

Program do badań układu sterującego – pomiarowego, pracującego w sieci CANopen, przeznaczonego do pojazdu szynowego

W artykule opisano program do badania układu sterującego – pomiarowego pracującego w sieci CANopen. Wykorzystano do tego język programowania ogólnego (język C++). Przez autorów artykułu zostały zrealizowane funkcje dotyczące sieci CANopen związane z konfiguracją i monitorowaniem pracy w sieci oraz funkcje związane z uruchomieniem programów badawczych. Przedstawiono przykład uruchamiania programu sterującego hamulcem pneumatycznym pojazdu szynowego z wykorzystaniem 12 - pozycyjnego zadajnika. Praca jest kontynuacją artykułu „Przyszłościowe współbieżne mikroprocesorowe inteligentne systemy mechatroniczne w sterowaniu i diagnostyce pojazdów szynowych” przedstawionego w czasopiśmie „Pojazdy Szynowe” 4/2007; 1/2008; 2/2008. Artykuł powstał w wyniku realizacji projektu badawczego KBN 4T 12C 04929 pt. „Rozproszone współbieżne mikroprocesorowe inteligentne podsystemy mechatroniczne w sterowaniu i diagnostyce pojazdów szynowych”.

1. Wprowadzenie

W pracach [1, 2, 3 i 4] przedstawiono wiadomości na temat sieci CAN. W artykule opisano program Copen zrealizowany przez autorów artykułu, spełniający rolę pomocniczą przy tworzeniu i badaniu algorytmów sterującego - pomiarowych. Algorytmy te mają być realizowane przez układy komputerowe pracujące w sieci CANopen.

Docelowy układ może składać się np. z kosztownych sterowników i komputerów przemysłowych, dlatego korzystne jest wstępne dopracowanie algorytmu przy wykorzystaniu sprzętu PC powszechnego użytku.

Funkcje programu dzielą się zasadniczo na dwie grupy:

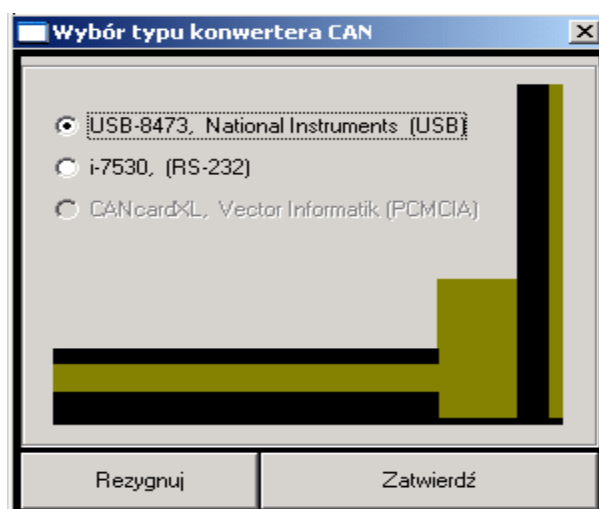
- funkcje dotyczące sieci CANopen związane z konfiguracją i monitorowaniem pracy sieci
- funkcje związane z uruchamianiem programów badawczych.

Typowy komputer PC, aby mógł współpracować z siecią CAN (która jest podstawą standardu CANopen), musi być wyposażony w sprzęg do tej sieci. Obecnie program może korzystać z trzech rodzajów sprzęgów. Są to konwertery:

- USB / CAN, typu USB-8473 firmy National Instruments
- RS232 / CAN, typu i-7530 (Tech base)
- PCMCIA / CAN, typu CANcardXL firmy Vector Informatik.

Wyboru typu sprzęgu dokonuje się przy starcie programu, co pokazuje rys. 1.

Możliwe do wyboru są te pozycje, które są aktualnie zainstalowane w komputerze.



Rys. 1. Okno wyboru typu złącza CAN

Po dokonaniu wyboru mamy na ekranie okno główne przedstawione na rys. 2. Funkcje dotyczące sieci są dostępne przez menu "Sieć", a funkcje programów badawczych przez menu "Programy".

