

**Студентська наукова робота:**

**«ЗАСТОСУВАННЯ УЛЬТРАЗВУКОВИХ ДАТЧИКІВ ДЛЯ УПРАВЛІННЯ  
НАПРЯМКОМ РУХУ МОБІЛЬНОГО РОБОТУ»**

**шифр роботи «SmartCar»**

## АНОТАЦІЯ

**Застосування ультразвукових датчиків для управління напрямком руху мобільного робота. Шифр роботи «SmartCar».** Актуальність роботи обумовлена стрімким розвитком сучасної мікропроцесорної техніки, що забезпечує сприятливі умови для розвитку величезної кількості областей, у тому числі створення мобільних роботів, основним завданням яких є орієнтування в просторі.

Метою роботи є розробка мобільного робота з автономним управлінням напрямком руху, завдяки застосуванню ультразвукових датчиків.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- проаналізувати сучасний стан останніх досліджень і публікацій з теми роботи;
- розробити мобільного робота;
- удосконалити локалізацію мобільного робота;
- розробити та впровадити алгоритм управління напрямком руху мобільного робота.

Наукова новизна та цінність отриманих результатів досліджень полягає у тому, що розроблений алгоритм управління напрямком руху дозволяє сприймати навколишнє середовище й орієнтуватися в ньому, тим самим забезпечує змістовну поведінку і виконання функціональної роботи мобільного робота за допомогою своїх "інтелектуальних" можливостей; для підвищення якості локалізації та виявлення перешкод було використано фільтр Калмана; а також встановлено залежність точності локалізації від температури повітря в приміщенні, де відбувається тестування мобільного робота.

Використана методика дослідження базується на методах теорії прогнозування та навігації, теорії ймовірностей і математичної статистики.

**Ключові слова:** мобільний робот, алгоритм, ультразвукові датчики, мікроконтролер, управління, напрямок руху, фільтр Калмана, локалізація, перешкода.

## ЗМІСТ

ВСТУП .....	3
1.РОЗРОБКА МОБІЛЬНОГО РОБОТА .....	5
1.1 Аналіз останніх досліджень і публікацій .....	5
1.2 Компоненти мобільного робота .....	6
1.3 Принцип дії ультразвукових датчиків відстані .....	9
1.4 Залежність точності локалізації мобільного робота від температури повітря в приміщенні.....	10
1.5 Широтно-імпульсна модуляція .....	13
1.6 Мікроконтролер STM32: реалізація переривань і таймерів.....	15
2.ВИКОРИСТАННЯ ФІЛЬТРУ КАЛМАНА ДЛЯ ЛОКАЛІЗАЦІЇ МОБІЛЬНОГО РОБОТА .....	18
2.1 Локалізація мобільного робота .....	18
2.2 Фільтр Калмана.....	20
2.3 Результати тестування.....	22
3.РОЗРОБКА АЛГОРИТМУ УПРАВЛІННЯ РУХОМ МОБІЛЬНОГО РОБОТА .....	25
3.1 Алгоритм управління рухом, ґрунтуючись на розташуванні мобільного робота.....	25
3.2 Результати використання розробленого алгоритму.....	28
ВИСНОВКИ.....	30
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ .....	31

## ВСТУП

**Актуальність роботи** обумовлена стрімким розвитком сучасної мікропроцесорної техніки, що забезпечує сприятливі умови для розвитку величезної кількості областей: медицина - глюкометри, інсулінові помпи, монітори тиску / пульсу / крові, вимірювачі холестерину; лічильники газу, тиску, води, температури; вимірювальні прилади - мультиметри, RLC-метри; сигналізація - центральні модулі, датчики задимлення / руху / відкриття, електронні клапани водопостачання, електронні замки, радіобрелоки сигналізації; промислова автоматизація - провідні та безпроводні датчики, приводи, драйвери і контролери виконавчих механізмів; радіокеровані моделі та бездротові телефони, джойстики та маніпулятори, годинники та дзвінки, іграшки та інші. Наряду з цим, не менш важливим прикладом мікропроцесорної техніки можна вважати розробку мобільних роботів, основним завданням яких є орієнтування в просторі. Мобільний робот - автоматична машина, яка передбачає наявність рухомого шасі з керованими приводами. Найбільш поширеними мобільними роботами є чотириколісні роботи. Завдання управління рухом мобільних колісних роботів крім теоретичного набувають останнім часом все більше практичне значення в різних областях техніки. Спектр вирішуваних завдань при цьому виявляється дуже широким: від ігрових завдань до спеціальних завдань забезпечення безпеки.

**Мета роботи** - розробка мобільного робота з автономним управлінням напрямком руху, завдяки застосуванню ультразвукових датчиків.

Для досягнення поставленої мети необхідно вирішити наступні **завдання**:

- проаналізувати сучасний стан останніх досліджень і публікацій з теми роботи;
- розробити мобільний робот;
- удосконалити локалізацію мобільного робота;
- розробити та впровадити алгоритм управління напрямком руху мобільного робота.

**Наукова новизна** отриманих результатів досліджень полягає у наступному: розроблено алгоритм управління напрямком руху, що базується на розташуванні мобільного робота, який здатний автоматизувати рух мобільного робота; для підвищення якості локалізації та виявлення перешкод було використано фільтр Калмана; а також встановлено залежність точності локалізації від температури повітря в приміщенні, де відбувається тестування мобільного робота.

**Цінність отриманих результатів** полягає в тому, що розроблений алгоритм управління напрямком руху дозволяє сприймати навколишнє середовище й орієнтуватися в ньому, тим самим забезпечує змістовну поведінку і виконання функціональної роботи мобільного робота за допомогою своїх "інтелектуальних" можливостей, а це є головним кроком до створення повністю автономних та багатофункціональних роботів.

**Використана методика дослідження.** Проведені в роботі дослідження базуються на методах теорії прогнозування та статистичного аналізу, теорії управління та навігації систем, теорії ймовірностей і математичної статистики.

## 1. РОЗРОБКА МОБІЛЬНОГО РОБОТА

### 1.1 Аналіз останніх досліджень і публікацій

В останні роки великий інтерес для дослідників представляють автономні мобільні роботи, а саме задача управління рухом мобільних колісних роботів.

Автори роботи [1] зосереджені на внутрішній системі локалізації робота на основі ультразвуку. Запропоноване рішення - це вимірювання положення роботів, обчислених за допомогою формул триангуляції.

Алгоритм, розроблений в роботі [2], дозволяє визначати положення роботів у кожний момент часу, використовуючи положення орієнтирів. Алгоритм враховує асинхронні кроки часу та розрізнені дані вимірювань для розробки його оцінок.

Система оцінки координат автоматизованих керованих транспортних засобів обговорюється в роботі [3]. Для вимірювання відстані застосовували ультразвукові хвилі. Система була протестована на колісному мобільному роботі в середовищі моделі 2D.

Автори роботи [4] запровадили новий підхід до навігації наземних транспортних засобів за допомогою дуже дешевого ультразвукового датчику та розширеного фільтру Калмана, що зменшило середню квадратичну помилку.




Існує багато інших цікавих алгоритмів уникнення перешкод. Деякі з них підходять для роботи у режимі реального часу, серед них підходи, засновані на нечіткій логіці [5], зоровій навігації [6-7] тощо.

Однак більшість алгоритмів навігації роботів покладаються на важкі та енергоємні датчики. Як наслідок, цим роботам потрібні потужні обчислювальні одиниці для встановлення на борту. Тому під час розробки мобільного робота використані ультразвукові датчики відстані безконтактного типу та мікроконтролер STM32, який на відміну від Arduino вперше застосовано для такого роду розробок, що забезпечило вищу продуктивність ніж використання скетчів для програмування мікроконтролерів, також має перевагу в співвідношенні ціна/якість.

## 1.2 Компоненти мобільного робота

Загалом при розробці мобільного робота були використані наступні компоненти:

Таблиця 1.1 - Компоненти мобільного роботу

№ п/п	Назва компоненту	Зовнішній вигляд компоненту
1.	32-бітовий мікроконтролер виробництва STMicroelectronics	
2.	Ультразвукові датчики відстані безконтактного типу HC-SR04	
3.	Драйвери двигунів L9110S	
4.	Макетна плата та двигуни	

### 1. 32-бітовий мікроконтролер виробництва STMicroelectronics.

Компанія STMicroelectronics є найбільшим виробником мікроконтролерів в світі, при цьому більша частина припадає на сімейства STM32. Популярність платформи STM32 багато в чому визначається величезним вибором моделей. На даний момент в номенклатурі STM32 налічується більше семи сотень представників, і кожен інженер зможе підібрати оптимальний мікроконтролер практично для кожної програми, виходячи з вимог, що пред'являються до продуктивності / рівню інтеграції / споживання / вартості.

