

Modułowy, 32-bitowy sterownik pralki

Andrzej Milecki, Grzegorz Pittner

Politechnika Poznańska

Streszczenie: W artykule przedstawiono podstawowe elementy stosowane w pralkach domowych. Opisano krótko sposób ich działania oraz obsługi przez sterownik. W dalszej części przedstawiono koncepcję własnego sterownika pralki zbudowanego na bazie mikrokontrolera 32-bitowego z rodziny STM32 [1, 2, 3, 4]. Opisano jego budowę, a szczególnie uwagę zwrócono na komunikację oraz oprogramowanie.

Słowa kluczowe: mikrokontroler, pralka, sterownik pralki

1. Wprowadzenie

Od kilkunastu lat można zaobserwować w Polsce znaczący rozwój sektora produkującego sprzęt gospodarstwa domowego. Działający w latach 80-tych XX w. producenci krajowi zostali w większości przejęci przez największe koncerny światowe. Zwiększyły one liczbę produkowanych urządzeń znacząco je przy tym unowocześniając i podjęły także produkcję zupełnie nowych. Chodzi tutaj o takie firmy jak Indesit-Łódź, Whirlpool-Polar Wrocław, Fagor-MCC-Wrozamet Wrocław, Electrolux, a ostatnio także i Samsung Wronki. Dzięki temu Polska jest poważnym centrum produkcyjnym sprzętu AGD. Jak można oszacować, już obecnie w przemyśle produkującym sprzęt gospodarstwa domowego w Polsce zatrudnionych jest bezpośrednio kilkanaście tysięcy pracowników.

Mimo tak rozbudowanej produkcji sprzętu gospodarstwa domowego, w Polsce nie prowadzi się poważnych specjalistycznych badań dotyczących tego obszaru. Potenci posiadają centra badawcze zlokalizowane głównie we Włoszech, w Niemczech i w Korei. Widząc tę znaczącą lukę na Politechnice Poznańskiej podjęto prace rozwojowe dotyczące budowy sterowników pralek. W ich ramach zaprojektowano i wykonano 2 nowe sterowniki. Pierwszy, to układ „low cost”, który jest odpowiedzią na obecny kryzys, a drugi to rozbudowany sterownik, którego konfigurację można łatwo zmieniać wymieniając poszczególne moduły. W obu sterownikach zastosowano nowatorskie rozwiązania, w tym metody sztucznej inteligencji.

2. Obsługa elementów we/wy pralek

We współczesnych pralkach domowych stosowane są następujące główne elementy, obsługiwane przez sterownik pralki: – wejścia od czujników: temperatury i poziomu wody, wyłącznika aquastop, zamknięcia drzwi, prędkości wirowania (położenia wirnika), – wyjścia do: zaworów elektromagnetycznych (kilka szt.), grzałki, blokady drzwi, silnika i pompy.

Dodatkowo sterownik musi obsługiwać również interfejs użytkownika. Zadaniem sterownika pralki jest obsługa ww.

elementów poprzez odpowiednie układy, aby pralka wykonywała zadany i zapisany w pamięci program oraz by w każdej chwili urządzenie było bezpieczne dla użytkownika i otoczenia. Wymogi bezpieczeństwa dla sprzętu AGD wynikają z norm PN-EN 60335-1 i PN-EN 60335-2-7.

W ramach opisywanych w niniejszym artykule prac zaproponowano i zbudowano sterownik pralki bazujący na mikrokontrolerze 32-bitowym z rodziny STM32 [1, 2, 3, 4].

Do pomiaru temperatury wody oraz powietrza (w pralko-suszarkach) stosowane są termistory NTC10k. Ich nieliniowa charakterystyka jest linearyzowana programowo przez sterownik. Sygnał z termistorów odczytywany jest za pomocą 12-bitowego ADC.

Do pomiaru poziomu wody stosowany jest presostat. Element ten zawiera dwa kondensatory oraz cewkę, której indukcyjność zmienia się w zależności od poziomu wody. Współpracuje on zwykle z tranzystorami tworząc generator sygnału prostokątnego, którego częstotliwość zależy od poziomu wody.

