

Mikrokontrolery do sterowania silnikami serii ST7MC, część 2

Dodatkowe materiały do artykułu
Publikujemy na CD-EP oraz www.ep.com.pl

Sterowanie trójfazowym silnikiem indukcyjnym

Trójfazowy asynchroniczny silnik indukcyjny wymaga zupełnie innego sposobu sterowania niż opisany powyżej, ale zmiana ta polega jedynie na odpowiednich modyfikacjach rejestrów konfiguracyjnych kontrolera. Układ *Motor Control Macrocell* w tej konfiguracji pracuje jako trójfazowy generator PWM, wytwarzający sygnały o obwodni sinusoidalnej. Z każdego z trzech wyjść generatora wytwarzane są dwa sygnały komplementarne doysterowania dolnego i górnego tranzystora mostka. Dodatkowo wprowadzany jest tzw. czas martwy (*Dead Time*) – opóźnienie między wyłączeniem dolnego, a załączeniem górnego tranzystora; pozwala to uniknąć jednoczesnego przewodzenia dwóch tranzystorów w tej samej gałęzi mostka. Uproszczony schemat układu sterowania silnikiem indukcyjnym przedstawia **rys. 5**. Doysterowania bramek tranzystorów mocy zastosowano trzy układy L6386 (specjalizowany driver dolnego i górnego tranzystora w mostku wysokonapięciowym). Wejścia MCIX mogą być wykorzystane do realizacji układu sterowania z zamkniętą pętlą regulacji obrotów silnika.

Rejestry konfiguracyjne umożliwiają dostosowanie wejść MCIX do wybranego rodzaju przetwornika obrotowo – impulsowego oraz wybór trybu pracy: pomiar pozycji, prędkości ob-

Dobrze znana Czytelnikom EP rodzina 8-bitowych mikrokontrolerów ST7 wzbogaciła się o serię układów dedykowanych do układów sterowania silnikami z wirującym magnesem (BLDC) oraz silnikami indukcyjnymi. Firma ST Microelectronics wprowadziła w nich wiele nowinek technicznych, nie występujących w innych kontrolerach „silnikowych”.

rotowej lub prędkości i kierunku (dla enkodera inkrementalnego). Korzystając z gotowych bibliotek dla języka C, można zbudować układ z zamkniętą pętlą o parametrach porównywalnych z falownikiem wykorzystującym procesor DSP (soft – start, soft – stop, optymalizacja parametrów pracy silnika, regulacja poślizgu).

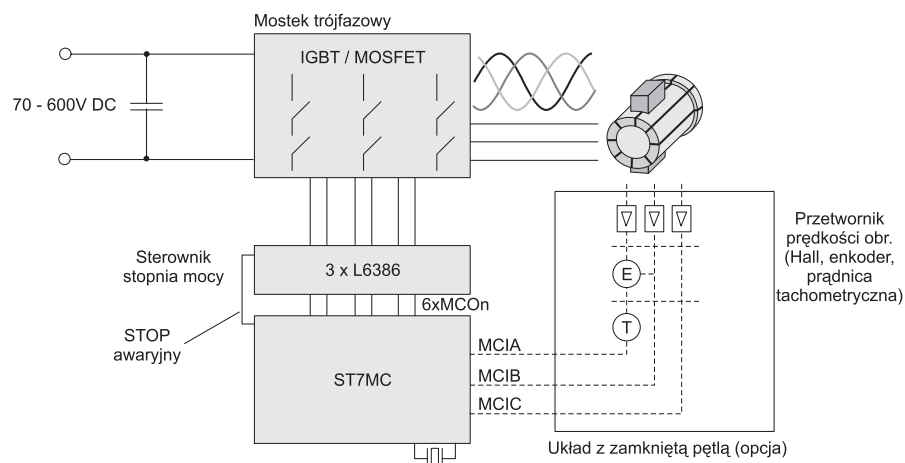
W podobny sposób realizuje się sterowanie silnikiem prądu zmiennego z wirującym magnesem (PMAC – Permanent Magnet AC Motor).

nego z wirującym magnesem (PMAC – Permanent Magnet AC Motor).

