

Міністерство освіти і науки України  
Національний університет «Запорізька політехніка»

кафедра програмних засобів

## **ЗВІТ**

з дисципліни «ПРОЕКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ»

з лабораторної роботи №4 на тему:

«УСТАНОВКА І НАСТРОЙКА ВЕБ-СЕРВЕРА НА RASPBERRY PI»

Виконали:

студенти групи КНТ-216

Н.Є. Трошина

М.В. Ярмачек

М.К. Костюкович

Прийняв:

к.т.н., професор

Г. В. Табунщик

2019 р.

**Мета роботи:**

Навчитися працювати з платою розширення та датчиками для Raspberry Pi.

**Завдання:**

1. Провести установку і базове налаштування пакетів Apache, PHP5, MySQL в середовищі Raspbian GNU / Linux.
2. Створити HTML-документ, який буде відображати створений веб-сервер і переконатися в його доступності по мережі.
3. Оформити звіт. Звіт з лабораторної роботи повинен містити: мету роботи; завдання на виконання роботи; тексти програм; результати роботи; висновки.

**Хід роботи:**

Почнемо з установки необхідного програмного забезпечення, нам знадобиться підключення до мережі. Для установки використовується пакетний менеджер apt-get. Перед установкою також рекомендується оновити репозиторії системи командою apt-get update.

Приступимо до встановлення ПО:

1. Для установки сервера apache використовуємо команду:

```
apt-get install apache2
```

2. Для установки сервера баз даних MySQL:

```
sudo apt-get install mysql-server
```

3. Для установки пакета PHP і зв'язки його з MySQL:

```
sudo apt-get install php5
```

```
sudo apt-get install php5-mysql
```

Після цих операцій веб-сервер буде встановлено і запущено.

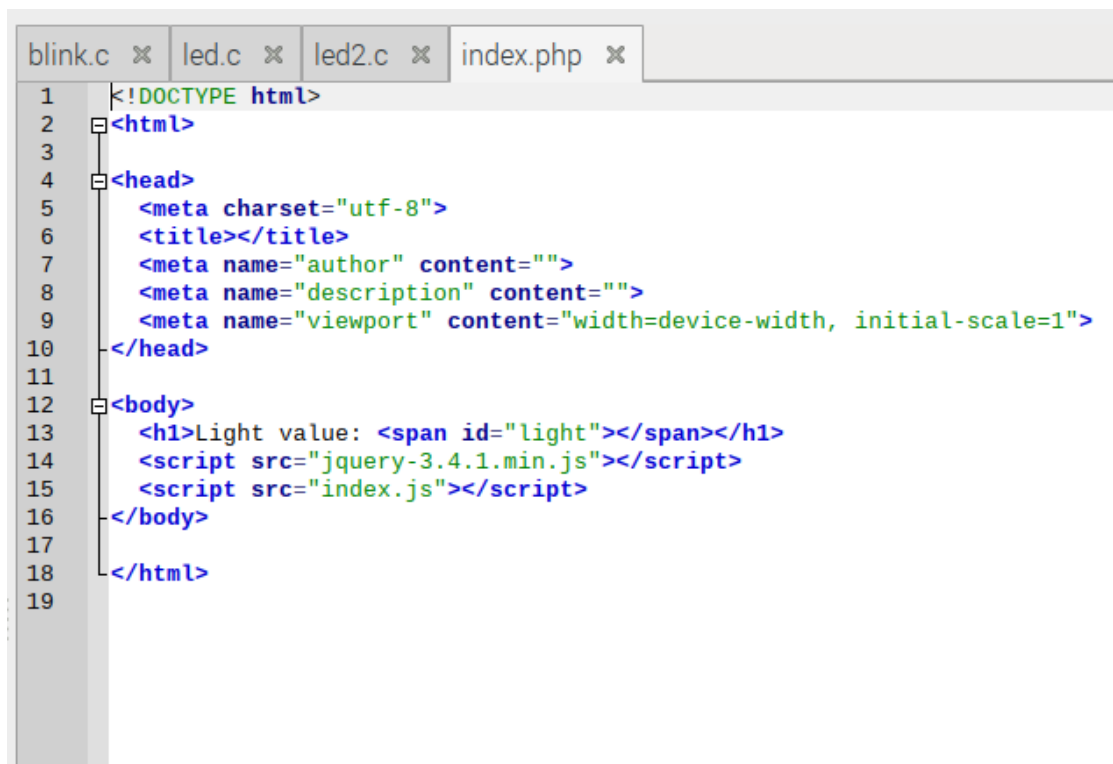
Для того, щоб перевірити його працездатність, ми можемо відкрити браузер і ввести в адресний рядок: localhost або 127.0.0.1(адреса локального сервера за замовчуванням).

За замовчуванням, файли, які віддає Apache, розташовуються в каталозі /var /www .

Для того, щоб відредагувати головну сторінку нашого сайту, потрібно запустити текстовий редактор:

```
sudo mousepad /var/www/index.html
```

У цей файл нам потрібно вписати базову HTML-розмітку (рис.4.1).



