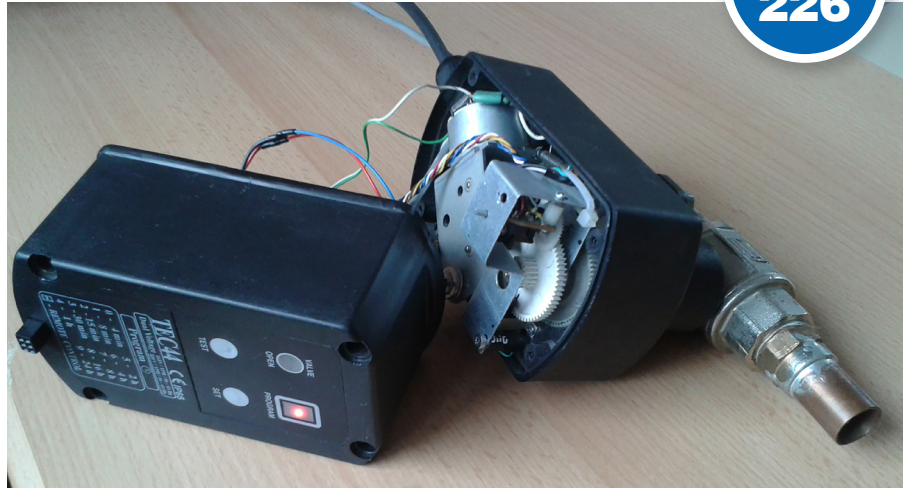


Dział „Projekty Czytelników” zawiera opisy projektów nadesłanych do redakcji EP przez Czytelników. Redakcja nie bierze odpowiedzialności za prawidłowe działanie opisywanych układów, gdyż nie testujemy ich laboratoryjnie, chociaż sprawdzamy poprawność konstrukcji.

Prsimy o nadsyłanie własnych projektów z modelami (do zwrotu). Do artykułu należy dołączyć podpisane oświadczenie, że artykuł jest własnym opracowaniem autora i nie był dotychczas nigdzie publikowany. Honorarium za publikację w tym dziale wynosi 250,- zł (brutto) za 1 stronę w EP. Przesyłanych tekstów nie zwracamy. Redakcja zastrzega sobie prawo do dokonywania skrótów.

Projekt
226

Skonstruowany przez mnie nastawnik znajdzie zastosowanie tam, gdzie jest wymagana elektryczna regulacja przepływu cieczy lub gazu. Jego budowę rozpocząłem z myślą o zastosowaniu w systemie ogrzewania pojazdu samochodowego. Zawór miał regulować przepływ gorącej wody przez nagrzewnicę. W miarę postępu prac projektowych postanowiłem rozszerzyć jego możliwości.



Sterownik zaworu kulowego (2)

Dostęp do menu uzyskuje się po wybraniu kontrolki MODBUS → Właściwości MODBUS (rysunek 9). Tu możemy dokonać ręcznego ustawienia parametrów transmisji dla urządzenia o określonym adresie w celu połączenia się z nim (w trybie offline). Na koniec wybieramy kontrolkę Set, a następnie Połącz (w okienku głównym) i czekamy na nawiązanie połączenia. Jeśli chcemy znaleźć urządzenie o określonym adresie, a nie znamy parametrów komunikacji, wprowadzamy adres urządzenia, a następnie wybieramy Find addr. O znalezieniużądanego urządzenia aplikacja poinformuje nas komunikatem: „Urządzenie znalezione!”. Trwałe połączenie nie zostanie przy tym nawiązane (rysunek 10).

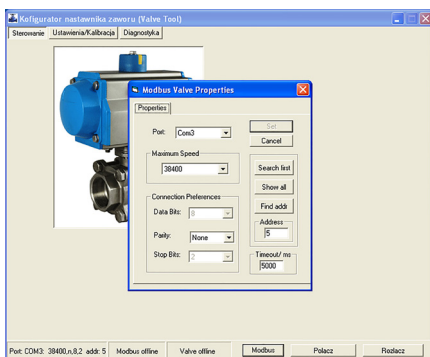
Za pomocą kontrolki Search first dokonujemy z kolei odszukania pierwszego (czasem jedyne)go urządzenia na magistrali. Po uzyskaniu komunikatu o znalezionym urządzeniu klikamy, jak poprzednio, na Set oraz Połącz, zestawiając w ten sposób połączenie. Jeśli chcemy dowiedzieć się ile urządzeń jest aktywnych na magistrali i jakie są ich nastawy komunikacji, używamy przycisku Show all. W dodatkowym okienku zostaną wtedy wyświetlone kolejno etykiety znajdujących, aktywnych urządzeń z ich nastawami transmisji.

