

Przełączniki programowalne **NEED**

Podręcznik użytkownika



© 2010 Relpol S.A.

SPIS TREŚCI

1. WSTĘP	6
2. INFORMACJE OGÓLNE	7
2.1. Charakterystyka	7
2.2. Opis panelu czołowego przekaźników programowalnych NEED	7
2.3. Struktura systemu i numery zamówieniowe	9
3. INSTALACJA	14
3.1. Kolejność instalacji	15
3.2. Montaż mechaniczny	15
3.2.1. Mocowanie na szynie montażowej (DIN 35mm)	15
3.2.2. Mocowanie na śruby	16
3.3. Zaciski przyłączeniowe, przewody	17
3.4. Podłączenie wejść dyskretnych 115/230V AC	18
3.5. Podłączenie wejść dyskretnych 220V DC	21
3.6. Podłączenie wejść dyskretnych 24V (12V) DC	22
3.7. Podłączenie wejść analogowych AC	24
3.8. Podłączenie wejść analogowych 220V DC	26
3.9. Podłączenie wejść analogowych 24V (12V) DC	27
3.10. Podłączenie wyjść	30
3.11. Podłączenie zasilania 115/230V AC	31
3.12. Podłączenie zasilania 220V DC	32
3.13. Podłączenie zasilania 24V (12V) DC	33
4. ZASOBY PRZEKAŹNIKA	34
4.1. System przekaźnika programowalnego NEED	34
4.2. Cykl programu	34
4.3. Zasoby przekaźnika programowalnego NEED	36
4.4. Wejścia cyfrowe	39
4.4.1. Wejścia cyfrowe normalnie otwarte	39
4.4.2. Wejścia cyfrowe normalnie zamknięte	39
4.5. Wyjścia cyfrowe	39
4.5.1. Wyjścia cyfrowe normalne	39
4.5.2. Wyjścia cyfrowe impulsowe	40
4.5.3. Wyjścia cyfrowe kasujące	40
4.5.4. Wyjścia cyfrowe ustawiające	41
4.5.5. Wyjścia cyfrowe normalne wykorzystane do dalszego sterowania	41
4.5.6. Wyjścia cyfrowe zanegowane wykorzystane do dalszego sterowania	41
4.6. Znaczniki	42
4.6.1. Znaczniki	42
4.6.2. Znaczniki Tekstowe	43
4.6.3. Znacznik MDIR	44
4.7. Timery	45
4.7.1. Timer „Opóźnione załączenie” (ON-DELAYD)	47
4.7.2. Timer „Opóźnione wyłączenie” (OFF-DELAYD)	47
4.7.3. Timer „Pojedynczy impuls” (SINGLE PULSE)	48
4.7.4. Timer „Impulsy” (FLASHING)	48
4.8. Liczniki	49
4.9. Zegary Tygodniowe	51
4.9.1. Działanie Zegara	52
4.9.2. Uwagi dotyczące konfiguracji Zegara	60
4.10. Zegar Czasu Rzeczywistego	63
4.11. Komparator – wejścia analogowe	64
4.12. Potencjometr	70
4.13. Wartości remanentne przekaźnika programowalnego	71

Zakres.....	71
4.14. Przyciski klawiatury	74
4.14.1. Przyciski jako wejścia cyfrowe normalnie otwarte	75
4.14.2. Przyciski jako wejścia cyfrowe normalnie zamknięte.....	75
5. OPIS JĘZYKÓW PROGRAMOWANIA.....	76
5.1. Programowanie w języku tekstowym STL.....	76
5.1.1. Struktura programu STL.....	76
5.1.1.1. Nazwy symboliczne.....	79
5.1.2. Opis Instrukcji STL.....	80
5.1.2.1. Instrukcja AND	80
5.1.2.2. Instrukcja nawiasów AND.....	80
5.1.2.3. Instrukcja AND NOT.....	82
5.1.2.4. Instrukcja nawiasów AND NOT	83
5.1.2.5. Instrukcja OR	84
5.1.2.6. Instrukcja nawiasów OR.....	85
5.1.2.7. Instrukcja OR NOT	86
5.1.2.8. Instrukcja nawiasów OR NOT	87
5.1.2.9. Instrukcja XOR	88
5.1.2.10. Instrukcja nawiasów XOR.....	88
5.1.2.11. Instrukcja XOR NOT.....	89
5.1.2.12. Instrukcja nawiasów XOR NOT	90
5.1.2.14. Instrukcja kasująca R (Reset).....	91
5.1.2.15. Instrukcja przyporządkowująca =.....	91
5.1.2.16. Instrukcja Przekaznik impulsowy FP	92
5.1.2.17. Instrukcje Timerów	93
5.1.2.18. Instrukcje Liczników	98
5.1.2.19. Instrukcje Zegara	104
5.1.2.20. Wejścia analogowe	105
5.1.2.21. Instrukcja 'L'	106
5.1.2.22. Instrukcja zawsze ustawiająca SET.....	114
5.1.2.23. Instrukcja zawsze kasująca CLR.....	115
5.2. Programowanie w języku graficznym LAD	116
5.2.1. Symbole w LAD	116
5.2.2. Wejścia	117
5.2.3. Wyjścia	117
5.2.4. Struktura programu w LAD.....	118
5.2.5. Struktura obwodu LAD	118
5.2.6. Opis używanych elementów.....	119
5.2.7. Konfiguracja.....	121
5.2.7.1. Konfiguracja wejść	121
5.2.7.2. Konfiguracja wyjść	122
5.2.7.3. Konfiguracja Znaczników (Markerów).....	122
5.2.7.4. Konfiguracja Timerów.....	123
5.2.7.5. Konfiguracja Liczników.....	124
5.2.7.6. Przykłady konfiguracji.....	125
5.2.8. Zasady umieszczania elementów.....	126
5.2.9. Rodzaje połączeń	127
5.2.9.1. Odwzorowanie wejścia na wyjście.....	127
5.2.9.2. Odwzorowanie zanegowanego wejścia na wyjście.....	127
5.2.9.3. Połączenia szeregowo	127
5.2.9.4. Połączenia równoległe	128
5.2.9.5. Połączenie szeregowo-równoległe	129
5.2.10. Nazwy symboliczne.....	130
5.2.11. Program w LAD.....	130
6. INSTALACJA I OPIS OPROGRAMOWANIA.....	131

6.1. Wymagania sprzętowe	131
6.2. Instalacja oprogramowania	131
6.3. Deinstalacja	131
6.4. Połączenie komputera z przekaźnikiem programowalnym	131
6.5. Szybki start – tworzenie aplikacji	132
6.6. Praca z programem PC Need	138
6.6.1. Opis głównego okna programu	138
6.6.2. Pasek Menu	139
6.6.3. Pasek narzędzi	141
6.6.4. Skróty klawiaturowe	142
6.7. Edytor programu STL	144
6.7.1. Edytor STL	144
6.7.2. Kompilacja STL	145
6.7.3. Konfiguracja edytora STL	146
6.8. Edytor programu LAD	147
6.8.1. Nowy program	147
6.8.2. Zapisanie programu	148
6.8.3. Otwarcie istniejącego programu	149
6.8.4. Wprowadzanie i edycja schematu połączeń	149
6.8.5. Edycja elementu	153
6.8.6. Komentarze	156
6.8.7. Konfiguracja edytora LAD	158
6.9. Ustawienia	159
6.9.1. Rodzaje ustawień	159
6.9.2. Ustawienia Timerów	161
6.9.3. Ustawienia Zegarów	162
6.9.4. Ustawienia Liczników	162
6.9.5. Ustawienia Komparatorów	163
6.9.6. Remanencja	163
6.9.7. Ustawienia wejść	164
6.9.8. Markery tekstowe	164
6.10. Podgląd zmiennych	166
6.11. Podgląd drabinki LAD	168
6.12. Hasło	169
6.12.1. Wprowadzanie hasła	169
6.12.2. Zmiana hasła	170
6.13. Zegar czasu rzeczywistego (RTC)	171
6.14. Kod źródłowy	172
6.15. Konwersja dokumentu	173
7. URUCHOMIENIE	175
7.1. Załączenie	175
7.1.1. Czynności wstępne dla wersji AC	175
7.1.2. Czynności wstępne dla wersji DC	175
7.1.3. Załączenie zasilania	175
8. INFORMACJE SPRZĘTOWE	176
8.1. Zasilanie przekaźnika	176
8.1.1. Zasilanie przekaźnika 115/230 V AC	176
8.1.2. Zasilanie przekaźnika 220 V DC	176
8.1.3. Zasilanie przekaźnika 24 (12) V DC	176
8.2. Wejścia	177
8.2.1. Wejścia 230 V AC	177
8.2.2. Wejścia 220 V DC	179
8.2.3. Wejścia 24 (12) V DC	180
8.3. Wyjścia	180
8.4. Opóźnienie wejść	181

8.4.1. Opóźnienia wejść dla przekaźnika NEED-230AC-x1-..	181
8.4.2. Opóźnienia wejść dla przekaźnika NEED-24DC -.., NEED-12DC -..,	183
NEED-220DC -..	183
8.5. Opóźnienie wyjść.....	184
9. PAMIĘĆ ZEWNĘTRZNA.....	185
9.1. Karty pamięci.....	185
9.2. Organizacja pamięci.....	186
9.3. Programowanie pamięci.....	186
9.3.1. Zapis programu.....	186
9.3.2. Zapis ustawień.....	187
9.3.3. Status pamięci EEPROM.....	187
9.3.4. Odczyt ustawień	188
9.4. Współpraca karty pamięci z przekaźnikiem NEED.....	188
10. WYŚWIETLACZ LCD i KŁAWIATURA	189
10.1. Informacje wstępne	189
10.2. Menu główne	191
10.3. Menu Konfiguracja.....	193
10.3.1. Menu Ustawienia	193
10.3.1.1. Timery.....	194
10.3.1.2. Menu Zegary.....	199
10.3.1.3. Menu Liczniki	203
10.3.1.4. Menu Komparatory.....	205
10.3.1.5. Menu Markery	208
10.3.1.6. Menu Wejścia	210
10.3.1.7. Menu Wyjścia.....	211
10.3.1.8. Menu Przyciski.....	213
10.3.1.9. Menu Szybki Licznik.....	213
10.3.1.10. Menu ASYM & MDIR.....	216
10.3.2. Menu System.....	217
10.3.2.1. Podświetlenie	218
10.3.2.2. Język.....	219
10.3.3. Menu Hasło.....	220
10.3.3.1. Wprowadzanie hasła	220
10.3.3.2. Zmiana hasła	221
10.3.4. Menu Wersja.....	224
10.4. Menu Czas	225
10.5. Menu Data.....	229
10.6. Menu Alarmy	232
10.7. Menu Pamięć zewnętrzna	235
11. PRZYKŁADY ZASTOSOWAŃ.....	237
11.1. Ocena wysokości detalu	237
11.2. Drzwi automatyczne	242
11.3. Dzwonki w szkole	250
11.4. Wykrywanie wad detali	256
11.5. Sterowanie ruchem wózków na zakręcie taśmociągu	260
11.6. Sterownik oświetlenia i wentylacji	264
11.7. Kontrola obciążenia	271
11.8. Sterowanie i zabezpieczenie silnika 3-fazowego	274
12. DANE TECHNICZNE	280
13. SŁOWNIK.....	291
14. INDEKS.....	260

1. WSTĘP

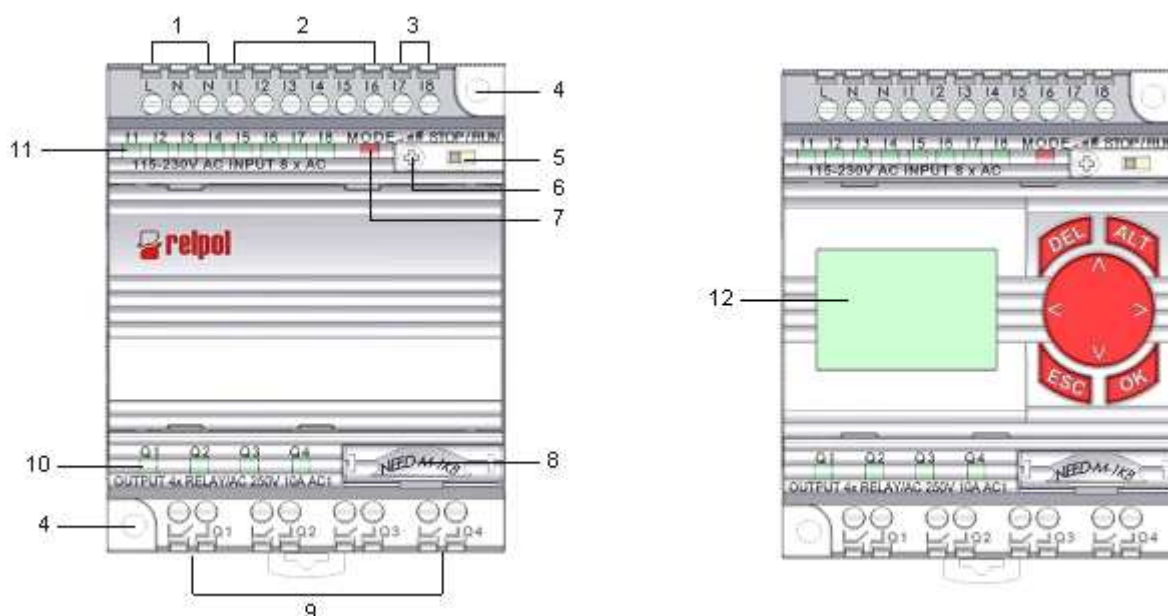
Przełącznik NEED jest przełącznikiem programowalnym, który może zastąpić skomplikowane połączenia przełącznikowe lub stycznikowe. Urządzenie to jest swobodnie programowalne, tzn. – w dowolnym czasie można korygować, zmieniać istniejący program w pamięci sterownika bez zmiany układów peryferyjnych całego otoczenia – co, w przypadku tradycyjnego sterowania przełącznikowego, praktycznie było niemożliwe. Duże możliwości i doskonałe parametry w połączeniu z funkcjonalnością przełącznika programowalnego pozwalają zaoszczędzić czas projektowania a przede wszystkim koszty wdrażanych aplikacji.

2. INFORMACJE OGÓLNE

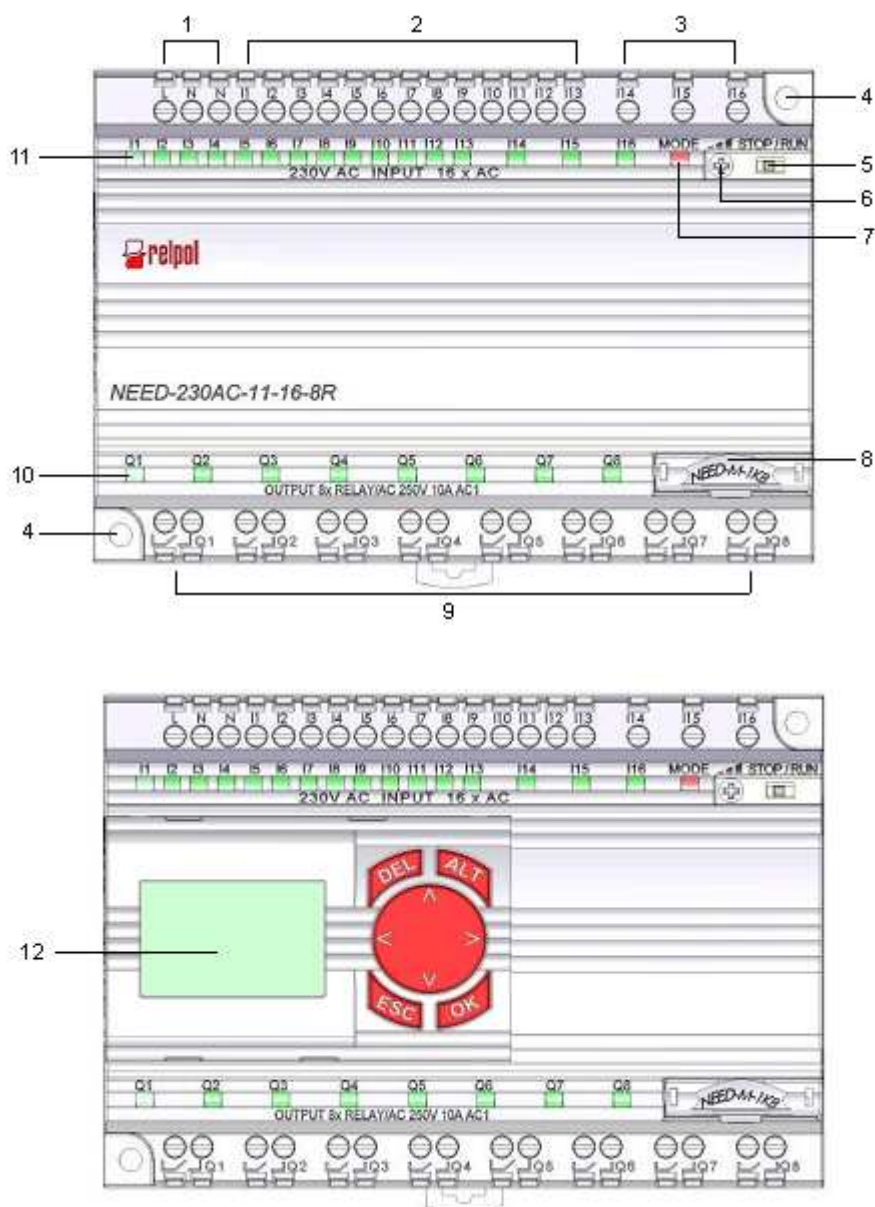
2.1. Charakterystyka

- 6 lub 13 wejść cyfrowych,
- 2 lub 3 wejścia cyfrowo-analogowe:
 NEED-230AC-... : 0–255V AC,
 NEED-220DC-... : 0–255V DC,
 NEED-24DC-..., NEED-12DC-... :
 zakres napięciowy: 0–25,5V DC (krok 0,1V) lub 0 – 12,75V (krok 0,05V),
 zakres prądowy: 0–51mA (krok 0,2mA) lub 0 – 25,5mA (krok 0,1mA),
- 4 lub 8 wyjść przekaźnikowych (230V AC/10A),
- podświetlany wyświetlacz o wysokim kontraście (4 wiersze po 12 znaków) i klawiatura,
- cztery programowalne przez użytkownika przyciski klawiatury
- potencjometr do zadawania wartości analogowych,
- zegar czasu rzeczywistego,
- automatyczna zmiana czasu lato/zima w różnych strefach czasowych,
- wskaźnik trybu pracy,
- przełącznik trybu pracy RUN/STOP,
- wskaźniki stanów wejść/wyjść,
- możliwość programowania LAD i STL,
- oprogramowanie PC,
- moduł pamięci zewnętrznej,
- szybki licznik/miernik częstotliwości do 20kHz,
- detekcja prawidłowego kierunku podłączenia faz L1, L2, L3,
- pomiar asymetrii faz L1, L2, L3.

2.2. Opis panelu czołowego przekaźników programowalnych NEED



Rys. 2.2.1. Opis panelu czołowego przekaźnika programowalnego NEED..-08-4x-...

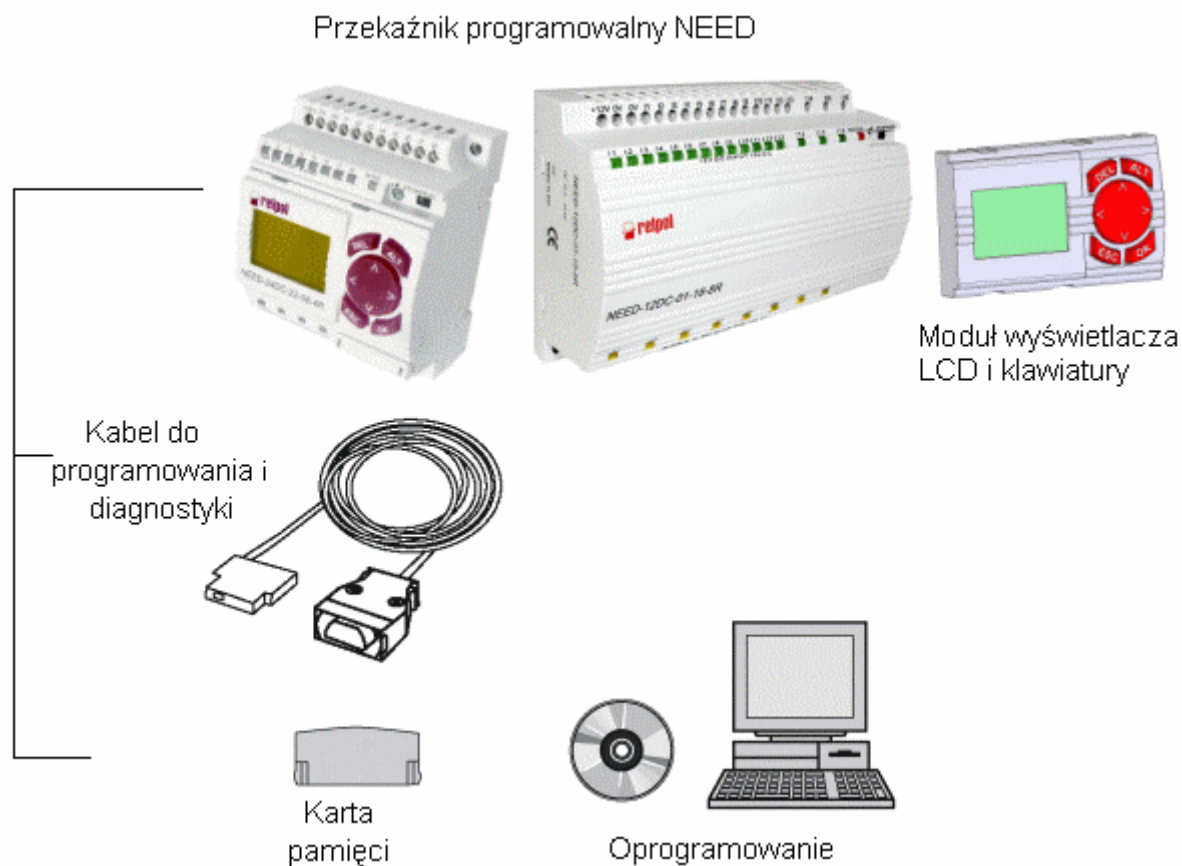


Rys. 2.2.2. Opis panelu czołowego przekaźnika programowalnego NEED...-16-8x-...

Tabela 2.2.1.

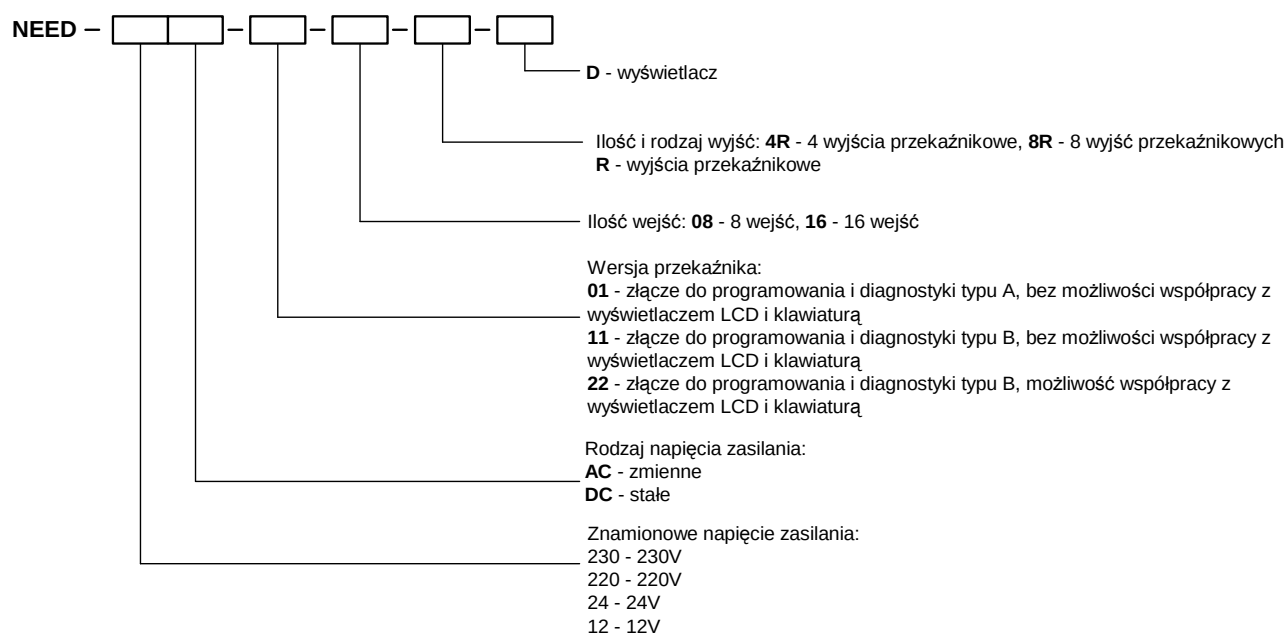
Oznaczenie	Opis
1	Zaciski śrubowe zasilania
2	Zaciski śrubowe wejść cyfrowych
3	Zaciski śrubowe wejść cyfrowo-analogowych
4	Otwory mocujące
5	Przełącznik trybu pracy RUN–STOP
6	Potencjometr do zadawania wartości analogowych
7	Wskaźnik LED stanu przekaźnika
8	Złącze do programowania oraz dodatkowego modułu pamięci programu
9	Zaciski śrubowe wyjść
10	Wskaźniki LED stanu wyjść
11	Wskaźniki LED stanu wejść
12	Wyświetlacz LCD z klawiaturą

2.3. Struktura systemu i numery zamówieniowe



Rys. 2.3.1. Struktura systemu.

Oznaczenie typu





1. Wersja NEED-...x1-.. nie współpracuje z wyświetlaczem LCD i klawiaturą.
2. Wersja NEED-...-22-08-4R posiada możliwość zastosowania dołączanego modułu wyświetlacza LCD i klawiatury. Można zamawiać przekaźnik bez modułu wyświetlacza i klawiatury np. NEED-24DC-22-08-4R lub z modułem wyświetlacza i klawiatury NEED-24DC-22-08-4R-D.
3. Wersja NEED-...-22-16-.. zawsze posiada wbudowany moduł wyświetlacza LCD i klawiatury.

Tabela 2.3.1. Typy przekaźników współpracujące z modułem wyświetlacza LCD.

Typ	Wbudowany wyświetlacz LCD z klawiaturą	Dołączany wyświetlacz LCD z klawiaturą
NEED-...x1-..	–	–
NEED-...-22-08-4x-..	–	Tak
NEED-...-22-16-8x-D	Tak	–

Przykład1:

NEED – 230AC – 01 – 08 – 4R

Przekaźnik programowalny NEED – znamionowe napięcie zasilania 230V AC – wersja 01, ze złączem do programowania typu A – 8 wejść – 4 wyjścia przekaźnikowe, bez możliwości podłączenia wyświetlacza LCD.

Przykład2:

NEED – 24DC – 22 – 08 – 4R

Przekaźnik programowalny NEED – znamionowe napięcie zasilania 24V DC – wersja 22, ze złączem do programowania typu B – 8 wejść – 4 wyjścia przekaźnikowe, z możliwością podłączenia wyświetlacza LCD.

Przykład3:

NEED – 24DC – 22 – 16 – 8R – D

Przekaźnik programowalny NEED – znamionowe napięcie zasilania 24V DC – wersja 22, ze złączem do programowania typu B – 16 wejść – 8 wyjść przekaźnikowych, z wbudowanym wyświetlaczem LCD.

Tabela 2.3.2. Kody zamówieniowe.

Nazwa	Oznaczenie
Przekaźnik programowalny NEED	Patrz oznaczenie typu
Moduł wyświetlacza LCD i klawiatury	NEED – LCD – A
Kabel RS232 do programowania i diagnostyki ze złączem typu A	NEED – PC – 15A
Kabel RS232 do programowania i diagnostyki ze złączem typu B	NEED – PC – 15B
Kabel USB do programowania i diagnostyki ze złączem typu B	NEED – PC – 15C
Karta pamięci ze złączem typu A	NEED – M – 1K
Karta pamięci ze złączem typu B dla NEED-...x1-..	NEED – M – 1KB
Karta pamięci ze złączem typu B dla NEED-...22-..	NEED – M – 4KB
Oprogramowanie	NEED – PC Need
Podręcznik użytkownika	Przekaźnik programowalny NEED –

	podręcznik użytkownika
--	------------------------

W Tabeli 2.3.3. przedstawione są wszystkie typy przekaźników programowalnych NEED wraz z informacjami o możliwości współpracy z modułem wyświetlacza LCD, kablem do programowania oraz pamięcią.

Tabela 2.3.3. Tabela wyboru.

Typ przekaźnika	Moduł LCD	Kabel	Pamięć
NEED-230AC-01-08-4R	–	NEED-PC-15A	NEED – M – 1K
NEED-12DC-01-08-4R	–	NEED-PC-15A	NEED – M – 1K
NEED-24DC-01-08-4R	–	NEED-PC-15A	NEED – M – 1K
NEED-230AC-11-08-4R	–	NEED-PC-15B NEED-PC-15C	NEED – M – 1KB
NEED-220DC-11-08-4R	–	NEED-PC-15B NEED-PC-15C	NEED – M – 1KB
NEED-12DC-11-08-4R	–	NEED-PC-15B NEED-PC-15C	NEED – M – 1KB
NEED-24DC-11-08-4R	–	NEED-PC-15B NEED-PC-15C	NEED – M – 1KB
NEED-230AC-11-16-8R	–	NEED-PC-15B NEED-PC-15C	NEED – M – 1KB
NEED-12DC-11-16-8R	–	NEED-PC-15B NEED-PC-15C	NEED – M – 1KB
NEED-24DC-11-16-8R	–	NEED-PC-15B NEED-PC-15C	NEED – M – 1KB
NEED-230AC-22-08-4R-D	NEED – LCD – A	NEED-PC-15B NEED-PC-15C	NEED – M – 4KB
NEED-12DC-22-08-4R-D	NEED – LCD – A	NEED-PC-15B NEED-PC-15C	NEED – M – 4KB
NEED-24DC-22-08-4R-D	NEED – LCD – A	NEED-PC-15B NEED-PC-15C	NEED – M – 4KB
NEED-230AC-22-16-8R-D	Wbudowany	NEED-PC-15B NEED-PC-15C	NEED – M – 4KB
NEED-12DC-22-16-8R-D	Wbudowany	NEED-PC-15B NEED-PC-15C	NEED – M – 4KB
NEED-24DC-22-16-8R-D	Wbudowany	NEED-PC-15B NEED-PC-15C	NEED – M – 4KB



NEED ze złączem do programowania i diagnostyki typu A może być programowany tylko za pomocą przewodu do programowania typu A. Karta pamięci musi posiadać także złącze typu A

Przykład4:

Przy zamówieniu przekaźnika: *NEED – 230AC – 01 – 08 – 4R* należy zamówić także:
NEED – PC – 15A – kabel do programowania i diagnostyki ze złączem typu A oraz
NEED – M – 1K - kartę pamięci ze złączem typu A.

Przykład5:

Przy zamówieniu przekaźnika: *NEED – 230AC – 22 – 08 – 4R – D* należy zamówić także:
NEED – PC – 15B lub NEED – PC – 15C – kabel do programowania i diagnostyki ze złączem typu B oraz NEED – M – 4KB.



Rys. 2.3.2. Kabel RS232 do programowania i diagnostyki oraz pamięć ze złączem typu B.



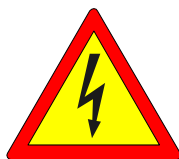
Rys. 2.3.3. Kabel RS232 do programowania i diagnostyki oraz pamięć ze złączem typu A.



Rys. 2.3.4. Kabel USB do programowania i diagnostyki ze złączem typu B.

3. INSTALACJA

Stosowane oznaczenia:



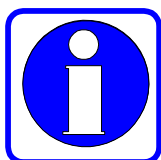
Niebezpieczeństwo porażenia prądem!



Nie prowadzić żadnych prac pod napięciem!



Ostrzeżenie!



Informacje i wskazówki.

Przed instalacją przekaźników programowalnych zapoznaj się z poniższymi uwagami!!!



W przekaźniku programowalnym na jego podłączeniach występują napięcia niebezpieczne dla życia ludzkiego.

- Wyłącz urządzenie/instalację, w którym montujesz przekaźnik programowalny.
- Zabezpiecz urządzenie/instalację przed przypadkowym załączeniem.
- Upewnij się, że żadne napięcie nie występuje w urządzeniu/instalacji.
- W przekaźniku programowalnym ustaw przełącznik w tryb STOP.
- Wykonaj wszystkie konieczne pomiary i sprawdzenia, aby nie doszło do niezamierzonego zadziałania przekaźnika programowalnego.
- Pamiętaj o odprowadzeniu ładunków elektrostatycznych przed dotknięciem przekaźnika.
- Koniecznie podłącz zabezpieczenia przeciwzwarceniowe i ochronne.
- Przestrzegaj zasad i zaleceń zawartych w instrukcji użytkownika.
- Montaż przekaźnika programowalnego powinien zostać dokonany przez osobę znającą zasady montażu elektrycznego.
- Pamiętaj, że zainstalowane urządzenia muszą być zabezpieczone przed nieumyślnym uruchomieniem.
- Wszystkie podłączenia przekaźnika programowalnego muszą być zgodne z odpowiednimi normami bezpieczeństwa.
- Parametry sieci zasilającej nie powinny przekraczać granicznych progów tolerancji przedstawionych w danych technicznych przekaźnika.
- Jeżeli stosujesz przekaźnik w układach, gdzie konieczne jest zatrzymanie awaryjne, określ zachowanie się układu w czasie zadziałania i odblokowywania systemu zatrzymania awaryjnego, w celu uniknięcia nieprzewidzianych sytuacji – np. niekontrolowanego startu systemu automatyki.
- Określ poprawność zachowania układu w reakcji na wyłączenie napięcia zasilającego oraz na jego ponowne załączenie.

Warunki bezpieczeństwa

- W celu zapewnienia bezpieczeństwa obsługi i niezawodnego działania urządzenia montaż przekaźnika programowalnego powinien zostać wykonany przez osobę znającą zasady montażu elektrycznego.
- W czasie montażu należy przestrzegać norm bezpieczeństwa pracy z urządzeniami elektrycznymi i zasad BHP.
- Przestrzegać warunków instalacji przekaźnika programowalnego.

3.1. Kolejność instalacji

1. Przygotowanie i zabezpieczenie miejsca instalacji.
2. Montaż mechaniczny.
3. Podłączenie przewodów:
 - podłączenie wejść
 - podłączenie wyjść.
 - podłączenie zasilania.

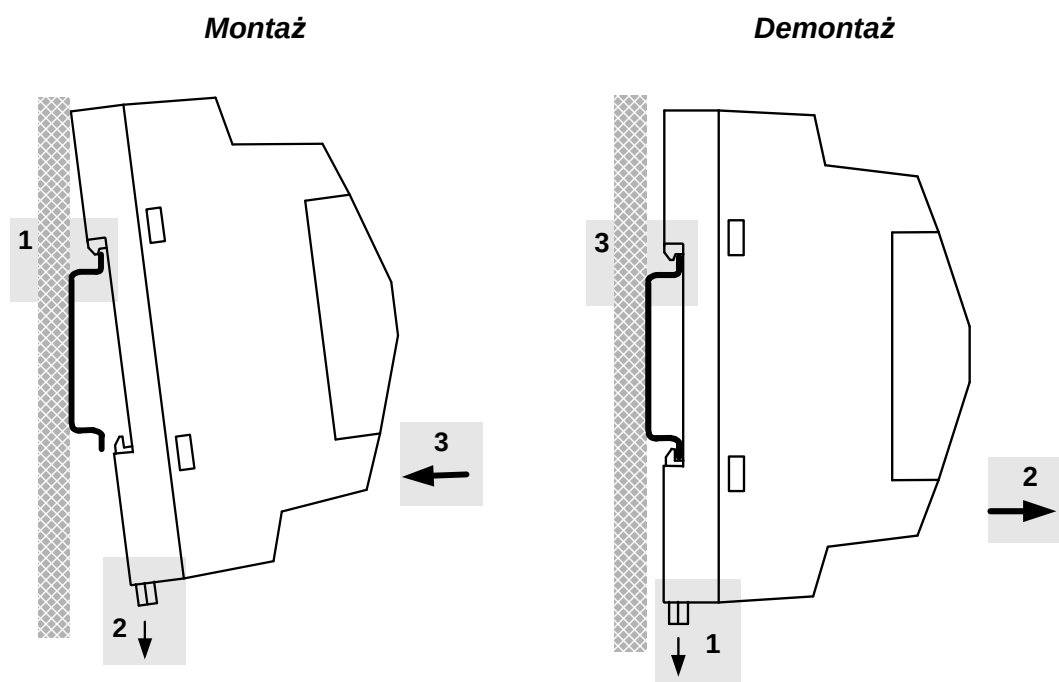
3.1.1. Przygotowanie i zabezpieczenie miejsca instalacji



- Wyłącz urządzenie/instalację, w którym montujesz przekaźnik programowalny.
- Uważaj, występuje niebezpieczeństwo porażenia prądem.
- Zabezpiecz urządzenie/instalację przed przypadkowym załączeniem.
- Upewnij się, że żadne napięcie nie występuje w urządzeniu/instalacji.
- Jeżeli nie jest możliwe całkowite odłączenie napięcia w strefie montażu, to należy zabezpieczyć miejsca występowania zagrożenia przed dotknięciem; zachowywać szczególną ostrożność!
- Sprawdź stan izolacji stosowanych przewodów.

3.2. Montaż mechaniczny

3.2.1. Mocowanie na szynie montażowej (DIN 35mm)



Rys. 3.2.1. Mocowanie przekaźnika na szynie montażowej.

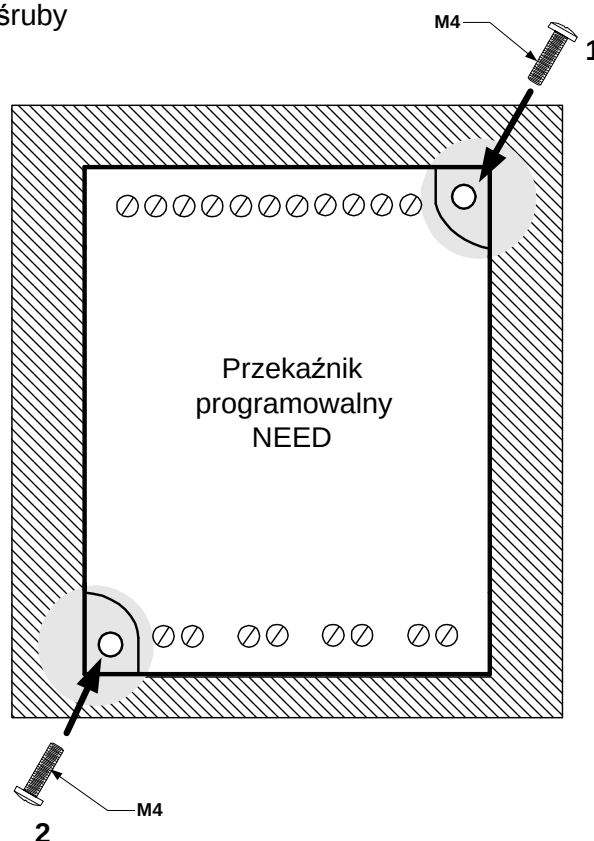
Montaż (rys. 3.2.1.)

1. Zaczep moduł na górnej części szyny montażowej.
2. Odciągnij dolny zaczep w dół.
3. Trzymając odciągnięty dolny zaczep dopchnij moduł od przodu i zwolnij odciągnięty zaczep.
4. Sprawdź pewność zamocowania modułu przekaźnika.

Demontaż (rys. 3.2.1.)

1. Odciągnij dolny zaczep w dół.
2. Trzymając odciągnięty dolny zaczep odchyl moduł od szyny.
3. Podnieś odchylony moduł i zdejmij z górnych zaczepów.

3.2.2. Mocowanie na śruby



Rys. 3.2.2. Mocowanie za pomocą śrub.

Montaż na 2 śruby (lub blachowkręty).

Średnica otworów przewidzianych pod montaż śrubowy: 5,5 mm.

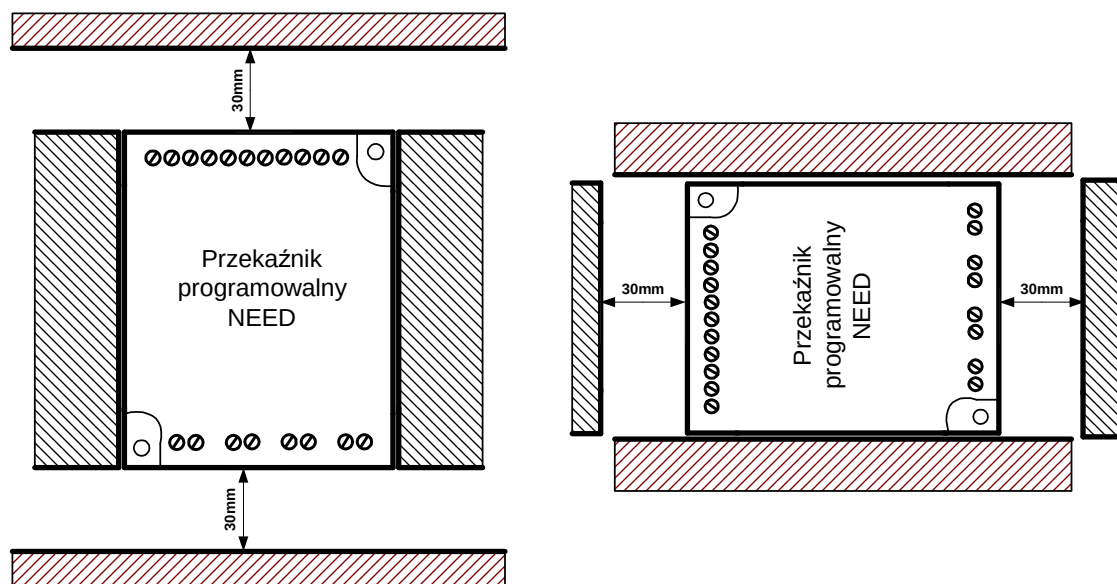
Uwaga: Do montażu na śruby nie są potrzebne żadne dodatkowe adaptory, wystarczy skorzystać z przewidzianych do tego celu otworów montażowych.

Odstępy montażowe:

Zaleca się zachować odstęp 3 cm pomiędzy krawędziami złącz wejściowych i wyjściowych przełącznika programowalnego a innymi elementami instalacji (koryta montażowe, inne aparaty, ściana szafy montażowej itp.). Pozwoli to na wygodne okablowanie oraz umożliwi dobre chłodzenie modułu. Odstępy montażowe przedstawiono na rys. 3.2.3.

Ściany boczne mogą przylegać bezpośrednio do innych aparatów, elementów obudowy itp. Uwagi powyższe odnoszą się zarówno do montażu poziomego jak i pionowego, istotny jest odstęp od krawędzi ze złączami.

Należy pamiętać także o pozostawieniu odstępu min. 25mm od frontu urządzenia, przy montażu w zamykanej szafie.



Rys. 3.2.3. Odstępy montażowe – montaż pionowy i poziomy.

3.3. Zaciski przyłączeniowe, przewody

Zastosowane zaciski przyłączeniowe umożliwiają zastosowanie przewodów o przekrojach doprowadzeń:

$0,25\text{mm}^2$ do 4mm^2 - przewód pojedynczy

$0,25\text{mm}^2$ do $2,5\text{mm}^2$ - przewód typu linka z końcówką tulejkową

Moment dokręcania śruby zacisku: $0,5\text{ Nm}$ (max $0,6\text{Nm}$).

- Przewody doprowadzeniowe powinny być jak najkrótsze, ale nie naciągnięte.
- W przypadku zastosowania długich przewodów należy stosować ich ekranowanie lub skręcenie przewodów parami (przewodu fazowego L bądź sygnałowego z przewodem zerowym N lub 0V z sygnałem wejściowym $+12(24)\text{V}$ dla wersji DC).
- Sugeruje się izolowanie obwodów prądu zmiennego i stałego oraz obwodów wytwarzających impulsy elektryczne poprzez odpowiednie prowadzenie przewodów. Można to zrealizować poprzez unikanie równoległego prowadzenia przewodów zasilających i sygnałowych, skręcanie parami, ekranowanie z jednostronnym uziemieniem ekranu.
- Przekrój przewodu dobrać uwzględniając prąd przepływający przez obciążenie.



Uwaga: We wszelkich nieujętych tutaj przypadkach należy stosować aktualne przepisy, standardy i regulacje urzędowe dotyczące instalacji elektrycznych.

3.4. Podłączenie wejść dyskretnych 115/230V AC



Wejścia należy łączyć do tej samej fazy, z której zasilany jest przekaźnik programowalny.

Odwrotne podłączenie zasilania tzn. zamienienie miejscami przewodów fazowego (L) i neutralnego (N) na wejściach zasilających przekaźnika programowalnego może spowodować wystąpienie niebezpiecznych napięć na jego zaciskach wejściowych, złączu komunikacyjnym oraz brak wykrywania stanów logicznych.



Wejścia nie są izolowane galwanicznie od sieci elektrycznej zasilającej przekaźnik.



Niebezpieczeństwo porażenia prądem – w przypadku zamiany przewodu zerowego N z fazowym L lub braku podłączenia przewodu N może występować na zaciskach wejściowych lub złączu komunikacyjnym napięcie o wysokości napięcia zasilającego.

Do wejść można podłączać elementy stykowe: przyciski (zwierne, rozwierne), łączniki, przełączniki, styki przekaźników, styczników oraz fotokomórki i czujniki zbliżeniowe 2 lub 3 przewodowe 115/230V AC.

Zakres napięć sygnałów wejściowych zgodnie z normą PN-EN 61131:

Wejście wyłączone: 0 do 40 V AC ('0' logiczne)

Wejście załączone: 85 do 260 V AC ('1' logiczna)

Prąd wejściowy dla NEED-230AC-xx-08-4:

I1 do I4 : 0,6 mA przy 230 V AC

I5, I6 : 8 mA przy 230 V AC – zwiększona odporność na zakłócenia, możliwość podłączenia długich przewodów – patrz specyfikacja techniczna.

I7, I8 : 0,9 mA przy 230 V AC

Prąd wejściowy dla wersji NEED-230AC-xx-16-8:

I1 do I11 : 0,6 mA przy 230 V AC

I12, I13 : 8 mA przy 230 V AC – zwiększona odporność na zakłócenia, możliwość podłączenia długich przewodów – patrz specyfikacja techniczna.

I4, I15, I16 : 1,5 mA przy 230 V AC

Wejścia mają charakter rezystancyjny poza wejściami I5, I6 dla wersji NEED-230AC-xx-08-4 lub I12, I13 dla NEED-230AC-xx-16-8 (charakter rezystancyjno-pojemnościowy), do których można podłączyć dłuższe przewody.

Nie należy stosować zbyt długich przewodów doprowadzających ze względu na ich pojemność i podatność na zakłócenia elektromagnetyczne – może to się objawić niekontrolowanymi stanami wejść logicznych np. błędne sygnalizowanie stanu załączenia wejścia.

Długości przewodów, jakie można podłączyć dla wejść zależą od układu wewnętrznego wejścia:

a) do wejść o normalnej odporności na zakłócenia można podłączyć przewód o długości do 10m – pomiar został wykonany dla najbardziej niekorzystnego przypadku prowadzenia przewodów fazowego i wejściowego równolegle obok siebie (podłączenie np. za pomocą dwużyłowego kabla sieciowego).

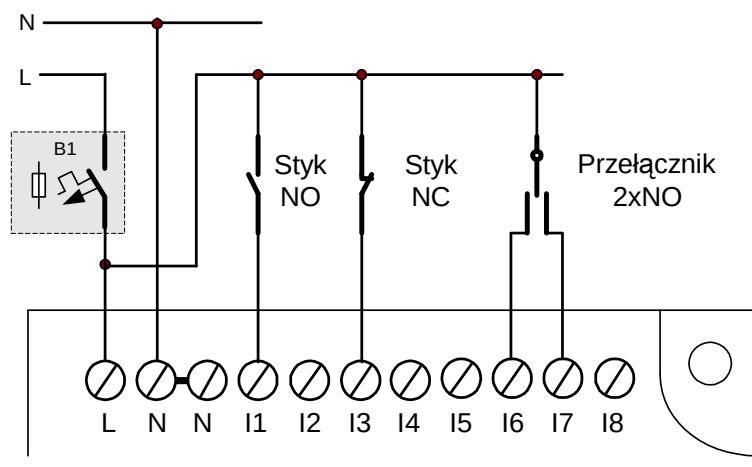
b) do wejść o zwiększonej odporności na zakłócenia można podłączyć przewód o długości 100m dlatego, że zawierają wewnętrzny kondensator 100nF zwiększający prąd wejściowy.

c) do wejść, które mogą pełnić rolę wejść analogowych podobnie jak do wejść zwykłych można podłączyć przewód o długości do 10m.

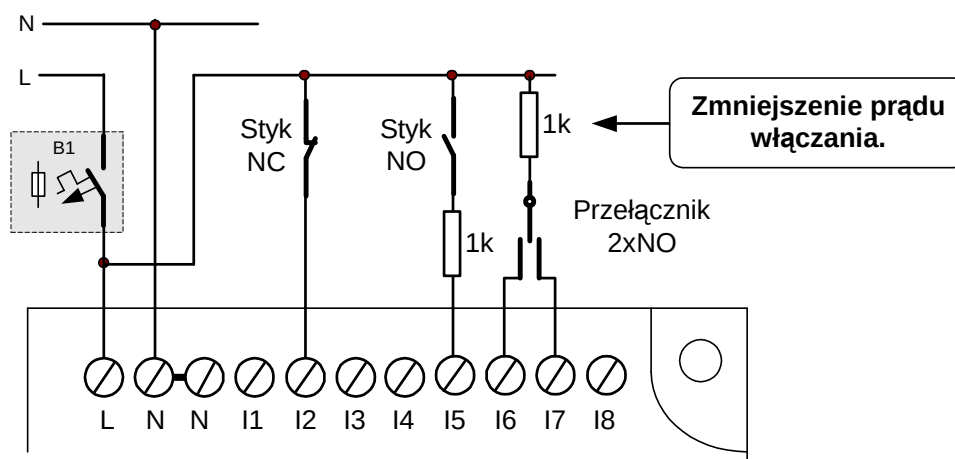


Wejścia I7 i I8 dla wersji NEED-230AC-xx-8-.. oraz I14, I15, I16 dla wersji NEED-230AC-xx-16-.. mogą być używane jako cyfrowe lub analogowe – zależy to od sposobu ich wykorzystania w programie.

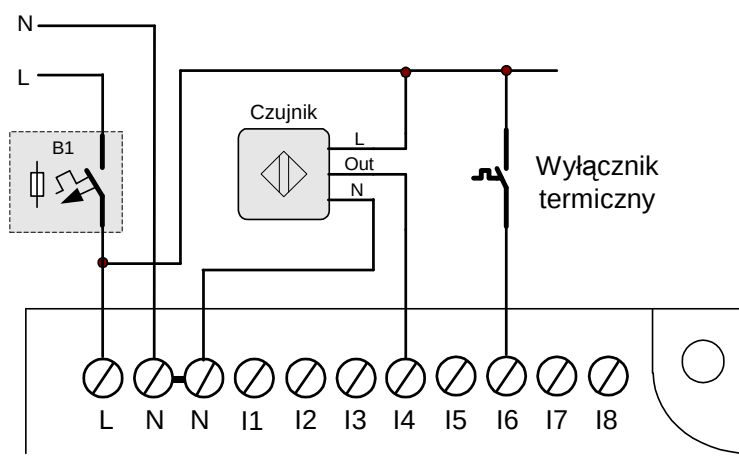
Dla wejść o zwiększonej odporności na zakłócenia aby ograniczyć prąd załączania zaleca się wpięcie szeregowo z elementem stykowym rezystora o wartości ok. 1k Ω 1W (rys. 3.4.2. przedstawia podłączenie dla wersji NEED-230AC-xx-8-..). Wejścia I5, I6 dla wersji NEED-230AC-xx-8-.. oraz I12, I13 dla wersji NEED-230AC-xx-16-.. posiadające wewnętrzne kondensatory można zbocznikować zewnętrznymi rezystorami (100k Ohm) włączonymi pomiędzy wejście, a przewód N w celu szybszego rozładowania ich pojemności.



Rys. 3.4.1. Podłączenie wejść – elementy stykowe.



Rys. 3.4.2. Podłączenie wejść – elementy stykowe + rezystory zmniejszające udar prądowy wejścia.



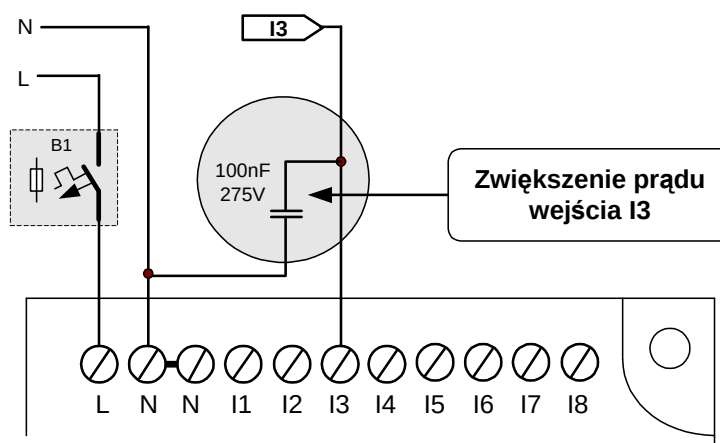
Rys. 3.4.3. Podłączenie wejść – czujnik zbliżeniowy, styk wyłącznika termicznego.



W celu zmniejszania zakłóceń na wejściach dyskretnych oraz zwiększenia długości przewodów, jakimi można podłączać do tych wejść elementy sterujące, należy zastosować elementy zewnętrzne zwiększające prąd w obwodzie wejścia oraz filtry wejściowe.

1. Zwiększenie prądu w obwodzie wejścia.

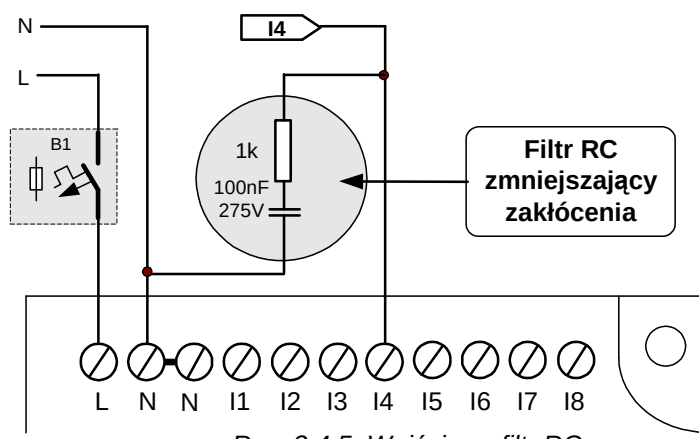
W celu zmniejszenia zakłóceń na wejściach I1..I4, I7, I8 dla NEED-230AC-xx-8-.. oraz I1..I11 i I14..I16 dla NEED-230AC-xx-16-.. można zastosować zewnętrzny kondensator, np. 100nF/275V klasy X1 lub X2 (zwiększenie prądu), podłączony pomiędzy zacisk wejściowy a zacisk N (rys. 3.4.4.).



Rys. 3.4.4. Zwiększenie prądu wejścia.

2. Filtr RC

Dla zmniejszenia zakłóceń na wejściach I1..I4, I7, I8 dla NEED-230AC-x1-8-.. oraz I1..I11 i I14..I16 dla NEED-230AC-11-16-.. można zastosować filtr RC (szeregowo połączony kondensator 100nF/275V klasy X1 lub X2 i rezystor 1k) podłączony pomiędzy zacisk wejściowy a zacisk N (rys. 3.4.5.).



Rys. 3.4.5. Wejściowy filtr RC.



Uwaga: Podanie pomiędzy zaciski wejściowe I1..In a N napięcia wyższego od maksymalnie dopuszczalnego może spowodować zniszczenie obwodów wejściowych przekaźnika programowalnego.

3.5. Podłączenie wejść dyskretnych 220V DC



Niebezpieczeństwo porażenia prądem – na zaciskach występują napięcia niebezpieczne dla życia.



Wejścia nie są izolowane galwanicznie od sieci elektrycznej zasilającej przekaźnik.



Uwaga: Podanie pomiędzy zaciski wejściowe I1..In a 0V napięcia wyższego od maksymalnie dopuszczalnego może spowodować zniszczenie obwodów wejściowych przekaźnika programowalnego.

Do wejść można podłączać elementy stykowe: przyciski (zwierne, rozwierne), łączniki, przełączniki, styki przekaźników, styczników oraz inne elementy przewidziane do pracy na napięcie 220V DC.

Zakres napięć sygnałów wejściowych zgodnie z normą PN-EN 61131:

Wejście wyłączone: 0 do 40 V DC ('0' logiczne)

Wejście załączone: 85 do 260 V DC ('1' logiczna)

Prąd wejściowy dla NEED-220DC-xx-08-4:

I1 do I6 : 0,6 mA przy 220 V DC

I7, I8 : 1,1 mA przy 220 V DC

Prąd wejściowy dla wersji NEED-220DC-xx-16-8:

I1 do I13 : 0,6 mA przy 220 V DC

I4, I15, I16 : 1,1 mA przy 220 V DC

Nie należy stosować zbyt długich przewodów doprowadzających ze względu na ich pojemność i podatność na zakłócenia elektromagnetyczne – może to się objawić niekontrolowanymi stanami wejść logicznych np. błędne sygnalizowanie stanu załączenia wejścia.

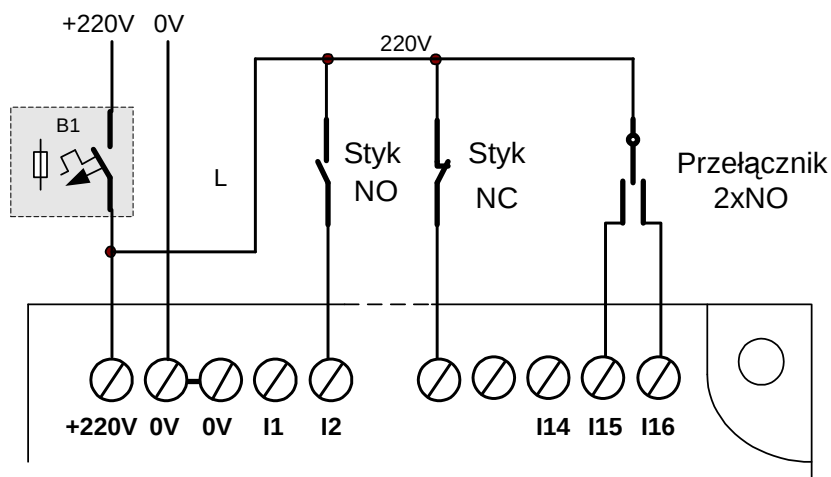
Długości przewodów, jakie można podłączyć dla wejść zależą od układu wewnętrznego wejścia:

- do wejść dyskretnych można podłączyć przewód o długości do 10m – pomiar został wykonany dla najbardziej niekorzystnego przypadku prowadzenia przewodów zasilającego i wejściowego równoległe obok siebie (podłączenie np. za pomocą dwużyłowego kabla).
- do wejść, które mogą pełnić rolę wejść analogowych można podłączyć przewód ekranowany o długości do 40m.



Wejścia mają charakter rezystancyjny.

Wejścia I7 i I8 dla wersji DC NEED..-xx-8-.. oraz I14, I15, I16 dla wersji NEED..-xx-16-.. mogą być wykorzystywane jako dyskretnie lub analogowe – zależy to od sposobu ich użycia w programie.



Rys. 3.5.1. Podłączenie wejść – elementy stykowe.

3.6. Podłączenie wejść dyskretnych 24V (12V) DC

Do zacisków wejściowych można podłączać elementy stykowe: przyciski (zwierne, rozwierne), łączniki, przełączniki, styki przekaźników, styczników oraz fotokomórki i czujniki zbliżeniowe 2 lub 3 przewodowe 24V (12V) DC.

Zakresy napięć sygnałów wejściowych są zgodne z normą PN-EN 61131.

Tabele 3.6.1 i 3.6.2 przedstawiają parametry wejść cyfrowych w zależności od wersji przekaźnika.

Tab. 3.6.1. Parametry wejść przekaźnika programowalnego **NEED-24DC-xx-8-...**

Napięcie zasilania	Wejście	Zakres sygnałów wejściowych		Prąd wejściowy	Rezystancja wejściowa
		Wej. wyłącz.	Wej. załącz.		
V	nr	V	V	mA	kΩ
24 DC	I1..I6	-3..5	15..30	3.3	7.44
	I7..I8	-3..5	15..30	2	12.36
12 DC	I1..I6	-1..4	8..26	3.3	3.65
	I7..I8	-1..4	8..26	1.1	10.92

Tab. 3.6.2. Parametry wejść przełącznika programowalnego **NEED-24DC-xx-16-...**

Napięcie zasilania	Wejście	Zakres sygnałów wejściowych		Prąd wejściowy	Rezystancja wejściowa
		Wej. wyłącz.	Wej. załącz.		
V	nr	V	V	mA	kΩ
24 DC	I1..I13	-3..5	15..30	3.3	7.44
	I14..I16	-3..5	15..30	2	12.36
12 DC	I1..I13	-1..4	8..26	3.3	3.65
	I14..I16	-1..4	8..26	3.3	10.92

Jeśli wejścia I14..I16 zostaną skonfigurowane jako prądowe to ich impedancja wynosi 49 Ω.



Uwaga: Podanie pomiędzy zaciski wejściowe I1..In a 0V napięcia wyższego od maksymalnie dopuszczalnego może spowodować zniszczenie obwodów wejściowych przełącznika programowalnego.

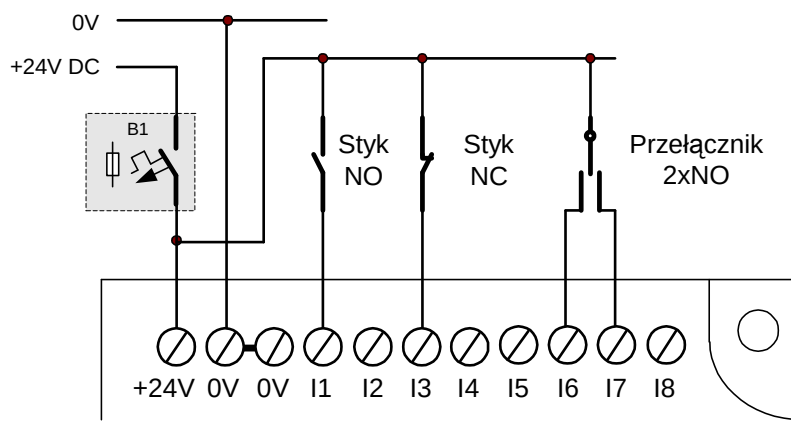


Uwaga: Podanie na wejścia prądowe prądu przewyższającego dopuszczalny może spowodować uszkodzenie obwodów wejściowych.

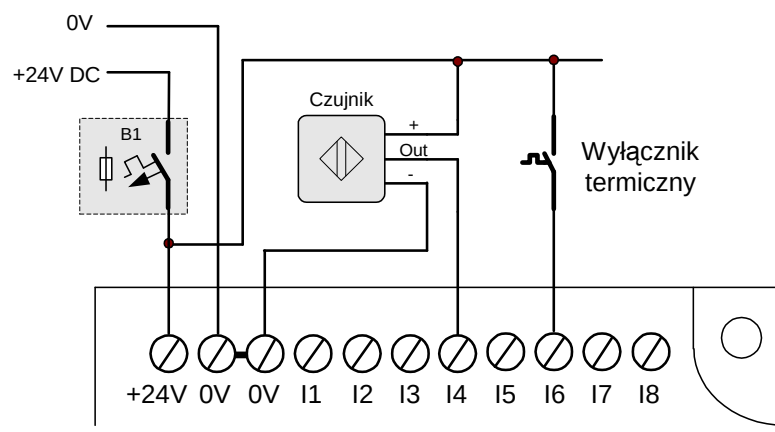


Wejścia mają charakter rezystancyjny.

Wejścia I7 i I8 dla wersji DC NEED..-xx-8-.. oraz I14, I15, I16 dla wersji NEED..-xx-16-.. mogą być wykorzystywane jako dyskretne lub analogowe – zależy to od sposobu ich wykorzystania w programie.



Rys. 3.6.1. Podłączenie wejść – elementy stykowe.



Rys. 3.6.2. Podłączenie wejść – czujnik zbliżeniowy, styk wyłącznika termicznego.

3.7. Podłączenie wejść analogowych AC



Niebezpieczeństwo porażenia prądem – w przypadku zamiany przewodu zerowego N z fazowym L lub braku podłączenia przewodu N może występować na zaciskach napięcie o wysokości napięcia zasilającego.



Wejścia analogowe nie są izolowane galwanicznie od sieci zasilającej przełącznik.

Zakres napięć sygnałów wejściowych dla wejść analogowych wynosi 0 do 255 V AC; dokładność +/- 3% wartości zakresu pomiarowego.



Dla prawidłowego funkcjonowania pomiaru analogowego nie jest konieczna zgodność fazy i częstotliwości mierzonego przebiegu z napięciem zasilającym przełącznik.



Wejścia analogowe, które będą wykorzystywane także jako cyfrowe, należy łączyć do tej samej fazy, co napięcie zasilające przełącznik programowalny NEED.

Wejścia analogowe mogą być używane jako wejścia dyskretne – wtedy stosuje się zasady podłączania jak dla wejść dyskretnych – patrz wyżej.

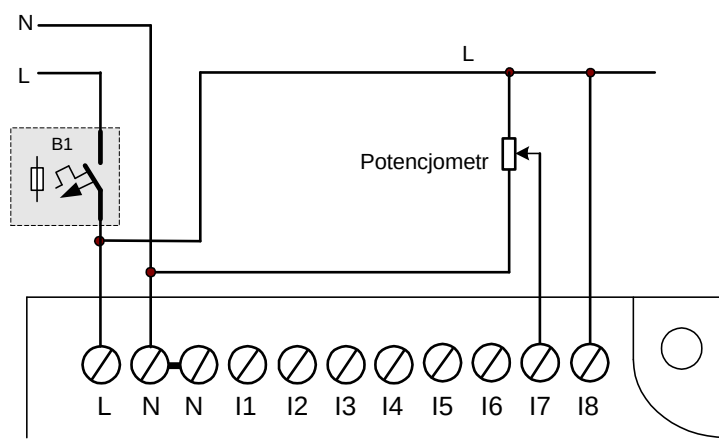
Pomiar analogowy dokonywany jest za pomocą układu uśredniającego. Wynik wskazywany jest w wartości skutecznej.

Wejścia analogowe, w związku z uśrednianiem, są mierzone z opóźnieniem.

Napięcie wejściowe (mierzone) musi być przez chwilę stabilne, aby pomiar był dokładny.

Wejścia analogowe dla wersji NEED-230AC-xx-8-4

W przełącznikach tego typu są to ostatnie dwa wejścia I7 i I8.

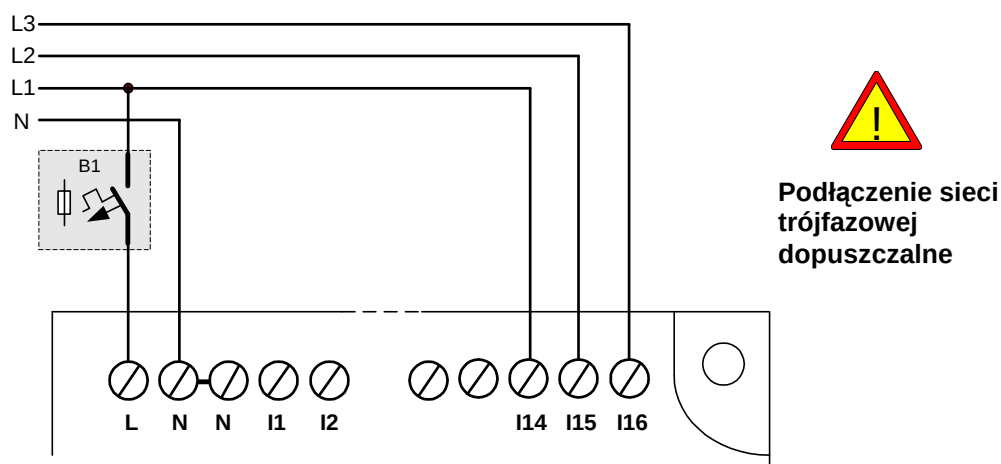


Podłączenie sieci trójfazowej niedopuszczalne

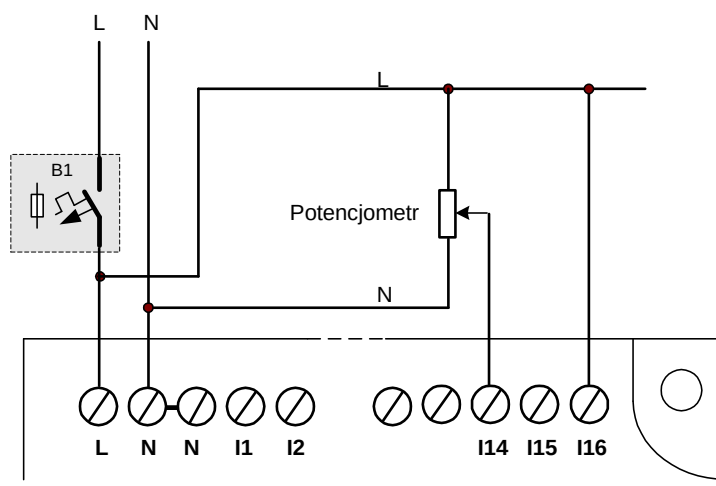
Rys. 3.7.1. Wejścia analogowe – potencjometr, kontrola napięcia sieci - NEED-230AC-01-8-..

Wejścia analogowe dla wersji NEED-230AC-xx-16-8

W przekaźnikach tego typu są to ostatnie trzy wejścia I14, I15, I16.



Rys. 3.7.2. Wejścia analogowe - NEED-230AC-01-16-..



Rys. 3.7.3. Wejścia analogowe – podłączenie potencjometru.



W wersji NEED-230AC-... niedostępne jest ładowanie czasów dla *Timerów* i progów dla *Liczników*.



Uwaga: Pamiętaj o odpowiednim doborze podłączanych elementów pod względem mocy oraz znamionowego napięcia pracy.



Uwaga: Pamiętaj, że wejście analogowe pobiera prąd. Może to znacznie zafałszować wynik pomiaru, jeżeli źródło napięcia mierzonego ma zbyt dużą impedancję wewnętrzną.



Uwaga: Elementy takie jak potencjometry, przełączniki itp. powinny być starannie zaizolowane ze względu na ryzyko porażenia prądem.

3.8. Podłączenie wejść analogowych 220V DC



Niebezpieczeństwo porażenia prądem – może występować na zaciskach napięcie o wysokości napięcia zasilającego.



Wejścia analogowe nie są izolowane galwanicznie od sieci zasilającej przełącznik.

Zakres napięć sygnałów wejściowych dla wejść analogowych wynosi 0 do 255 V DC (krok 1V) ; dokładność +/- 2% wartości zakresu pomiarowego.

Wejścia analogowe mogą być używane jako wejścia dyskretne – wtedy stosuje się zasady podłączania jak dla wejść dyskretnych – patrz wyżej.

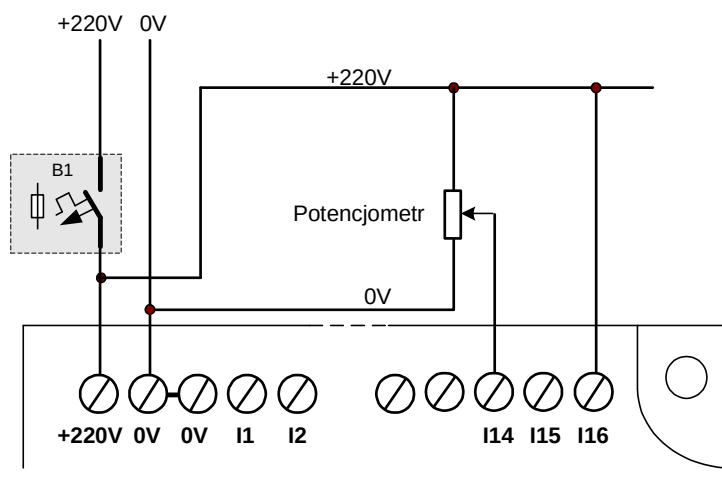
Napięcie wejściowe (mierzone) musi być przez chwilę stabilne, aby pomiar był dokładny.

Wejścia analogowe dla wersji NEED-220DC-xx-8-4

W przełącznikach tego typu są to ostatnie dwa wejścia I7 i I8.

Wejścia analogowe dla wersji NEED-220DC-xx-16-8

W przełącznikach tego typu są to ostatnie trzy wejścia I14, I15, I16.



Rys. 3.8.1. Wejścia analogowe NEED-220DC-11-16-8.



Uwaga: Pamiętaj o odpowiednim doborze podłączanych elementów pod względem mocy oraz znamionowego napięcia pracy.



Uwaga: Pamiętaj, że wejście analogowe pobiera prąd. Może to znacznie zafałszować wynik pomiaru, jeżeli źródło napięcia mierzonego ma zbyt dużą impedancję wewnętrzną.



Uwaga: Elementy takie jak potencjometry, przełączniki itp. powinny być starannie zaizolowane ze względu na ryzyko porażenia prądem.

3.9. Podłączenie wejść analogowych 24V (12V) DC

Poziom sygnałów wejściowych dla wejść analogowych skonfigurowanych jako napięciowe wynosi 0–25,5V DC (krok 0,1V) lub 0 – 12,75V (krok 0,05V).

Poziom sygnałów wejściowych dla wejść analogowych skonfigurowanych jako prądowe wynosi 0–51mA (krok 0,2mA) lub 0 – 25,5mA (krok 0,1mA).

Dokładność +/- 2% wartości zakresu pomiarowego.

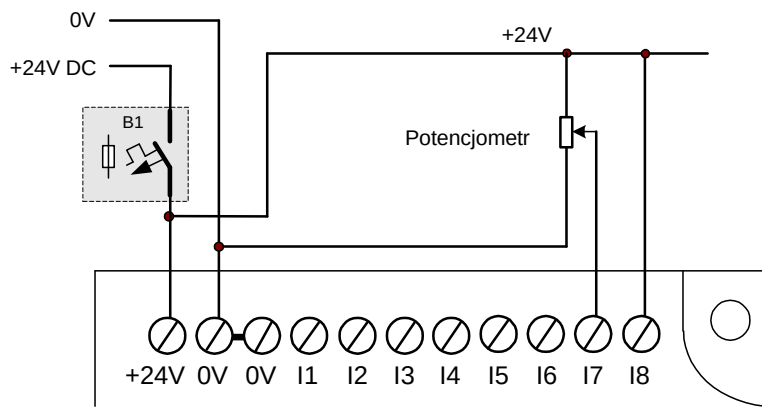
Rozdzielczość wejść analogowych: 8 bitów.



Wejścia analogowe mogą być używane jako wejścia cyfrowe – wtedy stosuje się zasady podłączania jak dla wejść dyskretnych – patrz pkt. 3.4.

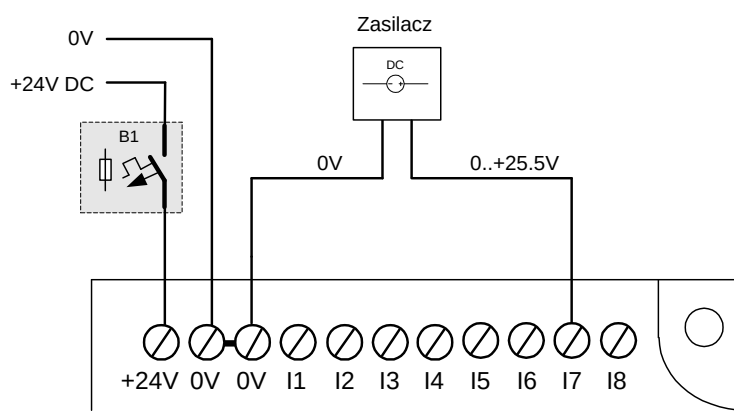
Wejścia analogowe to I7 i I8 dla wersji DC NEED..-xx-8-.. oraz I14, I15, I16 dla wersji NEED..-xx-16-...

Poniżej przedstawiono układ do zadawania napięcia na wejściu I7 oraz kontroli napięcia zasilania za pomocą wejścia I8 połączonego z „+” zasilania przekaźnika dla wersji NEED-24DC-x1-8-... Przy takim połączeniu potencjometrem możemy regulować nie tylko progi komparatorów, ale także zadawać wartości czasu dla *Timerów* oraz ustawiać progi *Liczników*.



Rys. 3.9.1. Wejścia analogowe – potencjometr.

W zależności od wersji możliwy jest odczyt napięcia zewnętrznego w zakresie 0-25.5V lub 0-12.5V. Układ połączeń dla zewnętrznego źródła napięcia dla wersji NEED-24DC-xx-8-.. przedstawiono na rys. 3.9.2.



Rys. 3.9.2. Wejścia analogowe – zakres.



Uwaga: Pamiętaj o odpowiednim doborze podłączanych elementów pod względem mocy oraz znamionowego napięcia pracy.

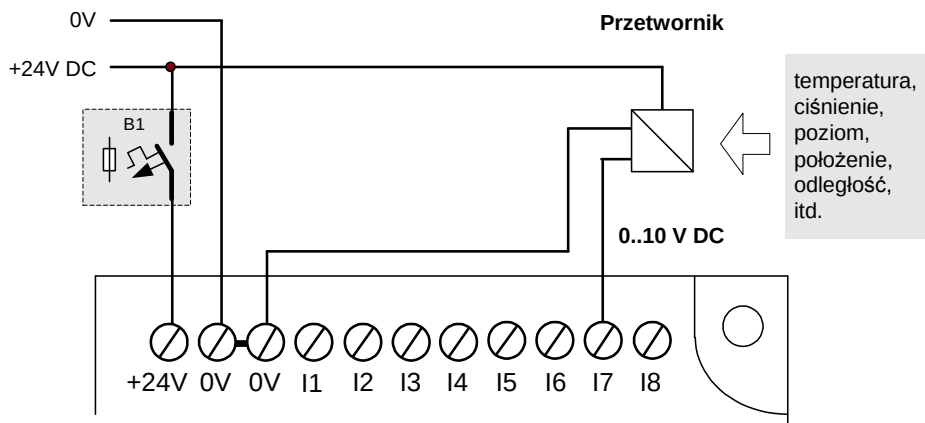


Uwaga: Pamiętaj, że wejście analogowe pobiera prąd. Może to znacznie zafałszować wynik pomiaru, jeżeli źródło napięcia mierzonego ma zbyt dużą impedancję wewnętrzną.

Przetwornik 0..10 V DC.

Do wejść analogowych można podłączać różnego typu przetworniki wielkości elektrycznych (napięcie, prąd, częstotliwość) lub nieelektrycznych (temperatura, ciśnienie, odległość, siła itp.), które posiadają standardowe wyjście analogowe napięciowe lub prądowe.

Dla przetwornika napięciowego wytwarzającego napięcie od 0 do 10V dla min. i max. przetwarzanej wielkości otrzymamy 100 punktową charakterystykę przetwarzania.



Rys. 3.9.3. Wejścia analogowe – przetwornik 0..10V dla wersji NEED-24DC-01-8-4

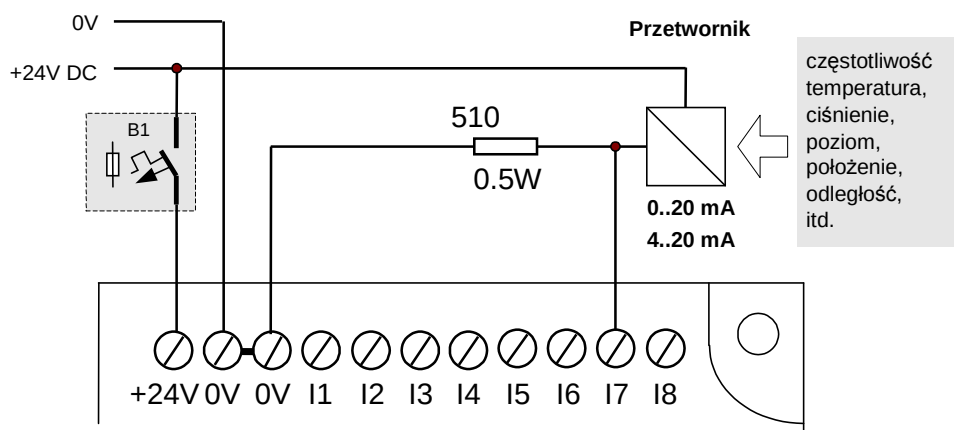
Przetwornik 0..20 mA.

Wersje NEED-24DC-xx-8-4 i NEED-12DC-xx-8-4 nie posiadają wbudowanego przetwornika prąd / napięcie.

W celu wykorzystania przetwornika z wyjściem prądowym w zakresie 0..20mA lub 4..20mA należy zastosować prosty układ przetwarzający prąd na napięcie. Uzyskamy to mierząc spadek napięcia na rezystorze 510 Ω , będącym obciążeniem przetwornika. Spadek napięcia jest proporcjonalny do wielkości przepływającego prądu w zależności: 1mA = ~ 0.5V. W obliczeniach uwzględniono rezystancję wewnętrzną wejścia analogowego przekaźnika.

Charakterystyczne punkty przetwarzania dla wersji 24V DC to:

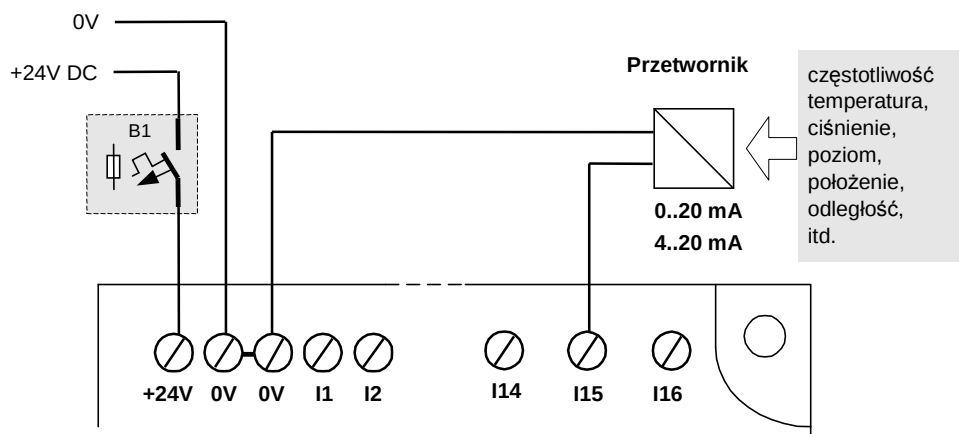
- 1mA → ~0,5V
- 4mA → ~1,9V
- 10mA → ~4,9V
- 20mA → ~9,8V



Rys. 3.9.4. Wejścia analogowe – przetwornik 0..20mA dla wersji NEED-24DC-01-8-4

Wersje NEED-24DC-xx-16-8 i NEED-12DC-xx-16-8 posiadają wbudowany przetwornik prąd / napięcie.

Przetworniki z wyjściem prądowym można podłączać bezpośrednio do wejść przekaźnika AI14, AI15, AI16 po ich uprzednim skonfigurowaniu w programie PC Need jako wejścia prądowe (I) i wgraniu ustawień do przekaźnika.



Rys. 3.9.5. Wejścia analogowe – przetwornik 20mA dla wersji NEED-24DC-11-16-8



Uwaga: Podanie pomiędzy zaciski wejść analogowych a 0V napięcia wyższego od maksymalnie dopuszczalnego może spowodować zniszczenie obwodów wejściowych przekaźnika programowalnego.



Uwaga: Nie należy podawać napięcia na wejścia AI14, AI15, AI16 ustawione jako prądowe.



Uwaga: Podłączenie źródła napięcia na wejścia AI14, AI15, AI16 ustawione jako prądowe może doprowadzić do ich uszkodzenia. Maksymalne napięcie na wejściu prądowym dla I=51mA wynosi 2.5V.



Uwaga: Podanie napięcia ze źródła o wyższej niż 51mA wydajności na wejścia AI14, AI15, AI16 ustawione jako prądowe, spowoduje zadziałanie zabezpieczenia wewnętrznego. Przekaźnik będzie cyklicznie próbował wejścia.

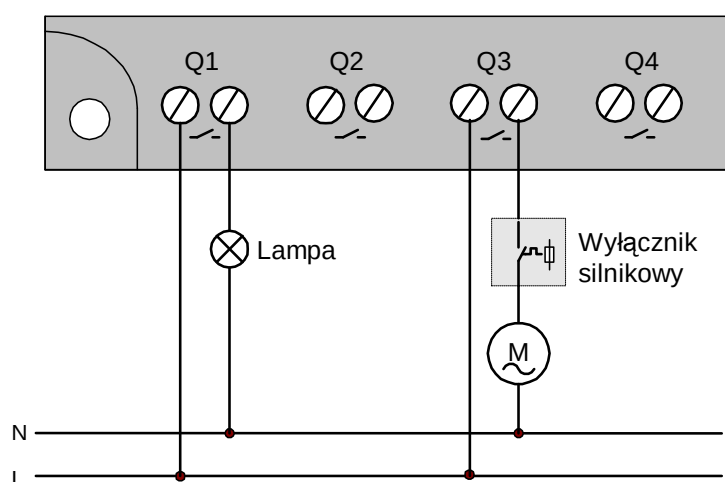
3.10. Podłączenie wyjść

Zaciski wyjściowe są połączone ze stykami wewnętrznych przekaźników elektromagnetycznych Q1..Qn. Wersja NEED..-xx-8-4R to 4 wyjścia przekaźnikowe. Wersja rozszerzona NEED..-xx-16-8R zawiera 8 wyjść przekaźnikowych.

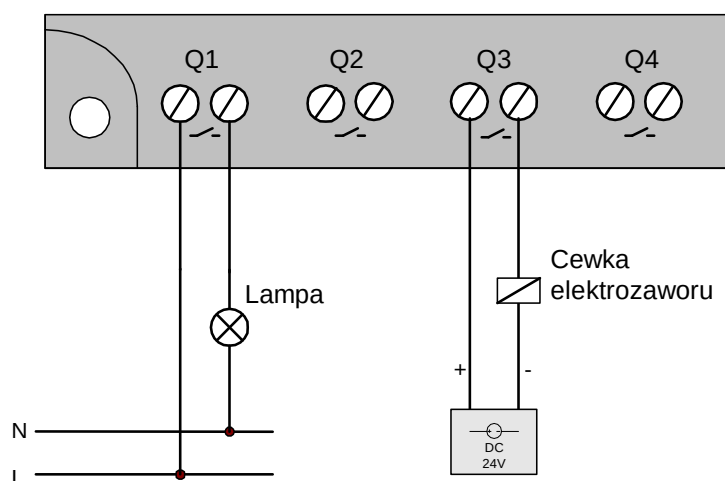
Wyjścia są bezpotencjałowe i izolowane galwanicznie od reszty układu i względem siebie – możemy zbudować niezależne układy sterowania.

Obciążalność 1 wyjścia – patrz specyfikacja techniczna – 230V, 10A przy rezystancyjnym charakterze obciążenia –kategoria AC1

Należy pamiętać o odpowiednim zabezpieczeniu obwodów wyjściowych (bezpiecznik) sterowanych przez przekaźnik w zależności od mocy i charakteru obciążenia, aby nie przekroczyć wartości podanych w specyfikacji technicznej.



Rys. 3.10.1. Wyjścia przekaźnikowe – zasilanie sieciowe 230V AC -wersja NEED..-x1-8-4R.



Rys. 3.10.2. Wyjścia przekaźnikowe – różne obwody zewnętrzne -wersja NEED..-x1-8-4R.

3.11. Podłączenie zasilania 115/230V AC



Napięcie zasilające przekaźnik jest niebezpieczne dla życia!

Od jakości wykonanych połączeń zależy bezpieczeństwo użytkownika!

Należy przestrzegać prawidłowego podłączenia przewodów doprowadzających napięcie zasilające – przewód fazowy L i przewód zerowy N.



Zamienienie miejscami przewodów na wejściach zasilających tzn. podłączenie fazy L na zacisk N oraz przewodu neutralnego N na zacisk wejściowy L przekaźnika programowalnego może spowodować wystąpienie niebezpiecznych napięć na zaciskach wejściowych I1..In oraz portach komunikacyjnych, a także brak wykrywania stanów logicznych.



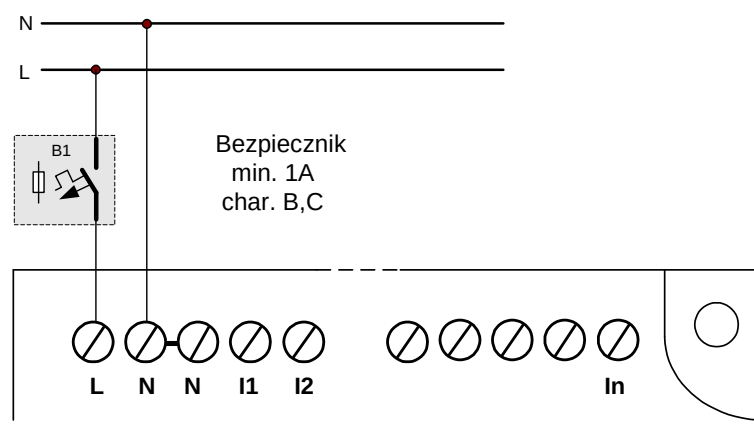
Znamionowe napięcie zasilania **115/230V AC 50/60 Hz**



Podanie napięcia międzyfazowego 400V AC pomiędzy zaciski L i N spowoduje zniszczenie przekaźnika programowalnego

Należy zabezpieczyć przekaźnik programowalny bezpiecznikiem o prądzie nominalnym 1A. Oczywiście zabezpieczenie nie może mieć zbyt dużej wartości bo nie spełni swojej roli – zalecane maksimum to 6A.

Jeśli zabezpieczenie będzie wspólne także dla wejść i wyjść, to należy uwzględnić prąd zabezpieczenia zasilania przekaźnika programowalnego.



Rys. 3.11.1. Zasilanie przekaźnika programowalnego NEED-230AC-xx-..”.

3.12. Podłączenie zasilania 220V DC



Napięcie zasilające przekaźnik jest niebezpieczne dla życia!

Od jakości wykonanych połączeń zależy bezpieczeństwo użytkownika!



Należy przestrzegać prawidłowego podłączenia przewodów doprowadzających napięcie zasilające – przewód +220V DC i przewód 0V.



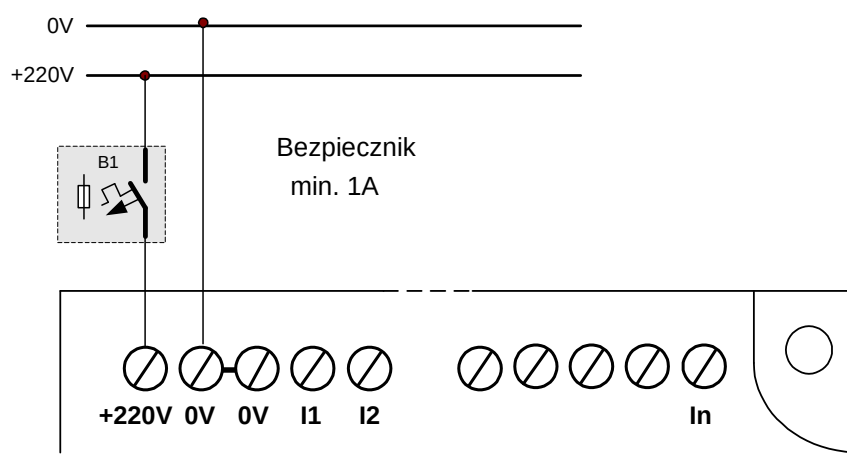
Uwaga: Podanie pomiędzy zaciski zasilające +220V a 0V napięcia wyższego od maksymalnie dopuszczalnego może spowodować zniszczenie przekaźnika programowalnego.



Znamionowe napięcie zasilania **220V DC**.

Należy zabezpieczyć przekaźnik programowalny bezpiecznikiem o prądzie nominalnym 1A. Oczywiście zabezpieczenie nie może mieć zbyt dużej wartości bo nie spełni swojej roli – zalecane maksimum to 6A.

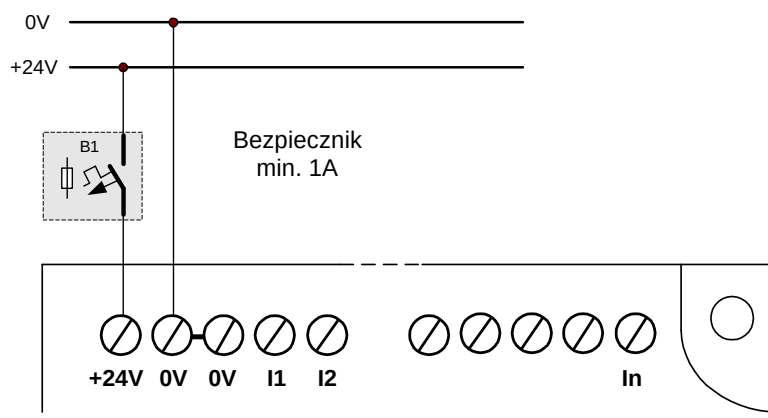
Jeśli zabezpieczenie będzie wspólne także dla wejść i wyjść, to należy uwzględnić prąd zabezpieczenia zasilania przekaźnika programowalnego.



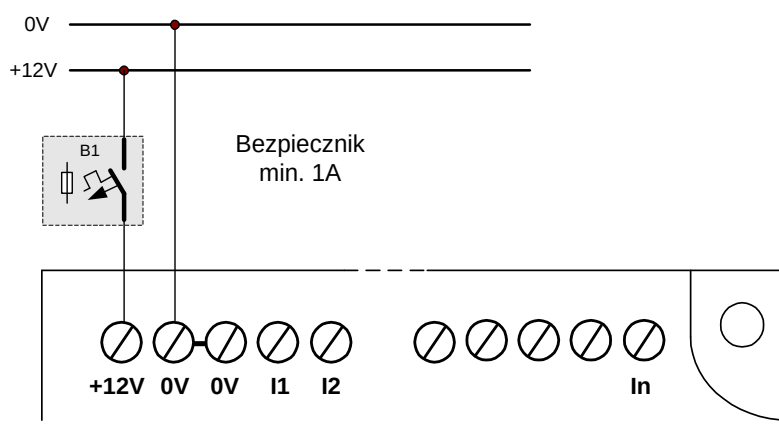
Rys. 3.12.1. Zasilanie przekaźnika programowalnego NEED-220DC-x1-..".

3.13. Podłączenie zasilania 24V (12V) DC

Bezpiecznik zabezpieczający przewody powinien mieć wartość powyżej 1A, gdyż w momencie włączania występuje uderzenie prądowe w wyniku ładowania wewnętrznego kondensatora w zasilaczu przekaźnika.



Rys. 3.13.1. Zasilanie przekaźnika programowalnego NEED-24DC-xx-...



Rys. 3.13.2. Zasilanie przekaźnika programowalnego NEED-12DC-xx-...



Uwaga: Podanie pomiędzy zaciski zasilające +24V (+12V) a 0V napięcia wyższego od maksymalnie dopuszczalnego może spowodować zniszczenie przekaźnika programowalnego.

4. ZASOBY PRZekaźNIKA

Sterowniki programowalne są urządzeniami, które w swojej strukturze obejmują dwa podstawowe elementy: jednostkę centralną (CPU) wraz z pamięcią oraz peryferia wejścia/wyjścia. Oczywiście do pełnej funkcjonalności potrzebne jest również jakieś urządzenie programujące oraz kabel do komunikacji ze sterownikiem. Wszystkie te cechy posiada również przekaźnik programowalny NEED.

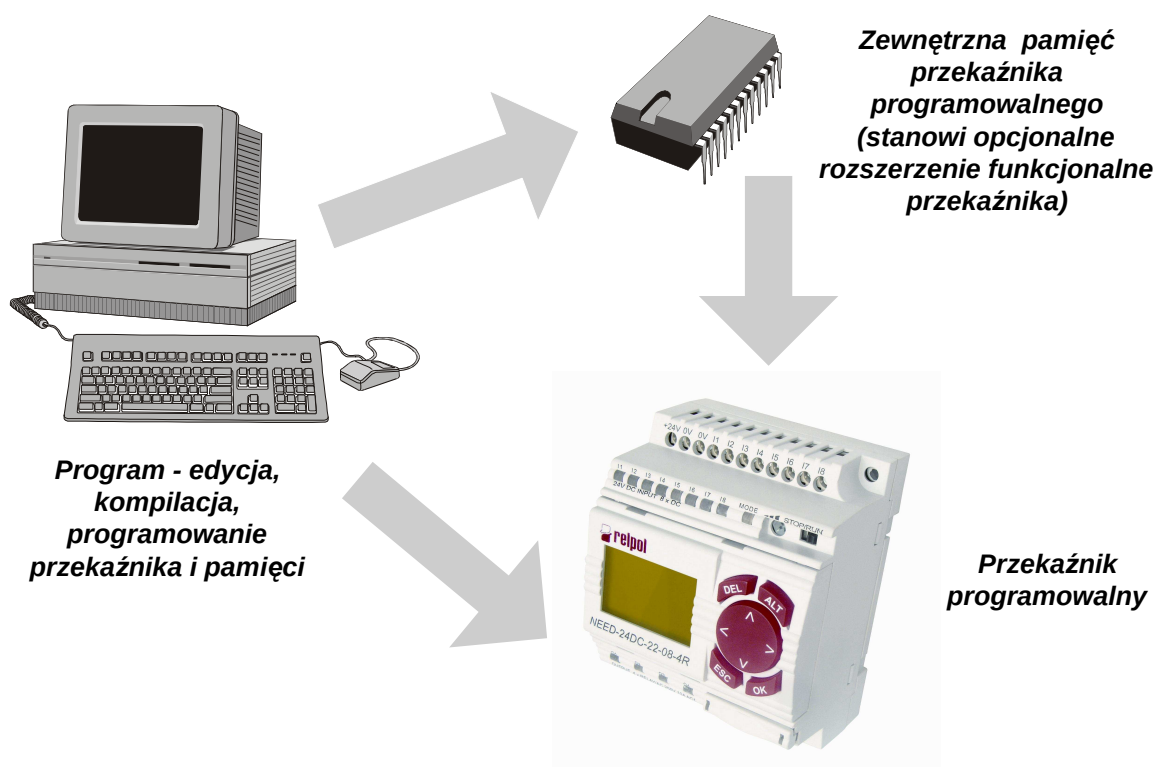
4.1. System przekaźnika programowalnego NEED

W skład systemu przekaźnika programowalnego wchodzi:

Program do edycji, kompilacji oraz ładowania programu do pamięci przekaźnika.

Zewnętrzna pamięć przekaźnika (nie jest ona wymagana - umożliwia łatwe przenoszenie programu pomiędzy PC i przekaźnikiem).

Przekaźnik programowalny.



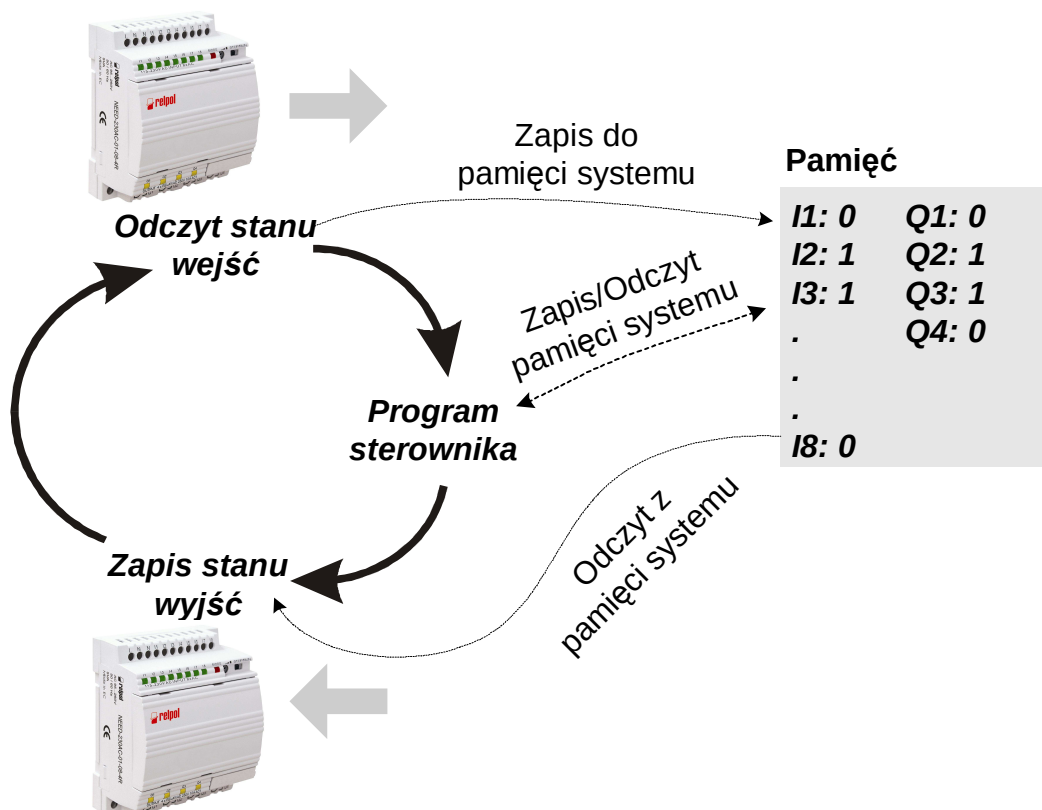
Rys. 4.1.1. System przekaźnika programowalnego NEED.

4.2. Cykl programu

Aby można było realizować różne aplikacje wykorzystując przekaźnik programowalny należy przede wszystkim stworzyć odpowiedni program i umieścić go w pamięci sterownika. Po uruchomieniu sterownik zaczyna przetwarzanie programu – od pierwszej instrukcji do ostatniej, po czym cykl jest powtarzany.

Na początku każdego cyklu stany wejść są zapisywane do specjalnych obszarów pamięci. W trakcie programu następuje odwołanie się nie do bezpośrednich stanów wejść/wyjść, ale do ich kopii umieszczonych w pamięci odwzorowującej proces. Podobnie jest z sygnałami wyjściowymi. Sterownik zapamiętuje te stany w pamięci odwzorowania procesu i dopiero z końcem każdego cyklu następuje przepisanie tychże stanów do wyjść przekaźnika.

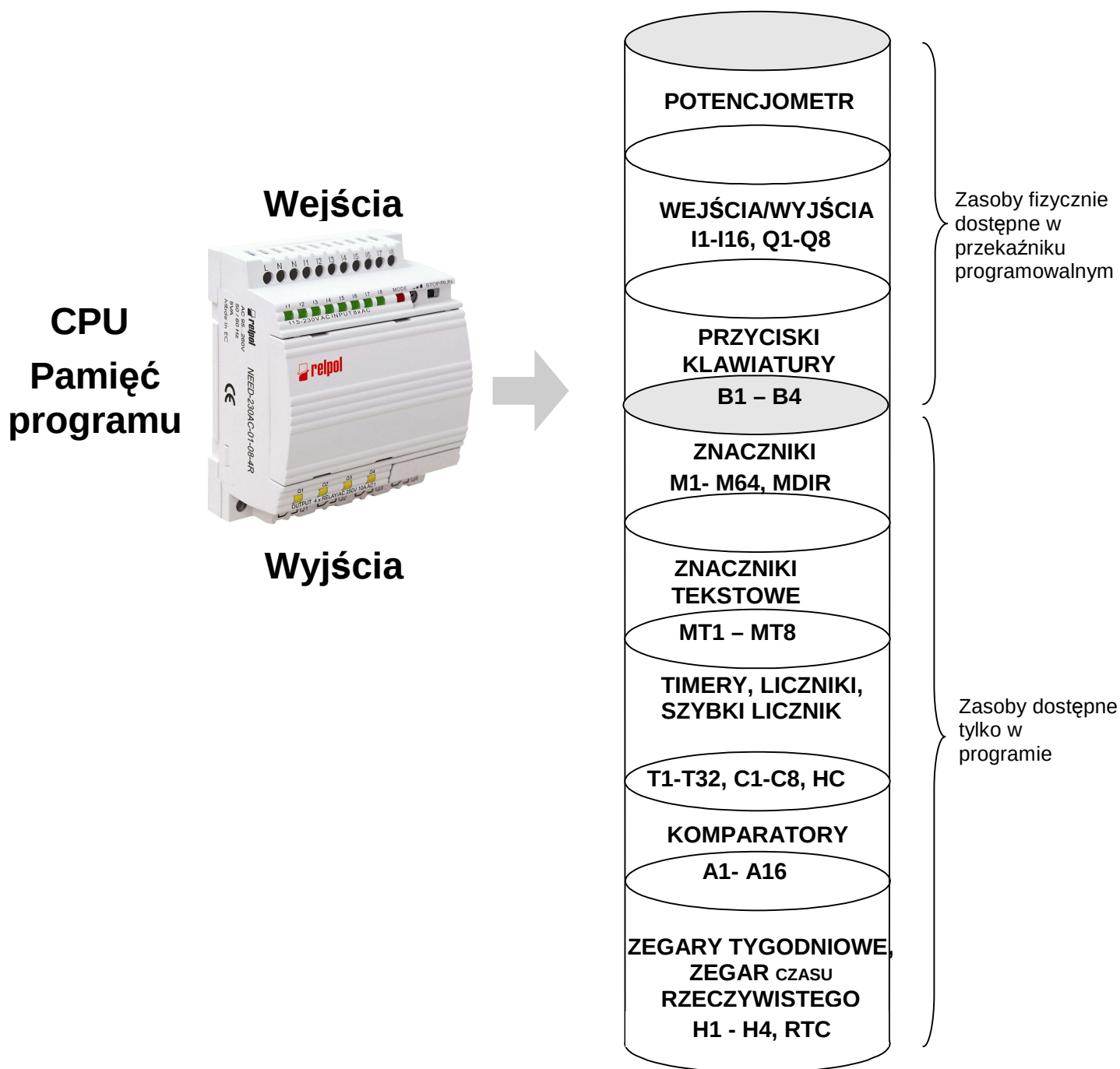
Cykl pracy sterownika przedstawiono na rys. 4.2.1.



Rys. 4.2.1. Cykl pracy sterownika.

Do prawidłowego zrozumienia i dobrego programowania przekaźnika NEED niezbędna jest znajomość jego wewnętrznych zasobów.

4.3. Zasoby przekaźnika programowalnego NEED



Rys. 4.3.1. Zasoby przekaźnika programowalnego NEED.

Do komunikacji ze światem zewnętrznym przekaźnik NEED używa wejść i wyjść. Praktycznie są to jedyne zasoby, które użytkownik postrzega z zewnątrz i w oparciu o które mogą być budowane nawet skomplikowane aplikacje. Jednak o prawdziwej sile każdego sterownika stanowią także jego zasoby wewnętrzne – „niewidzialne” z zewnątrz, do których dostęp ma jedynie programista. Na rys. 4.3.1. symbolicznie przedstawiono zasoby przekaźnika programowalnego, natomiast w tabeli 4.3. wyszczególniono ilość poszczególnych elementów zawartych w strukturze systemowej przekaźnika.

Odpowiednie używanie i wykorzystywanie zasobów przekaźnika programowalnego NEED zależy od użytkownika. Poniżej przedstawiamy opis poszczególnych elementów oraz sposoby zapisu dla różnych języków programowania.

Tab. 4.3. Zasoby przekaźnika programowalnego NEED

Nazwa	NEED-..-x1- 08- 4	NEED-..-11-16- 8
Wejścia cyfrowe „I”	I1 – I8	I1 – I16
Wejścia analogowe	I7 – I8	I14 – I16
Wyjścia cyfrowe „Q” przełącznikowe typu NO	Q1 – Q4	Q1 – Q8
Komparatory „A”	A1 – A8	A1 – A12
Znaczniki „M”	M1 – M16	M1 – M16
Znacznik „MDIR” określający kierunek podłączenia faz L1, L2, L3	–	MDIR
Timery „T”	T1 – T8	T1 – T16
Liczniki „C”	C1 – C8	C1 – C8
Szybki Licznik „HC” do 20kHz		HC1
Zegar Czasu Rzeczywistego	Tak	Tak
Automatyczna zmiana czasu lato/zima w różnych strefach czasowych	Nie	Tak
Zegary Tygodniowe „H”	H1 – H4	H1 – H4

Tab. 4.3. Zasoby przekaźnika programowalnego NEED – c.d.

