

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені І.І.МЕЧНИКОВА

(повне найменування вищого навчального закладу)

Факультет математики, фізики та інформаційних технологій

(повне найменування інституту, назва факультету (відділення))

Кафедра системного програмного забезпечення та технологій дистанційного навчання

(повна назва кафедри (предметної, циклової комісії))

**Дипломна робота**

на здобуття освітньо-кваліфікаційного рівня «бакалавр»

(освітньо-кваліфікаційний рівень)

на тему Розробка мікроконтролерного пристрою для визначення  
геометричних розмірів приміщень / Development of microcontroller device for  
determination of geometric sizes of premises

Виконала: студентка заочної форми навчання  
напряму підготовки 6.050102 Комп'ютерна інженерія  
(шифр і назва напряму підготовки, спеціальності)

Дейнекіна Ніна-Марія Давидівна

(прізвище, ім'я, по-батькові)

Керівник старший викладач Берков Ю.М.  
(науковий ступінь, вчене звання, прізвище та ініціали, підпис)

Рецензент к.т.н., доцент Пенко В.Г.  
(науковий ступінь, вчене звання, прізвище та ініціали)

Рекомендовано до захисту:

Протокол засідання кафедри

№      від «    »      2019 р.

Завідувач кафедри

Ю.О. Гунченко  
(підпис) (прізвище, ініціали)

Захищено на засіданні ЕК №     

протокол №      від «    »      2019 р.

Оцінка      /      /       
(за національною шкалою, шкалою ECTS, бали)

Голова ЕК

О.О. Арсірій  
(підпис) (прізвище, ініціали)

## ЗМІСТ

ЗМІСТ .....	5
ВСТУП .....	7
1. СУЧАСНІ ЗАСОБИ ВИМІРУ ВІДСТАНЕЙ.....	8
1.1. Лазерна рулетка.....	8
1.2. Промислові далекоміри .....	9
1.2.1. Далекомір Bosch PLR 30 (побутовий). ....	9
1.2.2. Лазерний далекомір BOSCH DLE 150 Professional. ....	10
1.3. Сучасні датчики відстані.....	10
1.3.1. Ультразвукові датчики відстані .....	11
1.3.2. Оптичні датчики відстані.....	12
1.4. Висновки до глави 1. ....	15
2. ARDUINO – ПЛАТФОРМА ДЛЯ СТВОРЕННЯ ПРОТОТИПІВ.....	16
2.1. Програмувальні відлагоджувальні плати сімейства Arduino .....	16
2.2. Відладочна плата Arduino UNO.....	18
2.2.1. Характеристики.....	18
2.2.2. Входи й виходи .....	19
2.3. Лазерний датчик відстані VL53L0X-V2 .....	21
2.4. Гіросенсори й акселерометри .....	21
2.5. LCD дисплеї.....	23
2.6. Сервопривід .....	24
2.7. Середовище програмування Arduino IDE. Мова Arduino.....	25
2.8. Висновки до глави 2. ....	31
3. РОЗРОБКА ПРИСТРОЮ ВИМІРУ ВІДСТАНІ Й ПЛОЩІ.....	32
3.1. Схема пристрою й конструктивне виконання. ....	32
3.2. Розробка програмного забезпечення для керування пристроєм. ....	34
3.2.1. Бібліотека Liquidcrystal .....	34
3.2.2. Бібліотека Wire.....	34
3.2.3. Бібліотека Servo .....	35
3.2.4. Бібліотека VL53L0X .....	36
3.3. Підключення датчиків до шини I <sup>2</sup> C. ....	37
3.4. Розробка програмного коду .....	42
3.5. Тестування та робота з приладом.....	45

3.6. Висновки до глави 3. ....	49
ВИСНОВКИ.....	50
СПИСОК ВИКОРИСТОВУВАНИХ ДЖЕРЕЛ.....	51
ДОДАТОК А. ПРОГРАМНИЙ КОД ПРОЕКТУ.....	52

## ВСТУП

Вимірювання відстані необхідно в багатьох випадках: як у повсякденному житті, так і у виробничій галузі. Промисловість випускає велику кількість приладів для вимірювання відстаней, але більшість з них мають дуже велику вартість і при цьому не завжди забезпечують бажаний сервіс. Прилади низького цінового діапазону взагалі можуть виконувати тільки найпростіші вимірювання, при цьому не гарантуючи будь яку точність цих вимірювань.

