

DOI: <https://doi.org/10.36910/6775-2524-0560-2020-41-26>

УДК: 004.056.5

<sup>1</sup>Здолбіцька Ніна Василівна, к.т.н., доцент

<https://orcid.org/0000-0002-1345-3581>

<sup>2</sup>Мельник Григорій Миколайович, к.т.н., доцент

<https://orcid.org/0000-0003-0646-7448>

<sup>1</sup>Мельник Василь Михайлович, к.ф.-м.н., доцент,

<http://orcid.org/0000-0001-8282-6639>

Колтунович Олексій Сергійович, студент

Мазуренко Владислав Вікторович, студент

<sup>1</sup>Луцький національний технічний університет

<sup>2</sup>Західноукраїнський національний університет

## АНАЛІЗ РОБОТИ ПОСЛІДОВНОГО ПРОТОКОЛУ UART ЗА ДОПОМОГОЮ ЦИФРОВОГО АНАЛІЗАТОРА

**Здолбіцька Н.В., Мельник Г.М., Мельник В.М., Колтунович О.В., Мазуренко В.В.** Аналіз роботи послідовного протоколу UART за допомогою цифрового аналізатора. Наведено короткий огляд використовуваних у даний час послідовних інтерфейсів передачі даних, вказані сфери застосування та способи апаратно-програмного декодування сигналів послідовних інтерфейсів. Розглянуто особливості паралельної роботи інтерфейсів USART на мікроконтролерній платформі. Проведено аналіз отриманих часових діаграм паралельної та послідовної роботи інтерфейсів.

**Ключові слова:** цифровий логічний аналізатор, сигнал, послідовний інтерфейс, протокол передачі даних, UART, LOGIC-U, Arduino UNO.

**Zdolbitska N.V., Melnyk G.M., Melnyk V.M., Koltunovich O.V., Mazurenko V.V.** Анализ работы последовательного протокола UART с помощью цифрового анализатора. Приведен краткий обзор используемых в настоящее время последовательных интерфейсов передачи данных, указанные сферы применения и способы аппаратно-программного декодирования сигналов последовательных интерфейсов. Рассмотрены особенности параллельной работы интерфейсов USART на микроконтроллерной платформе. Проведен анализ полученных временных диаграмм параллельной и последовательной работы интерфейсов.

**Ключевые слова:** цифровой логический анализатор, сигнал, последовательный интерфейс, протокол передачи данных, UART, LOGIC-U, Arduino UNO.

**Zdolbitska N.V., Melnyk G.M., Melnyk V.M., Koltunovich O.V., Mazurenko V.V.** Analysis of the serial UART protocol using a digital analyzer. A brief overview of currently used serial data interfaces is given, the areas of application and methods of hardware-software decoding of serial interface signals are indicated. Features of parallel operation of USART interfaces on a microcontroller platform are considered. The analysis of the received time diagrams of parallel and sequential operation of interfaces is carried out.

**Keywords:** digital logic analyzer, signal, serial interface, data transfer protocol, UART, LOGIC-U, Arduino UNO.

**Постановка задачі.** Під час проектування та розробки пристроїв однією із основних проблем є вибір інтерфейсів передачі даних, що якнайкраще підходять для виконання пристроєм його функцій та задач. Існує багато технічних реалізацій передачі даних. Кожен із них відрізняється від інших по пропускній здатності, способу передачі та принципу обміну інформацією.

Поширеність цифрових комунікацій та протоколів зв'язку породжує проблему їх налаштування та діагностики. Програмне та апаратне забезпечення для контролю на низькому рівні протоколів передачі даних, зокрема найпоширеніших послідовних, є зазвичай пропріетарним та потребує значних фінансових затрат.

Використання імітаційних віртуальних моделей, віртуальних пристроїв та програмних (софтверних) реалізацій протоколів зв'язку не дозволяє зі стовідсотковою гарантією віднайти причину збою чи нестабільної роботи реального обладнання. Тому на перший план виходять апаратно-програмні аналізатори цифрових протоколів, які дозволяють діагностувати проблему в реальному часі.

Останні покоління цифрових аналізаторів та сніферів дають відмінний результат аналізу, але для багатьох практичних задач можна використати більш прості та дешеві рішення, які забезпечують достатній рівень точності та мають розвинений програмний API-інтерфейс для реалізації на їх основі своїх продуктів.

**Аналіз досліджень.** Для моделювання та аналізу цифрових схем використовуються різні методики для покращення вивчення перетворення аналогового сигналу в цифровий [3,6,12].

