ГОДИШНИК

НА ШУМЕНСКИЯ УНИВЕРСИТЕТ „ЕПИСКОП КОНСТАНТИН ПРЕСЛАВСКИ”

ТЕХНИЧЕСКИ НАУКИ

Т. I Е

ANNUAL

OF KONSTANTIN PRESLAVSKY UNIVERSITY OF SHUMEN

Vol. I E

FACULTY OF TECHNICAL SCIENCES

Университетско издателство
„Епископ Константин Преславски”
2010

Настоящият годишник съдържа доклади,
изнесени на
годишната научна конференция на
Факултета по технически науки през 2009 г.

Отговорен редактор: *доц. Атанас Атанасов*
Ред. колегия: *доц. Пламен Чернокожев*
доц. Пламен Михайлов

ISSN 1311-834X

Съдържание

ГЕНЕРИРАНЕ НА СИГНАЛИ С LabVIEW 6.8

Илиян Илиев

Абстракт. *Показани са четири програми на LabView 8.6 за моделиране на базови импулси използвани при изследване на цифрови устройства. Същите може да се използват като подпрограми на програми симулиращи работата на отделни канали в радарни, сонарни и комуникационни системи.*

Ключови думи: *DSP, LabView, делта импулс, линейно изменящ се импулс, правоъгълен импулс, импулс с линейна честотна модулация, G код.*

GENERATING SIGNALS WITH LabVIEW 6.8

Iliyan V. Iliev

Abstract. *The four LabView 8.6 programs simulating base signals used in digital systems are shown. They can be used as subprograms in programs modeling some channels in real radar, sonar or communication systems.*

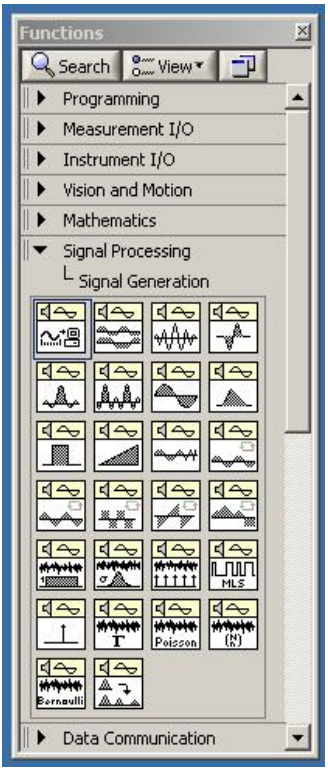
Key words: *DSP, LabView, delta impulse, ramp impulse, step impulse, FM impulse, G code.*

1. Въведение.

В статията са описани виртуални инструменти, които генерират основните сигнали използвани в различни радарни, сонарни и комуникационни системи. Симулирани са част от базовите сигнали, които се използват в цифровата обработка на сигнала (digital signal processing – DSP) и цифровите системи. Програмната среда LabView се доставя с вградени в нея функции за генериране на тези сигнали както софтуерно така и хардуерно. Отворената архитектура на средата позволява поддържането на хардуерни сигнални генератори както на National Instruments, така и на други фирми, които ги доставят с драйвери за LabView.

2. Генериране на основни функции.

Три от най-широко използваните сигнали в DSP са делта импулс, линейно изменящ се импулс и единичен импулс. На фигура 2.1 е показано местоположението на функционалната палета, където са разположени виртуалните инструменти на различни сигнални генератори на LabView 8.6.

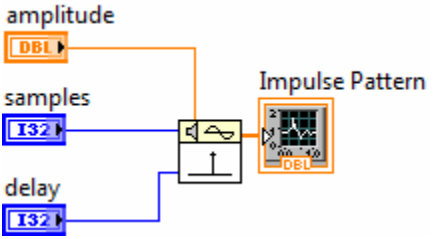


Фиг. 2.1. Функционална палета на VIs на сигнални генератори.

Дискретната делта функция се описва с формула (1) и се използва като входен сигнал за определяне на коефициентите на филтър във верига за филтрация или за генериране на шум с равномерен спектър. [2]

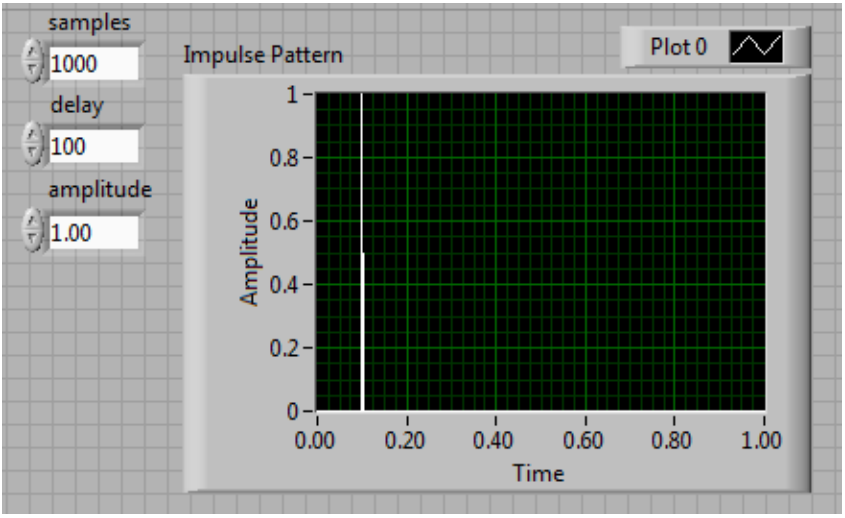
(1) $$\delta(n - n_d) = \begin{cases} 1 & n = n_d \\ 0 & n \neq n_d \end{cases}$$

За генерирането ѝ в LabView се използва ImpulsePattern.vi, който притежава три входни параметъра, брой на сампълите, амплитуда на делта импулса и закъснение. На фигура 2.2 е показан G кода на програма, която генерира и изобразява на графичен дисплей делта импулс с амплитуда 1 и закъснение 0.1 sec при честота на дискретизация 1 kHz.



Фиг. 2.2. G код за генериране и изобразяване на делта импулс.

Интерфейса на програмата и резултатът от нейната работа е показан на фигура 2.3.



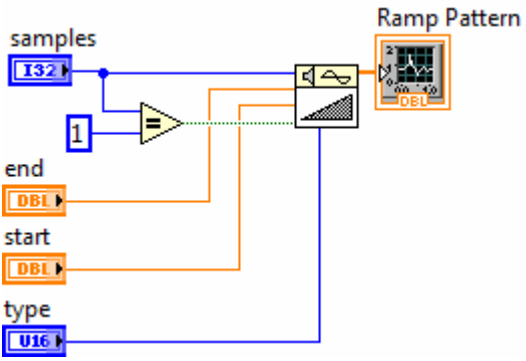
Фиг. 2.3. LabView програма за генериране на делта импулс.

Линейно изменящ се импулс в радарните и сонарни системи се използва в каналите за управление и в каналите за формиране на развивките на индикаторните устройства. Импулсът се синтезира от виртуалния инструмент Ramp Pattern.vi и в дискретна форма се описва с формулата,

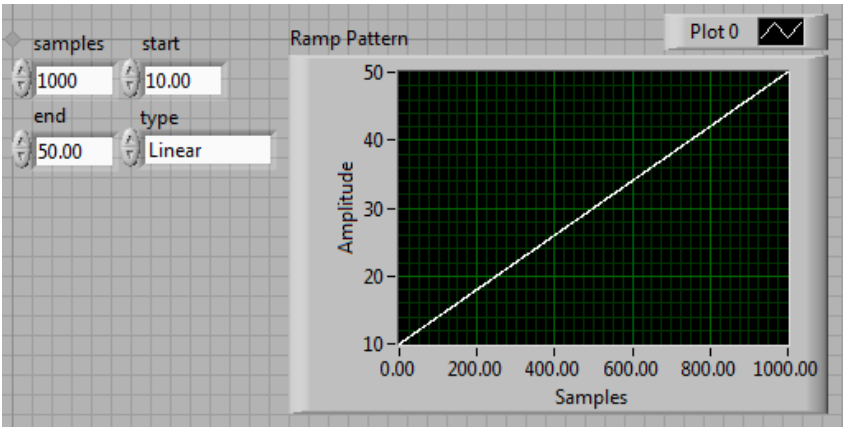
(2) $$y_i = y_0 + i\Delta y$$

за i= 0, 1, 2, … , n-1

където, y_0 е началната амплитуда – start, y_i е амплитудата на импулса в дискретния момент i. Δy= (start – end)/n е стъпката, с която се изменя амплитудата, където end е крайната амплитуда, а n броя на сампълите. G кодът на програмата синтезираща линейно изменящ се импулс е показан на фигура 2.4, а интерфейсът й на фигура 2.5.



Фиг. 2.4. G код на линейно изменящ се импулс – Ramp Pattern.



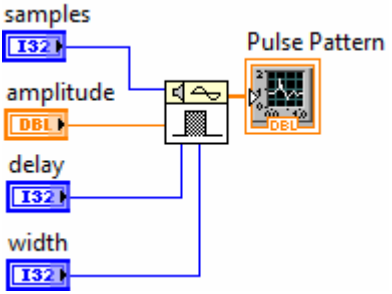
Фиг. 2.5. Интерфейс на програма генерираща линейно изменящ се импулс.

Единичният импулс в LabView 8.6 се синтезира от виртуалния инструмент Pulse Pattern.vi и се описва в дискретна форма с равенството,

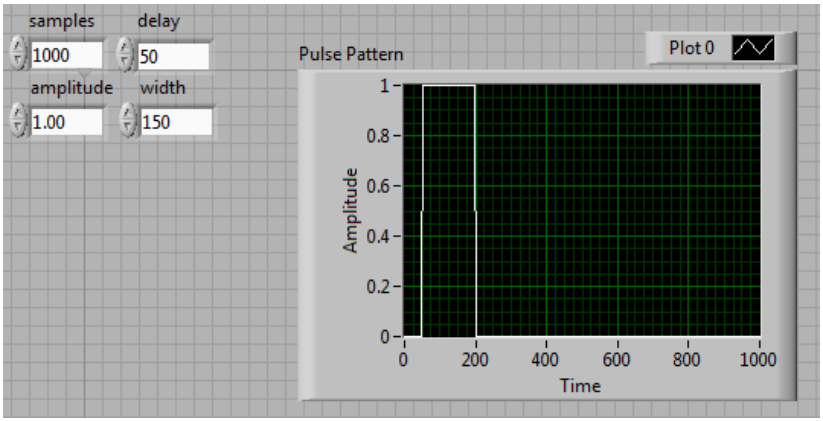
(3) $$y_t = \begin{cases} a & d < t < d + w \\ 0 & \text{за всички останали случай} \end{cases}$$

където, d е закъснението на сигнала, w дължината на импулса.

Виртуалният инструмент има четири входни параметъра, брой на сампълите, амплитуда, закъснение и дължина на импулса. На фигура 2.6 е показан G кода на програмата синтезираща импулс с дължина 150 отчета и закъснение 50 отчета при брой на сампълите 1000.



Фиг. 2.6. G код синтезиращ правоъгълен импулс.



Фиг. 2.7. Интерфейс на програма за генериране на правоъгълен импулс.

3. Генериране на честотно модулиран (FM) сигнал.

FM сигналът се отнася към групата на сложните сигнали. В радарните и сонарни системи широката база на сигнала позволява да се постигане висока разрешаваща способност по разстояние и максимални дистанции на откриване. В сонарните системи сигналът се използва за откриване на обекти в условия на силна реверберация.[3]

Честотно модулираният сигнал се описва с формулата, [1]

(3) $$s(t) = \sqrt{\frac{2E_t}{T}}\, a(t)\cos[2\pi f t + \varphi(t)]$$

където $\varphi(t) = 2\pi\mu\frac{t^n}{2}$;

E_t енергията на сигнала, T продължителност на сигнала, a(t) коефициент на амплитъдна модулация, φ(t) използваната фазова модулация, μ скорост на изменение на честотата, f централна честота.

В LabView 8.6 за генериране на FM импулс се използва виртуалния инструмент Chirp Pattern.vi, който има следните входни параметри, брой на сампълите, амплитуда, начална и крайна честоти в нормализирани единици cycles/sample. За моделиране на импулса се използва дискретното представяне на формула (3),

(4) $$y_i = A * \sin((0.5 * a * i + b)i)$$

за i= 0, 1, 2, … ,(n-1);
където A е амплитуда,

a= $2\pi(f_2 - f_1)/n$,

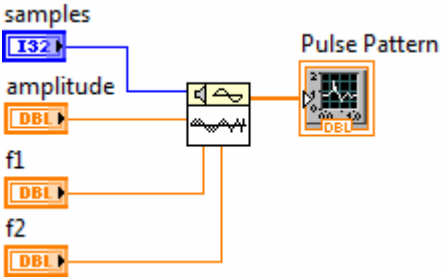
b= $2\pi f_1$,

f_1 е началната честота в нормализирани единици,

f_2 е крайната честота в нормализирани единици,

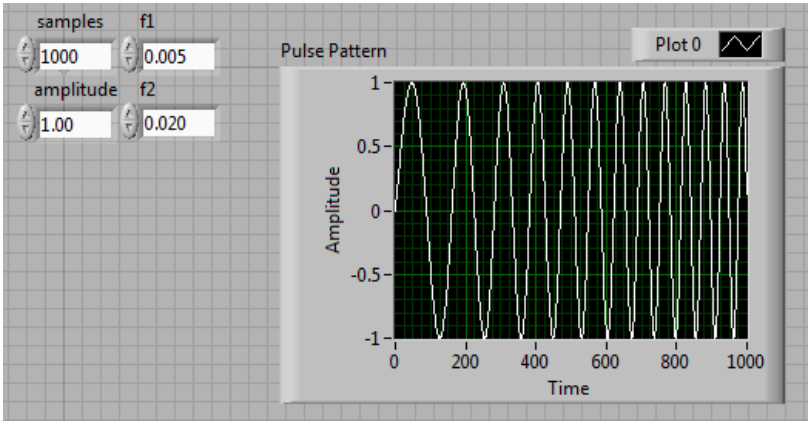
n е броя на сампълите.

На фигура 3.1 е показан G кода на програма генерираща FM сигнал, който в началото извършва 5 цикъла за секунада, а в края 20 цикъла за секунда.



Фиг. 3.1. G код на генератор на FM импулс в LabView 8.6.

Интерфейсът на програмата е показан на фигура 3.2.



Фиг.3.2. Интерфейс на програма за генериране на FM импулс.

В заключение от сравнението на показаните четири програми се вижда, че те са подобни като код и интерфейс. Съчетано с графичния код за програмиране това прави програмната среда LabView много лесна и удобна за симулиране и изследване на работата на цифрови устройства.

Литература:

1. Alan Oppenheim, “Applickation of digital signal processing”, Prentice hall, Inc., Englewood cliffs, New Jersey, 1978.
2. Cory L. Clarck, “LabView Digital Signal Processing and Digital Communications”, McGraw-Hill, Ney York, 2005.
3. Robert J. Urick, “Principles of underwater sound”, McGraw-Hill Book Company, Ney York, 1975.

ПРОВЕРКА ВЪЗМОЖНОСТИТЕ ЗА РЕГИСТРАЦИЯ ПРЕМИНАВАНЕ НА МЕТАЛНО ТЯЛО ПРЕЗ ТРЪБА

Христо Христов
Тихомир Трифонов

Anotation: A possibilities of registration passing of an iron body through a metal tube by a simple sensor was investigate in a present paper.

1. Основни хипотези и допускания.

Преминаването на метално тяло през тръба може да се регистрира с различни датчици, основаващи се на различни физични явления и процеси.

- магнитни;
- електромагнитни;
- вибрационни;
- други.

При използването на датчици, работещи на принципа регистрация само на магнитно поле, то се налага металното тяло да бъде източник на такова магнитно поле, т.е. то трябва да е намагнетизирано. Това създава определени проблеми и затова регистрацията с датчици, регистриращи са магнитно поле не е подходяща.

Използването на вибрационни датчици също не е удачно, защото при движението си в тръба, металното тяло създава множество трептения, с най-различни честоти. В този случай е трудно, в повечето случаи невъзможно, да се определи (регистрира) момента на преминаване на тялото покрай датчика.

Ето защо в тази работа е избран подходът за регистрация преминаването на метално тяло през тръба посредством електромагнитни датчици.

2. Описание на опитната установка.

Опитната установка се състои от:

а) метална тръба с диаметър ½ цола и дължина 1.5 м – виж фиг.1;

Фиг.1

б) намотка (соленоид) съдържащ 300 намотки, от меден лакиран проводник с диаметър 0.2 мм

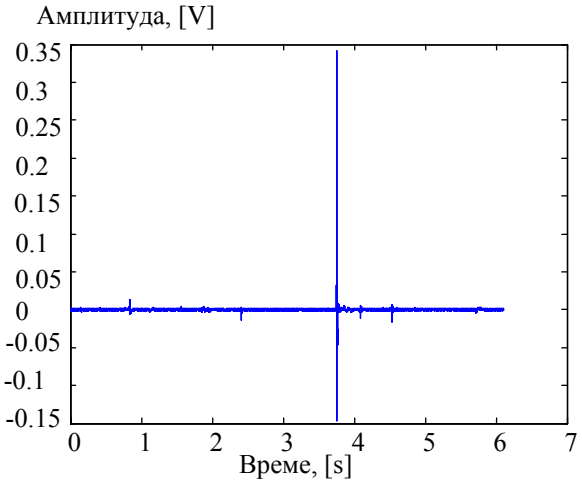
в) Соленоидът е фиксиран за тръбата посредством черна залепваща изолираща лента (изолирбанд) – виж фиг.2;

Фиг.2

Изходните проводници от соленоида са свързани посредство екраниран кабел и получения електрически сигнал се подава на микрофонния вход на персонален компътър.

3.**Експериментални данни.**

Получените експериментални данни са показни на фигурите по-долу.

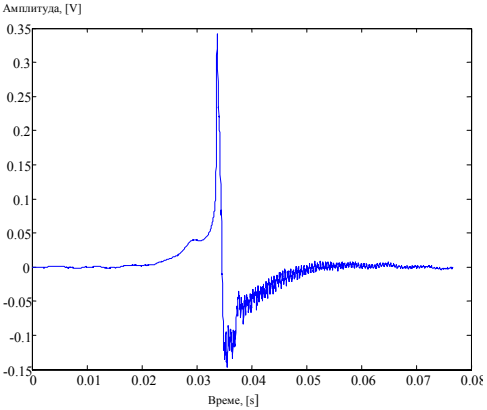


Фиг. 3

На фиг. 3 е показано експериментална графика на преминаващо метално топче през метална тръба. Между 3 (трета) и 4 (четвърта) сек. се наблюдава ясно изразен пик, когато топчето преминава през участъка от тръбата, където е навит соленоидът.

На фиг. 4 е показан характерът на генерираното напрежение. Вижда се как в началото напрежението нараства, достига максимум и след това скокообразно пада, достигайки отрицателни стойности.

