoby: cewka ustawiająca, cewka resetująca, podprogram i zero. Aby ustawić cewkę, użytkownik może wpisać „#(SET Yn)” lub „#(SET Mn)”, aby ustawić odpowiednie rejestry Y i M; Aby zresetować



cewkę, użytkownik może wpisać „#(RST Yn)” lub „#(RST Mn)”, aby zresetować odpowiednie Y i M rejestry; W przypadku podprogramu użytkownik może wypełnić nazwę odpowiedniego podprogramu, aby go wywołać; Dla zera nie ma potrzeby wypełniania, system nie zrobi nic, gdy zliczanie osiągnie cel.

7. Długość tabeli:

Liczba numerów docelowych, jej maksymalny limit to 100.

### Uwagi

1. W trybie bezwzględnym sekwencja liczb docelowych liczenia jednokierunkowego musi być przyrostowa, liczba docelowa liczenia dwukierunkowego musi być różna od ostatniej.
2. W trybie względnym liczby docelowe zliczania jednokierunkowego muszą być liczbami dodatnimi, liczba docelowa zliczania dwukierunkowego musi być różna od zera.
3. Pierwsza w sekwencji liczba docelowa nie może być równa 0.
4. Zaleca się ustawienie odstępu między dwoma sąsiednimi numerami docelowymi na więcej niż 50, w przeciwnym razie przerwanie może nie zostać uruchomione normalnie.
5. Cewkę ustawiania lub resetowania przerwania można ustawić za pomocą cewki modułu rozszerzającego.

### Przykład

#### Tabela instrukcji

#### Network 000

LD	M0
HCNT	CV251K999999 // włączyć licznik, gdy M0 jest włączone
HSCS	CV251 K999999K3 // włącz HSCS, gdy M0 jest włączone

#### Network

#### 001

LDP	M1
RST	C251 K1 // zresetować licznik, gdy M1 jest włączony, aby przerwanie mogło zostać wyzwolone od początku
POP	



## INSTRUKCJE NA CIĄGACH ZNAKÓW

### Uwagi

Instrukcje w tym rozdziale wykonują wzajemne konwersje między ciągiem znaków a liczbami. Liczby mogą być w formatach WORD, DOUBLE WORD i FLOAT; łańcuch znaków może być zakodowany tylko w formacie ASCII (nie obsługuje Unicode). Tabela poniżej pokazuje powszechnie używane kodowanie ASCII.

Kodowanie (szesnastkowe)	Znak	Kodowanie (szesnastkowe)	Znak	Kodowanie (szesnastkowe)	Znak
0x20	(spacja)	0x40	@	0x60	`
0x21	!	0x41	A	0x61	a
0x22	"	0x42	B	0x62	b
0x23	#	0x43	C	0x63	c
0x24	\$	0x44	D	0x64	d
0x25	%	0x45	E	0x65	e
0x26	&	0x46	F	0x66	f
0x27	'	0x47	G	0x67	g
0x28	(	0x48	H	0x68	h
0x29	)	0x49	I	0x69	i
0x2A	*	0x4A	J	0x6A	j
0x2B	+	0x4B	K	0x6B	k
0x2C	,	0x4C	L	0x6C	l
0x2D	-	0x4D	M	0x6D	m
0x2E	.	0x4E	N	0x6E	n
0x2F	/	0x4F	O	0x6F	o
0x30	0	0x50	P	0x70	p
0x31	1	0x51	Q	0x71	q
0x32	2	0x52	R	0x72	r
0x33	3	0x53	S	0x73	s
0x34	4	0x54	T	0x74	t
0x35	5	0x55	U	0x75	u
0x36	6	0x56	V	0x76	v
0x37	7	0x57	W	0x77	w
0x38	8	0x58	X	0x78	x
0x39	9	0x59	Y	0x79	y



0x3A	:	0x5A	Z	0x7A	z
0x3B	;	0x5B	[	0x7B	{
0x3C	<	0x5C	\	0x7C	
0x3D	=	0x5D	]	0x7D	}
0x3E	>	0x5E	^	0x7E	~
0x3F	?	0x5F	_	0x7F	DEL (usuń)

## I\_S

1. Instrukcja konwertuje dane typu WORD (IN) na ciąg 8-bajtowy (OUT) z maską formatu (FMT).
2. W masce formatu wpływ ma tylko tylko młodszy półbajt (nibble), bit starszy (c) półbajtu decyduje o tym, czy ciąg wyjściowy jest oddzielony kropką („.”), czy przecinkiem („,”), (low byte) - młodsze 3-bity (nnn) półbajtu decydują o długości liczby po kropce lub przecinku (maksymalnie 5). Szczegóły na rysunku poniżej
3. Ta instrukcja jest czasochłonna, lepiej jeśli wyzwalana jest przez sygnał zbocza.

IN	FMT	OUT(ASCII)		OUT+1(ASCII)		OUT+2(ASCII)		OUT+3(ASCII)	
		Low nibble	High nibble	Low nibble	High nibble	Low nibble	High nibble	Low nibble	High nibble
123	0x0004	(space)	(space)	0	,	0	1	2	3
-123	0x0004	(space)	-	0	,	0	1	2	3
123	0x000C	(space)	(space)	0	.	0	1	2	3
-123	0x000C	(space)	-	0	.	0	1	2	3

### Low byte of format mask (FMT)

				c	n	n	n
--	--	--	--	---	---	---	---

c: 1 for point, 0 for comma

nnn: length of number after point or comma, range is 0~5

c: 1 dla kropki, 0 dla przecinka.

(nnn: długość liczby po kropce-zakres:0-5)

## Ustawienia danych

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(IN)	Wejściowe dane WORD	K/H/D/CV/TV/AI/AO	-32768~32767	16-bitowa liczba całkowita ze znakiem
(FMT)	Maska decydująca o formacie danych wyjściowych	K/H/D/CV/TV/AI/AO	-	



(OUT)	Wyjście do przechowywania ciągu znaków	D	-	String
-------	--	---	---	--------

### Lista instrukcji

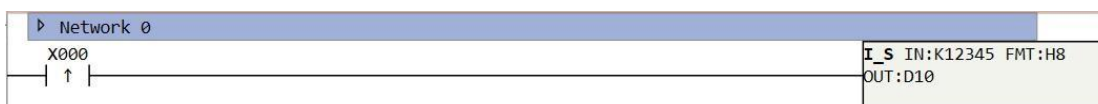
#### Network 000

LDP X000

I\_S K12345 H8D10 // K12345 jest konwertowany na „12345” i zapisywany w D10~D13 (użyj kropki, 0 cyfr po kropce)

POP

### Schemat drabinkowy



## DI\_S

1. Instrukcja konwertuje dane typu DWORD (IN) na ciąg 12-bajtowy (OUT) z maską formatu (FMT).
2. W masce formatu wpływ ma tylko młodszy półbajt (nibble), bit starszy (c) półbajtu decyduje o tym, czy ciąg wyjściowy jest oddzielony kropką („.”), czy przecinkiem („,”), (low byte) - młodsze 3-bity (nnn) półbajtu decydują o długości liczby po kropce lub przecinku (maksymalnie 5). Szczegóły na rysunku poniżej.
3. Ta instrukcja jest czasochłonna, lepiej jeśli wyzwalana jest przez sygnał zbocza.

IN	FMT	OUT(ASCII)		OUT+1(ASCII)		OUT+2(ASCII)		OUT+3(ASCII)		OUT+4(ASCII)		OUT+5(ASCII)	
		Low nibble	High nibble	Low nibble	High nibble	Low nibble	High nibble	Low nibble	High nibble	Low nibble	High nibble	Low nibble	High nibble
123456	0x0006	(space)	(space)	(space)	(space)	0	,	1	2	3	4	5	6
-123456	0x0006	(space)	(space)	(space)	-	0	,	1	2	3	4	5	6
123456	0x000E	(space)	(space)	(space)	(space)	0	.	1	2	3	4	5	6
-123456	0x000E	(space)	(space)	(space)	-	0	.	1	2	3	4	5	6

#### Low byte of format mask (FMT)

				c	n	n	n
--	--	--	--	---	---	---	---

c: 1 for point, 0 for comma

nnn: length of number after point or comma, range is 0~5

c: 1 dla kropki, 0 dla przecinka.

