

Prototypowanie modułów mikroprocesorowych do wykrywania wzorców ruchowych

Streszczenie. W artykule przedstawiono rozwiązania mikroprocesorowych modułów elektronicznych umożliwiających eksperymentalne rozpoznawanie wzorców ruchowych na podstawie danych pochodzących z czujników inercyjnych. Pokazano rozwiązania oparte na mikrokontrolerze ATmega328P oraz bardziej zaawansowane z użyciem mikrokontrolerów STM32L476RG oraz STM32L4R9. Omówiono możliwości czujników, w tym najnowszych rozwiązań zawierających elementy uczenia maszynowego. Przedstawiono oprogramowanie wspomagające proces przygotowywania projektów.

Abstract. The paper presents solutions of microprocessor electronic modules that enable an experimental recognition of movement patterns on the basis of data from inertial sensors. Simple solutions based on the ATmega328P microcontroller and more advanced ones with the use of STM32L476RG and STM32L4R9 microcontrollers were shown. The possibilities of sensors were discussed, including the latest solutions containing elements of machine learning. The software supporting the project preparation process was also presented. (**Prototyping of microprocessor modules for human activity recognition**).

Słowa kluczowe: mikrokontrolery, czujniki inercyjne, rozpoznawanie wzorców ruchowych.

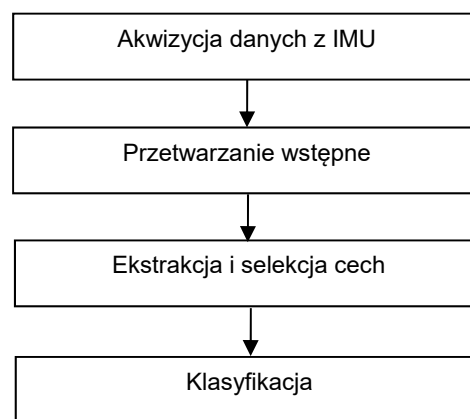
Keywords: microcontrollers, inertial sensors, human action recognition, HAR.

Wprowadzenie

Rozpoznawanie wzorców ruchowych jest jednym z dynamicznie rozwijających się zagadnień identyfikacji biometrycznej. Postęp technologiczny w ciągu ostatnich kilku lat sprawił, że urządzenia do monitorowania ruchu są tak małe, że jesteśmy w stanie umieścić je w smartfonach, czy w urządzeniach noszonych na nadgarstku lub nawet ukryć w odzieży. Skutkuje to rozwojem branży wearables, czyli urządzeń, które nie tylko przekonują entuzjastów technologii, ale również trafiają do osób, które cenią sobie poprawę jakości życia w aspekcie zdrowia. Badania pokazują, że wartość globalnego rynku technologii do noszenia wzrosła ze 116,2 mld USD w 2021 r. do 265,4 mln USD do 2026 r. Oczekuje się, że w latach 2021-2026 skumulowany roczny wskaźnik wzrostu (ang. compound annual growth rate, CAGR) wzrosła o 18,0% [1], który wynika głównie z rosnącego popytu na inteligentne urządzenia w sektorze opieki zdrowotnej.

Badane wzorce ruchowe można przyporządkować do kilku grup, przykładowo autorzy w [2] wyróżniają następujące grupy (i indywidualne ćwiczenia): ruch człowieka (chodzenie, bieganie, siedzenie, stanie w miejscu, leżenie, wchodzenie po schodach, schodzenie po schodach, jazda po schodach ruchomych i jazda windą), transport (jazda autobusem, jazda na rowerze i jazda samochodem), korzystanie z telefonu (wiadomości tekstowe, wykonywanie połączeń), codzienne czynności (jedzenie, picie, praca przy komputerze, oglądanie telewizji, czytanie, mycie zębów, rozciąganie, sprzątanie i odkurzanie), ćwiczenia sportowe (wiosłowanie, podnoszenie ciężarów, wirowanie, nordic walking i robienie pompek), wojskowe (czołganie się, kłęczenie, ocena sytuacji i otwieranie drzwi), górna część ciała (żucie, mówienie, potykanie, wzdychanie i poruszanie głową). Do badań nad skutecznością identyfikacji wykorzystywane są bazy danych, jednym z przykładów jest baza "SCUT-NAA dataset", która obejmuje następujące kategorie [3]: siedzenie i relaks, chód (wolny, normalny i szybki), bieg, skoki, wchodzenie po schodach, schodzenie po schodach, kolarstwo.

