Міністерство освіти і науки України

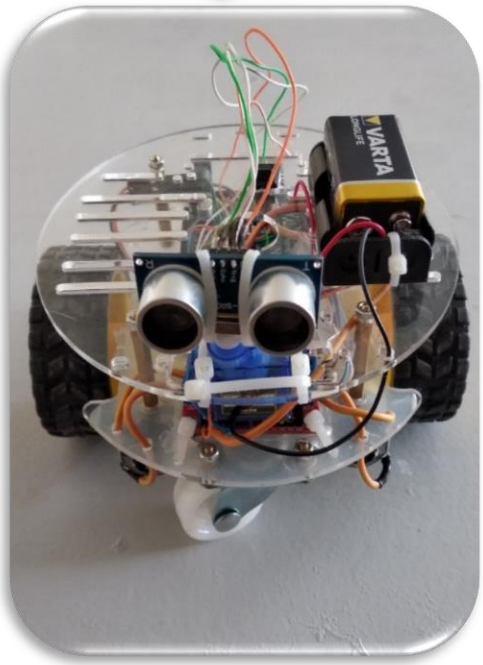
Кам’янський державний енергетичний технікум

Циклова комісія природничо – наукових дисциплін

Звіт

Гуртково-проєктної роботи

«Роботехніка»



Керівники гуртка:

викладач фізики Дорожка Т. М.

викладач математики і інформатики Шматок І.О.

Кам‘янське 2021

Паспорт гуртка:

Анотація: Дана робота полягає в підготовці здобувачів освіти до використання інформаційних технологій і засобів інформатизації при розв'язуванні поставлених перед ними творчих завдань. Кінцевим результатом проведеної роботи є сконструйований робот на платі ардуіно, перевага цієї платформи полягає в тому, що для її програмування не потрібен програматор. Написання програми здійснюється за правами схожості у мові С++. Дана конструкція робота є доступною, її можна удосконалювати. Даний робот також можна впроваджувати на практиці технічних спеціальностей.

Мета: використання навчальних конструкторів до досліджень у галузі робототехніки, обміну технічною інформацією і початковими інженерними знаннями, розвитку нових науково-технічних ідей у різних галузях фізики та інформатики, що дасть студентам конкурентну перевагу у сучасному житті.

Навчання за допомогою робототехніки надає можливість здобувачам освіти вирішувати реальні життєві проблеми, які потребують знань предметів зокрема:

- Математика (просторові поняття, геометрія – для розуміння способу руху роботів);
- Фізика (електроніка, принципи роботи датчиків, що є основою роботів);
- Технологія та дизайн (дизайн пристроїв частин роботів, їх конструювання);
- ІКТ (програмування робототехнічних систем)

Завдання гуртка:

• формувати уміння проектувати моделі роботів та їх збирати, створювати алгоритми та забезпечувати їх програмну реалізацію; розвивати навички роботи в середовищі операційної системи та мови програмування;

• розвивати уміння використовувати інформаційно-комунікаційні технології для ефективного розв'язання нетипових завдань щодо отримання та подання інформації через фізичні пристрої, обробки цих даних процесорним блоком, збереження даних для подальшого опрацювання;

• формувати науковий світогляд, як невід'ємну складову загальної культури людини; стійку мотивацію до навчання, до пізнання нового;

• інтелектуально розвивати особистість, розвивати у слухачів логічне мислення, пам'ять, увагу, інтуїцію;

• забезпечувати фізичне, екологічне, естетичне, громадянське виховання та формувати позитивні риси особистості.

План роботи гуртка

№ з/п	Тема	Дата проведення
1	Організаційні моменти роботи гуртка. Мета та задачі гуртка «Робототехніка». Запис студентів на гурток	
2	Використання інформаційних джерел для наукового пошуку	
3	Складання покрокового плану реалізації проекту. Знайомство з конструкторними моделями роботів для замовлення.	
4	Знайомство з схемами складання конструкцій	
5	Збір машини за схемою та деталями	
6	Загрузка і корегування інформаційного коду програми для механічного руху машини	
7	Тестування машині у роботі при використанні установленої програми	
8	Мультимедійна демонстрація механізму над яким працювали учасники гуртка	
9	Підведення підсумків роботи гуртка	

Теоретично – практична частина проекту

Передумовою до створення робота, була мета розвинути у студентів здібності: робототехніків, електронщиків, програмістів. Данний проект на різних етапах свого створення вимагає тих чи інших здібностей які були описані вище.

1. Конструкція робота

Робот – це сукупність механічних та електронних компонентів які взаємодіють зв'язані між собою. Конкретно наш робот складається з основи до якої кріпляться всі компоненти.

1.1 Основа виконана у виді кругів з органічного скла(плексигласу), які мають сквізні отвори та вирізи для кріплення датчиків, дротів і т.д. Між собою вони з'єднані чотирма спеціальними гвинтами в основі яких лежить шестикутник, з одного кінця якого знаходиться зовнішня різьба, а з іншого внутрішня.

