

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“Дніпровська політехніка”**

**ІНТЕРФЕЙСИ ТА ТЕХНОЛОГІЇ РАДІОЗВ'ЯЗКУ ІЗ ЗАХИСТОМ
ІНФОРМАЦІЇ**

ОСНОВИ ЗБОРУ, ОБРОБКИ ТА ПЕРЕДАЧІ ІНФОРМАЦІЇ

Методичні рекомендації до виконання лабораторних робіт

студентами денної форми навчання
за освітньо-професійною програмою підготовки бакалаврів за спеціальностями
151 Автоматизація та комп'ютерно-інтегровані технології,
123 Комп'ютерна інженерія

Дніпро
2019

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“Дніпровська політехніка”



ІНСТИТУТ ЕЛЕКТРОЕНЕРГЕТИКИ
Факультет інформаційних технологій
Кафедра автоматизації комп'ютерних систем

**ІНТЕРФЕЙСИ ТА ТЕХНОЛОГІЇ РАДІОЗВ'ЯЗКУ ІЗ ЗАХИСТОМ
ІНФОРМАЦІЇ**

ОСНОВИ ЗБОРУ, ОБРОБКИ ТА ПЕРЕДАЧІ ІНФОРМАЦІЇ

Методичні рекомендації до виконання лабораторних робіт

студентами денної форми навчання
за освітньо-професійною програмою підготовки бакалаврів за спеціальностями
151 Автоматизація та комп'ютерно-інтегровані технології,
123 Комп'ютерна інженерія

Дніпро
НТУ «ДП»
2019

Інтерфейси та технології радіозв'язку із захистом інформації. Основи збору, обробки та передачі інформації. Методичні рекомендації до виконання лабораторних робіт студентами денної форми навчання з освітньо-професійної програми підготовки бакалаврів за спеціальностями 151 Автоматизація та комп'ютерно-інтегровані технології, 123 Комп'ютерна інженерія / А.В. Кожевников, Д.С. Зибалов, В.І. Шевченко, В.В. Надточий ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро: НТУ "ДП", 2019. – 31 с.

Автори:

А.В. Кожевников, канд. техн. наук, доц.;

Д.С. Зибалов, асист.;

В.І. Шевченко, ст. викл.;

В.В. Надточий, ст. викл.

Затверджено до видання редакційною радою університету (протокол № 4 від 19.04.2019) за поданням методичної комісії за спеціальностями 151 Автоматизація та комп'ютерно-інтегровані технології, 123 Комп'ютерна інженерія (протокол № 5 від 11.04.2019).

Навчально-методичний посібник дає уявлення про інтерфейси, які використовуються в сучасних системах збирання й обробки інформації. Розглянуто послідовний інтерфейс передачі інформації (CAN) та технології радіозв'язку із захистом інформації, апаратні та програмні специфікації цих інтерфейсів. Велику увагу приділено оптимізації організації введення – виведення в інтерфейсах різних сімейств і поколінь.

Відповідальний за випуск завідувач кафедри автоматизації та комп'ютерних систем В.В. Ткачов, д-р техн. наук, проф.

ЗМІСТ

Вступ.....	4
Опис лабораторної установки.....	5
Лабораторна робота №1. Інтерфейс CAN (Controller Area Network).....	6
Лабораторна робота №2. Використання псевдовипадкових послідовностей в широкосмугових системах передачі інформації	22
Список використаної літератури.....	30

ВСТУП

Метою цього курсу лабораторних робіт є навчання студентів практичному застосуванню знань за матеріалом основних розділів дисципліни "Основи збирання, передачі і обробки інформації".

Основними завданнями курсу слід вважати практичне освоєння студентами принципів роботи послідовного інтерфейсу CAN та технології радіозв'язку з захистом інформації.

Курс включає дві лабораторні роботи, розрахованих на дві академічні години кожна (враховуючи час на отримання допуску до роботи). Перед тим, як приступити до виконання лабораторної роботи, студент повинен отримати допуск, надавши відповіді на питання викладача або пройти відповідний тест. За результатами виконання лабораторної роботи студент повинен скласти звіт згідно з приведеними вимогами до змісту і оформлення, а також захистити роботу, надавши відповіді на питання викладача. Лабораторні роботи виконуються кожним студентом індивідуально.

Опис лабораторної установки









Лабораторна установка складається з одного передавального і одного приймаючого напівкомплектів. Індикація реалізована на двох рідкокристалічних дисплеях типу RDX0032. Напівкомплекти зібрані на основі бюджетного мікроконтролера PIC18F4580 фірми Microchip.

Зовнішній вигляд лабораторної установки приведений на рис. 1.



Рисунок 1 – Зовнішній вигляд лабораторної установки

Контроллер містить 4 послідовні інтерфейси обміну інформацією. На основі UART за допомогою інтерфейсних мікросхем реалізовано три інтерфейси RS232, RS485, CL (Струмова петля). Інтерфейси I²C і SPI на цьому мікроконтролері реалізовані на одному модулі. Одночасне використання неможливе. CAN інтерфейс на цьому кристалі версії 2,0 V.

На нижній частині стенду знаходяться роз'єми для підключення осцилографа. Після подачі живлення на лабораторний макет передавальний і приймаючий напівкомплекти знаходяться в режимі очікування. На рідкокристалічних індикаторах відображується напис OFF. Для початку роботи із стендом необхідно спочатку включити за допомогою кнопки  приймальний напівкомплект (він знаходиться в нижній лівій частині стенду) і за допомогою кнопок  або  вибрати потрібний інтерфейс. Потім включаємо передавальний напівкомплект натискаючи на кнопку . Вибираємо потрібний інтерфейс за допомогою кнопки , послідовно натискаючи на цю кнопку. За допомогою кнопок  і  вибираємо свій варіант, який на рідкокристалічному дисплеї відображується в десятковій формі числення. Повідомлення передається циклічно одне за іншим. Під кнопкою включення знаходиться кнопка скидання .

ЛАБОРАТОРНА РОБОТА № 1

Тема: Інтерфейс CAN (Controller Area Network)

Мета: Вивчити принцип функціонування протоколу передачі інформації

1.1 Теоретичні відомості

1.1.1 Призначення інтерфейсу CAN

На початку восьмидесятих років фірма BOSCH розробила протокол "Control Area Network" (CAN), який був затверджений Міжнародною організацією по стандартах. Перша презентація цієї шини була проведена в 1986 році в місті Детройті на конференції співтовариства інженерів автомобілебудування (SAE).

У всіх високотехнологічних системах сучасного автомобіля застосовується CAN-протокол для зв'язку ЕБУ з додатковими пристроями і контроллерами виконавчих механізмів і різних систем безпеки. У деяких автомобілях CAN зв'язує ІММО, приладові панелі, SRS блоки і т. д. Також протокол CAN ISO 15765-4 увійшов до складу стандарту OBDII.

CAN протокол отримав всесвітнє визнання як універсальна, ефективна, надійна і економічно прийнятна платформа для майже будь-якого типу зв'язку даних в пересувних системах, машинах, технічному устаткуванні і індустріальній автоматизації. Заснована на базі протоколів високого рівня CAN-технологія успішно конкурує на ринку розподілених систем автоматизації. Протокол CAN застосовується в real-time системах для вирішення різних завдань. Зараз розвиваються декілька видів CAN протоколів високого рівня, таких як CAL, OSEK/VDX, SAE J1939, CANopen, DeviceNet, SDS, CAN-Kingdom, в основі яких лежить каналний протокол CAN2.0 (Bosch), і на основі цих протоколів можна вирішувати проблеми, що виникають в real-time системах, які неможливо вирішити за допомогою інших відомих протоколів, скажімо, TCP/IP.

1.1.2 Характеристики інтерфейсу

Характеристики інтерфейсу наведені в табл. 1.1. Максимальне число абонентів, які підключені до цього інтерфейсу фактично визначається здатністю навантаження застосованих приймачів.

Наприклад, при використанні трансивера фірми PHILIPS PCA82C250 до лінії можна підключити 110 приймачів. Будь-який приймач може реагувати як на один ідентифікатор, так і на декілька. На один ідентифікатор можуть реагувати декілька приймачів.

Протокол CAN має розвинену систему виявлення і сигналізації помилок. Для цих цілей використовується порозрядний контроль, пряме заповнення бітового потоку, перевірка пакету повідомлення CRC-поліномом, контроль форми пакету повідомлень, підтвердження правильного прийому пакету даних.

Хемінговий інтервал $d=6$. Система арбітражу протоколу CAN виключає втрату інформації і часу при "зіткненнях" на шині.

Таблиця 1.1 – Характеристики інтерфейсу CAN

Стандарт	ISO 11898
Швидкість передачі	1 Мбіт/с (максимум)
Відстань передачі	1000 м (максимум)
Характер сигналу, лінія передачі	диференціальна напруга, скручена пара
Кількість драйверів	64
Кількість приймачів	64
Схема з'єднання	напівдуплекс, багатоточкова

Інтерфейс із застосуванням протоколу CAN легко адаптується до фізичного середовища передачі інформації. Це може бути диференціальний сигнал, оптоволокну, просто відкритий колектор.

1.1.3 Стандарт

Під термінами "CAN стандарт" або "CAN протокол" розуміють функціональні можливості, які стандартизовані ISO 11898. Цей стандарт об'єднує фізичний рівень (Physical Layer) і рівень каналу даних (Data Link Layer) відповідно до 7-и рівневої OSI моделі. Таким чином, "CAN стандарт" відповідає рівню мережевого інтерфейсу в 4-х рівневій моделі TCP/IP. Проте, практична реалізація навіть дуже простих розподілених систем на базі CAN показує, що окрім сервісів рівня каналу даних, що надаються, вимагаються ширші функціональні можливості: передача блоків даних завдовжки більш ніж 8 байтів, підтвердження пересилки даних, розподіл ідентифікаторів, запуск мережі і функції супервізора вузлів. Оскільки ці додаткові функціональні можливості безпосередньо використовуються прикладним процесом, вводиться поняття рівня застосувань (Application Layer) і протоколів високого рівня. Зазвичай їх і називають терміном "CAN протоколи".