Nazwa	NEED-...-22- 08- 4	NEED-...-22-16- 8
Wejścia cyfrowe „I”	I1 – I8	I1 – I16
Wejścia analogowe	I7 – I8	I14 – I16
Wyjścia cyfrowe „Q” przełącznikowe typu NO	Q1 – Q4	Q1 – Q8
Komparatory „A”	A1 – A16	A1 – A16
Znaczniki „M”	M1 – M64	M1 – M64
Znaczniki tekstowe „MT”	MT1 – MT8	MT1 – MT8
Znacznik „MDIR” określający kierunek podłączenia faz L1, L2, L3		MDIR
Timery „T”	T1 – T32	T1 – T32
Liczniki „C”	C1 – C8	C1 – C8
Szybki Licznik „HC” do 20kHz	HC1	HC1
Zegar Czasu Rzeczywistego	1	1
Automatyczna zmiana czasu lato/zima w różnych strefach czasowych	Tak	Tak
Zegary Tygodniowe „H”	H1 – H8	H1 – H8
Przyciski klawiatury	B1 – B4	B1 – B4

4.4. Wejścia cyfrowe

Każde z wejść może być skonfigurowane jako normalnie otwarte i normalnie zamknięte. Zasoby te reprezentują fizyczne wejścia sterownika programowalnego.

4.4.1. Wejścia cyfrowe normalnie otwarte

Symbole wejść cyfrowych normalnie otwartych.

STL	LAD
A I1 lub O I1 lub X I1	I1 

SYMBOL: **In**, gdzie n oznacza numer wejścia n=1..8 dla NEED-...-08-..
n=1..16 dla NEED-...-16-..

STANY LOGICZNE:

'1' – Jest napięcie zasilające na wejściu.

'0' – Brak napięcia zasilającego na wejściu.

4.4.2. Wejścia cyfrowe normalnie zamknięte

Symbole wejść cyfrowych normalnie zamkniętych.

STL	LAD
AN I1 lub ON I1 lub XN I1	I1 

SYMBOL: **In**, gdzie n oznacza numer wejścia n=1..8 dla NEED-...-08-..
n=1..16 dla NEED-...-16-..

STANY LOGICZNE:

'0' – Jest napięcie zasilające na wejściu.

'1' – Brak napięcia zasilającego na wejściu.

4.5. Wyjścia cyfrowe

Wyjścia cyfrowe mogą być różnego typu. Należy jednak zawsze pamiętać, iż mamy do dyspozycji maksymalnie 8 fizycznych wyjść!

4.5.1. Wyjścia cyfrowe normalne

Symbole wyjść cyfrowych normalnych.

STL	LAD
= Q2	Q2 

SYMBOL: **Qn**, gdzie n oznacza numer wyjścia n=1..4 dla NEED-...-08-..
n=1..8 dla NEED-...-16-..

STANY LOGICZNE:

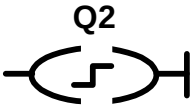
'1' – styki zwarte.

'0' – styki rozwarte.

Wyjście tego typu działa jak zwykły przekaźnik – tzn. podanie napięcia na cewkę powoduje zadziałanie przekaźnika.

4.5.2. Wyjścia cyfrowe impulsowe

Symbole wyjść cyfrowych impulsowych.

STL	LAD
FP Q2	

SYMBOL: **Qn**, gdzie n oznacza numer wyjścia n=1..4 dla NEED-...-08-..
n=1..8 dla NEED-...-16-..

STANY LOGICZNE:

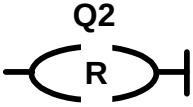
'1' – Jeżeli poprzedni stan wynosił 0 i wystąpiło dodatnie sterujące zbocze logiczne.

'0' – Jeżeli poprzedni stan wynosił 1 i wystąpiło dodatnie sterujące zbocze logiczne.

Wyjście to działa jak przerzutnik, który wyzwalany zboczem narastającym, zmienia stan swojego wyjścia na przeciwny.

4.5.3. Wyjścia cyfrowe kasujące

Symbole wyjść cyfrowych kasujących.

STL	LAD
R Q2	

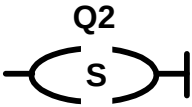
SYMBOL: **Qn**, gdzie n oznacza numer wyjścia n=1..4 dla NEED-...-08-..
n=1..8 dla NEED-...-16-..

STANY LOGICZNE:

'0' – Jeżeli wystąpił stan sterujący '1'.

4.5.4. Wyjścia cyfrowe ustawiające

Symbole wyjść cyfrowych ustawiających.

STL	LAD
S Q2	

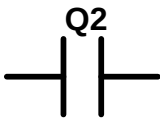
SYMBOL: **Qn**, gdzie n oznacza numer wyjścia n=1..4 dla NEED-...-08-..
n=1..8 dla NEED-...-16-..

STANY LOGICZNE:

'1' – Jeżeli wystąpił stan sterujący '1'.

4.5.5. Wyjścia cyfrowe normalne wykorzystane do dalszego sterowania

Symbole wyjść cyfrowych normalnych wykorzystywanych do dalszego sterowania.

STL	LAD
A Q2 lub O Q2 lub X Q2	

SYMBOL: **Qn**, gdzie n oznacza numer wyjścia n=1..4 dla NEED-...-08-..
n=1..8 dla NEED-...-16-..

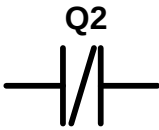
STANY LOGICZNE:

'1' – Jeżeli stan wyjścia fizycznego wynosi '1'.

'0' – Jeżeli stan wyjścia fizycznego wynosi '0'.

4.5.6. Wyjścia cyfrowe zanegowane wykorzystane do dalszego sterowania

Symbole wyjść cyfrowych wykorzystywanych do dalszego sterowania.

STL	LAD
AN Q2 lub ON Q2 lub XN Q2	

SYMBOL: **Qn**, gdzie n oznacza numer wyjścia n=1..4 dla NEED-...-08-..
n=1..8 dla NEED-...-16-..

STANY LOGICZNE:

'1' – Jeżeli stan wyjścia fizycznego wynosi '0'.

'0' – Jeżeli stan wyjścia fizycznego wynosi '1'.

4.6. Znaczniki

4.6.1. Znaczniki

Znacznik (ang. *Marker*) jest elementem logicznym i traktowany jest jako zmienna wykorzystywana w programie. Posiada swój wewnętrzny stan '0' lub '1'.

Ze *Znacznikiem* nie możemy wiązać fizycznie określonego wejścia lub wyjścia, ale możemy go wykorzystać do łączenia logicznych struktur programu. Czyli *Znaczniki* możemy traktować jak 64 zarezerwowane bity, do których możemy się odnieść jak do wejść lub wyjść – tzn. podlegają one takim samym „operacjom” (instrukcjom) jak wejścia i wyjścia, ale nie mają fizycznej reprezentacji w postaci styków.

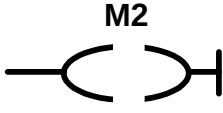
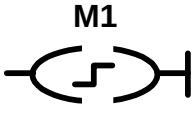
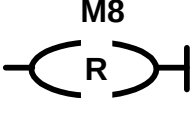
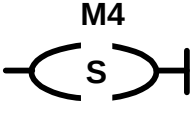
W składni instrukcji czy w symbolu graficznym zamiast litery I lub Q pojawia się symbol *Znacznika M*.

SYMBOL: **Mn**, gdzie n oznacza numer Markera.

STANY LOGICZNE:

'0' lub '1' w zależności od realizowanej funkcji w programie.

Symbol *Znaczników*.

STL	LAD
= M2	
FP M1	
R M8	
S M4	

SYMBOL: **Mn**, gdzie n oznacza numer Markera n=1..16 dla NEED-..x1-..

n=1..64 dla NEED-..-22-..

Symbole Znaczników c.d.

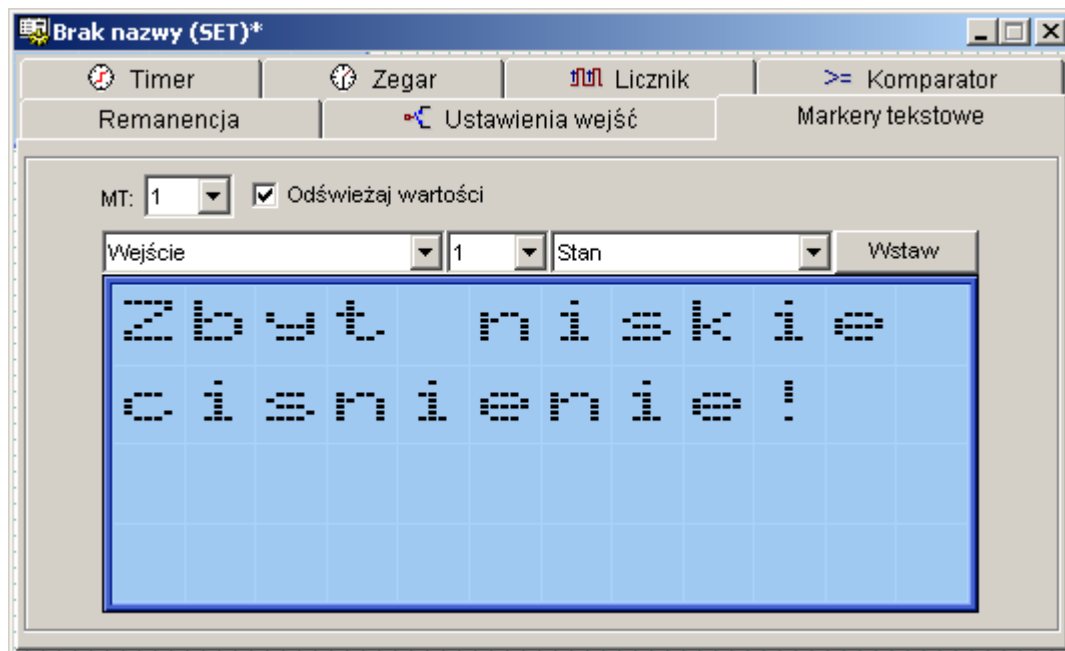
STL	LAD
A M2 lub O M2 lub X M2	
AN M9 lub ON M9 lub XN M9	

4.6.2. Znaczniki Tekstowe

Znaczniki Tekstowe wykorzystywane są do wyświetlania informacji i alarmów na wyświetlaczu LCD. Podlegają one takim samym regułom jak zwykłe Markery „M”.

SYMBOL: **MTn**, gdzie n oznacza numer Markera Tekstowego n=1..8 dla NEED-..22-..

Znaczniki MT konfiguruje się w „Ustawieniach”, gdzie do odpowiedniego Markera przypisuje się żądany tekst. Na rys. 4.6.2.1. przedstawiono przykładowy ekran konfiguracyjny, który został przyporządkowany do Znacznika MT1.



Rys.4.6.2.1. Okno konfiguracyjne Markera MT1.

4.6.3. Znacznik MDIR



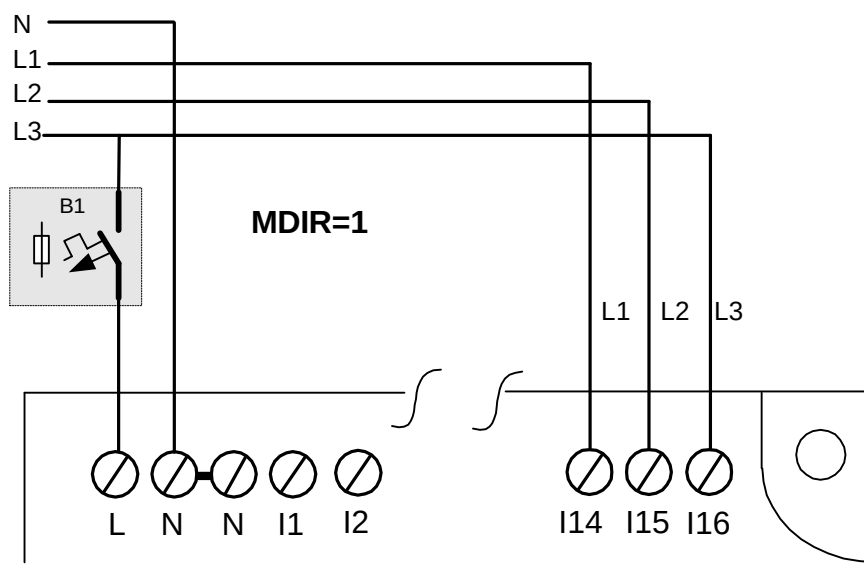
Dla wersji NEED-230VAC-...-16-8 oprócz wspomnianych wyżej 16 Znaczników istnieje jeszcze Znacznik MDIR.

Znacznik MDIR określa kierunek faz L1, L2, L3 podłączanych do wejść I14, I15, I16. Jeśli do tych wejść zostaną podłączone kolejno fazy L1 (do I14), L2 (do I15), L3 (do I16) to MDIR przyjmuje wartość '1', w przeciwnym wypadku MDIR ma wartość '0'.

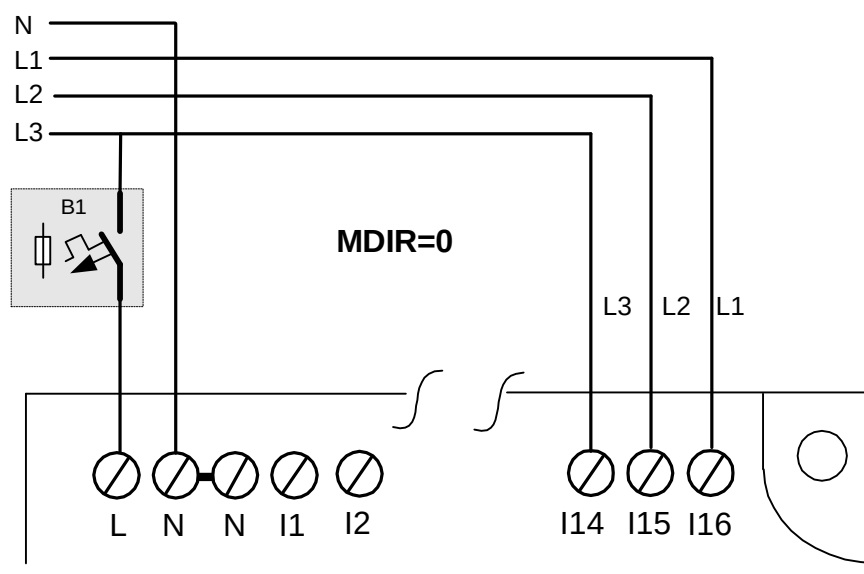
Znacznik MDIR jest tylko do odczytu.

Na rysunkach 4.6.3.1 i 4.6.3.2 przedstawiono przykładowe podłączenia, przy którym MDIR przyjmuje wartość kolejno '1' i '0'.

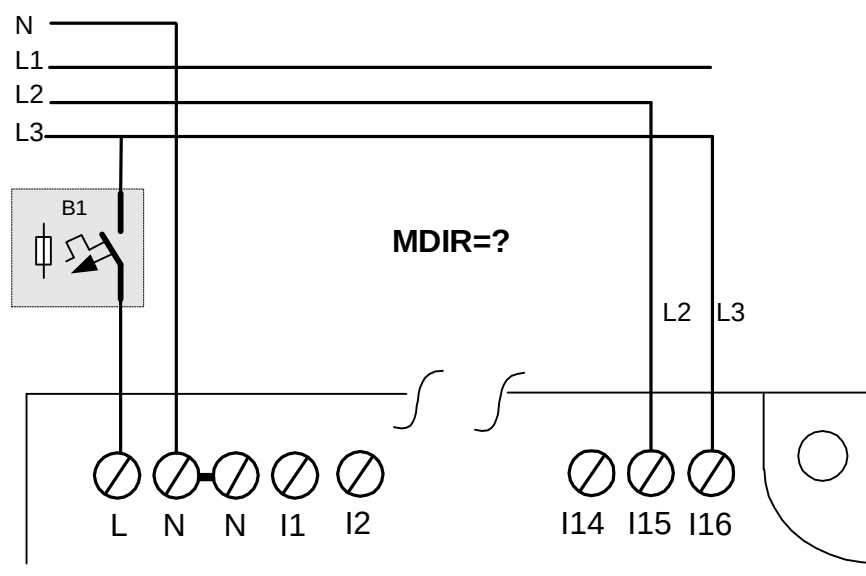
Jeśli do jednego z wejść analogowych I14, I15, I16 nie podłączymy przewodów sieciowych, wówczas Znacznik MDIR przyjmuje wartość losową – rysunek 4.6.3.3.



Rys.4.6.3.1. Podłączenie sieci trójfazowej, przy którym Znacznik MDIR przyjmuje wartość logiczną '1'.



Rys.4.6.3.2. Podłączenie sieci trójfazowej, przy którym Znacznik MDIR przyjmuje wartość logiczną '0'.



Rys.4.6.3.3. Podłączenie sieci trójfazowej, przy którym Znacznik MDIR przyjmuje wartości losowe '0' lub '1'.

4.7. Timery

SYMBOL: T_n, gdzie n – jest numerem *Timera* : n=1..8 dla NEED-..-x1-08-4
 n=1..16 dla NEED-..-11-16-8
 n=1..32 dla NEED-..-22-..

STANY LOGICZNE WEJŚĆ **TRIGGER**, **RESET**:

'0' lub '1' w zależności od realizowanej funkcji w programie.

STANY LOGICZNE WYJŚCIA:

'0' lub '1' w zależności od realizowanej funkcji w programie.

ZAKRESY CZASOWE:

Wartości odmierzanych czasów przedstawione są w tabeli 4.7.

Timer jest elementem czasowym, dzięki któremu możemy wykorzystywać sterowanie czasowe w przekaźniku programowalnym.

Każdy z 8 *Timerów* może być używany w jednej z konfiguracji:

- Załączenie z opóźnieniem (ON-DELAYED),
- Wyłączenie z opóźnieniem (OFF-DELAYED),
- Pojedynczy impuls (SINGLE PULSE),
- Impulsy (FLASHING).

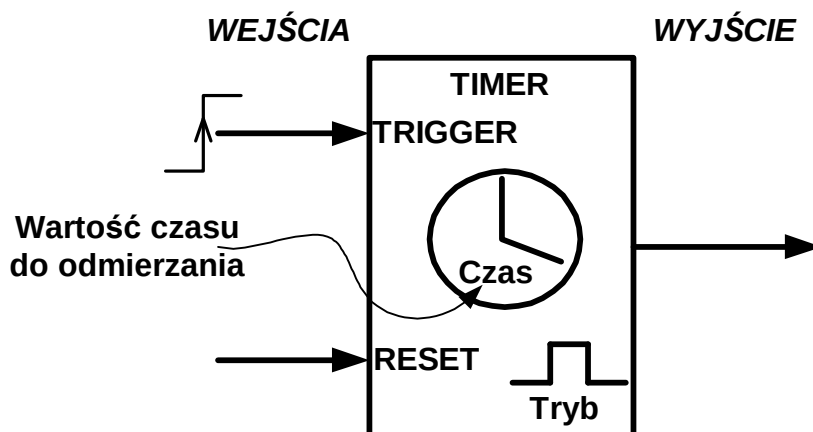
W strukturze logicznej *Timera* możemy wyróżnić wejścia, wyjście, tryb pracy oraz wartość czasu do odmierzania.

Wejścia i wyjścia *Timerów* można logicznie wiązać także poprzez sygnały bitowe (I,Q,M).

Wejścia.

W skład wejść wchodzi:

- wejście wyzwalające – TRIGGER (wejście to inicjuje działanie *Timera* np. rozpoczyna odmierzanie czasu)
- wejście resetujące – RESET (powoduje ustawienie wyjścia *Timera* w stan niski ('0') oraz zatrzymanie odmierzania czasu)



Rys. 4.7.1. Logiczna struktura Timera.

Czas do odmierzenia.

Czas do odmierzenia w *Timerach* ustawia się poprzez odpowiednie instrukcje ładujące (STL, LAD) lub za pomocą ustawień w pliku ustawień „*.set”. Zakresy czasowe *Timerów* przedstawiono w tabeli 4.7.

Tryb.

Rodzaj pracy *Timera* np. Załączenie z opóźnieniem, Pojedynczy impuls itp.

Tab. 4.7. Zakresy czasowe timerów.

Format czasu	Zakres	Krok
s.ms (sekundy.milisekundy)	0s.10ms – 99s.990ms	10ms
min.s (minuty.sekundy)	0min.1s – 99min.59s	1s
h.min (godziny.minuty)	0h.1min – 99h.59min	1min

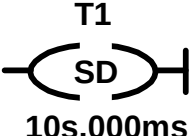
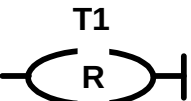
Wyjścia.

Wyjście *Timera* jest ustawiane lub resetowane w zależności od wyboru odpowiedniej funkcji czasowej (odpowiedniego typu *Timera*).

W programie wyjścia *Timerów* możemy używać tak, jak np. *Znaczników*, zastępując w oznaczeniu literę M literą T.

4.7.1. Timer „Opóźnione załączenie” (ON-DELAYD)

Symbole *Timera* SD.

STL	LAD
L 10s SD T1	
R T1	

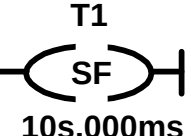
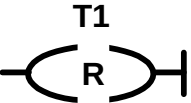
Realizowana funkcja czasowa:

Jeżeli na wejściu Trigger pojawi się zbocze narastające, a wejście Reset jest nieaktywne, to po czasie nastawionym w *Timerze*, następuje załączenie wyjścia *Timera* w stan '1' – wejście Trigger musi pozostać w stanie wysokim. Jeżeli wejście Trigger zmieni stan na '0', to nastąpi automatyczne resetowanie licznika czasu i kasowanie wyjścia.

Jeżeli wejście Reset zostanie ustawione na '1', to *Timer* zostaje skasowany w dowolnym momencie swojej pracy i przestanie odmierzać czas. Stan wyjścia powraca do stanu początkowego, czyli '0'. Ponowne rozpoczęcie liczenia czasu nastąpi dopiero po ustawieniu sygnału Reset w stan niski i pojawieniu się dodatniego zbocza na wejściu Trigger.

4.7.2. Timer „Opóźnione wyłączenie” (OFF-DELAYD)

Symbole *Timera* SF.

STL	LAD
L 10s SF T1	
R T1	

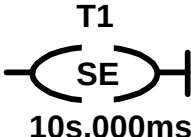
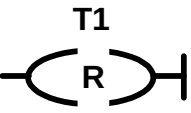
Realizowana funkcja czasowa:

Jeżeli wejście Trigger ma stan '1', przy stanie '0' na wejściu Reset, to wyjście jest załączone. Jeśli teraz wejście Trigger przejdzie w stan niski ('0' – zbocze opadające), to po czasie nastawionym w *Timerze*, następuje wyłączenie wyjścia *Timera* – ustawienie w stan '0'. Jeżeli wejście Trigger zmieni swój stan na '1' nastąpi automatyczne resetowanie licznika czasu i ustawienie wyjścia ponownie w stan wysoki ('1').

Jeżeli wejście Reset zostanie ustawione na '1', to *Timer* zostanie skasowany i przestanie odmierzać czas, a jego wyjście przyjmie stan '0'. Ponowne rozpoczęcie liczenia czasu nastąpi dopiero po ustawieniu sygnału Reset w stan niski ('0') i pojawieniu się ujemnego zbocza wyzwalającego na wejściu Trigger.

4.7.3. Timer “Pojedynczy impuls” (SINGLE PULSE)

Symbole *Timera* SE.

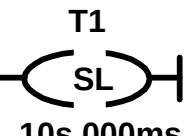
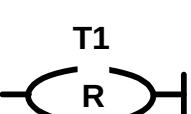
STL	LAD
L 10s SE T1	
R T1	

Realizowana funkcja czasowa:

Jeżeli na wejściu wyzwalającym Trigger pojawi się sygnał wyzwalający (zbrocze narastające), a wejście Reset jest nieaktywne, to układ załącza wyjście na nastawiony czas, a następnie powraca do stanu '0' – w tym czasie stan Trigger nie ma znaczenia (może przyjąć wartość '0'). Każde zbocze dodatnie na wejściu Trigger przedłuża impuls o kolejny nastawiony czas. Ustawienie wejścia Reset w dowolnym momencie kasuje wyjście *Timera* do stanu '0'. Ponowne wyzwolenie może nastąpić po ustawieniu wejścia Reset w stan '0' i po wystąpieniu kolejnego zbocza narastającego na wejściu Trigger.

4.7.4. Timer “Impulsy” (FLASHING)

Symbole *Timera* SL.

STL	LAD
L 10s SL T1	
R T1	

Realizowana funkcja czasowa:

Timer pełni funkcję generatora fali prostokątnej o wypełnieniu 50%. Układ zaczyna generować impulsy o nastawionym czasie trwania, gdy wejście wyzwalające Trigger przyjmuje stan '1'. Czasy trwania stanów '1' i '0' są równe, a częstotliwość pracy dla tego typu *Timera* wynosi: $f=1/(2T)$, gdzie T jest zadany czas do odmierzenia przez *Timer*. Ustawienie wejścia Reset w stan wysoki powoduje natychmiastowe ustawienie wyjścia w stan niski.

Gdy sygnał Trigger przyjmuje stan niski – wyjście *Timera* także przyjmuje stan '0'.

Zależności wejść Trigger, Reset i wyjścia *Timera* są następujące:

Po podaniu '1' na Trigger, gdy wejście Reset jest w stanie '0', wyjście najpierw pozostaje w stanie '0' przez ustawiony czas, a potem ustawia się w stan '1' i cykl się powtarza.

Jeżeli Trigger i Reset ustawione są równocześnie w stanie wysokim, to wyjście jest wyzerowane. W momencie, gdy Reset osiąga stan niski, a Trigger nadal pozostaje w stanie '1', to układ załącza wyjście na nastawiony czas, a potem je wyłącza, powtarzając cykl.

4.8. Liczniki

SYMBOL: **Cn**, gdzie n jest numerem *Licznika*, n=1..8

STANY LOGICZNE WEJŚĆ **CU, CD, RESET**:

'0' lub '1' w zależności od realizowanej funkcji w programie.

STANY LOGICZNE WYJŚCIA:

'0' lub '1' w zależności od realizowanej funkcji w programie.

ZAKRESY WARTOŚCI ZLICZANYCH:

0– 65535.

Licznik zlicza impulsy pojawiające się w momencie wystąpienia narastającego zbocza na wejściu wyzwalającym liczenie.

W strukturze logicznej *Licznika* możemy wyróżnić wejścia, wyjście oraz wartość liczbową impulsów do zliczenia.

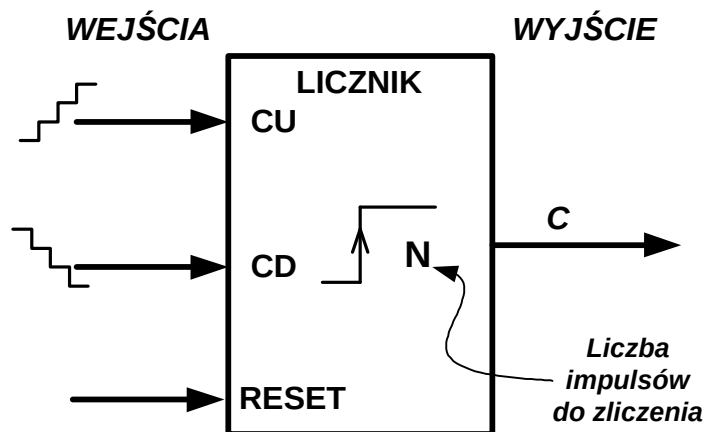
Wejścia i wyjście *Liczników* można logicznie wiązać także poprzez sygnały bitowe (I,Q,M).

Wejścia.

W skład wejść wchodzi:

– wejście resetujące – RESET – powoduje ustawienie wyjścia *Licznika* w stan niski ('0') oraz zatrzymuje zliczanie impulsów i zeruje *Licznik*.

– wejścia wyzwalające liczenie – CU,CD – sygnały podane na te wejścia, zbocza narastające, powodują liczenie odpowiednio w górę i w dół. Ten sam *Licznik* może więc zliczać jednocześnie i w górę, i w dół.



Rys. 4.8.1. Logiczna struktura Licznika.

Liczba impulsów do zliczenia.

Liczba impulsów do zliczenia od 0 do 65535 może być określona za pomocą instrukcji ładującej (STL) lub pliku ustawień („*.set”).

Wyjście.

Wyjście *Licznika* jest ustawiane w stan wysoki lub resetowane (ustawiane w stan niski) w zależności od stanu *Licznika* zliczającego impulsy.

Wyjście *Licznika* przyjmuje stan '1' jeśli zachodzi relacja:

Ilość impulsów aktualnie zliczonych jest większa lub równa od wartości zadanej do zliczenia.

Wyjście *Licznika* przyjmuje stan '0' jeśli zachodzi relacja:

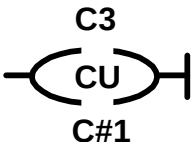
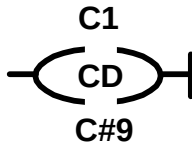
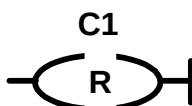
Ilość impulsów aktualnie zliczonych jest mniejsza od wartości zadanej do zliczenia lub wejście RESET jest aktywne.

Wystąpienie dodatniego zbocza na wejściu CU powoduje zwiększenie wartości zliczanej o 1.

Natomiast dodatnie zbocze na wejściu CD zmniejsza wartość o 1.

W programie wyjścia *Licznika* możemy używać tak, jak *Znaczników*, zastępując w oznaczeniu literę M literą C.

Symbole *Licznika*.

STL	LAD
L C#3 CU C1	
L C#9 CD C1	
R C1	

Zliczanie impulsów w górę:

Licznik zlicza impulsy pojawiające się na wejściu CU.

Zliczanie jest narastające. Jeśli wartość bieżąca *Licznika* jest większa lub równa zadanej liczbie impulsów, to wyjście *Licznika* przyjmuje stan '1'. Pozostaje ono tak długo w tym stanie, aż nie pojawi się na wejściu RESET stan wysoki, który kasuje wyjście i aktualny stan *Licznika* lub gdy *Licznik* zaczyna zliczać w dół i wartość bieżąca *Licznika* będzie mniejsza od wartości progowej.

Licznik nigdy się nie przepełnia, jeśli liczba zliczonych impulsów osiągnie wartość 65535, to *Licznik* przestaje zliczać w górę.

Zliczanie impulsów w dół:

Licznik zlicza impulsy pojawiające się na wejściu CD.

Zliczanie jest malejące. Jeśli wartość bieżąca *Licznika* jest większa lub równa zadanej liczbie impulsów, to wyjście *Licznika* przyjmuje stan '1'. Pozostaje ono tak długo w tym stanie, aż nie pojawi się na wejściu RESET stan wysoki, który kasuje wyjście i aktualny stan *Licznika*.

Licznik nigdy się nie przepełnia, jeśli liczba zliczonych impulsów osiągnie wartość 0, to *Licznik* przestaje zliczać w dół.



Dla wersji NEED-12DC-11-16-8, NEED-24DC-11-16-8, NEED-12DC-22-..., NEED-24DC-22-..., oprócz wspomnianych wyżej 8 *Liczników* istnieje jeszcze jeden, *Szybki Licznik* HC zliczający impulsy o maksymalnej częstotliwości 20kHz. HC jest *Licznikiem* sprzętowym, zlicza impulsy pojawiające się na wejściu **I11** dla NEED-12DC-11-16-8, NEED-24DC-11-16-8, NEED-12DC-22-16-8, NEED-24DC-22-16-8 oraz na wejściu **I4** dla NEED-12DC-22-08-4, NEED-24DC-22-08-4, . Wejścia CU, CD przejmują wtedy, oprócz funkcji kierunku zliczania, funkcję aktywacji *Szybkiego Licznika*.

Szybki Licznik może pracować w trybie częstotliwościowym – zlicza ilość impulsów pojawiających się na wejściu I11 lub I4 w czasie 1s.

Szybki Licznik, po osiągnięciu wartości maksymalnej - 65535, zaczyna liczenie od 0 po wykonaniu instrukcji Reset.



Dla wersji NEED-230VAC-11-16-8, NEED-230VAC-22-... *Szybki Licznik* HC mierzy częstotliwość sieci (50Hz lub 60Hz) – dla trybu częstotliwościowego. Natomiast w trybie licznika zlicza impulsy sieci zasilającej co 20ms (dla częstotliwości sieci zasilającej 50Hz) lub 16,6ms (dla częstotliwości sieci zasilającej 60Hz).

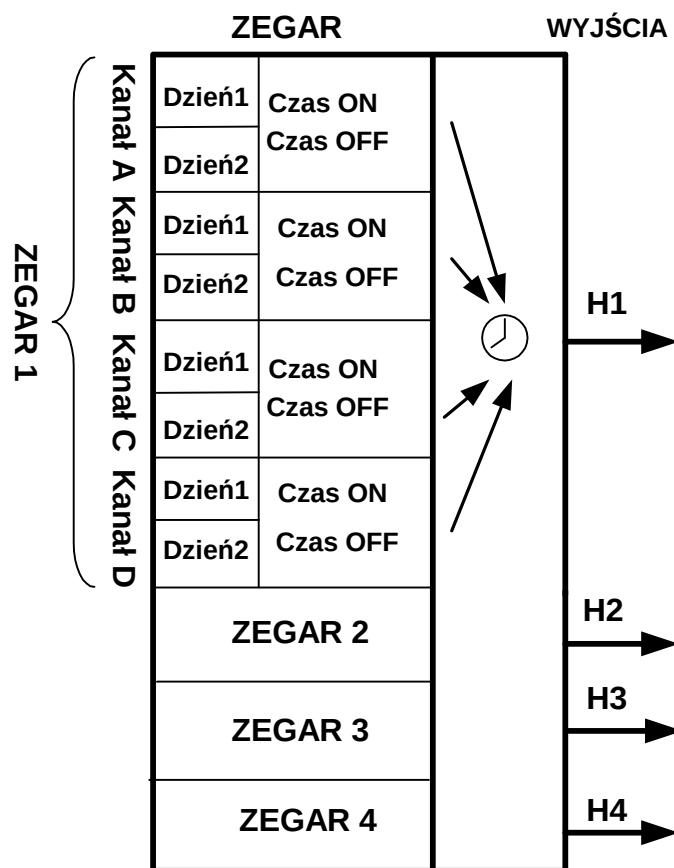
4.9. Zegary Tygodniowe

SYMBOL: **Hn**, gdzie n – jest numerem Zegara: n=1..4 dla NEED-...x1-..

n=1..8 dla NEED-...-22-..

STANY LOGICZNE WYJŚCIA:

'0' lub '1' w zależności od realizowanej funkcji w programie.

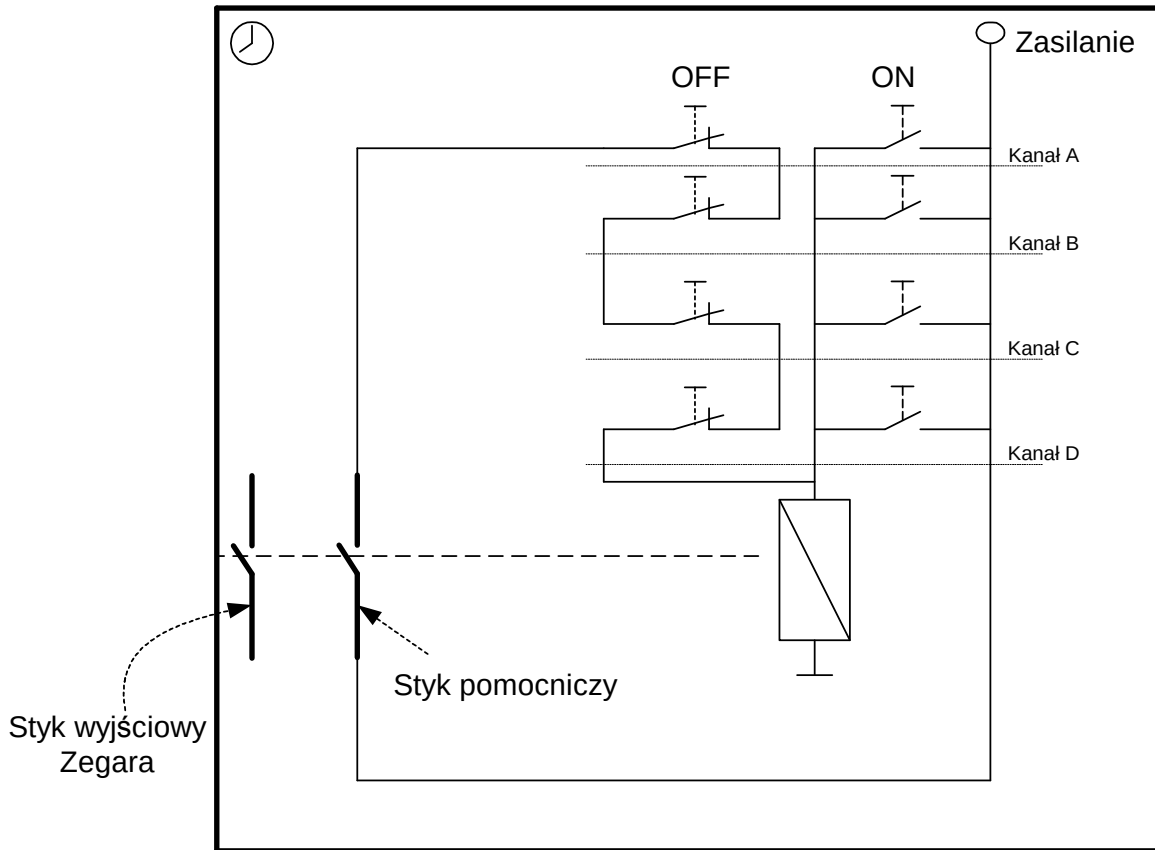


Rys. 4.9.1. Logiczna struktura Zegara dla NEED-...x1-...

Przy programowaniu przekaźnika NEED można użyć tygodniowe Zegary sterujące H1 – H4 dla NEED-...x1-.. lub H1 – H8 dla NEED-...-22-... Każdy Zegar posiada cztery kanały A, B, C, D. Wyjście Zegara jest wspólne dla czterech kanałów. Na rys. 4.9.1. przedstawiono logiczną strukturę Zegarów.

4.9.1. Działanie Zegara

Działanie Zegara w przekaźniku programowalnym możemy porównać do działania urządzenia, którego schemat ideowy przedstawia rys. 4.9.2. Załączenie Zegara odbywa się za pomocą przełączników „ON”, a wyłączenie za pomocą przełączników „OFF”. Ustawień Zegara dokonujemy używając programu PC Need.



Rys. 4.9.2. Schemat ideowy pojedynczego Zegara dla NEED-...x1-...

Przykład 1

Na rysunku 4.9.3. przedstawiono przykładowe okienko konfiguracyjne dla Zegara 1.



Rys. 4.9.3. Przykładowe okienko konfiguracyjne dla Zegara 1.

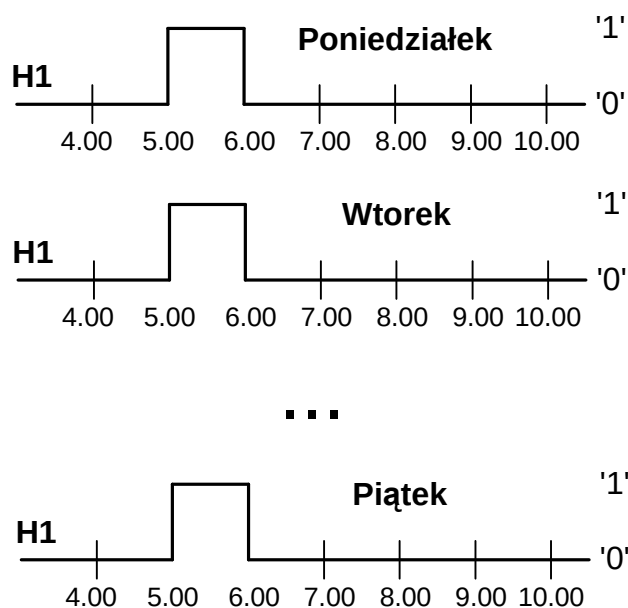
Dzień pierwszy – dzień pierwszy, w rozkładzie tygodniowym, w którym następuje załączenie/wyłączenie Zegara (w podanym przykładzie tym dniem jest poniedziałek).

Ostatni dzień – dzień ostatni, w rozkładzie tygodniowym, w którym następuje Załączenie/wyłączenie Zegara (w podanym przykładzie tym dniem jest piątek).

Godzina załączenia – godzina (dozwolone wartości od 0.00 do 23.59) załączenia wyjścia Zegara (w podanym przykładzie 5.00).

Godzina wyłączenia – godzina (dozwolone wartości od 0.00 do 23.59) wyłączenia wyjścia Zegara (w podanym przykładzie 6.00).

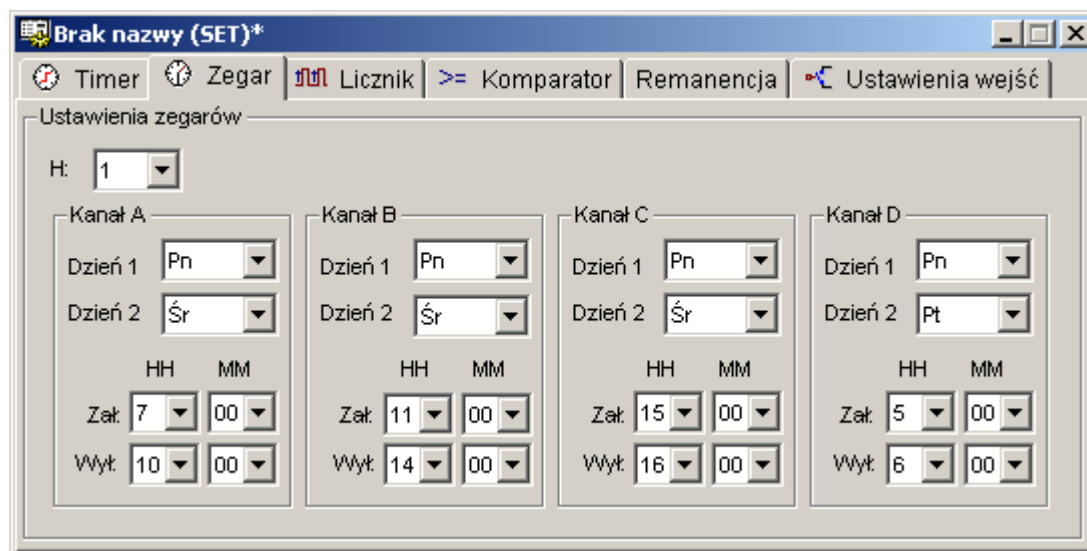
Dla przedstawionej konfiguracji Zegar 1 będzie ustawiał swoje wyjście w stan wysoki w każdy dzień od poniedziałku do piątku między 5.00 (godzina załączenia) a 6.00 (godzina wyłączenia). Przedstawia to rys. 4.9.4.



Rys. 4.9.4. Działanie Zegara 1 dla konfiguracji przedstawionej na rys. 4.9.4.

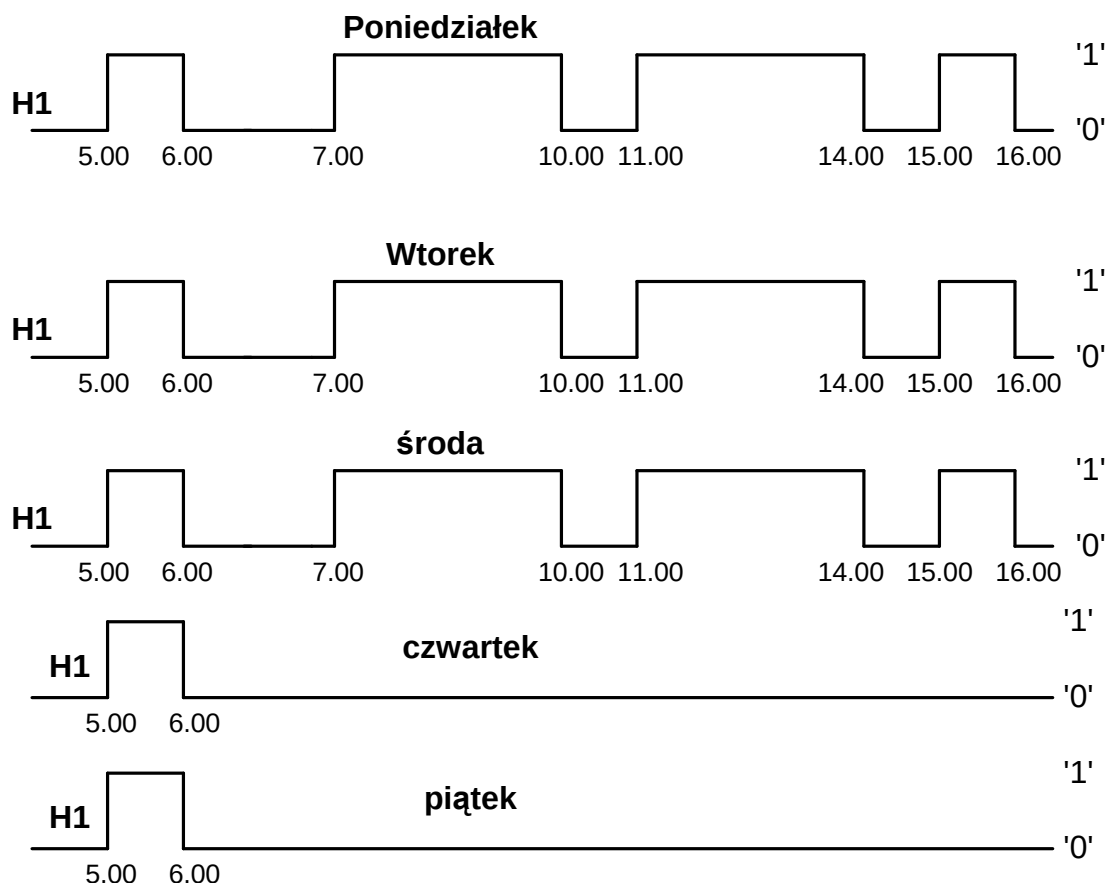
Przykład 2

Na rysunku 4.9.5. przedstawiono przykładowe okienko konfiguracyjne dla Zegara 1.



Rys. 4.9.5. Przykładowe okienko konfiguracyjne dla Zegara 1.

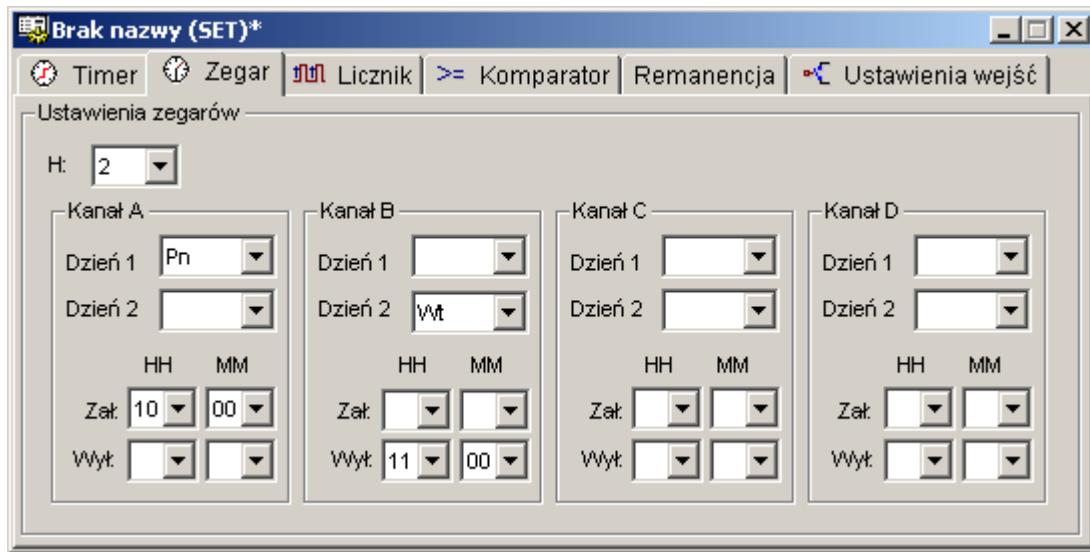
Dla przedstawionej konfiguracji Zegar 1 będzie ustawiał swoje wyjście w stan wysoki w każdy dzień od poniedziałku do środy włącznie, między 7.00 (godzina załączenia) a 10.00 (godzina wyłączenia), 11.00 (godzina załączenia) a 14.00 (godzina wyłączenia), 15.00 (godzina załączenia) a 16.00 (godzina wyłączenia). Dodatkowo od poniedziałku do piątku wyjście Zegara 1 będzie ustawiane między 5.00 a 6.00. Przedstawia to rys. 4.9.6.



Rys. 4.9.6. Działanie Zegara 1 dla konfiguracji przedstawionej na rys.4.9.6.

Przykład 3

Na rysunku 4.9.7. przedstawiono przykładowe okienko konfiguracyjne dla Zegara 2.



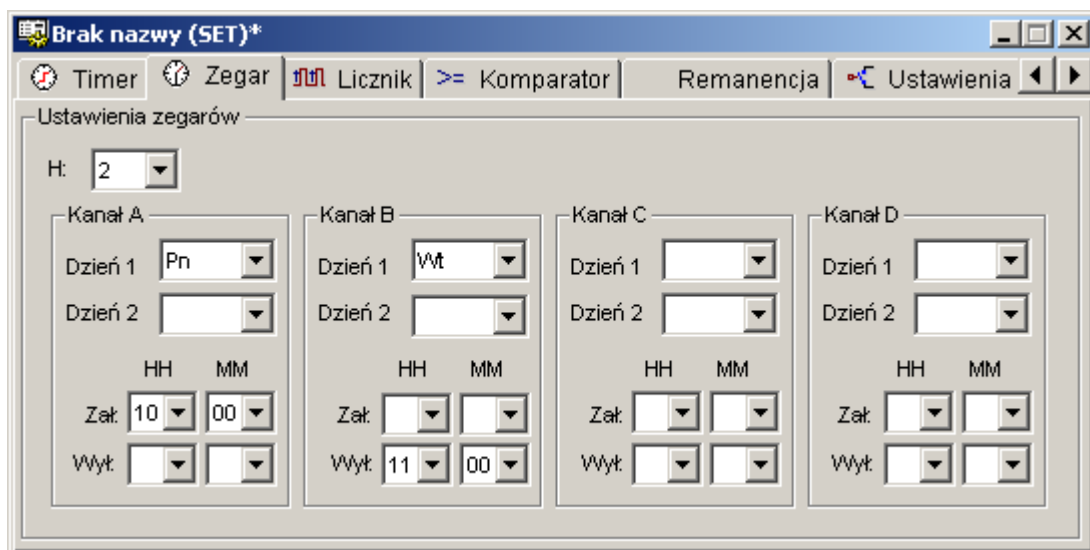
Rys.4.9.7. Przykładowe okienko konfiguracyjne dla Zegara 2.

Dla przedstawionej konfiguracji Zegar 2 będzie ustawiał swoje wyjście w stan wysoki w każdy poniedziałek o godz.10.00 (godzina załączenia) a we wtorek o godzinie 11.00 wyłączał się. Jeśli chcemy rozciągnąć sterowanie na kilka dni, to wtedy nie wypełniamy pola „Godzina załączenia” lub „Godzina wyłączenia” w odpowiednich kanałach. Rysunek 4.9.8. przedstawia diagram pracy Zegara 2.



Rys. 4.9.8. Działanie Zegara 2 dla konfiguracji przedstawionej na rys. 4.9.8.

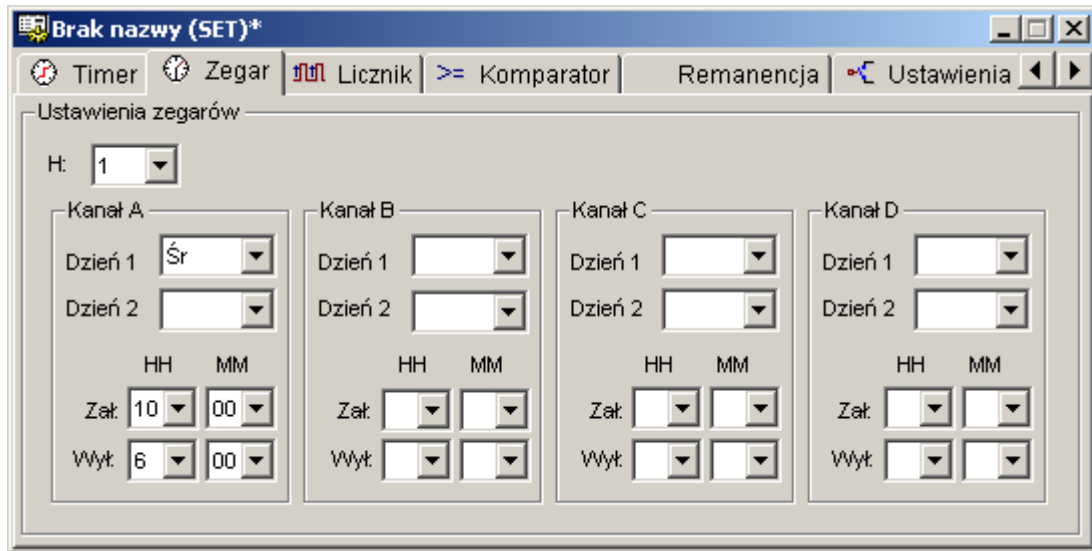
Identyczne działanie Zegara 2 uzyskamy, jeśli dokonamy konfiguracji tak, jak na rys. 4.9.9.



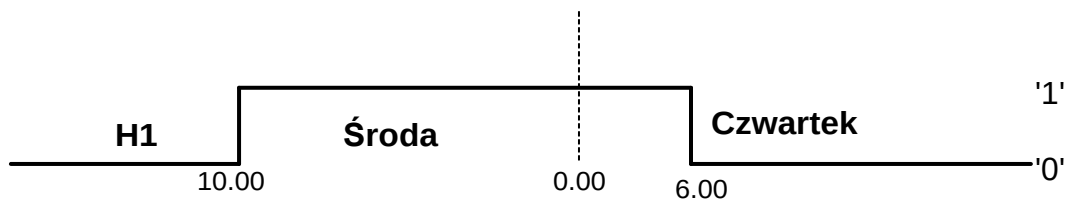
Rys. 4.9.9. Przykładowe okienko konfiguracyjne dla Zegara 2.

Przykład 4

Jeśli godzina załączenia jest późniejsza od godziny wyłączenia, to Zegar wyłącza swoje wyjście następnego dnia – konfiguracja tak, jak na rys. 4.9.10.



Rys. 4.9.10. Przykładowe okienko konfiguracyjne dla Zegara 1.



Rys. 4.9.11. Działanie Zegara 1 dla konfiguracji przedstawionej na rys. 4.9.10.



Jeśli godzina wyłączenia jest mniejsza od godziny załączenia, to przekaźnik programowalny NEED wyłącza wyjście używanego Zegara w następnym dniu.

Przykład 5

Załączenie wyjścia Zegara na 24 godziny.

Dla pracy 24-godzinnej należy dokonać konfiguracji Zegara 3 tak, jak na rys. 4.9.12.

The screenshot shows the 'Ustawienia zegarów' (Timer Settings) window. The timer is set to 3 hours. The configuration for the four channels is as follows:

Kanał	Dzień 1	Dzień 2	Zak. (HH:MM)	Wyk. (HH:MM)
Kanał A	Pn		10:00	
Kanał B		Wt		10:00
Kanał C				
Kanał D				

Rys. 4.9.12. Przykładowe okienko konfiguracyjne dla 24-godzinnego

Przykład 6

Należy pamiętać, iż stan wyjścia Zegara zależy od stanu wszystkich czterech kanałów.

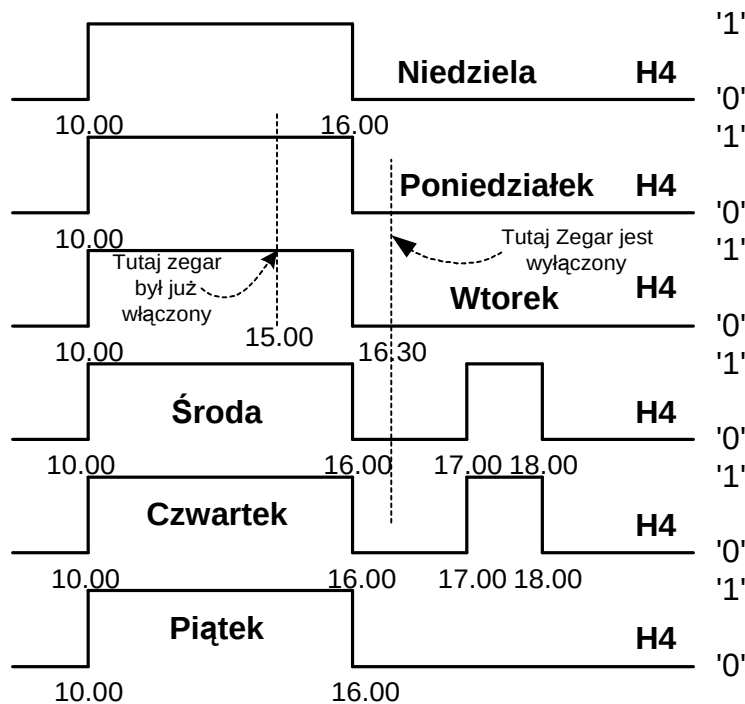
Rozpatrzmy konfigurację Zegara 4 przedstawioną na rys. 4.9.13.

The screenshot shows the 'Ustawienia zegarów' (Timer Settings) window. The timer is set to 4 hours. The configuration for the four channels is as follows:

Kanał	Dzień 1	Dzień 2	Zak. (HH:MM)	Wyk. (HH:MM)
Kanał A	Nd	Pt	10:00	16:00
Kanał B	Śr	Czw	17:00	18:00
Kanał C	Pn	Wt	15:00	16:30
Kanał D				

Rys. 4.9.13. Przykładowe okienko konfiguracyjne Zegara 4.

Zauważmy, iż czasy ustawione w kanale A i C pokrywają się – rys. 4.9.14.

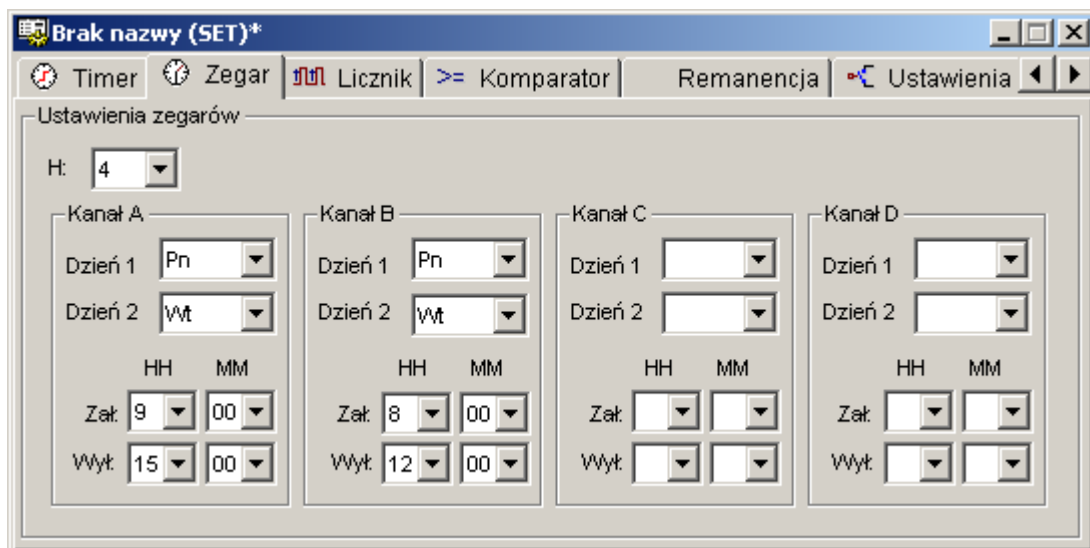


Rys. 4.9.14. Działanie Zegara 4 dla konfiguracji przedstawionej na rys. 4.9.13.

Czasy ustawień Zegara mogą nachodzić na siebie, należy zawsze pamiętać, iż wyjście Zegara załącza kanał z ustawioną wcześniejszą godziną załączenia, a wyłącza kanał z ustawioną wcześniejszą godziną wyłączenia.

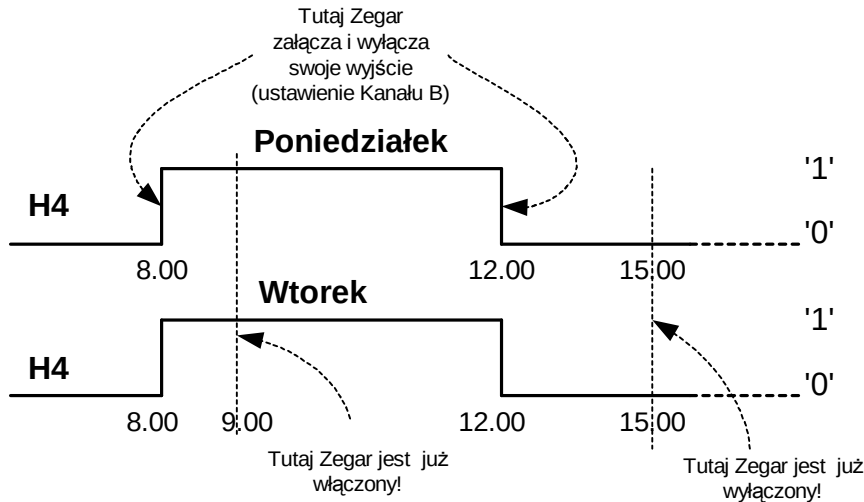
Przykład 7

Rozpatrzmy konfigurację Zegara 4 przedstawioną na rys. 4.9.15.



Rys. 4.9.15. Przykładowe okienko konfiguracyjne Zegara 4.

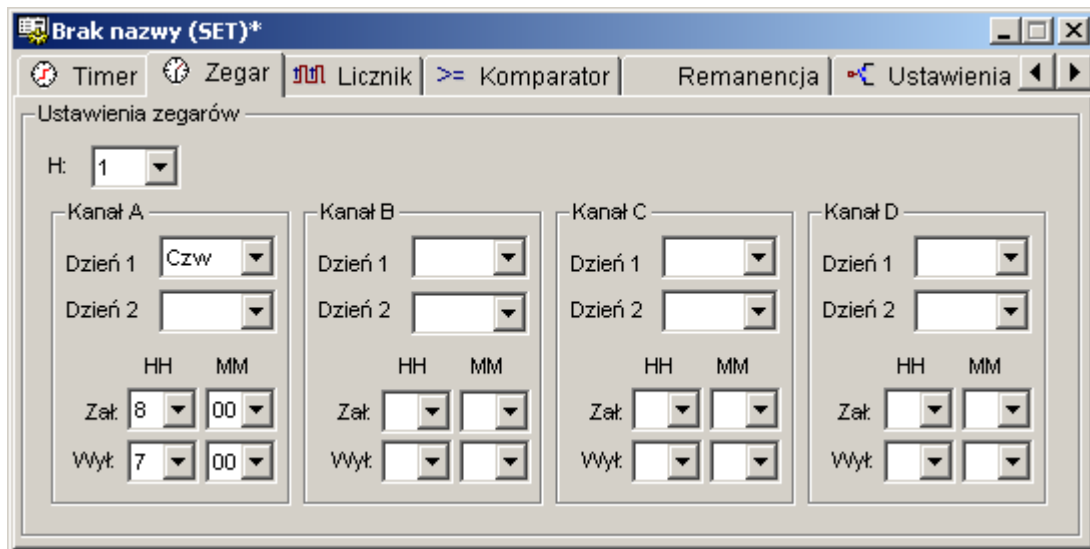
Zegar załączy swoje wyjście w poniedziałek i wtorek o godzinie 8.00, a wyłączy o godzinie 12.00 (a nie o 15.00! – stosujemy zasadę: pierwszy załącza i pierwszy wyłącza). Na rys. 4.9.16. przedstawiono diagram pracy dla Zegara 4.



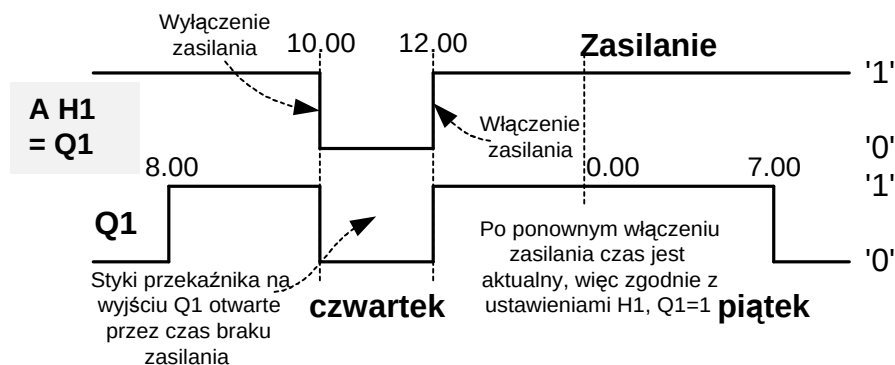
Rys. 4.9.16. Działanie Zegara 4 dla konfiguracji z rys. 4.9.16.

Przykład 8

Na rys. 4.9.17. przedstawiona jest konfiguracja Zegara 1. Jeśli między 10.00 a 12.00 zostanie wyłączone zasilanie, wówczas styki wyjściowe przekaźnika zostaną rozwarne, ale czas nadal będzie kontrolowany. Przy powrocie napięcia zasilającego o godz.12.00, wyjście Zegara H1 będzie w stanie wysokim – zgodnie z konfiguracją H1. Powyższą sytuację przedstawia rys. 4.9.18.



Rys. 4.9.17. Przykładowe okienko konfiguracyjne Zegara 1.



Rys. 4.9.18. Działanie Zegara 1 dla konfiguracji z rys. 4.9.17.



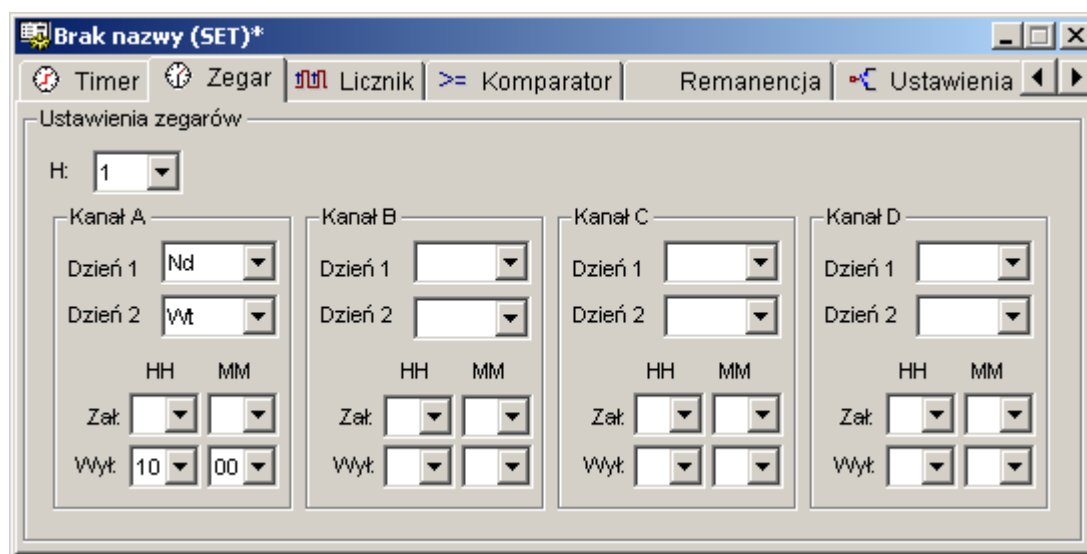
Przy zaniku zasilania czas w przekaźniku jest odmierzany dalej, jednak styki przekaźników wyjściowych nie zamykają się. Czas podtrzymywania Zegara w trakcie zaniku zasilania wynosi 64 godziny (dla 25°C).

4.9.2. Uwagi dotyczące konfiguracji Zegara

4.9.2.1. Jedno pole puste – 3 pola wypełnione (dla jednego kanału)

1. Wypełnione wszystkie pola, niewypełnione pole „Za”.

Przykład:



Rys. 4.9.2.1.1. Przykładowa konfiguracja Zegara – niewypełnione pole „Za”.

Od niedzieli do wtorku Zegar będzie wyłączał swoje wyjście o godz. 10.00.

2. Wypełnione wszystkie pola, niewypełnione pole „Wył”

Przykład:

Rys. 4.9.2.1.2. Przykładowa konfiguracja Zegara – wypełnione 3 pola.



Od niedzieli do wtorku Zegar będzie załączał swoje wyjście o godz. 5.00. W przypadku, gdy ustawiona jest tylko godzina załączenia, Zegar będzie zawsze załączony!

3. Wypełnione wszystkie pola, niewypełnione pole „Dzień 1”.

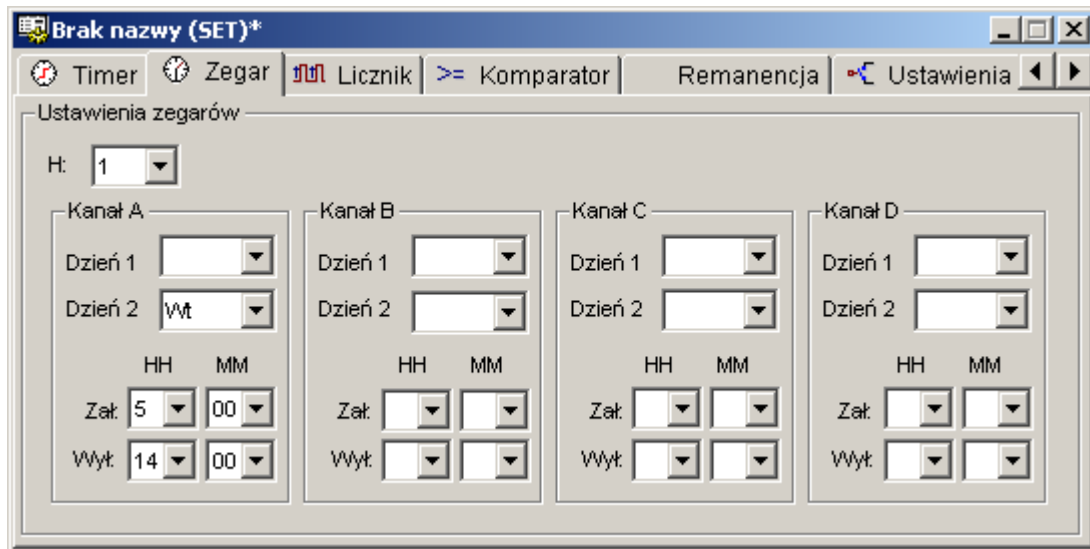
Przykład:

Rys. 4.9.2.1.3. Przykładowa konfiguracja Zegara – niewypełnione pole „Dzień 1”.

Zegar będzie załączał swoje wyjście tylko we wtorki o godz. 5.00, a wyłączał je tylko we wtorki o godz. 14.00.

4. Wypełnione wszystkie pola, niewypełnione pole „Dzień 2”.

Przykład:



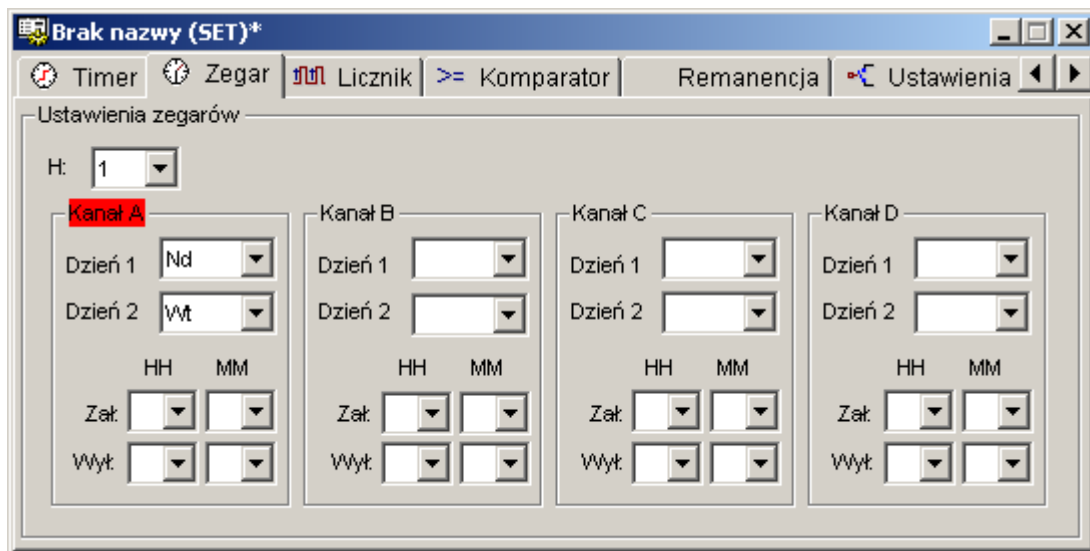
Rys. 4.9.2.1.4. Przykładowa konfiguracja Zegara – niewypełnione pole „Dzień 2”.

Zegar załączy swoje wyjście tylko w niedzielę o godz. 5.00, a wyłączy je tylko w niedzielę o godz. 14.00.

4.9.2.2. Dwa pola puste – 2 pola wypełnione (dla jednego kanału)

1. Niewypełnione pola „Zał”, „Wył”.

Przykład:

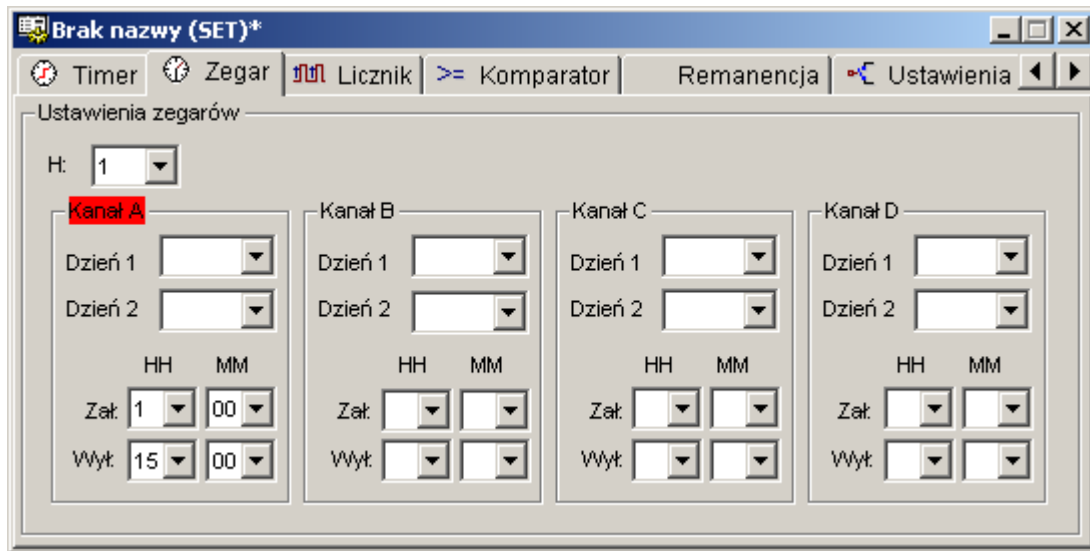


Rys. 4.9.2.2.1. Przykładowa konfiguracja Zegara – niewypełnione pola „Zał” i „Wy”.

Zegar nie działa – ustawienie niedozwolone, którego nie można przesłać do przekaźnika.

2. Niewypełnione pola „Dzień 1”, „Dzień 2”.

Przykład:



Rys. 4.9.2.2.2. Przykładowa konfiguracja Zegara – niewypełnione pola „Dzień 1” i „Dzień2”.

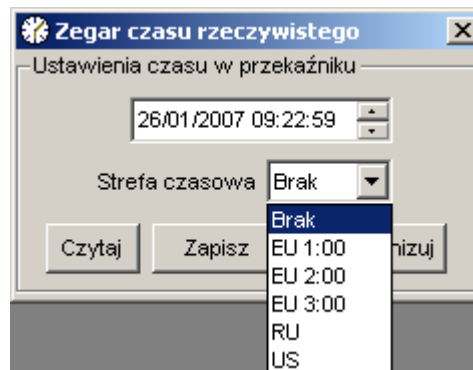
Zegar nie działa – ustawienie niedozwolone, którego nie można przesłać do przekaźnika.

4.9.2.3. Trzy pola puste (dla jednego kanału)

Zegar nie działa – ustawienie niedozwolone, którego nie można przesłać do przekaźnika.

4.10. Zegar Czasu Rzeczywistego

Zegar Czasu Rzeczywistego pozwala na ustawienie w przekaźniku NEED godziny, daty oraz dla NEED-...-11-16..., NEED-...-22-... strefy czasowej, dla której występuje odpowiednie przesunięcie czasu z letniego na zimowy i na odwrót. Ustawień dla Zegara Czasu Rzeczywistego dokonuje się za pomocą programu PCNeed. Na rys. 4.10.1. przedstawione jest okno, w którym dokonuje się odpowiednich ustawień.



Rys. 4.10.1. Okno ustawień dla Zegara Czasu Rzeczywistego.

Zegar Czasu Rzeczywistego pozwala na ustawienie następujących stref czasowych:

EU 1:00 – strefa, w której zmiana czasu na czas letni odbywa się w ostatnią niedzielę marca z godziny 1:00 na 2:00, a zmiana na czas zimowy w ostatnią niedzielę października z 2:00 na 1:00.

EU 2:00 – strefa, w której zmiana czasu na czas letni odbywa się w ostatnią niedzielę marca z godziny 2:00 na 3:00, a zmiana na czas zimowy w ostatnią niedzielę października z 3:00 na 2:00.

EU 3:00 – strefa, w której zmiana czasu na czas letni odbywa się w ostatnią niedzielę marca z godziny 3:00 na 4:00, a zmiana na czas zimowy w ostatnią niedzielę października z 4:00 na 3:00.

RU – strefa, w której zmiana czasu na czas letni odbywa się w ostatnią niedzielę marca z godziny 2:00 na 3:00, a zmiana na czas zimowy w ostatnią niedzielę października z 3:00 na 2:00.

US – strefa, w której zmiana czasu na czas letni odbywa się w drugą niedzielę marca z godziny 2:00 na 3:00, a zmiana na czas zimowy w pierwszą niedzielę listopada z 3:00 na 2:00.

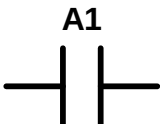
4.11. Komparator – wejścia analogowe

SYMBOL: **An**, gdzie n – jest numerem komparatora: n=1..8 dla NEED-...x1-08-..
n=1..12 dla NEED-...11-16-..
n=1..16 dla NEED-...22-..

STANY LOGICZNE WEJŚCIA:

'0' lub '1' w zależności od wielkości napięć analogowych i ustawień konfiguracyjnych przekaźnika programowalnego.

Symbole Komparatora

STL	LAD
A A1 lub O A1 lub X A1	
AN A1 lub ON A1 lub XN A1	

Układ przekaźnika programowalnego posiada w swojej strukturze dwa (NEED...-08-4) lub trzy (NEED...-16-8) wejścia analogowe. Na rys. 4.11.1. przedstawiono logiczną strukturę Komparatora dla przekaźnika NEED...-x1-08-4.

Sygnały analogowe w Komparatorze mogą być porównywane ze sobą, z ustaloną Wartością Wzorcową oraz z nastawą zewnętrznego potencjometru. Od wyniku tego porównania zależy stan wyjść Komparatora. Wyjścia ustawiane zawsze w stan wysoki ('1'), jeśli warunek porównania jest spełniony. Tabele 4.11.1. oraz 4.11.2. przedstawiają dostępne porównania.

Tab. 4.11.1. Możliwe konfiguracje porównań Komparatora dla NEED-...x1-08-4x.

Lp.	Typ porównania
1.	$I7 \geq$ Wartość Wzorcowa
2.	$I7 \leq$ Wartość Wzorcowa
3.	$I8 \geq$ Wartość Wzorcowa
4.	$I8 \leq$ Wartość Wzorcowa
5.	$I7 \geq$ Potencjometr
6.	$I7 \leq$ Potencjometr
7.	$I8 \geq$ Potencjometr
8.	$I8 \leq$ Potencjometr
9.	$I7 \geq I8$
10.	$I7 \leq I8$

Tab. 4.11.2. Możliwe konfiguracje porównań Komparatora dla NEED-...11- 16-8x.

Lp.	Typ porównania
1.	$I14 \geq$ Wartość Wzorcowa
2.	$I4 \leq$ Wartość Wzorcowa
3.	$I15 \geq$ Wartość Wzorcowa
4.	$I15 \leq$ Wartość Wzorcowa
5.	$I16 \geq$ Wartość Wzorcowa
6.	$I16 \leq$ Wartość Wzorcowa
7.	$I14 \geq$ Potencjometr
8.	$I14 \leq$ Potencjometr
9.	$I15 \geq$ Potencjometr
10.	$I15 \leq$ Potencjometr
11.	$I16 \geq$ Potencjometr
12.	$I16 \leq$ Potencjometr
13.	$I14 \geq I15$
14.	$I14 \leq I15$
15.	$I14 \geq I16$
16.	$I14 \leq I16$
17.	$I15 \geq I16$
18.	$I15 \leq I16$
19.*	$ASYM \geq$ Wartość Wzorcowa
20.*	$ASYM \leq$ Wartość Wzorcowa
21.*	$ASYM \geq$ Potencjometr
22.*	$ASYM \leq$ Potencjometr

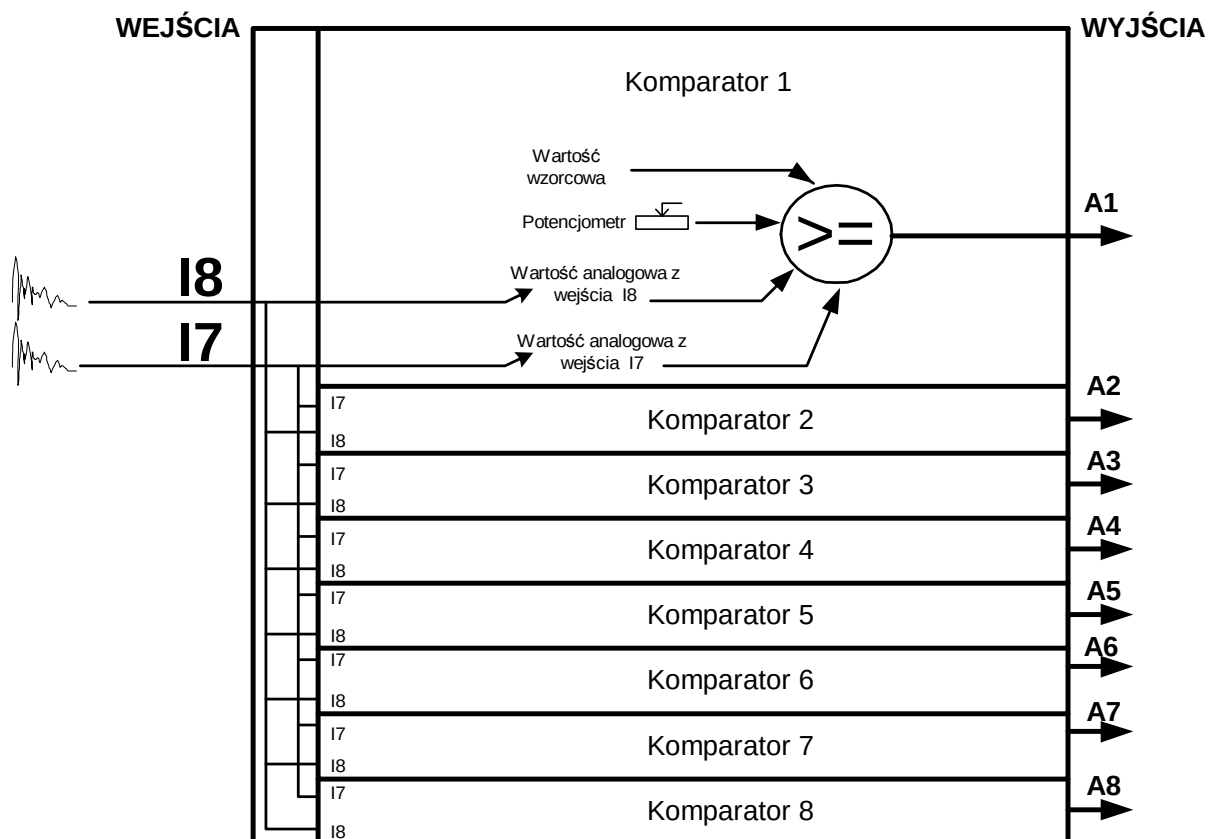
* - możliwe tylko dla NEED-230AC-01-16-8



ASYM jest wskaźnikiem asymetrii faz (tylko w wersji NEED-230AC-...-16-8). Pokazuje wartości skuteczne sumy faz L1, L2, L3. ASYM przyjmuje wartość 0V dla prawidłowych poziomów faz L1, L2, L3. Gdy występuje asymetria (poziom napięcia na którejś z faz jest różny od nominalnego) ASYM przyjmuje wartości większe od 0V.



ASYM wraz z Znacznikiem MDIR oraz Komparatorami sprawiają, iż NEED-230AC-..-16-8 może pełnić funkcję przekaźnika nadzorczego monitorując asymetrię, kolejność oraz poziom napięć faz L1, L2, L3.



Rys. 4.11.1. Logiczna struktura Komparatora dla przekaźnika NEED..-08-4.

Do porównań używamy

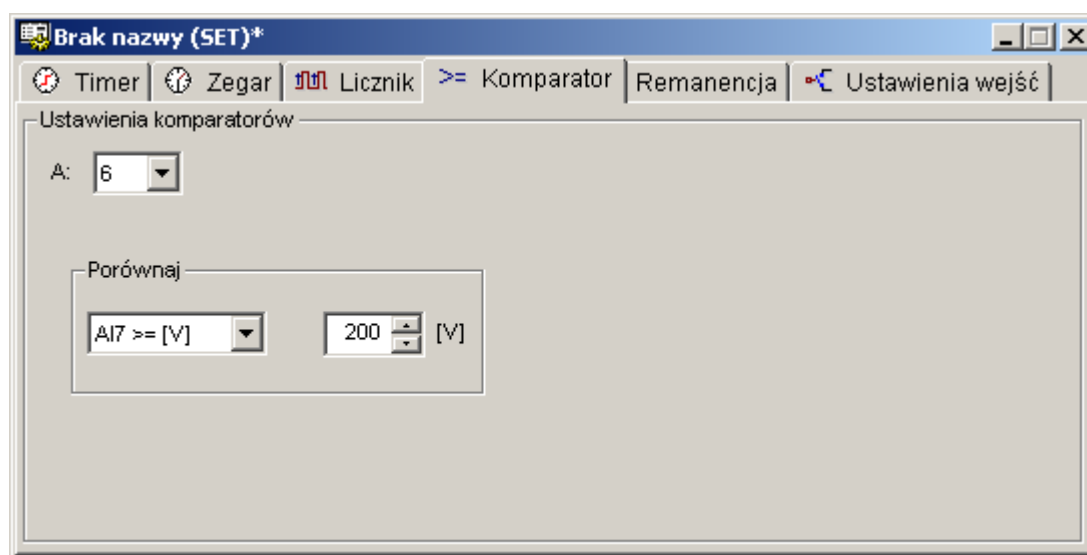
1. Wartości Wzorcowej (ustalanej na etapie konfiguracji w programie PC Need w zakresie: 0 – 255V dla NEED-230AC-.., NEED-220DC-.., 0 – 25,5V i 0 – 12,75V dla NEED-24DC-..-16-8., NEED-12DC-..-16-8).
2. Potencjometru (zakres regulacji 1 – 255 dla NEED-230AC-.., NEED-220DC-.. oraz 0,1 – 25,5 dla NEED-24DC-..-16-8., NEED-12DC-..-16-8) – dostępnego na płycie czołowej przekaźnika.
3. Wartości napięcia z wejść analogowych.

W tabeli 4.11.3 przedstawiono zakresy przyjmowane przez Wartość Wzorcową.

Tab. 4.11.3. Możliwe zakresy Wartości Wzorcowej do porównań komparatorów.

Typ	Zakres Wartości Wzorcowej
NEED-230AC-..	0 – 255V
NEED-220DC-..	0 – 255V
NEED-12DC-..	0 – 25,5V
NEED-24DC-..	0 – 25,5V
NEED-12DC-..-16-8	0 – 12,75V
NEED-24DC-..-16-8	0 – 12,75V

Przykład konfiguracji komparatora A6 do porównania z Wartością Wzorcową przedstawiono na rys. 4.11.3.



Rys. 4.11.3. Przykład konfiguracji komparatora A6 do porównania z Wartością Wzorcową.

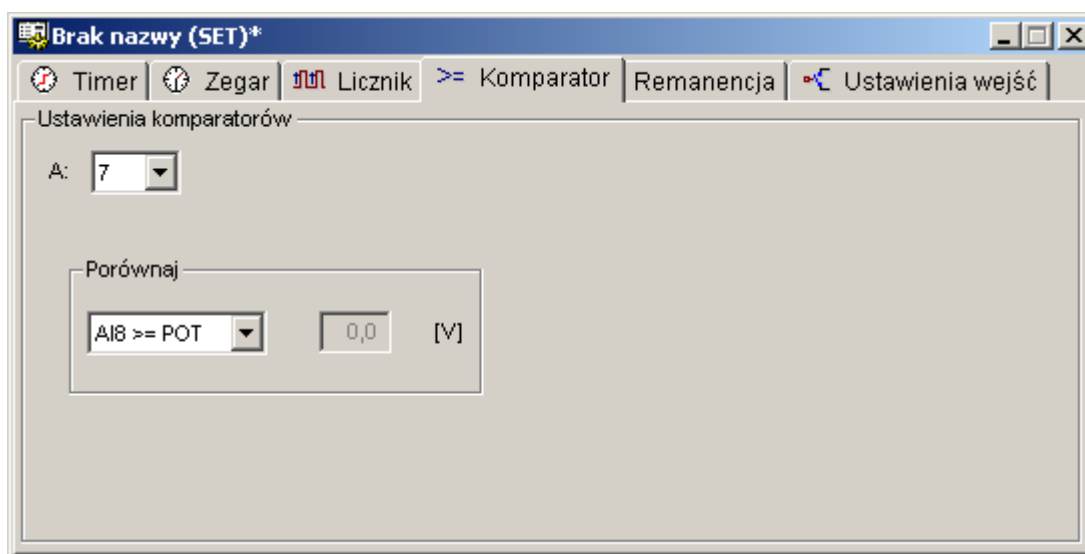
Wyjście Komparatora A6 zostanie ustawione w stan '1', gdy wartość napięcia na wejściu I7 będzie większa lub równa 200V.

W tabeli 4.11.4 przedstawiono zakres Potencjometru do porównań komparatorów.

Tab. 4.11.4. Możliwe wartości Potencjometru do porównań komparatorów.

Typ	Zakres Potencjometru
NEED-230AC-..	1 - 255
NEED-220DC-..	
NEED-12DC-..	
NEED-24DC-..	

Przykład konfiguracji komparatora A7 do porównania z Potencjometrem przedstawiono na rys. 4.11.4.



Rys. 4.11.4. Przykład konfiguracji komparatora A7 do porównania z Potencjometrem.

Wyjście Komparatora A7 zostanie ustawione w stan '1', gdy wartość napięcia na wejściu I8 będzie większa lub równa wartości ustalonej za pomocą Potencjometru.



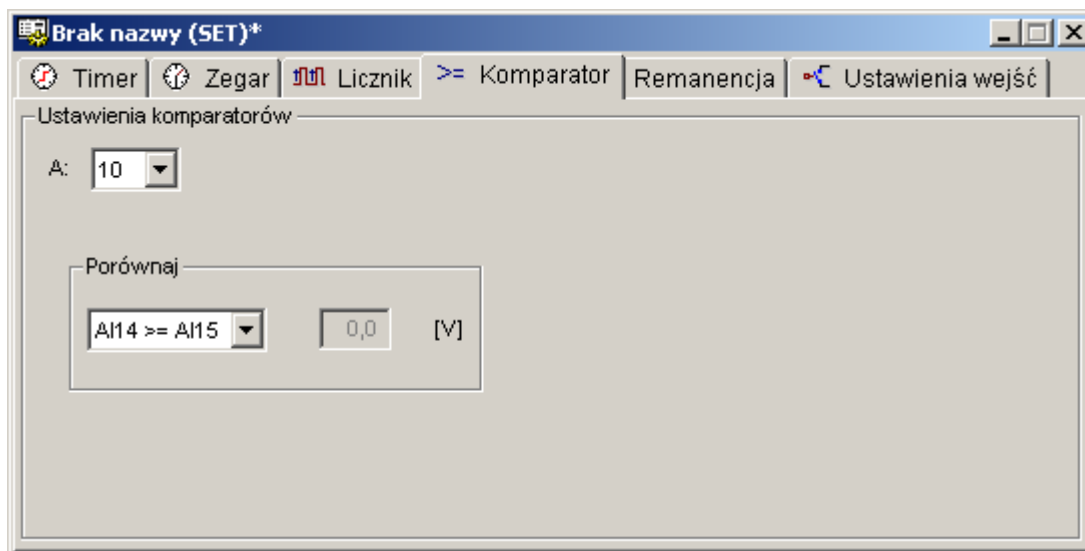
Dla NEED-24DC-..., NEED-12DC-.. Potencjometr przyjmuje wartości od 0,1 do 25,5. Należy zwrócić uwagę na to, iż w programie PC Need, w okienku „Podglądu zmiennych” POT przyjmuje wartości 1 – 255, ale do porównań komparatora używane są wartości od 1/10 (czyli 0,1) do 255/10 (czyli 25,5)!
Na przykład dla relacji „AI7<=POT” przy napięciu na AI7=5V, wyjście komparatora będzie ustawiane w stan wysoki np. dla ustawień zmiennej POT=50 (czyli 50/10) w „Podglądzie zmiennych”.
Aby więc dobrze ustawić Potencjometr dla przekaźników NEED-24DC-..., NEED-12DC-.. należy wskazania zmiennej POT, w „Podglądzie zmiennej”, zawsze dzielić przez 10.

W tabeli 4.11.5. przedstawiono zakresy wejść analogowych w przekaźniku NEED.

Tab. 4.11.5. Zakresy wejść analogowych w przekaźniku NEED.

Typ	Rodzaj trybu pracy wejścia analogowego	Zakres	Rozdzielczość
NEED-230AC-..	Napięciowe	0 – 255V	1V
NEED-220DC-..	Napięciowe	0 – 255V	1V
NEED-12DC-x1-08-4	Napięciowe	0 – 25,5V	0,1V
NEED-24DC-x1-08-4	Napięciowe	0 – 25,5V	0,1V
NEED-12DC-22-08-4	Napięciowe	0 – 25,5V/12,75V	0,1V/0,05V
NEED-24DC-22-08-4	Napięciowe	0 – 25,5V/12,75V	0,1V/0,05V
NEED-12DC-..-16-8	Napięciowe	0 – 25,5V/12,75V	0,1V/0,05V
NEED-24DC-..-16-8	Napięciowe	0 – 25,5V/12,75V	0,1V/0,05V
NEED-12DC-..-16-8	Prądowe	0 – 51mA/25,5mA	0,2mA/0,1mA
NEED-24DC-..-16-8	Prądowe	0 – 51mA/25,5mA	0,2mA/0,1mA

Przykład konfiguracji komparatora A10 porównującego wartości sygnałów na wejściach analogowych I14, I15 przedstawiono na rys. 4.11.5.



Rys. 4.11.5. Przykład konfiguracji komparatora A10 do porównania dwóch wartości analogowych

Wyjście Komparatora A10 zostanie ustawione w stan '1', gdy wartość napięcia na wejściu analogowym I14 będzie większa lub równa wartości napięcia na wejściu analogowym I15.



Jeśli jakieś wejście analogowe przekaźnika NEED skonfigurowane jest jako prądowe, to dla porównań komparatora brane są tylko wartości **napięcia** według następującego wzoru:

*Wartość napięcia dla komparatora[V] = 0,5*wartość prądu zmierzonego na wejściu[mA]*

Jest to liniowe skalowanie, gdzie 20mA odpowiada 10V.

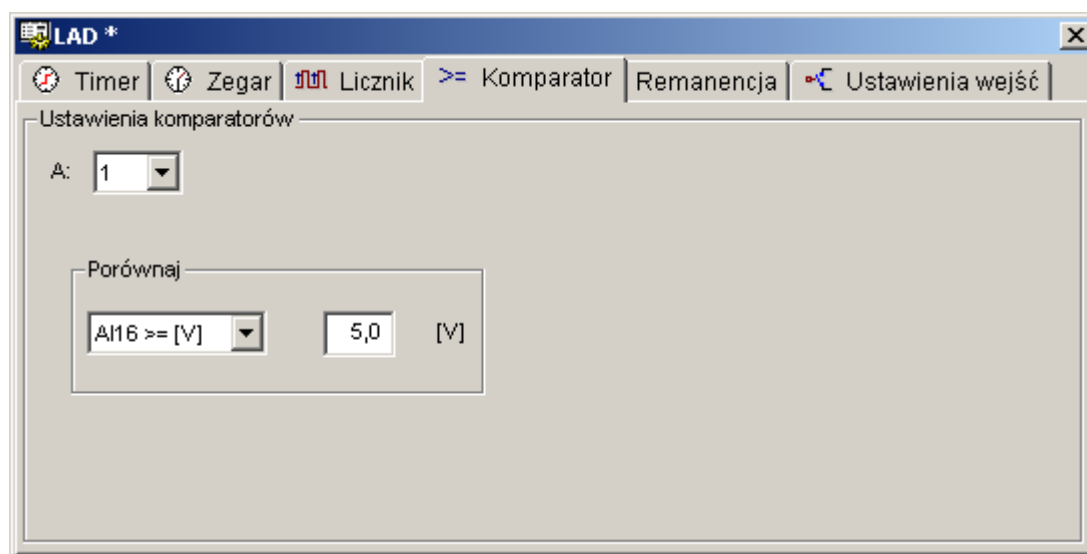
Przykład:

Do wejścia analogowego I16 podłączamy czujnik o wyjściu prądowym. Chcemy, aby mierzona wartość analogowa (np. ciśnienie), w „przeliczeniu” na prąd, nie przekroczyła 10mA.

Wtedy komparator A1 należy skonfigurować tak, jak na rys. 4.11.6. – zgodnie z przytoczoną powyżej formułą:

$$0,5 * 10\text{mA} = 5\text{V}$$

Do pola z lewej strony należy wpisać wartość 5V.



Rys. 4.11.6. Przykład konfiguracji komparatora A1 w programie PC Need, dla wejścia analogowego AI16 skonfigurowanego jako wejście prądowe.



Dla przekaźnika NEED-230AC-x1-.. wejścia analogowe są czytane co 4ms. Opóźnienie to nie zależy od ustawienia opóźnień dla wejść I7, I8 lub I14, I15, I16 przy konfiguracji w programie PC Need – patrz rozdział „8.4. Opóźnienie wejść”. Dla przekaźników: NEED-12DC-x1-.. oraz NEED-24DC-x1-.. ustawienie opóźnienia wejść analogowych I7, I8 lub I14, I15, I16 spowoduje uśrednienie odczytywanych wartości mierzonych według poniższej reguły:

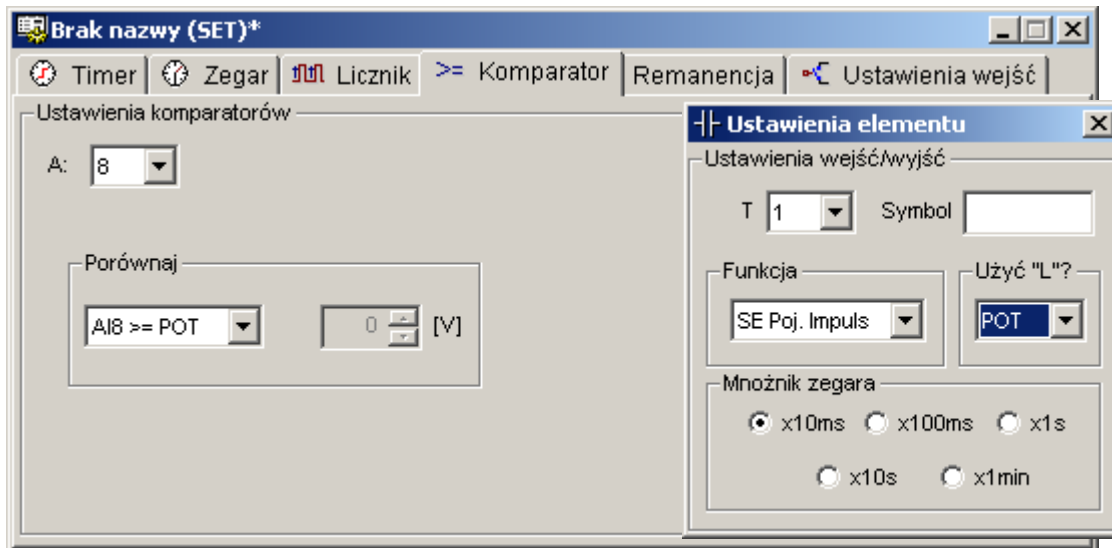
Wartość bieżąca = (wartość poprzednia + wartość odczytana z wejścia analogowego) / 2

Wejścia analogowe, dla wykonan przekaźnika NEED-12DC-x1-.. oraz NEED-24DC-x1-.. czytane są co 4ms.

4.12. Potencjometr

Potencjometr należy do zasobów typowo sprzętowych i można go używać do:
ustawiania czasów dla *Timerów*,
ustawiania wartości do zliczania dla *Liczników*,
ustawiania progu przełączania *Komparatora*.

Pełny obrót Potencjometru odpowiada wartościom od 1 do 255. Do ustawiania odpowiednich wartości zadawanych z Potencjometru służy instrukcja „L” (STL – patrz rozdział 5.1.2.21., LAD – patrz rozdział 5.2.), w której można programowo modyfikować zakresy Potencjometru dla lepszego dopasowania się do oczekiwanej wartości mierzonej. Przykład wykorzystania Potencjometru dla NEED-230AC-x1-.. przedstawiono na rys. 4.12.1.



W powyższym przykładzie *Timer* T1 będzie odmierzał czas o długości równej: *ustawienie Potencjometru x 10ms* (np. $12 \cdot 10 = 120ms$). Natomiast wyjście *Komparatora* A8 będzie w stanie wysokim, gdy wartość napięcia na wejściu analogowym I8 będzie większa od wartości ustawionej za pomocą Potencjometru (1 – 255).

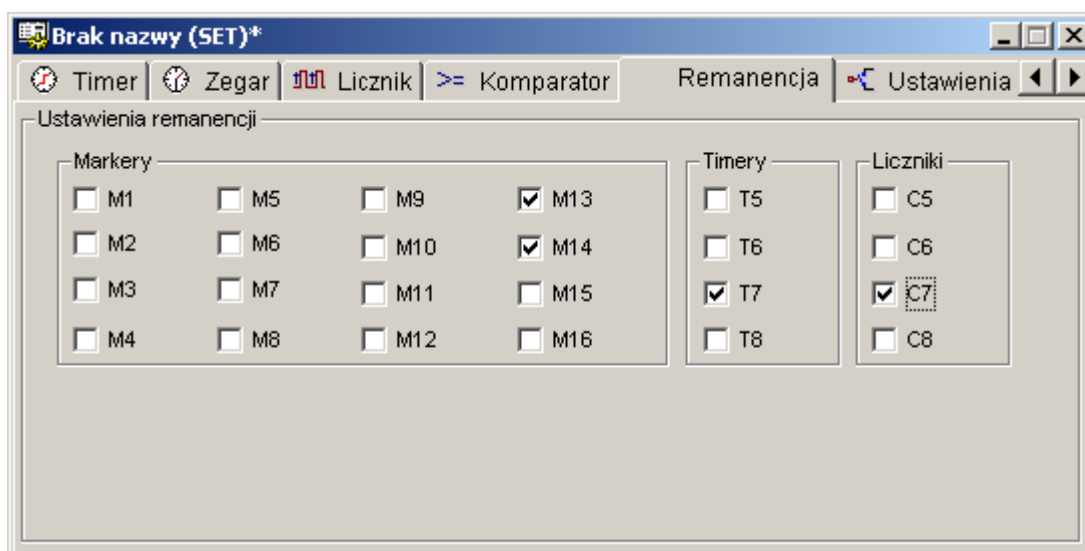
4.13. Wartości remanentne przekaźnika programowalnego

W procesach sterowania często zdarza się, iż potrzebne jest zachowanie danych po wyłączeniu zasilania. W przekaźniku programowalnym NEED można ustalić pewne „obszary” zasobów przekaźnika – zasoby remanentne, które mogą być podtrzymywane przy wyłączonym napięciu zasilającym lub po przełączeniu przekaźnika w tryb STOP. W tabeli 4.13.1. przedstawiono zasoby, które mogą być zdefiniowane jako remanentne.

Tab. 4.13.1. Zasoby remanentne w przekaźniku programowalnym NEED.

Zasoby remanentne	Zakres
Znaczniaki	M1– M16
Timery	T5 – T8
Liczniki	C5 – C8

Aby zdefiniować zasoby przekaźnika programowalnego jako remanentne należy w programie konfiguracyjnym PC Need zaznaczyć odpowiednie pola. Przykładowa konfiguracja wartości remanentnych przekaźnika została przedstawiona na rys. 4.13.2.



Rys. 4.13.2. Przykładowa konfiguracja zasobów remanentnych.

W powyższym przykładzie *Znaczniki* M13, M14, *Timer* 7 oraz *Licznik* 7 zostały skonfigurowane jako remanentne.

Ustalenie tej konfiguracji powinno się odbyć w trybie pracy STOP przekaźnika. Fabrycznie lub po operacji *RESET* zasoby remanentne są nieustawione.



Ustawienie remanencji może być przyczyną nieoczekiwanego działania programu ze względu na nieokreślone warunki początkowe. Przeładowanie programu powoduje wyzerowanie wartości remanentnych.

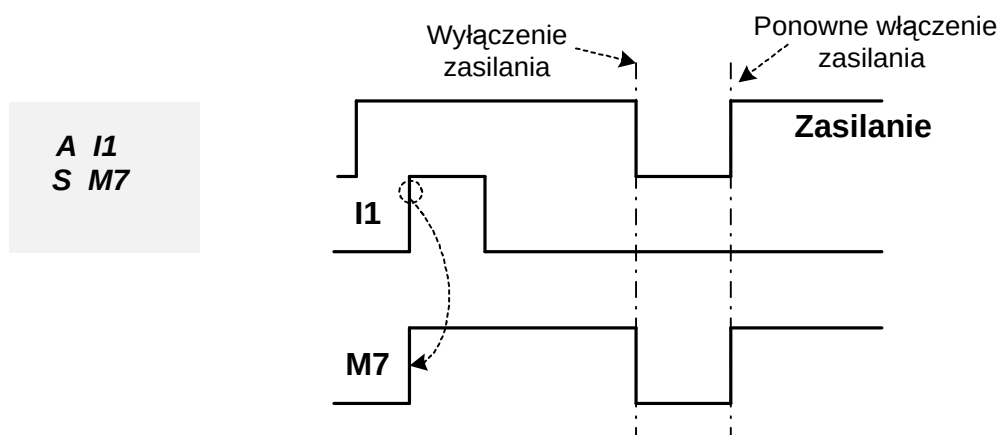
4.13.1. Uwagi dotyczące wartości remanentnych

1. Znaczniki.

Jeśli ustawimy dany *Znacznik* jako remanentny, to po wyłączeniu i ponownym załączeniu zasilania lub przejściu przekaźnika RUN → STOP → RUN, będzie on pamiętał swój stan logiczny sprzed wyłączenia.

Przykład:

Ustawmy *Znacznik* M7 jako remanentny.



Rys. 4.13.1.1. Działanie remanencji dla Znacznika M7.

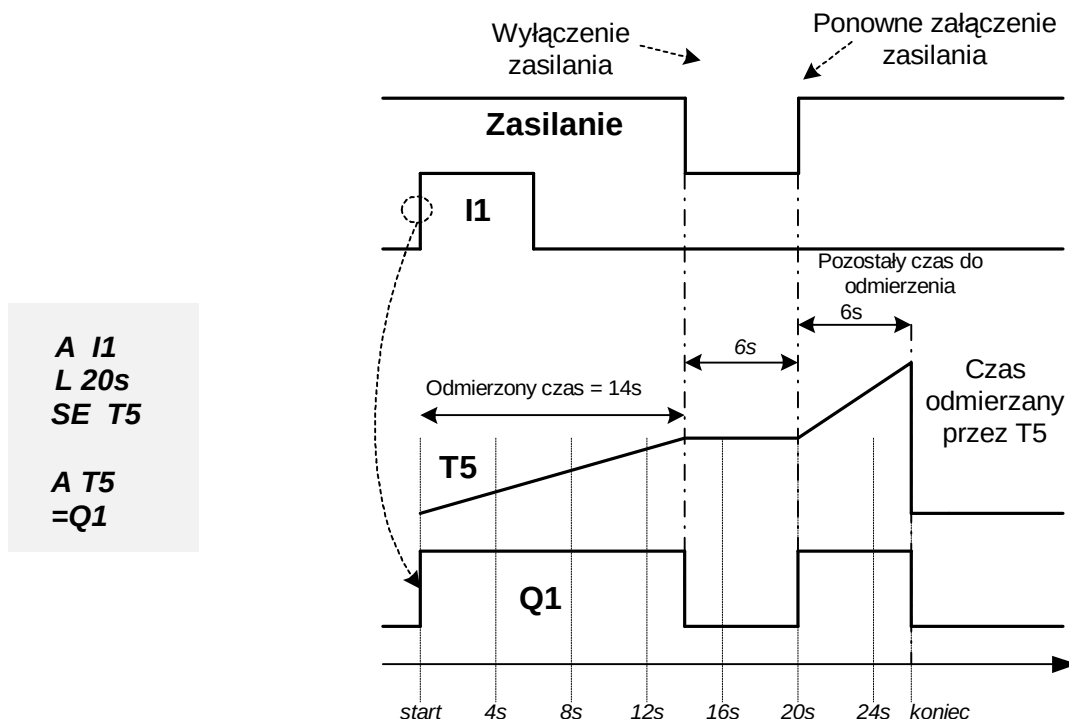
Po wyłączeniu i ponownym załączeniu zasilania, M7 ma stan wysoki, mimo, że I1='0'.