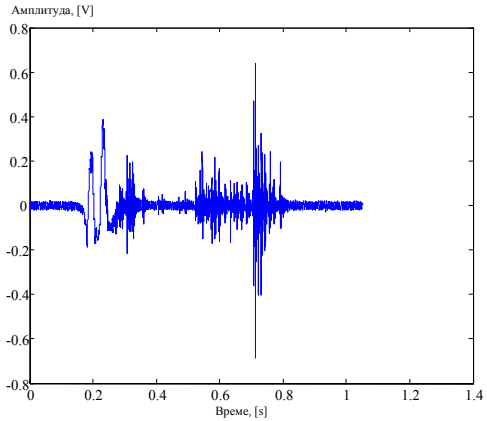
Това става в преиода между 0.03 и 0.04 сек. След това започва повишаване на напрежението – от отрицателни стойности, то достига до 0 волта. Наблюдава се и едно високочестотно



фиг.4

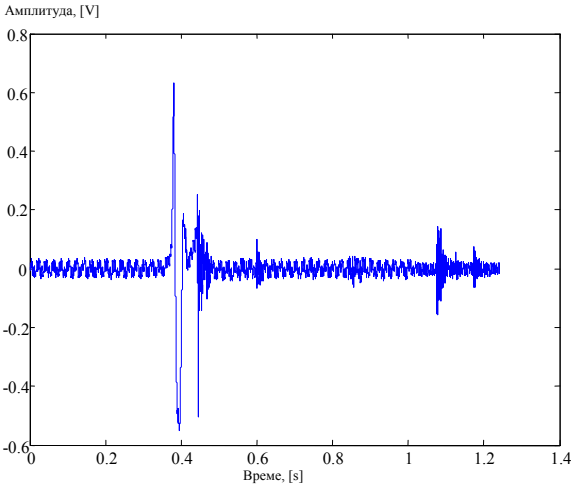
трептене, породено от търкалянето на топчето в тръбата в участъка след соленоида.

На фиг. 5 е показано преминаване на металното топче през тръбата в процес на търкаляне. Налице са няколко пика, като втория пик (при преминаване на топчето през участъка от тръбата с навит соленоид) е по силно изразен. Но в същото време се наблюдават - през целия процес на преминаване на тялото през тръбата, високочестотни трептения, породени от процеса на търкаляне. В този случай те играят ролята на смущения.



фиг.5

Подобни трептения се наблюдават и на фиг.6. Два ясно изразени пика, съпроводени с високочестотни трептения, породени от търкалянето на топчето в тръбата.



фиг. 6

4. Изводи и заключения.

А) Преминаването на метално тяло през метална тръба може да се регистрира с използването на опростен датчик - соленоид.

Б) Ако тялото изършва сложно движение – постъпателно съпроводено с търкаляне, се получават допълнителни високочестотни трептения. По отношение на основния сигнал в този случай те се явяват смущения.

В) За точната регистрация момента на преминаване е удачно да се ползва отрицателния фронт на сигнала. Необходимо е да се посочи критерий – например близост на сигнала до нулевата стойност, за да се определи момента на преминаване.

Г) Удачно е да се използва честотно преобразование и съответно честотен анализ – например Фурие преобразование, за да се получи основния храмоник и по него да се определи момента на преминаване.

АНАЛИТИЧЕН ОБЗОР НА ПРОТОКОЛ IP

Иван Цонев

Abstract: The paper offers The IP protocol of the TCP/IP model. Comparative analysis is done on the protocol functions and the protocol data units (PDU) for the versions 4 and 6.

Въведение - Протоколът IP (Internet Protocol) е от състава на протоколния стек TCP/IP, на базата на който се реализират компютърните комуникации в глобалната мрежа Интернет. Протоколът е разработен от д-р. Винт Сърф и д-р. Роберт Кан, като част от проект на американското министерство на отбраната. IP се използва за пръв път през 1969 г. в Advanced Research Projects Agency Network (ARPANET). След създаването си ARPANET бързо се разраства и след време се превръща в множество свързани американски мрежи и в Интернет. IP е основен и един от най-важните протоколи в съвременните компютъни мрежи. Той е протокол от мрежовото (network) ниво и е създаден някога за обмен на данни между различни компютърни мрежи. Основната му цел е да свързва хостове (станции, компютри) от различни мрежи.

Съобщенията се предават по мрежата, като се разделят на малки единици, наречени IP - пакети, които освен част от съобщението, съдържат адрес на подателя, адрес на получателя и други служебни данни, като например:

- номер на версия на протокола - 4 бита;
- информация, дали пакетът е част от по-голямо количество данни, съответно:
- DF (Don't fragment) - 1 бит;
- MF (More fragments) -1 бит;
- Fragment-Offset -13 бита;
- TTL - Time To Live или през колко инстанции на протокола може да премине даден пакет, преди да се счита за загубен - 8 бита;

- идентификатор на ползващия транспортен протокол - 8 бита.

Понастоящем широко се използва версия 4 на протокола, при която адресите са 32-битови числа и имат следния формат: xxx.xxx.xxx.xxx, където xxx е число между 0 – 255. Версия 4 носи със себе си редица ограничения, които ще бъдат преодоляни с новата версия 6. Протоколът не гарантира сигурното пристигане на съобщенията и не коригира грешките в пакетите. IP осигурява транспортните протоколи, като TCP и UDP. Мрежовият слой съставя маршрута за движение на пакетите от данни от един хост до друг хост, преминавайки през една или няколко свързани мрежи посредством рутери. Слоят разглежда пакетите като IP – дейтаграми. Протоколът има две основни функции:

- Фрагментация на съобщенията;
- Адресация на дейтаграмите.

Също той има и 3 спомагателни функции:

- Обработка на приоритетите;
- Класификация на безопасността;
- Разграничаване на TCP – сегментите.

Машините поддържащи слоя се делят на два вида – рутери и хостове. Рутерите имат два или повече интерфейса и изпращат пакети от някой интерфейс към друг, ако не са предназначени за тях самите. Хостовете са машини, които имат само един интерфейс и го използват за лични нужди. Пример за хост е всеки обикновен компютър, свързан към Интернет. Той от своя страна е свързан с рутера на Интернет доставчика. Предполага се, че мрежата се състои от свързани по между си по-малки мрежи (подмрежи). Адресната логика на протокола IP е йерархична. Адресното поле на протоколната спецификация е 32-битово, като са дефинирани 3 базови формата (класове) адреси A,B, и C. Първите 4 бита от адреса определят класа на мрежа. Останалите 24 бита се разделят на две части: мрежов идентификатор (network identifier - **netid**) и идентификатор на краен компютър (host identifier - **hostid**). Един типичен пример за клас A мрежа е ARPANET, а пример за клас C мрежа е една малка корпоративна мрежа в завод, училище, общинска администрация и т.н.

- Клас А - за големи подмрежи, първия байт на IP-адресите е от 1 до 126;
- Клас Б - за средно големи подмрежи, от 128 до 191 адреса;
- Клас С - малки мрежи, от 192 до 224.

1. **Анализ на протокола** - Интернет протоколът не поддържа управление на потока от данни и няма вградени средства, за да провери дали дадено изпратено съобщение е пристигнало успешно. При него единствено се използуват контролни суми на заглавната част, но не и за данните, които пренася дейтаграмата. Проверката дали данните са пристигнали успешно, управлението на потока и въобще надеждността на комуникацията се осигурява от протоколите от по-горните слоеве (за пример може да послужи протоколът от транспортния слой TCP). Програмните модули, които реализират Интернет протокола, присъстват във всеки хост, който участва в мрежата Интернет и във всеки маршрутизатор, който свързва мрежи. Тези модули използват общи правила за интерпретация на адресните полета и за фрагментиране и сглобяване на дейтаграми. Освен това, те разполагат и с процедури за вземане на решения за маршрутизация (това се отнася най-вече за маршрутизаторите).

Неделима част от IP е протоколът за контрол на съобщенията ICMP (Internet Control Message Protocol). Той осъществява няколко задачи в TCP/IP мрежите. Основното му предназначение е да съобщава на източника за невъзможност да бъде достигнато местоназначението и за грешки в маршрутизацията. Освен това, ICMP осигурява механизъм, който чрез съобщения “exo” и “отговор” тества дали даден възел в Интернет е достижим.

Предназначението на полетата на IP пакета (**datagram**) е следното:

1) ***Version*** - версия съдържаща текуща версия на IP протокола. Текущата версия е 4-та - IP v.4;

2) ***Header Length*** - дължина на заглавната част – съдържа действителния размер на заглавието на пакета, представен като брой 32-битови думи. Минималната дължина на пакета, ако не се използват допълнителните полета е пет 32-битови думи;

3) ***Type of Service*** - тип на услугата – поле, в което компютъра-източникът заявява тип на мрежовата услугата за текущата приложна сесия: висока надеждност; минимално времезакъснение; висока пропускателна способност; прилагане на схема с приоритети;

4) ***Total Length*** - пълна дължина на пакета (заглавната част и потребителските данни). Максимално допустимия размер на пакета е 65 535 бита;

(5) ***Identification*** - идентификация – обикновено едно съобщение се предава под формата на поредица от няколко пакета. Това поле служи идентификация за принадлежност на пакетите към Fragment offset - отместване на сегмента. Записва се информация за позицията на полето за данни от текущия фрагмент в оригиналния пакет. Отместването от началото на съобщението се представя като число кратно на 8 октета;

(6) ***Fragment Offset -*** отместване на сегмента. Записва се информация за позицията на полето за данни от текущия фрагмент в оригиналния пакет. Отместването от началото на съобщението се представя като число кратно на 8 октета;

(7) ***Time to Live -*** време на живот. Стойността се установява във възела-източник и се намалява при преминаването през всеки следващ мрежов възел. След достигане на стойност 0, пакетът се унищожава;

(8) ***Protocol -*** записва се типа на протокола от по-“високо” ниво, който се съдържа тялото на пакета. Използва се за протоколна идентификация в целевия възел;

(9) ***Header Check sum -*** контролна сума;

(10) ***Source IP address -*** IP адрес на източника (32 битово число);

(11) ***Destination IP address -*** IP адрес на получателя (32 битово число)**;**

1 – 11 са задължителни полета.

Опциите на IP протокола се отнасят до маршрутизирането на дейтаграмата. Те могат да бъдат от 0 до 10 32-битови думи. В днешно време те се използват рядко. Някои от тях са:

(12) ***Options -*** допълнителни възможности. Формата им зависи от приложението:

- за сигурност (Security) - полето за DATA от пакета е възможно да се криптографира. В Options се специфицира типа на кодирането;
- Source Routing - маршрутизиране от източника. В Options се описва маршрута и по този начин се изключва , маршрутизиращата логика ее възлите;
- Route Recording - запис на маршрута. В Options се записва информация за маршрута (трасиране на маршрута);
- Time Stamp - маркер за време – при желание да проверим времето за транспорт на пакета.

Първият байт на полето Options определя текущо използваната допълнителна възможност.

2. Сигурност на протокола - Сигурността е една от слабите страни на IP, въпреки че защитата, предлагана от VPN и IPSec, изглежда достатъчно стабилна. За двупосочните видеокомуникации в реално време, които са сред най-сложните за подсигуряване IP приложения, сега са разработени възможности за автентификация и криптиране чрез новия стандарт H.235. Целият трафик, който преминава през тунела между две точки във виртуалната частна мрежа, е криптиран. IPSec или протоколът за IP сигурност, представлява фамилия от отворени стандарти, които гарантират сигурни частни комуникации през Интернет. IPSec осигурява конфиденциалност, цялостност и достоверност на данните при преминаването им през обществените мрежи. Това е един основен компонент в

технологията, който е необходим за осигуряване на цялостно решение за сигурност. Този протокол може да адресира заплахите за сигурността в самата мрежова инфраструктура, без да се изискват скъпи модификации на хостове и приложения. IPSec осигурява криптиране и аутентикация на IP мрежовото ниво. Тъй като криптираните пакети изглеждат като обикновените IP пакети, те могат лесно да бъдат пренасочвани през IP мрежите точно както и обикновените пакети. Единствените устройства, които знаят за криптирането, са в крайните точки. Самият IPSec протокол използва различни технологии като DES криптиране и Цифрови сертификати. Технологията за криптиране гаранрира, че съобщенията не могат да бъдат подслушвани или четени от друг освен оторизирания получател. Съществуват няколко вида криптиращи алгоритми, но някои са по-сигурни от други. При повечето алгоритми оригиналните данни са криптирани чрез използването на определен ключ за криптиране, а получаващият компютър или потребител може да декриптира съобщението, като използва специфичен декриптиращ ключ. Алгоритмите за криптиране като DES, PGP и SSL определят как се създават и обменят тези ключове.

Заключение – IP версия 4 протоколът има редица недостатъци:

По-голямата част от адресите на мрежи от клас B са вече раздадени. Това означава, че в момента се раздават основно адреси на мрежи от клас C. При това големи организации, които притежават компютри, свързани с няколко мрежи, вместо да използват един мрежов адрес от клас B, получават няколко мрежови адреса от клас C.

Структурата на Интернет, в която се извършва маршрутизиране не е йерархична, и се изисква по един ред за всяка мрежа в маршрутните таблици. Тъй като броят на мрежите от клас C непрекъснато нараства, то нараства и размерът на маршрутните таблици. Възможно решение на този проблем се предлага от безкласовото маршрутизиране CIDR (Clessless Interdomian Routing). Също така е възможно обобщаването на IP

мрежа (supernetting) и записването на няколко мрежи от клас С само с един ред в маршрутните таблици.

В общия случай 32-битовите IP адреси не са достатъчни при днешните тенденции за разрастване на Интернет.

Някои групи адреси са заделени чрез RFC 1918 за частни IP адреси. Това означава, че тези адреси са достъпни за употреба от който и да е потребител, поради което едни и същи IP адреси от RFC 1918 могат да се използват в много мрежи. Обаче тези адреси не подлежат на маршрутизиране в Интернет. Те са широко използвани поради недостиг на регистрируеми (реални) IP адреси. Мрежи, които използват такива адреси, се нуждаят от превод на мрежовите адреси (NAT), за да се свържат с Интернет. Въпреки редицата мерки за икономисване на IPv4 адресите (като например, използване на NAT и частни IP адреси), броят на 32-битовите IP адреси е недостатъчен, за да поеме бъдещото разрастване на Интернет. Поради тази причина, вече има общо съгласие, че в близките 5 до 15 години трябва да се възприеме 128-битова схема за адресиране в Интернет. Какво е щяло да се разбира под IPv5, е съществувало само като експериментален на-IP поточен протокол в реално време, наречен ST2, описан в RFC 1819. Този протокол е изоставен в полза на RSVP.

Публикувани са редица спецификации (RFCI1752, RFCI1883 и др), които дефинират IP протокол от следващо поколение- IPng или IPv6. При IPv6, новият (но все още не широко използван) стандартен протокол за Интернет, адресите са 128-битови, което означава, че дори и при щедро даване на „нетблокове“, ще са достатъчни в обозримото бъдеще. Теоретично уникалните адреси са 18 445 618 199 572 250 625 (точно 264, или около 1,845*1019). Това огромно адресно пространство ще бъде рядко населено, което прави възможно отново да се кодира повече информация за маршрутизирането в самите адреси.

Адресът от версия 6 се записва с осем 4-цифрени (16-битови) шестнадесетични числа, разделени с двоеточия. Един низ от нули може да се прескочи, така че 1080::800:0:417A е същото, което и 1080:0:0:0:0:800:0:417A.

Глобалните уникални IPv6 адреси се състоят от две части: 64-битова маршрутизираща част, следвана от 64-битов идентификатор на хоста.

„Нетблоковете“ се характеризират като модерна алтернатива на IPv4 мрежов номер, следван от наклонена надясно черта и броят на съответните битове в мрежовия номер (с десетично число). Пример: 12AB::CD30:0:0:0:0/60 включва всички адреси, започващи с 12AB00000000CD3.

Освен по-голямото адресно пространство, IPv6 има много други подобрения спрямо IPv4, като автоматично преномериране и повишена сигурност чрез задължително използване на стандарта IPsec. Някои от основните предимства на IPv6 са следните:

Използва 128 битови адреси;

- Поддържа йерархия на адресите с цел да се осигури йерархично маршрутизиране и намаляване на размера на маршрутните таблици;

- Поддържа мобилни адреси и автоматична конфигурация на адресите;

- Поддържа шифриране и механизъм за идентификация на данните, групово разпространение на пакетите (multicast), възможност за задаване на тип на услугата (quality of service) и други.

По спецификацията и проверката на стандарта за протокола IPv6 работят голям брой организации на крайни потребители, както и фирми и групи за разработване на стандарти от много страни и международни организации.

Литература:

1. И. Цонев, С. Станев. Компютърни мрежи и комуникации, университетско издателство, „Епископ Константин Преславски”- Шумен, 2008;
2. http://phys.uni-sofia.bg/~burova/9.htm
3. http://delian.blogspot.com/2008/02/ipv4.html

ЕДИН СЪВРЕМЕНЕН ПОДХОД ЗА РЕАЛИЗАЦИЯ НА ИНТЕГРИРАНА СИСТЕМА ЗА НАБЛЮДЕНИЕ И УПРАВЛЕНИЕ НА КОРАБА

Йордан Сивков, Костадин Костадинов

A modern approach for integrated observation and control system of ship

Yordan A. Sivkov, Kostadin G. Kostadinov

Резюме: С развитието на съвременните микропроцесорни и компютърни технологии е възможно интегрирането на информацията получавана от всички сензори и създаването на единна система за управление на отделните системи на кораба. Използването на Ethernet мрежата за среда за предаването на данните и информацията от датчиците би спестило значителни финансови средства и физическо място.
Ключови думи: Ethernet мрежа, система за управление и наблюдение, микропроцесори, сензори.

Abstract: With progress of modern microprocessor and computer systems are possible to create integrated ship system from all sub-system. The using of Ethernet network for working area to transmit and receive data from sensors and modules are saved physical space and money.
Key words: Ethernet network, observation and control system, microprocessor, sensors.

Като всяка една сложна система корабът е съвкупност от голям брой съвместно или самостоятелно работещи системи, които в общия случай не използват съвместими интерфейси за обмяна на информация и възможност за предаване на състоянието си и получаване на сигнали за управление помежду си [1,3]. Вследствие на описаните по-горе особености има голяма

зависимост на надеждността и сигурността на кораба от квалификацията и висока концентрация на обслужващия го екипаж. Това от своя страна изисква постоянна преквалификация на членовете на екипажа поради липса на единни стандарти за изграждане и интегриране на новите системи. От друга страна всяка система поради липсата на или частичната интеграция използва свой набор от окабеляване, сензорна мрежа, индикаторни и управляващи устройства[1,3,4].

При проектирането и изграждането на всеки кораб, не е възможно технологично да се заложи целия потенциален ъпгрейд на системите за наблюдение и управление, както и евентуалните нови промишлени стандарти за обмен на данни. Това е невъзможно поради големия работен цикъл на кораба, обикновенно по-дълъг от „живота" на някой от стандартите.

От друга страна развитието на микроконтролерите дава възможност за реализиране на схеми с голяма изчислителна мощност, ниска консумация и малки размери. Това комбинирано с интегрираните възможности за комуникация /вградени серийни и паралелни интерфейси/, COTS /commercial off-the-shelf/ технологии и покриването на промишлените стандарти за работни параметри прави подходящи за прилагането им в корабни системи за наблюдение и управление[7].