Dodatkowe peryferia w bloku Motor Control Macrocell

Blok sterowania silnikiem jest wyposażony w jeszcze kilka przydatnych układów i funkcji:

- Możliwość konfiguracji wyjść MCON: dla każdego z sześciu wyjść można wybrać jego pola-



Rys. 5. Schemat sterowania silnika indukcyjnego

ryzację (wyjście proste lub zane-gowane) oraz stan w czasie re-se-tu (niski, wysoki, wysoka impe-dancja).

- Wejście stopu awaryjnego: wy-stero-wanie tego wejścia powoduje na-tychmiastowe wyłączenie silnika – wyjścia MCO ustawiane są w stan zdefiniowany dla resetu. Ten sam efekt można osiągnąć także pro-gramowo – ustawiając bit stopu w odpowiednim rejestrze.
- Komparator analogowy: może być wykorzystany do regulacji lub ograniczania prądu uzwojeń. Napięcie referencyjne kompara-tora można ustawić programowo na jedną z 7 wartości z zakresu 0,2 do 3,5 V.
- Dodatkowy 8-bitowy timer z pro-gramowanym preskalerem i roz-budowanymi funkcjami porówna-nia/przechwytywania. Doskonale się nadaje do pomiaru prędkości obrotowej.
- Wzmacniacz operacyjny: ma wy-prowadzone oba wejścia i wyjście, co umożliwia dowolne jego wy-korzystanie (wzmacniacz wstępny dla A/C, separator impedancji, komparator itp.). Poza tym po-przez ustawienie odpowiednich bitów konfiguracyjnych, wzmac-niacz może być wewnętrznie dołą-czony do wybranego wejścia przetwornika A/C lub do kompa-ratora sprzężenia prądowego.

Programowanie mikrokontrolerów ST7MC

Najbardziej efektywne jest wy-korzystanie gotowych bibliotek dla języka C, opracowanych przez firmę STM. Biblioteki oraz noty aplika-cyjne z przykładami można pobrać bezpłatnie w dziale *Downloads* na stronie firmowej www.mcu.st.com. Dostępne są oddzielne biblioteki do sterowania silnikiem indukcyjnym, silnikiem BLDC oraz synchronicz-nym silnikiem AC z wirującym ma-gnesem. Biblioteki zawierają funk-cje specyficzne dla danego rodzaju silnika, a także funkcje pomocnicze

List. 1. Przykładowy program sterowania silnika BLDC

```
While(1) //main loop
{
  if (Chk_Timer_WDG_Elapsed() == TRUE)WWD_Refresh();
  Chk_Power_Motor_Status();

  if ((u8)(GetMotorStatus() & FAULT_MSK) != 0)State = FAULT; //START_UP_FAILED
                                                                    //or MOTOR_STALLED
                                                                    //or HARD_FAILURE
                                                                    //or EMERGENCY_STOP?

  switch (State)
  {
  case IDLE:
    if (timer_10ms == 0) PORTS_RedLedOn(); //red LED back to normal after
                                                // overvoltage, overtemperature detection
    if(key_scan() == TRUE)State = START;
    break;

  case START:
    if(MTC_StartMotor() == TRUE) State = RUN;
    break;

  case RUN:
    if(GetMotorStatus() & AUTO_SWITCH)
    {
      if(ValBit(Flag_MTC,SAMP_EVT)) //update PWM?
      {
        Falling_bemf = (u8)(Get_RV3()); //read RV3 & set falling
                                          coefficient accordingly
        Rising_bemf = (u8)(Get_RV2()); //read RV2 & set rising Bemf
                                          //coefficient accordingly
        Set_Duty((u16)(PWM_FREQUENCY*25/100)); //25% duty cycle
      }
    }

    Chk_Motor_Stalled();
    if(timer_10ms == 0) PORTS_GreenLedOn();
    if(key_scan() == TRUE)State = BRAKE;
    break;

  case BRAKE:
    if(active_brake((PWM_FREQUENCY*17/100),2000) == TRUE)State = STOP;
                                                                    //Brake duty 17%
    break;

  case STOP:
    MTC_StopMotor();
    PORTS_RedLedOn();
    State = IDLE;
    break;

  case FAULT:
    default:
      MTC_StopMotor();
      PORTS_RedLedOn();
      if((u8)(GetMotorStatus() & FAULT_MSK) == 0)State = IDLE;
    break;
  }
}
```

