

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

Другий (магістерський)
(рівень вищої освіти)

Розроблення інтегрованої системи комутації прикладними пристроями в
автоматизованих системах управління
(тема)

Виконав:
студент 2 курсу, групи КІТПВм-20-1

Гіль А.А.

(прізвище, ініціали)

Спеціальності 151 Автоматизація та
комп'ютерно-інтегровані технології

(код і повна назва спеціальності)

Тип програми Освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерно-інтегровані
технології процеси і виробництво

(повна назва освітньої програми)

Керівник проф. каф. КІТАМ Филипенко О.І.

(посада, прізвище, ініціали)

Допускається до захисту
Зав. кафедри КІТАМ

(підпис)

Невлюдов І. Ш.

(прізвище, ініціали)

2021р.

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет _____ АКТ _____
Кафедра _____ КІТАМ _____
Рівень вищої освіти _____ другий (магістерський) _____
Спеціальність _____ 151 Автоматизація та комп'ютерно-інтегровані технології _____
Тип програми _____ Освітньо-професійна _____
Освітня програма _____ Комп'ютерно-інтегровані технології процеси і виробництво _____
(шифр і назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри КІТАМ _____
(підпис)
«___» _____ 2021р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _____ Гіль Анастасії Андріївні _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Розроблення _____ інтегрованої _____ системи _____ комутації _____
прикладними пристроями в автоматизованих системах управління _____

Затверджена наказом по університету від _____ № 1697 Ст від 08.11.2021 _____

2. Термін подання студентом роботи до екзаменаційної комісії _____ 17.12.21 _____

3. Вихідні дані до роботи
Модуль комутації цифрових інтерфейсів зв'язку, мікросхеми пам'яті SARAM і FRAM, мікроконтролер фірми STM32, PHY мікросхеми для Ethernet і CAN, перетворювачі інтерфейсів для RS485 і RS422, мова програмування C\C++.

4. Перелік питань, що потрібно опрацювати в роботі
Аналіз технічного завдання.

1. Аналіз предметної області.
 2. Обґрунтування та розробка інтегрованої системи комутації прикладними пристроями в автоматизованих системах управління.
 3. Моделювання роботи модулю комутації.
 4. Розробка технічного забезпечення інтегрованої системи комутації.
 5. Розробка алгоритмічно-програмного забезпечення інтегрованої системи комутації.
- Висновки.
Додатки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій

Слайди: Комп'ютерна презентація у форматі *.ppt

6. Консультанти розділів работ

Найменування Розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	проф. каф. КІТАМ Филипенко О.І.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз технічного завдання	03.09.21	Виконано
2	Аналіз предметної області	10.09.21	Виконано
3	Обґрунтування та розробка інтегрованої системи комутації прикладними пристроями в автоматизованих системах управління	21.10.21	Виконано
4	Моделювання роботи модулю комутації	08.11.21	Виконано
5	Розробка технічного забезпечення інтегрованої системи комутації	15.11.21	Виконано
6	Розробка алгоритмічно-програмного забезпечення інтегрованої системи комутації	22.11.21	Виконано
7	Формування висновків	29.11.21	Виконано
8	Оформлення пояснювальної записки	01.12.21	Виконано
9	Подання роботи на перевірку Інтернет сервісом Unichек	09.12.21	Виконано
11	Подання роботи на рецензію	09.12.21	Виконано
12	Подання роботи на підпис зав. Кафедри	17.12.21	Виконано
13	Подання кваліфікаційної роботи ЕК	17.12.21	Виконано
14			

Дата видачі завдання 08.11.2021

Студент _____
(підпис)

Гіль А.А.
(прізвище, ініціали)

Керівник роботи _____
(підпис)

проф. каф. КІТАМ Филипенко О.І.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 121 с., 4 табл., 40 рис., 2 дод., 45 джерела.

МІКРОКОНТРОЛЕРИ, АСУ, ІНТЕРФЕЙСИ ЗВ'ЯЗКУ, ВБУДОВУВАНІ СИСТЕМИ, ПРОТОКОЛИ ОБМІНУ, ЦИФРОВІ КОМУТАТОРИ.

Об'єкт дослідження – системи автоматизованого управління технологічними процесами.

Предмет дослідження – інтегровані системи комутації та топології побудови промислових мереж зв'язку в системах автоматизованого управління.

Мета роботи – підвищення якості роботи інтегрованої системи комутації та оптимізація процедури зв'язку між прикладними приладами системи, за рахунок зменшення кількості точок комутації сигналів шляхом розробки модулю комутації цифрових та аналогових інтерфейсів зв'язку.

Методи дослідження – аналіз наявних інтерфейсів та топології зв'язку, порівняння переваг та недоліків наявних методів комутації сигналів, моделювання та формалізація отриманих результатів.

Розглянуто системи, топології та алгоритми комутації цифрових інтерфейсів зв'язку. В результаті проведених досліджень розроблений концепт та проведено моделювання і формалізація отриманих результатів. Обрані ключові елементи модулю на базі яких побудована архітектура програмного забезпечення.

Впровадження може бути можливим в автоматизованих системах управління виробництвом. Також робота може бути корисна фахівцям галузі автоматизації та при навчанні студентів спеціальності 151-Автоматизація та комп'ютерно-інтегровані технології.

ABSTRACT

Explanatory note: 121 pages, 4 tables, 40 figures, 2 appendices, 45 sources.

**MICROCONTROLLERS, ACS, COMMUNICATION INTERFACES,
BUILT-IN SYSTEMS, EXCHANGE PROTOCOLS, DIGITAL SWITCHES.**

Object of research – automated process control systems.

The subject of research – integrated switching systems and topologies for the construction of industrial communication networks in automated control systems.

The purpose of the work – to improve the quality of the integrated switching system and optimize the communication procedure between the application devices of the system, by reducing the number of switching points by developing a switching module for digital and analog communication interfaces.

Research methods – analysis of existing interfaces and communication topology, comparison of advantages and disadvantages of existing signal switching methods, modeling and formalization of the obtained results.

Systems, topologies and algorithms for switching digital communication interfaces are considered. As a result of the conducted researches the concept is developed and modeling and formalization of the received results is carried out. The key elements of the module on the basis of which the software architecture is built are selected.

Implementation may be possible in automated production management systems. The work can also be useful for specialists in the field of automation and in teaching students majoring in 151-automation and computer-integrated technologies.

ЗМІСТ

Перелік скорочень	8
Вступ.....	9
Розділ 1 Аналіз предметної області	12
1.1 Ключові особливості промислових мереж.....	12
1.2 Типи промислових інтерфейсів зв'язку	14
1.3 Типи протоколів обміну в промислових мережах.....	22
1.4 Аналіз аналогічних рішень за подібним призначенням.....	30
1.5 Висновки до розділу 1	33
Розділ 2 Обґрунтування та розробка інтегрованої системи комутації прикладними пристроями в автоматизованих системах управління.....	35
2.1 Інтегрована система комутації прикладними пристроями в автоматизованих системах управління.....	35
2.2 Основна концепція модулю, що розробляється та досліджується	37
2.3 Принцип роботи з аналоговими та дискретними сигналами	38
2.4 Методи реалізації цифрових інтерфейсів зв'язку	41
2.5 Центральний керуючий контролер	45
2.6 Інтерфейс оператора	47
2.7 Висновки до розділу 2	48
Розділ 3 Моделювання роботи модулю комутації	49
3.1 Побудова моделі роботи модуля	50
3.2 Розрахунок основних параметрів моделі.....	52
3.2.1 Параметри обміну за інтерфейсом SPI	53
3.2.2 Параметри обміну за інтерфейсом I2C	55
3.2.3 Параметри обміну за інтерфейсом UART	57
3.2.4 Параметри обміну за інтерфейсом CAN.....	59
3.2.5 Параметри обміну за інтерфейсом Ethernet	61
3.3 Моделювання роботи потоків.....	62

3.3.1	Моделювання обміну за інтерфейсами по 32 байти.....	64
3.3.2	Моделювання обміну за інтерфейсами по 128 байт.....	68
3.3.3	Моделювання обміну за інтерфейсами по 512 байт.....	70
3.4	Висновки до розділу 3	73
Розділ 4 Розробка технічного забезпечення інтегрованої системи комутації		75
4.1	Вибір мікроконтролера.....	75
4.2	Вибір мікросхем для роботи з дискретними сигналами	80
4.3	Вибір мікросхем РНУ для Ethernet та CAN	81
4.4	Вибір мікросхем для роботи інтерфейсів RS-422/RS-485	82
4.5	Вибір мікросхем пам'яті	83
4.6	Висновки до розділу 4	86
Розділ 5 Розробка алгоритмічно-програмного забезпечення інтегрованої системи комутації.....		88
5.1	Структура зберігання даних підключених пристроїв	88
5.2	Алгоритм роботи потоку формування стека запитів	90
5.3	Алгоритм роботи потоку обміну із пристроями.....	92
5.4	Принцип роботи потоку опитування внутрішніх джерел даних	93
5.5	Принцип роботи потоку збору отриманих даних	93
5.6	Охорона праці.....	94
5.7	Висновки до розділу 5	97
Висновки		98
Перелік джерел посилання		99
Додаток А Лістинг програми керування модулем		104
Додаток Б Демонстраційний матеріал.....		109

ПЕРЕЛІК СКОРОЧЕНЬ

- АСУ – автоматизована система управління;
- АСУ ТП – автоматизована система управління технологічними процесами;
- САУ – системи автоматичного управління;
- CAN (Controller Area Network) – контролер мережевого зв'язку;
- CD (Collision Detection) – детектор колізій на лінії обміну;
- CS (Carrier Sense) – детектор доступу до лінії зв'язку;
- MA (Multiple Access) – багатоканальний доступ;
- MDI (Medium Dependent Interface) – середньо залежний інтерфейс;
- MII (Media Independent Interface) – медіа-незалежний інтерфейс;
- PCS (Physical Coding Sublayer) – підрівень фізичного кодування;
- PDU (Protocol Data Unit) – блок даних протоколу;
- PMA (Physical Medium Attachment) – середня фізична прихильність;
- RTU (Remote Terminal Unit) – віддалений термінальний блок;
- SPI (Serial Peripheral Bus) – послідовна периферійна шина;
- UART (Universal Asynchronous Receiver-Transmitter) – універсальний асинхронний приймач-передавач.

ВСТУП

Автоматизована система управління або АСУ – комплекс апаратних і програмних засобів, призначений для управління різними технологічними процесами. Залежно від ролі людини в процесі управління, форм зв'язку і взаємодії ланки «людина-машина», розподілу інформаційних і керуючих функцій між оператором і засобами контролю і управління усі системи можна розділити на два класи: інформаційні системи, що забезпечують збір та видачу інформації та управляючі системи, які забезпечують порядок зі збором інформації видачу команд виконавцям чи виконавчим механізмам [1].

Складовими частинами АСУ можуть бути окремі системи автоматичного управління і автоматизовані пристрої, які зв'язуються в один єдиний комплекс. Для інформаційного зв'язку всіх підсистем, використовують промислові інтерфейси зв'язку. Найбільш важливими параметрами інтерфейсу зв'язку є пропускна здатність і максимальна довжина кабелю.

Промислові інтерфейси зазвичай забезпечують гальванічну розв'язку між підключеними пристроями. Найбільш поширені в промислової автоматизації послідовні інтерфейси: RS-485, RS-232, RS-422, Ethernet, CAN, HART, AS-інтерфейс [2].

Для обміну інформацією пристрої повинні мати однаковий протокол обміну. У простій формі протокол – це набір правил, які керують обміном інформацією. Він визначає синтаксис і семантику повідомлень, операції управління, синхронізацію і стан при комунікації. Основними протоколами зв'язку в даний момент є: https, SMTP, FTP, SSH, Modbus, Modbus TCP, PROFIBUS [3].

Маючи великий стек протоколів, об'єднання різних пристроїв управління в одну централізовану систему не викликає складнощів. Набагато складніше підключити до системи пристрої, які відрізняються протоколом обміну і інтерфейсом зв'язку. У такому випадку використовуються комутатори

інтерфейсів або «хаби», основна мета яких забезпечення зв'язку між підключеними пристроями і системою управління по одному каналу.

Актуальність питань, що висвітлюються в кваліфікаційній роботі обумовлена тим, що розроблювальний пристрій забезпечує більш гнучку систему підключення і комутації різних протоколів і інтерфейсів за рахунок архітектури та спеціальної структури обміну між комутатором і майстер-пристроєм.

Об'єкт дослідження – системи автоматизованого управління технологічними процесами.

Предмет дослідження – інтегровані системи комутації та топології побудови промислових мереж зв'язку в системах автоматизованого управління.

Мета роботи – підвищення якості роботи інтегрованої системи комутації та оптимізація процедури зв'язку між прикладними приладами системи, за рахунок зменшення кількості точок комутації сигналів шляхом розробки модулю комутації цифрових та аналогових інтерфейсів зв'язку.

Під підвищенням якості розуміється зменшення кількості точок відмови системи за рахунок використання модуля, який гарантує задану дискретизацію обміну та стовідсоткову доставку інформації від підключених прикладних пристроїв до централізованої системи керування, за рахунок використання новітніх технологій та більш гнучкої архітектури програмного забезпечення.

Методи дослідження – аналіз наявних інтерфейсів та топології зв'язку, порівняння переваг та недоліків наявних методів комутації сигналів, моделювання та формалізація отриманих результатів.

Для розв'язання науково-прикладної задачі, в межах магістерської кваліфікаційної роботи вирішено такі задачі:

- аналіз основних топологій мереж та аналогічних рішень побудови промислових мереж зв'язку;
- аналіз основних інтерфейсів та засобів зв'язку у промислових мережах автоматизації підприємств;
- моделювання процесів комутацій інтерфейсів зв'язку;

- отримання вимог до апаратної частини модулю;
- розробка архітектури програмного забезпечення та структури зберігання інформації отриманої від прикладних приладів.

Кваліфікаційна робота виконана відповідно до стандарту ДСТУ 3008-2015 [4] і з урахуванням матеріалів навчального посібника [5-6] і методичних вказівок [7].

Результати кваліфікаційної роботи пройшли апробацію на міжнародних конференціях [8] та результати викладено у статтях [9].

Впровадження інтегрованої системи комутації прикладними пристроями для автоматизованих систем управління може бути можливим при навчанні студентів спеціальності 151-Автоматизація та комп'ютерно-інтегровані технології. Також робота може бути корисна фахівцям галузі автоматизації.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Обмін інформацією між пристроями, що входять до складу автоматизованої системи управління (комп'ютерами, контролерами, давачами, виконавчими пристроями), відбувається за допомогою промислових мереж. Промислові мережі відрізняються від офісної високою надійністю, спеціальним конструктивним виконанням, широким температурним діапазоном роботи, детермінованістю обміну і можливістю резервування.

В даний час налічується більше 50 типів промислових мереж (ModBus, Profibus, DeviceNet, CANopen, Ethernet, Foundation Fieldbus, Interbus, BitBus і ін.). Однак в переважній більшості АСУ та АСУ ТП використовують мережі ModBus і Profibus. В останні роки зріс інтерес до мереж на основі CANopen і DeviceNet [10].

1.1 Ключові особливості промислових мереж

Промисловою мережею називають комплекс обладнання і програмного забезпечення, який забезпечує обмін інформацією між кількома пристроями. Промислова мережа є основою для побудови розподілених систем збору даних і управління.

Оскільки в промисловій автоматизації мережеві інтерфейси можуть бути невід'ємною частиною пристрою, а мережеве програмне забезпечення прикладного рівня моделі OSI виконується на основному процесорі промислового контролера, то відокремити мережеву частину від пристроїв, що об'єднуються в мережу, фізично неможливо.

З'єднання промислової мережі з її компонентами виконується за допомогою інтерфейсів. Інтерфейсом називають логічну або фізичну межу між пристроєм і середовищем передачі інформації. Зазвичай цією межею є набір

електронних компонентів і пов'язаного з ними програмного забезпечення.

Взаємодія пристроїв в промислових мережах виконується відповідно до моделей клієнт-сервер або майстер-підлеглий. У моделі клієнт-сервер взаємодіє кілька об'єктів, сервером є об'єкт який займається збором інформації і формуванням команд, а клієнтом є об'єкт, що надає цю інформацію. У моделі взаємодії майстер-підлеглий є один майстер і безліч підлеглих пристроїв.

Підлегли пристрої очікують запиту від майстра, виконують його команди і формують відповідь. На відміну від моделі клієнт-сервер в моделі майстер-підлеглий, ведений пристрій ніколи не починає комунікацію першим, він чекає запиту від ведучого і відповідає на нього.

Передача інформації в мережі виконується через канал між передавальним і приймальним пристроєм. Канал є поняттям теорії інформації і включає в себе лінію зв'язку і приймально-передавальні пристрої. У загальному випадку замість терміну "лінія зв'язку" використовують термін "середовище передачі", в якості якої може виступати, наприклад, оптоволокно, ефір або кручена пара. У розподілених системах на основі промислових мереж є п'ять типів даних:

- сигнали, це результати вимірювань, отримані від давачів і вимірювальних приладів;
- команди, це повідомлення, які викликають деякі дії, наприклад, включення під-регулятора;
- стан, показує поточний або майбутній стан системи, в який вона повинна перейти;
- подія, яка зазвичай, настає при досягненні поточним параметром граничного значення. наприклад, подією може бути вихід температури за технологічно допустиму межу. за появою події повинні слідувати відповідні дії, тому для подій особливо важлива вимога гарантованого часу доставки;
- запит, це команда, що посилається для того, щоб отримати інформацію про стан пристрою або сигналу.

Мережі можуть мати топологію зірки, кільця, шини або змішану. «Зірка»

в промислової автоматизації практично не використовується. «Кільце» використовується в основному для передачі маркера в багатомасштабних мережах. «Шинна» топологія є загальноприйнятою, що є однією з причин застосування терміна «промислова шина» замість «промислова мережа». До загальної шини в різних місцях може бути підключено будь-яку кількість пристроїв [11].

Основними параметрами промислових мереж є продуктивність і надійність. Продуктивність мережі характеризується часом реакції і пропускнуною спроможністю.

Важливою характеристикою промислових мереж є надійність доставки даних. Надійність характеризується коефіцієнтом готовності, ймовірністю доставки даних, передбачуваністю часу доставки, безпекою та стійкістю до відмов.

1.2 Типи промислових інтерфейсів зв'язку

На даному етапі розвитку науки і техніки найбільш поширені в промислової автоматизації послідовні інтерфейси зв'язку: RS-232, RS-422, RS-485, Ethernet і CAN.

Стандарт RS-232 більш відомий як звичайний COM порт комп'ютера або послідовний порт. Інтерфейси RS-422 і RS-485 широко застосовуються в промисловості для з'єднання різного обладнання.

Інтерфейс RS-232 призначений для організації прийому-передачі даних між терміналом і комунікаційним обладнанням за схемою точка-точка. На рисунку 1.1 наведена стандартна схема підключення сигналів між терміналом і комунікаційним обладнанням.

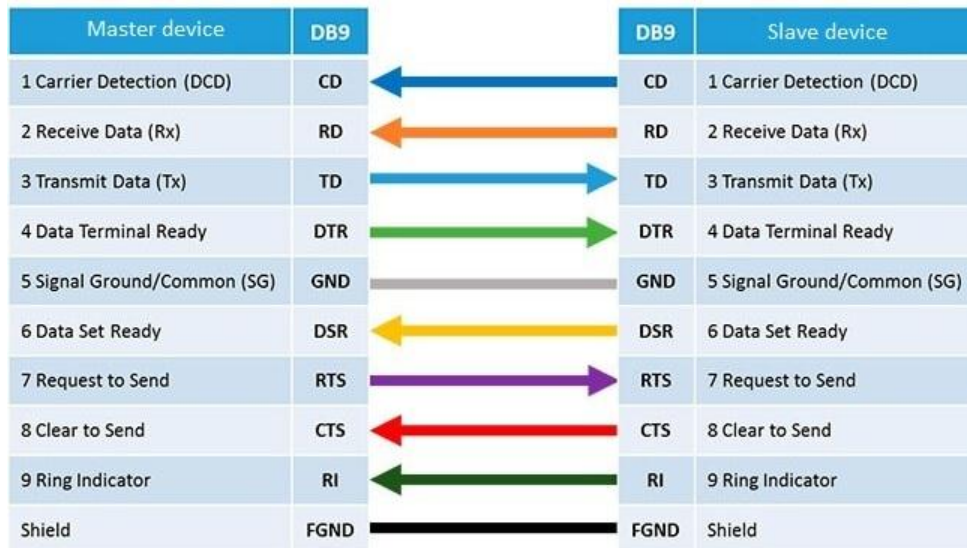


Рисунок 1.1 – Схема підключення пристроїв по стандарту RS232 [12]

Швидкість роботи RS-232 сильно залежить від відстані між пристроями, зазвичай на відстані 15 метрів швидкість дорівнює 9600 біт/с. На мінімальній відстані швидкість зазвичай дорівнює 115,2 кбіт/с, але є обладнання, яке підтримує швидкість до 921,6 кбіт/с.

Інтерфейс RS-232 працює в дуплексному режимі, що дозволяє передавати і приймати інформацію одночасно. Інформація по інтерфейсу RS-232 передається в цифровому вигляді логічними 0 і 1. Логічній «1» відповідає напруга в діапазоні від -3 до -15 В. Логічному «0» відповідає напруга в діапазоні від +3 до +15 В.

Інтерфейс RS-422 схожий на RS-232, тому що дозволяє одночасно відправляти і приймати повідомлення по окремих лініях (повний дуплекс), але використовує для цього диференційний сигнал, тобто різницю потенціалів між провідниками А і В.

Швидкість передачі даних в RS-422 так само залежить від відстані і може змінюватися в межах від 10 кбіт/с (1200 метрів) до 10 Мбіт/с (10 метрів). У мережі RS-422 може бути тільки один передавальний пристрій і до 10 приймаючих пристроїв. Лінія RS-422 (рисунок 1.2) являє собою 4 дроти для прийому/передачі даних і один загальний провід землі GND.

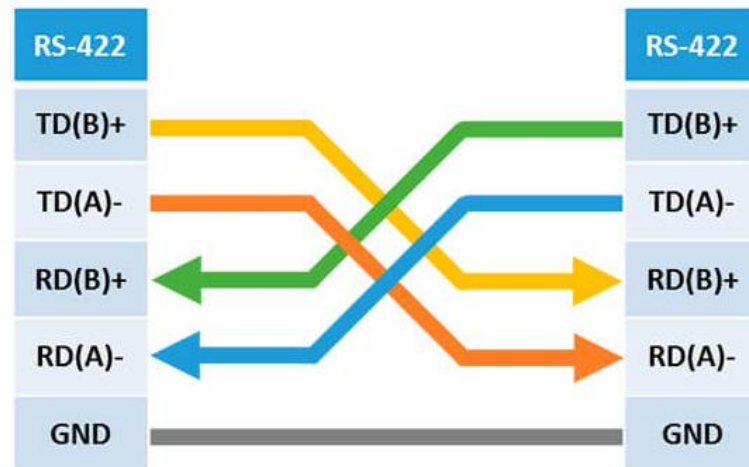


Рисунок 1.2 – Схема підключення пристроїв по стандарту RS-422 [12]

Скручування дротів (кручена пари) між собою дозволяє позбутися від наведень і перешкод, тому як наводка однаково діє на обидва дроти, а інформація витягується з різниці потенціалів між провідниками А і В однієї лінії.

Напруга на лініях передачі даних може знаходитися в діапазоні від -6 В до +6 В. Логічному 0 відповідає різниця між А і В більше +0,2 В. Логічною 1 відповідає різниця між А і В менше -0,2 В. Стандарт RS-422 не визначає конкретний тип роз'єму, зазвичай це може бути клемна колодка або роз'єм DB9 але так як відстань між приймачем і передавачем в мережі RS-422 може досягати 1200 метрів, то для запобігання відображення сигналу від кінця лінії ставиться спеціальний 120 Ом узгоджувальний резистор або "термінатор". Цей резистор встановлюється між RX + і RX- контактами на початку і в кінці лінії.

Але не дивлячись на простоту інтерфейсів RS-232 і RS-422 в промисловості найчастіше використовується інтерфейс RS-485, так як він дозволяє підключити декілька приймачів і передавачів на одну шину.

Інтерфейс RS-485 схожий на RS-422 тим, що також використовує диференційний сигнал для передачі даних. Існує два типи підключень по інтерфейсу RS-485, з двома і чотирма контактними лініями.

В одному сегменті мережі RS-485 може бути до 32 пристроїв, але за допомогою додаткових повторювачів і підсилювачів сигналів до 256 пристроїв.

Мережа RS-485 має на увазі що в один момент часу активним може бути тільки один передавач. Приклад сегменту мережі RS-485 наведено на рисунку 1.3.

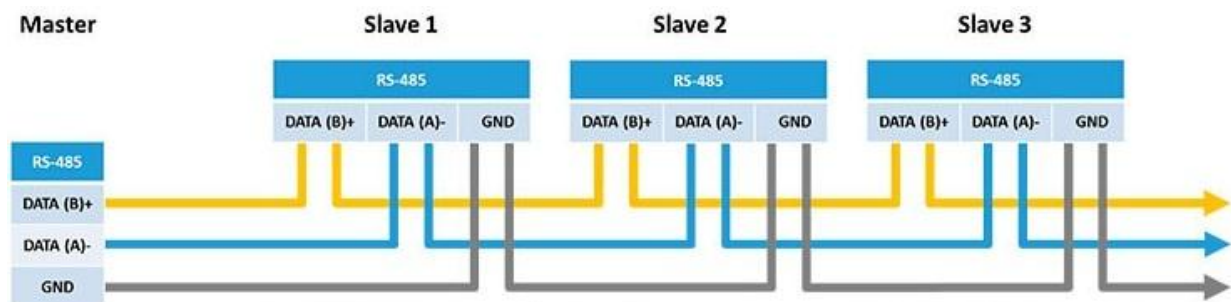


Рисунок 1.3 – Підключення пристроїв по інтерфейсу RS-485 [12]

Швидкість роботи інтерфейсу RS-485 також залежить від довжини лінії. Напруга на лініях знаходиться в діапазоні від -7 В до +12 В. Як і стандарт RS-422, стандарт RS-485 не визначає конкретний тип роз'єму, і для узгодження ліній на великих відстанях в RS-485 також ставлять погоджують резистори на початку і в кінці лінії [12].

В даний час крім застосування стандартизованих інтерфейсів зв'язку сімейства RS-485, особливу популярність отримав стандарт Ethernet, промислового застосування якого довгий час заважав метод випадкового доступу до мережі, який не гарантував доставку повідомлення в короткий і заздалегідь відомий час. Однак ця проблема була вирішена застосуванням комутаторів. Впровадженню Ethernet в промисловість сприяли його особливості: висока швидкість передачі (до 10 Гбіт/с), відповідність вимогам жорсткого реального часу при високій швидкодії, простота інтеграції і наявність великої кількості фахівців з обслуговування Ethernet.

