

<https://doi.org/10.31891/2219-9365-2024-77-12>

УДК 629.05

ЗОЛОТАРЬОВ Євгеній

Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського"  
email: [yevheniyzolotarev@gmail.com](mailto:yevheniyzolotarev@gmail.com)

БУРАУ Надія

Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського"  
email: [nbuaru@ukr.net](mailto:nbuaru@ukr.net)

## ЗАСТОСУВАННЯ ТЕХНОЛОГІЙ NODE-RED ДЛЯ ВІЗУАЛІЗАЦІЇ РУХУ ОБ'ЄКТІВ

Статтю присвячено розробці систем візуалізації руху безпілотних рухомих об'єктів, зокрема автономних безпілотних підводних апаратів, на основі використання інструменту для візуального програмування потоків даних Node-RED. Розробку системи візуалізації виконано на прикладі використання Node-RED з інтеграцією інерціальної навігаційної системи фірми «Inertial Labs», до складу якої входять акселерометри, гіроскопи, магнітометри, приймач глобальної навігаційної супутникової системи та антена. Для візуалізації руху об'єкту, на якому встановлена навігаційна система, використовуються дані про орієнтацію, швидкість та координати місцеположення. Демонструється структура системи візуалізації, засоби розробки та засоби візуалізації руху. Моніторинг системи візуалізації та середовища розробки відбувається у веб-сторінці, що дозволяє не використовувати спеціальне обладнання зі специфічним програмним забезпеченням.

Для системи візуалізації розроблено спеціальну програму в середовищі Node-RED, яка розділена за функціональним призначенням на програму запуску та зупинки навігаційної системи, програму отримання та обробки даних від навігаційної системи, та програму візуалізації даних. Як результат розробки системи представлено графічний інтерфейс, в якому реалізовано запуск та зупинку навігаційної системи, візуальне представлення даних сенсорів, візуалізація місцеположення об'єкта на мапі, текстові та/або графічні (індикаторні) представлення даних орієнтації, швидкості, тощо. Виконавча програма інтерфейсу в середовищі Node-RED має об'ємний вид, у якості прикладу наведено програму для формування візуалізації тексту для координат та програму з використанням скалярної векторної графіки для візуалізації кута курсу об'єкта.

Використання Node-RED дозволяє розробляти нові системи візуалізації, збору та аналізу даних про рух безпілотних об'єктів у складі навігаційних комплексів таких об'єктів для забезпечення можливості моніторингу їх функціонального технічного стану та виконання поставлених завдань.

Ключові слова: безпілотні апарати, система візуалізації руху, інерціальна навігаційна система, візуальне програмування потоків даних, Node-RED.

ZOLO TAROV Yevgenii, BOURAOU Nadiya

National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"

## APPLICATION OF NODE-RED TECHNOLOGIES FOR VISUALIZING THE MOVEMENT OF OBJECTS

The article is devoted to the development of visualization systems for the movement of unmanned moving objects, in particular, autonomous unmanned underwater vehicles, based on the use of the tool for visual programming of data streams Node-RED. The visualization system was developed using the example of using Node-RED with the integration of the Inertial Labs inertial navigation system, which includes accelerometers, gyroscopes, magnetometers, global navigation satellite system receiver. Orientation, speed and coordinates are used to visualize the movement of the object on which the navigation system is installed. The structure of the visualization system, development tools, and motion visualization tools are demonstrated. Monitoring of the visualization system and the development environment takes place in a web page, which allows you to avoid using special hardware with specific software.

A special program in the Node-RED environment has been developed for the visualization system, which is divided by functional purpose into a program for starting and stopping the navigation system, a program for receiving and processing data from the navigation system, and a data visualization program. As a result of the development of the system, a graphical interface is presented, which implements the start and stop of the navigation system, visual presentation of sensor data, visualization of the location of the object on the map, text and/or graphic (indicator) presentation of orientation, speed data, etc. The executive program of the interface in the Node-RED environment has a three-dimensional view, as an example, a program for forming a text visualization for coordinates and a program using scalar vector graphics (SVG) for visualizing the angle of the course of an object are given.

The use of Node-RED makes it possible to develop new systems of visualization, collection and analysis of data on the movement of unmanned objects as part of the navigation complexes of such objects to ensure the possibility of monitoring their functional technical condition and the fulfillment of assigned tasks.

Keywords: unmanned vehicles, motion visualization system, inertial navigation system, visual programming of data streams, Node-RED.

