Materiały do przedmiotu:

STEROWNIKI PLC I SIECI PRZEMYSŁOWE

Laboratorium nr 3

Autor:

dr inż. Łukasz Sobaszek

Fundusze Europejskie dla Rozwoju Społecznego

Rzeczpospolita Polska

Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00

Spis treści

Wprowadzenie, tematyka zajęć **3**
Cel ćwiczenia/laboratorium **5**
Instrukcja laboratoryjna/ćwiczeniowa **5**
Ćwiczenia samodzielne **9**
Pytania pomocnicze **12**
Podsumowanie **12**
Literatura **12**

Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00

Wprowadzenie, tematyka zajęć

Obszary pamięci sterownika PLC – typy danych i ich zastosowanie

Oprócz poznanych do tej pory wejść, wyjść oraz bitów pamięci sterownik PLC podczas realizacji programu może wykorzystywać różne obszary pamięci przeznaczone do przechowywania danych. Stąd też pamięć sterownika nie ogranicza się do pojedynczych bitów, ale umożliwia także wykorzystanie danych innego typu. Odpowiednia organizacja pamięci umożliwia jednoznaczne adresowanie danych oraz ich wykorzystanie w poszczególnych częściach programu.

Dane sterownika mogą być zatem przechowywane i przetwarzane w różnych formatach. Jednym z podstawowych typów jest BOOL, umożliwiający zapis pojedynczego stanu logicznego 0 lub 1. Do przechowywania większej liczby bitów wykorzystywane są m.in. sekwencje bitów – typy BYTE, WORD oraz DWORD. Wartości liczbowe mogą być reprezentowane z kolei przez typy INTEGER oraz REAL. Dobór odpowiedniego typu danych zależy od rodzaju przechowywanej informacji oraz operacji wykonywanych na niej w programie.

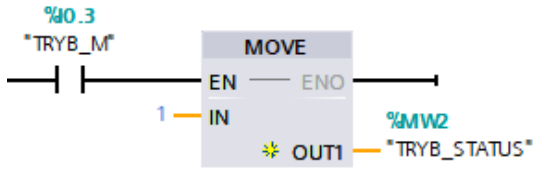
Wspomniane typy danych nie są jedynymi. Oprócz podstawowych typów logicznych i liczbowych w programowaniu sterowników PLC stosowane są również typy danych umożliwiające przechowywanie bardziej złożonych informacji, takich jak: czas, data, znaki czy ciągi znaków. Znajdują one zastosowanie m.in. podczas realizacji funkcji czasowych, rejestracji zdarzeń, zapisu parametrów procesu oraz wymiany danych z innymi urządzeniami.

Znajomość obszarów pamięci, sposobów adresowania oraz podstawowych typów danych jest niezbędna podczas tworzenia bardziej rozbudowanych programów sterujących. Pozwala na prawidłowe przechowywanie i przetwarzanie wartości pochodzących z urządzeń wejściowych, wyników obliczeń oraz parametrów wykorzystywanych do sterowania procesem.

Istotnym zagadnieniem w wykorzystaniu zmiennych różnego typu jest wykorzystanie instrukcji MOVE. Pozwala ona na operowanie na całych wartościach i służy do kopiowania wartości pomiędzy zmiennymi lub obszarami pamięci sterownika. Instrukcja ta kopiuje wartość podaną na wejściu IN do zmiennej wskazanej na wyjściu OUT (rysunek 1). Może być wykorzystywana m.in. do zapisywania wartości w pamięci sterownika, przenoszenia danych pomiędzy zmiennymi oraz przypisywania im określonych wartości.

Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00

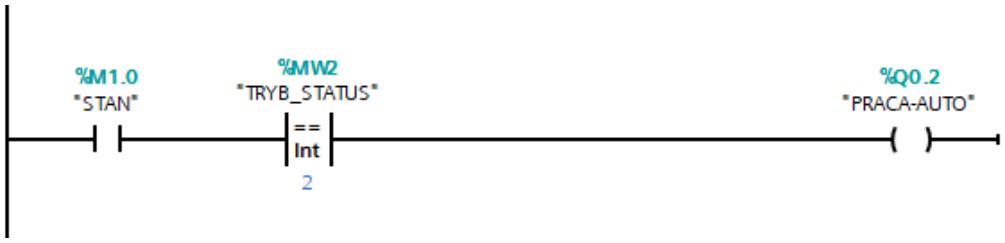


Rysunek 1. Wykorzystanie instrukcji MOVE do przypisania wartości liczbowej do zmiennej.

