

УДК 681.3.06

*Сергій Васильович Ленков,
Юрій Дмитрович Іванов,
Ігор Володимирович Пампуха,
Дмитро Володимирович Паярелі*

ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ СТРУКТУРНО-ЛОГІЧНИХ КОДУЮЧИХ ТА ДЕКОДУЮЧИХ ПЕРЕТВОРЕНЬ ІДНФ БФ ПРИ ВИКОРИСТАННІ МІКРОКОНТРОЛЕРІВ

Вступ та постановка завдання. Основні положення теорії структурно-логічних кодуючих та декодуючих перетворень інфімумних диз'юнктивних нормальних форм булевих функцій (БФ), які представляють дискретні дані, докладно викладенні в роботах автора [1, 2].

У роботі [3] показана можливість забезпечення необхідної завадостійкості дискретних даних під час оброблення їх у ЕОМ як на апаратному, так і на програмному рівні шляхом реалізації універсальної матриці структурно-логічних кодуючих і декодуючих перетворень у термінах n -мірного куба E^n для організації єдиного коду чого формату (ЄКФ), тобто кодової комбінації кода СЛК.

В основі реалізації універсальних матриць перетворень лежить використання природної надмірності логічних зв'язків змінних розгортання n -мірних кубів E^n , [4] яка задається природним чином при побудові кодових комбінацій структурно-логічних кодів (СЛК), що є в загальному випадку багатомірними кубами ЄКФ, одержаних при кодуванні ІДНФ БФ.

Мета статті — розглянути можливість реалізації універсальної матриці структурно-логічних перетворень для організації коду чого формату (ЄКФ), кодової комбінації коду СЛК. *Завдання статті:* обґрунтувати можливість застосування структурно-логічних перетворень, що кодують і декодують інфімумні диз'юнктивні нормальні форми (ІДНФ) булевих функцій (БФ), а також застосування мікроконтролера форми Atmel, а саме AVR-RISC Atmel, що дає змогу використовувати мову "С", що значно допомагає спростити складність прила-

ду, досягти гнучкості системи, а також використовувати новітні алгоритми перетворення ІДНФ БФ.

Основний зміст. Для проведення структурно-логічних кодуючих і декодуючих перетворень кон'юнкцій ІДНФ БФ необхідно організувати ЄКФ у вигляді куба n -ї мірності E^n , який забезпечував би достатню логічну надмірність змінних розгортання куба E^n для корекції завад на довжині кодової комбінації СЛК.

Довжина кодової комбінації кода СЛК визначається мірністю n' кубу $E^{n'}$ ЄКФ і складає

$$n_{\text{СЛК}} = 2^{n'} \cdot n_B \text{ [біт]}, \quad (1)$$

де $2^{n'}$ — число вершин n' -мірного кубу $E^{n'}$, n_B — число біт (розрядів) вершин кубу $E^{n'}$ ЄКФ.

Число розрядів вершин кубу E^n відповідає числу змінних, що становлять ці вершини.

Об'єм універсальної матриці перетворень (МП), який відповідає довжині кодової комбінації СЛК (1), складається з $2^{n'}$ стовпців та n_B розрядів відповідно, причому число стовпців МП кон'юнкцій ІДНФ БФ визначається як

$$2^{n'} = 2^{n_B - n}, \quad (2)$$

де n_B — число змінних кон'юнкцій ІДНФ, причому $n_B \geq n_{p \text{ max}}$, $n_{p \text{ max}}$ — число змінних кон'юнкцій ДДНФ БФ.

Універсальна МП кон'юнкцій ІДНФ БФ використовується як при кодуванні, так і декодуванні кодової комбінації СЛК $n_{\text{СЛК}}$, однак об'єм та структура такої матриці залишається незмінною, зрозуміло, що незмінність структури МП при будь-яких

структурно-логічних перетвореннях кон'юнкцій ІДНФ передбачає відповідне програмне забезпечення, яке повинне реалізувати процедури кодування й декодування кодів комбінацій СЛК у єдиних термінах n -мірного куба E^n ЄКФ.

Згідно із загальним методом кодування СЛК при використанні універсальної МП для кожної кон'юнкції ІДНФ, що кодується, здійснюється послідовний запис за $2^{n'}$ тактів у n_B рядків матриці МП значень n_B змінних кон'юнкції $x_0, x_1, \dots, x_{n'-1}$. Рядки матриці МП заповнюються значеннями змінних $x_i = 1$ або $x_i = 0$ по усіх $2^{n'}$ стовпцям, відповідно значенням змінних кон'юнкції ІДНФ, що кодується, заповнюються послідовно за $2^{n'}$ тактів двійковими номерами вершин куба $E^{n'}$, що відповідає досконалому розміщенню геометричних сусідів (ДРГС), причому

$$n'' = n_B - n_v \quad (3)$$

При декодуванні кодів комбінацій СЛК матриця МП необхідна тільки для запису каналного куба $E^{n'}$, деякі вершини якого викривлені помилками. Безпосереднє виправлення помилок змінних ІДНФ здійснюється у блоці визначника змінних (ВЗ).

