

doi:10.15199/48.2020.07.17

# Modularny system napędowy dla silników bezszczotkowych z interfejsem użytkownika opartym na rozwiązaniach Open Source

**Streszczenie.** Artykuł przedstawia modularny system napędowy przeznaczony dla silników bezszczotkowych z przykładowo zaimplementowanym sterowaniem polowo-zorientowanym i komunikacją USB z komputerem PC. Projekt interfejsu graficznego opartego na darmowym środowisku Scilab-Xcos służy do testowania sterowania silnikami i może być alternatywą dla tworzenia własnego oprogramowania interfejsu graficznego, który zwykle jest dostarczany z rozwiązaniami komercyjnymi. Oprogramowanie było testowane zarówno w środowisku Windows, jak i Linux. Badania dotyczyły regularności odmierzenia czasu oraz opóźnień komunikacyjnych, a także ogólnej funkcjonalności interfejsu do konfiguracji układu sterowania i możliwości wizualizacji przebiegów zmiennych. Pokazano możliwość strojenia regulatorów oraz podglądu pracy układu w różnych trybach funkcjonalności typowe dla sterowników komercyjnych.

**Abstract.** This paper presents modular drive system for brushless motors with Field Oriented Control algorithm and USB communication implemented. Graphical user interface based on Open Source Scilab-Xcos was used for configuration and testing purposes, and could be an alternative for custom software usually provided with commercial controllers. Developed software was tested on both Windows and Linux operating systems. We have verified time base and delays for various configurations, as well as general functionality of user interface. Configuration of the controllers and data visualization is very intuitive and fully resembles options available in commercial products. (**Modular drive system for brushless motors with user interface based on Open Source solutions**).

**Słowa kluczowe:** otwarte oprogramowanie, robotyka mobilna, interfejs graficzny, komunikacja USB

**Keywords:** open source software, mobile robotics, graphical interface, USB communication.

## Wstęp

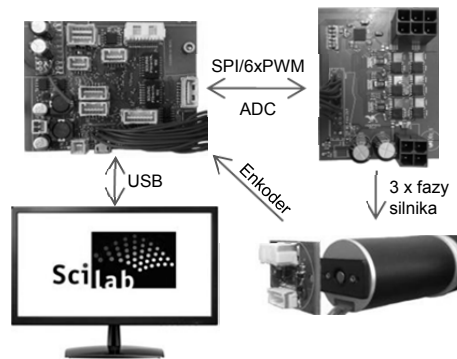
Prezentowany projekt dotyczy przede wszystkim rozbudowy możliwości technicznych inspekcyjnego robota mobilnego Raptors, jednak uniwersalność stworzonych rozwiązań pozwala na ich wykorzystanie w dowolnych średniej wielkości robotach mobilnych oraz innych napędach elektrycznych pracujących w trybie prądowym, prędkościowym lub pozycyjnym. Zaproponowane rozwiązanie ma dwie zasadnicze zalety: modularną strukturę sprzętową i otwarty interfejs użytkownika. Trzy układy elektroniczne (patrz Rys. 1) pozwalają na elastyczność konfiguracji w takim sensie, że ta sama jednostka sterująca może współpracować z różnymi układami mocy oraz wykorzystywać odpowiednie sprzężenie zwrotne (prądowe, prędkościowe lub pozycyjne). Wówczas, dla zastosowań robotyki mobilnej, gdzie w jednym pojeździe znajdziemy napędy różnej wielkości i pracujące w różnych trybach, jedynym komponentem skalowanym jest układ mocy. Użycie silników bezszczotkowych oraz metody sterowania polowo-zorientowanego (ang. FOC) daje najlepsze rezultaty jeżeli chodzi o sprawność układu napędowego w szerokim zakresie parametrów pracy oraz zapewnia wysoką dynamikę regulacji momentu obrotowego i prędkości [1], co w robotyce mobilnej zwłaszcza przy zasilaniu akumulatorowym jest kluczowym aspektem. Graficzny interfejs użytkownika (GUI) w przypadku systemów napędowych do zabudowy (ang. embedded) służy głównie celom testowym i konfiguracyjnym, czyli do podglądu danych oraz zadawania wartości parametrów sterujących. Użycie w tym celu otwartego (ang. open source) środowiska Scilab-Xcos obniża koszty tworzenia aplikacji i daje praktycznie nieograniczone możliwości rozbudowy. Rozwiązania typu open source i open hardware stają się coraz bardziej popularne nie tylko w zastosowaniach amatorskich ale stają się produktami komercyjnymi. Niniejsza propozycja wpisuje się przynajmniej częściowo w ten nurt. W dalszej części artykułu przedstawione zostaną szczegóły budowy układów, oprogramowania sterującego i komunikacyjnego oraz testy systemu przeprowadzone za pomocą stworzonego interfejsu użytkownika.

