

УДК 681.532.52

Я.А. Жабура, К.О. Беліков, О.В. Левченко

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», м. Київ, Україна

Реалізація комунікації між частотним перетворювачем і контролером за протоколом Movilink

В промисловій автоматизації широко використовується послідовна комунікація, що відносно просто реалізується і потребує меншої кількості фізичних пристроїв в порівнянні з паралельними інтерфейсами [1].

Протоколи комунікації в рамках моделі OSI поділяються на сім рівнів: фізичний, каналний, мережевий, транспортний, сеансовий, представлення і прикладний [2].

Найнижчим є фізичний рівень, що описує кабелі, роз'єми, топологію мережі, швидкість передачі даних, синхронізацію, фізичне представлення бітів сигналу. Канальний рівень описує поділ сигналів на кадри (груп бітів певного розміру, що включають початкові і кінцеві символи) і обробку помилок, пов'язаних з некоректною передачею кадрів. Мережевий рівень описує формування пакетів даних, адресацію пакетів в мережі, визначення найкоротших маршрутів, комутацію і маршрутизацію пакетів. Транспортний рівень описує передачу повідомлення в цілому.

Сеансовий рівень забезпечує започаткування, підтримання, синхронізацію і завершення зв'язку між пристроями. Рівень представлення описує синтаксис, значення повідомлень, кодування і стиснення даних. Прикладний рівень описує інтерфейс взаємодії користувача з процесом передачі даних.

Завданням роботи є забезпечити взаємодію програмованого логічного контролера (ПЛК) ОВЕН ПЛК100-24.Р-М та частотного перетворювача SEW MC07B0011-5A3-4-00/FSC11B (рис. 1). Протокол фізичного рівня RS-485 підтримується, як обраним ПЛК, так і частотним перетворювачем. При цьому частотний перетворювач в поєднанні з протоколом RS-485 підтримує тільки протокол рівня представлення Movilink, що не підтримується цим ПЛК.

Можливими рішеннями цієї задачі було би використання ПЛК компанії SEW, що підтримують цей протокол, або перетворювачів інтерфейсу, таких як [3]. Крім цього, можна використовувати дискретні і аналогові виходи ПЛК для

комунікації з частотним перетворювачем, але цей метод вимагає підключення модуля розширення аналогових виходів ПЛК, а також не дозволяє отримувати частину інформації від частотного перетворювача, наприклад, значення струму.

