

BOCIAN Stanisław¹
FRĄCZEK Jerzy²

ZASTOSOWANIE PROGRAMU MATLAB DO ZADAŃ POMIAROWYCH

W artykule przedstawiono sytuację wykorzystania programu Matlab do jednoczesnego wykonywania pomiarów sygnałów i prowadzenia związanych z nimi obliczeń. Dotyczy to zarówno prac związanych z projektowaniem i uruchamianiem urządzeń jak i prac związanych z ich badaniem i testowaniem.

APPLICATION THE MATLAB PROGRAM FOR MEASURING TASKS

This article presents the situation of using the Matlab program for the simultaneous measurements of signals and carrying out the related to them the calculations. This applies both to works related to design and start the devices and work associated with their investigation and testing.

1. WSTĘP

W praktyce laboratoryjnej i konstruktorskiej często spotykaną sytuacją jest potrzeba jednoczesnego wykonywania pomiarów sygnałów wykonywania związanych z nimi złożonych niekiedy obliczeń. Dotyczy to zarówno prac związanych z projektowaniem i uruchamianiem urządzeń jak i prac związanych z badaniem, testowaniem lub kontrolą urządzeń i układów.

Do prac obliczeniowych często wykorzystywany jest program Matlab firmy MathWorks. Program ten posiada również rzadziej wykorzystywane możliwości prowadzenia prac związanych z pomiarami i sterowaniem przy użyciu zewnętrznych urządzeń pomiarowo – sterujących.

Możliwości te obejmują wszystkie typowe w takiej sytuacji zadania. W zakresie pomiarów Matlab umożliwia:

- konfigurację wybranego urządzenia pomiarowego
- sterowanie przebiegiem pomiarów
- obserwację sygnałów podczas wykonywania pomiarów w postaci różnego typu wykresów lub chwilowych wartości liczbowych
- prezentację wykonanych zapisów po ich zakończeniu

¹ Stanisław BOCIAN Instytut Pojazdów Szynowych „TABOR” POLSKA; Poznań 61-055;Warszawska 181.
Telefon: + 48 (0)61 664 14 38; E-mail: Elektrotechnika @ tabor.com.pl

² Jerzy Frączek Instytut Pojazdów Szynowych „TABOR” POLSKA; Poznań 61-055;Warszawska 181.
Telefon: + 48 (0)61 664 14 38; E-mail: Elektrotechnika @ tabor.com.pl

– zapis zebranych wyników w pliku dyskowym.

Najnowsze wersje programu Matlab zawierają bardzo zaawansowane, wyspecjalizowane możliwości w omawianym zakresie. Ich wykorzystanie wymaga jednak odpowiednio zaawansowanego przygotowania.

W niniejszym tekście przedstawiono podstawowe własności programu Matlab zawarte głównie w pakiecie rozszerzającym: Data Acquisition Toolbox. Korzystanie z funkcji tego pakietu możliwe jest przy niewielkim nakładzie pracy szczególnie, gdy korzysta się z przygotowanych wcześniej szablonów. Materiał przedstawiony w tekście zakłada podstawową wiedzę o korzystaniu z programu Matlab.

Możliwe są różne podejścia do zagadnienia wykonania zadania pomiarowego. Najprostsze podejście to korzystanie bezpośrednio z komend programu Matlab wpisywanych w linii poleceń lub w skrypcie Matlaba czyli w pliku tekstowym zwanym m – plikiem. Inne podejście polega na przygotowaniu typowego dla systemu operacyjnego Windows okna zawierającego pola wykresu, przyciski, pola tekstowe i podobne elementy. Do przygotowania takiego okna służy zestaw funkcji programu Matlab o nazwie GUIDE.

Pierwsze podejście:

- umożliwia bezpośredni dostęp do wszystkich funkcji związanych z zagadnieniem pomiarów
- umożliwia maksymalnie elastyczne korzystanie z funkcji
- wymaga pewnej znajomości pakietu Data Acquisition Toolbox i zagadnień pomiarowych.

Drugie podejście:

- pozwala przygotować program Matlab do wykorzystania przez osobę nie znającą pakietu Data Acquisition Toolbox
- jest wygodne przy wykonywaniu wielu powtarzających się, podobnych działań.

W tekście omówiono podstawowe elementy umożliwiające szybkie przystosowanie programu Matlab do zadań pomiarowych w ramach pierwszego podejścia.

