міністерство освіти і науки України

**Прикарпатський національний університет**

**імені В. Стефаника**

*Факультет математики та інформатики*

*Кафедра інформаційних технологій*

*Програмування вбудованих систем*

Лабораторна робота № 2

Тема: «Ультразвуковий дальномір»

Варіант : 16

Виконала: ***Петрів В. В.***

Група ІПЗ-23

Дата: 16 жовтня 2023р.

Викладач: Лазарович І.М.

Івано-Франківськ - 2023

**Завдання**

Реалізувати ультразвуковий далекомір з виведенням відстані на РКІ та послідовний порт. Далекомір повинен працювати по принципу парктроніка: У верхньому рядку РКІ повинна відображатись відстань з періодом оновлення 0,3с, нижній рядок – смужка, яка візуально відображає відстань в діапазоні 0,1…2м, (0.1м – повністю заповнена, 2м – повністю порожна смужка). Якщо відстань до перешкоди менше 1м, то підсвітка РКІ повинна блимати з частотою 1Гц , і помірі зменшення відстані до 0,3м частота повинна збільшитись до 5 Гц пропорційно відстані.

Загальний алгоритм роботи програми:

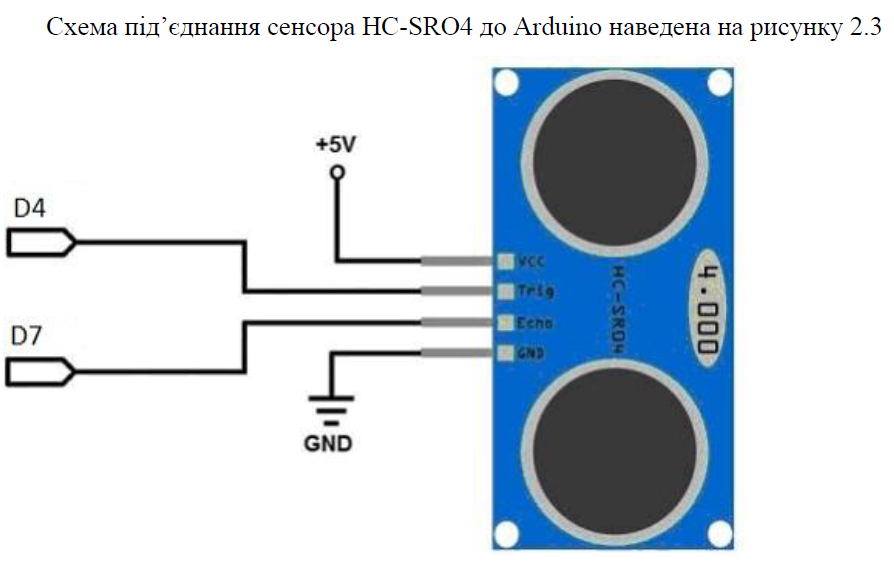
Setup:

* Визначаємо вхідні і вихідні піни для датчика;
* Ініціалізуємо РКІ;

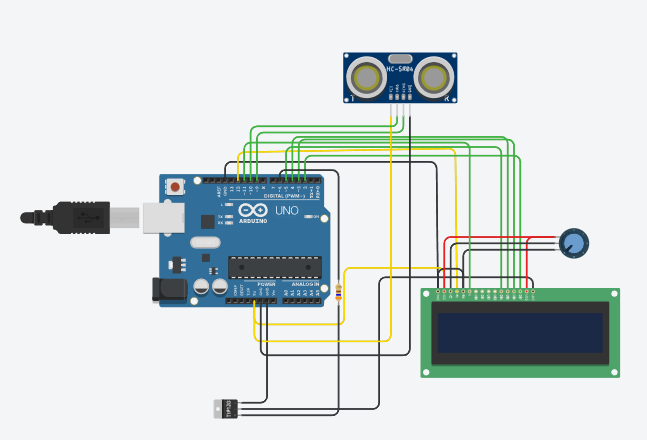
Main loop:

* Генеруємо короткий імпульс тривалістю 5 мікросекунд;
* Виставивши високий рівень сигналу, чекаємо близько 10 мікросекунд. У цей момент датчик посилатиме сигнали з частотою 40 КГц;
* Визначаємо час затримки акустичного сигналу на ехолокаторі;
* Перетворити час у відстань;
* Вивести дані на РКІ та в порт;
* Виконання затримки між вимірами згідно умов задачі.