Рисунок -1

1.2 Робот має 4 точки опори, цими опорами виступають два ведучих колеса та два допоміжних колеса.

Ведучі(основні) колеса - мають більший діаметр та резинове обрамлення з характерним протектором, який допомагає в створенні більшої сили тертя між поверхнею та колесом. Основні колеса можуть рухатися по одній осі.

Допоміжні колеса меншого діаметра і виконані з пластику без резинового обрамлення. Ці колеса потрібні для створення рівноваги, та частково для задання

напрямку руху, вони можуть рухатись в двох осях, але точно та безпосередньо керувати ними – не вийде.

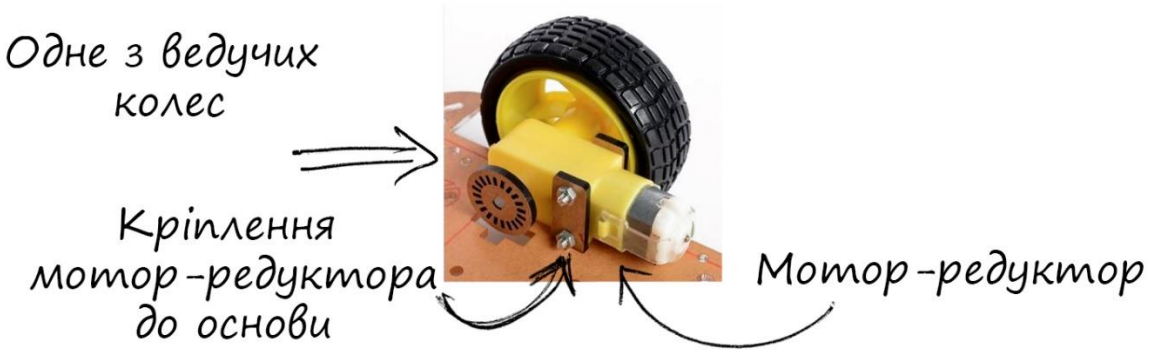


Рисунок - 2

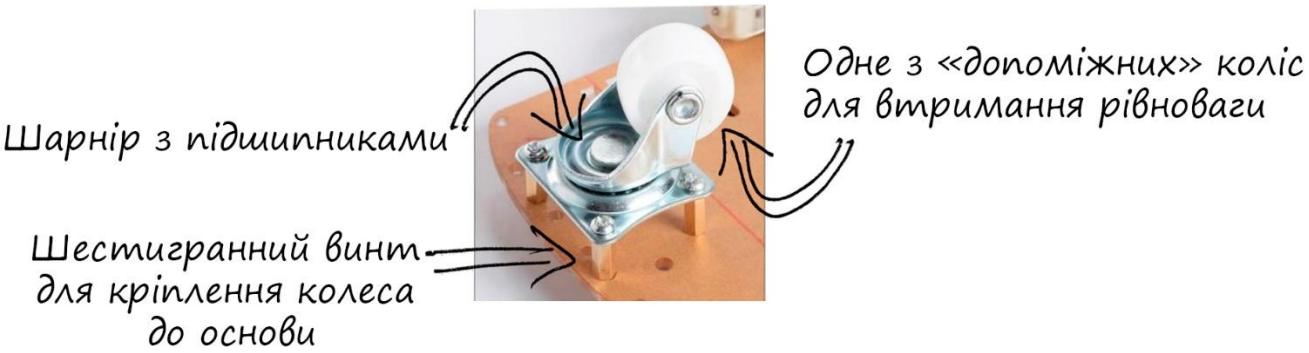


Рисунок – 3

2. Електромеханічна частина

Рух робота забезпечується за допомогою двох двигунів-редукторів. Двигун редуктор – це завершена конструкція в якій вихідний вал двигуна одразу з’єднується в редуктором який також має вихідний вал. Редуктор потрібний для однієї цілі – або зменшити обороти вихідного валу з більшим зусиллям в порівняні за стартовими показниками двигуна, або навпаки збільшити обороти вихідного валу з меншими зусиллям. Інакше кажучи: за допомогою редуктора ми можемо або збільшити швидкість і зменшити тягові показники, або навпаки зменшити швидкість, але збільшити тягові здібності двигуна.

Вихідні вали редукторів безпосередньо з’єднується з основними колесами рисунок 4.