У ролі визначника змінних є блок мажоритарних елементів, число яких визначається числом рядків n_B матриці МП, і відповідає числу рядків вершин куба $E^{n'}$ ЄКФ. Поріг винесення рішення мажоритарним елементом відносно дійсного значення кожної змінної $x_0, x_1, \dots, x_{n'-1}$ визначається як

$$M \geq 2^{n'-1} + 1 \quad (4)$$

Відповідні змінні у матриці МП щодо змінних кон'юнкції ІДНФ здійснює коректор кон'юнкцій, який виконує послідовний запис значень змінних по рядках матриці МП у режимі кодування, а в режимі корекції змінних кон'юнкції використовує значення змінних $x_0'', x_1'', \dots, x_{n'-1}''$ на виходах мажоритарних елементів відповідного блока.

Якщо здійснювати реалізацію основних функціональних блоків, таких як визначник змінних, коректор кон'юнкцій, на програмному рівні, а за основу МП прийняти регістрові структури, то подібна універсальна кодуючо-декодуюча матриця перетворень кодів СЛК може бути побудована з використанням мікроконтролерів для забезпечення необхідної завадостійкості дискретних даних у сучасних обчислювальних структурах.

Розробка алгоритму роботи універсальної кодуючо-декодуючої матриці перетво-

рень на мікроконтролерах передбачає розпізнавання кон'юнкцій ІДНФ та подальше їх перетворення згідно з основними етапами, розглянутими раніше.

Варто зазначити, оскільки кожна кон'юнкція має унікальний алгоритм відновлення, що універсальна матриця перетворень складається з матриці констант і змінних. Це пов'язано з тим, що число змінних $n_{v \max}$ кон'юнкції досконалої диз'юнктивної нормальної форми (ДДНФ) БФ, яка перетворюється, не може бути більше, ніж число розрядів n_B вершин кубу E^n , тобто $n_B \geq n_{v \max}$.

Унаслідок таких умов частка рядків МП заповнюється змінними кон'юнкції ІДНФ, що перетворюється, а решта рядків — двійковими числами, згідно з досконалим розміщенням геометричних сусідів (ДРГС). Використання ІДНФ БФ при структурно-логічних перетвореннях дає змогу значно зменшити об'єм закодованої інформації, оскільки ранг ІДНФ представлення БФ, що кодується, $r_{\text{ІДНФ}}$, є найменшим серед можливих варіантів представлення, саме мінімальних диз'юнктивних нормальних форм (МДНФ) та ДДНФ.

Апаратна реалізація структурно-логічних кодуючих і декодуючих перетворень кон'юнкцій ІДНФ потребує складних схемотехнічних рішень, що зумовлює виникнення апаратних збоїв, які значно погіршують умови перетворення даних. Досягнення більш ефективних результатів структурно-логічних перетворень можливо при програмно-схемотехнічному підході щодо питань реалізації, завдяки зменшенню апаратних втрат на побудову складних комбінаторних автоматів і використанню розробленого програмного забезпечення, що гарантовано зробить систему перетворення гнучкою та більш надійною.

Таким чином, в узагальненому випадку найбільш вагоме подання в ролі структурно-логічних перетворювачів ІДНФ БФ мають мікроконтролери, оскільки саме вони поєднують зазначені вимоги перетворення.

Однак, незважаючи на те, що мікроконтролери мають досить низьку швидкість у порівнянні з діапазоном сигналів, що лежить у межах десятків ГГц і зараз використовується, сучасні моделі мікроконтролерів дозволяють коректно моделювати функціонування матриці структурно-логічних перетворень, виявляти особливості цього метода й розробляти можливі структурні зміни контролерів для забезпечення сучасних вимог. Основні етапи структурно-логічних перетворень є універсальними і не потребують змін залежно від типу мікроконтролера та його характеристик. Розробку

універсального алгоритму роботи структурно-логічних перетворень будемо здійснювати на прикладі декодувальної матриці перетворень.

В наш час досить широке використання отримали мікроконтролери, що дають можливість використовувати для їхнього програмування мову "С". Це значно полегшує розробку програм, прискорює процес написання, полегшує пошук помилок.

