



DOI: 10.54664/BCFB1192

Сензорна мрежа за интелигентно, многосекторно управление на осветление

Кристина Николова, Теодор Калушков

Wireless Sensor Network for Smart, Multisectoral Lighting Management

Kristina Nikolova, Teodor Kalushkov

Abstract: *This article describes a mesh wireless sensor network that manages the lighting of large buildings and areas that allow separation of individual sectors within them. If there are no visitors in any of the sectors, the system turns off the lights in them and thus saves electricity. When any visitor appears, the system activates the lighting in the corresponding sectors and in the adjacent ones, thus preserving the comfort of the vision and the visitor is left with the impression that the whole area is fully illuminated. Apart from being a green technology, this approach provides additional benefits, such as easier guarding of open spaces and buildings with windows in the dark hours. The components used are easily available and allow for budget and, at the same time, efficient implementations.*

Keywords: *smart lighting; energy savings; building automation; smart grid; controller management.*

ВЪВЕДЕНИЕ

Част от концепциите за „интелигентен дом“ и „интелигентен град“ включват ефективно изразходване на електричеството и в частност управление на осветлението в посочените населени места. Целта на интелигентните системи е да намалят разходите, без да ограничават комфорта на населението, а точно обратното – хората да повишат качеството си на живот. Редуцирайки въглеродните емисии чрез енергийни икономии се въздейства положително и върху околната среда. Прилагането на посочените концептуални детайли изисква високо-технологични решения с повишена автоматизация на процесите по управление на градската среда. Съществуват много разработки в тази област [3,6].

Тази статия представя отговор на описаните предизвикателства чрез контролерни възли, които могат да комуникират помежду си чрез Wi-Fi мрежа от типа “peer-to-peer”. Всеки един от възлите е снабден със сензор за присъствие (движение) и има изход за управление на едно или много осветителни тела с различна според нуждите консумация. При необходимост е възможно към всеки контролер да се включи и сензор за осветеност, който да не позволява включване на осветлението, ако сградата има прозорци, от които да постъпва естествена слънчева светлина или ако системата се използва за осветление на открити пространства. Особеностите, които трябва да се отчетат при конкретни имплементации на разработената система са:

- осветяваните площи трябва да позволяват разделяне на физически и логически сектори;
- отделните сектори да трябва да бъдат с различен интензитет на посещаемост в конкретен момент или интервали от време;
- ефективността се повишава, когато посещаемостта на повечето сектори е ниска през повечето време;
- големината на осветените площи също е от значение за степента на ефективност, като връзката е право-пропорционална.

Конкретни приложения на разработката могат да включват осветление на музеи, галерии, пещери, ритуални зали, промишлени и обществени сгради, подземни и надземни паркинги, складове и други. Всички те консумират значителни количества електроенергия, ако се осветяват постоянно в работно време, дори и да използват съвременни осветителни тела с понижена консумация.