У роботах Перевознікова С.І. подано основні переваги при використанні логічних аналізаторів для діагностування цифрових схем пристроїв та тестування систем внутрішньосхемного пошуку несправностей цифрових пристроїв, і аналізу отриманих даних, який реалізовано введенням штучних каналів для передачі інформації між внутрішніми вузлами пристрою. Діагностування функціонально

представляє собою поєднання та взаємозв'язок програмного та апаратного забезпечення. У даному випадку апаратно більш спрямовано на пришвидшення процесу тестування даних і аналіз одержаних результатів. Програмне забезпечення при цьому має забезпечити підготовку пристрою діагностування у кращий спосіб шляхом реструктуризації схеми пристрою засобами тестування [7,10,13]. Сукупність засобів для обробки та відображення інформації, що найбільш пристосовані для зручності користувачів; втілено доступністю багатокористувацьких налаштувань у графічних системах інтерфейс користувача, зокрема гнучкими налаштуваннями і вікон, і окремих їх елементів. У працях Лісовець С.М. подано опис основних характеристик промислових інтерфейсів, зокрема RS-485, CAN, Ethernet, наведено рекомендації по їх застосуванню, враховуючи основні переваги та недоліки [4]. Отримання рекомендацій відносно застосування того або іншого інтерфейсу для обміну інформацією при проектуванні систем автоматизованого керування дозволить підвищити надійність роботи відповідних систем автоматизованого керування і зменшити витрати на проектування, створення і підтримку роботи відповідної промислової мережі [8]. Робота логічного аналізатора полягає в записі цифрового сигналу з подальшим його аналізом в необхідних програмах, щоб здійснити перевірку роботи цифрової системи з необхідною точністю [1,10,11].

### **Порівняльний аналіз та проблематика використання апаратних засобів для діагностики цифрових інтерфейсів.**

Враховуючи різноманіття послідовних цифрових інтерфейсів та протоколів передачі даних, постає проблема аналізу сигналів та діагностики інтерфейсів. Для цього служать цифрові пристрої – логічні аналізатори цифрових сигналів.

Логічний аналізатор – це цифровий прилад, що дає змогу відображати на одному екрані декілька сигналів, що протікають одночасно в цифрових системах або електронних схемах. Даний пристрій може декодувати протоколи, перетворювати захоплені дані з пристроїв у часові діаграми, особливо якщо користувачеві необхідно побачити часові співвідношення між декількома сигналами в даній цифровій системі і т. д.

На різних етапах розробки, налаштування та тестування інтерфейсів цифрових пристроїв можуть використовуватися різні апаратні спеціалізовані засоби, серед яких логічні аналізатори, осцилографи, програматори, емулятори, симулятори та інші налагоджувальні системи [1,2,11].

Інтерфейс – це сукупність уніфікованих програмних, апаратних та конструктивних засобів, що реалізують необхідні алгоритми взаємодії між різними функціональними блоками мікроконтролерних систем та складових, що входять до автоматизованих систем управління. Найвідомішими послідовними інтерфейсами периферійних пристроїв є RS-232C, RS-422A та RS-485. На сьогоднішній день серед найпопулярніших мікроконтролерних інтерфейсів можна виділити послідовний синхронний інтерфейс SPI (Serial Peripheral Interface), випущений компанією Motorola, і двохпровідний інтерфейс I2C (Inter-Integrated Circuit) компанії Philips. Взаємодія з периферійними виконавчими пристроями за допомогою високошвидкісного послідовного інтерфейсу дало початок розвитку схемотехніки вбудованих мікропроцесорних систем на базі 8-розрядних мікроконтролерів.

Відбувається постійне вдосконалення промислових послідовних інтерфейсів, масове впровадження мікроконтролерів в автомобільну електроніку, що в свою чергу висуває підвищені вимоги до надійності передачі інформації між мікропроцесорними системами управління різними агрегатами автомобіля, наприклад між контролерами педалі управління та гальмівної системи. Це започаткувало розробку нового стандарту послідовного зв'язку – CAN-інтерфейсу (Controller Area Network). На даний час CAN-інтерфейс став де-факто «золотим» стандартом в автомобільній електроніці та промислових мережах зв'язку. Саме тому більшість фірм, що займаються серійним виробництвом комплектуючих автомобільної та промислової електроніки випускають мікроконтролери із вбудованим апаратним модулем CAN-мережі.

Логічний аналізатор – еквівалентна заміна осцилографа, що дає можливість працювати безпосередньо з цифровими сигналами та схемами, в якому усунуто недоліки аналогових осцилографів. Осцилограф дозволяє спостерігати за цифровими або аналоговими сигналами в залежності від часу, на відміну від нього, логічний аналізатор дає розгорнуту характеристику логічних станів цифрового сигналу в межах вибірки по тактах чи часу.

Цифровий логічний аналізатор виводить на екран інформацію практично в такому ж вигляді, як і цифровий осцилограф, при цьому відкладаючи по горизонталі часові проміжки, а по вертикалі – рівень напруги. Так як форма сигналів, відображена на обох пристроях залежить від часу, то кажуть, що сигнал представлено у часовій області у вигляді часових діаграм.

Разом з тим протягом останніх років відбувається черговий етап модернізації сучасних цифрових осцилографів. Планується тенденція розробки моделей з найсучаснішими характеристиками. Започатковано новий підхід до розширення можливостей сучасного

осцилографа, використавши вбудований логічний аналізатор в осцилографах на апаратно-програмному рівні, що отримав назву осцилограф змішаних сигналів. Такі осцилографи крім аналогових входів мають цифрові входи. Це надає можливість відображати логічні діаграми рівнів на одній часовій сітці, вивівши їх одночасно з кількох цифрових каналів. Раніше для реалізації цього завдання використовували осцилограф і логічний аналізатор як окремі пристрої. Окремий лабораторний цифровий осцилограф змішаних сигналів потужніший, має опції аналізу, декодування, глибокої пам'яті та доповнений 16-канальним логічним аналізатором, але водночас має високу вартість [14].

