

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

БОНДАРЕЦЬ Дмитро Васильович

Контролер UART інтерфейсу / UART controller interface

спеціальність: 6.050102 - Комп'ютерна інженерія
освітньо-професійна програма - Комп'ютерні системи та мережі

Випускна квалікаційна робота

Виконав: студент групи КСМ-42/1
БОНДАРЕЦЬ Дмитро Васильович

Науковий керівник
к.т.н., Б.О.Масляк

ТЕРНОПІЛЬ 2019

РЕЗЮМЕ

Бакалаврська робота містить 53 сторінок пояснюючої записки, 15 рисунків, 8 таблиць, 2 додатки. Обсяг графічного матеріалу 2 аркуші формату А3.

Метою дипломної роботи є розробка контролера UART інтерфейсу.

Методи досліджень – методи побудови імітаційних моделей цифрових пристроїв.

В дипломній роботі, на основі аналізу навчальної та наукової літератури, стандартів підготовки фахівців з комп'ютерної інженерії, програм курсу комп'ютерні системи та комп'ютерна схемотехніка, поставлена та реалізована задача розробки контролера UART інтерфейсу. В роботі послідовно розглянуті питання методів побудови інтерфейсів. Розглянуто їх особливості. Виділено актуальність розробки даного контролера. Проектування структури контролера системи та деталізація складових створила умови для його реалізації в середовищі NI Multisim.

Ключові слова: АСИНХРОННИЙ, ПОСЛІДОВНИЙ, ІНТЕРФЕЙС, UART, СТРУКТУРНА СХЕМА, ЕЛЕКТРИЧНА СХЕМА, КОНТРОЛЕР, ПЕРЕДАЧА ІНФОРМАЦІЇ, ПРИЙМАЧ, ПЕРЕДАВАЧ.

RESUME

The diploma project contains 53 pages of explanatory note, 15 figures, 8 tables, 2 application. The volume of graphic material is 2 sheets of A3 format.

The purpose of the thesis is to development of a controller UART interface.

Methods of research - methods of constructing simulation models of digital devices.

In the dissertation, on the basis of the analysis of educational and scientific literature, the standards of training specialists in computer engineering, programs of the course computer systems and computer circuitry, the task of developing of a system for controller UART interface. In the work consistently considered the questions of methods. The topicality of the development of controller UART interface. Designing the structure of the system controller and detailing the components created the conditions for its implementation in the medium of NI Multisim.

Key words: ASYNCHRONOUS, SEQUENTIAL, INTERFACING, UART, STRUCTURED CIRCUIT, ELECTRICAL CIRCUIT, CONTROLLER, INFORMATION TRANSMISSION, RECEIVER, TRANSMITTER.

ЗМІСТ

| | |
|--|----|
| Вступ..... | 10 |
| 1 Застосування інтерфейсів при побудові комп'ютерних систем | 12 |
| 1.1 Сфери застосування інтерфейсів в комп'ютерних системах | 12 |
| 1.2 Програмно–апаратні методи побудови цифрових інтерфейсів | 16 |
| 1.3 Постановка задачі по проектуванню послідовного асинхронного інтерфейсу | 20 |
| 2 Проектування компонентів контролера послідовного асинхронного інтерфейсу | 21 |
| 2.1 Особливості побудови контролера асинхронної послідовної передачі інформації..... | 21 |
| 2.2 Проектування структури передавача інтерфейсу UART | 23 |
| 2.3 Проектування структури приймача інтерфейсу UART..... | 25 |
| 3 Апаратна реалізація контролера послідовного асинхронного інтерфейсу | 28 |
| 3.1 Моделювання роботи блоку зв'язку з системною шиною | 28 |
| 3.2 Апаратна реалізація UART передавача..... | 31 |
| 3.3 Апаратна реалізація UART приймача | 35 |
| 4 Техніко–економічне обґрунтування пристрою | 37 |
| 4.1 Стадії технологічного процесу | 37 |
| 4.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи ... | 38 |
| 4.3 Розрахунок матеріальних витрат | 40 |
| 4.4 Розрахунок витрат на електроенергію | 41 |
| 4.5 Обчислення накладних витрат | 42 |
| 4.6 Інші витрати..... | 43 |
| 4.7 Складання кошторису витрат та визначення собівартості..... | 43 |
| 4.8 Розрахунок ціни проекту | 44 |

| | | | | | | | |
|-----------|------|----------------|--------|------|---------------------------------|------|---------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | | |
| Змн. | Арк. | № докум. | Підпис | Дата | Контролер UART інтерфейсу | | |
| Розроб. | | Бондарець Д.В. | | | | | |
| Перевір. | | Масляк Б.О. | | | | | |
| Конс. | | Паздрій І.Р. | | | | | |
| Н. Контр. | | Гураль І.В. | | | | | |
| Затверд. | | Березький О.М. | | | Літ. | Арк. | Акрушів |
| | | | | | 8 | 63 | |
| | | | | | THEU. ФКІТ. КСМ–42/1 | | |

| | |
|--|----|
| 4.9 Визначення економічної ефективності і терміну окупності капітальних вкладень | 45 |
| Висновок | 47 |
| Список використаних джерел..... | 48 |
| Додаток А Світло-копія публікації | 51 |
| Додаток Б Довідка про використання | 54 |

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 9 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

ВСТУП

На сьогоднішній день інтерфейси стали невід'ємною частиною нашого життя. Інтерфейс — набір правил, засобів і методів взаємодії між елементами системи. Основою взаємодії в сучасних інформаційних системах є інтерфейс. Якщо інтерфейс не змінюється, це дає можливість змінювати сам об'єкт, не перероблюючи його інші види взаємодії з іншими об'єктами. Це дозволяє створювати інші пристрої чи функції, які будуть використовувати такий спосіб взаємодії, а також масштабувати інтерфейс.

Як вже зазначалося вище, інтерфейс – це, перш за все, набір правил. Як будь-які правила, їх можна узагальнити, зібрати в «кодекс», згрупувати за спільною ознакою [1]. Таким чином, ми прийшли до поняття «вид інтерфейсу» як об'єднання за схожістю способів взаємодії людини і комп'ютерів. Коротко можна запропонувати наступну схематичну класифікацію різних інтерфейсів спілкування людини і комп'ютера.

Сучасними видами інтерфейсів є:

1) Командний інтерфейс. Командний інтерфейс називається так по тому, що в цьому виді інтерфейсу людина подає «команди» комп'ютеру, а комп'ютер їх виконує і видає результат людині. Командний інтерфейс реалізований у вигляді пакетної технології та технології командного рядка.

2) WIMP – інтерфейс (Window – вікно, Image – образ, Menu – меню, Pointer – покажчик). Характерною особливістю цього виду інтерфейсу є те, що діалог з користувачем ведеться не за допомогою команд, а за допомогою графічних образів – меню, вікон, інших елементів. Хоча і в цьому інтерфейсі подаються команди машині, але це робиться «замасковано», через графічні образи [2]. Цей вид інтерфейсу реалізований на двох рівнях технологій: простий графічний інтерфейс і «чистий» WIMP – інтерфейс.

3) SILK – інтерфейс (Speech – мова, Image – образ, Language – мова, Knowledge – знання). Цей вид інтерфейсу найбільш наближений до звичайної,

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 10 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

людської формі спілкування. У рамках цього інтерфейсу йде звичайний «розмова» людини і комп'ютера. При цьому комп'ютер знаходить для себе команди, аналізуючи людську мову і знаходячи в ній ключові фрази. Результат виконання команд він також перетворює в зрозумілу людині форму. Цей вид інтерфейсу найбільш вимогливий до апаратних ресурсів комп'ютера, і тому його застосовують в основному для військових цілей.

Тому в даній роботі мною була вибрана тема контролер UART інтерфейсу. В даній роботі я розгляну типи підключення інтерфейсів, різні режими передачі між ними [3]. Також розкажу чому саме мною була вибрана дана тема, а саме типом розглядання плюсів та мінусів при підключенні та при передачі даних між інтерфейсами. Також мною буде розроблено структурні та електронні схеми даного приладу при передачі та при прийомі даних.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 11 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

1 ЗАСТОСУВАННЯ ІНТЕРФЕЙСІВ ПРИ ПОБУДОВІ КОМП'ЮТЕРНИХ СИСТЕМ

1.1 Сфери застосування інтерфейсів в комп'ютерних системах

Зв'язок пристроїв автоматизованих систем один з одним здійснюється за допомогою інтерфейсів (засобів сполучення. Інтерфейс представляє собою набір електронних схем і алгоритмів, ліній і шин, сигналів , призначену для здійснення обміну інформацією між пристроями.

Сукупність програмних, конструктивних, технічних, необхідних для створення різних функціональних елементів в автоматичних системах опрацювання інформації в умовах, запропонованих стандартом і спрямованих на забезпечення інформаційної, конструктивної та електричної сумісності зазначених елементів [4].

Всі компоненти об'єднуються між собою за допомогою інтерфейсів. Поняття інтерфейс є базовим для того щоб зрозуміти принципи функціонування комп'ютерів.

Конфігурації інтерфейсів розроблені виходячи з таких вимог: простота нарощування структури многопроцесорного комплексу і можливість доступу для діагностики та широка область застосування отримання потрібної продуктивності і організації стандартного обміну інформацією між блоками обчислювальної системи незалежно від їх швидкодії. Електричні з'єднання між висновками мікросхем виконуються електричними зв'язками або лініями [5]. Ці лінії згруповані за певним функціональним призначенням утворюють шину адреси, шину даних і шину управління. Сукупність шин утворює магістраль.

Інтерфейс можна уявити як набір сигнальних ліній (шину), з'єднаних за призначенням (дані, адреси, управління), з певними електричними характеристиками і протоколами обміну даними.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 12 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

Шина інтерфейсу обробляється контролерами та службовими пристроями (буфера, регістри, мости), а також програмною оболонкою (драйверами і операційною системою) [6].

Використовувані в теперішній час шини відрізняються за розрядністю, способом передачі сигналу (послідовні або паралельні), пропускну здатністю, кількістю і типом підтримуваних пристроїв, а також протоколом роботи. Шини можуть бути синхронними (здійснюють передачу даних тільки по тактових імпульсах) і асинхронними (здійснюють передачу даних в довільні моменти часу), а також використовувати різні схеми арбітражу (тобто способу спільного використання шини декількома пристроями). А зараз ми більш детально розглянемо паралельні і послідовні інтерфейси [7].

Послідовний інтерфейс – це основний тип інтерфейсу, за допомогою якого здійснюється взаємодія між комп'ютерами. Термін асинхронний означає, що при передачі даних не використовуються ніякі синхронізуючі сигнали й окремі символи можуть передаватися з довільними інтервалами.

Кожному символу, який передається через послідовне з'єднання, повинен передувати стандартний стартовий сигнал, а завершувати його передачу повинен стоповий сигнал.

