

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ СХІДНОУКРАЇНСЬКИЙ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ ФАКУЛЬТЕТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ КАФЕДРА
ПРОГРАМУВАННЯ ТА МАТЕМАТИКИ

До захисту допускається
В.о. завідувач кафедри
_____ Лифар В.О.
« ____ » _____ 2020 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломної роботи

бакалавр

(освітньо-кваліфікаційний рівень)

НА ТЕМУ:

Багатоканальна комп'ютерна система реєстрації
рівня аналогових сигналів

Керівник роботи:

(підпис)

Марченко Д. М.

(ініціали, прізвище)

Студент:

(підпис)

Зарубін К.В.

(ініціали, прізвище)

Група:

СКС-16

Сєверодонецьк 2020

ЛИСТ ПОГОДЖЕННЯ І ОЦІНЮВАННЯ
дипломної роботи студента гр. СКС-16 Зарубін К.В.

Науковий керівник

Професор, д.т.н.

Марченко Д. М.

Оцінка наукового керівника:

Рецензент:

ПІБ, місто роботи, посада

Оцінка рецензента:

Кінцева оцінка за результатами захисту:

Голова ЕК

Лифар В.О.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ
ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Програмування та математики
Освітньо-кваліфікаційний рівень бакалавр
Напрямок підготовки 6.050102 «Комп'ютерна інженерія»
(шифр і назва)
Спеціальність _____
(шифр і назва)

ЗАТВЕРДЖУЮ:

Завідувач кафедри ПМ
_____ В.О. Лифар
« _____ » _____ 20 ____ р.

**З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ**

Зарубін Костянтин Володимирович
(прізвище, ім'я, по батькові)

1. Тема роботи: Багатоканальна комп'ютерна система реєстрації
рівня аналогових сигналів
керівник проекту (роботи) Марченко Д.М., доктор технічних наук, професор
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від " ____ " ____ 202 р. № _____
2. Строк подання студентом роботи 07 червня 2020
3. Вихідні дані до роботи Аналіз варіантів побудови комп'ютерної системи.
Розробка апаратно-програмного забезпечення.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Аналітичний огляд, розробка апаратного забезпечення, розробка програмного забезпечення. Висновки
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

6. Консультанти розділів проекту (роботи)

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|--------|---|----------------|------------------|
| | | Завдання видав | Завдання прийняв |
| | | | |
| | | | |
| | | | |

7. Дата видачі завдання _____ 22 березня 2020 року _____

Керівник _____

(підпис)

Завдання прийняв до виконання _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів дипломного проекту (роботи) | Строк виконання етапів проекту (роботи) | Примітка |
|-------|---|---|----------|
| 1 | Складання плану роботи | 22.03.20 - 24.03.20 | |
| 2 | Аналіз літератури | 24.03.20 – 29.03.20 | |
| 3 | Вивчення і підбирання матеріалу | 29.03.20 – 20.04.20 | |
| 4 | Написання розділів | 20.04.20 – 25.05.20 | |
| 5 | Оформлення пояснювальної записки | 25.05.20 – 28.05.20 | |
| 6 | Оформлення графічного матеріалу | 28.05.20 – 03.06.20 | |
| 7 | Підготовка доповіді і слайдів для презентації | 03.06.20 – 07.06.20 | |

Студент _____

(підпис)

(прізвище та ініціали)

Науковий керівник _____

(підпис)

(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка містить 58 сторінок, 18 рисунків, 5 таблиць

Мета роботи – розробка комп'ютерної системи, яка дозволяє контролювати рівень напруги за сукупністю паралельних каналів, ідентифікувати ті канали, на яких рівень напруги перевищує попередньо завданий та здійснювати зберігання результатів вимірювань із заданою дискретністю у часі. При розробці було проаналізовано можливі шляхи вирішення поставленої задачі, розроблене апаратно-програмне забезпечення, виконане імітаційне моделювання.

Розроблена комп'ютерна система має малі габаритні розміри та може легко інтегруватися в якості вбудованої комп'ютерної системи до інших технічних систем реєстрації, автомати та управління.

Для розробки програмного забезпечення було використано середовище AVR Studio.

Для імітаційного моделювання та перевірки коректності функціонування розробленої комп'ютерної системи було використано САД / САМ пакет Proteus.

КОМП'ЮТЕРНА СИСТЕМА, МІКРОКОНТРОЛЕР, АНАЛОГО-
ЦИФРОВИЙ ПЕРЕТВОРЮВАЧ, MULTIMEDIA MEMORY CARD, AVR,
ВБУДОВАНА СИСТЕМА

ЗМІСТ

| | |
|--|----|
| ВСТУП | 7 |
| 1 АНАЛІЗ МОЖЛИВИХ РІШЕНЬ ТА ВИЗНАЧЕННЯ АПАРАТНО-ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РОЗРОБКИ | 9 |
| 1.1 Аналого цифровий перетворювач | 9 |
| 1.1.1 Аналого цифровий перетворювач прямого перетворення..... | 9 |
| 1.1.2 Послідовно-паралельні аналого цифрові перетворювачі..... | 10 |
| 1.1.3 Аналого-цифрові перетворювачі послідовного наближення або аналого-цифрові перетворювачі з порозрядним урівноваженням | 10 |
| 1.1.4 Аналого-цифровий перетворювач диференціального кодування.... | 11 |
| 1.1.5 Аналого-цифровий перетворювач порівняння з пилкоподібним сигналом..... | 12 |
| 1.1.6 Аналого-цифровий перетворювач з урівноваженням заряду | 13 |
| 1.1.7 Конвейєрні аналого-цифрові перетворювачі | 15 |
| 1.1.8 Сигма-дельта аналого-цифровий перетворювач..... | 15 |
| 1.2 Частота дискретизації..... | 17 |
| 1.3 Мікропроцесор | 18 |
| 1.4 Інтерфейс SPI | 19 |
| 1.5 Запам'ятовуючий пристрій | 23 |
| 2 РОЗРОБКА СТРУКТУРИ АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ..... | 28 |
| 3 РОЗРОБКА ПРИНЦИПОВОЇ СХЕМИ АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ ТА ВИБІР ЕЛЕМЕНТІВ | 30 |
| 4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ РЕЄСТРАЦІЇ АНАЛОГОВИХ СИГНАЛІВ..... | 38 |
| 4.1. Розробка структури алгоритмів програмного забезпечення..... | 38 |
| 4.2 Розробка програмного коду | 43 |
| 5 МОДЕЛЮВАННЯ РОЗРОБЛЕНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ..... | 52 |
| ВИСНОВКИ..... | 56 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... | 58 |

ВСТУП

Комп'ютерна, мікропроцесорна техніка все більше і більше входить у повсякденне життя. Вже дуже складно обійтися без комп'ютерів, мобільних телефонів, "інтелектуальної" побутової техніки та інших цифрових пристроїв. Порівняно з аналоговою схемотехнікою, цифрова має масу переваг. Таких, як відносна простота розробки, високу точність, завадостійкість, надійність і технологічність. В сучасних мікросхемах вбудовано більшість цифрових, мікропроцесорних пристроїв, які дозволяють прискорити процес розробки комп'ютерних систем на порядок. На відміну від аналогової схемотехніки, де елементна база не може бути стандартизована через специфіку роботи елементів, елементна база цифрових пристроїв має стандартизоване харчування, рівні логічного нуля і одиниці. Це дозволяє значно спростити схеми комутації елементів, спростити розрахунки.

Сучасна комп'ютерна техніка вимагає наявності простих, продуктивних, надійних в експлуатації керуючих пристроїв. Такими є мікропроцесори. Мікропроцесори (мікроконтролери) дозволяють перенести більшість складних в реалізації схем в область програмного управління, що набагато простіше і швидше. Програмування не вимагає складної і об'ємної переробки схемотехніки. Модернізацію пристрою, в наслідку, можливо провести при незначній зміні апаратного забезпечення. Всі ці причини призвели до масового буму схемотехнічних рішень на основі мікропроцесорів.

Однак, чим простіше апаратне забезпечення пристрою, тим складніше повинна бути його програмна частина. При розробці мікропроцесорних пристроїв необхідно пам'ятати про це. Занадто складна програма може вимагати занадто великих ресурсів від мікропроцесора. Високопродуктивні ж мікроконтролери мають підвищене споживання, вимагають тепловідводу від

великої площі. Така закономірність розробки отримала назву програмно-апаратний дуалізм мікропроцесорних засобів.

Проектована комп'ютерна система повинена фіксувати рівні напруги по 8-ми каналах, здійснювати індикацію каналів, на якому напруга є вищою за попередньо визначений рівень, а також вести реєстр напруг на кожному етапі контролю.

Крім цього, особлива увага приділяється засобам зберігання інформації на компактних сучасних носіях, такі як ММС-карти.

1 АНАЛІЗ МОЖЛИВИХ РІШЕНЬ ТА ВИЗНАЧЕННЯ АПАРАТНО-ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РОЗРОБКИ

1.1 Аналого цифровий перетворювач

Оскільки мікропроцесор не вміє обробляти аналоговий сигнал самостійно, потрібно перетворити напругу в цифровий код. Для цього застосовують спеціальні пристрої, які називають аналого-цифровими перетворювачами. Зазвичай вони реалізуються у вигляді мікросхем (рисунок 1.1), виробляються багатьма сучасними виробниками електроніки, такими, як Analog Devices, Freescale Semiconductor, STSemiconductor та іншими.

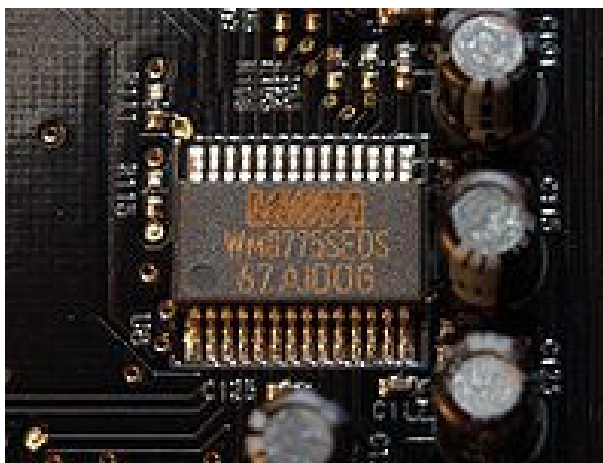


Рисунок 1.1 – Мікросхема аналого-цифрового перетворювача

Аналого-цифрові перетворювачі можуть будуватися за декількома принципами, які розглядаються далі.

1.1.1 Аналого цифровий перетворювач прямого перетворення

Аналого-цифровий перетворювач прямого перетворення або паралельний аналого-цифровий перетворювач містить по одному компаратору на кожен дискретний рівень вхідного сигналу. В будь-який

момент часу тільки компаратори, що відповідають рівням нижче рівня вхідного сигналу, видають на свій вихід сигнал перевищення.

Сигнали з усіх компараторів надходять на логічну схему, яка видає цифровий код, що залежить від того, скільки і яких компараторів показали перевищення. Паралельні аналого-цифрові перетворювачі дуже швидкі, але зазвичай мають роздільну здатність не більше 8 біт (256 компараторів у схемі), так як мають велику і дорогу схему.

Аналого-цифровий перетворювач цього типу мають дуже великий розмір кристала мікросхеми, високу вхідну ємність, і можуть видавати короточасні помилки на виході. Часто використовуються для відео або інших високочастотних сигналів.

1.1.2 Послідовно-паралельні аналого цифрові перетворювачі

Послідовно-паралельні аналого-цифрові перетворювачі зберігаючи високу швидкодію дозволяють значно зменшити кількість компараторів, які необхідні для перетворення аналогового сигналу в цифровий. Як правило, вони містять у своєму складі два-три паралельних аналого-цифрових перетворювача. Другий аналого-цифровий перетворювач служить для зменшення помилки квантування першого шляхом оцифрування цієї помилки. Для збільшення швидкості вихідного оцифрованого потоку даних в послідовно-паралельних аналого-цифрових перетворювачах застосовується конвеєрна робота паралельних аналого-цифрових перетворювачів.

