

Aleksander Piecuch

Programowanie może być interesujące – platforma Arduino

Dydaktyka Informatyki 12, 155-160

2017

Artykuł został opracowany do udostępnienia w internecie przez Muzeum Historii Polski w ramach prac podejmowanych na rzecz zapewnienia otwartego, powszechnego i trwałego dostępu do polskiego dorobku naukowego i kulturalnego. Artykuł jest umieszczony w kolekcji cyfrowej bazhum.muzhp.pl, gromadzącej zawartość polskich czasopism humanistycznych i społecznych.

Tekst jest udostępniony do wykorzystania w ramach
dozwolonego użytku.

Aleksander PIECUCH

*Prof. nadzw. dr hab., Uniwersytet Rzeszowski, Wydział Matematyczno-Przyrodniczy,
Katedra Inżynierii Komputerowej; Laboratorium Zagadnień Społeczeństwa Informatycznego,
ul. Prof. S. Pigoń 1, 35-310 Rzeszów; e-mail: apiecuch@ur.edu.pl*

PROGRAMOWANIE MOŻE BYĆ INTERESUJĄCE – PLATFORMA ARDUINO

PROGRAMMING CAN BE INTERESTING – ARDUINO PLATFORM

Słowa kluczowe: Arduino, programowanie.

Keywords: Arduino, programming.

Streszczenie

Niniejsze opracowanie jest propozycją nauczania programowania języka C/C++ bazującego na platformie Arduino dla III etapu edukacyjnego. Programowanie mikrokontrolerów daje możliwość nabycia szerokich kompetencji techniczno-informatycznych użytecznych zarówno w warunkach szkolnych (uczelnianych), jak i pozaszkolnych. W przekonaniu autora może wpłynąć na: poszerzenie horyzontów wiedzy techniczno-informatycznej, wybór dalszej ścieżki edukacyjnej, rozbudzenie nowych zainteresowań możliwych do urzeczywistnienia także w czasie wolnym.

Summary

This paper proposes teaching programming of language C/C++ based on the Arduino platform for the III education stage. Programming microcontrollers gives the opportunity of gaining a wide expertise in information technology useful both in school (university) as well as extracurricular activities. In the author's opinion it may influence: broadening the horizons of information technology knowledge, the choice of future educational path, awakening new interests possible to realize also in spare time.

Wstęp

Podstawowymi kompetencjami współczesnego człowieka stają się przede wszystkim kompetencje cyfrowe. Aktualnie odgrywają już bardzo dużą rolę, a należy się spodziewać, że w kolejnych latach ich znaczenie będzie sukcesywnie wzrastać. Pomysł włączenia nauki programowania do kanonu wykształcenia

ogólnego współczesnego człowieka jest krokiem w dobrym kierunku, aczkolwiek nie przesądającym jeszcze o „informatycznym” sukcesie. „Prognoście rynku pracy, mówiąc o kompetencjach cyfrowych, szczególne miejsce przypisują programowaniu. Niektórzy z nich stawiają tezę, że języki programowania staną się nowym alfabetem profesjonalistów XXI wieku. Co ważne, nie odnosi się to tylko do specjalistów ICT, ale w zasadzie do każdego”¹.

Programowanie powinno być interesujące

W stosunkowo długiej historii nauczania informatyki, mieliśmy już do czynienia z nauczaniem języków programowania. Były to np.: Basic, Logo, Pascal. Sama idea nauczania algorytmiki i programowania bez względu na to jak na nią patrzymy z perspektywy czasu, była słuszna. Cechy, które należałoby jej przypisać nie stanowią wyłącznie odniesienia do przedmiotów kształcenia informatycznego, ale mają znakomite znaczenie w uczeniu się i nauczaniu pozostałych przedmiotów ogólnokształcących. Zaliczyć do nich możemy umiejętność: dostrzegania i formułowania problemów, analizy, syntezy, planowania, logicznego myślenia, wnioskowania, oceniania. Z pewnością była to wartość dodana do procesów uczenia się chociaż nieuświadomiana sobie przez uczniów. Patrząc jednak globalnie z perspektywy nauczanego przedmiotu informatycznego² trudno dostrzec wymierne korzyści. Umiejętności programistyczne sprowadzały się niemal wyłącznie do rozwiązywania problemów natury matematycznej, a w konsekwencji tego były pozbawione użyteczności pozaszkolnej. Nie mogła zatem nadto dziwić niechęć uczniów do poznawania tajemników programowania. Uczenie programowania dla samego nauczania – mija się z celem. Konstruowanie programu na rozwiązywanie równania liniowego czy kwadratowego nie zainteresuje uczniów, a jedynie zniechęci ich do dalszej nauki. „(...) Uczniowie powinni widzieć **użyteczne wyniki własnych zmagania z programowaniem** i to takie, które mogą mieć zastosowanie w pozaszkolnej działalności uczniów. Aktualnie przyszłość w rozwoju społeczno-gospodarczym leży w: elektronice, automatyce i robotyce”³. W moim przekonaniu taki kierunek winno dziś się obracać przystępując do ambitnego planu nauczania programowania.

