

---

# Simulador do Transmissor do Sistema Is-95 Empregando a Plataforma Ptolemy

## Is-95 System Transmitter Simulator Based on Ptolemy Platform

André Matsumoto<sup>1</sup>; Daniel Yuiti Arabori<sup>2</sup>; Ailton Akira Shinoda<sup>3</sup>

---

### Resumo

Este artigo descreve o modelo e a implementação de um simulador da transmissão por meio do canal de tráfego reverso no sistema CDMA. Todos os passos de codificação e modulação, bem como alguns resultados são apresentados.

**Palavras-Chaves:** IS-95, simulação, CDMA, celular

---

### Abstract

This article describes the model and implementation of a transmitter simulator in the reverse traffic channel of CDMA system. All the steps of codification and modulation, as well as some results are shown.

**Key Words:** IS-95, simulation, CDMA, cellular

### Introdução

Ferramentas de simulação que possibilitem aos pesquisadores, projetistas ou operadores uma avaliação correta do desempenho dos sistemas celulares, sobre uma grande gama de condições, estão se tornando cada vez mais importantes (WOERNER et al., 1994; TRANTER; KOSBAR, 1994). Em sistemas de grande porte, é completamente inviável a análise analítica do desempenho, levando em consideração os aspectos dinâmicos das interações envolvidas entre as ERB's e os usuários tais como o handoff, velocidade, tempo de duração da chamada, distribuição dos usuários na área de serviço, entre outros. Assim, a simulação torna-se uma ferramenta

essencial, tanto na avaliação da performance como na otimização dos sistemas celulares.

Este artigo descreve o modelo e a implementação do sistema celular baseado no sistema CDMA/IS-95 (TELECOMMUNICATIONS INDUSTRY ASSOCIATION, 1993). O IS-95 usa espalhamento espectral de seqüência direta ocupando uma banda de 1.25 MHz, empregando código longo e Walsh.

O restante do artigo é organizado da seguinte forma. Na seção 2, é apresentado o sistema a ser modelado e simulado. Na seção 3, é descrita a ferramenta de simulação que foi utilizada para construir o simulador. Na seção 4, são apresentados alguns resultados do simulador. Finalmente, na seção 5, expõe-

---

<sup>1,2</sup> Alunos de Graduação em Engenharia Elétrica e de Iniciação Científica.

<sup>2</sup> Docente do Departamento de Engenharia Elétrica da Universidade Estadual de Londrina – e-mail: shinoda@uel.br.

se a conclusão do artigo com algumas propostas para trabalhos futuros.

## Sistema

Existem dois enlaces entre a estação base e cada estação móvel. O canal CDMA direto constitui a transmissão da estação base para a estação móvel e consiste do canal piloto, canal de sincronismo, canais de *paging* e canais de tráfego direto. Por outro lado, o canal CDMA reverso, estação móvel para a estação base, é composto pelos canais de acesso e canais de tráfego reverso. Como o desempenho do sistema celular IS-95 está ligado ao canal de tráfego reverso, a simulação é baseada nesse canal físico.

O canal de tráfego reverso é empregado para a transmissão de informação do usuário, conjuntamente com a sinalização para a estação base durante uma chamada. O canal de tráfego reverso pode usar taxas de transmissão de 9.6, 4.8, 2.4 ou 1.2 kbps dependendo da atividade da voz durante uma conversação. O dado transmitido no canal de tráfego reverso é agrupado em frames de 20 ms.

Cada canal de tráfego reverso é identificado por uma seqüência de código longo de usuário distinta que identifica unicamente uma estação móvel nos canais de tráfego reverso e direto. A estrutura completa do canal de tráfego reverso é mostrada na Figura 1 (TELECOMMUNICATIONS INDUSTRY ASSOCIATION, 1993; RHEE, 1998).

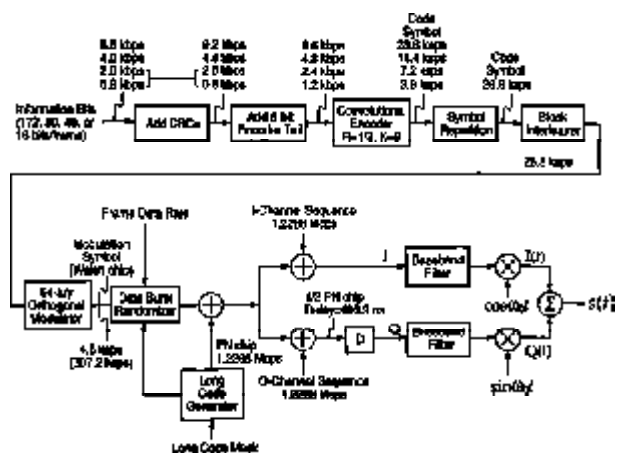


Figura 1. Estrutura do Canal de Tráfego Reverso.

