

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»**

Спеціальність: 123 «Комп'ютерна інженерія»

Освітньо-професійна програма: «Безпека комп'ютерних систем і мереж»

Група: 4КБ-01

Дипломний проект

здобувача освіти денної форми навчання

КБ.01.15.000.ДП

САГАЙДАК

КИРИЛА ВОЛОДИМИРОВИЧА

**м. Одеса
2024 р.**

Спеціальність: 123 «Комп'ютерна інженерія»

Освітньо-професійна програма: «Безпека комп'ютерних систем і мереж»

Група: 4КБ-01

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту на тему:

Розробка системи віддаленого контролю життєвих показників з захистом даних на рівні TLS

Проектний матеріал складається з пояснювальної записки на 75 сторінках та графічного (презентаційного) матеріалу на 15 аркушах (слайдах)

Дипломник _____ (Сагайдак К.В.)
Керівник _____ (Шувалова І.О.)

Консультанти:

з економічного розділу _____ (Іванченков В.С.)
з розділу охорони праці та техніки безпеки _____ (Чорновол Н.І.)
з нормоконтролю _____ (Петрашова В.І.)
старший консультант _____ (Кривченко Ю.В.)

До захисту допущений

Голова циклової комісії _____ (Кривченко Ю.В.)
Завідувач відділення _____ (Скорнякова О.В.)

Захист «12» 06 2024 р. Протокол ЕК № 1

Оцінка ЕК 5 (відмінно) 90%

Секретар ЕК _____

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Відділення комп'ютерних систем Комісія КТ та ПІ
Спеціальність 123 «Комп'ютерна інженерія»
Освітньо-професійна програма «Безпека комп'ютерних систем і мереж»

ЗАТВЕРДЖУЮ:
Заст. дир. з НВР Беркань І.В.
"15" 01 2024 р.

ЗАВДАННЯ

на дипломний проект

Здобувачеві освіти Сагайдак Кирилові Володимировичу
(прізвище, ім'я, по батькові)

1. Тема проекту Розробка системи віддаленого контролю життєвих показників з захистом даних на рівні TLS

затверджена наказом по коледжу від "02" листопада 2023р. № 244-А2-0-В




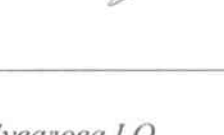

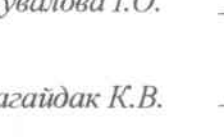

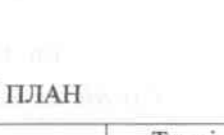
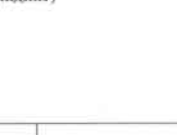
2. Термін здачі закінченого проекту 10.06.24р.

3. Вихідні дані до проекту Реалізувати вимірювання температури тіла у діапазоні 34÷42°C та пульсу у діапазоні 30÷220bpm; Використовувати апаратну платформу Arduino; Вимірювати серцебиття імпульсним датчиком на основі світлодіоду; температуру – цифровим датчиком DS18B20; Відправляти дані вимірювання за каналом Wi-Fi на мобільний телефон або комп'ютер; Забезпечити захищену передачу даних на рівні TLS (SSL-з'єднання); Середовище розробки програмного забезпечення – Arduino CC IDE

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити)
Огляд існуючих аналогів системи контролю життєвих показників;
Аналіз та вибір засобів розробки системи віддаленого контролю життєвих показників;
Моделювання та конструювання системи віддаленого контролю життєвих показників;
Розробка блок-схем алгоритмів програми; Розробка ПЗ віддаленого контролю життєвих показників та його тестування; Економічні розрахунки; Заходи ТБ

5. Перелік графічного (презентаційного) матеріалу (з точним зазначенням обов'язкових креслень, кількості слайдів)
Загальний принцип роботи датчика пульсу; Структурна схема системи віддаленого контролю життєвих показників; Склад плати Arduino Nano; Схема підключення датчиків до плати Arduino; Блок-схема алгоритму роботи системи віддаленого контролю життєвих показників з захистом даних на рівні TLS; Дані у таблиці Google Sheets на Google-диску; Сценарій використання функцій системи віддаленого контролю життєвих показників з застосуванням Google-диска

6. Консультанти по проекту, із зазначенням розділів проекту, що їх стосується

| Розділ | Консультант | Підпис, дата | |
|----------------------|-----------------|--|---|
| | | Завдання видав | Завдання прийняв |
| Основний розділ | Шувалова І.О. |  |  |
| Економічний розділ | Іванченков В.С. |  |  |
| Розділ охорони праці | Чорновол Н.І. |  |  |
| Нормоконтроль | Петрашова В.І. |  |  |
| Старший консультант | Кривченко Ю.В. |  |  |

7. Дата видачі завдання 15.01.24

Керівник Шувалова І.О.

(підпис)

Завдання прийняв до виконання

Сагайдак К.В.

(підпис)

КАЛЕНДАРНИЙ ПЛАН

| № з/р | Назва етапів дипломного проекту | Термін виконання етапів дипломного проекту (роботи) | Відмітка про виконання |
|-------|---|---|------------------------|
| 1 | Вступ. Постановка задачі проектування | 28.04.24 | Виконав |
| 2 | Огляд методів отримання даних температури та тиску | 1.05.24 | Виконав |
| 3 | Огляд методів обробки інформації датчиків | 3.05.24 | Виконав |
| 4 | Огляд існуючих систем моніторингу | 5.05.24 | Виконав |
| 5 | Вибір апаратного забезпечення проекту | 8.05.24 | Виконав |
| 6 | Вибір технології та програмних засобів розробки | 11.05.24 | Виконав |
| 7 | Розробка структури програмно-апаратних засобів | 14.05.24 | Виконав |
| 8 | Опис структури проекту та підключення датчиків | 17.05.24 | Виконав |
| 9 | Розробка блок-схеми алгоритму проекту | 20.05.24 | Виконав |
| 10 | Розробка програмного забезпечення проекту | 23.05.24 | Виконав |
| 11 | Реалізація моделі проекту та її випробування | 26.05.24 | Виконав |
| 12 | Випробування додатку та аналіз результатів | 29.05.24 | Виконав |
| 13 | Розробка титань з економіки | 3.06.24 | Виконав |
| 14 | Розробка титань з охорони праці | 7.06.24 | Виконав |
| 15 | Підготовка креслень, тексту ПЗ, підготовка до захисту | 9.06.24 | Виконав |

Дипломник



(підпис)

Керівник



(підпис)

ЗМІСТ

| | |
|---|----|
| Вступ..... | 7 |
| 1 Основний розділ..... | 8 |
| 1.1 Огляд існуючих систем контролю життєвих показників..... | 8 |
| 1.1.1 Фітнес-браслети..... | 8 |
| 1.1.2 Смарт-годинники..... | 14 |
| 1.1.3 «Розумний» одяг..... | 16 |
| 1.2 Аналіз технічного завдання на розробку..... | 18 |
| 1.3 Вибір версії платформи Arduino..... | 18 |
| 1.4 Аналіз складових платформи Arduino Nano..... | 19 |
| 1.5 Вибір та опис складових компонентів системи віддаленого контролю..... | 24 |
| 1.5.1 Датчик температури DS18B20..... | 24 |
| 1.5.2 Датчик пульсу Pulse sensor..... | 25 |
| 1.5.3 Wi-Fi-модуль NodeMCU V3 ESP8266..... | 27 |
| 1.6 Вибір середовища розробки програмного забезпечення..... | 29 |
| 1.6.1 Інтегроване середовище розробки Programino..... | 30 |
| 1.6.2 Інтегроване середовище розробки B4R (Basic for Arduino)..... | 30 |
| 1.6.3 Інтегроване середовище розробки Codeblocks for Arduino..... | 31 |
| 1.6.4 Інтегроване середовище розробки Arduino IDE..... | 33 |
| 1.7 Розробка схеми підключення датчиків та Wi-Fi-модулю..... | 34 |
| 1.8 Розробка програмного забезпечення системи віддаленого контролю..... | 36 |
| 1.8.1 Програмування мікроконтролера Arduino Nano..... | 37 |
| 1.8.2 Програмування Wi-Fi-модулю ESP8266 NodeMCU..... | 42 |
| 1.8.3 Налаштування сервісу Google-диска..... | 49 |
| 1.9 Експлуатація системи віддаленого контролю життєвих показників..... | 51 |
| 2 Економічний розділ..... | 55 |
| 3 Розділ охорони праці та техніки безпеки..... | 60 |
| 3.1 Вступ..... | 60 |

| | |
|---|----|
| 3.2 Аналіз небезпечних та шкідливих чинників, що впливають на працівника..... | 60 |
| 3.3 Розробка заходів з охорони праці..... | 61 |
| 3.3.1 Виробничі приміщення..... | 61 |
| 3.3.2 Мікроклімат робочої зони працівників, вентиляція..... | 61 |
| 3.3.3 Освітлення робочого місця, шум, вібрація..... | 62 |
| 3.3.4 Організація робочого місця користувача ПК..... | 62 |
| 3.3.5 Електробезпека..... | 63 |
| 3.4 Пожежна безпека..... | 64 |
| Висновки..... | 65 |
| Перелік використаних інформаційних джерел..... | 66 |
| Додаток А. Лістинг програми прийняття даних від сенсорів для Arduino Nano..... | 67 |
| Додаток Б. Слайди мультимедійної презентації..... | 68 |

ВСТУП

Температура тіла і частота серцевих скорочень є одними із найважливіших значень стану здоров'я користувача. Цей показник застосовується в медичній і спортивній сфері, чи ж просто в побуті. Нормальною (середньою) частотою серцевих скорочень в стані спокою вважається приблизно 60-80 ударів поза хвилину. Слід зазначити, що частота серцевих скорочень залежить з віку користувача, а так само її статі і можливих захворювань. В жінок частота серцевих скорочень на 5-10 ударів менша ніж в чоловіків.

В такій ситуації можливо самостійно використовувати спеціальні діагностичні пристрої – людина спроможне самостійно виміряти показники серцевих скорочень і температуру свого тіла. На Гугл-диску інформація у захищеній формі передбачає представлені в вже обробленому виді, тож лікар не повинен буде витратити період на їх розшифрування і складний аналіз.

В ситуації, коли людина вже знаходиться в лікарні і потребує нагляду, медичний персонал період з періоду перевіряє стан пацієнтів. Використання спеціальних діагностичних пристроїв спроможне пришвидшити і оптимізувати його роботу, оскільки життєві показники пацієнта одразу передбачає надходити до комп'ютера чергового медичного працівника. Доступ до них зможе отримати лікар, якому знадобилася медична карта пацієнта, та переглянути результати вимірювань поза весь період нагляду поза пацієнтом, а не тільки ті, котрі надійшли щойно.

Так само подібні пристрої спроможні використовувати спортсмени, аби контролювати свій пульс під період тренувань чи опісля них, чи будь-котрі люди задля буденного стеження поза показниками організму. Інша можлива ситуація – коли люди містять родичів похилого віку, вони спроможні спостерігати на відстані поза їхніми життєвими показниками та вчасно помітити відхилення.

В дипломному проекті виконується розробка системи дистанційного перевірки біологічних значень із захистом інформації на позначці TLS, котра спроможне попередити пацієнтів до того, як стан їх здоров'я критично погіршиться.

| | | | | | | |
|-----|------|----------|--------|------|--------------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 7 |

1 ОСНОВНИЙ РОЗДІЛ

1.1 Огляд існуючих систем перевірки біологічних значень

При розробці нових «розумних» медичних пристроїв виробники намагаються додати до них якомога більше додаткових функцій, що у свою чергу призводить до створення багатофункціональних пристроїв. В зв'язку із цим швидким прогресом стає дедалі складніше віднести ці пристрої до якоїсь певної категорії. Поза поміччю них користувач спроможне не тільки проводити моніторинг власного стану здоров'я, а й відправляти ці інформація лікарю. Повсюдне поширення IoT (Інтернету Речей) передбачає підмикання медичних приладів до мережі Інтернет задля відправки інформації перевірки на хмарні сервіси. В цьому розділі передбачає наведені приклади «розумних» побутових пристроїв перевірки.

1.1.1 Фітнес-браслети

Фітнес-наручник представляє собою прилад компактного розміру, що вдягається на руку і призначений задля перевірки фізичної активності користувача. У даний період виробництвом таких пристроїв займається велика число виробників електроніки, таких як: Jawbone, Garmin, Fitbit, Xiaomi, Samsung, Misfit та інші.

Подібні фітнес-браслети здатні відстежувати життєві показники користувача (є фітнес-трекерами) та містять різні процедури, побудову і цілі. Основною задачею фітнес-трекеру є мотивація свого власника на активну діяльність та контроль одержуваного навантаження. Задля виконання поставлених задач у подібних пристроях встановлені датчики, котрі фіксують інформація пульса, кількості пройдених кроків, витрачених калорій, значення стресу, якості сну, швидкості переміщення та довжини пройденої відстані. Отримана інформація спроможне передаватися на смартфон чи комп'ютер, а використання спеціально розробленого додатку дозволить зробити розрахунки активності користувача, зміни значень здоров'я та при необхідності дасть

| | | | | | | |
|-----|------|----------|--------|------|-------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 8 |

тиждень. Так само наручник буде нагадувати про необхідність рухатися задля того, аби допомогти користувачеві залишатися активним протягом всього дня – він буде сповіщати і просити робити 250 кроків кожен годину. Фітнес-наручник містить велику число вбудованих датчиків (рис.1.2), котрі спроможні фіксувати стан користувача. Він спроможне контролювати сон, слідкувати поза його якістю і тривалістю. Цим він спроможне допомогти виявити деякі розлади, коли вони є, і виробити правильні звички, що значно покращить самопочуття користувача.



Рисунок 1.2. Датчики в фітнес-браслеті Fitbit Charge 2

Трекер вимірює пульс і додаткові показники, завдяки чому можливо дізнатися не тільки про частоту серцевого ритму але та про максимальну число кисню, яку тіло спроможне використовувати під період сильних навантажень. Коли значення пульса передбачає виходити поза допустимі межі, наручник попередить про небезпеку. Задля керування браслетом і відображення всіх значень розроблений спеціальній додаток, що встановлюється на смартфон. В ньому можливо не тільки переглянути інформація, а й побудувати зрозумілі графіки задля наочного представлення значень [2].

| | | | | | | |
|-----|------|----------|--------|------|-------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 10 |

Фітнес-трекер Hesvitband S3 (рис. 1.3) містить форму браслета із вбудованими у нього датчиками та монітором задля виведення значень (так само результати можливо переглядати на смартфоні).



Рисунок 1.3. Фітнес-наручник Hesvitband S3

Цей наручник володіє безліччю різноманітних функцій:

- наручник вміє відображати на своєму екрані поточний період – задля цього він автоматично синхронізується із смартфоном. При цьому всі потрібні інформація зберігаються протягом 10 днів;
- завдяки вбудованому датчику наручник буде вимірювати частоту серцебиття у процесі навантажень чи відпочинку. Коли її значення складають більше допустимих значень (60-100 ударів у хвилину), тоді його екран почне мерехтати. При необхідності можливо запустити відстеження вручну, натиснувши на відповідну кнопку;
- наручник буде надавати інформацію щодо температури навколо зап'ястя, котра значно відрізняється з температури навколишнього середовища та спроможне сигналізувати про зміни здоров'я користувача.
- під період пробіжок чи ходьби наручник автоматично вимірює число кроків, виводячи загальні результати на дисплеї. Кожна доросла людина

містить проходити щодня близько 10 тисяч кроків, тому наручник буде мотивувати користувача на регулярні прогулянки;

- протягом усього дня, під період тренування та відпочинку наручник буде автоматично прораховувати число спалених калорій. Чим більше тренуватися, тим більше калорій спалиться, тим самим зміцнюється організм, зменшується ймовірність появи хронічних захворювань та появи зайвої ваги;
- наручник буде надавати контролювати процес сну. Аби бути у тонусі цілий день, кожній дорослій людині необхідний повноцінний 6-годинний відпочинок. Наручник зможе вимірювати тривалість сну, його фази та загальну продуктивність, показуючи наскільки організм відпочив минулої ночі;



Рисунок 1.4. Додаток задля Android-смартфону фітнес-браслету Hesvitband S3

- завдяки вбудованому датчику наручник буде вимірювати вологість шкіри. Вона залежить з її параметрів, навколишнього оточення, а так само з того, чим займається користувач у даний момент. Наприклад, при довгому бігу чи фізичних навантаженнях в спекотну погоду її рівень спроможне істотно

| | | | | | | |
|-----|------|----------|--------|------|-------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 12 |

знизитися, тому дуже важливо підтримувати її на позначці 50-70%. При недостатній вологості шкіра стає сухою та починає лущитися, а при надмірній – швидко втрачає воду (потовиділення). Nesvit S3 вміє відслідковувати ці найменші зміни, та відображати їх на своєму дисплеї.

Задля відображення головної інформації наручник містить додаток задля смартфона (рис.1.4). Окрім інформації по активності користувача, котра відображається в виді зручних та зрозумілих графіків, в додатку можливо вести свій профіль, де буде відображатися історія змін стану здоров'я та інша особиста інформація [1].

Компанія Partron створила фітнес-наручник (рис.1.5), що позиціонується як прилад задля міського мешканця. Цей наручник спроможне не тільки вимірювати звичайні показники активності користувача (пройдену відстань, калорії, тощо), але та давати поради щодо поліпшення свого стану, вимірювати стрес та якість сну.



Рисунок 1.5. Розумний наручник Partron PWB-100

Наручник Partron PWB-100 компілює всю зібрану інформацію, аналізує ці інформація та видає загальну оцінку стану організму користувача – враховується та рівень щоденної активності та стрес. Прилад містить наступні технічні характеристики:

| | | | | | | |
|-----|------|----------|--------|------|-------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 13 |

- дисплей OLED, 128 * 36 (відображається базова інформація);
- Корейська, Англійська, Китайська мови;
- бездротовий зв'язок Bluetooth 4.0;
- 12 годин роботи в автономному режимі із усіма увімкненими функціями;
- 4 дні роботи в режимі очікування;
- Сумісність із Android, iOS;
- модулі акселерометру, гіроскопу, датчику температури, барометру, пульсометру.

Прилад надає користувачеві наступні можливості:

- особистого тренера;
- визначення кількості пройдених кроків;
- визначення пройденої відстані;
- визначення пульсу;
- визначення температура тіла;
- визначення загального періоду активності;
- визначення періоду та якості сну;
- визначення висоти над рівнем моря;
- встановлення цілей по дистанції поза день, рівню спалених калорій, періоду сну, тощо.

Наявність датчику температури тіла – позитивна сторона приладу, адже вбудованим «градусником» спроможні похвалитися небагато браслетів і розумних годинників.

1.1.2 Смарт-годинники

Сьогодні дуже популярними серед новинок на ринку комп'ютерної техніки є смарт-годинники. Розумні годинники допомагають полегшити щоденне життя користувача, слідкувати поза шаленим ритмом життя, відстежувати власний стан здоров'я та поєднувати різні види занять.

На сучасному ринку представлені фітнес-браслети (пункт 1.1), котрі виконують деякі схожі із смарт-годинниками процедури.

| | | | | | | |
|-----|------|----------|--------|------|--------------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 14 |

Досить популярними серед користувачів є смарт-годинники FOSSIL 4 Gen Steel Mesh (рис.1.6).



Рисунок 1.6. Смарт-годинник FOSSIL 4 Gen Steel Mesh

Пам'ять у 4 Гб передбачає зберігати у пам'яті годинника всі необхідні файли, музику, документи тощо. Смарт-годинники FOSSIL 4 Gen Steel Mesh працюють 2-3 дні без підмикання до електричної енергії, дозволяють здійснювати дзвінки та на відстані контролювати справи бізнесу, поєднувати їх із заняттями спортом тощо. Містять вбудований пульсометр.