Rysunek 1 przedstawia fragment programu sterownika PLC w języku drabinkowym LAD, wykorzystujący instrukcję MOVE do przenoszenia danych. Styk normalnie otwarty TRYB_M (I0.3) steruje wykonaniem instrukcji MOVE, która przypisuje wartość liczbową 1 do zmiennej TRYB_STATUS (MW2). Wartość ta jest zapisywana w pamięci sterownika po spełnieniu warunku logicznego, umożliwiając identyfikację wybranego trybu pracy układu.

Kolejną grupą instrukcji wykorzystywanych podczas przetwarzania danych w sterownikach PLC są operatory relacji, umożliwiające porównywanie wartości dwóch zmiennych lub wartości stałych. Wynikiem porównania jest stan logiczny TRUE (1) lub FALSE (0), który może zostać wykorzystany do sterowania dalszym przebiegiem programu (rysunek 2).

Do podstawowych operatorów relacji należą: równość (==), nierówność (<>), większość (>), mniejszość (<), większość lub równość (>=) oraz mniejszość lub równość (<=). Operatory te znajdują zastosowanie m.in. podczas kontroli wartości granicznych, porównywania wyników pomiarów oraz realizacji warunków uruchamiania określonych elementów układu sterowania.



Rysunek 2. Wykorzystanie operatora relacji w celu aktywacji trybu pracy.

Rysunek 2 przedstawia fragment programu sterownika PLC w języku drabinkowym LAD, wykorzystujący instrukcję porównania wartości liczbowych. Styk normalnie otwarty STAN (M1.0) oraz instrukcja porównania sprawdzająca, czy wartość zmiennej TRYB_STATUS (MW2) jest równa 2, sterują wyjściem PRACA-AUTO (Q0.2). Wyjście zostaje aktywowane wyłącznie po spełnieniu obu warunków logicznych, umożliwiając uruchomienie układu w trybie automatycznym.

Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00

W środowisku TIA Portal operatory relacji dostępne są w postaci instrukcji porównujących, które można umieszczać na kolejnych szczeblach schematu drabinkowego LAD. Porównywane wartości przypisywane są do wejść znajdujących się u góry oraz u dołu wybranego bloku instrukcji, natomiast wynik porównania decyduje o stanie logicznym dalszej części szczebla.

Cel ćwiczenia/laboratorium

Celem ćwiczenia jest zapoznanie z organizacją pamięci sterownika PLC, podstawowymi typami danych oraz sposobami ich adresowania i wykorzystania w programach sterujących. W ramach zajęć student pozna zasady przechowywania i przetwarzania danych różnego typu oraz sposób wykorzystania instrukcji MOVE do przenoszenia wartości pomiędzy zmiennymi i obszarami pamięci sterownika.

Realizacja ćwiczenia pozwoli na zdobycie umiejętności samodzielnego definiowania i wykorzystywania zmiennych różnych typów, wykonywania podstawowych operacji na przechowywanych wartościach oraz sprawdzania poprawności działania przygotowanych programów. Zdobyta wiedza i umiejętności stanowić będą podstawę do tworzenia bardziej złożonych algorytmów sterowania wykorzystujących przetwarzanie danych i parametrów procesu.

Samodzielna realizacja oraz weryfikacja przygotowanych programów mają na celu kształtowanie dokładności, odpowiedzialności oraz profesjonalnego podejścia do rozwiązywania zadań inżynierskich.

Instrukcja laboratoryjna/ćwiczeniowa

Podstawowe typy danych wykorzystywane w sterownikach PLC można podzielić na trzy grupy:

- Typy logiczne i bitowe – obejmują typ BOOL, przechowujący pojedynczy stan logiczny (0 lub 1), oraz typy BYTE (8 bitów), WORD (16 bitów) i DWORD (32 bity), umożliwiające zapis sekwencji bitów. Przykładem zastosowania jest przechowywanie stanów przycisków, czujników oraz sygnałów sterujących.
- Typy całkowitoliczbowe – obejmują typy SINT i USINT (8 bitów), INT i UINT (16 bitów) oraz DINT i UDINT (32 bity), umożliwiające przechowywanie liczb całkowitych ze znakiem lub bez znaku. Przykładem zastosowania jest zliczanie wyprodukowanych

