Севастополь, 1996, С.193-194. 6. Заболотній С.В. Нелінійні перетворювання випадкових величин як перспективний аспект у побудові нових методів перевірки простих статистичних гіпотез // Праці 4-ї української конференції з автоматичного керування "Автоматика-97". – Т.2. – Черкаси, 1997. 7. Заболотній С.В., Коваль В.В. Застосування неортогонального розкладання для перевірки статистичних гіпотез. Матеріали 3-ї Всеукраїнської конференції молодих науковців "ІТОНТ-2002". – Черкаси. 2002. – С.233–234. 8. Заболотній С.В. Розпізнавання випадкових сигналів з дискретним часом на основі неортогонального розкладання випадкових величин // Праці Луганського відділення Міжнародної академії інформатизації № 1 (8). – Луганськ. – 2004. – С. 39–41. 9. Заболотній С.В., Коваль В.В., Салипа С.В. Виявлення відеосигналів із застосуванням нелінійних дискретних фільтрів з постійними коефіцієнтами // Електроніка та системи управління. – 2008. – № 3. – С.77–83. 10. Заболотній С.В. Статистичне розпізнавання образів на основі розкладу в просторі з порідним елементом // Вісник Нац. ун-ту "Львівська політехніка" "Комп'ютерні науки та інформаційні технології" – 2009. – № 638. – С.118–123.

УДК 004.35

Я.Р. Совин, Я.В. Решетар, В.В. Хома
Національний університет "Львівська політехніка",
кафедра захисту інформації

ЕЛЕКТРОННИЙ БЕЗПРОВІДНИЙ ІДЕНТИФІКАТОР ДЛЯ ДОСТУПУ ДО ПК

Наведено програмно-апаратні засоби, використані під час побудови електронного безпровідного ідентифікатора доступу до персонального комп'ютера на базі технології Bluetooth.

Hardware and software tools are resulted used for the construction of electronic wireless-token on the base of Bluetooth technology for access to the personal computer.

Вступ. З розвитком комп'ютерних технологій, проникненням їх у всі сфери життя і діяльності людини зростає необхідність в гарантуванні безпеки інформації. Це потребує розв'язання таких задач, як ідентифікація/автентифікація особи, управління доступом, захист інформації та трафіку. Засобами електронної ідентифікації можуть бути RFID-мітки, proximity-карти, брелоки TouchMemory, магнітні картки, смарт-карти, USB-ключі.

Головною тенденцією розвитку систем електронної ідентифікації є наділення програмно-апаратних засобів максимальною кількістю функцій зі збирання, оброблення інформації та прийняття рішень. Це дає змогу зменшити вплив "людського фактора" та звільнити користувача від виконання рутинних операцій.

Аналіз останніх досліджень та постановка задачі. На цей час серед засобів доступу до ПК найбільші функціональні можливості та найвищий рівень захисту мають електронні ідентифікатори у вигляді смарт-карт та USB-ключів або токенів [3, 9].

Смарт-карта – це пластикова карта з вбудованим мікроконтролером, який виконує функції розмежування доступу до інформації, що зберігається в пам'яті, обробки й обміну даними, а також

реалізацію криптографічних алгоритмів. У вбудованій EEPROM-пам'яті мікроконтролера зберігаються персональні дані користувача: паролі, ключі, сертифікати тощо [3, 4, 9].

Смарт-карти можуть бути як контактними так і безконтактними, проте в обох випадках потребують спеціальних зчитувачів. Достатньо висока ціна зчитувачів та самих смарт-карт поки що обмежує їх широке використання як електронних ідентифікаторів для доступу до ПК.

Системи ідентифікації та захисту інформації на основі USB-ключів широко застосовують у різноманітних сферах, де використовуються комп'ютери. Така система не потребує зчитувача, позаяк всі сучасні комп'ютери оснащені портами USB, що значно заощаджує кошти [9].

Найпоширенішими у використанні USB-ключами (ідентифікаторами) є так звані USB-токени – персональний засіб автентифікації і зберігання даних, що апаратно підтримує роботу з цифровими сертифікатами й електронними цифровими підписами. Конструктивно цей пристрій виконаний як звичайний флеш-носій типу USB Flash Drive, але за виконуваними функціям він багато в чому відповідає смарт-карті [9]. Користуватися цими пристроями дуже зручно, оскільки не потрібно запам'ятовувати безліч паролів і кодів доступу, а вся інформація зберігається в USB–токені. Крім того, на носії можуть бути цифрові підписи, сертифікати й інша інформація, яку небезпечно зберігати на відкритих носіях інформації.