Такими ж популярними в користувачів є розумні годинники garmin (рис.1.7). Розумні годинники garmin призначені задля перевірки поза спортивним способом життя. Смарт-годинники цієї фірми обладнані корисними функціями:

- відстеження бігового стилю користувача;
- визначення значення стресу організму;
- зберігання історії тренувань поза конкретний період періоду;
- визначення глибини занурення під воду;
- визначення фази сну користувача;
- визначення кількості спалених калорій;
- визначення значення кисню у крові користувача.



Рисунок 1.7. Смарт-годинник Garmin Forerunner 245 Music Aqua

Крім перелічених функцій, годинники цієї марки здійснюють так само нагляд поза спортивним віком користувача, показують актуальний прогноз погоди тощо. Ці годинники показують звіти поза певний період користування пристроєм. Поза поміччю таких звітів можливо збудувати статистику щодо власних занять спортом, ефективності своєї роботи під період спортивних занять тощо.

1.1.3 «Розумний» одяг

Окрім фітнес-трекерів і смарт-годинників існують та інші пристрої, котрі спроможні відслідковувати основні життєві показники користувача. Саме таку функцію та виконує «розумний» одяг, котрий поєднано із сучасними інформаційними технологіями. Зараз «розумний» одяг знаходить широке розповсюдження в сферах медицини, спорту чи навіть в військовій сфері.

Цей аксесуар спроможне допомогти людям, що містять фізіологічні вади чи захворювання, а так само допомагає стежити поза станом людей, що працюють із небезпечними речовинами. Такий одяг передбачає проводити контроль основних біологічних значень, таких як частота пульса, частота дихання, температура тіла і інші, проводити віддалено аналізи і здійснювати дистанційні медичні консультації.

«Розумний» одяг спроможне допомогти людині вижити у екстремальних умовах. Поза його поміччю можливо відслідковувати стан користувача і її

місцезнаходження, проводити контроль значення втоми водія чи пілота літака.

BioSerenity WEMU – це приклад реалізації медичного «розумного одягу». Він створений задля хворих епілепсією (рис.1.8). Зазвичай моніторинг стану подібних хворих здійснюється у спеціалізованих лікарнях, обладнаних стаціонарними ЕКГ-системами.



Рисунок 1.8. “Розумний” одяг BioSerenity WEMU

Розробники BioSerenity разом із організаціями по дослідженню епілепсії пропонують використовувати комплект «розумного» одягу, в котрий входить кофта і шапка із вбудованими датчиками ЕКГ, ЕЕГ і ЕМГ. В деяких випадках розробити правильний курс, що підійде пацієнту, без подібної інформації практично неможливо.

BioSerenity WEMU використовує запатентовану технологію сухих давачів в поєднанні із гнучкими електронними схемами, вбудованими у тканину задля того, аби створити зручну задля носіння повсякденну систему перевірки. Прототип такого одягу був розроблений поза участі лікарів. Інформація не тільки зберігається локально в пам’яті самого приладу, але й, коли пацієнт дасть свою згоду, завантажується в спеціальний хмарний сервіс, до якого містить доступ лише лікар хворого і спеціальні дослідницькі організації, що допомагають розробляти індивідуальний курс лікування [3].

1.2 Аналіз технічного завдання на розробку

Технічним завданням на дипломне проектування передбачена розробка системи дистанційного перевірки біологічних значень із захистом інформації на позначці TLS, призначеної задля перевірки температури і пульса користувача і передачі інформації в захищеному виді (TLS-із'єднання) на Гугл-диск.

Розроблюваний прилад містить вимірювати температуру тіла і пульс користувача цифровим методом. Основу приладу буде складає модуль Arduino Нано на платформі Arduino, котрий буде контролювати роботу датчиків і передавати інформацію на модуль Node-mcu, створений на основі мікроконтролеру ESP8285. Датчики, котрі передбачає підключені до Arduino Нано – цифровий датчик температури DS18D30, і імпульсний датчик серцевого ритму Pulse Sensor. Поза поміччю вбудованого в плату Node-mcu модулю WI-FI, вона буде передавати інформація із датчиків на Гугл-диск у режимі реального періоду в захищеному виді, де можливо буде переглядати всі показники температури і пульса і бачити, в котрий проміжок періоду вони були зроблені. TLS-із'єднання (SSL) встановлює зв'язок між клієнтом і сервером поза однойменним протоколом, що забезпечує шифрування і безпечний обмін інформацією. Це не дозволить зловмисникам отримати доступ до інформації користувачів приладу.

1.3 Вибір версії платформи Arduino

Різні версії плат Arduino є однаковими поза принципом своєї роботи, тобто при реалізації проекту можливо замінити одну плату на іншу поза умови відповідного обсягу пам'яті мікроконтролера і кількості виводів на платі. Частіше поза все користувачі використовують платформи Uno і Нано чи плату Leonardo, коли в цьому виникне необхідність.

Головна відмінність між платами Uno і Нано – це їх розмір. Arduino Uno на декілька сантиметрів більша поза Нано по ширині і довжині. Цим способом платформа Uno займає більше місця, що спроможне стати обмеженням у деяких проектах. Задля портативного приладу, такого як наручник на руці, така

| | | | | | | |
|-----|------|----------|--------|------|-------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 18 |

платформа буде занадто велика. Платформи Uno, у свою чергу, є ідеальним рішенням задля таких проектів, оскільки платформа спроможне поміститися в невеликий корпус і не буде псувати зовнішній вигляд приладу. Спільним в цих двох плат є МК ATmega328p, котрий встановлений на обидві із них. Цим способом, вони спроможні ділитися схожою програмою. Це стане в нагоді при переході із платформи Arduino Uno до платформи Arduino Nano – не доведеться переписувати код і підлаштовуватися під іншу систему. На платах Arduino Uno застосовується звичайний роз'єм USB (TYPE-B), у той період як Arduino Nano містить на своїй платі роз'єм міні-USB. Обидва стандарти підключень і відповідні кабелі є широко розповсюдженими.

Arduino Leonardo відрізняється з попередніх плат тим, що МК ATmega32u4, на якому він працює, містить вбудовану підтримку USB, що не потребує додаткового чіпу процесора, на котрий ATmega328 опирається задля передачі інформації USB. Це передбачає Leonardo відображатися на під'єднаному комп'ютері в виді миші чи клавіатури, на додачу до віртуального COM-порту.

В даному проекті задля реалізації приладу перевірки температури і пульса було вирішено використовувати плату Arduino Nano, котра виділяється серед інших плат своїм малим розміром, що особливо важливо задля даної специфіки.

1.4 Аналіз складових платформи Arduino Nano

Arduino Nano – невелика, сумісна і гнучка до макету платформа мікроконтролера, створена на основі мікросхеми ATmega328p чи ATmega168. Arduino Nano входить до трійки лідерів поза популярністю серед радіоаматорів-програмістів. Незважаючи на свій скромний розмір, вона практично нічим не поступається повноформатній версії Arduino Uno по функціоналу та спроможне використовуватися в проектах, де габарити відіграють істотну роль. Ранні версії Arduino Nano базувалися на основі мікроконтролера ATmega168. Починаючи із версії 3.0 у них встановлені більш просунуті ATmega328, зі збільшеним об'ємом

| | | | | | | |
|-----|------|----------|--------|------|--------------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 19 |

FLASH та EEPROM-пам'яті, а так само із більшою тактовою частотою. На рис.1.9 показаний зовнішній вигляд платформи Arduino Nano V3.0.

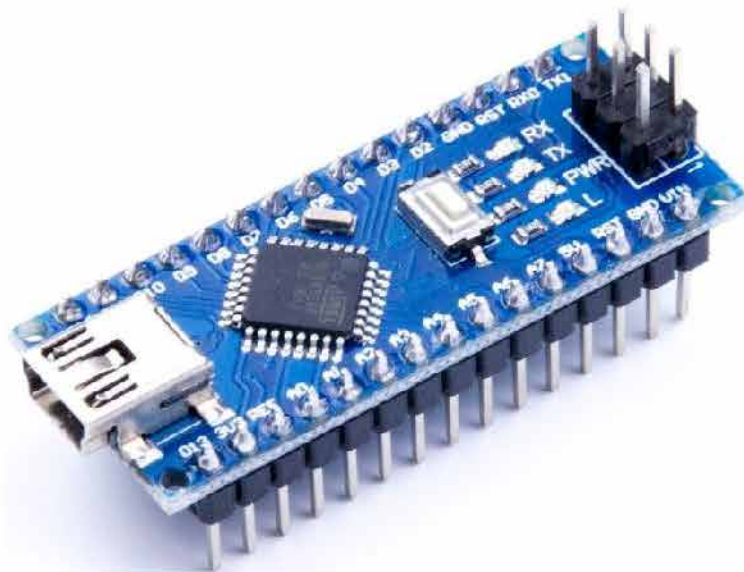


Рисунок 1.9. Платформа Arduino Нано

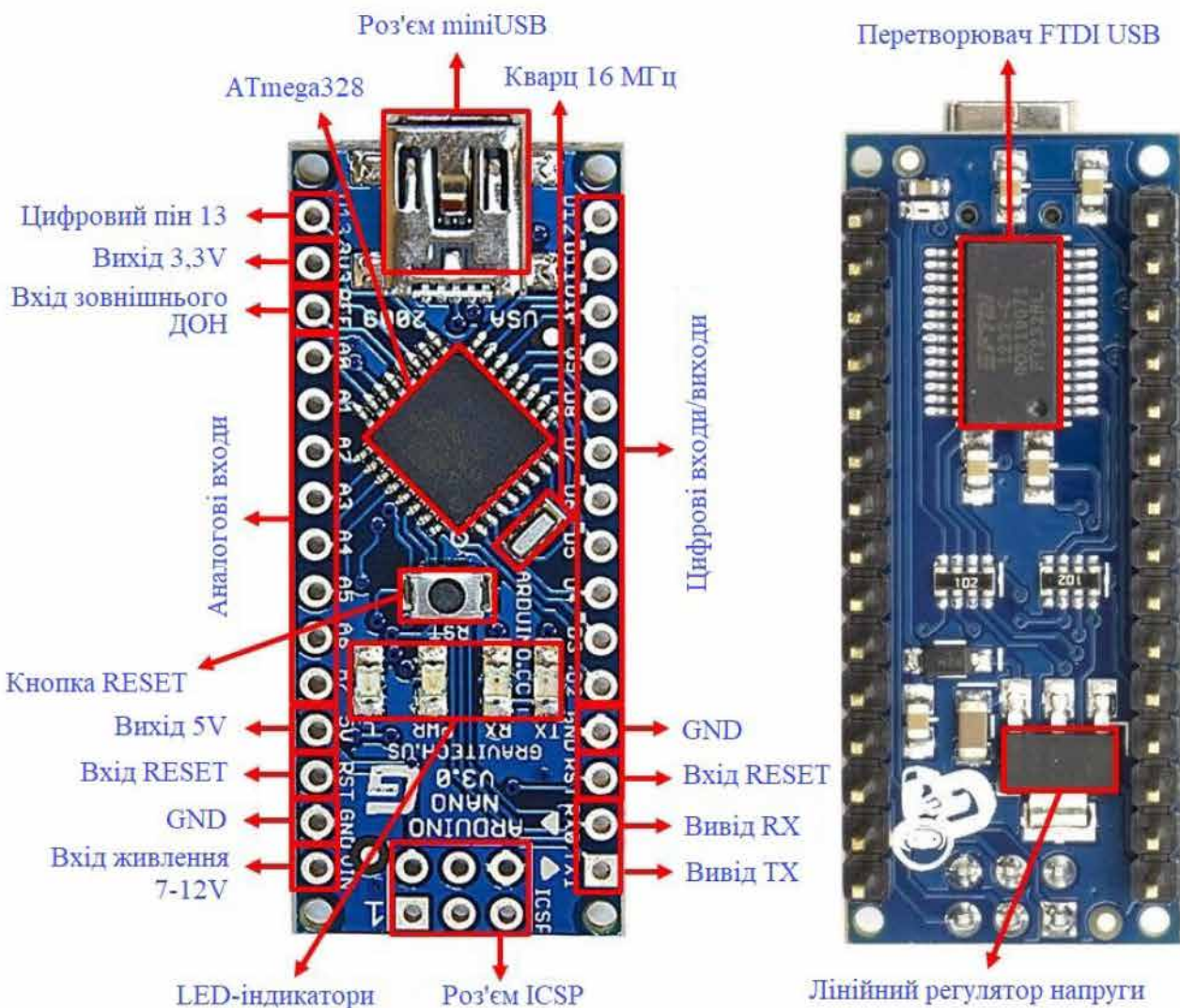


Рисунок 1.10. Склад платформи Arduino Нано

| | | | | |
|-----|------|----------|--------|------|
| | | | | |
| Зм. | Арк. | № докум. | Підпис | Дата |

КБ 01. 15 000. 00 ДП ПЗ

Арк.

20

Контактна панель Arduino Nano Pinout містить 14 цифрових виводів, 8 аналогових виводів, 2 скидання і 6 виводів живлення.

Vin – це вхідна напруга живлення задля платформи при використанні зовнішнього джерела живлення з 7 до 12 У.

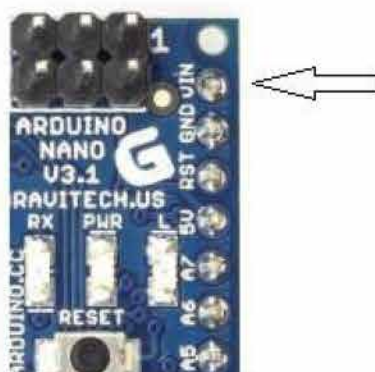


Рисунок 1.11. Виводи вхідної напруги живлення Vin

5V – це регульована напруга живлення платформи, котра застосовується задля живлення контролера і інших компонентів, розташованих на платі.

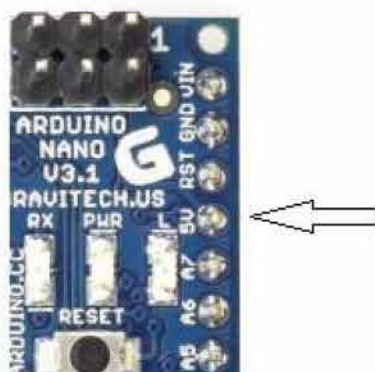


Рисунок 1.12. Виводи напруги живлення 5V

3.3V – це мінімальна напруга, що генерується регулятором напруги на платі.

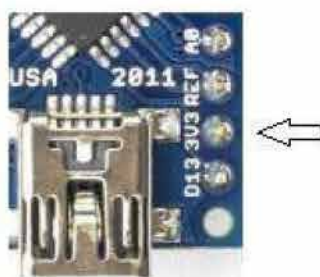


Рисунок 1.13. Виводи напруги живлення 3.3V

Rx, Tx – це виводи задля послідовного зв'язку, де Tx надає передачу інформації, а Rx – прийняття інформації.

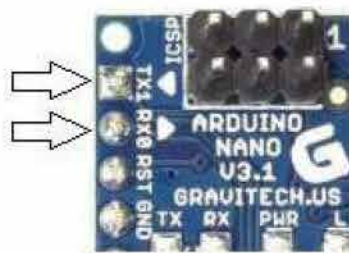


Рисунок 1.17. Виводи RXD, TXD

13 – це виводи задля увімкнення вбудованого світлодіоду.

AREF – виводи використовуються в якості джерела опорної напруги задля вхідної напруги.

PWM – виводи 3,5,6,9,10,11, що спроможні використовуватися задля забезпечення 8-канального виходу ШІМ (широкоімпульсної модуляції), котра застосовується задля отримання аналогових значень із цифровим джерелом.

SPI – виводи 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK), що використовуються задля SPI (послідовний периферійний інтерфейс). SPI є інтерфейсною шиною і у основному застосовується задля передачі інформації між мікроконтролерами і іншими периферійними пристроями, такими як датчики, регістри і SD-карти.

External Interrupts – виводи 2 і 3, що використовуються як зовнішні переривання в разі надзвичайної ситуації, коли треба зупинити основну програму і викликати важливі інструкції в цей момент. Основна програма поновлюється опісля виклику і виконання інструкції переривання.

I2C – зв'язок, виконаний із використанням виводів A4 і A5, де A4 представляє собою лінію послідовних інформації (SDA), а A5 представляє собою лінію послідовних тактових сигналів (SCL), що генеруються ведучим приладом задля синхронізації інформації між пристроями на шині I2C [5].

1.5 Вибір і опис складових компонентів системи дистанційного перевірки

В проєкті системи дистанційного перевірки температури і пульса, окрім платформи Arduino Нано, передбачає застосовується декілька давачів і модуль WI-FI. Давач температури представляє собою цифровий вимірювач, в якості якого обрано давач DS18D30. В якості датчику визначення пульса користувача обрано аналоговий давач Pulse sensor. В якості WI-FI модуля була обрана платформа Node-mcu на основі ESP8285. Модуль вмiє отримувати і передавати інформацію крізь мережу Інтернет в захищеному виді крізь TLS-із'єднання. Розглянемо детальніше складові приладу перевірки.

1.5.1 Давач температури DS18D30

Елемент Dallas DS18D30 представляє собою цифровий давач, що забезпечує визначення температури в градусах Цельсія із роздільною здатністю 9-12 біт, містить тривожну функцію, котра буде сигналізувати, коли температура вийде поза верхню чи нижню межу, котрі встановлює сам користувач. Діапазон вимірюваних датчиком температур складає з -55°C до 125°C із точністю $\pm 0.5^{\circ}\text{C}$ в інтервалі з -10°C до $+85^{\circ}\text{C}$. Окрім цього, він спроможне отримувати живлення напряду із лінії інформації, що прибирає необхідність в зовнішньому джерелі живлення. Так само Dallas DS18D30 спроможне працювати в режимі «паразитного живлення», коли живлення приладу відбувається напругою деяких паразитних сигналів в лінії. Кожен екземпляр Dallas DS18D30 містить свій унікальний 64-розрядний номер, котрий застосовується задля того, аби дозволити багатьом датчикам працювати на одній лінії. Це передбачає одному мікропроцесору керувати багатьма датчиками Dallas DS18D30, котрі розташовані на великій площі. Цим можливо користуватися задля вимірювати температуру в будівлі, механізмів машини чи ж в палаті лікарні, коли декількох пацієнтів під'єднати до одного приладу.

На рис. 1.18 показано, що давач містить три виводи:

– GND – земля;

| | | | | | | |
|-----|------|----------|--------|------|--------------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 24 |

- DQ – вивід сигналу інформації. Так само крізь цей вивід відбувається живлення в режимі «паразитного живлення»;
- VDD – вивід зовнішнього живлення. В разі, коли застосовується «паразитне живлення», містить бути підключений до «землі».

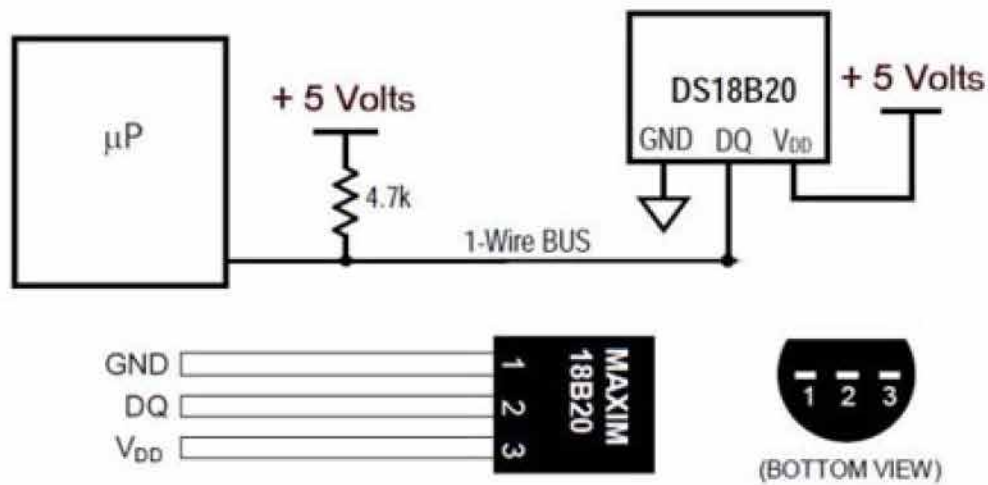


Рисунок 1.18. Призначення виводів і типове підмикання давача Dallas DS18D30

Не дивлячись на тип підмикання, інформаційний вивід DQ слід під'єднувати до виводу живлення крізь резистор 4.7 кОм. В разі, коли застосовується лише один давач, можливо обрати резистор на 10 кОм [7].

