

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**DINÂMICA DE CÉLULAS DE
PONTOS QUANTICOS ACOPLADAS**

MARCELO FERREIRA STELLA

ORIENTADOR: JOSÉ CAMARGO DA COSTA

**DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA
TELECOMUNICAÇÕES**

**PUBLICAÇÃO: PPGENE.DM - 300 /07
BRASÍLIA/DF: JUNHO – 2007**

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**DINÂMICA DE CÉLULAS DE
PONTOS QUÂNTICOS ACOPLADAS**

MARCELO FERREIRA STELLA

DISSERTAÇÃO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM TELECOMUNICAÇÕES.

APROVADA POR:

**Prof. José Camargo da Costa, Dr. (ENE-UnB)
(Orientador)**

**Prof. Alexandre Ricardo Soares Romariz, Dr. (ENE-UnB)
(Examinador Interno)**

**Prof. Paulo César de Moraes, Dr. (FIS-UnB)
(Examinador Externo)**

BRASÍLIA/DF, 20 DE JUNHO DE 2007

FICHA CATALOGRÁFICA

STELLA, MARCELO FERREIRA

Dinâmica de Células de Pontos Quânticos Acopladas [Distrito Federal] 2007.

xiii, 130p., 210 x 297 mm (ENE/FT/UnB, Mestre, Engenharia Elétrica, 2007).

Dissertação de Mestrado – Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Elétrica.

1. Pontos Quânticos

2. Nanoeletrônica

3. Autômatos Celulares

4. Simulação Numérica

I. ENE/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

STELLA, M. F. (2007). Dinâmica de Células de Pontos Quânticos Acopladas. Dissertação de Mestrado em Engenharia Elétrica, Publicação PPGENE.DM 300/07, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 130p.

CESSÃO DE DIREITOS

AUTOR: Marcelo Ferreira Stella.

TÍTULO: Dinâmica de Células de Pontos Quânticos Acopladas.

GRAU: Mestre

ANO: 2007

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte dessa dissertação de mestrado pode ser reproduzida sem autorização por escrito do autor.

Marcelo Ferreira Stella
SQN 214 D 513, Asa Norte.
70.873-040 – Brasília – DF – Brasil.

À minha esposa e à minha mãe

AGRADECIMENTOS

Agradeço primeiro a Deus por me permitir chegar até aqui, sem Ele, nada é possível e neste trabalho, não seria diferente. Agradeço também a minha amada esposa que soube me acompanhar e incentivar pelo caminho e a quem atribuo a parcela mais significativa desta conquista. Agradeço sempre a minha querida Mãe pela sua constante presença e incessante torcida. Agradeço ainda ao meu orientador, José Camargo, pela paciência e motivação dispensados compreendendo as dificuldades e sabendo orientar e aconselhar com maestria insuperável. Tenho também uma dívida de gratidão ao amigo Marcus Batistuta pela companhia, parceria e pelos constantes ensinamentos durante a jornada.

RESUMO

A proposta desta dissertação de Mestrado orienta-se na construção de um modelo numérico, baseado na solução da equação de Schrödinger, para o estudo da dinâmica de transporte em um conjunto de células de pontos quânticos acopladas lateralmente, que é a base para a construção de Automatos Celulares em dispositivos Quânticos - QCA's (Quantum-dot Cellular Automata).

Para tanto, faz uso do modelo em uma dimensão que descreve o comportamento biestável do elétron excedente em uma única célula com dois pontos quânticos acoplados. Também promove o acoplamento lateral entre essas células pelo uso da forma integral da Equação de Poisson no cálculo de potenciais eletrostáticos para a construção de circuitos mais complexos.

Este estudo, além de permitir a descrição do comportamento e a avaliação do desempenho (tempo para a estabilidade, resposta em frequência) de tais circuitos, chega à análise da transmissão de informações binárias para arranjos de células acopladas. Em paralelo, também é possível avaliar os modelos numéricos utilizados e desenvolver estratégias de otimização para a melhoria dos processos computacionais envolvidos.

ABSTRACT

This Master Degree dissertation presents a numerical model, based on Schrödinger's equation, to study the dynamical transport on laterally-coupled quantum dot cells. Those cells are the basis for constructing Quantum-dot Cellular Automata (QCA).

A simple one-dimensional model was used for the wavefunction dynamic analysis and a simple numerical scheme for solving electrostatic potentials was developed. Behavior prediction and performance evaluation (i.e. settling time, frequency response) have been obtained. The developed simulation strategy allows the calculation of state transit times along chains of coupled quantum dots cells.

The numerical algorithm was implemented in a MATLAB code for better performance, by using optimization strategies like array mathematics, variable's pre-allocation, minimization of "for" loops, allowing the simulation of even more complex QCA's arrays.

ÍNDICE

1. INTRODUÇÃO	1
2. REVISÃO BIBLIOGRÁFICA	3
2.1. LEI DE MOORE	3
2.2. AUTOMATOS CELULARES COM PONTOS QUANTICOS (QCA)	6
2.3. SIMPLIFICAÇÕES DA TEORIA ORTODOXA	8
2.3.1. BLOQUEIO COULOMBIANO	9
3. DESENVOLVIMENTO	11
3.1. MODELO NUMÉRICO DA SOLUÇÃO DA EQUAÇÃO DE SCHRÖDINGER	11
3.1.1. EQUAÇÃO DE SCHRÖDINGER DEPENDENTE DO TEMPO	11
3.1.2. PROCESSO DISSIPATIVO	14
3.1.3. MÉTODO NUMÉRICO - MODELO 1D	16
3.1.4. IMPLEMENTAÇÃO DO MÉTODO NUMÉRICO – MODELO 1D	17
3.2. ACOPLAMENTO ENTRE CÉLULAS	20
3.2.1. O POTENCIAL DE UM SISTEMA DE CARGA: CAMPO CONSERVATIVO	20
3.2.2. MÉTODO NUMÉRICO – MODELO 1D	21
3.2.3. IMPLEMENTAÇÃO DO ACOPLAMENTO PELO MÉTODO NUMÉRICO	23
4. RESULTADOS E DISCUSSÃO	25
4.1. DESCRIÇÃO DA CÉLULA BÁSICA	25
4.2. CENÁRIO COM 2 ELÉTRONS LIVRES	28
4.3. CENÁRIO COM 4 CÉLULAS ACOPLADAS	33
4.4. CENÁRIO COM 6 CÉLULAS ACOPLADAS	41
5. CONCLUSÃO	51
6. REFERÊNCIA BIBLIOGRÁFICA	53
APÊNDICE A - CÓDIGO	55
APÊNDICE B - PROCESSO DE OTIMIZAÇÃO E ADAPTAÇÃO	67

I. ORGANIZAÇÃO DO CÓDIGO ORIGINAL _____	67
II. OTIMIZAÇÃO NO MATLAB _____	68
III. ADAPTAÇÃO DO CÓDIGO PARA MÚLTIPLAS CÉLULAS _____	72
<u>APÊNDICE C - TRABALHOS CORRELATOS</u> _____	74
<u>APÊNDICE D - PUBLICAÇÕES REALIZADAS</u> _____	88

ÍNDICE DE FIGURAS

Figura 2-1 – Gráfico original desenhado por Moore em 1965	4
Figura 2-2 – Evolução dos Processadores Intel sob a perspectiva da Lei de Moore	4
Figura 2-3 – Pontos Quânticos Acoplados	6
Figura 2-4 – Propagação de estados em QD's	7
Figura 2-5 – Dinâmica dos Tunelamentos – Diagrama Esquemático	7
Figura 2-6 – Bloqueio Coulombiano – Junção Túnel	9
Figura 3-1 – Descrição do Potencial de Confinamento (concepção física 3D e numérica 1D)	12
Figura 3-2 – Descrição gráfica de Ψ em $t = 0$	13
Figura 3-3 – Par de Poços Quânticos Acoplados com Fonte de Tensão externa	14
Figura 3-4 – Fluxograma Básico	17
Figura 3-5 – 4 Células Acopladas Lateralmente	20
Figura 3-6 – Representação do Array Numérico usado nas simulações	22
Figura 3-7 – Diagrama de acoplamento por Poisson	22
Figura 4-1 – Célula básica utilizada nas simulações (modelo físico e numérico)	25
Figura 4-2 – Simulação com 4 Células – Diagrama Esquemático	26
Figura 4-3 – Dinâmica dos Tunelamentos – Diagrama Esquemático	26
Figura 4-4 – Variação temporal de V_E e seu efeito na célula 1 em t_m	27
Figura 4-5 – Distância entre células acopladas	27
Figura 4-6 – Células sem poços quânticos – “Elétron Livre”	28
Figura 4-7 – Simulação com 2 células – Elétrons Livres – Estado Inicial	29
Figura 4-8 – Simulação com 2 células – Elétrons Livres – Estado Final	30
Figura 4-9 – Posicionamento lateral das células	30
Figura 4-10 – Simulação com 2 células – Dinâmica da Célula 1	31
Figura 4-11 – Simulação com 2 células – Dinâmica da Célula 2	31
Figura 4-12 – Simulação com 2 células – Distribuição de carga nas células	32
Figura 4-13 – Simulação com 2 células – Corrente no centro das células	33
Figura 4-14 – Células em Autômatos com QCA's	33
Figura 4-15 – Simulação com 4 Células – Diagrama Esquemático	34
Figura 4-16 – Simulação com 4 Células – Estado Inicial	34
Figura 4-17 – Simulação com 4 Células – Estado Final	35
Figura 4-18 – Simulação com 4 células – Dinâmica da Célula 1	36
Figura 4-19 – Simulação com 4 células – Dinâmica da Célula 2	36
Figura 4-20 – Simulação com 4 células – Dinâmica da Célula 3	37
Figura 4-21 – Simulação com 4 células – Dinâmica da Célula 4	37
Figura 4-22 – Simulação com 4 células – Carga nos Poços	38
Figura 4-23 – Simulação com 4 células – Corrente no centro das células	39
Figura 4-24 – Simulação com 6 Células – Diagrama Esquemático	41
Figura 4-25 – Simulação com 6 Células – Estado Inicial	41

<i>Figura 4-26 – Simulação com 6 Células – Estado Final</i>	42
<i>Figura 4-27 – Simulação com 6 células – Dinâmica da Célula 1</i>	43
<i>Figura 4-28 – Simulação com 6 células – Dinâmica da Célula 2</i>	43
<i>Figura 4-29 – Simulação com 6 células – Dinâmica da Célula 3</i>	44
<i>Figura 4-30 – Simulação com 6 células – Dinâmica da Célula 4</i>	44
<i>Figura 4-31 – Simulação com 6 células – Dinâmica da Célula 5</i>	45
<i>Figura 4-32 – Simulação com 6 células – Dinâmica da Célula 6</i>	45
<i>Figura 4-33 – Simulação com 6 células – Carga nos Poços (Células 1, 2 3 e 4)</i>	46
<i>Figura 4-34 – Simulação com 6 células – Carga nos Poços (Células 5 e 6)</i>	47
<i>Figura 4-35 – Simulação com 6 células – Corrente no centro das células (Células 1, 2, 3 e 4)</i>	48
<i>Figura 4-36 – Simulação com 6 células – Corrente no centro das células (Células 5 e 6)</i>	49
<i>Figura 4-37 – Simulação com 6 células – Zoom da corrente no centro das células 5 e 6</i>	50
<i>Figura B-1 – Tela Inicial do Profiler do Matlab</i>	70
<i>Figura B-2 – Detalhamento de Custo no Profiler do Matlab</i>	70

LISTA DE SÍMBOLOS

- A – área
- C – capacitância
- C_{eff} – capacitância efetiva
- e – carga fundamental do elétron ($1,602 \times 10^{-19}$ coulombs)
- E – energia
- E_{Ω} – campo elétrico de desaceleração
- f – frequência
- h – constante de Planck ($6,626 \times 10^{-34}$ J.s)
- \hbar – constante de Planck barrada ($1,055 \times 10^{-34}$ J.s)
- i – número imaginário ou índice inteiro
- I – corrente
- j – número imaginário ou índice inteiro
- J – densidade de corrente
- L – comprimento
- m – massa ou índice inteiro
- n – densidade de probabilidade
- q – carga elétrica
- Q – carga elétrica total
- r – posição
- S – fator de amortecimento
- S_R – fator de amortecimento por dissipação ohmica
- S_{Ω} – fator de amortecimento por radiação eletromagnética
- t – tempo
- V – potencial
- V_e – fonte de tensão externa
- x – coordenada de posição
- X – componente real da função de onda
- Y – componente imaginária da função de onda
- α – constante auxiliar
- β – constante auxiliar
- ε – permissividade elétrica

Ψ – função de onda

ρ – densidade de carga

η – impedância do meio

ρ_{Ω} – resistividade

π – constante pi (3,1415...)

1. INTRODUÇÃO

A constante e acelerada evolução que permeia o ambiente tecnológico atual requer um crescimento cada vez maior das capacidades de processamento e armazenamento em circuitos integrados. Essa demanda sugere uma evolução à tecnologia de dispositivos direcionando seu desenvolvimento para produtos cada vez mais densos e velozes [1].

As tecnologias atuais para a construção de circuitos integrados, com maior representação nos transistores CMOS, não conseguirão acompanhar esta crescente demanda por integração [2][1]. O advento de novas tecnologias torna-se necessário para atingir densidades em escalas de poucos nanômetros e velocidades da ordem de Tera Hertz, por exemplo.

Dentre as novas tecnologias que estão sendo propostas, o uso de pontos quânticos acoplados para a construção de Autômatos Celulares (QCA – Quantum-dot Cellular Automata) vem sendo difundido como uma solução válida para o problema. Uma estrutura conceitual composta por células de pontos quânticos, acoplados entre si, foi originalmente proposta por Lent *et al* [3] permitindo a construção de QCA's sem estar restrita a nenhuma tecnologia em particular.

Este trabalho orienta-se à construção de um modelo numérico, baseado na solução da equação de Schrödinger, para o estudo da dinâmica de transporte em um conjunto de células de pontos quânticos acopladas lateralmente, que é a base para a construção de QCA's. Para tanto, faz uso do modelo em uma dimensão que descreve o comportamento biestável do elétron excedente em uma única célula com dois pontos quânticos acoplados. Também promove o acoplamento lateral entre essas células pelo uso da forma integral da Equação de Poisson no cálculo de potenciais eletrostáticos [4] para a construção de circuitos mais complexos.

Este estudo, além de permitir a descrição do comportamento e a avaliação do desempenho (tempo para a acomodação, resposta em frequência) de tais circuitos, chega à análise da transmissão de informações binárias por arranjos de células acopladas. Além disso, também é possível avaliar os modelos numéricos utilizados e desenvolver estratégias de otimização para a melhoria dos processos computacionais.

Em resumo, o capítulo 2 faz uma apresentação geral dos conceitos envolvidos no projeto passando pela sua contextualização e percorrendo os fundamentos de QCA's e suas características para a modelagem proposta. No capítulo 3 são apresentadas as metodologias utilizadas na construção do modelo numérico, bem como os recursos de otimização dos processos computacionais. Por fim, no capítulo 4 são apresentados alguns resultados de simulações pela modelagem de diferentes estruturas funcionais. Foi avaliado, em primeiro lugar, o comportamento dos elétrons em duas células acopladas lateralmente, sem a presença dos poços quânticos, para observar a repulsão entre eles. Em seguida, um conjunto com quatro células demonstram o comportamento básico encontrado na literatura [5][6][7] de transporte de informações em QCA's. Por fim, um conjunto com seis células faz a avaliação da dependência o transporte de informações em QCA's e as características físicas da estrutura que os compõe e que permitem ou não seu funcionamento conforme desejado. Vale ressaltar que, por estar sendo utilizado um modelo em uma dimensão, este trabalho avalia, em primeira análise, poços quânticos.

2. REVISÃO BIBLIOGRÁFICA

A evolução tecnológica atual, que é imposta aos circuitos integrados, permite estimar que o desenvolvimento de sistemas computacionais possui uma tendência para circuitos cada vez mais rápidos e com maior capacidade de processamento. O uso de circuitos digitais participando cada vez mais em nosso cotidiano é uma constante, fornecendo novas e melhores ferramentas para as mais diversas aplicações. Como consequência, a evolução natural desses circuitos, em busca de maiores capacidades de processamento e armazenagem, tende para dispositivos cada vez menores e mais rápidos. [1]

Este capítulo irá fazer uma apresentação geral dos conceitos envolvidos no projeto partindo da Lei de Moore, que estima qual a evolução tecnológica para circuitos integrados nos próximos anos fornecendo insumos que estimulam a busca de novas soluções para a construção de circuitos [8]. Serão feitas considerações sobre QCA's, sua descrição e caracterização. Por fim, fará uma introdução aos conceitos envolvidos na modelagem numérica proposta preparando o ambiente para o capítulo seguinte, da metodologia.

2.1. LEI DE MOORE

A Lei de Moore parte de uma observação empírica feita em 1965 [9][2] que o custo mínimo de um componente para o número de transistores em um circuito integrado duplica a cada 24 meses. Esta afirmação, que é atribuída a Gordon E. Moore, nascido em 1929 e cofundador da Intel, muitas vezes, é corrigida para o período de 18 meses para a duplicação do número de transistores.

A figura 2-1 apresenta o gráfico original, apresentado por Moore em 1965, para a composição de seu artigo. Este gráfico apresenta o Custo Médio de Manufatura por Componente em relação ao Número de Componentes por Circuito Integrado com o passar dos anos.

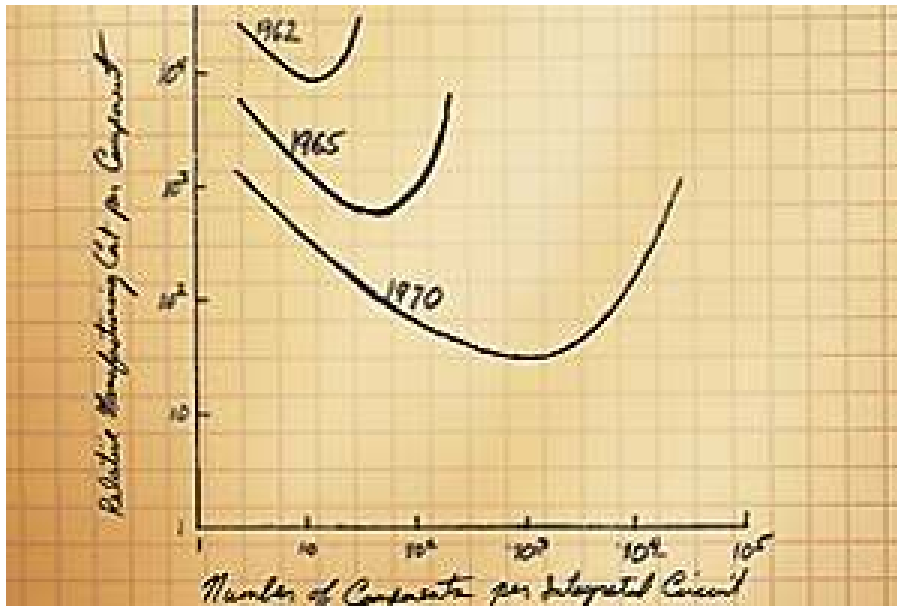


Figura 2-1 – Gráfico original desenhado por Moore em 1965

Esta proposição é genérica, pois entende que a complexidade de um circuito está relacionada apenas ao número de transistores sem levar em conta sua funcionalidade. Se forem levantadas outras questões como por exemplo as diferenças entre um *array* de memória e um processador, certamente a Lei de Moore se torna mais questionável.

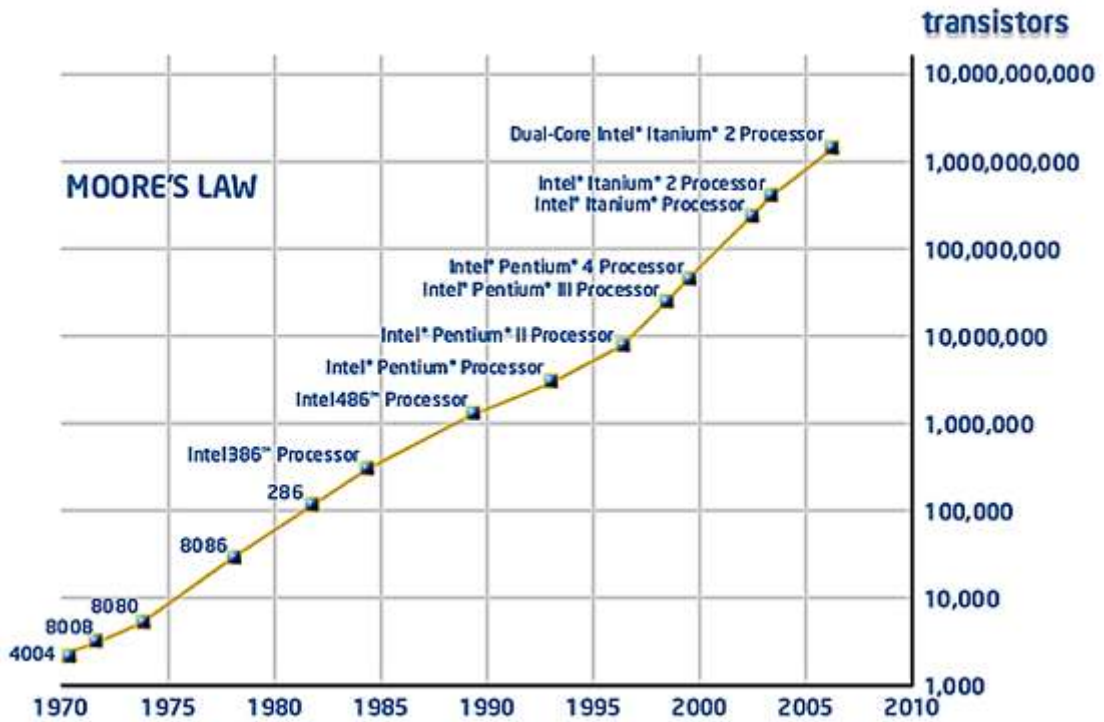


Figura 2-2 – Evolução dos Processadores Intel sob a perspectiva da Lei de Moore

A observação de Moore não foi chamada de Lei por ele mesmo. Com o tempo, através da observação, pode ser notado que a proposta original permanecia fiel. Na figura 2-2 pode ser observada a evolução dos processadores Intel, ao longo dos anos, em comparação à quantidade de transistores presentes em cada componente.

Contudo, como dito por ele mesmo, “esta tendência não pode permanecer para sempre. A natureza de exponenciais é que ao serem forçadas, eventualmente ocorre o desastre.” [10]. Dentre os principais fatores que tornam cada vez mais desafiadores os obstáculos para a produção de circuitos integrados, para a tecnologia CMOS, em função do aumento de densidade, podem ser citados [11]:

- A capacidade técnica em controlar resistências e capacitâncias parasitas em transistores nesta tecnologia;
- A limitação técnica em reduzir as resistências e capacitâncias em interconexões elétricas nos circuitos;
- A limitação técnica em manter controladas as cargas eletrostáticas próprias dos transistores que permitem à fonte (*gate*) controlar o chaveamento de estados;
- Flutuações ou imperfeições na dopagem dos semicondutores;
- Distribuição de energia para todos os transistores ao longo do circuito;
- Considerações térmicas resultantes desta mesma distribuição de energia e
- Resolver todos estes desafios mantendo os custos de produção em níveis aceitáveis.

Sendo assim, é prudente a busca por um novo paradigma voltado à computação que não se baseie nos transistores CMOS (que são os mais comuns). O paradigma computacional conhecido por Quantum-dot Cellular Automata (QCA) é uma possibilidade [12]. A computação por QCA está mais próxima da Lógica Booleana, por fazer uso da bi-estabilidade em sua concepção (item 2.2), estruturando-se sobre a polarização de células ao invés de voltagens e correntes para representar os dados binários. Além disso, os QCA's oferecem vantagens sobre a tecnologia convencional (CMOS) incluindo a compatibilidade com nanoestruturas, interconexões simplificadas e a possibilidade do aumento da densidade e velocidades computacionais.

2.2. AUTOMATOS CELULARES COM PONTOS QUANTICOS (QCA)

O contínuo aumento da densidade de elementos na construção de circuitos integrados tem permitido ganhos consideráveis em performance. Contudo, apesar de toda a evolução tecnológica envolvida, os circuitos atuais são construídos sob os mesmos paradigmas de seus mais primitivos antecessores [5]. Existe, portanto certa expectativa que a construção de circuitos com altos níveis de integração sejam baseados em novos paradigmas envolvendo arquiteturas interconectadas localmente como em Automatos Celulares e Redes Neurais.

Nesta necessidade de integração, o uso de estruturas em escalas nanométricas, onde a Mecânica Quântica se torna necessária para a compreensão do funcionamento dos dispositivos, torna-se preponderante. Desta forma, um QCA (*Quantum Cellular Automata*) faz uso de dispositivos quânticos em escalas nanométricas para a construção de novas arquiteturas de Autômatos Celulares.

O conceito de QCA (*Quantum Cellular Automata*) não está restrito a nenhuma tecnologia em particular podendo ser construídos, por exemplo, sobre pontos quânticos em semicondutores, junções-túnel em metais, estruturas com nano-tubos de carbono, *self-assembled dots* e até moléculas [13]. Este trabalho fez uso de Pontos Quânticos Acoplados – *Coupled Quantum Dots* (CQD) através de junções-túnel. A figura 2-3 abaixo exemplifica este conceito pelo uso de estruturas quânticas biestáveis. Esta escolha é feita pela proximidade de uma codificação binária natural nos dispositivos [5][14].

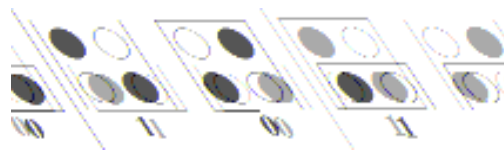


Figura 2-3 – Pontos Quânticos Acoplados

A redistribuição de cargas nas células, em busca da estabilidade, se faz de tal forma que a energia total do sistema seja minimizada [15]. Assim, pela simetria da estrutura, podem ser definidos dois estados discretos [6]. No exemplo da célula com quatro pontos quânticos (fig 2-3), há dois elétrons em excesso que tendem a ocupar os cantos opostos da estrutura.

Esta energia que os repele provém das forças eletrostáticas existentes. Sendo assim, para as duas possibilidades de ocupação dos pontos quânticos em cada célula (bi-estabilidade) podem ser atribuídos os estados lógicos fundamentais 0 e 1. Da mesma forma que para os circuitos CMOS, as células de pontos quânticos acopladas devem ser dimensionadas corretamente. Isto para que a bi-estabilidade ocorra e para que as transições obedeam a uma dinâmica desejada para o dispositivo [16]. A figura abaixo exemplifica esta necessidade.

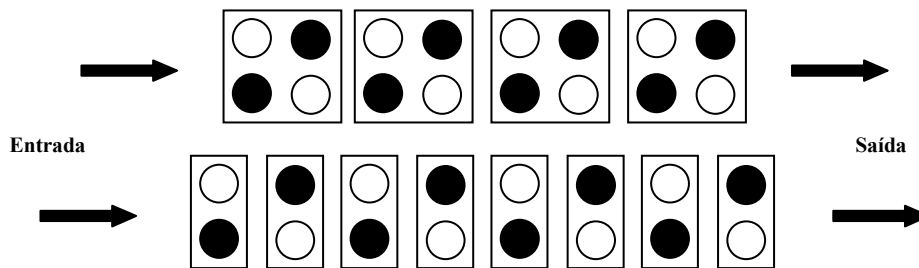


Figura 2-4 – Propagação de estados em CQD's

A propagação dos estados ocorre principalmente pela ação dos campos resultantes da redistribuição de cargas dentro das células e pelo processo de Tunelamento (barreira com energia maior que a do elétron e espessura da ordem de 1 nm – fig.4-1, compatíveis com este processo). Ou seja, há uma polarização de cada célula, que influencia o estado das suas vizinhas. Portanto, se for alterado o estado da primeira célula à esquerda por alguma ação externa, espera-se que as demais células se reorganizem propagando a informação de um lado ao outro da estrutura. A figura 2-5 apresenta esta propagação de estados.

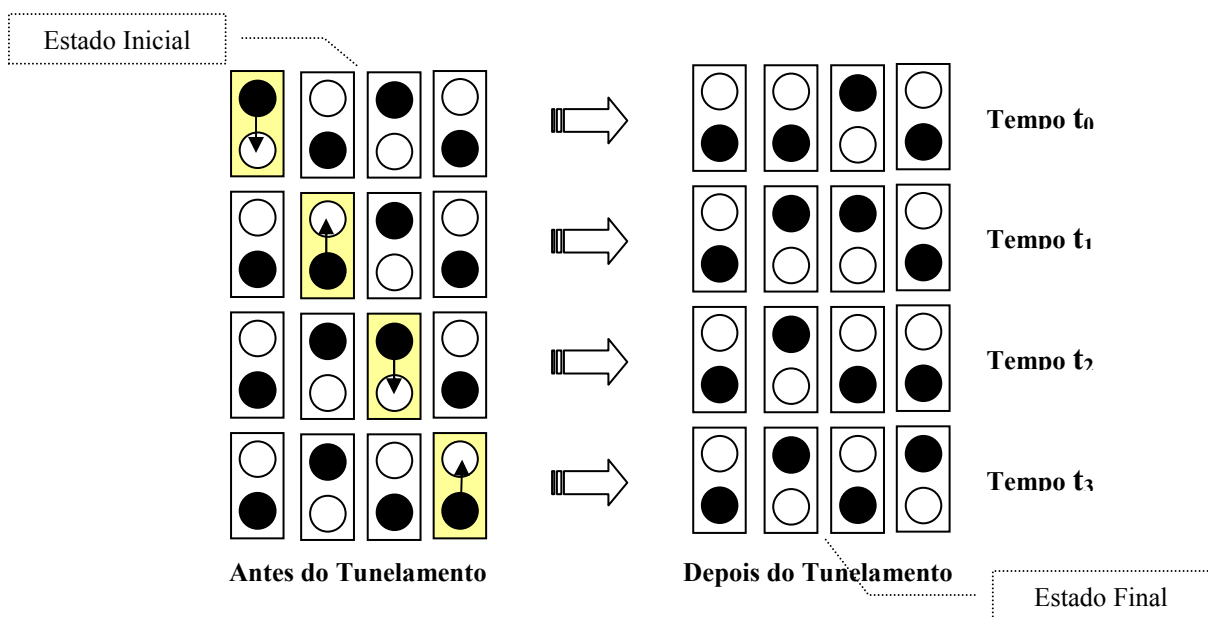


Figura 2-5 – Dinâmica dos Tunelamentos – Diagrama Esquemático

2.3. SIMPLIFICAÇÕES DA TEORIA ORTODOXA

A Teoria Ortodoxa tem guiado o desenvolvimento no modelamento e simulação de dispositivos mono-elétron [17]. Entretanto, esta teoria faz uso de três simplificações significativas que devem ser questionadas. Seriam elas:

1. A quantização das auto-energias do elétron confinado é ignorada: Variações nestes níveis de energia são assumidas irrelevantes em comparação com a energia térmica ($\Delta E \ll k_B T$). Contudo, o Bloqueio Coulombiano (item 2.3.1) somente pode ser observado claramente em temperaturas muito baixas, ou quando as dimensões e a auto-capacitância das ilhas de potencial (pontos ou poços) são muito pequenos. Desta maneira a quantização da energia do elétron não pode ser ignorada em estruturas muito pequenas.
2. O tempo de tunelamento é desprezado: O tempo de tunelamento é normalmente considerado como um valor constante, calculado por aproximação, combinando capacitâncias ou auto-capacitâncias e resistências de tunelamento, ou simplesmente considerando o tunelamento como instantâneo. Intuitivamente podemos dizer que nenhuma boa consideração sobre os limites de performance do dispositivo, como frequência máxima de funcionamento e tempo mínimo de transição de estados pode ser feita com o uso destas aproximações.
3. Eventos de co-tunelamento são ignorados: Os eventos de co-tunelamento são considerados desprezíveis se a resistência equivalente de todas as junções-túnel é muito maior que a unidade de resistência quântica. Esta característica limita a exploração da geometria dos dispositivos, já que eventos de co-tunelamento intencionais não podem ser explorados.

Estas simplificações vêm sendo consideradas extensivamente e novos e melhores modelos, que incluem os fenômenos ignorados pela Teoria Ortodoxa vêm sendo propostos [27]. Neste trabalho, uma maior atenção é dada para o processo de dissipação Ohmico e para as dinâmicas de tunelamento.

2.3.1. Bloqueio Coulombiano

O Bloqueio Coulombiano [18] é um conceito utilizado quando se trata de dispositivos mono-elétron e, apesar de não exercer maior influência sobre a construção de QCA's, é apresentado para fins de equalização conceitual.