1.1.3 Аналого-цифрові перетворювачі послідовного наближення або аналого-цифрові перетворювачі з порозрядним урівноваженням

Аналого цифрові перетворювачі послідовного наближення або аналого-цифрові перетворювачі з порозрядним урівноваженням містять компаратор,

допоміжний цифро-аналоговий перетворювач та регістр послідовного наближення. Аналого-цифровий перетворювач перетворює аналоговий сигнал в цифровий за N кроків, де N – бітова сила Аналого-цифрового перетворювача. На кожному кроці визначається по одному біту шуканого цифрового значення, починаючи від старшого розряду і закінчуючи молодшим.

Послідовність дій з визначення чергового біта полягає в наступному. На допоміжному цифро-аналоговому перетворювачі виставляється аналогове значення, утворене з бітів, вже визначених на попередніх кроках. Біт, який має бути визначений на поточному кроці, виставляється в 1, більш молодші біти встановлено у 0. Отримане на допоміжному цифро-аналоговому перетворювачі значення порівнюється з вхідним аналоговим значенням. Якщо значення вхідного сигналу більше за значення на допоміжному цифро-аналоговому перетворювачі, то визначений біт отримує значення 1, в іншому випадку 0. Таким чином, визначення підсумкового цифрового значення нагадує двійковий пошук.

Аналого-цифрові перетворювачі цього типу мають одночасно високою швидкістю і хорошу роздільну здатність. Проте за відсутності пристрою вибірки зберігання похибка буде значно більше. Найбільш критичним випадком отримання похибки є коли після оцифрування самого великого розряду сигнал починає змінюватися до завершення усіх процесів порівнянь.

1.1.4 Аналого-цифровий перетворювач диференціального кодування

Аналого-цифровий перетворювач диференціального кодування (англ. delta-encoded ADC) містить реверсивний лічильник, код з якого надходить на допоміжний цифро-аналоговий перетворювач. Вхідний сигнал і сигнал із допоміжного цифро-аналогового перетворювача порівнюються на компараторі. Завдяки негативному зворотному зв'язку компаратора з

лічильником код на лічильнику постійно змінюється так, щоб сигнал із допоміжного цифро-аналогового перетворювача якомога менше відрізнявся від вхідного сигналу. Після деякого часу різниця сигналів стає меншою, ніж молодший розряд, при цьому код лічильника зчитується як вихідний цифровий сигнал аналого-цифрового перетворювача.

Аналого-цифрові перетворювачі цього типу мають дуже великий діапазон вхідного сигналу і високу роздільну здатність, але час перетворення залежить від вхідного сигналу, хоча й обмежений зверху. В найгіршому випадку, час перетворення дорівнює

$$t_{max} = (2q) / f_c,$$

де q – роздільна здатність аналого-цифрового перетворювача;

f_c – частота тактового генератора лічильника.

Аналого-цифровий перетворювач диференціального кодування зазвичай є хорошим вибором для оцифрування сигналів реального світу, так як більшість сигналів у фізичних системах не схильні до стрибкоподібних змін. В деяких аналого-цифрових перетворювачах застосовується комбінований підхід: диференційне кодування і послідовне наближення; це особливо добре працює у випадках, коли відомо, що високочастотні компоненти в сигналі відносно невеликі.

1.1.5 Аналого-цифровий перетворювач порівняння з пилкоподібним сигналом

Аналого-цифрові перетворювачі порівняння з пилкоподібним сигналом (іноді, аналого-цифровий перетворювач цього типу називають інтегруючий аналого-цифровий перетворювач) містять генератор пилкоподібної напруги, компаратор і лічильник часу. Пилкоподібний сигнал лінійно наростає до

деякого рівня, потім швидко спадає до нуля. В момент початку наростання запускається лічильник часу. Коли пилоподібний сигнал досягає рівня вхідного сигналу, компаратор спрацьовує і зупиняє таймер; значення зчитується з лічильника і подається на вихід аналого-цифрового перетворювача. Даний тип аналого-цифрового перетворювача є найбільш простим за структурою і містить мінімальне число елементів. Разом з тим, найпростіші аналого-цифрові перетворювачі цього типу мають досить низьку точність і чутливість до температури і інших зовнішніх факторів. Для збільшення точності, генератор пилоподібного сигналу може бути побудований на основі лічильника та допоміжного цифро-аналогового перетворювача. Однак, така структура не має жодних переваг порівняно з аналого-цифровим перетворювачем послідовного наближення і аналого-цифровим перетворювачем диференціального кодування.

1.1.6 Аналого-цифровий перетворювач з урівноваженням заряду

До класу аналого-цифрових перетворювачів з урівноваженням заряду відносяться:

- аналого-цифровий перетворювач з двохстадійним інтегруванням;
- аналого-цифровий перетворювач з багатостадійним інтегруванням

Такі аналого-цифрові перетворювачі містять генератор стабільного струму, компаратор, інтегратор струму, тактовий генератор і лічильник імпульсів.

Перетворення відбувається в два етапи (двохстадійне інтегрування). На першому етапі значення вхідної напруги перетворюється на струм, пропорційний вхідній напрузі, який подається на інтегратор струму, заряд якого спочатку дорівнює нулю. Цей процес триває протягом часу TN , де T – період тактового генератора, а N – константа, тобто, велике ціле число, яке визначає час накопичення заряду. По закінченні цього часу вхід інтегратора

відключається від входу аналого-цифрового перетворювача і підключається до генератора стабільного струму. Полярність генератора така, що він зменшує заряд, накопичений в інтеграторі. Процес розряду триває до тих пір, поки заряд в інтеграторі не зменшиться до нуля. Час розряду вимірюється шляхом рахування тактових імпульсів від моменту початку розряду до досягнення нульового заряду на інтеграторі. Підрахована кількість тактових імпульсів і буде вихідним кодом аналого-цифрового перетворення.

Можна показати, що кількість імпульсів n , підрахована за час розряду, дорівнює:

$$n = U_{вх} N (R I_0) - 1,$$

де $U_{вх}$ - вхідна напруга аналого-цифрового перетворювача;

N – кількість імпульсів етапу накопичення (визначено вище);

R – опір резистора, який перетворює вхідну напругу в тік (опір резистивного шунту),

I_0 – значення струму від генератора стабільного струму, через який здійснюється лінійне розрядження інтегратору на другому етапі.

Таким чином, потенційно нестабільні системні параметри (насамперед, ємність конденсатора інтегратора) не входять до підсумкового виразу. Це є наслідком двохстадійності процесу: похибки, введені на першому та другому етапах, взаємно віднімаються.

Слід зазначити, що не висуваються жорсткі вимоги навіть до довготривалої стабільності тактового генератора і напруги зміщення компаратору: ці параметри повинні бути стабільні лише короткочасно, тобто протягом кожного перетворення (не більше за час $2TN$).

Фактично, принцип двохстадійного інтегрування дозволяє безпосередньо перетворювати співставлення двох аналогових величин: вхідного і опорного струму, у відношення числових кодів: n і N в термінах, визначених вище, практично без внесення додаткових помилок.

Типова роздільна здатність аналого-цифрового перетворювача цього типу становить від 10 до 18 двійкових розрядів. Додатковою перевагою є можливість побудови перетворювачів, нечутливих до періодичних перешкод, наприклад, перешкоди від мережевого живлення, завдяки точному інтегруванню вхідного сигналу за фіксований часовий інтервал.

Недоліком даного типу аналого-цифрових перетворювачів є низька швидкість перетворення. Аналог-цифровий перетворювач з урівноваженням заряду використовуються в вимірювальних приладах високої точності.

1.1.7 Конвейєрні аналого-цифрові перетворювачі

Конвейєрні аналого-цифрові перетворювачі використовують два або більше кроків-поддиапазонів. На першому кроці виробляється грубе перетворення (з низькою роздільною здатністю). Далі визначається різниця між вхідним сигналом і аналоговим сигналом, відповідним результату грубого перетворення (за допомогою цифро-аналогового перетворювача, на який подається грубий код). На другому кроці знайдена різниця піддається перетворенню, і отриманий код об'єднується з грубим кодом для отримання повного точного цифрового значення. Аналого-цифрові перетворювачі цього типу швидкі, мають високу роздільну здатність і невеликий розмір корпусу.

1.1.8 Сигма-дельта аналого-цифровий перетворювач

Сигма-дельта аналого-цифровий перетворювач (також існують екземпляри дельта-сигма аналого-цифрових перетворювачів) здійснює аналого-цифрове перетворення з частотою дискретизації, що в багато разів перевищує необхідну, і шляхом фільтрації залишає в сигналі тільки потрібну спектральну смугу.

Основні характеристики сигма-дельта аналого-цифрових перетворювачів – це висока роздільна здатність і частота дискретизації, а також малий рівень шумів.

Роздільна здатність аналого-цифрового перетворювача – це мінімальна зміна величини аналогового сигналу, яка може бути перетворена даним аналого-цифровим перетворювачем, пов'язана з його розрядністю. У разі одиничного вимірювання, без урахування шумів, роздільна здатність, безпосередньо, визначається розрядністю аналого-цифрового перетворювача.

Роздільна здатність аналого-цифрового перетворювача характеризує кількість дискретних значень, які перетворювач може видати на виході. У двійкових аналого-цифрових перетворювачах вимірюється в бітах, в трійчастих аналого-цифрових перетворювачах – вимірюється в трітах. Наприклад, двійковий аналого-цифровий перетворювач, здатний видати 256 дискретних значень (0...255), має роздільну здатність 8 бітів, оскільки $2^8 = 256$, трійчастий аналого-цифровий перетворювач, який має роздільну здатність 8 тріт, здатний видати 6561 дискретне значення (від 0 до 6560), оскільки $3^8 = 6561$.

Роздільна здатність за напругою дорівнює різниці напруг, відповідних максимальному і мініимальному вихідним кодам, поділений на кількість вихідних дискретних значень. Надалі наведені приклади.

Приклад 1.

- діапазон вхідних значень – від 0 до 10 вольт;
- роздільна здатність двійкового аналого-цифрового перетворювача 12 бітів: $2^{12} = 4096$ рівнів квантування;
- роздільна здатність двійкового аналого-цифрового перетворювача за напругою: $(10-0) / 4096 = 0,00244$ Вольт = 2,44 мВ;
- роздільна здатність трійчастого аналого-цифрового перетворювача 12 тріт: $3^{12} = 531441$ рівнів квантування;

– роздільна здатність трійчастого аналого-цифрового перетворювача за напругою: $(10-0) / 531441 = 0,0188 \text{ мВ} = 18,8 \text{ мкВ}$;

Приклад 2.

– діапазон вхідних значень: від 10 до 10 Вольт;

– роздільна здатність двійкового аналого-цифрового перетворювача 14 бітів: $2^{14} = 16\,384$ рівнів квантування;

– роздільна здатність двійкового аналого-цифрового перетворювача за напругою: $(10 - (-10)) / 16384 = 20 / 16384 = 0,00122 \text{ Вольт} = 1,22 \text{ мВ}$;

– роздільна здатність трійчастого аналого-цифрового перетворювача 14 тріт: $3^{14} = 4\,782\,969$ рівнів квантування;

– роздільна здатність трійчастого аналого-цифрового перетворювача за напругою: $(10 - (-10)) / 4782969 = 0,00418 \text{ мВ} = 4,18 \text{ мкВ}$

На практиці роздільна здатність аналого-цифрового перетворювача обмежена співставленням сигнал / шум вхідного сигналу. При великій інтенсивності шумів на вході аналого-цифрового перетворювача, розрізнення сусідніх рівнів вхідного сигналу стає неможливим, тобто погіршується роздільна здатність. При цьому реально досяжна роздільна здатність описується ефективною розрядністю (Effective Number Of Bits - ENOB), яка менша, ніж реальна роздільна здатність аналого-цифрового перетворювача. При перетворенні сильно зашумленого сигналу, молодші розряди вихідного коду практично марні, оскільки містять шум. Для досягнення заявленої розрядності, співставлення сигнал / шум вхідного сигналу мало б бути приблизно 6 дБ на кожен біт розрядності.

1.2 Частота дискретизації

Аналоговий сигнал є безперервною функцією часу. В аналого-цифровому перетворювачі він перетворюється на послідовність цифрових значень. Отже, необхідно визначити частоту вибірки цифрових значень з

аналогового сигналу. Частота, з якою виробляються цифрові значення, отримала назву частоти дискретизації аналого-цифрового перетворювача.