2. Timery.

Jeśli ustawimy dany *Timer* jako remanentny, to po wyłączeniu i ponownym załączeniu zasilania lub przejściu przekaźnika RUN → STOP → RUN, będzie on pamiętał swój stan logiczny oraz wartość czasu sprzed wyłączenia.

Przykład:

Ustawmy *Timer* T5 jako remanentny.



Rys.4.13.1.2. Działanie remanencji dla Timera T5.

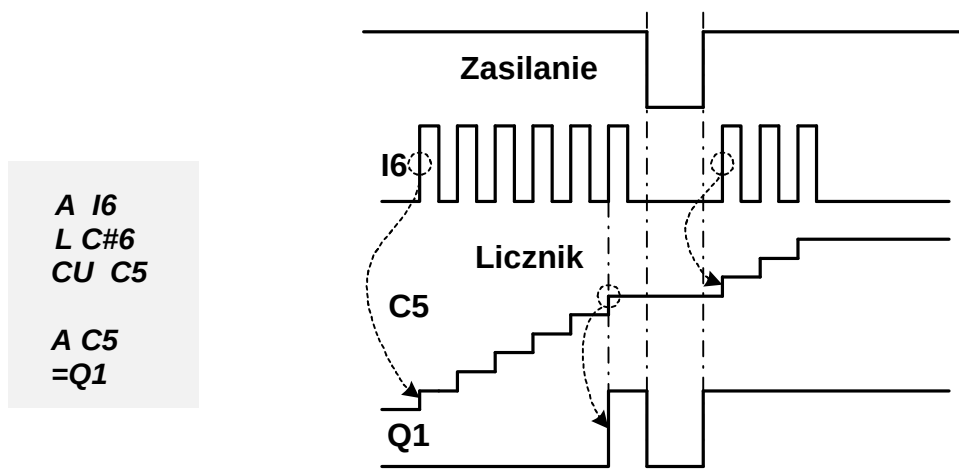
Po wyzwoleniu *Timer* T5 odmierza czas. Po wyłączeniu zasilania, w 14s odmierzony czas jest pamiętany, a po ponownym załączeniu zasilania, *Timer* kończy odmierzenie 20s – ustawia swoje wyjście w stan wysoki na pozostały czas 6s.

3. Liczniki.

Jeśli ustawimy dany *Licznik* jako remanentny, to po, np. wyłączeniu i ponownym załączeniu zasilania, będzie on pamiętał swój stan logiczny oraz ilość zliczonych impulsów przed wyłączeniem.

Przykład:

Ustawmy *Licznik* C5 jako remanentny.



Rys. 4.13.1.3. Działanie remanencji dla Licznika C5.

Impulsy, które pojawiają się na I6 zwiększają wartość *Licznika* C5. Po wyłączeniu i ponownym załączeniu zasilania, *Licznik* pamięta swoją wartość oraz stan wyjściowy sprzed wyłączenia. Gdy pojawią się następne impulsy wyzwalające, *Licznik* nie liczy od zera, ale od wartości zapamiętanej sprzed wyłączenia zasilania.

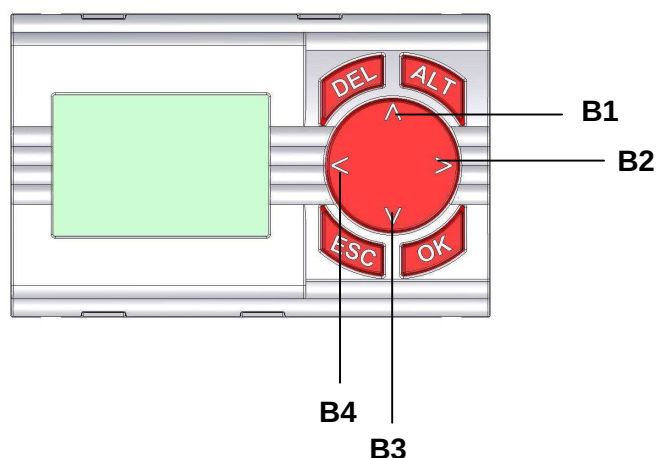


Licznik remanentny zliczy impuls, jeżeli stan na wejściu przed wyłączeniem zasilania był '0', a po załączeniu '1'. Licznik nieremanentny nie zliczy takiego impulsu.

Jeżeli przed wyłączeniem zasilania na wejściu *Licznika* był stan '1' i po załączeniu zasilania nadal utrzymywał się stan wysoki, to *Licznik* remanentny nie zliczy impulsu.

4.14. Przyciski klawiatury

Dla przekaźników NEED mogących współpracować z panelem LCD i klawiaturą można wyróżnić cztery przyciski B1 – B4.



Przycisk podlegają tym samym regułom co wejścia.

4.14.1. Przyciski jako wejścia cyfrowe normalnie otwarte

Symbole dla przycisków jako wejść cyfrowych normalnie otwartych.

STL	LAD
A B1 lub O B1 lub X B1	B1 

SYMBOL: **Bn**, gdzie n oznacza numer wejścia n=1..4 dla NEED-..-D

STANY LOGICZNE:

'1' – Przycisk jest wciśnięty.

'0' – Przycisk nie jest wciśnięty.

4.14.2. Przyciski jako wejścia cyfrowe normalnie zamknięte

Symbole przycisków jako wejść cyfrowych normalnie zamkniętych.

STL	LAD
AN B1 lub ON B1 lub XN B1	B1 

SYMBOL: **Bn**, gdzie n oznacza numer wejścia n=1..4 dla NEED-..-D

STANY LOGICZNE:

'0' – Przycisk jest wciśnięty.

'1' – Przycisk nie jest wciśnięty.



W przekaźnikach NEED-..-22-.. można korzystać z zasobów B1 – B4 niezależnie od tego, czy panel jest zamontowany, czy nie.

5. OPIS JĘZYKÓW PROGRAMOWANIA

Przełącznik NEED możemy programować używając dwóch języków programowania. Zdefiniowane one zostały w taki sposób, aby programowanie przełącznika było jak najbardziej efektywne oraz, aby użytkownik miał możliwość wyboru wygodnego dla siebie języka programowania. I tak do opisywania zadań sterowania możemy korzystać z:

- języka tekstowego – Lista Instrukcji (STL)
- języka graficznego – Schemat drabinkowy (LAD)

5.1. Programowanie w języku tekstowym STL

Język tekstowy STL (ang. *Statement List*) jest zbiorem instrukcji obejmujących operacje logiczne, relacje, jak również funkcje przerzutników, *Timerów*, *Liczników* itp., które umożliwiają odpowiednie zaprogramowanie przełącznika. Używanie języka tekstowego do programowania przełącznika NEED jest bardzo efektywne i najbardziej przybliża kod wynikowy do struktury wewnętrznej programu.

5.1.1. Struktura programu STL

Program w STL jest ciągiem kolejno wykonywanych instrukcji.

Każda instrukcja składa się z dwóch elementów:

- 1) Symbolu Instrukcji – identyfikatora (kodu), który w języku STL pełni rolę tzw. słowa kluczowego.
- 2) Argumentu – czyli zmiennej (wyjątek stanowi instrukcja SET i CLR).

<kod>	<argument>
A, A(, AN, AN(, O, O(, ON, ON(, X, X(, XN, XN(, S, R, =, FP, SD, SF, SE, SL, CU, CD, L, SET, CLR	I, Q, M, MT, MDIR, H, A, T, C, HC1, H, L–Licznik ¹⁾ , L–Timer ²⁾

1- L–Licznik – jest liczbą zadanych zliczeń dla Licznika.

2- L–Timer – jest zadany czasem do odmierzenia dla Timera.

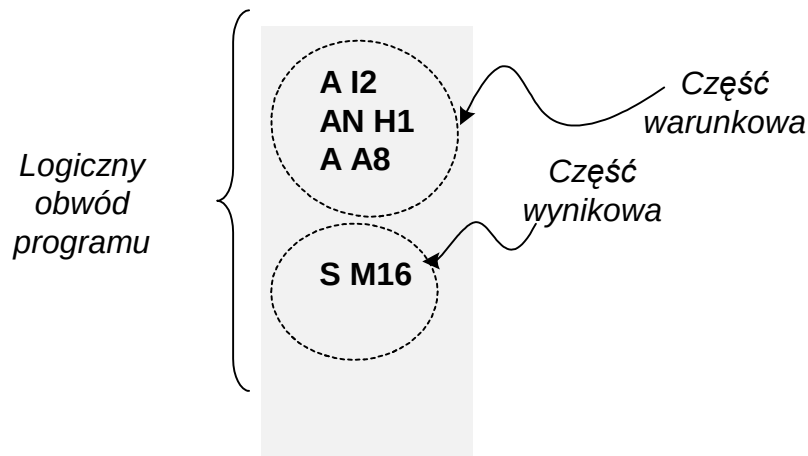
W zapisie logicznym pewnych sekwencji, z których składa się program możemy wyróżnić część warunkową (poprzedzającą) i część wynikową (następującą). Innymi słowy: jeśli spełnione są pewne warunki zapisane za pomocą określonych instrukcji i zmiennych, to ta sytuacja wywoła pewien skutek, który także jest określony za pomocą odpowiednich instrukcji i zmiennych. Taki zestaw części warunkowej i wynikowej nazywamy obwodem. Każda linia programu powinna zaczynać się od dozwolonej instrukcji, po której powinien występować argument. Dozwolony więc jest zapis typu:

```
A I1
A I2
S Q5
```

Zabroniony natomiast jest zapis typu:

```
I1
A I2
S Q5
```

W pierwszej linii przykładu powyżej brak jest nazwy instrukcji - może nią być instrukcja A, A(, AN, AN(, O, O(, ON, ON(, X, X(, XN, XN(.



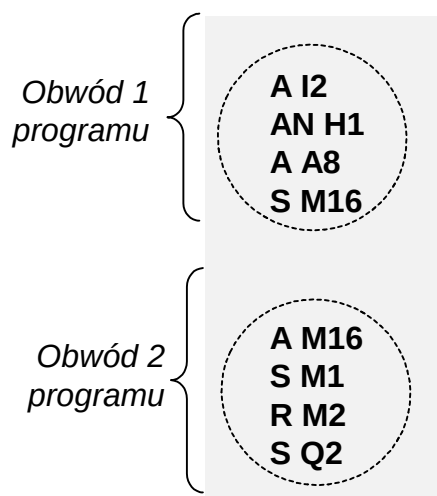
Rys. 5.1.1.1. Struktura programu STL.

Przykład:

1)	A I1	}	Część warunkowa (sprawdza relację: I1 AND A2)
2)	A A2		
3)	S Q4	}	Część wynikowa
4)	R Q1		

Część warunkową w tym przykładzie będą stanowić instrukcje zapisane w liniach 1,2. Jeśli na wejściu I1 i wyjściu Komparatora A2 będzie stan wysoki, to wyjście Q4 zostanie ustawione (stan '1'), a wyjście Q1 zostanie zresetowane (stan '0'). Instrukcje 'S Q4' oraz 'R Q1' stanowią część wynikową.

Instrukcje: A, A(, AN, AN(, O, O(, ON, ON(, X, X(, XN, XN(składają się na część warunkową obwodu a instrukcje S, R, =, FP, SD, SF, SL, SE, CD, CU tworzą część wynikową obwodu. Każdy oddzielny obwód powinien zaczynać się częścią warunkową a kończyć częścią wynikową.

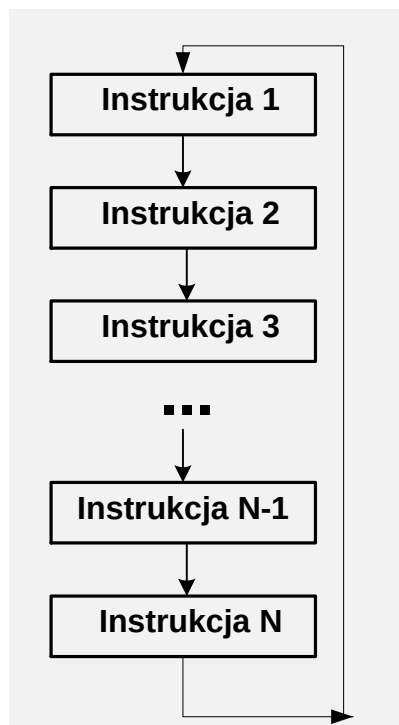


Rys.5.1.1.2. Przykład dwóch obwodów w STL.

W przekaźniku programowalnym znajduje się tylko jeden program, którego nie można dzielić na żadne wywoływane podprogramy.

Procesor w sterowniku realizuje poszczególne instrukcje kolejno, poczynając od pierwszej a kończąc na ostatniej. Po wykonaniu ostatniej instrukcji cykl programu powtarza się.

Przetwarzanie programu sterownika przedstawiono na rys. 5.1.1.3.



Rys. 5.1.1.3. Cykliczne przetwarzanie programu STL.

W tab. 5.1.1. przedstawiono wykaz wszystkich dostępnych instrukcji STL.

Tab. 5.1.1. Instrukcje STL.

STL Instrukcja	Opis	Operandy
A	Instrukcja AND	I,Q,M,MDIR,MT,A,H,HC1,C,T,B
A(	Instrukcja nawiasów AND	
AN	Instrukcja AND NOT	I,Q,M,MDIR,MT,A,H,HC1,C,T,B
AN(	Instrukcja nawiasów AND NOT	
O	Instrukcja OR	I,Q,M,MDIR,MT,A,H,HC1, C,T,B
O(	Instrukcja nawiasów OR	
ON	Instrukcja OR NOT	I,Q,M,MDIR,MT,A,H,HC1,C,T,B
ON(	Instrukcja nawiasów OR NOT	
X	Instrukcja XOR	I,Q,M,MDIR,MT,A,H,HC1,C,T,B
X(	Instrukcja nawiasów XOR	
XN	Instrukcja XOR NOT	I,Q,M,MDIR,MT,A,H,HC1,C,T,B
XN(	Instrukcja nawiasów XOR NOT	
S	Instrukcja ustawiająca	Q,M,MT
R	Instrukcja resetująca	Q,M,MT,T,C,HC1
=	Instrukcja przyporządkowująca	Q,M,MT
FP	Przekaźnik impulsowy	Q,M,MT
L	Instrukcja ładująca	Stała wartość operandu

Tab. 5.1.1. Instrukcje STL- c.d.

STL Instrukcja	Opis	Operandy
SD	Timer – Opóźnione załączenie	T
SE	Timer – Opóźnione wyłączenie	T
SF	Timer – Pojedynczy impuls	T
SL	Timer – Impulsy	T
CU	Licznik – Zliczanie w górę	C, HC1
CD	Licznik – Zliczanie w dół	C, HC1
SET	Instrukcja zawsze ustawiająca	
CLR	Instrukcja zawsze kasująca	

5.1.1.1. Nazwy symboliczne

Dla przekaźników NEED istnieje możliwość przypisania w projekcie nazw symbolicznych do zmiennych. Ułatwia to analizę programu i czyni go bardziej przejrzystym.

Aby skojarzyć zmienną z nazwą symboliczną należy użyć wyrażenia o następującej składni:

. DEFINE < nazwa symboliczna > = < zmienna >

Potem można używać zamiast adresu zmiennej np. Q1, I11 nazwy symbolicznej, która musi być poprzedzona znakiem „%” np.:

```
.DEFINE Pompa = Q1
```

```
.DEFINE Awaria = I11
```

```
A %Awaria
```

```
R %Pompa
```

W nazwach symbolicznych nie są rozróżniane małe i duże litery.

Nazwami symbolicznymi nie mogą być także nazwy elementów zasobów przekaźnika oraz instrukcji. Nazwy symboliczne muszą zaczynać się od litery lub znaku podkreślenia (_), po którym następuje ciąg liter lub cyfr. Nie można używać w nazwie znaku spacji.

Nazwy symboliczne nie mogą zaczynać się cyfrą. Mogą zawierać maksymalnie 30 znaków.

5.1.2. Opis Instrukcji STL

5.1.2.1. Instrukcja AND

SYMBOL – **A**

Instrukcja 'A' jest logiczną instrukcją typu AND.

SKŁADNIA:

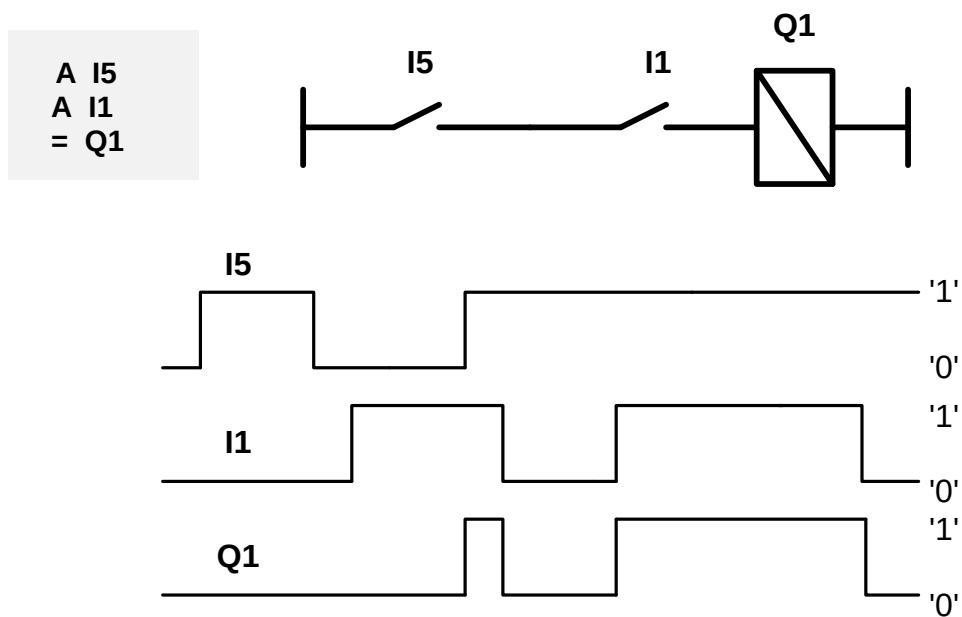
A < I, Q, M, MDIR, MT, A, H, C, HC1, T, B >

Czas wykonania instrukcji: 6μs.

Przykład:

STL

Schemat przekaźnikowy



Rys. 5.1.2.1. Przykładowe przebiegi czasowe na wejściach I5, I1 oraz na wyjściu Q1.

Powyższy przykład realizuje połączenie szeregowe.

wyjście Q1 zostanie ustawione (stan '1'), gdy oba wejścia będą miały stan wysoki – zgodnie z zasadami działania funkcji AND.

5.1.2.2. Instrukcja nawiasów AND

SYMBOL – **A(**

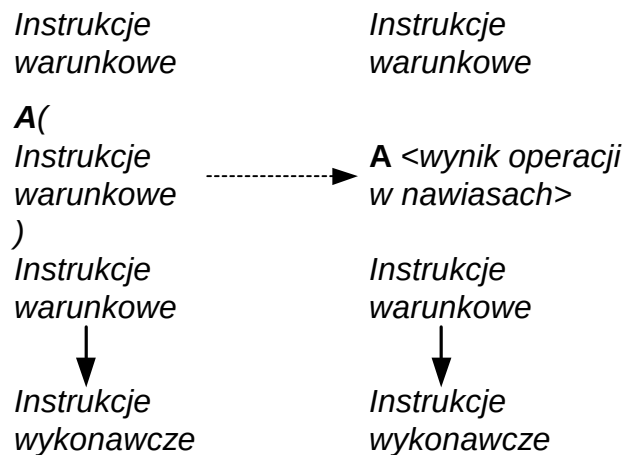
Instrukcja 'A(' jest logiczną instrukcją typu AND, której operandem jest wynik operacji logicznych w nawiasie.

SKŁADNIA:

A(**Instrukcje warunkowe****)**

Czas wykonania instrukcji: 6μs.

Na rys. 5.1.2.2.1. przedstawiono zasadę działania instrukcji 'A(' – wszystkie inne instrukcje nawiasów działają na tej samej zasadzie.



Rys. 5.1.2.2.1. Zasada działania instrukcji nawiasów 'A('.

Wykonujemy operacje w nawiasach. W wyniku tych operacji logicznych otrzymujemy jakiś rezultat ('0' lub '1'), którego używamy do następnych operacji logicznych np. dla programu:

```

A I1
A(
O M1
O M2
)
=Q1 //Jest to równoważne operacji logicznej I1 AND (M1 OR M2) = Q1
  
```

oraz stanów logicznych: M1='0', M2='0', I1='1'.

Można zapisać:

```

A I1      A '1'
A(        A '0' //bo '0' O '0' = '0'
O M1
O M2
)
=Q1      = '0'
  
```

Czyli dla analizowanych stanów wyjście Q1 będzie w stanie '0'. Natomiast dla stanów M1='1', M2='0', I1='1' otrzymujemy:

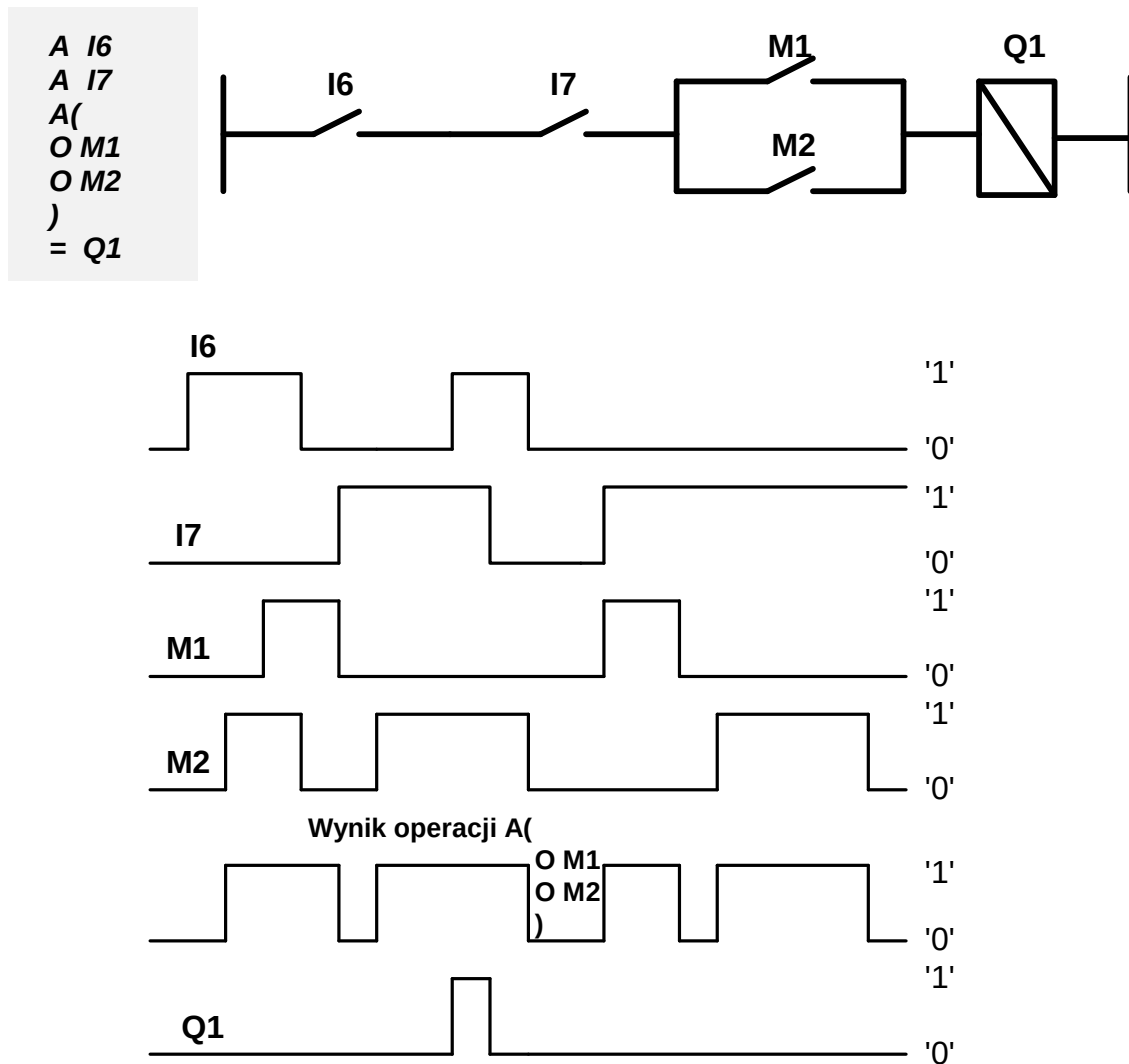
```

A I1      A '1'
A(        A '1' //bo '1' O '0' = '1'
O M1
O M2
)
=Q1      = '1'
  
```

Przykład:

STL

Schemat przekaźnikowy



Rys. 5.1.2.2.2. Przykładowe przebiegi czasowe na wejściach I6, I7, M1, M2 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy wejścia I6, I7 będą miały stan wysoki oraz gdy przynajmniej jeden ze Znaczników M1 lub M2, będzie w stanie '1'.

5.1.2.3. Instrukcja AND NOT

SYMBOL – **AN**

Instrukcja 'AN' jest logiczną instrukcją typu AND NOT (instrukcja AND z zanegowanym stanem operandu).

SKŁADNIA:

AN < I, Q, M, MDIR, MT, A, H, HC1, C, T, B >

Czas wykonania instrukcji: 6µs.

Przykład:

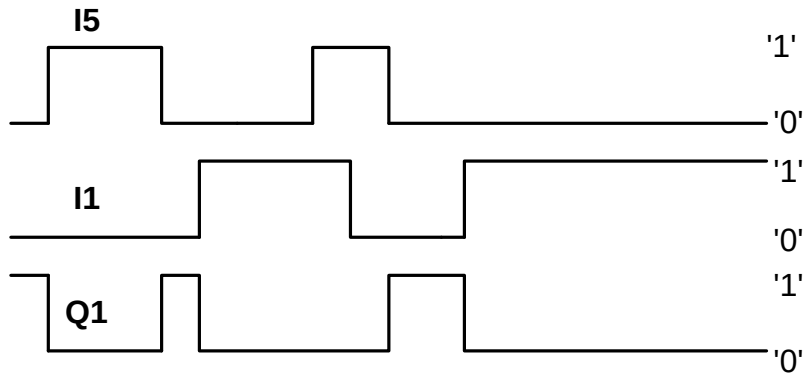
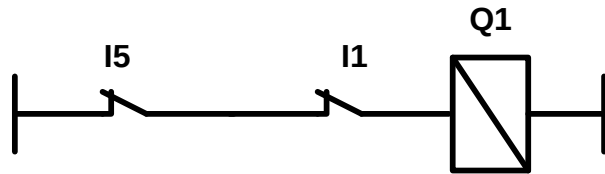
STL

Schemat przekaźnikowy

```

AN I5
AN I1
= Q1

```



Rys. 5.1.2.3. Przykładowe przebiegi czasowe na wejściach I5, I1 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy oba wejścia będą miały stan niski ('0').

5.1.2.4. Instrukcja nawiasów AND NOT

SYMBOL – **AN(**

Instrukcja 'AN(' jest logiczną instrukcją typu AND NOT, której operandem jest wynik operacji logicznych w nawiasie.

SKŁADNIA:

AN(

Instrukcje warunkowe

)

Czas wykonania instrukcji: 6μs.

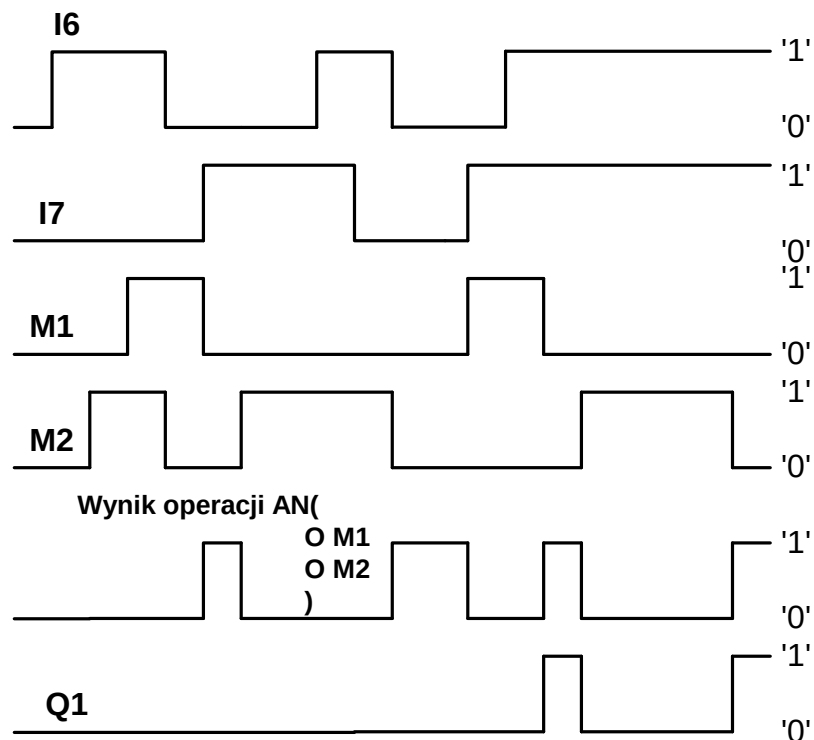
Przykład:

STL

```

A I6
A I7
AN(
O M1
O M2
)
= Q1

```



Rys. 5.1.2.4. Przykładowe przebiegi czasowe na wejściach I6, I7, M1, M2 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy wejścia I6, I7 będą miały stan wysoki oraz gdy oba Znaczniki M1 i M2 będą w stanie '0'. Jest to równoważne operacji logicznej $I6 \text{ AND } I7 \text{ AND } (M1 \text{ OR } M2) = Q1$.

5.1.2.5. Instrukcja OR

SYMBOL – O

Instrukcja 'O' jest logiczną instrukcją typu OR

SKŁADNIA:

O < I, Q, M, MDIR, MT, A, H, HC1, C, T, B >

Czas wykonania instrukcji: 6µs.

Przykład:

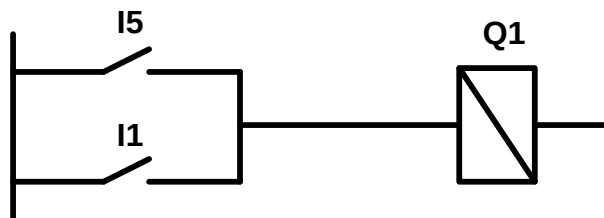
STL

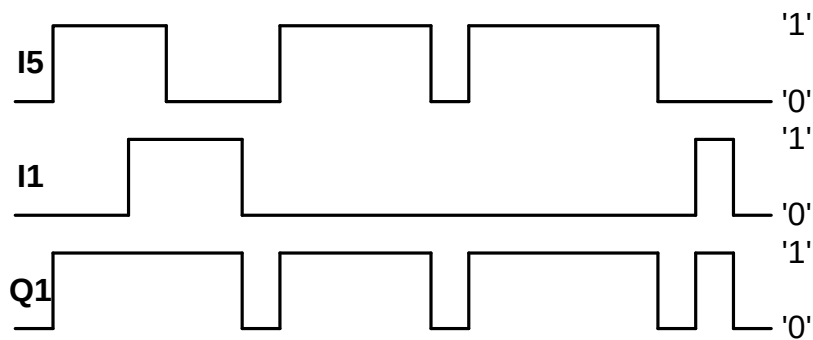
Schemat przekaźnikowy

```

O I5
O I1
= Q1

```





Rys. 5.1.2.5. Przykładowe przebiegi czasowe na wejściach I5, I1 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy jedno z wejść będzie miało stan wysoki ('1'). Realizacja połączenia równoległego.

5.1.2.6. Instrukcja nawiasów OR

SYMBOL – **O(**

Instrukcja 'O(' jest logiczną instrukcją typu OR, której operandem jest wynik operacji logicznych w nawiasie.

SKŁADNIA:

O(

Instrukcje warunkowe

)

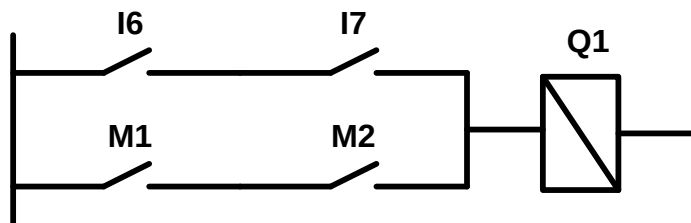
Czas wykonania instrukcji: 6μs.

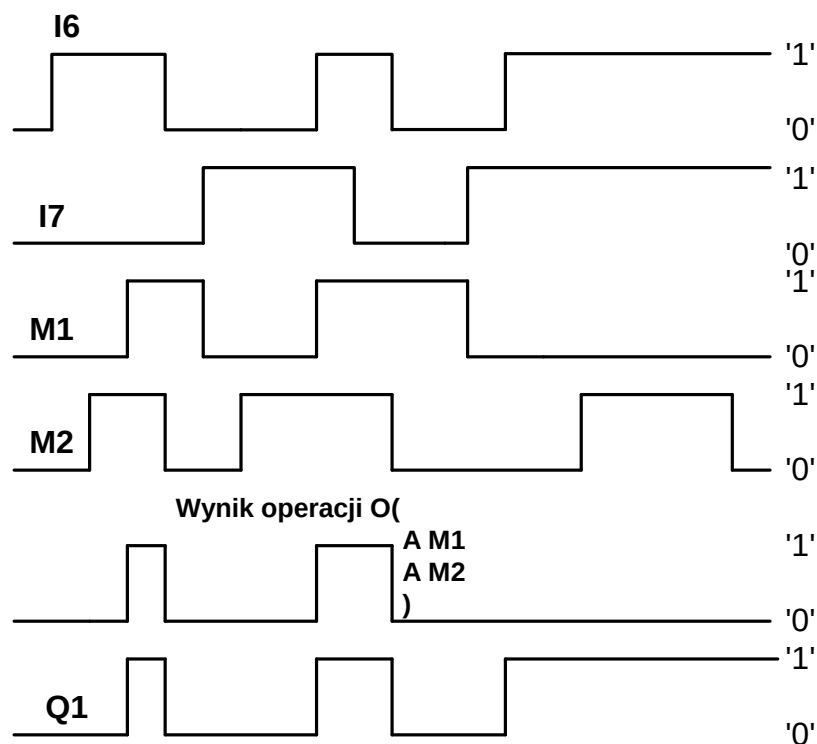
Przykład:

STL

Schemat przekaźnikowy

```
A I6
A I7
O(
A M1
A M2
)
= Q1
```





Rys. 5.1.2.6. Przykładowe przebiegi czasowe na I6, I7, M1, M2 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy wejścia I6, I7 będą miały stan wysoki lub, gdy oba Znaczniki M1 i M2 będą w stanie '1'.

5.1.2.7. Instrukcja OR NOT

SYMBOL – **ON**

Instrukcja 'O' jest logiczną instrukcją typu OR NOT (instrukcja OR z zanegowanym stanem operandu).

SKŁADNIA:

ON < I, Q, M, MDIR, MT, A, H, HC1, C, T, B >

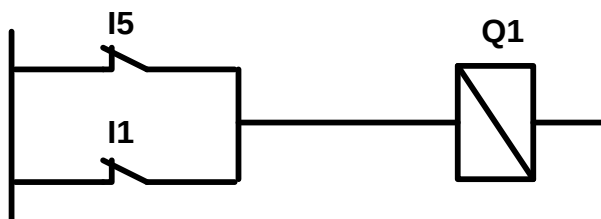
Czas wykonania instrukcji: 6µs.

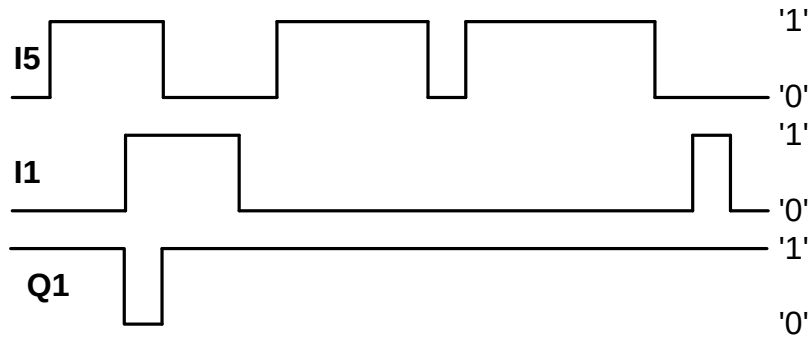
Przykład:

STL

Schemat przekaźnikowy

```
ON I5
ON I1
= Q1
```





Rys. 5.1.2.7. Przykładowe przebiegi czasowe na wejściach I5, I1 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy przynajmniej jedno z wejść będzie miało stan niski ('0').

5.1.2.8. Instrukcja nawiasów OR NOT

SYMBOL – **ON(**

Instrukcja 'ON(' jest logiczną instrukcją typu OR NOT wyniku operacji logicznych w nawiasie.

SKŁADNIA:

ON(

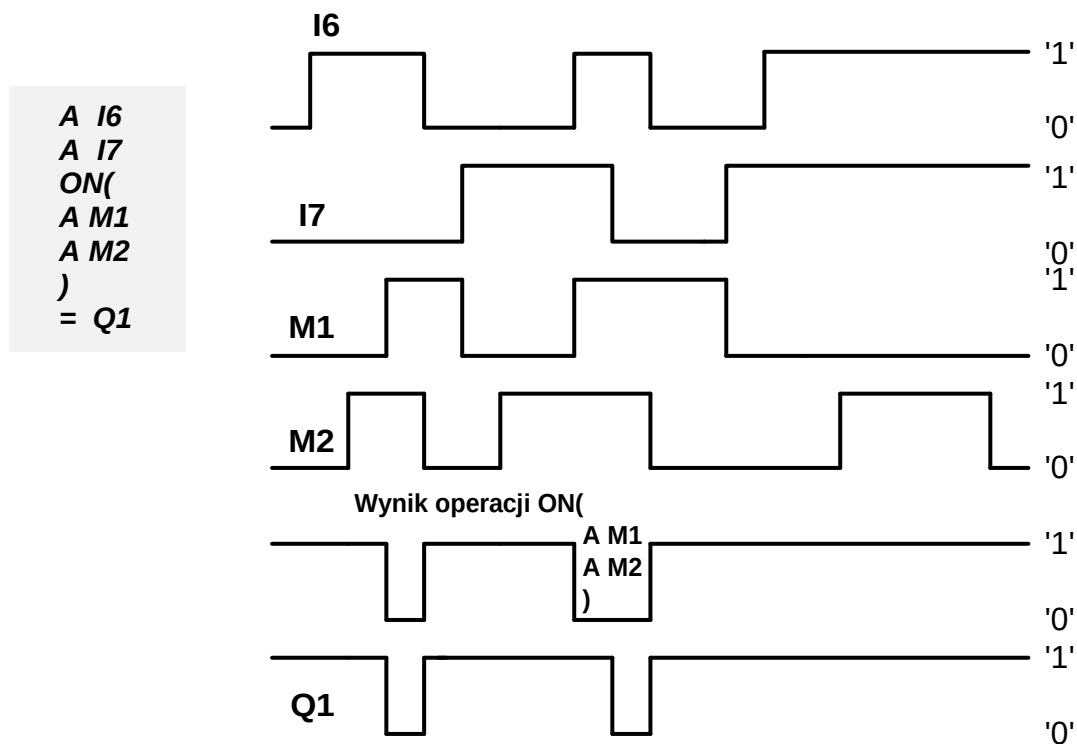
Instrukcje warunkowe

)

Czas wykonania instrukcji: 6µs.

Przykład:

STL



Rys. 5.1.2.8. Przykładowe przebiegi czasowe na I6, I7, M1, M2 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy wejścia I6, I7 będą miały stan wysoki lub, gdy

jeden ze Znaczników M1 lub M2 będzie w stanie '0'.

5.1.2.9. Instrukcja XOR

SYMBOL – X

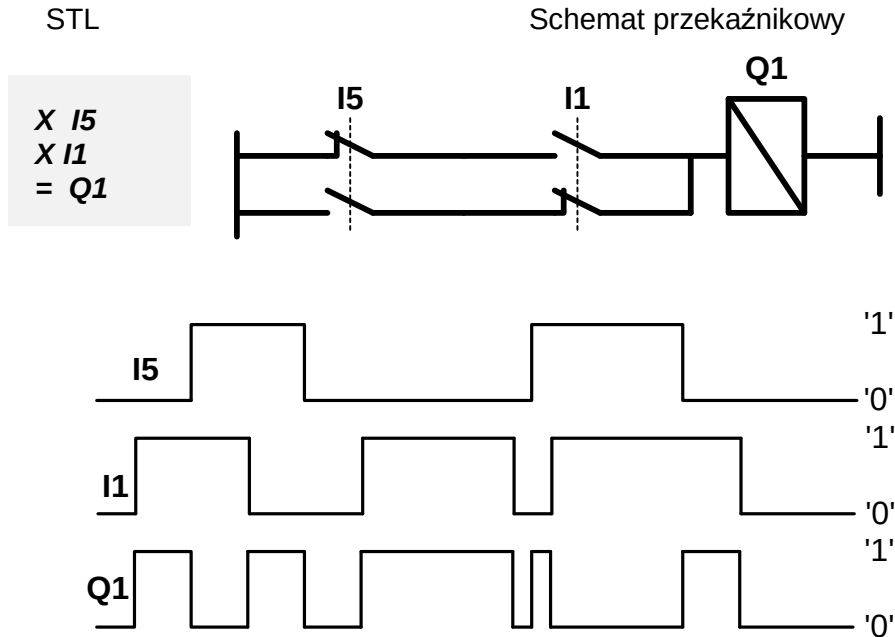
Instrukcja 'X' jest logiczną instrukcją typu XOR.

SKŁADNIA:

X <I, Q, M, MDIR, MT, A, H, HC1, C, T, B >

Czas wykonania instrukcji: 6μs.

Przykład:



Rys. 5.1.2.9. Przykładowe przebiegi czasowe na wejściach I5, I1 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy na wejściach I5 oraz I1 będą panowały stany przeciwne (I5='1' i I1='0' lub I5='0' i I1='1').

5.1.2.10. Instrukcja nawiasów XOR

SYMBOL – X(

Instrukcja 'X(' jest logiczną instrukcją typu XOR, której operandem jest wynik operacji logicznych w nawiasie.

SKŁADNIA:

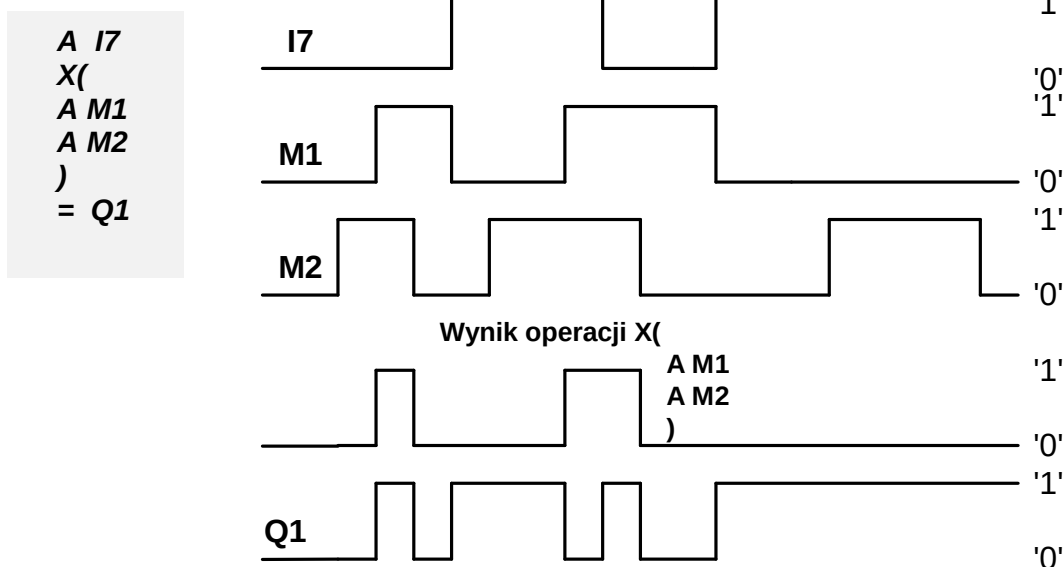
X(

Instrukcje warunkowe

)

Czas wykonania instrukcji: 6μs.

Przykład:
STL



Rys. 5.1.2.10. Przykładowe przebiegi czasowe na I7, M1, M2 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1') zgodnie z działaniem funkcji XOR tzn.:

Q1=1 dla I7=1 i któryś ze Znaczników ustawiony jest w stanie '0'.

Q1=1 dla I7=0 i oba Znaczniki są ustawione w stan wysoki ('1').

5.1.2.11. Instrukcja XOR NOT

SYMBOL – **XN**

Instrukcja 'XN' jest logiczną instrukcją typu XOR NOT.

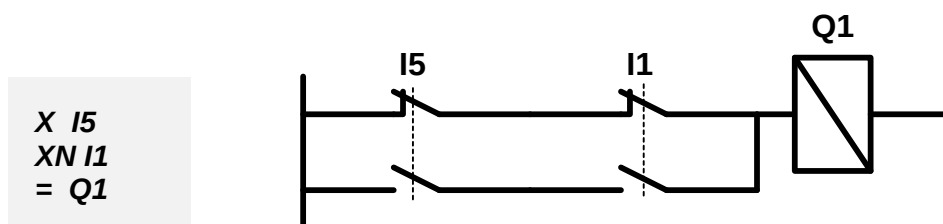
SKŁADNIA:

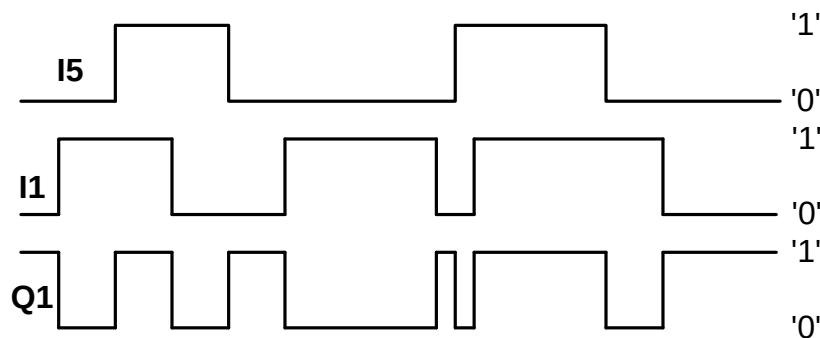
X < I, Q, M, MDIR, MT, A, H, HC1, C, T, B >

Czas wykonania instrukcji: 6µs.

Przykład:
STL

Schemat przekaźnikowy





Rys. 5.1.2.11. Przykładowe przebiegi czasowe na wejściach I5, I1 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy na wejściach I5 oraz I1 będą panowały takie same stany logiczne (I5='0' i I1='0' lub I5='1' i I1='1').

5.1.2.12. Instrukcja nawiasów *XOR NOT*

SYMBOL – **XN(**

Instrukcja 'XN(' jest logiczną instrukcją typu *XOR NOT* wyniku operacji logicznych w nawiasie.

SKŁADNIA:

XN(

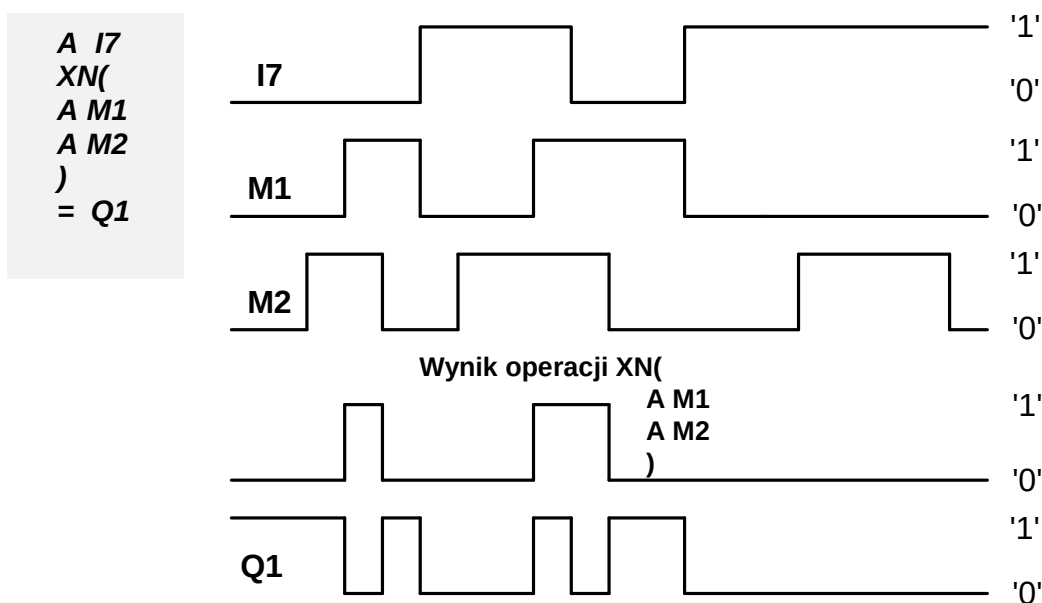
Instrukcje warunkowe

)

Czas wykonania instrukcji: 6μs.

Przykład:

STL



Rys. 5.1.2.12. Przykładowe przebiegi czasowe na I7, M1, M2 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1') zgodnie z działaniem funkcji *XOR NOT* tzn.:

Q1=1 dla I7=1 i oba Znaczniki M1, M2 są w stanie wysokim ('1').

Q1=1 dla I7=0 i któryś ze Znaczników jest w stanie niskim ('0').

5.1.2.13. Instrukcja ustawiająca S

SYMBOL – **S**

Instrukcja 'S' jest logiczną instrukcją ustawiającą argument w stan wysoki ('1').

SKŁADNIA:

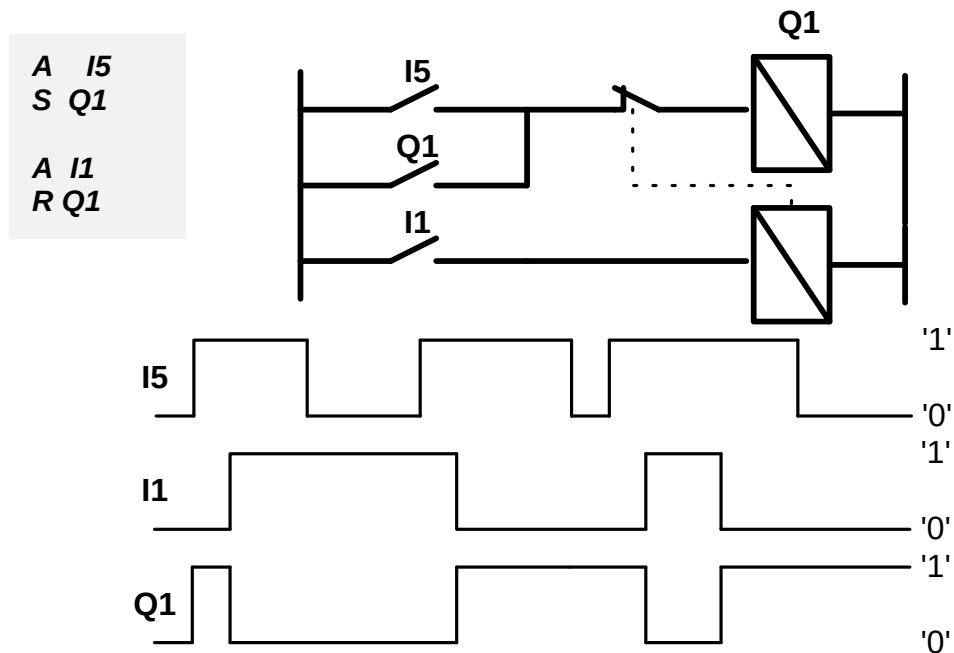
S < Q, M, MT >

Czas wykonania instrukcji: 6,5µs.

Przykład:

STL

Schemat przekąźnikowy



Rys. 5.1.2.13. Przykładowe przebiegi czasowe na wejściach I5, I1 oraz na wyjściu Q1.

Wyjście Q1 zostanie ustawione (stan '1'), gdy wejście I5 będzie miało stan wysoki ('1'). Będzie ono pozostawało w stanie wysokim tak długo, aż nie zostanie ustawiony stan niski ('0') instrukcją 'R' – wejście I1.

5.1.2.14. Instrukcja kasująca R (Reset)

SYMBOL – **R**

Instrukcja 'R' jest logiczną instrukcją ustawiającą argument w stan niski ('0').

SKŁADNIA:

R < Q, M, MT, T, C, HC1 >

Czas wykonania instrukcji: 6,5µs.

Przykład: patrz instrukcja „S”.

5.1.2.15. Instrukcja przyporządkowująca =

SYMBOL – **=**

Instrukcja '=' jest logiczną instrukcją, w której argument przyjmuje wartość (stan '0' lub '1') zależną od wyniku wcześniejszych operacji logicznych.

SKŁADNIA:

= < Q, M, MT >

Czas wykonania instrukcji: 6,7µs.

Przykład:

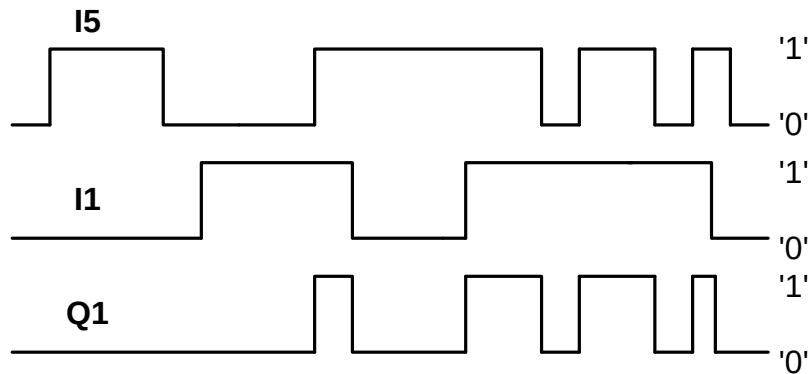
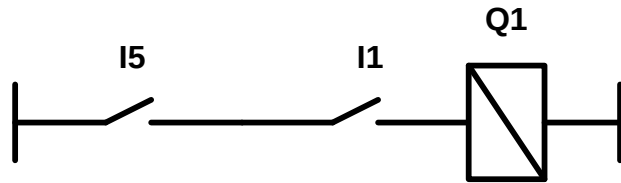
STL

Schemat przekaźnikowy

```

A I5
A I1
= Q1

```



Rys. 5.1.2.15. Przykładowe przebiegi czasowe na wejściach I5, I1 oraz na wyjściu Q1.

Stan wyjścia Q1 zależy od wcześniejszych operacji logicznych tzn. przyjmuje stan '0', gdy stan któregoś z wejść jest '0' lub przyjmuje stan '1', gdy stany obu wejść są równe '1'.

5.1.2.16. Instrukcja Przekaznik impulsowy FP

Przekaznik impulsowy pełni rolę przerzutnika wyzwalanego zboczem narastającym. Każdy narastający impuls zmienia stan wyjścia na przeciwny.

SYMBOL – **FP**

SKŁADNIA:

FP <Q, M, MT>

Czas wykonania instrukcji: 5,9µs.

Przykład:

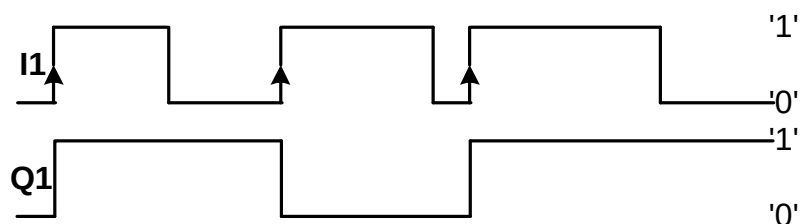
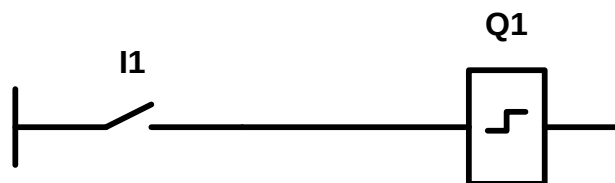
STL

Schemat przekaźnikowy

```

A I1
FP Q1

```



Rys. 5.1.2.16. Przykładowe przebiegi czasowe na wejściu I1 oraz na wyjściu Q1.

Jeżeli wyjście Q1 pozostaje w stanie niskim i na wejściu I1 pojawi się dodatnie zbocze sterujące, to wyjście Q1 zostanie ustawione w stan wysoki.
 Jeżeli wyjście Q1 pozostaje w stanie wysokim i na Wejściu I1 pojawi się dodatnie zbocze sterujący, to wyjście Q1 zostanie ustawione w stan niski.

5.1.2.17. Instrukcje Timerów

5.1.2.17.1. Timer Opóźnione załączenie (ON-DELAYED)

Timer realizuje funkcję opóźnione załączenie.

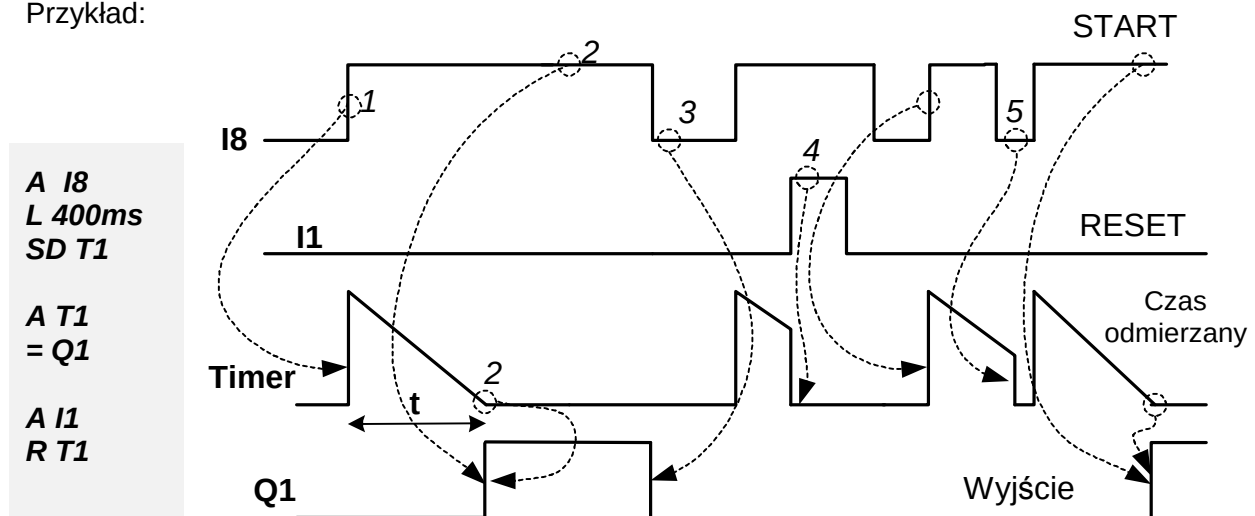
SYMBOL – **SD**

SKŁADNIA:

SD <T>

Czas wykonania instrukcji: 14,1µs.

Przykład:



Rys. 5.1.2.17.1. Przykładowe przebiegi czasowe sygnałów ilustrujące działanie Timera SD.

1.
Wejście I8 pełni rolę wejścia wyzwalającego (Triggera). Instrukcja ('L') ładująca określoną wartość czasu do odmierzenia, powinna się znaleźć bezpośrednio przed instrukcją Timera (SD).
Odmierzanie czasu następuje po wykonaniu instrukcji aktywacji Timera SD (zbocze narastające na wejściu I8).
2.
Po upływie czasu $t=400\text{ms}$ następuje ustawienie wyjścia Q1 w stan wysoki ('1').
Jednocześnie na wejściu wyzwalającym I8 powinien utrzymywać się sygnał wysoki ('1').
3.
Jeśli na wejściu Triggera I8 pojawi się stan niski, następuje skasowanie licznika odmierzanego czasu Timera T1, a wyjście Q1 zostanie ustawione w stan niski ('0').
4.
Jeśli na wejściu I1 resetującym T1 pojawi się stan wysoki następuje automatyczne skasowanie licznika odmierzanego czasu Timera T1, a wyjście Q1 zostanie ustawione w stan niski ('0').
Jeżeli instrukcja „L” nie zostanie użyta, to czas do odmierzenia przez T1 zostanie ustalony na podstawie pliku konfiguracyjnego „*.set” (okna Ustawień w programie PC Need).

5.1.2.17.3. Timer Pojedynczy impuls (SINGLE PULSE)

Timer realizuje funkcję pojedynczego impulsu.

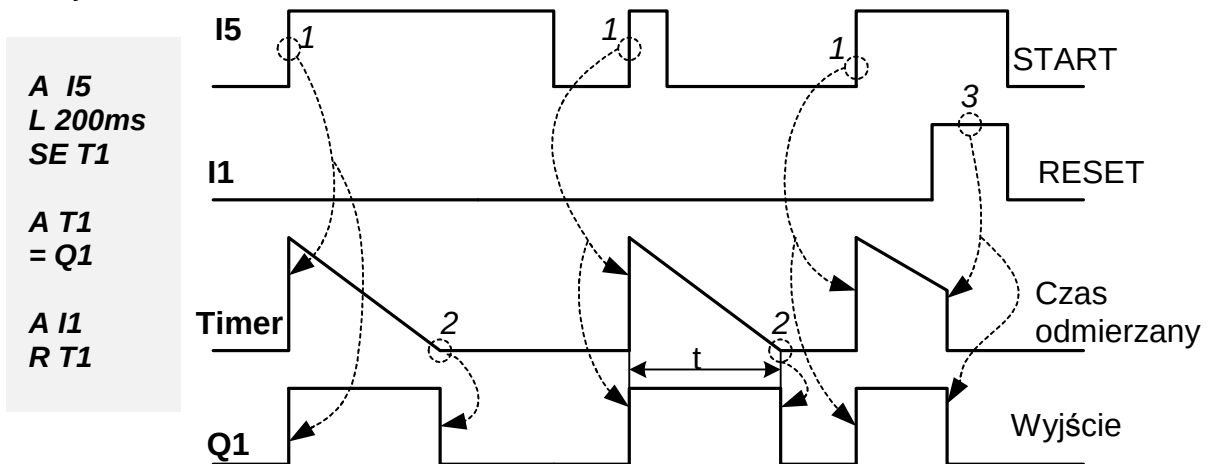
SYMBOL – **SE**

SKŁADNIA:

SE <T>

Czas wykonania instrukcji: 18,7µs.

Przykład:



Rys. 5.1.2.17.3. Przykładowe przebiegi czasowe sygnałów ilustrujące działanie Timera SE.

1.

Wejście I5 pełni rolę wejścia wyzwalającego (Triggera). Instrukcja ('L') ładująca określoną wartość czasu do odmierzenia powinna się znaleźć bezpośrednio przed instrukcją Timera SE.

Odmierzanie czasu następuje po wykonaniu instrukcji aktywacji Timera (narastające zbocze na wejściu I5).

2.

Przez $t=200\text{ms}$ wyjście Q1 będzie ustawione w stanie wysokim ('1'). Stan ten może zostać przedłużony, gdy nastąpi kolejne wyzwolenie na wejściu Trigger. Po odmierzeniu zadanej wartości czasu wyjście Timera powraca do stanu niskiego ('0'), czyli Q1 przyjmuje stan niski.

3.

Jeśli na wejściu I1 resetującym T1 pojawi się stan wysoki następuje skasowanie licznika odmierzanego czasu i wyjścia Timera T1.

Jeżeli instrukcja „L” nie zostanie użyta, to czas do odmierzenia przez T1 zostanie ustalony na podstawie pliku konfiguracyjnego „*.set” (okna Ustawień w programie PC Need).

5.1.2.17.4. Timer Impulsy (FLASHING)

Timer realizuje funkcję generatora fali prostokątnej o wypełnieniu 50 %.

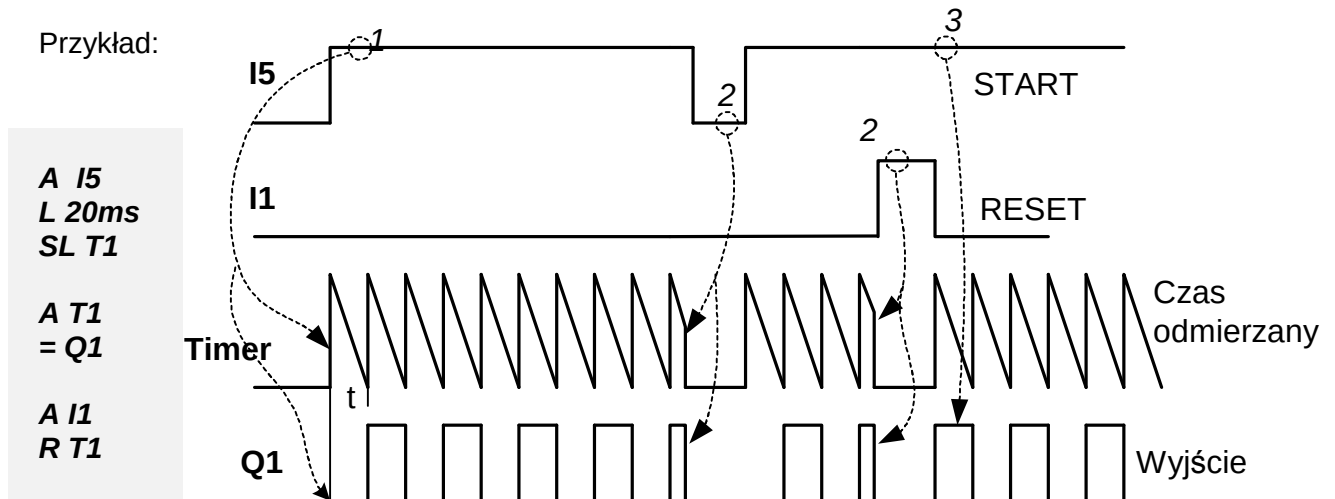
SYMBOL – **SL**

SKŁADNIA:

SL <T>

Czas wykonania instrukcji: 18,7μs.

Przykład:



Rys. 5.1.2.17.4. Przykładowe przebiegi czasowe sygnałów ilustrujące działanie Timera SL.

1.

Wejście I5 pełni rolę wejścia wyzwalającego (Triggera). Instrukcja ładująca określoną wartość czasu do odmierzenia powinna się znaleźć bezpośrednio przed instrukcją Timera SL.

Odmierzanie czasu następuje po wykonaniu instrukcji aktywacji Timera (poziom wysoki '1' na wejściu wyzwalającym I5).

Przez $t=20\text{ms}$ wyjście Q1 będzie ustawione w stanie niskim ('0'), a następnie przez kolejne 20ms będzie w stanie wysokim ('1'). Sytuacja ta będzie się powtarzała tak długo, jak długo na wejściu I5 będzie stan wysoki lub do momentu wystąpienia wysokiego stanu na wejściu resetującym I1.

2.

Jeśli na wejściu I5 (Trigger) pojawi się stan niski ('0') lub na wejściu I1 (Reset) stan wysoki ('1'), następuje zerowanie licznika odmierzanego czasu oraz wyjścia Timera.

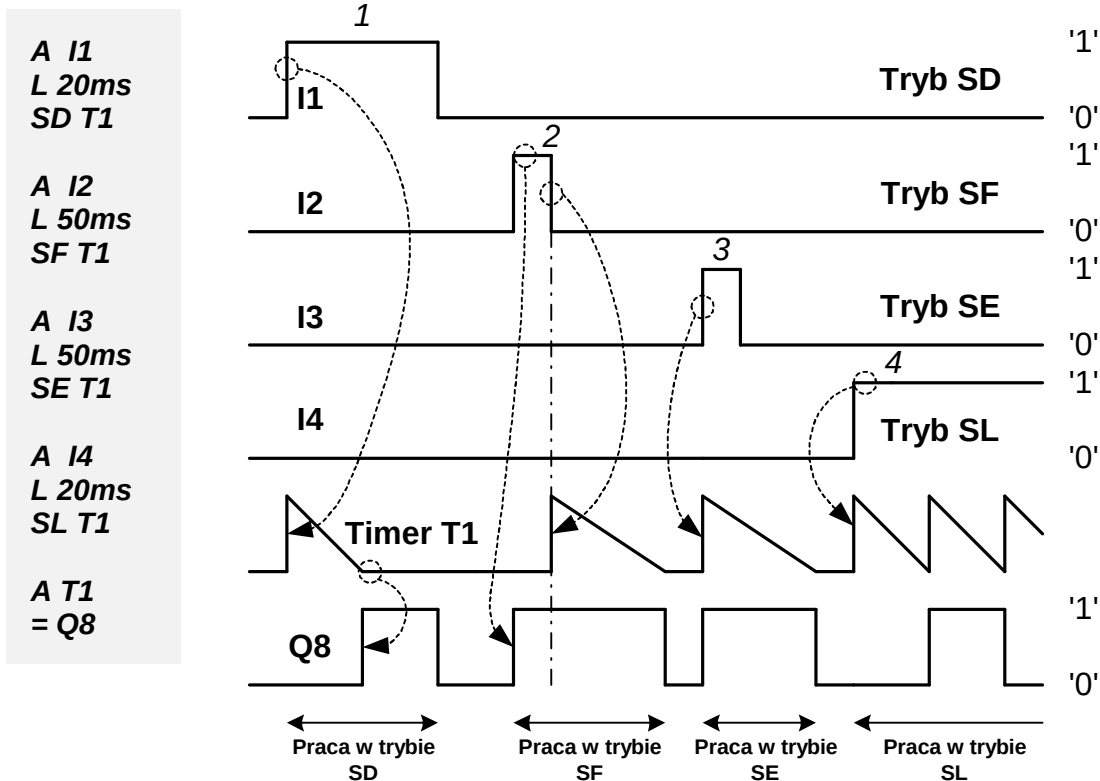
3.

Jeśli na wejścia Reset i Trigger podane są jednocześnie stany wysokie ('1') i po jakimś czasie poziom sygnału Reset zmieni się na niski ('0'), to następuje załączenie wyjścia Timera na $t=20\text{ms}$, potem wyłączenie wyjścia na 20ms, załączenie itd. Timer generuje na swoim wyjściu falę prostokątną, przesuniętą o 180° w stosunku do przebiegu z punktu 1.

Jeżeli instrukcja „L” nie zostanie użyta, to czas do odmierzenia przez T1 zostanie ustalony na podstawie pliku konfiguracyjnego „*.set” (okna Ustawień w programie PC Need).

5.1.2.17.5. Uwagi dotyczące wykorzystywania Timerów

Ten sam *Timer* można używać wiele razy, w różnych trybach.



Rys. 5.1.2.17.5. Przykładowe przebiegi czasowe sygnałów ilustrujące wielokrotne użycie Timera T1.

W powyższym przykładzie, jeśli na wejściu I1 pojawi się zbocze narastające, wówczas zostanie wyzwolony *Timer* T1 w trybie SD z czasem 20ms (1).

Jeśli na wejściu I2 pojawi się zbocze opadające, wówczas zostanie wyzwolony *Timer* T1 w trybie SF z czasem 50ms (2).

Jeśli na wejściu I3 pojawi się zbocze narastające, wówczas zostanie wyzwolony *Timer* T1 w trybie SE z czasem 50ms (3).

Jeśli na wejściu I4 pojawi się stan wysoki, wówczas zostanie wyzwolony *Timer* T1 w trybie SL z czasem 20ms (4). Na rys. 5.1.2.17.5. przedstawiono przykładowe przebiegi czasowe I1, I2, I3, I4 oraz Q8.

5.1.2.18. Instrukcje Liczników

5.1.2.18.1. Zliczanie w górę

SYMBOL – **CU**

SKŁADNIA:

CU <C>

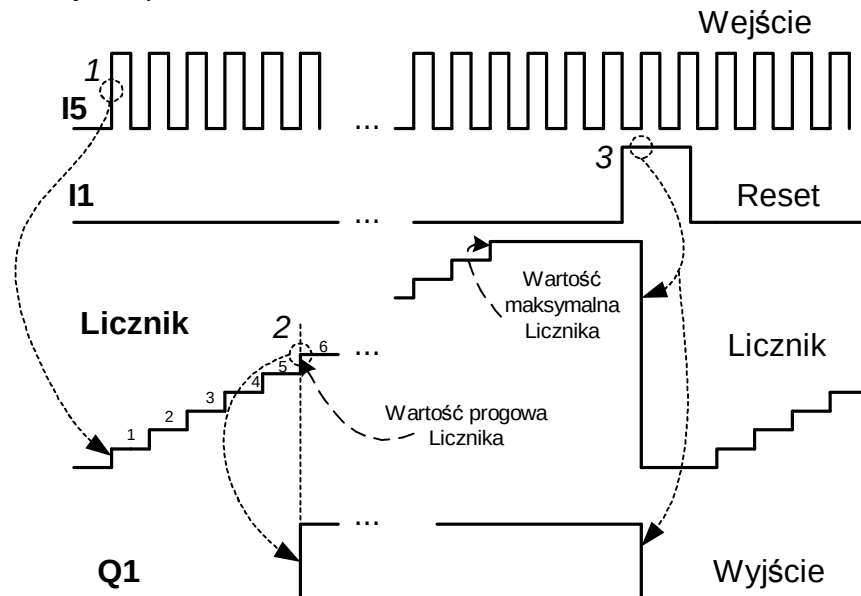
Czas wykonania instrukcji: 6,1µs.

Przykład:

A I5
L C#6
CU C1

A C1
=Q1

A I1
R C1



Rys. 5.1.2.18.1. Przykładowe przebiegi czasowe sygnałów ilustrujące działanie Licznika CU.

1.

Po wystąpieniu narastającego zbocza na wejściu wyzwalającym I5 nastąpi zwiększenie o 1 bieżącej wartości *Licznika* C1.

2.

Gdy wartość bieżąca *Licznika* osiągnie wartość progową (6), to wyjście Q1 zostanie ustawione w stan wysoki.

Jeśli na wejściu wyzwalającym nadal będą pojawiały się impulsy, *Licznik* będzie je zliczał, aż do osiągnięcia wartości maksymalnej 65535, pozostawiając swoje wyjście w stanie wysokim. *Licznik* nigdy się nie przepełnia – w momencie osiągnięcia maksymalnej wartości *Licznik* przestaje reagować na impulsy wyzwalające.

3.

Jeśli na wejściu resetującym I1 pojawi się stan wysoki – nastąpi skasowanie bieżącej wartości *Licznika* C1 i jego wyjścia. Po osiągnięciu stanu niskiego na tym wejściu możliwa jest dalsza praca *Licznika*.

Jeżeli instrukcja „L” nie zostanie użyta, to wartość progę, po osiągnięciu której *Licznik* C1 ustawia swoje wyjście w stan wysoki, zostanie ustalona na podstawie pliku konfiguracyjnego „*.set” (okna Ustawień w programie PC Need).

5.1.2.18.2. Zliczanie w dół

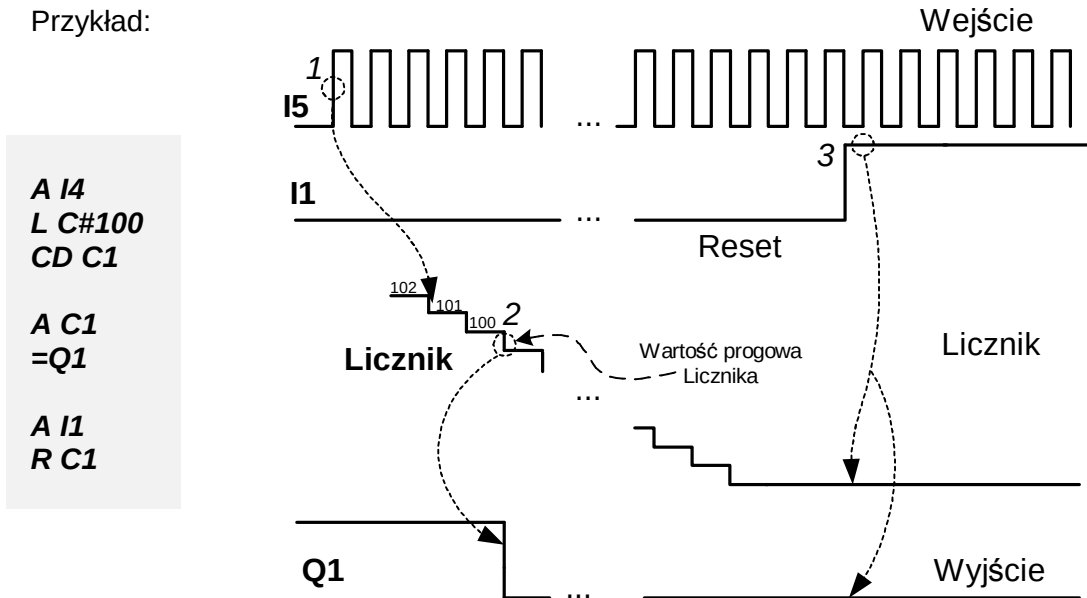
SYMBOL – **CD**

SKŁADNIA:

CD <C>

Czas wykonania instrukcji: 6,1µs.

Przykład:



Rys. 5.1.2.18.2. Przykładowe przebiegi czasowe sygnałów ilustrujące działanie Licznika CD.

1.

Po wystąpieniu narastającego zbocza na wejściu wyzwalającym I4 nastąpi zmniejszenie o 1 bieżącej wartości *Licznika* C1.

2.

Gdy wartość bieżąca *Licznika* impulsów spadnie poniżej wartości progowej (100), to wyjście Q1 zostanie ustawione w stan niski.

Jeśli na wejściu wyzwalającym I4 nadal będą pojawiały się impulsy, *Licznik* będzie je zliczał, aż do osiągnięcia wartości minimalnej 0.

Licznik nigdy się nie przepełnia – w momencie osiągnięcia minimalnej wartości, *Licznik* przestaje reagować na impulsy wyzwalające.

3.

Jeśli na wejściu I1 resetującym C1 pojawi się stan wysoki – nastąpi skasowanie bieżącej wartości *Licznika* C1 i jego wyjścia. Dalsza praca *Licznika* możliwa jest po osiągnięciu stanu niskiego na wejściu Reset.

Jeżeli instrukcja „L” nie zostanie użyta, to wartość progę, po osiągnięciu której *Licznik* C1 ustawia swoje wyjście w stan wysoki, zostanie ustalona na podstawie pliku konfiguracyjnego „*.set” (okna Ustawień w programie PC Need).



Maksymalna częstotliwość impulsów zliczających zależy od czasu wykonywanego programu. Stan wejścia liczącego musi być stabilny przynajmniej przez jeden cykl obiegu pętli programu.

5.1.2.18.3. Uwagi dotyczące wykorzystywania Liczników

1. Wykorzystanie *Szybkiego Licznika* HC

Aby korzystać z *Szybkiego Licznika* należy:

- podłączyć do wejścia I11 sygnał wyzwalający *Licznik*
- uaktywnić *Szybki Licznik* używając instrukcji CU lub CD, np.:

```
A I1
L C#25000
CU HC1
```

W powyższym przykładzie *Szybki Licznik* ustawi swoje wyjście w stan wysoki, jeżeli zliczanie zostanie uaktywnione stanem wysokim na wejściu I1 oraz wartość bieżąca *Licznika* będzie większa lub równa 25000.

```
A I1
L C#100
CD HC1
```

W powyższym przykładzie *Szybki Licznik* ustawi swoje wyjście w stan wysoki, jeżeli zliczanie zostanie uaktywnione stanem wysokim na wejściu I1 oraz wartość bieżąca *Licznika* będzie większa lub równa 100.

Jeżeli instrukcja „L” nie zostanie użyta, to wartość progu, po osiągnięciu którego *Szybki Licznik* ustawia swoje wyjście w stan wysoki, zostanie ustalony na podstawie pliku konfiguracyjnego „*.set” (okna Ustawień w programie PC Need).

Szybki Licznik zlicza w górę i w dół. Po osiągnięciu wartości maksymalnej 65535 *Szybki Licznik* przestaje zliczać. Ponowne liczenie od 0 następuje po wykonaniu instrukcji Reset.

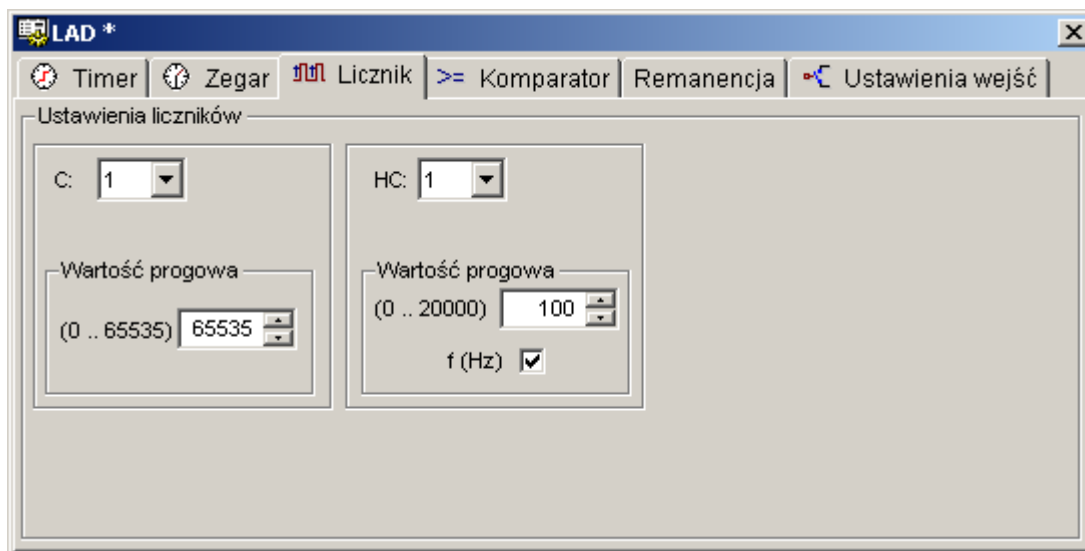
Szybki Licznik może mierzyć także częstotliwość – odpowiedni tryb pracy zostaje ustalony za pomocą konfiguracji w programie PCNeed.

Wejścia CU i CD dla *Szybkiego Licznika* nie są wejściami liczącymi, a jedynie ustalają kierunek zliczania.



Maksymalna gwarantowana częstotliwość pracy *Szybkiego Licznika* wynosi 20kHz.

Na rys. 5.1.2.18.3.1. przedstawiono przykładowe okno ustawień *Szybkiego Licznika* HC1



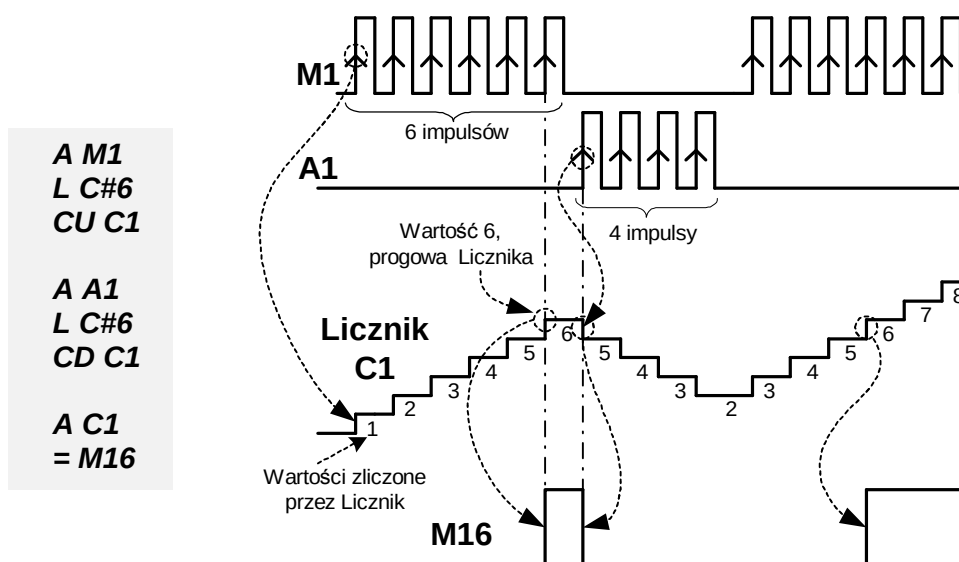
Rys. 5.1.2.18.3.1. Przykładowe okno konfiguracji Szybkiego Licznika HC1.

W podanym wyżej przykładzie *Szybki Licznik* ustawi swoje wyjście w stan wysoki, jeżeli ilość zliczanych impulsów w ciągu 1s będzie większa lub równa 100.

2. Jeden próg przełączający.

Aby ustawić jeden próg, przełączający wyjście *Licznika* w stan wysoki, należy używać tych samych argumentów (wartości do zliczania) w instrukcji *Load* dla CU i CD – rys. 5.1.2.18.3.2. Zbocza narastające, pojawiające się na M1 powodują zliczanie *Licznika* C1 w górę. Jeśli wartość zliczona przez C1 będzie większa lub równa 6 wówczas zostanie ustawione wyjście C1.

Zbocza narastające, pojawiające się na A1, powodują zliczanie *Licznika* C1 w dół. Jeśli wartość zliczona przez C1 będzie mniejsza od 6 wówczas wyjście C1 zostanie ustawione w stan niski.



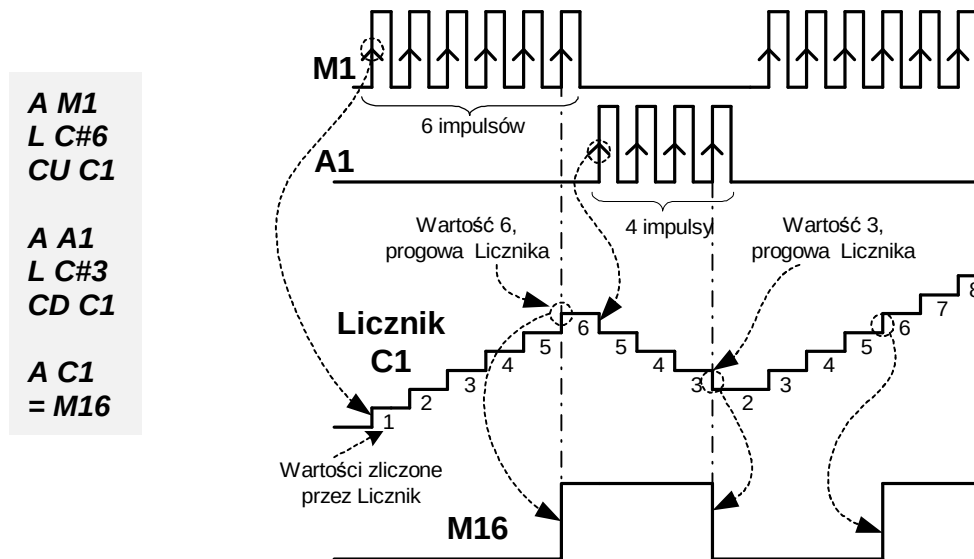
Rys. 5.1.2.18.3.2. Przykładowe przebiegi czasowe sygnałów ilustrujące działanie Licznika dla dwóch jednakowych progów przełączania.

3. Dwa progi przełączające (zakres).

Jeśli instrukcje *Load Liczników* używają różnych argumentów (wartości do zliczania), to zostają ustawione dwa progi przełączające – rys. 5.1.2.18.3.3.

Zbocza narastające, pojawiające się na M1 powodują zliczanie *Licznika* C1 w górę. Jeśli wartość zliczona przez C1 będzie większa lub równa 6 wówczas zostanie ustawione wyjście C1.

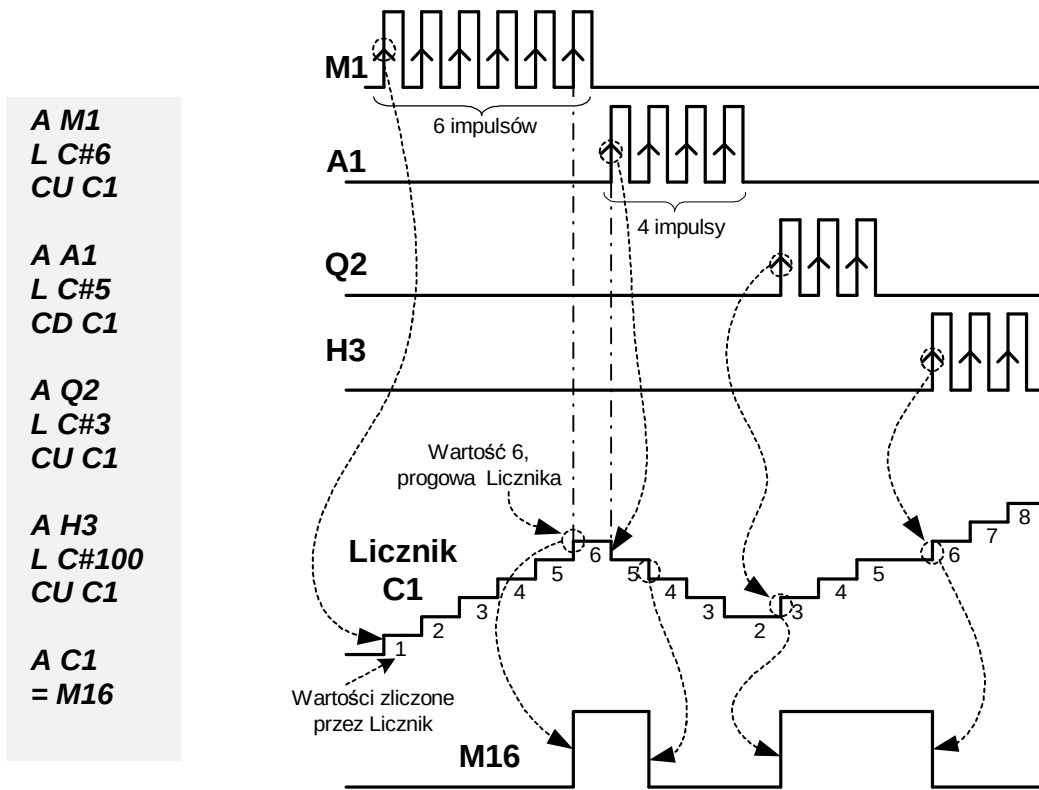
Zbocza narastające, pojawiające się na A1, powodują zliczanie *Licznika* C1 w dół. Dopiero, gdy wartość zliczona przez C1 będzie mniejsza od 3 – wyjście C1 zostanie ustawione w stan niski. Tak więc, przy zliczaniu w dół, wyjście C1 ustawione jest w stanie wysokim, gdy wartości zliczane przez *Licznik* będą pomiędzy 6 a 3.



Rys. 5.1.2.18.3.3. Przykładowe przebiegi czasowe sygnałów ilustrujące działanie Licznika dla dwóch różnych progów przełączania.

4. Kilka progów przełączających.

Można ustalić także kilka progów przełączających. Zawsze aktywne wejście „przejmuje kontrolę” nad *Licznikiem* i w zależności od aktualnej wartości zliczonej i ustawionego progu dla tego wejścia, następuje ustawienie bądź resetowanie wyjścia *Licznika* – rys. 5.1.2.18.3.4.



Rys. 5.1.2.18.3.4. Przykładowe przebiegi czasowe sygnałów ilustrujące działanie Licznika dla wielu różnych progów przełączania.

5.1.2.19. Instrukcje Zegara

Zegar jest zegarem czasu rzeczywistego i dokładna jego konfiguracja powinna być przeprowadzona przy użyciu programu PC Need, patrz rozdział 6. Dokładny opis działania Zegara przedstawiono w rozdziale 4.9. „Zegary”.

SYMBOL – **H**

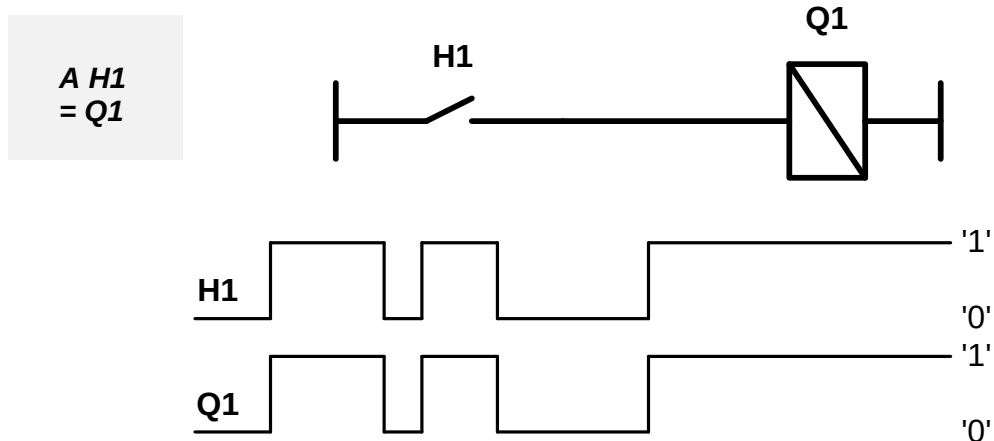
SKŁADNIA:

<instrukcje warunkowe> H <numer Zegara>

Przykład:

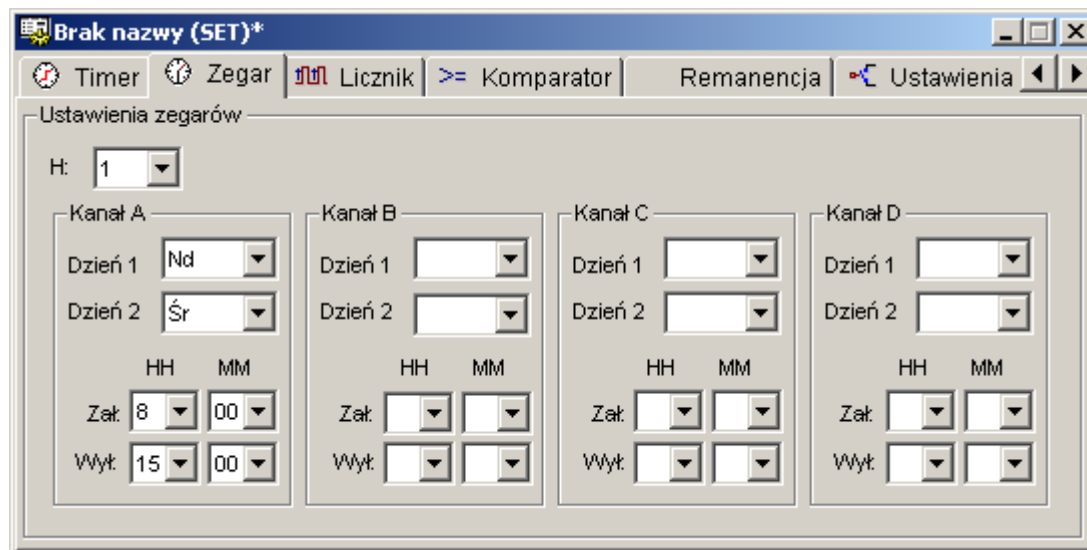
STL

Schemat przekaźnikowy



Rys. 5.1.2.19.1. Przykładowe przebiegi czasowe na styku H1 oraz na wyjściu Q1.

Konfigurujemy odpowiednio Zegar H1 za pomocą programu PC Need – patrz rozdział 6. Na rys. 5.1.2.19.2. przedstawiono przykładową konfigurację Zegara H1.



Rys. 5.1.2.19.2. Przykładowa konfiguracja Zegara H1.

Wyjście Q1 będzie ustawiane w takt zmian wyjścia zegara H1– od niedzieli do środy w godzinach od 8.00 do 15.00.

5.1.2.20. Wejścia analogowe

Dokładny opis działania wejść analogowych znajduje się w rozdziale 4.10. "Komparator – wejścia analogowe".

SYMBOL – **A**

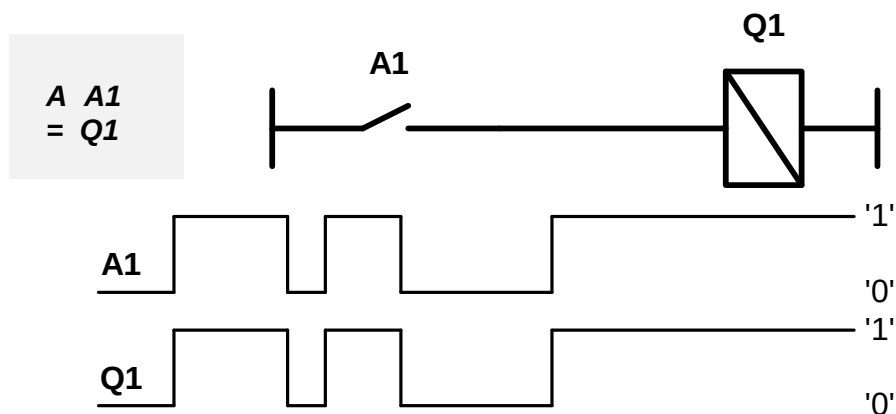
SKŁADNIA:

<instrukcje warunkowe> A <numer Komparatora>

Przykład:

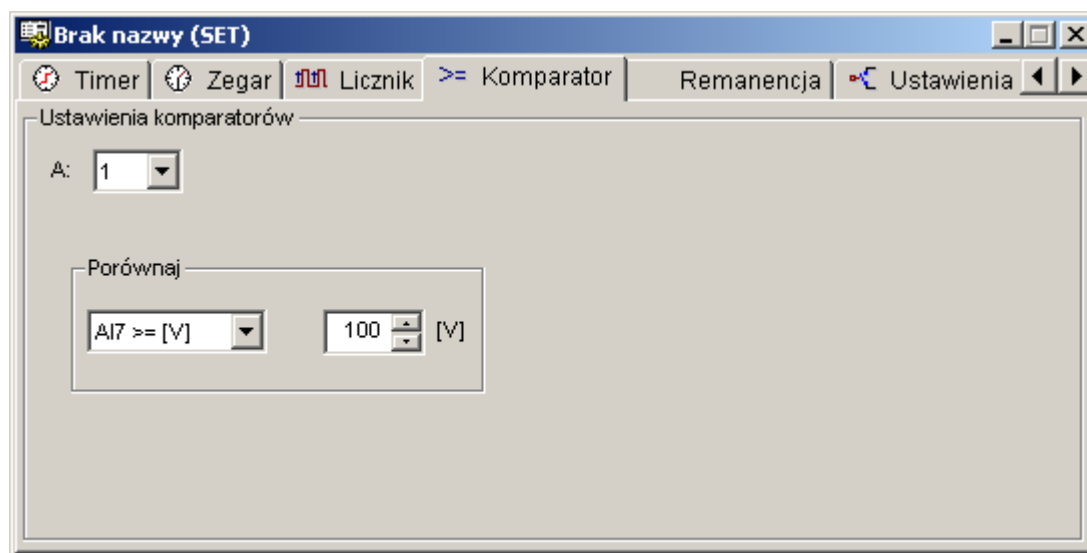
STL

Schemat przekaźnikowy



Rys.5.1.2.20.1. Przykładowe przebiegi czasowe na styku A1 oraz na wyjściu Q1.

Konfigurujemy odpowiednio wejścia analogowe za pomocą programu PC Need – patrz rozdział 6. Na rys. 5.1.2.20.2. przedstawiono przykładową konfigurację *Komparatora A1*.



Rys.5.1.2.20.2. Przykładowa konfiguracja komparatora A1.

Komparator porównuje wartość zadaną –100 z wartością analogową na wejściu I7. Jeśli wartość napięcia na wejściu I7 będzie większa lub równa od 100V, to *Komparator* przyjmie stan '1', w przeciwnym wypadku wyjście *Komparatora* będzie w stanie '0'. Wyjście Q1 podąża za zmianami na wyjściu *Komparatora A1*.

5.1.2.21. Instrukcja 'L'

Instrukcja 'L' służy do określenia odpowiednich czasów dla *Timerów* oraz wartości progowych (do zliczania) dla *Liczników*.

SYMBOL – **L**

SKŁADNIA:

L <wartość>

5.1.2.21.1. Instrukcja 'L' dla Timerów

5.1.2.21.1.1. Wartości stałe czasów dla Timerów

Czas wykonania instrukcji: 8,3μs

Parametr **<wartość>** dla instrukcji 'L' przyjmuje odpowiednie stałe wartości czasów z zakresów podanych w tab. 5.1.2.21.1. np.:

L 100ms SL T1	//Do Timera T1 pracującego w trybie SL zostanie załadowana wartość 100ms
L 10min SD T2	//Do Timera T2 pracującego w trybie SD zostanie załadowana wartość 10 min
L 1h.34min SF T8	//Do Timera T8 pracującego w trybie SF zostanie załadowana wartość //1h.34min

Tabela 5.1.2.21.1. przedstawia dostępne wartości czasów, które mogą być używane z instrukcją 'L'.

Tab .5.1.2.21.1. Argumenty instrukcji 'L' dla Timerów.

Format czasu	Zakres	Krok	Przykładowe wartości
s.ms (sekundy.milisekundy)	0s.10ms – 99s.990ms	10ms	0.50ms, 24s, 50s.120ms
min.s (minuty.sekundy)	0min.1s – 99min.59s	1s	2min, 32min, 98min.24s
h.min (godziny.minuty)	0h.1min – 99h.59min	1min	1h, 5h.18min

5.1.2.21.1.2. Wartości czasów dla Timerów określane na podstawie ustawienia Potencjometru

Czas wykonania instrukcji: 10,3μs.

Można również używać wartości czytanej z Potencjometru jako zadanego czasu do odmierzenia przez *Timery*, wtedy argument **<wartość>** instrukcji 'L' może przyjmować następujące wartości (patrz Tabela 5.1.2.21.2.):

1. x10ms

L Pot x10ms //Czas do odmierzenia = aktualna wartość Potencjometru (1 – 255)x10ms, //np. gdy wartość ustawiona Potencjometru = 25, to czas do odmierzenia = //25x10ms = 250ms.

2. x100ms

L Pot x100ms //Czas do odmierzenia = aktualna wartość Potencjometru (1 – 255) x 100ms, //np. wartość ustawiona Potencjometru = 15, to czas do odmierzenia = //15x100ms = 1500ms = 1,5s

3. x1s

L Pot x1s //Czas do odmierzenia = aktualna wartość Potencjometru (1 – 255) x 1s,
 //np. wartość ustawiona Potencjometru = 10, to czas do odmierzenia =
 //10x1s = 10s

4. x10s

L Pot x10s //Czas do odmierzenia = aktualna wartość Potencjometru (1 – 255) x 10s,
 //np. wartość ustawiona Potencjometru = 8, to czas do odmierzenia =
 //8x10s = 80s

5. x1min

L Pot x1min //Czas do odmierzenia = aktualna wartość Potencjometru (1 – 255) x 1min,
 //np. wartość ustawiona Potencjometru = 255, to czas do odmierzenia =
 //255x1min = 255min

Tab.5.1.2.21.1.2. Argumenty instrukcji 'L' dla Timerów uzyskiwane poprzez nastawy Potencjometru.

Zakres Potencjometru	Mnożnik	Zakres czasu
1 – 255	x 10ms	10ms – 2,55s
	x 100ms	100ms – 25,50s
	x 1s	1s – 4min15s
	x 10s	10s – 42min30s
	x 1min	1min – 255min0s

5.1.2.21.1.3. Wartości czasów dla Timerów określone na podstawie wartości napięć na wejściach analogowych napięciowych

Czas wykonania instrukcji: 10,3μs.



Istnieje możliwość używania, do odmierzenia czasów dla Timerów, wartości napięć czytanych z wejść analogowych I7, I8 dla NEED-12DC-...-08-4, NEED-24DC-...-08-4, lub I14, I15, I16 dla NEED-12DC-...-16-8, NEED-24DC-...-16-8.

Dla wejść analogowych napięciowych argument **<wartość>** instrukcji 'L' może przyjmować wartości czasów przedstawionych w tabeli 5.1.2.21.1.3.

Tab.5.1.2.21.1.3. Argumenty instrukcji 'L' dla Timerów uzyskiwane poprzez odczyt wejść analogowych napięciowych.

Zakres napięcia mierzonego na wejściu analogowym [V]	Mnożnik zakresu	Mnożnik ogólny	Zakres czasu
0,10 – 25,50 (krok 0,10)	x 10ms	x 10	10ms – 2s550ms
	x 100ms		100ms – 25s500ms
	x 1s		1s – 4min15s
	x 10s		10s – 42min30s
	x 1min		1min – 255min0s
0,05 – 12,75 (krok 0,05)	x 10ms	x 20	10ms – 2s550ms
	x 100ms		100ms – 25s500s
	x 1s		1s – 4min15s
	x 10s		10s – 42min30s
	x 1min		1min – 255min0s

Odmierzany czas dla przekaźników NEED-24DC-..., NEED-12DC-.. obliczamy:

Wartość napięcia na wejściu analogowym [V] x Mnożnik zakresu x Mnożnik ogólny = Czas odmierzan

W składni języka STL używamy symboli **AI7** lub **AI8** dla NEED-12DC-...-08-4, NEED-24DC-...-08-4 lub **AI14**, **AI15**, **AI16** dla NEED-12DC-...-16-8, NEED-24DC-...-16-8 np.:

L AI7 x1min //Czas do odmierzan

//AI7[V] x 1min x 10, (zakres 0,1V - 25,50V)

//np. wartość napięcia na wejściu analogowym AI7 = 20V,

//to czas do odmierzan

L AI14 x100ms //Czas do odmierzan

//analogowym AI14[V] (zakres 0,05V - 12,75V) x 100ms x 20,

//np. wartość napięcia analogowego AI14 = 10V,

//to czas do odmierzan



Zwiększoną rozdzielczość wejść analogowych (zakres pracy 0,05V – 12,75V) możemy stosować tylko dla przekaźników NEED-12DC-...-16-8 lub NEED-24DC-...-16-8, NEED-12DC-22-.. oraz NEED-24DC-22..

5.1.2.21.1.4. Wartości czasów dla Timerów określane na podstawie wartości prądów na wejściach analogowych prądowych

Czas wykonania instrukcji: 10,3μs.

Dla wejść analogowych prądowych (tylko dla NEED-12DC-...-16-8, NEED-24DC-...-16-8) argument **<wartość>** instrukcji 'L' może przyjmować wartości czasów przedstawionych w tabeli 5.1.2.21.1.4.

Tab.5.1.2.21.1.4. Argumenty instrukcji 'L' dla Timerów uzyskiwane poprzez odczyt wejść analogowych prądowych.

Zakres prądu mierzonego na wejściu analogowym [mA]	Mnożnik zakresu	Mnożnik ogólny	Zakres czasu
0,2 – 51,0 (krok 0,2)	x 10ms	x 5	10ms – 2s550ms
	x 100ms		100ms – 25s500ms
	x 1s		1s – 4min15s
	x 10s		10s – 42min30s
	x 1min		1min – 255min0s
0,1 – 25,50 (krok 0,1)	x 10ms	x 10	10ms – 2s550ms
	x 100ms		100ms – 25s500s
	x 1s		1s – 4min15s
	x 10s		10s – 42min30s
	x 1min		1min – 255min0s

Odmierzany czas dla przekaźników NEED-24DC-...-16-8, NEED-12DC-...-16-8 obliczamy:

Wartość prądu na wejściu analogowym [mA] x Mnożnik zakresu x Mnożnik ogólny = Czas odmierzanym

L AI16 x1min //Czas do odmierzanym = aktualna wartość prądu w [mA] (zakres 0,2 – 51) x
//1min x 5
//np. wartość prądu AI16 = 10mA
//to czas do odmierzanym = 10mA x 1min x 5 = 10minx5=50min

L AI14 x1s //Czas do odmierzanym = aktualna wartość prądu w [mA] (zakres 0,1 – 25,5) x
//1s x 10
//np. wartość prądu AI16 = 5mA
//to czas do odmierzanym = 5mA x 1s x 10 = 5s x 10 = 50s



Zwiększoną rozdzielczość wejść analogowych (zakres pracy 0,1– 25,50mA) możemy stosować tylko dla przekaźników NEED-12DC-...-16-8 lub NEED-24DC-...-16-8.

5.1.2.21.2. Instrukcja 'L' dla Liczników

5.1.2.21.2.1. Wartości stałe, progowe dla Liczników

Czas wykonania instrukcji: 8,3μs

Parametr **<wartość>** instrukcji 'L' przyjmuje odpowiednie stałe liczby, poprzedzone znakami **C#**, do zliczenia dla Liczników z zakresu 0–65535 np.:

L C#10

CU C1 //Ustawienie wartości 10 do zliczania przez Licznik C1 liczący w górę

L C#1000

CD C8 //Ustawienie wartości 1000 do zliczania przez Licznik C8 liczący w dół

5.1.2.21.2.2. Wartości progowe dla Liczników określane na podstawie ustawienia Potencjometru

Czas wykonania instrukcji: 10,3µs.