Se um condutor é interrompido por uma junção túnel, as cargas elétricas irão se mover através do sistema e se acumular na superfície próxima ao material isolante da junção (o outro lado da junção também receberá um acúmulo de cargas com sinal oposto). Pode-se imaginar, portanto, uma carga Q depositada na superfície em consequência da redistribuição dos elétrons ao redor da junção. Por outro lado, a mecânica quântica diz que a variação de Q só pode ser dada em quantidades discretas: quando um elétron tunela de um lado para o outro da junção alterando a carga em $-e$ ou $+e$, dependendo da direção de tunelamento (fig 2-6).

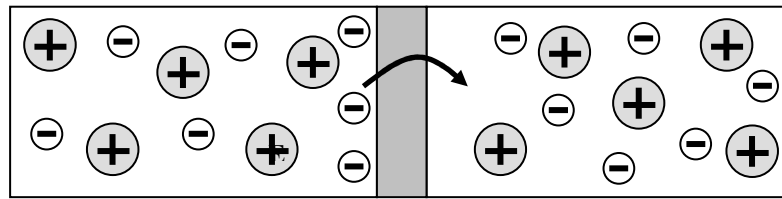


Figura 2-6 – Bloqueio Coulombiano – Junção Túnel

Este fenômeno pode ser observado quando as junções túnel são muito pequenas e a temperatura ambiente é muito baixa (temperaturas baixas diminuem as flutuações térmicas que estimulam a movimentação dos elétrons). No caso citado, se a carga Q , próxima a junção, for maior que $+e/2$, um elétron pode tunelar pela junção em uma determinada direção, subtraindo “ e ” de Q . O elétron faz este movimento, pois, neste processo, há uma redução da energia eletrostática do sistema. Da mesma forma, se a carga Q for menor que $-e/2$, um elétron pode tunelar através da junção só que na direção oposta, somando “ e ” a Q e, novamente, diminuindo a energia do sistema.

Contudo, se Q estiver entre $+e/2$ e $-e/2$, o tunelamento em qualquer direção iria aumentar a energia total do sistema. Sendo assim, se a carga inicial Q estiver dentro destes limites, o tunelamento não ocorrerá. Esta supressão de tunelamento é conhecida por Bloqueio Coulombiano. Vale ressaltar que este fenômeno em particular não ocorre em QCA's em função do número de elétrons envolvidos nesta estrutura.

O que aconteceria se fosse ligada uma fonte de corrente a este sistema? Inicialmente, se a carga de superfície Q for zero, então o sistema está dentro dos limites do Bloqueio Coulombiano e o tunelamento não está ocorrendo. Entretanto, a corrente proveniente da fonte através dos condutores iniciaria a carregar Q continuamente. Se a carga depositada exceder um dos limites ($+e/2$ ou $-e/2$), o tunelamento passa a ser possível. Um elétron, portanto, iria tunelar através da junção trazendo, novamente, a carga Q de volta à região de equilíbrio onde o tunelamento não ocorre (Bloqueio Coulombiano). A corrente continuaria adicionando carga a um dos lados da junção a uma taxa constante reiniciando o processo. Esta repetição produz o que é conhecido por *Single Electron Tunneling (SET) Oscillations*: onde a voltagem muda periodicamente com uma frequência igual à corrente dividida pela carga fundamental do elétron.

3. DESENVOLVIMENTO

Este capítulo irá descrever a concepção do modelo proposto, partindo da equação de Schrödinger dependente do tempo, passando pelos processos dissipativos adotados até o acoplamento entre células pelo uso da forma integral da equação de Poisson para potenciais eletrostáticos [19][20][21].

Vale ressaltar que este trabalho tem como ponto de partida um projeto já desenvolvido, que envolve a construção de um modelo que descreve, numericamente, o comportamento de dispositivos mono-elétron focando, principalmente, na dinâmica dos processos de tunelamento dissipativos em células de pontos quânticos acoplados [22]. Sendo assim, a contribuição aqui proposta se orienta, fundamentalmente, na implementação numérica do acoplamento lateral de tais células permitindo estudos como: transporte de informações em arrays de células, tempos de propagação em tais arrays, simulação de estruturas mais complexas.

A ferramenta utilizada para a simulação computacional do modelo numérico foi o MatLab [23]. Após execução de algumas simulações, foi observada a necessidade de ajustes no modelo computacional original [22]. Tais ajustes, envolvendo aceleração do código [24] e adaptação para simulação de múltiplas células, também serão descritos neste capítulo.

3.1. MODELO NUMÉRICO DA SOLUÇÃO DA EQUAÇÃO DE SCHRÖDINGER

3.1.1. Equação de Schrödinger dependente do tempo

Este projeto tem como base a solução da equação de Schrödinger dependente do tempo (3-1) para um único elétron.

$$\left[-\frac{\hbar}{2m} \nabla^2 + eV(\vec{r}) \right] \Psi(\vec{r}, t) = i\hbar \frac{\partial \Psi(\vec{r}, t)}{\partial t} \quad (3-1)$$

Na equação (3-1) temos que a função de onda (Ψ) é complexa e, a partir da definição de um potencial de confinamento V , independente do tempo, é possível calcular sua evolução no tempo e no espaço.

Neste caso, o elétron não é considerado como carga pontual, mas como uma distribuição de probabilidade de densidade de carga ρ (equação 3-2) com uma distribuição de probabilidade da densidade de corrente J (equação 3-3).

$$\rho = qn(x,t) = q(\Psi(x,t)\Psi^*(x,t)) \quad (3-2)$$

$$J = -\frac{iq\hbar}{2m}((\nabla\Psi)^*\Psi - \Psi^*(\nabla\Psi)) \quad (3-3)$$

A equação que demonstra que a continuidade para a carga e corrente é respeitada é:

$$\nabla \cdot \vec{J} = -\frac{\partial \rho}{\partial t} \quad (3-4)$$

O potencial de confinamento foi escolhido, inicialmente, de forma a representar uma célula contendo um par de pontos quânticos como na figura 3-1 abaixo.

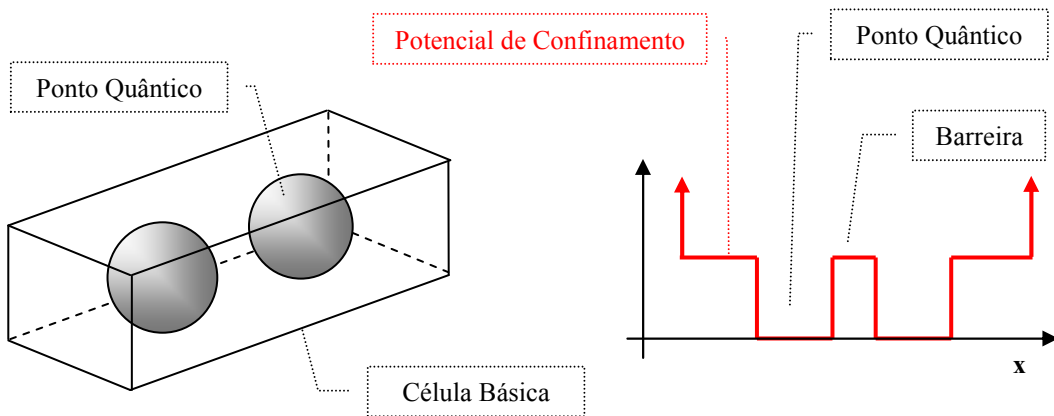


Figura 3-1 – Descrição do Potencial de Confinamento (concepção física 3D e numérica 1D)

Para que a evolução no tempo deste modelo represente a realidade física, a função de onda (Ψ) inicial, em $t = 0$, deve ser fisicamente consistente, seja qual for sua composição de estados. A figura 3-2 demonstra graficamente este estado inicial.

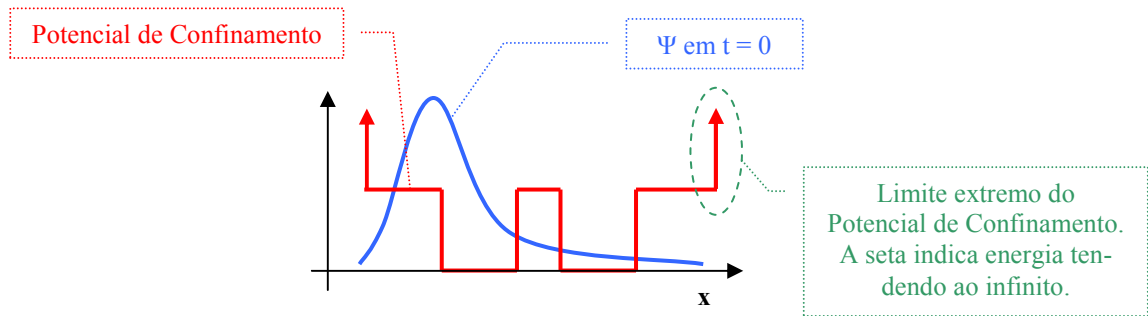


Figura 3-2 – Descrição gráfica de Ψ em $t = 0$

Para este primeiro instante no tempo ($t = 0$), a função de onda Ψ é escolhida em um estado que não o de menor energia do elétron, respeitando os princípios físicos dentro do modelo proposto. Durante a execução da simulação numérica, ocorre a evolução no tempo de Ψ que, pela geometria física da estrutura e demais processos envolvidos (dissipação, acoplamento), atinge um estado de menor energia. Desta forma, é possível avaliar a evolução do comportamento do elétron, pela sua curva função densidade de probabilidades, ao longo do tempo em resposta ao ambiente em que se encontra (descrito pelo modelo).

Sendo assim, um modelo em uma dimensão se inicia representando a dinâmica dos eventos de tunelamento dos elétrons em pontos e/ou poços quânticos acoplados, levando em conta princípios físicos fundamentais. Dentre estes princípios são destacadas as conservações de carga e de energia. O modelo também supera algumas das limitações introduzidas pelas simplificações feitas na Teoria Ortodoxa (item 2.3).

Em linhas gerais, o esquema utilizado nas simulações pode ser observado na fig 3-3, onde dois poços quânticos, acoplados através de uma junção de tunelamento, estão sobre a influência de um campo elétrico externo fornecido por uma fonte de tensão. Por simplicidade de representação não estarão sendo mais apresentados, nas figuras seguintes, os limites extremos do potencial de confinamento (fig 3-2) das células.

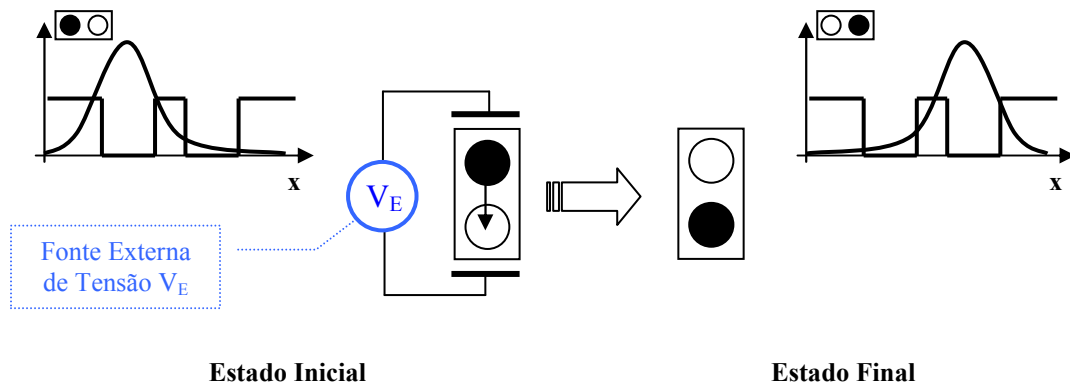


Figura 3-3 – Par de Poços Quânticos Acoplados com Fonte de Tensão externa

Neste modelo são considerados os seguintes aspectos:

1. Existência de uma fonte de tensão V_E externa que fornece uma ddp à célula;
2. Existência de um elétron excedente em algum lugar do sistema (fig. 3-2);
3. A energia inicial que este elétron excedente possui pode ser resultado de alguma superposição de mais de um estado de energia permitido, ou seja, no estado inicial o elétron possui energia em excesso;
4. Inexistência de cargas adicionais livres para se movimentar pela ação do campo elétrico externo aplicado;
5. Inexistência de campos magnéticos;
6. A energia está confinada na estrutura do dispositivo;
7. Eventos de tunelamento são considerados somente entre os poços, em um tempo finito, na presença de um processo dissipativo (que será descrito no item 3.1.2 a seguir).

3.1.2. Processo Dissipativo

A equação 3-5 abaixo é a equação de Schrödinger que representa a dinâmica do elétron excedente, com um fator de amortecimento S incluído em seu hamiltoniano.

$$\left[-\frac{\hbar}{2m} \nabla^2 + qV + S \right] \Psi(x,t) = i\hbar \frac{\partial \Psi(x,t)}{\partial t} \quad (3-5)$$

O fator S permite a dissipação da energia do elétron confinado, sendo resultado de processos dissipativos [19][20][21]. Isto significa que depois de um tempo finito o elétron perderá energia, decaindo para o seu estado fundamental, tendo como solução final uma função de onda Ψ estacionária. Nesse sistema dois processos dissipativos são importantes ($S = S_R + S_\Omega$). O primeiro se dá através da radiação eletromagnética de energia produzida pela aceleração da densidade de carga (S_R). O segundo se dá através do processo de dissipação ôhmica (S_Ω). Os processos de dissipação em dispositivos mono-elétron são bem documentados por Davidson [25]. A perda ôhmica é resultante da interação da densidade de corrente do elétron com a resistividade dos materiais (ρ_Ω). O modelo 1D neste trabalho utiliza um campo elétrico de desaceleração definido pela equação 3-6.

$$E_\Omega = \rho_\Omega \cdot J \quad (3-6)$$

Onde J é a densidade de corrente e ρ_Ω é a resistividade do material. A formulação completa do fator de amortecimento S , com a ação do campo de desaceleração incluso, pode ser vista na equação 3-7 abaixo.

$$S(x,t) = S_R + S_\Omega = S_R + q \int E_\Omega \cdot dx \quad (3-7)$$

3.1.3. Método Numérico - Modelo 1D

O método numérico utilizado é a regra da atualização discreta para a equação de Schrödinger unidimensional, baseado na técnica desenvolvida por Fredkin e Barton em 1975, que é reversível no tempo e garante a conservação de carga, como demonstrado por Feynman [26]. As equações estão apresentadas abaixo onde \mathbf{m} é um número inteiro que representa a posição, em valores discretos, no vetor para uma única dimensão (item 3.2.2).

$$\Psi = X + jY \quad (3-8)$$

$$X_m = X_m - \left[\alpha \frac{\Delta t}{\Delta x^2} (Y_{m+1} - 2Y_m + Y_{m-1}) + \beta \Delta t V_m^n Y_m \right] \quad (3-9)$$

$$Y_m = Y_m - \left[\alpha \frac{\Delta t}{\Delta x^2} (X_{m+1} - 2X_m + X_{m-1}) + \beta \Delta t V_m^{n+1} X_m \right] \quad (3-10)$$

Onde $\alpha = \frac{\hbar}{2m}$, e $\beta = -\frac{2}{\hbar}$.

As componentes da função de onda Ψ (real X e imaginária Y), em cada ponto do espaço são calculadas em seqüência. O potencial de confinamento V_m é recalculado numericamente a cada passo levando em consideração a distribuição de densidade de carga e de corrente, calculados de acordo com as equações 3-2 e 3-3 incluindo o processo dissipativo (equação 3-7). Vale ressaltar que os passos de tempo adotados nas simulações foram escolhidos de forma a garantir a estabilidade e convergência do algoritmo para os estados fundamentais disponíveis em função das condições de contorno e dentro das escalas de tempo fisicamente consistentes.

O programa desenvolvido parte de soluções consistentes de ψ para os elétrons nos momentos iniciais e apesar de não estar incluso um procedimento numérico para garantir a continuidade de ψ e ψ' em cada ponto da simulação entende-se que há consistência nas simulações em função do algoritmo e da formulação utilizados [26]. Contudo, esta validação deve ser incluída e está sugerida como uma continuidade deste trabalho.

3.1.4. Implementação do método numérico – Modelo 1D

A figura 3-4 apresenta o fluxograma básico do projeto onde estão representadas as operações executadas pelo modelo numérico implementado computacionalmente.

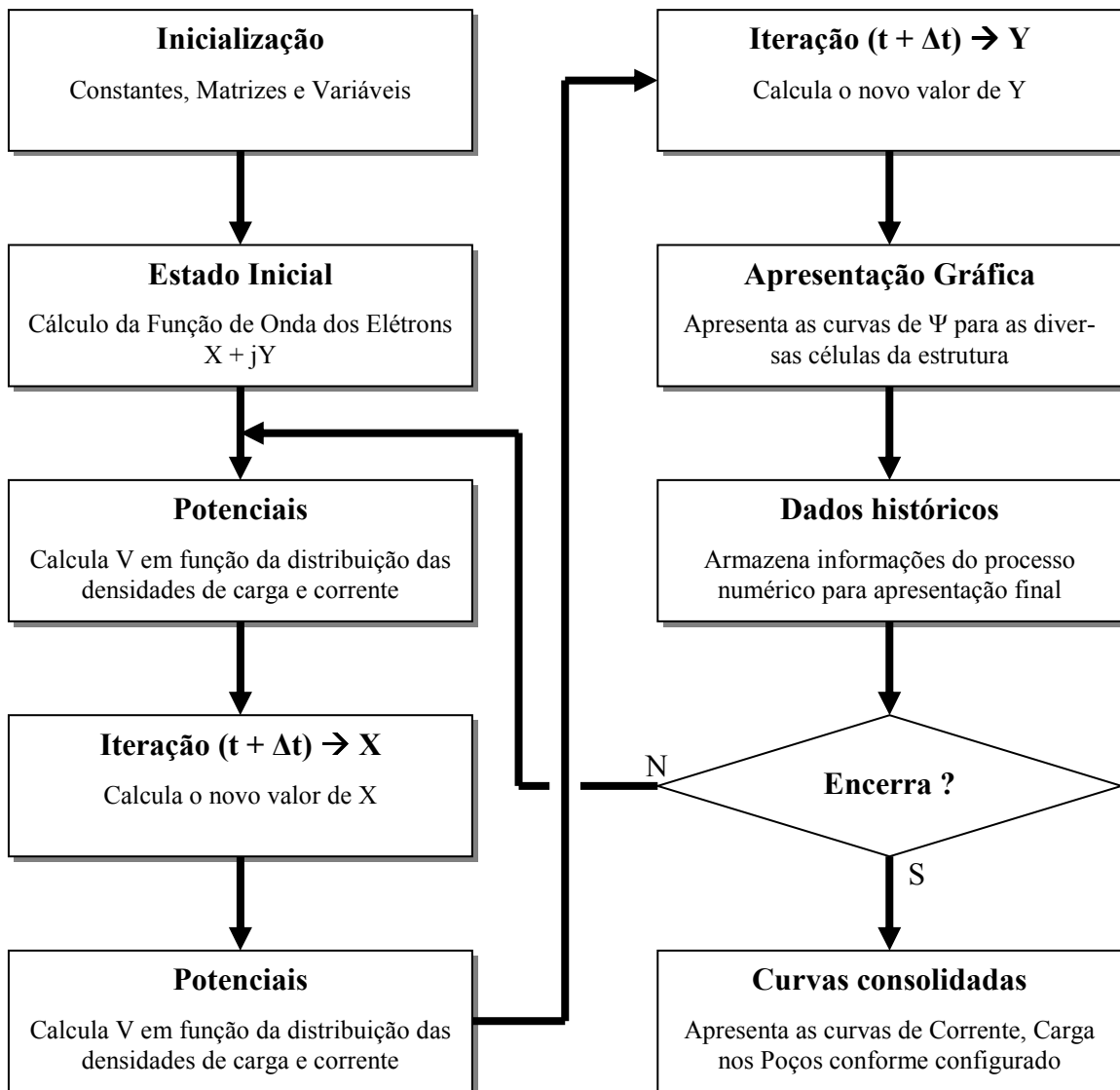


Figura 3-4 – Fluxograma Básico

O algoritmo desenvolvido no MatLab que realiza os cálculos da evolução temporal da função de onda Ψ pode ser visto a seguir. O segmento de código listado está apresentando apenas a etapa de cálculo de X e Y que são as partes real e imaginária da curva densidade de probabilidade do elétron.

```

while (A.cont < A.TempoTotal)

    % -----
    % Calcula X
    % -----

    % Calcula o Potencial de Confinamento Vtotal = Vtotal + VautoC + Vpoco + Vrampa;

    diff2Y = [zeros(M.qt,1), diff(EY,2,2), zeros(M.qt,1)];
    EX = EX - (C.alfa*(M.dt/(M.dx^2))*diff2Y + C.beta.*M.dt.*Vtotal.*EY);

    % -----
    % Calcula Y
    % -----

    % Calcula o Potencial de Confinamento Vtotal = Vtotal + VautoC + Vpoco + Vrampa;

    diff2X = [zeros(M.qt,1), diff(EX,2,2), zeros(M.qt,1)];
    EY = EY + (C.alfa*(M.dt/(M.dx^2))*diff2X+ C.beta.*M.dt.*Vtotal.*EX);

end

```

O cálculo do potencial de confinamento é obtido por 4 (quatro) curvas de potencial que representam papéis distintos. Seriam elas:

- ***Vtotal*** – representando a distribuição de densidade de carga e de corrente, calculados de acordo com as equações 3-2 e 3-3 incluindo o processo dissipativo (equação 3-7) (Os nomes das variáveis, após alterações no código, não são mais representativos de seu conteúdo):

```

diffX = [zeros(M.qt,1), diff(EX,1,2)];
diffY = [zeros(M.qt,1), diff(EY,1,2)];
EJ = -(C.q*C.h_/(C.m*M.A*M.dx))*(diffX.*EY - diffY.*EX);
EJ(:,1) = 0;
EJ(:,M.N) = 0;
Vtotal = M.dx .* M.R .* C.q .* cumsum(EJ,2);

```

- ***VautoC*** – Esta função representa o potencial resultante do acoplamento entre as células e será descrito no item 3.2;
- ***Vpoco*** – Este potencial corresponde a geometria do poço conforme figura 3-1 e foi implementada da seguinte forma:

```

% -----
% Monta Celulas com os Pocos Quanticos
% -----

function Monta_Celulas

    global A C M E;
    global Vtotal VautoC Vpoco Vrampa;
    global EX EY EJ;

    Vtotal = zeros(M.qt,M.N);           % Energia Potencial Total (Joules)
    VautoC = zeros(M.qt,M.N);          % Inicia Auto-Capacitancia (Joules)
    Vrampa = zeros(M.qt,M.N);          % Prepara Rampa (Joules)
    Vpoco = 7 * M.Efpi * ones(M.qt,M.N); % Inicia Celula - Poco (Joules)

    for k = M.P1.ini:M.P1.fim
        Vpoco(:,k) = 0;                 % Insere Poco 1
    end
    for k = M.P2.ini:M.P2.fim
        Vpoco(:,k) = 0;                 % Insere Poco 2
    end
end

```

- ***Vrampa*** – Este potencial corresponde a rampa introduzida na célula pela fonte de tensão V_E (figura 3-3):

```

% -----
% Insere Rampa
% -----

if (A.cont > A.RampaIni & A.cont < A.RampaPara)
    A.RampaCt = A.RampaCt + A.RampaVel;
    constante = 6 * 7 * M.Efpi * (A.RampaCt/60000) / M.N;
    Vrampa(M.qt,:) = constante * (M.N : -1 : 1);
end

if (A.cont > A.RampaFim & A.RampaCt > 0)
    A.RampaCt = A.RampaCt - A.RampaVel;
    constante = 6 * 7 * M.Efpi * (A.RampaCt/60000) / M.N;
    Vrampa(M.qt,:) = constante * (M.N : -1 : 1);
end
end

```


3.2. ACOPLAMENTO ENTRE CÉLULAS

Os itens anteriores descrevem a concepção e o funcionamento de uma célula básica, com 2 (dois) pontos quânticos acoplados e um elétron em excesso, modelada numericamente em 1 (uma) dimensão. Foram descritos os processos dissipativos [19][20][21] dentro desta célula que permitem ao elétron sair do estado inicial arbitrário, coerente com a estrutura e fisicamente consistente, e, em função da ação do campo de desaceleração, atingir os estados fundamentais disponíveis em função das condições de contorno.

De posse deste aparato numérico foi dado um novo passo em direção ao entendimento de circuitos mais complexos e transmissão de informações pela associação entre células. O ponto de partida adotado foi o acoplamento entre essas células, de forma lateral, pelo uso da forma integral da equação de Poisson. A figura 3-5 demonstra a intenção básica pretendida deste estudo.

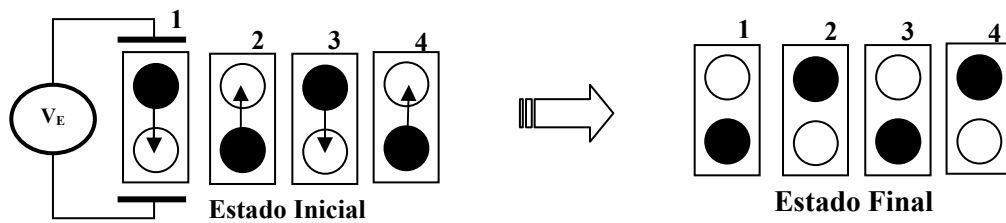


Figura 3-5 – 4 Células Acopladas Lateralmente

3.2.1. O potencial de um sistema de carga: campo conservativo

O potencial em um ponto do espaço pode ser definido como o trabalho realizado em carregar uma carga unitária positiva da referência zero até o ponto [4]. Assim, o potencial de uma carga pontual única, identificada por Q' , localizada em \mathbf{r}' , envolve a distancia $|\vec{r} - \vec{r}'|$ de Q' ao ponto definido por \mathbf{r} , no qual se estabelece o valor potencial. Para uma referência zero no infinito temos, na equação 3-11:

$$V(r) = \frac{Q'}{4\pi\epsilon|r - r'|} \quad (3-11)$$

Se forem continuamente adicionadas cargas pontuais e representando cada carga por um pequeno elemento de distribuição contínua de cargas volumétricas $\rho\Delta v$, temos:

$$V(r) = \frac{\rho(r_1)\Delta v_1}{4\pi\epsilon|r-r_1|} + \frac{\rho(r_2)\Delta v_2}{4\pi\epsilon|r-r_2|} + \dots + \frac{\rho(r_n)\Delta v_n}{4\pi\epsilon|r-r_n|} \quad (3-12)$$

Em permitindo o número de elementos n infinito, temos portando:

$$V(\vec{r}) = \int_{Volume} \frac{\rho(\vec{r}')d v'}{4\pi\epsilon|\vec{r}-\vec{r}'|} \quad (3-13)$$

Sendo o potencial $V(\mathbf{r})$ determinado em relação a uma referência zero de potencial situada no infinito, e é a medida exata do trabalho realizado em trazer uma carga unitária do infinito ao ponto definido por \mathbf{r} em que se está determinando o potencial. A densidade volumétrica de carga $\rho(\mathbf{r}')$ e o elemento diferencial de volume $d\mathbf{v}'$ se combinam para representar uma quantidade diferencial de carga $\rho(\mathbf{r}') d\mathbf{v}'$, localizada em \mathbf{r}' . A distância $|\vec{r}-\vec{r}'|$ é a distância do ponto da fonte (carga) ao ponto do potencial. A integral é uma integral múltipla (de volume).

Portanto, através da equação 3-12, que é uma forma integral da equação de Poisson ($\nabla \cdot \nabla V = -\rho/\epsilon$) para potenciais eletrostáticos em função de uma distribuição de cargas, é possível realizar o acoplamento entre as células pela inclusão no potencial de confinamento presente na equação de Schrödinger (equação 3-5).

3.2.2. Método numérico – Modelo 1D

Para realizar uma simulação numérica de um processo físico contínuo é necessário discretizar o problema de forma a permitir a construção do algoritmo numérico. O modelo de célula que está sendo simulado não foge a esta orientação. Sendo assim, a figura 3-6 demonstra, em linhas gerais, como foi discretizada cada célula de pontos quânticos.

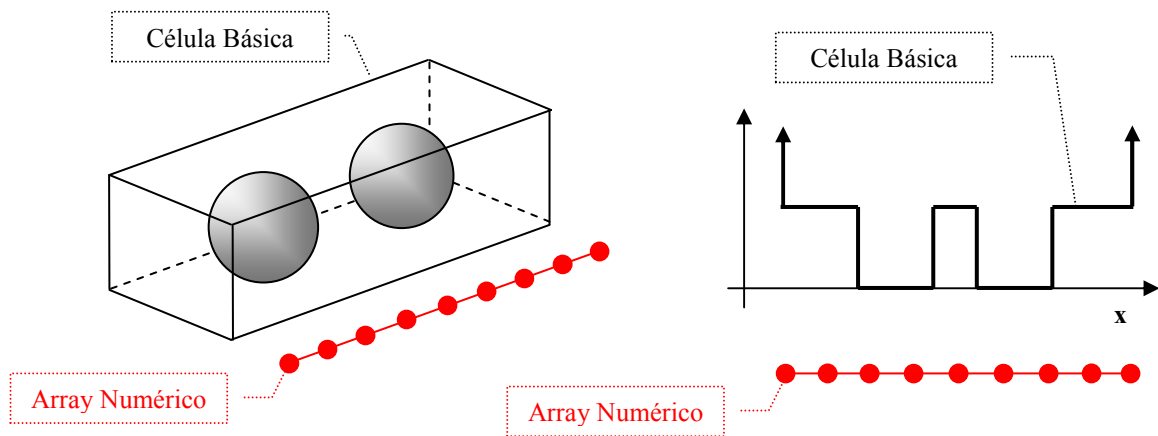


Figura 3-6 – Representação do Array Numérico usado nas simulações

Sendo assim, cada célula na simulação numérica é representada por arrays e, considerando suas dimensões físicas, podem ser representados como na figura 3-7. Nesta figura também pode ser visto o relacionamento entre as células pelo potencial visto de cada elemento do array de uma célula em função da “distribuição de cargas” na célula vizinha.

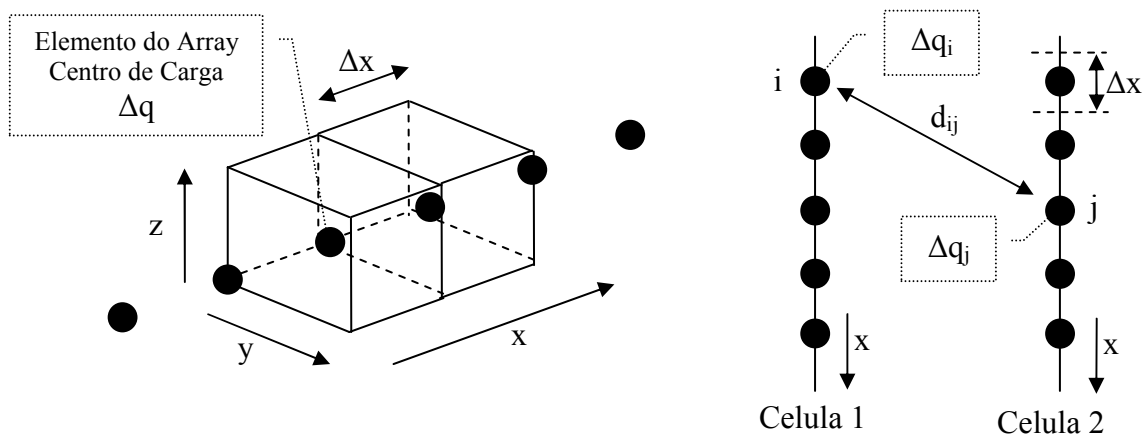


Figura 3-7 – Diagrama de acoplamento por Poisson

Portanto, do diagrama da figura 3-7, uma maneira de se obter a forma integral da equação de Poisson (equação 3-12) passa a ser utilizada, de forma discreta, como na equação 3-14.

$$V_j \approx \sum_{\forall i} V_{ij} \approx \sum_{\forall i} \frac{\Delta q_i}{4\pi\epsilon \cdot d_{ij}} \quad (3-14)$$

3.2.3. Implementação do acoplamento pelo método numérico

Partindo da formulação apresentada no item 3.2.2 chegamos à implementação numérica como se segue.