Надійність промислового Ethernet забезпечується резервуванням кабельних ліній і мережевих карт, а також спеціальним програмним забезпеченням. Для підвищення ступеня захисту системи управління від несанкціонованого доступу магістральну мережу Industrial Ethernet відокремлюють від офісної брандмауером. Відмінними ознаками промислового Ethernet є:

- відсутність колізій і детермінованість поведінки завдяки застосуванню комутаторів;
- індустріальні кліматичні умови;
- стійкість до вібрацій;
- підвищені вимоги до електромагнітної сумісності;
- можливість резервування.

Недоліком Ethernet є значний рівень «накладних витрат» в протоколах TCP/IP, які розраховані на пересилку даних великих обсягів. Якщо ж пристрій пересилає всього 8 байт, що типово для систем АСУ, то корисна інформація в протоколі становить всього 11 %, оскільки кожен пакет TCP/IP міститься 68 байт службової інформації.

Фізичний рівень Industrial Ethernet складається з середовища передачі і підрівнів:

- MDI для з'єднання лінії зв'язку з трансівером;
- PMA який включає в себе трансівер;
- PCS який забезпечує схему кодування, мультиплексування, синхронізації потоку символів, а також виконує вирівнювання спектра кодового сигналу, де мультиплексування, декодування;
- підрівня, який забезпечує узгодження інтерфейсу МПІ з MAC-рівнем.

При низькій швидкості обміну (10 Мбіт/с) використовується манчестерський код, при якому логічна одиниця кодується переходом сигналу з низького рівня на високий, а логічний нуль – переходом з високого рівня на низький. Недоліком манчестерського коду є широка смуга частотного спектра, пов'язана з необхідністю перемикавання рівнів сигналу при надходженні кожної двійкової цифри.

При переході до високих швидкостей обміну 100 Мбіт/с і вище, для зменшення перешкод, випромінюваних лінією передачі, використовується NRZI-кодування, при якому двійковій одиниці відповідає зміна рівня NRZI-сигналу, а нулю – її відсутність. Високий або низький рівень NRZI-сигналу

встановлюється в залежності від того, який рівень був до нього. Логічного одиниці відповідає зміна сигналу незалежно від того, який рівень був присутній на шині раніше (рисунок 1.4). Рівень не змінюється, якщо потрібно закодувати логічний нуль. Інакше кажучи, логічна одиниця при NRZI-кодуванні інвертує попереднє логічне стан, тому в назві коду присутнє слово «Inverted».

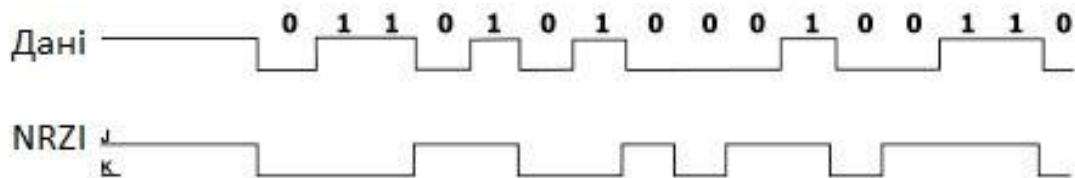


Рисунок 1.4 – Приклад кодування NRZI [13]

Структурна схема блоку РНУ для Industrial Ethernet наведено на рисунку 1.5. Блок складається з приймального і передавального каналу, які через ізолюючі трансформатори підключаються до лінії через роз'єм RJ-45. Трансформатори використовуються для ізоляції трансивера від високих синфазних напруг, які можуть з'являтися на лінії внаслідок електромагнітних і електростатичних наведень.

Блок CSMA/CD виконує прослуховування лінії, щоб визначити її зайнятість, забезпечує доступ багатьом учасникам мережі до загальної лінії передачі і виявляє колізії в лінії.

Якщо при прослуховуванні лінії пристрій виявляє, що під час роботи його передавача в лінії є дані від інших пристроїв, то передача припиняється і відновлюється після закінчення випадкового проміжку часу. Колізії виникають не тільки при одночасному початку передачі декількома пристроями, але і в разі, коли в мережі є велика затримка, наприклад, через велику довжину кабелю або великої кількості повторювачів. При наявності затримки можлива ситуація, коли одна станція почала передачу, але дані ще не дійшли до іншої станції, яка прослуховує лінію з метою почати передачу.

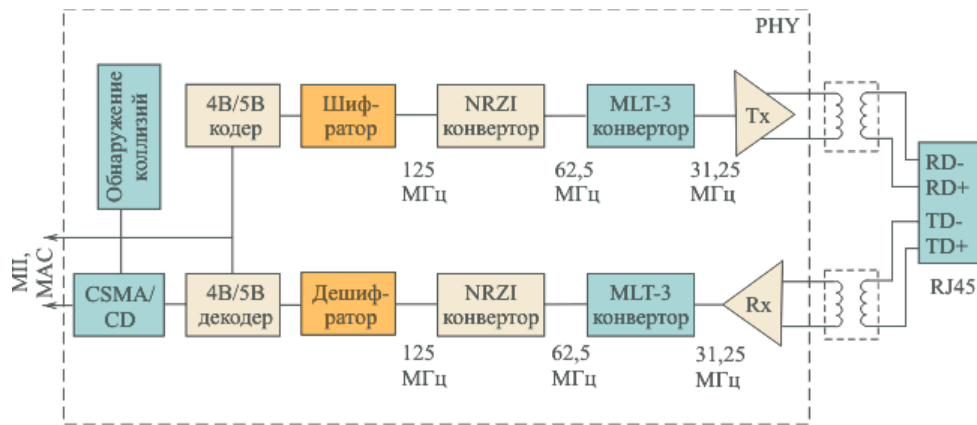


Рисунок 1.5 – Структурна схема РНУ стандарту Ethernet [13]

Для нормального розпізнавання колізії посилка повинна бути виявлена під час передачі кадру. Для цього затримка сигналу в мережі не повинна бути занадто великою. Так, при швидкості передачі 100 Мбіт/с довжина кабелю не повинні перевищувати 100 м і між будь-якими двома станціями повинно бути не більше чотирьох концентраторів.

Причина полягає в тому, що при виникненні колізій навантаженість мережі зростає внаслідок спроб повторної передачі одних і тих же кадрів. Це і стало основною причиною, по якій мережу Ethernet довгий час не знаходила застосування в промисловій автоматизації. Цю проблему вдалося вирішити тільки за допомогою комутаторів.

Комутатор представляє собою інтелектуальний багато портовий пристрій, який встановлює фізичний зв'язок між двома Ethernet-пристроями. При цьому інші пристрої мережі виявляються не підключені до каналу передачі. Таким чином, замість усупільненого каналу передачі комутатор дозволяє отримати кілька незалежних один від одного каналів між парами пристроїв.

В такому випадку мережа, виявляється, розділена на незалежні сегменти, в кожному з яких може відбуватися обмін інформацією незалежно від інших. Це призводить до збільшення пропускної здатності мережі, зниження коефіцієнта навантаженості, а також до усунення колізій [13].

Для виконання завдання сегментування мережі комутатор містить таблицю MAC-адресів мережевих пристроїв. Таблиця адресів може бути

занесена в комутатор при його налаштуванні або згенерована комутатором автоматично.

Оскільки пропускна здатність вихідного порту комутатора обмежена, вхідний трафік, виявляється, в кілька разів більше вихідного, а буферна пам'ять комутатора обмежена. Це може привести до втрати даних, якщо не взяти заходів для придушення вхідного трафіку. Керування трафіком виконується за допомогою спеціальної технології «Advanced Flow Control», коли мережевий адаптер, що підтримує цей стандарт, після отримання команди від комутатора «Призупинити передачу», припиняє передачу кадрів, а після команди «Відновити передачу» відновлює її [14].

Крім популярних інтерфейсів на базі RS-232 і Ethernet в системах автоматичного управління і контролю застосовують інтерфейс CAN, який являє собою комплекс стандартів для побудови розподілених промислових мереж, CAN використовує послідовну передачу даних в реальному часі з дуже високим ступенем надійності і захищеності.

CAN охоплює два рівні моделі OSI: фізичний і канальний. Стандарт не передбачає ніякого протоколу прикладного рівня моделі OSI. Тому для його втілення в життя різні фірми розробили кілька власних протоколів. CAN характеризується наступними основними властивостями:

- кожного повідомлення встановлюється свій пріоритет;
- гарантована величина паузи між двома актами обміну;
- гнучкість конфігурації і модернізації системи;
- ширококомовний прийомом даних із синхронізацією часу;
- допустимість кількох провідних пристроїв в мережі;
- здатність до виявлення помилок;
- автоматичний повтор передачі повідомлень, доставлених з помилкою, відразу, як тільки мережа стане вільною;
- автоматичне розпізнавання збоїв і відмов з можливістю автоматичного відключення модулів.

До недоліків можна віднести порівняно високу вартість CAN-пристроїв,

відсутність єдиного протоколу прикладного рівня, а також надмірну складність і заплутаність протоколів каналного і прикладного рівня, викладених в стандартах організації CAN in Automation (CiA). Саме через відсутність єдиного стандарту обміну інформацією в мережі CAN пристроїв, даний інтерфейс в основному, застосовується в автомобільній промисловості і не досить часто системах автоматизації виробництв [15].

1.3 Типи протоколів обміну в промислових мережах

Для організації промислових мереж використовується безліч протоколів передачі даних основними з них є: ModBus, PROFIBUS і CANopen. Протоколи обміну необхідні для передачі даних між давачами, контролерами і виконавчими механізмами; калібрування давачів; зв'язку нижнього і верхнього рівнів АСУ. Протоколи розробляються з урахуванням особливостей виробництва і технічних систем, забезпечуючи надійне з'єднання і високу точність передачі даних між різними пристроями.

Протокол і мережа ModBus є найпоширенішою в світі. Одним з переваг ModBus є відсутність необхідності в спеціальних інтерфейсних контролерах як у Profibus або CAN і проста його реалізація. Все це знижує витрати на освоєння стандарту як системними інтеграторами, так і розробниками обладнання.

Основним недоліком ModBus є мережевий обмін по типу провідний/ведений, що не дозволяє веденим пристроям передавати дані по мірі їх появи в зв'язку з чим потрібна висока інтенсивність опитування ведених пристроїв провідним.

Різновидами ModBus є протоколи ModBus Plus – многомастерний протокол з кільцевої передачею маркера і ModBus TCP розрахований на використання в мережах Ethernet.

Протокол ModBus має два режими передачі: Remote Terminal Unit – віддалене термінальний пристрій і ASCII. Стандарт передбачає, що режим RTU в протоколі ModBus повинен бути присутнім обов'язково, а режим ASCII є

опціонаним. Користувач може вибирати будь-який з них, але всі модулі, включені в мережу ModBus, повинні мати один і той же режим передачі.

Стандарт ModBus передбачає застосування фізичного інтерфейсу RS-485, RS-422 або RS-232. Найбільш поширеним для організації промислової мережі є 2-провідний інтерфейс RS-485.

Протокол ModBus передбачає, що тільки один провідний пристрій і до 247 ведених можуть бути об'єднані в промислову мережу, в якій обмін даними завжди ініціюється провідним.

Адреси з 1 по 247 є адресами ModBus пристроїв в мережі, а з 248 по 255 зарезервовані, в мережі ModBus не повинно бути двох пристроїв з однаковими адресами. Провідний пристрій може посилати запити всіх пристроям одночасно «широкомовний режим» або тільки одному. Для широкомовного режиму зарезервований адрес "0".

У протоколі ModBus RTU повідомлення починає сприйматися як нове після паузи на шині тривалістю не менше 3,5 символів (14 біт) таким чином величина паузи в секундах залежить від швидкості передачі.

Формат кадру протоколу ModBus наведено на рисунку 1.6. Поле адреси завжди містить тільки адресу веденого пристрою, навіть у відповідях на команду, надіслану провідним. Завдяки цьому ведучий пристрій знає, від якого модуля прийшла відповідь.

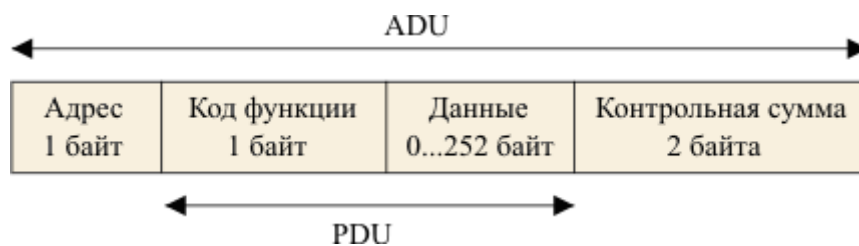


Рисунок 1.6 – Формат кадру протоколу ModBus [16]

Поле «Код функції» говорить модулю про те, яку дію потрібно виконати. Поле «Дані» може містити довільну кількість байт. У ньому може міститися інформація про параметри, які використовуються в запитах контролера або

відповідях модуля. Поле «Контрольна сума» містить контрольну суму CRC довжиною 2 байта.

У режимі RTU дані передаються молодшими розрядами вперед. За замовчуванням в RTU режимі біт паритету встановлюють рівним 1, якщо кількість двійкових одиниць в байті непарна, і рівним 0, якщо вона парна. Такий паритет називають парних (even parity) і метод контролю називають контролем парності. При парній кількості двійкових одиниць в байті біт паритету може дорівнювати 1. У цьому випадку говорять, що паритет є непарних (odd parity).

Повідомлення ModBus RTU передаються у вигляді кадрів, для кожного з яких відомо початок і кінець. Ознакою початку кадру є пауза на лінії. Кадр повинен передаватися безперервно. Якщо при передачі кадру виявляється пауза тривалістю більше 1,5 шістнадцятирічних символу (6 біт), то вважається, що кадр містить помилку і повинен бути відхилений приймають модулем.

Метод контролю даних за допомогою CRC використовується незалежно від перевірки паритету. Значення CRC встановлюється в провідному пристрої перед передачею. Після отримання повідомлення обчислюється CRC для всього повідомлення і порівнюється з його значенням, зазначеним у полі CRC кадру. Якщо обидва значення збігаються, вважається, що повідомлення не містить помилки.

Прикладний рівень протоколу ModBus є незалежним від фізичного і каналного, зокрема, він може використовувати протоколи Ethernet, інтерфейси RS-232, RS-422, RS-485, оптоволоконні, радіоканали і інші фізичні середовища для передачі сигналів.

При використанні протоколу прикладного рівня з різними протоколами транспортного і каналного рівня зберігається незмінним основний блок ModBus-повідомлення, разом із кодом функції і даними (PDU). До блоку PDU можуть додаватися додаткові поля при використанні його в різних промислових мережах і тоді він називається "ADU" Application Data Unit – елемент даних програми [16].

Стандартом ModBus передбачені три категорії кодів функцій: встановлені стандартом, що задаються користувачем і зарезервовані. Коди функцій є числами в діапазоні від 1 до 127. Коди в діапазоні від 65 до 72 і від 100 до 110 відносяться до заданим користувачем функцій, в діапазоні від 128 до 255 зарезервовані для пересилання кодів помилок у відповідному повідомленні.

Якщо топологія мережі використовує стандарт Ethernet і многомастерний обмін, то вигідніше використовувати протокол PROFIBUS. Мережа PROFIBUS використовує тільки перший і другий рівні моделі OSI. PROFIBUS має три модифікації.

- profibus DP використовує рівні 1 і 2 моделі OSI, а також призначений для користувача інтерфейс. Безпосередній доступ з призначеного для користувача додатки до каналного рівня здійснюється за допомогою DDLM (Direct Data Link Mapper). Інтерфейс забезпечує функції, необхідні для зв'язку з пристроями введення-виведення і контролерами. Profibus DP на відміну від FMS і PA побудований таким чином, щоб забезпечити найбільш швидкий обмін даними з пристроями, підключеними до мережі.

- profibus FMS використовує 7 рівень моделі OSI і застосовується для обміну даними з контролерами і комп'ютерами на реєстровому рівні. Profibus FMS надає велику гнучкість при передачі великих обсягів даних, але програє протоколу DP в популярності внаслідок своєї складності.

- profibus PA (Profibus for Process Automation) використовується для автоматизації технологічних процесів, він використовує фізичний рівень на основі стандарту IEC 1158-2, який забезпечує живлення мережевих пристроїв через шину і не сумісний з RS-485. Особливістю Profibus PA є можливість роботи у вибухонебезпечній зоні.

У мережі PROFIBUS у ролі ведених пристроїв зазвичай виступають пристрої введення-виведення і вимірювальні прилади. Вони не можуть самостійно отримати доступ до шини і тільки відповідають на запити провідного пристрою.

Особливі вимоги встановлені до мережевого кабелю. Він повинен мати

хвильовий опір від 135 до 165 Ом при погонній ємності не більше 35 пФ/м, площа поперечного перерізу провідників більше 0,34 кв. мм. і погонний опір не більше 110 Ом/км. Кабель повинен мати одну або дві кручені пари з мідним екраном у вигляді сітки або фольги.

По обидва боки лінії передачі підключаються погоджувальні резистори, які встановлені в усі мережеві роз'єми і підключаються за допомогою мікроперемикачів. При швидкостях передачі більше 1,5 Мбіт/с для узгодження лінії додатково використовуються плоскі (друковані) котушки індуктивності.

Для збільшення дальності передачі в PROFIBUS передбачена можливість роботи з оптоволоконним кабелем. При використанні скляного оптоволокна дальність зв'язку може бути збільшена до 15 км.

Profibus PA використовує фізичний рівень, що відповідає стандарту ІЕС 1158-2. Дані передаються за допомогою рівнів струму ± 9 . Для передачі даних використовується NRZ-кодування і 11-бітний формат, ідентичний формату HART-протоколу, що включає стартовий біт, 8 біт даних молодшими розрядами вперед, парний біт паритету і стоп-біт. Біт паритету дорівнює нулю, якщо кількість біт в слові парне і дорівнює одиниці в іншому випадку.

У мережі PROFIBUS для доступу провідних пристроїв до мережі використовується метод передачі маркера рисунок 1.7. У цьому методі мережу має логічну топологію кільця і кожне провідне пристрій отримує доступ до мережі тільки при отриманні маркера.

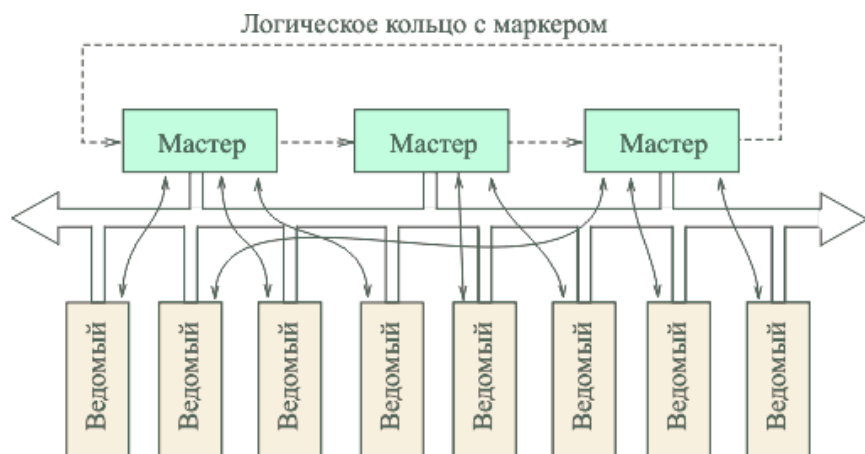


Рисунок 1.7 – Топологія PROFIBUS мережі [17]

Маркер виконує роль арбітра, який надає пристрою право доступу. Після закінчення певного часу цей пристрій має передати маркер наступному ведучому пристрою. Таким чином, кожному провідному пристрою виділяється точно заданий інтервал часу.

Profibus використовує два типи сервісів для передачі повідомлень:

- SRD (Send and Receive Data with acknowledge) відправка і прийом даних з повідомленням;
- SND (Send Data with No acknowledge) відправка даних без повідомлення.

Сервіс SRD дозволяє відправити і отримати дані в одному циклі обміну. Цей спосіб обміну найбільш поширений в Profibus і дуже зручний при роботі з пристроями введення-виведення, оскільки в одному циклі можна і відправити, і отримати дані.

Повідомлення в Profibus називається телеграмою. Телеграма може містити до 256 байт, з них 244 байта даних, плюс 11 службових байт (заголовок телеграми). Всі телеграми мають заголовки однакової довжини, за винятком телеграми з назвою Data_Exchange. Однак 11 байт службової інформації роблять Profibus дуже неефективним при передачі коротких повідомлень, але при великих обсягах даних такий формат телеграми досить ефективний [18].

Узагальнений вид телеграми протоколу Profibus представлений на рисунку 1.8.

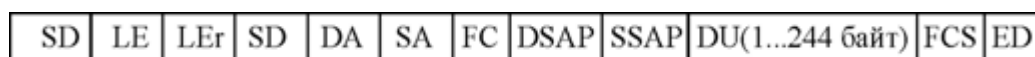


Рисунок 1.8 – Вид телеграми протоколу Profibus [17]

Представлена телеграма складається з одинадцяти частин:

- SD, стартовий роздільник. Використовується для вказівки початку телеграми і її формату;
- LE, довжина переданих даних (DA + SA + FC + DSAP + SSAP + DU);

- LEr, повторення поля LE з метою його резервування;
- DA, адреса пристрою-одержувача телеграми;
- SA, адреса відправника;
- FC, код типу телеграми (запит, повідомлення, відповідь, діагностичні дані, тип пристрою-майстер або ведений, пріоритет, повідомлення);
 - DSAP, пристрій-одержувач використовує це поле щоб визначити, який тип сервісу потрібно виконати;
 - SSAP, COM порт відправника;
 - DU, дані довжиною від 1 до 244 байт;
 - FCS, контрольна сума телеграми (сума значень полів DA + SA + FC + DU, по модулю 255);
 - ED, ознака кінця.

Крім загальнопромислових протоколів обміну, що не залежать від середовища передачі даних існують протоколи обміну, які жорстко пов'язані з середовищем поширення сигналу, одним з яких є CANopen. CANopen розроблений як мережевий протокол, який працює поверх фізичного інтерфейсу CAN, що дозволяє здійснювати обмін даними між пристроями від різних виробників, і гарантує взаємозамінність пристроїв. Прикладний рівень моделі OSI забезпечує інтерфейс між мережею та програмним додатком, яке може взаємодіяти з аналогічними додатками в інших мережах.

Канальний рівень CAN, практично неможливо використовувати в SCADA-системах, оскільки він оперує бітами, фреймами, полями. Для написання ж прикладних програм потрібно використовувати поняття: змінна, масив, подія, клієнт, сервер, ім'я пристрою і т.п.

Всі функціональні можливості прикладного рівня діляться між так званими сервісами. Програмні додатки взаємодіють між собою шляхом виклику відповідних сервісів прикладного рівня.

У протоколі CANopen існує поняття сервісного примітиву, який представляє собою мовну конструкцію, за допомогою якої програмний додаток

взаємодіє з прикладним рівнем. У CANopen існує чотири типи примітивів:

- запит додатка до прикладного рівня, що публікується додатком для виклику сервісу;
- індикація, яку публікує прикладний рівень для додатку, щоб повідомити про внутрішні події, виявлені прикладним рівнем або щоб показати, що сервіс запитаний;
- відповідь, що публікується додатком для прикладного рівня, щоб відповісти на раніше отриману індикацію;
- підтвердження, що публікується прикладним рівнем для додатка, щоб відзвітувати про результати раніше виданого запиту.

Сервіси також діляться на кілька типів:

- локальний сервіс, який виконує запит додатки без взаємодії з іншими сервісами того ж рангу;
- непідтверджений сервіс, який залучає до виконання запиту один або більше інших тимчасових сервісів;
- підтверджений сервіс, який може залучити тільки один сервісний об'єкт того ж рангу. Додаток видає запит до його локального сервісу. Цей запит передається сервісу того ж рангу, який передає його іншому додатку як індикацію. Інша програма видає відповідь, яка передається вихідному сервісу, який передає його як підтвердження запитувачу з додатком;
- сервіс, ініційований провайдером.

CANopen пропонує серію стандартизованих комунікаційних механізмів і функцій, які виконуються пристроями в мережі. Серія профілів доступна і підтримується організацією CiA.

Пристрій в мережі CANopen складається з трьох частин: комунікаційний інтерфейс до шини і програмний протокол обміну; словник об'єктів; інтерфейс до пристроїв введення-виведення і прикладна програма.

Комунікаційний інтерфейс і програмний протокол забезпечують сервіс по передачі і отриманню комунікаційних об'єктів. Словник об'єктів описує типи даних, комунікаційні об'єкти і прикладні об'єкти, використані в пристрої для

обміну через інтерфейс до пристроїв введення/виводу. Прикладна програма забезпечує внутрішнє управління функціями пристрою і інтерфейс до пристроїв введення-виведення.

Найбільш важливою частиною пристрою в CANopen є словник об'єктів. Під об'єктами розуміються типи даних, профілі пристроїв, комунікаційні об'єкти, реєстр помилок. Кожен об'єкт в словнику адресується 16-бітовим індексом.

У CANopen використовуються стандартизовані типи даних: Boolean, Integer, Unsigned Integer, Float, Date, Time. Є також кілька складних типів даних для PDO і SDO параметрів (Process Data Object – об'єкт даних технологічного процесу і Service Data Object – об'єкт сервісних даних).

Об'єкти PDO і SDO використовуються для передачі даних. Повідомлення PDO дозволяють передавати дані в реальному часі. Існує два типи об'єктів PDO.

Перший з них виконує передачу даних (Transmit-PDO або TPDO), другий – прийом даних (Receive PDO або RPDO). Комунікаційні параметри PDO визначають його комунікаційні можливості і описуються в словнику об'єктів.

Об'єкт SDO забезпечує доступ до словника об'єктів. SDO може використовуватися також для передачі груп даних від клієнта до сервера і навпаки.

Комунікаційна модель CANopen визначає різні комунікаційні об'єкти і сервіси, а також доступні режими запуску передачі повідомлень, підтримує передачу синхронних і асинхронних повідомлень [18].

1.4 Аналіз аналогічних рішень за подібним призначенням

В даний момент на ринку представлена велика кількість різних пристроїв і систем промислової автоматизації та управління промисловими апаратами. Основною проблемою пристроїв такого роду є велика кількість різнопланових інтерфейсів зв'язку як внутрішніх, так і мережевих, основними з яких є: SPI, UART, I2C, сімейство інтерфейсів RS-485, Ethernet і Can.

З точки зору розробників автоматизованих систем управління різноплановість в підключенні різних пристроїв, змушує, використовувати «вузький» набір інтерфейсів, який заточується під конкретну групу вирішуваних завдань або певну галузь виробництва. Розглядаючи ринок доступних АСУ, основні виробники: Siemens, ABB, Alan Bradley, WEYTEC, ОВЕН, Adam в основному застосовують стандартизовані інтерфейси.