## Opis sprzętu użytego na stanowisku badawczym

Do testów algorytmu sterowania polowo-zorientowanego wykorzystano silnik BLDC **EC4070S-2411** firmy **Vishan Motors** o mocy 70 W. Jest to silnik o jednej parze biegunów i trzech fazach, maksymalnej nominalnej prędkości 11000 rpm i napięciu zasilania 24 V. Dodatkowo silnik ma wbudowane czujniki Halla.

Główny moduł sterujący pracą silnika BLDC bazuje na układzie **XMC4800** firmy Infineon [2]. Układ posiada rdzeń **ARM cortex-M4**, zaś maksymalna częstotliwość taktowania wynosi 144 MHz. Oprogramowanie dla tego modułu zostało napisane w języku C w środowisku programistycznym **DAVE IDE**, które jest dedykowane dla programowania mikrokontrolerów firmy Infineon. Moduł posiada łącze **USB** do komunikacji z zewnętrznymi urządzeniami. Dzięki zaimplementowanemu protokołowi komunikacyjnemu mikrokontroler może łączyć się z warstwą graficzną **Scilab-Xcos**, która znajduje się na komputerze PC. Moduł mocy sterujący pracą silnika bezszczotkowego bazuje na układzie **DRV8323** [3] (Texas Instruments), który jest zintegrowanym sterownikiem tranzystorów MOSFET, dedykowanym dla aplikacji sterowania trójfazowego. Układ scalony **DRV8323** jest przeznaczony do aplikacji, w których napięcie zasilania nie przekracza 60 V. Sterowanie poszczególnymi bramkami tranzystorowymi odbywa się poprzez odpowiednie generowanie sygnałów PWM, które pochodzą z głównego modułu sterującego.

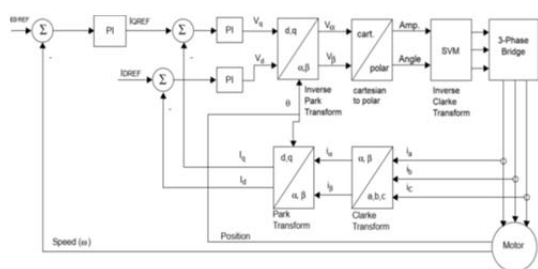
Układ enkoderowy został oparty na module **AS5147** [4] firmy AMS. Jest to bezstykowy magnetyczny enkoder dokonujący pomiaru położenia kąтового w zakresie od 0 do 360 stopni. Absolutna pozycja kątowna zorientowania pola magnetycznego magnesu jest mierzona z rozdzielczością 14 - bitów, co odpowiada kątowi około 0,022°. Wartość cyfrowa może być odczytana poprzez interfejs **SPI**. Dodatkowo układ posiada wyjścia **A**, **B** oraz **Index**, które realizują funkcję enkodera kwadraturowego. Rozdzielczość z wykorzystaniem wyjść kwadraturowych wynosi 11 - bitów, czyli około 0,176° na krok. Sygnały cyfrowe **A**, **B** oraz **Index** są podłączone do dedykowanych wejść mikrokontrolera **XMC4800**. Poprzez odpowiednie przeliczenia w układzie **XMC4800**, pozyskiwane są informacje o aktualnym położeniu wału silnika oraz prędkości obrotowej.



