УДК 681.3

Віктор МЕЛЬНИК
vamlnk2015@gmail.com
ORCID: 0000-0001-6981-5046
м. Миколаїв

МІКРОПРОЦЕСОРНА СИСТЕМА ДЛЯ КОДУВАННЯ ЗВУКОВОГО СИГНАЛУ

У статті розглядаються методи апаратного і програмного шифрування звукового сигналу, програми (скетчі) дискретизації сигналів та дискретних перетворень для мікропроцесора платформи Ардуино.

Програмне забезпечення системи складається з модулей дискретизації сигналу, його накопичення в буфері пам'яті, скетчами, які реалізують скремблювання – дескремблювання сигналу та скетчами, які реалізують зв'язок мікроконтролеру з зовнішніми пристроями

У статті зроблено вибір апаратного та програмного забезпечення цифрового пристою кодування сигналу на основі ефективних алгоритмів отримання спектра сигналу і створено відповідне програмне забезпечення для обробки сигналів. Програмне забезпечення виконано на мові С++ в середовищі Arduino IDE.

***Ключові слова**: мікропроцесор, скремблювання, скетч, дискретизація сигналу, алгоритм, програмне забезпечення.*

Постановка проблеми

У сучасних системах зв'язку застосовуються складні методи перетворення сигналів, спрямовані на підвищення надійності передачі інформації та захисту її від несанкціонованого доступу. Одним з таких методів є метод скремблювання (scramble - перемішування) сигналу [1].

Скремблювання – це оборотне перетворення цифрового потоку без зміни швидкості передачі з метою отримання властивостей, близьких до властивостей випадкової послідовності. Оригінал повідомлення можна відновити, застосувавши зворотний алгоритм. Стосовно до телекомунікаційних систем скремблювання підвищує надійність синхронізації пристроїв, підключених до лінії зв'язку, і зменшує рівень перешкод, які випромінюються на сусідні лінії багатожильного кабелю. Є й інша область застосування скремблерів – захист інформації, що передається, від несанкціонованого доступу.

Скремблювання широко застосовується в багатьох видах систем зв'язку для поліпшення статистичних властивостей сигналу. Зазвичай скремблювання здійснюється на останньому етапі цифрової обробки безпосередньо перед модуляцією.

Цифрова обробка сигналів в широкому сенсі цього поняття означає виконання різних операцій над одновимірними і багатовимірними сигналами. До одновимірних відносять телефонні та радіосигнали, до багатовимірних – телевізійні сигнали, фотографії дослідницького характеру, медичні рентгенограми, електронно-мікроскопічні фотографії молекул, радіо- і звуколокаційні карти, дані томографії та інші цілі, переслідувані при обробці таких сигналів [2].

Аналіз останніх досліджень і публікацій

Найсучасніше звукове обладнання, MP3-плеєри та мобільні телефони використовують цифрову обробку сигналів для частотної корекції, керуючи відносної потужністю високих і низьких частот при відтворенні музичних творів. Однак іноді не потрібно виводити змінену версію вхідного сигналу, коли цифрова обробка потрібна, тільки щоб прибрати з сигналу небажані перешкоди і тим самим отримати від датчика більш точне значення.

Цифрова обробка сигналів як напрямок розвитку науки і техніки зародилася в 1950 - х роках і спочатку являла собою досить екзотичну галузь радіоелектро-ніки, практична цінність якої була далеко не очевидною. Проте за минулі сімдесят років завдяки успіхам мікроелектроніки системи цифрової обробки сигналів не тільки втілилися в реальність, але і увійшли в наше повсякденне життя у вигляді CD- і DVD-програвачів, модемів, телефонів і багато чого іншого. Значною мірою це сталося в аудіотехніці, інтенсивно йде процес переходу телевізійного мовлення на цифрову основу [3].

