

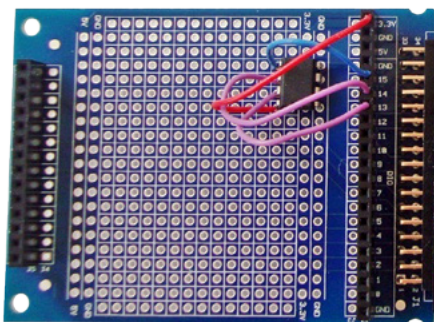
MyRIO – platforma edukacyjna od National Instruments (1)

Firma National Instruments zaproponowała kolejny system przeznaczony dla studentów. Tym razem jest on oparty to technologię wykorzystaną w popularnej platformie CompactRIO. Pozwala ona na przezwycięzenie zagadnień związanych z automatyką, robotyką, systemami kontrolno-pomiarowymi – szczególnie z wbudowanym systemami czasu rzeczywistego.

MyRIO jest względnie niedrogą platformą przeznaczoną do wykonywania wbudowanych systemów sterujących. Przeznaczona głównie dla uczelni i studentów pozwala inżynierom oraz studentom na budowę zaawansowanych systemów. Daje możliwość korzystania z wydajnego dwurdzeniowego procesora Cortex-A9, systemu czasu rzeczywistego, i technologii FPGA(Field Programmable Gate Array – programowalna struktura logiczna), pozwalającej „zbudować” brakujące elementy cyfrowego systemu przetwarzania danych. Oprócz tego udostępnia szereg wejść i wyjść analogowych i cyfrowych, również niezbędne interfejsy komunikacyjne. Pozwala budować systemy kontrolowane zdalnie za pomocą komputera i aplikacji w LabView, jak również tabletu czy smartfona wyposażonego w interfejs Wi-Fi. No i chyba największa zaleta, o której nie wspominałem, to możliwość konfigurowania wszystkich elementów systemu za pomocą LabView, które jest dostarczone na licencji studenckiej wraz z urządzeniem.

W zestawie

Zestaw zawiera moduł myRIO 1900 wraz z niezbędnym do pracy



oprogramowaniem oraz kilkoma akcesoriami, takimi jak: złącze śrubowe do przyłączenia sygnałów z gniazda MSP, złącze kątowe wraz z płytą prototypową pozwalającą na zbudowanie elementów dodatkowych np. wzmacniacza wejściowego lub układu zasilania czujników. W pudełku znajdziemy również przewody USB i audio, śrubokręt oraz zasilacz sieciowy. Na **fotografii 1** pokazano sterownik myRIO i płytę prototypową.

Architektura systemu myRIO

Schemat blokowy systemu pokazano na **rysunku 2**. Serce myRIO stanowi układ SoC firmy Xilinx typu Zynq-7010. Na jego pokładzie znajduje się dwurdzeniowy procesor

Cortex-A9 wspierający obliczenia na liczbach zmiennoprzecinkowych o pojedynczej i podwójnej precyzji oraz na wektorach dzięki instrukcjom NEON. Maksymalna, dopuszczalna częstotliwość taktowania układu wynosi 667 MHz. Mikroprocesor ma pamięć wewnętrzną Cache L1 o pojemności po 32 kB na instrukcje i dane, Cache L2 512 kB, wewnętrzną pamięć programu 256 kB oraz interfejsy do zewnętrznych pamięci DDR3, DDR2 i innych. Wyposażono go w liczne interfejsy komunikacyjne: 2×UART, 2×CAN, 2×I²C, 2×SPI, 4×32-bitowe GPIO, 8 kanałów DMA. Ale moim zdaniem największą zaletą jest umieszczenie w tej samej strukturze układu FPGA z rodziny Artix 7 zawierającego między innymi 28 tys. programowalnych komórek logicznych, 80 programowalnych bloków DSP o organizacji 18×25, a to tylko wybrane cechy tego układu.

Na płycie sterownika umieszczono pamięć Flash o pojemności 256 MB zawierającą system operacyjny czasu rzeczywistego (NI Linux Real-Time). Część tej pamięci może być przeznaczona na dane i aplikacje użytkownika. Pamięć DDR3 o pojemności 512 MB stanowi pamięć operacyjną. Interfejs Wi-Fi służy do konfigurowania i programowania sterownika. Oprócz niego myRIO ma dwa interfejsy USB – Host i Device.