Безперервний сигнал з обмеженою спектральною смугою піддається оцифруванню (тобто значення сигналу вимірюються через інтервали часу T – періода дискретизації) та вихідний сигнал може бути точно відновлений із дискретних в часі значень шляхом інтерполяції. Точність відновлення обмежена помилкою квантування. Проте, відповідно до теоремою Котельникова-Шеннона точне відновлення можливо, тільки якщо частота дискретизації вища, ніж подвоєна максимальна частота в спектрі сигналу.

Оскільки реальні аналого-цифрові перетворювачі не можуть здійснювати аналого-цифрове перетворення миттєво, вхідне аналогове значення має зберігатися постійним принаймні від початку до кінця процесу перетворення (цей інтервал часу називають час перетворення). Ця задача вирішується шляхом використання спеціальної схеми на вході аналого-цифрового перетворювача – пристрою вибірки зберігання – $U_{вх}$. $U_{вх}$, як правило, зберігає вхідну напругу в конденсаторі, який з'єднаний зі входом через аналоговий ключ. При замиканні ключа – відбувається вибірка вхідного сигналу (конденсатор заряджається до вхідної напруги), при розмиканні – зберігання. Багато аналого-цифрових перетворювачів, що виконані у вигляді інтегральних мікросхем містять вбудований $U_{вх}$.

1.3 Мікропроцесор

Мікроконтролери фірми Atmel є системами на кристалі, так як окрім самого ядра мікропроцесора в одній мікросхемі можуть перебувати: пам'ять (RAM, Flash, EEPROM), таймери, аналогові компаратори, асинхронні і синхронні прийомо-передавачі, аналого-цифровий перетворювач та інше.

Зазвичай в серію Mega входять контролери з аналого-цифровим перетворювачем розрядністю 10 біт, часом перетворення 50-200мкс, і з

мультиплексованим входом на 6-8 каналів. Оскільки, відповідно до технічного завдання, швидкодія і точність не мають особливих вимог, такі мікроконтролери з вбудованим аналого-цифровим перетворювачем цілком підходять для поставленої задачі проектування.

Існує кілька архітектур побудови мікропроцесорів, однією з найвідоміших є MCS51. Вона поступово застаріває, але за причини її простоти, і того, що вона впроваджена в багатьох системах, "вмирати" вона поки не збирається. Компанія Atmel розробила свою власну архітектуру AVR. Контролери, побудовані за цією архітектурою, мають більшу продуктивність і нижче енергоспоживання порівняно з MCS51. Хоча мають й багато загальних рис.

1.4 Інтерфейс SPI

Інтерфейс Serial Peripheral Interface (SPI) призначений для обміну даними з периферійними пристроями (LCD екрани, карти пам'яті і т. д).

При обміні даними за цим інтерфейсом мікроконтролер ATmega працює в режимі ведучого пристрою (MASTER), до якого можна підключити 1 або кілька ведених блоків (SLAVE). Передача даних починається, коли ведена пристрій активізується ведучим по лінії /SS. Передача даних відбувається наступним чином:

- ведучий пристрій записує байт, підлягаючий передачі, в регістр зсуву SPDR;
- при кожному тактовому імпульсі (тактові імпульси задаються виходом SCK) ведучий пристрій переміщує 1 біт даних на вихід MOSI, а ведений пристрій одночасно передає 1 біт даних на вхід MISO ведучого. Таким чином за 8 тактових імпульсів пристрою Master і Slave обмінюються 1 байтом даних (при чому передається один байт до веденого і один до ведучого);

– після закінчення передачі даних в регістрі SPCR встановлюється прапор запиту на переривання.

Поточна передача даних може бути передчасно завершена подачею в лінію SS логічної одиниці.

Схема передачі даних по SPI проілюстрована на рисунку 1.2.

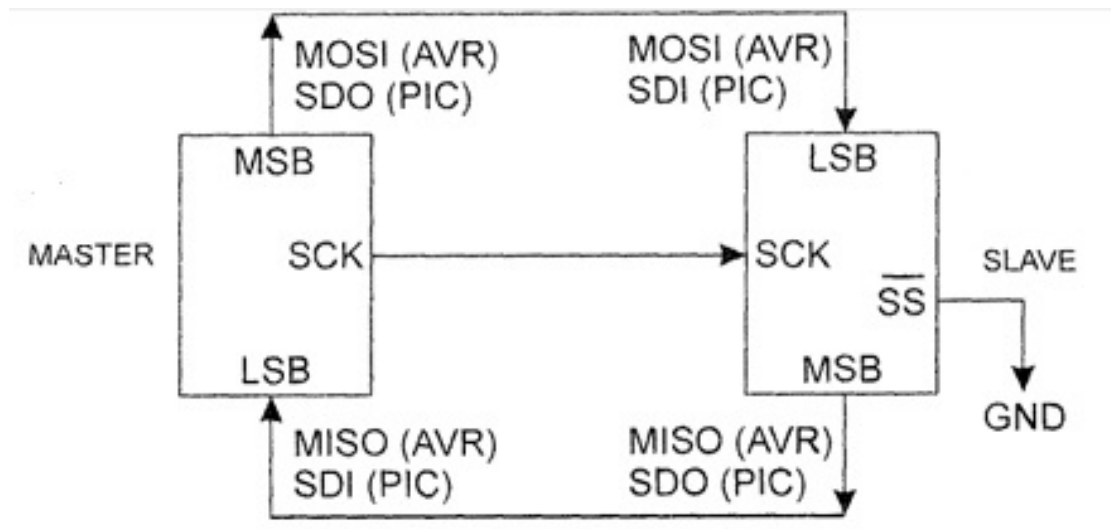


Рисунок 1.2 – Схема передачі даних по інтерфейсу SPI

До мікроконтролера ATМega можна підключити кілька ведених пристроїв. Активним буде той, на вхід SS якого подано логічний 0. Приклад підключення кількох SLAVE блоків до мікроконтролеру проілюстровано на рисунку 1.3.

В ATМega виходи SPI розташовані на порту В: В2 – лінія SS, В3 – MOSI, В4 – MISO, В7 – SCK

Структура регістру управління SPCR представлена в таблиці 1.1.

Таблиця 1.1 – Структура регістру управління SPCR

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|-----|------|------|------|------|------|------|
| SPIE | SPE | DORD | MSTR | CPOL | CPHA | SPR1 | SPR0 |

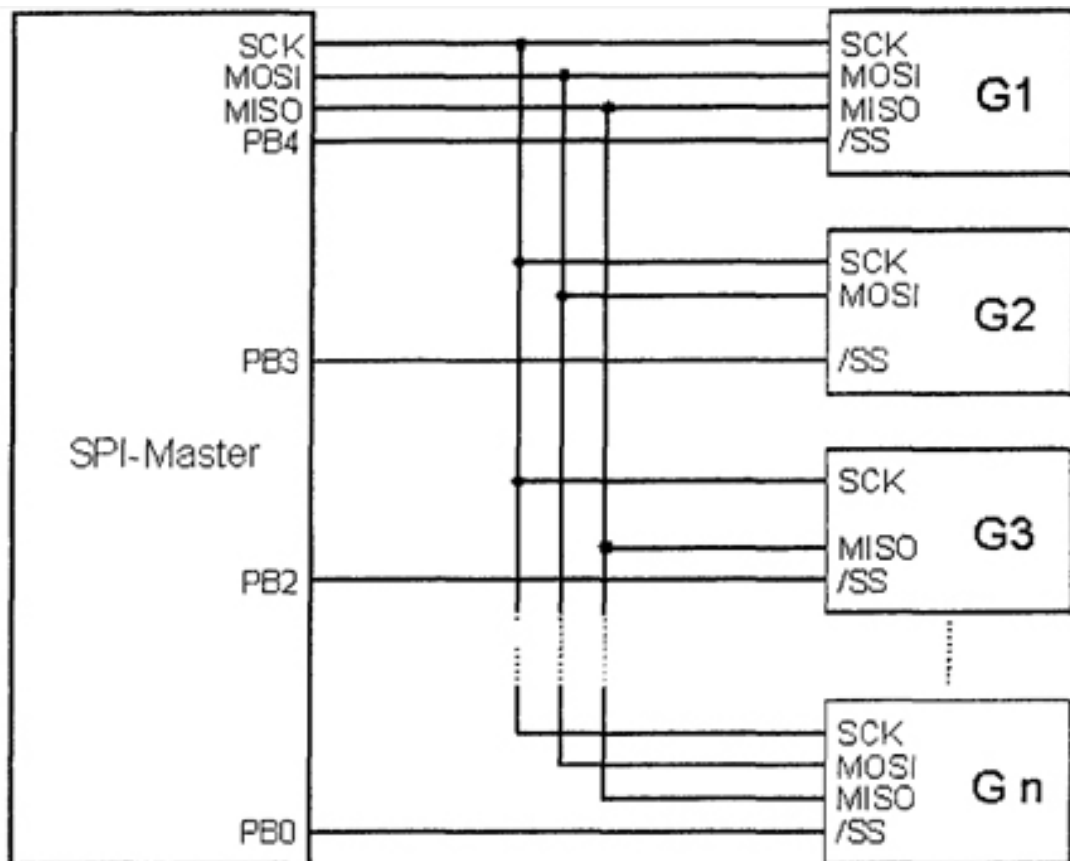


Рисунок 1.3 – Підключення декількох ведених пристроїв до мікроконтролера ATMega по інтерфейсу SPI

В таблиці 1.1 представлені наступні біти:

SPIE = 1: переривання по завершенню передачі даних по SPI дозволено;

SPE = 1: активація інтерфейсу SPI. При вступі сигналу скидання SPE переводиться в стан лог.0 та інтерфейс SPI відключається

DODR = 1: першим передається молодший розряд байту даних. При DODR = 0 першим передається старший розряд байту даних

MSTR = 0: ATMega визначається як ведений пристрій (SLAVE). MSTR = 1 – мікроконтролер з інтерфейсом SPI визначається як ведучий пристрій. Якщо лінія SS (вивід 2 порту B) налаштована в якості входу, то при низькому рівні сигналу на цій лінії MSTR скидається в логічний 0.

$CPOL = 1$ – на лінії SCK в неактивному стані перебуває сигнал високого рівня. За допомогою цього розряду, а також розряду CPHA встановлюється один із чотирьох можливих режимів передачі даних.

SPR0, SPR1 призначені для зміни частоти тактування на лінії SCK (в режимі Master). Якщо мікроконтролер працює в режимі SLAVE, то ці регістри не мають ніякого значення.

Вибір режиму роботи SPI представлений у таблиці 1.2.

Таблиця 1.2 – Вибір режиму роботи SPI

| CPOL | CPHA | Режим роботи SPI |
|------|------|--|
| 0 | 0 | MASTER переводить лінію SS в стан логічного 0. За спаданням фронту сигналу відповідний ведений блок включається в обмін даними. Передача починається з того, що пристрій MASTER поміщає в свій регістр SPDR байт, що підлягає передачі. Після 8-го тактового імпульсу лінії SCK передача завершується, в регістрах SPSR встановлюється прапор SPIF. Одночасно зі скиданням лінії SS в початковий стан (логічна 1) ведучий пристрій завершує передачу, ведений пристрій стає неактивним |
| 0 | 1 | |
| 1 | 0 | |
| 1 | 1 | |

Вибір режиму тактування представлений в таблиці 1.3., а формат регістру стану SPSR представлений в таблиці 1.4.

Таблиця 1.3 – Вибір режиму тактування

| SPR1 | SPR0 | Частота SCK |
|------|------|---------------------------------------|
| 0 | 0 | Тактова частота мікроконтролера / 4 |
| 0 | 1 | Тактова частота мікроконтролера / 16 |
| 1 | 0 | Тактова частота мікроконтролера / 64 |
| 1 | 1 | Тактова частота мікроконтролера / 128 |

Таблиця 1.4 – Формат регістру стану SPSR

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|---|---|---|---|---|---|
| SPIF | WCOL | - | - | - | - | - | - |

Прапор WCOL встановлюється у разі, коли під час передачі даних робиться спроба запису в регістр даних SPI.

Прапор SPIF вказує на завершення передачі і викликає запит на переривання (якщо переривання дозволені).