Gdy aplikacja zakończy przeszukiwanie, możemy dokonać wyboru określonego urządzenia oraz połączyć się z nim klikając na jego etykietę, a następnie na kontrolki Set

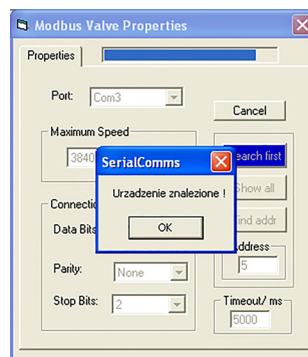
oraz Połącz. Ustawiane przez aplikację parametry komunikacji, zadawane podczas przeszukiwania, widoczne są w okienkach menu Właściwości MODBUS. Proces przeszukiwania możemy przerwać za pomocą kontrolki Cancel. Postęp procesu jest widoczny na pasku. Operacji wyszukiwania nie da się uaktywnić w trakcie trwania połączenia z określonym urządzeniem – konieczne jest najpierw przerwanie połączenia. W trakcie trwania połączenia jest natomiast możliwa zmiana jego parametrów transmisji. Po otwarciu menu Właściwości MODBUS zmieniamy określony parametr, a następnie wciskamy kontrolkę Set. Po chwili następuje restart nastawnika i praca z nową/nowymi nastawami komunikacji. Możliwe prędkości transmisji to: 2400, 4800, 9600, 14400, 19200, 28800, 38400 b/s. Inne parametry transmisji:

- 8 bitów danych.
- Bit parzystości i bity stopu mogą występować w jednej z trzech kombinacji: 1 bit parzystości i 1 bit stopu (Even Parity), 1 bit nieparzystości i 1 bit stopu (Odd Parity), brak bitu parzystości i 2 bity stopu (No Parity).
- Czas oczekiwania (MODBUS Timeout): 1...10 sekund.
- Adresy urządzeń: 1...10.

Do komunikacji nastawnika z komputerem (aplikacją Valve Tool) wykorzystałem adapter USB/RS485.



Rysunek 9. Dostęp do menu uzyskuje się po wybraniu kontrolki MODBUS → Właściwości MODBUS



Rysunek 10. O znalezieniużądanego urządzenia aplikacja poinformuje nas komunikatem: „Urządzenie znalezione!”

Funkcje MODBUS

Te informacje przydadzą się przy tworzeniu oprogramowania do sterownika, nadzorującego pracę nastawnika w sieci MODBUS. Niezbędne okaże się uważne i szczegółowe przeanalizowanie budowy i działania jego programu (**rysunek 11**). Oto skróty opis standardowych funkcji MODBUS wykorzystywanych przez nastawnik. Pamiętajmy, że standard przyjmuje, że rejestry są 16-bitowe, a przesyłanie odbywa się w częściach Hi i Lo (**listing 1** i **listing 2**).

Kalibracja obrotowa (auto-adaptacja)

Nastawnik daje możliwość pracy z różnymi silnikami, na różnych częstotliwościach. Wyboru dokonujemy w zakładce *Ustawienia/kalibracja* (**rysunek 12**). Po wciśnięciu

kontrolki *Edytuj* z rozwijalnej kontrolki *Nastawa silnika* wybieramy żadaną częstotliwość pracy. Mamy do wyboru następujące częstotliwości: 200 Hz; 400 Hz; 800 Hz; 1,6 kHz; 3,2 kHz; 6,4 kHz; 12,8 kHz; 25,6 kHz.

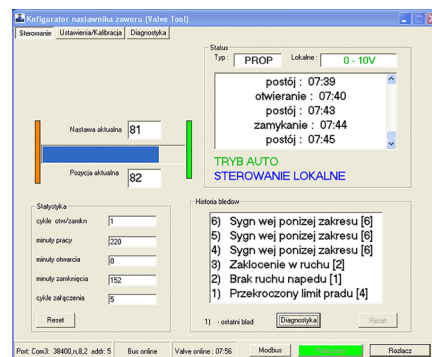
Jakiego wyboru dokonać? Teoretycznie, z dwóch powodów, najlepiej zastosować wyższą częstotliwość przełączeń tj. powyżej 15...20 kHz. Pierwszy powód to dźwięk, który mógłby wydawać silnik (przy niskiej częstotliwości – nieprzyjemny pisk). Częstotliwość 15...20 kHz jest teoretyczną granicą słyszalności człowieka, więc wyższe częstotliwości teoretycznie będą niesłyszalne. Drugi powód jest związany ze stabilnością pracy silnika. Gdyby silnik był zasilany sygnałem o niskiej częstotliwości, w skrajnych przypadkach mogłoby