Użytkownikowi oddano do dyspozycji interfejsy UART, I²C i SPI, które wyrowadzone na piny portów MXP, 8 sygnałów PWM, 32 linie cyfrowego wejścia wyjścia i dodatkowe 8 linii w standardzie TTL – 5 V, wejścia i wyjścia analogowe oraz napięcia niezbędne do zasilania dodatkowych układów wejściowych, jeden przycisk i cztery diody do dowolnego wykorzystania.

Zalety wynikające z architektury

Architektura myRIO ma wiele zalet, największą z nich jest połączenie dwurdzeniowego procesora i systemu operacyjnego czasu rzeczywistego ze strukturą FPGA. Takie rozwiązanie pozwala na efektywne wykorzystanie zasobów procesora i przerzucenie żmudnych i czasochłonnych obliczeń na algorytmy zrealizowane sprzętowo w strukturze FPGA. Najlepszym przykładem jest tutaj realizacja filtrów cyfrowych, które są nieodłącznym elementem systemu pomiarowego.



Fotografia 1. Sterownik myRIO z płytą prototypową

W każdym systemie pomiarowym występują przynajmniej filtry antyaliasingowe, na wejściach przetworników. Realizacja takich filtrów w sposób cyfrowy wiąże się z szeregiem operacji mnożenia i sumowania dla każdej próbki sygnału mierzonego i to wielokrotnie. Zakładając, że mierzony sygnał jest próbkowany z częstotliwością 100 kHz łatwo sobie wyobrazić, jak dużo obliczeń trzeba wykonać i to w ściśle określonym czasie, ponieważ bufony danych mają zwykle ograniczoną pojemność, a nowe dane ciągle napływają. Angażowanie do tego procesu CPU nie jest zbyt efektywne, ponieważ w tym czasie może on wykonywać bardziej złożone algorytmy. Kolejnym przykładem może być wyznaczenie szybkiej transformaty Fouriera FFT lub wyszukiwanie określonych wzorców w szybkozmennym sygnale cyfrowym w przypadku, gdy sygnał mierzony ma częstotliwość np. o połowę niższą niż częstotliwość pracy CPU, może ono nie nadążać z jego śledzeniem, a już na pewno nie będzie w stanie robić nic więcej. I tutaj idealnym rozwiązaniem jest przerzucenie tego zadania na sprzętowo zrealizowany w FPGA algorytm, który w efekcie tylko poinformuje mikroprocesor, gdy wykryty zostanie określony wzorec sygnału.

Kolejną zaletą jest integracja w jednym układzie systemu zdolnego przetwarzać

dane, z przetwornikami A/C i C/A, jak również szeregu wejść i wyjść cyfrowych oraz interfejsów komunikacyjnych. Pozwala to za pomocą niewielu zewnętrznych elementów budować kompletne systemy. Warto tutaj wspomnieć, że na złączach dostępne są napięcia niezbędne do zasilania dodatkowych układów wejściowych, również wzmacniaczy wymagających napięcia symetrycznego ± 15 V.

Nie bez znaczenia jest wyposażenie sterownika w interfejs Wi-Fi. Dzięki niemu możliwe jest swobodne programowanie systemu bez konieczności przyłączenia go do komputera i zdalne sterowanie budowanych aplikacjami, co może być szczególnie ważne dla kontrolowanych robotów czy innych urządzeń poruszających się samodzielnie. Tutaj może też się przydać wbudowany w układ akcelerometru, pozwalający wyznaczyć orientację w przestrzeni.

