

ДОДАТОК А

Апробація результатів наукових досліджень



Primedia
eLaunch



PROCEEDINGS OF THE
III INTERNATIONAL SCIENTIFIC
AND THEORETICAL CONFERENCE

CURRENT SCIENTIFIC
GOALS, APPROACHES
AND CHALLENGES

17.01.2025

RIGA
REPUBLIC OF LATVIA

 **SCIENTIA**
COLLECTION OF SCIENTIFIC PAPERS

SECTION 11. GENERAL MECHANICS AND MECHANICAL ENGINEERING

STOCHASTIC MATHEMATICAL MODEL GROUND
Gorbatyuk Ie.V., Mishchuk D.O., Komotska S.Ju.125

SECTION 12. AUTOMATION AND APPLIANCES MAKING

АВТОМАТИЗОВАНА СИСТЕМА УПРАВЛІННЯ РОБОТИЗОВАНИМ
МАНІПУЛЯТОРОМ З ВИКОРИСТАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ
Ахмад Фаді Халед127

ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ МЕТОДУ ДИСТАНЦІЙНОГО КЕРУВАННЯ
РОБОТИЗОВАНИМ БЕЗПІЛОТНИМ ЗАСОБОМ
Ахмад Даніель Халед131

РОЗРОБЛЕННЯ ХМАРНОГО СЕРЕДОВИЩА ДЛЯ ДИСТАНЦІЙНОГО
НАВЧАННЯ СТУДЕНТІВ
Ткачов М.Р.134

SECTION 13. ELECTRONICS AND TELECOMMUNICATIONS

КЛЮЧОВІ АСПЕКТИ ВИБОРУ ІНСТРУМЕНТІВ УПРАВЛІННЯ ПРОЄКТАМИ:
TRELLO ТА JIRA
Пацук В.О.137

SECTION 14. ENERGY AND POWER ENGINEERING


ОЦІНКА СТРУМІВ РАПТОВОГО КОРОТКОГО ЗАМИКАННЯ У СИНХРОННОМУ
ГЕНЕРАТОРІ
Пантелєєва І.В., Пономаренко О.М.140

SECTION 15. INFORMATION TECHNOLOGIES AND SYSTEMS

ДОСЛІДЖЕННЯ ПІДХОДІВ ДО ПІДВИЩЕННЯ НАДІЙНОСТІ ТА
ВІДМОВОСТІЙКОСТІ СКЛАДНИХ КОМП'ЮТЕРНИХ МЕРЕЖ У СИСТЕМАХ
ДИСТАНЦІЙНОГО НАВЧАННЯ НА ЗАСАДАХ ЗАСТОСУВАННЯ ЕКСПЕРТНИХ
СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ
Веселовська Г.В., Плахотнюк М.Ф., Радзівілов О.І.147

Ахмад Даніель Халєд

здобувач вищої освіти факультету Автоматики і комп'ютеризованих технологій
Харківський національний університет радіоелектроніки, Україна

Науковий керівник: Новоселов Сергій Павлович 

канд. техн. наук, доцент,
професор кафедри комп'ютерно-інтегрованих технологій автоматизації і робототехніки
Харківський національний університет радіоелектроніки, Україна

ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ МЕТОДУ ДИСТАНЦІЙНОГО КЕРУВАННЯ РОБОТИЗОВАНИМ БЕЗПЛОТНИМ ЗАСОБОМ

У сучасному світі мобільні роботи стають невід'ємною частиною автоматизованих систем, які широко застосовуються в різних галузях – від промислових підприємств до військової сфери та сільського господарства. Однак для ефективного управління мобільними роботами, особливо в складних та динамічних середовищах, необхідна підготовка операторів та інженерів, що включає не лише теоретичні знання, але й практичні навички. У цьому контексті програмні тренажери для моделювання процесу дистанційного керування мобільними роботами стають актуальним та важливим інструментом.

Програмні тренажери для дистанційного керування мобільними роботами дають можливість моделювати реальні ситуації без необхідності використання фізичних пристроїв.

Програмні тренажери та цифрові двійники мають спільну мету – підвищення ефективності роботи та оптимізація процесів керування [1, 2]. Однак вони застосовуються в різних контекстах: тренажери використовуються для навчання й тестування в симульованих умовах, тоді як цифрові двійники інтегруються з реальними системами для управління, прогнозування та аналізу. Обидві технології мають величезний потенціал для розвитку робототехніки, промислової автоматизації та інших галузей, де важлива точність, безпека та ефективність процесів.

Архітектура автоматизованої системи дистанційного керування мобільним роботом показана на (рис. 1).

До складу автоматизованої системи входять: пульт дистанційного керування, радіоприймач, контролер Arduino, модуль перетворення інтерфейсів з боку приймача, модуль перетворення інтерфейсів з боку персонального комп'ютера, персональний комп'ютер.