1.5.2 Давач пульса Pulse sensor

Дачач Pulse sensor представляє собою імпульсний аналоговий давач, що передбачає зафіксувати пульс користувача. Цей давач працює поза рахунок встановлених на ньому світлодіоду і фотоприймачу. Вони розташовані цим способом, що на фотоприймач потрапляє промінь світла котрий відбивається з перешкоди, якою спроможне бути палець, вухо, чи зап'ястя. Принцип роботи можливо побачити на рис. 1.19. Коли судини наповнюються кров'ю, вони починають поступово змінювати свою оптичну щільність, що одразу впливає на зміну кількості відбитого світла. Коли рівень світлового потоку, котрий випромінюється світлодіодом, буде постійним, інтенсивність світла, котра спроможне реєструватися фотоприймачем, буде залежати з наповненості судин кров'ю [8].

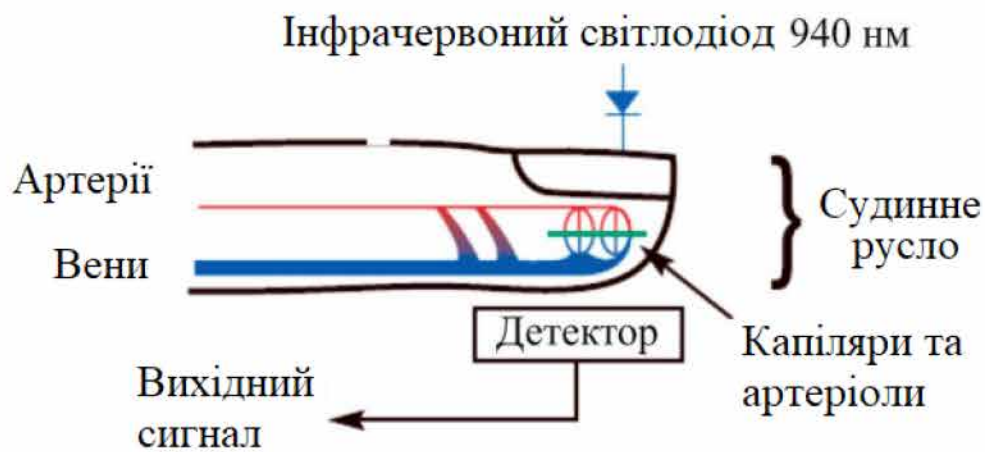


Рисунок 1.19. Загальний принцип роботи давача пульса

Електрична схема давача пульса розроблена спеціально, аби реєструвати тільки динамічну зміну інтенсивності світлового потоку, що приймається. Коли світловий потік є незмінним (незалежно з його інтенсивності), напруга на виході давача коливається у районі половини напруги живлення. В разі, коли відбувається зміна світлового потоку, напруга на виході давача відхиляється з середнього значення у бік зменшення чи збільшення, пропорційно зміні світлового потоку. Цим способом давач не потребує налаштування під кожну людину. Давач пульса містить розмір приблизно 16 міліметрів у діаметрі і близько 3 міліметрів завтовшки. Напруга живлення на датчику складає 5 В, а струм споживання – 4 мА.



Рисунок 1.20. Конструкція і виводи давача пульса Pulse Sensor

На рис.1.20 видно три виводи давача пульса, а саме:

- Signal – сигнальний вивід, що під’єднується до будь-якого аналогового контакту мікроконтролеру;
- VCC – живлення, з 3 до 5 У;
- GND – земля, що під’єднується до виводу GND мікроконтролера [9].

1.5.3 Wi-fi-блок Node-mcu V3 ESP8285

Модуль ESP8285 виробника Espressif Systems представляє собою Wi-Fi-мікрочіп, що включає у себе повний стек TCP/IP із захищеним TLS-із’єднанням (SSL).

Розглянемо детальніше складові приладу перевірки. Чіп спроектований задля пристроїв зі світу Інтернет Речей, а дана платформа передбачає спростити розробку, адже на ній вже реалізоване підмикання крізь порт USB, регулятор живлення, а всі виводи чіпу розведені зі стандартним кроком 2,54 мм (рис. 1.21). Платформа поставляється із прошивкою Node-mcu, котра передбачає програмувати її поза поміччю мови Lua чи поза поміччю Arduino ICP.

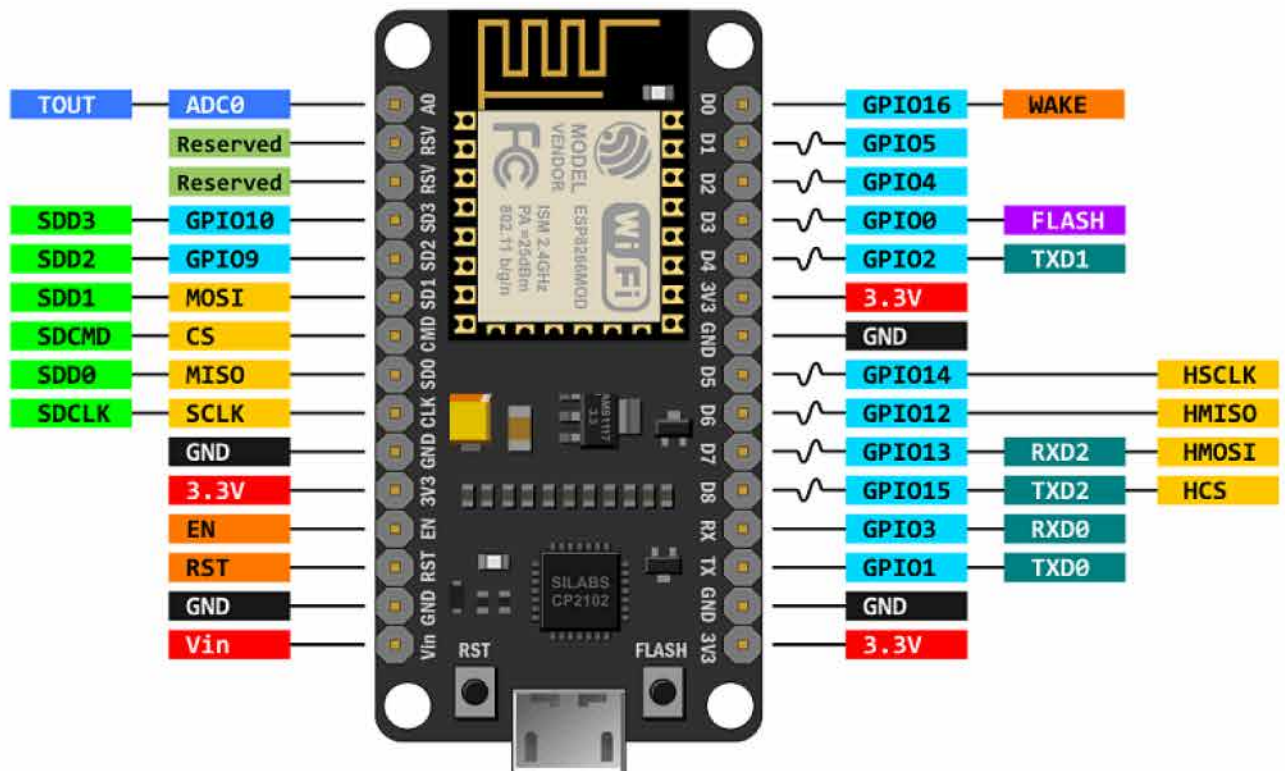


Рисунок 1.21. Конструкція і виводи Wi-fi-блоку ESP8285

Платформа містить такі характеристики:

- Wi-Fi стандарту 802.11 b / g / n;
- підтримка STA / AP / STA + AP режимів;
- струм на виводах 15 мА;
- живлення з 4,5 до 10 У макс.;
- швидкість передачі інформації 110-460800 бод поза секунду;
- діапазон робочих температур з -40 до +125 градусів по Цельсію;
- маса платформи 18 г [10].

На рис. 1.21 показані такі виводи:

- GND. На платі є 5 виводів GND (земля);
- виводи живлення. Vin – вивід задля підмикання зовнішнього джерела живлення 5 У. Стабілізатор AMS1117-3.3 передбачає подавати живлення на Vin в широкому діапазоні з 5 до 10 У. 3.3 У – контакт на котрий подається вихідна напруга схемного стабілізатора. Спроможне бути використаний задля живлення датчиків, що підключаються до платформи. Сумарне максимальне навантаження усіх виводів 3.3 У містить не перевищувати 300 мА. V USB – вивід, на котрий заведена напруга 5 У із USB-роз'єму;
- виводи GPIO (General Purpose Input Output) – виводи вводу/виводу загального призначення. Спроможні бути сконфігуровані як входи/виходи і програмно призначені на різні процедури;
- виводи керування. Reset – вивід застосовується задля скидання мікроконтролера ESP8285. EN – при подачі на контакт сигналу високого значення, МК ESP8285 переходить в робочий режим, а при сигналі низького значення – в режим енергозбереження. WAKE – контакт, що застосовується задля пробудження чіпу ESP8285 із режиму глибокого сну;
- АЦП (ADC). ADC0/TOUT – вивід вбудованого 10-розрядного аналого-цифрового перетворювача. Перетворені значення лежать у інтервалі 0-1023. Платформа розроблювання Node-mcu V3 поставляється із внутрішнім дільником напруги, тому вхідний діапазон становить з 0 до 3.3

| | | | | | | |
|-----|------|----------|--------|------|--------------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 28 |

- У. Діапазон вхідної напруги задля АЦП у кристалі ESP8285 з 0 до 1 У;
- UART. Асинхронний послідовний інтерфейс, що встановлює зв'язок із іншими пристроями по шині UART. В контролері ESP8285 вбудовані два модулі UART. Максимальна швидкість передачі інформації, що заявляється виробником 4,5 Mbps;
 - SPI. Послідовний периферійний інтерфейс Node-mcu містить два SPI (SPI і HSPI) в ведучому і підлеглому режимі;
 - SDIO. Інтерфейс безпечних цифрових входів/виходів, призначений задля комутації із флеш-пам'яттю стандарту SD по послідовній шині;
 - Reserved. Зарезервовані виводи;
 - FLASH. Кнопка Flash на Node-mcu підключає до землі GPIO0. Її можливо використовувати як звичайну кнопку. Коли програмно підтягнути вивід GPIO0 поза поміччю внутрішнього підтягуючого резистору до високого значення;
 - інтерфейс I2C. Послідовна асиметрична шина I2C застосовується задля підмикання датчиків і периферійних пристроїв. Node-mcu ESP8285 не містить апаратних виводів I2C, але інтерфейс можливо реалізувати програмно. Підтримуються як I2C Master, так й I2C Slave. Зазвичай в якості контактів I2C використовуються наступні виводи: GPIO5 – SCL, GPIO4 – SDA;
 - PWM (pulse-width modulation). Широтно-імпульсна модуляція (ШИМ) керує потужністю поза поміччю метода пульсуючого увімкнення і вимкнення виводу [11].

1.6 Вибір середовища програмного забезпечення

Існує велика число середовищ розроблювання, котрі дозволяють програмувати мікроконтролери. Кожне містить свої переваги та недоліки, різний функціонал та спосіб роботи. Розглянемо декілька популярних середовищ розроблювання і виберемо найдоречніший варіант задля реалізації програмного забезпечення.

| | | | | | | |
|-----|------|----------|--------|------|--------------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 29 |

1.6.1 Інтегроване середовище розроблення Programino

Programino – це платне інтегроване середовище розроблення, але його можливо випробувати протягом 14-ти днів безкоштовно. Programino (рис.1.22), як та інші середовища розроблення потребує встановленого середовища Arduino ICP. При першому запуску програми треба у налаштуваннях вказати шлях до виконуваного файлу `arduino.exe`. Мова, що застосовується в цьому середовищі, така сама як та у оригінальній Arduino ICP – це мова C. Отже, поза наявності навичок написання скетчів у Arduino ICP, не доведеться вивчати нову мову програмування, що є великим плюсом даного середовища розроблення. Орім цього, дане ICP пропонує такий зручний спосіб швидкої розроблення, як автодоповнення кода. При початку набирання кода середовище розроблення запропонує обрати один із декількох варіантів.

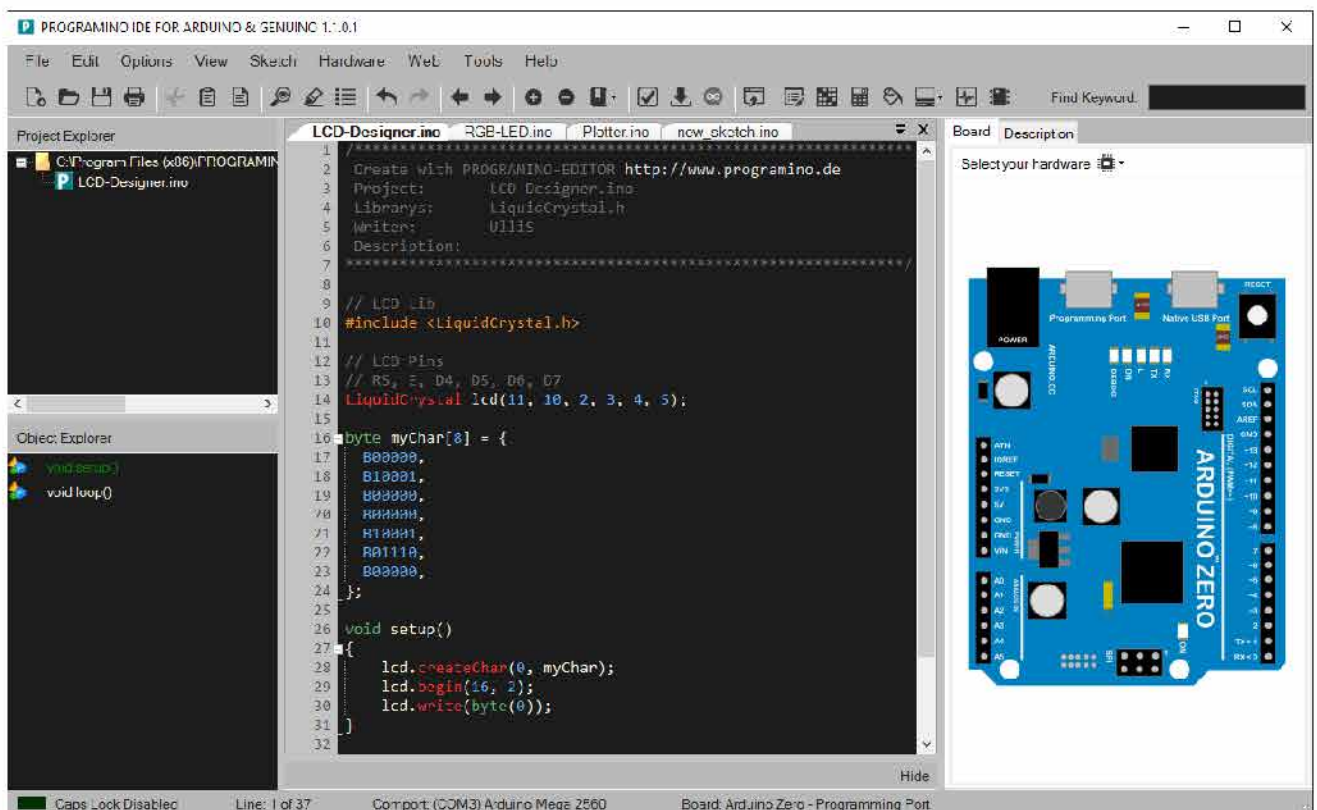


Рисунок 1.22. Середовище розроблення Programino

1.6.2 Інтегроване середовище розроблення B4R (Basic for Arduino)

Basic for Arduino – ще одна альтернатива Arduino ICP (рис.1.23). Це середовище розроблення унікальне тим, що використовує мову Basic, а не C.

файли ядра Arduino, стандартні бібліотеки Arduino, набір інструментів AVR, Arduino Builder, послідовний термінал та симулятор Arduino значення API (знаходиться в стадії розроблювання). Можливо відмітити такі процедури цього середовища: спеціальний проект-майстер задля розроблювання Arduino, скомпільовані файли ядра, котрі кешовані задля більш швидкої компіляції (в порівнянні із оригінальною Arduino ICP), вбудований попередньо сконфігурований AVR-компілятор, завантаження HEX-кодів на плату Arduino (підтримується платою Leonardo) [13].

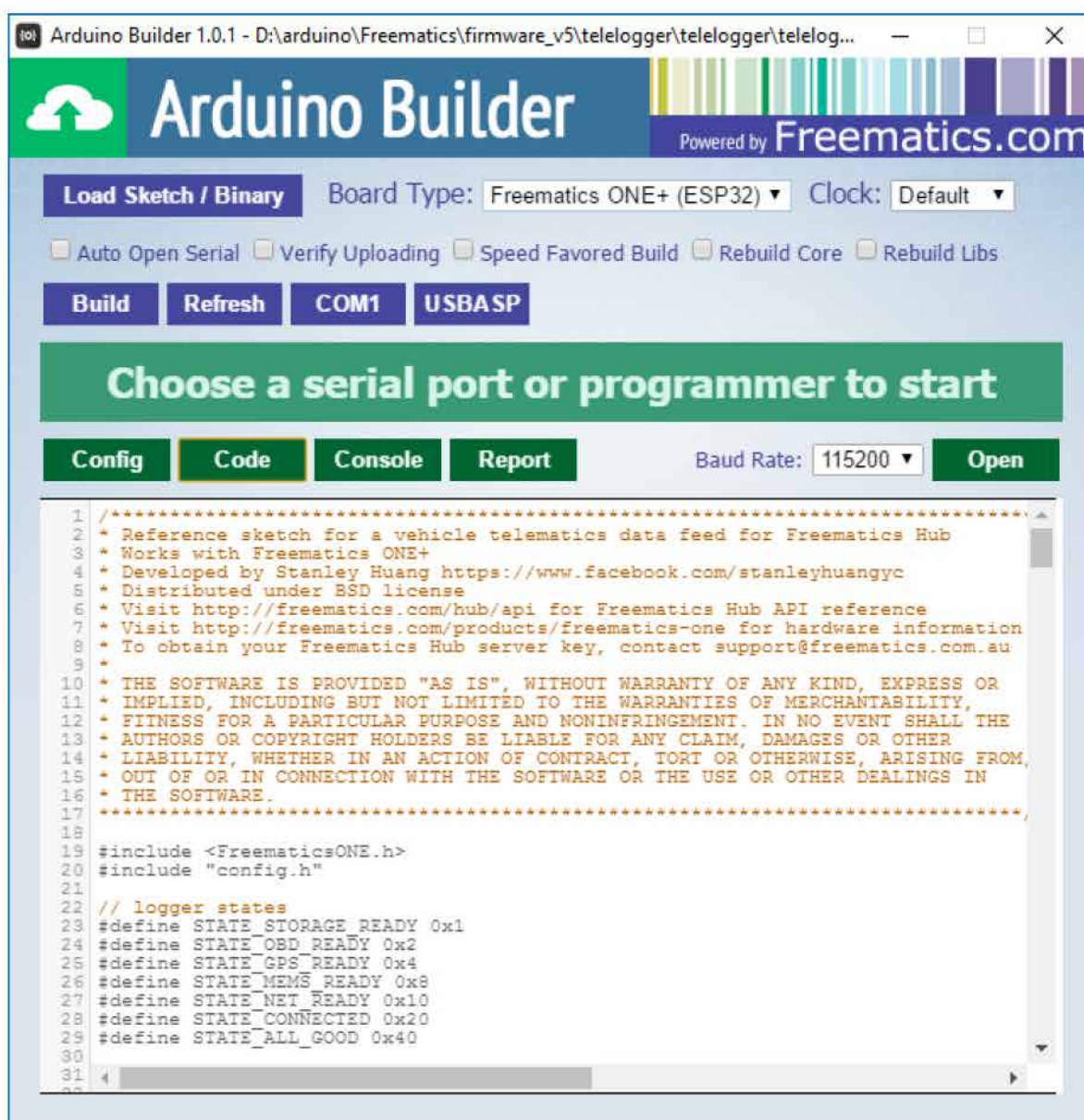


Рисунок 1.24. Середовище розроблювання Codeblocks

1.6.4 Інтегроване середовище розроблювання Arduino ICP

Arduino ICP – найпопулярніше середовище розроблювання, яке розробила сама компанія Arduino. У ньому є весь необхідний мінімум задля розроблювання програм: написання кода, перевірка кода, компіляція, завантаження скетчу у Arduino, монітор послідовного порту. Середовище Arduino ICP досить аскетичне, не містить нічого зайвого та ніяких зручностей користувачеві не пропонує (рис.1.25).