1.5 Запам'ятовуючий пристрій

Для ведення реєстру рівнів напруг буде потрібно досить великий обсяг доступної пам'яті, і явно власної пам'яті мікроконтролера буде замало. Тому буде потрібне застосування зовнішньої пам'яті. В даний час отримали достатньо широкого поширення так звані мультимедіа карти пам'яті (MicroSD – Micro Solid State Drive). В такій карті по мимо самого запам'ятовуючого пристрою (Flash-пам'яті) знаходиться електронна схема, що дозволяє організувати зв'язок з пристроєм по послідовному каналу (інтерфейс SPI), що дуже зручно, так як в багатьох мікроконтролерах є вбудований послідовний приймач-передавач такого типу. Об'єм таких карт пам'яті від декількох мегабайт до декількох сотень гігабайт і навіть терабайт.

Багато персональних комп'ютерів, ноутбуків, планшетів, смартфонів мають так звані card ридери, що дозволяють зчитувати дані з карт пам'яті.

Для того, щоб з карти пам'яті зчитати дані на будь-який пристрій, потрібна наявність на карті пам'яті файлової системи, інакше кажучи, вона повинна бути відформатована. При розмірі пам'яті до декількох десятків гігабайт, зазвичай, застосовується файлова система FAT16 або FAT12, інакше - FAT32 або ExFAT. Програмно організувати на мікроконтролері роботу з файловою системою вимагає додаткових системних ресурсів, до того ж за технічним завданням нема необхідності у перевключенні карти пам'яті до інших систем та стандартного доступу в операційних системах загального призначення. Тому, доцільним є організувати прямий доступ до запису даних в режимі SPI. Але при цьому доведеться реалізувати свій власний пристрій зчитування даних.

Існує кілька різновидів карт пам'яті, найпоширеніші: MMC і SD. SD – є більш досконалою версією MMC, і повністю сумісна з нею.

Зовнішній вигляд MMC карти пам'яті наведений на рисунку 1.4.



Рисунок 1.4 – Зовнішній вигляд MMC карти пам'яті

Даний інтерфейс дозволяє обмінюватися даними на високій швидкості, задіявши при цьому мінімальну кількість виводів мікроконтролера, які модулем SPI.

На рисунку 1.5 представлена діаграма напруги живлення і послідовність посилки команди. З рисунку видно що після включення карти необхідна вичекати кілька мілісекунд (1мс від 0.1 до 35 мс (наростання)) на стабілізацію. Протягом цього часу на CS, MOSI лінії має бути подана логічна одиниця. Далі відбувається затримка ініціалізації максимум 1 мс, при подачі на вхід CLK 74 імпульсів (тактів), після чого повинна йти команда CMD0.

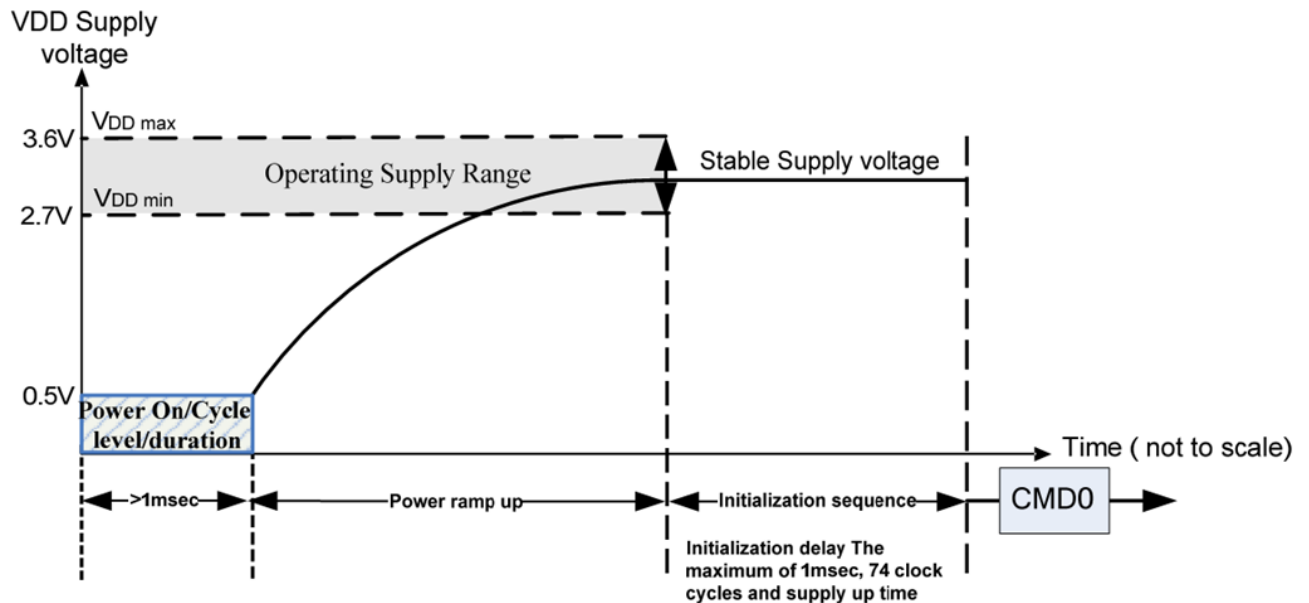


Рисунок 1.5 – Діаграма напруги живлення підчас включення карти пам'яті

SPI протокол вибирається після включення живлення та подачі команди скидання CMD0. Сама по собі карта SD працює в режимі SD. Вхід у режим здійснюється якщо сигнал SC при подачі команди CMD0 буде 0. При переході в режим SPI карта відповідає форматом R1. Цей формат, представлений на рисунку 1.6

Формат відповіді, представлений на рисунку 1.6, являє собою байт з прапорами, що визначають стан карти:

- 1-й біт – режим очікування;
- 2-й – помилка стирання;
- 3- й – невідома команда;
- 4-й – помилка команди;

5-й – помилка у послідовності стирання;

6-й – помилка адреса;

7-й – помилка аргументу.

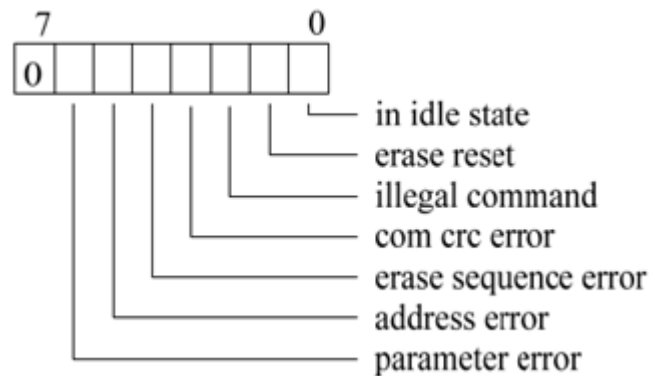


Рисунок 1.6 – R1-формат відповіді карти пам'яті

Правильні відповіді це буде 1 (у випадку команди CMD0) і нуль у інших випадках.

У специфікації є чітка послідовність ініціалізації для SPI. Для чого використовують команду CMD8 для перевірки робочого стану карти, де відбувається достатньо не простий алгоритм перевірки. Далі команда CMD58 для визначення типу карти SDS або SDHC і SDXC. А також команда CMD41 для запуску і перевірки ініціалізації. Досить не простий процес ініціалізації, але для простого запису даних можна використовувати більш спрощений процес. В режимі очікування єдиною можливою командою для картки CMD41, CMD8, CMD58, CMD59, а також для карт пам'яті SD 2.1мм – CMD1, яка ідентична команді CMD41. В стандарті ця команда вважається забороненою для ініціалізації, і використовується виключно для відмінності карт 1.4мм і 2.1мм.

Розглянемо формат команди. Кожна команда або блок даних складаються з восьми бітових байтів, які вирівнюються за сигналом CLK. Тобто, кожна команда вирівнюється по границі у 8 тактів. Повідомлення SPI

складаються з команди, відповіді і даних. Увесь зв'язок контролюється мікроконтролером. Всі команди мають довжину 6 байтів. Передача починається з першого лівого біта. В таблиці 1.5 представлений формат команди.

Старт біт – з 0 починається будь-яка команда.

Переданий біт – теж завжди дорівнює 1.

Індекс – безпосередньо команда, що передається.

Аргумент – для кожної команди аргумент вказано в таблиці специфікації.

CRC – перевірка надмірності коду. За замовчуванням у режимі SPI вона відключена. Тому вона використовується тільки для команди CMD0, яка надсилається до входу в режим і має значення CRC 0x95.

Стоп біт – кінець команди, що передається.

Таблиця 1.5 – Формат команди звернення до карти пам'яті

| | | | | | | |
|---------------------|-----------|------------------|---------------|----------|-------|---------|
| Bit position | 47 | 46 | [45:40] | [39:8] | [7:1] | 0 |
| Width (bits) | 1 | 1 | 6 | 32 | 7 | 1 |
| Value | '0' | '1' | x | x | x | '1' |
| Description | start bit | transmission bit | command index | argument | CRC7 | end bit |

2 РОЗРОБКА СТРУКТУРИ АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ

На рисунку 2.1 представлена структурна організація апаратного забезпечення багатоканальної комп'ютерної системи реєстрації аналогових сигналів. В основі апаратного забезпечення покладений мікроконтролер зі вбудованим аналого-цифровим перетворювачем та мультиплексором на 8 каналів, а також пристрої вибірки та зберігання. Для реалізації функції запам'ятовування, до складу комп'ютерної системи введено мультимедіа карту пам'яті (MMC карту пам'яті). Також, для відображення статусу кожного з 8 каналів система обладнана підсистемою світлодіодної індикації (лінійкою з 8 світлодіодів по одному для кожного каналу).

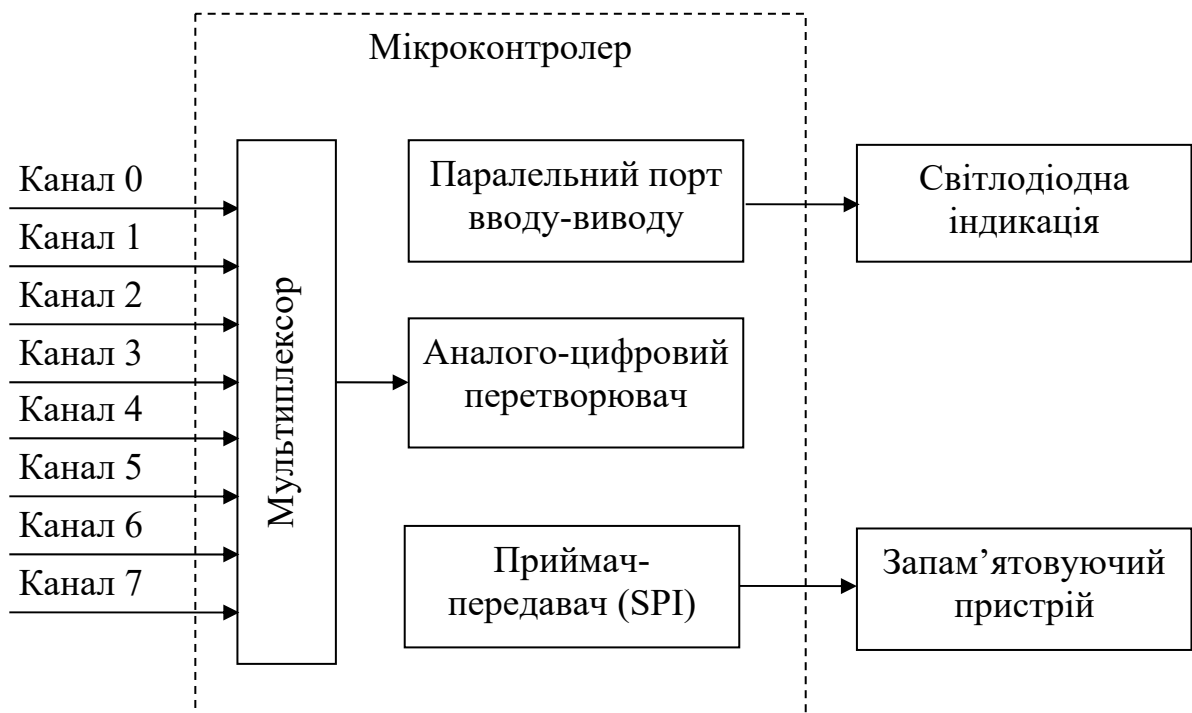


Рисунок 2.1 – Структурна організація апаратного забезпечення багатоканальної комп'ютерної системи реєстрації рівнів аналогових сигналів

На виводи мікроконтролера, які підключені до мультиплектору аналого-цифрового перетворювача подаються напруги по восьми каналах. Мікроконтролер по черзі перемикає мультиплексор, вибираючи потрібний канал. Після вибору каналу, здійснюється перетворення напруги на вході аналого-цифрового перетворювача в цифровий еквівалент. Отримані дані порівнюються з певним заданим рівнем і якщо напруга перевищує цей рівень, формується активний рівень сигналу на відповідному біті паралельного порту вводу-виводу, який засвічує відповідний світлодіод. Якщо рівень напруги на відповідному каналі не досягає певного заданого значення, то на відповідному біті порту вводу-виводу формується пасивний рівень сигналу, який забезпечує закритий стан відповідного світлодіода і він не світиться. Для простоти, порівнювати вхідну напругу будемо з половиною опорної аналого-цифрового перетворювача.

