

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ**

До захисту допускається
Завідувач кафедри
комп'ютерних наук та інженерії
Скарга-Бандурова І.С.

_____ 2019 р.
« ____ » _____

МАГІСТЕРСЬКА РОБОТА

НА ТЕМУ:

**«МЕТЕОСТАНЦІЯ НА ARDUINO З ВІЗУАЛІЗАЦІЄЮ І АНАЛІЗОМ ДАНИХ
ПОГОДНОЇ СТАНЦІЇ»**

Освітньо-кваліфікаційний рівень «Магістр»
Спеціальність 123 «Комп'ютерна інженерія»

Науковий керівник роботи: _____
(підпис) (ініціали, прізвище)

Кардашук В. С.

Консультант з охорони праці: _____
(підпис) (ініціали, прізвище)

Критська Я.О.

Студент: _____
(підпис) (ініціали, прізвище)

Давиденко ОВ..

Група:

КІ-17ДМ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет інформаційних технологій та електроніки

Кафедра комп'ютерних наук та інженерії

Освітньо-кваліфікаційний рівень магістр

Спеціальність 123 «Комп'ютерна інженерія»

«ЗАТВЕРДЖУЮ»

Завідувач кафедри

комп'ютерних наук та інженерії

д.т.н, доц. Скарга-Бандурова І. С.

_____ 2019 року

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Давиденку Олександрю Васильовичу

(прізвище, ім'я, по-батькові)

1. Тема проекту (роботи): «Метеостанція на Arduino з візуалізацією і аналізом даних погодної станції » затверджена наказом по університету № 220/48 від «18» жовтня 2018 р.

2. Строк здачі студентом закінченого проекту (роботи): 10.01.2019 р.

3. Вихідні дані проекту (роботи): матеріали переддипломної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити):

1. Огляд методів та побудови компактних метеостанцій.
2. Моделювання візуалізація даних.
3. Побудова тестової метеостанції та візуалізація її роботи.
4. Моделювання роботи станції в польових умовах.
5. Охорона праці та безпека в надзвичайних ситуаціях.

5. Перелік графічного матеріалу (з точною назвою обов'язкових креслень):

_____ не передбачено

6. Консультанти роботи, з вказівкою розділів, що до них відносяться

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Основна частина	Кардашук В. С.		
Охорона праці та безпека в надзвичайних ситуаціях	Критська Я. О.		

7. Дата видачі завдання _____Керівник _____ Кардашук В. С.
(підпис)Завдання до виконання прийняв _____ Давиденко О.В.
(підпис)**КАЛЕНДАРНИЙ ПЛАН**

№ п/п	Найменування етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітки
1.	Отримання завдання, збір матеріалів	18.10.18- 24.10.18	
2.	Огляд літератури й обґрунтування необхідності дослідження	25.10.18–28.10.18	
3.	Дослідження методів методів та побудови компактних метеостанцій	29.10.18 – 28.11.18	
4.	Візуалізація роботи станції	28.11.18 –05.12.18	
5.	Модельовання роботи станції в польових умовах	05.12.18 – 19.12.18	
6.	Оформлення пояснювальної записки	19.12.18 – 01.01.19	
7.	Підготовка та подання магістерської роботи до захисту	01.01.19 – 10.01.19	

Студент _____
(підпис)Науковий керівник _____
(підпис)

АНОТАЦІЯ

Давиденко О.В. Метеостанція на Arduino з візуалізацією і аналізом даних погодної станції.

У магістерській роботі розроблений програмно-апаратний комплекс метеостанції за допомогою апаратної частини Arduino та програмній частині візуалізації та обробки даних за допомогою мови C++.

Даний пристрій призначений для компактної роботи в польових умовах з візуалізацією даних про погоду на конкретно заданій місцевості. Станція має зручне програмне та апаратне забезпечення, яке можна змінити залежно від потреб користувача. Налаштовування різних датчиків, які можуть вимірювати рівень вологості, температуру, визначати освітленість протягом дня, силу та напрям вітру.

Відображати значення на дисплеї, передавати дані на сервер управління через WiFi модуль, на якому дані зберігаються у базі даних.

Для зв'язку з інтернет-сервером був обраний WiFi модуль ESP8266. Для плати Arduino випускається Ethernet плата розширення, яка використовує шину даних і не є доцільною.

Компактна метеостанція на платформі Arduino має зручний інтерфейс, візуалізацію і передачу даних на мобільний пристрій, що полегшує життя і роботу людей у сільгоспідприємствах.

Ключові слова: Arduino, WiFi модуль ESP8266, C++, метеостанція, візуалізація, база даних.

THE ABSTRACT

Davydenko O.V. Weather station on Arduino with visualization and analysis of weather station data.

In the master's work the software-hardware complex of the weather station with the help of Arduino hardware and the program part of visualization and data processing with the help of C ++ language have been developed.

This device is intended for compact operation in the field with the visualization of weather data in a specified area. The station has convenient software and hardware, which can be changed depending on the needs of the user. Adjustment of various sensors that can measure humidity, temperature, determine the illumination during the day, the strength and direction of the wind.

Display the values on the display, transmit data to the control server via a WiFi module where the data is stored in the database.

To connect to an Internet server, the ESP8266 WiFi module was selected. For an Arduino board, an Ethernet expansion card that uses a data bus is not appropriate.

The Arduino Compact Weather Station has a convenient interface, visualization and data transfer to a mobile device that facilitates the lives and work of people in agricultural enterprises.

Keywords: Arduino, WiFi module ESP8266, C ++, weather station, visualization, database.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ПОБУДОВИ КОМП'ЮТЕРНОЇ СИСТЕМИ	10
1.1 Аналіз вимог побудови комп'ютерної системи	10
1.2 IT-технології в сільському господарстві	11
1.3 Різновид метеостанцій	16
1.4 Класифікація цифрових метеостанцій	17
1.5 Роль метеостанції сільськогосподарській справі	18
1.6 Висновки до розділу 1	25
РОЗДІЛ 2 РОЗГЛЯД АПАРАТНО-ПРОГРАМНОГО КОМПЛЕКСУ	26
2.1 ARDUINO	26
2.2 Список використаного обладнання	27
2.3 PostgreSQL	36
2.4 Візуалізація	37
2.5 InterSystems DeepSee	39
2.6 Медіанна фільтрація	39
2.7 Висновки по розділу 2	41
РОЗДІЛ 3 РОЗРОБКА МЕТЕОСТАНЦІЇ	42
3.1 Налаштування метеостанції	42
3.2 Створення бази даних PostgreSQL	53
3.3 Висновок до розділу 3	54
РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ. ЕКОЛОГІЯ	55
4.1 Загальні питання з охорони праці	55
4.2 Аналіз стану умов праці	56
4.2.1 Вимоги до приміщень	56
4.2.2 Вимоги до організації місця праці	57
4.3 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу	58
4.4 Електробезпека	60
4.5 Освітлення	61
4.6 Вплив на навколишнє середовища	68
4.7 Висновки до розділу 4	69
ВИСНОВКИ	70
ПЕРЕЛІК ПОСИЛАНЬ	71
Додаток А Лістинг програми для обладнання метеостанції	74
Додаток Б Презентація	95
Додаток В	101

ВСТУП

На сьогодні більшість обладнання українських метеостанцій вважається застарілим, хоча Україна є членом Всесвітньої Метеорологічної Організації, а більшість прогнозів складається завдяки закордонним метеоцентрам та їх обладнанню і програмному забезпеченню зі швидкою візуалізацією даних по мережі [1]. Розвиток технології не стоїть на місці, в тому числі і сам розвиток вимірювання та прогнози метеоданих.

Тим паче, що на деяких регіональних метеоцентрах розрахунок погодних даних може бути хибним або не точним на певній місцевості, що вказує на потрібний розвиток та заснування в містечках та селах невеликих чи компактних метеостанцій зі швидким обміном даних прогнозу погоди.

Точність прогнозу погоди залишається необхідним в більшості регіонів, а також попереджає про екстремальні погодні явища такі як повінь, злива, посуха, ураган та інші [2]. Для цього український гідрометеорологічний центр збирає обробку даних, супутникові картографічні знімки погоди та доводить всю метеорологічну інформацію:

- з метеостанцій по Україні передаються регулярні метеорологічні дані спостережень інтервалом кожні 10 годин;
- проводиться оповіщення про небезпечні та стихійні метеорологічні явища;
- робляться супутникові знімки території Європи кожні 20 хвилин;
- з пунктів радіозондування робиться збір даних про розподіл метеорологічних параметрів до висоти 30 кілометрів.

Але, як вказано вище, прогноз погоди робиться на глобальній місцевості, а не для окремих територій.

Своєчасний прогноз завдяки компактным метеостанціям допоможе в розвитку сільського господарства [3]. На сьогодні земний клімат відрізняється від того що був декілька десятків років тому. Така глобальна кліматична зміна створює значні екологічні проблеми. Згідно прогнозів Міжнародного наукового центру погоди попереджає що протягом наступного століття температура підвищиться на 2-5 градусів за Цельсієм. Наслідки глобального потепління можуть спричинити кліматичні зміни, що призведе до змін деяких екосистем.

Прогноз погоди може сильно впливати на розвиток сільського господарства. Аграрний сектор становить 15-17 % ВВП на рік. Цей відсоток може бути набагато більшим, але за відсталості чи не бажання введення нових та модернізації деяких секторів сільського господарства, власного споживання, на експорт іде незначна частина зібраного врожаю. Прикладом впровадження нових технологій є Нідерланди та Ізраїль, де площа цих країн не

перевищує Одеську область, та цей територіальний фактор не заважає цим країнам забезпечувати власне населення аграрною продукцією, навпаки допомагає економіці країн заробляти значні грошові статки від експорту врожаю. При чому дохідний продаж від експорту завдяки сільськогосподарським інноваціям значно перевищує дохід від реалізації продукції.

Ранні заморозки чи потепління, засуха чи злива, надмірна вологість впливає на час насадження та ріст культурних рослин, початок різноманітних польових робіт. Також це впливає на появу різноманітних захворювань чи шкідників, що вплине на кількість майбутнього врожаю або навіть його знищить. Наприклад, значна кількість опадів за короткий період часу та довготривала спека можуть викликати грибкове захворювання корінної системи чи плоду таких рослин як кукурудза, пшениця, соняшник, гречка, буряк, багатьох фруктових та квіткових культур. Оскільки значна кількість хворобливих збудників належить до групи широко спеціалізованих фітопатогенів, що дуже довгий час добре зберігаються на рештках рослин та ґрунті і є джерелом інфекцій для широкого кола культур під час сівозмін, це спричиняє та посилює рівень біологічного забруднення посівних площ та агроєкосистем в цілому. Такі фактори дуже сильно б'ють по гаманцю фермерів, агропідприємств та країни в цілому.

Метеостанції, які встановлені за закріпленими за ними польовими площами можуть цьому запобігти завдяки вчасному втручання аграріїв. Запобіжна прополка від сорняку та зменшення вологи в ґрунті (пари), обприскування рослин спеціальними розчинами та добривами.

Залишається необхідність у більш довгострокових прогнозах. Саме у цьому і допоможе розробка компактних метеостанцій. Вже в багатьох країнах на фермерських угіддях в цілях експерименту або для покращення роботи аграріїв встановлюються метеостанції чи різноманітні метеодатчики[4]. У деяких випадках процес настільки автоматизований, що можливо прогнозувати ріст рослин та кількість майбутнього врожаю.

Деякі метеостанції чи самі датчики виробляє не тільки спеціалізоване виробництво, а і звичайні люди у тому числі і самі фермери. У наш час знайти обладнання та інформацію стало набагато простіше.

Об'єкт дослідження – метеостанції на платформі Arduino.

Предмет дослідження – методи побудови компактної станції для прогнозування погоди, вимірювання вологості, температури та сонячного світла протягом дня.

Методи дослідження. Для побудови моделі станції та прогнозування погоди використовувались методи моделювання з візуалізацією даних у режимі реального часу. На аналізі методу створений апаратно-програмний комплекс, який перевіряє дані через

медійний фільтр для отримання більш точного прогнозування, введення прогнозу на сервер управління та запуску на мобільному додатку в майбутній дороботці проекту.

Наукова новизна магістерської роботи полягає в дослідженні методів побудови метеостанції на платформі Arduino с візуалізацією даних. На основі дослідження вироблені рекомендації щодо подальшого використання та проектування станції.

Структура і обсяг роботи.

Магістерська робота складається зі вступу, 4 розділів, висновків, переліку посилань із 49 найменувань на 2 сторінках, 2 додатків на 7 сторінках. Загальний обсяг роботи складає 110 сторінок. Магістерська робота містить 55 рисунків та 4 таблиці.

РОЗДІЛ 1

АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ПОБУДОВИ КОМП'ЮТЕРНОЇ СИСТЕМИ

1.1 Аналіз вимог побудови комп'ютерної системи

Апаратно-програмний комплекс є технічним рішенням алгоритму роботи системи. Управління здійснюється, як правило завдяки виконанню коду з певного базового набору команд (системи команд) [5], описаних в конкретній документації [6].

Апаратно-програмний комплекс складається з двох основних частин:

1. Апаратна частина (англ. Hardware) - пристрій збору і обробки інформації, наприклад, комп'ютер, біометричний детектор тощо. Під архітектурою апаратного забезпечення комп'ютера розуміються внутрішні компоненти комп'ютера і, підключені до нього, пристрої введення (такі як комп'ютерна миша і клавіатура) і пристрої виведення (монітор). Внутрішні компоненти комп'ютера разом являють собою обчислювальний і керуючий пристрій, об'єднаний шиною. Найбільш поширені позначення - процесор, оперативна пам'ять і жорсткий диск. Шина з'єднує окремі компоненти у складну систему, т оскільки без шини розрізнені деталі не змогли б функціонувати. Процесор або ЦПУ (центральний процесорний пристрій) визначає основну швидкість роботи комп'ютера. Вимоги до трактування ЦПУ і власної потужності процесора постійно збільшуються, тому що також ростуть вимоги техніки. Програмне забезпечення, яке пізніше встановлюється на комп'ютер, вимагає все більшої потужності процесора.

Термін архітектура поширюється на пристрій комплектної системи, оскільки окремі компоненти повинні бути один з одним узгоджені, щоб гарантувати безперебійне протікання процесу. В іншому випадку, в майбутньому, це призведе до проблем з комп'ютером. Якщо елемент не підходить до інших компонентів, то вони так само не можуть споживати повну потужність. Окремі складові блоки, такі, як процесор, оперативна пам'ять або шина, при складанні корпусу повинні складати одне ціле. Інакше підключене до процесора програмне забезпечення не зможе виконувати свої завдання в повній мірі і поводження з комп'ютером буде ускладнене.

2. Програмна частина (англ. Software) - спеціалізоване програмне забезпечення (як правило, написане компанією-виробником апаратної частини), обробляє і інтерпретує дані, зібрані апаратною частиною. Наприклад, вбудоване програмне забезпечення, операційна система.

Прикладне програмне забезпечення - програма, призначена для виконання певних користувальницьких завдань і розрахована на безпосередню взаємодію з користувачем.

Програмне забезпечення є одним з видів забезпечення обчислювальної системи, поряд з технічним (апаратним), математичним, інформаційним, лінгвістичним, організаційним, методичним та правовим забезпеченням.

1.2 ІТ-технології в сільському господарстві

Висвітлюються питання щодо прийняття новітніх технологій у сільському господарстві – точне землеробство, GPS-навігація, дрони на полях, супутникові технології в агробізнесі, робототехніка тощо.

Зараз ми живемо в часи, коли світ постійно змінюється навколо нас і в різних сферах, технології розвиваються і впливають на весь наш побут, роботу, зникають чи модернізуються старі і виникають нові види професій. Будується нова ера індустріальної революції: робототехніка, штучний інтелект, 3D-друк, комп'ютерні симуляції, хмарні технології, віртуальна і доповнена реальність, блокчейн і безліч інших технологій [7].

Тому і аграрна сфера не стоїть осторонь новітніх технологій. На сьогодні постійне впровадження новітніх розробок є реальною запорукою сталого розвитку сільського господарства.

Новітні технології в сільському господарстві є дуже важливою темою [8]. Сьогодні сучасний розвиток будь-якого підприємства, галузі і країни в цілому, залежить перш за все від впровадження і використання інновацій в виробництво, що сприяє як підвищенню продуктивності праці, так і економії різних видів ресурсів, а також значному підвищенню ефективності сільськогосподарського виробництва на мінімальних витратах, тобто за короткий час спостерігається використання новітніх технологій, значне збільшення якісних і кількісних показників ефективності, а також підвищення рентабельності в агропромисловому комплексі.

У сфері сільського господарства розвинених країн щороку з'являються нові умови і робляться значні зусилля для впровадження нових технологій, поліпшення виробництва і поліпшення умов праці в агропромисловому комплексі.

На практиці найбільш ефективні технології розглядаються в рамках прикладних комп'ютерних програм, що забезпечує оптимізацію розміщення сільськогосподарських культур в зональній системі сівозміни, економне використання земельних ресурсів, контроль якості вирощуваних продуктів і кормів, а також процес їх зберігання, та забезпечення меншого забруднення ґрунтів шкідливими речовинами.

В результаті кризи, викликані військовими діями, скороченням ринків збуту, недосконалістю законодавчої бази, відсутністю державних стимулів, інноваційною

діяльністю, обмеженням внутрішніх і зовнішніх джерел фінансування інновацій і неможливістю їх швидкої мобілізації, низьким рівнем інвестицій, привабливістю галузі і особливо, відсутністю інновацій та сучасних технологій виробництва і вирощування сільськогосподарських продуктів [9].

У сформованих умовах зростає затребуваність науки як генератора науково-технічного прогресу, а необхідність загального розвитку галузі й галузевої науки розуміється як її науково-технічний розвиток, впровадження сучасних інформаційних технологій [10].

До основних напрямків прискорення й підвищення ефективності науково-технічного прогресу АПК на сучасному етапі можна віднести:

- концентрацію зусиль аграрної науки на найбільш пріоритетних напрямках, здатних прискорити рішення поставлених перед галуззю технічних, науково-технологічних і соціально-економічних завдань;

- більш широке використання об'єктів інтелектуальної власності в аграрному виробництві з урахуванням реалізації прав на об'єкти інтелектуальної власності при вступі країни у СОТ;

- удосконалення економічного механізму функціонування аграрних наукових організацій з метою забезпечення раціонального включення науково-технічного блоку в процес ринкового реформування аграрного сектора;

- формування інформаційно-консультаційної служби АПК для сприяння підвищенню ефективності розвитку аграрного виробництва й поліпшенню соціально-економічних умов життя на селі за допомогою поширення нових знань і освоєння досягнень науки й техніки у виробництві (дорадництво);

- розвиток підприємництва в науково-технічній сфері АПК і формування на цій основі нових організаційних структур для здійснення науково-технологічної й інноваційної діяльності;

- удосконалення керування науково-технічною діяльністю з урахуванням переходу від адміністративних до демократичних процедур керування й розширення самостійності наукових організацій;

- поглиблення міжнародного науково-технічного співробітництва шляхом активної участі в роботі міжурядових і міжвідомчих угодах.

У сучасних умовах аграрна наука повинна зосередити увагу на більш важливих напрямках своєї діяльності, здатних прискорити рішення поставлених перед АПК завдань.

Ланцюжок: «наукове завдання - її рішення - одержання конкретного результату» повинен запрацювати. В аграрній науці є величезна кількість незатребуваних, сучасних технологій і розробок для різних галузей АПК. Закладений у них потенціал можна

реалізувати тільки через створення регіональних центрів дорадництва, шляхом формування регіональних інноваційних програм, активно залучаючи до фінансування венчурні фонди.

Вирішальними факторами виводу АПК із кризового стану, стійкого зростання сільськогосподарського виробництва є успішний розвиток і освоєння науково-технічних досягнень і інноваційних пропозицій, сучасних інформаційних технологій. Це резерв, що часто вимагає деяких матеріальних витрат, але від якого залежать в остаточному підсумку ефективність сільськогосподарського виробництва, використання інноваційних досягнень на практиці.

На жаль, сьогодні інноваційний потенціал АПК у країні використовується в межах 4-5%, у той час як у США цей показник становить 50%.

Частка наукомісткої продукції в АПК України не перевищує 1% від загального обсягу, а в розвинених країнах становить більше 20%.

Керування процесами розробки й освоєння науково-технічних досягнень і передового виробничого досвіду в сільському господарстві України в XXI столітті повинно набути визначального значення.

Створення ефективного інформаційного середовища в сільському господарстві, зокрема в інженерно-технічній системі, є актуальним завданням, як для сфери керування, так і для виробництва й науки.

Без інформаційної складової важко говорити про освоєння науково-технічних досягнень, і навпаки, при поширенні інформації без засвоєння у виробництві не можна говорити про ефективну інформаційну сферу. Тому проблема впровадження (освоєння) науково-технічних досягнень і сучасних інформаційних технологій завжди відігравала виняткову роль у сільському господарстві.

Вважається, що початок XXI століття є переходом від енергетики до інформаційних технологій. У світовому масштабі це характеризується значним перерозподілом ресурсів на користь подальшої глобальної інформатизації. До кризи в Україні відзначався ріст інвестицій у розвиток інформатизації, зокрема у сільське господарство.

У цілому, незважаючи на високу частку в економіці України виробничого сектора, загальний рівень інформатизації підприємств на сьогодні украй низький. Значною мірою це пояснюється загальним економічним спадом у країні, в умовах якого підприємства не можуть дозволити собі великих фінансових вкладень у технології, що підвищують ефективність керування й виробництва, нехай у недалекому, але майбутньому часі. Однак уже зараз вимальовуються групи підприємств, зокрема у сільському господарстві, здатні стати лідерами у використанні найсучасніших і дорогих інформаційних систем.

Активність підприємства на ринку ІТ залежить, у першу чергу:

– від виробничого потенціалу, що характеризує загальний стан виробництва (спад, підйом) і, як наслідок, актуальність потреби підприємства в інформатизації;

– наявності інвестицій, кількість і структура яких (довготерміновість проектів) визначає потенціал підприємств як замовників ІТ, а також вибір типу інформаційних систем - систем, націлених на оптимізацію технологій виробництва (наприклад, САПР) і (або) систем, призначених для оптимізації керування підприємством (Управлінські ІС). За відсутності достатнього фінансування інформатизація, як правило, починалася із САПР. Наступний крок, що йде далі - масове впровадження інформаційних систем керування ресурсами (матеріально-технічними, трудовими й т.п.) підприємства;

– експортного потенціалу, що визначає інтенсивність роботи на світовому ринку. Як правило, ці підприємства прагнуть до максимальної відповідності їхньої діяльності світовим стандартам.

У сільськогосподарському виробництві України до перебудови можна виділити три рівня комп'ютеризації: розробку систем автоматизації управлінської й фінансово-відомчої діяльності (АСУ); систем автоматизованого проектування (САПР); систем автоматизації технологічних процесів (АСУ ТП).

Розвиток кожного рівня здійснювався незалежно один від одного, відповідно до вимог розрізнених підрозділів. Для українського АПК була також характерна в більшій мірі автоматизація рутинної роботи, а не посилення інтелектуальних можливостей керівників. Реалізація ідеї інформатизації в сільгоспвиробництві стримувалася відсутністю «соціального замовлення», належного технічного рівня цієї галузі економіки й соціально-освітнього рівня споживачів для сприйняття цієї ідеї. Реальна ефективність від впровадження інформаційних технологій залишала бажати кращого. У перший період незалежності, самостійного господарювання й становлення ринкової економіки, ситуація ще більше погіршилася. Через відсутність у сільгоспвиробника засобів на технологічне переозброєння така «розкіш» як інформаційні системи була віднесена до числа глибоко другорядних і поступово стала зникати із практики керування підприємством.

Останнім часом у сфері сільського господарства все частіше з'являються умови й додаються значні зусилля із впровадження інформаційних технологій. Зростанню інвестицій в ІТ сприяв ряд факторів: триваючі реформи економіки, приватизація, зростання прямих іноземних інвестицій, значний попит підприємств малого й середнього бізнесу, а також індивідуальних користувачів в сільській місцевості на персональні комп'ютери й програмне забезпечення. Єдиний європейський ринок підштовхує компанії, що вступають у конкурентну боротьбу, до збільшення витрат на ІТ і вдосконалення інформаційної інфраструктури.

Використання інформаційних систем, у тому числі в АПК, у новинку тільки для нашої економіки - Європа, Америка, Японія вже не один десяток років розвивають і вдосконалюють у себе методики використання інформаційних систем і самі програмні продукти, на основі яких вони будуються. Для України сьогодні оптимальний вихід - це користуватися наробітками, зробленими на Заході. Вони вже випробувані, показали свою ефективність і можуть якщо вже не вивести наш АПК у лідери, то не дати йому остаточно вмерти особливо після входження України у СОТ. Що стосується сфери використання, то варто говорити скоріше не про спрямованість діяльності підприємства, що використовує у себе інформаційні технології, а про його розмір. Сучасний ринок ІТ пропонує рішення практично для будь-якого виробництва, починаючи, наприклад, від вирощування пшениці й закінчуючи виготовленням різноманітних продуктів її переробки. Однак для кожного такого рішення є обмеження з мінімального розміру підприємства, у рамках якого впровадження буде ефективним.

З розвитком інформаційних технологій пов'язані питання підвищення ефективності функціонування різноманітних дорадчих центрів різного масштабу та спрямування, що працюють практично в кожному регіоні. Для забезпечення консультаційної роботи Міністерству аграрної політики України необхідна кропітка робота зі створення спеціалізованих баз даних, залучення довідкових даних діючих правових систем, пошукових систем мережі Інтернет, банку знань, прикладних програм, що забезпечують оцінку поточної ситуації й прогноз її розвитку в різних сферах АПК.