Сучасна мікроконтролерна база дозволяє створити майже будь який прилад, а велике розмаїття датчиків – виконати любі вимірювання. Самостійна розробка програмного керуючого коду дозволяє розробити саме ті режими вимірювання, які потрібні на даний час, а також виконувати необхідні обчислення з отриманими даними.

Основою створення вимірювального пристрою є правильний добір елементної бази, після чого буде створена схема пристрою. Далі має бути збірка макета пристрою за схемою, розробленою раніше й перевірка його працездатності. Одним з головних кроків у процесі розробки є програмування пристрою для його коректної роботи.

Таким чином, метою випускної кваліфікаційної роботи є моделювання й створення прототипу пристрою для виміру відстані та площі приміщення. Прилад має забезпечувати прийнятні діапазон та точність вимірювання.

## ВИСНОВКИ

В процесі виконання дипломної роботи було спроектовано та створено прототип приладу для вимірювання геометричних розмірів та площі приміщень. Для цього була використана платформа Arduino.

В ході проектування було протестовано декілька датчиків відстані та гіросенсорів, а також велика кількість бібліотек для керування ними. Експерименти показали, що більшість з них не підходять для виконання поставленої задачі. В результаті були застосовані:

1. Arduino Uno в якості керуючої плати. Можливо використання, з невеликими змінами коду, Arduino Mega 2560.
2. Лазерні датчики відстані VL53L0X-V2.
3. Гіросенсор на базі ADXL345 для визначення куту нахилу приладу. Це є необхідним для більш точних вимірювань в деяких режимах.
4. В якості відображаючого пристрою був вибраний LCD Keypad Shield, як більш підходящий для даного приладу і дозволяючи ним керувати за допомогою інтегрованих кнопок.
5. Механічна частина приладу приводиться в дію за допомогою сервомотора SG-90.

З використанням мови програмування Arduino було розроблено програму керування приладом, яка включає 7 режимів вимірювання одноразових і автоматичних безперервних.

Таким чином створений прототип приладу в повній мірі відповідає поставленій задачі і має собівартість значно меншу за промислові зразки. Крім того в приладі використані лазери класу 1, які не спроможні створювати небезпечний для людського ока рівень опромінювання. В промислових зразках використовуються як мінімум лазери класу 2. Але використання таких малопотужних лазерів звужує діапазон вимірювання.

Можливості створеного прототипу при незначній модифікації можуть бути суттєво розширені аж до створення лідару.

## СПИСОК ВИКОРИСТОВУВАНИХ ДЖЕРЕЛ

1. Дж. Блум «Изучаем Arduino. Инструменты и методы технического волшебства». – М.: Эком, 2015. – 336 с.
2. В. Петин «Проекты с использованием контроллера Arduino» – СПб.: БХВ-Петербург, 2014. – 400 с.
3. Т. Иго "Arduino, датчики и сети для связи устройств: Пер. с англ. " – 2-е изд. – СПб.: БХВ-Петербург, 2015. – 544 с.
4. Г. Шонфелдер, К. Шнайдер "Измерительные устройства на базе микропроцессора ATmega" – СПб.: БХВ-СПб, 2012. – 306 с.
5. Аппаратная платформа Arduino [Электронный ресурс].  
URL: <https://www.arduino.ua/>.
6. Sommer У. Программирование микроконтроллерных плат Arduino/Freduino. - СПб.: БХВ-Петербург, 2012. -256 с., ил. - (Электроника).
7. Байда А. С. Использование платформы Arduino при подготовке специалистов автомобильной отрасли // Научно-методический электронный журнал «Концепт». – 2016. – № 5 (май). – 0,4 п. л.
8. Хоровиц П., Хилл У. Искусство схемотехники: Пер. с англ.—Изд. 2-е.
9. — М.: Издательство БИНОМ. — 2016. —704 с., ил.
10. Микроконтроллеры AVR семейств Tiny и Mega фирмы ATMEL, 5-е изд., стер. – М.: Издательский дом «Додэка-XXI», 2008. – 560с.
11. Шонфелдер Г., Шнайдер К. Измерительные устройства на базе микропроцессоров ATmega. - СПб.: БХВ-Петербург, 2012. -288 с., ил. - (Электроника).

