УДК 656.13:681.3

Л.І. НЕФЬОДОВ, д.т.н., проф., зав. каф. ХНАДУ (м. Харків),
С.А. КРИВЕНКО, к.т.н., доц. ХНАДУ (м. Харків),
Є.М. МУСІЄНКО, аспірант ХНАДУ (м. Харків)

МЕТОД СИНТЕЗУ МОДЕЛІ ГЕОІНФОРМАЦІЙНОЇ СИСТЕМИ НА ОСНОВІ БАЗОВИХ ПОЛІНОМІВ

Надано новий метод для верифікації виконання розкладання поліному певного ступеня $P(x)$ на множники над полем Галуа GF(2). За основу прийнята модель корпорації Altera, яка застосовується в пакеті автоматизованого проектування мікросхем QuartusII(2009), аналогічну модель застосовує корпорація Xilinx та інші. Новизна нашого підходу полягає в тому, що на першому етапі застосовується часове моделювання в двох ієрархічних рівнях і передбачаються спеціальні заходи для виключення блокувань, а на прикінцевому етапі виконане натурне моделювання на реальній надсучасній мікросхемі, яка розвиває на три порядку більшу швидкість обчислення щодо певних поліномів інтервалу. Іл.: 6. .Бібліогр.: 11 назв.

Ключові слова: поле Галуа, GF(2), Altera, QuartusII, Xilinx.

Постановка проблеми. Сучасні глобальні інформаційні інфраструктури широко застосовують базові поліноми. Система супутникової навігації GPS [1] застосовує базові поліноми: $x^{10}+x^9+x^8+x^6+x^3+x^2+x^1$; $x^{12}+x^9+x^8+x^4+x^3+x^2+1$; $x^{12}+x^{11}+x^{10}+x^9+x^8+x^5+x^2+x+1$; $x^{12}+x^{11}+x^{10}+x^9+x^8+x^7+x^6+x^5+x^4+x^3+x^2+1$; $x^{12}+x^{11}+x^8+x^6+1$; $x^{10}+x^3+1$. Глобальна система для мобільного зв'язку GSM [2] застосовує базові поліноми: $x^6+x^5+x^3+x^2+1$; x^6+x^4+x+1; $x^6+x^4+x^3+x^2+1$ та багато інших. Універсальна система для мобільного зв'язку UMTS [3] застосовує базові поліноми: $x^8+x^6+x^5+x^4+1$; $x^9+x^8+x^7+x^6+x^4+x^2+x+1$ та багато інших. Загалом дуже велика кількість геоінформаційних систем (ГІС) [4] застосовує згадані поліноми (наприклад, EGNOS, GAGAN, GLONASS, GNSS, MSAS, NA-ESRD, NA-ESRK, QZSS, SBAS, WAAS). Актуальною проблемою є науково-обгрунтований вибір поліномів, який є ефективним за швидкодією та конкурентноздатним за складністю [5].

Аналіз літератури. Проблема факторингу одновимірного багаточленного поліному $P(x)$ над обмеженим полем F часто виникає в обчислювальній алгебрі [5]. Важливий випадок є тоді, коли F має маленьку розмірність і $P(x)$ має дуже високий але розкиданий ступінь, тобто це поліном $P(x)$ має тільки маленьку кількість відмінних від нуля складових. Щоб спростити роз'яснення, ми обмежуємо увагу щодо випадку, де F поле Галуа GF(2) і $P(x)$ – трином $P(x)= x^r + x^s + 1$, $r > s > 0$, хоча наведені нижче ідеї звертаються більш загалом і можуть бути узагальнені і корисні для факторингу розкиданих многочленів над полями маленької розмірності. Наша мета – надати метод з добре обгрунтованою складністю. З тих пір, як ми обмежуємо увагу до тричленів, ми формуємо середню величину над всіма тричленами фіксованого ступеня r. Наша спонука – розвивати швидкість попередніх методів для пошуку тричленів вищого ступеня, що не піддаються перетворенню [6]. Для наданого