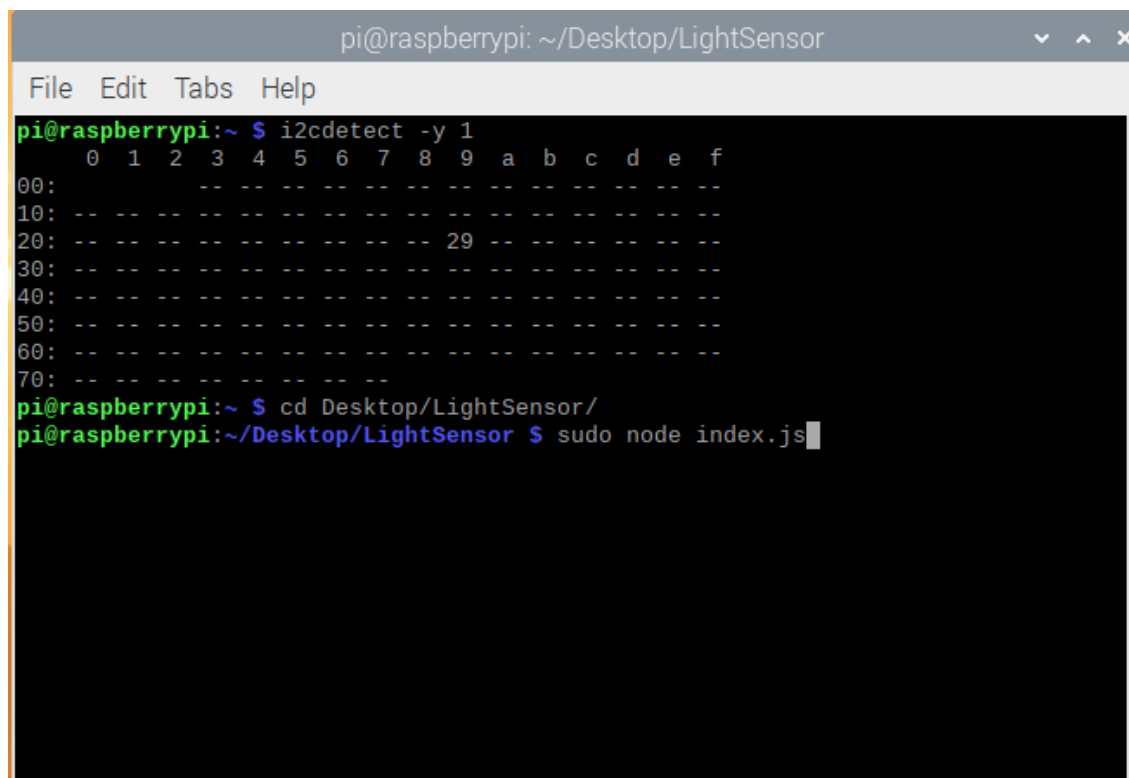
```

1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <meta charset="utf-8">
6      <title></title>
7      <meta name="author" content="">
8      <meta name="description" content="">
9      <meta name="viewport" content="width=device-width, initial-scale=1">
10 </head>
11
12 <body>
13     <h1>Light value: <span id="light"></span></h1>
14     <script src="jquery-3.4.1.min.js"></script>
15     <script src="index.js"></script>
16 </body>
17
18 </html>
19

```

Рисунок 4.1 – Розмітка HTML-сторінки

Після з'ясуємо який пін можливо використовувати для i2c зв'язку (застосовується для з'єднання низькошвидкісних периферійних компонентів з процесорами і мікроконтролерами), скористаємося командою `i2cdetect -y 1`.



```

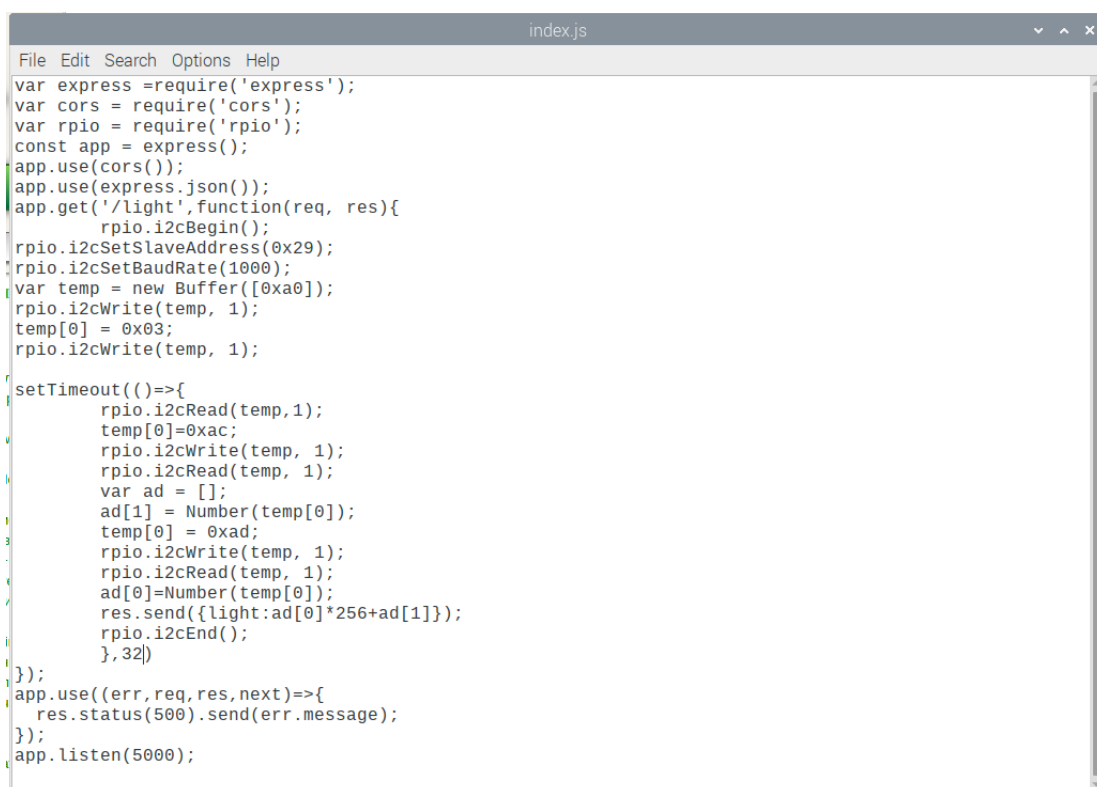
pi@raspberrypi: ~/Desktop/LightSensor
File Edit Tabs Help
pi@raspberrypi:~ $ i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  29  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
pi@raspberrypi:~ $ cd Desktop/LightSensor/
pi@raspberrypi:~/Desktop/LightSensor $ sudo node index.js