dojść do sytuacji, w której silnik zatrzymałby się, gdyby tranzystor nie przewodził i ruszał po załączeniu – na zmianę, w zależności od stanu przebiegu PWM. Mówiąc prościej, jeżeli częstotliwość sygnału będzie zbyt mała, obroty silnika będą niestabilne. Im wyższa częstotliwość PWM, tym mniejsze pulsacje napięcia wyjściowego, a zarazem większa stabilność pracy silnika. Nie należy jednak przesadzać z częstotliwością, ponieważ tranzystor może nie zdążyć się w pełni otworzyć lub zamknąć i układ nie będzie działał poprawnie. Zbyt duża częstotliwość sygnału PWM powoduje również wzrost strat na tranzystorze, ponieważ każde przełączenie powoduje straty energii. Tyle teoria. W omawianym projekcie, lepszym wyborem, ze względu na rodzaj zastosowanego przeze mnie silnika, okazuje się zastosowanie jak najniższej częstotliwości pracy. Dzięki temu polepsza się sterowalność prędkością silnika. Zwiększamy wtedy różnicę między prędkością maksymalną a minimalną nastawnika, rozszerzając jej zakres. Ma to istotne znaczenie, ponieważ program, uwzględniając różnicę między wartością zadaną położenia a rzeczywistą, wylicza prędkość zadaną. Większy uchyb wymusza większą prędkość obrotową nastawnika. Im uchyb mniejszy, tym nastawnik bardziej zwalnia, a to ma znaczenie wzięwszy pod uwagę jego dokładność oraz stabilność. W praktyce, jeśli np. zadamy otwarcie zaworu na 100%, jeśli uprzednio był

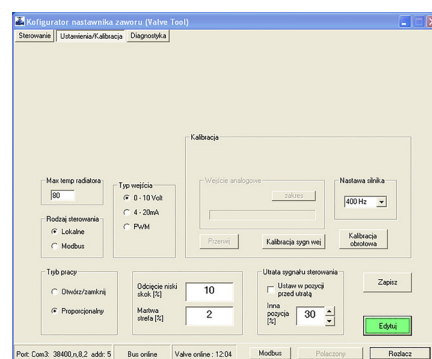
```
Listing 1. Funkcje MODBUS zaimplementowane w nastawniku
//zapis do nastawnika jednego rejestru z MASTER-a (kod 6/0x06/)
u08 writeSingleReg(void)
/*
MASTER wysła:
Slave addr/byte 0 - adres nastawnika
Function code /1 - kod funkcji 6(0x06)
HI register addr /2
LO register addr /3 - adres rejestru w nastawniku
HI register content /4
LO register content /5 - wartość rejestru do zapisu
LO CRC /6
HI CRC /7 - kontrola poprawności transmisji
////////////////////////////////////
Odpowiedź nastawnika:
RESPONSE FROM SLAVE = ECHO (nastawnik odpowiada tak samo)
*/

//zapis do nastawnika x rejestrów z MASTER-a (kod 16/0x10)
u08 writeMultipleReg(void)
/*
MASTER wysła:
Slave addr /byte 0 - adres nastawnika
Function code /1 - kod polecenia 16(0x10)
Starting Address Hi /2
Starting Address Lo /3 - adres pierwszego rejestru w nastawniku
No. of Registers Hi /4
No. of Registers Lo /5 - liczba przesyłanych rejestrów (Byte Count x 2)
Byte Count /6 - liczba przesyłanych bajtów
Data Hi /7
Data Lo /8 - wartości pierwszego rejestru do zapisu w nastawniku
=====
Data Hi/n //Data Lo/n+1 - wartości ostatniego rejestru
LO CRC/n+1 //HI CRC/n+1 - kontrola poprawności transmisji
////////////////////////////////////
Odpowiedź nastawnika
Slave Address /0
Function code /1
Starting Address Hi /2
Starting Address Lo /3
No. of Registers Hi /4
No. of Registers Lo /5
LO CRC/6
HI CRC/7
*/

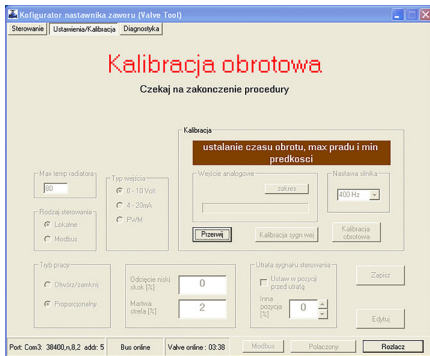
//odczyt z nastawnika x rejestrów przez MASTER(kod 3/0x03/)
u08 readHoldingReg(void)
/*
z MASTERA
Slave addr /byte 0 - adres nastawnika
Function code /1 - kod funkcji 3(0x03)
Starting Address Hi /2
Starting Address Lo /3 - adres pierwszego rejestru w nastawniku
Quantity of Registers Hi/4
Quantity of Registers Lo/5 - liczba rejestrów do odczytu
LO CRC /6
HI CRC /7 - kontrola poprawności transmisji
////////////////////////////////////
Odpowiedź z nastawnika:
Slave addr /0
Function code /1
Byte count /2 - liczba przesyłanych bajtów(Quantity of Registers x 2)
Register value Hi /3
Register value Lo /4 - wartości pierwszego, przesyłanego rejestru
=====
Register value Hi /n
Register value Lo /n+1 - wartości ostatniego, przesyłanego rejestru
LO CRC /n+1
HI CRC /n+1 - kontrola poprawności transmisji
```