За своєю внутрішньою архітектурою USB-ключі поділяються на ідентифікатори на основі смарт-карт процесорів та на основі захищених мікроконтролерів [9].

В USB-ключах на основі захищених мікроконтролерів процесор в повному обсязі виконує роботу ключа і є його основним елементом. Він реалізує інтерфейс USB, містить модуль пам'яті, з програмою і конфігураційними даними (Firmware Memory), а також модуль оперативної пам'яті.

З погляду суб'єкта ідентифікації, USB-ключ повинен ідентифікувати себе відносно ПК, якщо секретні дані, до яких передбачається доступ, містяться безпосередньо в ПК (на сервері). Сама процедура ідентифікації здійснюється з використанням односторонніх геш-функцій.

Метою статті є показати можливість реалізації ідентифікатора, який поєднує функції безконтактного доступу та електронного ключа, що дає змогу значно підвищити зручність роботи та надійність захисту, зменшивши витрати на впровадження.

Опис апаратної частини. Оскільки найпоширенішим, компактним та дешевим безпровідним інтерфейсом, яким оснащені ПК, є Bluetooth, його було вибрано для реалізації в електронному ідентифікаторі. Структурна схема ідентифікатора наведена на рис. 1.

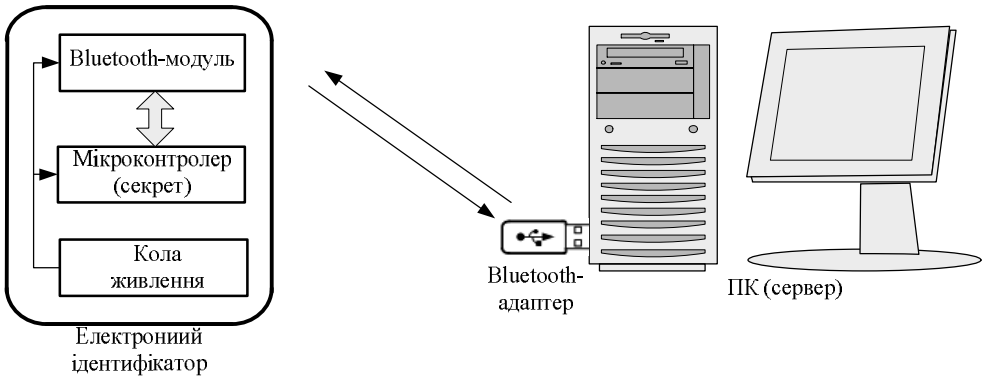

Рис. 1. Складові частини програмно-апаратної системи ідентифікації

Електронний ідентифікатор виконаний у вигляді брелока і складається з Bluetooth-модуля, мікроконтролера та схеми живлення. Мікроконтролер здійснює налаштування роботи модуля, приймання і передавання даних та реалізує криптографічний протокол ідентифікації. В EEPROM-пам'яті контролера зберігається унікальний секрет (ключ), який використовується в алгоритмі аутентифікації. ПК повинен містити вбудований або зовнішній Bluetooth-адаптер, який під'єднується до USB-порта.

Хоча специфікація технології Bluetooth і передбачає проходження процедури ідентифікації з подальшим шифрування трафіку, проте ряд відомих вразливостей та атак на Bluetooth-протокол вимагає здійснення додаткової процедури ідентифікації, крім передбачених механізмів безпеки.

Встановлена на ПК спеціальна програма щосекунди ініціює з'єднання із ідентифікатором та здійснює сеанс автентифікації. У разі невдалого завершення сеансу або неможливості встановити з'єднання через Bluetooth робота ПК блокується. Оскільки дальність дії Bluetooth-адаптерів становить 7–10 метрів (без перешкод), блокування буде відбуватиметься автоматично, як тільки користувач віддалиться від свого робочого місця на зазначену віддаль.