A Matriz de Acoplamento entre as células (equação 3-14) pode ser dividida em uma parte estática, dependente apenas da geometria do problema (distância entre as células, quantidade de elementos de cada array) e uma parte dinâmica, em função da distribuição de cargas ao longo da célula durante a simulação. A parte estática é preparada, no início da simulação, criando-se uma matriz **C.MAC** como se segue:

```
% -----  
% Monta Matriz de Acoplamento entre as Celulas  
% -----  
  
function Matriz_Acoplamento  
  
    global A C M E;  
    global Vtotal VautoC Vpoco;  
    global EX EY EJ;  
  
    % -----  
    % Modelo - Carga Distribuida  
    % -----  
  
    if (A.cap == 3)  
  
        for m = 1:M.qt  
            for f = 1:M.qt  
                for x = 1:M.N  
                    for y = 1:M.N  
  
                        if (m == f)  
                            C.MAC (x,y,m,f) = 0;  
                        else  
                            dist = (M.L / 2) * abs (m-f) / 3; % Distancia entre os centros das celulas  
                            C.MAC (x,y,m,f) = 1 / (4 * pi * C.e0 * sqrt (dist^2 + (abs(x-y)*M.dx)^2));  
                        end  
  
                    end  
  
                end  
  
            end  
  
        end  
  
    end
```

Desta forma chega-se à implementação final do acoplamento entre células que é realizado a cada interação durante o processo de simulação:

```

% -----
% Calcula Auto Capacitancia
% -----

function Calcula_AutoC

    global A C M E;
    global Vtotal VautoC Vpoco;
    global EX EY EJ;

    % -----
    % Carga Distribuida
    % -----

    if (A.cap == 3)

        mod_a = (EX.^2 + EY.^2);
        for m = 1:M.qt
            mod_b(m,:) = mod_a(m,:) ./ (sum(mod_a(m,:)) * M.dx);
        end

        q = C.q^2 * mod_b' * M.dx / 30;

        % Registra carga nos pocos
        if (rem (A.cont , A.max) == 0 & A.cont > 0)
            E.CargaEsquerda (:,A.cont/A.max + 1) = sum(q(M.P1.ini:M.P1.fim,:))';
            E.CargaDireita (:,A.cont/A.max + 1) = sum(q(M.P2.ini:M.P2.fim,:))';
        end

        % Realiza o acoplamento
        VautoC = zeros(M.qt,M.N);

        for m = 1:M.qt
            for f = 1:M.qt
                Vmf = C.MAC(:, :, m, f) * q(:, f);
                VautoC (m, :) = VautoC (m, :) + Vmf';
            end
        end

        % Nivel a potencial nos poços por serem metálicos
        for m = 1:M.qt
            VautoC (m,M.P1.ini:M.P1.fim) = sum (VautoC (m,M.P1.ini:M.P1.fim)) / (M.P1.fim -
M.P1.ini);
            VautoC (m,M.P2.ini:M.P2.fim) = sum (VautoC (m,M.P2.ini:M.P2.fim)) / (M.P2.fim -
M.P2.ini);
        end
    end
end

```

4. RESULTADOS E DISCUSSÃO

Neste capítulo serão apresentados alguns resultados de simulações para o acoplamento lateral de células de pontos quânticos pela modelagem de diferentes estruturas funcionais. Primeiramente, com o intuito de avaliar a consistência dos algoritmos com os princípios físicos fundamentais, duas células, sem os poços quânticos e acopladas lateralmente, foram modeladas a fim de avaliar a repulsão dos elétrons. Em seguida, um conjunto com quatro células acopladas foi simulado para verificar o comportamento básico encontrado na literatura [5][19][20] de transporte de informações nesse tipo de estrutura. Por fim, a simulação de um conjunto com seis células é utilizada para verificar a dependência do comportamento em função da geometria e demais características físicas.

4.1. DESCRIÇÃO DA CÉLULA BÁSICA

Como descrito ao longo do capítulo 3 – Desenvolvimento, este projeto simula o comportamento do elétron excedente em células com as características geométricas e dimensões da figura 4.1, acopladas lateralmente.

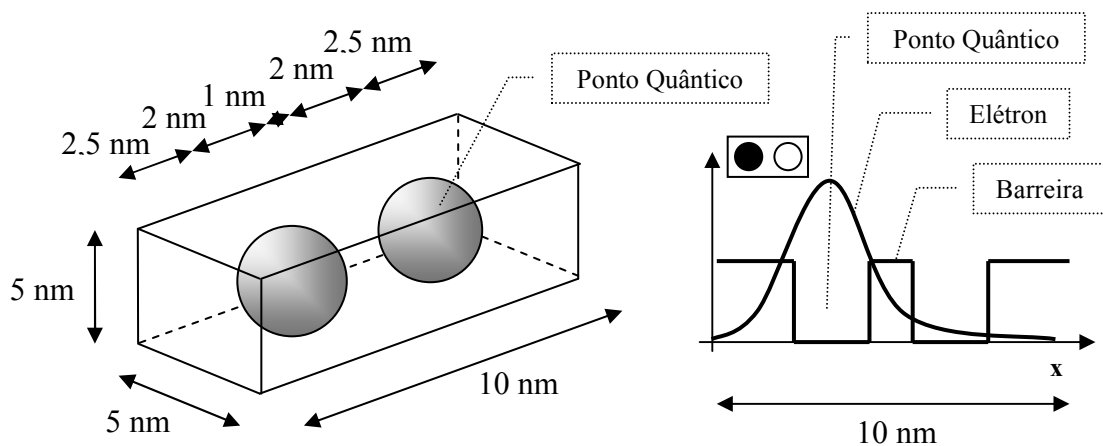


Figura 4-1 – Célula básica utilizada nas simulações (modelo físico e numérico)

A simulação numérica foi realizada utilizando o incremento de tempo de $1,0940 \times 10^{-16}$ segundos. Este valor foi escolhido em respeito ao critério de estabilidade: $|\alpha \cdot \Delta t / \Delta x^2| < 1/2$. Onde $\alpha = \hbar/2m$ [22].

As demais constantes físicas utilizadas nas simulações foram [22]:

- Massa efetiva do elétron $\rightarrow 9,110 \times 10^{-31}$ kg
- Carga fundamental do Elétron $\rightarrow 1,602 \times 10^{-19}$ C
- Constante de Plank $\rightarrow 6,626 \times 10^{-34}$ J.s
- Velocidade da Luz $\rightarrow 2,99792458 \times 10^8$ m/s
- Resistividade total $\rightarrow 0,5 \times 10^{12}$ ohms
- Permeabilidade magnética do vácuo $\rightarrow 1,2566370614 \times 10^{-6}$ H/m
- Permissividade elétrica do vácuo $\rightarrow 8,854187817 \times 10^{-12}$ F/m

O projeto visa avaliar a propagação da transição de estados entre células acopladas como fundamento para a construção de circuitos mais complexos com essa tecnologia. A figura 4.2 abaixo apresenta uma configuração com 4 células acopladas como exemplo.

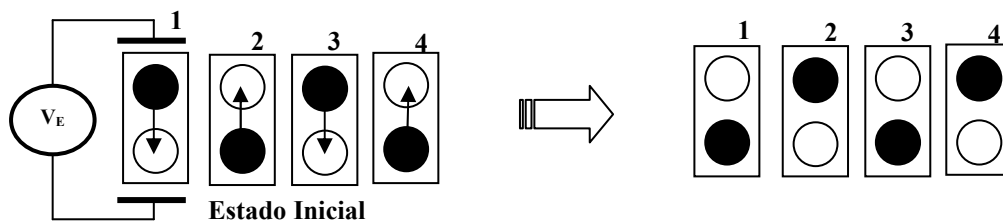


Figura 4-2 – Simulação com 4 Células – Diagrama Esquemático

A fonte de tensão V aplica uma diferença de potencial à célula 1 que provoca o tunelamento do elétron de um poço quântico para o outro, dentro desta célula. Essa redistribuição de cargas, influencia a célula 2 que por sua vez influencia a célula 3 e assim por diante.

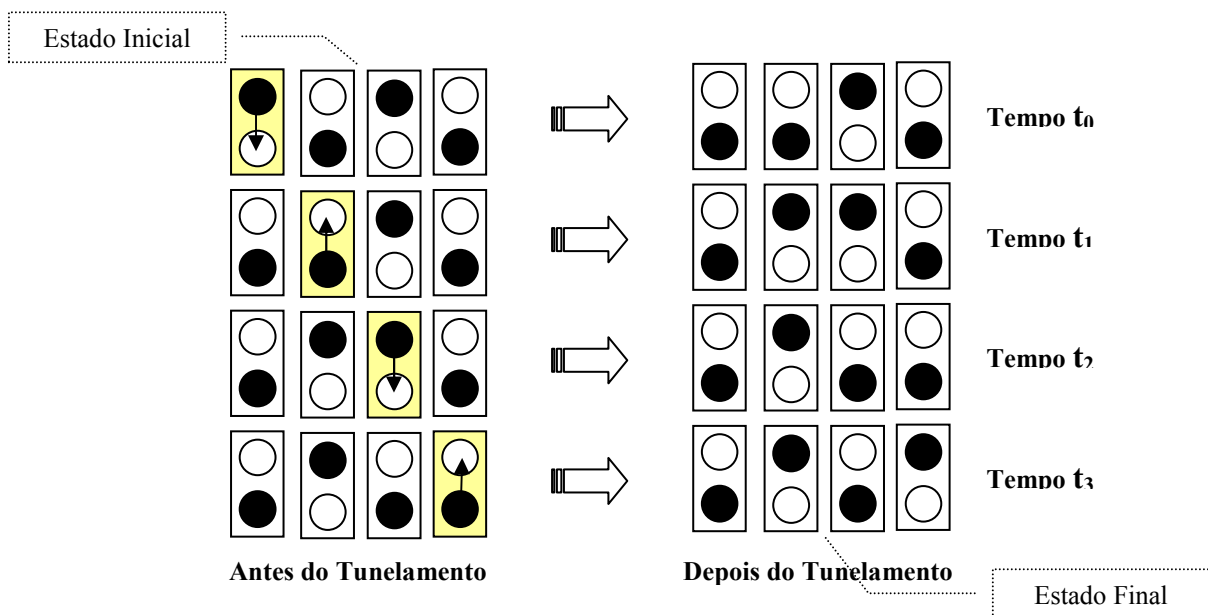


Figura 4-3 – Dinâmica dos Tunelamentos – Diagrama Esquemático

A seguir é apresentado um diagrama com a variação no tempo da tensão imposta à célula 1 pela fonte V (figura 4.2) e como esta tensão se distribui pela célula no modelo.

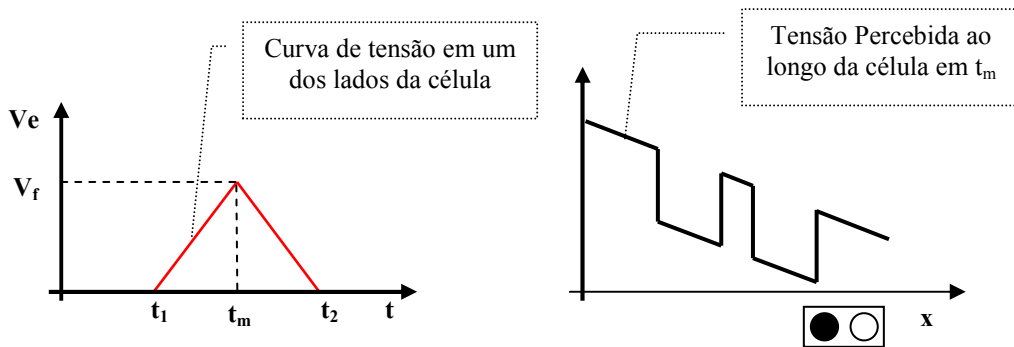


Figura 4-4 – Variação temporal de V_E e seu efeito na célula 1 em t_m

Onde " V_e " demonstra a diferença de potencial aplicada à célula 1 ao longo do tempo " t ". Os tempos " t_1 ", " t_2 " e " t_m " foram escolhidos com o intuito de permitir que a variação de tensão na célula 1 fosse suave o bastante para não impor uma aceleração excessiva ao elétron evitando que ele oscilasse entre os poços – baixa transferência de energia cinética. A tensão " V_f " foi dimensionada de forma a garantir o efeito de tunelamento. Esses valores foram escolhidos empiricamente pela realização de simulações e avaliação de resultados.

Um fator a ser considerado é o espaçamento entre as células. A carga nas células não está concentrada em seu centro geométrico, portanto, para uma melhor representação dos efeitos, foi feita uma redução da distância efetiva entre elas como apresentado na figura 4-5 abaixo:

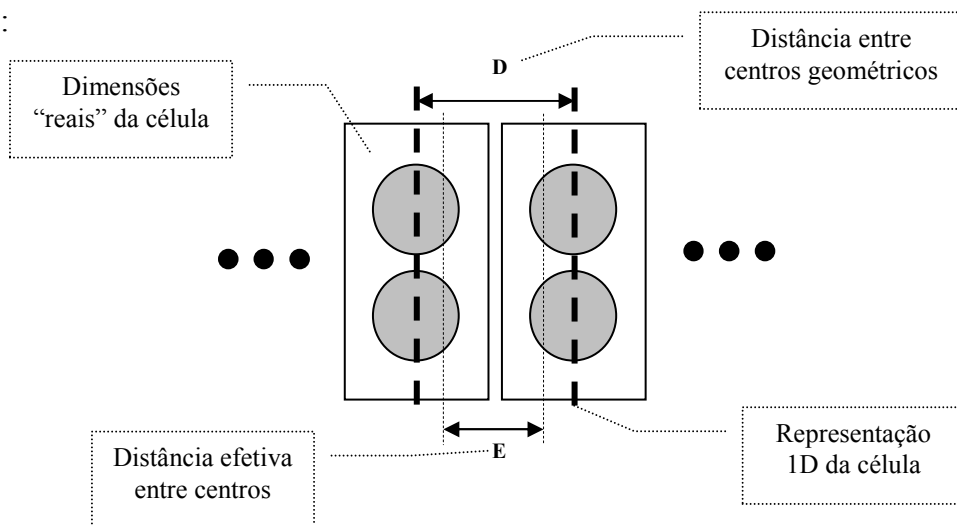


Figura 4-5 – Distância entre células acopladas

Cada célula é composta por poços quânticos com dimensões finitas. Sendo assim, sua influência sobre as células adjacentes não deve ser considerada apenas a partir dos centros geométricos de cada estrutura como demonstra a figura 4.5 (pontilhado maior), pois há uma distribuição de cargas ao longo de todas as dimensões dos poços quânticos. Para compensar esta característica do modelo, que representa cada célula através de um vetor em apenas 1 (uma) dimensão, no cálculo da distância (D) entre células, para efeitos de acoplamento, foi feita uma redução desta distância para um novo valor (E) (fig. 4.5 – pontilhado menor) que pode ser considerada como a “distância efetiva” entre células ($E = kD$, onde $0 > k > 1$).

4.2. CENÁRIO COM 2 ELÉTRONS LIVRES

Tendo as características descritas no item anterior (4.1) como padrões para as simulações, um primeiro modelo foi construído para avaliar a repulsão dos elétrons livres nas células.

Como premissas, o modelo usado considera que cada elétron está contido em sua célula e que a transmissão de informações entre células ocorre pelo acoplamento feito através da forma integral da equação de Poisson (item 3.2). Portanto, uma primeira avaliação se fez no estudo do comportamento dos elétrons em duas células acopladas lateralmente, sem a presença dos poços quânticos. A figura 4.6 é uma variação das figuras 4.1 e 4.2 com a exclusão dos poços. A partir deste modelo, ao realizar o acoplamento, espera-se que a ação de V_e sobre o elétron da célula 1 altere a sua posição e, com isso, que a posição do elétron na célula 2 seja, em consequência, alterada. Quer-se verificar aqui se tal evento é percebido na simulação.

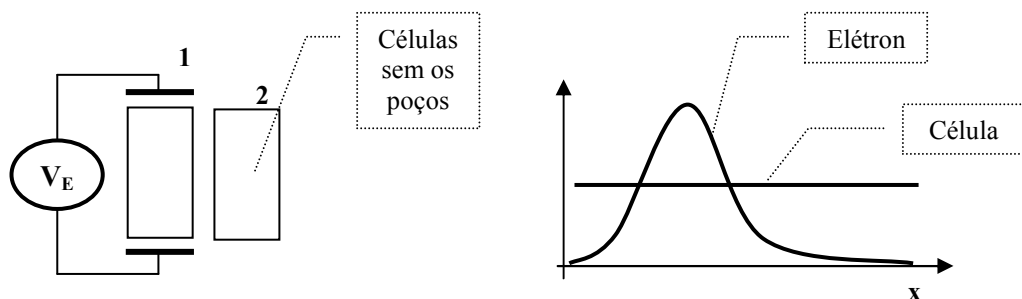


Figura 4-6 – Células sem poços quânticos – “Elétron Livre”

Para que fosse possível uma melhor avaliação do efeito de repulsão, no instante $t = 0$, foram arbitrados os mesmos valores de Ψ para as duas células. A figura 4.7 apresenta este estado inicial. As curvas em vermelho representam os elétrons pelas suas distribuições de carga e as curvas em azul, a influência de seus potenciais na célula adjacente.

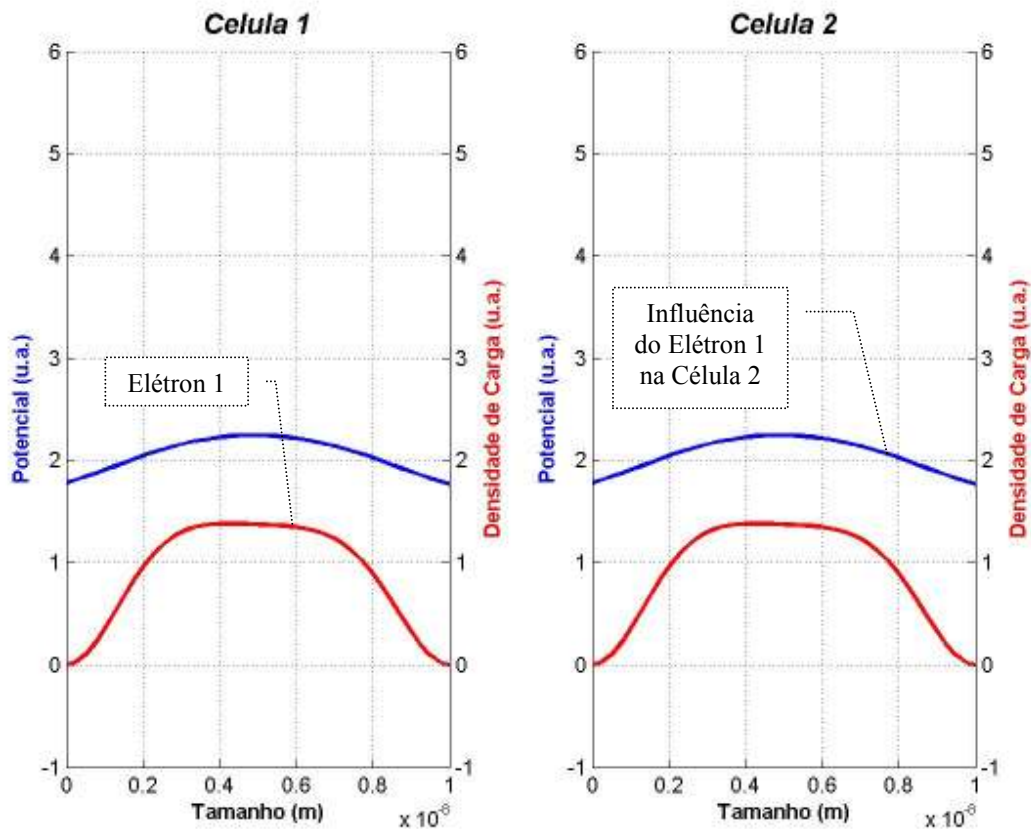


Figura 4-7 – Simulação com 2 células – Elétrons Livres – Estado Inicial

A partir deste estado, no instante $t = 6,5 \times 10^{-12}$ segundos (momento arbitrário após percepção visual da existência de estabilidade), teve início a atuação da rampa sobre a célula 1 do sistema. Com a introdução deste distúrbio em uma das células, a distribuição de cargas no sistema deixou de ser simétrica permitindo a influência de um elétron sobre o outro, retirando-os do estado de equilíbrio instável. Desta forma, o sistema tende a um novo estado de equilíbrio pela dissipação de energia (item 3.1.2) e efeitos de acoplamento.

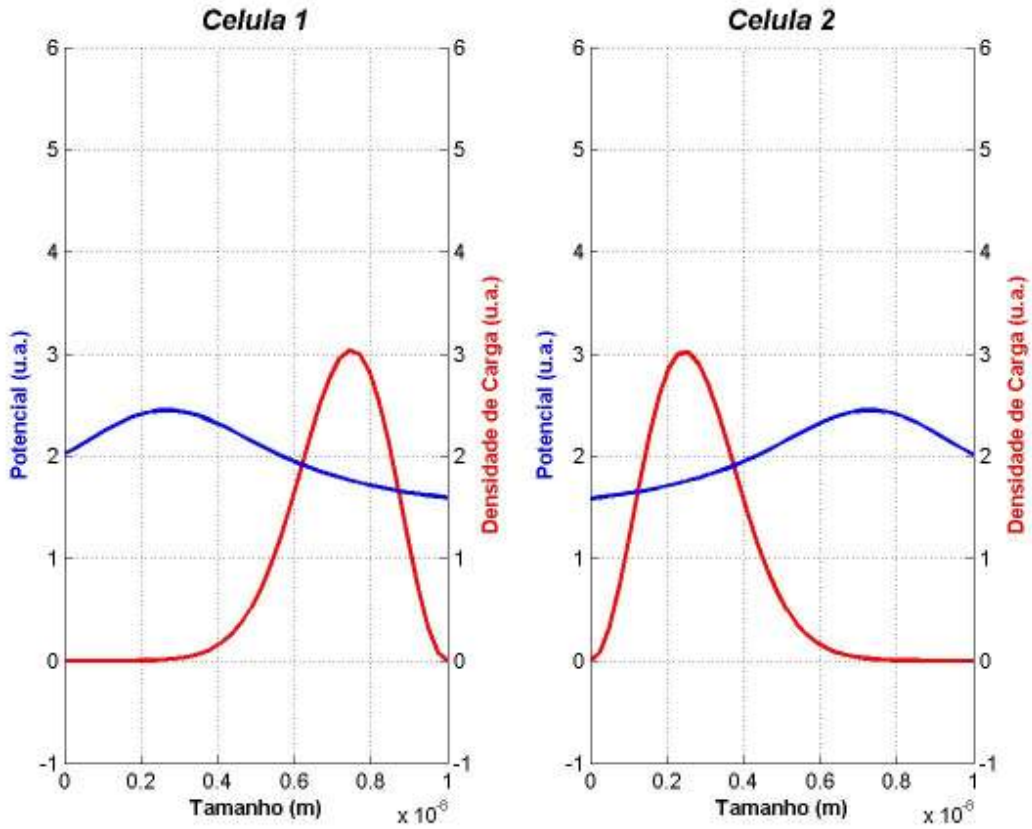


Figura 4-8 – Simulação com 2 células – Elétrons Livres – Estado Final

A figura 4.8 apresenta o novo estado de equilíbrio: na célula 1 o elétron se estabilizou no lado “direito” e na célula 2, no lado “esquerdo” (vale lembrar que as células estão posicionadas como mostrado na figura 4.9).

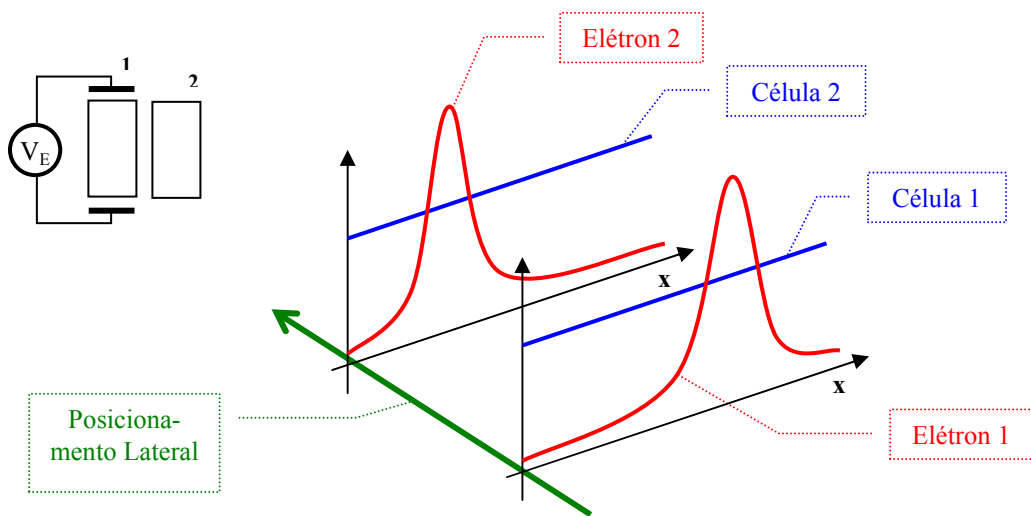


Figura 4-9 – Posicionamento lateral das células

As figuras 4.10 e 4.11 apresentam a dinâmica do processo de tunelamento com um recorte no tempo para ressaltar o fenômeno (momento da transição).

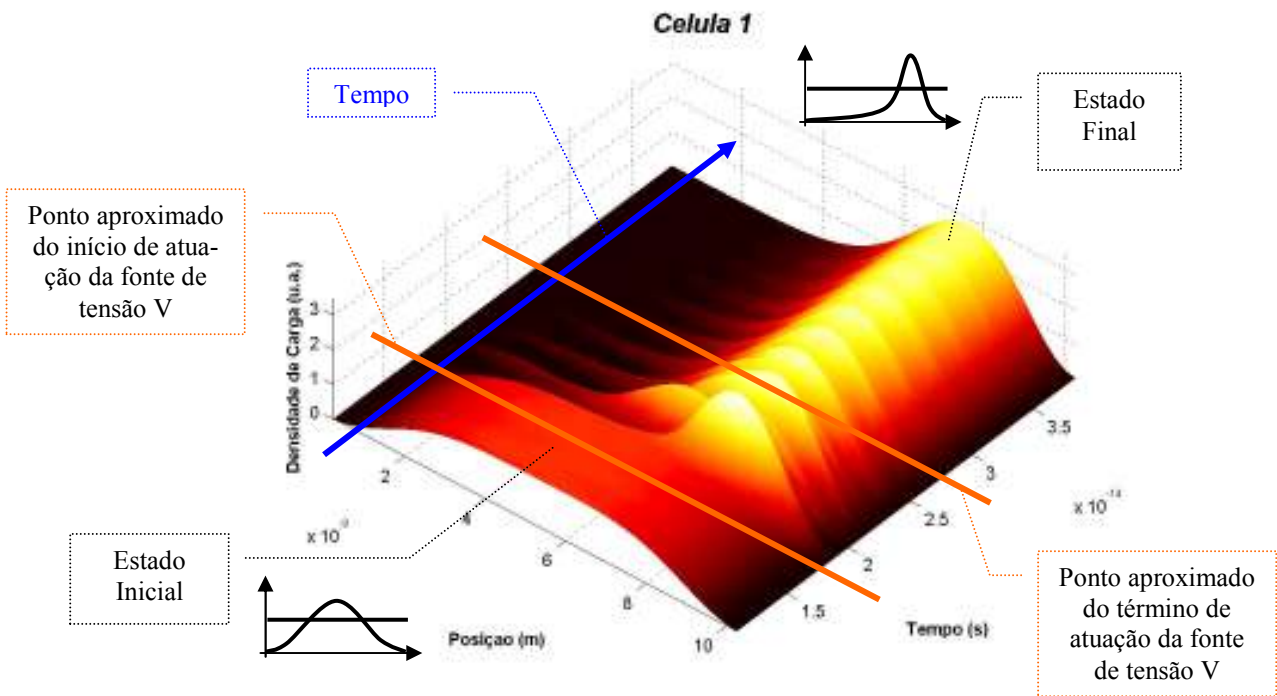


Figura 4-10 – Simulação com 2 células – Dinâmica da Célula 1

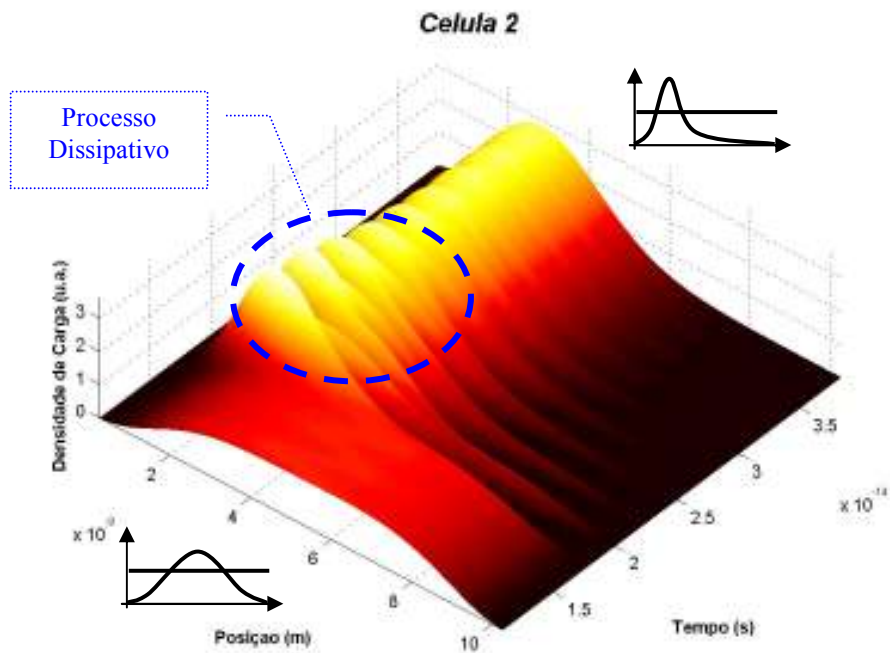


Figura 4-11 – Simulação com 2 células – Dinâmica da Célula 2

A figura 4.10 corresponde à primeira célula onde há a atuação da fonte de tensão V . Quando esta fonte inicia sua atuação, há uma evolução do estado inicial do sistema levando-o a uma nova situação de equilíbrio. O processo dissipativo, que ocorre em todas as células, é responsável pela retirada a energia cinética do sistema permitindo a transição para o novo estado estacionário. A figura 4.11 representa o segundo elétron que recebe a influência do primeiro em função da interação entre eles. Como este acoplamento ocorre nos dois sentidos, os elétrons tendem a se repelir atingindo uma nova configuração.

A figura 4.12 apresenta a variação da distribuição de carga nas células em função do tempo. Pode ser observado que no mesmo instante em que o elétron da célula 1 ocupa o lado direito da célula, o elétron da célula 2 também se movimenta só que em direção oposta.

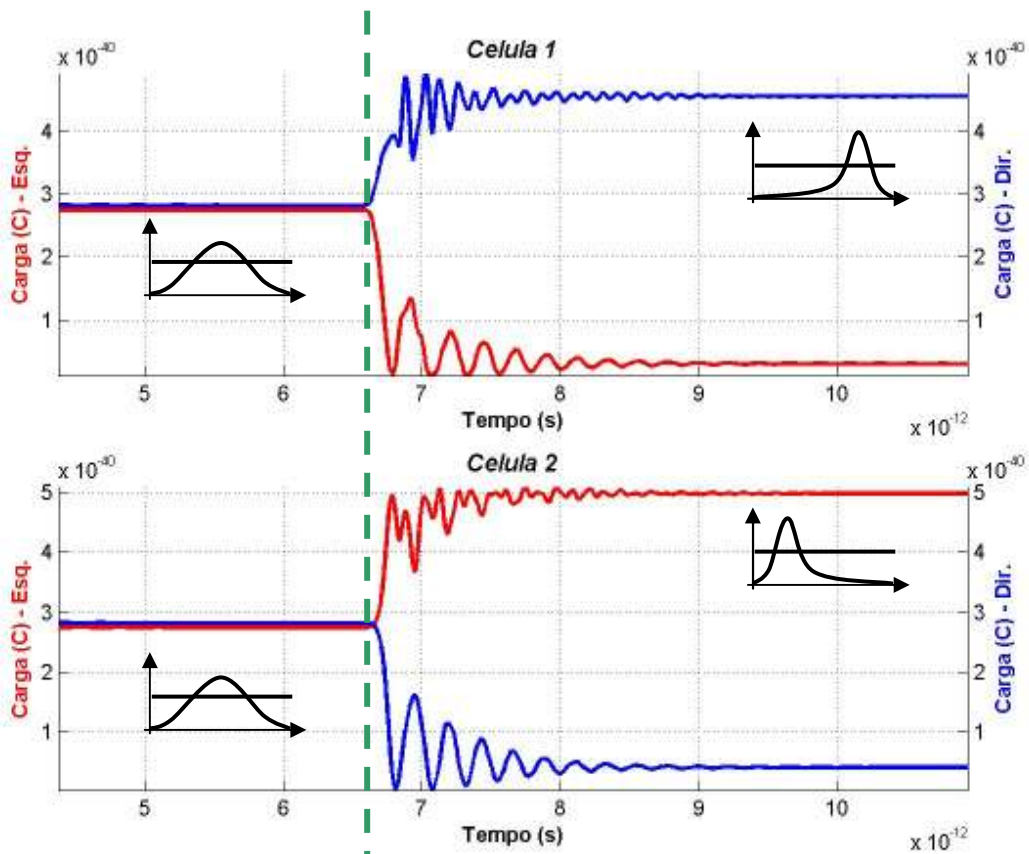


Figura 4-12 – Simulação com 2 células – Distribuição de carga nas células

A figura 4.13 apresenta a corrente medida no centro das células em função do tempo. Como referência, valores positivos representam corrente fluindo da esquerda para a direita.

