

# Algoritmo de Criptografia Posicional em Circuitos Programáveis

Rodolfo Barros Chiaramonte, Edward Moreno

Faculdade de Informática de Marília – Fundação de Ensino Eurípides Soares da Rocha  
Av. Hygino Muzzi Filho 529 – CEP 17525-901 – Marília – S.P – Brasil

rodbc@terra.com.br , edmoreno@fundanet.br

**Resumo.** *A segurança da informação digital é questão de preocupação para muita gente. Com isso foram desenvolvidas várias técnicas para dificultar o roubo de dados informatizados. Uma dessas técnicas é a criptografia digital, mas o modelo de criptografia existente possui alguns problemas por ser muito conhecidos. Portanto, é necessário um novo modelo de criptografia, que fuja dos padrões atuais. No entanto, deve-se tomar cuidado com a complexidade de modo que o novo modelo não aumente muito o tamanho e o tempo de processamento das informações criptografadas. Assim, nesse projeto pretende-se desenvolver circuitos digitais reprogramáveis, usando-se de FPGAs, com vários tipos de expressões posicionais que possam processar esse modelo de criptografia de maneira simples e rápida.*

**Palavras Chave.** *Circuitos Reconfiguráveis, Criptografia.*

## 1. INTRODUÇÃO

Um dos métodos mais utilizados para manter a segurança da informação é a criptografia. A criptografia vem sendo utilizada desde a época da escrita hieroglífica dos egípcios e hoje com a era da informática sua importância cresceu muito.

Vários tipos de algoritmos de criptografia diferentes estão sendo pesquisados, os mais conhecidos são o DES<sup>[Schn96]</sup> (Data Encryption Standard) e o RSA<sup>[RSA78]</sup> (deriva do nome de seus inventores, os professores do MIT Ronald Rivest, Adi Shamir e o professor do USC Leonard Adleman), esses algoritmos possuem respectivamente chave simétrica e assimétrica.

Os algoritmos de **chave simétrica** possuem apenas uma chave secreta para cifrar ou decifrar a mensagem, isso pode gerar dificuldades para o envio da chave secreta já que é necessário um meio seguro pra transporta-la.

Os algoritmos de **chave assimétrica** possuem duas chaves, uma para cifrar e outra para decifrar a mensagem. Se uma mensagem for cifrada com a chave publica, somente a chave privada pode decifrar a mensagem; caso a mensagem for cifrada com a chave privada, somente a chave publica poderá decifrar a mensagem, esse caso é utilizado para a autenticação do usuário.

Os algoritmos RSA e DES são muito conhecidos, motivo pelo qual possuem uma grande desvantagem, acaba se tornando fácil encontrar programas que conseguem quebrar em pouco tempo algumas chaves pequenas, no entanto esses algoritmos são muito eficientes com chaves grandes.

Refira-se como curiosidade que se estima que atualmente demora-se 0.2 segundos para uma chave de 40 bits, 3.6 horas para uma chave de 56 bits e anos para uma chave de 128 bits. Deste modo quanto maior for a chave mais difícil será descobri-la.<sup>[Sant00]</sup>

## 2. TRABALHOS CORRELATOS

O RSA<sup>[RSA78]</sup> é um sistema de criptografia de chave assimétrica que foi inventado por volta de 1977 pelos professores do MIT Ronald Rivest, Adi Shamir e o professor do USC Leonard Adleman.

O sistema consiste em gerar uma chave pública e uma chave privada através de números primos, o que dificulta a obtenção de uma chave a partir da outra.

Os algoritmos para a geração da chave pública e privada e para cifrar e decifrar as mensagens são simples. Observe-os a seguir:

- 1) Escolhe-se dois  $n^{\text{os}}$  primos grandes (**p** e **q**);
- 2) Gera-se um número **n** através da multiplicação dos números escolhidos anteriormente (**n = p . q**);
- 3) Escolhe-se um número **e**, tal que **e** é menor que **n** e **e** é relativamente primo à (**p-1**).(**q-1**);
- 4) Escolhe-se um número **d** tal que (**ed-1**) seja divisível por (**p-1**).(**q-1**);

Os valores **e** e **d** são chamados de expoentes público e privado, respectivamente. O par (**n,e**) é a chave pública e o par (**n,d**) é a chave privada. Os valores **p** e **q** devem ser mantidos em segredo ou destruídos.

## 3. DESCRIÇÃO DO SISTEMA POSICIONAL

O algoritmo de criptografia posicional consiste em que a posição do byte interfere sobre a chave utilizada na criptografia.

Como esse tipo de criptografia a sequência AAABBB, por exemplo, poderia ser criptografada como BCDFGH, sem acrescentar a ela nenhum bit. Para isso, admita o valor decimal dos bytes **A**, **B**, **C**, **D**, **E**, **F**, **G** e **H** de acordo com a tabela ASCII (**65**, **66**, **67**, **68**, **69**, **70**, **71** e **72** respectivamente). A esses valores são acrescidos o valor de sua posição, como o primeiro **A** está na posição um, o seu valor decimal criptografado será 66 (na tabela ASCII **B**), mas como o segundo **A** está na posição dois, deve-se acrescentar 2 ao seu valor ASCII, assim o valor decimal do segundo byte será 67 (na tabela ASCII **C**).