2. Funkcje konfiguracyjne programu

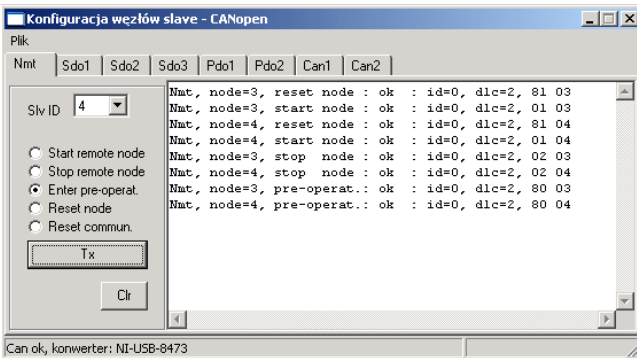
Funkcje dotyczące konfiguracji, testowania i monitorowania sieci są skupione w oknie otwieranym z menu: Sieć | Konfiguracja (rys. 2). Funkcje są pogrupowane na oddzielnych kartach odpowiednio do serwisów CANopen: NMT, SDO, PDO. Serwisy CANopen są omówione w [2]. Karty Can1 i Can2 służą do monitorowania sieci na poziomie komunikatów CAN.



Rys. 2. Okno główne programu

2.1. Serwis NMT

Karta Nmt (rys.3) pozwala na wykonywanie przez program funkcji mastera serwisu: Network Management (NMT). W każdej sieci CANopen musi być węzeł pełniący taką funkcję. Aby wykonać operację należy wybrać adres węzła slave (Slv ID), funkcję (Start remote node, ...) i wysłać komunikat naciskając klawisz Tx.



Rys. 3. Karta serwisu: Network Management (NMT)

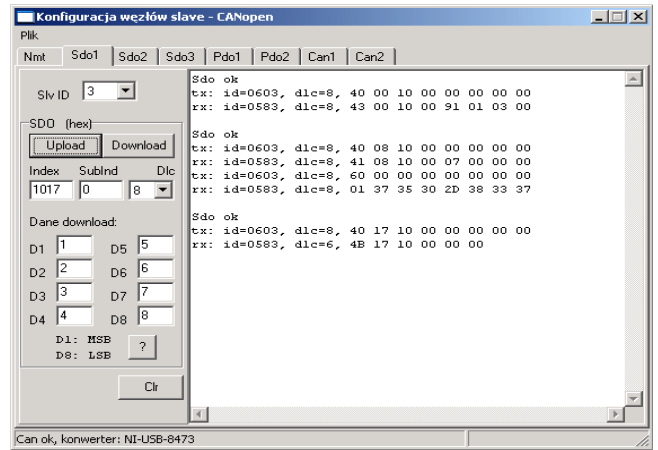
2.2. Serwis SDO

Karty Sdo1, Sdo2, Sdo3 (rys. 4, 5, 6) pozwalają na wykonywanie przez program funkcji klienta serwisu: Service Data Objects (SDO). Serwis SDO służy do konfiguracji słownika obiektów węzłów slave. Funkcje polegają na odczycie (Upload) i zapisie (Download) pozycji słownika.

Poszczególne pozycje słownika są określone przez indeks i subindeks i mogą zawierać dane złożone z 1 .. 8 bajtów. Karta Sdo1 pozwala na dostęp do dowolnej pozycji słownika. Aby odczytać pozycję należy podać indeks i subindeks i kliknąć: Upload. Aby zapisać pozycję należy dodatkowo podać liczbę zapisywanych bajtów (pole Dlc) oraz ich wartości (pola D1 .. D8). Aby korzystać z tej karty trzeba znać szczegółowy opis pozycji słownika obiektów zawarty np. w opisach standardu CANopen [4], [5] lub w dokumentacji konkretnego węzła slave.