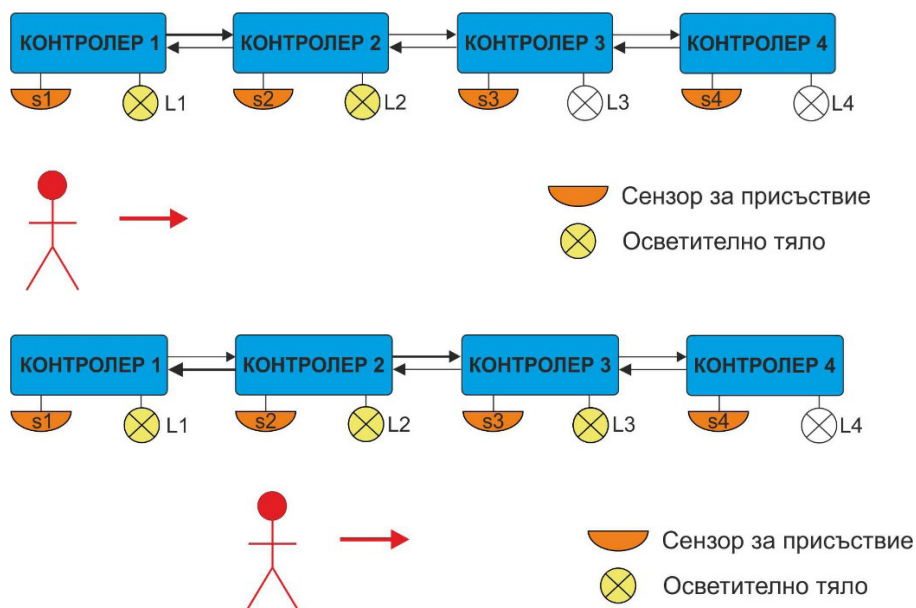
Традиционно използваните осветителни системи със сензор имат сериозен недостатък по отношение на комфорта на наблюдателя. Рязкото светване при детектиране на движение предизвиква неприятно усещане за посетителите и поради това те рефлексивно присвиват клепачи за да позволят на очите да се адаптират плавно към променения светлинен интензитет. Съществуват разбира се и осветителни тела с димиране (плавно усилване на интензитета на светлината), но обикновено се отличават с малко по-усложнена елементна база, а оттам и завишена цена.

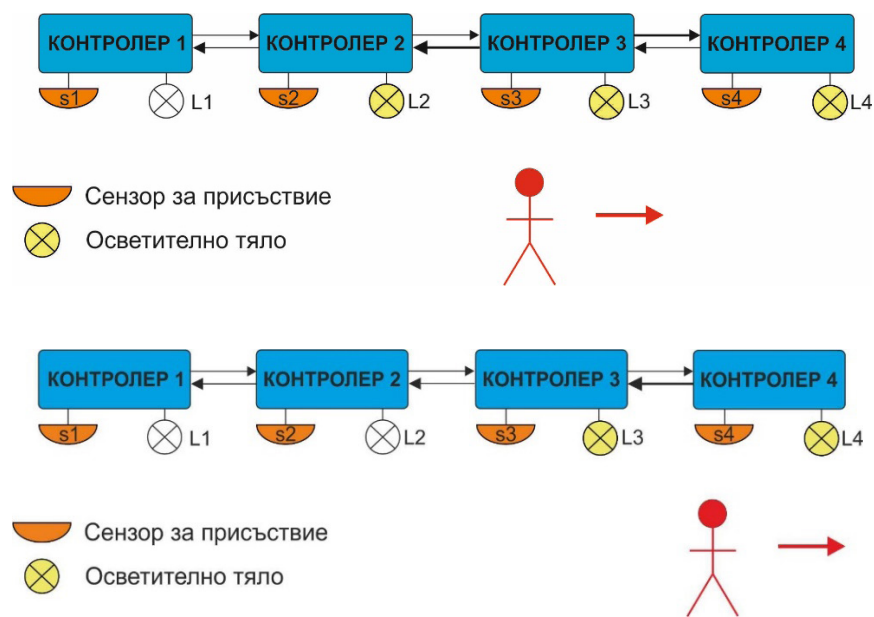
В предложената система, потребителя остава с впечатление, че се движи по изцяло осветени площи(светлинна вълна), без да усеща от къде се реализират икономии или да чувства зрителен дискомфорт.

ОПЕРАТИВНА ЛОГИКА НА КОНТРОЛЕРИТЕ

За апробиране на разработваната система е изграден опростен модел с прототип съставен от четири възела, който е предназначен за линейно разположение на отделните зони за осветление (тунел). На практика обаче, без съществени изменения, системата може да бъде разгърната и в равнинен модел (етаж от сграда) или обемен модел (многоетажна сграда).

Модела и съответно прототипа фиг.1. функционират чрез обмяна на информация за състоянието на сензорите между съседните възли.