Якщо розглядати фірму Siemens, то можна відзначити, що всі пристрої, що поставляються фірмою в основному, використовують стандарти мережі Ethernet і протокол обміну Profinet (рисунок 1.9) і іноді RS-485, використання цих інтерфейсів обумовлено ринком їх застосування тобто країнами Європи та Америки. Фірма поставляє велику кількість обладнання: модулі вводу/виводу, комутатори, вимірювальні прилади та перетворювачі.

Устаткування фірми Siemens дозволяє побудувати будь-яку систему автоматичного управління, але, для побудови системи управління будь-яким бізнес-процесом все обладнання лінії повинен мати підтримку Ethernet і протоколу Profinet, що не завжди можливо.

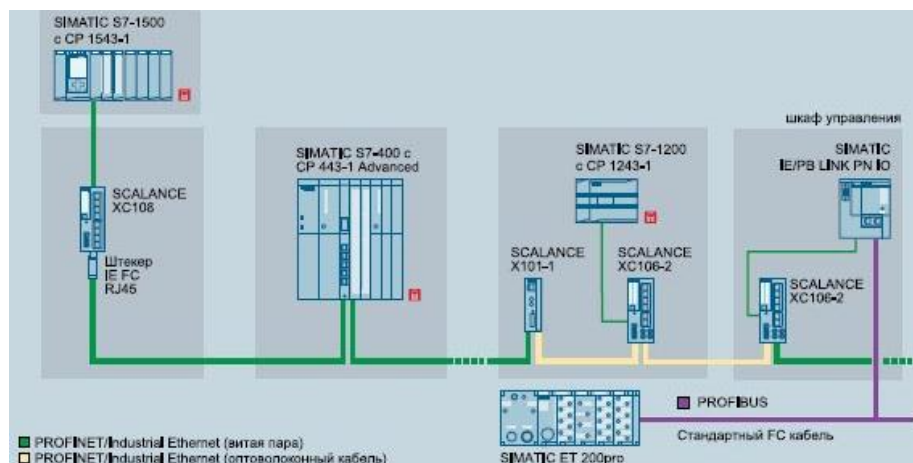


Рисунок 1.9 – Топологія мережі, пропонується фірмою Siemens [19]

Як і Siemens фірма ABB віддає більшу перевагу використанню Ethernet повністю підтримуючи протоколи Profinet і мережу EtherCAT (рисунок 1.10), але при цьому на відміну від Siemens, обладнання фірми ABB має можливість роботи за допомогою інтерфейсів RS-422 і RS-485 і обміну по протоколу

ModBus. Використання стандартних і простих інтерфейсів сімейства RS-485 відкриває для фірми можливості для роботи у більшості регіонів світу.

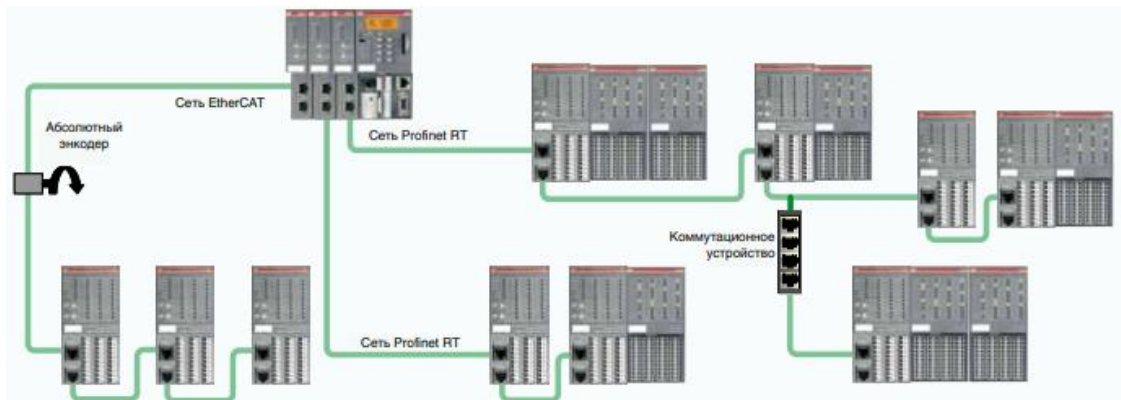


Рисунок 1.10 – Топологія мережі, пропонується фірмою ABB [20]

Як і у фірми Siemens всі об'єкти управління системою є окремими модулями, наприклад дискретні входи/виходи, аналогові входи/виходи, комутатори, вмикачі і т.д. Всі ці модулі є завершеними пристроями. Одним з найбільших недоліків устаткування промислової автоматизації фірми ABB є його висока ціна, низька підтримка з боку розробників, складна процедура модернізації системи, відсутність можливості підключення різних пристроїв по інтерфейсах UART, SPI, I2C та Can.

Якщо розглядати ринок, то особливу увагу можна приділити двом фірмам розробникам обладнання для автоматизації OVEN і Waytek. Фірми займаються випуском як загальнопромислового устаткування (ПЛК, модулі розширення, комутатори/перехідники) так і розробкою датчиків різних типів. Обидві фірми віддають перевагу інтерфейсам RS-485/RS-422, але і підтримують роботу по інтерфейсу Ethernet, протоколам ModBus-TCP і Profinet. Крім підтримки даних протоколів фірма розробляє серію різних перетворювачів інтерфейсів: UART – RS-485, SPI – RS-422, SPI – Ethernet і т.п.

Завдяки наявності великої кількості різних перетворювачів і комутаторів стає можливим побудова систем управління з використанням обладнання різних фірм і датчиків різних типів, що істотно знижує вартість проекту, його

ремонтпридатність, дана особливість і є основною конкурентною перевагою обох фірм.

Для реалізації проектів з великою кількістю різнопланових пристроїв, особливо якщо реалізований проект має на увазі тільки часткову модернізацію виділених ліній або частин підприємств, більшість системних інтеграторів використовують продукцію фірми Adam. Фірма займається розробкою кількох категорій пристроїв:

- цифрові комутатори, для мереж на базі Ethernet;
- перехідники для різних типів інтерфейсів: RS-485 – RS-232, UART RS-485, RS-485 – Ethernet і т.п.;
- блоки розширення, для підключення декількох пристроїв з однаковими інтерфейсами зв'язку до одного, спільного, каналу зв'язку з АСУ.

Саме через велику кількість різних варіантів комутації сигналів фірма Adam є одним з лідерів на ринку. Завдяки великому набору перехідників і блоків розширення стає можливим підключити практично будь-які типи давачів і пристроїв до однієї лінії зв'язку, побудувати мережу або підключитися до вже наявної мережі виробництва без внесення змін до її топології.

1.5 Висновки до розділу 1

Виходячи з виконаного аналізу видно, що основною проблемою всіх промислових систем управління є великий набір різних інтерфейсів обміну, як аналогових, так і цифрових.

З точки зору аналізу обладнання і топології які пропонують різні компанії, видно, що, такі великі компанії як ABB, Siemens, Philips чітко стандартизують і просувають свої технології, тобто для побудова системи автоматизації наприклад з обладнанням ABB, всі складові частини системи: давачі, контактори, вмикачі, виконавчі пристрої повинні мати інтерфейс зв'язку і протокол обміну які використовуються фірмою, що в більшості випадків не можливо. Саме ця проблема змушує системних інтеграторів проектувати

системи використовуючи все обладнання однієї фірми у наслідку чого сильно зростає вартість проекту.

Якщо розглядати запропоноване обладнання фірмами Adam, Owen і WayTec, з точки зору системного інтегратора або інженера проекту, то можна відзначити, що, наявність великого вибору різних комутаторів інтерфейсів дає можливість підключити до системи різні типи пристроїв що зменшує вартість проекту і покращує ремонтпридатність. Але, в такому випадку виникає інша проблема, в схемі системи з'являються додаткові вузли, які знижують її надійність так як кожен додатковий вузол комутації сигналів є потенційним місцем відмови роботи системи.

Проведений аналіз різних інтерфейсів зв'язку, топології промислових мереж і методів побудови систем управління показує, що для спрощення і прискорення процесу проектування систем управління, як з точки зору системних інтеграторів так і розробників прикладних пристроїв, необхідний модуль який зможе виконувати комутацію різних інтерфейсів зв'язку, підключати блоки розширення і підтримувати роботу в промислових мережах. Для реалізації модуля такого роду який буде враховувати всі переваги і недоліки конкурентів необхідно:

- провести розробку концепції модуля, яка буде враховувати всі вимоги і можливість подальшої модернізації;
- провести розрахунок та моделювання методів комутацій інтерфейсів зв'язку;
- обрати компоненти та засоби для реалізації модуля;
- опрацювати основні алгоритми роботи і структуру програмного забезпечення.

РОЗДІЛ 2

ОБҐРУНТУВАННЯ ТА РОЗРОБКА ІНТЕГРОВАНОЇ СИСТЕМИ КОМУТАЦІЇ ПРИКЛАДНИМИ ПРИСТРОЯМИ В АВТОМАТИЗОВАНИХ СИСТЕМАХ УПРАВЛІННЯ

Метою даного розділу магістерської кваліфікаційної роботи є обґрунтування вимог до інтегрованої системи комутації, визначення та розробка її критичних елементів на підставі проведеного аналізу в попередньому розділі.

2.1 Інтегрована система комутації прикладними пристроями в автоматизованих системах управління

Перед початком розробки будь-якого пристрою інженеру-конструктору, інженеру-схемотехніку або програмісту необхідно побудувати концепт тобто уявлення рішення задачі з урахуванням усіх переваг та недоліків.

Використання інтегрованих систем дозволяє поліпшити та оптимізувати процес виробництва, зменшити вплив людини та прискорити виготовлення продукції. Інтегрована система може складатися з великої кількості різних складових частин, складові частини системи можуть поділятися на цехи, лінії, підприємства тощо. Але не змінним залишається наявність промислових механізмів які взаємодіють з людиною.

Поліпшення взаємодії між людиною та машиною, виконуються за допомогою систем збору та обробки інформації, процес централізованої обробки інформації, спрощує алгоритм прийняття рішень та зменшує ризик помилки, при внесенні змін у виробничий процес. Тому для побудови концепту доцільно розглянути структурну схему інтегрованої системи керування яка наведена на рисунку 2.1.

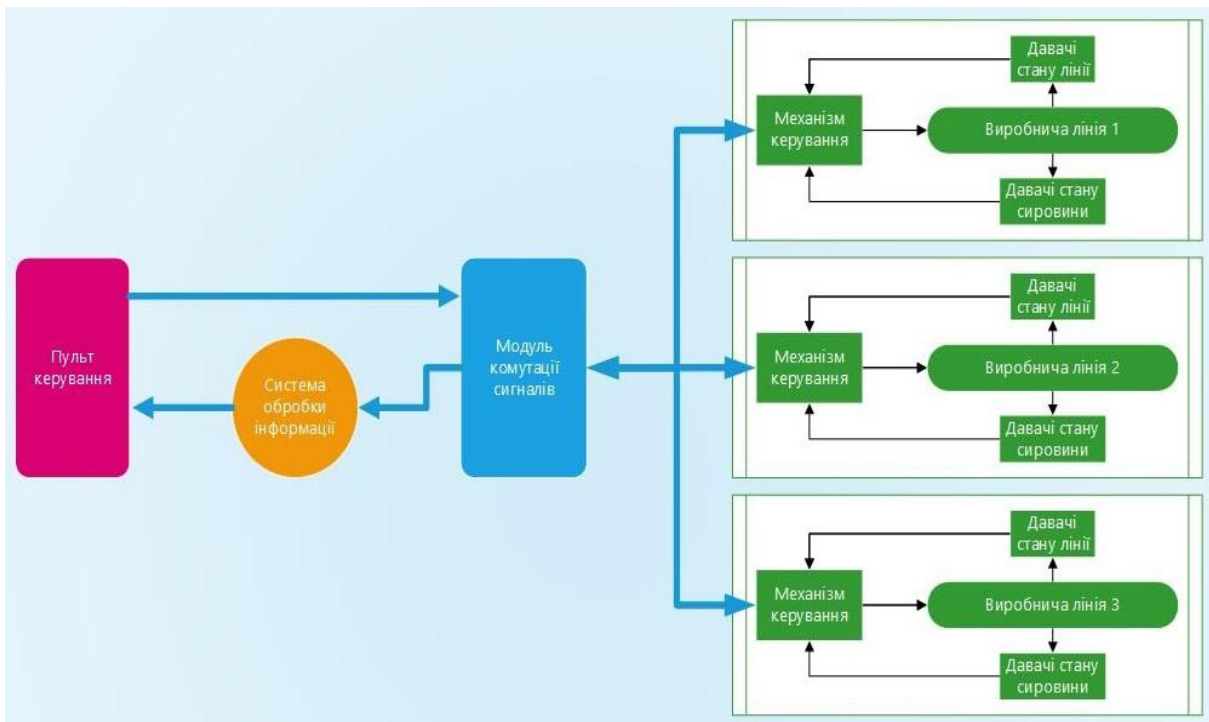


Рисунок 2.1 – Структурна схема інтегрованої системи

Інтегрована система (рис. 2.1), складається з трьох виробничих ліній, модулю комутації інтерфейсів обміну, системи обробки інформації та пульта керування лініями. Виробничі лінії можуть бути різними та мати різні інтерфейси обміну з механізмами керування, що може призводити к ускладненню схеми системи так, як для їх підключення буде необхідно використовувати різні перетворювачі інтерфейсів. Тому між системою обробки інформації та механізмами керування лініями доцільно встановити модуль комутації інтерфейсів зв'язку, який спрощує підключення та зменшує кількість вузлів комутації.

Якщо розглядати основні вимоги, то можна відзначити, що основним завданням модуля є комутація різних типів цифрових і аналогових сигналів, яка дозволить мінімізувати кількості точок відмови системи тим самим підвищуючи її надійність. Крім основної вимоги по надійності до основних завдань модуля можна віднести:

- підключення модуля до мереж з послідовними інтерфейсами зв'язку: RS-485, RS-422 і Ethernet;

- підтримка протоколів обміну на базі стеків: ModBus (RTU, TCP, ASCII), TCP-IP і Profinet;
- підключення до модуля пристроїв по UART, SPI, I2C, і CAN;
- здатність установки сигналу +24 В на виході модуля;
- здатність зчитування та фільтрації сигналу +24 В;
- здатність установки аналогових сигналів на виходах модулю в діапазоні від 4 мА до 10 мА;
- здатність вимірювання аналогових сигналів із заданою дискретністю вибірки;
- можливість конфігурації модуля як окремо, так і в складі промислової мережі;
- передачу і обробку команд із заданою дискретністю.

2.2 Основна концепція модулю, що розробляється та досліджується

Концепт має описувати принцип роботи, ключові компоненти, основний функціонал і формувати основні вимоги для вибору засобів його реалізації. Оскільки основним завданням модуля є мінімізація кількості точок відмови, комутація сигналів і підтримка основних промислових інтерфейсів зв'язку, то алгоритм побудови концепт можна розділити на чотири частини:

- робота з дискретними і аналоговими сигналами;
- комутація цифрових інтерфейсів і зв'язок з АСУ;
- об'єкти управління модулем;
- інтерфейс взаємодії з оператором.

Пропонований концепт модуля комутації цифрових і аналогових сигналів наведено на рисунку 2.2.

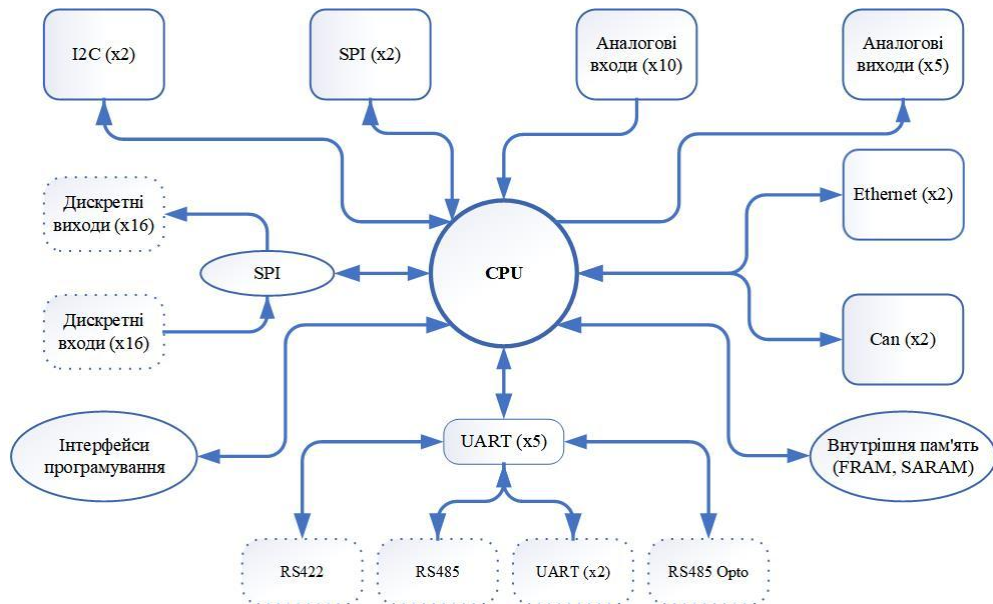


Рисунок 2.2 – Концепт модуля

На рисунку наведено ключові частини модуля, прямокутниками позначені компоненти, які вирішують основну задачу: комутацію сигналів і зв'язок з центральною системою управління, овалом зображені компоненти для програмування і підтримки роботи модуля, кругом зображений центральний керуючий контролер.

2.3 Принцип роботи з аналоговими та дискретними сигналами

Враховуючи проведений аналіз ринку обладнання для АСУ та АСУ ТП видно, що в основному всі представлені пристрої для управління дискретними і аналоговими сигналами працюють в діапазонах 24 В і від 4 мА до 10 мА. При цьому якщо розглядати основні топології схем автоматичного управління кількість пристроїв, що підключаються до аналогових і дискретних входів в одному вузлі, не перевищує 5-10 шт, виходячи з цього в концепт модуля закладено 16 дискретних входів і виходів, 10 аналогових входів і 5 аналогових виходів.

Для реалізації дискретної логіки і зменшення кількості портів введення/виводу контролера доцільно використовувати зсувні регістри. Зсувні

реєстри представляють собою ланцюжок розрядних схем, пов'язаних ланцюгами переносу. Основний режим роботи – зрушення розрядів коду від одного тригера до іншого на кожен імпульс тактового сигналу. Принцип роботи зсувних реєстрів простий, при появі тактового сигналу вміст основного реєстра переписується в додатковий, а при появі наступного сигналу повертається в основний, але вже в сусідні розряди, що відповідає зрушенню слова. Для підключення зсувного реєстру до мікроконтролера використовується SPI.

SPI – послідовний периферійний протокол обміну. Він призначений для зв'язку мікроконтролерів між собою, а також з усілякою периферією: датчиками, АЦП, мікросхемами пам'яті, годинниками [21]. Для передачі даних в SPI використовується чотири лінії:

- Master Input Slave Output (MISO) для прийому даних від веденого;
- Master Output Slave Input (MOSI) для передачі даних веденому;
- Serial Clock (SCK) для тактування лінії;
- Slave Select (SS) для вибору веденого пристрою.

Для вимірювання аналогових сигналів застосовується аналогово-цифровий перетворювач або АЦП, АЦП – це пристрій, що перетворює вхідний аналоговий сигнал в цифровий. Для задач вимірювання значення сигналу в довільний момент часу використовують асинхронний режим роботи АЦП з жорстко не прив'язаними до часу поодинокими аналого-цифровими перетвореннями.

Для задач вимірювання функціональної залежності зміни аналогового сигналу використовують синхронний режим роботи. Синхронний режим роботи АЦП без пропусків даних на як завгодно великому інтервалі часу називають також потоковим режимом. Синхронні АЦП, як правило, підтримують покадровий принцип збору даних, коли оцифровані данні вимірювання утворюють умовні кадри з заданою кількістю, яке відповідає заданим каналам вимірювання [22].

Використання внутрішнього АЦП мікроконтролера в даному випадку дає

велику перевагу, тому що оператор може виконати настройку діапазону роботи каналу, часу квантування і кількості вибірок на періоді.

Для реалізації вимог щодо формування аналогових вихідних сигналів в різних діапазонах, є два варіанти:

- цифро-аналоговий перетворювач, для перетворення цифрового коду в аналоговий сигнал;
- таймери для генерації ШІМ.

Використання першого варіанта, тобто цифро-аналогового перетворювача, неминуче, приводить до подорожчання модуля, тому що в основному все мікроконтролери мають 2-3 вбудованих модуля ЦАП. Тому для реалізації 5 аналогових виходів в схемі необхідно буде використовувати зовнішні мікросхеми ЦАП. У зв'язку з цим більш вигідним варіантом є використання таймерів і формування ШІМ.

Широтна-імпульсна модуляція (ШІМ), це спосіб управління подачею потужності до навантаження. Управління полягає в зміні тривалості імпульсу при постійній частоті проходження імпульсів. Широтна-імпульсна модуляція буває аналоговою, цифровою або двійковою.

В якості комутаційних елементів, при формування ШІМ, застосовуються біполярні і польові транзистори, що працюють в ключовому режимі. Це означає, що частина періоду транзистор повністю відкритий, а частину періоду повністю закритий.

Для формування ШІМ в цифровій електроніці використовуються лічильники або таймери, які завдяки наявності регістрів порівняння формують пилкоподібний сигнал, різної тривалості та скважності.

Цифрові пристрої з ШІМ працюють на постійній частоті, яка обов'язково перевершує час реакції керованого пристрою. Між фронтами тактових імпульсів, вихід ШІМ залишається стабільним, або на високому, або на низькому рівні, в залежності від поточного стану виходу цифрового компаратора, який порівнює рівні сигналів на лічильнику. Приклад формування ШІМ наведено на рисунку 2.3.

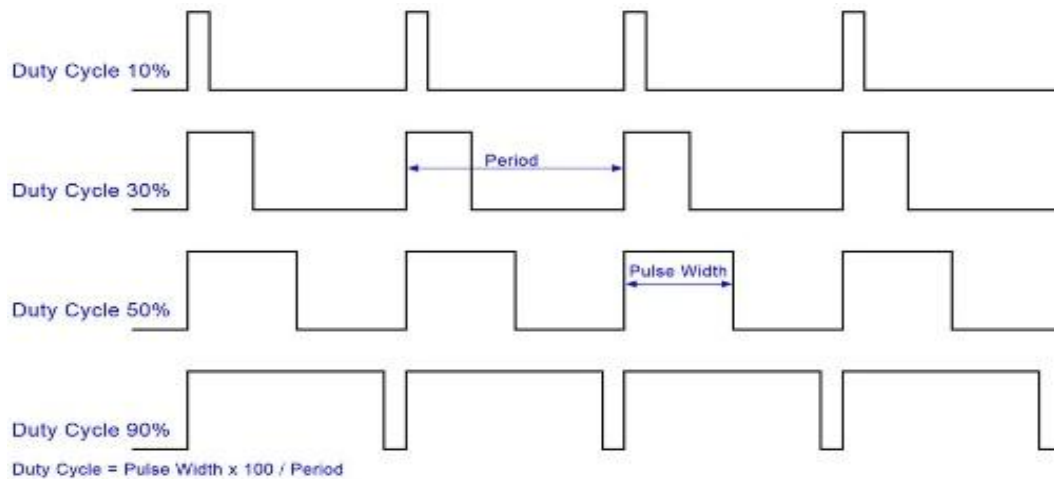


Рисунок 2.3 – Принцип формування ШІМ сигналу [23]

Ставлення повної тривалості періоду імпульсу до часу включення називається коефіцієнтом заповнення імпульсу. Так, якщо час включення складає 10 мкс, а період триває 100 мкс, то при частоті в 10 кГц, склавність буде дорівнювати 10 [23].

2.4 Методи реалізації цифрових інтерфейсів зв'язку

Аналіз ринку пристроїв для автоматизації систем управління вказує на те, що в основному для підключення різних датчиків, контакторів, вимикачів, пристроїв управління і АСУ використовується сім основних інтерфейсів зв'язку: SPI, I2C, UART, Can, RS-422, RS-485 і Ethernet.

Для підтримки роботи з пристроями по послідовному периферійному інтерфейсу (SPI) в концепті модуля виділено два незалежні канали. Завдяки наявності сигналу SS (slave select) до модуля SPI одночасно може бути підключено кілька пристроїв, в зв'язку з цим для кожного каналу SPI зарезервовано п'ять сигнальних ліній, що дозволяє підключити до модуля до 10 пристроїв.

Завдяки тому що SPI є апаратним модулем мікроконтролера, стає можливим виконання налаштування кожного каналу обміну під конкретний пристрій, для якого можна встановити: швидкість роботи, режим роботи

(повний дуплекс чи полу-дуплекс), кількість біт даних, дискретність опитування і т.д.

Крім роботи по інтерфейсу SPI концепт модуля враховує можливість підключення різних пристроїв по інтерфейсу I2C. I2C це послідовна асиметрична шина для зв'язку між інтегральними схемами. I2C використовує дві двонаправлені лінії зв'язку (SDA і SCL), інтерфейс застосовується для з'єднання низькошвидкісних периферійних компонентів з процесорами і мікроконтролерами.

Дані на шині передаються по двох проводах, провід даних і провід тактів. У топології мережі I2C є ведучий пристрій і до 127 ведених, підключення ведених пристроїв до шини здійснюється у вигляді монтажного «I». Приклад топології шини I2C наведено на рисунку 2.4.

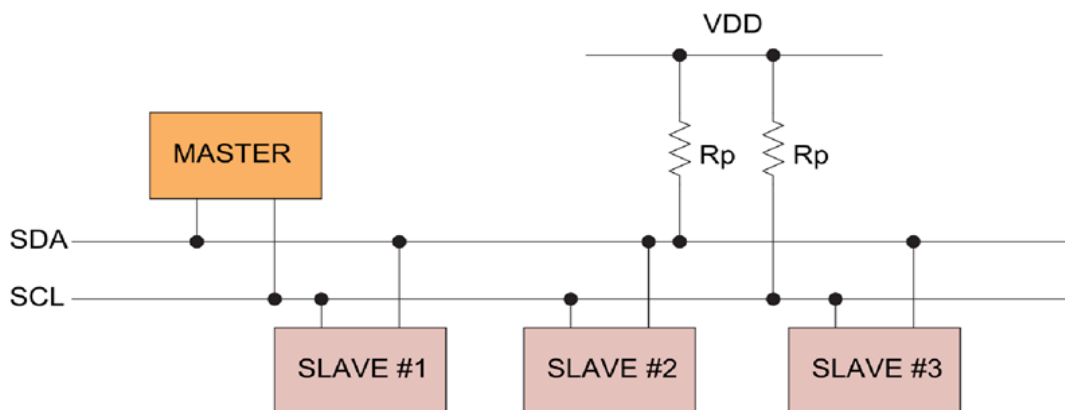


Рисунок 2.4 – Приклад топології шини I2C [24]

Обмін даними (рисунок 2.5) в шині I2C здійснюється за допомогою установки сигналу SDA в стан логічного «0», перекид сигналу в стан логічної «1» здійснюється автоматично, за рахунок перезарядження паразитних ємностей між двома сигнальними лініями. Весь процес передачі даних складається з стартовою послілки, даних і стоповою послілки.