Bluetooth-модуль реалізовано на мікросхемі LMX9838 фірми National Semiconductor. Це завершений компактний (10x17x2 мм) модуль, який містить вбудовані антену, 2.4 ГГц радіотракт, 16-бітний RISC контролер, інтерфейс універсального асинхронного приймача-передавача (UART) для обміну з хост-мікроконтролером (рис. 2). Модуль підтримує такі Bluetooth-профілі, як: GAP (Generic Access Profile), SDAP (Service Discovery Profile) та SPP (Serial Port Profile), що потрібні для роботи ідентифікатора. Швидкість обміну даними через профіль SPP задається хост-контролером, при максимальному значенні 921600 біт/секунда [7].

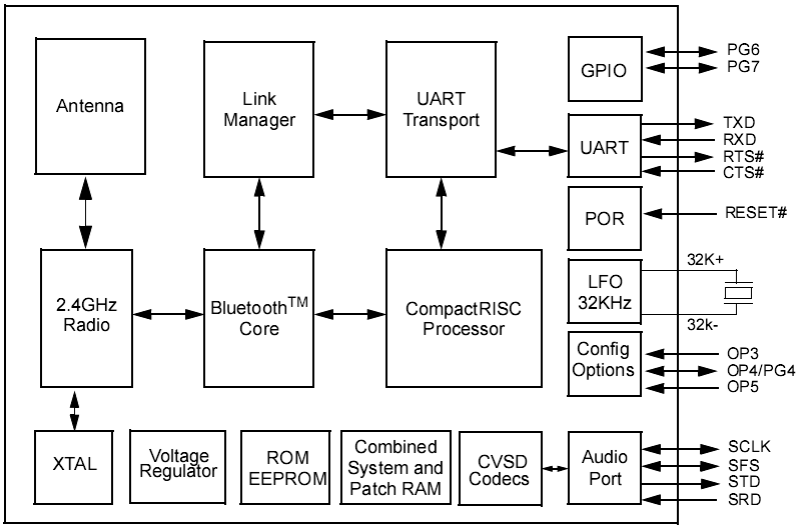

Рис. 2. Структурна схема Bluetooth-модуля LMX9838

Як хост-мікроконтролер вибрано ATmega16 фірми Atmel. Враховувалися достатньо великі обсяги пам'яті (Flash – 16 Кбайт, SRAM – 1 Кбайт) та висока продуктивність AVR-ядра (1MIPS/МГц). Також важливим є наявність у всіх мікроконтролерах сім'ї AVR спеціальних комірок (Lock Bits) для захисту вмісту Flash-пам'яті програм та EEPROM-пам'яті даних від зчитування або запису [6].

Фрагмент схеми ідентифікатора, який показує з'єднання хост-мікроконтролера з LMX9838, наведено на рис. 3.

На виводах $OP5 - OP3$ мікроконтролер задає швидкість обміну даними з LMX9838 через лінії RxD / TxD. Передавання даних і команд відбувається в пакетному режимі, а лінії CTS / RTS використовуються для узгодження швидкодії мікроконтролера і Bluetooth-модуля. Через вивід $RSTB$ мікроконтролер здійснює початкове встановлення Bluetooth-модуля.

Протокол ідентифікації. Виявивши Bluetooth-ключ, ПК (сервер), з метою його ідентифікації, надсилає певне випадкове повідомлення (за схемою одноразових паролів). Отримавши це повідомлення, ключ здійснює операцію гешування (підпису) з використанням секретного коду та повертає результат ПК. Програма на ПК здійснює аналогічні дії із гешування та порівнює результати. Якщо дайджести (геш-коди) збігаються, процес ідентифікації вважається успішним. Без знання секретного коду, який не покидає меж пристрою, здійснити правильну операцію гешування неможливо.