W polu informacyjnym pokazywana jest wymiana komunikatów w ramach każdej operacji SDO. Przez tx: oznaczone są komunikaty wysyłane, przez rx: odbierane

Karty Sdo2 i Sdo3 ułatwiają wykonywanie wybranych operacji SDO. Modyfikowane parametry są przedstawione w sposób wygodny dla

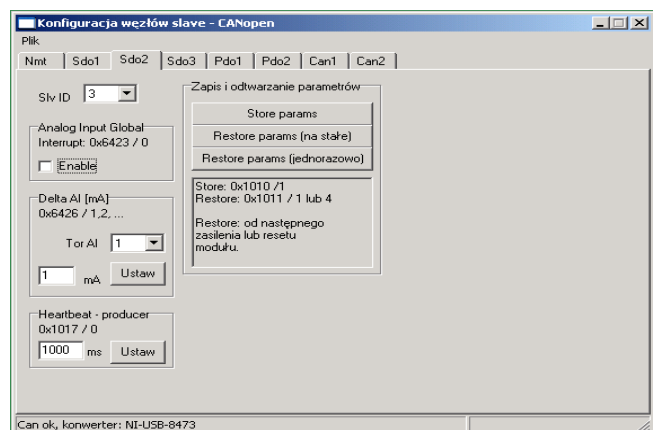


Rys. 4. Karta ogólna serwisu Service Data Objects (SDO).

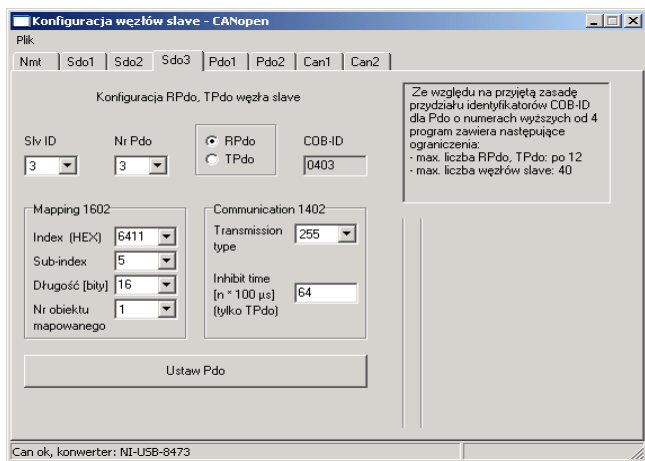
użytkownika. Ustalenie indeksu, subindeksu i zestawu bajtów danych wykonuje program. Przykładem na karcie Sdo2 rys. 3 jest parametr Heartbeat producer [2], związany z kontrolą poprawności pracy sieci. Parametr ten ustala odstęp czasowy wysyłania kolejnych komunikatów Heartbeat przez węzeł slave. Należy tylko podać czas w ms i kliknąć klawisz: Ustaw.

Karta Sdo3 rys. 6 zawiera elementy związane z konfiguracją obiektów PDO (transmisja danych procesowych). Zadanie obejmuje ustalenie parametrów mapowania (powiązania danych PDO z odpowiednimi pozycjami słownika obiektów węzła slave) i parametrów komunikacyjnych obiektów PDO [2]. Konfiguracja PDO jest konieczna, gdy domyślny zdefiniowany w [4] układ PDO jest w danym zastosowaniu niewystarczający. W szczególności potrzeba taka zachodzi, gdy oprócz stanów wejść i wyjść fizycznych należy przysyłać między węzłami wartości zmiennych obliczeniowych. W takiej sytuacji należy ustalić rozszerzony układ identyfikatorów COB-ID dla obiektów PDO. W [3], p.4.2 przedstawiono przykład takiego układu. Elementy na karcie Sdo3

rys. 6 ułatwiają konfigurację według powyższego układu.