Proces rozpoznawania wzorca składa się z typowych elementów procesu klasyfikacji (rys. 1). Dane z czujnika IMU są wstępnie przetwarzane, następnie dokonywana jest ekstrakcja i wybór cech, a cały proces kończy się właściwą klasyfikacją.



Rys. 1. Proces rozpoznawania wzorca ruchowego

Obecnie do wykrywania wzorców ruchowych (ang. human activity recognition HAR) można wykorzystać niedrogie inercyjne układy pomiarowe (IMU) wykonane w technologii układów mikroelektromechanicznych (MEMS). Badania pokazały, że skuteczne rozpoznawanie wzorców ruchowych jest możliwe, jeżeli szybkość próbkowania jest nieco większa niż 20 próbek na sekundę, natomiast wystarczający zakres pomiaru przyspieszenia to $\pm 2g$ [4]. Układy scalone IMU można stosunkowo łatwo podłączyć do mikrokontrolerów i budować autonomiczne, energooszczędne systemy e-AI (wbudowana sztuczna inteligencja) wykorzystujące biometrię ruchu.

Na podstawie uzyskanych danych pomiarowych dokonywana jest ekstrakcja cech. Wyróżnia się trzy grupy takich cech [2]:

- czasowe: średnia, odchylenie standardowe, wariancja, rozstęp międzykwartylowy (IQR), średnie odchylenie bezwzględne (MAD), korelacja między osiami, entropia i kurtoza,
- częstotliwościowe: dyskretna transformata Fouriera (DFT), dyskretna transformata cosinusowa (DCT),
- przekształceniowe: analiza składowych głównych (PCA), liniowa analiza dyskryminacyjna (LDA), model autoregresyjny (AR), filtry falkowe.

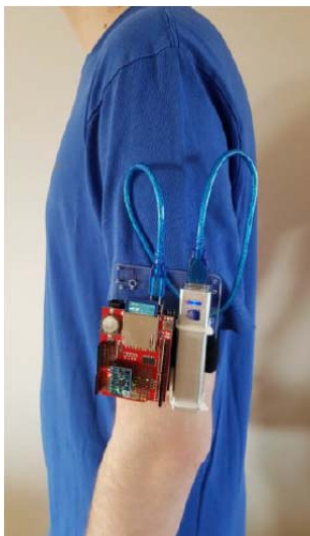
Etap klasyfikacji może wykorzystywać różne rozwiązania uczenia maszynowego [2]: drzewa decyzyjne, sieci

bayesowskie, algorytm k-najbliższych sąsiadów, maszyny wektorów nośnych, funkcje logiki rozmytej, regresję, ukryte modele Markowa (HMM) czy sieci neuronowe.

Rozwiązanie oparte na mikrokontrolerze ATmega328P

Mikrokontrolery takie jak ATmega328P (stosowane w popularnych układach Arduino UNO) [5,6] pozwalają na budowanie prostych systemów monitorowania. Na rysunku 2 przedstawiono prototyp urządzenia zawierającego:

- Arduino UNO - pełniące rolę mikrokontrolera,
- nakładkę (shield) z portem karty SD oraz zegarem czasu rzeczywistego (RTC),
- moduł MPU 6050 - zawierający trójosiowy akcelerometr i trójosiowy żyroskop [7],
- powerbank - pełniący rolę zasilacza.



Rys.2. Prototyp z wykorzystaniem Arduino UNO oraz shieldu z zegarem czasu rzeczywistego, kartą SD oraz IMU-6050

MPU-6050 umożliwia akwizycję 16-bitowych danych i charakteryzuje się następującymi parametrami:

- napięcie zasilania: 2, 3V - 3,4 V, pobór prądu: ok. 5 mA,
- interfejs komunikacyjny: I2C (TWI) - 400 kHz,
- akcelerometr: $\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$,
- żyroskop: ± 250 , ± 500 , ± 1000 , ± 2000 °/s.