O CRC é adicionado a cada 20 ms, desde que a taxa de transmissão seja 9600 bps ou 4800 bps. A função do CRC é verificar se houve erro na transmissão do frame, geralmente esta função é utilizada pela camada de rede. A seguir são acrescentados alguns bits ao final do frame, enconder tail bit, que representam uma seqüência fixa de bits adicionados para zerar o codificador convolucional. Após o enconder tail bit, o fluxo de dados é convolucionalmente codificado para correção de erros randômicos, entrelaçado em bloco (*block interleaved*) para proteção contra erros *bursts*, modulado pelos códigos Walsh 64-ário e espalhado pela seqüência direta antes da transmissão. Na simulação, optou-se pela taxa máxima de transmissão (9600 bps).

## Ptolemy

A plataforma de simulação é baseada no Ptolemy (UNIVERSITY OF CALIFORNIA AT BERKELEY, 1997), especificamente no domínio do *Synchronous DataFlow (SDF)*. Ptolemy oferece um ambiente interativo de simulação baseado em diagrama de blocos, onde cada bloco executa uma função pertinente ao processamento do sinal. É possível expandir a biblioteca original adicionando novos modelos desenvolvidos a partir daqueles existentes na biblioteca.

Ptolemy é capaz de:

- Construir modelos de simulação empregando a abordagem de hierarquização de blocos.
- Configurar e executar as simulações no domínio do tempo e frequência
- Visualizar e analisar os resultados da simulação utilizando gráficos no domínio da frequência e tempo, além de histogramas.
- Documentar os modelos de simulação e os resultados da análise/projeto

### Simulador

O simulador, implementado no Ptolemy, consiste de funções para geração de bits aleatórios, adição de CRC, encoder tail bit, código convolucional, interleaver, código de Walsh, código longo e modulação QPSK (Figura 2).

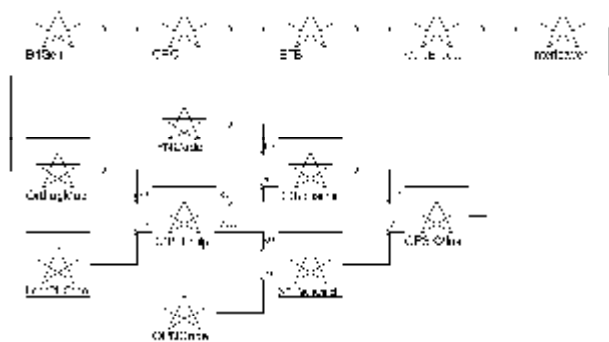


Figura 2. Diagrama de blocos do simulador

Os canais de tráfego CDMA utilizam um vocoder cuja taxa de transmissão varia de acordo com a atividade da voz do usuário. O gerador de bits randômicos simula atividade vocal máxima, com taxa de transmissão de 9600bps, cujo sinal corresponde a 172 bits por frame.

A Figura 3 e a Tabela 1 apresentam a saída do bloco gerador de bits randômicos, BitGen, referente a um frame.

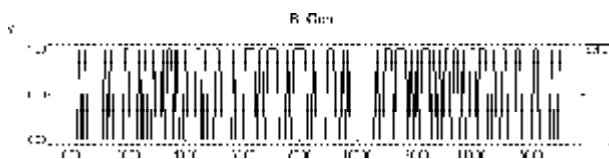


Figura 3. Saída do bloco BitGen

Tabela 1. Saída do bloco BitGen

```
0001010000001010000110001010010010110100100
0110100011000010001111011011110001011111001
0000111001010000000001001110111101010111010
1011011010011011100100110001100001100001001
```

Cada frame inclui um indicador de qualidade calculado sobre os bits de informação dentro do frame. O polinômio gerador para calcular o CRC é dado por:

$$g(x) = 1 + x + x^4 + x^8 + x^9 + x^{10} + x^{11} + x^{12} \tag{1}$$

O diagrama lógico é mostrado na Figura 4.