Tabela 3.1 – Exemplo Básico:

Sequência	<b>A</b>	<b>A</b>	<b>A</b>	<b>B</b>	<b>B</b>	<b>B</b>
Valor decimal na tabela ASCII	65	65	65	66	66	66
Valor da posição	1	2	3	4	5	6
Valor decimal ASCII (código criptografado)	<b>66</b>	<b>67</b>	<b>68</b>	<b>70</b>	<b>71</b>	<b>72</b>
Código em caracteres (criptografado)	<b>B</b>	<b>C</b>	<b>D</b>	<b>F</b>	<b>G</b>	<b>H</b>

Mas esse modelo ainda é simples de ser decifrado, por isso, é necessário somar ao valor do byte em ASCII um número gerado utilizando a posição do byte e uma expressão, por exemplo:  $4 * \text{posição} - 1$ .

Com essa expressão, a tabela acima ficaria como demonstrado na tabela 3.2 a seguir:

Tabela 3.2 – Exemplo de funcionamento com a expressão:  $4 * \text{posição} - 1$ 

Seqüência	A	A	A	B	B	B
Valor decimal na tabela ASCII	65	65	65	66	66	66
Valor da posição	1	2	3	4	5	6
Resultado da expressão	3	7	11	15	19	23
Valor decimal na tabela (código criptografado)	<b>68</b>	<b>72</b>	<b>76</b>	<b>81</b>	<b>85</b>	<b>89</b>
Código em caracteres (criptografado)	<b>D</b>	<b>H</b>	<b>L</b>	<b>Q</b>	<b>U</b>	<b>Y</b>

Para entender a tabela 3.2 considere que os valores **68**, **72**, **76**, **81**, **85** e **89** representam os caracteres **D**, **H**, **L**, **Q**, **U** e **Y** respectivamente na tabela ASCII.

Nesse tipo de criptografia, quanto maior o grau da expressão posicional, maior será a segurança. Por exemplo: Se a expressão posicional for uma equação do terceiro grau, terá uma segurança maior que um sistema com equação do segundo grau.

Vejamos a seguir como ficará essa seqüência criptografada com uma expressão do segundo grau:

$$3 \text{ posição} * \text{posição} - 5 \text{ posição} + 43$$

Tabela 3.3 – Exemplo com a expressão :  $3 \text{ posição} * \text{posição} - 5 \text{ posição} + 43$ 

Seqüência	A	A	A	B	B	B
Valor decimal na tabela ASCII	65	65	65	66	66	66
Valor da posição	1	2	3	4	5	6
Resultado da expressão	41	45	55	71	93	121
Valor decimal na tabela (código criptografado)	<b>106</b>	<b>110</b>	<b>120</b>	<b>137</b>	<b>159</b>	<b>187</b>
Código em caracteres (criptografado)	<b>j</b>	<b>n</b>	<b>X</b>	<b>ë</b>		<b>+</b>

Os casos acima são casos simples e não apresentam problemas para a criptografia, mas existem casos onde ocorre o estouro de um byte. Nesse caso, o valor do byte será “valor encontrado **mod** 255” onde **mod** retorna o resto da divisão.

A seguir exemplifico uma equação de 3º grau, onde ocorre o estouro de um byte e como será feito o processo de criptografia. A equação posicional utilizada será:

$$23 \text{ posição} * \text{posição} * \text{posição} + 26 \text{ posição} * \text{posição} - 45 \text{ posição} - 63$$

Tabela 3.4 – Exemplo com a expressão:  $23 \text{ posição} * \text{posição} * \text{posição} + 26 \text{ posição} * \text{posição} - 45 \text{ posição} - 63$ 

Seqüência	A	A	A	B	B	B
Valor decimal na tabela ASCII	65	65	65	66	66	66
Valor da posição	1	2	3	4	5	6
Resultado da expressão	-59	135	657	1645	3237	5571
Valor da expressão + valor ASCII original	<b>6</b>	<b>200</b>	<b>722</b>	<b>1711</b>	<b>3303</b>	<b>5637</b>
Valor decimal na tabela (código criptografado)	<b>6</b>	<b>200</b>	<b>212</b>	<b>181</b>	<b>243</b>	<b>27</b>
Código em caracteres (criptografado)	<b>_</b>	<b>+</b>	<b>È</b>	<b>Á</b>	<b>¾</b>	

Objetivo da Proposta: Pretende-se desenvolver circuitos digitais reprogramáveis utilizando FPGAs. Esses circuitos devem ser projetados de tal forma que torne simples e rápida a manipulação de dados criptografados utilizando equações de diversos graus. Além disso pretende-se verificar o desempenho do bem conhecido RSA e implementá-lo também em FPGAs e comparar a sua performance com o nossa proposta. Portanto será realizada uma comparação de desempenho (em software e hardware) de duas propostas de encriptação: RSA e o nosso (posicional com diferentes graus). Dependendo da análise comparativa poderá se propor uma alternativa híbrida com melhores características de desempenho e confiabilidade.

