

УДК 004.8+004.5

DOI: 10.31673/2412-9070.2021.035459

Б. І. СВЕРДЛЮК, інженер першої категорії;  
 Ю. К. КАГРАМАНОВА, інженер першої категорії;  
 В. О. ХОМЕНЧУК, аспірант;  
 В. О. ВЛАСЕНКО;  
 К. В. ПОЛОНСЬКИЙ,  
 Державний університет телекомунікацій, Київ

## КЕРУВАННЯ ДАВАЧАМИ РОЗУМНОГО БУДИНКУ ЗА ДОПОМОГОЮ ГОЛОСОВОГО ПОМІЧНИКА GOOGLE ASSISTANT

У результаті дослідження було створено робочу модель керування пристроями Розумного будинку та Інтернету речей за допомогою голосового інтерфейсу. Розроблена робоча модель ґрунтується на інтеграції Google Assistant, Dialogflow, PHP і MQTT. Голосові інтерфейси тісно пов'язані не лише з користуванням смартфонами, а й з носимою електронікою, автономним керуванням, колонками-асистентами та навіть іграми. Зі свого боку Інтернет речей сьогодні є не тільки концепцією, а ще й реальним рухом зі своїми технологіями та протоколами. Їх різноманіття та застосування відкритих технологій дає можливість гнучко підібрати інструменти для потрібного вам проекту з огляду на його специфіку.

Розумні системи керування освітленням набули великої популярності через зручність таких вирішень (віддалене ввімкнення та вимкнення, збільшення та зменшення яскравості), через можливість збору та аналізу даних та здатність приймати рішення залежно від інформації щодо зовнішніх давачів і подій.

**Ключові слова:** IoT; Інтернет речей; голосові інтерфейси; MQTT; PHP; Dialogflow; ESP; Google Assistant.

### Вступ

Останніми роками голосові інтерфейси набувають тісного зв'язку не лише з користуванням смартфонами, а й з носимою електронікою, автономним керуванням, колонками-асистентами та навіть іграми.

Проте Інтернет речей сьогодні можна розглядати не тільки як концепцію, а й як реальний рух зі своїми технологіями та протоколами. Їх розмаїття та використання відкритих технологій дає змогу гнучко підібрати інструменти для будь-якого потрібного вам проекту, беручи до уваги його специфіку.

**Аналіз дослідження.** У результаті дослідження запропоновано рекомендації щодо створення та організації системи зв'язку та ІТ-систем. Розроблено робочу модель такої взаємодії. Створено робочу систему розумного освітлення з використанням протоколу Інтернету речей Message Queuing Telemetry Transport (MQTT).

**Мета дослідження.** Організувати робочу модель віддаленого керування освітленням через смартфон на основі мікроконтролера ESP8266 з використанням інструменту для створення голосових інтерфейсів Google Dialogflow, webhooks, PHP та MQTT-брокера.

### Основна частина

Для дослідження нам знадобиться контролер ESP8266 або інші контролери з підімкненням до мережі «Інтернет», а також сервер із дійсним ssl сертифікатом (замість валідного ssl можна скористатися зворотним проху, в якого такий вже є) і MQTT-сервер (broker) (рис. 1).

Для інтеграції з Google Assistant використовуємо сервіс Google Actions і Dialogflow.

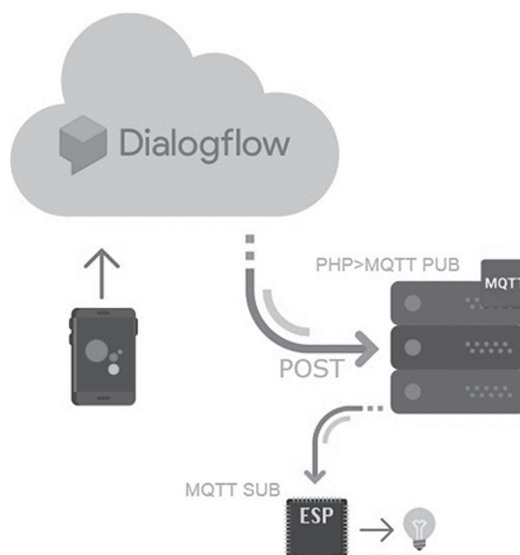


Рис. 1. Модель взаємодії системи розумного освітлення

**Створення та налагодження проекту.** Для початку потрібно авторизуватися на сервісі і створити проект, вибрати мову і регіон. Після цього потрібно вибрати сферу застосування. Нами було взято Kids & Family. З чужого досвіду маємо, що логічніше напрошується категорія Smart Home, але вона коректно працює лише з пристроями від Google, а отже, для проектів зі своїми давачами для коректної роботи краще вибрати іншу категорію.