Można również używać wartości czytanej z Potencjometru jako zadanej wartości do zliczania przez *Liczniki*, wtedy format instrukcji 'L' może przyjmować następującą postać:

1.

L Pot x1 //Wartość do zliczania z zakresu (1 – 255)x1 (np. Potencjometr ma
CU C1 //ustawioną wartość 23 – to wartość do zliczania przez C1 będzie
 //równa 23x1=23)

2.

L Pot x10 //Wartość do zliczania z zakresu (1 – 255)x10 (np. Potencjometr
CD C2 //ma ustawioną wartość 23 – to wartość do zliczania przez C2 będzie
 //równa 23x10=230)

3.

L Pot x100 //Wartość do zliczania z zakresu (1 – 255)x100 (np. Potencjometr
 //ma ustawioną wartość 23 – to wartość do zliczania przez C3 będzie
 //równa 23x100=2300)

Tab.5.1.2.21.2.1. Argumenty instrukcji 'L' dla Liczników uzyskiwane poprzez nastawy potencjometru.

Zakres Potencjometru	Mnożnik zakresu	Zakres liczb
1 – 255	x 1	1 – 255
	x 10	10 – 2550
	x 100	100 – 25500

5.1.2.21.2.3. Wartości progowe dla *Liczników* określane na podstawie wartości napięć na wejściach analogowych napięciowych

Czas wykonania instrukcji: 10,3µs.



Istnieje możliwość używania, do określania progów dla *Licznika*, wartości napięć czytanych z wejść analogowych I7, I8 dla NEED-12DC-...-08-4, NEED-24DC-...-08-4 lub I14, I15, I16 dla NEED-12DC-...-16-8, NEED-24DC-...-16-8. W takim przypadku argument **<wartość>** instrukcji 'L' może przyjmować wartości progów przedstawionych w tabeli 5.1.2.21.2.3.

Tab.5.1.2.21.2.3. Argumenty instrukcji 'L' dla Liczników uzyskiwane poprzez odczyt wejść analogowych.

Zakres napięcia na wejściu analogowym [V]	Mnożnik zakresu	Mnożnik ogólny	Zakres liczb
0,1 – 25,5 (krok 0,1)	x 1	x 10	1– 255
	x 10		10 – 2550
	x 100		100 – 25500
0,05 – 12,75 (krok 0,05)	x 1	x 20	1– 255
	x 10		10 – 2550
	x 100		100 – 25500

Ustawiony próg dla przekaźników NEED-24DC-..., NEED-12DC-.. obliczamy:

*Wartość napięcia na wejściu analogowym [V] x Mnożnik zakresu x Mnożnik ogólny =
próg dla Licznika*

W składni języka STL używamy symboli **AI7** lub **AI8** dla NEED-12DC-...-08-4, NEED-24DC-...-08-4 lub **AI14**, **AI15**, **AI16** dla NEED-12DC-...-16-8, NEED-24DC-...-16-8 np.:

L AI7 x100 //Wartość do zliczania = aktualna wartość napięcia na wejściu analogowym
//(zakres 0,1V – 25,5V) AI7[V] x 100 x 10,
//np. wartość napięcia na AI7=10V
//to wartość do zliczenia = 10V x 100 x 10 = 10V x 1000 = 10 000

L AI15 x10 // Wartość do zliczania = aktualna wartość napięcia na wejściu analogowym
// (zakres 0,05V – 12,75V) AI15[V] x 10 x 20,
//np. wartość napięcia na AI15 = 1V
//to wartość do zliczania = 1V x 10 x 20 = 1V x 200 = 200



Zwiększoną rozdzielczość wejść analogowych (zakres pracy 0,05V – 12,75V) możemy stosować tylko dla przekaźników NEED-12DC-...-16-8 lub NEED-24DC-...-16-8, NEED-12DC-22-..., NEED-24DC-22-...

5.1.2.21.2.4. Wartości progowe dla *Liczników* określone na podstawie wartości prądów na wejściach analogowych prądowych

Czas wykonania instrukcji: 10,3μs.



Istnieje możliwość używania, do określania progów dla *Licznika*, wartości prądów mierzonych na wejściach analogowych I14, I15, I16 dla NEED-12DC-...-16-8, NEED-24DC-...-16-8. W takim przypadku argument **<wartość>** instrukcji 'L' może przyjmować wartości progów przedstawionych w tabeli 5.1.2.21.2.4.

Tab.5.1.2.21.2.4. Argumenty instrukcji 'L' dla Liczników uzyskiwane poprzez odczyt wejść analogowych.

Zakres prądu na wejściu analogowym [mA]	Mnożnik zakresu	Mnożnik ogólny	Zakres liczb
0,2 – 51,0 (krok 0,2)	x 1	x 5	10– 255
	x 10		100 – 2550
	x 100		1000 – 25500
0,1 – 25,5 (krok 0,1)	x 1	x 10	10– 255
	x 10		100 – 2550
	x 100		1000 – 25500

Ustawiony próg dla przekaźników NEED-24DC-x1-..., NEED-12DC-x1-.. obliczamy:

Wartość prądu na wejściu analogowym [mA] x Mnożnik zakresu x Mnożnik ogólny =
próg dla Licznika

L AI16 x100 //Wartość do zliczania = aktualna wartość prądu na wejściu analogowym
//(zakres 0,2mA – 51mA) AI16 [mA] x 100 x 5,
//np. wartość prądu AI16 = 1mA
//to wartość do zliczenia = 1mA x 100 x 5 = 1mA x 500 = 500

L AI15 x10 // Wartość do zliczania = aktualna wartość prądu na wejściu analogowym
// (zakres 0,1mA – 25,5mA) AI15[mA] x 10 x 10,
//np. wartość prądu AI15 = 10mA
//to wartość do zliczenia = 10mA x 10 x 10 = 10mA x 100 = 1000



Zwiększoną rozdzielczość wejść analogowych (zakres pracy 0,1– 25,50mA) możemy stosować tylko dla przekaźników NEED-12DC-..-16-8 lub NEED-24DC-..-16-8.

Przykład zastosowania instrukcji 'L':

A I5
L 20s
SF T1

A I5
L C#10
CU C8

A I8
L Pot x1s
SE T2

A I5
L AI16 x10
CU C1

Do Timera T1 zostaje załadowana wartość 20s.

Dla Licznika C8 zostaje ustalona stałą wartość progowa 10 przełączająca jego stan wyjściowy z niskiego ('0') na wysoki ('1').

Do *Timera* T2 zostaje załadowana wartość z Potencjometru pomnożona przez 1s
 Dla *Licznika* C1 zostaje ustalona wartość progowa określona za pomocą wartości napięcia analogowego występującego na AI16, pomnożona przez 10 (Mnożnik zakres 0,1 – 25,5V) x 10 (mnożnik ogólny), przełączająca jego stan wyjściowy z niskiego ('0') na wysoki ('1').

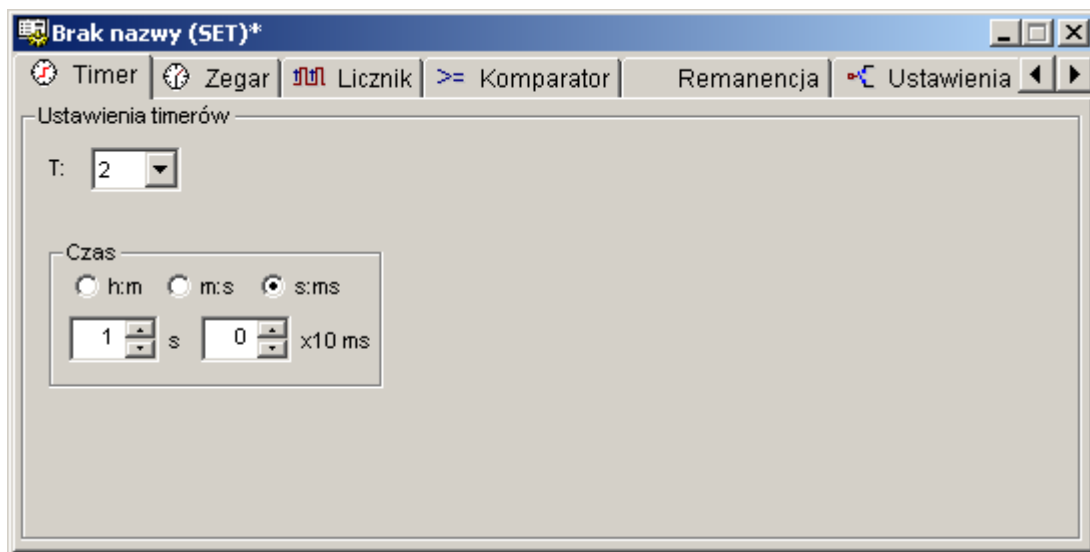
5.1.2.21.3. Uwagi dotyczące używania instrukcji 'L'

1. Jeśli w programie nie została wykonana żadna instrukcja *Load*, to wartości czasów odmierzanych przez *Timery* oraz wartości progowe dla *Liczników* określone są w programie PC Need, w pliku konfiguracyjnym „*.set” np.:

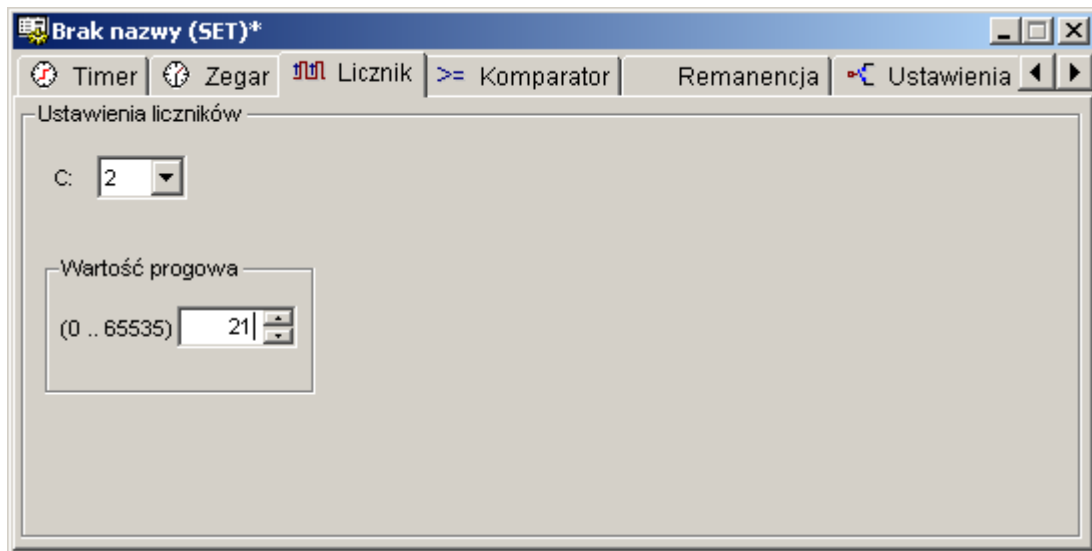
**A I3
SE T2**

**A I5
CD C2**

W powyższym przykładzie *Timer* T2 będzie odmierzał czas 1s, ustawiony w programie PC Need, natomiast *Licznik* będzie ustawiał/kasował swoje wyjście dla progu 21. Poniższe konfiguracje przedstawione są na rys. 5.1.2.21.3.1. oraz rys. 5.1.2.21.3.2.



Rys.5.1.2.21.3.1. Ustawienie czasu dla Timera T2.



Rys.5.1.2.21.3.2. Ustawienie czasu dla Licznika C2.

2. Jeśli w programie została wykonana instrukcja *Load*, to wszystkie wartości czasów do odmierzenia dla *Timerów* i wartości do zliczenia dla *Liczników*, występujących bezpośrednio po *Load*, określa ta instrukcja.

A I3
SE T2

A I8
L 1min
SE T3

A I8
SE T4

W powyższym przykładzie wyzwolenie *Timera* T2 narastającym zboczem na wejściu I3 spowoduje, iż T2 będzie odmierzał czas określony w programie PC Need, w pliku konfiguracyjnym.

Jeśli na wejściu I8 pojawi się zbocze narastające, to *Timer* T3 będzie odmierzał czas określony w instrukcji *Load* – 1min, a *Timer* T4 będzie odmierzał czas określony w pliku ustawień „*.set”.

5.1.2.22. Instrukcja zawsze ustawiająca SET

Instrukcja ‘SET’ ustawia na stałe stan wysoki ‘1’.

SYMBOL – **SET**

SKŁADNIA:

SET

Czas wykonania instrukcji: 8,9μs.

Instrukcja ‘SET’ jest instrukcją bezwarunkową (wykonywaną zawsze), ustawiającą na stałe stan logiczny ‘1’ w części warunkowej obwodu.

Przykład:

```
SET  
= Q4  
SL T1  
S M16
```

Wyjście Q4, Znacznik M16 po wykonaniu tej instrukcji, na stałe będą ustawione w stan wysoki '1'. Natomiast *Timer* T1 zostanie na stałe wyzwolony i będzie pracował w trybie generatora impulsów.

5.1.2.23. Instrukcja zawsze kasująca CLR

Instrukcja 'CLR' ustawia na stałe stan niski '0'.

SYMBOL – **CLR**

SKŁADNIA:

CLR

Czas wykonania instrukcji: 8,9µs.

Instrukcja 'CLR' jest instrukcją bezwarunkową (wykonywaną zawsze), ustawiającą na stałe stan logiczny '0' w części warunkowej obwodu.

Przykład:

```
CLR  
= Q4  
= M1  
SL T1
```

Znacznik M1 i wyjście Q1, po wykonaniu instrukcji 'CLR', na stałe będą ustawione w stan niski '0' natomiast *Timer* T1 nigdy nie wystartuje.

5.2. Programowanie w języku graficznym LAD

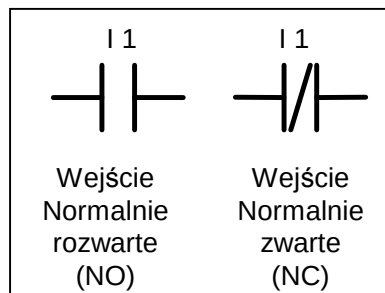
LAD (ang. *Ladder Diagram*) to prosty sposób programowania używany do edycji programów w sterownikach PLC. Dzięki zachowaniu podstawowych kanonów standardu języka, nie powinien on sprawić żadnych problemów użytkownikom, którzy mieli już do czynienia z podobnym sposobem tworzenia programów. Użytkownicy przełączników NEED, którzy spotykają się z nim po raz pierwszy, będą mieli możliwość zapoznania się i zastosowania tego sposobu programowania, który nawiązuje do „rysowania” schematów połączeń elektrycznych.

5.2.1. Symbole w LAD

Język schematów drabinkowych LAD – bazuje na symbolach logiki stykowo-przełącznikowej. W ten sposób mogą być reprezentowane styki (elementy wejściowe), wyjścia dwustanowe (odzwierciedlenie cewek przełącznika) oraz wyjścia funkcyjne.

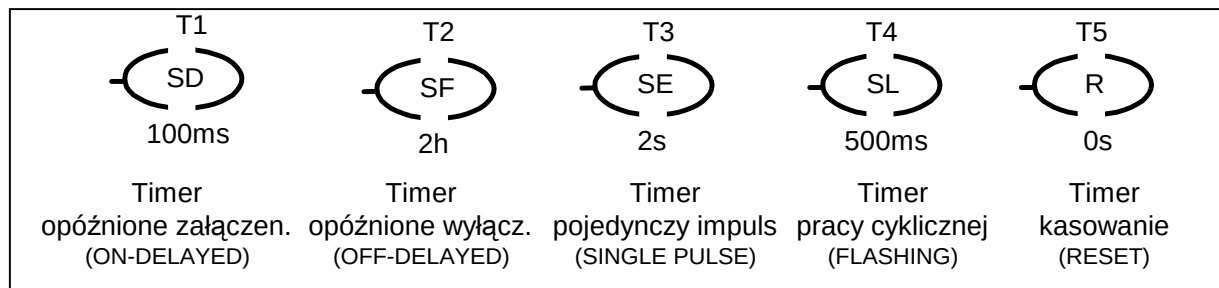
Podstawowe symbole języka LAD dla odzwierciedlenia wejść są przedstawione na rys.

5.2.1.1.

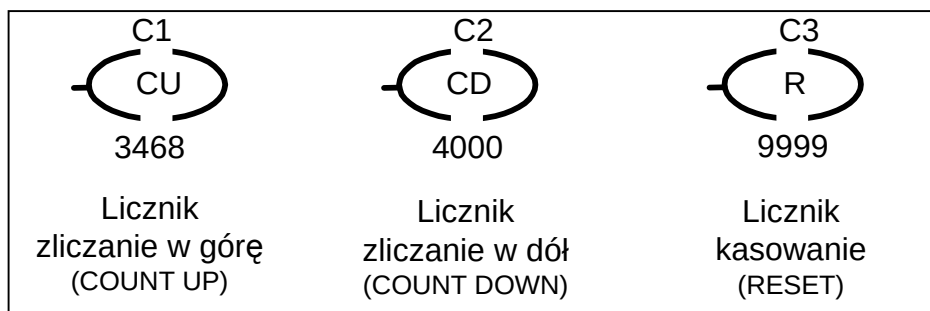


Rys. 5.2.1.1. Podstawowe elementy języka LAD – wejścia (styki)

Wyjścia funkcyjne to *Timery* rys. 5.2.1.2. oraz *Liczniki* rys. 5.2.1.3.

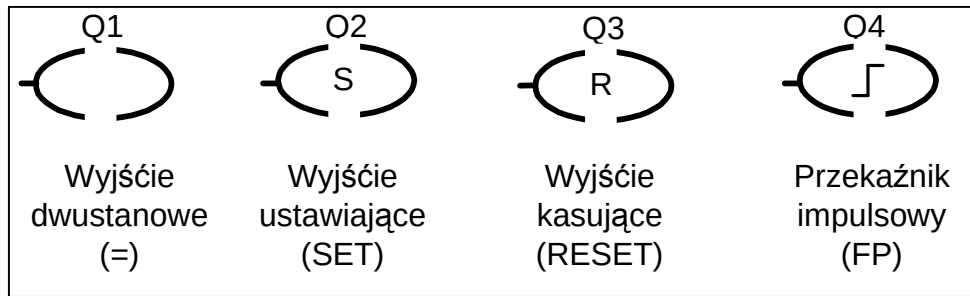


Rys. 5.2.1.2. Elementy języka LAD – Timery.



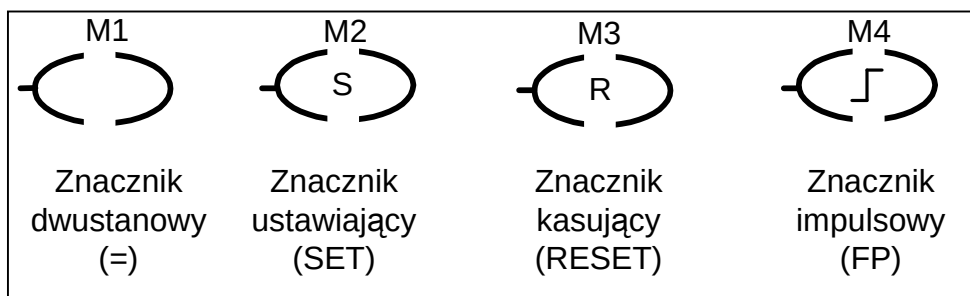
Rys. 5.2.1.3. Elementy języka LAD – Liczniki.

Symbole języka LAD dla odzwierciedlenia wyjść są przedstawione na rys. 5.2.1.4.



Rys.5.2.1.4. Elementy języka LAD – wyjścia.

Symbole języka LAD dla odzwierciedlenia Znaczników są przedstawione na rys. 5.2.1.5.



Rys.5.2.1.5. Elementy języka LAD – Znaczniki

5.2.2. Wejścia

Z punktu widzenia programu LAD wejściem może być nie tylko fizyczny styk elementu elektrycznego (wejście dyskretne), ale także stan (poziom logiczny) *Timera*, *Licznika*, *Zegara*, *Znacznika* lub nawet wyjścia. Ponieważ elementom tym w trakcie ich działania przypisywane są wartości dwustanowe ('0' lub '1') możemy je sprawdzać i uzależniać od nich działanie innych elementów obwodu.



Uwaga: Sprawdzanie wyjść polega jedynie na uzyskaniu informacji programowej o stanie rejestru, który steruje danym wyjściem fizycznym, tzn. nie uwzględnia się sprawności przełącznika i układu wykonawczego danego wyjścia.

5.2.3. Wyjścia

W najprostszym przypadku mamy do czynienia z dwustanowym elementem, takim jak przełącznik, którego cewka jest zasilana bądź nie. Możemy wtedy mówić, że przełącznik działa (jest aktywny), jeśli jego cewka jest zasilana, – czyli przyjmujemy dla niej określony stan logiczny. W naszym wypadku będziemy stosować logikę dodatnią, czyli będzie to stan '1' dla wyjścia załączonego, a dla niedziałającego wyjścia stan logiczny '0'.

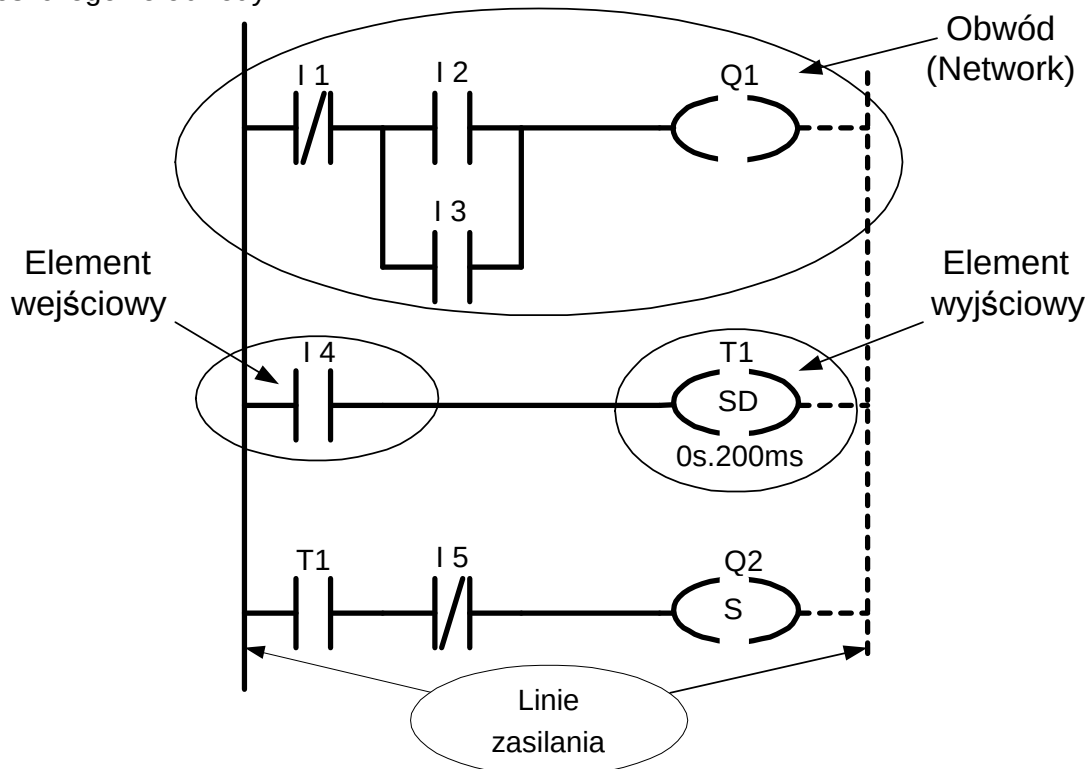
W zależności od przypisanej funkcji (patrz tab. 5.2.6.) wyjście może być ustawiane w ciągłej zależności od wejść (instrukcja „=”) – analogia do przełącznika działającego, jeśli cewka jest zasilana. Inaczej zachowują się wyjścia typu SET i RESET, gdzie po spełnieniu warunków następuje ustawienie na stałe (instrukcja „S”) stanu logicznego '1'. Stan taki pozostaje aż do momentu wykonania operacji kasującej (R) – odpowiada to zachowaniu się przełącznika z podtrzymaniem.

Wyjścia w LAD również mogą nie posiadać swoich fizycznych odpowiedników w strukturze przełącznikowej, są to tzw. wyjścia funkcyjne, które umożliwiają użycie takich elementów jak *Timer*, *Licznik*, *Zegar*, *Znacznik*. Elementy te są ustawione podobnie jak fizyczne wyjścia (przyjmują stan '0' lub '1'), w zależności od przypisanej im funkcji (patrz tab. 5.2.6.).

5.2.4. Struktura programu w LAD

Symbole umieszczane są w obwodach (ang. *Network*). Obwody umieszczone są w sposób podobny do szczebli (ang. *Rungs*) w drabinie. Kolejne obwody (szczeble drabiny) odczytywane są kolejno od góry do dołu. Po dojściu do ostatniego szczebla, proces śledzenia programu rozpoczyna się od początku.

Obwód ograniczony jest z lewej i prawej strony przez szyny prądowe. Prawa szyna może być widoczna, ale także może być pominięta na rysunku. Ze względu na analogię z schematem przekaźnikowym programy w LAD mogą być czytane jako przepływ prądu od lewej do prawej linii pionowej (np. lewa strona zasilanie, prawa potencjał masy) poprzez poszczególne obwody.



Rys. 5.2.3. Przykładowa aplikacja zrealizowana w języku LAD.

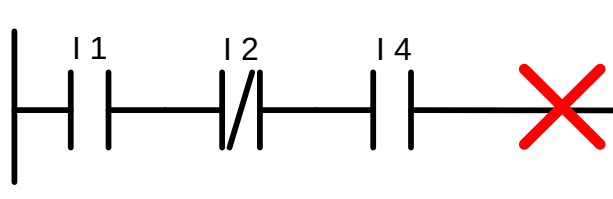
5.2.5. Struktura obwodu LAD

Obwód musi posiadać odpowiedni format i składnię. Oto kilka najważniejszych zasad:

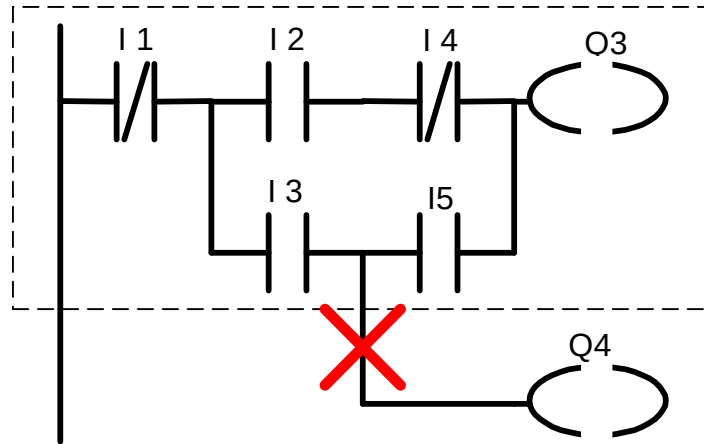
- każdy obwód może posiadać do 16 linii równoległych,
- ostatnim elementem szeregowego połączenia w danym obwodzie musi być jeden z elementów wykonawczych (wyjście dwustanowe lub funkcyjne),
- obwód może posiadać maksymalnie do 16 elementów wyjściowych,
- nie może wystąpić rozgałęzienie mające początek lub koniec wewnątrz innego odgałęzienia, które łączy się z „linią zasilającą” lub wyjściami.
- obwód musi mieć połączenie z liniami zasilania.

Poniżej przedstawiono przykłady niedozwolonych połączeń:

- Brak elementu wyjściowego



- Rozgałęzienie wewnątrz innego obwodu, którego jeden koniec (lub początek) łączy się z „linią zasilającą” lub wyjściami (w poniższym przykładzie wyjście Q4 nie może być podłączone do rozgałęzienia I3 i I5).



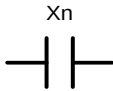
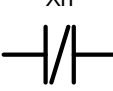
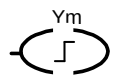
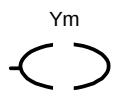
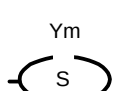
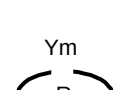
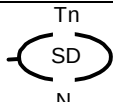
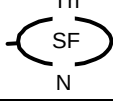
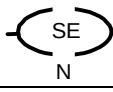
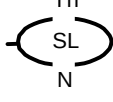
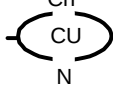
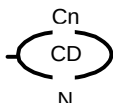
5.2.6. Opis używanych elementów

Elementowi logicznemu (symbolowi – patrz tab. 5.2.6.), który pełni w języku LAD funkcję wejścia bądź wyjścia sygnału, mogą być przyporządkowane różne zmienne tzn. wejściem sygnału może być nie tylko napięcie podawane na sprzętowe wejścia (ozn. I1..I8), ale także wejściem może być stan *Timera*, *Licznika*, *Zegara* oraz stan *Wyjścia*. Decyduje o tym opis na symbolu elementu. Cyfra w oznaczeniu wskazuje numer wejścia, które ma być sprawdzane. Analogicznie możemy ustawiać (bądź kasować) nie tylko fizyczne wyjścia, ale także *Znaczni* (wyjścia bez fizycznego wyprowadzenia) oraz stan *Timerów*, *Liczników* itp. W tabeli 5.2.6 przedstawiono symbole języka LAD z opisem i dopuszczalnymi zmiennymi XY sygnału dla danego elementu (X – wejścia, Y – wyjścia).

Wejście aktywne – wejście, którego stan pozwala na przepływ sygnału. (Logiczne '1' dla wejścia NO, logiczne '0' dla wejścia NC).

Wyjście aktywne – wyjście, które posiada sygnał logiczny '1'.

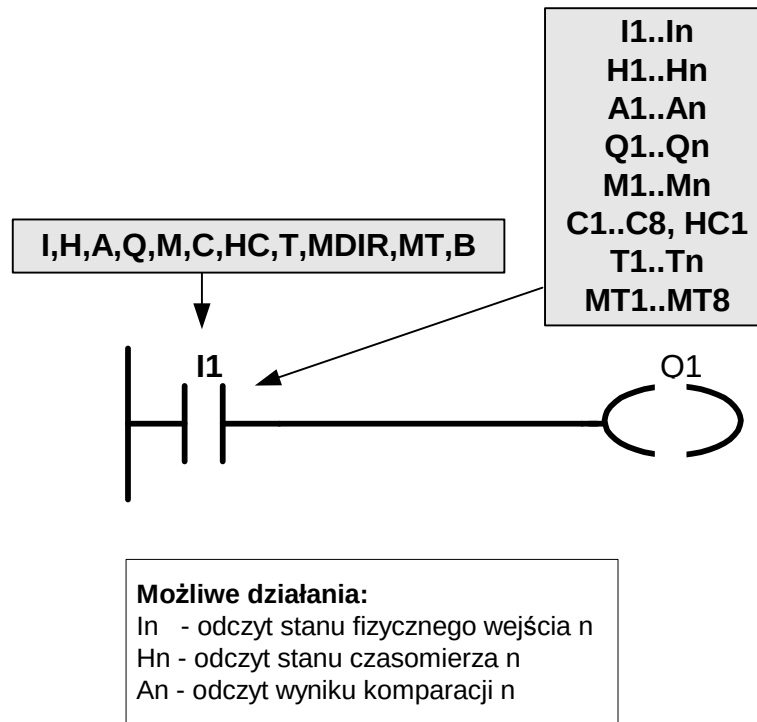
Tab. 5.2.6. Podstawowe symbole języka LAD.

LAD	Opis	Zmienna
 X_n	Wejście normalnie otwarte – NO. Wejście aktywne (styk zwarty), gdy wartość logiczna przypisanej zmiennej wynosi '1'. (Normally Open)	X: I, A, H, Q, M, T, C, HC, MDIR, MT, B n: numer wejścia danego rodzaju
 X_n	Wejście normalnie zamknięte – NC. Wejście aktywne (styk rozwarty), gdy wartość logiczna przypisanej zmiennej wynosi '0'. (Normally Closed)	
 Y_m	Przełącznik impulsowy – pełni rolę przerzutnika wyzwalanego zboczem narastającym. Każdy narastający impuls zmienia stan wyjścia na przeciwny. (FP)	Y: Q, M, MT m: numer wyjścia danego rodzaju
 Y_m	Wyjście Przyporządkowujące Ustawia wartość przypisanej zmiennej na '1', gdy podany zostanie do niego sygnał. Odpowiednik przełącznika o stykach otwartych (przepisanie stanu wejściowego na wyjście) (Assign)	Y: Q, M, MT m: numer wyjścia danego rodzaju
 Y_m	Wyjście Set Ustawia wartość przypisanej zmiennej na '1', gdy podany zostanie do niego sygnał i utrzymuje ten stan aż do momentu wykonania instrukcji 'Reset' lub do wyłączenia zasilania przełącznika programowalnego (przełącznik z podtrzymaniem)	Y: Q, M, MT m: numer wyjścia danego rodzaju
 Y_m	Wyjście Reset Ustawia wartość przypisanej zmiennej na '0', gdy podany zostanie do niego sygnał i utrzymuje ten stan aż do momentu wykonania instrukcji 'Set (S-STL)' lub do wyłączenia zasilania przełącznika programowalnego (kasowanie wyjścia)	
 T_n SD N	Timer Opóźnione załączenie Ustawia wartość $T_n = '1'$ po upływie zadanego czasu „N” od momentu uaktywnienia.	T n: nr Timera
 T_n SF N	Timer Opóźnione wyłączenie Utrzymuje wartość $T_n = '1'$ przez zadany czas „N” po zdjęciu sygnału aktywacji.	
 T_n SE N	Timer Pojedynczy impuls Po uaktywnieniu zostaje wygenerowany pojedynczy impuls o czasie trwania „N”.	
 T_n SL N	Timer Impulsy Jeśli aktywny – to generowana jest fala prostokątna (impulsy) o wypełnieniu 50% (czas trwania „N” stanu wysokiego i czas trwania „N” stanu niskiego impulsów).	
 C_n CU N	Licznik w górę Po uaktywnieniu zliczane są impulsy – stan Licznika jest zwiększany na, przypisanym do danego Licznika wejściu. Po osiągnięciu progu „N”, Licznik przyjmuje wartość '1'	C, HC n: nr Licznika
 C_n CD N	Licznik w dół Po uaktywnieniu zliczane są impulsy – stan Licznika jest zmniejszany na, przypisanym do danego Licznika wejściu. Po zmniejszeniu licznika wartości bieżącej poniżej progu „N”, Licznik przyjmuje wartość '0'	

5.2.7. Konfiguracja

5.2.7.1. Konfiguracja wejść

Każdemu wejściu w programie (obwodzie) musimy przyporządkować rodzaj i zmienną. Rodzaj przyporządkowujemy w sposób graficzny – wybierając styk normalnie otwarty lub styk normalnie zwarty, zmienną umieszczamy nad symbolem graficznym. Zmienna, która określa typ wejścia składa się z oznaczenia literowego i numeru.



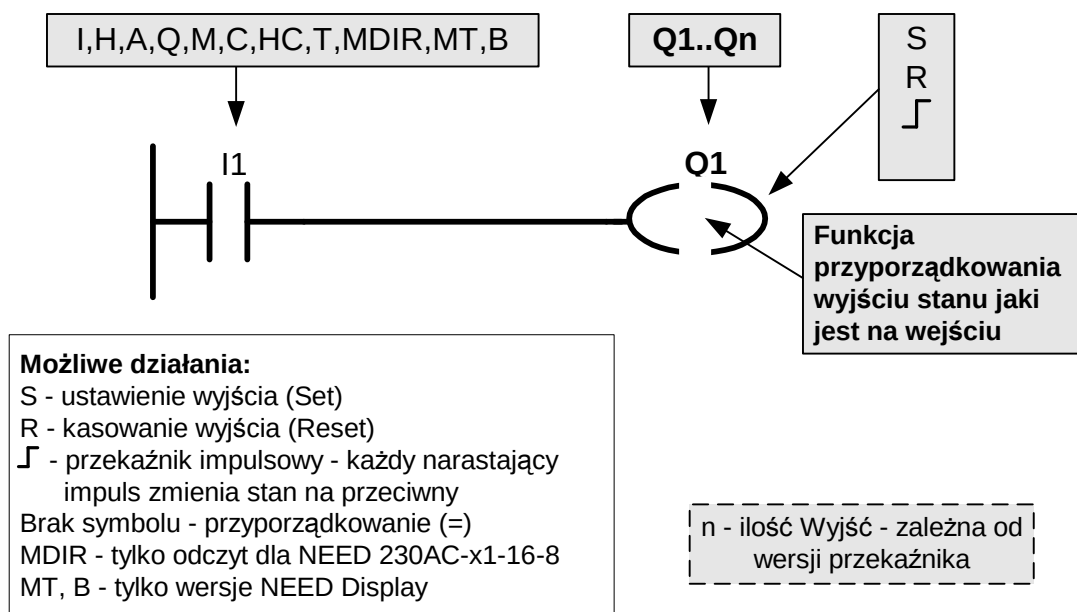
Rys. 5.2.7.1. Konfiguracja wejść.

Do dyspozycji mamy następujące zmienne:

In – wejścia,
 Hn – Zegary,
 An – komparacje analogowe,
 Qn – stany wyjść,
 Mn – stany Znaczników,
 Cn – stany Liczników,
 Tn – stany Timerów.
 HC1 – szybki licznik/miernik częstotliwości 0-20 kHz.
 MDIR – systemowy znacznik kierunku faz (tylko odczyt)
 MTn – Znaczniki tekstowe (Alarms NEED Display)
 Bn – przyciski panelu klawiatury NEED Display

5.2.7.2. Konfiguracja wyjść

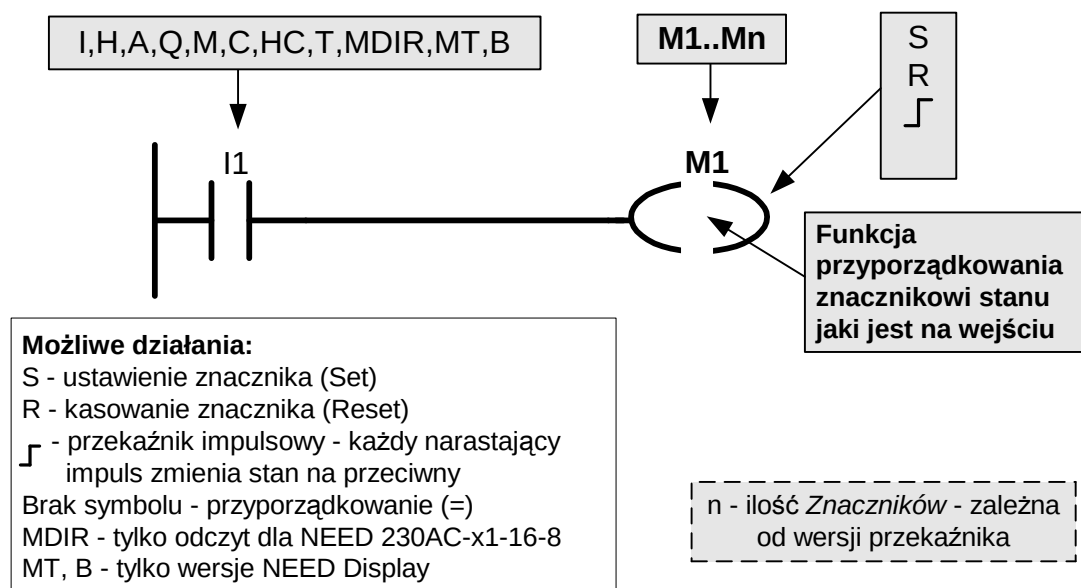
Fizyczne wyjścia, przedstawiamy za pomocą symbolu graficznego jak na rys. 5.2.7.2. W zależności od oczekiwanego zachowania wyjścia, stosujemy odpowiedni symbol graficzny. Nad symbolem graficznym umieszczamy literę Q, która oznacza wyjście oraz numer tego wyjścia.



Rys. 5.2.7.2. Konfiguracja wyjść.

5.2.7.3. Konfiguracja Znaczników (Markerów)

Znaczniki, podobnie jak wyjścia przedstawiamy za pomocą tego samego symbolu graficznego zamieniając Q na M (rys. 5.2.7.3). W zależności od oczekiwanej reakcji Znacznika stosujemy odpowiedni symbol wewnątrz graficznego oznaczenia. Nad symbolem graficznym umieszczamy literę M (ang. Marker) oraz numer. Znacznik MDIR jest znacznikiem systemowym tylko do odczytu.

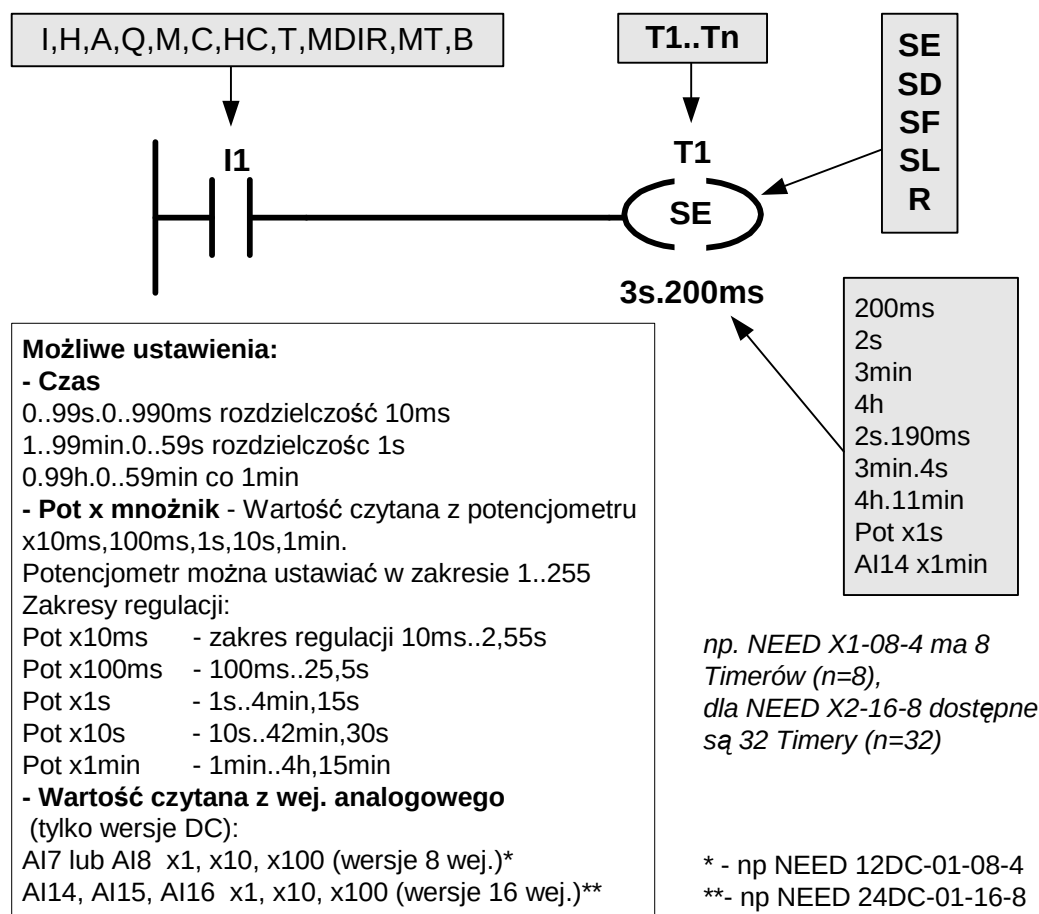


Rys. 5.2.7.3. Konfiguracja Znaczników.

5.2.7.4. Konfiguracja Timerów

Timery, przedstawiamy za pomocą tego samego symbolu graficznego jak wyjścia – rys.

5.2.7.4. W zależności od oczekiwanego sposobu działania Timera stosujemy odpowiedni symbol wewnątrz graficznego oznaczenia Timera. Nad symbolem graficznym umieszczamy literę T oraz numer.



Rys. 5.2.7.4. Konfiguracja Timerów.

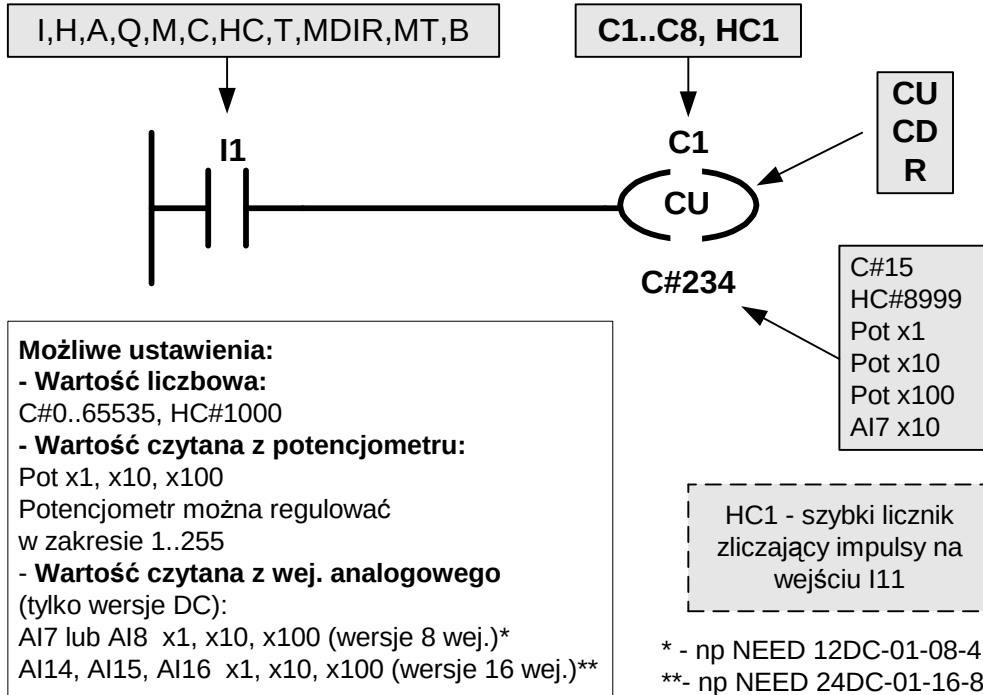


Istnieje możliwość używania, do odmierzania czasów dla Timerów, wartości napięć czytanych z wejść analogowych I7, I8 dla NEED-12DC-...-08-4, NEED-24DC-...-08-4 lub I14, I15, I16 dla NEED-12DC-...-16-8, NEED-24DC-...-16-8. Opisane dokładnie jest w rozdziale 5.1.2.21.1. Instrukcja 'L' dla Timerów.

5.2.7.5. Konfiguracja *Liczników*

Licznik, przedstawiamy za pomocą tego samego symbolu graficznego jak wyjścia – rys.

5.2.7.5. W zależności od oczekiwanego sposobu działania *Licznika* stosujemy odpowiednie symbole wewnątrz graficznego oznaczenia *Licznika*. Nad symbolem graficznym umieszczamy literę C, która oznacza *Licznik* (ang. *Counter*) oraz numer *Licznika*.



Rys. 5.2.7.5. Konfiguracja *Liczników*.



Istnieje możliwość używania, do ustawiania progu dla *Liczników*, wartości napięć czytanych z wejść analogowych I7, I8 dla NEED-12DC-...-08-4, NEED-24DC-...-08-4 lub I14, I15, I16 dla NEED-12DC-...-16-8, NEED-24DC-...-16-8. Opisane dokładniej jest w rozdziale 5.1.2.21.2. Instrukcja 'L' dla *Liczników*.



Wersje DC NEED-...-11-16-8 posiadają *Szybki Licznik* HC1 zliczający impulsy o maksymalnej częstotliwości 20kHz. HC1 jest *Licznikiem* sprzętowym, zlicza impulsy pojawiające się na wejściu I11. Wejścia CU, CD przejmują wtedy, oprócz funkcji kierunku zliczania, funkcję aktywacji *Szybkiego Licznika*.

Szybki Licznik może pracować w trybie częstotliwościowym – zlicza ilość impulsów pojawiających się na wejściu I11 w czasie 1s.

Szybki Licznik nigdy się nie przepęlnia. Próg do zliczania może być ustawiany w zakresie 0..65535. Wykonanie operacji Reset na *Szybkim Liczniku* kasuje stan i ilość zliczonych impulsów.

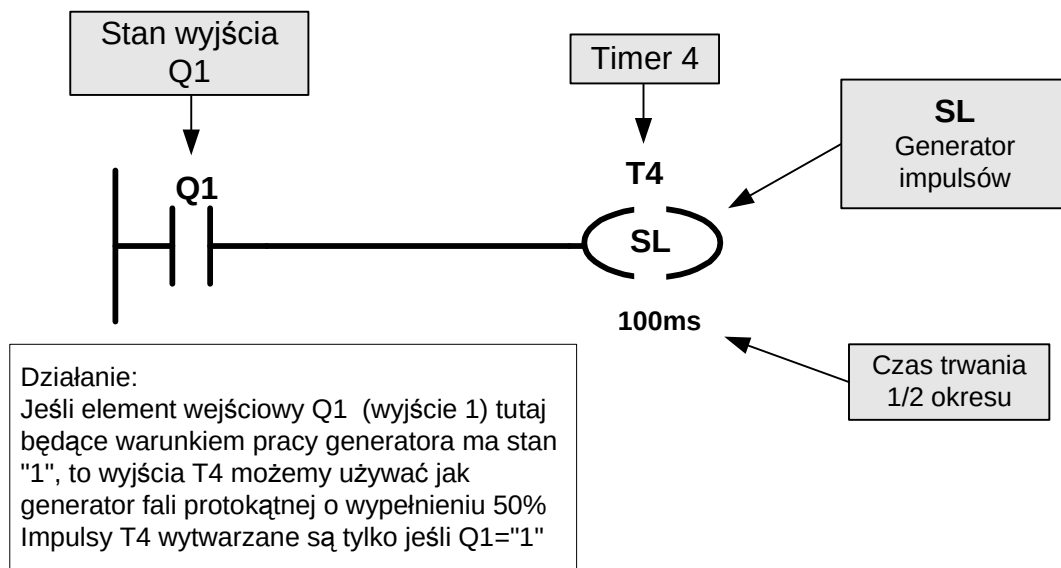


Dla wersji NEED-230AC-...-16-8 *Szybki Licznik* HC1 mierzy częstotliwość sieci (50Hz lub 60Hz) jeżeli wejście I11 jest aktywne.

Ponieważ częstotliwość sieci jest stała i znana można wykorzystać szybki licznik jako dodatkowy timer. Jeżeli próg = 1000 to dla 50Hz licznik przełączy się po $100 \times 20\text{ms} = 20\text{s}$.

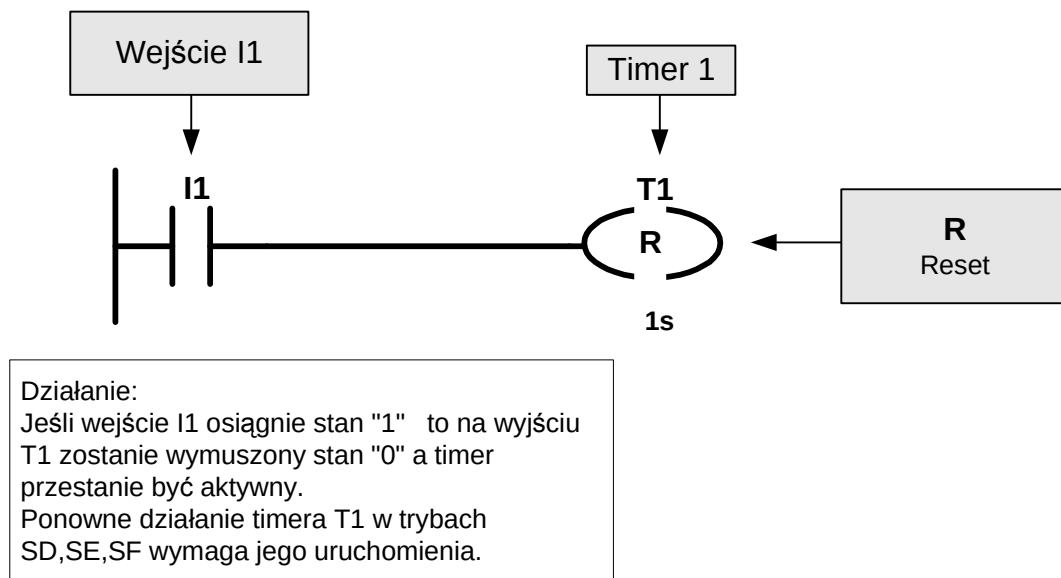
5.2.7.6. Przykłady konfiguracji

Przykład1: Timer SL Impulsy (Generator impulsów)



Rys. 5.2.7.6. Przykład konfiguracji Timera SL.

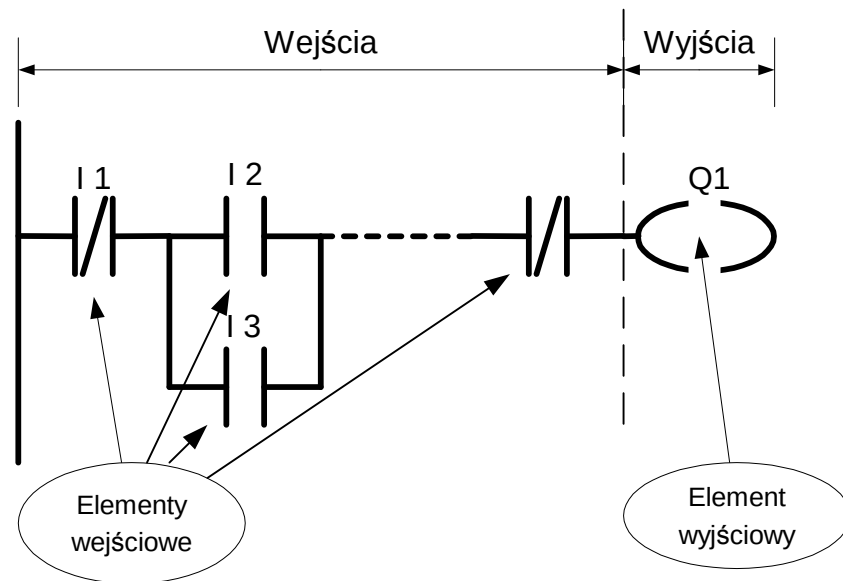
Przykład2: Reset Timera



Rys. 5.2.7.7. Przykład resetu Timera.

5.2.8. Zasady umieszczania elementów

Na rys 5.2.8.1. przedstawiono bardzo prosty obwód programu z rozmieszczeniem elementów zgodnie ze strukturą opisaną wcześniej. Dla ułatwienia na przykładach przedstawiane są dyskretne wejścia i wyjścia.



Rys. 5.2.8.1. Obwód w LAD.

Ogólnie obwód składa się z części wejściowej (warunkowej – poprzedzającej) i wyjściowej (wykonawczej – następującej). Pierwsza część określa warunki, jakie muszą być spełnione by zostało uaktywnione wyjście (element wykonawczy).

Elementy wejściowe mogą występować w różnych wzajemnych połączeniach, ich liczba podlega jedynie ograniczeniom ze względu na czytelność programu i możliwości edycyjne.



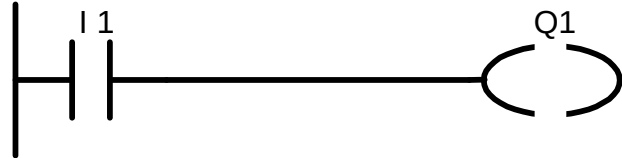
Uwaga: W przekaźniku NEED ilość wierszy jest ograniczona pojemnością pamięci programu zależną od wersji przekaźnika (np. wersja 24DC-X1-8-4 ma maksymalnie 862 bajty po kompilacji, a wersja 24DC-X2-16-8 3380 bajtów).

5.2.9. Rodzaje połączeń

Rozwiązanie układu sterowania wymaga stworzenia programu, który odpowiednio połączy zależności pomiędzy sygnałami wejściowymi i wyjściowymi.

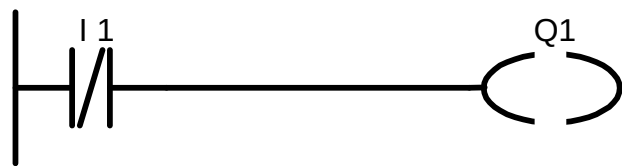
Podstawowe rodzaje połączeń przedstawiono poniżej.

5.2.9.1. Odwzorowanie wejścia na wyjście



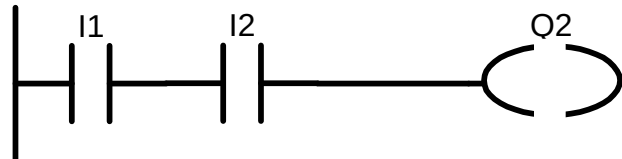
Stan wejścia I1 będzie przepisywany na wyjście Q1. Wyjście Q1 będzie aktywne ($Q1=1$), jeśli wejście I1 będzie w stanie logicznym '1'.

5.2.9.2. Odwzorowanie zanegowanego wejścia na wyjście



Zanegowany stan wejścia I1 będzie przepisywany na wyjście Q1. Wyjście Q1 będzie aktywne ($Q1=1$), jeśli wejście I1 będzie w stanie logicznym '0'.

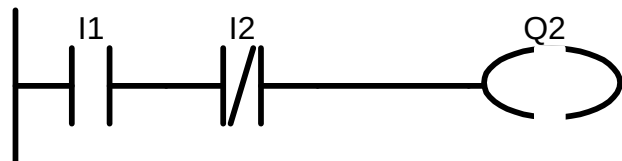
5.2.9.3. Połączenia szeregowe



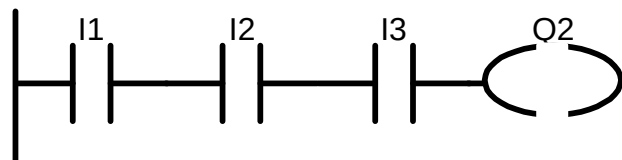
Przedstawiony powyżej układ realizuje funkcję iloczynu logicznego. Wyjście Q2 będzie aktywne ($Q2=1$), jeśli oba wejścia I1 i I2 będą w stanie logicznym '1'.

Poniżej przedstawiono inne rodzaje połączeń szeregowych

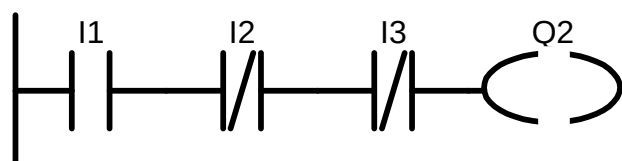
Wyjście Q2 będzie aktywne ($Q2=1$), jeśli wejście I1 będzie w stanie logicznym '1' a wejście I2 w stanie logicznym '0'.



Połączenie szeregowe 3 elementów. Wyjście Q2 będzie aktywne ($Q2=1$), jeśli wszystkie wejścia I1..I3 będą w stanie logicznym '1'.

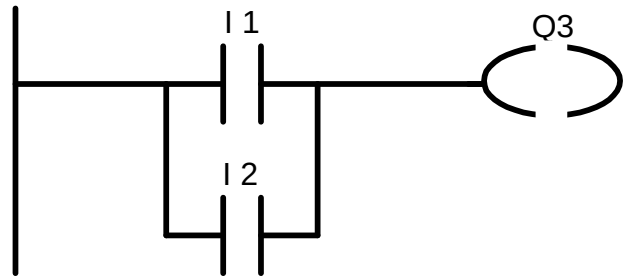


Połączenie szeregowe 3 elementów. Wyjście Q2 będzie aktywne ($Q2=1$), jeśli wejście I1 będzie w stanie logicznym '1' a wejścia I2, I3 będą w stanie logicznym '0'.



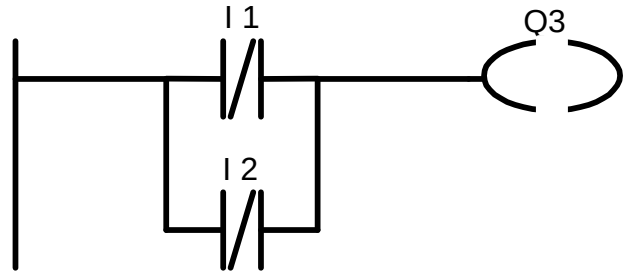
5.2.9.4. Połączenia równoległe

Przedstawiony obok układ realizuje funkcję sumy logicznej. Wyjście Q3 będzie aktywne ($Q3=1$), jeśli jedno z wejść I1, I2 (lub oba) będzie w stanie logicznym '1'.

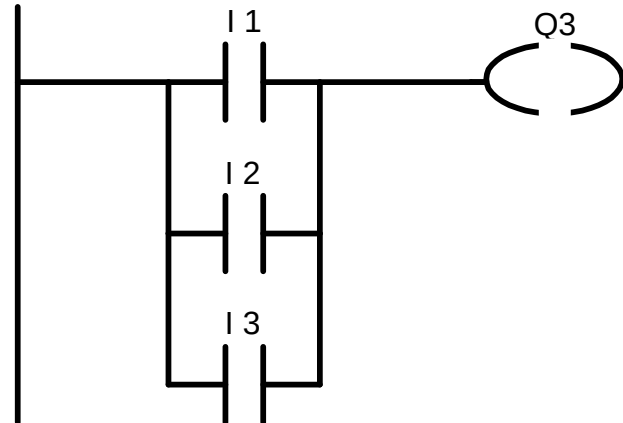


Poniżej przedstawiono inne rodzaje połączeń równoległych

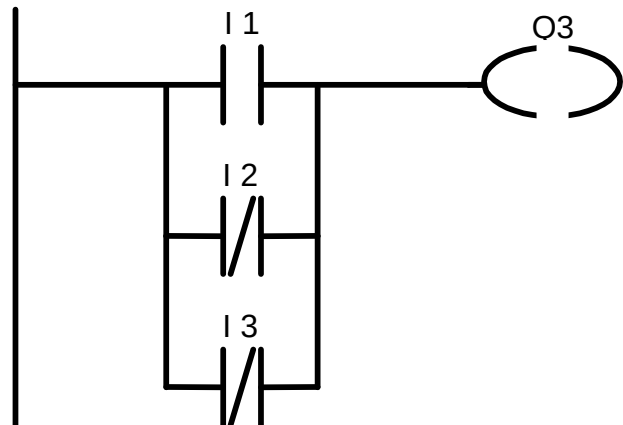
Wyjście Q3 będzie aktywne ($Q3=1$), jeśli jedno z wejść I1, I2 (lub oba) będzie w stanie logicznym '0'.



Przedstawiony obok układ realizuje funkcję sumy logicznej 3 elementów. Wyjście Q3 będzie aktywne ($Q3=1$), jeśli przynajmniej jedno z wejść I1, I2, I3 będzie w stanie logicznym '1'.



Suma logiczna 3 elementów. Wyjście Q3 będzie aktywne ($Q3=1$), jeśli wejście I1 będzie aktywne (stan '1') lub gdy jedno z wejść I2, I3 (lub oba) będzie w stanie logicznym '0'.



5.2.9.5. Połączenie szeregowo-równoległe

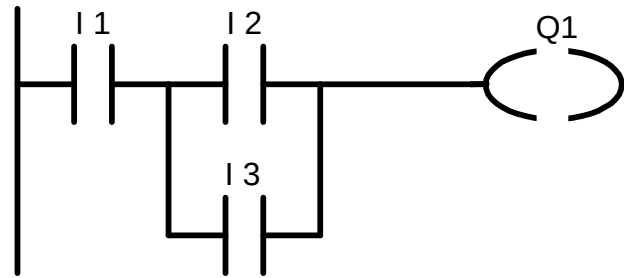
W celu przedstawienia układu sterowania opisane powyżej połączenia podstawowe można łączyć ze sobą w ramach dopuszczalnej ilości elementów wejściowych poziomych (3) i pionowych (150) zgodnie z zasadami tworzenia połączeń.

Jeśli algorytm wymaga użycia, dla wystereowania wyjścia, większej ilości elementów wejściowych, to należy odpowiednio zmodyfikować drabinkę połączeń używając do tego celu *Znaczników* tzn. podzielić zadania na mniejsze.

Poniżej przedstawiono przykłady obwodów wykorzystujących kombinacje połączeń szeregowo-równoległych wraz z interpretacją działania.

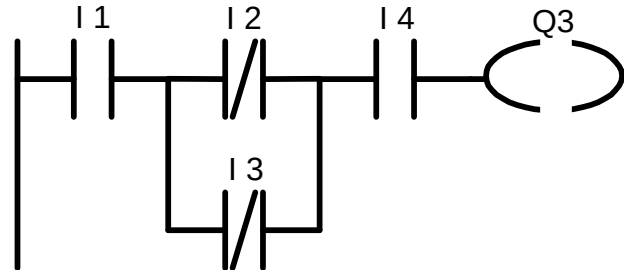
Układ realizujący połączenie szeregowe elementu I1 z równoległe połączonymi elementami I2, I3.

Działanie wyjścia Q1 jest następujące:
 $Q1 = '1'$, jeśli I1 jest aktywne (stan '1') i jedno z wejść I2, I3 (lub oba) jest w stanie logicznym '1'.

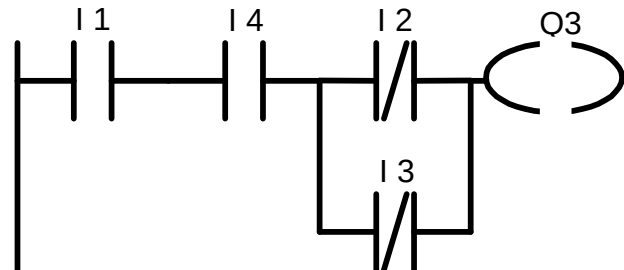


Układ realizujący połączenie szeregowe elementu I1 z równoległe połączonymi elementami I2, I3 oraz dalej szeregowo I4.

Działanie wyjścia Q3 jest następujące:
 $Q3 = '1'$, jeśli I1 i I4 jest aktywne (stan '1') i jedno z wejść I2, I3 (lub oba) jest nieaktywne (stan '0').



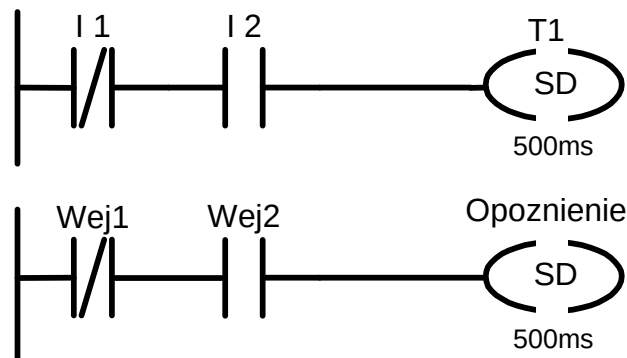
Równoważny powyżej układ przedstawiony jest w innej postaci: najpierw połączenie szeregowe I1, I4, a następnie połączenie równoległe I2 i I3.



5.2.10. Nazwy symboliczne

Dla przekaźników NEED istnieje możliwość przypisania nazw symbolicznych do zmiennych. Ułatwia to analizę programu i czyni go bardziej przejrzystym.

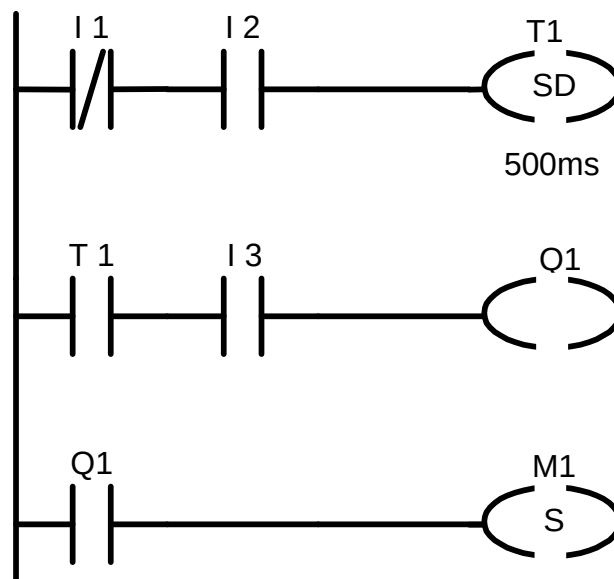
Możliwe jest przełączanie widoku zmienna/nazwa symboliczna. Na rys 5.2.10. przedstawiono obwód w zapisie zwykłym i poniżej z użyciem nazw symbolicznych.



Rys. 5.2.10. Przykład użycia symboli w LAD.

5.2.11. Program w LAD

Program składa się z obwodów, najprostszy program zawierać może tylko 1 obwód (linię programu). Poniżej przedstawiono program składający się z trzech obwodów.



Rys. 5.2.11. Przykładowy program zapisany w LAD.

Opis programu:

W pierwszym obwodzie jak na rys. 5.2.11. użyto wejść podłączonych bezpośrednio do przekaźnika programowalnego. Pierwsze wejście (I1) jest typu NC (normalnie zamknięte), drugie (I2) typu NO (normalnie otwarte), czyli załączenie *Timera* T1 nastąpi, jeśli I1 = '0' i I2 = '1'.

W drugim obwodzie (T1, I3, Q1) sprawdzamy stan *Timera* T1 (ustawianego w obwodzie 1) i wejścia I3. Jeśli *Timer* zostanie załączony (po czasie 500ms od spełnienia warunku I1='0' i I2='1') i wejście I3 będzie aktywne (I3 = '1'), to wyjście Q1 będzie w stanie wysokim (zasilane). Gdy wejście I3 zostanie wyłączone (I3 = '0') wyjście Q1 zostanie wyłączone.

Obwód 3 służy do zapamiętania załączenia wejścia Q1. Jeśli tylko wejście Q1 przyjmie stan „1”, to nastąpi ustawienie (na stałe) *Znacznika* M1 (M1 = '1').

Zauważmy, że powyższy program w zasadzie kończy się na ustawieniu *Znacznika* M1, gdyż dalsze operacje na *Znaczniku* M1 (np. resetowanie) nie są wykonywane.

6. INSTALACJA I OPIS OPROGRAMOWANIA

PC Need jest programem komputerowym, za pomocą którego można edytować, kompilować oraz ładować programy do pamięci przekaźnika programowalnego. Dodatkowo można także monitorować w trakcie pracy zasoby przekaźnika, dzięki czemu użytkownik może być na bieżąco informowany o stanach wejść, wyjść, *Timerach*, *Licznikach* itp. Pozwala to na całkowitą kontrolę aktualnie wykonywanego programu.

Prostota i różnorodność edycji programu (tekst lub grafika) sprawiają, iż PC Need jest bardzo wygodnym narzędziem, dzięki któremu nawet skomplikowane aplikacje powstają bardzo szybko, a czas ich uruchomienia jest krótki.

6.1. Wymagania sprzętowe

Dowolny komputer klasy PC ze złączem RS232 lub USB oraz z kartą graficzną VGA.

System operacyjny: Windows NT®, Windows 98®, Windows 2000®, Windows XP®, Windows Vista®.

6.2. Instalacja oprogramowania

1. Włóż płytę instalacyjną do napędu CD-ROM w komputerze.
2. Jeżeli program instalacyjny nie otworzy się automatycznie, to odszukaj na płycie plik „setup.exe” i uruchom go, aby rozpocząć instalację.
3. W czasie instalacji wskaż odpowiedni katalog, w którym zostanie zainstalowany PC Need. Jeżeli podczas instalacji została wybrana opcja umieszczenia ikon na pulpicie, to po właściwym zakończeniu instalacji na pulpicie powinna znaleźć się ikona programu. PC Need można także uruchomić z paska zadań.

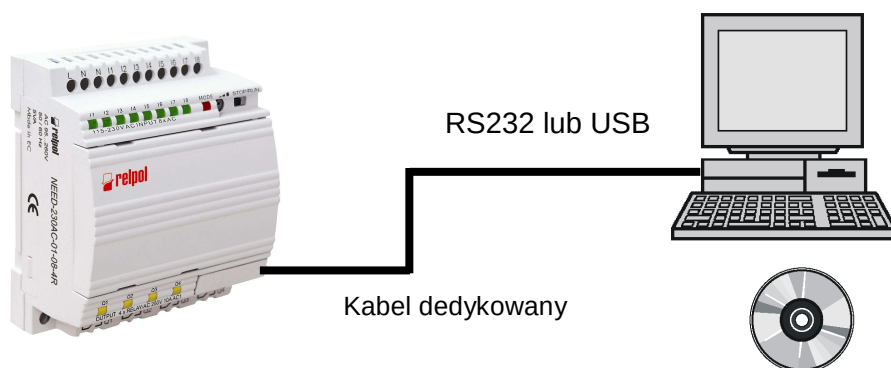
6.3. Deinstalacja

Aby usunąć program, należy użyć automatycznej deinstalacji: przycisk **Start > Programy > Relpol > Deinstalacja programu PC Need**.

Po wybraniu tej opcji program PC Need zostanie automatycznie usunięty z systemu.

6.4. Połączenie komputera z przekaźnikiem programowalnym

Komputer należy połączyć z przekaźnikiem programowalnym za pomocą dedykowanego kabla.



Rys. 6.4. Połączenie przekaźnika programowalnego NEED z komputerem.



Odwrotne podłączenie zasilania tzn. zamienienie miejscami przewodów fazowego (L) i neutralnego (N) na wejściach zasilających przekaźnika programowalnego dla wersji NEED-230AC-.. może spowodować wystąpienie niebezpiecznych napięć na porcie komunikacyjnym przekaźnika.

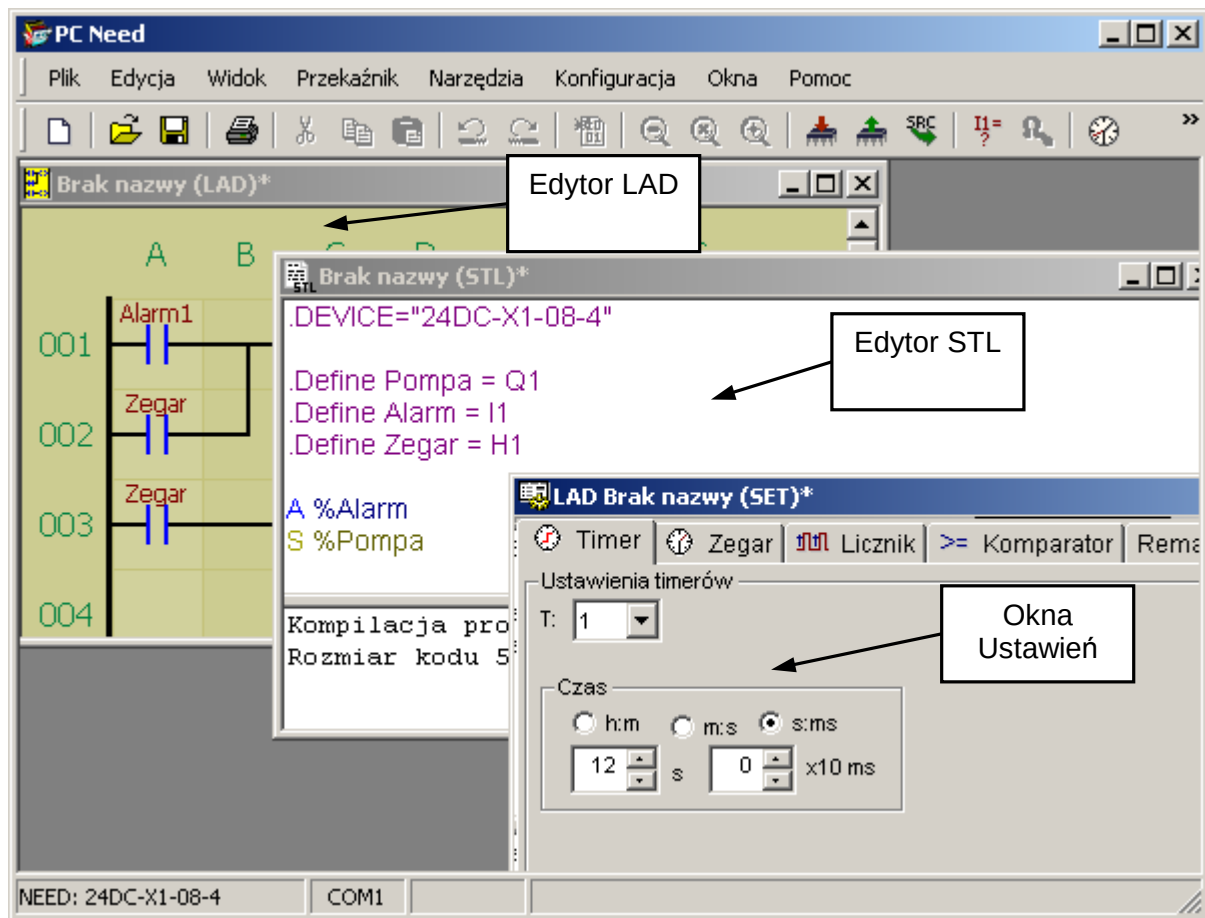
6.5. Szybki start – tworzenie aplikacji

Aby stworzyć konkretną aplikację sterowania opartą na przekaźniku programowalnym należy najpierw napisać odpowiedni program. W PC Need dostępne są dwa edytory:

- tekstowy do edycji programów STL,
- graficzny do edycji programów LAD.

Programy, w zależności od użytego edytora, zapisywane są jako pliki z rozszerzeniami „*.stn” (dla edytora tekstowego STL) lub „*.ldn” (dla edytora graficznego LAD).

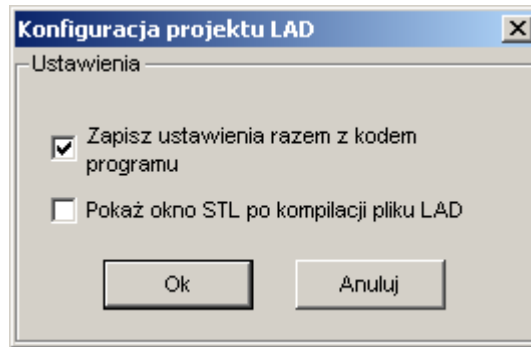
Do edycji zasobów przekaźnika NEED (ustawienie czasów załączenia dla Zegarów, ustawienie relacji porównań dla Komparatorów, wartości do zliczania i trybu pracy dla Liczników, czasu do odmierzenia i trybu pracy dla Timerów, wartości remanentnych) służy plik ustawień – „*.set”.



Rys. 6.5.1. Okna programu PC Need.

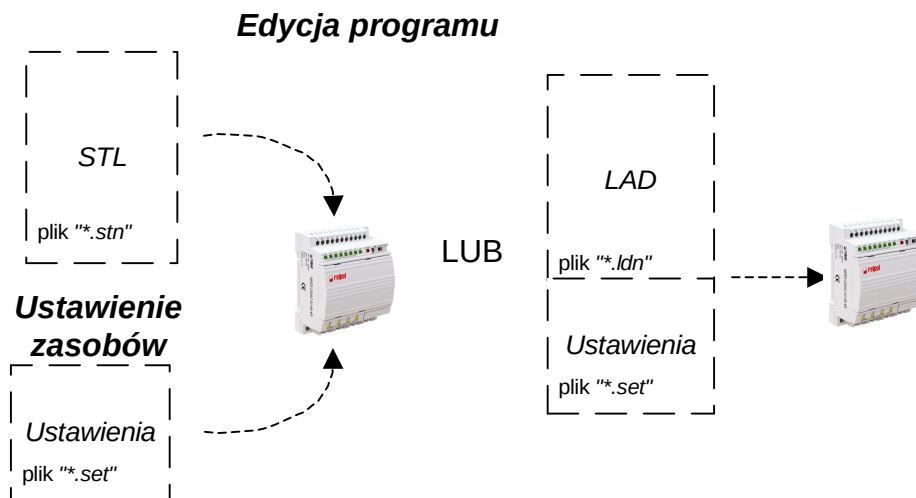
Projekt dla przekaźnika NEED powinien zawierać przynajmniej jeden plik z rozszerzeniem „*.stn” lub „*.ldn” (program użytkownika). Jeżeli programista korzysta z takich zasobów przekaźnika programowalnego jak Zegary czy Komparatory, wtedy oprócz pliku źródłowego (program STL lub LAD) do pamięci przekaźnika NEED należy załadować także ustawienia przekaźnika, edytowane w oknie ustawień – rys. 6.5.1.

Dla edytora LAD można w opcji, **Konfiguracja > Projekt LAD**, ustawić opcję **Zapisz ustawienia razem z kodem programu** (domyślnie opcja ta jest załączona – rys. 6.5.2.). Po zaznaczeniu tej opcji do pamięci przekaźnika załadowany zostanie plik programu „*.ldn” oraz okno ustawień.



Rys. 6.5.2. Okna konfiguracji projektu LAD.

Na rys. 6.5.2. przedstawiono schematycznie zawartość projektu dla przekaźnika programowalnego NEED.

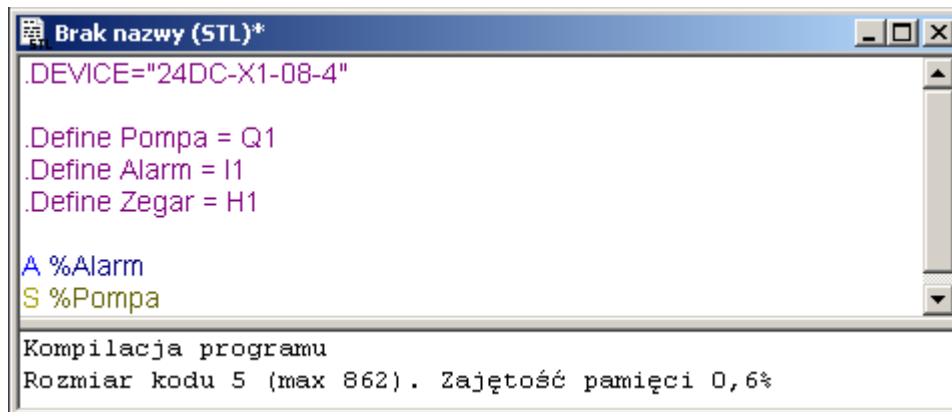


Rys.6.5.3. Pliki zawarte w projekcie dla przekaźnika programowalnego NEED.

Jeśli programista korzysta z Zegarów, Komparatorów, remanencji to, ustawienia powinny być zawsze ładowane do pamięci przekaźnika.

Przykład:

Projekt: program STL bez używania takich zasobów przekaźnika jak Zegary, remanencje, Komparatory itp. – rys. 6.5.4.

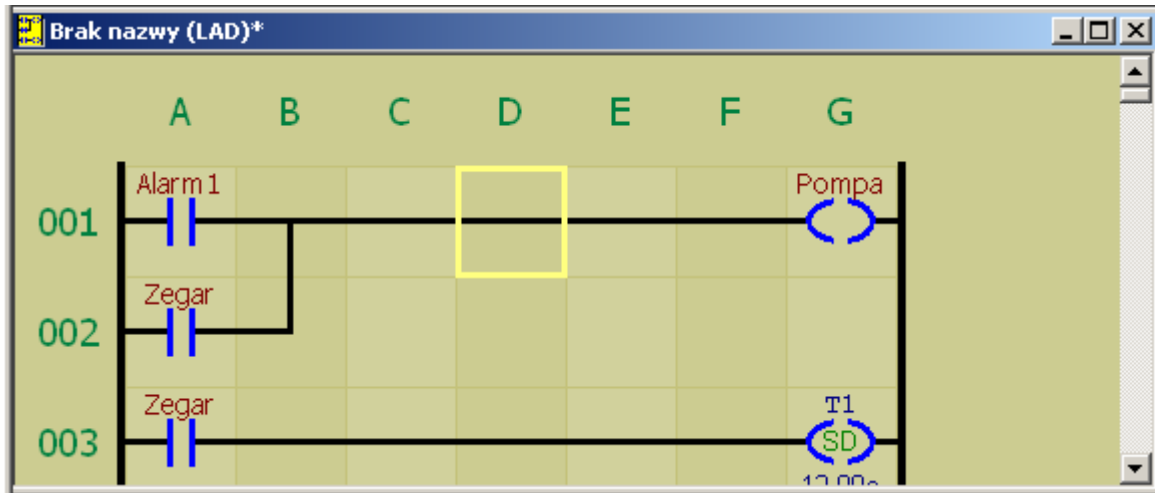


Rys.6.5.4. Program STL.

Do załadowania tylko plik „*.stn”.

Przykład:

Projekt: program LAD. Zaznaczona opcja „Zapisz ustawienia razem z kodem programu”

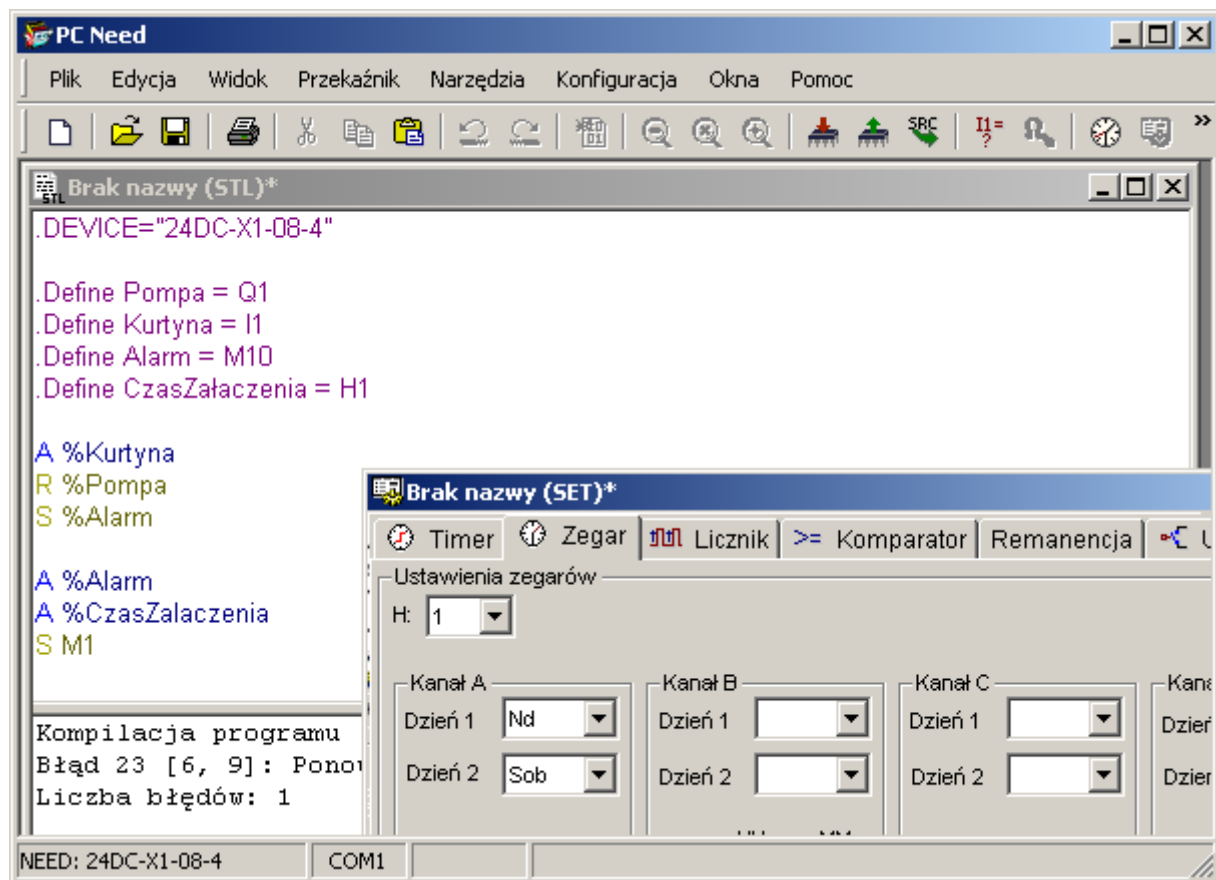


Rys.6.5.5. Program LAD.

Po wyborze z Menu: **Przełącznik > Transmisja > Zapis do przełącznika** do pamięci przełącznika programowalnego zostanie zapisany program LAD (plik „*.ldn”) oraz ustawienia.

Przykład:

Projekt: program STL , w którym wykorzystany jest Zegar i Komparator.

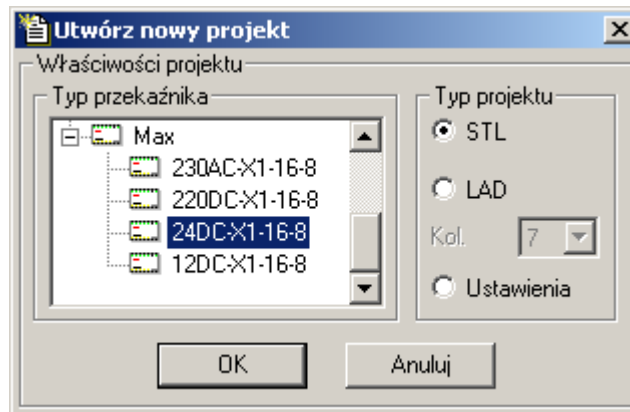


Rys.6.5.6. Program STL z wykorzystaniem Zegara i Komparatora.

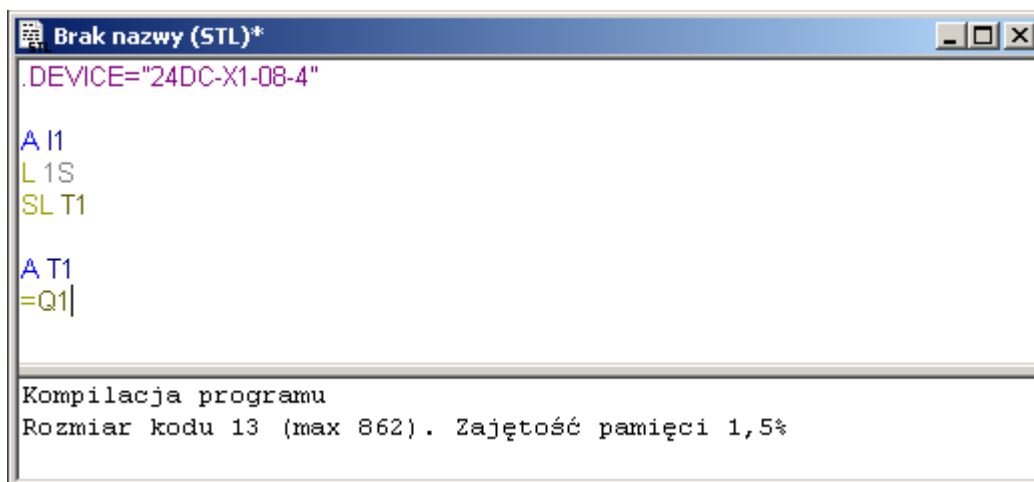
Do załadowania plik „Przykład2.stn” oraz Ustawienia „Przykład2.set” przekaźnika.

6.5.1. Przykład projektu – program STL

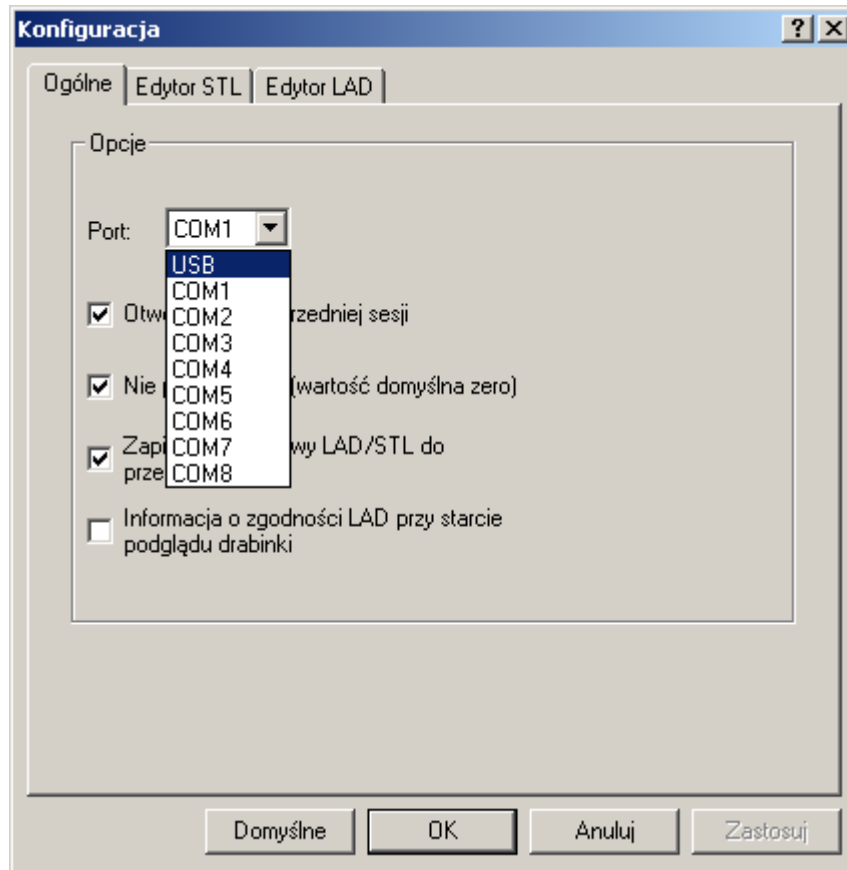
1. Uruchom program PC Need.
2. Utwórz nowy projekt np.: **Plik > Nowy** – w okienku wyboru (rysunek poniżej) wybierz odpowiedni typ przekaźnika oraz zaznacz w polu „Typ projektu” – **STL**.



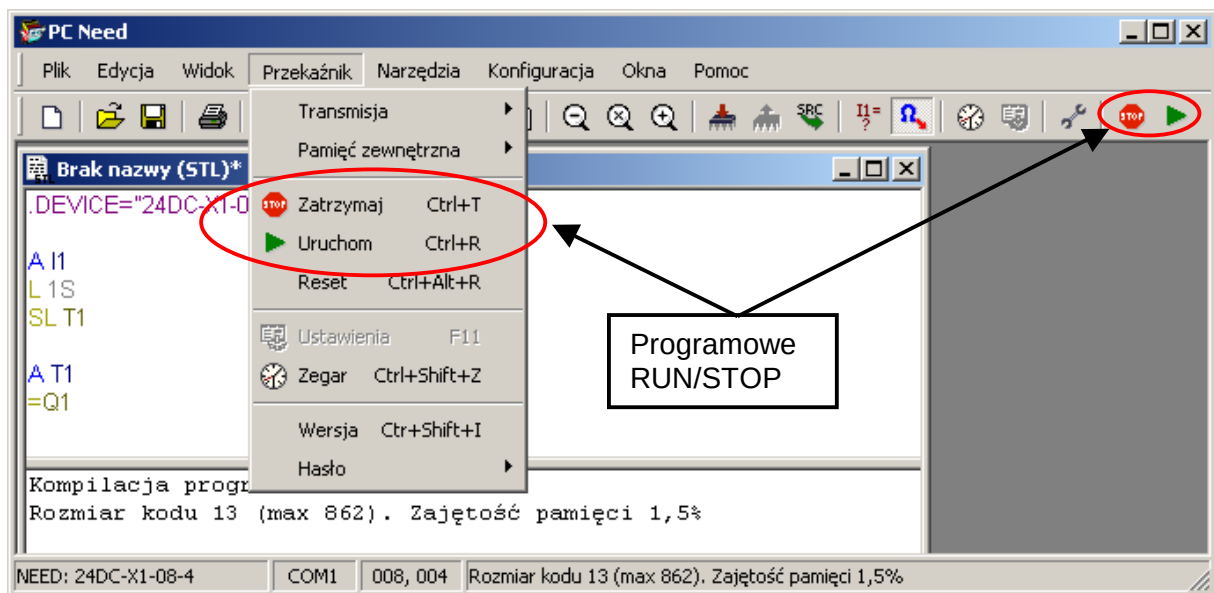
3. Napisz program np. taki jak poniżej:



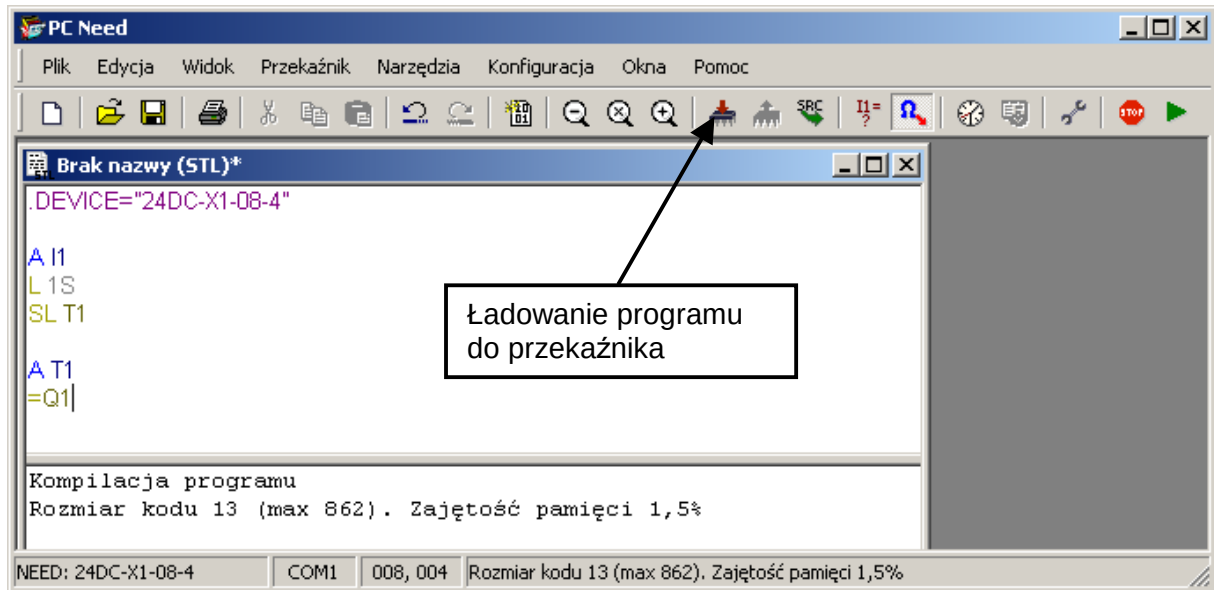
4. Połącz komputer z przekaźnikiem oraz dokonaj konfiguracji portu szeregowego RS232 lub USB: **Konfiguracja > Opcje > Port** – wybierz odpowiedni (wolny) port. Operację tą należy powtarzać tylko przy pierwszym uruchomieniu lub, gdy chcemy zmienić port komunikacyjny.



5. Ustaw przekaźnik w tryb STOP: (przełącznik lub **Przekaźnik > Zatrzymaj**).



6. Załaduj program do pamięci przekaźnika: **Przekaźnik > Transmisja > Zapis do przekaźnika**.

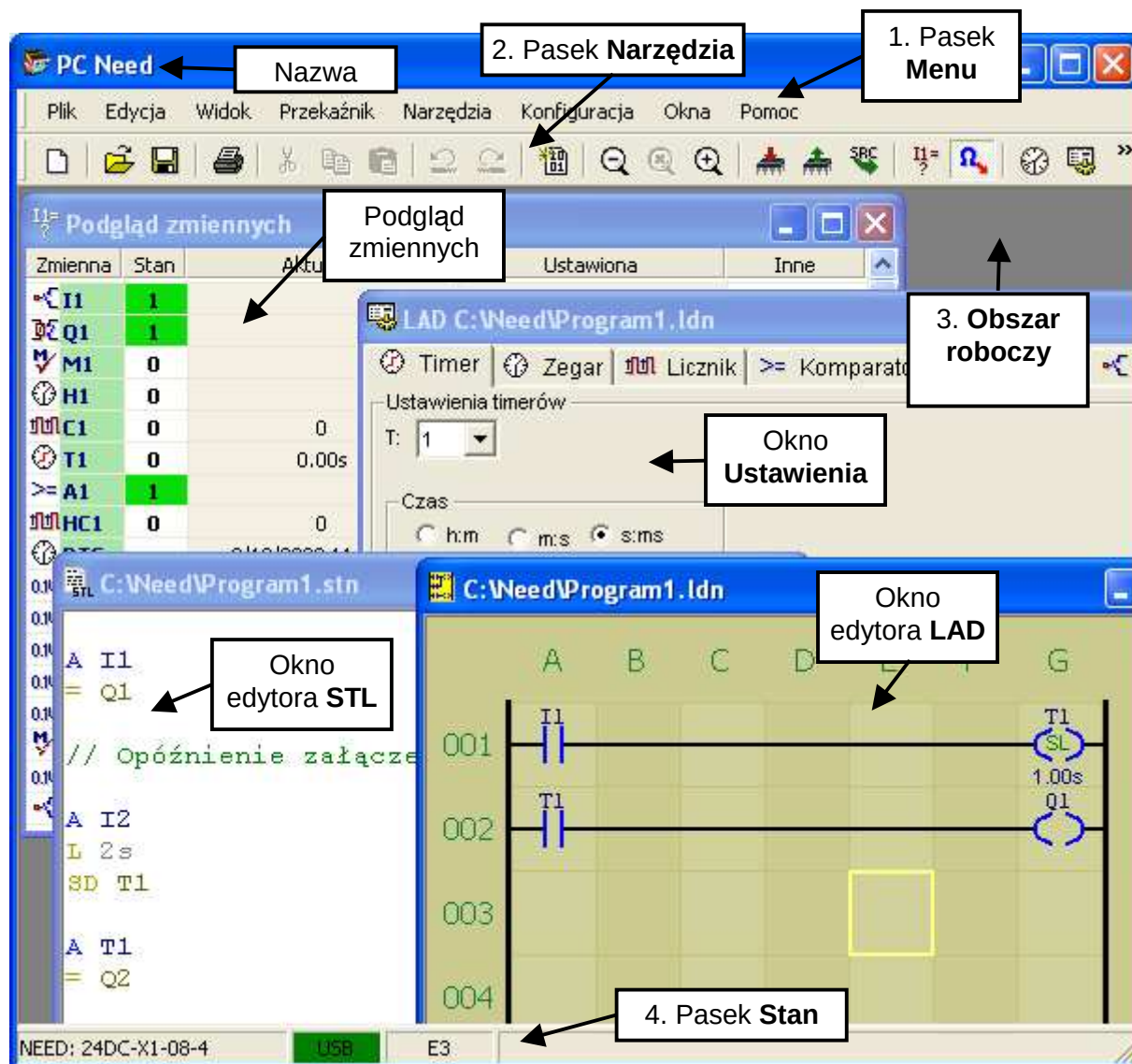


7. Przełącz przekaźnik w tryb RUN (przełącznik lub **Przekaznik > Uruchom**) i podaj na wejście I1 sygnał (stan wysoki). Wyjście Q1 powinno migać (cykliczne 1-sekundowe załączenie i 1-sekundowe wyłączenie).

6.6. Praca z programem PC Need

6.6.1. Opis głównego okna programu

Po uruchomieniu programu PC Need otwiera się okno – interfejs użytkownika.



Rys. 6.6.1. Interfejs użytkownika PC Need.

Interfejs użytkownika tworzą:

1. Pasek Menu.

Plik Edycja Widok Przekaznik Narzędzia Konfiguracja Okna Pomoc

2. Pasek Narzędzia.



3. Obszar roboczy – w nim otwierane okna, między innymi: **Edytor LAD, Edytor STL, Ustawienia, Podgląd zmiennych, Konfiguracja, Ustawienia elementu.**

4. Pasek Stan.

NEED: 24DC-X1-08-4 USB D3 Rozmiar kodu 9 (max 862). Zajętość pamięci 1,0%

6.6.2. Pasek Menu

Pliki – zarządzanie operacjami na plikach

- > **Nowy** – otwarcie okna wyboru typu przekaźnika i projektu: STL, LAD, Ustawienia
 - >> **STL** – utworzenie nowego pliku w edytorze języka STL
 - >> **LAD** – utworzenie nowego projektu w edytorze języka LAD
 - >> **Ustawienia** – utworzenie nowego pliku ustawień SET
- > **Otwórz** – otwarcie istniejącego pliku do edycji lub zmiany ustawień; otwierane pliki:
 - *.stn – pliki zapisane w języku tekstowym STL
 - *.ldn – pliki zapisane w języku drabinkowym LAD
 - *.set – pliki ustawień (SET)
 - („*” - nazwa pliku; .stn rozszerzenie - rodzaj pliku)
- > **Zapisz** – zapisanie pliku na dysku
- > **Zapisz jako** – zapisanie pliku na dysku poprzez utworzenie nowego pliku
- > **Dokument** – informacje o tworzonym oprogramowaniu (do tabeli wydruku)
- > **Konwertuj dokument** – przenoszenie programów LAD i SET na inne typy przekaźników.
- > **Ustawienie strony** – zarządzanie ustawieniami wydruku projektu
- > **Podgląd wydruku**
- > **Drukuj** – wydruk dokumentu
- > **Ostatnie projekty** – szybkie linki do ostatnich projektów
- > **Zakończ** – zakończenie pracy z programem PCNEED

Edycja – polecenia edycji programu

- > **Cofnij**: cofnięcie ostatniej operacji
- > **Powtórz**: powtórzenie ostatnio wykonanego cofnięcia
- > **Wytnij**: wycięcie zaznaczonej zawartości
- > **Kopiuj**: skopiowanie zaznaczonej zawartości
- > **Wklej**: wstawienie zawartości w zaznaczone miejsce
- > **Wyczyść**: usunięcie zaznaczonej zawartości
- > **Szukaj**: okno wyszukiwania (STL, LAD)
- > **Szukaj następny**: wyszukiwanie kolejnego takiego samego elementu (STL, LAD)
- > **Zamień**: możliwość zamiany ciągu znaków na inny

Widok – ustawianie parametrów okna programu NEED – aktywne, jeśli edytowany (otwarty) program w języku drabinkowym LAD

- > **Powiększenie**: dostosowanie obszaru w oknie edytora LAD
 - >> **Powiększ** – powiększenie rozmiaru
 - >> **Pomniejsz** – pomniejszenie rozmiaru
 - >> **Normalny** – rozmiar domyślny
- >> **Okno STL** – wyświetla okno STL z wynikowym kodem kompilacji języka LAD.

Przekaźnik – zestaw funkcji do obsługi przekaźnika

- > **Transmisja** – obsługa komunikacji z przekaźnikiem
 - >> **Zapis do przekaźnika** – kompilacja programu i przesłanie kodu wykonawczego programu do przekaźnika lub zapis nowych ustawień (w zależności od aktywnego aktualnie okna)
 - >> **Odczyt z przekaźnika** – odczyt ustawień z przekaźnika, lub odczyt wartości dla podglądu zmiennych (w zależności od aktywnego aktualnie okna)
 - >> **Odczyt kodu programu** – odczyt kodu źródłowego z przekaźnika
 - >> **Porównaj** – porównanie kodu aktualnego programu zapisanego na

dysku, z programem zapisanym w przekaźniku (porównanie z aktywnym otwartym programem w edytorze)

- > **Pamięć zewnętrzna** – obsługa modułu pamięci
 - >> **Zapis** – zapis aktualnego programu lub ustawień do modułu pamięci. Zapisywany jest aktualnie otwarty plik programu lub ustawień. Jeżeli chcemy wgrać program i ustawienia, to należy przeprowadzić zapis dwukrotnie, raz dla aktywnego okna programu, raz dla aktywnego okna pliku ustawień
 - >> **Odczyt** – odczyt ustawień z modułu pamięci
 - >> **Status** – informacje o stanie partycji pamięci oraz deaktywacja partycji
- > **Zatrzymaj** – polecenie wprowadzające przekaźnik w tryb STOP
- > **Uruchom** – polecenie wprowadzające przekaźnik w tryb RUN
- > **Reset** – kasowanie pamięci programu, ustawień i hasła w przekaźniku
- > **Ustawienia** – otwarcie okna edycji ustawień
- > **Zegar** – otwarcie okna zarządzania czasem w przekaźniku
- > **Wersja** – informacja o typie i wersji oprogramowania przekaźnika
- > **Hasło** – zabezpieczenie przed odczytem i zapisem programu w przekaźniku programowalnym
 - >> **Wprowadź** – wprowadzanie hasła do weryfikacji z hasłem w przekaźniku
 - >> **Zmień** – zmiana istniejącego hasła z jego weryfikacją

Narzędzia – zestaw funkcji do uruchamiania aplikacji

- > **Kompilacja** – kompilacja programu
- > **Ustawienia elementu** – otwarcie okna z parametrami styku (edytor LAD)
- > **Podgląd zmiennych** – otwarcie okna do odczytu aktualnych wartości zmiennych w przekaźniku. Uaktywnienie podglądu **Przekaźnik > Uruchom**. W tabeli należy wpisać, które zmienne mają podlegać odczytowi.
- > **Nazwy symboliczne** – zmiana sposobu wyświetlania zmiennych - rejestry/nazwy symboliczne

Konfiguracja – opcje programu PC Need

- > **Opcje**
 - Ogólne – wybór portu komunikacyjnego, decyzja o otwieraniu plików z poprzedniej sesji, rezygnacja z zabezpieczania hasłem
 - Edytor STL – ustawienia edytora domyślne lub własne
 - Edytor LAD – ustawienia edytora domyślne lub własne
- > **Projekt LAD** – otwarcie okna – opcje zapisu kodu programu i ustawień oraz decyzja o otwieraniu okna STL po kompilacji programu LAD
- > **Język** – otwarcie okna wyboru języka interfejsu programu.

Okna – zarządzanie otwartymi oknami w obszarze roboczym programu NEED

- > **Kaskadowo** – rozmieszczenie okien jedno nad drugim
- > **Sąsiadująco** – rozmieszczenie okien obok siebie
- > **Informacja o oknach**

Pomoc – plik pomocy i informacje o programie

- > **Indeks** – tematy pomocy
- > **PC Need – informacje** – o wersji programu i producencie.