### Постановка проблеми у загальному вигляді

#### та її зв'язок із важливими науковими чи практичними завданнями

Одним з важливих напрямків досліджень при розробці багатоцільових автономних безпілотних рухомих об'єктів, зокрема, автономних безпілотних підводних апаратів (АБПА), є розробка та вдосконалення систем забезпечення руху (систем енергозабезпечення, стабілізації, навігації та керування) та систем зв'язку і передачі інформації між АБПА та зовнішнім світом [1-3]. Важливим аспектом таких досліджень є необхідність комплексного підходу до вирішення завдань орієнтації, навігації, прийому-передачі та візуалізації інформації для АБПА. Відповідно до [2,3] система візуалізації руху АБПА є невід'ємною частиною програмно-апаратного навігаційного комплексу підводного апарату та його системи комунікації з судном-носієм. А визначення та відображення на борту судна-носія інформації про поточне місцезнаходження АБПА є важливим функціоналом навігаційного комплексу як для забезпечення можливості моніторингу функціонального технічного стану АБПА та виконання поставлених завдань, так і для дистанційного керування апаратом.

В роботі [3] проведено огляд технічних та програмних засобів візуалізації даних, на основі якого сформовано загальні функціональні вимоги до системи візуалізації руху АБПА, такі як: відображення даних у реальному часі в інтерфейсі системи; програмна підтримка різних типів сенсорів для збору даних; стабільність з'єднання; точність відображення позиції апарату. Для забезпечення зазначених вимог використовуються модульна архітектура програмного забезпечення (ПЗ) та сучасні технології програмування на основі веб-мов, таких, наприклад, як HTML, CSS і Node-RED [4,5].

Node-RED — це інструмент для візуального програмування потоків даних, який спочатку був розроблений командою IBM Emerging Technology Services, а тепер є частиною OpenJS Foundation [6]. Програмування потоків — це спосіб опису поведінки програми, як мережі чорних ящиків або «Вузлів» (Nodes), які мають чітко визначене призначення, тип вузла позначається відповідним кольором та іконкою. Кожному вузлу надаються деякі дані, з якими він оперує та передає ці дані далі, а за потік даних між вузлами відповідає мережа [6], що робить таку модель зручною для візуального подання та сприйняття процесів. Інструмент Node-RED дає можливість розробляти програмне забезпечення для програмно-апаратних систем, які включають в себе різноманітні IoT пристрої (IoT-Internet of Things, Інтернет речей), застосунки та сервіси [7].

На сьогоднішній день, не зважаючи на значний розвиток та поширення технологій IoT, кількість прикладних науково-технічних проєктів з використанням Node-RED все ще залишається обмеженою і не розкриває потужні можливості таких технологій для створення мережі «Сенсори - комп'ютерні системи» для систем керування та візуалізації руху АБПА. Такі обмеження значною мірою обумовлені технічними проблемами інтеграції навігаційних систем в системи візуалізації через відсутність відповідного програмного забезпечення, а також складністю побудови дисплеїв, схожих на аналогові прилади, через відсутність двовимірної графіки.

### Формулювання цілей статті

Метою статті є побудова системи візуалізації руху на прикладі використання Node-RED з інтеграцією інерціальної навігаційної системи (ІНС) фірми «Inertial Labs» [8].

### Виклад основного матеріалу

Для розробки системи візуалізації використаємо ІНС фірми «Inertial Labs», яка є безплатформною системою і забезпечує визначення з високою точністю місцеположення, швидкості та орієнтації (кутів курсу, тангажу та крену) як нерухомих, так і рухомих об'єктів [8]. До складу ІНС входить такий стандартний набір сенсорів: акселерометри; гіроскопи; магнітометри; приймач глобальної навігаційної супутникової системи (ГНСС) та антена. Основними вихідними даними з ІНС, необхідними для візуалізації руху об'єкту, є орієнтація, швидкість та координати місцеположення.

Систему візуалізації даних ІНС побудуємо з використанням розробленого в середовищі Node-RED програмного забезпечення (ПЗ) відповідно до наведеної на рис. 1 узагальненої функціональної схеми.

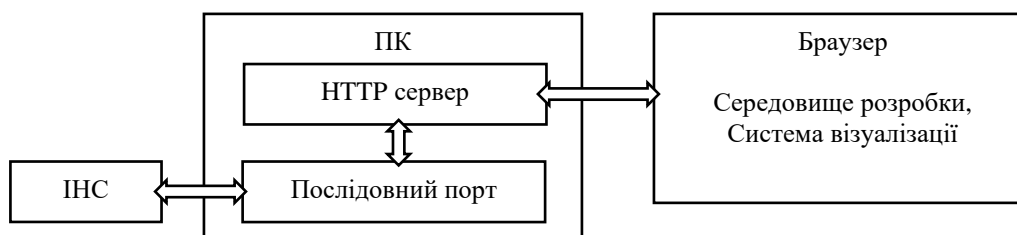


Рис. 1 Узагальнена функціональна схема системи візуалізації даних ІНС

Персональний комп'ютер (ПК) використовується в якості локального сервера на якому встановлено Node-RED. Інструкції з встановлення доступні за посиланням [9]. Для обміну даних між ІНС та ПК використовується послідовний порт. Правила обміну даних називають протоколом, ІНС використовує протокол, описаний в табл. 1.

