

## Лабораторна робота №1. Тема. Основи роботи з Node-RED

**Мета:**

**Цілі:**

Зауваження. Зробити короткі відомості.

Node-RED - це інструмент програмування для об'єднання апаратних пристроїв, API та онлайн-послуг новими та цікавими способами.

Редактор функціонує на основі браузера, а програма виглядає як об'єднані в потоки (flow) вузли (node), що вибираються з палітри.

Після редагування потоки можуть бути розгорнуті в середовище виконання в один клік. Вбудована бібліотека дозволяє зберігати корисні функції, шаблони або потоки для повторного використання. Легке середовище виконання побудоване на Node.js, що використовує подіємо-керовану неблокуючу модель. Це робить його ідеальним для роботи на краю (Edge) мережі на недорогих апаратних засобах, таких як Raspberry PI, а також у хмарі.

З більш ніж 225 000 модулів у репозиторію пакетів Node, легко збільшити діапазон вузлів палітри для додавання нових можливостей.

Потоки, створені в Node-RED, зберігаються за допомогою JSON, які можна легко імпортувати та експортувати для спільного використання з іншими.

Онлайн бібліотека потоків дозволяє вам поділитися своїми найкращими потоками зі світом.

### Про Node.JS

<https://habr.com/post/243335/>

<https://uk.wikipedia.org/wiki/Node.js>

### Інсталяція для Андроїд.

<https://nodered.org/docs/platforms/android>

Використання програми Termux у магазині додатків дозволяє легко запускати Node-RED на пристроях Android. Ви можете отримати його з Play Market.

1. З Play Market завантажте та встановіть [Termux](#)

2.

```
apt update
apt upgrade
apt install coreutils nano nodejs
npm i -g --unsafe-perm node-red
node-red
```

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

## 1. Інсталяція Node-RED під Windows

<https://nodered.org/docs/platforms/windows>

Підтримувані платформи: Windows 10, Windows 8, Windows 7

1.1. Завантажити msi-файл Node.JS LTS версії <https://nodejs.org/uk/>

1.2. Запустити на виконання msi-файл від імені адміністратора і встановити Node.JS, при виклику діалоговий вікон все залишати за замовченням

1.3. Після інсталяції запустити командний рядок (CMD), у якому введіть

```
node --version && npm -version
```

повинно вивести версію node() - npm ()

**npm (Node Package Manager)** - це менеджер пакунків для мови програмування JavaScript. Для середовища виконання Node.js є менеджером пакунків за замовчуванням. Включає в себе клієнт командного рядка, який також називається npm, а також онлайн-базу даних публічних та приватних пакунків, яка називається реєстром npm. Реєстр доступний через клієнт, а доступні пакунки можна переглядати та шукати через веб-сайт npm. Менеджер пакунків та реєстр керуються npm, Inc.

1.4. Install Node-RED . Інсталювання проводиться з використанням npm з командою install. Наберіть в командному рядку

```
npm install -g --unsafe-perm node-red
```

почнеться процедура інсталяції

1.5. Запустіть Node-RED з командного рядка

```
node-red
```

Можуть виникнути повідомлення про розблокування брандмауером, з якими треба погодитись.

1.6. Відкрийте браузер, перейдіть до редактору Node-Red, за посиланням <http://127.0.0.1:1880/>

Для того, щоб Node-RED виконувався, вікно з командним рядком не можна закривати.

## 2. Знайомство з Node-RED

Програма створена на Node-RED складається з **потоків (Flow)**, які виконуються як умовно незалежні програми. Потоки – це зв'язані між собою інформаційними **дротами (wires)** **вузли (Node)**, що виконують певну функцію. Таким чином, ідеологія програмування Node-RED дещо схоже на побудову програм на мові FBD, що є стандартною для програмування ПЛК (IEC 61131-3). Тим не менше, між цими мовами є значні відмінності.

2.1. Відкрийте в браузері редактор Node-RED, якщо він ще не відкритий. Ознайомтеся з його основними частинами (рис.1)

Детально про роботу з редактором Ви можете ознайомитися в інструкції користувача.

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

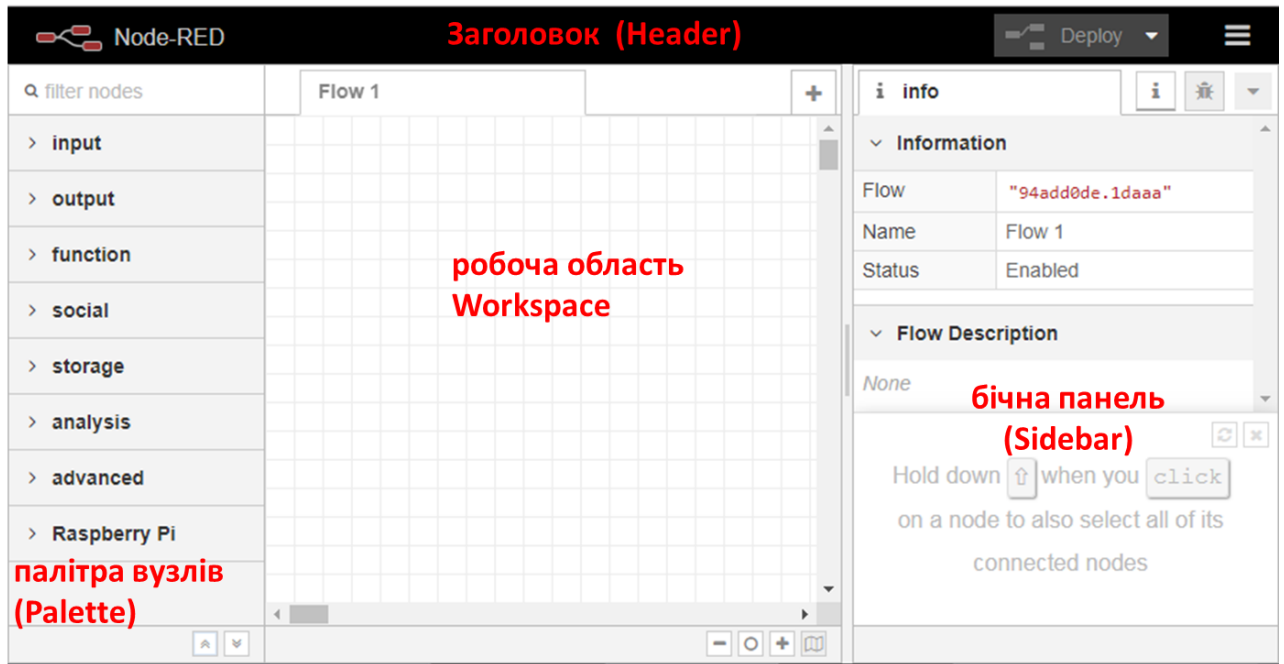


рис.1. Вигляд редактору Node-RED

2.2. Виберіть з палітри і розмістіть на робочій області вузли

- **Input->inject**
- **Output->Debug**

З'єднайте їх між собою. Повинно вийти як на рис.2. Блакитні кружечки значать, що зміна в вузлах ще не відобразилася в середовищі виконання, так як змінена програма не була в ньому розгорнута.

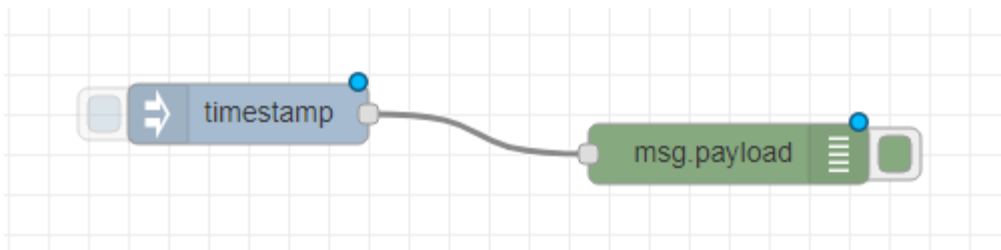


рис.2. Вигляд програми

2.3. В заголовку виберіть пункт **Deploy->Modified Nodes** (див.рис.3), після чого натисніть **Deploy (Розгортання)**.

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

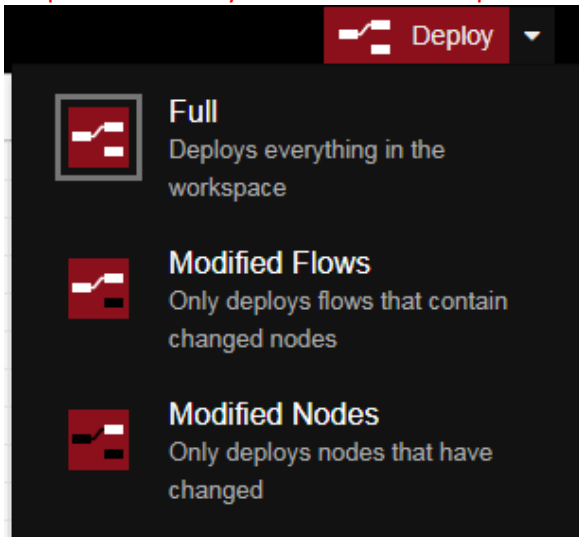


рис.3. Вибір варіанту розгортання

При вдалому розгортанні з'явиться повідомлення.



А в робочому просторі вузли вже будуть без блакитних кружечків.

2.4. Для перевірки роботи програми, на бічній панелі треба відобразити вікно **Debug messages** (налагоджувальні повідомлення) шляхом натиснення кнопки з «жучком».

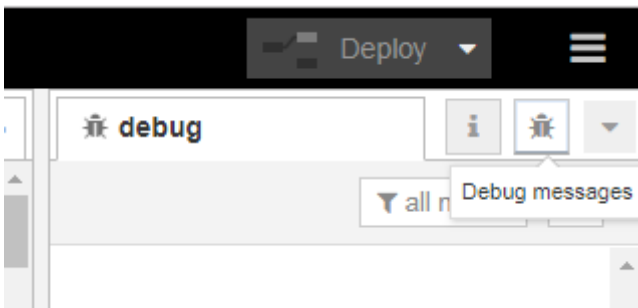


рис.4. Показати вікно повідомлень

Ліворуч вузла типу Inject з назвою «timestamp» є кнопка, яка приводить до ініціювання розрахунку ланцюжка вузлів, що починаються з нього. Натисніть на цю кнопку.

В результаті з'явиться повідомлення про успішне вприскування (Inject) а на панелі повідомлень з'явиться повідомлення (рис.5).

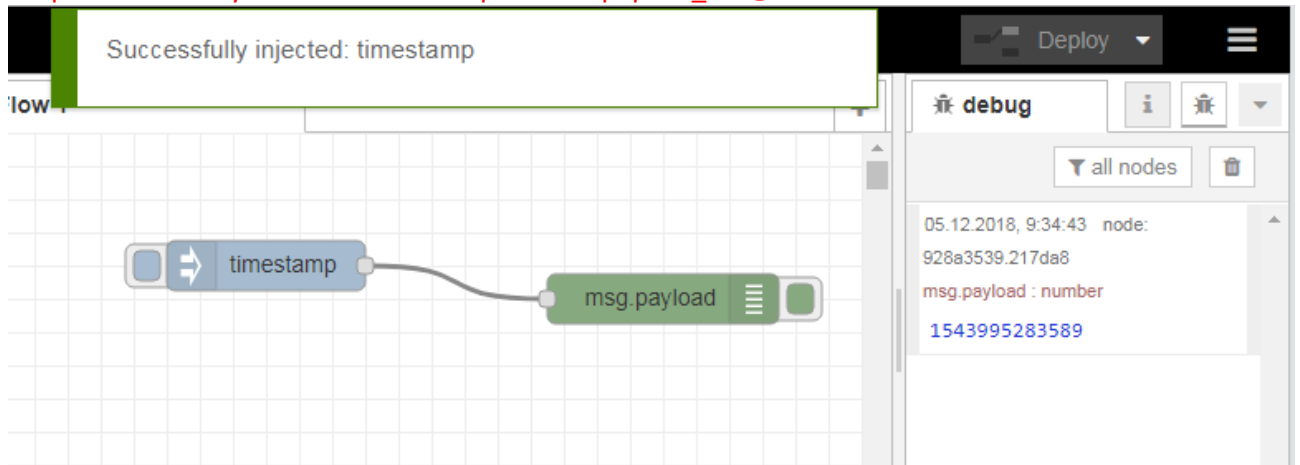


рис.5. Результат виконання програми

На цьому прикладі розглянемо, як виконується програма.

У більшості випадків перерахунок вузлів починається тоді, коли на його вхід подається повідомлення (message). Повідомлення – це прості об'єкти (типу структурні змінні) JavaScript що можуть мати будь який набір властивостей. Тобто в даній програмі після перерахунку вузла з іменем «timestamp» буде сформовано об'єкт-повідомлення (**msg**) і переданий по дроту вузлу з іменем (msg.payload).

На вхід вузла з іменем «timestamp» повідомлення не надходять, бо він є ініціатором розрахунку. Всі вузли палітри що входять в групу **Input** є ініціаторами розрахунку. Ініціація вузлів типу **Inject** відбувається шляхом ручного запуску по кнопці, або через певні інтервали часу, що налаштовується у вузлі. Ініціювання повідомлення це формування полів **msg** та відправка його іншим вузлам по дротам. Повідомлення, надіслане вузлом Inject, має властивості **payload** (корисне навантаження) та **topic** (тема). За замовченням Inject записує у властивість topic відмітку часу (timestamp – кількість мілісекунд з 1980 року).