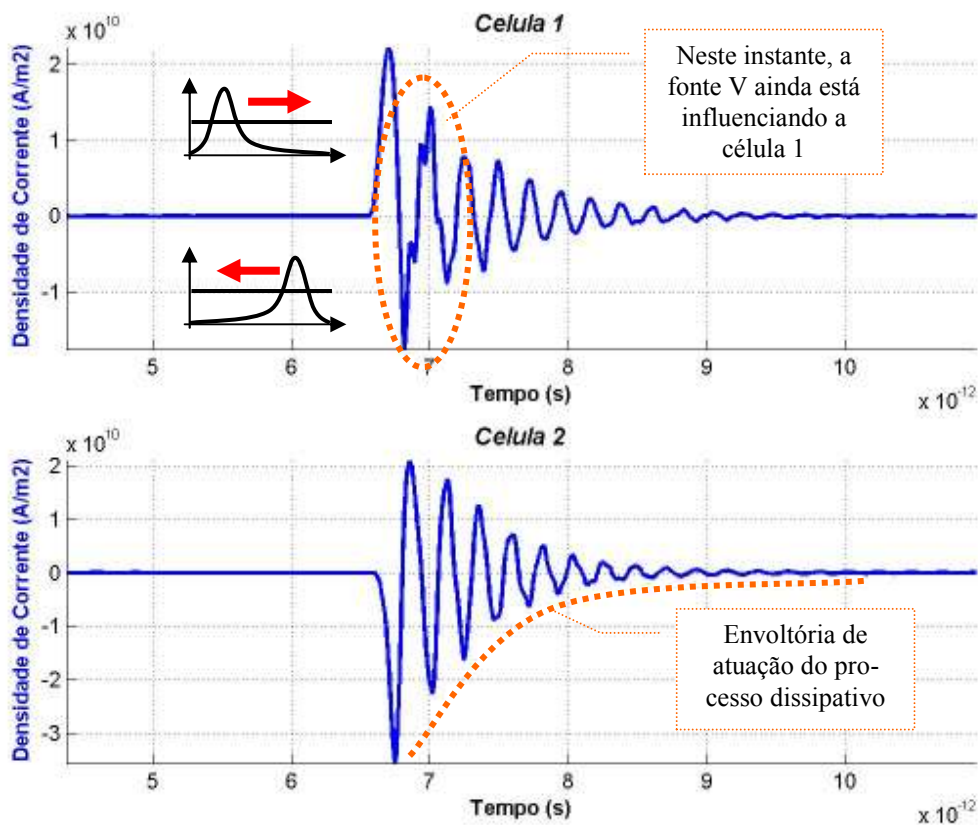


Figura 4-13 – Simulação com 2 células – Corrente no centro das células

Na figura 4.13 é possível observar a envoltória do processo dissipativo levando os elétrons aos seus estados finais de equilíbrio estático, de menor energia. A oscilação evidenciada na célula 1 não se deve, em princípio, a um artefato de simulação, mas à natureza quântica do elétron.

4.3. CENÁRIO COM 4 CÉLULAS ACOPLADAS

A proposta inicial, feita por Lent *et al* [3], de criação de autômatos com o uso de QCA's, faz uso de células com 4 pontos quânticos como na figura 4.14.a. Algumas variações de QCA's introduzem pontos extras nos centros das células (4.14.b) com a finalidade de facilitar ou controlar o processo de tunelamento. Este trabalho segue o modelo da figura 4.14.c.

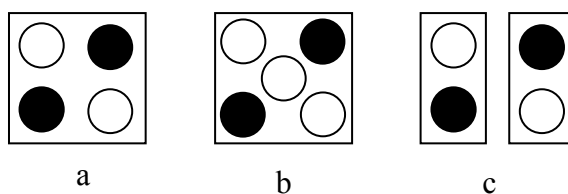


Figura 4-14 – Células em Autômatos com QCA's

Como o norteador deste projeto é a análise da dinâmica de células de pontos quânticos acopladas, um cenário foi construído para a avaliação da propagação da informação entre células. Para estudar o comportamento de estruturas mais complexas, foram simulados QCA's com um número crescente de células. Neste caso, foram usadas 4 células acopladas. Neste cenário, um fator de maior relevância a ser observado é o atraso na propagação da informação de um lado ao outro da estrutura. A figura 4.15 apresenta o diagrama esquemático deste cenário.

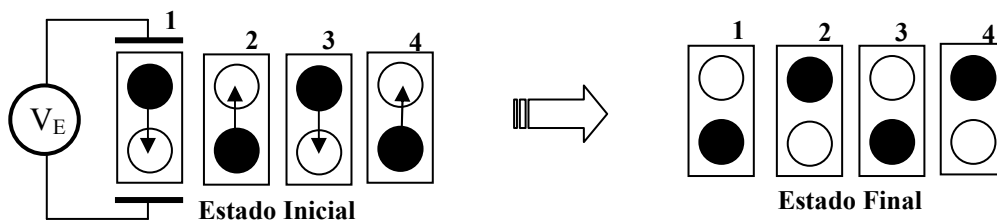


Figura 4-15 – Simulação com 4 Células – Diagrama Esquemático

As considerações para a aplicação da excitação na cadeia da figura 4-15 foram análogas às adotadas na simulação do item anterior (4.2).

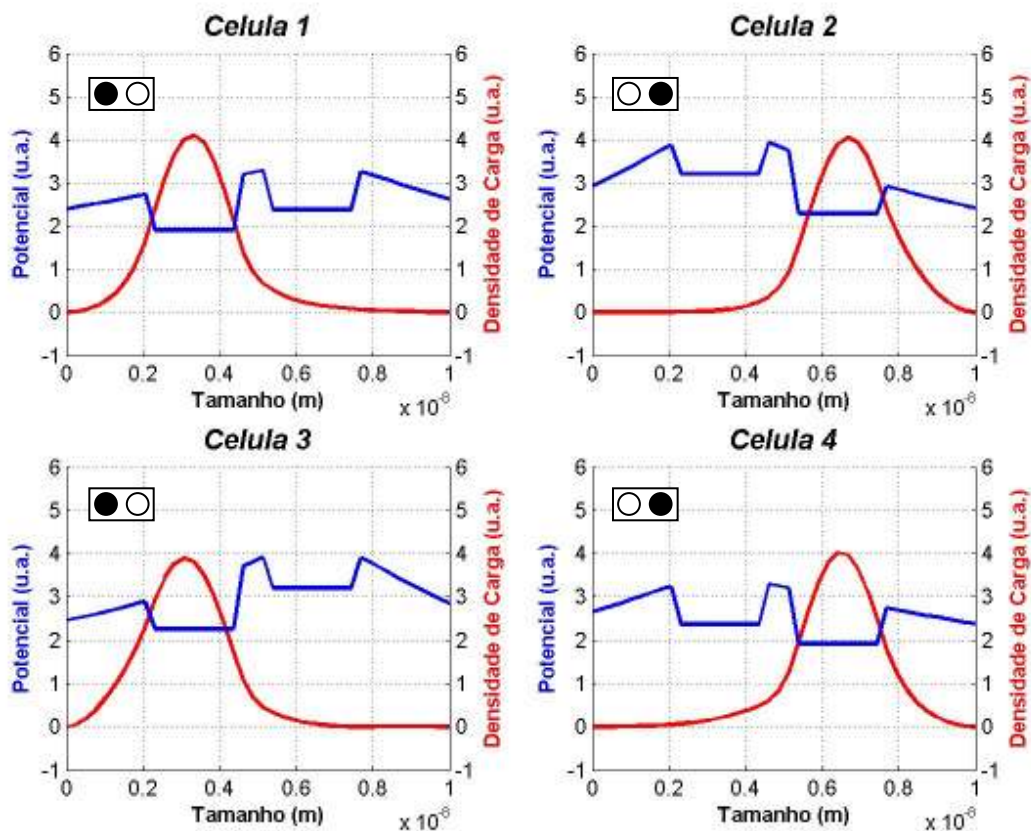


Figura 4-16 – Simulação com 4 Células – Estado Inicial

Após o início da simulação numérica, foi alcançado um estado estacionário onde cada elétron concentrou-se em um poço de cada célula, respectivamente. A partir deste estado, no instante $t = 5,0 \times 10^{-12}$ segundos (momento arbitrário após percepção visual da existência de estabilidade), teve início a atuação da rampa sobre a célula 1 do sistema seguindo a mesma rotina de simulação do cenário anterior. A figura 4.17 apresenta o sistema em sua nova posição de equilíbrio, após o tunelamento nas células. Pode ser observado o novo posicionamento dos elétrons.

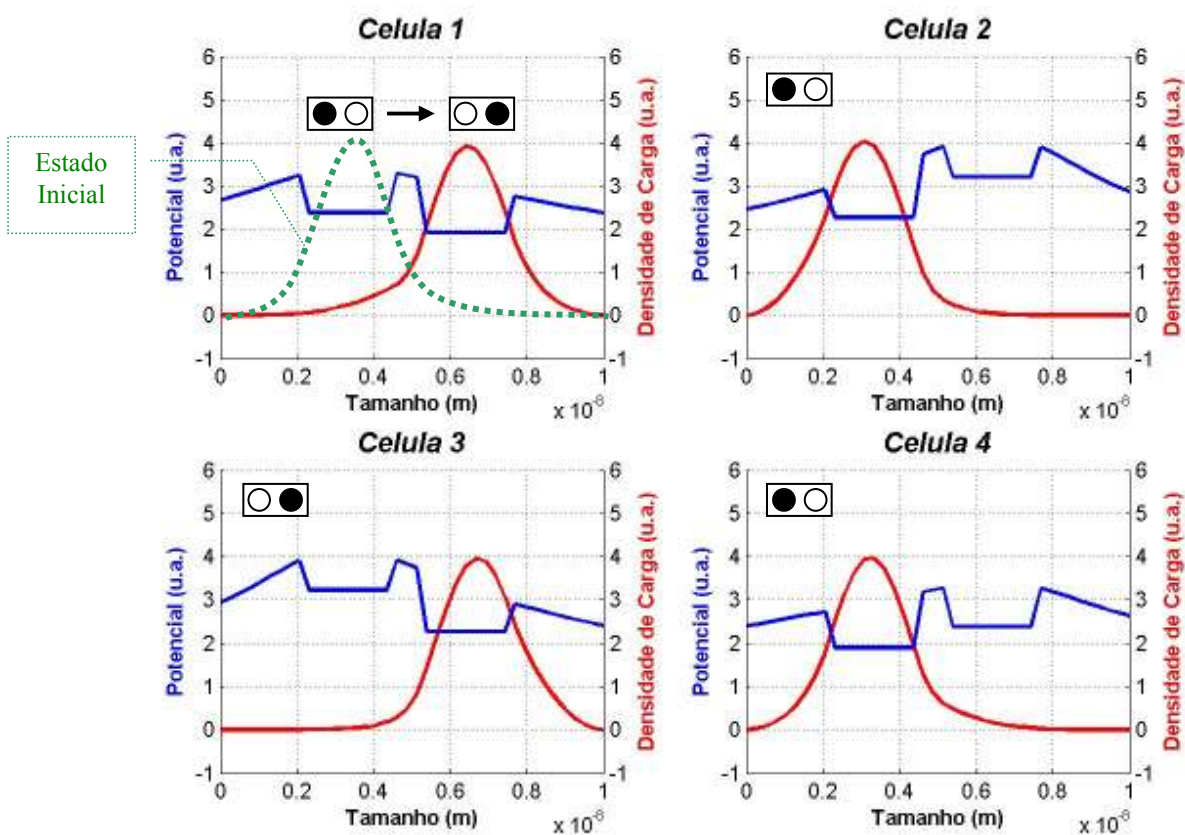


Figura 4-17 – Simulação com 4 Células – Estado Final

Neste modelo, cada célula tem o seu comportamento influenciado por todas as demais. As figuras 4.18, 4.19, 4.20 e 4.21 apresentam a dinâmica do processo de tunelamento com um maior detalhamento para cada célula.

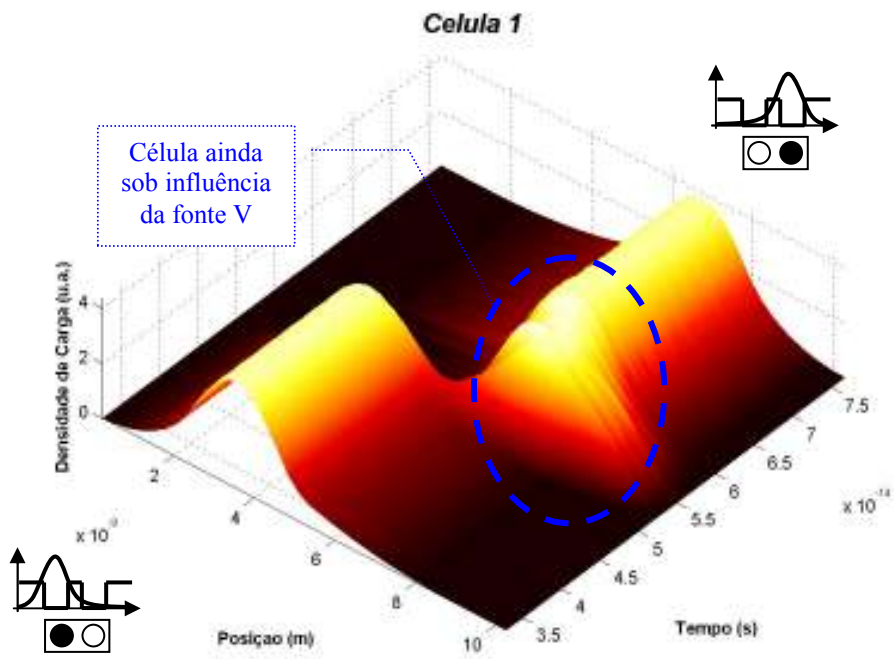


Figura 4-18 – Simulação com 4 células – Dinâmica da Célula 1

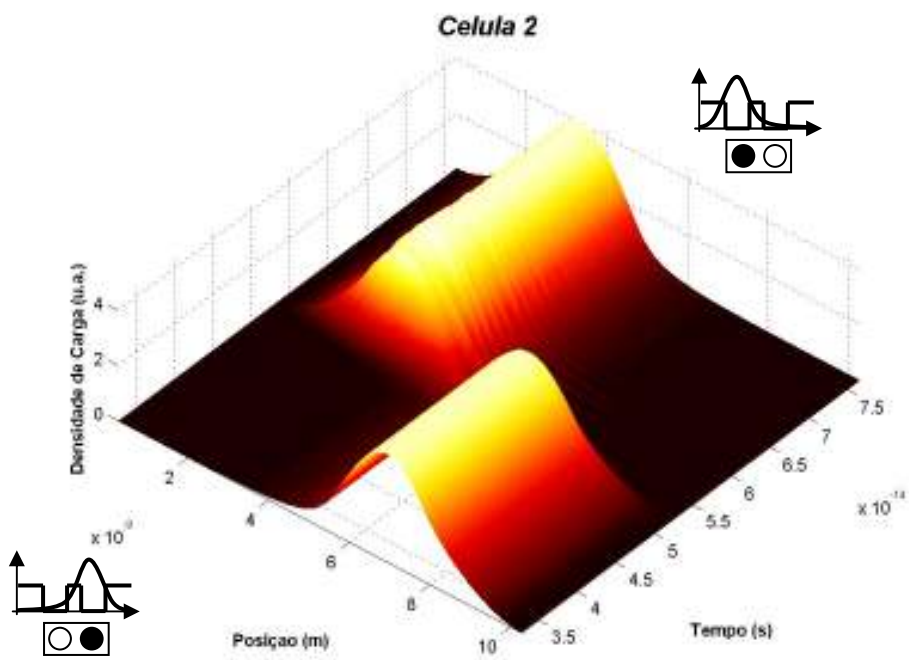


Figura 4-19 – Simulação com 4 células – Dinâmica da Célula 2

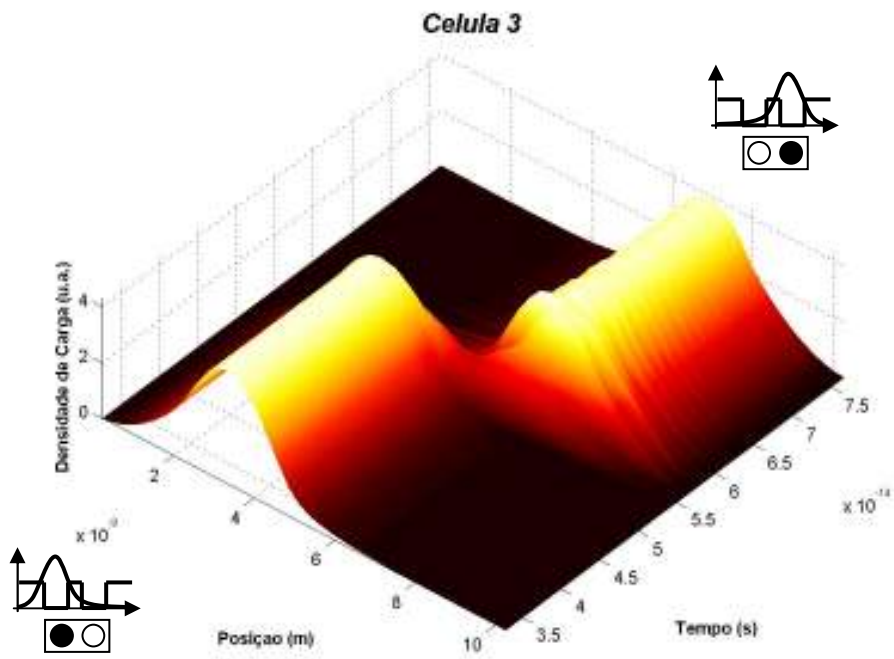


Figura 4-20 – Simulação com 4 células – Dinâmica da Célula 3

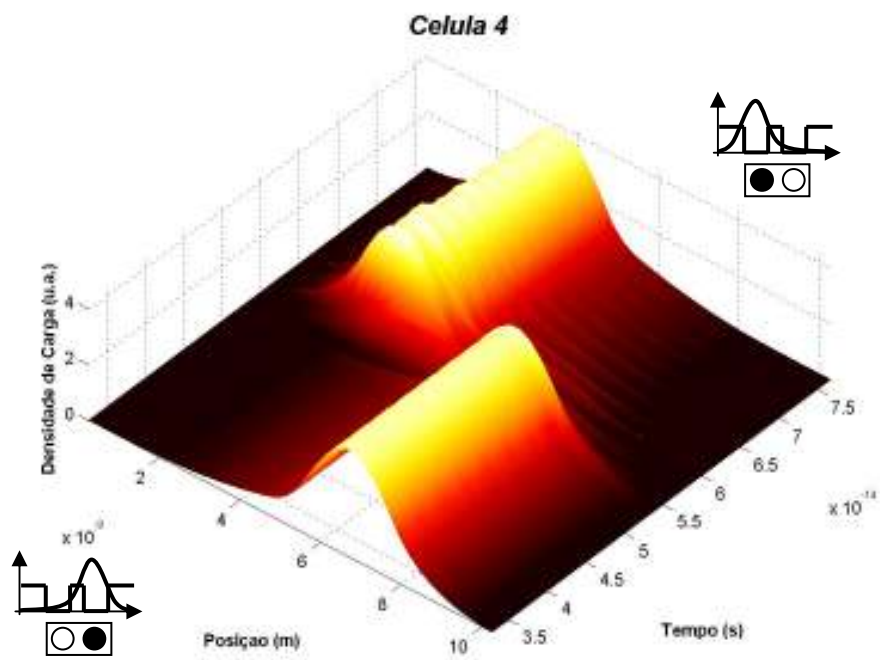


Figura 4-21 – Simulação com 4 células – Dinâmica da Célula 4

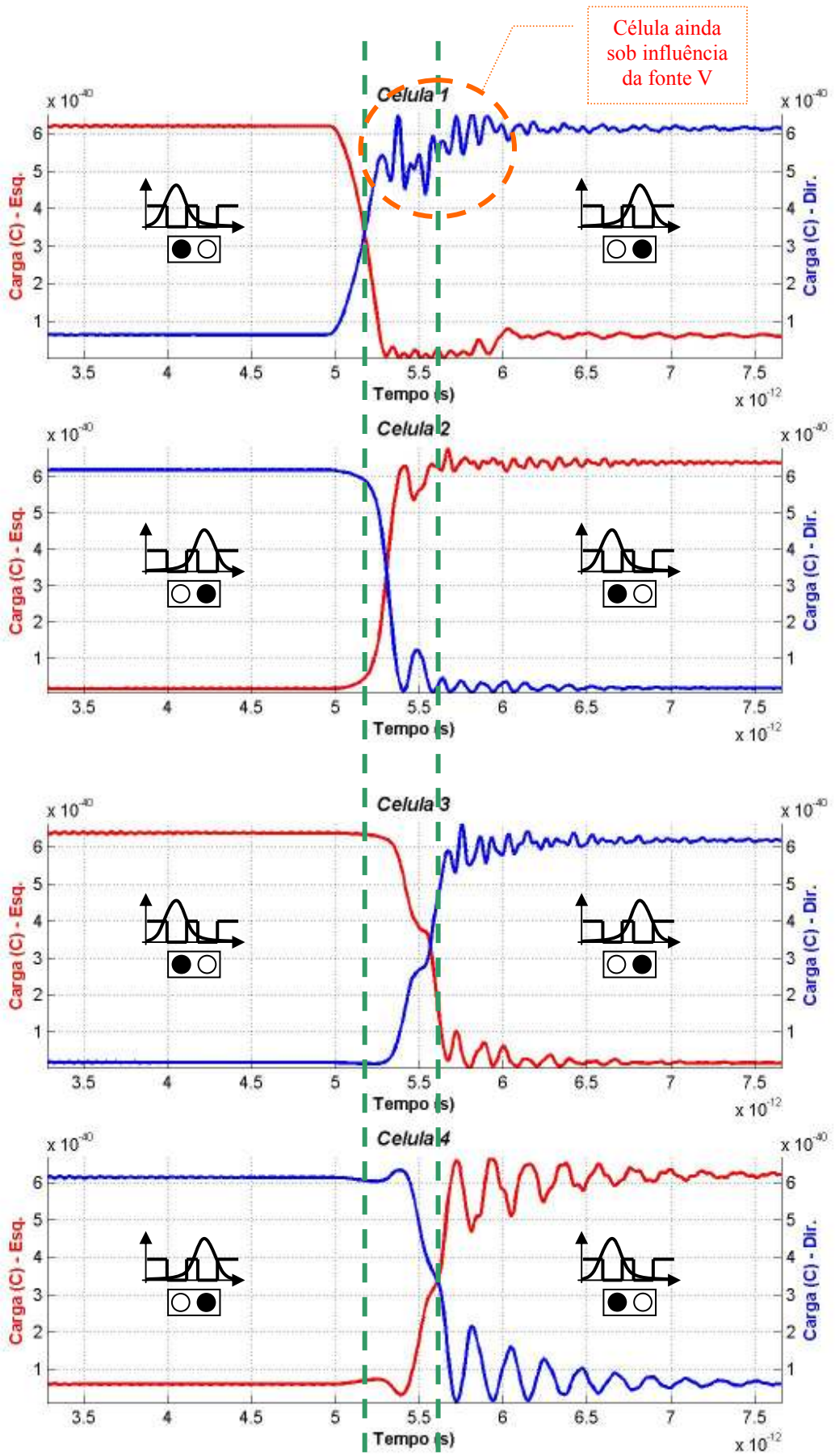


Figura 4-22 – Simulação com 4 células – Carga nos Poços

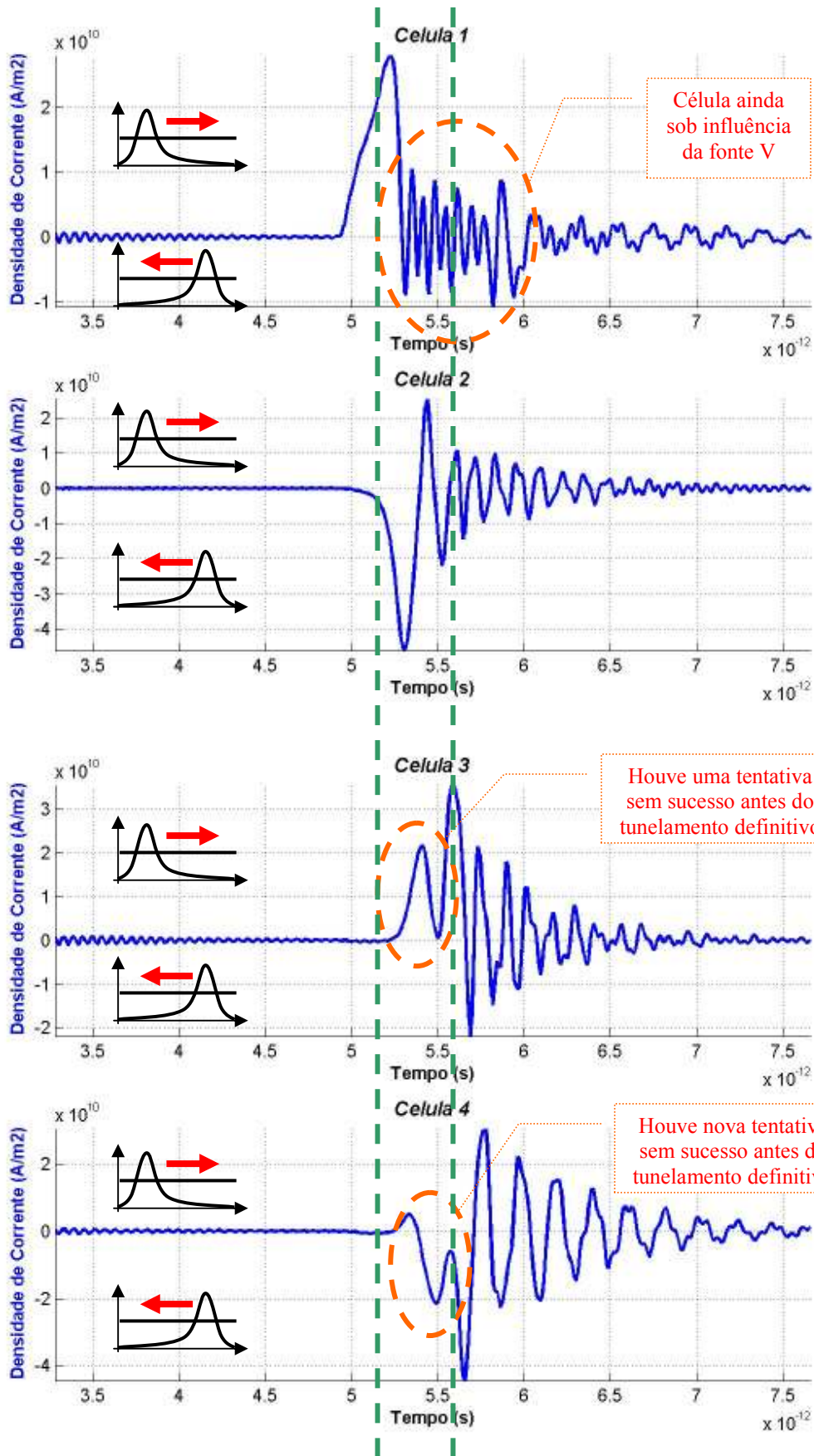


Figura 4-23 – Simulação com 4 células – Corrente no centro das células

A figura 4.22 apresenta a distribuição de carga nos poços durante a execução da simulação com ênfase nos processos de tunelamento. Para facilitar a análise e permitir comparações, no traçado das curvas, procurou-se manter as mesmas janelas de tempo.

Em 5 ps teve início a atuação da rampa de tensão (fonte V) sobre a célula 1. Aproximadamente em 5,2 ps o elétron desta célula se encontrava com sua curva de probabilidade igualmente distribuída entre os poços (ocorre o cruzamento das curvas no gráfico). Este ponto, pela facilidade de inspeção visual, foi escolhido como sendo o instante onde ocorreu o processo de tunelamento, isto para efeitos de medições.

O tunelamento na célula 1 provocou o processo de tunelamento pelas demais células da estrutura. Este era o comportamento desejado. A célula 4 tunelou, pelo padrão de marcação adotado, em, aproximadamente, 5,6 ps. Desta forma, foram feitos 4 chaveamentos binários em 0,4 ps, ou, 1 por 0,1 ps. Isto daria velocidades de clock de processadores, em linhas gerais, da ordem de THz.

Durante a execução da simulação, um fator chamou a atenção. Os tempos de tunelamento não se apresentaram simétricos. O tempo de transição entre as células 1 e 2 é diferente do tempo entre a 2 e a 3 e assim por diante. A maior diferença se dá entre as células 2 e 3.

Na figura 4.23, onde são traçadas as curvas da corrente no centro das células, e mantendo a convenção do item anterior (4.2) onde valores positivos significam correntes da esquerda para a direita, fica mais fácil visualizar a dinâmica do processo. Na célula 1, em função da atuação da fonte de tensão V, ocorre o tunelamento, contudo, após este evento, o elétron não apresenta correntes com maior intensidade, pois ainda está sob a influência da fonte (área assinalada na figura 4.32 – célula 1). O tunelamento na célula 2 aparenta “normalidade” com a corrente oscilando de forma amortecida como esperado. Já as células 3 e 4 apresentam uma tentativa de tunelamento sem sucesso, em função da natureza quântica do elétron, (marcado na figura 4.32 – células 3 e 4) antes do processo definitivo ocorrer. Esta diferença nos tempos de tunelamento levou à construção do próximo cenário onde foram incluídas mais duas células na estrutura com a finalidade de melhor avaliar este fenômeno.

4.4. CENÁRIO COM 6 CÉLULAS ACOPLADAS

Para continuar a avaliação do processo de transmissão de informação em arrays de células acopladas, um novo cenário foi construído. Agora, com 6 células e sendo mantidos os parâmetros iniciais de simulação dos cenários anteriores, o chamado “efeito dominó” (queda da primeira “pedra” se propaga indistintamente pelas demais) não mais ocorre. O tunelamento da célula 1 não é repassado ao longo do array até atingir a célula 6. A figura 4.24 apresenta o diagrama esquemático deste cenário onde, no estado inicial esperava-se a transição em todas as células e no estado final, o que foi obtido.