Rys.1. Struktura modularnego systemu napędowego

### Sterowanie polowo-zorientowane silnikiem BLDC

Badany silnik bezszczotkowy, choć nominalnie jest silnikiem BLDC, posiada praktycznie sinusoidalne przebiegi siły elektromotorycznej (*ang. B-EMF*), dlatego metoda sterowania polowo-zorientowanego nie będzie miała złego wpływu na jakość sterowania i w tym przypadku może być zastosowana. Metoda sterowania polowo-zorientowanego (*ang. Field Oriented Control*) jest jedną z metod sterowania trójfazowymi silnikami prądu przemiennego. Zaletą tej metody jest dokładne sterowanie wewnętrznym momentem obrotowym silnika, co pozwala na zapewnienie bardzo dobrych właściwości dynamicznych całego napędu. Taki algorytm sterowania polega na niezależnym od siebie sterowaniu dwiema prostopadłymi do siebie składowymi wektora przestrzennego prądu stojana: jedną odpowiedzialną za wytwarzanie momentu wewnętrznego (elektromagnetycznego) w silniku, i drugą odpowiedzialną za wytworzenie dodatkowego strumienia magnetycznego do-wzbudzającego lub od-wzbudzającego silnik. Dzięki temu możliwe jest uzyskanie wysokiej dynamiki działania napędu zarówno przy odtwarzaniu zmian wartości zadanej prędkości obrotowej wału silnika, jak i przy kompensowaniu raptownych zmian momentu obciążenia występującego na tym wale. Wymagania dynamiczne dla układu napędowego zostały spełnione pomimo bardzo prostej struktury bloków regulatorów odpowiadających za regulację prądów.



Rys.2. Schemat układu regulacji [6]

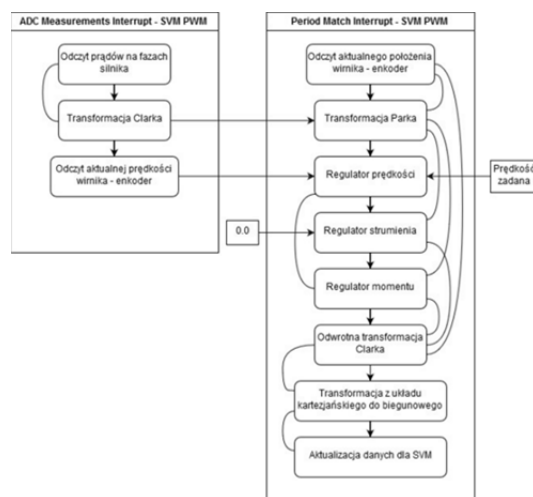
Na schemacie (Rys. 3) przedstawiono procedury, które wykonywane są strukturalnie podczas występowania przerwań od generatora sygnałów SVM PWM. Procedury wykonywane są od góry do dołu, zgodnie ze umieszczonymi strzałkami (*kolor czarny*). Niebieskie linie przedstawiają przepływ informacji z danej procedury do innej. Prostokąty poza obrysem funkcji oznaczają wartości zadane dla zaimplementowanych regulatorów.

### Interfejs użytkownika z użyciem oprogramowania Scilab-Xcos

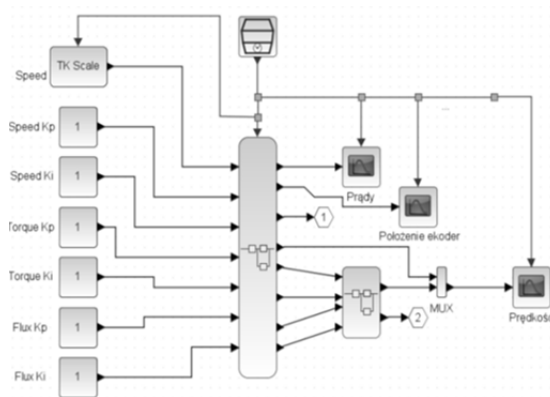
Interfejs graficzny do testowania modułu sterowania silnikiem BLDC został stworzony za pomocą bloków dostępnych z poziomu **Palette browser** oprogramowania Scilab-Xcos. Zadaniem interfejsu jest zadawanie wartości

sterowania, które ma realizować silnik i parametrów konfiguracyjnych dla układu regulacji, oraz podgląd zmiennych odebranych z modułu sterowania przez interfejs USB, co schematycznie pokazano na Rys. 4.

