

Zastosowanie mikrokontrolera STM32F410 do prezentacji zagadnień cyfrowego przetwarzania sygnałów

Streszczenie. W artykule przedstawiono zastosowanie modułu STM32F410 Nucleo do prezentacji działania algorytmów cyfrowego przetwarzania sygnałów (CPS, ang. DSP) w czasie rzeczywistym. W celu ułatwienia korzystania z modułu zaprojektowano dodatkową płytkę PCB (tzw. shield), posiadającą gniazda we/wy oraz układy wzmacniaczy. Przygotowany zestaw (moduł Nucleo + shield) umożliwia badania eksperymentalne z użyciem generatora sygnałowego i oscyloskopu lub systemu Analog Discovery 2. Do celów edukacyjnych opracowano zestaw instrukcji laboratoryjnych oraz oprogramowanie, które prezentują operacje cyfrowej generacji sygnałów, filtracji oraz obliczeń dyskretnej transformaty Fouriera za pomocą technik bezpośrednich, FFT i za pomocą algorytmu Goertzela.

Abstract. In this paper the use of the STM32F410 Nucleo module for presentation of the operation of digital signal processing algorithms (DSP) is discussed. In order to facilitate the use of the module, an additional PCB module (shield) has been designed, having I/O sockets and amplifier circuits. The Nucleo board with the shield enables experimental research using a signal generator and an oscilloscope or Analog Discovery 2 system. For educational purposes, a set of laboratory instructions and software has been developed, which present basic operations of digital signal generation, filtering and discrete Fourier transformation calculations using direct techniques, FFT, and the Goertzel algorithm. (**Application of STM32F410 microcontroller for presentation of digital signal processing**).

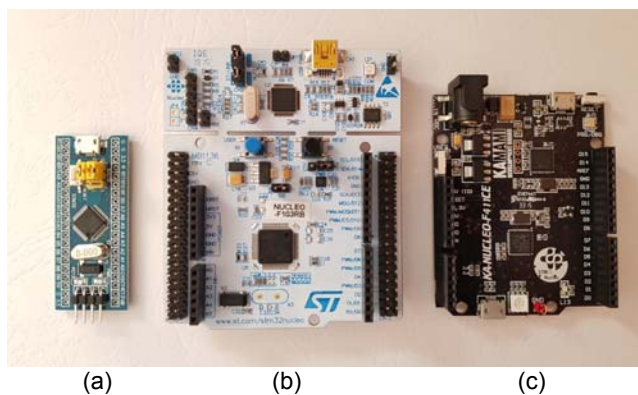
Słowa kluczowe: cyfrowe przetwarzanie sygnałów, CPS, filtry cyfrowe, STM32.
Keywords: digital signal processing, DSP, digital filters, STM32.

Wprowadzenie

Współczesne mikrokontrolery posiadają aktualnie wiele cech dostępnych wcześniej jedynie w procesorach sygnałowych. Przykładem takich mikrokontrolerów są produkty firmy STMicroelectronics, która bazując na rdzeniach ARM oferuje dwie interesujące rodziny o oznaczeniach STM32F1 i STM32F4, wykorzystujące odpowiednio rdzenie ARM: Cortex-M3 i Cortex-M4 [1, 2]. Do zalet rdzenia M4 (w stosunku do M3) należy zaliczyć m.in.:

- jednostkę obliczeniową MAC (Multiply and Accumulate)
- instrukcje SIMD (ang. Single Instruction, Multiple Data)
- zmiennoprzecinkowy koprocesor arytmetyczny FPU (ang. Floating-Point Unit).

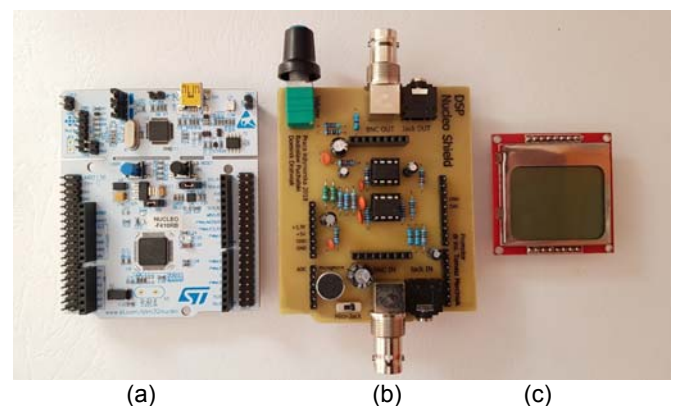
Te cechy przekładają się na większą wydajność w typowych algorytmach DSP, takich jak filtracja FIR i IIR, obliczenia FFT, regulator PID, operacje na macierzach (dodawanie, odejmowanie, mnożenie) [3, 4]. Atrakcyjną cechą wymienionych rodzin jest także 32-bitowa reprezentacja danych, która pozwala na uzyskanie dużego zakresu dynamicznego.