### 2. Ультразвуковий датчик відстані безконтактного типу HC-SR04.

На відміну від інфрачервоних далекомірів на ультразвуковий датчик HC-SR04 не впливають джерела світла або колір перешкоди. Датчик відстані є приладом безконтактного типу, і забезпечує високоточне вимірювання і стабільність. Діапазон дальності його вимірювання складає від 0 до 150 см, точність досягає 3 мм.

### 3. *Драйвер двигуна L9110S.*

Модуль управління моторами на мікросхемі L9110S використовується для контролю / обертання двох незалежних моторів або одного 4-х дротового 2-х фазного крокового двигуна. Драйвер має два інтерфейси для підключення електроживлення, мікроконтролера і керованих пристроїв.

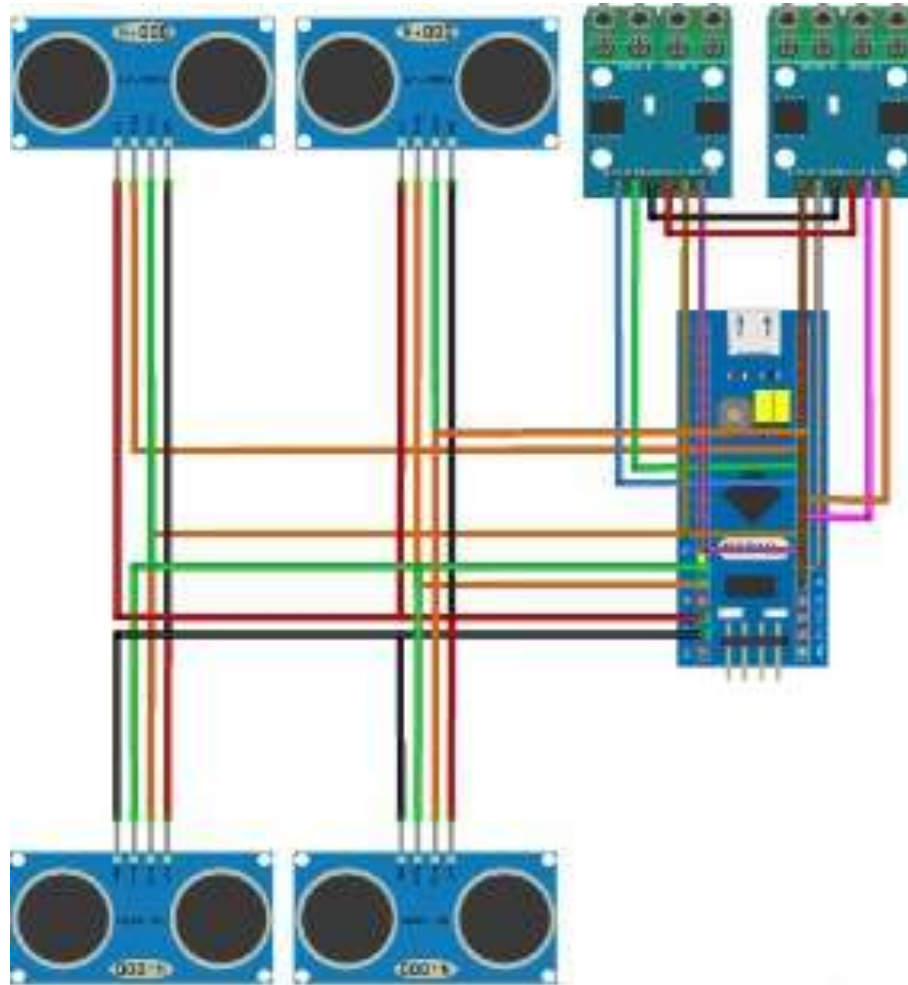


Рисунок 1.1 - Схема підключення компонентів мобільного робота

В роботі використано середовище розробки Keil  $\mu$ Vision [8], що представляє собою набір утиліт для виконання повного комплексу заходів з написання програмного забезпечення для мікроконтролерів на мові програмування C++. Дозволяє працювати з проектами будь-якого ступеня складності, починаючи з введення і редагування вихідних текстів і закінчуючи внутрішньосхемним налагодженням коду і програмуванням ПЗП мікроконтролера.



Для зручності роботи з мікроконтролером була використана бібліотека `stm32f10x.h` [9], а саме застосовувалася для:

#### 1. Портів вводу/виводу.

Побітова настройка:

```
GPIOC->CRH &=~ (1<<26);
GPIOC->CRH |= (1<<27);
GPIOC->ODR &=~ (1<<14);
```

За допомогою засобів бібліотеки:

```
GPIOB->CRL &=~ GPIO_CRL_CNF1;
GPIOB->CRL |= GPIO_CRL_MODE1;
```

#### 2. Використання змінних типу `uint`.

```
uint32_t dist_front;
uint32_t dist_back;
uint32_t dist_left;
uint32_t dist_right;
```

#### 3. Читання вхідного сигналу.

```
GPIOC->IDR & 1 << 14 (функція дає 0 або 1)
```

Було використано для реалізації перемикання між двома реалізованими режимами роботи мобільного робота:

- вибір безпечного місця, орієнтуючись на показники датчиків, відповідно до зазначених даних в програмі;
- автономне пересування, з пріоритетним переміщенням вперед, уникаючи зустрічних перешкод, напрямком повороту визначається шляхом порівняння показників датчиків.

#### 4. Зручного управління станом порту.

```
#define drv_1_0 GPIOA->BRR |= 1<<4
#define drv_1_1 GPIOA->BSRR |= 1<<4
#define drv_2_0 GPIOA->BRR |= 1<<5
#define drv_2_1 GPIOA->BSRR |= 1<<5
```

5. Роботи з таймерами всіх призначень, а так само для реалізації затримки в роботі програми.