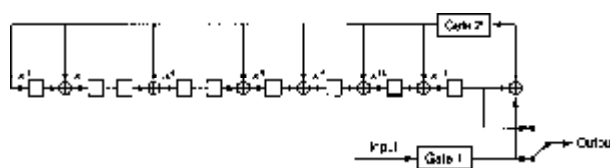


Figura 4. Circuito de cálculo do CRC

Todos os estágios do registrador de deslocamento são inicialmente configurados no nível lógico 1, a chave desconectada e as portas 1 e 2 conectadas. O registrador é deslocado 172 vezes. Após a leitura de todos os bits de informação, a porta 1 é desconectada, a chave é comutada para cima e o registrador é deslocado mais 12 vezes. Esses 12 bits adicionais são os bits de CRC.

A Figura 5 e a Tabela 2 mostra o resultado da inserção dos bits de CRC calculados no final do frame.



Figura 5. Saída do bloco CRC

Tabela 2. Saída dos 12 bits do CRC

110001100110

A seguir são acrescentados mais 8 bits no final do frame, *encoder tail bits* (ETB), configurados na lógica zero (0). A Figura 6 mostra o procedimento descrito.

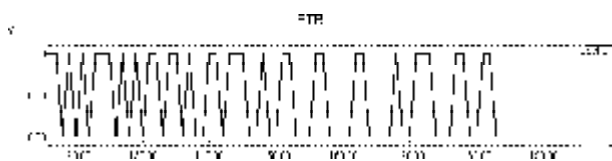


Figura 6. Saída do bloco ETB

Após o ETB, a seqüência de dados de informação é dividida em pequenos blocos, de comprimento  $k$ , e é codificado em símbolos de comprimento  $n$ . Um código convolucional  $(n, k, m)$  é implementado com  $k$  entradas,  $n$  saídas de circuito linear seqüencial com memória de ordem  $m$ . O canal reverso CDMA utiliza o código convolucional  $(3, 1, 8)$  cujas seqüências geradoras são  $g_1^{(1)} = 557_{(octal)} = 101101111_{(binário)}$ ,  $g_1^{(2)} = 663_{(octal)} = 110110011_{(binário)}$  e  $g_1^{(3)} = 711_{(octal)} = 111001001_{(binário)}$ . Esse esquema consiste de um simples terminal de entrada, um registrador de deslocamento de 8 estágios acoplado a três somadores módulo 2 e uma chave comutadora para tornar serial a saída do codificador, como ilustra a figura 7.

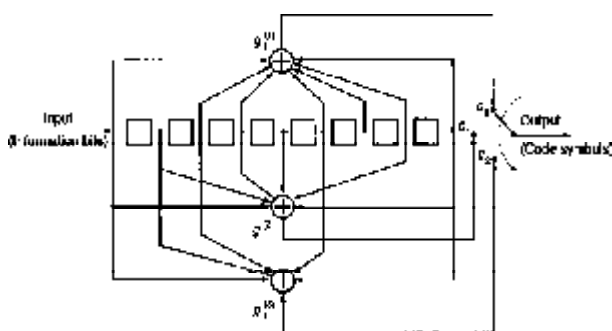


Figura 7. Codificador Convolucional (3,1,8)

Três símbolos são gerados para cada bit de informação de entrada no codificador. O estado inicial do codificador convolucional é admitido como sendo todos zero. A Figura 8 apresenta parcialmente a saída do bloco codificador convolucional, ConvEncoder, e a Tabela 3 os códigos completos gerados.



Figura 8. Saída do bloco ConvEncoder

Tabela 3. Saída do bloco ConvEncoder

Saída do ConvEncoder	
1	00000000011101101010111101111001
2	10110011000101011110111101001110
3	00100100000010000100000100101110
4	01011100011001011001110011101100
5	11010111100100100000101111110001
6	01101011111110001100001101001101
7	00110101100101110010111100100010
8	10100100111110111000100010011111
9	00001101011011101000111111001110
10	0001010001101001100110111011101100
11	01110111010011101001000000100100
12	0011101000111110110111000001101
13	10011001010111011111101111011010
14	10111111000101111100010010111111
15	11010100111100111010111101100010
16	01111111011011110010011111000110
17	011011010010011111101000110110101
18	00011111001101000001010001111000

Os símbolos da saída do codificador convolucional devem ser repetidos antes de serem entrelaçados quando a taxa de transmissão é menor que 9600 bps. Assim, não há repetição de símbolos para a taxa de dados de 9600 bps.

