

Fotografia 1. Drut ze szpuli przechodzi przez ramię z szeregiem rolek służących do kontroli napięcia drutu



Projekt maszyny do prostowania i cięcia drutu

Maszyna do prostowania i cięcia drutu jest powszechnie używana przy produkcji ogrodzeń panelowych. Drut zakupiony w szpulach jest rozwijany, prostowany i cięty na jednakowej długości fragmenty. Następnie odcinki te w dalszym procesie technologicznym są zgrzewane w gotowe panele. Od dokładności i powtarzalności produkcji zależy jakość gotowego wyrobu, dodatkowo maszyna musi pracować z odpowiednią wydajnością, aby zapewnić materiał do dalszej obróbki. Przy budowie maszyny konstruktor zdecydował się na zastosowanie prostych technologii, przykładowo nie zastosował cięcia w locie, co nie wpłynęło negatywnie na funkcjonalność maszyny, natomiast znacząco obniżyło koszty wykonania.

Maszyna składa się z szeregu modułów połączonych w jedną funkcjonalną całość. Pierwszym modułem, na który jest przez operatora ładowana szpula z drutem jest moduł odwijaka. Moduł ten służy do odwijania i prowadzenia drutu do następnych modułów maszyny, aby zapewnić jego sprawne odwijanie bez skręcania oraz pęknięcia. Moduł ten zbudowany jest z motoreduktora, który napędza podstawę z rolekami, na której umieszczana jest szpula. Drut z szpuli przechodzi przez ramię z szeregiem rolek służące do kontroli napięcia drutu (fotografia 1).

Pozycja tego ramienia jest regulowana przez siłownik pneumatyczny, zasilany powietrzem o ciśnieniu odpowiednio wybranym dla danej grubości drutu. Na siłowniku umieszczone są dwa czujniki położenia tłoka. Sygnały cyfrowe z tych czujników służą do zwiększania prędkości napędu w momencie, gdy ramię zostaje zbyt mocno naprężone, oraz zmniejszania prędkości w sytuacji, jeśli ramię jest rozprężone. Mechanizm ten stanowi prosty regulator prędkości odwijania drutu, w zależności od jego napięcia i jest niezbędny dla uniknięcia problemów z zrywaniem, czy też splątaniem odwijanego drutu.



Fotografia 2. Moduł zaciągania drutu składa się z silnika napędzającego poprzez pasek klinowy rolki zaciągające drut



Fotografia 3. Moduł cięcia składa się z prowadnicy drutu, na której umieszczona jest blokada mechaniczna i czujnik wykrycia dojazdu drutu do blokady

Kolejnym modułem jest moduł zaciągania drutu. Moduł ten składa się z silnika napędzającego poprzez pasek klinowy rolki zaciągające drut. Oraz siłowników pneumatycznych sterujących rolkami dociskowymi. Rolki te są luzowane w czasie pierwszego wprowadzania drutu do modułu oraz w czasie pracy automatycznej w zależności od wybranego algorytmu pracy maszyny, co zostanie dokładniej opisane przy module tnącym maszyny (fotografia 2).

Następny moduł stanowi wirujący walec służący do prostowania drutu. Walec wewnątrz jest specjalnie ukształtowany tak, aby wirując z wysoką prędkością powodował prostowanie przechodzącego przez niego drutu. Po przejściu przez moduł prostujący, drut za pomocą modułu zaciągu jest transportowany do modułu cięcia. Moduł cięcia składa się z prowadnicy drutu, na której w zależności od wymaganej długości drutu umieszczona jest blokada mechaniczna wraz z czujnikiem wykrycia dojazdu drutu do blokady (fotografia 3).

Czujnik taki w momencie dojazdu transportowanego drutu, informuje sterownik o konieczności uruchomienia gilotyny. Po odcięciu

drut łąduje w zasobniku, skąd może być pobrany przez operatora do dalszej obróbki. W zależności od grubości drutu można zmieniać algorytm pracy modułu zaciągu. Jeśli mamy drut o małej średnicy to w momencie dojazdu transportowanego drutu do blokady to należy unieść rolki dociskowe tak aby nie powodować zagięć. W sytuacji gdy obrabiamy drut o dużej średnicy, nie ma potrzeby unoszenia rolek dociskowych, dzięki czemu zaoszczędzamy kilkaset ms, oraz mniej zużywamy komponenty maszyny.