Розвиток нової точки зору на цифрову обробку сигналів було прискорено відкриттям в 1965 р. ефективних алгоритмів для обчислень перетворень Фур'є. Цей клас алгоритмів став відомий як швидке перетворення Фур'є (ШПФ, fast Fourier transform). Алгоритм швидкого перетворення Фур'є зменшив час обчислення перетворення Фур'є на кілька порядків. Це дозволило створити дуже складні алгоритми обробки сигналів у реальному часі. Крім того, з урахуванням можливостей дійсної реалізації алгоритму швидкого перетворення Фур'є на спеціалізованому цифровому пристрої, багато алгоритмів обробки сигналів, що були раніше непрактичними, стали знаходити втілення на спеціалізованих пристроях[4].

Постановка завдання

Метою цієї статті є розгляд проекту універсального лабораторного стенду (УЛС) для демонстрації ефекту скремблювання сигналу і передачі його в цифровому вигляді в ПК для подальшого дескремблювання.

Для цього треба розглянути методи апаратного і програмного скремблюванн на базі мікропроцесорів, програми (скетчі) дискретизації сигналів та дискретних перетворень для мікропроцесора платформи Arduino. Ці програми застосовуються для скремблювання і дескремблювання сигналу.

Виклад основного матеріалу

Arduino є електронною платформою, призначеною для швидкої збірки автоматичних пристроїв різного ступеня складності. В основі цього пристрою знаходяться мікропроцесорні модулі, датчики, а також інтерфейси до них. Різноманітні модифікації цієї платформи дозволяють вибрати мікропроцесор, який здатен робити визначені вище операції над дискретним сигналом у реальному часі. У запровадженні даного пристрою в прикладних задачах велике значення має комфортне середовище розробки. Кращим рішенням буде застосування C++. Ця програма дозволить значно спростити створення проектів по прошивці мікропроцесора.

Система, яка спроектована, може бути застосована у навчальному процесі для демонстрації процесу скремблювання та для наукових досліджень по удосконаленню його шляхом використання складних за своєю дією алгоритмів та пристроїв.

Ідея скремблювання заснована на тому, що виконане двічі складання по модулю 2 переданого символу з іншим символом не призводить до його зміни, проте в лінію замість послідовності X_1 передається послідовність Z, що має більшу кількість одиниць в порівнянні з вихідною послідовністю.

Тут мається на увазі наступна властивість логічної операції складання по модулю 2. Хай вихідний сигнал S формується за правилом $S=X\oplus P$, тоді після дескремблювання $S\oplus P=(X\oplus P)\oplus P=X$, тобто відбувається відновлення сигналу, бо $P\oplus P=0$.

Псевдовипадкова послідовність (ПСП) символів може бути отримана за допомогою застосування логічної операції над вже отриманними символами сигналу, які зберігаються в буфері. Так відома схема $B_i=X_i\oplus B_{i-1}\oplus B_{i-3}$

Генерація послідовності, що кодує біти, виробляється циклічно з невеликого початкового обсягу інформації – ключа за наступним алгоритмом. З поточного набору біт вибираються значення певних розрядів і додаються по модулю 2 між собою.

Всі розряди зсовуються на 1 біт, а тільки що отримане значення («0» або «1») ставиться в молодший розряд, що звільнився. Значення, що перебувало в найстаршому розряді до зсуву, додається в кодувальну послідовність, стаючи черговим її бітом.

З теорії передачі даних криптографія запозичила для запису подібних схем двійкову систему запису. За нею зображений на рисунку скремблер записується комбінацією «100112» – одиниці відповідають розрядам, з яких знімаються біти для формування зворотного зв'язку.

Принцип дії такого скремблеру наведено на рис. 1.