Zadawanie parametrów konfiguracyjnych dla układu sterowania polowo-zorientowanego odbywa się poprzez wpisanie współczynników do bloków **CONST\_m** - obok każdego bloku została umieszczona etykieta informująca, który parametr jest konfigurowany. Z kolei wyświetlanie zmiennych uzyskujemy za pomocą bloków **CSCOPE** - pod każdym blokiem umieszczona została etykieta określająca rodzaj wyświetlanych danych. Efekty działania interfejsu pokazano w rozdziale VI dotyczącym testów układu sterownia.



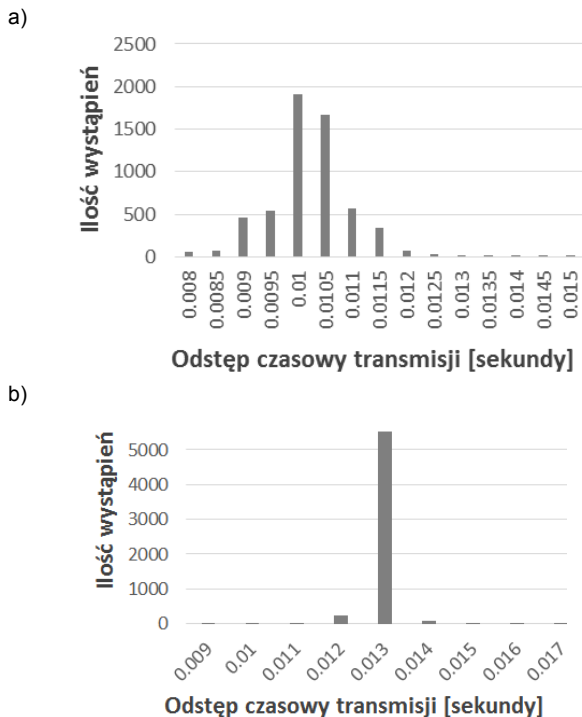
Rys.3. Schemat przedstawiający procedury wykonywane w programie dla XMC4800



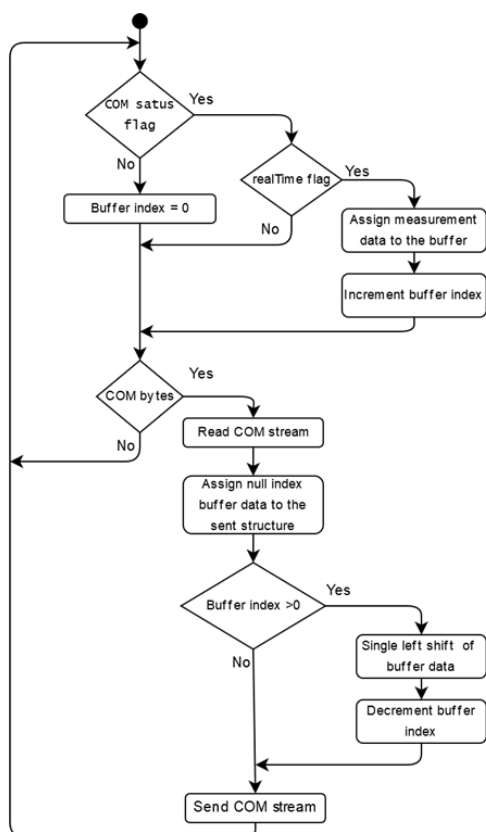
Rys.4. Widok aplikacji Scilab-Xcos z podłączonymi parametrami wejściowymi i wyjściowymi, po lewej stronie *widoczne bloki CONST\_m* pełniące funkcje zadawania wartości stałych, po środku blok komunikacji USB, po prawej bloki **CSCOPE** pełniące funkcje wykresów dla danych z układu sterownia.