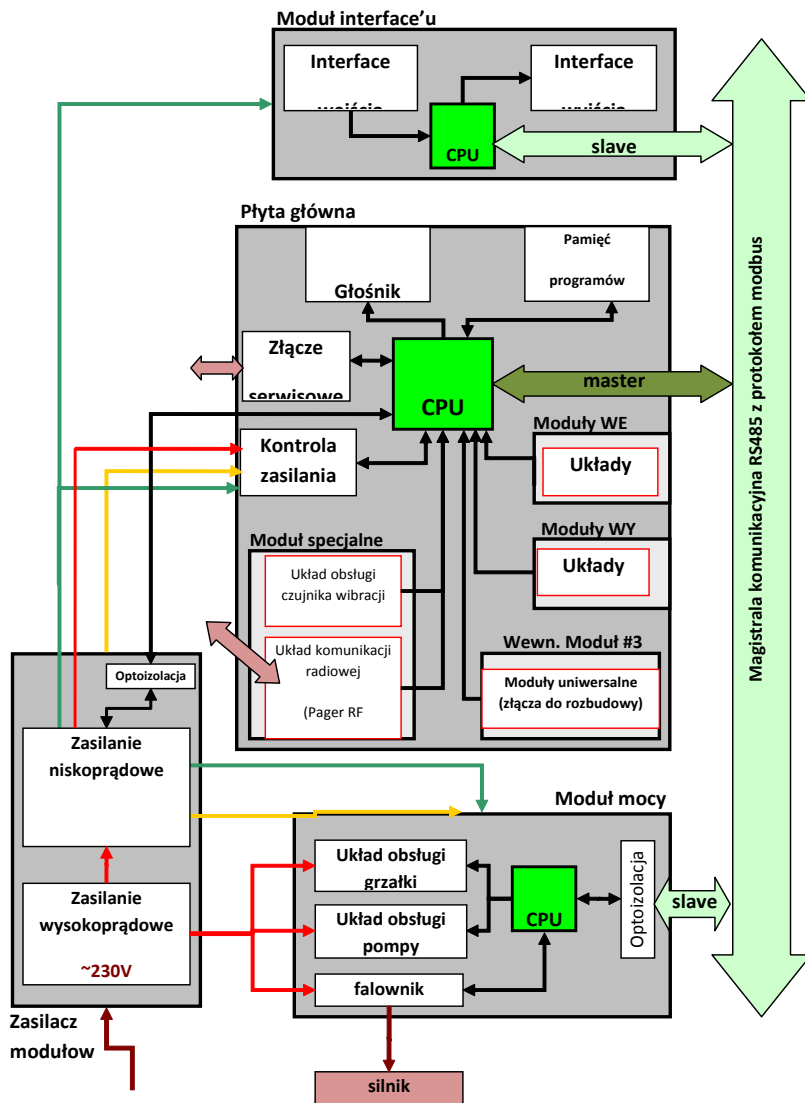
Pomiar prędkości wirowania bębna zależy od rodzaju zastosowanego silnika. Jeśli stosowany jest silnik komutatorowy, to najczęściej stosowana jest prądnica tachometryczna, synchroniczna. Coraz częściej jednak w pralkach stosuje się silniki bezszczotkowe, w których montowane są hallotrony przeznaczone do pomiaru położenia wirnika.

Drzwi pralki są zwykle blokowane i odblokowywane za pomocą rygla zasilanego napięciem 230 V AC. Sterowanie tym rygłem odbywa się poprzez dwa niezależne przekaźniki, jeden do zamykania, a drugi do otwierania blokady drzwi. Wykrywanie otwarcia drzwi odbywa się za pomocą czujnika zwiernego, zasilanego napięciem 15 V.

Do połączenia ze sterownikiem zastosowano optoizolację.