Читаємо і приймаємо угоди. У категорії Quick setup переходимо до пункту Decide how your Action is invoked. Придумуємо як зватимуть нашого бота і вибираємо голос із доступних, зберігаємо.

Переходимо до вкладки Actions і додаємо дію Custom Intenet.

**Dialogflow.** Після натискання кнопки Build нас перенаправляють на ресурс Dialogflow. Тут теж авторизуємося і розпочинаємо створення проєкту:

1. Вибираємо мову за замовчуванням і часовий пояс.

2. Переходимо до вкладки Fulfillment і вмикаємо пункт Webhook (інтерфейс взаємодії між двома програмами). Тут маємо вказати адресу скрипту (про скрипт детальніше розглядатимемо далі), на який Dialogflow буде надсилати POST-запит.

3. Налаштовуємо команди і відповіді. Вкладка Intents має кілька стандартних заготовлених команд і відповідей на них. У нашому разі потрібно створити нову команду. Для цього натискаємо на «+» у вкладці Intents. Придумуємо власний псевдонім команди і вибираємо «зберегти».

Саму сторінку можна умовно поділити на три категорії:

**Training phrases** — це слова, на які Assistant реагує командою. Вони можуть бути різними для однієї команди. Наприклад, такі слова і вирази, як *посвіти, вруби світло, увімкни лампу* тощо;

**Action** — сама дія або одне і конкретне слово, яке буде розуміти давач;

**Responses** — це відповіді Assistant після виконання команди (тут просто поле для творчості).

**Налаштування хостингу.** Який саме хостинг буде вибрано значення не має. У нашому разі як сервер було придбано найдешевший (5 \$) дроплет, на який було встановлено Debian 10.2.

**Налаштування зворотного проxy і DNS.** Цю частину можна пропустити, якщо у вас є хостинг із дійсним сертифікатом.

Сам сервіс Cloudflare пропонує безліч імен нижніх рівнів, тому якщо виникли проблеми, можливо варто в колонці Content біля `www ***`, `https // www ***` вказати свою IP-адресу замість дефолтної (`***` — це ваш домен).

Для взаємодії з Dialogflow потрібен SSL-сертифікат. Проте можна не встановлювати його на сервер, а замість цього скористатися DNS проxy Cloudflare (безкоштовний — basic). Для цього під час налаштування Cloudflare потрібно додати своє раніше придбане доменне ім'я, а до нього IP-адресу свого сервера (колонка Content). Також додати два A-записи (на зображенні це Value) у налаштуваннях провайдера імен. Варто зазначити, що додавання запису — це не секундна справа і може знадобитися навіть кілька робочих днів.

#### Установлення програм

1. Насамперед після першого запуску сервера потрібно не забути оновити базу даних пакетів `apt update`, а потім оновити встановлені пакети `apt upgrade`.

2. Після цього встановлюємо LAMP:

Apache

```
apt install apache2
```

PHP/MySQL

```
apt-get install php libapache2-mod-php php-mcrypt php-mysql.
```

3. Додаємо адресу сервера в `apache2.conf`. Це можна виконати командою `nano/etc/apache2/apache2.conf`, вказавши `ServerName ***` у кінці вашого файлу, де замість зірочок потрібно підставити IP сервера. Перевіряємо синтаксис і перезапускаємо службу Apache [1].

Після успішного виконання описаних дій під час введення вашого доменного імені в рядок браузера отримаємо сторінку вітання Apache2.

Якщо цього не сталося, перевіряємо чи коректно виконувалися команди, а також чи працює сторінка в разі незахищеного `http` з'єднання. Якщо все ж працює, можливо сервер слухає 80, а не 443 порт, або на ньому вже запущено якусь службу [2].

4. Установлення Бібліотеки Mosquitto для PHP. Пристрої Інтернету речей можуть використовувати різні протоколи для взаємодії. Одним із таких є протокол MQTT, який працює за принципом видавець-підписник. У такому разі підписники можуть отримувати інформацію від багатьох видавців. Але оскільки протокол розуміє тільки певні типи повідомлень, для нього потрібен перетворювач (broker). Ось його ми і налаштуємо.