### Testy komunikacji USB

Wymiana informacji między układem sterowania (mikrokontrolerem XMC4800) a interfejsem użytkownika (komputer PC) odbywa się za pomocą interfejsu USB, w którym węzłem zarządzającym jest PC. Powoduje to, że nie ma pewności (przynajmniej w systemach operacyjnych Windows lub Linux) czy dane są przesyłane w stałych interwałach czasowych. Przy nominalnej wartości okresu komunikacji 10 ms rozrzut ten jest na poziomie  $\pm 20\%$  (w systemie Windows 10), zaś w systemie Linux odstęp czasowy jest bardziej stabilny choć ma wartość większą od nominalnej o 30%, co pokazano na Rys. 5. Przy krótszych okresach rozrzut dochodzi do kilkuset procent [7].



Rys.5. Wyznaczone histogramy czasu transmisji (przy ustalonym czasie wynoszącym 10 ms, obejmującym 6 tys. próbek): a) dla systemu operacyjnego Windows, b) dla systemu operacyjnego Linux

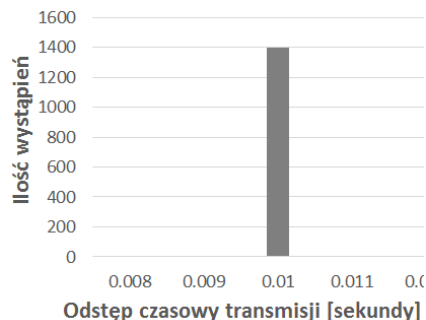


Rys.6. Schemat przedstawiający procedury dla komunikacji USB

Z punktu widzenia testów aplikacji i obserwowania wykresów online, czasy mogą być nieznacznie rozszynchronizowane a różnice w przebiegach (zmiany dynamiczne) pozostają niezauważalne. Ponieważ jednak sterowanie musi odbywać się z czasem rzeczywistym, dane

pojawiające się z określoną stałą częstotliwością należy odpowiednio buforować w urządzeniu, a następnie przesyłać do Scilab-Xcos przez USB.

W celu wyeliminowania błędu związanego z rozszynchronizowaniem czasowym między środowiskiem Scilab-Xcos a modulem sterowania, opracowane zostały procedury (Rys. 6), które zbierają dane do bufora z określonym interwałem czasowym. Dane z bufora wraz z czasem pomiaru są sukcesywnie wysyłane do Scilab-Xcos. Czas pomiaru został ustawiony na 10 ms i jest utrzymywany na tym poziomie przez mikrokontroler (Rys. 7).



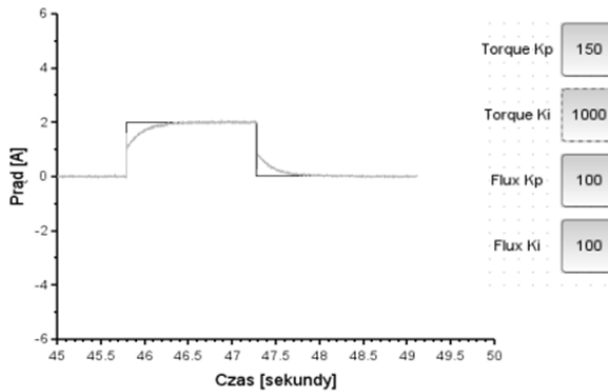
Rys.7. Histogram czasu transmisji (ustalony czas 10 ms, 1400 próbek), czas odmierzanym w układzie XMC4800