Вузол типу Debug «msg.payload» використовується для відображення повідомлень на бічній панелі Debug. Таким чином, після отримання повідомлення, цей вузол відображає його зміст на бічній панелі.

2.5. Змініть налаштування властивостей вузлів, як показано на рис.6: змініть імена вузлів, вкажіть тему (topic) та періодичність оновлення для вузла типу Inject. Вікно налаштування з'являється по подвійному кліку по вузлу.

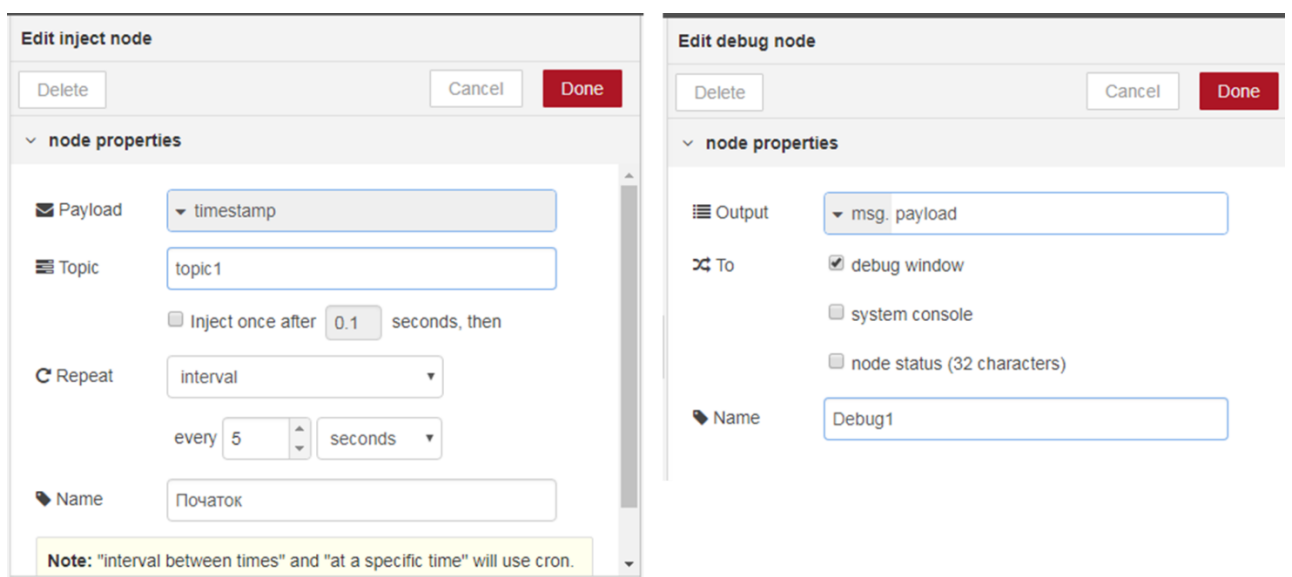


рис.6. Налаштування вузлів

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

Зробіть розгортання, та проаналізуйте зміст виведених у вікні Debug повідомлень.

2.6. Змініть вузол «Початок» так, щоб він формував корисне навантаження текстом «Це текстове повідомлення»(рис.7), та проаналізуйте як воно виводиться на вікно Debug.

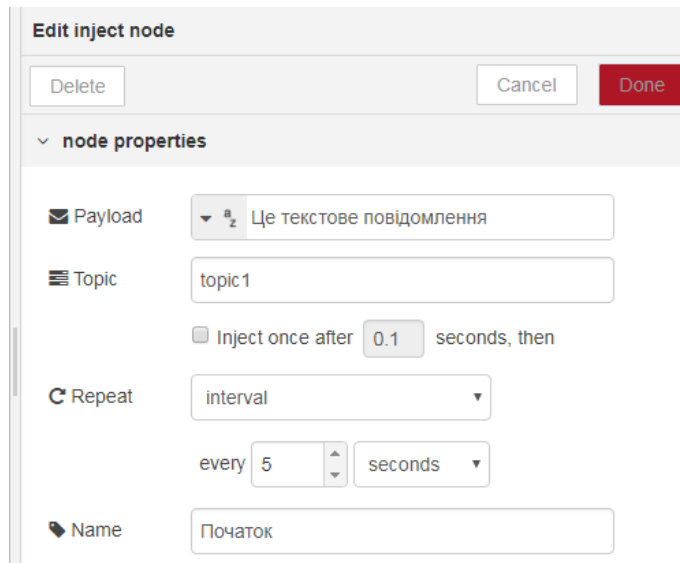


рис.7. Формування текстового повідомлення

2.7. Змініть вузол «Початок» так, щоб він знову формував корисне навантаження відміткою часу (Timestamp). Розгорніть (deploy) програму та проконтролюйте щоб відмітка часу кожні 5 секунд відображалася у вікні повідомлень

Ознайомтеся з роботою вузлів типу **change** та **delay** в інструкції користувача.

Змініть програму, як показано на рис.8, використовуючи вузли delay (“delay 1s” ...”delay 4s”) та change (“set1” ...”set5”). Для вузлів delay виставте затримки:

- delay 1s – 1 seconds
- delay 2s – 2 seconds
- delay 3s – 3 seconds
- delay 4s – 4 seconds

Для вузлів change виставте правило рівним «set», та змініть властивості «to» на наступні текстові поля:

- set1 – один
- set2 – два
- set3 – три
- set4 – чотири
- set5 – п'ять

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

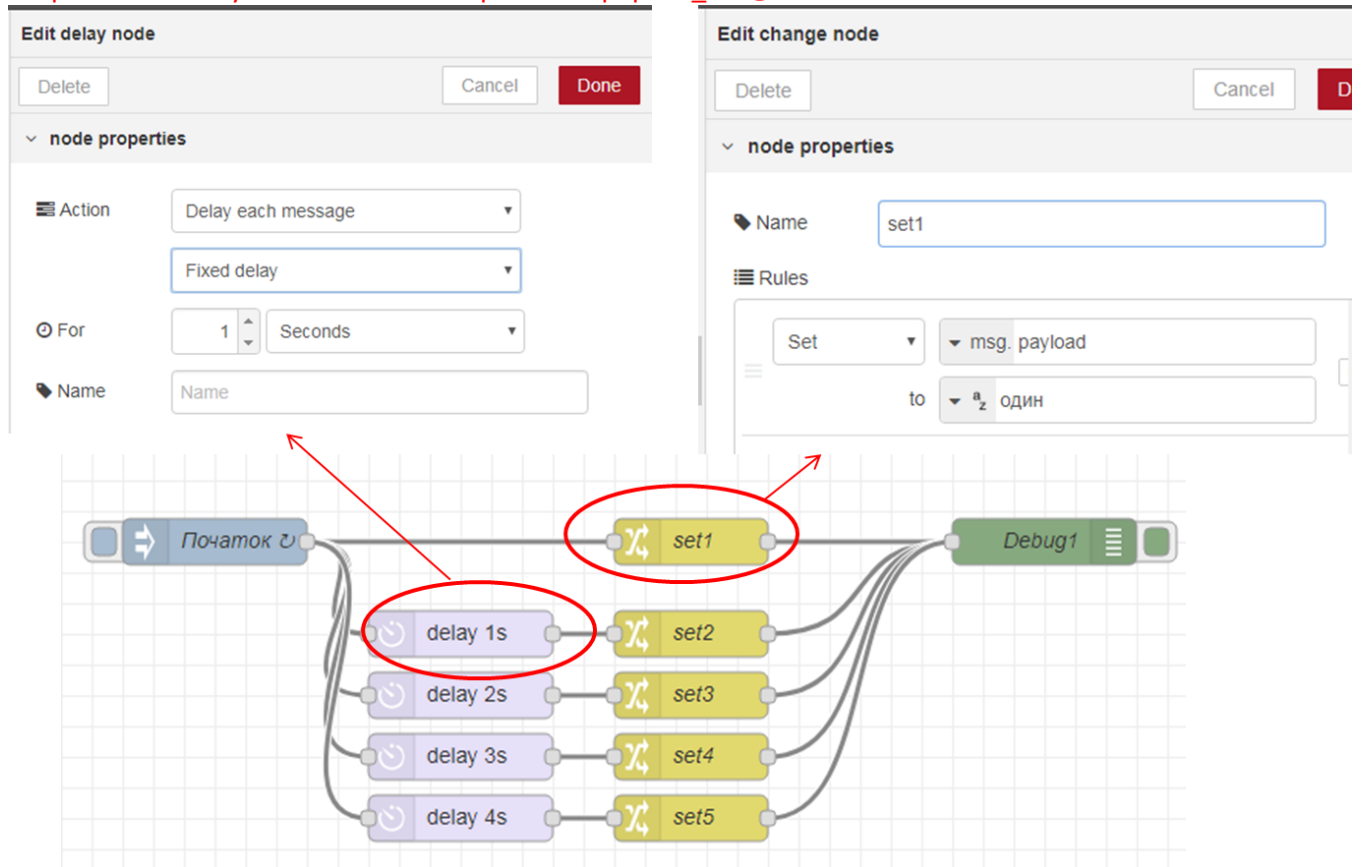


рис.8. Використання вузлів типу change та delay

Розгорніть (deploy) програму та проконтролюйте щоб кожної секунди у вікні повідомлень виводилося конкретне повідомлення від «один» до «п'ять».

## 2.8. Ознайомтеся з роботою вузлів типу function з інструкції користувача

Вузол function може обробляти повідомлення з використанням javascript. Змініть програму так, щоб відмітка часу виводилася в форматі дати та часу. Для цього використовується об'єкт типу Data та його метод toLocaleString(). Зробіть розгортання та переведіть вузол "Debug1" в режим приховання повідомлень.

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

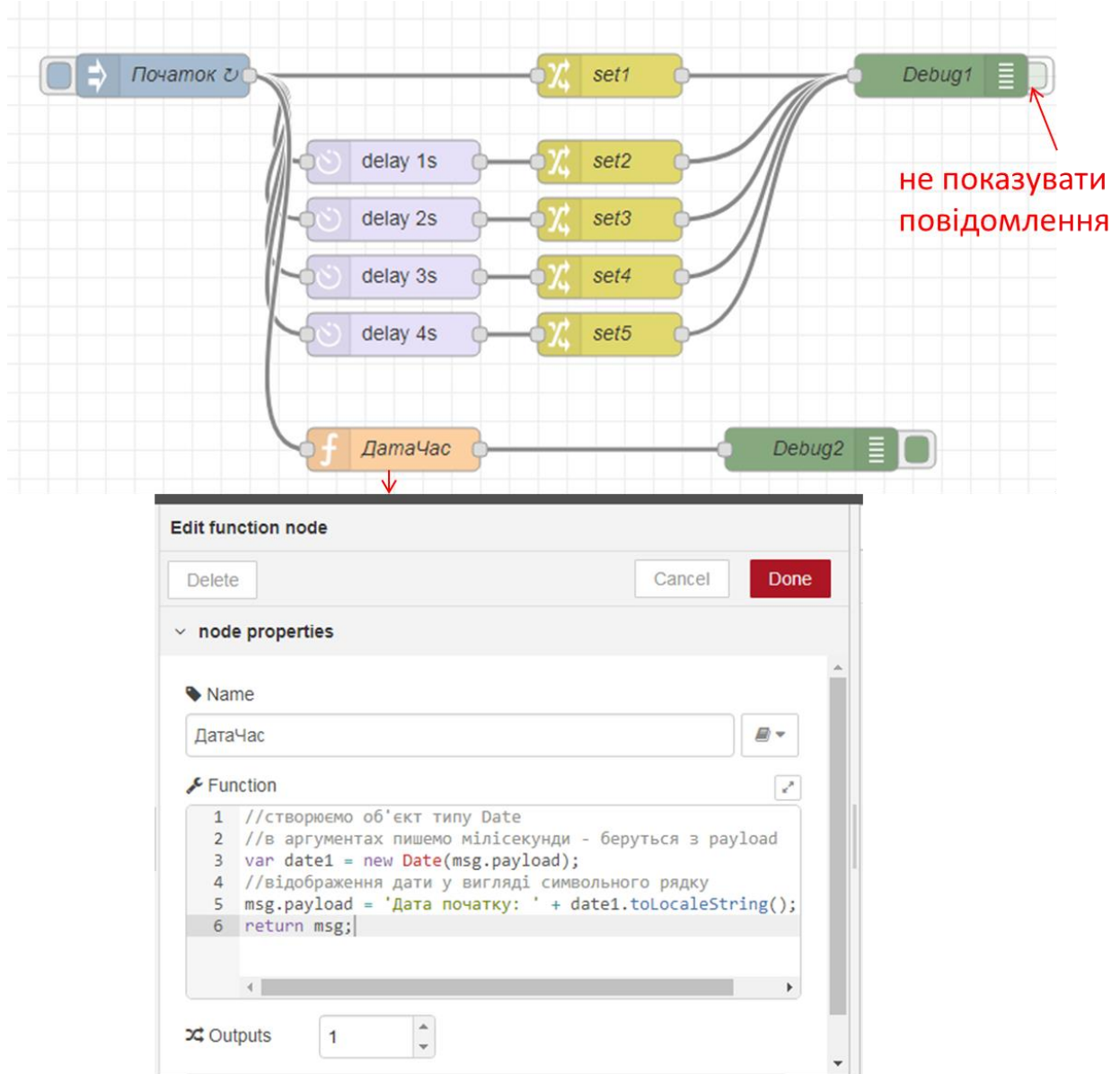


рис.9. Змінена програма з вузлом function

Про синтаксис javascript можна дізнатися тут <http://яваскрипт.укр>  
Про об'єкт Date можна прочитати тут <http://яваскрипт.укр/Date> або тут <https://learn.javascript.ru/datetime>

### 3. Підключення та ознайомлення з модулем node-red-dashboard

Node-RED дозволяє інстальовати та обновляти палітру вузлів. Це робиться через Manage Palette (рис.10) . Деталі інсталяції читайте в інструкції користувача.



Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

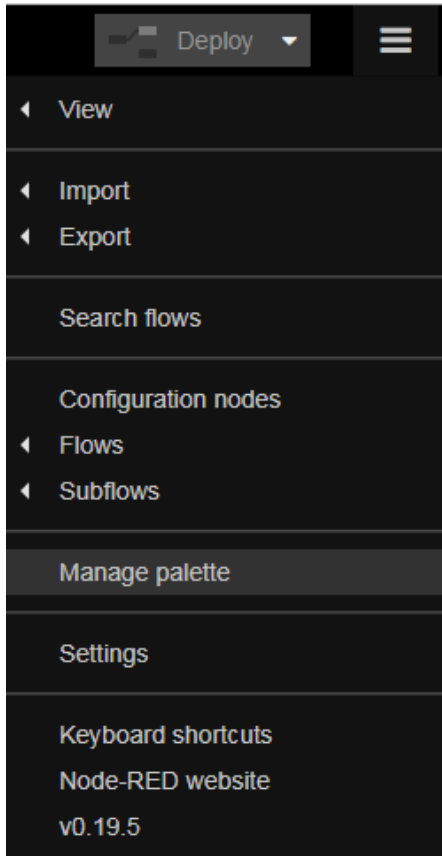


рис.10. Вибір Manage palette

3.1. В налаштуваннях палітри на вкладці Install в поле фільтру введіть “node-red-dashboard” і інстальуйте даний пакет (рис.11):

- натисніть кнопку install
- підтвердіть інсталяцію у вікні повідомлення
- після інсталяції закрийте вікно керування палітрою

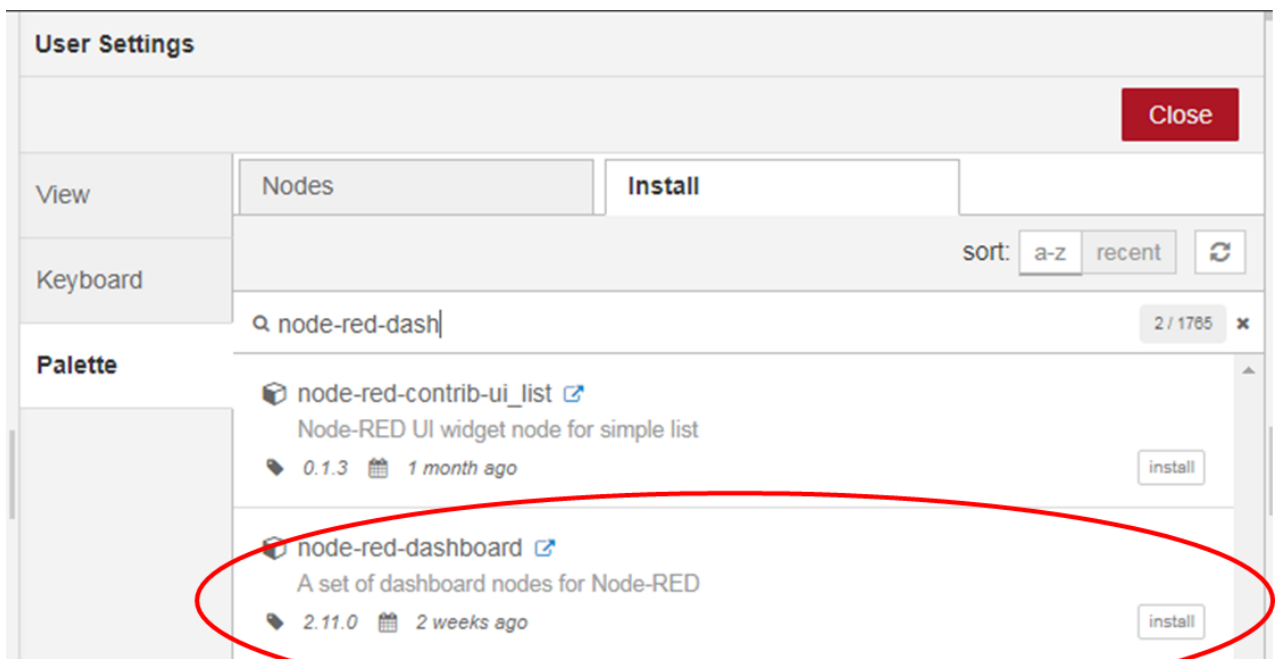


рис.11. Вибір Manage palette

Перевірте чи з'явилася в палітрі розділ «Dashboard»

10.12.2018

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

3.2. Після встановлення у бічній панелі з'явилася нова іконка з зображенням діаграми (рис.12) . Натисніть на ній.

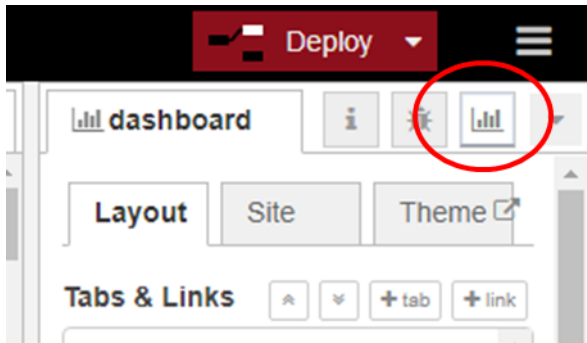


рис.12.

В Layout додайте дві закладки (tab) та змініть їх назви як це показано на рис.13.

**Однак ім'я першої закладки повинно називатися Вашим прізвищем та ім'ям, наприклад «Іваненко Іван».**

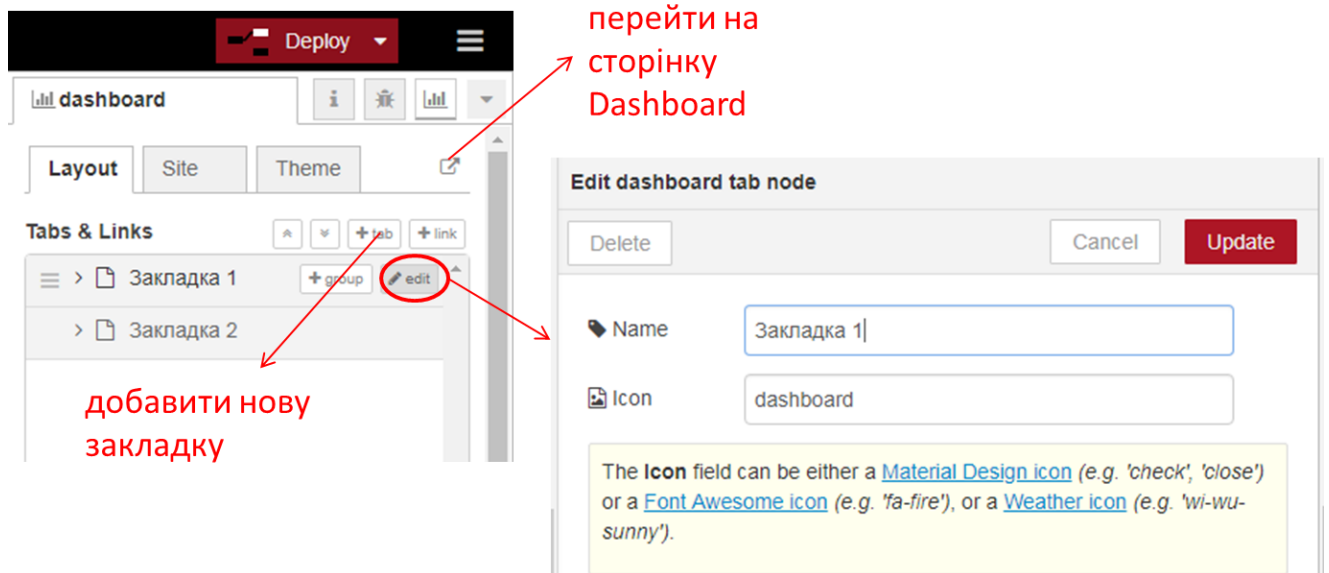


рис.13.

3.3. Модифікуйте програму, створивши вузол типу dashboard-> text і підєднавши його до вузла «ДатаЧас» (див. рис.14). Налаштуйте вузол відповідно до рис.14 **однак ім'я першої групи повинно називатися Вашим прізвищем «Іваненко».** Створіть ще одну групу, яка повинна називатися Вашим ім'ям «наприклад Іван».

Після усіх налаштувань зробіть розгортання, відкрийте створений Dashboard, шляхом натискання кнопки переходу (див.рис.13.), або ввівши в новій вкладці браузера <http://127.0.0.1:1880/ui>

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

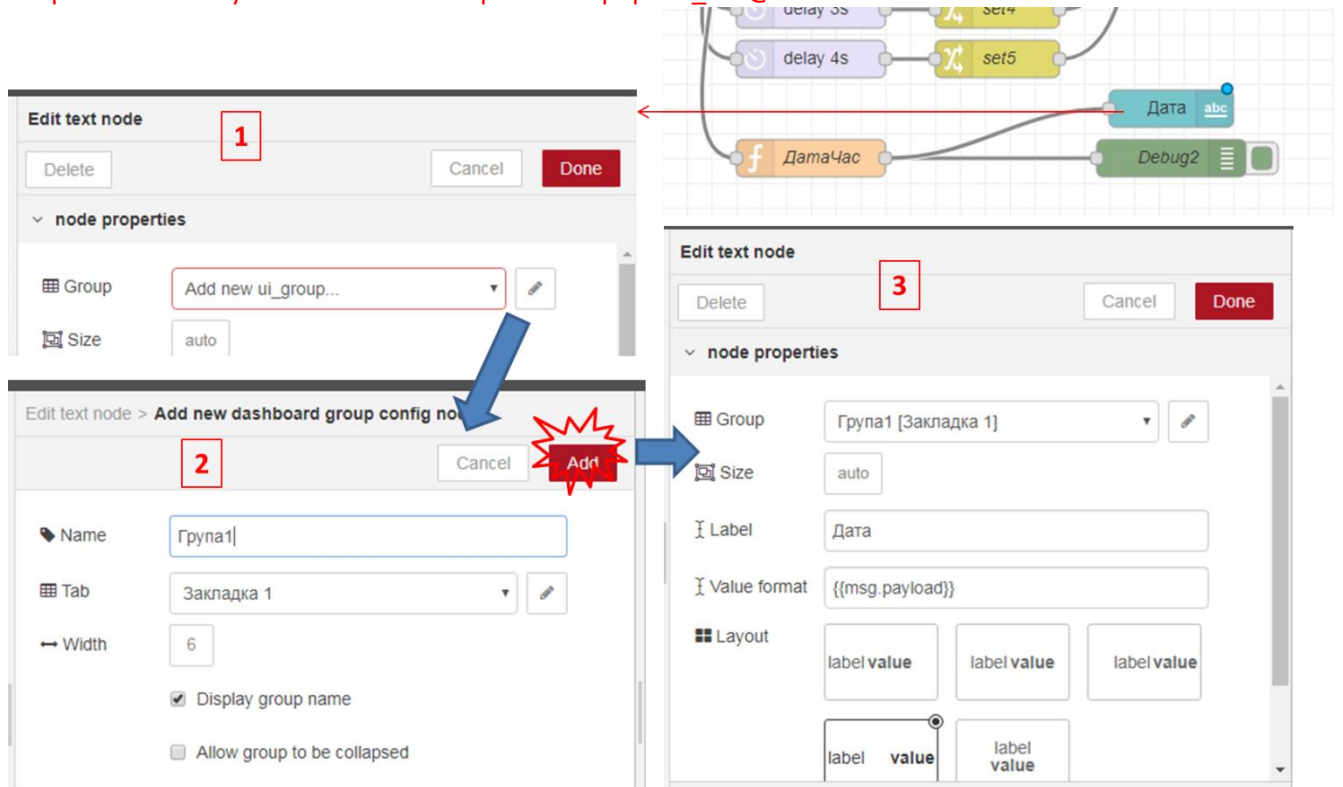


рис.14.

На вкладці повинно з'явитися щось типу такого, як показано на рис.15

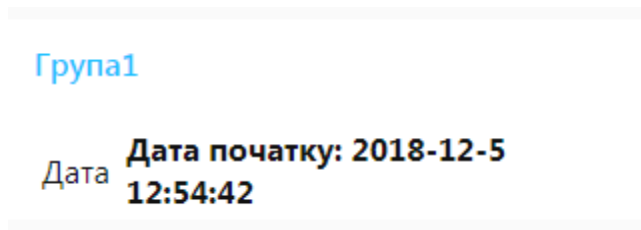


рис.15

3.4. Аналогічним чином зробіть для відображення числа прописом.