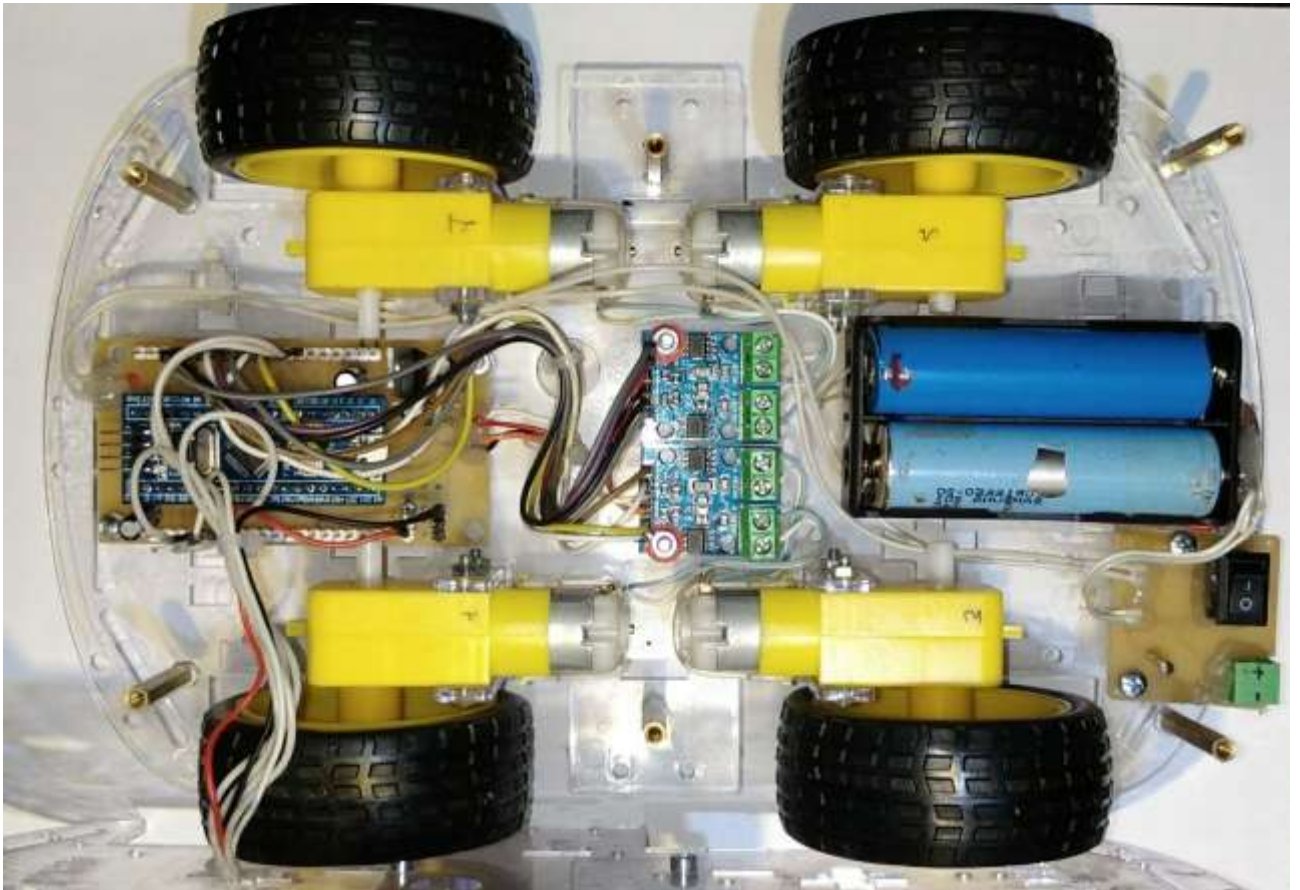


Рисунок 1.2 – Вигляд мобільного робота

В цілому, мобільний робот може рухатися вперед-назад і повертати; реалізовано режим автономного управління рухом, з можливістю уникання перешкод; корегує швидкість руху в залежності від відстані до перешкоди; також є простим для збирання і розбирання (для подальших модифікацій).

### **1.3 Принцип дії ультразвукових датчиків відстані**

За допомогою ультразвукових хвиль датчики руху виконують сканування навколишнього середовища на наявність перешкод при переміщенні мобільного робота.

Принцип дії таких датчиків полягає в наступному: на платі модуля є п'єзо-ультразвуковий передавач та мікрофон, який відчуває високочастотну хвилю (25-40 кГц), яка відбивається перешкодою. Такі звукові хвилі не сприймаються людським вухом. Датчик нічого не обчислює самостійно, а лише

дає імпульс певної тривалості. За допомогою мікроконтролера відбувається злагоджена робота приймача і передавача ультразвуку, обчислення відстані до об'єкту, розрахунок зміни параметрів ехо-імпульсу (рис. 1.3).

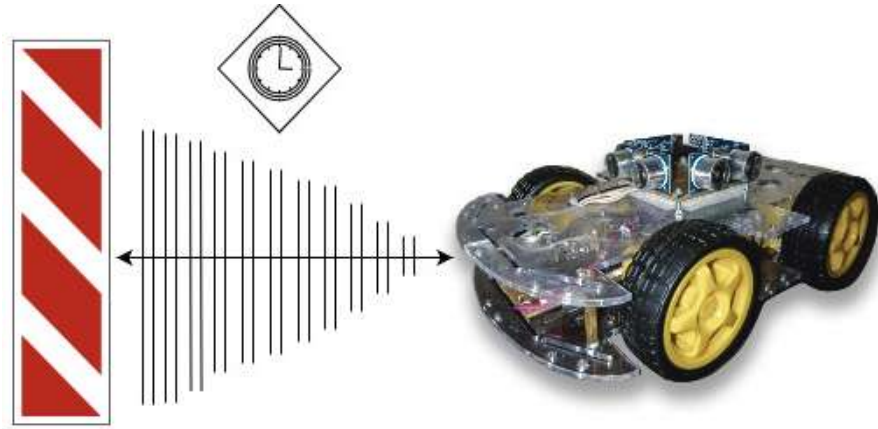


Рисунок 1.3 – Виявлення перешкоди на шляху мобільного робота

В основі роботи ультразвукових датчиків руху знаходиться ефект Доплера. Відповідно до ефекту Доплера, будь-який предмет, що перетинає ультразвуковий потік, впливає на інтерференційну картину хвиль. Ефект Доплера полягає в тому, що випускаються джерелом звукові хвилі з певною частотою, сприймаються приймачем на частоті, відмінній від частоти, коли джерело і (або) приймач рухаються відносно один одного. Тобто, доплерівська частота  $f_d$  дорівнює:

$$f_d = \frac{-V_{rs}}{c} f_s, \quad (1.1)$$

де  $V_{rs}$  – швидкість приймача відносно джерела,  $f_s$  – частота джерела,  $c$  – швидкість поширення хвилі, яку випромінює джерело.

#### 1.4 Залежність точності локалізації мобільного робота від температури повітря в приміщенні

Швидкість звуку в повітрі залежить від температури. Це впливає на

точність датчиків, а як наслідок і на точність локалізації мобільного робота. Залежність швидкості звуку  $v$  від температури повітря (табл. 1.2) виражається формулою (1.2):

$$v = \sqrt{\frac{\gamma \cdot R \cdot T}{M}}, \quad (1.2)$$

де  $\gamma = 1,40$  – показник адиабати повітря;  $M = 29 \cdot 10^{-3}$  кг/моль – молярна маса повітря;  $R = 8,31$  Дж/моль·К – газова постійна;  $T$  – температура повітря в Кельвінах.

Таблиця 1.2 - Залежність швидкості звуку від температури повітря

Температура повітря		Швидкість ультразвуку в залежності від температури повітря	
t, °C	T, K	м/с	км/год
0	273,15	331,0	1191,6
10	283,15	337,0	1213,2
20	293,15	342,9	1234,4
22	295,15	344,1	1238,8

Тоді, як зазвичай, для ультразвукових датчиків прийнято вважати  $v = 340$  м/с - швидкість ультразвуку.

Це означає, що при зміні температури навколишнього середовища точність вимірювання даних ультразвукових датчиків мобільного робота знижується. На рисунку 1.4 показано, як похибка датчиків, викликана температурними умовами, впливає на значення отриманих відстаней до перешкоди, де  $L_r$  - відстань до перешкоди, засноване на розрахунку залежно від температури в приміщенні,  $L_s$  - відстань до перешкоди, виходячи з даних ультразвукових датчиків, МАРЕ - помилка локалізації мобільного робота. Як видно, це призводить до затримки повороту в обхід перешкоди і зменшує точність локалізації мобільного робота.

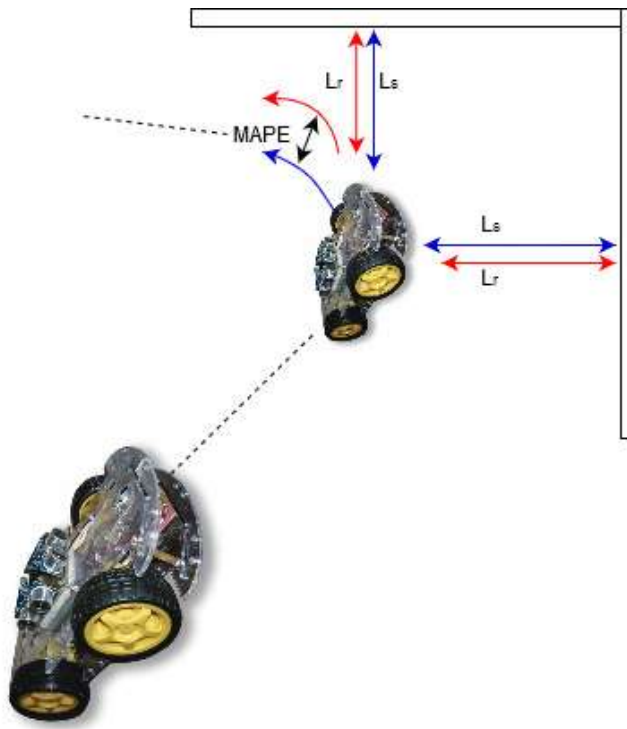


Рисунок 1.4 – Відстеження помилок

Для оцінки похибки, отриманої від датчиків, обчислюємо відстані до перешкоди для різних кімнатних температур (рис. 1.5). Отримані результати порівняли з даними ультразвукових датчиків.