Zasilacz sieciowy powinien dostarczać napięcie: 3,3 V dla mikrokontrolera i pamięci programów CMOS, 5 V dla elementów we/wy sterownika w standardzie TTL oraz 15 V dla zasilania przekaźników itp. Zasilacz w stanie jałowym nie powinien pobierać więcej niż 30 mW. Aby spełnić powyższe wymagania konieczne jest zastosowanie przetwornic i częściowe bądź całkowite odejście od stabilizatorów napięcia. Dodatkowo, aby zasilacz spełniał wymogi bezpieczeństwa wynikające z normy musi on być wyposażony w zabezpieczenie przeciążeniowe na poziomie uznawanym przez jednostkę certyfikującą za bezpieczny zarówno pod kątem prądu działającego jak i czasu reakcji.

Większość zaworów w pralkach zasilana jest prądem przemiennym, dlatego też do sterowania nimi wykorzystuje się przekaźniki, tyrystory albo triaki. Przekazy z uwagi na cenę i trwałość, sporą wymaganą powierzchnię na PCB są rozwiązaniem nie najlepszym. Wadą tyrystorów natomiast jest mała trwałość w sytuacji zwarcia w zaworze. Najlepszym rozwiązaniem wydaje się być zastosowanie triaków.



Rys. 1. Schemat blokowy sterownika pralki
 Fig. 1. Washing machine block scheme diagram

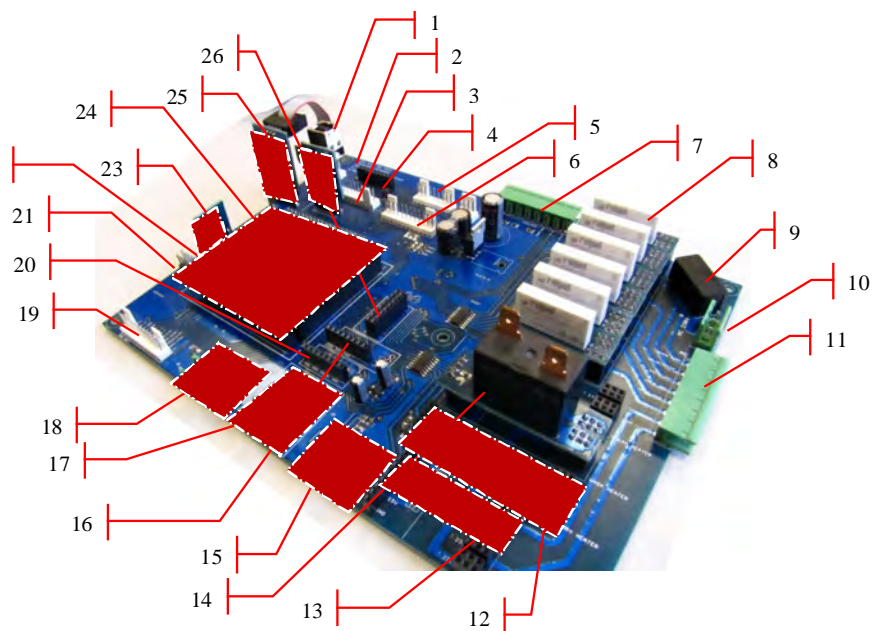
(tutaj falownikiem) oraz układ sterowania pompą.