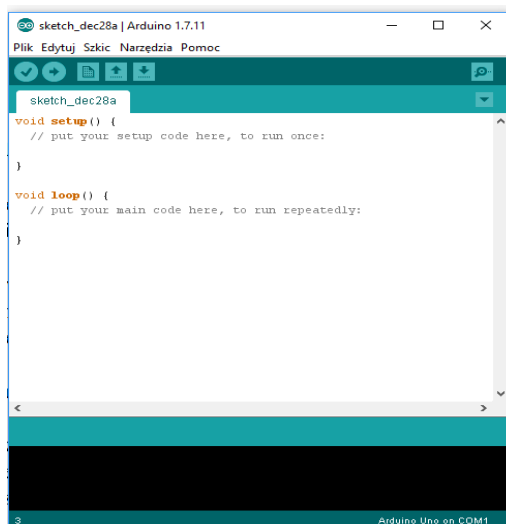
¹ J. Kulasa, A. Nizioł, *Umiejętność programowania jako element kształcenia studentów kierunków nieinformatycznych*, „Edukacja – Technika – Informatyka” 2016, nr 3(17), Rzeszów.

² Używam określenia przedmiot informatyczny, bowiem nazwa przedmiotu zmieniała się na przestrzeni lat.

³ A. Piecuch, *Nowe wyzwania przed edukacją informatyczną w świetle planowanej reformy systemu edukacji 2017/18*, „Technika a vzdělávání” 2016, nr 2, Wyd. UMB v Banská Bystrica.

Platforma programistyczna Arduino

Alternatywą dla nauki programowania „tablicowego”, może być platforma programistyczna dla systemów wbudowanych⁴ występująca pod nazwą Arduino. Projekt Arduino rozwijany jest od 2005 roku we Włoszech⁵. W założeniu miał stanowić platformę dla osób nie związanych zawodowo z programowaniem mikrokontrolerów. „Płyta Arduino od chwili pojawienia się na rynku stała się czymś więcej niż eksperymentalną platformą. Jest stylem i ideą otwartego oprogramowania i sprzętu, zmieniającą oblicze informatyki i edukacji. Dzięki łatwości zdobycia wiedzy potrzebnej do rozpoczęcia pracy z Arduino otworzyło nową drogę do budowania skomplikowanych urządzeń do praktycznych zastosowań. Połączenie tych cech, sprawia że Arduino jest znakomitą środowiskiem rozwojowym zarówno dla studentów, doświadczonych programistów, jak i projektantów”⁶.



Rys. 1. Zrzut ekranu edytora Arduino

Już z przytoczonych stwierdzeń wynika, że do realizacji własnych wizji programistycznych wcale nie jest potrzebna szeroka wiedza elektroniczna. Za wykorzystaniem Arduino w edukacji i nauce programowania przemawiają do-

⁴ Systemem wbudowanym jest system komputerowy składający się z odpowiednio dobranych komponentów sprzętowych i programowych, często zaprojektowany pod kątem określonej aplikacji programowej. Odpowiada ona za realizację jego funkcji i wpływa na sposób komunikacji z użytkownikiem, <http://automatykab2b.pl/tematmiesiaca/2168-systemy-wbudowane?showall=1>.