Результати кожного перетворення записуються до запам'ятовуючого пристрою за допомогою SPI-протоколу. При цьому, при запуску, мікроконтролер повинен ініціалізувати не тільки свій приймач-передавач, а й перевести карту пам'яті в режим SPI.

Для побудови спеціалізованої комп'ютерної системи передбачається використання наступних основних компонентів:

- мікроконтролер ATmega16L;
- запам'ятовуючий пристрій – карта пам'яті ММС (32 Мб);
- світлодіоди – малопотужні з резисторами, що обмежують струм через них;
- живлення від зовнішнього джерела постійного струму напругою 5-9В;
- вбудований стабілізатор напруги 3,3В для живлення елементів комп'ютерної системи.

3 РОЗРОБКА ПРИНЦИПОВОЇ СХЕМИ АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ ТА ВИБІР ЕЛЕМЕНТІВ

Для визначення схеми підключення мікроконтролера необхідно проаналізувати його характеристики.

Мікроконтролер ATmega16L має наступні основні характеристики:

- RISC-архітектура;
- вбудований 8-ми канальний 10-ти розрядний аналого-цифровий перетворювач;
- послідовний SPI-інтерфейс в режимі ведучий / відомий;
- 16 Кб вбудованої Flash-пам'яті;
- 1 Кб SRAM;
- два 8-ми бітних таймери-лічильники і один 16-ти бітний;
- внутрішні та зовнішні джерела переривань;
- тридцять дві програмовані лінії вводу-виводу;
- частота тактового генератора максимальна – 8 МГц;
- напруга живлення – 2.7-5.5В;
- струм в активному режимі – 1.1мА.

Ці данні необхідні для побудови принципової схеми апаратного забезпечення та обрання електричних режимів. Але, крім цього, для побудови схеми з'єднань необхідно проаналізувати цоколювку мікроконтролера (його footprint).

Цоколювка виводів мікроконтролера для варіанта PDIP-корпусу наведена на рисунку 3.1.

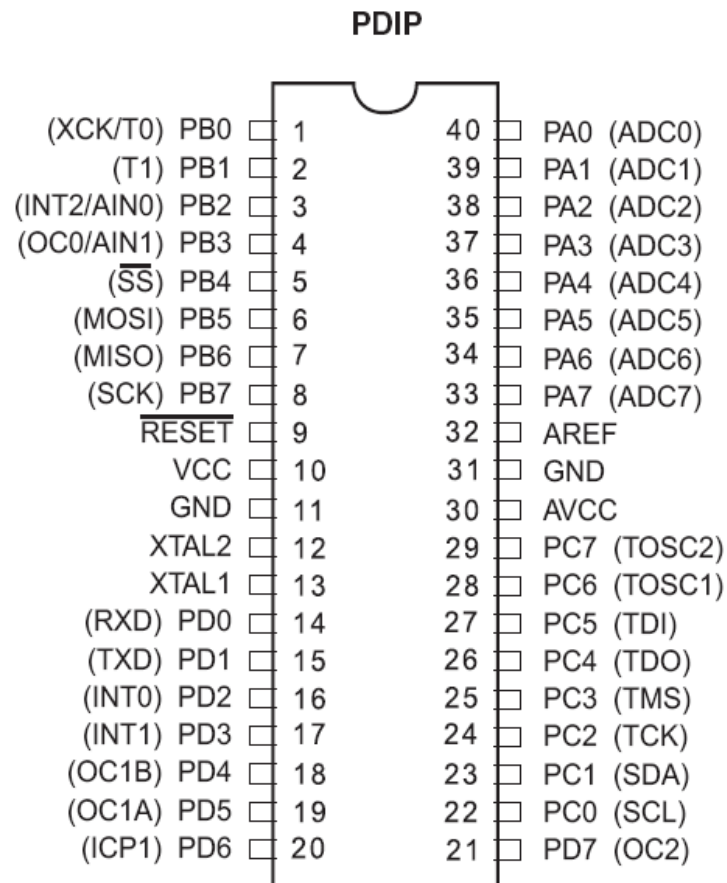


Рисунок 3.1 – Распиновка виводов ATmega16 в PDIP корпусе

На електричній принциповій схемі мікроконтролер буде зображатися так, як показано на рисунку 3.2.

Мікроконтролер також потребує обв'язування яке забезпечує його належне функціонування у режимах, які рекомендовані виробником. Таке обв'язування, зазвичай, використовують стандартне, яке зазначається виробником у документації до нього.

До елементів обв'язки мікроконтролера входить:

- зовнішній кварцовий резонатор (контролер може використовувати вбудований резонатор, але, при цьому, його частота буде не достатньо стабільна);
- резистор для підтяжки виводу скидання до позитивного полюсу напруги живлення;

– фільтруючі конденсатори в ланцюгу жавлення для забезпечення фільтрації можливих комутаційних перешкод, що можуть викликати відмови в роботі мікроконтролера;

– RC фільтр для фільтрації опорної напруги аналого-цифрового перетворювача від високочастотних перешкод.

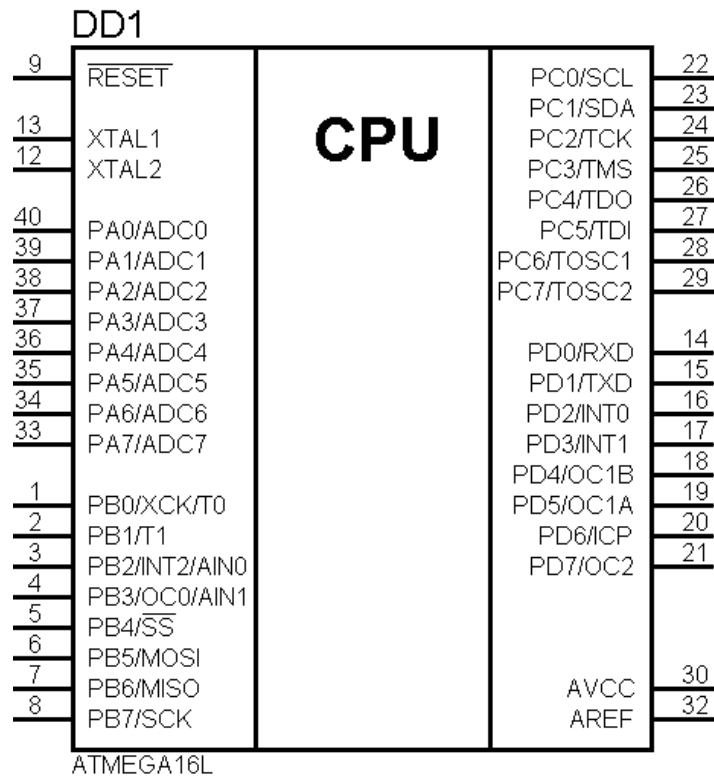


Рисунок 3.2 – Умовне позначення мікроконтролера на електричній принциповій схемі апаратного забезпечення комп'ютерної системи

ММС карта пам'яті повинна бути знімна, тому для неї буде передбачений слот підключення.

Стандартна схема контактів ММС карти пам'яті зображена на рисунку 3.3.

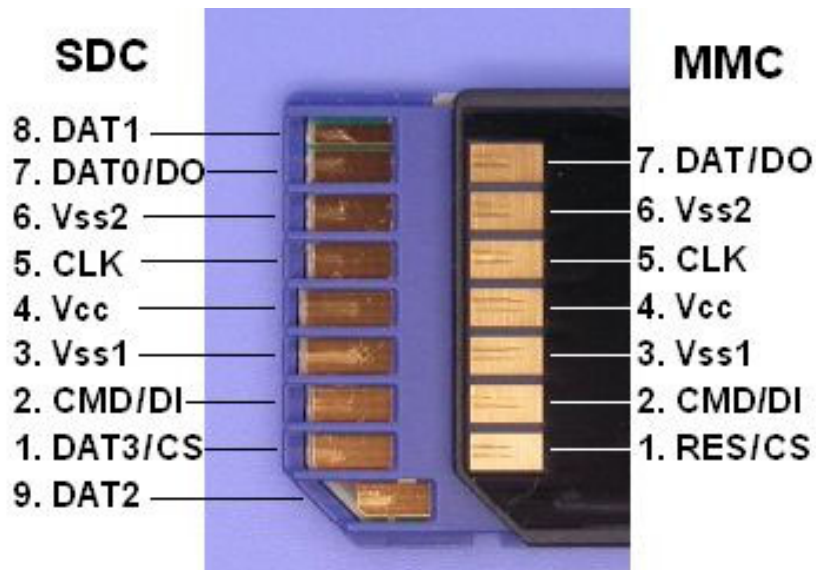


Рисунок 3.3 – Стандартна схема контактів MMC карти пам'яті

Для SPI-режиму доступу до MMC карти пам'яті необхідне використання наступних контактів:

- вивід 1 – CS (Crystal Select або в деяких інших застосуваннях використовується позначення Chip Select);
- вивід 2 – DI (Data In), MOSI (Master Out Slave In);
- вивід 3 – Vss (загальний вивід);
- вивід 4 – Vcc (вивід подавання напруги живлення);
- вивід 5 - CLK (вивід тактування);
- вивід 6 – Vss (загальний вивід);
- вивід 7 – DO (Data Out), MISO (Master In Slave Out);

Напруга живлення для MMC карт пам'яті зазвичай дорівнює 3.3В. Для того, щоб не погоджувати рівні напруги логіки мікроконтролера і карти пам'яті був обраний мікроконтролер зі зниженою напругою живлення.

Як вже було сказано в попередньому розділі, карту пам'яті потрібно перевести в SPI режим. Режим SPI є альтернативним режимом роботи з MMC / SDC, більш простим порівняно з використанням їхнього рідного інтерфейсу. MMC / SDC можна підключити через універсальний порт SPI або порт GPIO (універсальний порт вводу / виводу), вбудований в більшість мікроконтролерів, а також одноплатних комп'ютерів на кшталт RaspberryPi,

OrangePi, LatePanda та інші. Тому, режим SPI чудово підходить для недорогих вбудованих комп'ютерних систем. MMC не використовують синхронізацію SPI як таку, а обидва дії: защелкування і зсуву визначаються по передньому фронту SCLK, начебто це працює в режимі "SPI 0". Тому найбільш підходящим режимом для роботи з MMC / SDC по SPI є режим "SPI 0" (позитивним синхроімпульс, защелкування по передньому фронту, зсув по задньому фронту). Також, слід зазначити, що режим "SPI 3" теж підходить для цих цілей.

На основі вищесказаного, була сформована електрична принципова схема апаратного забезпечення багатоканальної спеціалізованої комп'ютерної системи реєстрації рівнів аналогових сигналів, яка наведена на рисунку 3.4.

Конденсатори C3, C4, C5 призначені для фільтрації живлячої напруги. Оберемо конденсатори C3, C4 електролітичні на 10мкФ для усунення низькочастотних перешкод, а C5 – 100нФ для фільтрації високочастотних перешкод. Оскільки сам мікроконтролер, під час його роботи, є джерелом високочастотних перешкод, C5 має бути розташований якомога ближче до його виводів. Ємності цих конденсаторів не розраховуються, а беруться відповідно до рекомендацій типової схеми включення мікроконтролера, яка регламентується виробником мікроконтролера. Таким чином, на наведній на рисунку 3.4 принциповій схемі C3 і C4 – 10мкФ, а C5 – 0,1мкФ.

