

УДК 004.6

Болілий Василь Олександрович –

кандидат фізико-математичних наук, доцент,

доцент кафедри інформатики та ІТ

Центральноукраїнського державного педагогічного університету імені

Володимира Винниченка

ORCID ID [0000-0002-1923-1058](https://orcid.org/0000-0002-1923-1058)

e-mail: basilb@kspu.kr.ua

Олійник Владислав Михайлович –

студент II курсу рівня магістр

фізико-математичного факультету

Центральноукраїнського державного педагогічного університету імені

Володимира Винниченка

ORCID ID [0000-0002-5584-1531](https://orcid.org/0000-0002-5584-1531)

e-mail: vladokoliynyk@gmail.com

ВИКОРИСТАННЯ ПЛАТФОРМИ ARDUINO ПРИ ВИВЧЕННІ ПРОГРАМУВАННЯ

Постановка та обґрунтування актуальності проблеми.

Теоретичний матеріал, що викладається студентам спеціальностей, пов'язаних із програмуванням, електронікою чи інформатикою має підкріплюватись на практиці. Проте, зазвичай, типових лабораторних робіт може бути недостатньо аби задовольнити зацікавленість студента в програмуванні. Вивчення платформи Arduino надає можливість застосовувати здобуті знання як на початку вивчення основ програмування (типи даних, логічні оператори, розуміння циклів, рекурсії, функціональне програмування), так і вже при поглибленому вивченні програмування (комп'ютерні мережі, шифрування даних) шляхом побудови реального пристрою для вирішення тієї чи іншої задачі та закріплення вивченого матеріалу.

Аналіз останніх досліджень і публікацій.

Інформативність, цілеспрямованість, керованість, пізнавальна активність студентів у процесі їхньої підготовки, як гідно освічених фахівців, завжди викликали і зараз викликають стурбованість і постійне творче незадоволення у причетних до освіти спеціалістів різного профілю [3, 4, 6, 7].

Організація навчального процесу визначається не тільки формою подання навчальної інформації, але і педагогічним механізмом регулювання навчальної діяльності і самою структурою навчального процесу. Тому навчальний процес в сучасному ВНЗ повинен бути спрямований не тільки на формування фундаментальних знань і умінь, але й на набуття студентами необхідних професійно-значущих міждисциплінарних компетенцій [2, 3]. Розв'язанню цієї проблеми і присвячується дана стаття.

Мета статті. Опис та обґрунтування доцільності використання елементів платформи Arduino на заняттях з курсу програмування.

Методи дослідження. Опис, порівняння, аналіз, узагальнення.

Виклад основного матеріалу дослідження.

Історія платформи Arduino почалась в Інституті Проектування Взаємодії Івреа (Interaction Design Institute Ivrea), коли Массімо Банзі вирішив створити власну, відносно дешеву, платформу для студентів чи інженерів, що підходить для конструювання електронних пристроїв та будь яких інших електронних саморобок аматорів та новачків сфери електроніки та програмування [1].

Arduino – фізична обчислювальна платформа з відкритим вихідним кодом, заснована на платі та середовищі розробки. Arduino може застосовуватися для розробки самостійних інтерактивних пристроїв або може бути пов'язана з програмою на комп'ютері. Плати можуть бути зібрані самостійно (всі креслення, оригінальні схеми, проекти у вільному доступі) або ж можна придбати вже зібрану плату від офіційного виробника, або від різних виробників, що копіюють плати [1, 5].

Середовище розробки може бути завантажено безкоштовно з офіційного сайту (<https://www.arduino.cc/en/Main/Software>).

Завдяки своїй простоті, доступності та широкому вибору, Arduino

використовується в тисячах різних проектах та додатках [5]. Програмне забезпечення Arduino досить гнучке для досвідчених інженерів та програмістів (мова програмування мікроконтролера – C++), і в той самий час, просте у використанні для початківців, тому що велика частина «складного» мови C++ просто не потрібна. З Arduino можна працювати, як на Mac, Linux, так і на Windows. Вчителі та студенти використовують Arduino для створення недорогих наукових інструментів, для дослідницької та освітньої діяльності в хімії, фізиці, або для початку роботи з програмуванням та робототехнікою. Arduino спрощує процес роботи з мікроконтролерами, але вона все ж має деякі значні переваги для вчителів, студентів та зацікавлених любителів над іншими системами [7].