Rysunek 11. Okno główne programu



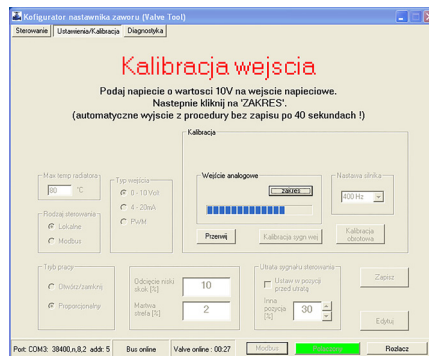
Rysunek 12. Nastawnik daje możliwość pracy z różnymi silnikami, na różnych częstotliwościach. Wyboru dokonujemy w zakładce *Ustawienia/kalibracja*



Rysunek 13. Po potwierdzeniu chęci wykonania operacji, na wybranej przez nas częstotliwości, nastawnik wykonuje procedurę autoadaptacji

całkowicie zamknięty, w początkowej fazie zawór otwierać się będzie bardzo szybko, a potem coraz bardziej będzie zwalniał, aż do całkowitego zatrzymania. Dzięki temu będzie miał mniejsze „skłonności” do oscylacji i będzie działał bardziej płynnie.

Powróćmy do naszej procedury. Po wybraniu częstotliwości pracy silnika wciśkami kontrolkę *Kalibracja obrotowa*.



Rysunek 14. Przebieg autoadaptacji, w tym kolejność jego etapów, można śledzić w zakładce Ustawienia/kalibracja, w oknie Kalibracja

Po potwierdzeniu chęci wykonania operacji, na wybranej przez nas częstotliwości, nastawnik wykonuje procedurę autoadaptacji (**rysunek 13**), aby optymalnie dostosować się nie tylko do specyfiki silnika, ale również mechanizmu. W grę wchodzi tutaj właściwości silnika, opory mechaniczne przekładni oraz samego zaworu. Pamiętać należy też o tym, że np. zmniejszanie

częstotliwości pracy silnika narzucać będzie potrzebę zwiększenia wypełnienia impulsu po to, aby utrzymać odpowiednio wysoką wartość napięcia średniego podawanego na silnik. Wejście w procedurę jest sygnalizowane poprzez szybkie, dwukolorowe miganie diody LED na nastawniku. W pierwszym etapie autoadaptacji zostaje wyznaczony zakres obrotowo – impulsowy (liczba impulsów zliczonych z czujników OPT1A-B, która przypada na pełny obrót nastawnika), maksymalna prędkość obrotowa oraz maksymalny prąd silnika. W drugim etapie zostaje wyznaczona minimalna prędkość obrotowa, zarówno podczas zamykania, jak i otwierania w pełnym zakresie obrotowym. Ta procedura polega na stopniowym zwiększaniu wypełnienia impulsu (momentu obrotowego), aby silnik ruszył przezwyciężając opory mechanizmu, a następnie pokonywał je w pełnym zakresie obrotowym. Jeśli zatrzyma się z powodu zwiększonych oporów, to jest zwiększane wypełnienie impulsu do momentu wystąpienia ruchu mechanizmu. Procedura zostaje zakończona po wykonaniu obrotu o $360^{\circ} \pm 20^{\circ}$. W trzecim