Стартовий сигнал – це 0 біт, названий стартовим бітом [8]. Його призначення – сповістити приймаючий пристрій про те, що наступні 8 бітів є байтом даних. Після символу передаються один або два степових біти, передачі символу, що сигналізують про кінець. Даний тип зв'язку найбільш характерний для телефонної мережі, в якій напрямки обслуговує окремий провідник. До послідовних портів можна підключити: модеми, плоттери, принтери, інші комп'ютери, пристрої зчитування штрих-кодів або схему управління пристроями.

В загальному у всіх пристроях, для яких необхідний двонаправлений зв'язок з комп'ютером використовується послідовний порт Rs232, який дозволяє передавати дані між різними пристроями, який зображений на малюнку 1.1.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 13 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

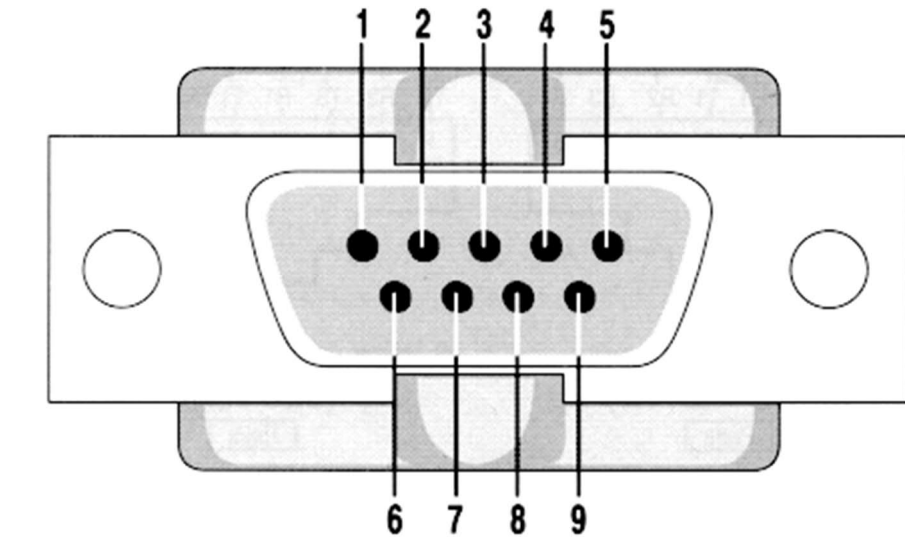


Рисунок 1.1 – порт RS232

Призначення усіх контактів 9-контактного рознімача RS-232C зображено у таблиці 1.1.

Таблиця 1.1 – Призначення контактів 9-контактного рознімача RS-232C

| Вивід | Найменування | Напрямок | Опис |
|-------|--------------|----------|----------------------|
| 1 | CD | ← | Визначення несучої |
| 2 | RXD | ← | Дані що приймаються |
| 3 | TXD | → | Дані що передаються |
| 4 | DTR | → | Готовність терміналу |
| 5 | GND | – | Корпус системний |
| 6 | DSR | ← | Готовність даних |
| 7 | RTS | → | Запит на відправку |
| 8 | CTS | ← | Готовність прийому |
| 9 | RI | ← | Індикатор |

У більшості схем дані передаються асинхронно, а саме у вигляді послідовності пакета даних. Кожен окремий пакет містить один символ коду ASCII, а інформації в пакеті вистачає для його декодування без окремого сигналу синхронізації [9].

Послідовні порти COM введення / виведення (асинхронні) призначені для забезпечення загального цільового інтерфейсу з великою кількістю різних видів електронного обладнання. BIOS сучасних комп'ютерів може підтримувати до 4-х COM-портів (COM1-COM3).

Фізично роз'єми COM – порту є дев'яти контактний двох рядний роз'єм, але деколи можна зустріти (в старих пристроях або в деяких модемах) двадцяти п'яти контактні роз'єми [10].

Послідовний інтерфейс зручний при великій довжині кабелю, тому при відправленні даних по довшому кабелю якість сигналу змінюється. Послідовне з'єднання краще ще й тим, що діаметр кабелю менший, і його простіше пропускати через стіни і електричні ізоляційні труби. Крім того, послідовне з'єднання ліпше з економічних міркувань, так як застосовується лише одна лінія даних і зменшується вартість кабелю.

А зараз розглянемо паралельний інтерфейс. Він відрізняється високою швидкістю, часто використовується для під'єднання принтера до комп'ютера, а ще для з'єднання комп'ютерів (при цьому вище швидкість передачі даних, ніж при з'єднанні через послідовні порти: 4, а не 1 біт за раз).

До даних портів можуть під'єднуватися усі. Часто двонаправлений паралельний порт для обміну даними між комп'ютерами, наприклад між настільним і портативним [11]. Для зв'язку комп'ютерів через паралельний порт треба спеціальний кабель. У Windows включена спеціальна програма, яка називається пряме кабельне з'єднання (Direct Cable Connection). Вона дозволяє об'єднати два комп'ютери через модемний нуль-кабель.

Паралельне з'єднання має ряд недоліків, одним з яких є фазовий зсув сигналу, через що довжина паралельних каналів, наприклад SCSI, обмежена і не повинна перевищувати 3 м. Проблема в тому, що, хоча 8 і 16-розрядні дані одночасно пересилаються передавачем, через затримки одні біти прибувають в приймач раніше інших. Отже, чим довше кабель, тим більше час затримки між першими і останніми прибулими битами на приймальному кінці. Послідовна

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 15 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

шина дозволяє одночасно передавати 1 біт даних. Відсутність затримок при передачі даних дозволяє значно збільшити тактову частоту [12].

Отже підведемо невеликі підсумки. З огляду на вагомні переваги послідовного над паралельним інтерфейсом, в подальших дослідженнях ми будемо розглядати саме його.

1.2 Програмно–апаратні методи побудови цифрових інтерфейсів

Виходячи з розділу 1.1 ми більш детально розглянемо послідовну передачу даних. Послідовна передача даних може здійснюватися в асинхронному або синхронному режимах. Для початку розглянемо асинхронну передачу.

Поняття "асинхронний" в даному випадку означає, що передаються слова слідуєть один за одним не безперервним потоком, а з довільними нерівними інтервалами [13].

При асинхронній передачі кожен символ передається різною послілкою. Стартові біти попереджують про початок передачі. Потім передається символ. Для визначення правильності передачі використовується біт парності (біт парності дорівнює 1, якщо кількість одиниць в символі непарній, і дорівнює 0 в іншому випадку). Останній біт сигналізує про кінець передачі.

Користувач пристрою повідомляє про те, що символи визначаються по стартовим і стоповим сигналам, а не до моменту їх відправлення. Асинхронний інтерфейс направлений на відправлення символів, а при відправленні задіюється наближено 20% інформації для ідентифікації кожного символу [14]. Послідовний означає, що передача даних відбувається з одиночного провідника, а біти в даному випадку відправляються послідовно, один за одним.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 16 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

Послідовний асинхронний інтерфейс зручний при великій довжині кабелю, тому при відправленні даних по довшому кабелю якість сигналу незначно зменшується. Послідовне з'єднання краще і тим, що діаметр кабелю менший, і його простіше пропустити через стіни і електричні ізоляційні труби. Крім того, послідовне з'єднання ліпше з економічних міркувань, так як задіюється лише одна лінія даних і зменшується вартість кабелю. Він надає поряд з крайньою простотою реалізації, високу захищеність на великих лініях. Головний елемент послідовного інтерфейсу – 16450 UART мікросхема (Universal Asynchrony Receiver Transmitter, універсальний асинхронний приймач–передавач), що надає найбільшу швидкість відправлення даних 115200 біт / с, часто встроєна в мікросхему південного моста системного чіпсета. Переваги: нескладна відпрацьована система і недороге частиною інтерфейсу обладнання.

Тепер розглянемо синхронну послідовну передачу. При використанні синхронного методу дані відправляються блоками. Для синхронізації роботи передавача і приймача на початку блоку відправляються біти синхронізації. Після них відправляються дані, код знаходження помилки і символ закінчення передачі. Код знаходження помилки обчислюється по вмісту поля даних і дозволяє визначити вірність прийнятої інформації. Наведу для прикладу схемку простого контролера для синхронної передачі даних по послідовній лінії зв'язку який зображений на малюнку 1.2.

Восьмизрядний буферний регістр призначений для тимчасового зберігання байта даних до завантаження в регістр зсуву. Записування байта даних в буферний регістр з шини даних системного інтерфейсу проходить так само, як і в паралельному інтерфейсі, лише при наявності одиниці в однорозрядному регістрі стану контролера. Одиниця в даному регістрі показує на готовність контролера прийняти наступний байт в буферний регістр [15].

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 17 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

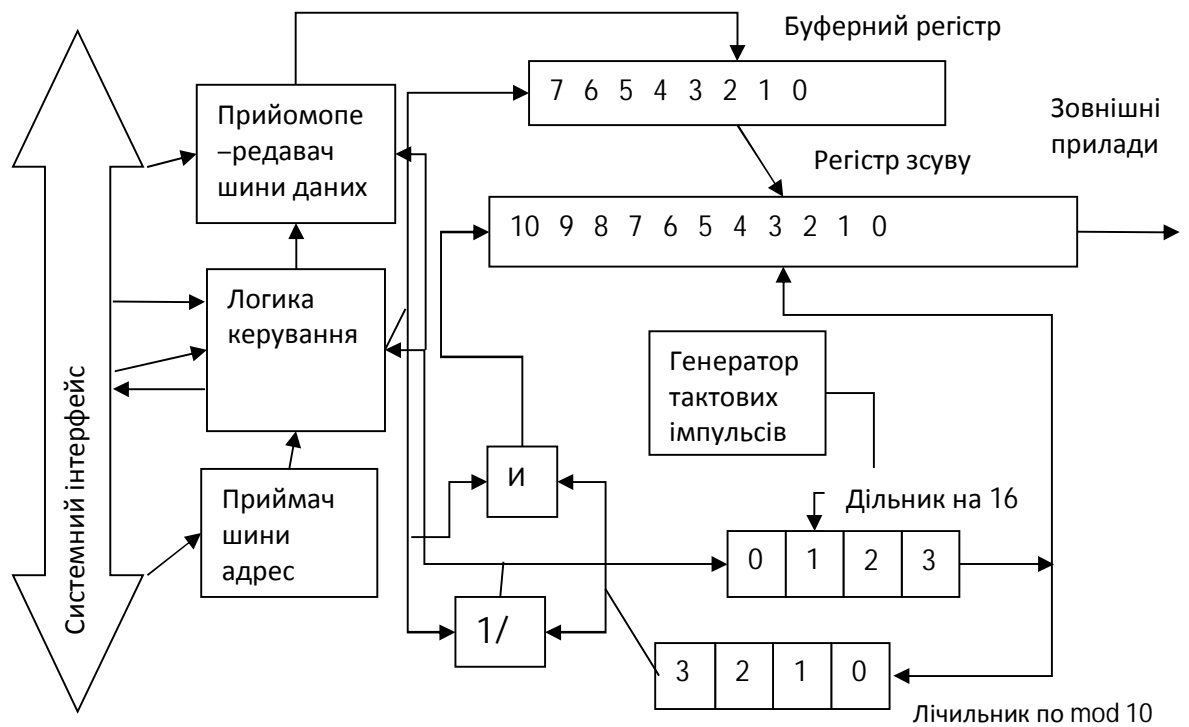


Рисунок 1.2 – Контролер послідовної синхронної передачі

Змінення даних з паралельного формату, в якому вони прийшли в буферний регістр контролера з системного інтерфейсу, в послідовний і передача їх на лінію зв'язку виробляються в регістрі зсуву за допомогою генератора тактових імпульсів і двійкового трохрозрядного лічильника імпульсів наступним чином. Послідовна лінія зв'язку контролера з зовнішнім пристроєм під'єднується до виходу найменшого розряду зсувного регістру. За чергового тактового імпульсу дані зсувного регістру зміщуються на один розряд вправо і в лінію зв'язку "Дані" видаються значення чергового розряду. Разом зі зміщенням на зовнішні прилади, передається по іншій окремій лінії "Синхронізація" тактовий імпульс. Так кожен прямуючи по лінії "Дані" біт інформації характеризується синхронізуючим сигналом по лінії "Синхронізація", що забезпечує його однозначне сприйняття на приймальному кінці послідовної лінії зв'язку.