nie związane z silnikiem (obsługa klawiatury, zapalenie LED, przetwarzanie analogowo – cyfrowe, obsługa transmisji szeregowej itp.).

Na list. 1 przedstawiono przykła-dowy program sterowania silnikiem BLDC w trybie bezczujnikowym. Pro-gram realizuje następujące funkcje:

- Po wciśnięciu przycisku START silnik jest uruchamiany ze współ-czynnikiem wypełnienia 25%.
- Po wciśnięciu przycisku STOP na 2 sekundy uruchamia się hamowa-nie dynamiczne ze współczynniki-em wypełnienia 17%, następnie stopień mocy zostaje wyłączony.
- Dodatkowo zapalane są diody

sygnalizacyjne: czerwona lub zie-lona oraz obsługiwany jest stan awaryjny zablokowania wirnika (*Motor Stalled*).

Uwaga: Przykład nie zawiera czę-ści konfiguracyjnej – np. deklaracje zmiennych, ustawienia rejestrów, przyporządkowanie końcówek we/wy.

Wybrane funkcje biblioteczne (zastosowane w przykładowym pro-gramie) opisane są poniżej:

MTC_StartMotor: zainicjowanie odpowiednich rejestrów i zmiennych, pozycjonowanie wirnika, ustawienie maski przerwań niezbędnych podczas pracy silnika, uruchomienie silnika.

MTC_StopMotor: wyłączenie

www.dtw.com.pl

toroidalne

transformatory mocy 50-400Hz (1-30 000VA), transformatory
mocy do przetwornic SPMS, precyzyjne transformatory
pomiarowe (przekładniki) prądu i napięcia, elementy
indukcyjne do filtrów, do przetwornic impulsowych,
elementy czujników, transformatory Ferrantiiego,
i inne wyżej nie wymienione.

automatyka
akustyka
przemysł
pedagogiczna

elektronika

www.dtw.com.pl

dtw elektronika
ul. krakowska 390, 32-080 zabierzów, poland, tel.: 0048/12/283 09 50, fax:0048/12/285 35 67

wszystkich tranzystorów mostka, zablokowanie przerwian związanych z silnikiem.

GetMotorStatus: zwraca 8-bajtowe słowo stanu silnika, wybrane bity: MOTOR_STALLED, AUTO_SWITCH, EMERGENCY_STOP.

Set_Duty (u16 duty): ustalenie wartości 16-bitowej zmiennej – współczynnika wypełnienia sygnału PWM, tzn. ustalenie prędkości obrotowej (momentu w trybie prądowym).

Chk_Motor_Stalled: wykrywanie stanu zablokowania wirnika poprzez kontrolę wartości bitów MTC w rejestrze MPCR kontrolera. Wartość maksymalna $MTC=15$ powoduje ustawienie bitu MOTOR_STALLED.

active_brake (u16 duty, u16 time): funkcja hamowania dynamicznego poprzez zasilenie jednego uzwojenia prądem stałym i zwarcie pozostałych. Dwie zmienne 16-bitowe: współczynnik wypełnienia PWM (*duty*), czas hamowania w ms (*time*). Zwraca TRUE po upływie czasu hamowania.

u8 GetRVn(): funkcja zwraca ustawienie potencjometru dołączonego do jednego z wejść analogowych kontrolera (RV1, RV2 lub RV3) jako wartość 8-bitową. W załączonym przykładzie potencjometry RV2 i RV3 ustalają wartości progowe dla komparatora napięcia BEMF.

key_scan(): zwraca TRUE gdy klawisz dołączony do pinu kontrolera jest wciśnięty przez określony czas. Funkcja realizuje filtrację drgań styków.