Като обобщим можем да систематизираме следните посочени по-горе предимства на такава система:

- висока производителност – дават възможности за реализиране на системи работещи в реално време с масиви от данни от голям брой сензори или източници;
- ниска консумация – дава възможност за реализиране на системи с автономно захранване /батерии или акумулатори/;
- малки размери на реализираните апаратури;
- използване на микроконтролери реализирани по COTS /commercial off-the-shelf/ технологии;
- ниска себестойност сравнено с използваните към момента реализации;
- възможност за разширяване на системата от сензори и управляващи модули;

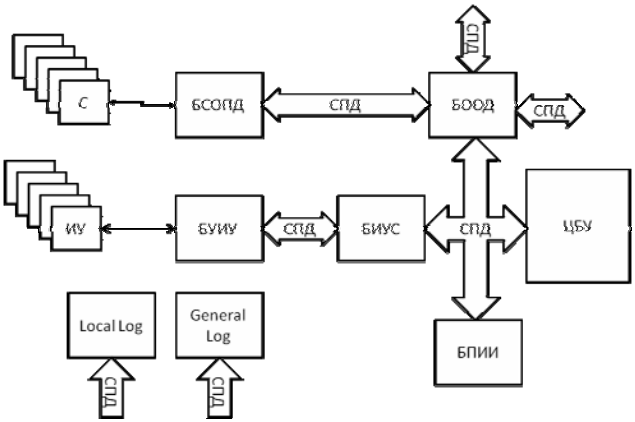
- използване на унифицирани стандарти за предаване на данни и възможност за замяна само на част от системата;

Базирайки се посочените по-горе предимства логично е да се достигне до напълно интегрирана система за наблюдение и контрол основаваща на микроконтролери и използващи подходящ протокол като средата за обмен на данни. В настоящия доклад се предлага едно решение, което има възможност да замести съществуващите системи или да се изгради напълно нова такава при конструирането на нови проекти кораби.

Като определящи изграждането на подобна система можем да определим следните условия:

- съвместимост със съществуващите стандарти за четене на данни от сензори;
- възможност за интегриране на различните подсистеми в единна система;
- удовлетворяване на изискванията за работните параметри на апаратурата в морски условия /температура, влажност и др./;
- възможност за обработване на данните от голям брой сензори и устройства и вземането на решение в „реално” време;
- висока надеждност на системата и възможност за документиране на събитията и данните;
- съвместимост с използваните изпълнителни устройства и възможност за тяхното управление;
- достатъчен капацитет на каналите за обмен на данни.

Удовлетворявайки поставените изисквания се изгражда системата посочена в фигура 1. Разглеждайки блоковете и посочените връзки се доближа значително до настоящите системи. Основната разлика е реализацията на блоковете и използваните канали за предаване на данни.

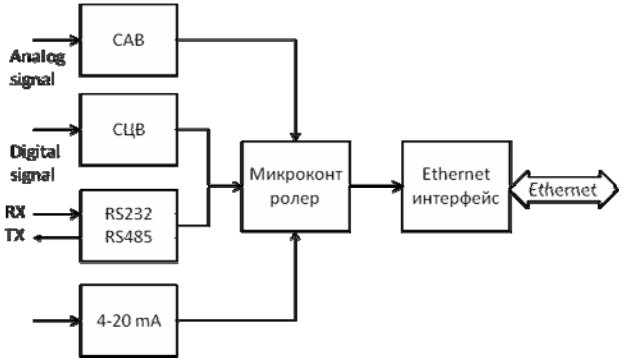


Фиг. 1 Интегрирана система за наблюдение и управление

БСОПД *- блок за събиране, обработка и предаване на данни*
БООД *– блок за обединяване и обработка на данни*
СПД *– среда за предаване на данни*
ЦБУ *– централна блок за управление*
БИУС *– блок за изработване на управляващите сигнали*
БУИУ *– блок за управление на изпълнителните устройства*
ИУ *– изпълнителни устройства*
С *– сензори*
БПИИ *– блок за предаване и извеждане на информация*

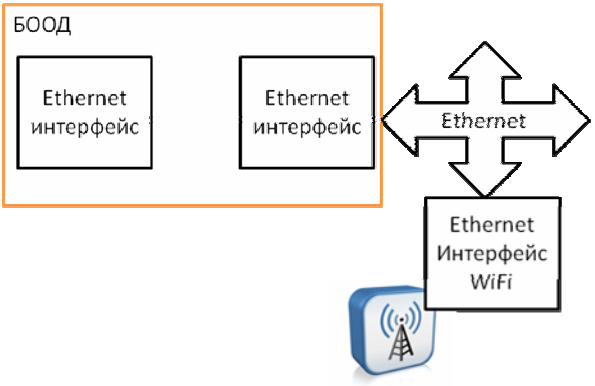
Системата залага на използването на съществуващите стандарти на сензорите и интегрирането на вече поставените и работещи датчици. Това се постига чрез БСОПД, който разполага с всички необходими входове, които са конфигурирани в зависимост от конкретния случай. Блока е изграден от съвременен RISC микроконтролер [7], като проведените опити се използват образци на фирмата Microchip, един от лидерите в тази област. На фиг. 2 е представена обобщена блокова схема на БСОПД. Схемата разполага с входове за аналогови, цифрови сигнали и данни предавани по използваните стандарти за сериен и паралелен обмен [2]. Информацията се преобразува от съответната буферна схема и се предава към микроконтролера, който от своя страна събира и обработва информацията и

формира съобщението предавано през Ethernet интерфейса в корабната мрежа [5,7].



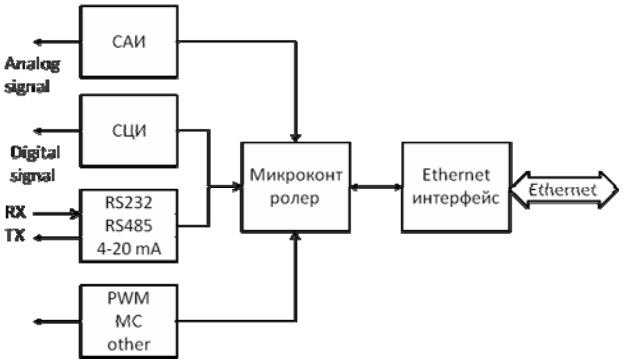
Фиг. 2 Обобщена блокова схема на БСОПД

Във всяка обособена част от корабното оборудване се предвижда интегрирането на описания модул, като от своя страна функционално и/или физически се обединява информацията от група апаратури или сложно изделие чрез БООД. Този блок изпълнява ролята на основен блок за управление за едно или група функционално обединени устройства. Пример за това е пожароизвестителната система [4] или основен дизел агрегат [1,3]. Ако използваме аналогията с компютърните мрежи, можем да определим БООД, като рутиращ и управляващ блок събиращ, обединяващ информацията от отделните БСОПД и я предава към ЦБУ, който от своя страна събира, обработва, записва и разпределя информацията между различните подсистеми управлявани от БООД. Така с използването на единна мрежа могат да се приема, предава и изобразява информация от всички подсистеми на кораба на един дисплей във всяка част от кораба /фигура 3/.



Фиг. 3 WiFi модул в системата

Допълнително Ethernet технологията позволява използването на широко използваните модули за безжична връзка, което от своя страна позволява използването на PDA устройства с подходящ софтуер даващи нужната информация на екипажа /обслужващия персонал/ във всяка една точка на кораба. Това изключително предимство може да предотврати аварии и намалява времето за обслужване и подобрява работата на екипажа. Тези мобилни индикатори в комбинация със стационарните такива, разположени в основните възли на кораба формират състава на блок БПИИ.



Фиг. 4 Обобщена блокова схема на БУИУ

Втората част на система е изградена от блокове осигуряващи изработването и изпълнението на управляващите сигнали. Тази подсистема е изградена от БИУС, БУИУ и ИУ. Характерното за микроконтролерите и хардуерните решения реализирани на тяхна основа е възможноста за интегриране на подсистемите за наблюдение и диагностика с подсистемите за контрол и управление е в един модул. За управление на мощни изпълнителни вериги и устройства се използва управление на междинни релета осигуряващи нужната мощност.

Последната част от системата са дневниците за събитията, които са дефинирани на две нива: локално /Local Log/ и общо /General Log/. Всяка една отделна подсистема и блок разполага със собствен дневник записващ информацията за неговото действие, която от своя страна се предава и към общия дневник за цялата система. Това гарантира двустепенно осигуряване на информацията, като дублира част от функциите на „черната кутия", което от своя страна дава допълнителна сигурност.

Тази система се базира на няколко основни принципа, които определят и нейните основни предимства:

- използва се среда за пренос на данни, стандартизирана в световен мащаб, лесна за употреба и осигуряваща необходимата пропускателна способност на канала. От друга страна позволява използването на WiFi модули без нуждата от допълнителни системи и преобразуватели. Друго предимство на използването на Ethernet интерфейс е премахването на големия брой кабелни трасета преминаващи през целия кораб и даващи предпоставка за възникване на аварии и обуславящи трудността на отстраняването им;
- блоковете и модулите на системата са изградени от микроконтролери позволяващи осигуряването на всички използвани стандарти за комуникация и сензори, висока производителност и възможност за обновяване на системата с и без промяна на хардуерното решение;
- интегриране на всички системи използвани в кораба;
- възможност за разделяне на системите в подсистемни клонове, които да интегрират определена дейност или група

дейности: като пример може да се даде пожароизвестителна и пожарогасителна системи, които могат да се отделят в отделен клон, като получават информация и от сензори от други клонове, например сензорите за температурата на дизел генератора и сензорите в кухнята.

Заключение

Посочената система е само принципен подход в изграждането на интегрирана система. Предлаганите стандарти за обмен на данни и елементи за изграждане на различните блокове значително надхвърлят изискванията към текущите нужди в кораба с това създават запас от производителност, който от своя страна гарантира развитие и надграждане на съществуваща система в зависимост от появилите се нужди. В реализирането се използва съвременна материална база с масово разпространение и производство, което от своя страна води до по-ниски цени.

Използването на подобна система не се ограничава само в корабостроенето а дава възможност за използване във всяка област изискваща интегриране на голям брой системи и тяхната работа съвместно. Например завод, цех или склад.

Литература:

1. Василев М., М. Колев, “Възможности на новите системи за управление, тип „Marex””, Морски научен форум том 1, 64-66, ВВМУ, Варна, 2006.
2. Драганов А.Г., “Аналого-цифров преобразувател на сигнали с PIC микроконтролер и записването им в персонален компютър”, Научна конференция, ВВМУ 2002, 103-108, Варна, 2002.
3. Манов М., ”Тенденции в развитието на интегрираните електроенергетични системи във военноморския флот”, Морски научен форум том 3, 127-135,ВВМУ, Варна, 2006.
4. Недев М., С. Янков, „Корабни пожароизвестителни и пожарогасителни уредби – състояние и перспективи”, Морски научен форум том 3, 142-150, ВВМУ, Варна, 2008.
5. Опенхайм А., “Приложение на цифровата обработка на сигналите”, ДИ ”Техника” София, 1982.
6. Lyons R.G., “Understanding digital signal processing”, Prentice Hall Ptr., 2001.
7. Microchip Data Sheets, www.microchip.com

ИЗСЛЕДВАНЕ СКОРОСТТА НА ИНТЕГРИРАНИ АЛГОРИТМИ НА ЛИНЕЙНИ ПРЕМЕСТВАЩИ РЕГИСТРИ С ОБРАТНИ ВРЪЗКИ В DSPIC

ст.ас. д-р Николай Николов

Използването на неразложими полиноми описващи алгоритмите на работа на линейни преместващи регистри като средство за криптиране е описано в редица публикации [2,3,4,5]. С тяхна помощ е възможно да се реализират генератори на псевдослучайни поредици с период 2^N-1. Описанието на различни криптиращи алгоритми е посочено в [2,3,4,5]. Алгоритмите за криптоанализ на такива псевдо-случайни поредици от тези полиноми са систематизирани и описани в [1,2,4].

Целта на изследването е да се оцени възможността за използване на широко разпространени сигнални процесори за генериране на гамиращи последователности и възможността за генериране на такива в „реално време" паралелно с генерирането на информационния поток.

Експериментите са проведени в следния ред:

- генериране са неразложими полиноми от 17 до 31 степен - като за всяка една от степените са генерирани полиноми с максимално и минимално възможното тегло;
- интегриране на получения полином в сигнален процесор;
- измерване на времето за генериране на бит и байт от алгоритъма;

Резултати

Получени са следните неразложими полиноми от 17 до 31 степен като за генерирането име бе използвана средата Matlab.

Полином от 17 степен с максимално тегло

$x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^9+x^8+x^7+x^6+x^5+x^3+x^2+x^1+x^0$;

Полином от 17 степен с минимално тегло

$x^{17}+x^{14}+x^0$;

Полином от 18 степен с максимално тегло

$x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^9+x^8+x^6+x^5+x^4+x^3+x^2+x^0$;

Полином от 18 степен с минимално тегло
$x^{18}+ x^{11} +x^{0}$;
Полином от 19 степен с максимално тегло
$x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{5}+x^{4}+x^{3}+x^{0}$;
Полином от 19 степен с минимално тегло
$x^{19}+x^{18}+x^{17}+x^{14}+x^{0}$;
Полином от 20 степен с максимално тегло
$x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{6}+x^{5}+x^{4}+x^{3}+x^{0}$;
Полином от 20 степен с минимално тегло
$x^{20}+x^{17}+ x^{0}$;
Полином от 21 степен с максимално тегло
$x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{6}+x^{5}+x^{4}+x^{2}+x^{0}$;
Полином от 21 степен с минимално тегло
$x^{21} +x^{19} +x^{0}$;
Полином от 22 степен с максимално тегло
$x^{22}+x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{6}+x^{5}+x^{3}+x^{2}+x^{0}$;
Полином от 22 степен с минимално тегло
$x^{22}+x^{21} +x^{0}$;
Полином от 23 степен с максимално тегло
$x^{23}+x^{22}+x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{6}+x^{5}+x^{4}+x^{3}+x^{2}+ x^{0}$;
Полином от 23 степен с минимално тегло
$x^{23} +x^{18}+x^{0}$;
Полином от 24 степен с максимално тегло
$x^{24}+x^{23}+x^{22}+x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{6}+x^{5}+x^{4}+x^{3}+x^{2}+x^{0}$;
Полином от 24 степен с минимално тегло
$x^{24}+x^{23} +x^{21}+x^{20} +x^{0}$;
Полином от 25 степен с максимално тегло
$x^{25}+x^{24}+x^{23}+x^{22}+x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{6}+x^{5}+x^{3}+x^{2}+x^{1}+x^{0}$;
Полином от 25 степен с минимално тегло
$x^{25}+x^{22}+x^{0}$;

Полином от 26 степен с максимално тегло
$x^{26}+x^{25}+x^{24}+x^{23}+x^{22}+x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{7}+x^{6}+x^{5}+x^{4}+x^{3}+x^{2}+x^{0}$;

Полином от 26 степен с минимално тегло
$x^{26}+x^{25}+x^{24}+x^{20}+x^{0}$;

Полином от 27 степен с максимално тегло
$x^{27}+x^{26}+x^{25}+x^{24}+x^{23}+x^{22}+x^{21}+x^{20}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{6}+x^{5}+x^{4}+x^{3}+x^{0}$;

Полином от 27 степен с минимално тегло
$x^{27}+x^{26}+x^{25}+x^{22}+x^{0}$;

Полином от 28 степен с максимално тегло
$x^{28}+x^{27}+x^{26}+x^{25}+x^{24}+x^{23}+x^{22}+x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{10}+x^{9}+x^{8}+x^{7}+x^{6}+x^{5}+x^{4}+x^{2}+x^{1}+x^{0}$;

Полином от 28 степен с минимално тегло
$x^{28}+x^{25}+x^{0}$;

Полином от 29 степен с максимално тегло
$x^{29}+x^{28}+x^{27}+x^{26}+x^{25}+x^{24}+x^{23}+x^{22}+x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{5}+x^{4}+x^{3}+x^{2}+x^{1}+x^{0}$;

Полином от 29 степен с минимално тегло
$x^{29}+x^{27}+x^{0}$;

Полином от 30 степен с максимално тегло
$x^{30}+x^{29}+x^{28}+x^{27}+x^{26}+x^{25}+x^{24}+x^{23}+x^{22}+x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+ x^{8}+x^{7}+x^{6}+x^{5}+x^{4}+x^{3}+x^{2}+x^{0}$;

Полином от 30 степен с минимално тегло
$x^{30}+x^{29}+x^{26}+x^{24}+x^{0}$;

Полином от 31 степен с максимално тегло
$x^{31}+x^{30}+x^{29}+x^{28}+x^{27}+x^{26}+x^{25}+x^{24}+x^{23}+x^{22}+x^{21}+x^{20}+x^{19}+x^{18}+x^{17}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{6}+x^{5}+x^{4}+x^{2}+x^{1}+x^{0}$;

Полином от 31 степен с минимално тегло
$x^{31}+x^{28}+x^{0}$;

Получените полиноми бяха интегрирани в dsPIC процесор 30F4011 работещ с 40 MHz тактова честота. Резултатите от използваните два критерия за оценка на интегрираните алгоритми са посочени в таблица 1

Таблица 1

Степен на използвания полином	Инстр. цикли за байт	Време за генериране на байт	Инстр. цикли за бит	Време за генериране на бит
17 степен с макс. тегло	626	62.6 μs	77	7.7 μs
17 степен с мин. тегло	402	402 μs	49	4.9 μs
18 степен с макс. тегло	642	64.2 μs	79	7.9 μs
18 степен с мин. тегло	418	41.8 μs	51	5.1 μs
19 степен с макс. тегло	658	65.8 μs	81	8.1 μs
19 степен с мин. тегло	466	46.6 μs	57	5.7 μs
20 степен с макс. тегло	706	70. 6 μs	87	8. 7 μs
20 степен с мин. тегло	450	45 μs	55	5.5 μs
21 степен с макс. тегло	722	72.2 μs	89	8.9 μs
21 степен с мин. тегло	466	46.6 μs	57	5.7 μs
22 степен с макс. тегло	770	77 μs	95	9.5 μs
22 степен с мин. тегло	482	48.2 μs	59	5.9 μs
23 степен с макс. тегло	834	83.4 μs	103	10.3 μs
23степен с мин. тегло	489	48.9 μs	63	6.3 μs
24 степен с макс. тегло	850	85 μs	105	10.5 μs
24 степен с мин. тегло	546	54.6 μs	67	6.7 μs
25 степен с макс. тегло	898	89.8 μs	111	11.1 μs
25 степен с мин. тегло	530	53 μs	65	6.5 μs

26 степен с макс. тегло	914	91.4 μs	113	11.3 μs
26 степен с мин. тегло	578	57.8 μs	71	7.1 μs
27 степен с макс. тегло	898	89.8 μs	106	10.6 μs
27 степен с мин. тегло	594	59.4 μs	73	7.3 μs
28 степен с макс. тегло	962	96.2 μs	119	11.9 μs
28 степен с мин. тегло	578	57.8 μs	71	7.1 μs
29 степен с макс. тегло	1010	101 μs	125	12.5 μs
29 степен с мин. тегло	594	59.4 μs	73	7.3 μs
30 степен с макс. тегло	1026	102.6 μs	127	12.7 μs
30 степен с мин. тегло	642	64.2 μs	79	7.9 μs
31 степен с макс. тегло	1074	107.4 μs	133	13.3 μs
31 степен с мин. тегло	626	62.6 μs	77	7.7 μs

Максималната честота за генериране на бит от последователността при полином от 31 степен е 1/(7.7 μs) 129kHz и 1/(13.3 μs) 75kHz. Последователността генерирана от полином от 31 степен е 2147483647, от което следва, че при 7.7 μs се повтаря след около 4h . При генериране на бит за 13.3 μs се поредицата би се повторила след 8h.