### Testy układu napędowego z zastosowaniem interfejsu użytkownika

Testy urządzenia zostały podzielone na trzy zasadnicze etapy: badanie pętli regulacji prądu, badanie pełnego regulatora kaskadowego oraz sprawdzenie wpływu zakłócenia na jakość regulacji. W pierwszym etapie, aplikacja testowa została zmodyfikowana, tak by pominąć regulator prędkości. W ten sposób sterowanie ograniczyło się do regulacji momentu obrotowego i zadawania prądów fazowych dla silnika BLDC. Funkcjonalność sterowania prądem jest tu kluczowym aspektem, bez tego nie ma możliwości sterowania kaskadowego, czyli zamknięcia drugiej pętli sprzężenia zwrotnego z regulatora prędkości. Drugi etap testów zakładał przetestowanie sterowania prędkością obrotową wału silnika. W ostatnim etapie przeprowadzono testy odporności układu na wprowadzone zakłócenie w postaci momentu hamującego. W badanych przypadkach  $K_p$  odnosi się do wzmocnienia części proporcjonalnej, natomiast  $K_i$  jest odwrotnością czasu zdwojenia  $T_i$ , gdzie czas próbkowania  $T_p$  wynosi 500  $\mu$ s.

- Badanie wpływu zmian parametrów regulatora prądu w stanach przejściowych

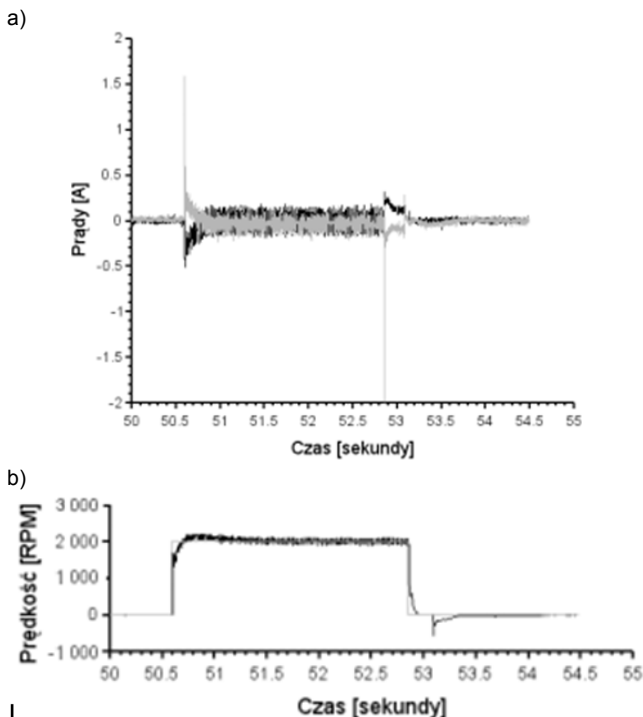
Istotnym, pierwszym etapem testowania stworzonej aplikacji jest obserwacja zachowań dynamicznych napędu, które w układzie sterowania z regulatorami zastosowanymi do niezależnej regulacji obu składowych wektora prądu stojana bardzo silnie zależą od ich typu i nastaw. Dlatego zostały przeprowadzone testy wpływu zmian tych parametrów na przebiegi wyjściowe obiektu regulacji, w tym przypadku na prąd odpowiedzialny za moment obrotowy. W układzie regulacji składowych prądu zastosowano regulatory typu PI, ponieważ muszą one kompensować gwałtowne zmiany SEM indukowanej w uzwojeniach stojana silnika przy zmianach jego prędkości obrotowej. Człony całkujące (astatyczne) w tych regulatorach pozwalają nie tylko na uzyskanie zerowego uchybu regulacji prądów w stanach ustalonych, ale także na ograniczenie jego wartości wywołanej liniową zmianą w czasie prędkości obrotowej silnika [5, 8]. Zmiany parametrów  $K_p$  oraz  $K_i$  dotyczyły głównie regulatora prądu  $I_q$ , ponieważ to on jest odpowiedzialny za generowanie momentu obrotowego na wale silnika. Wartość zadana wynosi 2 A, zaś przykładowy przebieg odpowiedzi skokowej pokazano na Rys. 8.



