

Marcin WYGAS<sup>1</sup>

Opiekun naukowy: Sławomir HERMA<sup>2</sup>

## **TECHNOLOGIE INFORMACYJNE, A NOWOCZESNE MOŻLIWOŚCI NAUCZANIA: METODY I SYMULACJE**

**Streszczenie:** Era cyfrowa radykalnie zmieniła sposób, w jaki uczymy się i nauczamy. Technologie informacyjne (TI) stały się nieodłącznym elementem edukacji, oferując zarówno nauczycielom, jak i uczniom nowe możliwości i narzędzia. Niniejszy artykuł ma na celu przedstawienie, jak technologie informacyjne wpłynęły na nowoczesne metody nauczania, ze szczególnym uwzględnieniem symulacji i innych innowacyjnych podejść.

**Słowa kluczowe:** symulacje, mikrokontrolery, programowanie

## **INFORMATION TECHNOLOGY AND MODERN TEACHING CAPABILITIES: METHODS AND SIMULATIONS**

**Summary:** The digital era has radically changed the way we learn and teach. Information Technology (IT) has become an integral part of education, offering both teachers and students new opportunities and tools. This article aims to present how information technology has influenced modern teaching methods, with a special focus on simulations and other innovative approaches.

**Keywords:** simulations, microcontrollers, programming

### **1. Technologie informacyjne w edukacji**

Technologie informacyjne to szeroki zakres narzędzi cyfrowych i zasobów, które mogą być wykorzystywane do celów edukacyjnych. Obejmują one oprogramowanie edukacyjne, platformy e-learningowe, narzędzia do komunikacji online, wirtualne laboratoria, tablice interaktywne, urządzenia mobilne i wiele innych. TI umożliwiają dostęp do ogromnych ilości informacji, interaktywne nauczanie, personalizację procesu uczenia się i zwiększenie zaangażowania uczniów.

---

<sup>1</sup> mgr, V Liceum Ogólnokształcące w Bielsku-Białej, marcin.wygas@lo5.bielsko.pl

<sup>2</sup> dr inż., Uniwersytet Bielsko-Bialski, Wydział Budowy Maszyn i Informatyki, sherma@ubb.edu.pl

Platformy e-learningowe, takie jak Moodle czy Google Classroom, pozwalają nauczycielom na tworzenie kursów online, udostępnianie materiałów edukacyjnych, przeprowadzanie testów i quizów, a także monitorowanie postępów uczniów.

Symulacje to narzędzia edukacyjne, które umożliwiają odtworzenie rzeczywistych sytuacji lub zjawisk w kontrolowanym środowisku wirtualnym. Dzięki temu uczniowie mogą eksperymentować, podejmować decyzje i obserwować konsekwencje swoich działań bez ryzyka wystąpienia negatywnych skutków w rzeczywistości. Przykładowo symulacje medyczne, które umożliwiają studentom medycyny i pielęgniarstwa praktykowanie procedur chirurgicznych i medycznych w bezpiecznym odizolowanym środowisku wirtualnym. Główne zalety wykorzystania TI i symulacji w nauczaniu to:

- Dostęp do zasobów edukacyjnych z dowolnego miejsca i o dowolnej porze.
- Możliwość indywidualizacji procesu uczenia się, dostosowywania tempa i stylu nauki do potrzeb ucznia.
- Ułatwienie zrozumienia trudnych i abstrakcyjnych pojęć dzięki wizualizacji i interakcji.
- Rozwój umiejętności krytycznego myślenia, rozwiązywania problemów i podejmowania decyzji.
- Zwiększenie zaangażowania i motywacji uczniów.

## 2. Tinkercad jako narzędzie do symulacji mikrokontrolerów

Tinkercad to platforma online, która umożliwia projektowanie i symulację obwodów elektronicznych, w tym tych opartych na mikrokontrolerach Arduino. Dzięki Tinkercad uczniowie mogą tworzyć wirtualne prototypy swoich projektów, testować je i debugować bez potrzeby fizycznego posiadania komponentów elektronicznych, a wszystko to może się odbywać pod kontrolą nauczyciela.