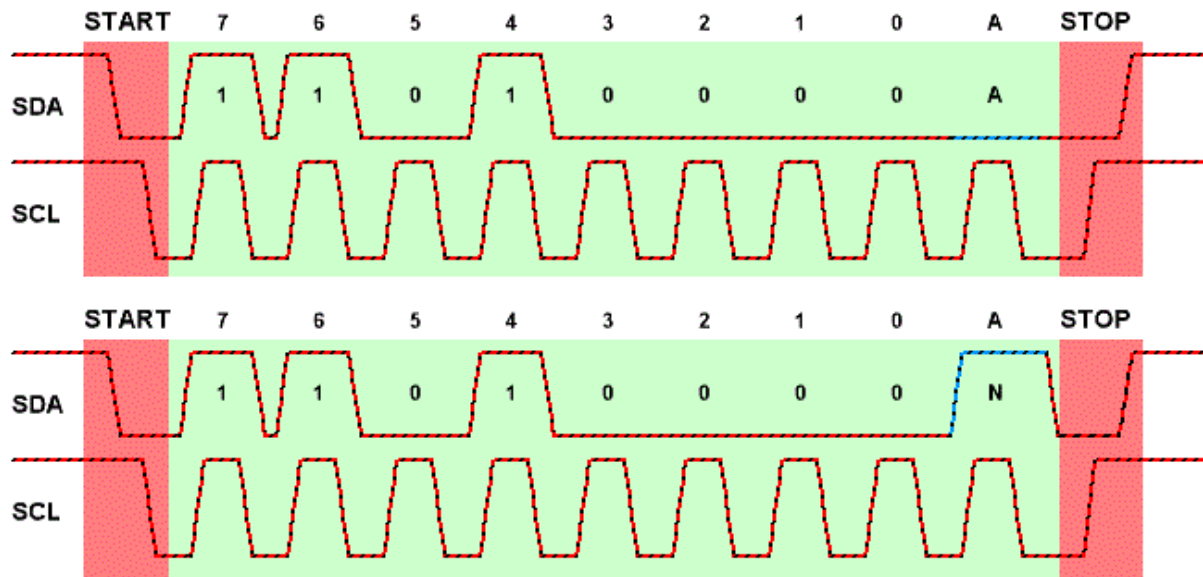


Рисунок 2.5 – Процес передачі даних по I2C [24]

Після формування стартової послідовності передача одного біта інформації відбувається по тактовому імпульсу, тобто коли лінія SCL знаходиться в стані логічного нуля провідне або ведене пристрій встановлює біт на лінії SDA після чого сигнал SCL відпускається і пристрої виконує зчитування даних з лінії. Таким чином I2C є протоколом, який абсолютно не залежить від тимчасових інтервалів. Даний принцип роботи інтерфейсу I2C дає можливість підключити пристрої з різною швидкістю обробки інформації [24].

Так як ринок устаткування для систем автоматичного управління враховує застосування послідовних інтерфейсів зв'язку RS-485 і RS-422, то для їх реалізації в концепті модуля зарезервовано п'ять апаратно-підтримуваних модулів UART.

UART – це універсальний асинхронний приймач, який являє собою логічну схему з одного боку яка підключена до шини обчислювального пристрою, а з іншого має дві сигнальні лінії для зовнішнього з'єднання.

Передача даних в UART здійснюється по одному біту в рівні проміжки часу. Цей часовий проміжок визначається заданою швидкістю UART і для конкретного з'єднання вказується в бодах тобто бітам в секунду. Існує загальноприйнятий ряд стандартних швидкостей: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600 бод [25].

Для синхронізації потоку даних, UART автоматично вставляє мітки, так звані стартовий і стоповий біти. При прийомі ці зайві біти видаляються з потоку. Зазвичай стартовий і стоповий біти обрамляють один байт інформації (8 біт), при цьому молодший інформаційний біт передається першим, відразу після стартового. Зустрічаються реалізації UART, які передають по 5, 6, 7 або 9 інформаційних біт. Обрамлені стартом і стопами біти є мінімальною послідовністю.

Стартовий біт завжди є логічним 0, тому приймач UART чекає перепаду з 1 в 0 і відраховує від нього часовий проміжок в половину тривалості біта. Якщо в цей момент на вході все ще 0, то запускається процес прийому мінімальної послідовності. Для цього приймач відраховує 9 бітових послідовностей поспіль (для 8-бітних даних), і в кожен момент фіксує стан входу. Перші 8 значень є прийнятими даними, останнє значення перевіряє (стоп-біт). Значення стоп-біта завжди дорівнює 1. Якщо реально прийняте значення інше, UART фіксує помилку [25].

Оскільки, синхронізуючі біти займають частину бітового потоку, то результуюча пропускна здатність UART менше швидкості з'єднання, загальний вигляд послідовності UART наведено на рисунку 2.6. Наприклад, для 8-бітних послідовностей формату 8-N-1 синхронізуючі біти займають 20 % потоку, що при фізичній швидкості лінії 115200 бод означає корисну швидкість передачі даних 92160 біт/с або 11520 байт/с.

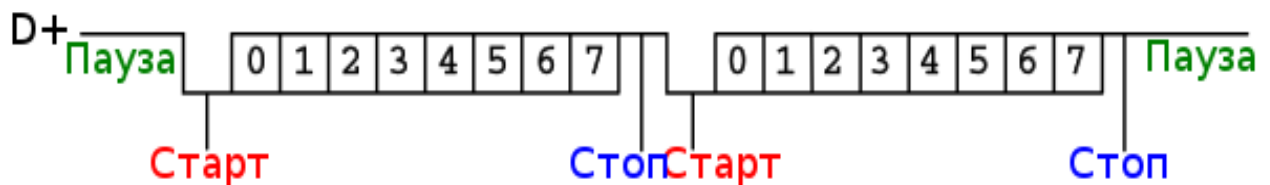


Рисунок 2.6 – Загальний вигляд послідовності UART [25]

Багато реалізації UART мають можливість автоматично контролювати цілісність даних методом контролю бітової парності. Коли ця функція включена, останній біт даних в мінімальній послідовності контролюється логікою UART і містить інформацію про парність кількості одиничних біт в цій

мінімальної послілки. Розрізняють контроль на парність, коли сума кількості одиничних біт в послілки є парним числом, і контроль на непарність, коли ця сума непарна [26].

У концепті модуля зарезервовано п'ять каналів UART:

- канал обміну за допомогою фізичного рівня RS-422;
- канал обміну за допомогою фізичного рівня RS-485;
- два каналу обміну по UART, для підключення пристроїв каналного рівня, наприклад енодера для вимірювання частоти обертання двигуна;
- оптичний канал зв'язку по RS-485 який зарезервований для обміну з АСУ в системах з великою кількістю електромагнітних завад.

Крім описаних інтерфейсів зв'язку, концепт враховує роботу по послідовному інтерфейсу CAN і двох каналах Ethernet. Крім цього, концепт передбачає, що обмін з централізованою системою управління є налаштуванням тобто обмін з АСУ може бути налаштований на один з каналів UART, RS-422, RS-485 або Ethernet.

2.5 Центральний керуючий контролер

В якості вузла управління модулем комутації цифрових і аналогових інтерфейсів виступає мікроконтролер, мікроконтролер це мікросхема, призначена для програмного управління електронними схемами. Основне призначення мікроконтролерів – це управління різними електронними пристроями. Таким чином, вони застосовуються не тільки в персональних комп'ютерах, а й майже у всій побутовій техніці, автомобілях, телевізорах, промислових роботах. Узагальнений приклад складових частин мікроконтролера наведено на рисунку 2.7.

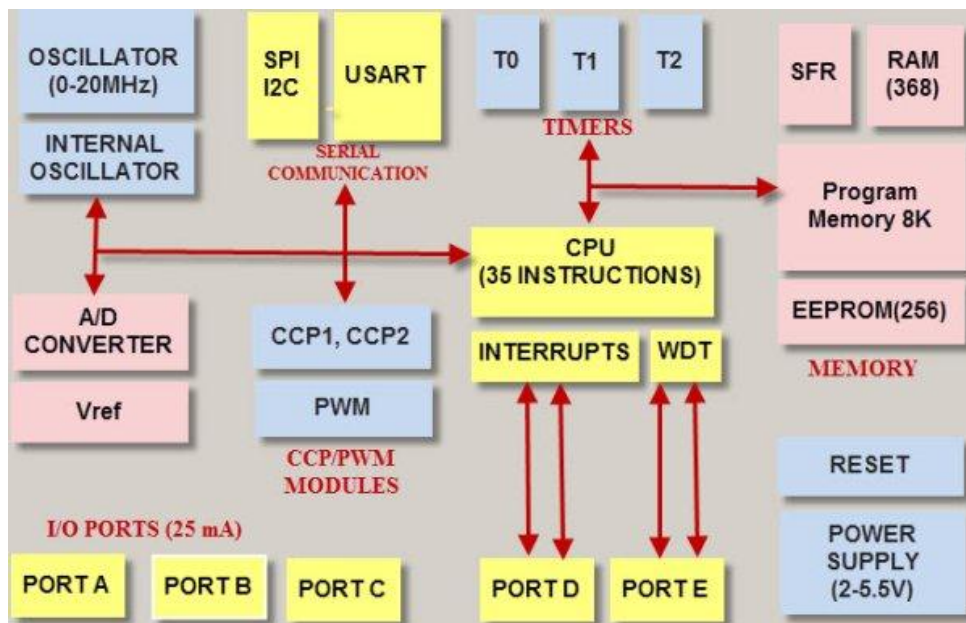


Рисунок 2.7 – Узагальнена схема мікроконтролера [26]

В основному всі мікроконтролери складаються з:

- арифметико-логічного пристрою (АЛП) яке призначене для виконання арифметичних і логічних операцій (ядро);
- оперативно-запам'ятовуючого пристрою (ОЗП) для тимчасового зберігання даних при роботі мікроконтролера;
- пам'яті програм, яка виконана у вигляді постійної пам'яті і призначена для запису програми управління мікроконтролером;
- пам'ять даних для зберігання все можливих констант, табличних значень функцій і т.д.

Крім основних модулів управління мікроконтролер на своєму кристалі може містити: аналогові компаратори, таймери, АЦП, ЦАП і багато інших інтерфейсів. Саме ці особливості даного типу мікросхем і дають можливість реалізувати різного типу пристрою для керування верстатами, моторами, автомобілями і т.д [26].

Для безвідмовної роботи модуля і можливості його модернізації, з урахуванням вимог і його навантаження концепт модуля враховує наявність додаткової пам'яті двох типів, енергонезалежною пам'яті для зберігання

налаштувань, журналів і допоміжної інформації і зовнішнє ОЗП для забезпечення підтримки роботи з протоколами Modbus, TCP-IP і Profinet.

Використання зовнішньої пам'яті дуже затребуване так як робота з мережевими протоколами обміну дуже затратне з точки зору пам'яті всередині мікроконтролера, наприклад стандартний буфер обміну Ethernet при роботі з протоколом TCP-IP має об'єм 1,5 кБ для одного каналу з'єднання або сокета виходячи з цього при роботі з декількома каналами зв'язку внутрішньої пам'яті мікроконтролера може просто не вистачити.

2.6 Інтерфейс оператора

Одна з ключових частин модуля, вимоги до інтерфейсу оператора в основному пов'язані з зручністю використання, інтерфейс оператора на пряму пов'язаний з витратами на навчання та підтримку роботи пристроїв, тобто чим складніше інтерфейс, тим більш кваліфікований персонал повинен обслуговувати його, що не завжди можливо. Більшість замовників віддають перевагу пристроям, які не вимагають витрат на навчання персоналу, виходячи з цього інтерфейс для налагодження та управління модулем повинен бути дуже простим.

Так як модуль підтримує можливість підключення до АСУ за кількома інтерфейсами зв'язку: RS-485, RS-422, Ethernet і CAN, то інтерфейс оператора повинен бути не залежним від каналу зв'язку, виходячи з цього усі реєстри налаштування пристрою повинні бути доступні для редагування декількома засобами:

- за допомогою роботи через стек протоколів Modbus;
- за допомогою роботи через стек TCP-IP тобто пристрій повинен містити веб-сервер;
- за допомогою роботи через інтерфейс програмування і додатки для ПК.

Всі перераховані варіанти роботи інтерфейсу оператора є прикладними програмами тобто для їх реалізації можливе залучення розробників різної галузі, що дозволить прискорити розробку і налагодження програмного забезпечення модуля.

Крім цього даний підхід дає можливість реалізації конкурентоспроможного пристрою, який дає працювати в різних системах управління і враховує більшість вимоги системних інтеграторів.

2.7 Висновки до розділу 2

В результаті побудови концепту модулю комутації цифрових сигналів, можна відзначити що, концепт враховує усі недоліки та переваги розглянутих пристроїв, в результаті розробки отримані основні критерії для вибору ключових компонентів модулю.

Спираючись на проведені дослідження цим критеріям необхідно виконати:

- вибір потрібних компонентів: мікроконтролеру, мікросхем зв'язку, розрахувати блок живлення і т.п;
- побудувати архітектуру програмного забезпечення;
- виконати вибір засобів для реалізації та налагоджування програми;
- промодельовати роботу модуля та зробити висновки про доцільність його використання з урахуванням отриманих результатів.

РОЗДІЛ 3

МОДЕЛЮВАННЯ РОБОТИ МОДУЛЮ КОМУТАЦІЇ

Метою третього розділу магістерської кваліфікаційної роботи є моделювання основних алгоритмів роботи модуля на основі принципів роботи ключових функціональних вузлів управління та формування вимог щодо його апаратної частини.

Завдання побудови моделі може вирішуватися на різних етапах розробки пристрою, на етапі проектування апаратної чи програмної частини або під час реалізації різних алгоритмів на об'єкті управління.

Тому моделювання, основних процесів, що протікають в керованому об'єкті, виконується завжди, модель може бути точною (суворою) або приблизною (м'якою). Суворі моделі максимально описують об'єкт, що моделюється, або явище, під час проектування «строкої» моделі враховуються всі його характеристики, принцип роботи, залежності та особливості функціонування об'єкта при різних типах впливів, в результаті чого, на виході, функціонування моделі практично повністю відповідає функціонуванню реального об'єкта. Перевагою моделей такого роду є висока точність отриманих результатів, проте, основним недоліком таких моделей є їх складність і час, необхідний для реалізації.

"М'яка" модель – це приблизний опис основних принципів роботи об'єкта, при побудові таких моделей частина характеристик об'єкта пропускається. У деяких випадках при побудові моделі беруться об'єкти, які є аналогами або мають велику кількість однакових характеристик. Перевагою таких моделей є швидкість їх проектування, але через відсутність частини характеристик під час моделювання падає точність отриманих результатів.

Основним завданням побудови моделі є отримання інформації про основні принципи взаємодії об'єкта з навколишнім середовищем, необхідні умови та методи побудови системи керування. Крім основних характеристик

об'єкта, модель дозволяє сформулювати основні вимоги до системи, визначити основні режими роботи та її компоненти, підтвердити доцільність її реалізації [27].

3.1 Побудова моделі роботи модуля

Так як модуль виконує комутацію цифрових та аналогових інтерфейсів і не виконує функцій прямого впливу на підлеглі об'єкти автоматизації, для побудови моделі зручно використовувати принцип побудови «м'якої» моделі.

Виходячи з концепту пристрою, наведеного на рисунку 2.1, модуль має кілька типів джерел збору даних, зовнішні, які підключаються до інтерфейсів: I2C, SPI, UART, RS-485, Ethernet та CAN та внутрішні які зосереджені безпосередньо в апаратній частині модуля, збір даних з яких здійснюється за допомогою ADC та зсувних регістрів. Крім цього, в концепті модуля є зовнішні джерела зберігання інформації.

Оскільки модуль підтримує роботу з великою кількістю пристроїв, які мають різні параметри обміну, час обробки запитів та формування відповідей, це накладає свої вимоги та обмеження на архітектуру та алгоритм роботи модуля. Для підтримки роботи системи та збереження її функціонування при максимальному навантаженні модуль повинен працювати в режимі багато потокової обробки даних. Використання потоків дозволить розвантажити роботу контролера, що управляє, і чітко детермінувати обмін з підлеглими пристроями.

На рисунку 3.1 зображено основні потоки отримання та обробки інформації від підлеглих джерел. Зеленим кольором є джерела отримання інформації та буфери для тимчасового зберігання даних сеансу обміну.

Зовнішні джерела отримання інформації розділені на дві групи, перша група – це інтерфейси Ethernet і CAN, вони винесені в окрему групу оскільки є мульти-мастерними і можуть одночасно використовуватися як у режимі ведучий, так і ведений. Друга група зовнішніх джерел – це інтерфейси, які

працюють тільки в режимі ведені [28].

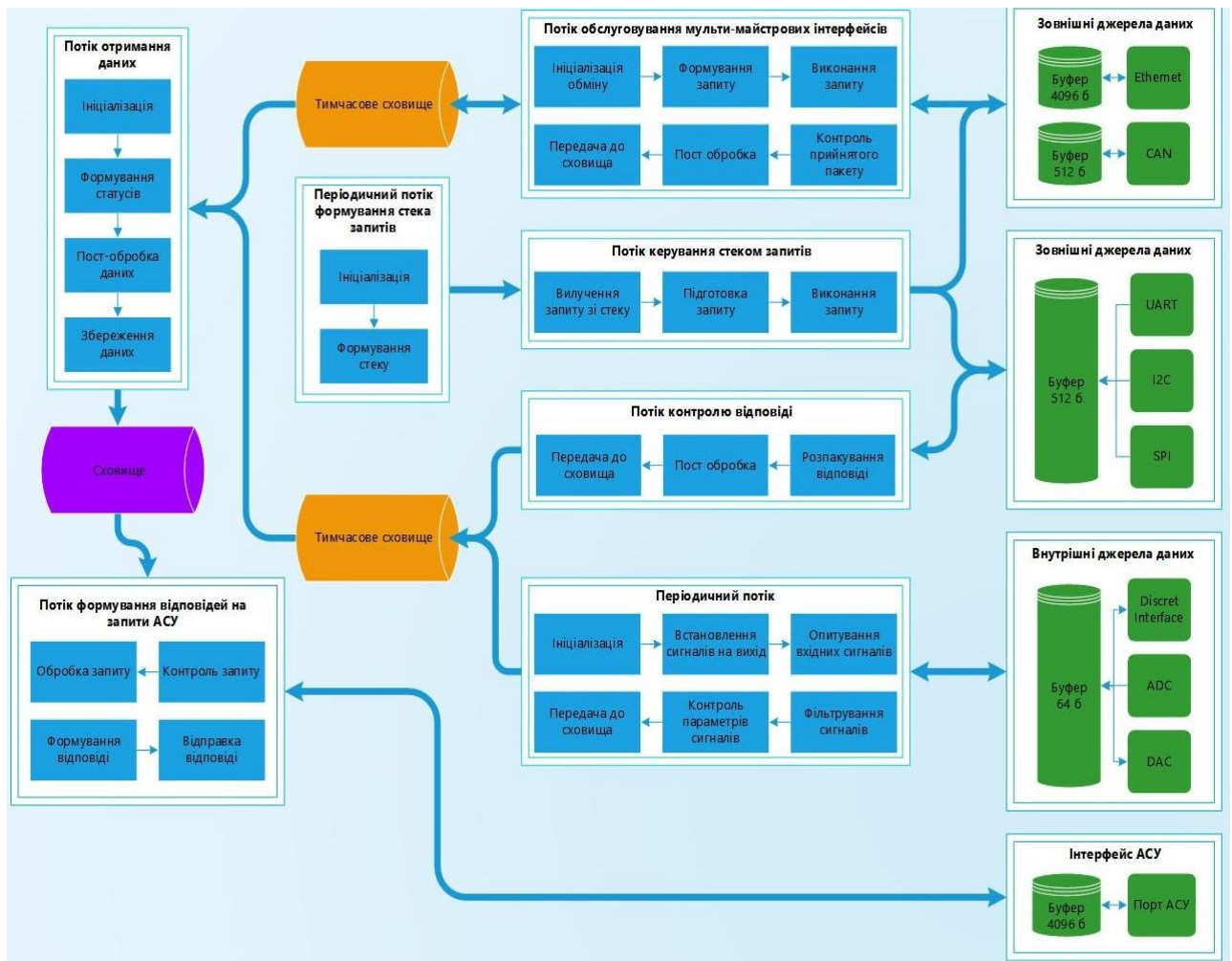


Рисунок 3.1 – Модель потоків збору даних

Ініціалізація обміну даними з підключеними пристроями виконується періодично, запуск обміну відбувається за допомогою потоку, який формує стек запитів до пристроїв після підготовки стека, ініціюється запуск безпосереднього потоку отримання інформації. Цей потік виконує два завдання, перше це налаштування інтерфейсу зв'язку для обміну, друге підготовка та відправка запиту. Обробка відповіді запит здійснюється за допомогою системи переривань і запуску потоку обробки прийнятого пакета (Call-back обробка).

Паралельно потокам формування стека запитів та безпосереднього обміну, виконується запуск потоку періодичної обробки даних внутрішніх інтерфейсів, даний потік виконує збір даних з каналів ADC та зсувних регістрів,

пост обробку даних сигналів та управління модулем DAC.

Після завершення опитування всієї периферії потік управління стеком запитів виконує запуск потоку збору даних, який виконує фінальну обробку, упаковує дані в структури обміну з АСУ і веде журнал подій.

Крім періодичних потоків обробки даних, у системі існує два асинхронні потоки, потік обміну з АСУ та потік обміну за інтерфейсами Ethernet та CAN. Потік обміну з АСУ має найвищий пріоритет, що гарантує дискретність опитування з провідним пристроєм.

Для побудови моделі та виконання розрахунків необхідно задати параметри моделі, оскільки модуль підтримує роботу з різними інтерфейсами зв'язку, які у свою чергу мають різні параметри обміну, для виконання розрахунків зручно побудувати таблицю, в якій будуть описані основні параметри обміну по кожному інтерфейсу. Побудова таблиці дозволить задати базові вимоги до моделі та апаратної частини модуля.

У таблиці необхідно представити базові параметри обміну: частоту роботи інтерфейсу, його середньостатистичні параметри, кількість даних за одну транзакцію, час передачі одного байта інформації з урахуванням сервісних даних інтерфейсу, середній розмір посилки, розмір сервісної частини протоколу обміну [29].

3.2 Розрахунок основних параметрів моделі

Так як робота з підключеними пристроями виконується за допомогою мікроконтролера і багато потокової обробки інформації, і з урахуванням того, що на сучасному ринку мікроконтролерів середня тактова частота ядра мікроконтролера знаходиться в діапазонах від 100 МГц до 200 МГц, це означає, що час виконання однієї асемблерної команди одна наносекунда. Так як модель є приблизною за часу виконання коду зручно взяти певний коефіцієнт, маніпулюючи яким можна врахувати програмну частину процесу обробки інформації. Також, для формування даних моделі необхідно виконати

підготовку наступних параметрів обміну:

- швидкість роботи інтерфейсу;
- час передачі одного байта інформації;
- час передачі одного байта інформації з урахуванням сервісних даних інтерфейсу;
- час передачі даних від веденого пристрою;
- час передачі даних керованого пристрою з урахуванням протоколу обміну;
- сумарний час обміну за інтерфейсом зв'язку.

3.2.1 Параметри обміну за інтерфейсом SPI

Інтерфейс SPI має широкий діапазон робочих частот, від 1 кГц до 50 МГц, крім цього існують модифікації цього інтерфейсу, які мають робочу частоту 100 МГц. Виходячи з аналізу різних типів пристроїв для систем автоматизації, середня частота обміну за цим інтерфейсом дорівнює 25 МГц.

Крім швидкості роботи інтерфейсу, SPI декларує довжину однієї посилки, загальноприйнятими є 8, 16 та 32 біти за один сеанс роботи шини але найрозповсюдженим є передача 8 біт даних за одну транзакцію роботи шини.

Таким чином, для розрахунку сеансу обміну за інтерфейсом SPI, прийmemo його наступні характеристики: $F_{spi} = 25$ МГц, $W_{spi} = 8$ біт.

Швидкість передачі одного біта інформації по інтерфейсу SPI обернена його частоті, формула (3.1) [30]:

$$T_{\text{біт}} = \frac{1}{F_{spi}, \text{МГц}} \text{нс.} \quad (3.1)$$

Підставивши дані у формулу, отримаємо, що час передачі одного біта по інтерфейсу SPI с заданими характеристиками дорівнює 40 нс. Швидкість обміну даними по будь-якому інтерфейсу зв'язку називається baudrate і вимірюється в байтах за секунду, для розрахунку пропускної спроможності

інтерфейсу SPI скористаємося формулою (3.2) [31]:

$$BaudRate_{spi} = \frac{F_{spi}, \text{Гц}}{W_{spi}, \text{біт} \cdot T, \text{с}}, \text{кБ/с.} \quad (3.2)$$

Підставивши дані у формулу, отримаємо:

$$BaudRate_{spi} = \frac{25 \cdot 10^9}{8 \cdot 10^9} = 3,125 \text{ кБ/с.}$$

Маючи час передачі одного біта інформації з інтерфейсу, можна отримати час передачі одного байта інформації, так як байт складається з 8 біт, час передачі розраховуватиметься таким чином, формула (3.3) [31]:

$$T_{байт} = T_{біт} \cdot W_{spi} \text{ нс.} \quad (3.3)$$

Для розрахунку часу усієї транзакції обміну необхідно враховувати кількість байт для читання або запису та сервісну частину обміну, так як на практиці інтерфейс SPI для роботи с даними використовує команди типу читання/запис, стирання та встановлення адреси, то формула розрахунку матиме наступний вигляд, формула (3.4):

$$T_{запит} = (K + D) \cdot T_{байт} \text{ нс,} \quad (3.4)$$

де K – кількість сервісних байт інформації;

D – кількість байт даних;

$T_{байт}$ – час передачі 8 байт інформації [31].

Оскільки процедури модифікації пам'яті завжди супроводжуються установкою адреси, K дорівнюватиме: один байт команда та чотири байти адреса. Коефіцієнт D вказує скільки даних необхідно отримати або записати у

підключеного пристрою, для розрахунків, коефіцієнт D приймемо рівним 32 байтам. Тоді підставивши дані формулу (3.4), отримаємо:

$$T_{\text{запит}} = (5 + 32) \cdot 320 = 11840 \text{ нс.}$$

Так як SPI інтерфейс працює у дуплексному режимі, тобто, в момент запису інтерфейс виконує зчитування, то час повного обміну з пристроєм дорівнюватиме часу $T_{\text{запит}}$, таким чином, час транзакції обміну буде дорівнювати 12 мкс.

3.2.2 Параметри обміну за інтерфейсом I2C

Як і SPI інтерфейс I2C має широкий діапазон робочих частот від 1 кГц до 400 кГц, але на відміну від інтерфейсу SPI, I2C має максимальну довжину посилки, що дорівнює одному байту [32]. Крім цього, інтерфейс I2C має сервісні біти для роботи інтерфейсу, старт біт, стоп біт та біт підтвердження від веденого або провідного пристрою. Таким чином характеристика інтерфейсу буде виглядати так: $F_{I2C} = 100 \text{ кГц}$, $W_{I2C} = 8 \text{ біт}$, $Wc_{I2C} = 3 \text{ біт}$.

