памяти, а далее, при формировании движений звеньев Робота используются при запуске таймеров.

```
PROGRAM COMMENTS
Network 1      Network Title
Инициализация переменных
LD      #Start:L0.0
S       M0.1, 1
MOVB    2#0, MB1
MOVW    2000, VW100
MOVW    180, VW102
MOVW    200, VW104
MOVW    220, VW106
MOVW    220, VW108

Network 2
Поворот руки по часовой стрелке
LD      M0.1
TON     T32, VW100

Network 3

LD      T32
S       M0.2, 1

Network 4
Выдвижение руки
LD      M0.2
TON     T33, VW102

Network 5

LD      T33
S       M0.3, 1

Network 6
поворот кисти
LD      M0.3
TON     T34, VW104
```

Рис.2. Фрагмент управляющей программы

Заключение

Рассмотрены особенности управления промышленным роботом ЦПР-1П. Предложены способы надежного управления роботом при выполнении загрузочно-разгрузочных работ. Первый способ можно использовать при большой номенклатуре загружаемых деталей. Второй способ – при большой серии однотипных деталей. Для проверки предложенных способов была написана управляющая программа в пакете *MicroWin_Step*7.

Библиографический список

1. Нестандартное оборудование. Пневмоманипулятор Балсити [Электронный ресурс]. - URL: https://www.youtube.com/watch?v=RlIxC3qVR8U (дата обращения 28.03.19)
2. Пневматические манипуляторы [Электронный ресурс]. - URL: http://m.fam-robotics.ru/ru/pnevmaticheskie-manipulyatory (дата обращения 28.03.19)
3. Siemens [Электронный ресурс]. - URL: https://www.siemens.com/ru/ru/home.html (дата обращения 28. 03.19)
4. SIMATIC [Электронный ресурс]. - URL: https://w3.siemens.com/mcms/topics/en/simatic/Pages/Default.aspx (дата обращения 28.03.19)
5. CPU-224 [Электронный ресурс]. - URL: https://www.siemens-ru.com/taxonomy/term/12 (дата обращения 28.03.19)
6. MicroWin Step7 [Электронный ресурс]. - URL: https://support.industry.siemens.com/cs/document/14191321/new-step-7-version-v5-2-now-available-?dti=0&lc=en-WW (дата обращения 28.03.19)

УДК 004.056.55

А.А. Гуринов

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ МЕТОДА ШИФРОВАНИЯ RSA

Филиал «Протвино» государственного университета «Дубна»
Секция «Информационные технологии»

Научный руководитель – Губаева Милета Михайловна, старший преподаватель кафедры информационных технологий филиала «Протвино» государственного университета «Дубна».

Создана программа, реализующая криптографический метод с открытым ключом RSA. Программа разработана на языке программирования C в среде Microsoft Visual Studio 2017.

Ключевые слова: асимметричное шифрование, криптосистема RSA, криптография с открытым ключом.

Об авторе

Гуринов Александр Анатольевич – студент 4 курса направления «Информатика и вычислительная техника» филиала «Протвино» государственного университета «Дубна»

A.A. Gurinov

SOFTWARE IMPLEMENTATION OF RSA ENCRYPTION METHOD

Scientific adviser – Gubaeva Mileta Mikhailovna, senior lecturer of the department information technology of the branch "Protvino" state University "Dubna".

Created a program that implements the RSA cipher. The program is developed in the C programming language in the environment Microsoft Visual Studio 2017.

Keywords: cryptography, RSA, public-key cryptography, asymmetric cryptography.

About the author

Gurinov Alexandr Anatolyevich – 4th year student of the direction "Informatics and computer engineering" of the branch "Protvino" state University "Dubna".

Необходимость безопасной передачи важных сообщений возникла еще в древности. Первые письменные упоминания об использовании относительно примитивных криптографических средств относят примерно к 3 тысячелетию до н. э. В основном в то время шифры применялись в военном деле, где несвоевременная или скомпрометированная информация могла стоить победы, не говоря уже о многочисленных людских потерях.