6.6.3. Pasek narzędzi

Dla najczęściej używanych opcji menu dostępne są przyciski na pasku narzędzi umożliwiające szybsze otwarcie danej funkcji programu po kliknięciu na wybrany przycisk. Poniżej w skrócie opis przycisków na pasku narzędzi.

	Nowy	Otwiera nowy projekt (plik)
	Otwórz	Otwiera istniejący projekt (plik)
	Zapisz	Zapisuje aktywny projekt
	Drukuj	Drukuje aktywny projekt
	Wytnij	Wycina zaznaczenie
	Kopiuj	Kopiuje zaznaczenie
	Wklej	Wkleja zaznaczenie
	Cofnij	Cofa ostatnią operację
	Ponów	Ponawia cofniętą operację
	Kompilacja	Kompilacja aktywnego projektu
	Pomniejsz	Pomniejsza rozmiar zawartości okna (LAD, Podgląd zmiennych)
	Widok normalny	Ustawia widok domyślny (LAD, Podgląd zmiennych)
	Powiększ	Powiększa rozmiar zawartości okna (LAD, Podgląd zmiennych)
	Zapis	Zapis (transmisja) do przekaźnika (LAD, STL, SET)
	Odczyt	Odczyt (transmisja) z przekaźnika (LAD, SET, Podgląd drabinki, Podgląd zmiennych)
	Odczyt źródła	Odczyt z przekaźnika programu źródłowego
	Podgląd zmiennych	Odczyt stanu wybranych zmiennych z przekaźnika
	Nazwy symboliczne	Przełączanie widoku adres/symbol
	Zegar	Otwiera okno zegara czasu rzeczywistego (RTC)
	Ustawienia	Otwiera okno ustawień
	Opcje	Dostęp do opcji programu (m in. wybór portu komunikacyjnego, opcje edytorów)
	Stop	Ustawienie trybu STOP w przekaźniku (Stop programu)
	Run	Ustawienie trybu RUN w przekaźniku (Start programu)

6.6.4. Skróty klawiaturowe

PC Need umożliwia wykonywanie większości poleceń za pomocą klawiatury.

Poniżej przypisane klawiszom bądź ich kombinacjom funkcje.

PC Need

F1	Wyświetl "Help'a"
F3	Szukaj następnego wystąpienia słowa (w STL) lub elementu w LAD. Wcześniej powinno być wyświetlone okno „Szukaj”
F5	Zapisz dane do przekaźnik (STL lub LAD lub/i SET)
F6	Czytaj dane z przekaźnika (SET)
F7	Kompilacja
F11	Wyświetl okno ustawień (dla aktywnego okna LAD)
F12	Wyświetl okno podglądu zmiennych
CTRL + "N"	Okno "Nowy projekt"
CTRL + "O"	Okno "Otwórz plik"
CTRL + "S"	Zapisz plik w aktywnym oknie
CTRL + "P"	Print - Wydruk dokumentu
ALT + F4	Zakończenie pracy z programem NEED
CTRL + "Z"	Operacja "Cofnij"
CTRL + "Y"	Operacja "Powtórz"
CTRL + "X"	Operacja "Wytnij"
CTRL + "C"	Operacja "Copy"
CTRL + INSERT	Operacja "Copy"
CTRL + "V"	Operacja "Wklej"
SHIFT + INSERT	Operacja "Wklej"
CTRL + NUM+	Powiększenie rozmiaru zawartości okna (Zwiększ zoom)
CTRL + NUM-	Pomniejszenie rozmiaru zawartości okna (Zmniejsz zoom)
CTRL + NUM*	Rozmiar domyślny zawartości okna (Zoom "normalny")
CTRL + T	Ustaw tryb STOP w przekaźniku
CTRL + R	Ustaw tryb RUN w przekaźniku
CTRL + ALT + "R"	Reset przekaźnika
SHIFT + CTRL + "Z"	Otwarcie okna zarządzania czasem w przekaźniku
CTRL + SHIFT + "I"	Informacja o typie i wersji oprogramowania w przekaźniku

SHIFT + CTRL + "A"	Wyświetl okno "Ustawienie elementu" (dla aktywnego okna LAD)
ALT + "/"	Wyświetl okno "PCNeed - Informacje"

Edytor LAD

A	Wstaw Komparator
C	Wstaw Licznik
D	Wstaw MDIR
H	Wstaw Zegar
I	Wstaw Wejście
M	Wstaw Znacznik
Q	Wstaw Wyjście
T	Wstaw Timer
SHIFT + "C"	Wstaw Szybki licznik (HC)
CTRL + "A"	Zaznacz cały schemat LAD
SPACJA	Rysuj "linię" poziomą
ENTER	Pokaż okno "Ustawienia elementu"
ALT lewy + ←	Jeśli aktualnie wybrany jest jakiś element to strzałki LEWO i PRAWO zmieniają funkcje NO/NC FP,SE, SD, SF,... Jeśli aktualnie nie jest wybrany żaden element to przytrzymanie lewego ALT i strzałki kierunku umożliwia rysowanie w poziomie linii połączeń elementów
ALT lewy + →	
ALT lewy + ↑	Jeśli aktualnie wybrany jest jakiś element to strzałki GÓRA i DÓŁ zmieniają numer rejestru. Jeśli aktualnie nie jest wybrany żaden element to przytrzymanie lewego ALT i strzałki kierunku umożliwia rysowanie w pionie linii połączeń elementów.
ALT lewy + ↓	
HOME	Działają tak jak w edytorach tekstowych. Można dodatkowo używać je razem z SHIFT'em do zaznaczania.
END	
Page Up	
Page Down	

Objaśnienie:

F2 – naciśnięcie klawisza funkcyjnego F2

CTRL+S – jednoczesne naciśnięcie klawisza Ctrl i Klawisza S

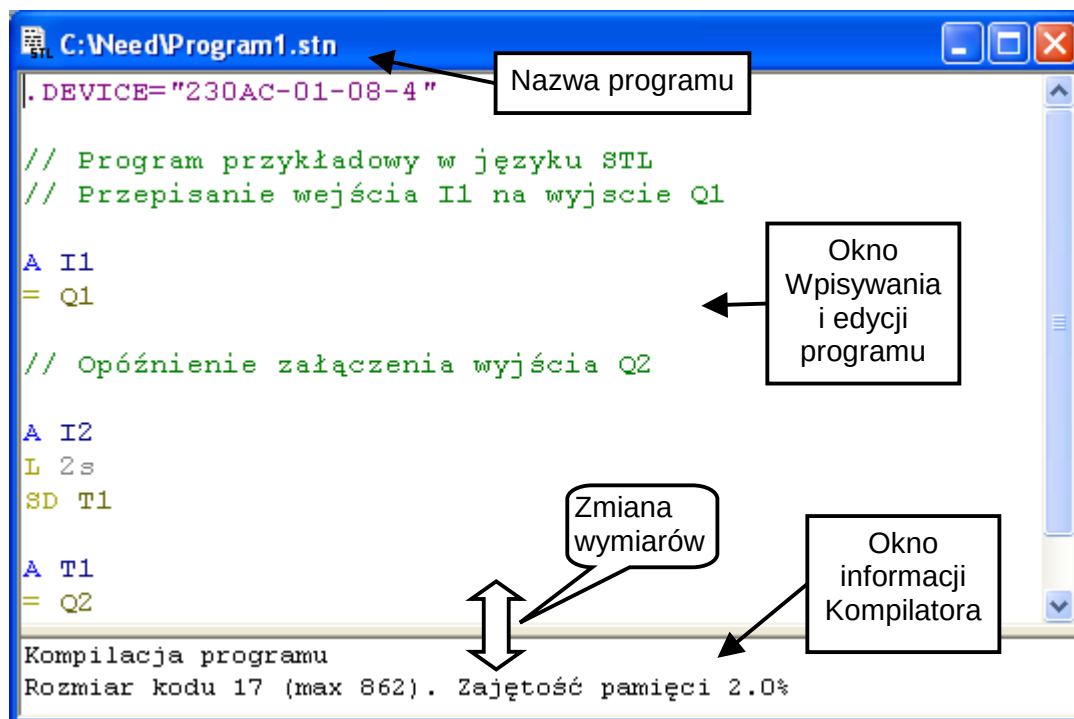
CTRL+NUM+ – jednoczesne naciśnięcie klawisza Ctrl i Klawisza + sekcji numerycznej klawiatury

CTRL+ALT+ "R" – jednoczesne naciśnięcie klawiszy Ctrl i Alt (lewy) oraz klawisza R

CTRL+SHIFT+ "Z" – jednoczesne naciśnięcie klawiszy Ctrl i SHIFT oraz klawisza Z

6.7. Edytor programu STL

Edycja programu w języku STL odbywa się w edytorze STL – okno poniżej.



Rys. 6.7.1. Edytor STL.

6.7.1. Edytor STL

Okno edytora otwierane jest w obszarze roboczym programu PC Need i podlega zachowaniom standardu Windows® – zmiana rozmiarów i położenia, zamykanie. Obsługa edytora jest podobna do obsługi prostego notatnika. Tekst wprowadzamy z klawiatury, stosując zasady składni podane w rozdziale 5.1. Programowanie w języku tekstowym STL.

Możliwe są operacje usuwania, przenoszenia, kopiowania zaznaczonego fragmentu bądź całego tekstu.

- **Zaznaczanie** – za pomocą myszki lub klawiatury (klawisze SHIFT+ strzałki) podświetl wybrany tekst.
- **Wycinanie** – zaznacz tekst, który ma być wycięty a następnie wykonaj polecenie *Wytnij*, czyli kombinację klawiszy Ctrl+X (jednoczesne naciśnięcie klawiszy Ctrl i X). Wycięty tekst będzie zachowany w schowku.
- **Usuwanie** – zaznacz tekst, który ma być usunięty a następnie wykonaj polecenie *Usuń* – klawisz Del.
- **Wklejanie** – ustaw kursor w miejscu, gdzie ma być początek wklejanego tekstu, wykonaj polecenie *Wklej* – kombinacja klawiszy Ctrl+V (jednoczesne naciśnięcie klawiszy Ctrl i V). Po wykonaniu operacji następuje wstawienie zawartości schowka systemowego.
- **Przenoszenie** – polega na wykonaniu operacji zaznaczenia (*Zaznaczanie*) wybranego tekstu, wycięcia go (*Wytnij* Ctrl+X) a następnie wklejenia go w żądanym miejscu (*Wklej* Ctrl+V).
- **Kopiowanie** – zaznacz tekst, który ma być skopiowany (patrz: *Zaznaczanie*) a następnie polecenie *Kopiuj* Ctrl+C (jednoczesne naciśnięcie klawiszy Ctrl i C).
- **Szukanie** – otwórz okno *Szukaj* (Ctrl+F) i wpisz poszukiwany tekst.
- **Zamiana** – otwórz okno *Zamień* (Ctrl+H) i wypełnij pola *Znajdź:* i *Zamień na:* oraz sposób wykonania akcji zamiany.

Komentarze

Edytowany program w celu zwiększenia czytelności możemy opatrzyć komentarzami. Tekst objęty komentarzem nie jest analizowany przy tworzeniu kodu wykonawczego. Komentarz, który rozpoczyna się znakiem `//` lub `;` obowiązuje do końca linii. Taki komentarz możemy zacząć od początku linii lub po wpisaniu instrukcji.

Przykłady:

```
// To jest komentarz od początku linii.
; To też jest komentarz od początku linii.
A I1 ;To jest komentarz po instrukcji.
```

W celu pominięcia w trakcie kompilacji większej liczby linii możemy zastosować komentarz typu: `/* tekst */`. W tym typie komentarza określamy obowiązkowo początek i koniec tekstu, który nie będzie należał do kodu programu.

Przykład:

```
/* A I1
A I2
= Q1
*/
A I3
=Q2
```

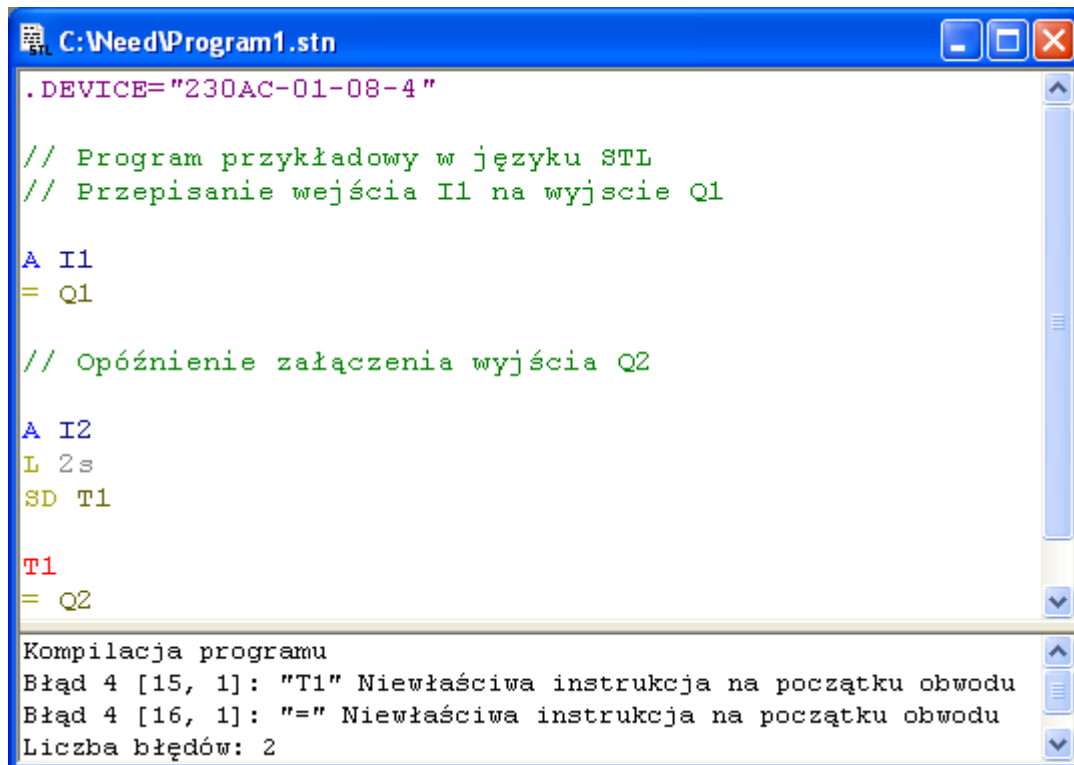
W powyższym przykładzie trzy pierwsze instrukcje będą pominięte przy kompilacji programu. Dopiero od instrukcji A I3 będzie tworzony kod wynikowy.

6.7.2. Kompilacja STL

Dolne okno początkowo jest puste, po wykonaniu komendy kompilacja (klawisz F7) pojawia się raport z kompilacji. Jeśli program jest poprawny, to otrzymamy komunikat jak na rys.

6.7.1. o zakończeniu kompilacji naszego programu oraz o rozmiarze kodu i procentowej zajętości pamięci w przekaźniku.

Jeśli program będzie zawierał błędy, to otrzymamy komunikat o typie błędu i jego położeniu [numer wiersza, numer kolumny] – rys. 6.7.2. wraz z komentarzem.



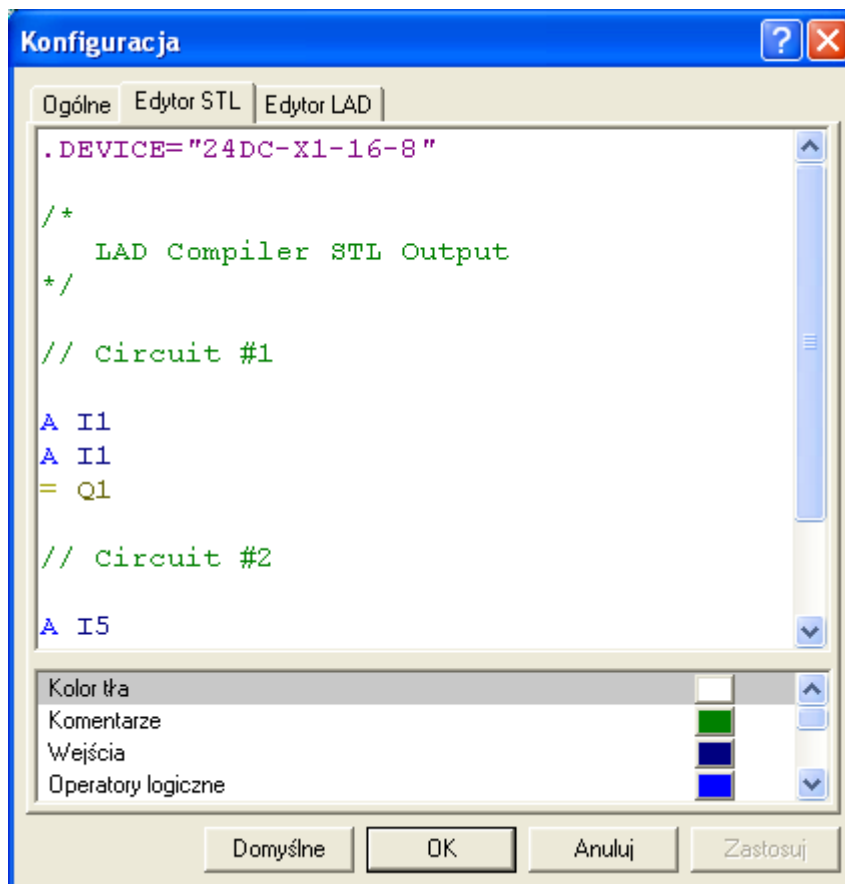
Rys. 6.7.2. Błąd kompilacji.

6.7.3. Konfiguracja edytora STL

W Menu **Konfiguracja > Opcje** w zakładce *Edytor STL* możemy dostosować wygląd okna edycji do własnych preferencji. Do ustawienia są:

- kolor tła
- kolor czcionki komentarzy
- kolor czcionki elementów wejściowych
- kolor czcionki operatorów logicznych
- kolor czcionki elementów wyjściowych
- kolor czcionki funkcji na wyjściach
- kolor czcionki argumentów
- kolor czcionki dyrektyw (.DEVICE, .DEFINE)
- kolor czcionki nierozpoznanego tekstu
- czcionka (rodzaj, styl, rozmiar, skrypt)

W każdej chwili możemy także przywrócić ustawienia domyślne.

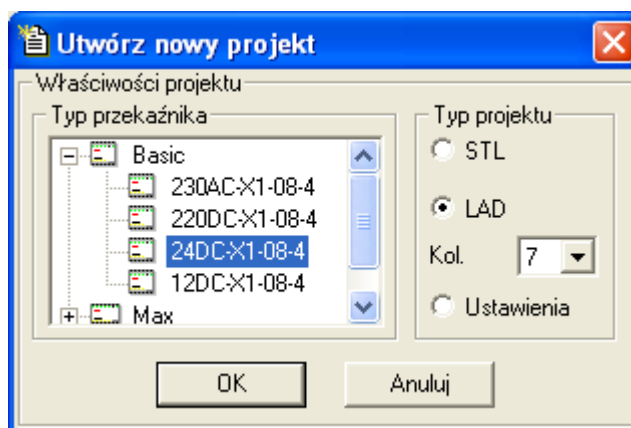


Rys. 6.7.3. Konfiguracja edytora STL.

6.8. Edytor programu LAD

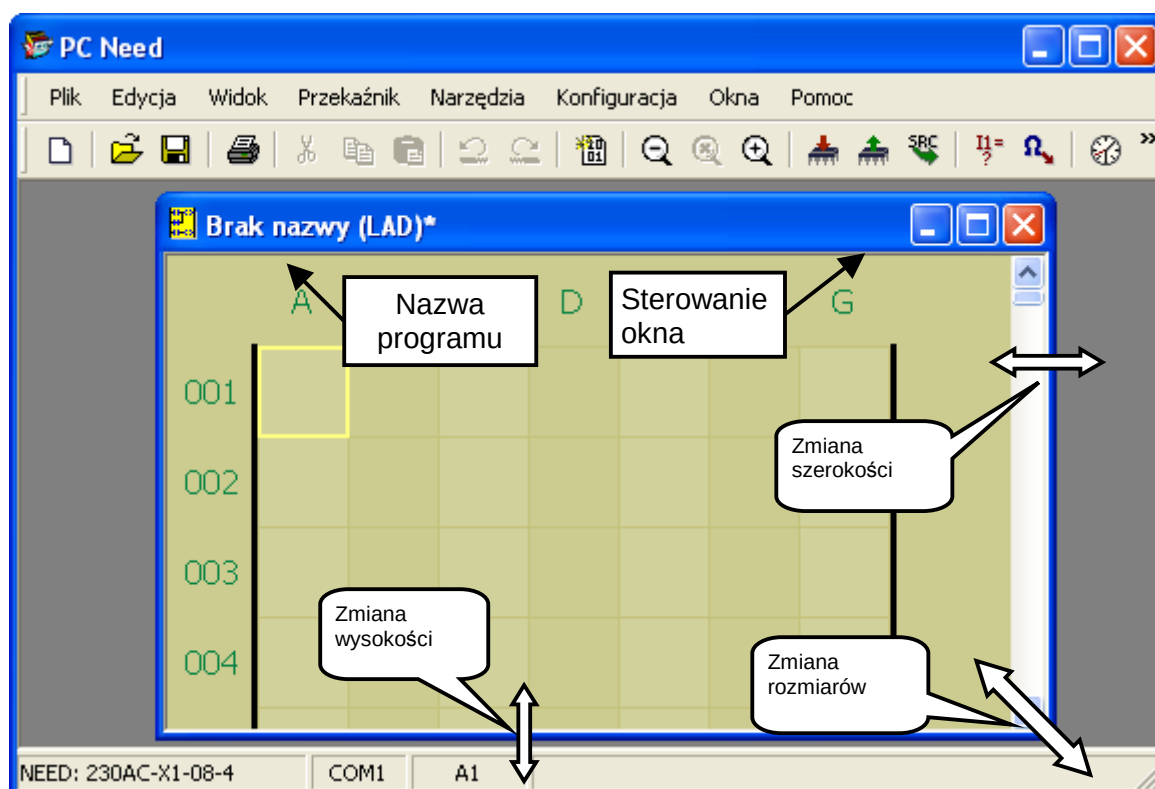
6.8.1. Nowy program

Po uruchomieniu programu PC Need, aby utworzyć program w języku LAD, wybieramy Menu **Pliki > Nowy > LAD**



Rys. 6.8.1.1. Nowy projekt LAD.

Wybieramy typ przekaźnika, typ projektu LAD oraz określamy ilość kolumn drabinki LAD. Wybór zatwierdzamy klawiszem *Enter* lub lewym przyciskiem myszy. W obszarze roboczym utworzy się okno **edytora LAD**.

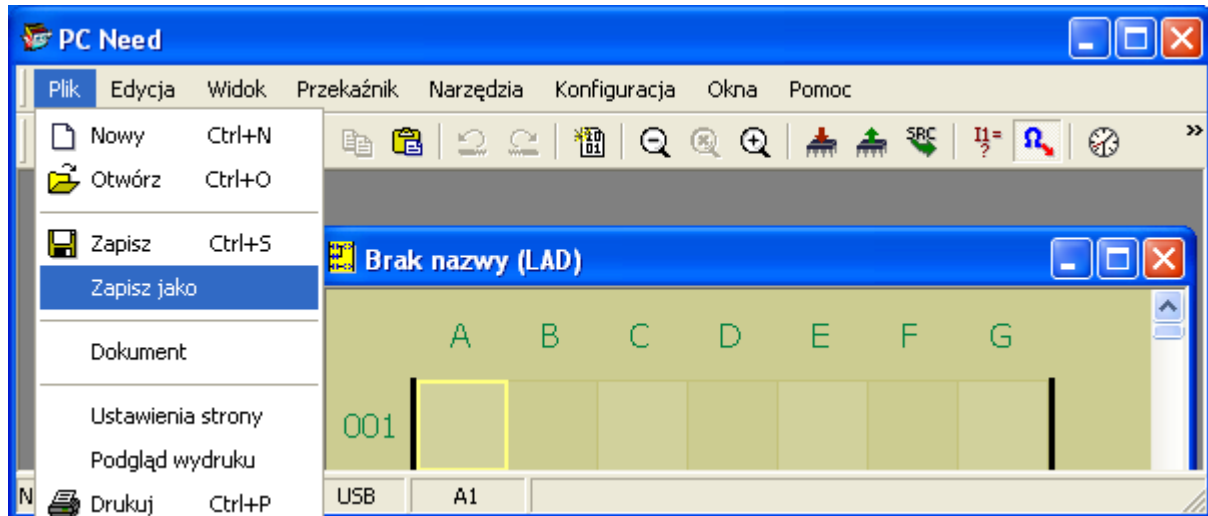


Rys 6.8.1.2. Edytor Lad – nowy program.

Za pomocą myszki można dostosować rozmiar okna do własnych potrzeb i preferencji. Korzystając ze standardowych przycisków sterujących oknami  można okno edytora LAD zmaksymalizować, zminimalizować, lub zamknąć (x).

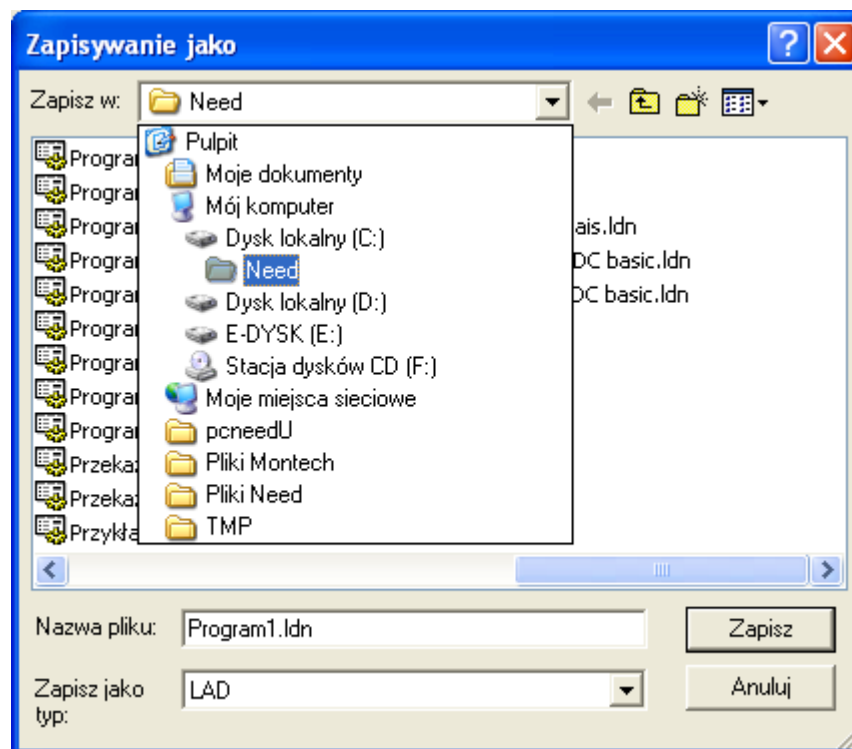
6.8.2. Zapisanie programu

Ponieważ nowo otwarty program nie ma nazwy (*Brak nazwy (LAD)*) należy go zapisać nadając mu odpowiednią nazwę. W tym celu w Menu Pliki wybieramy opcję **Zapisz jako**:



Rys. 6.8.2.1. Okno Zapisz jako.

Po zatwierdzeniu lewym klawiszem myszki otwiera się okno **Zapisz plik**:



Rys. 6.8.2.2. Okno Zapisz plik LAD.

Wybieramy położenie pliku (ścieżka dostępu) – w tym wypadku:

Zapisz w: *Need*;

Nazwa pliku: wpisujemy np. *Program1*;

Zapisz jako typ: *LAD* (*domyślne rozszerzenie nazwy pliku – .lbn*)

i zatwierdzamy operację klawiszem **Zapisz**

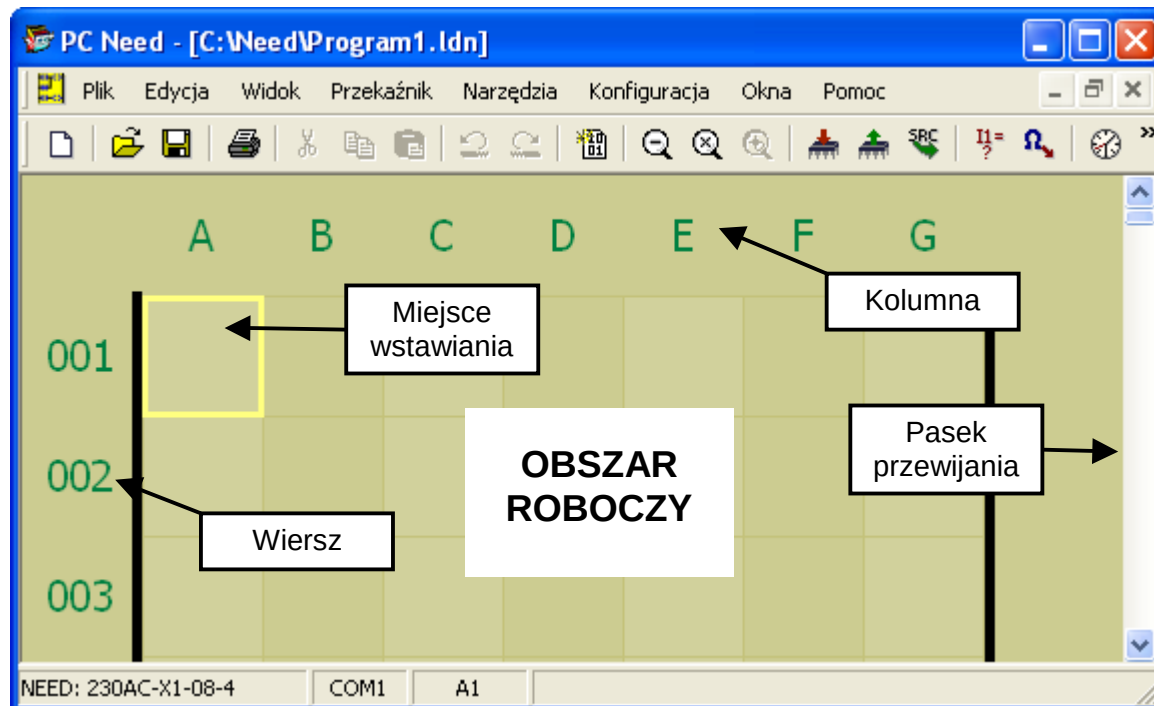
6.8.3. Otwarcie istniejącego programu

Jeśli chcemy otworzyć już istniejący dokument, to przy pomocy Menu **Pliki** i wybraniu **Otwórz** – otwiera się standardowe okno menadżera plików jak powyżej, w którym wybieramy plik z rozszerzeniem „*.ldn”.

Podobny efekt uzyskamy wykorzystując przycisk  z paska narzędzi lub skrót klawiaturowy **Ctrl+O**.

6.8.4. Wprowadzanie i edycja schematu połączeń

W Menu **Pliki** tworzymy nowy program LAD (patrz p. 6.8.1.) i nadajemy mu nazwę np. *Program1.ldn* (patrz p. 6.8.2). Dostosowujemy rozmiar okna i otrzymujemy okno edytora LAD:



Rys. 6.8.4.1. Okno edytora LAD.

Obszar roboczy, to siatka oparta na kwadratach, których położenie określa litera kolumny **A, C, E..** oraz numer wiersz **001..150**.

Kolumny **A, C, E..** są przeznaczone do wstawiania elementów wejściowych programu (Wejścia fizyczne, stan wyjść, *Znaczników, Timerów, Liczników, Zegarów, Komparatorów*) lub połączeń.

Kolumny **B, D, F..** są przeznaczone do wstawiania połączeń pomiędzy elementami. Ostatnia kolumna, to miejsce wstawiania elementów wyjściowych (fizyczne wyjścia, *Znaczniki Timery, Liczniki*).

Rysowanie schematu połączeń

Wewnątrz okna edytora LAD za pomocą myszki przemieszczamy się po kwadratowych polach siatki, aktualnie wybrane pole jest zaznaczone szarą obwódką. Dodatkowo na pasku stanu, w lewym dolnym rogu, podawane są współrzędne pola (wiersz, kolumna).

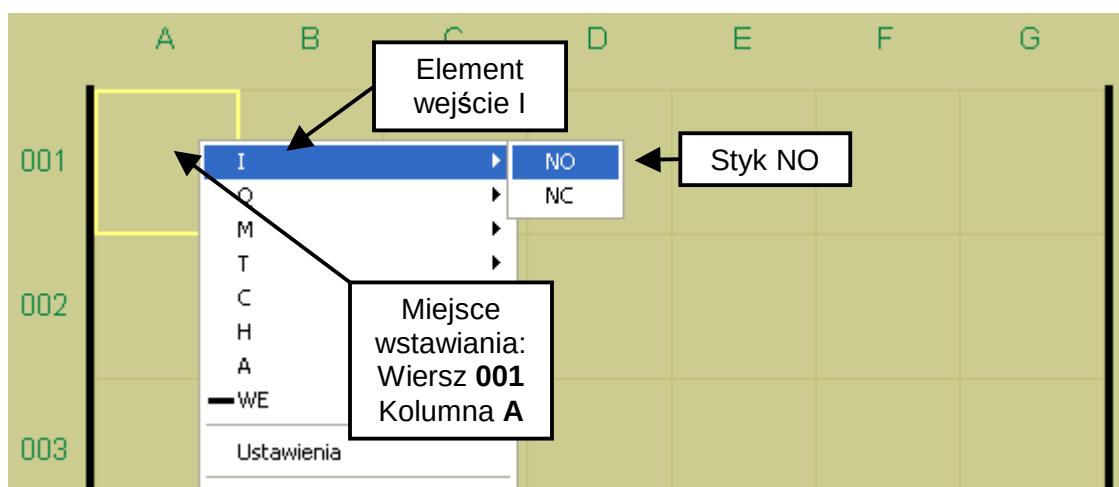
Pole wybieramy ustawiając nad nim kursor i klikając lewym klawiszem myszki.

Po wybraniu pola (zaznaczeniu), za pomocą prawego klawisza myszki, wybieramy z rozwijanego menu element lub połączenie (w zależności od bieżącej kolumny).

Stosowane symbole są zgodne z opisem języka LAD (rozdział 5.2.).

Wstawianie elementu wejściowego:

Klikając prawym klawiszem w kolumnie A, C lub E (poniżej jest to pole 001 A) wywołujemy rozwijane menu jak na rys. 6.8.4.2.

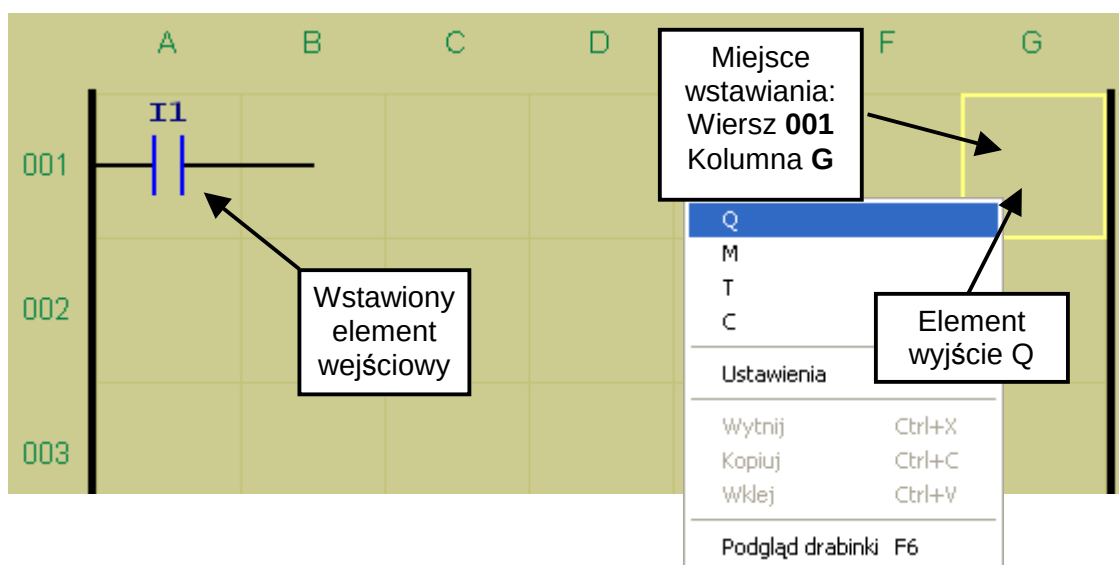


Rys. 6.8.4.2. Wstawianie elementu wejściowego.

Zatwierdzamy wybór lewym przyciskiem myszki lub klawiszem Enter.

Wstawianie elementu wyjściowego

Klikając prawym klawiszem w kolumnie G (na rys. poniżej jest to pole 001 G) wywołujemy rozwijane menu jak na rys 6.8.4.3.



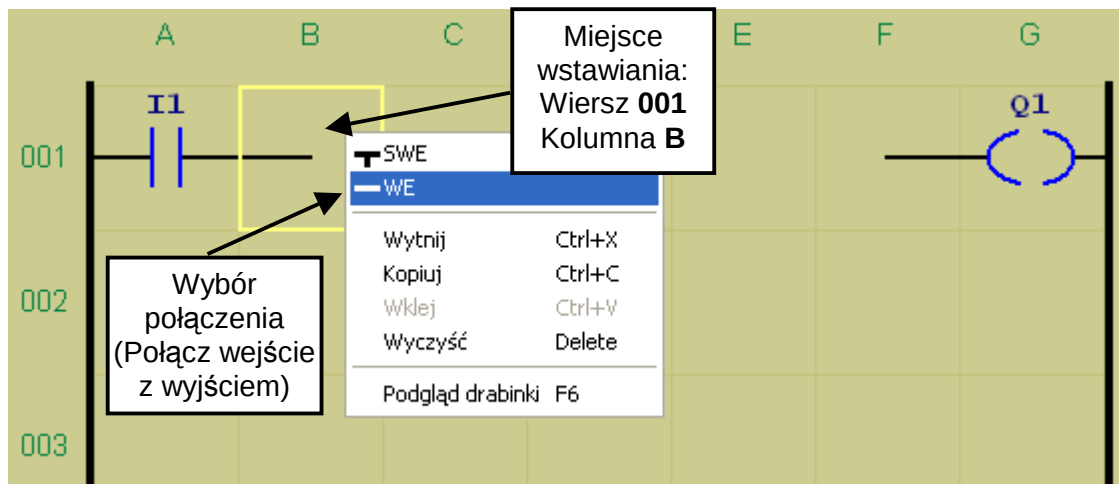
Rys. 6.8.4.3. Wstawianie elementu wyjściowego.

Zatwierdzamy wybór lewym przyciskiem myszki lub klawiszem klawiszem *Enter*.

Usuwanie elementu

Możliwe jest także usunięcie elementu – po wybraniu (zaznaczeniu) pola w którym znajduje się element, z rozwijanego menu (prawym klawiszem myszy) wybieramy (podświetlamy)

Wyczyść – po zatwierdzeniu lewym klawiszem element zostanie usunięty. Alternatywnie działanie uzyskamy używając klawisza Delete z klawiatury.

Wstawianie połączenia:

Rys. 6.8.4.4. Wstawianie połączenia.

Po wybraniu pola połączeń i naciśnięciu prawego klawisza myszy wyświetlane są, w rozwijanym menu, aktualnie możliwe do wykorzystania połączenia. Oprócz graficznego symbolu podany jest skrótowy opis kierunków będący kombinacją liter S, W, N, E.

S – South (dół)

W – West (lewo)

N – North (góra)

E – East (prawo)

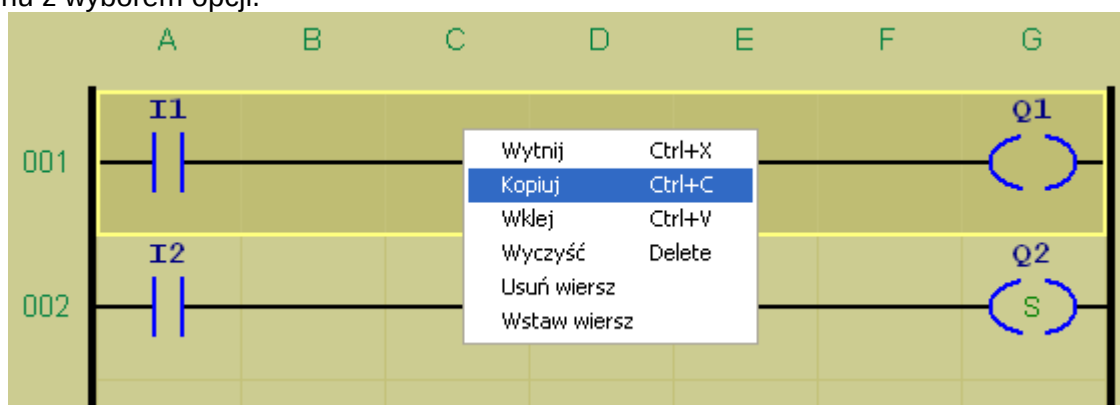
Możliwe jest także usunięcie połączenia – po wybraniu (zaznaczeniu) pola, w którym istnieje połączenie, z rozwijanego menu, (prawym klawiszem myszy) wybieramy (podświetlamy)

Wyczyść – po zatwierdzeniu lewym klawiszem, połączenie zostanie usunięte.

Edycja obszaru

Program PC Need umożliwia edycję projektu LAD poprzez operacje usuwania, przenoszenia, kopiowania zaznaczonego obszaru. Obszar do kopiowania jest prostokątem opartym na kwadratach siatki połączeń.

Zaznaczenie wykonujemy lewym klawiszem myszki, prawym klawiszem otwieramy rozwijane menu z wyborem opcji.

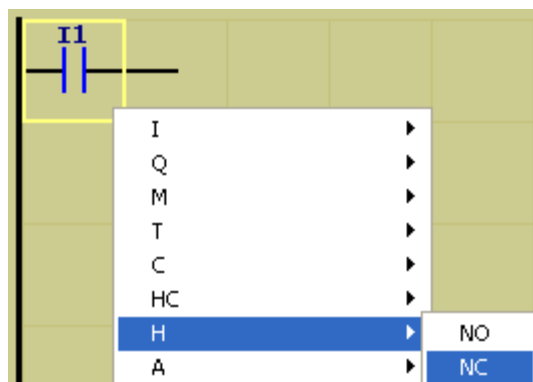


Rys. 6.8.4.5. Zaznaczenie wiersza i wybór opcji.

- **Wytnij** - Zaznaczony (ciemniejsze tło) wiersz lub obszar możemy przemieścić w inne miejsce korzystając z opcji *Wytnij*, a następnie wskazując początkowy lewy kwadrat siatki miejsca przeznaczenia i operację *Wklej*.
- **Kopiuj** - Zaznaczony obszar możemy skopiować w inne miejsce korzystając z opcji *Kopiuj*, a następnie wskazując początkowy lewy kwadrat siatki miejsca przeznaczenia i operację *Wklej*.

- **Wyczyść** -możemy wyczyścić zaznaczony obszar – otrzymamy czystą siatkę
- **Usuń wiersz** – zaznaczony wiersz zostanie usunięty
- **Wstaw wiersz** – w zaznaczone miejsce wstawiony zostanie czysty wiersz

Zmiana rodzaju wejścia.



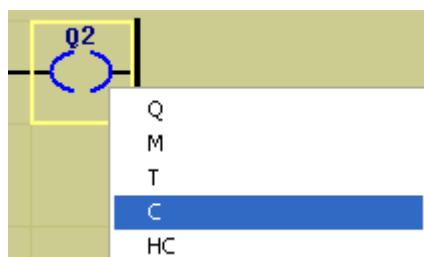
Klikamy lewym klawiszem na polu elementu, który chcemy zmienić (I1).

Prawym klawiszem rozwijamy menu, w którym wybieramy nowy rodzaj elementu (H) i typ styku (NO lub NC).

Zatwierdzamy wybór lewym klawiszem myszy.

Rys. 6.8.4.6. Zmiana rodzaju wejścia.

Zmiana rodzaju wyjścia.



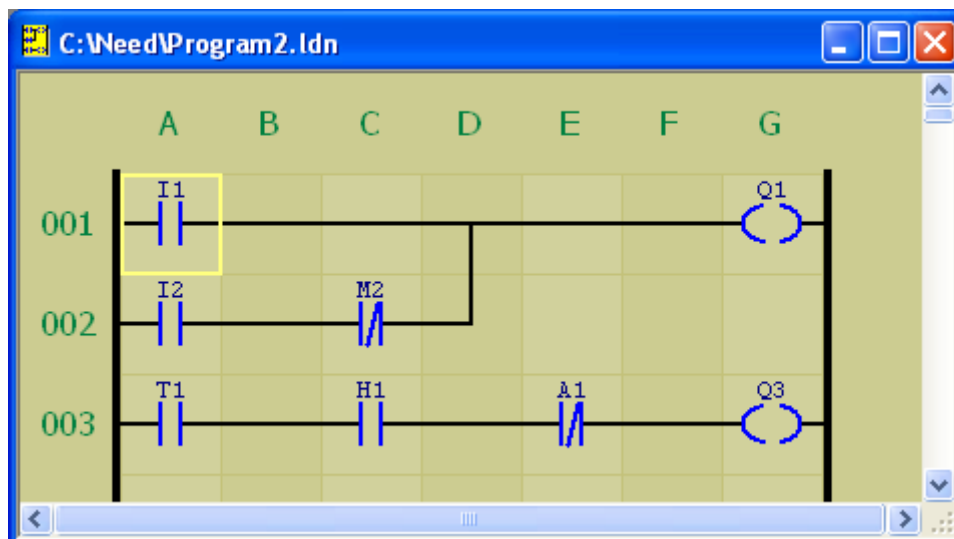
Klikamy lewym klawiszem na polu elementu, który chcemy zmienić.

Prawym klawiszem rozwijamy menu, w którym wybieramy nowy rodzaj (M).

Zatwierdzamy wybór lewym klawiszem myszy.

Rys. 6.8.4.7. Zmiana rodzaju wyjścia.

Postępując z powyższymi zasadami możemy utworzyć program jak na rys. 6.8.4.13. (poniżej)



Rys. 6.8.4.8. Program w LAD.

Z plikiem *Program2.lad* związane jest okno Ustawienia (*Program2.ladn*), które jest niezbędne do konfiguracji takich elementów jak **Timer**, **Zegar**, **Licznik**, **Komparator**, **Remanencja** oraz **Opóźnienia wejść**.

Wprowadzone w Ustawieniach wartości *Timerów* i *Liczników* są widoczne na schemacie LAD.

6.8.5. Edycja elementu

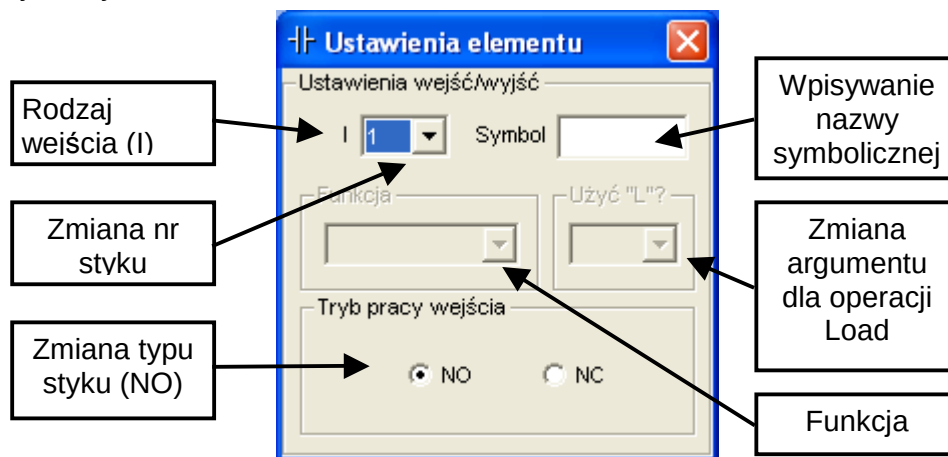
Każdy element umieszczony na schemacie (siatce połączeń) możemy poddać edycji: zmienić parametry, rodzaj i numer wejścia, wyjścia, typ styku.

Ustawienia elementu

Po dwukrotnym kliknięciu (lewym przyciskiem) na element umieszczony na schemacie wywołujemy okno *Ustawienia elementu*.

W zależności od rodzaju elementu dostępne są odpowiednie pola do zmiany.

Element wejściowy:



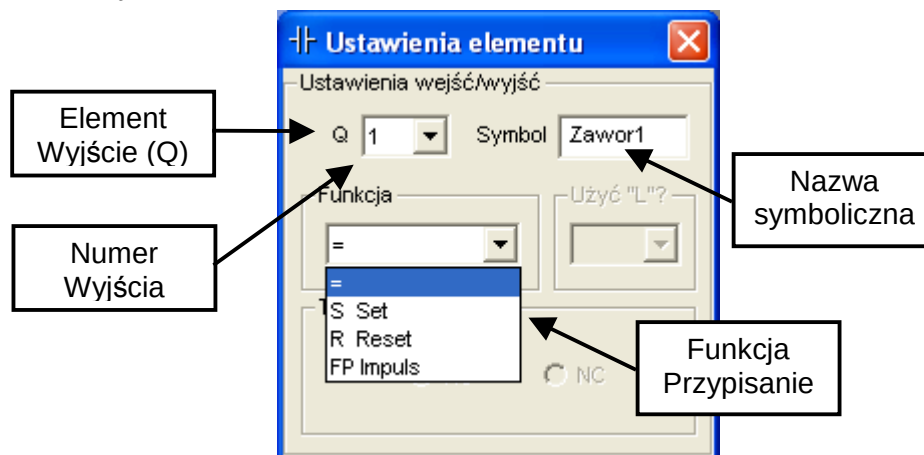
Rys. 6.8.5.1. Okno Ustawienia elementu.

Na rys. 6.8.5.1. przedstawiono *Ustawienia elementu* dla wejścia I (fizyczne wejście), podobnie wyglądają okna dla innych rodzajów wejść, A, H, Q, M, T, C. W zależności od rodzaju wejścia dostępny jest wybór numeru zależny od zasobów przekaźnika np. dla I jest to 1..8(16). Każde wejście ma wybierany typ styku NO lub NC. W polu Symbol możemy wprowadzić nazwę symboliczną dla danego elementu.

Dla wejść, okno elementu ma nieaktywne pole *Funkcja*, które jest używane tylko dla elementów wyjściowych.

Ustawienia elementu dla wyjścia Q i Znacznika (Markera) M, MT.

Możemy wybrać numer kolejny (np. 1..4(8) dla wyjścia) oraz przypisać funkcję. Dla wyjścia Q, Znacznika M i MT będą to operandy: =, S, R, FP. W polu Symbol możemy wprowadzić nazwę symboliczną.

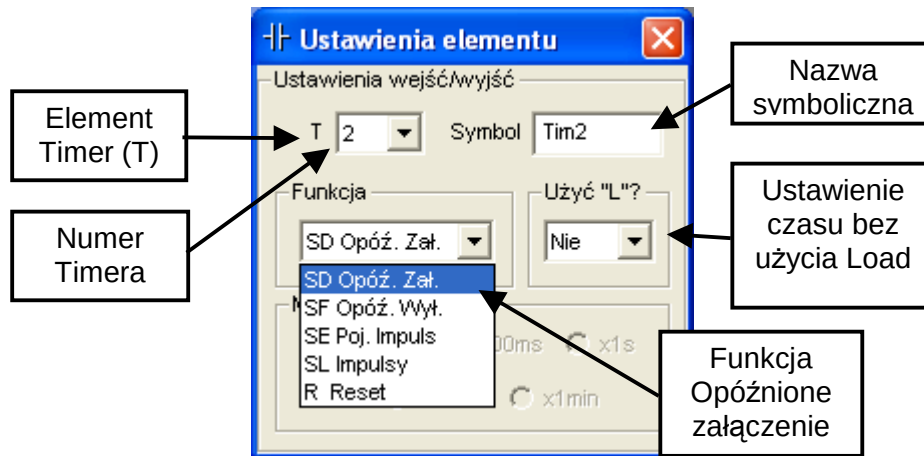


Rys. 6.8.5.2. Okno Ustawienia elementu Wyjście.

Dla wyjść okno *Ustawienia elementu* ma nieaktywne pole wyboru styku, natomiast aktywne jest pole *Funkcja*.

Ustawienia elementu dla wyjścia Timer.

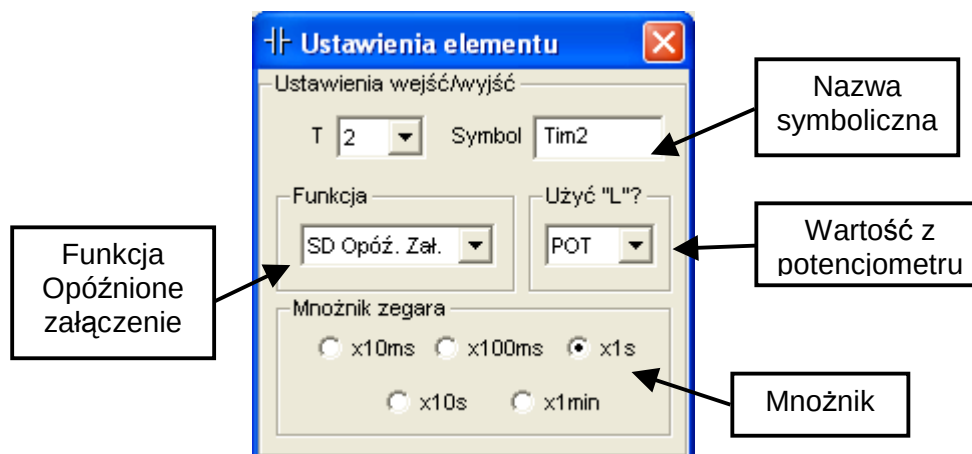
Dla wyjścia typu *Timer* (T) możemy wybrać parametry: numer 1..n oraz operandy: SD, SF, SE, SL, R. W polu Symbol możemy wprowadzić nazwę symboliczną dla danego *Timera*.



Rys. 6.8.5.3. Okno Ustawienia elementu Timer.

Po zaznaczeniu opcji *Użyć "L" – Pot*, *Timer* jako wartość do zliczania, bierze mnożnik zegara i wartość ustawioną za pomocą Potencjometru. W przykładzie poniżej dla mnożnika *x1s* wartość do zliczenia może być ustawiona w zakresie 1s .. 255s ((1-255) x 1s). Po zaznaczeniu opcji *Użyć "L" – Aln*, *Timer* jako wartość do zliczania, bierze mnożnik zegara i wartość odczytaną z wejścia analogowego In. Więcej na temat zastosowania użycia wartości analogowych do odmierzania czasów dla *Timerów* można znaleźć w rozdziale 5.1.2.21. Instrukcja ładowania (LOAD).

Po zaznaczeniu opcji *Użyć "L" – Stała*, ukaże się pole Czas w którym wprowadzamy wartość do zliczania dla *Timera*.



Rys. 6.8.5.4. Okno Ustawienia elementu Timer POT.

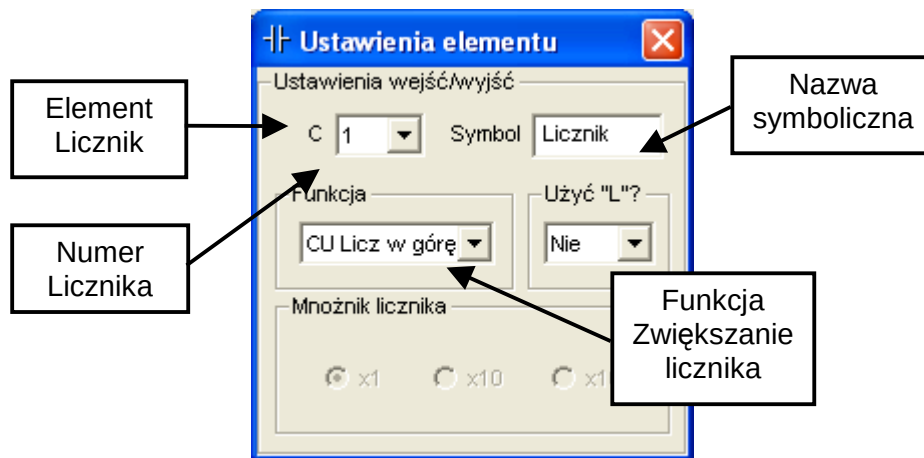


Uwaga: Zmiana wartości czasu odbywa się w oknie – *Ustawienia*.

Aln – to np. AI7, AI8 dla NEED-...-08-... i AI14, AI15, AI16 dla NEED-...-16-...

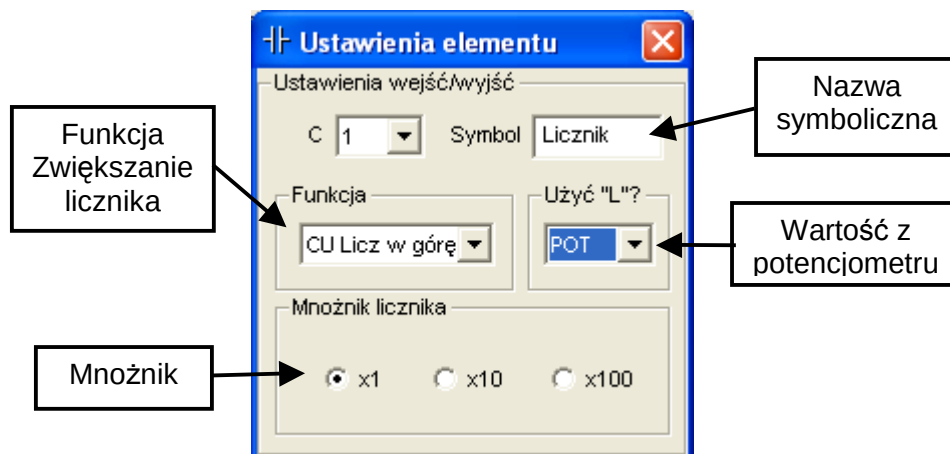
Ustawienia elementu dla wyjścia Licznik

Dla wyjścia typu *Licznik* (C, HC) możemy wybrać parametry: numer 1..8 oraz operandy: CU, CD, R. W polu Symbol możemy wprowadzić nazwę symboliczną dla danego *Licznika*.



Rys. 6.8.5.5. Okno Ustawienia elementu Licznik.

Po zaznaczeniu opcji *Użyć "L" – Pot*, *Licznik* jako wartość do zliczania, bierze mnożnik licznika i wartość ustawioną za pomocą Potencjometru. W przykładzie poniżej dla mnożnika x1 wartość do zliczenia może być ustawiona w zakresie 1..255 ((1-255) x 1).



Rys. 6.8.5.6. Okno Ustawienia elementu Licznik POT.

Po zaznaczeniu opcji *Użyć "L" – Ain*, *Licznik* jako wartość do zliczania, bierze mnożnik zegara i wartość odczytaną z wejścia analogowego In . Wartości czytane z wejść analogowych przyjmowane są podobnie jak odczyt Potencjometru w zakresie 1..255. Ain – to odpowiednio AI7, AI8 dla NEED..DC-xx-08-.. i AI14, AI15, AI16 dla NEED..DC-xx-16-...

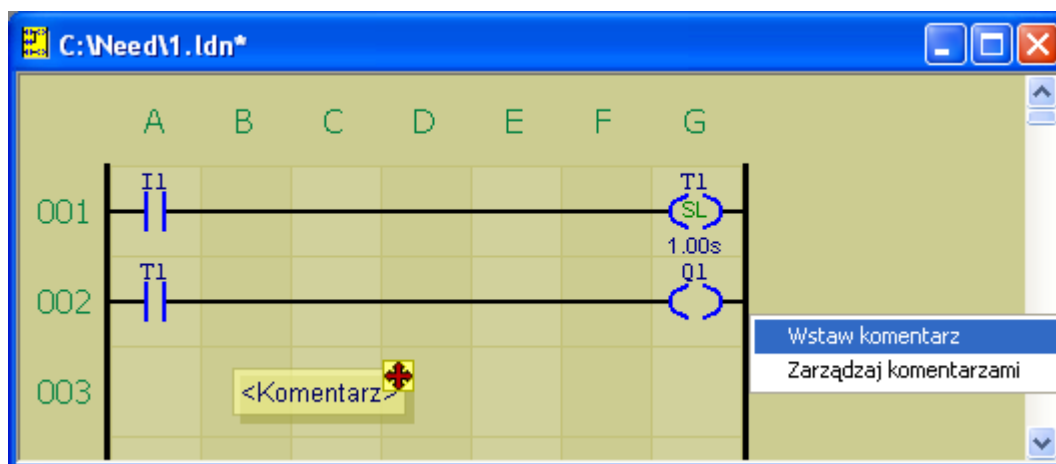
Po zaznaczeniu opcji *Użyć "L" – Stała*, ukaże się pole Wartość progowa w którym wprowadzamy stałą wartość do zliczania.



Uwaga: Zmiana wartości do zliczania odbywa się w oknie – *Ustawienia*.

6.8.6. Komentarze

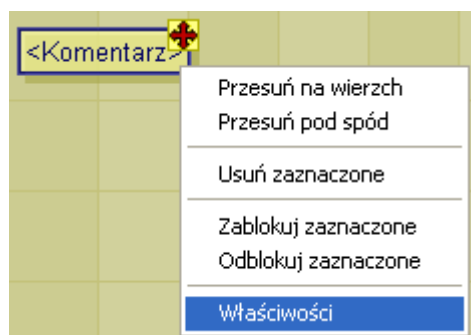
Edytowany program LAD można zaopatrzyć w komentarze tekstowe, zwiększające jego czytelność, ułatwiające późniejsze modyfikacje, lub inne informacje, np. opisy istotne dla sterowanego procesu.



Rys. 6.8.6.1. Wstawianie komentarza.

Komentarz wstawiamy z rozwijanego menu dostępnego pod prawym przyciskiem myszy. Opcja „Wstaw komentarz” umożliwia utworzenie nowego pola <Komentarz>, które możemy dowolnie umieścić na polu edytora LAD. Lewy przycisk umożliwia po 1-krotnym naciśnięciu zaznaczenie wybranego komentarza. Zaznaczony komentarz posiada wtedy identyfikującą go, widoczną ramkę. Wykonanie dwukrotnego kliknięcia powoduje otwarcie pola komentarza do edycji. Ten sam efekt umożliwia wybranie „Właściwości” z menu dostępnego pod prawym przyciskiem myszy. Dostępne tu są także inne opcje, sterujące zachowaniem pola komentarza:

- Przesuń na wierzch / na spód – położenie tekstu komentarza nad obiektami programu lub za obiektami programu.
- Usuń zaznaczone – usuwa zaznaczone komentarze
- Zablokuj / Odblokuj zaznaczone – możliwe lub niemożliwe przesuwanie komentarza
- Właściwości – otwiera okno edycji komentarza.

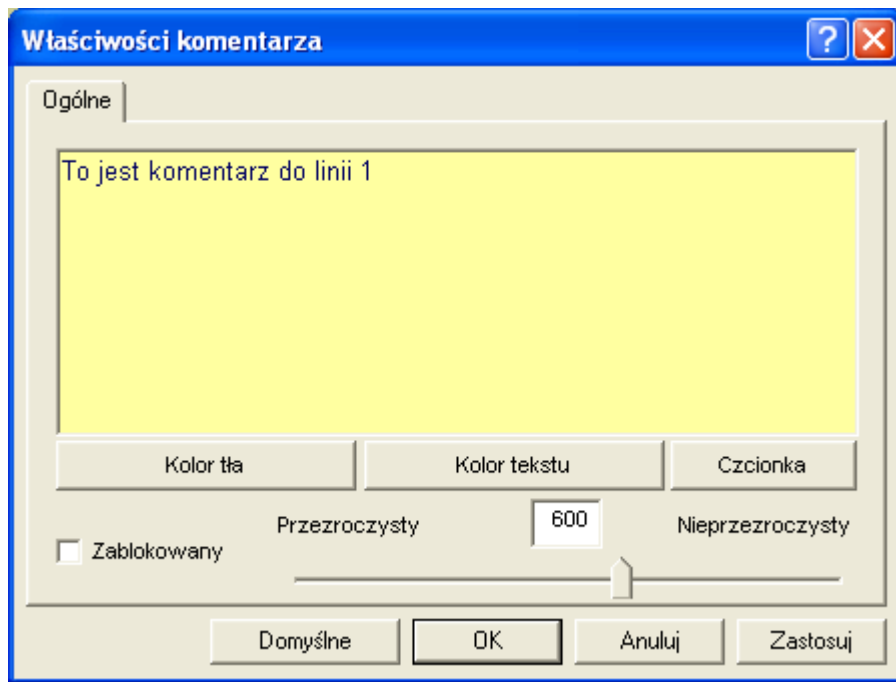


Rys. 6.8.6.2. Opcje pola komentarza.

Okno „Właściwości komentarza” umożliwia wprowadzenie tekstu komentarza, zmianę tła, koloru i czcionki tekstu. Wielkość wyświetlanego okna komentarza jest dostosowywana automatycznie do wprowadzonego tekstu. Dodatkowo możemy zablokować lub odblokować możliwość przemieszczania komentarza oraz ustawić przezroczystość pola komentarza względem okna edytora programu. W każdej chwili możemy przywrócić wartości domyślne edycji.

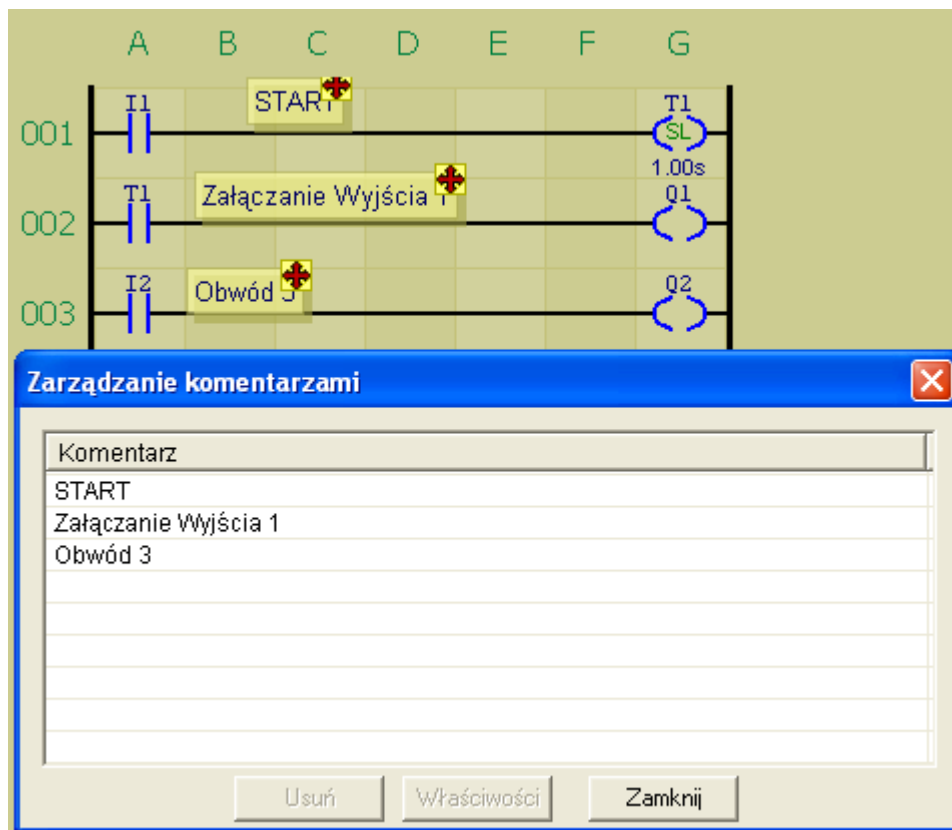


Uwaga: Po zamknięciu programu LAD komentarze są automatycznie zablokowane.



Rys. 6.8.6.3. Właściwości komentarza.

Opcja Zarządzaj komentarzami (Menu pod prawym klawiszem myszy) umożliwia podgląd wszystkich komentarzy w programie oraz usuwanie bądź edycję każdego z nich, po zaznaczeniu odpowiedniego wiersza i wybraniu opcji Usuń lub Właściwości.



Rys. 6.8.6.4. Zarządzanie komentarzami.



Uwaga: Po zaznaczeniu możliwe są operacje na większej ilości komentarzy.

6.8.7. Konfiguracja edytora LAD

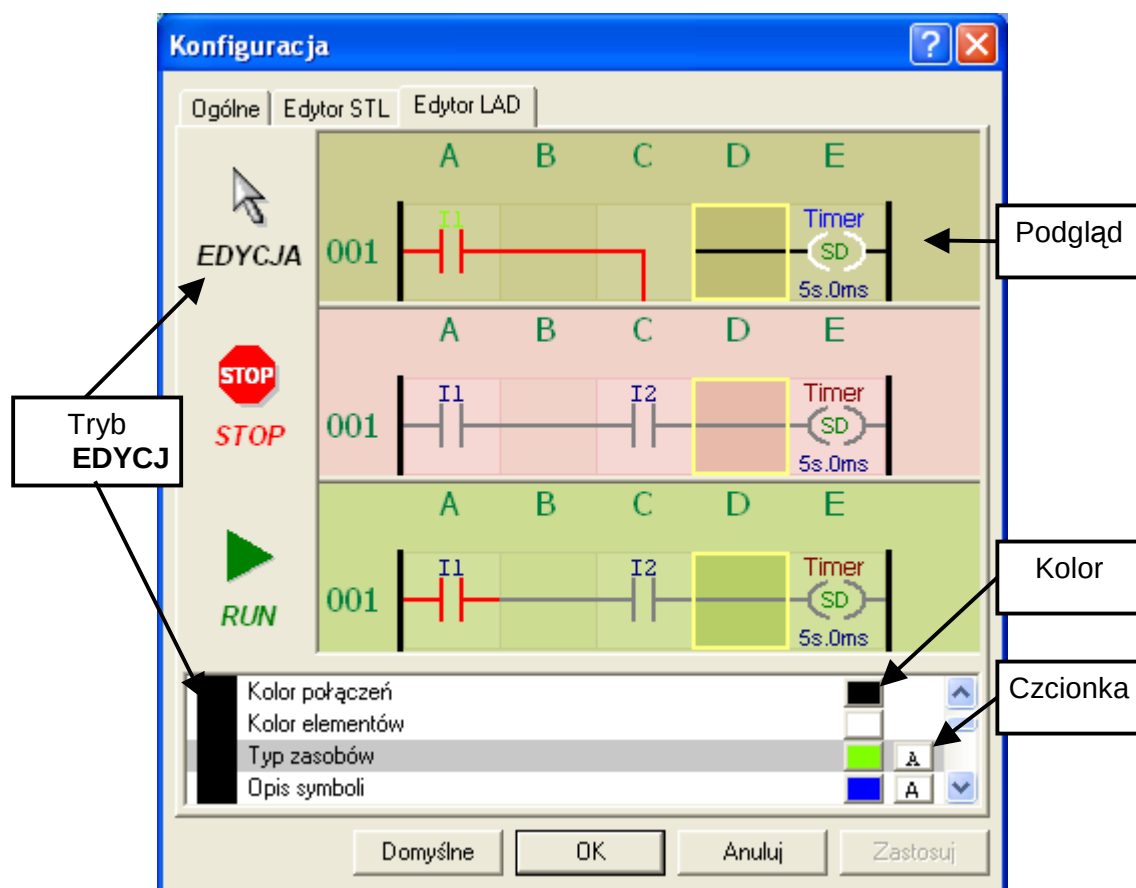
W Menu **Konfiguracja > Opcje** w zakładce *Edytor LAD* możemy dostosować wygląd okna edytora do własnych preferencji.

Do ustawienia niezależnie dla każdego z trybów EDYCJA, STOP, RUN są :

- kolor tła: okna i kolumn połączeń
- kolor siatki
- kolor tła kolumn styków
- kolor ramki kursora
- kolor tła zaznaczonego obszaru
- kolor połączeń
- kolor elementów
- kolor i czcionka typu zasobów
- kolor i czcionka nazw symbolicznych
- kolor i czcionka opisu funkcji
- kolor i czcionka opisu parametrów
- kolor i czcionka opisu kolumn i wierszy
- kolor błędnych połączeń i elementów

(czcionka - rodzaj, styl, rozmiar, skrypt).

W każdej chwili możemy także przywrócić ustawienia domyślne.



Rys. 6.8.7. Konfiguracja edytora LAD.

6.9. Ustawienia

6.9.1. Rodzaje ustawień

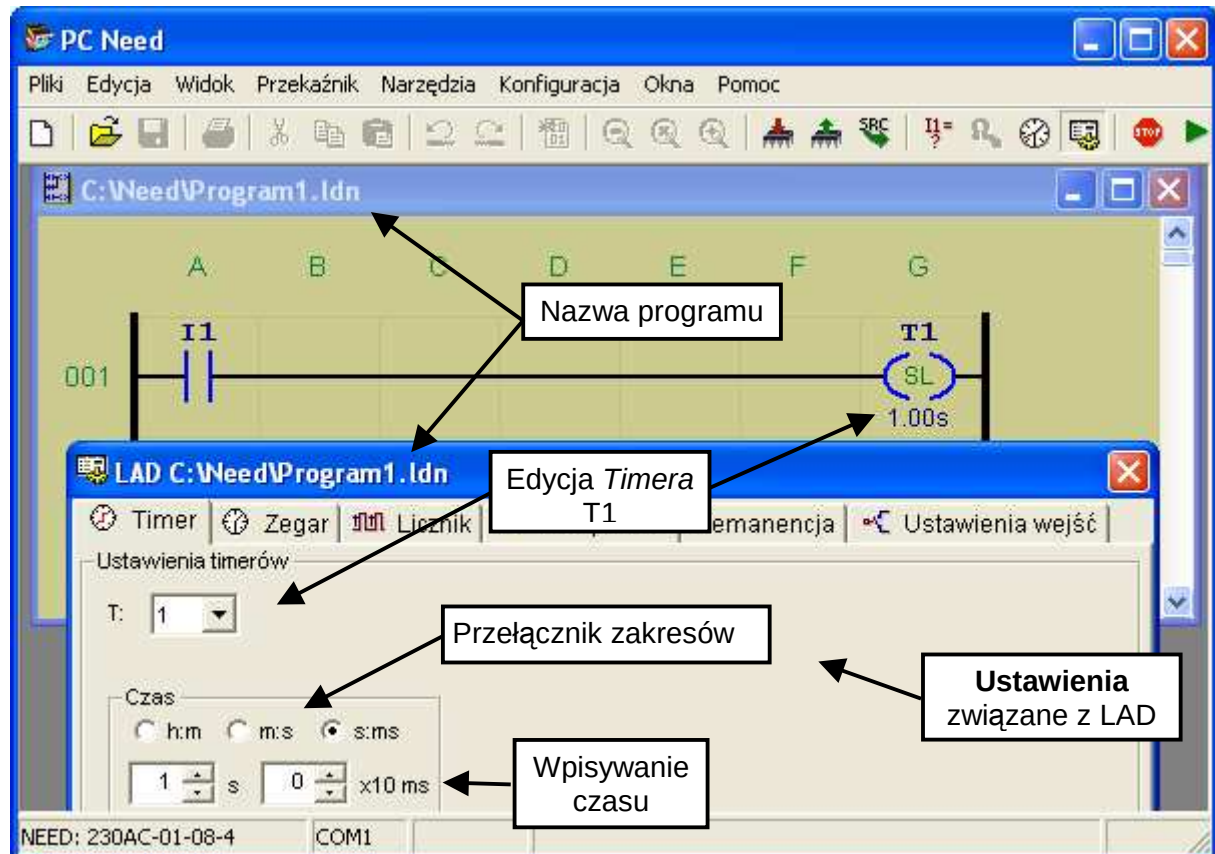
Wyróżniamy dwa typy ustawień:

1. Okno Ustawienia związane z programem LAD.
2. Plik Ustawienia niezależny.

W pierwszym przypadku dla aktywnego okna programu LAD tworzone są ustawienia przechowujące dane dla tego programu.

W drugim przypadku możemy utworzyć samodzielny plik (lub pliki) o różnych nazwach, które można niezależnie wgrać do przekaźnika, zastępując dotychczasowe ustawienia.

Ma to sens np., jeśli chcemy nie zmieniając programu zmienić czasy *Timerów*. Bez szukania w programie, tylko edytując plik ustawienia, możemy wprowadzić nowe wartości.



Rys. 6.9.1.1. Ustawienia związane z programem LAD.

Jeśli plik Ustawienia, związany z programem LAD, nie zostanie załadowany, to program w przekaźniku będzie wykonywany z ustawieniami, jakie ostatnio były w przekaźniku.

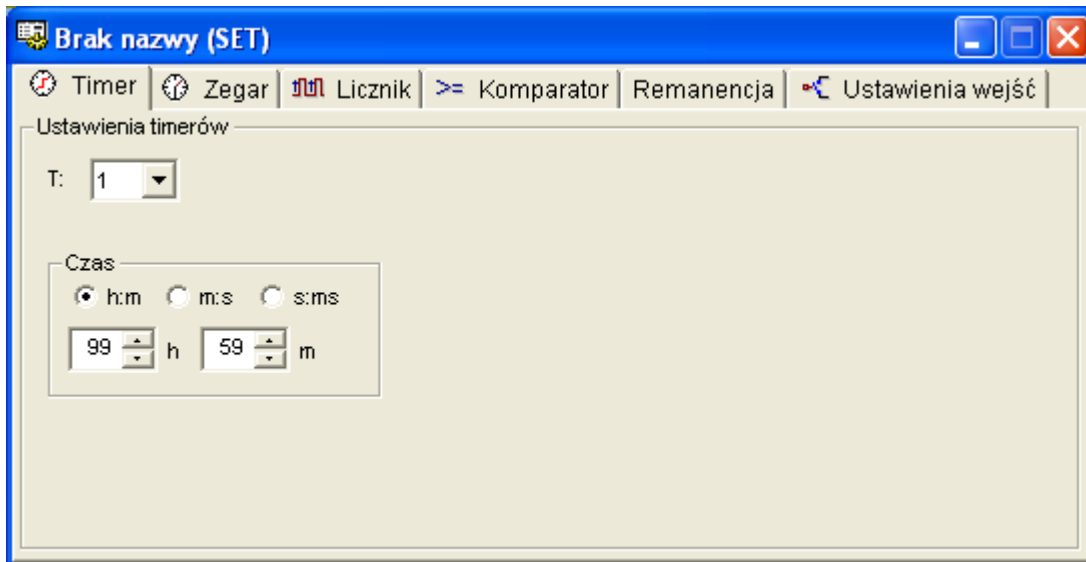
Po resecie będą to wartości maksymalne *Timerów* i *Liczników*, wyzerowane *Zegary*, ustawiony brak remanencji i załączone opóźnienia wejść.

Aby przejść do Ustawień w przypadku edycji programu LAD należy kliknąć ikonkę na pasku narzędzia lub alternatywnie wybrać w Menu: **Przekaźnik > Ustawienia** (ewentualnie klawisz skrót F10). Na rys. 6.9.1.2. przedstawiono efekt wykonania tego polecenia i edycji *Timera* 1 (tryb SE, czas 1s).

Aby utworzyć nowy plik SET należy w Menu **Plik** wybrać **Nowy** a następnie w oknie *Utwórz nowy projekt* zaznaczyć **Ustawienia** oraz wybrać wersję przekaźnika.

Ustawienia związane z programem LAD są zapisywane automatycznie przy zapisie tego programu, jeżeli opcja ta jest aktywna w konfiguracji projektu LAD.

Ustawienia utworzone samodzielnie należy zapisać podobnie, jak program LAD czy STL nadając im nazwę.



Rys. 6.9.1.2. Nowo utworzony plik ustawień (SET).

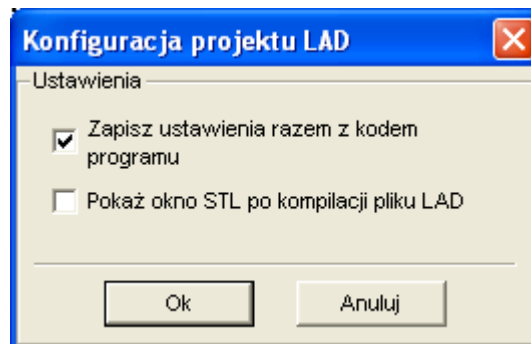
Domyślne rozszerzenie zapisywanego na dysku pliku Ustawienia to „*.set”.
Poniżej przedstawiono zasadnicze różnice dla ustawień związanych z programem LAD oraz dla niezależnego pliku *Ustawienia1.set*.



Uwaga: Plik Ustawienia jest ładowany niezależnie od programu. Domyślnie w programie NEED PC, dla edytora LAD, ładowany jest program a następnie ustawienia z nim związane.

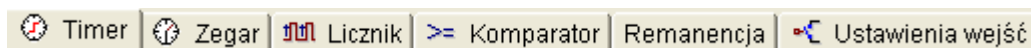
Automatyczne ładowanie ustawień dla projektu LAD można wyłączyć korzystając z opcji Menu:

Konfiguracja > Projekt LAD.



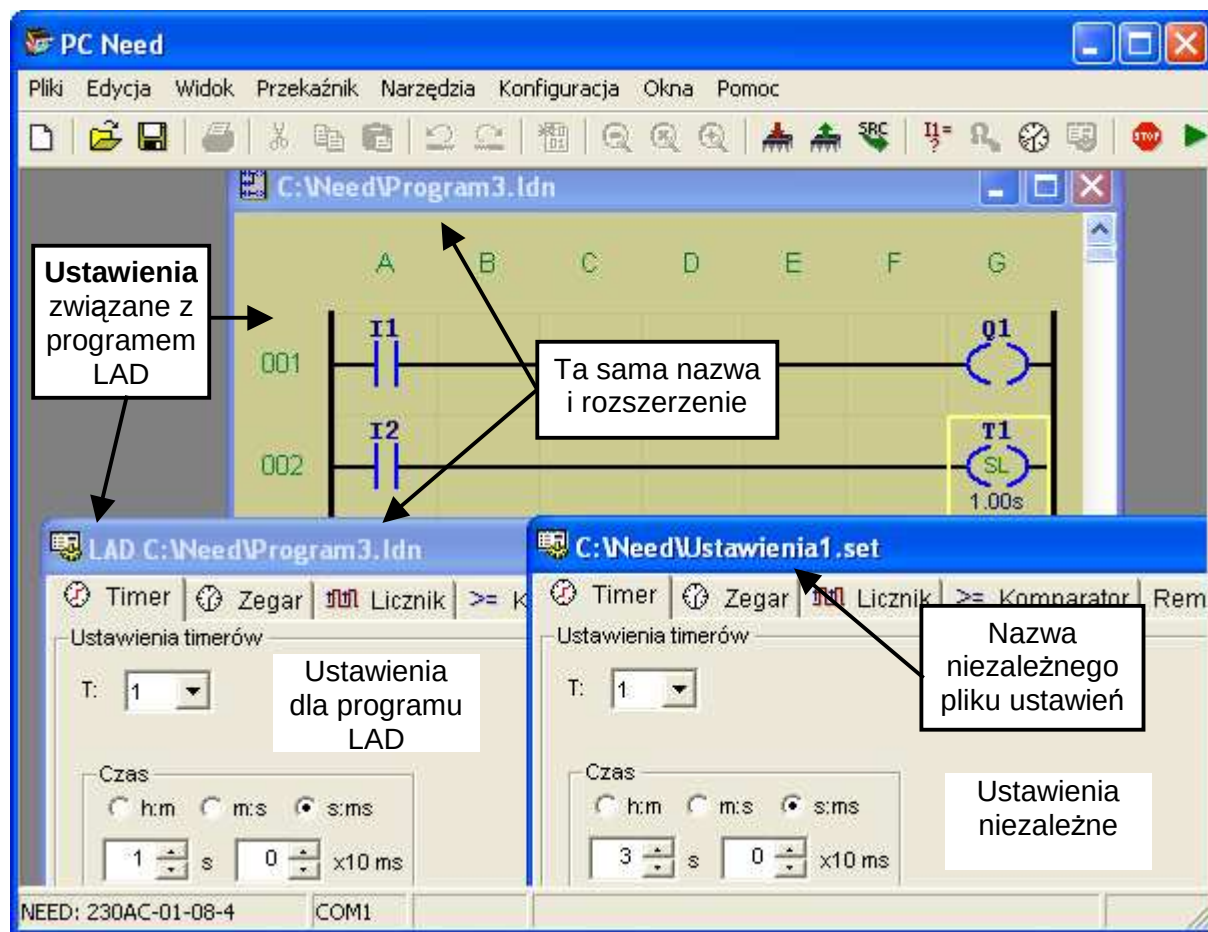
Rys. 6.9.1.3. Konfiguracja projektu LAD.

Posługując się zakładkami wybieramy typ zmiennych do ustawienia.



Okno *Ustawienia* skojarzone z programem LAD ma tylko możliwość zamknięcia natomiast plik SET ma jeszcze ikonizowanie

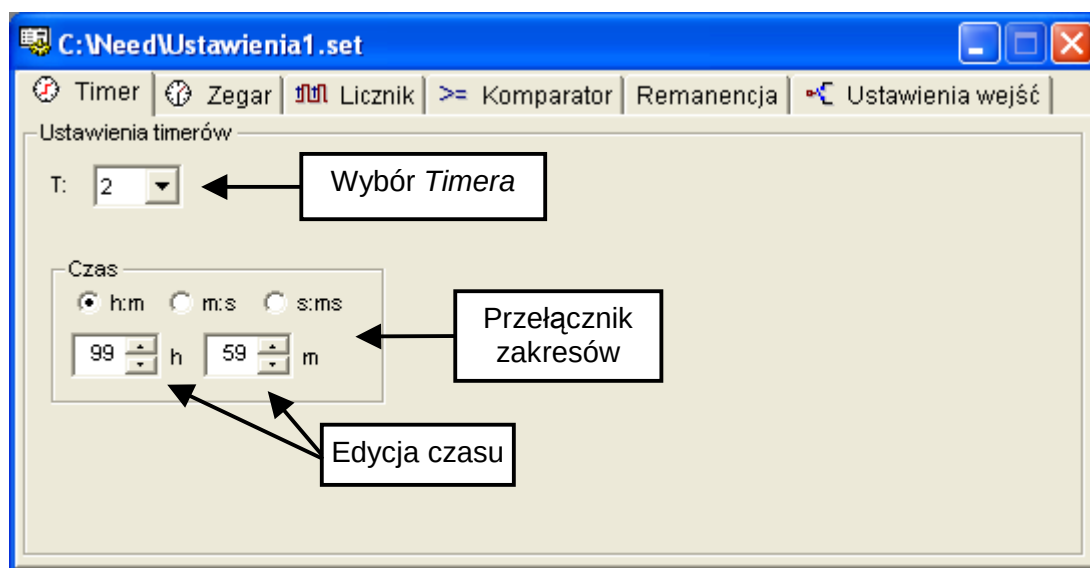
Poza nazwą i skojarzeniem edycja ustawień jest identyczna, polega na wypełnianiu opisanych pól lub wyborze wartości z rozwijanych menu.



Rys. 6.9.1.4. Różnice w ustawieniach.

6.9.2. Ustawienia Timerów

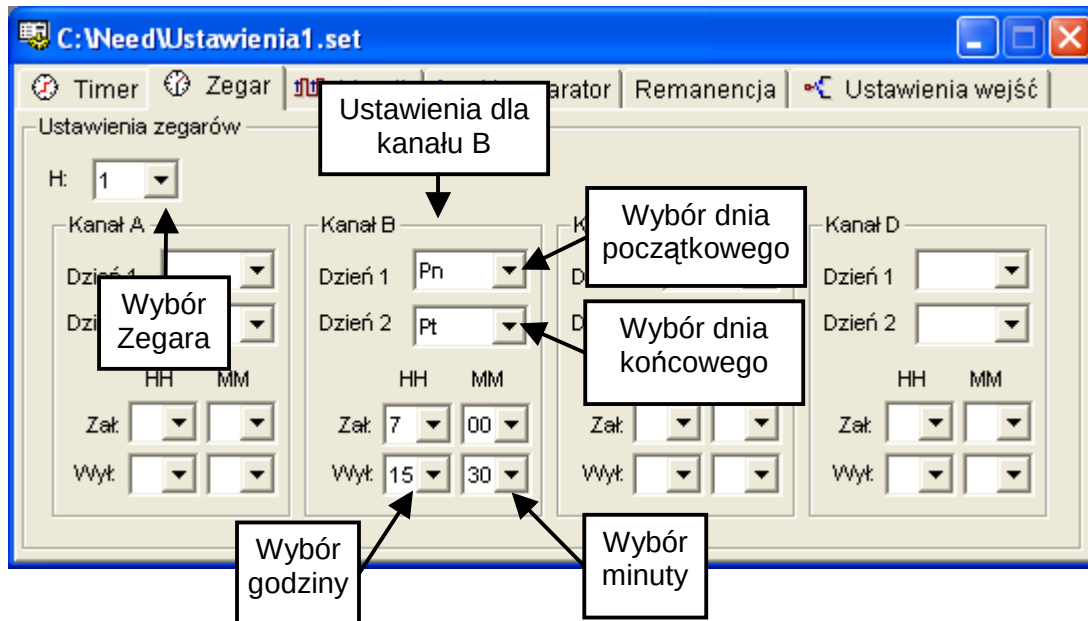
Wybieramy numer *Timera* a następnie przyporządkowujemy mu zakres (godz:minuty, minuty:sekundy, sekundy:milisekundy x 10). Potem wpisujemy zadaną wartość czasu w pola do edycji.



Rys. 6.9.2.1. Ustawienia Timerów.

6.9.3. Ustawienia Zegarów

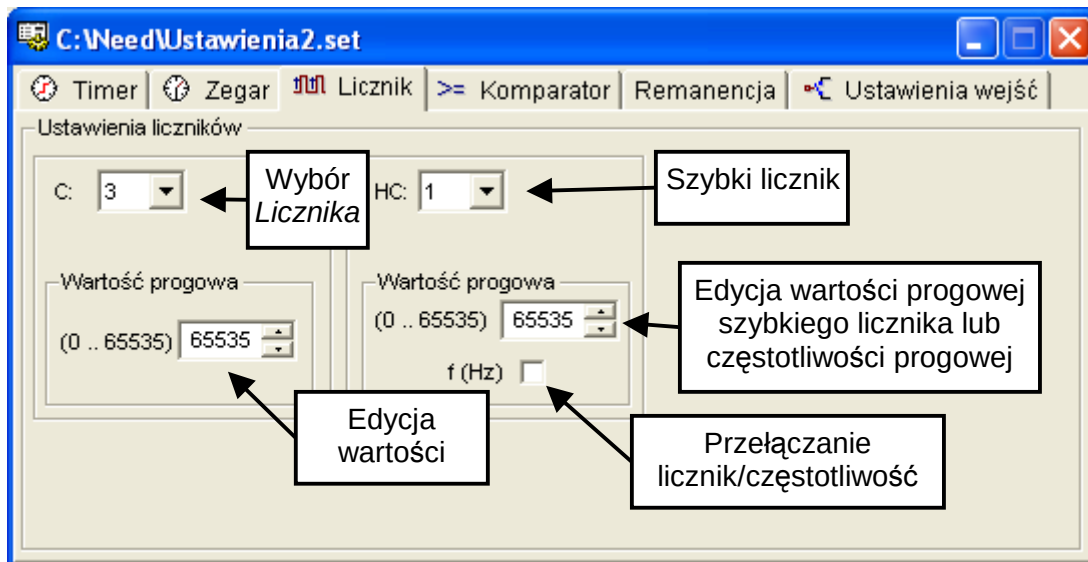
Wybieramy numer zegara (H1...H4) a następnie edytujemy wybrane kanały A..D ustawiając dni tygodnia, godziny i minuty.



Rys. 6.9.3.1. Ustawienia Zegarów.

6.9.4. Ustawienia Liczników

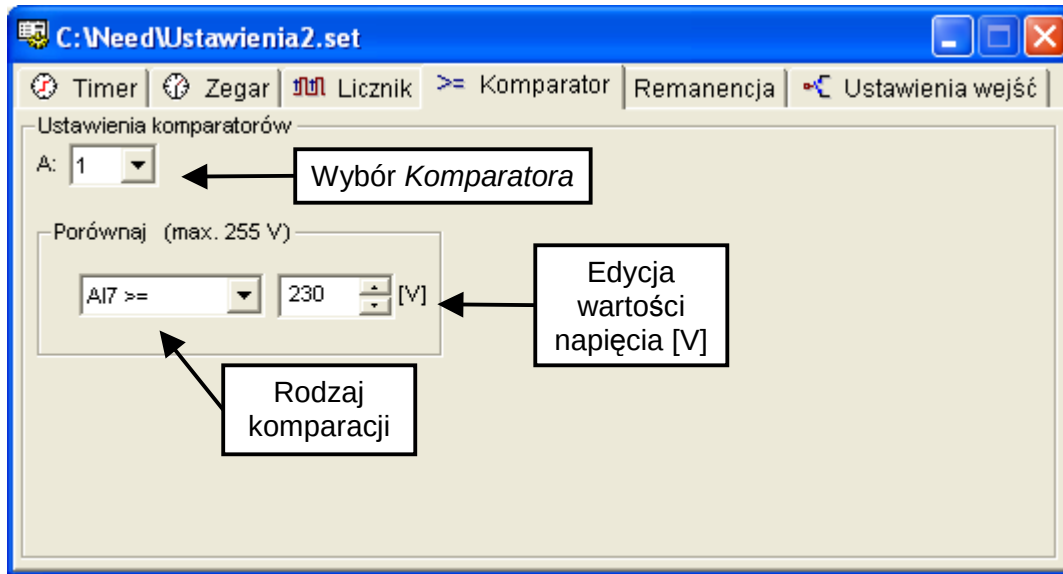
Wybieramy numer *Licznika*, a następnie wpisujemy wartość do zliczenia.



Rys. 6.9.4.1. Ustawienia Liczników.

6.9.5. Ustawienia Komparatorów

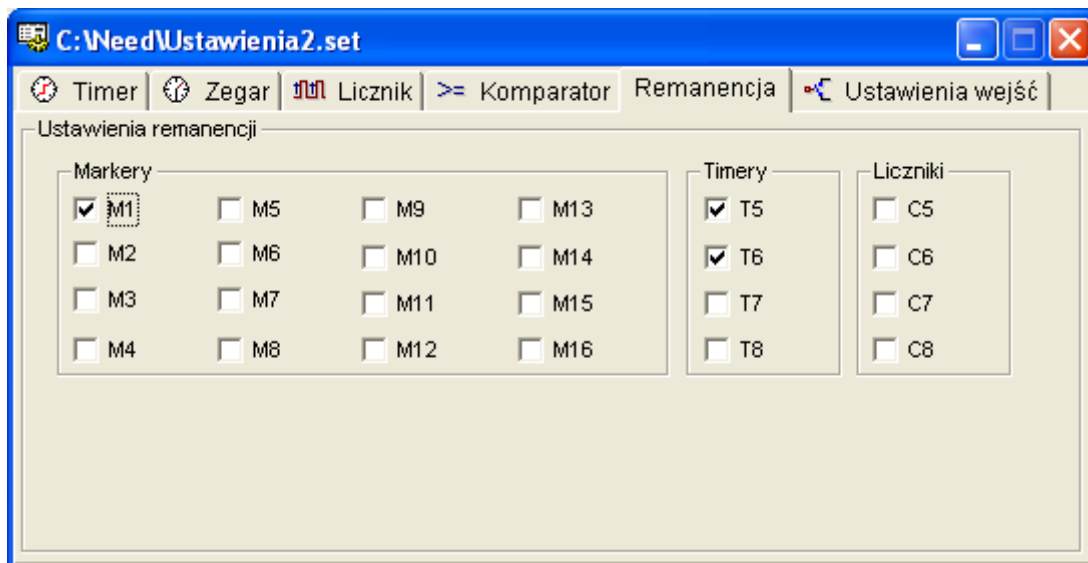
Wybieramy numer *Komparatora* i przypisujemy mu rodzaj porównania, a dla porównań z wartością stałą wpisujemy wartość napięcia w woltach.



Rys. 6.9.5.1. Ustawienia Komparatorów.

6.9.6. Remanencja

Wyboru zmiennej, która ma być remanetna dokonuje się poprzez zaznaczenie pola przy danej zmiennej (rys. 6.9.6.1 wybrano jako remanentne M1, T5 i T6).



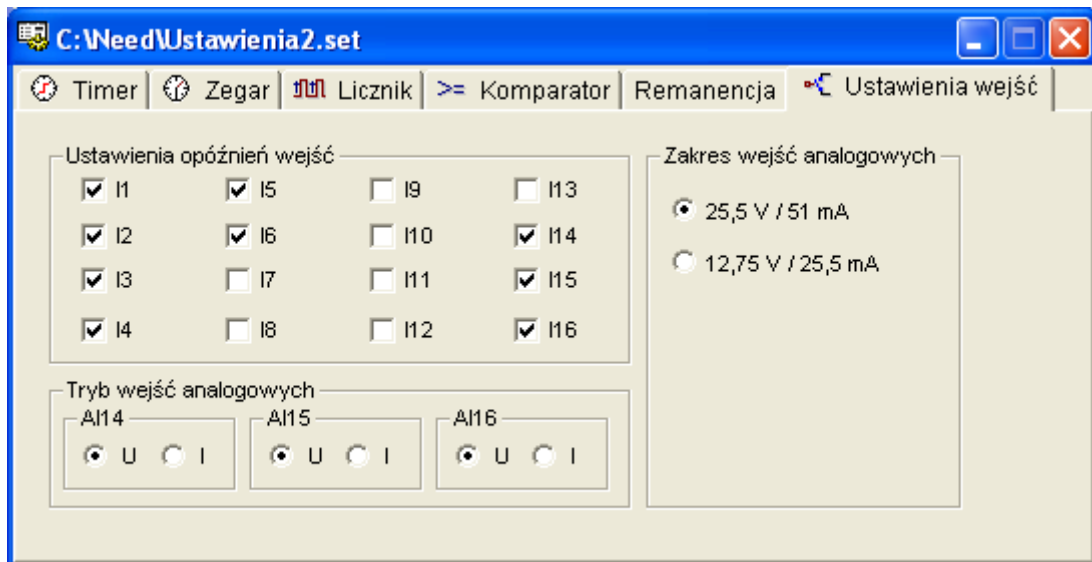
Rys. 6.9.6.1. Ustawienia Remanencji

6.9.7. Ustawienia wejść

Jeśli wejście ma być skanowane bez opóźnienia, to należy odznaczyć jego pole wyboru. (na rys. 6.9.7.1. wejścia od I7 do I13 nie będą opóźniane).

Domyślnie wejścia mają zaznaczone pola wyboru – jest ustawione opóźnienie wejść.

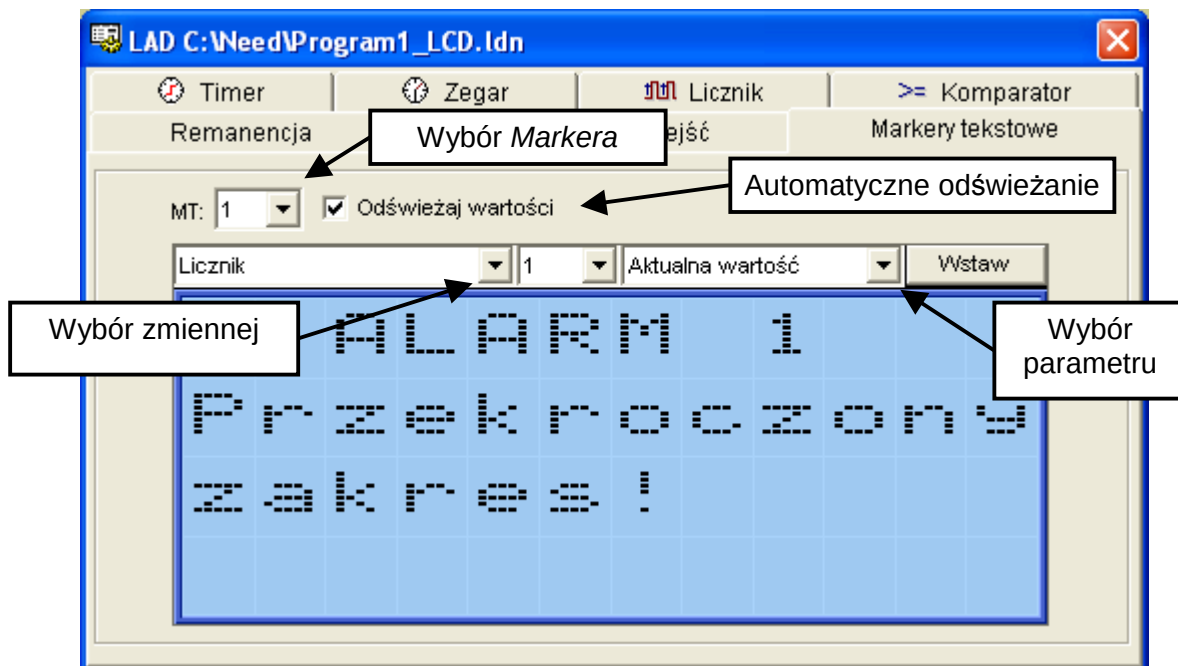
Dla wersji NEED ..DC-22-16-8 dodatkowe pola służą do konfiguracji zakresu i trybu wejść analogowych AI14..AI16.



Rys. 6.9.7.1. Ustawienia Opóźniania wejść

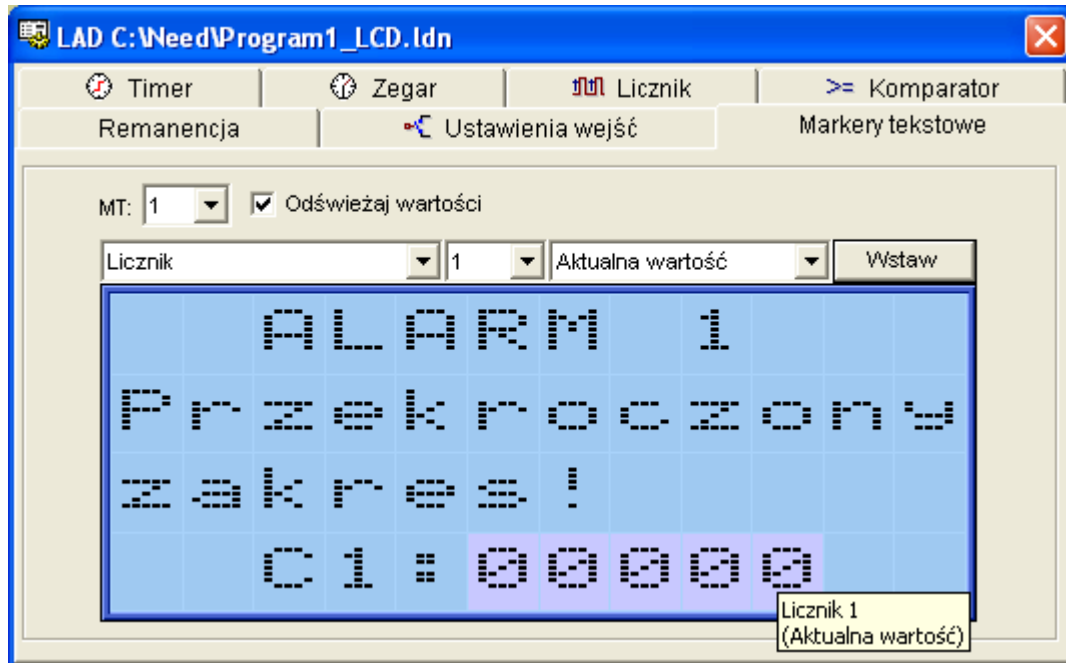
6.9.8. Markery tekstowe

Dla przekaźników z wyświetlaczem LCD formatka ustawień jest nieco inna, ma dodatkową zakładkę służącą do edycji tekstu informacji, jaka zostanie wywołana po ustawieniu odpowiadającego jej markera MTx.



Rys. 6.9.8.1. Ustawienia Markery tekstowe

Edytor Markerów tekstowych odzwierciedla pole wyświetlacza na przekaźniku typu Need Display. Umożliwia wstawianie znaków (tekst) oraz parametrów (zmienne).



Rys. 6.9.8.2. Wstawianie wartości licznika

Możliwe do wstawienia dla danego typu przekaźnika zmienne wybieramy z rozwijanej listy po lewej stronie. Na rys. 6.9.8.2. (wybrano Licznik). Dla każdej ze zmiennych określamy jaki parametr ma być prezentowany. Dla zmiennych binarnych (Wejścia, Wyjścia, Znaczniki) będzie to tylko Stan, dla innych takich jak np. Licznik można wybrać: Stan, Aktualna wartość, Ustawiona wartość).

Przełącznik „Odświeżaj wartości” (domyślnie włączony):

- ☒ - Podczas wyświetlania ekranu MTx wartości zmiennych są automatycznie odświeżane
- ☐ - Wartości zmiennych zostają pobrane z chwili aktywacji MTx i nie są później aktualizowane.

6.10. Podgląd zmiennych

Program PC Need został wyposażony w narzędzie do monitorowania wszystkich zmiennych w przekaźniku.

Okno Podgląd zmiennych wywołujemy poprzez Menu: **Narzędzia > Podgląd zmiennych** lub korzystając z przycisku . Skrót klawiaturowy F12.

Jeśli mamy połączenie z przekaźnikiem, to możemy korzystając z przycisku (Odczyt) lub Menu: **Przekaźnik > Transmisja > Odczyt z Przekaźnika** (skrót klawiaturowy F6) uruchomić ciągły podgląd wpisanych zmiennych.

Połączenie z przekaźnikiem jest sygnalizowane na pasku stanu:

- Dla aktywnego podglądu zmiennych i trybu RUN
- Dla aktywnego podglądu zmiennych i trybu STOP

Wyboru zmiennych podlegających podglądowi dokonuje się poprzez wpisanie ich w kolumnie *Zmienna*. Aby wpisać nową zmienną należy kliknąć dwukrotnie lewym klawiszem myszy lub nacisnąć *Enter* po ustawieniu kursora w wybranym wierszu, w kolumnie *Zmienna*. Korzystamy z mnemoniki, jakiej używamy do pisania programów oraz dodatkowo:

POT – Potencjometr

AI7, AI8 – wartości napięć na zaciskach wejściowych odpowiednio I7, I8.

AI14, AI15, AI16 – wartości napięć na zaciskach wejściowych odpowiednio I14, I15, I16.

RTC – Zegar Czasu Rzeczywistego – data i czas

RTC_D – tylko data

RTC_C – tylko czas

ASYM – asymetria faz

Zielony kolor podświetlenia w kolumnie *Zmienna* oraz nazwa portu komunikacyjnego na pasku stanu podświetlana na zielono oznacza pracę przekaźnika w trybie RUN.

Czerwony kolor podświetlenia sygnalizuje tryb STOP (zatrzymanie) przekaźnika.

W kolumnie *Stan* zmienne binarne o wartości „1” są podświetlane na zielono.

Zmienna	Stan	Aktualna	Ustawiona	Inne
I1	1			
Q1	0			
M1	0			
H1	0			
T1	0	0.47s	1.00s	SL
C1	0	0	65535	
A1	1			AI14 >= 0
POT		45		U/I
RTC		14/02/2007 9:54:51		
AI7				
AI14		0,0		U/I
AI15		0,0		U/I
I2	1			
T2	0	0h 00m	99h 59m	SL

Rys. 6.10.1. Podgląd zmiennych.

Wpisaną zmienną możemy skasować lub zastąpić inną.

W kolumnie *Stan* przedstawiany jest odczytana wartość 0 lub 1 dla zmiennych binarnych.

W kolumnie *Aktualna* przedstawione są bieżące (aktualne) wartości *Timerów* i *Liczników*

oraz wartość liczbowa dla POT i AI7, AI8, AI14, AI15, AI16.

Dla RTC jest podawana data i czas w formacie: dzień/miesiąc/rok godzina:minuta:sekunda.

W kolumnie *Ustawiona* przedstawione są zadane (ustawione) wartości *Timerów* i *Liczników*.

W kolumnie *Inne* podawane są dodatkowe informacje takie, jak tryb dla *Timera*, rodzaj komparacji dla *Komparatora*, itp.

W tabeli Podglądu jak na rys. 6.10.1. w wierszu, w którym wpisany jest T1 mamy informację, że *Timer* ma stan 0, aktualnie pozostało do odliczenia 0.47 s, ustawiony czas to 1 sekunda *Timer* jest ustawiony w trybie generacji impulsów (SL).

W kolumnie *Zmienna* można wpisywać także zmienne symboliczne, jeżeli ikonka  jest aktywna.

6.11. Podgląd drabinki LAD

Program PC Need umożliwia podgląd działania programu LAD w przekaźniku.