Однією з переваг платформи, є її невелика ціна. Плати Arduino відносно недорогі в порівнянні з іншими платформами мікроконтролерів. Найдешевша версія модуля Arduino може бути зібрана власноруч, і навіть попередньо зібрані комплекти та набори оригінального Arduino коштують менше 50\$. Arduino має колосально велику базу плат розширення та різноманітних модулів та датчиків а також бібліотек для зручного використання останніх [1, 4].

Массімо Банзі та його колеги були прихильниками відкритості коду програмного забезпечення. І оскільки їх метою було створення швидкої і легкодоступної платформи для розробки, вони розуміли, що буде краще відкрити проект для максимально можливої кількості людей, ніж закрити його. Через відкритість платформи існує велика кількість копій інших виробників та постачальників плат-аналогів, в яких ціна за плату сягає 5\$, а якість пристрою не гірша за оригінал.

Не кожен студент бажає займатись розробкою програмного забезпечення для комп'ютера чи телефона, або займатись веб-програмуванням. Деякі студенти вважають за краще зайнятися розробкою програмно-апаратного забезпечення й мати можливість розібратись в темі з азів, і саме таку можливість надає платформа Arduino з її низьким порогом входження. Студентам, що в подальшому мріють працювати в сфері розробки програмно-

апаратного забезпечення, робота на платформі Arduino буде дуже корисна з практичної точки зору. Необхідним мінімумом мають бути фундаментальні поняття фізики: струм, напруга і опір. Паралельно здобуваються навички написання цілісної великої програми, із розумінням структури всього проекту. Більше практики – більше ідей. При роботі з платою студент буде розуміти деякі поняття схемотехніки, яка з практикою переходить в робототехніку. Прикладом користі для студентів може служити те, що студент з легкістю зможе застосувати отриманий досвід роботи з платою. Під платформу Arduino випускається багато додаткових датчиків, модулів та розширювальних плат, за допомогою яких простий мікроконтролер можна перетворити на автоматизованого робота на радіокеруванні із передачею зображення з камери, чи власноруч створити одну із складових розумного дому, сигналізацію чи розумне освітлення із LED-стрічками і керуванням зі смартфона по Wi-Fi мережі [1]. Можливості Arduino обмежені лише фантазією інженера, розробника чи програміста.

Програма, написана в середовищі Arduino, називається “скетч”. Скетч пишеться в спеціальному середовищі для розробки Arduino IDE. Під час збереження і експорту проекту в області повідомлень з'являються пояснення, відображаються помилки. Вікно виведення тексту (консоль) показує повідомлення Arduino, що включають повні звіти про помилки та супровідну інформацію. Кнопки панелі інструментів дозволяють перевірити (скомпілювати) і записати програму в мікроконтролер («прошити» мікроконтролер), створити, відкрити й зберегти скетч, відкрити моніторинг послідовної шини зв'язку із мікроконтролером (консоль).

Arduino придатний і для демонстрації найпростіших задач із основ програмування. Ми виділили декілька завдань і представили як їх можна адаптувати для інтерактивної роботи із платформою Arduino.

Розглянемо плату Arduino Uno і підключений до 8-го піна через резистор на 220 Ом світлодіод (Рис.1.).

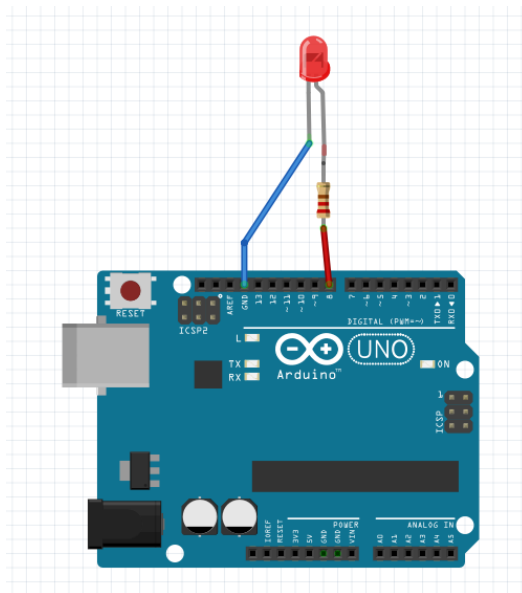


Рис.1. Схема макету. Плата Arduino Uno і світлодіод з резистором 220 Ом.