Скетчі – це програми, котрі створюються в середовищі Arduino ICP поза поміччю текстового редактора і пізніше збереженими в файлах із розширенням .ino. Текстовий редактор, присутній в середовищі, володіє усім необхідним функціоналом, а саме: заміни тексту, копіювання, вставки і пошуку. Так само в вікні програми є область повідомлень, котра спроможне давати користувачеві зворотній зв'язок і надсилати йому повідомлення про процеси і помилки, що виникають в системі [14].



Рисунок 1.25. Середовище розроблювання Arduino ICP

| Зм. | Арк. | № докум. | Підпис | Дата |
|-----|------|----------|--------|------|
| | | | | |

КБ 01. 15 000. 00 ДП ПЗ

Арк.

33

Цим способом, опісля проведеного огляду, можливо зробити висновок, що середовище розроблювання Arduino ICP володіє всіма необхідними функціями задля розроблювання програми керування пристроєм перевірки температури і пульса, не містить зайвих функцій, котрі могли б заважати зосередженій роботі із програмою. Так само важливою перевагою є те, що це середовище розроблювання є офіційним, не потребує додаткових обхідних налаштувань, яких вимагають інші ICP.

1.7 Розробка схеми підмикання датчиків і Wi-fi-блоку

Датчики температури і пульса повинні бути підключені до платформи Arduino Нано, котра буде приймати інформація з них (рис.1.26). Так само до платформи Arduino Нано треба підключити Wi-fi-блок задля відправки інформації вимірювань на Гугл-диск.

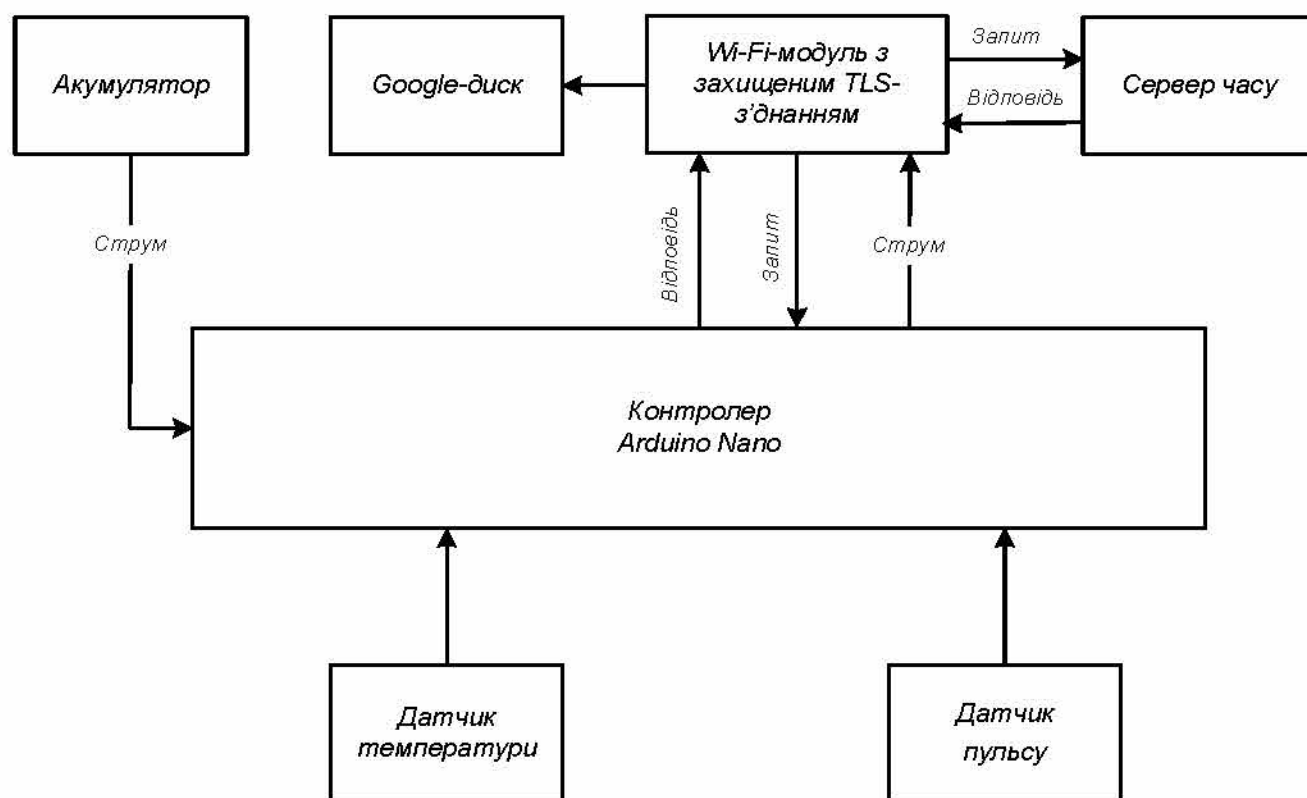


Рисунок 1.26. Структурна схема системи дистанційного перевірки біологічних значень із захистом інформації на позначці TLS

Датчик пульса Pulse sensor під'єднується до виводів землі (GND), живлення датчика 5V і аналогового виводу A0. Датчик температури DS18D30 під'єднується

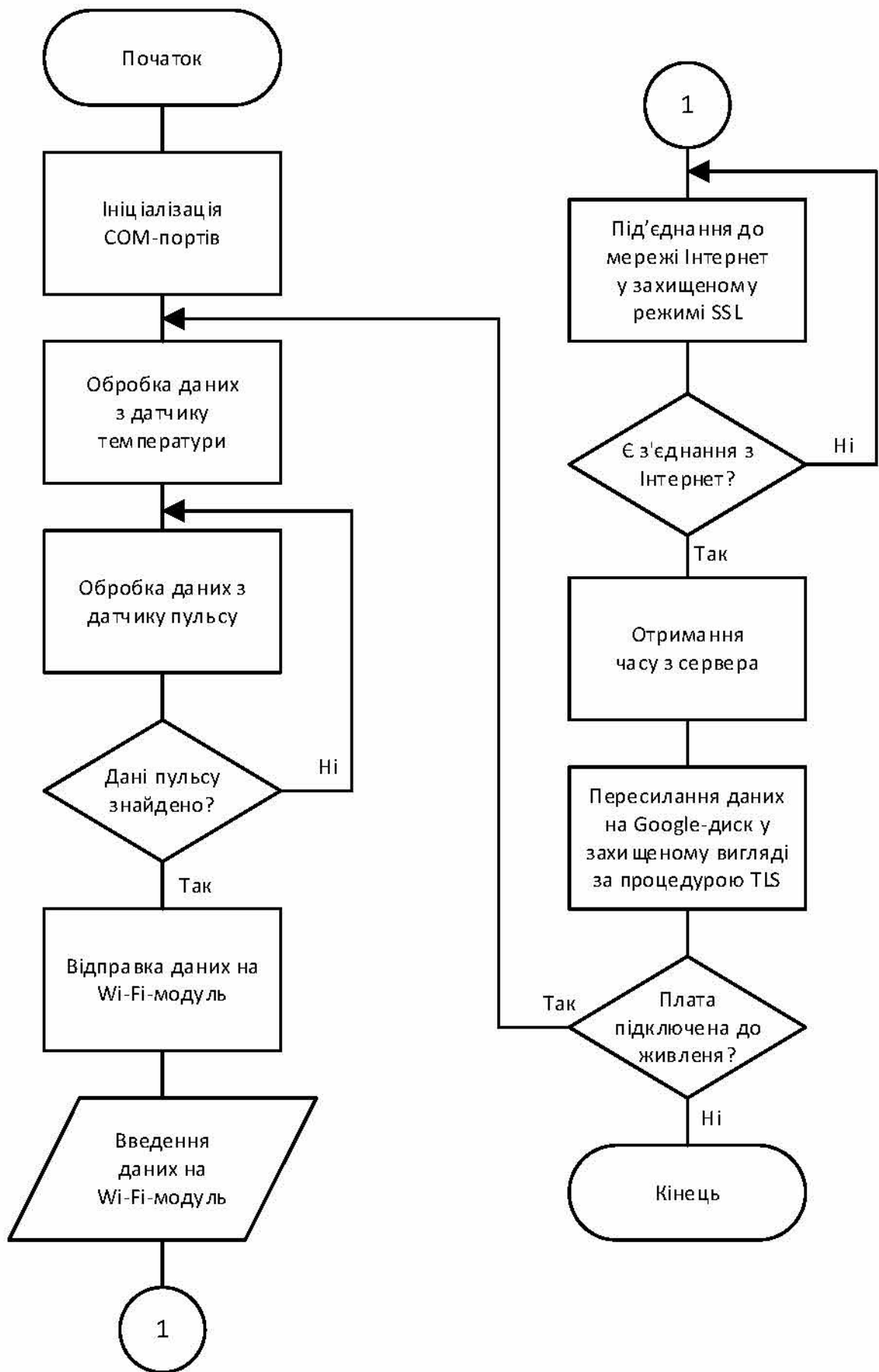


Рисунок 1.28. Блок-схема алгоритму роботи системи дистанційного перевірки біологічних значень із захистом інформації на позначці TLS

1.8 Розробка програмного забезпечення системи дистанційного перевірки

Опісля розроблювання структурної схеми системи дистанційного перевірки біологічних значень із захистом інформації на позначці TLS і схеми із'єднання елементів треба розробити алгоритм роботи приладу і записати програмний код поза поміччю обраного ІСР. Програма буде завантажена в МК платформи Arduino і МК Wi-fi-блоку.

Програмне забезпечення буде складатися із двох частин (програма задля роботи із датчиками і програма задля відправки інформації на Гугл-диск). Код, котрий повинен працювати із датчиками, буде завантажено в модуль Arduino Нано, а код, котрий буде відправляти всі інформація на Гугл-диск в захищеному виді, буде завантажено в МК Wi-fi-блоку ESP8285. Блок-схема алгоритму роботи приладу задля перевірки температури і пульса користувача наведена на рис.1.28.

1.8.1 Програмування мікроконтролера Arduino Нано

Програма задля мікроконтролера Arduino Нано повинна в режимі реального періоду отримувати інформація із давачів температури і пульса, а кожні 8 сек. відправляти інформація на Wi-fi-блок.

Директива `#define`, як зображено на рис.1.29, застосовується задля присвласнення імені константам перед компіляцією програми.

```
#define USE_ARDUINO_INTERRUPTS true
#define ONE_WIRE_BUS 2
```

Рисунок 1.29. Запис константи `#define`

Константи, що задані директивою `define`, не займають програмної пам'яті, оскільки компілятор заміняє всі звернення до них їх значеннями на етапі компіляції. Отже, вони служать задля зручності програміста і більшої читабельності тексту програми. Дуже важливо задати константі `USE_ARDUINO_INTERRUPTS` значення `true` задля налаштування низьких рівнів переривань. Це треба задля найбільш точного обчислення пульса. Так само цього вимагає бібліотека `PulseSensorPlayground.h`, котра застосовується в програмі

зادля роботи із датчиком пульса. Привласнення константи ONE_WIRE_BUS значення 2 вказує, що контакт задля інформації давача пульса під'єднано саме до цифрового контакту 2 платформи Arduino Нано.

```
#include <SoftwareSerial.h>
#include <PulseSensorPlayground.h>
#include <OneWire.h>
#include <DallasTemperature.h>
```

Рисунок 1.30. Підмикання сторонніх бібліотек

Директива #include застосовується в програмі задля імпортування сторонніх бібліотек в скетч (рис.1.30). Вона дає доступ до великого числа стандартних бібліотек і бібліотек, написаних спеціально задля Arduino [11]. Бібліотека SoftwareSerial.h передбачає реалізувати послідовний інтерфейс на будь-яких цифрових виводах Arduino поза поміччю програмних засобів. Вона передбачає програмно створювати декілька послідовних портів, що працюють на швидкості до 115200 бод. Бод – це певна число імпульсів, котра спроможне передаватися поза 1 секунду. Саме поза поміччю неї інформація передбачає передаватися із платформи Arduino Нано на Wi-fi-блок ESP8285. Поза поміччю бібліотеки PulseSensorPlayground.h виконується контроль давача пульса і зчитування із нього інформації. Бібліотека OneWire.h необхідна задля керування датчиками, котрі виробляються компанією Dallas. В проекті приладу задля перевірки температури і пульса застосовується давач температури саме такого типу. Бібліотека DallasTemperature.h застосовується сумісно із бібліотекою OneWire.h [12].

```
SoftwareSerial s(5,6);

OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
PulseSensorPlayground pulseSensor;

const int PulseWire = 0;
float Threshold = 550;
```

Рисунок 1.31. Оголошення змінних в програмі задля Arduino Нано

Оголошення змінних в програмі показано на рис.1.31. В першому рядку кода оголошено, що передавати інформація між платами треба крізь 5-й і 6-й цифрові виводи. Пізніше створюється екземпляр класу OneWire задля того, аби із його поміччю обмінюватися сигналами із пристроями по однодротяній (послідовній) шині, – в даному випадку із температурним датчиком. Далі відбувається передача об'єкта OneWire об'єкту sensors. В наступному рядку створено екземпляр об'єкту PulseSensorPlayground під назвою pulse sensor. Наприкінці створено дві змінні, одна із яких вказує на те, що інформація давача пульса передбачає зчитуватися із нульового аналогового піну. Інша змінна застосовується задля визначення того, котрий сигнал буде вважатися як удар серця, а котрий буде ігноруватися.

```
void setup() {
  delay(10000);
  s.begin(9600);
  Serial.begin(9600);

  pulseSensor.analogInput(PulseWire);
  pulseSensor.setThreshold(Threshold);

  if (pulseSensor.begin()) {
    Serial.println("Start pulse");
    sensors.begin();
  }
}
```

Рисунок 1.32. Реалізація процедури setup

Процедура setup (рис.1.32) необхідна задля ініціалізації системи. В цій процедурі знаходяться команди, котрі МК виконає один раз в момент завантаження при запуску системи. Команда Delay із параметром 1000 задає 10 сек. очікування задля мікроконтролера перед виконанням наступних команд. Інструкція S.begin(9600) ініціалізує роботу із послідовним портом, котрий буде передавати інформація на іншу плату на швидкості 9600 імпульсів поза секунду. Далі, використовуючи pulseSensor, котрий є екземпляром об'єкту PulseSensorPlayground, вказано, із якого аналогового входу МК буде зчитуватися

інформація давача пульса – в даному випадку це контакт А0. Пізніше вказано, котрі інформація пульса передбачає ігноруватися.

Крізь умовний оператор `if` вказано, що бібліотека `sensors` містить запускатися лише опісля того, як давач пульса налаштується і почне передавати актуальні інформація.

```
void loop() {  
  delay(8000);  
  int myBPM = pulseSensor.getBeatsPerMinute();  
  if (pulseSensor.sawStartOfBeat()) {  
    Serial.print("Pulse: ");  
    Serial.println(myBPM/2.7);  
  }  
}
```

Рисунок 1.33. Реалізація процедури `loop`

В процедурі `loop` (рис. 1.33) знаходяться лише ті команди, котрі передбачає виконуватися весь період, доки платформа Arduino увімкнена. Почавши із першої команди МК по черзі проходить всі наступні, поки не дійде до самого кінця, а виконавши останню команду, він переходить на початок і повторює цю послідовність. `void loop` – головна процедура входу в програму [13]. Команда `Delay` із параметром 8000 необхідна задля того, аби процедура `loop` виконувалася кожні 8 сек., що достатньо задля забезпечення високої точності вимірюваних інформації і їх зручної обробки.

В наступному рядку кода змінній `myBPM` привласнюється значення кількості ударів серця поза хвилину, використовуючи функцію `getBeatsPerMinute()`. Пізніше крізь умовний оператор `if` і функцію `sawStartOfBeat` перевіряється, чи є інформація з давача пульса. Коли програма отримує позитивний результат, тоді на екран виводиться значення пульса, а контролер продовжує виконання завдання.

```
sensors.requestTemperatures();  
float t = sensors.getTempCByIndex(0);  
Serial.print("TEMPERATURE = ");  
Serial.print(t);  
Serial.print("°C");  
Serial.println();
```

Рисунок 1.34. Процес отримання температури

На рис.1.34 показано, як відбувається запит `sensors.requestTemperatures()`, що звертається до всіх пристроїв на шині задля видачі інформації температури. Після отримання інформації температури можливо вивести їх чи переслати на Wi-Fi модуль. Процедура `getTempCByIndex(0)` повертає значення температури поза індексом, отже отримаємо буде температуру лише з першого давача. Це корисно, коли до шини приєднано декілька температурних давачів. Значення температури записується в змінну `t` типу інформації `float`.

На рис.1.35 зображено програмний код, котрий пересилає інформація на Wi-fi-блок ESP8285, а саме записує значення пульса і температури в змінну, яку пізніше зчитує модуль ESP8285 Node-mcu.

Умовний оператор `if` із командою `s.available()` необхідний задля того, аби перевірити чи є вхідні інформація в серії. Отже, спочатку виконується перевірка на наявність вхідних інформації. Коли відповідь позитивна, тоді інформація записуються і передаються на іншу плату. Спочатку відправляється результат перевірки пульса, а одразу поза ним – інформація температури тіла [13].

```
if (s.available() > 0)
{
    s.write(myBPM/2.7);
}

if (s.available() > 0)
{
    s.write(t);
}
```

Рисунок 1.35. Процес пересилання інформації на Wi-fi-блок ESP8285

Цим способом, програмне забезпечення, що завантажується в плату Arduino Нано, реалізує підмикання мікроконтролеру до давачів і перевірку початку надходження з них інформації. Після отримання необхідних інформації МК Arduino Нано надсилає ці інформація на Wi-fi-блок ESP8285 Node-mcu.

1.8.2 Програмування Wi-fi-блоку ESP8285 Node-mcu

Перед створенням програми задля Wi-fi-блоку ESP8285 Node-mcu треба визначити, котрі саме задачі передбачає стояти перед мікроконтролером цього модулю.

Перша процедура пов'язана із отриманням значень діагностики із платформи Arduino Нано. Вони передбачає отримуватися кожні 8 сек. і записуватися в відповідні змінні.

Друга процедура пов'язана із відправленням інформації на Гугл-диск. Задля цього треба приєднуватися до мережі Wi-Fi в захищеному режимі TLS-із'єднання. Оскільки треба, аби у процесі аналізу значень користувач бачив, в котрий період робилися визначення, треба разом із результатами перевірки відправляти та період визначення. Задля цього передбачене отримання періоду із сервера світового періоду. Кожен раз, коли прилад перевірки буде відправляти інформацію на Гугл-диск в захищеному виді, він так само разом із цим буде приєднуватися до сервера, отримувати поточний період, переводити його в зручний формат і завантажувати на Гугл-диск.