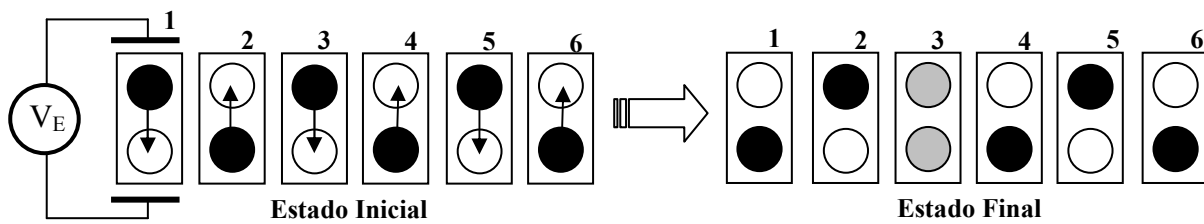


Figura 4-24 – Simulação com 6 Células – Diagrama Esquemático

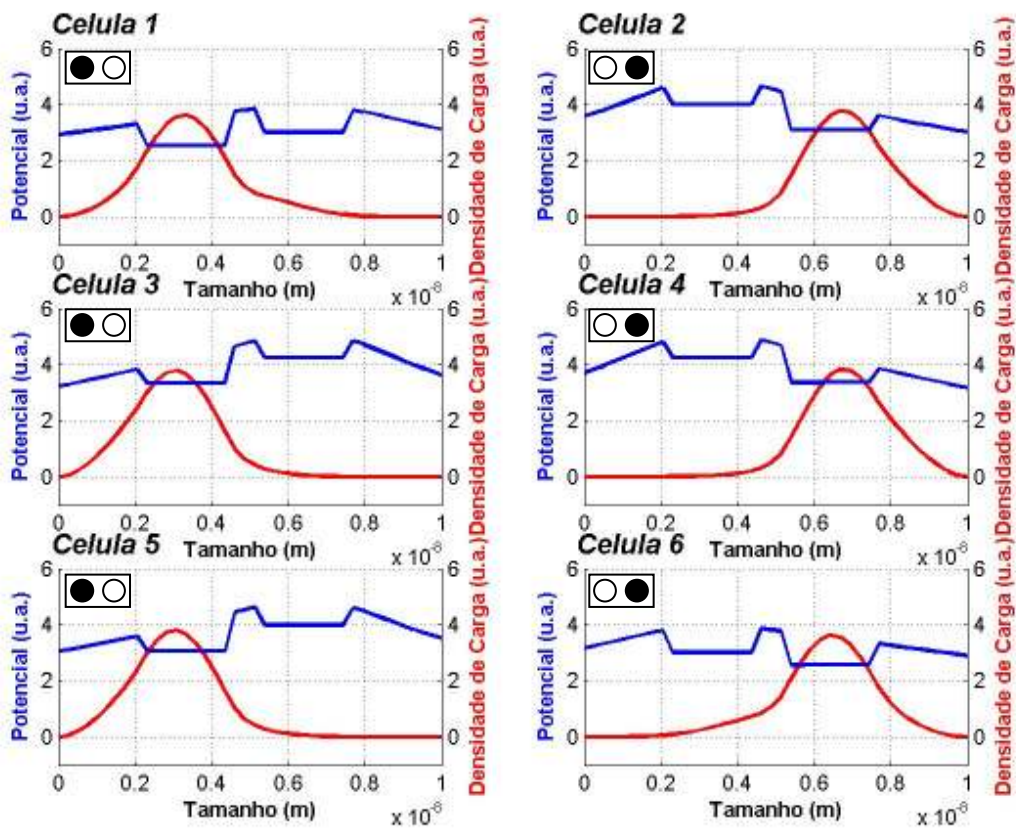


Figura 4-25 – Simulação com 6 Células – Estado Inicial

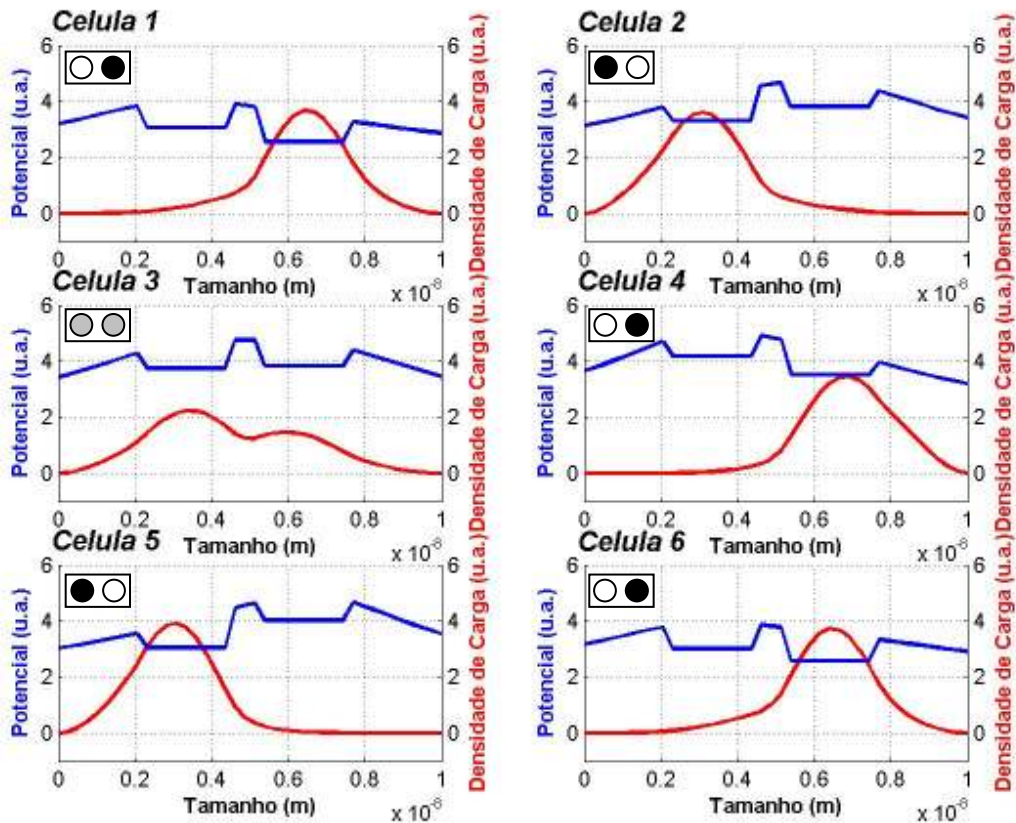


Figura 4-26 – Simulação com 6 Células – Estado Final

As figuras 4.25 e 4.26 apresentam os estados iniciais e finais da simulação. Como nos cenários anteriores, a escolha dos tempos de início e término de atuação da rampa foi feita por inspeção visual. A intensidade e o tempo de atuação da fonte de tensão V foram mantidos iguais aos cenários anteriores para permitir comparações.

As figuras 4.27, 4.28, 4.29, 4.30, 4.31 e 4.32 descrevem a dinâmica do processo ao longo do tempo. Pode ser observado que na célula 3 (fig. 4.29) o elétron não completa o tunelamento permanecendo em um estado onde não há escolha efetiva por um dos poços. As demais figuras (4.30, 4.31 e 4.32) apresentam uma pequena oscilação demonstrando que o efeito de tunelamento nas primeiras células foi visto por elas, contudo, sem maiores efeitos.

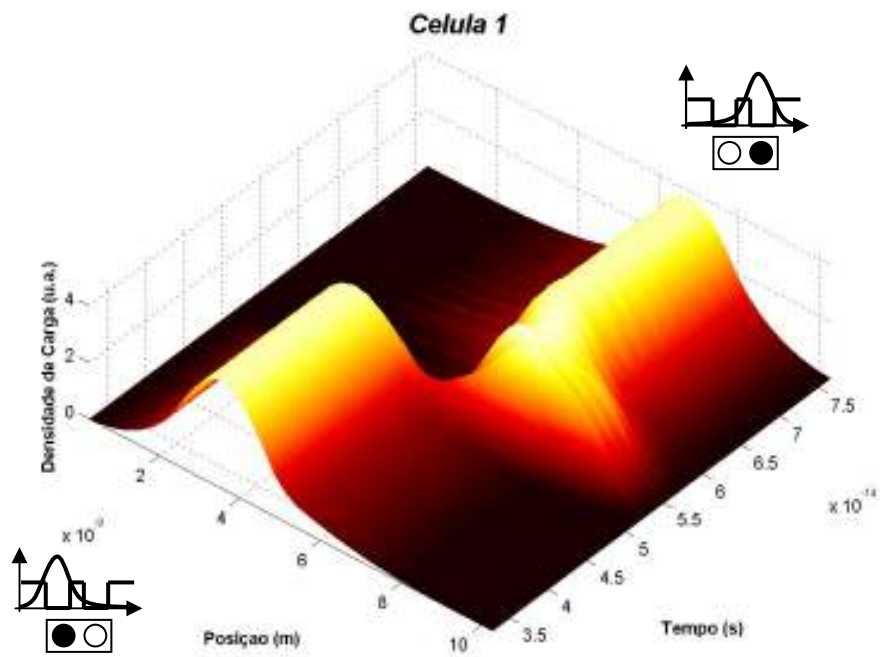


Figura 4-27 – Simulação com 6 células – Dinâmica da Célula 1

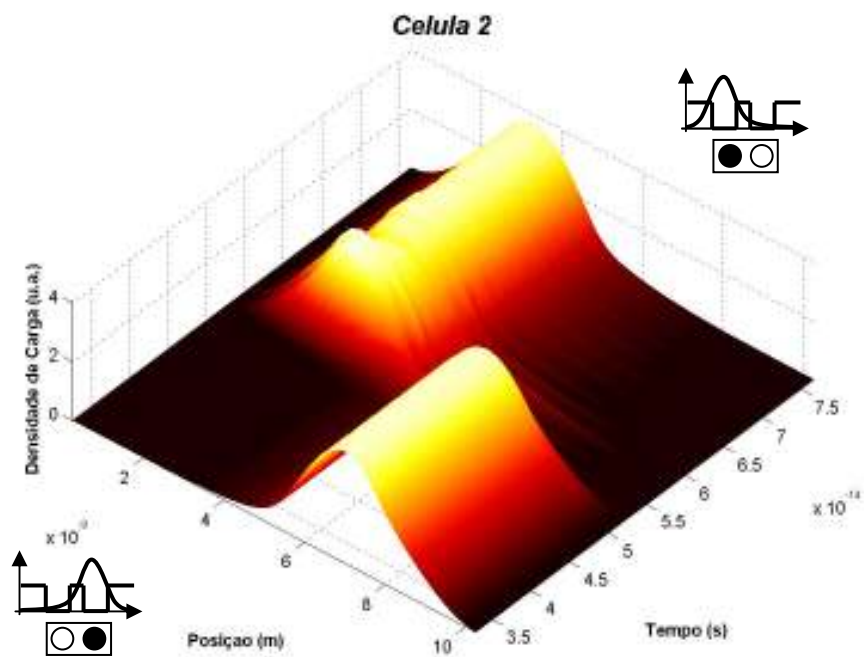


Figura 4-28 – Simulação com 6 células – Dinâmica da Célula 2

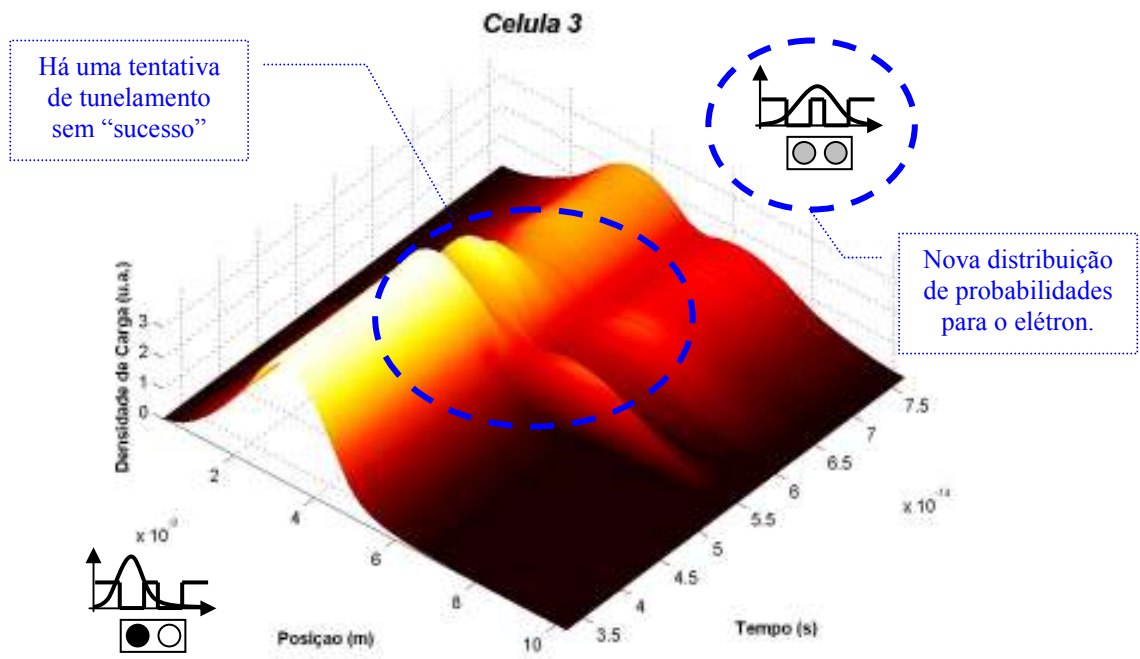


Figura 4-29 – Simulação com 6 células – Dinâmica da Célula 3

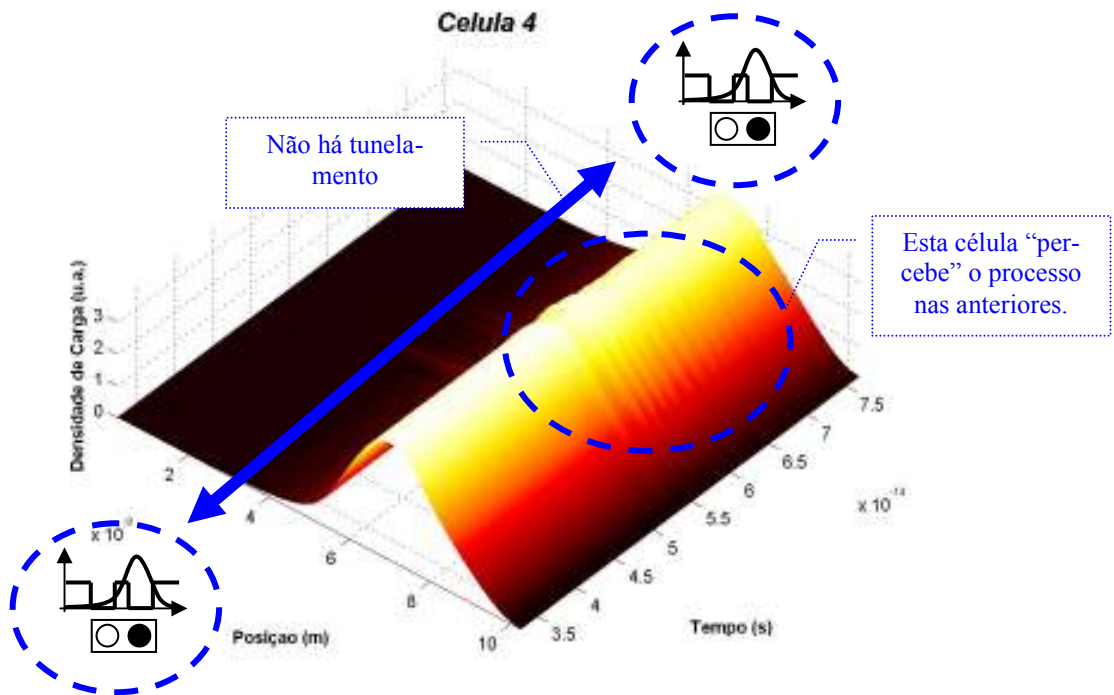


Figura 4-30 – Simulação com 6 células – Dinâmica da Célula 4

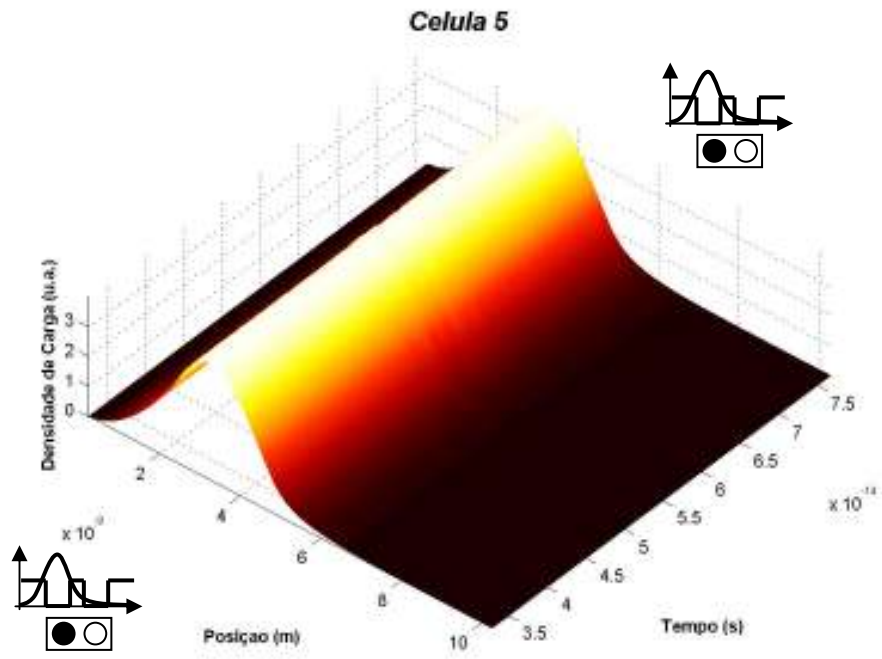


Figura 4-31 – Simulação com 6 células – Dinâmica da Célula 5

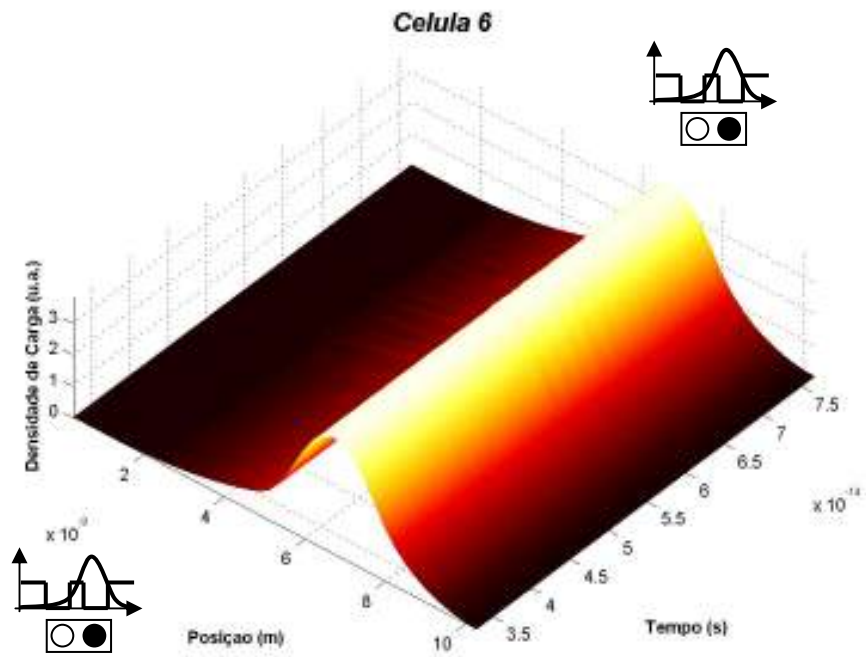


Figura 4-32 – Simulação com 6 células – Dinâmica da Célula 6

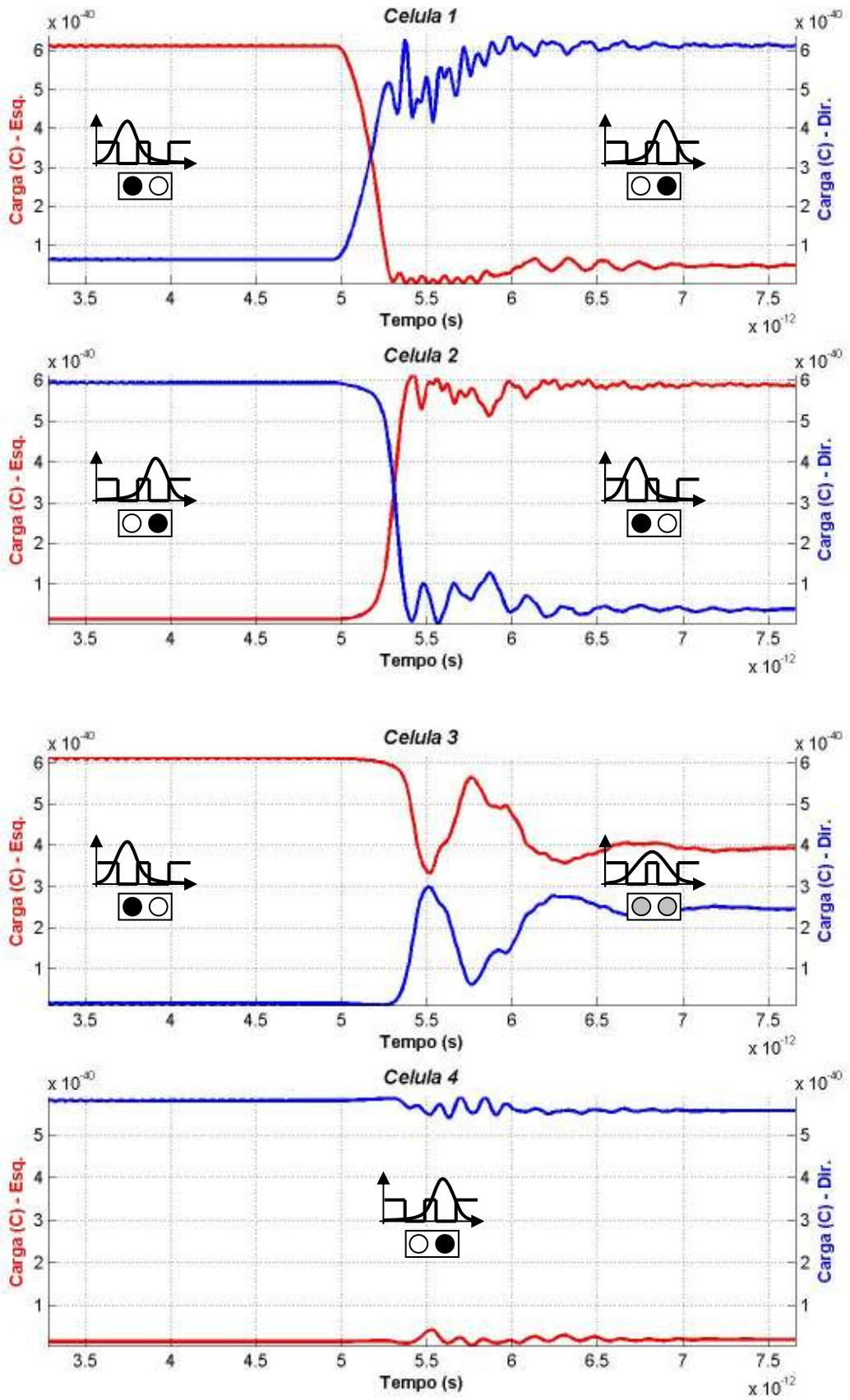


Figura 4-33 – Simulação com 6 células – Carga nos Poços (Células 1, 2 3 e 4)

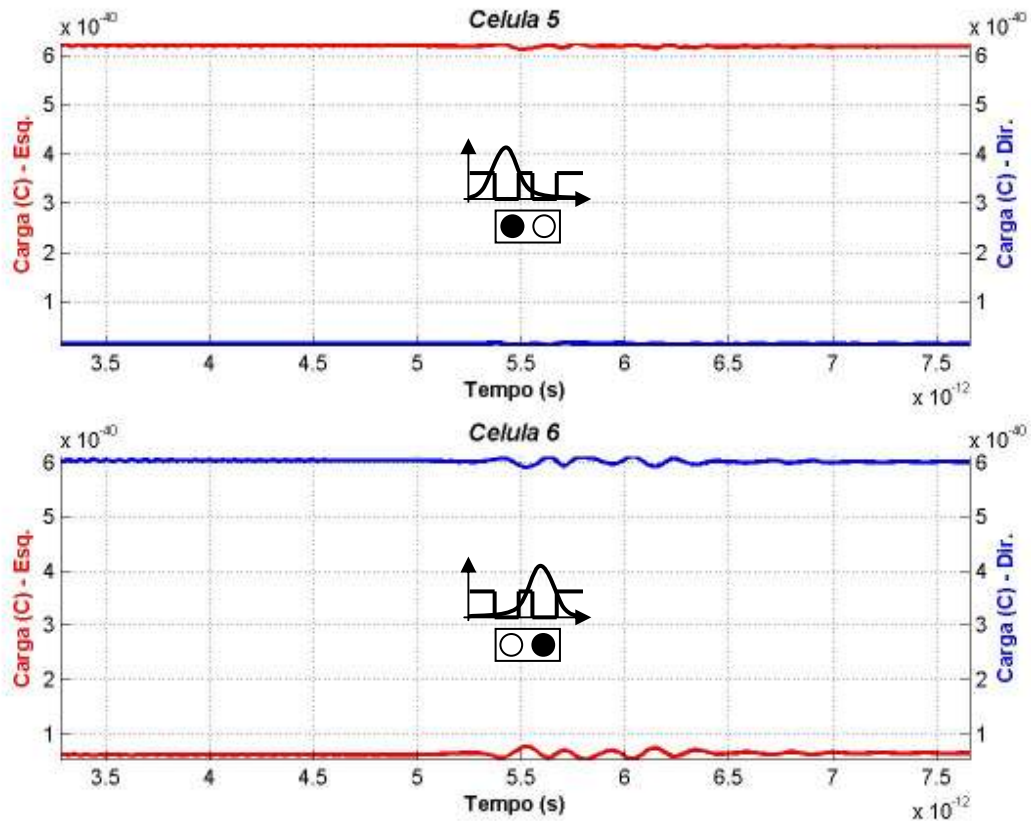


Figura 4-34 – Simulação com 6 células – Carga nos Poços (Células 5 e 6)

As figuras 4.33 e 4.34 apresentam a carga em cada poço das 6 células simuladas. Pode ser observado que, nas células 1 e 2, a transição ocorre como no cenário anterior, com 4 células. Já a célula 3 inicia o processo de tunelamento, mas não consegue completar. As demais células não apresentam alterações significativas.

Uma avaliação pode ser feita desta situação. A cadeia de células possui uma distribuição de cargas que, em função do acoplamento existente entre elas, oferece resistência ao processo de tunelamento. A fonte de tensão V , forçando o tunelamento da primeira célula, faz com que a distribuição de cargas no restante da estrutura se reorganize respondendo ao distúrbio introduzido no ambiente. Assim, por imposição da fonte, as primeiras células tunelam. Com as células 4, 5 e 6 mantendo sua distribuição original, gera, na célula 3, uma distribuição da curva de densidade de probabilidade do elétron sem preferência por nenhum dos poços, interrompendo a seqüência de tunelamentos.

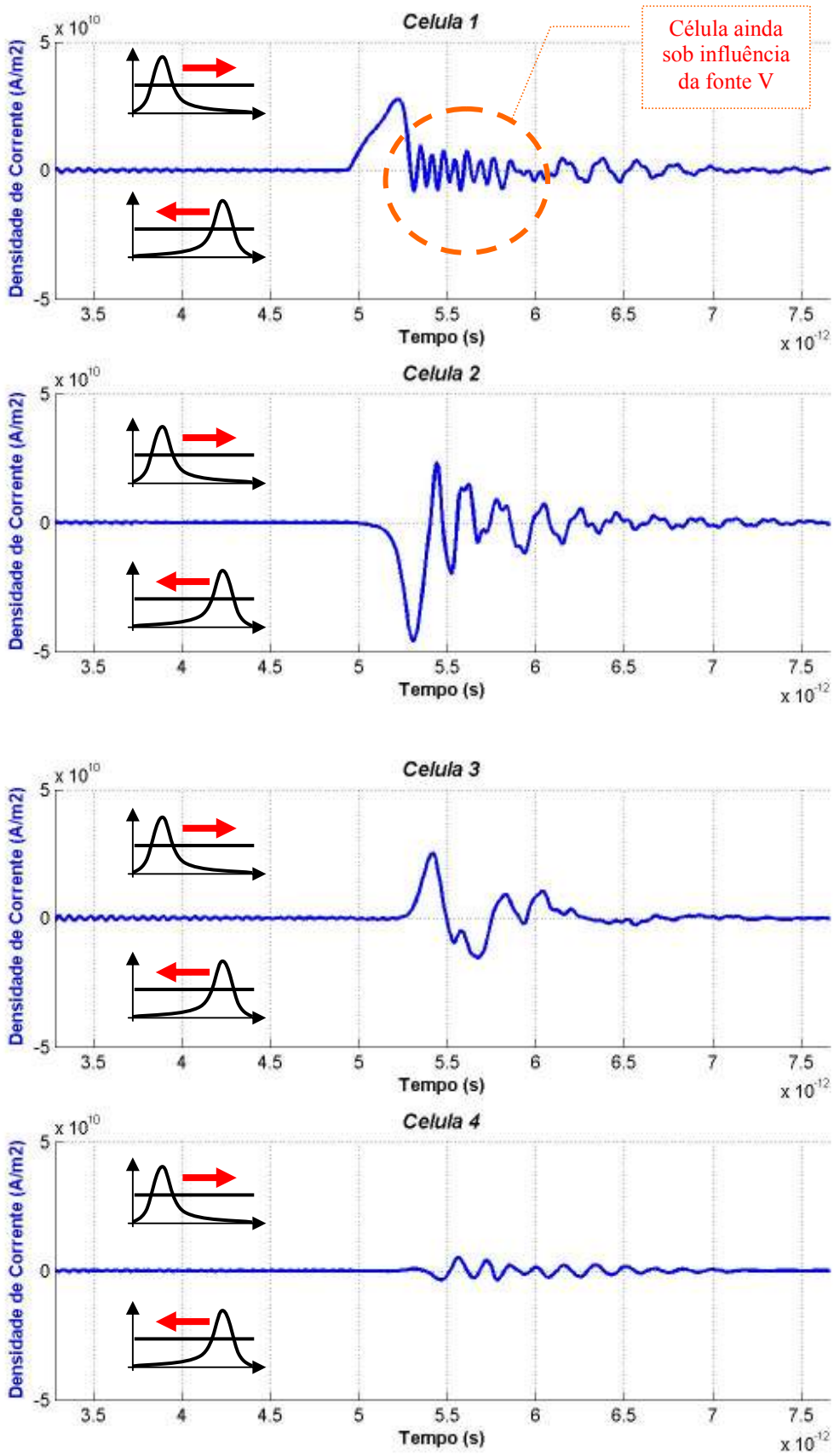


Figura 4-35 – Simulação com 6 células – Corrente no centro das células (Células 1, 2, 3 e 4)

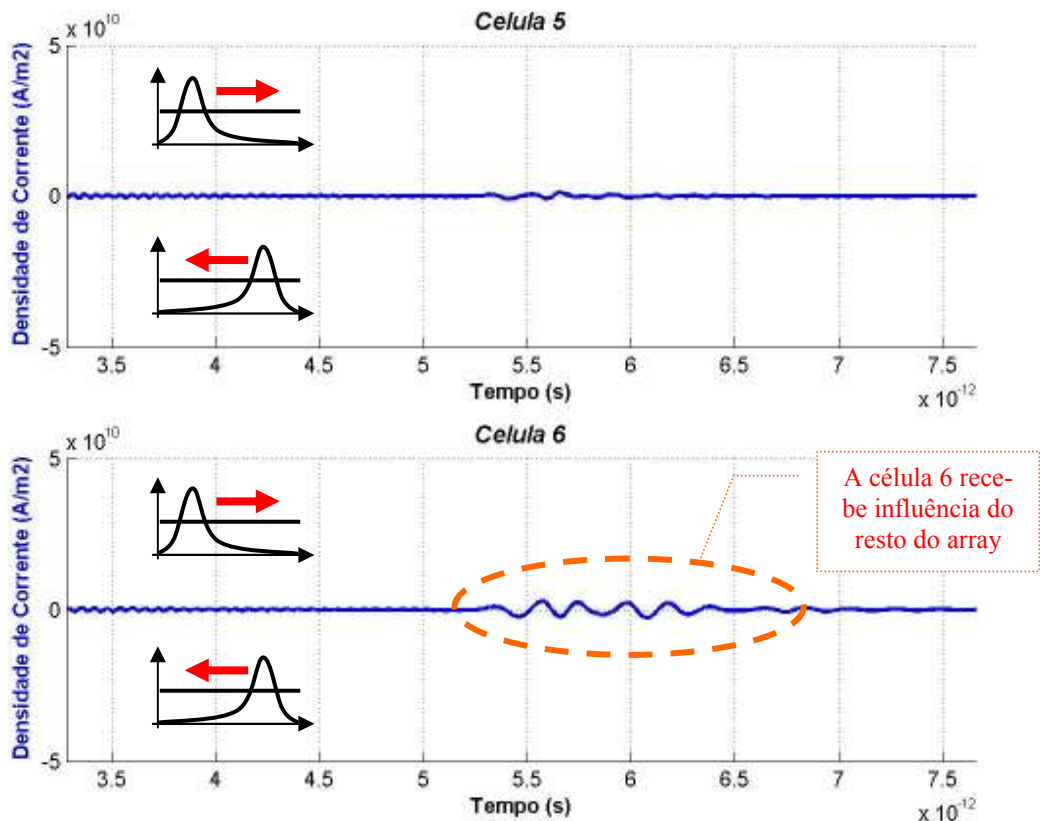


Figura 4-36 – Simulação com 6 células – Corrente no centro das células (Células 5 e 6)

Os gráficos das figuras 4.35 e 4.36 foram traçados mantendo-se as escalas. Desta forma, é possível observar que a intensidade de corrente nos centros das células vai decaindo com o afastamento da fonte. A influência do tunelamento da primeira célula é percebida até a célula 3. A partir da 4, a intensidade da corrente cai bruscamente demonstrando uma blindagem neste ponto. A célula 3 absorve a influência dos dois lados da estrutura.

A figura 4.37 apresenta uma ampliação de escala da figura 4.36. É possível observar que, desde o início da atuação da fonte V, as últimas células já começam a responder ao distúrbio introduzido na estrutura.