Изводи

1. Времето за генериране на последователността от всеки полином е правопропорционална на степента на полинома и неговото тегло.

2. Времето за генериране на бит от последователността позволява dsPIC да се използва „в реално време” за генериране на гамираща последователност при криптиране на речева информация.

Литература

1. Stoyanov, B., Recent Attacks Against Summation, Shrinking and Self-Shrinking Stream Ciphers – Short Survey, Fourth Scientific Conference with International Participation “Space, Ecology, Nanotechnology, Safety”- SENS 2008, Varna, Bulgaria, 4-7 June, 2008, Proceedings of SENS 2008, pp. 197-200,

2. Alexander Maximov Some Words on Cryptanalysis of Stream Ciphers - Ph.D. Thesis, June 16, 2006 Department of Information Technology Lund
University

3. Menezes, P. Van Oorschot, and S. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996.

4. Risto Hakala Linear Cryptanalysis of Two Stream Ciphers Master’s thesis degree of Master of Science in Technology Espoo, December 5, 2007.

5. Целков В., Здравков З., Стоянов Н., Стратегия за информационна сигурност в C4I системите на Българската армия -. Международна научна конференция “Хемус 2000”, Пловдив 2000 г.

АНАЛИЗ НА СЪВРЕМЕННОТО СЪСТОЯНИЕ НА МЕТОДИТЕ ЗА ПОТОЧНО ШИФРИРАНЕ

Боряна Узунова
Валери Тонев
Борислав Беджев

ANALYSIS OF THE PRESENT TRENDS IN THE METHODS FOR STREAM ENCODING

Boryana Uzunova
Valery Tonev
Borislav Bedzhev

Annotation: Stream ciphers are very important for the present communication systems, because they provide the protection of the transferred data and the resources of the systems in real time. With regard to this fact in the paper the present trends in the methods for stream encoding are analyzed. Paper is organized as follows. First, the basics of the stream ciphers are recalled. Second, on the base of the information from open literature and Internet sources the most common schemes of stream ciphers are explored. At the end some important conclusions concerning the present approaches to the development of stream ciphers are given.

1. Въведение

В условията на пазарна икономика и при остра конкурентна борба между различни организации възникват много проблеми със съхраняването и защитата на информацията във фирмените комуникационно-информационни системи (КИС). Това е така, защото бизнесът неразривно е свързан с получаването, натрупването обработката и използването на разнообразни масиви информация. Ето защо в бизнеса злонамерените лица преследват три основни цели [11]:

- тайно получаване на информация, която може да повлияе върху стратегиите и тактиките на конкурентната борба;

- въвеждане на изменения в информационните потоци на конкурентите, които да доведат до тяхната дезинформация;

- унищожаване на ценни за конкурентите сведения, чертежи, масиви от данни.

От системна гледна точка събирането на сведения за фирмите и частните лица става чрез различни методи, които могат да бъдат обединени в три основни групи:

- на базата на източници на информация със свободен достъп;

- чрез използване на субектите, които са носители на конфиденциална информация;

- чрез използване на така наречените технически канали за изтичане на информация.

Под защита на информацията се разбира съвкупността от мерки и действия, насочени към осигуряването на конфиденциалността и целостта на информацията в процеса на нейното събиране, съхранение, обработка и предаване [6], [7], [8].

Един от най-важните компоненти на всяка система за информационно-компютърна защита е подсистемата за криптографска защита. Шифрираща система или криптосистема е технология, използвана за скриване на съобщенията от нежелани получатели. Криптографията е наука, която създава такива технологии, докато криптоанализът е изкуството за разкриване или разбиване на криптосистеми с цел прочитане на дадено съобщение от някой, който не е оторизираният получател. Криптографията и криптоанализът са основните дялове на съвремнната криптология [6], [7], [8], [11].

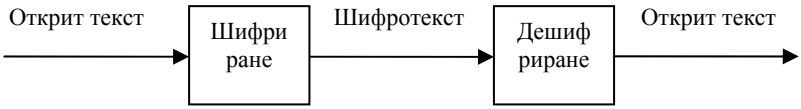
Оригиналното съобщение, което трябва да се изпрати се нарича ***открит*** (***явен***) ***текст*** (***plain text***). Шифрираното съобщение се нарича ***шифротекст*** (***cipher text***). ***Шифрирането*** е процес на преобразуването на простия текст в шифротекст, обикновено използвайки някакъв алгоритъм и ***ключ***. ***Ключът*** е този компонент, който се предава тайно между изпращащия и получаващия съобщението и може да бъде различен във всяко

едно съобщение. Ключът обикновено се нарича и ***криптопроменлива***. ***Дешифрирането*** е процесът на преобразуване на шифротекста в явен текст. Този процес обикновено произхожда от знанието на използвания шифриращ алгоритъм и ключа.

В последните двадесетина години криптологията се оформи като самостоятелна научна дисциплина, обслужваща много сфери на обществения живот: банково дело, търговия, комуникации и пр. Това е обусловено от появата на такива фундаментални идеи като асиметричната криптография (с открит ключ), доказано устойчивите протоколи, чиято надеждност е основана на гарантираната сложност на решенията на математическите задачи. Използването на тези методи за защита на информацията, гарантиращи надеждност при предаване на съобщенията, които са станали основа на протоколи и системи за установяване на автентичността (истинността) на съобщенията в общия случай, а в частност при определяне принадлежността на обектите на принципа "***свой – чужд***" ("***friend – foe***") [5], [6], [7], [8], [11].

Основните цели на криптографията са: о*сигуряване на конфиденциалност,* о*сигуряване на целостта на данните,* а*втентификация* и н*евъзможност за отричане на съобщението*.

Процесът на криптографското преобразуване е показан на фиг. 1.



Фиг. 1. *Процес на криптографско преобразуване*

Устойчивостта на криптоалгоритъмът (криптоустойчивост) *се заключава в способността на шифъра да противостои на атаките за неговото разшифроване.* Криптоустойчивостта зависи от сложността на алгоритъма за преобразуване, дължината на ключа и обема на ключовото множество [5], [6], [7], [8], [11].

Криптографските алгоритми могат да се класифицират по степента на доказване на нивото на устойчивост: ***безусловно устойчиви*** и ***доказано устойчиви***.

Безусловно устойчивите криптоалгоритми гарантирано не позволяват разкриването на ключа. Безопасността на ***доказано устойчивите криптоалгоритми*** се определя от сложността на решаването на добре изследвани задачи. Отличителна особеност на тези криптоалгоритми е тяхната „твърдост", т.е. невъзможността да бъдат модифицирани (усложнени) по пътя на незначителни промени.

Според броя на символите от явния текст, които се шифрират за една стъпка на шифриращия алгоритъм, шифрите се делят на : *поточни* и *блокови*.

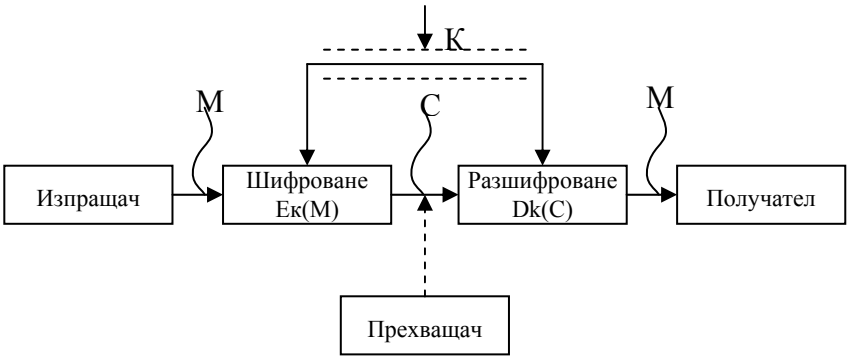
Блоковите шифри обработват големи блокове от открит текст едновременно (например по 64 и повече бита). Същата функция се използва и за дешифриране на съобщението. Следователно блоковите шифри са без памет.

Поточните шифри използват просто шифриращо преобразуване в зависимост от ключовия поток, който се използва. Ключовият поток може да бъде генериран по случаен закон или чрез алгоритъм, който да генерира ключове от първоначален малък ключ (наречен ядро), или от ядро и от предходни символи на шифъра. Този алгоритъм се нарича кодов генератор. Поточните шифри могат да се разглеждат като блокови като дължината на блока е само 1 бит (символ).

Шифриращата функция може да се променя, докато се обработва откритият текст. Ето затова се казва, че поточните шифри имат памет. Разликата между поточните и блоковите шифри не е точно дефинирана, защото при блоковите шифри, ако се добави малко памет като например в режим ***CBC*** (***Cipher Block Chaining***), се получава поточен шифър, обработващ големи блокове.

От гледна точка на операцията шифриране съществуват два вида алгоритми: ***симетрични*** и ***асиметрични*** [5], [6], [7], [8], [11].

Обобщената схема на криптографска система използваща симетричен алгоритъм за шифриране е следната (фиг. 2):



Фиг. 2. *Обща структурна схема на симетрична криптосистема*

При симетричните криптосистеми се използват еднакви секретни ключове в блока за шифриране и в блока за дешифриране. Техните основни предимства са:

- голяма скорост на шифриране и дешифриране както при хардуерна, така и при софтуерна реализация;
- лесни за реализация;

Същевременно за симетричните криптосистеми са характерни следните недостатъци:

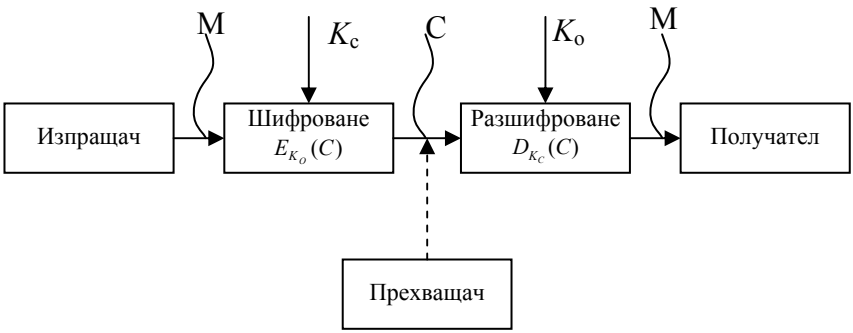
- секретният ключ е само един и всяка от страните участващи в процеса може да го компрометира случайно или целенасочено.
- при комуникация по двойки в мрежа от ***n*** участника са необходими ***n(n-1)/2*** ключа.
- изискват често смяна на ключа (при всяка сесия), което поражда проблем с разпространението на ключовете;

- не са удобни за използване в механизми за електронен подпис, защото изискват големи ключове за проверяващата трансформация и въвличане на трета доверена страна.

Когато процедурата шифриране е асиметрична по отношение на дешифрирането, тогава криптосистемите се наричат ***криптосистеми с открит ключ (двуключови)*** *или* ***асиметрични***.

Обобщената схема на асиметрична криптосистема с два ключа $\boldsymbol{K_c}$ и $\boldsymbol{K_o}$ е показана на фиг.3.

В тази криптосистема един от ключовете е *открит* (*явен*), а другият – *секретен*. Секретният ключ трябва да се предава на изпращача и получателя по защитен канал. При асиметричните системи по незащитен канал се предава откритият ключ, а секретният се съхранява в мястото на неговата генерация.



Предимствата на асиметричните криптосистеми са:

- двойката ключове (частен и публичен) могат да бъдат използвани многократно и през дълъг период от време (дори няколко години);

- в контраст на симетричния случай, при мрежа от много участници са необходими толкова двойки ключове, колкото са участниците;

- позволяват създаване на ефективни схеми за електронен подпис с проверяваща функция изискваща ключ с доста по-малки размери от съответните, построени със симетрични алгоритми.

Основните недостатъци на асиметричните криптосистеми са:

- относително по-бавни в сравнение със симетричните.

- при използване на криптиране ключът им е много по-дълъг от този на симетричните.

- дължината на добавения електронен подпис е относително голяма.

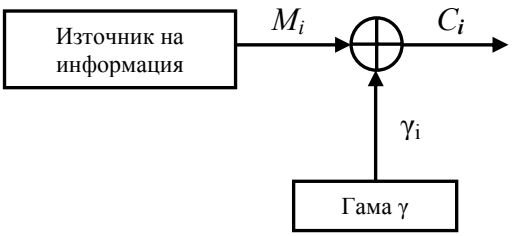
Поточните шифри се отличават с висока скорост на шифрирането, съизмерима със скоростта на постъпване на входните данни. Така се постига шифриране в реално време, независимо от обема и разрядите на потока шифрирани данни. В резултат на това поточните шифри намират много широко приложение в съвременните високоскоростни комуникационно – информационни системи [5], [6], [7], [8], [9], [10], [11]. По тази причина те са обект на по – задълбочено разглеждане по – нататък.

2. Анализ на съвременното състояние на методите за поточно шифриране

Поточните шифри използват просто шифриращо преобразуване в зависимост от ключовия поток, който се използва. Ключовият поток може да бъде генериран по случаен закон или чрез алгоритъм, който да генерира ключове от първоначален малък ключ (наречен ***ядро***), или от ядро и от предходни символи на шифъра [5], [6], [7], [8], [11]. Този алгоритъм се нарича ***кодов генератор***.

Прието е процесът на налагане по определен закон на гамата на шифъра върху открития текст да се нарича ***гамиране*** *(фиг.4). Под* ***гама на шифъра*** *се разбира*

псевдослучайна последователност, избирана по зададен алгоритъм и предназначена както за шифриране на открития текст, така и за дешифриране.



Фиг.4. *Структурна схема на поточен шифър*

Поточните шифри се класифицират както следва [5], [6], [7], [8], [11]:

1) Шифри с еднократно използване на ключа (one-time pad): Пример за такива шифри е шифърът на Вернам. Тук за всяко съобщение е необходимо да се използва нов ключ, който да е с дължина равна или по-голяма от дължината на предавания открит текст.

2) Синхронни шифри: В ***синхронните поточни шифри*** гамата се формира независимо от входната последователност, за всеки елемент (бит, символ, байт и т.н.), което е независимо от другите елементи. В синхронните поточни шифри отсъства ефектът на размножаване на грешката, т.е. броят на изкривените елементи в дешифрираното съобщение е равен на броя на изкривените елементи в шифрограмата, получена от канала за свръзка. При тези шифри е задължително да има синхронизация. Изпускането на един символ води до нарушаване на синхронизацията и до неправилното дешифриране на информацията.

3) Асинхронни (самосинхронизиращи се) шифри:

При ***самосинхронизиращите се*** поточни шифри елементите на входната последователност се шифрират с отчитането на N на брой предходни елемента, които вземат

участие при формирането на ключовата последователност. При тези шифри има разпространение на грешката, като в същото време за разлика от синхронните се възстановява синхронизацията автоматично след *N* елемента на шифрограмата.

Методът на гамирането позволява обмен на информацията в реален мащаб на времето, защото се реализира много просто.

Линейните рекурентни последователности са средство за математическо описание на един специфичен вид последователности за формиране гама на шифъра при метода на гамирането и при поточните шифри. Този метод се отличава с простота на практическата реализация и висока сложност на създадената гама. По тази причина, исторически той първи е бил приложен масово за защита на информацията в ***комуникационно–информационните системи*** (КИС) (в средата на $50^{-те}$ години на 20 век).

Общият вид на КИС, в която информацията се шифрира по метода на гамирането като за генератор на гамата се използва линейна рекурентна последователност, е показан на фиг. 5 [5], [6], [7], [8], [9], [10], [11].

Фиг.5: *Структура на КИС, в която информацията се шифрира с помощта на ЛРП*

Както се вижда от фиг. 5, всеки бит на открития текст се сумира с един бит от гамата (поредно число от линейната рекурентна последователност), при което се получава съответния бит от шифрограмата. Приложението на линейни рекурентни

последователности в КИС при метода на гамирането произтича от следните положителни свойства на линейните рекурентни последователности:

- Генераторите на гамите много лесно се реализират с дискретни цифрови елементи, което осигурява на устройствата надеждност, бързодействие и ниска цена.

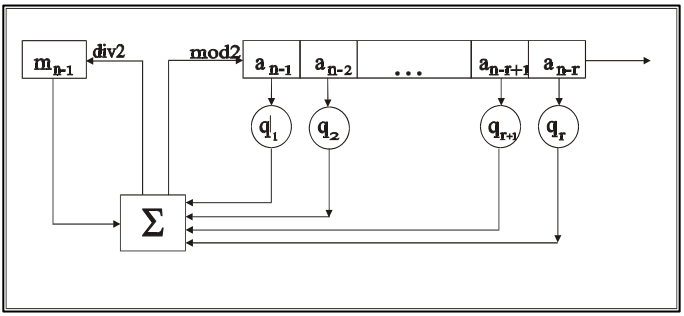
- Лесно е създаването на еднотипни генератори, тъй като от фиг. 5 се вижда, че всеки потребител на криптографски защитената КИС трябва да притежава такъв генератор.

- Формираната гама $\gamma_1, \gamma_2, \gamma_3, \ldots$ е много подобна на една напълно хаотична редица от битове, поради което е устойчива на криптоатаки.

Целта в криптографските приложения е периодът на гамата да бъде максимално голям, за да се постигне максимална криптоустойчивост на системата. Големият период дава възможност да се избегне така наречената ***атака с известен открит ключ***.

В началото на 60-те години на 20-ти век американските теоретици Берликемп и Меси разработват алгоритъм, чрез който криптоаналитиците въз основа на успешно прехванати ***2n*** бита от гамата могат да възстановят линейното рекурентно уравнение, формиращо ЛРП на фиг. 5. Тук ***n*** е броят на елементите на гамата, участващи в линейното рекурентно уравнение.

С цел осуетяване на атаката на Берлекемп – Меси в началото на 90-те години на 20-ти век американските криптографи-математици ***Е. Клапер*** и ***М. Горецки*** доразвиха идеята на ЛРП като разработиха нова архитектура за генерация на гами на основата на преместващи регистри, които те наричат ***преместващи регистри с обратна връзка с пренос*** (***ПРОВП***). Това са преместващи регистри, които имат неголямо количество памет. Клетките на регистъра се запълват с 0 и 1, а паметта съдържа неотрицателно цяло число (фиг. 6) [1], [2], [3], [5], [6].



Фиг. 6: *Преместващ регистър с обратна връзка с пренос*

За изследване на въпросите, свързани с теорията, реализацията и криптоанализа на преместващи регистри от този тип се използва математическия апарат на т.н. ***p-адични*** и ***N-адични числа***, като тук *p* е произволно просто, а *N* е съставно число [1], [2], [3], [5], [6].

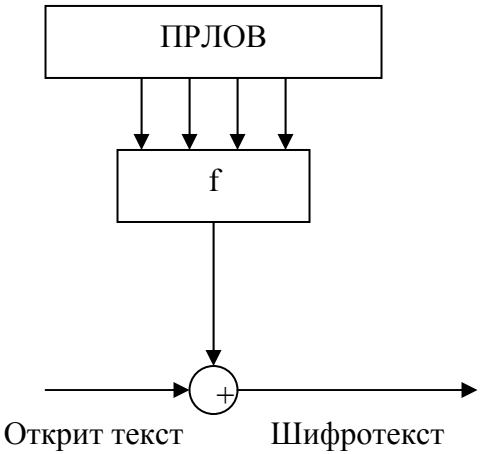
През 2004 г. Клапер и Горески показаха, че атаката на Берлекемп – Меси може да бъде модифицирана, така че криптоаналитиците на базата на прехванати ***2(n+m)*** бита от гамата да могат да определят структурата на ПРОВП и неговото начално състояние, което е равнозначно на разбиване на поточния шифър.