W momencie detekcji czujnika blokady z opóźnieniem uruchamiana jest gilotyna, w postaci noża napędzanego motoreduktorem z mimośrodem powodująca odcięcie odcinka

drutu. Natychmiast po odcięciu jest otwierana listwa zamykająca dolną część prowadnicy, co powoduje wyrzut zadanego odcinka do zasobnika (magazyńku) – fotografia 4.

Elementy wejściowe sterownika PLC

Sterowanie maszyny realizowane jest poprzez sterownik programowalny PLC. W tym przypadku został użyty sterownik firmy Mitsubishi Electric serii FX3G. Jest to sterownik serii kompaktowej, nadający się idealnie do średnich aplikacji. Wraz z sterownikiem został użyty moduł kart komunikacyjnej RS485 wykorzystywanej do komunikacji z falownikami.

Wejścia sterownika: X0 – IN_cz_DlugoscDrutu, IN_cz_CiecieDrutu, IN_p_Start, IN_p_Stop, IN_p_Ciecie, IN_p_DociskLewy, IN_p_DociskPrawy, IN_cz_PredkoscSzybkaOdwijak, IN_cz_PredkoscWolnaOdwijak, IN_OslonaZamknieta, IN_p_PosowTyl, IN_p_PosowPrzod, IN_NapedyAwaria, IN_Bezpieczenstwo.

Wyjścia sterownika: OUT_StartOdwijakTyl, OUT_ZaworDociskLewy, OUT_ZaworDociskPrawy, OUT_ZaworZrzut, OUT_Ciecie, OUT_StartModulProstujacy, OUT_StartZaciagDrutuTyl, OUT_StartOdwijakPrzod, OUT_StartZaciagDrutuPrzod.

Sterowanie elektryczne

Za sterowanie maszyną odpowiada panel HMI oraz szereg przycisków służących do wygodnej obsługi ręcznej urządzenia przykładowo w czasie załadunku drutu. Użyty panel operatorski do 5 calowy model firmy Mitsubishi GT1055, komunikuje on się z sterownikiem PLC za pośrednictwem interfejsu RS422. Panel ten służy do ustawiania pracy automatycznej, sterowania ręcznego, komunikowania awarii jak również ustawienia niektórych parametrów napędów. Napędy stanowią falowniki serii FRD700 firmy Mitsubishi Electric. Falowniki komunikują się z sterownikiem PLC za pośrednictwem interfejsu RS485 protokołem INVERTER PROTOCOL (fotografia 5).

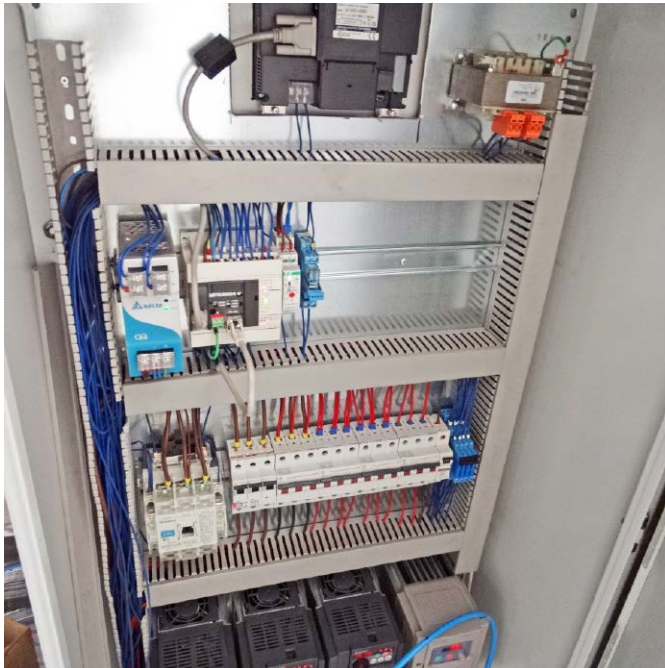
Pomimo użycia protokołu komunikacyjnego źródłem startu dla napędów tradycyjnie pozostaje wyjście cyfrowe sterownika PLC. Na życzenie klienta żeby zwiększyć niezawodność maszyny przy częstych cyklach pracy zamiast przekaźników elektromechanicznych zastosowana przekaźniki półprzewodnikowe. Za bezpieczeństwo odpowiada wyłącznik bezpieczeństwa który powoduje poza odpowiednią reakcją programu odcięcie wyjść sterownika PLC oraz zatrzymanie napędów poprzez wyłączenie stycznika głównego. Dodatkowo maszyna jest wyposażona w wyłącznik krańcowy informujący o zamknięciu osłony głównej. Sygnał z tego czujnika działa podobnie jak zatrzymanie awaryjne, z tą różnicą że w trybie ręcznym operator może pracować z otwartą osłoną, przytrzymując ręcznie odpowiedni przycisk, tak aby było możliwe załadowanie drutu do obróbki. Jako ciekawostkę można podać, że aby zaoszczędzić miejsce na przyciski