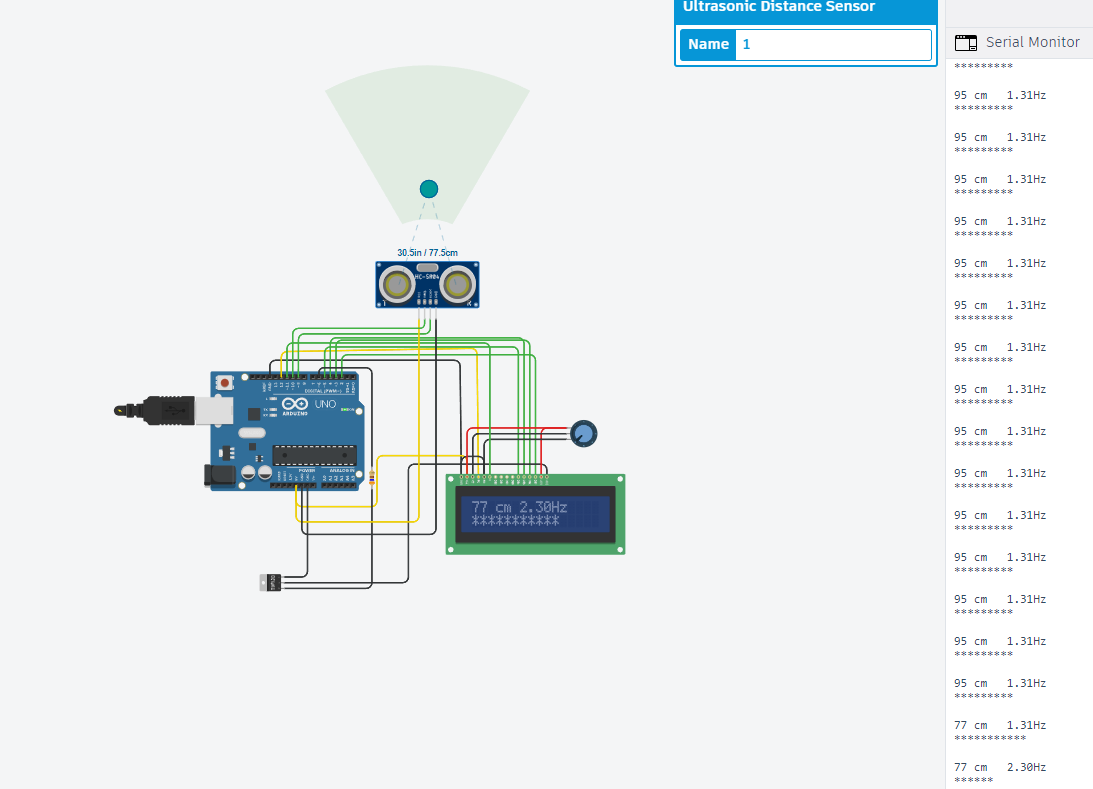
В звіті навести тему, сформульоване завдання, програмний код, схему з’єднань (фото), скріншот результатів виведення з послідовного порта (не менше 15-20 рядків).

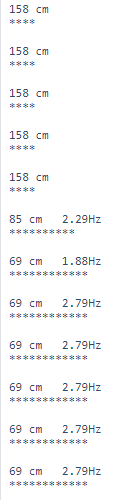
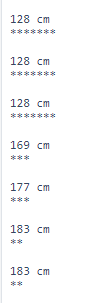


**Схема**

****

**Скрін виконання коду**

****

****

**Програмний код**

#include <LiquidCrystal.h>

const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

unsigned long previousMillis = 0;

unsigned long previousBlink = 0;

int triggerPin = 10, echoPin = 9;

int maxL = 16;

float frequency, distance;

long duration(int triggerPin, int echoPin){

digitalWrite(triggerPin, HIGH);

delayMicroseconds(5);

digitalWrite(triggerPin, LOW);

delayMicroseconds(10);

return pulseIn(echoPin, HIGH);

}

void setup() {

Serial.begin(9600);

lcd.begin(16, 2);

pinMode(triggerPin, OUTPUT);

pinMode(echoPin, INPUT);

}

void blink(unsigned long currentMillis) {

if (currentMillis - previousBlink >= (1. / frequency) \* 1000 && distance <= 100 && distance >= 30) {

previousBlink = currentMillis;

digitalWrite(6, LOW);

delay(100);

digitalWrite(6, HIGH);

}

}

void loop() {

unsigned long currentMillis = millis();

if(currentMillis - previousMillis >= 300){

previousMillis = currentMillis;

distance = (duration(triggerPin, echoPin) / 2) / 29.1;

lcd.clear();

lcd.setCursor(0, 0);

lcd.print(distance, 0);

lcd.print(" cm ");

Serial.print(distance, 0);

Serial.print(" cm");

Serial.print("\t");

if (distance <= 100 && distance >= 30) {

lcd.print(frequency, 2);

lcd.print("Hz");

Serial.print(frequency, 2);

Serial.print("Hz");

frequency = (235. - 2 \* distance) / 35;

}

Serial.print("\n");

lcd.setCursor(0, 1);

int length = map(distance, 10, 200, maxL, 0);

for(int i = 0; i < maxL; i++){

if(i < length){

lcd.print("\*");

Serial.print("\*");

}

}

Serial.print("\n\n");

}

blink(currentMillis);

}