Rys. 8. Odpowiedź układu regulacji na skokowe wymuszenie 2 A - przebieg szary, wartość zadana - przebieg czarny.

- Badanie wpływu zmian parametrów regulatora prędkości w stanach przejściowych

Testy parametrów dla regulatora prędkości, tak jak w przypadku testów regulatora prądu mają zapewnić sprawdzenie technicznych możliwości zmiany nastaw i obserwacji ich wpływu na przebiegi, oraz zgrubne dobranie nastaw dla regulatora prędkości, aby zachować wysoką dynamikę na zmiany wartości zadanej. Tak jak w poprzednim podpunkcie modyfikacjom ulegały parametry wzmocnienia dla poszczególnych członów regulatora prędkości, czyli odpowiednio współczynniki Kp oraz Ki. Silnik w testach pracował bez obciążenia. Skokowa zmiana wartości zadanej prędkości wynosiła 2000 obr/min. Przykładowe przebiegi składowych prądu stojana i prędkości obrotowej wału silnika zaobserwowane w napędzie pracującym przy nastawach regulatorów jak na Rys. 10, pokazano na Rys.9.

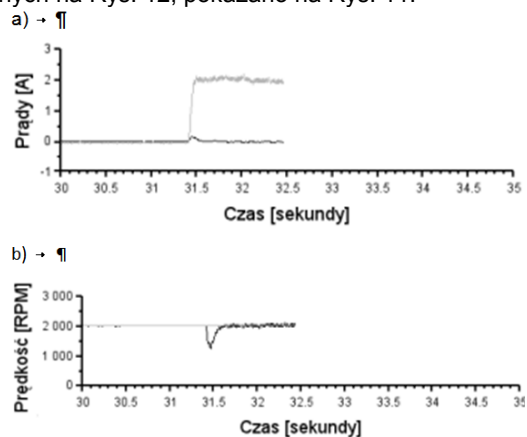


Rys. 9. Przebiegi przejściowe zaobserwowane w badanym napędzie w czasie jego odpowiedzi na skokową zmianę wartości zadanej prędkości: a) przebiegi obu składowych prądu stojana (linia czarna - prąd strumienia magnetycznego, linia szara - prąd momentu obrotowego silnika), b) przebieg odtwarzanej w napędzie prędkości przedstawiony na tle skoku wartości zadanej (linia czarna - prędkość rzeczywista, linia szara - zadana prędkość).



Rys. 10. Nastawy regulatorów w układach regulacji prędkości obrotowej wału silnika, momentu obrotowego i strumienia wzbudzenia.

- Badanie wpływu zakłócenia na obiekt  
Testy obciążenia silnika BLDC zostały przeprowadzone w celu sprawdzenia wpływu wprowadzonego zakłócenia skokowego podczas działania układu napędowego, który znajduje się w stanie ustalonym. Skokowa zmiana zadawanej prędkości dla silnika BLDC wynosi 2000 obr/min. Moment obciążenia generowany jest przez hamulec mechaniczny, na poziomie odpowiadającym prądowi ok. 2 A. Nierównomierny przebieg momentu obciążającego wynika z faktu, że w różnych położeniach wału silnika siła dociskająca hamulca nie jest stała. Takie zachowanie układu wynika głównie z niedokładności w wykonaniu elementów mechanicznych hamulca. Przykładowe przebiegi obu składowych prądu stojana i prędkości obrotowej zaobserwowane w czasie testów obciążenia przeprowadzanych przy nastawach regulatorów podanych na Rys. 12, pokazano na Rys. 11.



Rys. 11. Przebiegi przejściowe zaobserwowane w badanym napędzie w czasie raptownej zmiany obciążenia na wale silnika: a) przebiegi obu składowych prądu stojana (linia czarna - prąd strumienia magnetycznego, linia szara - prąd momentu obrotowego silnika), b) przebieg odtwarzanej w napędzie prędkości (linia czarna - prędkość rzeczywista, linia szara - zadana prędkość).