Так, например, шифр Цезаря – один из самых известных и простых методов использовался самим Цезарем для военной переписки со своими генералами. Для своего времени, несмотря на свою простоту, данный метод обеспечивал необходимую степень надежности.

Вплоть до середины 20 века криптография в основном ограничивалась моноалфавитными и полиалфавитными методами шифрования, которые являются симметричными способами кодирования. В симметричных способах для декодирования и кодирования сообщений используется один и тот же ключ, что влечет за собой ряд существенных недостатков.

Одним из главных и самым ограничивающим недостатком симметричных методов является необходимость непосредственной передачи ключа «из рук в руки», что делает практически невозможным участие неограниченного числа лиц в шифровании.

И только лишь после работ Шеннона [1], в которых впервые появляются строгие математические определения количества информации, функций шифрования, можно говорить о зарождении нового направления в криптографии - систем с открытым ключом, основывающихся на асимметричных методах.

Криптографическая система с открытым ключом – система шифрования и/или электронной подписи, при котором при которой открытый ключ передается по открытому каналу и используется для проверки электронной подписи и для шифрования сообщений.

Эти системы базируется на трудности нахождения обратной функции $f^{-1}(f(x))$ при неизвестном *x*, где f(x) легко вычисляется для любого аргумента. Несуществование легкого и быстрого метода нахождения такой обратной функции обеспечивает надежность шифрования. Они представляют собой разновидность асимметричного шифрования.

Асимметричное шифрование позволяет участие неограниченного числа участников шифрования за счет использования 2 ключей, что было проблемой в симметричных методах.

Основными принципами асимметричного шифрования с открытым ключом являются:

- Открытые принципы реализации алгоритмов генерации ключей и выполняющих операции шифрования, зависящих от них.
- Создание пары таких двух больших простых чисел, что знание открытого ключа никак бы не помогло в воссоздании образа закрытого.
- Открытый ключ является общеизвестным или передается ограниченному кругу лиц, с которыми планируется вести диалог.
- Доступ к закрытому ключу имеется только у владельца, который прилагает все необходимые усилия для его защиты.
- Метод шифрования является надежным, причем сообщение, закодированное открытым ключом, можно расшифровать только с помощью его парного закрытого ключа.

Первой системой с открытым ключом, пригодной и для шифрования, и для цифровой подписи, стала криптосистема RSA. Статья 1976 года Уитфилда Диффи и Мартина Хеллмана «Новые направления в криптографии» (*New Directions in Cryptography*) [2] стала отправным моментом для создания криптосистемы RSA.

Год спустя Рональд Ривестом, Ади Шамиром и Леонард Адлеманом из MIT была предложена модель криптографической системы с открытым ключом RSA, получившая свое название по первым буквам их фамилий, которая была впервые описана в журнале Scientific American. В основе данного шифра лежит теорема Эйлера $m^{\varphi(n)} \equiv 1\ (mod\ n)$.

В оригинальной статье года приводится следующий алгоритм генерации RSA-ключей:

- Выбираются два различных случайных простых числа *p* и *q* заданного размера.
- Вычисляется их произведение $n = p * q$, которое называется модулем.
- Вычисляется значение функции Эйлера (1) от числа n:
$$\varphi(n) = \varphi(p * q) = \varphi(p) * \varphi(q) = (p-1) * (q-1) \quad (1)$$
- Выбирается целое число *e* ($1 < e < \varphi(n)$) взаимно простое со значением функции $\varphi(n)$.
- Вычисляется число, *d* мультипликативно обратное к числу *e* по модулю $\varphi(n)$, то есть число, удовлетворяющее сравнению (2):
$$d * e \equiv 1\ (mod\ \varphi(n)) \quad (2)$$
- Пара {*e, n*} публикуется в качестве открытого ключа *RSA*.
- Пара {*d, n*} играет роль закрытого ключа *RSA* и держится в секрете.