Na podstawie tego schematu wykonano projekt całego sterownika, następnie zaprojektowano płytki PCB, zmontowano i uruchomiono. Widok wykonanej płytki przedstawiono na rys. 2. Sterownik zawiera następujące moduły: 1 – moduł komunikacji USB (do podłączenia komputera PC dla celów diagnostycznych albo programowania sterownika pralki), 2 – płyta główna sterownika pralki, 3 – złącze modułu interfejsu użytkownika (SPI z dodatkowymi sygnałami), 4 – złącze modułu komunikacji bezprzewodowej (złącze do komunikacji radiowej, np. do pagera), 5 – złącze uniwersalne do modułu (umożliwia komunikację z innymi sterownikami), 6 – złącze do modułu analizy wody, 7 – złącze zasilacza dedykowanego, 8 – moduł wyjścia średniej mocy (przełącznik do załączenia elementów pralki, np. blokady drzwi, elektrozaworów itp.), 9 – gniazdo bezpiecznika ~230V, 10 – złącze zasilania ~230V, 11 – złącze wyjść peryferyjnych (wyprowadzenia przełączników), 12 i 14 – gniazda do wpięcia dodatkowych przełączników dużej mocy, 13 – gniazda do wpięcia dodatkowych przełączników średniej mocy, 15 – układ obsługi sensora zamknięcia drzwi, 16 – złącze do modułu pomiaru wagi wsadu, 17 – układ obsługi presostatu, 18 – układ obsługi termistorów, 19 – złącze komunikacji równoległej z modułem mocy (SPI z kilkoma dodatkowymi sygnałami), 20 – złącze modułu pomiaru prędkości obrotowej, 21 – złącze programatora JTAG (do programowania CPU), 22 – układ zegara RTC, 23 – moduł pamięci EPROM, 24 – mikro-

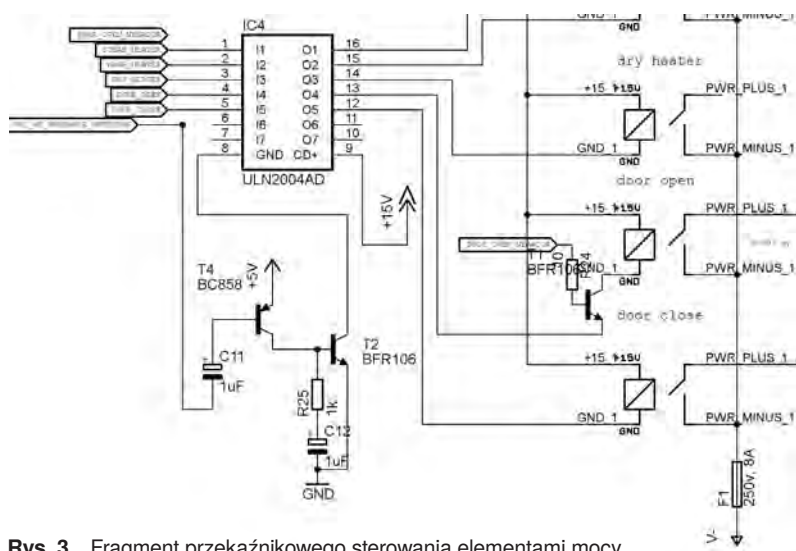
3. Budowa sterownika

Na rys. 1 pokazano schemat blokowy 32-bitowego typu STM32 sterownika pralki. Aby uprościć budowę sterownika i umożliwić jego łatwą rozbudowę zastosowano magistrale szeregowo do komunikacji z modułami pralki. W sterowniku można wyróżnić bloki funkcjonalne:

- płyta główna, na której znajdują się moduły: CPU, obsługi sensorów, moduł komunikacji USB, moduł komunikacji RS-485, moduł pamięci EEPROM, moduł brzęczyka,
- moduł interfejsu użytkownika (mogą wtedy być stosowane różne moduły zaczynając od przełącznikowych a kończąc na ekranach dotykowych),
- moduł mocy zawierający układ załączania grzałki, sterowania silnikiem



Rys. 2. Widok płytki głównej PCB sterownika
 Fig. 2. View of the controller main board



Rys. 3. Fragment przekąźnikowego sterowania elementami mocy
Fig. 3. Fragment of relay based power control

kontroler STM32, 25 – moduł komunikacji szeregowe RS-485, 26 – złącze modułu pomiaru drgań bębna.