System operacyjny

MyRIO pracuje pod kontrolą systemu operacyjnego czasu rzeczywistego, NI Linux Real-Time mającego mechanizmy umożliwiające wyróżnienie deterministycznych zadań. Dzięki systemowi priorytetów zapewnia on wielowątkowość przy zachowaniu determinizmu, pozwala na wyznaczenie wątku

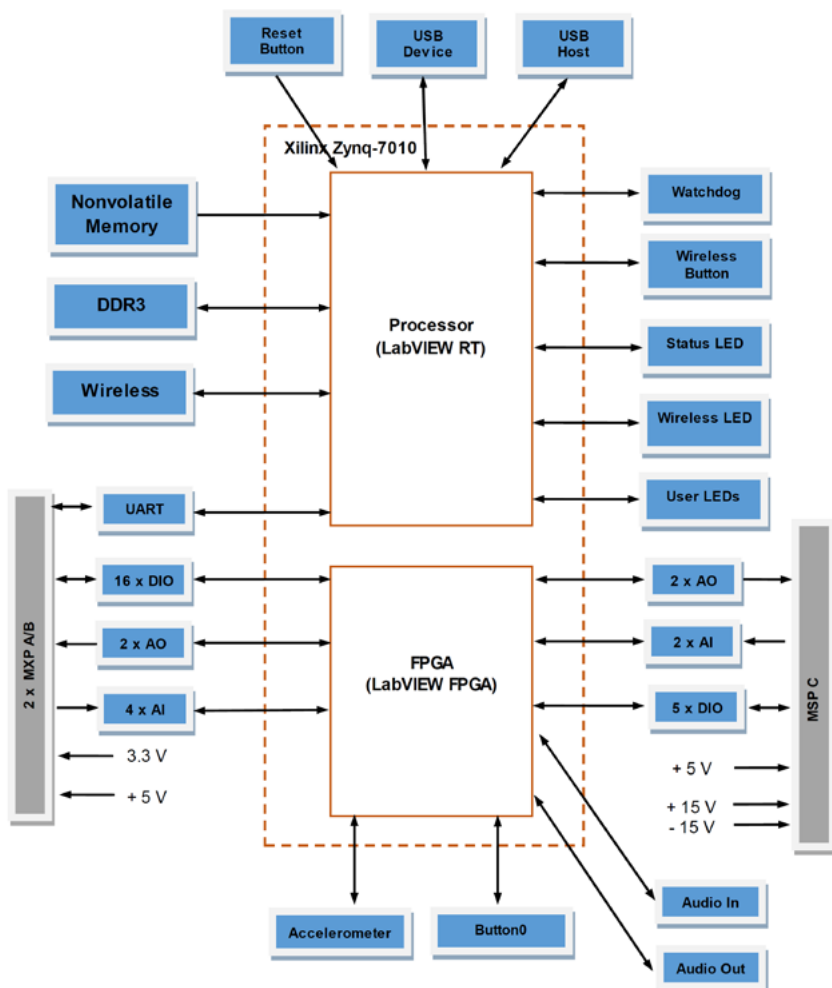
o najwyższym priorytecie, co gwarantuje wykonanie go w pierwszej kolejności. Zazwyczaj powinien być tylko jeden taki wątek, mimo iż wielordzeniowe procesory umożliwiają równoległe wykonywanie kilku wątków. Planując aplikację czasu rzeczywistego należy zwrócić uwagę na inne czynniki, takie jak współdzielone zasoby systemu np. pamięć, sterowniki, porty wejścia wyjścia i interfejsy komunikacyjne. Nieuwzględnienie wszystkich czynników może doprowadzić do sytuacji, w której zadanie o wyższym priorytecie będzie czekało na zwolnienie zasobów przez zadanie o niższym priorytecie. NI Linux Real-Time ma wbudowany mechanizm, który dobrze radzi sobie z takimi sytuacjami, ale dobrą praktyką jest ich unikanie.

Konfigurowanie systemu jest możliwe za pomocą programu *Measurement & Automation Explorer* lub przeglądarki internetowej. Sterownik może pracować niezależnie lub pod kontrolą aplikacji przygotowanej w LabView.

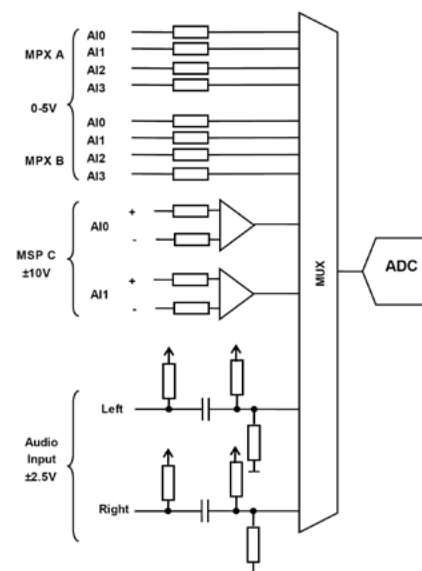
Komunikacja z otoczeniem

Oprócz interfejsów dostępnych w portach MPX i przeznaczonych głównie do przesyłania danych pomiędzy elementami systemu pomiarowego, myRIO wyposażono również w dwa porty USB oraz wspomniany moduł Wi-Fi.