За время использования данный алгоритм претерпел ряд усовершенствований:

- Использование функции Кармайкла $\lambda(n)$ вместо функции Эйлера $\varphi(n)$, что в большинстве случаев позволяет уменьшить размер *d*, а это в свою очередь означает меньшие временные затраты на расшифровку.
- Использование китайской теоремы об остатках для более быстрой расшифровки сообщений.

При использовании китайской теоремы функция дешифрации сообщения (3)
$$m = D(c) = c^d\ (mod\ n) = c^d\ (mod\ p * q) \quad (3)$$
сводится к следующей системе уравнений (4)
$$\begin{cases} m1 = c^d (mod\ p) \\ m2 = c^d (mod\ q) \end{cases}, \quad (4)$$
которая далее упрощается при помощи малой теоремы Ферма (5)
$$a^{p-1}\ (mod\ p) \equiv 1 \quad (5)$$
Тогда система принимает вид (6)
$$\begin{cases} m1 = c^d (mod\ p) = c^{k_1 * (p-1) + d_p} (mod\ p) = c^{d\ (mod\ p-1)}\ (mod\ p) \\ m2 = c^d (mod\ q) = c^{k_2 * (p-1) + d_Q} (mod\ q) = c^{d\ (mod\ q-1)}\ (mod\ q) \end{cases} \quad (6)$$

После этого преобразования уже можно применять китайскую теорему об остатках. В итоге дешифрование сводится к выполнению следующих операций:

- $d_p = d\ (mod\ p-1)$
- $d_Q = d\ (mod\ q-1)$
- $q_{inv} = q^{-1}\ (mod\ p)$
- $m_1 = c^{dP}\ (mod\ p)$
- $m_2 = c^{dQ}\ (mod\ q)$
- $h = q_{inv}(m_1 - m_2)\ (mod\ p)$
- $m = m_2 + hq$

В итоге из-за понижения разряда степени в 2 раза, 2 новые операции возведения в степень по модулю займут примерно в 4 раза меньше времени, нежели одна, но с исходной разрядностью.

За все 40 лет использования шифра RSA, он и по сей день не утратил свою высокую степень надежности а, следовательно, актуальность. И она сохранится до тех пор, пока не будет решена задача о быстрой факторизации больших чисел.

Программа, осуществляющая шифрование способом RSA, разработана на языке программирования C в среде Microsoft Visual Studio 2017. Выбор языка C обоснован высокой производительностью программного кода, что является важным при реализации таких затратных функций, как возведение в степень по модулю очень больших чисел, где большинство других высокоуровневых языков программирования не могут обеспечить подобное быстродействие.

Разработанная программа позволяет создание ключей, необходимых для шифрования, и непосредственно реализует сами функции шифрования и дешифрования.

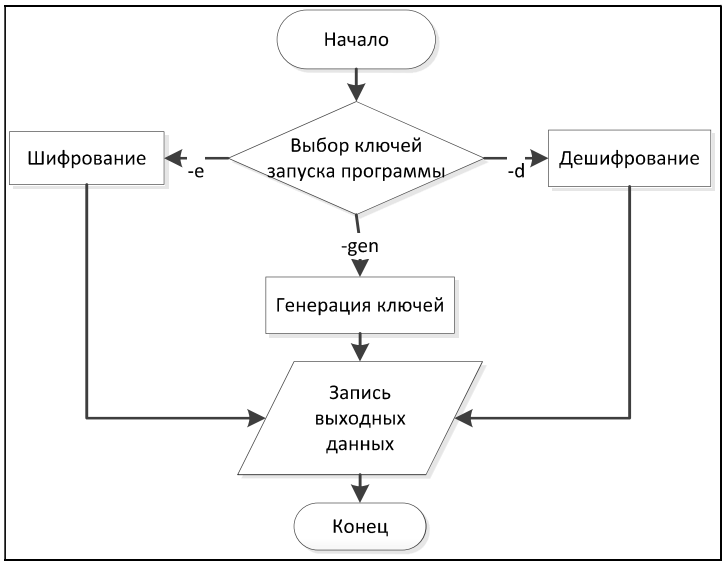

Рисунок 1 – общая схема программы

На рисунке 1 представлена общая схема работы программы. Она являет собой консольное приложение, для работы с которым используются ключи запуска.