1.1.4 Позначення і підключення пристроїв

Промислова мережа реального часу CAN є мережею із загальним середовищем передачі даних. Це означає, що усі вузли мережі одночасно приймають сигнали, які передаються по шині. Неможливо послати повідомлення якому-небудь конкретному вузлу. Усі вузли мережі приймають увесь трафік, який передається по шині.

Кожен вузол складається з двох складових. Це власне CAN контроллер, який забезпечує взаємодію з мережею і реалізує протокол, і мікропроцесор (CPU). CAN контроллери з'єднуються за допомогою диференціальної шини, яка має дві лінії, – CAN_H (can-high) і CAN_L (can-low), по яких передаються сигнали.

Логічний нуль реєструється, коли на лінії CAN_H сигнал вищий, ніж на лінії CAN_L. Логічна одиниця – у разі, коли потенціали ліній CAN_H і CAN_L однакові. Використання такої диференціальної схеми передачі робить можливою роботу CAN мережі в дуже складних зовнішніх умовах. Логічний нуль називається домінантним бітом, а логічна одиниця – рецесивним. Ці назви відбивають пріоритет логічної одиниці і нуля на шині CAN. При одночасній передачі в шину логічного нуля і одиниці, на шині буде зареєстровано тільки логічний нуль (домінантний сигнал), а логічна одиниця буде пригнічена (рецесивний сигнал).

Для з'єднання пристроїв використовують кабель типу вита пара. На обох кінцях витої пари мають бути погоджуючі резистори, опір яких дорівнює хвилевому опору кабелю. Для збільшення довжини, кількості вузлів або гальванічної розв'язки можуть бути використані повторювачі інтерфейсу, мережеві мости і шлюзи. Топологія мережі шинна, максимальна довжина відведення від шини при швидкості передачі 1 Мбіт/з не повинна перевищувати 30 см. Основні вимоги до лінії передачі і її характеристик близькі до RS485, проте в передавачах CAN є режим управління тривалістю фронтів імпульсів. Збільшення тривалості фронту дозволяє понизити вимоги до узгодження лінії на низьких частотах, збільшити довжину відведень і ослабити випромінювання електромагнітних завад. Виводи заземлень усіх передавачів мережі мають бути сполучені (якщо інтерфейси гальванічно не ізольовані). При цьому різниця потенціалів між виводами заземлень не повинна перевищувати 2 В. Гальванічна ізоляція рекомендується при довжині лінії більше 200 м, але не є обов'язковою вимогою стандарту.

Для електричного з'єднання пристроїв з CAN інтерфейсом стандарт передбачає два варіанти.

Перший варіант полягає в застосуванні Т-подібних розгалужувачів, які складаються з трьох 9-штирькових роз'ємів D, – sub, розташованих в одному корпусі, однойменні контакти яких сполучені між собою.

Другий варіант вимагає наявності в кожному CAN -пристрої двох роз'ємів. Для включення пристрою в мережу кабель розрізають і на його кінцях встановлюють частини у відповідь роз'ємів. Погоджуючий резистор повинен розташовуватися усередині роз'єму, який підключається до кінця кабелю. Для стандарту CANopen передбачена набагато більша різноманітність варіантів роз'ємів, у тому числі RJ10, RJ45, роз'ємний гвинтовий клемник, і ще більше десяти варіантів. На рис. 1.1 представлено підключення чотирьох пристроїв.

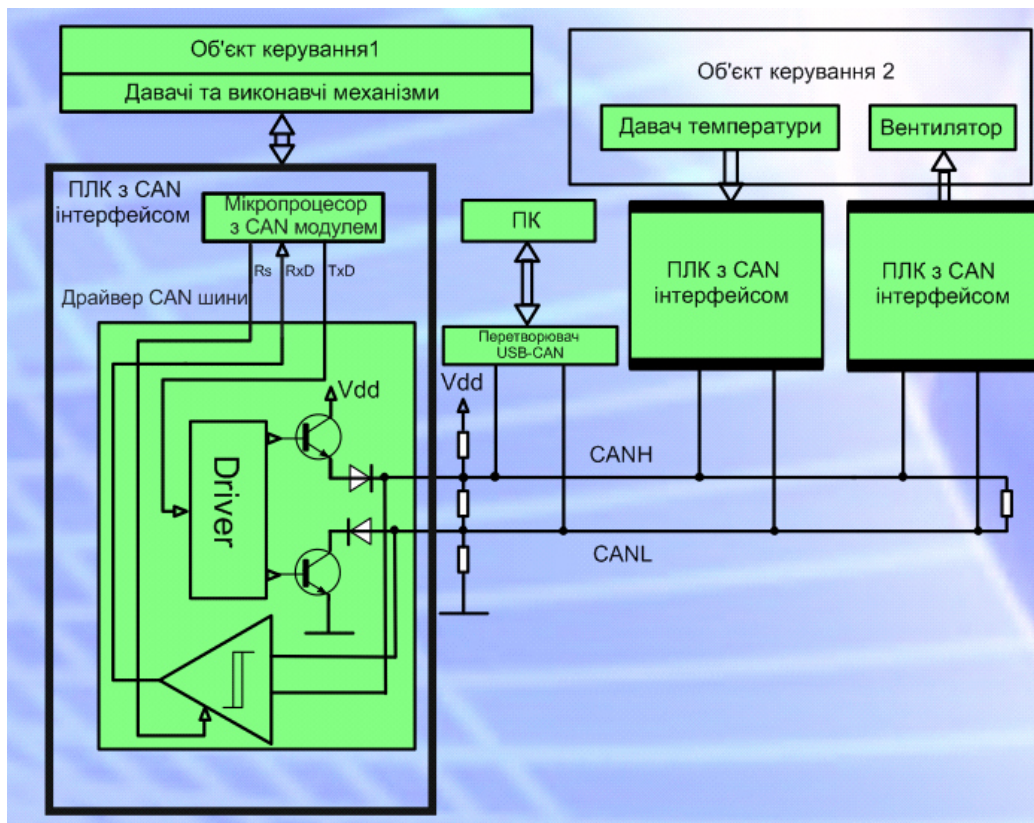


Рисунок 1.1 – Схема підключення чотирьох пристроїв

Приймач (трансивер) призначений для сполучення CAN контролера і фізичної шини. Він забезпечує прийом і передачу сигналу. Прийом даних здійснюється за допомогою компаратора, який забезпечує розпізнавання доміантного і рецесивного станів CAN-шини. Передача даних відбувається через передавальний тракт, що містить таймер, схему завдання крутизни сигналу, вузол управління передавачем, вихідний каскад на транзисторах і схему температурного захисту. Таймер призначений для обмеження тривалості доміантного стану. Формувач фронтів обмежує швидкість їх наростання, знижуючи електромагнітне випромінювання, і дозволяє відмовитися від зовнішніх синфазних дроселів.

Вузол управління передавачем здійснює управління двома транзисторами, що формують стан на лініях CAN-H і CAN-L. У доміантному стані обидва транзистори відкрито, забезпечуючи на лінії CAN-H високий рівень, а на лінії CAN-L – низький. У разі рецесивного стану відкритий тільки транзистор, що підтягує лінію до плюса живлення.

Схема температурного захисту оберігає мікросхему від виходу з ладу, відключаючи передавач, якщо температура кристала перевищує значення 160°C.

Для абстрагування від середовища передачі специфікація CAN уникає опису двійкових значень як "0" і "1", (рис. 1.2). Замість цього, CAN визначає два логічні стани: рецесивний (recessive) і доміантний (dominant).

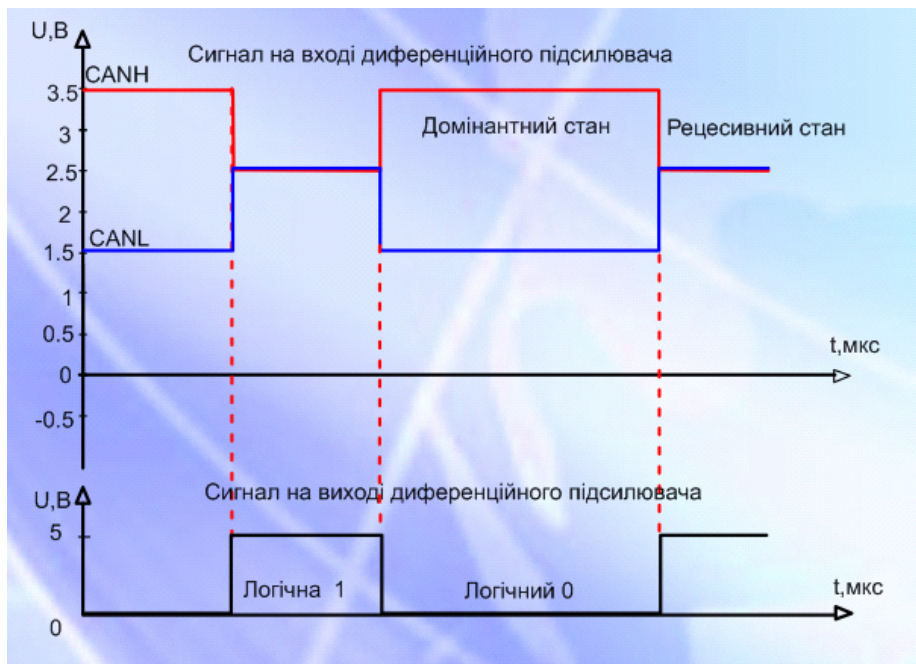


Рисунок 1.2 – Осцилограми сигналів на шині

Наприклад, при реалізації фізичного рівня на радіоканалі відсутність сигналу означає рецесивний біт, а наявність – доміантний, тоді як в стандартній реалізації дротяної мережі рецесив буває за наявності сигналу, а доміант, відповідно, за відсутності. Стандарт мережі вимагає від "фізичного рівня" фактично єдиної умови: щоб доміантний біт міг подавити рецесивний, але не навпаки. Логічне значення доміантного біта – "0". Логічне значення рецесивного біта – "1".