Fotografia 4. Moduł cięcia i magazynku



Fotografia 5. Sterownik PLC używany w maszynie



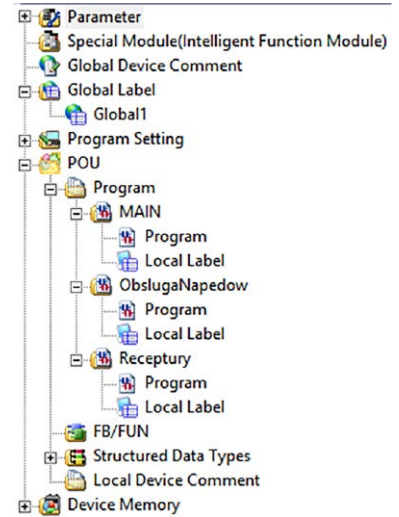
Fotografia 6. Szafa elektryczna urządzenia sterującego

sterujące do pracy ręcznej zdecydowałem się użyć przełącznika lewo prawo domyślnie do sterowania kierunkiem zaciągu drutu, natomiast po przytrzymaniu przycisku stop funkcja przełącznika zmienia się na obroty odwijak w lewo/ prawo (fotografia 6).

Obsługa maszyny

Obsługę maszyny tak pomyślano, aby zminimalizować liczbę czynności wykonywanych na panelu operatorskim. Po zaprogramowaniu receptur, obsługa panelu HMI sprowadza się do wyboru odpowiedniego programu adekwatnego do rodzaju obrabianego drutu. Parametry programu, które są możliwe do ustawienia przez operatora to:

- *Szybkość modułu prostującego* – określa szybkość z jaką będzie obracał się wirnik stanowiący moduł prostujący. Prędkość tą dobiera się w zależności od rodzaju drutu, oraz szybkości przesuwu.
- *Szybkość posuwu drutu* – określa szybkość z jaką będzie pobierany drut poprzez moduł zaciągania drutu.
- *Liczba sztuk* – określa liczbę sztuk do wykonania w 1 cyklu wykonania wybranego programu.
- *Odwijak MIN, MAX* – określa zakres szybkości minimalnej i maksymalnej dla modułu odwijaka. Parametry te ustawia się adekwatnie do szybkości zaciągania drutu, tak aby spowodować płynną pracę maszyny.



Rysunek 9. Struktura (drzewo) projektu

Menu użytkownika pokazano na **rysunku 7**. Dodatkowo, gdyby awarii uległ któryś z przycisków służących do sterowania ręcznego, operator może skorzystać z panelu sterowania ręcznego wyświetlanego na ekranie HMI (**rysunek 8**).

Dla operatora wyższego rzędu przygotowano zostało szereg ekranów gdzie można zmieniać odpowiednie parametry napędów, takie jak rampy bądź czasy sekwencji czasowych.