2. ELEMENTY ZWIĄZANE Z POMIARAMI

Aby wykorzystać Matlaba do zadań pomiarowych potrzebne są następujące elementy:

- sprzęt pomiarowy, przystosowany do współpracy z komputerem PC oraz od strony programowej przystosowany do współpracy z programem Matlab
- w ramach Matlaba musi być zainstalowany pakiet rozszerzający: 'Data Acquisition Toolbox'. Obecność pakietu można sprawdzić poleceniem 'ver'. Nazwa pakietu musi znajdować się na wyświetlonej liście pakietów dodatkowych
- w systemie operacyjnym muszą być zainstalowane sterowniki urządzenia pomiarowo – sterującego dostarczone przez producenta sprzętu
- w ramach Matlaba musi być zainstalowany element programowy zwany 'Driver Adaptor' (adapter). Element ten pośredniczy między Matlabem a wspomnianym wcześniej sterownikiem producenta sprzętu. Pakiet 'Data Acquisition Toolbox' zawiera pewną liczbę (zależną od wersji) adapterów dla kilku producentów (np. Advantech, Measurement Computing, National Instruments). Ich listę można znaleźć w systemie pomocy pakietu lub korzystając z funkcji daqhwinfo() (p.2.1).

W innych przypadkach (np. firmy IOtech) producent sprzętu dostarcza środki umożliwiające zainstalowanie odpowiedniego adaptera.

Szczególnym przypadkiem jest karta dźwiękowa. Jest ona zwykle obecna na każdym komputerze wraz ze swoim sterownikiem. Adapter karty jest również zawsze elementem pakietu 'Data Acquisition Toolbox'. Wejście mikrofonowe tej karty jest urządzeniem pomiarowym, które można łatwo wykorzystać np. do nauki obsługi pakietu. W funkcjach pakietu 'Data Acquisition Toolbox' karta dźwiękowa jest identyfikowana przez nazwę 'winsound'.

Nazwy identyfikujące sprzęt innych firm to na przykład 'mcc' dla Measurement Computing lub 'nidaq' dla firmy National Instruments.

2.1. Uzyskiwanie informacji o sprzęcie

Do uzyskiwania informacji o sprzęcie, który wraz ze sterownikiem i adapterem jest dostępny w Matlabie służy funkcja `daqhwinfo`. Funkcja ta zwraca informacje, które dla danego sprzętu mają stałe wartości – w odróżnieniu od właściwości (properties), omówionych w p.4, których wartości mogą być modyfikowane przez użytkownika.

Używanie funkcji `daqhwinfo` należy rozpocząć w formie:

```
>> out=daqhwinfo
```

Wyświetlone zostaną podstawowe, ogólne informacje o pakiecie 'Data Acquisition Toolbox' oraz utworzona zostanie zmienna `out`. Zmienna będzie wykorzystywana przy kolejnych odwołaniach do funkcji `daqhwinfo`.

Listę zainstalowanych adapterów uzyskamy za pomocą wywołania:

```
>> out.InstalledAdaptors
```

Informacje dotyczące konkretnego adaptera (tutaj: firmy Measurement Computing):

```
>> out=daqhwinfo('mcc')
```

przykładowa odpowiedź:

```
out=
  AdaptorDllName: [1x64 char]
  AdaptorDllVersion: '2.11 (R2007b)'
  AdaptorName: 'mcc'
  BoardNames: {'USB-1408FS'}
  InstalledBoardIds: {'0'}
  ObjectConstructorName: {1x3 cell}
```

Ważną pozycją, potrzebną w dalszej pracy jest identyfikator urządzenia lub urządzeń związanych z adapterem. Wartości identyfikatorów podane są w wierszu:

```
InstalledBoardIds: {'0'}
```

Jeżeli w systemie będzie obsługiwanych kilka urządzeń tej samej firmy, to każde z nich będzie miało przydzielony inny identyfikator. Wartość tego identyfikatora może być liczbą lub tekstem. Na przykład dla urządzeń firmy National Instruments identyfikatory mają postać: 'Dev1', 'Dev2', itd. W powyższym przykładzie identyfikatorem jest liczba 0.

W celu uzyskania informacji na temat poszczególnych podsystemów urządzenia należy najpierw utworzyć obiekt reprezentujący ten podsystem. Na przykład dla podsystemu wejść analogowych urządzenia firmy Measurement Computing tworzymy obiekt o nazwie np. `ai` (p.3.1):

```
>> ai=analoginput('mcc',0)
```

Liczba '0' jest omówionym wyżej identyfikatorem urządzenia.