Interfejs *USB device* umożliwia dołączenie sterownika do komputera. To za jego pomocą jest możliwy dostęp do systemu, konfigurowanie, wgrzywanie aplikacji, praca pod kontrolą LabView i konfigurowanie wbudowanego bloku FPGA. Interfejs *USB host* pozwala na dołączenie urządzenia typu *device*, np. kamery do akwizycji obrazu lub pamięci masowej służącej do gromadzenia danych pomiarowych. Interfejs Wi-Fi docenia wszyscy budujący samobieżne roboty lub pojazdy, które nie mogą być ograniczone przewodem łączącym je z jednostką sterującą. Ponadto,



Rysunek 2 Schemat blokowy sterownika myRIO



Rysunek 3. Struktura wejść analogowych

tak jak przez USB, jest możliwe wgrzywanie i uruchamianie aplikacji pod kontrolą LabView oraz konfigurowanie sterownika, co znacznie poprawia komfort pracy. Moduł może również pełnić rolę Routera.

Analogowe układy wejściowe

Tor pomiarowy jest klasycznym rozwiązaniem, w którym jeden przetwornik próbkuje sygnał w wielu kanałach jego strukturę przedstawiono na **rysunku 3**. Zastosowany przetwornik próbkuje sygnał wejściowy z rozdzielczością 12 bitów i maksymalną częstotliwością wynoszącą 500 kS/s.

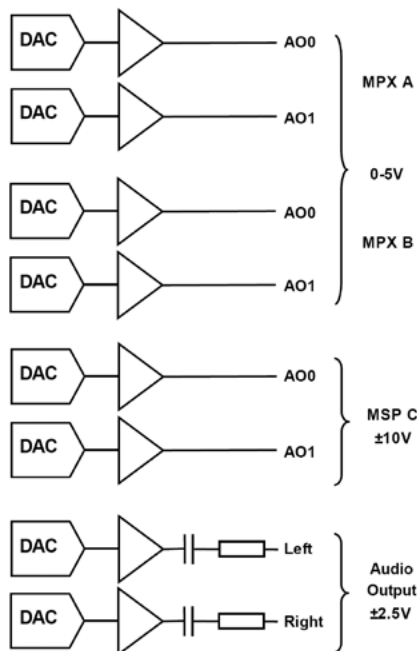
Układy wejściowe doprowadzające sygnały z portów MPX akceptują napięcia z przedziału 0...5 V, co daje rozdzielczość napięciową wynoszącą 1,221 mV. Minimalne pasmo przenoszenia dla tych wejść wynosi 300 kHz.

Sygnały wejściowe z portu MSP trafiają na wzmacniacze różnicowe akceptujące napięcia wejściowe o wartości ± 10 V. Rozdzielczość, z którą możemy mierzyć te napięcia wynosi 4,883 mV. Typowo pasmo przenoszenia wynosi 50 kHz.

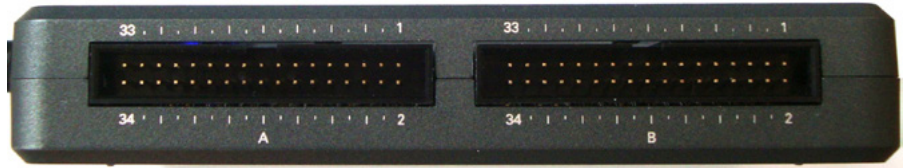
Dopuszczalny zakres napięcia dla wejść audio wynosi $\pm 2,5$ V przy rozdzielczości napięciowej 1,221 mV, natomiast pasmo przenoszenia zawiera się w przedziale 2 Hz...20 kHz.

