УДК 621.396.4 (043.2)

КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ ЗАХИЩЕНОГО КАНАЛУ КЕРУВАННЯ БЕЗПІЛОТНИМ ЛІТАЛЬНИМ АПАРАТОМ

Г. Ф. Конахович, д-р техн. наук, проф.;
Д. І. Бахтіяров, О. Ю. Лавриненко

Національний авіаційний університет

e-mail: bakhtiyaroff@nau.edu.ua

Розроблено комп'ютерну модель захищеного від несанкціонованих дій зловмисника каналу передавання сигналів керування безпілотними літальними апаратами. Проведено моделювання процесу створення потоку керуючої інформації у вигляді відео-імпульсів з широтно-імпульсною модуляцією з подальшим перетворенням їх в єдиний цифровий потік і його проходження через блочний алгоритм шифрування сертифікованого в Україні стандарту ГОСТ 28147-89, а також введення завадостійкого кодування за допомогою коду Хеммінга (31, 26), як модуляцію було використано квадратурну амплітудну модуляцію QAM-16. Експериментальним шляхом підтверджено працездатність розробленої моделі, що свідчить про коректність реалізації блочного шифроалгоритму та схеми завадостійкого кодування. Розроблена модель може бути використана для досліджень ефективності захисту інформації у каналі керування безпілотними літальними апаратами.

Ключові слова: ГОСТ 28147-89, БПЛА – безпілотний літальний апарат, завадостійке кодування; шифрування.

Computer model protected from unauthorized actions attacker channel signaling control unmanned aircraft. The modeling process of creating flow control information in the form of video pulses with pulse-width modulation with subsequent conversion of them into a single digital stream and its passage through the block encryption algorithm certified in Ukraine GOST 28147-89, and the introduction of noise-immune coding using Hamming code (31, 26) as the modulation used quadrature amplitude modulation QAM-16. Experimentally verified performance of the model, indicating that the correct implementation of block cipher algorithm and noise-immune coding scheme. The model can be used to study the effectiveness of information management in the channel unmanned aircraft.

Keywords: GOST 28147-89, UAV – Unmanned aerial vehicle, immunity encoding, encryption.

Вступ

Безпілотні літальні апарати (БПЛА) на сьогодні залишаються найперспективнішими, динамічно розвиваються та являють собою унікальні системи цільового та цивільного призначення. Уже кілька поколінь БПЛА виконують завдання повітряної розвідки. У перспективі разом з розвідувальними функціями, на БПЛА планується покладати і виконання цільових завдань.

На даний час безпілотна авіація України має низку проблем в галузі захищеності сигналу керування.

Мета роботи — розроблення комп'ютерної моделі захищеної від зловмисних дій системи передавання сигналів керування БПЛА, що може бути реалізована у межах чинного законодавства у сфері захисту інформації.

Аналіз досліджень та публікацій

Аналіз останніх досліджень показує, що актуальні питання захисту інформації в каналах передавання сигналів керування безпілотними літальними апаратами від несанкціонованих дій зловмисника недостатньо висвітлені в літературі.

Постановка завдання

Захищений радіоканал має відповідати таким вимогам та обмеженням: необхідно змоделювати захищену передачу сигналів керування від пуль-

та керування до сервоприводів та контролеру двигуна БПЛА.

Під захищеним передаванням розуміють забезпечення автентифікації, цілісності та доступності під час передавання сигналів управління для того, щоб:

1) противник не зміг замінити сигнали управління і посадити БПЛА — здійснюється шляхом використання сертифікованого криптографічного алгоритму ГОСТ 28147-89;

2) сигнали управління могли бути передані в умовах постановки перешкод помірної інтенсивності — здійснюється шляхом використання завадостійкого кодування.