Мікросхема-стабілізатор напруги обирається за напругою стабілізації та максимальним допустимим струмом. Оскільки сам мікроконтролер споживає близько декілька міліампер, основна частина струму піде на живлення мікросхеми карти пам'яті (до 100мА), і на живлення лінійки світлодіодів (до 160мА). Таким чином, параметри мікросхеми-стабілізатора напруги повинні бути наступними:

- напруга стабілізації 3,3 В;
- максимальний струм до 300мА.

Цим вимогам цілком задовольняє мікросхема LD1086V33, яка і прийнята для використання в схемі.

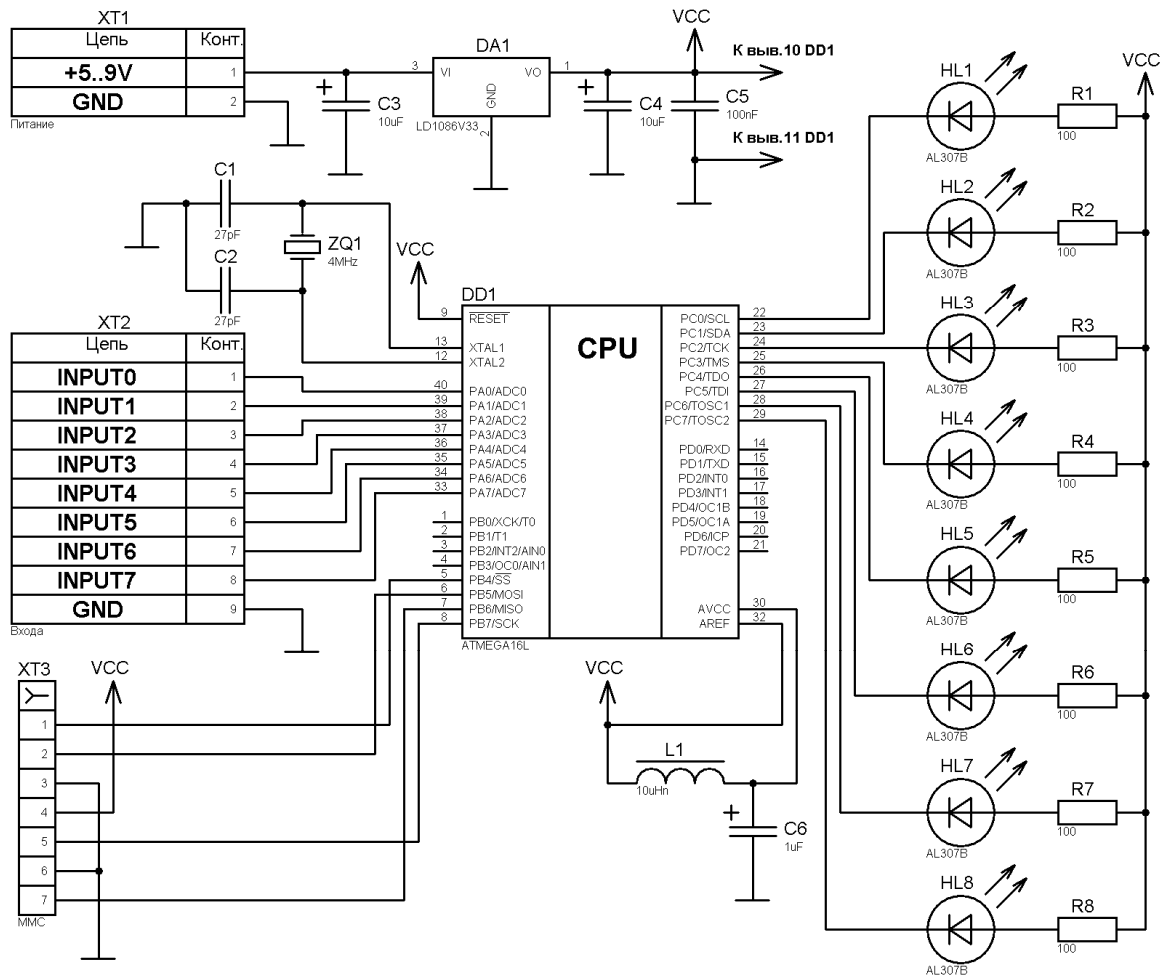


Рисунок 3.4 – Електрична принципова схема апаратного забезпечення багатоканальної спеціалізованої комп'ютерної системи реєстрації рівнів аналогових сигналів

Вивід Reset мікроконтролера підключаємо до позитивного полюсу живлення, таким чином перевівши мікроконтролер до активного режиму. Скидання при подачі живлення передбачене у схемі самого контролера, зовнішньою схемою потрібно тільки встановити час через який мікроконтролер перейде в активний режим за допомогою певних Fuse-бітів.

Частота тактового генератора даного мікроконтролера при використанні зовнішнього кварцового резонатора має знаходитися в межах від 1 до 8 МГц. До установки приймемо кварцовий резонатор на частоту 4 МГц. Ємності конденсаторів C1 і C2 рекомендуються в довідковій документації до контролеру та можуть бути в межах 12...27пкФ. Приймемо до установки конденсатори C1 і C2 ємністю по 27пкФ.

Виводи каналів, на яких потрібне вимірювати напругу, підключаються, безпосередньо, до виводів мікроконтролера adc0...adc7. Опорна напруга, як вже було сказано, буде однаковою з напругою живлення, тому вивід мікроконтролера AREF підключаємо до позитивного полюсу джерела живлення.

Живлення аналого-цифрового перетворювача мікроконтролера здійснюється по виводу AVCC. Для зниження шумів по ланцюгу живлення, підвищення точності вимірювання, а також для уникнення самозбудження рекомендується установлювати LC фільтр. Ємність конденсатора і індуктивність дроселя приймемо такі, які рекомендує документація до мікроконтролеру: C6 –1мкФ (бажано танталовий), L1 –10мкГн.

Головним критерієм для вибору світлодіодів є максимальний струм, що протікає через них під час спускання світла, він не повинен перевищувати максимально допустимий струм порту мікроконтролера. Для АТМega16 максимальний струм для одного виводу порту – 40 мА, струм споживаний контролером не повинен перевищувати 200мА. Таким чином струм одного світлодіода не повинен перевищувати 20мА. Приймемо до установки світлодіоди AL307В. Його максимальний прямиий струм – 20мА, пряме падіння напруги при струмі 10мА – 2В. На основі цих даних розрахуємо резистори R1-R8, які будуть обмежувати струм через світлодіоди:

$$R_{1-8} = \frac{U_{II} - U_{HL_PP.}}{I_{HL_PP.}}. \quad (4.1)$$

$$R_{1-8} = \frac{U_{II} - U_{HL_PP.}}{I_{HL_PP.}} = \frac{3 - 2}{15 * 10^{-3}} = 66.7 \text{ Ом.}$$

До установки приймемо резистори опором 68 Ом. Так як резистори працюють в ланцюзі малих струмів, то для встановлення підійдуть будь-які резистори, починаючи з потужності 0.125Вт.

Карта пам'яті повинна підключатися до виводів мікроконтролера, пов'язаними з послідовним синхронним приймачем-передавачем SPI, а саме CLK, MOSI, MISO. І ще один вивід порту буде потрібний для операції скидання карти пам'яті.

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ РЕЄСТРАЦІЇ АНАЛОГОВИХ СИГНАЛІВ

4.1. Розробка структури алгоритмів програмного забезпечення

Програмне забезпечення повинно реалізовувати послідовне переключення каналів аналогових сигналів для аналого-цифрового перетворення, а також запуск перетворення на аналого-цифровому перетворювачі через рівні проміжки часу. При цьому, аналого-цифровий перетворювач повинен бути попередньо ініціалізований.

Для вимірювання часових інтервалів скористаємося вбудованим таймером. Таймер T1 має більшу роздільну здатність ніж T0, а отже і точність, тому використовуємо саме його. Таймер також повинен бути попередньо ініціалізований.

При спрацьовуванні переривання від таймера (наприклад, після переповнення) буде запускатися підпрограма обробки переривання, в тіло якої помістимо код, що активує перетворення на аналого-цифровому перетворювачі, і подальше зберігання отриманих даних у пам'яті.

Особливістю роботи з ММС-картами пам'яті є те, що дані пишуться і зчитуються блоками. Розміри блоків можуть відрізнятися, для карт пам'яті ємністю 16, 32, 64 МБ вони зазвичай дорівнюють 512 байт. Тому спочатку будемо заповнювати масив розміром 512 байт в оперативному запам'ятовуючому пристрої мікроконтролера, а коли масив заповниться – записувати його до карти пам'яті. Але попередньо буде потрібно ініціалізувати SPI і карту пам'яті.

Алгоритм роботи основного циклу програмного забезпечення можна зобразити у вигляді блок – схеми, яка наведена на рисунку 4.1.

Основний цикл включає в себе початкову ініціалізацію інтервального таймера T0, послідовного приймача-передавача SPI, карти пам'яті ММС, яка настроюється на запис блоками по 512 байт у режимі SPI. Надалі, після ініціалізації, здійснюється перехід у основний цикл очікування переривань, за якими будуть виконуватися дії по переключенню аналогового каналу, аналого-цифрового перетворення, а також зберігання даних до оперативної пам'яті, а після наповнення блоку у 512 байт – запису його до карти пам'яті.

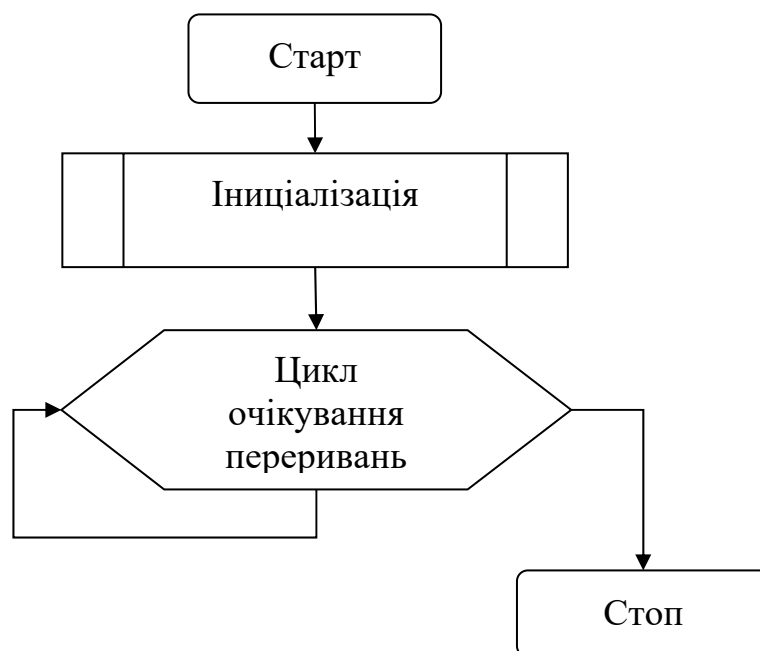


Рисунок 4.1 – Блок-схема основного циклу програмного забезпечення багатоканальної комп'ютерної системи реєстрації аналогових сигналів

Блок-схема алгоритму початкової ініціалізації показана на рисунку 4.2.

Процес ініціалізації включає наступні операції:

- ініціалізація портів вводу виводу, при якій здійснюється налаштування портів вводу виводу на режими: так порт для управління світлодіодами налаштується на вивід даних, а порт, який пов'язаний зі входами комутатору аналогових сигналів – на ввід даних;

– етап ініціалізації змінних полягає у створенні та резервуванні пам'яті для змінних, що будуть зберігати дані, що отримуються в процесі аналого-цифрового перетворення, а також дані для зберігання вмісту внутрішніх регістрів таймеру, а також буферу послідовного приймача-передавача даних, для обміну даними з мультимедіа картою пам'яті;



Рисунок 4.2 – Блок-схема алгоритму початкової ініціалізації

- ініціалізація аналого-цифрового перетворювача полягає у визначенні режиму роботи, засобами завантаження слова стану до регістру спеціального призначення;

- ініціалізація таймеру полягає у завданні його режиму, через завантаження слова-стану до внутрішнього регістру спеціального призначення, встановлення початкового стану керуючих бітів таймера, а також завантаження числа для рахування, яке буде визначати часовий інтервал спрацьовування таймера;

- ініціалізація послідовного приймача-передавача включає операції визначення напрямку передачі даних для його виводів, а також вибір режиму його роботи;

- завершення процедури ініціалізації полягає у дозволі переривань, що дозволяє подальшу обробку даних та процедури їх зберігання виконувати за запитами переривання таймера, а також здійснюється запуск таймера, який надалі працює в режимі автоперезавантаження.