Наведемо таблицю для порівняння характеристик логічного аналізатора і осцилографа [2,9].

Таблиця 1 – Порівняльна характеристика логічного аналізатора і осцилографа

| Логічний аналізатор  | Осцилограф   |
|--|--|
| Використовується для вимірювання цифрових сигналів   | Використовується для вимірювання амплітуди конкретних аналогових сигналів для визначення форми сигналу та його природи   |
| Має більше вимірювальних каналів, більше пам'яті, але, як правило, відповідно меншу частоту дискретизації на канал   | Обмежена кількість каналів, відповідно більша частота дискретизації, в більшості двох-канальні (у більш дорогих моделях – цифрових каналів більше)                             |
| Використовується при необхідності одночасно відстежувати безліч цифрових сигналів, і встановити їх взаємозв'язки   | Використовується при необхідності простежити та перевірити спотворення сигналу на певних елементах аналогових і цифрових приладів, дозволяє характеризувати цілісність сигналу |
| Використовується, якщо необхідно подати сигнали саме у такому вигляді, як бачить їх устаткування   | Використовується, якщо необхідна висока точність при визначенні часових інтервалів   |
| Використовується при необхідності синхронізувати деяку комбінацію сигналів на декількох лініях і вивести результат   | Для перевірки стабільності сигналу, тремтіння фази і спектру тремтіння, визначення зсуву фази  |
| Може декодувати інформацію на шинах мікропроцесорів і представляти її в прийнятному для сприйняття людиною вигляді   | Використовується для визначення співвідношення між корисним сигналом і шумом, а також з'ясування характеру шуму  |
| При налагодженні взаємодії програм і обладнання логічні аналізатори відстежують виконання вбудованих програм і аналізують ефективність виконання програми. Деякі логічні аналізатори дозволяють зіставити вихідний код з конкретними діями обладнання в досліджуваній системі. | Може вимірювати параметри фронту і напруги сигналу для оцінки часових характеристик, наприклад, час установки і фіксації   |
| Може виявляти і аналізувати порушення синхронізації і перехідних процесів на шинах   | Виявляти помилки перехідних процесів, наприклад, час наростання сигналу, затримка поширення сигналу; побачити невеликі скачки, імпульси на сигналі                             |

В залежності від функціонального призначення та завдань сьогодення логічні аналізатори поділяють на такі основні види: аналізатори логічних станів та аналізатори часових діаграм [1, 11].

Основні характеристики логічних аналізаторів: кількість та типи каналів, глибина пам'яті, форма подання та принципи обміну даних, частота дискретизації, способи синхронізації та передачі інформації, пропускна здатність. Логічні аналізатори працюють у двох режимах: реєстрація та відтворення сигналу.

Провівши аналіз доступних на даний час логічних аналізаторів для проведення лабораторних досліджень зупинилися на виборі аналізатора протоколів LOGIC-U (STD) [11]. Даний аналізатор представляє інформацію, що проходить по різних каналах (максимум 8 каналів) у зручному для сприйняття вигляді, дозволяє зчитувати і розшифровувати цифрові дані, розписуючи по пакетах передачу даних, виділяти стартові, стопові, адресні байти, обробляти усі канали відразу. Даний інструмент для забезпечення максимальної зручності випускається в декількох модифікаціях: LOGIC-U PRO, LOGIC-U PLUS, LOGIC-U (STD), LOGIC-U ISL (рис. 1).



Рис. 1. Модифікації логічного аналізатора LOGIC-U

Комплектація LOGIC-U PRO FULL включає: блок аналізатора, чохол, кабелі для аналогових та цифрових сигналів зі змінними наконечниками типу гачок та захват, зовнішній BNC дільник  
 © Здолбівська Н.В., Мельник Г.М., Мельник В.М., Колтунович О.В., Мазуренко В.В.

сигналів, кабель USB: A-miniB, книжка-інструкція по експлуатації та програмне забезпечення на диску (рис. 1).

### Дослідження роботи реалізацій послідовного протоколу UART на мікроконтролерній платформі та обґрунтування результатів дослідження.

Мета дослідження – за допомогою цифрового логічного аналізатора Logic-U дослідити роботу в реальному часі послідовних протоколів зв'язку між пристроями, зокрема протоколу UART, його апаратну реалізацію – інтерфейс USART та програмну – Software Serial на базі персонального PC та платформ Arduino Uno.

Об'єктом дослідження обрано фізично взаємозв'язані за допомогою послідовних інтерфейсів персональний комп'ютер – плата Arduino Uno – плата Arduino Uno (рис. 2).