1.1.5 Швидкість передачі і максимальна відстань

Співвідношення між швидкістю передачі і максимальною довжиною кабелю приведене в табл. 1.2. Обмеження довжини кабелю пов'язане з кінцевою швидкістю світла і механізмом побітового арбітражу. Під час арбітражу усі вузли мережі повинні отримувати поточний біт передачі одночасно, тобто сигнал повинен встигнути поширитися по усьому кабелю за одиничний відлік часу в мережі.

Таблиця 1.2 – Залежність швидкості передачі інформації від відстані

Швидкість передачі, Кбіт/с	Довжина, м
1000	40
500	100
250	200
125	500
10	6000

Час поширення сигналу залежить від довжини шини, вхідної затримки компаратора і вихідної затримки формувачів.

Кількість вузлів в мережі CAN теоретично не обмежена. Практично ж максимальна кількість обмежується на рівні 127.

1.1.6 Адресація і протоколи високого рівня

У CAN не існує явної адресації повідомлень і вузлів. Протокол CAN ніде не вказує, що поле арбітражу (Identification field + RTR) повинне використовуватися як ідентифікатор повідомлення або вузла. Таким чином, ідентифікатори повідомлень і адреси вузлів можуть знаходитися в будь-якому полі повідомлення (у полі арбітражу або в полі даних, або бути присутнім і там, і там). Також протокол не забороняє використовувати поле арбітражу для передачі даних.

Заповнення поля арбітражу і поля даних, а також розподіл адрес вузлів, ідентифікаторів повідомлень і пріоритетів в мережі є предметом розглядів так званих протоколів високого рівня. Назва HLP відбиває той факт, що протокол CAN описує тільки два нижні рівні еталонної мережевої моделі ISO/OSI, а інші рівні описуються протоколами HLP.

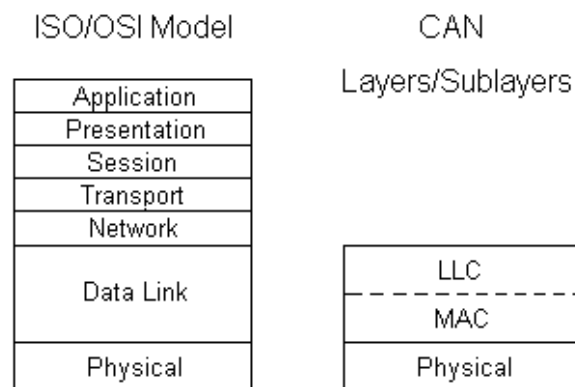


Рисунок 1.3 – Логічна структура протоколу CAN

Існує безліч таких високорівневих протоколів. Найбільш поширені з них це: DeviceNet, CAL/CANopen, SDS, CanKingdom

1.1.7 Передача даних. CAN 2.0 B

Згідно специфікації CAN 2.0 B кадрів даних діляться на такі: стандартний кадр даних, розширений кадр даних, кадри видаленого запиту даних (стандартний і розширений). Стандартний кадр даних приведений у табл. 1.3.

Таблиця 1.3 – Стандартний кадр даних.

Поле	Довжина, біт	Значення
Початок кадру (SOF)	1	Завжди логічний 0
Ідентифікатор	11	ID10...ID0
Запит на передачу (RTR)	1	Завжди логічний 0
Біт розширеного ідентифікатора (IDE)	1	Визначає довжину ідентифікатора (логічний 0 – стандартний кадр)
Зарезервований біт (r0)	1	Завжди логічний 0
Довжина даних (DLC)	4	DLC3...DLC0
Поле даних	0-8 байт	Данні довжиною вказаною у полі DLC
Контрольна сума (CRC)	15	Контрольна сума усього кадру
Розділювач CRC	1	Завжди логічна 1
Біт вставки (ACK)	1	Передавач передає рецесивний біт (логічна 1), а приймач якщо прийняв повідомлення ставить домінуючий біт (логічний 0)
Разграничитель підтвердження	1	Завжди логічна 1
Кінець кадру (EOF)	7	Завжди логічна 1
Міжкадровий інтервал	3	Завжди логічна 1

Стандартний кадр даних починається з біта початку кадру (Start Of Frame – SOF), який є "домінуючим" бітом і потрібний для жорсткої синхронізації усіх вузлів. За бітом SOF йде поле арбітражу, що містить 12 бітів: 11-бітовий ідентифікатор і біт видаленого запиту даних (Remote Transmission Request – RTR). За допомогою біта RTR кадр даних (у якому RTR має значення "домінуючий") відрізняється від кадру видаленого запиту (у якому RTR "рецесивний"). За полем арбітражу йде поле управління, що складається з шести бітів. Перший біт цього поля – біт розширеного ідентифікатора (Identifier Extension – IDE), який для стандартного кадру повинен мати значення "домінуючий". Наступний біт – це нульовий зарезервований біт (Reserved Bit Zero – RB0), який згідно з протоколом CAN є "домінуючим". Ті, що залишилися, чотири біта поля управління є кодом довжини даних (Data Length Code – DLC), що означає число байтів даних, що містяться в повідомленні. Після поля управління розташовується поле даних, що містить байти передаваних даних, довжина якого задається бітами DLC (0-8 байтів). Поле CRC, що йде за полем даних, використовується для виявлення помилок передачі. Воно складається з 15-бітової послідовності CRC і що йде за нею "рецесивного" біта-роздільника CRC. Останнє поле – двохбітове поле підтвердження (acknowledgement – ACK). Як біт вставки ACK передавальний вузол посилає "рецесивний" біт. Будь-який вузол, що прийняв кадр без помилок, підтверджує правильний прийом передачею у відповідь в цьому полі "домінуючого" біта (незалежно від того, чи налагоджений цей вузол на обробку цього повідомлення або ні). "Рецесивний" біт роздільника ACK

завершує поле підтвердження, він не може бути перезаписаний "домінантним" бітом.

Формат розширеного кадру даних представлений у табл. 1.4.

Таблиця 1.4 – Розширений кадр даних

Поле	Довжина, біт	Значення
Початок кадру (SOF)	1	Завжди логічний 0
Ідентифікатор А	11	ID10...ID0
Підміна запиту на передачу (SRR)	1	Завжди логічний 0
Біт розширеного ідентифікатора (IDE)	1	Визначає довжину ідентифікатора (логічний 0 – стандартний кадр)
Ідентифікатор В	18	EID17...EID0
Запит на передачу (RTR)	1	Завжди логічний 0
Зарезервований біт (RB1)	1	Завжди логічний 0
Зарезервований біт (RB0)	1	Завжди логічний 0
Довжина даних (DLC)	4	DLC3...DLC0
Поле даних	0-8 байт	Данні довжиною вказаною у полі DLC
Контрольна сума (CRC)	15	Контрольна сума усього кадру
Розділювач CRC	1	Завжди логічна 1
Біт вставки (ACK)	1	Передавач передає рецесивний біт (логічна 1), а приймач якщо прийняв повідомлення ставить домінуючий біт (логічний 0)
Разграничитель підтвердження	1	Завжди логічна 1
Кінець кадру (EOF)	7	Завжди логічна 1
Міжкадровий інтервал	3	Завжди логічна 1

У розширеному кадрі даних за бітом SOF йде поле арбітражу, що складається з 32 бітів. Перші 11 бітів – старші біти 29-бітового ідентифікатора (основний або стандартний ідентифікатор (Base – ID)). Далі слідує біт-замінник видаленого запиту (Substitute Remote Request – SRR), що має значення "рецесивний". За бітом SRR йде біт IDE, який є "рецесивним" і означає розширений кадр. Слід зауважити, що якщо арбітраж залишається нерозв'язним після передачі перших 11 бітів ідентифікатора, і один з вузлів, залучених в арбітраж, передає стандартний кадр (з 11-бітовим ідентифікатором), то стандартний кадр виграє арбітраж через установку "домінуючого" біта IDE. З тієї ж причини біт SRR в розширеному кадрі теж повинен мати значення "рецесивний", щоб вузол, передавальний стандартний кадр видаленого запиту, міг встановити "домінуючий" біт RTR. За бітами SRR і IDE йдуть ті, що залишилися 18 бітів ідентифікатора (розширений ідентифікатор (Extended ID)) і біт видаленого запиту передачі (RTR). Такий розподіл ідентифікатора на дві частини (старшу і молодшу) обумовлений необхідністю розташування біта IDE в одній і тій же позиції і для стандартних, і для розширених кадрів, щоб вузли в

межах однієї мережі могли передавати обидва типи кадрів. За полем арбітражу йде 6-бітове поле управління. Перші два біта зарезервовані і є "домінантними". Що залишилися чотири біта є кодом довжини даних DLC, що означає число байтів даних, які містяться в повідомленні. Частина кадру (поле даних, поле CRC, поле підтвердження, кінець кадру і міжкадровий інтервал), що залишилися, побудовані так само, як в стандартному кадрі даних.

Формат кадру видаленого запиту даних представлена у табл. 1.5.

Таблиця 1.5 – Формат кадру видаленого запиту даних

Поле	Довжина, біт	Значення
Початок кадру (SOF)	1	Завжди логічний 0
Ідентифікатор А	11	ID10...ID0
Підміна запита на передачу (SRR)	1	Завжди логічний 0
Біт розширеного ідентифікатора (IDE)	1	Визначає довжину ідентифікатора (логічний 0 – стандартний кадр)
Ідентифікатор В	18	EID17...EID0
Запит на передачу (RTR)	1	Завжди логічний 0
Зарезервований біт (RB1)	1	Завжди логічний 0
Зарезервований біт (RB0)	1	Завжди логічний 0
Довжина даних (DLC)	4	DLC3...DLC0
Контрольна сума (CRC)	15	Контрольна сума усього кадру
Розділювач CRC	1	Завжди логічна 1
Біт вставки (ACK)	1	Передатчик передає рецесивний біт (логічна 1), а приймач якщо прийняв повідомлення ставить домінуючий біт (логічний 0)
Разграничитель підтвердження	1	Завжди логічна 1
Кінець кадру (EOF)	7	Завжди логічна 1
Міжкадровий інтервал	3	Завжди логічна 1

Для вузла-одержувача є можливість запитати кадр даних у вузла-джерела. Для цього вузол-одержувач посилає кадр видаленого запиту з ідентифікатором, який відповідає ідентифікатору запрошеного кадру даних. У відповідь на цей кадр видаленого запиту відповідне вузол-джерело передасть кадр даних. Існує дві відмінності між кадром видаленого запиту і кадром даних. Перша відмінність кадру запиту полягає в тому, що біт RTR має значення "рецесивний", а друге – у відсутності поля даних. У разі, коли одночасно передаються кадр даних і кадр видаленого запиту з однаковими ідентифікаторами, кадр даних виграє арбітраж через "домінуючий" біт RTR.