Відповідно до характеристик інтерфейсу та формули (3.1). Час передачі одного біта інформації буде дорівнювати 10 мкс, для розрахунку baudrate інтерфейсу підставимо дані у формулу (3.2) та отримаємо:

$$BaudRate_{I2C} = \frac{100 \cdot 10^8}{8 \cdot 10^9} = 1,25 \text{ кБ/с.}$$

Для розрахунку часу передачі одного байта інформації за інтерфейсом I2C модифікуємо формулу (3.3), додавши до неї сервісну частину інтерфейсу I2C, в результаті модифікації отримаємо формулу (3.5):

$$T_{\text{байт I2C}} = T_{\text{біт}} \cdot (W_{I2C} + Wc_{I2C}) \text{ нс.} \quad (3.5)$$

де $T_{\text{біт}}$ – час передачі одного біта інформації;

W_{I2C} – кількість біт в байті;

$W_{C_{I2C}}$ – кількість сервісних біт.

Тоді підставивши дані у формулу, отримаємо що час передачі одного байта інформації веденому пристрою дорівнює 110 мкс:

$$T_{\text{байт } I2C} = 10000 \cdot (8 + 3) = 110000 \text{ нс.}$$

Маючи час передачі одного байта інформації та з урахуванням того, що I2C, як і інтерфейс SPI має командну структуру обміну, зручно використовувати аналогічні параметри обміну, але з однією відмінністю інтерфейс I2C є напівдуплексним, тобто провідний пристрій формує запит на отримання потрібної кількості даних за адресою, а ведений пристрій у свою чергу після аналізу запиту починає їх надсилати, таким чином підставивши дані у формулу (3.5) отримаємо:

$$T_{\text{запит}} = 5 \cdot 110000 = 550000 \text{ нс.}$$

Таким чином, час відправлення запиту веденому пристрою дорівнює 550 мкс, розрахунок часу отримання відповіді аналогічний розрахунку часу відправлення запиту, тому підставивши дані у формулу (3.4) отримаємо:

$$T_{\text{відповідь}} = 32 \cdot 110000 = 3520000 \text{ нс.}$$

Внаслідок чого, час транзакції обміну за інтерфейсом I2C дорівнює 3,5 мс. Маючи час надсилання запиту та час отримання відповіді, можна отримати сумарний розмір транзакції обміну з пристроєм формула (3.6):

$$T_{\text{обміну}} = (T_{\text{запит}} + T_{\text{відповідь}}) + T_{\text{таймаут}} \text{ нс,} \quad (3.6)$$

де $T_{\text{запит}}$ – час на надсилання запиту;

$T_{\text{відповідь}}$ – час отримання відповіді;

$T_{\text{таймаут}}$ – час необхідний веденому пристрою для формування відповіді.

Величина тайм-ауту на обробку відповіді від веденого пристрою у всіх об'єктів автоматизації має різну величину і залежить від тактової частоти роботи пристрою, у зв'язку з цим приймемо цей час рівним 1 мс, тоді підставивши дані у формулу (3.6) отримаємо:

$$T_{\text{обміну}} = (550000 + 3520000) + 1000 = 4071000 \text{ нс.}$$

В результаті розрахунків сумарний час обміну з одним веденим пристроєм за інтерфейсом I2C дорівнюватиме 4.1 мс.

3.2.3 Параметри обміну за інтерфейсом UART

На відміну від вище розглянутих інтерфейсів SPI та I2C, розрахунок параметрів обміну за інтерфейсом UART здійснюється іншим чином з двох причин, перша це стандартизований ряд швидкостей обміну за інтерфейсом: 300, 600, 1200, 2400, 4800, 9600, 19200, 3 115200, 230400, 460800 та 921600 біт/с. Друга наявність можливості налаштувати сервісну частину інтерфейсу обміну, а саме кількість стоп-біт, біт контролю парності. Крім цього, інтерфейс має обов'язковий стартовий біт та декларує передачу даних лише у форматі байта (8 біт). Тому параметри обміну для інтерфейсу UART будуть виглядати так: $\text{BaudRate}_{\text{UART}} = 115200 \text{ біт/с}$, $W_{\text{UART}} = 8 \text{ біт}$, $W_{\text{CUART}} = \text{стартовий біт} + \text{біт паритету} + \text{стоповий біт}$.

Виходячи з параметрів UART швидкість передачі одного біта інформації розраховується за формулою (3.7):

$$T_{\text{біт}} = \frac{10^9}{\text{BaudRate}_{\text{UART}}} \text{ нс.} \quad (3.7)$$

Тоді підставивши дані у формулу, отримаємо, що час передачі 1 біт інформації по інтерфейсу UART с заданими характеристиками, буде дорівнювати 8680 нс або 8,7 мкс. Маючи тривалість передачі одного біта інформації та скориставшись формулою (3.5), отримаємо:

$$T_{\text{байт } UART} = 8680 \cdot (8 + 3) = 95480 \text{ нс.}$$

У результаті розрахунків отримуємо, що для передачі одного байта інформації з урахуванням сервісних даних інтерфейсу необхідно 95,5 мкс. Крім швидкості передачі байт, під час обміну UART між послідовками існують паузи, яка зазвичай дорівнюють тривалості 3,5 байта даних на встановленій швидкості обміну, таким чином формула для розрахунку тайм-ауту має вигляд формула (3.8).

$$T_{\text{паузи}} = \frac{(W_{UART} + W_{C_{UART}}) \cdot 3,5}{\text{BaudRate}_{UART}} = T_{\text{байт } UART} \cdot 3,5 \text{ нс.} \quad (3.8)$$

Використовуючи цю формулу, отримуємо, що пауза між послідовками під час обміну UART для обраної швидкості обміну 115200 біт/с становить 334,25 мкс. Наявність пауз між кадрами даних, що передаються істотно знижує швидкість обміну.

Оскільки модуль є провідним пристроєм у мережі, розрахунок часу обміну даними з веденими пристроями, у разі, як і з I2C, ділиться на два етапи, перший етап – це передача запиту, другий етап отримання відповіді на запит. Оскільки UART є лише інтерфейсом обміну, по верх якого працює протокол обміну, то формула розрахунку часу відправки запиту буде виглядати наступним чином (формула 3.9):

$$T_{\text{запит}} = (PDU \cdot T_{\text{байт } UART}) + (PDU \cdot T_{\text{паузи}}) \text{ нс.} \quad (3.9)$$

Де, PDU є величиною, яка описує розмір команди протоколу обміну, в основному PDU містить у собі: команду (читання/запис), кількість байт даних, дані та контрольну суму. Наявність PDU помітно знижує швидкість обміну, оскільки корисні дані оздоблюються сервісною інформацією протоколу обміну.

В основному всі пристрої підключені по UART обмінюються даними за протоколом Modbus, даний протокол декларує ряд зарезервованих команд для читання та запису інформації, всі команди протоколу мають обов'язкові поля: адреса пристрою, код функції Modbus, початкова адреса для виконання операцій читання або запису кількість байт даних та два байти CRC. Тому для моделювання, приймемо розмір PDU рівним 8 байт. Таким чином, підставивши дані у формулу (3.9), отримаємо:

$$T_{\text{запит}} = (8 \cdot 95480) + (8 \cdot 334250) = 3437840 \text{ нс.}$$

В результаті розрахунку отримуємо, що час передачі запиту дорівнює 3,5 мс. Відповідь на запит за протоколом Modbus має таку ж величину PDU, відмінністю є кількість байт даних, що запитуються, для зручності подальших розрахунків приймемо кількість байт корисної інформації рівним 32 байтам, з урахуванням PDU сумарний розмір пакета буде дорівнювати 40 байтам тоді підставивши дані формули 3.9, отримаємо що час передачі відповіді буде дорівнювати 17,2 мс, тоді підставивши дані у формулу 3.6 сумарний час транзакції обміну дорівнюватиме 20,7 мс.

3.2.4 Параметри обміну за інтерфейсом CAN

Як інтерфейс UART, інтерфейс обміну CAN суворо декларує швидкість обміну по шині 1 Мбіт/с, крім швидкості обміну інтерфейс має велику кількість сервісної інформації: старт біт, поле арбітражу рівне 11 біт, біт дозволу доступу до шини, біт ідентифікації розширеного протоколу обміну, резервний біт, код довжини посилки який дорівнює 4 бітам, поле даних яке дорівнює 4 байтам, 16 біт контрольної суми, біт розмежування CRC, біт підтвердження,

бітрозмежувач підтвердження та 7 нульових біт для формування кінця кадру. Таким чином, характеристики інтерфейсу обміну будуть виглядати наступним чином: $\text{BaudRate}_{\text{CAN}} = 1 \text{ Мбіт/с}$, $W_{\text{CAN}} = 8 \text{ біт}$, $W_{\text{CCAN}} = 44 \text{ біт}$ і кількість переданих байт яке дорівнює 4, що зумовлено інтерфейсом.

Для розрахунку часу передачі одного біта інформації скористаємося формулою (3.6), в результаті підстановки даних отримаємо:

$$T_{\text{біт}} = \frac{10^9}{10^6} = 1000 \text{ нс.}$$

В результаті розрахунку отримаємо, що час передачі одного біта інформації дорівнює 1 мкс. Маючи час передачі одного біта інформації, здійснимо розрахунок часу передачі сервісної інформації протоколу, скориставшись формулою (3.4), в результаті підстановки даних отримаємо:

$$T_{w\text{c CAN}} = 1000 \cdot 44 = 44000 \text{ нс.}$$

Таким чином отримуємо що час передачі сервісної частини даних інтерфейсу дорівнює 44 мкс, тоді сумарний час передачі 4 байт інформації буде розраховуватися за формулою (3.10):

$$T_w = ((W_{\text{CAN}} \cdot 4) \cdot T_{\text{біт}}) + T_{w\text{c CAN}} \text{ нс.} \quad (3.10)$$

Підставивши дані у формулу час однієї транзакції обміну дорівнює 76 мкс. Обмін за інтерфейсом CAN здійснюється в режимі запит/відповідь, тому час читання 32 байт інформації буде дорівнювати формула (3.11):

$$T_{\text{обміну}} = \left(T_w \cdot \left(\frac{D}{4} \right) \right) \cdot 2, \text{ нс} \quad (3.11)$$

де T_w – час передачі 4 байт інформації;

D – кількість байт переданої інформації.

Підставивши дані у формулу (3.11) отримаємо що час читання 32 байт інформації дорівнює 1,2 мс.

3.2.5 Параметри обміну за інтерфейсом Ethernet

З усіх вище розглянутих інтерфейсів обміну, інтерфейс Ethernet є найшвидшим, швидкості обміну даного інтерфейсу стандартизовані: 10 Мбіт/с, 100 Мбіт/с, 1 Гбіт/с. На швидкість обміну впливає цілий ряд параметрів: тип підключеного пристрою, тип кабелю, довжини лінії і т.д. Виходячи з аналізу різних топологій промислових мереж видно, що в основному всі мережі працюють зі швидкістю 10 Мбіт/с, що обумовлено наявністю довгих ліній та повільних пристроїв.

Через високу популярність Ethernet в даний момент існує велика кількість різних протоколів обміну, але вони мають стандартну ширину кадру рівну 1526 байтам. Фрейм складається з 7 байт преамбули для синхронізації приймача, 6 байтової адреси одержувача та 6 байтової адреси відправника, довжину фрейму, яка дорівнює 2 байтам, 1500 байт даних, 2 однобайтових сервісних поля DSAP та SSAP, 2 байтове поле з вказаним CRC.

Таким чином підставивши характеристики інтерфейсу у формулу (3.7), отримаємо швидкість передачі одного біта інформації дорівнює:

$$T_{\text{біт}} = \frac{10^9}{10 \cdot 10^6} = 100 \text{ нс.}$$

В результаті розрахунку отримаємо, що швидкість передачі одного біта інформації дорівнюватиме 100 нс. оскільки обмін за інтерфейсом здійснюється, строго фреймами по 1526 байт, виконаємо розрахунок швидкості передачі корисних даних, для цього скористаємося формулою (3.12):

$$T_w = (W_{\text{Ethernet}} \cdot 8) \cdot T_{\text{біт}} \text{ нс.} \quad (3.12)$$

Підставивши дані у формулу, отримаємо що час передачі корисних даних дорівнює 1,2 мс, а час передачі сервісної частини кадру дорівнює 2,6 мкс. Оскільки обмін даними відбувається у режимі запит відповідь, то сумарний час отримання даних від веденого пристрою дорівнює 2,4 мс.

Дані результатів розрахунків для випадку обміну по 32 байти інформації з веденими пристроями окрім інтерфейсу Ethernet який має сталу величину розміру пакету наведені таблиці 3.1.

Таблиця 3.1 – Дані для моделювання роботи потоків

Інтерфейс	BaudRate, Мбіт/сек	$T_{\text{біт}}$, мкс	$T_{\text{байт}}$, мкс	$W_{\text{запит}}$, байт	$T_{\text{запит}}$, мс	$W_{\text{відповідь}}$, байт	$T_{\text{відповідь}}$, мс	$T_{\text{обміну}}$, мс
SPI	25	0,04	0,32	37	0,012			
I2C	0,1	10	1,1	5	0,55	32	3,5	4,1
UART	0,115	8,7	95,5	8	3,5	40	17,2	20,7
CAN	1	1	8	32	0,61	32	0,61	1,22
Ethernet	10	0,1	0,8	1526	1,22	1526	1,22	2,44

3.3 Моделювання роботи потоків

Крім розрахунків основних характеристик обміну, для моделювання роботи потоків отримання та обробки даних необхідно задати періоди запуску потоків. Відповідно до рисунку 3.1. в системі є два періодичні потоки, перший потік виконує ініціалізацію обміну із зовнішніми пристроями, внаслідок запуску даного потоку відбувається послідовний запуск потоку обміну даними, потоку обробки відповіді та потоку збору отриманої інформації [32].

Другий потік це потік обробки внутрішніх інтерфейсів. Даний потік запускається так само періодично, виконує опитування АЦП, зсувних регістрів та обробку DAC, але на відміну від потоку запуску обміну із зовнішніми пристроями даний потік завершує свою роботу відразу після обробки всіх інтерфейсів. Оскільки мікроконтролер виконує опитування внутрішніх

інтерфейсів набагато швидше, то приймемо період виклику цього потоку рівним 10 мс.

Для встановлення періоду виклику потоку обробки зовнішніх інтерфейсів потрібно враховувати, що у разі періодичного виклику потоків період його виклику має бути більшим, ніж час його виконання. Для розрахунку періоду скористаємося формулою (3.13):

$$T_{\text{виконання}} = T_{SPI} + T_{I2C} + T_{UART} + T_{CAN} + T_{Ethernet} \text{ мс.} \quad (3.13)$$

Таким чином підставивши дані у формулу 3.25, отримаємо сумарний час обміну з зовнішніми пристроями. Враховуючи, що інтерфейси CAN та Ethernet обробляються в окремому потоці та працюють у мультимайстерних мережах необхідно врахувати їхню можливу зайнятість під час запуску основного потоку збору даних. У такому разі формула розрахунку періоду запуску потоку обміну буде виглядати, як формула (3.14):

$$T_{\text{виконання}} = T_{SPI} + T_{I2C} + T_{UART} + (T_{CAN} \cdot 2) + (T_{Ethernet} \cdot 2) \text{ мс.} \quad (3.14)$$

Крім часу виконання опитування пристроїв необхідно враховувати час обробки відповіді від кожного з них а також, необхідно врахувати час на перемикання між потоками і час на запуск нової транзакції обміну.

Даний час буде в кожен момент різним, так як він залежить від поточного навантаження на диспетчері завдань і часу обробки відповіді на запит, з метою зручності моделювання приймемо даний час за константу, яка дорівнює 6 мс.

Так само необхідно врахувати той факт, що крім головного потоку обміну з пристроями в системі існують інші додаткові завдання, які так само вимагають час для їх виконання, цей час так само приймемо за константу рівну 10 мс. В результаті чого, фінальна формула розрахунку періоду виклику потоків збору інформації буде виглядати, як формула (3.15):

$$T_{\text{виконання}} = \sum (T_{\text{в}} + K_{\text{п}}) + \sum ((T_{\text{мв}} \cdot 2) + K_{\text{п}}) + K_{\text{idle}} \text{ мс}, \quad (3.15)$$

де $T_{\text{в}}$ – час опитування пристроїв, які є веденими (SPI, I2C, UART);

$T_{\text{мв}}$ – час опитування пристроїв, які можуть бути як провідними, так і веденими (Ethernet та CAN);

$K_{\text{п}}$ – константа, яка описує час обробки прийнятих даних та час роботи диспетчера завдань;

K_{idle} – додатковий час між сеансами обміну.

Для більш детального аналізу роботи системи, зробимо поетапне отримання циклограм, та виконаймо розрахунки при різних навантаженнях системи. Так як параметри обміну за інтерфейсами є сталими величинами які декларуються підключеними пристроями та топологіями розглянутих систем, то в якості змінних параметрів навантаження обміну використаємо розмір транзакції який з кожним разом будемо збільшувати у чотири рази.

3.3.1 Моделювання обміну за інтерфейсами по 32 байти

Параметри транзакцій обміну по 32 байти вже отримані і представлені у таблиці 3.1, окрім цього час виконання та періоди виклику потоків збору даних із внутрішніх інтерфейсів вже був прийнятий рівним 10 мс, тому для побудови графіків необхідно розрахувати лише час виконання транзакції обміну з усіма пристроями. Тому, підставивши данні у формулу 3.15 отримаємо що час обміну даними, по 32 байти, по усім інтерфейсам обміну буде дорівнювати 76,4 мс, з урахуванням кратності іншим циклам приймемо цю величину рівною 80 мс. Маючи усі параметри роботи потоків отримаємо циклограму транзакцій обміну даних яка зображена на рисунку 3.2.

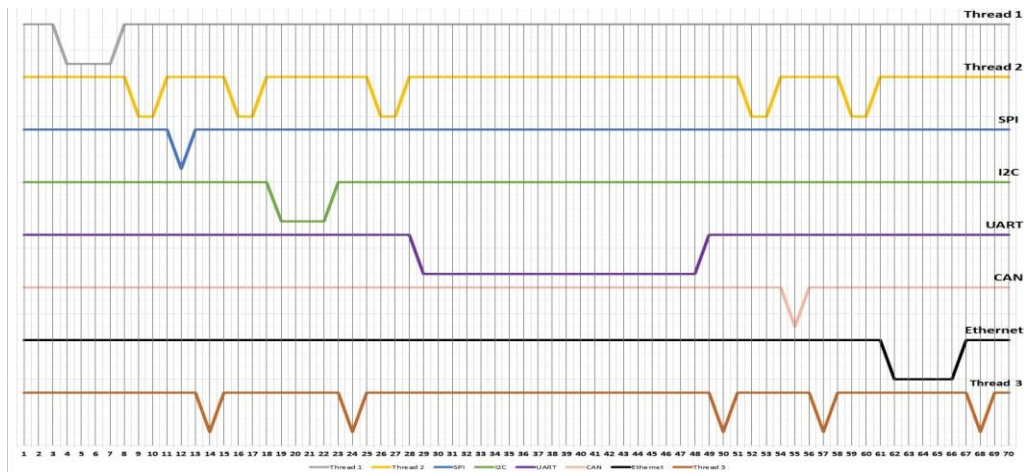


Рисунок 3.2 – Циклограма обміну із зовнішніми пристроями

На циклограмі зображено процес збору даних із зовнішніх пристроїв, «Thread 1» виконує запуск процедури обміну та формує стек запитів для кожного підключеного пристрою до інтерфейсу обміну, після заповнення стеку запитів потік ініціює запуск потоку «Thread 2».

Мета потоку «Thread 2» виконання запитів до ведених пристроїв, окрім запитів даних потік контролює тайм-аут відповіді від пристрою. В результаті запиту, завдяки системі переривань, виконується запуск потоку «Thread 3» який займається обробкою отриманої відповіді, та розблоковує процесу подальшого обміну. Повна процедура збору даних від ведених пристроїв з урахуванням фінальної обробки отриманої інформації наведено на рисунку 3.3.

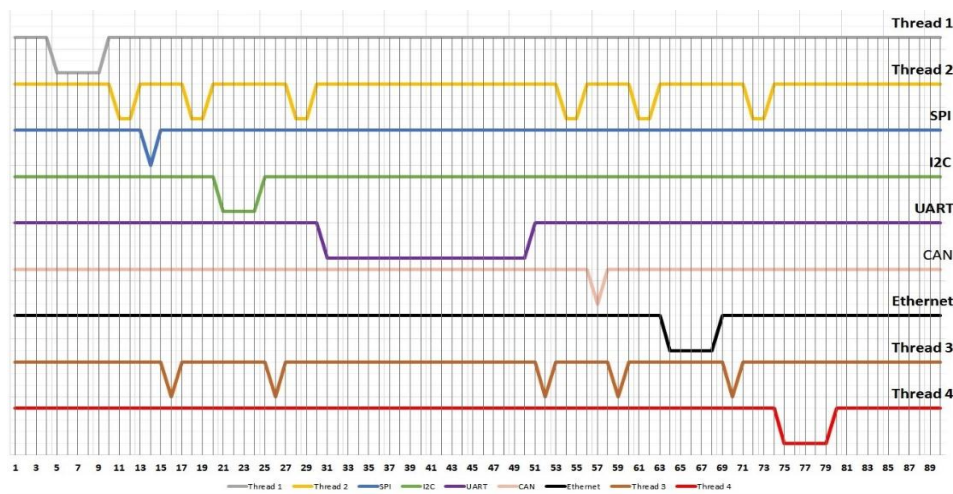


Рисунок 3.3 – Циклограма отримання та обробки даних від зовнішніх джерел

На циклограмі зображено процес повного циклу збору даних із зовнішніх пристроїв з урахуванням потоку Thread 4 який здійснює фінальну обробку всіх прийнятих даних. Крім роботи із зовнішніми джерелами, модуль також виконує опитування внутрішніх джерел з періодом 10 мс, циклограма роботи з урахуванням даного циклу наведено на рисунку 3.4.

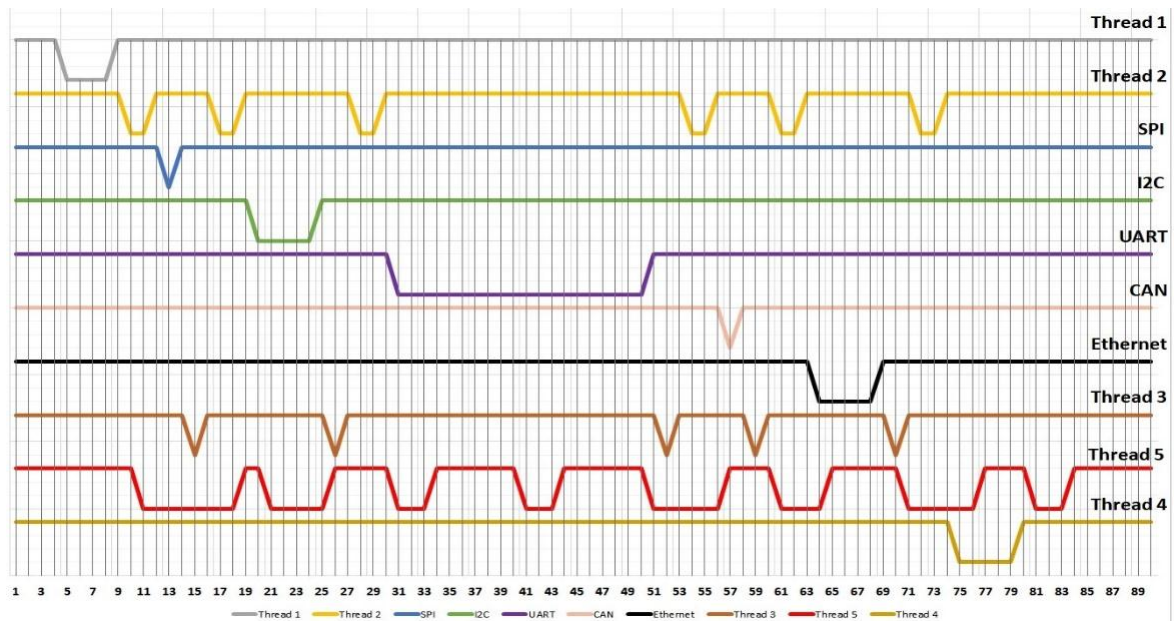


Рисунок 3.4 – Циклограма процесу отримання даних з усіх джерел

На даній циклограмі «Thread 5» є потоком який обслуговує внутрішні інтерфейси отримання даних, так як пріоритет даного потоку нижче пріоритетів потоку опитування зовнішніх джерел, то на циклограмі видно що хоч і період виклику потоку набагато вище час його виконання завжди різний це пов'язано з тим що пріоритет даного потоку нижче та його виконання відбувається у проміжках між отриманням даних із зовнішніх джерел [32].

Так само під час опитування зовнішніх джерел існує ймовірність, що мультимайстерні інтерфейси Ethernet або CAN будуть зайняті, циклограма роботи потоків в даному режимі наведена на рисунку 3.5.

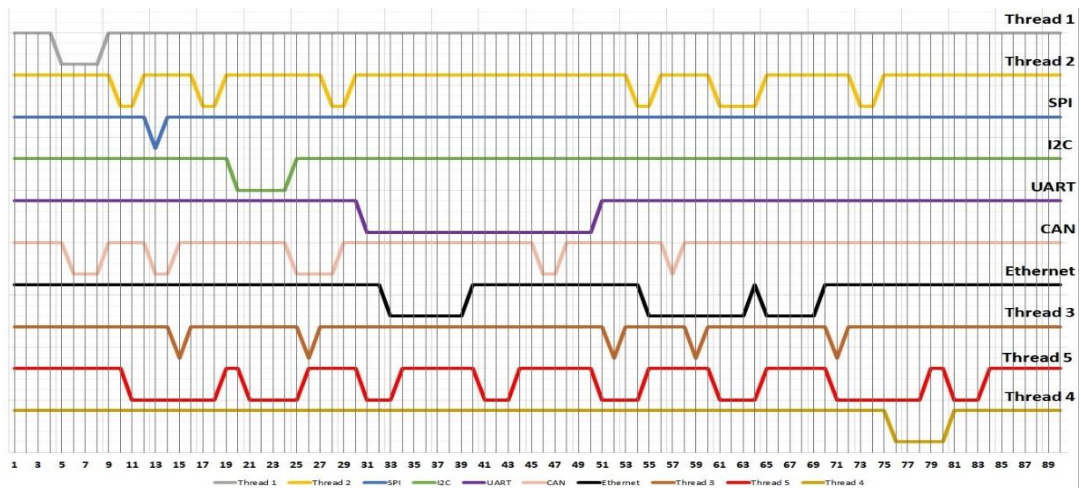


Рисунок 3.5 – Циклограма роботи модуля у разі паралельної роботи CAN та Ethernet

На цій циклограмі видно, що інтерфейси CAN і Ethernet під час запуску процедури збору даних знаходяться в режимі роботи, але CAN встиг переключитися в режим майстра і підготуватися до процедури обміну на той час як інтерфейс Ethernet знаходиться у зайнятому стані.