Алгоритм повинен виконувати ініціалізацію кон'юнкцій, відновлення структури дискретних даних, захист від апаратних помилок. Задана ІДНФ має досить просту структуру, тому при визначенні кон'юнкції ефективним буде лінійний алгоритм пошуку, що є оптимальним при роботі із невеликим об'ємом інформації.

Першим етапом згідно з методом структурно-логічних перетворень виступає визначення змінних, що будуть використовуватися в лічильниках та інших операціях, а також і регістрів, що відповідають тим чи іншим пристроям мікроконтролера. Для виконання поставленого завдання основним є визначення портів мікроконтролера, а саме визначення їхнього режиму роботи. В нашому випадку частина портів (їхня кількість може змінюватись залежно від умов) повинна бути визначена в режимі отримання даних, а інша — в режимі передачі даних. Оскільки мікроконтролер — універсальний пристрій, тобто може виконувати різноманітні операції, то для їхньої реалізації слід також визначити й режими роботи вбудованих у нього пристроїв (генераторів, таймерів та ін.). Їхня номенклатура залежить від конкретної моделі, тому це питання вирішується, коли вже обрана конкретна модель контролера. Усі перелічені операції відображені в пунктах 2—4 блок-схеми алгоритму.

Наступною умовою є обов'язкова реалізація циклу, що буде налаштований на режим зацікнення й робота мікроконтролера виконуватиметься впродовж усього часу, доки буде увімкнене живлення. Ця умова відображена в пункті 5 блок-схеми.

Далі переходимо до наступного етапу, а саме ініціалізації кон'юнкцій. В основі процесу визначення лежить алгоритм лінійного перебору, що по чергово порівнює всі змінні відповідно до умови. Тобто, мікроконтролер порівнюватиме рівні вхідних сигналів (логічний "0" та логічна "1") і, коли набір регістрів відповідатиме умові, він почне виконувати наступні операції. Слід зауважити, що у випадку, коли комбінація регістрів на вході не буде відповідати жодній із кон'юнкцій — всі вихідні регістри будуть переведені у стан логічної "1". Ця умова за-

значена у блоці під номером 45. Таку умову можна використовувати як сигналізування про можливий апаратний збій або розгляд синхронізації передавача і приймача.

Розглянемо більш детально послідовність виконання процесу визначення. Отже, у пункті 6 блок-схеми виконується перевірка виконання умови, а саме перевіряється, чи відповідає кодова послідовність вхідних імпульсів першій кон'юнкції ІДНФ. Якщо це ствердження істинне, то наступним є пункт 7, в якому виконується ініціалізація циклу, що буде працювати упродовж 4 тактів, що необхідні для повного відновлення даних. Наступним етапом є перевірка значення змінної лічильника. Така дія дає змогу реалізувати чітку послідовність кодових імпульсів, що відповідають зазначеним вище правилам. Якщо $i=0$, то далі виконуватиметься пункт 9 і на виводах порту В реалізуватиметься послідовність 1000. В іншому випадку буде виконаний пункт 10, в якому буде перевірятися умова $i=1$. У разі виконання цієї умови буде реалізована послідовність 1010. Наступні дії аналогічні пунктам 8—10 із лише одним зауваженням, що кожній ітерації відповідає своя кодова послідовність. Після закінчення роботи циклу програма повернеться до пункту 5 і почнеться наступна перевірка та ініціалізація кодових послідовностей. У випадку, якщо умова пункту 6 не буде виконана, програма перейде до пункту 15 блок-схеми алгоритму, де виконуватиметься перевірка виконання умови $x_3 x_0$. Як і на попередньому етапі, тут є два можливі шляхи: або умова виконується і відбудеться генерація відповідної даній кон'юнкції кодової послідовності (пункти 18—24), або ж програма перейде до перевірки наступної умови. Наступні кроки будуть аналогічними пунктам 6—15.

Особливістю такої роботи контролера є те, що відновлення проходить в автономному режимі, тобто поки мікроконтролер буде проводити генерацію відновленого коду, зміни на вході не будуть враховуватися. Таким чином, головним є перший такт, під час якого виконується ініціалізація, впродовж наступних трьох тактів будь-які зміни вхідних регістрів не призведуть до порушення генерації. Таким чином, виконуються дві умови — захист від апаратних помилок, а також значне зменшення об'єму інформації, що слід передати.

Слід зауважити, що правильна робота можлива лише за умови чіткої синхронізації між передавачем та приймачем. Не виключається можливість використання додаткового синхросигналу.

Блок-схема алгоритму перетворень декодувальної матриці наведена на рис. 1.