Rys. 5. Karta wybranych funkcji serwisu Service Data Objects (SDO)

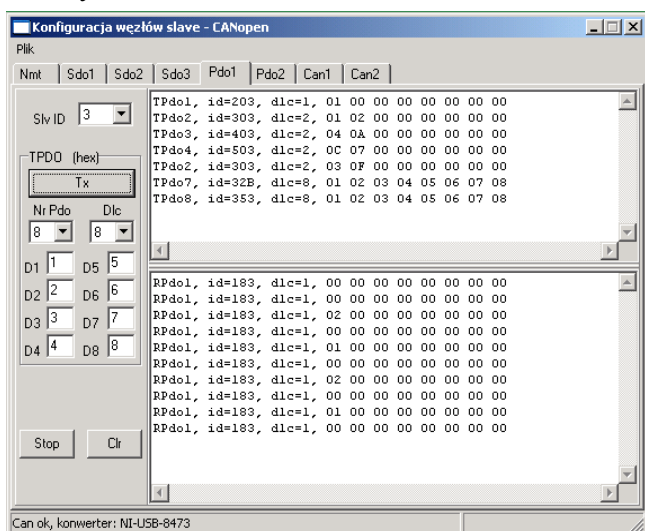


Rys. 6. Karta serwisu Service Data Objects (SDO) dla konfiguracji obiektów PDO

2.3. Serwis PDO

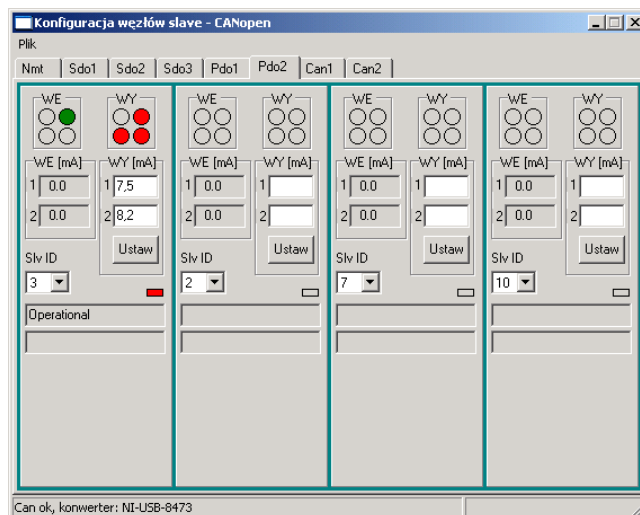
Serwis Process Data Objects (PDO) służy do transmisji danych procesowych podczas normalnej pracy węzła za pomocą obiektów PDO. Karta Pdo1 (rys. 7) służy do wysłania i odbierania obiektów PDO. Przy wysłaniu należy podać adres węzła slave, numer PDO, liczbę wysyłanych bajtów (Dlc) i wartość bajtów (pola D1 .. D8). Program ustala identyfikator komunikatu (COB-ID) przy czym dla PDO nr 1..4 według domyślnego układu identyfikatorów [4] a dla pozostałych według wspomnianego wcześniej układu przykładowego [3], p.4.2.

W polu informacyjnym górnym pokazywane są wysyłane komunikaty określone jako TPDO. W polu dolnym pokazywane są odbierane przez program obiekty określone RPDO.



Rys. 7. Karta ogólna serwisu Process Data Objects (PDO)

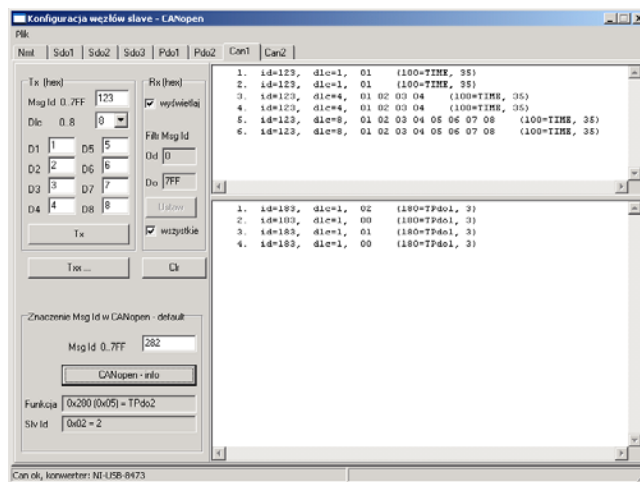
Karta Pdo2 (rys. 8) obsługuje wybrane obiekty PDO w sposób wygodny dla użytkownika. Obsługa dotyczy czterech wejść i czterech wyjść binarnych oraz dwóch wejść i dwóch wyjść analogowych. Jednocześnie można obsługiwać do czterech węzłów slave o *wybranych* adresach (Slv ID).