Результати дослідження

Комп'ютерна модель розроблена у програмному середовищі MathCAD версії 15. Це середовище обрано через зручність користування, зокрема через наявність графічного режиму роботи користувацького інтерфейсу. Під час створення моделі для набору команд, функцій, формул існує можливість використовувати як клавіатуру, так і кнопки на спеціальних панелях інструментів графічного інтерфейсу. Можливості MathCAD дозволяють здійснювати обчислення безпосередньо за уведеними формулами, що забезпечує поетапну перевірку коректності створюваної моделі. Крім того, будь-які змінні, формули, параметри можна змінювати, спостерігаючи наочно відповідні зміни результату. Тобто, це дозволяє зручно візуалізувати як сам текст математичної моделі, так і результати розрахунків. У більшості інших програмних середовищах (наприклад, Maple або MATLAB) обчислення здійснюються в режимі програмного інтерпретатора, що унеможливлює поетапну перевірку коректності фрагментів моделі під час її створення.

Під час моделювання захищеної системи передавання виявився і недолік Mathcad — недостатність вбудованих функцій моделювання (зокрема, для здійснення серійних розрахунків згідно криптографічного алгоритму ГОСТ 28147-89). Тим менш, зручність налагодження фрагментів моделюючої програми переважає цей недолік.

Тому у даному випадку обрано програмне середовище MathCad для створення у ньому моделі захищеної системи передавання сигналів керування БПЛА.

Структурна схема створюваної моделі зображена на рис. 1.

Як видно з рис. 1 модель має 9 етапів проходження інформації від утворення сигналу керування до його отримання, у тому числі 4 етапи на передавальній стороні й 4 на приймальній. Середовищем передавання інформації обрано ідеальний канал зв'язку з погляду завадостійкості природного характеру, оскільки інтерес являє стійкість щодо вмисно утворених завад.

У процесі виконання моделювання було вирішено низку таких завдань:

1. Створення інформаційного потоку на вході каналу передавання. Для цього були змодельовані сигнали керування, що надходять від реальної апаратури керування (формується потік керуючої інформації у вигляді відео-імпульсів з широтно-імпульсною модуляцією) з подальшим перетворенням їх в єдиний цифровий потік.

2. Створення моделі криптографічного перетворення (шифрування/розшифрування) за алгоритмом ГОСТ 28147-89 у режимі простої заміни елементів інформаційного потоку.

3. Моделювання завадостійкого кодування зашифрованого інформаційного потоку. Для цього змодельовано роботу завадостійкого кодера. А також створення моделі зворотного перетворення — процес завадостійкого декодування зашифрованого інформаційного потоку (робота завадостійкого декодера).

4. Моделювання роботи модулятора. Для цього створено модель модуляції зашифрованого й закодованого інформаційного потоку за допомогою QAM-16, а також процес зворотного перетворення, тобто демодуляції.

Після вирішення кожного з вище наведених завдань було отримано 4 окремі моделі, які в подальшому було об'єднано для створення цілісної моделі захищеної системи передавання сигналів керування БПЛА.

Створення інформаційного потоку на вході каналу зв'язку керування БПЛА

Вхідними даними для початку моделювання є імітація ШІМ-сигналу, яку отримуємо з виходу апаратури керування БПЛА. ШІМ сигнал має фіксовану довжину періоду $T \approx 20$ мс. А сам період складається з 8 імпульсів, що повністю описують положення двох джойстиків на пульті керування з чотирма можливими станами кожен. При знаходженні джойстика у верхньому положенні тривалість імпульсу збільшується, а при нижньому положенні — навпаки.

Для моделювання був заданий сигнал, що складається з трьох періодів по 8 імпульсів кожен, загальною тривалістю ~ 60 мс. У першому періоді другий і восьмий імпульс імітує положення джойстика в нижньому положенні, а п'ятий імпульс — верхнє положення; в другому періоді другий і п'ятий імпульс — верхнє положення джойстиків, восьмий імпульс — нижнє; в третьому періоді третій і шостий імпульс — імітували нижнє положення, а п'ятий і восьмий — верхнє. Ці дані приведені в табл. 1.