(nnn: długość liczby po kropce-zakres:0-5)

### Ustawienia danych

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna



(IN)	Wejściowe dane DWORD	K/H/D/CV /TV/Al/ AO	-2147483648~2147483647	32-bitowa liczba ze znakiem
(FMT)	Maska decydująca o formacie danych wyjściowych	K/H/D/CV /TV/Al/ AO	-	
(OUT)	Wyjście do przechowywania ciągu znaków	D	-	String

### Lista instrukcji

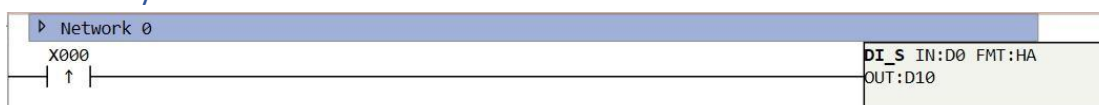
### Network 000

LDP X000

DI\_S D0 HA D10 //D0 jest konwertowane na łańcuch String i przechowywane w D10~D15 (użyj kropki, 2 cyfry po kropce)

POP

### Schemat drabinkowy



### R\_S

1. Instrukcja zaokrągla dane FLOAT (IN) i konwertuje go na ciąg o określonej długości (OUT) z maską formatu (FMT).
2. W masce formatu wpływ ma tylko młodszy bajt, starsze 4-bity (ssss) w bajcie decydują o długości łańcucha wyjściowego, 5-bit bajtu decyduje o tym, czy ciąg wyjściowy jest oddzielony kropką („.”), czy przecinkiem („,”), (low byte) - młodsze 3-bity (nnn) decydują o długości liczby po kropce lub przecinku (maksymalnie 7). Szczegóły na rysunku poniżej.
3. Ta instrukcja jest czasochłonna, lepiej jeśli wyzwalana jest przez sygnał zbocza.

IN	FMT	OUT(ASCII)		OUT+1(ASCII)		OUT+2(ASCII)	
		Low nibble	High nibble	Low nibble	High nibble	Low nibble	High nibble
1234.5	0x0061	1	2	3	4	,	5
-0.0004	0x0061	(space)	(space)	(space)	0	,	0
-3.67526	0x0069	(space)	(space)	-	3	.	7
1.95	0x0069	(space)	(space)	(space)	2	.	0

### Low byte of format mask (FMT)

s	s	s	s	c	n	n	n
---	---	---	---	---	---	---	---

ssss: length of output string, range is 3~15

c: 1 for point, 0 for comma

nnn: length of number after point or comma, range is 0~5



ssss: długość łańcucha wyjściowego, zakres: 3-15

c: 1 dla kropki, 0 dla przecinka.

(nnn: długość liczby po kropce-zakres:0-7)

### Ustawienia danych:

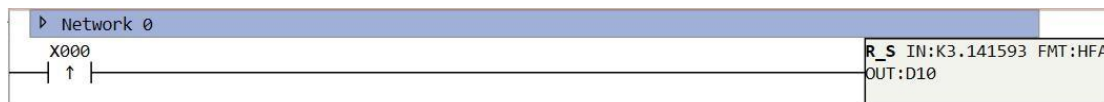
Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(IN)	Wejściowe dane FLOAT	K/D	$\pm 1.17549e-38F \sim \pm 3.40282e+38F$	FLOAT
(FMT)	Maska decydująca o formacie danych wyjściowych	K/H/D/CV/T V/AI/AO	-	
(OUT)	Wyjście do przechowywania ciągu znaków	D	-	String

### Network 000

LDP X000

R\_S K3.141593 HFA D10 //K3.141593 jest konwertowany na „3.14” i zapisywany w D10~D17 (łańcuch o długości 15, użyj kropki, 2 cyfry po kropce)

POP



## S\_I

1. Ta instrukcja konwertuje łańcuch na 16-bitową liczbę całkowitą ze znakiem (WORD).
2. Ta instrukcja konwertuje z (IDX)-tego znaku łańcucha, znakiem początkowym musi być cyfra („0” do „9”), „+”, „-” lub jakaś ilość spacji; Konwersja kończy się na znaku nie będącym cyfrą („0” do „9”) lub jednym z następujących znaków: „+”, „-”, „ ” i „.”. Rysunek poniżej pokazuje przykłady konwersji i nieprawidłowych danych wejściowych.
3. Jeśli liczba po konwersji jest za duża lub za mała w porównaniu z 16-bitową liczbą całkowitą ze znakiem, nastąpi przepełnienie lub niedopełnienie, **M8169** włączy się.
4. Jeśli ciąg wejściowy jest nieprawidłowy do przekonwertowania na liczbę całkowitą, nie będzie wyjścia, a **M8169** również się włączy. Rysunek poniżej pokazuje kilka przykładów konwersji i nieprawidłowych danych wejściowych.

**Prawidłowe dane wejściowe:**

**Nieprawidłowe dane wejściowe:**



## Valid input

Input string	Output integer
"123"	123
"-00456"	-456
"123.45"	123
" +2345"	2345
"000000123ABCD"	123

## Invalid input

Input string	Reason
"A123"	Start with 'A'
" "	Start with ' '
"++123"	End at '+'
"+-123"	End at '-'
" + 123"	End at ' '
"50000"	Overflow
"-50000"	Underflow

## Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(IN)	Wejście przechowujące łańcuch danych	D	-	String
(IDX)	Numer indeksu, aby rozpocząć konwersję	K/H/D/CV/TV/AI/AO	1~65535	16-bitowa liczba całkowita ze znakiem
(OUT)	Wyjście do danych WORD	D/CV/TV/AO	-32768~32767	16-bitowa liczba całkowita ze znakiem

## Lista instrukcji:

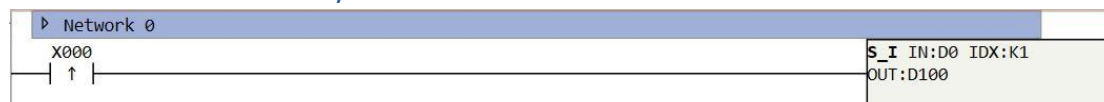
## Network 000

LDP X000

S\_I D0 K1 D100 //przekonwertuj ciąg w D0, zaczynając od indeksu 1 i zapisz wynik w postaci liczby całkowitej w D100

POP

## Schemat drabinkowy:



## S\_DI

1. Ta instrukcja konwertuje łańcuch na 32-bitową liczbę całkowitą ze znakiem (WORD).





2. Ta instrukcja konwertuje z (IDX)-tego znaku łańcucha, znakiem początkowym musi być cyfra („0” do „9”), „+”, „-” lub jakaś ilość spacji; Konwersja kończy się na znaku nie będącym cyfrą („0” do „9”) lub jednym z następujących znaków: „+”, „-”, „,” i „.”. Rysunek poniżej pokazuje przykłady konwersji i nieprawidłowych danych wejściowych.
3. Jeśli liczba po konwersji jest za duża lub za mała w porównaniu z 32-bitową liczbą całkowitą ze znakiem, nastąpi przepełnienie lub niedopełnienie, **M8169** włączy się.
4. Jeśli ciąg wejściowy jest nieprawidłowy do przekonwertowania na liczbę całkowitą, nie będzie wyjścia, a **M8169** również się włączy. Rysunek poniżej pokazuje kilka przykładów konwersji i nieprawidłowych danych wejściowych.

**Prawidłowe dane wejściowe:**

Valid input

Input string	Output integer
"123"	123
"-00654"	-654
"543.21"	543
" +54321"	54321
"000054321ABCD"	54321

**Nieprawidłowe dane wejściowe:**

Invalid input

Input string	Reason
"A98765"	Start with 'A'
" "	Start with ' '
"++98765"	End at '+'
"-98765"	End at '-'
" + 98765"	End at ' '
"3000000000"	Overflow
"-3000000000"	Underflow

**Ustawienia danych:**

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(IN)	Wejście przechowujące łańcuch danych	D	-	String
(IDX)	Numer indeksu, aby rozpocząć konwersję	K/H/D/CV/TV/ AI/AO	1~65535	16-bitowa liczba całkowita ze znakiem
(OUT)	Wyjście do danych WORD	D	-2147483648~2147483647	32-bitowa liczba całkowita ze znakiem

**Przykład:****Lista instrukcji:**

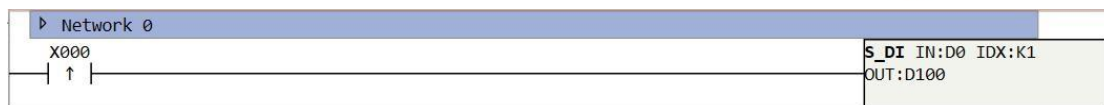
Network 000

LDP X000

S\_DI D0 K1 D100 // przekonwertuj ciąg w D0, zaczynając od indeksu 1 i zapisz wynik w postaci liczby całkowitej w D100D101