Implementacja programowa

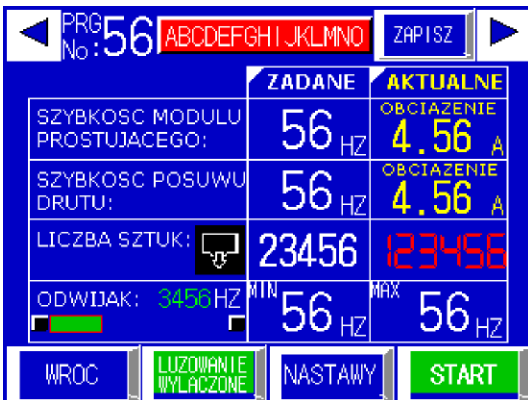
Oprogramowanie sterujące maszyną składa się z dwóch projektów, z czego jeden przypada na sterownik PLC natomiast drugi na panel HMI. Program sterownika został napisany w języku Structured Ladder, i składa się z kilku podprogramów, pokazanych na **rysunku 9**:

- MAIN – główny program sterujący pracą maszyny.
- ObsługaNapędow – podprogram do obsługi napędów serii FRD700.
- Receptury – wybieranie i edycja receptur.

W sterownikach Mitsubishi z serii FX3G przewidziano 4 instrukcje służące do komunikacji za pośrednictwem protokołu INVERTER PROTOCOL. Zaletą tych instrukcji jest łatwa implementacja programowa oraz fakt, iż mogą być one wywoływane jednocześnie dla tego samego portu:

- IVCK: Inverter Status CCheck – sprawdź status falownika (**rysunek 10, rysunek 11**). Za pomocą tej instrukcji możemy z poziomu programu monitorować status falownika.
- IVDR: Inverter Drive – steruj napędem (**rysunek 12, rysunek 13**). Za pomocą tej instrukcji możemy sterować przetwornicą z poziomu programu za pomocą RS485.
- IVRD: Inverter Parametr Read – odczytaj wybraną wartość parametru napędu (**rysunek 14**). Za pomocą tej instrukcji odczytujemy wartość wybranego parametru.
- IVWR: Inverter Parametr Write – zapisz wybraną wartość parametru napędu (**rysunek 15**). Za pomocą tej instrukcji wpisujemy wartość wybranego parametru.

Chcąc ustawić parametry falownika z poziomu panelu operatorskiego możemy zastosować poniższy schemat postępowania. W pierwszej kolejności tworzymy ekran z zmiennymi, które chcemy edytować. Tu będą to tylko: rampa przyśpieszania i rampa hamowania (**rysunek 16**). W czasie tworzenia ekranu HMI warto skorzystać z funkcji ograniczenia minimalnej i maksymalnej wartości, aby zabezpieczyć algorytm przed wpisaniem nieprawidłowej wartości (**rysunek 17**).



Rysunek 7. Menu użytkownika



Rysunek 8. Menu sterowania ręcznego

Następnie, w sterowniku PLC deklarujemy zmienne z adresami zgodnymi z tymi ustawionymi na panelu HMI (**rysunek 18**). W następnej kolejności piszemy stosowny kod, zgodnie z instrukcją umieszczoną wyżej (**rysunek 19**), gdzie:

- EN – uruchom funkcję,
- S1 – numer falownika w sieci,
- S2 – numer parametru, 7- czas przyspieszania, 8- czas hamowania,
- S3 – wartość parametru, czyli wartość zmiennejwyswietlona na panelu HMI,
- N – numer kanału komunikacyjnego ustawiany w parametrach PLC. Dla tego przypadku ustawiłem parametry, jak na **rysunku 20**.

Variable	Description	Data type
EN	Execution condition	Bit
Input variable	(S1) Inverter station number	ANY16
	(S2) Inverter instruction code	ANY16
	(S3) Channel to be used (K1:ch1,K2:ch2 ¹)	ANY16
Output variable	ENO	Bit
	(D) Device storing the read value	ANY16

Rysunek 10. Argumenty i zmienne polecenia IVCK

Instruction code specified in (S2)	Read contents	Corresponding inverter									
		F700	A700	E700	D700	V500	F500	A500	E500	S500	
H79	Operation mode	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H6F	Output frequency (number of rotations)	✓	✓	✓	✓	✓ ¹	✓	✓	✓	✓	
H70	Output current	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H71	Output voltage	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H72	Special monitor	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H73	Special monitor selection number	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H74	Abnormal contents	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H75	Abnormal contents	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H76	Abnormal contents	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H77	Abnormal contents	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H79	Inverter status monitor (extension)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H7A	Inverter status monitor	✓	✓	✓	✓	✓	✓	✓	✓	✓	
H8E	Set frequency (read from E2PROM)	✓	✓	✓	✓	✓ ¹	✓	✓	✓	✓	
H8D	Set frequency (read from RAM)	✓	✓	✓	✓	✓ ¹	✓	✓	✓	✓	
H7F	Link parameter extended setting	These codes cannot be specified in (S2) of the IVCK instruction. They are automatically processed when a "second parameter specification code" is specified the IVRD instruction.									
H8C	Second parameter changing	These codes cannot be specified in (S2) of the IVCK instruction. They are automatically processed when a "second parameter specification code" is specified the IVRD instruction.									