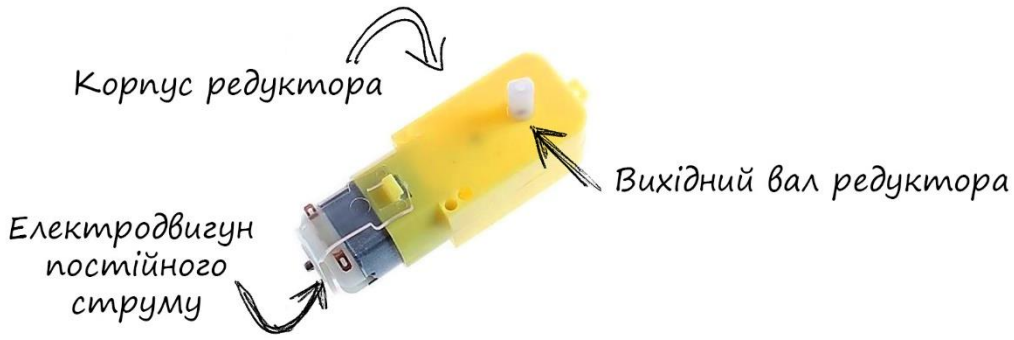


Рисунок – 4

В нашому роботі використовуються електродвигуни постійного струму колекторного типу. Основні технічні характеристики двигуна:

Напруга живлення: 3-6 Вольт;

Передатне число: 1:48;

Споживаємий струм: 70 мА;

Максимальний струм (при 6В): 670 мА;

Вага: 30г;

Швидкість без навантаження: 180 об/хв (при 6В);

Обертовий момент: 2 кг/см (при 6В).

3. Електронна частина

Мозком всього робота є – мікроконтролер ATmega328p. Оскільки мікроконтролер потрібно живити, під'єднати до датчиків, двигунів та іншої периферії, а також найголовніше – завантажувати в нього мікропрограму(скетч), зручним для нас методом(через порт USB нашого ПК), цей мікроконтролер розпаяний на платі та виконаний у виді платформи Ардуіно (Arduino). Це дозволяє з легкістю під'єднати нові датчики(кнопки, дисплеї, мікрофони, світлодіоди, двигуни, динаміки, і ще дуже багато чого), та від'єднувати старі, змінювати схему та модернізувати під свої потреби або зробити нового робота.

Вигляд мікроконтролера в різних корпусах без обв'язки: рисунок 5,6

Рисунок - 5

Рисунок – 6

До такого чіпа дуже важко під’єднати різні кнопки, датчики, двигуни, а тим більше програмувати його, та якщо схема планується модернізуватися.

Тому ми використовуємо платформу Ардуіно. Платформа Ардуіно має велику кількість різних плат з різними контролерами, розміром пам’яті, кількістю входів/виходів та різними габаритами. Конкретно у нашому випадку ми використовуємо Ардуіно Уно (Arduino UNO) рисунок 7.

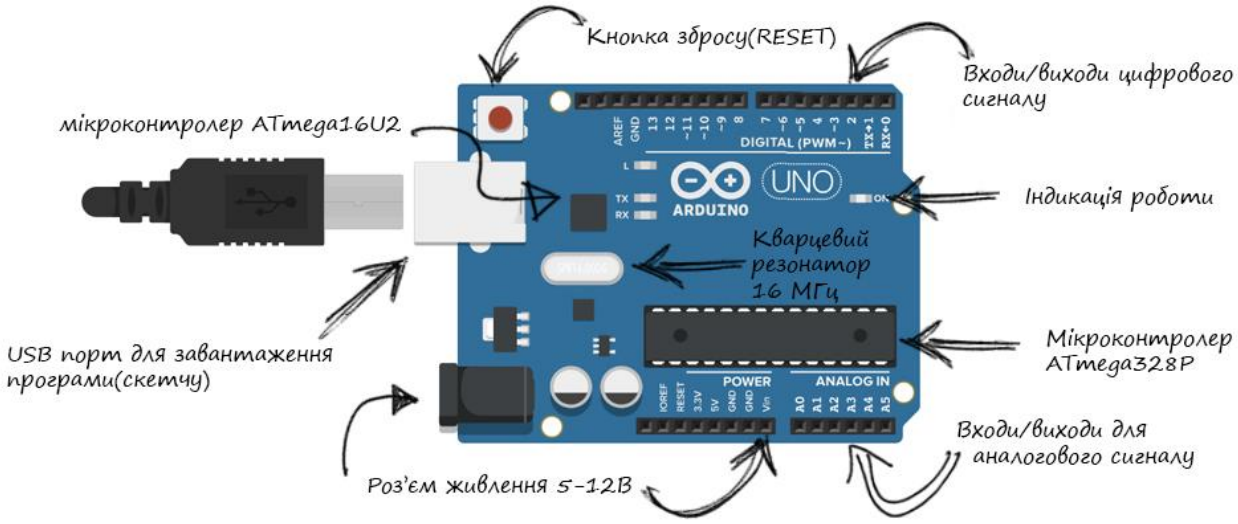


Рисунок – 7

Мікроконтролери дуже «ніжні» пристрої, вони можуть керувати великою кількістю пристроїв, але не можуть крізь себе пропускати великий струм. Мікроконтролер просто приймає та відправляє керуючий сигнал напругою від 0 до 5 вольт з максимальним струмом 40 мА. Виходячи з цього, керувати двома двигунами безпосередньо під’єднавши їх до Ардуіно – не вийде, бо плата вийде з ладу. Тому для керування двигунами, нам потрібен посередник – драйвер двигунів(з англ driver – водій). Драйвери бувають для різних типів двигунів, з різними характеристиками і т.д. Драйвер представляє собою плату на якій основний компонент – чіп, який повинен приймати керуючий сигнал з нашого

мікроконтролера і в залежності від цього – керувати ввімкненням, вимкненням, зміною напряму руху наших двигунів.