Формат кадру помилки представлений у табл. 1.6.

Таблиця 1.6 – Формат кадру помилки

Поле	Довжина, біт	Значення
Початок кадру (SOF)	1	Завжди логічний 0
Ідентифікатор	11	ID10...ID0
Запит на передачу (RTR)	1	Завжди логічний 0
Біт розширеного ідентифікатора (IDE)	1	Визначає довжину ідентифікатора (логічний 0 – стандартний кадр)
Зарезервований біт (RB0)	1	Завжди логічний 0
Довжина даних (DLC)	4	DLC3...DLC0
Поле даних	0-8 байт	Данні довжиною вказаною у полі DLC
Кадр помилки		
Флаг помилки	6	Завжди логічний 0
Флаг у відповідь	≤6	Логічний 0
Розділювач помилки	8	Завжди логічна 1
Міжкадровий інтервал чи кадр перезавантаження		

Кадр помилки формується вузлом, що виявив на шині помилку. Кадр помилки складається з двох полів – поля мітки помилки і поля роздільника помилки, що йде за ним. Існує два типи мітки помилки, що видаються залежно від стану вузла, що виявив помилку і що згенерувало кадр помилки. Якщо помилку на шині виявляє вузол, що знаходиться в стані активної помилки, він перериває передачу поточного повідомлення, виставляючи активну мітку помилки. Ця мітка є послідовністю з шести "домінантних" бітів, яка порушує правило бітового наповнення. Усі інші вузли розпізнають виниклу у результаті помилку і теж починають формувати кадри помилок. (error echo flags). Поле флага помилки включає від шести до дванадцяти послідовних "домінантних" бітів (формованих одним або декількома вузлами). Поле роздільника помилки завершує кадр помилки. Після завершення кадру помилки, шина повертається в нормальний стан, і перерваний вузол намагається повторити передачу зруйнованого повідомлення. Якщо помилку на шині виявляє вузол, що знаходиться в стані пасивної помилки, він передає пасивний флаг помилки і поле роздільника помилки, що йде за ним. Пасивна мітка помилки складається з шести послідовних "рецесивних" бітів, а сам кадр помилки в цілому складається з 14 "рецесивних" бітів. З цього виходить, що, якщо помилка шини виявлена не передавальним вузлом, передача кадру пасивної помилки не вплине на інші вузли мережі. Якщо мітку пасивної помилки формує передавальний вузол, тоді це змусить інші вузли формувати кадри помилок. Після передачі кадру помилки, вузол, що знаходиться в стані пасивної помилки, повинен чекати появи на шині шести послідовних "рецесивних" бітів перед спробою відновлення зв'язку з шиною. Роздільник помилки складається з восьми "рецесивних" бітів і дозволяє вузлам повністю відновити зв'язок з шиною після помилки, що сталася.

Формат кадру перезавантаження приведений у табл. 1.7.

Таблиця 1.7 – Формат кадру перезавантаження

Поле	Довжина, біт	Значення
Початок кадру (SOF)	1	Завжди логічний 0
Ідентифікатор	11	ID10...ID0
Запит на передачу (RTR)	1	Завжди логічний 0
Біт розширеного ідентифікатора (IDE)	1	Визначає довжину ідентифікатора (логічний 0 – стандартний кадр)
Зарезервований біт (RB0)	1	Завжди логічний 0
Довжина даних (DLC)	4	DLC3...DLC0
Контрольна сума (CRC)	15	Контрольна сума усього кадру
Розділювачь CRC	1	Завжди логічна 1
Біт вставки (ACK)	1	Передавач передає рецесивний біт (логічна 1), а приймач якщо прийняв повідомлення ставить домінуючий біт (логічний 0)
Разграничитель подтверждения	1	Завжди логічна 1
Кінець кадру (EOF)	7	Завжди логічна 1
Кадр перезавантаження		
Флаг перезавантаження	6	Завжди логічний 0
Флаг у відповідь	≤6	Логічний 0
Розділювач перезавантаження	8	Завжди логічна 1
Міжкадровий інтервал чи кадр перезавантаження		

Кадр перезавантаження може формуватися тільки під час міжкадрового інтервалу. Так кадр перезавантаження можна відрізнити від кадру помилки (кадр помилки видається під час передачі повідомлення). Кадр перезавантаження складається з двох полів: мітки перезавантаження і роздільника перезавантаження. Мітка перезавантаження складається з шести "домінуючих" бітів, за якими йдуть мітки у відповідь перезавантаження, формовані іншими вузлами (як і для мітки помилки, максимальна сумарна довжина мітки перезавантаження складає 12 "домінуючих" бітів). Роздільник перезавантаження складається з восьми "рецесивних" бітів. Кадр перезавантаження може формуватися вузлом в результаті двох умов:

1) Вузол виявляє "домінуючий" біт під час міжкадрового інтервалу, що є забороненою умовою.

2) Через внутрішні умови вузол не в змозі почати прийом наступного повідомлення. Вузол може формувати максимум два послідовні кадри перезавантаження, щоб затримати передачу наступного повідомлення.

1.1.8 Контроль доступу до середовища передачі

Міжкадровий інтервал відділяє попередній кадр (будь-якого типу) від наступного кадру даних або видаленого запиту. Міжкадровий простір складається з мінімум трьох "рецесивних" бітів, які називаються перервою

(Intermission). Він потрібний, щоб у вузлів був час для внутрішньої обробки даних перед початком наступного кадру. Після перерви лінія шини залишається в рецесивному стані (шина вільна), поки не почнеться передача наступного повідомлення. Поле арбітражу CAN -кадру використовується в CAN для вирішення колізій доступу до шини методом не деструктивного арбітражу. Суть методу не деструктивного арбітражу полягає в наступному. У разі, коли декілька контролерів починають одночасну передачу CAN кадру в мережу, кожен з них порівнює біт, який збирається передати на шину з бітом, який намагається передати на шину конкуруючий контролер (рис. 1.4).

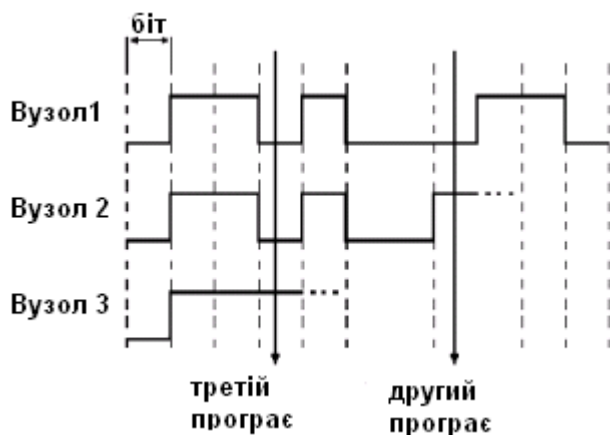


Рисунок 1.4 – Побітовий арбітраж на шині CAN

Якщо значення цих бітів рівні, обидва контролери передають наступний біт. І так відбувається до тих пір, поки значення передаваних бітів не виявляться різними. Тепер контролер, який передавав логічний нуль (пріоритетніший сигнал) продовжуватиме передачу, а інший (інші) контролер перерве свою передачу до того часу, поки шина знову не звільниться.

1.1.9 Методи виявлення помилок

Bit monitoring – кожен вузол під час передачі бітів в мережу порівнює значення передаваного їм біта зі значенням біта, яке з'являється на шині. Якщо ці значення не співпадають, то вузол генерує помилку Bit Error. Під час арбітражу на шині (передача поля арбітражу в шину) цей механізм перевірки помилок відключається.

Bit stuffing – коли вузол передає послідовно в шину 5 біт з однаковим значенням, то він додає шостий біт з протилежним значенням. Приймаючі вузли цей додатковий біт видаляють. Якщо вузол виявляє на шині більше 5 послідовних біт з однаковим значенням, то він генерує помилку Stuff Error.

Frame Check – деякі частини CAN -повідомлення мають однакоє значення в усіх типах повідомлень. Протокол CAN точно визначає які рівні напруги і коли повинні з'являтися на шині. Якщо формат повідомлень порушується, то вузли формують помилку Form Error.

ACKnowledgement Check – кожен вузол отримавши правильне повідомлення по мережі посилає в мережу доміантний (0) біт. Якщо ж цього не відбувається, то передавальний вузол реєструє помилку Acknowledgement Error.

CRC Check – кожне повідомлення CAN містить CRC суму, і кожен приймаючий вузол підраховує значення CRC для кожного отриманого повідомлення. Якщо підраховане значення CRC суми, не співпадає зі значенням CRC в тілі повідомлення, приймаючий вузол формує помилку CRC Error.