Najważniejszą częścią sterownika jest moduł mikrokontrolera (24). Jego centralną część stanowi jednostka CPU, którą zbudowano na bazie 32-bitowego mikrokontrolera typu STM32F103ZET6. Należy on do rodziny 32-bitowych mikroprocesorów STM32. Zbudowany jest na bazie rdzenia CortexM3. Mikrokontroler wyposażony jest w wewnętrzne szybkie magistrale pozwalające na wydajną wymianę danych, adresów i sygnałów sterujących. Magistrale taktowane są odpowiednio częstotliwością:

72 MHz oraz 36 MHz. Najważniejsze cechy mikrokontrolera to:

- układ SMD, 144 wyprowadzenia, 512 KB pamięci FLASH, do 64 KB pamięci SRAM,
- napięcie zasilające: 2V to 3,6V,
- interfejsy: 3 × USART, 2 × UART, 3 × SPI, (2 × I2S), 2 × I2C, USB 2.0,
- maksymalnie 112 wejść/wyjść binarnych,
- 8 układów czasowo-licznikowych w tym 4 liczniki 16-bitowe, generator sygnału PWM (16 kanałów),
- przetworniki: 3 × 12-bit, 1 s A/D (do 21 kanałów), 2 × 12-bit D/A, DMA (12 kanałów),
- tryby oszczędzania energii: Steep, Stop i Stanby,
- podtrzymanie baterijne zegara RTC i rejestrów.

Moduł mikrokontrolera (24) jest podłączony za pośrednictwem złącza, dzięki czemu można go łatwo wymienić na inny. Pozwala to na dość swobodne kształtowanie mocy obliczeniowej sterownika. Możliwe jest umieszczenie w gnieździe także mikroprocesora 8-bitowego. Omawiany moduł zawiera wszystkie układy potrzebne do pracy mikrokontrolera, takie jak: układ zegarowy, układ podtrzymania baterijnego, układ RTC, filtry zasilania, filtry indukcyjne przetwornika analogowo-cyfrowego, interfejs programowania oraz debugger'a.

W sterowniku zaimplementowane są dwie magistrale szeregowe: SPI oraz RS-485. Pierwsza z nich służy do wykonywania wszelkich prac uruchomieniowych, do testowania oraz do komunikacji z pamięcią EPROM. Wyprowadzono ją także na złącza, co umożliwia dołączenie dodatkowych, nie zaplanowanych na obecnym etapie urządzeń. Druga magistrala przeznac-

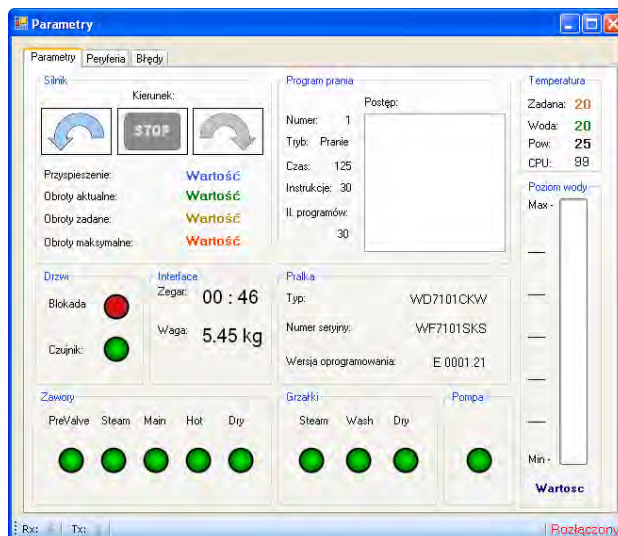
zona jest do komunikacji z zewnętrznymi modułami sterownika pralki. Chodzi tutaj w szczególności o komunikację z modulem dużej mocy oraz z interfejsem użytkownika (panelem pralki). Moduł dużej mocy odpowiedzialny jest za sterowanie silnikiem pralki, włączaniem i wyłączaniem grzałki oraz pompy wody. Modułowa konstrukcja umożliwia podłączenie do sterownika dowolnego zespołu napędowego. Może to więc być sterownik silnika komutatorowego albo sterownik nowoczesnego silnika bezszczotkowego (*direct drive*). Właśnie ten ostatni silnik jest zastosowany w pralce, do której opracowano nowy sterownik. Zastosowanie magistrali RS-485 pozwala na dołączenie w zasadzie dowolnego interfejsu użytkownika, począwszy od powszechnie znanych z pokręteł wyboru programu, kilkoma przełącznikami oraz wyświetlaczem, a skończywszy na inter-

terfejsie z kolorowym wyświetlaczem LCD i rezystancyjnym panelem dotykowym zamontowanym bezpośrednio na powierzchni wyświetlacza.