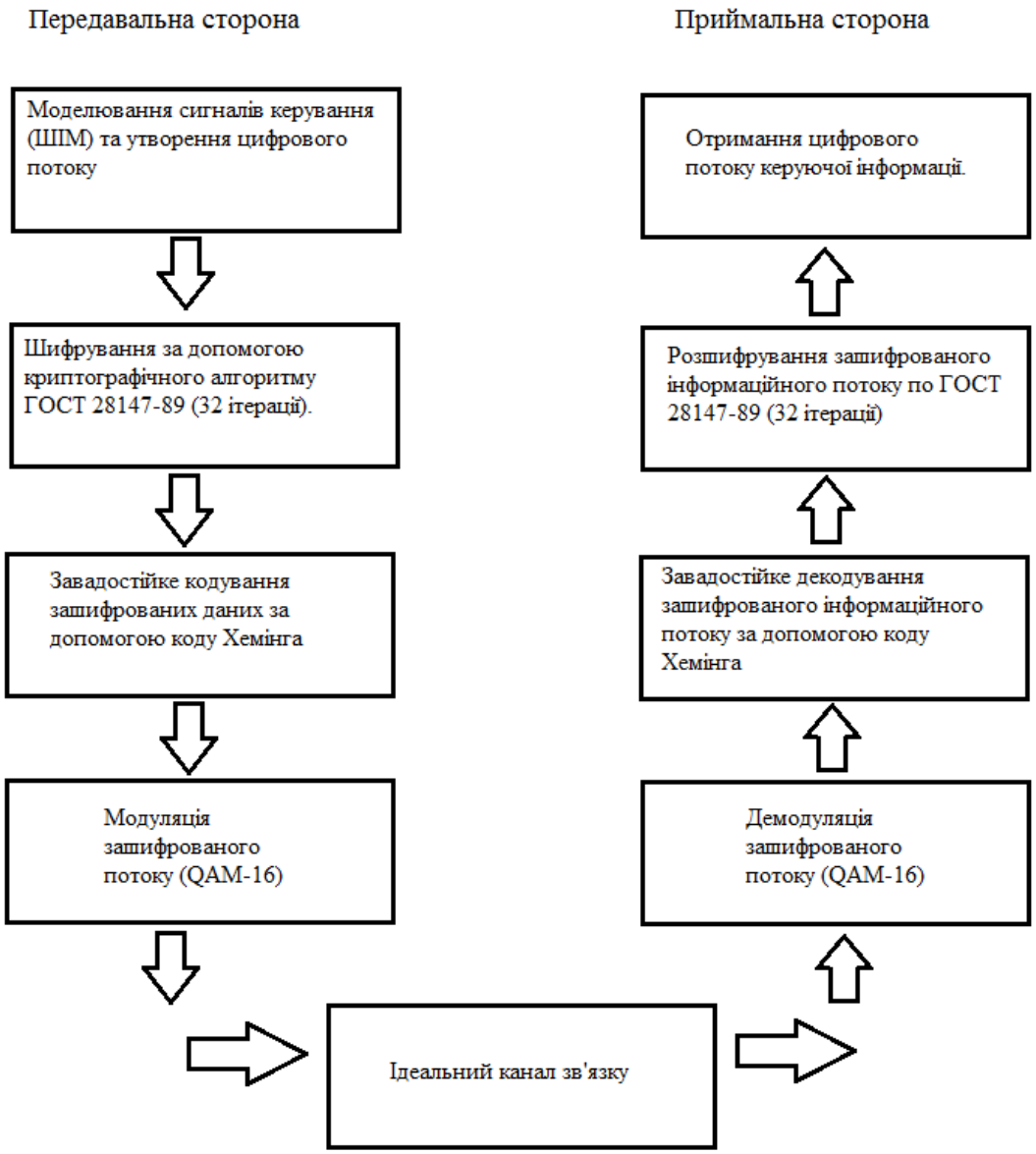

Рис. 1. Структурна схема моделі захищеного каналу передавання сигналів керування до БПЛА

Таблиця 1

Імітація положення джойстиків у кожному імпульсі

№ імпульса	1	2	3	4	5	6	7	8
Період 1		↓			↑			↓
Період 2		↑			↑			↓
Період 3			↓		↑	↓		↑

Для наглядного зображення станів джойстиків-контролерів на апаратурі керування безпілотним літальним апаратом, показано табл. 2, де наведені ці дані. Після записування й оброблення даних в програмному середовищі MathCad версії 15 отримуємо сигнал, форму якого показано на рис. 2. Для спрощення було прийнято, що тривалість імпульсу відповідає положенню

- ручки апаратури керування:
- верхнє положення — 1 мс;
- середнє положення — 1,5 мс;
- нижнє положення — 2 мс.