Rys. 8. Karta pomocnicza serwisu Process Data Objects (PDO)

2.4. Transmisja CAN

Na karcie Can1 (rys. 9) można wysłać i odebrać dowolny komunikat CAN. Komunikaty są prezentowane odpowiednio w górnym i dolnym polu informacyjnym. Dla komunikatów odbieranych można zastosować jednozakresowy filtr powodujący pomijanie pewnego zakresu identyfikatorów.



Rys. 9. Karta wysyłania i odbierania komunikatów CAN

Karta Can2 (rys. 10) służy do prezentacji odbieranych komunikatów CAN. W odróżnieniu od karty Can1 komunikaty o tym samym identyfikatorze są prezentowane w tym samym wierszu zawierającym licznik odebranych komunikatów (kolumna N).

3. Funkcje badawcze programu

Program Copen zawiera program badawczy: Hamowanie 1. Algorytm tego programu jest omówiony w [3]. W skrócie algorytm polega na realizacji hamowania sterowanego 12 - pozycyjnym zadajnikiem. Hamowanie składa się z zadanej liczby odcinków

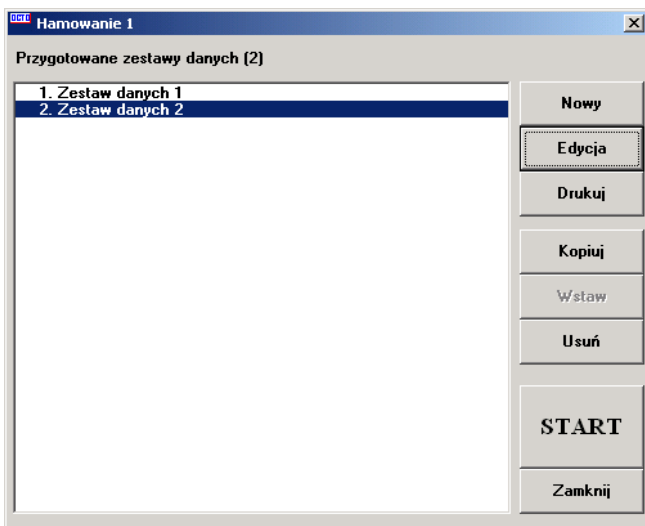
Nm	Sdo1	Sdo2	Sdo3	Pdo1	Pdo2	Can1	Can2	
Lp	Msg Id	Dlc	Dane				N	CANopen: funkcja, (SlvId: dec, hex)
1	703	1	05				61	700=ErrCtr, (3, 3)
2	083	8	00 50 81 00 01 00 00 00				4	080= EMCY, (3, 3)
3	183	1	00				18	180= TPdo1, (3, 3)
4	203	1	0E				4	200= RPdo1, (3, 3)
5	303	2	03 0F				2	300= RPdo2, (3, 3)
6	403	2	04 0A				1	400= RPdo3, (3, 3)
7	503	2	0C 07				1	500= RPdo4, (3, 3)
8	32B	8	01 02 03 04 05 06 07 08				1	300= RPdo2, (43, 2B)
9	353	8	01 02 03 04 05 06 07 08				1	300= RPdo2, (83, 53)
10	123	8	01 02 03 04 05 06 07 08				6	100= TIME, (35, 23)
11								
12								
13								

Rys. 10. Karta dodatkowa dla prezentacji odbieranych komunikatów CAN

hamowania o zadany czas trwania. Na każdym odcinku załączona jest wybrana pozycja zadajnika hamowania. Układ hamulcowy jest sterowany dwoma zaworami sterowanymi analogowo napięciem 0 .. 10V. Jeden z zaworów podnosi ciśnienie w układzie, a drugi opuszcza według ustalonego algorytmu [3].