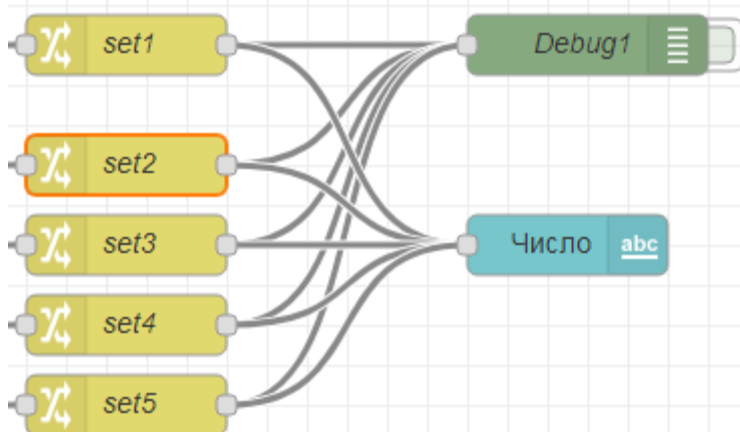


рис.16

3.5. Додайте до програми фрагмент, як показано на рис.17.

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

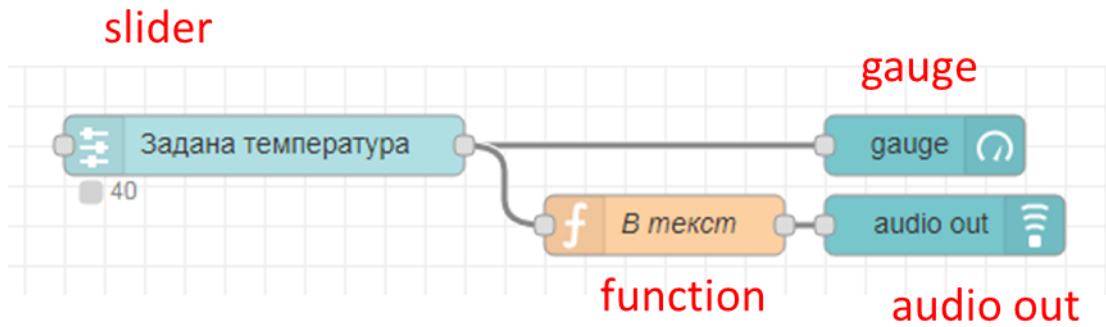


рис.17

Налаштування вузлів показані на рис.18-21

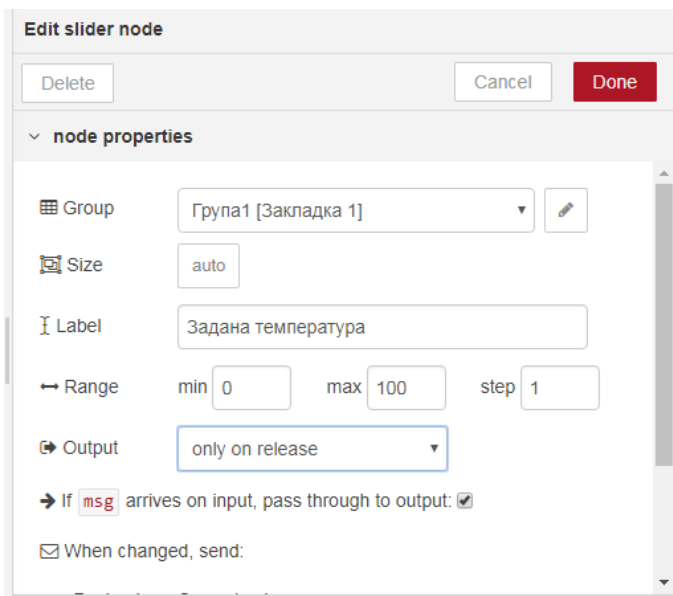


рис.18

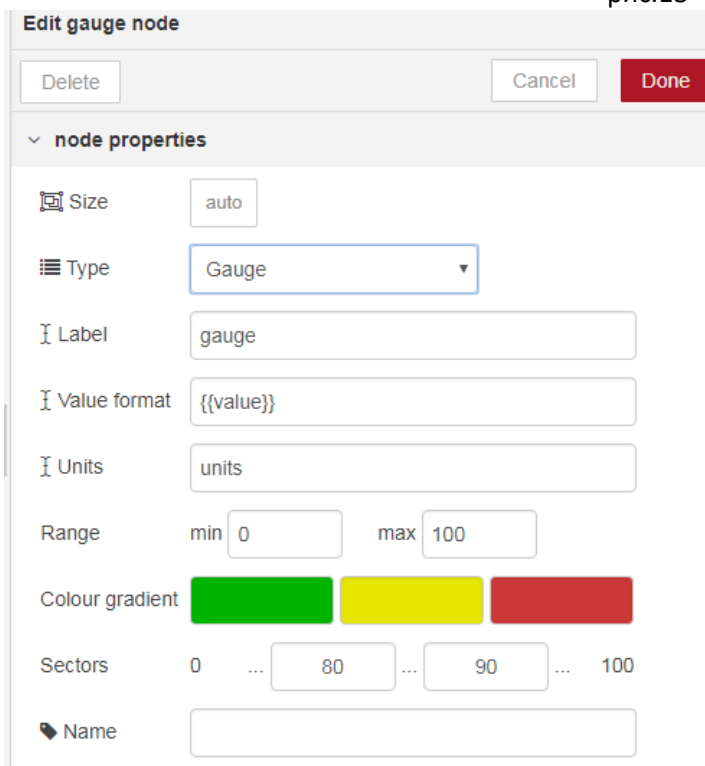


рис.19

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

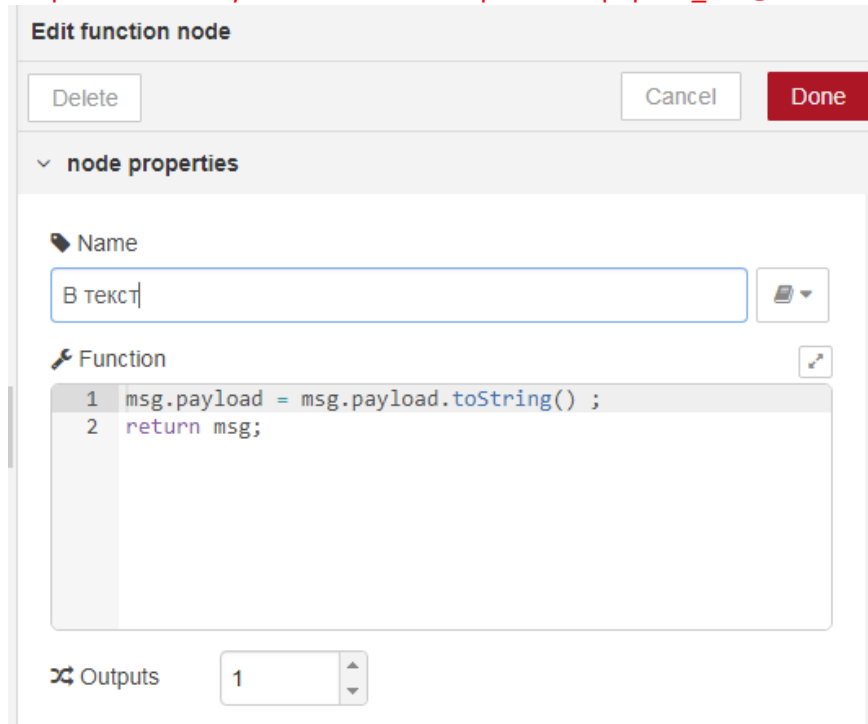


рис.20

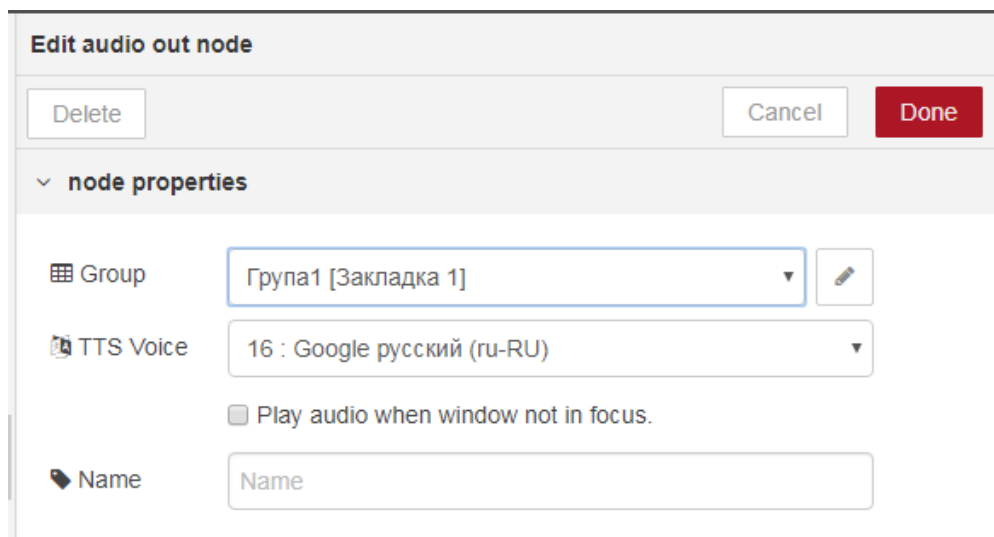


рис.21

Зробіть розгортання і подивіться результат на вікні Dashboard.

3.6. Ознайомтеся з принципами роботи вузла **switch**. Модифікуйте програму відповідно до наведеної на рис.22. Вузли налаштуйте відповідно до рис.23-26.

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

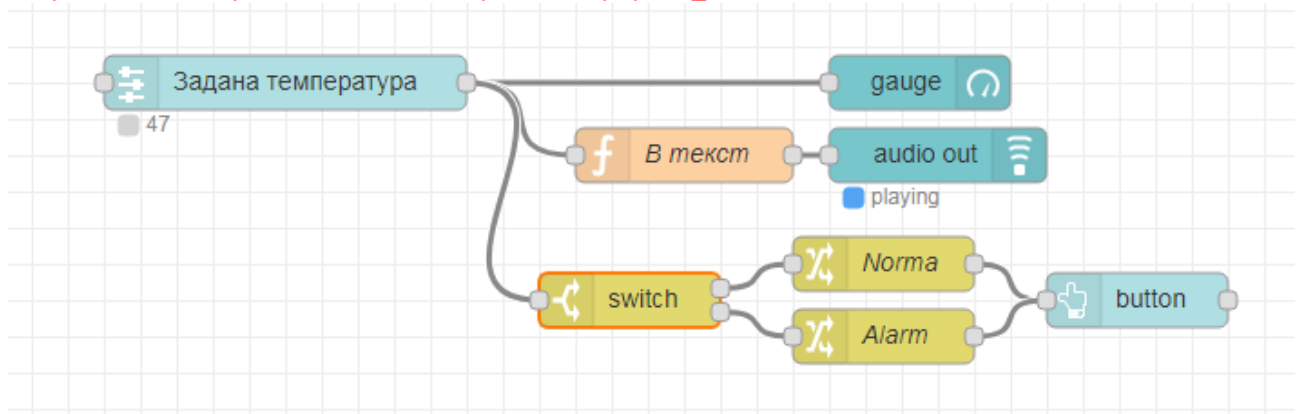


рис.22. Фрагмент модифікованої програми

**Edit switch node**

Delete Cancel Done

node properties

Name

Property

is between  → 1

and

>=  → 2

рис.23. Налаштування вузла switch

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

The screenshot shows a dialog box titled "Edit change node" with a "Delete" button on the left, "Cancel" and "Done" buttons on the right. Under the "node properties" section, the "Name" field contains "Norma". The "Rules" section contains two entries: 1) "Set" action on "msg. payload" to "HOPMA", and 2) "Set" action on "msg. color" to "GREEN". Each rule has a small "x" icon to its right for removal.

рис.24. Налаштування вузла норма

The screenshot shows a dialog box titled "Edit change node" with a "Delete" button on the left, "Cancel" and "Done" buttons on the right. Under the "node properties" section, the "Name" field contains "Alarm". The "Rules" section contains two entries: 1) "Set" action on "msg. payload" to "АВАРІЯ", and 2) "Set" action on "msg. color" to "RED". Each rule has a small "x" icon to its right for removal.

рис.25. Налаштування вузла Alarm

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

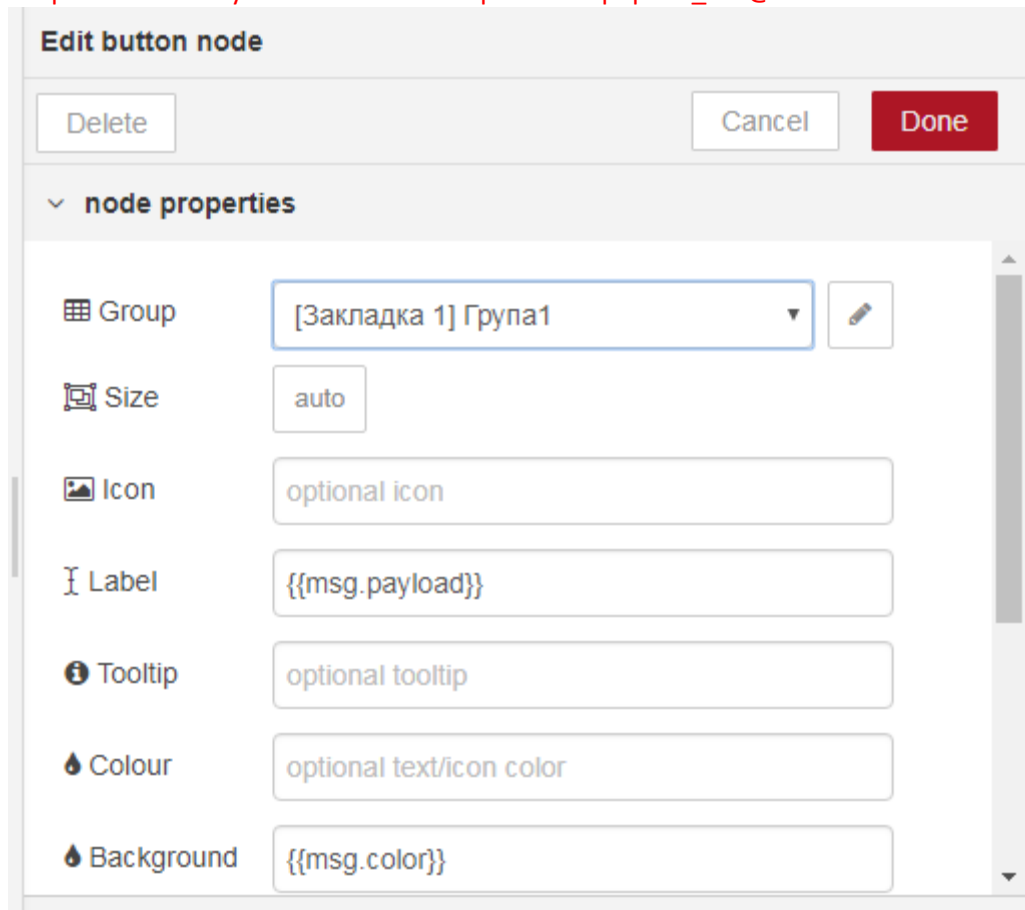


рис.26. Налаштування вузла Button

Зробіть розгортання проекту, перевірте як працює програма. Для цього на сторінці веб-інтерфейсу змініть значення заданої температури в діапазоні 0-50, а потім >50.