Для продовження роботи з даним сигналом, його потрібно перевести в вигляд цифрового потоку даних, для цього використано модель аналого-цифрового перетворювача з розрядністю, що дорівнює одиниці. Також були підібрані такі параметри при перетворенні аналогового сигналу, щоб довжина вихідного цифрового потоку становила 640 біт.

Отриманий цифровий потік зображено на рис. 3 (показано лише перші 12 біт).

Таблиця 2

Дані про положення джойстиків-контролерів у кожному періоді передавання сигналів керування

	Стан джойстика	
	Джойстик 1	Джойстик 2
Період 1	→	↘
Період 2	→	↗
Період 3	↙	↑

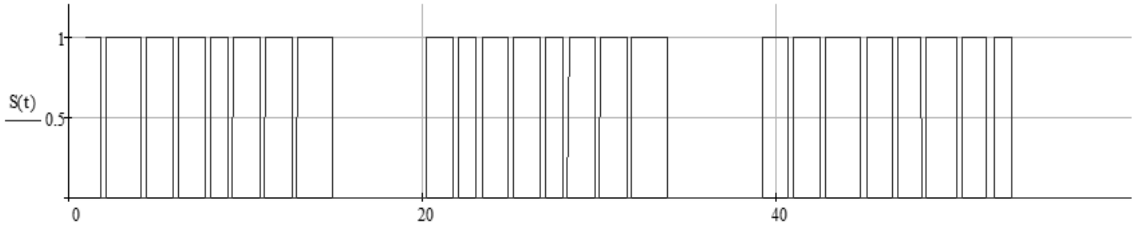

Рис. 2. Зображення ШІМ-сигналу керування

$x^T =$

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	1	1	1	1	1	1	1	1	1	...

Рис. 3. Цифровий потік даних від апаратури керування.

Шифрування даних з використанням криптографічного алгоритму ГОСТ 28147-89

Подальше перетворення цифрового потоку полягає в його шифруванні за допомогою стандарту симетричного шифрування ГОСТ 28147-89 у режимі простої заміни. Оскільки це блочний шифроалгоритм, то він працює з блоками даних кратними 64-м бітам. На етапі аналого-цифрового перетворення було сформовано потік довжиною 640 біт, а потім він був перетворений у масив розмірністю 10 на 64.

Саме з масивами такої розмірності й працює алгоритм ГОСТ 28147-89 у даній моделі (у подальшому будуть показані саме масиви для зручності відображення). Згідно з цим стандартом вхідна послідовність даних повинна пройти через 32 раунди шифрування, для повноцінного закриття інформації. Як ключ обрано випадкову бінарну послідовність довжиною 256 біт, яка була потім поділена на 8 ключів по 32 символів у кожному. У процесі шифрування виконують стандартні для цього режиму (простої заміни) функції (перелік даних в порядку їх виконання): розбиття вхідних даних на 32-бітні на L- и R-підблоки, визначення номера підключа, завантаження в програму 32-бітного ключа з текстового файлу, підсумовування (за $\mathrm{mod}2^{32}$) R-підблока із 32-бітним ключем, розбиття результата підсумовування на 4-бітні S-блоки, задання таблиці замін для S-блоків, конкатенація S-блоків в 32-бітні під блоки, циклічний зсув 32-бітних підблоків вліво на 11 розрядів, підсумовування по mod2 отриманих 32-бітних підблоків с L-підблоками, формування L-підблоку на виході, формування шифротексту шляхом конкатенації L- і R-підблоків на виході. Їх виконання повторюється в кожному раунді.

На рис. 4 показано вигляд цифрового потоку після проходження ним першого раунду шифрування (показано перші 10 біт). Як видно, порядок біт змінено, порівняно з вхідною послідовністю.

На рис. 5 показано вигляд послідовності після проходження всіх 32-х раундів шифрування (показано лише перші 10 біт).

$SHIFR2 := Vec(C1)^T =$

	1	2	3	4	5	6	7	8	9	10	11
1	1	1	1	1	1	1	1	1	1	1	...

Рис. 4. Цифровий потік отриманий після першого раунду шифрування

$SHIFR33 := Vec(C1)^T =$

	1	2	3	4	5	6	7	8	9	10	11
1	0	1	0	0	0	1	1	0	1	0	...