Таблиця 1

**Протокол обміну даних ІНС**

Номер байту	0	1	2	3	4, 5	6 .. (n+5)	n+6, n+7
Назва			Тип	Номер	Довжина	Корисна інформація	Контрольна сума
Тип даних	uint8	uint8	uint8	uint8	uint16	n uint8 (змінюється)	uint16
Нотатки	0xAA	0x55					

Для запуску ІНС необхідно подати відповідні команди з ПК через послідовний порт по протоколу, описаному в табл. 1. Використаємо дві команди, що описані в табл. 2.

Таблиця 2

**Конфігураційні команди ІНС**

Номер байту	0	1	2	3	4	5	6	7	8
Назва			Тип	Номер	Довжина		Корисна інформація	Контрольна сума	
Тип даних	uint8	uint8	uint8	uint8	uint8	uint8	n uint8	uint16	
Команда Стоп	0xAA	0x55	0x00	0x00	0x07	0x00	0xFE	0x05	0x01
Команда OPVT2AHR	0xAA	0x55	0x00	0x00	0x07	0x00	0x58	0x5F	0x00

При відправці команд від ПК до ІНС поля «Тип» і «Номер» повідомлення заповнюються нулями. Номер команди передається в полі «Корисна інформація», де для «0xFE» - відповідає команді «Стоп», а «0x58» відповідає команді «OPVT2AHR». Після відправки з ПК команди «Стоп» ІНС перестає працювати. Відправка з ПК команди «OPVT2AHR» (0x58) запускає видачу даних від ІНС до ПК по протоколу, описаному в табл. 1. В полі «Корисна інформація» буде міститись постійна структура формату даних для команди «OPVT2AHR», що наведена в табл. 3 [10].

Таблиця 3

**Корисна інформація формату даних OPVT2AHR**

Опис	Тип даних	Нотатки
Кути орієнтації	uint16	Курс, град*100
	int16	Тангаж, град *100
	int16	Крен, град *100
Гіроскопи	3×int32	Кутова швидкість, град/сек*1.0e5
Акселерометри	3×int32	Прискорення, g*1.0e6
Магнітометри	3×int16	Магнітне поле, нТл/10
Статус	uint16	Статус ІНС
Напруга	uint16	V*100
Температура	int16	°C*10
Позиція	int64	Широта, град *1.0e9
	int64	Довгота, град *1.0e9
	int32	Висота, м*1000
Швидкості	3×int32	Східна, північна, вертикальна швидкість, м/с *100
Позиція приймача	int64	Широта ГНСС, град *1.0e9
	int64	Довгота ГНСС, град *1.0e9
	int32	Висота ГНСС, м*1000
Швидкості приймача	int32	ГНСС горизонтальна швидкість, м/с *100
	uint16	ГНСС шляховий курс, град*100
	int32	ГНСС вертикальна швидкість, м/с*100
Мітка часу	uint32	Мілісекунди від початку контрольного тижня ГНСС
Статус часу	uint8	вказує на якість опорного часу ГНСС
Інформація приймача	uint8	Сервісна інформація
Кількість супутників	uint8	
Затримка часу	uint16	затримка логу швидкості в мілісекундах.
Статус двох антенного курсу	uint8	Характеризує якість рішення для двох антенного курсу
Курс з двох антенного приймача	uint16	Курс ГНСС, град*100
Часова затримка логів	3×int16	Часова затримка логів приймача для двох антенного курсу, позиції та швидкості відносно ГНСС часу.
Барометр	uint16	Тиск, Па/2
	int32	Барометрична висота, м*1000
Нові дані приймача	uint8	

Наведена в табл. 3 інформація має достатньо даних, щоб використати її в системі візуалізації.

Для системи візуалізації в середовищі Node-RED створимо програму, яка складається з потоків (Flow). Потоки – це вузли (Nodes), пов’язані між собою інформаційними дротами (Wires), що виконуються як умовно незалежні програми при виникненні певних подій [11]. Середовище розробки відкривається в браузері за посиланням <http://127.0.0.1:1880> на ПК. Розділимо програму для системи візуалізації на 3 частини за функціональним призначенням: запуск та зупинка роботи ІНС; отримання та обробка даних; візуалізація даних. Розглянемо розробку кожної з програм.

