

Rozdział 13 Zastosowania łącza komunikacyjnego FBs-PLC

W rozdziale 12 zawarto informację, że FBs-PLC może obsługiwać funkcję komunikacyjną „Interfejsu sterującego programem drabinkowym” przy realizacji zastosowań sieci wielopunktowej FATEK CPU lub przy łączeniu się z inteligentnymi urządzeniami peryferyjnymi za pośrednictwem portów Port 1 ~ Port 4.

Podłączenie FBs-PLC można zrealizować za pomocą funkcji CLINK(FUN151). Ponadto, FBs-PLC obsługuje interfejs komunikacyjny Modbus. Stacją master protokołu komunikacyjnego Modbus może być Port1 ~ Port 4 wykorzystujący FUN150 do łączenia Modbus z urządzeniami peryferyjnymi typu slave.

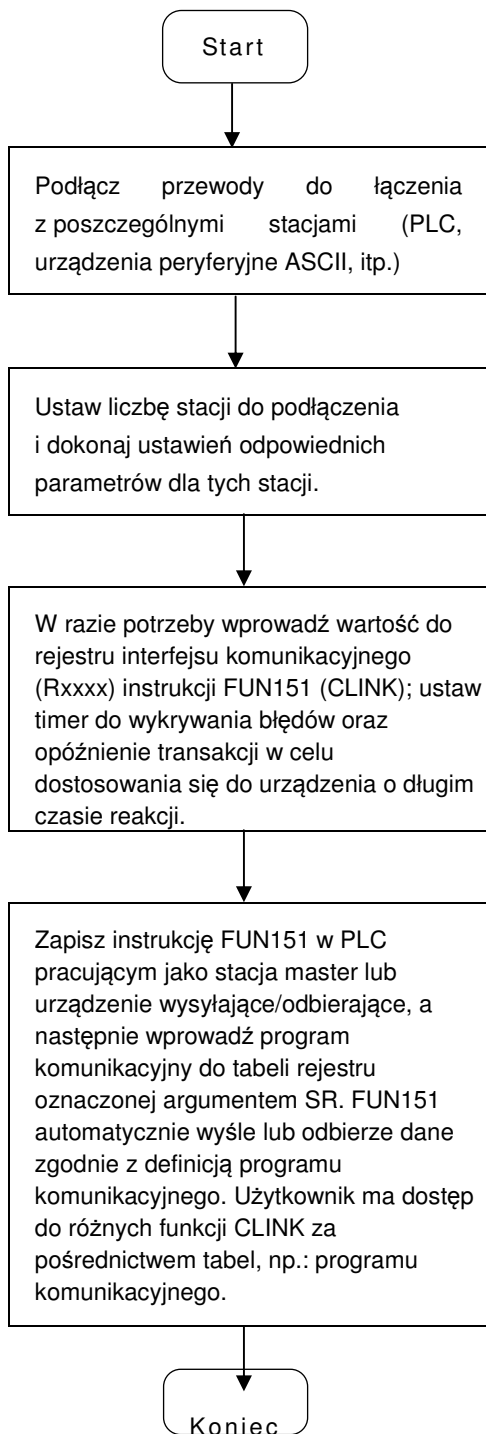
Interfejs RS-232 służy do realizacji połączeń od punktu do punktu, natomiast RS-485 do połączeń długodystansowych lub komunikacji wielopunktowej.

Instrukcja FUN151 (CLINK) obsługuje cztery tryby, od MD0 do MD3, z czego tryb MD3 jest zarezerwowany przez port 2 dla „Szybkiej sieci CPU FATEK”. Pozostałe tryby zarezerwowane są dla „Standardowego łącza komunikacyjnego”. Poniższa tabela zawiera różnice pomiędzy różnymi trybami instrukcji CLINK.

Kategoria		Element	Prędkość transmisji	Bity danych	Kod transmisji	Detekcja błędów	Prędkość przetwarzania komendy
FUN151 (CLINK)	Szybki LINK (MD3) * Tylko Port 2	38,4k bps 921,6k bps	8-bitowy	Kod binarny	CRC-16	Natychmiast	
	Standardowy LINK (MD0~MD2) * Port 1 ~ Port 4	4,8k bps 921,6k bps	7- lub 8-bitowy Regulowany	Kod ASCII	Suma kontrolna	Przetwarzanie w fazie Housekeeping cyklu skanu (na końcu skanu)	
FUN150 (M-BUS)	Modbus Master	4,8k bps 921,6k bps	7 / 8-bitowy	Kod binarny / Kod ASCII	CRC-16 / Suma kontrolna	Przetwarzanie przy skanowaniu instrukcji FUN150.	

13.1 Zastosowanie instrukcji FUN151

13.1.1 Procedury



- Liczba stacji może być ustawiona w zakresie od 1 do 254 bez powtórzeń.

- Szczegóły dotyczące parametrów komunikacji znajdują się w opisie „Ustawień dotyczących komunikacji”.

13.1.2 Opis trybów i zastosowań programu dla FUN151

W niniejszym rozdziale opisane zostaną cztery tryby (MD0 do MD3) instrukcji FUN151 (CLINK) oraz przykłady praktycznego zastosowania programu.

FUN151 CLINK	Instrukcja FUN151: MD0 (PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

Symbol drabinkowy



Pt : Przypisanie portu 1~4
 MD : 0, of praca jako stacja master Fatek w sieci CPU Link (zastosowanie protokołu komunikacyjnego Fatek)
 SR : Początkowy rejestr programu komunikacyjnego (opis w przykładzie)
 WR : Początkowy rejestr dla instrukcji (opis w przykładzie). Steruje 8 rejestrami. Nie może zostać użyty do innych programów.

Zakres	HR	ROR	DR	K
Argument	R0 □ R3839	R5000 □ R8071	D0 □ D3999	
Pt				1~4
MD				0
SR	○	○	○	
WR	○	○*	○	

Opisy

1. FUN151 (CLINK) : MD 0, PLC pracuje jako stacja master w sieci FATEK CPU Link za pośrednictwem portów 1~ 4.
2. Stacja PLC master może łączyć się z 254 stacjami slave za pomocą interfejsu RS485.
3. Tylko stacja master wymaga zastosowania instrukcji FUN151; stacja slave nie wymaga takiego zastosowania.
4. Instrukcja wykorzystuje metodę kodowania programu lub wypełniania tabeli do planowania sterowania przepływem danych, tj. tym, z której stacji slave pobierać dane do zapisania w stacji PLC master lub które dane pobrać ze stacji PLC master do zapisania w przypisanej stacji slave. Do zdefiniowania transakcji jednego pakietu danych potrzeba jedynie siedmiu rejestrów.
5. Jeżeli „EN” zmieni się z 0→1, statusy obu wejść sterujących „PAU” i „ABT” będą 0, a Port1/2/3/4 nie był wykorzystywany przez żadne inne instrukcje komunikacyjne [tj. M1960 (Port1) / M1962 (Port2) / M1936 (Port3) / M1938 (Port4) = 1], to instrukcja ta rozpocznie sterowanie portem 1/2/3/4 i ustawi status M1960/M1962/M1936/M1938 na 0 (co oznacza, że dany port jest zajęty), a następnie przejdzie do realizacji transakcji danych. W przypadku, gdy port 1/2/3/4 był już wykorzystywany (M1960/M1962/M1936/M1938 = 0), to instrukcja ta przejdzie w stan oczekiwania i pozostanie w nim do momentu, aż instrukcja komunikacyjna sterująca portami zakończy transakcję danych lub zatrzyma/porzuci wykonywane zadanie w celu zwolnienia portów (M1960/M1962/M1936/M1938 = 1). Opisywana instrukcja ustawi wówczas status M1960/M1962/M1936/M1938 na 0 i niezwłocznie przejdzie do realizacji transakcji danych.
6. Jeżeli podczas realizacji transakcji status wejścia sterującego „PAU” zmieni się na 1, to instrukcja ta po zakończeniu zadania zwolni prawo do sterowania portem (M1960/M1962/M1936/M1938 = 1). Po kolejnym przejściu praw do sterowania instrukcja rozpocznie realizację następnego pakietu transakcji.
7. Jeżeli podczas realizacji transakcji status wejścia sterującego „ABT” zmieni się na 1, to instrukcja ta niezwłocznie porzuci realizację transakcji i zwolni prawa do sterowania (M1960/M1962/M1936/M1938 = 1). Po kolejnym przejściu praw do sterowania instrukcja rozpocznie realizację pierwszego pakietu transakcji.
8. Podczas realizacji transakcji danych, wyjście „ACT” będzie aktywne.
9. Jeżeli po zakończeniu realizacji transakcji pakietu danych wystąpi błąd, to wyjścia „DN” i „ERR” będą aktywne.
10. Jeżeli po zakończeniu realizacji transakcji pakietu danych nie wystąpi błąd, to wyjście „DN” będzie aktywne.

FUN151 CLINK	Instrukcja FUN151: MD0 (PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

【Sygnały z interfejsu】

Przełączniki i rejestry dedykowane do konkretnych portów :

Wspólny port Sygnały	Port 1	Port 2	Port 3	Port 4
1. Wskaźnik gotowości portu	M1960	M1962	M1936	M1938
2. Wskaźnik zakończenia portu	M1961	M1963	M1937	M1939
3. Parametry komunik. portu	R4146	R4158	R4043	R4044
4. Opóźnienie TX i przerwa RX	R4147	R4159	R4045	R4048
5. Ustawienia czasu przerwy RX	D4043			
6. Wzbudzenie zbocza	D4044			

1. Wskaźnik gotowości portu : Sygnał jest generowany w CPU.

WŁ, port jest wolny i gotowy.

WYŁ, port jest zajęty; transakcja danych w toku.

2. Wskaźnik zakończenia portu : Sygnał jest generowany w CPU.

Kiedy program komunikacyjny zakończy realizację transakcji ostatniego pakietu danych, to wskaźnik ten będzie aktywny przez czas jednego skanowania (dla kolejnej transakcji danych).

Po zakończeniu realizacji ostatniego pakietu danych przez program komunikacyjny, sygnał ten będzie wciąż aktywny (dla transmisji jednego pakietu danych).

3. Parametry komunikacyjne portu :

Ustawienia parametrów komunikacyjnych portu (patrz rozdział o ustawieniach parametrów komunikacyjnych)

4. Opóźnienie TX i przerwa RX

Zawartość młodszego bajtu definiuje czas przerwy w odbiorze instrukcji CLINK; jednostką jest 0,01 sekundy (wartość domyślna to 50, co oznacza 0,5 sekundy). Instrukcja CLINK wykorzystuje funkcję czasu przerwy w odbiorze w celu oceny aktywności stacji slave. W przypadku łączenia wielopunktowego należy prawidłowo ustawić tę wartość (musi być ona większa niż czas 1 skanowania stacji slave o najdłuższym czasie skanowania) w celu skrócenia czasu reakcji wśród aktywnych stacji w przypadku wielu nieaktywnych stacji slave (przerwy).

Zawartość starszego bajtu definiuje czas przerwy w transmisji pomiędzy dwoma pakietami danych przy zastosowaniu instrukcji CLINK; jednostką jest 0,01 sekundy (wartość domyślna to 0).

5. Ustawianie czasu przerwy RX D4043 :

	Port1	Port2	Port3	Port4
Ustawienie czasu przerwy RX	Młodszy bajt R4147	Młodszy bajt R4159	Młodszy bajt R4045	Młodszy bajt R4048

FUN151 CLINK	Instrukcja FUN151: MD0 (PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

D4043 : Ustawienie jednostki czasu na 0,01 lub 0,1 sekundy dla wykrycia przerwy RX

Starszy bajt	Młodszy bajt							
56H	b7	b6	b5	b4	b3	b2	b1	b0

Jeżeli starszy bajt D4043 ≠56H (kod szesnastkowy), to jednostką czasu jest 0,01 sekundy.

Jeżeli starszy bajt D4043 = 56H (kod szesnastkowy), to jednostkę czasu określa młodszy bajt D4043;

b1=0, jednostka timera przerwy = 0,01 sekundy (Port 1)

=1, jednostka timera przerwy = 0,1 sekundy (Port 1)

b2=0, jednostka timera przerwy = 0,01 sekundy (Port 2)

=1, jednostka timera przerwy =0,1 sekundy (Port 2)

b3=0, jednostka timera przerwy =0,01 sekundy (Port 3)

=1, jednostka timera przerwy =0,1 sekundy (Port 3)

b4=0, jednostka timera przerwy =0,01 sekundy (Port 4)

=1, jednostka timera przerwy =0,1 sekundy (Port 4)

Na przykład, jeżeli D4043=560AH, oznacza to, że jednostką czasu dla portu 1 i 3 jest 0,1 sekundy i 0,01 sekundy dla portu 2 i 4

Jeżeli młodszy bajt R4147=50, oznacza to, że port 1 ma 5 sekund na wykrycie przerwy RX;

Jeżeli młodszy bajt R4159=50, oznacza to, że port 2 ma 0,5 sekundy na wykrycie przerwy RX;

6. Wzbudzenie zbocza D4044 :

Starszy bajt D4044=00H

Młodszy bajt D4044 : Ustawienia mające na celu zwiększenie efektywności komunikacji.

Starszy bajt	Młodszy bajt							
00H	b7	b6	b5	b4	b3	b2	b1	b0

Jeżeli starszy bajt D4044=00H (kod szesnastkowy), to młodszy bajt D4044 określa port komunikacyjny :

b1=0, Minimum 3 czasy skanowania do realizacji jednej transakcji danych (Port 1)

=1, Minimum 2 czasy skanowania do realizacji jednej transakcji danych (Port 1)

b2=0, identycznie jak w przypadku b1=0 (Port 2)

=1, identycznie jak w przypadku b1=1 (Port 2)

b3=0, identycznie jak w przypadku b1=0 (Port 3)

=1, identycznie jak w przypadku b1=1 (Port 3)

b4=0, Port 4 identycznie jak w przypadku b1=0 (Port 4)

=1, Port 4 identycznie jak w przypadku b1=1 (Port 4)

Na przykład, jeżeli D4044=0006H, oznacza to, że do realizacji transakcji danych przez port 1 i 2 wymagane są minimum 2 czasy skanowania i 3 czasy skanowania w przypadku portu 3 i 4.

FBs-PLC LINK

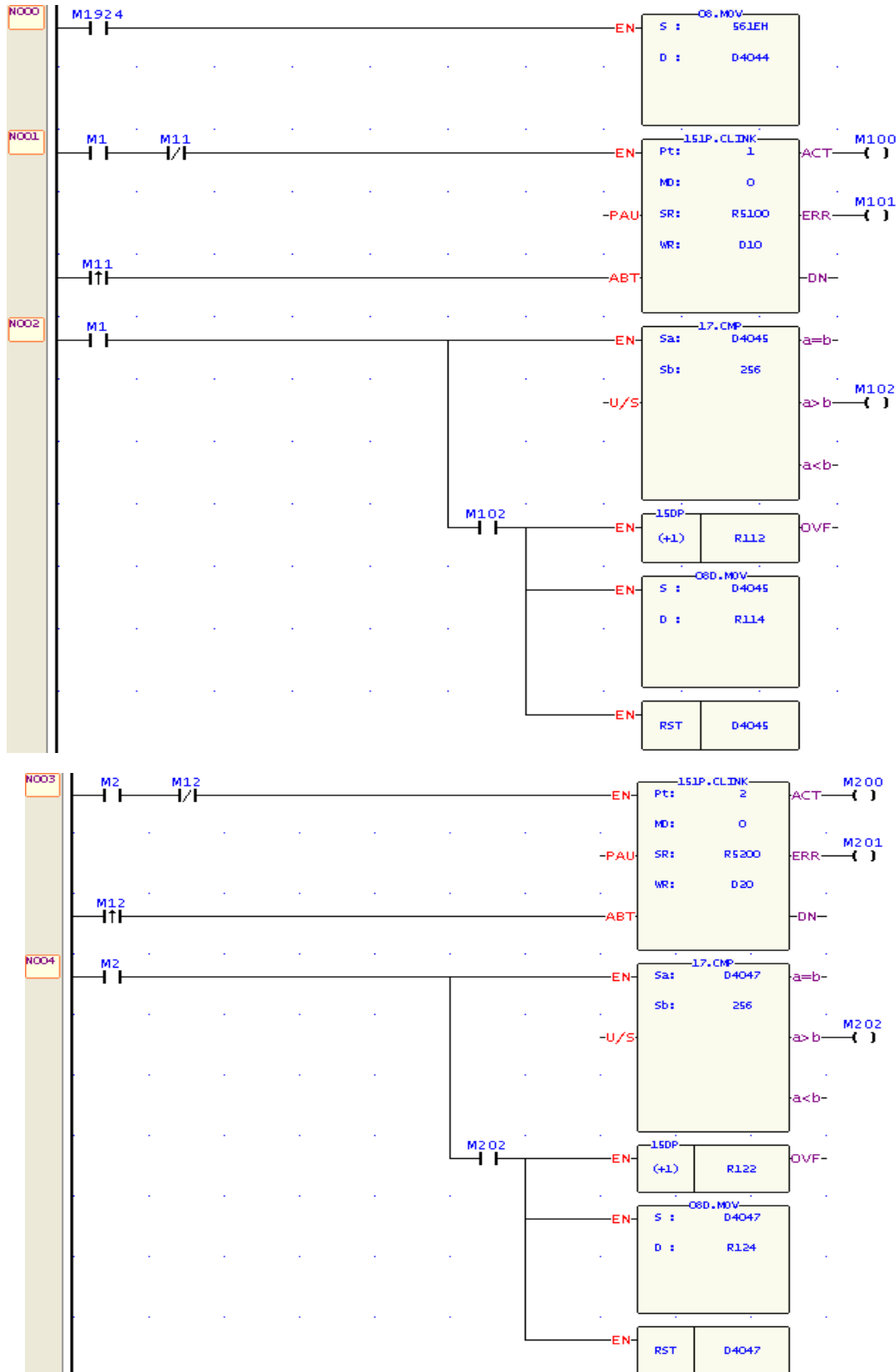
FUN151 CLINK	Instrukcja FUN151: MD0 (PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)	FUN151 CLINK																		
<p>Starszy bajt D4044=56H</p> <p>D4044 : Ustawienie jednego wzbudzenia zbocza w celu zrealizowania transakcji danych lub wzbudzenia tylko jednego zbocza w celu ciągłej realizacji transakcji.</p> <div style="text-align: center; margin: 10px 0;"> <table border="1" style="margin: auto;"> <tr> <td style="text-align: center; width: 150px;">Starszy bajt</td> <td colspan="8" style="text-align: center;">Młodszy bajt</td> </tr> <tr> <td style="text-align: center; width: 150px;">56H</td> <td style="text-align: center;">b7</td> <td style="text-align: center;">b6</td> <td style="text-align: center;">b5</td> <td style="text-align: center;">b4</td> <td style="text-align: center;">b3</td> <td style="text-align: center;">b2</td> <td style="text-align: center;">b1</td> <td style="text-align: center;">b0</td> </tr> </table> </div> <p>Jeżeli starszy bajt D4044≠56H(kod szesnastkowy), do realizacji jednej transakcji danych wymagane jest wzbudzenie tylko jednego zbocza.</p> <p>Jeżeli starszy bajt D4044=56H(kod szesnastkowy), to młodszy bajt D4044 port określa port komunikacyjny :</p> <ul style="list-style-type: none"> b1=0, jedno wzbudzenie zbocza wymagane do realizacji jednej transakcji (Port 1) =1, jedno wzbudzenie zbocza wymagane do ciągłej realizacji transakcji (Port 1) b2=0, identycznie jak w przypadku b1=0 (Port 2) =1, identycznie jak w przypadku b1=1 (Port 2) b3=0 identycznie jak w przypadku b1=0 (Port 3) =1, identycznie jak w przypadku b1=1 (Port 3) b4=0, identycznie jak w przypadku b1=0 (Port 4) =1, identycznie jak w przypadku b1=1 (Port 4) <p>Na przykład, jeżeli D4044=5618H, oznacza to, że do realizacji transakcji danych przez port 1 i 2 wymagane jest jedno wzbudzenie zbocza i tylko jedno wzbudzenie zbocza w przypadku portu 3 i 4.</p> <ul style="list-style-type: none"> ●WR+0 i WR+1 instrukcji komunikacyjnej informują o wyniku komunikacji przy każdej transakcji danych w przypadku, gdy do realizacji jednej transakcji wymagane jest wzbudzenie jednego zbocza. ●W przypadku tylko jednego wzbudzenia zbocza wymaganego do realizacji transakcji danych, wyniki komunikacji wskazywane będą przez rejestry: <ul style="list-style-type: none"> D4045 i D4046 : Wynik komunikacji przez Port 1 (jak w przypadku WR+0 i WR+1 powyżej) D4047 i D4048 : Wynik komunikacji przez Port 2 (jak w przypadku WR+0 i WR+1 powyżej) D4049 i D4050 : Wynik komunikacji przez Port 3 (jak w przypadku WR+0 i WR+1 powyżej) D4051 i D4052 : Wynik komunikacji przez Port 4 (jak w przypadku WR+0 i WR+1 powyżej) <p>Jeżeli użytkownik zechce zatrzymać proces realizacji transakcji danych należy uaktywnić wyjście ABT.</p>			Starszy bajt	Młodszy bajt								56H	b7	b6	b5	b4	b3	b2	b1	b0
Starszy bajt	Młodszy bajt																			
56H	b7	b6	b5	b4	b3	b2	b1	b0												