Na podstawie przeprowadzonych eksperymentów opracowano zminiaturyzowany system (rys.3) [8], dodatkowo wyposażony w moduł Bluetooth BLE 4.0 HM-17, który pozwala na zdalną (nawet do 100m) komunikację za pomocą smartfona, dla którego aplikacja została opracowana w open-sourcowym wizualnym środowisku programistycznym - MIT App Inventor, które pozwala na tworzenie aplikacji w języku JAVA, bez konieczności znajomości tego języka [9].

Taka opaska elektroniczna umożliwia monitorowanie osób wymagających szczególnej opieki i wsparcia ze strony opiekunów. Opracowana opaska ma za zadanie powiadomić osobę opiekującą się użytkownikiem w przypadku upadku, co pozwoliłoby na szybszą reakcję i udzielenie odpowiedniej pomocy. Głównym założeniem było osiągnięcie jak największego zasięgu urządzenia przy minimalnych wymiarach i jak najmniejszym poborze mocy.

Należy zauważyć, że zaawansowane rozpoznawanie wzorców z zastosowaniem mikrokontrolera ATmega328P jest dość ograniczone, ale podejmowane są ciekawe badania związane z redukcją modeli. W artykule [10] pokazano nowatorski algorytm oparty na drzewie

decyzyjnym, nazwany Bonsai, do wydajnego przewidywania na urządzeniach IoT - takich jak te oparte na płycie Arduino Uno z 8-bitowym mikrokontrolerem ATmega328P pracującym z częstotliwością 16 MHz bez natywnej obsługi zmiennoprzecinkowej, 2 KB RAM i 32 KB pamięci flash. Bonsai zachowuje dokładność prognoz, jednocześnie minimalizując rozmiar modelu i koszty prognoz poprzez [10]: opracowanie modelu drzewa, który uczy się pojedynczego, płytkiego, rzadkiego drzewa z potężnymi węzłami; rzadkie rzutowanie wszystkich danych na niskowymiarową przestrzeń, w której drzewo jest uczone; oraz (c) wspólne uczenie się wszystkich parametrów drzewa i projekcji.



Rys.3. Prototyp opaski monitorującej na nadgarstek (IMU-6050)

Aktualnie popularnym, wchodzącym w skład rozwiązań Arduino, jest moduł Arduino Nano 33 BLE Sense, bazujący na mikrokontrolerze nRF52480 [11]. Moduł zawiera 9-osiową (3 osiowy żyroskop, akcelerometr i magnetometr) jednostkę IMU LSM9DS1 komunikującą się poprzez magistralę I2C. Dodatkowo moduł Arduino Nano BLE Sense zawiera układ bluetooth U - blox NINA B306, czujnik temperatury i ciśnienia LPS22HB, czujnik wilgotności HTS221, sensor natężenia światła, kolorów oraz gestów ADPS-9960, mikrofon cyfrowy MP34DT05, i układ kryptograficzny ATECC608A. Przy programowaniu tego modułu wykorzystywane są rozwiązania TensorFlow Lite [11].

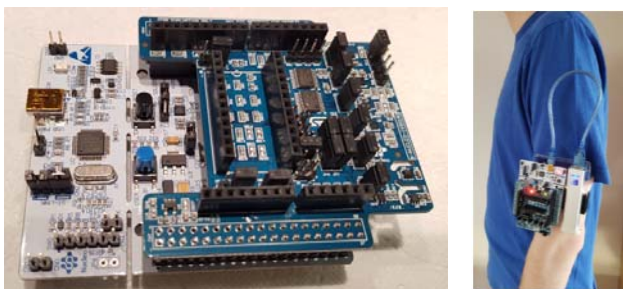
Rozwiązanie oparte na mikrokontrolerze STM32L476

Mikrokontrolery rodziny STM32L, bazujące na niskomocowych rdzeniach ARM Cortex-M4 z DSP i FPU, są zdecydowanie bardziej wydajne obliczeniowo od rozwiązań ATmega i pozwalają realizować zaawansowane algorytmy klasyfikacji wzorców. Dodatkowo firma ST przygotowała szereg zaawansowanych czujników IMU oraz modułów eksperymentalnych.