Предвид на това на съвременния етап в развитието на поточните шифри се използват два метода за повишаване на тяхната криптоустойчивост. Тяхната обща идея е да се използват ЛРП и/или ПРОВП като градивни елементи в по – сложни конструкции, така че да се гарантира високо ниво на защита на информацията на базата на положителните страни на ЛРП и ПРОВ - високо бързодействие и ниска цена.

Първият метод се заключва в подходящо комбиниране на изходите на няколко паралелно работещи ЛРП и/или ПРОВП. В литературата подобни генератори на гамиращи последователности са наричани ***комбиниращи*** [1], [2], [3], [4], [5], [6].

Другата алтернатива е гамата да се формира чрез нелинейна функция от състоянието на един единствен регистър. Такива регистри е прието да се наричат ***филтър-генератори*** или ***филтриращи генератори***.

На фиг. 7 и 8 са представени структурите на посочените два типа генератори.

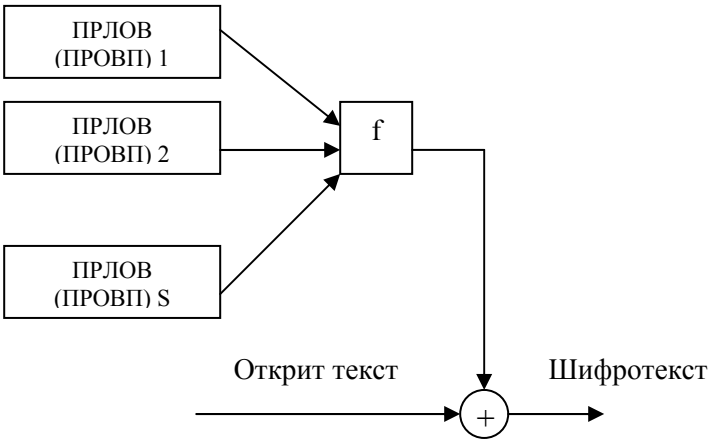


Фиг. 7: *Филтър - генератор на псевдо–случайни последователности (гами)*

Анализът на данните, посочени в откритата литература, показва, че на съвременния етап въпреки изключително интензивните научни изследвания няма разработени общи математически методи за синтезиране на нелинейната функция ***f*** (фиг. 7), гарантиращи висока криптоустойчивост на филтър – генераторите на гами. Основната част от разработките в разглежданото направление на развитие на поточните шифри са концентрирани върху построяването на ***последователности с пълна дължина***, наричани още ***последователности на де Брейн***. Идеята на тези разработки е гамите (последователностите), представляващи ЛРП, да се допълнят по подходящ начин със сегмент от ***n*** последователни нули, защото

той отсъства при класическите ЛРП (тук ***n*** е броят на елементите на гамата, участващи в линейното рекурентно уравнение) [1], [2], [3], [4], [5], [6], [8], [11].

Както се вижда от фиг. 8, в най-общ вид идеята на метода на комбиниращия генератор е да се формира криптографски силна последователност чрез обработка (комбиниране) на изходите на няколко ***преместващи регистри с линейна обратна връзка*** (ПРЛОВ), реализиращи ЛРП.



Фиг. 8: *Комбиниращ генератор на псевдо–случайни последователности (гами)*

Схемата от фиг. 8 от 1984г. насам е обект на интензивни изследвания поради следните причини:

Първо, общата схема на филтър – генераторите (фиг. 7) всъщност е частен случай на общата схема на комбиниращите генератори (фиг. 8) при S = 1.

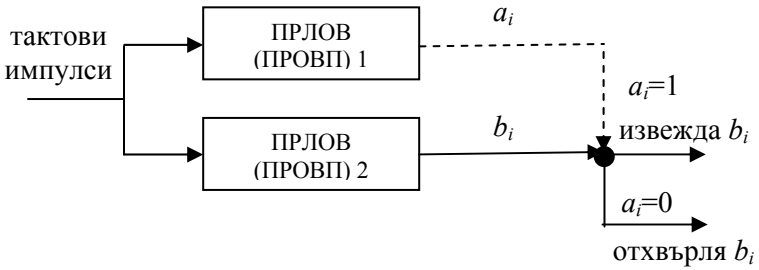
Второ, независимо, че градивните елементи са ПРЛОВ и/или ПРОВП, резултантната гама при редица конкретни конструкции е практически неотличима от напълно случайна последователност. Така например ***Рюпел*** е доказал, че ако *R*-те задействани в комбиниращия генератор ПРЛОВ формират

"парциални" гами, чиито периоди n_i, $i = 1,2,...,R$ са различни взаимно прости числа, тогава "резултантната" гама ще има максимална линейна сложност $\Lambda(Z)$.Тук под $\Lambda(Z)$ се разбира дължината на ПРЛОВ, който би могъл да бъде построен в резултат на криптоатаката на Берлекемп-Меси.

Предвид на изложените причини, основното внимание в изложението по – нататък ще бъде фокусирано върху основните схеми на реализация на комбиниращи генератори.

Най – просто идеята на комбиниращия генератор се реализира като функция ***f*** от фиг. 8 се избира да бъде сумиране по модул 2, при което схемата се нарича ***сумиращ комбиниращ генератор***.

Друга схема на построение на комбиниращите генератори, намираща много голямо приложение на съвременния етап от развитието на поточните шифри, е така наречения ***свиващ генератор*** (фиг. 9) [1], [2], [3], [4], [5], [6], [8], [11].



Фиг. 9: *Структурна схема на свиващ генератор*

Свиващият генератор се състои от два ПРЛОВ или ПРОПВ, като единият регистър управлява работата на втория регистър, както е показано на фиг. 9. Схемата работи по следния алгоритъм:

- ако в такт i на изхода на ПРЛОВ 1 има 1 (т.е. a_i=1) на изхода на схемата се получава стойността b_i от изхода на ПРЛОВ 2;

- ако в такт *i* на изхода на ПРЛОВ 1 има 0 (т.е. a_i=0) схемата не произвежда изходна стойност.

Двата регистъра се управляват синхронно от тактов генератор.

Пример: Нека ПРЛОВ 1 се състои от 3 тригера като обратните връзки се описват с многочлена x^3+x+1, а ПРЛОВ 2 е регистър с 5 тригера като полиномът, описващ обратните връзки, е x^5+x^3+1. Нека първоначалните състояния на регистрите са съответно [1 0 0] и [0 0 1 0 1]. При тези условия ПРЛОВ 1 произвежда периодично последователността 0; 0; 1; 1; 1; 0; 1, а ПРЛОВ 2 формира периодично последователността 1; 0; 1; 0; 0; 0; 0; 1; 0; 0; 1; 0; 1; 1; 0; 0; 1; 1; 1; 1; 1; 0; 0; 0; 1; 1; 0; 1; 1; 1; 0. "Резултантната" гама, формирана от свиващия генератор ще бъде 1; 0; 0; 0; 0; 1; 0; 1; 1; 1; 1; 1; 0; 1; 1; 1; 0; ...

Предимствата на свиващите генератори са: голям период на повторение, висока линейна сложност и добри статистически свойства.

Ако ПРЛОВ 1 и ПРЛОВ 2 имат обратни връзки, описвани с различни примитивни полиноми от степени L_1 и L_2 съответно, тогава свиващият генератор формира гама с максимален период на повторение $(2^{L_1}-1).(2^{L_2}-1)$, като тук $2^{L_1}-1$ и $2^{L_2}-1$ са съответно периодите на повторение на ПРЛОВ 1 и ПРЛОВ 2. Линейната сложност на такъв генератор е от порядъка на $L_1 + L_2$.

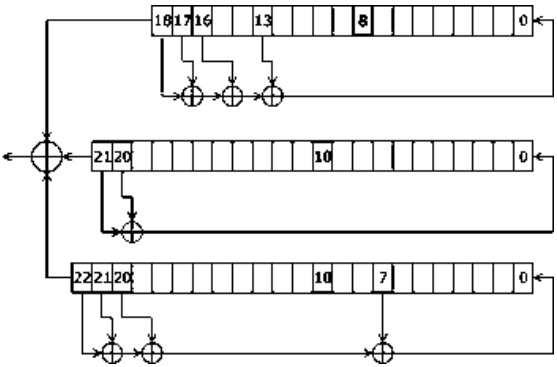
Разновидност на свиващия генератор е ***самосвиващият се генератор***. В тази схема вместо два ПРЛОВ има само един ПРЛОВ, който се "самоуправлява" по следния начин. Ако на изхода на ПРЛОВ в такт *i* има 1 изходната стойност на ПРЛОВ от такт *i+1* става поредния бит на гамата. Ако на изхода на ПРЛОВ в такт *i* има 0, то изходна стойност на ПРЛОВ от такт *i+1* се отхвърля.

Както се вижда, самосвиващите се генератори се реализират с двойно по-малко памет от свиващия генератор, но и скоростта им на работа е двойно по-малка.

Анализът на данните, изнесени в откритата литература, показва, че сумиращите и свиващите комбиниращи генератори се използват изключително широко за шифриране на информацията в реално време в съвременните

КИС. Този извод се подкрепя от следните факти [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11].

Алгоритъм *A5* е поточен шифър, използван за шифриране в GSM (*Group Special Mobile*) комуникациите в канала между мобилния апарат и базовата станция. Информацията в другия канал не се шифрира. Състои се от три ПРЛОВ с дължини 19, 22 и 23 (фиг. 10). Изходната последователност е *XOR* на трите регистъра. ***A5*** използва различни тактови честоти. Всеки регистър се тактува в зависимост от неговия среден бит, извършващ логическата операция *XOR* с обратните функции на средните битове на трите регистъра.



LFSR номер	Дължина битове	Полином на обратната връзка	Тактов бит	Изходни битове
1	19	$x^{19}+x^5+x^2+x+1$	8	13, 16, 17, 18
2	22	$x^{22}+x+1$	10	20, 21
3	23	$x^{23}+x^{15}+x^2+x+1$	10	7, 20, 21, 22

Фиг. 10: *Алгоритъм **A5***

A5 е доста ефективен, защото преминава всички статистически тестове. Единствената негова слабост е, че регистрите му са къси, за да направят пълното претърсване

невъзможно. Ако се използват по-дълги регистри с по-добри полиноми на обратните връзки ще се получи и по-сигурна схема.

Алгоритъмът Hughes XPD/KPD е създаден от ***Hughes Aircraft Corp***, която го монтира в радиостанциите от тактическо ниво на сухопътните войски на чуждите армии, купуващи комуникационно оборудване от САЩ. Създаден е през 1986г. и е наречен ***XPD*** (от ***Exportable Protection Device***). По-късно е преименуван на ***KPD*** (***Kinetic Protection Device***) и е декласифициран. Използва 61-битов ПРЛОВ. Има 210 примитивни полинома на обратната връзка, които са одобрени от ***National Security Adency*** (NSA), която е водещата държавна организация на САЩ в областта на сигурността на информацията. Ключът избира един от тези полиноми (те се съхраняват в ROM паметта), както и първоначалното зареждане на ПРЛОВ. Освен това има осем различни нелинейни филтъра, всеки от които има шест изхода, формиращи 1 бит от изходната му последователност. Изходните битове от осемте филтъра се комбинират, за да се получи 1 байт, който се използва за шифриране и дешифриране на открития текст.

Алгоритъмът Nanoteq е разработка на южноафриканска компания. Алгоритъмът използва 127-битов ПРЛОВ с фиксиран полином на обратната връзка, ключът на схемата е първоначалното зареждане на регистъра с обратна връзка. $127^{-те}$ бита на регистъра са редуцирани до един бит на шифропотока използвайки 25 примитивни клетки.

Алгоритъмът Rambutan се продава само като хардуерен модул и е одобрен за защита на информацията до ниво ***класифицирано***. Алгоритъмът е секретен и чипът не се продава в търговската мрежа.

Rambutan има 112-битов ключ (плюс битове за четност) и може да функционира в три режима. Предполага, че е поточен шифър с ПРЛОВ. Има пет преместващи регистъра, всеки от които е дълъг около 80 бита. Полиномите на обратните връзки са с около 10 члена всеки. Всеки регистър осигурява четири входа на голяма и сложна нелинейна функция, която формира един бит.

Алгоритъмът RC4 е най-широко разпространеният софтуерен поточен шифър, използван в много протоколи като например ***Secure Sockets Layer*** (***SSL***) (за защита на Интернет

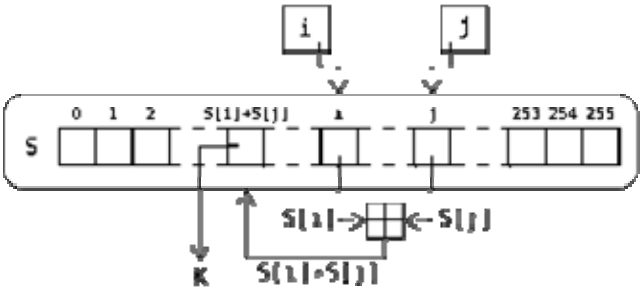
трафика) и ***WEP*** (за защита на безжичните мрежи). Този алгоритъм впечатлява с простотата си, но за сметка на това се оказва доста прост спрямо условията, които криптографите са поставили и някои приложения на този алгоритъм водят до доста несигурни криптосистеми (включително *WEP*). Не се препоръчва за употреба в новите системи, но въпреки това някои системи, базирани на ***RC4***, са достатъчно сигурни от практическа гледна точка.

Алгоритъмът представлява поточен шифър с променлива дължина на ключа. Разработен е през 1987г. от ***Рон Ривест*** за компанията ***RSA Data Security, Inc***.

RC4 има *S*-блок (блок за замяна) с размер 8x8: S_0, S_1,..., S_{255}. Този блок представлява пермутация на числата от 0 до 255. Пермутацията е функция на ключа с променлива дължина. Също така има два брояча *i* и *j*, които са установени в нула.

Генерирането на ключов поток се състои от две части:

- пермутация на всички 256 байта (обозначени като блок *S*);
- изменят се два 8-битови показателя (обозначени като *i* и *j*).



Фиг. 11: *Структура на блока за замяна при* ***RC4***

Пермутацията на блока *S* се извършва с ключ с променлива дължина, обикновено между 40 до 256 бита, използвайки ***key-scheduling algorithm*** (***KSA***). След като се изпълни тази операция, потока битове се генерира с използването на псевдо–случаен алгоритъм за генерация (*pseudo-random generation algorithm* - ***PRGA***).

Шифрирането при ***RC4*** става бързо – около 10 пъти по-бързо, отколкото при алгоритъма ***Data Encription Standard*** (DES), които е основен блкоков шифър, използван от американските и канадските частни фирми.

Алгоритъмът SEAL (*Software-Optimized Encryption Algorithm*) е много бърз поточен шифър, оптимизан за машини, работещи с 32-битови думи. **Една от характерните особености на този шифър е това, че той не е традиционен поточен шифър, а представлява сам по себе си семейство псевдослучайни функции**. Също така облекчава проблема със синхронизацията, свойствен на поточните шифри.

Основните идеи на ***SEAL*** са:

- използването на голяма, тайна, ключова таблица на *S*-блока (*T*);
- алтернативни аритметични операции, които не са комутативни (сумиране и *XOR*);
- използването на вътрешни състояния на шифъра, които не разкриват открития текст;
- променлива стъпкова функция, зависеща от номера на стъпката и променлива функция на итерацията, зависеща от номера на итерацията;
- използване на известни и отработени алгоритми за запълване на таблиците.

Алгоритъмът ***SEAL*** изисква изпълнението на 5 елементарни машинни операции при шифрирането на всеки байт информация. С това той се превръща в един от най-бързите програмно реализирани алгоритми за шифриране.

Алгоритъмът WAKE (*Word Auto Key Encryption)* е поточен шифър, създаден от ***Давид Уилър*** (***David Wheeler***) през 1993г. Шифърът работи в режим на обратна връзка по шифър, генерирайки ключов поток от предишните шифротекстови блокове. Алгоритъмът използва *S*-блок с 256 32-битови думи. Този шифър е бърз, но уязвим от атаки по избран открит текст и от атаки по избран шифротекст.

Обобщена характеристика на използваните на съвременния етап поточни шифри е дадена в следващата Таблица 1. Таблицата съдържа данни за годината на създаване на съответния поточен шифър, дължината на неговия ключ, както и бързодействието му. Криптоустойчивостта на посочените в таблицата поточни шифри се оценява на базата на известните до момента криптоатаки срещу тях.

Таблица 1: *Основни характеристики на използваните на съвременния етап поточни шифри*

Поточен шифър	Дата на създаване	Скорост (цикли / байт)	(битове)			Атака	
			Ефективна ключова дължина	Вектор за инициализация	Вътрешно състояние	Най-известна	Изчислителна сложност
A5/1	1989	за гласови данни (W_{phone})	54	114	64	Active KPA OR KPA Time-Memory Tradeoff	~2 sec. или $2^{39.91}$ операции
A5/2	1989	за гласови данни (W_{phone})	54	114	64?	Active	4.6 milliseconds
FISH	1993	много бърз (W_{soft})	голяма	***?***	***?***	Known-plaintext attack	2^{11} операции
Grain	преди 2004	бърз	80	64	160	Key-Derivation	2^{43} операции
ISAAC	1996	2.375 ($W_{64\text{-}bit}$) - 4.6875 ($W_{32\text{-}bit}$)	8-8288 обикновено 40-256	N/A	8288	(2006) First-round Weak-Internal-State-Derivation	4.67×10^{1240} (2001)
MUGI	1998-2002	***?***	128	128	1216	N/A (2002)	~2^{82} операции
PANAMA	1998	2	256	128?	1216?	Hash Collisions (2001)	2^{82} операции
Phelix	преди 2004	до 8 (W_{x86})	256 + a 128-bit Nonce	128?	***?***	N/A (2005)	N/A (2005)
Py	преди 2004	2.6	8-2048? обикновено 40-256?	64	8320	Cryptanalytic Theory (2006)	2^{75} операции
Rabbit	2003-(Февр)	3.7(W_{P3})-9.7(W_{ARM7})	128	64	512	N/A (2006)	N/A (2006)
RC4	1987	изключително бърз	8-2048 обикновено 40-256	8	2064	Shamir Initial-Bytes Key-Derivation OR KPA	2^{13} или 2^{33}

Поточен шифър	Дата на създаване	Скорост (цикли / байт)	(битове)			Атака	
			Ефективна ключова дължина	Вектор за инициализация	Вътрешно състояние	Най-известна	Изчислителна сложност
Salsa20	преди 2004	8.91 (W_{G4}) - 16.44 (W_{P4})	256 + a 64-bit Nonce	512	512 + 384 (key+IV+index)	Differential (2005)	N/A (2005)
Scream	2002	4 - 5 (W_{soft})	128 + a 128-bit Nonce	32?	64-bit round function	*?*	*?*
SEAL	1997	много бърз ($W_{32\text{-}bit}$)	*?*	32?	*?*	*?*	*?*
SNOW	преди 2003	много добра ($W_{32\text{-}bit}$)	128 или 256	32	*?*	*?*	*?*
SOBER-128	2003	*?*	до 128	*?*	*?*	Message Forge	2^{-6}
SOSEMANUK	преди 2004	много добра ($W_{32\text{-}bit}$)	128	128	*?*	*?*	*?*
Trivium	преди 2004	4 (W_{x86}) - 8 (W_{LG})	80	80	288	Brute force attack (2006)	2^{135} операции
Turing	2000-2003	5.5 (W_{x86})	*?*	160	*?*	*?*	*?*
VEST	2005	42 (W_{ASIC}) - 64 (W_{FPGA})	променлива обикновено 80-256	променлива обикновено 80-256	256 - 800	N/A (2006)	N/A (2006)
WAKE	1993	бърз	*?*	*?*	8192	CPA & CCA	голяма

3. Заключение

На базата на всичко, изложено до тук, могат да се направят следните изводи.