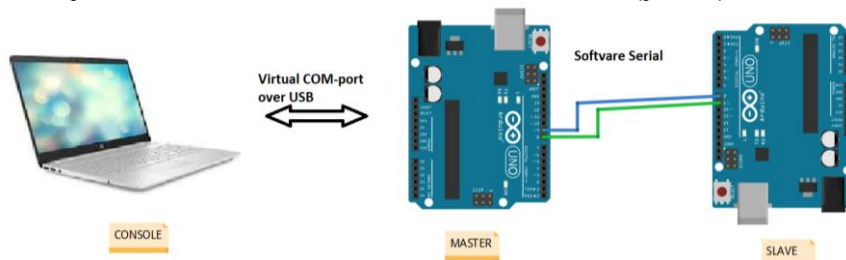


Рис. 2. Функціональна схема об'єкта дослідження

Предметом дослідження виступає послідовний протокол UART. Зокрема його апаратна та програмна реалізація на базі персонального комп'ютера (віртуальний COM-порт для з'єднання USB), апаратна (USART) та програмна (Software Serial) реалізації на базі платформи Arduino Uno.

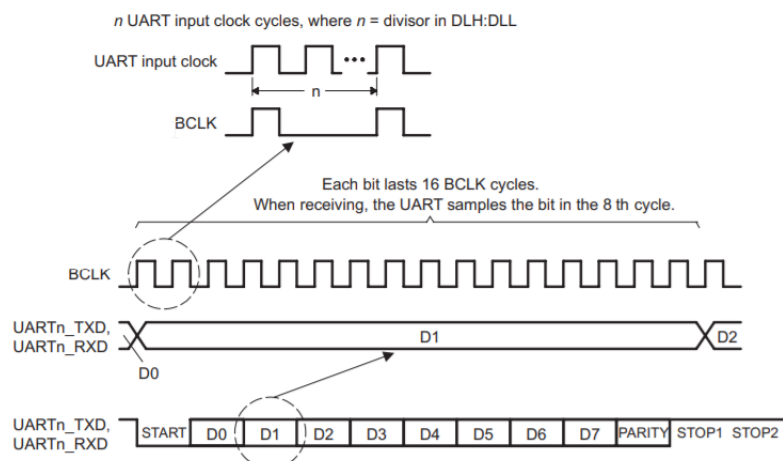


Рис. 3. Схема передачі даних по протоколу UART (). Сигнали на лініях Data Bit, BCLK, та UART Input Clock

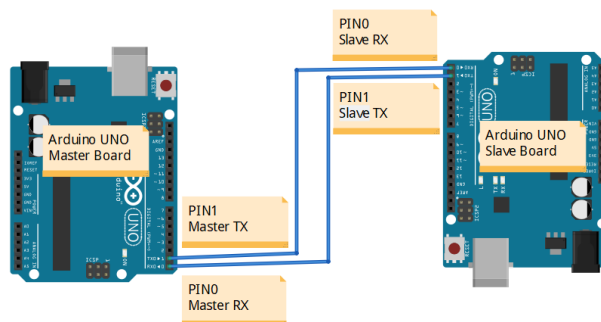
Коли говоримо про асинхронний інтерфейс, то маємо на увазі, що дані передаються без підтримки зовнішнього тактового сигналу. Послідовний інтерфейс, який надалі розглядатимемо в даній статті, є найбільш поширеним асинхронним протоколом.

Універсальний асинхронний приймач (Universal Asynchronous Receiver-Transmitter) – це фізичний пристрій прийому і передачі даних по двох проводах. Він дозволяє двом пристроям обмінюватися даними на різних швидкостях. У специфікацію протоколу UART не входять аналогові рівні на яких ведеться спілкування між пристроями, UART – це протокол передачі одиниць і нулів, електричну специфікацію на себе беруть інші стандарти, такі як TTL (transistor-transistor logic – транзисторно-транзисторна логіка), RS-232, RS-422, RS-485 та інші (RS, англ. recommended standard, – рекомендований стандарт). На даний момент у мікроконтролерах використовується в основному TTL (або точніше CMOS) UART для з'єднання не більше двох пристроїв.

У кожного пристрою, що підтримує UART зазвичай позначені два виводи: RX і TX. TX – означає transmit (передаю), RX – receive (приймаю). Звідси стає зрозуміло, що RX одного пристрою потрібно підключати до TX іншого. Якщо підключити RX одного пристрою до RX іншого, то обидва пристрої будуть слухати один одного, відповідно на фізичному рівні нічого не відбудеться. Якщо з'єднати TX і TX – це вже більш небезпечно, оскільки це виходи низького опору пристроїв і якщо на одному буде логічна одиниця, а на другому нуль – по з'єднувальному дроту піде струм короткого замикання (це залежить від конкретної програмної або апаратної реалізації). Хоча в сучасних чіпах від цього є захист, про всяк випадок, не варто на нього

орієнтуватися. Так само необхідно об'єднати референсні рівні двох пристроїв (GND-GND), якщо не мається на увазі гальванічна розв'язка.

На платформі Arduino і Arduino-сумісних платах апаратний UART позначається символами RX і TX поруч з відповідними виводами. Зокрема, у нашому випадку, на Arduino UNO це цифрові виводи 0 і 1 (рис. 4).