Listę wszystkich właściwości podsystemu uzyskamy wywołując funkcję `daqhwinfo` w poniższy sposób :

```
>> out=daqhwinfo(ai)
```

fragment przykładowej odpowiedzi:

```
out=
  AdaptorName: 'mcc'
  Bits: 14
  ...
  InputRanges: [8x2 double]
  ...
```

Na uzyskanej liście właściwości znajduje się między innymi pozycja `InputRanges` (czyli lista dostępnych zakresów wejściowych wejść analogowych).

Rozszerzone informacje na temat tej właściwości, czyli wartości liczbowe określające poszczególne zakresy uzyskamy wywołując funkcję `daqhwinfo` w postaci:

```
>> out=daqhwinfo(ai,'InputRanges')
```

przykładowa odpowiedź:

```
out=
  -5.0000  5.0000
  -10.0000 10.0000
```

3. PROGRAMOWANIE Z LINII POLECEŃ (OKNO COMMAND WINDOWS)

3.1. Minimalny zestaw poleceń

Minimalny zestaw poleceń programu Matlab pozwalający wykonać pomiar określonej liczby próbek obejmuje:

- utworzenie analogowego obiektu wejściowego związanego ze sprzętem pomiarowym (na przykładzie wejścia mikrofonowego):

```
>> ai = analoginput('winsound',0)
```

 liczba 0 jest identyfikatorem urządzenia. Sposób ustalania identyfikatora jest omówiony w p. 2.1
- dodanie jednego toru pomiarowego do obiektu urządzenia:

```
>> addchannel(ai,1)
```

 Obie powyższe operacje są wykonywane jednorazowo przed rozpoczęciem pomiarów.
- ustalenie częstotliwości próbkowania: 10000 próbek / sekundę:

```
>> set(ai,'SampleRate',10000)
```
- ustalenie liczby próbek, które mają być zebrane.
 Z podanej wcześniej częstotliwości próbkowania i liczby zbieranych próbek wynika czas wykonywania pomiarów. Tutaj ustalenie liczby 50000 próbek oznacza czas 5s ($t=50000/10000$):

```
>> set(ai,'SamplesPerTrigger',50000)
```
- rozpoczęcie pomiarów.

```
>> start(ai)
```
- odebranie wyników wykonanych pomiarów z bufora sterownika do pamięci programu, czyli tutaj do zmiennej 'zapis', która może być dalej dowolnie przetwarzana w Matlabie.
 Ponieważ wcześniej ustalono pomiar w jednym torze pomiarowym `addchannel(ai,1)`, to zmienna 'zapis' będzie wektorem zawierającym 50000 wierszy.

- >> **zapis = getdata(ai)**
- prezentacja zebranych próbek w postaci wykresu i wyświetlenie siatki:
 - >> **plot(zapis)**
 - >> **grid on**
- zakończenie pracy obiektu 'ai' oraz usunięcie go z pamięci (wykonywane jednorazowo po zakończeniu realizacji zadań pomiarowych):
 - >> **delete(ai)**
 - >> **clear ai**

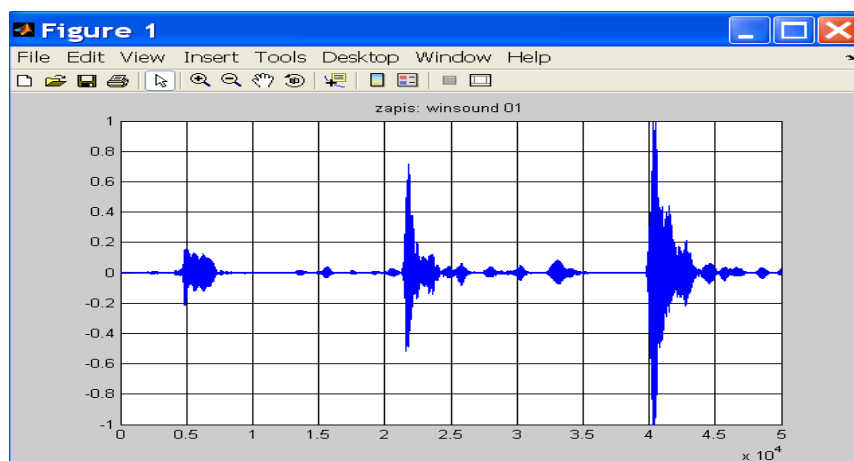
Powyższe polecenia (napisane pogrubioną czcionką) należy zapisać w pliku tekstowym z rozszerzeniem .m, np. 'test01.m'.