Error confinement – помилка обмеження. Кожен вузол мережі CAN, під час роботи намагається виявити одну з п'яти можливих помилок. Якщо помилка виявлена, вузол передає в мережу Error Frame, руйнуючи тим самим увесь поточний трафік мережі (передачу і прийом поточного повідомлення). Усі інші вузли виявляють Error Frame і приймають відповідні дії (скидають прийняте повідомлення). Крім того, кожен вузол веде два лічильники помилок: Transmit Error Counter (лічильник помилок передачі) і Receive Error Counter (лічильник помилок прийому). Ці лічильники збільшуються або зменшуються у відповідність з декількома правилами. Самі правила управління лічильниками помилок досить складні, але зводяться до простого принципу, помилка передачі призводить до збільшення Transmit Error лічильника на 8, помилка прийому збільшує лічильник Receive Error на 1, будь-яка коректна передача/прийом повідомлення зменшають відповідний лічильник на 1. Ці правила призводять до того, що лічильник помилок передачі передавального вузла збільшується швидше, ніж лічильник помилок прийому приймаючих вузлів. Це правило відповідає припущенню про велику вірогідність того, що джерелом помилок є передавальний вузол. Коли, значення хоч би одного з двох лічильників помилок перевищує межу 127, вузол переходить в стан Error Passive. Коли значення хоч би одного з двох лічильників перевищує межу 255, вузол переходить в стан Bus Off. Вузол знаходиться в стані Error Active у разі виявлення помилки на шині передає в мережу Active Error Flags. Active Error Flags складається з 6 доміантних біт, тому усі вузли його реєструють. Вузол в стані Passive Error передає в мережу Passive Error Flags при виявленні помилки в мережі. Passive Error Flags складається з 6 рецесивних біт, тому інші вузли мережі його не помічають, і Passive Error Flags лише призводить до збільшення Error лічильника вузла. Вузол в стані Bus Off нічого не передає в мережу.

1.2 Завдання

1.2.1. Відповідно до номера варіанту встановити на передавачі інтерфейс CAN і передаваний інформаційний байт в десятковій формі (номер за списком в журналі).

1.2.2. Встановити на приймачі інтерфейс CAN і переконатися в правильному прийомі інформаційного байта (цифри на приймачі і передавачі мають бути однакові).

1.2.3. Підключити виводи CANH до входу А осцилографа і CANL до землі осцилографа і зняти осцилограму сигналу.

1.2.4. Проаналізувати результати.

1.2.5. Зробити висновки.

1.3 Вимоги до звіту

1.3.1 Назва учбової дисципліни, лабораторної роботи.

1.3.2 П.І.Б. студента, група.

1.3.3 Мета лабораторної роботи.

1.3.4 Початкові дані згідно з варіантом завдання.

1.3.5 Осцилограма інтерфейсних сигналів

1.3.6 Аналіз осцилограми

1.3.7. Висновки.

1.4 Контрольні питання

1) В яких одиницях вимірюється швидкість передачі даних для інтерфейсу CAN?

2) Яка максимальна швидкість обміну інформацією для інтерфейсу CAN?

3) Яка максимальна ємність інтерфейсних провідників?

4) Як формується посилка даних цього інтерфейсу?

5) Як відбувається синхронізація передачі даних?

6) Де застосовується інтерфейс CAN?

7) Які недоліки інтерфейсу CAN?

1.5 Приклад виконання дослідницької частини лабораторної роботи

Відповідно до варіанту на передавальному напівкомплекті, натискаючи на верхню ліву кнопку встановлюємо інтерфейс CAN і за допомогою кнопок, що знаходяться праворуч встановлюємо передаване число в десятковій формі. Внизу ліворуч знаходиться кнопка скидання (рис. 1.5).

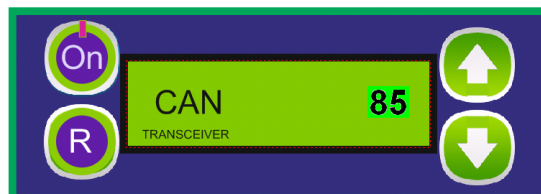


Рисунок 1.5 – Передавальний напівкомплект

Потім на приймаючому напівкомплекті, який зображений на рис. 1.6, за допомогою верхньої лівої кнопки вибираємо режим CAN. У режимі приймача на екрані відображуватиметься число, що приймається, в десятковій формі.

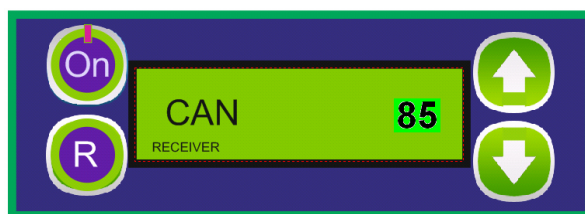


Рисунок 1.6 – Приймаючий напівкомплект

Потім підключаємо до ліній CANH і CANL осцилограф за допомогою підготовлених заздалегідь провідників. Прийнята осцилограма зображена на рис. 1.7.

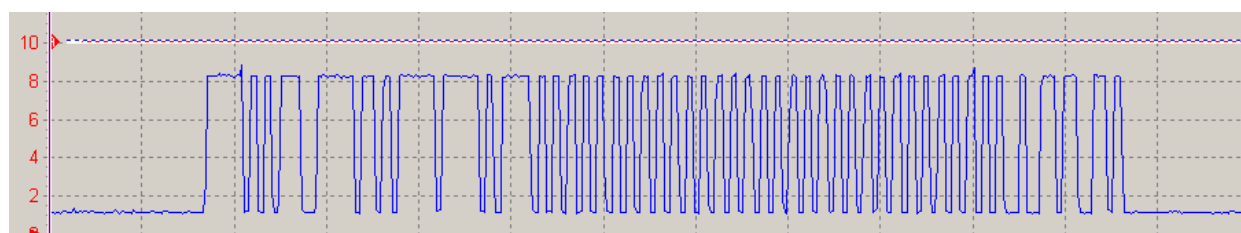


Рисунок 1.7 – Прийнята осцилограма

У технічній документації вказано, що початок кадру формується при подачі на лінію логічного нуля, а завершує кадр подача на лінію 7 біт EOF зі значенням логічної одиниці. На рис. 1.7 відмічений кадр інформаційного повідомлення на шині CAN. Вимірюємо тривалість імпульсу підтвердження прийому повідомлення АСК, а потім малюємо поверх осцилограми прямокутну сітку із меншою стороною рівної тривалості імпульсу підтвердження АСК (рис. 1.8-1.9). Також необхідно звернути увагу на те, що коли передається більше п'яти біт однакового логічного рівня поспіль, тоді контролер після п'яти біт ставить біт протилежного рівня (біт вставки), який потрібен для перезарядки лінії і далі продовжується передача.

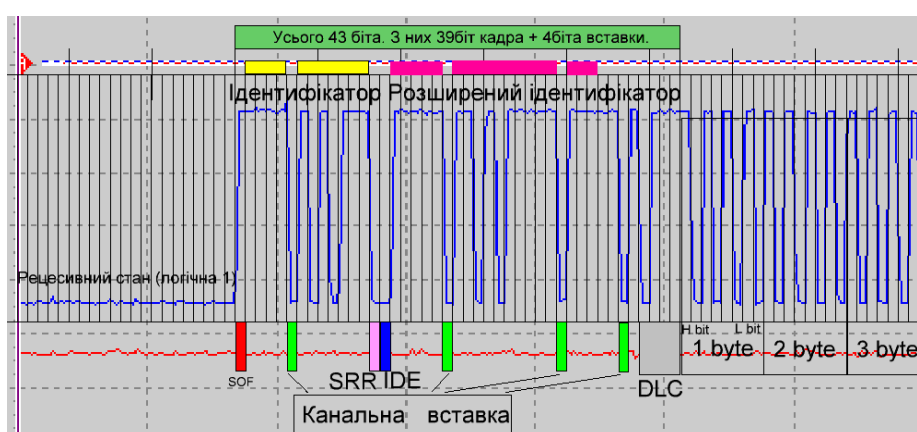


Рисунок 1.8 – Прийнята осцилограма кадру та аналіз осцилограми CAN кадру

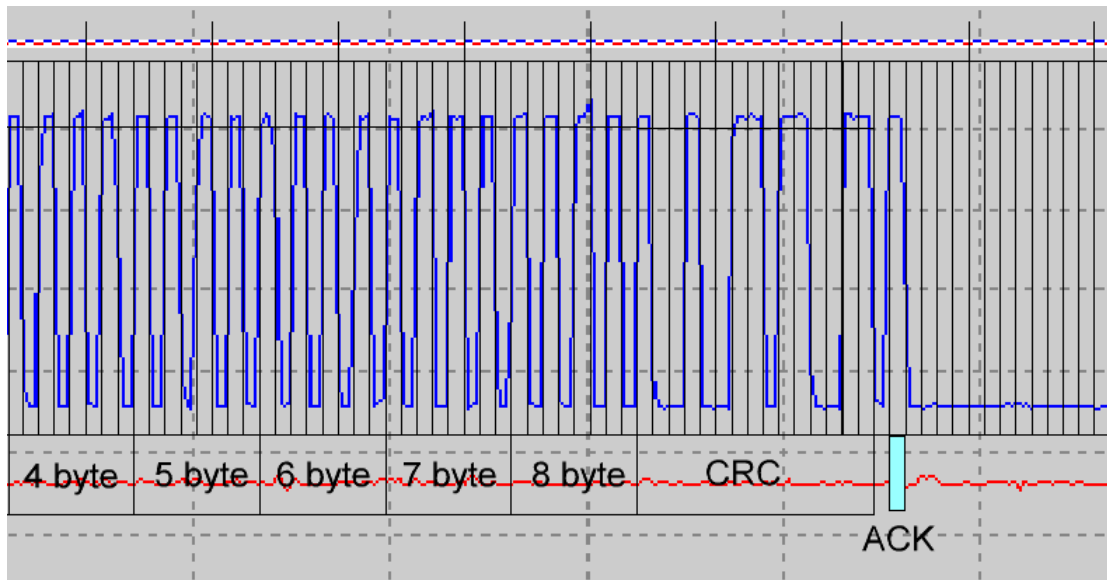


Рисунок 1.9 – Аналіз осцилограми CAN кадру
 Аналіз прийнятого CAN кадру приведений в табл. 1.8.

