

Біляєв С.М.,  
Кіндєревич С.П.,  
Маліновський В.М.,  
Герасименко М.В.,  
Шевчук М.І.

## БЛОК МІКРОПРОГРАМНОГО УПРАВЛІННЯ ДЛЯ RISC ПРОЦЕСОРА НА ПЛІС

Інститут комп'ютерних технологій  
Національного авіаційного університету

*Актуальність проектування процесорів RISC архітектури на програмованих логічних інтегральних схемах обумовлено по-перше класичними перевагами RISC архітектури для реалізації нестандартної апаратури, по-друге підвищенням швидкодії, можливістю реалізації архітектури будь-якої складності за невисокої ціни, широкими можливостями та гнучкістю проектування. Наведений огляд особливостей функціонування блоків мікропрограмного управління, розглянуті переваги застосування програмованих логічних інтегральних схем для реалізації процесорів RISC – архітектури. Розроблений та досліджений блок мікропрограмного управління на програмованих логічних інтегральних схемах*

### **Вступ**

Одним із методів підвищення продуктивності обчислювальних систем є вдосконалення елементної бази, зокрема застосування програмованих логічних інтегральних схем (ПЛІС) для розробки нестандартних вузлів в цифрових ЕОМ [1]. Мікропроцесорні ядра розроблені на ПЛІС, які є дуже перспективними у наш час, відповідають вимогам високої швидкодії і розробляються з метою вирішення спеціалізованих задач та простоти адаптації до об'єктів управління.

За функціональним призначенням розрізняють універсальні і спеціалізовані МП. Універсальні МП мають алгоритмічно універсальний набір команд, за допомогою якого можна здійснювати перетворення інформації відповідно до будь-якого заданого алгоритму. Продуктивність таких процесорів практично не залежить від специфіки розв'язуваних задач [2]. Спеціалізовані МП призначені для рішення обмеженого і строго визначеного кола задач, іноді навіть для рішення однієї конкретної задачі. До спеціалізованих МП належать: сигнальні; медійні та мультимедійні; транспютери; мікроконтролери.

Блок управління є складовою частиною пристрою управління, що входить до складу процесору [3]. Проектування блоку мікропрограмного управління проводилось на програмованих логічних інтегральних схемах. На відміну від звичайних цифрових мікросхем, ПЛІС не містять в собі готової конфігурації від виробника, вони складаються з великої кількості однотипних логічних комірок, які програмуються розробником. Сучасні ПЛІС мають надвисоку ступінь інтеграції, кількість логічних комірок доходить до кількох мільйонів. Окрім того сучасні ПЛІС мають високу швидкість, потужність і низьке енергоспоживання, що дозволяє реалізовувати на них найскладніші пристрої для цифрової обробки сигналів, складні інтелектуальні контролери та нейронні [4].

### **Призначення БМУ з точки зору принципу мікропрограмного управління**

Основне призначення блоку управління – формування всіх управляючих сигналів, які забезпечують виконання кожної команди у ЕОМ. За функціональними можливостями блоки управління поділяються на блоки управління з жорсткою логікою, призначені для реалізації певно-

го набору мікроалгоритмів, реалізовані у вигляді управляючих автоматів, і блоки управління з гнучкою логікою, до яких належать блоки мікропрограмного управління (БМУ). Такі блоки управління дозволяють забезпечити модифікацію та запис нових мікропрограм, змінювати логіку управління в залежності від записаної в пам'яті мікропрограми без зміни конфігурації пристрою [5].

Принцип мікропрограмного управління забезпечує найбільшу гнучкість при організації багатфункціональних мікропроцесорних модулів і шляхом певної комбінації мікрокоманд дає змогу здійснити проблемну орієнтацію мікроЕОМ. Завдяки цьому принципу є можливість використовувати макрооперації в МПС та ефективніше виконувати команди і програми, ніж при використанні підпрограм.