Plik należy uruchomić w typowy dla Matlaba sposób, to znaczy:

- katalog, w którym znajduje się plik musi znajdować się na liście ścieżek Matlaba – polecenie: menu File | Set Path
- w oknie Command Window, w linii poleceń należy wpisać nazwę pliku bez rozszerzenia:
 - >> **test01**

i nacisnąć klawisz Enter.

Gdy minie czas zbierania zadanej liczby próbek (tutaj: 5 s.) na ekranie pojawi się okno pokazane na rys. 1.



Rys. 1. Przykładowy 5–sekundowy zapis z wejścia mikrofonowego karty dźwiękowej. Wartości sygnału są podane bez przeliczania w [V].

3.2. Narzędzia wspomagające pracę z uzyskanym przebiegiem

Uzyskany w poprzednim punkcie przebieg jest zapisany w zmiennej: 'zapis', która automatycznie została umieszczona w przestrzeni roboczej Matlaba (Workspace). Zmienna ta jest wektorem i można jej używać w dowolnych operacjach obliczeniowych, zapisywać do plików dyskowych lub prezentować w dowolny, inny sposób.

W ramach wyświetlonego okna można przeprowadzać szereg operacji przy pomocy

narzędzi zgrupowanych w trzech dodatkowych panelach. Panele można otwierać z menu View lub przy pomocy ikony z paska skrótów (rys 2).

– Figure Palette

Panel jest podzielony na trzy obszary:

- New Subplots: służy do tworzenia kolejnych wykresów dwu – i trój – wymiarowych
- Variables: zawiera zmienne obszaru roboczego Matlab'a, które można umieszczać na wykresach (np. przeciągając je myszką)
- Annotations: zestaw narzędzi do umieszczania na wykresach opisów, strzałek i innych elementów graficznych

– Plot Browser

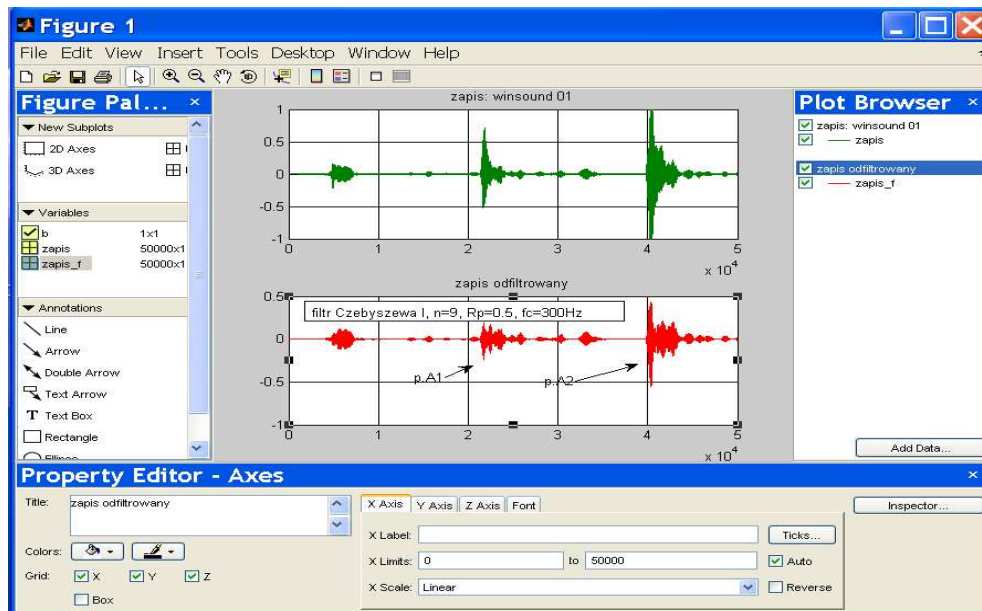
Zawiera listę wszystkich pól wykresów i linii przebiegów. Kliknięcie na dany element powoduje jego zaznaczenie i umożliwia modyfikację jego parametrów.