POP

**Schemat drabinkowy:**



## S\_R

1. Ta instrukcja konwertuje łańcuch na dane FLOAT (DWORD).
2. Ta instrukcja konwertuje z (IDX)-tego znaku łańcucha, znakiem początkowym musi być cyfra („0” do „9”), „+”, „-” lub jakaś ilość spacji; Konwersja kończy się na znaku nie będącym cyfrą („0” do „9”) lub jednym z następujących znaków: „+”, „-”, „ ” i „.”. Rysunek poniżej pokazuje przykłady konwersji i nieprawidłowych danych wejściowych.
3. Konwersja na FLOAT nie spowoduje przepełnienia ani niedomiaru, ale jeśli znaczące cyfry liczby konwersji są większe niż 7, wynik zostanie zaokrąglony.
4. Jeśli ciąg wejściowy jest nieprawidłowy do przekonwertowania na dane zmiennoprzecinkowe, nie będzie wyjścia, a **M8169** się włączy. Rysunek poniżej pokazuje kilka przykładów konwersji i nieprawidłowych danych wejściowych.

### Prawidłowe dane wejściowe:

#### Valid input

Input string	Output float
"123"	123.0
"-00456"	-456.0
"123.45"	123.45
" +2345"	2345.0
"00.000000123"	0.000000123

### Nieprawidłowe dane wejściowe:

#### Invalid input

Input string	Reason
"A123"	Start with 'A'
" "	Start with ' '
"++123"	End at '+'
"+-123"	End at '-'
" + 123"	End at ' '

### Ustawienia danych:

Wejścia/ wyjścia	Opis	Operand	Zakres	Typ danych
(EN)	Wejście załączające/ wyłączające instrukcję		0/1	Wartość logiczna
(IN)	Wejście przechowujące łańcuch danych	D	-	String
(IDX)	Numer indeksu, aby rozpocząć konwersję	K/H/D/CV/TV/AI/AO	1~65535	16-bitowa liczba całkowita ze znakiem
(OUT)	Wyjście do danych WORD	D	$\pm 1.17549\text{e-}38\text{F} \sim \pm 3.40282\text{e+}38\text{F}$	FLOAT



Przykład:

Lista instrukcji:

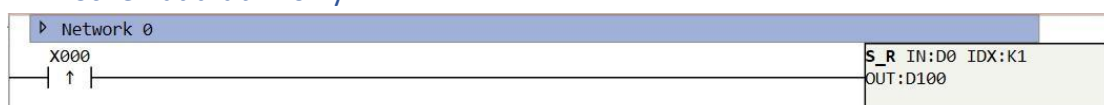
Network 000

LDP X000

S\_R D0 K1 D100 // przekonwertuj ciąg w D0, zaczynając od indeksu 1 i zapisz wynik w postaci FLOAT w D100D101

POP

Schemat drabinkowy:

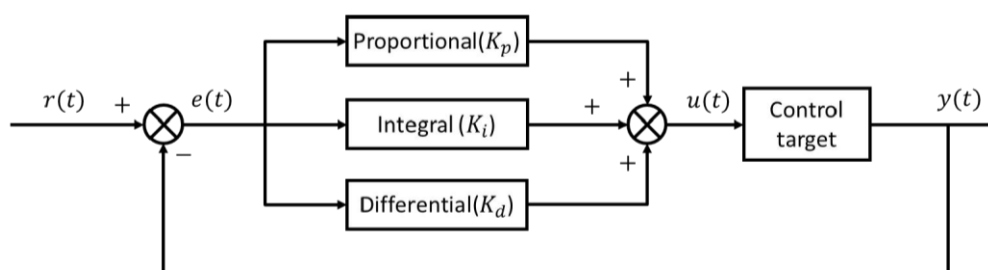




## INSTRUKCJE STEROWANIA PID

### EPID

Regulacja PID jest rodzajem regulacji w pętli zamkniętej. Zgodnie z błędem ( $e(t)$ ) pomiędzy wartością mierzoną ( $y(t)$ ) a wartością docelową ( $r(t)$ ), stosuje się proporcjonalność ( $K_p$ ), całkę ( $K_i$ ), współczynnika różniczkowego ( $K_d$ ). Szczegóły patrz rysunek poniżej:



Relacje matematyczne  $r(t)$ ,  $e(t)$ ,  $u(t)$ , i  $y(t)$ :

$$e(t) = r(t) - y(t)$$

$$u(t) = K_p \left[ e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right]$$

W powyższym wzorze ( $K_p$ ) jest współczynnikiem proporcjonalności,  $T_i$  jest czasem całkowanym.  $T_d$  to czas różniczkowy. Podczas gdy w tym przypadku istnieje minimalny odstęp czasowy dla próbkowania ( $T$ ), wzór można zapisać jako:

$$u_k = K_p \left[ e_k + \frac{T}{T_i} \sum_{j=0}^k e_j + T_d \frac{e_k - e_{k-1}}{T} \right]$$

Czas całkowania  $T_i$  i czas różniczkowy  $T_d$  można również przekształcić na współczynnik całkowy  $K_i$  i współczynnik różniczkowy  $K_d$ :

$$u_k = K_p e_k + K_i \sum_{j=0}^k e_j + K_d (e_k - e_{k-1}) \quad \left( K_i = \frac{K_p T}{T_i}, K_d = \frac{K_p T_d}{T} \right)$$

#### Wprowadzenie do instrukcji:

1. Ta instrukcja wykonuje regulację PID na wyjściu według wartości zmierzonej i wartości docelowej. Obsługuje wyjście wartości przełączania i analogowe wyjście ilościowe i może kontrolować maksymalnie 2 pętle.



2. Użytkownik może ręcznie dostosować parametry PID, a także użyć funkcji samodostrajania, aby ustawić początkowe parametry PID. Samostrojenie wykorzystuje metodę oscylacji krytycznych i zaleca się dostosowanie wyjścia urządzenia (wartości mierzonej) blisko wartości docelowej, aby obniżyć czas samostrojenia.

3. Ta instrukcja ma trzy tryby do wyboru i użytkownik może je przełączać podczas pracy.

Przesunięcie adresu	Parametr		Typ danych	O-odczyt, Z-zapis.	Opis
0	Sampling cycle	Cykl próbkowania	DWORD	O/Z	Przedział czasu do próbkowania błędów, jednostką jest ms (milisekunda). W przypadku przełączania trybu wartości domyślnie jest to 1000 ms; Dla trybu ilości analogowych domyślnie jest to 0 i musi być przypisane przez użytkownika.
2	Proportional coefficient	Współczynnik proporcjonalności	FLOAT	O/Z	Współczynnik proporcji $K_p$ regulacji PID, nie działa, gdy jest ustawiony na 0.
4	Integral time	Czas integralny	FLOAT	O/Z	Czas całkowania $T_i$ regulatora PID, którego jednostką jest ms, nie działa, gdy jest ustawiony na 0.
6	Integral coefficient	Współczynnik całkowany	FLOAT	O	$K_i$ Współczynnik całkowany regulacji PID. $K_i = K_p T_i / T$
8	Differential time	Czas różniczkowy	FLOAT	O/Z	$T_d$ Czas całkowania regulacji PID, którego jednostką jest ms, nie działa, gdy jest ustawiony na 0.
10	Differential coefficient	Współczynnik różniczkowy	FLOAT	O	$K_d$ (Współczynnik różnicowy) regulacji PID. $K_d = K_p T_d / T$
12	Dead band range	Zakres martwego pasma	FLOAT	O/Z	Gdy błąd między wartością docelową a wartością mierzoną jest mniejszy niż zakres strefy martwej, regulacja PID jest nieskuteczna.
14	Output high limit	Górny limit wyjścia	FLOAT	O/Z	Górny limit wyjścia, dla trybu wartości przełączania, jest to limit współczynnika wypełnienia; Dla trybu ilości analogowych jest to limit wyjścia analogowego.
16	Output low limit	Dolny limit wyjścia	FLOAT	O/Z	Dolny limit wyjścia, dla trybu wartości przełączania, jest to limit współczynnika wypełnienia; Dla trybu ilości analogowych jest to limit wyjścia analogowego.
18	PID configuration	Konfiguracja PID	HEX	O/Z	Konfiguracja do przetwarzania regulacji PID, szczegóły znajdują się w poniższym wstępie
20	PID calculated value	Wartość obliczona PID	DWORD	O	Tylko w trybie wartości przełączania zapisz czas zamknięcia podczas cyklu próbkowania, którego jednostką jest ms. Jest skuteczny w trybie ilości analogowych.
22	PID valid range	Prawidłowy zakres PID	FLOAT	O/Z	Gdy błąd między wartością docelową a wartością zmierzona przekroczy prawidłowy zakres PID, PID wyświetli górny/dolny limit zgodnie ze znakiem błędów i kierunkiem działania..
24	Error code	Kod błędów	HEX	O	Kod błędów legalności zapisu w ustawieniach parametrów, gdy instrukcja jest włączona.