Rys. 1. Moduły z mikrokontrolerami (a) STM32F103RC8T6, (b) STM32F103RB, (c) STM32F411CE

Popularność rozwiązań STM wynika m.in. z dostępności szeregu modułów eksperymentalnych, które są zbliżone

funkcjonalnie do modułów Arduino [5]. Na rys. 2 pokazano kilka przykładowych modułów z mikrokontrolerami STM32. Najtańszym rozwiązaniem jest STM32F103RC8T6 (rys. 1a), który kosztuje zaledwie kilka dolarów, przy czym do zaprogramowania wymaga dołączenia dodatkowego programatora/debugera. Na wielu kursach internetowych (np. [6]), popularnym układem jest STM32F103RB Nucleo (rys. 1b) ze zintegrowanym programatorem i wieloma pinami we/wy. Atrakcyjnym rozwiązaniem z rdzeniem Cortex-M4 jest moduł KA-NUCLEO-F411 (rys. 1c) zawierający dodatkowe elementy jak dioda RGB [7]. W celach dydaktycznych DSP można wykorzystać moduł STM32F407 z modułem Wolfson Audio [8, 9].



Rys. 2. Moduł STM32F410 (a), płytki rozszerzająca (b) oraz wyświetlacz (c)

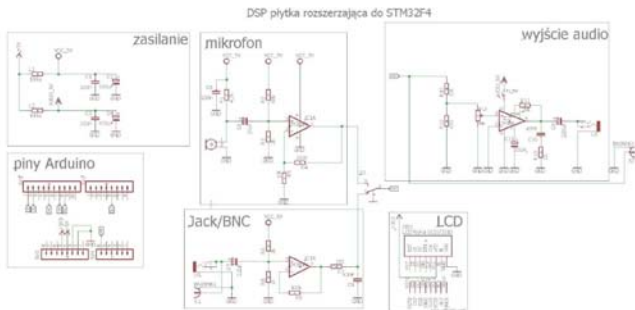
Moduł STM32F410 z shieldem jako system DSP

Do opracowania ćwiczeń laboratoryjnych z zakresu DSP wybrano moduł STM32F410 Nucleo (rys. 2a), który oprócz rdzenia M4, zawiera przetwornik cyfrowo-analogowy, umożliwiający odtworzenie sygnału analogowego na podstawie próbek przetworzonych przez algorytm DSP. Mikrokontroler STM32F410 charakteryzuje się następującymi cechami [11]:

- zasilanie 1,7 – 3,6 V
- częstotliwość 100 MHz

- pamięć flash 128 kB
- pamięć OTP 512 B
- pamięć SRAM 32 kB
- 12-bitowy ADC - 16 kanałów
- 12-bitowy DAC
- 50 portów wejścia-wyjścia
- interfejsy komunikacyjne: I2C, USART, SPI/I2C.

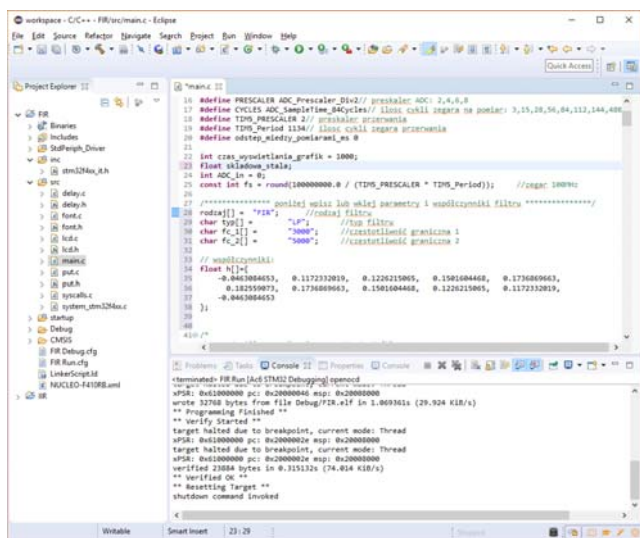
Moduł jest wyposażony w złącza rozszerzeń kompatybilne z Arduino Uno R3 oraz z ST Morpho. Moduł zawiera także programator/debugger zgodny z ST-Link v.2.1 [12].