## ДОДАТОК А. ПРОГРАМНИЙ КОД ПРОЕКТУ.

```

#define DEBUG 1

//

#include <Liquidcrystal.h> // Підключення бібліотеки
LCD дисплея

Liquidcrystal lcd(8, 9, 4, 5, 6, 7); //Призначення
портів для керування LCD

#include <Servo.h> //використовуємо бібліотеку для
роботи із сервоприводом

Servo myservo; //повідомляємо змінну myservo типу Servo


#include <Wire.h> //Бібліотека для шини I2C

#include <VL53L0X.h> // Библиотека для лазерного
далекоміра

const int adxl345 = 0x53; // I2C адреса ADXL345

//

#define VL53L0X_XSHUT1_PIN    A1 // Ці пини
знадобляться для зміни адреси другого VL53L0X з 29 на
30

#define VL53L0X_XSHUT2_PIN    A2 // Ці пини
знадобляться для зміни адреси другого VL53L0X з 29 на
30

//

#define VL53L0X1_ADDRESS      29 // рядок просто для
інформації, дефолтний адреса VL53L0X.

#define VL53L0X2_ADDRESS      30 // це нова адреса
другого датчика

VL53L0X VL53L0X_1;

VL53L0X VL53L0X_2;

```

```

int x,y,z; //Змінні для зберігання кутів відхилення для
соответствующих координат

int mode=-1; //Номер режиму

int d=50; // Відстань між далекомірами

int L1=0; // Показання 1го далекоміра (у мм)

int L2=0; // Показання 2го далекоміра (у мм)

float L_1=0; //Відстань(м), обмірюване 2мя далекомірами
- 1я координата. З обліком їх взаємного розташування
(+d)

float L_2=0; //Відстань (м), обмірюване 2мя
далекомірами - 2я координата. З обліком їх взаємного
розташування (+d)

float S=0; //Площа приміщення (кв.м.)

#define LONG_RANGE // Вибір режиму виміру максимального
растояния

//#define HIGH_SPEED // Вибір режиму максимальної
швидкості вимірів

//#define HIGH_ACCURACY // Вибір режиму максимальної
точності вимірів

void setup()
{
//Початкові установки дисплея
lcd.begin(16, 2);
lcd.clear(); //Очищення екрана
analogwrite(10, 255); //Регулювання яскравості екрана
lcd.setCursor(0, 0);

    lcd.print("Select MODE");//Початкове меню вибору
режиму вимірів

    lcd.setCursor(0, 1);

    lcd.print("Press UP/DOWN");//Підказка по кнопках

```



```

myservo.attach(A3); //прив'язуємо привод до порту A3

// Зміна адреси лазерного далекоміра
// На старті відразу "гасимо" пины XSHUT обох датчиків
pinmode(VL53L0X_XSHUT1_PIN, OUTPUT);
pinmode(VL53L0X_XSHUT2_PIN, OUTPUT);
//
#ifdef DEBUG
    Serial.begin(9600);
    Serial.println("");
    Serial.println(F("Two VL53L0X."));
#endif
Wire.begin();

// Міняємо адресу другого датчика й "піднімаємо" обое
датчика
pinmode(VL53L0X_XSHUT2_PIN, INPUT);
delay(10);
VL53L0X_2.setaddress(VL53L0X2_ADDRESS);
pinmode(VL53L0X_XSHUT1_PIN, INPUT);
delay(10);
// Запускаємо обое VL53L0X
VL53L0X_1.init();
VL53L0X_2.init();
VL53L0X_1.setTimeout(250);
VL53L0X_2.setTimeout(250);
// Щоб використовувати безперервний синхронізований
режим, укажіть бажаний період виміру в мс.
//VL53L0X_1.startcontinuous(50);