Необхідність зменшення техногенних навантажень у сільському господарстві, впливів технологій, що впливають на навколишнє середовище, а також підвищення безпеки продуктів харчування при їх виробництві і зберіганні - основні фактори, що підвищують значимість інформаційних технологій. Істотне прискорення інформатизації сільського господарства - ключовий фактор майбутнього стійкого його розвитку.

Здійснення науково-технічного прогресу в сільському господарстві базується на властивих йому економічних і біологічних законах. Внаслідок цього науково-технічний прогрес в аграрному виробництві має свої особливості, які впливають зі специфіки збирання, зберігання, транспортування й переробки сільськогосподарської продукції.

Сільськогосподарські культури відрізняються біологічним складом, строками вегетації й ступенем використання продукції в їжу без переробки й після доробки. Отже розвиток науково-технічного прогресу повинен здійснюватися в напрямку більш повного й кращого використання біологічних особливостей конкретного виду сільськогосподарських культур. Особлива увага повинна бути звернена на розвиток галузей і виробництв по зберіганню, переробці й реалізації продукції.

Сільськогосподарська продукція є багатотоннажною, швидкопсувною й мало малотранспортабельною, що пов'язане з біохімічною будовою плодів, коренеплодів і інших видів культур і наявністю в їхньому складі великої кількості води. У процесі збирання, заготівлі й транспортування спостерігаються значні втрати маси продукції, погіршення її товарного виду й споживчих властивостей. З огляду на це, наукові дослідження повинні бути спрямовані на розробку перспективних способів зберігання, завантаження, транспортування й наступної доробки продукції.

Розробка й впровадження нових машин і технологій у сільському господарстві відкриває широкий шлях для ефективного використання трудових, матеріальних і фінансових ресурсів. Все це буде сприяти подальшому нарощуванню обсягів виробництва сільськогосподарської продукції й більш повному задоволенню зростаючих потреб населення при обмеженому використанні ресурсів суспільства.

Найбільш відомі на сьогодні технології реалізовані в рамках прикладних комп'ютерних програм. Це програми оптимізації розміщення сільськогосподарських культур у зональних системах сівозміни за розрахунками доз добрив; проведення комплексу землевпорядних робіт і керування земельними ресурсами; ведення державного земельного кадастру; забруднення ґрунтів; оцінки економічної ефективності виробництва; розробки технологічних карт обробки сільськогосподарських культур; регулювання режиму живлення рослин і мікроклімату в теплицях; контролю процесу зберігання зернопродуктів, якості продукції й кормів, що вирощуються, керування технологічними процесами в переробці й зберіганні продукції зерна й багато чого іншого.

1.3 Різновид метеостанцій

Метеостанція - спеціальний заклад [11], який має метеорологічну ділянку, що відповідає певним вимогам, на якому вчасно, відповідно до єдиної методології і в певній послідовності встановлюються стандартні прилади для безперервних метеорологічних вимірювань (спостережень за погодою і кліматом) і передаються зібрані дані в Гідрометеорологічний центр або іншим споживачам. Існують аналогові і цифрові метеостанції.

На класичній (аналоговій) метеостанції знаходяться:

- термометри для вимірювання температури повітря і ґрунту на різних глибинах;
- максимальні і мінімальні термометри для вимірювання максимальної та мінімальної температури повітря в період між стандартними часом спостереження;
- барометр для вимірювання атмосферного тиску;

- гігрометр для вимірювання вологості;
- анеморумбометр (або лопать) для вимірювання швидкості і напрямку вітру;
- опадомір для вимірювання опадів та інші.

1.4 Класифікація цифрових метеостанцій

На додаток до датчиків, перерахованих вище, дорожні метеорологічні станції використовують датчик температури поверхні і датчик температури на глибині 30 см (під кришкою), а також контролер і модуль GPRS для передачі даних в інформаційні центри [12]. Для інформування водіїв про погодні умови використовуються інформаційні щити із зазначенням температури поверхні і повітря. Також на дисплеї можуть відображатися попередження (WET ROAD, SIDE WIND і т. д.)

Лісові метеостанції слугують для запобігання можливості лісових пожеж. Найчастіше ці метеостанції працюють на батарейках. Станції збирають кліматичні дані, такі як вологість деревини, ґрунт і температура на різних висотах лісу. Дані обробляються і моделюється карта пожежної активності, яка допомагає пожежним легше впоратися з можливим загорянням або запобігти поширенню вогню.

Гідрологічні метеостанції проводять метеорологічні та гідрологічні спостереження за погодними умовами океанів, морів, річок, озер і боліт. Такі метеостанції розташовані на континентах, на морських плавучих станціях, а також на річкових, озерних і болотних станціях спостереження.

Внутрішні домашні метеостанції з'явилися на ринку порівняно недавно. Предками побутових метеостанцій були звичайні барометри. Функціональність домашньої метеостанції аналогічна метеорологічній станції, обробляється тільки набагато менше даних, що надходять від одного або декількох датчиків, встановлених за вікном і в інших приміщеннях. Домашні метеостанції показують температуру в приміщенні, температуру поза приміщенням, вимірюють вологість, атмосферний тиск і на основі обробки даних процесором формують прогноз погоди на день. Працюють як від електричної мережі, так і від змінних батарей.

В останнє десятиліття спостерігається швидке зростання числа виробників і моделей домашніх метеостанцій, що поставляються на ринок. Станом на початок 2019 року існують сотні моделей цифрових домашніх метеостанцій на будь-який смак, з різним набором функцій користувача і вартістю від декількох десятків до сотень доларів США. Загальна тенденція полягає в використанні бездротових радіо датчиків, а також функції корекції часу метеостанції з використанням радіосигналів атомного годинника. В Європі найпередовіші (і

найдорожчі, вартістю не менше 60-70 євро) домашні метеостанції крім цих параметрів визначають швидкість і напрям вітру, кількість опадів, рівень УФ-випромінювання, а також дозволяють розподіляти дані серед інших користувачів (в Інтернеті або за допомогою любительського радіозв'язку) і, крім використання власних бездротових датчиків температури і вологості, отримують дані про погоду протягом 3-5 днів із використанням супутникових сигналів (технологія WeatherDirect). Останні найбільш поширені в Німеччині, зокрема, на станціях: Aura, Galileo, Twister 300 і аналогічних компаніях TFA-Dostmann .

1.5 Роль метеостанції сільськогосподарській справі.

Поява метеостанцій на полі пов'язана з природним бажанням фермерів, якщо ви не контролюєте погоду, то хоча б прогнозуєте її зміни на короткий проміжок часу. Сьогодні існує цілий список метеостанцій, які можна купити, однак варто звернути увагу на деякі їх особливості, які, як правило, приховані в деталях [13].

Якщо ви порівняєте професійну метеостанцію і любительську метеостанцію, то, крім різниці в ціні на порядок, в професійній погодній середовищі ви отримаєте скомпільовану погодну ситуацію на полі і навколо нього в радіусі декількох кілометрів. Ви можете накопичувати цю інформацію і представляти її у вигляді графіків. А аматорська станція, в кращому випадку, покаже ситуацію на ЖК-екрані, зможе зберегти інформацію на накопичувачі (флеш-пам'яті) зображений на рисунку 1.1.



Рисунок 1.1 - Професійна метеостанція на поле

Професійна станція встановлюється на відкритій місцевості, як правило, охороняється та відображає дані як на рисунку 1.2. Любительська станція може бути встановлена на підвіконні, а датчики - поза домом.

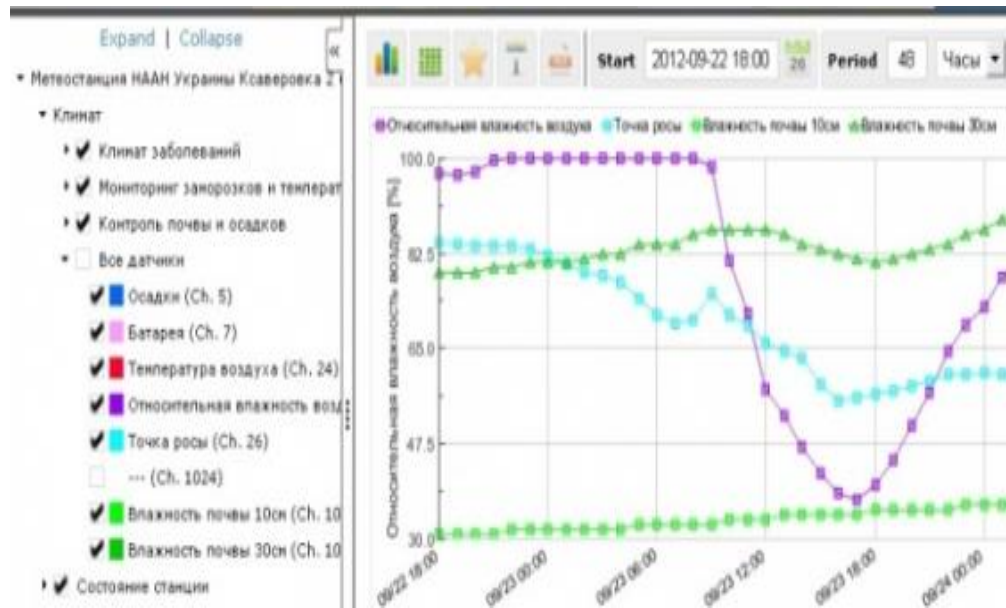


Рисунок 1.2 - Основні показники професійної метеорологічної станції

Однак професійна станція в 90% випадків (інші 10% можна придбати додатково) оснащена сонячними батареями, які заряджають її батареї і можуть працювати автономно, без втручання людини, протягом декількох років, періодично додаючи нові дані в пам'ять сервера. При покупці аматорської станції користувач обирає між батареями і домашнім джерелом живлення.

На професійній метеостанції первинний набір датчиків настільки широкий, наскільки це необхідно для професійної роботи аграріїв на полі, чим не може "похвалитися" аматорська станція. Для неї датчики потрібно купувати додатково і тільки зі списку, який виробник пропонує для звичайного (домашнього) використання. Аматорська метеостанція представлена на рисунку 1.3.

Вище наведено типовий інтерфейс професійної метеостанції, використовуваної для сільського господарства (рис. 1.2), з використанням сонячної батареї представлений на рисунку 1.4.

Як бачите, станція може одночасно вимірювати до семи показників, давати прогноз погоди на кілька днів вперед, відстежувати вміст вологи в ґрунті відразу в два шари (0-10, 10-30 см). Може бути додатково укомплектованою датчиками самодіагностики станції. Станції цього класу інтегровані в єдину мережу Агрокластера і мають важливе значення при плануванні відвезення техніки в поле, початку посівного сезону і контролю вологості ґрунту протягом вегетаційного періоду. У разі нестачі вологи активізуються іригаційні системи під їх контролем.

В цілому, такі станції можуть вести безперервний збір показників і передавати їх по бездротовому зв'язку в будь-яку точку світу. Вони заздалегідь попереджають про сприятливі

погодні умови для розвитку хвороб і шкідників на полі, визначають швидкість випаровування і транспірації для конкретних культур, автоматизують і контролюють дощувальні і іригаційні системи тощо.



Рисунок 1.3 - Приклад домашньої метеостанції

Однак слід сказати про одну особливість професійних метеостанцій. Майже всі метеостанції професійного рівня не відправляють дані, отримані з полів, безпосередньо на сервер власника, а відправляють цю інформацію виробнику, який, як правило, знаходиться за кордоном. Користувач може отримати цю інформацію, зареєструвавшись і отримавши пароль на сайті виробника цієї станції. Здавалося б, дрібниця, але спотворювати, наприклад, таку інформацію дуже легко, або тимчасово відключати сайт для профілактичного обслуговування. І найголовніше, такі станції постійно з'являються по всій Україні, а стратегічна інформація з первинними метеорологічними показниками йде за кордон для аналізу. Якщо потенційний покупець не бентежиться, то вибір явно на користь професійній метеостанції.



Рисунок 1.4 - Приклад комплектації професійної метеостанції сонячною батареєю.

Типовим є те, що в майбутньому кожен аграрій чи фермер буде мати власну метеостанцію. Оскільки існує безліч різноманітних компактних станцій прогнозу погоди, вони мають одну мету - виконати роботу, хоча і відрізняються за функціоналом та обладнанням, а також засобам безпеки апаратного програмного забезпечення. У деяких випадках існує щомісячна платна підписка на цю послугу. Така дія відштовхує певний відсоток клієнтів, тому вони вимушені звернути увагу на вітчизняні аналоги. Щомісячна платна підписка сприятиме розвитку та фінансуванню сільського господарства, збільшенню відсотків врожайності та подальшому розвитку технологій для сільського сектора промисловості [16].

Приведена порівняльна таблиця 1 для схожих за параметрами метеостанцій.

Таблиця 1 - Порівняння аналогів.

Метеостанція	Нива 1+	National Geographic 3-View black	La Crosse WS1650
Ціна	Від 1500 грн.	Від 2600 грн.	Від 5000 грн.
Апаратне забезпечення комплексу	Середня	Середня	Висока
Програмне забезпечення комплексу	Слабке	Середня	Висока
Кількість датчиків	Від 4 на одну станцію	Від 8 на одну станцію	Від 8 на одну станцію
Зручність використання	Легка	Середня	Середня

Більш точніша порівняльна характеристика National Geographic 3-View black та La Crosse WS1650 приведена нижче.

National Geographic 3-View black

Невелика за розмірами триекранна метеостанція Bresser National Geographic 3[17]-View зберігає і обробляє дані про погоду, призначена для побутових потреб. Вона здатна відображати не тільки кімнатну температуру. Зовнішній виносний датчик метеостанції може показувати вуличну температуру в реальному часі, а також прогнозує утворення ожеледі, підвищення або зниження температури. Кнопки управління знаходяться на передній панелі, з їх допомогою можна швидко і просто налаштувати метеостанцію. На трьох екранах відображаються символічні значки вуличної погоди, температура і час.

Зовнішній датчик необхідний для збору інформації про погодні умови. Він може встановлюватися як всередині приміщення, так і на вулиці. У комплекті передбачено 1 весняний датчик, але їх можна підключити до 3 штук. Бездротове з'єднання дозволяє взаємодіяти з основним блоком на відстані до 30 метрів. У додаткові можливості метеостанції входять підсвічування дисплея, будильник з повтором сигналу, календар і можливість відображення часу в 12 або 24-годинному форматі.

Прогноз погоди:

Символьний (ясно, мінлива хмарність, хмарно, дощ, злива, сніг, снігопад)

Температура в приміщенні:

- Одиниці виміру: °C;
- Діапазон вимірювання: від -5 ° C до + 50 ° C;
- Дозвіл виміру: 0.1 ° C;

- Точність виміру: +/- 1 ° C;
- Фіксація Min / Max значень температури з реєстрацією часу і дати.

Температура:

- Одиниці виміру: °C;
- Діапазон вимірювання: -20 ° C до + 50 ° C;
- Дозвіл виміру: 0.1 ° C;
- Точність виміру: +/- 1 ° C;
- Фіксація Min / Max значень температури з реєстрацією часу і дати.

Вологість:

- Одиниці виміру: % RH;
- Діапазон вимірювання: від 10% до 95%;
- Дозвіл виміру: 1%;
- Точність вимірювання: +/- 5%.

Дана таблиця показує приблизну звірку декількох метеостанцій у тому числі і станцію проекту. Приблизна похибка прогнозу становить від 17 до 28 відсотків на невеликій площині використання, при тому що станція має власні прилади для заміру температури, вологості, надлишку світлового дня, атмосферного тиску тощо, а не бере дані з клієнт-сервера бази даних прогноза погоди даної станції. Оновлення даних в середньому повинно відбуватися кожні 5 годин, для онлайн станцій кожні 3 години.

La Crosse WS1650 [18]

Забезпечує вимір таких метеопараметрів, як величина атмосферного тиску, швидкість і напрям вітру, рівень опадів, температури і відносної вологості повітря. Вбудована функція тривожних сповіщень попередить вас про вихід реальних показників за вказані межі. Всі датчики скомпоновані в єдиному блоці, що значно спрощує і прискорює процес установки і монтажу.

Сучасний лаконічний дизайн і найширший спектр функцій дозволить з рівним успіхом використовувати цю метеостанцію як в квартирі, так і на присадибних ділянках. Прогноз погоди на 12 годин. Символьний прогноз погоди. На дисплеї відображаються піктограми (ясно, мінлива хмарність, хмарно, дощ, злива / гроза, снігопад).

Атмосферний тиск:

- Одиниця виміру: hPa, inHg, mmHg (міліметри ртутного стовпа).
- Діапазон вимірювання атмосферного тиску: від 850 hPa до 1050 hPa (від 25.10 inHg до 31.00 inHg; від 637.5 mmHg до 787.5 mmHg).
- Дозвіл виміру: 0.1 hpa (0.003 inHg, 0.08 mmHg)
- Точність вимірювання: +/- 5 hpa (0.015 inHg, 0.38 mmHg)

- Діаграма зміни атмосферного тиску за останні 24 години
- Можливість відображення тиску над рівнем моря і реального тиску на місці установки метеостанції

- Автоматична реєстрація мінімальних і максимальних значень атмосферного тиску

Температура:

- Одиниці виміру: C або °F
- Діапазон вимірювання: -40 ° C до + 60 ° C (-40 ° F до 140 ° F)
- Дозвіл виміру: 0.1 ° C або 0.2 ° F
- Точність виміру: +/- 1 ° C (+/- 2 ° F)
- Оновлення показань кожні 47 сек
- Автоматична реєстрація мінімальних і максимальних значень температури
- Звуковий сигнал при виході температури за встановлені значення

Вологість:

- Одиниці виміру: % RH
- Діапазон вимірювання: від 1% до 99%
- Дозвіл виміру: 1%
- Точність вимірювання: +/- 7%
- Оновлення показань кожні 47 сек
- Індикація тенденції зміни вологості
- Автоматична реєстрація мінімальних і максимальних значень вологості
- Звуковий сигнал при виході рівня вологості за встановлені значення

Також в деякі системи можливо ввести власні корективи чи оновити програмний комплекс. Наприклад вирахувати точку роси. Точка роси - це температура, за якої повітря (при охолодженні) насичується водяною парою [19]. Що вище вологість повітряної маси, тим вище точка роси. Точка роси також іноді використовується в якості запобігання дискомфорту. Коли точка роси перевищує 65 градусів (+ 18C), багато людей починають говорити, що повітря «липке». У точці роси 70, багато хто відчуває себе некомфортно. Загальна оцінка точки роси, TDP, може бути знайдена з використанням формул 1 та 2 і констант, як показано нижче:

$$tdp = \frac{c * \gamma}{(b - \gamma)} \quad (1)$$

$$\gamma = \ln(\text{humidity}/100) + \frac{b * tempC}{(c + tempC)} \quad (2)$$

де, tdp - точка розрахунку роси; γ - розрахунок вологості.

1.6 Висновки до розділу 1

У першому розділі розглянуто основні проблеми щодо погоди для сільського господарства, роль метеостанції та її функціонал для покращення роботи аграріїв. Проведений порівняльний аналіз між схожими за характеристиками станціями, приведений розрахунок таких критеріїв як точка роси. На сьогодні відсоток нових технологій в тому числі IT-технологій в сільському господарстві України ще не дуже значний. Але кожного року він збільшується, так само як і збільшує свою роль в цьому секторі української економіки. Поява компактних метеостанцій на сільхоз угіддях є тому найкращим прикладом. Наші аграрії йдуть у ногу з часом і не хочуть відставати від своїх колег з інших розвинутих країн. Збільшити врожайність, зменшити витрати на паливо, добрива, покращити якість сільськогосподарської продукції, стимулювати розвиток агропідприємств. Кліматичне та географічне розташування країни дає гарну спробу в цьому переконатися. Хочь які родючі не були українські чорноземи, без догляду та вчасного втручання врожайність рослинних культур може бути під загрозою. Розвиток вітчизняних метеостанцій, професійних чи аматорських, в тому числі і проект на платформі Arduino допоможуть фермерам.

РОЗДІЛ 2

РОЗГЛЯД АПАРАТНО-ПРОГРАМНОГО КОМПЛЕКСУ

2.1 ARDUINO

Данна станція буде розроблена на платформі Arduino та Arduino-сумісному програмно-апаратному комплексі, з додаванням допоміжних датчиків для заміру даних про погоду.

Arduino - це бренд апаратного і програмного забезпечення для створення простих систем автоматизації та робототехніки, призначений для непрофесійних користувачів. Програмне забезпечення складається з вільної програмної оболонки (IDE) для написання програм, їх компіляції і апаратного забезпечення для програмування. Апаратне забезпечення являє собою набір монтованих друкованих плат, що продаються як офіційними, так і сторонніми виробниками. Повністю відкрита архітектура системи дозволяє вільно копіювати або доповнювати лінійку продуктів Arduino [20].

Під брендом Arduino доступні кілька плат з мікро контролером (англійські плати) і карти розширення. Більшість плат з мікро контролером оснащені мінімально необхідним набором обв'язки для нормальної роботи мікроконтролера (стабілізатор потужності, кварцовий резонатор, ланцюги скидання тощо). Arduino і Arduino-сумісні плати спроектовані таким чином, щоб при необхідності їх можна було розширити шляхом додавання нових компонентів в пристрій. Ці карти розширення підключаються до Arduino через роз'єми, встановлені на них. Існує ряд плат з уніфікованою конструкцією, яка дозволяє структурно жорстко з'єднувати плату процесора і плати розширення, які можна укласти через штирові лінії. Крім того, виробляються дошки зменшених розмірів (наприклад, Nano, Lilypad) і спеціальні конструкції для задач робототехніки. Незалежні виробники також виробляють широкий спектр різних датчиків і виконавчих механізмів, які в тій чи іншій мірі сумісні з базовою структурою Arduino.

Концепція Arduino не включає корпус або монтажні конструкції. Розробник самостійно обирає спосіб установки і механічного захисту плат. Сторонні виробники випускають комплекти роботизованою електромеханіки, орієнтовані на роботу спільно з платами Arduino. Мова програмування Arduino є стандартною C++.

Інтегроване середовище розробки Arduino (IDE) - це кроссплатформний додаток (для Windows, macOS, Linux), написаний на мові програмування Java. Він використовується для написання і завантаження програм на плату Arduino. [21]

Вихідний код для середовища IDE випущений під загальнодоступною ліцензією GNU версії 2. Arduino IDE підтримує мови C і C ++, використовуючи спеціальні правила структурування коду. Arduino IDE надає бібліотеку програмного забезпечення з проекту Wiring, яка надає безліч загальних процедур введення і виведення. Для написаного користувачем коду потрібні тільки дві базові функції для запуску ескізу і основного циклу програми, які скомпільовані і пов'язані з заглушкою програми main () в виконувану циклічну виконавчу програму за допомогою ланцюжка інструментів GNU, також включеного в дистрибутив IDE. В Arduino IDE використовується програма avrdude для перетворення коду, в текстовий файл в шестнадцатизначному кодуванні, яке завантажується в плату Arduino програмою-завантажувачем у вбудованому програмному забезпеченні плати.

2.2 Список використаного обладнання

Повний список устаткування, яке використовується в установці:

- Arduino Uno з Weather shield і датчиками погоди;
- 2x XBee Shield і 2x модулів XBee з розширеним діапазоном від Digi International;
- Arduino Mega з Ethernet Shield;
- 2 комплекти для монтажу Arduino;
- 2x 5В трансформатор для живлення Arduino;
- XBee Explorer Dongle для програмування передавача XBee;
- допоміжний датчик температури та вологості DHT11;
- барометр BMP280.

Барометр BMP280 [22]

Модуль вимірювання атмосферного тиску заснований на датчику BMP-280 (рис.2.1) від BOSCH. Цей датчик є покращеною версією датчика BMP180 і відрізняється від нього меншими розмірами, зниженим енергоспоживанням, високою точністю роботи і наявністю точного заводського калібрування і двома послідовними інтерфейсами: I2C і SPI.

Від попередніх моделей (BMP085 і BMP180) датчик відрізняється трьома режимами роботи:

- SLEEP - режим зниженого енергоспоживання
- FORCED - режим, аналогічний, режиму роботи датчиків BMP085 і BMP180. За команди контролера датчик виходить з режиму сну, проводить вимірювання, видає результати вимірювання контролера і переходить в режим зниженого енергоспоживання

– NORMAL - унікальний для цього датчика режим. Датчик самостійно прокидається, проводить вимірювання тиску і температури і засинає. Всі тимчасові параметри цього режиму програмуються незалежно. Зчитувати дані в цьому режимі можна в будь-який час.

У датчику передбачена фільтрація результатів вимірювань з налаштуванням таких параметрів фільтрації:

- OVERSAMPLING для температури (16,17,18,19,20 біт);
- OVERSAMPLING для тиску (16,17,18,19,20 біт);
- TSB - час між між вимірами (0.5,62.5,125,250,500,1000,2000,4000 мс);
- FILTER_COEFFICIENT - коефіцієнт фільтрації.

Характеристики:

- напруга живлення: від 1.71 В до 3.6 В;
- макс швидкість I2C інтерфейсу: 3.4 МГц;
- струм: 2.7мкА при частоті відліків в 1 Гц;
- інтерфейс: I2C, SPI (4 Провід), SPI (3 Провід);
- калібрування: заводська;
- рівень шуму: до 0.2 Па (1.7 см) і 0.01 температури;
- діапазон вимірюваного тиску: від 300hPa до 1100hPa (9000 м до -500 м);
- розмір: 21 мм x 18 мм.



Рисунок 2.1 - Барометр BMP280

Технічні характеристики датчика DHT11 [23]. Датчики DHT (рис.2.2) складаються з двох основних частин: ємнісного датчика вологості і термістора. Також в корпусі встановлена проста мікросхема для перетворення аналогового сигналу в цифровий. Зчитувати цифрового сигналу на виході досить просто. Ви можете використовувати будь-який контролер, не обов'язково Arduino.