Фиг. 1. Реакция на системата при преминаване на човек през работните зони

При липса на засечено присъствие, всички осветителни тела са изгасени. Когато в зоната на покритие на сензора към контролер 1 се появи човек, светват едновременно осветителните тела свързани към контролери 1 и 2 (1 предава информацията от сензора си на 2). За да не се включи осветлението внезапно пред посетителя е възможно сензорът към контролер 1 да бъде дублиран и единия да се разположи пред помещението на първа зона. Когато посетителя премине във зоната обхваната от сензора на контролер 2, той предава информация за детектираното присъствие на съседните контролери 1 и 3. Осветителните тела към първи и втори контролер остават да светят и освен това се включват и тези към контролер 3. При постъпване в трета зона свети осветлението във втора трета и четвърта – съответно контролер 3 изпраща до 2 и 4 информация за задействането на сензора. В този момент осветлението към контролер 1 изчаква зададеното закъснение от 5 секунди и изгасва. При навлизане в четвъртата зона е осветено в трета и четвърта. Контролер 4 предава на 3, че открива присъствие, а контролер 2 изчаква 5 секундното закъснение и изгасва. След напускане на четвърта зона, осветлението в нея и в предходната 3 остава да работи още 5 секунди и също изгасва

ЕТАПИ НА ПРОЕКТИРАНЕ И ПРАКТИЧЕСКА РЕАЛИЗАЦИЯ

Първоначалните етапи от проектирането включват логическото разделяне на площите и определяне броя и вида на необходимите осветителни тела – корпусното им изпълнение (водо- и влагоустойчивост), техният интензитет, цвят на светлината, ъгъл на насоченост, височина на монтажа, външния им вид от естетична гледна точка и консумирана мощност.

Следва да се подберат подходящи сензори за присъствие(движение) отново по корпусно изпълнение (за таван или за стена, водо- и влагоустойчивост), чувствителност и ъгъл на детектиране. В случая е по-подходящо да се изберат сензори за присъствие, а не за движение, защото те работят с по-висока „резолюция“ и детектират дори и най-малките движения в работните площи. Тази спецификация свежда до минимум вероятността от попадане в ситуация, при която осветлението може да угасне, докато посетителите са в някоя от работните зони.

Следващия етап от проектирането е да се прецизират местата на осветителните тела, сензорите и контролерите. Основно изискване към точките на монтаж на контролерите е да имат

достатъчно добро WiFi покритие със съседните възли. При необходимост, разстоянията могат да бъдат намалени и съответно добавени допълнителни контролерни възли.

Когато са набелязани местата на контролерните възли се опроводяват веригите за хранване на контролерите и силовите връзки до осветителните тела.

За експерименталния прототип са подбрани контролери NodeMCU(ESP8266), поради тяхната популярност, ниска цена, безплатна среда за разработка и множеството поддържани библиотеки. Изискванията за брой и тип на поддържаните програмируеми входове и изходи са също напълно покрити за конкретния случай.

NodeMCU контролерите имат номинално хранващо напрежение 5V и типичен ток на консумация под 100mA. Затова е добре всяка точка да бъде осигурена със хранване с посочените параметри. Важно е в случая да се има предвид, че поради ниското хранващо напрежение, не е удачно да се използва един трансформатор за много възли, разположени на дълги разстояния. Поради пада на напрежение по линията, това може да рефлектира върху стабилността на работа на сензорната мрежа.

Преходът между управляващата контролерна част и силовата осветителна, може да се реализира чрез електро-магнитни релета или при необходимост контактори. В определени ситуации обаче, шумът издаван при превключването на тези устройства може да смути посетителите и затова те могат да бъдат заменени с електронни релета(SSR), които работят безшумно и изключват възможността от отказ поради залепване на контактите. Независимо от избрания вариант за управление на силовите вериги, изброените компоненти трябва да бъдат пресметнати във всеки индивидуален случай съобразно максимално допустимия им ток на работа от една страна и от общо консумирания ток на управляваните в секторите осветителни тела.