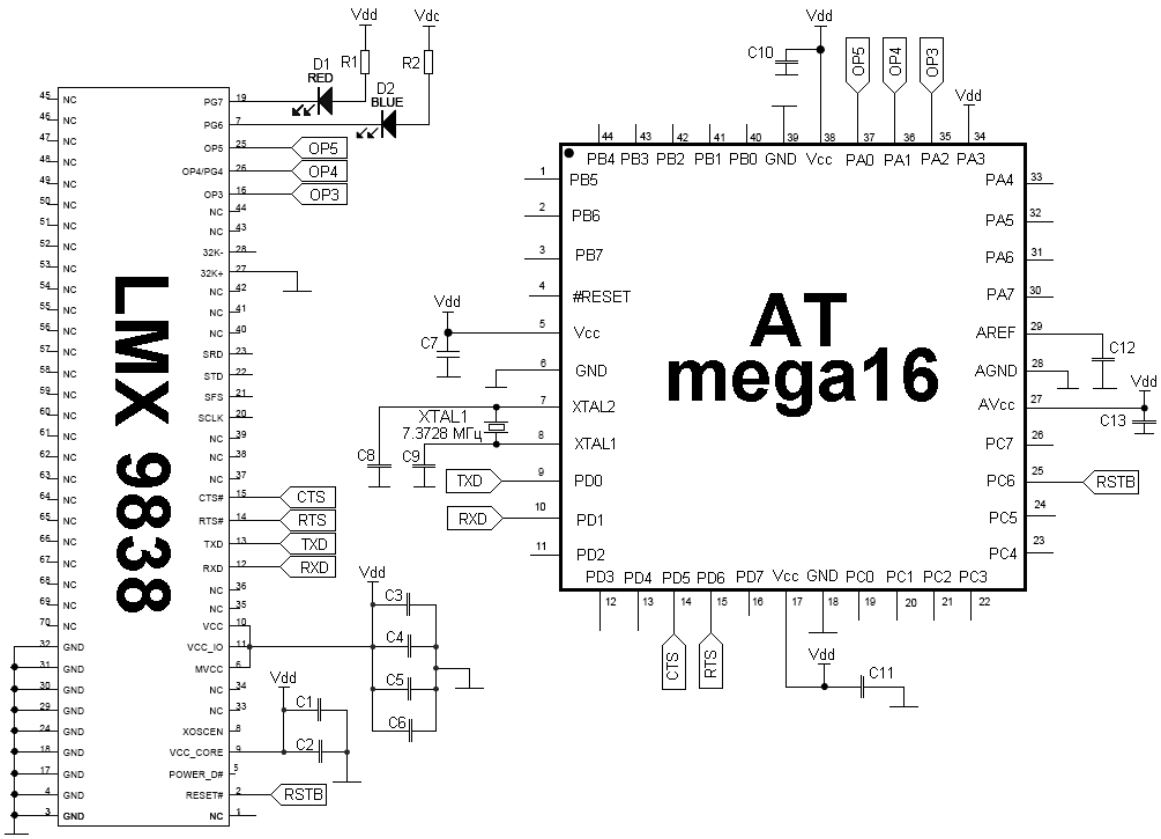

Рис. 3. Схема під'єднання Bluetooth-модуля LMX9838 до керуючого мікроконтролера

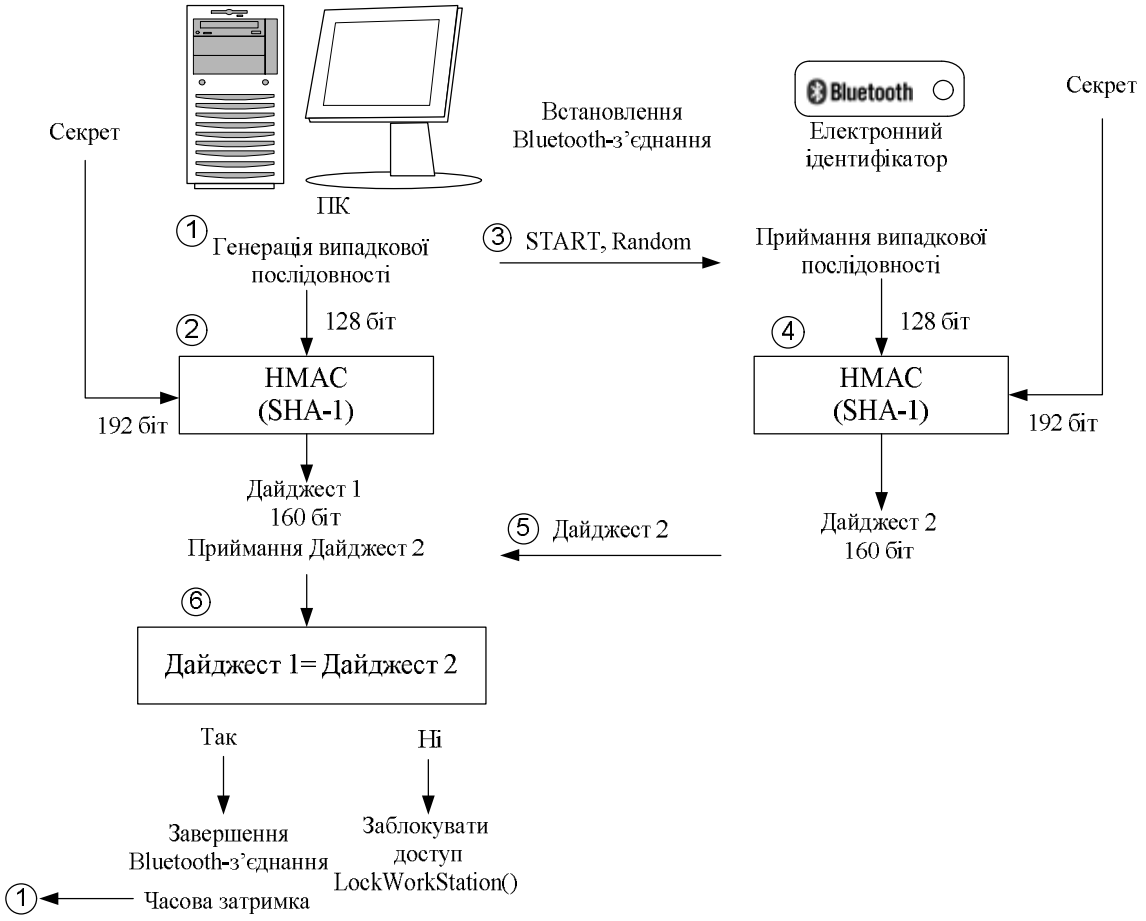

Рис. 4. Схема ідентифікації за допомогою електронного ідентифікатора

Для реалізації протоколу ідентифікації було вибрано найпоширеніший алгоритм HMAC (The Keyed-Hash Message Authentication Code), що відповідає вимогам ДСТУ [1, 5].

Перевагами HMAC є: можливість використання без модифікацій існуючих геш-функцій, можливість заміни геш-функції у випадку компрометації, висока швидкість роботи алгоритму, можливість використання ключів.

Як зазначалося, алгоритм HMAC не визначає конкретну геш-функцію H, що має бути використана. При розробленні електронного ідентифікатора було вибрано геш-функцію SHA-1, з огляду на те, що вона є стандартизована в Україні [2].

В алгоритмі SHA-1 розмір вхідного блока b становить 512 біт, а геш-коду n – 160 біт. Довжина секрету (ключа) K було вибрано такою, що дорівнює 192 бітам, а випадкового повідомлення M – 128 біт. Рис. 4 ілюструє перебіг процедури ідентифікації.

Опис програмної частини. Програмне забезпечення складається з програми для мікроконтролера ідентифікатора (клієнтська частина) та серверної програми, яка встановлюється на ПК.