Є дві версії датчиків DHT. Вони виглядають майже однаково. Основні відмінності в технічних характеристиках:

- дуже дешевий;
 - живлення від 3 до 5В;
 - максимальне споживання струму складає 2,5 мА під час перетворення (коли запитуються дані);
 - призначений для вимірювання рівня вологості в діапазоні від 20% до 80%.
- Точність вимірювань знаходиться в діапазоні 5%;
- вимірює температуру в діапазоні від 0 до 50 градусів з точністю плюс або мінус 2%;
 - частота вимірювання не більше 1 Гц (один вимір в секунду);
 - розмір корпусу: 15,5 мм x 12 мм x 5,5 мм;
 - 4 роз'єми. Відстань між сусідніми - 0,1 ".

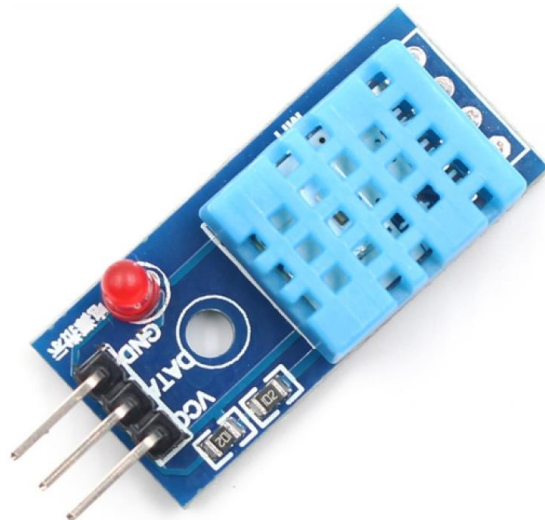


Рисунок 2.2 - Датчик DHT11

Arduino Uno R3 [24] є додатком до всіх функцій попередньої плати Uno тепер використовує ATmega16U2 замість 8U2, знайденого в Uno (або FTDI, знайденого в попередніх поколіннях). Це враховує більш високі швидкості передачі і більше пам'яті. Для Linux або Mac не потрібні драйвери (файл inf для Windows необхідний і включений в Arduino IDE), а також можливість відображати Uno в якості клавіатури, миші, джойстика тощо. Можна не боятися випадкового електричного розряду, оскільки Uno R3 також має пластикову опорну плату для його захисту!

Uno R3 також додає висновки SDA і SCL поруч з AREF. Крім того, поруч з контактом RESET розташовані два нових контакти. Одним з них є IOREF, який дозволяє екранам адаптуватися до напруги, що надходить від плати. Інший не підключений і

зарезервований для майбутніх цілей. Uno R3 працює з усіма існуючими щитами, але може адаптуватися до нових щитів, які використовують ці додаткові контакти (рис 2.3).



Рисунок 2.3 - Arduino Uno

SparkFun Weather Shield [25] - це простий у використанні щит Arduino, який надає вам доступ до атмосферного тиску, відносної вологості, яскравості і температури. На цьому екрані також є підключення до додаткових датчиків, таких як швидкість вітру, напрямок, дощомір і GPS для визначення місця розташування та надточної синхронізації. У Weather Shield використовуються датчик вологості / температури Si7021, датчик атмосферного тиску MPL3115A2 і датчик освітлення ALS-PT19. Екран спирається на бібліотеки Si7021 і MPL3115A2 Arduino. Кожен екран поставляється з двома незайнятими роз'ємами RJ11 (для додаткового підключення датчиків дощу і вітру) та 6-контактним роз'ємом GPS (для додаткового підключення GPS-модуля GP-735). Нарешті, кожен Weather Shield може працювати від 3,3 В до 16 В і має вбудовані регулятори напруги і перетворювачі сигналів (рис 2.4).

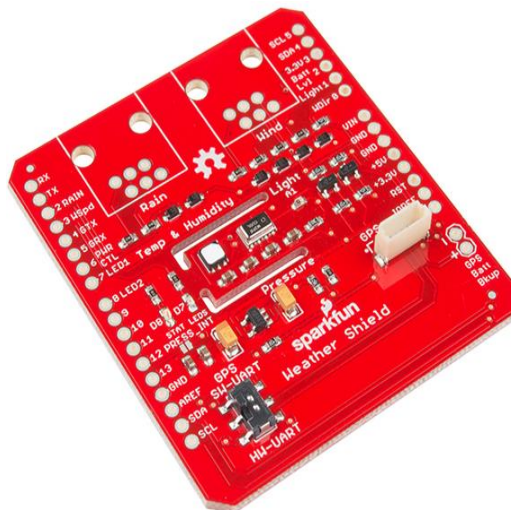


Рисунок 2.4 – Weather Shield

РЧ-модулі Digi XBee 802.15.4 ідеально підходять для додатків, що вимагають низької затримки і передбачуваного часу обміну даними. Продукти Digi XBee 802.15.4, що забезпечують швидкий і надійний зв'язок в двоточковій, тимчасовій, багатоточковій / зіркоподібній конфігураціях, дозволяють легко і надійно підключатися до кінцевої точки. Будь то розгортання в якості простої заміни кабелю для простого послідовного зв'язку або в складі більш складної мережі датчиків з маточиною і спицями, радіочастотні модулі Digi XBee 802.15.4 забезпечують максимальну продуктивність і простоту розробки. Модулі XBee 802.15.4 безперешкодно взаємодіють із сумісними шлюзами, адаптерами пристроїв і розширниками діапазону, надаючи розробникам справжню можливість підключення за межами горизонту [26] (рис 2.5).

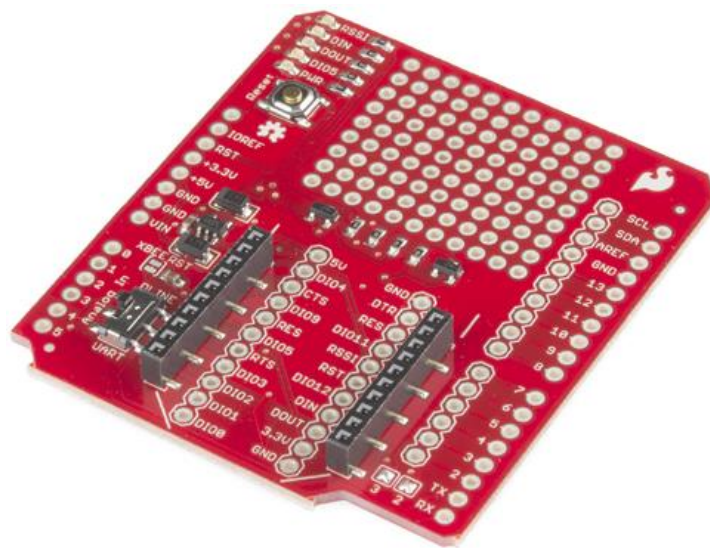


Рисунок 2.5 – XBee Shield

Радіоприймачі XBee [27]- це відмінний спосіб додати бездротові можливості у проект Arduino. Форм-фактор екрану сполучається безпосередньо з будь-якої платою розробника, що має стандартну площу Arduino, і оснащується можливостями бездротового зв'язку з використанням популярного модуля XBee (рис 2.6). Цей пристрій працює з усіма модулями XBee, включаючи Series 1 і 2, стандартну і Pro версії.

Послідовні контакти (DIN і DOUT) XBee підключаються через перемикач DPDT, який дозволяє вибрати підключення до контактів UART (D0, D1) або будь-яким цифровим контактам на Arduino (D2 і D3 за замовчуванням). Живлення від 5-вольта контакту Arduino і регулюється на платі до 3,3 В постійного електричного струму перед подачею на XBee. Екран також забезпечує зміщення рівня на виходах DIN і DOUT XBee. В останній версії доданий перемикач рівня замінений на більш надійний перемикач рівня MOSFET.



Рисунок 2.6 – Модуль XБee з розширеним діапазоном

Arduino Mega [28] - це мікроконтролерна плата на базі АТmega2560. Він має 54 цифрових входи/виходи (з яких 14 можуть використовуватися в якості ШІМ-виходів), 16 аналогових входів, 4 UART (апаратні послідовні порти), кварцовий генератор 16 МГц, USB-з'єднання, роз'єм живлення, роз'єм ICSP, і кнопка скидання . Він містить все необхідне для підтримки мікроконтролера.

Mega 2560 R3 також додає контакти SDA і SCL поруч з AREF. Крім того, поруч з контактом RESET розташовані два нових контакти. Одним з них є IOREF, який дозволяє екранам адаптуватися до напруги, що надходить від плати. Інший не підключений і зарезервований для майбутніх цілей. Mega 2560 R3 працює з усіма існуючими щитами, але може адаптуватися до нових щитів, які використовують ці додаткові контакти (рис 2.7).

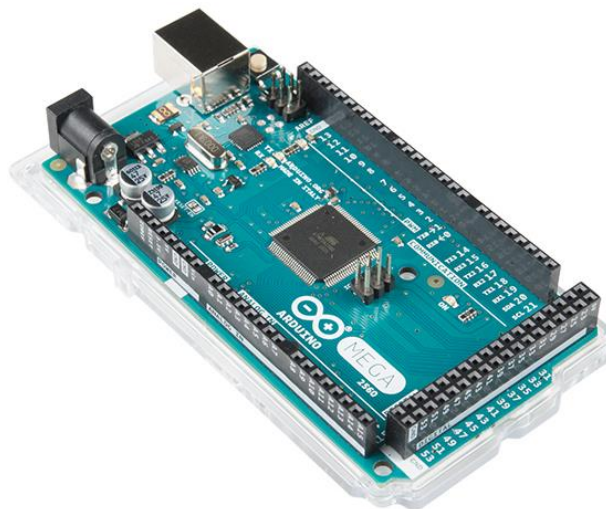


Рисунок 2.7 – Модуль Arduino Mega

Arduino Ethernet Shield 2 [29] дозволяє платі Arduino підключитися до Інтернету. Він заснований на чіпі Wiznet W5500 Ethernet. Wiznet W5500 надає мережевий стек (IP), що підтримує як TCP, так і UDP. Він підтримує до восьми одночасних з'єднань сокетов. Ethernet Shield 2 підключається до плати Arduino за допомогою довгих провідних роз'ємів, що проходять через екран. Це зберігає розташування шпильок в цілості й дозволяє накладати поверх нього ще один щит.

Є вбудований слот для карт пам'яті micro-SD, який можна використовувати для зберігання файлів для обслуговування по мережі. Він сумісний з Arduino Uno і Mega (з використанням бібліотеки Ethernet). Вбудований кардрідер micro-SD доступний через SD Library. При роботі з цією бібліотекою SS знаходиться на контакті 4. Первісна версія Shield містила повнорозмірний слот для SD-карти. Екран також включає в себе контролер скидання, щоб забезпечити правильне перезавантаження модуля Ethernet W5500 при включенні харчування. Попередні версії Shield були сумісні з Mega і повинні були бути скинуті вручну після включення живлення. Останній перегляд плати виставляє 1.0 терморегулятори на версії 3 Ради Arduino UNO. Ethernet Shield 2 має стандартне з'єднання RJ-45 з вбудованим лінійним трансформатором і підтримкою Power over Ethernet. Модуль PoE, який вам знадобиться, є окремим компонентом, який необхідно додати. Цей екран також має кілька вхідних / вихідних і комунікаційних інтерфейсів TinkerKit, які можна знайти на верхній стороні плати (рис 2.8).

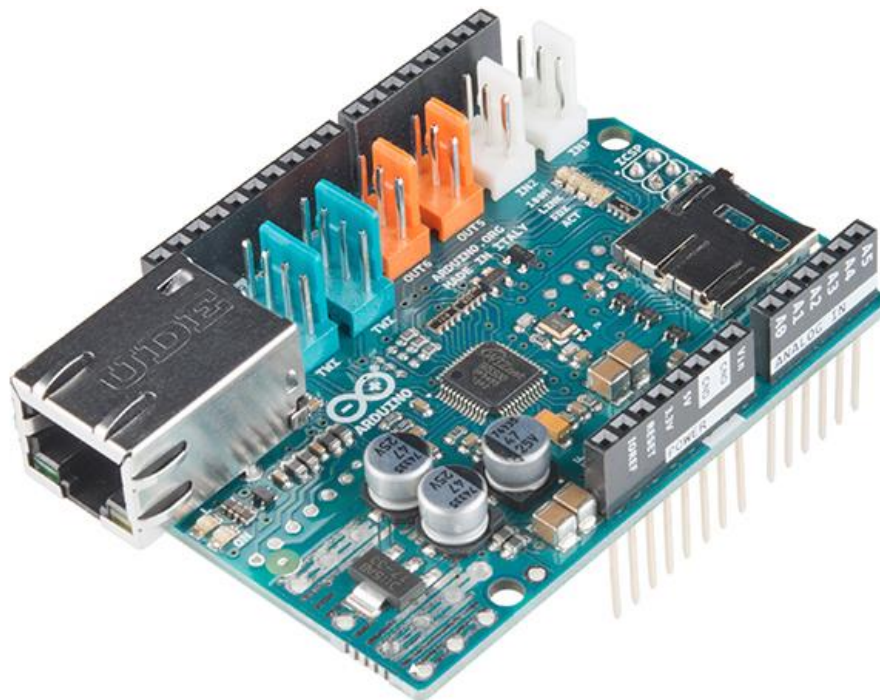


Рисунок 2.8 – Модуль Arduino Ethernet Shield

Комплекти монтажу [30] призначені для роботи з Arduino Uno R3, Leonardo і новими платами Arduino в майбутньому. Вони ідеально підходять для очищення роз'єму USB-B і

відмінно підходять для укладання декількох екранів. Цей комплект включає 4 роз'єми (2 8-контактних, 1 10-контактний і 1 6-контактний), яких достатньо для підключення екрану до плати Arduino (рис 2.9).

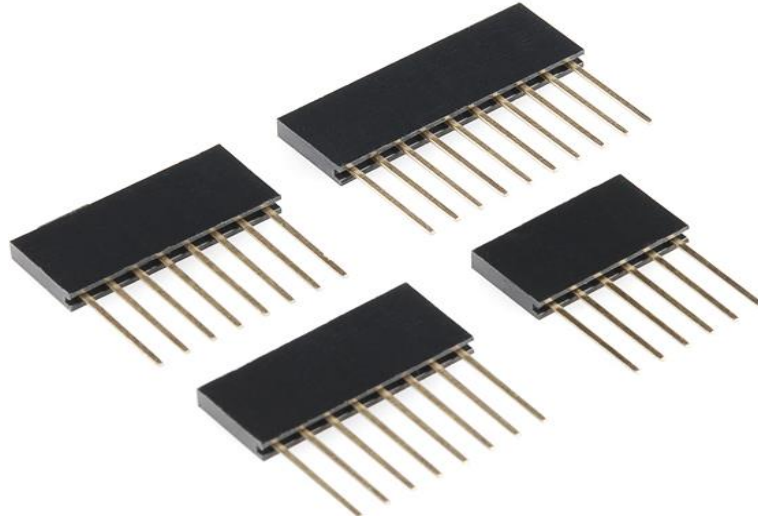


Рисунок 2.9 – Комплекти для монтажу Arduino

Це апаратний ключ SparkFun XBee Explorer [31] для лінійки модулів Digi XBee. За допомогою ключа XBee Explorer можна підключити пристрій безпосередньо до USB-порту і використовувати його в якості шлюзу між комп'ютером і XBee. Цей пристрій працює з усіма модулями XBee, включаючи Series 1 і Series 2.5, стандартну і Pro версію. Вбудований регулятор напруги працює до 500 мА (рис 2.10).



Рисунок 2.10 – Модуль XBee Explorer Dongle

Родзинкою цієї плати є конвертер FT231X USB-to-Serial. Ось що переводить дані між вашим комп'ютером і XBee. Також є кнопка скидання і регулятор напруги, щоб забезпечити

XBee достатньою кількістю енергії. Крім того, є чотири світлодіоди, які допоможуть, якщо знадобиться налагодити XBee: RX, TX, RSSI (індикатор рівня сигналу) і індикатор живлення. Ця плата також розбиває кожен з висновків введення-виведення XBee на пару сумісних з макетом заголовків.

Також потрібно 2x 5В трансформатора [32] для живлення Arduino з USB-перезодом на кабель з підключеним джерелом живлення центрального блоку буде від мережі 220 (рис 2.11). Ще є окремий вивід на 5В.. Більшість плат вимагає наявності живлення в діапазоні від 4.5 до 9 вольт через роз'єм живлення і 4.5-5 вольт через USB.

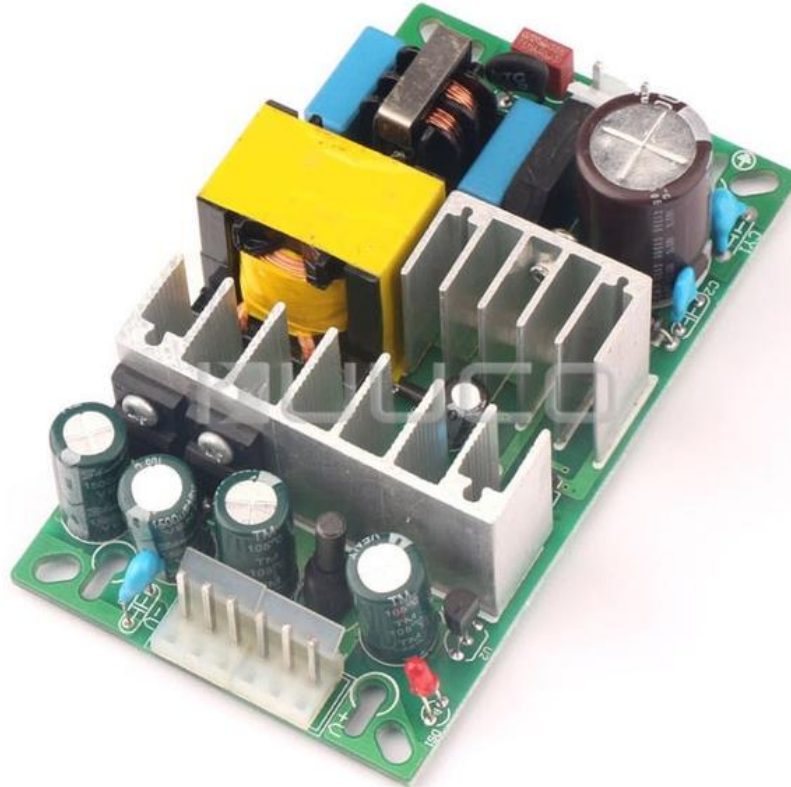


Рисунок 2.11 – Power Supply Module AC 110-220v

2.3 PostgreSQL

Сама метеостанція не може обійтися без бази даних. В цьому випадку була обрана PostgreSQL.

PostgreSQL [33] — є об'єктно-реляційна система керування базами даних (СКБД). Ця система керування є чудовою альтернативою комерційним СКБД (Oracle Database, Microsoft SQL Server та IBM DB2), так і для СКБД з відкритим кодом (MySQL, Firebird, SQLite).

Існує великий список типів даних, які підтримує PostgreSQL, крім числових, з плаваючою точкою, текстових, булевих і інших очікуваних типів даних (а також безлічі їх варіацій).

PostgreSQL може бути розширений користувачем для власних потреб практично в будь-якому аспекті. Інші СКБД такі як MySQL, MariaDB і Firebird теж мають деякі з цих типів даних, але тільки PostgreSQL підтримує їх всі. Користувач в PostgreSQL може самостійно створювати нові, необхідні йому типи даних та програмувати для них механізми індексування за допомогою додатку GiST.

PostgreSQL співпрацює з різними мовами програмування, такими як: C, C++, Java, Python, Clojure. Все це виконується за допомогою модуля GiST для написання збережених процедур, тригерів і функцій PL/Java.

У PostgreSQL функції виконуються із привілеями користувача, який їх викликав, або із привілеями користувача, який їх написав. В даній СКБД є підтримка індексів наступних типів: B-дерево, хеш, R-дерево, GIN. При необхідності можна створити нові типи індексів.

В базі даних створено три таблиці 2-4 для таких значень:

- temperature
- humidity
- brightness.

Таблиця 2.1 - Температура

Назва поля	Тип даних	Опис
TITLE_Temperature	VARCHAR	Надає назву вікну вихідних даних.
PUBLISH_DATE	TITESTAMP	Вказує час та дату створення вікна графіка.

Таблиця 2.2 - Вологість

Назва поля	Тип даних	Опис
TITLE_Humidity	VARCHAR	Надає назву вікну вихідних даних.
PUBLISH_DATE	TITESTAMP	Вказує час та дату створення вікна графіка.

Таблиця 2.3 - Яскравість

Назва поля	Тип даних	Опис
TITLE_Brightness	VARCHAR	Надає назву вікну вихідних даних.
PUBLISH_DATE	TITESTAMP	Вказує час та дату створення вікна графіка.

2.4 Візуалізація

Візуалізація інформації - це інтерактивне дослідження візуального представлення абстрактних даних для підвищення рівня знань людини. Абстрактні дані включають як числові, так і нечислові дані, такі як текстова і географічна інформація. Однак візуалізація інформації відрізняється від наукової візуалізації: «існує інфовіс [34], коли вибрано просторове уявлення, і це scivis [35], коли дано просторове уявлення» [36, 37].

Дані самі по собі – як послідовність бітів та байтів, що зберігається у файлі на жорсткому диску комп'ютера – є невидимими. Щоб побачити їх та зрозуміти їхнє значення, потрібно візуалізувати ці дані. У цій главі я маю намір використовувати ширше визначення терміну «візуалізація», яке включає навіть просте текстове представлення даних. Наприклад, елементарне завантаження бази даних у програму електронних таблиць вже можна розглядати як візуалізацію даних. Невидимі дані раптом перетворюються на видиму «картину» на екрані нашого комп'ютера. Отже, запитання не в тому, чи слід візуалізувати дані, а в тому, який вид візуалізації є найбільш придатним у тій чи іншій ситуації

Інакше кажучи, мова йде про те, коли є сенс виходити за межі табличної візуалізації? Коротка відповідь така – майже завжди. Одних лише таблиць однозначно замало для того, щоб дати нам загальне представлення про базу даних. І таблиці самі по собі не дозволяють нам відразу ж визначити певні закономірності в наборі даних. Найбільш типовий приклад – це географічні закономірності, які можна спостерегти тільки після того, як дані будуть візуалізовані за допомогою карти. Але є й інші типи закономірностей, про які йтиметься далі в цій главі.

Візуалізація надає унікальний метод бачення бази даних. Для візуалізації даних існують найрізноманітніші способи.

Таблиці є дуже потужними, коли ви маєте справу з порівняно малою кількістю позицій даних. Вони показують категорії та кількісні дані в найбільш структурованій та організованій формі, і розкривають свій повний потенціал у поєднанні із можливістю сортувати та відфільтровувати дані. На додачу до цього Едвард Тафт рекомендує включати невеликі діаграми у колонки з даними, наприклад, один стовпчик на рядок, або маленький лінійний графік (відомий також як спарклайн, *sparkline*). Але як вже було сказано, таблиці, без сумніву, мають свої обмеження. Вони прекрасно показують вам одновимірні дані (наприклад, перша десятка чогось), але погано пристосовані для порівняння різно вимірних даних (наприклад, зміна чисельності населення в країні протягом часу).

Графіки та діаграми загалом дають вам змогу вказувати різні виміри у ваших даних у формі візуальних пропорцій геометричних форм. Про ефективність тих чи інших візуальних властивостей написано дуже багато, і коротко це можна сформулювати так: колір – це складно, позиція – це все. Лінійні графіки особливо зручні для показу змін за певний період часу, тоді як стовпчикові діаграми ідеальні для порівняння категоріальних даних. Елементи діаграм можна розміщувати поверх один одного. Якщо ви хочете порівняти малий набір груп у ваших даних, є потужний спосіб це зробити, показавши кілька екземплярів подібних графіків (цей тип графіків має назву *small multiples*). В усіх графіках чи діаграмах ви можете використовувати різні види масштабування, щоб відобразити різні аспекти ваших даних (тобто, лінійний чи логарифмічний масштаб).

Фактично, більшість із даних, з якими ми маємо справу, у той чи інший спосіб пов'язана з реальними людьми. Сила карт у тому, що вони прив'язують дані до нашого власного фізичного світу. Уявімо базу даних злочинів, яка містить географічну прив'язку. Для вас критично важливо бачити, де відбулися ці злочини. Також карти можуть висвітлити географічні закономірності в даних, наприклад, зміну тренду від Півночі до Півдня, або від міських до сільських регіонів.

2.5 InterSystems DeepSee

InterSystems DeepSee[38] - це вбудована аналітична технологія і набір інструментів для створення систем підтримки прийняття ефективних рішень, в тому числі, і з застосуванням прогнозних моделей. DeepSee працює зі структурованими і неструктурованими даними.

InterSystems DeepSee надає розробникам впроваджувати в свої додатки аналітичну OLAP-функціональність, яка здатна працювати на оперативних базах даних додатків без створення окремої інфраструктури для вирішення аналітичних задач.

У DeepSee змінено архітектуру системи. Замість монолітного рішення, що включає в себе весь функціонал - від виконання запитів до призначеного для користувача інтерфейсу, нова архітектура поділяється на кілька рівнів, до кожного з яких є доступ у розробника прикладних систем.