1. В съвременната криптографията се търси компромис между две противоречиви изисквания – бързодействие и сложност. Това е така, защото колкото е по-сложна процедурата по шифриране, толкова е по-криптоустойчива системата. От друга страна колкото е по-сложна процедурата за шифриране, толкова е по-сложна процедурата за дешифриране, при което се намалява скоростта на предаване на данните (поради голямото време за тяхната обработка) и се повишават разходите за практическа реализация и поддържане на системата.

Ето защо в съвременните системи за поточно шифриране висока криптоустойчивост при приемливо бързодействие и цена се постига като "парциалните" гами, формирани от няколко паралелно работещи бързи и евтини ПРЛОВ или ПРОВП, се обединяват по подходящ начин. Този основен подход най – общо се нарича метод на комбиниращия генератор.

2. Целта на комбиниращите генератори е да се спази изискването на ***Клод Шенон*** – между открития текст и криптограмата не трябва да има статистическа връзка.

3. Не съществува поточен шифър идеален за всяка задача, дори и такъв предлагащ високо ниво на сигурност. Като правило ефикасността винаги се пренебрегва за сметка на сигурността.

4. Шифърът осигурява отлична сигурност, ако шифротекстът и откритият текст са статистически независими. Критерии за практическа оценка на сигурността на поточните шифри са:

1. очаквано ниво на сигурност;
2. дължина на ключа;
3. производителност;
4. размер на блока;
5. сложност на криптографската операция;
6. разширяване на данните;
7. разпространение на грешки.

Литература:

1. Bedzhev B. Y., Tasheva Zh. N., Stoyanov B. P., N-adic Summation-Shrinking Generator. Basic properties and empirical evidences, Cryptology ePrint Archive, http://eprint.iacr.org/2005/068.pdf

2. Bedzhev B. Y., Tasheva Zh. N., Stoyanov B. P., P-adic Shrinking–Multiplexing Generator, Proceedings of the Third IEEE Workshop IDAACS'2005, Sofia, Bulgaria, 05-07.09.2005, pp. 443-448

3. Bedzhev B. Y., Tasheva Zh. N., Stoyanov B. P., Vasileva Sv., Application and Statistic Testing of the 5-adic Summation – Shrinking Generator, The ACM Digital Library (The Guide to Computing Literature)

4. Bedzhev B. Y., Tasheva Zh. N., Tashev T., The generalized Shrinking – Multiplexing Generator, The ACM Digital Library (The Guide to Computing Literature)

5. Bedzhev B. Y., Tashev T. D., Analysis of the WMAN Security Problems, Proceedings of the International Scientific Conference of the Military Technical Academy, Bucharest, 01-02.11.2007, ISBN 978-973-640-127-5, Section 8, pp.8.61 – 8.67

6. Menezes A., van Oorschot P., Vastone S., Applied Cryptography, CRC Press, 1996.

7. Sklar, B., Digital Communications. Fundamental and Applications (Second Edition) – Prentice Hall PTR, 2002.

8. Жельников В. "Криптогрофия от папируса до компютера", М. ABF, 1997г.

9. Пенчева, Е. GSM комуникации, Нови знания, 2000.

10. Попов, М.К., Клетъчни радиотелефонни системи – С.: Св.Г.Победоносец, София, 1998.

11. Романцев Ю.В., Тимофеев П.А., Шангин В.Ф. "Защита информации в компьютерых системах и сетях", "Радио и связ", М., 1999г.

ИЗПОЛЗВАНЕ НА СИСТЕМАТА E-LEARNING SHELL В ОБУЧЕНИЕТО ПО ИНЖЕНЕРНИТЕ ДИСЦИПЛИНИ

Христо Христов
Цветослав Цанков

Anotation: In this paper one discus an apply possibilities of e-Learning Shell web based platform. We strongly recommend to the Professors use these platform in educational process.

Системата e-Learning Shell е WWW базирана платформа. Това дава възможност тя да се използва в новите условия за обучение, при които студентите от редовно обучение са задължени да посещават 50 % от аудиторните занятия.

1. Основни положения в съвременните методи и техники на преподаване на студентите по инженерните дисциплини.

ЛЕКЦИЯ - изнася се от преподавателя и се използва за предаване на голям по обем и сравнително труден по характер материал. Особеност в структурата на лекцията е ползването на метода на теоретичното доказателство.да се владее този метод означава да се владее от преподавателят доказателството като логическо средство както на равнището на формалната, така и на равнището на диалектическата логика. Това предплага:

- ясно и точно формулиране на теоретичното положение, чиято истиност или неистиност ще бъде доказана
- строг научен подбор на фактологичния материал на аргументите, които трябва да имат безспорна доказателствена стойност.
- подходящо ползване на формите на доказателството (индуктивни; дедуктивни; традуктивни)
- формиране на изводи, точно произтичащи от приведените аргументи.

БЕСЕДА - един от основните методи също с широко приложение в обучението. Реализира се като фронтално поставяне в определена последователност на въпроси от преподавателя към студентите - те могат да бъдат свързани с новото учебно съдържание, но основното им предназначение е да се рализира обмен на информация между всички страни в акта на обучение. Главната цел на беседата е да се постиген по задълбочено осмисляне на учебното съдържание и да серазкрият реалните посоки в които то може да бъде приложено. Предимство на метода е възможноста за активно включване в мисловната деиност на всички студенти.

Изисквания към въпросите:

- да бъдат логически свързани, така че сравнително пълно да търсят съответните отговори;
- да бъдат формулирани ясно, точно и не много-словно;
- да небъдат подсказващи или внушаващи;
- беседата да бъде управляем процес, тоест отгово-рите да се следят от всички и своевременно да се коригират и допълват при необходимост.

Има два типа беседа:

- ЕВРИСТИЧНА - когато се извлича ново знание възоснова на миналият опит
- КОНТРОЛНА (проверовъчна) - когато преподава-телят има за цел да установи степента на усвоеност на съответният материал.

ДИСКУСИЯ - характерът на учебното съдържание във философският цикъл и особенстите на студентската възраст са добри предпоставки за широкото им приложение. В практиката на обучението най-често се реализират импровизирани дискусии, те бързо приповдигат тонуса на учебната работа, но като явление трудно поддаващо се на ръководене, не оказват съществено влияние върху съдържателният аспект на обучението. По-целесъобразно е прилагането на ръководената от преподавателя дискусия, което предпо-лага предварителна организация от страна на преподавателя и студентите. Целесъобразно е да се ползва в хода на семинарите.

ПРЕЗЕНТАЦИЯ - изнася се от преподавателя, използва се за представяне на тема или идея. Студентите могат да си водят бележки и да участват в обсъждането.

ВОДЕНЕ НА БЕЛЕЖКИ ПОД РЪКОВОДСТВОТО НА ПРЕПОДАВАТЕЛЯ - той предава някакъв обем информация като показва (инструктира) обучаемите как да си водят бележки.

УПРАЖНЕНИЯ ЗА ЗАПОМНЯНЕ - предлагат се задачи специално насочени към превръщане на даден обем от информация в част от дълготрайната памет.

СЕМИНАР ПОД РЪКОВОДСТВОТО НА ПРЕПОДАВАТЕЛЯ - съдържа набор от упражнения, които се провеждат на някакъв етап с цел да въведат стъпка по стъпка някаква идея. Препоръчва се свободно общуване, както и водене на бележки от страна на студентите.

ПЪЗЕЛ – преподавателят поставя някаква задача с повишена трудност, като изисква от обучаемите да размишляват върху нея и на всяка цена да намерят някакво решение.

МОЗЪЧНА АТАКА - преподавателят задава въпрос и задължително записва всички отговори след това те се обсъждат, сравняват, обобщават и приемат с консенсус.

АКТУАЛИЗАЦИЯ - чрез подходящ набор от въпроси преподавателят цели да актуализира в оперативната памет по-рано усвоени знания.

ИНСТРУКЦИЯ – НАБЛЮДЕНИЕ - ОТЧЕТ – преподавателят поставя задача, проследява изпълнението и, като накрая го анализира.

РАБОТА В ГРУПА – преподавателят поставя пред всички задачи, които трябва да намерят решение чрез работа в групи от 3 до 5 човека.

УПРАЖНЕНИЕ ЗА КООПЕРИРАНЕ (работа в екип) - поставя се една задача в групата, чието решение изисква кооперирание на действията между участниците.

ИЗУЧАВАНЕ НА КАЗУС - на студентите се предста-вя някакъв казус, чието решение изисква от тях да демонстрират определени действия или идеи.

ФОРМАЛЕН ДЕБАТ - реализира се чрез дебатиране по правила върху предварително обявена тема.

САМОСТОЯТЕЛНО УЧЕНЕ (самоподготовка) - на студентите се дават конкретни указания и инструкции за самостоятелна работа по определени теми.

2. Възможности на системата e-Learning Shell.

Системата предлага следните основни възможности.

- начало на работа със системата: необходимо е се да попълни регистрационна форма.

- първоначално заявяване на: факултет, специалност, дисциплина;

- работа със структури от вида: факултети, специалност, дисциплина;

- различни права за достъп: администртор, преподавател, студент;

- смяна на парола.

Основен интерес в процеса на обучение представлява използването на структурата „дисциплина", която е предвидена в системата. След избор на дисциплина от порталната част, системата за управление на Web базираните дисциплини отваря съответния сайт. Web-базираната черупка е ориентирана към три групи потребители – преподаватели, студенти и гости. Най големи права имат преподавателите. За да използват тези права, те попълват регистрационна форма и след като получат одобрение от администратора могат да влезнат във владение на своето пространство.

Влизането след успешна регистрация става от диалогово поле, извикано от бутон „Регистрация", като се въведат име и парола.

Администратора е въвел дисциплините, които са за съответните специалности от отделните факултети. Достъпът до публикуваните материали се осъществява от менюто в лявата му част и предлага следните възможности:

- *Анотация* – тук можете да направите описание на изучаванта дисциплина, както и да напишете мотото на дисциплината;

- *Програма* – тази опция дава възможност за публикуване на учебната програма, темите или въпросите, които се разглеждат в нея, както и кратки коментарии по тях;

- *Литература* – можете да опишете литературата, която да ползват студентите. Желателно е това да става по БДС. Можете например да я разделите на задължителна и препоръчителна;

- *Лекции* – тук можете да публикувате вашите лекции, които изнасяте пред студентите. От гледна точка на съвременните педагогически ценности това е основния фактологически материал, необходим за подготовката на студентите. Всички твърдения, теореми, правила, съждения, както и най-големия обем информация се получава именно тук. Всеки преподавател е длъжен да предоставя отговорно лекциите, които публикуват. Обикновено те са във Word или PDF формат. **Не се препоръчва** това да са лекции или съдържание от издадени ваши учебници. В някои случаи е добре да се публикува даже само плана на лекцията, а съдържанието да се маркира с по няколко думи;

- *Тестове* - особено популярен способ за текущ контрол, самоконтрол и формиране на оценка в последните десетилетия са тестовите форми. Недвусмислията при въпросите в тестовете, както и другите задължителни моменти при съставянето им в тестологията можем да посочим основно като изключителни предимства в оценяването. Шаблонизирането при проверката на големи маси от тестове също се цени особено много в съвременната педагогическа дейност. Те се разработват по учебното съдържание на изучаваната дисциплина. За всеки един тест е удачно да се разработят по 25 – 30 въпроса. Когато студентите решават даден тест, то черупката сама подбира по случаен начин въпросите и съставя теста – състоящ се от 8 въпроса;

- *Упражнения* - в препоръчителните формати може да качите файловете в които са упражненията по съответната дисциплина. Основно те са във вид на задачи, разработени специално за конкретна тема, допълваща знанията по нея и целяща студентите евристически да достигнат до някои умения за постигане на целите, посочени в условията;

- *Задачи* – те също могат да са във Word или PDF формат, обновявайте и редактирайте своите материали, посочва се датата

на обновяването и обикновено когато се види, че файл е отдавна не е със голяма тежест. За разлика от упражненията, задачите могат да се поставят за самостоятелна работа с конкретни за всеки обучаем стойности, резултатите от тях, попълнени в курсови работи могат да послужат във формирането на успеха на студентите;

- *График* – тук в табличен вид можете да укажете седмичен график (за четна и нечетна) седмица, на провеждане на занятията по дисциплината;

- *Конспект* – той се пренася директно в текстово поле, при необходимост може да се отпечата, но не и да се копира и да се редактира злонамерено от студентите;

- *Преподавател* – тук се публикува кратка информация за водещия преподавател и асистента (ако има такъв);

- *Съобщения* – тук е добре да се следи рубриката за да могат да се информират студен-тите за важни събития, настъпили, забавени или нововъведени, свързани към учебния процес;

- *Форум* – удобна кореспонденция точно на хора, които имат отношение към конкретната дисциплина, спестява се много писане или търсене на имена и адреси на запознати с въпроси, без да се ангажира вниманието на преподавателя;

- *Статистика* – преподавателя може да наблюдава успехи от тестове, посещения и други опции, като по този начин може да определи доколко даден студент се е подготвял;

- *Изход*.

3. Възможности за прилагане на системата e-Learning Shell в съответствие с методиката за обучение по инженерни дисциплини.

От гледна точка на методика за придобиване на инженерни знания, системата e-Learning Shell предлага:

А) възможности за натрупване на познание – това става с опциите:

- Програма;
- Литература;
- Лекции;

Б) Знания – това са опциите:

- Задачи (преглед и разработка на реферат по тема),

- Съобщения (обсъждане на учебния материал)

В) Прилагане на знанията – това е възможно при:

- Задачи (курсови задачи, курсов проект, разработка на практически реферат, решаване на реален проблем);

Г) Оценяване на обучаемия – в опции като:

- Тест – пряко текущо оценяване. Разработването на тестове по темите от програмата дава възможност без участието на преподавателя всеки сам да покаже своите знания. Това е много удобна опция за текущ контрол, чиято роля в последните години рязко нараства.

- Задачи – оценяване на курсови задачи, курсов проект или реален проблем.

- Статистика – непряка преценка за работата на студента. С тази опция може да се проследи кога и какво е правил студента – чел е лекции, решавал е курсови задачи, правил е тест и т.н.

4. Изводи и заключения.

От всичко казано дотук може да се направят следнитеосновни изводи:

Системата e-Learning Shell не може да обхване 100 % от методика за обучение в инженерните дисицплини. Тя се явява много полезно помощно средство в процеса на обучение, като дава възможност за:

- подпомага преподавателя при водене на учебния процес;

- подпомага учебния процес на студентите, като на практика дава възможмост и за дистанционно обучение;

- облекчава работата на преподавателя и дава възможност за по-добро и обективно оценяване текущите знания на студентите.

ИЗСЛЕДВАНИЕ И ПРЕСМЯТАНЕ НА КИНЕМАТИЧНА СХЕМА НА СПЕКТРОФОТОМЕТЪР

Стилиян Стоянов
Христо Христов

Annotation: Examine principle of work on very popular devices of these type one discuss a possible design of optoelectronic device with diffraction lattice.

1. Принципна схема.

Принципната схема на уред с дифракционна решетка е изградена на базата на приемна оптична система с обектив, сферично огледало, дифракционна решетка и оптичен компонент пред сензор. Входният сигнал чрез приемна оптична система се насочва върху сферичното огледало, от което след отразяване във вид на успореден сноп лъчи, същите се дифрагират от дифракционната решетка и чрез оптичния компонент се концентрират върху фотокатода на сензор. При завъртане на дифракционната решетка около ос, разположена под ъгъл φ спрямо оптичната ос на сферичното огледало, се осигурява сканиране в определен диапазон по дължина на вълна.

В качеството на диспергиращ елемент в спектралните уреди с дифракционна решетка се използват различни типове решетки – плоски, вдлъбнати, с различна форма и размери и брой на щрихите. Спектрофотометър с плоска дифракционна решетка позволява промяна на ъгъла на падане на снопа лъчи, постъпващи от сферичното огледало върху дифракционната решетка чрез промяна на нейното положение, при което се получава използването на един сензор вместо система от такива.

Падащият и дифрагираният сноп лъчи се намират в главната (меридионална) равнина на решетката, т. е. в равнина, перпендикулярна на щрихите на решетката. Приема се, че ъгълът на падане ψ и ъгълът на дифракция φ имат еднакви знаци, тъй като лежат от едната страна на нормалата към решетката.

2. Кинематична схема.

Кинематичната схема на оптоелектронен спектрофотометър е представена на фиг. 19. Основните елементи на уреда са: корпус (1), водещо зъбно колело (2), стъпков електродвигател (3), приемно плоско огледало (4), втулка (5), водимо зъбно колело (6), плоско отразяващо огледало (7), приемно сферично огледало (8), подвижен диск (9), стъпков електродвигател (10), водещи зъбни колела (11), цилиндрична винтова пружина (12), водещо рамо (13), дифракционна решетка (14), сферично огледало (15), оптична система (16), сензор (17), ексцентрично рамо (18), водими зъбни колела, корпус на задвижващия механизъм.

Уустановен е задвижващ механизъм, състоящ се от стъпков електродвигател (10), водими зъбни колела (11), цилиндрична винтова пружина (12), водещо рамо (13) и ексцентрично рамо. Задвижващият механизъм управлява въртенето на плоската дифракционна решетка (14).

Задвижването на приемното плоско огледало (4) се извършва от стъпков електродвигател (3) чрез водещо зъбно колело (2), водимото зъбно колело (6) и втулката (5), неподвижно свързана с огледалото.

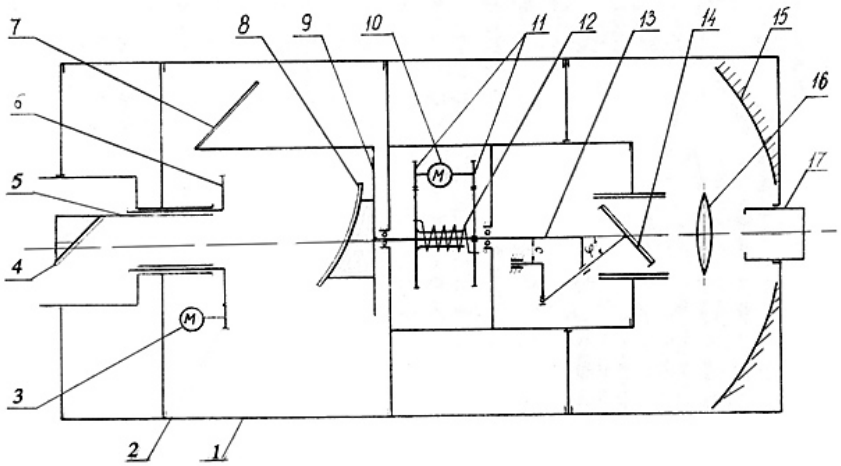
Втулката позволява на огледалото да заема две гранични положения – транспортно (прибрано огледало) и работно (изнесено извън корпуса огледало).