В нашому роботі ми використовуємо драйвер L298N, рисунок 8.

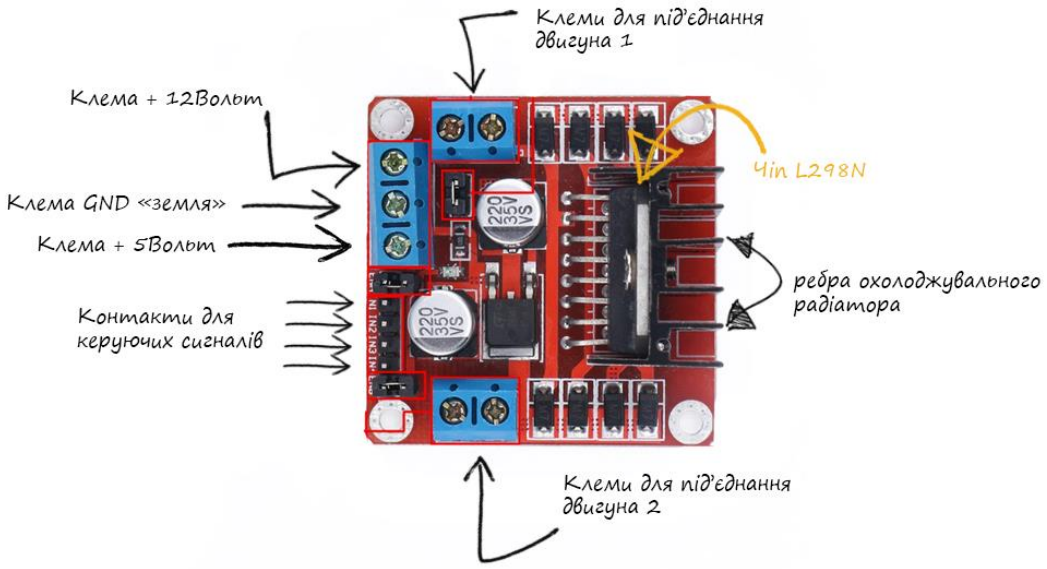


Рисунок – 8

Тепер мікроконтролер здатний керувати моторами за допомогою драйвера.

Ми можемо керувати напрямом оберту коліс і поєднуючи різні комбінації, робот може їхати вперед, назад, вправо, вліво та зупинятися.

Оскільки робот повинен уникати перешкоди, йому потрібно відчувати де знаходиться перешкода, це можна реалізувати за допомогою різних датчиків або систем датчиків. В нашому випадку, робот «відчуває» що попереду нього знаходиться перешкода, за допомогою ультразвукового датчика. Його принцип дуже простий. Знаючи швидкість відправленого сигналу, рахується час між відправленням та отриманням цього сигналу, так знаходиться відстань. Ми використовуємо ультразвуковий дальномір HC-SR04.

Основні технічні характеристики:

Напруга живлення: 5 В

Споживання в режимі спокою: 2 мА

Споживання при роботі: 15 мА

Діапазон відстаней: 2–400 см

Ефективний кут спостереження: 15°

Робочий кут спостереження: 30°

Рисунок – 9

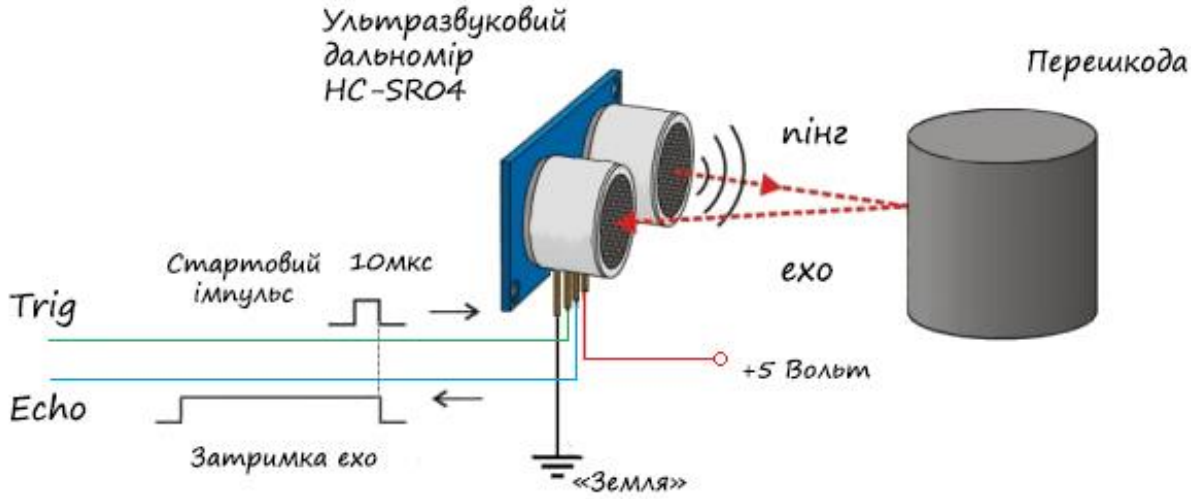


Рисунок -10

Оскільки ефективний кут спостереження даного дальноміра 15°, вочевидь цього не достатньо для гарного результату і точного визначення перешкоди, тому наш датчик потрібно повертати вліво та вправо, щоб зробити замір. Для цього ми використовуємо кроковий двигун SG-90, рисунок 11.

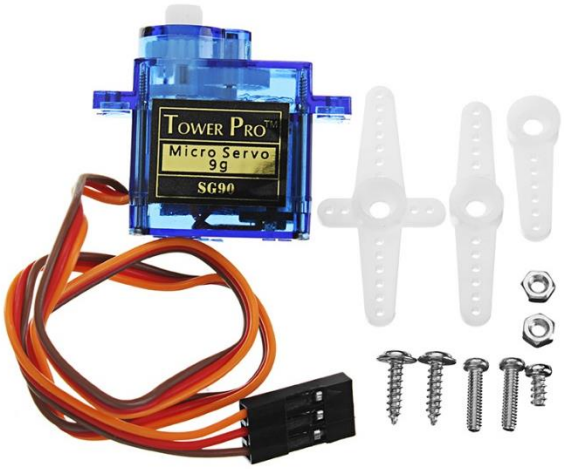


Рисунок – 11

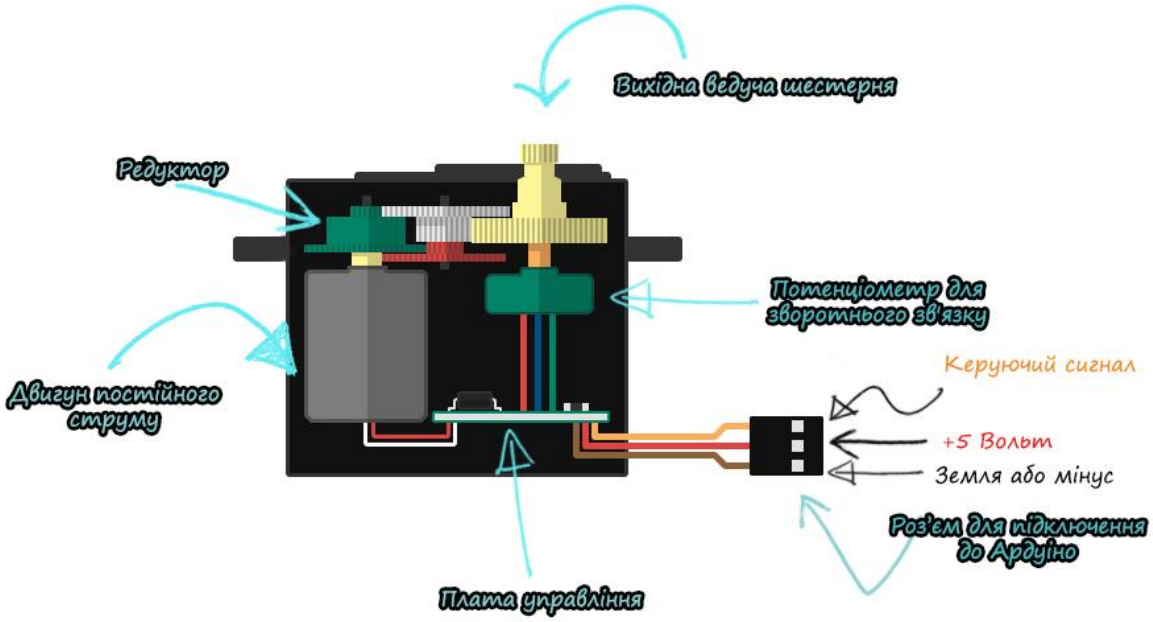


Рисунок -12

Внутрішня конструкція та принцип дії можна побачити на рисунку 12.

Основні технічні характеристики крокового двигуна SG-90:

Вага: 9 грам

Розміри 21.5мм X 11.8мм X 22.7мм

Напруга живлення 4,8 ... 6 В

Момент (зусилля) 1.2 кг*см при напрузі 4,7В

Час повороту на 60° 0,12 с. при напрузі 4,7В

4. Програмна частина

Великою перевагою платформи ардуіно, є те що для її програмування не потрібен програматор, це значно полегшує роботу з платформою. Для написання

коду(скетчу) для ардуіно, використовується середовище під назвою «Arduino IDE». Написання програми здійснюється за правилами схожими в мові C++ , синтаксис, оператори і т.д все взято з мови C++.