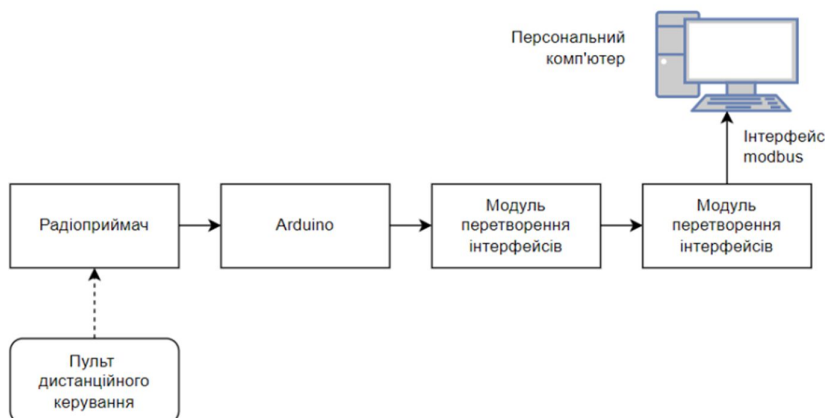


Рис. 1. Архітектура автоматизованої системи дистанційного керування мобільним роботом

Як видно з (рис. 1) автоматизована система для дистанційного керування мобільним роботом складається з кількох основних компонентів, кожен з яких виконує свою специфічну функцію для забезпечення передачі та обробки команд.

Пульт дистанційного керування є основним засобом для передачі команд оператором мобільному роботу. Flysky FS-i4X працює на радіочастотному зв'язку 2,4 ГГц і дозволяє оператору керувати роботом шляхом зміни положення джойстиків та інших елементів управління. Пульт генерує ШІМ (широко-імпульсну модуляцію) сигнали, які передаються на приймач.

Приймач отримує радіосигнал від пульта і перетворює його на ШІМ-сигнали, які передаються на контролер. Цей компонент є посередником між пультом дистанційного керування та іншими частинами системи.

Контролер Arduino використовується для зчитування та обробки ШІМ-сигналів з радіоприймача. Контролер зчитує тривалість імпульсів кожного каналу і перетворює ці дані в числові значення, які можуть бути передані через інтерфейси зв'язку.

Модуль, встановлений на персональному комп'ютері, приймає пакети даних у форматі протоколу Modbus, перетворюючи їх у зрозумілі для програми-тренажера команди.

ПК є основним елементом для моделювання та обробки даних. Програма-тренажер, що працює на ПК, отримує дані від Arduino через протокол Modbus, аналізує ці дані та візуалізує їх у вигляді симуляції.

Висновки. Система, що описана у даній роботі, є ефективним засобом для тренування операторів мобільних роботів у режимі симуляції. Вона поєднує апаратні та програмні компоненти, такі як пульт дистанційного керування, контролер Arduino та програму-тренажер, що дозволяє повністю моделювати процес дистанційного керування мобільним роботом. Така система дозволяє знизити витрати на реальні тестування та підвищити якість навчання операторів у безпечному середовищі.

Список використаних джерел:

1. Невлюдов І. Ш. Застосування цифрових двійників технічних засобів автоматизації для розроблення програмно-технічних комплексів АСУ ТП : Навчальний посібник / І. Ш. Невлюдов, С. П. Новоселов, О. В. Сичова. – Харків: Видавництво Іванченка І. С., 2023. – 267 с.
2. J. Ren, L. Nalpantidis, N. A. Andersen and O. Ravn, "Building Digital Twin of Mobile Robotics Testbed Using Centralized Localization System," 2023 11th International Conference on Control, Mechatronics and Automation (ICCSMA), Grimstad, Norway, 2023, pp. 139-145, doi: 10.1109/ICCSMA59762.2023.10375050.