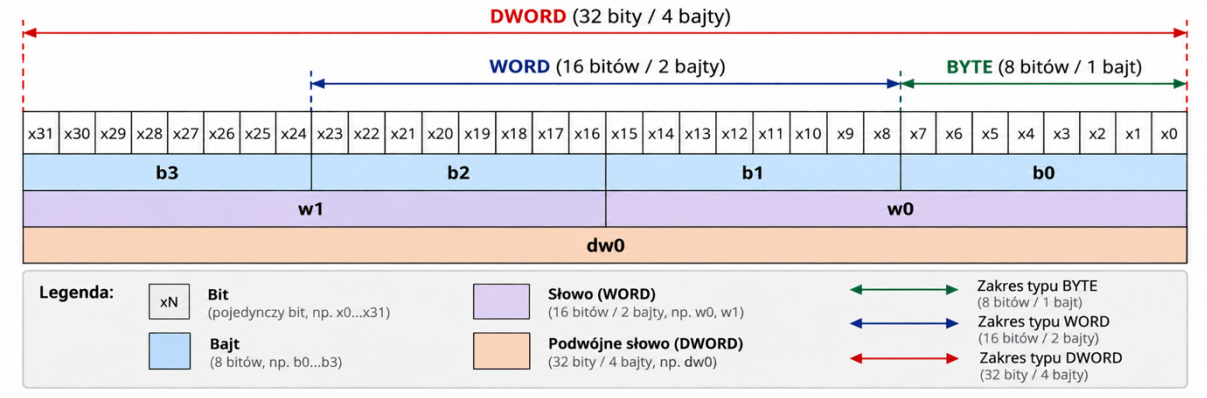
Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00

elementów, rejestrowanie liczby cykli pracy maszyny oraz przechowywanie wartości zadanych.

- Typy zmiennoprzecinkowe – obejmują typy REAL (32 bity) i LREAL (64 bity), umożliwiające przechowywanie liczb rzeczywistych, również zawierających część ułamkową. Przykładem zastosowania jest zapis wartości pomiarowych, takich jak temperatura (23,5°C), ciśnienie (2,75 bar) czy prędkość obrotowa silnika.

Dobór odpowiedniego typu danych powinien uwzględniać zakres przechowywanych wartości, wymaganą dokładność oraz ilość zajmowanej pamięci sterownika PLC (rysunek 3).



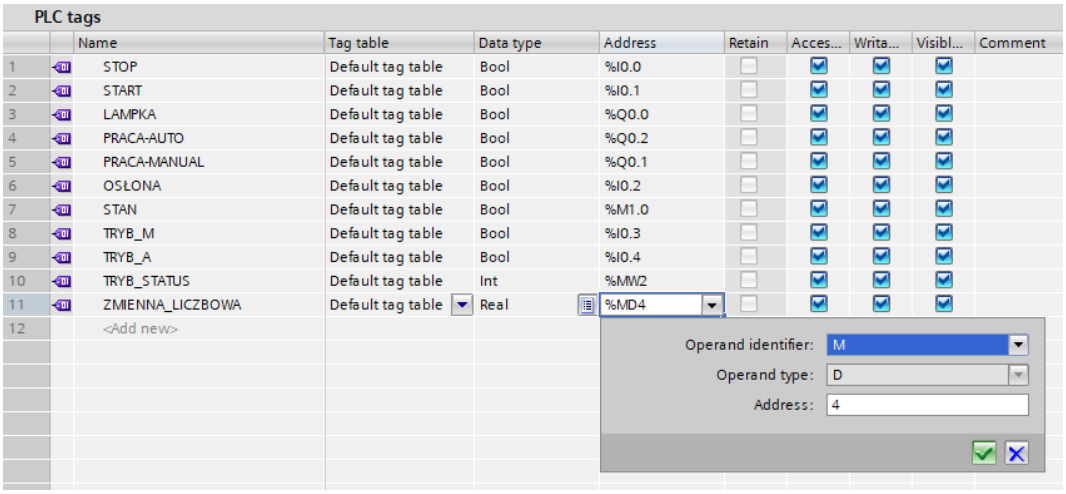
Rysunek 3. Struktura typów danych oraz sposób organizacji bitów w pamięci sterownika PLC.