Analogowe układy wyjściowe

Budowę analogowego toru wyjściowego ilustruje **rysunek 4**. Tor jest utworzony z 8 przetworników C/A o rozdzielczości 12 bitów i szybkości konwersji 345 kS/s. Każdy przetwornik współpracuje z wzmacniaczem wyjściowym zwiększającym jego wydajność prądową. Sygnały wyjściowe doprowadzone do portów MXP mogą przyjmować wartości z przedziału 0...5 V z krokiem 1,221 mV. Port



Rysunek 4. Struktura wyjść analogowych



Rozmieszczenie sygnałów w portach MPX

	+3.3V	DIO10 / PWM2	DIO9 / PWM1	DIO8 / PWM0	DIO7 / SPI.MOSI	DIO6 / SPI.MISO	DIO5 / SPI.CLK	DIO4	DIO3	DIO2	DIO1	DIO0	A13	A12	A11	A10	+5V
33	31	29	27	25	23	21	19	17	15	13	11	9	7	5	3	1	
34	32	30	28	26	24	22	20	18	16	14	12	10	8	6	4	2	
DIO15 / I2C.SDA	DIO14 / I2C.SCL	DGND	DGND	DIO13	DGND	DIO12 / ENC.B	DGND	DIO11 / ENCA	DGND	UART.TX	DGND	UART.RX	DGND	AGND	A01	A00	

Fotografia 5. Porty MPX A i MPX B

MSP udostępnia wyjścia bipolarne o napięciu ± 10 V i rozdzielczości 4,883 mV.

Wyjścia audio zapewniają pasmo przenoszenia do 50 kHz. Impedancję wyjściową stanowi szeregowe połączenie rezystora 100 Ω z kondensatorem 22 μ F.

Porty MPX

Użytkownikowi oddano do dyspozycji dwa porty rozszerzeń MPX A i MPX B. Ich sygnały zostały wyprowadzone na 2-rzędowe złącze o rastrze 2,54 mm i 34 wyprowadzeniach. Rozmieszczenie sygnałów jest identyczne na obu złączach. Tabela na **rysunku 5** przedstawiona numery wyprowadzeń z przypisanymi sygnałami portu MPX. Każdy port udostępnia: 4 wejścia analogowe, 2 wyjścia analogowe, interfejs UART, do 16 linii cyfrowego wejścia wyjścia, 3 sygnały PWM, interfejs SPI, interfejs I²C, wejścia enkodera, napięcia zasilające 3,3 V i 5 V.

Port MSP

Port MSP został wyprowadzony na 1-rzędowe złącze o rastrze 3,8 mm. W zestawie znajdziemy pasujące do niego złącze z zaciskami śrubowymi. Dostępne sygnały analogowe mogą przyjmować wartości z przedziału ± 10 V. Oprócz sygnałów analogowych i cyfrowych, dostępne są również napięcia zasilające +15 V, -15 V i +5 V. Wydajność prądowa źródeł napięcia jest nieduża, ale z pewnością wystarczy do zasilenia dodatkowego obwodu wejściowego zbudowanego z kilku wzmacniaczy operacyjnych bez konieczności martwienia się o zasilacz, co jest ważną zaletą. Tabela na **rysunku 6** opisuje szczegółowe rozmieszczenie sygnałów w złączu MSP. W porcie MSP C są dostępne: napięcia zasilające ± 15 V (max 32 mA) oraz 5 V, 2 wyjścia analogowe ± 10 V, 2 wejścia analogowe sygnału różnicowego ± 10 V, 8 wejść wyjść cyfrowych,

2 wyjścia PWM, 2 wejścia enkoderów. Obok złącza MSP C jest dostępne również wejście i wyjście sygnału audio.