### Програма запуску та зупинки ІНС

Фізичні основи роботи ІНС побудовані на інтегруванні сигналів акселерометрів і гіроскопів, тому в таких системах особливу увагу приділяють процедурі початкової виставки. Перед початком роботи для забезпечення нормального режиму функціонування ІНС користувач повинен впевнитись в нерухомості об’єкта, на якому встановлена ІНС для запуску. Запуск і зупинка ІНС – це конфігураційні команди, які були описані вище в табл. 2. Графічний вид написаної програми в Node-RED для відправки з ПК команд запуску та зупинки ІНС через послідовний порт наведено на рис. 2.



Рис. 2 Графічний вид Node-RED програми для запуску та зупинки ІНС через послідовний порт

Вузли-кнопки створюють на графічному інтерфейсі кнопки «Старт» та «Стоп». Вони з’єднані з вузлами-функціями «OPVTHR\_fun» та «STOP\_fun». У вузлах-функціях написаний код на мові програмування Java script, що створює бінарне повідомлення, описане в табл. 2. Далі вузли-функції з’єднується з вузлом послідовного порту «COM7», що виконує відправку повідомлення з ПК до ІНС. Приклад коду, що виконується у вузлах-функціях, наведено в табл. 4:

Таблиця 4

Код у вузлах-функціях «OPVTHR\_fun» і «STOP\_fun»

Назва вузла-функції	Код
OPVTHR_fun	<code>msg.payload = Buffer.from("AA5500000700585F00","hex"); return msg;</code>
STOP_fun	<code>msg.payload = Buffer.from("AA5500000700FE0501","hex"); return msg;</code>

У якості елементів керування для відправки відповідних команд на графічному інтерфейсі будуть створені кнопки «Старт» та «Стоп». Мова програмування Node-RED подійно-орієнтована, тому натискання кнопки - це подія, яка запускає вище описані ланцюги зв’язків для відправки через послідовний порт відповідної команди від ПК до ІНС.

### Програма отримання та обробки даних від ІНС

Після натискання в графічному інтерфейсі кнопки «Старт», ІНС проводить процедуру початкової виставки, після чого ІНС видає повідомлення до ПК через послідовний порт з фіксованою частотою 50 Hz. Структура повідомлення описана вище в табл. 1 та табл. 3. Графічний вид програми в середовищі Node-RED для отримання та обробки даних від ІНС наведено на рис. 3.

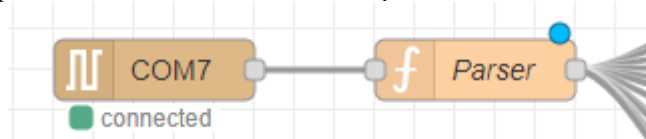


Рис. 3 Графічний вид Node-RED програми для отримання та обробки даних від ІНС

Програма складається з двох вузлів:

- «COM7» - вузол послідовного порту, що приймає повідомлення від ІНС до ПК у вигляді масиву, як послідовність байт;
- «Parser» - вузол-функція обробляє прийняту послідовність байт від вузла «COM7», перевіряє цілісність повідомлення та передає декодовану інформацію у вузли, що відповідають за візуалізацію даних.

Код для вузла-функції «Parser», написаний на Java script, можна поділити на дві частини – ядро і декодування інформації. Ядро перевіряє правильність повідомлення, а декодування представляє собою перетворення послідовностей байт в числа з плаваючою точкою.

Для перетворення байт в числа використовувались стандартні методи мови програмування Java Script [12]:

- readInt16LE() – перетворення 2 байт зі знаком в число;
- readUInt16LE() – перетворення 2 байт без знаку в число;
- readInt32LE() – перетворення 4 байт зі знаком в число;
- readUInt32LE() – перетворення 4 байт без знаку в число;
- readBigInt64LE() – перетворення 8 байт зі знаком в число.

Перетворені байти в числа та направляються вузлам, що відповідають за візуалізацію даних.

### Програма візуалізації даних

Декодовані дані з ІНС відображаються в режимі реального часу на веб-сторінці «<http://127.0.0.1:1880/ui>» графічного інтерфейсу системи візуалізації, що показаний на рис 4. Доступ до графічного інтерфейсу можливий і з інших пристроїв, таких як мобільний телефон або планшет при підключенні їх до локальної мережі. В такому випадку для доступу до середовища розробки або візуалізації даних в веб-сторінці необхідно вказувати локальний IP адрес, замість IP - 127.0.0.1.