ступеня r, ми хочемо знаходити всі триноми $x^r + x^s + 1$, що не піддаються перетворенню. У відомих прикладах значення r – це обов'язково експонента Мерсена (Mersenne), тобто, $2^r - 1$ є просте число Мерсена. В такому випадку трином ступеня r, що не піддається перетворенню, обов'язково примітивний. В даній роботі розроблений метод, без обмеження до представників чисел Мерсена, у цьому випадку потрібен розклад поліномів ступеня від 1 до $2^r - 1$ на множники для того, щоб перевірити примітивність (подивіться наприклад [7]). Часто розглядають експоненту Мерсена $r = \pm 1$ за модулем 8, тому що в іншому випадку теорема Свена (Swan) [7] виключає тричлени, що не піддаються перетворенню ступеня r (за винятком $s = 2$ або $s = r - 2$, але ці випадки звичайно легкі для управління: наприклад якщо $r = 13466917$ або 20996011 ми маємо $r = 1$ за модулем 3, так $x^r + x^2 + 1$ ділимо $x^2 + x + 1$). Прості числа Мерсена можуть бути знайдені на сайті великого пошуку в Інтернеті простих чисел Мерсена (Great Internet Mersenne Prime Search-GIMPS) [8]. На час написання, сім з восьми найбільших відомих чисел Мерсена задовольняють умові $r = 1$ за модулем 8: $r = 6972593$; 24036583; 25964951; 30402457; 32582657; 37156667; 42643,801 і 43112609. У випадку найменшого $r = 6972593$ примітивний тричлен був знайдений на основі використання ефективного виконання наївного алгоритму (Brent, Larvala і Zimmermann) [6]. Цей алгоритм не дозволяє розглянути більші числа Мерсена, тому що алгоритм має для обчислювальної складності грубо оцінку порядку r^3 і для наступного випадку $r = 24036583$ необхідно приблизно в 41 раз більше часу, ніж для $r = 6972593$. Запропонований [9] новий швидкий алгоритм (Brent і Zimmermann) дозволяє знайти два примітивні тричлени порядку $r = 24036583$ в менший час, ніж наївний алгоритм для $r = 6972593$. Прискорення методу над наївним алгоритмом для $r = 24036583$ має коефіцієнт 560. Таким чином, задача пошуку базових поліномів залишається актуальною. Крім того, необхідно зняти обмеження для порядку поліному експонентою Мерсена. Сучасні пакети автоматизованого проектування мікросхем [10] дозволяють зняти це обмеження. Вони мають у своєму складі дві моделі функціональної та часової верифікації систем, причому функціональне моделювання здійснюється із значно більшою швидкістю, натурне моделювання має ще більшу швидкість. Якщо розглядати натурні моделі, то тут існує два обмеження: час і розмір мікросхеми. Час обмежимо одним тижнем – це приблизно 1000000 секунд. При цьому число повторень псевдовипадкової послідовності на тактовій частоті 1 ГГц буде складати 1000000000 млн. і для цього знадобиться приблизно 50 тригерів, що ніяк не відповідає можливостям сучасних мікросхем.

Метою даної роботи є метод підвищення швидкодії моделі ГІС, за рахунок розробки методу синтезу моделі ГІС на основі базових поліномів шляхом зменшення обчислювальної складності. Для досягнення цієї мети можна застосувати відомий метод управління проектом та розв'язати наступні вісім задач [11]: (1) за допомогою програмного забезпечення

MATLAB/Simulink синтезувати модель елементу ГІС системи на основі штатних блоків пакету Simulink і призначених для користувача блоків, наприклад, блоків Altera DSP Builder; (2) застосувати спеціалізований блок, наприклад, блок SignalCompiler, для синтезу і аналізу проекту; (3) імітувати роботу виробу на основі моделі в пакеті Simulink, наприклад, виконати аналіз сигналів моделі за допомогою осцилографа; (4) запустити програму SignalCompiler і встановити параметри імітації і синтезу, наприклад, за допомогою засобу RTL корпорації Altera; (5) виконати RTL моделювання; (6) використовувати вихідні файли, вироблювані блоком SignalCompiler засобів Altera DSP Builder для виконання синтезу засобами RTL; (7) компілювати проект системи в середовищі Quartus II; (8) завантажити програмну інформацію проекту в макет системи і провести комплексні випробування апаратного і програмного забезпечення системи.