Кількість переданих в лінію тактових сигналів і переданих біт інформації рахується лічильником тактових імпульсів. Коли вміст лічильника буде рівним семи. В одну лінію передані вісім біт інформації, створюється управляючий

сигнал "Завантаження", що надає запис в регістр зсуву кожного байта з буферного регістра. Даним керуючим сигналом ставиться в "1" регістр стану. Наступним тактовим імпульсом лічильник буде скинутий в "0", і розпочнеться наступний цикл видавання восьми бітів інформації з зсувного регістру в лінію передачі.

Синхронна послідовна передача різних окремих бітів даних на лінію зв'язку має бути без будь-якої перерви, і кожен наступний байт даних має бути загрузений в буферний регістр з системного інтерфейсу за певний період, що не збільшує часу передавання 8 бітів в послідовну лінію зв'язку [16].

При завантаженні байта даних в буферний регістр, обнуляється регістр стану контролера. Нуль даному регістрі показує, що в лінію зв'язку відправляється байт даних з зсувного регістру, а усі наступні передаються байти даних завантаження в зсувний регістр.

Контролер для синхронного послідовного приймання даних з зовнішніх приладів складається з тих же компонентів, що і контролер для послідовної синхронної передачі, за винятком генератора тактових імпульсів.

Недоліки:

- Необхідно більше висновків, ніж для інтерфейсу.
- Ведений пристрій не може керувати потоком даних.
- Немає підтвердження прийому даних з боку відомого пристрою.
- Немає певного стандарту протоколу виявлення помилок.
- Відсутність офіційного стандарту.
- SPI поступається таким стандартам, як UART і CAN.
- Наявність безлічі варіантів реалізації інтерфейсу.
- Відсутність підтримки гарячого підключення пристроїв.

Зваживши всі плюси та мінуси синхронного та асинхронного послідовного інтерфейсу мною був вибраний послідовний асинхронний інтерфейс.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 19 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

1.3 Постановка задачі по проектуванню послідовного асинхронного інтерфейсу

Після ознайомлення з перевагами послідовного асинхронного інтерфейсу мною був вибраний наступний план дій. Потрібно більш детально побудувати структурну схему контролера, розглянути табличку істиності в яких випадках як передаються дані. Детально розглянемо структурні схеми побудовані мною. Також потрібно буде побудувати структурні схеми приймача та передавача послідовного асинхронного інтерфейсу [17].

Потім після побудування та ознайомлення з даними схемами потрібно буде вибрати середовище в якому я буду проектувати. Щоб вибрати середовище я розгляну більш детально декілька з них та виберу найкраще для мене.

В середовищі потрібно побудувати електричні схеми а саме передавача та приймача послідовного асинхронного інтерфейсу. Також більш детально описати кожна з них, а саме які деталі використовували які дані при вході ми виставляли. А також як налаштовували дані схемки вцілому. Після розроблення схем в певному середовищі та налаштуванні кожної з них я описав деталі чому саме дані мікросхеми та інші прилади я вибрав.

Все це я розробив завдяки проведеним дослідженням у першому розділі. А саме розглянувши різновидності інтерфейсів їхні типи підключення і т.д. Та провівши певні дослідження на рахунок переваг. Отже у наступних розділах нам потрібно розробити усе що я сказав раніше та детально описати кожна схем.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 20 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

2 ПРОЕКТУВАННЯ КОМПОНЕНТІВ КОНТРОЛЛЕРА ПОСЛІДОВНОГО АСИНХРОННОГО ІНТЕРФЕЙСУ

2.1 Особливості побудови контролера асинхронної послідовної передачі інформації

В даному підрозділі нам потрібно буде розглянути більш детально асинхронний (старт–стоп) режим для побудови узагальненої структурної схеми інтерфейсу UART [18].

При обмінюванні даними на фізичному рівні 1 інформації є біт, тому засобами які підтримують побітову синхронізацію між передавачем та приймачем є засоби фізичного рівня.

Канальний рівень керує кадрами даних і можна синхронізувати між передавачем та приймачем на рівні кадрів. Виявлення початку першого байта кадру, виявлення меж полів кадру і виявлення ознаки закінчення кадру це все входить в дію приймача.

Часто вистачає забезпечити синхронізацію на визначених двох рівнях, а саме бітовому і кадровому для того, щоб приймач і передавач могли надати обмін інформацією. Однак при гіршій якості лінії зв'язку (зазвичай це відноситься до телефонних комутованих каналах) для зниження вартості апаратури і збільшення надійності відправлення даних вводять додаткові кошти синхронізації на рівні байт. Даний режим роботи є асинхронним чи старт–стопним [19]. Другою причиною застосування даного режиму роботи є наявність пристроїв, які генерують байти даних в рандомні моменти часу. В цьому режимі кожен байт даних характеризується певними сигналами (старт) і (стоп). Дія даних сигналів в тому, щоб сповістити приймач про надходження даних і дати йому достатньо часу для виконання певних функцій, зв'язаних із синхронізацією, до поступання наступного біта.

Сигнал старт завжди має довжину, в один тактовий інтервал, а сигнал стоп може продовжуватись (1; 1,5 або 2) такту. Даний режим називається

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 21 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

асинхронним тому, що усі байти можуть бути дещо зміщені у часі щодо побітових тактів попередніх байтів [20]. Байт не впливає на правильність прийнятих даних тому, що на початку усіх байтів відбувається додаткова синхронізація приймача з джерелом за рахунок бітів "старт".

При синхронному режимі передавання старт-стопного біти між кожної пари байт відсутні. Призначені для користувача дані збираються в кадр, який передує байтами синхронізації.

На основі даної інформації мною була створена узагальнена структурна схема інтерфейсу UART, який зображений на рисунку 2.1.

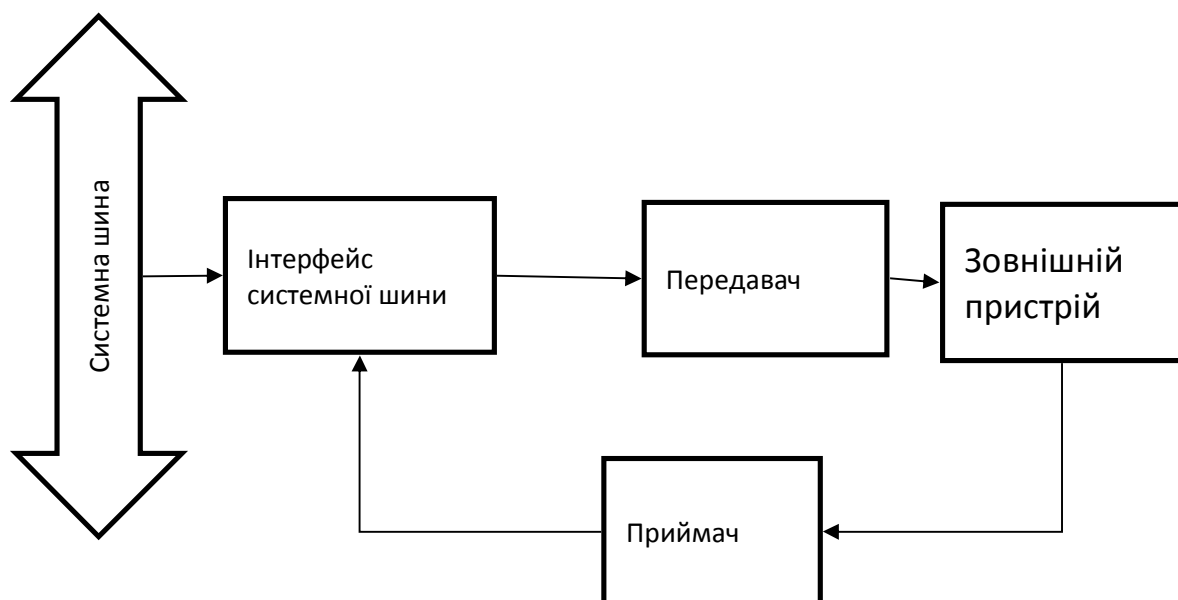


Рисунок 2.1 – Узагальнена структурна схема інтерфейсу UART

На даній структурній схемі мною було зображено взаємодію системної шини з системним інтерфейсом, а також як поступає інформація до передавача та приймача. На основі даної загальної структурної схеми в подальшому буде розроблено структурні схеми передавача та приймача інтерфейсу UART [21].

2.2 Проектування структури передавача інтерфейсу UART

Розроблення послідовного асинхронного обміну даними із зовнішнім пристроєм ускладнюється тим, що на приймальній і передавальній стороні послідовної лінії зв'язку використовуються налаштовані на однакову частоту, але фізично різні генератори тактових імпульсів [22]. Отже, загальна синхронізація відсутня. Розглянемо на прикладах організацію контролерів послідовних інтерфейсів для послідовних асинхронних передачі.