Мікропрограмне управління підвищує гнучкість пристроїв за рахунок можливості зміни мікропрограм, збільшує регулярність структури пристроїв за рахунок широкого використання матричних структур типу пам'яті, забезпечує паралельне розв'язування задач за розосередженого управління та розподіленості пам'яті, підвищує надійність пристроїв за рахунок використання чіпів пам'яті, спрощує контроль функціонування пристроїв, тому що контроль блока мікропрограмного керування зводиться до контролю вмісту запам'ятовуючого пристрою. Певною комбінацією мікрокоманд можна створити набір команд, який максимально відповідатиме призначенню системи, тобто створити технологічну мову. У секційних процесорах набір мікрокоманд можна змінити, використовуючи інші мікросхеми пам'яті мікрокоманд [6].

Переваги мікропрограмного способу управління полягають у тому, що:

- спрощується розробка процесора й збільшується степінь регулярності його структури;
- забезпечується можливість зміни логіки управління без зміни архітектури процесора;

- спеціалізовані та проблемно-орієнтовані процесори реалізуються шляхом модифікації системи команд стандартного універсального процесора, без зміни архітектури;

- підвищується ефективність системного та прикладного програмного забезпечення завдяки мікропрограмній реалізації часто використовуваних мікропрограм і окремих стандартних функцій.

Принцип функціонування БМУ відповідає принципу мікропрограмного управління архітектури фон-Неймана або, як її ще називають, принстонській архітектурі.

Блоки управління, як складові процесорів, можуть бути реалізовані в різних архітектурах процесорних ядер – *RISC*- та *CISC*-архітектурах.

*RISC*-процесори (*Reduced Instruction Set Computer*) – реалізують обчислення зі скороченим набором команд. Архітектура процесорів, побудована на основі скороченого набору команд характеризується наявністю команд фіксованої довжини, більшої кількості регістрів, операцій типу регістр-регістр, а також відсутністю непрямої адресації. Концепція *RISC* розроблена Джоном Коком (*John Cocke*) із *IBM Research*, назва придумана Девідом Паттерсоном (*David Patterson*) [6]. Серед перших реалізацій цієї архітектури були процесори *MIPS*, *PowerPC*, *SPARC*, *Alpha*, *PA-RISC*. В мобільних пристроях широко використовуються *ARM*-процесори.

*CISC*-процесори (*Complex Instruction Set Computing*) – відповідають концепції проектування процесорів, яка характеризується наступним набором властивостей:

- нефіксоване значення довжини команди;
- арифметичні дії кодуються в одній інструкції;
- невелике числом регістрів, кожний з яких виконує визначену функцію.

Типовими представниками є процесори на основі x86 системи команд (крім сучасних *Intel Pentium 4*, *Pentium D*, *Core*, *AMD Athlon*, *Phenom*, які є гібридними) та процесори *Motorola MC680x0* [7].

*RISC*-процесори швидші і більш економічні ніж *CISC*-процесори. Одним із найперспективніших напрямків розвитку сучасної цифрової техніки є розробка процесорів *RISC* архітектури на ПЛІС. З одного боку це обумовлено класичними перевагами *RISC* архітектури для реалізації високопродуктивних спеціалізованих пристроїв для рішення нестандартних задач. З іншого боку застосування сучасних ПЛІС обумовлює високу швидкодію, реалізацію архітектури любой складності за невисокої ціни та затрат часу, широкі можливості та гнучкість проектування.

### **Позитивні моменти застосування ПЛІС для побудови мікропроцесорів**

ПЛІС являє собою кристал, на якому розташована велика кількість простих логічних елементів. Спочатку ці елементи не сполучені між собою. З'єднання елементів – перетворення розрізнених елементів в електричну схему, здійснюється за допомогою електронних ключів, розташованих в цьому ж кристалі. Електронні ключі управляються спеціальною пам'яттю, у комірці якої заноситься код конфігурації цифрової схеми. Таким чином, записавши в пам'ять ПЛІС певні коди, можна зібрати цифровий пристрій будь-якої міри складності, що залежить від кількості елементів на кристалі і параметрів ПЛІС. На відміну від мікропроцесорів, в ПЛІС можна організувати алгоритми цифрової обробки на апаратному рівні. При цьому швидкодія цифрової обробки різко зростає.