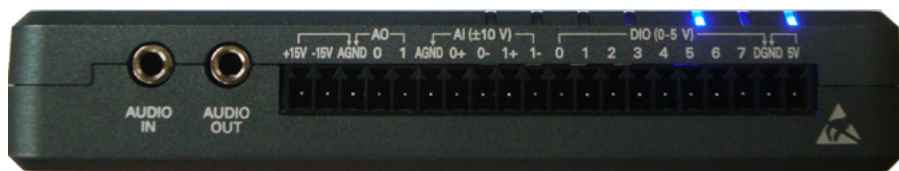
Programowanie

Jak wszystkie produkty National Instruments, również myRIO jest programowany za pomocą środowiska LabView. Pozwala ono na wykorzystanie zalet graficznego języka programowania do przygotowania aplikacji dla systemu czasu rzeczywistego i konfigurowania struktury FPGA, bez konieczności zgłębiania szczegółów konstrukcyjnych urządzenia. Nie wymaga przy tym posiadania specjalistycznej wiedzy i znajomości specjalistycznych języków opisu sprzętu, takich jak VHDL czy Verilog. Dzięki temu ta technologia jest dostępna praktycznie dla każdego. Programy przygotowane w LabView są przetwarzane na język C, a następnie wywoływane są narzędzia Xilinx, które generują pliki wynikowe wgrzywane później do myRIO. Wszystko to dzieje się automatycznie – z punktu widzenia programisty jest wystarczające uruchomienie pojedynczego programu. Oczywiście, myRIO można też programować za pomocą języka C/C++ przy wykorzystaniu środowiska opartego na popularnym Eclipse. Opis sposobu konfiguracji tego środowiska można znaleźć na stronie National Instruments.

Wraz ze sterownikiem jest dostarczane oprogramowanie LabView na licencji studenckiej, pozwalającej na korzystanie z niego w celach edukacyjnych. Oprócz podstawowej wersji LabView przeznaczonej dla myRIO, możemy zainstalować kilka przydatnych modułów.

Moduły niezbędne do obsługi wszystkich elementów architektury myRIO to:

- Real – Time Module – pozwala przygotować aplikację czasu rzeczywistego wbudowaną w system wykonawczy.
- FPGA Module – moduł graficznego programowania struktur FPGA, program



Rozmieszczenie sygnałów w porcie MSP

+15V	-15V	AGND	A00	A01	AGND	AIO+	AIO-	A11+	A11-	DIO0/ENCO.A	DIO1	DIO2/ENCO.B	DIO3/PWM0	DIO4/ENC1.A	DIO5	DIO6/ENC1.B	DIO7/PWM1	DGND	5V	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	

Fotografia 6. Widok układu od strony złącza MSP

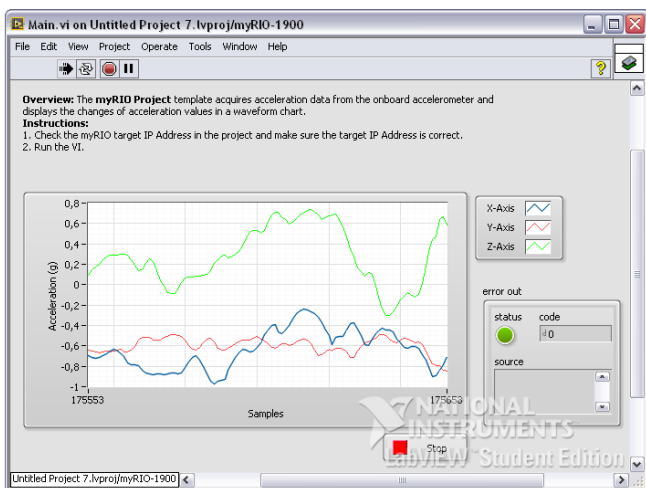
przygotowany w nim jest kompilowany i wgrany do struktury FPGA.

Dodatkowe moduły dostępne na licencji studenckiej wraz z LabView myRIO:

- Vision Development Module – zespół funkcji wspomagających przetwarzanie obrazu.
- LabVIEW Control Design and Simulation Module – zbiór funkcji do budowy projektowania i symulowania pracy regulatorów.
- LabVIEW MathScript RT Module – pozwala korzystać w LabView z plików Matlaba, pod rygiem systemu czasu rzeczywistego.
- LabVIEW Robotnic Module – zbiór funkcji wspomagających projektowanie układów sterowania robotami.

Pierwszy Program

Pracę z myRIO możemy zacząć bardzo szybko. Po zainstalowaniu oprogramowania i dołączeniu układu do portu USB komputera na ekranie zostanie wyświetlone okno monitora (rysunek 7), z którego możemy przejść wprost do konfigurowania naszego układu wybierając z menu *Configure NI myRIO*. Następnie, wybierając *Launch the I/O Monitor* za pomocą myszki możemy testować wszystkie wyjścia, odczytać wartości napięcia na wejściach analogowych czy wygenerować przykładowy przebieg. Możliwe jest również przesłanie danych poprzez dostępne interfejsy.