2.6 Медіанна фільтрація

Достатньо часто застосовується метод попередньої обробки сигналів [39]. Специфічною особливістю медіанних фільтрів є вибірковість по відношенню до елементів масиву, які представляють собою немонотонну складову послідовність чисел в межах вікна (апертури) фільтра, і різко виділяються на тлі сусідніх відліків. У той же час на монотонну складову послідовність медіанний фільтр не діє, залишаючи її без змін. Завдяки цій особливості, медіанні фільтри при оптимально обраній апертурі можуть, наприклад, зберігати без спотворень різкі межі об'єктів, ефективно пригнічуючи некорельовані або слабо корельовані перешкоди і малорозмірні деталі. Ця властивість дозволяє застосовувати медіанну фільтрацію для усунення аномальних значень в масивах даних, зменшення викидів і імпульсних перешкод. Характерною особливістю медіанного фільтра є його нелінійність. У багатьох випадках застосування медіанного фільтра виявляється більш ефективним в порівнянні з лінійними фільтрами, оскільки процедури лінійної обробки є оптимальними при рівномірному або гауссовому розподілі перешкод, що в реальних сигналах може бути далеко не так. У випадках, коли перепади значень сигналів великі в порівнянні з дисперсією аддитивного білого шуму, медіанний фільтр дає менше значення середньоквадратичної помилки в порівнянні з оптимальними лінійними фільтрами. Особливо ефективним медіанний фільтр виявляється при очищенні сигналів від імпульсних шумів при обробці зображень, акустичних сигналів, передачі кодових сигналів тощо. Однак детальні

дослідження властивостей медіанних фільтрів як засобу фільтрації сигналів різного типу є досить рідкісними.

Медіани давно використовувалися і вивчалися в статистиці як альтернатива середнім арифметичним значенням відліків в оцінці вибіркового середнього значення. Медіаною числової послідовності x_1, x_2, \dots, x_n при непарному n є середній за значенням член ряду, що виходить при упорядкуванні цієї послідовності за зростанням (або зменшенням). Для парних n медіану зазвичай визначають як середнє арифметичне двох середніх відліків впорядкованої послідовності.

Медіанний фільтр являє собою віконний фільтр, що послідовно ковзає по масиву сигналу і повертає на кожному кроці один з елементів, що потрапили у вікно (апертуру) фільтра. Вихідний сигнал y_k змінного медіанного фільтра шириною $2n+1$ для поточного відліку k формується з вхідного часового ряду $\dots, x_{k-n}, x_k, x_{k+1}, \dots$ відповідно до формули 1:

$$y_k = \text{med}(x_{k-n}, x_{k-n+1}, \dots, x_{k-1}, x_k, x_{k+1}, \dots, x_{k+n-1}, x_{k+n}) \quad (1)$$

$$\text{де } \text{med}(x_1, \dots, x_m, \dots, x_{2n+1}) = x_{n+1};$$

x_m – елементи варіаційного ряду, тобто ранжирування в порядку зростання значень формули 2.

$$x_1 = \min(x_1, x_2, \dots, x_{2n+1}) \leq x(2) \leq x(3) \leq \dots \leq x_{2n+1} = \max(x_1, x_2, \dots, x_{2n+1}).$$

Таким чином, медіанна фільтрація здійснює заміну значень відліків в центрі апертури медіанного значення вихідних відліків усередині апертури фільтра. На практиці апертура фільтра для спрощення алгоритмів обробки даних, як правило, встановлюється з непарним числом відліків, що і буде прийматися при розгляді в подальшому без додаткових пояснень.

Переваги медіанних фільтрів:

- проста структура фільтра, як для апаратної, так і для програмної реалізації;
- фільтр не змінює ступінчасті і пилковидні функції;
- фільтр добре пригнічує поодинокі імпульсні перешкоди і випадкові шумові викиди відліків.

Недоліки медіанних фільтрів:

- медіанна фільтрація нелінійна, оскільки медіана суми двох довільних послідовностей не дорівнює сумі їх медіан, що в ряді випадків може ускладнювати математичний аналіз сигналів;
- фільтр викликає сплюснення вершин трикутних функцій;
- придушення білого і гаусового шуму менш ефективно, ніж у лінійних фільтрів.

Слабка ефективність спостерігається також при фільтрації флюктуаційного шуму;

- при збільшенні розмірів вікна фільтра відбувається розмиття крутих змін сигналу і стрибків.

Недоліки методу можна зменшити, якщо застосовувати медіанну фільтрацію з адаптивною зміною розміру вікна фільтра в залежності від динаміки сигналу і характеру шумів (адаптивна медіанна фільтрація). Як критерій розміру вікна можна використовувати, наприклад, величину відхилення значень сусідніх відліків щодо центрального ранжированного відліку.

2.7 Висновки по розділу 2

В даному розділі розглянуто апаратно-програмний комплекс, базу даних, технологічний набір для візуалізації даних погоди та спосіб фільтрації некоректних даних для побудови метеостанції. Все це дає змогу підбирати різноманітні компоненти що зможуть задовольнити користувача. InterSystems DeepSee надає розробникам впроваджувати в свої додатки аналітичної OLAP-функціональності. Сама платформа Arduino дає змогу підбирати різні компоненти, а велика кількість датчиків для вимірювання погодних факторів дасть змогу розширити та покращити функціонал станцій, що є великим плюсом для сільгосп підприємств та фермерів в тому, що кліматична зона та положення сільгоспугідь різні.

РОЗДІЛ 3

РОЗРОБКА МЕТЕОСТАНЦІЇ

3.1 Налаштування метеостанції

Програмний комплекс створений за допомогою мови програмування C++.

Зберемо на Arduino схему, яка читає дані з COM порту і включає світлодіод на вказану кількість мілісекунд (рис. 3.1).

```

/* Led.ino
 *
 * Пример получения данных по COM порту
 * Подключите светодиод к выводу ledPin
 *
 */

// Вывод светодиода (цифровой)
#define ledpin 8

// "Буфер" поступающих данных
String inString = "";

void setup() {
  Serial.begin(9600);
  pinMode(ledpin, OUTPUT);
  digitalWrite(ledpin, LOW);
}

void loop() {
  // Получаем данные из COM порта
  while (Serial.available() > 0) {
    int inChar = Serial.read();
    if (isdigit(inChar)) {
      // Получаем 1 символ,
      // Прибавляем его к строке
      inString += (char)inChar;
    }
  }
}

```

Рисунок 3.1 – Код COM-порту

У Cachelé напишемо метод, що підключається до COM порту і відправляє рядок 1000 \n (рис. 3.2):

```

/// Отправляем на порт строку 1000\n
ClassMethod SendSerial()
{
  set port = "COM1"
  open port:::" 0801n0":/BAUD=9600) // Открываем устройство
  set old = $IO // Записываем текущее устройство ввода-вывода
  use port // Переключаемся на com порт
  write $Char(10) // Отправка пробного пакета данных
  hang 1
  write 1000 _ $Char(10) // Передаём строку 1000\n
  use old // Переключаем вывод на терминал
  close port // Закрываем устройство
}

```

Рисунок 3.2 – Підключення рядка до COM-порту

По-перше, було вирішено, де розмістити метеостанцію. Це було зроблено на території присадибної ділянки під дахом навісу. Таке місце було обрано, оскільки воно було під впливом погодних явищ, але мало дах для монтажу електроніки та їх захисту від прямого сонячного світла та дощу, які можуть збільшити в майбутньому похибку вимірювань та виводу апаратури зі строю.. Через те, що в кінцевому рахунку хотіли передавати інформацію сторонньому агрегатору даних, було враховувано це при виборі місця і заліза.

Обране розташування було поза радіусом дії Wi-Fi станції сервера, в нашому випадку його роль виконував звичайний ноутбук. Щоб зробити це, було під'єднано Arduino Uno до передавача високої потужності XBee. Потім дані передаються до приймаючого модуля XBee на Arduino Mega вже всередині будівлі на ноутбук сервер (рис.3.3). Який був підключений до Інтернету та дані відправляються в безкоштовний сервіс агрегації даних, ThingSpeak.com.

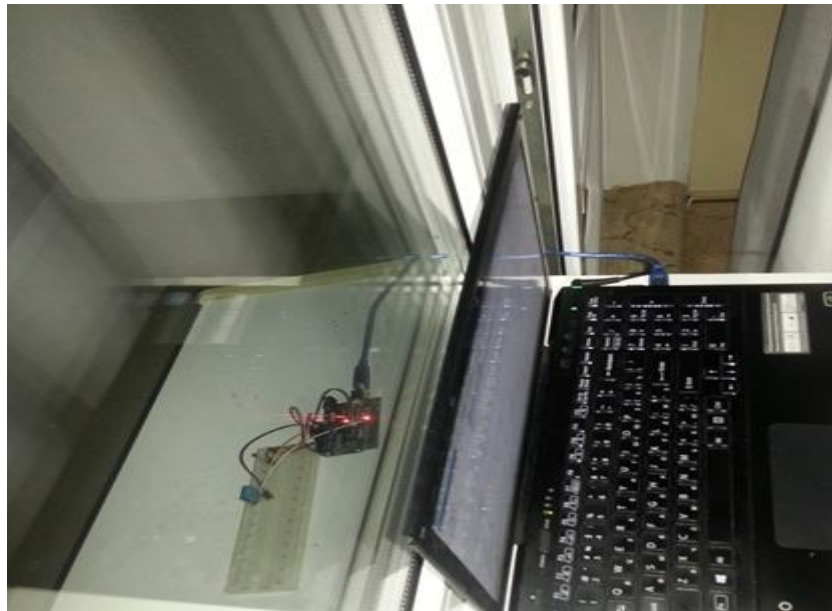


Рисунок 3.3 – Комп'ютер-сервер та приймач даних с метеостанції

Зовнішнє Arduino встановлено під навісом на вулиці і відповідає за збір вимірювань і передачу даних на Arduino в приміщенні. Для реалізації такого обміну, спочатку було з'єднано контакти від Weather shield і XBee shield. Потім з'єднали XBee shield з Arduino Uno відповідно до документації, наданої з Shield. У XBee shield є XBee трансивер високої потужності. Ми використовували програмне забезпечення X-CTU для програмування потрібної адреси одержувача і завантаження потрібної прошивки в XBee shield., датчики температури, вологи та світла були підключені до weather shield за допомогою роз'ємів RJ-45, що входять в комплект поставки.

Для складання метеостанції використовували фоторезистор і датчик DHT11 зображений на рисунку 3.4 (температура і вологість).

```

/* Meteo.ino
 *
 * Программа, регистрирующая влажность, температуру и яркость
 * Отправляет результаты на COM port
 * Формат вывода: H=1.0;T=1.0;LL=1;
 */

//Пин фоторезистора (аналоговый)
int lightPin = 0;

// Пин DHT-11 (цифровой)
int DHpin = 8;

// Массив, хранящий данные DHT-11
byte dat[5];

```

Рисунок 3.4 – Підключення датчику DHT11

Після завантаження коду на Arduino вона починає посилати дані на COM порт в наступному форматі та схем підключень на рисунку 3.5-3.7:

H = 34.0; T = 24.0; LL = 605;

де:

- H - вологість (від 0 до 100 відсотків);
- T - температура в градусах Цельсія;
- LL - освітленість (від 0 до 1023).

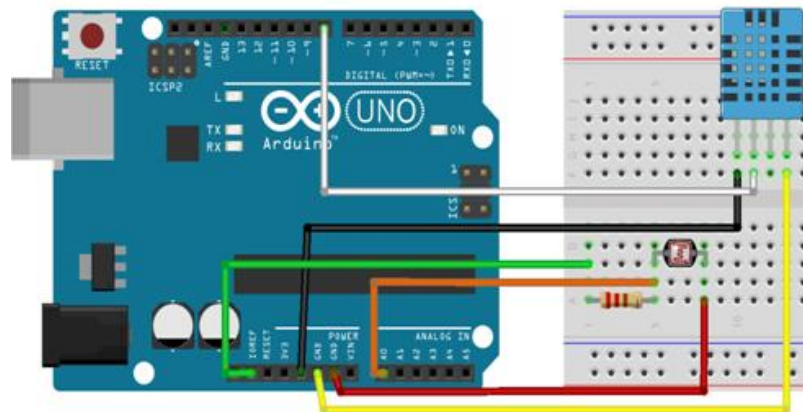


Рисунок 3.5 - Схема підключень

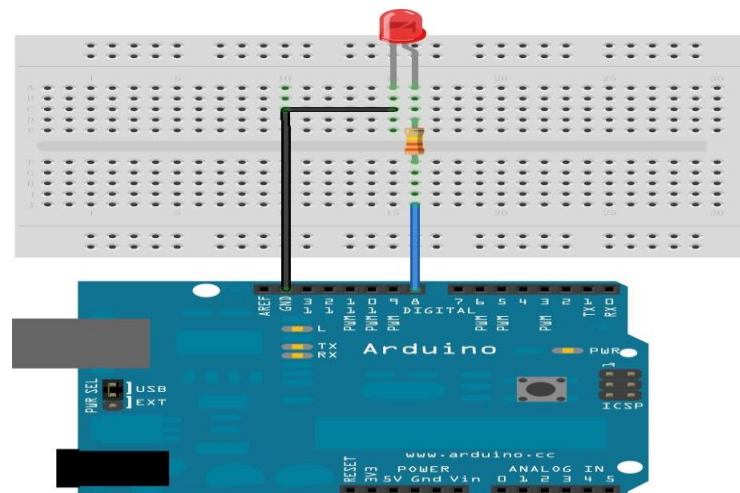


Рисунок 3.6 - Схема підключень

В Cache зберігатимемо дані за допомогою класу, який був написаний при використанні Arduino.Info (рис3.8).

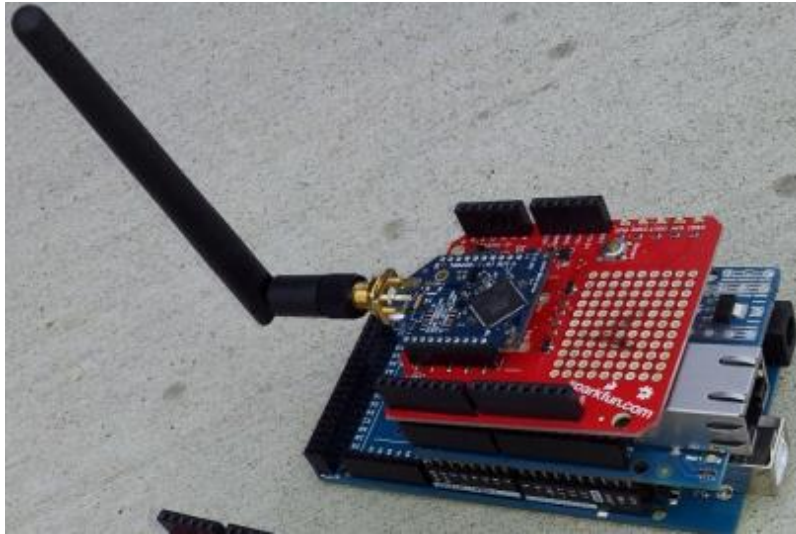


Рисунок 3.7 - Зібрана компоновка

```

Class Arduino.Info Extends %Persistent
{
    Parameter SerialPort As %String = "com1";

    Property DateTime As %DateTime;

    Property Temperature As %Double;

    Property Humidity As %Double(MAXVAL = 100, MINVAL = 0);

    Property Brightness As %Double(MAXVAL = 100, MINVAL = 0);

    Property Volume As %Double(MAXVAL = 100, MINVAL = 0);

    ClassMethod AddNew(Temperature = 0, Humidity = 0, Brightness = 0, Volume = 0)
    {

```

Рисунок 3.8 - Клас Arduino.Info

І додамо туди метод, який буде приймати дані в форматі Arduino, і перетворювати їх об'єкти класу Arduino.Info (рис.3.9).

```

/// Получаем поток данных в формате H=34.0;T=24.0;LL=605;\n
/// И преобразуем их в объекты класса Arduino.Info
ClassMethod ReceiveSerial(port = {..#SerialPort})
{
    try {
        open port:(:::" 0801n0"/BAUD=9600)
        set old = %IO
        use port
        for {
            read x //Читаем одну строку
            set Humidity = %Piece(%Piece(x,";",1),"=",2)
            set Temperature = %Piece(%Piece(x,";",2),"=",2)
            set Brightness = %Piece(%Piece(x,";",3),"=",2)
            if (x '=' "") {
                do ..AddNew(Temperature, Humidity, Brightness) // Добавля
ем данные

```

Рисунок 3.9 – Метод перетворення об'єктів

Внутрішнє Arduino знаходиться всередині нашого будинку, і відповідає за отримання даних від зовнішнього Arduino, за перевірку достовірності даних, і їх відправку на ThingSpeak. Щоб побудувати таку систему, спочатку припаяли контакти на Ethernet shield і другий XBee shield. Потім вставили Ethernet shield і XBee shield в Mega Arduino відповідно до документації. Знову використовували X-CTU для програмування необхідної адреси одержувача і завантаження потрібної прошивки на XBee shield.

Далі, запрограмували Arduino отримати XBee повідомлення і пересилати пакети на ThingSpeak зі швидкістю один раз на хвилину. Перед відправкою повідомлення ThingSpeak, ми налаштували обліковий запис і сконфігурували інформацію про канал і розташування.

Після цього нам потрібно запустити Arduino і виконати в терміналі метод ReceiveSerial (рис.3.10).

```
write ##class(Arduino.Info).ReceiveSerial()
```

Рисунок 3.10 – Метод ReceiveSerial

Отримавши дані про погоду з ThingSpeak ми використовуємо команду thingSpeakFetch для перегляду доступних полів даних, одночасного імпорту всіх полів і зберігання даних про температуру, вологість в своїх змінних. Документація про підтримку ThingSpeak (рис.3.11-3.13).

```
[d,t,ci] = thingSpeakFetch(12397,'NumPoints',8000); % fetch last 8000 minutes of data
```

Рисунок 3.11 - ThingSpeakFetch

8000 точок - це максимальна кількість точок, яке ThingSpeak дозволяє запросити за раз. Для нашої частоти вимірювань, це відповідає приблизно 6 дням вимірювань.

```
tempF = d(:,4); % field 4 is temperature in deg F
baro = d(:,6); % pressure in inches Hg
humidity = d(:,3); % field 3 is relative humidity in percent
windDir = d(:,1);
windSpeed = d(:,2);
tempC = (5/9)*(tempF-32); % convert to Celsius
availableFields = ci.FieldDescriptions'
```

Рисунок 3.12 – ThingSpeakFetch

```

availableFields =

'Wind Direction (North = 0 degrees)'
'Wind Speed (mph)'
'% Humidity'
'Temperature (F)'
'Rain (Inches/minute)'
'Pressure ("Hg)'
'Power Level (V)'
'Light Intensity'

```

Рисунок 3.13 - ThingSpeakFetch

Щоб отримати краще розуміння наших даних, ми знаходимо мінімальне, максимальне і середнє значення для даних, які імпортували і знаходимо час, для максимальних і мінімальних значень. Це дає швидкий шлях до валідації даних з метеостанції (рис.3.14).

```

figure(2)
[ax, h1, h2] = plotyy(t,[tempF tdpf],t,humidity);
set(ax(2), 'XTick', [])
set(ax(2), 'YColor', 'k')
set(ax(1), 'YTick', [0,20,40,60,80,100])
set(ax(2), 'YTick', [0,20,40,60,80,100])
datetick(ax(1), 'x', 'keeplimits', 'keepticks')
set(get(ax(2), 'YLabel'), 'String', availableFields(3))
set(get(ax(1), 'YLabel'), 'String', availableFields(4))
grid on
legend('Location', 'South', 'Temperature', 'Dew point', 'Humidity')

```

Рисунок 3.14 - Валідації даних з нашої метеостанції

summary =

availableFields	maxData	times_max
'Wind Direction (North = 0 degrees)'	338	'10-Jul-2019 05:01:32'
'% Humidity'	86.5	'15-Jul-2019 04:51:24'
'Temperature (F)'	96.7	'12-Jul-2019 16:28:55'
'Rain (Inches/minute)'	0.04	'15-Jul-2019 13:47:13'
'Pressure ("Hg)'	30.23	'11-Jul-2019 09:25:07'
'Power Level (V)'	4.44	'10-Jul-2019 10:25:01'
'Light Intensity'	0.06	'12-Jul-2019 13:23:38'
meanData	minData	times_min

NaN	0	'10-Jul-2019 04:54:32'
3.0272	0	'10-Jul-2019 01:33:14'
57.386	25.9	'12-Jul-2019 13:39:39'
80.339	69.6	'11-Jul-2019 06:59:54'
5.625e-05	0	'10-Jul-2019 01:02:11'
30.04	29.78	'15-Jul-2019 13:04:08'
4.4149	4.38	'11-Jul-2019 09:22:06'

Якщо отримуємо несподівані значення, такі як максимальне атмосферний тиск 40 атмосфер або максимальну температуру 1700 градусів, можемо припустити, що дані є некоректними.

Тепер необхідно візуалізувати дані і спостерігати їх поведінку протягом останніх 5 або 6 днів (рис. 3.15 – 3.16).

```
figure(2)
[ax, h1, h2] = plotyy(t, [tempF tdpf], t, humidity);
set(ax(2), 'XTick', [])
set(ax(2), 'YColor', 'k')
set(ax(1), 'YTick', [0, 20, 40, 60, 80, 100])
set(ax(2), 'YTick', [0, 20, 40, 60, 80, 100])
datetick(ax(1), 'x', 'keeplimits', 'keepticks')
set(get(ax(2), 'YLabel'), 'String', availableFields(3))
set(get(ax(1), 'YLabel'), 'String', availableFields(4))
grid on
legend('Location', 'South', 'Temperature', 'Dew point', 'Humidity')
```

Рисунок 3.15 – Візуалізація даних погоди 5-6 днів

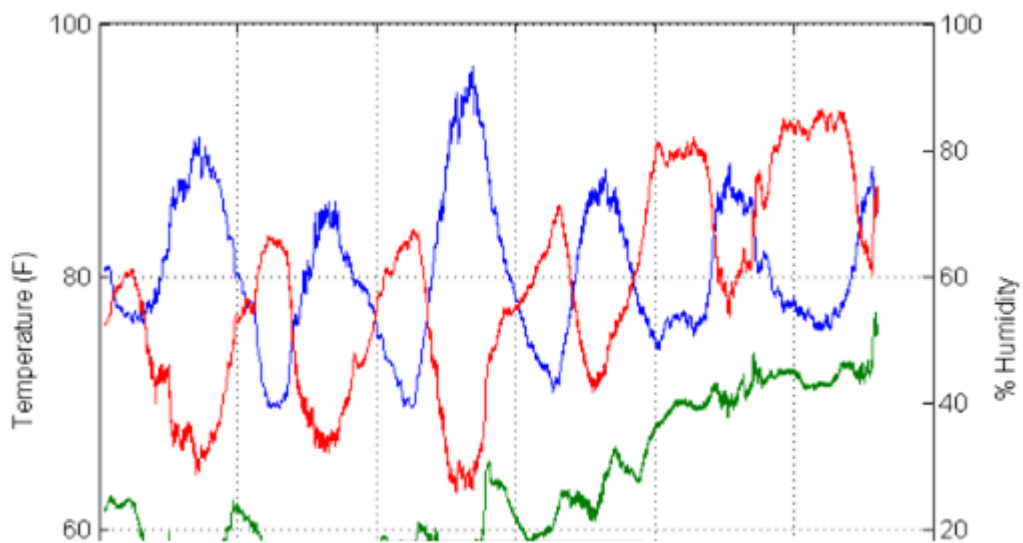


Рисунок 3.16 – Графік візуалізації даних погоди 5-6 днів

Останнє питання, на яке хотіли відповісти - чи дійсно атмосферний тиск падає перед дощем? Щоб зробити це, отримали дані з метеостанції за відомий дощовий день. На цей раз, зацікавлені в барометричному тиску і кількості опадів. Датчик обертається і спорожняється, коли випало 0,01 дюймів опадів. Код Arduino вважає кількість випорожнень за кожну хвилину і передає відповідне значення опадів в ThingSpeak (рис.3.17).

```
[d,t,ci] = thingSpeakFetch(12397,'DateRange',{'6/4/2014','6/6/2014'}); % get data
baro = d(:,6); % pressure
extraData = rem(length(baro),60); % computes excess points beyond the hour
baro(1:extraData) = []; % removes excess points so we have even number of hours
rain = d(:,5); % rainfall from sensor in inches per minute
```

Рисунок 3.17 – Значення опадів в ThingSpeak

Якщо підсумувати всі опади за 5 червня, бачимо, що отримали 0,48 дюймів опадів, що становить 13% від середньомісячних 3,68 дюймів, що показує, що день був дійсно дуже дощовим. Щоб отримати більш повне уявлення про те, коли випав максимум опадів, перетворили дані в часові дані, як показано нижче.

```
rain(1:extraData) = [];
t(1:extraData) = [];
rainHourly = sum(reshape(rain,60,[]))'; % convert to hourly summed samples
maxRainPerMinute = max(rain)
june5rainfall = sum(rainHourly(25:end)) % 24 hours of measurements from June 5
baroHourly = downsample(baro,60); % hourly samples
timestamps = downsample(t,60); % hourly samples
```

Іноді виникають проблеми з наявними даними. Наприклад, 17 липня, записали кілька недостовірних даних про температуру. Давайте подивимося, як використовувати код для їх фільтрації (рис.3.18).

```
[d,t] = thingSpeakFetch(12397,'DateRange',{'5/30/2014','5/31/2014'});
rawTemperatureData = d(:,4);
newTemperatureData = rawTemperatureData;
minTemp = min(rawTemperatureData) % wow that is cold!
```

Рисунок 3.18 – Код фільтрації

Використання порогового фільтра для видалення викиду даних елементів, які не пройшли пороговий тест. У цьому випадку є деякі значення, які свідомо недостовірні, такі як значення температури -1766 градусів за Фаренгейтом. Тому можна використовувати дані, які включають тільки значення температури, між 0 і 120, які є прийнятними величинами для літнього сезону (рис.3.19-3.22).

```
tnew = t';
outlierIndices = [find(rawTemperatureData < 0); find(rawTemperatureData > 120)];
tnew(outlierIndices) = [];
newTemperatureData(outlierIndices) = [];
```

Рисунок 3.19 – Використання порогового фільтру

Побудуємо графіки очищених і вихідних даних.

```
figure(4)
subplot(2,1,2)
plot(tnew,newTemperatureData,'-o')
datetick
xlabel('Time of Day')
ylabel(availableFields(4))
title('Filtered Data - outliers deleted')
grid on
subplot(2,1,1)
plot(t,rawTemperatureData,'-o')
datetick
xlabel('Time of Day')
ylabel(availableFields(4))
title('Original Data')
grid on
```

Рисунок 3.20 – Використання порогового фільтру

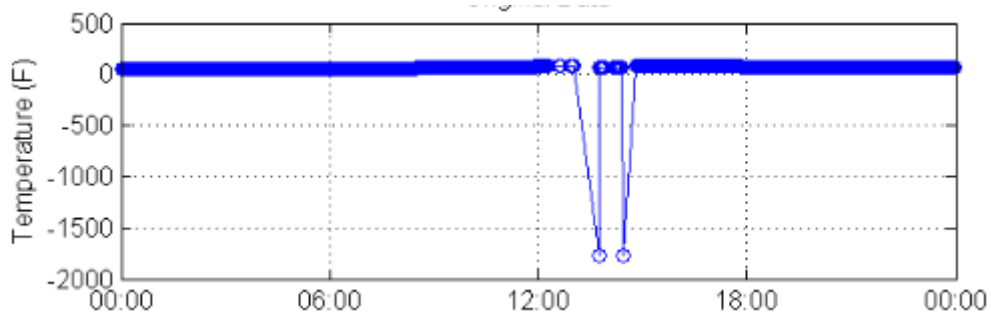


Рисунок 3.21 – Графіки неочищених і вихідних даних

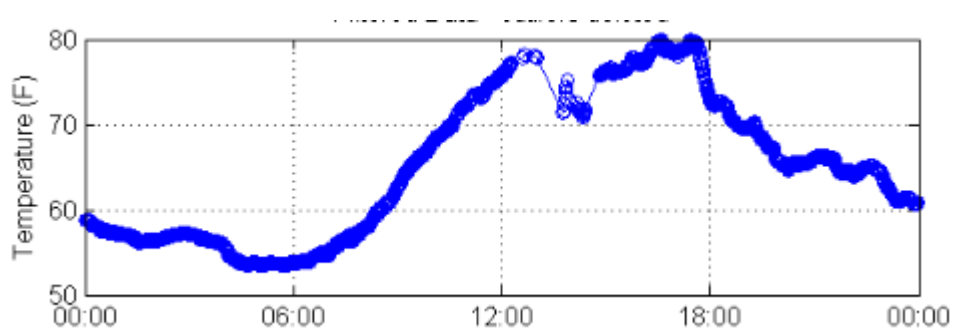


Рисунок 3.22 – Графіки очищених і вихідних даних

Ще один спосіб видалення поганих даних - застосувати медіанний фільтр (рис.3.23). Медіанний фільтр не вимагає стільки знань про набір даних. Він буде просто видаляти значення, які виявляються поза середнього найближчих сусідів. Результати фільтрації - вектор тієї ж довжини що і вихідний, на відміну від видалення точок даних, що призводить

до розривів в даних і укорочення запису. Цей тип фільтра також може бути використаний для видалення шуму з сигналу.

```
n = 5; % this value determines the number of total points used in the filter
```

Рисунок 3.23 – Команда медіанного фільтру

Великі значення n позначають кількість «сусідів» для порівняння. При температурах, зібраних раз в хвилину, обираємо $n = 5$, тому що температура не повинна, як правило, сильно змінюватися протягом 5 хвилин (рис.3.24-3.25).