Логічна послідовність операцій при надходженні переривання від переповнення таймера підчас рахування, представлена на рисунку 4.3.

Реініціалізація таймера-лічильника полягає у підготовці його для наступного циклу лічіння. Таким чином, на цьому етапі здійснюється останов таймера та перезавантаження його внутрішнього регістру лічіння новими даними, які визначають інтервал рахування.

Надалі виконується процес аналого цифрового перетворення напруги, яка подана на вхідний канал n , результатом якого є отримання двохбайтового коду, що відповідає рівню напруги.

Отримане шістнадцятирозрядний число порівнюється з програмно заданим рівнем (як вже було сказано раніше, рівень порівняння відповідає половині опорної напруги аналого-цифрового перетворювача, але це значені може задаватися іншим, за однією тільки умовою – воно повинно мати значення, яке лежить в межах динамічного діапазону перетворювача).

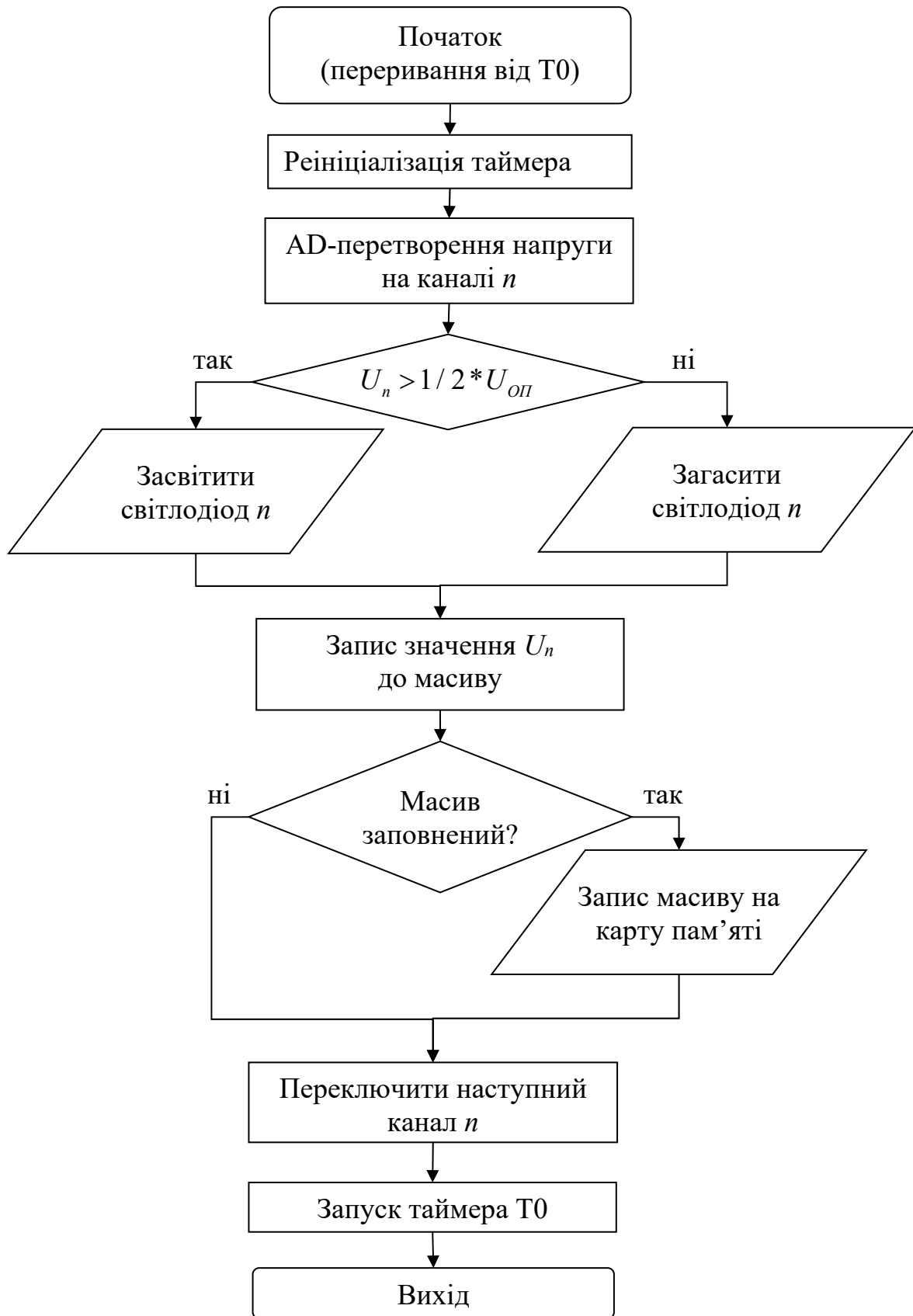


Рисунок 4.3 – Блок-схема алгоритму обробки переривання від інтервального таймеру-лічильнику T0

Якщо в процесі порівняння перетворене значення перевищує заданий поріг, то на відповідний порт вводу виводу формується високий рівень сигналу, який засвічує відповідний світлодіод. У протилежному випадку – відповідний біт отримує низький рівень і світлодіод не світиться. Стан кожного світлодіода зберігається між моментами аналізу відповідного каналу.

Після порівняння, отримане значення вноситься до масиву. Надалі, якщо масив заповнюється, тобто обсяг даних в ньому досягає значенню, що відповідає блоку запису до карти пам'яті (512 байт) данні з масиву записуються до карти пам'яті.

Наступним етапом здійснюється переключення активного каналу для того, щоб на наступному циклі обробки переривання, коли таймер буде переповнений, знову здійснити обробку по цьому алгоритму для наступного каналу.

Після цього здійснюється запуск таймера і він починає відраховувати наступний часовий інтервал.

4.2 Розробка програмного коду

Для розробки програмного коду знадобяться знання про регістри управління мікроконтролера, які відповідають за роботу аналого-цифрового перетворювача, послідовного приймача-передавача SPI, портів вводу-виводу, таймера.

За роботу портів відповідають три регістри (для кожного порту вводу-виводу):

- DDRx – вказує в якому напрямку працюють виводи порту x. Якщо біт встановлений, вивід порту працює на виведення;

- PINx – повертає стан виводів на порті x;

- PORTx – визначає рівні логічного сигналу на виводах порту x, якщо порт працює на виведення. Якщо вивід порту включений на введення, то установка біта дозволяє підключити внутрішній підтягуючий резистор;

Аналого-цифровий перетворювач управляється наступними регістрами:

– ADMUX – дозволяє вказати джерело опорної напруги, управляє мультиплексором;

– ADCSRA – дозволяє включати / відключати аналого-цифровий перетворювач, запускати перетворення, обирати частоту тактування;

- ADCL і ADCH – відповідно молодший і старший байт результату перетворення;

За роботу таймера T0 відповідають наступні регістри:

– TCCR0B – дозволяє вибрати дільник частоти системного тактового генератора (далі ці тактові імпульси будуть підраховуватися лічильником таймера);

– TCNT0H і TCNT0L – регістри де зберігається поточне значення таймера / лічильника;

– TIMSK – регістр, що дозволяє переривання від таймера.

За роботу послідовного приймача-передавача SPI відповідають наступні регістри:

– SPCR – дозволяє запускати послідовний приймач-передавач, дозволяти його переривання, визначати деякі дії;

– SPDR – реєстр даних послідовного приймача-передавача, куди заносяться дані для передачі і зчитуються при прийомі.

Дана периферія достатньо багатофункціональна і регістрів, що відповідають за її роботу трохи більше. Також і виконують вони більше функцій ніж було перераховано. Тут же наведені лише ті, які потрібні для розробки програмного забезпечення.

Програмний код було розроблено на мові програмування C. Причина цьому – її простота і компактність (порівняно з асемблером), вільно поширювані якісні компілятори.

Для розробки програмного коду був використаний компілятор AVR-GCC. Зручним середовищем розробки програмного забезпечення є також вільно-поширювана AVR Studio від Atmel.

Код програмного забезпечення багатоканальної комп'ютерної системи реєстрації аналогових сигналів наведений далі, разом з коментарями.

```
#include "avr/io.h" //підключення бібліотеки
                        // вводу-виводу
#include "avr/interrupt.h" //підключення бібліотеки
                        // прерываний

//Ініціалізація змінних
unsigned int vbuf, n=0; //змінні для зберігання
                        // результату перетворення, а
                        // також його номера
unsigned char pos=0, sector[512]; // змінна, що буде
                        // містити поточний канал, а також
                        // ініціалізація масиву для
                        // накопичення даних блока для
                        // запису на карти пам'яті

// ініціалізація порту вводу-виводу
#define SPIDI      6    // біт 6 порту SPI (пін7):
                        // дані з MMC
#define SPIDO      5    // біт 5 порту SPI (пін6):
                        // дані в MMC
```

```

#define SPICLK    7    // біт 7 порту SPI (пін8):
                    // синхронізація
#define SPICS     4    // біт 4 порту SPI (пін5):
                    // сигнал вибору мікросхеми

// Підпрограма пересилки команди до порту SPI
char SPI(char d)
{ // пересилка символу через SPI
  char received = 0;
  SPDR = d;
  while(!(SPSR & (1<<SPIF)));
  received = SPDR;
  return (received);
}

// Підпрограма пересилки даних до порту SPI
char Command(char befF, unsigned int AdrH, unsigned int
AdrL, char befH )
{
  SPI(0xFF);
  SPI(befF);
  SPI((unsigned int)(AdrH >> 8));
  SPI((unsigned int)AdrH);
  SPI((unsigned int)(AdrL >> 8));
  SPI((unsigned int)AdrL);
  SPI(befH);
  SPI(0xFF);
  return SPI(0xFF); // повертається останній
                    // переданий символ

```

```

}

// Підпрограма ініціалізації SPI
void SPIinit(void)
{
    DDRB&=~(1<<SPIDI); // порт B SPI data input на вивід
    DDRB|=(1<<SPICLK); // порт B SPI clock на вивід
    DDRB|=(1<<SPIDO); // порт B SPI data out на вивід
    DDRB|=(1<<SPICS); // порт B SPI chip select на вивід
    SPCR=(1<<SPE) | (1<<MSTR) | (1<<SPR1) | (1<<SPR0); //
                                                    //режими роботи SPI
    PORTB&=~(1<<SPICS); // chip select в низький
                        // рівень (MMC обрано)
}

// Підпрограма ініціалізації MMC
int MMC_Init(void)
{
    char i;
    PORTB |= (1 << SPICS); // відключити MMC
    // старт MMC в SPI режимі
    for(i=0; i < 10; i++) SPI(0xFF); // послідовно 10*8=80
                                        // тактових імпульсів
    PORTB &= ~(1 << SPICS); //включили MMC
    if (Command(0x40,0,0,0x95) != 1) goto mmcerror;
                                                    // скидання MMC
st: // якщо MMC відсутня, очікуємо її
    if (Command(0x41,0,0,0xFF) !=0) goto st;
    return 1;
}

```

```

mmcerror:
return 0;
}

// Підпрограма запису RAM сектору до MMC
int writeramtommc(void)
{
int i;
unsigned int c;
// режим запису по 512 байт
if (Command(0x58,0,512,0xFF) !=0)
return 1;
SPI(0xFF);
SPI(0xFF);
SPI(0xFE);
// записуємо сектор RAM в MMC
for (i=0;i<512;i++)
{
SPI(sector[i]);
}dummy
//наприкінці пересилається 2 завершуючих байти
SPI(0xFF);
SPI(0xFF);
c = SPI(0xFF);
c &= 0x1F; //0x1F = 0b.0001.1111;
if (c != 0x05)
// 0x05 = 0b.0000.0101
return 1;
// очікуємо доки MMC не звільниться