FUN151
CLINK

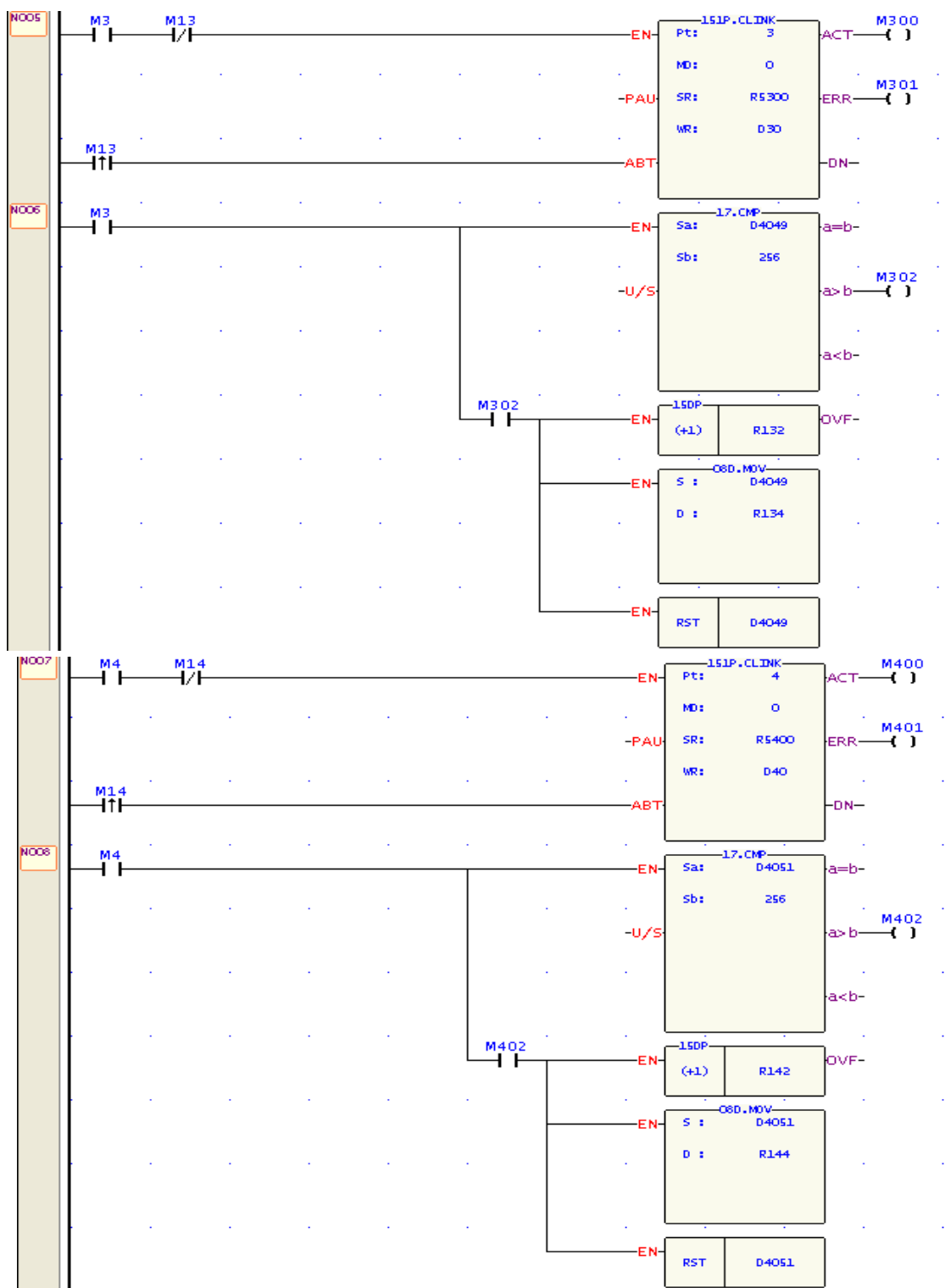
Instrukcja FUN151: MD0
(PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)

FUN151
CLINK

Program próbny do jednego wzbudzenia zbrocza w celu realizacji transakcji danych

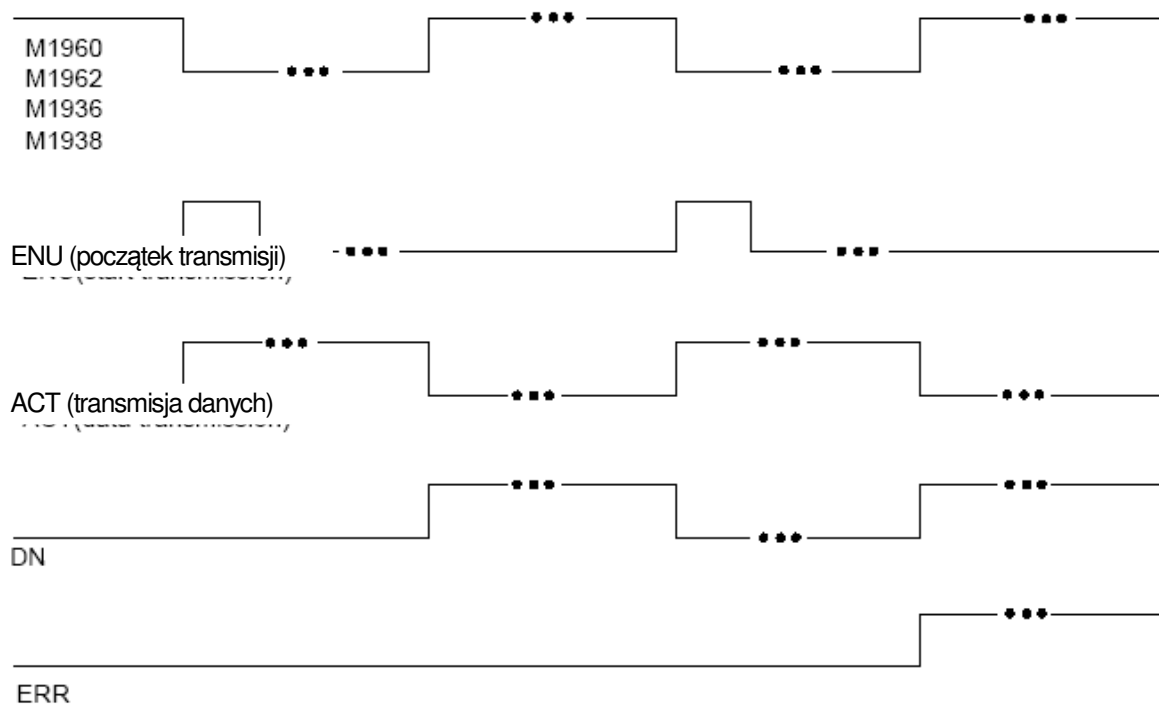


FUN151 CLINK	Instrukcja FUN151: MD0 (PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------



FUN151 CLINK	Instrukcja FUN151: MD0 (PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

Przebiegi sygnałów wejściowych i wyjściowych



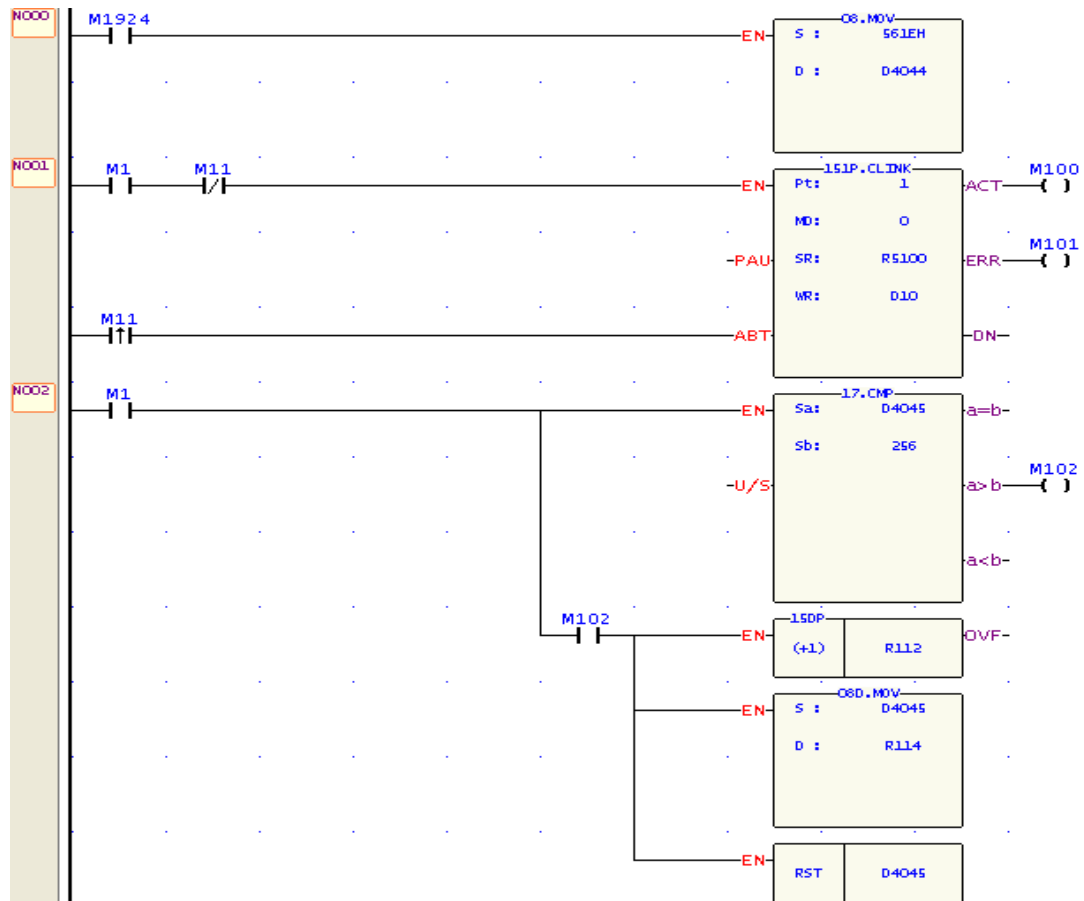
- Uwaga: 1. Jeżeli jedna transakcja zakończy się bez błędu, to tylko wyjście DN będzie aktywne.
 2. Jeżeli jedna transakcja zakończy się z błędem, to wyjścia ERR i DN będą aktywne.
 3. M1961/M1963/M1937/M1939 będą aktywne przez czas jednego skanu do momentu, aż zakończy się transakcja jednego pakietu.

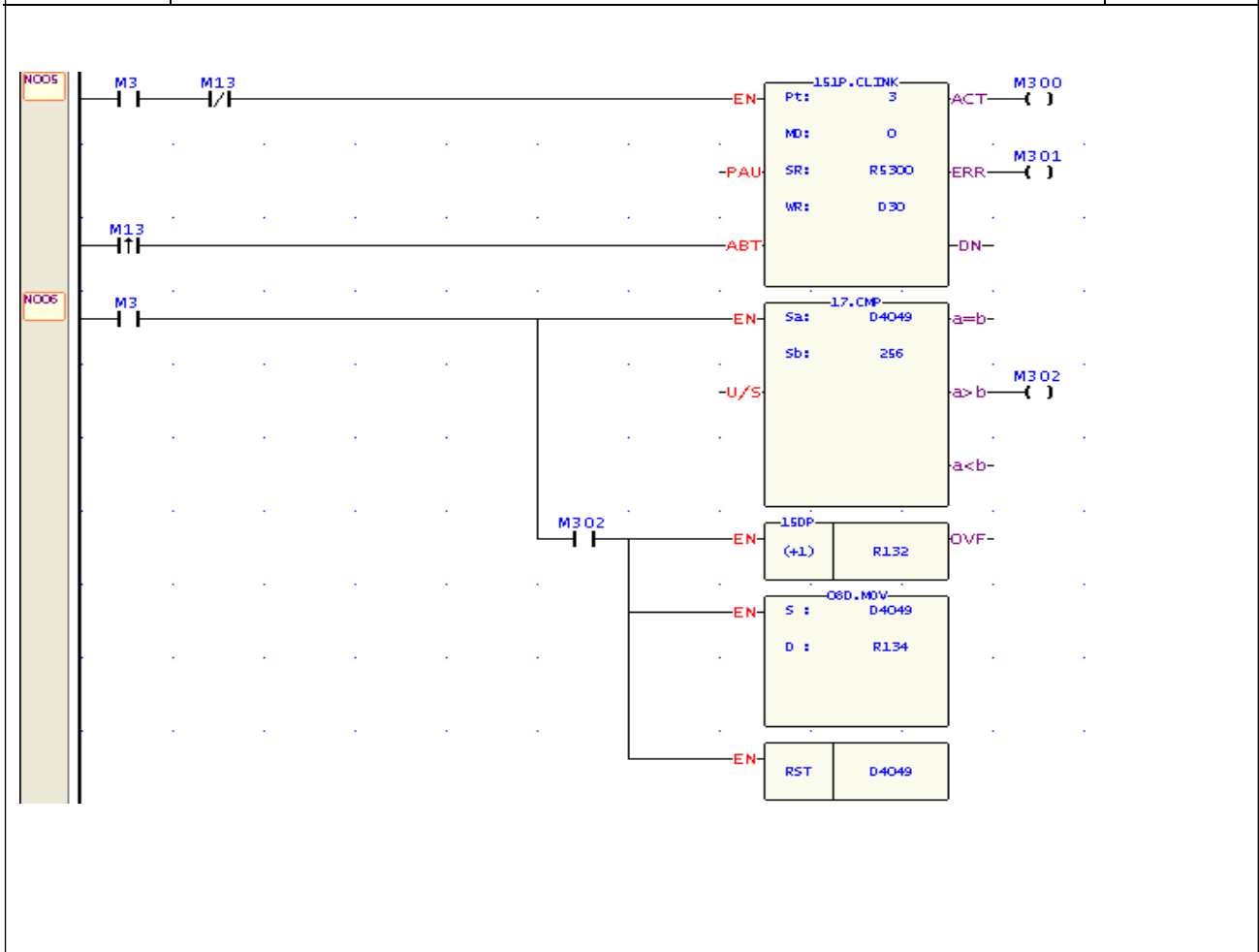
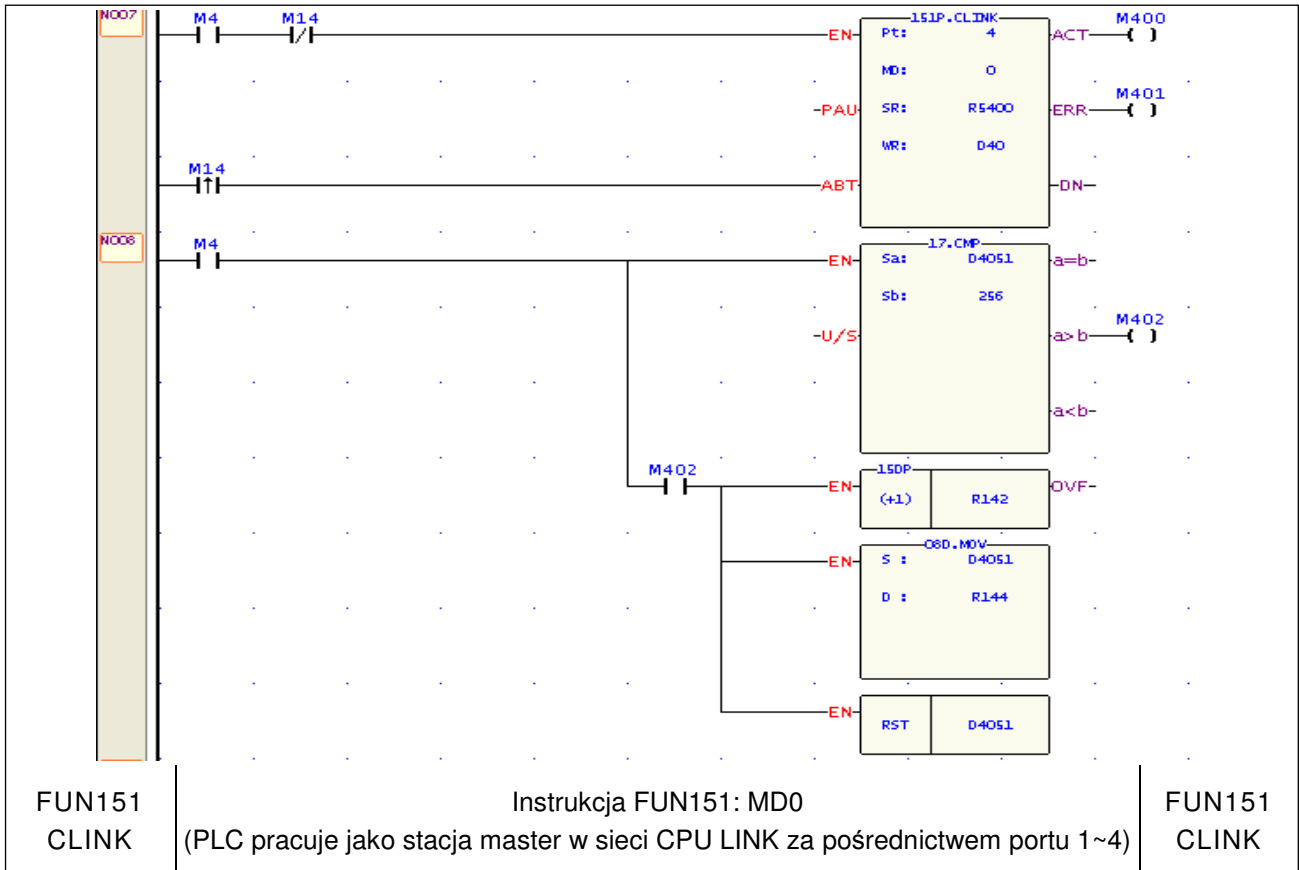
FUN151
CLINK

Instrukcja FUN151: MD0
(PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)

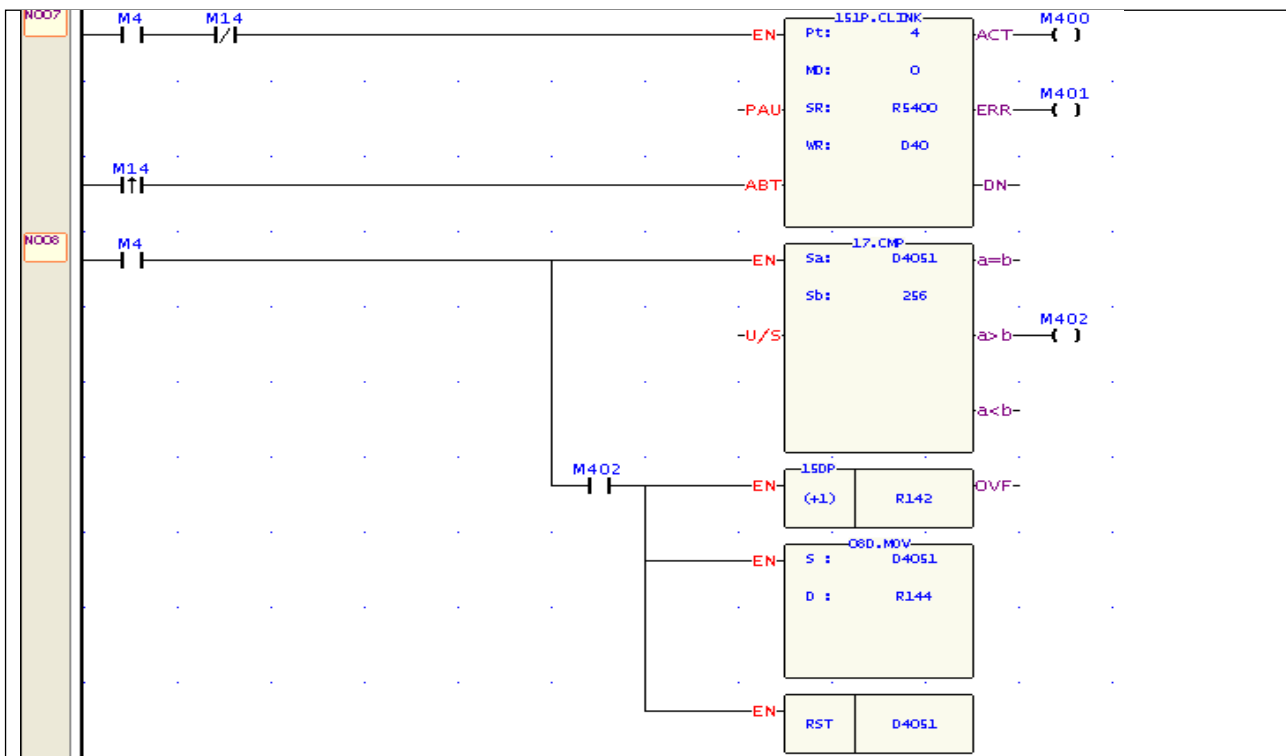
FUN151
CLINK

Program do wzbudzenia tylko jednego zbrocza i ciągłej realizacji transakcji danych