Тому потік формування запитів залишився в стані очікування звільнення ресурсу і після отримання доступу одразу розпочав обмін. Загальний вид роботи потоків наведено на рисунку 3.6.

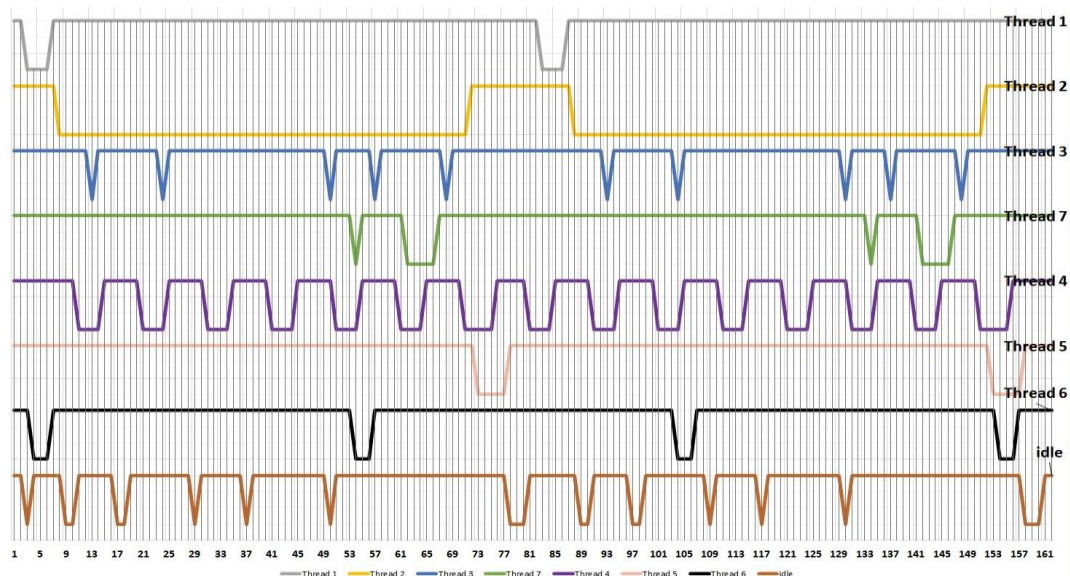


Рисунок 3.6 – Загальний вид роботи потоків

На рисунку 3.6 зображено узагальнену діаграму роботи потоків управління модулем. На діаграмі зображено вісім потоків управління, перший потік формує стек запитів до ведених пристроїв, другий, це узагальнений потік процедури обміну даними, третій потік виконує початкову обробку даних, четвертий виконує збір даних з внутрішній інтерфейсів, п'ятий потік здійснює фінальну обробку даних та їх підготовку для відправки АСУ, шостий потік це обробник запитів від АСУ, сьомий потік виконує обробку інтерфейсів Ethernet і CAN, потік idle займається формуванням статусів поточного стану модуля.

3.3.2 Моделювання обміну за інтерфейсами по 128 байт

Перед початком моделювання роботи потоків обміну, по аналогії за обміном по 32 байти, виконаємо розрахунки параметрів транзакцій обміну та періоду виклику потоку отримання даних. Результати розрахунків транзакцій обміну наведені в таблиці 3.2.

Таблиця 3.2 – Дані для моделювання роботи потоків у разі обміну по 128 байт

Інтерфейс	BaudRate, Мбіт/сек	$T_{\text{біт}}$, мкс	$T_{\text{байт}}$, мкс	$W_{\text{запит}}$, байт	$T_{\text{запит}}$, мс	$W_{\text{відповідь}}$, байт	$T_{\text{відповідь}}$, мс	$T_{\text{обміну}}$, мс
SPI	25	0,04	0,32	133	0,05			
I2C	0,1	10	1,1	5	0,55	128	14,1	15,6
UART	0,115	8,7	95,5	8	3,5	136	58,4	61,9
CAN	1	1	8	128	2,1	128	2,14	4,3
Ethernet	10	0,1	0,8	1526	1,2	1526	1,2	2,4

Розрахувавши величини транзакцій обміну та підставивши їх до формули (3.15), отримаємо період запуску потоків обміну з пристроями який рівняється 132 мс, але для кратності, прирівняємо цю величину рівною 140 мс. В результаті підстановки даних отримаємо циклограму роботи потоків обміну даними яка представлена на рисунку 3.7.

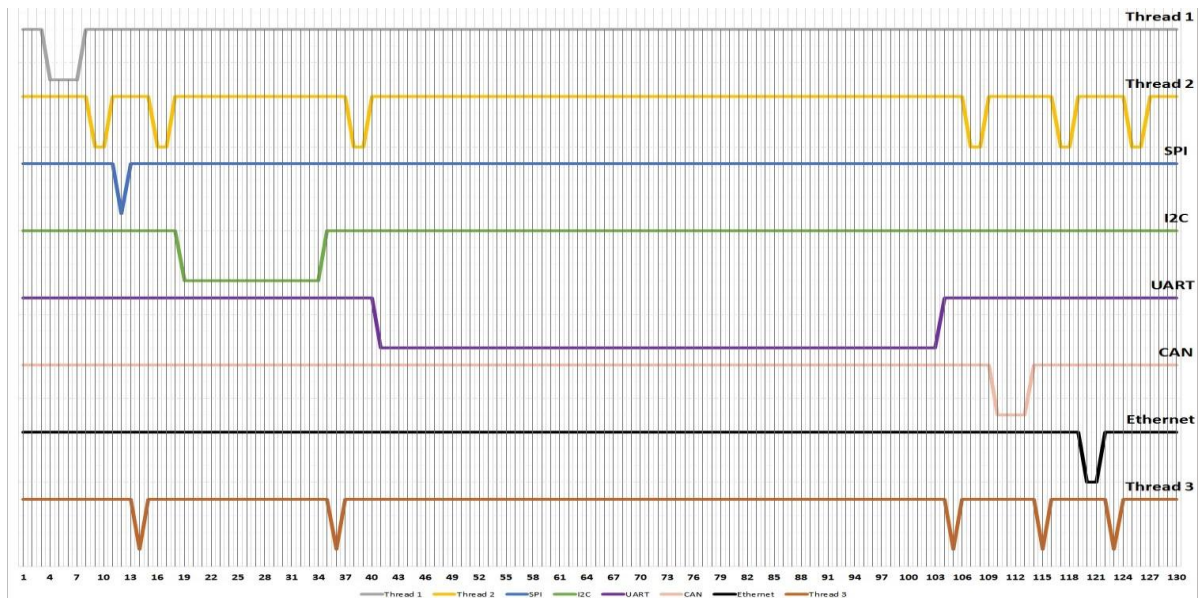


Рисунок 3.7 – Циклограма обміну с зовнішніми пристроями

Як видно з циклограми в результаті збільшення кількості байт даних при обміну з 32 до 128, істотно збільшився час сумарної транзакції обміну, при чому час формування стеку запитів та обробки відповіді залишився не змінним. Також можна відмітити що час обміну за інтерфейсами SPI, CAN та Ethernet також практично не збільшився. Повна циклограма обміну даними с зовнішніми джерелами представлена на рисунку 3.8.

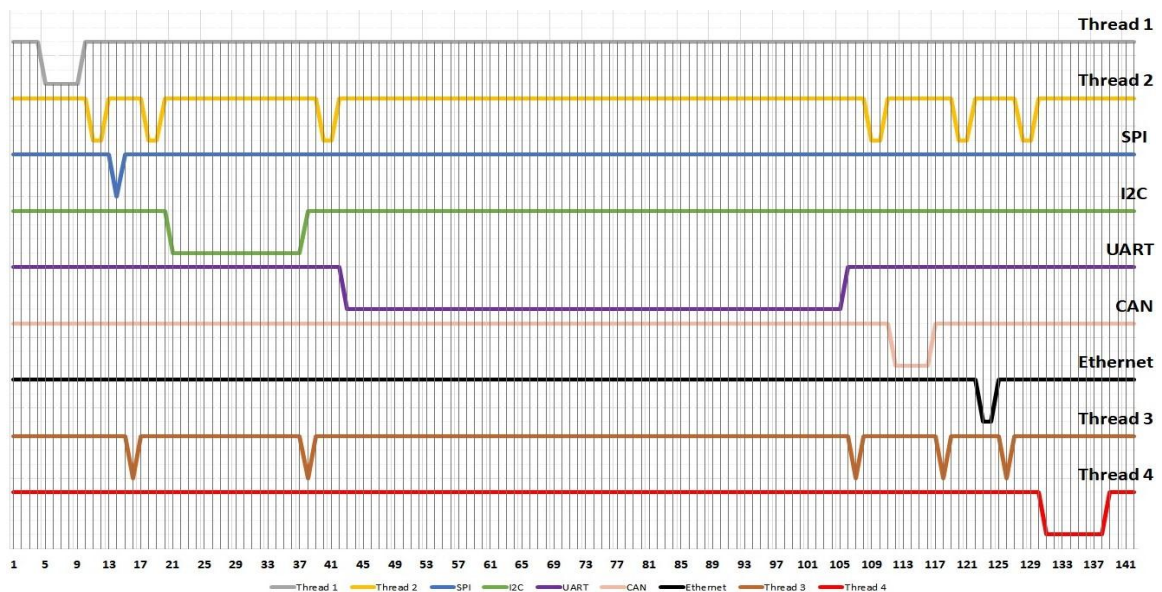


Рисунок 3.8 – Циклограма отримання та обробки даних від зовнішніх джерел

Згідно з даними циклограми видно що окрім збільшення часу обміну між приладами, також і збільшився час фінальної обробки даних, а також можна відмітити що сумарний час отримання даних став рівним 135 мс при періоді запуску потоку 140 мс, що означає що вільного часу для інших інтерфейсів залишилося всього 5 мс. Циклограма процесу повного отримання даних від усіх джерел представлена на рисунку 3.9.

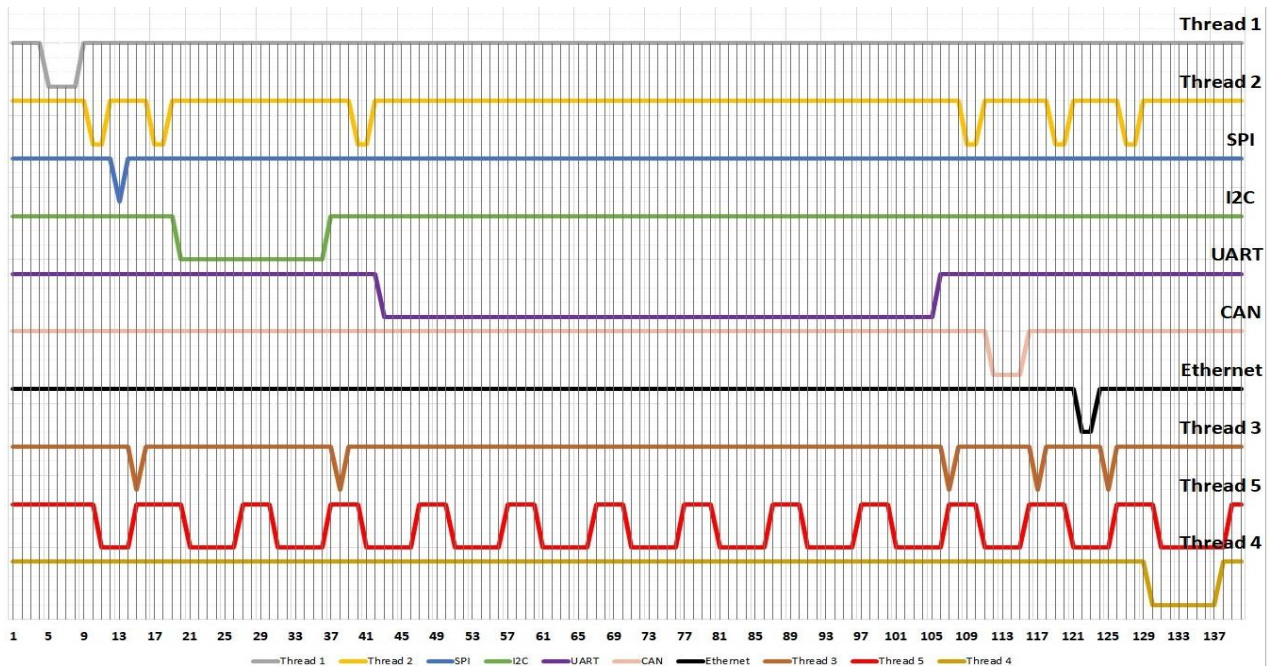


Рисунок 3.9 – Циклограма процесу отримання даних з усіх джерел

Згідно з наведеною циклограмою можна відміти що окрім збільшення сумарного часу отримання та обробки даних збільшився час роботи потоку отримання даних від внутрішніх джерел.

3.3.3 Моделювання обміну за інтерфейсами по 512 байт

Аналогічно з пунктами 3.3.1 та 3.3.2 перед початком моделювання необхідно виконати розрахунки параметрів транзакції та періоду виклику потоку обміну інформацією. Результати розрахунків за формулами (3.1 – 3.12) представлені у таблиці 3.3.

Таблиця 3.3 – Дані для моделювання роботи потоків у разі обміну по 512 байт

Інтерфейс	BaudRate, Мбіт/сек	$T_{\text{біт}}$, мкс	$T_{\text{байт}}$, мкс	$W_{\text{запит}}$, байт	$T_{\text{запит}}$, мс	$W_{\text{відповідь}}$, байт	$T_{\text{відповідь}}$, мс	$T_{\text{обміну}}$, мс
SPI	25	0,04	0,32	517	0,17			
I2C	0,1	10	1,1	5	0,55	512	56,3	58
UART	0,115	8,7	95,5	8	3,5	520	223	227
CAN	1	1	8	512	2,1	512	8,28	17
Ethernet	10	0,1	0,8	1526	1,2	1526	1,2	2,4

Підставивши отримані дані у формулу 3.15, отримаємо що час виклику потоку обміну даними буде дорівнювати 364 мс, з урахуванням кратності іншим потокам, приймемо період виклику рівним 370 мс. Маючи усі данні отримаємо циклограму транзакцій обміну даними с зовнішніми пристроями (рисунок 3.10).

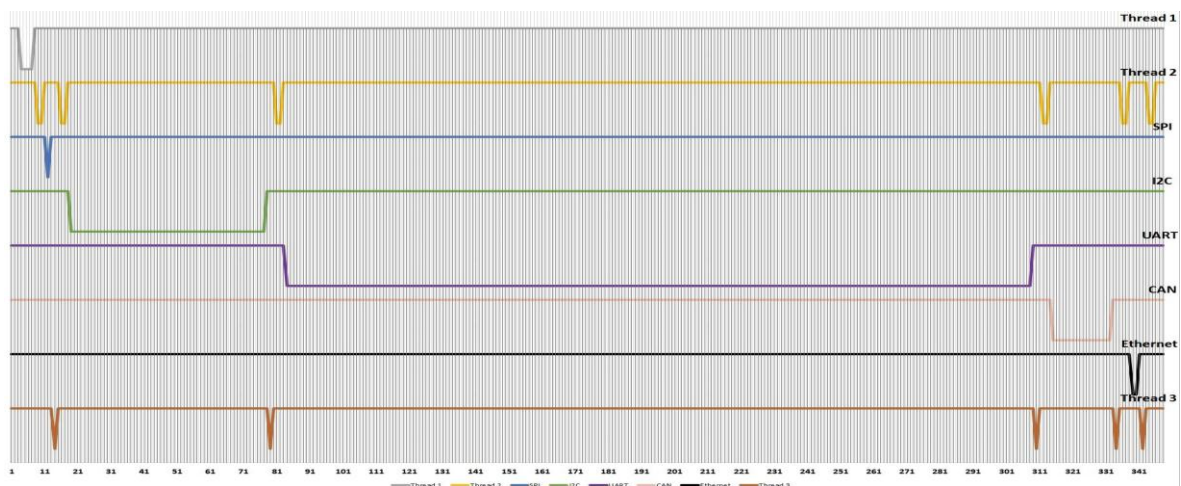


Рисунок 3.10 – Циклограма обміну с зовнішніми пристроями

Аналізуючи отриману циклограму видно, що збільшення кількості байт під час обміну, істотно збільшило сумарний час роботи потоку. Особливу увагу можна приділити інтерфейсам UART та I2C, які в більшій мірі впливають на сумарний час, в той момент як час обміну за інтерфейсом Ethernet і SPI

практично не змінився. На рисунку 3.11 представлена повна циклограма отримання даних від зовнішніх пристроїв з урахуванням обробки отриманої інформації.

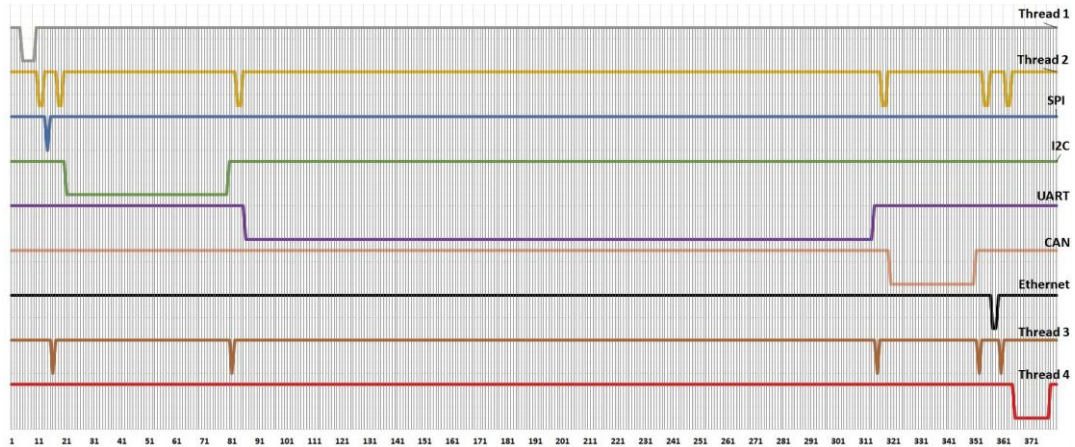


Рисунок 3.11 – Циклограма отримання та обробки даних від зовнішніх джерел

З циклограми видно що аналогічно як і для обміну по 128 байт інформації, час транзакції обміну та обробки отриманої інформації пропорційно збільшився. Також з циклограми видно що сумарний час виконання операцій дорівнює 367 мс, при періоді виклику 370 мс, в такому разі на виконання роботи іншими потоками залишається лише 3 мс. Загальна циклограма отримані усіх даних, як з внутрішніх джерел так і зовнішніх зображена на рисунку 3.12.

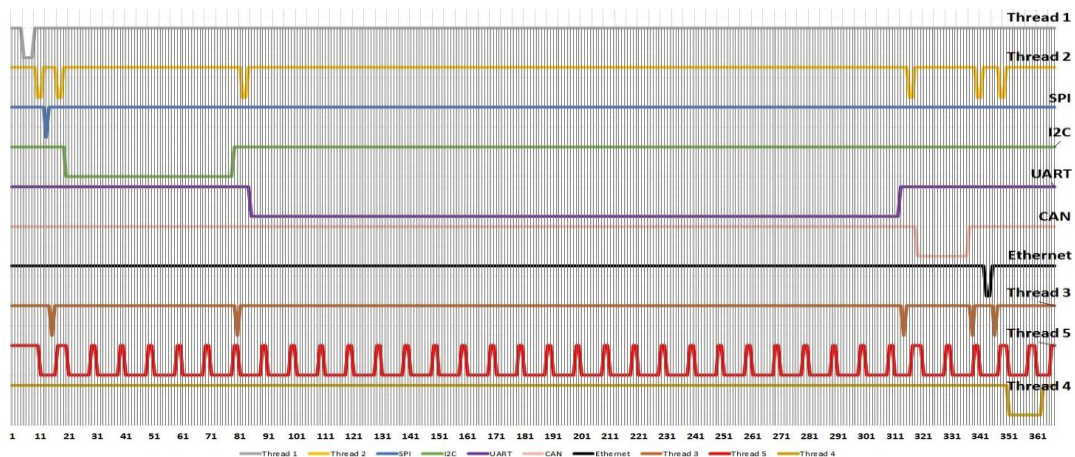


Рисунок 3.12 – Циклограма процесу отримання даних з усіх джерел

3.4 Висновки до розділу 3

В результаті моделювання роботи основних функціональних вузлів модуля комутації цифрових інтерфейсів зв'язку, можна зробити висновок, що для забезпечення коректного і безвідмовного функціонування модуля необхідно враховувати всі нюанси роботи кожного використовуваного інтерфейсу зв'язку і вести обмін даними використовуючи багато потокову обробку. Результати розрахунків при сталих параметрах обміну але зі зміною кількістю байт наведені у таблиці 3.4.

Таблиця 3.4 – Результати розрахунків обміну зі зміною кількістю байт

Інтерфейс	$W_{\text{обмін}}$, байт	$T_{\text{обміну}}$, мс	$T_{\text{виклик}}$, мс	$W_{\text{обмін}}$, байт	$T_{\text{обміну}}$, мс	$T_{\text{виклик}}$, мс	$W_{\text{обмін}}$, байт	$T_{\text{обміну}}$, мс	$T_{\text{виклик}}$, мс
SPI	37	0,012	80	133	0,05	140	517	0,17	370
I2C	37	0,05	80	133	4,1	140	517	15,6	370
UART	48	20,7	80	144	61,9	140	528	227	370
CAN	64	1,22	80	256	4,3	140	1024	17	370
Ethernet	3052	2,4	80	3052	2,4	140	3052	2,4	370

Як видно із розрахунків збільшення кількості байт даних під час транзакцій обміну істотно впливає на період виклику потоків отримання інформації від ведених пристроїв. Також якщо проаналізувати отримані графіки можна відзначити що, збільшення розмірів транзакцій впливає не тільки на час виконання потоку обміну а і на виконання інших потоків в системі, час роботи яких істотно зменшується при сталому періоді виклику і у разі ще більшого розширення транзакції, паралельні потоки почнуть функціонувати не правильно, так як їх час виконання буде більше за час періоду виклику.

Проаналізувавши отримані результати в таблиці можна також відзначити що найбільшого впливу на час виконання потоків в системі мають інтерфейси UART та I2C, внаслідок досить низької швидкості обміну. Тому збільшення кількості даних за одну транзакцію накладає обмеження на роботу системи.

Так як кількість даних під час обміну не є сталою величиною і може

змінюватися то, для вирішення проблеми з накладанням обмежень по кількості байт в транзакції, при сталих параметрах інтерфейсів, вимагає використання потужного обчислювального ядра мікроконтролера, в якого усі інтерфейси підтримуються апаратно, та мають свої виділені канали прямого доступу в пам'ять DMA, використання яких розвантажить процедуру отримання даних, під час якої інші потоки керування отримають контроль над системою.

РОЗДІЛ 4

РОЗРОБКА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕГРОВАНОЇ СИСТЕМИ КОМУТАЦІЇ

Метою даного розділу магістерської кваліфікаційної роботи є вибір основних ключових компонентів модуля на підставі розробленого концепту згідно за результатами моделювання. Відповідно до концепту необхідно визначити наступні компоненти:

- мікроконтролер, для керування модулем;
- реєстри до роботи з дискретними сигналами;
- РНУ для роботи в мережі Ethernet;
- РНУ для роботи в мережі CAN;
- мікросхеми для перетворення UART на RS-485 та RS-422;
- мікросхеми пам'яті для зберігання налаштувань та робочих буферів модуля.

З усіх перелічених компонентів особливі вимоги пред'являються до вибору мікроконтролера, частина яких була отримана в результаті моделювання, крім цього, від вибраного мікроконтролера буде залежати вибір мікросхем пам'яті, РНУ для Ethernet і CAN.

4.1 Вибір мікроконтролера

Мікроконтролер – це спеціальна мікросхема, призначена для керування різними електронними пристроями. Розробники мікроконтролерів вигадали – об'єднати процесор, пам'ять, ПЗУ та периферію всередині одного корпусу, зовні схожого на звичайну мікросхему. З того часу виробництво мікроконтролерів щорічно у багато разів перевищує виробництво процесорів, а потреба в них не знижується.

Мікроконтролери випускають десятки компаній, причому виробляються

не тільки сучасні 32-бітні мікроконтролери, а й 16 і навіть 8-бітні. У середині кожного сімейства часто можна зустріти майже однакові моделі, що відрізняються швидкістю роботи ЦПУ та обсягом пам'яті [33].

Основними компаніями, які займаються розробкою та випуском мікроконтролерів є Microchip що випускає мікроконтролери PIC та AVR, Texas Instruments який базується в основному на випуску мікроконтролерів для DSP обробки та управління перетворювачами частоти, STM32 який є одним з лідерів з випуску мікроконтролерів для систем управління різними типами пристроїв, Siemens та Philips які випускають вузькоспеціалізовані мікроконтролери.

Згідно з результатами моделювання та концепту, до мікроконтролера пред'являються такі вимоги:

- підтримка багато потокової обробки даних;
- частота роботи ядра щонайменше 100 МГц;
- наявність апаратних модулів копіювання даних;
- апаратна підтримка роботи із зовнішніми мікросхемами пам'яті;
- низьке енергоспоживання;
- наявність інтерфейсів UART у кількості щонайменше 5 шт., наявність інтерфейсу SPI у кількості 3 шт., інтерфейс I2C 2 шт., апаратна підтримка Ethernet 2 шт. і CAN також 2 шт.

Виходячи з вимог можна відзначити, що мікроконтролери AVR і PIC не підходять для реалізації модуля, тому що вони не мають достатньої кількості інтерфейсів зв'язку та апаратну підтримку інтерфейсів CAN та Ethernet. Крім фірми Microchip, так само, не підходять і мікроконтролери фірм Siemens і Philips через свою ціну, складну в освоєнні периферію і не стандартні середовища розробки програмного забезпечення.

У зв'язку з цим необхідно проаналізувати два сімейства мікроконтролерів STM32 та Texas Instrument's. Фірма Texas Instrument's виробляє широку лінійку мікроконтролерів, всі вони базуються на двох типах ядер: C2000 та ARM based. Ядро C2000 є власною розробкою фірми, дане ядро розроблено спеціально для

швидкої обробки цифрових сигналів.

Особливістю ядра C2000 є потужна вимірювальна аналогова система, можливість револьверного опитування АЦП, велика кількість багатофункціональних ШІМ таймерів та наявність спеціалізованих інтерфейсів для роботи з енкодерами. Поряд з усіма перевагами дане ядро має слабо розвинену периферію, малу кількість UART-ів, не має багато каналного контролера DMA для кожного інтерфейсу зв'язку.