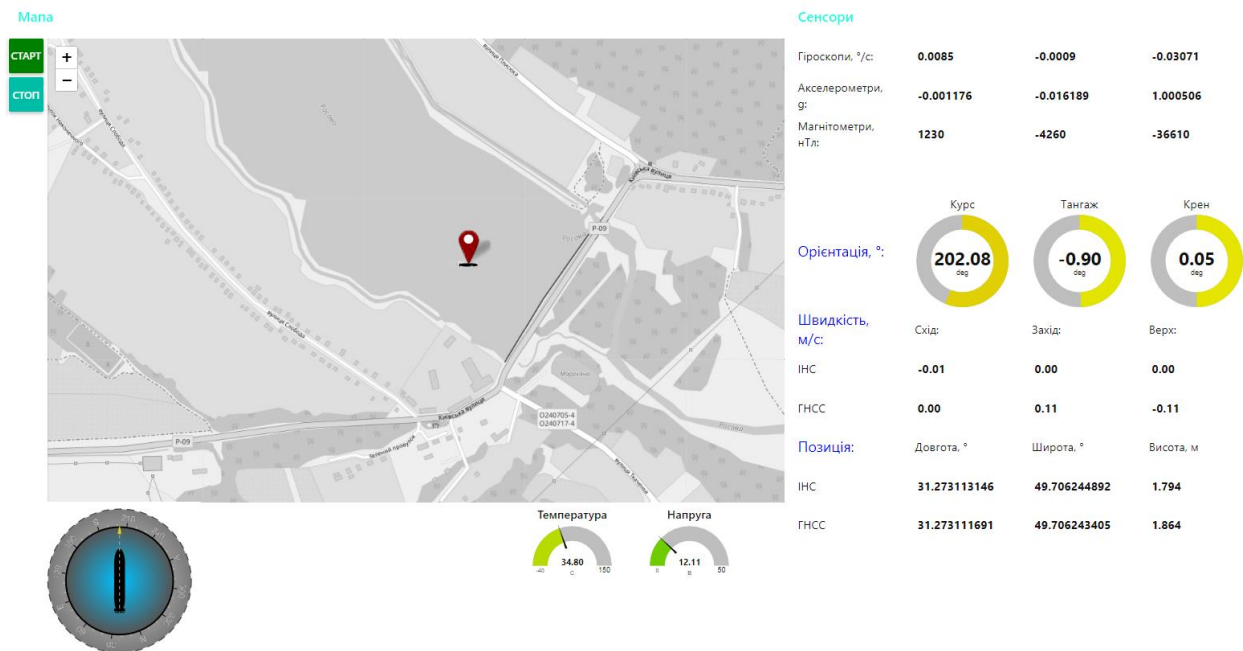


Рис. 4. Графічний інтерфейс системи візуалізації даних

Графічний інтерфейс забезпечує візуальне представлення даних:

- гіроскопи, акселерометри, магнітометри – об'єднані в групу сенсори, що показана в правому верхньому куті, вихідна інформація сенсорів показується у вигляді тексту;
- кути орієнтації – зображені трьома графічними модулями (widget), що розміщені під групою сенсорів, кути показані у вигляді тексту та кругового індикатора. Окремо для кута курсу зроблена анімаційна скалярна векторна графіка (SVG), на якій зображено нерухомий АБПА з осью лінійю та коловою шкалою поділок з кроком 10 град, що обертається в залежності від значення курсу ІНС (розміщена в лівому нижньому куті);
- швидкість – данні швидкості ІНС та ГНСС виводяться в географічній системі координат (схід, північ, вертикаль) і об'єднані в групу швидкості, що знаходиться під групою даних орієнтації. Значення швидкості показуються у вигляді тексту;
- позиція – данні позиції ІНС (широта, довгота), ГНСС виводяться в географічній системі координат, об'єднані в групу даних позиції і знаходяться під групою даних швидкості. Значення позиції показуються в вигляді тексту;
- напруга живлення, температура – зображені в якості двох графічних модулів, що розміщені внизу по центру, значення показуються у вигляді тексту та напів-кругового індикатора зі стрілкою;

- мапа – візуалізація місцеположення за значеннями широти та довготи на мапі «OpenStreetMap». Для зображення іконки АБПА використана SVG графіка(силует об'єкту АБПА). Координати беруться з ІНС, а напрямок залежить від значення курсу ІНС. ГНСС використовує стандартну іконку червоної мітки;

- в лівому верхньому куті знаходяться кнопки «Старт» і «Стоп», при натисканні яких виконуються операції відповідно запуску та зупинки ІНС;

Виконавча програма інтерфейсу в середовищі Node-RED має об'ємний вид, тому наведемо лише приклад формування візуалізації тексту для координат та використання скалярної векторної графіки (SVG графіки) для курсу.

- **Програма візуалізації позиції (координат)**

Позиція (місцеположення) АБПА за поточними значеннями широти та довготи відображається на мапі «OpenStreetMap», дані надходять з ІНС та ГНСС у вигляді іконічної інформації. Для більшої інформативності виконано дублювання позиції з ІНС та ГНСС у вигляді тексту.