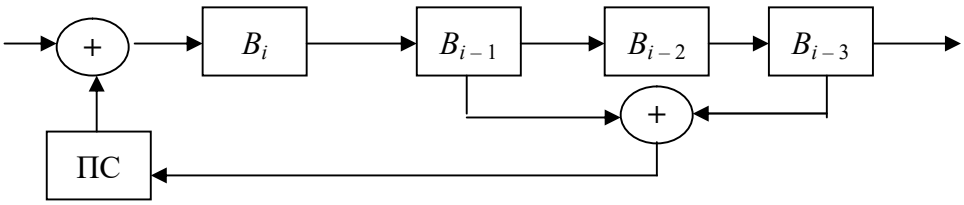

Рис. 1. Принцип дії апаратного скремблеру

Складовими частинами апаратної частини, яка проектується на базі платформи Arduino є: мікроконтролер з вбудованим АЦП, джерело сигналу, наприклад мікрофон, або потенціометр, інтерфейси (як досить спеціалізовані I2C, SPI, CAN, так і поширені COM-порт, USB, Bluetooth, WiFi, Ethernet і ін.), мікроконтролер для прийому сигналу з вбудованим ЦАП сигналу, пристрій для демонстрації дескремблюванного сигналу.

Якою б складною не була процедура скремблювання, найменший елемент, з яким вона оперує, це перетворений фрагмент сигналу, який можна зробити коротше якогось певного інтервалу через інтерференційні явища при передачі в каналі.

Основними властивостями скремблера є:

1) низька складність реалізації;

2) висока якість відновленої мови;

3) наявність залишкової інформації в закритому сигналі, яка може бути використана конкуруючої стороною.

Цифрові системи закриття мови не передають будь-якої частини початкового мовного сигналу як це роблять аналогові. Мовні компоненти кодуються в цифровий потік даних, який в подальшому змішується з псевдовипадковою послідовністю по одному з криптографічних алгоритмів. Отримане таким чином закрите мовне повідомлення передається за допомогою модему в канал зв'язку, на іншому кінці якого робляться зворотні перетворення з метою отримання відкритого мовного сигналу. Такі системи називають кодерами – це процедури, що представляють мовний сигнал моделлю; параметри моделі, що змінюються в часі, шифрують як потік даних і передають за допомогою модемів [3].

Основними властивостями кодеров є:

– велика складність реалізації, як правило, на основі цифрових сигнальних процесорів (DSP);

– якість відновленої мови визначається швидкістю передачі даних в каналі і складністю моделі;

– принципова відсутність залишкової будь-якої інформації в закритому сигналі, будь-який алгоритм шифрування даних створює некорельований потік даних, виключає статистичні залежності між закритим і відкритим уявленнями сигналу.

Основними характеристиками будь-якої криптосистеми є її максимальна безпека і продуктивність. Так, система RSA працює приблизно в тисячі разів повільніше ніж скремблирование і вимагає, щоб ключі були дуже довгі. Використання систем з відкритим ключем може бути обмежено завданням обміну ключами з подальшим їх застосуванням в класичній криптографії, тобто використання так званих гібридних систем [1].

У скремблері алгоритм захисту не є ізольованим, а закладений в сам алгоритм

перетворення сигналу. Шифрування тут полягає в формуванні генератором псевдовипадкової послідовності бітів, що визначають значення індексів в алгоритмі перетворення мовного сигналу.

У розробленному алгоритмі ключем можна вважати число, що використовується для отримання коду бітів і так як це число можна змінювати кожного разу при встановленні з'єднання, то можна бути впевненим, що це додасть більше труднощів зловмисникам при спробі отримання доступу к даним.

Останнім часом сфера застосування скремблюючих алгоритмів значно скоротилася, що пояснюється в першу чергу зниженням обсягів побітної послідовної передачі інформації, для захисту якої були розроблені дані алгоритми. Головною проблемою шифрів на основі скремблерів є синхронізація передавального (кодуючого) і приймального (декодуючого) пристроїв.

При пропуску або помилковому вставлянні хоча б одного біта вся інформація, що передається необоротно втрачається. Тому в системах шифрування на основі скремблерів дуже велику увагу приділяють методам синхронізації.

