Міністерство освіти і науки України

Запорізький національний технічний університет

кафедра програмних засобів

**ЗВІТ**

з лабораторної роботи № 3

з дисципліни «Проектування інформаційних систем» з теми:

**«РОБОТА З ПЛАТОЮ РОЗШИРЕННЯ ТА ДАТЧИКАМИ ДЛЯ RASPBERRY PI ЗА ДОПОМОГОЮ NODE-RED»**

Виконала:

студент групи КНТ-215 К.С.Полякова

Прийняла:

к.т.н., доцент Г.В. Табунщик

## 1. Мета роботи

Навчитися працювати з платою розширення та датчиками для Raspberry Pi, використовуючи Node-RED.

## 2. Завдання до роботи

## 2.1 Застосувавши конспект лекцій та додаткову літературу, вивчити принципи роботи з Node-RED.

## 2.2 Використовуючи приклади, реалізувати програму, яка буде послідовно вмикати та вимикати світлодіоди на платі розширення.

## 2.3 Оформити звіт.

## 3. Теоретичні відомості

Node-RED – це потужний open-source інструмент, який спрощує програмування при створенні проектів в області інтернет речей. Він використовує технологію візуального програмування: щоб налаштувати проект на виконання будь-якої задачі, користувач підключає один до одного блоки коду, які називаються «вузлами». Система з підключених один до одного вузлів називається «потоком».

Node-RED надає редактор потоків на основі браузера, використовуючи широкий діапазон вузлів в палітрі. Функції JavaScript створюються за допомогою текстового редактора. Вбудована бібліотека дозволяє зберігати корисні функції, шаблони або потоки для повторного використання. На рис 3.1 зображене головне вікно редактора потоків:

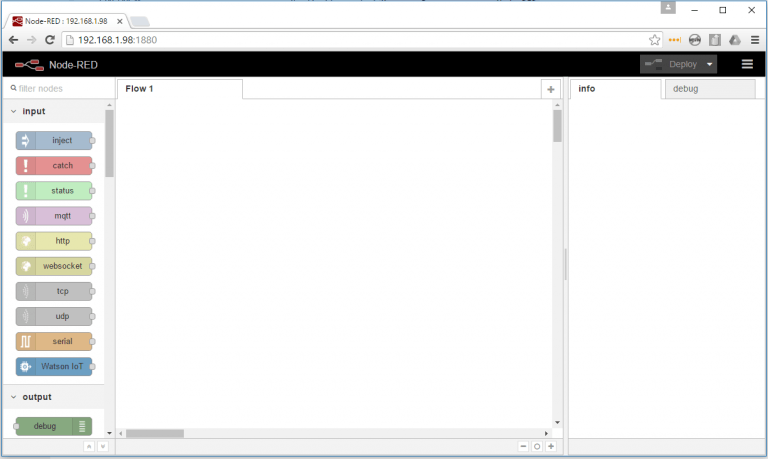


Рисунок 3.1 – Редактор потоків

**4. Результат виконання роботи**

Під час виконання лабораторної роботи було розроблено два варіанта програми, яка керує роботою світлодіодів.

В програмі використовувалися такі типи вузлів :

1. Inject – вставляє в повідомлення часову мітку або налаштований користувачем текст. Може бути налаштований для введення вручну, із заданим інтервалом або в певний час.
2. Function – універсальний програмований функціональний вузол. Використовуючи стандартний JavaScript, вузол може бути адаптований для виконання складної обробки на своїх вхідних повідомленнях, що генерують одне або кілька вихідних повідомлень.
3. Delay – загальний вузол, який затримує повідомлення на певний або випадковий час. Також можна налаштувати обмеження потоку повідомлень (наприклад, 10 повідомлень за секунду).
4. Rpi\_gpio out – вихідний вузол Raspberry Pi. Очікує msg.payload з 0 або 1 (або true або false). Обраний фізичний контакт буде переведено у стан HIGH або LOW в залежності від переданого значення. Початкове значення контакту також може бути встановлено у 0 або 1. При використанні режиму PWM очікується вхідне значення числа від 0 до 100.

В першому випадку керування світлодіодів виконується за допомогою Inject, Delay та Rpi\_gpio out вузлів. На рисунку 4.1 зображено схему потоку для виконання завдання за допомогою вузла Delay .

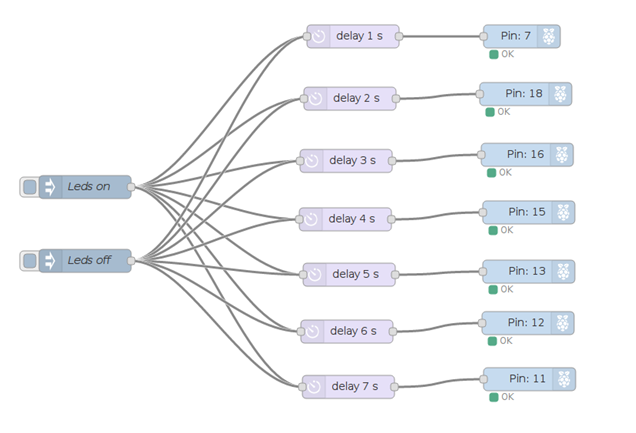


Рисунок 4.1 – Потік з використанням вузла Delay.

В другому випадку керування світлодіодів виконується за допомогою Inject, Function та Rpi\_gpio out вузлів. На рисунку 4.2 зображено схему потоку для виконання завдання за допомогою вузла Function.

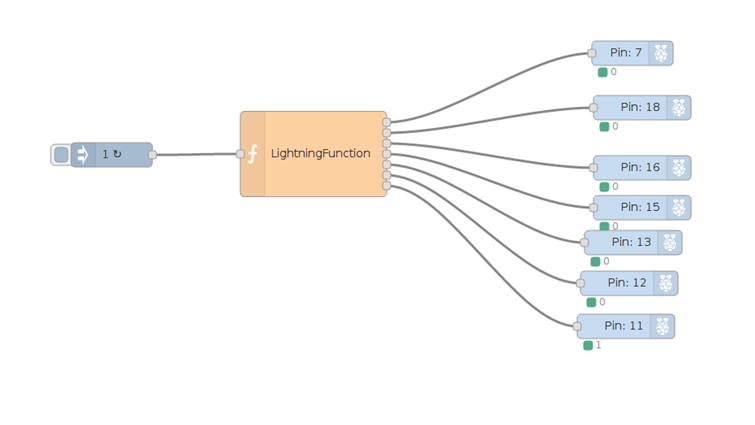


Рисунок 4.2 – Потік з використанням вузла Function.

Код вузла Function приведено нижче :

var ledIndx = context.get('ledIndx');

if (!ledIndx)

var ledIndx = 0;

var arr = [{payload : 0}, {payload : 0},

{payload : 0}, {payload : 0}, {payload : 0},

{payload : 0}, {payload : 0}];

arr[ledIndx] = {payload : 1};

ledIndx++;

if (ledIndx > 6)

ledIndx = 0;

context.set('ledIndx', ledIndx);

//return [{payload : 0}, {payload : 0},

//{payload : 0}, {payload : 0}, {payload : 0},

//{payload : 0}, {payload : 0}];

return arr;

**5. Висновок**

На цій лабораторній роботі я навчилася працювати з платою розширення та датчиками для Raspberry Pi, використовуючи Node-RED, створила програми що керують роботою світлодіодів.