1. Якщо у вас не встановлено PECL, встановлюємо його:

```
apt pecl install
```

```
2. install peclMosquitto-alpha
```

```
3. apt install mosquitto mosquitto-clients [3].
```

**Налаштування PHP скрипту, підписки на топик.** Це, власне, сам скрипт. Його призначення приймати POST-запит від Dialogflow, виокремлювати з нього Action і передавати його як повідомлення на брокер у тему, назву якої зазначено в кінці скрипту. Як ви назвете скрипт значення не має. До речі, це і є скрипт, який ми вказували у вкладці Fulfillment. Поміщаємо скрипт за адресою `/var/www/html`.

Топіки створюються підписниками. Аби створити топик, скористаємося командою

```
mosquitto_sub -h localhost -t / test / light.
```

Замість `localhost` можна задати будь-яку іншу адресу або домен, де ми хочемо створити видавця (pub).

Замість `/ test / light` можна задати будь-яку тему. У нашому разі головне, щоб її було зазначено у скрипті.

Повідомлення створюються видавцями. Щоб створити повідомлення, можна скористатися командою

```
mosquitto_pub -h localhost -t / test / light -m «light».
```

Але вона нам не знадобиться, оскільки нашим видавцем (pub) буде наш додаток. Схема така: коли наш додаток отримує команду, він відправляє запит на наш скрипт, скрипт на брокера, а брокер на підписника (esp8266) (рис. 2).

\* password — її пароль. Після const char \* mqtt\_server вказати IP-адресу свого сервера [5].

**Результат.** Після компіляції скетчу ми отримуємо додаток, який інтегровано в Google Assistant і керує давачами. Натомість смартфона ми

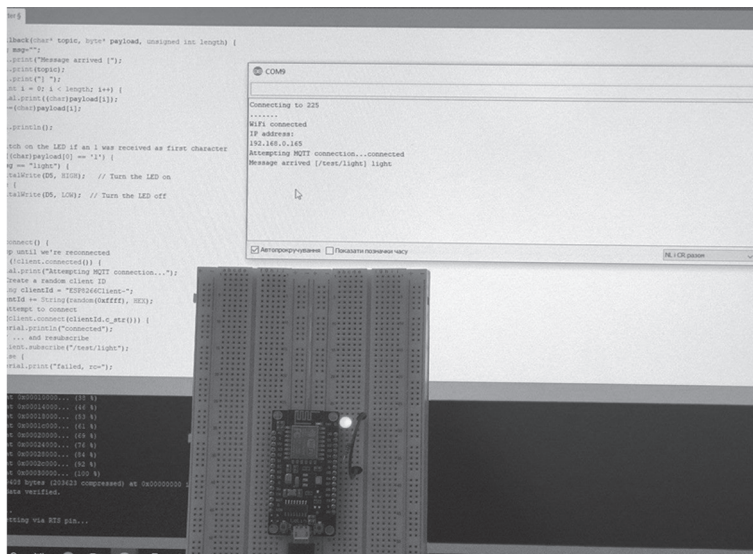


Рис. 2. Керування освітленням через ESP8266

Перевіряти відправлення повідомлень ми будемо через вкладку Test Action Console [4].

**Прошивка ESP8266.** Для прошивки будемо використовувати Arduino IDE. Якщо IDE встановлюється вперше, потрібно не забути про драйвер ch340. По дефолту в Arduino цієї плати немає.

У Файл >> Налаштування потрібно вказати адресу додаткових плат: [arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](https://arduino.esp8266.com/stable/package_esp8266com_index.json).

В Інструменти >> плата >> менеджер плат потрібно встановити пакет esp8266.

У скетчі після const char \* ssid маємо зазначити назву своєї Wi-Fi мережі. Після const char

використовували вебдодаток, але також тестували на Android — результат однаковий. Головне, якщо тестування відбувається зі смартфона, не забудьте сказати: «Говорити з додатком \*\*\*».

**Висновки**

У майбутньому локальні та глобальні протоколи IoT матимуть свої сфери використання, можливо з часом якийсь із протоколів покаже більшу ефективність. Проте уніфікованість цих протоколів уможливіло інтегрування їх з іншими сервісами, мережами спостереження, API та Webhook. І саме таке різноманіття дає змогу творити та винаходити нові методи взаємодії.