Ця частина програми працює наступним чином. При зміні значення температури, в msg.payload значення поступає на обробку в вузол switch, де на один із 2-х виходів формується повідомлення в залежності від тієї умови, яка спрацювала.

При виконанні умови  $0 < \text{msg.payload} < 50$  (is between), повідомлення передається на перший вихід, до якого в свою чергу приєднаний вузол "Norma" (тип function->change). Той задає текстове значення для властивості msg.payload рівним «НОРМА» і формує нову властивість msg.color рівною "GREEN". Далі msg поступає вузол «button», який використовується для відображення тексту в прямокутнику. Значення тексту задається полі Label, а колір в полі Background. При формуванні динамічних значень для вузлів, використовується формат angular фільтрів, в якому вказується підстановка в подвійних фігурних дужках.

Аналогічна обробка проводиться при спрацюванні у вузлі switch умови  $\text{msg.payload} > 50$ . Повідомлення згенерується на другому виході, який активує перерахунок вузла "Alarm" що буде формувати текст та колір для кнопки.

Детальніше про використання фільтрів ангуляр <https://metanit.com/web/angular/2.4.php>  
<http://javastudy.ru/angularjs/angularjs-filters/>

#### 4. Робота з поштою (не обов'язкова частина)

4.1. Для роботи з поштою необхідно дізнатися налаштування поштового сервера, який Ви будете використовувати для відправки повідомлень, і на якому є Ваш обліковий запис. **Ми рекомендуємо для тестування створити новий аккаунт на одному з поштових серверів.**

Випишіть налаштування:



Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

Для вихідних повідомлень (відправки пошти):

- SMTP Server
- port

Для вхідних повідомлень (отримання пошти):

- POP3 Server або IMAP Server
- port

Інформацію про налаштування поштових сервісів можна отримати в довідці по цим серверам.

Нижче наведений приклад деяких із найбільш вживаних:

Приклади:

Поштовий сервіс	Для відправки Сервер вихідних повідомлень		Для отримання Сервер вхідних повідомлень		Примітка
	SMTP Server	smtp.ukr.net	IMAP Server	imap.ukr.net	
www.ukr.net	port	465 або 2525	port	993	В налаштуваннях треба увімкнути IMAP/SMTP (рис.27)
	SMTP Server	smtp.ukr.net	IMAP Server	imap.ukr.net	
www.gmail.com	port	465(SSL) або 587(TLS)	port	993	В налаштуваннях треба увімкнути IMAP (рис.28) Активувати доступ до додатків <a href="https://myaccount.google.com/lesssecureapps">https://myaccount.google.com/lesssecureapps</a> (рис.29)
	SMTP Server	smtp.gmail.com	IMAP Server	imap.gmail.com Вимагає SSL: так	

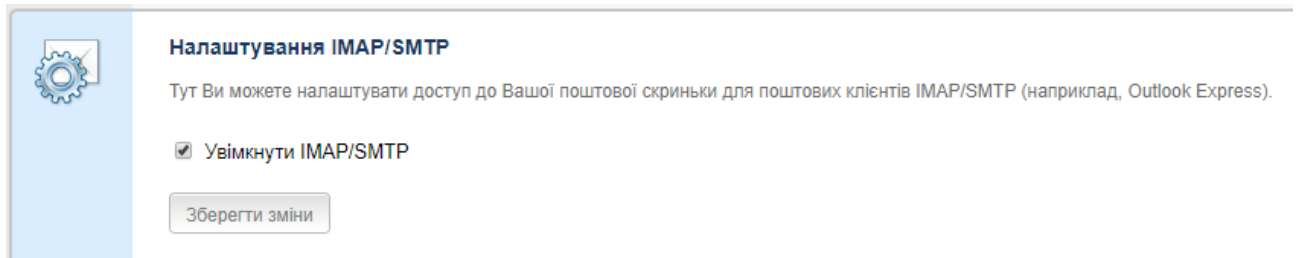


рис.27. Налаштування ukr.net для активації отримання та відправки пошти.

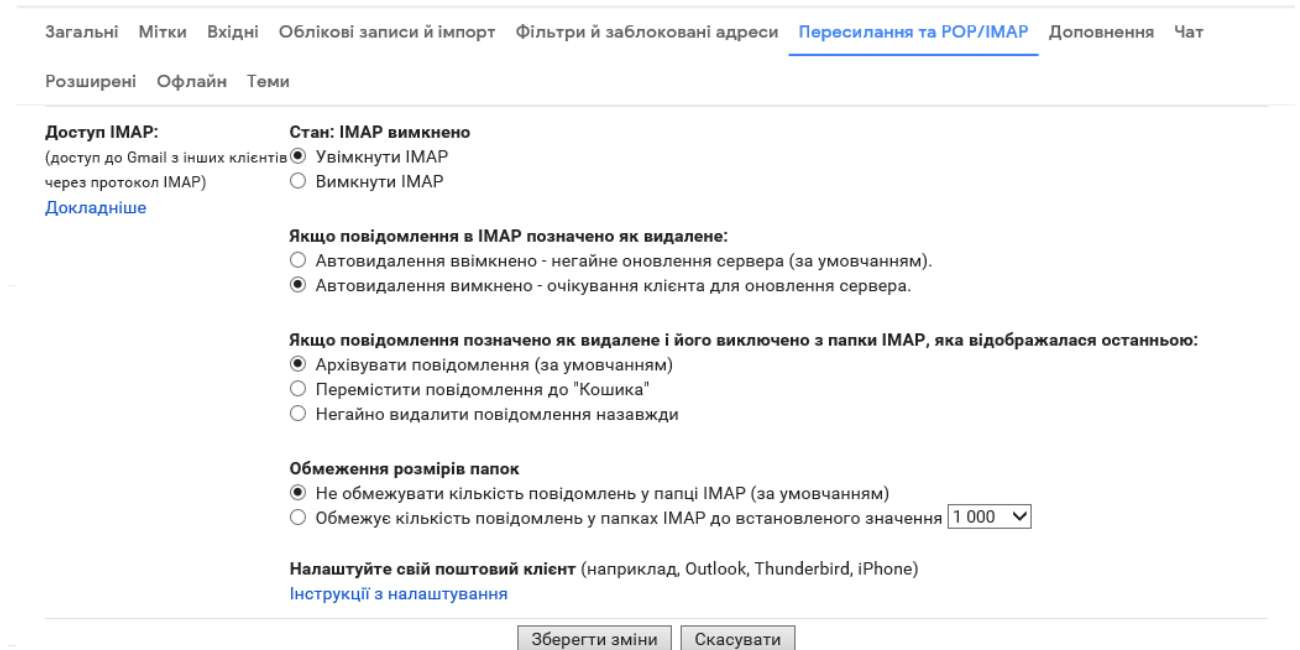


рис.28. Налаштування gmail.com для активації отримання та відправки пошти.

Для сервісів gmail навіть після активації дозволу на взаємодію з поштовими клієнтами через IMAP/SMTP, при використанні додатку NodeRED може прийти повідомлення про блокування доступу (рис.29). Для активації доступу тимчасово (до закінчення лабораторної роботи) виставте дозвіл доступу до менш безпечних додатків (рис.30)

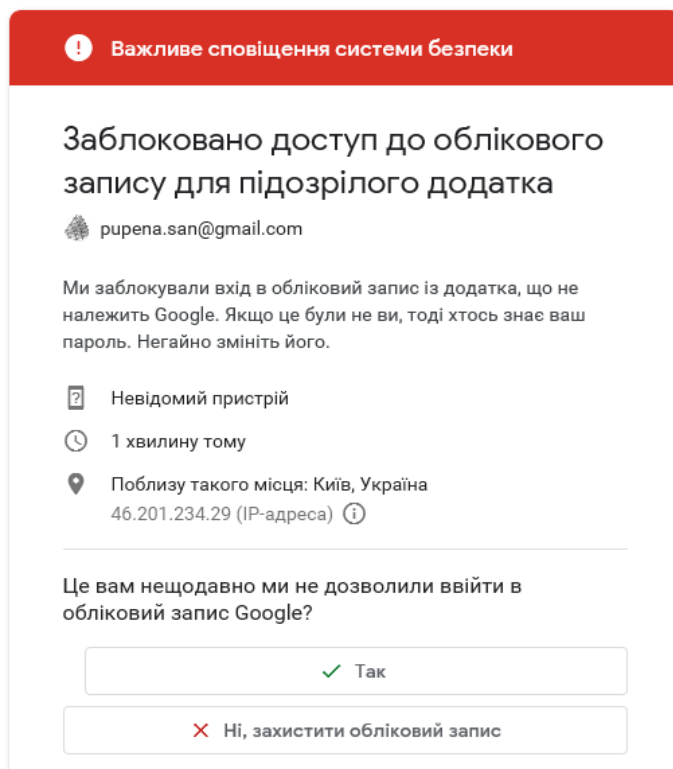


рис.29. Повідомлення gmail.com про блокування доступу.

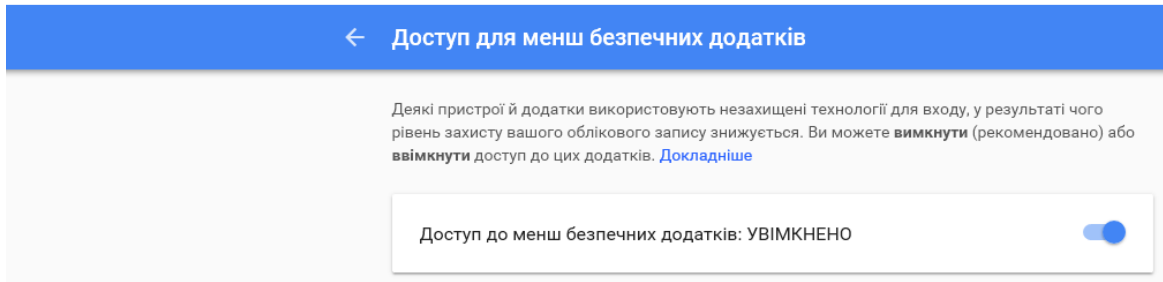


рис.30. Активація доступу до менш безпечних додатків (<https://myaccount.google.com/lessecureapps>).

необхідно також дозволити <https://myaccount.google.com/lessecureapps>

4.2. Модифікуйте програму фрагментом показаним на рис.31. Налаштування вузів показано на рис.32-33. У поля user-id та password необхідно вказати користувача та пароль для акаунта. Інші налаштування беруться з визначених в попередньому пункті. В поле отримувача введіть власну поштову скриньку.

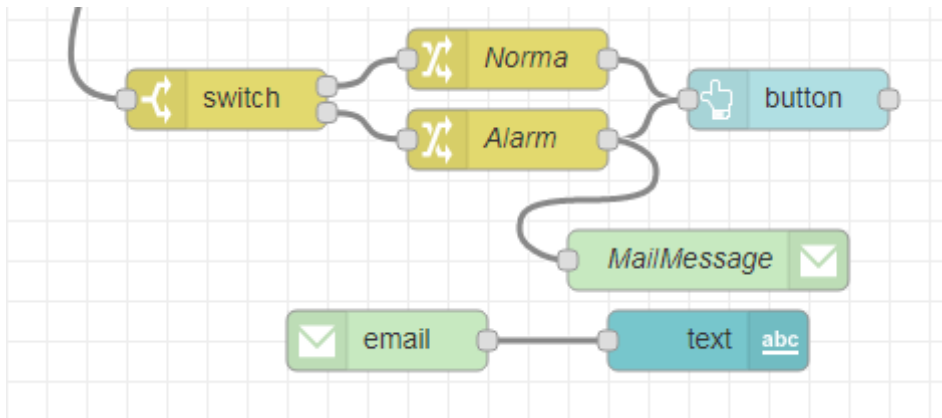


рис.31. Модифікований фрагмент програми

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

**Edit email node**

Delete Cancel Done

node properties

To [redacted]@ukr.net

Server smtp.ukr.net

Port 465  Use secure connection.

Userid [redacted]@ukr.net

Password .....