25	PID mode	Tryb PID	WORD	O/Z	Rejestr danych rejestruje i przełącza tryb przetwarzania PID, wartość 0 dla trybu regulacji rutynowej, wartość 1 dla trybu regulacji ręcznej, wartość 2 dla trybu regulacji rozmytej z samoregulacją, w innym przypadku wartość dla wstrzymania regulacji (wyjście pozostaje bez zmian).
26	Fuzzy self-adjusting status	Status rozmytej samoregulacji	WORD	O/Z	Rejestr danych rejestruje i przełącza stan regulacji rozmytej samodostrajającej, wartość 1 dla stanu samodostrajającego, w innym przypadku wartość dla stanu nie dopasowującego się.
27-33	Reserved	Zarezerwowany	-	-	Obszar zarezerwowany dla systemu
34	Self-tuning status	Stan samostrojzenia	WORD	O/Z	Rejestr danych rejestruje i przełącza stan samostrojzenia, wartość 0 dla samostrojzenia zakończonego, wartość 1 dla procesu samostrojzenia, wartość 2 dla samostrojzenia zakończonego niepowodzeniem. Użytkownik może wpisać 1 do tego rejestru, aby ponownie uruchomić samostrojzenie
36	Self-tuning timeout	Limit czasu samodostrajania	DWORD	O/Z	Przekroczenie limitu czasu dla samostrojzenia, jeśli czas samostrojzenia przekroczy limit czasu, samostrojzenie zakończy się niepowodzeniem.
40-49	Reserved	Zarezerwowany	-	-	Obszar zarezerwowany dla systemu

W przewodniku po instrukcji EPID użytkownik może ustawić główny rejestr D, aby mapować parametry do rejestrów D, aby parametry mogły być monitorowane i dostosowywane podczas pracy, zasady mapowania parametrów w tabeli powyżej. Parametry zostaną zainicjowane przez ustawienie w przewodniku, użytkownik może również zainicjować te parametry za pomocą instrukcji przesyłania. Poniżej przedstawiono szczegóły parametrów:

#### 1. Sampling cycle – Cykl próbkowania

Cykl próbkowania ( $T$ ) to przedział czasu, w którym system próbkuje wartość mierzoną i wartość docelową, jego jednostką jest ms (milisekunda). Dodatkowo jest to również cykl przełączania wyjścia.

Cykl próbkowania musi być dłuższy niż okres skanowania PLC.

#### 2. Proportional coefficient - Współczynnik proporcjonalności

Współczynnik proporcjonalności ( $K_p$ ) skaluje błąd sterowania urządzeniem, gdy jest duży, reakcja sterowania będzie szybka, ale wystąpią przeregulowania i fluktuacje na wyjściu urządzenia; Gdy jest mały, wyjście urządzenia będzie stabilne, ale odpowiedź będzie powolna.

#### 3. Integral time - Czas całkowania

Czas całkowania ( $T_i$ ) dzieli skumulowany błąd od startu do prądu, gdy jest większy, całka przynosi większy efekt; Gdy jest mniejsza, całka przynosi mniejszy efekt.

#### 4. Integral coefficient - Współczynnik całkowy

Współczynnik całkowy ( $K_p$ ) skaluje nagromadzenie błędów w celu wyeliminowania różnicy między wartością zmierzoną a wartością docelową wynosi  $K_p T / T_i$ .

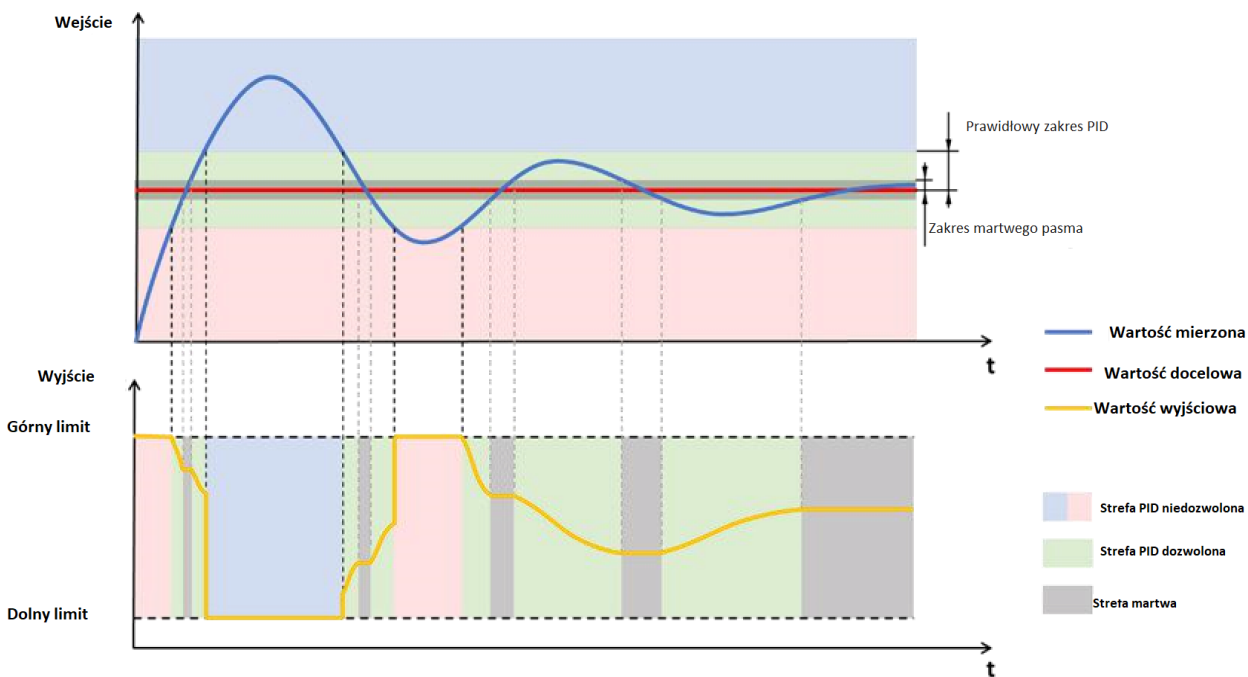
#### 5. Differential time - Czas różniczkowy



Czas różniczkowy ( $T_d$ ). mnoży różnicę błędów od bieżącego do ostatniego, kiedy to jest większy, różnica ma silniejszy efekt; Gdy różnica działa słabiej.

#### 6. Differential coefficient – Współczynnik różniczkowy

Współczynnik całkowy ( $K_d$ ) skaluje gradient (szybkość zmian) błędu podążającego za zmianą wartości mierzonej, jest równy  $K_p T_d / T$ .



#### 7. Dead band range - Zakres martwego pasma

Patrz rysunek powyżej; gdy błąd między wartością docelową a wartością mierzoną jest mniejszy niż zakres strefy nieczułości, regulacja PID jest nieefektywna, wartość wyjściowa pozostanie niezmienną, błąd jest poza strefą nieczułości. Funkcją strefy martwej jest uniknięcie niepotrzebnych obliczeń w stanie stabilnym (błąd jest mały i stabilny).

#### 8. Output high limit - Górny limit wyjścia

Patrz rysunek powyżej; ten parametr ogranicza najwyższą wartość wyjścia, musi być większy niż dolny limit wyjścia. W trybie wartości przełączania limit wyjściowy nie może być większy niż cykl próbkowania ( $T$ ).