За да се премине към програмирането на комуникационните модули, предварително се вличат MAC (Media Access Control) -адресите на всеки един от тях и се вписват в адресна схема. Тя най-общо представя MAC-адреса на всеки конкретен възел и адресите на неговите съседи. Тази информация е необходима, за да може да бъде заложена в индивидуалния програмен код на всеки възел, така че той да може да си комуникира само със съседите. Комуникацията е безжична, а адресацията между възлите е посредством MAC-адресите. Използва се ESP-NOW протокола. Той е разработен от компанията Espressif Systems, която е създала контролерите ESP8266 и ESP32. ESP-NOW е бърз протокол за комуникация, който може да бъде използван за обмен на малки съобщения (до 250 байта) между ESP8266 устройства. Съществуват различни практически приложения, разработени върху посочения протокол [1,2,4,10]. Той използва безжична връзка с ниска мощност и честота на радио-вълните от 2,4 GHz. В случая сдвояването между устройствата е необходимо преди тяхната комуникация. След като сдвояването е направено, връзката е защитена и е peer-to-peer. ESP-NOW е подходящ не само за интелигентно осветление, но и за устройства с дистанционно управление, сензори и други.

След попълване на MAC-адресната схема се преминава към програмирането на отделните контролери.

ПРОГРАМЕН КОД

За реализацията на програмната част на прототипа са написани четири отделни скици с код за четирите контролера. Логиката на всяка от тях е еднаква, като се различават единствено по дефинираните като получатели съседни контролери.

В началото се инсталира Arduino IDE средата за разработка на кода. След това се добавят библиотеките, които ще са необходими. ESP8266WiFi.h е Wi-Fi библиотека за ESP8266, позволяваща на NodeMCU да изпраща и получава данни. espnow.h е библиотека, която позволява използването на ESP-NOW протокола, позволяващ комуникацията между устройства без централен Wi-Fi възел (точка за достъп). Връзката е от тип peer-to-peer. Използваните библиотеки не се инсталират допълнително. За да бъдат достъпни, е достатъчно само да се добави поддръжка за NodeMCU ESP8266 контролера в ArduinoIDE програмната среда.

Следва дефиниране на MAC адресите на съседните контролери и на 2 променливи. В първата се съхранява стойността при четене от сензора за присъствие, а във втората – стойността, която ще бъде получавана от съседните контролери. След това се създава структура от данни, която ще съдържа стойността, изпращана към съседните контролери. Следват две променливи от тип `struct_message`, първата от които се използва за изпращане на данни, а втората за съхраняване на данни при получаване.

При изпращане на данни се активира функция, която има роля да проследява процеса и да изписва в серийния монитор дали изпращането е било успешно или не. При получаване на данни се активира функция, която извежда на серийния монитор информация за получените данни.

Чрез използването на функциите `OnDataSent` и `OnDataRecv` лесно може да се проследи дали осъществяването на комуникация между контролерите е успешно. Следва подпрограмата `setup()`, която се изпълнява веднъж при стартиране на контролера. Използва се за установяване на серийна комуникация между платката и компютъра, по този начин могат да се принтират данни в серийния монитор. Задава се ESP-NOW роля, като в случая ролята е COMBO, защото контролерът ще изпраща и получава данни едновременно. Регистрират се контролерите, с които ще се осъществява комуникация, като се задава и тяхната роля, която също е COMBO.

Следва подпрограмата `loop()`, която представлява цикъл и кодът в него се изпълнява постоянно през определено време. Зададено е да се изпълнява през интервал от 100 милисекунди в случаите, когато няма сигнал за включване на осветителното тяло, тоест нито сензора на конкретния контролер, нито сензорите на неговите съседни отчитат присъствие. Интервалът е кратък, за да се гарантира бърза реакция за включване на осветителните тела.