Rysunek 11. Specyfikacja polecenia IVCK

Instruction name	Operation	Execution form	Expression in each language	
			Structured ladder/FBD	ST
IVDR	16 bits	Continuous		IVDR(EN,s1,a2,a3,n);

Rysunek 12. Polecenie IVDR

Hexadecimal instruction code of inverter specified in (S2)	Written contents	Corresponding inverter								
		F700	A700	E700	D700	V500	F500	A500	E500	S500
HFB	Operation mode	✓	✓	✓	✓	✓	✓	✓	✓	✓
HF3	Special monitor selection number	✓	✓	✓	✓	✓	✓	✓	✓	✓
HF9	Operation command (extension)	✓	✓	✓	✓	✓	✓	✓	✓	✓
HFA	Operation command	✓	✓	✓	✓	✓	✓	✓	✓	✓
HEE	Set frequency (written to EEPROM)	✓	✓	✓	✓	✓ ³	✓	✓	✓	✓
HED	Set frequency (written to RAM)	✓	✓	✓	✓	✓ ³	✓	✓	✓	✓
HFD ¹	Inverter reset ²	✓	✓	✓	✓	✓	✓	✓	✓	✓
HF4	Abnormal contents all clear	✓	✓	✓	✓	✓	✓	✓	✓	✓
HFC	Parameter all clear	✓	✓	✓	✓	✓	✓	✓	✓	✓
HFC	User clear	✓	✓	✓	✓	✓	✓	✓	✓	✓
HFF	Link parameter extended setting	✓	✓	✓	✓	✓	✓	✓	✓	✓

Rysunek 13. Specyfikacja polecenia IVDR

Variable	Description	Data type
EN	Execution condition	Bit
Input variable	(S1) Inverter station number	ANY16
	(S2) Inverter parameter number	ANY16
	(S3) Set value to be written to the inverter parameter or device storing the data to be set	ANY16
Output variable	ENO	Bit
	(D) Device storing the read value	ANY16

Rysunek 14. Argumenty i zmienne polecenia IVRD

Variable	Description	Data type
EN	Execution condition	Bit
Input variable	(S1) Inverter station number	ANY16
	(S2) Inverter parameter number	ANY16
	(S3) Set value to be written to the inverter parameter or device storing the data to be set	ANY16
Output variable	ENO	Bit
	(D) Device storing the read value	ANY16

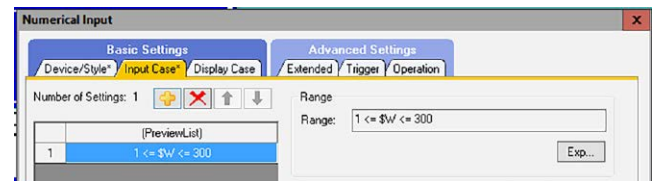
Rysunek 15. Argumenty i zmienne polecenia IWWR

Należy zwrócić uwagę na flagę ZapiszParametr która po stronie HMI stanowi przycisk Zapisz. Równie ważna, co flaga ZapiszParametr jest flaga CzytajParametr. Flaga ta ustawia się automatycznie w momencie przechodzenia po poszczególnych ekranach serwisowych. W momencie wykrycia takiego przejścia jednorazowo zostaje wykonana instrukcja odczytania parametrów według kodu z **rysunku 21**.

Innym fragmentem programu, który chciałbym omówić jest kod obsługujący mechanizm ucięcia i zrzutu drutu. Obie sekwencje inicjowane są za pomocą pobudzenia czujnika blokady drutu, natomiast bardzo istotne jest zapewnienie odpowiednich czasów sekwencji, aby ucięty drut na pewno spadł do magazynku, a nadjeżdżający nie zablokował maszyny.