#### 9. Output low limit - Dolny limit wyjścia

Patrz rysunek powyżej; ten parametr ogranicza najniższą wartość wyjściową, musi być mniejsza niż górna granica wyjściowa. W trybie wartości przełączania dolna granica wyjścia musi być nieujemna. Właściwy górny i dolny limit wyjściowy może skrócić czas samostrojenia. Jeśli górny limit jest zbyt wysoki, a dolny limit zbyt niski, samodostrajanie spowoduje większe i dłuższe wahania mocy wyjściowej, co będzie kosztować więcej czasu; Jeżeli górny limit jest zbyt niski, a dolny limit zbyt wysoki, mierzona wartość może nie osiągnąć wartości docelowej i samodostrajanie się nie powiedzie.

#### 10. PID configuration - Konfiguracja PID





Bit	15:4	3	2:1	0
Value	Reserved	SD	TYPE	Dir

Ten rejestr danych rejestruje i steruje następującą konfiguracją regulacji PID:

- **Dir:** kierunek działania urządzenia docelowego powinien być ustawiony według rzeczywistego przypadku. Gdy ten bit jest włączony, kierunek działania jest do przodu, błąd jest równy wartości mierzonej minus wartość docelowa, wyjście PID powinno rosnąć/spadać, gdy wartość mierzona wzrasta/spada; Gdy ten bit jest wyłączony, kierunek działania jest wsteczny, błąd jest równy wartości docelowej minus wartość mierzona, wyjście PID powinno maleć/rosnąć wraz ze wzrostem/spadkiem wartości mierzonej.
- **TYPE:** typ sterowania PID, wartość 00 dla PID, wartość 01 dla PI, wartość 10 dla P. PID ma zastosowanie do rzeczywistych przypadków z dużym opóźnieniem, takich jak regulacja temperatury; PI ma zastosowanie w rzeczywistym przypadku z szybką reakcją, taką jak kontrola ciśnienia i kontrola przepływu.
- **SD:** bit ten włącza lub wyłącza automatyczne wykrywanie stanu stabilnego, wartość 0 oznacza włączenie, wartość 1 wyłączenie, domyślnie jest to 0. Automatyczne wykrywanie stanu stabilnego jest używane do rozmytej samoregulacji, gdy jest włączone, system automatycznie wejdzie w samoregulację, gdy błąd będzie stabilny; Gdy jest wyłączona, użytkownik musi ręcznie ustawić stan samoregulacji.

#### 11. PID calculated value - Wartość obliczona PID

Ta wartość jest tylko do odczytu i obowiązuje tylko w trybie przełączania wartości. Rejestruje obliczoną moc wyjściową ( $u(t)$ ) regulacji PID, jest to również czas zamknięcia wartości przełączania w cyklu próbkowania.

#### 12. PID valid range - Prawidłowy zakres PID

Patrz rysunek powyżej; gdy błąd między wartością docelową a wartością zmierzona mieści się w zakresie strefy nieczułości, regulacja PID będzie ważna; W przeciwnym razie regulacja PID będzie nieważna, w tym przypadku, jeśli kierunek działania jest do przodu, wyjście przeskoczy do górnego/dolnego limitu, gdy błąd wynosi plus/minus; Jeśli kierunek działania jest wsteczny, wyjście przeskoczy do dolnej/górnej granicy, gdy błąd wynosi plus/minus.

#### 13. Error code- Kod błędu

Bit	15:3	2	1	0
Value	Reserved	SC	HL	ST

Gdy parametry są ustawione nieprawidłowo, PID nie będzie obliczać, a wyjście będzie takie samo, ten rejestr danych rejestruje błąd ustawienia parametru:

- **SC:** gdy cykl próbkowania (T) jest ustawiony jako niedozwolony/dozwolony, ten bit włącza się/wyłącza.
- **HL:** gdy górny limit wyjścia jest ustawiony jako niedozwolony/dozwolony, bit ten włącza się/wyłącza.
- **ST:** gdy prawidłowy zakres PID jest ustawiony jako niedozwolony/dozwolony, bit ten włącza się/wyłącza.

#### 14. PID mode - Tryb PID

Użytkownik może przełączać tryby PID, ustawiając różne wartości w tym rejestrze danych: wartość 0 dla trybu regulacji rutynowej, wartość 1 dla trybu regulacji ręcznej, wartość 2 dla trybu samoregulacji rozmytej, w innym przypadku wartość dla zatrzymania regulacji.

- **Tryb sterowania rutynowego - Routine control mode:** w tym trybie regulacja PID będzie działać z parametrami PID ustawionymi przez użytkownika, użytkownik może również modyfikować status samostrojenia, aby włączyć samodostrojenie.





- Tryb sterowania ręcznego - Manual control mode: w tym trybie regulacja PID będzie działać jako wyjście ustawione przez użytkowników. Gdy ten tryb zostanie przełączony na tryb sterowania rutynowego lub tryb samoregulacji rozmytej, wyjście będzie płynnie przechodzić.
- Tryb samoregulacji rozmytej - Fuzzy self-adjusting mode : w tym trybie regulacja PID wykryje, czy błąd jest stabilny. Jeśli błąd jest stabilny, regulator PID przejdzie w tryb samoregulacji i automatycznie dostosuje parametry PID.
- Zatrzymanie regulacji - Stopping control : wyjście pozostanie takie samo, gdy regulacja zostanie zatrzymana.

#### 15. Fuzzy self-adjusting status – Status rozmytej samoregulacji

Ten rejestr danych rejestruje, czy sterowanie wchodzi w stan samoregulacji w trybie rozmytej samoregulacji, wartość 1 dla stanu samoregulacji, w przeciwnym razie wartość nie.

Sterowanie automatycznie wejdzie w tryb samoregulacji, gdy błąd się ustabilizuje, i automatycznie dostosuje parametry PID ( $K_p$ ,  $K_i$ ,  $K_d$ ), użytkownik może również ustawić wartość 1, aby wymusić wejście w tryb samoregulacji, ale ta operacja będzie kosztować więcej czasu, aby ustabilizować błąd, a nawet zakończy się niepowodzeniem, jeśli wejdzie w ten tryb, kiedy błąd nie jest stabilny.

#### 16. Self-tuning status - Stan samostrojzenia

Ten rejestr danych rejestruje i kontroluje stan samostrojzenia, wartość 0 oznacza zakończenie samostrojzenia, wartość 1 oznacza przetwarzanie samostrojzenia, wartość 2 oznacza niepowodzenie samostrojzenia, w innym przypadku wartość oznacza wstrzymanie regulacji (wyjście pozostaje takie samo).

Samodostrajanie służy do ustawienia początkowych parametrów PID, gdy regulacja PID zostaje uruchomiona. Gdy samostrojzenie jest włączone, wyjście będzie przeskakiwać naprzemiennie między górnym i dolnym limitem, aby wprowadzić fluktuację mierzonej wartości w celu osiągnięcia wartości docelowej.

Zaleca się wyregulować wyjście tak, aby wartość mierzona była zbliżona do wartości docelowej w trybie regulacji ręcznej, tak aby samodostrajanie było szybkie, a początkowy parametr PID ustawiony racjonalnie.

#### 17. Self-tuning timeout - Limit czasu samostrojzenia

Ten rejestr danych ustawia limit czasu samostrojzenia, limit czasu samostrojzenia wynosi nie mniej niż 60 minut. Jeśli czas samostrojzenia przekroczy limit czasu, statusem samostrojzenia zostanie niepowodzenie samostrojzenia.



## DODATEK TABELA REJESTRÓW ORAZ MARKERÓW

### Rejestry funkcji specjalnych systemu

Systemowe rejestry funkcji specjalnych to rejestry D (rejestry danych) i rejestry M (rejestry bitowe), które mają specjalne funkcje. Szczegóły znajdują się w poniższej tabeli:

Rejestr	Opis	O- odczyt, Z-zapis.
<b>D-rejestry (rejestry danych)</b>		
<b>D8040~D8047</b>	Rejestry danych monitorują ostatnie 8 aktywnych bitów stanu.	O
<b>D8048</b>	Bieżący tryb <b>IST</b> (0: brak, 1: tryb ręczny, 2: tryb powrotu do zera, 3: tryb pojedynczego kroku, 4: tryb jednocyklowy, 5: tryb automatyczny).	O
<b>D8066</b>	Numer stacji COM0, zakres od 1 do 127.	O
<b>D8067</b>	Numer stacji COM1, zakres od 1 do 127.	O
<b>D8068</b>	Tryb szybkości transmisji COM0, zakres od 0 do 7, co odpowiada 1200 ~ 115200 w konfiguracji.	O
<b>D8069</b>	Tryb szybkości transmisji COM0, zakres od 0 do 7, co odpowiada 1200 ~ 115200 w konfiguracji.	O
<b>D8070</b>	Bit danych COM0, w stałej wartości 0, odpowiadający 8 bitom danych.	O
<b>D8071</b>	Bit danych COM1, w stałej wartości 0, odpowiadający 8 bitom danych.	O
<b>D8072</b>	Bit stopu komunikacji portu COM, zakres 0-1 odpowiadający 1 lub 2 bitom.	O
<b>D8073</b>	Bit stopu komunikacji portu COM1, zakres 0-1 odpowiadający 1 lub 2 bitom.	O
<b>D8074</b>	Sprawdź tryb bitowy COM0, zakres od 0 do 2, odpowiadający BRAK, PARZYSTY i NIEPARZYSTY.	O
<b>D8075</b>	Sprawdź tryb bitowy COM0, zakres od 0 do 2, odpowiadający BRAK, PARZYSTY i NIEPARZYSTY.	O
<b>D8076~D8079</b>	Rezerwacja systemu.	O/Z
<b>D8080~D8085</b>	Nieokreślony.	
<b>D8086</b>	Numer stacji COM2, zakres od 1 do 127.	O
<b>D8087</b>	Tryb szybkości transmisji COM2, zakres od 0 do 7, co odpowiada 1200 ~ 115200 w konfiguracji.	O
<b>D8088</b>	Bit danych COM2, być w stałej wartości 0, odpowiadającej 8 bitom danych.	O
<b>D8089</b>	Bit stopu COM2, zakres od 0 do 1, odpowiadający 1 lub 2 bitom stopu.	O
<b>D8090</b>	Sprawdź tryb bitowy COM2, zakres od 0 do 2, odpowiadający BRAK, PARZYSTY i NIEPARZYSTY.	O
<b>D8092</b>	Y000 czas przyspieszania/zwalniania portu impulsowego wielostopniowego wyjścia impulsowego.	O/Z
<b>D8093</b>	Y001 czas przyspieszania/zwalniania portu impulsowego wielostopniowego wyjścia impulsowego.	O/Z
<b>D8094</b>	Y002 czas przyspieszania/zwalniania portu impulsowego wielostopniowego wyjścia impulsowego.	O/Z



<b>D8095</b>	Y003 czas przyspieszania/zwalniania portu impulsowego wielostopniowego wyjścia impulsowego.	O/Z
<b>D8108</b>	Numer segmentu błędu portu impulsowego Y000 (dla wielostopniowego wyjścia impulsowego)	O
<b>D8109</b>	Numer segmentu błędu portu impulsowego Y001 (dla wielostopniowego wyjścia impulsowego).	O
<b>D8110</b>	Numer segmentu błędu portu impulsowego Y002 (dla wielostopniowego wyjścia impulsowego).	O
<b>D8111</b>	Numer segmentu błędu portu impulsowego Y003 (dla wielostopniowego wyjścia impulsowego).	O
<b>D8124</b>	Numer segmentu prądu portu impulsowego Y000 (dla wielostopniowego wyjścia impulsowego).	O
<b>D8125</b>	Numer segmentu prądu portu impulsowego Y001 (dla wielostopniowego wyjścia impulsowego).	O
<b>D8126</b>	Numer segmentu prądu portu impulsowego Y002 (dla wielostopniowego wyjścia impulsowego).	O
<b>D8127</b>	Numer segmentu prądu portu impulsowego Y003 (dla wielostopniowego wyjścia impulsowego).	O
<b>D8140(D8141)</b>	Licznik impulsów: Sumaryczna liczba wysłanych impulsów portu impulsowego Y000. Aktualna pozycja silnika przy pozycjonowaniu portu Y000.	O/Z
<b>D8142(D8143)</b>	Licznik impulsów: Sumaryczna liczba wysłanych impulsów portu impulsowego Y001. Aktualna pozycja silnika przy pozycjonowaniu portu Y001.	O/Z
<b>D8144(D8145)</b>	Licznik impulsów: Sumaryczna liczba wysłanych impulsów portu impulsowego Y002. Aktualna pozycja silnika przy pozycjonowaniu portu Y002.	O/Z
<b>D8146(D8147)</b>	Licznik impulsów: Sumaryczna liczba wysłanych impulsów portu impulsowego Y003. Aktualna pozycja silnika przy pozycjonowaniu portu Y003.	O/Z
<b>D8173</b>	Docelowa liczba przerw timer 0, zakres od 1 do 32767ms.	O/Z
<b>D8174</b>	Docelowa liczba przerw timer 0, zakres od 1 do 32767ms.	O/Z
<b>D8175</b>	Bieżący okres skanowania (jednostka to 0,1 ms).	O
<b>D8176</b>	Kod błędu 0x00: brak błędu 0x09: błąd zegara czasu rzeczywistego 0x0A: przekroczenie limitu czasu timera watchdoga 0x0C: limit czasu komunikacji portu szeregowego 0x0D: Upłynął limit czasu komunikacji portu USB 0x0E: wyjątek danych regionu XML, 0x0F: wyjątek danych regionu konfiguracji 0x10: wyjątek danych regionu HashCode 0x11: Wyjątek danych tabeli Modbus 0x12: Wyjątek danych tabeli PLSTABLE 0x13 Wyjątek danych regionu PLSBLOCK 0x17: liczba segmentów wielostopniowego wyjścia impulsowego poza zakresem 0x18: częstotliwość pierwszego segmentu impulsu wielostopniowego niższa niż domyślna częstotliwość początkowa 0x19: częstotliwość niższa niż dolna granica 0x1A: docelowa liczba impulsów jednostopniowych niższa od dolnego limitu. 0x1B: czas przyspieszania/zwalniania impulsu jednostopniowego poniżej dolnego limitu	O/Z



	<p>0x1C: parametry nie spełniają warunku ważności przyspieszania/zwalniania</p> <p>0x1D: ostatni segment impulsu wielostopniowego nie spełnia warunku prawnego przyspieszania/zwalniania</p> <p>0x1E: Upłynął limit czasu ładowania oprogramowania sterującego FPGA</p> <p>0x1F: Błąd sprawdzania identyfikatora EPCS4</p> <p>0x20: EPCS4 usuwa błąd danych</p> <p>0x21: Upłynął limit czasu aktualizacji oprogramowania układowego EPCS4</p> <p>0x22: Błąd pakietu danych aktualizacji EPCS4</p> <p>0x23: Błąd sprawdzania odpowiedzi EPCS4</p> <p>0x24: błąd parametrów interpolacji</p> <p>0x25: błąd perspektywny interpolacji</p> <p>0x26: błąd prędkości interpolacji (być równy 0 lub za duży)</p> <p>0x27: błąd współrzędnych punktu końcowego (ruch nie może wynosić 0)</p> <p>0x28: błąd promienia interpolacji łuku</p>	
	<p>0x29: zaraz przekroczysz granicę</p> <p>0x2A: parametry interpolacji nie spełniają warunku prawnego przyspieszania/zwalniania</p> <p>0x2B: błąd akordu za duży</p> <p>0x2C: przekroczenie maksymalnej częstotliwości impulsów</p> <p>0x2D: Błąd danych ECAM</p> <p>0x2E: Błąd parametrów ECAM</p> <p>0x34: niska wersja modułu rozszerzeń, nowa funkcja ma zastosowania</p> <p>0x35: niska wersja PLC, nowa funkcja nie ma zastosowania</p> <p>0x36: błąd komunikacji modułu rozszerzeń</p> <p>0x37: przekroczenie limitu czasu retransmisji modułu rozszerzeń</p> <p>0x38: błąd zapytania o typ modułu rozszerzeń</p> <p>0x39: błąd przekazywania parametrów modułu rozszerzeń</p> <p>0x3A: Błąd konfiguracji modułu portu Ethernet</p> <p>0x3B: Import wykresu CAD do błędu uzgadniania FLASH</p> <p>0x3F: ustawienie parametru HCNT nieracjonalne</p> <p>0x40: Błąd odczytu/zapisu CADflash</p> <p>0x41: Błąd pakietu danych pobierania CADflash</p> <p>0x42: błąd odczytu danych wykresu</p> <p>0x43: błąd interpolacji splajnu</p> <p>0x44: sygnał zerowy powrotu do zera wyłączony</p> <p>0x45: zbyt długie przesunięcie przyspieszenia/zwalniania</p> <p>0x46: konflikt portów szybkiego licznika</p> <p>0x48: przekroczenie granicy ECAM FIFO</p> <p>0x49: błąd portu wyjścia impulsowego platformy-systemu</p> <p>0x4A: Błąd skoku segmentów TBL</p> <p>0x4B: przekroczenie limitu czasu oceny pstryknięcia szczytowego impulsu</p> <p>0x4C: brak częstotliwości ZRNR</p> <p>0x7A: błąd przepełnienia zbioru</p> <p>0x7B: dostęp do rejestrów wewnętrznych poza zakresem</p>	
<b>D8177</b>	Maksymalny okres skanowania (jednostką jest milisekunda).	O/Z