Rys. 12. Nastawy regulatorów w układach regulacyjnych napędu obowiązujące w czasie wspomnianej wyżej jego próby obciążeniowej.

## Podsumowanie

Artykuł przedstawia modułowy system napędowy dla silników bezszczotkowych ze sterowaniem połowo-zorientowanym oraz projekt interfejsu graficznego opartego na darmowym środowisku Scilab-Xcos, które służy do testowania sterowania silnikami oraz może być alternatywą dla tworzenia własnego oprogramowania interfejsu graficznego pozwalającego na obsługę on-line takiego napędu. Oprogramowanie może być uruchomione zarówno w środowisku Windows, jak i Linux. Zaproponowane rozwiązanie komunikacyjne oparte na interfejsie USB i odmierzaniu czasu przez moduł sprzętowy pozwala uzyskać funkcjonalność konfiguracyjną i testową analogiczną z możliwościami oferowanymi przez oprogramowanie komercyjne. Dodatkowo środowisko Scilab-Xcos może posłużyć do dalszego rozwinięcia algorytmu sterowania oraz przetwarzania danych otrzymanych z modułu sterowania. Rozwój oprogramowania może iść w kierunku idei sterowania hardware-in-the-loop, implementacji algorytmów Soft Start i Soft Stop oraz zamknięcia zewnętrznej pętli regulacji pozycji co pozwoli na stosowanie tego samego sterownika dla napędów manipulatora.

**Autorzy:** mgr inż. Damian Wroński, Politechnika Łódzka, Instytut Automatyki, ul. Stefanowskiego 18/22 90-924 Łódź, E-mail: damian.wronski.95.dws@gmail.com; dr hab. inż. Grzegorz Granosik, prof. PŁ, Politechnika Łódzka, Instytut Automatyki, ul. Stefanowskiego 18/22 90-924 Łódź, E-mail: granosik@p.lodz.pl

## LITERATURA

- [1] Shiyong Lee, Ph.D., Tom Lemley, Gene Keohane, A comparison study of the commutation methods for the three-phase permanent magnet brushless DC motor, Electrical Manufacturing Technical Conference 2009: Electrical Manufacturing and Coil Winding Expo. 2009. p. 49-55.
- [2] Infineon, XMC4800 Data Sheet [https://www.infineon.com/dgdl/Infineon-XMC4700-XMC4800-DS-v01\\_01-EN.pdf?fileId=5546d462518ffd850151908ea8db00b3](https://www.infineon.com/dgdl/Infineon-XMC4700-XMC4800-DS-v01_01-EN.pdf?fileId=5546d462518ffd850151908ea8db00b3) - data dostępu: 14.08.2019
- [3] Texas Instruments, DRV832x6 to 60-V Three-Phase Smart Gate Driver <http://www.ti.com/lit/ds/symlink/drv8323r.pdf> - data dostępu: 14.08.2019
- [4] Austria Mikro Systeme, AS514714 Magnetic Rotary Position Sensor, [https://ams.com/documents/20143/36005/AS5147\\_DS000307\\_2-00.pdf/6921a55b-7cba-bf20-78c0-660d62bd0a5b](https://ams.com/documents/20143/36005/AS5147_DS000307_2-00.pdf/6921a55b-7cba-bf20-78c0-660d62bd0a5b) - data dostępu: 14.08.2019
- [5] Dębowski A., Automatyka. Napęd elektryczny, WNT, Warszawa 2017
- [6] Microchip, Sensorless Field Oriented Control of a PMSM, <http://ww1.microchip.com/downloads/en/appnotes/01078b.pdf> - data dostępu: 14.08.2019
- [7] Wroński D, Robotyczny system napędowy silników bezszczotkowych wraz z interfejsem użytkownika, Praca dyplomowa, Politechnika Łódzka 2019
- [8] Dębowski A., Automatyka – podstawy teorii, WNT, Warszawa 2016 (wyd. II)