Przykład użycia: Nauczyciel może przygotować zadanie dla uczniów, polegające na zaprojektowaniu prostego obwodu z użyciem Arduino, LED i rezystora w Tinkercad. Uczniowie, korzystając z platformy, mogą połączyć elementy w wirtualnym środowisku, a następnie napisać i przetestować kod odpowiedzialny za miganie diody LED. Cały proces odbywa się online, co ułatwia naukę i pozwala na szybkie iteracje projektu. Pomocą w realizacji zajęć z tego tematu bardzo pomocny może się okazać darmowy kurs na stronie FORBOT.PL [1] (<https://tiny.pl/g2g92>)

Zalety wykorzystania Tinkercad w nauczaniu [3]:

- Dostępność: Platforma jest dostępna online, co oznacza, że uczniowie i nauczyciele mogą z niej korzystać z dowolnego miejsca i na różnych urządzeniach.
- Bezpieczeństwo: Symulacje w Tinkercad eliminują ryzyko związane z pracą z prawdziwymi komponentami elektronicznymi, takie jak zwarcia czy uszkodzenia sprzętu.
- Niski koszt: Brak potrzeby zakupu fizycznych komponentów obniża koszty nauki i eksperymentowania.

- Interaktywność i zaangażowanie: Praca z Tinkercad jest interaktywna i angażująca, co przyczynia się do lepszego zrozumienia materiału i zwiększenia motywacji uczniów do nauki.
- Wyzwania i ograniczenia: Podobnie jak w przypadku innych technologii informacyjnych, korzystanie z Tinkercad w edukacji wiąże się z wyzwaniami. Należy do nich potrzeba zapewnienia uczniom dostępu do stabilnego połączenia internetowego, konieczność szkolenia nauczycieli w zakresie korzystania z platformy, a także ograniczenia samego narzędzia, które może nie oferować wszystkich funkcji dostępnych w fizycznych mikrokontrolerach Arduino [3].

Warto zwrócić uwagę iż podstawa programowa dla szkół średnich w Polsce określa cele i treści kształcenia z przedmiotu informatyka. Użycie symulatorów, takich jak Tinkercad dla Arduino [2], może bezpośrednio przyczynić się do ich realizacji. Poniżej przedstawiam przykłady, jak można zintegrować użycie tych symulatorów z podstawą programową:

- **Algorytmika i programowanie** (podstawa programowa: realizowanie algorytmów, programowanie strukturalne i obiektowe, testowanie i debugowanie programów):  
*Uczniowie mogą używać Tinkercad do pisania, testowania i debugowania programów dla Arduino. Przykładowe zadanie to stworzenie programu, który kontroluje sekwencję świateł LED, co pozwala na naukę podstaw algorytmiki, struktur sterujących i składni języka programowania.*
- **Przetwarzanie danych** (podstawa programowa: pozyskiwanie, przetwarzanie i prezentacja danych):  
*Uczniowie mogą zrealizować projekty z wykorzystaniem czujników podłączonych do Arduino, na przykład czujnika temperatury lub wilgotności. Dane z czujników mogą być następnie przetwarzane i prezentowane na ekranie LCD lub wysyłane do komputera i analizowane przy użyciu arkusza kalkulacyjnego.*
- **Architektura komputerów i systemy operacyjne** (podstawa programowa: budowa i działanie komputera, systemy operacyjne):  
*Praca z Arduino za pośrednictwem Tinkercad umożliwia uczniom zrozumienie, jak działają mikrokontrolery, które są podstawowym składnikiem wielu współczesnych systemów komputerowych. Uczniowie mogą nauczyć się, jak mikrokontroler komunikuje się z innymi elementami układu, jakie pełni funkcje i jakie są ograniczenia jego działania.*
- **Sieci komputerowe** (podstawa programowa: podstawy budowy i działania lokalnych sieci komputerowych, komunikacja w sieci):  
*Uczniowie mogą realizować projekty z użyciem modułów komunikacyjnych dla Arduino (na przykład modułów Bluetooth lub Wi-Fi), aby nauczyć się podstaw budowy i działania sieci komputerowych. Przykładowe zadanie to stworzenie prostego systemu do komunikacji między dwoma mikrokontrolerami Arduino za pośrednictwem modułu komunikacyjnego.*
- **Bezpieczeństwo informacyjne** (podstawa programowa: zasady bezpiecznego korzystania z komputera i sieci, ochrona danych osobowych):  
*Realizując projekty z Arduino, uczniowie mogą nauczyć się, jak zabezpieczać komunikację między urządzeniami i jak chronić dane przesyłane za pośrednictwem modułów komunikacyjnych. Przykładem może być zadanie,*

w którym uczniowie implementują prosty protokół szyfrowania danych wysyłanych między dwoma mikrokontrolerami.