FBs-PLC LINK

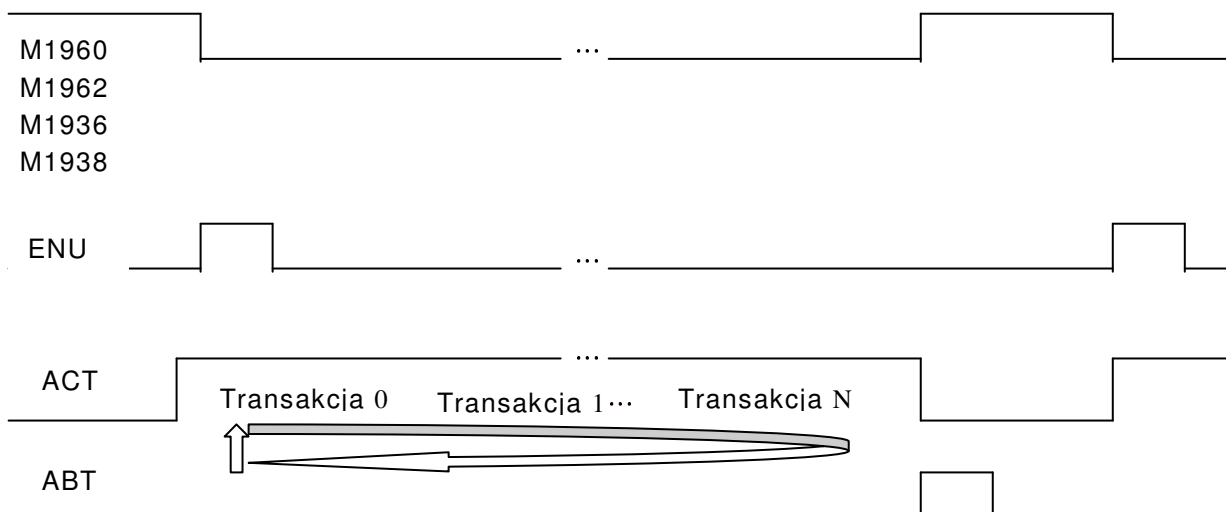


FUN151
CLINK

Instrukcja FUN151: MD0
(PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)

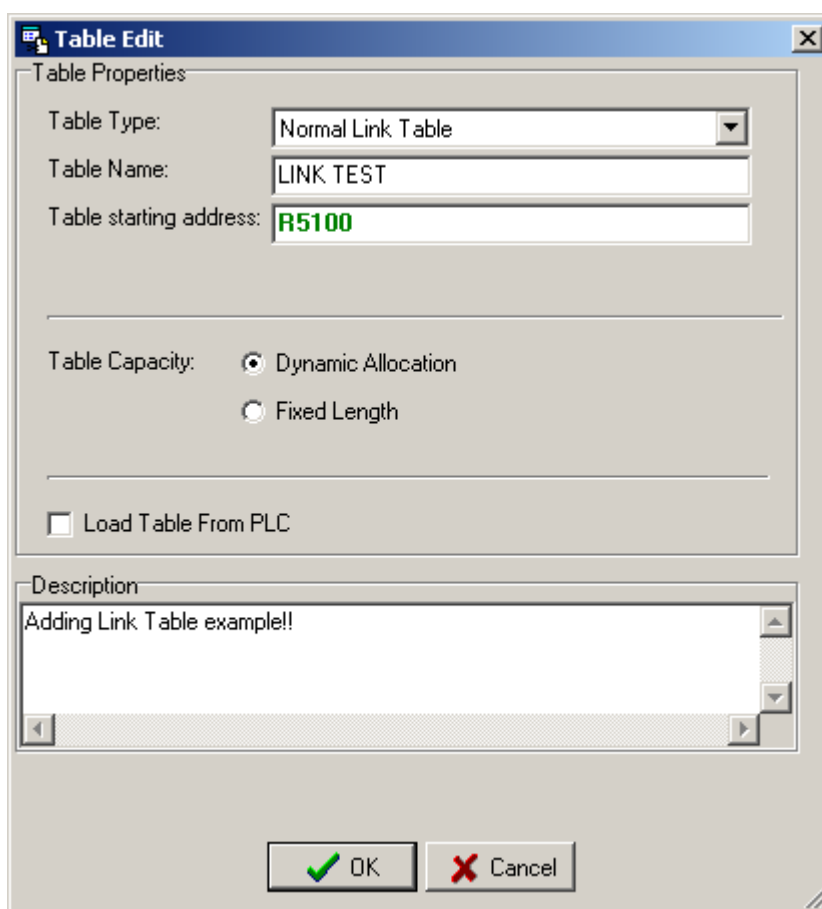
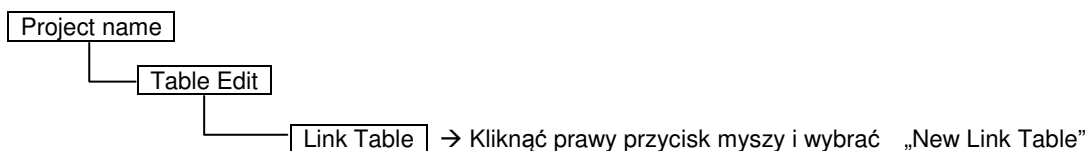
FUN151
CLINK

Przebiegi sygnałów wejściowych i wyjściowych



FUN151
CLINKInstrukcja FUN151: MD0
(PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)FUN151
CLINK**Edycja tabeli komunikacyjnej za pomocą Win ProLadder**

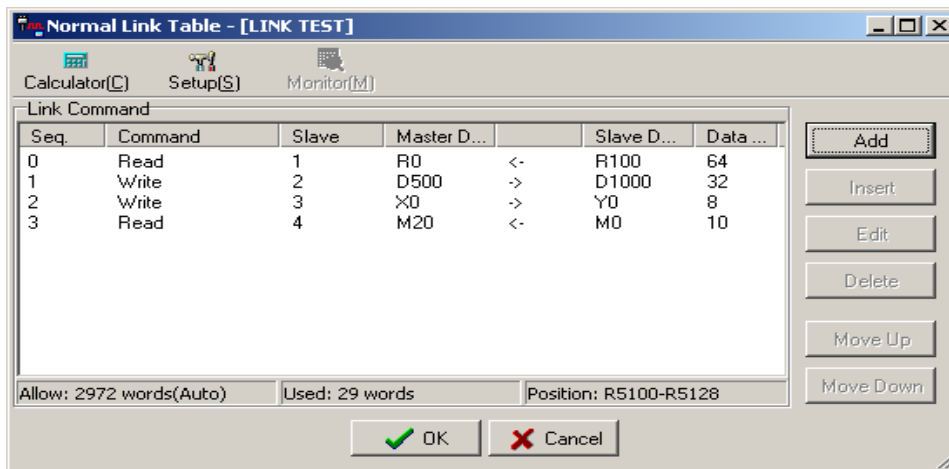
W oknie projektu wybrać opcję „Link Table”:



- Table Type : Dla MD0 wybrać „Normal Link Table”. ; Dla MD3 wybrać „High Speed Link Table”.
- Table Name : Nazwa tabeli dla edycji lub debugowania.
- Table Starting address : Adres początkowego rejestru tabeli komunikacyjnej do zapisu listy wymiany danych.

※ W celu ułatwienia edycji, odczytu i obsługi programu komunikacyjnego rozdzieliliśmy funkcje na FUN150 i FUN151. Po wybraniu FUN150 lub FUN151 i naciśnięciu przycisku „Z” pojawi się okno „Table Edit”, w którym użytkownik może edytować tabelę komunikacyjną.

FUN151 CLINK	Instrukcja FUN151: MD0 (PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------



Opis argumentu SR

SR : Początkowy rejestr programu komunikacyjnego instrukcji CLINK

SR+0	Całkowita liczba transakcji	<ul style="list-style-type: none"> • Ważny jest młodszy bajt; do zdefiniowania transakcji jednego pakietu danych wymaganych jest 7 rejestrów. • Ważny jest młodszy bajt, 0~254 (0 master oznacza, że PLC wysyła dane do wszystkich stacji slave PLC, a stacja slave PLC nie odpowiada). • Ważny jest młodszy bajt; =1, oznacza odczyt danych ze stacji slave PLC; =2, oznacza zapis danych w stacji slave PLC. • Ważny jest młodszy bajt; zakres wynosi 1~64. • Ważny jest młodszy bajt w zakresie od 0 do 13; określa typ danych stacji master PLC (patrz następna strona). • Ważne jest słowo; określa początkowy adres danych (master). • Ważny jest młodszy bajt w zakresie od 0 do 13; określa typ danych stacji slave PLC (patrz następna strona). • Ważne jest słowo; określa początkowy adres danych (slave).
SR+1	Nr stacji slave do realizacji transakcji	
SR+2	Kod komendy	
SR+3	Długość danych transakcji	
SR+4	Typ danych master PLC	
SR+5	Początkowa referencja master PLC	
SR+6	Typ danych slave PLC	
SR+7	Początkowa referencja slave PLC	
SR+8	Nr stacji slave do realizacji transakcji	
SR+9	Kod komendy	
SR+10	Długość danych transakcji	
SR+11	Typ danych master PLC	
SR+12	Początkowa referencja master	
SR+13	Typ danych slave PLC	
SR+14	Początkowa referencja slave PLC	

} Opis transakcji drugiego pakietu.

FUN151 CLINK	Instrukcja FUN151: MD0 (PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

Typ danych, kod i nr referencyjny master / slave

Kod danych	Typ danych	Numer referencyjny
0	X (wejście dyskretne)	0~255
1	Y (wyjście dyskretne)	0~255
2	M (wewnętrzny przekaźnik M)	0~1911
3	S (przekaźnik krokowy S)	0~999
4	T (styk timera)	0~255
5	C (styk licznika)	0~255
6	WX (słowo na dyskretnym wejściu ,16 bitów)	0~240, musi być wielokrotnością 8.
7	WY (słowo na dyskretnym wyjściu ,16 bitów)	0~240, musi być wielokrotnością 8.
8	WM (słowo w wewnętrznym przekaźniku,16 bitów)	0~1896, musi być wielokrotnością 8.
9	W S (słowo w przekaźniku krokowym,16 bits)	0~984, musi być wielokrotnością 8.
10	TR (rejestr timera)	0~255
11	CR (rejestr licznika)	0~199
12	R (rejestr danych Rxxxx)	0~3839
13	D (rejestr danych Dxxxx)	0~4095

Uwaga : Typ danych dla stacji master i slave musi być konsekwentny, to znaczy, że jeżeli wartość dla stacji master będzie liczbą z zakresu od 0 do 5, to wartość dla stacji slave musi być także liczbą z zakresu od 0 do 5; natomiast jeżeli wartość dla stacji master będzie liczbą z zakresu od 6 do 13, to wartość dla stacji slave musi być także liczbą z zakresu od 6 do 13.

operand opis argumentu WR dla FUN151:MD0

	Starszy bajt	Młodszy bajt	
WR+0	Kod wyniku	Nr transakcji	<ul style="list-style-type: none"> Kod wyniku określa wynik transakcji; 0= normalny, inna wartość= nienormalny. Nr transakcji określa, która transakcja jest aktualnie realizowana. Nr stacji określa numer stacji realizującej transakcję. Kod komendy <ul style="list-style-type: none"> =40H, odczyt statusu systemu ze stacji slave PLC. =44H, odczyt statusów zmiennej dyskretnej ze stacji slave PLC. =45H, zapis statusów zmiennej dyskretnej w stacji slave PLC. =46H, odczyt rejestrów ze stacji slave PLC. =47H, zapis rejestrów w stacji slave PLC. WR+4's b0=1, Port jest zajęty, a instrukcja oczekuje na przyznanie praw do transmisji w celu realizacji transakcji danych. <ul style="list-style-type: none"> b4=1 , performing instrukcja nie jest realizowana po raz pierwszy. b12 , Wskazanie na wyjściu „ACT” b13 , Wskazanie na wyjściu „ERR”. b14 , Wskazanie na wyjściu „DN”.
WR+1	Nr stacji	Kod komendy	
WR+2	Do operacji wewnętrznych		
WR+3	Do operacji wewnętrznych		
WR+4	Do operacji wewnętrznych		
WR+5	Do operacji wewnętrznych		
WR+6	Do operacji wewnętrznych		
WR+7	Do operacji wewnętrznych		

Kod wyniku:

- 0, transakcja przebiegła pomyślnie.
- 2, błąd długości danych (długość danych wynosi 0 lub jest większa od 64 dla jednej transakcji).
- 3, błąd kodu komendy (kod komendy jest większy od 2).
- 4, błąd typu danych (typ danych jest większy od 13; patrz kod typu danych).
- 5, błąd numeru referencyjnego (patrz numer referencyjny).
- 6, nieprawidłowość w typie danych (np.: 0~5 dla stacji master i 6~13 dla stacji slave).
- A, brak odpowiedzi ze stacji slave (błąd przerwy).
- B, błąd komunikacji (odebrano dane o błędzie).

FUN151 CLINK	Instrukcja FUN151: MD0 (PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

- W celu ułatwienia programowania i rozwiązywania błędów, WInProladder obsługuje środowisko do edycji tabeli komunikacyjnej instrukcji FUN151; Aby przejść do trybu edycji tabel należy najpierw wprowadzić całą instrukcję FUN151, a następnie przesunąć kursor na pozycję tej funkcji i nacisnąć przycisk „Z”. Użytkownik może stworzyć tu nową tabelę komunikacyjną lub wyświetlić istniejącą tabelę posługując się interfejsem przyjaznym dla użytkownika.

Tabela komunikacyjna dla FUN151:MD0

Nr sekwencji	Komenda	Slave	Dane master	Dane slave	Długość
0 ~ nnn	Czytaj (=1) Zapisz (=2)	Numer stacji slave PLC realizującej transakcję. Numer stacji=0, Stacja master PLC wysyła dane do wszystkich stacji slave PLC. Stacje slave nie odpowiadają. Numer stacji=N, Numer stacji slave PLC realizującej transakcję ze stacją master PLC. N=1 ~ 254	Typ danych i numer referencyjny transakcji pakietu danych dla stacji master PLC. X0 ~ X255 Y0 ~ Y255 M0 ~ M1911 S0 ~ S999 T0 ~ T255 C0 ~ C255 WX0 ~ WX240 WY0 ~ WY240 WM0 ~ WM1896 WS0 ~ WS984 TR0 ~ TR255 CR0 ~ CR199 R0 ~ R3839 D0 ~ D4095	Typ danych i numer referencyjny transakcji pakietu danych dla stacji slave PLC. X0 ~ X255 Y0 ~ Y255 M0 ~ M1911 S0 ~ S999 T0 ~ T255 C0 ~ C255 WX0 ~ WX240 WY0 ~ WY240 WM0 ~ WM1896 WS0 ~ WS984 TR0 ~ TR255 CR0 ~ CR199 R0 ~ R3839 D0 ~ D4095	Długość danych dla tej transakcji. 1 ~ 64

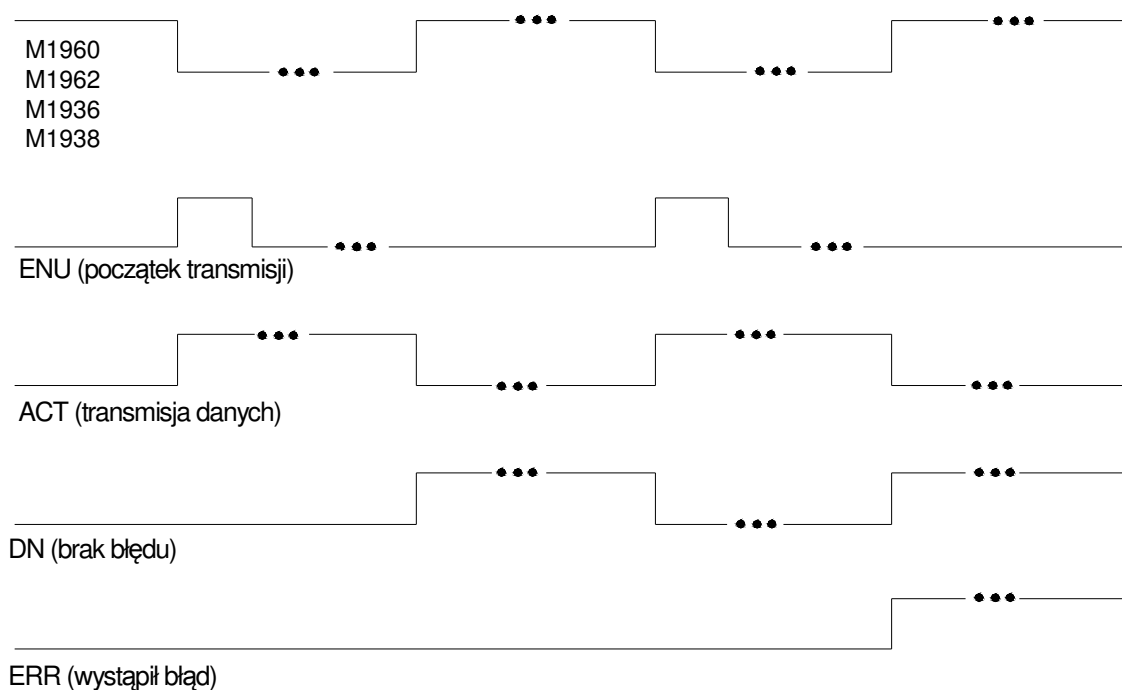
Przykład programu

Jeżeli M1/M2/M3/M4 = WŁ, a odpowiedni port nie jest zajęty przez żadną inną instrukcję komunikacyjną (M1960, M1962, M1936, M1938 = ON), to instrukcja CLINK rozpocznie realizację transakcji danych. W trakcie realizacji transakcji danych M1960, M1962, M1936, M1938 = WYŁ. Po zakończeniu realizacji M1960, M1962, M1936, M1938 = WYŁ. Zmiana M1960, M1962, M1936, M1938 z WYŁ ↔ WŁ (wejście sterujące „ENU” FUN151) może automatycznie rozpocząć transakcję kolejnego pakietu danych (po zakończeniu ostatniej transakcji program powróci do pierwszego pakietu w celu realizacji automatycznej transmisji cyklicznej).

- Wskaźniki wyjściowe : „ACT” WŁ : Transakcja w toku
„ERR” WŁ : Wystąpił błąd (patrz kod wyniku)
„DN” WŁ : Zakończono jedną transakcję

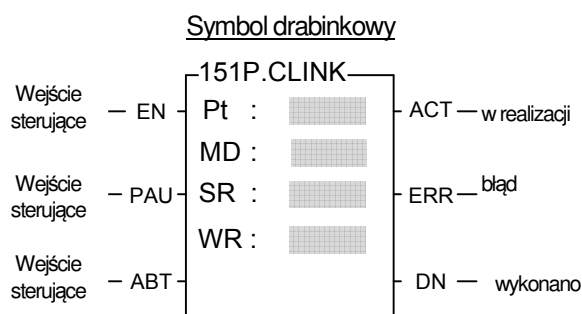
FUN151
CLINKInstrukcja FUN151: MD0
(PLC pracuje jako stacja master w sieci CPU LINK za pośrednictwem portu 1~4)FUN151
CLINK

Przebiegi sygnałów wejściowych i wyjściowych



- Uwaga :
1. Jeżeli jedna transakcja zakończy się bez błędu, to tylko wyjście DN będzie aktywne.
 2. Jeżeli jedna transakcja zakończy się z błędem, to wyjścia ERR i DN będą aktywne.
 3. M1961/M1963/M1937/M1939 będą aktywne przez czas jednego skanu do momentu, aż zakończy się transakcja jednego pakietu.

FUN151 CLINK	Instrukcja FUN151: MD1 (PLC pracuje jako stacja wysyłająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	---	-----------------



Pt : port przypisanie portu, 1~4
 MD : 1, połączenie z inteligentnymi urządzeniami peryferyjnymi wyposażonymi w interfejs komunikacyjny.
 SR : Początkowy rejestr tabeli transmisji danych
 WR : Początkowy rejestr dla realizacji instrukcji (patrz przykład). Steruje 8 rejestrami i nie może być wykorzystywany przez inne programy.