ДОДАТОК Б
Вихідний код програми

```
#include <Arduino.h>

// Module connection pins (Digital Pins)
#define CH1 10
#define CH2 9
#define CH3 8
#define CH4 7
#define CH5 6
#define CH6 5

#define TXEN 2
#define test 3
#define ID 1

unsigned long del = 90000;
unsigned long count = 0;

unsigned long start_CH1;
unsigned long stop_CH1;
int pwm_CH1;

unsigned long start_CH2;
unsigned long stop_CH2;
int pwm_CH2;

unsigned long start_CH3;
unsigned long stop_CH3;
int pwm_CH3;

unsigned long start_CH4;
```

```
unsigned long stop_CH4;  
int pwm_CH4;
```

```
unsigned long start_CH5;  
unsigned long stop_CH5;  
int pwm_CH5;
```

```
unsigned long start_CH6;  
unsigned long stop_CH6;  
int pwm_CH6;
```

```
bool is_CH1 = false;  
bool is_CH2 = false;  
bool is_CH3 = false;  
bool is_CH4 = false;  
bool is_CH5 = false;  
bool is_CH6 = false;
```

```
//uint8_t data[] = { 0xff, 0xff, 0xff, 0xff };  
uint16_t au16data[16] = {3, 4, 5, 6, 0, 0, 0, 7, 8, 9, 0, 0, 0, 0, 1, 1 };  
//au16data[0] -> CH1  
//au16data[1] -> CH2  
//au16data[2] -> CH3  
//au16data[3] -> CH4  
//au16data[4] -> CH5  
//au16data[5] -> CH6
```

```
Modbus slave(ID, Serial, TXEN); // this is slave @1 and RS-232 or USB-  
FTDI
```

```
void checkPWM()
{
  //Channel 1
  if (digitalRead(CH1) == HIGH){
    if (is_CH1 == false)
    {
      start_CH1 = millis();
      is_CH1 = true;
    }
  }
  else if (is_CH1 == true){
    stop_CH1 = millis();
    pwm_CH1 = stop_CH1 - start_CH1;
    au16data[0] = pwm_CH1;
    is_CH1 = false;
  }

  //Channel 2
  if (is_CH2 == true)
  {
    if (digitalRead(CH2) == LOW) {
      stop_CH2 = millis();
      pwm_CH2 = stop_CH2 - start_CH2;
      au16data[1] = pwm_CH2;
      is_CH2 = false;
    }
  }
  else {
    if (digitalRead(CH2) == HIGH){
      start_CH2 = millis();
```

```
    is_CH2 = true;
  }
}

//Channel 3
if (is_CH3 == true)
{
  if (digitalRead(CH3) == LOW) {
    stop_CH3 = millis();
    pwm_CH3 = stop_CH3 - start_CH3;
    au16data[2] = pwm_CH3;
    is_CH3 = false;
  }
}
else {
  if (digitalRead(CH3) == HIGH){
    start_CH3 = millis();
    is_CH3 = true;
  }
}

//Channel 4
if (is_CH4 == true)
{
  if (digitalRead(CH4) == LOW) {
    stop_CH4 = millis();
    pwm_CH4 = stop_CH4 - start_CH4;
    au16data[3] = pwm_CH4;
    is_CH4 = false;
  }
}
```

```
}  
else {  
    if (digitalRead(CH4) == HIGH){  
        start_CH4 = millis();  
        is_CH4 = true;  
    }  
}  
  
//Channel 5  
if (is_CH5 == true)  
{  
    if (digitalRead(CH5) == LOW) {  
        stop_CH5 = millis();  
        pwm_CH5 = stop_CH5 - start_CH5;  
        au16data[4] = pwm_CH5;  
        is_CH5 = false;  
    }  
}  
else {  
    if (digitalRead(CH5) == HIGH){  
        start_CH5 = millis();  
        is_CH5 = true;  
    }  
}  
  
//Channel 6  
if (is_CH6 == true)  
{  
    if (digitalRead(CH6) == LOW) {  
        stop_CH6 = millis();
```

```
    pwm_CH6 = stop_CH6 - start_CH6;
    au16data[5] = pwm_CH6;
    is_CH6 = false;
  }
}
else {
  if (digitalRead(CH6) == HIGH){
    start_CH6 = millis();
    is_CH6 = true;
  }
}
}
```

```
void setup()
```

```
{
  Serial.begin(115200);
  slave.start();
```

```
  pinMode(CH1, INPUT);
  pinMode(CH2, INPUT);
  pinMode(CH3, INPUT);
  pinMode(CH4, INPUT);
  pinMode(CH5, INPUT);
  pinMode(CH6, INPUT);
```

```
  pinMode(TXEN, OUTPUT);
  pinMode(test, OUTPUT);
```

```
  checkPWM();
```

```
    digitalWrite(test, HIGH);
}

void work_modbus()
{
    au16data[0] = pwm_CH1;
    au16data[1] = pwm_CH2;
    au16data[2] = pwm_CH3;
    au16data[3] = pwm_CH4;
    au16data[4] = pwm_CH5;
    au16data[5] = pwm_CH6;

}

void loop()
{

    pwm_CH1 = pulseIn(CH1, HIGH);
    au16data[0] = pwm_CH1;

    pwm_CH2 = pulseIn(CH2, HIGH);
    au16data[1] = pwm_CH2;

    pwm_CH3 = pulseIn(CH3, HIGH);
    au16data[2] = pwm_CH3;

    pwm_CH4 = pulseIn(CH4, HIGH);
    au16data[3] = pwm_CH4;

    pwm_CH5 = pulseIn(CH5, HIGH);
```

```
au16data[4] = pwm_CH5;

pwm_CH6 = pulseIn(CH6, HIGH);
au16data[5] = pwm_CH6;

slave.poll( au16data, 16 );
pwm_CH1 = 0;
pwm_CH2 = 0;
pwm_CH3 = 0;
pwm_CH4 = 0;
pwm_CH5 = 0;
pwm_CH6 = 0;

}
```

ДОДАТОК В

Демонстраційний матеріал