Графічний вид програми, яка реалізує візуалізацію позиції в середовищі розробки Node-RED, зображено на рис. 5:

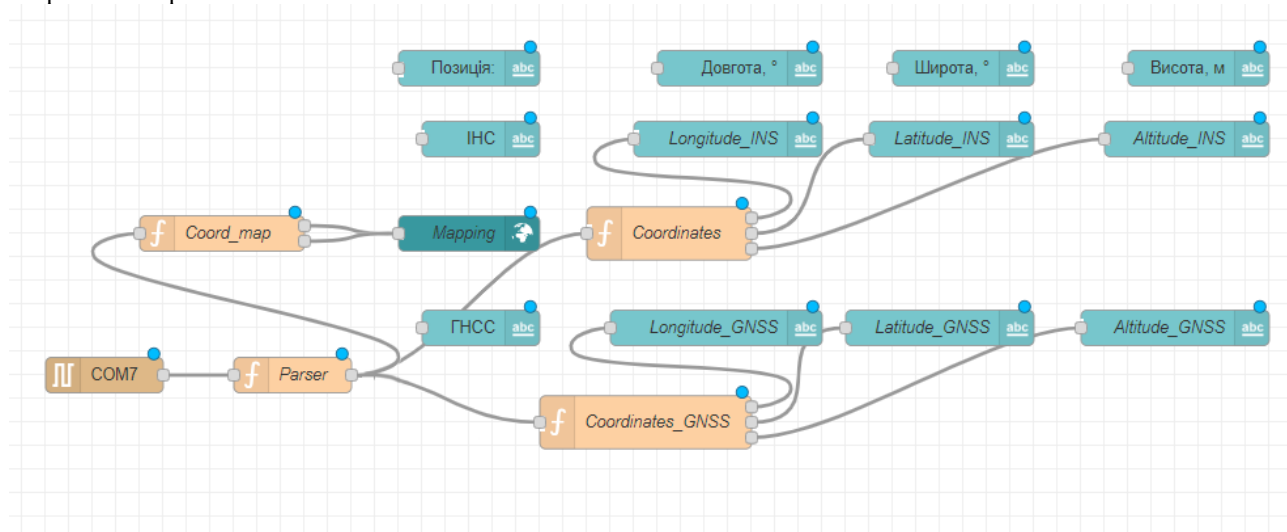


Рис. 5. Графічний вид Node -RED програми для візуалізації позиції

Вузол послідовного порту «COM7» з'єднаний з вузлом-функцією «Parser», з якого отримується декодована інформація для візуалізації даних. Вузол-функція «Coord\_map» формує два повідомлення в необхідному форматі для вузла мапи «Mapping», який відображає положення АБПА з різних джерел інформації.

Текстові вузли, які не з'єднані жодним потоком, є статичними написами на рис.4. Вузол-функція «Coordinates» має три виходи, що з'єднуються з вузлами тексту «Longitude\_INS», «Latitude\_INS», «Altitude\_INS», що показують значення в текстовому вигляді довготи, широти, висоти з ІНС. Вузол-функція «Coordinates\_GNSS» має три виходи що з'єднуються з вузлами тексту «Longitude\_GNSS», «Latitude\_GNSS», «Altitude\_GNSS», які показують значення в текстовому вигляді довготи, широти, висоти з ГНСС. Аналогічним чином відображається інформація для гіроскопів, акселерометрів та магнітометрів, а також дані орієнтації, напруги та температури. Код для отримання даних з вузлів-функцій «Coordinates», «Coordinates\_GNSS» написаний з використанням мови Java script, наведено в табл. 5.

Таблиця 5

Код у вузлах- функціях для даних позиції

Назва вузла функції	Код
Coordinates	return [msg.Longitude, msg.Latitude, msg.Altitude];
Coordinates_GNSS	return [msg.Longitude_GNSS, msg.Latitude_GNSS,msg.Altitude_GNSS];

- **SVG графіка**

В лівому нижньому куті інтерфейсу на рис. 4 використовується розроблена анімаційна скалярна векторна графіка для зображення поточного курсу об'єкта. Ідею використання анімаційної двовимірної графіки було запропоновано в [13], графічний вид розробленої Node -RED програми наведено на рис. 6.