Rysunek 3 przedstawia strukturę podstawowych typów danych wykorzystywanych w pamięci sterownika PLC: BYTE, WORD oraz DWORD. Typ BYTE zajmuje 8 bitów (1 bajt), WORD składa się z 16 bitów (2 bajtów), natomiast DWORD obejmuje 32 bity (4 bajty). Schemat przedstawia sposób grupowania pojedynczych bitów w bajty (b0–b3), słowa (w0–w1) oraz podwójne słowo (dw0). Poszczególne bity oznaczono symbolami od x0 do x31, co pozwala zobrazować zależności między typami danych oraz ich organizację w pamięci sterownika.

Zmienne deklaruje się w taki sam sposób jak zmienne wejściowe (I), wyjściowe (Q) oraz znaczniki wewnętrzne (M), nadając im odpowiednią nazwę i adres pamięci. Różnica polega na wyborze odpowiedniego typu danych, np. BYTE, WORD, INT lub REAL, zamiast stosowanego wcześniej typu BOOL (rysunek 4).

Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00



Rysunek 4. Deklarowanie zmiennej w tablicy.

Ważne!

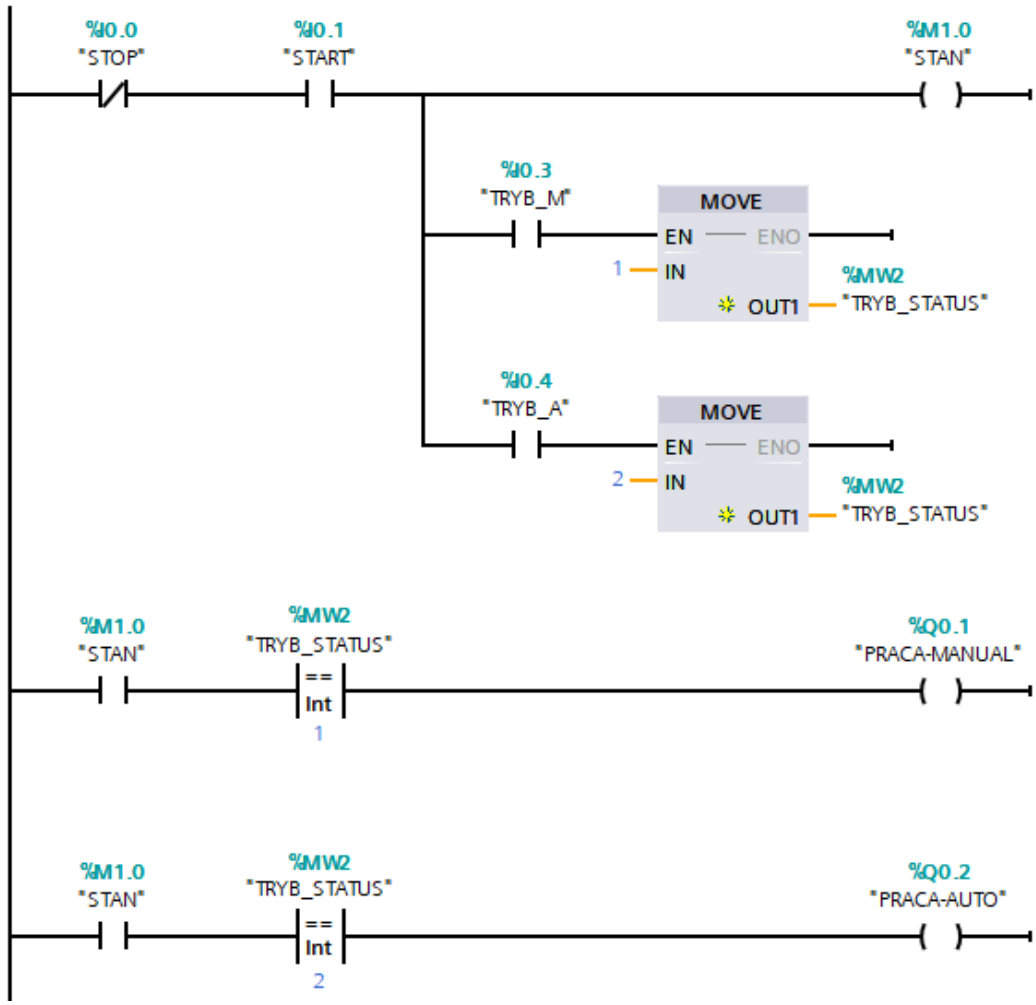
Podczas definiowania zmiennych należy zwrócić szczególną uwagę na sposób adresowania pamięci sterownika PLC oraz liczbę bajtów zajmowanych przez poszczególne typy danych. Nieprawidłowe przypisanie adresów może prowadzić do nakładania się obszarów pamięci różnych zmiennych, a w konsekwencji do niezamierzonej zmiany ich wartości i błędnego działania programu. Przykładowo zmienna typu WORD o adresie %MW0 zajmuje bajty MB0 i MB1, dlatego przypisanie innej zmiennej do adresu %MB1 spowoduje nakładanie się ich obszarów pamięci.