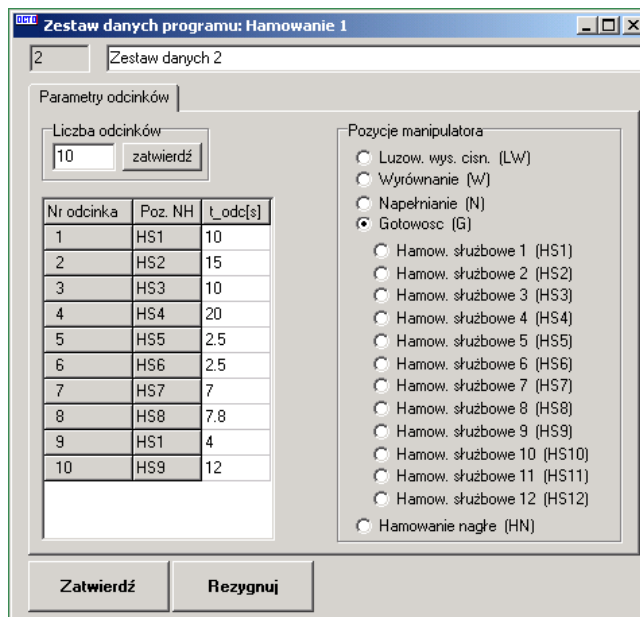
3.1. Przygotowanie programu badawczego

Można przygotować dowolną liczbę zestawów danych i wykonywać program z tym zestawem wielokrotnie. Okno na rys. 11 służy do organizowania i uruchamiania programu z poszczególnymi zestawami danych.



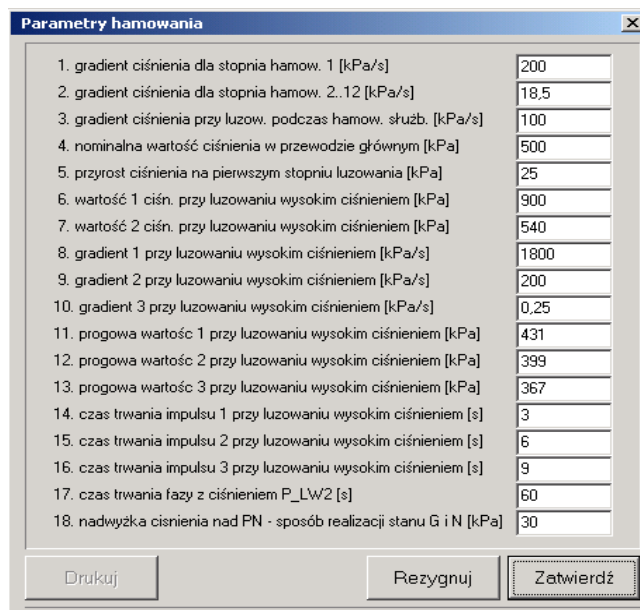
Rys. 11. Okno organizowania zestawów danych i uruchamiania programów badawczych

Przygotowanie poszczególnych zestawów danych odbywa się w kolejnym oknie (rys. 12). Po ustaleniu liczby odcinków programu dla każdego odcinka ustala się czas jego trwania (t_{odc}) i wybiera z listy pozycję zadajnika hamowania.



Rys. 12. Okno przygotowania danych programu Hamowanie 1

Algorytm regulacji ciśnienia jest określony zestawem parametrów ustalanych wspólnie dla wszystkich zestawów danych w oknie pokazanym na rys. 13.