```

```

//VL53L0X_2.startcontinuous(50);

#if defined LONG_RANGE

    // нижче межі швидкості зворотного сигналу ( за
    замовчуванням 0,25 MCPS)

    VL53L0X_1.setsignalratelimit(0.1);

    VL53L0X_2.setsignalratelimit(0.1);

    // побільшати періоди лазерного імпульсу ( за
    замовчуванням 14 і 10 PCLK)

VL53L0X_1.setvcselfpulseperiod(VL53L0X::Vcselfperiodprera
nge, 18);

VL53L0X_1.setvcselfpulseperiod(VL53L0X::Vcselfperiodfinal
range, 14);

VL53L0X_2.setvcselfpulseperiod(VL53L0X::Vcselfperiodprera
nge, 18);

VL53L0X_2.setvcselfpulseperiod(VL53L0X::Vcselfperiodfinal
range, 14);

#endif

#if defined HIGH_SPEED

    // зменшити часовий бюджет до 20 мс ( за
    замовчуванням близько 33 мс)

    VL53L0X_1.setmeasurementtimingbudget(20000);

    VL53L0X_2.setmeasurementtimingbudget(20000);

#elif defined HIGH_ACCURACY

    // побільшати тимчасовий бюджет до 200 мс

    VL53L0X_1.setmeasurementtimingbudget(200000);

    VL53L0X_2.setmeasurementtimingbudget(200000);

#endif

```

```

// запишемо адресу регістру DEVID
Wire.begintransmission(adxl345);
Wire.write(byte(0x00));
Wire.endtransmission();

// прочитаємо регістр DEVID:
Wire.requestfrom(adxl345, 1);
while (Wire.available()) {
    byte c = Wire.read();
    Serial.print("ID = ");
    Serial.println(c, HEX);
}

// переведемо акселерометр у режим вимірів
Wire.begintransmission(adxl345);
Wire.write(byte(0x2D));
Wire.write(byte(0x08));
Wire.endtransmission();
}

byte key() { //визначення натиснутої кнопки
    //1-640 SELECT, 2-411 LEFT, 3-101 UP, 4-256 DOWN, 5-0
    RIGHT soprotivlenie knopok
    int val = analogread(0);
    //Zadaem nomer knopki
    if (val < 50) return 5;
    else if (val < 150) return 3;
}

```

```

else if (val < 350) return 4;
else if (val < 500) return 2;
else if (val < 800) return 1;
else if (val <= 1023) return 0;
}

```

```

int gyro(){ //зчитування й обчислюємо показання
гіросенсора

    // запишемо адресу початку даних по осях X, Y і Z:
Wire.begintransmission(adxl345);
Wire.write(byte(0x32));
Wire.endtransmission();

    // прочитаємо 6 байтів значень XYZ:
int i = 0;
byte xyz[6];
Wire.requestfrom(adxl345, 6);
while (Wire.available()) {
    byte c = Wire.read();
    xyz[i] = c;
    i++;
}

    // порахуємо й відобразимо значення X, Y, Z:
x = word(xyz[1], xyz[0]);
y = word(xyz[3], xyz[2]);
z = word(xyz[5], xyz[4]);

```

```

    return x,y; // надалі нам понадобятся дані тільки по
    двом осям
}

```

```

int dist(){ // зчитуємо показання в мм із далекомірів
    L1=VL53L0X_1.readrangesinglemillimeters();
    L2=VL53L0X_2.readrangesinglemillimeters();
    return L1, L2;
}

```

```

void loop()
{
    //вибір режиму виміру кнопкою UP (код 3)

    if(key()==3){mode++;
    if(mode==7)mode=0;lcd.clear();lcd.print("Mode");lcd.pri
    nt(mode);Serial.print("Mode");Serial.print(mode);}

    delay(150);

    //вибір режиму виміру кнопкою DOWN (код 4)

    if(key()==4){mode--;
    if(mode<0)mode=6;lcd.clear();lcd.print("Mode");lcd.prin
    t(mode);Serial.print("Mode");Serial.print(mode);}

    delay(150);

    if(key()==1&&mode>-1){ //Підтвердження обраного
    режиму кнопкою Select (код 1)

    switch (mode) {

        case 0: // mode0 Виміряти відстань двома датчиками
        в циклі

                // Зупинити - кнопка Ritgh (код 5)

                do {

                    gyro();