Rys. 3. Schemat płytki rozszerzającej [13]

Dla modułu STM32F410 Nucleo zaprojektowano płytkę rozszerzającą (tzw. shield) ułatwiającą korzystanie z takiego zestawu (rys. 2b oraz rys. 3) [13].

Płytkę rozszerzającą posiada gniazda BNC, które ułatwiają dołączenie generatora sygnałowego oraz oscyloskopu. Zastosowano także inne elementy, pozwalające wykorzystać dodatkowe źródła sygnału, bądź obsługując wyniki jego przetwarzania bez konieczności podłączania specjalizowanego sprzętu.

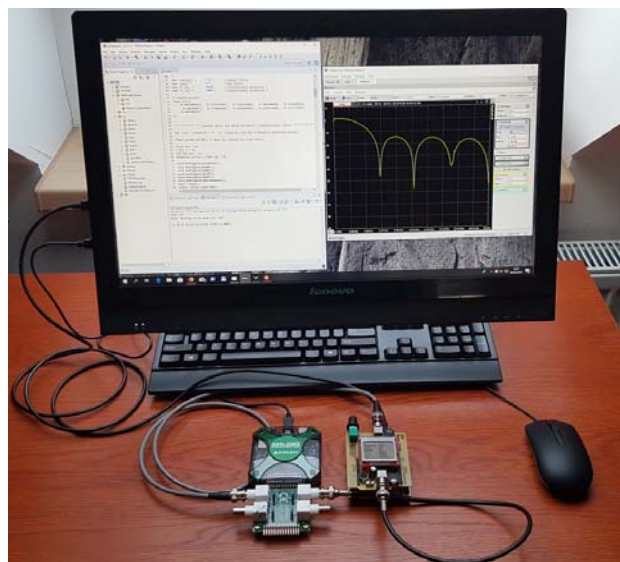


Rys. 4. Środowisko System Workbench

Zgodnie z przyjętymi założeniami, shield posiada do wyboru dwa źródła sygnału analogowego. Pierwszym z nich jest wbudowany w płytkę rozszerzającą mikrofon elektretowy wraz z układem wzmacniacza (LM358), natomiast drugie źródło to złącze Mini-Jack 3,5 mm, do którego można łatwo podłączyć sygnał z karty dźwiękowej komputera PC. Wybór źródła sygnału jest realizowany za pomocą przełącznika. Sygnał na wyjściu przetwornika cyfrowo-analogowego mikrokontrolera jest wygładzany za pomocą filtra RC, a następnie podawany na wejście wzmacniacza audio (LM386). Na wyjściu wzmacniacza audio zamontowano złącze Mini-Jack 3,5 mm, aby w łatwy sposób można było podłączyć głośnik w celu odsłuchania przetworzonych danych wejściowych. Shield zawiera także złącza do modułu wyświetlacza Nokia (rys. 2c), który

pozwała wyświetlać dodatkowe komunikaty, związane z działaniem określonego programu DSP.

Do programowania modułu wybrano narzędzie System Workbench for STM32 (SW4STM32), które jest środowiskiem przygotowanym na bazie Eclipse (rys. 4) i może być uruchomione na wielu systemach oraz obsługuje układy z rodziny Nucleo, Discovery oraz płytki ewaluacyjne [14]. Podczas programowania w tym narzędziu możemy korzystać z biblioteki Standard Peripheral Library lub z STM32Cube HAL. Oprogramowanie posiada wbudowany kompilator oraz debugger języków C lub C++. Ważną zaletą jest również brak ograniczenia rozmiaru kodu.