Zakres	HR	ROR	DR	K
Argument	R0 □ R3839	R5000 □ R8071	D0 □ D3999	
Pt				1~4
MD				1
SR	○	○	○	
WR	○	○*	○	

Opis

1. FUN151:MD1 sprawia, że PLC pracuje jako urządzenie nadawcze w komunikacji z inteligentnymi urządzeniami peryferyjnymi wyposażonymi w interfejs komunikacyjny.
2. Stacja master PLC może łączyć się z urządzeniami peryferyjnymi wyposażonymi w identyczny protokół komunikacyjny za pośrednictwem interfejsu RS-485.
3. Protokół/format komunikacyjny zapisywany jest za pomocą programu DRABINKOWEGO, który musi być zgodny z połączonymi urządzeniami peryferyjnymi.
4. Jeżeli „EN” zmieni się z 0→1, status na obydwu wejściach „PAU” i „ABT” będzie 0, a port 1/2/3/4 nie był zajęty przez inne instrukcje komunikacyjne [tj. M1960 (Port1)/M1962 (Port2)/M1936 (Port3)/M1938 (Port4) = 1], to instrukcja ta niezwłocznie przejmie kontrolę nad portem 1/2/3/4 i ustawi status M1960/M1962/M1936/M1938 na 0 (co oznaczać będzie zajętość portu), a następnie przejdzie do realizacji transakcji danych. W przypadku, gdy port 1/2/3/4 był już zajęty (M1960/M1962/M1936/M1938 = 0), to instrukcja przejdzie w stan oczekiwania i pozostanie w nim do momentu aż instrukcja komunikacyjna sterująca portami zakończy realizację transakcji lub zatrzyma / porzuci swoje zadanie i zwolni prawa do kontroli (M1960,/M1962,/M1936/M1938 =1). Opisywana instrukcja przejmie sterowanie, ustawi status M1960/M1962/M1936/M1938 na 0 i niezwłocznie przejdzie do realizacji transakcji danych.
5. Jeżeli podczas realizacji transakcji status wejścia „PAU” zmieni się na 1, to instrukcja ta po zakończeniu transmisji aktualnie przesyłanych danych zatrzyma wykonywane zadanie i zwolni prawa do kontroli portu (ustawi M1960/M1962/M1936/M1938 na 1).
6. Jeżeli podczas realizacji transakcji status wejścia „ABT” zmieni się na 1, to instrukcja ta zakończy transmisję danych i niezwłocznie zwolni prawa do kontroli portów (ustawi M1960/M1962/M1936/M1938 na 1).
7. W trakcie realizacji transakcji, wejście „ACT” będzie aktywne.
8. Jeżeli po zakończeniu realizacji transakcji pakietu danych (zakończona transmisja lub zakończone zadanie „wyślij i odbierz”) pojawi się błąd, to wyjścia „DN” i „ERR” będą aktywne.
9. Jeżeli po zakończeniu realizacji transakcji pakietu danych (zakończona transmisja lub zakończone zadanie „wyślij i odbierz”) nie pojawi się żaden błąd, to wyjście „DN” będzie aktywne.

FUN151 CLINK	Instrukcja FUN151: MD1 (PLC pracuje jako stacja wysyłająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	---	-----------------

【Sygnały z interfejsu】

Przełączniki i rejestry dedykowane do portu :

Sygnały \ Port kom.	Port 1	Port 2	Port 3	Port 4
1. Wskaźnik gotowości portu	M1960	M1962	M1936	M1938
2. Wskaźnik zakończenia portu	M1961	M1963	M1937	M1939
3. Parametry komunik. portu	R4146	R4158	R4043	R4044
4. Opóźnienie TX i przerwa RX	R4147	R4159	R4045	R4048
5. Ustawienia czasu przerwy RX	D4043			
6. Wzbudzenie zbocza	D4044			

1. Wskaźnik gotowości portu : Sygnał jest generowany w CPU.

WŁ, port jest wolny i gotowy.

WYŁ, port jest zajęty; transakcja danych w toku.

2. Wskaźnik zakończenia portu : Sygnał jest generowany w CPU.

WŁ, transakcja danych została zakończona.

3. Parametry komunikacyjne portu :

Ustawienia parametrów komunikacyjnych portu (patrz rozdział o ustawieniach parametrów komunikacyjnych)

4. Opóźnienie TX i przerwa RX :

Zawartość młodszego bajtu definiuje czas przerwy w odbiorze instrukcji CLINK; jednostką jest 0,01 sekundy (wartość domyślna to 50, co oznacza 0,5 sekundy).

Instrukcja CLINK wykorzystuje funkcję czasu przerwy w odbiorze w celu oceny aktywności stacji slave. Jeżeli po wysłaniu komendy odczytu / zapisu z PLC do stacji slave, stacja slave nie odpowie w przeciągu czasu określonego tym parametrem, to oznacza to, że podczas komunikacji zaistniała przerwa. W przypadku łączenia wielopunktowego należy prawidłowo ustawić tę wartość (musi być ona większa niż czas 1 skanowania stacji slave o najdłuższym czasie skanowania) w celu skrócenia czasu reakcji wśród aktywnych stacji w przypadku wielu nieaktywnych stacji slave (przerwy).

Zawartość starszego bajtu jest nieważna w tym trybie.

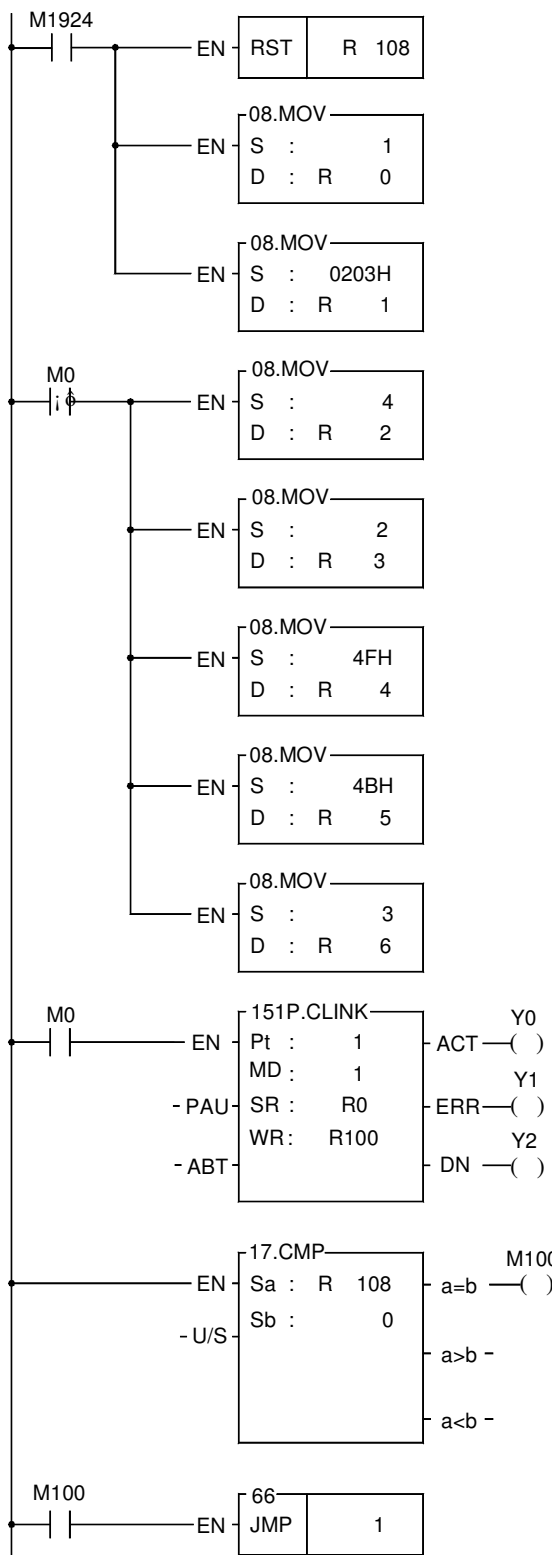
5. Ustawienia czasu przerwy RX: Szczegóły w niniejszym rozdziale, strona 13-4~13-5

6. Wzbudzenie zbocza: Szczegóły w niniejszym rozdziale, strona 13-5~13-6.

※ Jeżeli komunikat zostanie odebrany bez kodu końcowego i M1956=1, to do określenia tego czy dane zostały odebrane wykorzystywany jest starszy bajt R4148 czasu przerwy; jednostką jest 0,001 sekundy (domyślnie 0CH, 12ms).

FUN151 CLINK	Instrukcja FUN151: MD1 (PLC pracuje jako stacja wysyłająca za pośrednictwem portu 1~4)	FUN151 CLINK
<p data-bbox="177 371 729 405">Test pętli zwrotnej na przykładzie programu</p> <p data-bbox="220 432 1390 575">Stacja A PLC wysyła dane do stacji B PLC (stacja B PLC odsyła odebrane dane z powrotem do stacji A – test pętli zwrotnej) i sprawdza czy odpowiedź ze stacji B jest tożsama z wysłanymi danymi. W ten sposób, za pomocą prostego testu można zbadać, czy sprzęt i oprogramowanie portu 1 PLC działa prawidłowo czy błędnie.</p>		

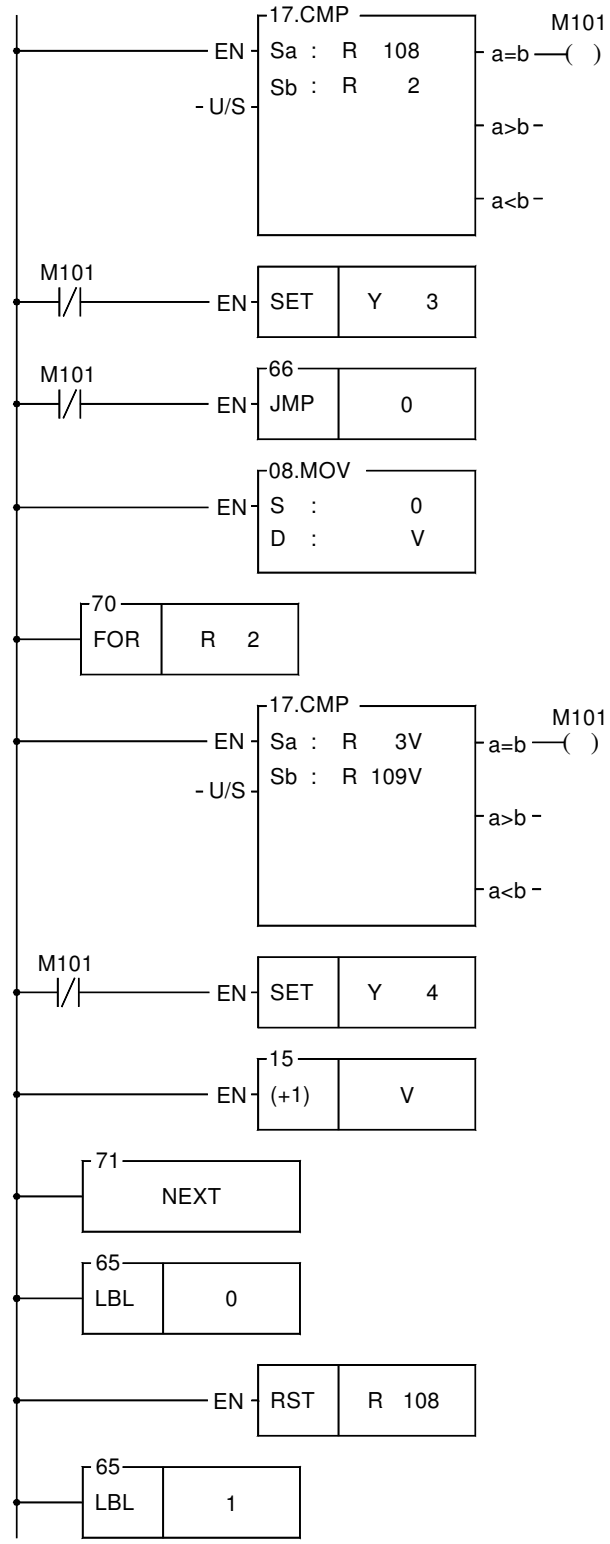
FUN151 CLINK	Instrukcja FUN151: MD1 (PLC pracuje jako stacja wysyłająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	---	-----------------



- Zeruj długość odbieranych danych. (trybie „wyślij” instrukcja ta nie jest wymagana).
- Ustaw tryb roboczy:
 - ustaw tryb "wyślij i odbierz" (R0=1)
- Ustaw początkowy (02H) i końcowy (03H) kod odbioru wiadomości (regularny odbiór R1=0 jest możliwy także bez podania początkowych i końcowych kodów).
- Pakuj dane do transmisji:
- Ustaw długość transmisji danych (R2=N).
- Wprowadź dane do transmisji:
 - Wprowadź dane 1 (R3= ' STX ')
 - Wprowadź dane 2 (R4= ' O ')
 - Wprowadź dane 3 (R5= ' K ')
 - Wprowadź dane 4 (R6= ' ETX ')
- Po wybraniu trybu „wyślij i odbierz”, program wykorzysta instrukcję porównawczą do określenia, czy wiadomość została odebrana; jeżeli tak, to M100 = WYŁ., a program przetworzy odebrane dane. (Program ten nie jest wymagany w trybie „wyślij”.)

FUN151 CLINK	Instrukcja FUN151: MD1 (PLC pracuje jako stacja wysyłająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	---	-----------------

- Program przetwarzający odebrane dane.
- Szczegóły dotyczące odebranych danych znajdują się na następnej stronie.



• Porównaj długość odebranych i wysyłanych danych.

• Zasygnalizuj błąd w przypadku nieprawidłowej długości.

• Wyzeruj wskaźnik V.

• Porównaj zgodność wszystkich odebranych i wysyłanych danych.

• Porównaj odebrane i wysłane dane jedno po drugim.

• Zasygnalizuj błąd w przypadku różnicy danych.

• Po zakończeniu przetwarzania wyzeruj długość odebranych danych i przygotuj układ do odbioru nowych danych.

FUN151 CLINK	Instrukcja FUN151: MD1 (PLC pracuje jako stacja wysyłająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	---	-----------------

Opis argumentu SR dla FUN151: MD1

SR : Początkowy rejestr tabeli transmisji danych

SR+0	Wyślij lub Odbierz i wyślij	<ul style="list-style-type: none"> • Ważny jest młodszy bajt, =00H, tylko wyślij, brak odpowiedzi ze stacji slave =01H, wyślij i odbierz dane (odbierz tylko w przypadku braku błędów) =81H, wyślij i odbierz dane (odbierz nawet w przypadku błędów)
SR+1	Początkowy i końcowy kod odbioru	<ul style="list-style-type: none"> • Starszy bajt : Początek tekstu dla odbioru. Młodszy bajt : Koniec tekstu dla odbioru.
SR+2	Długość transmisji	<ul style="list-style-type: none"> • Maksymalna długość danych do wysłania to 511
SR+3	Dane 1	<ul style="list-style-type: none"> • Ważny jest młodszy bajt
SR+4	Dane 2	<ul style="list-style-type: none"> • Ważny jest młodszy bajt
SR+5	Dane 3	<ul style="list-style-type: none"> • Ważny jest młodszy bajt
SR+6	Dane 4	<ul style="list-style-type: none"> • Ważny jest młodszy bajt
• • •	• • •	• • •
	Dane N	

Uwaga 1 : W przypadku wybrania trybu „wyślij”, początkowy/końcowy kod odbioru jest nieistotny.

- 2 : Przed rozpoczęciem transmisji danych w trybie „wyślij i odbierz” program musi najpierw określić początkowy i końcowy kod wiadomości zwrotnej i zapisać go w rejestrze kodów początkowych/końcowych (np.: SR+1=0203H, 02H jest kodem końcowym a 03H kodem początkowym) w celu zapewnienia prawidłowego odbioru danych. Protokół komunikacyjny z początkowym/końcowym kodem ułatwia identyfikację każdego pakietu i sprawia, że program komunikacyjny jest prosty i wydajny.
- 3 : Kiedy układ jest w trybie „wyślij i odbierz”, a wiadomość zwrotna nie zawierała początkowego kodu, program wypełnia starszy bajt rejestru początkowych/końcowych kodów zerami. Program także ustawi starszy bajt R4148 (czas pomiędzy wykrywaniem poszczególnych wiadomości) w celu sprawdzenia czy pakiet danych został odebrany w całości; jednostką jest 0,001 sekundy (domyślnie 0CH, 12ms).

FUN151 CLINK	Instrukcja FUN151: MD1 (PLC pracuje jako stacja wysyłająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	---	-----------------

Protokół komunikacyjny bez końcowego kodu zależy od czasu pomiędzy wykrywaniem poszczególnych wiadomości, który służy do oceny, czy odebrany został kompletny pakiet danych (czas ten musi być większy od maksymalnego opóźnienia pomiędzy bajtami przy odpowiedzi), co umożliwia zakończenie zadania dopiero po odebraniu całego pakietu. Mówiąc ogólnie, dane są wysyłane bajt po bajcie, a co za tym idzie, pomiędzy poszczególnymi bajtami pojawiają się przerwy (większe od czasu pomiędzy wykrywaniem poszczególnych wiadomości) oznaczające, że transmisja pakietu danych przebiegła pomyślnie.

Opis argumentu WR dla FUN151:MD1

	Starszy bajt	Młodszy bajt	
WR+0	Kod wyniku	0	• Kod wyniku =0, OK ; = inne wartości, nieprawidłowy.
WR+1	Do operacji wewnętrznych		• Rejestry robocze dla instrukcji CLINK.
WR+2	Do operacji wewnętrznych		
WR+3	Do operacji wewnętrznych		
WR+4	Do operacji wewnętrznych		• WR+4 : b0=1, W toku
WR+5	Do operacji wewnętrznych		b12 = wskazanie na wyjściu „ACT”
WR+6	Do operacji wewnętrznych		b13 = wskazanie na wyjściu „ERR”
WR+7	Do operacji wewnętrznych		b14 = wskazanie na wyjściu „DN”
WR+8	Całkowita ilość odebranych danych		• Całkowita ilość odbieranych bajtów (rejestr do zapisu długości odbieranych danych; zawiera początkowy i końcowy kod).
WR+9	Dane 1		• Pierwszy bajt odebranych danych (jest to początkowy kod); Starszy bajt =0.
•	Dane 2		• Drugi bajt odebranych danych; Starszy bajt =0.
•	Dane 3		• Trzeci bajt odebranych danych; Starszy bajt =0.
•	⋮		
•	Dane N		• N-ty bajt odebranych danych (jest to końcowy kod); Starszy bajt =0.

Kod wyniku : 0, transakcja zakończona pomyślnie.

2, błąd długości danych (wartość wynosi 0, lub pakiet transakcji wynosi ponad 511)


A, brak odpowiedzi ze stacji slave

B, nieprawidłowa komunikacja (błąd odbioru danych)

• Wskazanie na wyjściu : „ACT” WŁ : Transakcja w toku

„ERR” WŁ : Wystąpił błąd

„DN” WŁ : Zakończono jedną transakcję

FUN151 CLINK	Instrukcja FUN151: MD2 (PLC pracuje jako stacja odbierająca za pośrednictwem portu 1~4)	FUN151 CLINK																														
<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p><u>Symbol drabinkowy</u></p>  </div> <div style="width: 50%;"> <p>Pt : przypisanie portu, 1~4</p> <p>MD : 2, PLC oczekuje na odebranie wiadomości wysłanych z inteligentnych urządzeń peryferyjnych</p> <p>SR : Początkowy rejestr tabeli transmisji danych</p> <p>WR : Początkowy rejestr dla realizacji instrukcji (patrz przykład). Steruje 8 rejestrami i nie może być wykorzystywany przez inne programy.</p> </div> </div> <table border="1" data-bbox="577 703 1018 913"> <thead> <tr> <th>Zakres</th> <th>HR</th> <th>ROR</th> <th>DR</th> <th>K</th> </tr> </thead> <tbody> <tr> <td>Argument</td> <td>R0 □ R3839</td> <td>R5000 □ R8071</td> <td>D0 □ D3999</td> <td></td> </tr> <tr> <td>Pt</td> <td></td> <td></td> <td></td> <td>1~4</td> </tr> <tr> <td>MD</td> <td></td> <td></td> <td></td> <td>2</td> </tr> <tr> <td>SR</td> <td>○</td> <td>○</td> <td>○</td> <td></td> </tr> <tr> <td>WR</td> <td>○</td> <td>○*</td> <td>○</td> <td></td> </tr> </tbody> </table>			Zakres	HR	ROR	DR	K	Argument	R0 □ R3839	R5000 □ R8071	D0 □ D3999		Pt				1~4	MD				2	SR	○	○	○		WR	○	○*	○	
Zakres	HR	ROR	DR	K																												
Argument	R0 □ R3839	R5000 □ R8071	D0 □ D3999																													
Pt				1~4																												
MD				2																												
SR	○	○	○																													
WR	○	○*	○																													
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;">Opis</div> <ol style="list-style-type: none"> 1. FUN151 : MD2 sprawia, że PLC może w każdej chwili odbierać wiadomości wysyłane przez urządzenia peryferyjne wyposażone w interfejs komunikacyjny. 2. Protokół komunikacyjny zapisywany jest za pomocą programu DRABINKOWEGO, który musi być zgodny z połączonymi urządzeniami peryferyjnymi. 3. Jeżeli „EN” zmieni się z 0→1, status na obydwu wejściach „PAU” i „ABT” będzie 0, a port 1/2/3/4 nie był zajęty przez inne instrukcje komunikacyjne [tj. M1960 (Port1)/M1962 (Port2)/M1936 (Port3)/M1938 (Port4) = 1], to instrukcja ta niezwłocznie przejmie kontrolę nad portem 1/2/3/43 i ustawi status M1960/M1962/M1936/M1938 na 0 (co oznaczać będzie zajętość portu). W przypadku, gdy port 1/2/3/4 był już zajęty (M1960/M1962/M1936/M1938 = 0), to instrukcja przejdzie w stan oczekiwania i pozostanie w nim do momentu aż instrukcja komunikacyjna sterująca portami zakończy realizację transakcji lub zatrzyma/porzuci swoje zadanie i zwolni prawa do kontroli. 4. Jeżeli podczas realizacji transakcji status wejścia „PAU” zmieni się na 1, to instrukcja ta niezwłocznie porzuci operację odbierania (ustawi M1960/M1962/M1936/M1938 na 1). 5. W trakcie odbierania wejście „ACT” będzie aktywne 6. Jeżeli po zakończeniu realizacji transakcji pakietu danych (zakończona transmisja lub zakończone zadanie „wyslij i odbierz”) pojawi się błąd, to wyjścia „DN” i „ERR” będą aktywne przez czas jednego skanu. 7. Jeżeli po zakończeniu realizacji transakcji pakietu danych (zakończona transmisja lub zakończone zadanie „wyslij i odbierz”) nie pojawi się żaden błąd, to wyjście „DN” będzie aktywne. 																																