На відміну від звичайних цифрових мікросхем, ПЛІС не містять в собі готової конфігурації від виробника, вони складаються з великої кількості однотипних логічних комірок, які програмуються розробником. Для програмування використовуються програматори і середовища для налагодження, які дозволяють задати бажану структуру цифрового пристрою у вигляді принципової електричної схеми чи програми на спеціальних мовах описання апаратури *Verilog*, *VHDL*, *AHDL* та інші [7].

Сучасні ПЛІС мають надвисоку степінь інтеграції, кількість логічних комірок доходить до кількох мільйонів. Окрім того сучасні ПЛІС мають високу швидкодію, потужність і низьке енергоспоживання, що дозволяє реалізовувати на них найскладніші пристрої для цифрової обробки сигналів, складні інтелектуальні контролери та нейрочипи.

Достоїнствами технології проектування пристроїв на основі ПЛІС є: мінімальний час розробки схеми (потрібно лише занести в пам'ять ПЛІС конфігураційний код); на відміну від звичайних елементів цифрової схемотехніки тут відпадає необхідність в розробці і виготовленні складних друкованих плат; швидке перетворення однієї конфігурації цифрової схеми в іншу (заміна коду конфігурації схеми в пам'яті); для створення пристроїв на основі ПЛІС не потрібне складне технологічне виробництво [4].

Альтернативою ПЛІС є: базові матричні кристали, що вимагають заводського виробничого процесу для програмування; *ASIC* – спеціалізовані замовлені великі інтегральні схеми, які при дрібносерійному і одиничному виробництві істотно дорожчі; спеціалізовані комп'ютери, процесори, наприклад, цифровий сигнальний процесор, або мікроконтролери, які через програмного способу реалізації алгоритмів повільніше ПЛІС.

### **Розробка БМУ**

Для безпосередньої реалізації пристрою застосовувалась мікросхема *CycloneII EP2C35F672C6* фірми *Altera*.

До складу розробленого БМУ входять пам'ять мікрокоманд (ПМК), регістр адреси мікрокоманд (РАМК), регістр мікрокоманд (РМК), дешифратор мікрооперацій (ДшМО). В ПМК зберігаються управляючі програми, які інтерпретуються БУ, в результаті чого на виході регістра мікрокоманд формуються управляючі сигнали, які по шині управління надходять на всі блоки ЕОМ.

Кожній команді, яка записана у основній пам'яті ЕОМ, відповідає мікропро-

грама, що зберігається в пам'яті блоку мікропрограмного управління [3].

БМУ функціонує у відповідності з принципом мікропрограмного управління, що полягає в наступному: під час виконання мікропрограми в кожному такті із постійної пам'яті БМУ зчитується та розшифровується чергова мікрокоманда. В результаті виконання мікрокоманди формуються управляючі сигнали необхідної тривалості, що поступають на всі функціональні частини обчислювальної системи, а також формується адреса наступної мікрокоманди.

Спосіб формування адреси мікрокоманди визначається управляючими сигналами на входах інформаційного слова мікроінструкції переходу  $I3 - I0$  та вході умови. Внутрішні управляючі сигнали

виконують настройку блоків БМУ на виконання необхідних функцій. Мікроінструкція переходу використовується для управління комутацією на вихід адреси нової мікрокоманди відповідних джерел формування адреси.

Так, наприклад, під час виконання відповідних мікроінструкцій ( $I3 - I0$ ) вміст РА/ЛЦ (регістр адреси / лічильник циклів) зменшується на одиницю за додатним перепадом  $CLK$ , що дозволяє використовувати РА/ЛЦ в якості лічильника циклів. Таким чином, у РА/ЛЦ надходить інформація з зовнішніх входів і у подальшому використовується або в якості адреси наступної мікрокоманди, або як кількість повторів циклів.

Часова діаграма роботи БМУ зображена на рис. 1.