Окрім програмування Wi-fi-блоку ESP8285 Node-mcu треба так само налаштувати і запрограмувати сервіс Гугл-диску задля того, аби він міг отримувати інформацію із платформи Arduino Нано.

```
#include <SoftwareSerial.h>
#include <ESP8266WiFi.h>
#include <WiFiUdp.h>
#include "HTTPSRedirect.h"
#include "DebugMacros.h"

SoftwareSerial s(D6, D5);
```

Рисунок 1.36. Підмикання необхідних бібліотек

Бібліотека SoftwareSerial.h реалізовує послідовний інтерфейс на цифрових виводах Arduino поза поміччю програмних засобів. Поза її поміччю інформація передаються із платформи Arduino Нано на Wi-fi-блок ESP8285 (рис.1.36). Бібліотека ESP8266WiFi.h застосовується задля підмикання Wi-fi-блоку до мережі Wi-Fi, аби почати відправку чи отримання інформації. Бібліотека

WiFiUdp.h застосовується задля програмування UDP-функціоналу. UDP чи протокол користувачьких датаграм є одним із ключових мережевих протоколів задля Інтернет. Він передбачає комп'ютерним додаткам надсилати повідомлення іншим IP-хостам без необхідності попереднього встановлення каналів зв'язку. Бібліотека HTTPSRedirect.h заснована на бібліотеці Wifi. Вона надає користувачеві можливості задля легкого спілкування із серверами поза протоколами HTTP і HTTPS. Задля зв'язку поза протоколом HTTPS треба мати сертифікати підмикання сайтів. Бібліотека DebugMacros.h допомагає зберігати і обробляти запити у цьому рядку. Нижче в кодї оголошено, що сигнали між платами передбачає передаватися крізь цифрові виводи D6 і D5 [14].

```
unsigned int puls;
unsigned int temp;
String sheetPulse = "";
String sheetTemp = "";
const char* ssid = "TP-Link_BA54";
const char* password = "slastion4"; //Put WiFi password within the quotes
const char* host = "script.google.com";
const char *GScriptId = "AKfycbzbzqZP-nVgw1DcSAQYdtFsza__IUCySWWIw4o0KjDLFfhIAdxBSO"
const int httpsPort = 443; //the https port is same
```

Рисунок 1.37. Оголошення змінних в програмі задля Wi-fi-блоку

Під період оголошення і ініціалізації змінних в програмі задля Wi-fi-блоку спочатку задаються змінні, котрі передбачає зберігати значення пульса і температури типу unsigned int (рис.1.37). Цей тип схожий на int, але замість з'ємних чисел спроможне зберігати лише додатні значення в діапазоні з 0 до 65535. Далі створено змінні із типом String, в котрі передбачає записуватися інформація перевірки. Ці змінні передбачає об'єднані в одну спільну змінну, котра пізніше буде передаватися на Гугл-диск. Пізніше задаються змінні, котрі зберігають інформацію, необхідну задля підмикання до Wi-Fi і Гугл-диска. Вони містять тип char, котрий займає в пам'яті 1 байт і зберігає символні значення. Однак в пам'яті символи зберігаються в виді чисел. Відповідне кодування можливо знайти в таблиці ASCII-кодів символів. Змінні ssid і password зберігають назву і пароль з точки доступу до Інтернет. Їх можливо змінювати,

коли прилад буде використовуватися в іншій мережі. В змінній `host` зберігається назва сайту, доступ до якого містить отримувати МК, задля завантаження інформації. Змінна `GScriptId` зберігає у собі ідентифікатор скрипту, вбудований в URL веб-додатку. Вона застосовується кодом Arduino задля доступу до листа Гугл. Цей код можливо отримати при налаштуванні Гугл-диску. Поза замовчуванням задля `https` застосовується порт 443, саме таке значення надається змінній `httpsPort`.

```
const char* ntpServerName = "0.ua.pool.ntp.org";
const int NTP_PACKET_SIZE = 48; // NTP time stamp
byte packetBuffer[ NTP_PACKET_SIZE]; //buffer to
```

Рисунок 1.38. Визначення змінних серверного періоду

В змінній `ntpServerName`, як показано на рис. 1.38, буде зберігатися ім'я серверу, на котрий МК буде надсилати запит задля отримання реального періоду. Це вітчизняний сервер, тому період запиту буде мінімальний. Константа `NTP_PACKET_SIZE` містить значення типу `int`, яке дорівнює 48, адже відмітка періоду в відповіді сервера знаходиться в перших 48 байтах повідомлення. NTP (network time protocol) є стандартним мережевим інтернет-протоколом інформації періоду, котрий застосовується задля синхронізації годинника комп'ютера із певним точним часом.

```
client = new HTTPSRedirect(httpsPort);
client->setInsecure();
client->setPrintResponseBody(true);
client->setContentTypeHeader("application/json");
Serial.print("Connecting to ");
Serial.println(host);

bool flag = false;
for (int i = 0; i < 5; i++) {
  int retval = client->connect(host, httpsPort);
  if (retval == 1) {
    flag = true;
    break;
  }
  else
    Serial.println("Connection failed. Retrying...");
}
if (!flag) {
  Serial.print("Could not connect to server: ");
  Serial.println(host);
}
```

Рисунок 1.39. Процедура створення TLS-із'єднання

В даному проекті протокол NTP застосовується не задля синхронізації періоду комп'ютера, а задля того, аби отримувати реальний період, котрий буде заноситися в електронну таблицю поряд із даними вимірювань [13]. Наприкінці показаного фрагмента кода оголошено змінну-буфер, в якій передбачає зберігатися вхідні і вихідні повідомлення. Буферні змінні необхідні лише задля обміну даними.

На рис.1.39 зображений фрагмент кода, де клас HTTPSRedirect застосовується задля створення нового захищеного TLS-із'єднання. TLS (transport layer security) – криптографічний протокол захисту транспортного значення. Він забезпечує захищену передачу інформації між вузлами в мережі Інтернет, використовує асиметричне шифрування задля автентифікації, симетричне шифрування задля конфіденційності і коди автентичності повідомлень задля збереження їх цілісності. Далі показано під'єднання до ресурсу script.gугл.com, аби мати можливість пересилати змінні. Крізь умовний оператор if обмежено число спроб підмикання до 5 разів, а опісля цього тоді припиняється виконання кода [14].

```
WiFi.hostByName(ntpServerName, timeServerIP);  
  
sendNTPpacket(timeServerIP);  
  
delay(1000);
```

Рисунок 1.40. Процедура відправки NTP-пакета

На рис. 1.40 в першому рядку кода відбувається із'єднання із сервером, ім'я якого було задане раніше, при ініціалізації змінних. Далі, поза поміччю команди sendNTPpacket, відбувається відправлення на сервер реального періоду NTP-пакету. Опісля цього відбувається очікування (1 секунда), аби дізнатися, чи із'явилася відповідь. NTP-протокол використовує задля своєї роботи протокол UDP і враховує період передачі. Система NTP дуже стійка до змін латентності під період передачі. В четвертій версії протоколу система NTP спроможне досягати швидкості 10 мс (1/100 с) при роботі в Інтернет і до 0.2 мс (1/5000 с) в локальних мережах.

```

int cb = udp.parsePacket();
if (!cb) {
    Serial.println("no packet yet");
} else {
    Serial.print("packet received, length=");
    Serial.println(cb);

    udp.read(packetBuffer, NTP_PACKET_SIZE);

```

Рисунок 1.41. Процедура зчитування NTP-паketу

Процедура `Udp.parsePacket()` перевіряє наявність прийнятого UDP-паketу і повертає його розмір (рис.1.41). Процедура `parsePacket()` повинна визиватися перед зчитуванням буфера методом `udp.read()`. Поза поміччю процедури `parsePacket()` ініціалізовано змінну `cb` як цей паket. Коли вона порожня, тоді виводиться повідомлення про те, що отримати паket не вдалося. Інакше виводиться повідомлення про вдалу спробу і інформація зчитані із цього паketу. Пізніше програма зчитує інформація в буфер поза поміччю процедури `udp.read()`. Коли параметри у ній не задані, тоді вона повертає черговий прийнятий символ.

```

String hour = String(((epoch % 86400L) / 3600)+3);
Serial.print(':');
String minute = "";
if (((epoch % 3600) / 60) < 10) {
    // In the first 10 minutes of each hour, we'll want
    Serial.print('0');
    minute = String('0')+ String((epoch % 3600) / 60);
} else {
    minute = String((epoch % 3600) / 60);
}
String second = "";
Serial.print((epoch % 3600) / 60); // print the minute
Serial.print(':');
if ((epoch % 60) < 10) {
    // In the first 10 seconds of each minute, we'll want
    second = String('0') + String(epoch % 60);
    Serial.print('0');
} else {
    second = String(epoch % 60);
}

```

Рисунок 1.42. Процедура розрахунку періоду

Змінна `epoch` на рис.1.42 зберігає в собі поточний період в форматі сек., починаючи із 1970 року. Задля того, аби спочатку дізнатися точний період у

години, ця змінна ділиться на 86400, оскільки це число сек. в добі, а пізніше на 3600, задля того аби дізнатися число годин. Пізніше треба знайти період в хвилинах, задля чого змінна `epoch` ділиться на число годин в секундах. Пізніше це число ділиться на 60 сек.. Кінцеве число буде точним часом в хвилинах. Поза поміччю умовного оператора `if` виконується перевірка – коли число менше поза 10, тоді перед кількістю хвилин буде додано цифру 0, аби період складався із двох цифр. Опісля цього треба дізнатися період в секундах, задля чого треба поділити `epoch` на 60 із остачею. Коли виходить число менше 10, тоді перед ним додається 0. Кінцевим результатом опісля цих розрахунків отримано три змінні – `second`, `minute` і `hour`. Опісля конкатенації змінних утворюється змінна `realtime`, котра та буде передаватися на Гугл-диск.

```

s.write("s");
if(s.available()>0)
{
puls=s.read();
}
if(s.available()>0)
{
temp=s.read();
}

```

Рисунок 1.43. Процедура зчитування значень перевірки

Wi-fi-блок записує керуючий символ, надсилає його до модулю Arduino Нано і починає прослуховувати інформація із нього. Процедура `available()` повертає число байтів (символів), доступних задля зчитування із буферу послідовного порту (рис.1.43). Під символами розуміються інформація, котрі вже прийняті і зберігаються в послідовному буфері приймача. Процедура `available()` є наслідувачем допоміжного класу `Stream`. Коли є доступні інформація, тоді процедура `available()` зчитує байт [14].

Змінні `sheetPulse` і `sheetTemp` створюються задля збереження інформації пульса і температури тіла користувача (рис.1.44).

```

Serial.print("Pulse: "); Serial.print(puls);
sheetPulse = String(puls); //convert integer h
Serial.print("% Temperature: "); Serial.print(temp); Serial.println("°C ");
sheetTemp = String(temp);
static int error_count = 0;
static int connect_count = 0;
const unsigned int MAX_CONNECT = 20;
static bool flag = false;
payload = payload base + "\"" + sheetTemp + "," + sheetPulse + "," + realtime + "\"";

```

Рисунок 1.44. Ініціалізація змінних задля передачі

Значення цих змінних виводяться на екран. Коли в процесі рефакторингу кода треба буде оцінювати значення змінних, швидше це зробити в ICP Arduino ICP, ніж крізь Гугл-диск. Так само ініціалізується змінна payload, в якій крізь кому буде записано значення змінних sheetTemp, sheetPulse і теперешній період realtime. В програмі так само присутня перевірка на граничне значення пульса. Коли значення пульса виявиться меншим, ніж можливе в користувача, тоді інформація на Гугл-диск відправлятися не передбачає. Цим способом таблиця не буде містити некоректної інформації.

```

}if (puls > 40) {
Serial.println("POST or SEND Sensor data to Google Spreadsheet:");
if (client->POST(url2, host, payload)) {
;
}

```

Рисунок 1.45. Процедура перевірки на наявність інформації пульса

Інформація завантажуються поза помічню процедури client->Post (рис.1.45). Її аргументи – url2, тобто сайт, де зберігається скрипт Гугл-диску. Host – змінна, котра зберігає назву сайту, а payload – містить самі інформація.

Цим способом, МК Wi-fi-блоку отримує дві змінні з модулю Arduino Нано – зі значеннями пульса і значенням температури, а так само відправляє запит на сервер реального періоду задля отримання періоду вимірювань. Відповідь сервера обробляється задля відображення чисел в виді годин, хвилин і сек.. Отримуючи всі необхідні результати, МК зв'язується із Гугл-диск і пересилає на нього інформацію. Опісля цього Гугл-диск треба налаштувати задля прийняття і відображення інформації перевірки.

1.8.3 Налаштування сервісу Гугл-диска

Редактор електронних таблиць Гугл Sheet забезпечує зручний спосіб зберігання чи обробки інформації в формі електронних таблиць і спроможне бути інтегрований із десятками інших сервісів, представлених Гугл, наприклад Maps. Поза поміччю API і використання gscript істотно спрощуються програмне заповнення інформації в Гугл Sheet. Задля відправки інформації на Гугл Sheet треба спочатку створити лист, а пізніше підготувати його задля отримання інформації із розробленого приладу перевірки, створивши gscript задля прив'язки листа до приладу. На сайті docs.gугл.com треба увійти в особистий обліковий запис Гугл. Опісля відкриття сторінки треба із випадаючого списку обрати Гугл sheet. Опісля створення електронної таблиці треба дати їй ім'я, наприклад, MyData. Опісля зміни назви таблиці можливо змінити ім'я листа, наприклад, mainpage. Опісля створення файлу проекту треба записати код, що буде відповідати поза коректну роботу електронної таблиці і розподіляти інформація по стовпцям. Задля цього треба обрати вкладку «Інструменти» і перейти на «Редактор сценаріїв», як показано на рис. 1.46.

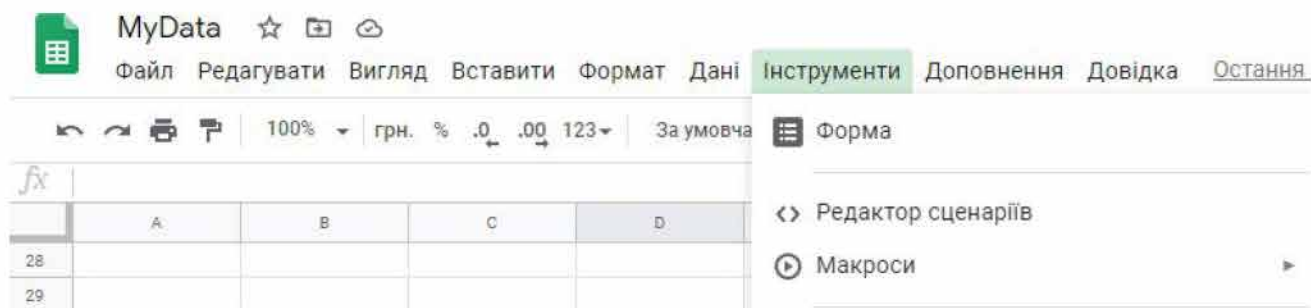


Рисунок 1.46. Меню створення файлу із скриптом в Гугл Sheet

Опісля відкриття нової вкладки, в якій буде знаходитись редактор скриптів Гугл, файлу надано ім'я "Project". На рис.1.47 показано скрипт, котрий буде керувати електронною таблицею [15].

```
var SS = SpreadsheetApp.openById('1nIcfyFKh2nG5Ape0jVAik7WUWGZ8D_f11Xp3Ku0Wcyo');  
var sheet = SS.getSheetByName('mainpage');  
var str = "";
```

Рисунок 1.47. Скрипт задля електронної таблиці Гугл sheet

В кодї програми на рис. 1.48 задаються дві змінні: перша із них, `SS` – посилання на електронну таблицю, а друга, `sheet` – назва листа таблиці, на котрий передбачає завантажуватися інформація перевірки. Процедура `getSheetByName` повертає лист із вказаним іменем. Коли декілька листів містять одне та те саме ім'я, тоді повертається те ім'я, яке стоїть раніше, чи повертається `null`, коли немає листа із вказаним іменем.

Поза поміччю виклику процедури `getRange('A1')` можливо отримати секцію `A1` на першому листі. Клас `range` застосовується задля доступу і зміни діапазонів електронної таблиці. Діапазон спроможне бути окремою секцією на листі чи групою сусідніх секцій на листі. Метод класу `setValue()` встановлює значення діапазону. Значення спроможне бути числовим, строковим, логічним чи датою. Метод `getValue()` повертає значення верхньої лівої секції в діапазоні. Значення спроможне бути різних типів у залежності з значення секції. Порожні секції повертають порожній рядок. Далі, поза поміччю процедури `setValue()` встановлюються значення `BPM` і `Time` в секціях `B1` і `C1` задля того, аби було зрозуміло, котрі значення записуються в котрий стовпець [15].

```
var range = sheet.getRange('A1');
var retval = range.setValue(val).getValue();
var now = Utilities.formatDate(new Date(), "
//sheet.getRange('B1').setValue(now);
sheet.getRange('B1').setValue('BPM');
sheet.getRange('C1').setValue('Time');
```

Рисунок 1.48. Реалізація процедури `getRange()`

Опісля створення програмного кода задля функціонування електронної таблиці, його треба опублікувати задля збереження і автономного запуску.

На верхній панелі інструментів проекту треба обрати вкладку «Опублікувати» (рис.1.49), а в меню, що із'явилося, натиснути кнопку «Ввести у дію як веб-програму...» і перейти до сторінки публікації. Там треба заповнити інформацію про версію проекту, e-mail автора, вказати, кому надається доступ до додатку, хто спроможне змінювати програму і додавати інформація. Опісля цього треба надати кілька дозволів та програма буде готовою до роботи.

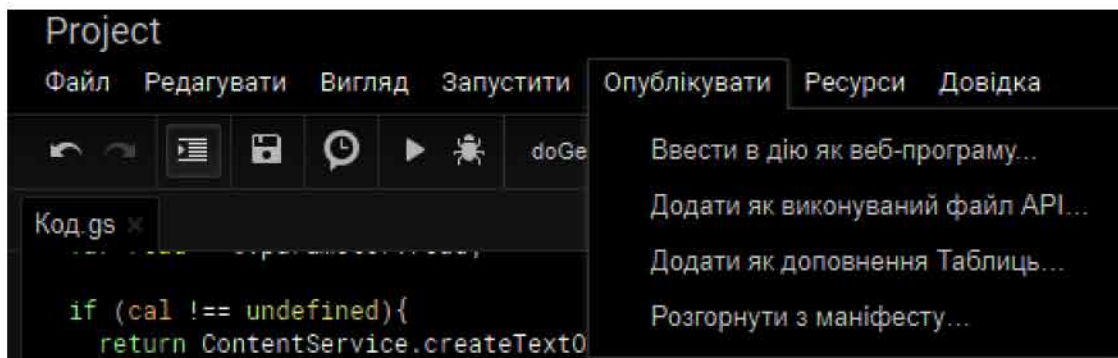


Рисунок 1.49. Процес публікації веб-програми

На рис.1.50 показана схема сценарію використання функцій системи дистанційного перевірки біологічних значень із застосуванням Гугл-диска.