– Property Editor

Służy do modyfikacji parametrów wybranego elementu. Bezpośrednio w panelu znajdują się wybrane parametry (np. siatki wykresów). Wszystkie parametry są dostępne po wybraniu klawisza 'More properties....'.

Na rys. 2 pokazano prosty przykład wykorzystania tych narzędzi. Najpierw w środowisku Matlab'a wykonano filtrację sygnału zapisanego w zmiennej 'zapis'. Funkcje związane z filtrami cyfrowymi znajdują się w pakiecie rozszerzającym Matlab'a o nazwie 'Signal Processing Toolbox'.

Zastosowano filtr cyfrowy Czebyszewa typu I (rys. 2) Wynik filtracji przypisano zmiennej 'zapis_f'. Zmienna jest widoczna w obszarze 'Variables' panelu 'Figure Palette'. Przebieg odfiltrowany został pokazany na dodatkowym polu wykresu.



Rys.2. Okno wykresów wraz z panelami zawierającymi narzędzia wspomagające.

4. ZAAWANSOWANE MOŻLIWOŚCI PAKIETU

W p.3 pokazano najprostszy przypadek wykorzystania programu Matlab do zadań pomiarowych. Matlab pozwala jednak na wykorzystanie praktycznie wszystkich możliwości zastosowanego urządzenia.

4.1. Konfiguracja sprzętu

Ogólnie sprzęt pomiarowy umożliwia wykorzystywanie go w różnego typu zadaniach pomiarowych. Różne sposoby działania uzyskuje się przez odpowiednie ustawienie parametrów urządzenia. Matlab umożliwia zarówno odczyt możliwych do zastosowania wartości parametrów jak i ustawienie wymaganych wartości.

Jak pokazano w p. 3.1. praca z urządzeniem pomiarowym w ramach Matlaba zaczyna się od utworzenia obiektu programowego. Obiekt ten do momentu jego usunięcia reprezentuje odpowiedni podsystem urządzenia. W przypadku pomiarów jest to podsystem wejść analogowych. Tutaj obiektowi nadano nazwę 'ai'. Obiekt został utworzony poleceniem: `ai = analoginput('winsound',0)`. Do obiektu trzeba dodać tory pomiarowe, które mają być wykorzystywane. Tutaj dodano jeden tor poleceniem: `addchannel(ai,1)`.

Wszystkie fizyczne parametry urządzenia są dostępne jako właściwości tego obiektu (w Matlabie nazywane Properties). Właściwości dzielą się na:

- właściwości urządzenia wspólne dla wszystkich torów pomiarowych (common properties)
- właściwości poszczególnych torów pomiarowych (channel properties).

Właściwości dzielą się również na:

- bazowe (base properties) – zdefiniowane w Matlabie dla wszystkich typów sprzętu pomiarowego
- specyficzne (device-specific properties) – dodatkowe właściwości określone przez producenta sprzętu.

Listę wszystkich właściwości można znaleźć w systemie pomocy w rozdziale 'Data Acquisition Toolbox', w podrozdziałach odpowiednio:

- Base Properties – Categorical List
- Device-Specific Properties – By Vendor.

Do nadawania wartości właściwościom oraz do ich odczytywania służą polecenia odpowiednio 'set' i 'get'. Jeżeli parametrem tych funkcji będzie tylko nazwa obiektu: `set(ai)`, `get(ai)`, to wynikiem ich działania będzie pełna lista właściwości wspólnych urządzenia. Przy czym funkcja `set` wyświetli dodatkowo możliwe do zastosowania zakresy wartości, a funkcja `get` wyświetli wartości aktualne właściwości.

Zmiana wartości pojedynczej właściwości realizowana jest poleceniem 'set' w poniższej postaci:

- dla właściwości wspólnych:

```
set(nazwa_obiektu, 'nazwa_właściwości', nowa_wartość)
np. nadanie częstotliwości próbkowania wartości 10000 Hz:
>> set(ai, 'SampleRate', 10000)
lub używając pomocniczej zmiennej np. 's_rate'
>> s_rate=10000
>> set(ai, 'SampleRate', s_rate)
```

– dla właściwości toru pomiarowego trzeba dodatkowo wskazać numer tego toru:

```
set(nazwa_obiektu.Channel(nr_toru), 'nazwa_właściwości', nowa_wartość)
np. ustawienie zakresu wejściowego toru nr 1 na wartość +/-2V
>> set(ai.Channel(1), 'InputRange', [-2 2])
```