FUN151 CLINK	Instrukcja FUN151: MD2 (PLC pracuje jako stacja odbierająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

【 Sygnały z interfejsu 】

Przełączniki i rejestry dedykowane do portu:

Sygnały \ Port kom.	Port 1	Port 2	Port 3	Port 4
1. Wskaźnik gotowości portu	M1960	M1962	M1936	M1938
2. Wskaźnik zakończenia portu	M1961	M1963	M1937	M1939
3. Parametry komunik. portu	R4146	R4158	R4043	R4044
4. Opóźnienie TX i przerwa RX	R4147	R4159	R4045	R4048

1. Wskaźnik gotowości portu : Sygnał jest generowany w CPU.

WŁ, port jest wolny i gotowy.

WYŁ, port jest zajęty; transakcja danych w toku.

2. Wskaźnik zakończenia portu : Sygnał jest generowany w CPU.

WŁ, transakcja danych została zakończona.

3. Parametry komunikacyjne portu:

Rejestr przeznaczony jest do ustawień parametrów komunikacyjnych portu (patrz rozdział dotyczący ustawień parametrów komunikacyjnych).

4. Opóźnienie TX i przerwa RX :

Młodszy bajt określa długość przerwy dla instrukcji FUN151:MD2; jednostką jest 0,01 sekundy (domyślnie 32H). Kiedy PLC odbierze wiadomość, na którą musi odpowiedzieć (tryb „odbierz i wyślij”), a program DRABINKOWY nie będzie w stanie przetworzyć i odesłać wiadomość przez czas określony tym parametrem, to CPU zaniecha odpowiedzi i automatycznie powróci do stanu odbierania. Jeżeli FUN151:MD2 będzie w trybie „odbierz”, wartość ta będzie bez znaczenia..

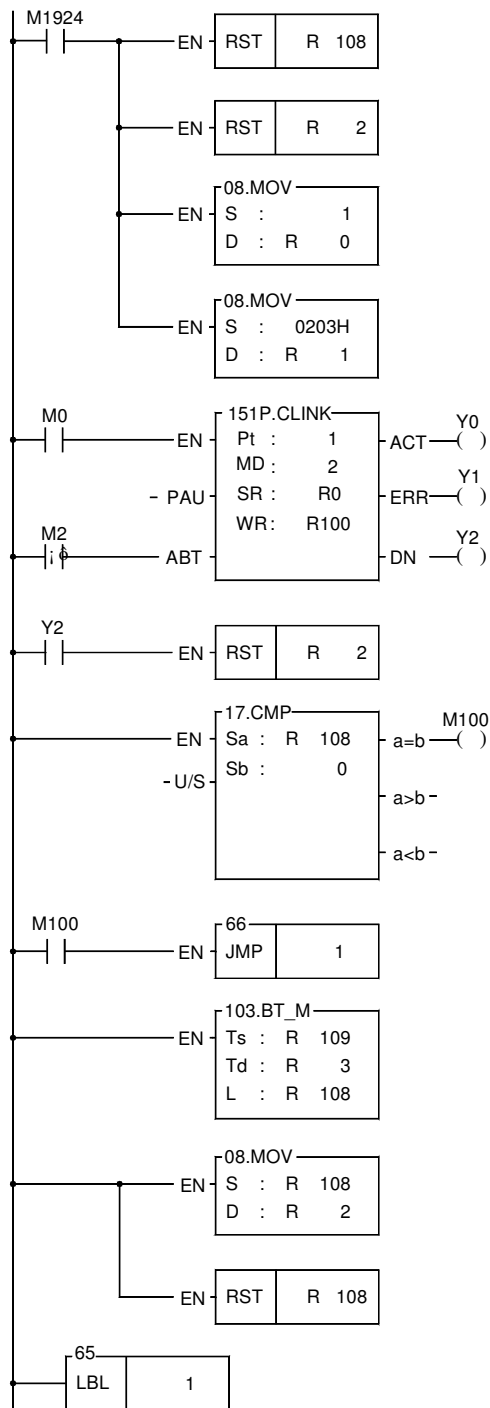
Zawartość starszego bajtu jest nieważna w tym trybie.

Uwaga 1 : Po aktywacji FUN151 : MD2, pozostanie ona przez cały czas w trybie odbierania; po aktywacji wejścia „PAU” lub „ABT” instrukcja wyjdzie z trybu odbierania i przejdzie w tryb oczekiwania.

2 : W przypadku zmiany początkowego/końcowego kodu odbioru, aby rozpocząć odbiór wiadomości należy aktywować sygnał na wejściu „PAU” lub „ABT” i zmienić status na wejściu sterującym „EN” z 0→1.

FUN151 CLINK	Instrukcja FUN151: MD2 (PLC pracuje jako stacja odbierająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

Test pętli zwrotnej na przykładzie programu (Stacja PLC odsyła odebrane dane do stacji master, z której zostały wysłane dane)



Zeruj długość odbieranych danych.

- Zeruj długość wysyłanych danych. (program ten nie jest wymagany dla trybu „odbierz”).

- Ustaw tryb roboczy:
 - Ustaw tryb „odbierz i wyślij”.

- Ustaw początkowy (02H) i końcowy (03H) kod (regularny odbiór R1=0 jest możliwy także bez podania początkowych i końcowych kodów).

- Wyzeruj długość danych po zakończeniu transmisji (instrukcja ta nie jest wymagana dla trybu „odbierz”).

- Po wybraniu trybu „wyślij i odbierz”, program wykorzysta instrukcję porównawczą do określenia czy wiadomość została odebrana; jeżeli tak, to M100 = WYŁ, a program przetworzy odebrane dane.

- Skopiuj wszystkie odebrane dane do rejestrów.

- R108 jest długością odbieranych danych.

- Po przetworzeniu odebranych danych wypełnij długość danych do odesłania w celu rozpoczęcia transmisji zwrotnej.

- Wyzeruj długość odebranych danych. (data gotowość do odbioru nowych danych).

FUN151 CLINK	Instrukcja FUN151: MD2 (PLC pracuje jako stacja odbierająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

Opis argumentu SR dla FUN151: MD2

SR : Początkowy rejestr tabeli danych zwrotnych

SR+0	Odbierz lub Odbierz i wyślij	<ul style="list-style-type: none"> • Ważny jest młodszy bajt, =00H, Odbierz bez błędu, brak odpowiedzi z PLC =01H, Odbierz bez błędu, odpowiedź z PLC =80H, Odbierz nawet z błędem, brak odpowiedzi z PLC =81H, Odbierz nawet z błędem, odpowiedź z PLC
SR+1	Początkowy / Końcowy kod odbioru	<ul style="list-style-type: none"> • Starszy bajt : Opisuje początkowy kod odbioru. Młodszy bajt : Opisuje końcowy kod odbioru
	Długość danych zwrotnych	<ul style="list-style-type: none"> • Maksymalna długość wynosi 511. Rozpocznij transmisję danych zwrotnych jeżeli długość nie jest 0
SR+4	Dane zwrotne 1	<ul style="list-style-type: none"> • Ważny jest młodszy bajt
•	Dane zwrotne 2	<ul style="list-style-type: none"> • Ważny jest młodszy bajt
•	•	
•	Dane zwrotne N	<ul style="list-style-type: none"> • Ważny jest młodszy bajt

Uwaga 1 : W przypadku wybrania trybu „odbierz”, CPU zapisuje odebrane dane w rejestrach, ustawia długość po odebraniu pakietu danych i niezwłocznie rozpoczyna odbiór kolejnego pakietu.

2 : W przypadku wybrania trybu „odbierz i wyślij”, CPU zapisuje odebrane dane w rejestrach i ustawia długość po odebraniu pakietu danych. Następnie rozpoczyna oczekiwanie na długość danych zwrotnych (która nie może być zerem) w celu rozpoczęcia transmisji (dlatego też w przypadku wybrania tego trybu, program musi wyzerować długość danych przed całkowitym zapisem danych zwrotnych w rejestrach; po zakończeniu zapisywania danych zwrotnych program może ustawić ich długość).

3 : Przed rozpoczęciem odbioru program musi zapisać początkowy i końcowy kod do rejestru początkowego/końcowego kodu (np.: SR+1=0A0DH, 0AH jest początkowym kodem, a 0DH jest końcowym kodem), aby uniknąć błędów odbioru. Protokół komunikacyjny z początkowym/końcowym kodem ułatwia identyfikację każdego pakietu i sprawia, że program komunikacyjny jest prosty i wydajny.

4 : W przypadku odbioru wiadomości bez początkowego kodu, program zapisze starszy bajt początkowego/końcowego kodu jako 0; w przypadku odbioru wiadomości bez końcowego kodu program zapisze młodszy bajt początkowego/końcowego kodu jako 0. Program także ustawi starszy bajt R4148 (czas pomiędzy wykrywaniem nowych wiadomości) w celu sprawdzenia czy pakiet danych został odebrany w całości; jednostką jest 0,001 sekundy (domyślnie 0CH, 12ms). Protokół komunikacyjny bez końcowego kodu zależy od czasu pomiędzy wykrywaniem nowych wiadomości, który służy do oceny, czy odebrany został kompletny pakiet danych (czas ten musi być większy od maksymalnego opóźnienia pomiędzy bajtami przy odpowiedzi), co umożliwia zakończenie zadania dopiero po odebraniu całego pakietu. Mówiąc ogólnie, dane są wysyłane bajt po bajcie, a co za tym idzie, pomiędzy poszczególnymi bajtami pojawiają się przerwy (większe od czasu pomiędzy wykrywaniem poszczególnych wiadomości) oznaczające, że transmisja pakietu danych przebiegła pomyślnie.

5 : Jeżeli trybie „odbierz” odbierana wiadomość nie ma końcowego kodu, to czas pomiędzy każdym wysłanym pakietem danych musi być dłuższy niż czas pomiędzy wykrywaniem nowych wiadomości. W innym wypadku odbiorca nie będzie w stanie prawidłowo rozróżnić pakietów danych.

FUN151 CLINK	Instrukcja FUN151: MD2 (PLC pracuje jako stacja odbierająca za pośrednictwem portu 1~4)	FUN151 CLINK
-----------------	--	-----------------

Opis argumentu WR dla FUN151:MD2

	Starszy bajt	Młodszy bajt	
WR+0	Kod wyniku	0	Kod wyniku =0, OK ; = inne wartości, nieprawidłowy.
WR+1	Do operacji wewnętrznych		<ul style="list-style-type: none"> • Rejestry robocze dla instrukcji CLINK
WR+2	Do operacji wewnętrznych		
WR+3	Do operacji wewnętrznych		
WR+4	Do operacji wewnętrznych		
WR+5	Do operacji wewnętrznych		<ul style="list-style-type: none"> • WR+4 : b0=1, W toku b12= wskazanie na wyjściu „ACT” b13= wskazanie na wyjściu „ERR” b14= wskazanie na wyjściu „DN”
WR+6	Do operacji wewnętrznych		
WR+7	Do operacji wewnętrznych		
WR+8	Całkowita ilość odebranych danych		<ul style="list-style-type: none"> • Całkowita ilość odbieranych bajtów (rejestr do zapisu długości odbieranych danych; zawiera początkowy i końcowy kod). • Pierwszy bajt odebranych danych (jest to początkowy kod); Starszy bajt =0.
WR+9	Dane 1		
□	Dane 2		<ul style="list-style-type: none"> • Drugi bajt odebranych danych; Starszy bajt =0.
.	Dane N		
.	Dane N		<ul style="list-style-type: none"> • N-ty bajt odebranych danych (jest to końcowy kod); Starszy bajt =0.

Uwaga : Po odebraniu pakietu danych przez GPU, program zapisze je w rejestrach i ustawi długość odebranych danych. Przed rozpoczęciem odbioru przez program DRABINKOWY, użytkownik może wyzerować długość odebranych danych. Umożliwi to wykrycie odbioru nowego pakietu danych w momencie, gdy długość odbieranych danych przestanie być równa zero. Program DRABINKOWY po odebraniu danych zeruje długość danych w rejestrze. Kontrola długości odebranych danych pozwala w ten sposób zorientować się, że program rozpoczął odbieranie nowego pakietu.

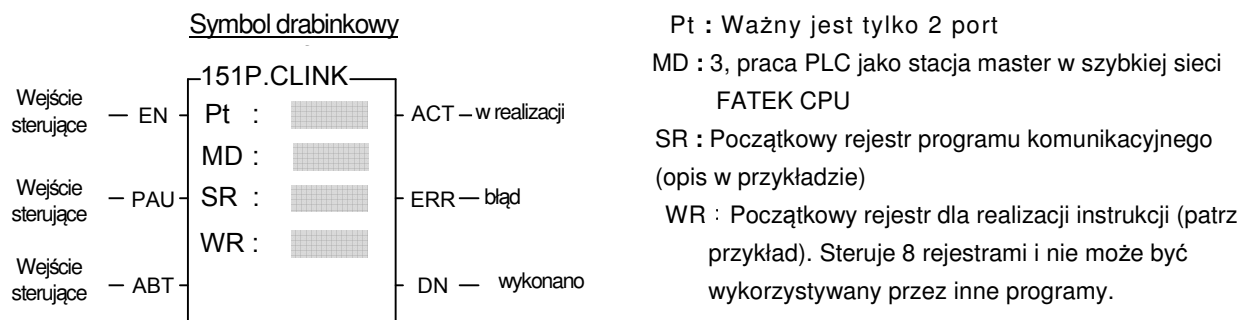
Kod wyniku : 0, transakcja zakończona pomyślnie.

- 2, błąd długości danych (wartość wynosi 0, lub pakiet transakcji wynosi ponad 511)
- A, nie można odesłać danych w zadanym czasie przerwy (dla trybu „odbierz i wyślij”).
- B, nieprawidłowa komunikacja (błąd odbioru danych)

Wskazanie na wyjściu :

- „ACT” WŁ : Odbieranie w toku
- „ERR” WŁ : Podczas transakcji poprzedniego pakietu wystąpił błąd, wyjście będzie aktywne przez czas skanu.
- „DN” WŁ : Poprzednia transakcja zakończyła się bez błędu, wyjście będzie aktywne przez czas skanu.

FUN151 CLINK	Instrukcja FUN151: MD3 (PLC pracuje jako stacja master w „Szybkiej sieci FATEK CPU” za pośrednictwem portu 2)	FUN151 CLINK
-----------------	--	-----------------



Zakres	HR	ROR	DR	K
Argument	R0 □ R3839	R5000 □ R8071	D0 □ D3999	
Pt				1~4
MD				3
SR	○	○	○	
WR	○	○*	○	

Opis:

1. FUN151 : MD3, umożliwia szybką transmisję danych pomiędzy urządzeniami Fatek PLC (czas odpowiedzi nie będzie uzależniony od czasu skanu PLC).
2. Stacja master PLC może łączyć się z maks. 254 stacjami slave PLC w celu realizacji transmisji danych za pośrednictwem portu RS-485.
3. FUN151 : MD3 wymagana jest tylko przez stację master PLC.
4. Numerem stacji master PLC musi być 1. W innym przypadku (jeżeli numer jest inny niż 1, ale stacja musi być stacją master) stację należy przypisać do rejestru R4054.
5. M1958 dla stacji slave PLC musi być WŁ (M1958 WYŁ tylko dla połączeń innych niż szybkie). Jest to bez znaczenia dla stacji master PLC.
6. Maksymalna prędkość transmisji przy szybkim łączeniu wynosi 921,6 kbps, a minimalna 38,4 kbps (z możliwością regulacji). Liczba bitów danych jest stała i wynosi 8 bitów. Dane przesyłane są w postaci kodu binarnego (który jest dwa razy szybszy od kodu ASCII), a kontrola błędów odbywa się przy wykorzystaniu CRC-16, co jest bardziej niezawodne niż metoda sumy kontrolnej.
7. Metoda szybkiej transmisji danych opiera się na koncepcji WSPÓLNEJ PAMIĘCI DANYCH, np.: gdy stacja master PLC wysła zawartość R0 do R31, to zawartość R0~R31 dla wszystkich stacji slave PLC będą identyczne jak w przypadku stacji master; gdy stacja slave PLC 2 wysła zawartość R32~R47, to zawartość R32~R47 stacji master PLC i innych stacji slave PLC będzie identyczny jak w przypadku stacji PLC nr 2, itd.
8. Kiedy PLC jest w trybie STOP, port 2 wchodzi w tryb interfejsu standardowego umożliwiającego połączenie z WinProLadder, MMI lub kontrolerem graficznym (parametr komunikacyjny ustawiany jest przez R4158).
9. Instrukcja obsługuje metodę kodowania programu lub wypełniania tabeli do sterowania przepływem danych, tj.: tym, jaki rodzaj danych zostanie wysłany z danej stacji PLC do wszystkich aktywnych stacji PLC. Definicja wymaga jedynie 7 rejestrów (5 wykorzystywanych fizycznie i 2 zarezerwowane); każde 7 rejestrów definiuje jedną transakcję komunikacyjną.
10. Jeżeli „EN” zmieni się z 0→1, a status na obu wyjściach „PAU” i „ABT” będzie 0, to instrukcja ta przejmie kontrolę nad portem 2, ustawi M1962 na „0” (kontrolowany) i niezwłocznie przejdzie do realizacji transakcji danych (przy założeniu, że port 2 nie jest kontrolowany przez żadną inną instrukcję komunikacyjną; M1962=1). W przypadku, gdy port 2 jest kontrolowany (M1962=0), instrukcja ta przejdzie w stan oczekiwania i pozostanie w nim do chwili, aż instrukcja zakończy transmisję lub zatrzyma/porzuci realizowaną operację w celu zwolnienia praw do kontroli (M1962=1); następnie instrukcja rozpocznie transakcję danych i ustawi M1962 na „0”.

FUN151 CLINK	Instrukcja FUN151: MD3 (PLC pracuje jako stacja master w „Szybkiej sieci FATEK CPU” za pośrednictwem portu 2)	FUN151 CLINK								
<p>11. Jeżeli „PAU” lub „ABT” = 1, instrukcja niezwłocznie zerwie szybkie połączenie danych (M1962 = WŁ).</p> <p>12. W trakcie szybkiego łączenia, wyjście „ACT” jest aktywne.; port 2 jest zajęty.</p> <p>13. Jeżeli przy rozpoczęciu realizacji szybkiego połączenia pojawi się błąd, to wyjście „ERR” będzie aktywne i nie dojdzie do szybkiego połączenia.</p> <p>【 Sygnały z interfejsu 】</p> <p>M1958 : Podczas szybkiej transmisji danych, stacja slave PLC musi ustawić M1958 na WŁ (nie jest to wymagane w przypadku stacji master PLC). W przypadku innego rodzaju transmisji danych, stacja slave PLC musi ustawić M1958 na WYŁ.</p> <p>M1962 : Sygnał jest generowany z CPU. WŁ oznacza, że port 2 jest dostępny. WYŁ oznacza, że port 2 jest zajęty.</p> <p>M1963 : Sygnał jest generowany z CPU. Kiedy M1967 jest WŁ (sygnał jest sterowany przez program użytkownika) i zakończyła się transakcja ostatniego pakietu danych, to CPU ustawi M1962 i M1963 na WŁ, a szybka transmisja danych zostanie zatrzymana. Przed ponownym uruchomieniem szybkiego połączenia, instrukcja musi przełączyć wejście „ABT” (przerwij transmisję) na WŁ, a następnie zmienić status wejścia sterującego „EN” z 0→1. Kiedy M1967 będzie WYŁ (sygnał jest sterowany przez program użytkownika), to po zakończeniu realizacji transakcji ostatniego pakietu danych nastąpi automatyczne rozpoczęcie ponownej realizacji transakcji pierwszego pakietu danych (M1962 i M1963 w stanie WYŁ).</p> <p>M1967 : Sterowanie jednokrotne lub cykliczne (za pomocą programu użytkownika) WŁ, jeden cykl, który zakończy się po całkowitym zakończeniu realizacji transakcji pakietu danych. WYŁ, wiele cykli; po zakończeniu realizacji transakcji ostatniego pakietu danych rozpocznie się realizacja pierwszego pakietu.</p> <p>R4054 : Stacja PLC (ale nie nr 1) pracuje jako stacja master przy realizacji szybkiego połączenia.</p> <table border="1" data-bbox="319 1339 845 1411"> <tr> <td></td> <td style="text-align: center;">Starszy bajt</td> <td style="text-align: center;">Młodszy bajt</td> <td></td> </tr> <tr> <td>R4054</td> <td style="text-align: center;">55</td> <td style="text-align: center;">Numer stacji.</td> <td>H</td> </tr> </table> <p>Jeżeli numerem stacji PLC nie jest 1, to instrukcja zapisze numer stacji (który jest przechowywany w młodszym bajcie R4055) w młodszym bajcie R4054, zapisze 55H w starszym bajcie R4054 i zmieni status na wejściu sterującym „EN” z from 0→1; mimo, że stacja PLC nie jest numerem 1, to może być ona stacją master przy realizacji szybkiego połączenia.</p> <p>R4055 : Jeżeli starszym bajtem R4055 nie jest 55H, to młodszy bajt R4055 określa numer stacji PLC. Jeżeli starszym bajtem R4055 jest 55H, to młodszy bajt R4055 określa numer stacji PLC.</p> <p>R4058 : Program lub tabela do realizacji transakcji danych musi uwzględniać przypadek, kiedy to stacja slave wyśle dane do innych stacji, a stacja master PLC bezbłędnie wykryje czy stacja slave jest aktywna. W przypadku, gdy w programie lub do realizacji transakcji danych istnieje tylko jedna stacja master wysyłająca dane do stacji slave, to stacja master PLC nie ma możliwości bezbłędnej wykrycia czy stacja slave jest aktywna.</p>				Starszy bajt	Młodszy bajt		R4054	55	Numer stacji.	H
	Starszy bajt	Młodszy bajt								
R4054	55	Numer stacji.	H							

FUN151 CLINK	Instrukcja FUN151: MD3 (PLC pracuje jako stacja master w „Szybkiej sieci FATEK CPU” za pośrednictwem portu 2)	FUN151 CLINK
-----------------	--	-----------------

W celu zastosowania programu do kontroli nieprawidłowego działania stacji master i slave PLC, użytkownik musi wykazać się umiejętnością programowania (sam program jest bardzo prosty; wywołuje on sygnał zmiany WŁ ↔ WYŁ generowany przez stację PLC wysyłającą dane. Jeżeli w zadanym okresie czasu stacja odbiorcza PLC nie wykryje sygnału zmiany WŁ ↔ WYŁ, oznacza to, że nastąpił błąd w komunikacji).