Завдання 1. П'ять разів мигнути світлодіодом, після чого має слідувати затримка довша ніж тривалість спалаху світлодіода.

Будь-який скетч для Arduino складається із двох фундаментальних частин: це дві функції `setup()` та `loop()`. Перша служить для першочергових налаштувань пристрою і код функції виконується один раз при першому старті пристрою. Друга функція, `loop()` повторюється в нескінченному циклі доки пристрій не буде вимкнено.

В лістингу 1 представлено код скетчу для розв'язання поставленої задачі.

При початковому налаштуванні в функції `setup()` викликом функції `pinMode(8, OUTPUT)` програма встановлює режим роботи заданого 8-го піну як виходу - `OUTPUT`. Функція `digitalWrite(8, HIGH)` подає `HIGH` значення на цифровий пін під номером 8, це означає що на 8-му піні з'явився сигнал в 5 В. Відповідно при виклику функції із другим аргументом `LOW` подача сигналу припиняється. Функція `delay()` служить для виклику затримки на вказане в аргументі число мілісекунд.

Далі конструкція повторюється ще чотири рази, і в кінці програми за допомогою `delay(1000)` відбувається затримка виконання програми на 1 секунду. Після завершення виконання коду функції `loop()`, вона починає виконуватись заново, із початку.

```

void setup() {
  pinMode(8, OUTPUT);
}
void loop() {
  //спалах 1
  digitalWrite(8, HIGH);
  delay(300);
  digitalWrite(8, LOW);
  delay(200);

  //спалах 2
  digitalWrite(8, HIGH);
  delay(300);
  digitalWrite(8, LOW);
  delay(200);

  //спалах 3
  digitalWrite(8, HIGH);
  delay(300);
  digitalWrite(8, LOW);
  delay(200);

  //спалах 4
  digitalWrite(8, HIGH);
  delay(300);
  digitalWrite(8, LOW);
  delay(200);

  //спалах 5
  digitalWrite(8, HIGH);
  delay(300);
  digitalWrite(8, LOW);
  delay(200);

  //затримка
  delay(1000);
}

```

Лістинг 1. Скетч програми «Завдання 1».

Видно, що в коді програми є п'ять ідентичних блоків, які можна удосконалити змінивши скетч наступним чином:

```

void setup() {
  pinMode(8, OUTPUT);
}
void loop() {
  for (int i=0; i < 5; i++) {
    digitalWrite(8, HIGH);
    delay(300);
    digitalWrite(8, LOW);
    delay(200);
  }

  delay(1000);
}

```

Лістинг 2. Удосконалений скетч програми «Завдання 1».

Удосконалений скетч має цикл `for`, який спрощує написання коду навіть для такого простого прикладу, в декілька разів.

Що ж демонструє цей приклад? По-перше, роботу нескінченного **циклу** `loop()`, який не має умови виходу. По-друге, **цикл** `for`, тіло якого виконується стільки разів, скільки задано в умові `6`.

```
void setup() {
  pinMode(8, OUTPUT);
}

void loop() {
  for (int i=0; i < 5; i++) {
    blinkDiode();
  }

  delay(1000);
}

void blinkDiode() {
  digitalWrite(8, HIGH);
  delay(300);
  digitalWrite(8, LOW);
  delay(200);
}
```

Лістинг 3. Скетч програми «Завдання 1» із використанням користувацьких функцій.

В лістингу 3 нова користувацька функція залежить від конкретної схеми, запропонованої на Рис. 1., в новій функції з'явилось число `8`, яке незрозуміле при розгляданні функції поза межами задачі. Як можна позбутись такої залежності? Лістинг 4 демонструє написання незалежної функції.

```
void setup() {
  pinMode(8, OUTPUT);
}

void loop() {
  for (int i=0; i < 5; i++) {
    blinkDiode(8, 300, 200);
  }
  delay(1000);
}
```

Лістинг 4. Скетч програми «Завдання 1» із використанням користувацької функції із передачею параметрів.

В цьому прикладі у функцію передаються всі необхідні для роботи

параметри: номер піна, до якого підключено світлодіод, час на який треба увімкнути світлодіод. Функція не залежить від конкретної збірки, і даний приклад демонструє важливість передбачення різних моментів при вирішенні задачі та потребує розуміння вивчення задачі не «в лоб».