Програма для ATmega16 написана мовою C з використанням компілятора CodeVisionAVR. Обчислення значення HMAC займає 766300 тактів, що при тактовій частоті 7.3728 МГц становить 104 мс.

```
#include <wincrypt.h>
.....
BYTE KeyBlob[] =
{ 0x08, 0x02, 0x00, 0x00, 0x03, 0x66, 0x00, 0x00,    // BLOB header
  0x18, 0x00, 0x00, 0x00,                             // Key length, in bytes
  '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0',              // Secret 192 bits
  '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0' };

for(BYTE ii = 0; ii < 24; ii++)  KeyBlob[ii + 12] = Secret_key[ii];

HCRYPTPROV  hProv;
HCRYPTKEY   hKey;

CryptAcquireContext(&hProv,  NULL,  MS_ENHANCED_PROV,  PROV_RSA_FULL, CRYPT_VERIFYCONTEXT);

CryptImportKey(hProv,  KeyBlob,  sizeof(KeyBlob),  0,  CRYPT_EXPORTABLE, &hKey);

HCRYPTHASH hHash;
CryptCreateHash(hProv, CALG_HMAC, hKey, 0, &hHash);

HMAC_INFO hmac_i;
hmac_i.HashAlgid = CALG_SHA1;

CryptSetHashParam(hHash, HP_HMAC_INFO, (const BYTE*) &hmac_i, 0) ;

BYTE Rand_Message[16];
CryptGenRandom(hProv, 16, Rand_Message);

CryptHashData(hHash, Rand_Message, 16, 0);

DWORD count = 0;
CryptGetHashParam(hHash, HP_HASHVAL, NULL, &count, 0) ;

char* hash_value = static_cast<char*>(malloc(count + 1));
ZeroMemory(hash_value, count + 1);

CryptGetHashParam(hHash, HP_HASHVAL, (BYTE*)hash_value, &count, 0);

for ( BYTE ii = 0; ii < 20; ii++ )  HMAC_digest[ii] = hash_value[ii];
```