⁵ Por. <http://forbot.pl/blog/artykuly/programowanie/kurs-arduino-w-robotyce-1-wstep-id936>

⁶ R. Anderson, D. Cervo, *Arduino dla zaawansowanych*, Gliwice 2014.

datkowo niska cena płytki z mikrokontrolerem, dostępność szerokiej gamy modułów rozszerzających funkcjonalność zestawu, otwarte środowisko programistyczne⁷ bazujące na języku C/C++. Zrzut ekranu edytora Arduino pokazano na rys. 1. Do programowania nie jest potrzebny żaden dodatkowy programator, bowiem komunikacja pomiędzy komputerem PC a modułem Arduino odbywa się za pośrednictwem złącza USB.

Ponadto uruchamianie zestawu nie wymaga wykonywania połączeń lutowicznych, ponieważ te można zrealizować za pomocą płytki stykowej i zestawu przewodów połączeniowych, a co najważniejsze z edukacyjnego punktu widzenia, uczeń ma możliwość na bieżąco obserwować działanie napisanego przez siebie programu i dokonywać jego korekt.

Cechy Arduino

Na rys. 2 pokazano przykładowe modele płytek Arduino. W rzeczywistości na rynku dostępnych jest o wiele więcej modeli, co oznacza, że każdy do własnych potrzeb może wybrać optymalne rozwiązanie.



Rys. 2. Różne modele Arduino; a) Nano; b) Uno; c) Mega

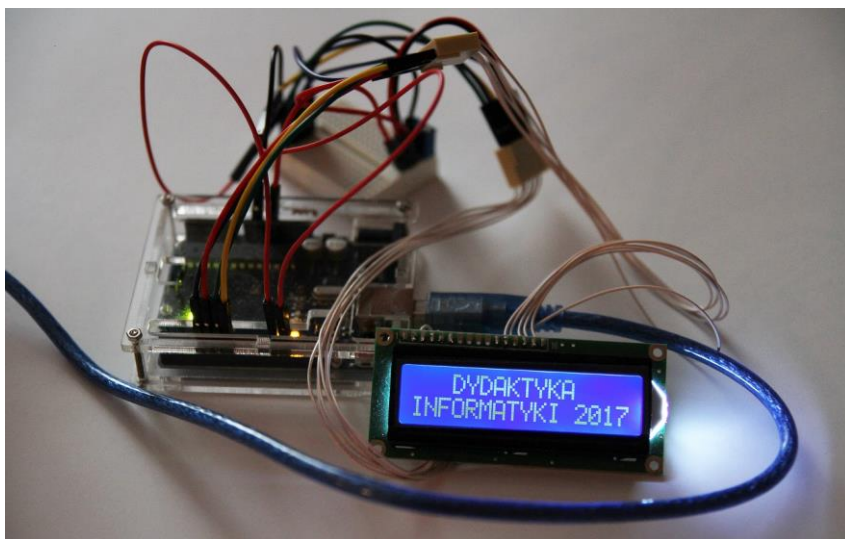
Poszczególne modele w zależności od zastosowanego mikrokontrolera udostępniają na złączach różną liczbę sygnałów (wejścia/wyjścia) analogowych i cyfrowych (w tym PWM). Ponadto dostępne są dla użytkownika: złącze USB, złącze zasilające, przycisk Reset. Warto też wspomnieć, że w zależności od modelu do dyspozycji użytkownika pozostaje różna ilość pamięci⁸. Układ może być zasilany ze złącza USB⁹ lub zewnętrznego zasilacza o napięciu z zakresu (7–12) VDC.

⁷ Oprogramowanie można bezpłatnie pobrać ze strony: <https://www.arduino.cc/en/Main/Software>

⁸ Szczegóły konstrukcyjne i parametry techniczne dla poszczególnych modeli Arduino można bez większego kłopotu odszukać w Internecie na licznych forach dyskusyjnych poświęconych problematyce programowania.

⁹ O tym czy układ może być zasilany ze złącza USB decyduje pobór prądu całego układu. Typowo dla USB nie może przekroczyć 0,5A.

Co można zbudować na Arduino?



Rys. 3. Zaprogramowany moduł Arduino UNO wyświetlający komunikat

Źródło: opracowanie własne [fot. A.P.]