Реалізацію алгоритму декодувальної матриці перетворень та написання програми кодування згідно з визначеним методом декодування здійснено для моделі мікроконтролера AVR AT mega 16, який має велику кількість вбудованих та периферійних пристроїв, які забезпечують виконання додаткових функцій. Та найголовніше,

що архітектура серії AT mega розроблялась з урахуванням вимог мови програмування "C".

На основі структури та принципу роботи AT mega 16 була розроблена функціональна схема макету, який реалізує алгоритм структурно-логічного перетворення декодувальної матриці (рис. 2).

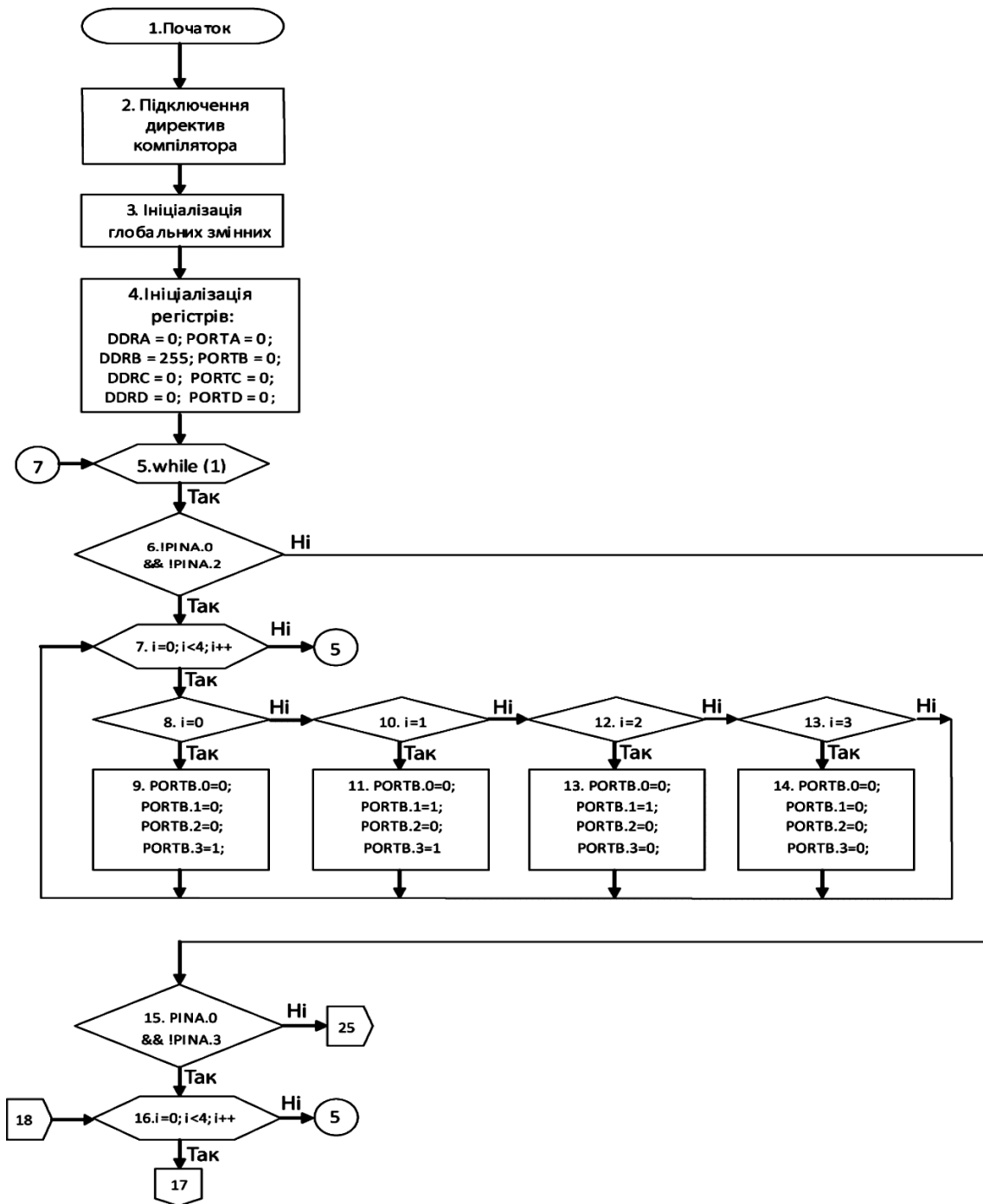


Рис. 1. Алгоритм перетворень декодувальної матриці

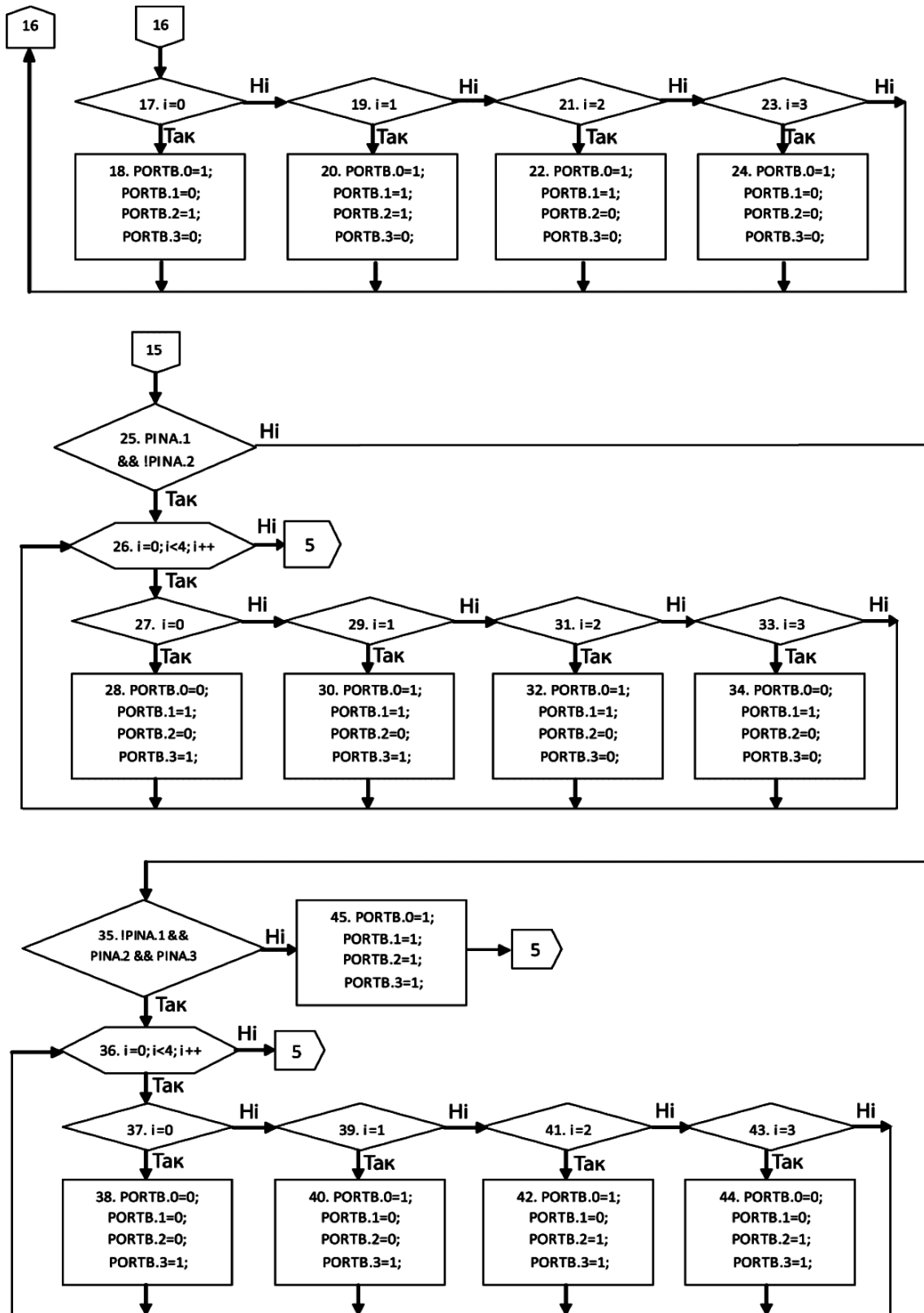


Рис. 1 Алгоритм перетворень декодуючої матриці (продовження)