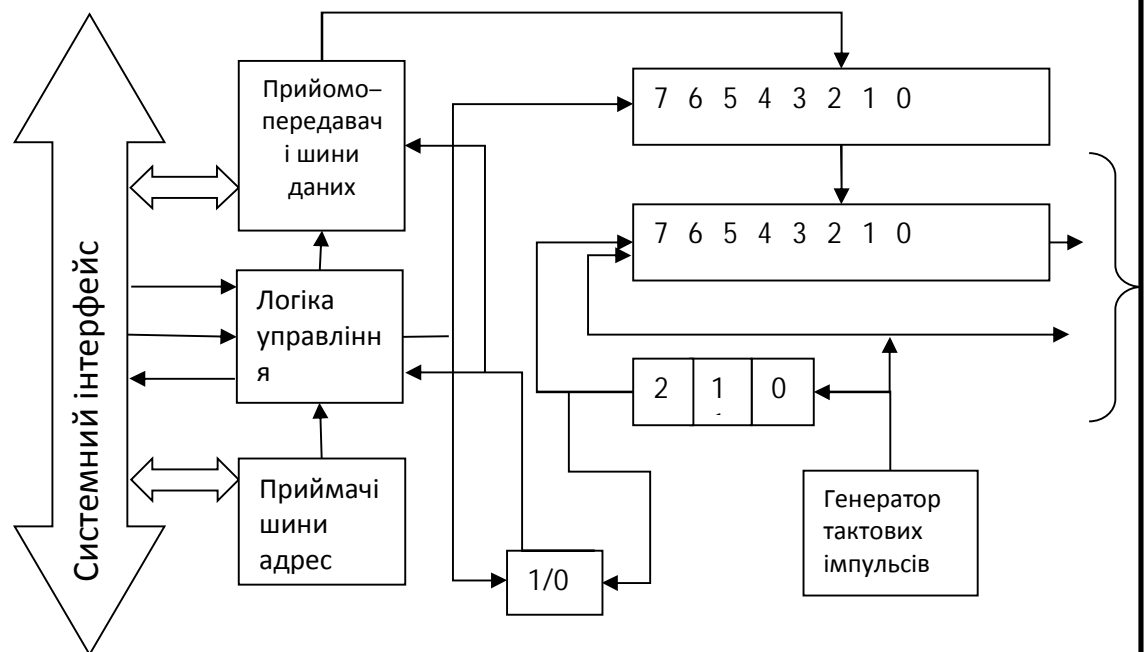


Рисунок 2.1 – Структурна схема передавача інтерфейсу UART

Контролер для асинхронної передачі даних на зовнішній інтерфейс по послідовній лінії зв'язку представлений на рисунку 2.2.

Після передавання кожного байта даних в регістр стану записується 1 [23]. Вихідний одиничний сигнал регістра повідомляє процесор про те що контролер готовий до прийому наступного байта даних і передавання його по лінії зв'язку в зовнішній прилад. Цей самий сигнал не дозволяє створення імпульсів з схеми виготовлення імпульсів зсуву – дільника частоти сигналів

тактового генератора на шістнадцять [24]. Лічильник імпульсів зсуву (лічильник по mod 10) перебуває в нульовому стані і його одиничний вихідний сигнал передається на вентиль, готуючи ланцюг виготовлення сигналу регістра зсуву.

Процес передавання байта даних розпочинається з того, що процесор, виконуючи команду "Вивід", ставить цей байт на шині даних. Одночасно процесор формує керуючий сигнал системного інтерфейсу "Вивід", за яким проводяться запис переданого байта в буферний регістр, скидання регістра стану і сигналу на вентилю "Завантаження". Байт що передається переписується в розряди від нуля до восьми регістру зсуву, в 0 розряд регістру зсуву заходить 0 (стартовий біт), а в розряди 9 і 10 – 1 (стопові біти). Ще знімається сигнал "Скидання" з дільника частоти, він розпочинає збирати імпульси генератора тактової частоти і в період приймання шістнадцятого тактового імпульсу виробляє імпульс зсуву [25].

На вивідній лінії контролера "Дані" зберігається стан 0 (значення стартового біта) до того часу, поки не буде перший імпульс зсуву. Імпульс зсуву поміняє стан лічильника імпульсів зсуву і переписує в нульовий розряд зсувного регістру перший інформаційний біт байта даних що передавався. Стан який відповідає значенню цього біта, буде зберігатися на лінії "Дані" до кожного наступного імпульсу зсуву [26].

Так само будуть передані інші інформаційні біти, 1 стоповий біт і, нарешті, 2 стоповий біт, при передаванні якого лічильник імпульсів зсуву знову перейде в нульовий стан. Це приведе до запису 1 в регістр стану. Одиничний сигнал з виходу регістра на дозволить формуванню імпульсів зсуву, а також повідомляє процесор про готовність до приймання наступних даних. Після закінчення передавання наступного кадру (стартового біта, інформаційного байта і двох степових біт) контролер зберігає в лінії зв'язку рівень логічної одиниці (значення другого стопового біта).

Завдяки даній схемі зрозуміло як передаються дані на інтерфейсі UART. Потім по даній схемці буде розроблено і електронна схемка даного пристрою.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 24 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

2.3 Проектування структури приймача інтерфейсу UART

Оскільки передавач уже є зображений потрібно розробити і структурну схему приймача. Мною була розроблена делальна структурна схемка приймача інтерфейсу UART, який в свою чергу буде зображений на рисунку 2.3. Рівень логічної одиниці передається на лінії "Дані" в контролер для асинхронного прийому даних. Даний рівень задає умови для вироблення сигналу, що не дозволяє роботу дільника частоти генератора тактових імпульсів. Дійсно, після приймання попереднього байта даних лічильник імпульсів зсуву (лічильник по mod 9) стоїть в нульовому стані і на вентиль передаються два одиничні сигнали: з лічильника зрушень і з лінії "Дані". На виході вентилля виготовляється сигнал скидання дільника частоти сигналів тактового генератора, який не дозволяє формуванню імпульсів зсуву [27].

У період розпочатку передавання нового кадру на лінії "Дані" появиться рівень логічного нуля і цим буде знято сигнал скидання з дільника частоти. Стан чотирьох–розрядного подільника частоти буде змінюватися. Коли на лічильнику збереться значення вісім, він відправить сигнал, що перейде на входи регістра зсуву і лічильника імпульсів зсуву. Так як частота сигналів генератора тактових імпульсів приймача мусить бути такоюж як частотою генератора тактових імпульсів передавача, то читання біта буде приблизно посередині певного інтервалу, виділеного для передавання біта даних. Часу, який потрібний для вироблення 16 тактових імпульсів. Це все працює для змінення ймовірності помилки через певні відмінності частот генераторів приймача і передавача, спотворення форми відправлених сигналів [28]. Наступне зміщення відбудеться після проходження 16 тактових імпульсів. Посередині тимчасового інтервалу передавання першого інформаційного біта.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 25 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

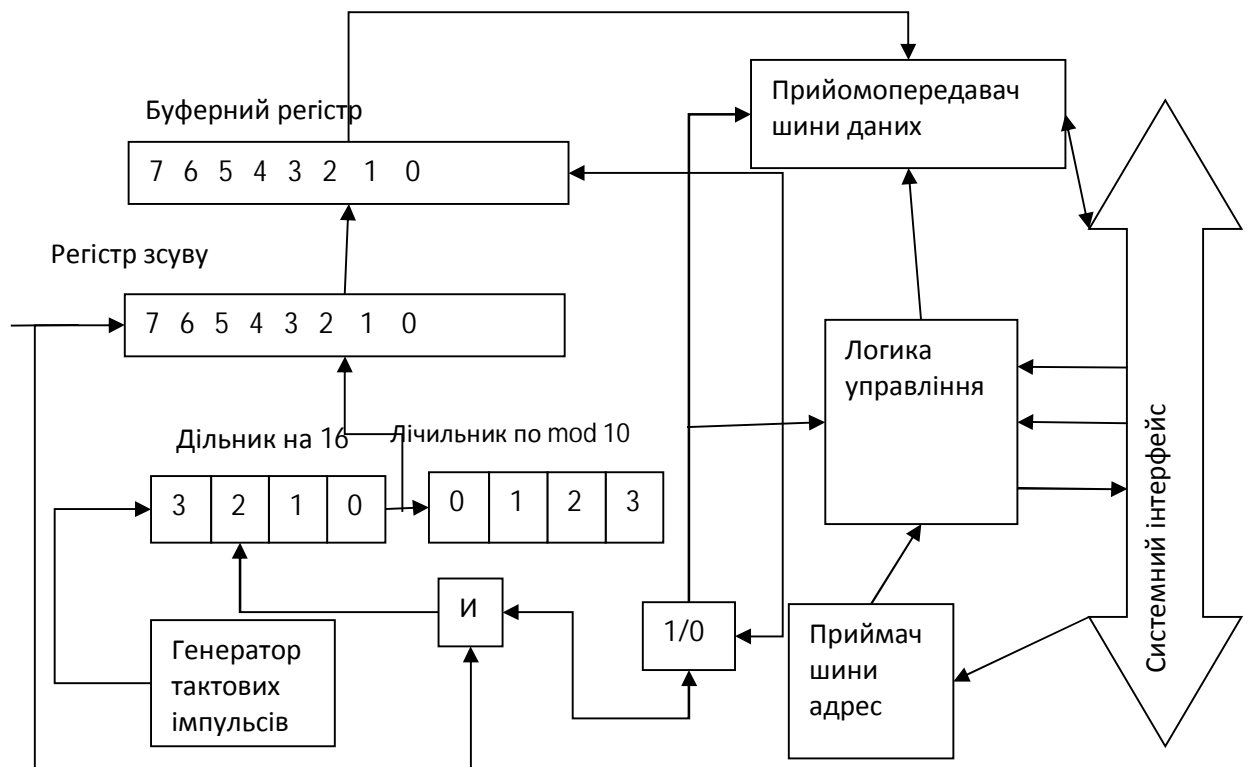


Рисунок 2.3 – Структурна схема приймача інтерфейсу UART

При прийманні в регістр зсуву 8 інформаційного біта з нього "виведеться" стартовий біт і в зсувний регістр буде переданий весь переданий байт інформації. В даний же момент лічильник імпульсів зсуву переведеться в нульовий стан і на його виході буде одиничний сигнал, за яким вміст регістру зсуву перейде в буферний регістр, а в регістр стану передаця одиниця і він буде повідомляти процесор про закінчення приймання наступного байта, вентиль приготується до виготовлення сигналу "Скидання" (цей сигнал створиться після надходження першого стопового біта) [29].

Получивши сигнал приготування, процесор задіє команду "Введення" . При цьому виготовляється управляючий сигнал системного інтерфейсу "Введення", за яким робляться передавання прийнятого байта даних з буферного регістра в процесор (сигнал "Читання") і скидання регістра стану.

Зазначу, що в контролері не має схеми керування степовими бітами прийнятого кадру. І не показані також схеми керування непарності або парності (паритету) надісланої інформації. В теперішніх контролерах є такі схеми, і якщо контролер не отримує з лінії зв'язку певної кількості степових бітів або

виготовляється сигнал помилки паритету в схемі контролю парності, то прийняті в поточному кадрі біти даних не використовуються і контролер чекає передавання нового стартового біта.

Обмін даними з зовнішнім приладом через послідовні лінії зв'язку використовується в мікро ЕОМ, а саме в тих випадках, коли не треба великої швидкості обміну [30]. З тим застосування в них послідовних ліній зв'язку з зовнішніми приладами є двома важливими причинами:

– послідовні лінії зв'язку прості по своїй організації: два дроти при симплексному і напівдуплексному передачі і максимум чотири – при дуплексному.

– в мікро ЕОМ використовуються зовнішні пристрої, обмін з якими необхідно вести в послідовному коді.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 27 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

3 АПАРАТНА РЕАЛІЗАЦІЯ КОНТРОЛЕРА ПОСЛІДОВНОГО АСИНХРОННОГО ІНТЕРФЕЙСУ

3.1 Моделювання роботи блоку зв'язку з системною шиною

Розробку та моделювання роботи схеми здійснимо з використанням системи автоматизованого проектування. Також поверхнево оглянемо деякі середовища і виберемо в якому будемо працювати далі.