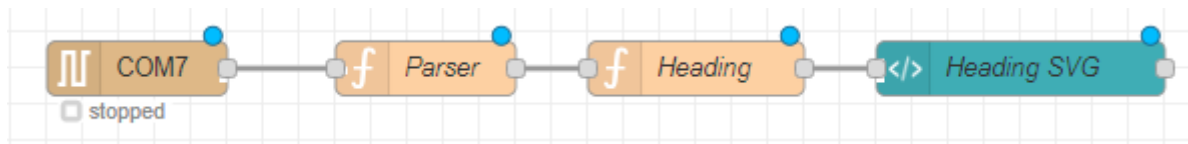


Рис. 6 Графічний вид Node -RED програми для скалярної векторної графіки

Вузли «COM7» і «Parser» повторно використовуються, як і в попередніх програмах. Вузол-функція «Heading» передає значення кута курсу у вузол «Orientation SVG» так само, як і описані в табл. 5 вузли-функції даних позиції. Розробка графіки зроблена в онлайн програмному забезпеченні Voxy SVG [14]. Вихідний код з Voxy SVG використовується у вузлі віджеті «Heading SVG», тіло якого являє собою код на мові розмітки гіпертекстових документів HTML. Обертання колової шкали поділок в залежності від кута курсу реалізовано за допомогою наведеного нижче коду:

```
// Отримання id номера шару з SVG, що відповідає за колову шкалу  
var Circle = document.getElementById('rotated');  
// Обретання колової шкали поділок  
Circle.setAttribute('transform', 'rotate({{-msg.payload}}, 69.4 , 72.3)')
```

Об'єкт «Circle» приймає параметри SVG графіки для шару, що відповідає коловій шкалі поділок (при розробці графіки коловій шкалі присвоїли id назву «rotated»). Застосування методу «setAttribute (кут повороту, центр обертання x, центр обертання y)» для об'єкту «Circle» забезпечує обертання колової шкали поділок.

Таким чином, створене в середовищі Node-RED програмне забезпечення дає можливість прийняти від інерціальної навігаційної системи та відобразити на розробленому графічному інтерфейсі множину даних інформації про місцезнаходження, просторове положення та швидкість об'єкту, на якому буде встановлено таку навігаційну систему.

#### Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Запропоновано та обгрунтовано вирішення проблеми візуалізації руху АБПА на основі комплексного підходу, відповідно до якого система візуалізації є невід'ємною складовою навігаційно-пілотажного комплексу АБПА, структура і характеристики якого повинні забезпечувати задані показники точності навігації та позиціонування, автономності роботи, передачі даних, стійкості та надійності роботи в умовах впливу зовнішніх факторів. Обгрунтовано використання інструментарію для візуального програмування потоків даних Node-RED для побудови системи візуалізації руху АБПА, на основі чого розроблено програмне забезпечення для системи візуалізації даних інерціальної навігаційної системи фірми «Inertial Labs». Результат розробки системи представлено графічним інтерфейсом, в якому реалізовано запуск та зупинку ІНС, візуальне представлення даних сенсорів, візуалізація місцезнаходження об'єкта на мапі, текстові та/або графічні (індикаторні) представлення даних орієнтації, швидкості, температури.

Створена система є базовою, вдосконалення якої може забезпечити розробка нової векторної графіки для кутів крену і тангажу в одному стилі, використання тривимірної графіки на базі програмного забезпечення «three.js» для візуалізації кутового руху, а також розробка бібліотеки для Node-RED спільноти, що дозволить експортувати її в інші передові проекти.

#### Література

1. Gafurov S. A., Klochkov E.V. Autonomous unmanned underwater vehicles development tendencies. *Procedia Engineering*. 2015. No106, P. 141 – 148.
2. Underwater vehicles / Edited by Aleksander V. Inzartsev. 2009. 582 p.
3. Бурау, Н. І., Золотарьов, Є. О. (2023). Системи візуалізації руху безпілотних підводних апаратів. Вчені записки Таврійського національного університету імені В. І. Вернадського. Серія: Технічні науки. Тавр. нац. ун-т ім. В. І. Вернадського, Том.34 (73), № 3, С. 83-89. <https://doi.org/10.32782/2663-5941/2023.3.1/14>
4. Jayasinghe M. C. Graphical User Interface for the Supervisory Motion Control of BlueROV1.1. *2nd International Conference On Electrical Engineering (EECON)*. 2018. <https://doi.org/10.1109/eecon.2018.8540998>
5. A different approach to controlling the Bluerov2. Blue Robotics Community Forums. URL: <https://discuss.bluerobotics.com/t/a-different-approach-to-controlling-the-bluerov2/3072> (дата звернення: 14.05.2023).
6. Node-RED. OpenJS Foundation & Contributors. URL:<https://nodered.org/> (дата звернення:20.12.2023).