Рис. 5. Цифровий потік після 32-х раундів шифрування

На рис. 6 зображена функція для порівняння двох бінарних масивів в програмному середовищі MathCad версії 15, результатом якої є число, що показує кількість розбіжностей між цими масивами, тобто кількість позицій в яких вони відмінні, також показано порівняння відкритого і зашифрованого тексту.

```
Compare(A,B) := | N ← 0
                | for i ∈ 1..rows(A)
                |   for j ∈ 1..cols(A)
                |     N ← N + 1 if A_{i,j} ≠ B_{i,j}
                | N

Compare(VhPos1,C1) = 536
```

Рис. 6. Функція для порівняння масивів і результат порівняння відкритого та зашифрованого текстів

Як видно з рис. 6 відкритий і зашифрований текст відрізняються в 536-ти позиціях з 640.

У відсотковому відношенні це дорівнює 84 %. Отже, зашифровані дані суттєво відрізняються від вихідних.

Завадостійке кодування

Наступним етапом комп'ютерного моделювання є процес створення схеми завадостійкого кодування зашифрованих даних, які для цього з матричного вигляду були перетворені в цифровий масив, за допомогою коректувального коду Хеммінга. Для проведення кодування було обрано наступні характеристики коду — число контрольних розрядів, що дорівнює 5, тобто вхідний масив поділяється на блоки по 26 біт, і після підстановки в кожний блок по 5 контрольних бітів, довжина закодованого блоку становить 31 біт. Тобто зашифрований потік даних завдовжки 640 біт у процесі кодування поділиться на 640/26 ≈ ≈ 25 блоків (останній доповниться нулями), у кожен з яких додають 5 перевірочних бітів у позиціях, що дорівнюють степені двійки (0,1,4,8,16). У результаті отримаємо кодову швидкість 25/31. А коректувальна здатність коду — виправлення одного хибно прийнятого біту в кожному блоці. Тобто, теоретично, даний код здатен виправити 25 помилок у загальному масиві даних, за умови, що ці помилки будуть виникати на відстані ≈31 біт одна від одної. На рис. 7 показано цифровий потік зашифрованих даних отриманий після використання схеми завадостійкого кодування Хеммінга (показано перші 10 біт цифрового масиву). За рахунок перевірних символів і додаткових нулів в останньому блоці довжина зросла до 775 біт.

$KOD := Vec(huints)^T =$

	1	2	3	4	5	6	7	8	9	10	11
1	1	1	0	0	1	0	0	1	0	1	...

Рис. 7. Цифровий потік отриманий після завадостійкого кодування

Модем QAM-16

Наступним етапом комп'ютерного моделювання є імітація передавання зашифрованої цифрової послідовності керуючої інформації за допомогою квадратурної амплітудної модуляції QAM-16, де несучим коливанням є синусоїда з частотою 433 МГц (частота несучого коливання) реальної апаратури керування БПЛА цивільного призначення. Під час модуляції вхідний цифровий потік поділяється на 2 підпотоки, які передаються з різними фазами і відрізняються на 90 градусів. Кожен потік перетворює (модулює) сигнал-носій, змінюючи її амплітуду відповідно до схеми модуляції. Оскільки 775 бітів націло не розділяються на 4 потоки, тому 3 останні біти втрачаються, але оскільки це нульові біти додані під час кодування, то реальної інформативної цінності вони не несуть. Як середовище передавання для комп'ютерного моделювання було обрано ідеальний канал зв'язку, оскільки він не вносить ніяких змін в сигнал, що передається, а це є важливим для того, щоб відслідкувати і контролювати всі процеси, що виконуються над сигналом.

Далі модульований сигнал поступає на демодулятор, в якому поєднані компаратор і мультиплексор. На виході демодулятора отримано послідовність довжиною 772 біти (показано перші 10 біт), зображену на рис. 8.

$\_X^T =$

	0	1	2	3	4	5	6	7	8	9	10	11
0	1	1	0	0	1	0	0	1	0	1	1	...

Рис. 8. Отриманий цифровий потік після операцій модуляції/демодуляції

Для перевірки коректуючої здатності завадостійкого кодування Хеммінга, що використано в комп'ютерній моделі, штучно введено 8 помилок у демодульовану послідовність.