Zadeklarowane wcześniej zmienne można wykorzystać w instrukcji MOVE, która umożliwia przenoszenie wartości pomiędzy określonymi obszarami pamięci sterownika PLC. Aby dodać instrukcję MOVE do programu w języku LAD, należy w panelu *Instructions* rozwinąć kategorię *Basic instructions* i odnaleźć *Move operations*, a następnie wybrać instrukcję MOVE i przeciągnąć ją do odpowiedniej gałęzi programu. Po umieszczeniu instrukcji należy przypisać zmienną lub wartość źródłową do wejścia IN oraz zmienną docelową do wyjścia OUT1.

Podczas pracy ze zmiennymi liczbowymi przydatne są również operatory relacyjne, umożliwiające porównywanie wartości dwóch zmiennych lub zmiennej z określoną wartością liczbową. Aby dodać operator relacyjny do programu w języku LAD, należy również skorzystać z biblioteki instrukcji, a następnie wybrać odpowiednią instrukcję porównania, np. równości (==), nierówności (<>), większości (>) lub mniejszości (<), i przeciągnąć ją do wybranej gałęzi programu. Po umieszczeniu instrukcji należy określić typ porównywanych danych oraz przypisać wartości lub zmienne (rysunek 5).

Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00



Rysunek 5. Przykład programu w języku LAD wykorzystującego polecenie blok MOVE oraz operatory relacji

Rysunek 5 przedstawia program sterownika PLC w języku drabinkowym LAD, umożliwiający wybór trybu pracy układu za pomocą instrukcji MOVE. Naciśnięcie przycisku START (I0.1), przy nieaktywnym przycisku STOP (I0.0), powoduje ustawienie bitu pamięci STAN (M1.0), który sygnalizuje aktywację układu. Wybór trybu ręcznego (TRYB_M, I0.3) powoduje zapisanie wartości 1 do zmiennej TRYB_STATUS (MW2), natomiast wybór trybu automatycznego (TRYB_A, I0.4) przypisuje tej zmiennej wartość 2. W kolejnych segmentach programu instrukcje porównania sprawdzają wartość zmiennej TRYB_STATUS i, w zależności od wybranego trybu oraz stanu bitu STAN, aktywują odpowiednie wyjście sterownika: PRACA-MANUAL (Q0.1) lub PRACA-AUTO (Q0.2).

Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00

Ćwiczenia samodzielne

Ćwiczenie nr 1

Wykorzystując poznane zagadnienia opracuj program w języku LAD umożliwiający sterowanie pracą przenośnika taśmowego (rysunek 7) w dwóch trybach: ręcznym i automatycznym. Do sterowania układem wykorzystaj zmienne przechowywane w pamięci sterownika PLC, instrukcję MOVE oraz operatory relacyjne.

Układ powinien działać w następujący sposób:

Aktywacja układu i wybór trybu pracy:

- Naciśnięcie przycisku S_M powoduje aktywację układu i umożliwia wybór trybu pracy.
- Naciśnięcie przycisku S_B1 powoduje wybór trybu ręcznego (TRYB_PRACY = 1).
- Naciśnięcie przycisku S_B2 powoduje wybór trybu automatycznego (TRYB_PRACY = 2).
- Naciśnięcie przycisku S_B3 powoduje zatrzymanie taśmociągu i dezaktywację układu (TRYB_PRACY = 0).
- Wybór trybu pracy jest możliwy wyłącznie po aktywacji układu przyciskiem S_M.