Таблиця 1.8 – Формат прийнятого кадру

Ім'я поля	Значення
Початок кадру	00b
Ідентифікатор А	000 00101000b
SRR	1b
IDE	1b
Розширений бітовий ідентифікатор В	00 00000101 00000000b
R2-R0	000b
Довжина даних (DLC0-DLC3)	1000b – 8 байт
Поле даних (byte1-byte8)	01010101bin – 85d
CRC	011011001001100bin
CRC роздільник	1bin
ACK	0bin

ЛАБОРАТОРНА РОБОТА № 2

Тема: Використання псевдовипадкових послідовностей в широкосмугових системах передачі інформації

Мета роботи: Дослідження методики формування псевдовипадкових послідовностей (ПВП)

2.1 Теоретичні відомості

2.1.1 Методика формування ПВП

Широкосмугові системи передачі інформації з шумоподібними сигналами розроблялися для умов експлуатації каналів з перешкодами невизначеного характеру, у тому числі і навмисними та для використання у тих випадках, коли характеристики спектру перешкод в каналі зв'язку можуть змінюватися в часі і залежать від структури ліній зв'язку.

Основа дії таких систем – розсередження енергії сигналу в смузі частот, яка у ряді випадків перевищує смугу частот інформаційного (первинного) сигналу повідомлення на 2-3 порядки. На стороні прийому сигнал «збирається» по всій смузі частот каналу з обробкою його методом кореляції.

Послідовність двійкових елементів отримують з використанням рекурентних співвідношень, які визначаються примітивними породжуючими многочленами ступеня m .

Довжина ПВП Хафмана

$$n = 2m - 1, \quad (2.1)$$

при цьому перші “ m ” елементів визначають так звану початкову фазу ПВП.

Для цілей первинного ознайомлення з методом розширеного спектру досить обмежитися значенням $m=3$, для якого довжина ПВП $n=2\cdot 3-1=7$, породжуючий многочлени $P(x)_1=x^3+x^2+1$ або $P(x)_2=x^3+x+1$ й відповідні рекурентні співвідношення:

$$\alpha_i = \alpha_{i-3} \oplus \alpha_{i-2}, \quad (2.2)$$

$$\alpha_i = \alpha_{i-3} \oplus \alpha_{i-1}. \quad (2.3)$$

Відмітимо, що число доданків в рекурентних співвідношеннях завжди буде парним. ПВП на основі сум (2.2, 2.3) можуть бути отримані шляхом апаратної реалізації з використанням регістра зсуву і суматора по модулю 2 (рис. 2.1).

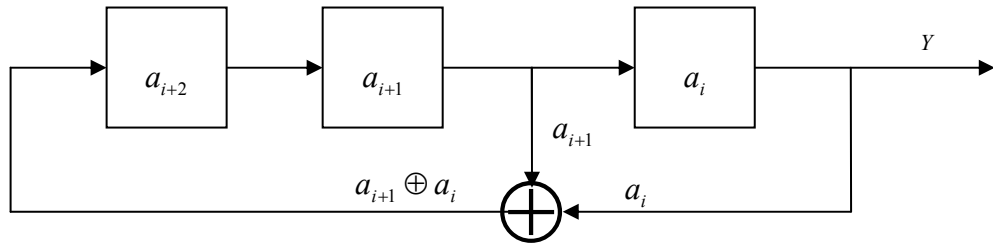


Рисунок 2.1 – Структурна схема апаратної реалізації з використанням регістра зсуву і суматора по модулю 2

У загальному випадку регістр зсуву складається з ячілок (елементів, що запам'ятовують, або тригерів). У кожен момент часу, який визначається тактовою частотою, вміст всіх ячілок зрушується на одну ячілку (розряд) праворуч, причому вміст ячілок, відповідних членам многочлена $P(x)$, підсумовується за модулем 2 і поступає в крайню ліву ячілку.

Якщо в описаному прикладі регістр у момент часу i містить числа a_{i+2}, a_{i+1}, a_i , то у момент часу $i+1$ в ньому знаходяться числа $a_{i+3} = a_{i+1} \oplus a_i, a_{i+2}, a_{i+1}$, (5). Іншими словами, регістр зсуву зі зворотним зв'язком генерує нескінченну послідовність $a_0, a_1, a_2, \dots, a_i$, що задовольняє рекурсивному співвідношенню $a_{i+3} = a_{i+1} \oplus a_i, i = 0, 1, \dots$. Для запуску регістра зсуву необхідного задати початкові значення a_0, a_1, \dots, a_{m-1} . У табл. 2.1-2.3 приведені значення елементів пам'яті регістра зсуву на всіх сімох тактах зсуву при початковій комбінації зсуву 010, 101, 001 відповідно.

Таблиця 2.1 – Стани елементів пам'яті регістра зсуву при початковій комбінації зсуву 010

Порядковий № стану	Стан регістрів			Y
	a_{i+2}	a_{i+1}	a_i	
1	0	1	0	0
2	1	0	1	1
3	1	1	0	1
4	1	1	1	1
5	0	1	1	0
6	0	0	1	0
7	1	0	0	1

Таблиця 2.2 - Стани елементів пам'яті регістра зсуву при початковій комбінації зсуву 101

Порядковий № стану	Стан регістрів			Y
	a_{i+2}	a_{i+1}	a_i	
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0
4	0	1	1	0
5	0	0	1	1
6	1	0	0	0
7	0	1	0	1

Таблиця 2.3 – Стани елементів пам'яті регістра зсуву при початковій комбінації зсуву 001

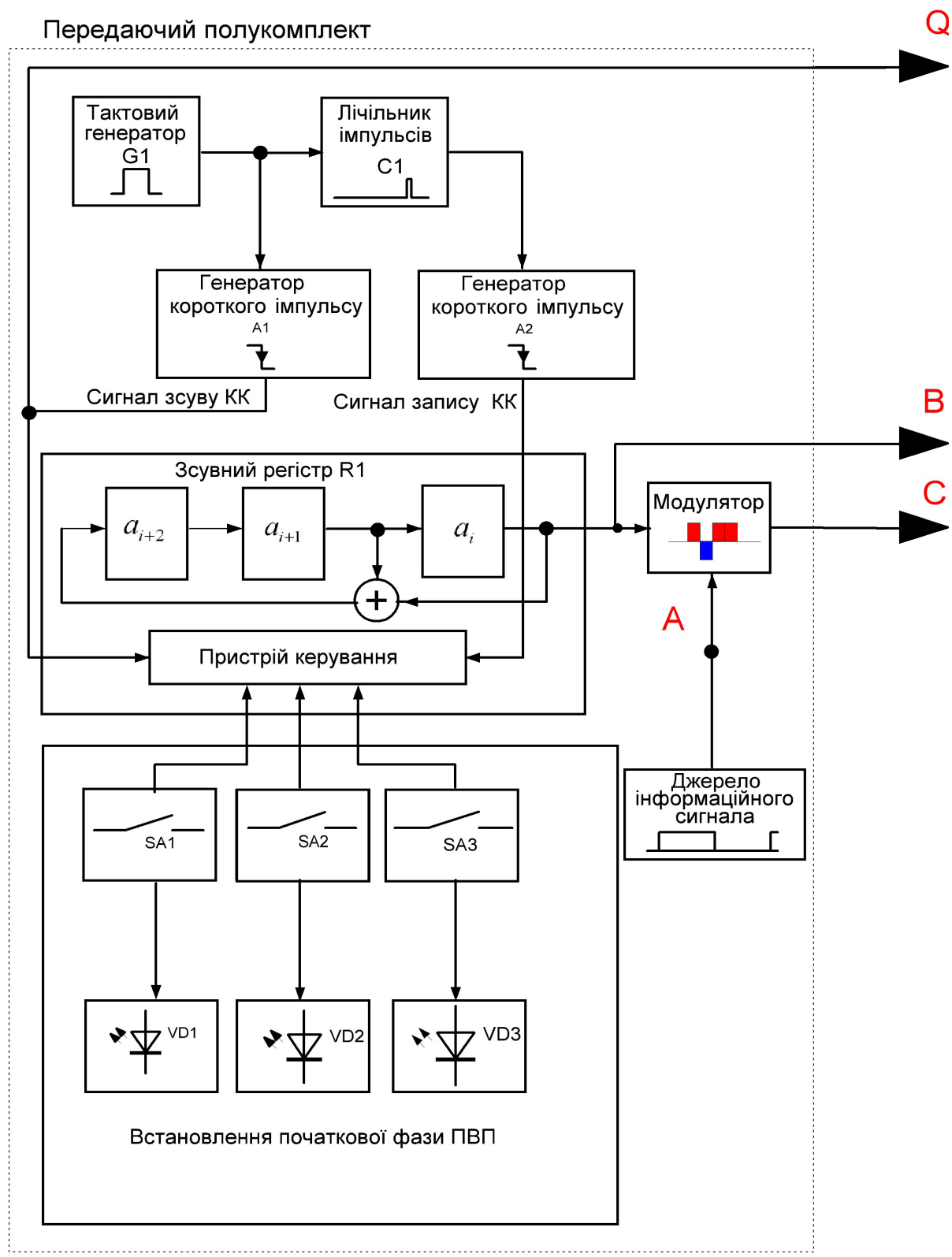
Порядковий № стану	Стан регістрів			Y
	a_{i+2}	a_{i+1}	a_i	
1	0	0	1	1
2	1	0	0	0
3	0	1	0	1
4	1	0	1	1
5	1	1	0	1
6	1	1	1	0
7	0	1	1	0

2.2 Опис лабораторного стенду

На рис. 2.2 приведена структурна схема лабораторного стенду зібраного на дискретних елементах для дослідження процесів, що відбуваються при передачі інформації за допомогою ШПС.

Стенд для лабораторних робіт складається з двох напівкомплектів – передавального і приймаючого. Передавальний напівкомплект формує пачки імпульсів, що повторюються. Генератор G1 формує прямокутні імпульси з частотою дотримання 90 кГц, які поступають на одновібратор A1 і лічильник імпульсів С1. Одновібратор A1 по зрізу імпульсів від генератора G1 формує імпульси, що зрушують інформацію в регістрі R1 циклічно праворуч. Лічильник імпульсів С1 формує цикл заданої довжини, у кінці якого формується короткий імпульс скидання в нуль. По зрізу цього імпульсу одновібратор A2 формує короткий імпульс, по якому в ячійку регістра зрушення записується початкова фаза ПВП з модуля введення на кнопках SA1, – SA3. Сигнал з виходу регістра зрушення R1 поступає на модулятор. Процес модуляції інформаційного сигналу псевдовипадковою послідовністю відбувається так: якщо передається логічний 0, то псевдовипадкова послідовність передається інвертована, а якщо 1-неінвертована.