Для моделювання можна використовувати різні програмні засоби: Proteus VSM, Electronics Workbench, LT Spice IV, Multisim та інші. Постає питання у тому, який програмний засіб кращий і який доцільніше використовувати під час моделювання .

Electronics Workbench 5 – розроблений фірмою Interactive Image Technologies. Вона дозволяє моделювати аналогово цифрові, прості і складні цифрові електромеханічних пристроїв, пристроїв автоматики, схем що складаються з блоків із певними передавальними функціями.

В ній є дуже велика бібліотека моделей різних пристроїв. Вона дає змогу створювати і відредаговувати принципові схеми, під'єднувати різні види вимірювальних приладів. Завдяки можливості підключати вимірювальні прилади ця програма дає змогу наблизитися до промислових зразків. Можна використовувати різні види вимірювальних приладів які будуть підключені. Вона дозволяє проектувати розрив компонентів у вигляді короткого замикання. Мені як користувачеві програма дає можливість налаштувати різні параметри схеми які потребуються при проектуванні. Для даної програми є величезна інтернет база в якій є понад дванадцять мільйонів компонентів.

Multisim – це програма яка дає можливість проектувати та моделювати електронні схемотехнічні елементи. Вона одна з небагатьох програм яка використовує SPICE–симуляцію, яка була розроблена в Берклі .

NI Multisim дає можливість виконувати розробку, тестування різних електричних приладів для навчання та для виробничих цілей. Дана програма

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 28 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

має найновіші версії ПЗ для аналізу різних електронних ланцюгів. Дана програма дозволяє поєднувати проектування та тестування цим самим даючи величезні можливості технологій віртуальних приладів. Її можна використовувати і для створення принципів електричних схем .

Вона має надзвичайно велику базу різних видів компонентів. Ще в новіших версіях має таку функцію яка вказує на помилки В ній зібрані різні види нових вимірювачів, а ще і підтримка моделей транзисторів.

PROTEUS VSM – набір програмних забів які призначені для моделювання електронних схем. Даний набір є системою схемотехнічного моделювання. Вона дає можливість моделювання, проектування роботів програмованих пристроїв. Бібліотека компонентів містить довідкові дані, а саме мікропроцесорів , мікроконтролерів. Також можна моделювати друковані плати. Дана програма є базкоштовною ознайомчою версією яка є повністю функціональною, але не можна зберігати файли. За допомогою даної програми можна підключатися до компютера .

Micro–Cap має зручний інтерфейс яким є простим для використання, а також не має великиг вимог до комп'ютера. Можна створювати власні макромоделі, які полегшують роботу в подальшій розробці, ще тут є демонстраційний режим який дає змогу зрозуміти як працює прилад.

Та є один дуже вагомий міну у студентській версії даної програми не можна створювати складі схеми тому, що не має достатньої кількості компонентів, можна обробляти невеликі обєми даних, також в ній відсутня програма складання, ще надзвичайно великий ряд обмежень що не дає змоги простому користувачу проектувати щось більш складне в даній схемі і це в свою чергу дость погано .

З огляду програмних засобів я виділив для себе Multisim. Тому, що Multisim:

1. Простий графічний редактор, що дозволяє досить просто малювати на екрані електронні схеми у звичному зображенні.
2. Велика бібліотека сучасних електронних компонентів

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 29 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

(дискретних, інтегральних аналогових, цифрових і змішаних аналого–цифрових). Бібліотека відкрита, легко може поповнюватися новими елементами, у тому числі й вітчизняними.

3. Велика бібліотека моделей електронних схем, що дозволяє використовувати готові практичні розробки й легко модернізувати їх під конкретне завдання.

В даній програмі я розробив блок зв'язку з системною шиною. Керування передаванням інфи по шині PCI виконується трьома сигналами – FRAME#, IRDY#, TRDY# остальні сигнали мають лише допоміжні ролі. В той час поки дані сигнали FRAME# і IRDY# відсутні, шина буде в стані спокою. Першим тактом після появи даного сигналу FRAME# є фаза адреси . В даний період усі девайси на шині перевіряють стан лінії AD і C / BE# з яким водночас з сигналом FRAME# надаються коди операцій, що будуть використані в даній транзакції. Період адреси проходить всього лиш один або два такти. По закінченню даної фази розпочинається фаза даних. Кожна фаза закінчується коли в одному такті будуть одночасно активні сигнали IRDY#, TRDY# і вони займають не менше одного такту. Джерело даних ставить на даній лінії AD потрібні дані, а на лінії C / BE# – сигнали, що демонструють, які саме байти AD містять реально передану інформацію. Після того він показує готовність певним сигналом. Приймач даних активним сигналом готовності (TRDY# при записі, IRDY# при читанні) оповіщає джерело про те, що він здатний здійснити прийом. Видання даних нічим не зв'язане з прийомом. Джерело передає їх дуже швидко і ставить сигнал готовності, після чого чекає на появу готовності у приймача, потім починається нова транзакція . На основі цього я розробив електронну схему блоку зв'язку з системною шиною, яка зображена на рисунку 3.1.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 30 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

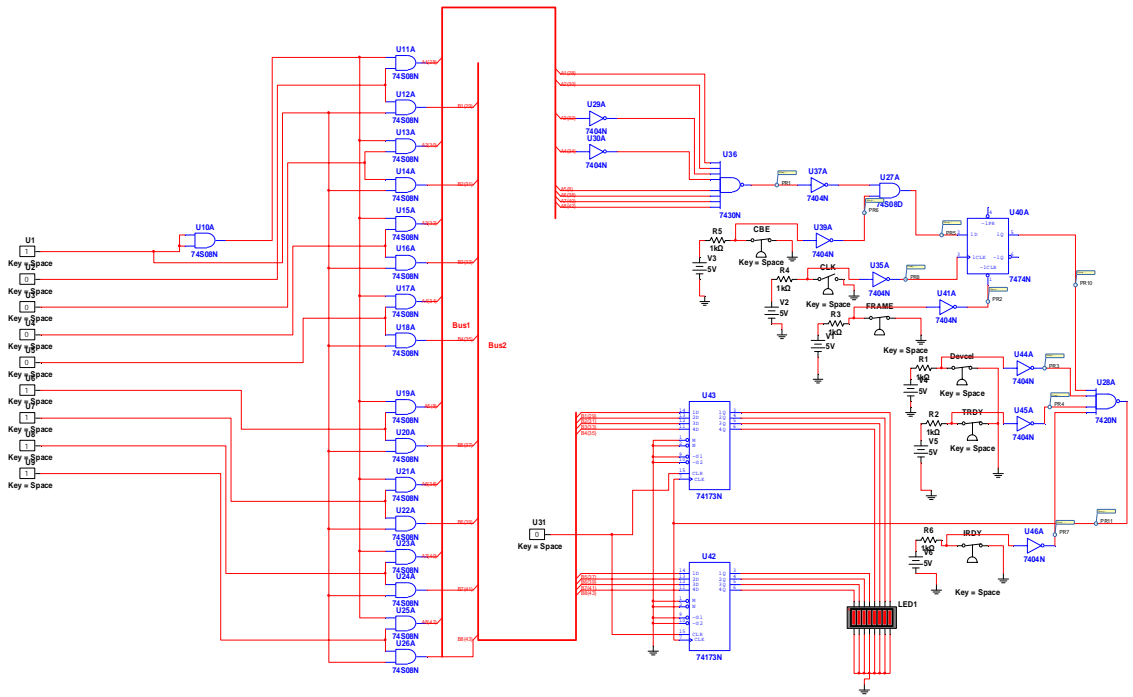


Рисунок 3.1 – Апаратна реалізація блоку зв'язку з системною шиною

Оскільки шина даних та адрес в інтерфейсі суміщені, то в функціональну схему вводиться мультиплексом. Дана схема блоку зв'язку з системною шиною забезпечить зв'язок комп'ютера з приймачем та передавачем контролера UART.

На ній зображено як дані адреси поступають на відповідний дешифратор адреси. Цією транзакцією ініціалізується зовнішній приймач інформації. Цей сигнал є керуючим сигналом, який разом з іншими відповідними управляючими сигналами забезпечує запис вивідної інформації в регістр.

3.2 Апаратна рефлізація UART передавача

Як було показано на рисунку передавача (рисунок 3.1), одним з основних компонентів передавача інформації виступають буферний та зсувний регістри. В якості регістрів пропонується обрати мікросхему універсального регістру 74198N, умовно-графічне позначення якої приведено на рисунку 3.2.

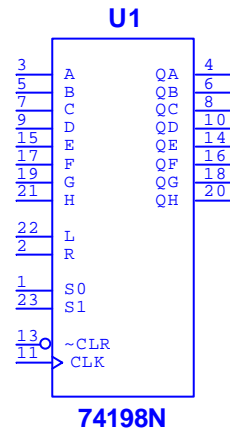


Рисунок 3.2 – Графічне позначення регістра 74198N

Таблиця 3.1 – Таблиця істиності

| Режим | Входи | | | | | | | Виходи | | |
|---------------------|-------|----|----|----|----|-----|----------------|----------------|--------------------------------|----------------|
| | R | S1 | S0 | DR | DL | C | D _n | Q ₀ | Q ₁ –Q ₆ | Q ₇ |
| Скидання | 0 | X | X | X | X | X | X | 0 | 0–0 | 0 |
| Зберігання | 1 | 0 | 0 | X | X | X | X | Q ₀ | Q ₁ –Q ₆ | Q ₇ |
| | 1 | 1 | 0 | X | 0 | 0–1 | X | Q ₁ | Q ₂ –Q ₇ | 0 |
| Зсув вліво | 1 | 1 | 0 | X | 1 | 0–1 | X | Q ₁ | Q ₂ –Q ₇ | 1 |
| | 1 | 0 | 1 | 0 | X | 0–1 | X | 0 | Q ₀ –Q ₅ | Q ₆ |
| Зсув вправо | 1 | 0 | 1 | 1 | X | 0–1 | X | 1 | Q ₀ –Q ₅ | Q ₆ |
| Паралельна загрузка | 1 | 1 | 1 | X | X | 0–1 | D _n | D ₀ | D ₁ –D ₆ | D ₇ |

Мікросхема 74198N це восьмизарядний синхронний зсувний регістр. Цей регістр може працювати в чотирьох режимах: режимі паралельної завантаження, режимах зсуву вліво і зсуву вправо (послідовна завантаження), режимі зберігання. Вибір режиму здійснюється входами S0 і S1.

Якщо на входах S0 і S1 встановлені низькі рівні (S0 = 0, S1 = 0), регістр знаходиться в режимі зберігання даних (зображено в таблиці 3.1).

Якщо на входах S0 і S1 встановлені високі рівні (S0 = 1, S1 = 1), регістр працює в режимі паралельної завантаження. У цьому режимі на входи даних D0 – D7 подається слово (8–розрядне число), яке потрібно записати в регістр.