<b>D8178</b>	Odczyt potencjometru VR0.	O/Z
<b>D8179</b>	Odczyt potencjometru VR1.	O/Z
<b>D8180~D8192</b>	System zarezerwowany	O/Z
<b>D8194(D8195)</b>	Częstotliwość podstawowa portu impulsów Y000 (maksymalna częstotliwość 5 Hz ~ Y000)	O/Z
<b>D8196(D8197)</b>	Częstotliwość podstawowa portu impulsów Y001 (częstotliwość maksymalna 5 Hz ~ Y001)	O/Z
<b>D8198(D8199)</b>	Częstotliwość podstawowa portu impulsów Y002 (maksymalna częstotliwość 5 Hz ~ Y002)	O/Z
<b>D8200(D8201)</b>	Częstotliwość podstawowa portu impulsów Y003 (maksymalna częstotliwość 5 Hz ~ Y003)	O/Z
<b>D8202(D8203)~D8212(D8213)</b>	Częstotliwość podstawowa portu impulsowego Y004~Y007 i Y010~Y011 (częstotliwość maksymalna 5 Hz ~ Y000)	O/Z
<b>D8214(D8215)</b>	Maksymalna częstotliwość portu impulsów Y000	O/Z
<b>D8216(D8217)</b>	Maksymalna częstotliwość portu impulsów Y001	O/Z
<b>D8218(D8219)</b>	Maksymalna częstotliwość portu impulsów Y002	O/Z
<b>D8220(D8221)</b>	Maksymalna częstotliwość portu impulsów Y003	O/Z
<b>D8222(D8223)~D8232(D8233)</b>	Maksymalna częstotliwość portu impulsowego Y004~Y007 i Y010~Y011.	O/Z
<b>D8234</b>	Czas zwalniania portu Y0 w trybie pozycji krańcowej <b>FOLLOW</b> (adres zwiększa się o 6 dla Y1~Y5)	O/Z
<b>D8236(D8237)</b>	Dodatnia pozycja graniczna portu Y0 w trybie <b>FOLLOW</b> (adres zwiększa się o 6 dla Y1~Y5)	O/Z
<b>D8238(D8239)</b>	Ujemna pozycja graniczna portu Y0 w trybie <b>FOLLOW</b> (adres zwiększa się o 6 dla Y1~Y5)	O/Z
<b>D8294</b>	Czas zwalniania portu Y0 w trybie położenia krańcowego platformy systemowej (adres wzrasta o 6 dla Y1~Y5)	O/Z
<b>M-rejestry (rejestry bitowe)</b>		
<b>M8035</b>	Kierunek zliczania szybkiego licznika CV235.	O
<b>M8036</b>	Kierunek zliczania szybkiego licznika CV236.	O
<b>M8037</b>	Kierunek zliczania szybkiego licznika CV237 .	O
<b>M8038</b>	Kierunek zliczania szybkiego licznika CV238 .	O
<b>M8039</b>	Kierunek zliczania szybkiego licznika CV239 .	O
<b>M8040</b>	Kierunek zliczania szybkiego licznika CV240 .	O
<b>M8041</b>	Kierunek zliczania szybkiego licznika CV241 .	O
<b>M8042</b>	Kierunek zliczania szybkiego licznika CV242 .	O
<b>M8043</b>	Kierunek zliczania szybkiego licznika CV243 .	O
<b>M8044</b>	Kierunek zliczania szybkiego licznika CV244 .	O
<b>M8045</b>	Kierunek zliczania szybkiego licznika CV245 .	O



<b>M8046</b>	Bit flagi błędu inicjalizacji modułu rozszerzeń.	O
<b>M8050</b>	Mnożenie częstotliwości szybkiego licznika CV250 (0: w logice mnożenia x2, 1: w logice mnożenia x4)	O/Z
<b>M8051</b>	Mnożenie częstotliwości szybkiego licznika CV251 (0: w logice mnożenia x2, 1: w logice mnożenia x4)	O/Z
<b>M8052</b>	Mnożenie częstotliwości szybkiego licznika CV252 (0: w logice mnożenia x2, 1: w logice mnożenia x4)	O/Z
<b>M8053</b>	Mnożenie częstotliwości szybkiego licznika CV253 (0: w logice mnożenia x2, 1: w logice mnożenia x4)	O/Z
<b>M8054</b>	Mnożenie częstotliwości szybkiego licznika CV254 (0: w logice mnożenia x2, 1: w logice mnożenia x4)	O/Z
<b>M8055</b>	Mnożenie częstotliwości szybkiego licznika (0: w logice mnożenia x2, 1: w logice mnożenia x4)	O/Z
<b>M8056</b>	Kierunek zliczania szybkiego licznika CV250 .	O
<b>M8057</b>	Kierunek zliczania szybkiego licznika CV251 .	O
<b>M8058</b>	Kierunek zliczania szybkiego licznika CV252 .	O
<b>M8059</b>	Kierunek zliczania szybkiego licznika CV253 .	O
<b>M8060</b>	Kierunek zliczania szybkiego licznika CV254 .	O
<b>M8061</b>	Kierunek zliczania szybkiego licznika CV255 .	O
<b>M8062</b>	Tryb wysokiego - niskiego poziomu <b>PWMS</b> (0: wysoki po niskim, 1: wysoki przed niskim).	O/Z
<b>M8064</b>	COM2 przesyła dane.	O
<b>M8065</b>	COM2 odbiera dane.	O
<b>M8066</b>	COM2 transmisja zakończona	O/Z
<b>M8067</b>	COM2 odbiór danych zakończony.	O/Z
<b>M8068</b>	Globalna polaryzacja wszystkich portów wyjściowych impulsów (odwrócenie kierunku impulsów, gdy wynosi 1).	O/Z
<b>M8069</b>	Typ krzywej częstotliwości przyspieszania/zwalniania (0: typ w kształcie litery T, 1: typ w kształcie litery S).	O/Z
<b>M8070</b>	Jeśli chcesz zignorować błąd wyjścia impulsowego portu Y0, ustaw domyślnie 0 (0: ignoruj i kontynuuj wysyłanie impulsu, 1: zatrzymaj wysyłanie impulsu).	O/Z
<b>M8071</b>	Jeśli chcesz zignorować błąd wyjścia impulsowego portu Y1, ustaw domyślnie 0 (0: ignoruj i kontynuuj wysyłanie impulsu, 1: zatrzymaj wysyłanie impulsu).	O/Z
<b>M8072</b>	Jeśli chcesz zignorować błąd wyjścia impulsowego portu Y2, ustaw domyślnie 0 (0: ignoruj i kontynuuj wysyłanie impulsu, 1: zatrzymaj wysyłanie impulsu).	O/Z