Рисунок -13

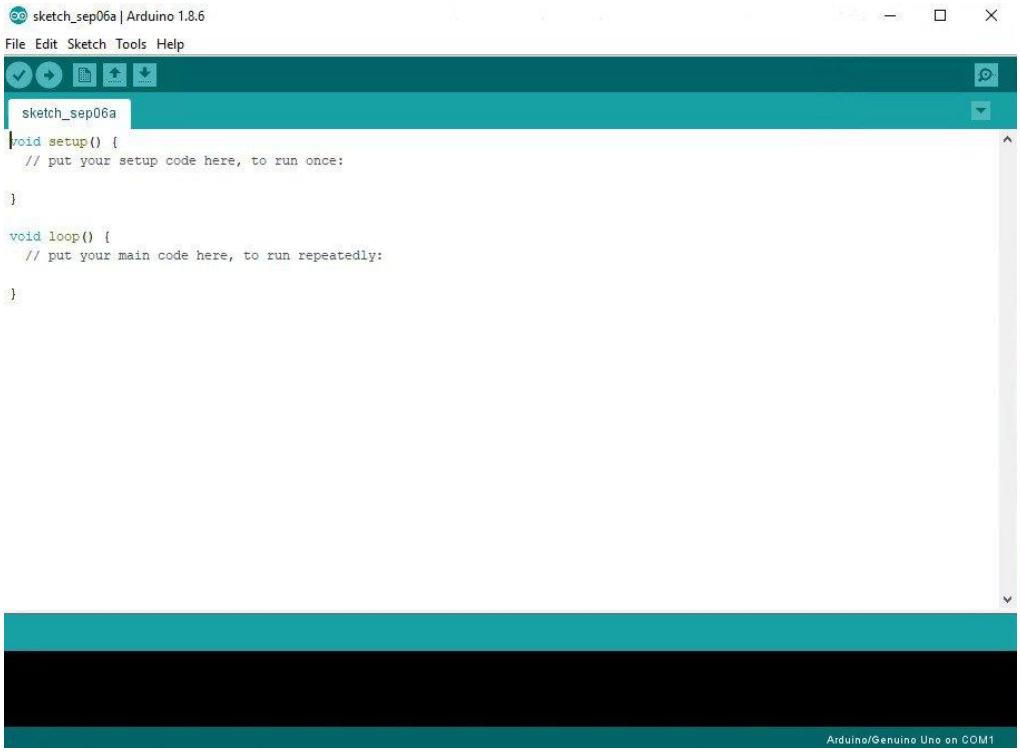


Рисунок - 14

Наша програма повинна приймати дані з ультразвукового датчика, який буде повертатися на деякий кут вліво та вправо, потім в залежності від того, чи є перешкода, робити поворот робота колесами, приблизно на 180 градусів і задавати рух вперед, при цьому аналізуючи сенсором можливу перешкоду попереду.