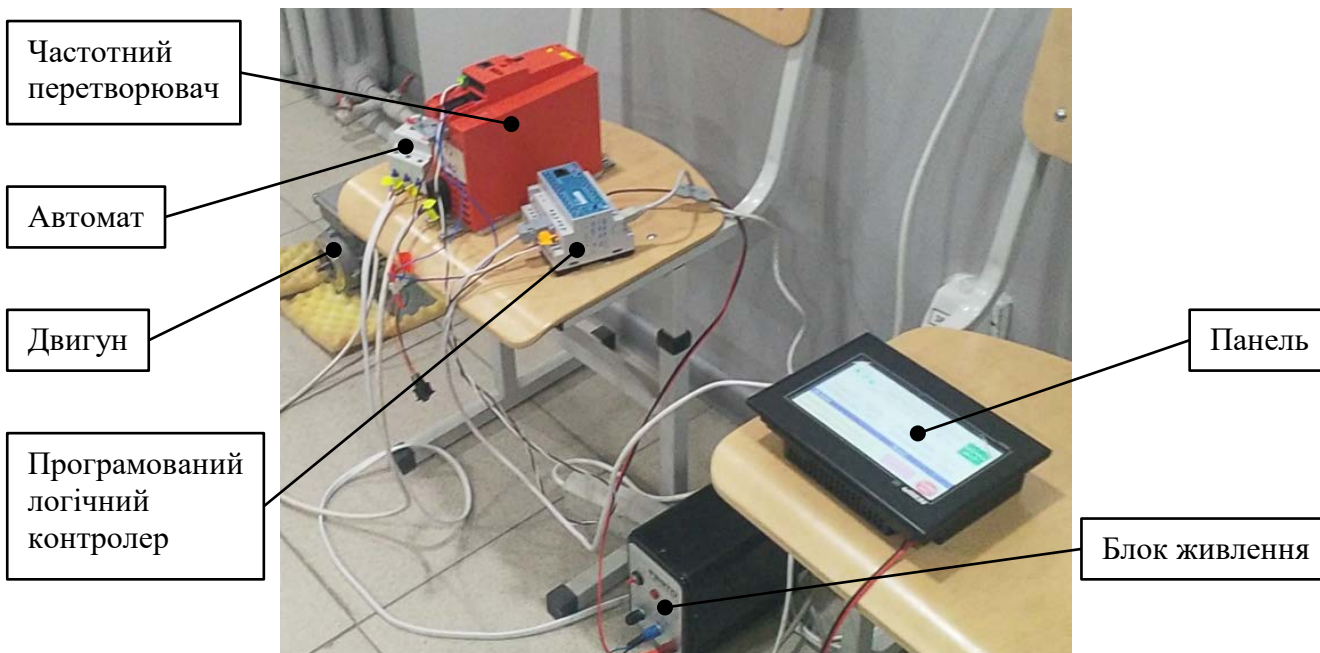


Рис. 1. Підключення частотного перетворювача до ПЛК.

331

Рішенням, що буде розглянуто в цій статті, є реалізація протоколу Movilink за допомогою бібліотеки UNM.lib [4], що дозволяє напряму працювати з протоколами фізичного рівня, такими як RS-485 та RS-232. Схожий функціонал мають бібліотеки SysLibCom.lib, призначена для роботи з CoDeSys v2, як і UNM.lib. Для роботи з CoDeSys v3 існує схожа бібліотека OwenCommunication.

Протокол Movilink описує таку структуру повідомлення:

Idle... [SD] [ADR] [TYP] [PDU] [BCC] ...

В цій структурі Idle це затримка між повідомленнями, що забезпечується бібліотекою автоматично. Перший байт повідомлення [SD] є стартовим символом, що визначає напрямок повідомлення, байт [ADR] містить в собі адресу частотного перетворювача.

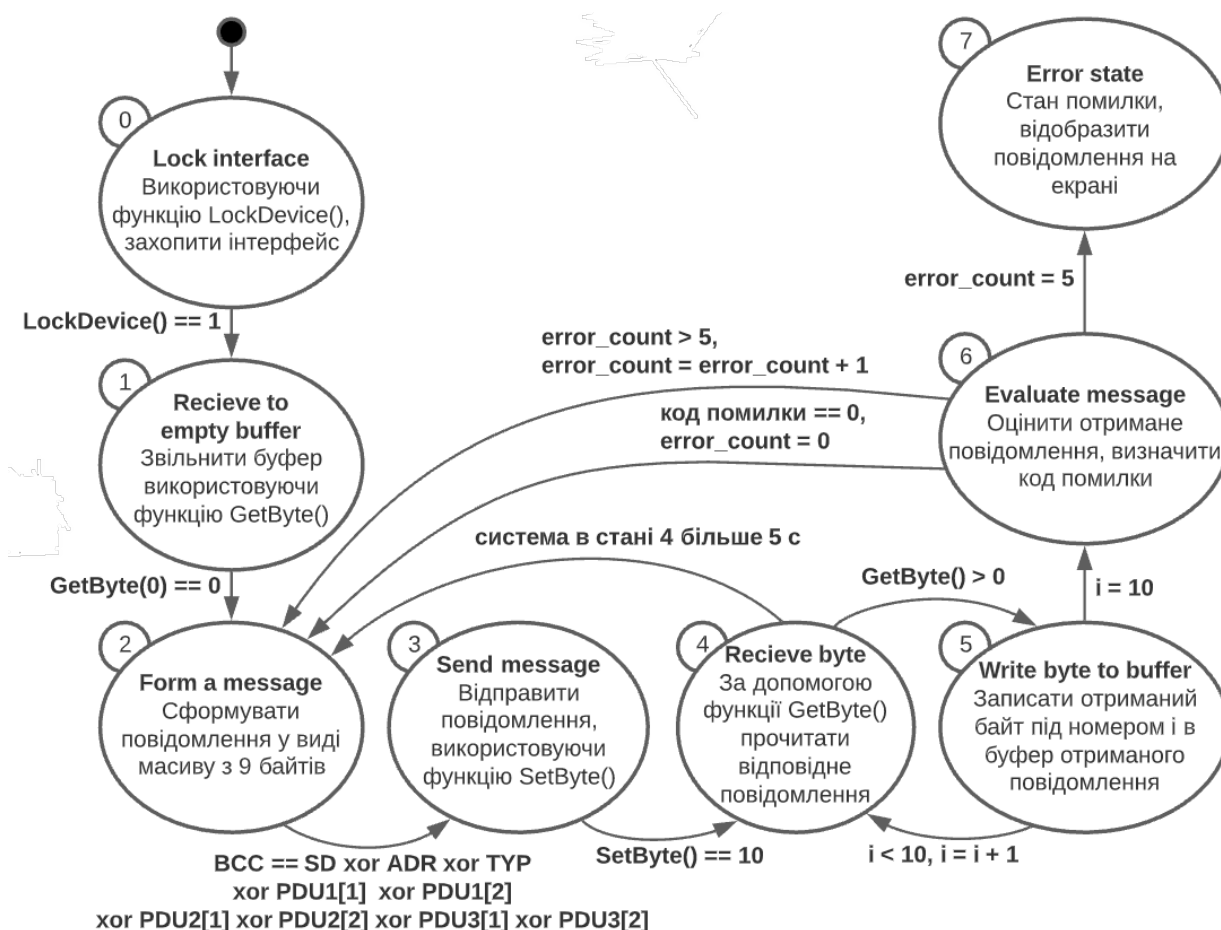
Блок [PDU] називається протокольним блоком даних. Він може мати розмір від двох до чотирнадцяти байт і містить в собі інформацію, що передається між ПЛК і частотним перетворювачем. Розглянемо варіант передачі, в якому передаються три значення параметрів (шість байт). Тип протокового блоку даних позначається байтом [TYP].