Метод синтезу моделі ГІС на основі базових поліномів. На жаль, швидкодія програмного забезпечення MATLAB/Simulink сьогодні не дозволяє розв'язати першу задачу відомими методами і синтезувати моделі елементів ГІС на основі базових поліномів великого порядку. Тому необхідно розробити нову модель в вигляді ієрархічного проекту який має два рівні, на верхньому рівні застосовано один модуль з ім'ям p37toplevel01 на нижньому рівні чотири модулі з іменами: p37prn; p37controller; p37counter; p37prn2.

Виводи модуля: вхід тактового сигналу, що синхронізує роботу модуля (CLKin); вхід запуску (StartTop); вхід скидання (Resetin); вхідна шина для вектору ініціалізації (A[width..1]); вихідна шина стану генератора псевдовипадкової послідовності (Gstate[Width..1]); вихід елемента XOR (trixor); вихід елемента *r* полінома (widthxorout); вихід елемента *s* полінома (saxorout); вихід квітування результату верифікації (Done); вихідна шина автомата з обмеженим числом станів (E[4..1]); вихідна шина лічильника тактів (Tact[Width..0]); вихідна шина поточного значення вектора ініціалізації (Ini[Width..1]); вихід елемента XOR (trixor), вихід елемента *r* полінома (2widthxorout); вихід елемента *s* полінома (2saxorout); вихід квітування результату верифікації (2Done). Процедура сертифікації починається з появою логічної одиниці на вході запуску (Start). Тривалість активного рівня сигналу – один період тактового сигналу CLK. Значення вхідних чисел залишається незмінним з моменту появи логічної одиниці на вході Start до закінчення процедури. При Done = 1 на виході Gstate[] має бути відповідний вектор ініціалізації.

Відзначимо, що модуль (P37toplevel01) є модулем верхнього рівня в ієрархії описів. До складу модуля верифікації входять: модуль підготовки даних (p37prn2); генератор на основі поліному (P37prn); лічильник тактів (P37Counter); блок управління роботою модуля сертифікації (P37Controller). Модель реалізована на базі модулів з параметрами *r*, *s* (Width,Sa), які можна змінювати без обмежень. Процедура синтезу складного модуля верхнього рівня розділена на етапи. Спочатку створені окремі модулі і перевірена їхня

робота. Потім виконана компіляція наведеної вище схеми за допомогою пакету автоматизованого проектування цифрових мікросхем MAX+PlusII і перевірена робота моделі в цілому.

Спочатку отримаємо результат на прикладі двох поліномів $1+x+x^4$, $1+x+x^2+x^3+x^4$. Перший поліном хоча і є триномом для якого $r = 4$, $s = 1$, але $r = 4$ не є експонентою Мерсена, тому для перевірки, чи є поліном базовим необхідно виконати ділення поліномів за модулем два 15 разів:

$(x^{15}+1)/(x^4+x+1) = x^{11}+x^8+x^7+x^5+x^3+x^2+x+1$;
$(x^{14}+1)/(x^4+x+1) = x^{10}+x^7+x^6+x^4+x^2+x+1$ залишок x^3;
$(x^{13}+1)/(x^4+x+1) = x^9+x^6+x^5+x^3+x+1$ залишок x^3+x^2;
$(x^{12}+1)/(x^4+x+1) = x^8+x^5+x^4+x^2+1$ залишок x^3+x^2;
$(x^{11}+1)/(x^4+x+1) = x^7+x^7+x^3+x$ залишок x^3+x^2+x+1;
$(x^{10}+1)/(x^4+x+1) = x^6+x^3+x^2+1$ залишок x^2+x;
$(x^9+1)/(x^4+x+1) = x^5+x^2+1$ залишок x^3+x+1;
$(x^8+1)/(x^4+x+1) = x^4+x$ залишок $x+1$;
$(x^7+1)/(x^4+x+1) = x^3+1$ залишок x^3+x;
$(x^6+1)/(x^4+x+1) = x^2$ залишок x^3+x^2+1;
$(x^5+1)/(x^4+x+1) = x$ залишок x^2+x+1;
$(x^4+1)/(x^4+x+1) = 1$ залишок x;
$(x^4+x+1)/(x^3+1) = x$ залишок 1;
$(x^4+x+1)/(x^2+1) = x^2+1$ залишок x;
$(x^4+x+1)/(x+1) = x^3+x^2+x$ залишок 1.