Далі сигнал з виходу модулятора поступає в лінію зв'язку. З лінії зв'язку модульований сигнал поступає на демодулятор(корелятор). Як синхронізуючий сигнал приймача використовується початкова ПСП передавача. На приймальній стороні цей сигнал подається на зсувний регістр R2, з якого можна знімати ПВП з п'ятьма початковими фазами, включаючи початкову на стороні передачі. Цей сигнал є опорним для двонапівперіодного імпульсного фазового детектора, що виконує функцію корелятора. На його виході встановлюється інтегратор, роль якого виконує ФНЧ у вигляді RC ланцюга.



Приймаючий полукомплект

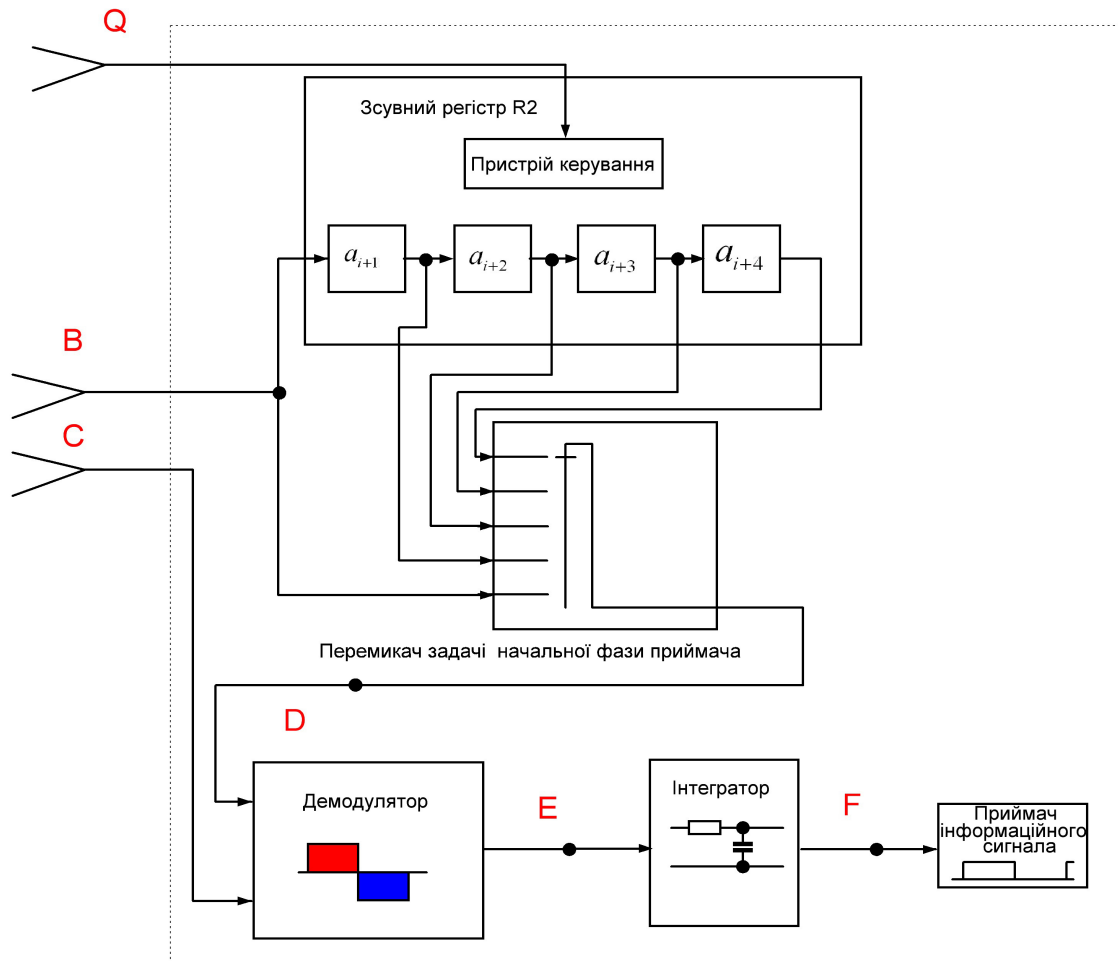


Рисунок 2.2 – Структурна схема апаратної реалізації системи передачі та прийому ШПС з використанням регістра зсуву і суматора за модулем 2

2.3 Завдання

Відповідно до номера варіанту побудувати ПСП для заданого неприведеного многочлена і вектору початкової фази. Потім встановити за допомогою перемикачів на стенді задану початкову фазу і зняти осцилограми в контрольних точках А-Е як при збігу так і при неспівпаданні ПСП в прямому і синхронізуючому каналах. Проаналізувати результати. Зробити висновки.

Варіант №	Вектор початкової фази	Многочлен $P(x)$, що породжує
1,8,15	111	$x^3 + x^2 + 1$
2,9,16	001	$x^3 + x^2 + 1$
3,10,17	010	$x^3 + x^2 + 1$
4,11,18	011	$x^3 + x^2 + 1$
5,12,19	100	$x^3 + x^2 + 1$
6,13,20	101	$x^3 + x^2 + 1$
7,14,21	110	$x^3 + x^2 + 1$

2.4 Вимоги до звіту

Назва учбової дисципліни, лабораторної роботи.

Ф.І. П. студента, шифр групи.

Мета лабораторної роботи.

Початкові дані згідно з варіантом завдання.

Рекурентне співвідношення для визначення членів псевдовипадкової послідовності.

Таблиця згенерованої ПСП в межах одного періоду.

Осцилограми в контрольних точках А-Е.

Висновки.

2.5 Контрольні питання

1. Поясніть в чому полягають відмінності між істинно випадковими і псевдовипадковими послідовностями чисел.

2. Сформулюйте властивості неприведених многочленів.

3. Приведіть рекурентне співвідношення для визначення значення члена псевдовипадкової двійкової послідовності, яка може бути породжена неприведеним многочленом.

4. Дайте визначення кореляційній функції для функції однієї змінної.

2.6 Приклад виконання роботи

2.6.1 Побудова ПВП

Для неприведеного многочлена x^3+x^2+1 і вектору початкової фази 101 сформуємо ПСП. Дані теоретичного розрахунку приведені в таблиці.

Порядковий № стану	Стан реєстрів			Y
	a_{i+2}	a_{i+1}	a_i	
1	0	1	0	0
2	1	0	1	1
3	1	1	0	1
4	1	1	1	1
5	0	1	1	0
6	0	0	1	0
7	1	0	0	1

2.6.2 Режим збігу початкових фаз передавача і приймача

На рис. 2.3 в точці діаграми А показана діаграма інформаційного сигналу, що є чергуванням високого (логічна «1») і низького (логічний «0») рівнів скважністю 2.

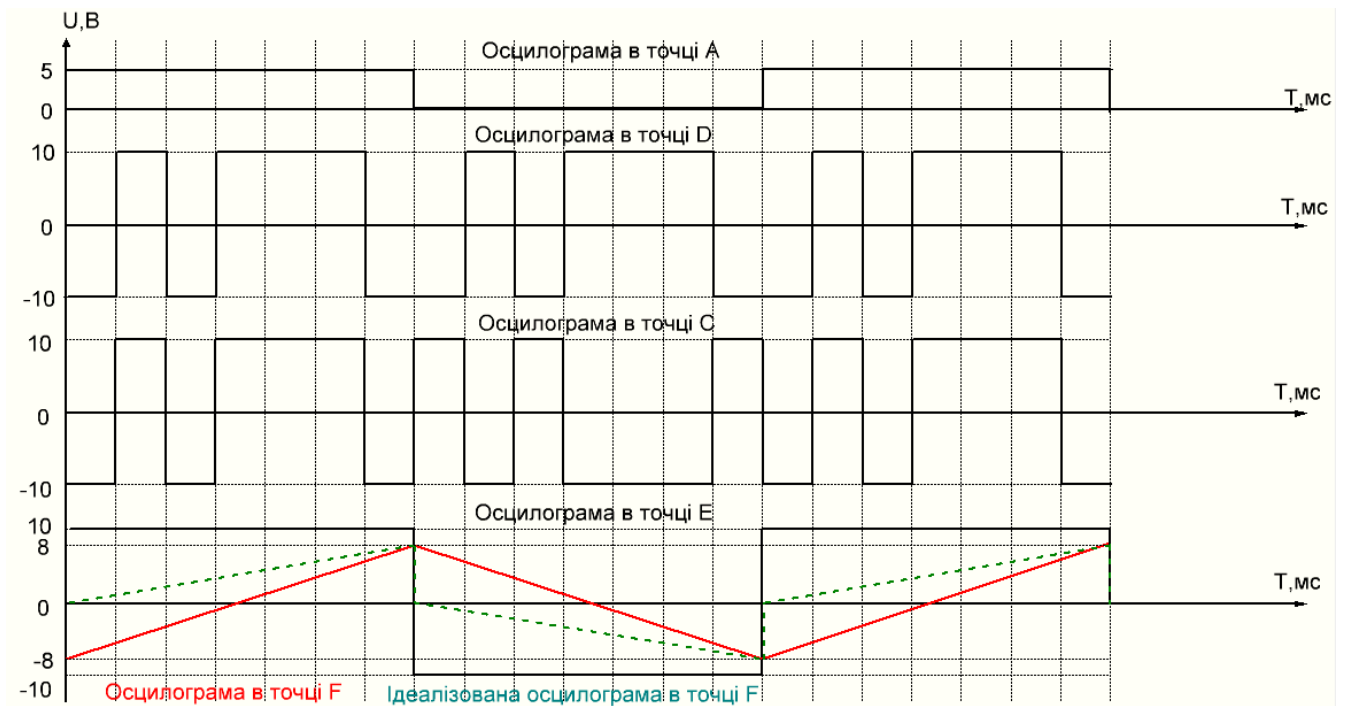


Рисунок 2.3 – Тимчасові діаграми пристрою при початковій фазі передавача 010 та приймача 010

На інтервалі інформаційного елементу «0» або «1» розташовується 7 тактів псевдовипадкової послідовності з початковою фазою 010. В цілому на інтервалі одиничного елементу ПВП має вигляд 0101110. Ця діаграма приведена на рис. 2.3 в точці D. Відповідно до цієї ПВП модулятор формує двополярну імпульсну послідовність, де «0» відповідає амплітуді -10В, а «1» – 10В. В мить, коли інформаційний елемент змінюється елементом «0», імпульсна послідовність ПВП інвертується. Опорна ПВП приймача відображається імпульсним сигналом на рис. 2.3 в точці С, з якого виходить, що на інтервалі інформаційного елементу «1» імпульсних послідовностей співпадають по полярності, а на інтервалі інформаційного елементу «0» мають різні полярності.