Listing 2. Wykaz i znaczenie rejestrów

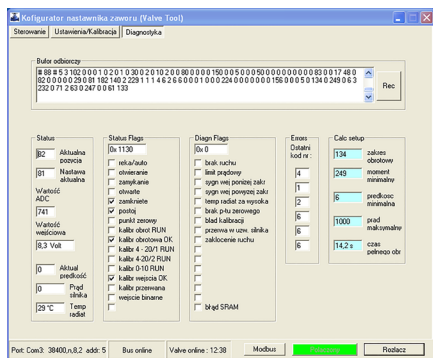
```

PositDemandModbus_reg [0] //write/read, pozycja - wartość zadana MODBUS (0-100%)
LocalInputContrMode_reg [1] //write/read, konfiguracja wejścia // 1-10V = 1 /4-20mA = 2 /PWM = 3
valveControlType_reg [2] //write/read, typ zaworu //on/off = 1 /prop = 2
ControlSource_reg [3] //write/read, źródło sterowania //local = 1/modbus = 2
LossContrInpPosit_reg [4] //write/read, ustawienie zaworu na pozycji po utracie sygnału wejściowego (0-100%)
DeadBand_reg [5] //write/read, strefa martwa - histereza
LowTravelCutoff_reg [6] //write/read, wymuszenie zamknięcia - poniżej tej wartości (0-100%)
Freq_veloc_motor_reg [7] //read/write, częstotliwość pracy silnika /prędkość napędu
RadiatorTempLimit_reg [8] //read/write, wartość graniczna temperatury radiatora (stopnie Cels.)
CalibrRotateStatus_reg [9] //read/write, status kalibracji obrotowej
CommandStatus_reg [10] //read/write, polecenie z MASTER - MODBUS
//=====
// MODBUS komunikacja
CommBaudrate_reg [11] //write/read, prędkość transmisji
CommAddress_reg [12] //write/read, adres urządzenia
CommParity_reg [13] //write/read, typ parzystości
CommTimeout_reg [14] //write/read, maksymalny czas oczekiwania na nawiązanie połączenia
//=====
ResetDiagnostic_reg [15] //write/read, command-polecenie kasowania błędów
ResetStatistics_reg [16] //write/read, command-polecenie zerowania danych statystyki
//unused1_reg [17]
//unused2_reg [18]
LocalInputValueUnit_reg [19] //read, wartość sygnału wejściowego w jednostkach (V, mA)
CalibrateDemand_reg [20] //write/read, command-kalibracja, polecenie operacji (1-5)
StatusFlags_reg [21] //read, flagi statusowe
ActualPosition_reg [22] //read, aktualna pozycja (0-100%)
DiagnosticFlags_reg [23] //read, flagi diagnostyczne
MotorCurrent_reg [24] //read, prąd silnika (0-1023)
RadiatorTemperature_reg [25] //read, temperatura radiatora (stopnie Cels.)
ActualDemandSource_reg [26] //read, pozycja-wartość zadana
TestRegister_reg [27] //read, kontrola poprawności komunikacji
LocalInputValue_reg [28] //read, wartość sygnału wejściowego ADC (0-1023)
Serial Firmware_reg [29] //read, numer Firmware
//=====
//SRAM LOCATION //=====
FaultMemory1_reg [30] //write/read, pamięć błędów-rejestr1
FaultMemory2_reg [31] //write/read, pamięć błędów-rejestr2
FaultMemory3_reg [32] //write/read, pamięć błędów-rejestr3
//=====
// Statystyka
NumberCycles_regH [33] //write/read, liczba cykli zaknij-otwórz
NumberCycles_regL [34] //write/read
HoursRunning_regH [35] //write/read, liczba minut pracy
HoursRunning_regL [36] //write/read
HoursOpen_regH [37] //write/read, liczba minut otwarcia
HoursOpen_regL [38] //write/read
HoursClosed_regH [39] //write/read, liczba minut zamknięcia
HoursClosed_regL [40] //write/read
NumberPwrUpCycl_regH [41] //write/read, liczba cykli załączeń
NumberPwrUpCycl_regL [42] //write/read
//=====
// Parametry kalibracji obrotowej
ValveCounterRange_reg [43] //write/read, zakres obrotowo - impulsowy
Start_pwmDuty_reg [44] //write/read, minimalne wypełnienie PWM
VelocityMinValue_reg [45] //write/read, zarejestrowana minimalna prędkość zaworu
CurrentMaxValue_reg [46] //write/read, zarejestrowany maksymalny prąd silnika
RotateTime_reg [47] //write/read, zarejestrowany czas otwierania
//=====
// END SRAM LOCATION //=====
TemperatureCount_reg [48] //read, temperatura radiatora(0-1023)
Status1_reg [49] //read, unused
Status2_reg [50] //read, unused