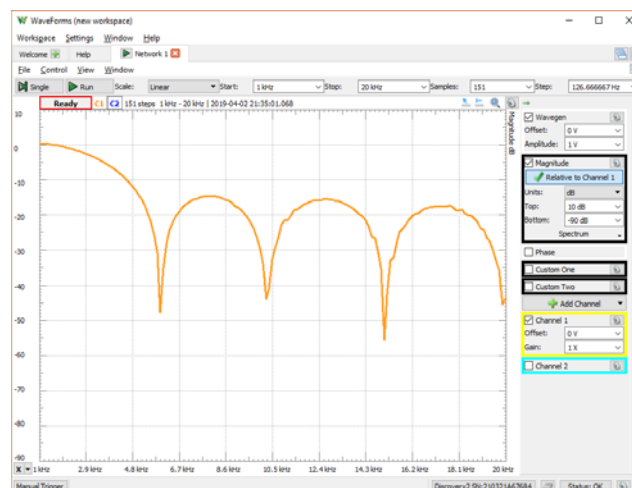


Rys. 5. Analog Discovery 2 wykorzystane do badania poprawności działania filtrów cyfrowych

Ćwiczenia DSP z wykorzystaniem modułu STM32F410

Moduł STM32F410 z założonym shieldem ułatwia badanie typowych algorytmów cyfrowego przetwarzania sygnałów [9]. W tym celu opracowano zestaw instrukcji laboratoryjnych oraz programów obejmujących:

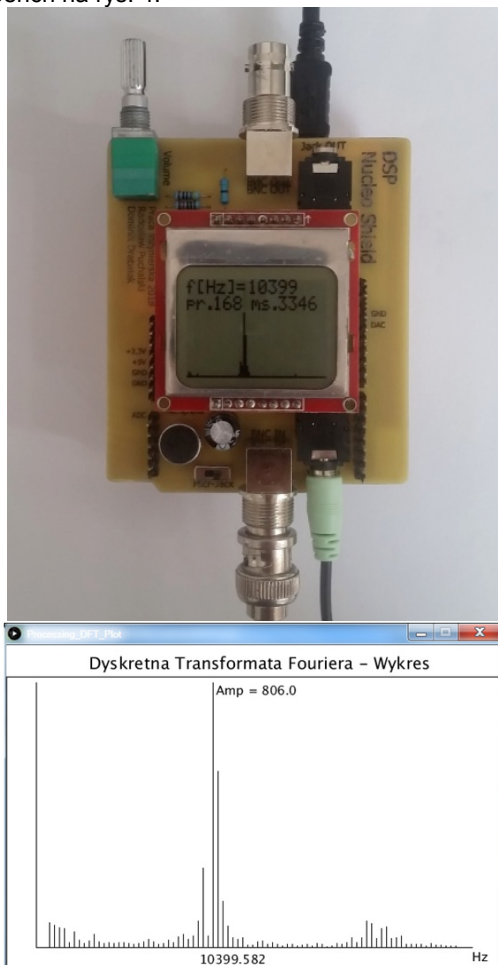
- konwersję A/C i C/A – podstawowy program do akwizycji sygnału analogowego i jego odtworzenia na podstawie próbek
- filtrację typu FIR (o skończonej odpowiedzi impulsowej) oraz IIR (o nieskończonej odpowiedzi impulsowej)
- analizę widmową DFT za pomocą algorytmu FFT
- analizę określonych częstotliwości za pomocą algorytmu Goertzela.



Rys. 6. Przykładowa charakterystyka częstotliwościowa filtra zbadana z wykorzystaniem Analog Discovery 2

Programy do konwersji A/C i C/A można wykorzystać do obserwacji zjawiska aliasingu, a także do zademonstrowania możliwości generacji sygnałów za pomocą różnych technik (np. tablicy look-up-table czy zastosowania generatora IIR).

Przygotowanie oprogramowania do filtracji sygnałów technikami FIR i IIR, wymaga uzupełnienia programu dla mikrokontrolera o wartości współczynników (rys. 4), które można wyznaczyć za pomocą np. oprogramowania Filter Designer w środowisku Matlab. Wygodnym urządzeniem do sprawdzenia poprawności działania filtra jest wykorzystanie Analog Discovery 2 i funkcji Network Analyzer (rys. 5) [15]. Na rys. 6 pokazano zbadaną charakterystykę filtra FIR o współczynnikach widocznych w oknie main.c środowiska Workbench na rys. 4.



Rys. 7. Przykład analizy DFT – widok na wyświetlaczu oraz wizualizacja w programie Processing

Analiza DFT może być wizualizowana na ekranie płytki rozszerzającej lub przy wykorzystaniu środowiska Processing v3.3.7 (rys. 7). Jest to środowisko, które w łatwy sposób pozwala opracować aplikacje okienkowe. W przypadku algorytmu Goertzela [16], wśród ośmiu ustalonych przez użytkownika częstotliwości, przygotowany program wyszukuje jedną lub dwie dominujące częstotliwości (zgodnie ze standardem sygnalizacji tonowej DTMF).