```

```

    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("X=");
    lcd.print(x);
    lcd.setCursor(8,0);
    lcd.print("Y=");
    lcd.print(y);
    lcd.setCursor(0,1);
    dist();
    lcd.print("L=L1+L2=");
    lcd.print(L1+L2+d);
    delay(300);
}

while(key()!=5);

break;

case 1: // model Виміряти відстань двома датчиками
однократно
        //Повторний вимір - Select, вибір іншого
режиму - UP/DOWN
    gyro();
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("X=");
    lcd.print(x);
    lcd.setCursor(8,0);
    lcd.print("Y=");
    lcd.print(y);
    lcd.setCursor(0,1);
    dist();

```

```

        lcd.print("L=L1+L2=");

        lcd.print(L1+L2+d);

        break;

    case 2: // mode2 Вимір площі приміщення
ОДНОКРАТНО!!!

        //Повторний вимір - Select, вибір іншого
режиму - UP/DOWN

        myservo.write(10); //ставимо вал під 10 градусів
        delay(200); //чекаємо 0,2 секунди


        gyro();

        lcd.clear();

        lcd.setCursor(0,0);

        lcd.print("X=");

        lcd.print(x);

        lcd.setCursor(8,0);

        lcd.print("Y=");

        lcd.print(y);


        if(abs(x)<10&&abs(y)<10)

        {dist();

            L_1=(L1+L2+d)/1000.0; //одержуємо одна відстань
("ширину") у метрах

        }

        else {L_1=0.0; // якщо не виставлений рівень
(нахил більше 10 градусів) - обнулить результат
вимірів і повідомити про помилку

            lcd.setCursor(0,1);

            lcd.print("H_ERR_1");

        }

```

```

    delay(1000);

    myservo.write(100); //ставимо вал під 100
градусів, тобто повертаємо датчики на 90 градусів

    delay(200); //чекаємо 0,2 секунди


    gyro();

    lcd.clear();

    lcd.setCursor(0,0);
    lcd.print("X=");
    lcd.print(x);
    lcd.setCursor(8,0);
    lcd.print("Y=");
    lcd.print(y);


    if (abs(x)<10&&abs(y)<10)
    {dist();

        L_2=(L1+L2+d)/1000.0; //одержуємо інша відстань
("довжину") у метрах

    }

    else {L_2=0.0; // якщо не виставлений рівень
(нахил більше 10 градусів) - обнулить результат
вимірів і повідомити про помилку

        lcd.setCursor(0,1);
        lcd.print("H_ERR_2");
    }


    S=L_1*L_2; // Розрахунки площі приміщення у кв.м.

    lcd.setCursor(0,1);
    lcd.print("S=");
    lcd.print(S);

```



```

    lcd.print(" m2");

    // Контрольний вивід у монітор порту для
налагодження

    Serial.print("L_1=");
    Serial.print(L_1);
    Serial.println(" m");

    Serial.print("L_2=");
    Serial.print(L_2);
    Serial.println(" m");
    Serial.print("S=");
    Serial.print(S);
    Serial.println("m2");
    break;

case 3: // mode3 Вимір одним датчиком (L1) у циклі
        // Зупинити - кнопка Ritgh (код 5)
    do {
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("Distance 1 (mm)");
        lcd.setCursor(0,1);
        dist();
        lcd.print(L1);
        delay(300);
    } while(key()!=5);
    break;

```

```

    case 4: // mode4 Вимір одним датчиком (L1)
однократно

        //Повторний вимір - Select, вибір іншого
режиму - UP/DOWN

        lcd.clear();

        lcd.setCursor(0,0);

        lcd.print("Distance 1 (mm)");

        lcd.setCursor(0,1);

        dist();

        lcd.print(L1);

        break;

    case 5: //mode5; Вимір другим датчиком (L2) у циклі
        // Зупинити - кнопка Ritgh (код 5)

        do{

            lcd.clear();

            lcd.setCursor(0,0);

            lcd.print("Distance 2 (mm)");

            lcd.setCursor(0,1);

            dist();

            lcd.print(L2);

            delay(300);

        }while(key()!=5);

        break;

    case 6: //mode6; //Вимір другим датчиком (L2)
однократно

        //Повторний вимір - Select, вибір іншого
режиму - UP/DOWN

        lcd.clear();

        lcd.setCursor(0,0);

        lcd.print("Distance 2 (mm)");

```

```
    lcd.setCursor(0,1);  
    dist();  
    lcd.print(L2);  
    break;  
}  
delay(250); // Затримка 0,25 сек  
}  
}
```