Name Name

node settings

рис.32. Налаштування вузла MailMessage типу Social e-Mail

**Edit email node**

Delete Cancel Done

node properties

Refresh 15 seconds

Protocol IMAP

Use SSL?

Server imap.ukr.net

Port 993

Userid [redacted]

Password .....

рис.33. Налаштування вузла MailMessage типу Social e-Mail-in

Зробіть розгортання проекту, виставте значення заданої температури так, щоб згенерувалося повідомлення Alram. На пошту повинно прийти повідомлення, протягом 15 секунд воно повинно відобразитися на веб-сторінці в полі текст (рис.34).

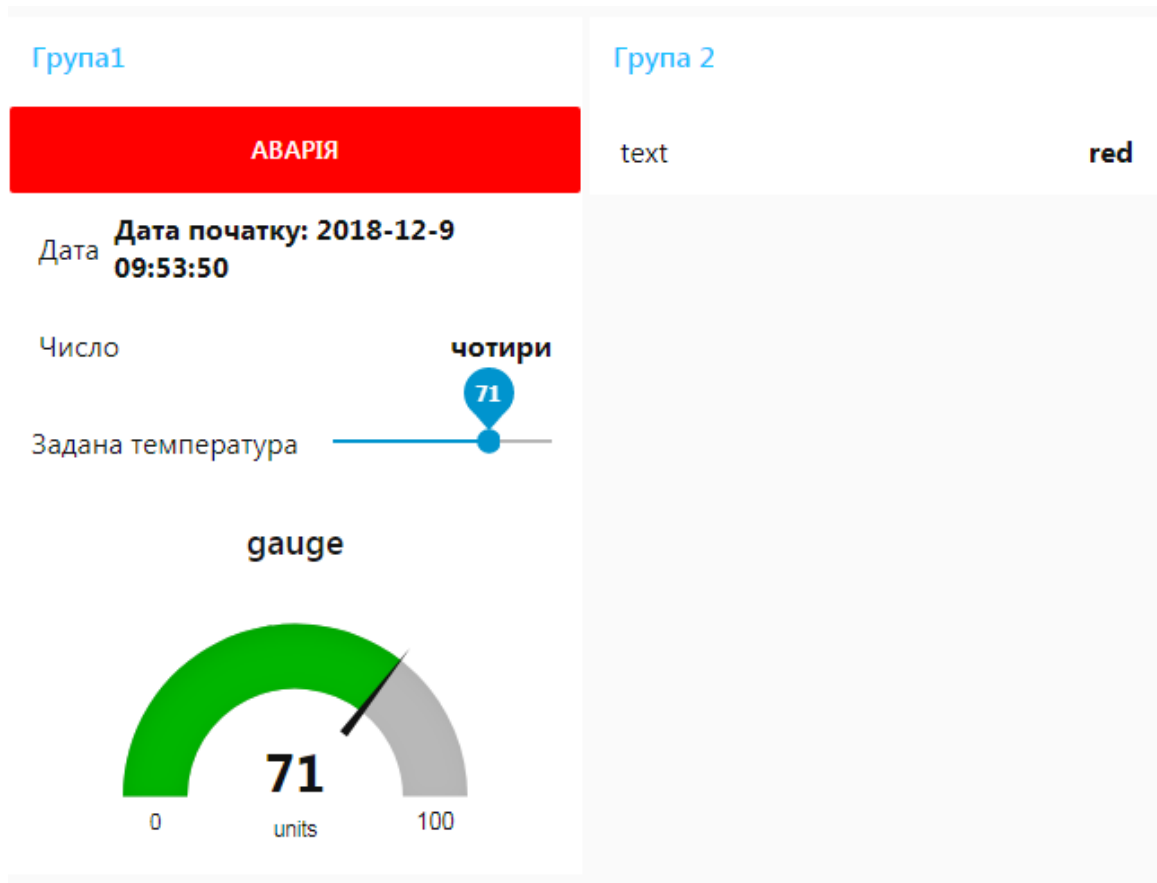


рис.34. Приклад зовнішнього вигляду UI.

## 5. Робота з Modbus

Node-RED, як правило використовується або на стороні Edge або в якості хмарного додатку. Якщо Node-RED використовується на стороні Edge в якості програми для концентратора або шлюзу чи маршрутизатору, наприклад на апаратній платформі Raspberry PI, він може збирати дані з різних пристроїв по протоколам промислових мереж. Найбільш поширеним і простим протоколом на сьогоднішній день є Modbus, тому в спільноті Node-RED розробили кілька бібліотек з ним. Таким чином, як варіант Raspberry PI буде взаємодіяти з пристроями по протоколу Modbus TCP/IP, а з іншого боку він буде взаємодіяти з хмарними додатками та сервісами(рис.35)

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

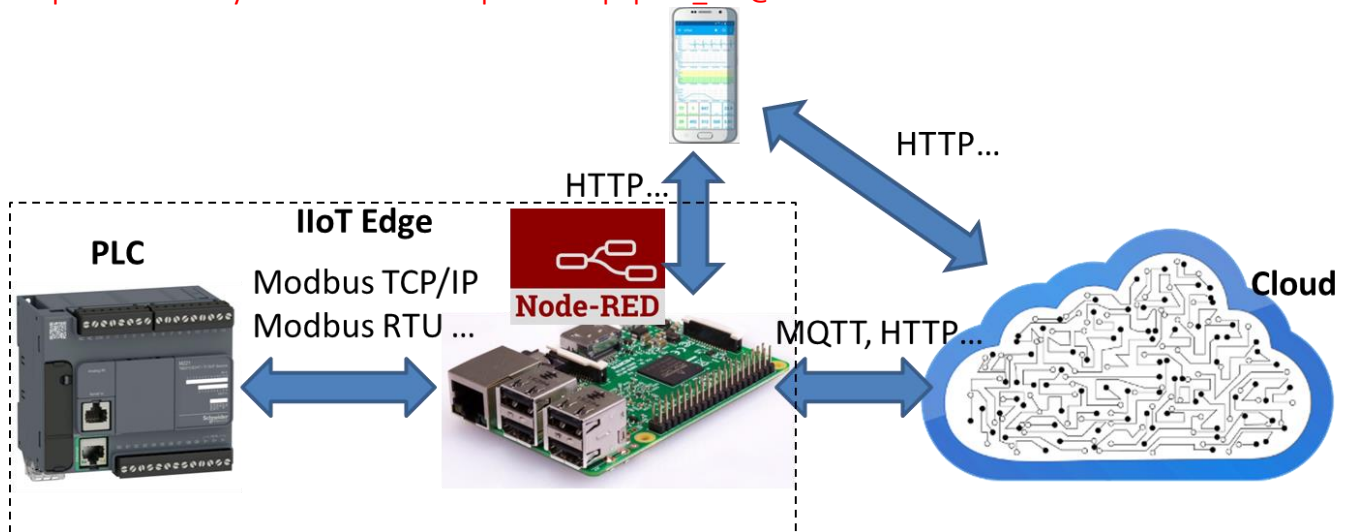


рис.35. Приклад структури рішення IIoT з використанням Raspberry PI та Node-RED.

Для тестування такого рішення можна на перших порах обійтися тільки програмними складовими. Замість ПЛК можна використати імітатор ПЛК, що підтримує Modbus TCP/IP, а замість Raspberry PI – віртуальну машину з ОС Raspbian з усім встановленим ПО, як в реальному залізі. Ще простіше – використовувати тільки Node-RED, який буде з'єднуватися з імітатором ПЛК або імітатором Modbus TCP/IP Server (рис.36). У даній лабораторній роботі в якості імітатора PLC буде використаний пакет Mod\_RSsim, який треба попередньо встановити. А в якості бібліотеки Modbus для Node-RED - node-red-contrib-modbus tcp.

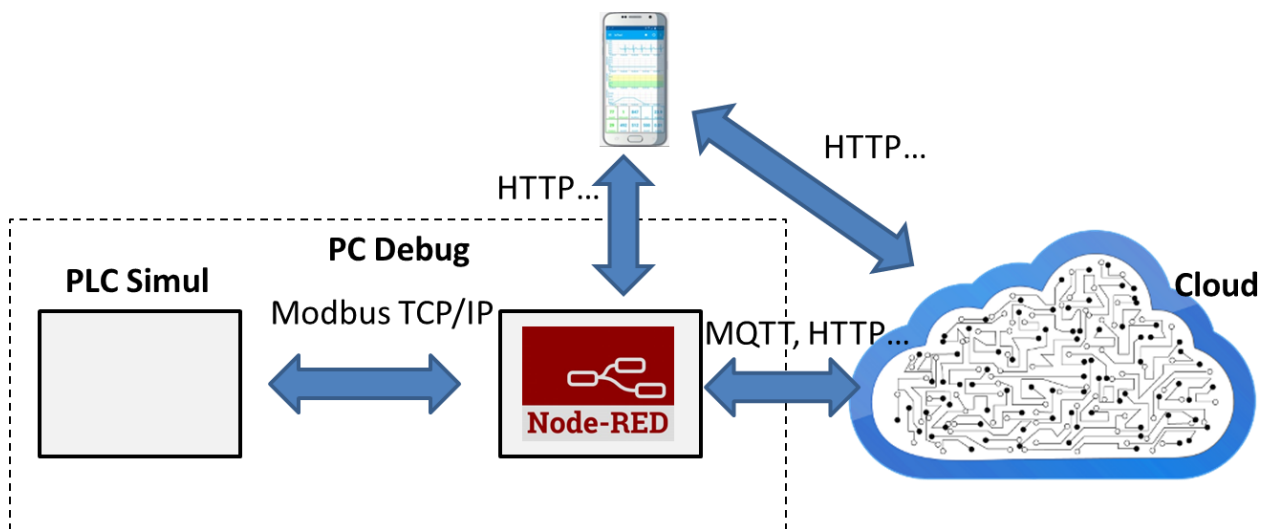


рис.36. Приклад структури рішення IIoT з використанням Raspberry PI та Node-RED.

5.1. Встановити пакет Modbus (node-red-contrib-modbus tcp) використовуючи Manage Palette (рис.37)

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

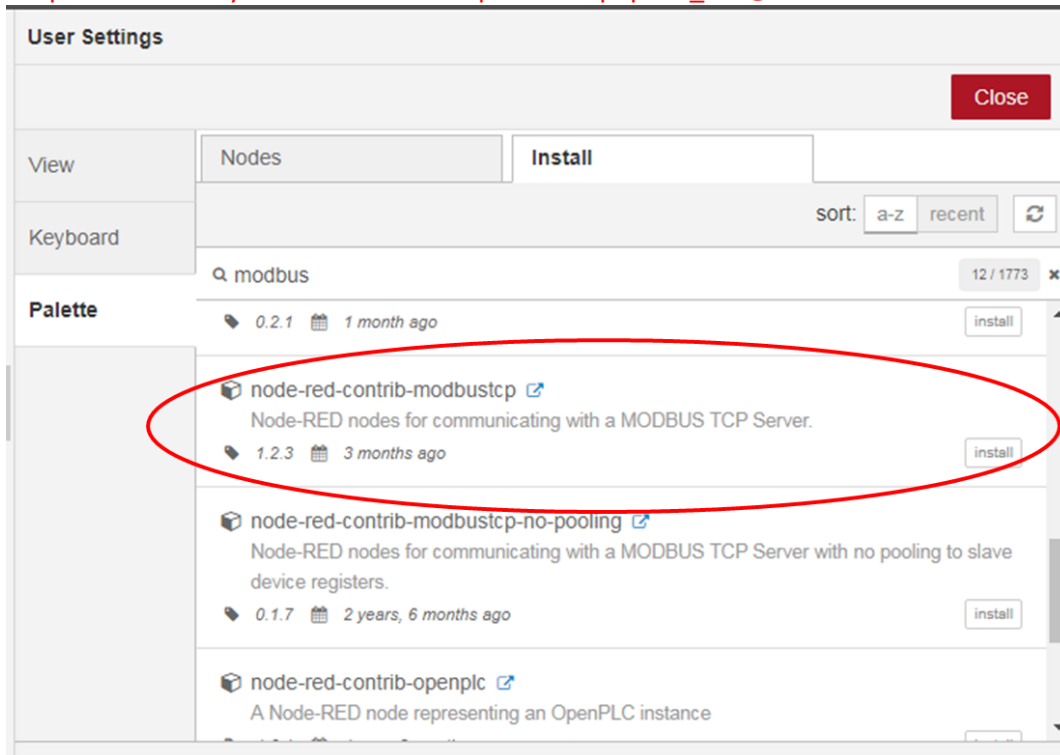


рис.37. Встановлення пакету node-red-contrib-modbustcp.