Докато осветителното тяло на контролера не свети, през 100 милисекунди се извършва прочитане на данни от сензора. Следва условен оператор, който проверява дали има отчетено присъствие или не. В случай, че има, се изпълнява включване на осветлението с продължителност 5 секунди и се изпращат се данни до съседните контролери – в случая само до номер 2 (`broadcastAddress2`), тъй като за пример се разглежда кодът на контролер номер 1. Също така контролерът включва осветителното тяло за 5 секунди и след това го изгася.

В случаите, когато сензорът на контролера не отчете присъствие, се продължава с проверка в друг условен оператор, който проверява дали има получен сигнал от съседните контролери. Ако има, контролерът включва осветителното тяло и го изгася след 5 секунди. Когато се достигне до края на подпрограмата `loop()`, тя се повтаря отново.

При необходимост посочените закъснения могат да бъдат модифицирани.

Така описаният код има следния цялостен вид:

```
#include <ESP8266WiFi.h> //required library
#include <espnow.h> //required library
uint8_t broadcastAddress2[] = {0x2C, 0xF4, 0x32, 0x4B, 0x46, 0x88}; //MAC address of
the neighboring controller 2 2C:F4:32:4B:46:88
int val; //sensor reading variable
int incomingVal = 0; //received sensor reading variable from the neighboring controllers
const long interval = 100;
unsigned long previousMillis = 0;
typedef struct struct_message {
    int stat;
} struct_message; //data structure value for sending to the neighboring controllers
struct_message IRSensorReadings; //variable for sending data
struct_message incomingReadings; //variable for keeping received data
void OnDataSent(uint8_t *mac_addr, uint8_t sendStatus) {
    Serial.println(„\r\nLast Packet Send Status:\t“);
```

```
if (sendStatus == 0) {
  Serial.println(„Delivery Success!“);
}
else {
  Serial.println(„Delivery Fail!“);
}
}
//sending verification function
void OnDataRecv(uint8_t * mac, uint8_t *incomingData, uint8_t len) {
  memcpy(&incomingReadings, incomingData, sizeof(incomingReadings));
  Serial.print(„Bytes received: „);
  Serial.println(len);
  Serial.print(„Value received: „);
  Serial.println(incomingReadings.stat);
  incomingVal = incomingReadings.stat;
}
//received data verification function
void getReadings() {
  val = digitalRead(5);
}
void printIncomingReadings() {
  Serial.println(„INCOMING READINGS:“);
  Serial.print(„IR Sensor: „);
  Serial.println(incomingVal);
}
void setup()
{
  Serial.begin(115200);
  pinMode(14, HIGH);
  WiFi.mode(WIFI_STA);
  WiFi.disconnect();
  if (esp_now_init() != 0) {
    Serial.println(„Error initializing ESP-NOW“);
    return;
  }
  esp_now_set_self_role(ESP_NOW_ROLE_COMBO);
  //setup the controller to send
and receive data
  esp_now_register_send_cb(OnDataSent);
  esp_now_add_peer(broadcastAddress2, ESP_NOW_ROLE_COMBO, 1, NULL, 0);
  //registration
of other controllers for communication
  esp_now_register_recv_cb(OnDataRecv);
}
void loop()
{
  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis >= interval) {
    previousMillis = currentMillis;
    val = digitalRead(5);
    //reading PIR sensor
    if (val == 0) {
      IRSensorReadings.stat = 1;
      esp_now_send(broadcastAddress2, (uint8_t *) &IRSensorReadings, sizeof(IRSensorReadings));
    }
  }
}
```



```

digitalWrite(14, HIGH);
delay(5000);
digitalWrite(14, LOW);
} //turning ON, sending to neighbors, wait, turning OFF
if (incomingVal == 1) {
digitalWrite(14, HIGH);
delay(5000);
digitalWrite(14, LOW);
incomingVal = 0;
} //if received 1, turning ON, wait, turning OFF
}
}
}

```

ЗАКЛЮЧЕНИЕ И БЪДЕЩА РАБОТА

Апробираният прототип на описаната система демонстрира пропорционална ефективност, според мащабите на осветяваните сгради (площи). Такива системи могат да бъдат изградени за много повече на брой зони и осветителни тела в различни области на индустрията. Сензорната мрежа може да се използва при проектирането на интелигентни домове и интелигентни градове, защото това са сфери, които тепърва ще дават ефективен отговор за справяне с актуалните енергийните проблеми и екологичните предизвикателства в индустрията и ежедневието на хората.