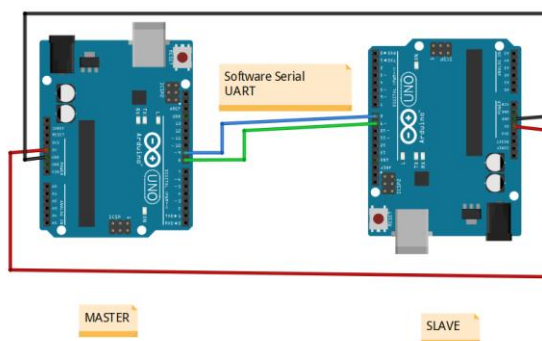


**Рис. 4.** Апаратна реалізація – зв'язок за допомогою USART між двома платами Arduino UNO

Крім апаратного UART практично в усіх програмованих цифрових системах існує можливість використовувати програмний UART. Це справедливо і по відношенню досліджуваної системи на базі персонального комп'ютера та плат Arduino. Програмний порт добре підходить для простих проектів, не критичних до часу роботи коду, або для налагодження проектів, що дозволяє не відключати модулі, які використовують UART під час завантаження скетчів у флеш пам'ять. Як стверджують розробники, при виконанні софтверного коду реалізації UART, ніякий інший код не може виконуватися поки програма займається зчитуванням даних з нього (буфера UART) і передача може здійснюватися тільки в напівдуплексному або симплексному режимах.

Цей факт і покладено в основу дослідження для відслідкування програмних ланцюжків коду. До того ж, на програмний RX можна призначити тільки ті виводи (піни), які підтримують переривання по зміні рівнів зовнішнього сигналу. На використаній в дослідженні платформі Arduino UNO, наприклад, це усі цифрові піни, крім 13-го.

Досліджувана система двох мікроконтролерів має схемотехнічну реалізацію програмного Software Serial протоколу згідно схеми рис. 5.



**Рис. 5.** Схема з'єднань для програмної реалізації UART між двома платами Arduino Uno

Для передачі даних та налаштування роботи по послідовному протоколу на стороні ПК використаємо стандартні налаштування послідовного інтерфейсу та консолі (рис. 6):

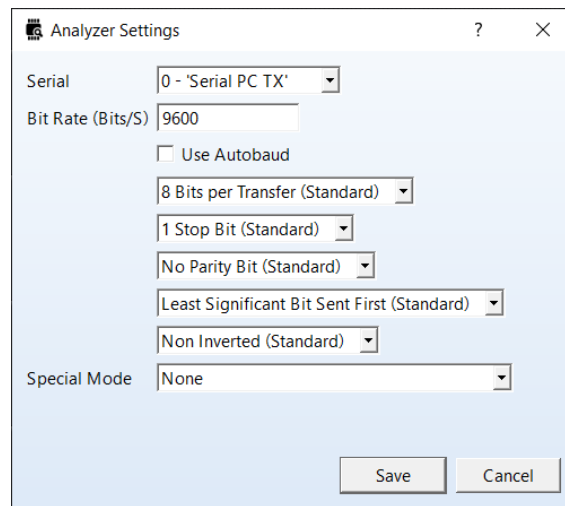


Рис. 6. Налаштування програмного забезпечення на стороні ПК

В ролі програмного забезпечення логічного аналізатора будемо використовувати програму Saleae Logic версії 1.1.15 (рис. 7). Вибірка та частота дискретизації відповідно 1М та 1 МГц.

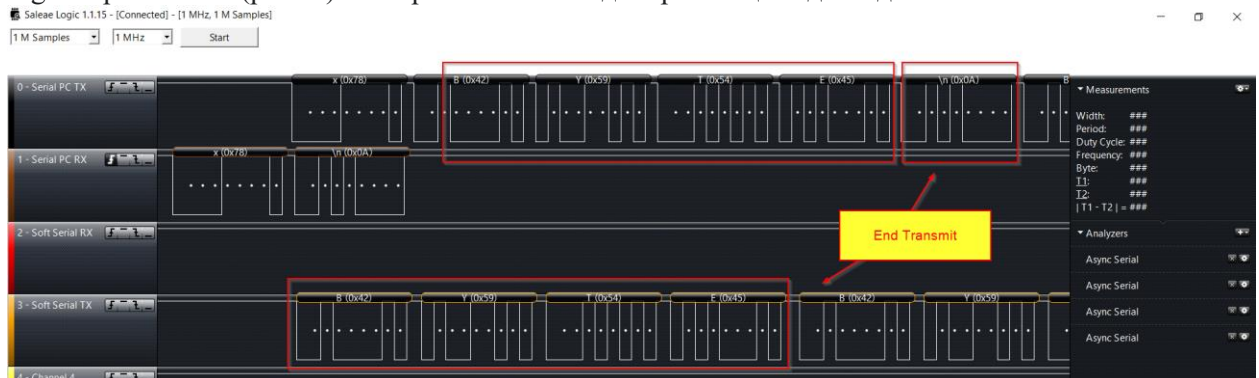


Рис. 7. Програмне забезпечення логічного аналізатора Saleae Logic версії 1.1.15

Для забезпечення синхронізації та стабільності роботи, налаштування UART на стороні мікроконтролерів – аналогічні, але реалізовані програмним чином, із підключенням стандартних бібліотек Serial/Software Serial.

Оскільки команди (фрагменти програмного коду) виконуються в обчислювальних системах послідовно, можливо із перериваннями на виконання деяких інших системних операцій, то можна стверджувати, що порядок виконання операторів може впливати на кінцевий результат обробки та передачі даних. Якщо для чистоти експерименту виокремити передачу та прийом даних по послідовному протоколу реалізовану одночасно декількома інтерфейсами, тоді можливо за допомогою логічного аналізатора протоколу простежити особливості та відповідності передачі даних, їх залежність від апаратної платформи чи програмного забезпечення.