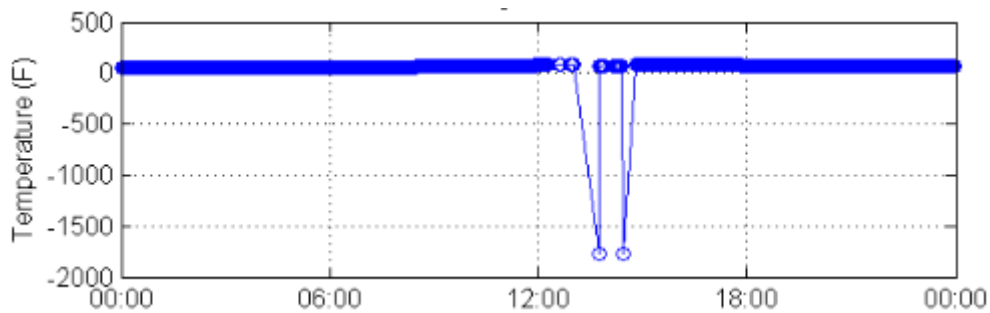


Рисунок 3.24– Графіки неочищених і вихідних даних

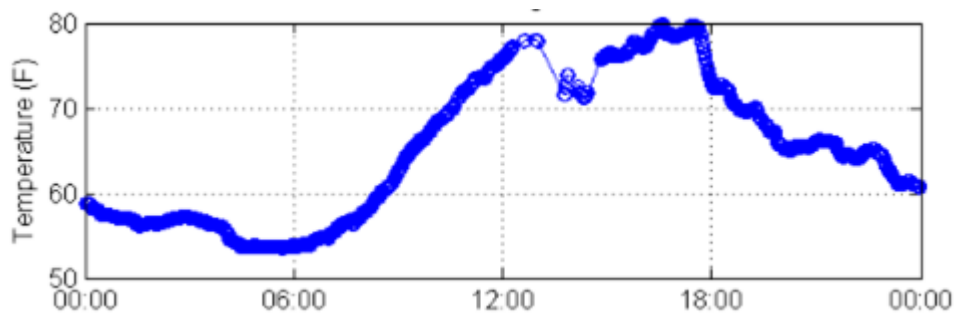


Рисунок 3.25 – Графіки очищених і вихідних даних

До ранку дані накопичилися (~ 24347 записів) і ми візуалізували їх в BI DeepSee [], ось що вийшло. Самі дані зберігаються в базі даних PostgreSQL.

Графік яскравості, температури та вологості. Явно видно світанок в районі 5:50 на рисунку 3.26-3.28.

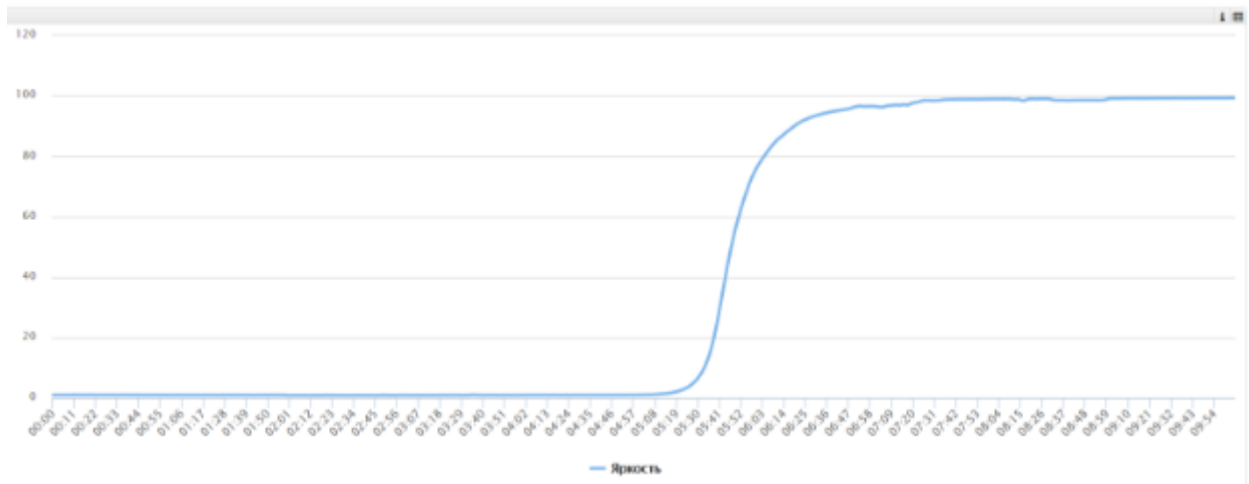


Рисунок 3.26 – Графік яскравості

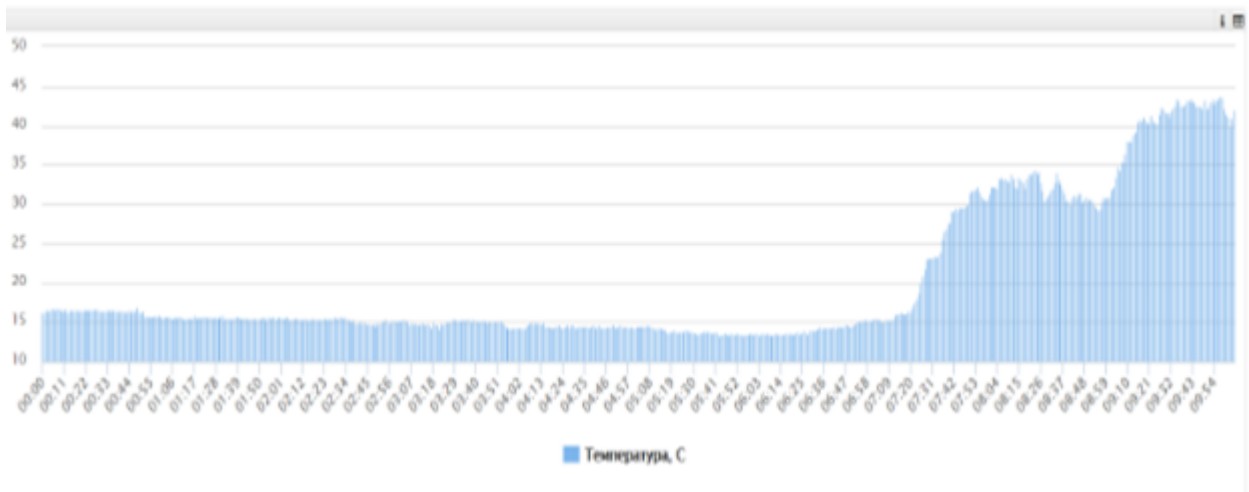


Рисунок 3.27 – Графік температури

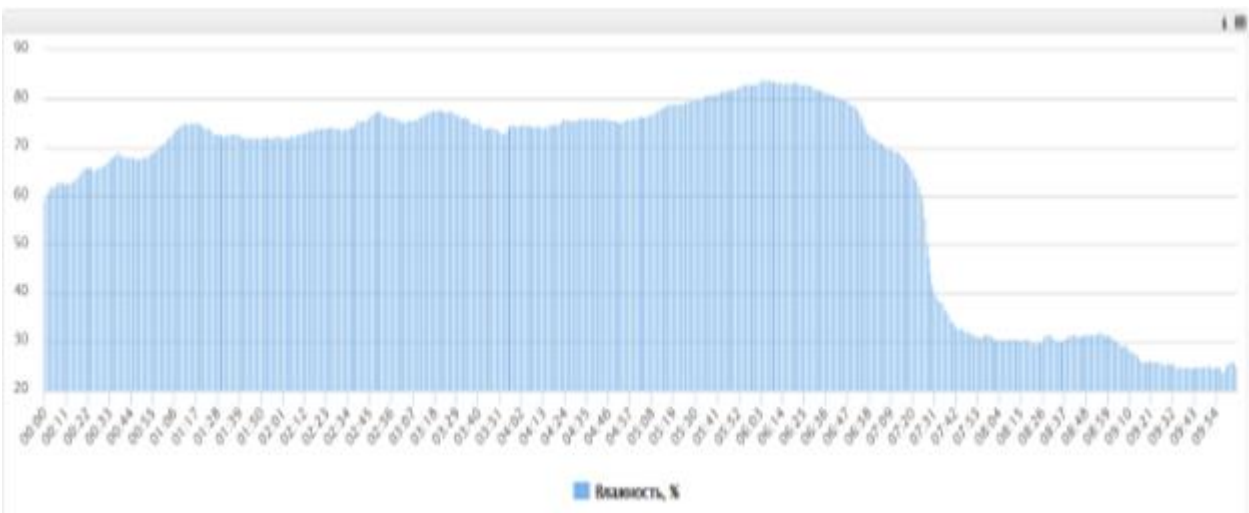


Рисунок 3.28 – Графік вологості

3.2 Створення бази даних Postgresql

Підключення бази даних Postgresql в проект виконується наступним чином.

В проект project.clj вноситься файл postgresql.clj, що встановлює драйвер PostgreSQL JDBC.

Драйвер PostgreSQL JDBC (для стислості PgJDBC) дозволяє програмам Java підключатися до бази даних PostgreSQL з використанням стандартного, незалежного від бази даних Java коду.

Також виконується запити з query, функцій query, insert, update та execute.

Функції в Postgresql мають дві категорії, які використовують зворотні виклики або канали core.async. Функції на основі каналів повертають канал, в який поміщається результат запиту або виключення.

Використана міграція, яка призначена для сумісності з потоком роботи на основі додатку git, де кілька гілок тем можуть існувати одночасно і бути об'єднані в провідну галузь в непередбаченому порядку.

Міграції розглядаються для завершення незалежно.

Додаємо залежність Migratus: [migratus "0.9.5"]

Та вводимо код :

CREATE TABLE, якщо НЕ EXISTS foo(id BIGINT) ; до каталогу папки Resources/migrations/20111206154000-create-foo-table.up.sql.

А також додаємо код DROP TABLE IF EXISTS FOO; до каталогу папки Resources/migrations/20111206154000-create-foo-table.down.sql.

Це дасть можливість створити міграцію для таблиць та їх значень в базі даних. Також це важливо, для того щоб JDBC мав метод, який дозволить відправити кілька SQL-команд для виконання. Сам Migratus розділить введені команди та виконає їх в транзакції.

Функції всередині міграції треба додатково обернути, для цього в корінь міграції вводим:

```
DO $ func $
```

```
НАЧАТЬ
```

```
PERFORM ім'я схеми function_name ('foo', 10);
```

У файлі міграції 20111206154000-create-add-board-table.up.sql. створено чотири таблиці: categories, boards, thread та message.

Файл міграції при кодуванні у разі виникнення помилки чи зміни самого коду на досконаліший, дає можливість зробити крок назад, що поверне попередню версію коду. Це зручна функція при виконання роботи програмування з великою кількістю файлів та коду.

Всі отримані дані були візуалізовані за допомогою BI DeepSee. InterSystems Caché дозволяє організувати взаємодію з великою кількістю пристроїв безпосередньо.

3.3 Висновок до розділу 3

На дослідженій технології було спроектовано компактну метеостанцію з протестованим робочим режимом. Протягом певного часу було зібрано дані прогнозу погоди, а саме вологість, температуру та освітленість дня. Зібрана станція завдяки платформі Arduino та їх компонентам, різноманітним датчикам, таким як DHT11, барометр BMP280 та інші. Дані погоди передавались зі станції розташованої на вулиці на комп'ютер сервер у вигляді візуалізації цих даних у вигляді графіку для фінального зображення. Також продемонстрували деякі способи фільтрації даних, які не проходять первинну перевірку. В подальшій розробці цього проекту буде внесення захисного корпусу для обладнання, покращення апаратного-програмного комплексу, як наладженої роботи датчиків для вимірювань параметрів погоди, збільшення їх кількості та розширення функціоналу. Кращий робочий інтерфейс для візуалізації даних на екрані. Передавати дані на мобільні платформи через Internet. Розвиток IT-технологій сприятиме розвитку сільського господарства та аграрного сектора в цілому. Поява метеостанції на полі допоможе фермерам менше залежати від примх природних стихій.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ. ЕКОЛОГІЯ

В даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Завданням даної магістерської роботи було розробка комп'ютерної система прогнозування кількості транспортних засобів, і як результат було створено Weather Climatic Condition Calculation Agriculture Transport. За цим програмним продуктом в подальшому розроблятиметься реальна система, яка значно полегшить процес організації транспортного процесу при збиранні врожаю ранньої пшениці з метою підвищення ефективності. Так як в процесі проектування використовувалося персональний комп'ютер, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера на якому буде розроблятися/використовуватися розроблена програма.

4.1 Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці. Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець). Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Роботодавець

повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників. Він не має права вимагати від працівника виконання роботи, поєднаної з явною небезпекою для життя, а також в умовах, що не відповідають законодавству про охорону праці. Працівник має право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або людей, які його оточують, і навколишнього середовища.

4.2 Аналіз стану умов праці

Робота над створенням системи автоматизованої обробки експериментальної інформації проходитиме в приміщенні квартири. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером.

4.2.1 Вимоги до приміщень

Геометричні розміри приміщення зазначені в табл. 4.1.

Таблиця 4.1 – Розміри приміщення

Найменування	Значення
Довжина, м	5
Ширина, м	5
Висота, м	3
Площа, м ²	25
Об'єм, м ³	125

Згідно з ДБН В.2.2-15-2005 «житлові будинки» [40] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм — не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

Для забезпечення потрібного рівного освітленості кімната має вікно та систему загального рівномірного освітлення, що встановлена на стелі. Для дотримання вимог пожежної безпеки встановлено порошковий вогнегасник та систему автоматичної пожежної

сигналізації.

4.2.2 Вимоги до організації місця праці

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [41] і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 4.2 – Характеристики робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	750	680 ÷ 800
Висота простору для ніг, мм	730	не менше 600
Ширина простору для ніг, мм	660	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	470	400 ÷ 500
Ширина сидіння, мм	400	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	600	не менше 300
Ширина опорної поверхні спинки, мм	500	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 ÷ 800

Робочий стіл на досліджуваному місці також містить достатньо простору для ніг. Крісло, що використовується в якості робочого сидіння, є підйомно поворотним, має підлокітники і можливість регулювання за висотою і кутом нахилу спинки, також воно м'яке і виконане з екологічної шкіри, що дає можливість працювати у комфорті. Екран монітору знаходиться на відстані 0.8 м, клавіатура має можливість регулювання кута нахилу 5-15°. Отже, за всіма параметрами робоче місце відповідає нормативним вимогам.

Приміщення кабінету знаходиться на другому поверсі трьох поверхової будівлі і має об'єм 125 м³, площу – 25 м². У цьому кабінеті обладнано три місця праці, з яких два укомплектовані ПК.

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість — близько 50%. Швидкість руху повітря не перевищує 0,2 м/с. Шум в лабораторії знаходиться на рівні 50 дБА. Система вентилявання приміщення — природна неорганізована, а опалення — централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою восьми люмінесцентних ламп, забезпечує рівень освітленості не менше 200 Лк.

У кабінеті є електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом. ПК та периферійні пристрої можуть бути джерелами електромагнітних випромінювань, аерозолів та шкідливих речовин (часток тонеру, оксидів нітрогену та озону).

За ступенем пожежної безпеки приміщення належить до категорії В. Кабінет оснащений переносним вуглекислотним вогнегасником ВВК-5 .

Наявна аптечка для надання долікарської допомоги, а також у кабінеті роблять вологе прибирання та щоденно провітрюють приміщення.

4.3 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу

Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання НПАОП 0.00.-1.28-10 «Правил охорони праці під час експлуатації електронно-обчислювальних машин», які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Переважно роботи за проектами виконують у кабінетах чи інших приміщеннях, де використовують різноманітне електрообладнання, зокрема персональні комп'ютери (ПК) та периферійні пристрої.

Основними робочими характеристиками персонального комп'ютера є наступні:

- робоча напруга $U = +220\text{В} \pm 5\%$;
- робочий струм $I = 2\text{А}$;
- споживана потужність $P = 350\text{Вт}$.

Робочі місця мають відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, затверджених постановою Головного державного санітарного лікаря України від 10.12.98 N 7 ДСанПіН 3.3.2-007-98 [42].

За умов роботи з ПК виникають наступні небезпечні та шкідливі чинники: несприятливі мікрокліматичні умови, освітлення, електромагнітні випромінювання, шум, вібрація, електричний струм, електростатичне поле, напруженість трудового процесу та інше.

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 4.3).

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількіс на оцінка	Нормативні документи
1	2	3	4
фізичні			
- підвищена температура поверхонь обладнання	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	2	ДСН 3.3.6.042-99 [43]
- підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини	-//-	4	ГОСТ 12.1.030-81[103] ГОСТ 13109-97[105]
- недостатнє освітлення робочої зони	порушення гігієнічних параметрів виробничого середовища	3	ДБН В.2.5-28:2015[44]
- підвищена яскравість світла	порушення умов праці (організації місця праці- налагодження моніторів)	1	ДСанПіН 3.3.2.007-98[45]
психофізіологічні:			
- нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	- пошук інформації для постановки теми; - пошук та аналіз аналогів і літератури; - пошук наявних технологій, моделювання та аналіз алгоритмів; - виконання роботи за темою диплома, тестування; - оформлення роботи	4	НПАОП 0.00-1.28-10[106] ДСанПіН 3.3.2.007-98[46]
- фізичні (статичне – сидіння)	порушення умов праці (організації місця праці- сидіння користувача,) та організації робочого часу - безпервна робота)	2	НПАОП 0.00-1.28-10[106] ДСанПіН 3.3.2.007-98[47]

4.4 Електробезпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ (правила улаштування електроустановок), мають апаратуру захисту від струму короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування, виконана як окрема групова три-провідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Штепсельні з'єднання та електророзетки крім контактів фазового та нульового робочого провідників мають спеціальні контакти для підключення нульового захисного провідника. Електромережа штепсельних розеток для живлення персональних ПК, укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання. Металеві труби та гнучкі металеві рукави заземлені. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

Вченими та науковцями встановлено, що під час роботи за комп'ютером найбільшому ризику підлягають:

- органи зору;
- м'язово-скелетна система;
- репродуктивна функція жінок;
- нервово-психічна діяльність з можливим формуванням стресу і депресій;
- шкіра;
- імунна система.

Близько 70% користувачів комп'ютерів скаржаться на порушення функції органу зору. Зір є важливою складовою багатогранної діяльності людини. А робота за комп'ютером, особливо безперервна, може привести до розвинення короткозорості, зниження гостроти зору, різні кон'юнктивіти та безліч інших очних хвороб. Сльозотеча, головний біль, мерехтіння в очах, головокружіння також супроводжують користувачів комп'ютерів. У найтяжчих випадках можуть виникнути глаукома, катаракта та дистрофія сітківки, які ведуть до повної сліпоти. Причина цих проблем із зором полягає у миготінні монітору, а це подразнює діє на зоровий аналізатор. А ще недостатня чіткість символів, їх спотворення на екрані, низька зручність читання безпосередньо впливають на продуктивність праці та на очі. Лікарі радять користувачам після 40 років проходити щорічне обстеження в

офтальмолога, а при необхідності — раніше й частіше.

Скелетно-м'язова система також страждає через сидяче положення користувачів. У людей, які багато працюють за комп'ютером, можуть виникнути больові відчуття в м'язах та суглобах, потилиці, попереку, пальцях рук. Виникненню цих захворювань сприяє неправильне положення тіла щодо клавіатури, відхилення ліктів від тулуба, нераціональне розміщення передпліччя та кисті рук. Робота із клавіатурою є інтенсивною та динамічною роботою кисті, яка супроводжується одночасним напруженням м'язів передпліччя й плеча. Це приводить до швидкої втоми, до розвитку нейроміозитів, оніміння, тремору та повільної рухливості пальців. Навіть до нападів судом, болі в руках і плечах уночі. Усі ці порушення можуть стати причиною інвалідності, тому вони також вимагають відповідних заходів профілактики.

Велике зорове та нервово-емоційне напруження викликають порушення функціонального стану нервової системи. Це проявляється погіршенням психологічного стану й працездатності. У медичній літературі широко описані психічні розлади, які діагностуються у користувачів. Це агресивність, нервозність, фрустрація, тривога, депресія, пригніченість, порушення сну, стресові ситуації. Стрес може бути корисним: він тренує організм і підвищує його можливості, наприклад, захисту. Проте стрес може досягнути такого рівня напруження, що виснажує захисні сили організму. Це може спровокувати різноманітні захворювання і навіть смерть. До таких захворювань належать гіпертонічна хвороба, інфаркти міокарда та інсульты, виразкові хвороби травного тракту, навіть цукровий діабет. Людина піддається дії сильних стрес-факторів, до яких належать гігієнічні умови праці в робочому приміщенні, самого трудового процесу та мікроклімату в колективі. Цій ситуації сприяють напружена нервово-психічна діяльність, гіподинамія, одноманітність, не завжди сприятлива організація робочого місця, соціальна ситуація тощо.

4.5 Освітлення

Світло є природною умовою існування людини. Воно впливає на стан вищих психічних функцій і фізіологічні процеси в організмі. Хороше освітлення діє тонізуюче, створює гарний настрій, покращує протікання основних процесів вищої нервової діяльності.

Збільшення освітленості сприяє поліпшенню працездатності навіть в тих випадках, коли процес праці практично не залежить від зорового сприйняття. При поганому освітленні людина швидко втомлюється, працює менш продуктивно, виникає потенційна небезпека помилкових дій і нещасних випадків.

Освітленість приміщення має велике значення при роботі на ПЕОМ. Вона багато в

чому визначається колірною і мережевий обстановкою. Для зменшеного поглинання світла стеля і стіни вище панелей (1,5-1,7м.). Якщо вони не облицьовані звукопоглинальним матеріалом, фарбуються білою водоемульсійною фарбою (коефіцієнт відбиття повинен бути не менше 0,7). Для забарвлення стіни панелей рекомендується віддавати перевагу світлим фарбам.

Основний потік природного світла при цій повинен бути зліва. Не допускається спрямування основного світлового потоку природного світла праворуч, ззаду і спереду працівника на ПЕОМ.