Тому можна зробити висновок, що поліном $1+x+x^4$ є базовим. Аналогічно для поліному $1+x+x^2+x^3+x^4$ можна зробити висновок, що він не є базовим. Для цього необхідно знову виконати ділення поліномів за модулем два:

$(x^{15}+1)/(x^4+x^3+x^2+x+1) = x^{11}+x^{10}+x^6+x^5+x+1$;
$(x^{14}+1)/(x^4+x^3+x^2+x+1) = x^{10}+x^9+x^5+x^4$ залишок x^3+x^2+x;
$(x^{13}+1)/(x^4+x^3+x^2+x+1) = x^9+x^8+x^4+x^3$ залишок x^3+1;
$(x^{12}+1)/(x^4+x^3+x^2+x+1) = x^8+x^7+x^3+x^2$ залишок x^2+1;
$(x^{11}+1)/(x^4+x^3+x^2+x+1) = x^7+x^6+x^2$ залишок $x2+1$;
$(x^{10}+1)/(x^4+x^3+x^2+x+1) = x^6+x^5+x+1$.

Тобто на поліном $x^4+x^3+x^2+x+1$ діляться без залишку поліноми $x^{15}+1$ та $x^{10}+1$, а може і інші поліноми більше низького порядку. Цей приклад підтверджує, що верифікація поліному не є простою процедурою, так для перевірки поліному $x^{33}+x^{20}+1$ необхідно виконати ділення поліномів ступеня не вище $r = 8589934591$ приблизно 10 мільйонів разів.

Розглянемо граф переходів автомата з обмеженим числом станів (P37Controller), який виконає сертифікацію поліному $x^{33}+x^{20}+1$ в складі запропонованої системи. Умовне графічне зображення автомата, що управляє роботою модуля, наведене на рис. 1.

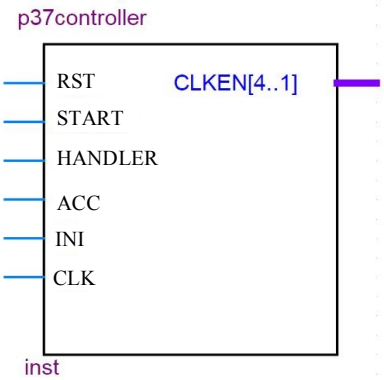

Рис. 1. Умовне графічне зображення автомата, що управляє роботою модуля

Автомат має виводи: CLK – вхід синхронізації; RST – вхід асинхронного скидання автомата на початковий стан; START – вхід запуску автомата; HANDLER – вхід ручного управління; ACC – вхід індикації збігу вектору ініціалізації; INI – вхід індикації наступного вектора ініціалізації; CLKEN[4] – вихідний сигнал дозволу ініціалізації генератора; CLKEN [3] – вихідний сигнал дозволу на зміну стану генератора; CLKEN [2] – вихідний сигнал дозволу підготовки даних; CLKEN [1] – вихідний сигнал, що дозволяє запис нового вектора ініціалізації. Логіку роботи автомата задає граф переходів, наведений на рис. 2.