Alternatywnie oba powyższe polecenia można zapisać w postaci:

```
>> ai.SampleRate=10000
>> ai.Channel(1).InputRange = [-2 2]
```

W podobny sposób, używając polecenia 'get' można odczytywać aktualne ustawienia poszczególnych właściwości, np:

```
>> s_rate=get(ai, 'SampleRate')
lub alternatywnie
>> s_rate= ai.SampleRate.
```

Uwaga:

Niektóre właściwości mogą w nietypowy sposób zależeć od producenta sprzętu. Przykładem jest typ wejścia analogowego w urządzeniach pomiarowych firm Advantech i Measurement Computing. Właściwość ta (wspólna dla torów pomiarowych) w Matlabie nazywa się InputType i oznacza typ wejścia ze wspólną masą lub różnicowe. Dla urządzeń tych producentów w Matlabie właściwość ta może być tylko odczytywana. Do jej zmiany należy użyć np. programu fabrycznego producenta.

Należy zaznaczyć, że powyżej wskazano tylko podstawowe sposoby użycia poleceń set i get. Szersze możliwości, na przykład operowanie zakresem torów pomiarowych, są opisane w systemie pomocy.

4.2. Trigger – start pomiarów

Jednym z zagadnień związanych z pomiarami jest ustalenie momentu ich rozpoczęcia. W przykładzie z p. 3.1. tym momentem jest wykonanie polecenia **start(ai)**. Jest jednak szereg innych możliwości. Ogólnie pomiary rozpoczynają się w momencie wystąpienia zdarzenia zwanego trigger. Zdarzenie to jest zdefiniowane za pomocą szeregu właściwości urządzenia, z których najważniejsze są omówione niżej (ustawianie i odczytywanie właściwości – p. 4.1.).

– TriggerType (rodzaj triggera):

Właściwość ta może przyjąć jedną z trzech wartości: Immediate, Manual lub Software.

Producenci sprzętu mogą definiować dodatkowe wartości rozszerzając możliwości swojego sprzętu w tym zakresie:

- Immediate (natychmiast): pomiary są rozpoczynane bezpośrednio po wykonaniu polecenia start(). Jest to wartość domyślna właściwości TriggerType i dlatego w przykładzie z p. 3.1. przypadek ten wystąpił bez wcześniejszego ustawiania tej wartości. W przypadku tego ustawienia nie można zastosować zjawiska zwanego pretrigger, czyli zapisu pewnej liczby próbek zebranych bezpośrednio przed wystąpieniem triggera.
- Manual (ręcznie): w tym przypadku wykonanie polecenia start() jest tylko przygotowaniem do rozpoczęcia pomiarów. Pomiary rozpoczną się po wykonaniu dodatkowego polecenia: trigger(). Pomiedzy jednym a drugim poleceniem może być realizowana funkcja pretriggera.
- Software (programowo): Pomiary właściwe rozpoczną się gdy wystąpią zdefiniowane

dotatkowo warunki. Warunki te są określone przez kolejne właściwości:

- TriggerChannel: oznacza nr toru pomiarowego
- TriggerCondition: oznacza rodzaj zmiany napięcia na powyższym torze wymagany do wystąpienia zdarzenia trigger.
- TriggerConditionValue: oznacza wartość powyższego napięcia.

Na przykład wykonanie poleceń:

```
>> set(ai,'TriggerType','Software')
>> set(ai,'TriggerChannel',2)
>> set(ai,'TriggerCondition','Falling')
>> set(ai,'TriggerConditionValue',4.5)
```

oznacza, że pomiary rozpoczną się, gdy napięcie na torze nr. 2 (TriggerChannel) spadnie poniżej (TriggerCondition=Falling) wartości 4,5V 'przecinając ją' od wartości wyższej do niższej (warunek nie jest spełniony, gdy w momencie wykonania polecenia start() napięcie na torze nr. 2 już jest poniżej 4,5V).

Wspomniane wcześniej zjawisko pretriggera (oraz potriggera – zbierania próbek po zakończeniu pomiarów właściwych) jest konfigurowane za pomocą właściwości: TriggerDelay i TriggerDelayUnits.