Інформація з входів D0 – D7 записується в регістр і з'являється на виходах Q0 – Q7 по передньому фронту імпульсу синхронізації С. В режимі зсуву вправо ($S0 = 1, S1 = 0$) дані, які необхідно записати в регістр, послідовно (по одному біту) надходять на вхід DR. Запис кожного нового біта відбувається по передньому фронту імпульсу С, при цьому вміст регістра зсувається на один біт в сторону старшого розряду Q7, і нові дані записуються в молодший розряд Q0. 74198 У режимі зсуву вліво ($S0 = 0, S1 = 1$) записуються дані послідовно надходять на вхід DL. По передньому фронту імпульсу С вміст регістра зсувається на один біт в сторону молодшого розряду Q0, і нові дані записуються в старший розряд Q7. При роботі з регістром 74198N слід враховувати, що на входи S0, S1 перепад від високого рівня до низького можна подавати, тільки коли на вході С присутня напруга високого рівня. Скидання регістра здійснюється подачею 0 на асинхронний вхід R.

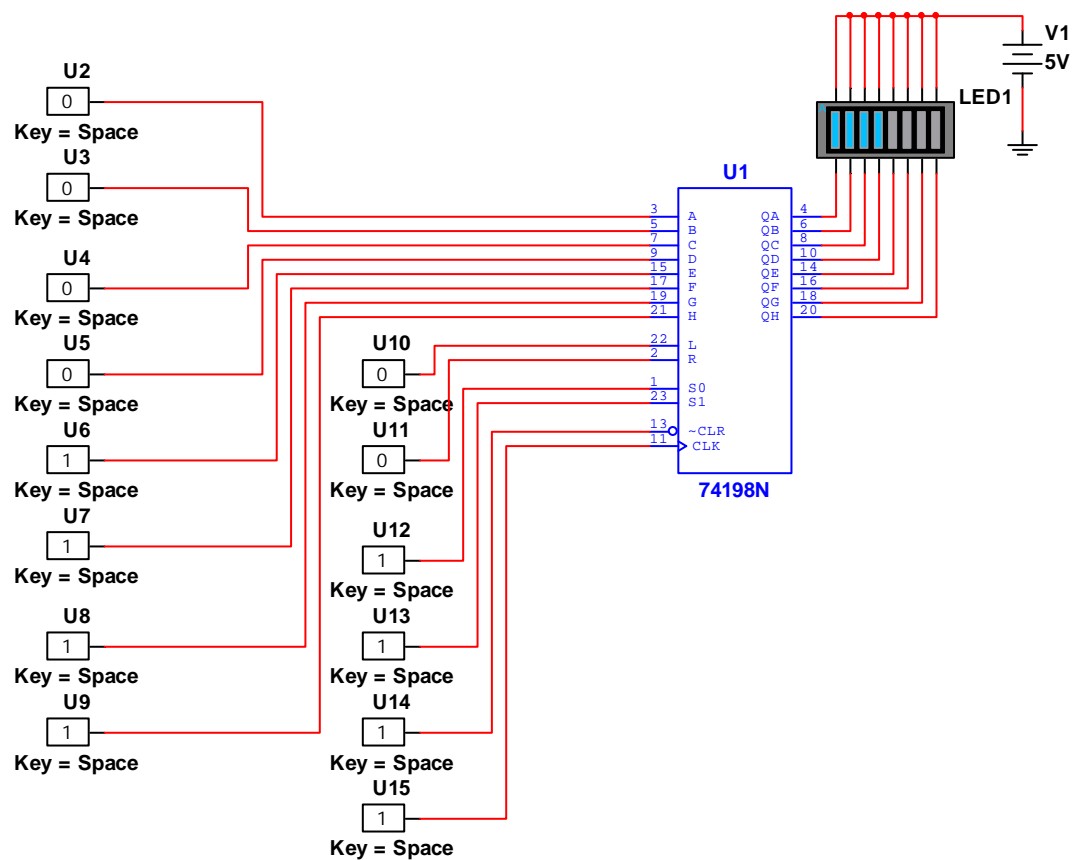


Рисунок 3.3 – схема роботи регістра при паралельному режимі

З метою використання регістра 74198N розглянемо його функціонування в трьох режимах – паралельного запису інформації її зсуву вліво та вправо. Схема дослідження паралельного запису інформації приведена на рисунку 3.3. В даній схемі на входах відповідно до таблиці 3.1 було вказано певні данні, які передавались на мікросхему. До мікросхеми було підключено лед діоди щоб на пряму бачити, як передається інформація через даний регістр. Також було досліджено роботу регістру в режимі зсуву вліво та зсуву вправо. Дану схемку роботи при зсуві вправо та вліво з зобразив на рисунку 3.4.

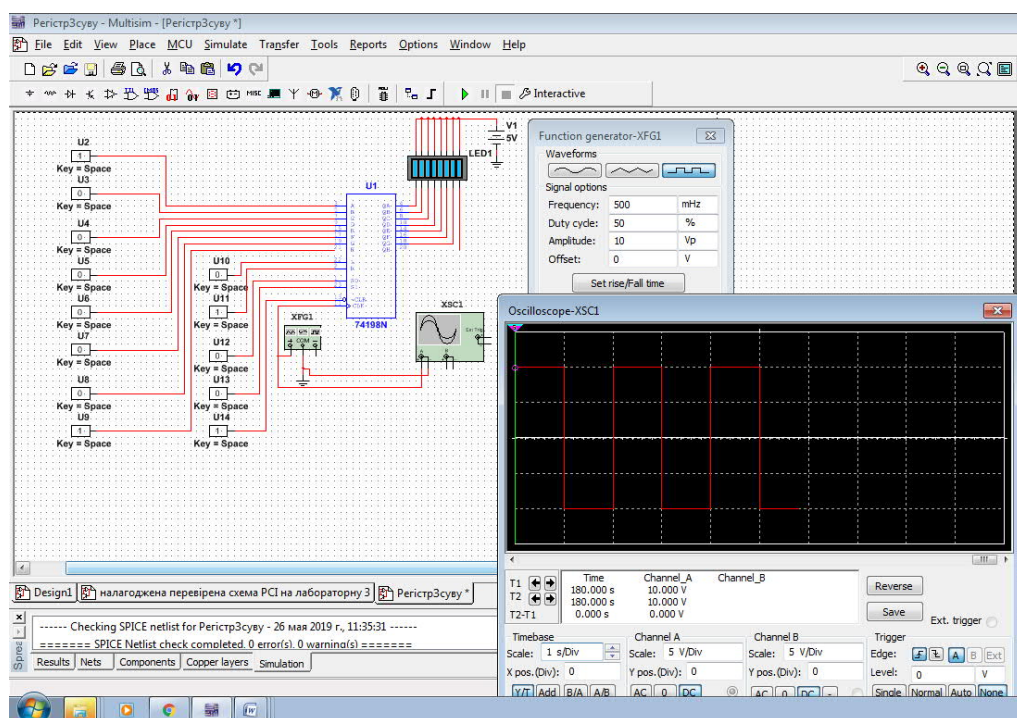


Рисунок 3.4 – Схема дослідження роботи регістру в режимах зсуву вліво та вправо

При зсуві вправо в налаштуванні схеми ми вказали що $S1=0$, $S0=1$. Запис кожного нового біта відбувається по передньому фронту імпульсу C, при цьому вміст регістра зсувається на один біт в сторону старшого розряду Q7, і нові дані записуються в молодший розряд Q0. А при зсуві вліво задали параметри $S0=0$, $S1=1$. Запис біта протилежний до запису біту при зсуві вправо.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 34 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

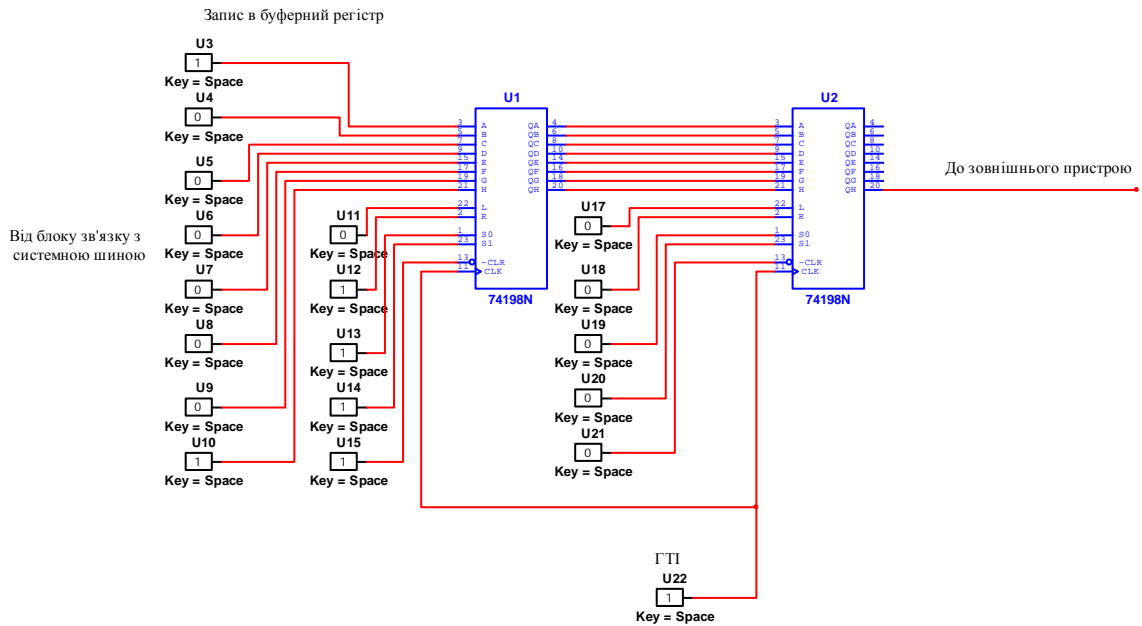


Рисунок 3.5 – Схема вузла передавача інформації по інтерфейсу UART

На даному етапі мною були розроблені передавачі інформації різних типів. Використовуючи дані схеми я зумів розробити схему передавального вузла контролера UART. Даний вузол зображений на рисунку 3.5.

На даній схемі видно що це паралельна послідовна передача інформації до зовнішнього користувача.

3.3 Апаратна реалізація UART приймача

На даному етапі мною був розроблений приймач який приймає дані від передавача та передає їх назад до блоку зв'язку, а саме на системну шину. Дана схемка зображена на рисунку 3.6. При побудові даної схеми нами були задані певні параметри за допомогою яких була налаштована дана схема. Ця схемка служить приймачем інформації як ви можете побачити самі на даному рисунку.