4. Oprogramowanie i uruchomienie sterownika

Zasadniczy program sterownika napisano w języku C++. Jest on zapisany w pamięci wewnętrznej FLASH mikrokontrolera. Programy prania zapisane są w pamięci EPROM. Stanowi ona zewnętrzny moduł, który komunikuje się z procesorem za pośrednictwem magistrali SPI. Dzięki temu można bardzo łatwo ją wymienić, co pozwala na szybką wymianę programów prania. W pierwszym etapie stworzono oprogramowanie robocze (*firmware*) sterownika. Jest to program dla nadzorowania modułów wewnętrznych mikrokontrolera. Pozwala na obsługę:

- komunikacji programatora z CPU,
- konfiguracji i testowanie modułów wewnętrznych CPU,
- zegarów, przetworników AD I DA,



Rys. 4. Widok ekranu pokazującego aktualny stan pracy pralki
Fig. 4. View of the window with washing machine parameters

- modułów PWM, DMA, NVIC, EXTI,
- magistral: RS485, UART, USB, SPI.

W drugim kroku opracowano oprogramowanie do obsługi modułów zewnętrznych mikrokontrolera, takich jak:

- pomiar temperatury wody,
- pomiar poziomu wody,
- moduł brzęczyka,
- zaworów, pompy, blokady drzwi.

Z tego poziomu możliwe jest także:

- skalowanie ADC i wykonywanie testów DMA,
- konfigurowanie przerwań EXTI, NVIC,
- testowanie priorytetów przerwań,
- obsługiwanie interfejsów komunikacyjnych,
- zarządzanie pamięcią EEPROM,
- przez USB,
- komunikacja ze sterownikiem mocy.

W dalszej kolejności stworzono egzekutor komend programu prania, który umożliwia obsługę instrukcji niskiego i wysokiego poziomu oraz obsługę:

- wykonywania pętli i instrukcji warunkowych,
- wykonywania wzorów matematycznych,
- wykrywania błędów w programie prania,
- wykrywanie błędów egzekucji programu,

W programie przewidziano także obsługę trybu serwisowego pralki, który pozwala na testowanie egzekutora i obsługę testów serwisowych podzespołów pralki i urządzeń peryferyjnych sterownika pralki.

Wykonano liczne testy i sprawdzenie poprawności działania oprogramowania i sterownika podczas symulacji błędów pralki. Napisano także program na komputer klasy PC z interfejsem graficznym MS Windows, przeznaczony dla serwisantów (rys. 4). Program oferuje 2 tryby pracy oraz obsługę błędów sterownika głównego. Pierwszy tryb pracy to DEBUG (Parametry), w którym użytkownik ma możliwość podglądu pracy sterownika i stanu poszczególnych urządzeń. Tryb DEBUG nie wpływa na pracę sterownika pralki oraz na wykonywany program prania. Jest to tryb domyślny, ustawiany po uruchomieniu programu. Okno „Parametry” zawiera następujące informacje o stanie urządzeń i wykonywanym programie:

- Silnik – pokazuje informacje o silniku takie jak: kierunek, wartości przyspieszenia, obroty zadane: aktualne i maksymalne;
- Program prania – zawiera informacje o aktualnym programie prania i jego postępie;
- Temperatura – pokazuje wartość temperatur: zadanej wody, aktualnej powietrza i wody oraz temperatury CPU sterownika;
- Interface – pokazuje uproszczone informacje przesyłane do interfejsu pralki;
- Pralka – informacje o oprogramowaniu i rodzaju pralki,
- Poziom wody – pokazuje aktualny stan poziomu wody w słupku;
- Drzwi, Zawory, Grzałki, Pompa – diody LED pokazują aktualny stan danego elementu (diody zielona – załączony, dioda czerwona – wyłączony).