Робота на ПЕОМ може здійснюватися за таких видах освітлення:

- загальному штучному освітленні, коли відео монітори розташовуються по периметру приміщення або при центральному розташуванні робочих місць у два ряди по довжині кімнати з екранами, звернені в протилежні сторони;

- суміщене освітлення (природне + штучне) тільки при одному і трьох рядном розташуванні робочих місць, коли екран і поверхню робочого столу знаходяться перпендикулярно світла несучій стіні. При цьому штучне освітлення буде виконане стельовими або підвісними люмінесцентними світильниками, рівномірно розміщеними по стелі рядами паралельно світловим прорізам так, щоб екран відео монітора знаходився в зоні захисного кута світильника, і його проєкції не доводилися на екран. Працюючі на ПЕОМ не повинні бачити відображення світильників на екрані. Застосовувати місцеве освітлення при роботі на ПЕОМ не рекомендується.

Природне освітлення, коли робочі місця з ПЕОМ розташовуються в один ряд по довжині приміщення на відстані 0,8 - 1,0 м від стіни з віконними прорізами, і екрани знаходяться перпендикулярно цієї стіни. Основний потік природного світла при цій повинен бути зліва. Не допускається спрямування основного світлового потоку природного світла праворуч, ззаду і спереду працює на ПЕОМ. Оптимальна відстань очей до екрана відео монітора повинна становити 60-70 см, допустиме не менше 50 см. Розглядати інформацію ближче 50 см не рекомендується.

У проекті, що розробляється, передбачається використовувати суміщене освітлення. У світлий час доби використовуватиметься природне освітлення приміщення через віконні отвори, в решту часу використовуватиметься штучне освітлення. Штучне освітлення створюється газорозрядними лампами.

Штучне освітлення в робочому приміщенні передбачається здійснювати з використанням люмінесцентних джерел світла в світильниках загального освітлення, оскільки люмінесцентні лампи мають високу потужність (80 Вт), тривалий термін служби (до 10000 годин), спектральний складом випромінюваного світла, близький до сонячного.

При експлуатації ЕОМ виконується зорова робота IVв розряду точності (середня точність). При цьому нормована освітленість на робочому місці (E_n) рівна 200 лк. Джерелом природного освітлення є сонячне світло.

У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:2015 [48]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє СНіП і для даного приміщення в світлий час доби достатньо природного освітлення.

Розрахунок освітлення.

Для будівель виробництв світловий коефіцієнт приймається в межах 1/6 - 1/10:

$$\sqrt{a^2 + b^2} \cdot S_b = (1/8 \div 1/10) \cdot S_n \quad (4.1)$$

де S_b – площа віконних прорізів, м²;

S_n – площа підлоги, м².

$$S_n = 5 \cdot 4 = 20 \text{ м}^2$$

$$S_{\text{вік}} = 1/8 \cdot 20 = 2.5 \text{ м}^2$$

Приймаємо 1 вікно площею $S = 1,6 \text{ м}^2$ кожне.

Світильники загального освітлення розташовуються над робочими поверхнями в рівномірно-прямокутному порядку. Для організації освітлення в темний час доби передбачається обладнати приміщення, довжина якого складає 5 м, ширина 5 м, світильниками ЛПО2П, оснащеними лампами типа ЛБ (дві по 80 Вт) з світловим потоком 5A00 лм кожна.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників n виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} \quad (4.2)$$

де E – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

S – освітлювана площа, м²; $S = 20 \text{ м}^2$;

Z – поправочний коефіцієнт світильника ($Z = 1,15$ для ламп розжарювання та ДРЛ; $Z = 1,1$ для люмінесцентних ламп) приймаємо рівним 1,1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575

M – число люмінесцентних ламп в світильнику – 2;

F – світловий потік лампи – 5400 лм (для ЛБ-80).

Підставивши числові значення у формулу (А.2), отримуємо:

$$n = \frac{300 \cdot 20 \cdot 1,15 \cdot 1,5}{5400 \cdot 0,575 \cdot 2} \approx 1,6$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

Всі прилади, що працюють від електромережі, впливають на навколишнє їх електромагнітне поле - фізичне поле, яке взаємодіє з усіма тілами, що володіють хоча б мінімальним електричним зарядом. До таких тіл належить і людський організм. Наше тіло виробляє чимало електричних імпульсів. Сигнали нервової системи, скорочення серцевого м'яза і ряд інших функцій здійснюються за допомогою струму електричних імпульсів по живим волокнам. Електромагнітне випромінювання від приладів створює обурення в фізичному полі. На даний момент загальна «маса» таких збурень вже стала критичною і перетворилася на своєрідний вид екологічного забруднення, який неможливо побачити неозброєним оком. Найчастіше ми не відчуваємо впливу електромагнітного випромінювання, але якщо воно досягає колосальної потужності, то людина відчуває його як викид тепла. Досить потужне випромінювання можна зафіксувати за допомогою спеціальної апаратури. Але то вплив, який чинить на нас щоденне «спілкування» з електроприладами та обчислювальною технікою, залишається непоміченим.

Сьогодні багато дослідників говорять про те, що тривалого, нехай і слабого, електромагнітного випромінювання досить, щоб спровокувати такі серйозні захворювання, як хвороба Альцгеймера або Паркінсона, рак, еректильну дисфункцію, а також всілякі порушення сну і пам'яті. Найбільше страждають від подібного негативного впливу, звичайно, діти. І це ще одна причина, по якій варто обмежувати час, проведений підростаючим поколінням перед екраном монітора. Електромагнітне випромінювання негативно впливає і на розвиток плода, тому вагітні жінки теж знаходяться в групі ризику. Найгірше на здоров'я людини відбивається регулярне використання комп'ютера, телевізора і мікрохвильової печі. З усіх електроприладів ця трійця має найсильнішим випромінюванням і притому випускає хвилі НВЧ-діапазону. Монітор також створює серйозні електромагнітні обурення. Виробники техніки знають про цю особливість, тому передня частина монітора

нерідко отримує захисне покриття. А ось бічні і задні стінки - немає. Крім усього перерахованого вище, ПК деіонізує навколишнє середовище і виділяє в повітря шкідливі речовини. Це відбувається при нагріванні корпусу і материнської плати. Іншими словами, в приміщенні, де постійно працює обчислювальна техніка, повітря сухе, деіонізоване і важкий для дихання. Якщо дихальні органи - «слабке місце» в організмі або вже встигли постраждати від якихось інших факторів, великий ризик розвитку алергії або іншого захворювання органів дихання.

Щоб звести до мінімуму негативний вплив електромагнітного випромінювання від монітора, досить дотримуватися простих правил: Вибираючи монітор, краще віддати перевагу жидкокристаллическому варіанту. Випромінювання моніторів з електронно-променевою трубкою набагато сильніше, ніж у РК-аналогів. Постарайтеся розташувати монітор в кутку. Стіни будуть поглинати електромагнітне випромінювання, яке випускають бічні і задні стінки. Не забувайте вимикати монітор, якщо відходите ненадовго від робочого столу. Використання спеціальних захисних екранів і раніше актуально, особливо якщо в сім'ї є діти. Монітор повинен стояти від вашого крісла не ближче, ніж на відстані витягнутої руки. Чи не присувайте його занадто близько до обличчя і не нахиляйтеся до екрану. Існує також ряд універсальних правил для роботи за комп'ютером, які допоможуть вам зберегти власне здоров'я і продовжити термін життя техніки: Системний блок повинен розташовуватися якнайдалі від вас. Не ставте комп'ютер поруч зі спальним місцем, а краще взагалі не кладіть комп'ютерний стіл в спальні - це ідеальний варіант, який, на жаль, можуть дозволити собі далеко не всі. Не залишайте комп'ютер включеним, якщо не використовуєте його. І, відповідно, не включайте його без необхідності. Крім усього іншого, це ще й зменшить знос техніки. Намагайтеся скоротити час, який ви проводите за комп'ютером. Якщо ж ваша професійна діяльність проходить перед екраном монітора, як можна частіше переривайте роботу, щоб трохи пройтися або просто випити чаю. У свій вільний час намагайтеся не сидіти перед монітором.

Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом, приміщення в якому проводяться всі роботи відносяться до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36 В, 220 В, та 360 В. Опір контура заземлення повинен мати не більше 4 Ом.

Розрахунок проводять за допомогою методу коефіцієнта використання (екранування) електродів. Коефіцієнт використання групового заземлювача η – це відношення діючої провідності цього заземлювача до найбільш можливої його провідності за нескінченно великих відстаней між його електродами. Коефіцієнт використання вертикальних

заземлювачів η в залежності від розміщення заземлювачів та їх кількості знаходиться в межах 0,4-0,99. Взаємну екрануючу дію горизонтального заземлювача (з'єднувальної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача η .

Послідовність розрахунку.

а) Визначається необхідний опір штучних заземлювачів $R_{шт.з.}$:

$$R_{шт.з.} = \frac{R_d \cdot R_{пр.з.}}{R_{пр.з.} - R_d} \quad (4.3)$$

де $R_{пр.з.}$ – опір природних заземлювачів;

R_d – допустимий опір заземлення.

Якщо природні заземлювачі відсутні, то $R_{шт.з.} = R_d$.

Підставивши числові значення у формулу (4.3), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 40 \text{ Ом}$$

б) Опір заземлення в значній мірі залежить від питомого опору ґрунту ρ , Ом*м. Приблизне значення питомого опору глини приймаємо $\rho = 40$ Ом*м (табличне значення).

в) Розрахунковий питомий опір ґрунту, $\rho_{розр.}$, Ом*м, визначається відповідно для вертикальних заземлювачів $\rho_{розр.в.}$ і горизонтальних $\rho_{розр.г.}$, Ом*м за формулою:

$$\rho_{розр.} = \psi \cdot \rho, \quad (4.4)$$

де ψ – коефіцієнт сезонності для вертикальних заземлювачів І кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів $\rho_{розр.в.} = 1,7$ і горизонтальних $\rho_{розр.г.} = 5,5$ Ом*м.

$$\rho_{розр.в.} = 1,7 \cdot 40 = 68 \text{ Ом} \cdot \text{м}$$

$$\rho_{розр.г.} = 5,5 \cdot 40 = 220 \text{ Ом} \cdot \text{м}$$

г) Розраховується опір розтікання струму вертикального заземлювача R_v , Ом, за (4.5).

$$R_v = \frac{\rho_{розр.}}{2 \cdot \pi \cdot l_v} \left(\frac{\ln 2 \cdot l_v}{d_{ст}} + \frac{1 \cdot \ln 4 \cdot t + 1}{4 \cdot t - 1} \right) \quad (4.5)$$

де l_v – довжина розр. вертикального заземлювача (для труб - 2–3 м; $l_v = 3$ м); $d_{ст}$ – діаметр стержня (для труб - 0,03–0,05 м; $d_{ст} = 0,05$ м);

t – відстань від поверхні землі до середини заземлювача, яка визначається за формулою (4.6):

$$t = h_v + \frac{l_v}{2} \quad (4.6)$$

де h_v – глибина закладання вертикальних заземлювачів (0,8 м); тоді

$$t = 0,8 + \frac{3}{2} = 2,3 \text{ м}$$

$$R_g = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left(\ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \ln \frac{4 \cdot 2,3 + 3}{4 \cdot 2,3 - 3} \right) = 18,5 \text{ Ом}$$

г) Визначається теоретична кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання η_v :

$$n = \frac{2 \cdot R_g}{R_0} = \frac{2 \cdot 18,5}{4} = 9,25 \quad (4.7)$$

Визначається коефіцієнт використання вертикальних електродів групового заземлювача без врахування впливу з'єднувальної стрічки $\eta_v = 0,57$ (табличне значення).

д) Визначається необхідна кількість вертикальних заземлювачів з урахуванням коефіцієнта використання η_v , шт:

$$n_g = \frac{2 \cdot R_g}{R_0 \cdot \eta_g} = \frac{2 \cdot 18,5}{4 \cdot 0,57} = 16,2 \approx 16 \quad (4.8)$$

е) Визначається довжина з'єднувальної стрічки горизонтального заземлювача l_c , м:

$$l_c = 1,05 \cdot L_v \cdot (n_g - 1) \quad (4.9)$$

де L_v – відстань між вертикальними заземлювачами, (прийняти за $L_v = 3$ м); n_g – необхідна кількість вертикальних заземлювачів.

$$l_c = 1,05 \cdot 3 \cdot (16 - 1) \approx 48 \text{ м}$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова: $R_{заг} < 4$ Ом, а саме:

$$R_{заг} = \frac{18,5 \cdot 8,1}{18,5 \cdot 0,3 + 8,1 \cdot 16 \cdot 0,57} = 1,9 \leq R_0$$

При виникненню пожеж при роботі на ПЕОМ від таких можливими джерел запалювання як:

- іскри і дуги коротких замикань;
- перегрів провідників, резисторів та інших радіодеталей ПЕОМ, від тривалої перевантаження та наявності перехідного опору;
- іскри при розмиканні і розмиканні ланцюгів;
- розряди статичної електрики;

– необережному поводженню з вогнем, а також вибухи газо-повітряних і пароповітряних сумішей.

Важливу увагу слід звернути на пожежну безпеку підприємства в цілому і окремих його приміщень. В приміщеннях не повинно накопичуватися сміття, непотрібний папір, мотлох та ін. речі, які не використовуються у виробничому процесі. Наявний вільний аварійний вихід за межі приміщення в разі пожежі, бути передбачені вогнегасники. Вони повинні бути в робочому стані і перевірятися згідно з нормами. У приміщеннях повинна бути пожежна сигналізація, вогнегасник.

У разі виникнення пожежі необхідно повідомити в найближчу пожежну частину, убезпечити інших працівників і по можливості прийняти кроки по запобіганню можливих наслідків та усуненню пожежі.

4.6 Вплив на навколишнє середовища

Весь мир окутан мережею інформації, яку забезпечує нам інтернет. Більшість людей життя свою вже не представляє без персонального комп'ютера, планшета, телефону, ми перестали учити той шкода, яку самі добровільно причиняємо своєму організму і стану навколишнього середовища.

Учені і дослідники впевнені, що пора прийняти заходи по зменшенню обсягів шкоди, яку щодня наноситься навколишньому середовищу і здоров'ю. По деяким даним дослідників ООН, щоб створити один середньостатистичний персональний комп'ютер, потрібно в 10 разів більше хімічних речовин і палива, ніж вага кінцевого винаходу.

Багато сировини, яку використовують при зборці комп'ютерів, є токсичною. Вископуване паливо тільки загострює нерешену проблему глобального потепління. Відходи виробництва також не зникають, перетворюючись в сміттєві гори, або переробляються, маючи негативний вплив на екологію.

Багато користувачів і виробників помиляються, вважаючи, що з зменшенням і вдосконаленням комп'ютерів, їх негативний вплив на екологію зменшується.

Тому вчені підкреслюють важливість вторинного використання обладнання. Замість того щоб викинути гаджет в сміття, краще остаточно впевнитися, що він не потребує відновлення і тільки потім звертатися в сервісний центр комп'ютера. Можливо, він буде частково корисний в іншій збірці.

Залишається тільки сподіватися, що настане час, коли технології будуть допомагати людині, не причиняючи незворотної шкоди здоров'ю навколишнього середовища.

4.7 Висновки до розділу 4

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом написаному у роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника.

Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Були наведені розміри приміщення та наведено значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці робітника, а також – наведені інструкції з охорони праці, техніки безпеки, екології при роботі на комп'ютері.

ВИСНОВКИ

В першому розділі магістерської роботи було здійснено огляд технології побудови компактної метеостанції. Розглянуто основні проблеми служби метеоцентрів, вплив погоди та непередбачених стихій на врожайність, захворювання рослин, появу шкідників, роботу фермерів та подальший вплив на аграрний сектор економіки. Наведення прикладу внесення ІТ-технологій в розвиток сільського господарства розвинених країн та України вцілому. Майбутня роль польових метеостанцій в українському сільському господарстві, проблеми внедріння станції в роботу аграріїв. Розглянуто різновид та класифікація метеостанцій. Проведено порівняння схожих аналогів, їх характеристик, цін та функціоналу.

В другому розділі здійснено дослідження апаратно-програмного комплексу на основі платформи Arduino та її програмного забезпечення, Спосіб передачі, різновид та функціонал збираємих компонентів та датчиків таких як DHT11 (вимірювання вологи та температури) та барометр BMP280 для метеостанції. Приведені дані про характеристики об'єктів для проекту компонентів. Розглянуто роль та метод візуалізації даних прогноза погоди для станції, фільтрація не коректних даних погоди завдяки медіанній фільтрації та база даних для зберігання отриманої інформації.

В третьому розділі представлені основні етапи побудови самої метеостанції. Наведено почергове підключення компонентів таких як датчик DHT11, Arduino Uno R3 для більш швидкої передачі даних, приймач XBee для безпроводного зв'язку, трансформатор живлення та інші компоненти. Наведена схема їх підключення та код для налаштування їх роботи.

Застосований пороговий та медіанний фільтр для видалення не коректних даних отриманих з частини комплексу станції яка розташована на вулиці. Приведена візуалізація даних на екрані для таких погодних характеристик як вологість, температура та освітленість дня.

В перспективі розвитку можна додати можливість вимірювання атмосферного тиску, напрямку вітру, покращення роботи програмного та апаратного забезпечення. Поява захисного корпусу. Розроблення додатку для мобільних платформ для більш зручного користування.

ПЕРЕЛІК ПОСИЛАНЬ

1. ІТ-технології в сільському господарстві. <http://www.tsatu.edu.ua/25723/>
2. Додух Н.О. Новітні технології в сільському господарстві.
3. Сучасні інноваційні технології в сільському господарстві: основна характеристика та перспективи впровадження. http://papers.univ.kiev.ua/1/ekonomika/articles/bilinska-v-modern-innovative-technologies-in-agriculture-the-main-characteristi_24645.pdf
4. Метеостанція на поле - помічник від небесної канцелярії. <https://propozitsiya.com/meteostanciya-na-pole-pomoshchnik-ot-nebesnoy-kancelyarii>
5. Е.Таненбаум. Архітектура комп'ютера. <https://dic.academic.ru/dic.nsf/ruwiki/1610>
6. Недокументовані можливості. <https://dic.academic.ru/dic.nsf/ruwiki/1571772>
7. Інформатизація агропромислового комплексу. https://pidruchniki.com/1333080861365/informatika/informatizatsiya_agropromislovogo_kompleksu
8. Розумне поле. <https://www.ukrinform.ua/rubric-technology/2518643-rozumne-pole-u-silske-gospodartsvo-aktivno-zalucut-novi-tehnologii.html>
9. ПРОБЛЕМИ ІННОВАЦІЙНОЇ ДІЯЛЬНОСТІ В УКРАЇНІ. <http://www.economy.nayka.com.ua/?op=1&z=4545>
10. Застосування інформаційних технологій у сільському господарстві. <http://um.co.ua/10/10-12/10-121663.html>
11. Метеостанція. <https://ru.wikipedia.org/wiki/Метеостанция>
12. Класифікація цифрових метеостанцій. http://www.meteostanciya.com/tab_01.php
13. Метеостанція на поле - помічник від небесної канцелярії. <https://propozitsiya.com/meteostanciya-na-pole-pomoshchnik-ot-nebesnoy-kancelyarii>
14. Прогноз погоди в Україні і світі online. <https://news.agro-center.com.ua/ru/weather-%20ru#.XDcVW4zVLIW>
15. Візуалізація даних: корисні ресурси, література і інструменти. <https://test.ru/2014/03/14/data-visualization/>
16. МЕТЕОСТАНЦІЯ НА ПОЛЕ ПОМОЖЕТ КАРТОФЕЛЕВОДАМ. http://atagos.com.ua/news/meteostantsiya_na_pole_pomozhet_kartofelevodam/
17. National Geographic 3-View black. http://www.meteomaster.com.ua/meteogood_N41400.htm
18. La Crosse WS1650. <https://opticalmarket.com.ua/meteostantsija-la-crosse-ws1650-black.html>
19. Точка роси. https://ru.wikipedia.org/wiki/Точка_росы

20. Arduino. <https://ru.wikipedia.org/wiki/Arduino>
21. Arduino Software. <https://www.arduino.cc/en/main/software>
22. Барометр BMP280. <https://wiki.iarduino.ru/page/trema-modul-pressure-meter/>
23. Датчик DHT11. <http://arduino-diy.com/arduino-datchiki-temperature-i-vlazhnosti-DHT11-i-DHT22>
24. Arduino Uno. <https://www.sparkfun.com/products/11021>
25. SparkFun Weather Shield. <https://www.sparkfun.com/products/13956>
26. Модуль XBee з розширеним діапазоном.
<https://www.digi.com/products/embedded-systems/rf-modules/2-4-ghz-modules/xbee-802-15-4#overview>
27. XBee Shield . <https://www.sparkfun.com/products/12847>
28. Arduino Mega. <https://www.sparkfun.com/products/11061>
29. Arduino Ethernet Shield. <https://www.sparkfun.com/products/11166>
30. Комплект монтажу Arduino. <https://www.sparkfun.com/products/11417>
31. SparkFun XBee Explorer. <https://www.sparkfun.com/products/11697>
32. Power Supply Module AC 110-220v/. <https://ru.aliexpress.com/item/Power-Supply-Module-AC-110v-220v-to-DC-24V-6A-AC-DC-Switching-Power-Supply-Board/32711742016.html>
33. PostgreSQL. <https://uk.wikipedia.org/wiki/PostgreSQL>
34. Проект спрямований на забезпечення суспільства платформою і форумом інтеграції останніх подій. <https://gurt.org.ua/articles/37609/>
35. Наукова візуалізація. https://uk.wikipedia.org/wiki/Наукова_візуалізація
36. Process and Pitfalls in Writing Information Visualization Research Papers.
<http://www.cs.ubc.ca/labs/imager/tr/2008/pitfalls/>
37. Information Visualization. https://infovis-wiki.net/wiki/Information_Visualization
38. InterSystems DeepSee. <http://www.intersystems.com/ru/our-products/embedded-technologies/deepsee/>
39. Медіанна фільтрація. https://ru.bmstu.wiki/Медианная_фильтрация
40. НПАОП 0.00-4.15-98 Про розробку інструкцій з охорони праці
41. ДСанПіН 3.3.2.007-98 Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин
42. ДБН В.2.2-15-2005 Будинки і споруди
43. ДБН В.2.5-28:2015 Природне і штучне освітлення
44. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих
45. ГОСТ 12.1.030-81 ССБТ Електробезпе́чність. Захисне заземлення. Занулення

46. ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности. Допустимые уровни на рабочих местах и требования к проведению контроля.