**Список використаної літератури**

1. URL: [digitalocean.com/community/tutorials/linux-apache-mysql-php-lamp-ubuntu-16-04-ru](https://digitalocean.com/community/tutorials/linux-apache-mysql-php-lamp-ubuntu-16-04-ru)
2. URL: [losst.ru/pochemu-apache-ne-zapuskaetsya](https://losst.ru/pochemu-apache-ne-zapuskaetsya)
3. URL: [readthedocs.org/projects/mosquitto-php/downloads/pdf/latest/](https://readthedocs.org/projects/mosquitto-php/downloads/pdf/latest/)
4. URL: [calameo.com/books/002652204fe285c5fce5-php-script](https://calameo.com/books/002652204fe285c5fce5-php-script) <https://telegra.ph/php-script-iot-03-12>
5. URL: [forum.arduino.cc/index.php?topic=539416.0](https://forum.arduino.cc/index.php?topic=539416.0)

Б. И. Свердлюк, Ю. К. Каграманова, В. А. Хоменчук, В. А. Власенко, К. В. Полонский

**УПРАВЛЕНИЕ ДАТЧИКАМИ УМНОГО ДОМА С ПОМОЩЬЮ GOOGLE ASSISTANT**

В результате исследования была создана рабочая модель управления устройствами Умного дома и Интернета вещей с помощью голосового интерфейса. Разработанная рабочая модель основывается на интеграции Google Assistant, Dialogflow, PHP и MQTT. Голосовые интерфейсы тесно связаны не только с использованием смартфонов, но и с носимой электроникой, автономным управлением, колонками-ассистентами и даже играми. Со своей стороны Интернет вещей сегодня является не только концепцией, но и реальным движением со своими технологиями и протоколами. Их многообразие и применение открытых технологий дает возможность гибко подобрать инструменты для нужного вам проекта, учитывая его специфику. Умные системы управления освещением приобрели большую популярность из-за удобства таких решений (удаленное включение

и выключение, увеличение и уменьшение яркости), из-за возможности сбора и анализа данных, а также способности принимать решения в зависимости от информации по внешним датчикам и событиям.

**Ключевые слова:** IoT; Интернет вещей; голосовые интерфейсы; MQTT; PHP; Dialogflow; ESP; Google Assistant.

*B. I. Sverdlyuk, Yu. K. Kagramanova, V. O. Khomenchuk, V. O. Vlasenko, K. V. Polonskiy*

### **CONTROL OF SMART HOUSE SENSORS USING GOOGLE ASSISTANT**

*As a result of the research, a working model was created by smart home devices and the Internet of Things using a voice interface. Integrated Google Assistant, Dialogflow, PHP and MQTT. Voice interfaces are closely related not only to the use of smartphones, but also to wearable electronics, autonomous control, speaker speakers and even games. In turn, the Internet of Things today is not only a concept, but also a real movement with its technologies and protocols. Their variety and use of open technologies allows you to flexibly choose the tools for the project you need, taking into account its specifics. Smart lighting control systems are very popular because of the convenience of such solutions (remote on and off, increase and decrease brightness), the ability to collect and analyze data, the ability to make decisions depending on the ratio of external sensors and events.*

*Google Assistant is an artificial intelligence-powered virtual assistant developed by Google that is primarily available on mobile and smart home devices. Unlike the company's previous virtual assistant, Google Now, the Google Assistant can engage in two-way conversations.*

*Dialogflow is a natural language understanding platform used to design and integrate a conversational user interface into mobile apps, web applications, devices, bots, interactive voice response systems and related uses.*

*The ESP8266 is a low-cost Wi-Fi microchip, with a full TCP/IP stack and microcontroller capability, produced by Espressif Systems in Shanghai, China.*

*The Message Queuing Telemetry Transport (MQTT) is a lightweight, publish-subscribe network protocol that transports messages between devices. The protocol usually runs over TCP/IP; however, any network protocol that provides ordered, lossless, bi-directional connections can support MQTT. It is designed for connections with remote locations where a «small code footprint» is required or the network bandwidth is limited.*

**Keywords:** IoT; Internet of Things; voice interfaces; MQTT; PHP; Dialogflow; ESP; Google Assistant.