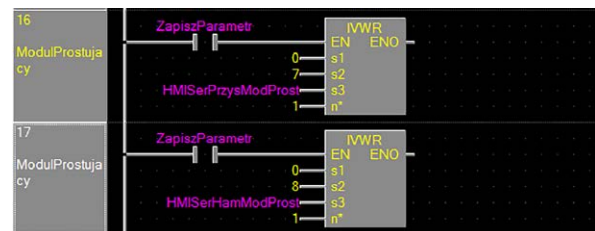
Rysunek 16. Okno edycji ramp



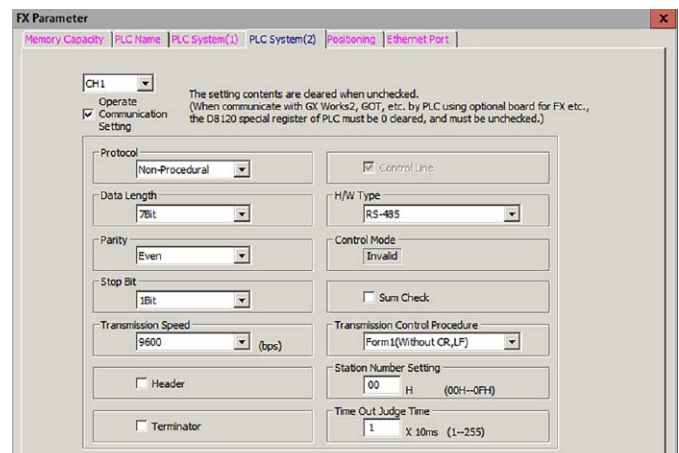
Rysunek 17. Wpisanie wartości granicznych

22	VAR_GLOBAL	HMISerPrzysModProst	Word(Signed)	D955
23	VAR_GLOBAL	HMISerHamModProst	Word(Signed)	D956

Rysunek 18. Deklarowanie zmiennych w oprogramowaniu sterownika



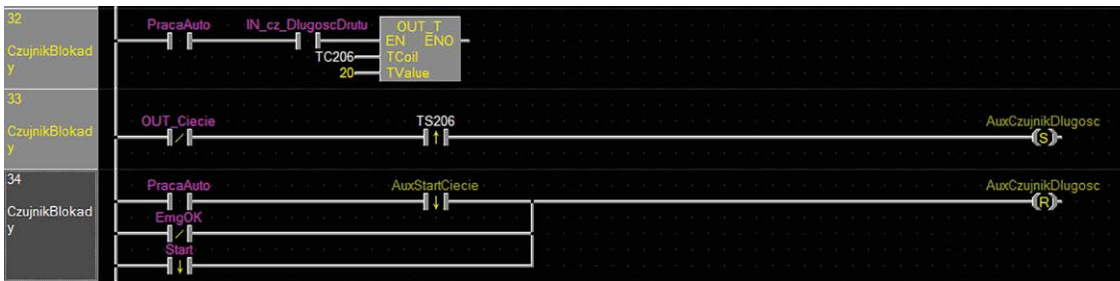
Rysunek 19. Oprogramowanie sterownika PLC