47. „Электрическая энергия. Совместимость технических средств электромагнитных. Нормы качества электроэнергоснабжения общего назначения”

48. НПАОП 0.00-1.28-10 Правила охорони праці під час експлуатації електронно- обчислювальних машин

49. ДБН В.2.5-67:2013 Опалення, вентиляція та кондиціонування

Додаток А

Лістинг програми для обладнання метеостанції

```

Program.Niva
/*
 * A simple programme that will record humidity, temperature and brightness
 * temperature and brightness and send them to COM port
 * Output format: H=1.0;T=1.0;LL=1;
 *
 */
//PhotoResistor Pin
int lightPin = 0; //the analog pin the photoresistor is
                //connected to
                //the photoresistor is not calibrated to any units so
                //this is simply a raw sensor value (relative light)
// DHT-11 Brick pin
int DHpin = 8;
byte dat[5];
//LED Pin
int ledPin = 13; //the pin the LED is connected to
                //we are controlling brightness so
                //we use one of the PWM (pulse width
                // modulation pins)

byte read_data()
{
    byte data;
    for(int i=0; i<8; i++)
    {
        if(digitalRead(DHpin) == LOW)
        {
            while(digitalRead(DHpin) == LOW);
            delayMicroseconds(30);
            if(digitalRead(DHpin) == HIGH)
                data |= (1<<(7-i));

            while(digitalRead(DHpin) == HIGH);
        }
    }
    return data;
}
// Get data from DHT-11 Brick sensor
void start_test()
{
    digitalWrite(DHpin,LOW);
    delay(30);
    digitalWrite(DHpin,HIGH);
}

```

```

delayMicroseconds(40);
    pinMode(DHpin,INPUT);
    while(digitalRead(DHpin) == HIGH);
    delayMicroseconds(80);
    if(digitalRead(DHpin) == LOW);
    delayMicroseconds(80);
    for(int i=0;i<4;i++)
        dat[i] = read_data();
    pinMode(DHpin,OUTPUT);
    digitalWrite(DHpin,HIGH);
}
// Initial setup
void setup()
{
    Serial.begin(9600);
    pinMode(DHpin,OUTPUT);
}
/*
* loop() - this function will start after setup
* finishes and then repeat
*/
void loop()
{
    delay(1000);
    int lightLevel = analogRead(lightPin); //Read the lightlevel

    start_test();
    Serial.print("H=");
    Serial.print(dat[0], DEC);
    Serial.print('.');
    Serial.print(dat[1],DEC);
    Serial.print(";T=");
    Serial.print(dat[2], DEC);
    Serial.print('.');
    Serial.print(dat[3],DEC);
    Serial.print(";LL=");
    Serial.print(lightLevel);
    Serial.println(";");
}
/*
* Программа, регистрирующая влажность, температуру и яркость
* Отправляет результаты на COM port
* Формат вывода: H=1.0;T=1.0;LL=1;
*/

//Пин фоторезистора (аналоговый)
int lightPin = 0;

```

```

// Пин DHT-11 (цифровой)
int DHpin = 8;

// Массив, хранящий данные DHT-11
byte dat[5];

// Первоначальная настройка
void setup()
{
    Serial.begin(9600);
    pinMode(DHpin,OUTPUT);
}

/*
* Выполняется после setup()
* Основной бесконечный цикл
*/
void loop()
{
    delay(1000); // Замер примерно 1 раз в секунду
    int lightLevel = analogRead(lightPin); //Получаем уровень освещённости

    temp_hum(); // Получаем температуру и влажность в переменную dat
    // И выводим результат
    Serial.print("H=");
    Serial.print(dat[0], DEC);
    Serial.print('.');
    Serial.print(dat[1],DEC);
    Serial.print(";T=");
    Serial.print(dat[2], DEC);
    Serial.print('.');
    Serial.print(dat[3],DEC);
    Serial.print(";LL=");
    Serial.print(lightLevel);
    Serial.println(";");
}

// Получить данные от DHT-11 в dat
void temp_hum()
{
    digitalWrite(DHpin,LOW);
    delay(30);
    digitalWrite(DHpin,HIGH);
    delayMicroseconds(40);
    pinMode(DHpin,INPUT);
    while(digitalRead(DHpin) == HIGH);
}

```

```

    delayMicroseconds(80);
    if(digitalRead(DHpin) == LOW);
    delayMicroseconds(80);
    for(int i=0;i<4;i++)
    {
        dat[i] = read_data();
    }
    pinMode(DHpin,OUTPUT);
    digitalWrite(DHpin,HIGH);
}

// Получить часть данных от DHT-11
byte read_data()
{
    byte data;
    for(int i=0; i<8; i++)
    {
        if(digitalRead(DHpin) == LOW)
        {
            while(digitalRead(DHpin) == LOW);
            delayMicroseconds(30);
            if(digitalRead(DHpin) == HIGH)
            {
                data |= (1<<(7-i));
            }
            while(digitalRead(DHpin) == HIGH);
        }
    }
    return data;
}

/* Keypadtest.ino
 *
 * Пример использования библиотеки Keypad
 * Подключите Keypad к выводам Arduino указанным в
 * rowPins[] and colPins[].
 *
 */

// Репозиторий библиотеки:
// https://github.com/Chris--A/Keypad
#include <Keypad.h>

const byte ROWS = 4; // Четыре строки
const byte COLS = 4; // Три столбцы
// Карта соответствия кнопок и символов
char keys[ROWS][COLS] = {
    {'1','2','3','A'},

```

```

    {'4','5','6','B'},
    {'7','8','9','C'},
    {'*','0','#','D'}
};
// Подключите разъёмы keypad 1-8 (сверху-вниз) к Arduino разъёмам 11-4. 1->11, 2->10, ... , 8->4
// Подключите keypad ROW0, ROW1, ROW2 и ROW3 к ЭТИМ выводам Arduino
byte rowPins[ROWS] = { 7, 6, 5, 4 };
// Подключите keypad COL0, COL1 and COL2 к ЭТИМ выводам Arduino
byte colPins[COLS] = { 8, 9, 10, 11 };

// Инициализация Keypad
Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

#define ledpin 13

void setup()
{
    Serial.begin(9600);
}

void loop()
{
    char key = kpd.getKey(); // Получаем нажатую кнопку
    if(key)
    {
        switch (key)
        {
            case '#':
                Serial.println();
            default:
                Serial.print(key);
        }
    }
}

```

The MIT License (MIT)

Copyright (c) 2015 InterSystems Corp.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all

copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```

/* Led.ino
 *
 * Пример получения данных по COM порту
 * Подключите светодиод к выводу ledPin
 *
 */
// Вывод светодиода (цифровой)
#define ledpin 8

// "Буфер" поступающих данных
String inString = "";

void setup() {
    Serial.begin(9600);
    pinMode(ledpin, OUTPUT);
    digitalWrite(ledpin, LOW);
}

void loop() {
    // Получаем данные из COM порта
    while (Serial.available() > 0) {
        int inChar = Serial.read();
        if (isDigit(inChar)) {
            // Получаем 1 символ,
            // Прибавляем его к строке
            inString += (char)inChar;
        }
        // Доходим до новой строки
        if (inChar == '\n') {
            // Включаем светодиод
            digitalWrite(ledpin, HIGH);
            int time = inString.toInt();
            delay(time);
            digitalWrite(ledpin, LOW);
            // Обнуляем полученную строку
            inString = "";
        }
    }
}
<?xml version="1.0" encoding="UTF-8"?>

```

```

<Export generator="Cache" version="25">
<Routine name="ZAUTENTICATE" type="MAC" languagemode="0"><![CDATA[
ZAUTENTICATE(ServiceName,Namespace,Username>Password,Credentials,Properties) PUBLIC {
    #Include %occStatus
    Write "Logged as " _ $Username
    Quit $$$OK
}
GetCredentials(ServiceName,Namespace,Username>Password,Credentials) Public {
    #Include %occErrors
    #Include %occStatus
    Quit ##class(Arduino.Delegated).GetCredentials(.Username, .Password)
}
]]></Routine>
</Export>
Arduino-Brightness.pivot.dfi
Arduino-Brightness02.pivot.dfi
Arduino-BrightnessEd.pivot.dfi
Arduino-Humidity.pivot.dfi
Arduino-Humidity02.pivot.dfi
Arduino-Main.dashboard.dfi
Arduino-NightMeasures.dashboard.dfi
Arduino-NightMeasures02.dashboard.dfi
Arduino-Temperature.pivot.dfi
Arduino-Temperature02.pivot.dfi
Arduino.pkg
/* Led.ino
*
* Пример получения данных по COM порту
* Подключите светодиод к выводу ledPin
*
*/

// Вывод светодиода (цифровой)
#define ledpin 8

// "Буфер" поступающих данных
String inString = "";

void setup() {
    Serial.begin(9600);
    pinMode(ledpin, OUTPUT);
    digitalWrite(ledpin, LOW);
}

void loop() {
    // Получаем данные из COM порта
    while (Serial.available() > 0) {

```



```

int inChar = Serial.read();
if (isDigit(inChar)) {
    // Получаем 1 символ,
    // Прибавляем его к строке
    inString += (char)inChar;
}

// Доходим до новой строки
if (inChar == '\n') {
    // Включаем светодиод
    digitalWrite(ledpin, HIGH);
    int time = inString.toInt();
    delay(time);
    digitalWrite(ledpin, LOW);
    // Обнуляем полученную строку
    inString = "";
}
}
}
/// Отправляем на порт строку 1000\n
ClassMethod SendSerial()
{
    set port = "COM1"
    open port:(::" 0801n0"/:BAUD=9600) // Открываем устройство
    set old = $IO // Записываем текущее устройство ввода-вывода
    use port // Переключаемся на com порт
    write $Char(10) // Отправка пробного пакета данных
    hang 1
    write 1000 _ $Char(10) // Передаём строку 1000\n
    use old // Переключаем вывод на терминал
    close port // Закрываем устройство
}
/* Keypadtest.ino
*
* Пример использования библиотеки Keypad
* Подключите Keypad к выводам Arduino указанным в
* rowPins[] and colPins[].
*
*/

// Репозиторий библиотеки:
// https://github.com/Chris--A/Keypad
#include <Keypad.h>

const byte ROWS = 4; // Четыре строки
const byte COLS = 4; // Три столбцы
// Карта соответствия кнопок и символов

```

```

char keys[ROWS][COLS] = {
    {'1','2','3','A'},
    {'4','5','6','B'},
    {'7','8','9','C'},
    {'*','0','#','D'}
};

// Подключите разъёмы keypad 1-8 (сверху-вниз) к Arduino разъёмам 11-4. 1->11, 2->10, ... , 8->4
// Подключите keypad ROW0, ROW1, ROW2 и ROW3 к ЭТИМ выводам Arduino
byte rowPins[ROWS] = { 7, 6, 5, 4 };
// Подключите keypad COL0, COL1 and COL2 к ЭТИМ выводам Arduino
byte colPins[COLS] = { 8, 9, 10, 11 };

// Инициализация Keypad
Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

#define ledpin 13

void setup()
{
    Serial.begin(9600);
}

void loop()
{
    char key = kpd.getKey(); // Получаем нажатую кнопку
    if(key)
    {
        switch (key)
        {
            case '#':
                Serial.println();
            default:
                Serial.print(key);
        }
    }
}

/// Получение одной строки данных (до конца строки)
ClassMethod ReceiveOneLine() As %String
{
    port = "COM1"
    set str=""
    try {
        open port:(::" 0801n0":/BAUD=9600)
        set old = $io // Запоминаем текущее устройство ввода-вывода
        use port
        read str // Читаем, пока не встретим символ конца строки
        use old
    }
}

```

```

        close port
    } catch ex {
        close port
    }
    return str
}
}
/* Meteo.ino
*
* Программа, регистрирующая влажность, температуру и яркость
* Отправляет результаты на COM port
* Формат вывода: H=1.0;T=1.0;LL=1;
*/

//Пин фоторезистора (аналоговый)
int lightPin = 0;

// Пин DHT-11 (цифровой)
int DHpin = 8;

// Массив, хранящий данные DHT-11
byte dat[5];

// Первоначальная настройка
void setup()
{
    Serial.begin(9600);
    pinMode(DHpin,OUTPUT);
}

/*
* Выполняется после setup()
* Основной бесконечный цикл
*/
void loop()
{
    delay(1000); // Замер примерно 1 раз в секунду
    int lightLevel = analogRead(lightPin); //Получаем уровень освещённости

    temp_hum(); // Получаем температуру и влажность в переменную dat
    // И выводим результат
    Serial.print("H=");
    Serial.print(dat[0], DEC);
    Serial.print('.');
    Serial.print(dat[1],DEC);
    Serial.print(";T=");
    Serial.print(dat[2], DEC);
    Serial.print('.');

```

```

    Serial.print(dat[3],DEC);
    Serial.print(";LL=");
    Serial.print(lightLevel);
    Serial.println(";");
}

// Получить данные от DHT-11 в dat
void temp_hum()
{
    digitalWrite(DHpin,LOW);
    delay(30);
    digitalWrite(DHpin,HIGH);
    delayMicroseconds(40);
    pinMode(DHpin,INPUT);
    while(digitalRead(DHpin) == HIGH);
    delayMicroseconds(80);
    if(digitalRead(DHpin) == LOW);
    delayMicroseconds(80);
    for(int i=0;i<4;i++)
    {
        dat[i] = read_data();
    }
    pinMode(DHpin,OUTPUT);
    digitalWrite(DHpin,HIGH);
}

// Получить часть данных от DHT-11
byte read_data()
{
    byte data;
    for(int i=0; i<8; i++)
    {
        if(digitalRead(DHpin) == LOW)
        {
            while(digitalRead(DHpin) == LOW);
            delayMicroseconds(30);
            if(digitalRead(DHpin) == HIGH)
            {
                data |= (1<<(7-i));
            }
            while(digitalRead(DHpin) == HIGH);
        }
    }
    return data;
}

Class Arduino.Info Extends %Persistent
{

```

```

Parameter SerialPort As %String = "com1";

Property DateTime As %DateTime;

Property Temperature As %Double;

Property Humidity As %Double(MAXVAL = 100, MINVAL = 0);

Property Brightness As %Double(MAXVAL = 100, MINVAL = 0);

Property Volume As %Double(MAXVAL = 100, MINVAL = 0);

ClassMethod AddNew(Temperature = 0, Humidity = 0, Brightness = 0, Volume = 0)
{
    set obj = ..%New()
    set obj.DateTime=$ZDT($H,3,1)
    set obj.Temperature=Temperature
    set obj.Humidity=Humidity
    set obj.Brightness=Brightness/1023*100
    set obj.Volume=Volume
    write $SYSTEM.Status.DisplayError(obj.%Save())
}
/// Получаем поток данных в формате H=34.0;T=24.0;LL=605;\n
/// И преобразуем их в объекты класса Arduino.Info
ClassMethod ReceiveSerial(port = {..#SerialPort})
{
    try {
        open port:(::" 0801n0"/:BAUD=9600)
        set old = $IO
        use port
        for {
            read x //Читаем одну строку
            set Humidity = $Piece($Piece(x, ";", 1), "=", 2)
            set Temperature = $Piece($Piece(x, ";", 2), "=", 2)
            set Brightness = $Piece($Piece(x, ";", 3), "=", 2)
            if (x '= "') {
                do ..AddNew(Temperature, Humidity, Brightness) // Добавляем данные
            }
        }
    } catch anyError {
        close port
    }
}
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-Temperature02.pivot.dfi">

```

```

<pivot xmlns="http://www.intersystems.com/deepsee/library" name="Temperature02" folderName="Arduino" title=""
description="" keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-
18T08:21:41.106Z" createdBy="_SYSTEM" category="" bookCover="" mdx="" cellWidth="120" columnHeaderStyle=""
rowHeaderStyle="" cellStyle="" rowLabelSpan="true" columnLabelSpan="true" cellHeight="22" showEmptyRows="false"
showEmptyColumns="false" cubeName="ARDUINO" caption="" listing="" listingRows="" showStatus="true" pageSize="100"
colorScale="" rowTotals="false" columnTotals="false" rowTotalAgg="sum" columnTotalAgg="sum" rowTotalSource="page"
showZebra="false" showRowCaption="true" printTitle="" printSubtitle="" printSubtitleOn="" showUser="" printPageSize=""
printOrientation="1" printMarginTop="" printMarginLeft="" printMarginRight="" printMarginBottom="" printLabelWidth=""
printCellWidth="" autoExecute="true" manualMode="false" userMDX="" chartMarginTop="" chartMarginLeft=""
chartMarginRight="" chartMarginBottom="" maxRows="" borderLeftCell="" borderRightCell="" borderTopCell=""
borderBottomCell="" borderLeftCol="" borderRightCol="" borderTopCol="" borderBottomCol="" borderLeftRow=""
borderRightRow="" borderTopRow="" borderBottomRow="" showFilters="" showListingFilters="" showDate=""
listingFontSize="" showZebraStripes="" filterTableStyle="" filterTableCaptionStyle="" filterTableItemStyle=""
nowDisplayFormat="" measureLocation="columns" backgroundImage="" backgroundOpacity=".12">
  <rowAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false" filterExpression=""
orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm=""
levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle=""
levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowAxisOptions>
  <columnAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false"
filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction=""
aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false"
levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </columnAxisOptions>
  <rowLevel spec="[Время измерения].[H1].[DateTime].Members" key="" value="" text="Дара" headEnabled="false"
headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC"
aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType=""
drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowLevel>
  <measure spec="[Measures].[Temperature]" key="" value="" text="Temperature" headEnabled="false" headCount=""
filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false"
aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0"
advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </measure>
</pivot>
</Document></Export>
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-Temperature.pivot.dfi">
<pivot xmlns="http://www.intersystems.com/deepsee/library" name="Temperature" folderName="Arduino" title="" description=""
keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-18T07:38:40.414Z"
createdBy="_SYSTEM" category="" bookCover="" mdx="" cellWidth="120" columnHeaderStyle="" rowHeaderStyle=""
cellStyle="" rowLabelSpan="true" columnLabelSpan="true" cellHeight="22" showEmptyRows="false"
showEmptyColumns="false" cubeName="ARDUINO" caption="" listing="" listingRows="" showStatus="true" pageSize="100"
colorScale="" rowTotals="false" columnTotals="false" rowTotalAgg="sum" columnTotalAgg="sum" rowTotalSource="page"
showZebra="false" showRowCaption="true" printTitle="" printSubtitle="" printSubtitleOn="" showUser="" printPageSize=""
printOrientation="1" printMarginTop="" printMarginLeft="" printMarginRight="" printMarginBottom="" printLabelWidth=""

```

```

printCellWidth="" autoExecute="true" manualMode="false" userMDX="" chartMarginTop="" chartMarginLeft=""
chartMarginRight="" chartMarginBottom="" maxRows="" borderLeftCell="" borderRightCell="" borderTopCell=""
borderBottomCell="" borderLeftCol="" borderRightCol="" borderTopCol="" borderBottomCol="" borderLeftRow=""
borderRightRow="" borderTopRow="" borderBottomRow="" showFilters="" showListingFilters="" showDate=""
listingFontSize="" showZebraStripes="" filterTableStyle="" filterTableCaptionStyle="" filterTableItemStyle=""
nowDisplayFormat="" measureLocation="columns" backgroundImage="" backgroundOpacity=".12">
  <rowAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false" filterExpression=""
orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm=""
levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle=""
levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowAxisOptions>
  <columnAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false"
filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction=""
aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false"
levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </columnAxisOptions>
  <rowLevel spec="[Время измерения].[H1].[Minute].Members" key="" value="" text="Минуты" headEnabled="false"
headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC"
aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType=""
drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowLevel>
  <filter spec="[Время измерения].[H1].[DateTime].Members" key="&[63813]" value="" text="2015-09-18"
headEnabled="false" headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression=""
orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat=""
levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false"
drilldownSpec="" enabled="true">
  </filter>
  <measure spec="[Measures].[Temperature]" key="" value="" text="Temperature" headEnabled="false" headCount=""
filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false"
aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0"
advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </measure>
</pivot>
</Document></Export>
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-NightMeasures02.dashboard.dfi">
<dashboard xmlns="http://www.intersystems.com/deepsee/library" name="NightMeasures02" folderName="Arduino" title=""
description="" keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-
18T08:22:17.416Z" createdBy="_SYSTEM" category="" bookCover="" scheme="" worklistCount="2" snapTo="true"
snapGrid="true" gridRows="10" gridCols="10" canResize="true" canModify="true" showTitleBar="true" companyName=""
companyLogo="" companyStyle="">
  <widget name="WBrightness02" type="pivot" subtype="columnChart" subtypeClass="barChart" title=""
dataSource="Arduino/Brightness02.pivot" dataLink="" drillDownDataSource="" width="200" height="200" sidebarContent=""
showSidebar="false" sidebarWidth="" maximized="false" homeRowL="0" homeColL="0" colSpanL="10" rowSpanL="3"
showToolBar="true" theme="" dataColorList="">

```

```

    <control name="" action="applyFilter" target="" targetProperty="[Время измерения].[Н1].[DateTime]" location="widget"
type="auto" controlClass="" label="Дата" title="" value="" text="" readOnly="false" valueList="" displayList=""
activeWhen=""></control>
</widget>
<widget name="WTemperature02" type="pivot" subtype="columnChart" subtypeClass="barChart" title=""
dataSource="Arduino/Temperature02.pivot" dataLink="" drillDownDataSource="" width="200" height="200" sidebarContent=""
showSidebar="false" sidebarWidth="" maximized="false" homeRowL="3" homeColL="0" colSpanL="10" rowSpanL="3"
showToolBar="true" theme="" dataColorList="">
</widget>
<widget name="WHumidity02" type="pivot" subtype="columnChart" subtypeClass="barChart" title=""
dataSource="Arduino/Humidity02.pivot" dataLink="" drillDownDataSource="" width="200" height="200" sidebarContent=""
showSidebar="false" sidebarWidth="" maximized="false" homeRowL="6" homeColL="0" colSpanL="10" rowSpanL="3"
showToolBar="true" theme="" dataColorList="">
</widget>
</dashboard>
</Document></Export>
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-NightMeasures.dashboard.dfi">
<dashboard xmlns="http://www.intersystems.com/deepsee/library" name="NightMeasures" folderName="Arduino" title=""
description="" keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-
18T07:40:10.016Z" createdBy="_SYSTEM" category="" bookCover="" scheme="" worklistCount="0" snapTo="true"
snapGrid="true" gridRows="10" gridCols="10" canResize="true" canModify="true" showTitleBar="true" companyName=""
companyLogo="" companyStyle="">
<widget name="WBrightness" type="pivot" subtype="columnChart" subtypeClass="barChart" title=""
dataSource="Arduino/Brightness.pivot" dataLink="" drillDownDataSource="" width="200" height="200" sidebarContent=""
showSidebar="false" sidebarWidth="" maximized="false" homeRowL="0" homeColL="0" colSpanL="10" rowSpanL="3"
showToolBar="true" theme="" dataColorList="">
    <control name="" action="applyFilter" target="*" targetProperty="[Время измерения].[Н1].[DateTime]" location="widget"
type="auto" controlClass="" label="Дата" title="" value="" text="" readOnly="false" valueList="" displayList=""
activeWhen=""></control>
</widget>
<widget name="WTemperature" type="pivot" subtype="columnChart" subtypeClass="barChart" title=""
dataSource="Arduino/Temperature.pivot" dataLink="" drillDownDataSource="" width="200" height="200" sidebarContent=""
showSidebar="false" sidebarWidth="" maximized="false" homeRowL="3" homeColL="0" colSpanL="10" rowSpanL="3"
showToolBar="true" theme="" dataColorList="">
</widget>
<widget name="WHumidity" type="pivot" subtype="columnChart" subtypeClass="barChart" title=""
dataSource="Arduino/Humidity.pivot" dataLink="" drillDownDataSource="" width="200" height="200" sidebarContent=""
showSidebar="false" sidebarWidth="" maximized="false" homeRowL="6" homeColL="0" colSpanL="10" rowSpanL="3"
showToolBar="true" theme="" dataColorList="">
</widget>
</dashboard>
</Document></Export>
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-Main.dashboard.dfi">

```



```

<dashboard xmlns="http://www.intersystems.com/deepsee/library" name="Main" folderName="Arduino" title="" description=""
keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-18T07:56:00.169Z"
createdBy="UnknownUser" category="" bookCover="" scheme="" worklistCount="2" snapTo="false" snapGrid="true"
gridRows="10" gridCols="10" canResize="true" canModify="true" showTitleBar="true" companyName="" companyLogo=""
companyStyle="">
  <widget name="Widget1" type="pivot" subtype="lineChart" subtypeClass="lineChart" title=""
dataSource="Arduino/BrightnessEd.pivot" dataLink="" drillDownDataSource="" width="200" height="200" sidebarContent=""
showSidebar="false" sidebarWidth="" maximized="false" homeRowL="0" homeColL="0" colSpanL="8" rowSpanL="8"
showToolBar="true" theme="" dataColorList="">
  </widget>
</dashboard>
</Document></Export>
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-Humidity02.pivot.dfi">
<pivot xmlns="http://www.intersystems.com/deepsee/library" name="Humidity02" folderName="Arduino" title="" description=""
keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-18T08:20:51.172Z"
createdBy="_SYSTEM" category="" bookCover="" mdx="" cellWidth="120" columnHeaderStyle="" rowHeaderStyle=""
cellStyle="" rowLabelSpan="true" columnLabelSpan="true" cellHeight="22" showEmptyRows="false"
showEmptyColumns="false" cubeName="ARDUINO" caption="" listing="" listingRows="" showStatus="true" pageSize="100"
colorScale="" rowTotals="false" columnTotals="false" rowTotalAgg="sum" columnTotalAgg="sum" rowTotalSource="page"
showZebra="false" showRowCaption="true" printTitle="" printSubtitle="" printSubtitleOn="" showUser="" printPageSize=""
printOrientation="1" printMarginTop="" printMarginLeft="" printMarginRight="" printMarginBottom="" printLabelWidth=""
printCellWidth="" autoExecute="true" manualMode="false" userMDX="" chartMarginTop="" chartMarginLeft=""
chartMarginRight="" chartMarginBottom="" maxRows="" borderLeftCell="" borderRightCell="" borderTopCell=""
borderBottomCell="" borderLeftCol="" borderRightCol="" borderTopCol="" borderBottomCol="" borderLeftRow=""
borderRightRow="" borderTopRow="" borderBottomRow="" showFilters="" showListingFilters="" showDate=""
listingFontSize="" showZebraStripes="" filterTableStyle="" filterTableCaptionStyle="" filterTableItemStyle=""
nowDisplayFormat="" measureLocation="columns" backgroundImage="" backgroundOpacity=".12">
  <rowAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false" filterExpression=""
orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm=""
levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle=""
levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowAxisOptions>
  <columnAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false"
filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction=""
aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false"
levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </columnAxisOptions>
  <rowLevel spec="[Время измерения].[H1].[DateTime].Members" key="" value="" text="Дата" headEnabled="false"
headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC"
aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType=""
drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowLevel>
  <measure spec="[Measures].[Humidity]" key="" value="" text="Влажность" headEnabled="false" headCount=""
filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false"

```

```

aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0"
advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </measure>
</pivot>
</Document></Export>
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-Brightness.pivot.dfi">
<pivot xmlns="http://www.intersystems.com/deepsee/library" name="Brightness" folderName="Arduino" title="" description=""
keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-18T07:38:15.365Z"
createdBy="_SYSTEM" category="" bookCover="" mdx="" cellWidth="120" columnHeaderStyle="" rowHeaderStyle=""
cellStyle="" rowLabelSpan="true" columnLabelSpan="true" cellHeight="22" showEmptyRows="false"
showEmptyColumns="false" cubeName="ARDUINO" caption="" listing="" listingRows="" showStatus="true" pageSize="100"
colorScale="" rowTotals="false" columnTotals="false" rowTotalAgg="sum" columnTotalAgg="sum" rowTotalSource="page"
showZebra="false" showRowCaption="true" printTitle="" printSubtitle="" printSubtitleOn="" showUser="" printPageSize=""
printOrientation="1" printMarginTop="" printMarginLeft="" printMarginRight="" printMarginBottom="" printLabelWidth=""
printCellWidth="" autoExecute="true" manualMode="false" userMDX="" chartMarginTop="" chartMarginLeft=""
chartMarginRight="" chartMarginBottom="" maxRows="" borderLeftCell="" borderRightCell="" borderTopCell=""
borderBottomCell="" borderLeftCol="" borderRightCol="" borderTopCol="" borderBottomCol="" borderLeftRow=""
borderRightRow="" borderTopRow="" borderBottomRow="" showFilters="" showListingFilters="" showDate=""
listingFontSize="" showZebraStripes="" filterTableStyle="" filterTableCaptionStyle="" filterTableItemStyle=""
nowDisplayFormat="" measureLocation="columns" backgroundImage="" backgroundOpacity=".12">
  <rowAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false" filterExpression=""
orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm=""
levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle=""
levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
    </rowAxisOptions>
    <columnAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false"
filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction=""
aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false"
levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
      </columnAxisOptions>
      <rowLevel spec="[Время измерения].[H1].[Minute].Members" key="" value="" text="Минуты" headEnabled="false"
headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC"
aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType=""
drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
        </rowLevel>
        <filter spec="[Время измерения].[H1].[DateTime].Members" key="&[63813]" value="" text="2015-09-18"
headEnabled="false" headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression=""
orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat=""
levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false"
drilldownSpec="" enabled="true">
          </filter>
          <measure spec="[Measures].[Brightness]" key="" value="" text="Brightness" headEnabled="false" headCount=""
filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false"
aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0"
advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">

```