Останній байт повідомлення [BCC] є контрольним байтом, що розраховується як виняткова диз'юнкція решти байтів повідомлення.

На рис. 2. показано схему скінченного автомата, що складається з восьми станів, і дозволяє реалізувати комунікацію між ПЛК і частотним перетворювачем по протоколу Movilink.

В стані 0 відбувається захоплення інтерфейсу, що дозволяє бібліотеці UNM працювати з відповідним портом ПЛК, номер якого передається в функцію LockDevice(). У випадку успішного захоплення інтерфейсу, система переходить в стан 1.

Стан 1 відповідає за очищення вхідного буфера. Якщо функція GetByte() повертає нуль (кількість отриманих байт), система переходить в стан 2.



332

Рис. 2. Схема скінченного автомата, що показує запропонований алгоритм комунікації.

В стані 2 формується повідомлення, що в даному випадку складатиметься з трьох службових байтів, трьох слів одиниць протокольних даних і одного

контрольного байта. В стан 3 система переходить, тільки якщо контрольний байт сформовано правильно, що дозволяє забезпечити, що повідомлення було сформоване повністю, навіть якщо його формування займе більше одного скану.

Відправка повідомлення відбувається в стані 3 за допомогою функції SetByte(), що приймає дані типу string. Оскільки масив зберігається в пам'яті неперервно, достатньо створити покажчик на string(10) з адресою масиву, сформованого в стані 2. Якщо кількість відправлених байт, повернена функцією SetByte() дорівнює 10, повідомлення відправлено і система переходить в стан 4.

В стані 4 система отримує відповідне повідомлення від частотного перетворювача за допомогою функції GetByte(), що повертає кількість отриманих байт. Якщо система знаходиться в стані 4 більше п'яти секунд, повідомлення, надіслане в стані 3, не було оброблено частотним перетворювачем і необхідно надіслати його знову. За цією умовою система переходить в стан 3. Якщо функція GetByte() повертає одиницю, система переходить в стан 5.

Отриманий байт записується в буфер отриманого повідомлення в стані 5. Змінна і відповідає номеру отриманого байта в буфері і збільшується на одиницю при кожному переході з 5 стану в 4. Якщо повідомлення повністю записане в буфер, система переходить в стан 6, в іншому випадку переходить в 4 стан.