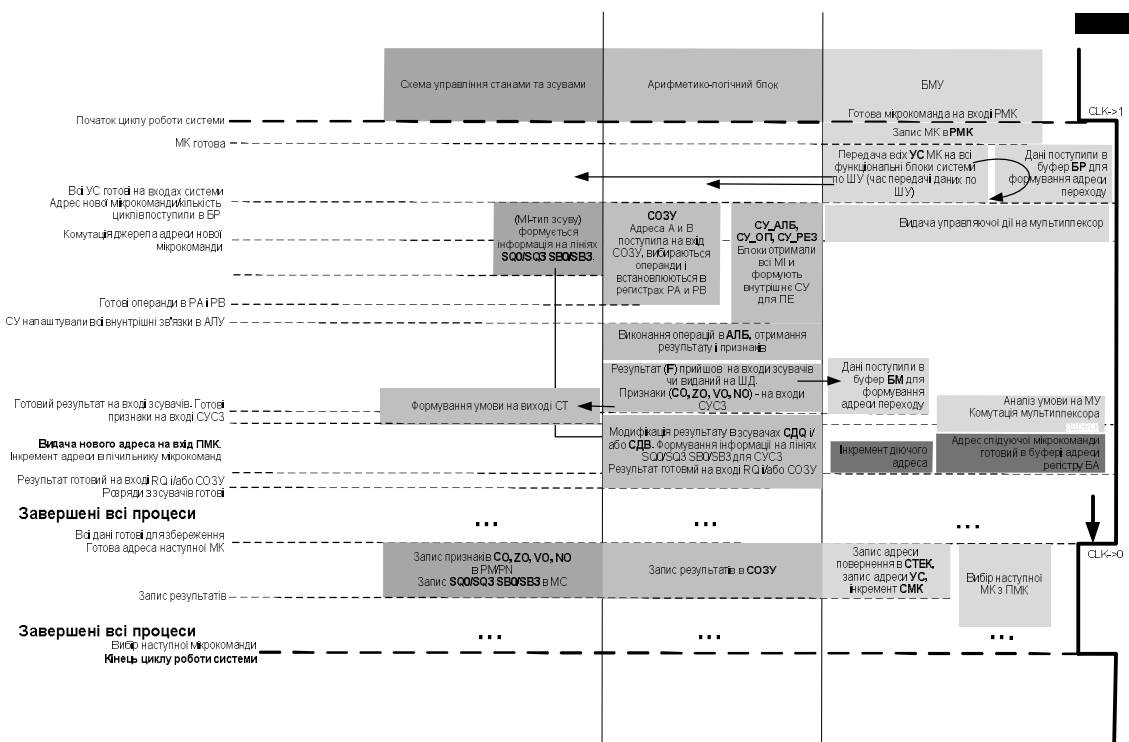


Рис. 1. Часова діаграма роботи БМУ

**Принцип роботи СТЕКУ**

Стек складається з показчика стеку (ПС) і накопичувача (СТЕК), що має п'ять регістрів (глибина стеку дорівнює п'яти). Схема ПС є реверсивним лічильником, який управляється внутрішніми сигналами і змінює свій стан за додатним перепадом  $CLK$ . Стек організований за принципом *LI/FO* (*Last Input / First Output*

– останній прийшов / перший обслугований). Схема ПС вказує на регістр, у який за останнім зверненням здійснювався запис даних.

Залежно від виконуваних мікроінструкцій ( $I3 - I0$ ) стек працює в наступних режимах:

– очищення стека – встановлення показчика стеку в нуль ( $ПС=0$ );



реалізована на вбудованих у мікросхему блоках пам'яті, блоку формування внутрішніх сигналів управління, лічильника мікрокоманд, стеку, мультиплектора вибору наступної адреси мікрокоманди. Досягнута адаптація до об'єктів управління, яка реалізована за рахунок перепрочи-

вання пам'яті мікропрограм, що дозволить змінити логіку управління процесора без будь яких змін в архітектурі пристрою. В статті наведена схема та діаграма роботи Стеку як складової частини БМУ.