Завдання 2. Передати з однієї плати Arduino на іншу повідомлення “Hello, world!” за допомогою можливостей радіо-модуля NRF24L01.

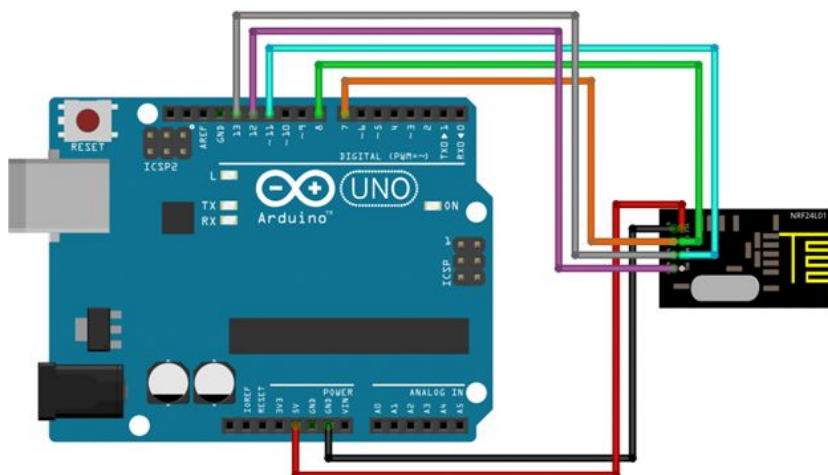


Рис.1. Схема макету. Плата Arduino Uno і радіо-модуль NRF24L01.

Завдяки готовим бібліотекам робота із модулем NRF24L01 полягає в простому першочерговому налаштуванні модуля й не потребує від програміста знань із області електроніки, достатньо лише розуміння рівнів передачі OSI. Основними питаннями перед розробником будуть «Що передавати?» та «Як приймати?».

Над логікою роботи треба буде подумати, адже передавати дані можна, побітно, побайтно, «пакетом» – масивом байтів. На лістингу 5 показано вихідний код скетчу для успішного налаштування модуля-передавача.

Налаштування модуля-приймача майже аналогічне, окрім декількох деталей. В показаних налаштуваннях `radio.openReadingPipe(address[0])` – встановлює ім'я поточного модуля на «1Node» з наперед заданого масиву `address[]`. Для приймача, цей рядок буде інший: `radio.openReadingPipe(1, address[0])` – вказується, що поточний модуль є першим, і слухає все, що має назву «1Node». Другою відмінністю налаштувань буде початок й закінчення прослуховування для різних модулів:

radio.stopListening() та radio.startListening() для відправника та приймача відповідно.

```
#include <SPI.h>
#include "nRF24L01.h"
#include "RF24.h"

RF24 radio(9,10);

byte address[][6] = {"1Node","2Node","3Node","4Node","5Node","6Node"};

void setup(){
  Serial.begin(9600);

  radio.begin();
  radio.setAutoAck(0);
  radio.setRetries(0,15);

  radio.setPayloadSize(32);

  radio.openWritingPipe(address[0]);
  radio.setChannel(0x60);

  radio.setPALevel(RF24_PA_MAX);
  radio.setDataRate(RF24_250KBPS);

  radio.powerUp();
  radio.stopListening();
}

void loop(void) {
}
```

Лістинг 5. Скетч програми «Завдання 2» із першочерговими налаштуваннями модуля NRF24L01 для передавача.

Вся робота тіла функції loop(), в рамках поставленої задачі, для відправника має виклик лише однієї функції: radio.write(const void *buf, uint8_t len) – де перший аргумент, це – дані що ми передаємо, а другий – розмір змінної що ми передаємо. Для приймаючої сторони в циклі функції loop() треба створити ще один цикл, що «слухає ефір» та в разі надходження сигналу, зчитує його (див. Лістинг 6.).

Розглянута постановка задачі є досить простою при умові, що приклад буде лише демонстрацією роботи із модулем. Повноцінна задача передачі даних потребує справді складного алгоритму. В розглянутому прикладі приймач читає з ефіру повідомлення розміру str. Якщо ж довжина пакетів невідома, то

необхідно реалізувати буфер прочитаних з ефіру даних.

```
void loop() {  
  while ( radio.available(pipeNo) ) {  
    radio.read( &str, sizeof(str) );  
    Serial.print("<< ");  
    Serial.println(str);  
  }  
}
```

Лістинг 6. Скетч функції loop() модуля-приймача програми «Завдання 2».