333

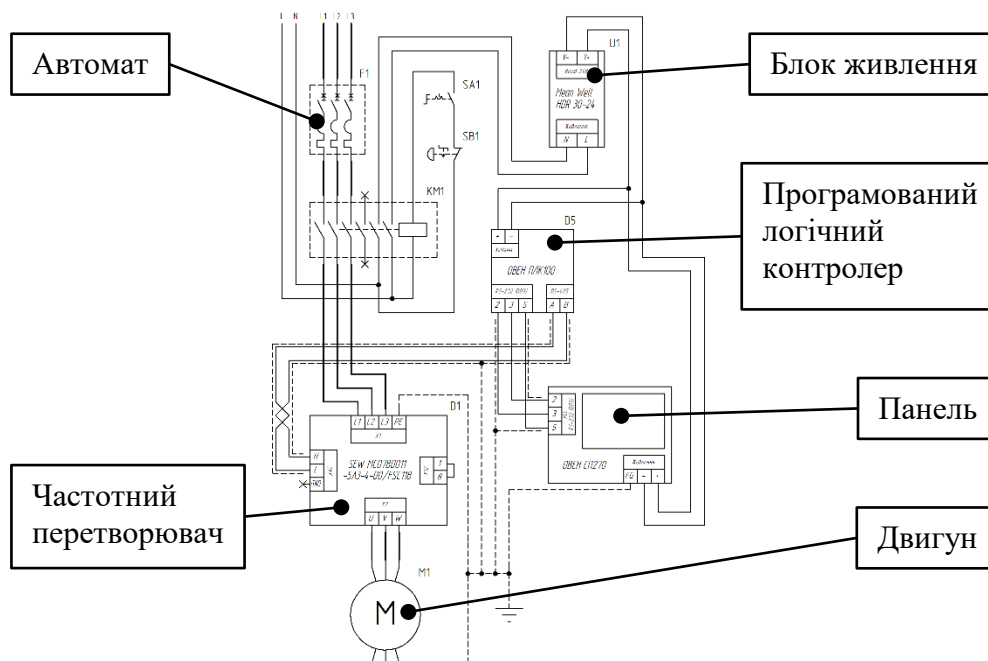


Рис. 3. Електрична схема приладів, що використані для практичної перевірки отриманих результатів.

В стані 6 відбувається оцінка отриманого повідомлення. Якщо отримане повідомлення не містить коду помилки, система переходить в стан 2 для формування наступного повідомлення. Якщо було отримано код помилки, то система надсилає нове повідомлення ще чотири рази, щоразу збільшуючи лічильник помилок. Якщо система отримувала код помилки п'ять разів підряд, то вона переходить в стан 7, в якому відображається повідомлення про помилку.

Запропонований алгоритм було перевірено практично за допомогою ПЛК ОВЕН ПЛК100-24.Р-М та частотного перетворювача SEW MC07B0011-5A3-4-00/FSC11B, підключених, як показано на рис. 3.

На електричній схемі зображено вказані ПЛК і частотний перетворювач. До ПЛК підключена панель для спрощення подачі і отримання сигналів з ПЛК. До частотного перетворювача підключено асинхронний двигун. Захист системи забезпечується автоматом і контактором аварійної зупинки. На рис. 1 зображено фото збірки, що використовувалася для перевірки.

Висновки:

Запропоновано структуру кінцевого автомату, що дозволяє реалізувати комунікацію по протоколу Movilink між обладнанням компанії SEW і ПЛК, що підтримують бібліотеку UNM.lib. В процесі роботи було практично реалізовано цей метод керування частотним перетворювачем SEW MC07B0011-5A3-4-00/FSC11B за допомогою ПЛК ОВЕН ПЛК100-24.Р-М. Запропонований кінцевий автомат можна використовувати як основу для реалізації комунікації між приладами по іншим протоколам, у випадку, коли прилади мають спільний протокол фізичного рівня.

334

Список використаних джерел

1. Mehta B. R. Industrial Process Automation Systems / B. R. Mehta, Y. Jaganmohan Reddy., 2014. ISBN: 9780128010983.
2. Sen S. K. Fieldbus and Networking in Process Automation / Sunit Kumar Sen., 2017. ISBN: 9781351831680.
3. MOVILINK® Interface (RS-232, RS-485) [Електронний ресурс] // WAGO. – 2021. – Режим доступу до ресурсу: <https://www.wago.com/global/i-o-systems/movilink-interface-rs-232-rs-485/p/767-5204>.
4. Библиотека UNM. Руководство по применению. Версия 01 – Москва: Производственное Объединение ОВЕН, 2010. – 17 с.