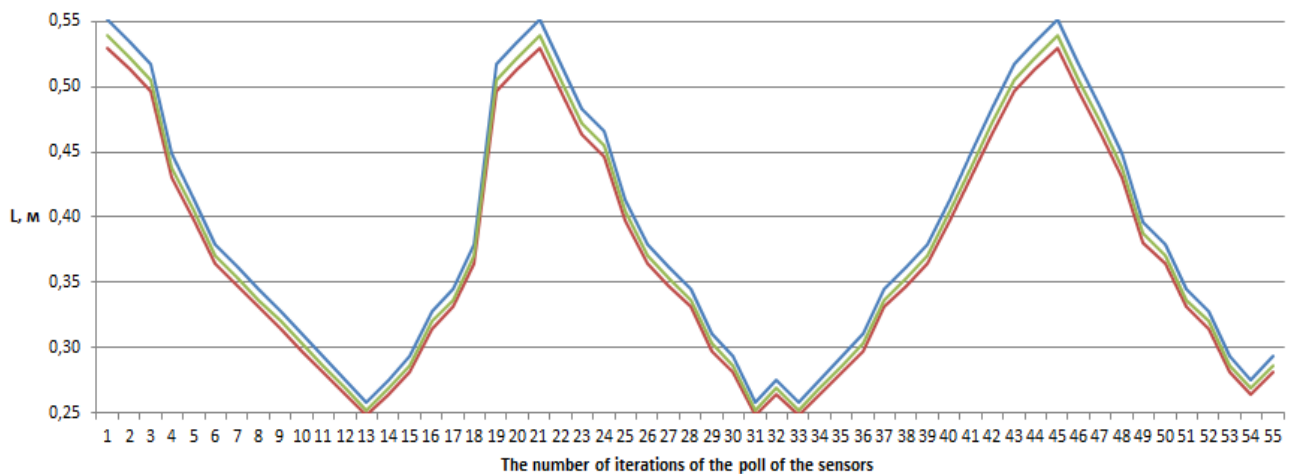


Рисунок 1.5 – Відстань до перешкод при різній кімнатній температурі (червона лінія для 0°C, зелена лінія для 10°C і синя лінія для 22°C)

Таким чином, найменша похибка при локалізації мобільного робота спостерігається при реальній температурі приміщення 22°C (табл. 1.3).

Таблиця 1.3 - Оцінка значень MAPE в залежності від температури приміщень

Температура t, °C	Швидкість ультразвуку, м/с	MAPE, %
0	331,0	4,00
10	337,0	2,26
20	342,9	0,55
22	344,1	0,21

Вирішити ж проблему локалізації для різного температурного режиму можна, встановивши датчик температури і регулювати значення швидкості ультразвуку в залежності від показань датчика температури.

### 1.5 Широтно-імпульсна модуляція

Мікроконтролери зазвичай не можуть видавати довільну напругу. Вони можуть видавати або напругу живлення, або землю (тобто 0 В). Рівнем напруги управляється швидкість враження обертання мотору. Для симуляції неповної напруги використовується широтно-імпульсна модуляція (ШІМ, pulse-width modulation (PWM)). Основною причиною застосування ШІМ є прагнення до підвищення ККД при побудові вторинних джерел живлення електронної апаратури та в інших вузлах. У ШІМ, в якості ключових елементів, використовуються транзистори не в лінійному, а в ключовому режимі, тобто транзистор весь час або вимкнений, або замкнутий. Реалізований в контролерах широтно-імпульсний модулятор складається з двох блоків: лінійного інтегратора (І-ланки) і трипозиційного релейного елемента.

Широтно-імпульсний модулятор генерує послідовність імпульсів з скважністю, пропорційною рівню сигналу на його вході. Параметр його настройки, тобто мінімальна тривалість імпульсу, встановлюється за допомогою зони нечутливості релейного елемента широтно-імпульсного модулятора.

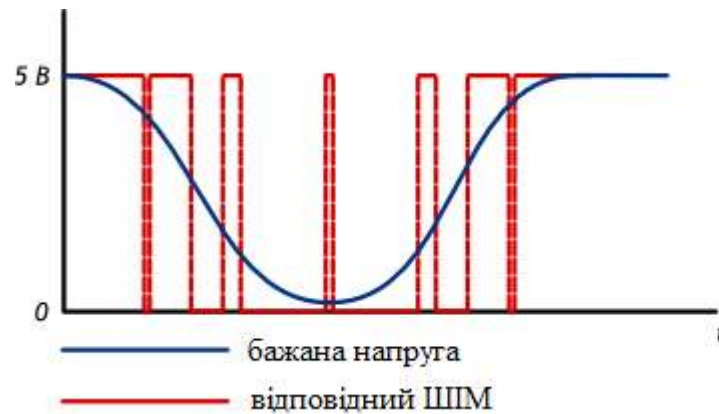


Рисунок 1.6 – Приклад ШІМ на незначній частоті

Вихід мікроконтролера перемикається між землею і Vcc тисячі разів в секунду. Тобто, розігнаний мотор не може зупинити вал за мілісекунди, тому ШІМ-сигнал змусить обертатися його в неповну силу.

Наступний код демонструє, як можна реалізувати управління швидкістю за допомогою ШІМ на прикладі другого таймеру:

```
void drive_3(char d, char speed)
{
char temp;
char sp_1;
sp_1 = speed;
if(sp_1 >100){sp_1 = 100;}
if(d == 1){drv_3_0;
temp = (100 - sp_1);
TIM2->CCR3 = temp;
return;}
if(d == 2){drv_3_1;
TIM2->CCR3 = sp_1;
return;}
if(d == 0){drv_3_1;
TIM2->CCR3 = 0;
return;}
}
```

За допомогою четвертого таймеру, використовуючи ШІМ і в парі порти pinb6 й pinb7, можна фіксувати показання часу та довжину імпульсу:

```

//init capture PWM for timer4 pinb7 and pinb6
void init_capture_pwm(void)
{
RCC->APB1ENR |= 1<<2;
TIM4->CCMR1 |= 1<<9|1<<0; //cc1s = 01, cc2s = 10
TIM4->CCER |= 1<<5; //CC2P = 1
TIM4->SMCR |= 1<<6|1<<4|1<<2; //ts = 101; //SMS = 100
TIM4->CCER |= 1<<4|1<<0; //CC1E,CC2E = 1 enable
capture
TIM4->DIER |= 1<<2; //IRQ of falling IC2 capture
TIM4->PSC = 72; //500kHz
TIM4->CR1 |= 1<<0; //enable
//NVIC_SetPriority(TIM4_IRQn,3);
NVIC_EnableIRQ(TIM4_IRQn);
//TIM4->CNT
}

```

## 1.6 Мікроконтролер STM32: реалізація переривань і таймерів

Оскільки розробка мобільного робота на мікроконтролері STM32 - це повноцінний проект з автономним режимом роботи, то без реалізації переривань і таймерів не обійтися.

Затримка переривання є однією з ключових характеристик мікроконтролера, і це вирішальне значення для багатьох програм із вимогами реального часу.

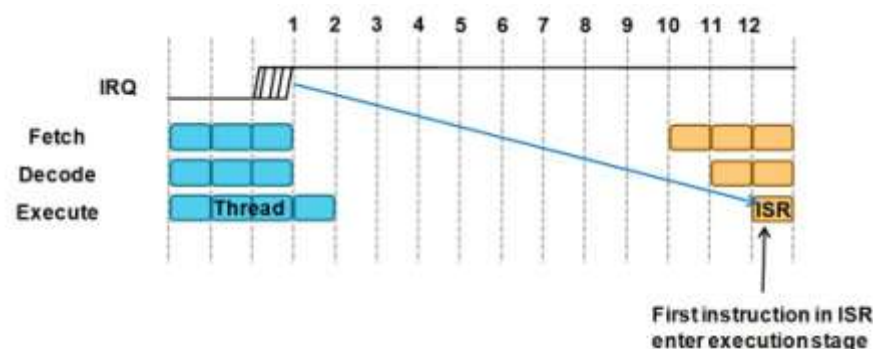


Рисунок 1.7 - Визначення затримки переривання

Переривання (Interrupt Request, або IRQ) - це сигнал, що генерується за певної події, і зазвичай приводить до виконання певної частини коду.