R4059 : Logowanie błędu nieprawidłowo działającej stacji slave PLC podczas realizacji szybkiego połączenia.

	Starszy bajt	Młodszy bajt	
R4059	Nieprawidłowy kod	Nieprawidłowa wartość	H

Młodszy bajt : Nieprawidłowa suma wartości

Starszy bajt : Nieprawidłowy kod

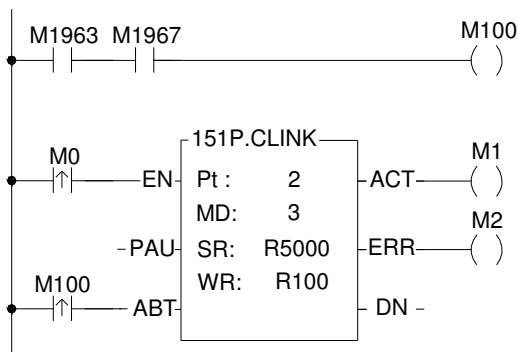
- OAH, Brak odpowiedzi ze stacji slave
- OBH, Błąd danych
- 01H, Błąd ramkowania
- 02H, Błąd przekroczenia zakresu
- 04H, Błąd parzystości
- 08H, Błąd CRC

Opis metody kontroli nieprawidłowej komunikacji jest taki sam, jak w przypadku R4058.

R4160 : Czas przerwy na porcie 2 Rx/Tx (przy realizacji szybkiego połączenia). Do ustalenia odpowiedniej nastawy system będzie się opierać na parametrze komunikacyjnym. W przypadku, gdy starszym bajtem R4160 nie będzie 56H, to użytkownik nie musi wykonywać żadnych ustawień. Jeżeli natomiast starszym bajtem R4160 jest 56H, to młodszy bajt R4160 będzie zarezerwowany do ustawień ręcznych.

R4161 : Rejestr parametrów komunikacyjnych dla realizacji szybkiego połączenia za pośrednictwem portu 2.

Przykład programu 1 (PLC nr 1 pracuje jako stacja master przy realizacji szybkiego połączenia)



- Jeżeli R5000 ~ R5199 będą rejestrami ROR, to program komunikacyjny zostanie zapisany razem z programem DRABINKOWYM.
- Jeżeli M1967 jest WŁ, to program zrealizuje jeden cykl transmisji. Przed ponowną realizacją szybkiego połączenia, program musi zakończyć aktualną operację i zrestartować M0.

FUN151 CLINK	Instrukcja FUN151: MD3 (PLC pracuje jako stacja master w „Szybkiej sieci FATEK CPU” za pośrednictwem portu 2)	FUN151 CLINK
-----------------	--	-----------------

Opis argumentu WR dla FUN151:MD3

	Starszy bajt	Młodszy bajt
WR+0	Kod wyniku	
WR+1	Do operacji wewnętrznych	
	⋮	
WR+7	DO operacji wewnętrznych	

Kod wyniku : 0 : format prawidłowy format
 2 : Błąd długości danych (Długość wynosi 0 lub jest większa od 32)
 3 : Błąd kodu komendy (Komenda nie jest 4)
 4 : Błąd typu danych (Typem danych nie jest 12 ani 13)
 5 : Błąd referencji danych

- W celu ułatwienia programowania i rozwiązywania błędów, WInProLadder obsługuje środowisko do edycji tabeli komunikacyjnej instrukcji FUN151; Aby przejść do trybu edycji tabel, należy najpierw wprowadzić całą instrukcję FUN151, a następnie przesunąć kursor na pozycję tej funkcji i nacisnąć przycisk „Z”. Użytkownik może stworzyć tu nową tabelę komunikacyjną lub wyświetlić istniejącą tabelę posługując się interfejsem przyjaznym dla użytkownika.

Tabela komunikacyjna dla FUN151:MD3

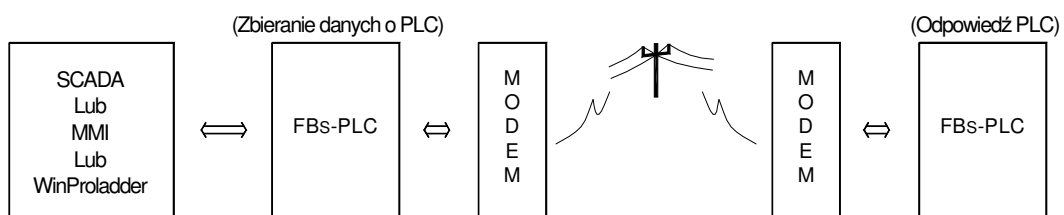
※Dla FUN151 : MD 3 ważny jest tylko port 2.

Nr sekwencji	Komenda	Nr stacji	Dane	Wszystkie stacje	Długość
0~nnn	Szybkie połączenie (=4)	Numer stacji do transmisji danych 1~254	Dane zostaną przesłane R0~R3839 D0~D3999	Dane zostaną odebrane R0~R3839 D0~D3999	Długość danych w ramach transakcji 1~32

FUN151 CLINK	Połączenie modemu z CPU przez Port 1	FUN151 CLINK
-----------------	--------------------------------------	-----------------

- PLC można połączyć z MODEMEM przez port komunikacyjny 1 oraz za pomocą sieci telekomunikacyjnej w celu wymiany informacji pomiędzy modemem a mobilnym PLC. Połączenie realizuje się w następujący sposób:
 - Automagiczne zebranie danych z punktu końcowego.
 - Automagiczne powiadamianie o sytuacjach alarmowych i nieprawidłowościach.
 - Zastosowanie dostępnego oprogramowania urządzenia kontrolnego lub MMI itp. w celu stworzenia automatycznego systemu monitorującego opartego na sieci o szerokim zakresie. Rozwiązanie takie nie wymaga specjalnego projektowania w celu zmniejszenia ryzyka i limitów czasowych.

- Konfiguracja sprzętowa i ustawienia :



Zbieranie danych PLC :

- Nie ma potrzeby zapisywania nr telefonu w CPU
- Starszy bajt R4149 = 55H (funkcja MODEMU)

Odpowiedź PLC :

- Starszy bajt R4149 = 55H (funkcja MODEMU)
- R4140 ~ R4145 ustala numer telefonu do zbierania danych PLC (dozwolona funkcja numerów wewnętrznych).

np.: jeżeli numer telefonu to 02-28082192, to

R4140=8220H, R4141=1280H i R4142=0E29H.

Jeżeli numer telefonu to : 02-28082192 wew. 100, to R4140=2A20H, R4141=2808H, R4142=A291H, R4143=AAAAH, R4144=001AH, R4145=000EH.

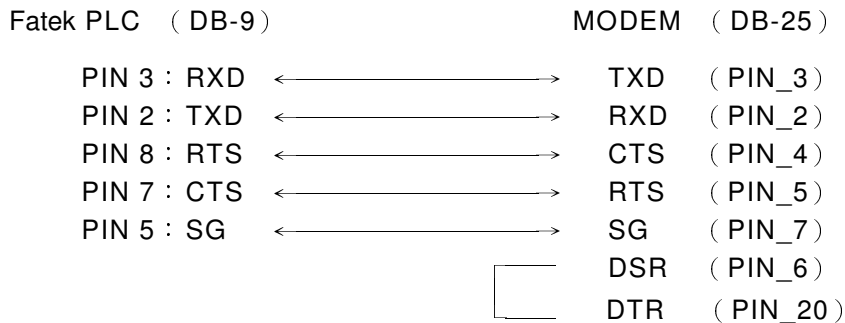
- Wyjaśnienie: R4140 ~ R4145 jest rejestrem do zapisu numerów telefonu; „E” jest ostatnim znakiem numeru; „A” jest znakiem opóźnienia wybierania (zazwyczaj wybieranie numeru wewnętrznego lub międzynarodowego jest możliwe przy wykorzystaniu opóźnienia wybierania; czas opóźnienia określany jest ustawieniem MODEMU i wynosi ok. 2 sekund). „B” for oznacza „#”, a „C” oznacza „*”.
- Program wykorzystuje instrukcję CLINK (FUN151:MD0) do zapisania lub odczytania danych z PLC do gromadzenia danych ogólnych (patrz przewodnik po instrukcji FUN151:MD0).

*** Maksymalna prędkość transmisji może osiągnąć 115200 bps (oba końcowe punkty komunikacyjne powinny być odpowiednie).

- Parametry komunikacyjne muszą być 8-bitowe, a dla większości modemów lepszym rozwiązaniem jest nieparzystość

FUN151 CLINK	Połączenie modemu z CPU przez Port 1	FUN151 CLINK
-----------------	--------------------------------------	-----------------

· Połączenia portu komunikacyjnego 1 PLC i MODEMU :



Wybieranie sygnału przez MODEM

M1959 : WYŁ, wybieranie „tonowe” ;
 WŁ, wybieranie „impulsowe”

M1964 : WYŁ→WŁ, połącz ;
 WŁ→WYŁ, rozłącz

R4163 : Młodszy bajt R4163 jest wykorzystywany do sterowania zastosowaniem instrukcji X w trakcie wybierania numeru przez MODEM.

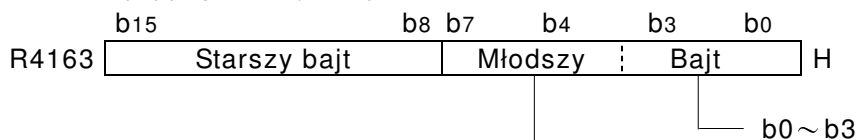
=1, nie wykrywa sygnału wybierania ani sygnału zajętości podczas wybierania numeru przez MODEM.

=2, wykrywa tylko sygnał wybierania, ale nie wykrywa sygnału zajętości podczas wybierania numeru przez MODEM.

=3, łączy bezpośrednio bez wykrywania sygnału wybierania, ale wykryje sygnał zajętości po wybraniu numeru przez MODEM.

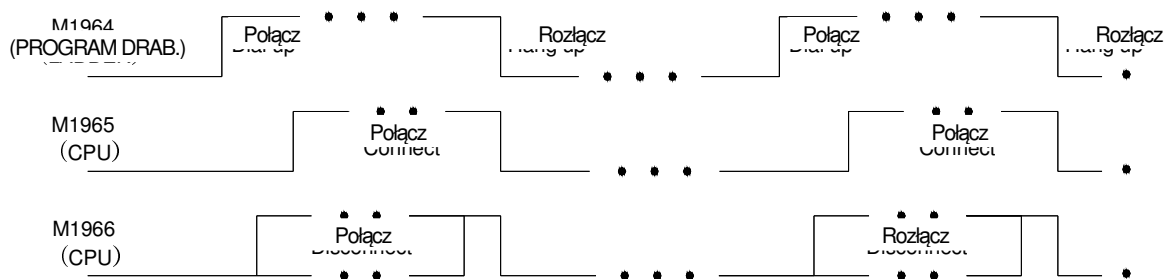
=4, wykrywa sygnał wybierania i sygnał zajętości podczas wybierania numeru przez MODEM.

Dla innych wartości, układ pracuje jak dla wartości 4; ustawienie musi być dostosowane do systemu obowiązującego w danym kraju.



B7 ~ b4=0, AT i F.....ustawienie domyślne
 B7 ~ b4=1, AT i Y.....ustawienie użytkownika

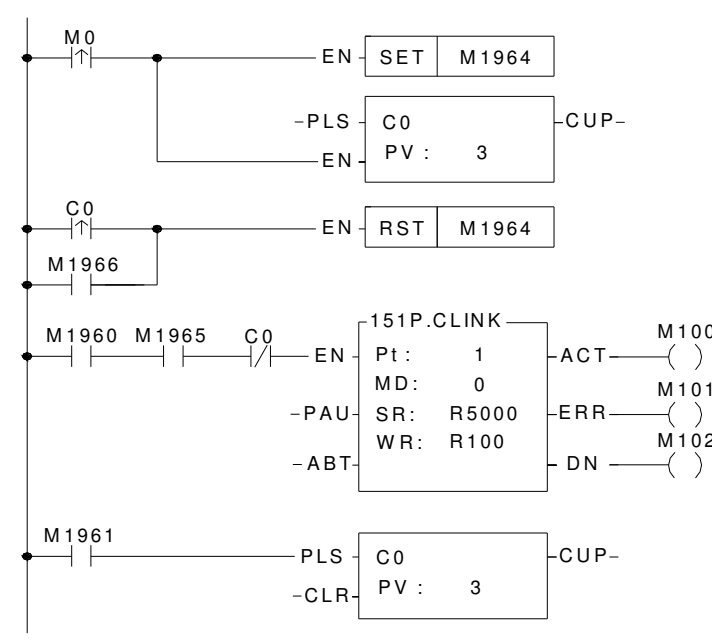
Starszy bajt R4163 wykorzystywany jest do ustawienia ilości zliczanych sygnałów w celu uruchomienia trybu automatycznej odpowiedzi przez Modem.



FUN151 CLINK	Połączenie modemu z CPU przez Port 1	FUN151 CLINK
-----------------	--------------------------------------	-----------------

- Uwaga 1 : Spośród M1965 i M1966, tylko jeden będzie aktywny; oba nie mogą być aktywne jednocześnie.
- 2 : Czas oczekiwania na połączenie wynosi 1 minutę; w przypadku, gdy połączenie jest niemożliwe nastąpi dwukrotne ponowienie próby połączenia (łącznie 3 próby). W przypadku, gdy wszystkie próby się nie powiedą, to CPU ustawi status M1966 na WŁ (połączenie nieudane).
- 3 : W przypadku niestabilnej jakości połączenia i ryzyka rozłączenia, użytkownik może wykorzystać funkcję wykrywania instrukcji CLINK w celu sterowania ponownymi próbami połączenia M1964 (czas opóźnienia musi wynosić ponad 10 sekund).
- 4 : Kiedy tryb PLC zmieni się z URUCHOM na STOP, CPU automatycznie zmieni tryb MODEMU na odbiór, w którym modem może odebrać zdalne połączenie.
- 5 : Jeżeli PLC nie wybiera numeru lub MODEM jest w trybie połączenia, to CPU automatycznie zmieni tryb MODEMU na odbiór, w którym modem może odebrać zdalne połączenie

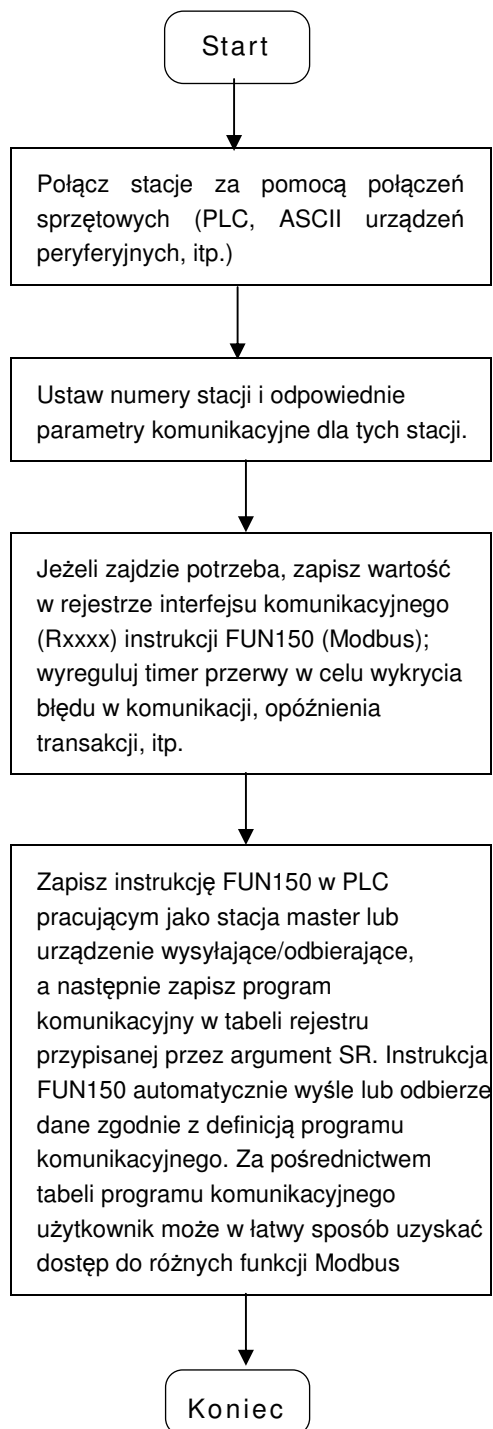
Przykład programu



- Połącz kiedy status M0 zmieni się z 0→1.
- Zerowanie ilości zliczonych transakcji.
- Rozłącz po zakończeniu transakcji lub połączenia.
- Jeżeli R5000~R5199 będą rejestrami ROR, to program komunikacyjny zostanie zapisany razem z programem DRABINKOWYM
- Zlicz po zakończeniu wszystkich transakcji

13.2 Zastosowanie instrukcji FUN150(Modbus)

13.2.1 Procedury zastosowania



- Numer stacji może być ustawiony w zakresie od 1 do 254 bez powtórzeń.

- Parametry komunikacyjne opisane są w rozdziale „Ustawienia komunikacyjne”.

13.2.2 Opis zastosowania FUN150 w programie

W niniejszym rozdziale opisane są przykłady praktycznych zastosowań instrukcji FUN150 (Modbus) w programach.