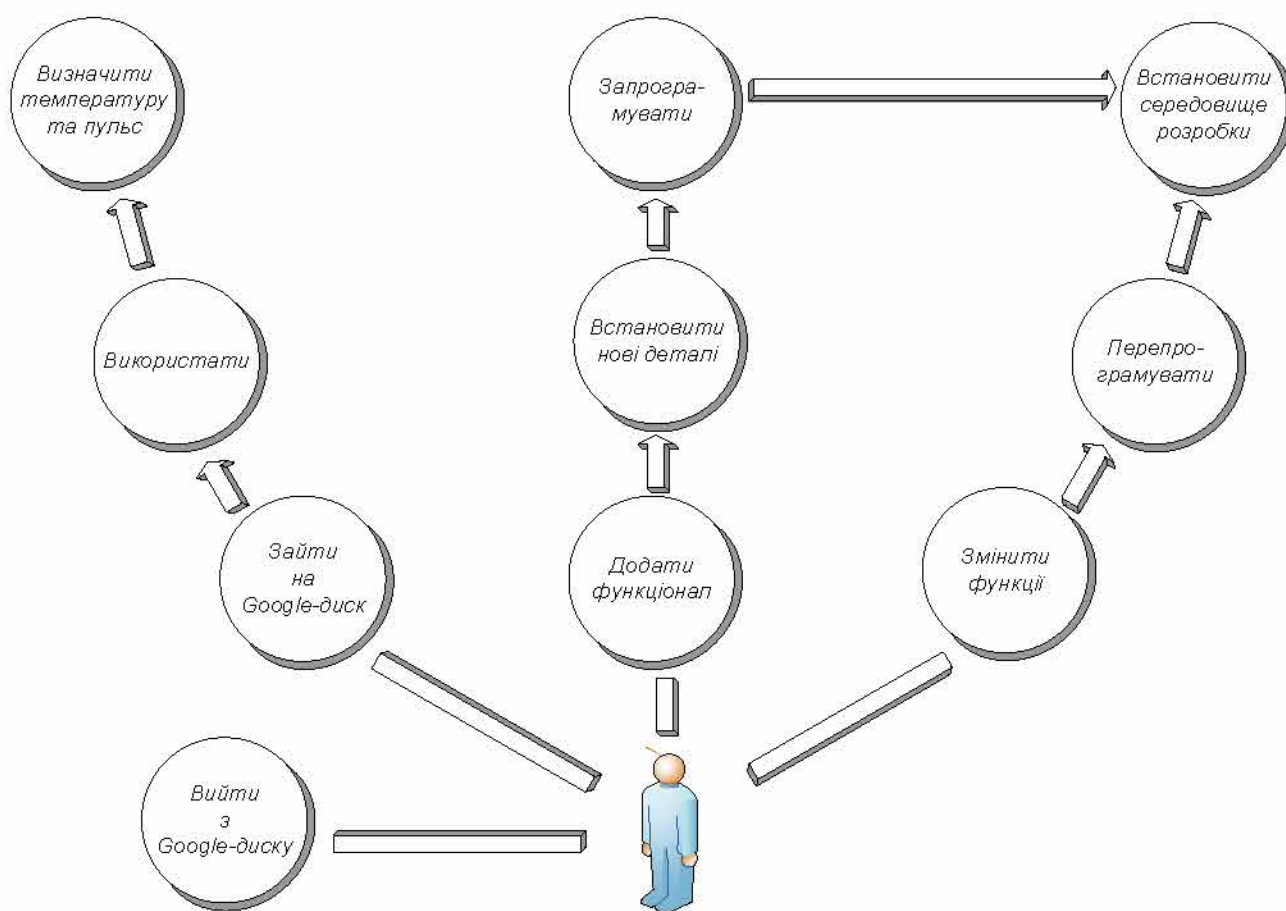


Рисунок 1.50. Сценарій використання функцій системи дистанційного перевірки біологічних значень із застосуванням Гугл-диска

1.9 Експлуатація системи дистанційного перевірки біологічних значень

Опісля етапів підмикання модулів приладу і розроблювання програмного забезпечення, його тестування і відлагодження, треба завантажити програми на

обидві платформи (Arduino Нано і Wi-fi-блок), а Гугл-диск налаштувати на приймання інформації. Опісля цього можливо запускати систему дистанційного перевірки біологічних значень.

Оскільки платформа Wi-fi-блоку ESP8285 Node-mcu крізь контакт Vin живиться з 5-вольтового контакту на платі Arduino Нано, тоді необхідний лише один роз'єм USB задля повноцінного функціонування приладу. Опісля подачі живлення на плату, незалежно з того, під'єднана вона до комп'ютеру, чи лише блоку живлення, крізь 10 сек. починають вимірюватися температура і пульс користувача.

Користувачеві системи дистанційного перевірки біологічних значень треба дотримуватися декілька правил при вимірюванні:

- треба відпочити 3-5 хвилин, не розмовляти і не рухатися перед тим, як проводити визначення;
- треба сісти в зручне крісло, що підтримує спину. Ноги і руки при цьому треба тримати не схрещеними;
- руки треба покласти на стіл на позначці серця чи на іншу рівну поверхню.

Коли прилад запущено, результати вимірювань можливо побачити в меню монітора порту середовища розроблювання Arduino ICP. Задля того, аби зайти в монітор порту, треба виконати декілька кроків. Треба зайти в середовище розроблювання Arduino ICP і на панелі інструментів обрати вкладку «Інструменти». Натиснувши на ній, треба обрати порт, до якого під'єднано прилад – це номер USB-порту комп'ютеру, котрий застосовується задля програмування приладу. Так само треба вибрати назву використовуваної платформи Arduino Нано. Це треба обов'язково зробити, інакше середовище розроблювання Arduino ICP не зможе із'єднатися із платою.

На рис. 1.51 наведено приклад відображення значень у моніторі порту. Опісля того, як інформація пульса знайдено, виводиться напис “Start pulse”, а пізніше значення пульса і температури, котрі оновлюються кожні 8 сек..

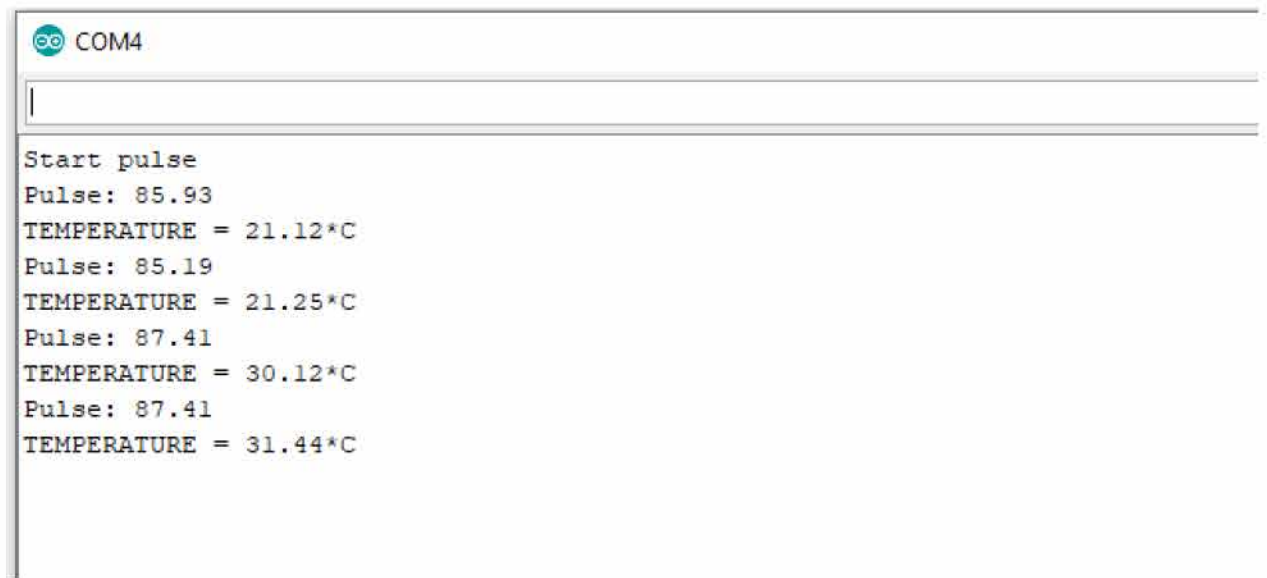


Рисунок 1.51. Стан монітора порту в процесі відображення значень

Коли прилад запуститься, в Гугл Sheet на Гугл-диску, коли він порожній, колонки отримують відповідні назви, як це показано на рис. 1.52. Коли ж в Гугл Sheet на Гугл-диску вже є записи значень, тоді нові передбачає додаватися знизу до попередніх.

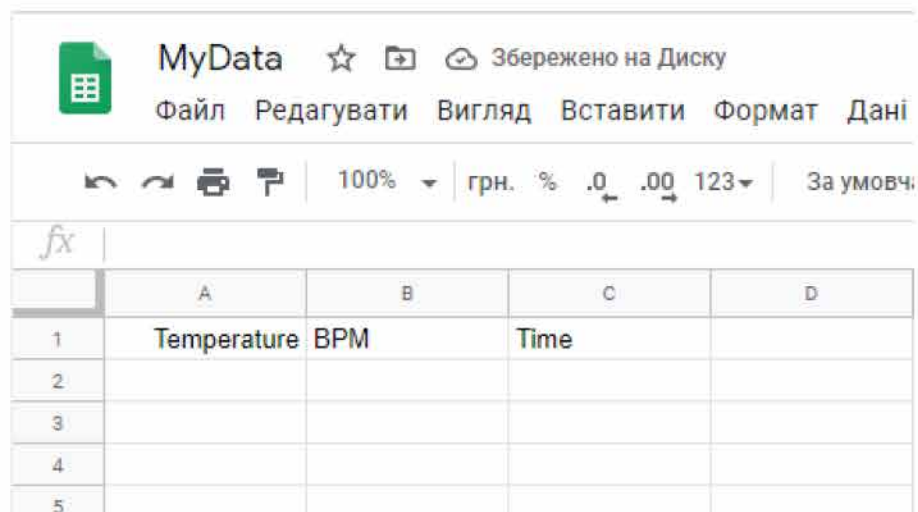


Рисунок 1.52. Нові колонки в електронній таблиці Гугл Sheet на Гугл-диску

Опісля створення нових колонок в електронній таблиці Гугл Sheet, вони кожні 8 сек. заповнюються новими даними. На рис. 1.53 показано заповнену таблицю Гугл Sheet. В ній присутні інформація температури, результати перевірки пульса, а так само точний період, коли було виконане відповідне визначення.

MyData ☆ 📄 🗑️

Файл Редагувати Вигляд Вставити

↶ ↷ 🖨️ 📄 100% | грн. % .0 .00 12

fx

| | A | B | C |
|----|-------------|-----|----------|
| 1 | Temperature | BPM | Time |
| 2 | 36 | 78 | 13:27:25 |
| 3 | 36 | 78 | 13:30:08 |
| 4 | 36 | 78 | 13:30:14 |
| 5 | 36 | 85 | 13:30:21 |
| 6 | 36 | 85 | 13:30:27 |
| 7 | 36 | 85 | 13:30:34 |
| 8 | 36 | 85 | 13:30:40 |
| 9 | 36 | 85 | 13:30:47 |
| 10 | 36 | 77 | 16:57:59 |
| 11 | 36 | 77 | 16:57:59 |
| 12 | 36 | 77 | 16:58:10 |
| 13 | 36 | 77 | 16:58:16 |
| 14 | 36 | 82 | 16:58:27 |
| 15 | 36 | 82 | 16:58:33 |
| 16 | 36 | 85 | 16:58:40 |
| 17 | 36 | 85 | 16:58:46 |
| 18 | 36 | 84 | 17:00:50 |

Рисунок 1.53. Інформація в таблиці Гугл Sheet на Гугл-диску

Електронну таблицю Гугл Sheet із отриманими даними користувач приладу перевірки температури і пульса спроможне переглядати власноруч, це спроможне зробити лікар чи будь-котра людина, якій буде надано доступ на Гугл-диск. При цьому доступ до інформації не спроможне отримати сторонній користувач, адже інформація захищені поза протоколом TLS. Оскільки датчики приладу містять не дуже високу точність, виникає незначна похибка, котра особлива помітна на початку перевірки, коли прилад налаштовується. Цим способом, треба не брати до уваги інформація, котрі надійшли протягом декількох перших сек. роботи приладу.

| | | |
|-----------------------------|--------|--|
| 8. Адміністративні витрати | 29.79 | $V_a = V_{oz} * 0,3$ $V_a = 99.32 * 0,3$ |
| 9. Витрати на збут | 15.22 | $V_{зб} = C_{вир} * 0,02$ $V_{зб} = 761.15 * 0,02$ |
| 10. Інші операційні витрати | 7.61 | $V_{оп} = C_{вир} * 0,01$ $V_{оп} = 761.15 * 0,01$ |
| <i>Повна собівартість</i> | 813.77 | $C_{пов.} = C_{вир} + V_a + V_{зб} + V_{оп}$ $C_{пов.} = 761.15 + 29.79 + 15.22 + 7.61$ |

Оптову ціну виробу визначаємо по формулі:

$$C_o = C_{пов} + \Pi = 813.77 + 162.75 = 976.52 \text{ грн.}$$

Ціну реалізації виробу встановлюємо із урахуванням ПДВ:

$$C_p = C_o + \Pi_з,$$

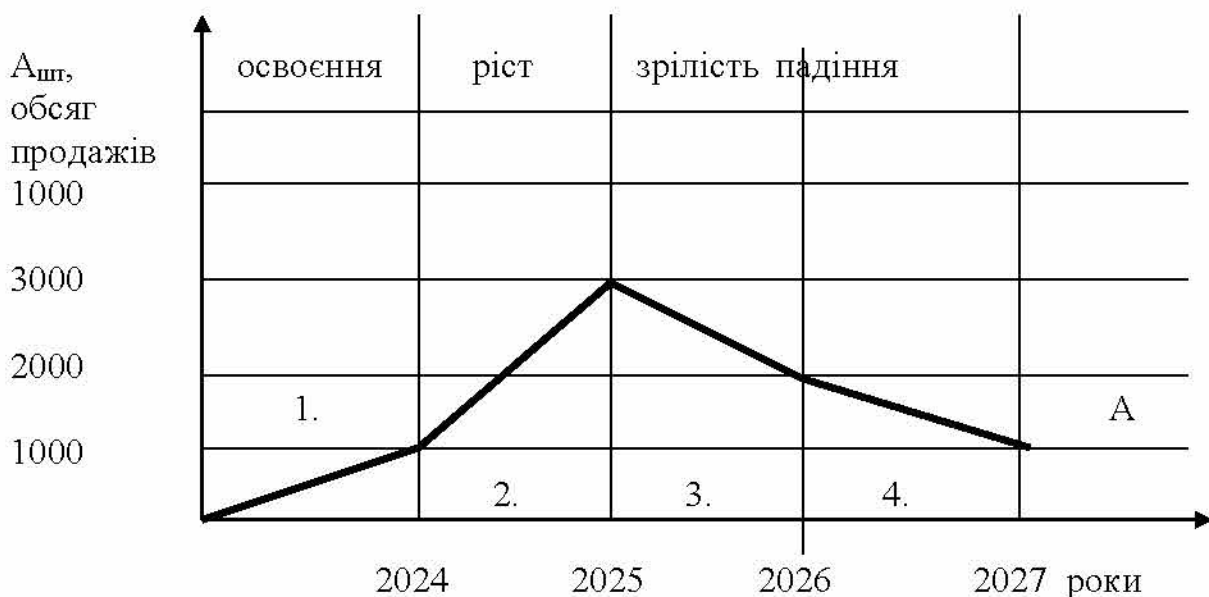
де $\Pi_з$ – податкове зобов'язання із ПДВ:

$$\Pi_з = C_o * 0,2 = 976.52 * 0,2 = 195.3 \text{ грн.}$$

Звідси:

$$C_p = 976.52 + 195.3 = 1171.82 \text{ грн.}$$

Прогноз обсягів продаж даного виробу. Отримана у таблиці 2.2. повна собівартість являє собою витрати виготовлення ($C_{пк}$) одиниці виробу задля даного року виробництва. Запропонуємо прогноз обсягів продажів даного виробу на другій стадії життєвого циклу виробу «Виробництво» із розподілом по роках (прогноз продажів передбачаємо на 4 роки). Характерні зони промислового випуску виробу представлені на рис.2.1.



| | | | | |
|-----|------|----------|--------|------|
| | | | | |
| Зм. | Арк. | № докум. | Підпис | Дата |

КБ 01. 15 000. 00 ДП ПЗ

Арк.

56

Рисунок 2.1. Зони промислового випуску виробу

У 2021 році обсяг продажів передбачається у розмірі 1000 шт. під замовлення.

У наступному році прогнозується збільшення обсягу продажів, тому витрати виробництва визначаємо по формулі:

$$C_{\text{пі}} = C_{\text{пі}} \left(\frac{A_i}{A_{i-1}} \right)^{0,23},$$

де

A_i – обсяг продажів (виробництва) в 1 рік розрахункового періоду, шт.;

A_{i+1} – обсяг продажів (I+1)-ом року, шт.;

0,23 – показник ступеня, що характеризує вплив росту обсягів виробництва на собівартість продукції.

Звідси випливає, що

$$C_{2021} = 813.77 \text{ грн.}$$

$$C_{2022} = C_{2021} * (1000/3000)^{0,23} = 813.77 * (1000/3000)^{0,23} = 629.6 \text{ грн.}$$

При відсутності росту обсягів виробництва, тобто коли обсяг продажів чи не змінюється чи зменшується у наступному році, витрати виробництва приймаються на рівне попереднього року.

$$C_{2024} = C_{2023} = C_{2022} = 629.6 \text{ грн}$$

Плановий прибуток, що включається у оптову ціну підприємства, задля наступного року при збільшенні обсягу продажів, визначаємо по формулі:

$$P_{i+1} = C_{m+1} * \frac{P}{100}$$

Звідси:

$$P_{2021} = 162.75 \text{ грн.}$$

$$P_{2022} = P_{2023} = P_{2024} = 629.6 * 0,1 = 62.96 \text{ грн.}$$

Оптову ціну підприємства у наступні роки розрахункового періоду визначаємо по формулі:

$$C_{O_{i+1}} = C_{n_{i+1}} + P_{i+1}$$

| | | | | | | |
|-----|------|----------|--------|------|-------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 57 |

Звідси:

$$Ц_{2021}=976.52 \text{ грн.}$$

$$Ц_{2022}= Ц_{2023}= Ц_{2024}=629.6+62.96 =692.56 \text{ грн}$$

Податкове зобов'язання визначається по формулі:

$$Пз_{i+1} = Ц_{O_{i+1}} * 0.2$$

Звідси:

$$Пз_{2021}=195.3 \text{ грн.}$$

$$Пз_{2022}= Пз_{2023}= Пз_{2024}=692.56 * 0,2=138.51 \text{ грн.}$$

Ціну реалізації одиниці продукції у наступні роки визначаємо по формулі:

$$Ц_{P_{i+1}} = Ц_{O_{i+1}} + Пз_{i+1}$$

Звідси:

$$Ц_{p2021}=1171.82 \text{ грн.}$$

$$Ц_{p2022}= Ц_{p2023}= Ц_{p2024}=692.56 +138.51 =831.07 \text{ грн.}$$

Вартісну оцінку значень поза розрахунковий період (P_T) визначаємо по формулі:

$$P_T = \sum_{i=t_p}^{t_k} A_i * Ц_{P_i} * \alpha_i$$

де t_p , t_k – відповідно розрахунковий та кінцевий рік розрахункового періоду;

$Ц_{pi}$ – ціна реалізації у i -тім році, грн.;

α_i – коефіцієнт, що включає фактор періоду, тобто коефіцієнт приведення різночасних витрат та значень до розрахункового року.

Коефіцієнт α_i визначаємо по формулі:

$$\alpha_i = |1 + E_H|^{t_p - t_i}$$

де E_H – норматив ефективності капітальних вкладень, $E_H = 0,1$;

t_p – розрахунковий рік розрахункового періоду;

t_i – i -ій рік розрахункового періоду, витрати та результати якого приводяться до розрахункового року.