Візьмемо для прикладу порівняння та аналізу наступні два фрагменти програмного коду (сторона Master, таблиця 2).

Таблиця 2 – Фрагменти програмного коду циклічної передачі даних

|  |  |  |
|--|--|--|
| <pre>void loop(){   if (Serial.available()){     Serial.write(Serial.read()); // eho     Serial.write("BYTE"); // BYTE-eho     // BYTE-string transmit     softSerial.write("BYTE");   } }</pre> |  | <pre>void loop(){   if (Serial.available()){     Serial.write(Serial.read()); // eho     // BYTE-string transmit     softSerial.write("BYTE");     Serial.write("BYTE"); // BYTE-eho   } }</pre> |
|--|--|--|

Фрагменти зліва та справа однаково посилають та опитують програмний та апаратний послідовні інтерфейси, відмінність полягає у послідовності опитування. На перший погляд результат має бути аналогічним, тобто пересилка даних має відбутися паралельно. На практиці маємо відмінну від очікуваної роботу скомпільованого програмного коду.

Результат роботи лівого фрагмента коду (таблиця 2) та розгорнутий аналіз часових діаграм можемо побачити відповідно на рисунках 7 та 8:

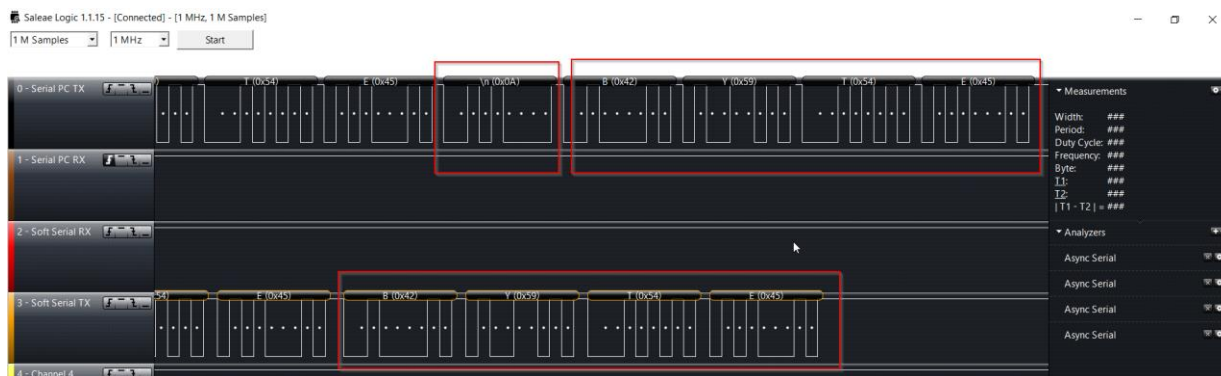


Рис. 8. Дубляж (повторення) послідовності байтів повідомлення «BYTE»

Результат роботи правого фрагмента коду (таблиця 2) наведено на рис. 9 та рис.10:

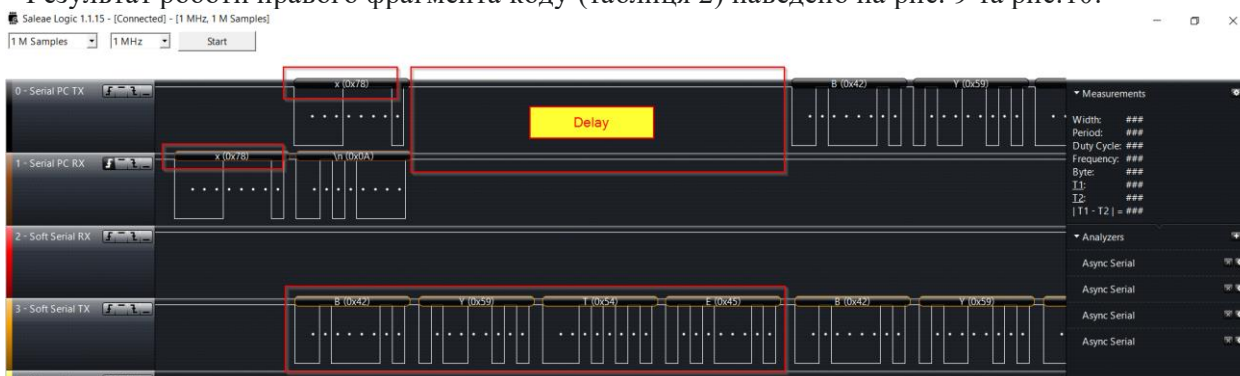


Рис. 9. Дубляж послідовності байтів та затримка виконання повідомлення «BYTE»