Позиції, де введено помилки були обрані випадково в межах блоків даних, а саме 1, 157, 258, 309, 360, 450, 511, 670, а також було додано останні три нульові біти, які були видалені під час модуляції, для коректності процесу декодування.

На рис. 9 зображено перші 10 біт цифрових послідовностей з помилками й без них.

1.

$_X^T$ =

	0	1	2	3	4	5	6	7	8	9	10	11
0	1	0	0	0	1	0	0	1	0	1	1	...

X^T =

	0	1	2	3	4	5	6	7	8	9	10	11
0	1	1	0	0	1	0	0	1	0	1	1	...

2.

$_X^T$ =

	152	153	154	155	156	157	158	159	160	161	162	163
0	1	1	0	0	1	1	0	1	0	0	0	...

X^T =

	152	153	154	155	156	157	158	159	160	161	162	163
0	1	1	0	0	1	0	0	1	0	0	0	...

3.

$_X^T$ =

	304	305	306	307	308	309	310	311	312	313	314	315
0	1	0	0	1	0	1	0	0	0	0	0	...

X^T =

	304	305	306	307	308	309	310	311	312	313	314	315
0	1	0	0	1	0	0	0	0	0	0	0	...

4.

$_X^T$ =

	355	356	357	358	359	360	361	362	363	364	365	366
0	1	1	0	1	0	0	1	0	1	0	1	...

X^T =

	355	356	357	358	359	360	361	362	363	364	365	366
0	1	1	0	1	0	1	1	0	1	0	1	...

5.

$_X^T$ =

	445	446	447	448	449	450	451	452	453	454	455	456
0	0	1	1	0	1	0	0	1	0	0	1	...

X^T =

	445	446	447	448	449	450	451	452	453	454	455	456
0	0	1	1	0	1	1	0	1	0	0	1	...

6.

$_X^T$ =

	506	507	508	509	510	511	512	513	514	515	516	517
0	1	0	1	1	0	1	1	0	0	0	1	...

X^T =

	506	507	508	509	510	511	512	513	514	515	516	517
0	1	0	1	1	0	0	1	0	0	0	1	...

7.

$_X^T$ =

	665	666	667	668	669	670	671	672	673	674	675	676
0	1	0	0	0	0	0	0	1	0	0	0	...

X^T =

	665	666	667	668	669	670	671	672	673	674	675	676
0	1	0	0	0	0	1	0	1	0	0	0	...

Рис. 9. Демонстрація штучно введених помилок в цифровий потік

Процеси, що виконуються на приймальній стороні

Під час декодування виконується процес пошуку неправильних біт та їх виправлення, а також видалення контрольних розрядів. У результаті отримаємо цифрову послідовність керувальної інформації довжиною 640 біт, як і до процесу кодування (показано перші 10 біт), зображено на рис. 10. Під час проходження процесу декодування всі 8 штучно введених помилок були виправлені, про що свідчать результати порівняння послідовностей отриманих після кодування, до введення помилок, і декодування.

$DKOD := Vec(DEKOD)^T =$

	1	2	3	4	5	6	7	8	9	10	11
1	0	1	0	0	0	1	1	0	1	0	...

Рис. 10. Цифровий потік після декодування

Порівняння виконано, за допомогою вище згаданої функції, результат показано на рис. 11.

$$Compare(C33, T) = 0$$

Рис. 11. Результат порівняння кодованих і декодованих даних

Як видно з рис. 11, ці два потоки повністю збігаються, а отже, це свідчить, що модель декодера виконала свою роль і виправила всі 8 помилок, що були випадковим чином розміщені в цифровому потоці, а також коректно видалила перевірочні символи й правильно розмістила блоки вихідних даних.

У результаті після операцій кодування/декодування, модуляції/демодуляції отримали такі самі дані, як і після процесу шифрування, незважаючи на помилки і маніпуляції з кінцевими бітами. Для отримання цифрового потоку сигналу керування ці дані потрібно розшифрувати.

Розшифрування відбувається за тим самим стандартом ГОСТ 21847-89, а сама процедура є зворотною за послідовністю виконання операцій. При розшифруванні також виконується 32 раунди, де в кожному наступному раунді розшифрується результат розшифрування попереднього раунда.