```

Рисунок 4.2 – Пін для i2c зв'язку

З'ясувавши, що на платі пін, який нам потрібен, має номер 29, будемо його використовувати в подальшому.

Тепер реалізуємо програму для зчитування даних з датчику освітленості, номер піну якого ми й з'ясовували, та подальшою відправкою на клієнт (рис.4.3).



```


index.js
File Edit Search Options Help
var express = require('express');
var cors = require('cors');
var rprio = require('rprio');
const app = express();
app.use(cors());
app.use(express.json());
app.get('/light', function(req, res){
    rprio.i2cBegin();
    rprio.i2cSetSlaveAddress(0x29);
    rprio.i2cSetBaudRate(1000);
    var temp = new Buffer([0xa0]);
    rprio.i2cWrite(temp, 1);
    temp[0] = 0x03;
    rprio.i2cWrite(temp, 1);

    setTimeout(()=>{
        rprio.i2cRead(temp, 1);
        temp[0]=0xac;
        rprio.i2cWrite(temp, 1);
        rprio.i2cRead(temp, 1);
        var ad = [];
        ad[1] = Number(temp[0]);
        temp[0] = 0xad;
        rprio.i2cWrite(temp, 1);
        rprio.i2cRead(temp, 1);
        ad[0]=Number(temp[0]);
        res.send({light:ad[0]*256+ad[1]});
        rprio.i2cEnd();
    }, 32)
});
app.use((err, req, res, next)=>{
    res.status(500).send(err.message);
});
app.listen(5000);

```

Рисунок 4.3 – JS-реалізація коду

Також необхідно вказати з яким інтервалом сервер буде оновлювати дані з датчика, файл знаходиться за адресою /var /www /html (рис.4.4).



```

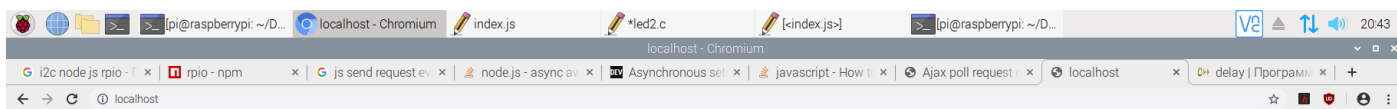
<index.js>
File Edit Search Options Help
setInterval(function(){bob()}, 3000);

function bob () {
    fetch("http://10.0.79.165:5000/light").then((res)=>{
        res.json().then((data)=> {
            $("#light").text(data.light);
        });
    });
}

```

Рисунок 4.4 – Реалізація оновлення даних кожні 3 с

Фінальний вигляд сторінки, дані на якій оновлюються кожні 3 секунди представлено на рисунку 4.5.



**Light value: 588**

Рисунок 4.5 – Загальний вигляд сторінки

### Висновки:

Так як Raspberry Pi представляє з себе повноцінний комп'ютер з підтримкою роботи в мережі, ми можемо розгорнути на ньому класичний веб-сервер. В даному випадку ми скористаємося зв'язкою, яку називають LAMP (Linux, Apache, MySQL, PHP). Raspberry Pi можна використовувати як тестовий сервер для розробника, або як невеликий веб-сервер для власних потреб. Також, тому як Raspberry Pi часто використовують для збору даних з різних датчиків, за допомогою веб-сервера можна організувати зручний їх перегляд.

Дана реалізація дозволяє не оновлювати сторінку щоразу, щоб отримати нові дані, а робить це самостійно без участі користувача.