Якщо розглядати мікроконтролери TI, що базуються на ARM ядрах, то можна помітити, що в основному вони працюють у парі з ядром C2000. Таким чином на одному кристалі знаходиться два практично не залежні ядра. Такий тандем дозволяє розширити функціональні особливості мікроконтролерів фірми TI, оскільки у разі застосування двох ядерного мікроконтролера, стає можливим реалізувати систему управління, наприклад вентиляторами, за допомогою тільки одного мікроконтролера, який розміщено на одній платі з силовими модулями управління.

Такий підхід фірми TI розширює функціональні особливості мікроконтролера, тим самим підвищуючи його конкурентоспроможність на ринку, так як застосування ядер ARM дає можливість апаратної підтримки таких інтерфейсів як CAN, Ethernet, Quad SPI, MSCBSP, eMMC, багатоканальний DMA контролер, які доступні в ядрі C2000 [34]. Приклад архітектури мікроконтролера фірми Texas Instruments моделі TMS320F28388D наведено на рисунку 4.1.

На відміну від Texas Instruments, фірма STM32 займається випуском мікроконтролерів виключно на базі ядер ARM. У зв'язку з цим дані мікроконтролери володіють більш розвиненою периферією на відміну від мікроконтролерів TI, мікроконтролери STM32 мають апаратну підтримку інтерфейсів для роботи з камерами, дисплеями та екранами на базі інтерфесів LTDC, MIPI-DSI, HDMI, крім цього дані мікроконтролери мають спеціалізовані виділені апаратні каналами та буфери для роботи з інтерфейсами Ethernet, CAN, RS-422 та RS-485.

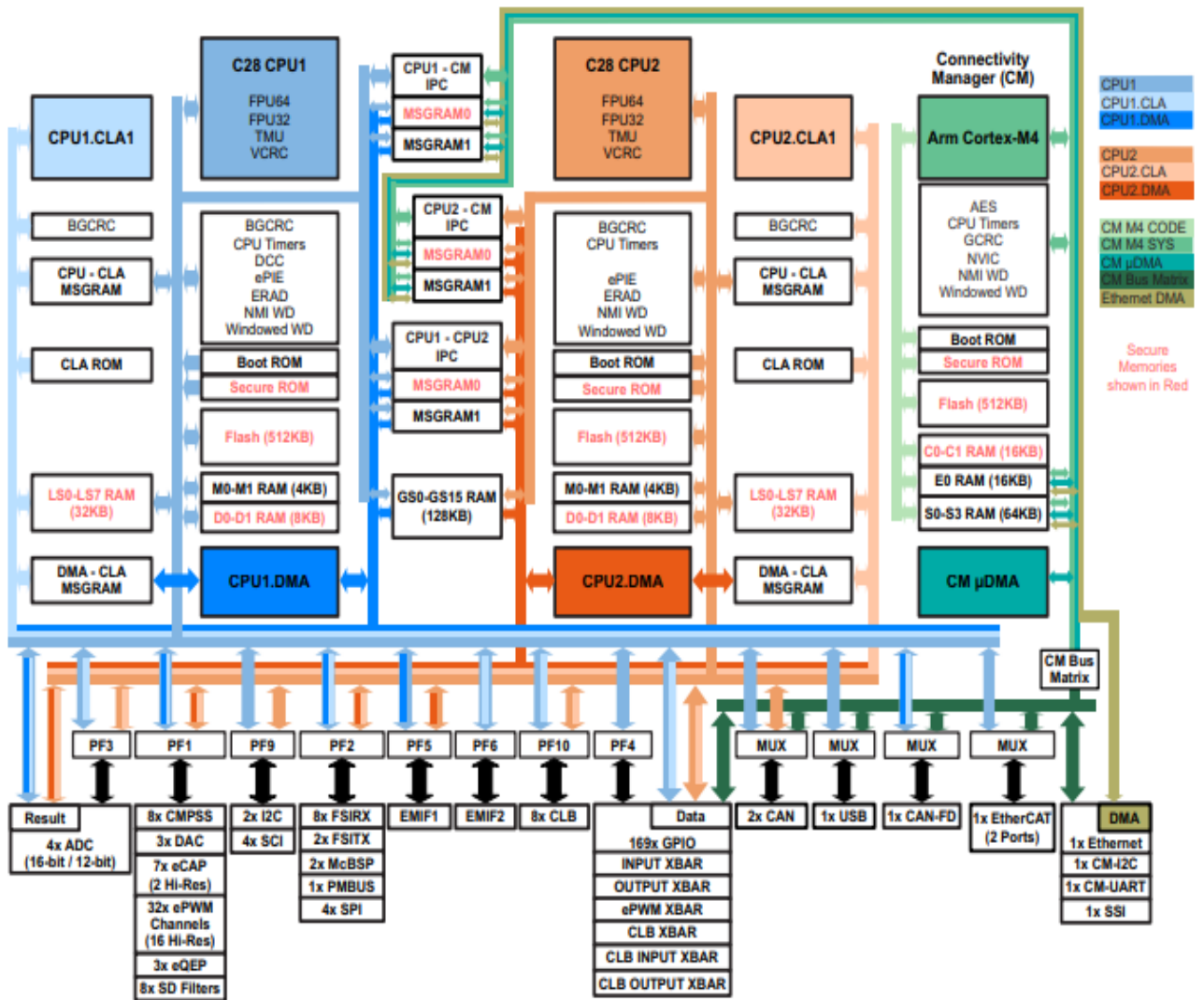


Рисунок 4.1 – Блок схема мікроконтролера з урахуванням ядер C2000 і ARM Cortex-M4 [34]

Окрім цього основною відмінністю будь-якого мікроконтролера фірми STM є швидкість виконання команд з внутрішньої FLASH пам'яті контролера яка не має затримок на відміну від мікроконтролерів фірми TI, у яких код примусово поміщається в ОЗУ для його швидкого виконання [34]. Але, незважаючи на наявність великої кількості інтерфейсів, контролери STM32 значно поступаються мікроконтролерам TI в частині аналогових вимірювань та цифрової обробки сигналів. Приклад архітектури мікроконтролера STM32 наведено на рисунку 4.2.

простою в експлуатації, як модулі управління дискретною логікою будуть застосовані регістри 74НС165 та 74НС595.

4.3 Вибір мікросхем PHY для Ethernet та CAN

PHY – інтегральна схема, необхідна для реалізації функцій фізичного рівня моделі OSI в контролері мережевого інтерфейсу. PHY підключає пристрій каналного рівня (часто званий MAC як акронім від керування доступом до середовища) на фізичний носій, такий як оптичне волокно або мідний кабель. Пристрій PHY зазвичай включає функції як підрівня фізичного кодування, так і рівня, залежного від фізичного середовища. Приклад блок схеми PHY та її складових частин наведено на рисунку 4.4.

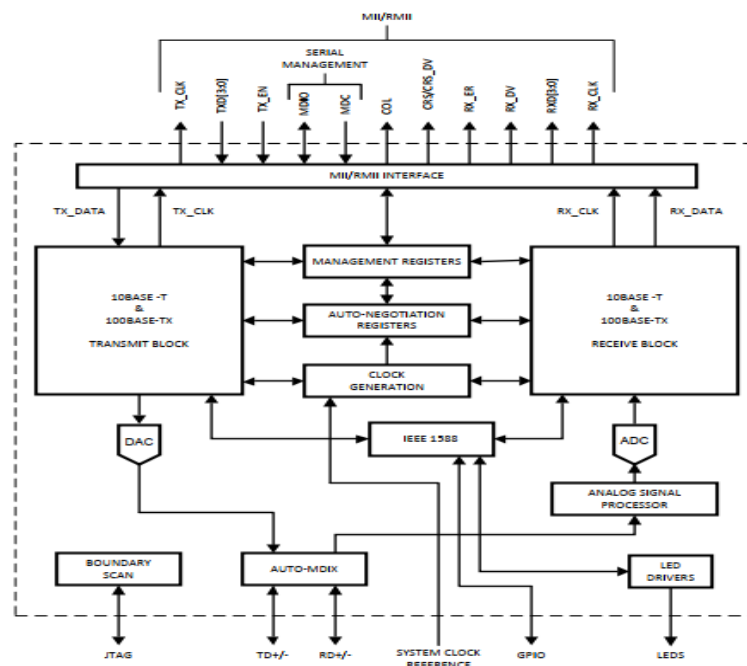


Рисунок 4.4 – Блок схема PHY [3]

В даний момент розрізняють велику кількість мікросхем PHY, дані мікросхеми використовуються практично у всіх високорівневих та високошвидкісних інтерфейсах таких як: USB, IrDA, SATA, WiFi і тому подібних інтерфейсах. Іншими словами, мета мікросхеми PHY забезпечити

перетворення аналогового сигналу інтерфейсу на цифровий [37].

Основними лідерами PHY для мереж Ethernet є фірми Intel, Texas Instruments та Realtek. Мікросхеми фірм Intel та Realtek більше підходять для використання в потужних та високошвидкісних мережах зі швидкостями обміну від 100 Мбіт/с до 1 Гбіт/с, що з погляду промислової мережі є надмірним. Тому для реалізації обміну по мережі Ethernet буде використано мікросхему PHY DP83822 фірми Texas Instruments.

Мікросхеми PHY для CAN інтерфейсів також випускаються великою кількістю фірм, але основною популярністю через якість і надійність користуються мікросхеми CAN PHY фірм Motorola, Microchip і Texas Instruments. Але оскільки інтерфейс CAN у країнах СНД не користується великою популярністю через свою складність, то найвигідніше використовувати мікросхему SN65HVD233-НТ фірми Texas Instruments у зв'язку з її популярністю серед радіоаматорів.

4.4 Вибір мікросхем для роботи інтерфейсів RS-422/RS-485

На відміну від мікросхем PHY для інтерфейсів CAN і Ethernet, мікросхеми перетворення цифрових сигналів інтерфейсу UART в RS-422 і RS-485 є найпопулярнішими, так як ці інтерфейси добре зарекомендували себе в промислових мережах через свою простоту та надійність. Дані мікросхеми за своєю архітектурою є набагато простіші приклад блок схеми перетворювача наведено на рисунку 4.5.

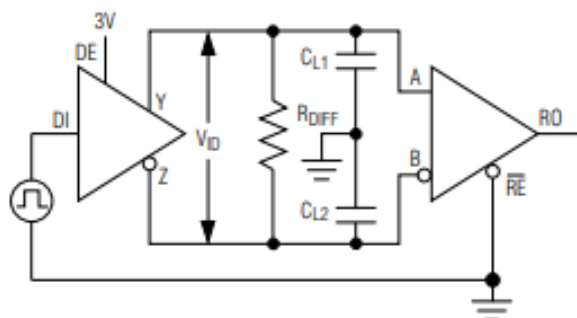


Рисунок 4.5 – Найпростіша схема перетворювача [38]

Основним завданням мікросхеми є перетворення цифрового сигналу UART мікроконтролерів на аналоговий диференціальний сигнал для підключення та передачі даних у мережі [38]. Мікросхеми можуть бути двох типів з автоматичним перемиканням лінії RX/TX або ручним. Без умовним лідером на ринку перетворювачів сигналів UART в RS є фірма MAX electronix, яка випускає перетворювачі всіх можливих типів для сімейства інтерфейсів RS. У зв'язку з цим як перетворювач для RS-485 буде використаний MAX485, а для роботи з RS-422 MAX490.

4.5 Вибір мікросхем пам'яті

Виходячи з концепту роботи модуля, для коректного функціонування, необхідно два типи зовнішньої пам'яті, що підключається: ОЗУ і енергонезалежна пам'ять. ОЗУ необхідно для утримання стека протоколів обміну між пристроями та розміщення буферів з даними, енергонезалежна пам'ять необхідна для зберігання параметрів та налаштувань модуля. На даному етапі розвитку технологій існує три типи пам'яті: RAM, ROM та гібридна пам'ять. Класифікація типів пам'яті наведена на рисунку 4.6 [39].

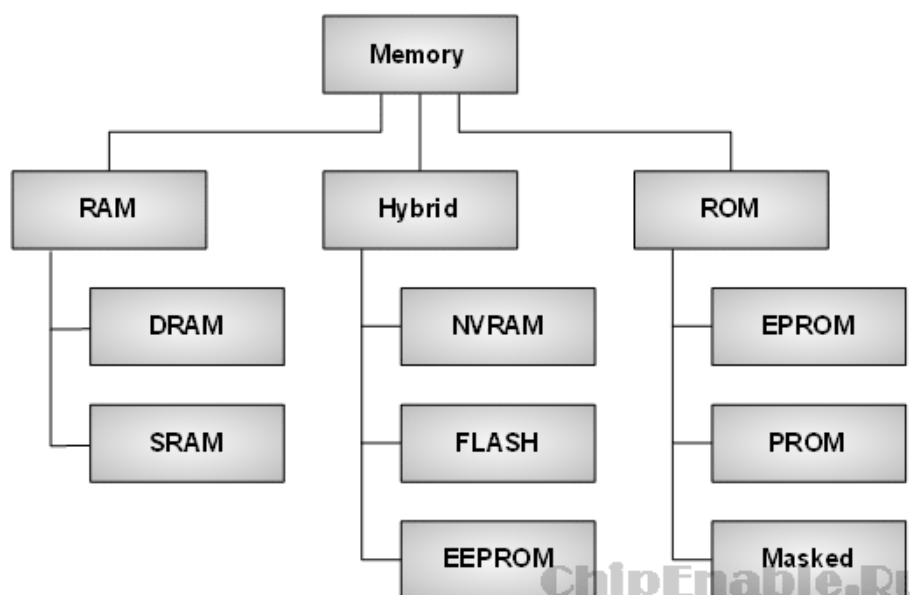


Рисунок 4.6 – Класифікація пам'яті [39]

Сімейство ОЗУ містить два важливі типи пристроїв: статичне ОЗУ (SRAM) і динамічне ОЗУ (DRAM). Головна різниця між ними – це довговічність даних, що зберігаються ними. SRAM зберігає свій вміст до тих пір, поки до мікросхеми подається напруга. Якщо напруга відключена, або тимчасово відсутня, вміст чіпа буде втрачено назавжди. DRAM, з іншого боку, має надзвичайно короткий період тривалості роботи даних – зазвичай близько чотирьох мілісекунд, навіть якщо енергія подається безперервно.

Головною перевагою SRAM перед DRAM є час зберігання інформації, простота роботи та швидкість доступу. Але найголовнішим мінусом SRAM є вартість його розробки.

Типи пам'яті в сімействі ПЗУ характерні методами, які використовуються для запису на них нових даних (зазвичай званих програмними) та числом разів, яке вони можуть бути перезаписані. Загальна особливість всіх цих пристроїв – це їхня здатність зберігати дані програми назавжди, навіть під час перебоїв у живленні.

Найперші ПЗУ були прошитими пристроями, які містили заздалегідь запрограмований набір даних або інструкцій. Вміст ПЗУ повинен був бути визначений до виробництва чіпа, так як фактичні дані могли бути використані для компонування транзисторів усередині мікросхеми. Прошита пам'ять все ще використовується, хоча тепер вона називається масковим ПЗП (masked ROMs), щоб відрізнити її від інших типів пам'яті.

На одну сходинку вище за масочні ПЗУ знаходиться PROM (програмоване ПЗУ), яке купується в незапрограмованому стані. Якщо подивитися на вміст непрограмованих PROM, то видно, що дані складені повністю з одиниць. Процес запису даних на PROM передбачає спеціальне обладнання, зване програмуючим пристроєм. Програмуючий пристрій записує дані на пристрій по одному елементу інформації одночасно, подаючи електричний заряд до вхідних контактів чіпа. Запрограмований якимось таким чином, вміст PROM більше ніколи змінити не вдасться. Якщо код, або дані, що зберігаються в PROM, необхідно змінити, пристрій, що знаходиться в обігу,

доведеться викинути. В результаті PROM або OTP відомі як одноразово програмовані пристрої.

EPROM програмується тим же способом, що і PROM. Однак пристрої EPROM можуть бути очищені та повторно перезаписані. Незважаючи на велику вартість порівняно з PROM, здатність до перепрограмування робить EPROM невід'ємною частиною розробки ПЗ і процесу тестування.

У зв'язку з розвитком технології, останніми роками межа між ОЗУ та ПЗУ стерлася. Тепер деякі типи пам'яті поєднують особливості обох типів. Ці пристрої не належать до жодної з груп і можуть бути віднесені до гібридних пристроїв пам'яті. У гібридну пам'ять можна записувати інформацію і зчитувати її звідти за необхідністю, як і у випадку з ОЗУ, а зберігати її без електроенергії, так само як і в ПЗУ. Два типи гібридних пристроїв – EEPROM та Flash походять від ПЗУ. Їх зазвичай використовують для зберігання коду. Третій гібрид – NVRAM є модифікованою версією SRAM.

EEPROM – це пам'ять, що електрично стирається і програмується. По суті, вона подібна до EPROM пристроїв, але операція стирання відбувається за допомогою електрики, а не завдяки впливу ультрафіолетових променів. Будь-який байт у EEPROM може бути стертий і перезаписаний. Головний компроміс за такою покращеною функціональністю – більш висока вартість.

Флеш-пам'ять (Flash memory) поєднує найкращі риси розглянутих накопичувальних пристроїв, описаних вище. Флеш-пам'ять має високу щільність, низьку ціну, енергонезалежність, швидкість (для читання, але не для запису) та перепрограмування за допомогою електрики. Ці переваги величезні і як прямий результат – використання флеш-пам'яті у вбудованих системах різко збільшилося. З погляду ПЗ, технології флеш та EEPROM дуже схожі. Головна різниця в тому, що флеш-пристрої можуть бути очищені одночасно в межах цілого сектора, а не байт за байтом. Типові розміри сектора перебувають у проміжку від 256 байт до 16 кілобайт. Незважаючи на цей недолік, флеш-пам'ять набагато популярніша, ніж EEPROM і швидко витісняє також багато інших ROM пристроїв.

Третій член у сімействі гібридної пам'яті – NVRAM (енергонезалежна RAM). Енергонезалежність також є характеристикою ROM-пристроїв та гібридних типів пам'яті, описаних вище. Однак NVRAM фізично сильно відрізняється від цих пристроїв. NVRAM – це, як правило, SRAM із резервним акумулятором. Коли живлення увімкнене, NVRAM діє як будь-яка інша SRAM. При відключенні живлення NVRAM бере з батареї стільки енергії, скільки потрібно для збереження даних. NVRAM досить часто зустрічається у вбудованих системах. Однак вона дорого коштує навіть дорожче, ніж SRAM – через наявність батареї, так що її програми зазвичай обмежені здатністю зберігати кілька сотень байт важливої системної інформації, яку не можна зберегти будь-яким іншим шляхом.

Виходячи з виконаного аналізу різних мікросхем, пам'яті можна відзначити, що для реалізації модуля найбільше підходить пам'ять SARAM через швидкість роботи і NVRAM через можливість швидкого запису і зберігання інформації навіть після зняття живлення з модуля [39].

Досліджуючи ринок даних мікросхем, можна відзначити, що всі світові бренди займаються виробництвом і випуском даних мікросхем. У зв'язку з цим, вибір мікросхем пам'яті буде ґрунтуватися на їх популярності серед радіоаматорів, тому в якості мікросхема пам'яті SARAM взята пам'ять IS61WV25616EDBLL фірми Samsung, а як пам'ять NVRAM взята мікросхема FM25V02A.

4.6 Висновки до розділу 4

В результаті проведеного аналізу як основні ключові частини для роботи модуля комутація цифрових і аналогових інтерфейсів зв'язку були вибрані наступні компоненти:

- мікроконтролер – STM32H747AG;
- зсувні регістри для обслуговування дискретних сигналів – 74HC165 та 74HC595;

- РНУ для роботи в мережі Ethernet – DP83822;
- РНУ для роботи в мережі CAN – SN65HVD233-НТ;
- мікросхеми для перетворення UART на RS-485/RS-422 – MAX485 та MAX490;
- мікросхеми пам'яті – SARAM моделі IS61WV25616EDBLL фірми Samsung та NVRAM моделі FM25V02A фірми Cypress.

Для подальшої розробки модуля необхідно виконати побудову архітектури програмного забезпечення яка враховує алгоритми роботи з підключеними пристроями, необхідні методи зберігання інформації, переваги та недоліки пристроїв розглянутих у розділі два.

РОЗДІЛ 5

РОЗРОБКА АЛГОРИТМІЧНО-ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕГРОВАНОЇ СИСТЕМИ КОМУТАЦІЇ

Архітектура програмного забезпечення системи відображає організацію або структуру системи та дає пояснення того, як вона поводить ся. Система є набором компонентів, що виконують певну функцію або набір функцій. Іншими словами, архітектура програмного забезпечення забезпечує міцну основу, на якій може бути побудоване програмне забезпечення.

Ряд архітектурних рішень та компромісів впливають на якість, продуктивність, ремонтпридатність та загальний успіх системи. Нездатність врахувати загальні проблеми та довгострокові наслідки може піддати систему ризику.

Існує безліч архітектурних схем та принципів високого рівня, які зазвичай використовуються в сучасних системах. Їх часто називають архітектурними стилями. Архітектура програмної системи не обмежується одним архітектурним стилем. Натомість, комбінація стилів часто становить повну систему [40].

5.1 Структура зберігання даних підключених пристроїв

Так як модуль комутації цифрових інтерфейсів зв'язку одночасно може працювати з більш ніж десятком пристроїв, які можуть обмінюватися великою кількістю даних, виникають проблеми з їх зберіганням, так як на етапі розробки архітектури зберігання даних їх сумарна кількість не відома. У зв'язку з цим необхідно вводити обмеження на обсяг даних, що передаються і приймаються. Крім того, структури зберігання даних повинні дозволяти здійснювати швидку навігацію та пошук даних на запит від майстер пристрою.

Маючи дані вимоги перед проектуванням структур зберігання даних,

зручно візуалізувати процес руху даних від інтерфейсу до сховища даних. Схема руху даних наведена на рисунку 5.1.

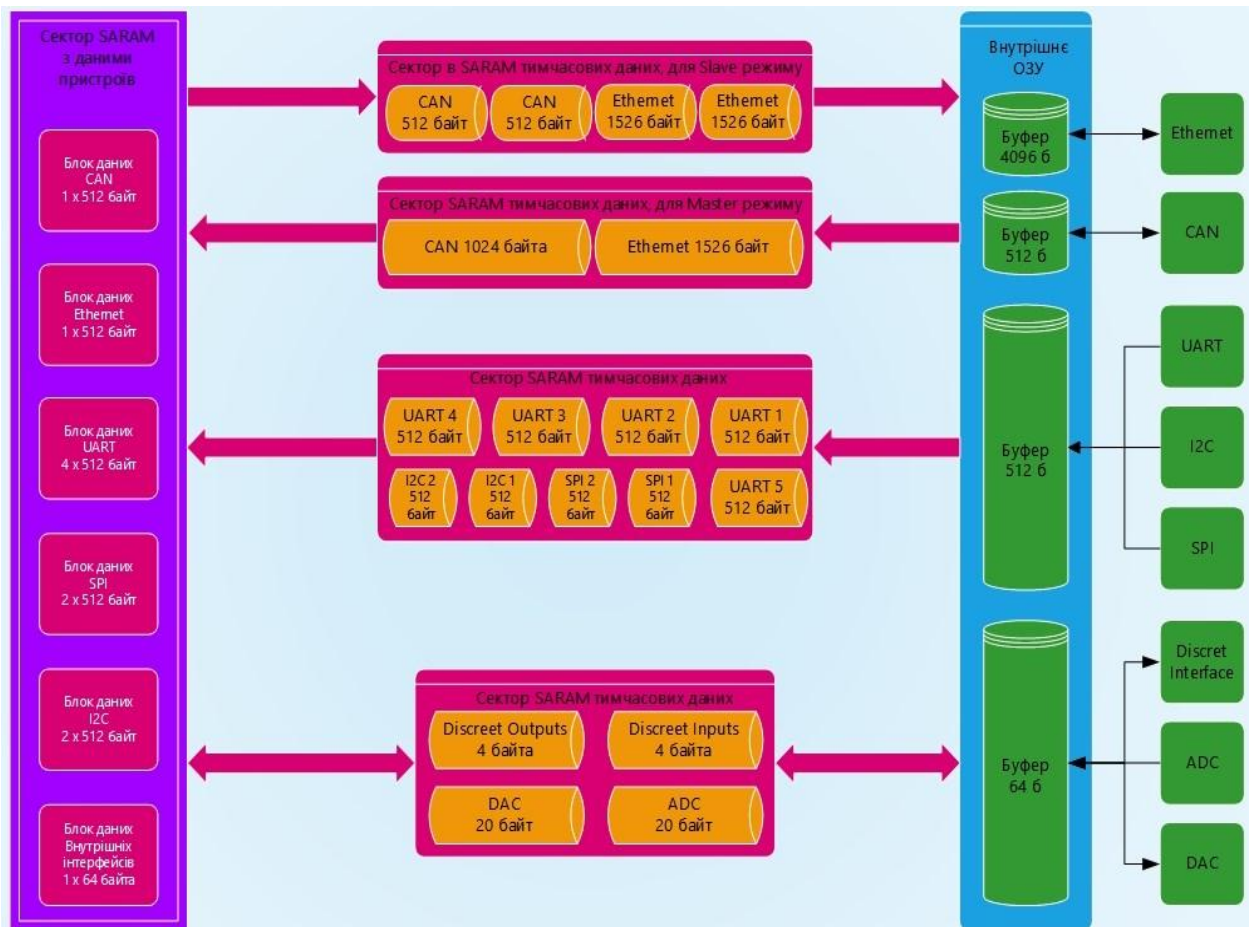


Рисунок 5.1 – Схема руху даних від джерела до сховища

Як видно із схеми для забезпечення збереження даних використовується подвійна буферизація даних, першим буфером зберігання «сирих» даних є буфер інтерфейсу у внутрішньому ОЗУ мікроконтролера, відповідно до схеми зберігання отриманих даних використовується 5184 байта. У цей буфер дані потрапляють за допомогою апаратного модуля прямого доступу до пам'яті DMA, який у мікроконтролерів STM32 виділено під кожен інтерфейс обміну.

Передача даних з буфера приймача інтерфейсу здійснюється потоком, який контролює відповідь від джерела, даний потік здійснює первинну обробку даних і запускає виділений канал DMA передачі даних з буфера приймача в проміжний буфер зберігання інформації [40].