#### 4. TÉCNICA DE AVALIAÇÃO DE DESEMPENHO

Implementar os algoritmos de criptografia utilizando a ferramenta XILINX e FPGAs. A utilização de FPGAs é importante para a reprogramação do circuito para os diversos níveis de segurança que o algoritmo pode trabalhar e para fácil troca de chaves quando necessário.

Será realizada uma comparação de desempenho entre os algoritmos implementados e se possível, será implementado um algoritmo híbrido, tomando o cuidado de manter o máximo de desempenho.

#### 5. CRONOGRAMA DE ATIVIDADES

Com a finalidade de alcançar e cumprir os objetivos do projeto, as seguintes atividades serão realizadas nos seguintes meses do ano 2001.

Março-Abril	Maio-Junho	Julho-Agosto	Setembro- Novembro	Dezembro-Fevereiro
Pesquisa dos algoritmos de criptografia (RSA). Já realizado	Implementação e Desempenho do algoritmo RSA utilizando a linguagem C. Já realizado	Implementação e Desempenho do algoritmo de Criptografia Posicional utilizando a linguagem C. Já realizado	Implementação dos dois tipos de algoritmos criptográficos utilizando FPGAs na ferramenta XILINX.	Estudo dos gráficos de desempenho dos algoritmos e desenvolvimento de um algoritmo híbrido. Apresentação e análise dos resultados e Escrita de um relatório final.

#### 6. REFERÊNCIAS OU BIBLIOGRAFIA

- [Rodr00] Criptografia e aplicativos: PGP, HTTPS; Wagner Dias Rodrigues; Consulta realizada em Março de 2000 no site [<http://www.del.ufrj.br/~wdias/teleinfo/indice.htm>];
- [Ribe00] Segurança de Redes de Comunicação; Marcia L. S. Agüena; Nassim Chamel Elias; Thiago Pirola Ribeiro; Consulta realizada em Março de 2000 no site [<http://www.dc.ufscar.br/~tribeiro/>];
- [Kaya94] High-Speed RSA Implementation. Cetin Kaya Koc, Technical Report of RSA Laboratories, RSA Data Security INC., Redwood City, CA., USA, Nov. 1994.
- [Kaya95] RSA Hardware Implementation. Cetin Kaya Koc, Technical Report of RSA Laboratories, RSA Data Security INC., Redwood City, CA., USA, August, 1995.

- [Stal95] Cryptography and Networks Security: Principles and Practices. William Stallings, Ed. Prentice Hall, Second Edition, 1995.
- [Sant00] SEGURANÇA DE REDES DE COMPUTADORES. Autor: Rodrigo Eduardo dos Santos. Consulta realizada em Outubro, 2000, no site ([http://www.res.hpg.com.br/l\\_segur\\_red2.htm](http://www.res.hpg.com.br/l_segur_red2.htm)).
- [Heac01] História e Aplicações da Criptografia – Verdade @bsoluta. Consulta realizada em 23 de Abril de 2001, no site [[http://www.absoluta.org/crity/crity\\_h.htm](http://www.absoluta.org/crity/crity_h.htm)].
- [Tera00] Segurança de Dados Criptografia em Redes de Computadores. Routo Terada, Ed. Edgard Blücher, 1ª Edição, 2000.
- [RSA78] A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. R. L. Rivest, A. Shamir and L. Adleman. Communications of the ACM, 21(2):120-126, February 1978.
- [DH76] New Directions in cryptography. W. Diffie and M Hellman. IEEE Trans. Inform. Theory IT-22, (Nov. 1976), 644-654.
- [Knut97] The Art of Computer Programming – Volume 2: Seminumerical Algorithms – Third Edition, Donald E. Knuth, Ed. Addison Wesley, 1997.
- [Hulm99] Security and Encryption - Understanding the RSA Cryptosystem – Alison Huml – 1999. Consulta realizada em 05/07/2001 no site [<http://www.hobnob.com/mcs225/>].
- [Mats93] Mitsuri Matsui. Linear Criptanalysis Method for DES Cipher. In advances in Criptology, Eurocrypt 93, Lectures Notes in Computer Science Vol. 765, Springer Verlag, pp. 386-397, 1993.
- [Schn96] Bruce Schneier. Applied Cryptography: Protocols, Algorithms and Source in C. 2nd Ed. John Wiley and Sons, New York, 1996.
- [Patt00] C. Patterson, “Hardware Performance DES Encryption in Virtex FPGAs Using Jbits”, in IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM 2000) (K. L. Pocek and J. M. Arnold, eds.), April 2000.
- [Kaps98] J. P. Kaps and C. Paar, “Fast DES implementation on FPGAs and its application to a universal key-search machine”, in Fifth Annual Workshop on Selected Areas in Cryptography (S. Tavares and H. Meijer, eds.), vol. LNCS 1556, (Berlin, Germany), Springer-Verlag, August 1998. Conference Location: Queen’s University, Kingston, Ontario, Canada.