Подібно можна реалізувати й відправлення повідомлень: відправляти масив даних по-байтово (першим байтом повідомлення вказувати його довжину, іншими байтами саме повідомлення). Приймаюча сторона має прийняти таке повідомлення, «зрозуміти» аналог заголовка, правильно прочитати і розібрати повідомлення.

Основна задача при роботі з Arduino – програмування, але завдяки простоті платформи можна протестувати код на реальному пристрої відмінному від великого комп’ютера. Усвідомлення того, що можна реалізувати простий приймач і передавач як маленький, незалежний пристрій породжує інтерес до теми та надихає на більш поглиблене вивчення. «А якщо створити інший приймач і спробувати перехопити повідомлення?», «Як визначити правильного адресата та надсилати повідомлення конкретному приймачу?», «Як зашифрувати дані?» і багато інших питань можна перетворити в «не сухі» теоретичні лабораторні роботи, де результатом буде не просто програма на комп’ютері, а окремий незалежний пристрій.

Висновки з дослідження і перспективи подальших розробок.

Arduino – платформа де рамки встановлює лише фантазія людини. Arduino легка в освоєнні платформа, яка не потребує глибоких знань в алгоритміці чи фізиці або електротехніці. Arduino – це конструктор яким цікаво гратись та граючись можна пізнавати та засвоювати матеріал набагато якісніше. Розглянуті приклади лабораторних робіт не є складними, проте вони демонструють, що платформу Arduino можна застосувати навіть на низькому рівні володіння програмуванням. Основна ціль Arduino у розглянутому напрямі

– зацікавленість студента та не сухі теоретичні відомості «як працює програма». Для платформи можна адаптувати безліч лабораторних робіт, існує чимало збірників, програм і курсів [4], розроблених спеціально для вступних занять з вивчення програмування.

Платформа може бути корисною й для школи. При знайомстві з Arduino в шкільній програмі у дітей буде зацікавленість і певне розуміння, що можна робити за допомогою програмування, що цикли та змінні це не лише комірки пам'яті комп'ютера, це реально працюючий пристрій який може світитись або їздити за певним алгоритмом.

СПИСОК ДЖЕРЕЛ

1. Kushner D. The Making of Arduino [Електронний ресурс] / David Kushner – Режим доступу до ресурсу: <https://spectrum.ieee.org/geek-life/hands-on/the-making-of-arduino>. – Дата перегляду: 12.10.2018.
2. Болілий В. О. Інформаційний освітній простір Кіровоградського державного педагогічного університету / Василь Олександрович Болілий, Вікторія Володимирівна Копотій // Наукові записки КДПУ. Серія: Проблеми методики фізико-математичної і технологічної освіти / ред. кол.: С. П. Величко [та ін.]. – Кропивницький: КДПУ ім. В. Винниченка, 2016. – Вип. 10, ч. 3. – С. 107-112.
3. Застосування електронного навчання для підготовки й підвищення кваліфікації фахівців ІТ-галузі у вищих навчальних закладах [Текст]: монографія / А.В. Васильєв, Ю.О. Зубань, Ю.М. Коровайченко, С. М. Шкарлет. - Суми: СумДУ, 2013. - 138 с.
4. Ким Т. Ю., Артикбаев М. А., Маринина Е. В. Роль студентов в использовании платформы Arduino в высших учебных заведениях // Образование и воспитание. — 2018. — №3.1. — С. 18-20. — Режим доступу до ресурсу: <https://moluch.ru/th/4/archive/94/3369/> – Дата перегляду: 12.10.2018.
5. Курс «Arduino для начинающих» [Електронний ресурс] – Режим

доступу до ресурсу: <http://edurobots.ru/kurs-arduino-dlya-nachinayushhix/>. – Дата перегляду: 12.10.2018.

6. Остапчук С. А. До проблеми використання платформи Arduino у вивченні робототехніки / Сава Адамович Остапчук, Микола Ілліч Садовий // Наукові записки ЦДПУ. Серія: Педагогічні науки = Research Bulletin. Series: Pedagogical Sciences / ред. кол.: В.Ф. Черкасов [та ін.]. – Кропивницький : РВВ ЦДПУ ім. В. Винниченка, 2018. – Вип. 168. – С. 178-181.