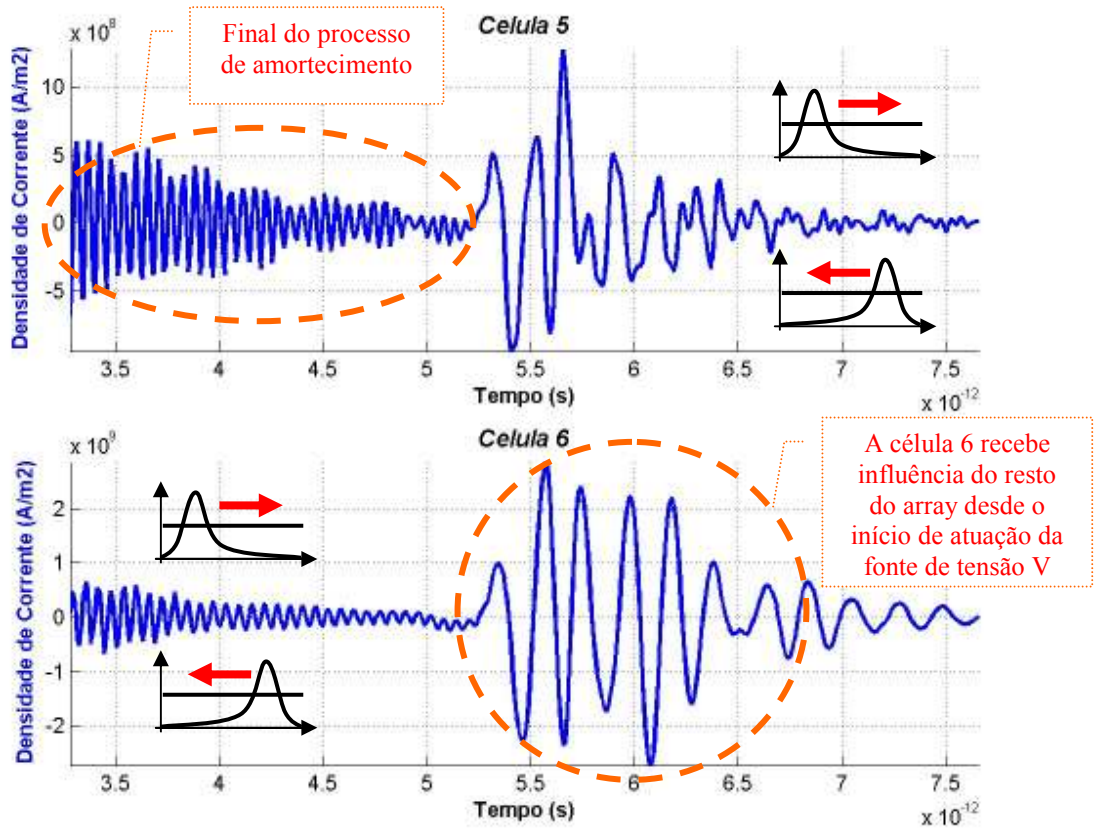


Figura 4-37 – Simulação com 6 células – Zoom da corrente no centro das células 5 e 6

Uma nova simulação foi feita alterando-se a distância efetiva entre os centros (fig. 4.5) e houve sucesso no tunelamento ao longo da estrutura. Com isso, como esperado, é possível concluir que a transmissão de informações por arranjos de QDC's acoplados depende da geometria das células e do acoplamento entre elas.

5. CONCLUSÃO

Neste trabalho foi utilizado um modelo matemático, baseado na equação de Schrödinger e no Método dos Momentos para a análise do comportamento de células de pontos quânticos acopladas. Tal modelo, implementado numericamente, permitiu a simulação de *arrays* de células acopladas lateralmente e a extração de informações relacionadas ao comportamento de cada elétron e da interação entre eles.

A partir do modelo desenvolvido foi feita a análise de um primeiro conjunto com duas células acopladas, sem os poços quânticos, para observar a repulsão entre os elétrons livres da estrutura. As curvas de evolução no tempo foram traçadas descrevendo o comportamento de cada elétron, a partir do estado inicial arbitrário até o estado final de menor energia, com a possibilidade de avaliar, também, o processo dissipativo e sua envoltória. Através das curvas de corrente e distribuição de carga, foi possível avaliar a função de onda (Ψ) se afastando, uma da outra, para os lados opostos em cada célula, de forma simétrica, validando, assim, o acoplamento esperado do projeto.

Foi construído em seguida um *array* com quatro células acopladas para avaliar uma estrutura básica, dentro do proposto por Lent et al [3], para a construção de novos dispositivos pelo uso de QCA's. Traçadas as curvas individuais, que demonstram o comportamento de cada elétron em suas células através do tempo, foi observado efeito da transmissão de informação entre células em conformidade com o proposto inicialmente para este trabalho. A partir das curvas de distribuição de carga nos poços, e das de corrente no centro de cada célula, estimou-se o circuito em análise operando com velocidades de *clock*, em linhas gerais, da ordem de THz. Este fator demonstra o uso de QCA's como uma provável tecnologia para a construção de circuitos integrados que atende as necessidades por produtos mais densos, por ser da ordem de poucos nanômetros, e velozes, por operar em velocidades altas (THz).

Na simulação com quatro células, contudo, observou-se que os eventos de tunelamento não estavam equidistantes no tempo. Para avaliar esta assimetria, montou-se uma nova estrutura com seis células acopladas da qual se pôde observar que a transmissão de informações através do *array* não ocorreu. A partir do estudo do comportamento dos elétrons e das de-

mais curvas de distribuição de carga e corrente foi possível avaliar a dependência da eficiência e robustez destes dispositivos em função das características físicas de cada célula. Desta observação apreende-se que, contrario ao que se observou anteriormente, o uso de QCA's como tecnologia para substituir circuitos integrados carece ainda de maior compreensão e análise.

O programa desenvolvido parte de soluções consistentes de ψ para os elétrons nos momentos iniciais e apesar de não estar incluso um procedimento numérico para garantir a continuidade de ψ e ψ' em cada ponto da simulação entende-se que há consistência nas simulações em função do algoritmo e da formulação utilizados. Contudo, esta validação, bem como a avaliação da conservação de carga, devem ser incluídas e, portanto, sugeridas como continuidade deste trabalho.

Complementando o estudo realizado sobre QCA's, foram estudadas e desenvolvidas técnicas para a otimização e versatilidade no uso dos modelos numéricos quando da implementação computacional do problema proposto. Os recursos de parametrização (que permitiram alterações nas configurações e modelos de simulação) e aceleração de código (que forneceram as velocidades de simulação adequadas ao desenvolvimento do modelo computacional) foram de grande valia na realização do trabalho.

Por fim, novas linhas de estudo devem ser realizadas na construção de circuitos mais complexos, na avaliação da interdependência entre os parâmetros do modelo e robustez dos circuitos, na construção de modelos em 2D e 3D para simulações no estudo de circuitos e, não menos importante, a implementação física destes dispositivos para comprovar na prática o proposto na teoria.

6. REFERÊNCIA BIBLIOGRÁFICA

- [1] – J. C. da Costa et al, *Considerations about nanoelectronic GSI processors*, Journal of Analog Integrated Circuits and Signal Processing, Vol. 24, No. 1, April 1997
- [2] – Schaller R R 1997 IEEE Spectrum 53-9
- [3] – C. S. Lent et al, Applied Physics Letters, 62, 7, p.714-716, February (1993).
- [4] – Eletromagnetismo – William H. Hayt Jr. – Livros Técnicos e Científicos Editora
- [5] – Craig S. Lent, P. Douglas Tougaw, Wolfgang Porod, and Gary H. Bernstein – *Quantum Cellular Automata* – Department of Electrical Engineering – University of Notre Dame – Notre Dame, IN 46556 USA
- [6] – Wolfgang Porod – *Quantum-dot devices and Quantum-dot cellular automata* – Department of Electrical Engineering, University of Notre Dame – Notre Dame, IN 46556 – USA
- [7] – K. Walus, G. Schulhof, and G. A. Jullien – *Circuit Design Based on Majority Gates for Applications with Quantum-Dot Cellular Automata* – University of Calgary – Electrical and Computer Engineering – Calgary, Canada & R. Zhang and W. Wang – University of Western Ontario – Electrical and Computer Engineering – London, Canada
- [8] – Wolfgang Porod, Craig S. Lent, Gary H. Bernstein, Alexei O. Orlov, Islamshah Amalani, Gregory L. Snider, and James L. Merz – *Quantum-dot cellular automata: computing with coupled quantum dots* – Int. J. Electronics, Vol.86, No.5, p.549-590 (1999)
- [9] – Gordon E. Moore – *Gramming more components onto integrated circuits* – Electronics, Vol. 38, No.8, April 1965
- [10] – Manek Dubash – *Moore's Law is dead, says Gordon Moore* – Techworld, 13 Abril 2005
- [11] – Narendra, Siva G. – *Leakage in Nanometer CMOS Technologies* – Chandrakasan, Anantha – 2006 – Springer Ed.
- [12] – Árpád I. Csurgay, Wolfgang Porod, and Craig S. Lent – *Processing with Near-Neighbor-Coupled Time-Varying Quantum-Dot Arrays* - IEEE
- [13] – Lent C S and Tougaw P D – 1997 – Proc. IEEE – 85 541-57

- [14] – G. L. Snider, A. O. Orlov, I. Amlani, X. Zuo, G. H. Bernstein, C. S. Lent, J. L. Merz, and W. Porod – *Quantum-dot cellular automata* – Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana 46556
- [15] – John Timler and Craig S. Lent – *Maxwell's demon and quantum-dot cellular automata* – Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana 46556
- [16] – Gary H Bernstein†, Islamshah Amlani, Alexei O Orlov, Craig S Lent and Gregory L Snider – *Observation of switching in a quantum-dot cellular automata cell* – Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556, USA
- [17] – Y. Imry, *Reviews of Modern Physics*, 71, 2, p.S306, Centenary (1999)
- [18] – Konstantin K Likharev and Tord Claeson – *Single electronics* – Scientific American – June 1992
- [19] – M. V. Batistuta, M. F. Stella, H. Biagi, and J. C. da Costa, in *Electrochemical Society Proceedings, Proceedings of the 19th Symposium on Microelectronics Technology and Devices, SBMicro 2004, Porto de Galinhas, Brasil, PV 2004-03*, p.151, The Electrochemical Society, Pennington, NJ (2004).
- [20] – M. V. Batistuta, M. F. Stella, H. Biagi, and J. C. da Costa, in *Electrochemical Society Proceedings, PV 2004-03*, p. 151, The Electrochemical Society, Pennington, NJ (2004)
- [21] – M. V. Batistuta, M. F. Stella, H. C. de Biagi; J. C. da COSTA, *IEEE Proceedings of NanoSingapore 2006*, p.1-4, Piscataway, NJ (2006)
- [22] – Marcus Vinicius Batistuta – *Modelagem e Simulação de Pontos Quânticos Acoplados* – Tese de Doutorado – Departamento de Energia Elétrica – UnB – 2007
- [23] – MATLAB Version 6.5, from The Math Works Inc., Upper Saddle River, NJ, USA
- [24] – Pascal Getreuer – *Writing Fast MATLAB Code* – August 11, 2004
- [25] – A. Davidson, *Physical Review A*, 41, 6, p.3395, March (1990)
- [26] – A. J. G. Hey, editor, *Feynman and Computation*, Perseus Books, New York (1999)
- [27] – G. Timp, *Nanotechnology*, Springer-Verlag, New York (1999)

APÊNDICE A - CÓDIGO

```
function Celulas_2QD_Acoplamento

% -----
% Universidade de Brasilia
% Departamento de Engenharia Eletrica
%
% Programa : Celulas_2QD_Acoplamento_07.m
%
% Simulacao para potenciais amortecedores em celulas com 2 pontos quanticos cada,
% acopladas por elementos capacitivos. A quantidade de automatoss eh definida como
% parametro da simulacao
%
% Criado em : 20/05/2005
% Ultima atualizacao : 26/11/2006
% -----

clear all
home

global A; % Variaveis de apoio
global C; % Constantes
global M; % Modelos (Celulas)
global E; % Evolucao temporal da simulacao

global Vtotal; % Energia Potencial Total (Joules)
global VautoC; % Auto-Capacitancia (Joules)
global Vpoco; % Energia Potencial da Celula - Poco (Joules)
global Vrampa; % Energia Potencial da Rampa (Joules)
global EX EY EJ; % Psi Real, Psi Imaginario, Densidade de Corrente

% -----
% Variaveis de Apoio
% -----

A.cont = 0; % Contador para amostragens do grafico
A.max = 100; % Espaco entre amostragens do grafico
A.graf = 2; % Modelos de Grafico:
% 1 - Ribbon
% 2 - Sub-Plot
% 3 - Plot

A.cap = 3; % Calculos de Capacitancia
% 1 - Carga Pontual - Esferas - todas as celulas
% 2 - Carga Pontual - Apenas celulas adjacentes
% 3 - Carga Distribuida - Celulas adjacentes

A.TipoGraf = 3; % Escolhe o grafico final
% 1 - Corrente no centro da celula (tudo)
% 2 - Corrente no centro da celula (tulelamento)
```

```

%      3 - Plota a carga nos pocos (tudo)
%      4 - Plota a carga nos pocos (tunelamento)
%      5 - Grafico 3D - uma celula (ver A.TipoGrafCel)
A.TipoGrafCel = 1; % Escolhe a celula a ser mostrada no grafico 3D
A.RampaIni = 50000; % Marcador para Inicio da Rampa
A.RampaPara = 55000; % Marcador de Pausa da Rampa Aplicada
A.RampaFim = 55000; % Marcador para Termina da Rampa
A.RampaVel = 5; % Velocidade de inclinacao da Rampa
A.RampaCt = 0; % Acumulador da inclinacao da Rampa (contador local)
A.TempoTotal = 100000; % Tempo total da Simulacao

% -----
% Constantes fisicas e dos materiais
% -----

C.m = 9.110e-31; % Massa Efetiva do Elétron [Kg]
C.q = 1.602e-19; % Carga Fundamental [C]
C.h = 6.626e-34; % Constante de Plank [J.s]
C.h_ = C.h/(2*pi); % Constante de Plank [J.s]
C.c = 2.99792458e8; % Velocidade da Luz [m/s]
C.Ro = 0.5e12; % Resistividade Total [ohms]
C.mi0 = 1.2566370614e-6; % Permeabilidade magnética do vácuo [H/m]
C.e0 = 8.854187817e-12; % Permissividade elétrica do vácuo [F/m]
C.alfa = C.h_/(2*C.m); % Apoio
C.beta = -2/C.h_; % Apoio
C.MAC = 0; % Matriz de Acoplamento entre as Celulas

% -----
% Parametros do modelo
% -----

M.qt = 6; % Quantidade de Celulas
M.L = 10e-9; % Comprimento total
M.N = 40; % Número de Células do Array.
M.dx = M.L/(M.N-1); % Incremento espacial [m]
M.dt = 1.0940e-016; % Incremento temporal [s]
M.x = 0:M.dx:M.L; % Define os valores do array x de posição
M.Efpi = (pi^2)*(C.h_^2)/(2*C.m*(M.L^2)); % Energia fundamental do poço infinito
M.P1.ini = 10; % Inicio do poco 1 na celula
M.P1.fim = 18; % Fim do poco 1 na celula
M.P2.ini = 22; % Inicio do poco 2 na celula
M.P2.fim = 30; % Fim do poco 2 na celula
M.A = (M.L/2)^2; % Sessao de area da Celula [m^2]
M.R = C.Ro * M.A; % Resistividade Total usada [ohms.m]

```

```

% -----
% Evolucao Temporal da Simulacao
% -----

E.Corrente = zeros (M.qt, A.TempoTotal);

% -----
%
%
% >>>>>>>> Evolucao Temporal das Funcoes de Onda <<<<<<<<<
%
% -----

Matriz_Acoplamento;           % Monta a Matrix de acoplamento entre as celulas
Monta_Celulas;                 % Monta celulas com os pocos quanticos
Inicia_Eletrons;               % Monta a distribuicao de PSI nas celulas

tic
while (A.cont < A.TempoTotal)

    % -----
    % Insere Rampa
    % -----

    if (A.cont > A.RampaIni & A.cont < A.RampaPara)
        A.RampaCt = A.RampaCt + A.RampaVel;
        constante = 6 * 7 * M.Efpi * (A.RampaCt/60000) / M.N;
        Vrampa(M.qt,:) = constante * (M.N : -1 : 1);
    end

    if (A.cont > A.RampaFim & A.RampaCt > 0)
        A.RampaCt = A.RampaCt - A.RampaVel;
        constante = 6 * 7 * M.Efpi * (A.RampaCt/60000) / M.N;
        Vrampa(M.qt,:) = constante * (M.N : -1 : 1);
    end

    % -----
    % Calcula X
    % -----

    diffX = [zeros(M.qt,1), diff(EX,1,2)];
    diffY = [zeros(M.qt,1), diff(EY,1,2)];
    EJ = -(C.q*C.h_/(C.m*M.A*M.dx))*(diffX.*EY - diffY.*EX);
    EJ(:,1) = 0;
    EJ(:,M.N) = 0;

    Calcula_AutoC;

    Vtotal = M.dx .* M.R .* C.q .* cumsum(EJ,2);
    Vtotal = Vtotal + VautoC + Vpoco + Vrampa;

```

```

diff2Y = [zeros(M.qt,1), diff(EY,2,2), zeros(M.qt,1)];
EX = EX - (C.alfa*(M.dt/(M.dx^2))*diff2Y + C.beta.*M.dt.*Vtotal.*EY);

% -----
% Calcula Y
% -----

diffX = [zeros(M.qt,1), diff(EX,1,2)];
diffY = [zeros(M.qt,1), diff(EY,1,2)];
EJ = -(C.q*C.h/(C.m*M.A*M.dx))*(diffX.*EY - diffY.*EX);
EJ(:,1) = 0;
EJ(:,M.N) = 0;

Calcula_AutoC;

Vtotal = M.dx .* M.R .* C.q .* cumsum(EJ,2);
Vtotal = Vtotal + VautoC + Vpoco + Vrampa;

diff2X = [zeros(M.qt,1), diff(EX,2,2), zeros(M.qt,1)];
EY = EY + (C.alfa*(M.dt/(M.dx^2))*diff2X+ C.beta.*M.dt.*Vtotal.*EX);

% -----
% Imprime Grafico
% -----

A.cont = A.cont + 1;
if (rem (A.cont , A.max) == 0)
    Mostra_Grafico;
    E.Corrente (:,A.cont/A.max) = EJ (:,20);
%     E.Dinamica (:,:,A.cont/A.max) = 1e-8 * (EX.^2 + EY.^2);
    E.Dinamica (:,:,A.cont/A.max) = 3e20 * Vtotal;
end

end

toc

% -----
% Mostra Graficos da Evolucao da Simulacao
% -----

MaxX = fix(A.TempoTotal/A.max);
clf
if (A.TipoGraf == 1)
    for k = 1:M.qt
        subplot (M.qt, 1, M.qt - k + 1);
        hold on
        plot (M.dt*A.max*(1:MaxX), E.Corrente(k,1:MaxX), 'b');
        grid on
        hold off
    end
end
end

```

```

if (A.TipoGraf == 2)
    for k = 1:M.qt
        subplot (M.qt, 1, M.qt - k + 1);
        hold on
        plot (M.dt*A.max*(400:800), E.Corrente(k,400:800), 'b');
        grid on
        hold off
    end
end

if (A.TipoGraf == 3)
    for k = 1:M.qt
        subplot (M.qt, 1, M.qt - k + 1);
        hold on
        plot (M.dt*A.max*(1:MaxX), E.CargaEsquerda(k,1:MaxX), 'b'); % Plota a carga nos
pocos (Esquerda)
        plot (M.dt*A.max*(1:MaxX), E.CargaDireita (k,1:MaxX), 'b'); % Plota a carga nos
pocos (Direita)
        grid on
        hold off
    end
end

if (A.TipoGraf == 4)
    for k = 1:M.qt
        subplot (M.qt, 1, M.qt - k + 1);
        hold on
        plot (M.dt*A.max*(400:800), E.CargaEsquerda(k,400:800), 'b'); % Plota a carga nos
pocos (Esquerda) (zoom)
        plot (M.dt*A.max*(400:800), E.CargaDireita (k,400:800), 'b'); % Plota a carga nos
pocos (Direita) (zoom)
        grid on
        hold off
    end
end

if (A.TipoGraf == 5)
    hold on
    SubDim (1:M.N, MaxX:-1:1) = E.Dinamica(A.TipoGrafCel, :, :);
    surf (200:700, 1:M.N, SubDim (1:M.N,700:-1:200))
    colormap (gray)
    shading interp
    view (-50,70)
    grid on
    hold off
end

drawnow;

% -----
%

```



```

% >>>>>>>>> Funcoes de Apoio <<<<<<<<<<
%
% -----
% -----
% Mostra Graficos
% -----

function Mostra_Grafico

    global A M Vtotal VautoC Vpoco EX EY;

    if (A.graf == 1)

        A.graf1 = zeros (M.qt, M.N);
        A.graf2 = zeros (M.qt, M.N);
        A.graf1 = 1e-8 * (EX.^2 + EY.^2);           % Plota densidade nas celulas ("eletron")
        A.graf2 = 3e20 * Vtotal;                   % Plota V das celulas ("celula")

        clf
        grid on
        hold on
        ribbon(M.x,A.graf1')
        ribbon(M.x,A.graf2')
        axis([0 M.qt 0 M.L -4 8]);
        colormap (jet)
        shading interp
        view (94,4)
        hold off
        drawnow;

    end

    if (A.graf == 2)

        clf
        for k = 1:M.qt
            subplot (M.qt / 2, 2, M.qt - k + 1);
            hold on
            plot (M.x, 1e-8 * (EX(k,:) .^2 + EY(k,:) .^2), 'b'); % Plota densidade nas ce-
            lulas ("eletron")
            plot (M.x, 3e20 * Vtotal(k,:), 'k'); % Plota V das celulas
            ("celula")
            % plot (M.x, 3e20 * VautoC(k,:), 'g'); % Plota V auto-
            capacitancia
            grid on
            axis([0 M.L -2 8]);
            hold off
        end
        drawnow;
    end

```

```

end

if (A.graf == 3)

    clf
    for k = 1:M.qt
        hold on
        if (rem(k,2) == 1)
            plot (M.x, .8 * 1e-8 * (EX(k,:) .^2 + EY(k,:) .^2), 'b--'); % Plota den-
sidade nas celulas ("eletron")
        else
            plot (M.x, .8 * 1e-8 * (EX(k,:) .^2 + EY(k,:) .^2), 'b'); % Plota den-
sidade nas celulas ("eletron")
        end
        plot (M.x, 3e20 * Vtotal(k,:), 'k'); % Plota V das celulas
("celula")
        plot (M.x, 3e20 * Vpoco(k,:), 'k'); % Plota V das celulas
("celula")
        grid on
        axis([0 M.L -4 8]);
        hold off
    end
    drawnow;

end

% -----
% Monta Celulas com os Pocos Quanticos
% -----

function Monta_Celulas

global A C M E;
global Vtotal VautoC Vpoco Vrampa;
global EX EY EJ;

Vtotal = zeros(M.qt,M.N); % Energia Potencial Total (Joules)
VautoC = zeros(M.qt,M.N); % Inicia Auto-Capacitancia (Joules)
Vrampa = zeros(M.qt,M.N); % Prepara Rampa (Joules)
Vpoco = 7 * M.Efpi * ones(M.qt,M.N); % Inicia Celula - Poco (Joules)

for k = M.P1.ini:M.P1.fim
    Vpoco(:,k) = 0; % Insere Poco 1
end
for k = M.P2.ini:M.P2.fim
    Vpoco(:,k) = 0; % Insere Poco 2
end

% -----
% Prepara os Eletrons no estado inicial
% -----

```

```

function Inicia_Eletrons

    global A C M E;
    global Vtotal VautoC Vpoco;
    global EX EY EJ;

    psi1 = (C.h_*pi*pi)/(2*M.L*M.L*C.m);           % Apoio
    psi2 = (2*C.h_*pi*pi)/(M.L*M.L*C.m);           % Apoio

    EX = zeros(M.qt,M.N);
    EY = zeros(M.qt,M.N);
    EJ = zeros(M.qt,M.N);

    for k = 1:M.qt

        EX(k,:) = (1/sqrt(M.L))*( sin((pi/M.L)*M.x) + sin((2*pi/M.L)*M.x) );
        % Psi real
        EY(k,:) = (1/sqrt(M.L))*(-sin(psi1*(M.dt))*sin((pi/M.L)*M.x) +
        sin(psi2*(M.dt))*sin((2*pi/M.L)*M.x)); % Psi imaginario
        EJ(k,:) = zeros(1,M.N);                     % Densidade de Corrente

        if (rem(k,2) == 1)                           % Inverte eletrons de celulas impares
            tmp.X = EX(k,:);
            tmp.Y = EY(k,:);
            for w = 1:M.N
                EX(k,w) = tmp.X(M.N - w + 1);
                EY(k,w) = tmp.Y(M.N - w + 1);
            end
        end
    end

    % -----
    % Calcula Auto Capacitancia
    % -----

```

```

function Calcula_AutoC

    global A C M E;
    global Vtotal VautoC Vpoco;
    global EX EY EJ;

    % -----
    % Carga Pontual
    % -----

    if (A.cap == 1 | A.cap == 2)
        V = zeros (1,2*M.qt);
        q = zeros (2*M.qt,1);

        na = (EX.^2 + EY.^2);
    end

```

```

for k = 1:M.qt
    nb = na(k,:) ./ (sum(na(k,:))*M.dx);
    q(2*(k-1)+1,1) = - C.q * sum(nb(M.P1.ini:M.P1.fim)) * M.dx;
    q(2*(k-1)+2,1) = - C.q * sum(nb(M.P2.ini:M.P2.fim)) * M.dx;
end

% V = q \ C.MAC .* 5e-40;
V = C.MAC \ q;

VautoC = zeros(M.qt,M.N);
for k = 1:M.qt
    VautoC(k,M.P1.ini:M.P1.fim) = V(2*(k-1)+1);
    VautoC(k,M.P2.ini:M.P2.fim) = V(2*(k-1)+2);
end
end

% -----
% Carga Distribuida
% -----

if (A.cap == 3)

    mod_a = (EX.^2 + EY.^2);
    for m = 1:M.qt
        mod_b(m,:) = mod_a(m,:) ./ (sum(mod_a(m,:)) * M.dx);
    end

    q = C.q^2 * mod_b' * M.dx / 30;

    if (rem (A.cont , A.max) == 0 & A.cont > 0)
        E.CargaEsquerda (:,A.cont/A.max + 1) = sum(q(M.P1.ini:M.P1.fim,:))'; % Regis-
tra carga nos pocos
        E.CargaDireita (:,A.cont/A.max + 1) = sum(q(M.P2.ini:M.P2.fim,:))';
    end

    VautoC = zeros(M.qt,M.N);

    for m = 1:M.qt
        for f = 1:M.qt
            Vmf = C.MAC(:, :, m, f) * q(:, f);
            VautoC (m, :) = VautoC (m, :) + Vmf';
        end
    end

    for m = 1:M.qt
        VautoC (m,M.P1.ini:M.P1.fim) = sum (VautoC (m,M.P1.ini:M.P1.fim)) / (M.P1.fim -
M.P1.ini);
        VautoC (m,M.P2.ini:M.P2.fim) = sum (VautoC (m,M.P2.ini:M.P2.fim)) / (M.P2.fim -
M.P2.ini);
    end
end

```

```

end

% -----
% Monta Matriz de Acoplamento entre as Celulas
% -----

function Matriz_Acoplamento

global A C M E;
global Vtotal VautoC Vpoco;
global EX EY EJ;

% -----
% Modelo - Esferas com todas as celulas
% -----

if (A.cap == 1)

    a = ((M.Pl.fim - M.Pl.ini)/2)*M.dx;           % Raio do poco (pocos iguais)
    d = (M.N/2)*M.dx;                             % Distancia entre os pocos (mesmo
entre celulas)

    Cap = ones (2*M.qt, 2*M.qt);

    % -----
    % Cap. entre a celula e o terra
    % -----

    for k = 1:2*M.qt
        apoio = 1 + a/d + (a/d)^2 / (1-(a/d)^2);
        Cap(k,k) = 4 * pi * C.e0 * a * apoio / C.q;
    end

    % -----
    % Cap. entre as celulas
    % -----

    for k = 0:M.qt-1
        for w = k:M.qt-1

            if (abs(k-w) > 0)
                c = (abs(k-w)) * d;
                apoio = 1 + a/c + (a/c)^2 / (1-(a/c)^2);
                Cap(2*k+1,2*w+1) = 4 * pi * C.e0 * a * apoio / C.q;
                Cap(2*w+1,2*k+1) = 4 * pi * C.e0 * a * apoio / C.q;
                Cap(2*k+2,2*w+2) = 4 * pi * C.e0 * a * apoio / C.q;
                Cap(2*w+2,2*k+2) = 4 * pi * C.e0 * a * apoio / C.q;
            end

            c = sqrt((abs(k-w))^2 + 1) * d;

```

```

apoio = 1 + a/c + (a/c)^2 / (1-(a/c)^2);
Cap(2*k+1,2*w+2) = 4 * pi * C.e0 * a * apoio / C.q;
Cap(2*w+1,2*k+2) = 4 * pi * C.e0 * a * apoio / C.q;
Cap(2*k+2,2*w+1) = 4 * pi * C.e0 * a * apoio / C.q;
Cap(2*w+2,2*k+1) = 4 * pi * C.e0 * a * apoio / C.q;
end
end

% -----
% Matriz de acoplamento
% -----

C.MAC = zeros(2*M.qt, 2*M.qt);

for k = 1:2*M.qt
    for w = 1:2*M.qt
        C.MAC(k,k) = C.MAC(k,k) + Cap(k,w);
    end
    for w = k+1:2*M.qt
        C.MAC(k,w) = - Cap(k,w);
        C.MAC(w,k) = - Cap(k,w);
    end
end
end

end

% -----
% Modelo - Apenas celulas adjacentes
% -----

if (A.cap == 2)

    Cap = zeros(2*M.qt, 2*M.qt);

    for k = 0:M.qt-1
        for w = k:M.qt-1
            if (abs(k-w) == 1)
                Cap(2*k+1,2*w+2) = 3e1;
                Cap(2*w+1,2*k+2) = 3e1;
                Cap(2*k+2,2*w+1) = 3e1;
                Cap(2*w+2,2*k+1) = 3e1;
            end
        end
    end

    end

% -----
% Matriz de acoplamento
% -----

C.MAC = zeros(2*M.qt, 2*M.qt);

```

```

for k = 1:2*M.qt
    for w = 1:2*M.qt
        C.MAC (k,k) = C.MAC (k,k) + Cap(k,w);
    end
    for w = k+1:2*M.qt
        C.MAC (k,w) = - Cap (k,w);
        C.MAC (w,k) = - Cap (k,w);
    end
end

C.MAC = Cap;
end

% -----
% Modelo - Carga Distribuida
% -----

if (A.cap == 3)

    for m = 1:M.qt
        for f = 1:M.qt
            for x = 1:M.N
                for y = 1:M.N

                    if (m == f)
                        C.MAC (x,y,m,f) = 0;
                    else
                        dist = (M.L / 2) * abs (m-f) / 3; % Distancia entre os centros das celulas
                        dist = dist * .7; % Ajuste na distancia entre as celulas
                        C.MAC (x,y,m,f) = 1 / (4 * pi * C.e0 * sqrt ( dist^2 + (abs(x-y)*M.dx)^2
));
                    end

                end

            end

        end

    end

end

```

APÊNDICE B - PROCESSO DE OTIMIZAÇÃO E ADAPTAÇÃO

i. Organização do Código Original

Com o intuito de modificar o código original, orientado à simulação de apenas uma célula, para simulações com um número maior de células, decidiu-se em conta os recursos do Matlab na criação de funções internas. Desta forma, foi iniciada essa etapa do trabalho pela organização das estruturas matemáticas em funções que passaram a interagir pela passagem de parâmetros entre si. Segue, como exemplo de função, o cálculo da corrente na célula:

```
% -----  
% Calcula a Corrente nas Celulas  
% -----  
  
function Calcula_J  
    global C M EX EY EJ;  
    persistent diffX diffY;  
  
    diffX = [zeros(M.qt,1), diff(EX,1,2)];  
    diffY = [zeros(M.qt,1), diff(EY,1,2)];  
    EJ = -(C.q*C.h_/C.m*M.dx)*(diffX.*EY - diffY.*EX);  
    EJ(:,1) = 0;  
    EJ(:,M.N) = 0;
```

Esta estrutura de código requer a definição de variáveis globais que podem ser vistas e alteradas por qualquer função dentro do programa. Portanto, ao chamar, por exemplo, a função `Calcula_J`, os valores dos arrays `C`, `M`, `EX`, `EY` e `EJ` devem ser referenciados como "globais" para que tenham visibilidade dentro da função.

Outra facilidade utilizada foi a organização das informações em arrays estruturados. Como no exemplo abaixo, a definição das constantes físicas e dos materiais no modelo estão agrupadas dentro do array global `C`. Portanto, `C.m`, que contém o valor da massa efetiva do elétron pode ser endereçado de forma complementar a `C.q` – carga fundamental.


```

% -----
% Constantes físicas e dos materiais
% -----

C.m = 9.110e-31;           % Massa Efetiva do Elétron [Kg]
C.q = 1.602e-19;          % Carga Fundamental [C]
C.h = 6.626e-34;          % Constante de Plank [J.s]
C.h_ = C.h/(2*pi);        % Constante de Plank [J.s]
C.c = 2.99792458e8;       % Velocidade da Luz [m/s]
C.Ro = 0.5e12;            % Resistividade Total
C.mi0 = 1.2566370614e-6;  % Permeabilidade magnética do vácuo [H/m]
C.e0 = 8.854187817e-12;   % Permissividade elétrica do vácuo [F/m]
C.alfa = C.h_/(2*C.m);    % Apoio
C.beta = -2/C.h_;         % Apoio
C.MAC = 0;                % Matriz de Acoplamento entre as Celulas

```

Como ponto positivo desta organização, a facilidade em manipular o código e produzir variantes específicas se tornou evidente. Com isso, foi possível experimentar mais, com maior rapidez, sem comprometer a consistência ou a robustez do modelo.