Przykładowym rozwiązaniem eksperymentalnym jest moduł Nucleo STM32L476 [12] z płytą rozszerzającą X-NUCLEO-IKS01A3 [13], która posiada m.in. inercyjną jednostkę pomiarową (IMU) LSM6DSO iNEMO 6DoF o podstawowych parametrach [14]:

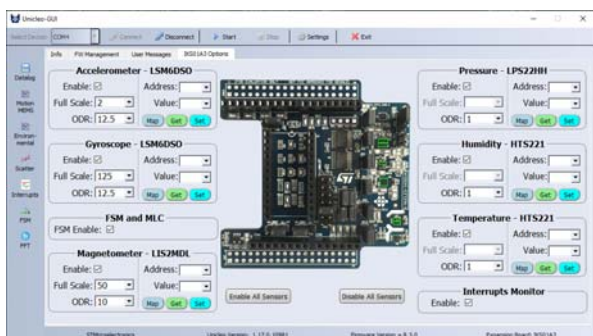
- pobór prądu: 0.55 mA,
- akcelerometr $\pm 2/\pm 4/\pm 8/\pm 16$ g,
- żyroskop $\pm 125/\pm 250/\pm 500/\pm 1000/\pm 2000$ °/s

Ciekawą cechą LS6DSO jest skończona maszyna stanów (ang. finite state machine FSM). Programowanie maszyny skończonej jest ułatwione dzięki dedykowanemu narzędziu Unico [15], które jest graficznym interfejsem użytkownika dla wybranych modułów opartych na STM32.

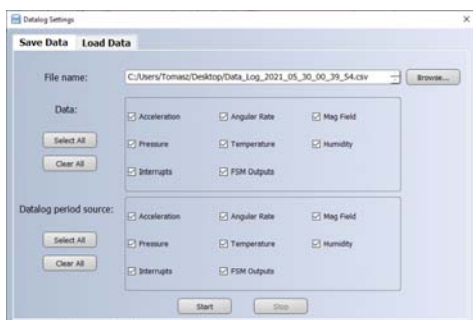


Rys. 4. Nucleo L476RG z dołączonym X-NUCLEO-IKS01A3

Programowanie algorytmów rozpoznawania wzorców ruchowych ułatwia pakiet AlgoBuilderSuite [16] w skład którego wchodzi AlgoBuilder, Unico-GUI oraz Unicleo GUI. Na rysunku 5 pokazano okno konfiguracji elementów wykorzystywanego shieldu X-NUCLEO-IKS01A3, natomiast rys. 6 przedstawia okno zapisu do pliku csv.



Rys. 5. Konfiguracja X-NUCLEO-IKS01A3 w Unicleo GUI



Rys. 6. Możliwość rejestracji sygnałów pomiarowych do pliku csv

Na rysunku 7 zamieszczono wizualizację danych pochodzących z akcelerometru, żyroskopu, magnetometru, a także pomiaru temperatury, wilgotności oraz ciśnienia.

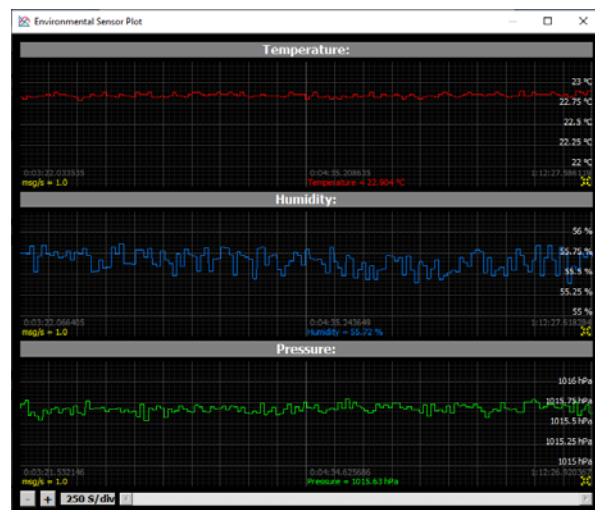
Narzędzia AlgoBuilderSuite pozwalają przygotować prototypy aplikacji dla mikrokontrolerów STM32 i czujników MEMS, wykorzystując już istniejące algorytmy (np. fuzję sensoryczną czujników), w łatwej do interpretacji bloków przetwarzania danych.

Moduł eksperymentalny SensorTile.box

Wygodniejszym rozwiązaniem do testów eksperymentalnych bazujących na STM32 jest moduł SensorTile.box [17] (rys. 8 i 9), który zawiera mikrokontroler STM32L4R9 do którego dołączono między innymi układ LSM6DSOX [18]. W porównaniu ze wspomnianym wcześniej LSM6DSO, układ LSM6DSOX umożliwia dodatkowo realizację algorytmu uczenia maszynowego w postaci ośmiu drzew decyzyjnych.