7. Соменко Д.В., Соменко О.О. Вільнопоширюване апаратне та програмне забезпечення для організації навчально-дослідницької роботи майбутніх вчителів природничо-математичних дисциплін // Наукові записки КДПУ. Серія: Проблеми методики фізико-математичної і технологічної освіти / ред. кол.: С. П. Величко [та ін.]. – Кропивницький : КДПУ ім. В. Винниченка, 2017. – Вип. 11, ч. 1. – С. 122-128.

REFERENCES

1. Kushner D. The Making of Arduino [WWW document] / David Kushner – URL <https://spectrum.ieee.org/geek-life/hands-on/the-making-of-arduino>. – (12.10.2018).
2. Bolilyj, V.O., Kopotiy, V.V. (2016). Informatsiinyi osvittii prostir Kirovohradskoho derzhavnoho pedahohichnoho universytetu. [Information Educational Environment of Kirovohrad State Pedagogical University]. Naukovi zapysky. –Vypusk 10. –Serii: Problemy metodyky fizyko-matematychnoi i tekhnolohichnoi osvity. Chastyna 2, 107-112.
3. A.V. Vasy`l`yev, Yu.O. Zuban`, Yu.M. Korovajchenko, S.M. Shkarlet. Zastosuvannya elektronnoho navchannya dlya pidgotovky` j pidvy`shhennya kvalifikaciyi faxivciv IT-galuzi u vy`shhy`x navchal`ny`x zakladax. [Application of e-learning for training and advanced training of IT specialists in higher education institutions]. Summy, 2013.
4. Kim T. Yu., Artikbaev M. A., Marinina E. V. Rol studentov v ispolzovanii

- platformy Arduino v vysshih uchebnyh zavedeniyah [The role of students in using the Arduino platform in higher education institutions] 2018. URL <https://moluch.ru/th/4/archive/94/3369/> (12.10.2018).
5. Course «Arduino for begginers» [WWW document] – URL <http://edurobots.ru/kurs-arduino-dlya-nachinayushhix/>. – (12.10.2018).
 6. Ostapchuk S.A. Do problemy` vy`kory`stannya platformy` Arduino u vy`vchenni robototexniki` [TO THE PROBLEM OF USING THE ARDUINO PLATFORM IN THE STUDY OF ROBOTICS] Research Bulletin. Series: Pedagogical Sciences. – Kropyvninsky, 2018.
 7. Dmytro V. Somenko, Olena O. Somenko. Vil`noposhy`ryuvane aparatne ta programne zabezpechennya dlya organizaciyi navchal`no-doslidny`cz`koyi roboty` majbutnix vchy`teliv pry`rodny`cho-matematy`chny`x dy`scy`plin [FREE DISTRIBUTED HARDWARE AND SOFTWARE FOR ORGANIZATION OF TEACHING AND RESEARCH WORK OF THE FUTURE TEACHER OF NATURAL AND MATHEMATICAL SCIENCE]. – Kropyvninsky, 2017.

ВІДОМОСТІ ПРО АВТОРІВ

БОЛІЛИЙ Василь Олександрович – кандидат фізико-математичних наук, доцент кафедри інформатики та ІТ Центральноукраїнського державного педагогічного університету імені Володимира Винниченка.

Наукові інтереси: диференціальні рівняння, задачі з точками звороту; проблеми модернізації навчального процесу; ІКТ у освіті; технології дистанційного навчання.

Олійник Владислав Михайлович – магістрант кафедри інформатики та ІТ Центральноукраїнського державного педагогічного університету імені Володимира Винниченка.

Наукові інтереси: веб-програмування, веб-дизайн, проектування користувальницьких інтерфейсів, користувацький досвід

INFORMATION ABOUT THE AUTHORS

Bolilyj Vasyl Oleksandrovych – Candidate of Sciences (Physics and Mathematics), Associate Professor of the Department of Informatics and Information Technologies of the Volodymyr Vynnychenko Central Ukrainian State Pedagogical University.

Circle of research interests: differential equations; problems with turning points; problems of teaching process modernization; ICT in education; distance learning technology.

Oliinyk Vladyslav Mykhailovych – graduate student of the Department of Informatics and Information Technologies of the Volodymyr Vynnychenko Central Ukrainian State Pedagogical University.