### 3. Przykład projektu wykorzystania mikrokontrolerów przez uczniów.

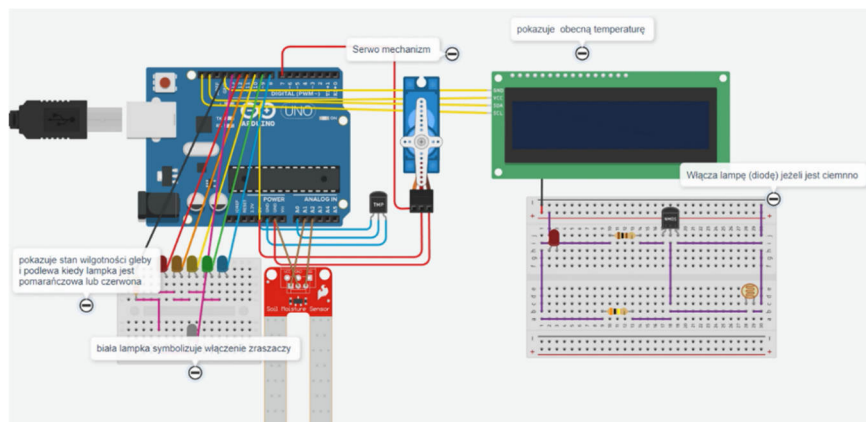
Na zajęciach informatyki w klasie 1 liceum po 3h lekcyjnych wprowadzenia do tematu uczniowie otrzymali następujący problem do rozwiązania:

*„Problem dotyczy ogrodnictwa, takiego drobnotowarowego. W niezbyt dużym gospodarstwie do hodowli rozsad i dla przyspieszenia rozwoju świeżych nasadzeń używa się inspektorów, czyli sporych płytkich skrzyń, zakrywanych ramką drewnianą lub metalową, przeszkloną albo owiniętą folią. W chłodny dzień inspekty zapewniają podwyższoną temperaturę, korzystną dla roślin. Z kolei w dzień słoneczny ramy trzeba uchylać, żeby wewnątrz nie było zbyt gorąco. W dużych gospodarstwach typu przemysłowego stosuje się tunele szklane i foliowe, w których okna uchylane są mechanicznie. W przypadku inspektów mechanizacja się jednak nie opłaca, a przy ich dużej liczbie otwieranie i zamykanie okien stanowi uciążliwą pracę. Co zrobić? I to jest zadanie dla Was.”*



Rysunek 11. Przykład standardowego inspektora bez automatyki

Sposób podejścia uczniów do zadania był różny. Jedni pozostali w warunkach symulacji, inni zaś wyszli z środowiska wirtualnego wykorzystując do tego celu stare pudełko po butach, akwarium czy inne rzeczy z recyklingu a następnie zintegrowały je już z fizycznymi mikrokontrolerami Arduino które zakupili bądź otrzymali ze szkoły. Poniżej przykładowa praca uczennicy klasy pierwszej liceum.



Rysunek 12. Projekt symulacji inspektora ogrodniczego



Rysunek 13. Projekt w warunkach rzeczywistych wykonany na podstawie symulacji

Poniżej znajdują się kod programu do projektu:

```
#include <Servo.h>
//deklaracja biblioteki servo mechanizmu

Servo servo_8;
//przypisanie pinu nr 8 do sterownia servo mechanizmem

int baselineTemp = 0;
int celcius = 0;

//deklaracja zmiennych i przypisanie im wartości 0.
void setup()
```

```
{
  Serial.begin(9600); //zainicjowanie konsoli do debugowania
  servo_8.attach(8); //zainicjowanie serwomechanizmu
  pinMode(A0, INPUT); // przypisanie pinowi A0 odczytu
  z czujnika temperatury, zadeklarowanie pinu jako pin
  odczytujący informacje.

  pinMode(3, OUTPUT); // przypisanie pinowi D3 diody Zielonej
  pinMode(4, OUTPUT); // przypisanie pinowi D4 diody Czerwonej
}

void loop()
{
  celcius = analogRead(A0); //odczyt temperatury z czujnika
  Serial.print("TEMPERATURA =");
  Serial.print(celcius);
  Serial.print("°C");
  Serial.println();

  if ( celcius >= 30 ) {
    digitalWrite(3, HIGH); //zapalenie diody zielonej
    digitalWrite(4, LOW); //zgaszenie diody czerwonej
    servo_8.write(90); //zmiana pozycji serwa (OKNO UCHYLONE)
    Serial.print("Okno uchylone / dioda zielona świeci");
    Serial.println();
  } else {
    digitalWrite(3, LOW); //zgaszenie diody zielonej
    digitalWrite(4, HIGH); //zapalenie diody czerwonej
    servo_8.write(0); //zmiana pozycji serwa (OKNO ZAMKNIĘTE)
    Serial.print("Okno zamknięte / dioda czerwona świeci");
    Serial.println();
  }
  delay(1000); // Pętla powtarza się co 1000 milisekund(s)
}
```

#### Biblioteka i inicjalizacja:

- **#include <Servo.h>**: Importuje bibliotekę Servo, która umożliwi kontrolę serwomechanizmów.
- **Servo servo\_8**; Tworzy obiekt serwomechanizmu, który będzie sterowany za pomocą pinu nr 8.