Drugi tryb SERWIS (Peryferia) służy do obsługi urządzeń pralki dołączonych do sterownika oraz do przeprowadzania testów. Tryb ten wpływa na pracę sterownika, np. wymusza zatrzymanie i zakończenie danego programu

prania. Obsługa błędów (Błędy) pozwala na odczyt błędów zapisanych w pamięci EEPROM oraz na ich skasowanie. Odczytane błędy można zapisać do pliku tekstowego albo odczytać z pliku.

Wykonano także program do tworzenia programów prania, w którym opracowano specjalizowany interfejs programowania programów prania, egzekutor i walidator komend prania.

Zbudowany sterownik podłączono do pralki (rys. 5.) i przeprowadzono jego testy uruchomieniowe. W ramach testów końcowych wykonano następujące czynności:

- o podłączenie sterownika do pralki,
- o napisanie programu piorącego wykorzystującego wszystkie po kolei elementy peryferyjne,
- o uruchomienie dowolnego programu i sprawdzenie czy zakończył się sukcesem,
- o uruchomienie dowolnego programu raz jeszcze z testem regulatora temperatury,
- o uruchomienie dowolnego programu raz jeszcze z testem regulatora poziomu wody,
- o uruchomienie dowolnego programu raz jeszcze i odłączenie zasilania w trakcie trwania programu,
- o uruchomienie dowolnego programu raz jeszcze i zakończenie prania przed czasem („przerwij”),
- o uruchomienie dowolnego programu raz jeszcze i wstrzymanie programu aby dodać pranie,
- o uruchomienie dowolnego programu raz jeszcze z symulacją błędu zaworu wlewowego,
- o uruchomienie dowolnego programu raz jeszcze z symulacją błędu pompy,
- o uruchomienie dowolnego programu raz jeszcze z symulacją błędu termistora (brak termistora),
- o uruchomienie programu raz jeszcze z symulacją błędu termistora (zwarcie),



Rys. 5. Widok pralki z podłączonymi do niej sterownikami: głównym i dużej mocy (falownik)

Fig. 5. View washing machine and controllers connected to it: main controller and high power controllers

- o uruchomienie programu raz jeszcze z symulacją błędu grzałki,
- o uruchomienie programu raz jeszcze z odłączonym modułem mocy.

Przetestowano i modyfikowano także wymianę danych pomiędzy sterownikiem centralnym a modułami zewnętrznymi za pośrednictwem magistral:

- RS-485 – komunikacja z odrębnymi płytkami PCB (interfejs użytkownika, moduł mocy),
- SPI – komunikacja z zaawansowanymi układami cyfrowymi w obrębie PCB sterownika centralnego zawierającego CPU (Bluetooth, EEPROM, nadajnik pagera).

Sprawdzono również poprawność komunikacji sygnałów:

- analogowych i quasi-cyfrowych (przebiegi prostokątne dla czujników i modułu wibracji),
- cyfrowych 0/1 dla większości układów wejść i wyjść,
- na magistrali USB (do komunikacji CPU sterownika pralki z PC).

W chwili obecnej sterownik jest gotowy do integracji całego oprogramowania i tworzenia nowych rozwiązań.