Відповідно до цього двонапівперіодний імпульсний фазовий детектор видає двохполярний сигнал виділеного повідомлення. Діаграма цього сигналу приведена на рис. 2.3 діаграма в точці Е. У режимі безперервного інформаційного сигналу ...10101... на виході корелятора він є трикутною формою з максимумами в моменти кінця кожного інформаційного елементу. Діаграма цього сигналу приведена на рис. 2.3 діаграма в точці F. У ці моменти пороговий елемент приймача і виявляє, який інформаційний елемент прийнятий. У реальних СПІ після спрацьовування приймача інтегратор скидається в 0. Діаграма цього сигналу приведена на рис. 3.3 діаграма в точці F пунктирною лінією.

2.6.3 Режим неспівпадання початкових фаз передавача і приймача

На рис. 2.4 в точці діаграми А показана діаграма інформаційного сигналу, що є чергуванням високого (логічна «1») і низького (логічний «0») рівнів скважністю 2. На інтервалі інформаційного елементу логічний «0» або «1» розташовується 7 тактів псевдовипадкової послідовності з початковою фазою 001. В цілому на інтервалі одиничного елементу ПВП має вигляд 0010111. Ця діаграма приведена на рис. 2.4 в точці D. Відповідно до цієї ПВП модулятор формує двохполярну імпульсну послідовність, де логічний «0» відповідає амплітуді -10В , а «1»- 10В .

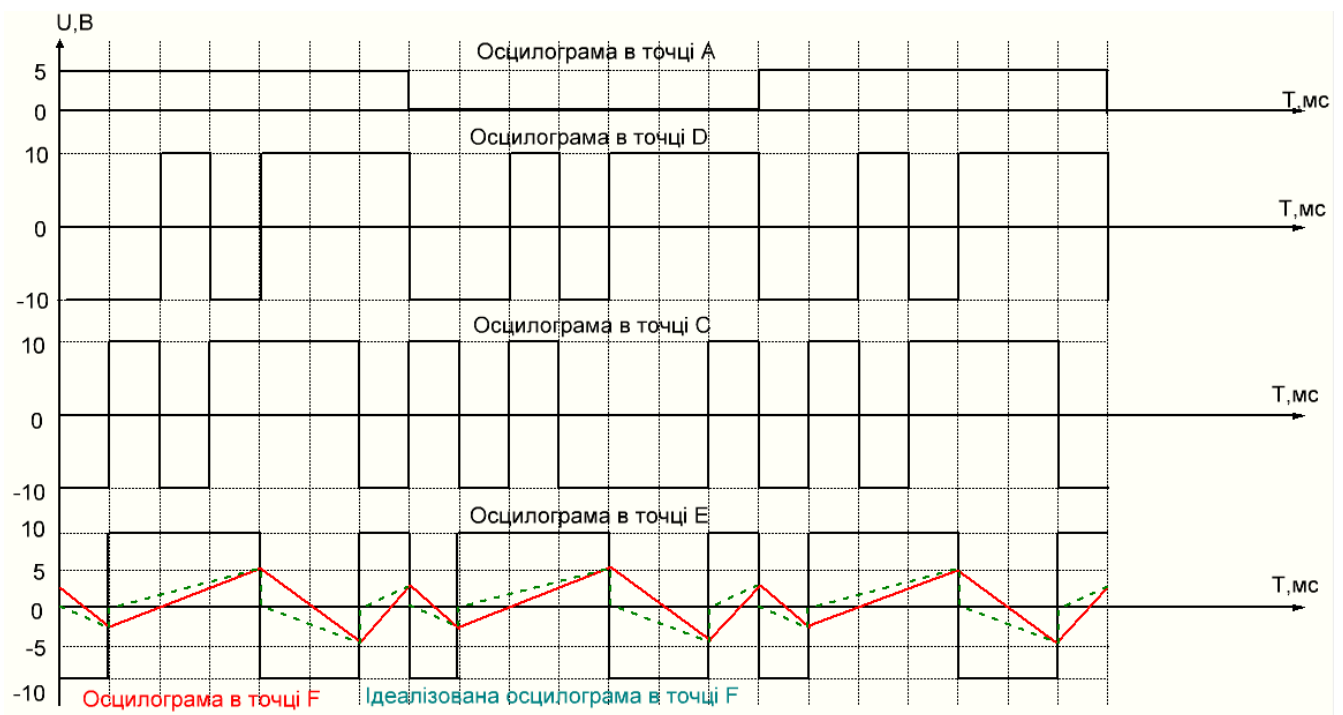


Рисунок 2.4 – Тимчасові діаграми пристрою при початковій фазі передатчика 010 та приймача 001

В мить, коли інформаційний елемент змінюється елементом «0», імпульсна послідовність ПВП інвертується. Опорна ПВП приймача відображається імпульсним сигналом на рис. 2.4 в точці С, з якого виходить, що на інтервалі інформаційного елементу «1» одна частина імпульсних послідовностей співпадає по полярності, а інша частина не співпадає і те ж саме спостерігається на інтервалі інформаційного елементу «0». Відповідно до цього двонапівперіодний імпульсний фазовий детектор видає двохполярний сигнал у вигляді чергування різнополярних імпульсів виділеного повідомлення. Діаграма цього сигналу приведена на рис. 2.4 діаграма в точці Е.

У режимі безперервного інформаційного сигналу ...1010... на виході корелятора він є багатокутною формою з мінімумами в моменти кінця кожного інформаційного елементу. Діаграма цього сигналу приведена на рис. 2.4 в точці F.

У ці моменти пороговий елемент приймача не виявляє ніякого інформаційного елементу. У реальних СПП після спрацьовування приймача інтегратор скидається в 0. Діаграма цього сигналу приведена на рис. 2.4 діаграма в точці F пунктирною лінією.

2.6.4 Аналіз теоретичних і експериментальних досліджень

На початку першого такту сигнал на виході корелятора по амплітуді має значення $\frac{10}{7}$ В. По закінченню першого такту сигнал на виході корелятора приймає значення $-\frac{10}{7}$ В, а у кінці четвертого такту сигнал на виході корелятора приймає значення $-\frac{10}{7} + 3 * \frac{10}{7} = \frac{20}{7}$ В, у кінці шостого такту $-\frac{20}{7} - 2 * \frac{10}{7} = 0$ В і по закінченню сьомого такту рівень сигналу на виході корелятора прийме значення $\frac{10}{7}$ В. Головна ідея перевірена і полягає в тому, що у момент переходу через нуль інформаційного сигналу при неспівпаданні початкових фаз приймача і передавача сигнал на виході корелятора менший за значенням, чим при співпадінні фаз, що при встановленні порогового елемента дає нам змогу приймати сигнал при співпадінні фаз. Сигнал при неспівпадінні початкових фаз у нашому випадку не перевищує значення $\pm \frac{10}{7}$ В, а при співпадінні початкових фаз рівень сигналу у момент переходу через нуль інформаційного сигналу перевищує $\pm 2,5$ В, що забезпечує стабільний прийом.

Список використаної літератури

1. Шумоподібні сигнали в системах передачі інформації. Під ред. В.Б. Пестрякова. – М., “Сов. радіо”, 1973, – 424 с.
2. Ю.С. Льозін. Вступ у теорію радіотехнічних систем. – М.: Радіо та зв'язок, 1985, – 384с.
3. Л.Е. Варакін. Системи зв'язку з шумоподібними сигналами. - М.: Радіо та зв'язок , 1985, – 384с.
4. Котельніков В.А. Теорія потенціальної завадостійкості. – М.: Госенергоіздат, 1956. – 152 с.
5. Д'яченко М.Д. Цифровий захист Учбовий посібник / М.Д. Д'яченко, С.К. Поднебенна. – Маріупіль: ГВУЗ «ПГТУ», 2014. – 304 с.: ил.

Кожевников Антон Вячеславович
Зибалов Дмитро Сергійович
Шевченко Владислав Іванович
Надточий Володимир Валентинович

ІНТЕРФЕЙСИ ТА ТЕХНОЛОГІЇ РАДІОЗВ'ЯЗКУ ІЗ ЗАХИСТОМ ІНФОРМАЦІЇ

ОСНОВИ ЗБОРУ, ОБРОБКИ ТА ПЕРЕДАЧІ ІНФОРМАЦІЇ

Методичні рекомендації до виконання лабораторних робіт

студентами денної форми навчання
з освітньо-професійної програми підготовки бакалаврів за спеціальностями
151 Автоматизація та комп'ютерно-інтегровані технології,
123 Комп'ютерна інженерія

Видано в редакції авторів

Підписано до друку 18.02.2019. Формат 30x42/4.
Папір офсетний. Ризографія. Ум. друк. арк. 1,9.
Обл.-вид. арк. 1,9. Тираж 30 пр. Зам. №

Національний технічний університет
«Дніпровська політехніка»
49005, м. Дніпро, просп. Д. Яворницького, 19.