SensorTile.box jest wyposażony w szereg czujników (w tym dodatkowo mikrofon) oraz komunikację Bluetooth oraz zapis na kartę microSD. Niewielkie rozmiary (rys. 8) oraz autonomiczne zasilanie ułatwiają dokonywanie badań

eksperymentalnych. Firma ST przygotowała 3 możliwości używania modułu: podstawowy z predefiniowanymi funkcjonalnościami, tryb eksperta z możliwością definiowania nowych aplikacji oraz tryb profesjonalny pozwalający na pełną kontrolę programistyczną. Dwie pierwsze możliwości, dla ułatwienia, są dostępne z poziomu urządzenia typu smartfon.



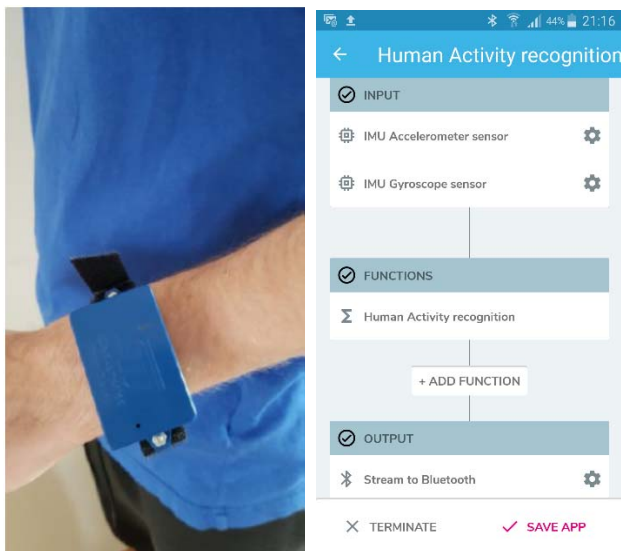
Rys. 7. Wizualizacja danych w oprogramowaniu Unicleo GUI



Rys. 8. Układ elektroniczny modułu SensorTile.box

Przy programowaniu modułu SensorTile.box można skorzystać z pakietu funkcji FP-AI-SENSING1 [19], zawierającego przykłady, które umożliwiają połączenie węzła IoT ze smartfonem za pośrednictwem BLE. Pakiet umożliwia budowanie zaawansowanych aplikacji, takich jak rozpoznawanie aktywności człowieka czy klasyfikacja scen dźwiękowych na podstawie wyjącej generowanych przez sieci neuronowe (NN). NN są implementowane przez wielosieczną bibliotekę obsługującą zarówno arytmetykę

zmiennoprzecinkową, jak i stałoprzecinkową, generowaną przez rozszerzenie X-CUBE-AI [20] dla narzędzia STM32CubeMX.



Rys.9. SensorTile.box umiejscowiony na nadgarstku podczas eksperymentu oraz ekran konfiguracji aplikacji

Podsumowanie

W artykule pokazano wybrane możliwości prototypowania rozwiązań pozwalających na klasyfikację wzorców ruchowych (HAR). Można zauważyć, że współczesne mikrokontrolery umożliwiają realizację algorytmów cyfrowego przetwarzania sygnałów [21] a także algorytmów uczenia maszynowego, pozwalając na efektywne obliczenia krawędziowe (edge computing).

Czujniki inercyjne zawierają coraz częściej rozbudowane możliwości cyfrowego przetwarzania sygnałów a nawet samodzielnej realizacji algorytmów uczenia maszynowego, takich jak drzewa decyzyjne. Dodatkowo producenci takich czujników dostarczają moduły ewaluacyjne i środowiska projektowe, które ułatwiają proces prototypowania i dostosowania systemu do wykrywania dedykowanych wzorców ruchowych.