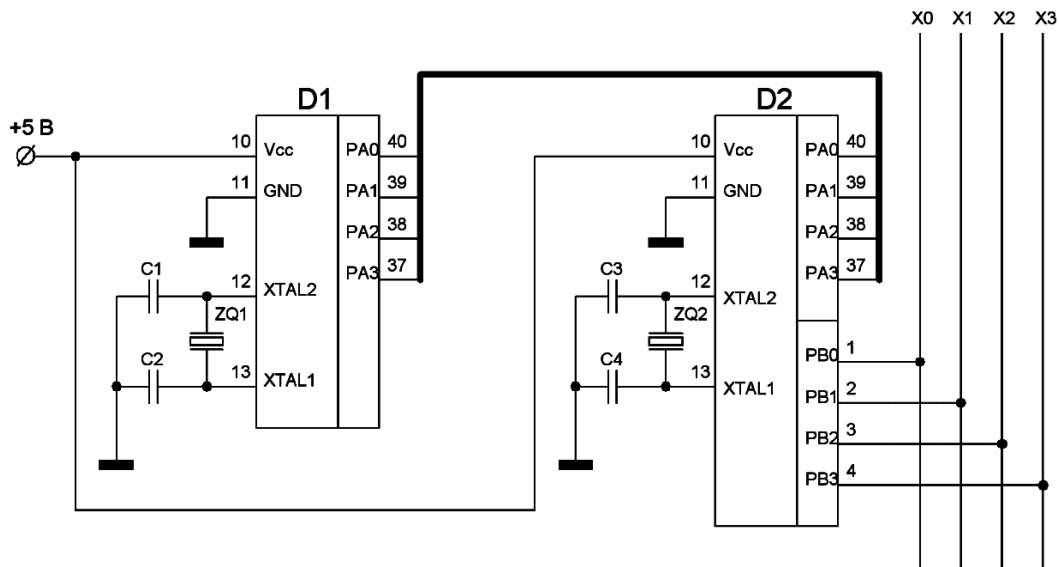


Рис. 2. Схема макету, який реалізує алгоритм перетворень декодувальної матриці де D1 — мікроконтролер, що виконує роль генератора (передавача) кон'юнкцій; D2 — мікроконтролер, що виконує функції декодувальної матриці перетворень; ZQ — тактуючі кварцеві резонатори.

D1 повинен виконувати функції генератора кон'юнкцій, тому, як видно зі схеми, на виводах порту A генеруються сигнали (режим запису), що відповідають заданій ІДНФ. Далі сигнали поступають на виводи порту A мікросхеми D2, що встановлені в режимі читання. Для визначення режиму читання/запису в програмі контролерів слід визначити відповідні регістри таким чином:
 D1: `DDRA = 255;` // переводить 8 виводів порту A в режим запису
 D2: `DDRA = 0.`

Тобто, для визначення режиму роботи порту необхідно DDRx надати необхідне значення: 0 — режим “читання”, 1 — режим “запису”.

Далі слід розробити програму, що буде реалізовувати процес генерації кон'юнкцій, проте не слід забувати про головну особливість розробленого методу: для матриці перетворень головним є перший тактуючий імпульс, наступні 3 такти вона працює в автономному режимі. Тому для реалізації цієї умови скористаємося функцією затримки, що визначається директивою `<delay.h>` та реалізується командами `delay_us (x)` або `delay_ms (x)`. Оскільки в ролі тактуючих генераторів використовуються кварцеві резонатори на 8 МГц, то використовуємо команду `delay_us (x)` де $x=36$ мкс. Повний текст програми наведено в додатку Д.

Порт A мікросхеми D2 в цьому випадку буде в режимі “читання”, а порт B у режимі “запис”. Усі інші особливості та принципи роботи матриці наведені у графічному зобра-

женні алгоритму її роботи, наведеному вище, а повний текст міститься в додатку.

Наступним етапом є компіляція написаних програм і генерація керуючого програмного коду, яким буде “прошитий” контролер. Для цього будемо використовувати CodeVisionAVR. Це безкоштовний компілятор, що генерує досить потужний і компактний програмний код. Його можна завантажити з ресурсу www.hpinfo.tech.ro/cvavr.zip. У своєму складі він має достатньо прикладів та потужну довідку, що дає змогу ознайомитися із особливостями програмування AVR. Також досить докладний опис роботи із цим програмним продуктом можна знайти на ресурсі <http://www.123avr.com>.

Також необхідним програмним продуктом є програма-симулятор VMLAB 3.14. Його можна завантажити з ресурсу <http://www.amctools.com/download.htm>. Ця програма допомагає моделювати роботу мікроконтролера до практичної реалізації, відшукати помилки і виправити їх. Для написання програми для модуляції використовується PSpice-подібна мова. З роботою цього симулятора можна ознайомитися і на ресурсі <http://www.123avr.com>. Найголовніша його особливість — можливість роботи в мультипроцесорному режимі, тобто спостерігати за синхронною роботою двох мікроконтролерів. Тексти програм для виконання симуляції наведені в додатку Д.

Часові діаграми роботи декодувальної матриці перетворень наведені на рис. 3 і 4.

На рис. 3 часова діаграма, що показує час затримок контролера на кожному з виводів.