Entrelaçamento (*interleaving*) é o processo de permuta de uma seqüência de símbolos utilizado para alcançar a diversidade de tempo. O bloco *interleaver* forma uma série de 576 células, com 32 linhas e 18 colunas que entrelaça todos os códigos símbolos no canal de tráfego reverso – entrada por coluna e saída por linha – antes da modulação ortogonal. O *interleaving* é uma técnica eficiente para corrigir *multiple burst* ou *long burst*, característicos de ambientes multipercursos. A Figura 9 ilustra parcialmente a saída do bloco Interleaver. A Tabela 4 mostra todos os símbolos de saída do Interleaver.



Figura 9. Saída do bloco Interleaver

**Tabela 4.** Saída do bloco Interleaver

Saída do Interleaver		Saída do Interleaver	
1	010010010000111000	17	11010101111111010
2	000111000010001110	18	011001000001110010
3	011001110011010110	19	110000100000101100
4	010110100111111101	20	000100000111100011
5	000101001001110111	21	110110111101101000
6	001110111110011111	22	110100101001011101
7	010011000011010101	23	110011101100101100
8	010011101010110111	24	111011101100101110
9	000011110000001000	25	000110011100110110
10	100101011110101100	26	110111001100101101
11	100101011101001111	27	101110100010111011
12	110011110001111001	28	100010010101010011
13	001001011111100100	29	111101011101110001
14	110100101011110111	30	011101011111010110
15	100010111011011110	31	011000111000111100
16	010100110100111110	32	100011010001010010

A cada seis bits é transmitido um dos 64 símbolos de modulação possíveis gerados pela função de Walsh. A matriz de Hadamard, com  $n=2^6$ , é a função ortogonal de Walsh  $64 \times 64$ . Esta matriz pode ser gerada pela função recursiva:

$$H_{2n} = \begin{bmatrix} H_n & H_n \\ H_n & -H_n \end{bmatrix} \quad (2)$$

Uma vez que a taxa de símbolo de código na saída do bloco interleaver é 28.8 kcps, a taxa de símbolo na saída do modulador ortogonal torna-se  $28.8 / 6 = 4.8$  kcps ou  $4.8 \times 64 = 307.2$  kcps. Símbolos de modulação são selecionados de acordo com o índice de modulação de símbolo (MSI):

$$MSI = c_0 + 2c_1 + 4c_2 + 8c_3 + 16c_4 + 32c_5 \quad (3)$$

onde  $c_i, i = 0...5$ , representa cada grupo de seis símbolos de código da saída do *bloco interleaver*. O período de tempo necessário para transmitir um símbolo de modulação é igual a  $1/4800 = 208.333$  ms. O período de tempo associado com  $1/64$  do símbolo de modulação é referente a um chip Walsh e será igual a  $1/(4800 \times 64) = 1/307200 = 3.255$  ms. A Figura 10 apresenta parcialmente a saída do bloco modulador ortogonal, OrthoMod.

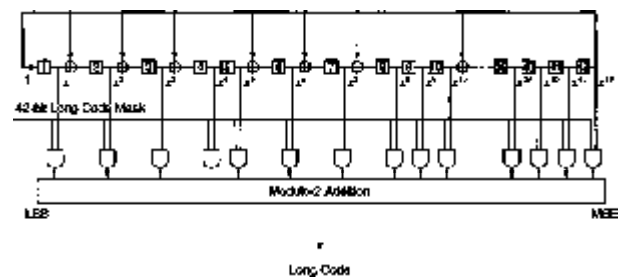


**Figura 10.** Saída do bloco OrthoMod

A próxima função a ser efetuada é o espalhamento de seqüência direta. Esta operação de espalhamento envolve um somador módulo-2 entre a saída do modulador 64-ário e o código longo. O código longo possui período  $2^{42}-1$  e é gerado pela recursão linear definida pelo seguinte polinômio primitivo característico:

$$p(x) = 1 + x + x^2 + x^3 + x^5 + x^6 + x^7 + x^{10} + x^{16} + x^{17} + x^{18} + x^{19} + x^{21} + x^{22} + x^{25} + x^{26} + x^{27} + x^{31} + x^{33} + x^{35} + x^{42} \quad (4)$$

Cada chip do código longo é denominado PN chip e é gerado a partir da soma módulo-2 interna dos resultados do produto entre uma máscara de 42 bits e o registrador de deslocamento à esquerda de 42 estágios. A máscara de código longo, *long code mask*, utilizada na seqüência contém informações sobre a identificação da unidade móvel envolvida na conexão específica. O diagrama lógico do gerador de código longo é mostrado na Figura 11.