Електродвигателят осигурява осцилиращо движение на огледалото, при което се дава възможност за сканиране на дадената повърхност.

Дифракционната решетка с помощта на два сачмени лагера лагерува във втулка, образуваща плъзгащ лагер с корпуса на задвижващият механизъм.

Светлинният поток попаднал върху плоското огледало (4) се отразява от сферичното огледало (8), плоското огледало (7), сферичното огледало (15) и попада върху плоската дифракционна решетка (14), от която дифрагиран през оптичната система (16) постъпва към сензор (17) за регистриране. От различните ъгли на падане на светлинния поток върху дифракционната решетка (14) се осигурява различен спектър на лъчението, попадащо върху

фотокатода на сензор. Регистрираната спектрална линия се идентифицира от положението на задвижващо устройство и датчика за положението.



Фиг. 1. Кинематична схема на спътников спектрофотометър

При въртенето на задвижващото рамо (13) от електродвигателя (10), зъбни колела (11) се привежда в движение и носача на решетката (14) като се завърта около собствената си ос и се накланя под различен ъгъл φ спрямо централната ос на уреда. По такъв начин се осъществява сканиране на светлинния поток в даден диапазон на спектъра.

Наличието на двете зъбни предавки (11), свързани чрез пружината (12), работеща на усукване позволява да се обере хлабината в зъбните предавки, с което се осигурява повишаване точността на уреда.

3. Кинематични изчисления.

Въз основа на фиг. 1 за синусния механизъм се получава:

(1) $$\sin\varphi_{\max} = \frac{R+e}{L},$$

(2) $$\sin\varphi_{\min} = \frac{R-e}{L},$$

където: L - дължина на рамото на дифракционната решетка.

От тук могат да се определят стойностите на радиуса R от оста на ексцентричния диск до плъзгащата връзка и разстояние e на оста на диска на уреда:

(3) $$R + e = L\sin\varphi_{\max,}$$

(4) $$R - e = L\sin\varphi_{\min},$$

(5) $$R = \frac{L}{2}\left(\sin\varphi_{\max} + \sin\varphi_{\min}\right),$$

(6) $$e = \frac{L}{2}\left(\sin\varphi_{\max} - \sin\varphi_{\min}\right).$$

Пресметнати са стойностите на R и e ($\theta = 8^0 03$) при вариране отново с λ, ρ и две разновидности на рамото *L=100 mm* и *L=200 mm,* и са извършени аналогични пресмятания при ъгъл $\theta = 4^0 01$.

Въз основа на извършените оптични изчисления се избира дифракционна решетка с диапазон на $\lambda = 200...600\ nm., \rho = 1302\ щрихи/mm.$

Поради това, че има мъртъв участък от реброто, който не позволява използването на пълен обхват от $0...180^0$, е необходимо ъглите $\varphi_{\max}$ и $\varphi_{\min}$ да бъдат съответно корегирани, като се увеличи ($\varphi_{\max}$) и намали ($\varphi_{\min}$), поради което радиусът R се закръгля на *R=32 mm*, а разстоянието *e=14 mm.*

4. Определяне свойствата на уреда.

Основни характеристики и параметри, които се определят при проектиране на уреда и характеризират изискванията към него са:

Точност на уреда. Характеризира се с допустимата грешка на измерване при зададени условия и методика на измерване, с инструментална грешка (абсолютна или относителна, гранична или средно квадратична).

От основното уравнение след диференциране се получава:

(7) $$dR + de = dL\sin\varphi_{\max} + L\cos\varphi_{\max} d\varphi_{\max} .$$

Ако се замести диференциала с технологичния допуск на размерите на изпълнените детайли, същността на уравнението (7) няма да се измени и може да се изчисли грешката, която ще се получи на ъгъл φ в резултат на грешките на размерите *R, e, L* :

(8) $$TR + Te = TL\sin\varphi_{\max} + L\cos\varphi_{\max} .T\varphi$$

където: *TR* - допуск на изработване на *R*;
Te - допуск на ексцентрицитета;
TL - допуск на дължината *L* на рамото на решетка;
$T\varphi$ - допуск на установяване на ъгъл φ .

От тук:

(9) $$T\varphi = \frac{TR + Te - TL\sin\varphi_{\max}}{L\cos\varphi_{\max}} .$$

Ако се вземе пред вид, че плъзгащата връзка ще бъде реализирана с хлабина (евентуално сглобка от порядъка на $\varnothing ... F8/h7$) хлабината във връзката също ще окаже влияние върху точността на ъгъла φ , при което при максимална хлабина φ max грешката на радиуса вече ще бъде:

$$TR^{|} = TR + Jmax,$$

вследствие на което (9) ще получи вида:

(10) $$T\varphi = \frac{TR' + Te - TL\sin\varphi_{\max}}{L\cos\varphi_{\max}} .$$

Като се заместят съответните стойности за:

TR	=	0,025	При *R=32mm* и 7 клас на точност;
J_{max}	=	0,030	За сглобка $\varnothing 3F8/h7$;
Te	=	0,018	За *e=14mm* и 7 клас на точност;
TL	=	0,035	При*L=100mm*.

$$T\varphi = \frac{0{,}025 + 0{,}030 + 0{,}018 - 0{,}035.0.455169}{100.0{,}890405},$$

$$T\varphi = 6{,}4.10^{-4}\left[rad\right],$$

$$T\varphi = 0^0 2'12''.$$

Допустимата грешка на мъртвия ход според техническата характеристика трябва да бъде $\pm 3'$, следователно възможната грешка е в допустимите граници.

Литература:

1. П. Гецов. Спътникови системи за екологичен мониторинг. Сб. Научни Трудове „Научно-технологичен трансфер" ИКИ – БАН. Шумен, 2000 год.
2. Гецов П., Жеков Ж. и др. Влияние параметрите на оптико-електронните уреди върху скоростта на сканиране на пространството. Сб. Труд. на Технич. колеж. Смолян. 2003 год.
3. Кузнецов Г.И. Опыт исползования комплекса апара-туры для атмосферно-оптических наблюдений озона и озо-ноактивных компонент атмосферы. Тр. совещ. по озону. Москва. 2000 год.
4. Мардиросян Г. Аерокосмически методи в екологията и изучаването на околната среда. Акад. Издат. „Марин Дринов". 2003 год.
5. Чуриловский В.Н. Теория оптических приборов. Машиностроение. М-Л. 1996 год.
6. Mardirossian G. Ecological Studies Ranking Among the Top-Priority Research Areas of the Space Research Institute at BAS. Годишник на Техническия Университет – Варна. 2001 год.

ИЗПОЛЗВАНЕ НА GRIB ДАННИ В ИНТЕРЕС НА ПОВИШАВАНЕ НА ТОЧНОСТТА ПРИ ОПРЕДЕЛЯНЕ НА МЕСТОПОЛОЖЕНИЕ ЧРЕЗ ИЗПОЛЗВАНЕ НА СПЪТНИКОВИ РАДИОНАВИГАЦИОННИ СИСТЕМИ

Мирослав Цветков

USING GRIB DATA TO CALCULATE TROPOSPHERE CORRECTIONS FOR GNSS

Miroslav Tsvetkov
Naval Academy "N. Y. Vaptsarov"

Анотация: В настоящия доклад е представен един подход за изчисляване на стойността на тропосферното закъснение на радиосигналите, излъчвани от навигационните спътници на спътниковите радионавигационни системи (СРНС). Целта е, изчислената грешка да бъде предавана като корекция към GPS приемниците на потребителите за коригиране на измерваните псевдоразстояния към навигационните спътници.

Abstract: This paper represents an approach to calculate a value of the troposphere error of radio signals of navigational satellites of Global Navigation Satellite Systems. The output data can be used as troposphere corrections to the measured pseudoranges form GPS receivers.

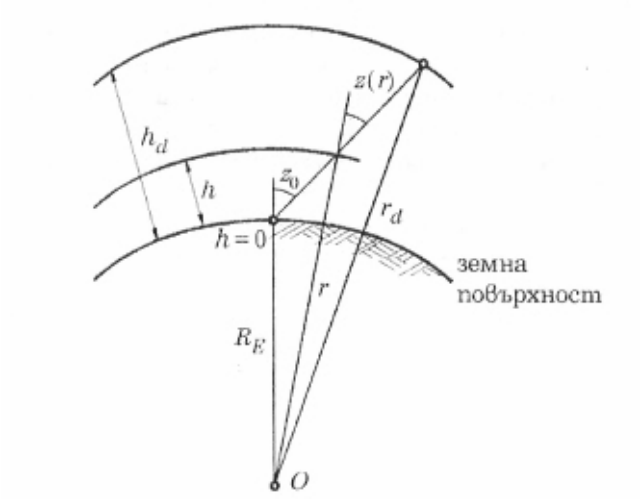
Ключови Думи: GPS, GRIB, MatLab, спътникови радионавигационни системи, тропосфера;

Key words: GPS, Global Navigation Satellite Systems, GRIB, MatLab, Troposphere;

Нарастващото използване на спътниковите радио-навигационни системи поставя пред тях предизвикателството за

осигуряване на все по-висока абсолютна точност на потребителите си. Един от факторите, който влияе на точността при определяне на местоположение чрез използване на СРНС е влиянието на неутралната атмосфера.

Влиянието на неутралната атмосфера (т.е. на не йонизираната част) е известна като тропосферна рефракция или просто тропосферно закъснение. Неутралната атмосфера е не дисперсна среда за радиовълни с честоти до 15 GHz, което означава, че разпространението на GPS радиосигналите е честотно независимо. Съгласно теорията на Хопфийлд, тропосферата може да бъде разделена на суха и мокра, където сухата компонента се получава от сухата атмосфера, а мократа – от водните пари. Около 90% от тропосферното закъснение се получава от сухата компонента, която е функция на температурата и атмосферното налягане, и около 10% от мократа компонента. Тропосферното закъснение геометрично е представено на фиг. 1.



Фиг.1 Геометрично представяне на тропосферното закъснение

Най-общо закъснението на радиосигналите при преминаването им през тропосферата може да бъде изразено чрез формула 1.

(1) $$\Delta_{TRO}[m] = ZHD.MF_H + ZWD.MF_W ,$$

където, ZHD е закъснението във вертикално (зенитно) направление причинено от сухата компонента, ZWD е закъснението във вертикално (зенитно) направление причинено от мократа компонента. За изчисляване на тропосферното закъснение на радиосигнали приемани от произволен ъгъл на възвишение е необходимо въвеждането на коефициенти отчитащи стойността му. Във формула 1 това са MF_H и MF_W съответно за сухата и мократа компоненти. MF_H и MF_W най-често се наричат предавателни функции (mapping functions), или функции на изображение.

За изчисляване на тропосферното закъснение са изведени различни тропосферни модели и предавателни функции [4, 5, 7]. Грешката от тропосферното закъснение за разлика от грешката от йоносферната рефракция не може да бъде елиминирана чрез методи използващи две или повече честоти. Грешката от тропосферното закъснение не търпи съществени (резки) промени в кратки периоди от време (няколко минути), освен това тя може напълно да бъде отстранена чрез използване на диференциални техники.

В таблица 1 е показано разпределението на основните грешки влияещи върху точността на определяне на местоположение, чрез използване на СРНС.

Таблица 1 Разпределение на основните типове грешки влошаващи точността при определяне на местоположение чрез СРНС

Източник на грешка	**Символ**		**Стойност (1σ)**	**Стойност [%]**
Йоносферна рефракция	**URE**	Δ_{ION}	7 m	45,75
Координатно-времево обезпечаване на НС		Δ_{SV}	3,6 m	23,53
Тропосферна рефракция		Δ_{TRO}	2 m	13,07
Многопътност	**UEE**	Δ_{MP}	1,2 m	7,84
Шум на приемника		Δ_{R}	1,5 m	9,81
Всичко		Δ_{PR}	**15,3 m**	**100**

Съгласно таблица 1 абсолютната стойност на грешката предизвикана от тропосферната рефракция е около 2 метра при ниво на доверителна вероятност 1σ и представлява около 13% от еквивалентната средноквадратична грешка при определяне на разстояние до навигационен спътник (User Equivalent Range Error – UERE). Еквивалентната средноквадратична грешка се определя чрез формула 2,

$$UERE = \sqrt{(URE^2 + UEE^2)}\,, \tag{2}$$

където, URE (User Range Error) грешка в разстоянието от потребител до навигационен спътник е сума от грешките причинени от йоносферната рефракция, тропосферната рефракция и координатно-времевото осигуряване на навигационния спътници; UEE (User Equipment Error) е сумата на грешките включващи: многопътност и шум на приемника на потребителя, и зависи предимно от техническите характеристики на приемника и антената на потребителя.

Казано с други думи, при редуцирането или елиминирането на грешката причинена от тропосферната рефракция може да се постигне повишаване на абсолютната

точност при определяне на местоположение чрез използване на спътникови радионавигационни системи с до 13%.

Като най-точен, според литературните източници [4, 5, 7] се посочва моделът на Хопфийлд, разработен за нуждите на коригирането на GPS измерванията, провеждани при геодезически изследвания. За входни данни се използват текущите стойности на температурата на въздуха, атмосферното налягане и относителната влажност.

Гореизложеното показва, че за изчисляване на грешката е необходимо да се осигурят данни за текущите стойности на температурата на въздуха, относителната влажност и атмосферното налягане.

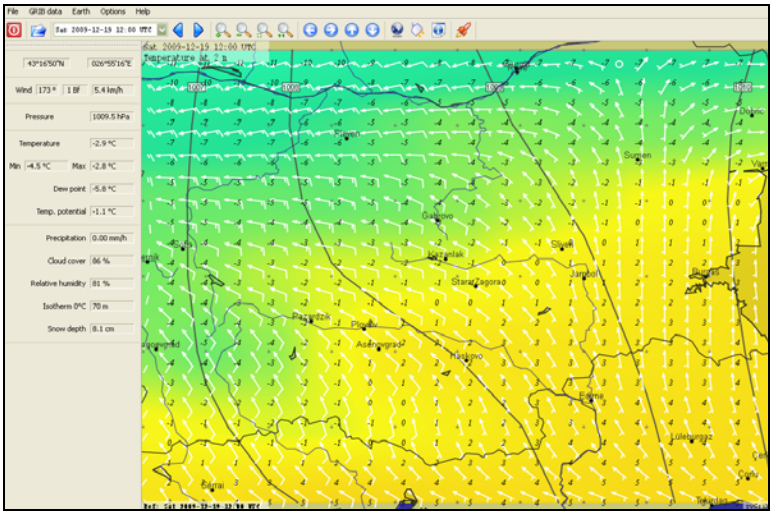
Основен източник на данни за тропосферата е Световната метеорологична организация [3], която събира метеорологични данни от метеорологични станции разположени по цялото земно кълбо.

Събраните данни се разпространяват безплатно чрез интернет кодирани в двоичния GRIB (Gridded Binary Format) формат. GRIB форматът притежава съществени предимства пред традиционните формати GRID и GRAF. Основните предимства на този формат и по-специално на разширения GRIB издание 2 (GRIB2) се заключават в неговата гъвкавост и описателност, които са ключови за научно-изследователски и технически развой [8]. В таблица 2 е показано съдържанието на форматите GRIB1 и GRIB2.

Таблица 2 Съдържание на формати GRIB1 и GRIB2

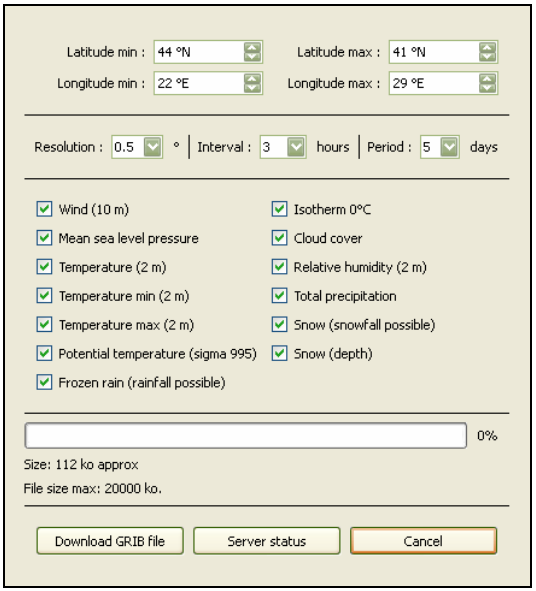
GRIB1	**GRIB2**
SECTION 0 Indicator section SECTION 1 Product definition section SECTION 2 (Grid description section) SECTION 3 (Bit-map section) SECTION 4 Binary data section SECTION 5 7777 (End section)	SECTION 0 Indicator Section SECTION 1 Identification Section SECTION 2 (Local Use Section) SECTION 3 Grid Definition Section SECTION 4 Product Definition Section SECTION 5 Data Representation Section SECTION 6 (Bit-map Section) SECTION 7 Binary Data Section SECTION 8 End Section

За разлика от своя предшественик GRIB1, форматът GRIB2 се състои от осем секции, което позволява да се предава допълнителна информация (нови продукти), като дългосрочни прогнози, климатични прогнози и други изходни данни от модели за предсказване. Изображение на текущи и прогнозни данни от GRIB2 файл са показани на фигура 2. Изображението е синтезирано от специализирания софтуерен продукт zyGRIB версия 3.4.0.



Фиг. 2 Изображение на GRIB2 данни за района на Република България

Данните са заредени от сървъра на NOAA (National Oceanic and Atmospheric Administration). Конфигурацията на zyGRIB за заявката за GRIB данни е показана на фигура 3. Зададен е районът, чрез използване на географските координати на точки образуващи правоъгълник включващ територията на Република България. Зададена е стъпката на точките от мрежата (в случая точките са зададени през 0,5º), интервалът от време (през 3 часа) и времевия период (пет денонощия). Отбелязани са общо тринадесет параметъра, който включват и температура на въздуха, относителна влажност и атмосферно налягане.



Фиг. 3 Конфигуриране на заявка за GRIB данни чрез zyGRIB

За интерпретиране и обработка на GRIB2 данни с цел осъществяване на научно-изследователска и приложна дейност може да се използва безплатния инструмент matlab-cdi toolbox на софтуерния продукт MatLab [2].

След като бъдат извлечени и преобразувани в подходящ формат, те се подават към алгоритмите за изчисляване на тропосферното закъснение във вертикално направление, а с отчитане на зенитните ъгли към навигационните спътници (формула 1) се изчислява точната стойност на тропосферната грешка за всеки навигационен спътник.

Така изчислената стойност може да бъде кодирана в формат RTCM SC-104 [6] и да бъде подадена като корекция към GPS приемник.

Заключение:

Чрез гореизложеният подход се осигурява един начин за елиминиране на негативното влияние на неутралната атмосфера по отношение на точността на определяне на местоположение

чрез използване на спътникови радионавигационни системи. Използвайки този подход се очаква точността на GPS позиционирането да се повиши с до 18%.