<b>M8073</b>	Jeśli chcesz zignorować błąd wyjścia impulsowego portu Y3, ustaw domyślnie 0 (0: ignoruj i kontynuuj wysyłanie impulsu, 1: zatrzymaj wysyłanie impulsu).	O/Z
<b>M8086</b>	Nieprawidłowe ustawienie parametrów portu wyjścia impulsowego Y0.	O
<b>M8087</b>	Nieprawidłowe ustawienie parametrów portu wyjścia impulsowego Y1.	O
<b>M8088</b>	Nieprawidłowe ustawienie parametrów portu wyjścia impulsowego Y2.	O
<b>M8089</b>	Nieprawidłowe ustawienie parametrów portu wyjścia impulsowego Y3.	O
<b>M8102</b>	Kierunek impulsu wyjścia impulsowego Y0 (0: ujemny, 1: dodatni)	O
<b>M8103</b>	Kierunek impulsu wyjścia impulsowego Y1 (0: ujemny, 1: dodatni)	O
<b>M8104</b>	Kierunek impulsu wyjścia impulsowego Y2 (0: ujemny, 1: dodatni)	O
<b>M8105</b>	Kierunek impulsu wyjścia impulsowego Y3 (0: ujemny, 1: dodatni)	O
<b>M8134</b>	<b>PLSY</b> port Y0 zajęty, wysyła impulsy.	O
<b>M8135</b>	<b>PLSY</b> port Y1 zajęty, wysyła impulsy.	O
<b>M8136</b>	<b>PLSY</b> port Y2 zajęty, wysyła impulsy.	O
<b>M8137</b>	<b>PLSY</b> port Y3 zajęty, wysyła impulsy.	O
<b>M8144</b>	Platforma systemowa 1 wysyła impuls.	O
<b>M8145</b>	Platforma systemowa 2 wysyła impuls.	O
<b>M8146</b>	Platforma systemowa 3 wysyła impuls.	O
<b>M8147</b>	Platforma systemowa 4 wysyła impuls.	O
<b>M8148</b>	Platforma systemowa 5 wysyła impuls.	O
<b>M8150</b>	Zamknięte tylko podczas pierwszego okresu skanowania.	O
<b>M8151</b>	Zamknięte, gdy program zaczyna pracę.	O
<b>M8152</b>	Otwarte, gdy program zaczyna pracę.	O
<b>M8153</b>	Niskie napięcie baterii RTC (0: OK, 1: BŁĄD).	O
<b>M8154</b>	Błąd napięcia DC 24V (0: OK, 1: BŁĄD).	O
<b>M8155</b>	Błąd konfiguracji IO (0: OK, 1: BŁĄD).	O
<b>M8156</b>	Błąd sprzętowy PLC (0: OK, 1: BŁĄD).	O
<b>M8157</b>	Błąd odświeżania danych wyjściowych (0: OK, 1: BŁĄD).	O
<b>M8158</b>	Wysyłanie impulsu z interwałem 10 milisekund (0: NISKI, 1: WYSOKI).	O
<b>M8159</b>	Wysyłanie impulsu o interwale 100 milisekund (0: NISKI, 1: WYSOKI).	O
<b>M8160</b>	Wysyłanie impulsu o interwale 1-sekundy (0: NISKI, 1: WYSOKI).	O
<b>M8161</b>	Wysyłanie impulsu o interwale 1 minuty (0: NISKI, 1: WYSOKI).	O
<b>M8162</b>	Wyczyść wszystkie dane ze strefy przechowywania (0: BRAK, 1: WYCZYŚĆ).	O/Z
<b>M8163</b>	Wyczyść wszystkie dane w strefie przechowywania (0: BRAK, 1: WYCZYŚĆ).	O/Z





<b>M8164</b>	Wyłącz wszystkie wyjścia (0: WŁĄCZONE, 1: WYŁĄCZONE).	O/Z
<b>M8165</b>	Instrukcja zmiany kierunku (0: PRAWO, 1: LEWO).	O
<b>M8166</b>	Flaga przepiętnienia bitu instrukcji przesunięcia (0: OK, 1: BŁĄD).	O
<b>M8167</b>	Flaga bitowa zera instrukcji przesunięcia (0: OK, 1: BŁĄD).	O
<b>M8168</b>	Błąd konwersji BCD (0: OK, 1: BŁĄD).	O
<b>M8169</b>	Przekroczenie wyniku obliczeń (0: OK, 1: BŁĄD).	O
<b>M8170</b>	Wynik obliczenia to minus (0: OK, 1: BŁĄD).	O
<b>M8171</b>	Wynik obliczenia wynosi zero (0: OK, 1: BŁĄD).	O
<b>M8172</b>	Dzielnik wynosi zero (0: OK, 1: BŁĄD).	O
<b>M8173</b>	Adres przesunięcia poza zakresem ().	O
<b>M8174</b>	PLC w stanie błędu (0: OK, 1: BŁĄD).	O
<b>M8175</b>	Stan pracy PLC (0: STOP, 1: RUN).	O
<b>M8176</b>	COM0 przesyła dane.	O
<b>M8177</b>	COM1 przesyła dane.	O
<b>M8178</b>	COM0 odbiera dane.	O
<b>M8179</b>	COM1 odbiera dane.	O
<b>M8180</b>	Transmisja COM0 zakończona.	O/Z
<b>M8181</b>	Transmisja COM1 zakończona.	O/Z
<b>M8182</b>	Odbiór COM0 zakończony.	O/Z
<b>M8183</b>	Odbiór COM1 zakończony.	O/Z
<b>M8184</b>	Błąd komunikacji modułu rozszerzeń 1.	O
<b>M8185</b>	Błąd komunikacji modułu rozszerzeń 2.	O
<b>M8186</b>	Błąd komunikacji modułu rozszerzeń 3.	O
<b>M8187</b>	Błąd komunikacji modułu rozszerzeń 4.	O
<b>M8188</b>	Błąd komunikacji modułu rozszerzeń 5.	O
<b>M8189</b>	Błąd komunikacji modułu rozszerzeń 6.	O
<b>M8190</b>	Błąd komunikacji modułu rozszerzeń 7.	O
<b>M8191</b>	Błąd komunikacji modułu rozszerzeń 8.	O
<b>M8192</b>	Lokalna polaryzacja portu wyjścia impulsowego Y0 (odwrócenie kierunku impulsu, gdy wynosi 1).	O/Z
<b>M8193</b>	Lokalna polaryzacja portu wyjściowego impulsów Y1 (odwrócenie kierunku impulsów, gdy wynosi 1).	O/Z
<b>M8194</b>	Lokalna polaryzacja portu wyjściowego impulsów Y2 (odwrócenie kierunku impulsów, gdy wynosi 1).	O/Z





<b>M8195</b>	Lokalna polaryzacja portu wyjściowego impulsów Y3 (odwrócenie kierunku impulsów, gdy wynosi 1).	O/Z
<b>M8202</b>	Tryb zatrzymania EDRVI/EDRVA (0: zatrzymanie natychmiastowe, 1: zatrzymanie wolniej).	O/Z
<b>M8203</b>	Tryb zatrzymania FOLLOW w pozycji krańcowej (0: natychmiastowe zatrzymanie, 1: wolniejsze zatrzymanie).	O/Z
<b>M8204</b>	Tryb zatrzymania FOLLOW poniżej sygnału limitu (0: natychmiastowe zatrzymanie, 1: wolniejsze zatrzymanie).	O/Z
<b>M8205</b>	Sygnał dodatniego limitu portu Y0.	O/Z
<b>M8206</b>	Sygnał ujemnego limitu portu Y0.	O/Z
<b>M8207</b>	Sygnał dodatniego limitu portu Y1.	O/Z
<b>M8208</b>	Sygnał ujemnego limitu portu Y1.	O/Z
<b>M8209</b>	Sygnał dodatniego limitu portu Y2.	O/Z
<b>M8210</b>	Sygnał ujemnego limitu portu Y2.	O/Z
<b>M8211</b>	Sygnał dodatniego limitu portu Y3.	O/Z
<b>M8212</b>	Sygnał ujemnego limitu portu Y3.	O/Z
<b>M8225</b>	Jeśli utrzymywać kierunek po zatrzymaniu wysyłania impulsu (0: brak utrzymania, 1: utrzymanie).	O/Z
<b>M8234</b>	<b>PLSY</b> Zakończono wysyłanie impulsów portu Y0.	O
<b>M8234</b>	<b>PLSY</b> Zakończono wysyłanie impulsów portu Y1.	O
<b>M8236</b>	<b>PLSY</b> Zakończono wysyłanie impulsów portu Y2.	O
<b>M8237</b>	<b>PLSY</b> Zakończono wysyłanie impulsów portu Y3.	O



## KONTAKT

Akcesoria CNC Elżbieta Taraszkiewicz  
16-300 Augustów  
Polska  
ul. 1 Pułku Ułanów Krechowieckich 18

Telefon:  
[730 35 35 35](tel:730353535)

Napisz do nas:  
[handlowy@ebmia.pl](mailto:handlowy@ebmia.pl)

Wszelkie materiały (treści, teksty, itp.) przedstawione w instrukcji w obrębie domen: [www.ebmia.pl](http://www.ebmia.pl) oraz [www.akcesoria.cnc.info.pl](http://www.akcesoria.cnc.info.pl) są objęte prawem autorskim i podlegają ochronie na mocy „Ustawy o prawie autorskim i prawach pokrewnych” z dnia 4 lutego 1994 r. (tekst ujednolicony: Dz.U. 2006 nr 90 poz. 631).

Kopiowanie, przetwarzanie, rozpowszechnianie tych materiałów w całości lub w części bez zgody autora jest zabronione.