Rysunek 8. Aplikacja testowa

Wybierając *Launch the Getting Started Wizard* przejdziemy do panelu testowego pokazanego na **rysunku 8**. Możemy obserwować na nim wskazania akcelerometru i sterować diodami. Przechodząc do LabView należy wybrać *Create Project* → *myRIO* → *myRIO Project* i przechodząc kolejne etapy wygenerować przykładowy projekt pobierający dane z akcelerometru i wyświetlający je na wykresie. Panel czołowy aplikacji pokazano na **rysunku 9**. Wystarczy wcisnąć *Run*, poczekać aż zakończy się proces kompilacji, Wygenerowany kod zostanie przesłany i uruchomiony w myRIO. Na jego podstawie można zacząć tworzyć własne projekty.

Podsumowanie

MyRIO jest dobrą platformą treningową, pozwalającą przećwiczenie wielu zagadnień z dziedziny automatyki, robotyki i przetwarzania sygnałów, jednocześnie pokazując aspekty związane z przygotowaniem systemów czasu rzeczywistego. Studenci zapoznawszy się z platformą nie będą mieli kłopotów podczas pracy z systemem CompactRIO. Programowanie w LabView pozwala na budowanie systemów sterujących, nie wymagając przy tym od studentów znajomości mikrokontrolerów lub innych języków programowania. Dzięki temu można skupić się na faktycznych zagadnieniach związanych z tematyką przedmiotu. Mimo że jest to układ przeznaczony do nauki, jego możliwości są bardzo



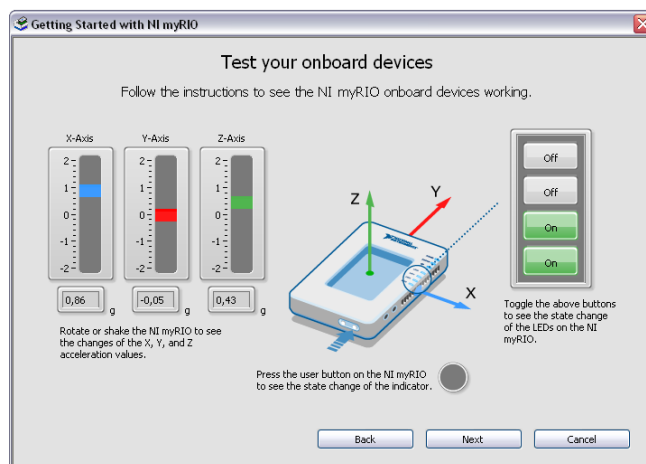
Rysunek 7. Okno monitora myRIO

duże. Praktycznie w większości wypadków rozdzielczość przetworników jest wystarczająca, wydajność procesora często nie będzie w pełni wykorzystana, nawet przy złożonych aplikacjach. Takie zasoby w połączeniu modulem FPGA dają dużo swobody i pozwalają na postawienie przed studentami znacznie trudniejszych zadań.

Firma National Instruments proponuje również zestawy montażowe pozwalające na zapoznanie się z różnymi zagadnieniami. Na stronie internetowej znajdziemy przykłady ćwiczeń laboratoryjnych wraz z instrukcjami. Oprócz zastosowań czysto akademickich, dla programisty LabView jest to doskonały przyrząd warsztatowy, pozwalający na wykonaniu dosłownie „na oczekaniu” brakującego elementu systemu pomiarowego. Może służyć jako oscyloskop, rejestrator i analizator poziomów logicznych, a nawet zastąpić mikroprocesor sterujący. Algorytmy sterujące można wygodnie przetestować z poziomu LabView obserwując przebiegi w różnych częściach układu. Po upewnieniu się, że wszystko będzie działało poprawnie, algorytm może być przeniesiony na docelowy mikroprocesor.

W kolejnym artykule opiszę sposób wykonania uniwersalnego przyrządu warsztatowego wykorzystując przy tym zasoby dostępne w myRIO, a następnie system pomiarowy sterowany za pomocą przeglądarki internetowej.

Wiesław Szaj
wszaj@prz.edu.pl



Rysunek 9. Panel czołowy automatycznie generowanego projektu