Вартісну оцінку поза розрахунковий період визначаємо по формі, приведеної у таблиці 2.3.

| | | | | | | |
|-----|------|----------|--------|------|-------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 58 |

Таблиця 2.3. Розрахунок вартісної оцінки значень

| Найменування показника | Позначення | Розрахунок виробничого періоду | | | | Всього |
|--|---------------------------|--------------------------------|-----------|---------|----------|---------|
| | | 1-й | 2-й | 3-й | 4-й | |
| Обсяг продажів, шт | A_i | 1000 | 3000 | 2000 | 1000 | 7000 |
| Ціна реалізації, грн. | $Ц_{pi}$ | 1171.82 | 831.07 | 831.07 | 831.07 | - |
| Вартісна оцінка значень, грн. | $A_i * Ц_{pi}$ | 1171820 | 2493210 | 1662140 | 831070 | - |
| Коефіцієнт, що враховує фактор періоду | α_i | 0,91 | 0,83 | 0,75 | 0,68 | - |
| Вартісна оцінка значень із урахуванням фактору періоду, грн. | $A_i * Ц_{pi} * \alpha_i$ | 1066356.2 | 2069364.3 | 1246605 | 565127.6 | 4947453 |

Виробництво дає змогу одержати дохід поза 4 роки 1171.82 тис. грн.

Податкове зобов'язання, отримане при реалізації продукції по рокам розрахункового періоду складає:

$$ПЗ_{i+1}^r = ПЗ_i * A_i$$

$$ПЗ_{2024}^r = 195.3 * 1000 = 195300 \text{ грн.}$$

$$ПЗ_{2025}^r = 138.51 * 3000 = 415530 \text{ грн.}$$

$$ПЗ_{2026}^r = 138.51 * 2000 = 277020 \text{ грн.}$$

$$ПЗ_{2027}^r = 138.51 * 1000 = 138510 \text{ грн.}$$

Податковий кредит по рокам розрахункового періоду складає:

$$ПК_i^r = 0,2 * (В_m + В_{пк}) * A_i$$

$$ПК_{2024}^r = 0,2 * (110.35 + 342.1) * 1000 = 90490 \text{ грн.}$$

$$ПК_{2025}^r = 0,2 * (110.35 + 342.1) * 3000 = 271470 \text{ грн.}$$

$$ПК_{2026}^r = 0,2 * (110.35 + 342.1) * 2000 = 180980 \text{ грн.}$$

$$ПК_{2027}^r = 0,2 * (110.35 + 342.1) * 1000 = 90490 \text{ грн.}$$

Податкове зобов'язання, яке належить перахуванню до бюджету визначається по формулі:

$$ПДВ_i^6 = ПЗ_i^r - ПК_i^r$$

$$ПДВ_{2024}^6 = 195300 - 90490 = 104810 \text{ грн.}$$

| | | | | | | |
|-----|------|----------|--------|------|--------------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 59 |

3 РОЗДІЛ ОХОРОНИ ПРАЦІ І ТЕХНІКИ БЕЗПЕКИ

Безпека праці, спрямована на створення небезпечних та нешкідливих умов праці. На сучасному етапі розвитку виробництва вона набуває все більше важливого значення

Безпека праці на підприємстві спроможне бути на належному позначці тільки тоді, коли всебічно відповідає вимогам трудового законодавства, державним стандартам України, нормам та правилам, розроблених задля збереження здоров'я працюючих. Важливе місце при цьому належить виконанню організаційних вимог із охорони праці, а так само трудовій і виробничій дисципліні працюючих.

У даному розділі дипломного проекту розглядається питання охорони праці програміста при розробці системи дистанційного перевірки біологічних значень із захистом інформації на позначці TLS.

3.1 Аналіз небезпечних і шкідливих чинників, що впливають на працівника

Оператори та програмісти зіштовхуються із впливом таких фізично небезпечних та шкідливих виробничих факторів, як підвищений рівень шуму, підвищена температура зовнішнього середовища, відсутність чи недостатня освітленість робочої зони, електричний струм, статична електрика тощо.

На робочому місці програміста повинні бути створені умови задля безпечної і високопродуктивної праці.

3.2 Розробка заходів із охорони праці

3.2.1 Вимоги до виробничих приміщень задля експлуатації РС

Об'ємно-планувальні рішення будівель і приміщень задля роботи із РС містять відповідати вимогам ДСанПіН 3.3.2.007–98. Розміщення робочих місць із РС в підвальних приміщеннях, на цокольних поверхах заборонено. Площа на одне робоче місце становить не менше ніж 6,0 м², а об'єм – не менше ніж 20,0 м³.

| | | | | | | |
|-----|------|----------|--------|------|-------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 60 |

Виробничі приміщення повинні обладнуватись шафами задля зберігання документів, магнітних дисків, полицями, стелажми, тумбами тощо, із урахуванням вимог до площі приміщень. В приміщеннях із РС слід щоденно робити вологе прибирання. Приміщення із РС містять бути оснащені аптечками першої медичної допомоги.

При приміщеннях із РС містять бути обладнані побутові приміщення задля відпочинку під період роботи, кімната психологічного розвантаження. У кімнаті психологічного розвантаження слід передбачити встановлення пристроїв задля приготування й роздачі тонізуючих напоїв, а так само місця задля занять фізичною культурою (СНиП 2.09.04. – 87).

3.2.2 Гігієнічні вимоги до параметрів виробничого середовища приміщень із РС

Гігієнічні вимоги до параметрів виробничого середовища включають вимоги до параметрів мікроклімату, світло, шуму й вібрації, рівнів електромагнітного і іонізуючого випромінювання.

В виробничих приміщеннях на робочих місця із РС містять забезпечуватись оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря (ГОСТ 12.1.005–88, СН 4088–86) (табл. 3.1).

Таблиця 3.1. Норми мікроклімату задля приміщень із РС

| Пора року | Категорія робіт | Температура повітря, °С, не більше | Відносна вологість повітря, % | Швидкість руху повітря, м/с |
|-----------|-----------------|------------------------------------|-------------------------------|-----------------------------|
| Холодна | легка – Іа | 22-24 | 40-60 | 0,1 |
| | легка – Іб | 21-23 | 40-60 | 0,1 |
| Тепла | легка – Іа | 23-25 | 40-60 | 0,1 |
| | легка – Іб | 22-24 | 40-60 | 0,2 |

Позначці позитивних та негативних іонів в повітрі приміщень із РС містять відповідати санітарно-гігієнічним нормам №2152–80 (табл. 3.2).

Таблиця 3.2. Позначці іонізації повітря приміщень при роботі на РС

| Рівні | Кількість іонів в 1 см ³ повітря | |
|-----------------------|---|-----------|
| | n+ | n- |
| Мінімально необхідні | 400 | 600 |
| Оптимальні | 1500-3000 | 3000-5000 |
| Максимально допустимі | 50000 | 50000 |

Приміщення задля роботи із РС повинні мати природне і штучне світло відповідно до СНиП П-4-79/ Природне світло містить здійснюватись крізь світлові прорізи, орієнтовані переважно на північ чи північний схід, та забезпечувати коефіцієнт природної освітленості (КПО) не нижче, ніж 1,5%.

Штучне світло у приміщеннях із робочими місцями, обладнаними РС містить здійснюватися системою загального рівномірного світло. В виробничих і адміністративно-громадських приміщеннях, в разі переважної роботи із документами, допускається застосування системи комбінованого світло (крім системи загального світло, додатково встановлюються світильники місцевого світло)

Значення освітленості на поверхні робочого столу у зоні розміщення документів містить становити 300–500 лк, Коли це неможливо забезпечити системою загального світло, допускається використовувати місцеве світло. При цьому світильники місцевого світло слід встановлювати цим способом, аби не створювати бликів на поверхні екрана, а освітленість екрана містить не перевищувати 300 лк.

Як джерела світла задля штучного світло містять застосовуватись переважно люмінесцентні лампи типу ЛБ. В разі влаштування відбитого світло в виробничих і адміністративно-громадських приміщеннях допускається застосування металогалогенних ламп потужністю 250 Вт. Допускається застосування ламп розжарювання в світильниках місцевого світло. Позначці звукового тиску у октавних смугах частот, позначці звуку і еквівалентні позначці звуку на робочих місцях, обладнаних РС, містять відповідати вимогам СН 3223–85, ГОСТ 12.1.003–83, ГР 2411–81 (табл. 3.3).

Таблиця 3.3. Допустимі позначчі звуку, еквівалентні позначчі звуку та позначчі звукового тиску у октавних смугах частот

| Вид трудової діяльності | Рівні звукового тиску в дБ в октавних смугах із середньгеометричними частотами, Гц | | | | | | | | | |
|---|--|----|-----|-----|-----|------|------|------|------|--|
| | 31,5 | 63 | 125 | 250 | 500 | 1000 | 2000 | 4000 | 8000 | Рівні звуку, еквівалентні рівні звуку, дБА/дБАекв. |
| Програмісти ЕОМ | 86 | 71 | 61 | 54 | 49 | 45 | 42 | 40 | 38 | 50 |
| Оператори в залах обробки інформації на ЕОМ та оператори комп'ютерного набору | 96 | 83 | 74 | 68 | 63 | 60 | 57 | 55 | 54 | 65 |
| В приміщеннях для розташування шумних агрегатів ЕОМ | 103 | 91 | 83 | 77 | 73 | 70 | 68 | 66 | 64 | 75 |

Значення напруженості електростатичного поля на робочих місцях із РС (як в зоні екрана дисплея, так та на поверхнях обладнання, клавіатури, друкувального приладу) містять не перевищувати гранично допустимих поза ГОСТ 12.1.045–84, СН 1757–77 (табл. 3.4).

Таблиця 3.4. Допустимі параметри електромагнітних випромінювань електростатичного поля

| Види поля | Допустимі параметри поля | | Допустима поверхнева щільність потоку енергії (інтенсивність потоку енергії), Вт/м ² |
|--|-----------------------------------|---------------------------------|---|
| | за електричною складовою (E), В/м | за магнітною складовою (H), А/м | |
| Напруженість електромагнітного поля | | | |
| 60 кГц до 3 мГц | 50 | 5 | |
| 3 мГц до 30 мГц | 20 | – | |
| 30 мГц до 50 мГц | 10 | 0,3 | |
| 50 мГц до 300 мГц | 5 | – | |
| 300 мГц до 300 гГц | – | – | 10 |
| Електромагнітне поле оптичного діапазону в ультрафіолетовій частині спектра: | | | |
| УФ-С (220-280 нм) | | | 0,001 |
| УФ-В (280-320 нм) | | | 0,01 |
| УФ-А (320-400 нм) | | | 10,0 |
| у видимій частині спектра: | | | |
| 400-760 нм | | | 10,0 |
| в інфрачервоній частині спектра: | | | |
| 0,76-10,0 мкм | | | 35,0-70,0 |
| Напруженість електричного поля ВДТ | | | 20 кВ/м |

Значення напруженості електромагнітних полів на робочих місцях із РС містять відповідати нормативним значенням (ГДР №3206–85, ГДР N4131–86, СНК 5802–91, ГОСТ 12.1.006–84) (табл. 3.4).

3.2.3 Гігієнічні вимоги до організації, обладнання робочих місць із РС

Обладнання й організація робочого місця із БДТ містять забезпечувати відповідність конструкції всіх елементів робочого місця і їх взаємного розташування ергономічним вимогам із урахуванням характеру й особливостей трудової діяльності (ГОСТ 12,2.032–78, ГОСТ22.269–76, ГОСТ 21.889–76).

Конструкція робочого місця користувача РС містить забезпечити підтримання оптимальної робочої пози.

Робочі місця із РС слід так розташовувати відносно світлових прорізів, аби природне світло падало збоку, переважно зліва. При розміщенні робочих столів із РС слід дотримуватись таких відстаней: між бічними поверхнями БДТ – 1,2 м; з тильної поверхні одного РС до екрана іншого – 2,5 м.

Екран РС містить розташовуватися на оптимальній відстані теж очей користувача, що становить 600...700 мм, але не ближче ніж поза 600 мм із урахуванням розміру літерно-цифрових знаків та символів.

Розташування екрана РС містить забезпечувати зручність зорового спостереження в вертикальній площині під кутом +30* до нормальної лінії погляду працюючого.

Клавіатуру слід розташовувати на поверхні столу на відстані 100...300 мм з краю, звернутого до працюючого. В конструкції клавіатури містить передбачатися опорний прилад (виготовлений із матеріалу із високим коефіцієнтом тертя, що перешкоджає мимовільному її зсуву), котрий дає змогу змінювати кут нахилу поверхні клавіатури у межах 5... 15°.

Задля забезпечення захисту та досягнення нормованих рівнів комп'ютерних випромінювань треба застосовувати при екранні фільтри, локальні світлофільтри (засоби індивідуального захисту очей) і інші засоби захисту.

| | | | | | | |
|-----|------|----------|--------|------|--------------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 64 |

ВИСНОВКИ

Протягом виконання дипломного проекту розроблено систему дистанційного перевірки біологічних значень із захистом інформації на позначці TLS, котра представляє собою прилад задля перевірки температури і пульса користувача на основі платформи Arduino Нано, а так само програмне забезпечення із мережевими функціями і вбудованим захистом передаваних інформації. Реалізація функцій безпеки дозволила захистити особисті фізико-медичні показники з можливості їх перегляду несанкціонованими користувачами.

Пульс в проекті вимірюється поза поміччю імпульсного датчику Pulse Sensor, а температура – цифровим датчиком DS18D30. Результати вимірювань обробляються засобами модулю мікроконтролера Arduino Нано і поза поміччю послідовного інтерфейсу передаються на Wi-fi-блок ESP8285 Node-mcu, а із нього, в захищеному SSL-режимі крізь мережу Інтернет, завантажуються на Гугл-диск. Користувач зможе переглядати результати вимірювань із будь-якого мобільного чи стаціонарного комп'ютеру, на якому є Інтернет і доступ до Гугл Sheet на Гугл-диску. Змодельований прилад спроможне бути поміщеним в корпус відповідного розміру. Використані датчики можливо замінити на більш точні і компактні аналоги. При заміні давачів на інші програмний код не потребує великих змін, достатньо буде лише замінити назви давачів поза поміччю середовища розроблювання Arduino ICP.

Модернізація системи спроможне виконуватися поза такими напрямками:

- додати звукове супроводження задля того, аби прилад видавав сигнал кожен раз при отриманні імпульсу чи вмикав сигнал тривоги при досягненні небезпечних задля користувача значень;
- інформація визначення спроможні бути відправлені на мобільний телефон поза поміччю GSM- чи Bluetooth-модуля;
- додати давач визначення артеріального тиску;
- створити нейронну мережу, котра буде аналізувати стан і самопочуття користувача і попереджати про ризики виникнення кризів.

| | | | | | | |
|-----|------|----------|--------|------|-------------------------|------|
| | | | | | КБ 01. 15 000. 00 ДП ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 65 |


ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Сторчак К.П. Технології Інтернет речей. Навчальний посібник. – Київ: ДУТ, 2021. – 68 с.
2. Павловський О.М. Мікроконтролери та мікропроцесорна техніка. Лабораторний практикум: навч. посіб. / КПІ ім. Ігоря Сікорського, 2021
3. Пархоменко А.В. Методичні вказівки до лабораторних робіт з дисципліни «Вбудовані біомедичні системи та бездротові сенсорні мережі»: – Запоріжжя: НУ «Запорізька політехніка», 2019. – 77 с.
4. Петин В.А. Arduino и Raspberry Pi в проектах Internet of Things, ISBN: 978-5-9775-3646-2, 2016, 320с.
5. Жураковський Б. Ю., Зенів І.О. Технології інтернету речей: навч. посіб. Київ: КПІ ім. Ігоря Сікорського, 2021. – 271 с.
6. Дэвид Роуз, Дивовижні технології. Дизайн та інтернет речей, 336 с.
7. Кожемяко В. П., Тарновський М. Г., Павлов С. В. Схемотехніка сучасного приладобудування. Частина IV. – Вінниця: ВНТУ, 2003. – 136 с.
8. Кожемяко В. П., Павлов С. В., Тарновський М. Г. Оптоелектронна схемотехніка. Навч. посіб. – Вінниця: УНІВЕРСУМ-Вінниця, 2008. – 189 с.
9. Електронний курс Технології індустрії 4.0. Електронний ресурс:
<http://edu.asu.in.ua/course/view.php?id=4#section-1>
10. Arduino and Android through Wi-Fi. Електронний ресурс:
<http://electronics.stackexchange.com/questions/56633/simplest-way-to-connect-arduino-to-android-wirelessly>
11. Електронний ресурс: <https://www.arduino.cc/en/guide/introduction>
12. Електронний ресурс: <https://www.theengineeringprojects.com/2018/06/introduction-to-arduino-nano.html>
13. Електронний ресурс: Android Developers.
<http://developer.android.com/index.html>
14. Електронний ресурс: <https://arduino-master.ru/program/arduino-void-loop-i-void-setup/>
15. Електронний ресурс: <https://doc.arduino.ua/ru/prog/SoftwareSerialAvailable>.

Лістинг програми прийняття даних від сенсорів для Arduino Nano

```
#define USE_ARDUINO_INTERRUPTS true
#include <PulseSensorPlayground.h>
int val;
int tempPin = 1;
const int PulseWire = 0;
const int LED13 = 13;
float Threshold = 550;
PulseSensorPlayground pulseSensor;
void setup() {
  Serial.begin(9600);
  pinMode( A4, INPUT );
  pulseSensor.analogInput(PulseWire);
  pulseSensor.blinkOnPulse(LED13);
  pulseSensor.setThreshold(Threshold);
  if (pulseSensor.begin()) {
    Serial.println("Start");
    pinMode(LED, OUTPUT);
    digitalWrite(LED, HIGH);
  }
}
void loop() {
  val = analogRead(tempPin);
  float mv = ( val/1024.0)*50;
  Serial.print("TEMPRATURE = ");
  Serial.print(mv);
  Serial.print(" *C");
  Serial.println();
  int myBPM = pulseSensor.getBeatsPerMinute();
  if (pulseSensor.sawStartOfBeat()) {
    Serial.print("Pulse: ");
    Serial.println(myBPM/2.7);
  }
  delay(1000);
}
```

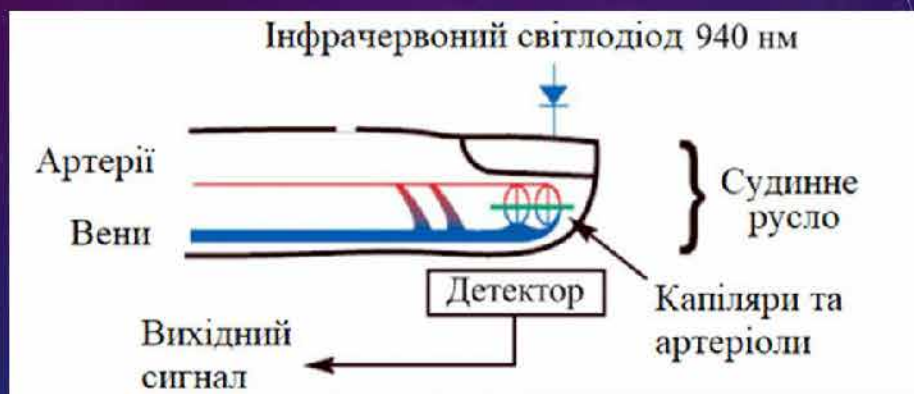
ДОДАТОК Б. Слайди мультимедійної презентації