7. ProgIngContrSystems (б. д.). Матеріали дисципліни «Програмна інженерія в системах управління». [https://pupenasan.github.io/ProgIngContrSystems/%D0%9B%D0%B5%D0%BA%D1%86/2\\_nodered.html](https://pupenasan.github.io/ProgIngContrSystems/%D0%9B%D0%B5%D0%BA%D1%86/2_nodered.html)
8. GPS-Aided Inertial Navigation System. [https://www.inertiallabs.com/wp-content/uploads/2023/10/INS-U\\_Datasheet\\_rev-2.12\\_Oct\\_2023.pdf](https://www.inertiallabs.com/wp-content/uploads/2023/10/INS-U_Datasheet_rev-2.12_Oct_2023.pdf)
9. Running Node-RED locally : Node-RED. (б. д.). Node-RED. <https://nodered.org/docs/getting-started/local>
10. Example of parsing data of the OPVT and OPVT2AHR data formats. (б. д.). Inertial Labs. <https://inertiallabs.com/example-of-parsing-data-of-the-opvt-and-opvt2ahr-data-formats/>
11. ЛР1. Основи роботи з Node-RED: Знайомство з Node-RED. (б. д.). Школа автоматики. <http://edu.asu.in.ua/mod/book/view.php?id=123&chapterid=306>
12. Node.js Buffer.readUInt16LE() Method - GeeksforGeeks. (б. д.). GeeksforGeeks. [https://www.geeksforgeeks.org/node-js-buffer-readuint16le-method/?ref=ml\\_lbp](https://www.geeksforgeeks.org/node-js-buffer-readuint16le-method/?ref=ml_lbp)
13. Yaser Ali Husen. (2023, 15 грудня). Rotate SVG Simulation in Node-RED [Відео]. YouTube. <https://www.youtube.com/watch?v=xM17HLvLoI4> [6] <https://boxy-svg.com/>
14. Boxy SVG Editor. (б. д.). Boxy SVG Editor. <https://boxy-svg.com/>

### References

1. Gafurov S. A., Klochkov E.V. Autonomous unmanned underwater vehicles development tendencies. Procedia Engineering. 2015. No106, P. 141 – 148.
2. Underwater vehicles / Edited by Aleksander V. Inzartsev. 2009. 582 p.
3. Burau, N. I., Zolotarov, Ye. O. (2023). Systemy vizualizatsii rukhu bezpilotnykh pidvodnykh aparativ. Vcheni zapysky Tavriiskoho natsionalnoho universytetu imeni V. I. Vernadskoho. Seriya: Tekhnichni nauky. Tavri. nats. un-t im. V. I. Vernadskoho, Tom.34 (73), № 3, S. 83-89. <https://doi.org/10.32782/2663-5941/2023.3.1/14>
4. Jayasinghe M. C. Graphical User Interface for the Supervisory Motion Control of BlueROV1.1. 2nd International Conference On Electrical Engineering (EECON). 2018. <https://doi.org/10.1109/eecon.2018.8540998>
5. A different approach to controlling the Bluerov2. Blue Robotics Community Forums. URL: <https://discuss.bluerobotics.com/t/a-different-approach-to-controlling-the-bluerov2/3072> (data zvernennia: 14.05.2023).
6. Node-RED. OpenJS Foundation & Contributors. URL:<https://nodered.org/> (data zvernennia:20.12.2023).
7. ProgIngContrSystems (б. д.). Materialy dystsypliny «Prohramna inzheneriia v systemakh upravlinnia». [https://pupenasan.github.io/ProgIngContrSystems/%D0%9B%D0%B5%D0%BA%D1%86/2\\_nodered.html](https://pupenasan.github.io/ProgIngContrSystems/%D0%9B%D0%B5%D0%BA%D1%86/2_nodered.html)
8. GPS-Aided Inertial Navigation System. [https://www.inertiallabs.com/wp-content/uploads/2023/10/INS-U\\_Datasheet\\_rev-2.12\\_Oct\\_2023.pdf](https://www.inertiallabs.com/wp-content/uploads/2023/10/INS-U_Datasheet_rev-2.12_Oct_2023.pdf)
9. Running Node-RED locally : Node-RED. (б. д.). Node-RED. <https://nodered.org/docs/getting-started/local>
10. Example of parsing data of the OPVT and OPVT2AHR data formats. (б. д.). Inertial Labs. <https://inertiallabs.com/example-of-parsing-data-of-the-opvt-and-opvt2ahr-data-formats/>
11. ЛР1. Osnovy roboty z Node-RED: Znaiomstvo z Node-RED. (б. д.). Shkola avtomatyky. <http://edu.asu.in.ua/mod/book/view.php?id=123&chapterid=306>
12. Node.js Buffer.readUInt16LE() Method - GeeksforGeeks. (б. д.). GeeksforGeeks. [https://www.geeksforgeeks.org/node-js-buffer-readuint16le-method/?ref=ml\\_lbp](https://www.geeksforgeeks.org/node-js-buffer-readuint16le-method/?ref=ml_lbp)
13. Yaser Ali Husen. (2023, 15 hrudnia). Rotate SVG Simulation in Node-RED [Video]. YouTube. <https://www.youtube.com/watch?v=xM17HLvLoI4> [6] <https://boxy-svg.com/>
14. Boxy SVG Editor. (б. д.). Boxy SVG Editor. <https://boxy-svg.com/>