```


Listing 3. Rodzaje i opis błędów

(!) - błąd krytyczny
NO_MOV_ERR (1 !) - brak ruchu, wystąpi gdy silnik wysterowany, kalibracja nieaktywna i zerowa prędkość, po opóźnieniu **NO_MOVE_DELAY**
ANOM_MOV_ERR (2) - zakłócenie ruchu, wystąpi jeśli prędkość obrotowa mniejsza od minimalnej a większa od zera, po opóźnieniu **ANOM_MOVE_DELAY**
NO_ZERO_MOV_ERR (3 !) - brak punktu zerowego, wystąpi w trakcie zamykania gdy nie ma kalibracji obrotowej i przekroczona 1/2 czasu pełnego obrotu przy minimalnej
CURR_OVERLOAD_ERR (4 !) - przekroczony limit prądu silnika - wystąpi jeśli nie ma kalibracji obrotowej a wartość prądu silnika jest większa od jej maksimum (zarejestrowanej w trakcie kalibracji), po opóźnieniu **CURRENT_OVERLOAD_DELAY**
MOTOR_BREAK_ERR (5 !) - przerwa w obwodzie silnika - wystąpi jeśli silnik wysterowany, jego prędkość zerowa i prąd zerowy, po opóźnieniu **noCurrentDelay = NO_CURRENT_DELAY_CALIBR** (dłuższe dla kalibracji obrotowej) lub **noCurrentDelay = NO_CURRENT_DELAY_NORM** (krótsze dla normalnej pracy)
LOW_INP_ERR (6) - sygnał wejściowy poniżej zakresu, wystąpi dla wejścia 1...10 V gdy sygnał na wejściu poniżej 1 V, dla wejścia 4...20 mA, gdy sygnał poniżej 4 mA, po opóźnieniu **INPUT_ERROR_TIME** (z uwzględnieniem offsetów)
OVER_INP_ERR (7) - sygnał wejściowy powyżej zakresu (z uwzględnieniem offsetów)
TEMP_ERR (8 !) - przegrzany radiator, temperatura radiatora przekroczyła programowalny limit
CALIBR_ERR (9) - błąd kalibracji obrotowej; żądanie przerwania kalibracji z poziomu aplikacji lub z panelu za pomocą przycisku SET lub minął jej czas albo istnieje przerwa w obwodzie silnika lub przekroczona temperatura radiatora (w trakcie kalibracji)
MODBUS_TIMEOUT_ERR (10) - brak połączenia z urządzeniem MASTER MODBUS przez określony czas
SRAM_ERR (11) - błąd odczytu pamięci SRAM



Rysunek 15. Podgląd wartości procedury autoadaptacji

etapie autoadaptacji następuje wyznaczenie czasu pełnego obrotu oraz ciąg dalszy ustalenia maksymalnego prądu silnika, tym razem przy prędkości minimalnej. Ustalona wartość tego prądu (w funkcji czasu) jest brana pod uwagę w trakcie kontroli mającej na celu zasygnalizowanie przeciążenia prądowego. Czas pełnego obrotu jest brany pod uwagę w związku z monitorowaniem punktu zerowego (braku jego wystąpienia po upływie tego czasu). Prędkość minimalna (w funkcji czasu) potrzebna jest do zasygnalizowania anomalii tzn. zwolnienia ruchu nastawnika.