ЛИТЕРАТУРА

1. **Eridani, D., Rochim, A. F., and Cesara, F. N.** 2021. “Comparative Performance Study of ESP-NOW, Wi-Fi, Bluetooth Protocols based on Range, Transmission Speed, Latency, Energy Usage and Barrier Resistance.” – In: 2021 International Seminar on Application for Technology of Information and Communication (iSemantic), pp. 322–328. DOI: 10.1109/iSemantic52711.2021.9573246. [19.08.2022]
2. **Hoang, T. N., Van, S.-T., and Nguyen, B. D.** 2019. “ESP-NOW Based Decentralized Low-Cost Voice Communication Systems for Buildings.” – In: 2019 International Symposium on Electrical and Electronics Engineering (ISEE), pp. 108-112. DOI: 10.1109/ISEE2.2019.8921062. [23.08.2022]
3. **Khanchuea, K., and Siripokarpirom, R.** 2019. “A Multi-Protocol IoT Gateway and WiFi/BLE Sensor Nodes for Smart Home and Building Automation: Design and Implementation.” – In: 10th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES), pp. 1–6. DOI: 10.1109/ICTEmSys.2019.8695968. [17.08.2022]
4. **Koushik, M. S. et al.** 2021. “Design and Development of Wireless Sensor Network Based Data Logger with ESP-NOW Protocol.” – In: 2021 6th International Conference for Convergence in Technology (I2CT), pp. 1–5. DOI: 10.1109/I2CT51068.2021.9417914. [23.08.2022]
5. **Suprianto, G., and Wirawan, W.** 2018. Implementation of Distributed Consensus Algorithms for Wireless Sensor Network Using NodeMCU ESP8266. In: 2018 Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS), pp. 192–196, DOI: 10.1109/EECCIS.2018.8692952. [17.08.2022]
6. Retrieved 25 August 2022 from <https://www.smarthomebulgaria.bg/bg/kakvo-%D0%B5-um%D0%B5n-dom>
7. Retrieved 5 September 2022 from <https://napravisam.net/?p=27978>
8. Retrieved 6 September 2022 from <https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor>
9. Retrieved 8 September 2022 from <https://envirementalb.com/security-system-using-pir-sensor/>
10. Retrieved 12 September 2022 from <https://www.espressif.com/en/products/socs/esp8266>
11. Retrieved 12 September 2022 from <https://www.teachmicro.com/solid-state-relay-tutorial-wifi-switch>

ИНФОРМАЦИЯ ЗА АВТОРИТЕ

Кристина Николова – студент, специалност „Софтуерно инженерство“, Факултет „Математика и информатика“, Великотърновски университет „Св. св. Кирил и Методий“, програмист в „Стемо“ ЕООД, e-mail: inikolova50@abv.bg

Теодор Калушков – главен асистент, доктор, Факултет „Математика и информатика“, Великотърновски университет „Св. св. Кирил и Методий“, e-mail: t.kalushkov@ts.uni-vt.bg

ABOUT THE AUTHORS

Kristina Nikolova – student in Software Engineering, Faculty of Mathematics and Informatics, St. Cyril and St. Methodius University of Veliko Tarnovo, Junior Programmer at ‘Stemo’ EOOD, e-mail: inikolova50@abv.bg

Teodor Kalushkov – Senior Lecturer, PhD, Faculty of Mathematics and Informatics, St. Cyril and St. Methodius University of Veliko Tarnovo, e-mail: t.kalushkov@ts.uni-vt.bg