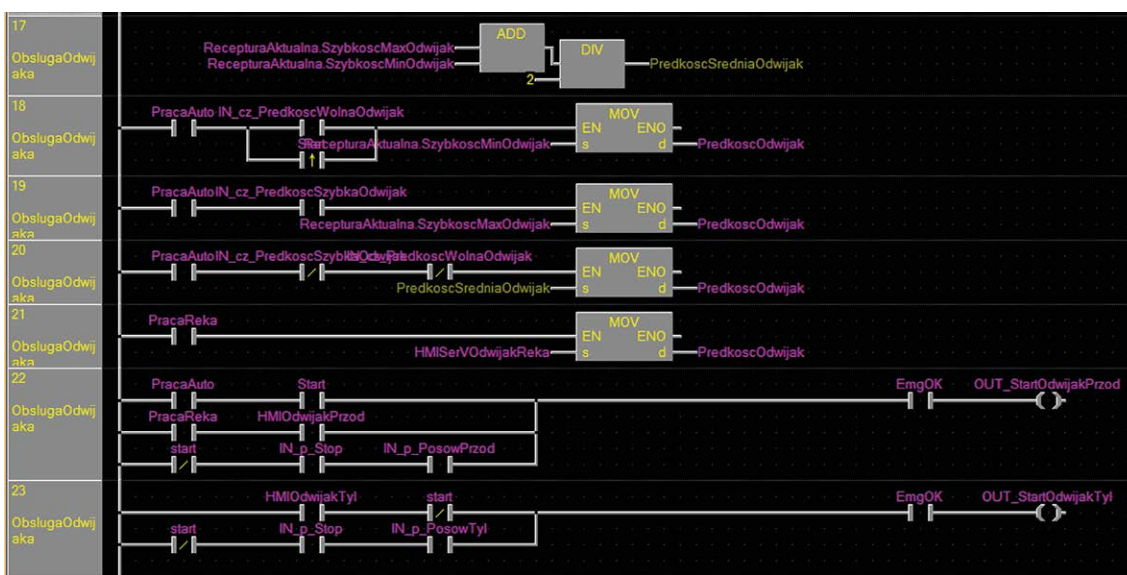
Rysunek 20. Parametry komunikacji



Rysunek 21. Instrukcja odczytania parametrów



Rysunek 22. Utworzenie zmiennej pomocniczej AuxCzujnikDlugosc



Rysunek 23. Obsługa odwiijaka



Rysunek 24. Wizualizacja procesu odwiijania

W pierwszej kolejności tworzę zmienną pomocniczą AuxCzujnikDlugosc, która będzie zapewniała prawidłową informację o pobudzeniu czujnika blokady drutu (rysunek 22). Zmienna ta zostanie ustawiona, gdy czujnik blokady będzie pobudzony przez co najmniej 200 ms, a kasowana po zakończeniu procesu cięcia. Proces cięcia rozpoczyna się z opóźnieniem determinowanym z poziomu menu serwisowego, aby upewnić się, że drut na pewno jest zablokowany w pozycji maksymalnej. Zakończenie cięcia wskazuje czujnik wykrywający jeden obieg mechanizmu mimośrodowo sterującego gilotyną. W tym momencie należy zwrócić uwagę, że nóż tnący musi zawsze zatrzymać się w pozycji wyjściowej, tak aby drut mógł być

swobodnie transportowany poprzez prowadnicę. Razem z uruchomieniem gilotyny jednocześnie otwiera się listwa zrzucająca drut do magazynku. Po czasie liczonym od momentu otwarcia listwy, listwa zamyka się, co kończy cykl. Jeżeli sterujemy napędem cięcia za pomocą falownika warto zastosować metodę hamowania silnika za pomocą prądu stałego. W takim przypadku należy zastosować silnik z obcym chłodzeniem.

Obsługa receptur stanowi dokładnie ten sam kod, który został opisany przy okazji poprzedniego artykułu.

Innym algorytmem, który warto omówić w przypadku maszyny tego typu jest obsługa mechanizmu odwiijaka. Jak już wcześniej wspomniałem mechanizm taki wyposażony jest w siłownik pneumatyczny, na którym są umieszczone dwa czujniki położenia tłoka. Pozycja tłoczyska wskazuje na napięcie ramienia i informuje sterownik czy powinien przyspieszyć obroty odwiijania, czy też zwolnić, aby nie rozplątać zbyt dużej ilości drutu (rysunek 23). Algorytm działa następująco – maszyna zawsze startuje od wolnej

prędkości tj ODWIJAK MIN, po najechaniu na czujnik wskazujący na napięcie ramienia, zwiększa szybkość do ODWIJAK MAX, po zluźnieniu napięcia drutu i najechaniu na wolny zwalnia do szybkości ODWIJAK MIN. Gdy tłok znajduje się pomiędzy czujnikami, jest wyliczana algebraiczna średnia od prędkości minimalnej i maksymalnej i z taką częstotliwością porusza się silnik odwiijaka. Należy zauważyć, że wszystkie zmiany szybkości są dość płynne z uwagi na fakt ustawienia niezerowych ramp. Z punktu widzenia operatora opisany mechanizm jest zwizualizowany na panelu HMI w sposób pokazany na rysunku 24.

Podsumowanie

Działanie maszyny można obejrzeć na filmie <https://goo.gl/UafJQM>. Jest to na tyle prosta aplikacja że można opisać algorytmy z powodzeniem zaimplementować na sprzęcie innych firm.

Tomasz Świątek
tomekfx@o2.pl