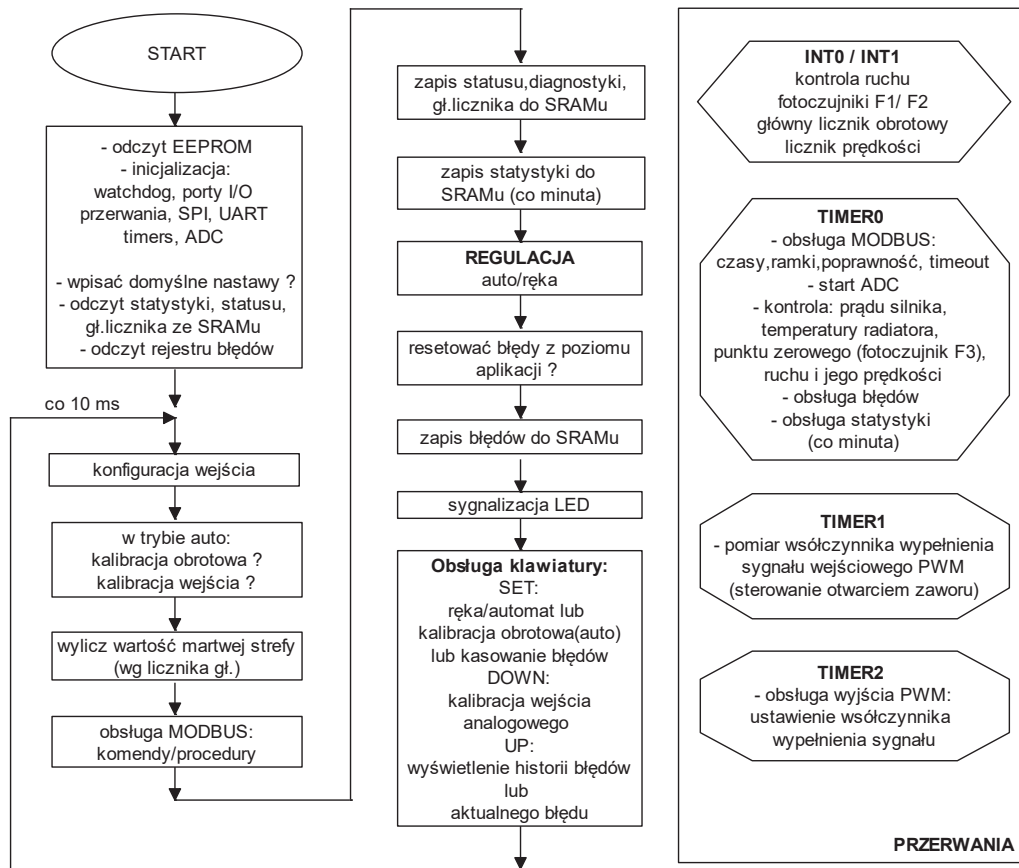
Czas trwania kalibracji obrotowej jest monitorowany na bieżąco. Jeśli któryś z etapów procedury przekroczy 50 sekund, nastąpi wyjście z procedury autoadaptacji i zasygnalizowanie błędu. Stanie się tak również wtedy, gdy przez określony w nastawach programu czas, nie będzie wykrywany prąd w obwodzie silnika. Przebieg autoadaptacji, w tym kolejność jego etapów, można śledzić w zakładce *Ustawienia/kalibracja*, w oknie *Kalibracja* (rysunek 14) oraz

w zakładce *Diagnostyka*, w oknie *Kalibracja obrotowa*. W zakładce *Diagnostyka*, w ramce *Cal setup* (rysunek 15) mamy dodatkowo podgląd wartości procedury autoadaptacji. Zakończenie wyznaczania danej wartości jest sygnalizowane przez zmianę koloru jej okienka na seledynowy. Z poziomu pracy bez aplikacji, autoadaptację uruchamiamy poprzez długie wciśnięcie przycisku SET w trybie automatacyjnym. W tym trybie, domyślna częstotliwość pracy silnika ustawiona będzie na 6,4 kHz. Przerwywamy procedurę przez ponowne wciśnięcie przycisku SET. W trybie pracy z aplikacją, autoadaptację można w każdej chwili zakończyć przy użyciu kontrolki *Przerwij* w oknie *Kalibracja*. W przypadku przerwania procedury

(w jakimkolwiek trybie) wymagana będzie ponowna autoadaptacja nastawnika.

Kalibracja wejścia

Aby zapewnić dokładne odwzorowanie położenia nastawnika względem sygnału wejściowego, wymagane jest też przeprowadzenie kalibracji wejścia. Dotyczy to wejścia napięciowego oraz prądowego. Z poziomu pracy z aplikacją do procedury kalibracji wejścia napięciowego wchodzimy wciskając kontrolkę *Edit* w oknie *Ustawienia/Kalibracja*, a następnie *Kalibracja sygn. wej.* Oczywiście, wcześniej nastawnik musi być skonfigurowany do pracy z wejściem napięciowym. Po potwierdzeniu chęci wykonania procedury podajemy, zgodnie ze wskazówką aplikacji, na wejście napięciowe nastawnika