FUN150 M-BUS	Instrukcja dla stacji master MODBUS RTU/ASCII (PLC pracuje jako stacja master MODBUS RTU/ASCII za pośrednictwem portu 1~4)	FUN150 M-BUS																									
<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p style="text-align: center;"><u>Symbol drabinkowy</u></p> </div> <div style="width: 45%;"> <p>Pt : 1~4, port komunikacyjny pracujący jako stacja master MODBUS RTU</p> <p>SR : Początkowy rejestr programu komunikacyjnego</p> <p>WR : Początkowy rejestr instrukcji, Steruje 8 rejestrami i nie może być użyty w żadnym innym programie.</p> </div> </div> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>Zakres</th> <th>HR</th> <th>ROR</th> <th>DR</th> <th>K</th> </tr> </thead> <tbody> <tr> <td>Argument</td> <td>R0 □ R3839</td> <td>R5000 □ R8071</td> <td>D0 □ D3999</td> <td></td> </tr> <tr> <td>Pt</td> <td></td> <td></td> <td></td> <td>1~4</td> </tr> <tr> <td>SR</td> <td>○</td> <td>○</td> <td>○</td> <td></td> </tr> <tr> <td>WR</td> <td>○</td> <td>○*</td> <td>○</td> <td></td> </tr> </tbody> </table>			Zakres	HR	ROR	DR	K	Argument	R0 □ R3839	R5000 □ R8071	D0 □ D3999		Pt				1~4	SR	○	○	○		WR	○	○*	○	
Zakres	HR	ROR	DR	K																							
Argument	R0 □ R3839	R5000 □ R8071	D0 □ D3999																								
Pt				1~4																							
SR	○	○	○																								
WR	○	○*	○																								
<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;">Opis</div> <ol style="list-style-type: none"> 1. Instrukcja FUN150 (M-BUS) sprawia, że PLC pracuje jako stacja master MODBUS RTU/ASCII za pośrednictwem portu 1~4. W związku z tym komunikacja z inteligentnymi urządzeniami peryferyjnymi przez protokół RTU/ASCII jest bardzo łatwa. 2. Stacja master PLC może łączyć się z 247 stacjami slave za pośrednictwem interfejsu RS485. 3. Jedynie stacja master PLC wymaga zastosowania instrukcji M-BUS. 4. Instrukcja wykorzystuje metodę kodowania programu lub wypełniania tabeli do planowania sterowania przepływem danych, tj. tym, z której stacji slave pobierać dane do zapisania w stacji PLC master lub które dane pobrać ze stacji PLC master do zapisania w przypisanej stacji slave. Do zdefiniowania transakcji jednego pakietu danych potrzeba jedynie siedmiu rejestrów. 5. Jeżeli „EN” zmieni się z 0→1, status wejścia sterującego „ABT” będzie 0, a Port1/2/3/4 nie był wykorzystywany przez żadne inne instrukcje komunikacyjne [tj. M1960 (Port1) / M1962 (Port2) / M1936 (Port3) / M1938 (Port4) = 1], to instrukcja ta rozpocznie sterowanie portem 1/2/3/4 i ustawi status M1960/M1962/M1936/M1938 na 0 (co oznacza, że dany port jest zajęty), a następnie przejdzie do realizacji transakcji danych. W przypadku, gdy port 1/2/3/4 był już wykorzystywany (M1960/M1962/M1936/M1938 = 0), to instrukcja ta przejdzie w stan oczekiwania i pozostanie w nim do momentu, aż instrukcja komunikacyjna sterująca portami zakończy transakcję danych lub zatrzyma/porzuci wykonywane zadanie w celu zwolnienia portów (M1960/M1962/M1936/M1938 = 1). Opisywana instrukcja ustawi wówczas status M1960/M1962/M1936/M1938 na 0 i niezwłocznie przejdzie do realizacji transakcji danych. 6. Jeżeli podczas realizacji transakcji status wejścia sterującego „ABT” zmieni się na 1, to instrukcja ta porzuci realizowaną transakcję i zwolni prawo do sterowania portem (M1960/M1962/M1936/M1938 = 1). Po kolejnym przejściu praw do sterowania instrukcja rozpocznie realizację następnego pakietu transakcji. 7. Jeżeli dla protokołu MODBUS RTU „A/R” = 0, to dla protokołu MODBUS ASCII „A/R” = 1 8. Podczas realizacji transakcji danych, wyjście „ACT” będzie aktywne. 9. Jeżeli po zakończeniu realizacji transakcji pakietu danych wystąpi błąd, to wyjścia „DN” i „ERR” będą aktywne. 10. Jeżeli po zakończeniu realizacji transakcji pakietu danych nie wystąpi błąd, to wyjście „DN” będzie aktywne. <p>Uwaga : Tryb MODBUS ASCII obsługiwany jest przez OS w wersji 4.24 lub późniejszych</p>																											

FUN150 M-BUS	Instrukcja dla stacji master MODBUS RTU/ASCII (PLC pracuje jako stacja master MODBUS RTU/ASCII za pośrednictwem portu 1~4)	FUN150 M-BUS
-----------------	---	-----------------

【 Sygnały z interfejsu 】

Przełączniki i rejestry dedykowane do konkretnych portów :

Port kom. Sygnały	Port 1	Port 2	Port 3	Port 4
1. Wskaźnik gotowości portu	M1960	M1962	M1936	M1938
2. Wskaźnik zakończenia portu	M1961	M1963	M1937	M1939
3. Parametry komunik. portu	R4146	R4158	R4043	R4044
4. Opóźnienie TX i przerwa RX	R4147	R4159	R4045	R4048
5. Ustawienia czasu przerwy RX	D4043			
6. Wzbudzenie zbocza	D4044			

1. Wskaźnik gotowości portu : Sygnał jest generowany w CPU.

WŁ, port jest wolny i gotowy.

WYŁ, port jest zajęty; transakcja danych w toku.

2. Wskaźnik zakończenia portu : Sygnał jest generowany w CPU.

Kiedy program komunikacyjny zakończy realizację transakcji ostatniego pakietu danych, to wskaźnik ten będzie aktywny przez czas jednego skanowania (dla kolejnej transakcji danych).

Po zakończeniu realizacji ostatniego pakietu danych przez program komunikacyjny, sygnał ten będzie wciąż aktywny (dla transmisji jednego pakietu danych).

3. Parametry komunikacyjne portu :

Ustawienia parametrów komunikacyjnych portu (patrz rozdział o ustawieniach parametrów komunikacyjnych).

4. Opóźnienie TX i przerwa RX:

Zawartość młodszego bajtu definiuje czas przerwy w odbiorze instrukcji M-BUS; jednostką jest 0,01 sekundy (wartość domyślna to 50, co oznacza 0,5 sekundy).

Kiedy stacja master PLC wyśle komendę czytaj / zapisz do stacji slave, a stacja slave nie wyśle odpowiedzi w przeciągu zadanego czasu, to oznaczać to będzie, że pojawiła się przerwa w komunikacji. W przypadku łączenia wielopunktowego, należy prawidłowo ustawić tę wartość (musi być ona większa niż czas 1 skanowania stacji slave o najdłuższym czasie skanowania) w celu skrócenia czasu reakcji wśród aktywnych stacji w przypadku wielu nieaktywnych stacji slave (przerwy).

Zawartość starszego bajtu definiuje czas przerwy w transmisji pomiędzy dwoma pakietami danych przy zastosowaniu instrukcji M-BUS; jednostką jest 0,01 sekundy (wartość domyślna to 0).

Przy połączeniu punkt-do-punktu, ta wartość może być ustawiona na „0” w celu skrócenia czasu komunikacji i zwiększenia jej wydajności. W przypadku łączenia wielopunktowego oraz czasu skanu stacji master o wiele dłuższego niż stacji slave, ta wartość może być także ustawiona na „0” w celu skrócenia czasu komunikacji i zwiększenia jej wydajności. Jeżeli natomiast czasy skanu stacji master i slave są zbliżone, to należy tak ustawić wartość (na dłuższa od 1 czasu skanu stacji slave o najdłuższym czasie skanu), aby osiągnąć najlepszą i bezbłędną jakość komunikacji.

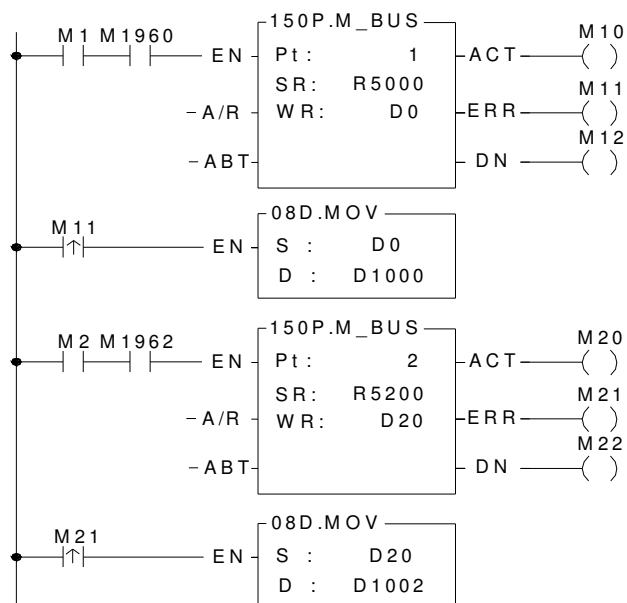
FUN150 M-BUS	Instrukcja dla stacji master MODBUS RTU/ASCII (PLC pracuje jako stacja master MODBUS RTU/ASCII za pośrednictwem portu 1~4)	FUN150 M-BUS
-----------------	---	-----------------

5. Ustawienia czasu przerwy RX: Szczegóły w niniejszym rozdziale na stronie 13-4~13-5

6. Wzbudzenie zbocza: Szczegóły w niniejszym rozdziale na stronie 13-5~13-6

※ Jeżeli komunikat zostanie odebrany bez kodu końcowego i M1956=1, to do określenia tego czy dane zostały odebrane wykorzystywany jest starszy bajt R4148 czasu przerwy; jednostką jest 0,001 sekundy (domyślnie 0CH, 12ms).

Przykład programu (Automatyczna transmisja cykliczna)



- Przed rozpoczęciem programowania skonfiguruj R5000 ~ R5399 jako rejestr tylko do odczytu (ROR). Dzięki temu po zapisaniu programu program drabinkowy będzie automatycznie zawierać program komunikacyjny.
- W przypadku pojawienia się błędu komunikacyjnego, zapisz komunikat o błędzie w D1000 i D1001 w celu ułatwienia analizy błędu lub logowania.

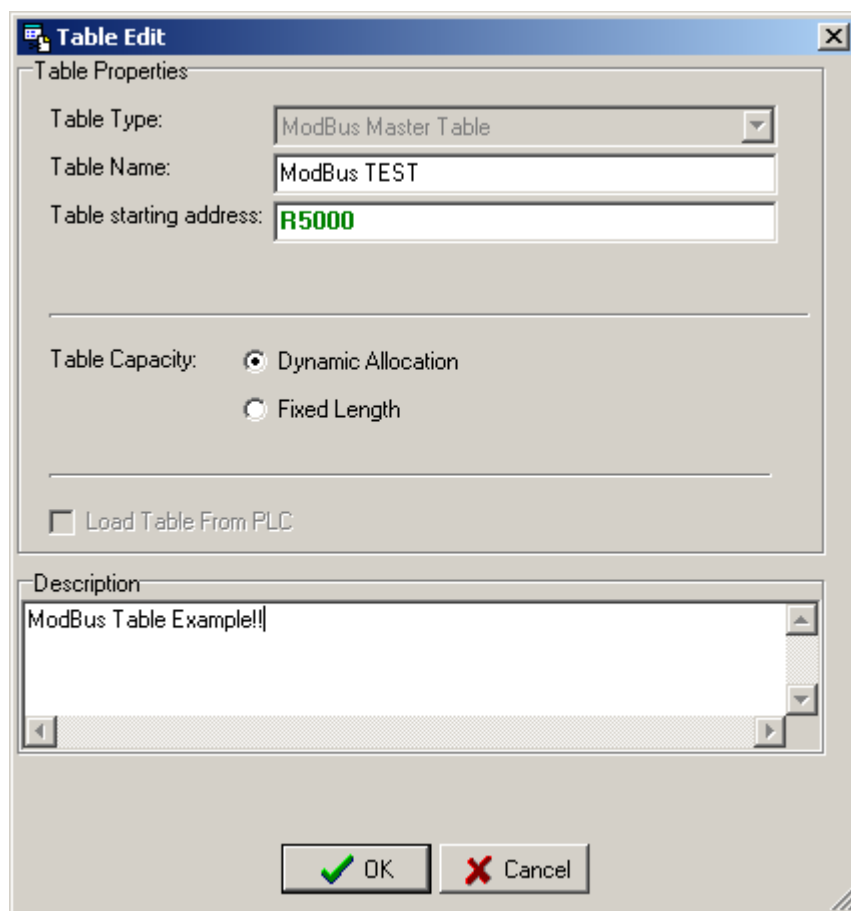
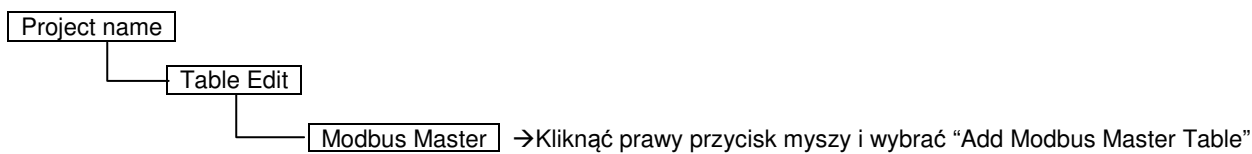
FUN150 M-BUS	Instrukcja dla stacji master MODBUS RTU/ASCII (PLC pracuje jako stacja master MODBUS RTU/ASCII za pośrednictwem portu 1~4)	FUN150 M-BUS
-----------------	---	-----------------

Przykład programu

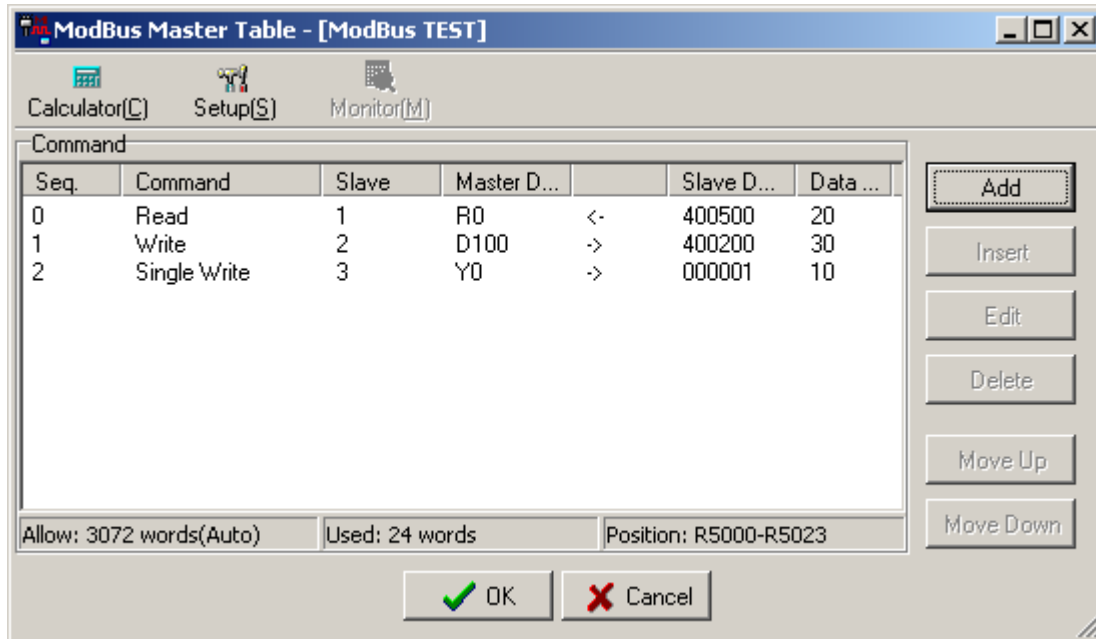
1. Jeżeli „EN” zmieni się z 0→1, a port 1 nie jest zajęty przez żadną inną instrukcję komunikacyjną (M1960 = WŁ), to instrukcja M-BUS rozpocznie realizację transakcji danych. W trakcie realizacji transakcji danych M1960 = WYŁ. Po zakończeniu realizacji transakcji M1960 = WŁ. Zmiana M1960 z WYŁ ↔ WŁ (zmiana statusu wejścia sterującego „EN” M-BUD z 0→1) może automatycznie rozpocząć transakcję kolejnego pakietu danych (po zakończeniu ostatniej transakcji program powróci do pierwszego pakietu w celu realizacji automatycznej transmisji cyklicznej).
2. Jeżeli „EN” zmieni się z 0→1, a port 2 nie jest zajęty przez żadną inną instrukcję komunikacyjną (M1962 = WŁ), to instrukcja M-BUS rozpocznie realizację transakcji danych. W trakcie realizacji transakcji danych M1962 = WYŁ. Po zakończeniu realizacji transakcji M1962 = WŁ. Zmiana M1962 z WYŁ ↔ WŁ (zmiana statusu wejścia sterującego „EN” M-BUD z 0→1) może automatycznie rozpocząć transakcję kolejnego pakietu danych (po zakończeniu ostatniej transakcji program powróci do pierwszego pakietu w celu realizacji automatycznej transmisji cyklicznej).

Edycja tabeli komunikacyjnej za pomocą WinProladder

W oknie projektu wybrać opcję „Modbus Master”::



- Table Type : Parametr zostanie ustawiony na „Modbus Master Table ”.
- Table Name : Nazwa tabeli dla edycji lub debugowania.
- Table Starting address : Adres początkowego rejestru tabeli komunikacyjnej.



FUN150 M-BUS	Instrukcja dla stacji master MODBUS RTU/ASCII (PLC pracuje jako stacja master MODBUS RTU/ASCII za pośrednictwem portu 1~4)	FUN150 M-BUS
-----------------	---	-----------------

Początkowy rejestr programu komunikacyjnego instrukcji M-BUS

SR Początkowy rejestr programu komunikacyjnego instrukcji M-BUS

SR+0	A5h	50h	<ul style="list-style-type: none"> • A550h, oznacza ważny program M-BUS
SR+1	07h	Całkowita liczba transakcji	<ul style="list-style-type: none"> • Młodszy bajt: całkowita liczba transakcji; opis jednej transakcji wymaga 7 rejestrów.
SR+2	Nr stacji slave do realizacji transakcji		<ul style="list-style-type: none"> • Ważny jest młodszy bajt, 0~247 (0 master oznacza, że PLC wysyła dane do wszystkich stacji slave PLC, a stacja slave PLC nie odpowiada).
SR+3	Kod komendy		<ul style="list-style-type: none"> • Ważny jest młodszy bajt ; =1, „Odczytaj dane ze stacji slave” =2, „Zapisz wiele danych do stacji slave” =3, „Zapisz pojedyncze dane do stacji slave”
SR+4	Długość danych transakcji		<ul style="list-style-type: none"> • Ważny jest młodszy bajt; zakres 1~125 (Reg.) lub 1~255 (zmienna dyskretna).
SR+5	Typ danych master PLC		<ul style="list-style-type: none"> • Ważny jest młodszy bajt; zakres 1~3 lub 12~13; określa typ danych dla stacji PLC (patrz następna strona).
SR+6	Początkowa referencja master PLC		<ul style="list-style-type: none"> • Ważne jest słowo; określa początkowy adres danych (master).
SR+7	Typ danych stacji slave		<ul style="list-style-type: none"> • Ważny jest młodszy bajt; zakres 0 lub 4; określa typ danych dla stacji slave (patrz następna strona).
SR+8	Początkowa referencja stacji slave		<ul style="list-style-type: none"> • Ważne jest słowo; określa początkowy adres danych (slave).
SR+9	Nr stacji slave do realizacji transakcji		<div style="display: flex; align-items: center;"> <div style="font-size: 3em; margin-right: 10px;">}</div> <div>Opis drugiego pakietu transakcji</div> </div>
SR+10	Kod komendy		
SR+11	Długość danych transakcji		
SR+12	Typ danych master PLC		
SR+13	Początkowa referencja master PLC		
SR+14	Typ danych stacji slave		
SR+15	Początkowa referencja stacji slave		
• • •			
SR+2+ n×7	Zarezerwowany		<ul style="list-style-type: none"> • N oznacza całkowitą liczbę transakcji.

FUN150 M-BUS	Instrukcja dla stacji master MODBUS RTU/ASCII (PLC pracuje jako stacja master MODBUS RTU/ASCII za pośrednictwem portu 1~4)	FUN150 M-BUS
-----------------	---	-----------------

- Kod danych typ i numer referencyjny stacji MASTER (FATEK PLC)

Kod danych	Typ danych	Numer referencyjny
1	Y (Przełącznik wyjściowy)	0 ~ 255
2	M(Wewnętrzny przełącznik M)	0 ~ 1911
3	S (Przełącznik krokowy)	0 ~ 999
12	R (Rejestr danych Rxxxx)	0 ~ 3839
13	D (Rejestr danych Dxxxx)	0 ~ 3999

- Kod danych typ i numer referencyjny stacji slave (slave MODBUS)

Kod danych	Typ danych	Numer referencyjny
0	Wyjście dyskretne	1 ~ 65535
4	Rejestr utrzymujący	1 ~ 65535
1	Wejście dyskretne (wersja OS 4.22 ↑)	1 ~ 65535
3	Rejestr wejściowy (wersja OS 4.22 ↑)	1 ~ 65535

- WR : Początkowy rejestr dla instrukcji M-BUS (FUN150)

Starszy bajt Młodszy bajt

WR+0	Kod wyniku	Nr transakcji
WR+1	Numer stacji	Kod komendy
WR+2	Do operacji wewnętrznych	
WR+3	Do operacji wewnętrznych	
WR+4	Do operacji wewnętrznych	
WR+5	Do operacji wewnętrznych	
WR+6	Do operacji wewnętrznych	
WR+7	Do operacji wewnętrznych	

- Kod wyniku określa wynik transakcji; 0 = normalny, inna wartość = nienormalny.
- Nr transakcji określa, która transakcja jest aktualnie realizowana (rozpoczyna się od 0)
- Nr stacji określa numer stacji realizującej transakcję.
 Kod komendy =01H, odczytaj status 0xxxxx ze stacji slave
 =02H, odczytaj status 2xxxxx ze stacji slave
 =03H, odczytaj dane 3xxxxx ze stacji slave
 =04H, odczytaj dane 4xxxxx ze stacji slave
 =05, wymuś działanie pojedynczej cewki dla stacji slave
 =06H, ustaw pojedynczy rejestr w stacji slave
 =0FH, wymuś działanie wielu cewek dla stacji
 =10H, ustaw wiele rejestrów w stacji slave
- WR+4 B0=1, Port jest zajęty, a instrukcja oczekuje na pozyskanie praw do realizacji transakcji danych
 B4=1, ta instrukcja nie jest realizowana po raz pierwszy.
 B12, wskazanie na wyjściu „ACT”
 B13, wskazanie na wyjściu „ERR”
 B14, wskazanie na wyjściu „DN”

Kod wyniku : 0, Transakcja .