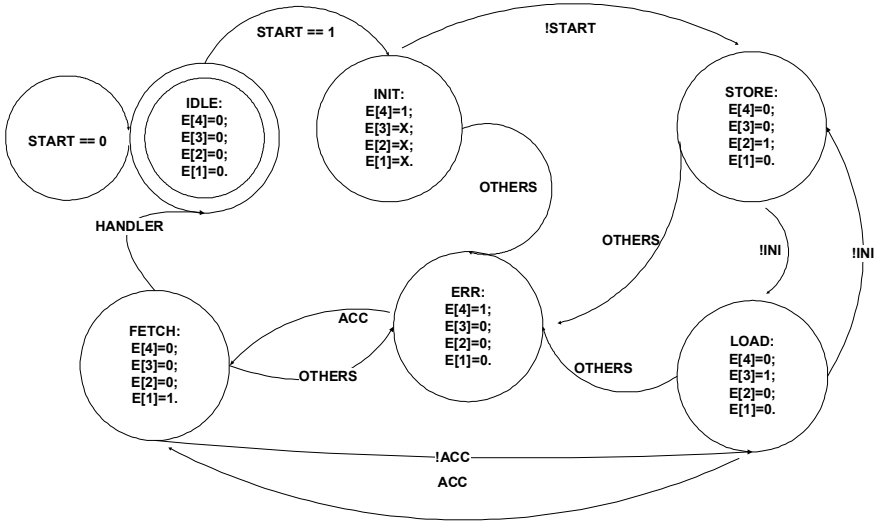

Рис. 2. Граф автомату, який має усього 6 станів

Перелік станів автомата. Стан IDLE. Стан очікування (вихідний стан автомата).

Стан Fetch. Здійснюється підготовка вектора ініціалізації.

Стан Init. В тригери генератора записуються розряди вектора ініціалізації.

Стан Store. Зберігаються результати розрахунку для визначення наступного стану генератору.

Стан Load. Виконується зміна стану генератора.

Стан ERR. Стан помилки.

Модуль автомату синтезований засобами пакета QuartusII з використанням для його опису операторів Case, IF THEN.

Результати досліджень. Результати отримані шляхом розв'язання задач (4) – (8). Хоча цей шлях і є більш довгим у порівнянні з відомим [11], але розв'язати задачі (2), (3) на сьогодні не можливо, тобто не можливо застосувати спеціалізований блок, наприклад, блок SignalCompiler, для синтезу і аналізу проекту та імітувати роботу виробу на основі моделі в пакеті Simulink. Результати розв'язання задачі (4) наведені у відповідному звіті корпорації Altera, де наведені параметри імітації і синтезу. Результати розв'язання задачі (5) наведені на рис. 3.

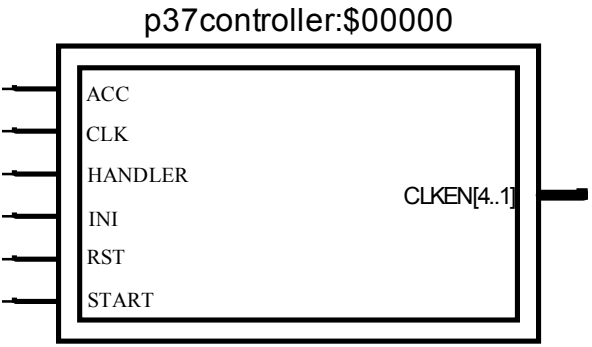

Рис. 3. Результати RTL моделювання модуля верхнього рівня

Цей результат отримано шляхом виконання RTL моделювання в пакеті QuartusII. Тут наведено лише один модуль, автомат з обмеженою кількістю станів, з чотирьох модулів проекту мікросхеми. Видно, що проект виконано коректно, принаймні співпадають назви входів та виходів (див. рис. 1).

Результати розв'язання задачі (6) наведені на рис. 4.

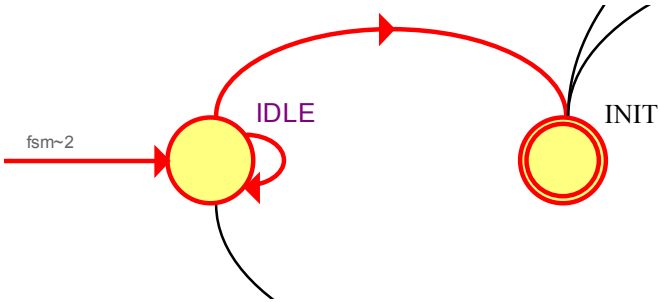

Рис. 4. Результати RTL моделювання цифрового автомату

Виконання синтезу графу переходів виконано засобами RTL. Тут наведено перехід зі стану очікування IDLE до стану INIT запису в тригери генератора розрядів вектора ініціалізації (див. рис. 2).

Для розв'язання сьомої задачі виконана компіляція проекту системи в середовищі QuartusII 9.1. Виконано функціональне та часове моделювання кожного з модулів.

На рис. 5 наведений фрагмент результатів моделювання генератора, який побудований на основі полінома четвертого порядку $r = 4$.