5.2. Завантажте Modbus PLC Simulator (Mod\_RSsim) з <http://www.plcsimulator.org/>

- завантажте та встановіть дистрибутив <http://www.plcsimulator.org/downloads/SimSetup.msi?attredirects=0>
- завантажте файл для правки реєстру з ключем [http://www.plcsimulator.org/downloads/Vista\\_key.reg?attredirects=0](http://www.plcsimulator.org/downloads/Vista_key.reg?attredirects=0) та запустіть його на виконання

5.3. Запустіть на виконання Modbus PLC Simulator C:\Program Files (x86)\EmbeddedIntelligence\Mod\_RSsim  
Виставте значення в Prot: Modbus TCP

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

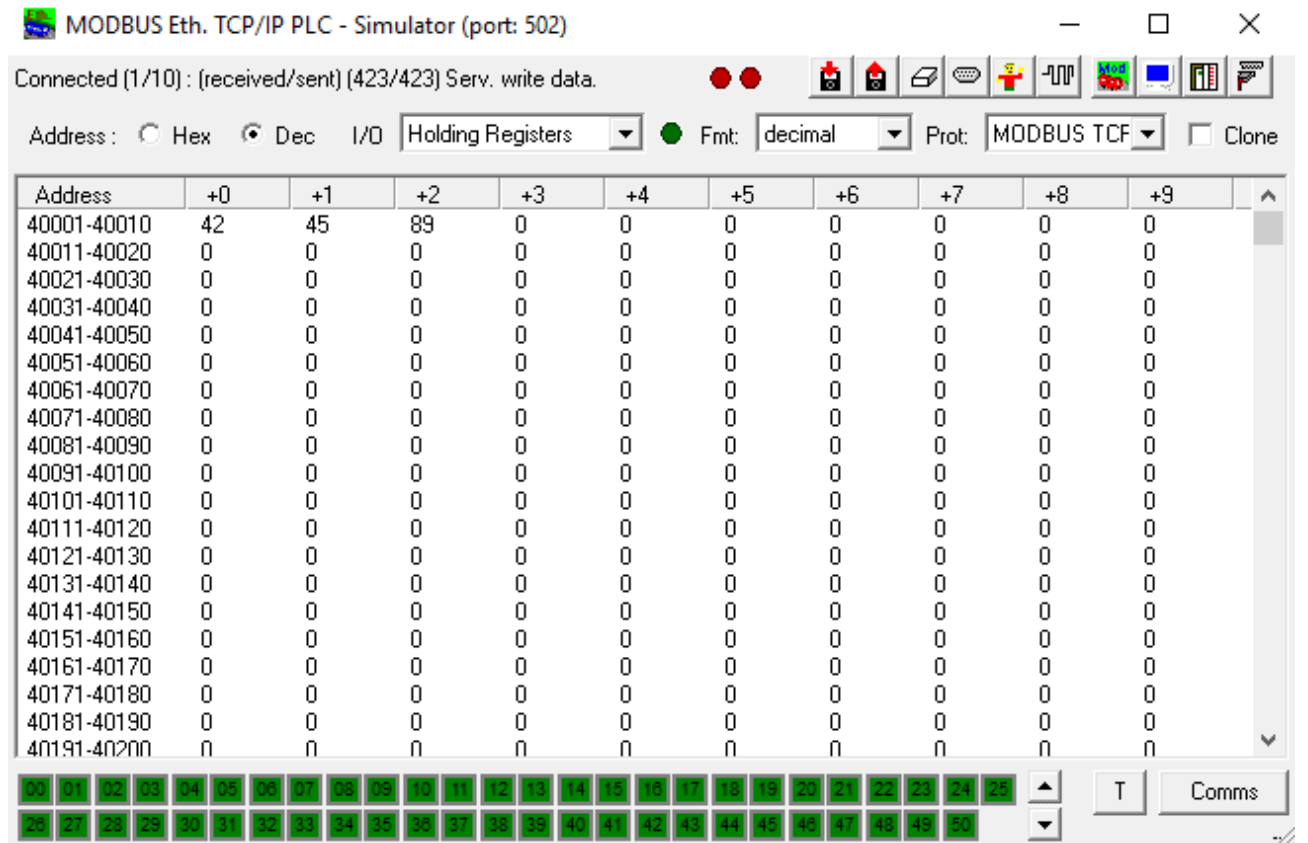


рис.38 Налаштування Mod\_RSsim.

5.4. Ознайомтеся з правилами роботи з бібліотекою <https://flows.nodered.org/node/node-red-contrib-modbustcp>

5.5. З розділу палітри Inputs вставте елемент modbustcp-read, зайдіть в налаштування. Праворуч поля Server натисніть кнопку з олівцем для створення нового серверу (рис.39).



Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

Server Add new modbustcp-server...

IEEE-754

Edit modbustcp node > Add new modbustcp-server config node

Cancel Add

Name PLCSIM

Host 127.0.0.1

Port 502

Unit Id 1

Reconnect Interval (s)

10

рис.39 Створення серверу Modbus TCP/IP.

Після створення серверу налаштуйте зчитування десяти Holding реєстрів починаючи з 0-го, як це показано на рис.40.

Edit modbustcp node

Delete Cancel Done

node properties

Name Читання реєстрів

Topic topic

FC FC 3: Read Holding Registers

Address 0

Quantity 10

Poll Rate 5 second(s)

Server PLCSIM

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

рис.40 Налаштування зчитування по Modbus TCP/IP.

Зробіть фрагмент програми, показаний на рис.41. Зробіть розгортання проекту, деактивуйте усі виводи debug окрім останнього.

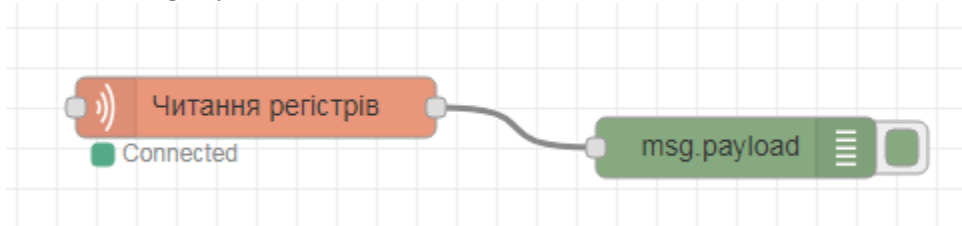


рис.41 Фрагмент програми зчитування.

Змініть значення перших десяти реєстрів у програмі Mod\_RSsim. Активуйте вікно виводу Debug, там повинні виводитися значення реєстрів у вигляді масиву.



рис.42 Значення зчитуваних реєстрів у вигляді масиву.

Зверніть увагу, що тепер msg.payload є масивом з десяти елементів (Array[10]), тому для роботи з цими значеннями, наприклад виводу на відображення на ВЕБ-сторінці необхідно їх попередньо обробити.

5.6. Модифікуйте програму відповідно до рис.43 – 46. Зробіть розгортання та перевірте чи вірно відображаються значення.

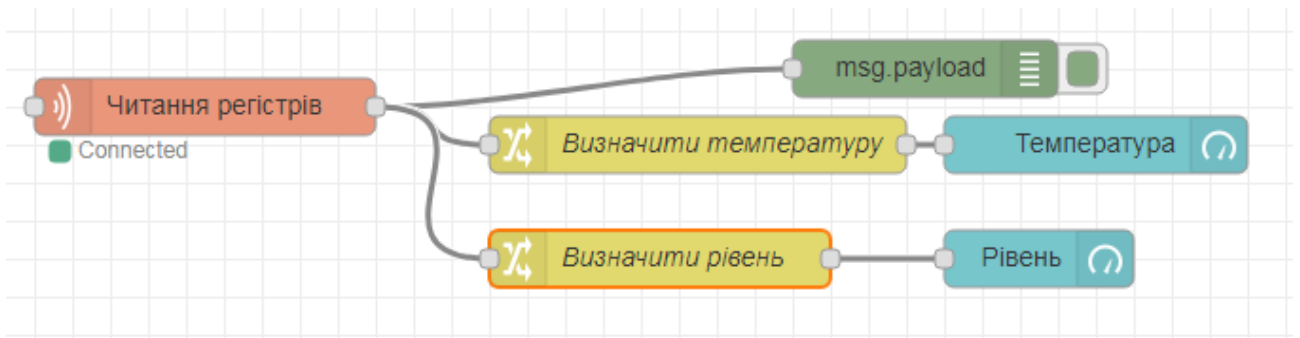


рис.43 Модифікована програма .

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

The screenshot shows the 'Edit change node' dialog box. At the top, there are three buttons: 'Delete', 'Cancel', and 'Done'. Below this is a section titled 'node properties'. The 'Name' field contains the text 'Визначити температуру'. Under the 'Rules' section, there is a 'Set' dropdown menu, a 'msg. payload' field, and a 'to' field containing 'msg. payload[0]'. There is also a vertical scrollbar on the right side of the rules area.

рис.44 Налаштування вузла «визначити температуру».

The screenshot shows the 'Edit change node' dialog box. At the top, there are three buttons: 'Delete', 'Cancel', and 'Done'. Below this is a section titled 'node properties'. The 'Name' field contains the text 'Визначити рівень'. Under the 'Rules' section, there is a 'Set' dropdown menu, a 'msg. payload' field, and a 'to' field containing 'msg. payload[1]'. There is also a vertical scrollbar on the right side of the rules area.

рис.45 Налаштування вузла «визначити рівень».

The screenshot shows the 'Edit gauge node' dialog box. At the top, there are three buttons: 'Delete', 'Cancel', and 'Done'. Below this is a section titled 'node properties'. The 'Group' field is a dropdown menu with the text '[Закладка 1] Група 2'. The 'Size' field contains '3 x 3'. The 'Type' dropdown menu is set to 'Donut'. The 'Label' field contains 'Температура'. The 'Value format' field contains '{{value}}'. The 'Units' field contains '°C'. The 'Range' section has 'min' set to '0' and 'max' set to '100'. There is a vertical scrollbar on the right side of the properties area.

рис.46 Налаштування вузла «температура».

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. pupena\_san@ukr.net

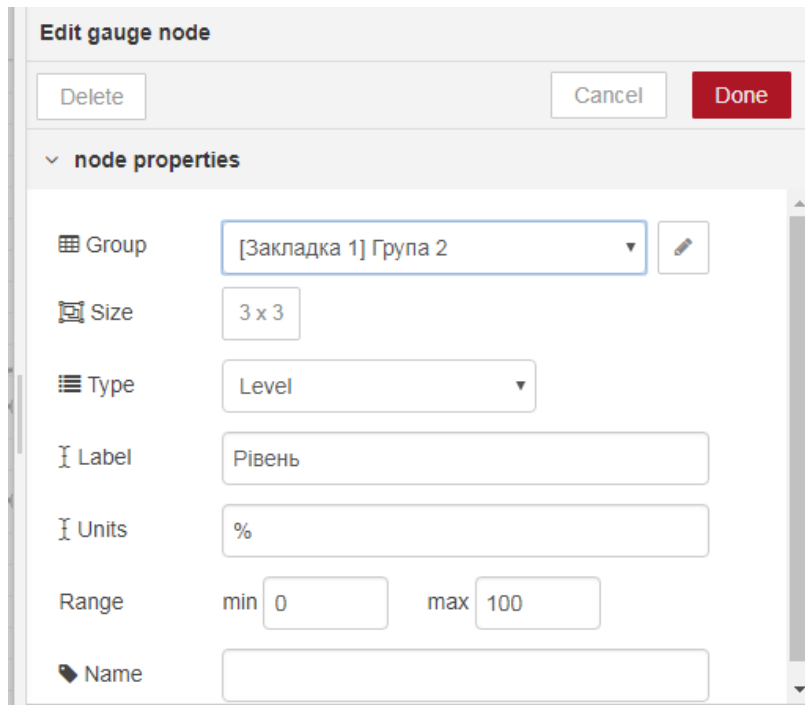


рис.46-1 Налаштування вузла «рівень».

5.7. Для запису по Modbus використайте вузол modbustcp-write з розділу палітри outputs. Модифікуйте програму відповідно до рис.47. Зробіть розгортання проекту і перевірте чи змінюється значення Holding реєстру в Mod\_RSsim при зміні його через елемент «Задана температура»

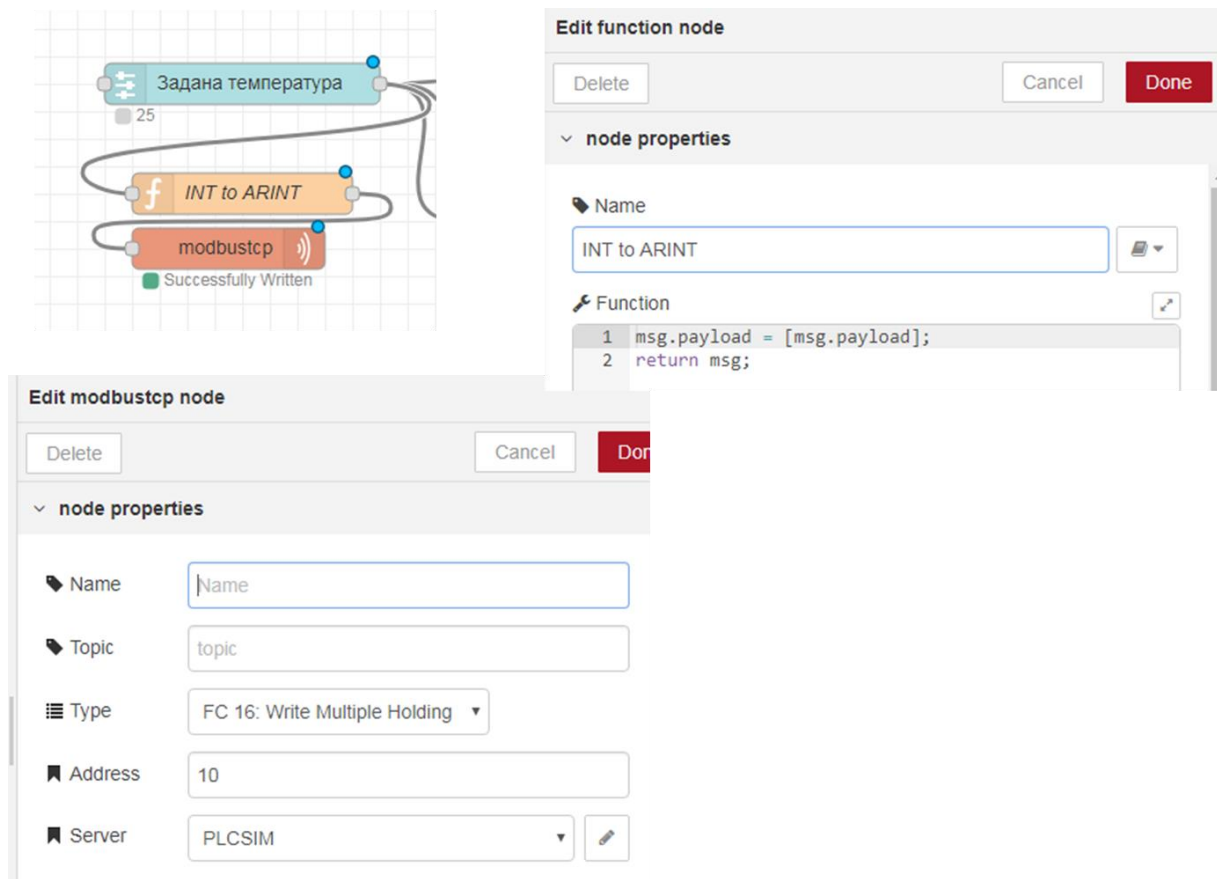


рис.47 Фрагмент програми та налаштування запису по Modbus.