Circle of research interests: web-programming, web-design, UI/UX

**БОЛІЛИЙ Василь Олександрович, ОЛІЙНИК Владислав Михайлович.
ВИКОРИСТАННЯ ПЛАТФОРМИ ARDUINO ПРИ ВИВЧЕННІ
ПРОГРАМУВАННЯ**

Анотація. Arduino – фізична обчислювальна платформа з відкритим вихідним кодом, заснована на платі та середовищі розробки. Arduino може застосовуватися для розробки самостійних інтерактивних пристроїв або може бути пов'язана з програмою на комп'ютері. Завдяки своїй простоті, доступності та широкому вибору, Arduino використовується в тисячах різних проектах та додатках. Програмне забезпечення Arduino досить гнучке для досвідчених інженерів та програмістів, і в той самий час, просте у використанні для початківців. З Arduino можна працювати, як на Mac, Linux, так і на Windows.

В статті розглянуто питання використання платформи Arduino при вивченні основ програмування. Показано переваги використання відкритої платформи Arduino. В роботі аналізуються можливі напрямки поєднання традиційних методів та інноваційних технологій у навчанні програмуванню. Пропонуються варіанти використання програмно-апаратної платформи Arduino. Описуються реальні приклади і задачі, які можна адаптувати для інтерактивної роботи з платформою Arduino.

Ключові слова: Arduino, програмування, відкритий код, організація навчального процесу, скетч, програма

**БОЛИЛЫЙ Василий Александрович, ОЛИЙНИК Владислав Михайлович.
ИСПОЛЬЗОВАНИЕ ПЛАТФОРМИ ARDUINO ПРИ ИЗУЧЕНИИ
ПРОГРАМИРОВАНИЯ**

Аннотация. Arduino - физическая вычислительная платформа с открытым исходным кодом, основанная на плате и среде разработки. Arduino может применяться для разработки самостоятельных интерактивных устройств или может быть связана с программой на компьютере. Благодаря своей

простоте, доступности и широкому выбору, Arduino используется в тысячах различных проектах и приложениях. Программное обеспечение Arduino достаточно гибкое для опытных инженеров и программистов, и в то же время, простое в использовании для начинающих. С Arduino можно работать, как на Mac, Linux, так и на Windows.

В статье рассмотрены вопросы использования платформы Arduino при изучении основ программирования. Показаны преимущества использования открытой платформы Arduino. В работе анализируются возможные направления сочетания традиционных методов и инновационных технологий в обучении программированию. Предлагаются варианты использования программно-аппаратной платформы Arduino. Описываются реальные примеры и задачи, которые можно адаптировать для интерактивной работы с платформой Arduino.

Ключевые слова: *Arduino, программирование, открытый код, организация учебного процесса, скетч, программа*

BOLILYJ Vasyl Oleksandrovyeh, OLIINYK Vladyslav Mykhailovych. PLATFORM ARDUINO IN STUDY OF THE PROGRAMMING

Abstract. *Arduino is a physical open source computing platform based on the board and development environment. Arduino can be used to develop standalone interactive devices or it can be linked to a program on a computer. Boards can be created on one's own or a pre-made kit can be bought. Due to its simplicity, accessibility and wide selection, the board have been used in thousands of different projects and applications. The Arduino software is flexible enough for experienced engineers and programmers, and at the same time, easy to use for beginners. Arduino can work on Mac, Linux or Windows. Teachers and students use Arduino to create low-cost scientific tools for research and education in chemistry and physics, or to start working on programming and robotics. Arduino simplifies the working process with microcontrollers and has some significant benefits over other systems for teachers, students and other interested people.*

Not every student wants to be a web developer or a software developer for a computer or phone. Some students prefer to develop hardware and they need to understand the subject from the basics. This opportunity is provided by the Arduino platform with its low entry threshold. For those who dream of working in the software and hardware development field, Arduino platform will be very useful from a practical point of view.

The purpose of the article is to describe and justify the feasibility of using the elements of the Arduino platform on the programming courses in university.

The article describes the possibility to use the Arduino platform in studying the basics of programming. The advantages of using the open Arduino platform are shown. In the article the possible directions of combining traditional methods and innovative technologies in studying programming subjects are analyzed. cases for using the Arduino platform are suggested. real examples and tasks that can be adapted to interact with the Arduino platform are described.

Key words: *Arduino, programming, open source, organization of educational process, sketch, program*