На практиці для цих цілей зазвичай застосовується комбінація двох методів:

а) додавання в потік інформації синхронізуючих бітів, наперед відомих приймальній стороні, що дозволяє їй при незнаходження такого біта активно почати пошук синхронізації з відправником;

б) використання високоточних генераторів тимчасових імпульсів, що дозволяє в моменти втрати синхронізації виробляти декодування прийнятих бітів інформації «по пам'яті» без синхронізації [4].

Взагалі платформи Arduino не найкращі пристрої для цифрової обробки сигналів. Вони не здатні приймати аналоговий сигнал, введений з високою швидкістю, а їх аналогові виходи обмежені можливостями технології широтно-імпульсної модуляції (ШІМ). Виняток становить модель Arduino Due, яка має кілька АЦП, швидкий процесор і два справжніх ЦАП. Тобто модель Due володіє достатніми апаратними можливостями для того, щоб її можна було використовувати для оцифровки звукового стереосигнала і виконання якихось маніпуляцій з ним.

Висновки і перспективи досліджень

В результаті проведеного дослідження було розроблено метод скремблювання звукових сигналів і алгоритм його реалізації на основі чисел Фібоначчі з метою захисту аудіо даних від несанкціонованого підслуховування. Проаналізовано властивості і критерії оцінки системи приховуванні інформації, описано принцип дії алгоритму скремблювання

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Зегэнда, Д.П. Защита информации в компьютерных системах [Текст] / Д.П. Зегэнда, А.М. Ивашко. – С.- Пб.: СПб ГТУ, 1992. – 100 с.
2. Белов, А.В. Микроконтроллеры AVR в радиолюбительской практике [Текст] / А.В. Белов. – М.: Русский язык, 2007. – 352 с.
3. Сергієнко, А.Б. Цифрова обробка сигналів: Підручник для ВУЗів [Текст] / А.Б. Сергієнко. – СПб.: Пітер, 2006. – 751 с.
4. Богданов, Е.В. Схемы кодирования и правила декодирования информации [Текст] / Е.В. Богданов, Т.Х. Вьонг, Т.Т. Нгуен, В.С. Давыдов // Изв. Вузов России. Радиоэлектроника. – 2014. – №3. – С. 27-33.

Victor MELNIK
Mykolaiv

MICROPROCESSOR SYSTEM FOR COVERING SOUND SIGNAL

The article deals with methods of hardware and software encryption of a sound signal, programs (sketches) of discretization of signals and discrete transformations for the Arduino platform processor.

The software of the system consists of signal sampling modules, its accumulation in the memory buffer, and skeletons that implement scrambling - descrambling the signal and sketches that implement the microcontroller's communication with external devices

The article chooses the hardware and software of the digital device for encoding a signal based on efficient algorithms for obtaining the spectrum of the signal, and the corresponding software for signal processing is created. The software is run in C++ in the Arduino IDE environment.

Keywords: *microprocessor, scrambling, sketch, signal sampling, algorithm, software.*

Виктор МЕЛЬНИК
Николаев

МІКРОПРОЦЕСОРНА СИСТЕМА ДЛЯ КОДУВАННЯ ЗВУКОВОГО СИГНАЛУ

В статье рассматриваются методы аппаратного и программного шифрования звукового сигнала, программы (скетчи) дискретизации сигналов и дискретных преобразований для процессора платформы Ардуино.

Программное обеспечение системы состоит из модулей дискретизации сигнала, его накопления в буфере памяти, скетчами, которые реализуют скремблирования - дескремблирование сигнала и скетчами, которые реализуют связь микроконтроллера с внешними устройствами

В статье сделан выбор аппаратного и программного обеспечения цифрового устройства для кодирования сигнала на основе эффективных алгоритмов получения спектра сигнала и создано соответствующее программное обеспечение для обработки сигналов. Программное обеспечение выполнено на языке С ++ в среде Arduino IDE.

Ключевые слова: *микропроцессор, скремблирования, скетч, дискретизация сигнала, алгоритм, программное обеспечение.*

Стаття надійшла до редколегії 25.10.2018