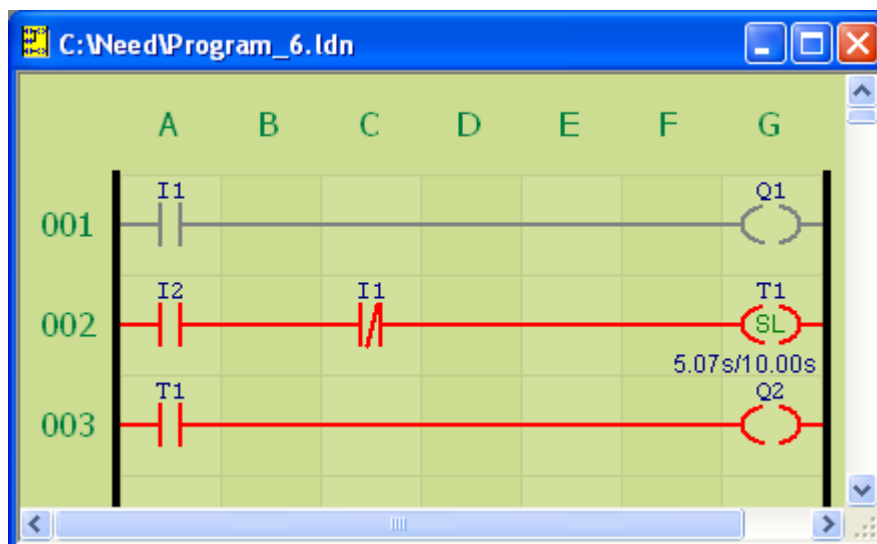
Jeśli mamy aktywne połączenie z przekaźnikiem, to możemy klikając prawym klawiszem myszy, w aktywnym oknie programu LAD rozwinąć menu z którego wybieramy opcję *Podgląd drabinki*.

Połączenie z przekaźnikiem jest sygnalizowane pulsującym napisem *Online* na pasku stanu:

- Dla aktywnego podglądu drabinki i trybu RUN NEED: 24DC-X1-16-8 USB
- Dla aktywnego podglądu drabinki i trybu STOP NEED: 24DC-X1-16-8 USB

Przy braku lub wyłączonej komunikacji wyświetlany jest napis *Offline*.

W celu zakończenia podglądu drabinki należy nacisnąć prawy klawisz myszy na aktywnej drabinie i wybrać Stop F6. Alternatywą jest naciśnięcie klawisza F6.



Rys. 6.11.1. Okno - Podgląd drabinki.

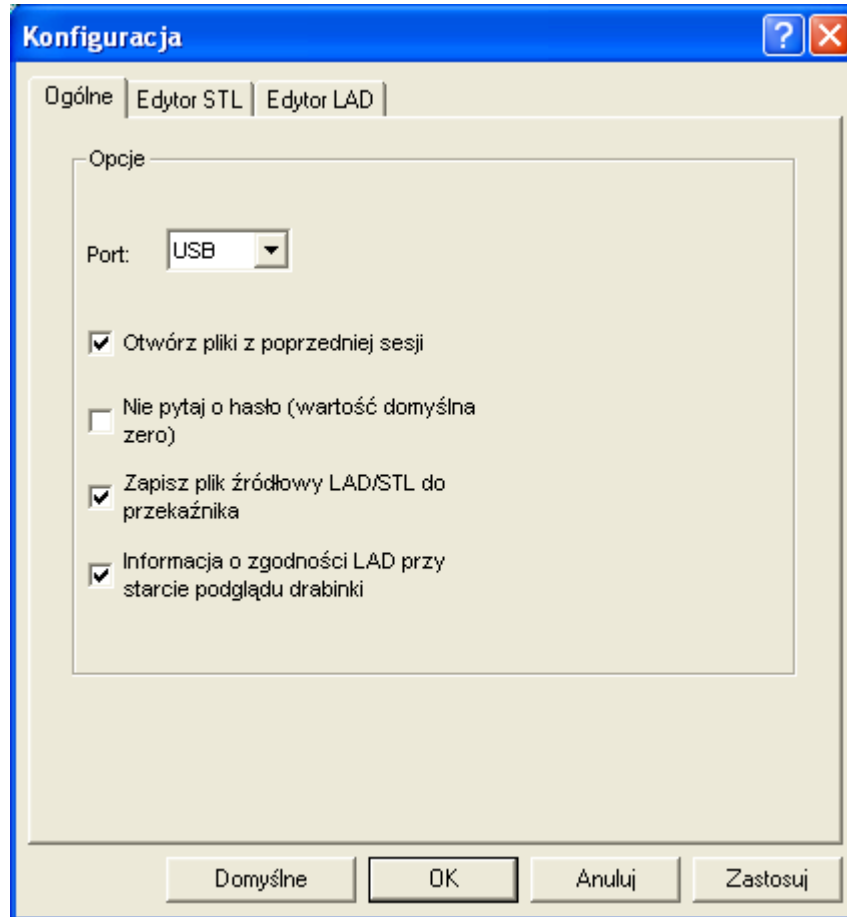
W podglądzie drabinki na czerwono podświetlane są aktywne obwody lub elementy, na szaro nieaktywne. Na rys. 6.11.1 aktywne obwody to 002, 003. Nieaktywny jest obwód 001 gdyż stan wejścia I1 wynosi „0”. W obwodzie 002 wejście I1 jest typu NC, jego stan wynosi „0”. Z punktu widzenia LAD jest ono aktywne. Dodatkowo wyświetlane są informacje o wartości czasu *Timera* bieżącego i ustawionego. Dla *Liczników* podawany jest wartość zliczona i ustawiony próg.

W trybie STOP drabinka jest nieaktywna, nie ma możliwości podglądu.

6.12. Hasło

W celu zablokowania dostępu osobom niepowołanym przekaźnik programowalny NEED może być zabezpieczony czterocyfrowym hasłem (0 do 9999).

Jeżeli chcemy korzystać z domyślnego hasła przy programowaniu Needa, to należy w Menu **Konfiguracja > Opcje** w zakładce *Ogólne* odznaczyć opcję „Nie pytaj o hasło” rys 6.12.1.



Rys. 6.12.1. Uruchomienie pytania o hasło

Domyślne hasło to cyfra 0.

Hasło jest przechowywane w pamięci EEPROM przekaźnika.

Reset przekaźnika przywraca domyślne hasło.

6.12.1. Wprowadzanie hasła

Po uruchomieniu programu PC Need, jeśli nie była zaznaczona opcja „Nie pytaj o hasło”, w celu nawiązania komunikacji z przekaźnikiem NEED należy jednokrotnie podać hasło. Hasło podajemy korzystając z **Menu > Przekaźnik > Hasło > Wprowadź** lub wpisując hasło na żądanie przy obsłudze polecenia związanego z połączeniem z przekaźnikiem. Jeśli hasło nie było ustawione (hasło: 0) to wystarczy zaakceptować polecenie – będzie ono wykonane.

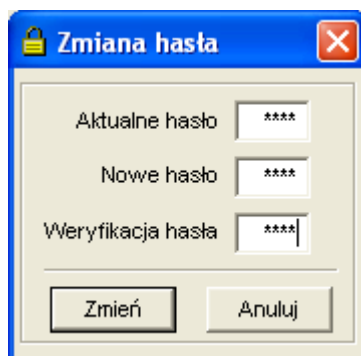


Rys. 6.12.2. Okno Wprowadzanie hasła.

6.12.2. Zmiana hasła

W celu ustawienia lub zmiany istniejącego hasła należy wybrać polecenie: **Menu > Przekaznik > Hasło > Zmień**. W oknie jak na rys. 6.12.2. wprowadzamy obowiązujące hasło (Aktualne hasło) oraz hasło jakie zamierzamy zastosować (Nowe hasło). Dodatkowo w polu „Weryfikacja hasła” powtarzamy nowe hasło w celu uniknięcia pomyłki przy wpisywaniu hasła.

Hasło jest zapisywane w pamięci przekaźnika.



Rys. 6.12.3. Okno Zmiana hasła.



Uwaga: Hasło jest przechowywane w pamięci przekaźnika. Reset przekaźnika kasuje hasło i ustawia hasło na domyślne (hasło = 0). Jednocześnie kasowane są ustawienia i pamięć programu.



Uwaga: W przypadku użycia do zaprogramowania przekaźnika pamięci zewnętrznej hasło w niej zawarte musi być takie jak w przekaźniku. W przypadku, gdy hasła będą różne przekaźnik nie będzie reagował na obecność pamięci zewnętrznej.

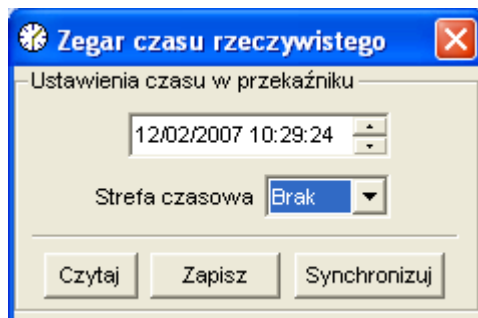
Hasło w pamięci zewnętrznej jest takie, jakie było aktualnie ustawione w programie PC Need podczas jej programowania.

6.13. Zegar czasu rzeczywistego (RTC)

Okno obsługi zegara RTC wywołujemy korzystając z **Menu > Przekaznik > Zegar**

(Ctrl+Shift+Z z klawiatury) lub korzystając z ikonki  na pasku narzędzi.

Możliwy jest podgląd aktualnego czasu w sterowniku – przycisk *Czytaj*, ustawienie dowolnej daty i czasu i skorzystanie z klawisza *Zapisz* oraz ustawienie czasu takiego jaki aktualnie ma urządzenie programujące (PC) przy pomocy opcji *Synchronizuj*.

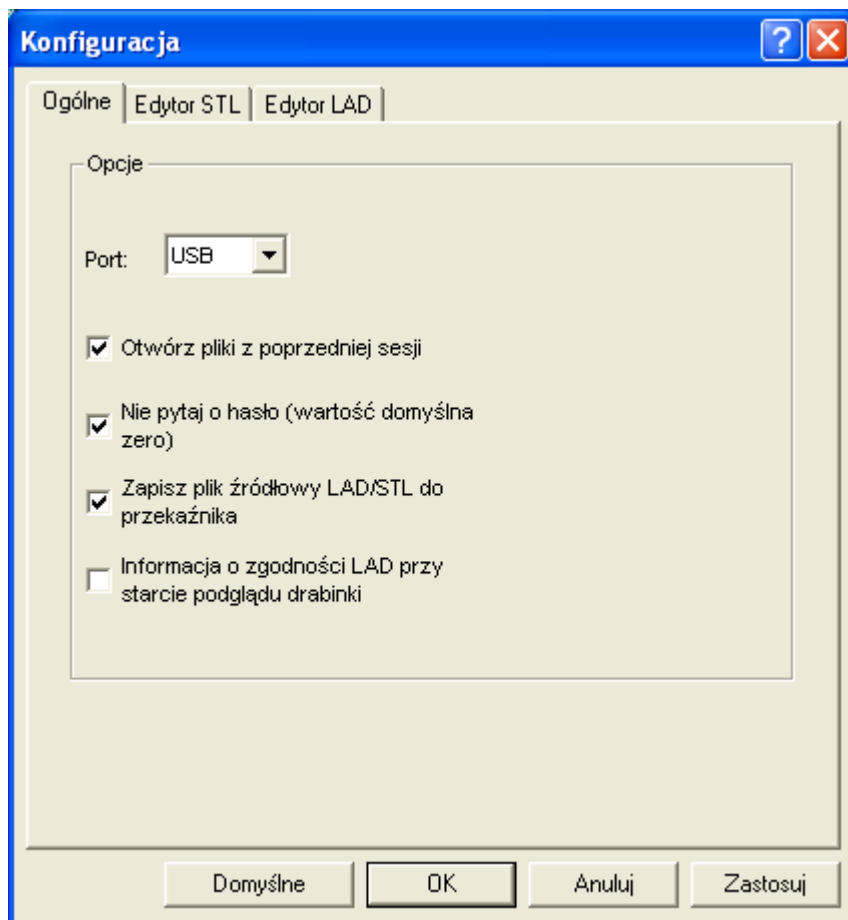


Rys. 6.13.1 Okno Zegar czasu rzeczywistego.

Pole wyboru *Strefa czasowa* umożliwia ustawienie automatycznej zmiany czasu zimowego na letni i odwrotnie w zależności od położenia geograficznego. Wybór *Brak* oznacza, że przekaznik nie będzie uwzględniał zmiany czasu letniego na zimowy i odwrotnie.

6.14. Kod źródłowy

Przełączniki w wersji NEED ...-16-8 umożliwiają zachowanie kodu źródłowego bezpośrednio w przełączniku. Niemożliwe jest załadowanie kodu do pamięci zewnętrznej.



Rys. 6.14.1. Okno Konfiguracja Programu.

Ładowanie programu źródłowego następuje w trakcie transmisji do przełącznika bezpośrednio po załadowaniu kodu wykonawczego i ustawień.

Ładowanie kodu źródłowego jest domyślną ustawioną opcją. Możemy zrezygnować z przechowywania kodu źródłowego w przełączniku odznaczając opcję „Zapisz plik źródłowy” w menu: **Konfiguracja > Opcje** w zakładce *Ogólne*.

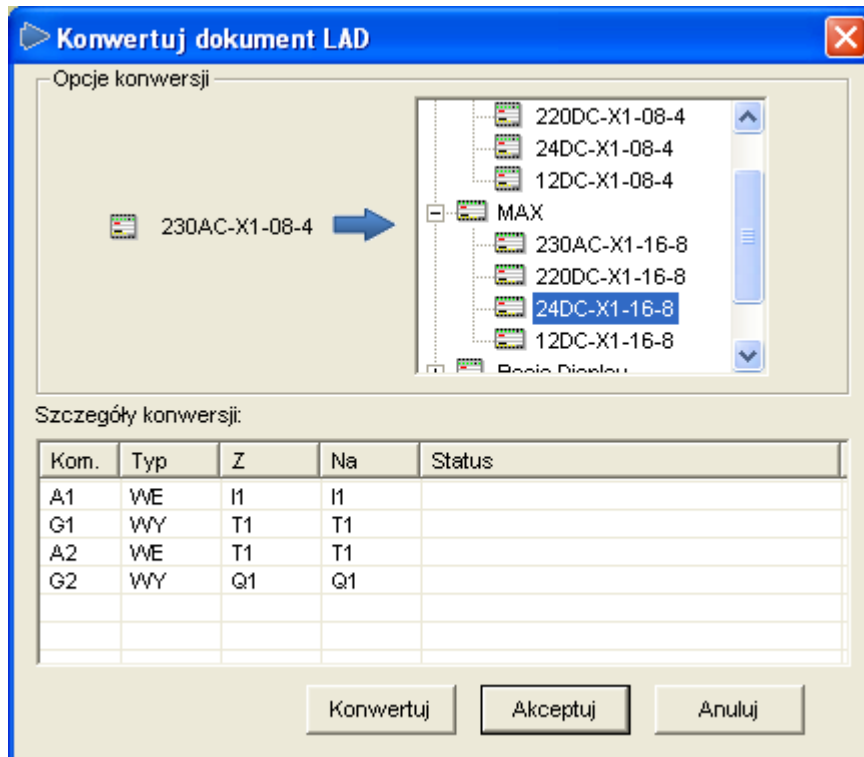
Odczyt kodu źródłowego z przełącznika wykonujemy poprzez menu: **Przełącznik >**

Transmisja > Odczyt kodu programu lub korzystając z przycisku .

Po wczytaniu zawartości projektu PC Need automatycznie utworzy nowe okno w edytorze (LAD, STL) w jakim napisany był program umieszczony w przełączniku.

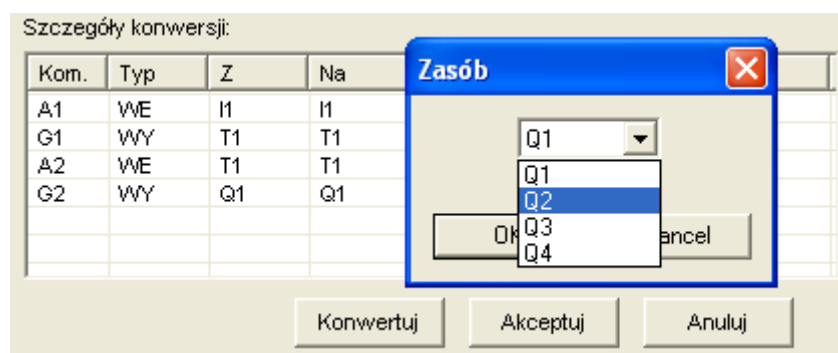
6.15. Konwersja dokumentu

Dla programów LAD i SET możliwe jest automatyczne przenoszenie kodu programu i ustawień pomiędzy różnymi typami przekaźników. Oczywiście nie jest możliwa pełna konwersja we wszystkich przypadkach ze względu na różne zasoby przekaźników, dlatego należy przejrzeć szczegóły konwersji.



Rys. 6.15.1. Okno Konwertuj dokument LAD.

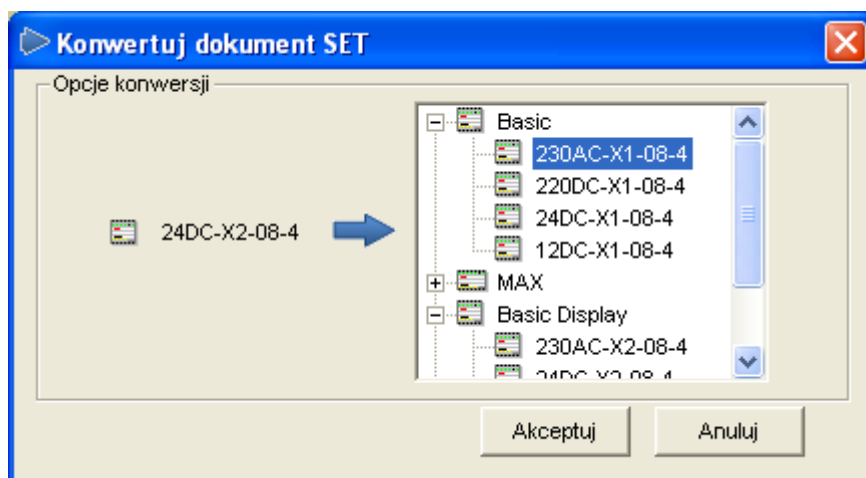
Każdy z docelowych zasobów w tabeli „Szczegóły konwersji” można zmienić, lub przypisać nową wartość w przypadku sygnalizacji **Brak zasobu**. Sygnalizacja o braku zasobu wymaga usunięcia wszystkich zasobów, które nie mają odpowiedników w docelowym programie. Np. konwertując program z użyciem *Znaczników MT (NEED z wyświetlaczem)* do wersji bez wyświetlacza należy je usunąć.



Rys. 6.15.2. Zmiana Zasobu.

Na rys. 6.15.2 pokazano dostępne opcje zmiany dla komórki G2 (kolumna Kom.) czyli zastąpienia wyjścia WY (kolumna Typ) z Q1 (kolumna Z) na (kolumna Na) Q2.

Dla Ustawień konwersja jest w pełni automatyczna. Po podaniu dla otwartego aktywnego okna Ustawień typu docelowego i zaakceptowaniu otrzymujemy nowy dokument. Przenoszone są tylko ustawienia dostępne dla docelowego typu przekaźnika, a pozostałe ustawiane na wartości domyślne.



Rys. 6.15.3. Konwersja Ustawień

7. URUCHOMIENIE

7.1. Załączenie

7.1.1. Czynności wstępne dla wersji AC



1. Sprawdzić prawidłowość podłączenia napięcia zasilającego:
zacisk L: przewód fazowy 230V AC
zacisk N: przewód zerowy
2. Sprawdzić prawidłowość podłączenia wejść i wyjść przełącznika;
Uwaga: wejścia I1.. In sterowane przez przewód fazowy L
3. Ustawić przełącznik RUN/STOP w pozycję STOP.
4. Układy sterowane przez przełącznik programowalny zabezpieczyć przed dostępem niepowołanych osób – przy pierwszym uruchomieniu istnieje ryzyko niekontrolowanego działania maszyn (napędy, pompy, wentylatory) i urządzeń lub wystąpienia niebezpiecznych napięć na wejściach. Może to być spowodowane np. błędem w programie lub okablowaniu.

7.1.2. Czynności wstępne dla wersji DC

1. Sprawdzić prawidłowość podłączenia napięcia zasilającego:
 - zacisk +24V DC: przewód zasilający dodatni zasilania 24V DC dla wersji 24DC
 - zacisk +12V DC: przewód zasilający dodatni zasilania 12V DC dla wersji 12DC
 - zacisk +220V DC: przewód zasilający dodatni zasilania 220V DC dla wersji 220DC
 - zacisk 0V: masa zasilania
2. Sprawdzić prawidłowość podłączenia wejść i wyjść przełącznika;
Uwaga: wejścia I1.. In sterowane przez napięcie dodatnie względem zacisku 0V.
3. Ustawić przełącznik RUN/STOP w pozycję STOP.
4. Układy sterowane przez przełącznik programowalny zabezpieczyć przed dostępem niepowołanych osób – przy pierwszym uruchomieniu istnieje ryzyko niekontrolowanego działania maszyn (napędy, pompy, wentylatory) i urządzeń lub wystąpienia niebezpiecznych napięć na wejściach. Może to być spowodowane np. błędem w programie lub okablowaniu.

7.1.3. Załączenie zasilania

1. Dołączyć zewnętrzne zasilanie do zacisków przełącznika programowalnego.
2. Sprawdzić działanie niezależnych elementów bezpieczeństwa (jeżeli są) – np. wyłącznik awaryjny zasilania.
3. Ocenić prawidłowość sygnalizacji diodami LED wejść przełącznika programowalnego.
4. Przetawić przełącznik RUN/STOP w pozycję RUN.

Obserwować zachowanie układu – w wypadku nieprawidłowego działania sprawdzić układ połączeń i program sterujący.



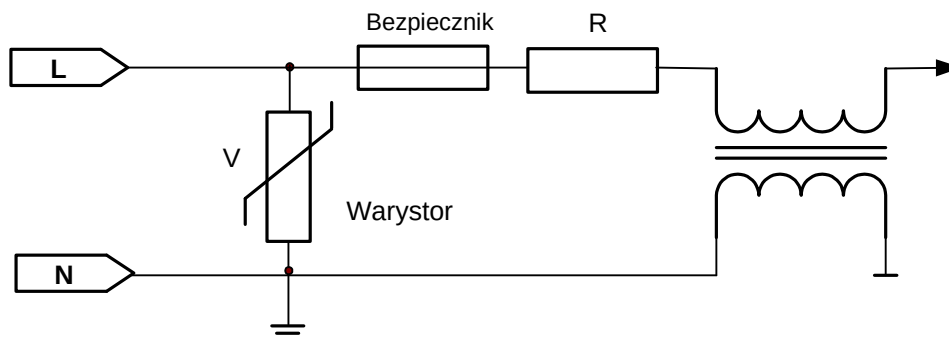
Od momentu przestawienia w tryb RUN, natychmiast uruchamiany jest program, który przejmuje kontrolę nad wyjściami.

8. INFORMACJE SPRZĘTOWE

8.1. Zasilanie przekaźnika

8.1.1. Zasilanie przekaźnika 115/230 V AC

Schemat ideowy obwodu zasilania przekaźnika NEED przedstawiono na rys. 8.1.1.



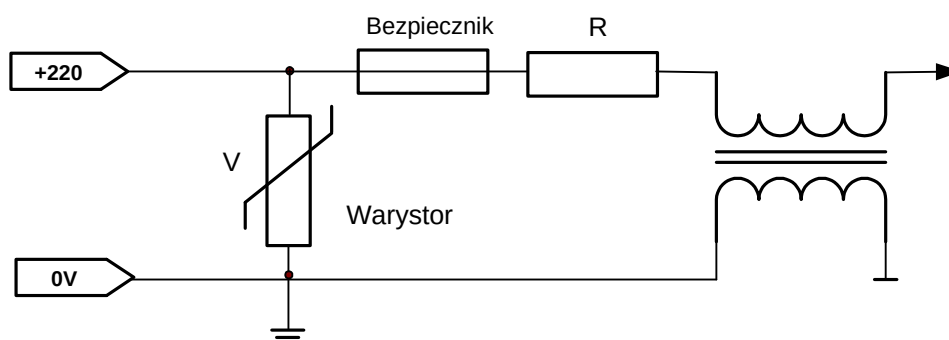
Rys.8.1.1. Schemat ideowy zasilania przekaźnika NEED AC



Układ zasilacza przekaźnika NEED nie jest odseparowany galwanicznie od sieci elektrycznej. Oznacza to, że w przypadku zamiany przewodów na zaciskach L i N na złączu komunikacyjnym mogą wystąpić napięcia groźne dla życia.

8.1.2. Zasilanie przekaźnika 220 V DC

Schemat ideowy obwodu zasilania przekaźnika NEED przedstawiono na rys. 8.1.2.



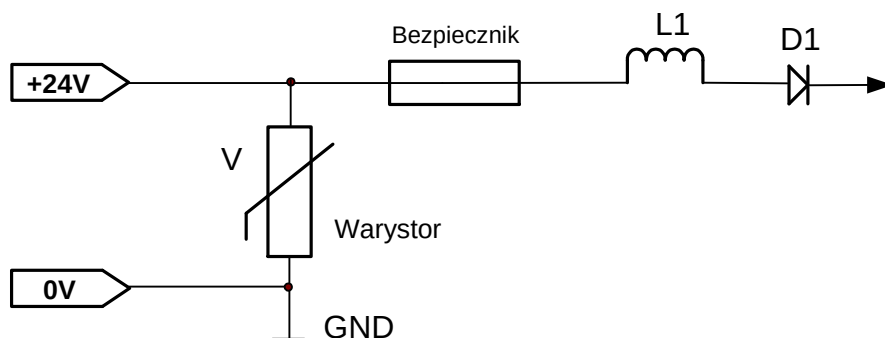
Rys.8.1.2. Schemat ideowy zasilania przekaźnika NEED 220V DC



Układ zasilacza przekaźnika NEED nie jest odseparowany galwanicznie od sieci elektrycznej. Oznacza to, że na zaciskach i na złączu komunikacyjnym mogą wystąpić napięcia groźne dla życia.

8.1.3. Zasilanie przekaźnika 24 (12) V DC

Schemat ideowy obwodu zasilania przekaźnika NEED dla wersji 24V DC przedstawiono na rys. 8.1.3. Wersja 12 V DC różni się doбором elementów.



Rys.8.1.3. Schemat ideowy zasilania przekaźnika NEED DC



Układy zasilania przekaźnika w wersji DC są zabezpieczone przed odwrotnym podłączeniem napięcia zasilającego.

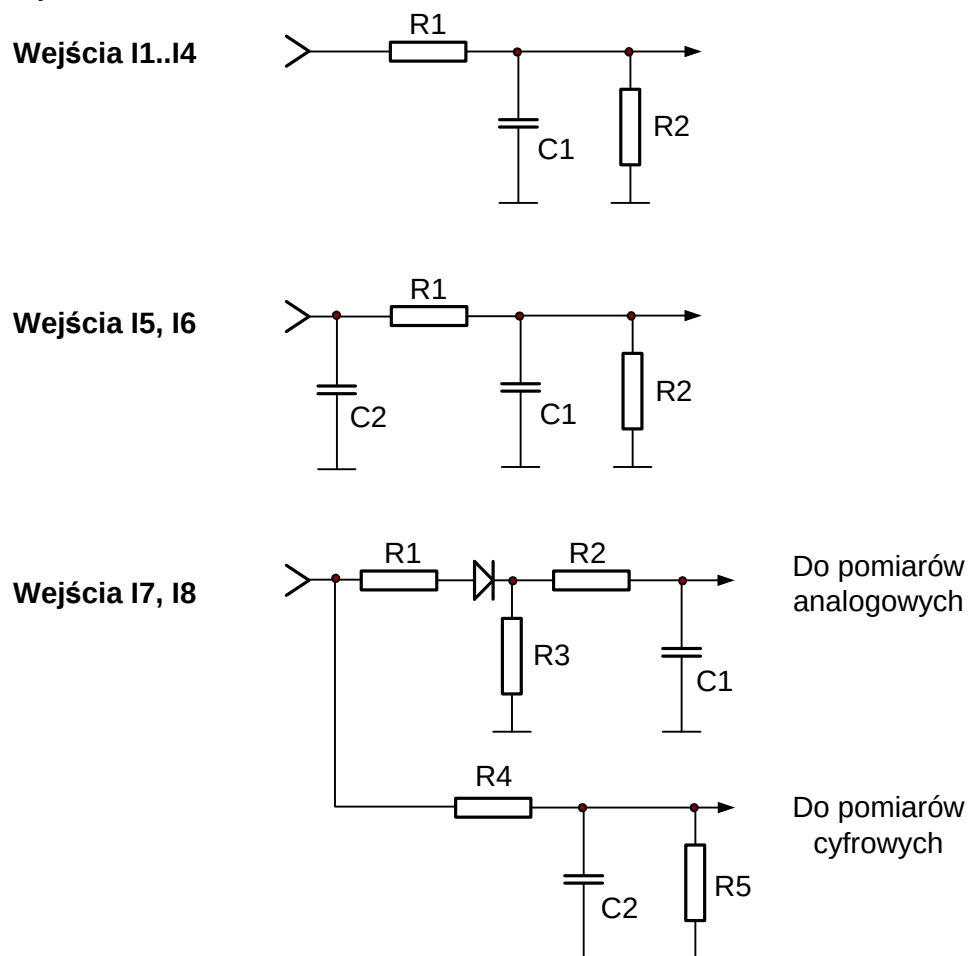
8.2. Wejścia

8.2.1. Wejścia 230 V AC

Schemat ideowy układów wejściowych przekaźników NEED-230AC-xx-8-.. został przedstawiony na rys. 8.2.1 a NEED-230AC-xx-16-.. na rys. 8.2.2.

Wejścia analogowe to I7 i I8 dla wersji DC NEED..-xx-8-.. oraz I14, I15, I16 dla wersji NEED..-xx-16-...

Wejścia o zwiększonej odporności na zakłócenia to I5 i I6 dla wersji AC NEED..-xx-8-.. oraz I12, I13 dla wersji AC NEED..-xx-16-...



Rys. 8.2.1. Schemat ideowy wejść przekaźnika NEED-230AC-xx-8-4.

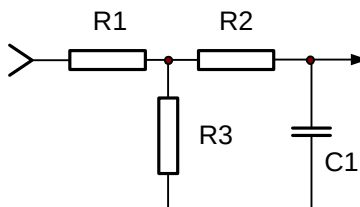


Wejścia o zwiększonej odporności na zakłócenia posiadają kondensator przeciwzakłóceńowy, który pozwala dołączać do nich długie przewody. Wejścia I7, I8 pełnią rolę wejść cyfrowych i analogowych – patrz rozdział „4.11. Komparator – Wejście analogowe”.

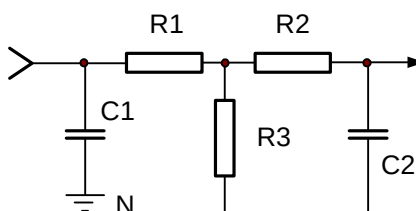


Wejścia przekaźnika NEED nie są odseparowane galwanicznie od zasilania.

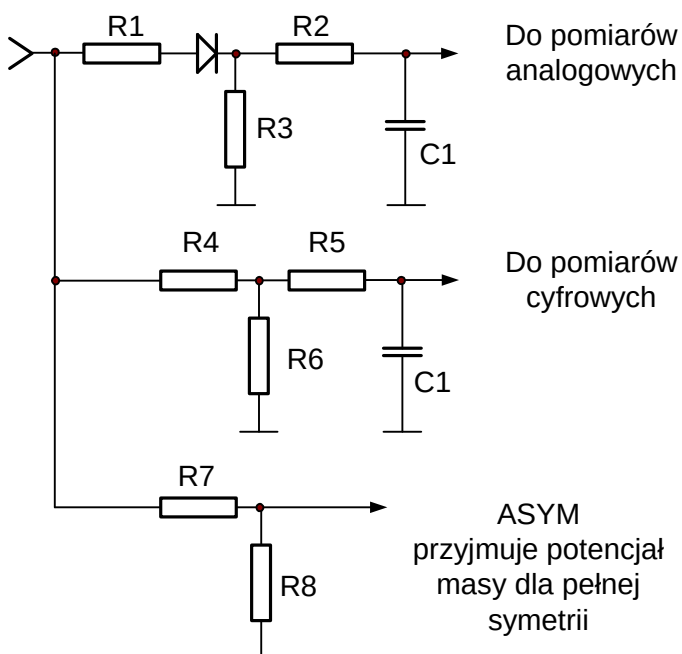
Wejścia I1..I11



Wejścia I12, I13



Wejścia I14..I16



Rys. 8.2.2. Schemat ideowy wejść przekaźnika NEED-230AC-xx-16-8.



Wejścia o zwiększonej odporności (I11, I12) na zakłócenia posiadają kondensator (filtr przeciwzakłóceńowy), który pozwala dołączać do nich długie przewody. Wejścia I14, I15, I16 pełnią rolę wejść cyfrowych i analogowych – patrz rozdział „4.11. Komparator – Wejście analogowe”.

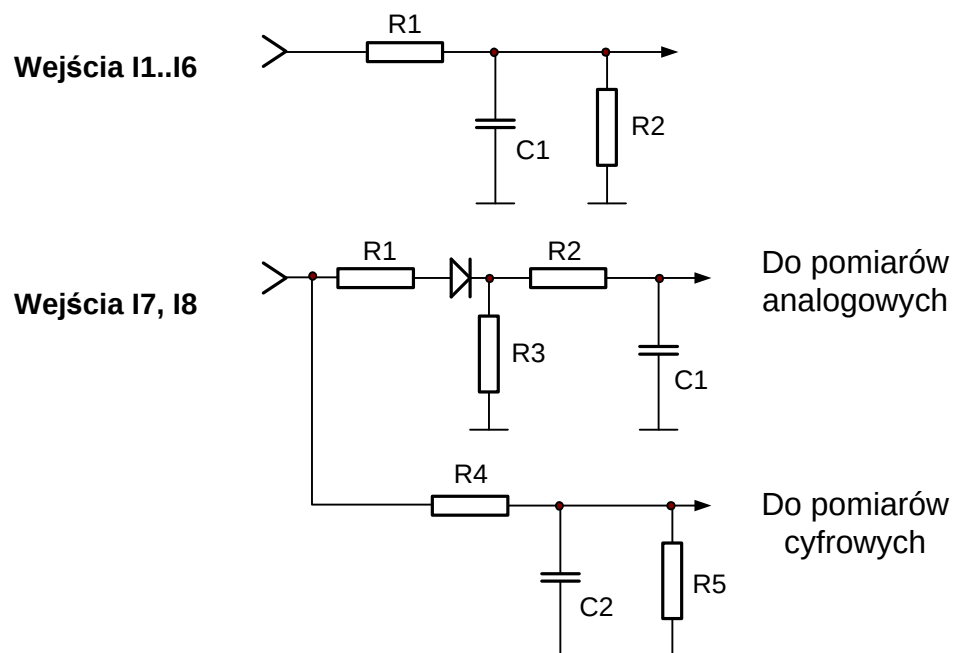


Wejścia przekaźnika NEED-230AC-xx nie są odseparowane galwanicznie od zasilania (sieci elektrycznej).

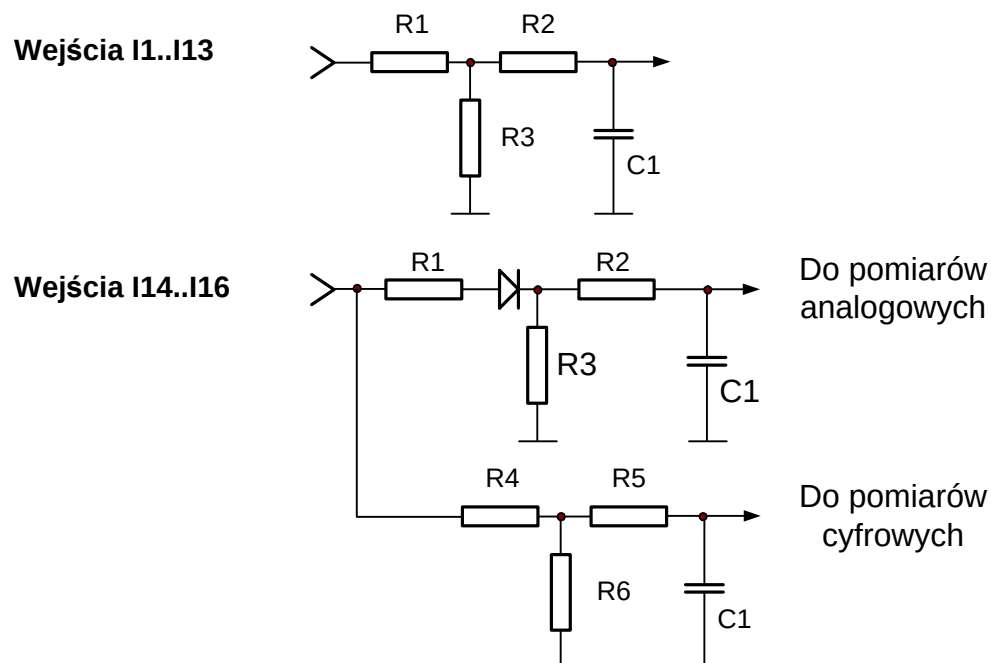
8.2.2. Wejścia 220 V DC

Schemat ideowy układów wejściowych przekaźnika NEED-220DC-11-8-.. został przedstawiony na rys. 8.2.3. a NEED-220DC-11-16-.. na rys. 8.2.4.

Wejścia analogowe to I7 i I8 dla wersji NEED-220DC-11-8-.. oraz I14, I15, I16 dla wersji NEED-220DC-11-16-...



Rys. 8.2.3. Schemat ideowy wejść przekaźnika NEED-220DC-11-8-4.



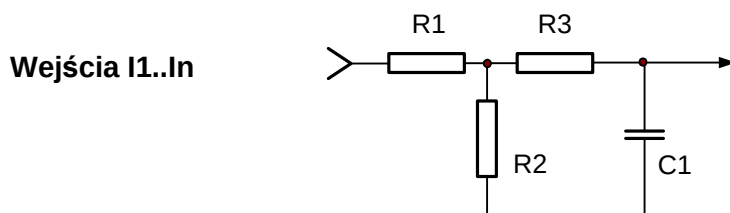
Rys. 8.2.4 Schemat ideowy wejść przekaźnika NEED-220DC-11-16-8.



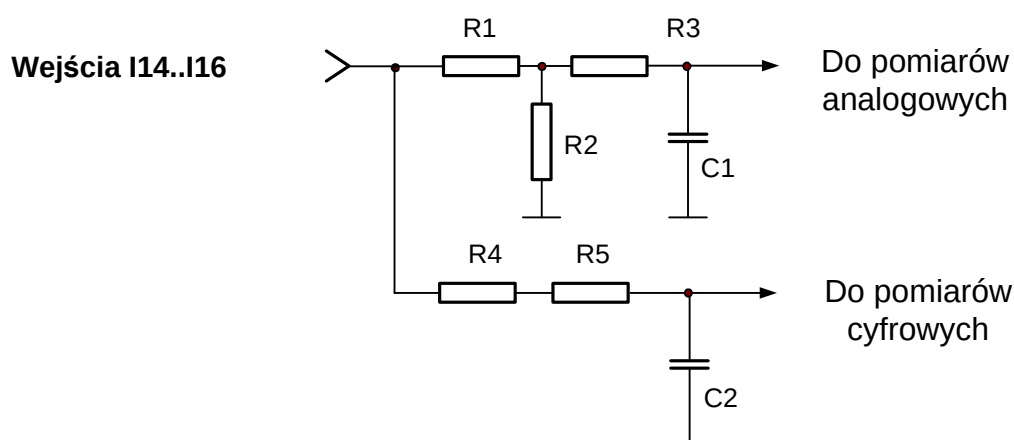
Wejścia przekaźnika NEED-220DC-11-... nie są odseparowane galwanicznie od zasilania.

8.2.3. Wejścia 24 (12) V DC

Schemat ideowy układów wejściowych przekaźnika NEED DC został przedstawiony na rys. 8.2.5. Wszystkie wejścia cyfrowe i cyfrowo-analogowe dla wersji NEED-..DC-xx-8-4 mają podobny układ połączeń. Wejścia analogowo-cyfrowe mają inny układ połączeń dla wersji NEED-..DC-xx-16-8 – pokazano to na rys. 8.2.6.



Rys. 8.2.5. Schemat ideowy wejść przekaźnika NEED DC.



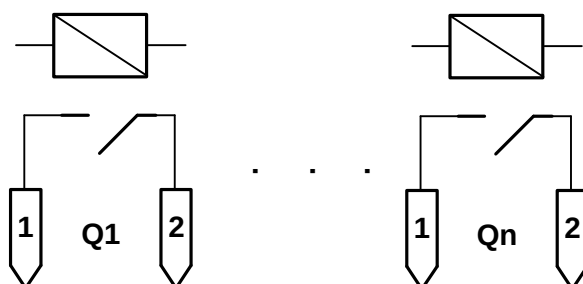
Rys. 8.2.6. Schemat ideowy wejść analogowo-cyfrowych przekaźnika NEED-24DC-xx-16-8.



Wejścia przekaźnika NEED-..DC-xx-... nie są odseparowane galwanicznie od zasilania.

8.3. Wyjścia

Schemat ideowy układów wejściowych przekaźnika NEED został przedstawiony na rys. 8.3.



Rys. 8.3. Schemat ideowy wyjść przekaźnika NEED.

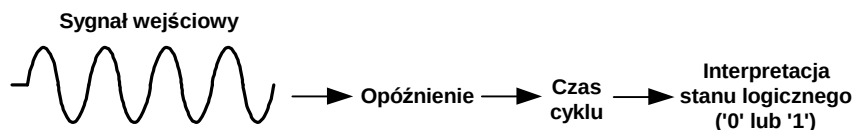
Wyjścia przekaźnika NEED to bezpotencjałowe styki przekaźników.



Wyjścia przekaźnika NEED są odseparowane galwanicznie od wejść oraz zasilania.

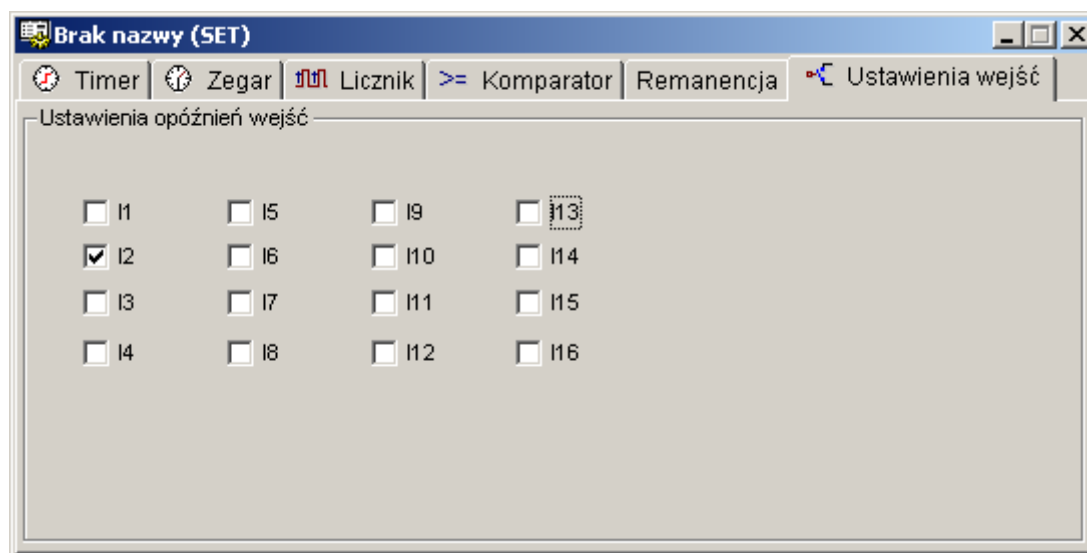
8.4. Opóźnienie wejść

W wielu problemach sterowania bardzo kłopotliwą rzeczą jest drganie styków np. przełącznika. W przekaźniku programowalnym NEED można ustawić odpowiednie opóźnienie wejść tak, aby wyeliminować te niekorzystne zjawiska. Przetwarzanie sygnałów wejściowych w przekaźniku NEED pokazuje rys. 8.4.1.



Rys. 8.4.1. Przetwarzanie sygnałów wejściowych w przekaźniku NEED.

W przekaźniku NEED opóźnienie wejść można ustalać za pomocą konfiguracji programowej (patrz rozdział „6. Instalacja i opis oprogramowania”). Przykładowe okienko konfiguracyjne programu PC Need z ustawieniem opóźnienia dla wejścia I2 przedstawiono na rys. 8.4.2.



Rys. 8.4.2. Przykładowa konfiguracja opóźnienia dla wejść.

8.4.1. Opóźnienia wejść dla przekaźnika NEED-230AC-x1-..

Jeśli wejścia ustawimy bez opóźnienia, to przekaźnik NEED, co 20ms (jedna, dodatnia połówka przebiegu sinusoidalnego dla częstotliwości napięcia zasilającego 50Hz) – będzie sprawdzał stan sygnału wejściowego i od razu po sprawdzeniu, będzie dokonywał interpretacji czy poziom napięcia pojawiający się na wejściu zaliczyć do stanu wysokiego lub niskiego. Czyli maksymalny czas interpretacji stanu sygnału wejściowego bez opóźnienia wynosi:

20ms + czas cyklu programu.

W maksymalnie tak długim czasie, sygnał na wejściu bez opóźnienia, może być „zauważony” i prawidłowo zinterpretowany przez przekaźnik.

Jeśli wejścia ustawimy z opóźnieniem, to przekaźnik NEED co ok. 20ms (dla częstotliwości napięcia zasilającego 50Hz) będzie dokonywał interpretacji sygnału wejściowego. Jeśli po trzecim sprawdzeniu stan na badanym wejściu nie zmieni się, to przekaźnik dokona odpowiedniej interpretacji, zaliczając poziom napięcia do stanu niskiego lub wysokiego. Czyli maksymalny czas interpretacji stanu sygnału wejściowego z opóźnieniem wynosi:

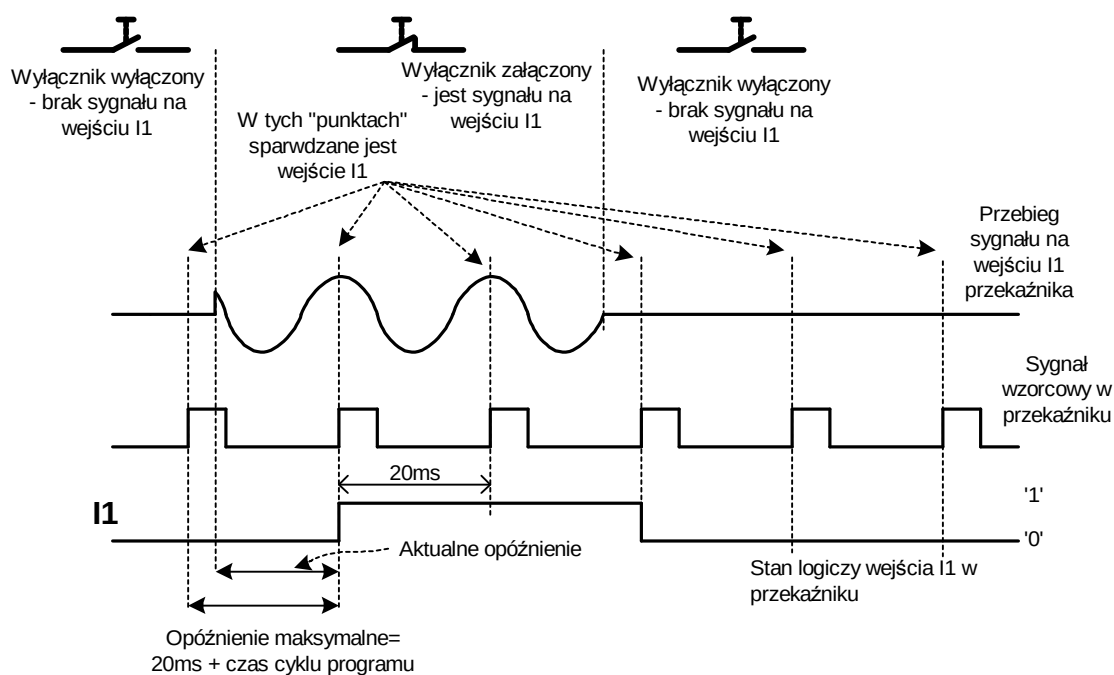
60ms + czas cyklu programu.

W maksymalnym czasie 60ms (jednak nie krótszym niż 40ms), sygnał na wejściu NEED'a,

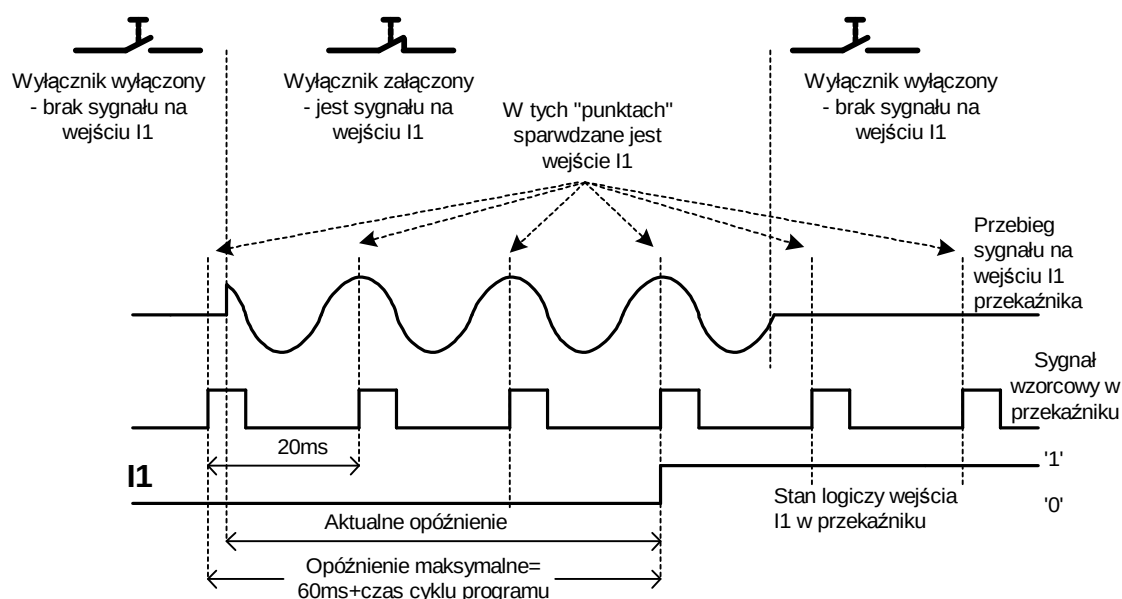
może być „zauważony” i prawidłowo zinterpretowany przez przekaźnik.
W tabeli 8.1. przedstawiono czasy opóźnień dla wejść przekaźnika programowalnego.
Tab.8.1. Czasy opóźnień wejść przekaźnika NEED.

Częstotliwość napięcia zasilającego	Maksymalny czas opóźnienia sygnału wejściowego	
	Opóźnienie załączone	Opóźnienie wyłączone
f=50Hz	60ms+czas cyklu	20ms+czas cyklu
f=60Hz	49,8ms+czas cyklu	16,6ms+czas cyklu

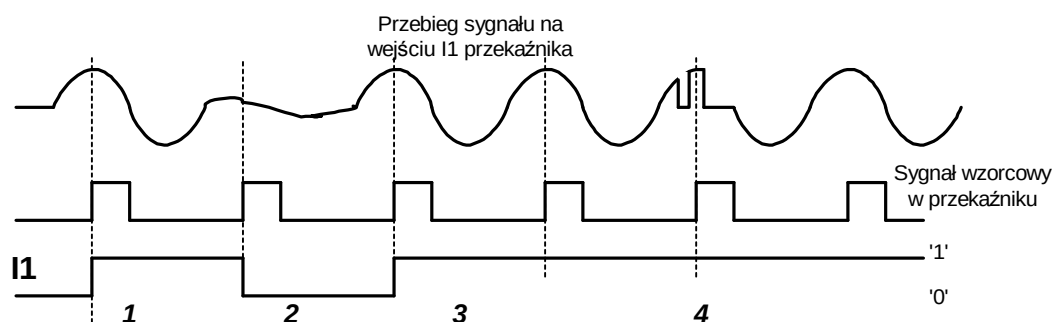
Na rys. 8.4.3. oraz rys. 8.4.4. przedstawiono interpretację stanów logicznych wejść przekaźnika NEED-230AC-..



Rys.8.4.3. Interpretacja stanu logicznego wejścia I1 przekaźnika dla nie ustawionego czasu opóźnienia – NEED-230AC-..



Rys.8.4.4. Interpretacja stanu logicznego wejścia I1 przekaźnika dla ustawionego czasu opóźnienia – NEED-230AC-..



Rys.8.4.5. Przykładowa interpretacja stanu logicznego wejścia I1 przekaźnika dla nie ustawionego czasu opóźnienia – NEED-230AC-..

Dla przebiegu sygnału wejściowego, jak na rys. 8.4.5., możemy wyróżnić następujące etapy: sygnał wejściowy jest w stanie wysokim (1), więc przekaźnik zinterpretuje go jako logiczną jedynkę. Jednak, jeśli przekaźnik nie wykryje w następnych 20ms prawidłowego poziomu sinusoidy, wtedy zmieni stan swojego wejścia na zero logiczne (2). Po kolejnych 20ms sygnał wejściowy zostaje zinterpretowany jako stan wysoki (3). Impulsy krótkie mogą być detekowane prawidłowo, jeśli „natrafiają” na odpowiedni moment w interpretacji sygnału wejściowego przez przekaźnik (4).

8.4.2. Opóźnienia wejść dla przekaźnika NEED-24DC -.., NEED-12DC -.., NEED-220DC -..

Jeśli wejścia ustawimy bez opóźnienia, to przekaźnik NEED, raz na obieg pętli programu, będzie sprawdzał stan sygnału wejściowego i od razu po sprawdzeniu, będzie dokonywał interpretacji czy poziom napięcia pojawiający się na wejściu zaliczyć do stanu wysokiego lub niskiego. Czyli maksymalny czas interpretacji stanu sygnału wejściowego bez opóźnienia wynosi:

maksymalny czas cyklu programu.

W maksymalnie tak długim czasie, sygnał na wejściu bez opóźnienia, może być „zauważony” i prawidłowo zinterpretowany przez przekaźnik.

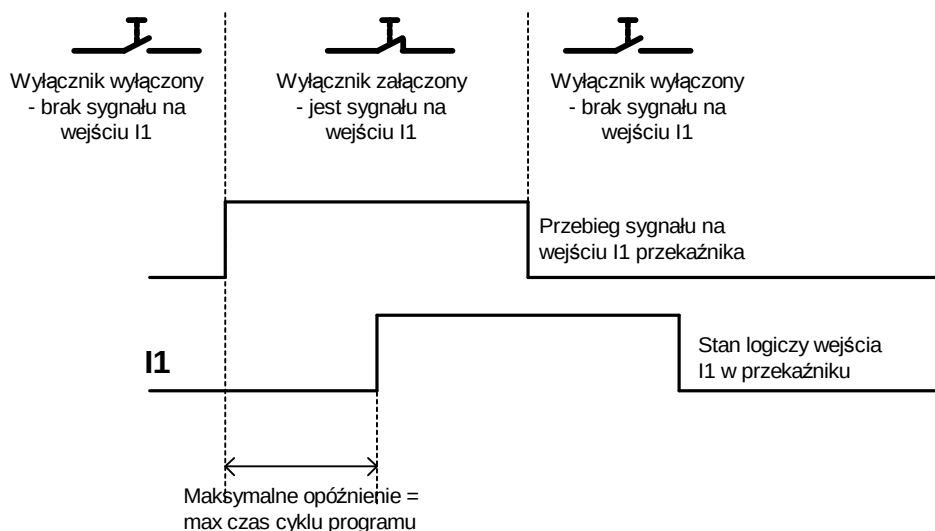
Jeśli wejścia ustawimy z opóźnieniem, to przekaźnik NEED co, 21ms będzie dokonywał interpretacji sygnału wejściowego. Czyli maksymalny czas interpretacji stanu sygnału wejściowego z opóźnieniem wynosi:

21ms + czas cyklu programu.

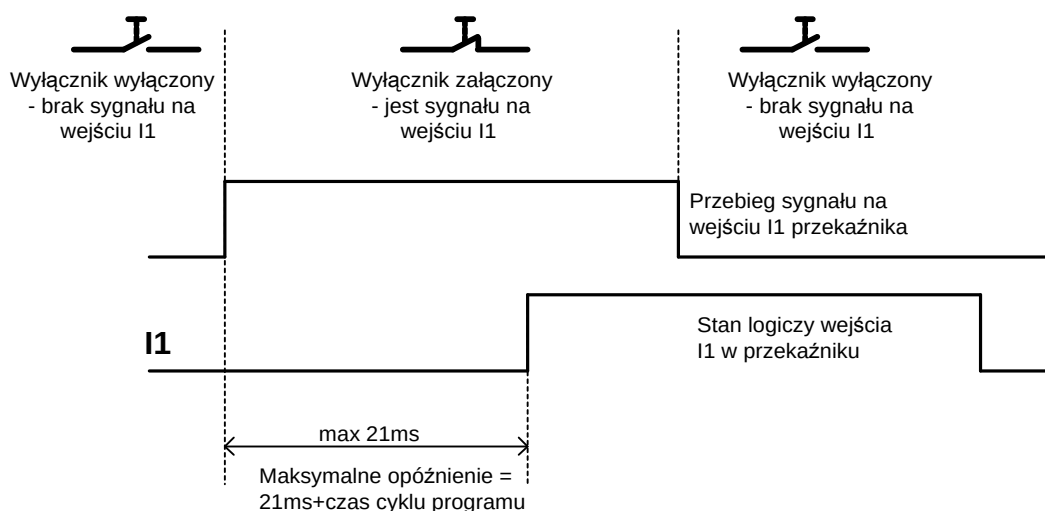
W maksymalnym czasie 21ms sygnał na wejściu NEED’a, może być „zauważony” i prawidłowo zinterpretowany przez przekaźnik.



Jeżeli w trakcie odmierzania czasu opóźnienia, dla przekaźników NEED DC sygnał wejściowy ulegnie zmianie, to liczenie czasu opóźnienia rozpoczynane jest od początku.



Rys.8.4.6. Przykładowa interpretacja stanu logicznego wejścia I1 przekaźnika dla nie ustawionego czasu opóźnienia – NEED-24DC-..., NEED-12DC-..., NEED-220DC-..



Rys.8.4.7. Przykładowa interpretacja stanu logicznego wejścia I1 przekaźnika dla ustawionego czasu opóźnienia – NEED-24DC-..., NEED-12DC-..., NEED-220DC-...



Przetwarzanie sygnałów wejściowych w przekaźniku NEED odbywa się z opóźnieniem.

8.5. Opóźnienie wyjść

Wyjścia w przekaźniku NEED nie są opóźniane – są ustawiane tak szybko, jak jest to możliwe. Jednakże należy wziąć pod uwagę opóźnienia wynikające z zastosowanych elementów wyjściowych, sterujących np. dla wersji NEED-230AC-..-08- 4R czas zadziałania wyjść to:

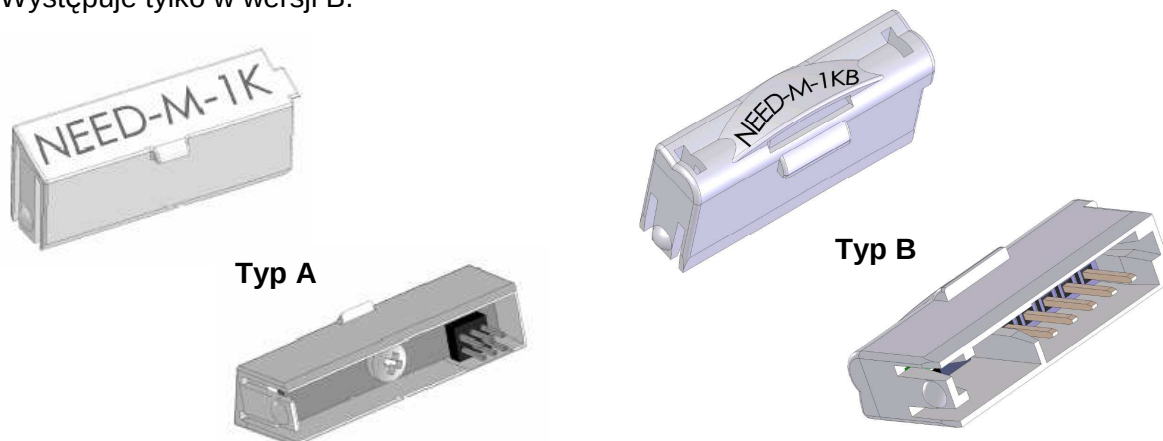
Czas zadziałania przekaźnika wyjściowego + czas cyklu.

9. PAMIĘĆ ZEWNĘTRZNA

9.1. Karty pamięci

Dla zwiększenia funkcjonalności przekaźnika NEED dostępne są karty pamięci zewnętrznej NEED-M-1K i NEED-M-4K. Odpowiednio są to moduły EEPROM o pojemności 1 Kbajt i 4Kbajt. Pamięć ta może być wykorzystywana do przepisywania programu do przekaźnika NEED bez używania komputera.

Pamięć NEED-M-1K (1Kbajt) służy tylko do przekaźników w wersji NEED ..-01-...-..., NEED ..-11-...-.... Występuje w 2 wersjach różniących się rodzajem złącza – starszej A i nowszej B. Pamięć NEED-M-4K (4Kbajt) służy tylko do przekaźników które posiadają oznaczenie wersji NEED ..-22-...-... (Wersja **22** np. NEED 24DC-**22**-16-8 z wyświetlaczem LCD). Występuje tylko w wersji B.

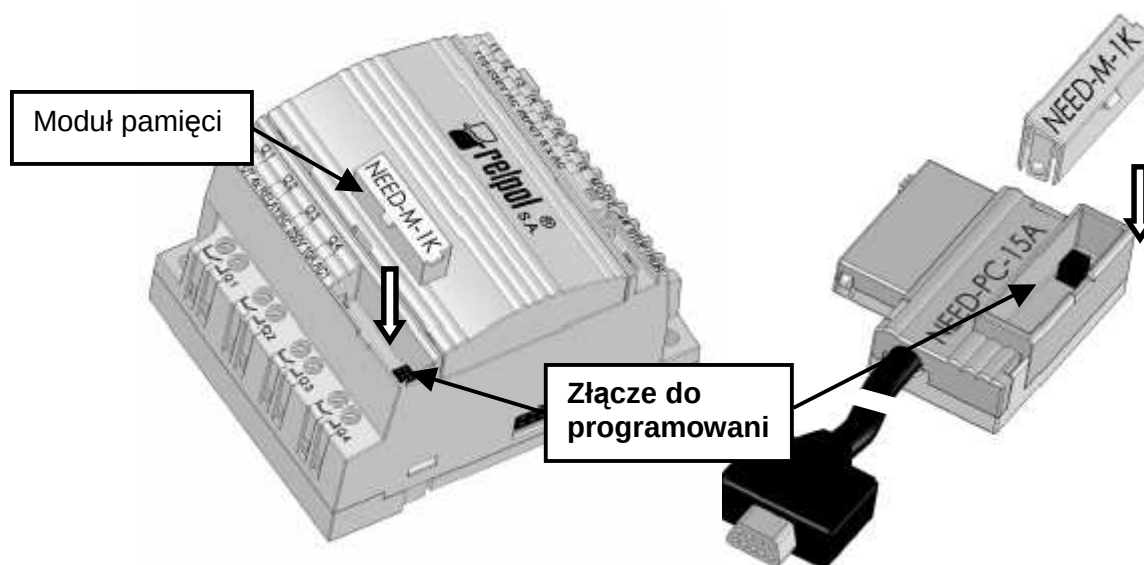


Rys. 9.1.1. Moduł pamięci zewnętrznej – widok z góry i z dołu.

Moduł programujemy korzystając z przewodu do programowania przekaźnika NEED. W tym celu umieszczamy moduł w przeznaczonym do tego złączu, we wtyczkę kabla programującego (rys. 9.1.2).

Możliwy jest także odczyt z partycji pamięci ustawień tam zapisanych.

Zaprogramowaną kartę pamięci po wyjęciu z wtyczki kabla programującego umieszczamy w przekaźniku zamiast wtyku przewodu programującego – przewód programujący i pamięć korzystają z tego samego złącza w przekaźniku NEED.



Rys. 9.1.2. Miejsce instalacji modułu pamięci zewnętrznej w przekaźniku i złączu kabla.

Uwaga: Przewodu z włożonym modulem pamięci nie należy podłączać do złącza programującego przekaźnika i używać do programowania przekaźnika.

Uwaga: Używanie przewodu do programowania z włożonym modulem pamięci może doprowadzić do niekontrolowanego przesyłu danych do przekaźnika lub modułu pamięci.

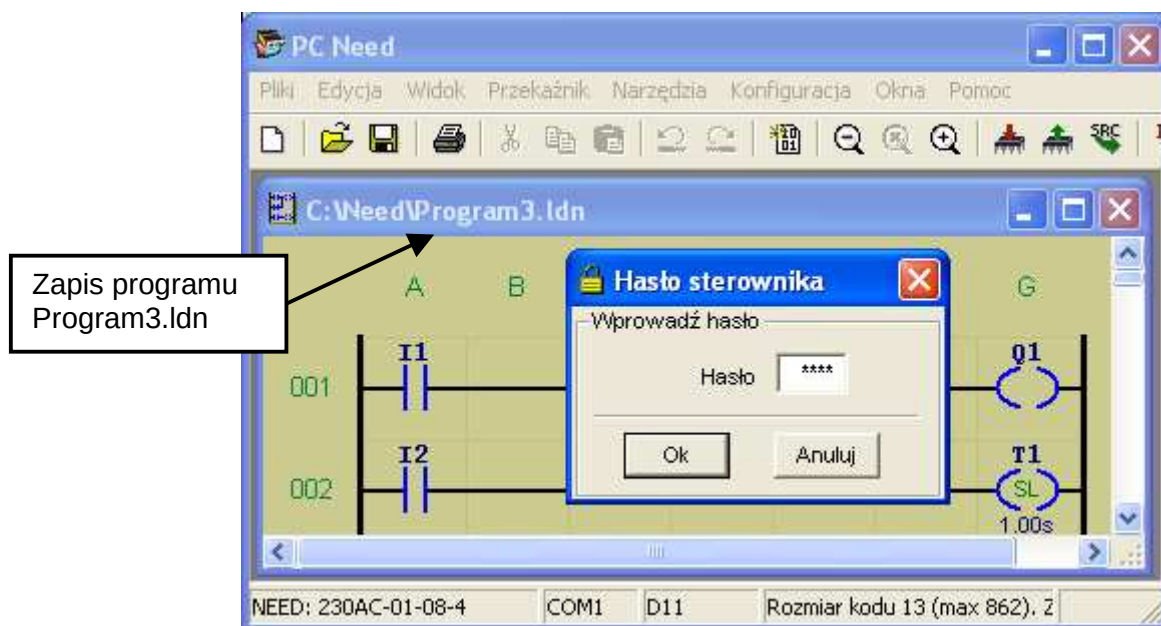
9.2. Organizacja pamięci

Pamięć na karcie została podzielona na 2 partycje – jedna służy do przechowywania kodu programu, druga do przechowywania ustawień. Przy programowaniu karty pamięci możemy dokonać wyboru, które partycje są aktywne. Jeśli partycja jest aktywna, to zapisane w niej dane są przepisywane do pamięci przekaźnika NEED. Możliwe są więc sytuacje, że wgrywamy tylko kod programu, wgrywamy tylko nowe ustawienia, wgrywamy kod programu i ustawienia. Jeśli obie partycje będą nieaktywne, to przekaźnik nie załaduje żadnych danych do swojej wewnętrznej pamięci.

9.3. Programowanie pamięci

9.3.1. Zapis programu

Jeśli mamy podłączoną kartę pamięci do złącza przewodu programującego, to ustawiamy jako aktywne (na wierzchu, niebieskie podświetlenie nagłówka) okno programu, którego kod chcemy zapisać do pamięci. Następnie wykonujemy komendę **Przekaźnik > Pamięć zewnętrzna > Zapis**.



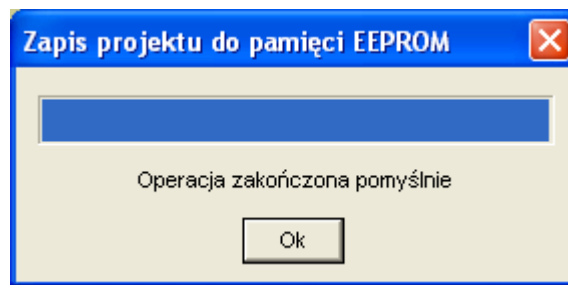
Rys. 9.3.1.1. Zapis Programu.

Po otwarciu okna *Zapis programu* ustawiamy hasło zgodne z ustawionym w przekaźniku a następnie naciskamy przycisk *Start*. Pytanie o hasło ma miejsce tylko wtedy, gdy nie była ustawiona opcja „Nie pytaj o hasło” w konfiguracji programu.

Uwaga: Jeśli wpisane hasło przy programowaniu pamięci jest inne niż hasło w przekaźniku NEED, wtedy program nie jest przepisywany do wewnętrznej pamięci przekaźnika programowalnego.

Uwaga: Dla poprawnego zapisu wymagane jest aby moduł pamięci (1k / 4k) był zgodny z typem programowanego przekaźnika.

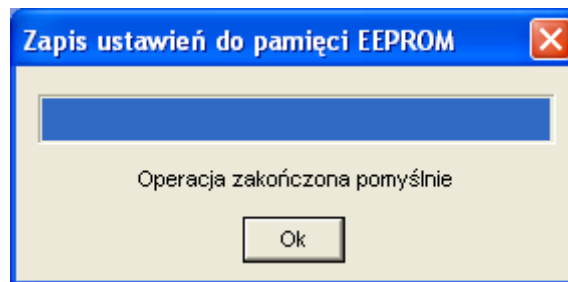
Po wykonaniu operacji otrzymamy komunikat o zakończeniu zapisu programu.



Rys. 9.3.1.2. Okno Zapis programu.

9.3.2. Zapis ustawień

Jeśli mamy podłączoną kartę pamięci do złącza przewodu programującego, to ustawiamy jako aktywne (na wierzchu, niebieskie podświetlenie nagłówka) okno Ustawienia. Następnie wykonujemy komendę **Przełącznik > Pamięć zewnętrzna > Zapis**.

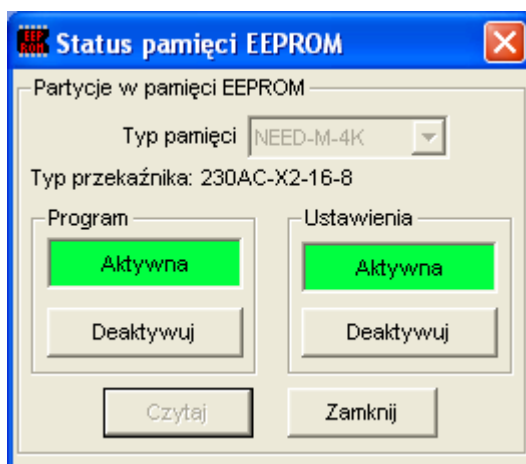


Rys. 9.3.2.1. Okno zapis Ustawień.

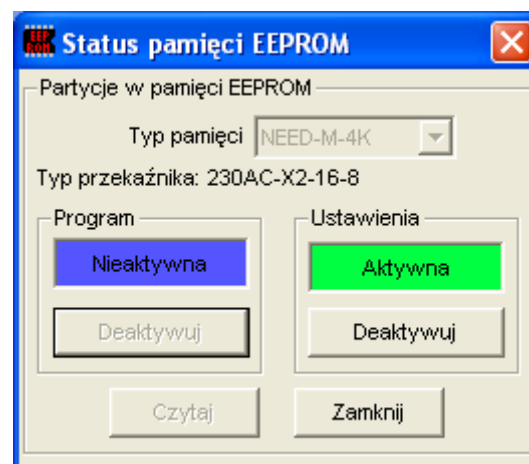
Po otwarciu okna *Zapis ustawienia przełącznika* decydujemy się czy zabezpieczamy ustawienia hasłem a następnie naciskamy przycisk *Start*. Podane hasło musi być zgodne z hasłem wpisanym do przełącznika. W innym przypadku pamięć nie zostanie skopiowana. Po wykonaniu zapisu otrzymamy komunikat o zakończeniu zapisu ustawień.

9.3.3. Status pamięci EEPROM

Opcję kontroli statusu pamięci wywołujemy: **Przełącznik > Pamięć zewnętrzna > Status**. Możemy uczynić nieaktywnymi – przycisk *Zdeaktywuj* – partycje *Program* lub *Ustawienia* lub obie. Po deaktywacji np. partycji *Program* (rys. 9.3.3.2) do przełącznika NEED skopiowane zostaną dane jedynie z partycji *Ustawienia*.



Rys. 9.3.3.1. Okno Aktywne partycje.



Rys. 9.3.3.2. Okno Nieaktywna partycja Program

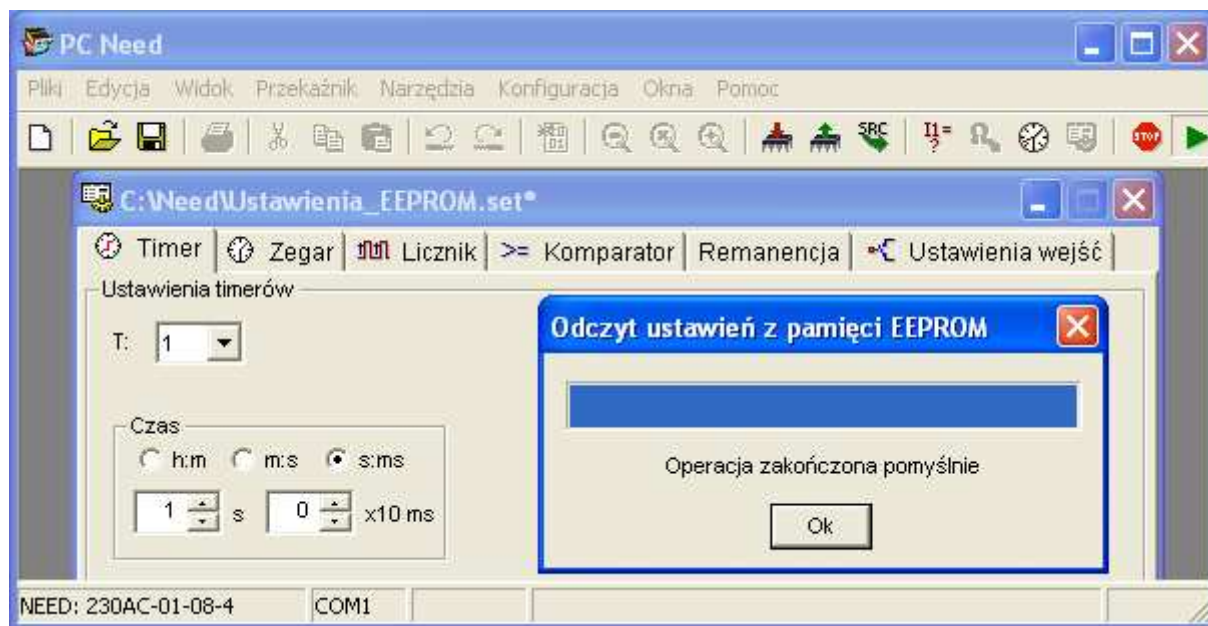


Uwaga: Partycję aktywuje się poprzez wgranie nowej zawartości.

9.3.4. Odczyt ustawień

Możliwy jest odczyt partycji *Ustawienia* z karty pamięci.

W tym celu należy ustawić w obszarze roboczym programu PC Need jako aktywne okno Ustawienia, do którego odczytane dane mają trafić. W przykładzie poniżej utworzony został do tego celu plik *Ustawienia_EEPROM.set*. Po wykonaniu komendy **Przełącznik > Pamięć zewnętrzna > Odczyt** – okno poniżej – dane z partycji zostaną skopiowane do pliku z rozszerzeniem *.set*.



Rys. 9.3.4.1. Okno Odczyt ustawień.

9.4. Współpraca karty pamięci z przełącznikiem NEED



Kartę pamięci należy wkładać przy odłączonym napięciu zasilającym przełącznik NEED.



Na porcie komunikacyjnym dla wersji 115/230V AC mogą występować napięcia niebezpieczne dla zdrowia i życia.



Wkładanie karty przy załączonym zasilaniu przełącznika grozi uszkodzeniem układu pamięci i przełącznika

1. Przy odłączonym zasilaniu należy umieścić moduł karty pamięci w złączu programującym przełącznika NEED.
2. Po załączeniu zasilania modułu przełącznika następuje kopiowanie zawartości pamięci (aktywnych partycji) do pamięci przełącznika NEED. O kopiowaniu informuje migająca na czerwono dioda MODE.
3. Po skopiowaniu zawartości, przełącznik ustawia tryb pracy zależnie od położenia przełącznika trybu pracy. Jeśli ustawiony jest tryb RUN, to przechodzi automatycznie do wykonywania programu.



Uwaga: Zawartość pamięci jest kopiowana jednokrotnie w momencie załączenia zasilania. Po przekopiowaniu moduł pamięci może być usunięty ze złącza przełącznika.



Uwaga: Przepisanie danych z pamięci do sterownika jest możliwe gdy:

- hasło jest prawidłowe,
- pamięć posiada dane dla właściwego typu przekaźnika,
- partycje są aktywne,
- dane w przekaźniku są inne niż w pamięci.



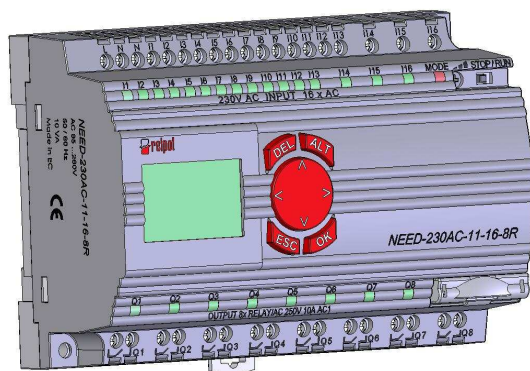
Uwaga: Pamięć zewnętrzna nie umożliwia przenoszenia kodu źródłowego LAD/STL do przekaźnika.

Przekaźniki Need Display umożliwiają skopiowanie aktualnych ustawień na kartę pamięci. Dostęp do tej opcji uzyskuje się poprzez wybranie w menu LCD ekranu Pamięć. Zapis możliwy jest tylko w trybie STOP przekaźnika. Wybranie przycisku OK wymusza przejście w tryb STOP i zapis ustawień, ESC umożliwia opuszczenie ekranu bez zapisu.

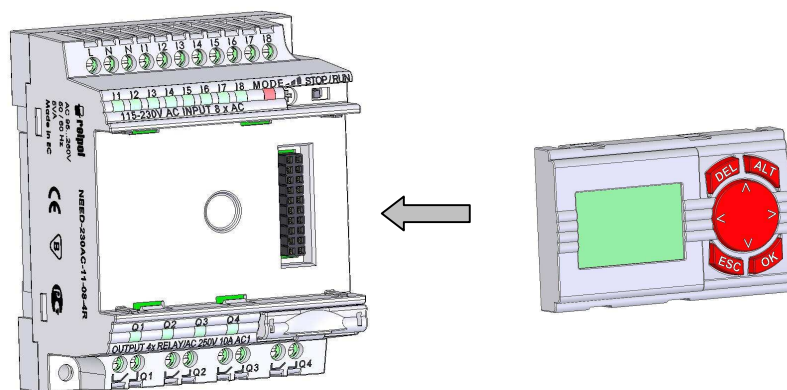
10. WYŚWIETLACZ LCD I KŁAWIATURA

10.1. Informacje wstępne

Przekaźniki NEED-..22-16-8-D posiadają wbudowany wyświetlacz LCD z klawiaturą (rys.10.1.1), a do przekaźnika NEED-..22-08-4 można dołączać zewnętrzny panel wyświetlacza LCD z klawiaturą (rys.10.1.2).

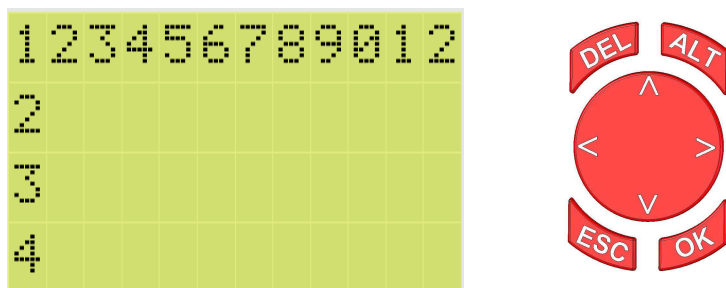


Rys.10.1.1. NEED-..22-16-8-D z wbudowaną klawiaturą.



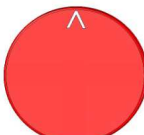
Rys.10.1.2. NEED-..22-08-4-D z dołączaną klawiaturą.

Wyświetlacz może pomieścić 4 linie po 12 znaków. Klawiaturę natomiast stanowi 8 przycisków.



Rys.10.1.3. Wyświetlacz tekstowy 4x12 znaków i klawiatura.

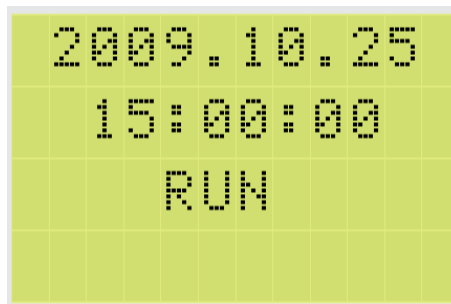
Tabela 10.1.1. Funkcje przycisków

Strzałka w lewo 	1. Przesunięcie kursora o jeden znak w lewo lub powrót do poprzedniego pola do edycji.
Strzałka w prawo 	2. Przesunięcie kursora o jeden znak w prawo lub skok do następnego pola do edycji.
Strzałka do góry 	1. Przesunięcie kursora do góry. 2. Zwiększenie wartości wybranej cyfry (wartości progowe czasów np. <i>Timerów</i>, <i>Liczników</i>). 3. Zmiana typu lub parametru zasobu (np. <i>Licznika</i> na <i>Timer</i>, <i>Zegar</i>, zmiana kanału <i>Zegara</i>, rodzaju komparacji itp.) 4. Zmiana wartości binarnej na przeciwną (stan Wyjść, <i>Merkerów</i>)
Strzałka w dół 	1. Przesunięcie kursora do dołu. 2. Zmniejszenie wartości wybranej cyfry (wartości progowe czasów np. <i>Timerów</i>, <i>Liczników</i>). 3. Zmiana typu lub parametru zasobu (np. <i>Licznika</i> na <i>Timer</i>, <i>Zegar</i>, zmiana kanału <i>Zegara</i>, rodzaju komparacji itp.) 4. Zmiana wartości binarnej na przeciwną (stan Wyjść, <i>Merkerów</i>)
Klawisz „OK.” 	1. Wejście w następny poziom menu danego parametru. 2. Zatwierdzenie zmiany wartości danego parametru, zapisanie.
Klawisz „ESC”	1. Wyjście z menu danego parametru (do poziomu sprzed ostatniego OK.). 2. Anulowanie zmiany danego parametru.

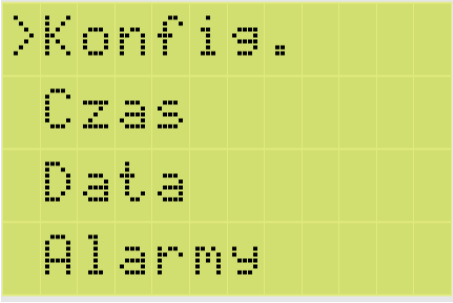
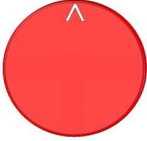
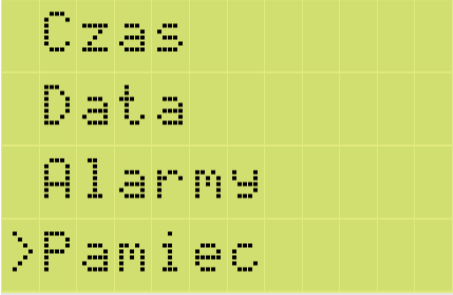
	3. Powrót do ekranu alarmu.
Klawisz „DEL” 	Nie używany
Klawisz ALT 	Przeglądanie aktywnych lub niepotwierdzonych alarmów.

10.2. Menu główne

Po załączeniu zasilania na wyświetlaczu pokazana jest data, czas oraz tryb pracy przekaźnika (RUN, STOP).

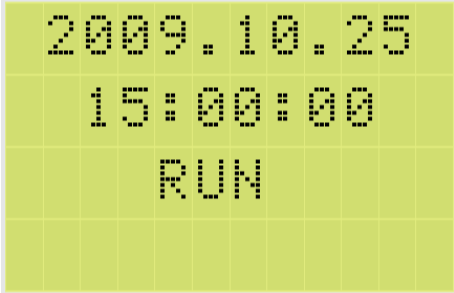
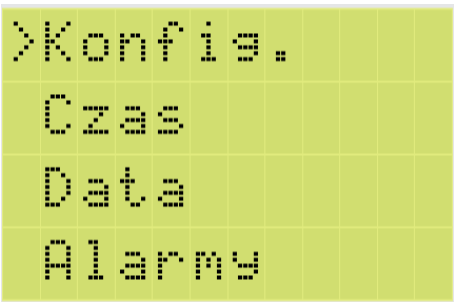


Uwaga: Jeśli przekaźnik jest w trybie RUN (przełącznik trybu pracy na RUN) i wyzwolony jest Alarm (aktywny MTx) to zamiast ekranu głównego wyświetlone będzie ekran alarmu.

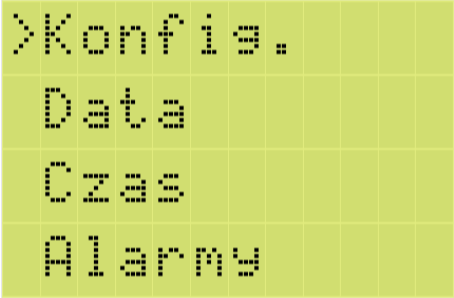
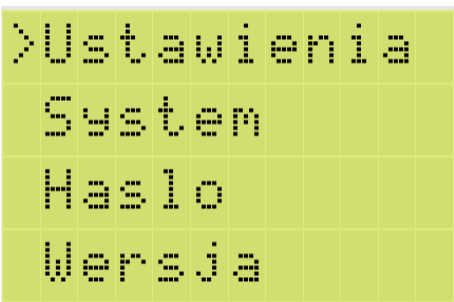
Aby wejść do menu konfiguracyjnego należy nacisnąć przycisk OK 	
Za pomocą klawiszy strzałek góra/dół poruszamy się po menu.  lub 	

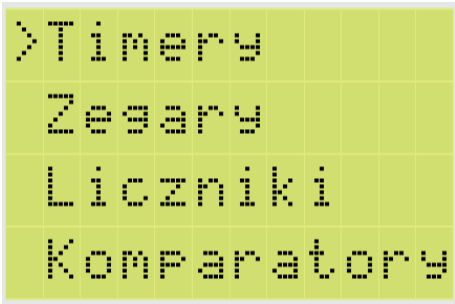
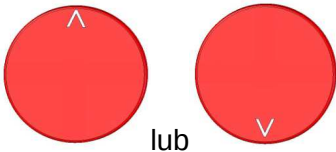
10.3. Menu Konfiguracja

Umożliwia zmiany w ustawieniach programu i zarządzanie przekaźnikiem. W jego skład wchodzi: Ustawienia, System, Hasło, Wersja.

	Ekran po włączeniu zasilania (przełącznik trybu pracy na RUN). 
	Możliwy wybór: >Konfiguracja Czas Data Alarmy Pamięć

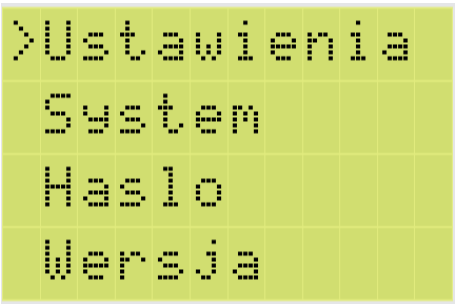
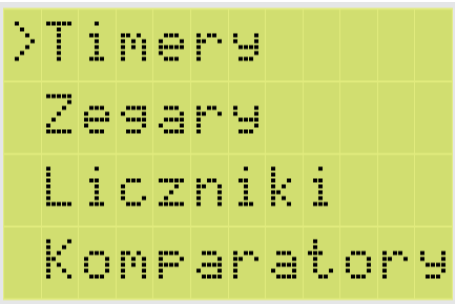
10.3.1. Menu Ustawienia

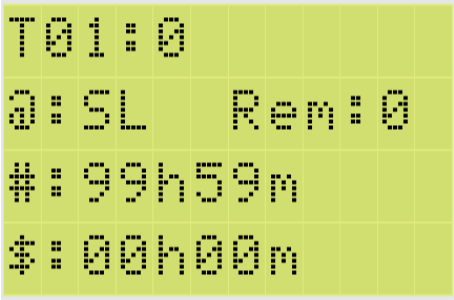
	
	

	Wybór zasobów do zmiany ustawień za pomocą strzałek:  Zasoby do wyboru: > Timery - Zegary - Liczniki - Komparatory - Markery - Wejścia - Wyjścia - Przyciski - Szybki licznik
---	--

10.3.1.1. Timery

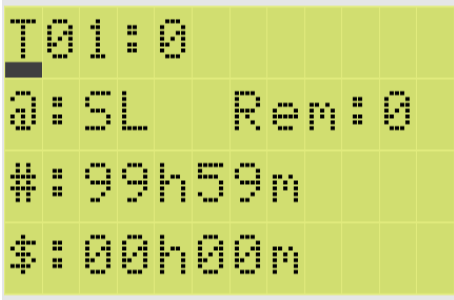
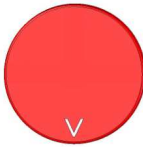
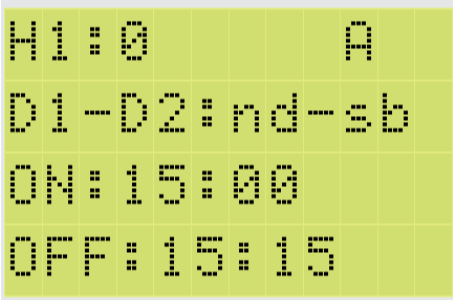
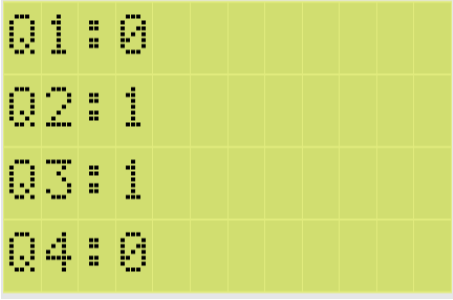
1. Przejście do ekranu konfiguracyjnego *Timery*.

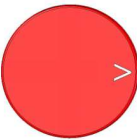
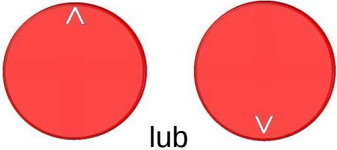
	T – Timer 01 – Nr Timera (01 – 32) : 0 – Stan wyjścia Timera: 1 – Timer załączony 0 – Timer wyłączony @ - Tryb pracy (SE, SD, SL, SF) Rem: – Remanencja (tylko dla T5 – T8) 0 – Remanencja załączona 1 – Remanencja wyłączona # - Wartość czasu do odmierzenia (maks. 99h59min) \$ - Wartość bieżąca czasu
---	---

2. Przejście do innych zasobów.

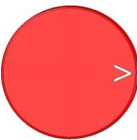
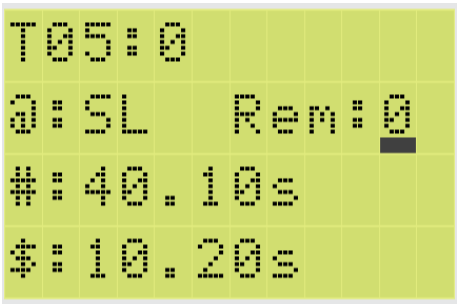
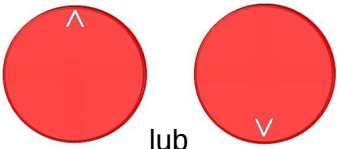
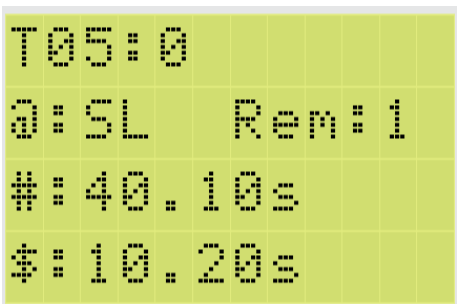
Żeby zmienić zasoby (np. przejść do wyświetlania ekranów dla Liczników, Zegarów itd.) należy ustawić kursor na pierwszym polu wyświetlacza (litera T)

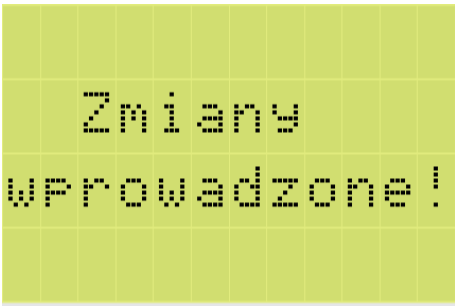
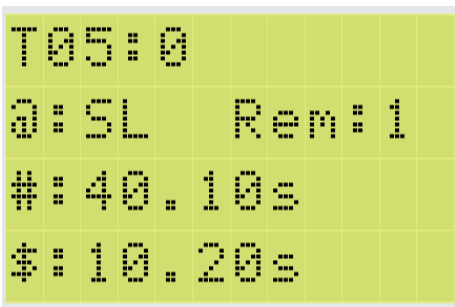
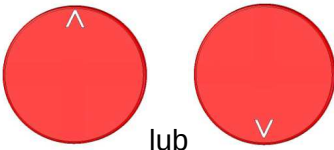
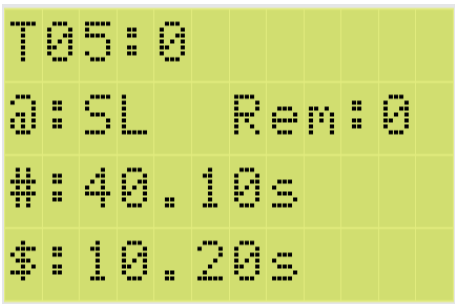
	Za pomocą klawiszy strzałek góra/dół wybrać typ zasobu (pierwszą literę nazwy zasobu)   lub
Naciskając 1 raz strzałkę w dół  przechodzimy do zasobów H - Zegary	
Naciskając 1 raz strzałkę w dół  przechodzimy do zasobów Q - wyjścia	

3. Zmiana numeru *Timera*

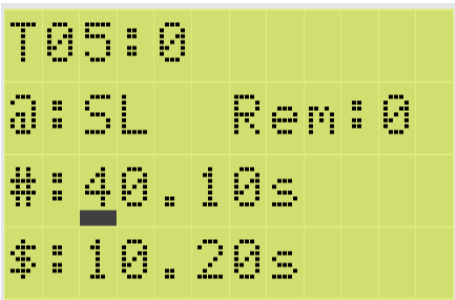
Żeby zmienić numer <i>Timera</i> (np. z T1 na T5) należy: Ustawić kursor na wskazanym obok polu (naciśnięcie przycisku „strzałka w prawo”)	
	Za pomocą klawiszy strzałek góra/dół ustawić nr Timera 

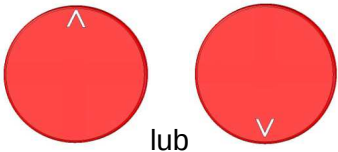
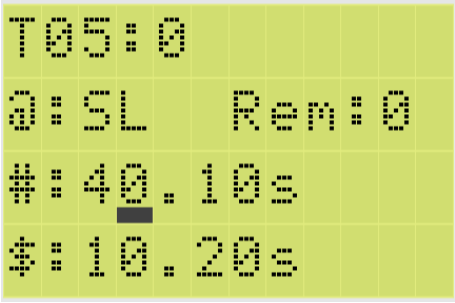
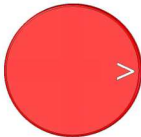
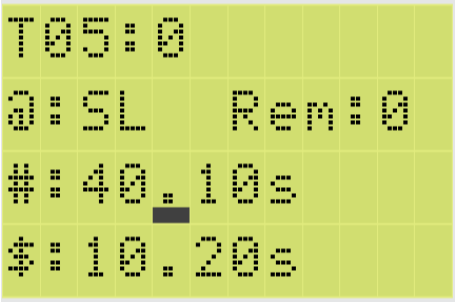
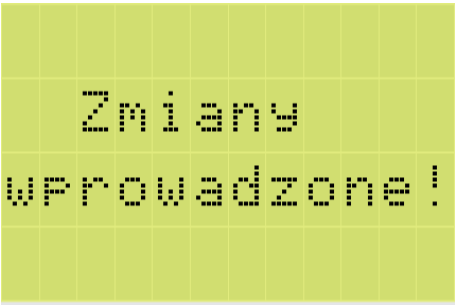
4. Załączenie / Wyłączenie remanencji

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu)	
	Załączenie remanencji Nacisnąć 
	Jeśli remanencja dla Timera T5 była wyłączona (parametr Rem:0), to teraz zostanie ona załączona. Zaakceptować zmiany. 

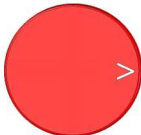
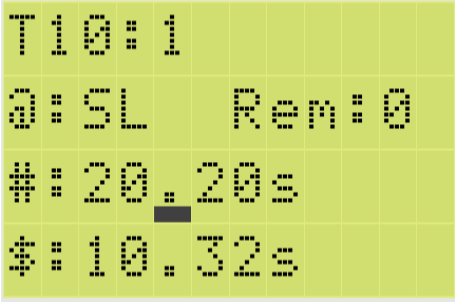
	
	Wyłączenie remanencji Nacisnąć  lub
	Jeśli remanencja dla Timera T5 była załączona (parametr Rem:1), to teraz zostanie ona wyłączona. Zaakceptować zmiany. 
	

5. Ustawienie czasu do odmierzenia.

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	
--	--

Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby wybrać żadaną wartość czasu. 	
Przesuwać kursor na następne pola (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na żdanym polu) i, w podobny sposób jak wyżej, ustawić odpowiednią wartość czasu. 	
Ustawiony czas do odmierzenia dla Timera T5: 40s100ms Zaakceptować zmiany 	

6. Zmiana zakresu czasu do odmierzenia.

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	
---	--

Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół, aby wybrać żądaną wartość zakresu czasu (s - sekundy, m - minuty, h - godziny) lub	
Ustawiony czas do odmierzenia dla Timera T10: 20m20s Zaakceptować zmiany	

10.3.1.2. Menu Zegary

1. Przejsie do ekranu konfiguracyjnego Zegary.

	H – Zegar 1 – Nr Zegara (1 – 8) :0 – Stan wyjścia Zegara: 0 – Zegar wyłączony 1 – Zegar załączony A - Kanał (A, B, C, D) D1-D2: – Dni tygodnia: nd - Niedziela pn – Poniedziałek wt – Wtorek sr – Środa cz – Czwartek pt – Piątek so - Sobota ON: – Godzina załączenia Zegara OFF: – Godzina wyłączenia Zegara

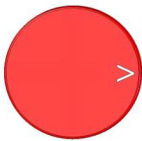
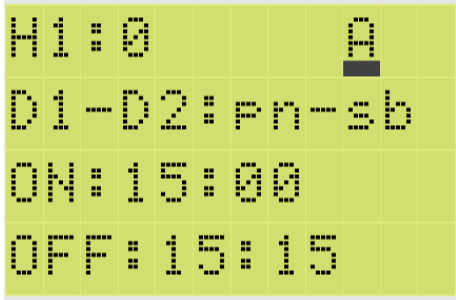
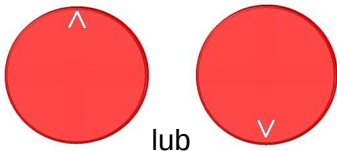
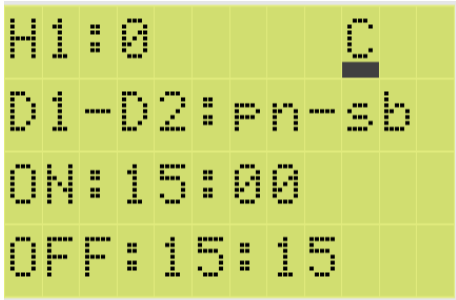
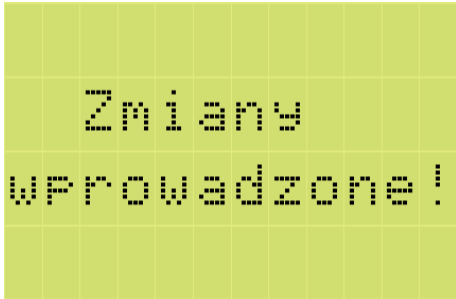
2. Przejście do innych zasobów

Przejście do innych zasobów odbywa się tak samo, jak to pokazano dla Timera.

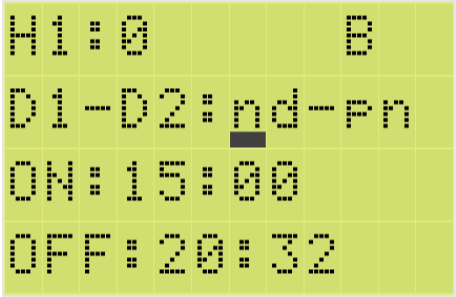
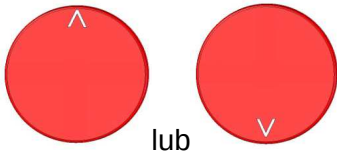
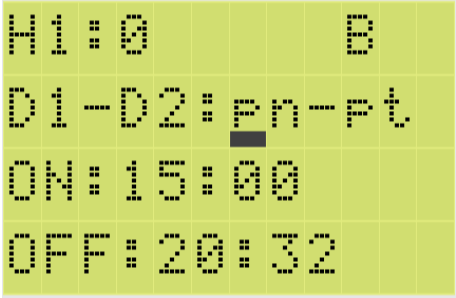
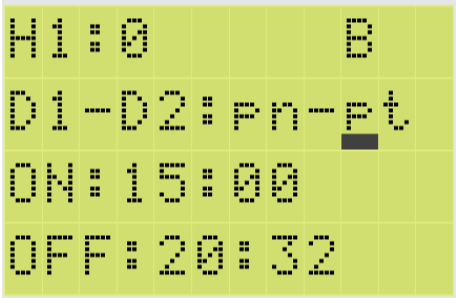
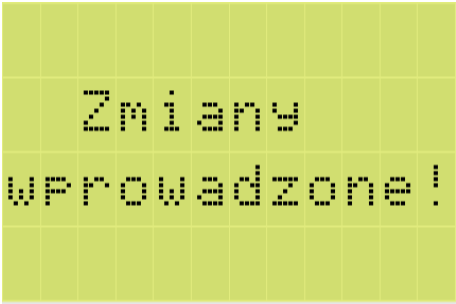
3. Zmiana numeru Zegara

Zmiana numeru Zegara (np. z H1 na H4) odbywa się tak samo, jak to pokazano dla Timera.

4. Zmiana Kanału Zegara

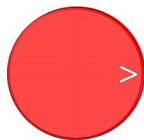
Żeby zmienić kanał Zegara (np. z A na B) należy: Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo, aż do ustawienia kursora na wskazanym polu). 	
Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół, aby wybrać żądany kanał (tutaj np. kanał „C”). 	
Zaakceptować zmiany 	

5. Ustawienie dni załączenia/wyłączenia Zegara (D1 – D2).

Żeby wybrać dni załączenia/wyłączenia wyjścia Zegara (np. od poniedziałku do piątku: "pn – pt") należy: Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo, aż do ustawienia kursora na wskazanym polu 	
Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół, aby wybrać żądany dzień tygodnia (tutaj „pn” poniedziałek).  lub	
Przesunąć kursor na następne pole (nacisnąć przycisk „strzałka w prawo), oraz w analogiczny sposób wybrać drugi dzień załączenia/wyłączenia Zegara (tutaj piątek – „pt”). 	
Zaakceptować zmiany	

6. Ustawienie godziny załączenia Zegara (ON).

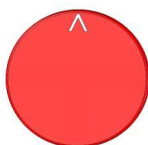
Żeby wybrać godziny załączenia wyjścia Zegara, należy ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo, aż do ustawienia kursora na żądanym polu)



```

H1:0      B
D1-D2:Fn-Ft
ON:15:00
OFF:20:32
  
```

Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół, aby wybrać żądaną wartość czasu .



lub



```

H1:0      B
D1-D2:Fn-Ft
ON:05:00
OFF:20:32
  
```

Przesunąć kursor na następne pole godziny załączenia ON (nacisnąć przycisk „strzałka w prawo”) oraz w analogiczny sposób ustawić pozostałe pola godziny.

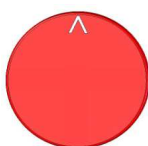


lub



```

H1:0      B
D1-D2:Fn-Ft
ON:02:00
OFF:20:32
  
```

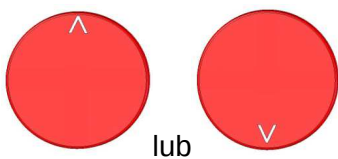
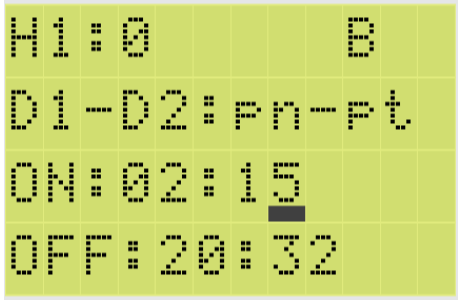
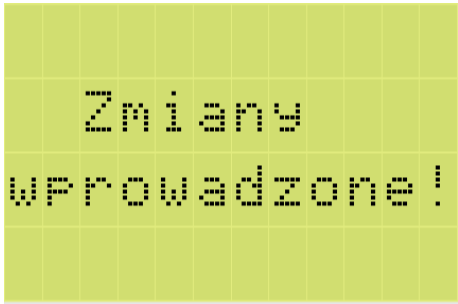


lub



```

H1:0      B
D1-D2:Fn-Ft
ON:02:10
OFF:20:32
  
```

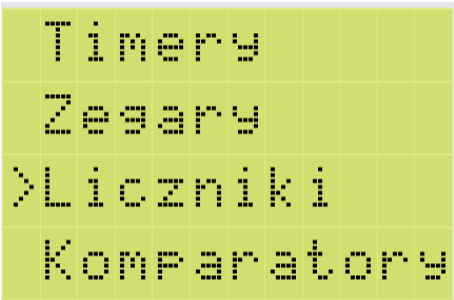
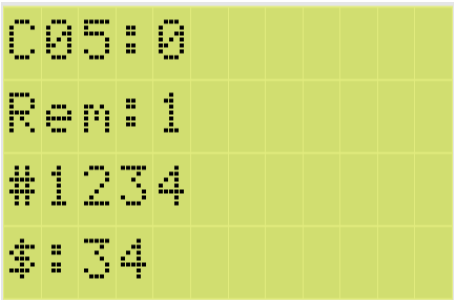
	
Ustawiona godzina załączenia Zegara H1, dla kanału B: 02:15.	
6.4. Zaakceptować zmiany 	

7. Ustawienie godziny wyłączenia Zegara (OFF).

Ustawienie godziny wyłączenia Zegara (edycja OFF) odbywa się w analogiczny sposób jak dla godziny załączenia.

10.3.1.3. Menu Liczniki

1. Przejście do ekranu konfiguracyjnego Liczniki.

	
	C – Licznik 05 – Nr Licznika (1 – 8) :0 – Stan wyjścia Licznika: 0 – Wyjście Licznika wyłączona 1 – Wyjście Licznika załączona Rem: – Remanencja 0 – Remanencja załączona 1 – Remanencja wyłączona #: – Próg Licznika \$: – Wartość bieżąca Licznika

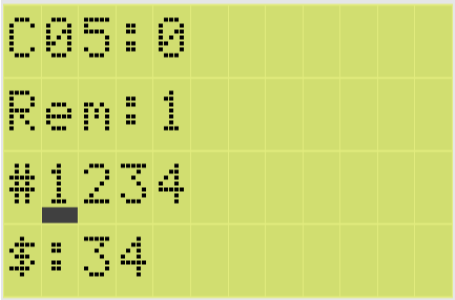
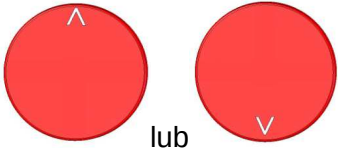
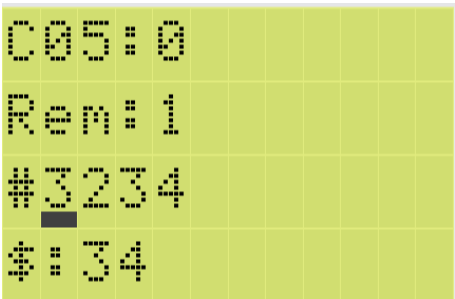
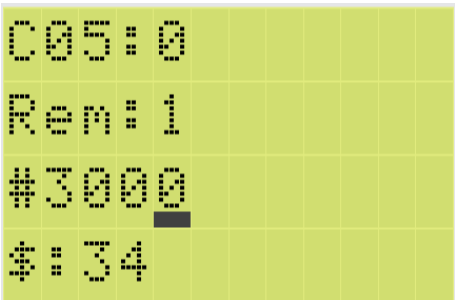
2. Przejście do innych zasobów

Przejście do innych zasobów odbywa się tak samo, jak to pokazano dla Timera.