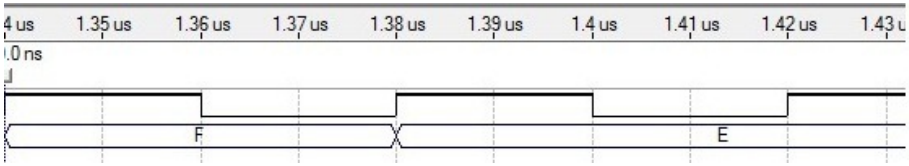

Рис. 5. Часова діаграма для вектора ініціалізації F

На часовій діаграмі видно два стани з 6 станів генератору F, E, C, 9, 3, 7 які він проходить. Загальна кількість станів 6 менша за максимально можливу 15 для $r = 4$. Потім генератор послідовно проходить стани 1, 2, 5, A, 4, 8. І нарешті генератор послідовно проходить стани 6, D, B, як це наведено на рис. 6.

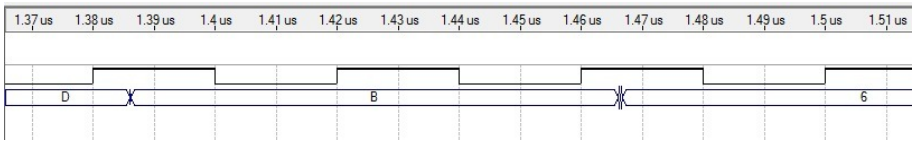

Рис. 6. Часова діаграма для псевдовипадкової послідовності

Загальна кількість станів 3 менша за максимально можливу 15 для $r = 4$, але три варіанти моделювання охоплюють всі 15 можливі стани генератора.

Цей генератор синтезовано на основі базового поліному $1+x+x^4$, тому він проходить всі можливі стани від 1 до F та формує псевдовипадкову послідовність. Невеликий порядок поліному $r = 4$ обрано для демонстрації принципу дії запропонованого методу. Для отримання кількісних характеристик методу виконана сертифікація базового поліному $1+x^{20}+x^{33}$, для цього розв'язана восьма задача, тобто завантажена програмна інформація проекту в макет системи і проведені комплексні випробування апаратного і програмного забезпечення системи. Виконано часове моделювання системи, яка синтезована в пакеті автоматизованого проектування QuartusII для мікросхеми сім'ї StratixIII. Часове моделювання за допомогою запропонованого методу дало виграш приблизно в 1000 разів. Моделювання виконано на комп'ютері наступної конфігурації: процесор – Intel®Celeron®CPU 530@1,73 GHz; пам'ять – 1,00 ГБ; 32-розрядна операційна система Windows Vista.

Висновки. Таким чином, розроблено метод синтезу моделі ГІС на основі базових поліномів. За основу прийнята модель корпорації Altera, яка застосовується в пакеті автоматизованого проектування мікросхем QuartusII, аналогічну модель застосовує корпорація Xilinx та інші. Новизна підходу полягає в тому, що модель, на відміну від існуючих, побудована в двох ієрархічних рівнях і передбачаються спеціальні заходи для виключення блокувань. В результаті застосування запропонованого методу, швидкість верифікації базових поліномів була підвищена на три порядки. В якості перспективи розвитку досліджень можна запропонувати розробку моделі для пакету Matlab/Simulink.