Contudo, um ponto negativo veio à tona. A criação de funções e a estruturação das informações acabaram gerando uma redução na velocidade da simulação. Em suma, ganhou-se qualidade e organização, mas perdeu-se eficiência computacional.

Para sanar este problema levou-se mão da otimização de códigos no MatLab [24]. O agrupamento de rotinas em funções auxiliou bastante esta etapa, pois os comandos de análise da ferramenta apresentavam seus resultados para a eficiência do código separando por referências à funções internas (definidas no corpo do programa) ou externas (provenientes do próprio MatLab).

ii. Otimização no MatLab

As simulações, por suas próprias características, demandaram um esforço computacional que mereceu uma maior atenção. Algumas técnicas [24] foram empregadas para melhorar os tempos de simulação. Isto permitiu rápidas correções de percurso durante o desenvolvimento, pela repetida execução dos programas permitindo que variações pudessem ser testadas com maior velocidade. Seguem os conceitos básicos utilizados nos processos de otimização do código.

1. O Profiler

Desde a versão 5.0 do Matlab foi introduzida uma ferramenta chamada *Profiler* que auxilia o desenvolvimento pela identificação dos pontos ineficientes de processamento dentro do código. Um exemplo do uso do *Profiler* pode ser visto para o código de exemplo listado abaixo:

```
function result = example1(Count)

for k = 1:Count
    result(k) = sin(k/50);
    if result(k) < -0.9
        result(k) = gammaln(k);
    end
end
end
```

A avaliação deste código, pelo *Profiler* se inicia, antes de qualquer coisa, pela sua ativação e eventual limpeza de dados anteriores, caso existam:

```
>> profile on
>> profile clear
```

Em seguida, a função, ou o código desejado, deve ser executado normalmente. Atenção deve ser dada ao fato de que tal execução deverá acontecer por um período definido e terminar por si só, não entrando, portanto, em um “loop” infinito:

```
>> example1 (5000);
```

Por fim, resta a execução da análise do código pela geração do relatório de avaliação em HTML. O comando que deve ser executado seria:

```
>> profreport ('example1');
```

A partir deste procedimento, o “Profiler” irá abrir uma página HTML com as informações da execução da função como na figura B-1 abaixo:

MATLAB Profile Report: Summary

Report generated 30-Jul-2004 16:57:01

Total recorded time: 3.09 s
Number of M-functions: 4
Clock precision: 0.016 s

Function List

Name	Time	Calls	Time/call	Self time	Location		
example1	3.09	100.0%	1	3.094000	2.36	76.3%	C:\MATLABR11\work\example1.m
gammaIn	0.73	23.7%	3562	0.000206	0.73	23.7%	C:\MATLABR11\toolbox\matlab\specfun\gammaIn.m
profile	0.00	0.0%	1	0.000000	0.00	0.0%	C:\MATLABR11\toolbox\matlab\general\profile.m
profreport	0.00	0.0%	1	0.000000	0.00	0.0%	C:\MATLABR11\toolbox\matlab\general\profreport.m

Figura B-1 – Tela Inicial do Profiler do Matlab

Uma inspeção em “example1” retorna o detalhamento da função e seus respectivos tempos de execução. Isto pode ser visto abaixo (fig 3-9):

```
100% of the total time in this function was spent on the following lines:  
  
      3: for k = 1:Count  
2.11 68% 4:     result(k) = sin(k/50);  
      5:  
0.14  5% 6:     if result(k) < -0.9  
0.84 27% 7:         result(k) = gammaIn(k);  
      8:     end
```

Figura B-2 – Detalhamento de Custo no Profiler do Matlab

As linhas que mais consumiram recursos do ambiente são apresentadas, com seus tempos de execução, percentagem dentro da função e número da linha. As linhas anterior e posterior também são apresentadas para facilitar a análise. No exemplo dado, pode ser observado que as linhas de menor desempenho seriam a 4 e a 7.

2. Pré-alocação de Arrays

Outra técnica para otimização dos códigos se refere a pré-alocação das matrizes. Uma característica interessante do MatLab é a capacidade de redimensionar dinamicamente uma matriz, facilitando a programação:

```
>> a = 2
a =
    2
>> a(2,6) = 1
a =
    2    0    0    0    0    0
    0    0    0    0    0    1
```

Como a matriz “a” foi redimensionada, internamente, a memória alocada teve que ser realocada para um tamanho maior. Se uma matriz tiver que ser redimensionada várias vezes, como em um “loop”, o custo na velocidade torna-se perceptível. Para evitar este problema, pode ser feita uma pré-alocação de matrizes, por exemplo, com o comando “zeros”. O programa a seguir demonstra o código não otimizado para o cálculo aproximado do seno e cosseno de período 100:

```
a(1) = 1;
b(1) = 0;

for k = 2:8000
    a(k) = 0.99803 * a(k-1) - 0.06279 * b(k-1);
    b(k) = 0.99803 * b(k-1) + 0.06279 * a(k-1);
end
```

Uma avaliação pelo *Profiler* demonstra que este código demora cerca de 0,47 segundos (depende do “hardware” utilizado). O maior tempo é gasto no redimensionamento das matrizes “a” e “b” dentro do “loop”. Depois da execução do código, os dois arrays se tornam vetores com 8.000 elementos, portanto, para pré-alocar, é necessário que os arrays sejam criados com esse número de elementos. A seguir, o novo código otimizado:

```
a = zeros(1,8000);    % Pré-alocação
b = zeros(1,8000);

a(1) = 1;
b(1) = 0;
```

```

for k = 2:8000
    a(k) = 0.99803 * a(k-1) - 0.06279 * b(k-1);
    b(k) = 0.99803 * b(k-1) + 0.06279 * b(k-1);
end

```

Com esta modificação, o código passa a ser executado em 0,14 segundos (cerca de três vezes mais rápido).

A pré-alocação é, normalmente, fácil de ser feita. Neste caso, foi necessário apenas identificar o tamanho certo que deveria ser considerado. Há situações onde esta avaliação não é tão simples de ser feita. Nestes casos, uma solução é superdimensionar a pré-alocação e, se necessário, eliminar o excesso quando requerido.

```

a = zeros(1,10000); % pré-alocação
count = 0;

for k = 1:10000
    v = exp(rand(1)*rand(1));

    if v > 0.5 % Acrescenta condicionalmente ao array
        count = count + 1;
        a(count) = v;
    end
end

a = a(1:count); % Elimina o excesso

```

iii. Adaptação do código para Múltiplas Células

Nesta etapa do trabalho procurou-se concentrar todos valores que endereçavam as células em variáveis, evitando o uso de "*magic numbers*" (números que devem ser representados por variáveis, mas são inseridos diretamente em fórmulas, laços ao longo do programa dificultando o desenvolvimento de novos cenários). A seqüência de código abaixo demonstra este mecanismo:

```

% -----
% Parametros do modelo
% -----

M.qt = 2; % Quantidade de Celulas
M.L = 10e-9; % Comprimento total
M.N = 40; % Número de Células do Array.

```

```

M.dx = M.L/(M.N-1); % Incremento espacial [m]
M.dt = M.dx/C.c; % Incremento temporal [s] - Valor inicial
M.P1.ini = 10; % Inicio do pogo 1 na celula
M.P1.fim = 18; % Fim do pogo 1 na celula
M.P2.ini = 22; % Inicio do pogo 2 na celula
M.P2.fim = 30; % Fim do pogo 2 na celula

% -----
% Monta Celulas com os Pocos Quanticos
% -----

function Monta_Celulas
    global A C M E;
    global Vtotal VautoC Vpoco Vrampa;
    global EX EY EJ;

    Vtotal = zeros(M.qt,M.N); % Energia Potencial Total (Joules)
    VautoC = zeros(M.qt,M.N); % Inicia Auto-Capacitancia (Joules)
    Vrampa = zeros(M.qt,M.N); % Prepara Rampa (Joules)
    Vpoco = 7 * M.Efpi * ones(M.qt,M.N); % Inicia Celula

    for k = M.P1.ini:M.P1.fim
        Vpoco(:,k) = 0; % Insere Pogo 1
    end
    for k = M.P2.ini:M.P2.fim
        Vpoco(:,k) = 0; % Insere Pogo 2
    end
end

```

Como pode ser observado, por exemplo, **M.qt** passa a representar o número de células no modelo. Sendo assim, tem-se a liberdade, pela alteração de apenas uma variável (M.qt), alterar a quantidade de células acopladas que entrarão em simulação.

APÊNDICE C - TRABALHOS CORRELATOS

Universidade de Brasília
Faculdade de Tecnologia
Departamento de Engenharia Elétrica
Laboratório de (à definir)

Experimento 1 – Pontos Quânticos Acoplados

Este experimento tem por objetivo iniciar os conceitos de Autômatos Celulares pelo uso de Células de Pontos Quânticos Acoplados (QDCA – Quantum Dot Cellular Automata).

1 – Introdução

Vivenciamos uma evolução tecnológica em circuitos integrados que nos permite observar um direcionamento para o desenvolvimento de sistemas computacionais mais rápidos com maior capacidade de processamento. Cada vez mais temos o uso de circuitos digitais participando de nosso cotidiano, fornecendo novas e melhores ferramentas para as mais diversas aplicações. Como consequência, a evolução natural desses circuitos, em busca de maiores capacidades de processamento e armazenagem, tende para dispositivos cada vez menores e mais rápidos.

O conceito da Lei de Moore, originalmente proposta em 1965, mencionando o crescimento exponencial da densidade dos Circuitos Integrados, permanece basicamente intacto. O advento de novas tecnologias de fabricação está conseguindo prolongá-lo por mais alguns anos. Contudo, limitações fundamentais como dissipação de energia, campos elétricos fortes nas portas e nos drenos e os atrasos de interconexão estão cada vez mais influenciando negativamente a curva de crescimento prevista.

Sendo assim, é prudente a busca por um novo paradigma voltado à computação que não se baseie nos conceitos atuais de transistores com alto grau relativo no consumo de energia e com interconexões lentas. O paradigma computacional conhecido por Quantum-dot Cellular Automata (QCA) é uma possibilidade. A computação por QCA segue os já disseminados conceitos da Lógica Booleana, mas se estrutura sobre a polarização de células ao invés de voltagens e correntes para representar os dados binários. Os QCA oferecem muitas vantagens sobre a tecnologia convencional (ex: CMOS) incluindo a compatibilidade com nano estruturas, interconexões simplificadas e a possibilidade do grande aumento na densidade, dissipação de energia e velocidades computacionais.

O conceito de QCA não se restringe a nenhuma tecnologia em particular podendo ser construídos, por exemplo, sobre pontos quânticos em semicondutores, junções tunnel em metais, estruturas com nanotubos de carbono, self-assembled dots e até moléculas. A figura 1 abaixo exemplifica este conceito e já nos apresenta a analogia com funções lógicas pela presença dos estados binários 0 e 1.

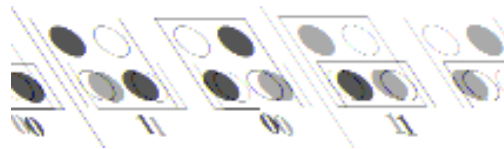


Figura 1 – Pontos Quânticos Acoplados – Representação Binária

Na figura 1, em cada estrutura, com seus pontos quânticos (círculos), há elétrons em excesso (pontos preenchidos). A distribuição de carga nas células ocorre de tal forma que a energia total do sistema tende a ser minimizada, colocando os elétrons nas extremidades. Assim, pela simetria da estrutura, podem ser definidos dois estados discretos. No exemplo da célula com quatro pontos quânticos, há dois elétrons em excesso que tendem a ocupar os cantos opostos da estrutura. Esta energia, que os repelem, provém das forças eletrostáticas. Sendo assim, na estrutura, as duas possibilidades de ocupação dos pontos quânticos podem ser atribuídas aos estados lógicos fundamentais 0 e 1 (bi-estabilidade). O mesmo mecanismo pode ser observado na estrutura com apenas dois pontos quânticos.

Da mesma forma que para os circuitos CMOS, as células de pontos quânticos acoplados devem ser dimensionadas corretamente. Isto para que a bi-estabilidade ocorra e para que as transições obedeçam a uma dinâmica otimizada. Além disso, para que possam ser criados circuitos lógicos com essa tecnologia, a velocidade de propagação de estados entre células torna-se fundamental. A figura 2 abaixo exemplifica esta necessidade.



Figura 2 – Propagação de estados em CQD's

A propagação dos estados ocorre principalmente pela ação dos campos resultantes da redistribuição de cargas dentro das células. Ou seja, há uma polarização de cada célula, que influencia o estado das suas vizinhas. Ao chavear um elétron na primeira célula do array, pela ação de um campo externo, os demais elétrons se redistribuem para minimizar a energia total do sistema. A figura 3 abaixo demonstra este princípio.

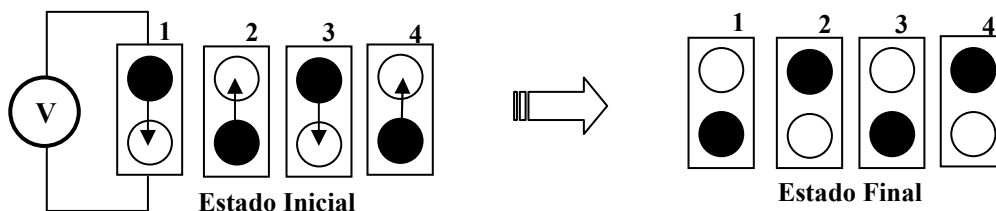


Figura 3 – Propagação de estados em CQD's

Uma possível estrutura para a célula com 4 pontos quânticos pode ser observada a seguir:

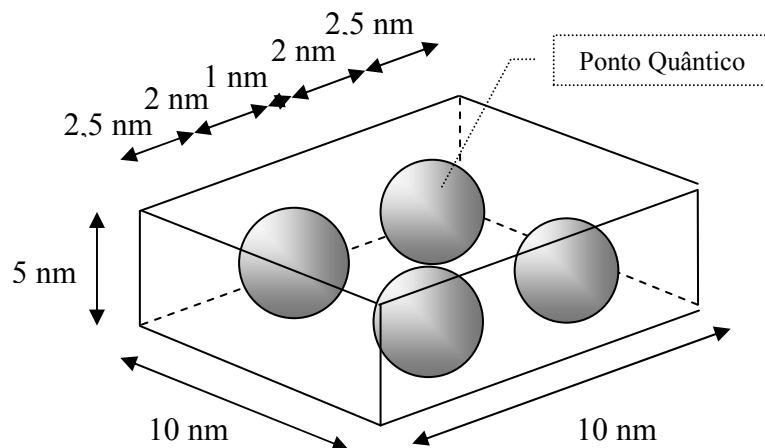


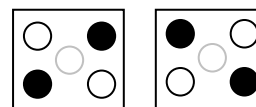
Figura 4 – Célula com 4 pontos quânticos

Em alguns casos, um quinto ponto quântico pode ser incluído no centro da célula com o objetivo principal de auxiliar o processo de tunelamento.

2 – Pré-Laboratório

Produzir as seguintes definições:

- Ponto Quântico;
- Tunelamento;
- Células com 4 pontos quânticos acoplados;
- Descrição da dinâmica da célula;
- Como se dá o transporte de informações.
- Conceito de QCA



3 – Procedimento de Laboratório

Localizar o Software QCADesigner (www.qcadesigner.ca) e montar descrição das funcionalidades macro de sua interface.

- Inserir, movimentar e apagar células;
- Definir pontos de entrada e saída;
- Definir células pré-polarizadas;
- Efetuar simulação de 8 células em série, obter e interpretar os gráficos;

O manual de utilização do software pode ser encontrado em:

<http://www.qcadesigner.ca/manual/index.html>

3 – Bibliografia

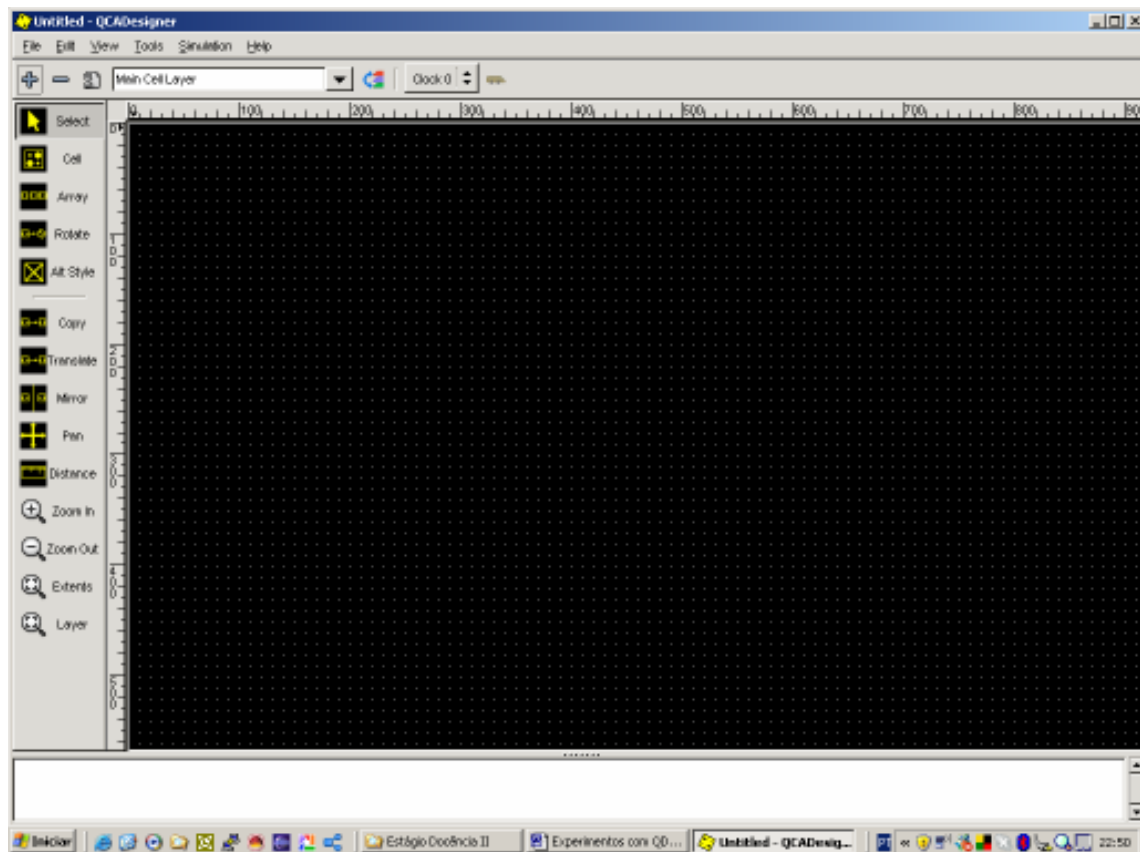
- “QUANTUM CELLULAR AUTOMATA” – Craig S. Lent, P. Douglas Tougaw, Wolfgang Porod, and Gary H. Bernstein – Department of Electrical Engineering – University of Notre Dame – Notre Dame, IN 46556 USA
- Site: www.qcadesigner.ca
- “Quantum-dot devices and Quantum-dot cellular automata” – Wolfgang Porod – Department of Electrical Engineering, University of Notre Dame – Notre Dame, IN 46556 - USA

Experimento 1 – Pontos Quânticos Acoplados Roteiro para o Instrutor

Este experimento tem por objetivo iniciar os conceitos de Autômatos Celulares pelo uso de Células de Pontos Quânticos Acoplados (QDCA – Quantum Dot Cellular Automata). Para tanto, será apresentada a ferramenta QCADesigner que permite a simulação dos conceitos básicos de autômatos celulares.

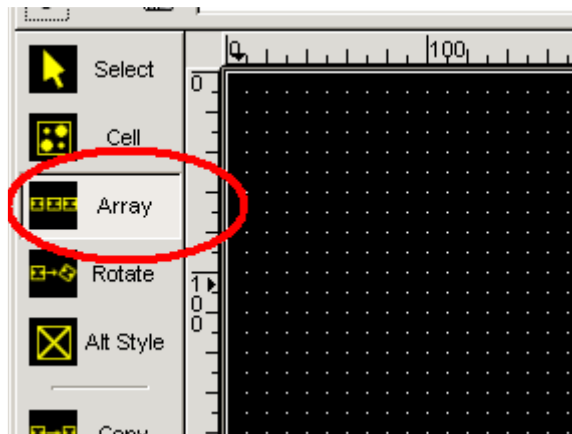
1 – Roteiro

Ao iniciar o QCADesigner será apresentada a seguinte tela:

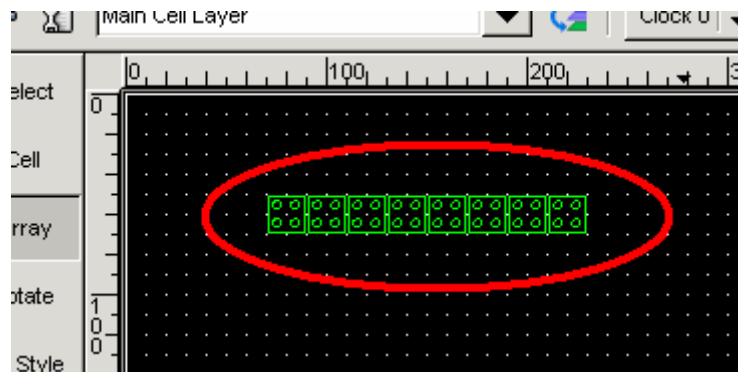


Neste primeiro experimento, será construído e analisado um array de 8 células com 4 pontos quânticos cada. Para tanto, os seguintes passos deverão ser seguidos:

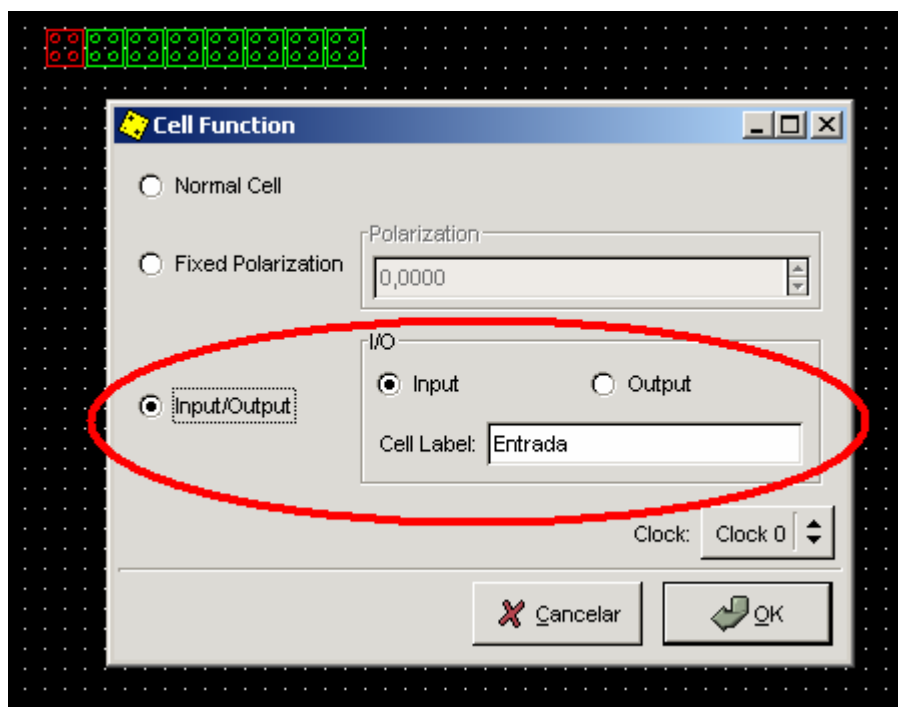
Passo 1 – Selecionar a ferramenta de Array:



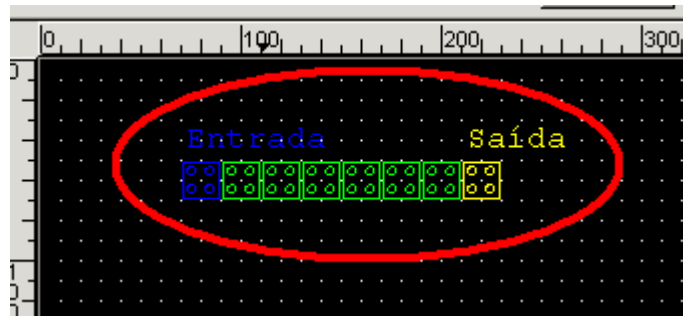
Passo 2 – Criar um array com 8 elementos no ambiente:



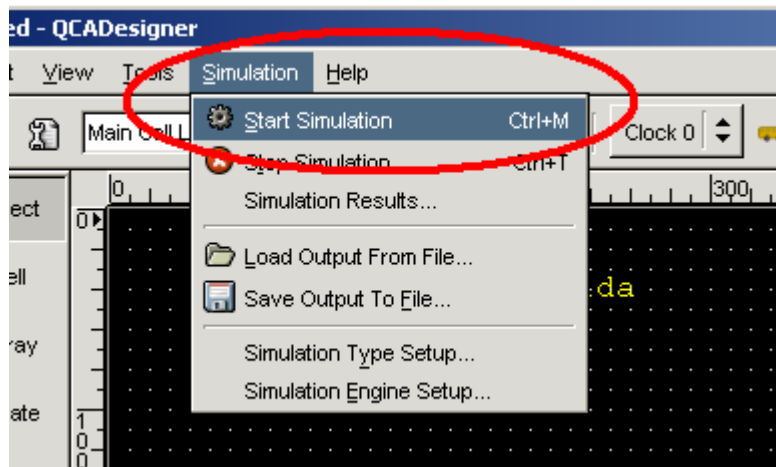
Passo 3 – Definir pontos de entrada e saída para a simulação. Para tanto, deve-se selecionar uma célula e clicar duas vezes sobre ela obtendo o quadro abaixo:



Defina as células como a seguir:



Passo 4 – Realizar a simulação:

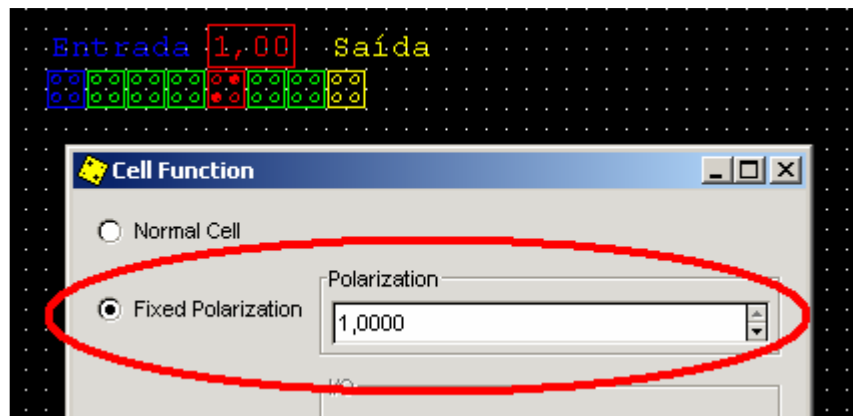


Passo 5 – O seguinte resultado deverá ser obtido:



Esta ferramenta faz uso de um clock para sincronizar os efeitos de tunelamento. Tal clock é implementado pela inserção de um quinto ponto quântico no centro das células. Como foram assumidas as configurações padrão do programa, o clock em ação, nesta simulação, é o “Clock 0”. Portanto, enquanto este se encontra em uma energia tal que favoreça a transição de estados nas células, inicialmente, a “Saída” não tem preferência de estado distribuindo a energia igualmente entre os pontos quânticos. Quando o “Clock 0” passa a uma energia mais baixa, os elétrons excedentes nas células tendem a um estado estacionário. Como a “Entrada” está alta, as transições nas células internas do Array fazem com que a “Saída” também esteja alta, acompanhando a “Entrada”. Desta forma, acompanhando as curvas no tempo, infere-se que este circuito replica a informação da “Entrada” na “Saída”.

Uma variação interessante seria fixar o valor de uma célula no interior do array:



Assim, pode ser observado que, independente da “Entrada”, a “Saída” permanece fixa em “1”:



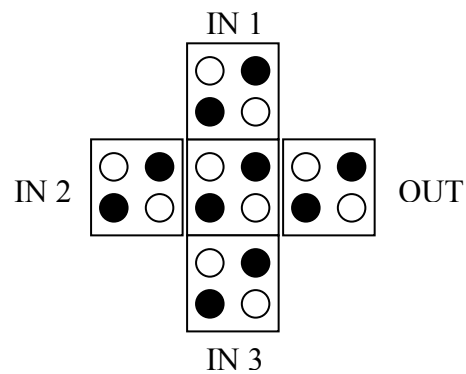
Experimento 2 – Majority Gate

Este experimento visa dar os primeiros passos no ambiente de simulação do QCADesigner pela construção de um Majority Gate

1 – Pré-Laboratório

Produzir as seguintes definições:

- Definição de um Majority Gate;
- Princípio de Funcionamento;



2 – Procedimento de Laboratório

Implementar no QCADesigner um Majority Gate e realizar a simulação.

3 – Bibliografia

- “Circuit Design Based on Majority Gates for Applications with Quantum-Dot Cellular Automata” – K. Walus, G. Schulhof, and G. A. Jullien – University of Calgary – Electrical and Computer Engineering – Calgary, Canada & R. Zhang and W. Wang – University of Western Ontario – Electrical and Computer Engineering – London, Canada

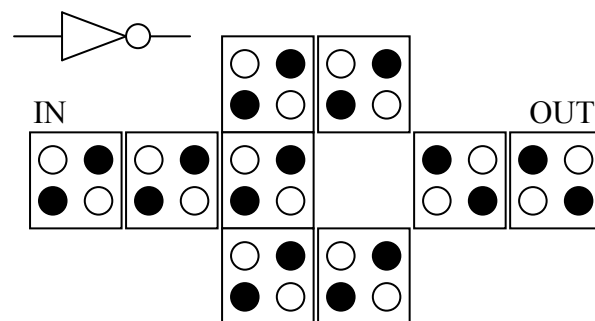
Experimento 3 – Porta Inversora

Este experimento tem por objetivo simular uma Porta Inversora e avaliar as possíveis configurações de Clock.

1 – Pré-Laboratório

Produzir as seguintes definições:

- Definição de uma Porta Inversora;
- Princípio de Funcionamento;



2 – Procedimento de Laboratório

Implementar no QCADesigner uma Porta Inversora e realizar a simulação.

3 – Bibliografia

- “Quantum-dot cellular automata” - G. L. Snider, A. O. Orlov, I. Amlani, X. Zuo, G. H. Bernstein, C. S. Lent, J. L. Merz, and W. Porod – Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana 46556

Universidade de Brasília
Faculdade de Tecnologia
Departamento de Engenharia Elétrica
Laboratório de (à definir)

Experimento 4 – Porta “E”

Este experimento tem por objetivo simular uma Porta Lógica “E” e avaliar o uso da Vector Table nas opções de simulação.

1 – Pré-Laboratório

Produzir as seguintes definições:

- Definição de uma Porta Lógica “E”;
- Princípio de Funcionamento;
- Definição de uma Vector Table, dentro do QCADesigner.

2 – Procedimento de Laboratório

Implementar no QCADesigner uma Porta Lógica “E” de 2 bits e orientar a simulação configurando uma Vector Table adequada com as 4 possibilidades de entrada.

3 – Bibliografia

- “Quantum-dot cellular automata: computing with coupled quantum dots” – WOLFGANG POROD, CRAIG S. LENT, GARY H. BERNSTEIN, ALEXEI O. ORLOV, ISLAMSHAH AMLANI, GREGORY L. SNIDER, and JAMES L. MERZ

Universidade de Brasília
Faculdade de Tecnologia
Departamento de Engenharia Elétrica
Laboratório de (à definir)

Experimento 5 – Porta “OU”

Este experimento tem por objetivo simular uma Porta Lógica “OU” e avaliar o uso do agrupamento em BUS.

1 – Pré-Laboratório

Produzir as seguintes definições:

- Definição de uma Porta Lógica “OU”;
- Princípio de Funcionamento;

2 – Procedimento de Laboratório

Implementar no QCADesigner uma Porta Lógica “OU” e montar a configuração de BUS para as entradas e saída.

3 – Bibliografia

- “Quantum-dot cellular automata: computing with coupled quantum dots” – WOLFGANG POROD, CRAIG S. LENT, GARY H. BERNSTEIN, ALEXEI O. ORLOV, ISLAMSHAH AMLANI, GREGORY L. SNIDER, and JAMES L. MERZ

Universidade de Brasília
Faculdade de Tecnologia
Departamento de Engenharia Elétrica
Laboratório de (à definir)

Experimento 6 – Porta “XOR”

Este experimento tem por objetivo simular uma Porta Lógica “XOR” e avaliar o uso de Layers.