#### Zmienne globalne:

- **int baselineTemp = 0**; Zmienna na bazową temperaturę.
- **int celcius = 0**; Zmienna do przechowywania odczytanej wartości temperatury.

#### Funkcja setup():

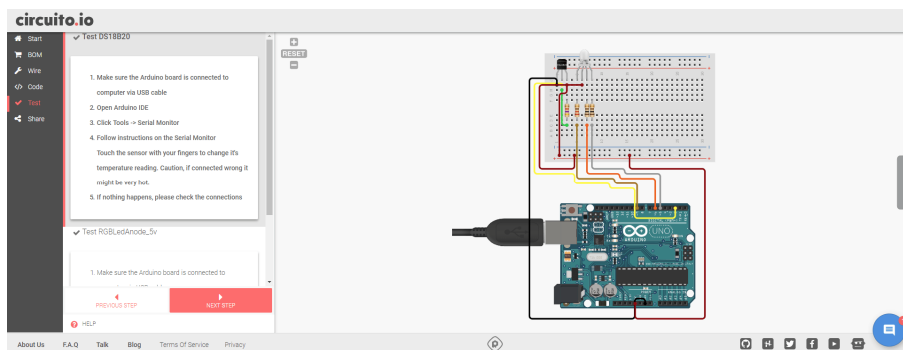
- **Serial.begin(9600)**; Inicjalizuje komunikację szeregową.

- **servo\_8.attach(8);**Przypisuje pin nr 8 do kontrolowanego serwomechanizmu.
- **pinMode(A0, INPUT);** Ustawia pin A0 jako wejście do odczytu danych z czujnika temperatury.
- **pinMode(3, OUTPUT);** Ustawia pin D3 jako wyjście do diody zielonej.
- **pinMode(4, OUTPUT);** Ustawia pin D4 jako wyjście do diody czerwonej.

### Funkcja loop():

- **celcius = analogRead(A0);** Odczytuje wartość temperatury z czujnika podłączonego do pinu A0.
- Warunki **if** i **else** kontrolują stan diod oraz serwomechanizmu w zależności od odczytanej temperatury. Jeśli temperatura jest równa lub wyższa niż 30 stopni Celsjusza, zielona dioda się zapala, serwomechanizm uchyla okno, a czerwona dioda się wyłącza. W przeciwnym wypadku, zielona dioda się wyłącza, czerwona zapala, a serwomechanizm zamyka okno.
- **delay(1000);**: Pętla jest powtarzana co 1000 milisekund (1 sekunda), co pozwala na regularny odczyt temperatury i aktualizację stanu systemu.

Inną ciekawą platformą do nauki mikrokontrolerów zwłaszcza dla osób młodszych i mniej technicznych jest **circuito.io**. To program, który na podstawie listy układów wygeneruje dla Was **schemat montażowy, kod testowy oraz dokładną instrukcję budowy i montażu**.



*Rysunek 14. Platforma circuito.io. Po lewej stronie widać szczegółowe instrukcje podłączania komponentów*

Circuito.io jest bardzo ciekawym udogodnieniem. Szczególnie zadowoleni będą początkujących, którzy chcą coś szybko przetestować. Korzystając z takiej platformy można łatwo uchronić się przed wieloma błędami, których znalezienie zajęłoby później stosunkowo dużo czasu [4].

Podsumowując, symulatory takie jak Tinkercad dla Arduino oferują bogate możliwości integracji z podstawą programową nauczania informatyki w szkole

średniej [3]. Umożliwiają one uczniom praktyczne zastosowanie wiedzy teoretycznej, rozwijanie umiejętności programistycznych i inżynierskich, a także zrozumienie złożoności współczesnych systemów komputerowych.

## LITERATURA

1. Serwis internetowy Forbot.pl – Kurs Arduino dla początkujących: <https://forbot.pl/blog/kurs-arduino-podstawy-programowania-spis-tresci-kursu-id5290>, 01.11.2023
2. O'SHEA A.: Elektronika i internet rzeczy. Przewodnik dla ludzi z prawdziwą pasją. Wyd. Helion 2021
3. WILD J.: Designing and programming Arduino-based electronics projects using Thinkercad, Lighting Source Inc, 2022
4. Serwis internetowy: <https://circuitio.io> (dostęp 11-2023)