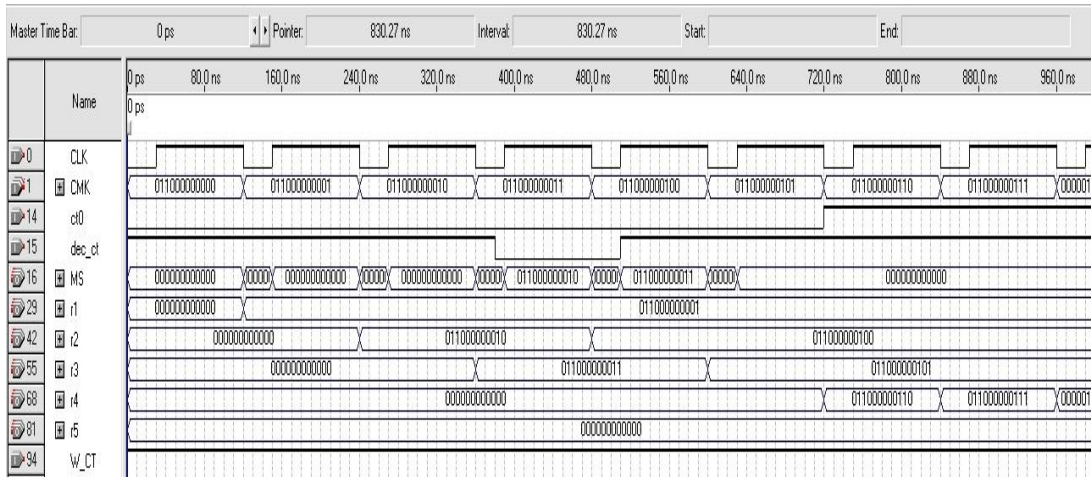


Рис. 3. Часова діаграма роботи стеку

Для синтезу і моделювання БМУ застосована САПР *QUARTUS II Altera*. Розробка виконана на мові опису апаратури інтегральних схем *VHDL*. На побудову стеку затрачено лише 86 логічних елементів, що становить менше одного відсотка від всієї ємності мікросхеми *FPGA Cyclone II* класу *EP2C35*. Після отримання та аналізу часових характеристик рис. 3. та рис. 2. відповідно і з рис. 2. видно що 90 нс забагато для формування наступної мікрокоманди, ми маємо можливість скоротити час активного стану. В порівнянні з характеристиками відомого пристрою *Am2911* фірми *Advanced Micro Devices* [8], ми маємо вииграш в часі під час формування наступної мікрокоманди.

### Список літератури

1. Клименко І.А., Біляєв С.М., Пономарчук Д.С. Процесорне ядро для обчислювальної системи на ПЛІС // «Прикладна математика та комп'ютеринг ПМК-2010»: друга наук. конф. магістрів та аспірантів присвячена 20-річчю факультету прикладної математики, 14-16 квіт. 2010 р.: Зб. тез доп. – К., 2010. – С.197–201.

2. Якименко Ю.І. Мікропроцесорна техніка. – К.: ІВЦ "Видавництво «Політехніка», "Кондор", 2004. – 440 с.

3. Жабін В.І., Жуков І.А., Клименко І.А., Стіренко С.Г. Арифметичні та управляючі пристрої цифрових ЕОМ: навч. посіб. – К.: ВЕК+, 2008. – 176 с.

4. Соловьев В.В. Проектирование цифровых систем на основе программируемых логических интегральных схем. – М.: 2007. – 636 с.

5. Угрюмов Е.П. Цифровая схемотехника. – СПб.: БХВ-Петербург, 2004. – 528 с.

6. Грушницкий Р.И. Проектирование систем с использованием микросхем программируемой логики – СПб.: БХВ-Петербург, 2002. – 608 с.

7. Библио П.Н. Основы языка VHDL. изд. 3. – М.: ЛКИ, 2007. – 328 с.

8. Мик Дж., Брик Дж. Проектирование микропроцессорных устройств с разрядно-модульной организацией: Пер с англ. – М.: Мир, 1984. – Кн. 1. – 253 с.

Подано до редакції 18.05.10