1 – Pré-Laboratório

Produzir as seguintes definições:

- Definição de uma Porta Lógica “XOR”;
- Princípio de Funcionamento;

2 – Procedimento de Laboratório

O QCADesigner vem com uma porta XOR como exemplo. Caberá, neste experimento, sua simulação e análise.

3 – Bibliografia

Como fonte bibliográfica, usar o site do QCADesigner.

Universidade de Brasília
Faculdade de Tecnologia
Departamento de Engenharia Elétrica
Laboratório de (à definir)

Experimento 7 – Somador Completo

Este experimento tem por objetivo simular um Somador Completo.

1 – Pré-Laboratório

Produzir as seguintes definições:

- Definição de um Somador Completo;
- Princípio de Funcionamento;
- Tempos de Propagação.

2 – Procedimento de Laboratório

Montar e simular um Somador Completo de 2 bits.

3 – Item extra para análise

A ferramenta QCADesigner não possui uma análise maior com relação aos tempos de tunelamento e seu impacto no funcionamento dos circuitos. Portanto, deverá ser elaborado um ensaio sobre este tema e suas implicações sob a ótica do projeto de processadores.

4 – Bibliografia

- “Quantum-dot cellular automata” - G. L. Snider, A. O. Orlov, I. Amlani, X. Zuo, G. H. Bernstein, C. S. Lent, J. L. Merz, and W. Porod – Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana 46556

- “Quantum-dot cellular automata: computing with coupled quantum dots” – WOLFGANG POROD, CRAIG S. LENT, GARY H. BERNSTEIN, ALEXEI O. ORLOV, ISLAMSHAH AMLANI, GREGORY L. SNIDER, and JAMES L. MERZ

APÊNDICE D - PUBLICAÇÕES REALIZADAS

Coupled Quantum Dot Cell Dynamics

M. F. Stella, M. V. Batistuta and J. C. da Costa

Departamento de Engenharia Elétrica
Universidade de Brasília,
CP 4386, Brasília, DF 70904-970, Brazil

A simulation scheme was developed for the dynamic transport behavior of a number of laterally coupled double quantum dot cells. A simple one dimensional model was used for the wavefunction dynamics analysis and solving the electrostatic potentials. Behavior prediction and performance evaluation (i.e. settling time, frequency response) have been obtained. The developed simulation strategy allows the calculation of state transit times along chains of coupled quantum dots.

Introduction

As next generations of integrated circuits are developed, the continual reduction of device dimensions is leading to faster devices and denser structures. Smaller devices might lead to new paradigms for information processing based on locally-interconnected architectures such as Cellular Automata (CA) (1). As alternatives, Single Electronics, with Coupled Quantum Wells (CQW) and Coupled Quantum Dots (CQD) have been proposed. The Coulomb Blockade effects and modulated tunneling transport phenomena can be observed on such devices. In order to accurately estimate dynamic behavior in CQD-based nano-electronic circuits, single-electron transport models must be developed and robust numerical simulation strategies are needed (2).

A basic cell structure was originally proposed by Lent et al (3) for the implementation of cellular automata. This concept is not restricted to a particular technology and can be built from, for example, quantum dot cells, metallic tunnel junctions, carbon nano-tubes, self-assembled dots and even molecules (4). This work focus on coupled double dot cells (5), including bi-stability with just a single excess electron on each cell. Logic states can be attributed to such cells, based on electric charge distribution, as represented in Fig.1. In this example, the dark dot indicates where the charge is mostly concentrated in the cell.



Fig. 1. Double-dot cell states with one excess electron

Another possible scheme was also proposed in the form of laterally coupled double-dot single-electron cells – Fig.2. By this arrangement, binary wire or string can be implemented creating a quantum-dot cellular automata (6).

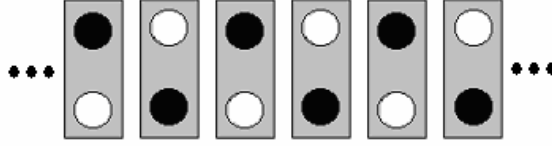


Fig. 2. Wire or string of laterally coupled double-dot cells

In earlier works (5,8,9), the dissipative dynamical behavior of single electrons in coupled quantum wells, and its numerical implementation, making use of fundamental physical principles, such as charge and energy conservation, was built, including a deceleration field, acting directly on the electron's dynamic damping behavior (5).

This work employs the integral form of Poisson's equation for electrostatic potentials to describe the coupling between cells, without charge exchange. As a result it allows the analysis of digital information exchanges between cells, while providing the necessary dynamics to the QCA (quantum-dot cellular automata) structure as proposed by Lent et al (3).

Transport dynamics of coupled cells

The numerical simulations consider that each basic cell has two dots and a single excess electron on it. This electron may tunnel through the barrier that exists between the pair of quantum dots under the action of time-varying potentials. Fig. 3 represents this basic cell.

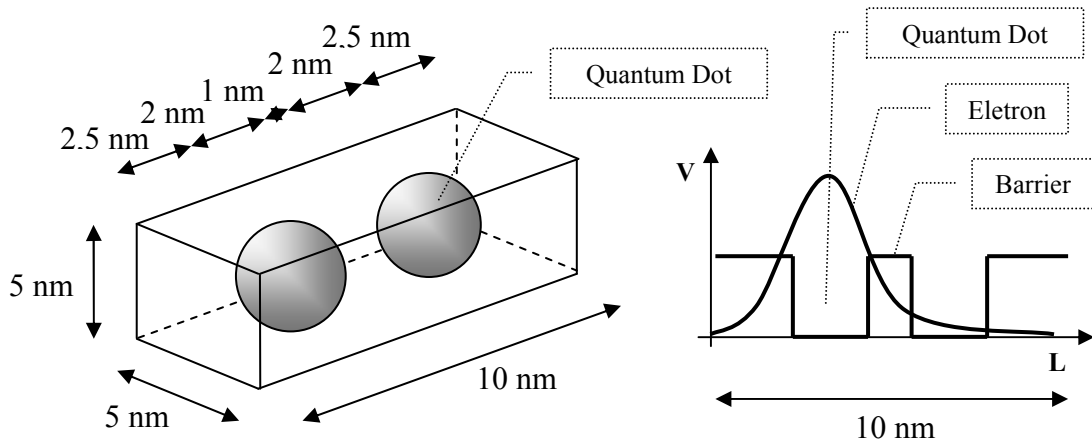


Fig. 3. Basic Double Quantum Dot Cell used on simulation.
(Physical representation – left – and numerical one – right)

For each excess electron a separate Schrödinger's equation is established in one dimension (eq. 1).

$$\left[-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + eV(x,t) + S(x,t) \right] \Psi(x,t) = i\hbar \frac{\partial \Psi(x,t)}{\partial t} \quad [1]$$

Here, at eq. 1, is presented the damping factor S , that is responsible for the energy dissipation, the electric potential V is defined by the time-invariant energy profile of the quantum dots, plus the externally applied potential, e is the fundamental electronic charge and m is the effective mass.

Charge density distributions (ρ), at a given time, are calculated from the instant values of the wavefunctions, for each cell, from equation 2.

$$\rho(x, t) = e.n(x, t) = e.(\Psi(x, t).\Psi^*(x, t)) \quad [2]$$

The total charge inside each dot of a cell (q_i), at a given time, is calculated from equation 3. Integration of charge density distribution is carried out for each dot's domain, in one dimension.

$$q_i(t) = \int_i \rho(x, t) dx \quad [3]$$

The integral form of Poisson's equation for electrostatic potentials, as on equation 4, is used to make the cells couple to each other, without exchange of charge. The distance between each dot is represented by $|\vec{r} - \vec{r}'|$.

$$V(\vec{r}) = \int_{Volume} \frac{\rho(\vec{r}') dv'}{4\pi\epsilon|\vec{r} - \vec{r}'|} \quad [4]$$

Numerical Simulation

Each cell on the numerical simulation is composed by an array and, considering its physical dimensions, can be presented as in Fig. 4.

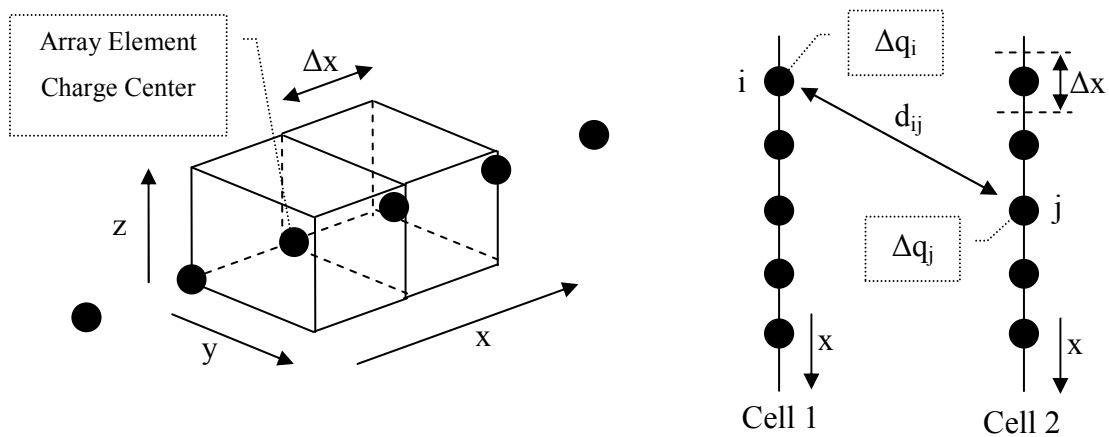


Fig. 4. Poisson coupling diagrams

So, the potential of the element j of the Cell 1 array can be describe as in equation 5 that is the discrete representation of the integral form of Poisson's equation (eq. 4).

$$V_j \approx \sum_{\forall i} V_{ij} \approx \sum_{\forall i} \frac{\Delta q_i}{4\pi\epsilon \cdot d_{ij}} \quad [5]$$

The main objective is to study the state transition propagation along the coupled cells. A four cells configuration, disposed like in Fig. 5, has been chosen as a start point to analyze the wavefunction dynamics of each cell.

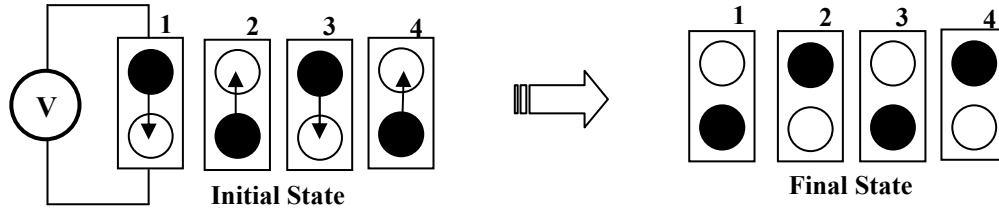


Fig. 5. Four cell configuration

The voltage source applies a potential difference to the first cell and makes the excess electron tunnel from one dot to the other. The new charge redistribution induces new tunneling events along the chain of cells. Fig. 6 shows potential distributions and probability density functions along the four cells used on this simulation at the beginning and at the end of the process.

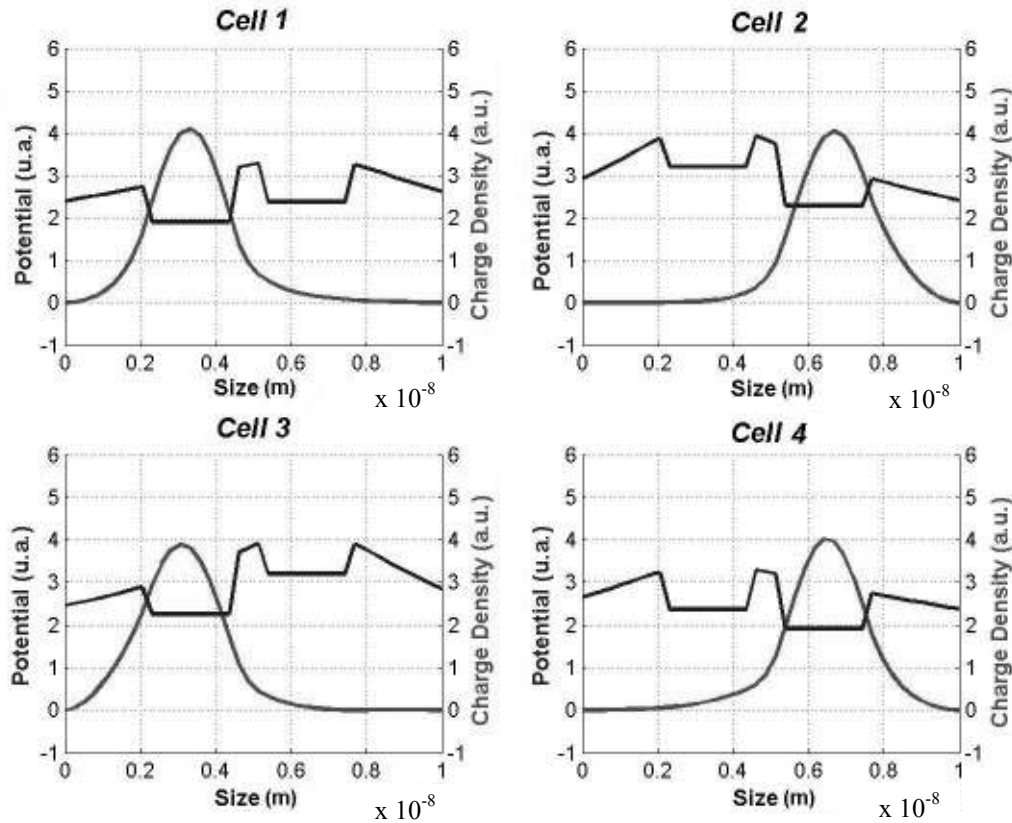


Fig. 6a. Initial potential distribution over the cells

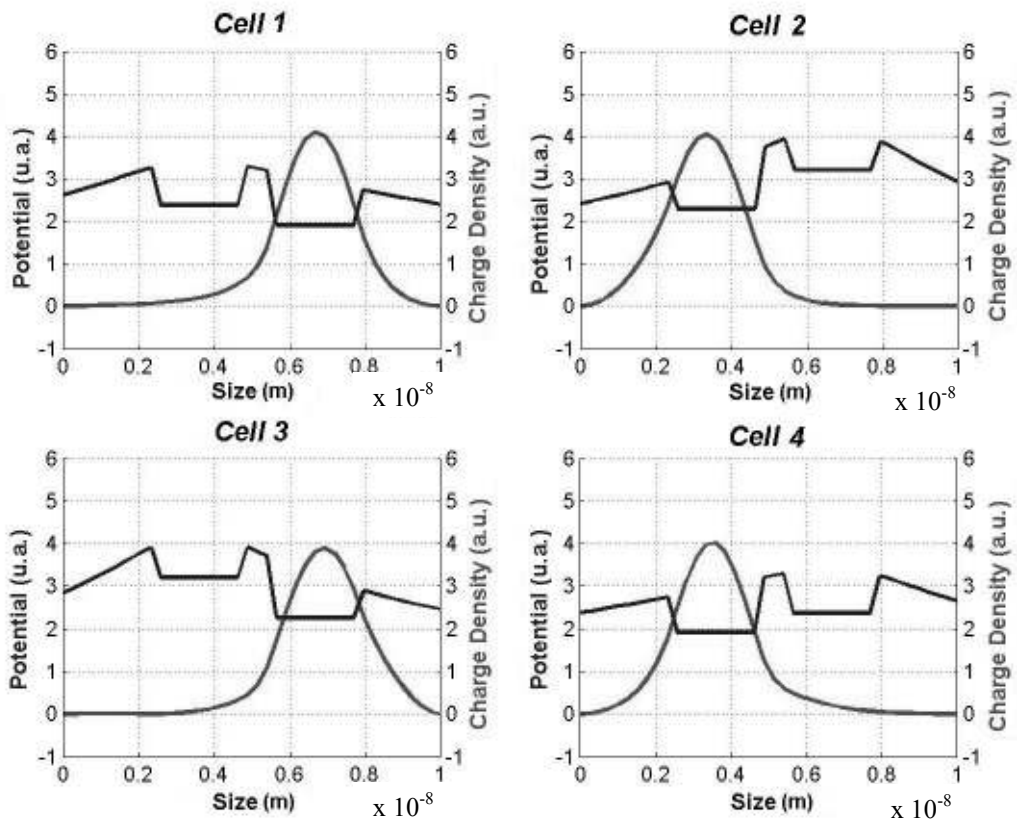


Fig. 6b. Final potential distribution over the cells

Fig 7 shows the state evolution for each cell (simultaneously observed) for about 2 ps.

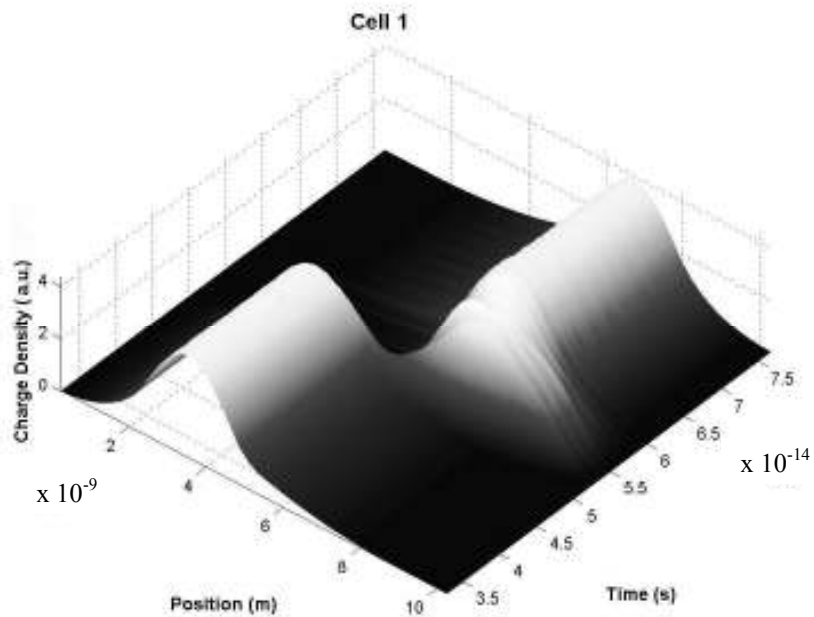


Fig. 7a. Tunneling process over the four cells – Cell 1

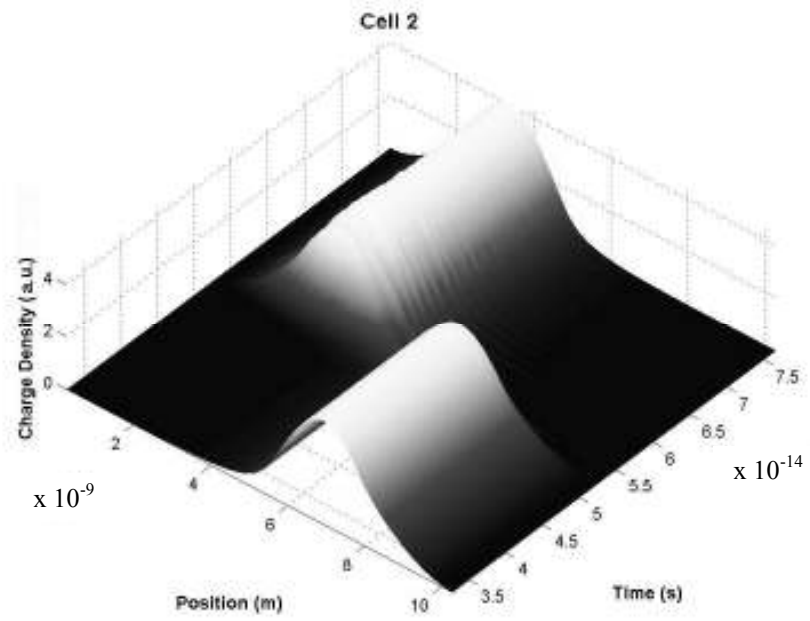


Fig. 7a. Tunneling process over the four cells – Cell 2

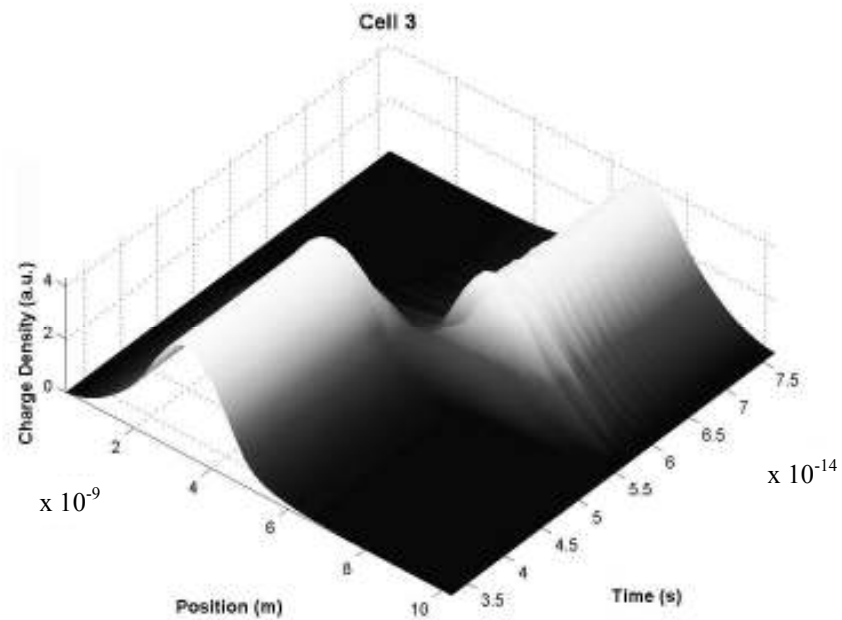


Fig. 7a. Tunneling process over the four cells – Cell 3

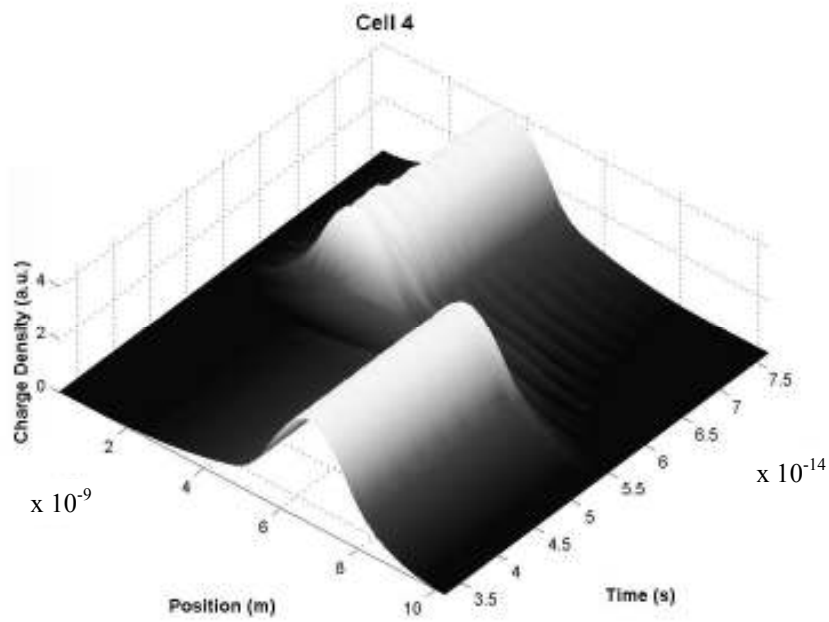


Fig. 7a. Tunneling process over the four cells – Cell 4

Simulation Results

The total charge on each dot of each cell has been plotted (Fig 8) against time. Near time T_a the first cell is affected by the external voltage source (Fig 5).

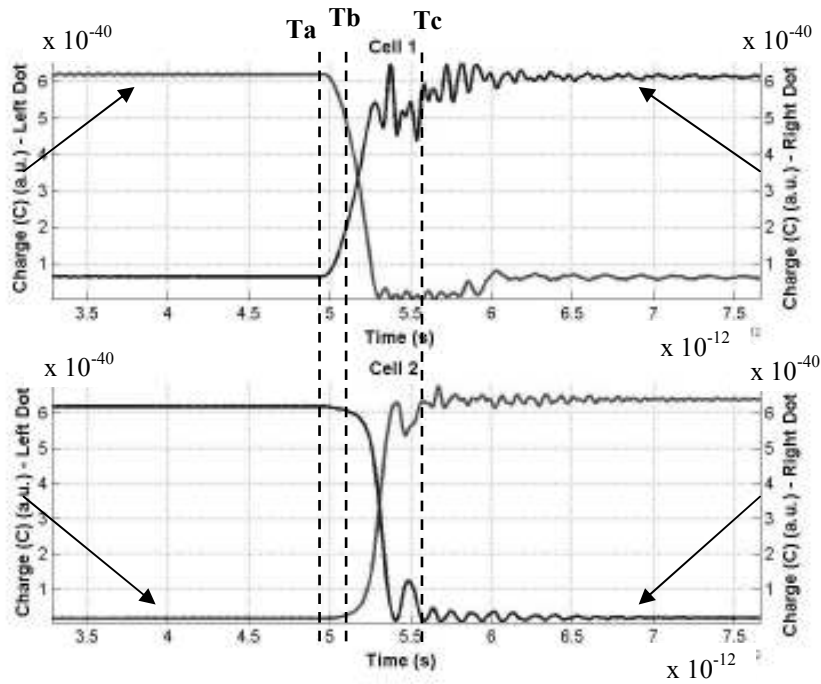


Fig. 8a. Charge distribution over time on the four cells (cells 1 and 2)

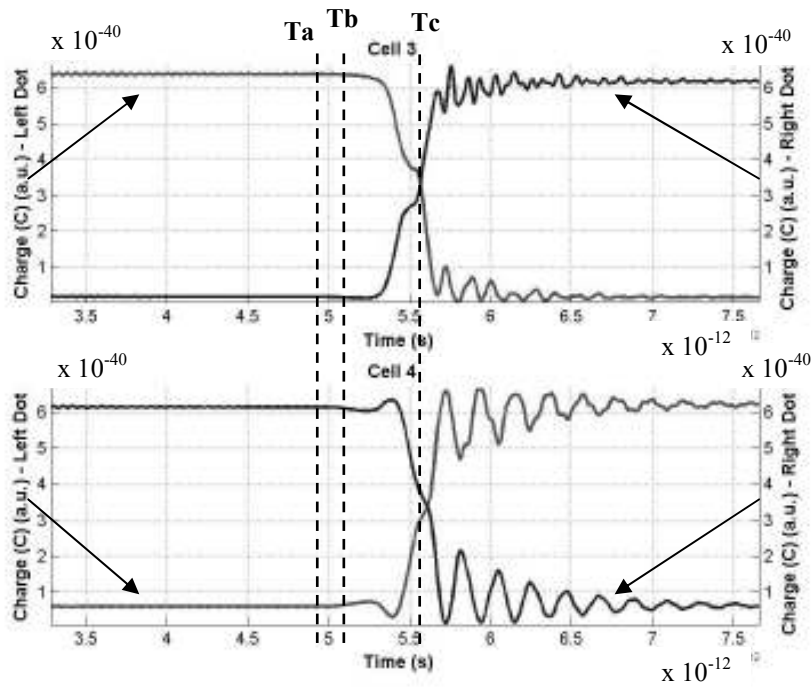


Fig. 8b. Charge distribution over time on the four cells (cells 3 and 4)

At T_b the tunneling effect is in its middle creating, as a reference, a time mark of the transition event. Extending these references to the other cells, the total tunneling time over the structure can be measured. The final tunneling has occurred near time T_c making the total process lasting $0,6 \times 10^{-12} \text{s}$.

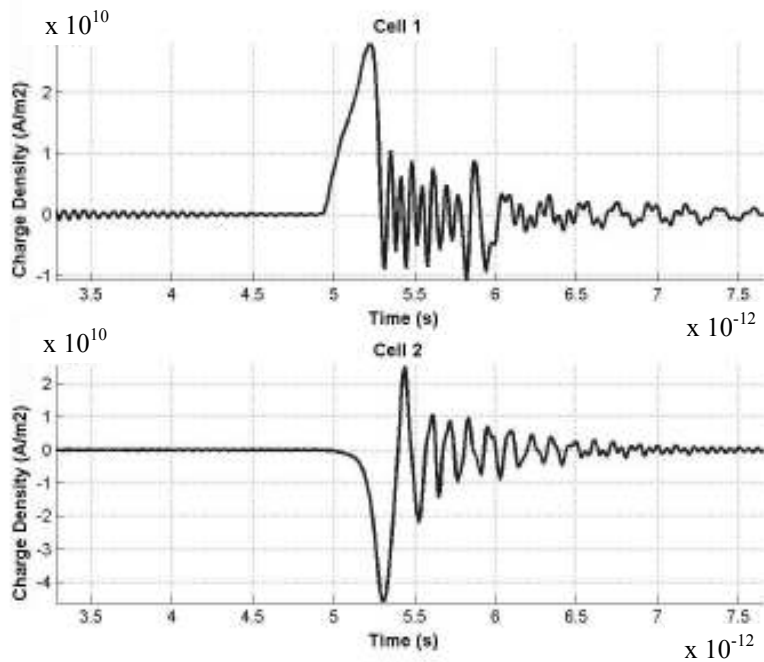


Fig. 9a. Current flow at the middle of each cell (cells 1 and 2)

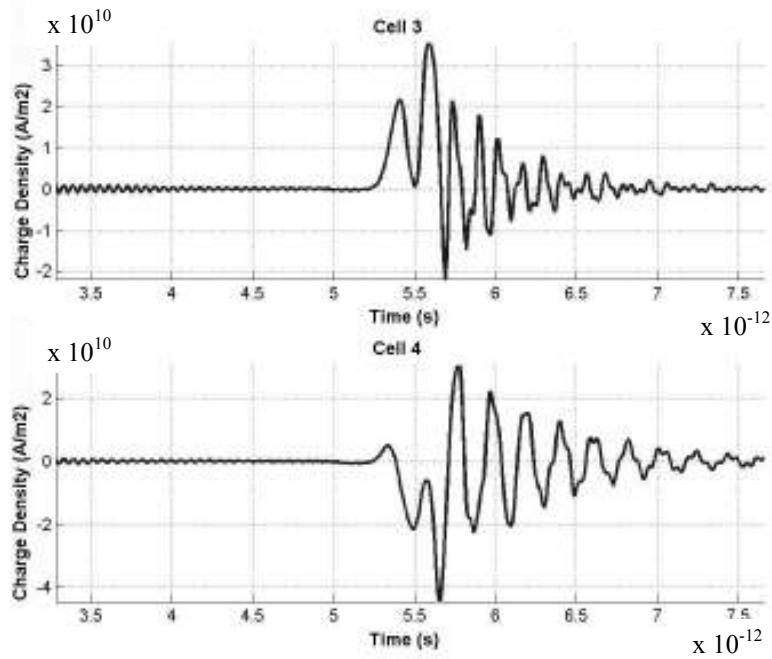


Fig. 9b. Current flow at the middle of each cell (cells 3 and 4)

Fig. 9 presents the current flow at the tunneling barrier in each cell, in the same time frame. Positive values means charge flowing from left to right. From this figure a full description of the state transition events is provided as well as an accurate determination of the respective settling times. The influence of the external voltage source over the first cell provokes the asymmetric behavior shown in figures 8 and 9.

Conclusion and future work

In this work an approach for realistic estimations of the time-behavior of state transitions and state propagation on CQD chains, which allows a better perception of the ultimate performance limits obtainable for such structures. The following studies shall tackle the influence of the number of cells in a chain over its own dynamic performance.

Acknowledgments

The authors gratefully acknowledge CAPES-Brasil, CNPq-Brasil and PADCT-Brasil for financial support.

References

1. Craig S. Lent, P. Douglas Tougaw, Wolfgang Porod, and Gary H. Bernstein, in *Quantum Cellular Automata*, Department of Electrical Engineering - University of Notre Dame - Notre Dame, IN 46556.
2. G. Timp, *Nanotechnology*, Springer-Verlag, New York (1999).
3. C. S. Lent et al, *Applied Physics Letters*, 62, 7, p.714-716, February (1993).
4. Lent C S and Tougaw P D – 1997 – *Proc. IEEE* – 85 541-57.
5. M. V. Batistuta, M. F. Stella, H. Biagi, and J. C. da Costa, in Electrochemical Society Proceedings, *Proceedings of the 19th Symposium on Microelectronics Technology and Devices*, SBMicro 2004, Porto de Galinhas, Brasil, PV 2004-03, p.151, The Electrochemical Society, Pennington, NJ (2004).
6. A. O. Orlov et al, *Applied Physics Letters*, 74, 19, May (1999).
7. MATLAB Version 6.5, from The Math Works Inc., Upper Saddle River, NJ, USA
8. M. V. Batistuta, M. F. Stella, H. Biagi, and J. C. da Costa, in *Electrochemical Society Proceedings*, PV 2004-03, p. 151, The Electrochemical Society, Pennington, NJ (2004)
9. M. V. Batistuta, M. F. Stella, H. C. de Biagi; J. C. da COSTA, *IEEE Proceedings of NanoSingapore 2006*, p.1-4, Piscataway, NJ (2006)