Лістинг програми:

```
#include <Servo.h>          //Библиотека сервомотора. Есть по стандарту
#include <NewPing.h>     //Библиотека Ультразвукового датчика (нужно установить)
//Пины контроллера L298N
```

```
const int LeftMotorForward = 10;
const int LeftMotorBackward = 9;
const int RightMotorForward = 6;
const int RightMotorBackward = 5;
//Пины ультразвукового датчика
#define trig_pin A2 //Аналоговый вход 2
#define echo_pin A1 //Аналоговый вход 1
#define maximum_distance 200
boolean goesForward = false;
int distance = 100;
NewPing sonar(trig_pin, echo_pin, maximum_distance); //sensor function
Servo servo_motor; //our servo name
void setup(){
  pinMode(RightMotorForward, OUTPUT);
  pinMode(LeftMotorForward, OUTPUT);
  pinMode(LeftMotorBackward, OUTPUT);
  pinMode(RightMotorBackward, OUTPUT);

  servo_motor.attach(11); //Пин подключения сервомотора
 servo_motor.write(86);
  delay(2000);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
}
void loop(){
  int distanceRight = 0;
```

```
  int distanceLeft = 0;
  delay(50);
  if (distance <= 30){
    moveStop();
    delay(300);
    moveBackward();
    delay(35);
    moveStop();
    delay(300);
    distanceRight = lookRight();
    delay(300);
    distanceLeft = lookLeft();
    delay(300);
    if (distance >= distanceLeft){
      turnRight();
      moveStop();
    }
    else{
      turnLeft();
      moveStop(); }}
  else{
    moveForward();  }
    distance = readPing();}
int lookRight(){
  servo_motor.write(120);
  delay(500);
  int distance = readPing();
  delay(100);
  servo_motor.write(86);
  return distance;}
int lookLeft(){
  servo_motor.write(60);
```

```
  delay(500);
  int distance = readPing();
  delay(100);
  servo_motor.write(86);
  return distance;
  delay(100);}
int readPing(){
  delay(70);
  int cm = sonar.ping_cm();
  if (cm==0){
    cm=250; }
  return cm;}

void moveStop(){
  digitalWrite(RightMotorForward, LOW);
  digitalWrite(LeftMotorForward, LOW);
  digitalWrite(RightMotorBackward, LOW);
  digitalWrite(LeftMotorBackward, LOW);}
void moveForward(){
 if(!goesForward){
 goesForward=true;
    digitalWrite(LeftMotorForward, HIGH);
    digitalWrite(RightMotorForward, HIGH);
   digitalWrite(LeftMotorBackward, LOW);
    digitalWrite(RightMotorBackward, LOW); }}
void moveBackward(){
  goesForward=false;
  digitalWrite(LeftMotorBackward, HIGH);
  digitalWrite(RightMotorBackward, HIGH);
  digitalWrite(LeftMotorForward, LOW);
  digitalWrite(RightMotorForward, LOW);  }
void turnRight(){
```

```
  digitalWrite(LeftMotorForward, HIGH);
  digitalWrite(RightMotorBackward, HIGH);
  digitalWrite(LeftMotorBackward, LOW);
  digitalWrite(RightMotorForward, LOW);
  delay(500);
  digitalWrite(LeftMotorForward, HIGH);
  digitalWrite(RightMotorForward, HIGH);
  digitalWrite(LeftMotorBackward, LOW);
  digitalWrite(RightMotorBackward, LOW);}
void turnLeft(){
 digitalWrite(LeftMotorBackward, HIGH);
  digitalWrite(RightMotorForward, HIGH);
  digitalWrite(LeftMotorForward, LOW);
  digitalWrite(RightMotorBackward, LOW);
 delay(500);
  digitalWrite(LeftMotorForward, HIGH);
  digitalWrite(RightMotorForward, HIGH);
  digitalWrite(LeftMotorBackward, LOW);
  digitalWrite(RightMotorBackward, LOW);
}
```

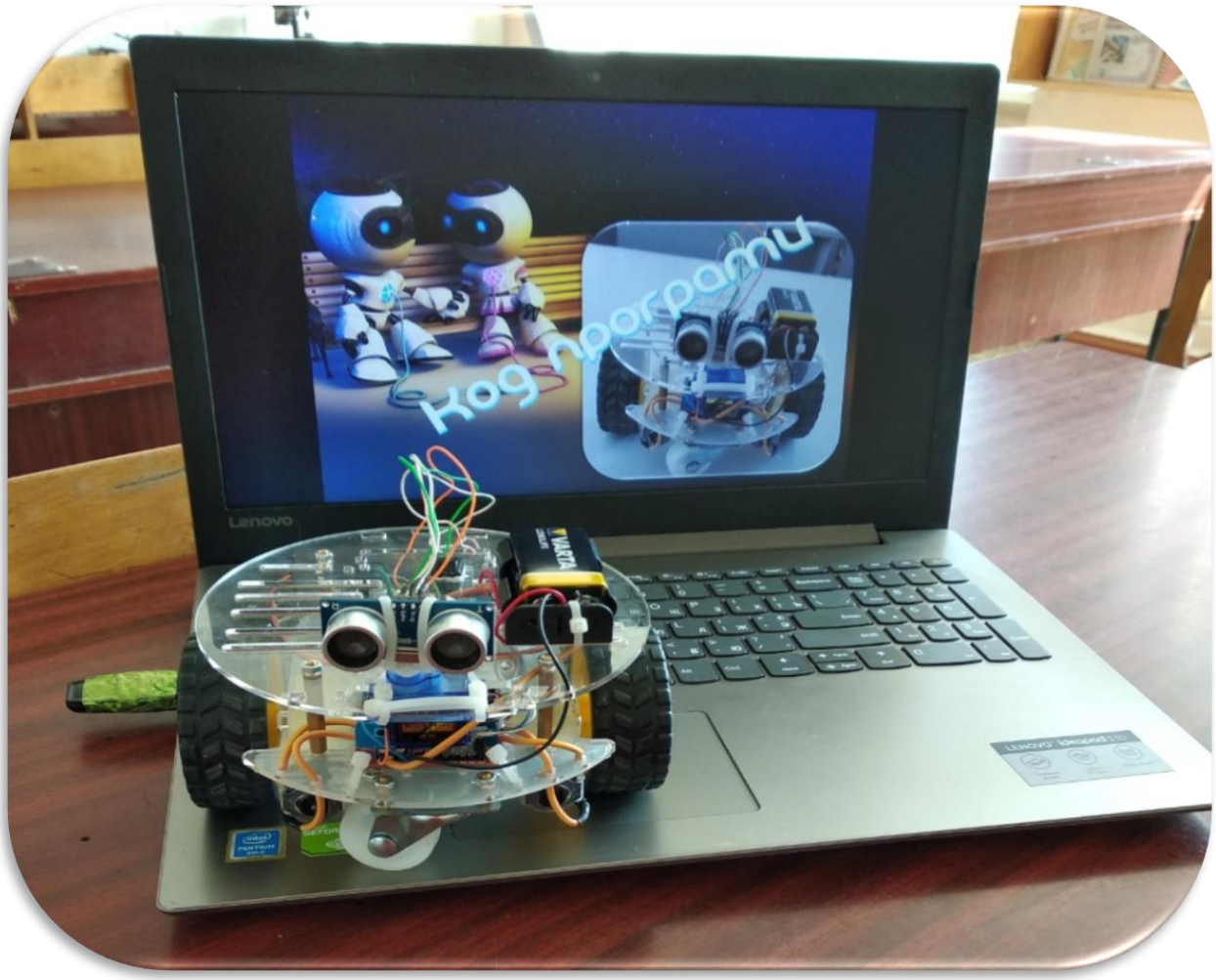
Висновок: навчання робототехніки сприяє розвитку комунікативних здібностей молоді, розвиває навички взаємодії, самостійності при прийнятті рішень, розвиває творчий потенціал студентів.