5. Podsumowanie

W artykule przedstawiono konstrukcję nowatorskiego sterownika pralki zbudowanego na bazie nowoczesnego mikrokontrolera 32-bitowego z rodziny STM32. Mikrokontrolery 32-bitowe stają się coraz tańsze, a oferowana przez nie moc obliczeniowa, w porównaniu z mikrokontrolerami 8-bitowymi jest nieporównywalnie większa. Daje to olbrzymie możliwości konstruktorom i programistom pralek. W zaprojektowanej i wykonanej konstrukcji sterownika szczególną uwagę zwrócono na:

- rozdzielenie najważniejszych funkcji na osobne moduły, tak, aby można było stosunkowo łatwo konfigurować sterownik, stosowanie do potrzeb i ceny pralki (tzw. „półki”),
- oddzielenie (odseparowanie za pomocą optoizolacji) elementów dużej mocy od sterownika, tak aby funkcje niebezpieczne i wymagające długiej i kosztownej certyfikacji „zamknięte” zostały w odrębnej obudowie,
- zapewnienie możliwości bardzo łatwej wymiany interfejsu użytkownika, dzięki podłączeniu go za pomocą interfejsu uniwersalnego (interfejs pralki jest najczęściej zmieniany przez marketing),
- zaimplementowanie „na stałe” funkcji podstawowych, np. obsługi termistorów i presostatu,
- zaimplementowanie modułów wyjściowych średniej mocy w formie elementów wymiennych (przełączniki wymiennane) gdyż to one uszkadzają się najczęściej,
- zaimplementowanie funkcji specjalnych w formie modułów wymiennych (opcji),
- zaimplementowanie CPU w formie modułu wymiennego, co pozwala na zmianę mikrokontrolera,
- zbudowanie zasilacza jako osobnego urządzenia (łatwość certyfikacji i możliwość szybkiego dostosowania do potrzeb – konfiguracji).

W ramach dalszych prac rozwojowych przewiduje się:

- wykonanie typoszeregu innowacyjnych interfejsów użytkownika komunikujących się przy wykorzystaniu nie stosowanych dotychczas metod i zasad,

- wykonanie modułu symulującego wibracje bębna pralki i przeprowadzenie badań nad możliwością ich kompensacji i tłumienia,
- wykonanie moduły do komunikacji bezprzewodowej RF dla pagera,
- wykonanie modułu komunikacji bezprzewodowej Bluetooth dla podłączenia komputera klasy PC i telefonu komórkowego
- wykonanie nowej wersji sterownika zawierającej poprawione błędy wykryte w czasie testów obecnej, pierwszej wersji prototypowej.

Praca naukowa finansowana ze środków na naukę ze środków Narodowego Centrum Badań i Rozwoju w latach 2010-2012 w ramach projektu rozwojowego pt.: „Inteligentne sterowniki pralek”.

Bibliografia

1. Paprocki K., Mikrokontrolery STM32 w praktyce, btc, 2009.
2. Peczarski M., Mikrokontrolery STM32 w sieci Ethernet w przykładach, btc, 2011.
3. <http://www.st.com/internet/mcu/class/1734.jsp>
4. <http://www.stm32.eu/> ■

Modular, 32-bit washing machine controller

Abstract: In the paper the basic elements used in house washing machines are presented. The principle of their operation are shortly described. In the next part of the paper authors own conception of new washing machine controller construction is presented. The proposed controller is based on 32-bit STM32 microcontroller. The controller design was described. A special attention was given to the worked out software.

Keywords: microcontroller, washing machine controller

prof. dr hab. inż. Andrzej Milecki

Profesor zwyczajny w Instytucie Technologii Mechanicznej, Wydział Budowy Maszyn i Zarządzania Politechniki Poznańskiej. Kierownik Zakładu Urządzeń Mechatronicznych. Specjalność naukowa: budowa i eksploatacja maszyn, mechatronika, elektrohydraulika, automatyzacja maszyn i urządzeń.

e-mail: andrzej.milecki@put.poznan.pl



mgr inż. Grzegorz Pittner

Asystent w Instytucie Technologii Mechanicznej, Wydział Budowy Maszyn i Zarządzania Politechniki Poznańskiej. Zainteresowania naukowe: mechatronika, sterowniki mikroprocesorowe, automatyzacja maszyn i urządzeń.

e-mail: grzegorz.pittner@put.poznan.pl