Rysunek 16. Algorytm pracy nastawnika

10 V, po czym wciskamy kontrolkę *Zakres*. Otrzymujemy informację o pomyślnym wykonaniu kalibracji, chyba że podane napięcie było zbyt niskie lub za wysokie. Do procedury z poziomu pracy bez aplikacji dostajemy się poprzez dłuższe wciśnięcie przycisku *ZAMKNIJ* w trybie automatycznym. Przycisk trzymany wciśnięty do momentu, gdy szybko zacznie pulsować zielona dioda LED. Następnie, po podaniu napięcia 10 V, wciskamy przycisk *SET*. Nastawnik potwierdza przyjęcie wartości 5-krotnym mignięciem czerwonej diody LED, po czym wychodzi z procedury, chyba że podane napięcie nie mieści się w akceptowalnym zakresie.

W podobny sposób wchodzimy się do procedury kalibracji wejścia prądowego pod warunkiem, że konfiguracja nastawnika uwzględnia wejście prądowe. Procedura składa się z dwóch etapów. W pierwszym wzorcujemy sygnał minimalny, w drugim maksymalny. Postępujemy tutaj zgodnie ze wskazówkami aplikacji – podajemy prąd o natężeniu 4 mA i potwierdzamy to kontrolką *Zero*, następnie podajemy 20 mA i potwierdzamy to kontrolką *Zakres*. Wzorcowanie sygnału minimum jest sygnalizowane przez szybkie pulsowanie zielonej LED, a sygnału maksimum przez wolne pulsowanie. Również tutaj podane sygnały wejściowe muszą mieścić się w akceptowalnym zakresie. Błąd powoduje powrót do początku procedury. W przypadku powodzenia następuje wyjście z niej, co jest sygnalizowane 5-krotnym mignięciem czerwonej diody LED. Po wejściu do procedury z poziomu pracy bez aplikacji, podajemy 4 mA (w trakcie szybkiego pulsowania zielonej LED) i potwierdzamy przyciskiem *SET*. Teraz dioda pulsuje wolno – podajemy 20 mA i ponownie potwierdzamy. Następuje wyjście z procedury, co jest sygnalizowane pulsowaniem diody LED na czerwono.

Procedurę kalibracji obu wejść możemy w każdej chwili przerywać: z poziomu pracy z aplikacją – za pomocą kontrolki *Przerwij*, a z poziomu pracy bez aplikacji za pomocą przycisku *Otwórz*. Nastawnik przechodzi wtedy w stan pracy awaryjnej, co wymusza potrzebę ponownej kalibracji wejścia. Komunikaty o błędach zamieszczono na **listingu 3**.

W razie trudności może się okazać konieczne wprowadzenie do pamięci nastawnika wartości domyślnych. Operacji dokonujemy przez równoczesne wciśnięcie przycisków *Otwórz* – *Zamknij* przy odłączonym, a następnie podanym zasilaniu. Gdy dioda LED1 zacznie szybko migać na czerwono, puszcza przyciski i kiedy zmienia kolor na zielony, potwierdzamy operację za pomocą przycisku *SET*. Nastawnik wprowadza wtedy do pamięci EEPROM następujące nastawy domyślne:

- Sterowanie lokalne – proporcjonalne, 1...10 V (jeśli brak zworek JP6, JP7).
- Nastawa pozycji w razie utraty sygnału = 25%.
- Strefa martwa = 2%.
- Odcięcie dolnego zakresu = 0%.
- Częstotliwość kluczowania silnika = 6,4 kHz.
- Limit temperatury radiatora = 80°C.
- Prędkość transmisji = 9600 b/s, adres urządzenia = 5, bez parzystości, modus timeout = 5 s.
- Zakres obrotowy = 100.
- Zakres 1...10 V = 1000.
- Zakres min 4...20 mA = 100.
- Zakres 4...20 mA = 1000.
- Tryb automatyczny.

Po przyjęciu tych wartości następuje zerowanie rejestrów statusu i diagnostyki, licznika obrotowego oraz statystyki w pamięci SRAM. Wykonanie tych operacji umożliwia zainicjowanie urządzenia. Następnie jest wymagana kalibracja obrotowa. Dla ułatwienia analizy pracy nastawnika na **rysunku 16** zamieszczono algorytm pracy nastawnika.

Grzegorz Sipiora
dzesek@gmail.com

ELEKTRONIKA PRAKTYCZNA

teraz zawsze z Tobą w wersji mobilnej



REKLAMA

m.ep.com.pl