Список літератури: **1**. 3GPP TS 26.073. "ANSI-C code for the Adaptive Multi Rate (AMR) speech codec" – Режим доступу: URL: http://www.3gpp.org/ftp/Specs/archive/26_series/26.073/ 24.06.2009 р. – Заголовок з екрану. **2**. 3GPP TS 26.090. "3rd Generation Partnership Project;Technical Specification Group GSM/EDGE Radio Access Network;Channel coding (Release 8)" – Режим доступу: URL:http://www.3gpp.org/ftp/Specs/archive/45_series/45.003/ 24.06.2009 р. – Заголовок з екрану. **3**. 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Multiplexing and channel coding (FDD)(Release 8)" – Режим доступу: URL:http://www.3gpp.org/ftp/Specs/archive/25_series/25.212/ 24.06.2009 р. – Заголовок з екрану. **4**. 3GPP TS 22.071. "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Location Services (LCS); Service description; Stage 1 (Release 8)" – Режим доступу: URL: http://www.3gpp.org/ftp/Specs/archive/26_series/22.071/ 20.09.2009 р. – Заголовок з екрану. **5**. *Gathen J.* Polynomial factorization over F2 / *J. von zur Gathen and J. Gerhard* // Math. Comp. 71 (2002), 1677–1698. **6.** *Brent R.P.* A primitive trinomial of degree 6972593 / *R.P. Brent, S. Larvala and P. Zimmermann* // Math.Comp. 74 (2005), 1001–1002, – Режим доступу: URL: http://wwwmaths.anu.edu.au/ brent/pub/pub224.html – Заголовок з екрану. **7**. *Gathen J.* Modern Computer Algebra / *J. von zur Gathen and J. Gerhard* // Cambridge University Press, Cambridge, UK, 1999. **8**. *Woltman G.* GIMPS, The Great Internet Mersenne Prime Search / *G. Woltman* – Режим доступу: URL: http://www.mersenne.org / – Заголовок з екрану. **9**. *Brent R.P.* A Multi-level Blocking Distinct-degree Factorization Algorithm / *R.P. Brent and P. Zimmermann* // Math. Comp. 74 (2007), 1001–1002, – Режим доступу: URL: http://www.inria.fr – Заголовок з екрану. **10**. *Нефьодов Л.І.*

Застосування пакету MAX+PlusII при викладанні дисципліни "Гнучка автоматизація виробництв" / *Л.І. Нефьодов, С.А. Кривенко* // Сучасні технології підготовки фахівців в умовах подальшого розвитку вищої освіти України: Матеріали міжнародної науково-методичної конференції. – Харків: ХНАДУ, 2005. – С. 53-54. **11**. *Нефедов Л.И.* Управление проектом создания геоинформационной системы для транспортных перевозок / *Л.И. Нефедов, С.А. Кривенко, А.П. Стативка* // Автомобильный транспорт / Сб. научн. тр. – Харьков: Изд-во ХНАДУ. – 2006. – Вып.18. – С. 42-46.

УДК 656.13:681.3

Метод синтеза модели геоинформационной системы на основе базовых полиномов / Нефедов Л.И., Кривенко С.А., Мусиенко Е.Н. // Вестник НТУ "ХПИ". Тематический выпуск: Информатика и моделирование. – Харьков: НТУ "ХПИ". – 2010. – № 21. – С. 117 – 125.

Предложен новый метод для верификации выполнения разложения полинома определенной степени $P(x)$ на множители над полем Галуа GF(2). За основу принятая модель корпорации Altera, которая применяется в пакете автоматизированного проектирования микросхем QuartusII(2009), аналогичную модель применяет корпорация Xilinx и другие. Новизна нашего подхода заключается в том, что на первом этапе применяется временное моделирование в двух иерархических уровнях и предусматриваются специальные мероприятия для исключения блокировок, а на завершающем этапе выполненное натурное моделирование на реальной сверхсовременной микросхеме, которая развивает на три порядка большую скорость вычисления. Ил.: 6. Библиогр.: 11 назв.

Ключевые слова: поле Галуа, GF(2), Altera, QuartusII, Xilinx.

UDC 656.13:681.3

Primitive polynomial model synthesis method for the geoinformation system / Nefedov L.I., Krivenko S.A., Musienko E.N. // Herald of the National Technical University "KhPI". Subject issue: Information Science and Modelling. – Kharkov: NTU "KhPI". – 2010. – №. 21. – P. 117 – 125.

We give a new method for performing the distinct-degree factorization of a polynomial P(x) over GF(2). The model of Altera Corporation was accepted for basic. The Quartus II(2009) development software provides a complete design environment for system-on-a-programmable-chip (SOPC) design. Similar model is applied by the corporation of Xilinx et al. We use a multi-level blocking strategy. The Quartus II Classic Timing Analyzer makes it possible to analyze the performance of all design logic and guides the Fitter to meet your timing goals. Timing simulation produced in 1000 times faster. Figs: 6. Refs 11 titles.

Key words: fields Galua, GF(2), Altera, QuartusII, Xilinx.

Поступила в редакцію 05.10.2009