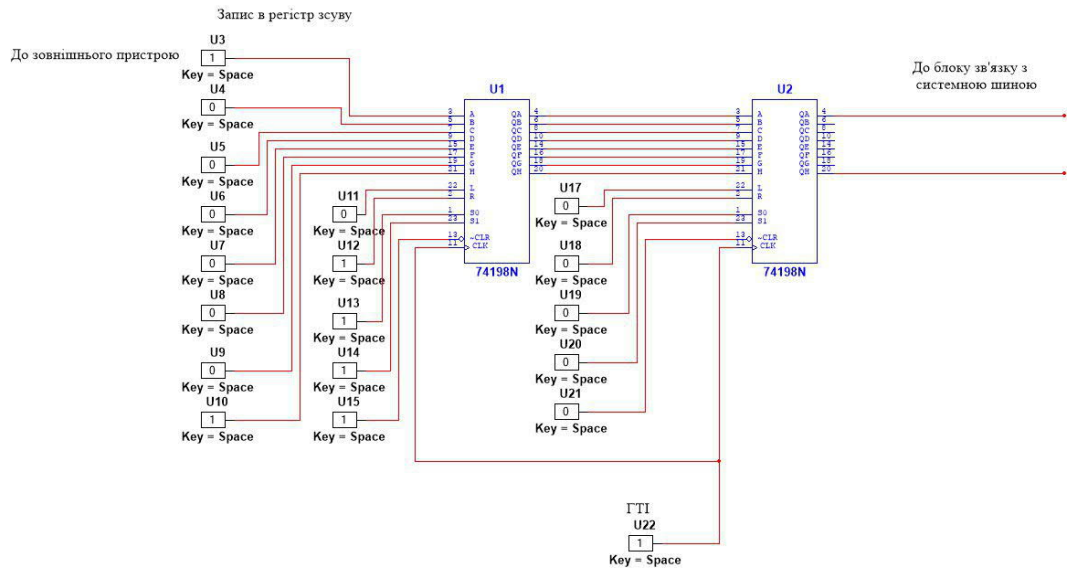


Рисунок 3.6 – схема приймача інформації по інтерфейсу UART

Після налаштування було проведено на працездатність дану схемку. Перевіряв я це також в програмному середовищі Multisim.

4 ТЕХНІКО–ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ПРИСТРОЮ

4.1 Стадії технологічного процесу

У розробці проектного рішення задіяні наступні спеціалісти – розробники, а саме: керівник проекту (К); студент (С); консультант техніко–економічного розділу (КТЕО).

Форму поділу робіт по всіх основних етапах і видах робіт, які повинні бути виконані показано в таблиці 4.1.

Таблиця 4.1 – Середній час виконання проекту та стадії технологічного процесу

| Назва операції (стадії) | Виконавець, посада | Середній час виконання операції, год. |
|--|-------------------------|---------------------------------------|
| Підготовка | Студент | 7 |
| Розробка проекту системи забезпечення наскрізного запису | Керівник ДП | 16 |
| | Консультант ТЕО, доцент | 2 |
| | Студент | 219 |
| Проектування технічної частини | Студент | 24 |
| Встановлення та налаштування прогр. Зас. | Студент | 8 |
| Тестування | Студент | 3 |
| Разом | | 279 |

4.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи

Витрати на оплату праці включають заробітну плату (ЗП) всіх категорій працівників, безпосередньо зайнятих на всіх етапах проектування.

Витрати на оплату праці розробників проекту визначаються за формулою:

$$B_{ОП} = \sum_i^N \sum_j^M n_{ij} \cdot t_{ij} \cdot C_{ij}, \quad (4.1)$$

де n_{ij} – чисельність розробників i -ої спеціальності j -го тарифного розряду, осіб;

t_{ij} – затрачений час на розробку проекту співробітником i -ої спеціальності j -го тарифного розряду, год;

C_{ij} – годинна ставка працівника i -ої спеціальності j -го тарифного розряду, грн.

Середньо годинна ставка працівника може бути розрахована за формулою:

$$C_{ij} = \frac{C_{ij}^0 (1+h)}{PЧ_i}, \quad (4.2)$$

де C_{ij} – основна місячна заробітна плата розробника i -ої спеціальності j -го тарифного розряду, грн.;

h – коефіцієнт, що визначає розмір додаткової заробітної плати (при умові наявності доплат);

$PЧ_i$ – місячний фонд робочого часу працівника i -ої спеціальності j -го тарифного розряду, год. (приймаємо 168 год.).

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 38 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

Таблиця 4.2 – Вихідні дані для розрахунку витрат на оплату праці

| Посада виконавців | Місячний оклад (стипендія), грн. | Коефіцієнт Додаткової з/п |
|---|--|---------------------------------|
| Керівник ДП, викладач | 7293 | 0,47 |
| Консультант техніко–економічного розділу, доцент | 7293 | 0,47 |
| Студент | 1400 | 0 |

Середня годинна ставка керівника та консультанта техніко–економічного розділу ДП дорівнює:

$$C_{ij} = \frac{7293 \cdot (1 + 0,47)}{168} = 20,4 \text{ (грн/год)} \quad (4.3)$$

Середня годинна оплата студента дорівнює:

$$C_{ij} = \frac{1400}{168} = 8,33 \text{ (грн/год)} \quad (4.4)$$

Витрати на оплату праці складають:

$$B_{ОП} = 16 \cdot 20,4 + 2 \cdot 20,4 + 261 \cdot 8,33 = 2166,53 \text{ (грн)} \quad (4.5)$$

Крім того, слід визначити відрахування на соціальні заходи. Величну відрахувань у спеціальні державні фонди визначають у відсотковому співвідношенні від суми основної та додаткової заробітних плат. Згідно діючого нормативного законодавства сума відрахувань у спеціальні державні фонди складає 20,5% від суми заробітної плати:

$$B_{\Phi} = \frac{20,5}{100} \cdot 2166,53 = 444,14 \text{ (грн)} \quad (4.7)$$

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 39 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

4.3 Розрахунок матеріальних витрат

Матеріальні витрати це вартість витрачених матеріалів, малоцінних та швидкозношуваних предметів на виробництво продукції, робіт або послуг, а також матеріалів і МШП, витрачених на адміністративні, збутові та інші потреби підприємства.

Загальна сума витрат на матеріальні ресурси (B_M) визначається за формулою:

$$B_M = \sum_{i=1}^n K_i \cdot C_i, \quad (4.8)$$

де K_i – витрата i -го типу матеріалу, натуральні одиниці вимірювання;

C_i – ціна за одиницю i -го типу матеріалу, грн.; i – тип матеріального ресурсу;

n – кількість типів матеріальних ресурсів.

Звідси, витрати на матеріальні ресурси дорівнюватимуть: 1744,9грн. Проведені розрахунки занесемо у таблицю 4.3.

Таблиця 4.3 – Зведені розрахунки покупних виробів

| Найменування матеріальних ресурсів | Од. виміру | Факт. витрачено матеріалів | Ціна за одиницю, грн. | Сума, грн. | Транспортні витрати (10% від суми) | Загальна сума, грн. |
|------------------------------------|------------|----------------------------|-----------------------|------------|------------------------------------|---------------------|
| Дешифратор 74154N | шт. | 1 | 5,7 | 5,7 | 0,57 | 6,27 |

Продовження таблиці 4.3

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 40 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

| Найменування матеріальних ресурсів | Од. виміру | Факт. витрачено матеріалів | Ціна за одиницю, грн. | Сума, грн. | Транспортні витрати (10% від суми) | Загальна сума, грн. |
|------------------------------------|------------|----------------------------|-----------------------|------------|------------------------------------|---------------------|
| Тригер 74ALS74AN | шт. | 1 | 17 | 17 | 1,17 | 18,17 |
| Регістр 74LS377N | шт. | 16 | 11 | 176 | 17,6 | 193,6 |
| Шинний формувач 74ALS245AN | шт. | 32 | 18,8 | 601,6 | 60,16 | 661,76 |
| 74ALS30AM 8-INPUT NAND | шт. | 2 | 6,25 | 12,5 | 1,25 | 13,75 |
| 74ALS32N 2-INPUT OR | шт. | 3 | 14 | 42 | 4,2 | 46,2 |
| 74ALS08M 2-INPUT AND | шт. | 1 | 5 | 5 | 0,5 | 5,5 |
| 74ALS00N 2-INPUT NAND | шт. | 1 | 10 | 10 | 1 | 11 |
| 74ALS11AM 3-INPUT AND | шт. | 1 | 7,9 | 7,9 | 0,79 | 8,69 |
| 74ALS04BM INVERTER | шт. | 1 | 4,6 | 4,6 | 0,46 | 4,46 |
| Паяльний набір | шт. | 1 | 500 | 500 | 50 | 550 |
| Папір (формат А4) | уп. | 1 | 70 | 70 | 7 | 77 |
| Тонер для принтера | уп. | 1 | 135 | 135 | 13,5 | 148,5 |
| Р а з о м | | | | | | 1744,9 |

4.4 Розрахунок витрат на електроенергію

Якщо для розробки КС використовується електрообладнання, то необхідно розрахувати витрати на електроенергію.

Загальна сума витрат на електроенергію розраховується за формулою:

$$B_E = \sum_{i=1}^n P_i \cdot k_i \cdot T_i \cdot Ц, \quad (4.9)$$

де P_i – паспортна потужність i -го електрообладнання, кВт;

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 41 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

k_i – коефіцієнт використання потужності i -го електрообладнання (приймається 0.7 , 0.9);

T_i – час роботи i -го устаткування за весь період розробки, год;

C – ціна електроенергії, грн / кВт· год;

i – тип електрообладнання;

n – кількість електрообладнання.

Для розробки дипломного проекту використовується один ноутбук потужністю $P = 0,5$ кВт, який за весь період розробки працює 100 годин, та друкуючий пристрій потужністю $P = 0,37$ кВт, який працює 2 години, і паяльник який працює 6 годин.

Проміжні розрахунки на витрату електроенергії подані в таблиці 4.4

Таблиця 4.4 – Витрати на електроенергію

| Найменування устаткування | Паспортна потужність, кВт | Коефіцієнт використання потужності | Час роботи обладнання для розробки, год | Ціна електроенергії, грн/кВт·год | Сума, грн. |
|---------------------------|---------------------------|------------------------------------|---|----------------------------------|------------|
| Ноутбук | 0,5 | 0,9 | 100 | 0,9 | 40,5 |
| Паяльник | 0,7 | 0,9 | 6 | 0,9 | 3,78 |
| Разом | | | | | 44,28 |

4.5 Обчислення накладних витрат

Накладні витрати пов'язані з обслуговуванням виробництва, утриманням апарату управління підприємства (фірми) та створення необхідних умов праці можуть становити 60–100 % від суми основної та додаткової заробітної плати працівників. Накладні витрати для даного проекту подані далі.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 42 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

$$B_{\Phi} = 0,7 \cdot B_{ОП}, \quad (4.10)$$

де H_B – накладні витрати.

$$H_B = 0,7 \cdot 2166,53 = 1516,57 \quad (4.11)$$

4.6 Інші витрати

Інші витрати є витратами, які не враховані в попередніх статтях. Вони складають 10% від заробітної плати:

$$I = 3815,88 \cdot 0,1 = 216,65 \quad (4.12)$$

4.7 Складання кошторису витрат та визначення собівартості

Загальні витрати ($K_{ПР}$) розрахуємо за формулою:

$$K_{ПР} = B_{ОП} + B_{\Phi} + B_M + B_{ЕЛ} + H_B + I \quad (4.13)$$

Тобто:

$$K_{ПР} = 6133,07 \quad (4.14)$$

Результати проведених розрахунків зведемо у таблицю 4.5.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 43 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

Таблиця 4.5 – Кошторис витрат

| Зміст витрат | Сума, грн. |
|--|------------|
| Витрати на оплату праці (осн. і дод. ЗП) | 2166,53 |
| Відрахування на соціальні заходи | 444,14 |
| Матеріальні витрати | 1744,9 |
| Витрати на електроенергію | 44,28 |
| Накладні витрати | 1516,57 |
| Інші витрати | 216,65 |
| Разом | 6133,07 |

4.8 Розрахунок ціни проекту

Договірна ціна (C_D) для проектних рішень розраховується за формулою:

$$C_D = B_{КС} \cdot \left(1 + \frac{p}{100}\right), \quad (4.15)$$

де $B_{КС}$ – кошторисна вартість, грн.;

p – середній рівень рентабельності, % (приймаємо 30% за погодженням з керівником).

$$C_D = 9440,03 \cdot 1,3 = 7972,99 \text{ (грн.)} \quad (4.16)$$

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 44 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

4.9 Визначення економічної ефективності і терміну окупності капітальних вкладень

Економічна ефективність – досягнення найбільших результатів за найменших затрат живої та уречевленої праці. Економічна ефективність є конкретною формою дії закону економії часу. За капіталістичного способу виробництва узагальнюючий показник економічної ефективності — норма прибутку.