Відповідна частина коду називається Interrupt Service Routine (ISR), або просто оброблювачем переривання. Такі переривання називають зовнішніми, так як вони генеруються зовнішніми по відношенню до мікроконтролеру подіями. Переривання можуть виникати і в самому мікроконтролері, це дозволяє реалізувати асинхронну передачу даних. Частина мікроконтролера, що відповідає за управління переривань, називається Nested Vectored Interrupt Controller, або NVIC. По-перше, адреси ISR зберігаються в спеціально відведеній ділянці пам'яті, так званій vector table. Саме цю адресу (тобто, елемент vector table) прийнято називати вектором. По-друге, переривання мають налаштовуванні пріоритети. Якщо під час обробки переривання прийде більш пріоритетне переривання, що виконується в даний момент, ISR буде припинений, і управління буде передано до більш пріоритетного переривання. Коли він відпрацює, управління буде повернене назад в обробник менш пріоритетного переривання. Звідси і слово nested (вкладені) в назві пристрою.

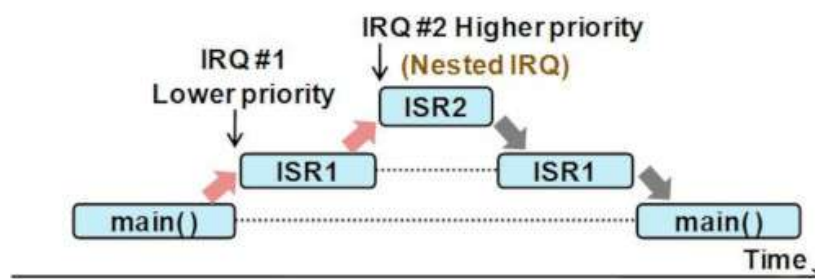


Рисунок 1.8 – Підтримка вкладених переривань

Також в мікроконтролер вбудовані спеціальні пристрої, які називаються таймерами, які вміють генерувати переривання раз в заданий інтервал часу. Таймери мають імена TIM1, TIM2, TIM3, і так далі.

Найважливішою функцією майже всіх додатків на мікроконтролерах є обчислення затримок часу в програмі. В ядро процесора STM32 вбудований 24-бітний системний таймер STK (system timer, SysTick), який рахує в зворотному напрямку від завантаженої в таймер величини до нуля. У момент досягнення значення 0 лічильник автоматично на наступному тактовому перепаді

перезавантажує сам себе значенням з регістра STK\_LOAD, і далі продовжує рахунок вниз, враховуючи кожен імпульс такту, що прийшов. Коли процесор зупинений (halted) для налагодження, лічильник не зменшується. Таймер SysTick спеціально призначений для операційних систем реального часу (real-time operating system, RTOS), але може використовуватися просто як стандартний лічильник вниз. Базова адреса для блоку регістрів SysTick дорівнює 0xE000E010. Лічильник SysTick працює від тактів процесора. Якщо тактовий сигнал зупинений в режимі зниженого енергоспоживання, то лічильник SysTick припиняє рахунок. Нижче наведено код, який ініціалізує SysTick наступним чином:

```
void delay(int zd)
{
    SysTick ->VAL = 0;
    SysTick ->LOAD =zd*9000;
    SysTick ->CTRL |= (1<<0);
    while (!(SysTick ->CTRL & (1<<16))) {}
    SysTick ->CTRL &=~ (1<<0);
}
```

## 2. ВИКОРИСТАННЯ ФІЛЬТРУ КАЛМАНА ДЛЯ ЛОКАЛІЗАЦІЇ МОБІЛЬНОГО РОБОТА

### 2.1 Локалізація мобільного робота

Здатність мобільного робота визначати своє поточне місцезнаходження та сканувати навколишнє середовище є основоположним для його автономної роботи, що зазвичай називається локалізацією. Методи локалізації засновані на виборі орієнтирів зі сканування та їх розпізнаванні для отримання геометричного положення робота. Методи цього типу включають:

- локалізацію Маркова;
- локалізацію з використанням узагальненого фільтра Калмана (GKF);
- локалізацію методу фільтрації частинок.

Для здійснення уникнення перешкод роботом необхідно знати відстані до предметів навколо нього. Для оцінки зміни положення робота використовуються датчики кута двигунів (кодери), візуальні одометри або інерційні вимірювальні прилади. Дуже часто неточності в датчиках впливають на локалізацію мобільного робота. Ці неточності викликані різними факторами; це внутрішні перешкоди датчиків і шум навколишнього середовища. Для подолання цих шумів, для покращення локалізації мобільного робота необхідно застосувати метод фільтрації.

Локалізація мобільного робота проводилась у типовому приміщенні, що містить різноманітні статичні перешкоди. Під час експериментів середня швидкість мобільного робота становила 0,4 м/с.

Робот визначає наявність предметів та відстань до них, спочатку надсилаючи ультразвуковий промінь, а потім отримуючи його із затримкою відбиття. Ультразвукові сигнали, що генеруються приймачем на частоті 40 кГц, відбиваються від перешкоди і повертаються до неї через певний проміжок часу. Саме цей часовий проміжок стає характеристикою, яка допомагає визначити відстань до об'єкту.

Розрахунок відстані  $L$  базується на отриманому часі й обчислюється за такою формулою:

$$L = v \cdot t; t = T/2 \Rightarrow L = v \cdot T/2, \quad (2.1)$$

де  $t$  - час хвилі від датчика до об'єкту,  $T$  - час хвилі від датчика до об'єкту і назад,  $v = 340$  м / с - швидкість ультразвуку.

Поділ на два необхідний, оскільки сигнал проходить відстань до об'єкта і назад, коли потрібна лише відстань до об'єкту. Для вимірювання тривалості сигналу використовується таймер. Таймер встановлений так, що 1 біт становить 0,000001 с. Тому формула (2.1) виглядатиме так:

$$L = v \cdot T/2 \Rightarrow L = v \cdot T_{im} \cdot 10^{-6} / 2, \quad (2.2)$$

де  $T_{im}$  - кількість відкликів таймера в мікросекундах.

Заключна формула розрахунку виглядатиме так:

$$L = 340 \cdot T_{im} \cdot 10^{-6} / 2 = 170 T_{im} / 10,6 \quad (2.3)$$

Щоб виразити цю відстань у см:

$$L_{cm} = 17 T_{im} / 10^3 = 0,017 T_{im}. \quad (2.4)$$

Однак у мікроконтролері не передбачені операції з плаваючою комою. Тому для обчислення відстані необхідно використовувати формулу, де число 0,017 було замінено взаємно оберненим:

$$L_{cm} = T_{im} / 58. \quad (2.5)$$

Наведених формул цілком достатньо для правильного вимірювання

відстані; однак, якщо є необхідність у вдосконаленні вимірювань точності, необхідно застосувати фільтрацію для регулювання інтенсивності шуму при адапційному взятті даних з датчиків. Тому для локалізації мобільного робота використано фільтр Калмана.

## 2.2 Фільтр Калмана

Усі різновиди фільтра Калмана належать до групи байєсівських підходів і можуть, зокрема, використовуватися для локалізації мобільних роботів.

Фільтр Калмана являє собою лінійний рекурсивний алгоритм прогнозування, що використовується для оцінки параметрів моделі процесу. Починаючи з початкових оцінок, фільтр Калмана дозволяє прогнозувати параметри моделі та коригувати його з кожним новим виміром. Його здатність поєднувати ефекти шуму як процесу, так і вимірювань, зробила його дуже популярним у багатьох галузях досліджень та додатках, зокрема в області автономної та допоміжної навігації. З цієї причини фільтр Калмана регулярно використовується системами, що розвиваються з плином часу, де оцінка стану в наступний момент отримується в два етапи. На першому етапі, який називається прогнозуванням, оцінка стану виконується на основі деякої моделі динаміки системи. Другий крок - корекція, оскільки оцінка покращується при деяких вимірюваннях. Таким чином, фільтр Калмана оцінює вектор стану динамічної системи і використовує ряд неповних, неточних вимірювань.

Оцінка відстані виконується за допомогою наступного алгоритму, розробленого авторами в [10-11] та оновленого відповідно до цілей локалізації мобільних роботів. Алгоритм складається з 5 етапів наступним чином.

*Крок 1.* Ініціалізація.

Встановити  $k = 0$ ,  $x_0^-$ ,  $P_0^-$ .

*Крок 2.* Екстраполяція.