3. Zmiana numeru Licznika

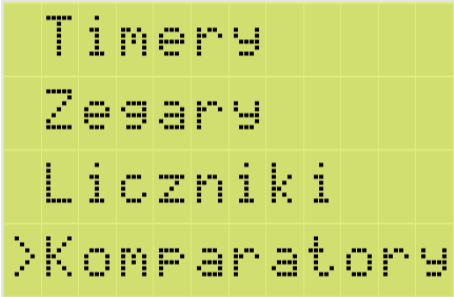
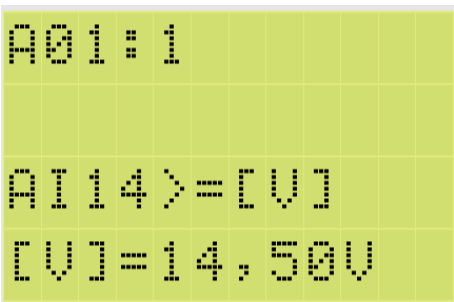
Zmiana numeru Licznika (np. z C5 na C8) odbywa się tak samo, jak to pokazano dla Timera.

4. Ustawienie progu Licznika.

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	
Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby wybrać żądaną wartość nastawianego progu. 	
Przesuwać kursor na następne pola (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na żądanym polu) i, w podobny sposób jak wyżej, ustawić odpowiednią wartość. 	
Ustawiony próg zliczania dla Licznika C05 to 3000. Zaakceptować zmiany 	

10.3.1.4. Menu Komparatory

1. Przejście do ekranu konfiguracyjnego Komparatory.

	
	A – Komparator 01 – Nr Komparatora (1 – 16) :0 – Stan wyjścia Komparatora: 0 – Wyjście Komparatora wyłączone 1 – Wyjście Komparatora załączone AI14>=[V] – Typ porównania [V]=14,50 – Wartość ustawionego progu – jeśli napięcie na AI14 będzie większe lub równe 14,50V, to wyjście Komparatora zostanie ustawione w stan wysoki.

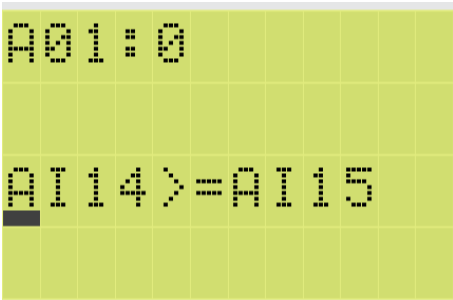
2. Przejście do innych zasobów

Przejście do innych zasobów odbywa się tak samo, jak to pokazano dla Timera.

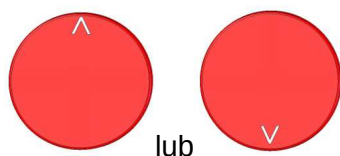
3. Zmiana numeru Komparatora

Zmiana numeru Komparatora (np. z A1 na A2) odbywa się tak samo, jak to pokazano dla Timera.

4. Wybór relacji Komparatora.

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	
--	--

Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby wybrać żądaną wartość relacji Komparatora (patrz „5. Możliwe porównania dla Komparatorów”).

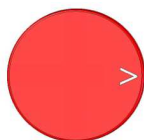


A01:0

AI16>=[V]

[V]=0V

Przesuwać kursor na następne pola (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na żądanym polu) i, w podobny sposób jak wyżej, ustawić odpowiednią wartość.

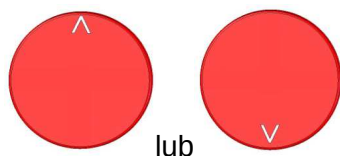


A01:0

AI16>=[V]

[V]=0V

Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby ustawić żądaną wartość wzorcowa wybranej relacji Komparatora (np.120V)



A01:0

AI16>=[V]

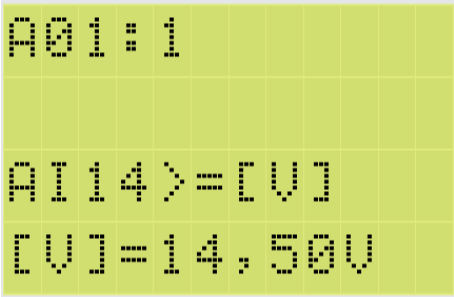
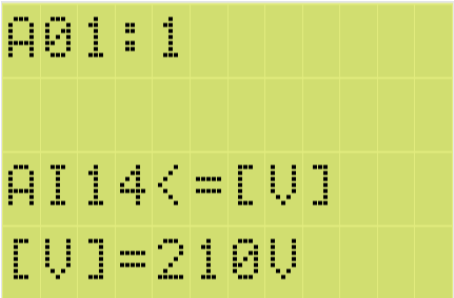
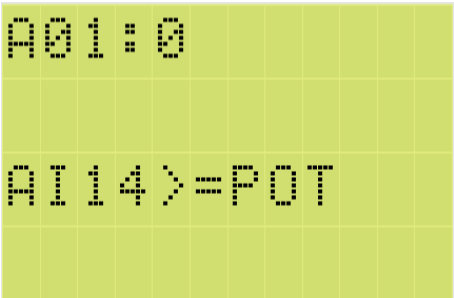
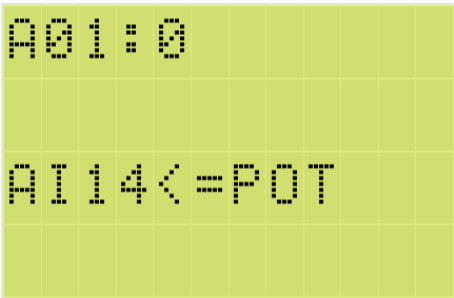
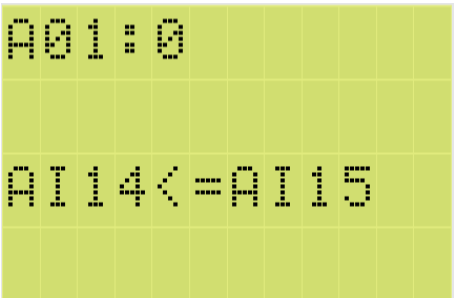
[V]=120V

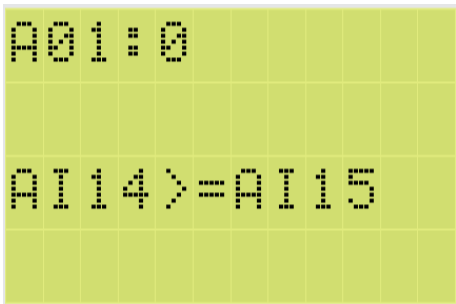
Zaakceptować zmiany



Zmiany
Wprowadzone!

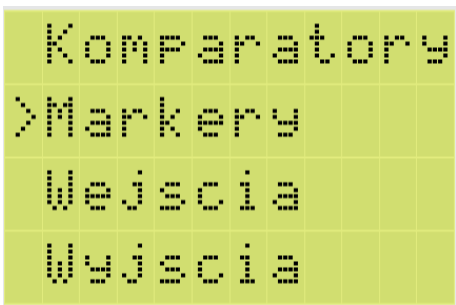
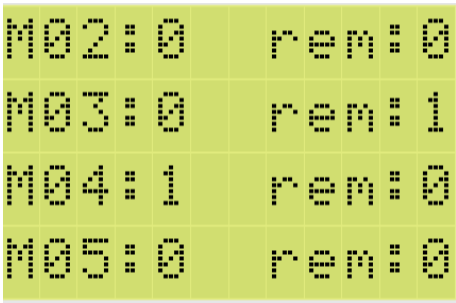
5. Możliwe porównania dla Komparatorów.

	1. AI14>=[V] (AI15>=[V], AI16>=[V]) np. [V]=14.50V Jeśli napięcie na AI14 (AI15, AI16) będzie większe lub równe 14,50V, to wyjście Komparatora zostanie ustawione w stan wysoki.
	2. AI14<=[V] (AI15<=[V], AI16<=[V]) np. [V]=210V Jeśli napięcie na AI14 (AI15, AI16) będzie mniejsze lub równe 210V, to wyjście Komparatora zostanie ustawione w stan wysoki.
	3. AI14>=POT (AI15>=POT, AI16>=POT) Jeśli napięcie na AI14 (AI15, AI16) będzie większe lub równe od wartości wzorcowej ustawionej za pomocą wbudowanego Potencjometru, to wyjście Komparatora zostanie ustawione w stan wysoki.
	4. AI14<=POT (AI15<=POT, AI16<=POT) Jeśli napięcie na AI14 (AI15, AI16) będzie mniejsze lub równe od wartości wzorcowej ustawionej za pomocą wbudowanego Potencjometru, to wyjście Komparatora zostanie ustawione w stan wysoki
	5. AI14<=AI15 (AI14<=AI16, AI15<=AI14, AI15<=AI16, AI16<=AI14, AI16<=AI15) Jeśli napięcie na AI14 będzie mniejsze lub równe od napięcia na AI15, to wyjście Komparatora zostanie ustawione w stan wysoki.

	6. AI14>=AI15 (AI14>=AI16, AI15>=AI14, AI15>=AI16, AI16>=AI14, AI16>=AI15) Jeśli napięcie na AI14 będzie większe lub równe od napięcia na AI15, to wyjście Komparatora zostanie ustawione w stan wysoki.
---	---

10.3.1.5. Menu Markery

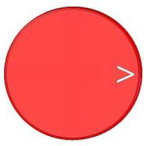
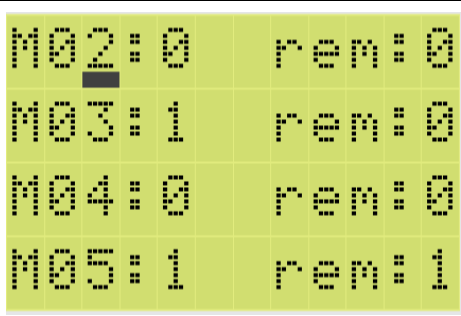
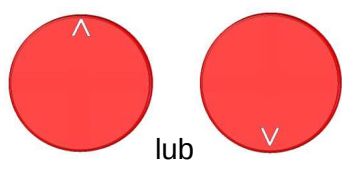
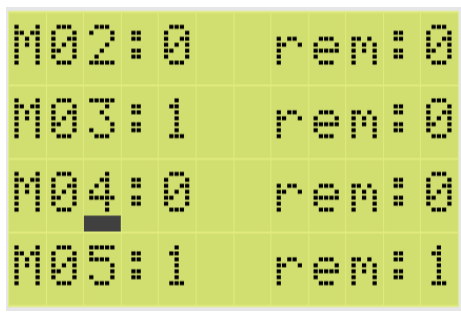
1. Przejście do ekranu konfiguracyjnego Markery.

	
	M – Marker 02 – Nr Markera (1 – 64) :0 – Stan wyjścia Markera: 0 – Wyjście Markera wyłączzone 1 – Wyjście Markera załączone Rem: – Remanencja 0 – Remanencja załączona 1 – Remanencja wyłączona

2. Przejście do innych zasobów

Przejście do innych zasobów odbywa się tak samo, jak to pokazano dla Timera.

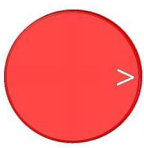
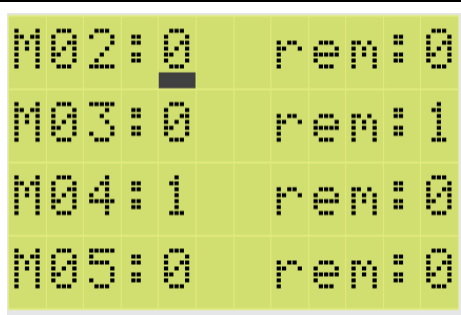
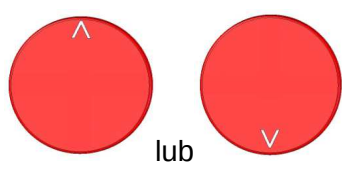
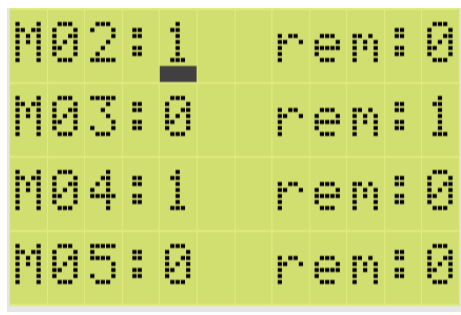
3. Zmiana numeru Markera

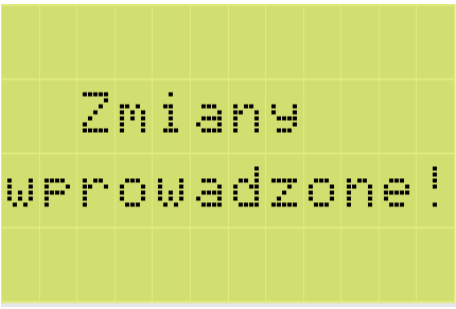
Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	 <pre> M02: 0 rem: 0 M03: 1 rem: 0 M04: 0 rem: 0 M05: 1 rem: 1 </pre>
Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby przejść do kolejnych Markerów. 	 <pre> M02: 0 rem: 0 M03: 1 rem: 0 M04: 0 rem: 0 M05: 1 rem: 1 </pre>

4. Załączenie / Wyłączenie remanencji.

Załączenie/Wyłączenie remanencji dla Markerów M1-M16 odbywa się tak samo, jak to pokazano dla Timera.

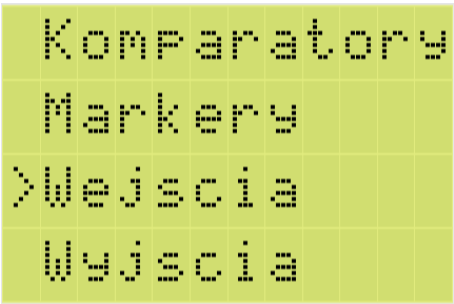
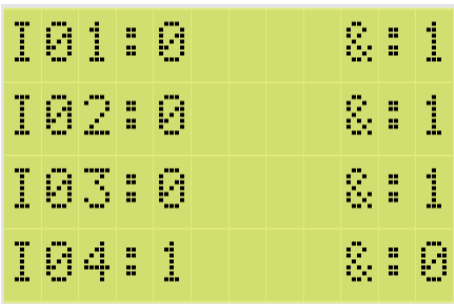
5. Ustawienie wyjścia Markera.

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	 <pre> M02: 0 rem: 0 M03: 0 rem: 1 M04: 1 rem: 0 M05: 0 rem: 0 </pre>
Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby załączyć/wyłączyć wyjście Markera. 	 <pre> M02: 1 rem: 0 M03: 0 rem: 1 M04: 1 rem: 0 M05: 0 rem: 0 </pre>

Zaakceptować zmiany 	
--	--

10.3.1.6. Menu Wejścia

1. Przejście do ekranu konfiguracyjnego Wejścia.

	
	I – Wejście 01 – Nr Wejścia (1 – 8 lub 1-16) :0 – Stan Wejścia: 0 – Na wejściu stan niski 1 – Na wejściu stan wysoki &: – Opóźnienie wejścia 1 – Opóźnienie załączone 0 – Opóźnienie wyłączzone

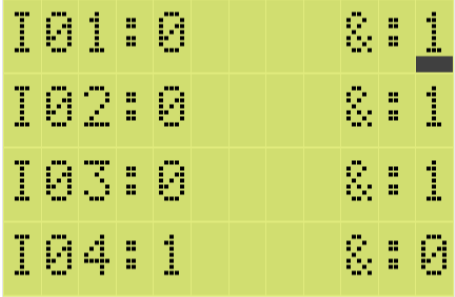
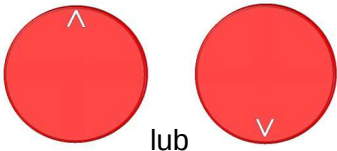
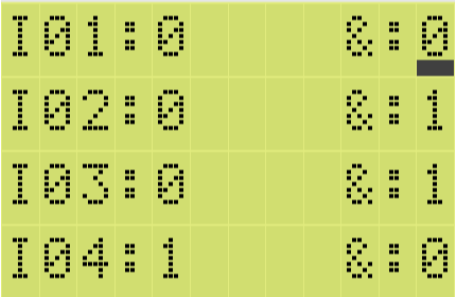
2. Przejście do innych zasobów

Przejście do innych zasobów odbywa się tak samo, jak to pokazano dla Timera.

3. Zmiana numeru Wejścia

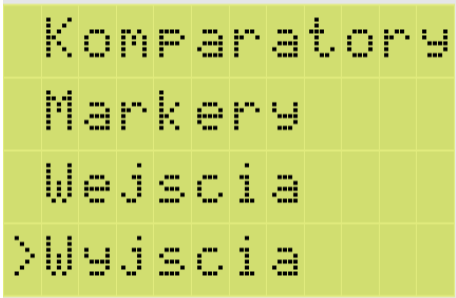
Zmiana numeru Wejścia (np. z I1 na I10) odbywa się tak samo, jak to pokazano dla Markerów.

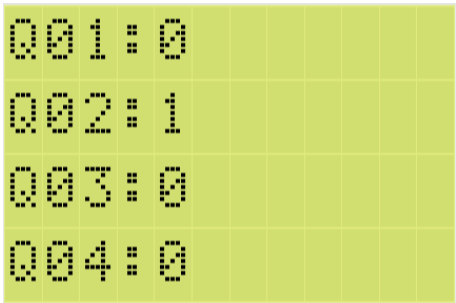
4. Załączenie / Wyłączenie opóźnienia.

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	 <pre> I01:0 &:1 I02:0 &:1 I03:0 &:1 I04:1 &:0 </pre>
Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby załączyć/wyłączyć opóźnienie wejścia. 	 <pre> I01:0 &:0 I02:0 &:1 I03:0 &:1 I04:1 &:0 </pre>
Zaakceptować zmiany 	 <pre> Zmiany wprowadzone! </pre>

10.3.1.7. Menu Wyjścia

1. Przejście do ekranu konfiguracyjnego Wyjścia.

 <pre> Komparatory Markery Wejscia >Wyjscia </pre>	
--	---

	Q – Wyjście 01 – Nr Wyjścia (1 – 4 lub 1-8) :0 – Stan Wyjścia: 0 – Na wyjściu stan niski 1 – Na wyjściu stan wysoki
---	--

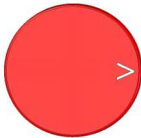
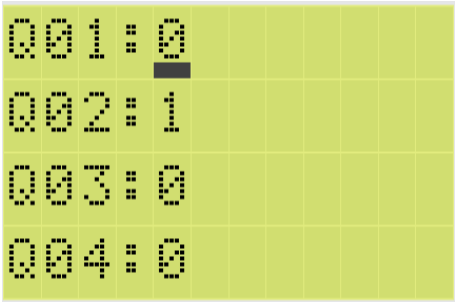
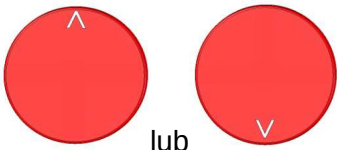
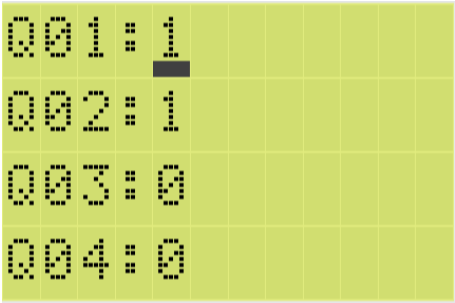
2. Przejście do innych zasobów

Przejście do innych zasobów odbywa się tak samo, jak to pokazano dla Timera.

3. Zmiana numeru Wyjścia

Zmiana numeru Wyjścia (np. z Q1 na Q7) odbywa się tak samo, jak to pokazano dla Timera.

4. Załączenie / Wyłączenie Wyjścia.

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	
Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby załączyć/wyłączyć wyjście.  lub	
Zaakceptować zmiany 	



Jeżeli w programie LAD/STL stan wyjścia jest wymuszany, to stan danego wyjścia nie zostanie zmieniony.

10.3.1.8. Menu Przyciski

1. Przejście do ekranu konfiguracyjnego Przyciski.

	B – Przycisk 1 – Nr Przycisku :0 – Stan Przycisku: 0 – Przycisk naciśnięty 1 – Przycisk zwolniony

2. Przejście do innych zasobów

Przejście do innych zasobów tylko poprzez naciśnięcie przycisku ESC.

10.3.1.9. Menu Szybki Licznik

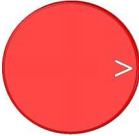
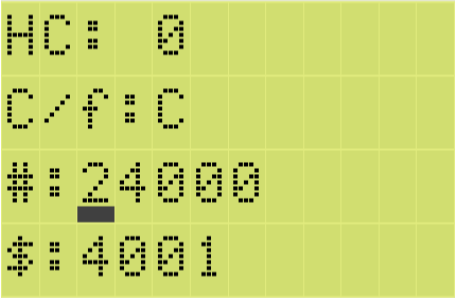
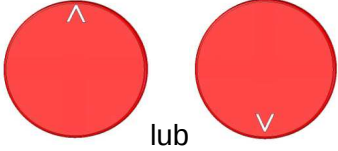
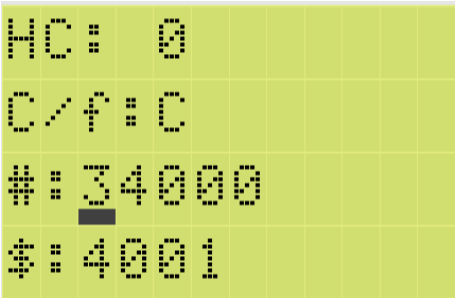
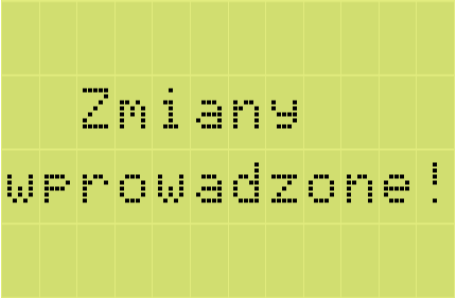
1. Przejście do ekranu konfiguracyjnego Szybki Licznik.

	HC – Szybki Licznik :0 – Stan wyjścia Szybkiego Licznika: 0 – Wyjście Licznika wyłączone 1 – Wyjście Licznika załączone C/f – Licznik / Miernik Częstotliwości #: – Próg Szybkiego Licznika \$: – Wartość bieżąca Szybkiego Licznika

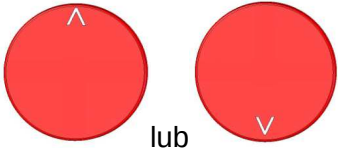
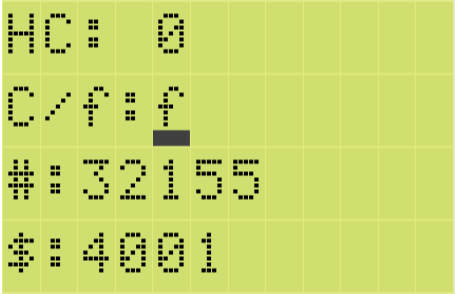
2. Przejście do innych zasobów

Przejście do innych zasobów odbywa się tak samo, jak to pokazano dla Timera.

3. Ustawienie wartości do zliczania przez Szybki Licznik.

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	
Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby wybrać żadaną wartość nastawianego progu. 	
Przesuwać kursor na następne pola (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na żdanym polu) i, w podobny sposób jak wyżej, ustawić odpowiednią wartość. 	
Ustawiony próg zliczania dla Szybkiego Licznika HC to 32155. Zaakceptować zmiany 	

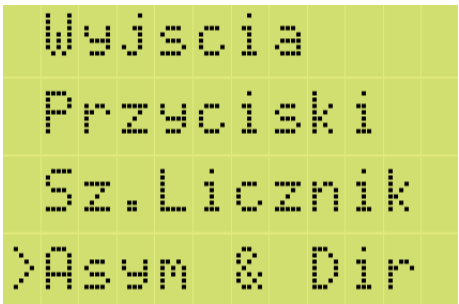
4. Ustawienie trybu pracy: pomiar częstotliwości lub zliczanie impulsów

Ustawić kursor na wskazanym obok polu (naciskanie przycisku „strzałka w prawo”, aż do ustawienia kursora na wskazanym polu) 	
Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół”, aby wybrać żądany tryb pracy: C - zliczanie impulsów, f – pomiar częstotliwości. 	
Ustawiony tryb pracy: pomiar częstotliwości. Zaakceptować zmiany 	

10.3.1.10. Menu ASYM & MDIR

(Tylko dla wersji NEED-230AC-11-16-8 oraz NEED-230AC-22-16-8).

1. Przejście do ekranu konfiguracyjnego ASYM & MDIR.

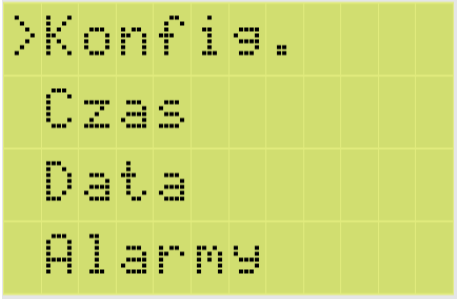
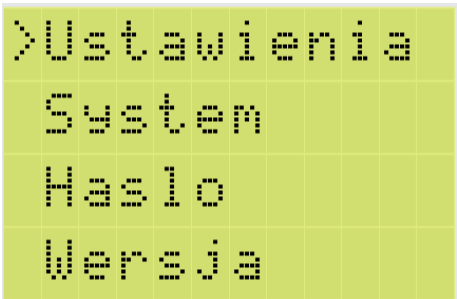
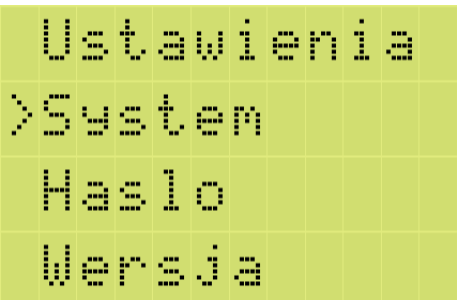
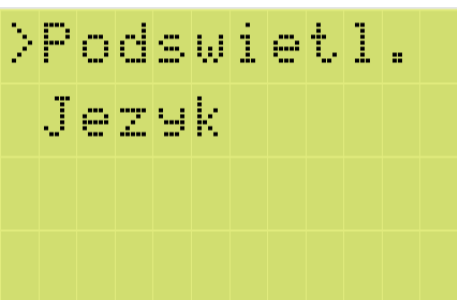
	
	Wartość parametru ASYM jest równa 43V, oraz Marker MDIR ma wartość 1.

10.3.2. Menu System

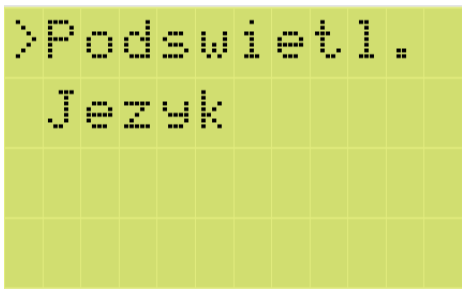
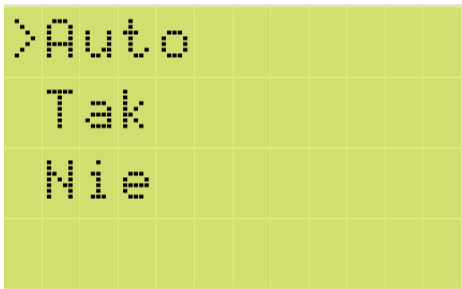
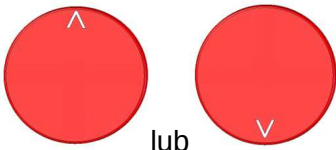
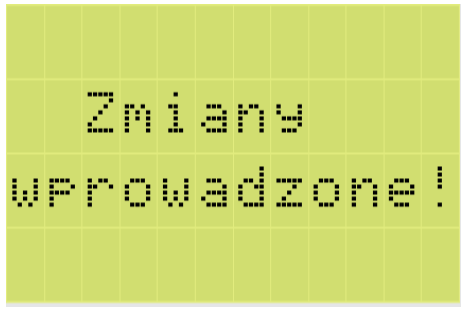
Menu System umożliwia:

- ustawianie sposobu podświetlania wyświetlacza LCD
- wybór języka obsługi modułu LCD.

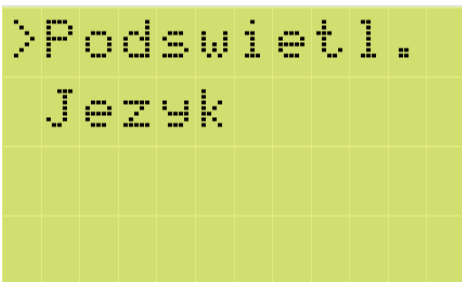
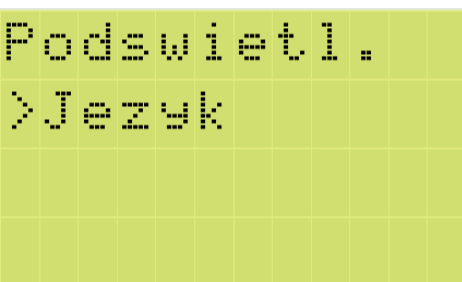
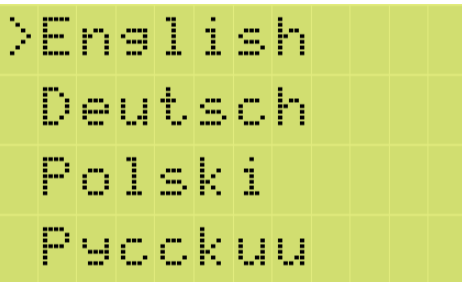
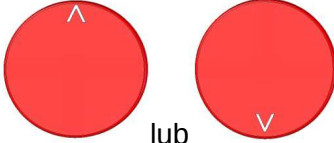
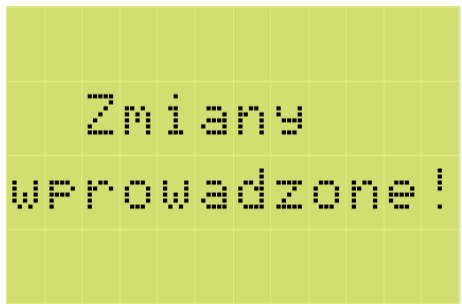
Przejdźcie do Menu System z Menu Konfiguracja

	
	Przejdź o 1 linię niżej za pomocą: 
	
	Menu System: >Podświetl. - ustawianie sposobu podświetlania wyświetlacza LCD >Język - wybór języka obsługi modułu LCD

10.3.2.1. Podświetlenie

	
	Wybrać za pomocą przycisków „strzałka w górę” lub „strzałka w dół” rodzaj podświetlenia  lub
>Auto – podświetlenie automatycznie się wyłącza gdy klawiatura jest nieużywana przez 5s. >Tak – podświetlenie załączone na stałe >Nie – brak podświetlenia	
	

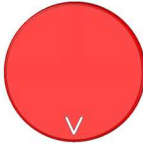
10.3.2.2. Język

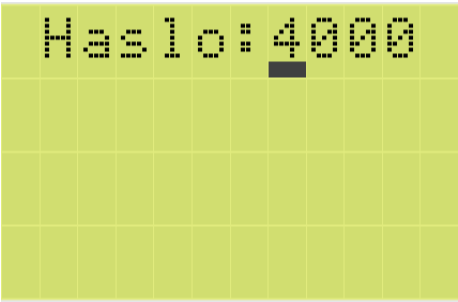
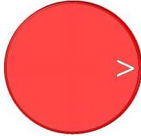
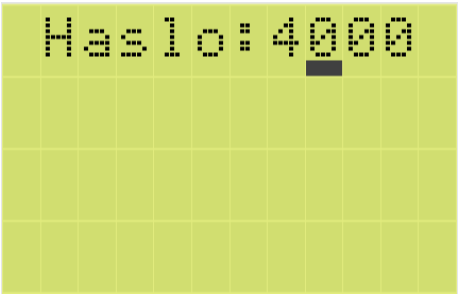
	Przejsć o 1 linię niżej za pomocą: 
	
	Wybrać za pomocą przycisków „strzałka w górę” lub „strzałka w dół” język wyświetlacza 
>English – język angielski. >Deutsch – język niemiecki >Polski – język polski >Русский – język rosyjski	
	

10.3.3. Menu Hasło

<pre>>Konfig. Czas Data Alarmy</pre>	
<pre>>Ustawienia System Hasło Wersja</pre>	Przejsć o 2 linie niżej za pomocą: 
<pre>Ustawienia System >Hasło Wersja</pre>	

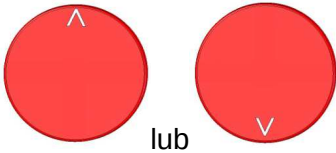
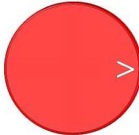
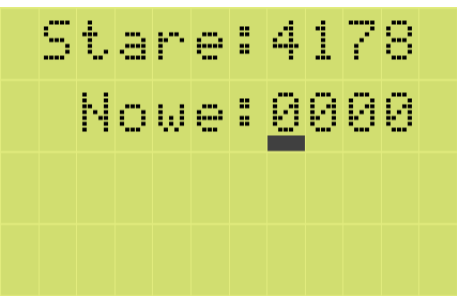
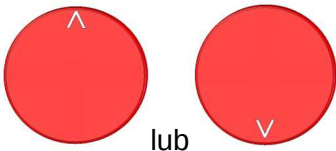
10.3.3.1. Wprowadzanie hasła

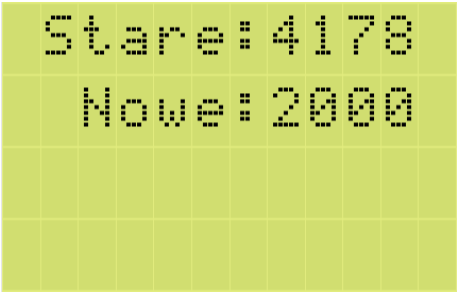
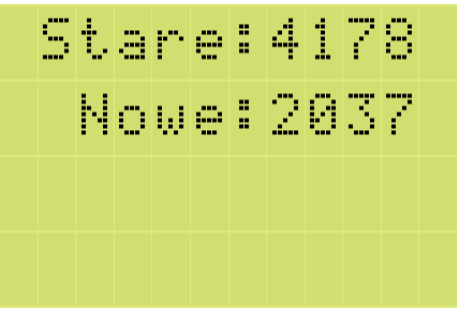
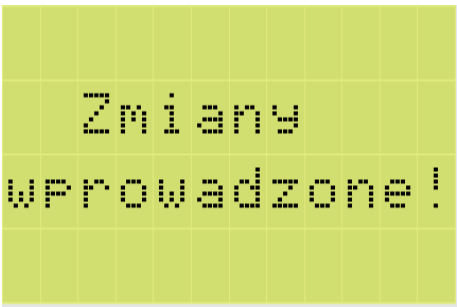
<pre>>Wprowadz Zmiana</pre>	
<pre>Hasło: 0000</pre>	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” 1 cyfrę hasła  lub 

	Ustawić kursor na następną cyfrę hasła (naciśnąć przycisk „strzałka w prawo”) 
	Ustawić kursor na następną cyfrę hasła (naciśnąć przycisk „strzałka w prawo”) 
Ustawić 2-ą i następne cyfry hasła.	Zatwierdzić ustawione hasło: 
	

10.3.3.2. Zmiana hasła

	Przejsć o 1 linię niżej za pomocą: 
---	---

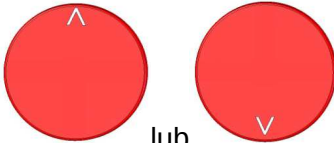
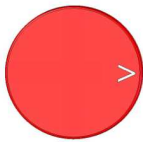
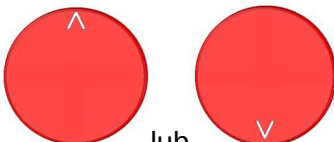
	
	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” 1 cyfrę starego hasła 
	Ustawić kursor na następną cyfrę hasła (naciśnąć przycisk „strzałka w prawo”) 
Ustawić 2, 3 i 4 cyfrę starego hasła (podobnie jak dla 1 cyfry)	Przejsć do wpisywania nowego hasła - nacisnąć przycisk „strzałka w prawo” 
	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” 1 cyfrę nowego hasła 

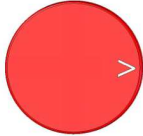
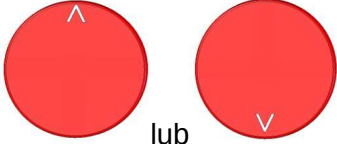
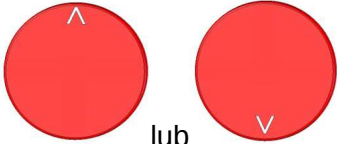
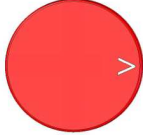
	Ustawić 2, 3 i 4 cyfrę nowego hasła (podobnie jak dla 1 cyfry)
	Zatwierdzić ustawione hasła: 
	

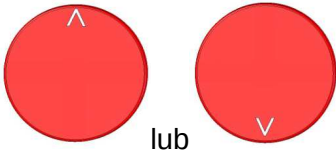
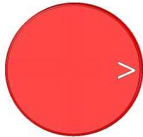
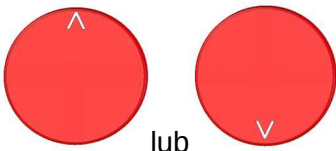
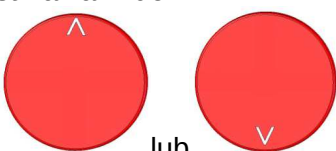
10.3.4. Menu Wersja

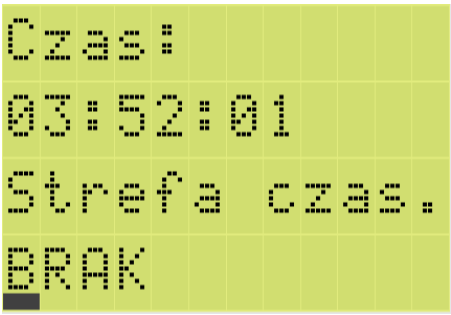
<pre>>Konfig. Czas Data Alarmy</pre>	
<pre>>Ustawienia System Hasło Wersja</pre>	Przejsć o 3 linie niżej za pomocą: 
<pre> Ustawienia System Hasło >Wersja</pre>	
<pre>NEED MAX DISPLAY 24DC-22-16 -8 v1.00</pre>	Informacja o typie i wersji oprogramowania przekaźnika.

10.4. Menu Czas

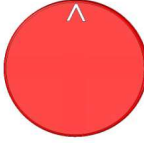
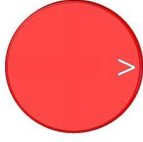
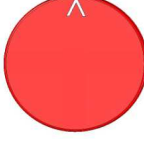
<pre>>Konfig. Czas Data Alarmy</pre>	Przejsć o 1 linie niżej za pomocą: 
<pre>Konfig. >Czas Data Alarmy</pre>	
<pre>Czas: 14:15:40 Strefa czas. EU o 2:00</pre>	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” 10-tych godzin  lub
<pre>Czas: 04:15:40 Strefa czas. EU o 2:00</pre>	Ustawić kursor na następną cyfrę - godziny (naciśnąć przycisk „strzałka w prawo”) 
<pre>Czas: 04:15:40 Strefa czas. EU o 2:00</pre>	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” godzinę  lub

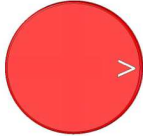
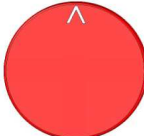
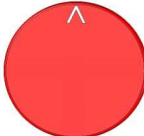
Czas: 03:15:40 Strefa czas. EU o 2:00	Ustawić kursor na następnym polu – 10-ty minut (naciśnąć przycisk „strzałka w prawo”) 
Czas: 03:15:40 Strefa czas. EU o 2:00	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” 10-ty minut  lub
Czas: 03:55:40 Strefa czas. EU o 2:00	Ustawić kursor na następną cyfrę – minuty (naciśnąć przycisk „strzałka w prawo”) 
Czas: 03:55:40 Strefa czas. EU o 2:00	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” minutę  lub
Czas: 03:52:40 Strefa czas. EU o 2:00	Ustawić kursor na następnym polu – 10-ty sekund (naciśnąć przycisk „strzałka w prawo”) 

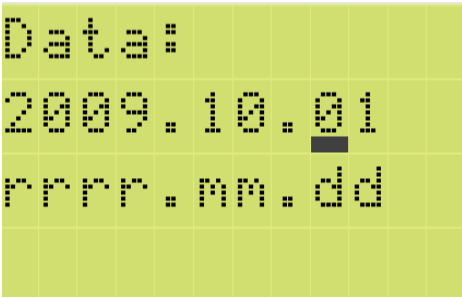
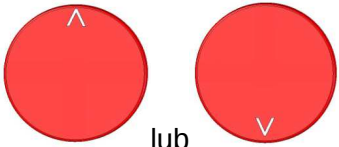
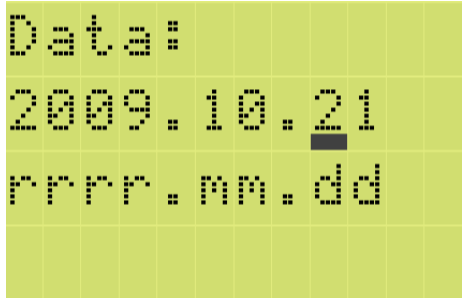
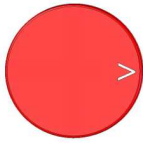
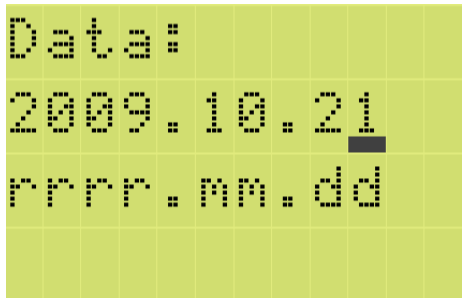
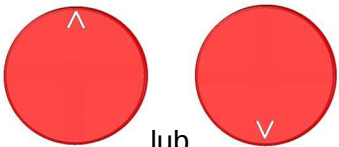
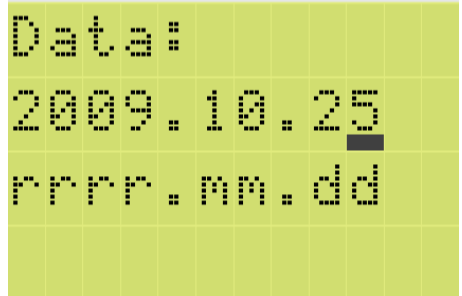
Czas: 03:52:40 Strefa czas. EU o 2:00	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” 10-ty sekund 
Czas: 03:52:00 Strefa czas. EU o 2:00	Ustawić kursor na następną cyfrę – sekundy (naciśnąć przycisk „strzałka w prawo”) 
Czas: 03:52:00 Strefa czas. EU o 2:00	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” sekundę 
Czas: 03:52:01 Strefa czas. EU o 2:00	Ustawić kursor na następnym polu – Strefa czasowa (naciśnąć przycisk „strzałka w prawo”) 
Czas: 03:52:01 Strefa czas. EU o 2:00	Wybrać strefę czasową za pomocą „strzałka w górę” lub „strzałka w dół” 

 Czas: 03:52:01 Strefa czas. BRAK	Strefa czasowa do wyboru: - BRAK - EU o 1:00 - EU o 2:00 - EU o 3:00 - RU - US
	 Zmiany wprowadzone!

10.5. Menu Data

<pre>>Konfig. Czas Data Alarmy</pre>	Przejsć o 2 linie niżej za pomocą: 
<pre>Konfig. Czas >Data Alarmy</pre>	
<pre>Data: 2001.01.01 rrrr.mm.dd</pre>	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” 10-tych lat   lub
<pre>Data: 2001.01.01 rrrr.mm.dd</pre>	Ustawić kursor na następną cyfrę – jednostki lat (naciśnąć przycisk „strzałka w prawo”) 
<pre>Data: 2001.21.01 rrrr.mm.dd</pre>	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” rok   lub

Data: 2009.01.01 rrrr.mm.dd	Ustawić kursor na następnym polu – 10-ty miesiąc (naciśnąć przycisk „strzałka w prawo”) 
Data: 2009.01.01 rrrr.mm.dd	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” 10-ty miesiąc  lub 
Data: 2009.11.01 rrrr.mm.dd	Ustawić kursor na następną cyfrę – jednostki miesiąca (naciśnąć przycisk „strzałka w prawo”) 
Data: 2009.11.01 rrrr.mm.dd	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” miesiąc  lub 
Data: 2009.10.01 rrrr.mm.dd	Ustawić kursor na następnym polu – 10-ty dni (naciśnąć przycisk „strzałka w prawo”) 

	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” 10-tych dni  lub
	Ustawić kursor na następną cyfrę – dzień (naciśnąć przycisk „strzałka w prawo”) 
	Ustawić za pomocą przycisków „strzałka w górę” lub „strzałka w dół” dzień  lub
	Zatwierdzenie wprowadzonych zmian 
	

10.6. Menu Alarmy

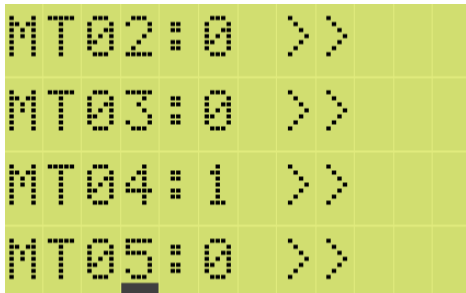
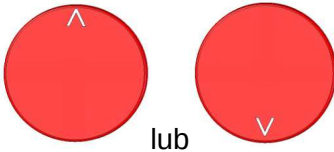
Podgląd stanu, ustawianie Markerów tekstowych, podgląd tekstów alarmów przypisanych markerom tekstowym.

1. Przejście do ekranu konfiguracyjnego Alarmy.

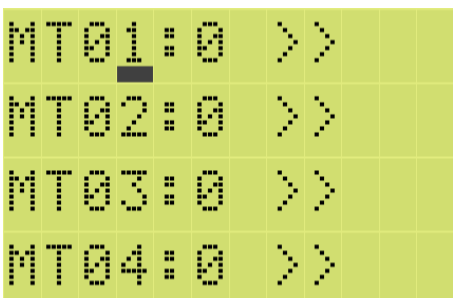
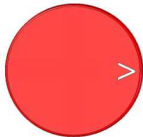
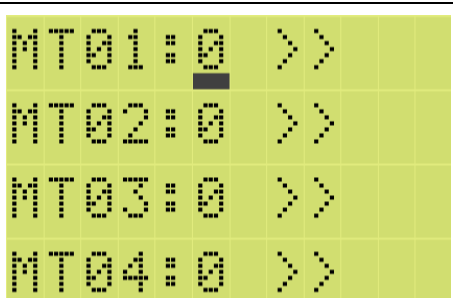
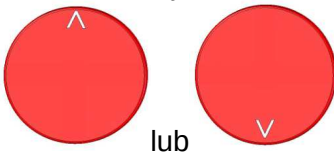
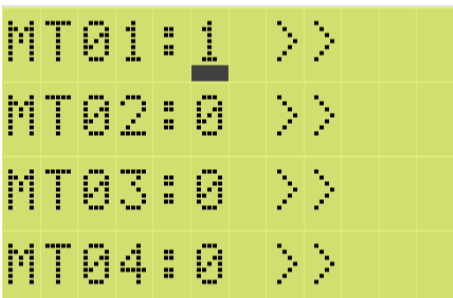
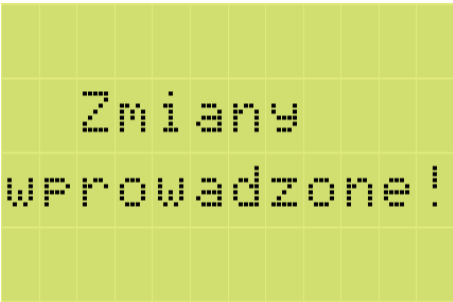
<pre>>Konfig. Czas Data Alarmy</pre>	Przejdź o 3 linie niżej za pomocą: 
<pre>Konfig. Czas Data >Alarmy</pre>	
<pre>MT01:0 >> MT02:0 >> MT03:0 >> MT04:0 >></pre>	Alarmy MTx MT – Marker tekstowy 01 – nr Markera (1 – 8) :0 – stan wyjścia Markera: 0 – Wyjście Markera wyłączzone 1 – Wyjście Markera załączone >> – podgląd tekstu alarmu

2. Zmiana zakresu podglądanych numerów Markerów tekstowych

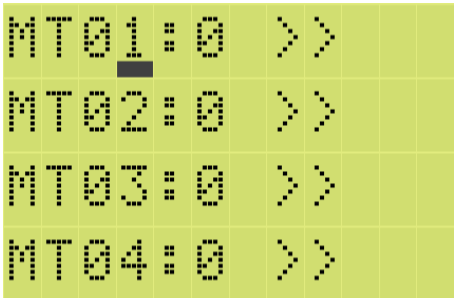
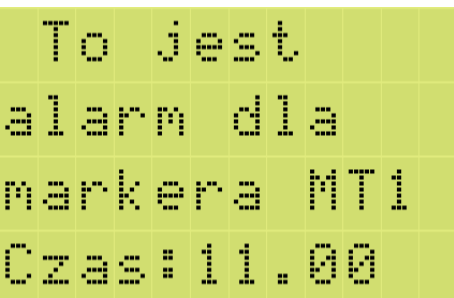
<pre>MT0<u>1</u>:0 >> MT02:0 >> MT03:0 >> MT04:0 >></pre>	Na ekranie jednocześnie wyświetlane są 4 kolejne Markery tekstowe. W celu zmiany zakresu ustawić kursor na polu nr (MT0<u>1</u>) i przejść niżej za pomocą klawisza "strzałka w dół" 
---	---

	Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół, aby ponownie zmienić zakres.  lub
---	--

3. Ustawienie stanu Markera.

	Przenieść kursor z pola nr Markera tekstowego na pole jego stanu (przycisk „strzałka w prawo”) 
	Nacisnąć przycisk „strzałka w górę” lub „strzałka w dół, aby załączyć/wyłączyć Marker  lub
	Zaakceptować zmiany 
	

4. Podgląd tekstu alarmu

	Przenieść kursor na pole podgląd >> (przycisk „strzałka w prawo”) 
	Nacisnąć przycisk „strzałka w górę” aby wyświetlić tekst alarmu przypisany do MTx 
	Nacisnąć klawisz „ESC” aby powrócić do ekranu Alarmy 

10.7. Menu Pamięć zewnętrzna

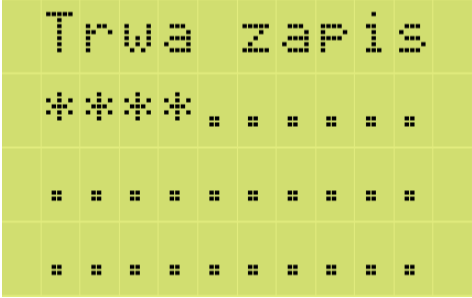
Umożliwia zapis do pamięci zewnętrznej Ustawień z przekaźnika.

1. Przejście do ekranu konfiguracyjnego pamięć.

<pre>>Konfig. Czas Data Alarmy</pre>	Przejdź o 4 linie niżej za pomocą: 
<pre> Czas Data Alarmy >Pamięć</pre>	

2. Obsługa zapisu do pamięci zewnętrznej.

<pre> Brak pamięci!</pre>	Komunikat w przypadku braku pamięci zewnętrznej
<pre>Zapis do Pam. zewn. tylko w trybie STOP</pre>	Komunikat o potrzebie zatrzymania wykonywania programu przez przekaźnik. Po zatrzymaniu (STOP) nacisnąć klawisz „OK.” w celu rozpoczęcia procedury zapisu. 

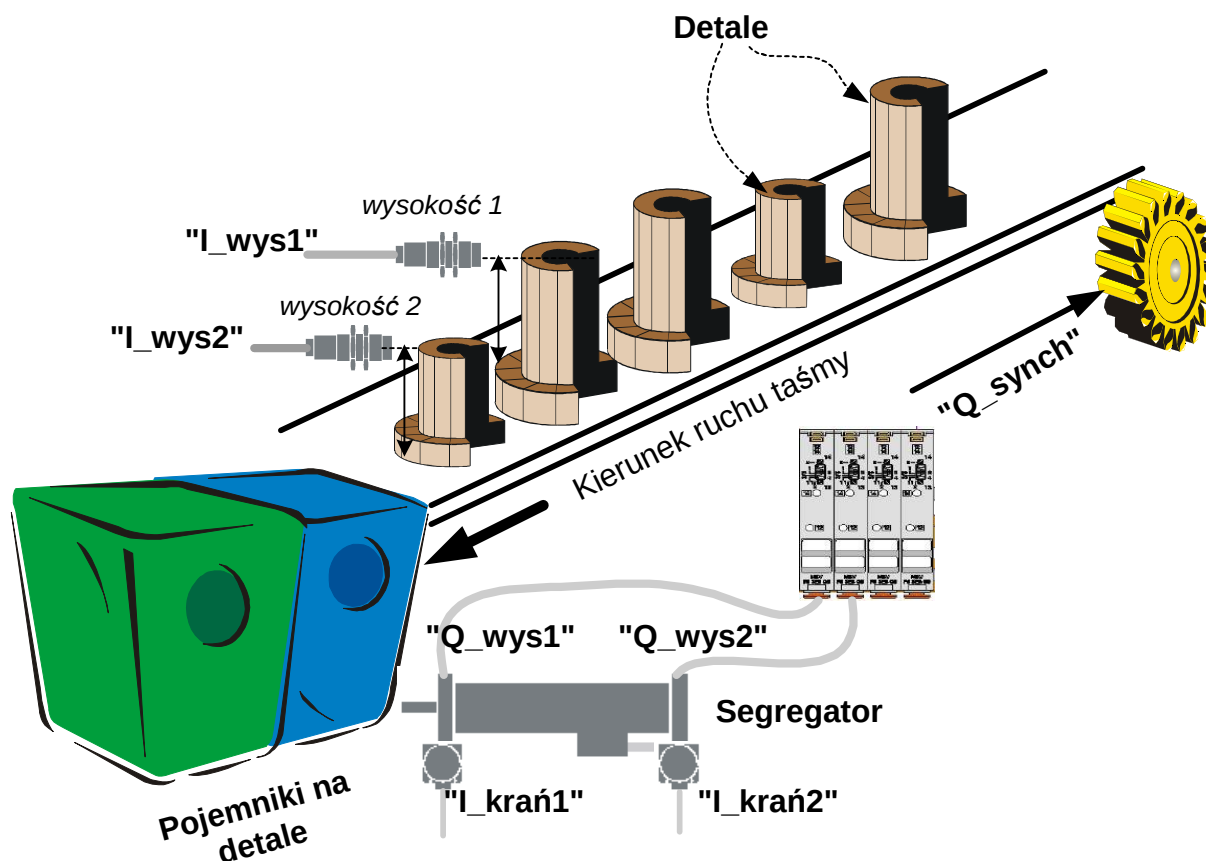
 Trwa zapis ****.	Informacja o postępie zapisu.
 Koniec zapisu	Zapis zakończony – nacisnąć klawisz ESC aby potwierdzić komunikat i zakończyć operację zapisu. 

11. PRZYKŁADY ZASTOSOWAŃ

11.1. Ocena wysokości detalu

Często w procesie produkcyjnym zachodzi konieczność segregacji detali ze względu na ich wymiary geometryczne. Można to robić ręcznie, mierząc jakiś wymiar lub automatycznie – stosując przełącznik programowalny NEED oraz kilka czujników zewnętrznych.

Jeśli przyjmimy np. dwie kategorie wysokości detali, to dla skompletowania całego sterowania wystarczy przełącznik programowalny NEED oraz dwa czujniki detekujące w odpowiedni sposób zadany wymiar geometryczny.



Rys. 10.1.1. Detekcja wysokości detali.

Opis zadania:

Należy dokonać segregacji detali, przesuwających się na podajniku taktowym, ze względu na ich wysokość.

Dobór sprzętu:

- 1) Dla prawidłowego wykonania zadania należy dobrać dwa czujniki o odpowiednim zasięgu. Do detekcji wysokości (jeśli są to detale metalowe) można zastosować czujniki indukcyjne. Nazwijmy czujnik detekujący wysokość 1 (detal wysoki) „I_wys1”, a czujnik detekujący wysokość 2 (detal niski) „I_wys2”. Zakładamy, że ruch taśmy jest synchronizowany, więc potrzebny jest jeszcze sygnał zezwalający na ponowny przesuw taśmy – nazwijmy go „Q_synch”.
- 2) Segregator – może to być siłownik sterowany zaworem elektromagnetycznym impulsowym (po podaniu sygnału sterującego na jedną cewkę elektromagnesu zawór pozostaje w zajmowanym położeniu również po zaniku tego sygnału, aż do chwili podania sygnału na drugą cewkę), na którym zamocowane będą dwa pojemniki.

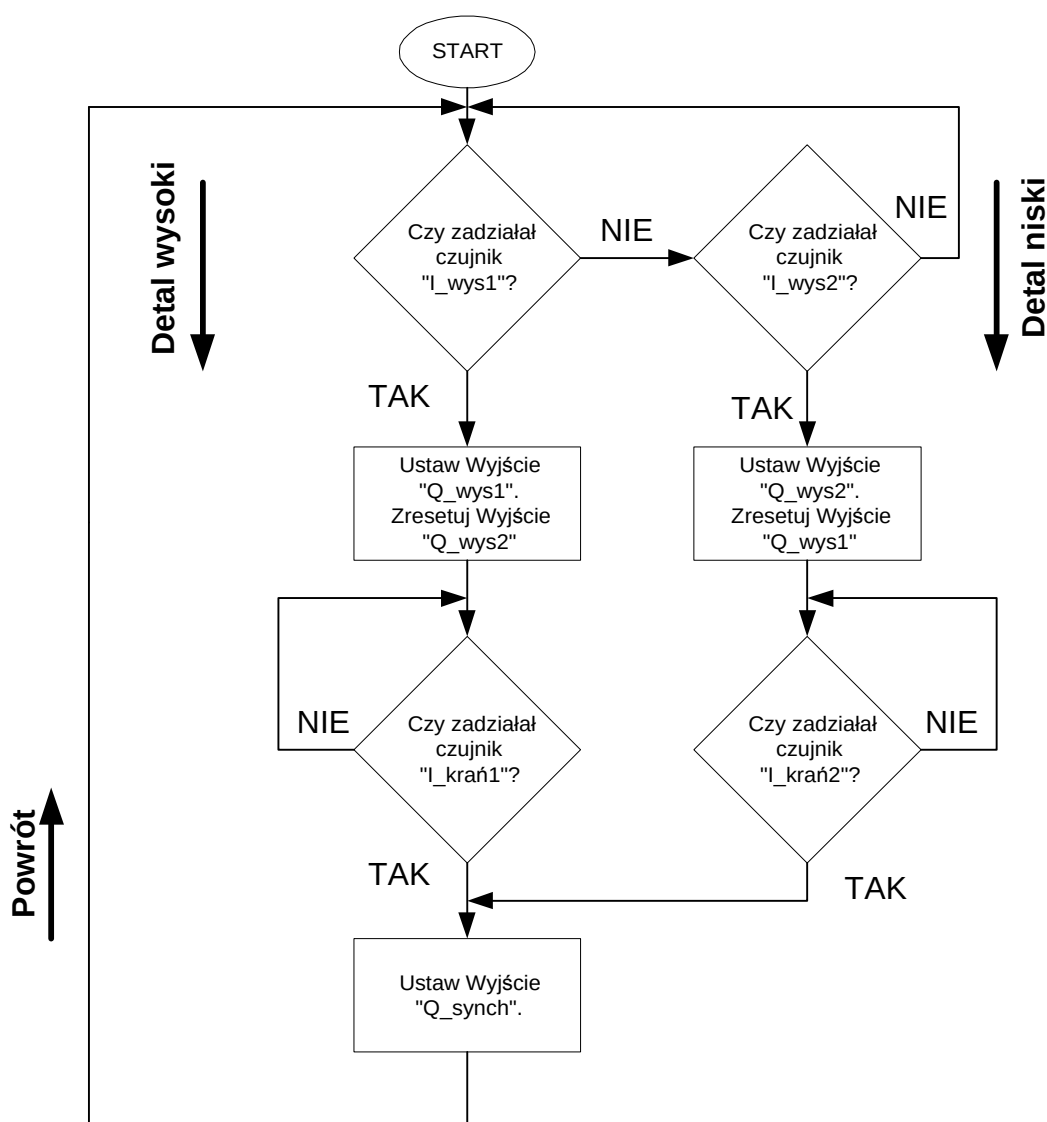
Pozycje krańcowe siłownika oznaczać będą: „Wysokość 1 detali”, „Wysokość 2 detali”. Nazwijmy wyjścia sterujące elektrozaworami – „Q_wys1” oraz „Q_wys2”. Odpowiadające im czujniki krańcowych pozycji „I_krań_1”, „I_krań_2”.

3) Przekaznik programowalny: potrzebujemy 4 Wejścia, 3 wyjścia.

Na rys. 11.1.1. przedstawiono idee analizowanego zadania, a na rys. 11.1.2. przykładowy schemat połączeń elektrycznych.

Algorytm:

Na początek będziemy sprawdzać wysokość detali. Jeśli zadziała czujnik „I_wys1” to wiemy na pewno, że jest to detal wyższy. Jeśli zadziała czujnik „I_wys2” i nie zadziała czujnik „I_wys1”, to będzie to detal niższy. W zależności od wysokości detalu przesuwamy (załączamy wyjście podłączone do elektrozaworu) pojemnik albo na detale wyższe, albo na detale niższe. Po całej operacji (gdy pojemnik jest ustawiony na właściwy typ detalu) dajemy sygnał synchronizacji do przesuwu taśmy. Dokładny algorytm działania przedstawiono poniżej.



Zauważmy, że nasz program „nigdzie się nie zatrzymuje”! Sterownik nie czeka na żaden sygnał startu – program jest po prostu przetwarzany przez procesor od pierwszej linii do ostatniej.

Program

Uporządkujmy naszą konfigurację sprzętową:

Adres	Wejścia	Adres	Wyjścia
I1	„I_wys1”	Q1	„Q_wys1”
I2	„I_wys2”	Q2	„Q_wys2”
I3	„I_krań1”	Q3	„Q_synch”
I4	„I_krań2”		

Spróbujmy przełożyć teraz algorytm na język programowania.

STL

```
//Detekcja detalu wysokiego
A I1          //Jeżeli I1=1, to ustaw Q1. Jeżeli I1=0 to nic nie rób
AN T1         //Timer T1 musi być w stanie niskim
S Q1          //Przestaw pojemnik na wysokość „wysoki”
R Q2          //Q2=0, Q1=1

//Detekcja detalu niskiego
AN I1         //Czujnik do detekcji wyższych detali nie działa I1=0 oraz
A I2          //Czujnik do detekcji detali niższych I2=1
AN T1         //Timer T1 musi być w stanie niskim
S Q2          //Przestaw pojemnik na wysokość „niski”
R Q1          //Q2=1, Q1=0

//Detal wysoki
//Detekcja czujnika krańcowego I3 na siłowniku
A I3          //Jeśli pojawi się zbocze narastające na I3
A Q1          //oraz będzie ustawione Q1
=M1           //to ustaw Znacznik M1

//Detal niski
//Detekcja czujnika I4 na siłowniku
A I4          //Jeśli pojawi się zbocze narastające na I4
A Q2          //oraz będzie ustawione Q2
=M2           //to ustaw Znacznik M2

//Detekcja przerwy między detalami
AN I1         //Brak detalu „wysokiego”
AN I2         //Brak detalu niskiego
R M1          //Kasowanie Znaczników pomocniczych M1,M2
R M2

//Wyzwolenie Timera T1 do synchronizacji
O M1          //Jeśli Znacznik M1 lub
O M2          //Znacznik M2 są w stanie '1'
L 1s         //to wyzwól Timer1 w trybie Pojedynczy impuls
SE T1        //o czasie trwania 1s

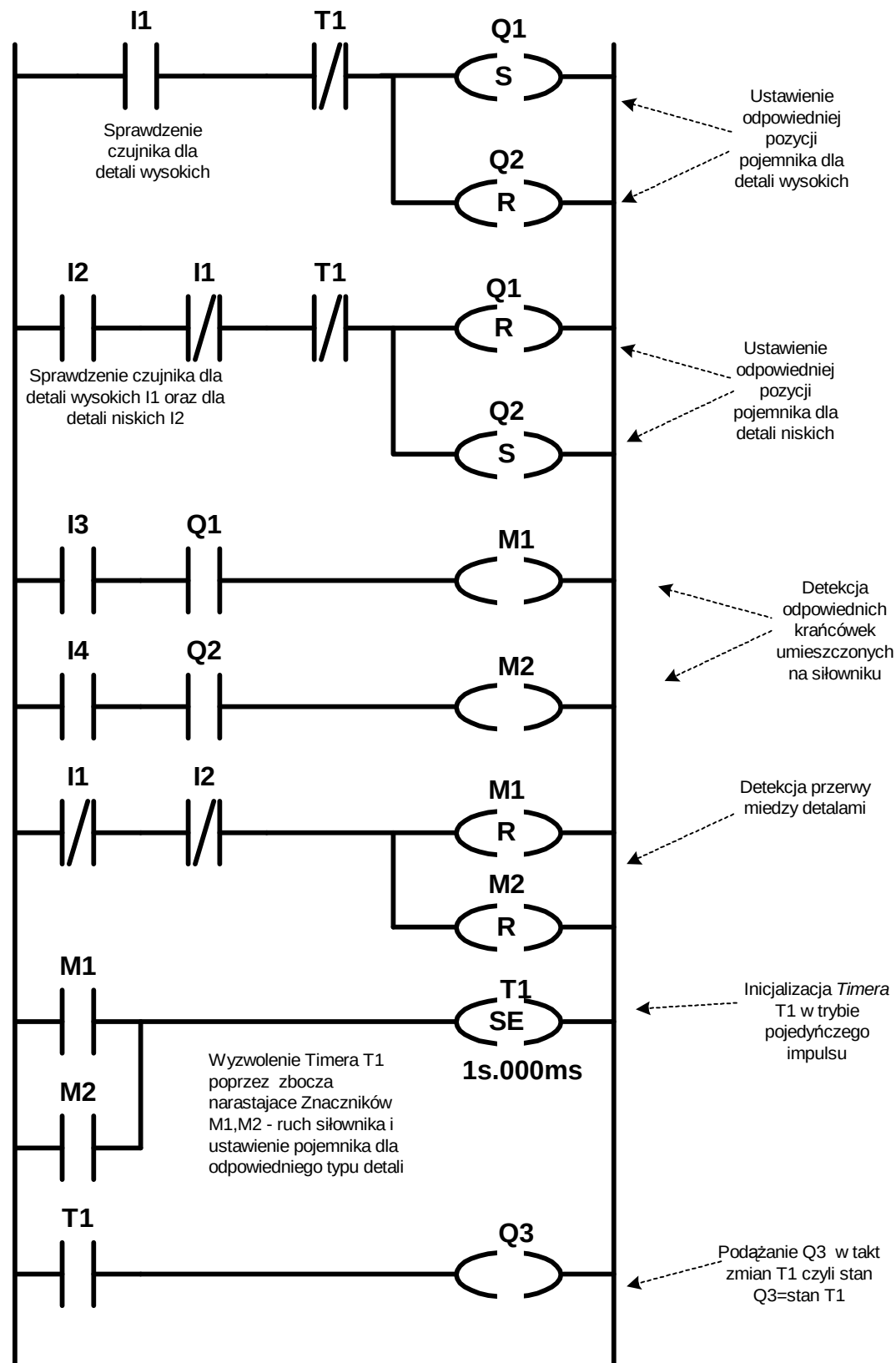
A T1          //Ustaw Q3 tak jak T1
=Q3
```

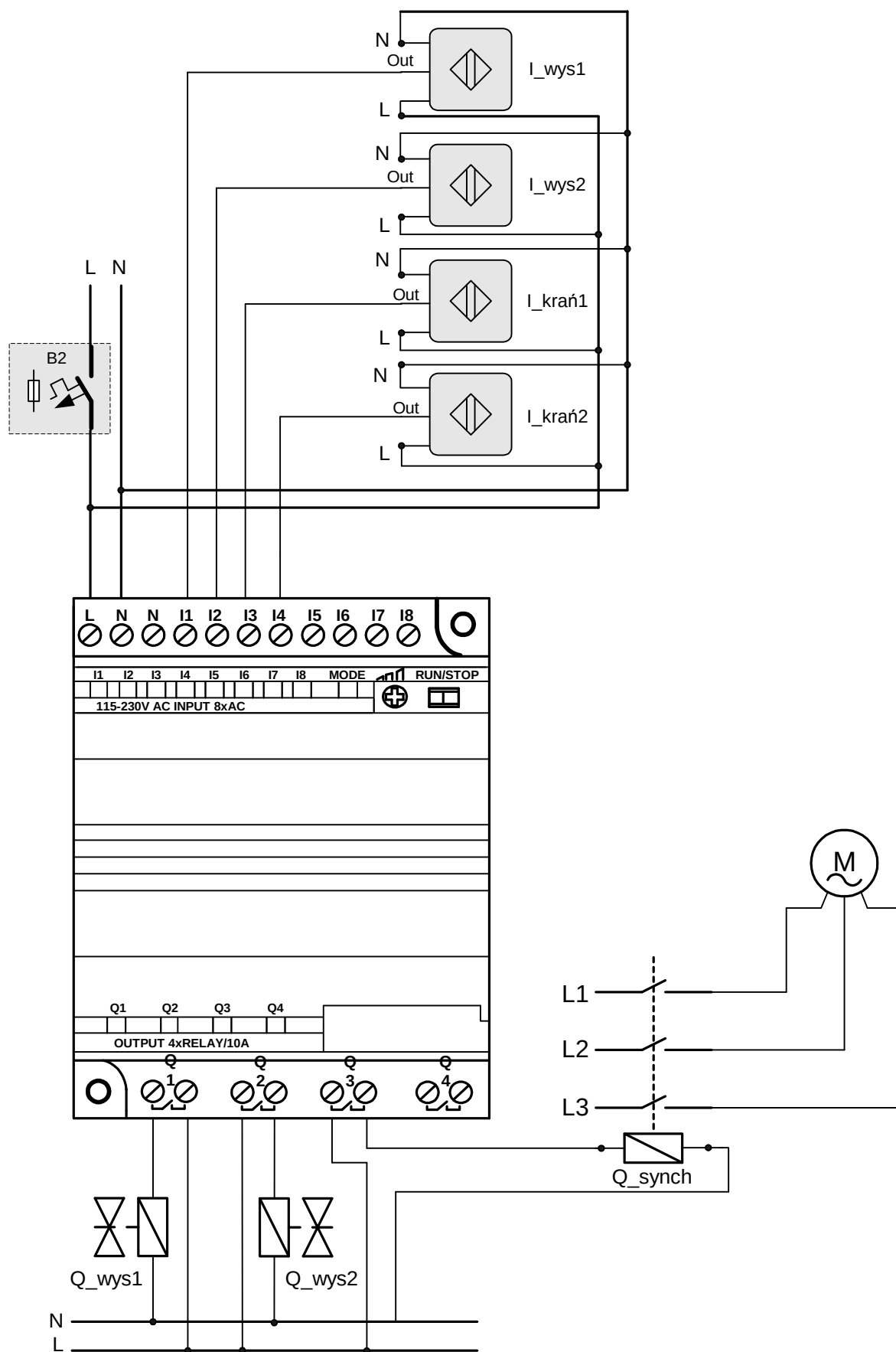
Do generacji impulsu wykorzystaliśmy *Timer* T1 w trybie Pojedynczego Impulsu. Czyli

pojawienie się zbocza narastającego na wejściu I3 lub I4 spowoduje wygenerowanie pojedynczego impulsu synchronizacji na Q3.

Poniżej przedstawiono wersję programu w języku graficznym LAD.

LAD

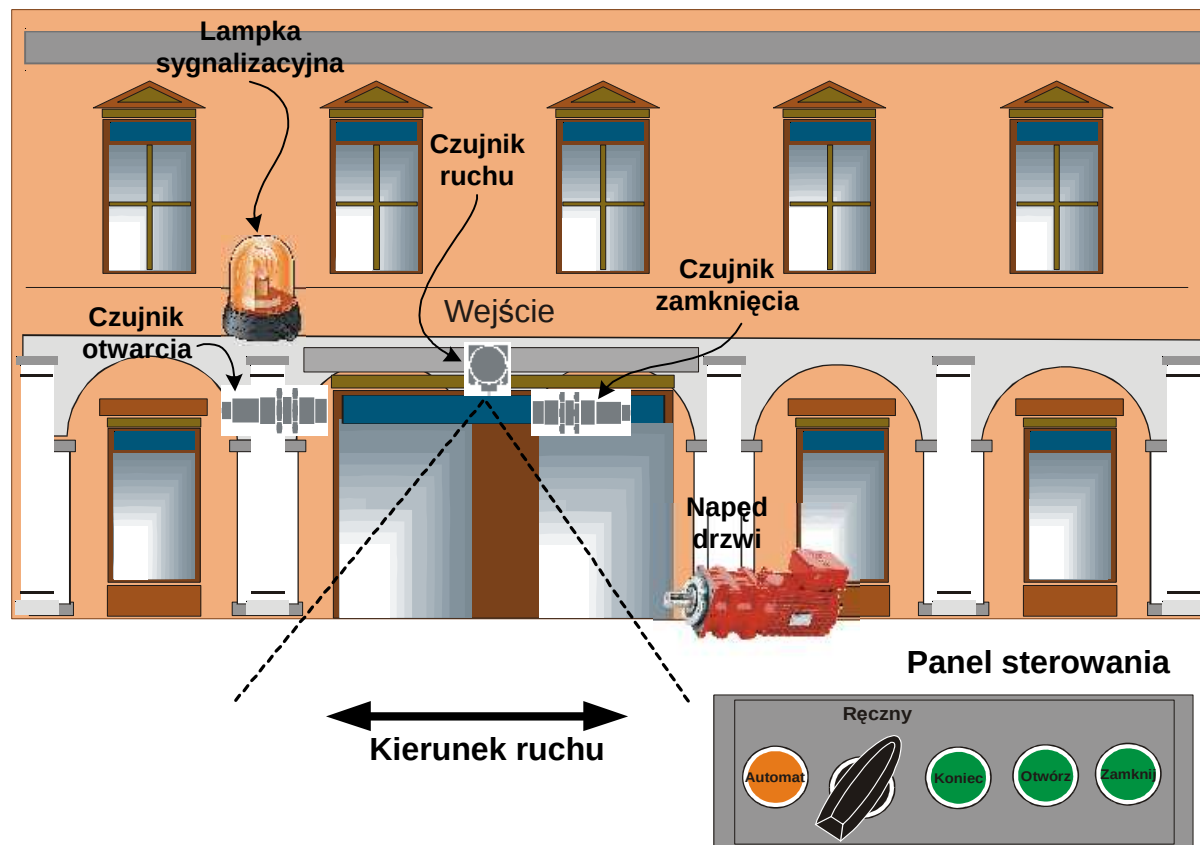




Rys. 10.1.2. Przykładowy schemat połączeń elektrycznych układu do detekcji wysokości detalu.

11.2. Drzwi automatyczne

Sterowanie drzwiami automatycznymi jest chyba wszystkim dobrze znane. Sklepy, urzędy, banki itp. bardzo często wykorzystują drzwi automatyczne, ale przy pomocy przekaźnika programowalnego NEED można tradycyjne sterowanie ubogacić o nowe funkcje, znacznie usprawniające nie tylko ruch klientów, ale i funkcjonalność całego budynku.



Rys. 10.2.1. Sterowanie drzwiami automatycznymi.

Opis zadania:

Należy sterować otwieraniem i zamykaniem drzwi automatycznych w budynku.

Dobór sprzętu:

- 1) Należy dobrać odpowiedni czujnik do detekcji ruchu. Zasięg tego czujnika powinien być taki, aby czas reakcji drzwi nie tamował ruchu. Czyli, aby otwarcie drzwi odbywało się odpowiednio wcześniej, przed dojściem klienta. Czujnik zewnętrzny nazwijmy „I_zew”, a wewnętrzny „I_wew”. Aby detekować położenie drzwi należy także zainstalować czujniki położenia krańcowych. Nazwijmy je „I_otwarte” dla otwartych drzwi i „I_zamknięte” dla zamkniętych. Dla zwiększenia funkcjonalności drzwi dołączmy jeszcze przełącznik, którym będziemy mogli ustawiać 3 tryb pracy: *Automat* – wszystko funkcjonuje tak, jak w czasie normalnych godzin pracy, *Koniec* – drzwi zostają otwarte tylko dla osób wychodzących z budynku. Dobrą regułą przy takiego typu sterowaniu jest także sterowanie ręczne – wyposażmy więc nasz układ w przyciski „Otwórz” oraz „Zamknij” służące do ręcznego otwierania i zamykania drzwi – w trybie pracy *Ręczny* (nie jest załączone ani *Zamykanie*, ani *Automat*). Wszystkie przyciski zbierzmy w jednym miejscu – w panelu sterowania.
- 2) Drzwi powinny być napędzane silnikiem ze sprzęgłem, które zabezpiecza przed zaciśnięciem. Oznaczmy sygnał sterujący ruchem silnika „Q_zamknij” (ruch do

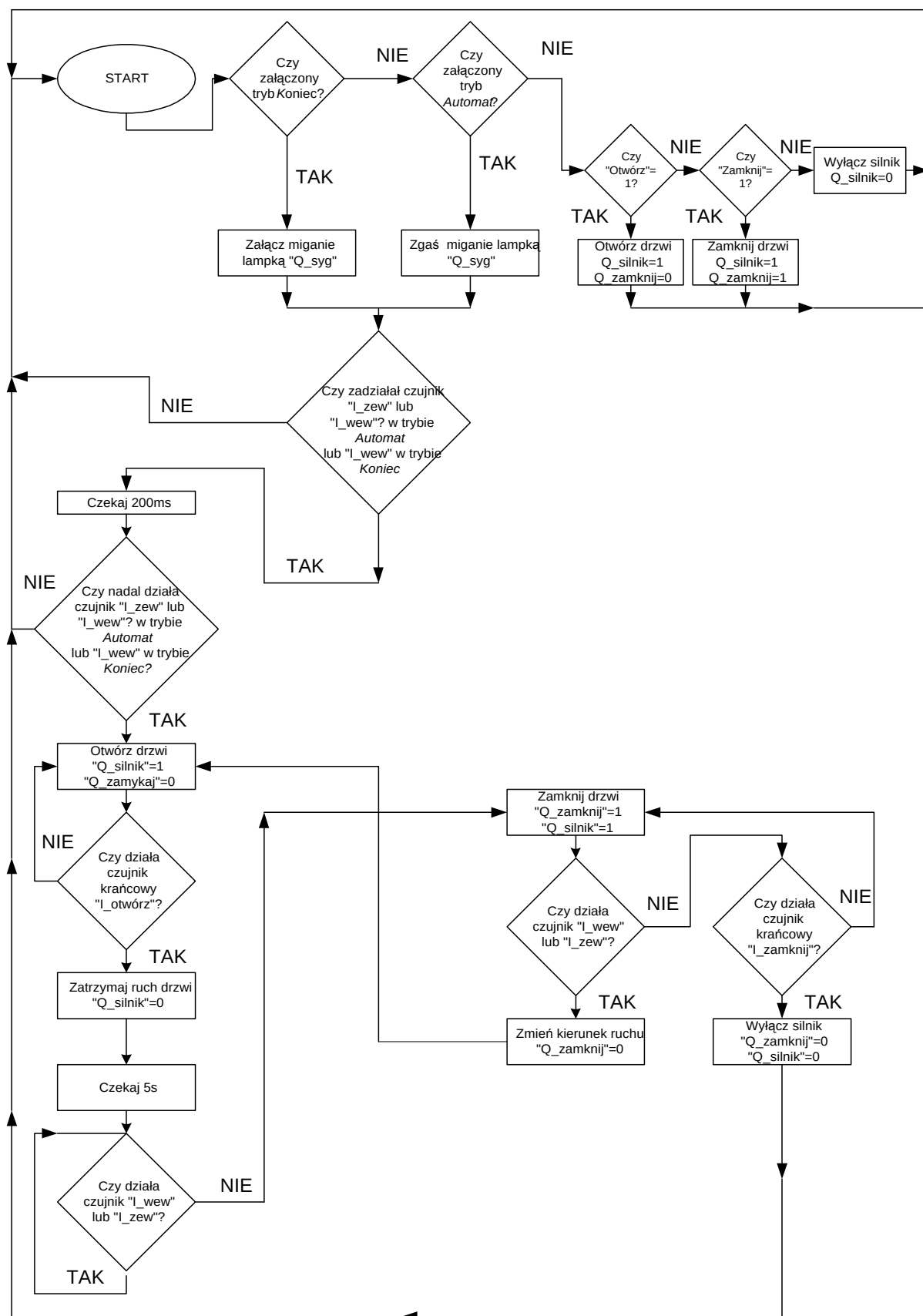
- przodu – zamykanie – stycznik załączony, ruch do tyłu – otwieranie – stycznik wyłączony) oraz „Q_silnik” – wyjście załączające silnik. W cały układ włączmy jeszcze lampkę sygnalizacyjną „Q_syg”, której miganie będzie oznaczało „Zamykanie” sklepu.
- 3) Przekaznik programowalny NEED: będziemy potrzebować 8 wejść i 3 wyjścia.

Algorytm

Na początku ustalamy tryb pracy, który sygnalizowany jest świeceniem lampki (w tym przykładzie lampka miga dla trybu *Zamykanie*). Drzwi mają otwierać się, gdy sygnał na wyjściu detektora ruchu jest wysoki. Aby uniknąć reakcji otwierania na przypadkowe wyzwolenia, układ reaguje dopiero po 200ms tzn., jeśli po 200ms od wyzwolenia nadal czujnik ruchu wskazuje na przemieszczenie, wówczas drzwi się otwierają. Czas zwłoki musi oczywiście być dobrany w taki sposób, aby osoba wchodząca lub wychodząca nie musiała czekać na otwarcie drzwi (także odpowiednie ustawienie oraz czułość detektorów ruchu).

Po otwarciu, drzwi pozostają w takim stanie przez ok. 5s, a następnie zamykają się. Każdy wykryty ruch przez czujnik podczas zamykania powoduje ponowne otwarcie drzwi. Do zatrzymania drzwi służą czujniki krańcowe. Jeśli zostanie ustalony tryb pracy *Ręczny*, otwieranie drzwi odbywa się za pomocą przycisku „Otwórz”

Algorytm



Program

Konfiguracja sprzętowa:

Adres	Wejścia	Adres	Wyjścia
I1	„I_wew”	Q1	„Q_zamknij”
I2	„I_zew”	Q2	„Q_silnik”
I3	„I_otwarte”	Q3	„Q_syg”
I4	„I_zamkniete”		
I5	„Automat”		
I6	„Koniec”		
I7	„Zamknij”		
I8	„Otwórz”		

STL

```

O I1 //Wprowadzamy Znacznik M1, który będzie informował nas
O I2 // o zadziałaniu któregoś z czujników ruchu: „I_wew”, „I_zew”.
=M1

A M1 // Znacznik M2 jest ustawiany, gdy mamy tryb „Automat” oraz
A I5 // nastąpiła detekcja ruchu wewnątrz i na zewnątrz budynku
=M2

A I1 // Znacznik M3 jest ustawiany, gdy mamy tryb „Zamykamy”
A I6 // oraz nastąpiła detekcja ruchu wewnątrz budynku
=M3

AN I5 //Ustalenie trybu pracy “Ręczny”
AN I6
=M13 //Znacznik trybu pracy “Ręczny”

A I6 // I6 jako sygnał wyzwalający T3
L 1s //Ustawienie Timera 3 w trybie migania (długość impulsu 1s)
SL T3

A T3 //Zapalenie lampki sygnalizacyjnej dla trybu pracy „Zamykanie”
=Q3

O M2 // Znacznik M4 jest ustawiany, gdy nastąpiła detekcja ruchu
O M3 // po którejś ze stron, w którymś z trybów
= M4

A M4
L 200ms //Opóźnienie 200ms generowane przez Timer T1 w trybie
SD T1 //Pojedynczy Impuls

A T1 //Ustawienie Znacznika pomocniczego M6 po czasie 200ms od
S M6 //wyzwolenia – od zadziałania jakiegoś czujnika ruchu

A T1 //Sprawdzenie po 200ms czy po którejś ze stron czujnik detekuje ruch
A M4
R M5 //Ustalenie kierunku ruchu drzwi – otwieranie

```

A M6 //Działanie silnika drzwi wejściowych
 AN I3 //do momentu zadziałania czujnika I3
 = M16

A M6 //Działanie silnika drzwi wejściowych
 AN I4 //do momentu zadziałania czujnika I3
 = M15

O M15 //Załączenie lub wyłączenie silnika drzwi
 O M16
 =Q2

A I3 //Wyzwolenie Timera T2 pracującego w trybie Opóźnione Załączenie
 L 5s //czyli ustawienie "szybnego" czasu otwarcia drzwi
 SD T2
 R M6 //Kasowanie Znacznika M6

A T2 //Ustawienie pomocniczego Znacznika M5
 AN M4
 S M5

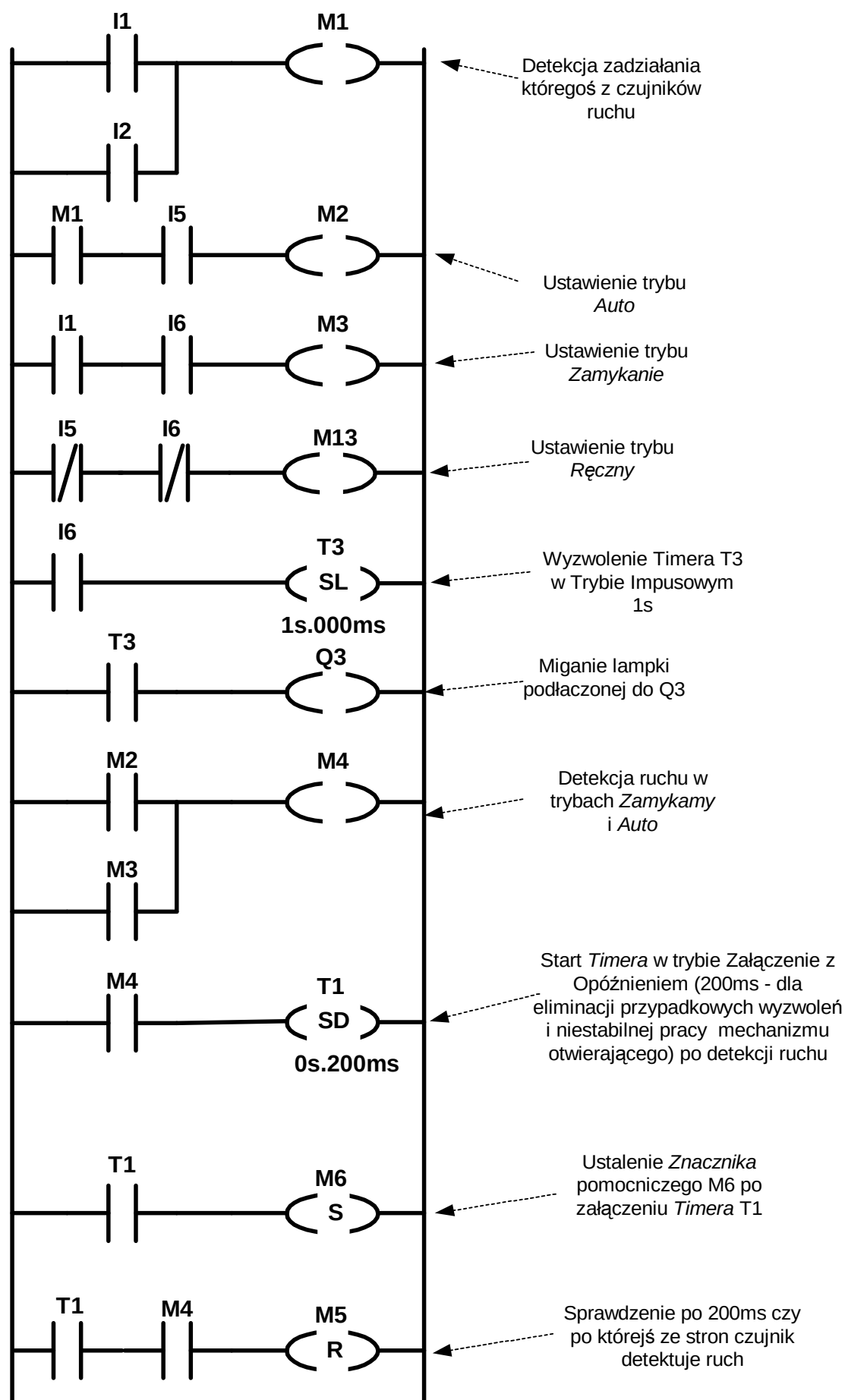
A M5 //Wyłączenie Q1 po osiągnięciu
 AN I4 //pozycji krańcowej sygnalizowanej przez I4
 =Q1

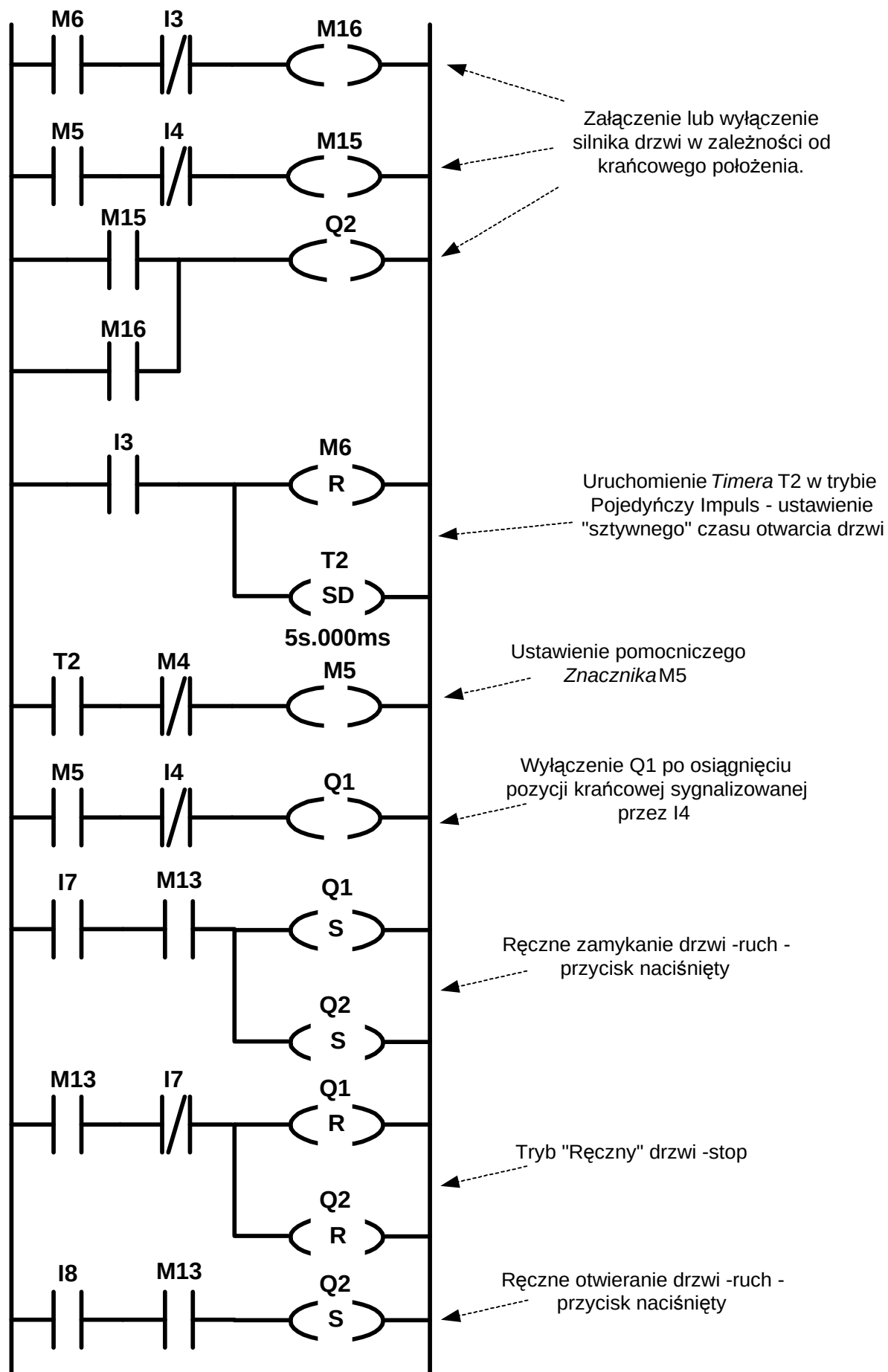
A I7 //Ręczne zamykanie drzwi –ruch – przycisk naciśnięty
 A M13
 S Q1
 S Q2

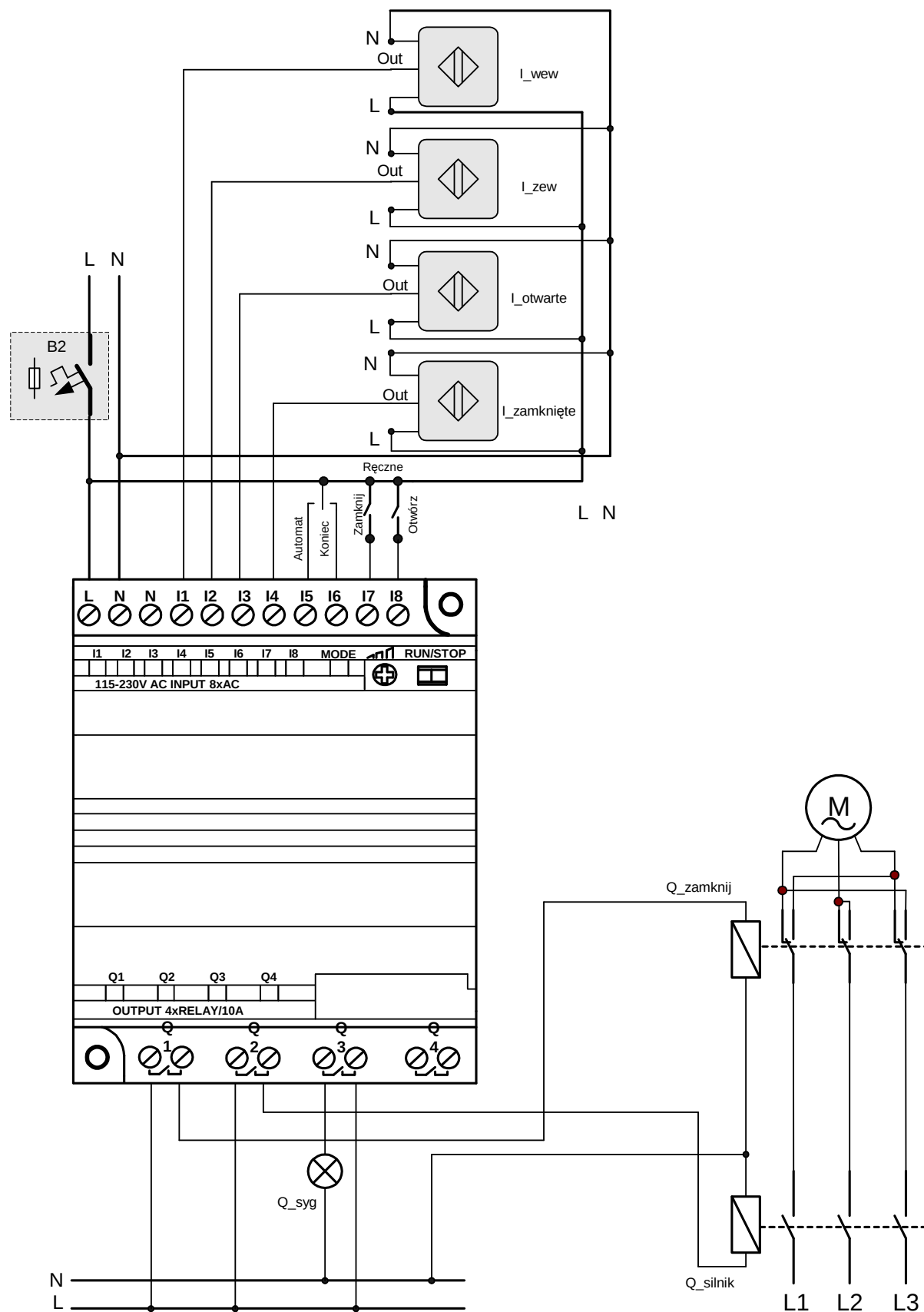
AN I7 //Tryb „Ręczny” drzwi –stop.
 A M13
 R Q1
 R Q2

A M13 //Ręczne otwieranie drzwi –ruch – przycisk naciśnięty
 A I8
 S Q2

LAD







Rys. 10.2.2. Przykład połączeń elektrycznych do sterowania drzwiami automatycznymi.

11.3. Dzwonki w szkole

Często zdarza się, iż w szkołach, zakładach pracy instalowane są czasomierze-zegary załączające w określonych godzinach różne urządzenia (dzwonki, alarmy, grzałki itp.) Używając przekaźnika programowalnego NEED można stworzyć własny, prosty, czasowy system sterowania – bardziej przystosowany do lokalnych wymogów i potrzeb.

Opis zadania:

Należy stworzyć system dzwonienia oparty o następujący rozkład zajęć w szkole:

Lekcja	Czas trwania	Dzwonek na przerwę		Dzwonek na lekcję	
		Początek załączenia	Początek wyłączenia	Początek załączenia	Początek wyłączenia
Lekcja 1	8.00 – 8.45	8.45	8.46	8.49	8.50
Lekcja 2	8.50 – 9.35	9.35	9.36	9.39	9.40
Lekcja 3	9.40 – 10.25	10.25	10.26	10.34	10.35
Lekcja 4	10.35 – 11.20	11.20	11.21	11.49	11.50
Lekcja 5	11.50 – 12.35	12.35	12.36	12.44	12.45
Lekcja 6	12.45 – 13.30	13.30	13.31	13.39	13.40
Lekcja 7	13.40 – 14.25	14.25	14.26	14.34	14.35
Lekcja 8	14.35 – 15.20	15.20	15.21	15.29	15.30

Dobór sprzętu:

- 1) Należy dobrać odpowiedni panel z przyciskami, które zapewniłyby ręczny tryb sterowania dzwonekami oraz wyłączenie dzwoneków w wyznaczonych okresach np. w czasie wakacji, ferii świątecznych, soboty itp.

Nazwijmy:

- Przełącznik – trybu ręcznego – „I_ręczny” (w tym trybie możliwe tylko „dzwonienie ręczne”),
- Przełącznik – trybu automatyczny – „I_auto”,
- Przycisk – załączający dzwonek w trybie ręcznym – „I_włącz”,

- 2) Przekaźnik programowalny NEED: 3 wejścia, 1 wyjście.

Program

Konfiguracja sprzętowa:

Adres	Wejścia	Adres	Wyjścia
I1	„I_ręczny”	Q1	„Q_dzwonek”
I2	„I_auto”		
I3	„I_włącz”		

Algorytm

Do załączeń i wyłączeń dzwonka użyjemy Zegarów, w następującej konfiguracji:

Zegar 1

Brak nazwy (SET)*

Timer Zegar Licznik >= Komparator Remanencja Ustawienia wejść

Ustawienia zegarów

H: 1

Kanał A		Kanał B		Kanał C		Kanał D	
Dzień 1	Pn	Dzień 1	Pn	Dzień 1	Pn	Dzień 1	Pn
Dzień 2	Sob	Dzień 2	Sob	Dzień 2	Sob	Dzień 2	Sob
HH	MM	HH	MM	HH	MM	HH	MM
Zak: 8	45	Zak: 8	49	Zak: 9	35	Zak: 9	39
Wyk: 8	46	Wyk: 8	50	Wyk: 9	36	Wyk: 9	40

Zegar 2

Brak nazwy (SET)*

Timer Zegar Licznik >= Komparator Remanencja Ustawienia wejść

Ustawienia zegarów

H: 2

Kanał A		Kanał B		Kanał C		Kanał D	
Dzień 1	Pn	Dzień 1	Pn	Dzień 1	Pn	Dzień 1	Pn
Dzień 2	Sob	Dzień 2	Sob	Dzień 2	Sob	Dzień 2	Sob
HH	MM	HH	MM	HH	MM	HH	MM
Zak: 10	25	Zak: 10	34	Zak: 11	20	Zak: 11	49
Wyk: 10	26	Wyk: 10	35	Wyk: 11	21	Wyk: 11	50

Zegar 3

Brak nazwy (SET)*

Timer Zegar Licznik >= Komparator Remanencja Ustawienia wejść

Ustawienia zegarów

H: 3

Kanał	Dzień 1	Dzień 2	Zak. HH	Zak. MM	Wyt. HH	Wyt. MM
Kanał A	Pn	Sob	12	35	12	36
Kanał B	Pn	Sob	12	44	12	45
Kanał C	Pn	Sob	13	30	13	31
Kanał D	Pn	Sob	13	39	13	40

Zegar 4

Brak nazwy (SET)*

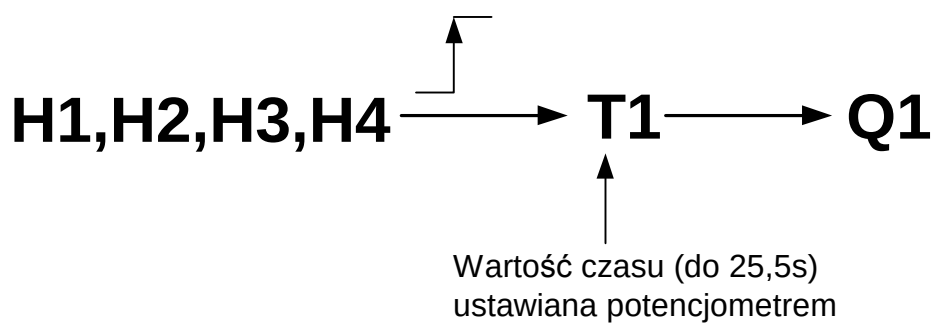
Timer Zegar Licznik >= Komparator Remanencja Ustawienia wejść

Ustawienia zegarów

H: 4

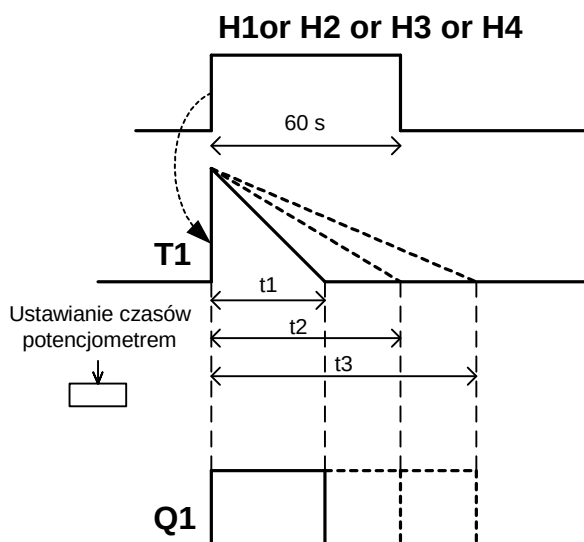
Kanał	Dzień 1	Dzień 2	Zak. HH	Zak. MM	Wyt. HH	Wyt. MM
Kanał A	Pn	Sob	14	25	14	26
Kanał B	Pn	Sob	14	34	14	35
Kanał C	Pn	Sob	15	20	15	21
Kanał D	Pn	Sob	15	29	15	30

Zauważmy jedną niedogodność – Zegary możemy ustawiać z dokładnością do 1 minuty. Jak w takim razie poradzić sobie z dzwonekami trwającymi np. tylko 7s (czas dzwonienia 1 minuty jest zbyt długi). Możemy zastosować *Timer* w trybie Pojedynczy impuls, dzięki któremu będzie możliwa regulacja czasów dzwonienia – rys. 11.3.2. przedstawia sposób załączania i wyłączania dzwonka.



Rys. 10.3.1. Sposób sterowania dzwonkiem.

Oczywiście nastawy Potencjometru określają tylko w bardzo przybliżony sposób wartości czasów, ale dla ustalenia czasu dzwonienia, tego typu szacowania dają zadawalające rezultaty. Czyli wyjścia Zegarów H1, H2, H3, H4 są załączone przez 1 minutę (najmniejszy możliwy przedział czasowy do ustalenia dla Zegarów). Zegary te wyzwalają Timer 1 (wartość czasu do odmierzenia regulowana Potencjometrem), który z kolei ustawia wyjście Q1. Na rys. 11.3.2. przedstawiony jest sposób „kształtowania” długości czasu dzwonków.



Rys. 10.3.2. Sposób „kształtowania” czasu dzwonienia.