Tryb ręczny:

- Po wybraniu trybu ręcznego taśmociąg pozostaje zatrzymany i zapalają się lampki sygnalizacyjne przycisków sterowania S_B1 i S_B2.
- Naciśnięcie przycisku S_B1 powoduje uruchomienie taśmociągu w lewo.
- Naciśnięcie przycisku S_B2 powoduje uruchomienie taśmociągu w prawo.
- Zwolnienie przycisku sterującego powoduje zatrzymanie taśmociągu.
- Aktywacja czujnika optycznego LB powoduje zatrzymanie taśmociągu.

Tryb automatyczny:

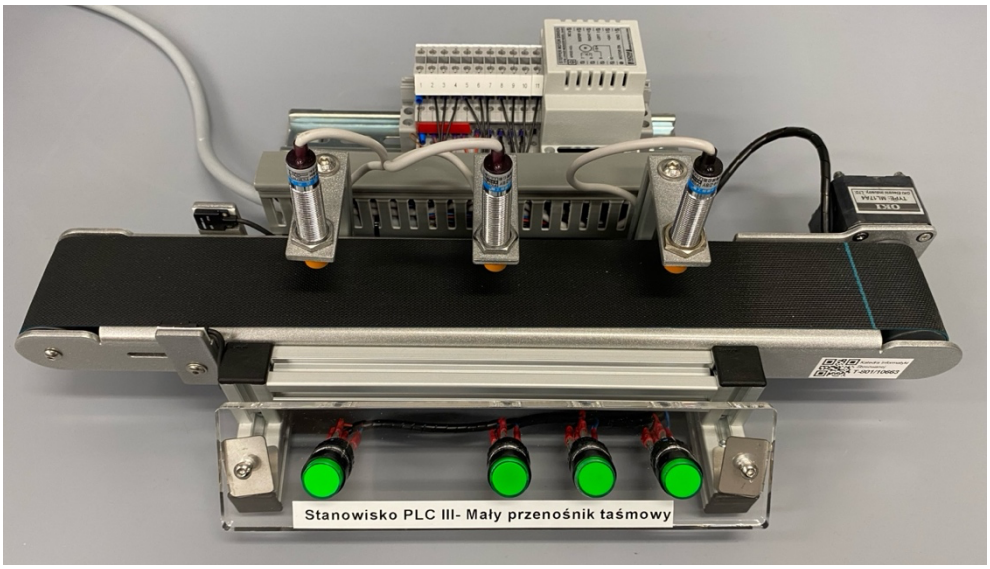
- Po wybraniu trybu automatycznego taśmociąg automatycznie rozpoczyna ruch w lewo i zapala się lampka sygnalizacyjna przycisku S_B2, migając z częstotliwością 1Hz.
- Aktywacja czujnika optycznego LB powoduje zatrzymanie taśmociągu.

Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00

- Ponowne uruchomienie taśmociągu następuje po ponownym wybraniu trybu automatycznego.
- Naciśnięcie przycisku S_B3 powoduje natychmiastowe zatrzymanie taśmociągu i dezaktywację układu.
- Podczas naciśnięcia przycisku S_B3 zostaje on podświetlony na czas jego wyboru.

Ważne: Program powinien uniemożliwiać jednoczesne uruchomienie taśmociągu w obu kierunkach. Zmiana trybu pracy powinna powodować zatrzymanie aktualnie wykonywanego ruchu.



Rysunek 7. Wykorzystywane stanowisko laboratoryjne – model przenośnika taśmowego

Tabela 2. Lista sygnałów do realizacji ćwiczenia.

Lista sygnałów		
Nazwa	**Adres**	**Komentarz**
LB	I0.0	Bariera świetlna (czujnik optyczny)
S_M	I0.1	Przycisk nr 1 po lewej stronie
S_B1	I0.2	Przycisk nr 1 w prawym segmencie
S_B2	I0.3	Przycisk nr 2 w prawym segmencie
S_B3	I0.4	Przycisk nr 3 w prawym segmencie
B1	I0.5	Czujnik 1

Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00

B2	I0.6	Czujnik 2
B3	I0.7	Czujnik 3
S_M	Q0.1	Lampka przycisku S_M
S_B1	Q0.2	Lampka przycisku S_B1
S_B2	Q0.3	Lampka przycisku S_B2
S_B3	Q0.4	Lampka przycisku S_B3
T_P	Q0.5	Taśmociąg – ruch w prawo
T_L	Q0.6	Taśmociąg – ruch w lewo
BUZZER	Q0.7	Sygnalizator dźwiękowy

Ćwiczenie nr 2

Wykorzystując poznane zagadnienia rozbuduj opracowany program sterowania przenośnikiem taśmowym o funkcję automatycznej identyfikacji i transportu elementów metalowych z wykorzystaniem czujnika optycznego LB oraz dwóch czujników indukcyjnych B1 i B3. Do realizacji zadania wykorzystaj instrukcję MOVE, zmienne pamięci wewnętrznej sterownika PLC oraz operatory relacyjne.

Układ powinien działać w następujący sposób:

- W trybie ręcznym czujniki indukcyjne nie wpływają na kierunek ruchu taśmociągu.
- W trybie automatycznym taśmociąg transportuje elementy w lewo.
- Wykrycie elementu metalowego przez czujnik B3 powoduje zapisanie wartości 1 do zmiennej STATUS_ELEMENTU typu INT.
- Po przemieszczeniu elementu do czujnika B1 taśmociąg zostaje zatrzymany, a zmienna STATUS_ELEMENTU przyjmuje wartość 2.
- Jeżeli czujnik optyczny LB wykryje element, który nie został rozpoznany jako metalowy przez czujnik B3, taśmociąg kontynuuje pracę, a zmienna STATUS_ELEMENTU przyjmuje wartość 3.
- Naciśnięcie przycisku S_B3 powoduje zatrzymanie taśmociągu i wyzerowanie zmiennej STATUS_ELEMENTU.
- Ponowne uruchomienie taśmociągu po zatrzymaniu przy czujniku B1 powinno być możliwe dopiero po ponownym wybraniu trybu automatycznego.

Fundusze Europejskie dla Rozwoju Społecznego
Rzeczpospolita Polska
Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00

Pytania pomocnicze

1. Jakie podstawowe obszary pamięci występują w sterownikach PLC i do czego są wykorzystywane?
2. Czym różnią się typy danych BOOL, BYTE, WORD oraz DWORD?
3. Jakie są różnice pomiędzy typami danych INT i REAL?
4. W jaki sposób adresowane są zmienne przechowywane w pamięci sterownika PLC?
5. Do czego służy instrukcja MOVE i na czym polega jej działanie?
6. Jakie operatory relacji są dostępne w języku LAD i do czego służą?
7. W jaki sposób wynik działania operatora relacji może wpływać na realizację programu sterującego?
8. W jakich praktycznych zastosowaniach można wykorzystać instrukcję MOVE oraz operatory relacji?

Podsumowanie

Podczas zajęć laboratoryjnych poznano podstawowe obszary pamięci sterownika PLC, sposoby ich adresowania oraz zastosowanie różnych typów danych. Przeanalizowano działanie instrukcji MOVE oraz operatorów relacji, wskazując ich znaczenie w przetwarzaniu danych i realizacji warunków logicznych. Doskonalono umiejętności tworzenia programów w języku LAD, wykorzystujących zmienne różnych typów do realizacji określonych funkcji sterujących. Zwrócono uwagę na znaczenie prawidłowego doboru typów danych oraz możliwości praktycznego zastosowania poznanych instrukcji w układach automatyki przemysłowej.

Literatura

1. Kasprzyk J., *Programowanie sterowników przemysłowych*, PWN, Warszawa 2017.
2. Kwiecień R., *Komputerowe systemy automatyki przemysłowej*, Helion, Gliwice 2013.
3. Siemens, *Przewodnik programowania dla S7-1200/S7-1500*, 2019.
4. Siemens, *Podręcznik pierwsze kroki z SIMATIC S7-1200 Easybook*, 2014.

Fundusze Europejskie dla Rozwoju Społecznego | Rzeczpospolita Polska | Dofinansowane przez Unię Europejską

Projekt współfinansowany ze środków Unii Europejskiej w ramach: FUNDUSZE EUROPEJSKIE DLA ROZWOJU SPOŁECZNEGO 2021-2027 (FERS) "POLLUB z nami nowoczesne technologie" FERS.01.05-IP.08-0319/23-00