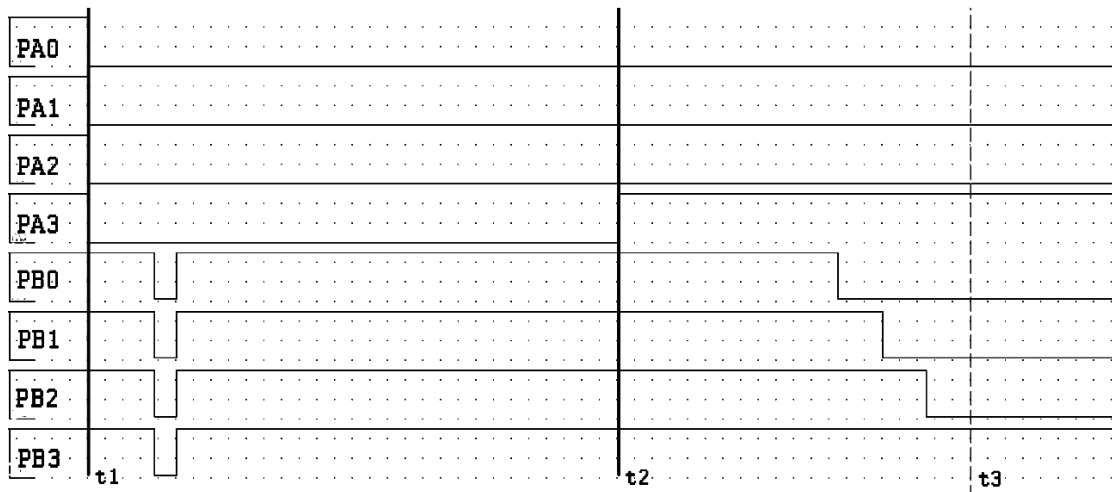


Рис. 3. Часова діаграма затримок контролера

Перша кон'юнкція ІДНФ має вигляд $\bar{x}_2 \bar{x}_0$, отже сигнали, що генеруються мікроконтролером D1, відповідають наступному: PORTA.0=0, PORTA.1=0, PORTA.2=0, PORTA.3=1. Крок координатної сітки цієї діаграми складає 100 нс, тому не важко визначити час затримки, протягом якої мікроконтролер здатний відреагувати на вхідні сигнали. Для виконання цього виміру

скористаємося часом між появою рівня логічної 1 на порті PA3, що відповідає моменту часу t_2 , та відповідною зміною на порті PB3 — час t_3 . Отже, при тактовій частоті 8 МГц затримка становить 2 мкс. Затримка між передніми фронтами сигналів PB складає 250 нс.

На рис. 4 відображено повний цикл генерації усієї ІДНФ.

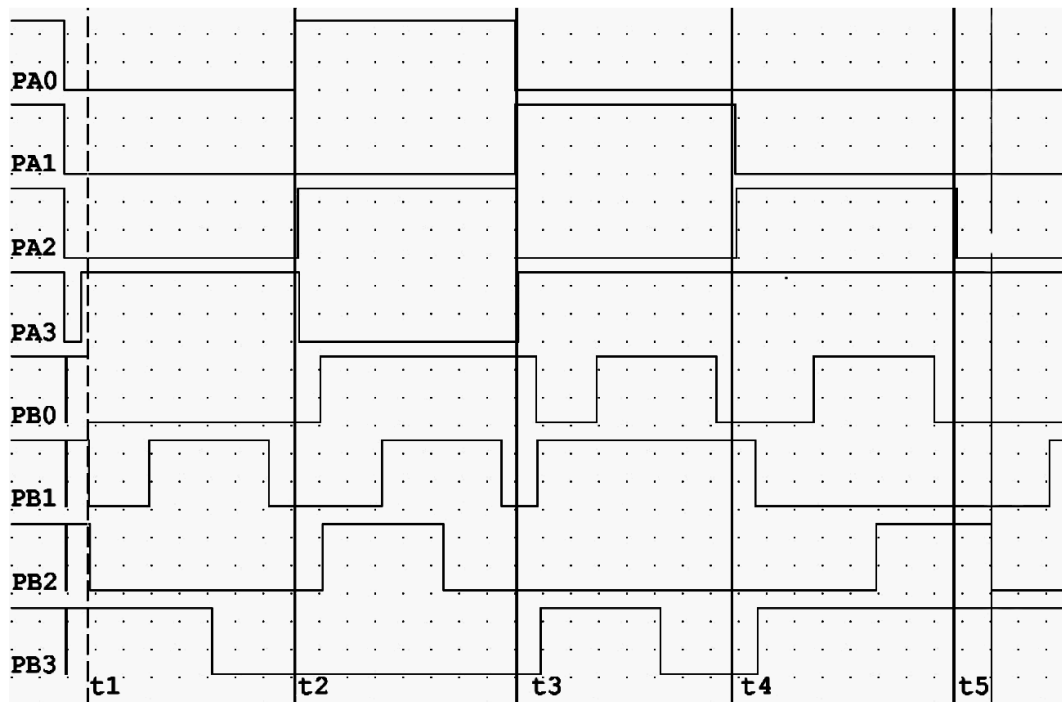


Рис. 4. Повний цикл генерації ІДНФ

Ця часова діаграма відображає усі особливості роботи декодуєної матриці перетворень. У момент часу t_1 починається генерація першої кон'юнкції, що триває до моменту часу t_2 . Як видно з діаграми, отримавши початкову послідовність регістрових