Фінальна упаковка даних буфер зберігання здійснюється потоком, який запускається після повного звільнення стека запитів до підлеглих пристроїв, даний потік виконує фінальну обробку даних під час якої дані переміщуються в основний буфер зберігання. Структури даних та розташування секторів у пам'яті наведено на рисунку 5.2.

```
typedef struct{
    Master_Slave_Data_sector    Ethernet_Can_Sector;    // Сектор даних Ethernet і CAN пристроїв - 9196 байт
    UART_Data_sector           UART_Sector;            // Сектор даних UART пристроїв - 2560 байт
    SPI_Data_sector            SPI_Sector;            // Сектор даних SPI пристроїв - 1024 байт
    I2C_Data_sector            I2C_Sector;            // Сектор даних I2C пристроїв - 1024 байт
    u16                        ADC_buffer[ADC_CHANNEL_CNT]; // Сектор даних ADC - 20 байт
    u16                        DAC_buffer[DAC_CHANNEL_CNT]; // Сектор даних DAC пристроїв - 20 байт
    Discret_Signals_type       discretInputs[SHIFT_REGS_CNT]; // Сектор даних дискретних входів - 2 байти
    Discret_Signals_type       discretOutputs[SHIFT_REGS_CNT]; // Сектор даних дискретних виходів - 2 байти
}SARAM_Data_Sector;
```

Рисунок 5.2 – Загальний вигляд сектора даних пристроїв у пам'яті SRAM

5.2 Алгоритм роботи потоку формування стека запитів

Метою роботи цього потоку є сформувати стек запитів, тобто, сформувати структури на для виконання запитів підлеглим пристроям. Оскільки всі пристрої працюють з різними інтерфейсами зв'язку, які в свою чергу мають різні вказівники на області зберігання даних, структура запиту до пристрою повинна містити всю необхідну інформацію для запуску сеансу обміну, вказівники для копіювання даних, ідентифікатори для визначення стану лінії та вказівники на call-back функції обробки відповідей. Структура даних управління запитом наведена на рисунку 5.3.

```
typedef struct{
    Interface_Id    interface;    // Ідентифікатор інтерфейсу
    void            *pInterfaceBuf;    // Вказівник на буфер інтерфейсу
    void            *pTemporaryBuf;    // Вказівник на тимчасовий буфер
    void            *pStorageBuf;    // Вказівник на буфер зберігання інформації
    void            (*callBack)(void);    // Вказівник на функції обробки відповіді
    void            (*resetInterface)(void);    // Вказівник на функцію сбросу інтерфейсу
    void            (*initForExchange)(void);    // Вказівник на функцію налагодження інтерфейсу перед початком обміну
    Interface_state (*getInterfaceState)(void);    // Вказівник на функцію отримання статусу інтерфейсу
    void            (*sendData)(void *buf, u32 count);    // Вказівник на функцію передачі даних
}Exchange_type;
```

Рисунок 5.3 – Структура запиту пристрою

Заповнення цієї структури відбувається на етапі запуску модуля, на підставі налаштувань користувача, наявності підключених інтерфейсів та інших параметрів, заданих користувачем в результаті конфігурації модуля.

Заповнення стека запитів відбувається так само на підставі налаштувань користувача, наприклад, якщо період запуску опитування з пристроями дорівнює 100 мс, а період отримання даних по інтерфейсу UART 1 необхідний раз на 300 мс, то структура запиту цього інтерфейсу буде додана в стек не кожні 100 мс кожні 300 мс. Результатом успішного формування стека запитів, тобто, коли стек пристроїв не порожній, буде запуск потоку обміну даними з пристроями. Алгоритм формування стека запитів наведено на рисунку 5.4.

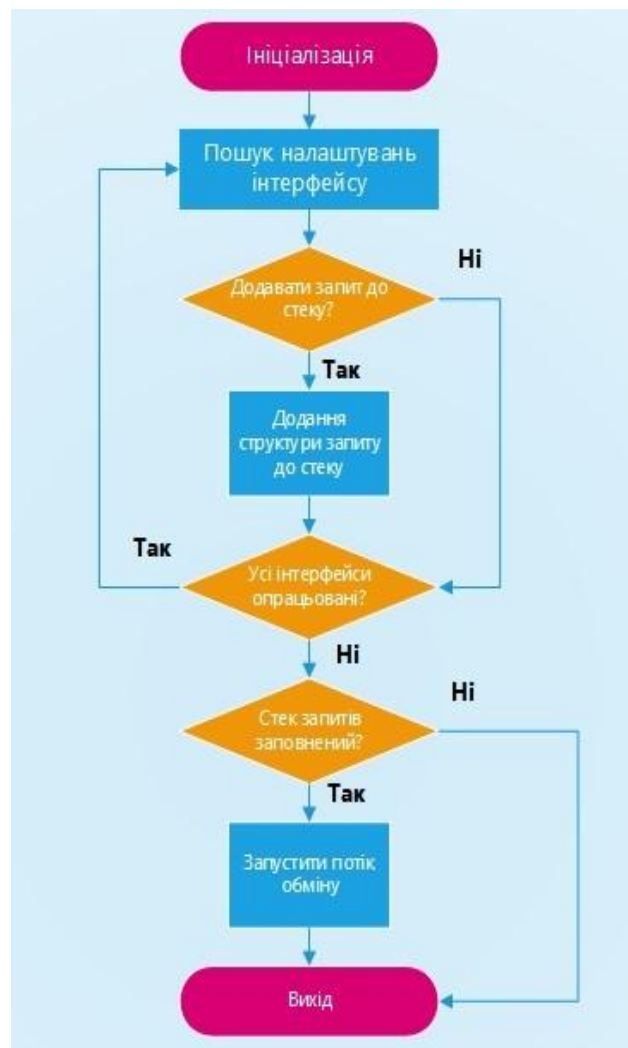


Рисунок 5.4 – Блок схема алгоритму формування стека запитів

5.3 Алгоритм роботи потоку обміну із пристроями

В результаті успішного формування стека запитів потік обміну з пристроями отримає управління над модулем з найвищим пріоритетом серед усіх потоків в системі. Метою даного потоку є формування запитів та контроль отримання відповідей від пристроїв, що знаходяться у стеку. Даний потік щоразу при запуску проводить перевірку стану стека, перевіряє стан лінії зв'язку з інтерфейсом, виконує підготовку інтерфейсу та запускає обмін.

Крім цього, цей потік здійснює контроль над часом виконання обміну по кожному інтерфейсу зв'язку і у випадку, якщо якийсь із інтерфейсів не завершив транзакцію обміну вчасно цей потік перериває процедуру обміну та формує відповідний статус. Алгоритм роботи цього потоку наведено на рисунку 5.5.

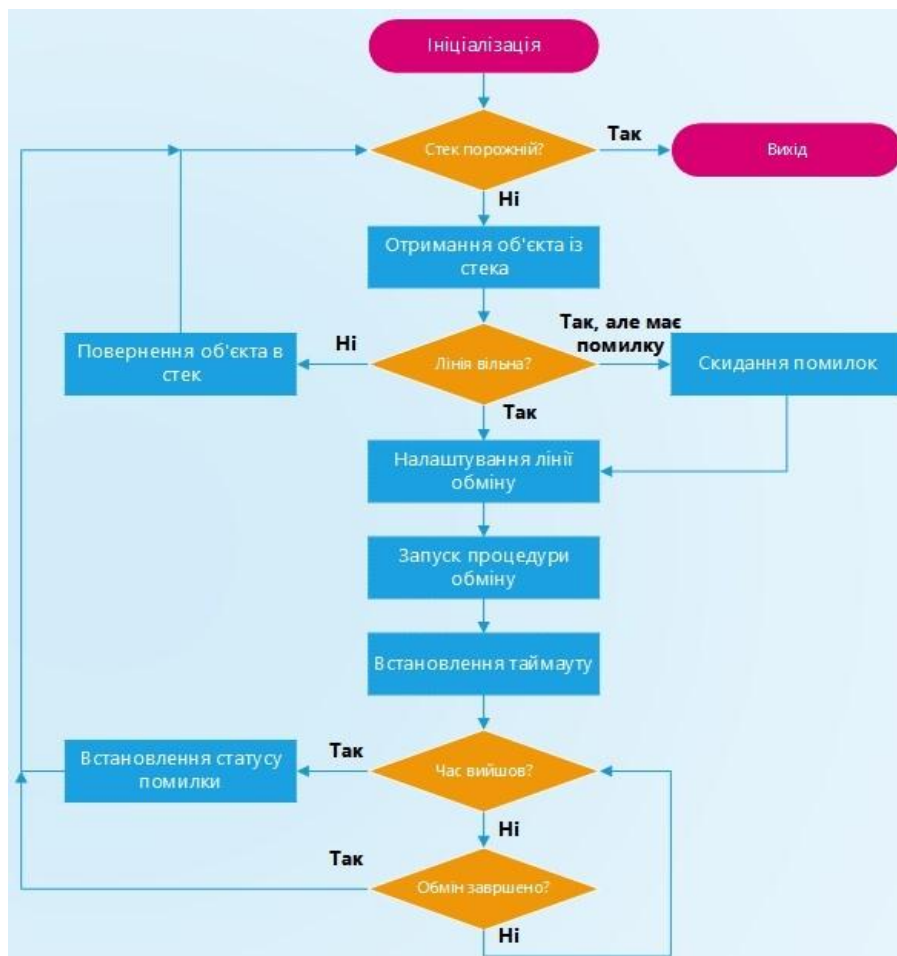


Рисунок 5.5 – Алгоритм роботи потоку обміну із пристроями

Як видно з алгоритму, даний потік виконує контроль обміну з пристроєм на всіх його етапах, цей контроль необхідний і критично важливий з точки зору функціонування пристрою, так як у випадку, якщо один з інтерфейсів зависне під час обміну, або виявиться вимкненою робота системи виявиться порушеною що може призвести до різних наслідків [41].

5.4 Принцип роботи потоку опитування внутрішніх джерел даних

Мета даного потоку – виконання періодичного збору даних з дискретних та аналогових входів, а також встановлення дискретних та аналогових виходів. Опитування аналогових входів відбувається автоматично за рахунок наявності апаратних таймерів ШІМ та каналу DMA, який за сигналом від таймера виконує читання результатів перетворення АЦП.

В результаті запуску цього потоку відбувається фільтрація аналогових сигналів за рахунок усереднення накопиченого значення каналів АЦП, а також фільтрація цифрових дискретних сигналів. Крім цього, даний потік здійснює встановлення значень на дискретних та аналогових виходах модуля, а також готує дані для потоку збору даних, який викликається після завершення роботи потоку обміну даними із зовнішніми пристроями.

5.5 Принцип роботи потоку збору отриманих даних

Метою даного потоку є фінальна обробка та пакування всіх отриманих даних у структури зберігання, а також їх підготовка для відправки в систему управління, до якої підключений модуль. Оскільки під час запуску даного потоку, потік обміну даними із зовнішніми пристроями, який має найвищий пріоритет, завершив своє виконання, даний потік має найвищий пріоритет, що гарантує цілісність зібраних даних.

Під час виконання операцій даним потоком, решта модулів управління системою, крім переривань, знаходяться в призупиненому стані, і в проміжки

часу, коли потік обробки даних виконує операції копіювання, ділять процесорний час порівну. Такий підхід дає можливість розвантажити всі цикли управління модулем та підвищити надійність обробки інформації.

5.6 Охорона праці

Типові інструкції з охорони праці під час експлуатації електроустановок, проведення електричних вимірювань та випробувань розроблені на основі Міжгалузевих правил з охорони праці (правил безпеки) під час експлуатації електроустановок. Ці Типові інструкції призначені для підготовки інструкцій з охорони праці для всіх категорій працівників, зайнятих технічним обслуговуванням електроустановок, що проводять у них оперативні перемикання, що організовують та виконують будівельні, монтажні, налагоджувальні, ремонтні роботи, випробування та вимірювання. Загальні положення:

- інструкція з охорони праці є документом, який встановлює для працівників вимоги щодо безпечного виконання робіт;
- знання інструкції з охорони праці є обов'язковим для всіх працівників;
- керівник структурного підрозділу зобов'язаний створити на робочому місці умови, що відповідають правилам охорони праці, забезпечити працівників засобами захисту та організувати вивчення ними інструкції [42].

На кожному підприємстві повинні бути розроблені та доведені до відома всього персоналу безпечні маршрути проходження по території підприємства до місця роботи та плани евакуації на випадок пожежі та аварійної ситуації. Кожен працівник зобов'язаний:

- дотримуватись вимог цієї Інструкції;
- негайно повідомляти свого безпосереднього керівника, а за його відсутності керівника про нещасний випадок і про всі помічені ним порушення інструкції, а також про несправності споруд, обладнання та захисних пристроїв;

- утримувати в чистоті та порядку робоче місце та обладнання;
- забезпечувати на своєму робочому місці збереження засобів захисту, інструменту, пристроїв, засобів пожежогасіння та документації з охорони праці;
- за порушення вимог Інструкції працівник несе відповідальність відповідно до чинного законодавства;
- під об'єктом випробувань (вимірювань) слід розуміти один або кілька однотипних об'єктів, що випробовуються (вимірюються) одночасно тим самим засобом випробувань (вимірювань);
- під випробуваннями (вимірюваннями) обладнання слід розуміти випробування (вимірювання) діючих електроустановок, що знаходяться в експлуатації, а також випробування (вимірювання), що здійснюються під час монтажу або ремонту обладнання [43].

Вимоги до персоналу:

- до проведення вимірювань та випробувань електрообладнання допускається персонал, який пройшов спеціальну підготовку та перевірку знань Правил охорони праці (правил безпеки) при експлуатації електроустановок (далі – Правил) комісією, до складу якої включаються фахівці з випробувань обладнання, які мають V групу – в електроустановках напругою понад 1000 В і IV групу – в електроустановках напругою до 1000 В;
- до проведення вимірювань та випробувань електрообладнання допускаються працівники не молодші 18 років, які пройшли попередній термін;
- для виробничого навчання працівнику має бути надано термін, достатній для ознайомлення з обладнанням, апаратурою, оперативними схемами та одночасного вивчення необхідної для цієї посади нормативної та технічної літератури;
- до роботи з електровимірювальними приладами повинні допускатися працівники, які пройшли інструктаж та навчання безпечним методам праці, перевірку знань правил та інструкцій відповідно до посади

стосовно роботи з присвоєнням відповідної групи з електробезпеки і не мають медичних протипоказань;

- працівників, що поєднують професії, навчають та інструктують за правилами безпеки праці в повному обсязі за їхньою основною та сумісною професіями (посадами);

- допуск до самостійної роботи оформляється відповідним розпорядженням структурного підрозділу підприємства;

- кваліфікаційне посвідчення для чергового персоналу під час виконання службових обов'язків може зберігатись у начальника зміни цеху або при собі відповідно до місцевих умов;

- працівники, які не пройшли перевірку знань у встановлений термін, до самостійної роботи не допускаються;

- за порушення правил охорони праці залежно від характеру порушень проводиться позаплановий інструктаж або позачергова перевірка знань;

- виконавець робіт, зайнятий випробуваннями електрообладнання, а також працівники, які проводять випробування одноосібно з використанням стаціонарних випробувальних установок, повинні пройти місячне стажування під контролем досвідченого працівника;

- у разі нещасного випадку працівник зобов'язаний надати першу допомогу потерпілому до прибуття медичного персоналу;

- при нещасному випадку з самим працівником, залежно від тяжкості травми, він звертається за медичною допомогою або сам надає першу допомогу (самопоміч). Про кожен нещасний випадок чи аварію постраждалий чи очевидець зобов'язаний негайно сповістити свого безпосереднього керівника;

- кожен працівник повинен знати місцезнаходження аптечки та вміти нею користуватися;

- працівник, який бере участь у проведенні вимірювань та випробувань електрообладнання, повинен працювати у спецодязі та

застосовувати засоби захисту, що видаються відповідно до діючих галузевих норм [44].

Під час проведення дії з випробування електронних приладів працівнику мають бути безкоштовно видані такі засоби індивідуального захисту:

- комбінезон або костюм бавовняний – на 1 рік;
- рукавиці комбіновані індивідуальні – на 3 міс.;
- каска захисна – на 2 роки;
- калоші діелектричні – чергові;
- рукавички діелектричні – чергові.

При видачі подвійного змінного комплекту спецодягу термін носіння подвоюється. Залежно від характеру робіт та умов їх виконання працівникові тимчасово безкоштовно видаються додатковий спецодяг та захисні засоби для цих умов [45].

5.7 Висновки до розділу 5

Розроблена архітектура програмного забезпечення має гнучку структуру, яка дозволяє модернізувати систему, виконувати її оптимізацію та відповідає розробленому концепту, а також результатам моделювання. Отримана архітектура враховує всі переваги та недоліки розглянутих раніше пристроїв цифрової комутації сигналів.

ВИСНОВКИ

В результаті виконання магістерської кваліфікаційної роботи було виконано розробку модуля комутації цифрових аналогових інтерфейсів зв'язку, що дало підвищення якості роботи системи та дозволило оптимізувати процедуру зв'язку між прикладними приладами системи, за рахунок зменшення кількості точок комутації сигналів, шляхом розробки модулю комутації цифрових та аналогових інтерфейсів зв'язку, який має більш гнучку архітектуру та гарантує точність та дискретність отримання даних.

На підставі проведеного аналізу наявних систем цифрової комутації сигналів та аналізу основних принципів побудови промислових мереж на базі інтерфейсів Ethernet, CAN, RS-485 та RS-422 був побудований концепт модуля, який враховує всі переваги та недоліки наявних систем та розглянутих топологій мереж.

Використовуючи отриманий концепт роботи модуля, була побудована модель, в результаті моделювання якої були отримані вимоги до апаратної та програмної частини модуля

Ґрунтуючись на вибраних мікросхемах та результатах моделювання, була побудована архітектура програмного забезпечення модуля та основні алгоритми роботи, які виконують читання, обробку та пакування даних, отриманих від підключених пристроїв.

Впровадження розробленого модуля можливе в комп'ютерно-інтегрованих автоматизованих системах на виробництві для складальних ліній, так і в навчальних лабораторіях в освітньому процесі на лабораторних практикумах і практичних роботах в рамках спеціальності 151.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ASUlearn [Електронний ресурс] // режим доступу: <https://ASUlearn.ru/view/hShd5EvktZQ.html>. Дата звернення: 16.10.2021. Заголовок з екрану.
2. Bookasutp [Електронний ресурс] // режим доступу: https://bookasutp.ru/Chapter2_2.aspx. Дата звернення: 16.10.2021. Заголовок з екрану.
3. Plcsystems [Електронний ресурс] // режим доступу: <https://plcsystems.by/articles/651>. Дата звернення: 16.10.2021. Заголовок з екрану.
4. ДСТУ 3008-15. Документація. Звіти у сфері науки та техніки. структура та правила оформлення. – Введ. 2015-06-22. – К. Держстандарт України, 2017 – 29 с. 3.
5. Дипломне проектування для студентів усіх форм навчання спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології»: довід. / І.Ш. Невлюдов, А.О. Андрусевич, О.В. Токарева, Г.В. Пономарьова – Київ, 2018. – 320 с.
6. Положення про кваліфікаційну роботу здобувача вищої освіти на другому (магістерському) рівні [Електронний ресурс] : Наказ ХНУРЕ від 06 травня 2021 р. No 143. – Режим доступу : https://nure.ua/wp-content/uploads/Main_Docs_NURE/143-vid-06.05.2021-pro-vvedennja-v-diju-rishennja-vchenoi-radi-universitetu.pdf.
7. Методичні вказівки до підготовки атестаційної роботи бакалавра для студентів усіх форм навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» / Упоряд.: І.Ш. Невлюдов, О.В. Токарева, Г.В. Пономарьова. – Харків: ХНУРЕ, – 2019. – 36 с.
8. А. Гіль Промислові інтерфейси та протоколи передачі даних інтегрованих систем для автоматизованого управління в умовах Industry 4.0

Гіль А., Чала О., Филипенко О. Виробництво & Мехатронні Системи 2021: матеріали V-ої Міжнародної конференції, Харків, 21-22 жовтня 2021 р.: Харків, 2021. С.127-30.

9. Бондаренко Ю. В. Аналіз програмного забезпечення для моделювання та тестування параметрів виробничої лінії / Ю. В. Бондаренко, А. А. Гіль, Є. Ю. Валківська // Збірник студентських наукових статей «Автоматизація та приладобудування» / Ю. В. Бондаренко, А. А. Гіль, Є. Ю. Валківська. – Харків, 2020. – С. 220–223.

10. ControlengRussia [Електронний ресурс] // режим доступу: <https://controlengrussia.com/promy-shlenny-e-seti/protokoly/promy-shlenny-e-shiny-i-setevy-e-standarty>. Дата звернення: 17.10.2021. Заголовок з екрану.

11. Основи побудови промислових мереж автоматики / Борисов А.М.-Челябінськ: Видавничий центр ЮУрГУ, 2012. – 108 с.

12. IPC2U [Електронний ресурс] // режим доступу: <https://ipc2u.ru/articles/prostye-resheniya/otlichiya-interfeysov-rs-232-rs-422-rs-485>. Дата звернення: 17.10.2021. Заголовок з екрану.

13. Журнал «ІСУП» [Електронний ресурс] // режим доступу: <https://isup.ru/articles/3/6017/>. Дата звернення: 17.10.2021. Заголовок з екрану.

14. Bookasutp [Електронний ресурс] // режим доступу: https://bookasutp.ru/Chapter2_9.aspx. Дата звернення: 17.10.2021. Заголовок з екрану.

15. GAW [Електронний ресурс] // режим доступу: <http://www.gaw.ru/html.cgi/txt/interface/can/start.htm>. Дата звернення: 17.10.2021. Заголовок з екрану.

16. IPC2u [Електронний ресурс] // режим доступу: <https://ipc2u.ru/articles/prostye-resheniya/modbus-rtu/>. Дата звернення: 17.10.2021. Заголовок з екрану.

17. SIMENS [Електронний ресурс] // режим доступу: <https://www.siemens->

pro.ru/docs/kip/Flowmeters/Others/Profibus/1_Network_PROFIBUS.htm/ Дата
звернення: 17.10.2021. Заголовок з екрану.

18. MicroSin [Електронний ресурс] // режим доступу:
<http://microsin.net/programming/arm/canopen-overview.html>. Дата звернення:
19.10.2021. Заголовок з екрану.

19. Siemens-pro [Електронний ресурс] // режим доступу:
<https://www.siemens-pro.ru/promo/scalance-promo.html>. Дата звернення:
19.10.2021. Заголовок з екрану.

20. EP [Електронний ресурс] // режим доступу
http://www.ep.ru/product/katalogs/ABB/20_Avtomatic.pdf. Дата звернення:
19.10.2021. Заголовок з екрану.

21. HubStub [Електронний ресурс] // режим доступу
<https://hubstub.ru/programming/62-osnovy-raboty-protokola-spi.html>. Дата
звернення: 24.10.2021. Заголовок з екрану.

22. Icard [Електронний ресурс] // режим доступу
<https://www.lcard.ru/lexicon/adc>. Дата звернення: 24.10.2021. Заголовок з екрану.

23. Electrical School [Електронний ресурс] // режим доступу
<http://electricalschool.info/electronica/1759-shirotno-impulsnaja-moduljacija.html>.
Дата звернення: 24.10.2021. Заголовок з екрану.

24. EasyElectronics [Електронний ресурс] // режим доступу
<http://easyelectronics.ru/interface-bus-iic-i2c.html>. Дата звернення: 24.10.2021.
Заголовок з екрану.

25. 3D-diy [Електронний ресурс] // режим доступу [https://3d-
diy.ru/wiki/arduino-moduli/interfeys-peredachi-dannykh-uart/](https://3d-diy.ru/wiki/arduino-moduli/interfeys-peredachi-dannykh-uart/). Дата звернення:
24.10.2021. Заголовок з екрану.

26. Future to day [Електронний ресурс] // режим доступу
<https://future2day.ru/mikrokontroller/>. Дата звернення: 24.10.2021. Заголовок з
екрану.

27. Science Fandom [Електронний ресурс] // режим доступу https://science.fandom.com/ru/wiki/Математическая_модель/. Дата звернення: 28.11.2021. Заголовок з екрану.

28. Поняття та категорії [Електронний ресурс] // режим доступу <http://ponjatija.ru/node/582/>. Дата звернення: 28.11.2021. Заголовок з екрану.

29. Круковская Т.Ю. Введення у ітераційні методи. Основи проектування: навчальний посіб. / Круковская Т.Ю. Санкт-Петербург, 2022. – 56 с.

30. Горлач Б. А. Математичне моделювання. Побудова моделей та чисельна реалізація: навчальний посіб. / Горлач Б. А., Шахов В. Г. Москва, 2021. – 292 с.

31. Монаков А.А. Математичне моделювання. Побудова моделей та чисельна реалізація: навчальний посіб. / Горлач Б. А., Шахов В. Г. Москва, 2021. – 292 с.

32. Веремей Е.И. Лінійні системи із зворотним зв'язком. Основи проектування: навчальний посіб. / Круковская Т.Ю. Санкт-Петербург, 2021. – 489 с.

33. Electronoff [Електронний ресурс] // режим доступу <https://electronoff.ua/academy/post/chto-takoe-mikrokontrollery-naznachenie-i-ustrojstvo.php/>. Дата звернення: 01.12.2021. Заголовок з екрану.

34. TI [Електронний ресурс] // режим доступу <https://www.ti.com/applications/industrial/factory-automation/overview.html>. Дата звернення: 01.12.2021. Заголовок з екрану.

35. STM electronics [Електронний ресурс] // режим доступу https://www.st.com/resource/en/reference_manual/. Дата звернення: 01.12.2021. Заголовок з екрану

36. Easyelectronics [Електронний ресурс] // режим доступу <http://easyelectronics.ru/sdvigovuj-registr.html/>. Дата звернення: 01.12.2021. Заголовок з екрану.

37. WiKi board [Електронний ресурс] // режим доступу <https://wikiboard.ru/wiki/PHY/>. Дата звернення: 01.12.2021. Заголовок з екрану.
38. Analog [Електронний ресурс] // режим доступу <https://www.analog.com/ru/products/interface-isolation/rs-232-rs-422-rs-485-rs-562.html>. Дата звернення: 02.12.2021. Заголовок з екрану.
39. Chip Enable [Електронний ресурс] // режим доступу <https://chipenable.ru/index.php/how-connection/57-tipy-pamyati-vstraivaemyh-sistem.html>. Дата звернення: 02.12.2021. Заголовок з екрану.
40. Datascience [Електронний ресурс] // режим доступу <https://datascience.eu/ru/компьютерное-зрение/архитектура-программного-обеспечени/>. Дата звернення: 02.12.2021. Заголовок з екрану.
41. Habr [Електронний ресурс] // режим доступу <https://habr.com/ru/post/276593/>. Дата звернення: 02.12.2021. Заголовок з екрану.
42. Управління Держпраці у Тернопільській області [Електронний ресурс] // <https://te.dsp.gov.ua/ohorona-pratsi-na-pidpryyemstvi-shho-potribno-znaty/>. Дата звернення: 02.12.2021. Заголовок з екрану.
43. Верховна Рада України [Електронний ресурс] // режим доступу <https://zakon.rada.gov.ua/laws/show/z0508-18#Text/>. Дата звернення: 02.12.2021. Заголовок з екрану.
44. ДНАОП [Електронний ресурс] // режим доступу <https://dnaop.com/html/31965/doc-instrukcijaz-ohoroni-pracipid-chas-montazhu-vmorinnih-merezh/>. Дата звернення: 02.12.2021. Заголовок з екрану.
45. Інструкції для навчальних закладів України [Електронний ресурс] // режим доступу <https://osvita-docs.com/node/41/>. Дата звернення: 02.12.2021. Заголовок з екрану.