Рис. 5. Реалізації криптографічних функцій на базі CryptoAPI для серверної програми

Серверна програма написана мовою C++ у середовищі Microsoft Visual Studio. Для реалізації криптографічних функцій було вибрано бібліотеку CryptoAPI. Бібліотека CryptoAPI є складовою частиною операційних систем сім'ї Windows. У CryptoAPI реалізація всіх алгоритмів (шифрування, гешування, генерація випадкових чисел тощо) повністю виведена із складу самої бібліотеки і

реалізується в окремих, незалежних динамічно завантажуваних бібліотеках – “криптопровайдерах” (Cryptographic Service Provider – CSP). Сам же CryptoAPI надає кінцевому користувачеві уніфікований інтерфейс роботи з CSP [8].

Вплинути на хід алгоритму, реалізованого в криптопровайдері, неможливо, оскільки компоненти криптосистеми Windows (усі без винятку) повинні мати цифровий підпис (тобто підписується й dll-файл криптопровайдера). Отже, така реалізація є захищенішою порівняно з власним написанням відповідних криптографічних алгоритмів.

Фрагмент програми, який здійснює обчислення HMAC з використанням CryptoAPI, наведено на рис. 5 (для зменшення обсягу опущено перевірки викликів функції та інші неістотні деталі).

Для реалізації функцій Bluetooth вибрано безкоштовний стек протоколів Software Development Kit BCM1000-BTW фірми Broadcom. У BCM1000-BTW входять dll-бібліотеки та C++ інтерфейс їх використання. Обмін випадковим повідомленням та дайджестом між Bluetooth-адаптером ПК та ідентифікатором реалізовано через профіль SPP, який, по суті, являє собою віртуальний COM-порт.

Для перевірки функціонування програмно-апаратних засобів було створено макет (рис. 6), який складається з відлагоджувальної плати STK500 із встановленими мікроконтролером ATmega16 і кварцовим резонатором та відлагоджувальної плати LMX9838Donge з Bluetooth-модулем LMX9838. Обмін даними між палатами здійснюється через роз’єм RS-232. Як Bluetooth-адаптер було використано DBT-122 фірми D-Link. На ПК встановлено операційну систему Windows XP/SP2. Тестування підтвердило коректність функціонування системи ідентифікації.