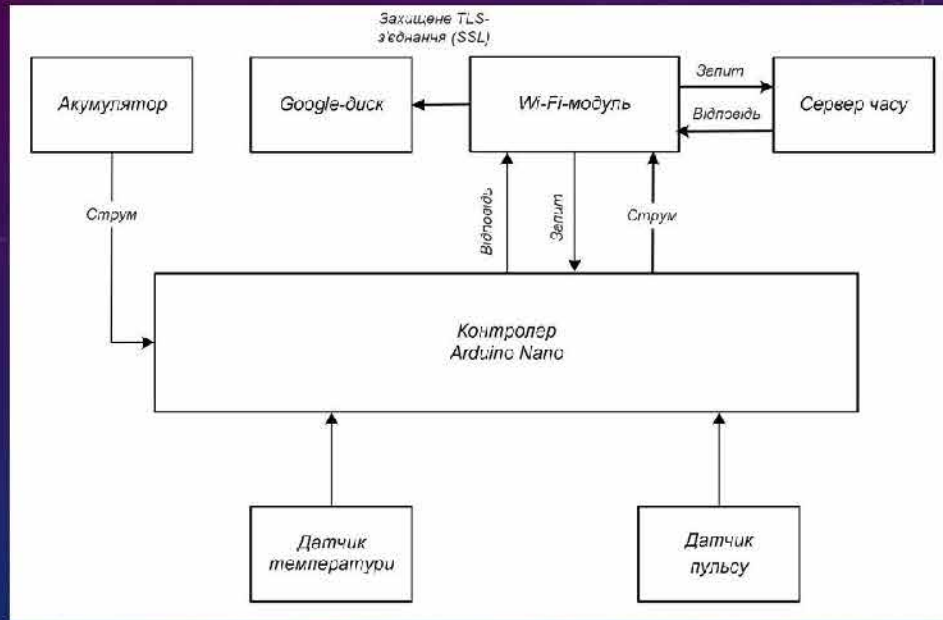
*РОЗРОБКА СИСТЕМИ ВІДДАЛЕНОГО
КОНТРОЛЮ ЖИТТЄВИХ ПОКАЗНИКІВ
З ЗАХИСТОМ ДАНИХ НА РІВНІ TLS*

*Сагайдак Кирило, гр. 4КБ-01
ВСП «ОТФК ОНТУ»*

ЗАГАЛЬНИЙ ПРИНЦИП РОБОТИ ДАТЧИКА ПУЛЬСУ



СТРУКТУРНА СХЕМА СИСТЕМИ ВІДДАЛЕНОГО КОНТРОЛЮ ЖИТТЄВИХ ПОКАЗНИКІВ З ЗАХИСТОМ ДАНИХ НА РІВНІ TLS



СКЛАД ПЛАТИ ARDUINO NANO

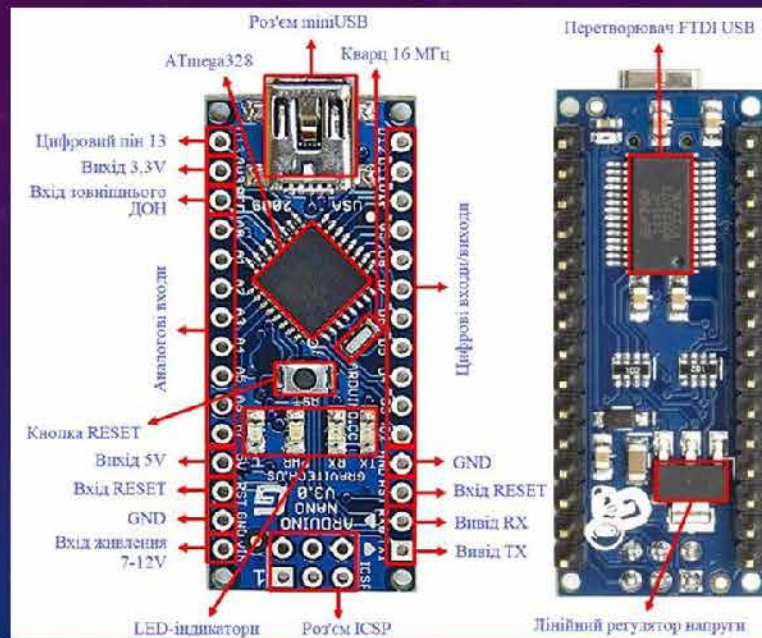
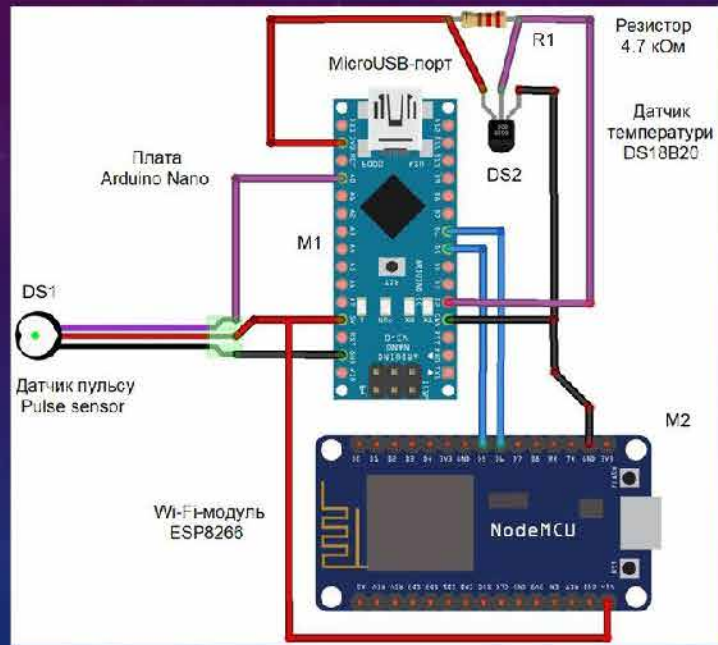
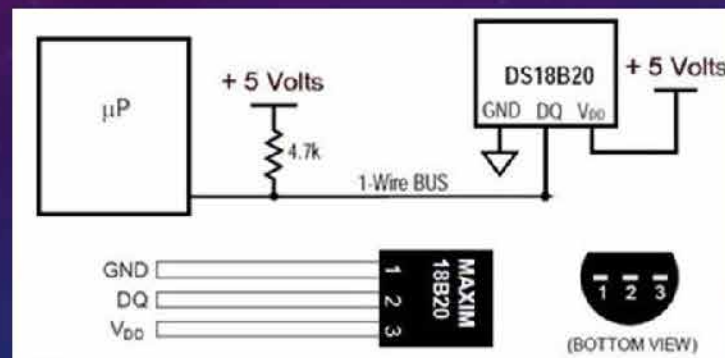


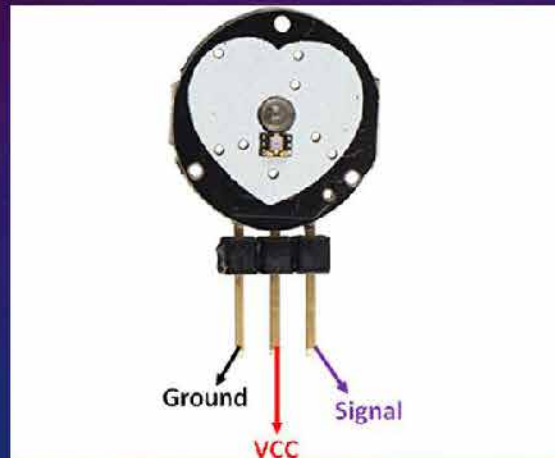
СХЕМА ПІДКЛЮЧЕННЯ ДАТЧИКІВ ДО ПЛАТИ ARDUINO



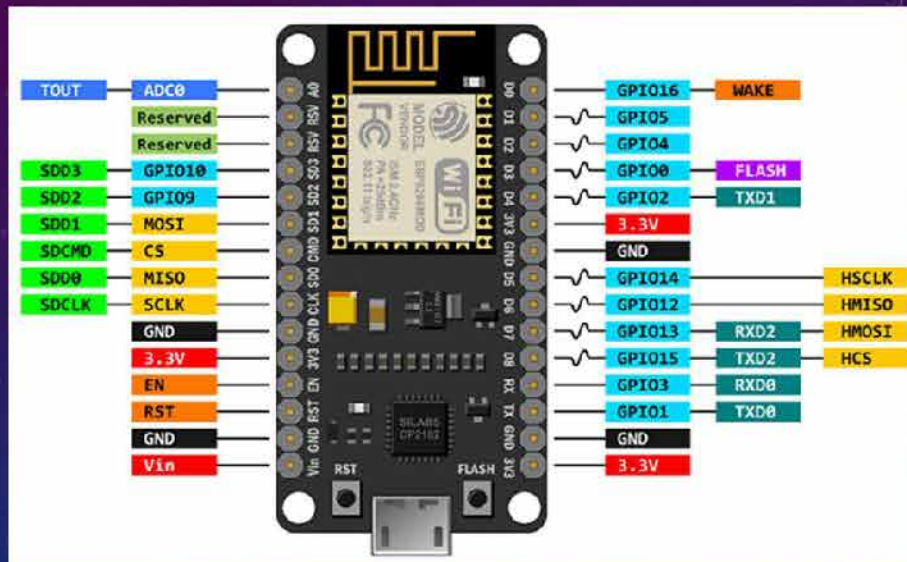
ПРИЗНАЧЕННЯ ВИВОДІВ ТА ТИПОВЕ ПІДКЛЮЧЕННЯ ДАТЧИКА DALLAS DS18B20



КОНСТРУКЦІЯ ТА ВИВОДИ ДАТЧИКА ПУЛЬСУ PULSE SENSOR



КОНСТРУКЦІЯ ТА ВИВОДИ WI-FI-MОДУЛЮ ESP8266

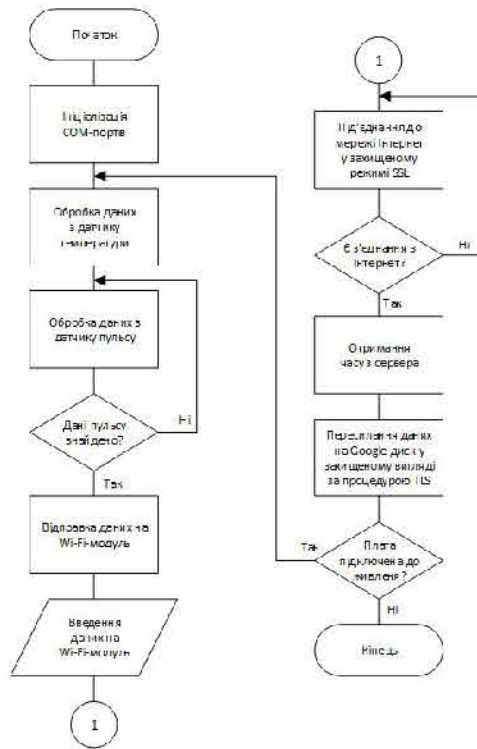


СЕРЕДОВИЩЕ РОЗРОБКИ PROGRAMINO



СЕРЕДОВИЩЕ РОЗРОБКИ ARDUINO IDE





БЛОК-СХЕМА АЛГОРИТМУ РОБОТИ СИСТЕМИ ВІДДАЛЕНОГО КОНТРОЛЮ ЖИТТЄВИХ ПОКАЗНИКІВ З ЗАХИСТОМ ДАНИХ НА РІВНІ TLS

СТАН МОНІТОРА ПОРТУ У ПРОЦЕСІ ВІДОБРАЖЕННЯ РЕЗУЛЬТАТІВ

```

    COM4
    Start pulse
    Pulse: 85.93
    TEMPERATURE = 21.12°C
    Pulse: 85.19
    TEMPERATURE = 21.25°C
    Pulse: 87.41
    TEMPERATURE = 30.12°C
    Pulse: 87.41
    TEMPERATURE = 31.44°C
  
```


ВИСНОВКИ

Система віддаленого контролю життєвих показників з захистом даних на рівні TLS дозволить визначити температуру та пульс людини, побудована на основі плати Arduino Nano з мережевими функціями та вбудованим захистом передаваних даних. Реалізація функцій безпеки дозволила захистити особисті фізико-медичні показники від можливості їх перегляду несанкціонованими користувачами.

Результати вимірювань з Wi-Fi-модулю у захищеному SSL-режимі через мережу Інтернет завантажуються на Google-диск. Користувач зможе переглядати результати вимірювань з будь-якого мобільного чи стаціонарного комп'ютеру, на якому є Інтернет та доступ до Google Sheets на Google-диску.

Використані датчики можна замінити на більш точні та компактні аналоги. При заміні датчиків на інші програмний код не потребує великих змін, достатньо буде лише замінити назви датчиків за допомогою середовища розробки Arduino IDE.

Модернізація системи може виконуватися за такими напрямками:

- додати звукове супроводження для того, щоб пристрій видавав сигнал кожен раз при отриманні імпульсу або вмикав сигнал тривоги при досягненні небезпечних для людини показників;
- дані вимірювання можуть бути відправлені на мобільний телефон за допомогою GSM- або Bluetooth-модуля;
- додати датчик вимірювання артеріального тиску;
- створити нейронну мережу, яка буде аналізувати стан та самопочуття людини та попереджати про ризики виникнення кризів.

ВІДГУК

керівника на дипломний проект здобувача (здобувачки) освіти
відділення комп'ютерних систем

Сагайдак Кирилові Володимировичу

(прізвище, ім'я та по батькові)

Спеціальність: 123 «Комп'ютерна інженерія»

Освітня програма: «Безпека комп'ютерних систем і мереж»

Тема дипломного проекту: Розробка системи віддаленого контролю
життєвих показників з захистом даних на рівні TLS

ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ

а) обсяг і якість виконання проекту (графічного матеріалу і розрахунково-пояснювальної записки) Дипломний проект виконано відповідно технічному завданню. Пояснювальна записка до дипломного проекту містить 75 сторінок. У пояснювальній записці описано процес розробки системи віддаленого контролю життєвих показників з захистом даних на рівні TLS та пристрою для моніторингу температури та пульсу людини на основі плати Arduino Nano. Графічна частина складається з 15 слайдів, оформлених у вигляді презентації, передбачених технічним завданням. Якість виконання пояснювальної записки та слайдів добра.

б) самостійність роботи над проектом: Протягом виконання дипломного проекту здобувач освіти Сагайдак Кирило поступово та послідовно виконував всі етапи, проявив ініціативу в створенні загальної концепції та реалізації роботи. Всі роботи здобувач освіти виконував самостійно, з оглядом на рекомендації керівника.

в) теоретична підготовка випускника (випускниці): Здобувач освіти Сагайдак Кирило під час роботи над дипломним проектом вивчив достатньо багато літературних та інтернет-джерел за даною тематикою.

Вважаю, що теоретична підготовка дипломника достатня і він готовий до захисту проекту.

г) вміння розв'язувати виробничі та конструкторські питання Під час виконання дипломного проекту здобувач освіти Сагайдак Кирило показав вміння організовано працювати над поставленим завданням, застосовувати знання у галузі програмування, електроніки, працювати у середовищі програмування Arduino IDE, складати схеми за допомогою сучасних комп'ютерних програмних засобів та САПР, таких як Microsoft Visio, Corel Draw.

Оцінка розрахункової частини Відмінно
Оцінка графічної частини Відмінно
Загальна оцінка Відмінно

Прізвище, ім'я, по батькові керівника дипломного проекту Шувалова Ірина Олегівна

Місце роботи і посада керівника дипломного проекту ВСП «Одеський технічний фаховий коледж ОНТУ», викладач циклової комісії комп'ютерних технологій та програмної інженерії

Підпис 

« 9 » 06 2024 р.

РЕЦЕНЗІЯ

на дипломний проект здобувача (здобувачки) освіти
відділення комп'ютерних систем

Сагайдак Кирила Виолодимировича

(прізвище, ім'я та по батькові)

Спеціальність 123 «Комп'ютерна інженерія»

Освітня програма «Обслуговування комп'ютерних систем та мереж»

Керівник дипломного проекту (роботи) Шувалова Ірина Олегівна

(прізвище, ім'я та по батькові)

Тема дипломного проекту (роботи) Розробка системи віддаленого контролю життєвих показників з захистом даних на рівні TLS.

Обсяг розрахунково-пояснювальної записки 76 сторінок

Обсяг графічної (презентаційної) частини 15 аркушів (слайдів)

ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ (РОБОТИ)

а) заключення про ступінь відповідності виконаного дипломного проекту завданню

Представлений на рецензію дипломний проект відповідає затвердженій темі та виконаний відповідно технічному завданню. Дипломний проект присвячений проблемі віддаленого контролю життєвих показників та складається з пояснювальної записки, додатку з програмним кодом та мультимедійної презентації, що містить приклади роботи програми.

б) характеристика виконання кожного розділу дипломного проекту

Пояснювальна записка складається з основного розділу (аналізу предметної області, проектування, апаратно-програмної реалізації, тестування апаратно-програмного комплексу), економічного розділу, розділу охорони праці та додатків. Перелічені розділи поетапно охоплюють розробку, виконані докладно та обґрунтовано. Розділ охорони праці містить загальну інформацію та вимоги до техніки безпеки оператора КТ. Економічний розділ проекту містить розрахунок витрат на НДР та реалізацію проекту.

в) оцінка якості виконання пояснювальної записки та графічної частини дипломного проекту

Графічна частина складається з 15 слайдів мультимедійної презентації, виконаної у програмному продукті MS PowerPoint, які містять ілюстративні схеми, скриншоти роботи програмного застосунку, передбачені технічним завданням. Пояснювальна записка виконана акуратно та у відповідності до норм. Якість виконання графічної частини проекту та пояснювальної записки відмінна, розробку виконано у повному обсязі.

г) перелік позитивних якостей дипломного проекту Реалізовано системо віддаленого контролю життєвих показників, що вимірює пульс і температуру.

Система створена на базі сучасної та популярної платформи Arduino.

Система зберігає данні у файл з історією.

Дані у системі захищені протоколом TLS.

д) основні недоліки дипломного проекту _____

Доцільно було б реалізувати вбудований дисплей для виводу поточних даних.

В деяких частинах пояснювальної записки присутні незначні помилки оформлення.

Оцінка розрахункової частини _____ Відмінно

Оцінка графічної частини _____ Відмінно

Загальна оцінка _____ Відмінно

Прізвище, ім'я, по батькові рецензента _____ Царьов Роман Юрійович

Місце роботи і посада рецензента _____ Державний університет інтелектуальних технологій і зв'язку, ст. викладач, зав. кафедри комп'ютерної інженерії та інформаційних систем



14 червня 2024 р.

Ім'я користувача:
Катерина Григоріївна Краснокутська

ID перевірки:
1016243067

Дата перевірки:
10.05.2024 21:52:33 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
10.05.2024 21:52:56 EEST

ID користувача:
100011688

Назва документа: **4КБ-01_Кирило_Сагайдак**

Кількість сторінок: **59** Кількість слів: **10963** Кількість символів: **77367** Розмір файлу: **2.68 MB** ID файлу: **1016026505**

15.6% Схожість

Найбільша схожість: **4.51%** з Інтернет-джерелом (<https://card-file.ontu.edu.ua/server/api/core/bitstreams/6c95086b-bff...>)

15.6% Джерела з Інтернету 345

Сторінка 61

Не знайдено джерел з Бібліотеки

0% Цитат

Вилучення цитат вимкнено

Вилучення списку бібліографічних посилань вимкнено

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 26

**ДОЗВІЛ
НА РОЗМІЩЕННЯ
ВИПУСКНОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ
(ДИПЛОМНОГО ПРОЕКТУ)
В ЕЛЕКТРОННОМУ РЕПОЗИТАРІЇ ВСП «ОТФК ОНТУ»**

Ми, що нижче підписалися,

Сагайдак Кирило Володимирович,
здобувач освіти гр. 4КБ-01, та

Шувалова Ірина Олегівна,
керівник дипломного проекту,

не заперечуємо щодо розміщення електронного варіанту пояснювальної записки до дипломного проекту фахового молодшого бакалавра на тему:

«Розробка системи віддаленого контролю життєвих показників з захистом даних на рівні TLS» (автор роботи – Сагайдак К.В., керівник роботи – Шувалова І.О.)

виконаного у ВСП «Одеський технічний фаховий коледж Одеського національного технологічного університету» в 2024 році, у повному обсязі в електронному репозитарії ВСП «ОТФК ОНТУ» для вільного доступу через мережу Інтернет.

Несемо відповідальність за ідентичність електронного та друкованого варіантів випускної кваліфікаційної роботи і даємо згоду на обробку персональних даних.

Виконавець



/ Сагайдак К.В. /

Керівник



/ Шувалова І.О. /

«10» червня 2024 р.