а) Оцінка стану екстраполяції:

$$x_{k+1}^- = F \cdot x_k^- + B \cdot u_k, \quad (2.6)$$

де  $x_{k+1}^-$  - прогнозований час ехо-імпульсу до перешкоди в момент часу  $k+1$ ;  $x_k^-$  - прогнозований час в момент часу  $k$ ;  $F$  - матриця переходу станів (динамічна модель системи), яка описує залежність від часу між  $x_k^-$  та  $x_{k+1}^-$ , отриману з історичних даних;  $B$  - матриця застосування керуючого впливу;  $u_k$  - керуючий вплив в попередній момент часу.

б) Передбачення помилки коваріації:

$$P_{k+1}^- = F \cdot P_k^- \cdot F^T + Q_k, \quad (2.7)$$

де  $P_{k+1}^-$  - прогноз помилки;  $P_k^-$  - помилка в попередній момент часу;  $Q_k$  - коваріація шуму процесу.

*Крок 3. Обчислення коефіцієнту посилення фільтра Калмана*

$$K_k = \frac{P_k^- \cdot H^T}{H \cdot P_k^- \cdot H^T + R'}, \quad (2.8)$$

де  $K_k$  - посилення Калмана;  $H$  - матриця вимірювань, що відображає ставлення вимірювань і станів;  $R'$  - коваріація шуму вимірювання.

*Крок 4. Уточнення*

а) Уточнення оцінки з урахуванням виміру  $z_k$ :

$$x_k = x_k^- + K_k \cdot (z_k - H \cdot x_k^-), \quad (2.9)$$

де  $z_k$  - вимір в поточний момент часу.

б) Уточнення помилки коваріації:

$$P_k = (I - K_k \cdot H) \cdot P_k^-, \quad (2.10)$$

де  $I$  - матриця ідентичності.

Якщо стан системи описується однією змінною, то  $I = 1$ , а матриці вироджуються в звичайні рівняння.

*Крок 5:* поки мобільний робот не досягне перешкоди, призначить  $k = k + 1$  і повернутися до кроку 2.

Нижче наведено фрагмент коду локалізації мобільного робота з вбудованою фільтрацією Калмана:

```
void TIM4_IRQHandler(void)
{
int temp;
int keep_temp;
    TIM4->SR &=~ 1<<4|1<<12|1<<0;
keep_temp = TIM4->CCR2;
keep_temp = keep_temp/58;
#define kof 0.8
    if(keep_temp == 0){keep_temp = dist_front; }
        temp = keep_temp;
out_filtr = kof*(float)temp + (1-kof)*data_old;
data_old = out_filtr;
dist_front = (int)out_filtr;
        if(dist_front<= 2 ){dist_front = 0;}
}
```

Цей метод відповідає за оновлення показань датчиків, які є контрольними точками. При зміні положення робота змінюється також поточне положення і нове положення оновлюється за допомогою фільтра Калмана. Після цього робот отримує орієнтири з навколишнього середовища і намагається об'єднати дані з тими, які були доступні раніше.

### 2.3 Результати тестування

В ході експерименту зчитувались показання ультразвукових датчиків

руху мобільного робота. Частота опитування датчиків залежить від відстані між роботом і перешкодою та досягає 80 разів за секунду на відстані 20 см.

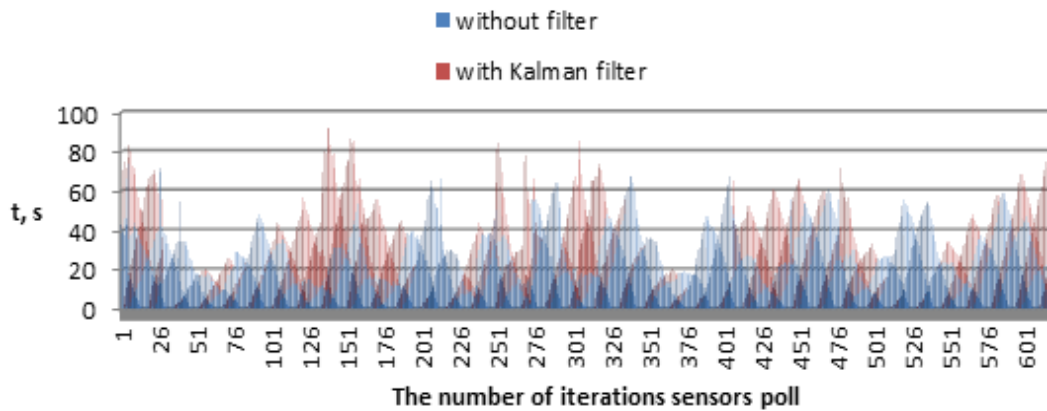


Рисунок 2.1 – Час руху ультразвукових хвиль

На рисунку 2.1 показано час руху ультразвукових хвиль від датчиків мобільного робота до перешкод з і без використання фільтра Калмана.

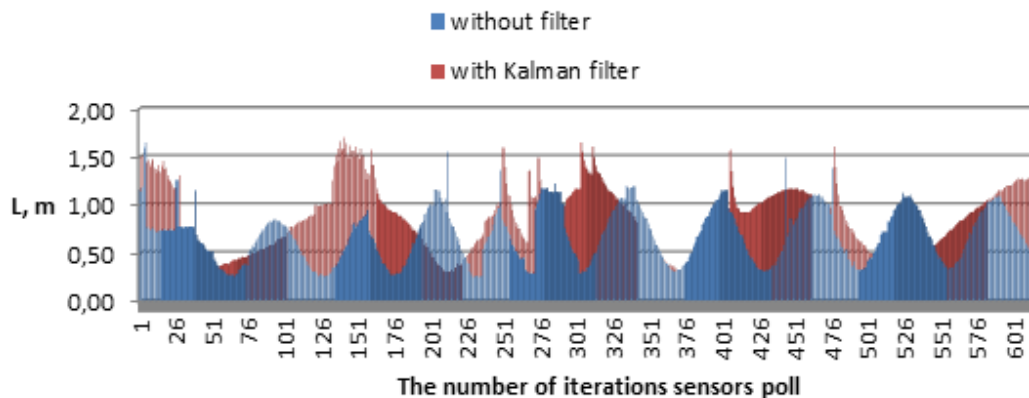


Рисунок 2.2 – Відстань між мобільним роботом і перешкодою

Отримані дані були перераховані згідно методу локалізації. На рисунку 2.2 показано відстань між мобільним роботом і перешкодою з фільтром Калмана і без нього.

На діаграмах 2.1-2.2 та з таблиці 2.1 видно, що застосування фільтра дозволило уникнути додаткових перешкод при зніманні даних з датчика, адаптивно регулювати інтенсивність шуму, що допомогло згладити показання



ультразвукових датчиків мобільного робота, що покращує процес руху та визначення перешкод на шляху мобільного робота. Показник RMSE часу та відстані між мобільним роботом та перешкодою склав відповідно 4,15 с та 0,07 м.

Таблиця 2.1 – Оцінка точності даних

RMSE		
Характеристики	$t_{im}$	$L_{cm}$
без фільтру	5.47 с	0.09 м
з фільтром Калмана	4.15 с	0.07 м

У цій роботі фільтр Калмана використовується для вдосконалення локалізації мобільного робота, оснащеного мікроконтролером та чотирма ультразвуковими датчиками. Запропоновано новий алгоритм запобігання уникненню перешкод у режимі реального часу, достатньо швидкий для впровадження на вбудованих мікроконтролерах автономних мобільних роботів. Застосування фільтра Калмана дозволяє роботів уникнути додаткових перешкод при передачі даних від датчика, адаптуючи адаптивну інтенсивність шуму, що забезпечує плавніше переміщення робота.

### 3. РОЗРОБКА АЛГОРИТМУ УПРАВЛІННЯ РУХОМ МОБІЛЬНОГО РОБОТА

#### 3.1 Алгоритм управління рухом, ґрунтуючись на розташуванні мобільного робота

Завдання створення роботів, здатних переміщатися без допомоги людини, уникаючи зіткнення з перешкодами, тобто здійснення автономного руху, складається в основному з трьох задач.

По-перше, спланований шлях повинен пролягатиме від точки А в точку В.

По-друге, цей шлях повинен забезпечувати рух робота з обходом можливих перешкод.

По-третє, шлях повинен серед всіх можливих шляхів, які відповідають першим двом вимогам, бути в певному сенсі оптимальним.

Основними методами управління рухом вважаються:

- методи на основі графів;
- методи на основі клітинної декомпозиції;
- методи потенційних полів [12];
- оптимізаційні методи;
- методи на інтелектуальних алгоритмах [13-14].