```



```

while (SPI(0xFF) != (char)0xFF);
return 0;
}

// Підпрограма загальної ініціалізації
void init(void)
{
    SPIinit(); // ініціалізація SPI
    MMC_Init(); // ініціалізація MMC
    DDRC=0xff; // порт C на виведення
    ADCSRA|=(1<<ADEN) | (1<<ADPS2) | (1<<ADPS1); // режими
                                                //аналого-цифрового перетворювача
    TIMSK=(1<<TOIE1); // ініціалізація таймера
    TCCR1B=(1<<CS11);
    TCNT1L=0xED;
    TCNT1H=0x85;
    sei(); // дозвіл переривань
}

// Підпрограма обробки переривання при переповненні
                                                // таймера
ISR(TIMER1_OVF_vect)
{
    TCNT1L=0xED; // ініціалізація таймера
    TCNT1H=0x85;
    ADMUX=pos; // переключення каналу
    ADCSRA|=(1<<ADSC); //старт перетворення
    while (ADCSRA&(1<<ADSC)); // очікування доки
                                                // перетворення не закінчиться
}

```

```

vbuf=ADCL+(ADCH<<8); // запам'ятовування результату
                        // перетворення
sector[n]=(unsigned char)vbuf;
n++;
sector[n]=(unsigned char)(vbuf>>8); // запис до RAM
if(vbuf<512) PORTC|=(1<<pos); // якщо напруга на
                                // каналі менша
                                // за половину опорної -
                                // погашення світлодіода
else PORTC&=~(1<<pos); // інакше засвічення
                        // світлодіода

n++;
if(n>511) // якщо сектор заповнений
{
    n=0;
    writeramtommc(); // запис до MMC
}
if((++pos)>7) // після останнього каналу 7
              // обираємо канал 0
{
    pos=0;
}
sei();
}

// Головна програма
int main(void)
{
    init();

```

```
while(1);  
return 0;  
}
```

5 МОДЕЛЮВАННЯ РОЗРОБЛЕНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ

Для моделювання та перевірки коректності функціонування розробленого апаратно-програмного забезпечення було використано середовище автоматизованого проектування та моделювання Proteus. З цією метою була розроблена імітаційна модель апаратного забезпечення розробленої багатоканальної комп'ютерної системи реєстрації аналогових сигналів, вигляд якою представлений на рисунку 5.1.

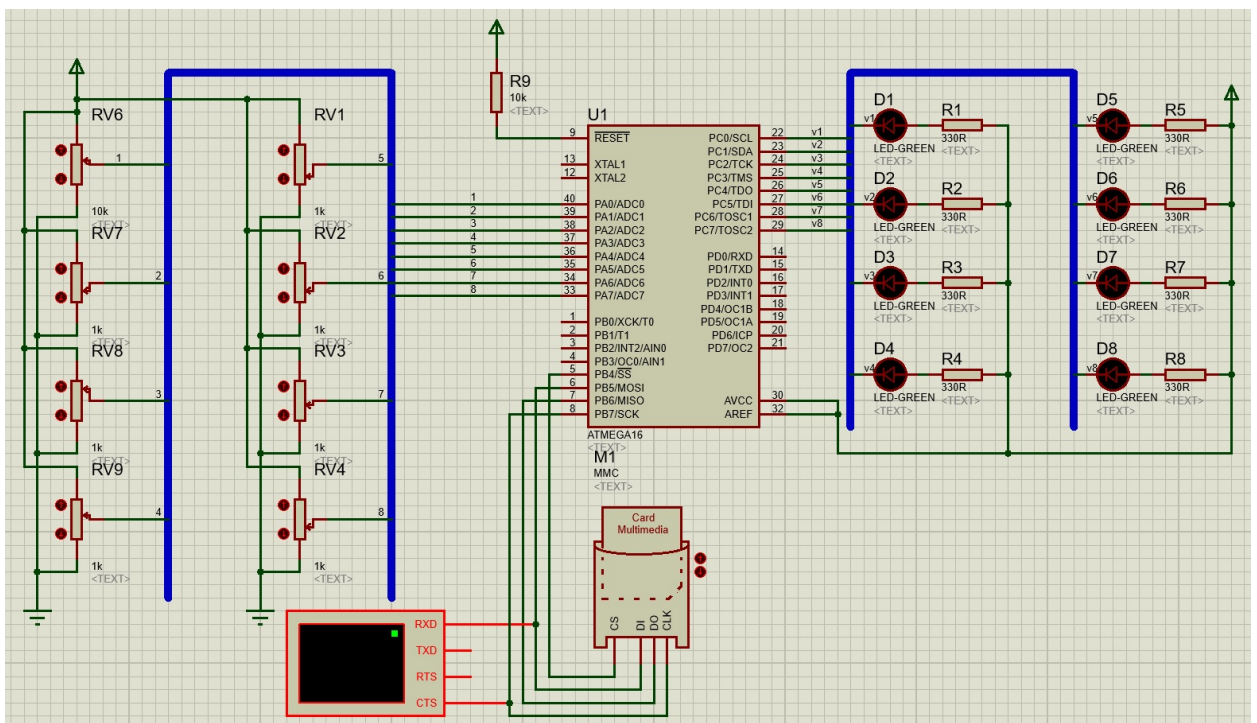


Рисунок 5.1 – Вигляд розробленої імітаційної моделі багатоканальної комп'ютерної системи реєстрації аналогових сигналів

В розробленій моделі, можливість зміни входних аналогових рівнів імітована вісьмома змінними резисторами, які включені по схемі потенціометрів. Таким чином, переводячи середній вивід кожного резистора ближче до нижнього виводу, моделюється зменшення рівня входного

сигналу, а переведення середнього виводу ближче до верхнього виводу – призводить до збільшення рівня вхідного сигналу.

Для наглядної ілюстрації процесу передачі даних на карту пам'яті до моделі введений елемент Virtual Terminal, який настроєний на вивід даних у шістнадцятирозрядному форматі.

Після запуску моделі на виконання, здійснюється ініціалізація, в тому числі і карти пам'яті. Процес пересилки управляючих байтів ініціалізації по інтерфейсу SPI від мікроконтролера до карти пам'яті представлений на рисунку 5.2.

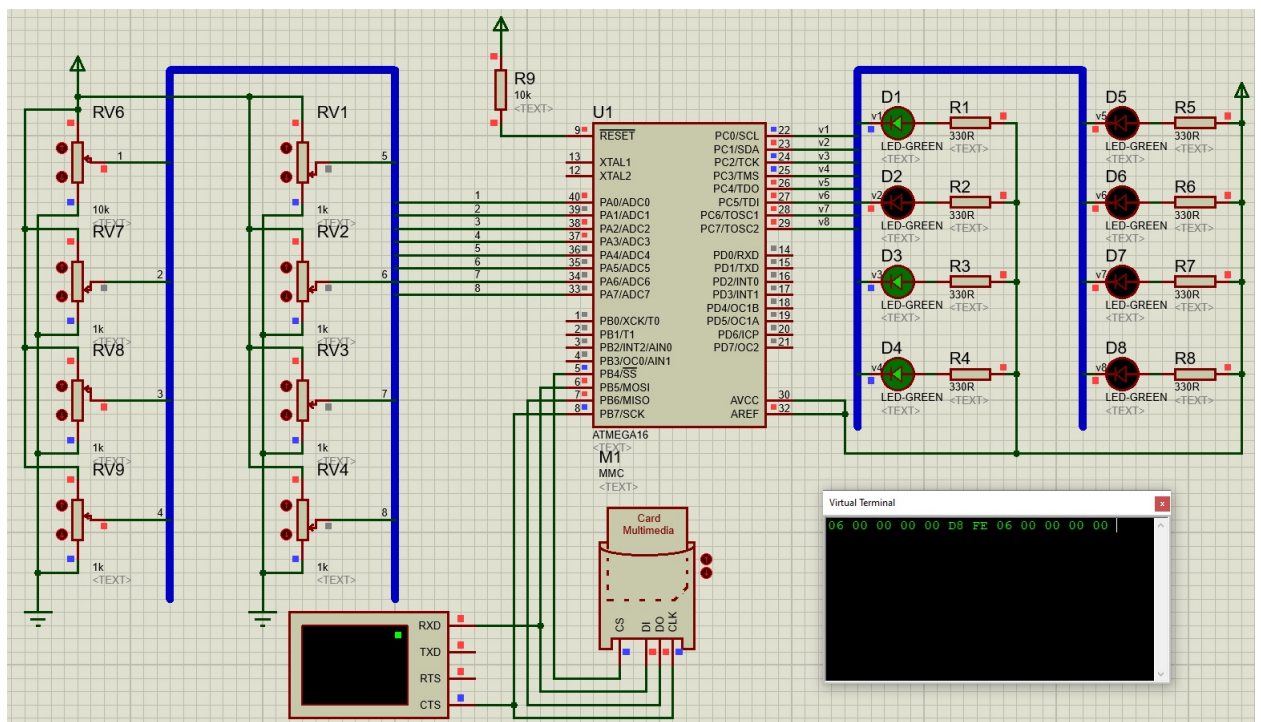


Рисунок 5.2 – Ілюстрація процесу ініціалізації карти пам'яті

Також, на рисунку 5.2 представлений процес інформування про порівняння поточних рівнів аналогових сигналів на входах світлодіодною індикацією. Так, згідно моделі видно, що за поточних значень резисторів, що задають рівні вхідних напруг, на трьох каналах є перевищення рівня половини опорної напруги. Це також підтверджується положеннями середніх

сторонніми програмами, наприклад Winimage. Ілюстрація створеного файлу представлена на рисунку 5.4.




| Имя | Дата изменения | Тип | Размер |
|--|------------------|-----------------|--------|
|  firmware.hex | 18.04.2020 21:45 | Файл "HEX" | 4 КБ |
|  image.mmc | 15.04.2020 16:36 | Файл "MMC" | 1 КБ |
|  simulation | 18.04.2020 19:24 | Proteus Project | 21 КБ |

Рисунок 5.4 – Ілюстрація створеного файлу образу вмісту карти пам'яті

ВИСНОВКИ

Результатом даної роботи стала розробка багатоканальної комп'ютерної системи реєстрації аналогових сигналів та запису реєстру рівнів на зовнішню карту пам'яті, що працює в режимі SPI.

Розроблена система здатна вести спостереження за кількома джерелами напруги, отримані дані записуються до реєстру, що зберігається у енергонезалежної пам'яті. Перевищення напруги на вході певного рівня відображається світінням світлодіодів, відповідних входів.

Перевірку функціонування розробленого апаратно-програмного комплексу було проведено у середовищі моделювання Proteus. Результати моделювання показали коректність розробленого апаратно-програмного забезпечення.

Недоліком розробленої системи є використання SPI режиму для запису карти пам'яті, що унеможлиблює її читання на інших пристроях, що обладнані картридерами відповідного формату. В процесі моделювання вміст образу може бути переглянутий спеціальним програмним забезпеченням, таки як Wunimage, але, на сьогодні, це програмне забезпечення не розповсюджується вільно.

Однак, вказаний недолік також має і перевагу з точки зору захисту даних. Записані дані не можуть бути прочитані іншими пристроями, а значить можуть передані для обробки та редагуваннями за межами розробленої комп'ютерної системи.

Для розширення можливостей по операціям з картою пам'яті можливи два варіанти:

- 1) використання драйверів файлової системи FAT для запису в стандартному форматі, але при цьому дані будуть доступні для перегляду та редагування на сторонніх пристроях;

2) модернізація розробленої комп'ютерної системи з точки зору запровадження функції двонапрявленого доступу до карти пам'яті та можливістю відображення записаних на неї даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1) http://ru.wikipedia.org/wiki/Аналогово-цифровой_преобразователь
- 2) Как использовать карты MMC / SDC. http://elm-chan.org/docs/mmc/mmc_e.html
- 3) Описание алгоритма работы с картой SD MMC. www.mindrunway.ru
- 4) Atmel ATmega (ATmega16 / ATmega32) - MMC (Multi Media Card) Flash Memory Extensio. <http://www.captain.at>
- 5) Документация к ATmega16. www.atmel.com/dyn/resources/prod_documents/doc2466.pdf
- 6) Микроконтроллеры AVR: от простого к сложному (Серия “Библиотека инженера”) / М.С. Голубцов – М.: СОЛОН-Пресс, 2003. 288с.
- 7) Справочник по светодиодам. http://www.radio-portal.ru/index.php?name=EZCMS&page_id=311