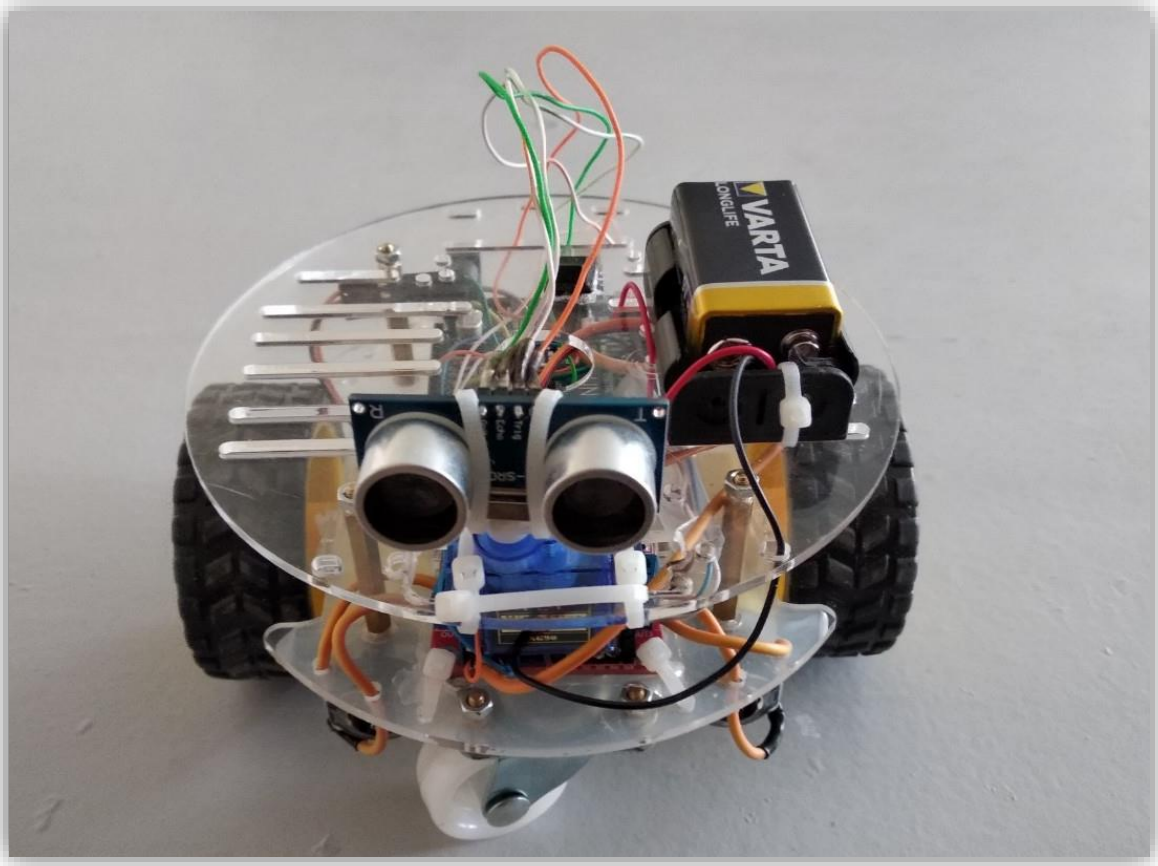
Програмування роботів дозволяє без зусиль організувати міжпредметні зв’язки інформатики з предметами природничо-математичного циклу.

Студенти краще розуміють навчальний матеріал, коли вони що- небуть самостійно створюють або винаходять, і це використовується із врахуванням численних перспектив подальшого розвитку.

OFF

STAND-ALONE


Lenovo
VARTA


VARTA
LONGLIFE