Autorzy: dr inż. Tomasz Marciniak, Politechnika Poznańska, Instytut Automatyki i Robotyki, ul. Piotrowo 3a, 60-965 Poznań, E-mail: tomasz.marciniak@put.poznan.pl, Wojciech Marciniak, Koło Naukowe Decybel na Wydziale Automatyki, Robotyki i Elektrotechniki Politechniki Poznańskiej, E-mail: wojciech.to.marciniak@student.put.poznan.pl, prof. dr hab. inż. Adam Dąbrowski, Politechnika Poznańska, Instytut Automatyki i Robotyki, ul. Piotrowo 3a, 60-965 Poznań, E-mail: adam.dabrowski@put.poznan.pl

LITERATURA

- [1] Wearable Technology Market by Product (Wristwear, Headwear, Footwear, Fashion & Jewelry, Bodywear), Type (Smart Textile, Non-Textile), Application (Consumer Electronics, Healthcare, Enterprise & Industrial), and Geography - Global Forecast to 2026, <https://www.marketsandmarkets.com/Market-Reports/wearable-electronics-market-983.html>, dostęp 29.05.2021
- [2] Lara O.D., Labrador M.A., A Survey on Human Activity Recognition using Wearable Sensors, *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3 (2013), s. 1192 – 1209
- [3] Xue Y., Jin L., A naturalistic 3D acceleration-based activity dataset & benchmark evaluations, 2010 *IEEE International Conference on Systems, Man and Cybernetics*, (2010), pp. 4081-4085
- [4] Maurer U., Smailagic A., Siewiorek D. P., and Deisher M., Activity recognition and monitoring using multiple sensors on different body positions, *Proc. International Workshop on Wearable and Implantable Body Sensor Networks*, (Washington, DC, USA), IEEE Computer Society, (2006), pp. 4 pp.-116
- [5] Arduino Home, <https://www.arduino.cc/>, dostęp 30.05.2021
- [6] ATmega328P8-bit AVR Microcontroller with 32K Bytes In-SystemProgrammable Flash DATASHEET https://www1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf, dostęp 30.05.2021
- [7] MPU-6000 and MPU-6050 Product Specification Revision 3.4, InvenSense, (2013)
- [8] Piekuta D., Opaska monitorująca na rękę ze zdalnym powiadomianiem, *praca dyplomowa inżynierska, promotor Tomasz Marciniak, Politechnika Poznańska*, (2018)
- [9] MIT App Inventor, <https://appinventor.mit.edu/>, dostęp 30.05.2021
- [10] Kumar A., Goyal S., Varma M. Resource-efficient Machine Learning in 2 KB RAM for the Internet of Things, *Proceedings of the 34th International Conference on Machine Learning*, PMLR 70:1935-1944, (2017)
- [11] Pete Warden and Daniel Situnayake, Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers, *O'Reilly Media, Inc.*, (2019)
- [12] STM32 Nucleo-64 development board with STM32L476RG MCU, supports Arduino and ST morpho connectivity, <https://www.st.com/en/evaluation-tools/nucleo-l476rg.html>, dostęp 30.05.2021
- [13] Getting started with the X-NUCLEO-IKS01A3 motion MEMS and environmental sensor expansion board for STM32 Nucleo, Rev. 3, STMicroelectronics, 2020
- [14] LSM6DSO iNEMO inertial module: always-on 3D accelerometer and 3D gyroscope, Rev. 2, STMicroelectronics, 2019
- [15] Unico GUI User manual, UM1049, Rev. 6, STMicroelectronics, (2019)
- [16] Getting started with the AlgoBuilder application for the graphical design of Algorithms, Rev. 9, STMicroelectronics, (2020)
- [17] How to use the wireless multi sensor development kit with customizable app for IoT and wearable sensor applications Rev. 6, STMicroelectronics, (2021)
- [18] LSM6DSOX: Machine Learning Core, Rev 5, STMicroelectronics, (2021)
- [19] Getting started with the STM32Cube function pack for ultra-low power IoT nodes for AI applications based on audio and motion sensing – User manual, Rev 6, STMicroelectronics, (2020)
- [20] Getting started with X-CUBE-AI Expansion Package for Artificial Intelligence (AI), Rev 7, STMicroelectronics, (2021)
- [21] Marciniak T., Podbucki K., Suder J., Dąbrowski A. (2020) Analysis of Digital Filtering with the Use of STM32 Family Microcontrollers. In: Bartoszewicz A., Kabziński J., Kacprzyk J. (eds) *Advanced, Contemporary Control. Advances in Intelligent Systems and Computing*, vol 1196. Springer, Cham. https://doi.org/10.1007/978-3-030-50936-1_25