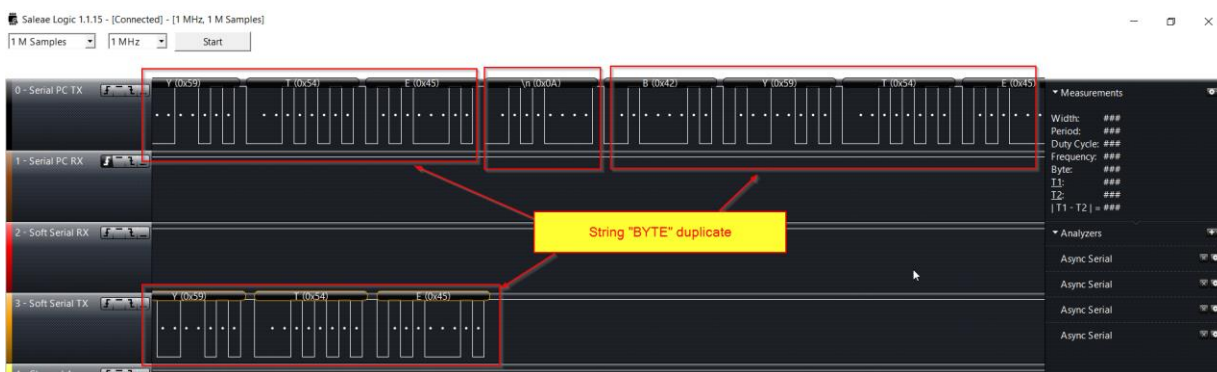


Рис. 10. Дубляж (повторення) послідовності байтів повідомлення «BYTE» після перестановки рядків фрагмента програмного коду

Результат роботи консолі ПК – однаковий в обох випадках реалізації програмного коду (рис. 11)

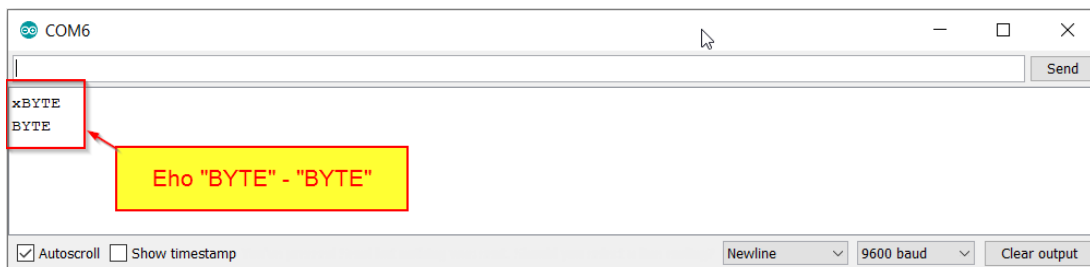


Рис. 11. Дубляж результатів роботи програми. Консоль ПК

### Висновки.

Відповідно до отриманих результатів можна стверджувати, що аналіз роботи мікроконтролерного коду програми та її налагодження за допомогою консолі ПК не може бути виконано для окремих випадків паралельно-послідовної роботи інтерфейсів. У даному випадку

реальну картину можна отримати тільки за допомогою безпосереднього аналізу цифрових сигналів досліджуваного інтерфейсу за допомогою логічного аналізатора чи осцилографа.

Усі результати, представлені в роботі, отримані автором самостійно, зокрема здійснено розробку пристрою-установки для дослідження та моделювання послідовних протоколів. Скомпільовані прошивки для мікроконтролерних ядер можуть бути використані аналогічно для діагностики SPI, I2C, USART, та ін. На основі виявлених особливостей роботи програмного коду дано рекомендації по застосуванню стандартних бібліотек та додатків. Проведено аналіз технічних рішень для передачі даних із використанням послідовних протоколів зв'язку провідних та безпроводних систем. Розглянуто особливості паралельної роботи інтерфейсів на мікроконтролерній платформі. Наведений опис отриманих часових діаграм паралельної та послідовної роботи інтерфейсів.

Асинхронний послідовний протокол UART має ряд вбудованих правил – механізмів, які допомагають забезпечити надійну і безпомилкову передачу даних. Це ті механізми, які дозволяють передавати дані без використання зовнішнього тактового сигналу:

- біти даних;
- біти синхронізації;
- біти перевірки парності;
- швидкість передачі.

Завдяки поєднанню цих правил-параметрів, протокол дуже гнучкий. Для успішного зв'язку потрібно переконатися, що обидва пристрої на шині налаштовані на використання однакових правил.