Platforma Arduino jest uniwersalną bazą sprzętowo-programistyczną pozwalającą na realizację bardzo prostych jak również złożonych projektów (rys. 3). Jedynym ograniczeniem są wyobrażenia i potrzeby konstruktora-programisty. Rozpoczynając naukę programowania, należy brać pod uwagę fakt, że sposób programowania mikrokontrolerów różni się nieco od pisania programów typowo matematycznych, gdzie rezultatem działania programu jest konkretny wynik liczbowy lub wykres. Z pewnością nie należy również rozpoczynać nauki od bardzo ambitnych projektów. Najlepiej zacząć od mało ambitnych projektów polegających np. na wysterowaniu diody LED przechodząc następnie do programowania przycisków i sukcesywnie zwiększając stopień trudności konstrukcji i programu.

Zdobywana systematycznie wiedza i umiejętności programowania mikrokontrolerów powinny umożliwić z powodzeniem konstruowanie i oprogramowanie np.: zegara, stopera, termometru, kompasu, robota mobilnego i wielu innych¹⁰, a kończąc na elementach inteligentnego domu (mieszkania).

Ukierunkowanie nauczania programowania na aspekty techniczne wydaje się być rozwiązaniem z „przyszłością”. Po pierwsze uczniowie poznają bardzo

¹⁰ Moduł Arduino zawiera wyłącznie mikrokontroler z niezbędnymi elementami wej/wyj. Celem realizacji konkretnych funkcjonalności np. takich, które wymieniono w tekście, konieczne jest dokupienie odpowiednich modułów peryferyjnych lub ich samodzielne wykonanie.

konkretny język programowania C/C++, który jest powszechnie stosowany i rozwijany. Nabywają przy tej okazji umiejętności kreatywnego myślenia i rozwiązywania problemów zarówno technicznych, jak i typowo informatycznych. Z pewnością są też stymulatorem rozwoju kompetencji i zainteresowań, które potencjalnie mogą skierować uczniów na ścieżkę edukacyjną związaną z branżą informatyki, elektroniki czy też automatyki i robotyki, a przecież w tych właśnie dyscyplinach upatruje się szans rozwoju dla polskiej gospodarki.

Tytułem komentarza dodajmy, że nauka programowania bazującego na mikrokontrolerach jest możliwa do realizacji na III etapie edukacyjnym, czyli wówczas gdy uczniowie będą mieli podstawową wiedzę z zakresu: fizyki, techniki, algorytmiki i języka programowania. Wyposażenie uczniów w taką wiedzę i konkretne umiejętności daje możliwości jej wykorzystania w warunkach pozaszkolnych.

Zakończenie

Postęp w mikroelektronice obecnie stwarza niepowtarzalne warunki dla procesów nauczania. Nauka programowania, którą zapowiada MEN od roku szkolnego 2017/2018, nie musi być czysto teoretyczna i pozbawiona użyteczności pozaszkolnej. Niniejsze opracowanie należy potraktować jako inspirację do podjęcia działań edukacyjnych na rzecz dotąd niestosowanych w praktyce szkolnej rozwiązań. Włączenie do procesu kształcenia nowoczesnych platform bazujących na mikrokontrolerach w znakomity sposób może ułatwić naukę programowania, uatrakcyjnić zajęcia, a jednocześnie dać namacalne dowody na sens uczenia się języków programowania. Wiedza i praktyczne umiejętności będą na pewno przydatne w dalszej nauce (studiowaniu), rozwijaniu własnych zainteresowań, a być może dla tych którzy zechcą podjąć pracę zawodową stanowić będą cenną umiejętność pożądaną przez pracodawców.

Bibliografia

- Anderson R., Cervo D., *Arduino dla zaawansowanych*, Helion, Gliwice 2014.
<http://automatykab2b.pl/tematmiesiaca/2168-systemy-wbudowane?showall=1>
<http://forbot.pl/blog/artykuly/programowanie/kurs-arduino-w-robotyce-1-wstep-id936>
<https://www.arduino.cc/en/Main/Software>
Kulasa J., Nizioł A., *Umiejętność programowania jako element kształcenia studentów kierunków nieinformatycznych*, „Edukacja – Technika – Informatyka” 2016, nr 3(17), Rzeszów.
Piecuch A., *Nowe wyzwania przed edukacją informatyczną w świetle planowanej reformy systemu edukacji 2017/18*, „Technika a vzdelávanie” 2016, nr 2, Wyd. UMB v Banská Bystrica.