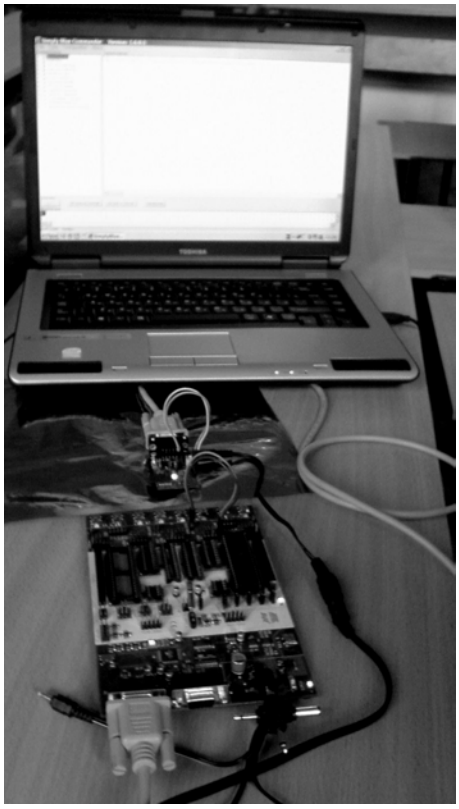

Рис. 6. Макет електронного ідентифікатора

Висновки. Розроблений та відпрацьований на макеті електронний ідентифікатор є зручним та дешевим рішенням для організації контролю доступу до ПК. Ресурси керуючого мікроконтролера (використано 25 % пам’яті) дадуть надалі змогу розширити функціональні можливості пристрою за рахунок підтримки шифрування/розшифрування файлів та захищеного зберігання даних (при встановленні мікросхеми NAND флеш-пам’яті).

1. ДСТУ ISO/IEC 10118-1:2003. Методи захисту. Геш-функції. Частина 1. Загальні положення. – К. Держспоживстандарт України, 2004. – 6 с. 2. ДСТУ ISO/IEC 10118-3:2005. Методи захисту. Геш-функції. Частина 3. Спеціалізовані геш-функції. – К. Держспоживстандарт України, 2006. – 84 с. 3. Smart Card Handbook / Rankl W., Effing W. – West Sussex: John Wiley & Sons, Ltd, 2003. – 1088p. 4. Smart Card Applications. Design Models for using and programming smart cards / Rankl W. – West Sussex: John Wiley & Sons, Ltd, 2007. – 217 p. 5. FIPS 198-1. National Institute of Standards and Technology. The Keyed-Hash Message Authentication Code (HMAC). Federal Information Processing Standards Publication 198-1, July 2008. 6. Евстифеев А. В. Микроконтроллеры AVR семейства Mega. Руководство пользователя. – М.: Издательский дом "Додэка-XXI", 2007. – 592 с. 7. LMX9838 Software Users Guide. – National Semiconductor, 2006. – 198 p. 8. Прикладная криптография. Использование и синтез криптографических интерфейсов / Л.Ю. Щербаков, А.В. Домашев. – М.: Издательско-торговый дом "Русская Редакция", 2003. – 416 с. 9. Электронная идентификация. Бесконтактные электронные идентификаторы и смарт-карты / В.Л. Дшхунян, В.Ф. Шаньгин. – М.: ООО "Издательство АСТ": Издательство "НТ Пресс", 2004. – 695 с.

УДК 621.374

Н.М. Лужецька, В.Б. Дудикевич, А.Я. Горпенюк
Національний університет "Львівська політехніка",
кафедра захисту інформації

ДОСЛІДЖЕННЯ ПОКАЗНИКОВОГО КОНВЕЄРНОГО ЧИСЛО-ІМПУЛЬСНОГО ФУНКЦІОНАЛЬНОГО ПЕРЕТВОРЮВАЧА

Подано результати синтезу та дослідження конвеєрної структури показникового число-імпульсного функціонального перетворювача. Показано, що максимальна частота роботи такої структури є значно вищою за максимальну частоту роботи класичної структури. Запропоновано способи покращення точності конвеєрної структури показникового перетворювача.

The results of conveyor structure of exponential pulse-number functional transformer synthesis and research are given. Maximal work frequency such structure is considerably higher than maximal work frequency of classic structure. Also are offered the methods of conveyor structure of exponential transformer exactness improvement.

Постановка проблеми. Показникові число-імпульсні функціональні перетворювачі (ЧІФП) використовуються у вимірювальній та обчислювальній техніці для побудови апроксимуючих вимірювальних перетворювачів, для швидкого виконання операцій множення та ділення сигналів тощо [1]. Сьогодні актуальною залишається задача підвищення максимальної частоти роботи показникових ЧІФП. Її розв'язання дає змогу застосовувати вищу частоту квантування, збільшивши точність вимірювання. Однак підвищення максимальної частоти роботи класичних ЧІФП призводить до зменшення їх розрядності, що збільшує похибку перетворення ЧІФП. Це протиріччя обмежує можливості подальшого підвищення точності вимірювальних перетворювачів, побудованих із застосуванням класичного показникового ЧІФП [1, 2]. Для вирішення проблеми залежності максимальної частоти роботи ЧІФП та його розрядності було розроблено конвеєрні