Економічна ефективність – це співвідношення між отриманим прибутком та затраченими коштами. Вона обчислюється за формулою (4.17):

$$E_{\phi} = \Pi_p / K_B \quad (4.17)$$

де Π_p – очікуваний прибуток ;

$K_{\Pi P}$ – кошторисна вартість.

Очікуваний прибуток можна розрахувати із співвідношення:

$$\Pi_p = \text{Ц}_D - K_{\Pi P} = 7972,99 - 6133,07 = 1839,93(\text{грн.}) \quad (4.18)$$

Після проведених розрахунків отримуємо:

$$E_{\phi} = 1839,93 / 7972,99 = 0,23 \quad (4.19)$$

Поряд із економічною ефективністю розраховують термін окупності капітальних вкладень (T_p):

$$T_p = \frac{1}{E_{\phi}} \quad (4.20)$$

Тобто:

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 45 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

$$T_P = 1 / E_\phi = 1 / 0,23 = 4,3 \text{ (роки.)} \quad (4.21)$$

Прийнятним вважається термін окупності близький до 7 років. Розраховані економічні показники проекту занесемо до таблиці 4.6.

Таблиця 4.6 – Економічні показники розробки

| Показник | Значення |
|-------------------------|----------|
| Собівартість, грн. | 6133,07 |
| Плановий прибуток, грн. | 1839,93 |
| Ціна, грн. | 7972,99 |
| Економічна ефективність | 0,23 |
| Термін окупності, рік | 4,3 |

Враховуючи основні економічні показники з таблиці 4.6, можна зробити висновок, що при економічній ефективності 0,23 та терміні окупності – 4,3 роки проводити роботи по впровадженню даного контролера UART інтерфейсу є доцільним та економічно вигідним. Тому, з метою зниження вартості, варто було б здійснювати закупівлю обладнання у офіційних дилерів вказаних марок обладнання.

ВИСНОВОК

В процесі роботи над бакалавратською роботою, мною було досліджено одну з найважливіших складових комп'ютера – інтерфейси. Було опрацьовано та розглянуто велику кількість інформації стосовно комунікацій, інтерфейсів. Також мною було розглянуто різновидності інтерфейсів, різновиди їх підключення.

На основі вивчених даних було розроблено структурні схеми передавача та приймача послідовного асинхронного інтерфейсу. Уже на основі цієї структурної схеми було вибрано основні елементи нашої схеми. Було вивчено їхні даташити, способи та режими їхньої роботи, а їхню роботу було перевірено в програмному забезпеченні NI Multisim 14.0, яке для мене уже було знайомим з попередніх курсів.

В процесі реалізації було розроблено п'ять основні схеми, які теж були реалізовані в NI Multisim 14.0. Першою схемою був блок зв'язку з системною шиною. Другою була схема за допомогою якої я досліджував роботу регістру при паралельному режимі передавання інформації. На третій схемі дослідили роботу схеми регістра в режимі зсуву вправо та вліво. На четвертій схемі передавача інформації по інтерфейсу UART. На п'ятій схемі приймача інформації по інтерфейсу UART.

В ході роботи над техніко–економічним розділом було обраховано можливі витрати і обраховано приблизну вартість пристрою контролера UART інтерфейсу. Було виявлено, що схема є економічно ефективною. В додатку Б розміщена довідка про використання, що так само підтверджує якість пристрою. А в додатку А є тези з конференції, які були написані в результаті роботи над бакалавратською роботою.

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 47 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мельник А.О. Архітектура комп'ютера: навчальний посібник. Луцьк, 2008. С. 307–397.
2. Гук М. Ю. Аппаратные средства IBM PC. Энциклопедия. Санкт–Петербург, 2006. С. 334–398.
3. Жуйков В.Я., Терещенко Т.О., Ямненко Ю.С., Заграничний А.В. Мікропроцесорна техніка: навчальний посібник. Київ, 2016. 168 с.
4. Асинхронний послідовний інтерфейс: веб–сайт URL: <https://studfiles.net/preview/3741158/page:26/> (дата звернення: 12.02.19).
5. Підключення інтерфейсів та їх проектування: веб–сайт. URL: <https://studfiles.net/preview/3907471/page:144/> (дата звернення: 13.02.19).
6. Інтерфейси в комп'ютері та її специфікації: веб–сайт. URL: https://studopedia.su/2_14286_elektronna–ramyat.html (дата звернення 13.02.19).
7. Оптимізація роботи інтерфейсів: веб сайт. URL: <https://www.br.com.ua/kurs/Computers/52199–1.html> (дата звернення 17.03.19).
8. Класифікація інтерфейсів та її специфікації: веб–сайт. URL: https://studopedia.com.ua/1_24947_klasifikatsiya–elektronnoi–ramyati.html (дата звернення: 17.02.19).
9. Типи інтерфейсів їх класифікації. Коди корекції помилок: веб–сайт. URL: <http://www.wikipage.com.ua/1x8f9f.html> (дата звернення: 18.03.19).
10. Архітектура персонального комп'ютера: веб–сайт. URL https://elearning.sumdu.edu.ua/free_content/lectured:1a259358378153792bb8645df287e86d790fc40d/20160903092057/44894/index.html (дата звернення: 18.03.19).
11. Структура асинхронних послідовних інтерфейсів: веб–сайт. URL: https://studopedia.su/2_14287_shvidkodiya–i–produktivnist–ramyati.html: (дата звернення: 2.05.19).
12. Послідовні та паралельні інтерфейси: веб–сайт. URL: <https://studfiles.net/preview/3907471/page:128/> (дата звернення: 5.05.19)

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата | | 48 |

13. Послідовні інтерфейси їх характеристики: веб-сайт. URL: <http://ua-referat.com/%9A%D0%B5%D1%8D0%9A%D0%B5%D1%88-%D0%BF%D0%B0%D0%BC%60%D1%8F%D1%82%D1%8C> (дата звернення: 10.02.19).

14. Оперативна (основна Main Memory) пам'ять (ОП): веб-сайт. URL: <http://um.co.ua/4/4-11/4-119672.html> (дата звернення: 5.05.19).

15. Взаємодія інтерфейсів з зовнішніми приладами: веб-сайт. URL: <https://studfiles.net/preview/3907471/page:60/> (дата звернення: 10.05.19).

16. Значення інтерфейсів. Як вони працюють: веб-сайт. URL: <https://gadgetsroom.ru/the-value-of-the-cache-memory-how-the-cache-works/> (дата звернення: 10.05.19).

17. Верхній рівень структурної організації комп'ютера: веб-сайт. URL: <http://lib.exdat.com/docs/756/index-859-16.html?page=3> (дата звернення: 10.05.19).

18. Класифікація інтерфейсів та типи їх підключення: веб-сайт. URL: <http://pzks.nmu.org.ua/ua/labs/os5.pdf> (дата звернення: 10.04.19).

19. Підключення інтерфейсів: веб-сайт. URL: <https://biglib.info/5434-keshuvannya-ramyut.html> (дата звернення: 18.05.19).

20. Асинхронні послідовні інтерфейси їх підключення: веб-сайт. URL: <https://studopedia.org/6-105219.html> (дата звернення: 10.05.19).

21. Гросуляк П.І Архітектура інтерфейсів: веб-сайт. URL: <https://homework.net.ua/organizatsiya-ram-yati-protsesora/> (дата звернення: 10.05.19).

22. Бондарець Д.В. Моделювання електричних схем з використанням SPICE – моделей / Каштелян І.В., Данило А.М. // Тези доповідей Науково-практичної конференція молодих вчених і студентів «Інтелектуальні комп'ютерні системи та мережі». ч.2 – Тернопіль: ТНЕУ. – 2019. – С. 16. NI Multisim: веб-сайт. URL: <http://cxem.net/software/multisim.php> (дата звернення: 10.04.19).

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 49 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |

23. NI Multisim: веб-сайт. URL: https://en.wikipedia.org/wiki/NI_Multisim
(дата звернення: 10.04.19).

24. An FPGA Approach to Quantifying Coherence Traffic Efficiency on Multiprocessor Systems: веб-сайт. URL: https://www.researchgate.net/publication/4290181_An_FPGA_Approach_to_Quantifying_Coherence_Traffic_Efficiency_on_Multiprocessor_Systems (дата звернення: 10.04.19).

25. Пирамидальный дешифратор и его особенности: веб-сайт. URL: <https://www.ngpedia.ru/id6p3.html> (дата звернення: 10.05.19).

26. Дешифратори. Класифікація: веб-сайт. URL: <https://studopedia.org/2-150638.html> (дата звернення: 10.04.19).

27. Методичні вказівки до оформлення курсових проектів, звітів про проходження практики, випускних кваліфікаційних робіт для студентів спеціальності «Комп'ютерна інженерія» / І.В. Гураль, Л.О. Дубчак / Під ред. О.М. Березького. Тернопіль: ТНЕУ, 2019. 33 с.

28. Методичні рекомендації до виконання дипломного проекту з освітньо-кваліфікаційного рівня «Бакалавр» напряму підготовки 6.050102 «Комп'ютерна інженерія» фахового спрямування «Комп'ютерні системи та мережі» / О.М. Березький, Л.О. Дубчак, Г.М. Мельник, Ю.М. Батько, С.В. Івасьєв / Під ред. О.М. Березького. Тернопіль ТНЕУ, 2016, 60с.

29. Методичні вказівки до написання техніко-економічного розділу дипломних проектів освітньо-кваліфікаційного рівня «бакалавр» підготовки 6.050102 комп'ютерна інженерія/ І.Р. Паздрій Тернопіль: ТАНГ, 2014. 37 с.

30. Підключення інтерфейсів: веб-сайт. URL: <https://studopedia.org/2-150638.html> (дата звернення: 10.06.19).

| | | | | | | |
|-----|------|----------|--------|------|---------------------------------|------|
| | | | | | БР.КСМ. . 07107/15.00.00.000 ПЗ | Арк. |
| | | | | | | 50 |
| Зм. | Арк. | № докум. | Підпис | Дата | | |