Управління руху ґрунтується на розташування мобільного робота, і як воно відноситься до світової системи координат, що дозволить йому рухатися по точному шляху вздовж послідовності заздалегідь певних точок (рис. 3.1).

Рух робота складається з прямолінійних відрізків окремо від розворотів на місці. Мобільний робот прагне звести до мінімуму загальну подолану відстань, так що завжди відразу повертається передньою частиною до наступної точки та їде прямо до неї.

На першому кроці управління руху, припустимо, що поточний стан мобільного робота  $(x, y, \theta)$  і наступної точкою є  $(W_x, W_y)$ .

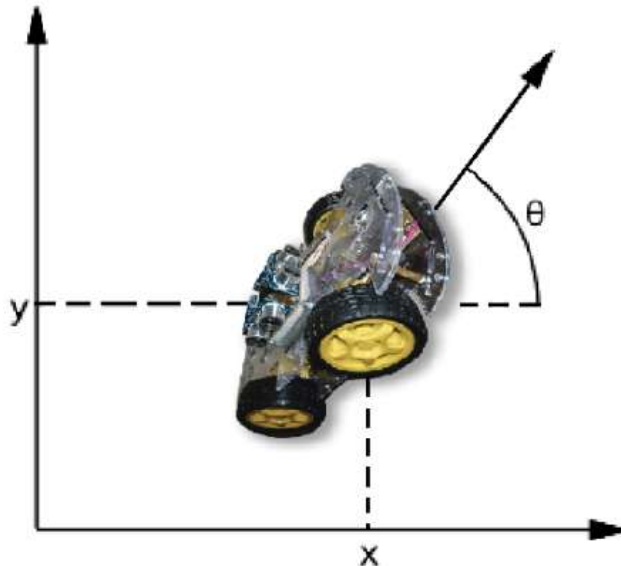


Рисунок 3.1 - Управління руху, ґрунтуючись на розташуванні мобільного робота

Спочатку робот повинен повернутися до зазначеної точки. Вектор спрямування повинен вказувати на:

$$\begin{pmatrix} d_x \\ d_y \end{pmatrix} = \begin{pmatrix} W_x - x \\ W_y - y \end{pmatrix}. \quad (3.1)$$

Абсолютне значення кута в градусах  $\alpha$ , до якого мобільний робот повинен повернутися:

$$\alpha = \tan^{-1} \frac{d_y}{d_x}. \quad (3.2)$$

Необхідно переконатися, що  $\alpha$  знаходиться в правильному квадранті  $-\pi < \alpha \leq \frac{\pi}{2}$ . Робот уже повернутий на певний кут, тому кут на який він повинен повернутися  $\beta = \alpha - \theta$ .

Щоб мобільний робот рухався найбільш ефективно, потрібно змістити кут, додавши або віднімаючи  $2\pi$ , щоб  $-\pi < \beta \leq \pi$ . Після цього, мобільний робот повинен рухатися по прямій на відстань:

$$d = \sqrt{d_x^2 + d_y^2} . \quad (3.3)$$

Тобто, алгоритм управління рухом мобільного робота (рис. 3.2) полягає в тому, що робот сканує навколишнє середовище для визначення наявності перешкод на шляху.

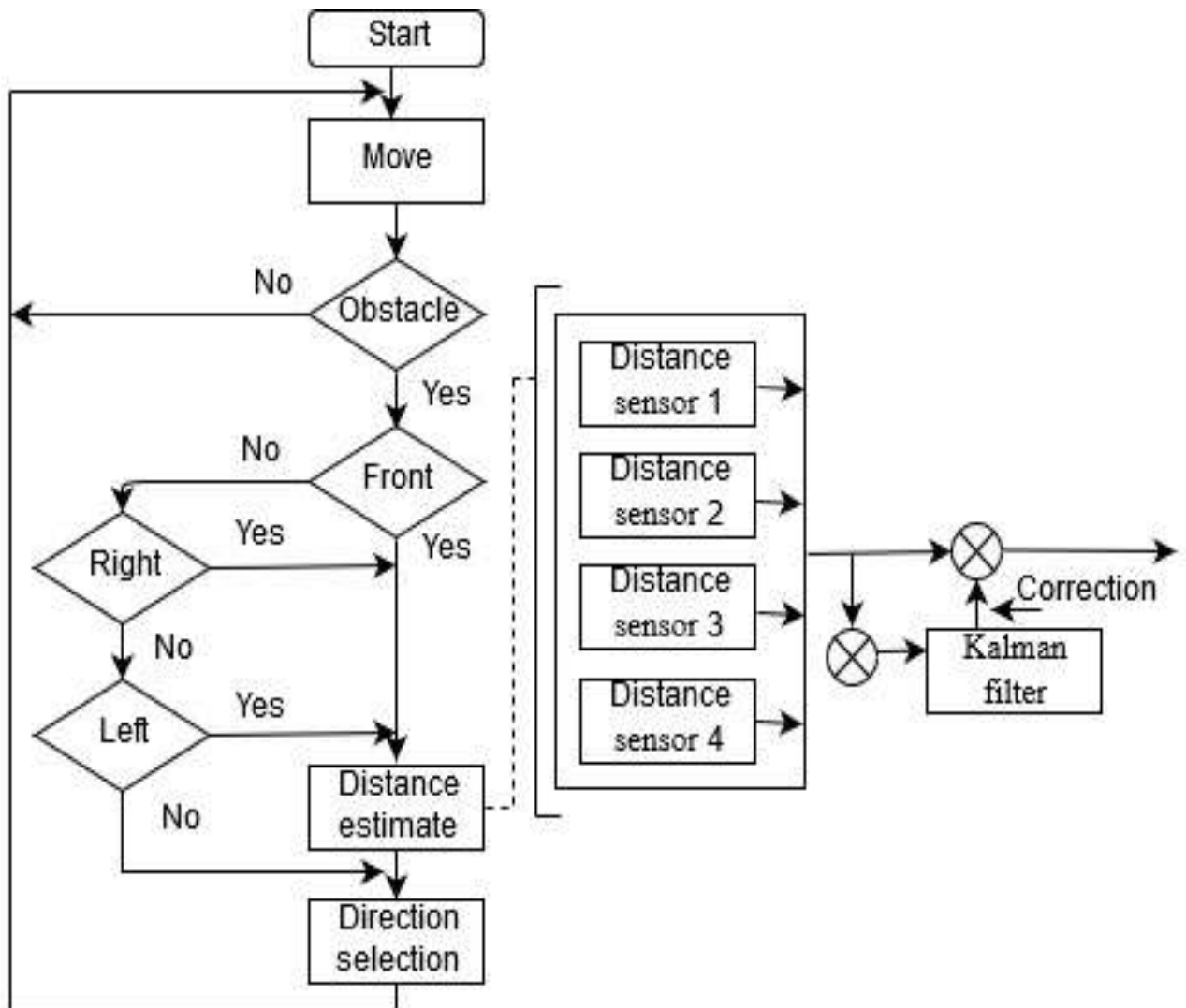


Рисунок 3.2 - Алгоритм управління рухом мобільного робота

Після знаходження шляху, на якому відсутні перешкоди, мобільний робот при русі електромоторів в різних напрямках робить розворот і починає рухатися вперед до тих пір, поки не зустрінеться з новою перешкодою (рис. 3.3).

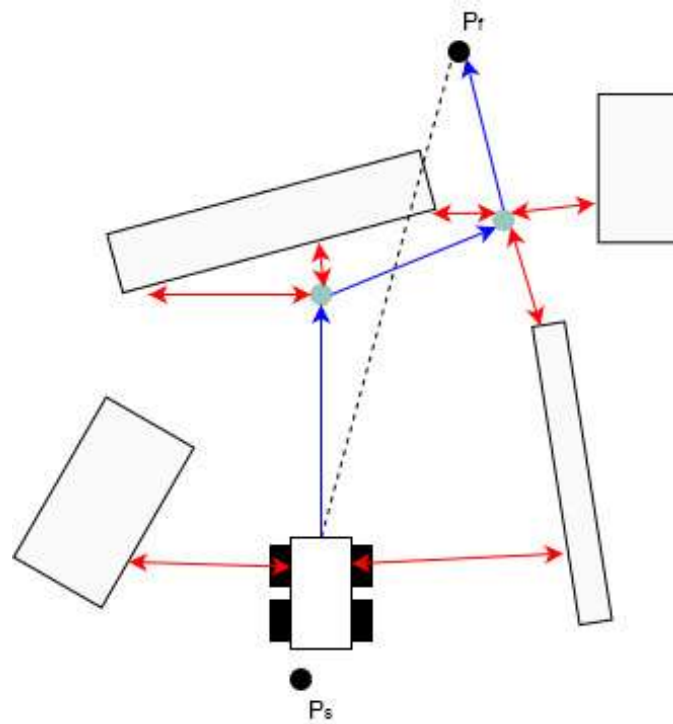


Рисунок 3.3 - Схема уникнення перешкод

### 3.2 Результати використання розробленого алгоритму

Згідно розробленого алгоритму було проведено тестування управління рухом мобільного робота, під час якого були взяті показання ультразвукових датчиків руху мобільного робота (рис. 3.4).