Программа имеет следующие ключи запуска:

- -gen – генерация закрытого и открытого ключей.
- -e [file].ext [pub_key].bin – шифрование файла [file].ext с помощью открытого ключа [pub_key].bin, на выходе получим зашифрованный файл OUT_ENCRYPTED.bin
- -d [file].bin [priv_key].bin – расшифровка файла [file].bin с помощью закрытого ключа [priv_key].bin, на выходе получим зашифрованный файл OUT_DECRYPTED.txt

На рисунке 2 представлен алгоритм генерации ключей

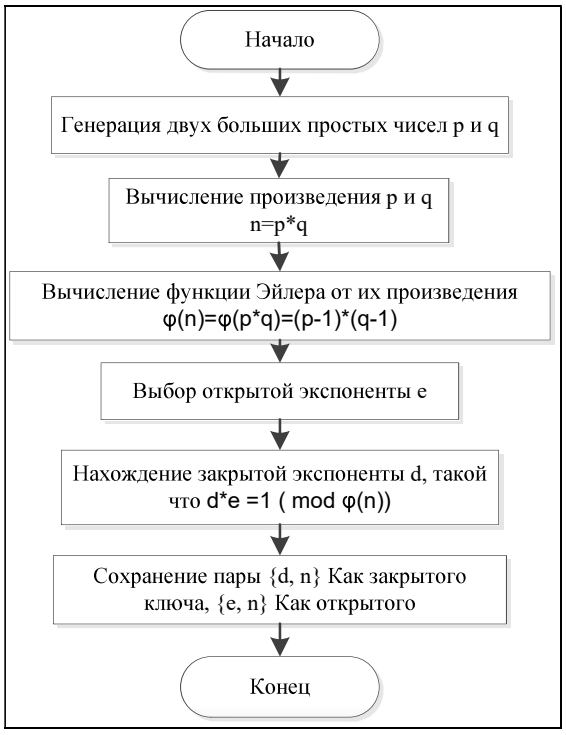

Рисунок 2 – генерация ключей

Одной из главных трудностей в этом является генерация больших простых чисел. Для их нахождения в основном используются вероятностные тесты, так как аналитической функции для их нахождения нет. Хотя эти тесты не всегда могут гарантировать, что при их выполнении, число будет простым, но при достаточном количестве испытаний и применении сразу нескольких тестов вероятность ошибочного нахождения составного числа сводится к нулю.

Тест простоты Ферма является одним из таких вероятностных тестов, который дает достаточно надежные результаты. Он имеет следующий вид:

Пусть n > 1 — натуральное число. Тогда для любых a будет выполняться соотношение (7):

$$a^{n-1}(mod\ n) \equiv 1 \quad (7)$$

Если таких чисел a не существует или не выполняется сравнение, то n — составное число. Для большей уверенности в том, что n действительно является простым, тест прогоняют несколько раз с разными значениями числа a.

Наряду с ним могут применяться и другие тесты для обеспечения большей уверенности в простоте числа, например, тест Миллера-Рабина, тест Люка, тест Соловея-Штрассена и другие.

Хотя и существует достаточное количество истинных тестов простоты, их надежность никак не может компенсировать время, затрачиваемое на нахождение простого числа в некотором диапазоне.