змінних, мікроконтролер починає генерувати послідовність імпульсів, що повністю відповідає правилам, зазначеним у попередніх розділах. Однак, хоча в момент часу t_2 на виводах PA з'являється наступна кодова послідовність, на виводах PB досі ви-

конується генерація першої кон'юнкції, що зумовлено наявністю затримок у роботі контролера. Затримка реагування становить 5 мкс.

У відповідний момент часу генеруються кон'юнкції:

- 1) $t1 - \bar{x}_2 \bar{x}_0$;
- 2) $t2 - \bar{x}_3 x_0$;
- 3) $t3 - \bar{x}_2 x_1$;
- 4) $t4 - x_3 x_2 \bar{x}_1$.

Основні особливості роботи матриці структурно-логічних перетворень можуть бути виявлені у процесі моделювання на стадії проектування, що дасть змогу внести необхідні корективи в разі викриття помилок.

Моделювання роботи мікроконтролера досить зручно робити у програмному продукті VMLAB 3.14, що також допомагає моделювати одночасну роботу одразу двох контролерів. Ця функція є дуже важливою в нашому випадку, оскільки дає змогу проаналізувати коректність роботи матриці, а саме здатність розпізнавати змінні та генерувати відповідні кон'юнкції.

Таким чином, за рахунок використання мікроконтролерів, зокрема серії AT mega, можна значно підвищити ефективність структурно-логічних кодуючих та декодуючих перетворень ІДНФ БФ при сучасному підході до їх реалізації.

Література

1. **Метод** структурно-логічного кодування інфімумних диз'юнктивних нормальних форм булевих функцій в базисі куба / Ю. Д. Іванов, І. В. Пампуха, О. С. Захарова, Г. Б. Жиров // Зб. наук. праць Військового ін-ту Київ. нац. ун-ту ім. Тараса Шевченка. — К., 2006. — № 5. — С. 46—49.
2. **Узагальнений** метод структурно-логічного декодування інфімумних форм подання булевих функцій / Ю. Д. Іванов, І. В. Пампуха, В. О. Осипа, М. М. Охрамович // Зб. наук. праць Військового ін-ту Київського нац. ун-ту ім. Тараса Шевченка. — К., 2006. — № 4. — С. 48—53.
3. **Особливості** корегуючих властивостей структурно-логічних кодів / С. В. Ленков, Ю. Д. Іванов, І. В. Пампуха, К. Ф. Боряк // Наук.-техніч. журнал "Захист інформації". — К., 2007. — № 4(36). — С. 75—81.
4. **Основи** реалізації природної структурно-логічної надмірності диз'юнктивних нормальних форм представлення даних / Ю. Д. Іванов, І. В. Пампуха, Д. О. Перегудов, О. С. Захарова // Вісн. Київ. нац. ун-ту ім. Тараса Шевченка: Військово-спеці. науки. — 2007. — № 14. — С. 12—15.

В статье рассматривается возможность реализации универсальной матрицы структурно-логических превращений для организации единственного кодирующего формата (ЕКФ), кодовой комбинации кода СЛК. Для повышения эффективности структурно-логических кодирующих и декодирующих превращений инфимумных дизъюнктивных нормальных форм (ИДНФ) булевых функций (БФ) был использован микроконтроллер формы Atmel, а именно AVR-RISC Atmel, который позволяет использовать язык "С", значительно упростить сложность прибора, достичь гибкости системы, а также использовать новейшие алгоритмы превращения ИДНФ БФ.

Ключевые слова: универсальная матрица, ИДНФ превращения, кодировки СЛК, ЕКФ, микроконтроллер.

The article considers the possibility of creation of a universal matrix of structural-logical transformations for organizing a single encoding format (ЕКF), the SLK code combination. For the increase of efficiency of structural-logical encoding and decoding transformations of infimum disjunctive normalized forms of (IDNF) of boole functions (BF) the microcontroller of Atmel form was used, namely AVR-RISC Atmel, which allows to use the language "C", to simplify complexity of the device considerably, to attain the system flexibility, and also to use the newest algorithms of transformation of IDNF BF.

Key words: universal matrix, IDNF of transformation, codes of SLK, ЕKF, microcontroller.