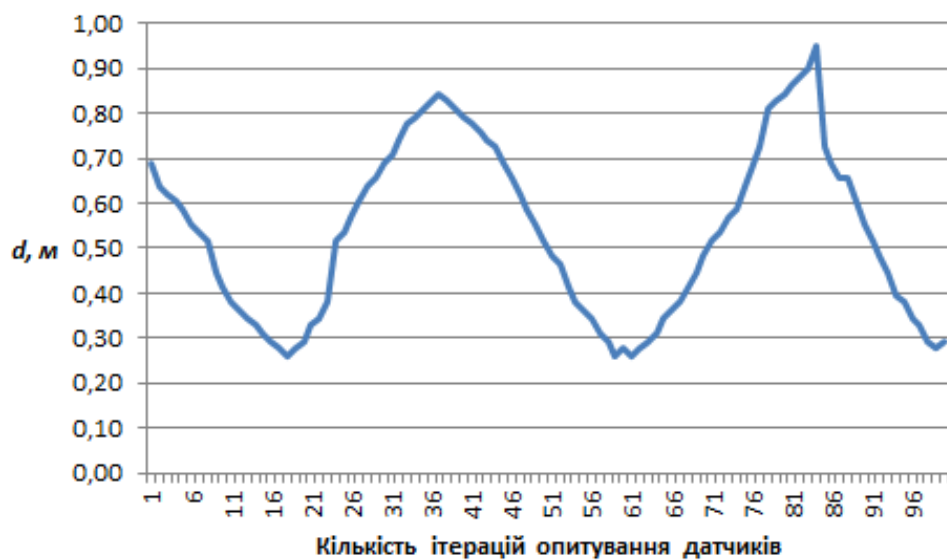


Рисунок 3.4 - Тестування алгоритму управління рухом мобільного робота

На діаграмі точки мінімуму відповідають максимальному наближенню мобільного робота до перешкоди, після чого здійснюється поворот до безпечного напрямку та рух на безпечну відстань.

Розроблено алгоритм управління руху, що базується на розташуванні мобільного робота, який передбачає наступне: спочатку робот рухається прямо до цілі. Якщо виявлена перешкода, то перше, що потрібно зробити, це визначити її розташування відносно робота; коли перешкода знаходиться безпосередньо на шляху робота, йому потрібно вирішити, в який бік повернути, тобто обрати найбільш оптимальним напрямком руху. Мобільний робот перераховує напрямок, намагаючись знайти вільний шлях, і продовжує свій рух у новому напрямку, поки ціль не стане видимою або до появи нової перешкоди.

Результати, отримані завдяки тестуванню розробленого алгоритму, підтверджують прийнятність алгоритму, здатного автоматизувати пересування мобільного робота, який є головним кроком на шляху до створення повністю автономних і багатофункціональних роботів.

## ВИСНОВКИ

В результаті розроблено мобільний робот з ультразвуковими датчиками відстані, які дуже затребувані в робототехнічних проектах через свою відносну простоту, достатню точність та доступність.

Для вдосконалення локалізації використовувався фільтр Калмана, в результаті чого RMSE становив 4,15 с і 0,07 м, а також встановлено залежність точності локалізації від температури повітря в приміщенні, де відбувається тестування мобільного робота.

Застосування фільтрації дозволяє уникнути додаткових перешкод при зніманні даних з датчика, регулювати інтенсивність шуму, що забезпечує плавніше переміщення робота. Наступною перевагою запропонованого підходу є те, що стабільність фільтра можна відстежувати в режимі онлайн, а також надійність та ступінь використання інформації спостережень адаптивно коригуються, завдяки чому продуктивність фільтра може бути поліпшена.

Розроблено алгоритм управління напрямком руху мобільного робота, принцип дії якого полягає в наступному: ультразвукові датчики сканують навколишню місцевість для визначення наявності перешкод на шляху мобільного робота. Після знаходження шляху, на якому відсутні перешкоди, мобільний робот при русі електромоторів в різних напрямках робить розворот і починає рухатися вперед до тих пір, поки не зустрінеться з новою перешкодою.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- [1] Nagy C. Ultrasound-Based Indoor Robot Localization Using Ambient Temperature Compensation / C. Nagy, Z. Biró-Ambrus, L. Márton // *Acta Universitatis Sapientiae Electrical and Mechanical Engineering*. – Vol. 8. – 2016. – pp. 19-28.
- [2] Petrovski K. On the Kalman Filter Approach for Localization of Mobile Robots / K. Petrovski, S. Jovanovski, M. Mirchev, L. Basnarkov // *International Conference on ICT Innovations*. – 2016. – pp. 123-133.
- [3] Stańczek P. Digital signal processing in ultrasonic based navigation system for mobile robots / P. Stańczek // *ITM Web of Conferences*. – Vol. 15. – 2017.
- [4] Moussa M. Ultrasonic based heading estimation for aiding land vehicle navigation in GNSS denied environment / M. Moussa, A. Moussa, N. El-Sheimy // *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. – Vol. XLII-1. – 2018. – pp. 315-322.
- [5] Kondratenko Y. A simulation model for robot's slip displacement sensors / Y. Kondratenko, O. Gerasin, A. Topalov // *International Journal of Computing*. – Vol.15, Issue 4. – 2016. – pp. 224-236.
- [6] Li S.E. Kalman filter-based tracking of moving objects using linear ultrasonic sensor array for road vehicles / S.E. Li, G. Li, J. Yu, C. Liu, B. Cheng, J. Wang, K. Li // *Mechanical Systems and Signal Processing*. – Vol. 98. – 2018. – pp. 173-189.
- [7] Keke G. A fuzzy controller: Using monocular computer vision to see and avoid obstacle for quadcopter / G. Keke, L. Wei, T. Ligu // *5th intern. workshop on computer science and engineering: Information processing and control engineering: WCSE 2015-IPSE: Proc. Chenghu: Science and Engineering Inst.* – 2015.
- [8] <https://www.keil.com/download/>
- [9] <http://www.keil.com/dd/docs/arm/st/stm32f10x/stm32f10x.h>



[10] Mohd Nasir N.Z. Autonomous mobile robot localization using Kalman filter / N. Z. Mohd Nasir, M. A. Zakaria, S. Razali, M.Y. bin Abu // MATEC Web of Conferences. – Vol. 90. – 2017.

[11] Skarga-Bandurova I. A Framework for Real-Time Public Transport Information Acquisition and Arrival Time Prediction Based on GPS Data / I. Skarga-Bandurova, M. Derkach, A. Velykzhanin // Dependable IoT for Human and Industry: Modeling, Architecting, Implementation. –River Publishers Series in Information Science and Technology (Eds. V. Kharchenko, Ah L. Kor, A. Rucinski). – 2018. – pp. 411-431.

[12] Alvarez D. 3D robot formations path planning with fast marching square / D. Alvarez, J.V. Gomez, S. Garrido, L. Moreno // J. of Intelligent and Robotic Systems. – 2015. Vol. – 80, No. 3-4. – pp. 507–523.

[13] Ma J.-C. Multi-behavior fusion-based path planning for mobile robot / J.-C. Ma, Q. Zhang, L.-Y. Ma, W. Xie // Beijing Ligong Daxue Xuebao // Trans. of Beijing Inst. of Technology. – 2014. – Vol. 34, No. 6. – pp. 576–581.

[14] Yu J. Dynamical Tracking of Surrounding Objects for Road Vehicles using Linearly-Arrayed Ultrasonic Sensors / J. Yu, S.E. Li // IEEE Intelligent Vehicles Symposium (IV) . – 2016. – pp. 72-77.