#### Список бібліографічних посилань.

1. Блінов Ю.В., Зінченко Ю.С. Розробка і дослідження структури логічного аналізатора цифрових сигналів. Донецький національний технічний університет. Інформатика и компьютерные технологии – Режим доступу <http://ea.donntu.edu.ua/handle/123456789/14217>
2. Бобало Ю.Я. (2014). Осцилографи та методи вимірювання радіотехнічних величин: навч. посібник. Ю.Я. Бобало, Л.А. Недоступ, М.Д. Кіселичник, О.В. Надобко. Львів: Видавництво Львівської політехніки. 88.
3. Горбенко Р.А., Корж Ю.М., Сомов С.В., Бендес Ю.П. (2011). Дослідження процесів аналого-цифрового перетворення за допомогою комп'ютерного моделювання. Обробка інформації в складних технічних системах Системи обробки інформації. Випуск 5 (95). 27-30.
4. Лісовець С.М., Дідович І.О. (2016) Швидкісні інтерфейси для зв'язку мікроконтролерів із периферійними пристроями технології та дизайн. № 2 (19). 1-7.
5. Логический анализатор цифровых сигналов – анализатор протоколов. Приставка к ПК Logos U PRO. Руководство по эксплуатации (2009). Харьков. 26.
6. Макаренко В.В., Маланчук Є.З., Рудик А.В., Співак В.М. (2017) Моделювання та аналіз цифрових схем. Рівне: НУВГП. 454.
7. Озеранський В.С., Перевозніков С.І. (2018) Особливості використання логічних аналізаторів для діагностування схем цифрових пристроїв. Матеріали XLVII науково-технічної конференції підрозділів ВНТУ. Вінниця. 14-23 березня 2018 р.
8. Олифер В. Г., Олифер Н. А. (2003) Основы сетей передачи данных. М.: ИНТУИТ.РУ. 248.
9. Основні відомості про логічні аналізатори. [Електронний ресурс]. Режим доступу: [http://www.eliks.ru/upload/tektronix/12\\_qxd.pdf](http://www.eliks.ru/upload/tektronix/12_qxd.pdf)
10. Перевозніков С.І. (2012) Особливості формування компонентних структур тестування для систем внутрішньосхемного пошуку несправностей цифрових пристроїв. С.І. Перевозніков, В.С. Озеранський, Л.В. Крупельницький // Вісник Вінницького політехнічного інституту. № 12. 62–71.
11. Перцовский М., Воробьев Е., Трифонов А. (2000). Применение логических анализаторов в тестировании цифровой техники. № 2. 6-12. [Електронний ресурс] – Режим доступу: <http://www.cta.ru/cms/f/?/366656.pdf>
12. Цирульник С.М., Ткачук В.М., Роптанов В.І. (2018). Методика застосування логічного аналізатора для дослідження інтерфейсів інформаційно-вимірювальних систем. Вчені записки ТНУ імені В.І. Вернадського. Серія: технічні науки. Том 29 (68) Ч. 1 № 1. 58-63.
13. Koltunovych O.S., Zdobitska N.V. (2020). Decoding and analysis of serial bus signals using digital signals logic analyzer. Збірник тез доповідей XIII Міжнародної науково-практичної конференції «Інформаційні технології і автоматизація – 2020», (Одеса, 22 - 23 жовтня 2020 р.). Одеська нац. акад. харч. технологій. Одеса: ОНАХТ, 2020. 55-57.
14. <https://masteram.com.ua/uk/articles-and-video/rigol-digital-oscilloscopes-and-company-profile/>

#### References

1. Blinov Y.V., Zinchenko Y.E. Development and research of the structure of the logical analyzer of digital signals. Donetsk National Technical University. Informatics and computer technologies – Access mode <http://ea.donntu.edu.ua/handle/123456789/14217>
2. Bobalo Yu.Ya. (2014). Oscilloscopes and methods of measuring radio quantities: textbook manual. Yu. Ya. Bobalo, LA Nedostup, MD Kiselychnyk, OV Nadobko. - Lviv: Lviv Polytechnic Publishing House. 88.



3. Gorbenko R.A., Korzh Yu.M., Somov S.V., Bendes Yu.P. (2011). Research of analog-to-digital conversion processes by means of computer modeling. Information processing in complex technical systems of Information processing systems. 5 (95). 27-30.
4. Lisovets S.M., Didovich I.O. (2016). High-speed interfaces for communication of microcontrollers with peripherals of technology and design № 2 (19). 1-7.
5. Logical analyzer of digital signals - protocol analyzer. PC console Logoc U PRO. Operation manual. (2009) Kharkiv. 26.
6. Makarenko V.V., Malanchuk E.Z., Rudyk A.V., Spivak V.M. (2017). Modeling and analysis of digital circuits. Rivne: NUVGP. 454.
7. Ozeransky V.S., Perevoznikov S.I. (2018). Features of the use of logic analyzers for diagnosing circuits of digital devices. Proceedings of the XLVII scientific and technical conference of VNTU, Vinnytsia, March 14-23.
8. Olifer V.G., Olifer N.A. (2003). Fundamentals of data transmission networks. M.: INTUIT.RU. 248.
9. Basic information about logic analyzers. [Electronic resource]. Access mode: [http://www.eliks.ru/upload/tektronix/12\\_qxd.pdf](http://www.eliks.ru/upload/tektronix/12_qxd.pdf)
10. Perevoznikov S.I. (2012). Peculiarities of formation of component testing structures for systems of internal circuit fault finding of digital devices. S.I. Perevoznikov, B.C. Ozeransky, L.V. Krupelnytsky. Bulletin of Vinnytsia Polytechnic Institute. № 12. 62-71.
11. Pertsovsky M., Vorobiev E., Trifonov A. (2000). Application of logic analyzers in testing digital equipment. № 2. 6-12. [Electronic resource] – Access mode: <http://www.cta.ru/cms/i/?/366656.pdf>
12. Tsurulnik S.M., Tkachuk V.M., Roptanov V.I. (2018). Methods of using a logic analyzer to study the interfaces of information and measuring systems. Scientific notes of TNU named after VI Vernadsky. Series: technical sciences. Volume 29 (68). Part 1 № 1. 58-63.