## 5.8. Необов'язкове завдання.

Спробуйте в якості імітатора ПЛК використати:

- імітатор Unity PRO з демо-проектом, наприклад з лабораторних робіт по ЛМІ <https://drive.google.com/open?id=0B2FfwwwweBSVRWw4eDVreE1Lb1k>
- імітатор Unity PRO або M221 з проектом керування роботизованою установкою <http://www.iasu-nuft.pp.ua/virtualnij-trenazer-projectprogrammer>

## 6. Робота з JS об'єктами та обробка системної інформації

### 6.1. Встановіть в Node-RED модуль node-red-contrib-os

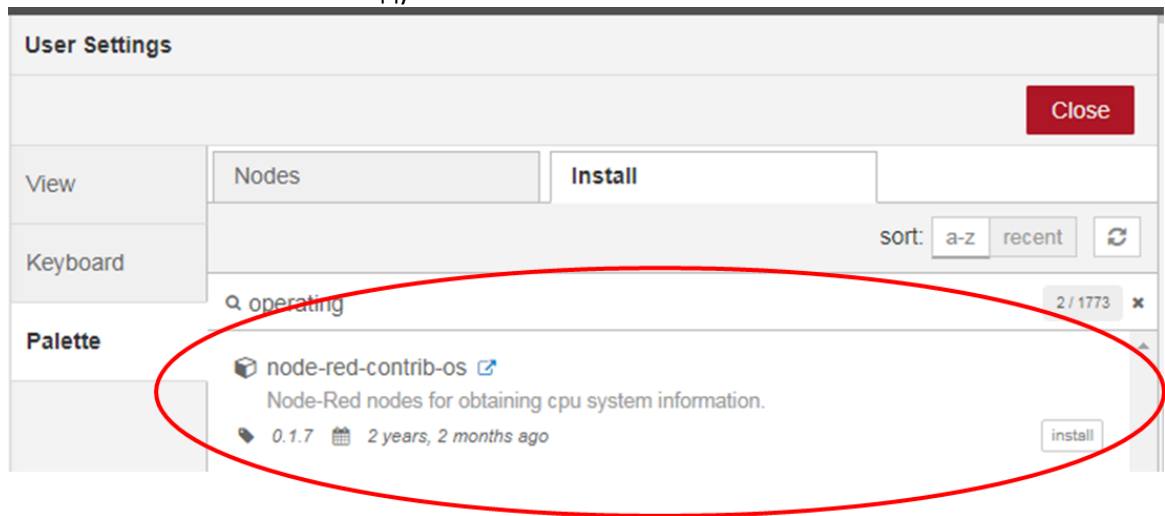


рис.48 Встановлення модуля node-red-contrib-os.

6.2. З нововстановленого модуля використайте вузол типу NetworkIntf для створення фрагменту програми, як на рис.49.

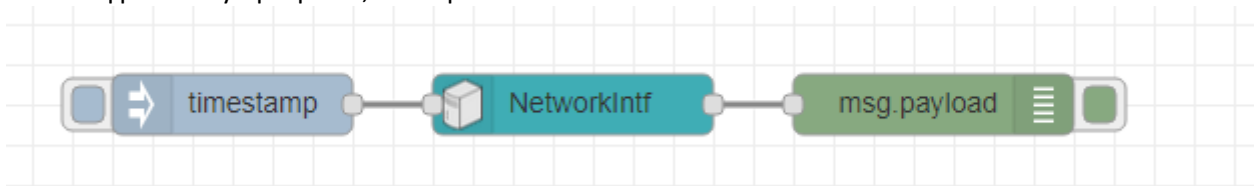


рис.49 Фрагмент програми для виведення інформації по наявним мережним картам

Зробіть розгортання програми, ініціюйте формування повідомлення, проаналізуйте вивід. Приклад виведеної інформації показаний на рис.50. Як видно, інформація надається у вигляді JS об'єкту, який включає в себе об'єкт NetworkInterfaces, що в свою чергу вміщує кілька мережних інтерфейсів, що є масивами об'єктів, що представляють певний протокол..

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

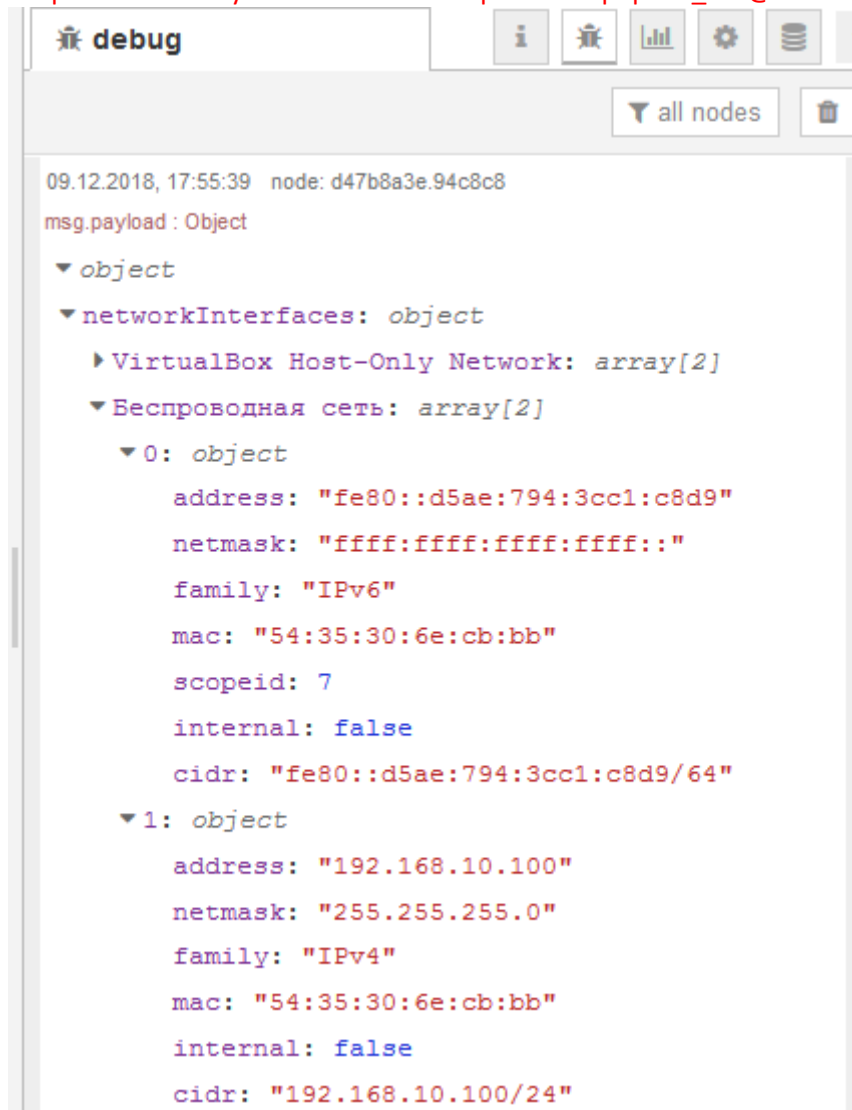


рис.50 Виведення інформації про мережу.

Про об'єкти в JavaScript можна почитати тут <http://яваскрипт.укр/object> або тут <http://learn.javascript.ru/object>. Для перебору усіх властивостей об'єкту можна скористатися конструкцією for..in <http://learn.javascript.ru/object-for-in>

6.3. Створіть програму, що буде виводити перелік MAC адрес для мережних карт, що встановлені на Вашому ПК (рис.51-52).

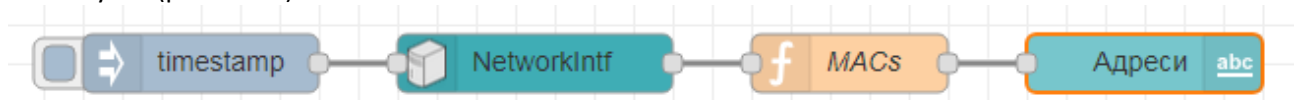


рис.51 Виведення інформації про мережні карти.

Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

```
Edit function node > JavaScript editor

1 var obmsg = msg.payload.networkInterfaces;//об'єкт networkInterfaces
2 var obInterface = {};//об'єкт Interface
3 var MACs = [];//масив адрес MAC
4 var i = 0;
5 //перебираємо усі властивості (ключі) в networkInterfaces
6 for (var keyIf in obmsg) {
7   obInterface = obmsg [keyIf]; //об'єкт Interface по назві мережної карти
8   MACs[i++] = 'MAC' + i + ' ' + obInterface[0].mac; //отримуємо MAC-адресу по 0-му протоколу
9 }
10 msg.payload = MACs;//передаємо в повідомленні
11 return msg;
12
```

рис.52 Програма для вузла MACs.

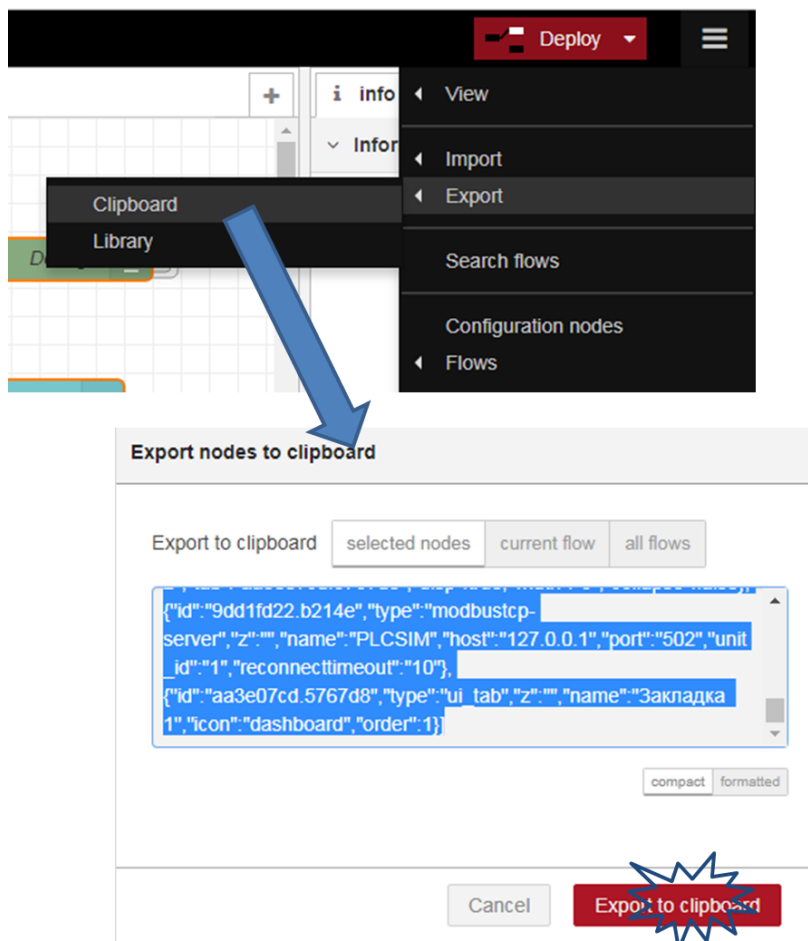
Про JSON <http://яваскрипт.укр/json>

У JavaScript та IoT часто використовуються дані типу JSON. Детально про JSON та їх обробку Ви можете почитати тут <https://flows.nodered.org/node/node-red-contrib-os>

**Зробіть копії екранів програми та візуалізації – це буде звітом до Вашої роботи.**

## 7. Експорт потоку

Відкрийте потік (Flow) створений в лабораторній роботі. Виділіть всі вузли шляхом комбінації Ctrl+A. Усі вузли повинні виділитися червоним контуром. Через меню бокової панелі зробіть експорт виділеного в буфер обміну (рис.53).



Робоча чернетка. Автор - Олександр Пупена, доц. ІАСУ НУХТ. Поширюється для тестування і зворотного зв'язку з побажаннями і правками. [pupena\\_san@ukr.net](mailto:pupena_san@ukr.net)

рис.53. Експорт потоку Node-RED в буфер обміну.

Створіть текстовий файл з назвою Lab1.txt і скопіюйте з буферу обміну текст експорту. Збережіть файл, він вам знадобиться для звіту та для захисту лабораторних робіт у майбутньому.