На рис. 12 показана цифрова послідовність, отримана після проходження 32 раундів розшифрування (показано перші 12 біт), що є ідентичною рис. 2 (вхідним цифровим потоком даних від апаратури керування).

$C := Vec(C1)^T =$

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	1	1	1	1	1	1	1	1	1	...

Рис. 12. Цифровий потік отриманий після дешифрування

Після більш детального огляду цього потоку і його порівняння з вхідними даними отримаємо дані показані на рис. 13.

$$Compare(VhPos1, T) = 0$$

Рис. 13. Результат порівняння вхідного потоку і цифрового потоку після дешифрування

Згідно з результатами порівняння вхідного цифрового потоку з апаратури керування БПЛА й цифрового потоку, отриманого після вище описаних операцій шифрування/розшифрування, кодування/декодування, модуляції/демодуляції в результаті маємо таку саму цифрову послідовність сигналу керування, що була отримана після аналогово-цифрового перетворення. Для порівняння даний потік можна візуалізувати і представити у вигляді графіка функції, що й показано на рис. 14.

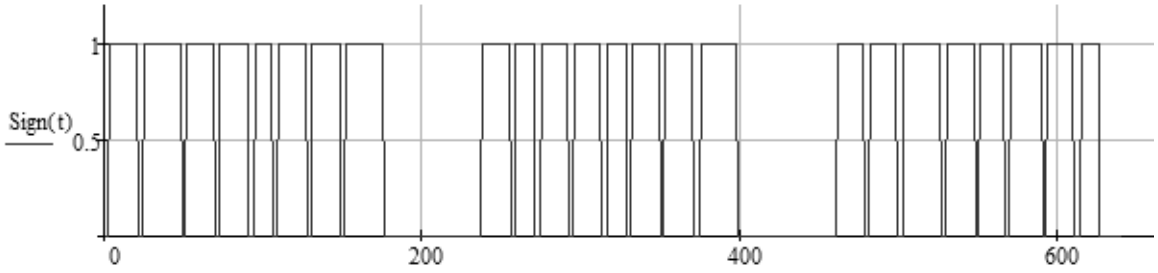

Рис. 14. Графічне зображення вихідного цифрового потоку

Одразу можна помітити його ідентичність до імітації сигналу з керуючої апаратури. Також неозброєним оком можна побачити, що другий і восьмий імпульс у першому періоді розширені, що свідчить про положення джойстика «униз», а звужений п'ятий імпульс про положення джойстика «угору» , це справедливо також і для другого й третього імпульсів.

Отже, це свідчить про те, що після всіх операцій з отриманим цифровим потоком від апаратури керування він залишився незмінним і БПЛА правильно б сприйняв отримані команди керування.

Як доказ даного твердження, подано графічне зображення на якому показано зіставлення вхідного і вихідного потоків (рис. 15).

На рис. 15 суцільною лінією показано вихідний потік, тобто той, що отриманий після дешифрування, а пунктирною лінією — вхідний цифровий потік.

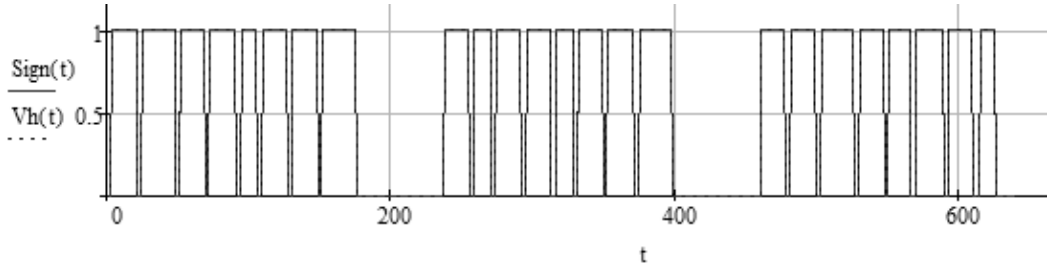

Рис. 15. Поєднане зображення вхідного і вихідного цифрових потоків

Висновки

У результаті проведеної роботи було розроблено комп'ютерну модель захищеного від несанкціонованих дій зловмисника каналу передавання сигналів керування БПЛА. Промодельовано процес створення цифрового потоку керуючої інформації у вигляді відео-імпульсів різної тривалості з широтно-імпульсною модуляцією від апаратури керування і його проходження через сертифікований в Україні блочний шифроалгоритм стандарту ГОСТ 28147-89, а також уведення завадостійкого кодування за допомогою коду Хеммінга (31,26). Як модуляцію було використано QAM-16.