**Figura 11.** Gerador do código longo

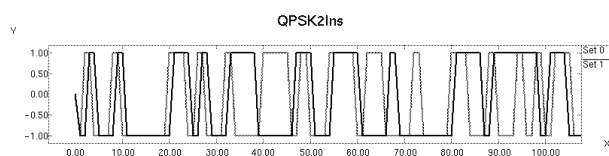
Após o espalhamento pelo código longo, o sinal é multiplicado pelas seqüências piloto I e Q em quadratura que são geradas pelas fórmulas de recursão a partir dos polinômios característicos

$$P_I(x) = 1 + x^5 + x^7 + x^8 + x^9 + x^{13} + x^{15} \quad (5)$$

$$P_Q(x) = 1 + x^3 + x^4 + x^5 + x^6 + x^{10} + x^{11} + x^{12} + x^{15} \quad (6)$$

As seqüências de máximo comprimento geradas têm comprimento  $2^5 - 1$  e são modificadas inserindo-se um “zero” após 14 “zeros” consecutivos, resultando em seqüências balanceadas de período  $2^{15}$ .

Finalmente o frame passa pelo bloco modulador OQPSK (QPSK2Ins) fornecendo os sinais I(t) e Q(t) obtidos através da filtragem em banda base. Os sinais I(t) e Q(t) modulam, respectivamente, as funções cosseno e seno, com amplitude +1 ou -1, os quais produzem ondas BPSK ortogonais cuja soma resulta na forma de onda QPSK. A Figura 12 e a Tabela 5 mostram os sinais complexos que representam a forma de onda QPSK na saída do bloco modulador.



**Figura 12.** Saída do bloco QPSK2Ins Tabela 1. Saída do bloco BitGen

**Tabela 5.** Saída do bloco QPSK2Ins

( 0, 0)	(-1,-1)	( 1, -1)	( 1, 1)	(-1, 1)	(-1,-1)
(-1,-1)	(-1,-1)	( 1, -1)	( 1, 1)	(-1, 1)	(-1,-1)
(-1,-1)	(-1,-1)	(-1,-1)	(-1,-1)	(-1,-1)	(-1,-1)
(-1,-1)	(-1,-1)	( 1, -1)	( 1, 1)	( 1, 1)	( 1, 1)
(-1, 1)	(-1,-1)	( 1, -1)	( 1, 1)	(-1, 1)	(-1,-1)
(-1,-1)	(-1,-1)	( 1, -1)	( 1, 1)	(-1, 1)	(-1, 1)
(-1, 1)	(-1, 1)	(-1, 1)	(-1,-1)	( 1,-1)	( 1,-1)
( 1,-1)	( 1,-1)	( 1,-1)	( 1,-1)	(-1,-1)	(-1, 1)
( 1, 1)	( 1, 1)	(-1, 1)	(-1,-1)	(-1,-1)	(-1,-1)
( 1,-1)	( 1, 1)	( 1, 1)	( 1, 1)	( 1, 1)	( 1, 1)
(-1, 1)	(-1,-1)	( 1, -1)	( 1, -1)	...	

## Conclusão

Este trabalho descreveu o modelamento e a implementação de um simulador da transmissão através do canal de tráfego reverso no sistema CDMA. Munido de todos os passos de codificação e modulação, bem como dos resultados obtidos torna-se desejável a implementação de um simulador da recepção do canal de tráfego reverso CDMA compatível com este, de modo que se possa verificar a eficiência do sistema. Assim sendo, este trabalho torna-se um pré-requisito indispensável para futuros trabalhos.

## Referências

UNIVERSITY OF CALIFORNIA AT BERKELEY. *Almagest: Ptolemy User's Manual*. Berkeley, 1997.

WOERNER, B. D. et al. Simulation Issues for Future Wireless Modems. *IEEE Comm. Mag.*, v.32, n.7, p.42-53, jul., 1994.

RHEE, M. Y. *CDMA Cellular Mobile Communications and Network Security*. Upper Saddle River: Prentice Hall PTR, 1998.

TELECOMMUNICATIONS INDUSTRY ASSOCIATION. *TIA/EIA Interim Standard, Cellular System Dual-Mode Mobile Station: Base Station Compatibility Standard for Dual-Mode Wideband Spread Spectrum Cellular System*, TIA/EIA/IS-95. Washington, 1993.

TRANter, W. H., KOSBAR, K.L. Simulation of Communication Systems. *IEEE Comm. Mag.*, p. 26-35, jul., 1994.