W przykładzie z p. 3.1. dla ustalenia liczby próbek, które mają być zebrane ustawiana jest właściwość SamplesPerTrigger: set(ai,'SamplesPerTrigger',50000). Oznacza ona, że na jedno wystąpienie triggera zbieranych jest 50000 próbek z każdego toru pomiarowego (w tym przykładzie jest 1 tor). W omawianym przykładzie trigger wystąpi jeden raz – po wykonaniu polecenia start(). Wynika to z domyślnego ustawienia właściwości TriggerType=Immediate. W innych przypadkach po poleceniu start() trigger może wystąpić wielokrotnie. Przy każdym wystąpieniu zostanie zebranych 50000 próbek z każdego toru.

4.3. Charakterystyki przeliczeniowe

Typowo zmierzone próbki są odbierane ze sterownika urządzenia w jednostkach napięcia [V]. W ramach pakietu Data Acquisition Toolbox' można ustalić natychmiastowe przeliczanie na jednostki tej wielkości, która faktycznie jest mierzona (np. ciśnienie, siła, temperatura,...) według charakterystyki liniowej.

Do tego celu służą poniższe właściwości toru pomiarowego:

- SensorRange: zakres napięcia wyjściowego zastosowanego przetwornika pomiarowego
- UnitsRange: zakres pomiarowy przetwornika w jednostkach docelowych odpowiadający jego zakresowi napięcia wyjściowego
- Units: tekst opisujący jednostki docelowe, np. 'MPa', 'km/h', ...

Jeżeli zmierzoną wartość próbki [V] oznaczymy przez Up, to wartość przeliczona

W będzie: $W = U_p * (\text{UnitsRange} / \text{SensorRange})$

Przykład.

Do toru pomiarowego nr. 1 przyłączono przetwornik ciśnienia o zakresie pomiarowym 0..1 MPa. Sygnał wyjściowy tego przetwornika odpowiadający powyższemu zakresowi, to 0÷10 V. Wartości zakresów podaje się jako dwupozycyjne wektory:

```
>> set(ai.Channel(1),'SensorRange',[0 10])
>> set(ai.Channel(1),'UnitsRange',[0 1])
>> set(ai.Channel(1),'Units','MPa')
```

Zmierzone napięcie np. 2V zostanie przeliczone na wartość ciśnienia p:

$$p = 2 * (1 / 10) = 0,2 \text{ Mpa.}$$

4.4. Zapis pomiarów do pliku

Wyniki pomiarów można zapisać do pliku dyskowego wraz z informacjami dotyczącymi sesji pomiarowej w formacie wewnętrznym Matlaba lub w postaci tekstowej. Zapis w formacie wewnętrznym jest wygodny, jeżeli zamierzamy dalej opracowywać te wyniki przy pomocy Matlaba. Postać tekstowa umożliwia np. archiwizację wyników lub wykorzystanie ich w innych programach np. arkuszach kalkulacyjnych.

W ramach Matlaba można także korzystać z bibliotek łączonych dynamicznie (.dll) tworzonych w innych systemach. Matlab zawiera również mechanizmy umożliwiające korzystanie z funkcji napisanych bezpośrednio w językach programowania C, Fortran i Java.

5. WNIOSKI

W dalszych pracach należało by wykorzystać program MATLAB do jednoczesnego wykonywania pomiarów lub generowania sygnałów i realizacji złożonych niekiedy obliczeń. Dotyczy to zarówno prac związanych z projektowaniem i uruchamianiem urządzeń jak i prac związanych z badaniem, testowaniem lub kontrolą urządzeń i układów.

6. BIBLIOGRAFIA

- [1] Frączek J.: *Zastosowanie programu MATAB do zadań pomiarowych – wprowadzenie. Opracowanie OR – 10048*. Archiwum, IPS „TABOR”. Poznań wrzesień 20011.
- [2] Zalewski A., Cegiela R.: *Matlab – obliczenia numeryczna i ich zastosowanie*. Nakom, Poznań 1996.
- [3] MATLAB Reference Guide; High-Performance Numeric Computation and Visualization Software. The Math Works Inc.
- [4] MATLAB User's Guide for Microsoft Windows. The Math Works Inc.
- [5] USB-1408FS, USB-based Analog and Digital I/O Module. User's Guide. Measurement Computing.

*Praca wykonana w ramach Projektu Badawczego KBN nr N N509 398236
„Mikrosystemy cyfrowe do inteligentnego, rozproszonego i współbieżnego sterowania
pojazdami szynowymi”*