Biblioteka dla silnika trójfazowego AC zawiera odmienny zestaw funkcji, dostosowany do specyfiki silnika, poniżej opisano wybrane funkcje podstawowe:

- Funkcje inicjujące rejestry i zmienne: MTC_ResetPeripheral, MTC_InitPeripheral, MTC_InitSineGen. Służą do ustalenia konfiguracji pracy silnika.
- Funkcje ustalające napięcie i częstotliwość fali sinusoidalnej oraz kierunek obrotów (kolejność faz): MTC_UpdateSine, MTC_Toggle_Direction, MTC_Set_ClockWise_Direction, MTC_Set_CounterClockWise_Direction.
- Funkcje odczytu bieżących parametrów (napięcia, prędkości, poślizgu, kierunku obrotów): MTC_GetVoltage, MTC_GetStatorFreq, MTC_GetSlip, MTC_GetRotation_Direction.
- Funkcje hamowania dynamiczne-

go: MTC_StartBraking, MTC_Brake, MTC_StopBraking.

Poza funkcjami podstawowymi klasy MTC dostępne są także bardziej rozbudowane procedury aplikacyjne (klasa ACM), umożliwiające sterowanie silnikiem AC w trybie falownikowym. W układzie z otwartą pętlą sprzężenia sterowanie odbywa się w podstawowym trybie stałego stosunku U/f, natomiast w układzie z pomiarem prędkości stosowane jest sterowanie skalarne z kontrolą poślizgu i optymalizacją sprawności silnika.

Procedury ACM nie mają bezpośredniego dostępu do zasobów sprzętowych, działają na zasadzie wywołań funkcji MTC z odpowiednimi parametrami.

- Procedury miękkiego startu dla zamkniętej i otwartej pętli: ACM_SoftStart, ACM_SoftStartOL.
- Aktywacja trybu optymalizacji poślizgu: InitSlipFreqReg, ACM_SlipRegulation. Dla celów pomiarowych stosowana jest także funkcja utrzymania stałych obrotów przez określony czas ACM_SustainSpeed.
- Funkcje odczytu parametrów pętli regulacji proporcjonalnej i poślizgu: ACM_GetPIParam, ACM_GetOptimumSlip.

Programista może skorzystać z predefiniowanych charakterystyk regulacji (miękkiego startu, U/f, skalarnej), oprócz tego w notach aplikacyjnych opisane są metody doboru charakterystyk do specyficznych wymogów aplikacji.

Znacznym ułatwieniem tworzenia aplikacji może być zestaw ewaluacyjny, oferowany przez firmę STM. Zestaw zawiera kompletny sterownik ze stopniem mocy, wejścia przetworników prędkości i czujników Halla, izolowany port szeregowy z konwerterem USB, dodatkowy port szeregowy LIN/SCI, klawisze, potencjometry. W zestawie jest testowy silnik BLDC, opcjonalnie można zakupić także silnik indukcyjny. Stopień mocy może sterować silnikami BLDC lub indukcyjnymi o mocy do 1000 W i napięciu pracy od 12 do 300 V. Dołączone oprogramowanie zawiera środowisko programistyczne IDE oraz bardzo wygodny interfejs graficzny, umożliwiający programowanie parametrów w rejestrach konfiguracyjnych mikrokontrolera.

Jacek Przepiórkowski

PROGRAMOWALNE MODEMY GSM/GPRS



- GPRS klasy B (4+1)
- Stos TCP/IP we wszystkich modułach
- Programowanie w C
- Solidna konstrukcja
- Szeroki zakres temperatur roboczych
- Akcesoria: anteny, złącza, zasilacze, kable



Autoryzowany dystrybutor



Sony Ericsson

50-071 Wrocław, Plac Wolności 7
tel. (71) 783 12 60
fax (71) 783 12 69
info.poland@eurolis.com

41-200 Sosnowiec, ul. Kiepury 45/1
tel. (32) 291 99 35
fax (32) 296 90 80
info.poland@eurolis.com