Podsumowanie

W artykule przedstawiono możliwości wykorzystania modułu STM32F410 Nucleo do realizacji algorytmów cyfrowego przetwarzania sygnałów działających w czasie rzeczywistym. Mikrokontroler STM32F410 posiada przetwornik cyfrowo-analogowy, co ułatwia badania

eksperymentalne i obserwację pełnego procesu przetwarzania sygnału analogowego. Ułatwieniem w korzystaniu z takiego rozwiązania jest opracowana płytka rozszerzająca.

Coraz większa popularność rozwiązań STM, stosunkowo nieduży koszt takich modułów oraz dostępność środowisk programistycznych, daje możliwość samodzielnego przeprowadzania eksperymentów także w warunkach poza laboratorium. Atrakcyjnym urządzeniem do przeprowadzania eksperymentów jest Analog Discovery 2, czyli wielofunkcyjny przyrząd do pomiarów, wizualizacji, generowania i rejestrowania sygnałów.

Praca powstała w wyniku realizacji działalności statutowej DS-2019.

Autorzy: dr inż. Tomasz Marciniak, prof. dr hab. inż. Adam Dąbrowski, Politechnika Poznańska, Wydział Informatyki, Instytut Automatyki i Robotyki, Zakład Układów Elektronicznych i Przetwarzania Sygnałów, ul. Piotrowo 3a, 60-965 Poznań, E-mail: tomasz.marciniak@put.poznan.pl ; inż. Radosław Puchalski, inż. Dominik Dratwiak, Wojciech Marciniak, Politechnika Poznańska, Wydział Informatyki, Studenckie Koło Naukowe Decybel, ul. Jana Pawła II nr 24, 60-965 Poznań, E-mail: wojciech.to.marciniak@student.put.poznan.pl.

LITERATURA

- [1] STMicroelectronics, *Datasheet - production data, STM32F103x8 STM32F103xB* (2015)
- [2] STMicroelectronics, *Reference manual, RM0401, STM32F410 advanced ARM - based 32-bit MCUs* (2015)
- [3] STMicroelectronics, *Application note, AN4904, Migration of microcontroller applications from STM32F1 to STM32F4 Access lines, Series* (2016)
- [4] D.S. Reay, *Digital Signal Processing Using the ARM Cortex-M4*, Wiley (2016)
- [5] R. Anderson, D. Cervo, *Arduino dla zaawansowanych*, Helion (2014)
- [6] Forbot, *Kurs STM32*, <https://forbot.pl/blog/stm32-praktyce-1-platforma-srodowisko-id2733>
- [7] A. Kurczyk, *Mikrokontrolery STM32 dla początkujących*, Wydawnictwo BTC (2019)
- [8] *STM32F4DISCOVERY Discovery kit with STM32F407VG MCU* (2016), <https://www.st.com/en/evaluation-tools/stm32f4discovery.html>
- [9] T. Marciniak, A. Dąbrowski, Nauczanie zagadnień cyfrowego przetwarzania sygnałów z zastosowaniem modułów z mikrokontrolerem, *Przegląd Elektrotechniczny*, R 94, Nr 9/2018, 125-127
- [11] STMicroelectronics, *Datasheet - production data, STM32F410x8 STM32F410xB* (2017)
- [12] *STM32 Nucleo-64 development board with STM32F410RB MCU*, <https://www.st.com/en/evaluation-tools/nucleo-f410rb.html>
- [13] R. Puchalski, D. Dratwiak, *Edukacyjny system cyfrowego przetwarzania sygnałów z wykorzystaniem modułu Nucleo*, praca dyplomowa inżynierska, promotor: T. Marciniak, Politechnika Poznańska (2018)
- [14] *System Workbench for STM32: free IDE on Windows, Linux and OS X*, <https://www.st.com/en/development-tools/sw4stm32.html>
- [15] *The Analog Discovery 2: A portable USB laboratory for everyone*, <https://analogdiscovery.com/>
- [16] A. Dąbrowski, T. Marciniak, Canonic Goertzel algorithm and drawbacks of various Goertzel algorithm formulations, *IEEE Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA) Conference* (2017), Poznań, 259-262