На рисунке 3 представлен алгоритм шифрования и дешифрования

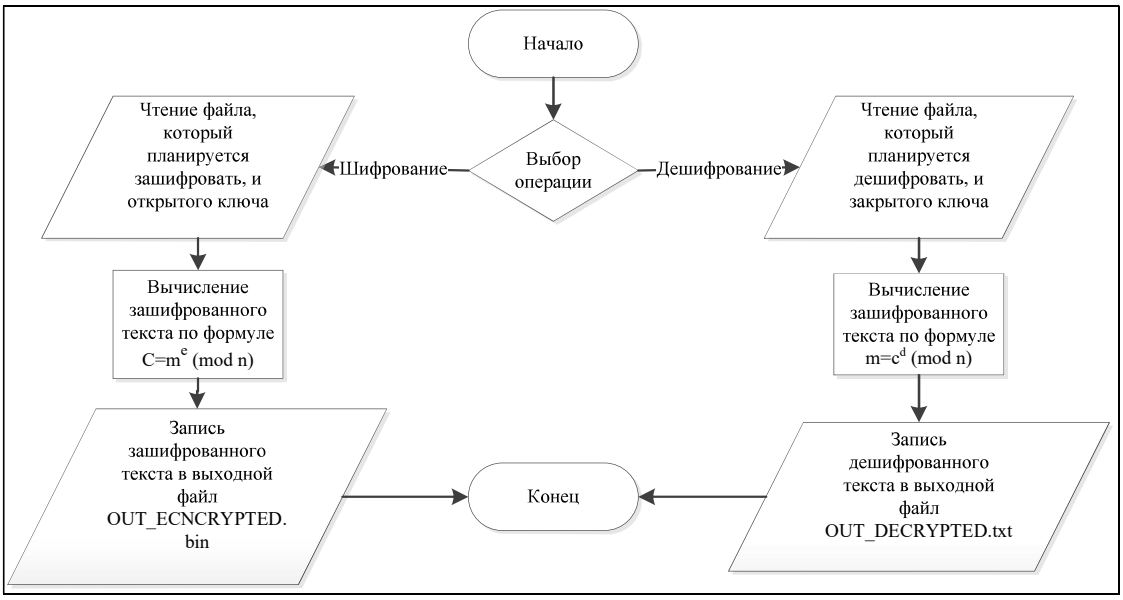

Рисунок 3 - алгоритм шифрования и дешифрования

Была разработана программа, реализующая шифрование с помощью метода RSA, написана библиотека для работы с очень большими числами, где реализованы функции для работы с большими числами и описана структура для их хранения. Были изучены основы защиты информации и теоретические обоснования стойкости криптосистемы RSA.

Библиографический список

1. C. E. Shannon, «A Mathematical Theory of Communication», Bell System Technical Journal, vol. 27, pp. 379-423, 623-656, July, October, 1948
2. IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. IT-22, NO. 6, pp. 644-654, NOVEMBER 1976
3. Арнольд И.В. Теория чисел: Учебное пособие/И.В. Арнольд. – М.:ЛЕНАНД, 2017 – 288 с.
4. Нильс Фергюсон, Брюс Шнайер. Практическая криптография = Practical Cryptography: Designing and Implementing Secure Cryptographic Systems. - М. : Диалектика, 2004. - 432 с

УДК 62+3

В. С. Исхаков, Н.А. Шумакова, Д.А. Калинкин

ГЕНЕРАТОР ТОКА ВЫСОКОЙ ЧАСТОТЫ ДЛЯ ТЕХНОЛОГИЧЕСКИХ ОПЕРАЦИЙ

Филиал «Протвино» государственного университета «Дубна»
Секция «Естественные и инженерные науки»

Научный руководитель – Дягилев Владимир Иванович, кандидат технических наук, доцент кафедры автоматизации технологических процессов и производств филиала «Протвино» государственного университета « Дубна».

В статье рассматривается устройство генератора прямоугольного и синусоидального напряжений высокой частоты и исследуются его свойства. Этот генератор предназначен для проведения лабораторных работ по курсам «Электротехника и электроника» и «Измерительная техника и приборы».

Ключевые слова: транзистор, колебательный контур, технологическая установка.

Об авторах