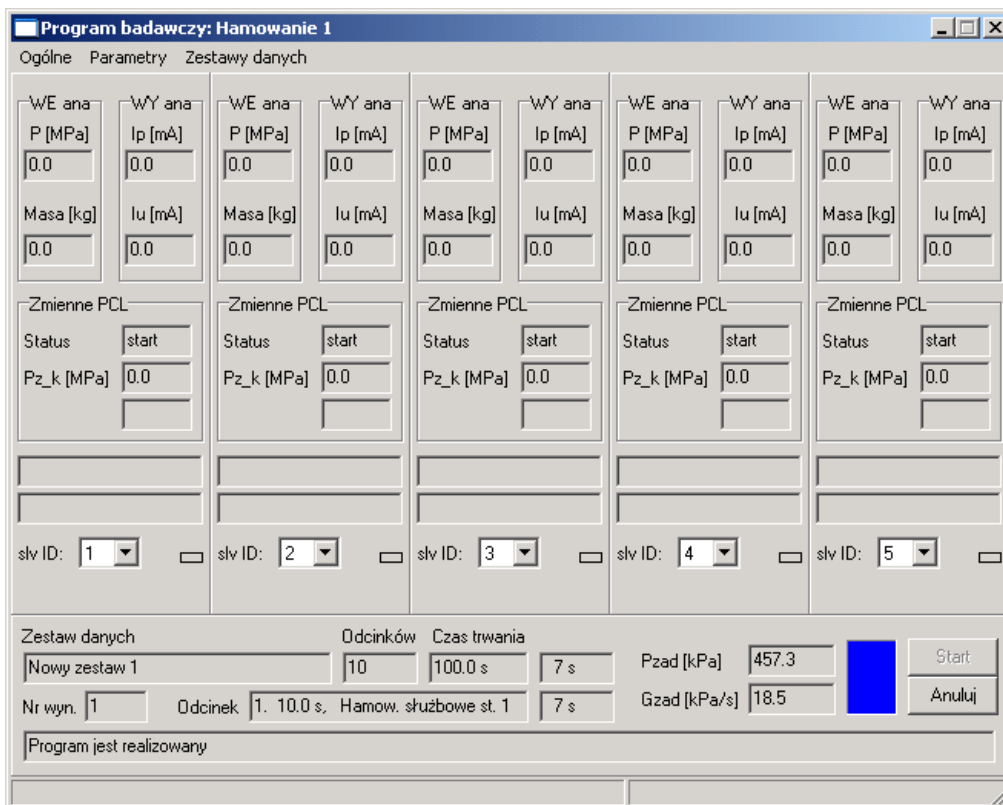


Rys. 13. Okno ustalania parametrów hamowania

3.2. Realizacja programu badawczego

Po wybraniu klawisza: Start w oknie zestawów danych (p.3.1) otwiera się okno z rys. 14. Po wybraniu kolejnego klawisza: Start rozpoczyna się realizacja algorytmu programu badawczego.

W oknie są prezentowane różne elementy związane z przebiegiem programu. W dolnym panelu są wartości zadane ustalone zgodnie z przygotowanym zestawem danych i wysyłane do stacji slave realizujących hamowanie. W górnej części okna jest prezentowany aktualny stan węzłów slave (jednocześnie do pięciu węzłów).



Rys. 14. Okno realizacji programu badawczego.

4. Podsumowanie

Przedstawiono program wspierający badanie układu pomiarowo - sterującego pracującego w sieci CAN open na etapie jego projektowania i uruchamiania. Program umożliwi konfigurację i monitorowanie pracy sieci oraz realizację algorytmów pomiarowo - sterujących. Powyższe funkcje pozwalają na badanie różnych wariantów pracy układu przeznaczonego do pojazdów szynowych przed wykonaniem jego wersji docelowej.

Literatura

- [1] Bocian S.: *Przyszłościowe współbieżne mikroprocesorowe inteligentne systemy mechatroniczne w sterowaniu i diagnostyce pojazdów szynowych (1)*. *Pojazdy Szynowe Nr 4/2007*.
- [2] *Koncepcja układu do badania algorytmów sterujących w pojazdach szynowych. Opis układu sieciowego CANopen*. Instytut Pojazdów Szynowych „Tabor” w Poznaniu, OR-9200.
- [3] *Koncepcja układu do badania algorytmów sterujących w pojazdach szynowych w oparciu o sieć CANopen. Program testowy algorytmów sterujących*. Instytut Pojazdów Szynowych „Tabor” w Poznaniu, OR-9280.
- [4] *CANopen. Application Layer and Communication Profile. CiA Draft Standard 301. CAN in Automation, Am Weichselgarten 26, D-91058 Erlangen, 2002, www.can-cia.org*
- [5] *CANopen. Device Profile for Generic I/O Modules. CiA Draft Standard 401. CAN in Automation, Am Weichselgarten 26, D-91058 Erlangen, Version 2.1, 2002, www.can-cia.org*