Підтверджено працездатність розробленої моделі: у процесі моделювання на виході моделі отримано таку ж цифрову послідовність, що генерувалася на вході. Отже, це свідчить про коректність реалізації шифроалгоритму та схеми завадостійкого кодування. У процесі дослідження також було перевірено коректуючу здатність вище зазначеного коду шляхом внесення в послідовність 8 бітових помилок у випадково обрані позиції, що розташовані одна від одної на довжині блока коду (31 біт). У результаті декодування було усунено всі 8 помилок, що підтвердило очікуваний результат. Розроблена модель може бути використана для досліджень ефективності захисту інформації у каналі керування безпілотними літальними апаратами.

ЛІТЕРАТУРА

1. *Давиденко А. Н.* Анализ вопросов закрытия информационного канала связи с беспилотным летательным аппаратом / А. Н. Давиденко, С. Я. Гильгурт, А. С. Потенко, А. К. Евдина // Зб. наук. пр. ІПМЕ НАН України. — К., 2014. — Вип. 71. — С. 61–64.

2. *Системы* обработки информации. Защита криптографическая. Алгоритм криптографического преобразования: ГОСТ 28147 89. — М. : Госстандарт СССР, 1989. — 26 с.

3. *Конахович Г. Ф.* Теорія електричного зв'язку / Г. Ф. Конахович, І. О. Мачалін, О. Ф. Пузиренко. — К. : ТОВ «НВП Інтерсервіс», 2013. — 368 с.

4. *Експлуатація* телекомунікаційних систем / Г. Ф. Конахович, В. М. Чуприн, І. О. Мачалін, О. П. Ткаліч. — К. : Центр учбової літератури, 2014. — 372 с.

5. *Бахтіяров Д. І.* Аналіз мультиплексованого сигналу ШІМ на виході обладнання керування БПЛА FlySky CT-6A та методологічні підходи до побудови захищеного каналу керування БПЛА / Д. І. Бахтіяров, К. О. Марук // Проблеми розвитку глобальної системи зв'язку, навігації, спостереження та організації повітряного руху CNS/ATM: тези доповідей науково-технічної конференції; Київ, 17–19 листопада. — 2014. — С. 44.

6. *Бахтіяров Д. І.* Аналіз ефективності комплексного застосування заходів завадозахищеності для підвищення стійкості функціонування засобів керування БПЛА / Д. І. Бахтіяров, І. О. Козлюк // Проблеми розвитку глобальної системи зв'язку, навігації, спостереження та організації повітряного руху CNS/ATM: тези доповідей науково-технічної конференції; Київ, 17–19 листопада. — 2014. — С. 50.

7. *Луцький М. В.* Аналіз існуючих систем завадостійкого кодування та їх порівняльна характеристика за часом затримки сигналу / М. В. Луцький, Д. І. Бахтіяров // Проблеми розвитку глобальної системи зв'язку, навігації, спостереження та організації повітряного руху CNS/ATM: тези доповідей науково-технічної конференції; Київ, 17–19 листопада. — 2014. — С. 53.

8. *Моделювання* захищеного каналу керування безпілотним літальним апаратом / Г. Ф.Конахович, І. О. Козлюк, Д.І. Бахтіяров, М. В. Луцький // Безпека інформації в інформаційно-телекомунікаційних системах: матеріали XVI Міжнародної науково-практичної конференції; Київ, 26–28 травня. — 2015. — С. 45–48.

9. *Юдін О. К.* Захист інформації в мережах передачі даних: підручник / О. К. Юдін, О. Г. Корченко, Г. Ф. Конахович. — К. : ТОВ «НВП Інтерсервіс», 2009. — 716 с.

Стаття надійшла до редакції 23.06.2015