- 2, Błąd długości danych (limit dla długości równej 0 lub przekraczającej wartość graniczną).
- 3, Błąd kodu komendy (Kod komendy równy jest 0 lub większy od 3)
- 4, Błąd typu danych
- 5, Błąd numeru referencyjnego
- 6, Nieprawidłowy typ danych (np: 1~3 dla stacji master i 12~13 dla stacji slave).
- 7, Błąd portu (port inny niż 1~4)
- 8, Nieprawidłowa tabela komunikacyjna

FUN150 M-BUS	Instrukcja dla stacji master MODBUS RTU/ASCII (PLC pracuje jako stacja master MODBUS RTU/ASCII za pośrednictwem portu 1~4)	FUN150 M-BUS
-----------------	---	-----------------

A, Brak odpowiedzi ze stacji slave (błąd przerwy).

B, Błąd komunikacji (odebrano dane o błędzie lub nieprawidłowa odpowiedź).

- W celu ułatwienia programowania i rozwiązywania błędów, WinProladder obsługuje środowisko do edycji tabeli komunikacyjnej instrukcji FUN150; Aby przejść do trybu edycji tabel należy najpierw wprowadzić całą instrukcję FUN150, a następnie przesunąć kursor na pozycję tej funkcji i nacisnąć przycisk „Z”. Użytkownik może stworzyć tu nową tabelę komunikacyjną lub wyświetlić istniejącą tabelę posługując się interfejsem przyjaznym dla użytkownika.

Tabela komunikacyjna M-BUS

Nr sekwencji	Komenda	Slave	Dane master	Dane slave	Długość
0 ~ nnn	Czytaj (=1) Zapisz (=2) Zapisz pojed. (=3)	Numer stacji slave PLC realizującej transakcję. Numer stacji=0, Stacja master PLC wysyła dane do wszystkich stacji slave PLC. Stacje slave nie odpowiadają. Numer stacji=N, Numer stacji slave PLC realizującej transakcję ze stacją master PLC. N=1 ~ 254	Typ danych dla stacji MASTER podczas transakcji Y0 ~ Y255 M0 ~ M1911 S0 ~ S999 R0 ~ R3839 D0 ~ D3999	Typ danych dla stacji SLAVE podczas transakcji 000001 ~ 065535 (czytaj/zapisz) 400001 ~ 465535 (czytaj/zapisz) 100001 ~ 165535 (czytaj) 300001 ~ 365535 (czytaj)	Długość danych dla transakcji przy rejestrze w zakresie 1 ~ 125 przy zmiennej dyskretnej w zakresie 1 ~ 255

※ WinProladder umożliwia łatwą edycję tabeli dla stacji MASTER M-BUS :

Nr sekwencji	Komenda	Slave	Dane master	Dane slave	Długość danych	
0	Czytaj	1 ~ 247	Y0 ~ Y255	←	000001 ~ 065535	1 ~ 255
			M0 ~ M1911	←	000001 ~ 065535	1 ~ 255
			S0 ~ S999	←	000001 ~ 065535	1 ~ 255
			Y0 ~ Y255	←	100001 ~ 165535	1 ~ 255
			M0 ~ M1911	←	100001 ~ 165535	1 ~ 255
			S0 ~ S999	←	100001 ~ 165535	1 ~ 255
			R0 ~ R3839	←	400001 ~ 465535	1 ~ 125
			D0 ~ D3999	←	400001 ~ 465535	1 ~ 125
			R0 ~ R3839	←	300001 ~ 365535	1 ~ 125
			D0 ~ D3999	←	300001 ~ 365535	1 ~ 125
1	Zapisz	0 ~ 247	Y0 ~ Y255	→	000001 ~ 065535	1 ~ 255
			M0 ~ M1911	→	000001 ~ 065535	1 ~ 255
			S0 ~ S999	→	000001 ~ 065535	1 ~ 255
			R0 ~ R3839	→	400001 ~ 465535	1 ~ 125
			D0 ~ D3999	→	400001 ~ 465535	1 ~ 125
2						
.						
.						

FBs-PLC LINK

Modbus Slave	Mapowanie adresów pomiędzy MODBUS i FATEK (Port 1~4 pracuje jako stacja slave za pośrednictwem protokołu MODBUS)	Modbus Slave
-----------------	---	-----------------

- FBs-PLC może wykorzystać FUN150 do pracy jako stacja master MODBUS lub slave (poprzez Port1 ~ Port4, przy porcie 0 związanym z protokołem komunikacyjnym FATEK) oraz łączyć się z inteligentnymi urządzeniami peryferyjnymi FBs-PLC.
- Poniżej lista zmapowanych adresów dla protokołów MODBUS i FATEK:

Reguła mapowania

MODBUS		FATEK
Kod 5	0XXXX	Elementy dyskretne Ynnn · Xnnn · Mnnnn · Snnn · Tnnn · Cnnn
	4XXXX	Rejestry danych Rnnnn · Dnnnn · Tnnn · Cnnn
Kod 6	00XXXX	Elementy dyskretne Ynnn · Xnnn · Mnnnn · Snnn · Tnnn · Cnnn
	40XXXX	Rejestry danych Rnnnn · Dnnnn · Tnnn · Cnnn

Dostępny zakres (kod 5)

MODBUS	FATEK	Opis
00001 ~ 00256	Y0 ~ Y255	Dyskretne wyjście
01001 ~ 01256	X0 ~ X255	Dyskretne wejście
02001 ~ 04002	M0 ~ M2001	Dyskretny przekaźnik M
06001 ~ 07000	S0 ~ S999	Dyskretny przekaźnik S
09001 ~ 09256	T0 ~ T255	Status T0 ~ T255
09501 ~ 09756	C0 ~ C255	Status C0 ~ C255
40001 ~ 44168	R0 ~ R4167	Rejestr utrzymujący
45001 ~ 45999	R5000 ~ R5998	Rejestr utrzymujący lub ROR
46001 ~ 48999	D0 ~ D2998	Rejestr danych
49001 ~ 49256	T0 ~ T255	Aktualna wartość T0 ~ T255
49501 ~ 49700	C0 ~ C199	Aktualna wartość C0 ~ C199 (16-bitowa)
49701 ~ 49812	C200 ~ C255	Aktualna wartość C200 ~ C255 (32-bitowa)

Modbus Slave	Mapowanie adresów pomiędzy MODBUS i FATEK (Port 1~4 pracuje jako stacja slave za pośrednictwem protokołu MODBUS)	Modbus Slave
-----------------	---	-----------------

Dostępny zakres (Kod 6)

MODBUS	FATEK	Opis
000001 ~ 000256	Y0 ~ Y255	Dyskretne wyjście
001001 ~ 001256	X0 ~ X255	Dyskretne wejście
002001 ~ 004002	M0 ~ M2001	Dyskretny przekaźnik M
006001 ~ 007000	S0 ~ S999	Dyskretny przekaźnik S
009001 ~ 009256	T0 ~ T255	Status T0 ~ T255
009501 ~ 009756	C0 ~ C255	Status C0 ~ C255
400001 ~ 404168	R0 ~ R4167	Rejestr utrzymujący
405001 ~ 405999	R5000 ~ R5998	Rejestr utrzymujący lub ROR
406001 ~ 408999	D0 ~ D2998	Rejestr danych
409001 ~ 409256	T0 ~ T255	Aktualna wartość T0 ~ T255
409501 ~ 409700	C0 ~ C199	Aktualna wartość C0 ~ C199 (16-bitowa)
409701 ~ 409812	C200 ~ C255	Aktualna wartość C200 ~ C255 (32-bitowa)

※※ Dostępny zakres specjalnego rejestru i przekaźnika

MODBUS	FATEK	Opis
02001 ~ 03912	M0 ~ M1911	Wewnętrzny przekaźnik ogólnego przeznaczenia
03913 ~ 04002	M1912 ~ M2001	Specjalny wewnętrzny przekaźnik
40001 ~ 43840	R0 ~ R3839	Rejestr ogólnego przeznaczenia
43841 ~ 43904	R3840 ~ R3903	Wejściowy rejestr analogowy lub cyfrowy
43905 ~ 43968	R3904 ~ R3967	Wyjściowy rejestr analogowy lub cyfrowy
43969 ~ 44168	R3968 ~ R4167	Specjalny rejestr

FBs-PLC LINK

Modbus Slave	Port 1~4 symuluje urządzenie slave MODBUS	Modbus Slave
<p>Dodaj mapowanie nowego adresu dla protokołu kom. Stacji slave MODBUS: w przypadku wyjścia poza zakres dostępu, PLC wyśle komunikat o błędzie komunikacyjnym</p>		
Nr rejestru.	Wartość	Opis
R3968	=A55AH	Mapowanie nowego adresu dla protokołu komunikacyjnego slave MODBUS (opis poniżej)
	= Inne	Mapowanie istniejącego adresu dla protokołu kom. Slave MODBUS
R3969	0 ~ 65535	<ul style="list-style-type: none"> . Przypisz początkowy adres do dyskretnego wyjścia MODBUS . 0 ~ 65535 : oznacza dyskretne wyjście 000001 ~ 065536 . Zastosuj do kodu funkcji 01, 05, 15 protokołu MODBUS
R3970	0 ~ 2001	<ul style="list-style-type: none"> . Przypisz początkowy adres wewnętrznego przekaźnika FATEK . 0 ~ 2001 : oznacza wewnętrzny przekaźnik M0 ~ M2001 . Zastosuj do kodu funkcji 01, 05, 15 protokołu MODBUS
R3971	1 ~ 2001	<ul style="list-style-type: none"> . Przypisz zakres dostępu dla dyskretnego wyjścia (MODBUS) i wewnętrznego przekaźnika (FATEK) . 1 ~ 2001 : oznacza zakres dostępu pomiędzy punktami 1 ~ 2001 . Grupa R3969 ~ R3971 do mapowania dyskretnego wyjścia (MODBUS) i wewnętrznego przekaźnika (FATEK) w celu uzyskania dostępu (R3968 powinien być A55AH)
R3972	0 ~ 65535	<ul style="list-style-type: none"> . Przypisz początkowy adres dyskretnego wejścia MODBUS . 0 ~ 65535 : oznacza wejście dyskretne 100001 ~ 165536 . Zastosuj do kodu funkcji 02 protokołu MODBUS
R3973	0 ~ 2001	<ul style="list-style-type: none"> . Przypisz początkowy adres wewnętrznego przekaźnika FATEK . 0 ~ 2001 : oznacza wewnętrzny przekaźnik M0 ~ M2001 . Zastosuj do kodu funkcji 02 protokołu MODBUS
R3974	1 ~ 2001	<ul style="list-style-type: none"> . Przypisz zakres dostępu dla dyskretnego wejścia (MODBUS) i wewnętrznego przekaźnika (FATEK) . 1 ~ 2001 : oznacza zakres dostępu pomiędzy punktami 1 ~ 2001 . Grupa R3972 ~ R3974 do mapowania dyskretnego wejścia (MODBUS) i wewnętrznego przekaźnika (FATEK) w celu uzyskania dostępu (R3968 nie ma znaczenia)
R3975	0 ~ 65535	<ul style="list-style-type: none"> . Przypisz początkowy adres wejścia rejestru MODBUS . 0 ~ 65535 : oznacza wejście rejestru 300001 ~ 365536 . Zastosuj do kodu funkcji 02 04 protokołu MODBUS
R3976	0 ~ 3839	<ul style="list-style-type: none"> . Przypisz początkowy adres rejestru R FATEK . 0 ~ 3839 : register oznacza rejestr R R0 ~ R3839 . Zastosuj do kodu funkcji 02 04 protokołu MODBUS

Modbus Slave	Port 1~4 symuluje urządzenie slave MODBUS	Modbus Slave
--------------	---	--------------

R3977	1 ~ 3840	<ul style="list-style-type: none"> . Przypisz zakres dostępu dla wejścia rejestrowego (MODBUS) i rejestru R (FATEK) . 1 ~ 3840 : oznacza zakres dostępu pomiędzy słowami 1 ~ 3840 . Grupa R3975 ~ R3977 do mapowania wejścia rejestrowego (MODBUS) register rejestru R (FATEK) w celu uzyskania dostępu (R3968 nie ma znaczenia)
R3978	0 ~ 65535	<ul style="list-style-type: none"> . Przypisz początkowy adres rejestru utrzymującego MODBUS . 0 ~ 65535 : oznacza rejestr utrzymujący 400001 ~ 465536 . Zastosuj do kodu funkcji 03, 06, 16 protokołu MODBUS
R3979	0 ~ 3839	<ul style="list-style-type: none"> . Przypisz początkowy adres rejestru R FATEK . 0 ~ 3839 : oznacza rejestr R R0 ~ R3839 . Zastosuj do kodu funkcji 03, 06, 16 protokołu MODBUS
R3980	1 ~ 3840	<ul style="list-style-type: none"> . Przypisz zakres dostępu dla rejestru utrzymującego (Modbus) i rejestru R (FATEK) . 1 ~ 3840 : oznacza zakres dostępu pomiędzy słowami 1 ~ 3840 . Grupa R3978 ~ R3980 do mapowania rejestru utrzymującego (MODBUS) i rejestru R (FATEK) w celu uzyskania dostępu (R3968 powinien być A55AH)

Na przykład. R3968=A55AH, oznacza mapowanie nowego adresu protokołu kom. stacji slave MODBUS

R3969=0, R3970=1000, R3971=100: Mapowanie 000001 ~ 000100 (MODBUS)

M1000~M1099 (FATEK)

R3972=10, R3973=1100, R3974=50: Mapowanie 100011 ~ 100060 (MODBUS)

M1100 ~ M1149 (FATEK)

R3975=50, R3976=1000, R3977=10: Mapowanie 300051 ~ 300060 (MODBUS)

R1000 ~ R1009 (FATEK)

R3978=100, R3979=2000, R3980=200: Mapowanie 400101 ~ 400300 (MODBUS)

R2000 ~ R2199 (FATEK)

FBs-PLC LINK

Modbus Slave	Konfiguracja portu 1~4 do pracy jako protokół MODBUS	Modbus Slave																		
<p>● Port 1~4 obsługuje protokół kom. MODBUS RTU/ASCII (Slave)</p> <p>. Metoda 1 (Wszystkie wersje OS FBs PLC mnoga obsługiwać tą metodę)</p> <p>R4047 : Starszy bajt = 55H · port skonfiguruj port kom. protokołu MODBUS RTU = Inne wartości · Port 1~4 nie obsługuje protokołu MODBUS RTU (FATEK jako domyślny)</p> <p>Młodszy bajt : Przypisz port do protokołu MODBUS RTU</p> <p>Format jak poniżej:</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Starszy bajt</th> <th colspan="8" style="text-align: center;">Młodszy bajt</th> </tr> <tr> <td style="text-align: center; border: 1px solid black;">55</td> <td style="text-align: center; border: 1px solid black;">b7</td> <td style="text-align: center; border: 1px solid black;">b6</td> <td style="text-align: center; border: 1px solid black;">b5</td> <td style="text-align: center; border: 1px solid black;">b4</td> <td style="text-align: center; border: 1px solid black;">b3</td> <td style="text-align: center; border: 1px solid black;">b2</td> <td style="text-align: center; border: 1px solid black;">b1</td> <td style="text-align: center; border: 1px solid black;">b0</td> </tr> </thead> </table> <p>b0, Zarezerwowany;</p> <p>b1=0, Port 1 pracuje jako protokół FATEK =1, Port 1 pracuje jako protokół MODBUS RTU</p> <p>b2=0, Port 2 pracuje jako protokół FATEK =1, Port 2 pracuje jako protokół MODBUS RTU</p> <p>b3=0, Port 3 pracuje jako protokół FATEK =1, Port 3 pracuje jako protokół MODBUS RTU</p> <p>b4=0, Port 4 pracuje jako protokół FATEK =1, Port 4 pracuje jako protokół MODBUS RTU</p> <p>. . .</p> <p>b7~b5, Zarezerwowany</p> <p>※ Umożliwia przypisanie wielu portów do protokołu MODBUS RTU, gdzie status odpowiedniego bitu musi być 1.</p> <p>Na przykład: R4047=5502H, Przypisz Port 1 jako protokół MODBUS RTU; R4047=5504H, Przypisz Port 2 jako protokół MODBUS RTU; R4047=5506H, Przypisz porty Port 1 i Port 2 jako protokół MODBUS RTU ◦</p>			Starszy bajt	Młodszy bajt								55	b7	b6	b5	b4	b3	b2	b1	b0
Starszy bajt	Młodszy bajt																			
55	b7	b6	b5	b4	b3	b2	b1	b0												

Modbus Slave	Konfiguracja portu 1~4 do pracy jako protokół MODBUS	Modbus Slave																															
<p>. Metoda 2 (metodę obsługuje FBs PLC OS V4.24 lub późniejsze wersje)</p> <p>R4047 : Górny bajt = 56H · of skonfiguruj port kom. FATEK lub protokół kom. Modbus RTU/ASCII = Inne wartości, nie obsługują powyższej funkcji</p> <p>Młodszy bajt : Przypisanie portu dla protokołów komunikacyjnych</p> <p>Format jak poniżej :</p> <table style="margin-left: 40px;"> <tr> <td style="text-align: center;">Górny bajt</td> <td style="text-align: center;">Dolny bajt</td> </tr> <tr> <td style="text-align: center;">56</td> <td style="text-align: center;">b7 b6 b5 b4 b3 b2 b1 b0</td> </tr> </table>			Górny bajt	Dolny bajt	56	b7 b6 b5 b4 b3 b2 b1 b0																											
Górny bajt	Dolny bajt																																
56	b7 b6 b5 b4 b3 b2 b1 b0																																
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;">Bity</th> <th style="width: 15%;">Wartość</th> <th style="width: 70%;">Opis</th> </tr> </thead> <tbody> <tr> <td rowspan="3" style="text-align: center;">b1b0</td> <td style="text-align: center;">0 lub 1</td> <td>Port 1 pracuje jako protokół FATEK</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Port 1 pracuje jako protokół MODBUS RTU</td> </tr> <tr> <td style="text-align: center;">3</td> <td>Port 1 pracuje jako protokół MODBUS ASCII</td> </tr> <tr> <td rowspan="3" style="text-align: center;">b3b2</td> <td style="text-align: center;">0 lub 1</td> <td>Port 2 pracuje jako protokół FATEK</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Port 2 pracuje jako protokół MODBUS RTU</td> </tr> <tr> <td style="text-align: center;">3</td> <td>Port 2 pracuje jako protokół MODBUS ASCII</td> </tr> <tr> <td rowspan="3" style="text-align: center;">b5b4</td> <td style="text-align: center;">0 lub 1</td> <td>Port 3 pracuje jako protokół FATEK</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Port 3 pracuje jako protokół MODBUS RTU</td> </tr> <tr> <td style="text-align: center;">3</td> <td>Port 3 pracuje jako protokół MODBUS ASCII</td> </tr> <tr> <td rowspan="3" style="text-align: center;">b7b6</td> <td style="text-align: center;">0 lub 1</td> <td>Port 4 pracuje jako protokół FATEK</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Port 4 pracuje jako protokół MODBUS RTU</td> </tr> <tr> <td style="text-align: center;">3</td> <td>Port 4 pracuje jako protokół MODBUS ASCII</td> </tr> </tbody> </table>			Bity	Wartość	Opis	b1b0	0 lub 1	Port 1 pracuje jako protokół FATEK	2	Port 1 pracuje jako protokół MODBUS RTU	3	Port 1 pracuje jako protokół MODBUS ASCII	b3b2	0 lub 1	Port 2 pracuje jako protokół FATEK	2	Port 2 pracuje jako protokół MODBUS RTU	3	Port 2 pracuje jako protokół MODBUS ASCII	b5b4	0 lub 1	Port 3 pracuje jako protokół FATEK	2	Port 3 pracuje jako protokół MODBUS RTU	3	Port 3 pracuje jako protokół MODBUS ASCII	b7b6	0 lub 1	Port 4 pracuje jako protokół FATEK	2	Port 4 pracuje jako protokół MODBUS RTU	3	Port 4 pracuje jako protokół MODBUS ASCII
Bity	Wartość	Opis																															
b1b0	0 lub 1	Port 1 pracuje jako protokół FATEK																															
	2	Port 1 pracuje jako protokół MODBUS RTU																															
	3	Port 1 pracuje jako protokół MODBUS ASCII																															
b3b2	0 lub 1	Port 2 pracuje jako protokół FATEK																															
	2	Port 2 pracuje jako protokół MODBUS RTU																															
	3	Port 2 pracuje jako protokół MODBUS ASCII																															
b5b4	0 lub 1	Port 3 pracuje jako protokół FATEK																															
	2	Port 3 pracuje jako protokół MODBUS RTU																															
	3	Port 3 pracuje jako protokół MODBUS ASCII																															
b7b6	0 lub 1	Port 4 pracuje jako protokół FATEK																															
	2	Port 4 pracuje jako protokół MODBUS RTU																															
	3	Port 4 pracuje jako protokół MODBUS ASCII																															