Използвана литература:

1. http://gps.solution-pro.net
2. http://www.sat.ltu.se/docs/matlabgrib/
3. http://www.wmo.int/
4. Jiming Guo, Richard B. Langley, A New Tropospheric Propagation Delay Mapping Function for Elevation Angels Down to 2 deg
5. Kyle O'Keefe, Comparison of Troposphere Mapping Functions, University of Calgari, 2001
6. RTCM Special Committee No.104, RTCM Recommended Standards for Differential GNSS, ver.2.2, 1998
7. Torben Shulter, On Ground-GPS Tropospheric Delay Estimation, University of Munhen, 2001
pages/prog/www/WMOCodes/Guides/GRIB/GRIB1-Contents.html
8. WORLD METEOROLOGICAL ORGANIZATION, Introduction to GRIB Edition1 and GRIB Edition 2

ВЪРХУ РЕЗУЛТАТИТЕ ОТ ТРАНСФОРМИРАНЕТО НА ТОЧКИ, РЕЗУЛТАТ ОТ *GPS* ИЗМЕРВАНИЯ

Пламен Михайлов

ON THE RESULTS OF THE TRANSFORMATION OF POINTS, A RESULT OF THE MEASUREMENTS WITH GPS

Ass. Prof. D-r eng. Plamen Mihajlov

ABSTRACT: An analysis of the results from two instances of transformation of geodetic coordinates B, L, H on the ellipsoid WGS8, a result from the measurements with GPS in plane coordinates and heights in coordinate system 1970, is made. For the purpose, first, the mathematic formulae and the transformation order in coordinate system 1970 are given. Then, the results from the transformation in the two instances are examined: the first one uses the coordinates of the supporting points in coordinate system WGS84, given by the Cadastre agency, and the second one – the coordinates of the same points a result of GPS measurements. The obtained results are clearly almost identical in terms of the coordinates of the points and the heights in coordinate system 1970.

Key words: GPS, transformation, geoid,, quasigeoid

1. Трансформиране на координатите на точки от координатна система WGS84 в координатна система 1970 год.

Повод за тази публикация стана изискването да се разполага с координати в координатна система WGS84 на точките от ДГМ, използвани за трансформирането на резултатите в координатна система 1970 г. Тези координати се получават от Агенцията по кадастър. Получи се така, че в момента не разполагахме с такива, и се наложи да изчакаме. Замислих се, че при метода за трансформиране, реализиран от мен, равнинните координати на точките се получават чрез афинна трансформация.

При това положение дори и да съществуват някакви разлики в равнинните координати на точките в система WGS84, след афинната трансформация не би трябвало да ги има.

Но първо накратко ще разгледам реда за трансформиране на резултатите от измерванията:

В резултат от *GPS* измервания се получават елипсоидните координати и височина (*B*, *L*, *H*) на точките, или техните пространствени правоъгълни декартови координати (*X, Y*, *Z*) в координатна система WGS84. В този случай е необходимо да се трансформират резултатите в координатна система 1970 год. Разработеният от мен метод включва следните основни моменти:

1. Получаване на равнинните правоъгълни координати в координатна система 1970 год.

А.) Преобразуване на елипсоидни координати и височина (*B*, *L*, *H*) в равнинни правоъгълни координати за елипсоида *WGS 84*.

Б) Трансформиране на равнинните правоъгълни координати в координатна система 1970 год.

2. Получаване на нормалните (надморски) височини на точките.

Ще се спра накратко на всеки от тях.

1 А). Преобразуване на елипсоидни координати (*B*, *L*) в равнинни правоъгълни координати [2]. За целта се използва напречна Меркаторова проекция (Гаус – Крюгерова проекция) в 6 градусова зона на основата на параметрите на елипсоида *WGS 84*. Формулите имат следния вид [7]:

(1)

$$x = S(B) + \frac{1}{2} N.\cos^2 B.t.l^2 + \frac{1}{24} N.\cos^4 B.t.\left(5 - t^2 + 9\eta^2\right)l^4 + ...$$

(2)
$$y = N.\cos B.l + \frac{1}{6} N.\cos^3 B.\left(1 - t^2 + \eta^2\right)l^3 +$$
$$+ \frac{1}{120} N.\cos^5 B.\left(5 - 18t^2 + t^4\right)l^5 ....$$

където:

- $S(B)$ ... дължина на дъгата по меридиана от екватора
- N ... радиус на кривина в първи вертикал
- L_0 ... дължина на осовия меридиан
- $l = L - L_0$ ... разлика в дължините
- $t = \tan B$ и $\eta^2 = e'^2 \cos^2 B$ ... помощни величини

Дължината на дъгата по меридиана $S(B)$ е елипсоидното разстояние от екватора до точката и се изчислява по следната формула:

(3)
$$S(B) = a.\left[B + \beta \sin 2B + \gamma \sin 4B + \delta \sin 6B + ...\right]$$

където:

$$a = \frac{a+b}{2}\left(1 + \frac{1}{4}n^2 + \frac{1}{64}n^4 + ...\right);$$

$$\beta = -\frac{3}{2}n + \frac{9}{16}n^3 - \frac{3}{32}n^5 + ...$$

(4) $$\gamma = \frac{15}{16}n^2 + \frac{15}{32}n^4 + ...; \qquad \delta = -\frac{35}{48}n^3 + \frac{105}{256}n^4 - ...$$

и $$n = \frac{a-b}{a+b}$$

1 Б). Трансформиране на равнинните правоъгълни координати в координатна система 1970 год [2]. За целта се използва афинна трансформация с 6 параметъра. За всяка точка с известни координати се съставят по две уравнения от вида:

(5) $$Xpr = Xu.a_1 + Yu.b_1 + c_1;$$
$$Ypr = Xu.a_2 + Yu.b_2 + c_2$$

Въвеждат се приблизителни стойности на параметрите $a_1 = 1$; $b_2 = 1$ и за всяка точка се съставят по две уравнения на поправките от вида:

(6) $$V_X = Xu + Yu + 1 + 0 + 0 + 0 + f_X;$$

$$V_Y = 0 + 0 + 0 + Xu + Yu + 1 + f_Y$$

със свободни членове разликата между изчислената и истинската стойност на съответната координата.

При това положение са необходими минимум три точки с известни координати, за да се реши задачата. За да има контрол е добре те да бъдат по-вече. Изравнените стойности на параметрите се получават след изравнение по МНМК. С така получените трансформационни параметри се трансформират координатите на всички точки в координатна система 1970 год.

2. Получаване на нормалните (надморски) височини на точките. За да се реши този проблем, се изхожда от връзката между нормалните и елипсоидните височини на точките [1, 3, 4].

(7) $$H = H^N + \zeta$$

където

H – геодезическа височина (височината над международния елипсоид)

$\boldsymbol{H^N}$ – нормална височина (превишение);

$\boldsymbol{\zeta}$ – аномалия на височината.

Проблемът е, че не се знае аномалията на височината за дадения район. Той може да се реши, като се въведе сравнително опростен модел на квазигеоида, които да важи за дадена област, неголяма по площ, като например,

(8) $$\zeta = C_0 + C_1.X + C_2.Y$$

Където:

- $X = \dfrac{M.(B - B_0)}{\rho}$; $\qquad Y = \dfrac{N.\cos B_0.(L - L_0)}{\rho}$

- ***M*** и ***N*** са съответно меридианният и напречният радиуси на кривина на сечението.

- ***B*** и ***L*** са геодезическите географски координати на точката от района,

- $\boldsymbol{B_0, L_0}$ - геодезическите координати за някаква средна точка за територията.

- $\boldsymbol{C_0}$, $\boldsymbol{C_1}$, $\boldsymbol{C_2}$ са коефициенти, характерни за този район.

Съгласно (7) и (8) може да се напише за геодезическата височина на дадена точка

(9) $$H_i = H^N + C_0 + C_1.X + C_2.Y$$

От тук превишението между тях ще бъде:

(10) $$H_k - H_i = H^N_k - H^N_i + C_1.X_{ik} + C_2.Y_{ik}$$

Ако се означи измененението на аномалията на височината с d_ζ **:**

(11) $$d_\zeta = C_1.X_{ik} + C_2.Y_{ik}$$

за $X_{ik} = \frac{M.(B_k - B_i)}{\rho''}$; $Y_{ik} = \frac{N.\cos B_0.(L_k - L_i)}{\rho''}$

Съответно за всяко "измерено" геодезическо превишение ***dH*** между точките ***i*** и ***k*** ще е налице уравнение на поправките от вида:

(12) $$V_{ik} = -VH^N_i + VH^N_k + d_\zeta + f_{ik}$$

Където:

- V_{ik} е поправка към "измереното" геодезическо превишение;

- VH^N_i, VH^N_k са поправки към измерените геометрични (нормални) височини на точките ***i*** и ***k;***

- f_{ik} е свободен член, изчислен по формулата:

(13) $$f_{ik} = dH^N_{ik} - dH_{ik} + d_\zeta$$

За "измерени" геодезически превишения ***dH*** между точките се използват изчислени такива между текущата точка и всяка от опорните. Така съставените уравнения на поправките от вида 13 участват в изравнение по МНМК, в резултат от което се получават изравнените стойности на геометричните (нормални) височини на новите точки и поправки към "измерените" геодезически превишения между точките.

2. Сравняване на резултати от трансформирането на точки, резултат от *GPS* измервания.

За да проверя догадките си, се наложи да извърша трансформацията на мрежата, върху която работехме, по два начина. При първия случай превързах получените резултати в координатна система WGS84 чрез координатите на една от точките от ДГМ (**100018**), а при втория чрез получените от измерванията координати на едната референтна точка, която нарекох свободна система. Получиха се следните резултати:

1. В система WGS84

точка	B(°’”)	L(°’”)	H(m)
102172	431616.47051	274654.64086	363.761
102175	431337.71532	274807.61401	122.989
102179	431543.44885	274954.23042	349.745
107034	431731.40638	274717.18951	367.673
100018	431445.43462	274624.79943	116.754
100045	431626.70225	274848.15651	273.599
110221	431550.66428	274643.29560	220.440
110133	431609.10246	274908.78288	229.031

2. В свободна система

точка	B(°’”)	L(°’”)	H(m)
102172	431616.49152	274654.67491	364.784
102175	431337.73624	274807.64791	124.016
102179	431543.46960	274954.26421	350.788
107034	431731.42724	274717.22340	368.704
100018	431445.45554	274624.83335	117.782
100045	431626.72313	274848.19043	274.628
110221	431550.68595	274643.32979	221.517
110133	431609.12379	274908.81785	230.025

След сравняване на резултатите от таблица 1 и 2 се вижда се, че максималните разлики в координатата ***B*** на точката са около 0.021”, а в координатата ***L*** – 0.035”, което води до разлики в правоъгълните координати (WGS84) от порядъка на 65-75 см. Разликата в елипсоидните височини на точките е значително по-голяма и се движи около 100 см.

3. Разлики в равнината (WGS84) см.

точка	dx	dy	dl	dh(m)
102172	65.56	76.17	100.50	-1.023
102175	65.29	75.88	100.10	-1.027
102179	64.80	75.57	99.55	-1.043
107034	65.09	75.79	99.90	-1.031

100018	65.27	75.93	100.13	-1.028
100045	65.19	75.86	100.02	-1.029
110221	67.57	76.58	102.13	-1.077
110133	66.60	78.21	102.72	-0.994

На долната таблица са показани максималните разлики в координатите на точките след трансформацията. От всичко около 170 нови точки има разлики >1 см. в положението на точките в координатна система (1970 г.) на 11 точки, като същите са максимално 3.58 см., което е в рамките на допустимата точност за РГО.

По подобен начин стои и въпросът с получаването на нормалните височини на точките. От долната таблица се вижда, че се получиха максимални разлики от двете трансформации за височините на някои от точките от порядъка на 4-5 см., което е в рамките на допустимото. И това при положение, че разликата в елипсоидните височини е около 1м. Възможно е това да се дължи и на не добрата конфигурация на опорните точки.

4. Максимални разлики в равнината (1970 г.) см.

Точка	dX(см.)	dY(см.)	dl(см.)	dH(см.)
110052	1.20	0.20	1.22	-0.70
110133	1.70	2.50	3.02	-4.40
110134	0.10	2.60	2.60	-3.20
110137	-1.80	1.10	2.11	-2.30
110138	-1.50	-1.10	1.86	-2.10
110204	1.60	-3.20	3.58	-0.80
110217	-2.60	1.80	3.16	4.90
110221	2.20	0.50	2.26	5.10
110242	-0.70	-1.10	1.30	1.10
110255	0.30	-1.40	1.43	-1.10
110260	-0.90	-0.70	1.14	0.30

По този начин напълно се потвърди моята хипотеза. Освен това се потвърди и впечатлението, че за получаване на коректни резултати в не по-малка степен са необходими и координатите на

същите точки в координатна система 1970 г. За това, за да разполагаме с координати на точките в единна координатна система за територията на цялата страна, е необходимо да се определят координатите на достатъчен брой точки едновременно в система WGS84 и в система 1970 г. Същите да се дават от службите по кадастър на фирмите, работещи в областта на геодезията. По този начин вече няма да има разлики в координатите на точките между различните фирми, ако използват за трансформация от система WGS84 в система 1970 г. различни опорни точки.

Схема на мрежата

Използвана литература

1. **Михайлов Пл.** – " Получаване на нормалните височини на точки, резултат от *GPS* измервания", НВУ "Васил Левски", Факултет "А, ПВО и КИС" - Шумен, Научна сесия 2006.

2. **Михайлов Пл.** – "Върху трансформациите на резултатите от GPS измервания", НВУ "Васил Левски", Факултет "А, ПВО и КИС" - Шумен, Научна сесия 2005.

3. **Михайлов Пл**. Съвместно изравнение на геодезически и *GPS* измервания за нуждите на държавната геодезическа мрежа, Автореферат на дисертация, 2005 г.

4. **Андреев А, Михайлов Пл**. Използване на нормални височини при изравнение с *GPS*, НВУ "Васил Левски" - В.Търново, Военно-научен форум 2004 с международно участие.

5. **Вълев** Г. Интегрална геодезия и оптимизация на геодезически мрежи – лекционен курс, записки, София - Шумен, 2003 г.

6. **Вълев** Г. **Минчев** М. Комбиниране на резултатите от класически и *GPS* измервания за изграждане на съвременни геодезически мрежи. Геодезия, картография, земеустройство, бр. 5-6, 1995 г.

7. B. **Hofmann**-Wellenhof, H. Lichtenegger and J. Collins, Global Positioning System, Theory and Practice, Springer-Verlag, Wien New York

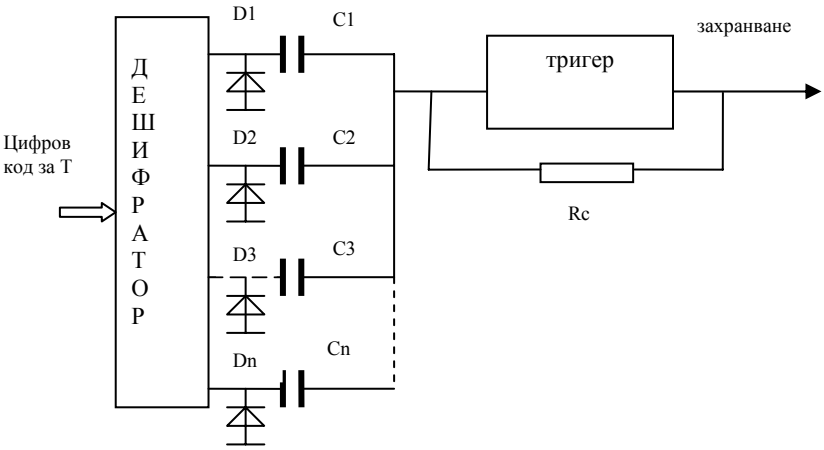
УПРАВЛЯЕМ ГЕНЕРАТОР НА ПРАВОЪГЪЛНИ ИМПУЛСИ

Атанас Атанасов

Abstract: This paper refers to square – wave generator with digital control concerns

В много електронни устройства често се налага да се използват генератори на правоъгълни импулси с променящи се в процеса на работа параметри. Известен е мултивибратор, във времезадаващата верига на който е включен набор от кондензатори, чрез които се генерират импулси с различна честота на повторение. Недостатък на този генератор е че не може да се извършва автоматично високоскоростно изменение на честотата на повторение на генерираните импулси. Този недостатък може да бъде отстранен чрез превключване на необходимия за даден момент времезадаващ кондензатор в съответствие с постъпващ цифров код. Структурната схема на такъв генератор е показана по долу.

Схемата включва тригер на Шмит, на който времезадаващата верига е свързана с n на брой кондензатори, като всеки следващ е с два пъти по-голяма стойност на капацитета от предишния. Всеки кондензатор е свързан инверсните изходи на дешифратор и с разряден диод.



На входовете на дешифратора постъпва в цифров вид кодът за периода на повторение на импулсите, който се дешифрира при което на един от изходите му се появява сигнал „0”, в резултат на което един от времезадаващите кондензатори се включва към времезадаващата верига на тригера и при постъпване на стробиращ импулс, започва неговият заряд по веригата: заряден резистор, кондензатор, дешифратор. Когато напрежението достигне нивото на сработване на тригера, той преминава в неустойчиво състояние, при което започва разряд на кондензатора през зарядно-разрядния резистор, тригера и разрядния диод. Когато напрежението достигне нивото на сработване на тригера, той се връща в изходно състояние. При това се формира последователност от импулси с честота на повторение, определяща се от капацитета на съответния кондензатор. Еквивалентният капацитет между входа на тригера и маса се определя от израза:

$$C = \left(X_{n-1}.C_{n-1}\right) + \left(X_{n-2}.C_{n-2}\right) + \ldots X_0.C_0$$

където:

$X_{n-1} \ldots X_0$-стойности на съответните разряди в кода за периода на повторение.

Периодът на повторение на генерираните импулси ще бъде равен на:

$$T_{n.изх.} = t_{r.m.} + T_n$$

където:

$t_{r.m.}$-продължителност на импулса на генератора.

От своя страна продължителността на импулса се определя от израза:

$$t_{r.m.} = 5C_{e.\max.}.R^0_{изх.}$$

Където:

$$C_{e.\max.} = C_0 + C_1 + C_2 \ldots C_{n-1};$$

$R^0_{изх.}$-изходно съпротивление на логическия елемент в изходно състояние.

Тъй като $R^0_{изх.}$ практически има много малка стойност, можем с достатъчно голяма точност да смятаме ,че $t_{r.m.} << T_n$ и $T_{n.изх.} \approx T_n$. За времето на T_n напрежението на кондензатора се изменя от нула до праговата стойност на напрежението U_1. Затова периодът на повторение се определя от израза:

$$T_{n.изх.} = R.C_e \ln \frac{E_{c.e.}}{E_{c.e.} + U_1}$$

където: R е съпротивлението, определящо изходния ток на тригера на Шмит при логическа нула на входа му.

Така разгледаният генератор на правоъгълни импулси осигурява изменение на периода, респективно честотата на повторение на генерираните импулси в съответствие с постъпващ цифров код.

ГОДИШНИК

НА ШУМЕНСКИЯ УНИВЕРСИТЕТ „ЕПИСКОП КОНСТАНТИН ПРЕСЛАВСКИ”

ТЕХНИЧЕСКИ НАУКИ

Т. I Е

Формат 16/60/84

ISSN 1311-834X
Университетско издателство
Епископ Константин Преславски