```

</measure>
</pivot>
</Document></Export>
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-Brightness02.pivot.dfi">
<pivot xmlns="http://www.intersystems.com/deepsee/library" name="Brightness02" folderName="Arduino" title="" description=""
keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-18T08:21:15.544Z"
createdBy="_SYSTEM" category="" bookCover="" mdx="" cellWidth="120" columnHeaderStyle="" rowHeaderStyle=""
cellStyle="" rowLabelSpan="true" columnLabelSpan="true" cellHeight="22" showEmptyRows="false"
showEmptyColumns="false" cubeName="ARDUINO" caption="" listing="" listingRows="" showStatus="true" pageSize="100"
colorScale="" rowTotals="false" columnTotals="false" rowTotalAgg="sum" columnTotalAgg="sum" rowTotalSource="page"
showZebra="false" showRowCaption="true" printTitle="" printSubtitle="" printSubtitleOn="" showUser="" printPageSize=""
printOrientation="1" printMarginTop="" printMarginLeft="" printMarginRight="" printMarginBottom="" printLabelWidth=""
printCellWidth="" autoExecute="true" manualMode="false" userMDX="" chartMarginTop="" chartMarginLeft=""
chartMarginRight="" chartMarginBottom="" maxRows="" borderLeftCell="" borderRightCell="" borderTopCell=""
borderBottomCell="" borderLeftCol="" borderRightCol="" borderTopCol="" borderBottomCol="" borderLeftRow=""
borderRightRow="" borderTopRow="" borderBottomRow="" showFilters="" showListingFilters="" showDate=""
listingFontSize="" showZebraStripes="" filterTableStyle="" filterTableCaptionStyle="" filterTableItemStyle=""
nowDisplayFormat="" measureLocation="columns" backgroundImage="" backgroundOpacity=".12">
  <rowAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false" filterExpression=""
orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm=""
levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle=""
levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
    </rowAxisOptions>
    <columnAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false"
filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction=""
aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false"
levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
      </columnAxisOptions>
      <rowLevel spec="[Время измерения].[H1].[DateTime].Members" key="" value="" text="Дата" headEnabled="false"
headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC"
aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType=""
drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
        </rowLevel>
        <measure spec="[Measures].[Brightness]" key="" value="" text="Brightness" headEnabled="false" headCount=""
filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false"
aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0"
advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
          </measure>
        </pivot>
      </Document></Export>
    <?xml version="1.0" encoding="UTF-8"?>
    <Export generator="Cache" version="25">
    <Document name="Arduino-BrightnessEd.pivot.dfi">
    <pivot xmlns="http://www.intersystems.com/deepsee/library" name="BrightnessEd" folderName="Arduino" title="" description=""
keywords="" owner="" shared="true" public="false" locked="false" resource="" timeCreated="2015-09-18T07:50:38.212Z"

```

```

createdBy="UnknownUser" category="" bookCover="" mdx="" cellWidth="120" columnHeaderStyle="" rowHeaderStyle=""
cellStyle="" rowLabelSpan="true" columnLabelSpan="true" cellHeight="22" showEmptyRows="false"
showEmptyColumns="false" cubeName="Arduino" caption="" listing="" listingRows="" showStatus="true" pageSize="100"
colorScale="" rowTotals="false" columnTotals="false" rowTotalAgg="sum" columnTotalAgg="sum" rowTotalSource="page"
showZebra="false" showRowCaption="true" printTitle="" printSubtitle="" printSubtitleOn="" showUser="" printPageSize=""
printOrientation="1" printMarginTop="" printMarginLeft="" printMarginRight="" printMarginBottom="" printLabelWidth=""
printCellWidth="" autoExecute="true" manualMode="false" userMDX="" chartMarginTop="" chartMarginLeft=""
chartMarginRight="" chartMarginBottom="" maxRows="" borderLeftCell="" borderRightCell="" borderTopCell=""
borderBottomCell="" borderLeftCol="" borderRightCol="" borderTopCol="" borderBottomCol="" borderLeftRow=""
borderRightRow="" borderTopRow="" borderBottomRow="" showFilters="" showListingFilters="" showDate=""
listingFontSize="" showZebraStripes="" filterTableStyle="" filterTableCaptionStyle="" filterTableItemStyle=""
nowDisplayFormat="" measureLocation="columns" backgroundImage="" backgroundOpacity=".12">
  <rowAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false" filterExpression=""
orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm=""
levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle=""
levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowAxisOptions>
  <columnAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false"
filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction=""
aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false"
levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </columnAxisOptions>
  <rowLevel spec="[Время измерения].[H1].[Minute].Members" key="" value="" text="Минуты" headEnabled="false"
headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC"
aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType=""
drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowLevel>
  <filter spec="[Время измерения].[H1].[Minute].Members" key="&[0]:&[604]" value="" text="00:00:10:04"
headEnabled="false" headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression=""
orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat=""
levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false"
drilldownSpec="" enabled="true">
  </filter>
  <measure spec="[Measures].[Brightness]" key="" value="" text="Brightness" headEnabled="false" headCount=""
filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false"
aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0"
advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </measure>
</pivot>
</Document></Export>
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-Humidity.pivot.dfi">
<pivot xmlns="http://www.intersystems.com/deepsee/library" name="Humidity" folderName="Arduino" title="" description=""
keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-18T07:38:57.616Z"
createdBy="_SYSTEM" category="" bookCover="" mdx="" cellWidth="120" columnHeaderStyle="" rowHeaderStyle=""
cellStyle="" rowLabelSpan="true" columnLabelSpan="true" cellHeight="22" showEmptyRows="false"

```

```

showEmptyColumns="false" cubeName="ARDUINO" caption="" listing="" listingRows="" showStatus="true" pageSize="100"
colorScale="" rowTotals="false" columnTotals="false" rowTotalAgg="sum" columnTotalAgg="sum" rowTotalSource="page"
showZebra="false" showRowCaption="true" printTitle="" printSubtitle="" printSubtitleOn="" showUser="" printPageSize=""
printOrientation="1" printMarginTop="" printMarginLeft="" printMarginRight="" printMarginBottom="" printLabelWidth=""
printCellWidth="" autoExecute="true" manualMode="false" userMDX="" chartMarginTop="" chartMarginLeft=""
chartMarginRight="" chartMarginBottom="" maxRows="" borderLeftCell="" borderRightCell="" borderTopCell=""
borderBottomCell="" borderLeftCol="" borderRightCol="" borderTopCol="" borderBottomCol="" borderLeftRow=""
borderRightRow="" borderTopRow="" borderBottomRow="" showFilters="" showListingFilters="" showDate=""
listingFontSize="" showZebraStripes="" filterTableStyle="" filterTableCaptionStyle="" filterTableItemStyle=""
nowDisplayFormat="" measureLocation="columns" backgroundImage="" backgroundOpacity=".12">
  <rowAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false" filterExpression=""
orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm=""
levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle=""
levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowAxisOptions>
  <columnAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false"
filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction=""
aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false"
levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </columnAxisOptions>
  <rowLevel spec="[Время измерения].[H1].[Minute].Members" key="" value="" text="Минуты" headEnabled="false"
headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC"
aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType=""
drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </rowLevel>
  <filter spec="[Время измерения].[H1].[DateTime].Members" key="&[63813]" value="" text="2015-09-18"
headEnabled="false" headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression=""
orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat=""
levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false"
drilldownSpec="" enabled="true">
  </filter>
  <measure spec="[Measures].[Humidity]" key="" value="" text="Влажность" headEnabled="false" headCount=""
filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false"
aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0"
advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
  </measure>
</pivot>
</Document></Export>
<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25">
<Document name="Arduino-Humidity02.pivot.dfi">
<pivot xmlns="http://www.intersystems.com/deepsee/library" name="Humidity02" folderName="Arduino" title="" description=""
keywords="" owner="" shared="true" public="true" locked="false" resource="" timeCreated="2015-09-18T08:20:51.172Z"
createdBy="_SYSTEM" category="" bookCover="" mdx="" cellWidth="120" columnHeaderStyle="" rowHeaderStyle=""
cellStyle="" rowLabelSpan="true" columnLabelSpan="true" cellHeight="22" showEmptyRows="false"
showEmptyColumns="false" cubeName="ARDUINO" caption="" listing="" listingRows="" showStatus="true" pageSize="100"
colorScale="" rowTotals="false" columnTotals="false" rowTotalAgg="sum" columnTotalAgg="sum" rowTotalSource="page"

```

```

showZebra="false" showRowCaption="true" printTitle="" printSubtitle="" printSubtitleOn="" showUser="" printPageSize=""
printOrientation="1" printMarginTop="" printMarginLeft="" printMarginRight="" printMarginBottom="" printLabelWidth=""
printCellWidth="" autoExecute="true" manualMode="false" userMDX="" chartMarginTop="" chartMarginLeft=""
chartMarginRight="" chartMarginBottom="" maxRows="" borderLeftCell="" borderRightCell="" borderTopCell=""
borderBottomCell="" borderLeftCol="" borderRightCol="" borderTopCol="" borderBottomCol="" borderLeftRow=""
borderRightRow="" borderTopRow="" borderBottomRow="" showFilters="" showListingFilters="" showDate=""
listingFontSize="" showZebraStripes="" filterTableStyle="" filterTableCaptionStyle="" filterTableItemStyle=""
nowDisplayFormat="" measureLocation="columns" backgroundImage="" backgroundOpacity=".12">
  <rowAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false" filterExpression=""
orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction="" aggFunctionParm=""
levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false" levelStyle=""
levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
    </rowAxisOptions>
    <columnAxisOptions spec="" key="" value="" text="" headEnabled="false" headCount="" filterEnabled="false"
filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false" aggFunction=""
aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0" advanced="false"
levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
        </columnAxisOptions>
        <rowLevel spec="[Время измерения].[H1].[DateTime].Members" key="" value="" text="Дата" headEnabled="false"
headCount="" filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC"
aggEnabled="false" aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType=""
drillLevel="0" advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
            </rowLevel>
            <measure spec="[Measures].[Humidity]" key="" value="" text="Влажность" headEnabled="false" headCount=""
filterEnabled="false" filterExpression="" orderEnabled="false" orderExpression="" orderDirection="BDESC" aggEnabled="false"
aggFunction="" aggFunctionParm="" levelCaption="" levelFormat="" levelSummary="" levelType="" drillLevel="0"
advanced="false" levelStyle="" levelHeaderStyle="" suppress8020="false" drilldownSpec="" enabled="true">
                </measure>
            </pivot>
        </Document></Export>

```

Додаток Б
Презентація

МЕТЕОСТАНЦІЯ НА ARDUINO З ВІЗУАЛІЗАЦІЄЮ І АНАЛІЗОМ ДАНИХ ПОГОДНОЇ СТАНЦІЇ



Давиденко Олександр Васильович, студент магістратури кафедри комп'ютерних наук та інженерії

Карлашук Володимир Сергійович – науковий керівник, к.т.н, доцент кафедри комп'ютерних наук та інженерії

Додаток Б.1- Слайд 1

Більшість метеобладнання українських метеостанцій вважається застарілим, а більшість прогнозів складається завдяки закордонним метеоцентрам та їх обладнанню і програмному забезпеченню з швидкою візуалізацією даних по мережі. Точність прогнозу погоди залишається необхідним в більшій конкретизації по регіонах, а також попереджатиме про екстремальні погодні явища, таких, як повінь, злива, посуха, ураган та інші.



Своєчасний прогноз завдяки компактним метеостанціям допоможе в розвитку сільського господарства в цілому. Іноді регіональні прогнози в корені відрізняються від прогнозу на місцевості, а від точності прогнозу залежить час посадки і збору врожаю. Врожайність багатьох культур залежить не тільки від кількості опадів, але і від часу їх випадання, періодичності. Особливо актуально це для районів, де клімат посушливий.

Додаток Б.1- Слайд 2

У першому розділі розглянуто проблеми сільського господарства через вплив погоди та роль метеостанції у вирішенні цієї проблеми, її можливості, приведена таблиця порівняння аналогів станцій та методи її побудови.

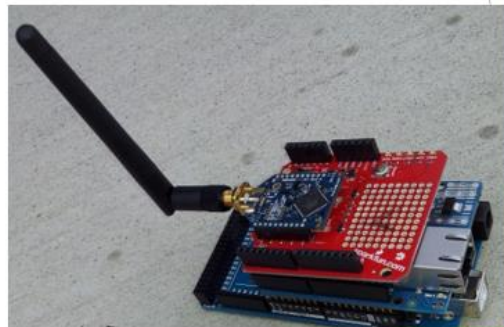
Метеостанція	Нива 1+	National Geographic 3-View black	La Crosse WS1650
Ціна	Від 1500 грн.	Від 2600 грн.	Від 5000 грн.
Апаратне забезпечення комплексу	Середня	Середня	Висока
Програмне забезпечення комплексу	Слабке	Середня	Висока
Кількість датчиків	Від 4 на одну станцію	Від 8 на одну станцію	Від 8 на одну станцію
Зручність використання	Легка	Середня	Середня

Додаток Б.1- Слайд 3

У другому розділі приведено апаратно-програмний комплекс для побудови станції.

Використовується плата Arduino з підключеними до неї для початку роботи фоторезистором і датчиком DHT11, який буде відповідати за температуру, вологість та розрахунок точки роси. При тестуванні роботи потрібно виключити потрапляння обладнання під прямий вплив погодних умов (сонячне проміння, вологість), так як все це може негативно вплинути на роботу і аналіз даних прогнозу погоди та відображення його точності. Перелік необхідного обладнання для розроблення наступний:

- Arduino Uno з Weather shield і датчиками погоди;
- 2x XBee Shield і 2x модулів XBee з розширеним діапазоном від Digi International
- Arduino Mega з Ethernet Shield;
- 2 комплекти для монтажу Arduino;
- 2x 5В блок живлення Arduino.



Додаток Б.1- Слайд 4

Датчиків та компонентів

Барометр BMP280



Цей датчик є покращеною версією датчика BMP180 і відрізняється від нього меншими розмірами, зниженим енергоспоживанням, високою точністю роботи і наявністю точної заводського калібрування

Датчик DHT11



Цифровий датчик температури і вологості, що дозволяє калібрувати цифровий сигнал на виході. Складається з ємнісного датчика вологості і термістора.

Weather Shield



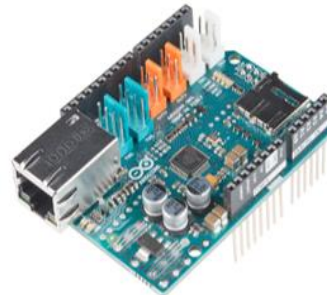
Простий у використанні щит Arduino, який надає вам доступ до атмосферного тиску, відносної вологості, яскравості і температури. На цьому екрані також є підключення до додаткових датчиків, таким як швидкість вітру, напрямок, дощомір.

Додаток Б.1- Слайд 5

Датчиків та компонентів

Модуль XBee Explorer

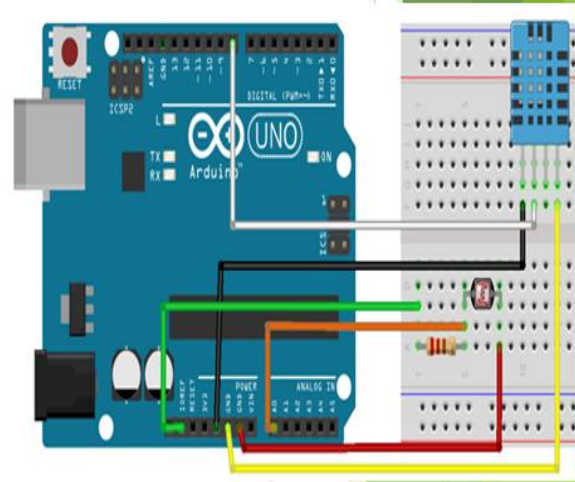
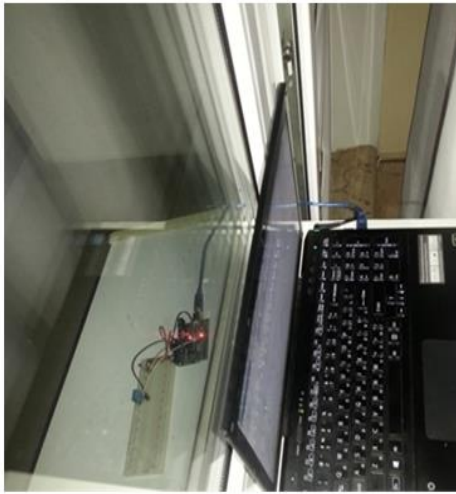
За допомогою ключа XBee Explorer можемо підключити пристрій безпосередньо до USB-порту і використовувати його в якості шлюзу між вашим комп'ютером і XBee.

Модуль Arduino Mega

Дозволяє платі Arduino підключатися до Інтернету. Він заснований на чіпі Wiznet W5500 Ethernet. Wiznet W5500 надає мережевий стек (IP), що підтримує як TCP, так і UDP.

Додаток Б.1- Слайд 6

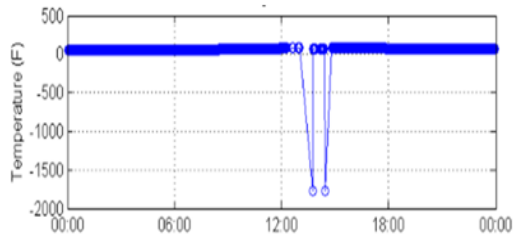
У третьому розділі приведено налаштування програмного комплексу станції з підключенням до плати Arduino датчиків вимірювання температури, вологи та сонячного світла. Підключені компоненти для збору та передачі даних з самої станції на клієнт-сервер з подальшою корекцією даних та їх візуалізацією.



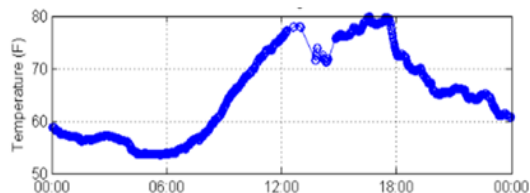
Додаток Б.1- Слайд 7

Застосування медіанної фільтрації для не коректних даних

До фільтрації



Після фільтрації



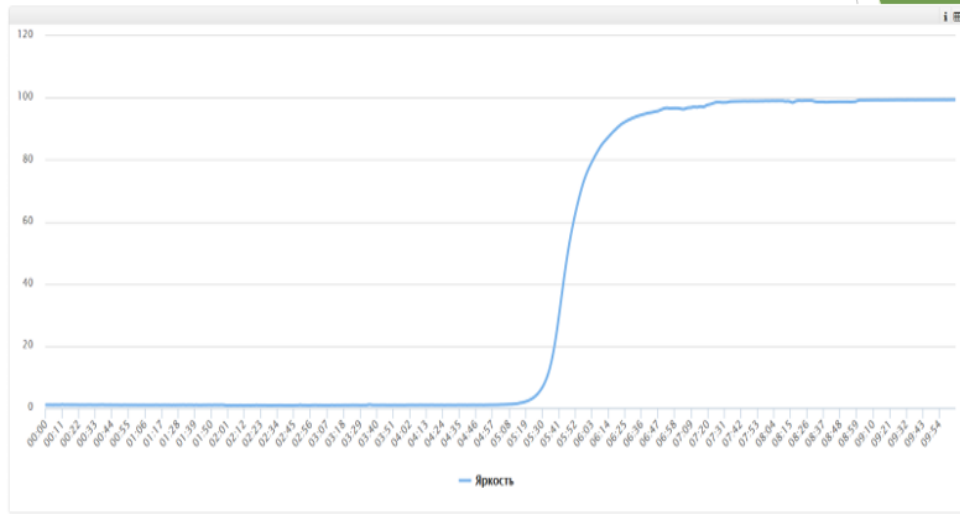
Медіанна фільтрація - достатньо часто застосовується метод попередньої обробки сигналів. Специфічною особливістю медіанних фільтрів є вибірковість по відношенню до елементів масиву, які представляють собою немонотонну складову послідовності чисел в межах вікна (апертури) фільтра, і різко виділяються на тлі сусідніх значень.

Медіанний фільтр являє собою віконний фільтр, послідовно ковзає по масиву сигналу і повертає на кожному кроці один з елементів, що потрапили у вікно (апертуру) фільтра. Вихідний сигнал y_k змінного медіанного фільтра шириною $2n+1$ для поточного відрітку k формується з вхідного часового ряду $\dots, x_{k-1}, x_k, x_{k+1}, \dots$ відповідно до формули:

$$y_k = \text{med}(x_{k-n}, x_{k-n+1}, \dots, x_{k-1}, x_k, x_{k+1}, \dots, x_{k+n-1}, x_{k+n})$$

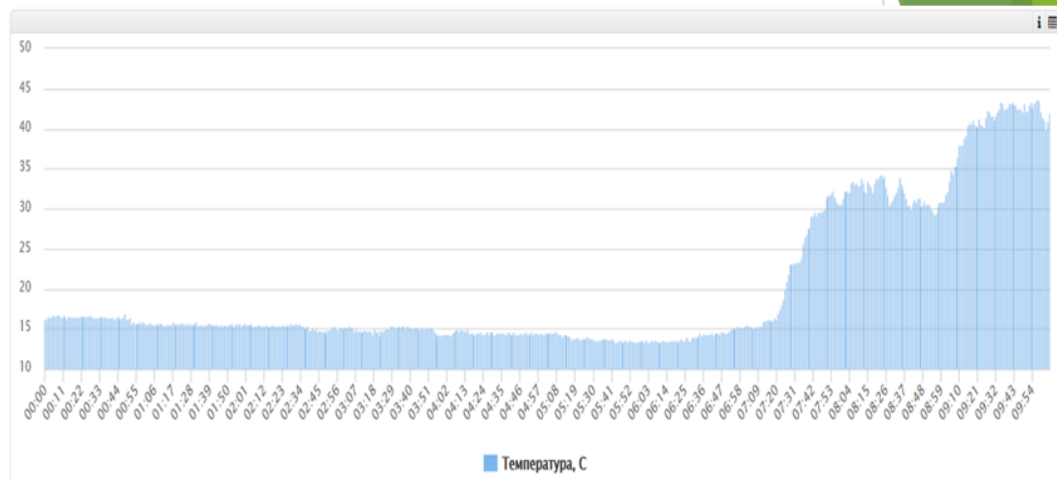
Додаток Б.1- Слайд 8

Вікно візуалізації даних складається з трьох вікон, таких як:
 - Графік яскравості. Відповідає за схід і захід сонця (світанок в районі 5:50)
 Збір даних відбувається завдяки фоторезистору.



Додаток Б.1- Слайд 9

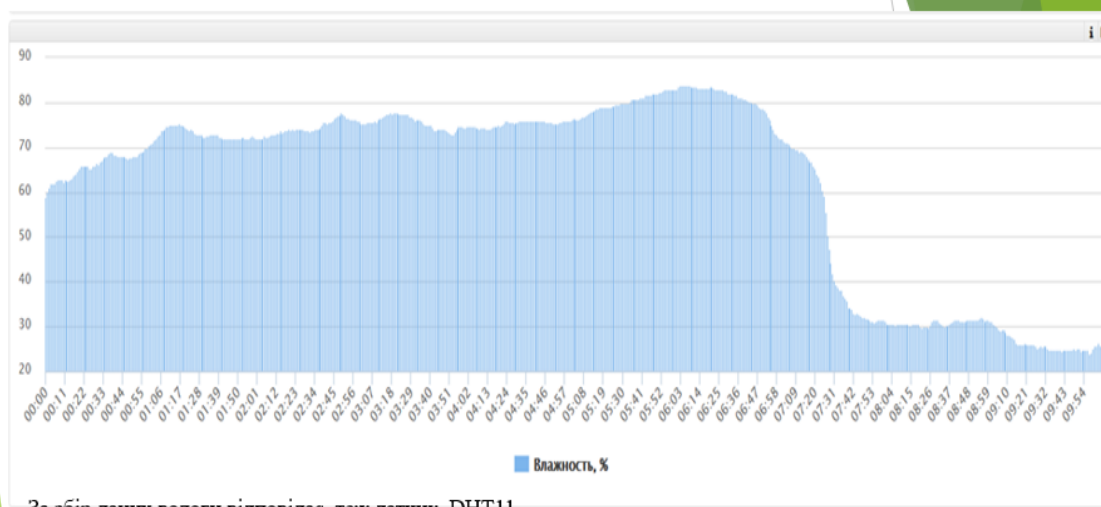
- Графік температури



За збір даних температури відповідає датчик DHT11

Додаток Б.1- Слайд 10

– Графік вологості



Додаток Б.1- Слайд 11

На сьогодні від розвитку метеобладнання і моніторингу погоди залежить сільськогосподарська діяльність. В період збору та обробки інформації, отриманні данні виводяться в трьох окремих вікнах - яскравість, температура та вологість з графіком роботи. В перспективі розвитку можна додати можливість вимірювання атмосферного тиску, напрямку вітру, покращення роботи програмного та апаратного забезпечення. Розвиток компактних погодних метеостанцій з видачою інформації прогнозу погоди на невеликій території, простота використання у польових умовах допоможе аграріям в плануванні їхньої роботи.



Додаток Б.1- Слайд 12

Додаток В