STL

```

O H1          //Załączenie H1 albo,
O H2          //załączenie H2 albo,
O H3          //Załączenie H3 albo,
O H4          //Załączenie H4
L Pot x100ms  //Powoduje wyzwolenie Timera 1 w trybie Pojedynczy impuls
SE T1         //o czasie trwania ustalonym za pomocą Potencjometru

```

//Tryb AUTO

```

A I2          //Jeśli ustawiony jest tryb AUTO, to dzwonek pracuje normalnie
A T1
=Q1

```

//Tryb Ręczny

```

A I1          //Jeśli ustawiony jest tryb Ręczny, to dzwonek reaguje na
A I3          //wciśnięcie przycisku I3 – załączenie dzwonka
S Q1

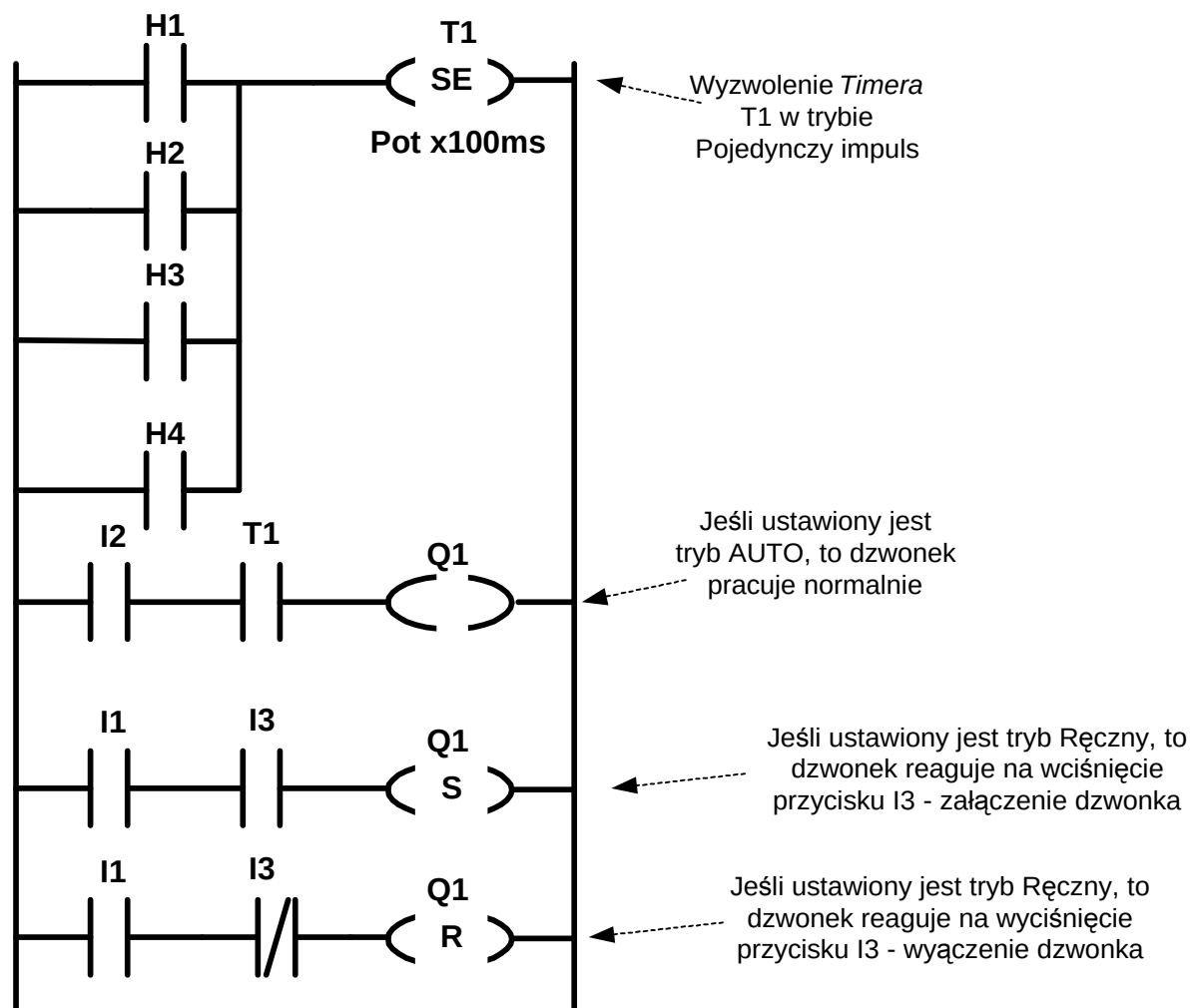
```

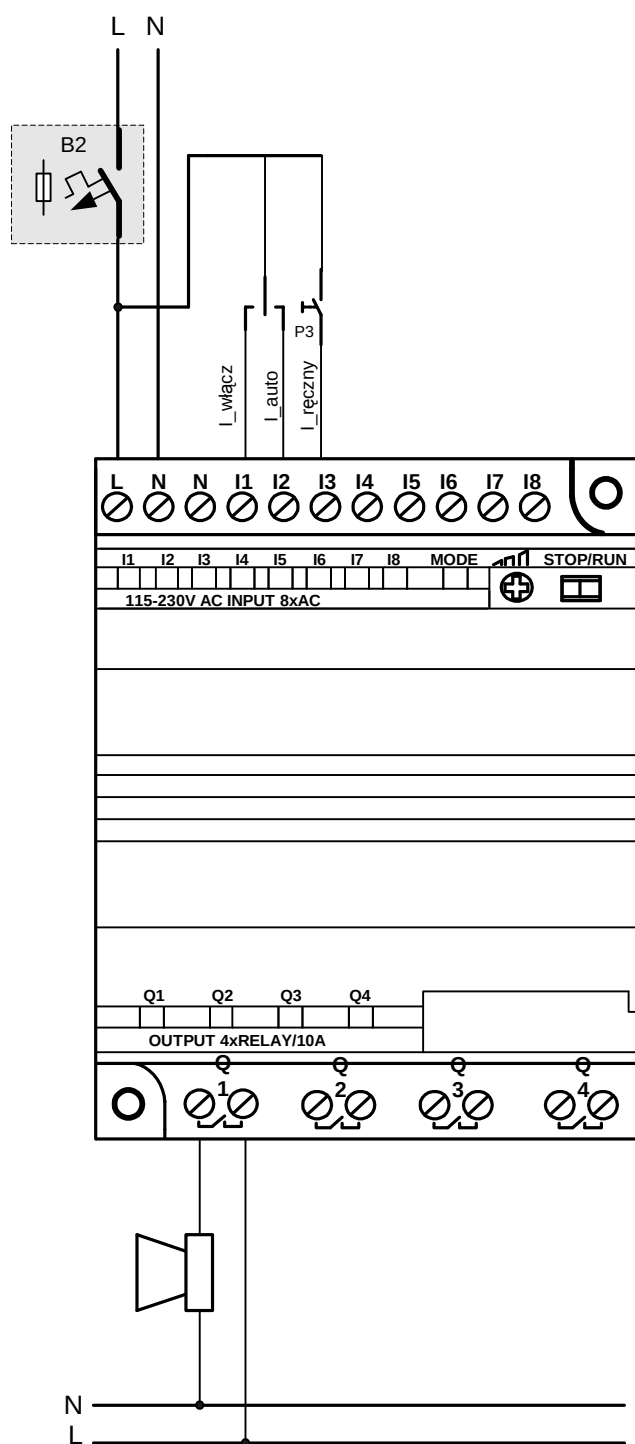
```

A I1          //Jeśli ustawiony jest tryb Ręczny, to dzwonek reaguje na
AN I3         //wyciśnięcie przycisku I3 – wyłączenie dzwonka
R Q1

```

LAD

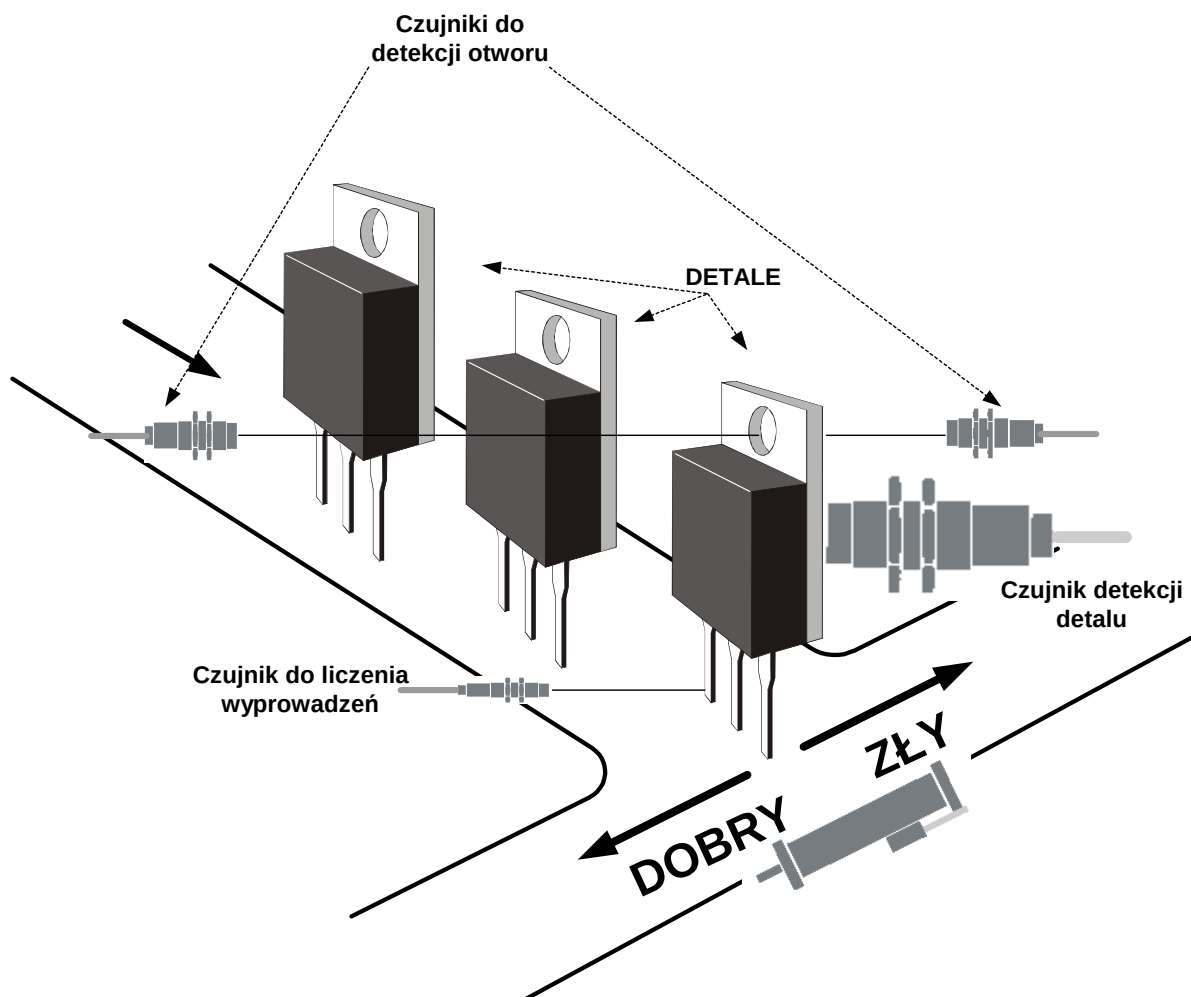




Rys. 10.3.3. Przykład połączeń elektrycznych dla automatycznych dzwonek w szkole.

11.4. Wykrywanie wad detali

W procesie produkcyjnym często zachodzi potrzeba wykrywania wadliwych detali. Za pomocą przekaźnika programowalnego NEED można w prosty i w tani sposób stworzyć system kontrolujący jakość produkowanych detali.



Rys.10.4.1. Wykrywanie wadliwych detali.

Opis zadania:

Należy stworzyć system pozwalający sprawdzać detale (otwór w obudowie tranzystora i liczba wyprowadzeń). Po detekcji złego detalu należy odsegregować go od pozostałych.

Dobór sprzętu:

- 4) Do detekcji otworu potrzebna jest para czujników optycznych (nadajnik i odbiornik), natomiast zliczać wyprowadzenia tranzystora może czujnik laserowy o małej średnicy plamki światła. Dołączmy jeszcze czujnik detekujący transportowany detal – uprości to znacznie nasz późniejszy program.
- 5) Segregator – może to być siłownik sterowany zaworem elektromagnetycznym impulsowym (po podaniu sygnału sterującego na jedną cewkę elektromagnesu zawór pozostaje w zajmowanym położeniu również po zaniku tego sygnału, aż do chwili podania sygnału na drugą cewkę), na którym zamocowane będą np. przegrody mechaniczne do przekierowania ruchu złych detali.
- 6) Przekaźnik programowalny NEED: 3 wejścia, 2 wyjścia.

Algorytm

W momencie pojawienia się detalu (działa czujnik) na linii transportowej powinna nastąpić aktywacja liczenia wyprowadzeń. W tym samym czasie powinien także być sprawdzony otwór w obudowie tranzystora.

Program:

Konfiguracja sprzętowa:

Adres	Wejścia	Adres	Wyjścia
I1	Czujnik do detekcji otworu w obudowie tranzystora	Q1	Pozycja segregatora <i>DOBRE</i>
I2	Czujnik liczący wyprowadzenia	Q2	Pozycja segregatora <i>ZŁE</i>
I3	Czujnik detekujący detal		

STL

```

A I3          //Detekcja obecności detalu, zapisanie stanu
= M1          //czujnika I3 do M1
R M3          //Kasowanie Znacznika „Dobry detal” – w ten sposób przegroda
              //zostaje zawsze w ostatniej pozycji i nie jest przestawiana
              //za każdym razem, gdy pojawia się detal

A M1          //Sprawdzenie otworu, przy obecności detalu
A I1
S M2

A I2          //Ustawienie Licznika C1 do zliczania 3 wyprowadzeń
L C#3         //tranzystora
CU C1

AN M1         //Sprawdzenie czy jest otwór i czy zostały zliczone
A M2          //trzy wyprowadzenia tranzystora, gdy detal przestał
A C1          //być “widziany” przez czujnik I3
S M3          //Ustawienie Znacznika „Dobry detal”

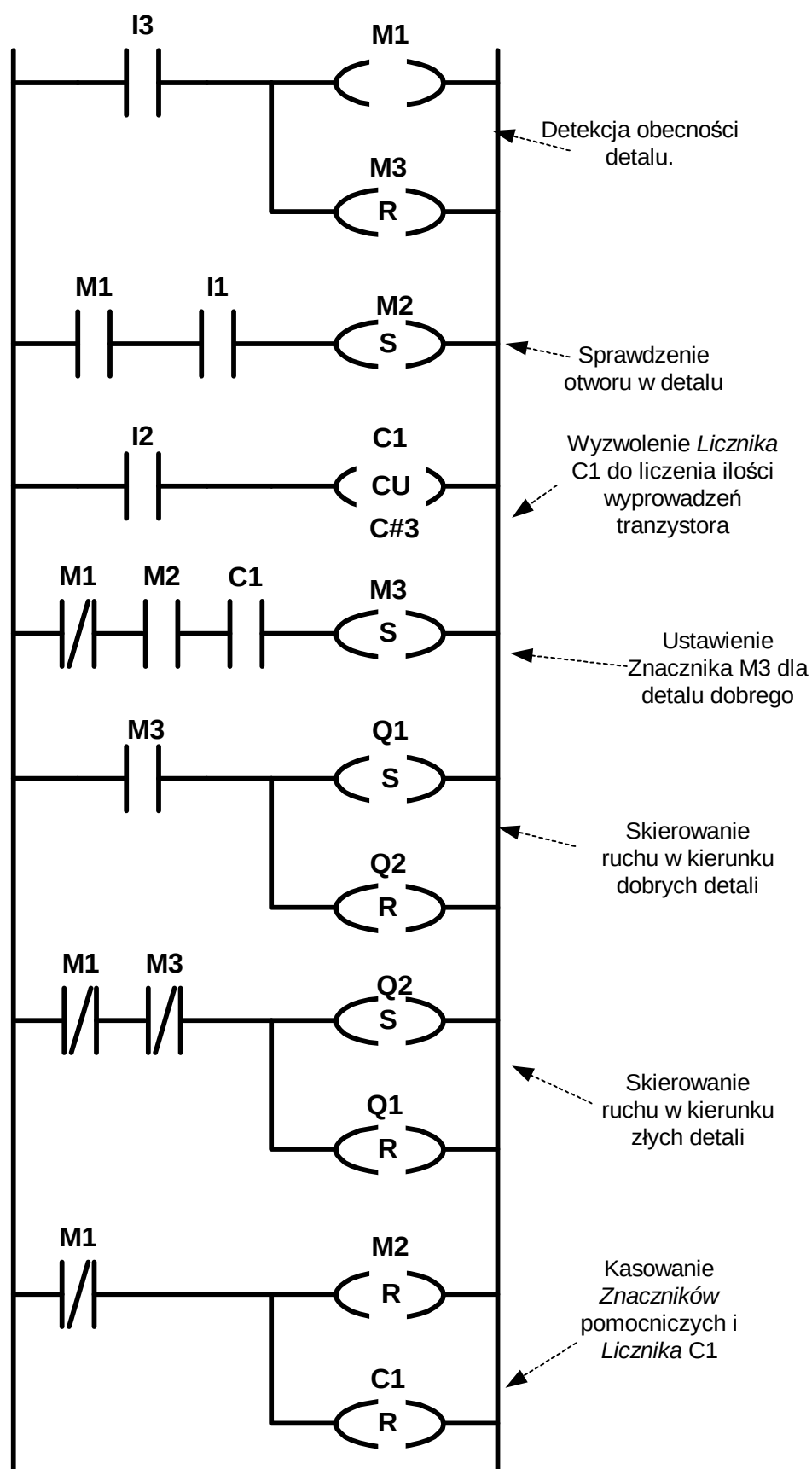
A M3          //Jeśli tranzystor dobry to zostaje przestawiona przegroda
S Q1          //uwalniająca ruch w kierunku detali dobrych
R Q2

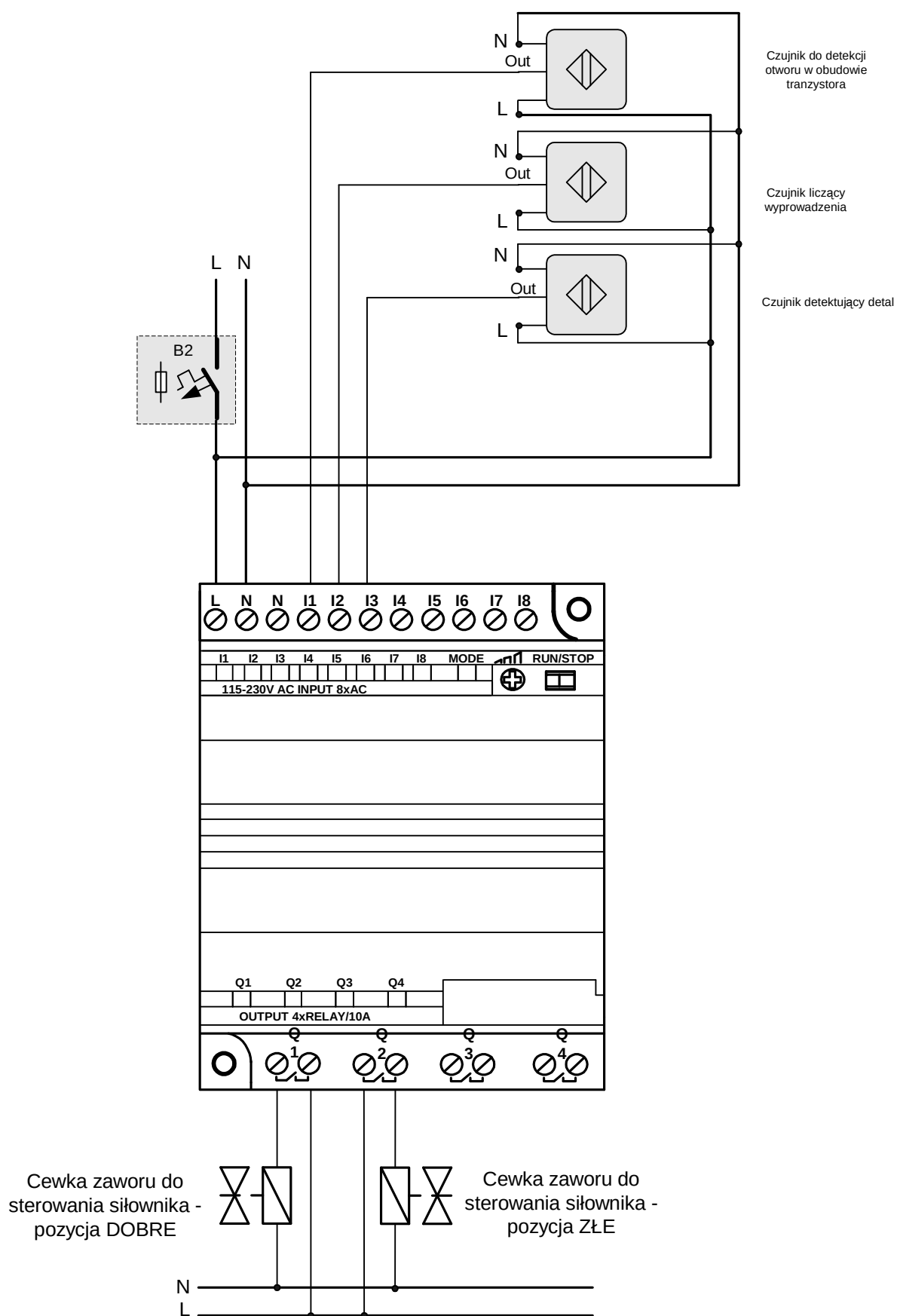
AN M1         //Jeśli tranzystor zły to zostaje przestawiona przegroda
AN M3         // uwalniająca ruch w kierunku detali złych
R Q1
S Q2

AN M1         //Kasowanie Znaczników pomocniczych i Licznika C1
R M2
R C1

```

LAD





Rys. 10.4.2. Przykład połączeń elektrycznych dla wykrywania wad detali.

11.5. Sterowanie ruchem wózków na zakręcie taśmociągu

Opis zadania.

Należy zrealizować przemieszczenie wózków z jednej strony taśmociągu na drugą.

Realizowane to jest przez obracający się talerz napędzany silnikiem M1.

Tylko jeden wózek może się znajdować na obrotowym talerzu.

Nie można wpuszczać następnego wózka, jeśli poprzedni nie opuścił bieżni talerza, lub taśmociąg jest wypełniony (kolejka wózków za zakrętem).

Dodatkowo na zakręcie wózki mogą być zdejmowane, ale muszą być z powrotem odłożone.

Do zrealizowania zadania potrzebne będą elementy sterowania, które poglądowo przedstawiono na rys. 11.5. – czujniki (wejścia) I1 i I2, wyjścia Q1, Q2 i Q3.

Wyjścia czujników podłączamy pod wejścia przełącznika programowalnego następująco:

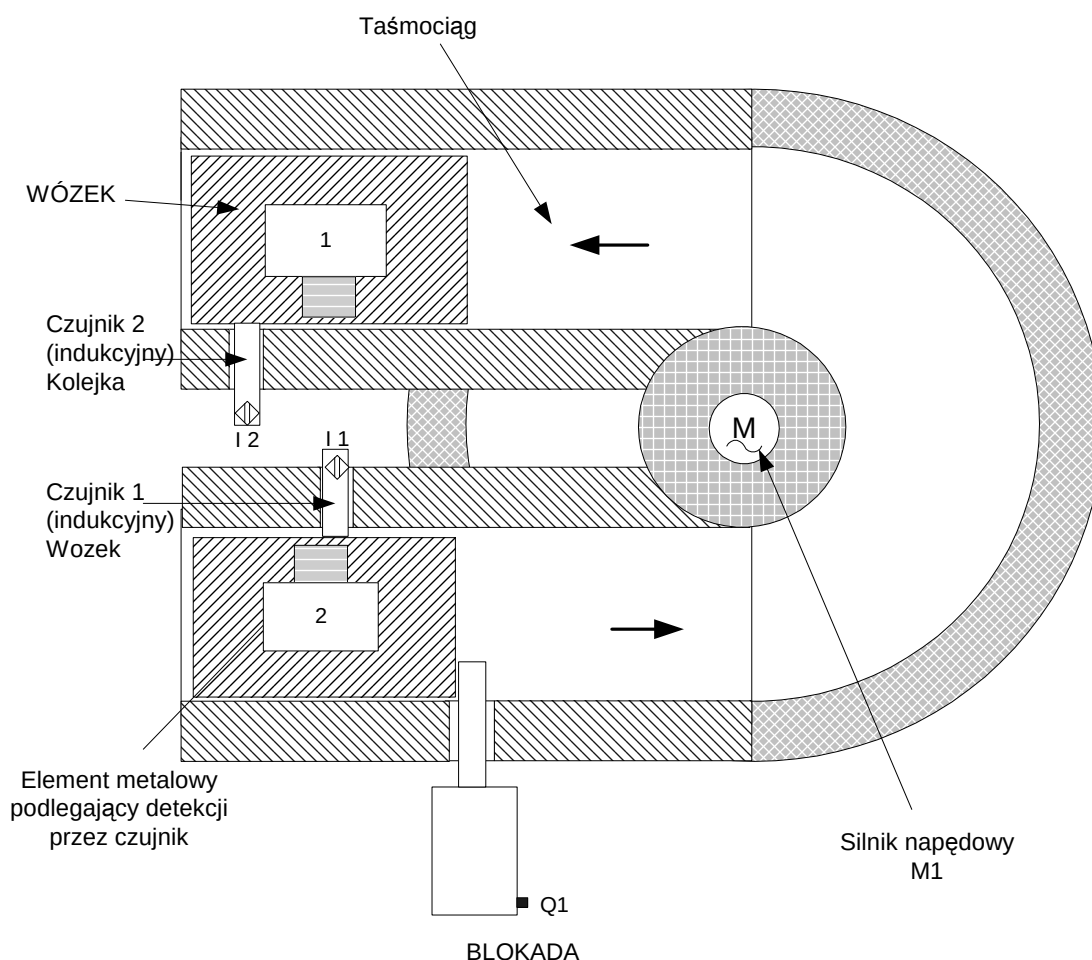
Wej. I1 – czujnik indukcyjny obecności wózka (230 V AC PNP)

Wej. I2 – czujnik indukcyjny kolejki, jednocześnie przejazdu wózka na drugą stronę taśmociągu (230 V AC PNP).

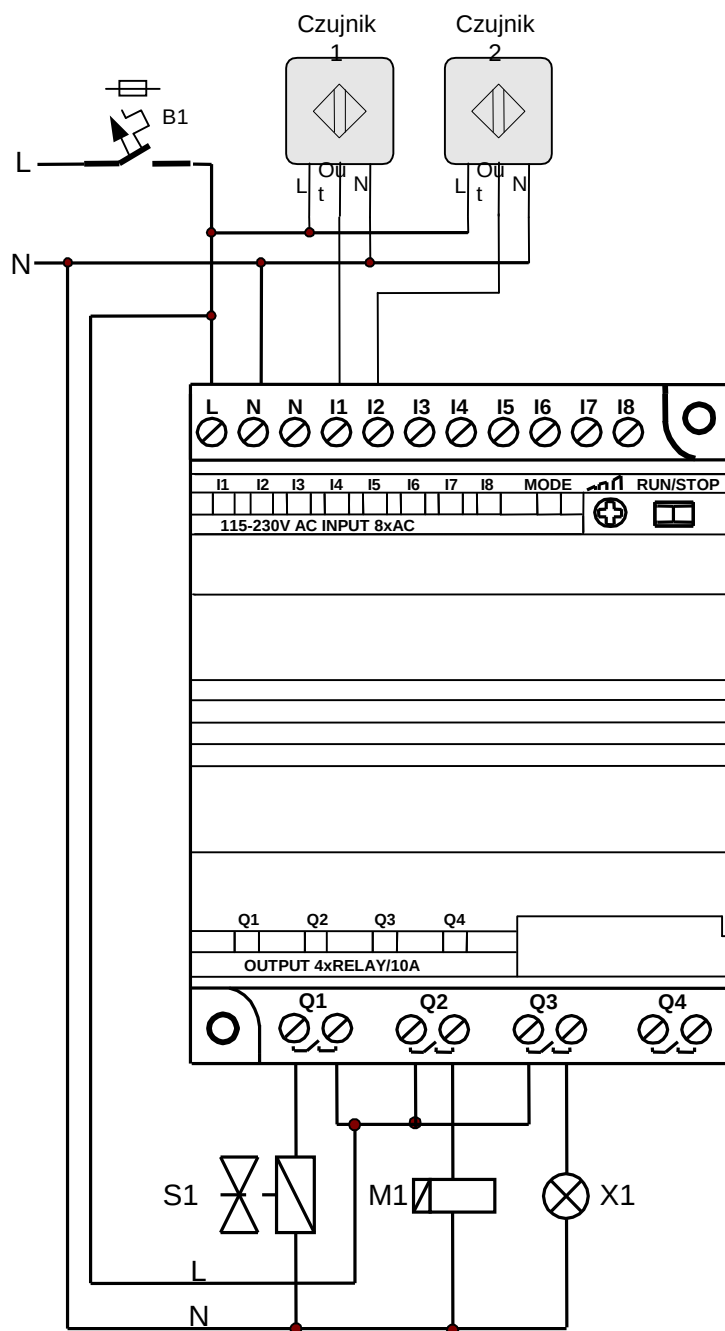
Wyj. Q1 – cewka elektrozaworu sterującego siłownikiem pneumatycznym S1 (230V AC).

Wyj. Q2 – załączanie silnika M1.

Wyj. Q3 – lampka informująca o zgodności liczby wózków wjeżdżających i wyjeżdżających.



Rys. 11.5.1. Sterowanie zakrętem taśmociągu.

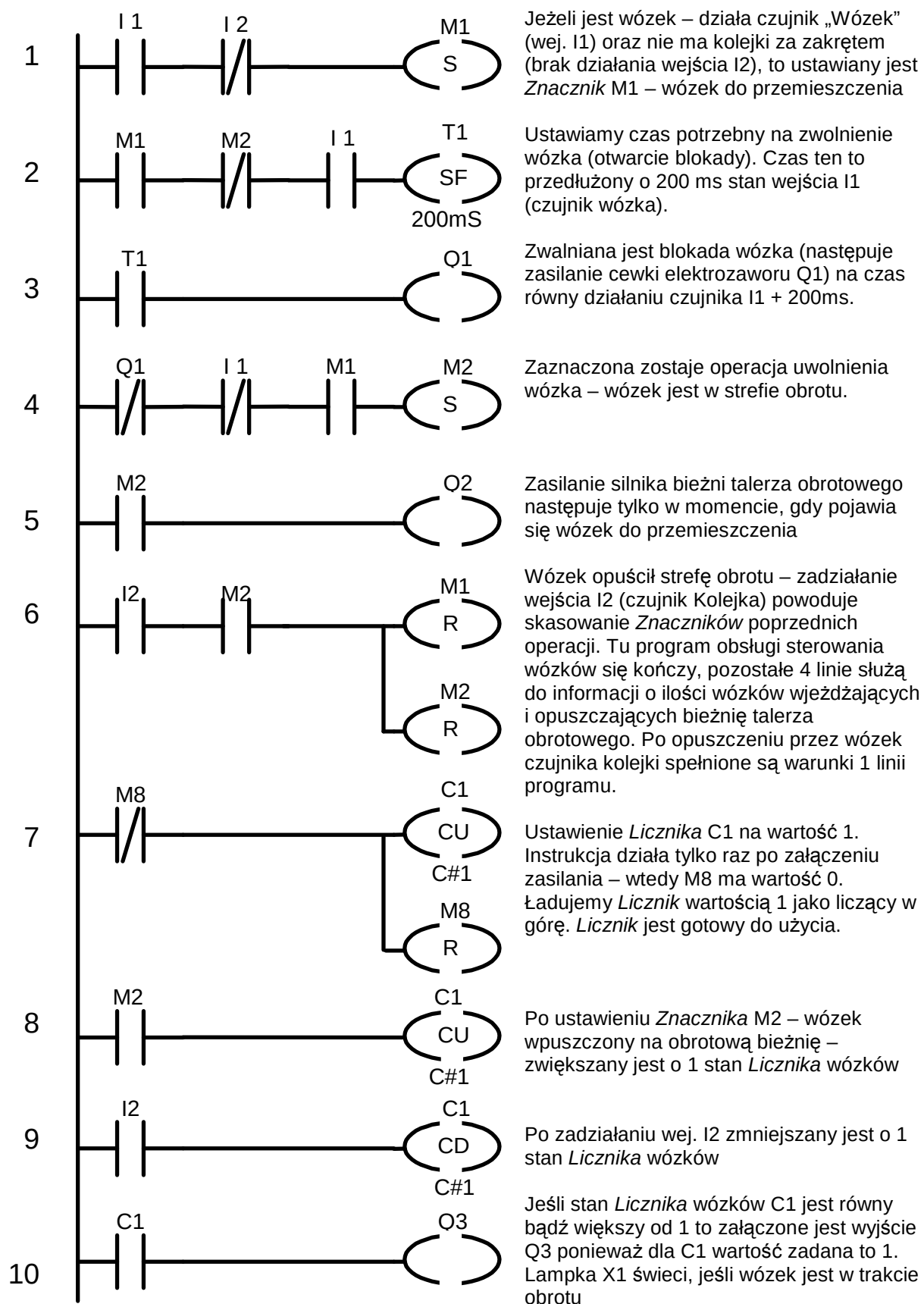


Rys. 11.5.2 Sterowanie zakretem taśmociągu – schemat elektryczny

Poniżej przedstawiono programy napisane w języku LAD i STL. Numery w pierwszej kolumnie służą do oznaczenia poszczególnych obwodów programu w celu porównania zapisu LAD i STL. Nie są one częścią programu.

W normalnym zapisie STL przerwy między instrukcjami nie muszą występować. Zwiększają one jednak przejrzystość programu. Dodatkowo można wprowadzić komentarze, które pomagają analizować (śledzić) program oraz później pozwalają na łatwą modyfikację.

Program LAD



Program STL

1	A I1 AN I2 S M1	Jeżeli jest wózek – działa czujnik „Wózek” (wej. I1) oraz nie ma kolejki za zakrętem (brak działania wejścia I2), to ustawiany jest <i>Znacznik</i> M1 – wózek do przemieszczenia
2	A M1 AN M2 A I1 L 200mS SF T1	Ustawiamy czas potrzebny na zwolnienie wózka (otwarcie blokady). Czas ten to przedłużony o 200 ms stan wejścia I1 (czujnik wózka). <i>Znacznikiem</i> M1 i M2 zapobiegają ponownemu zadziałaniu <i>Timera</i> w wypadku pojawienia się kolejnego wózka przy czujniku I1(M1) i zadziałaniu przed opuszczeniem wózka strefy obrotu (M2).
3	A T1 = Q1	Zwalniana jest blokada wózka (następuje zasilanie cewki elektrozaworu Q1) na czas równy działaniu czujnika I1 + 200ms.
4	AN Q1 AN I1 A M1 S M2	Zaznaczona zostaje operacja uwolnienia wózka – wózek jest w strefie obrotu.
5	A M2 = Q2	Zasilanie silnika bieżni talerza obrotowego następuje tylko w momencie, gdy pojawia się wózek do przemieszczenia
6	A I2 A M2 R M1 R M2	Wózek opuścił strefę obrotu – zadziałanie wejścia I2 (czujnik Kolejka) powoduje skasowanie <i>Znaczników</i> poprzednich operacji. Tu program obsługi sterowania przemieszczaniem wózków się kończy, pozostałe 4 linie służą do informacji o ilości wózków wjeżdżających i opuszczających bieżnię talerza obrotowego. Po opuszczeniu przez wózek czujnika kolejki spełnione są warunki 1 linii programu.
7	AN M8 L C#1 CU C2 S M8	Ustawienie <i>Licznika</i> C2 na wartość 1. Instrukcja działa tylko raz po załączeniu zasilania – wtedy M8 ma wartość 0. Ładujemy <i>Licznik</i> wartością 1 jako liczący w górę. <i>Licznik</i> jest gotowy do użycia. Ustawiamy M8 na '1' co zapewnia, że do momentu wyłączenia zasilania ten obwód (6) nie będzie miał wpływu na działanie programu.
8	A M2 L C#1 CU C2	Po ustawieniu <i>Znacznika</i> M2 – wózek wpuszczony na obrotową bieżnię – zwiększany jest o 1 stan <i>Licznika</i> wózków
9	A I2 L C#1 CD C2	Po zadziałaniu (wej. I2) zmniejszany jest o 1 stan <i>Licznika</i> wózków
10	A C2 = Q3	Jeśli stan <i>Licznika</i> wózków C2 jest równy bądź większy od 1 to załączone jest wyjście Q3, ponieważ dla C1 wartość zadana to 1. Lampka X1 świeci, jeśli wózek jest w trakcie obrotu

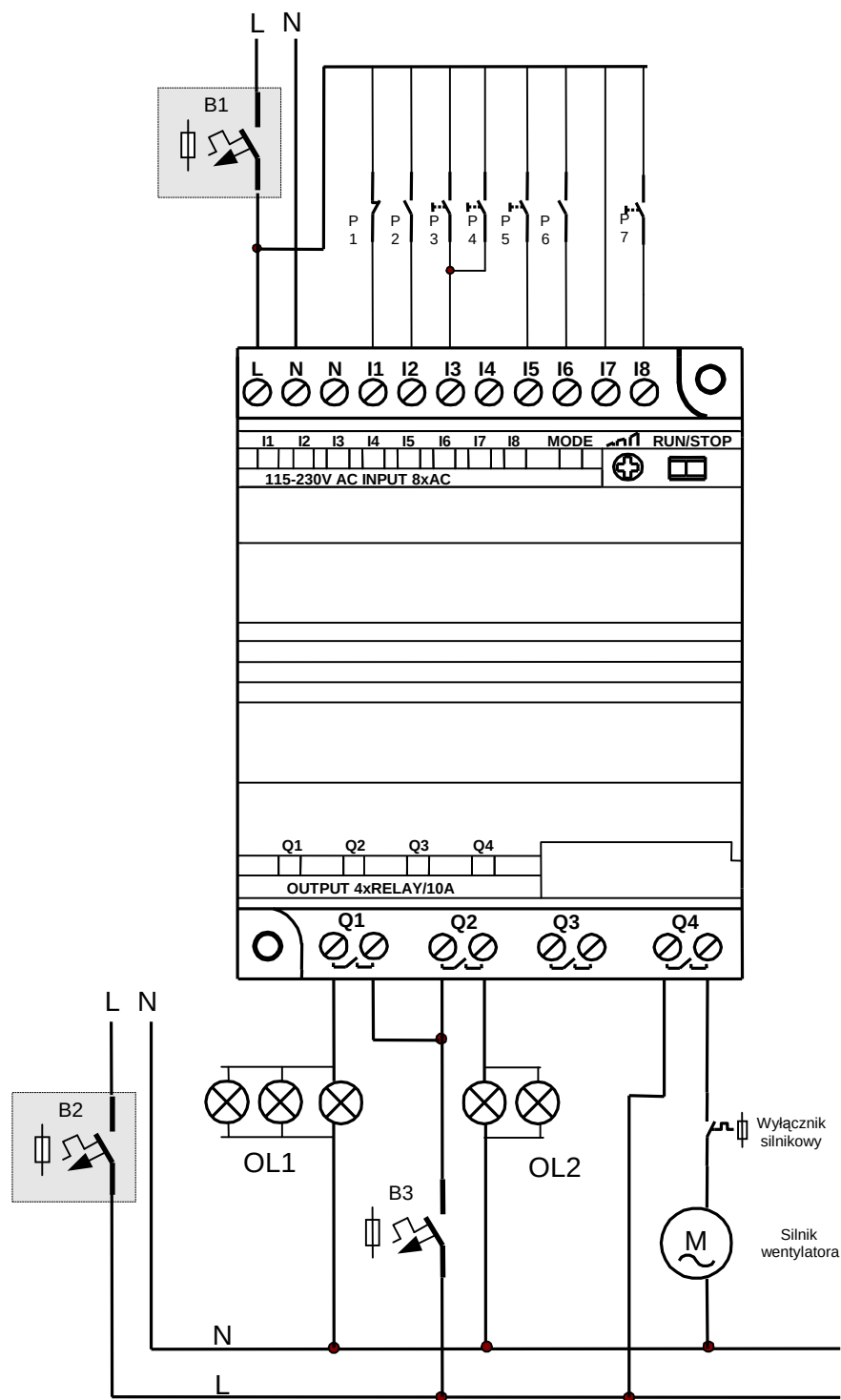
Komentarz do programu

Sytuacja wyjściowa (przed uruchomieniem programu w przekaźniku programowalnym) jest następująca – siłownik S1 BLOKADA (sterowany elektrozaworem Q1) jest ciągle wysunięty. Po uruchomieniu przekaźnika programowalnego (START) analizowany jest stan wejść i wyjść układu. Następnie linia po linii wykonywane są instrukcje programu – opis w komentarzu obok programu.

11.6. Sterownik oświetlenia i wentylacji

Opis zadania

Zadaniem przedstawionego układu jest kontrola oświetlenia np. biura, hali produkcyjnej, sklepu itp. Często się zdarza, że wychodząc do domu zapominamy o wyłączeniu zbędnego oświetlenia lub o załączeniu oświetlenia tzw. nocnego, które jest niezbędne dla ochrony obiektu. Dodatkowo, dzięki sygnalizacji diod LED umieszczonych na przekaźniku programowalnym, mamy informację o załączonych obwodach i działaniu przycisków. Układ umożliwia centralne wyłączenie i załączenie napięcia, ręcznie lub automatycznie o określonej godzinie lub np. po załączeniu/wyłączeniu zewnętrznego alarmu.



Rys. 11.6.1. Schemat układu połączeń.

Realizacja.

Do załączania użyjemy łączników chwilowych, tzn. takich, które przewodzą prąd jedynie w chwili naciskania. Dzięki możliwości ich oprogramowania możemy ich użyć w ten sposób, że pierwsze naciśnięcie załączy obwód, drugie go rozłączy, nie blokując możliwości automatycznego wyłączenia/załączenia. Dodatkowy przełącznik (posiadający dla odmiany dwa stany stabilne) umożliwi zmianę rodzaju pracy z automatycznej na ręczną i odwrotnie. W trybie ręcznym układ nie reaguje na centralne wyłączenie.

Dzięki wykorzystaniu przekaźnika programowalnego uzyskujemy komfort obsługi, oszczędność energii oraz możliwość modyfikacji układu bez wykonywania zmian w instalacji. Przedstawione rozwiązanie pokazuje jak bardzo elastycznie można „kształtować” funkcje oświetlenia każdego pomieszczenia.

W układzie jak na rys. 11.6. zastosowano elementy:

P1 – styk wyłącznika awaryjnego.

P2 – przełącznik trybu pracy.

P3, P4 – łączniki dla obwodu lamp L1.

P5 – łącznik dla obwodu lamp L2.

P6 – styk od układu alarmowego (niezależnie działający układ alarmowy).

P7 – załączanie wentylatora – łącznik chwilowy.

wejście I7 – kontrola napięcia zasilającego.

Q1 – sterowanie obwodu OL1.

Q2 – sterowanie obwodu OL2.

Q4 – załączanie silnika wentylatora.

Działanie:

Rozwarcie styku P1 unieruchamia wszystkie obwody wyjściowe.

P2 rozarty – tryb ręczny, P2 zwarty tryb automatyczny.

P3 lub P4 – pierwsze naciśnięcie załączy układ Obwód Lamp 1, kolejne rozłączy.

P5 – działanie jak P3 i P4 tyle, że dla obwodu Obwód Lamp 2.

P6 – styk informujący o uzbrojeniu zewnętrznego alarmu.

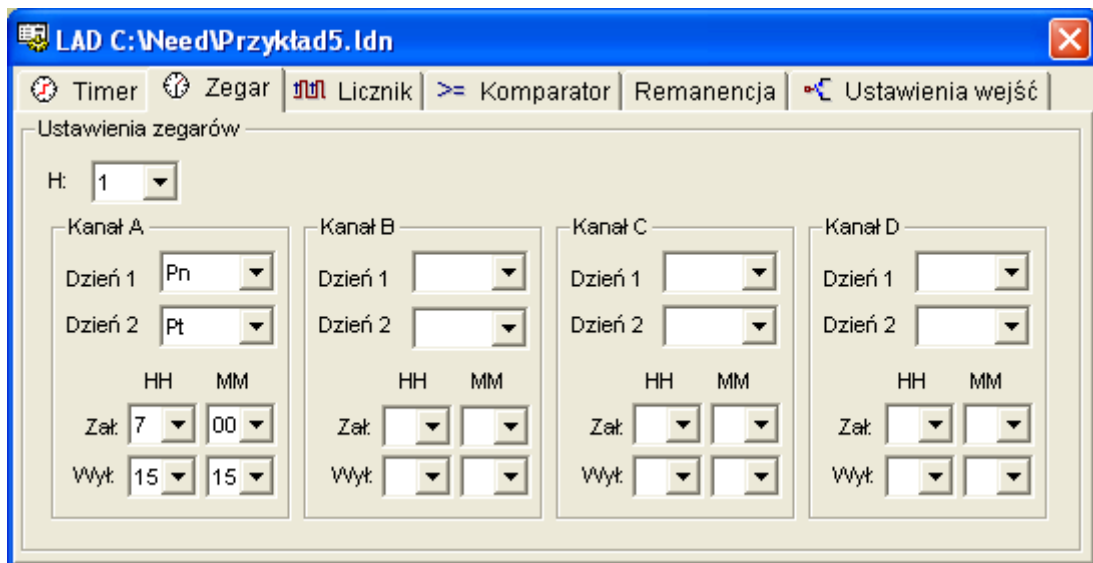
P7 – wł/wył wentylatora.

W trybie automatycznym oświetlenie jest wyłączane/załączane stykiem alarmu lub zgodnie z ustawionym *Zegarem* lub przyciskami P3, P4, P5.

W trybie ręcznym na oświetlenie działają jedynie przyciski P3, P4, P5.

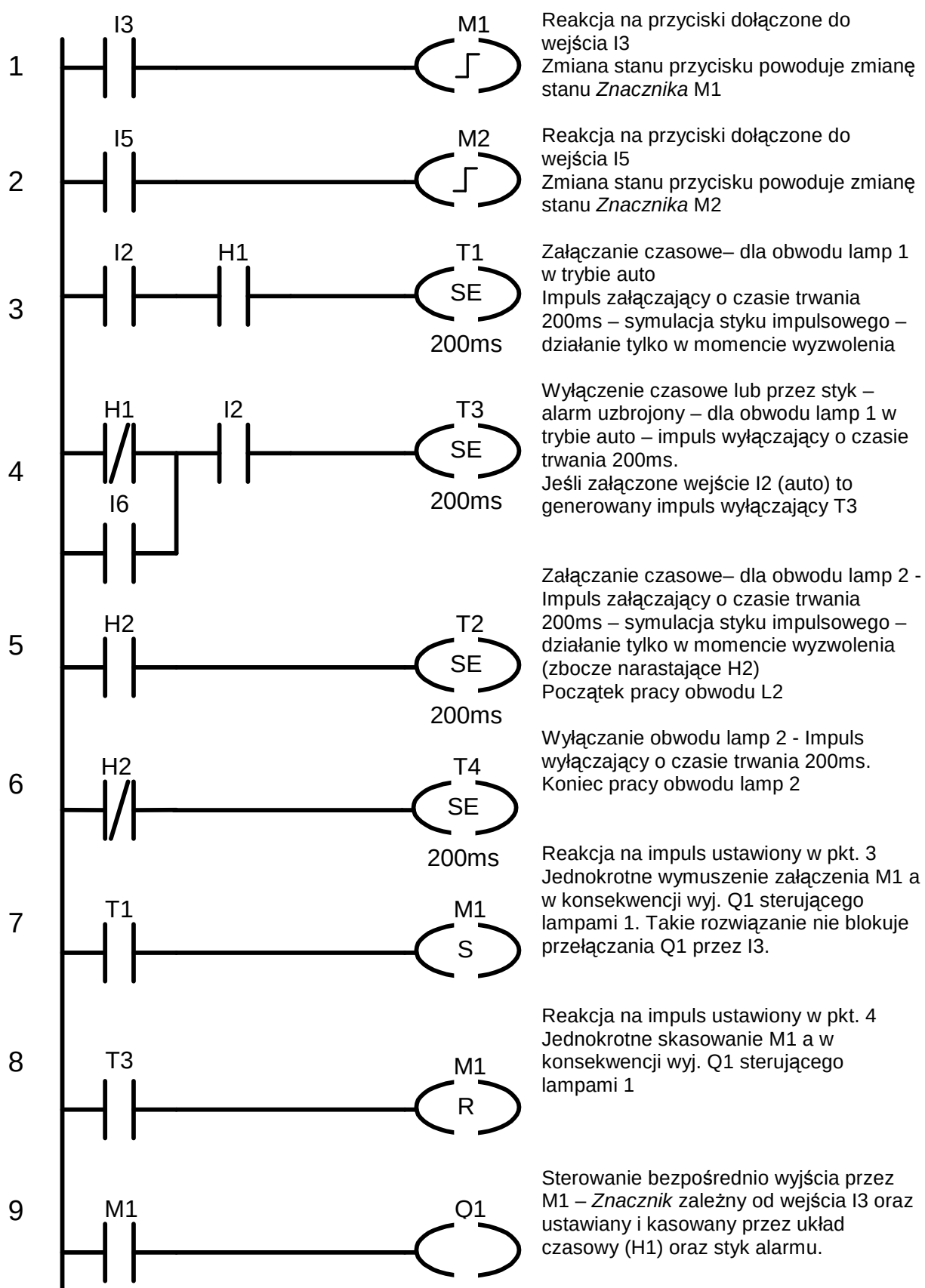
Wentylator wyciągowy działa w ustawionych godzinach pracy.

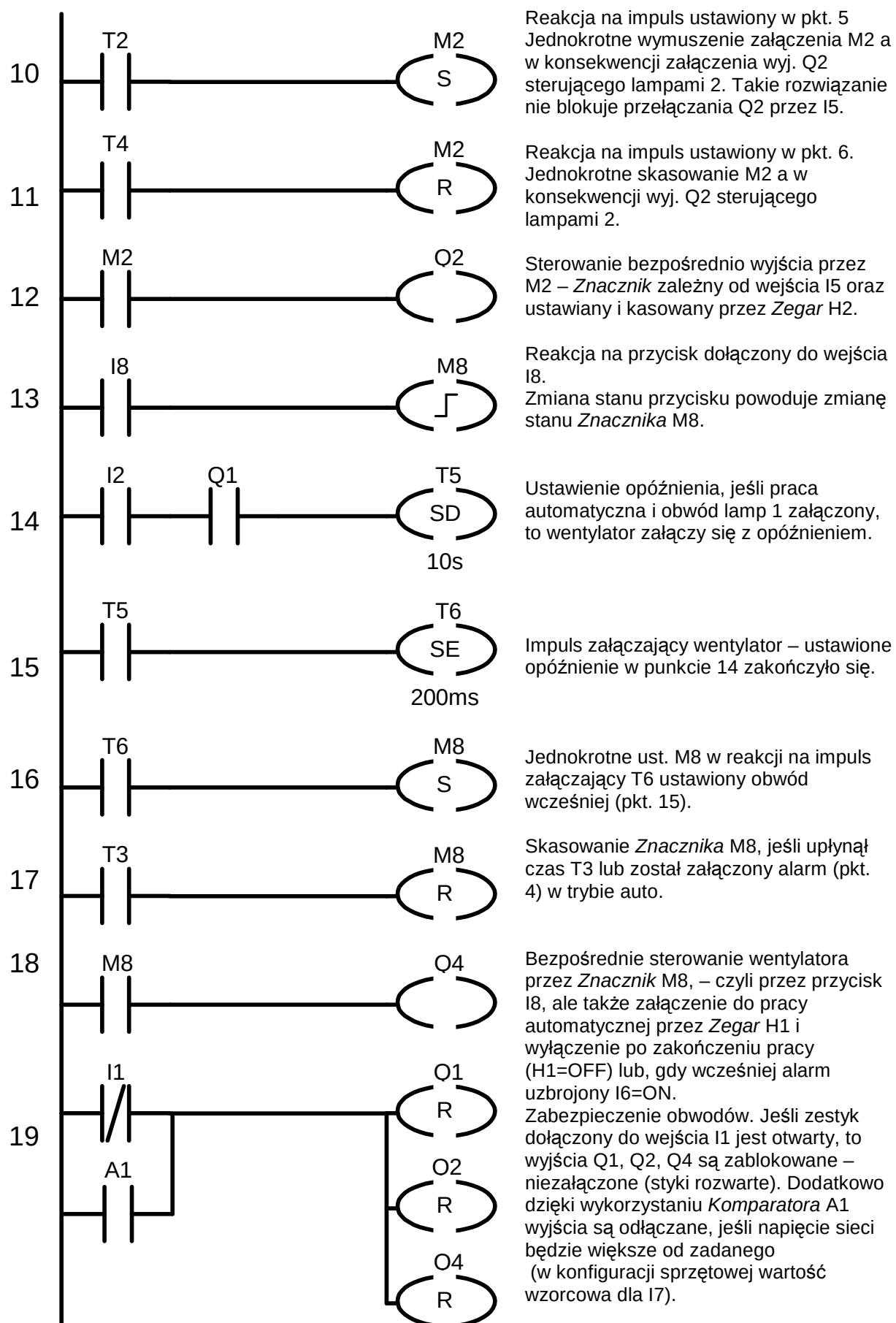
Poniżej przedstawiono przykładową konfigurację *Zegara* ustawionego na załączanie codziennie od poniedziałku do piątku w godz. 7.00 do 15.15.



Rys. 11.6.2. Konfiguracja Zegara

Program w LAD





Program w STL

1	// Zał/wył obwodu 1 - przyciski A I3 FP M1	Reakcja na przyciski dołączone do wejścia I3. Zmiana stanu przycisku powoduje zmianę stanu <i>Znacznika</i> M1.
2	// Zał/wył obwodu 2 - przycisk A I5 FP M2	Reakcja na przyciski dołączone do wejścia I5. Zmiana stanu przycisku powoduje zmianę stanu <i>Znacznika</i> M2.
3	// Załączanie czasowe automat. // Obwód L1 - Impuls załączający A I2 A H1 L 200ms SE T1	Załączanie czasowe – dla obwodu lamp 1 w trybie auto. Impuls załączający o czasie trwania 200ms – symulacja styku impulsowego – działanie tylko w momencie wyzwolenia.
4	// Wyłączanie czasowe lub przez alarm // Obwód L1 w trybie auto AN H1 O I6 = M10 A I2 A M10 L 200ms SE T3	Wyłączenie czasowe lub przez styk – alarm uzbrojony – dla obwodu lamp 1 w trybie auto – impuls wyłączający o czasie trwania 200ms. M10 – suma logiczna czasu wyłączenia (H1) i I6 – wcześniej ktoś uzbroił alarm. Jeśli załączone wejście I2 (auto) to generowany impuls wyłączający T3.
5	// Załączanie czasowe // Obwód L2 - Impuls załączający A H2 L 200ms SE T2	Załączanie czasowe – dla obwodu lamp 2 – Impuls załączający o czasie trwania 200ms – symulacja styku impulsowego – działanie tylko w momencie wyzwolenia → H2 =1. Początek pracy obwodu L2.
6	// Wyłączanie czasowe // Obwód L2 AN H2 L 200ms SE T4	Wyłączanie obwodu lamp 2 – Impuls wyłączający o czasie trwania 200ms. Koniec pracy obwodu lamp 2.
7	// Załączanie obwodu 1 A T1 S M1	Reakcja na impuls ustawiony w pkt. 3. Jednokrotne wymuszenie załączenia M1a w konsekwencji Wyj. Q1 sterującego lampami 1. Takie rozwiązanie nie blokuje przełączania Q1 przez I3.
8	// Wyłączanie obwodu 1 A T3 R M1	Reakcja na impuls ustawiony w pkt. 4. Jednokrotne skasowanie M1 a w konsekwencji wyj. Q1 sterującego lampami 1.
9	// Wyjście Q1 O M1 = Q1	Sterowanie bezpośrednio wyjścia przez M1 – <i>Znacznik</i> zależny od wejścia I3 oraz ustawiany i kasowany przez <i>Zegar</i> H1 oraz styk alarmu.

10	// Załączanie obwodu 2 A T2 S M2	Reakcja na impuls ustawiony w pkt. 5. Jednokrotne wymuszenie załączenia M2 a w konsekwencji wyj. Q2 sterującego lampami 2. Takie rozwiązanie nie blokuje przełączania Q2 przez I5.
11	// Wyłączanie obwodu 2 A T4 R M2	Reakcja na impuls ustawiony w pkt. 6. Jednokrotne skasowanie M2 a w konsekwencji wyj. Q2 sterującego lampami 2.
12	// Wyjście Q2 A M2 = Q2	Sterowanie bezpośrednio wyjścia przez M2 – <i>Znacznik</i> zależny od wejścia I5 oraz ustawiany i kasowany przez <i>Zegar</i> H2.
13	// Załączenie przycisku P8 // - Wentylator A I8 FP M8	Reakcja na przycisk dołączony do wejścia I8. Zmiana stanu przycisku powoduje zmianę stanu <i>Znacznika</i> M8.
14	// Załączenie - Wentylator // Ustawienie opóźnienia A I2 A Q1 L 10s SD T5	Ustawienie opóźnienia, jeśli praca automatyczna i obwód lamp 1 załączony, to wentylator załączy się z opóźnieniem.
15	// Impuls załączający A T5 I 200ms SE T6	Impuls załączający wentylator – ustawione opóźnienie zakończyło się.
16	// Zał. wentylatora automatyczne A T6 S M8	Jednokrotne ust. M8 w reakcji na impuls załączający T6 ustawiony powyżej (pkt. 15).
17	// Wyłączenie wentylatora // Automatyczne czasowe lub przez alarm A T3 R M8	Jednokrotne skasowanie <i>Znacznika</i> M8, jeśli upłynął czas H1 lub został załączony alarm (pkt. 4) w trybie auto.
18	// Wyjście Q4 A M8 = Q4	Bezpośrednie sterowanie wentylatora przez <i>Znacznik</i> M8 – czyli przez przycisk I8, ale także załączenie do pracy automatyczne przez <i>Zegar</i> H1 i wyłączenie po zakończeniu pracy (H1=OFF) lub gdy wcześniej alarm uzbrojony I6=ON.
19	// Zabezpieczenie/wyłączenie AN I1 O A1 R Q1 R Q2 R Q4	Zabezpieczenie obwodów. Jeśli zestyk dołączony do wejścia I1 jest otwarty, to wyjścia Q1, Q2, Q4 są zablokowane – niezałączone (styki rozwarte). Dodatkowo dzięki wykorzystaniu <i>Komparatora</i> A1 wyjścia są odłączane, jeśli napięcie sieci będzie większe odadanego (w konfiguracji sprzętowej wartość wzorcowa dla I7).

Komentarz do programu.

Jest to tylko jeden z możliwych wariantów użycia przekaźnika programowalnego NEED, mający pokazać użycie różnych instrukcji. np. funkcja FP pozwala na naturalne posługiwanie się łącznikami chwilowymi jako załącznikami/wyłącznikami oświetlenia.

Użycie wbudowanego zegara stwarza szereg możliwości czasowego sterowania układami. Wykorzystanie wejścia analogowego pozwala zabezpieczyć obwody sterowane przed skutkami niewłaściwego poziomu napięć zasilających (pod warunkiem zasilania obwodów wykonawczych z tej samej fazy co napięcie zasilające przekaźnik programowalny).

Dla uproszczenia założmy, że zastosowany dodatkowy element przekładnika prądowego, który będzie przetwarzał prąd płynący w obwodzie głównym na proporcjonalne napięcie. Im wyższy pobór prądu tym wyższe napięcie po stronie wtórnej. Pamiętając o ograniczeniach sprzętowych możemy kontrolować moc do 2300W ($I=10A$, $U_z = 230VAC$). Zakładając, że napięciu 100V na uzwojeniu wtórnym przekładnika odpowiada przepływ 10A w obwodzie kontrolowanym, możemy ustawić wartość prądu np. 5A poprzez wpisanie jako wartości wzorcowej 50. Zamiast wartości wzorcowej możemy użyć wbudowanego Potencjometru i wtedy ręcznie ustawić żadaną wartość. Korzystając z funkcji *Komparatora* analogowego A2 ($I7 \geq$ wartość wzorcowa) możemy sterować wyjściem Q1 w zależności od pobieranego prądu. Po przekroczeniu żadanego prądu ($I7 \geq 50V$) wyjście odłącza obciążenie, co jest sygnalizowane wyjściem Q2. Ponowne załączenie jest możliwe po 10 sekundach od wyłączenia (oczywiście można ustawić dowolny czas).

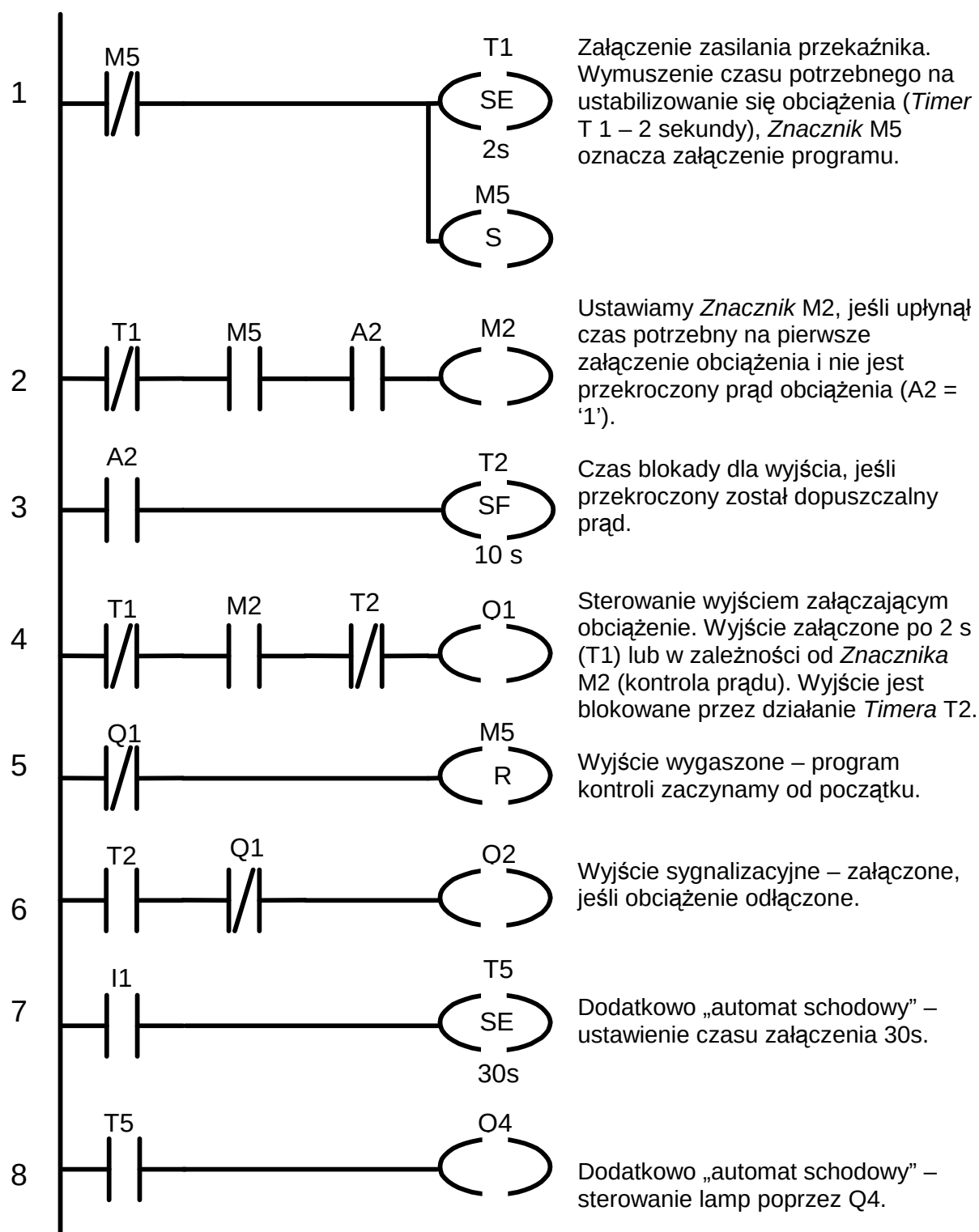
Program STL

- | | | |
|---|--------------------------------|--|
| 1 | AN M5
L 2S
SE T1
S M5 | Załączenie zasilania przełącznika. Wymuszenie czasu potrzebnego na ustabilizowanie się obciążenia (<i>Timer</i> T 1 – 2 sekundy), <i>Znacznik</i> M5 oznacza załączenie programu. |
| 2 | AN T1
A M5
A A2
= M2 | Ustawiamy <i>Znacznik</i> M2, jeśli upłynął czas potrzebny na pierwsze załączenie obciążenia i nie jest przekroczony prąd obciążenia ($A2 = '1'$). |
| 3 | A A2
L 10s
SF T2 | Czas blokady dla wyjścia, jeśli przekroczony został dopuszczalny prąd. |
| 4 | AN T1
A M2
AN T2
= Q1 | Sterowanie wyjściem załączającym obciążenie. Wyjście załączone po 2 s od załączenia (T1) lub w zależności od <i>Znacznika</i> M2 (kontrola prądu). Wyjście jest blokowane przez działanie zwłoki czasowej (T2) po przekroczeniu warunku <i>Komparacji</i> A2 (przekroczona wartość prądu). |
| 5 | AN Q1
R M5 | Wyjście wygaszone – program kontroli zaczynamy od początku. |
| 6 | A T2
AN Q1
=Q2 | Wyjście sygnalizacyjne – załączone, jeśli obciążenie odłączone. |
| 7 | A I1
L 30s
SE T5 | Dodatkowo „automat schodowy” – ustawienie czasu załączenia 30s. |
| 8 | A T5
= Q4 | Dodatkowo „automat schodowy” – sterowanie lamp poprzez Q4. |

Uwaga do programu

Należy pamiętać, aby w konfiguracji *Znacznik* M5 nie był *Znacznikiem* remanentnym – po załączeniu zasilania nie powinien być zapamiętywany jego stan.

Program LAD



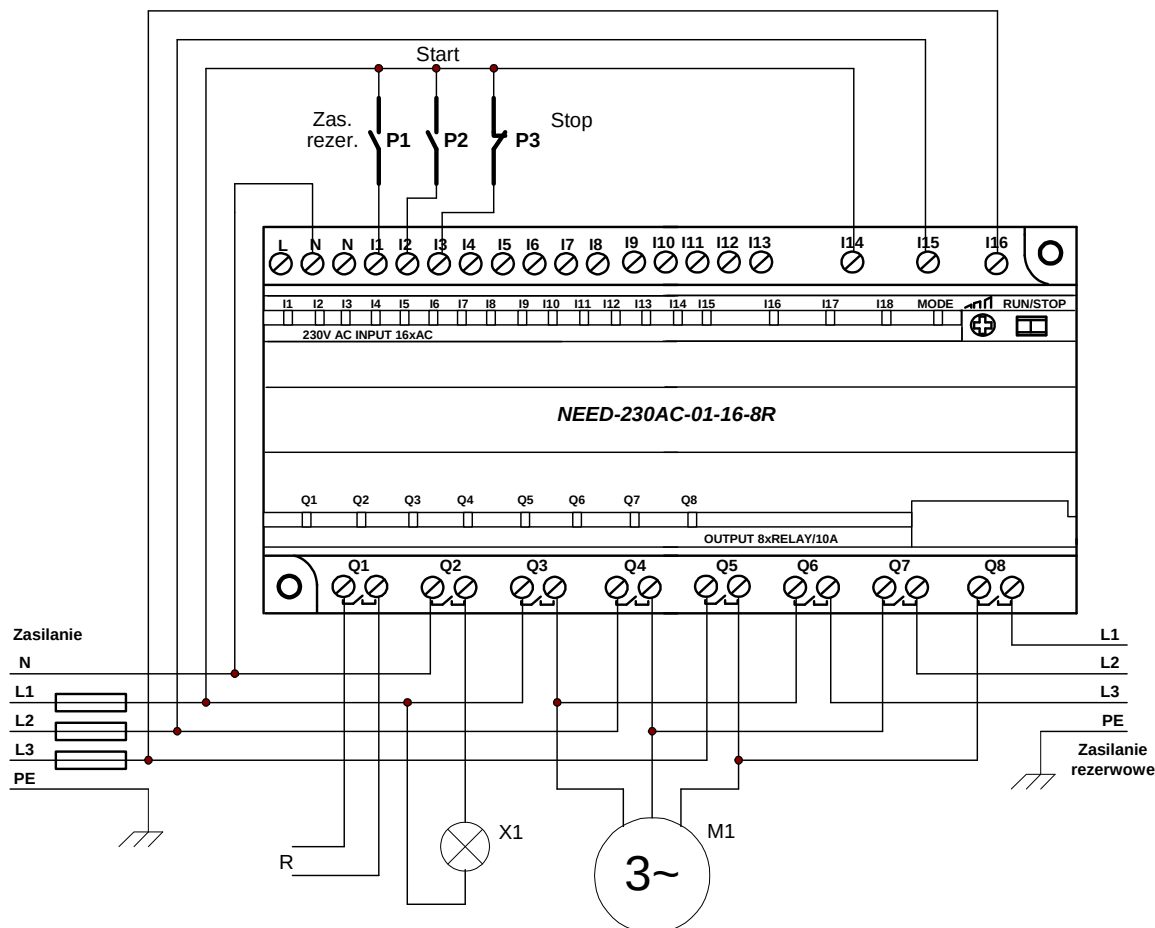
Komentarz do programu

Obwody 1..6 odnoszą się do schematu na rys. 11.7. Obwody 7 i 8 pokazują dalsze możliwości wykorzystania wolnych wejść/wyjść do zrealizowania prostego sterowania czasowego oświetleniem klatki schodowej.

11.8. Sterowanie i zabezpieczenie silnika 3-fazowego

Opis zadania.

Zadaniem przedstawionego układu jest sterowanie silnika i nadzór nad parametrami sieci zasilającej silnik 3-fazowy małej mocy. Dzięki możliwościom przekaźnika NEED 230AC-11-16-8 możemy bez żadnych dodatkowych elementów zrealizować załączanie/wyłączenie silnika 3-fazowego, kontrolować parametry zasilającej i w razie potrzeby przełączyć na zasilanie rezerwowe.



Rys. 11.8.1. Schemat podłączeń elektrycznych.

Włączanie / wyłączanie silnika odbywa się za pomocą przycisków START / STOP.

Przekaźnik „Zasilanie rezerwowe” umożliwia wybór trybu pracy:

- nie załączony – w razie nieprawidłowości sieci zasilającej silnik zostanie odłączony.
- załączony - w razie nieprawidłowości sieci zasilającej silnik będzie pracował na zasilaniu rezerwowym i powróci automatycznie do pracy na zasilaniu sieciowym, jeśli parametry sieci zasilającej wrócą do zadanych granic.

Wbudowana w przekaźnik funkcjonalność umożliwia wykonanie przekaźnika nadzorczego sieci zasilającej spełniającego następujące funkcje:

- a) Kontrola poziomów napięć poszczególnych faz (minimalne U_{min} i maksymalne napięcie zasilające U_{max})
- b) Nadzór kolejności faz
- c) Nadzór zaniku fazy
- d) Nadzór asymetrii – U_{asym}

Powyższe parametry mogą być dowolnie konfigurowane w ustawieniach programu.

Możemy ustawić minimalne i maksymalne napięcie oddzielnie dla każdej fazy, minimalny i maksymalny poziom asymetrii. Modyfikując program możemy zrezygnować z kontroli wybranych parametrów, jeśli jest to nieistotne dla sterowanego układu.

Program STL

Program zbudowany jest z trzech części:

1. Określenie typu przekaźnika
2. Definicja nazw symbolicznych
3. Program sterujący (kod źródłowy)

```

1 // Typ przekaźnika
  .DEVICE="230AC-X1-16-8"

2 // Zmienne symboliczne
  .DEFINE A1_L1min=A1
  .DEFINE A2_L2min=A2
  .DEFINE A3_L3min=A3
  .DEFINE A4_L1max=A4
  .DEFINE A5_L2max=A5
  .DEFINE A6_L3max=A6
  .DEFINE A7_Asym=A7
  .DEFINE A8_Asym=A8
  .DEFINE I1_Zal_rez=I1
  .DEFINE I2_START=I2
  .DEFINE I3_STOP=I3
  .DEFINE M1_Umin=M1
  .DEFINE M2_Umax=M2
  .DEFINE M3_Uzas=M3
  .DEFINE M4_Asym=M4
  .DEFINE M5_Zas_OK=M5
  .DEFINE M6_ZALACZ=M6
  .DEFINE Q1_R=Q1
  .DEFINE Q2_Sygn=Q2
  .DEFINE Q3_L1=Q3
  .DEFINE Q4_L2=Q4
  .DEFINE Q5_L3=Q5
  .DEFINE Q6_L1rez=Q6
  .DEFINE Q7_L2rez=Q7
  .DEFINE Q8_L3rez=Q8
  .DEFINE T1_Opozn=T1
  .DEFINE T2_sygn=T2

```

```

3 // Program
  A %A1_L1min
  A %A2_L2min
  A %A3_L3min
  = %M1_Umin

```

Znacznik M1 ustawiony jeśli napięcie każdej fazy jest większe od Umin.

```

  A %A4_L1max
  A %A5_L2max
  A %A6_L3max
  = %M2_Umax

```

Znacznik M2 ustawiony jeśli napięcie każdej fazy jest mniejsze od Umax.

```

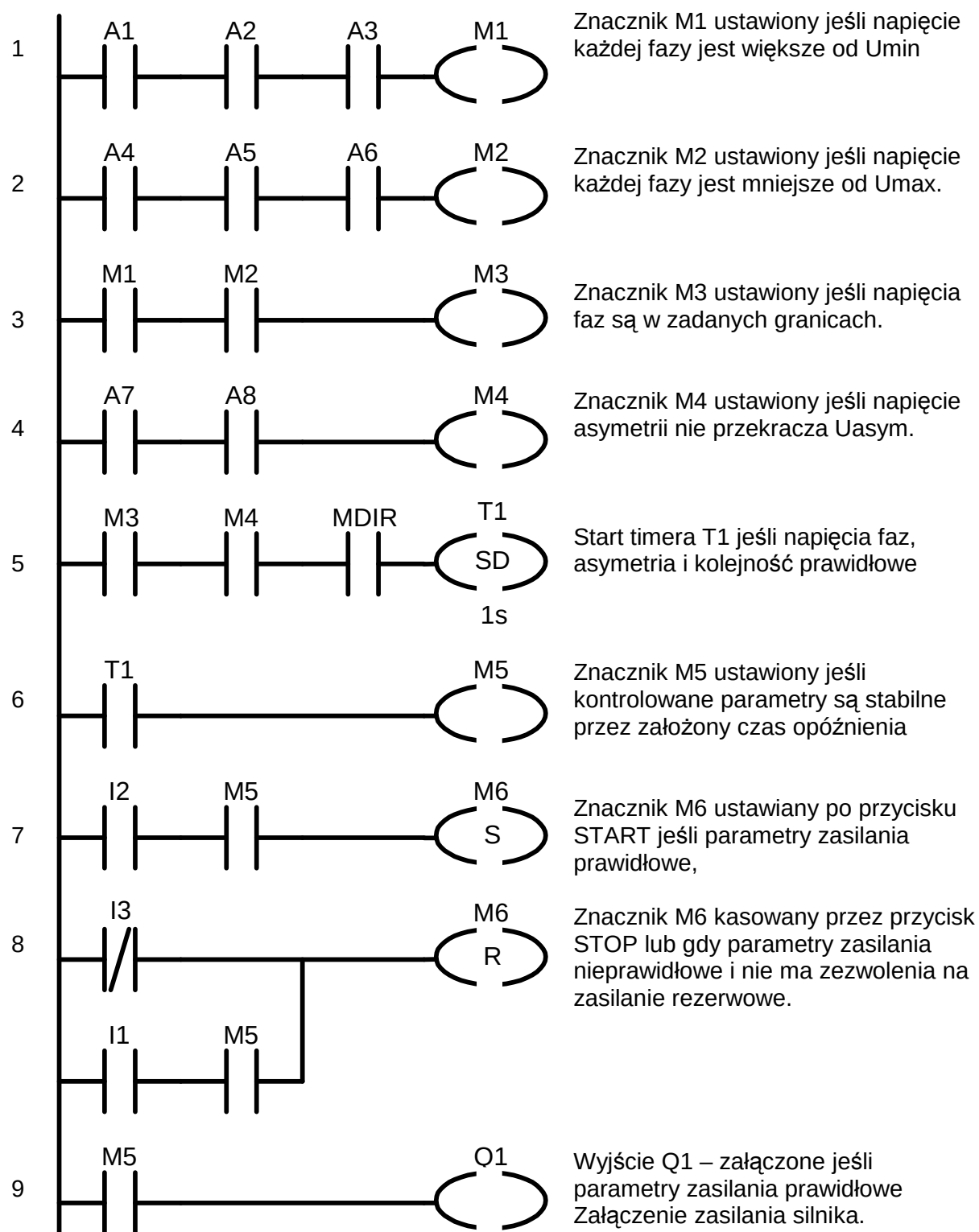
  A %M1_Umin
  A %M2_Umax
  = %M3_Uzas

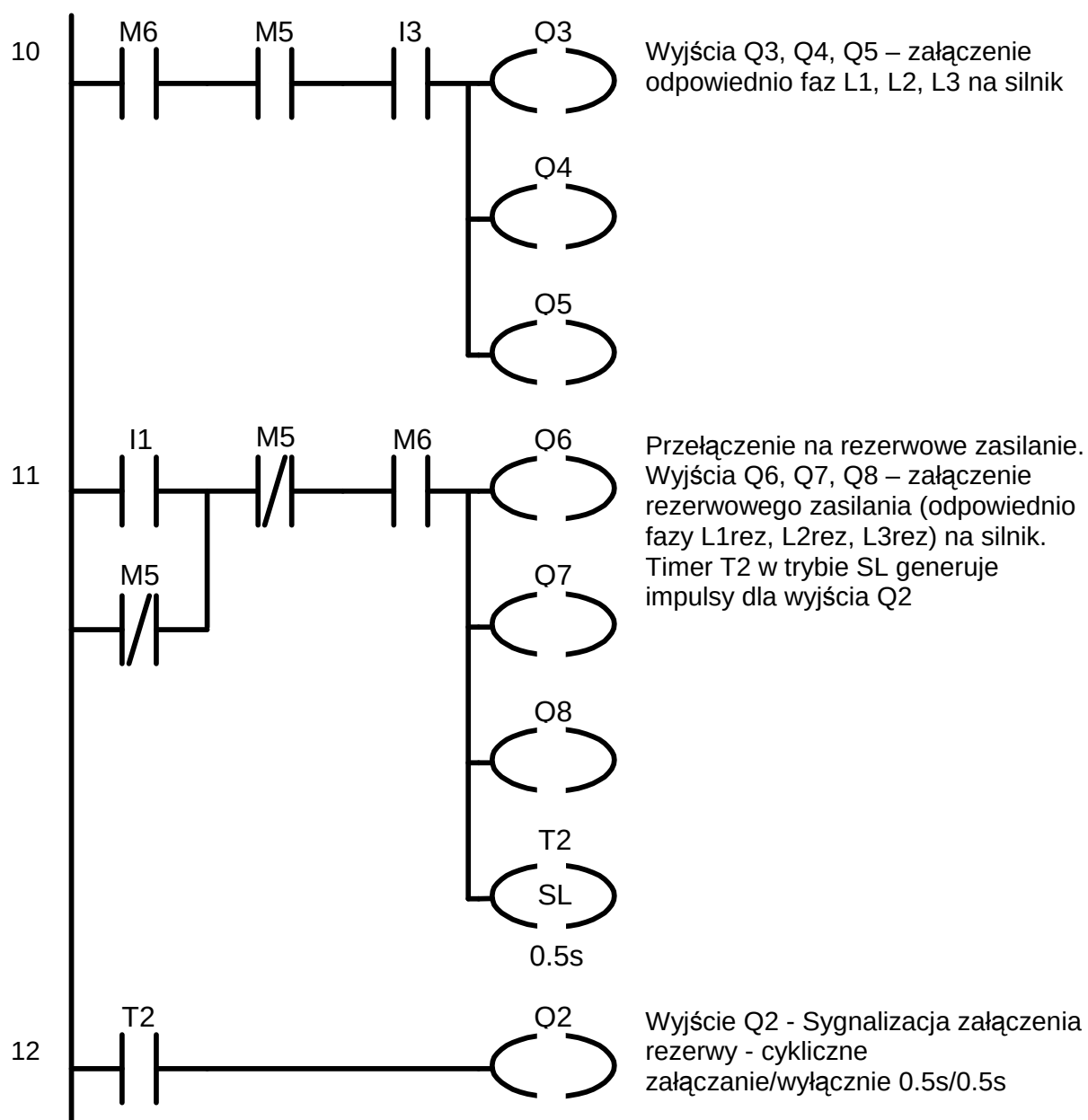
```

Znacznik M3 ustawiony jeśli napięcia faz są w zadanych granicach.

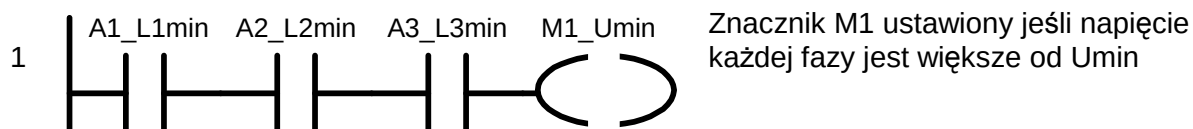
<pre> 3 A %A7_Asym A %A8_Asym = %M4_Asym A %M3_Uzas A %M4_Asym A MDIR SD %T1_Opozn // 1.00s A %T1_Opozn = %M5_Zas_OK A %I2_START A %M5_Zas_OK S %M6_ZALACZ A(ON %I3_STOP O(AN %I1_Zal_rez AN %M5_Zas_OK)) R %M6_ZALACZ A %M5_Zas_OK = %Q1_R A %M6_ZALACZ A %M5_Zas_OK A %I3_STOP = %Q3_L1 = %Q4_L2 = %Q5_L3 A(O %I1_Zal_rez ON %M5_Zas_OK) AN %M5_Zas_OK A %M6_ZALACZ = %Q6_L1rez = %Q7_L2rez = %Q8_L3rez SL %T2_sygn // 0.50s A %T2_sygn = %Q2_Sygn </pre>	Znacznik M4 ustawiony jeśli napięcie asymetrii nie przekracza Uasym. Start timera T1 jeśli napięcia faz, asymetria i kolejność prawidłowe Znacznik M5 ustawiony jeśli kontrolowane parametry są stabilne przez założony czas opóźnienia Znacznik M6 ustawiany po przycisku START jeśli parametry zasilania prawidłowe, kasowany przez przycisk STOP lub gdy parametry zasilania nieprawidłowe i nie ma zezwolenia na zasilanie rezerwowe. Wyjście Q1 – załączone jeśli parametry zasilania prawidłowe Załączenie zasilania silnika. Wyjścia Q3, Q4, Q5 – załączenie odpowiednio faz L1, L2, L3 na silnik Przełączenie na rezerwowe zasilanie. Wyjścia Q6, Q7, Q8 – załączenie rezerwowego zasilania (odpowiednio fazy L1rez, L2rez, L3rez) na silnik. Timer T2 w trybie SL generuje impulsy dla wyjścia Q2 Wyjście Q2 - Sygnalizacja załączenia rezerwy - cykliczne załączanie/wyłączanie 0.5s/0.5s.
---	--

Program LAD





Program LAD można również zaopatrzyć w nazwy symboliczne (podobnie jak w STL). Jeśli przypiszemy symbole rejestrów i przełączymy widok na nazwy symboliczne to zamiast nazwy rejestru będzie widoczna nazwa symboliczna. Poniżej przykład dla pierwszej linii.



Ustawienia

Parametry sieci zasilającej są kontrolowane poprzez wejścia analogowe I14, I15, I16. Napięcia minimalne U_{min} i maksymalne U_{max} dla poszczególnych faz są ustawiane w komparatorach A1..A6 odpowiednio:

A1: AI14 $\geq$ 200V

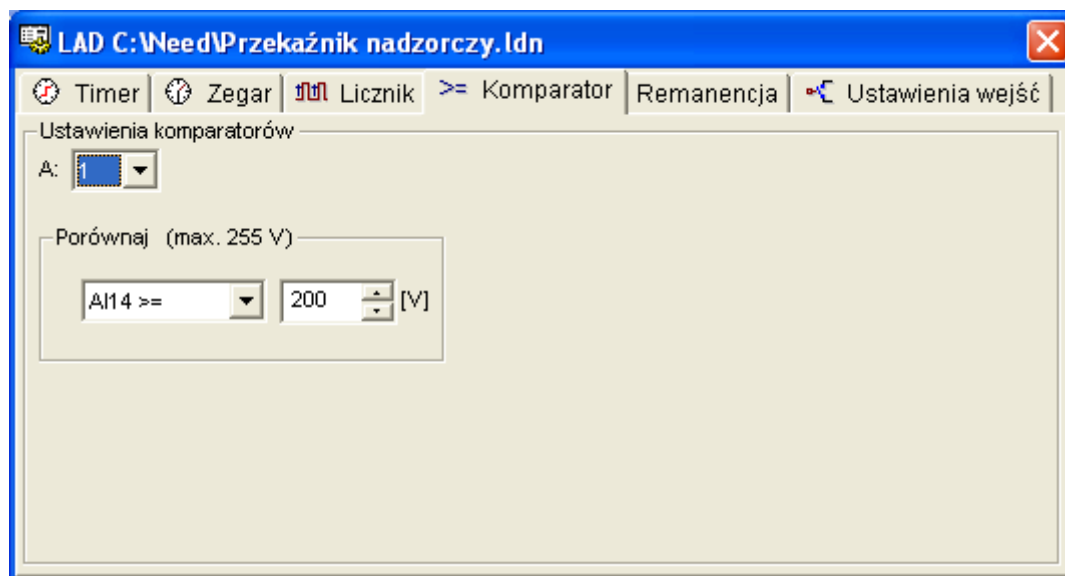
A2: AI15 $\geq$ 200V

A3: AI16 $\geq$ 200V

A4: AI14 $\leq$ 240V

A5: AI14 $\leq$ 240V

A6: AI14 $\leq$ 240V



Rys. 11.8.1. Ustawianie komparatora.

Wszystkie fazy mają ustawione takie samo napięcie minimalne 200V i maksymalne 240V. Można oczywiście ustawić je dowolnie.

Komparatory A7 i A8 służą do zadania minimalnego i maksymalnego napięcia asymetrii 0..10V.

A7: ASYM $\geq$ 0V

A8: ASYM $\leq$ 10V

Dodatkowo ustawić możemy czas opóźnienia Timera T1, czyli minimalny czas stabilności prawidłowych parametrów sieci oraz czas impulsów sygnalizacji T2.

12. DANE TECHNICZNE

Dane ogólne

Opis przekaźnika NEED...-08-4R	6 wejść cyfrowych AC 2 wejścia analogowo - cyfrowe 4 wyjścia cyfrowe przekaźnikowe NO Zegar Czasu Rzeczywistego
NEED...-16-8R NEED...-16-8R	13 wejść cyfrowych AC 3 wejścia analogowo - cyfrowe 8 wyjść cyfrowych przekaźnikowych NO Zegar Czasu Rzeczywistego Szybki Licznik Kontrola sieci trójfazowej (wejścia analogowe)
Możliwość podłączenia wyświetlacza z klawiaturą NEED...-x1-08-4R NEED...-22-08-4R NEED...-11-16-8R NEED...-22-16-8R	Nie Tak Nie Tak
Użytkowanie	W instalacjach niskiego napięcia
Napięcie zasilania NEED-230AC-.. NEED-220DC-.. NEED-24DC-.. NEED-12DC-..	95V ÷ 260V AC, 50Hz/60Hz 154V – 242V DC 19,6V ÷ 28,8V DC 10,2V ÷ 14,4V DC
Montaż: Położenie Mocowanie Miejsce pracy	Dowolne Na szynie montażowej o szerokości 35mm lub 2 wkręty Ø 4mm W szafie sterowniczej, rozdzielnicy instalacyjnej zgodnej z PN-EN 61131-2
Instalacja	Urządzenie może być montowane tylko przez osobę znającą montaż instalacji elektrycznych
Przewody łączeniowe	1×2,5mm ² 2×1mm ²
Maksymalny moment dokręcania śrub zacisków przyłączeniowych	0,6Nm
Zgodność z normą	PN-EN 61131-2
Certyfikaty	CE, B, UL, VDE, GOST
Wymiary: NEED...-08-4R: Szerokość Długość Wysokość	72mm 90mm 55mm

Masa: NEED-...x1-08-4R NEED-...22-08-4R-D	210g 250g
NEED-...16-8R: Szerokość Długość Wysokość	132mm 90mm 55mm
Masa: NEED...11-16-8R NEED...22-16-8R-D	370g 413g

Warunki i wymagania klimatyczne i mechaniczne

Temperatura robocza	Od -20°C do +55°C
Wilgotność względna	Od 10% do 95%, bez kondensacji
Ciśnienie atmosferyczne	795hPa do 1080hPa
Stopień zanieczyszczenia	2
Drgania dopuszczalne w stanie roboczym (PN-EN 60068-2-6)	5Hz do 9Hz (stała amplituda 3,5mm) 9Hz do 150Hz (stałe przyspieszenie 1g)
Udary (PN-EN 60068-2-27)	6 uderzeń (półsinusoida 15g/11ms)
Upadki płaskie i podtrzymywane (PN-EN 60068-2-31)	100mm, 2 próby 30°, 2 próby
Temperatura składowania/transportu	-40°C do +70°C
Swobodne spadki (PN-EN 60068-2-32): wyrób w opakowaniu transportowym wyrób w opakowaniu sprzedażnym	1 000mm 300mm

Wymagania bezpieczeństwa

Znamionowe napięcie izolacji:	300V AC
Znamionowe napięcie udarowe	2 500V
Kategoria przepięciowa:	Klasa II
Stopień ochrony obudowy (PN-EN 60529)	IP 20
Klasa odporności ogniowej (UL94)	V0

Wymagania kompatybilności elektromagnetycznej (EMC)

Emisja zaburzeń promieniowych (PN-EN 61131-2 , PN-EN 55011) NEED..-08-4R NEED..-22-16-8R	Klasa wartości granicznej A, grupa 1
NEED..-11-16-8R	Klasa wartości granicznej B, grupa 1
Odporność na wyładowanie elektrostatyczne (PN-EN 61000-4-2)	8kV – wyładowanie powietrzne, 4kV – wyładowanie powierzchniowe
Podatność na pola elektromagnetyczne o częstotliwości radiowej (PN-EN 61000-4-3, PN-EN 61131-2), narażenie 10V/m	80MHz – 1 000MHz 1,4GHz – 2,0GHz
Seria szybkich elektrycznych stanów przejściowych (PN-EN 61000-4-4): NEED-230AC-..	2kV – przewody zasilające 2kV – przewody sygnałowe
NEED-220DC-..., NEED-24DC-..., NEED-12DC-..	2kV – przewody zasilające 1kV – przewody sygnałowe
Udar wysokiej energii (PN-EN 61000-4-5): NEED-230AC-.. port zasilania	2kV – sygnał niesymetryczny 1kV – sygnał symetryczny
porty obwodów wejściowych	2kV – sygnał niesymetryczny 1kV – sygnał symetryczny
NEED-220DC-..., NEED-24DC-..., NEED-12DC-.. port zasilania	1kV – sygnał niesymetryczny 0,5kV – sygnał symetryczny
porty obwodów wejściowych (linie nieekranowane)	0,5kV – sygnał niesymetryczny 0,5kV – sygnał symetryczny
Odporność na zaburzenia o częstotliwości radiowej, narażenie 3V/m	26 – 80MHz

Obwód zasilania

Napięcie zasilania: NEED-230AC-.. wartość znamionowa zakres pracy	115V / 230V AC, 60Hz/50Hz 95V – 260V
NEED-220DC-.. wartość znamionowa zakres pracy	220V DC 154V – 242V DC
NEED-24DC-.. wartość znamionowa zakres pracy	24V DC 19,6V – 28,8V DC
NEED-12DC-.. wartość znamionowa zakres pracy	12V DC 10,2V – 14,4V DC
Prąd znamionowy (dla wysokich stanów na wszystkich wejściach i wyjściach) NEED-230AC-...-08-4R, NEED-230AC-22-08-4R-D NEED-220DC-...-08-4R, NEED-24DC-...-08-4R, NEED-24DC-22-08-4R-D NEED-12DC-...-08-4R, NEED-22DC-22-08-4R-D	< 22mA < 14mA < 125mA < 250mA
NEED-230AC-...-16-8R, NEED-230AC-22-16-8R-D NEED-220DC-...-16-8R NEED-24DC-...-16-8R, NEED-24DC-22-16-8R-D NEED-12DC-...-16-8R, NEED-12DC-22-16-8R-D	< 44mA < 28mA < 210mA < 420mA
Maksymalny pobór mocy: NEED-230AC-...-08-4R, NEED-230AC-22-08-4R-D. NEED-220DC-...-08-4R NEED-24DC-...-08-4R, NEED-24DC-22-08-4R-D NEED-12DC-...-08-4R , NEED-22DC-22-08-4R-D	< 5VA 8VA < 3W < 3W < 3W
NEED-230AC-...-16-8R, NEED-230AC-22-16-8R-D NEED-220DC-...-16-8R NEED-24DC-...-16-8R, NEED-24DC-22-16-8R-D NEED-12DC-...-16-8R, NEED-12DC-22-16-8R-D	10VA <6W <5W <5W
Wyższe harmoniczne w sygnale zasilania NEED-230AC-..	< 10% wartości napięcia składowej podstawowej
Zabezpieczenie prądowe w obwodzie zasilania NEED-24DC-.., NEED-12DC-..	600mA Zabezpieczenie przed zamianą biegunów
Zabezpieczenie prądowe w obwodzie zasilania NEED-230AC-.., NEED-220DC-..	500mA
Odporność na łagodną i szybką zmianę napięcia zasilania (PN-EN 61131-2)	Tak
Odporność na przerwy napięcia zasilania (PN-EN 61131-2)	20ms
Podtrzymanie Zegara Czasu Rzeczywistego	64h w T=+25°C

	24h w T=+55°C
--	---------------

Specyfikacja obwodów wejściowych

Typ wejść cyfrowych (PN-EN 61131-2)	typ 1 (wejścia odbierające prąd)
Liczba: NEED...-08-4R	8 (I1-I8)
NEED...-16-8R	16 (I1-I16)
Wizualizacja stanu logicznego	diody LED
Napięcie znamionowe: NEED-230AC...-08-4 dla stanu logicznego '1' dla stanu logicznego '0'	85V – 260V 0V – 40V
NEED-230AC...-16-8 dla stanu logicznego '1' dla stanu logicznego '0'	85V – 260V 0V – 32V
NEED-220DC... dla stanu logicznego '1' dla stanu logicznego '0'	80V – 260V 0V – 40V
NEED-24DC... dla stanu logicznego '1' dla stanu logicznego '0'	15 – 40V -3V – 5V
NEED-12DC... dla stanu logicznego '1' dla stanu logicznego '0'	8V – 26V -1,5V – 4V
Prąd wejściowy dla stanu logicznego '1', przy napięciu znamionowym	
NEED-230AC...-08-4	0,6mA (I1 – I4) 8,0mA (I5 – I6) 0,9mA (I7 – I8)
NEED-220DC...-08-4	0,6mA (I1 – I6) 1,1mA (I7 – I8)
NEED-24DC...-08-4	3,3mA (I1 – I6) 2,0mA (I7 – I8)
NEED-12DC...-08-4	3,3mA (I1 – I6) 1,1mA (I7 – I8)
NEED-230AC...-16-8	0,6mA (I1 – I11) 8,0mA (I12 – I13) 1,5mA (I14 – I16)
NEED-220DC...-16-8	0,6mA (I1 – I13) 1,1mA (I14 – I16)

NEED-24DC-...-16-8	3,3mA (I1 – I13) 2,0mA (I14 – I16)
NEED-12DC-...-16-8	3,3mA (I1 – I13) 1,1mA (I14 – I16)
Impedancja wejściowa:	
NEED-230AC-..08-4 I1 – I4 I5 – I6 (charakter pojemnościowy) I7 – I8	400kΩ 28.75kΩ 200kΩ dla półfali dodatniej 400kΩ dla półfali ujemnej
NEED-230AC-...-16-8 I1 – I11 I12 – I13 (charakter pojemnościowy) I14 – I16	400kΩ 28.75kΩ 200kΩ dla półfali dodatniej 400kΩ dla półfali ujemnej
NEED-220DC-.. I1 – I13 I14 – I16	400kΩ 200kΩ
NEED-24DC-.. I1 – I6 I7, I8	7,44kΩ 12,36kΩ
NEED-12DC-.. I1 – I6 I7, I8	3,65kΩ 10,92kΩ
NEED-24DC-...-16-8, NEED-12DC-...-16-8 I14 – I16 dla wejść skonfigurowanych jako prądowe	49Ω
Maksymalny czas opóźnienia przy przejściu ze stanu logicznego '0' na '1': NEED-230AC-.. eliminacja odbić styków ZAŁ eliminacja odbić styków WYŁ	60ms 20ms
NEED-220DC-..., NEED-24DC-..., NEED-12DC-.. eliminacja odbić styków ZAŁ eliminacja odbić styków WYŁ	21ms 0,20ms + czas cyklu programu
Maksymalny czas opóźnienia przy przejściu ze stanu logicznego '1' na '0' :	
NEED-230AC-.. eliminacja odbić styków ZAŁ eliminacja odbić styków WYŁ	60ms 20ms
NEED-220DC..., NEED-24DC..., NEED-12DC-.. eliminacja odbić styków ZAŁ	21ms

eliminacja odbić styków WYŁ	0,25ms + czas cyklu programu
Znamionowe napięcie izolacji:	300V AC
Separacja galwaniczna: od napięcia zasilania względem siebie względem wyjść	nie nie tak
Maksymalna dopuszczalna długość przewodów (przewód L i sygnałowy prowadzone razem): NEED-230AC-...-08-4: dla wejść cyfrowych I1 – I4 dla wejść cyfrowych I5 – I6 dla wejść cyfrowych I7 – I8 NEED-230AC-...-16-8: dla wejść cyfrowych I1 – I11 dla wejść cyfrowych I12 – I13 dla wejść cyfrowych I14 – I16 NEED-220DC-.. NEED-24DC-..., NEED-12DC-..	 10m 100m 10m 10m 100m 10m 10m 100m

Specyfikacja obwodów wyjściowych

Typ wyjść cyfrowych (PN-EN 61131-2)	przełącznikowe – styki NO, niezabezpieczone (wyjścia cyfrowe AC dostarczające prąd)
Liczba: NEED...-08-4R	4
NEED...-16-8R	8
Wizualizacja stanu logicznego	diody LED
Równoległe łączenie wyjść dla zwiększenia obciążalności	niedopuszczalne
Zewnętrzne zabezpieczenie obwodu wyjść	16A (wyłącznik instalacyjny B16)
Znamionowy prąd obciążenia w kat. AC1	10A AC
Znamionowe napięcie obciążenia w kat. AC1	250V AC
Minimalny prąd zestyków	10mA
Minimalne napięcie zestyków	10V
Rezystancja styków	<100mΩ

Całkowity prąd wyjściowy (PN-EN 61131-2): NEED..-08-4R	40A (4x10A)
NEED..-16-8R	80A (8x10A)
Znamionowe napięcie izolacji wzmocniona podstawowa	300V między wejściami a wyjściami między wyjściami
Napięcie probiercze przerwy stykowej	1 000V AC
Czas zadziałania	7ms
Czas powrotu	3ms
Maksymalna częstotaść łącheń: - przy obciążeniu znamionowym (kategoria AC1) - bez obciążenia	600cykli/h 72 000cykli/h
Trwałość łącheniowa: - w kategorii AC1 - w zależności od stałej czasowej T (L/R=40ms)	>0,7×10 ⁵ (10A 250V AC) >10 ⁵ (0,15A 220V DC)
Trwałość mechaniczna	3×10 ⁷ cykli łącheniowych
Separacja galwaniczna względem: - napięcia zasilania - wejść cyfrowych - złącza PC i karty pamięci	tak tak tak

Specyfikacja wejść analogowych

Typ wejścia	wejścia analogowe
Liczba: NEED..-08-4	2 (I7 – I8)
NEED..-16-8	3 (I14 – I16)
Rodzaj wejść: NEED-230AC-..	napięciowe AC
NEED-220DC-..., NEED-24DC-..., NEED-12DC-..	napięciowe DC
NEED-24DC-...-16-8, NEED-12DC-...-16-8	napięciowe/prądowe DC
Zakres analogowych sygnałów wejściowych: NEED-230AC-.. NEED-220DC-.. NEED-12DC-x1-..., NEED-24DC-x1-.. NEED-12DC-22-..., NEED-24DC-22-.. NEED-24DC-...-16-8 (tryb prądowy) NEED-12DC-...-16-8 (tryb prądowy)	0V - 255V AC 0V - 255V DC 0V - 25,5 DC 0V - 25,5/0 – 12,75 V DC 0 – 51/25,5 mA DC 0 – 51/25,5 mA DC
Czas konwersji	ilość wejść analogowych x 1ms
Rozdzielczość cyfrowa: NEED-230AC-.. NEED-220DC-.. NEED-12DC-x1-08-4	1V 1V 0,1V

NEED-12DC-22-08-4, NEED-12DC-..-16-8 NEED-24DC-x1-08-4 NEED-24DC-22-08-4, NEED-24DC-..-16-8 NEED-12DC-22-08-4, NEED-12DC-..-16-8 (tryb prądowy) NEED-24DC-22-08-4, NEED-24DC-..-16-8 (tryb prądowy)	0,1/0,05V 0,1V 0,1/0,05V 0,2/0,1mA 0,2/0,1mA
Maksymalne dopuszczalne przeciążenie ciągłe: NEED-230AC-.. NEED-220DC-.. NEED-24DC-.. NEED-12DC-..	300V AC 300V DC 40V DC 26V DC
Błąd wejścia analogowego przy 25°C: NEED-230AC-.. NEED-220DC-.. NEED-24DC-.., NEED-12DC-..	±3% zakresu pomiarowego ±2% zakresu pomiarowego ±2% zakresu pomiarowego
Przesłuch między kanałami	36dB
Nieliniowość	±3%
Separacja galwaniczna: - z napięciem zasilania - względem wejść cyfrowych - względem wyjść cyfrowych - względem złącza programującego	nie nie tak nie

Jednostka centralna i pamięć

Pojemność pamięci programu użytkownika NEED-..-08-4 NEED-..-16-8	862 bajty 835 bajty
Rodzaj dostępnej pamięci	EEPROM
Języki programowania (PN-EN 61131-3)	Tekstowy (STL) Graficzny (LAD)

Zasoby programowe NEED...x1-08-4: Znaczniki Timery dokładność Timerów Liczniki (zliczanie w górę/dół) Wartości zliczane Komparatory Zegar tygodniowy Zegar Czasu Rzeczywistego Dokładność Zegara Czasu Rzeczywistego NEED...-22-.. Znaczniki Znaczniki Tekstowe Timery dokładność Timerów Liczniki (zliczanie w górę/dół) Wartości zliczane Szybki Licznik (do 20kHz) Komparatory Zegar tygodniowy Zegar Czasu Rzeczywistego (automatyczna zmiana czasu w różnych strefach czasowych) Dokładność Zegara Czasu Rzeczywistego Zasoby dodatkowe dla NEED-230AC-22-16-8: Znacznik kierunku faz Asymetria sieci trójfazowej NEED...-11-16-8 Znaczniki Timery Dokładność Timerów Liczniki (zliczanie w górę/dół) Szybki Licznik (do 20kHz) Wartości zliczane Komparatory Zegar tygodniowy Zegar Czasu Rzeczywistego (automatyczna zmiana czasu w różnych strefach czasowych) Dokładność Zegara Czasu Rzeczywistego Zasoby dodatkowe dla NEED-230AC-...-16-8: Znacznik kierunku faz Asymetria sieci trójfazowej	M1 – M16 T1 – T8 $\pm 1\%$ wartości ustawionej + (0 - 10)ms C1- C8 0-65535 A1-A8 (H1 – H4)×4 kanały 1 ± 3 s/dzień M1 - M64 MT1 - MT8 T1 – T32 $\pm 1\%$ wartości ustawionej + (0 - 10)ms C1 – C8 0-65535 HC1 A1 – A16 (H1 – H8)×4 kanały 1 ± 3 s/dzień MDIR ASYM M1 – M16 T1 – T16 $\pm 1\%$ wartości ustawionej + (0 - 10)ms C1 – C8 HC1 0-65535 A1 - A12 (H1 – H4)×4 kanały 1 ± 3 s/dzień MDIR ASYM
--	---

Remanencja: Czas podtrzymania Zegara	64h (przy 25°C) 24h (przy 40°C)
Znaczniki	M1 – M16
Liczniki	C5 – C8
Timery	T5 – T8

Zewnętrzna karta pamięci

Opis złącza NEED-M-1K (złącze typu A) pojemność 1KB	Złącze dwurzędowe
NEED-M-1KB (złącze typu B) pojemność 1KB	Złącze jednorzędowe
NEED-M-4KB (złącze typu B) pojemność 4KB	Złącze jednorzędowe
Wymiary (długość × wysokość × głębokość)	30mm x 11mm x 5mm
Masa	2g
Typ pamięci	EEPROM
Pojemność pamięci	1KB
Typ interfejsu łącza	I ² C

Kabel dedykowany

Opis złącza NEED-PC-15A (złącze typu A)	Złącze dwurzędowe
NEED-PC-15B (złącze typu B)	Złącze jednorzędowe
NEED-PC-15C (złącze typu B)	Złącze jednorzędowe
Długość kabla	2m
Masa	100g
Sposób podłączenia: z PC: NEED-PC-15A/15B NEED-PC-15C	9 pin D-Sub USB
z przekaźnikiem	Gniazdo dedykowane
Typ kabla: NEED-PC-15A/15B NEED-PC-15C	RS 232 USB
Szybkość transmisji danych cyfrowych: RS232	19200bit/s
USB	1,5Mbit/s
Kontrola danych	Sumy kontrolne

13. SŁOWNIK

Czas cyklu	– czas przetwarzania wszystkich instrukcji programu
Hasło	– zabezpieczenie przed kopiowaniem programu znajdującego się w sterowniku
Karta pamięci	– zewnętrzna pamięć przekaźnika programowalnego, z której można przepisywać program do wewnętrznej pamięci przekaźnika
Kompilacja	– sprawdzanie poprawności programu i generowanie kodu zrozumiałego dla procesora przekaźnika programowalnego
Konfiguracja	– ustawienie odpowiednich wartości parametrów dla przekaźnika programowalnego
LAD	– język graficzny programowania przekaźnika
Licznik	– element logiczny zasobów wewnętrznych przekaźnika używany w programie przy realizacji zliczających funkcji sterowania
Ładowanie programu	– zapis skompilowanego programu z PC do pamięci przekaźnika programowalnego
Pamięć programu	– obszar pamięci przekaźnika przeznaczony do zapisywania programu sterowania przez użytkownika
Program	– zapis określonego procesu sterowania za pomocą odpowiedniego języka programowania
Przekaźnik programowalny	– przekaźnik posiadający wejścia (styki), wyjścia (cewki) oraz programowalne zasoby logiczne wraz z pamięcią
RUN	– jeden z trybów pracy przekaźnika, w którym następuje normalne przetwarzanie programu
STL	– język tekstowy programowania przekaźnika
STOP	– jeden z trybów pracy przekaźnika, w którym przekaźnik nie wykonuje programu – wyjścia przekaźnika są odłączone
Timer	– element logiczny zasobów wewnętrznych przekaźnika używany w programie przy realizacji czasowych funkcji sterowania

Trigger	– wejście wyzwalające odmierzenie czasu w Timerach
Wejście	– fizyczne wejście przekaźnika, do którego dołącza się zewnętrzne sygnały z czujników, styków itp.
Wejście analogowe	– fizyczne wejście przekaźnika, do którego dołącza się sygnały analogowe
Wyjście	– fizyczne wyjście przekaźnika, do którego podłącza się urządzenia sterowane: lampki, styczniki, zawory elektromagnetyczne itp.
Zasoby remanentne	– elementy logiczne przekaźnika, które pamiętają swój stan po wyłączeniu napięcia zasilającego
Zegar	– element logiczny zasobów wewnętrznych przekaźnika używany w programie przy realizacji funkcji sterowania wykorzystującego czas rzeczywisty
Znacznik	– element logiczny zasobów wewnętrznych przekaźnika używany w programie przy realizacji funkcji sterowania

14. INDEKS

- =, instrukcja, 91
- AND -A, 80
- AND NOT -AN, 82
- AND NOT(-AN(, 83
- AND(-A(, 80
- ASYM, 65
- CD, instrukcja Licznika, 99
- CLR, instrukcja, 115
- CU, instrukcja Licznika, 98
- Cykl programu, 34
- Deinstalacja, 131
- Działanie, Licznik CD, 50
- Działanie, Licznik CU, 50, 98
- Działanie, Timer SD, 47, 93
- Działanie, Timer SE, 48
- Działanie, Timer SF, 47, 94
- Działanie, Timer SL, 48, 96
- Działanie, Zegar, 52
- Edytor, LAD, 147
- Edytor, STL, 144
- FP, instrukcja, 92
- Instalacja, 131
- Instrukcje dla wejść analogowych, 105
- Instrukcje Zegara, 104
- Komparator, porównania, 65
- Konfiguracja, Zegar, 52
- LAD, 116
- LAD, Liczniki, 124
- LAD, obwód, 118
- LAD, program, 118
- LAD, Timery, 123
- LAD, wejścia, 117, 121
- LAD, wyjścia, 117, 122
- LAD, zasady, 126
- LAD, Znaczniki, 122
- Liczniki, 49
- Liczniki, liczba impulsów do zliczenia, 49
- Liczniki, wejścia, 49
- Liczniki, wyjście, 49
- LOAD – L,, 106
- Menu, 138
- Menu, opis, 139
- Mocowanie na szynie DIN, 15
- Mocowanie na śruby, 16
- Narzędzia, 138
- Opóźnienie wejść, 181
- Opóźnienie wyjść, 184
- OR NOT(-ON(, 87
- OR -O, 84
- OR(-O(, 85
- Pamięć programowanie, 186
- Pamięć zewnętrzna, 185
- Pamięć, kopiowanie, 188
- Pamięć, partycje, 186
- pliki, rodzaje, 159
- Podgląd zmiennych, 166
- Podłączenie, wejścia cyfrowe, 18, 21, 22
- Podłączenie, wejścia analogowe, 24, 26, 27
- Podłączenie, wyjścia, 30
- Podłączenie, zasilanie, 31, 32, 33
- Połączenie, z komputerem, 131
- Potencjometr, 70
- Projekt, 132
- Przykłady, zastosowania, 237
- Remanencje, 71
- Reset -R, 91
- S, instrukcja, 91
- SE, instrukcja Timera, 95
- SET, instrukcja, 114
- SF, instrukcja Timera, 94
- SL, instrukcja Timera, 96, 97
- STL, 76
- Symbole, Komparator, 64
- Symbole, LAD, 116, 119
- Symbole, Timer SD - Opóźnione załączenie, 46
- Symbole, Timer SE – Pojedynczy impuls, 47
- Symbole, Timer SF - Opóźnione wyłączenie, 47
- Symbole, Timer SL - Impulsy, 48
- Symbole, wejścia cyfrowe, 39, 74
- Symbole, wyjścia cyfrowe impulsowe, 40
- Symbole, wyjścia cyfrowe kasujące, 40
- Symbole, wyjścia cyfrowe normalne, 39
- Symbole, wyjścia cyfrowe sterujące, 41
- Symbole, wyjścia cyfrowe ustawiające, 41
- Symbole, wyjścia cyfrowe zanegowane sterujące, 41
- Symbole, Znaczniki, 42
- Szybki Licznik, 50
- Timer, czas do odmierzenia, 46
- Timer, wejścia, 45
- Timer, wyjścia, 46
- Timer, zakresy czasowe, 46
- Timery, 45
- Typ, oznaczenie, 9
- Uruchomienie, 175
- Ustawienia, 159
- Ustawienia, Komparatory, 163
- Ustawienia, Liczniki, 162
- Ustawienia, opóźnienia wejść, 164
- Ustawienia, Remanencje, 163
- Ustawienia, Timery, 161
- Ustawienia, Zegary, 162

Wyświetlacz LCD, klawiatura, 189
XOR NOT - XN , 89
XOR NOT(- XN (, 90
XOR - X , 88
XOR(- X (, 88
Zaciski, przewody - rodzaj, 17

Zasilanie, 176
Zasoby przekaźnika, 36
Zegar Czasu Rzeczywistego, 63
Zegar Tygodniowy, 51
Znacznik MDIR, 44
Znaczniki, 42