



**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE ENGENHARIA DE COMPUTAÇÃO
DISCIPLINA DE PROJETO FINAL**

**GUSTAVO HOEFLICH DAMASO DE OLIVEIRA
VALDIR ANTONIAZZI JÚNIOR**

RELATÓRIO TÉCNICO FINAL

DESIGN DE UM CIRCUITO INTEGRADO CONVERSOR ANALÓGICO-DIGITAL

**CURITIBA
4º BIMESTRE DE 2010**

GUSTAVO HOEFLICH DAMASO DE OLIVEIRA
VALDIR ANTONIAZZI JÚNIOR

DESIGN DE UM CIRCUITO INTEGRADO CONVERSOR ANALÓGICO-DIGITAL

Relatório técnico final apresentado à disciplina de Projeto Final do curso de Engenharia de Computação da PUC-PR como requisito à obtenção de nota parcial em projeto de conclusão de curso.

Orientador: Prof. Ivan Jorge Chueiri.

CURITIBA
4º BIMESTRE DE 2010

FOLHA DE APROVAÇÃO

DESIGN DE UM CIRCUITO INTEGRADO CONVERSOR ANALÓGICO-DIGITAL

Relatório técnico final apresentado à disciplina de Projeto Final do curso de Engenharia de Computação da Pontifícia Universidade Católica do Paraná como requisito à obtenção de nota parcial em projeto de conclusão de curso.

COMISSÃO EXAMINADORA

Prof. Dr. Luiz Augusto de Paula Lima Junior
Pontifícia Universidade Católica do Paraná

Prof. Mestre Ivan Jorge Chueiri
Pontifícia Universidade Católica do Paraná

Curitiba, ____ de _____ de 2010.

DEDICATÓRIA

Aos nossos pais,
pelo esforço incomensurável
em todos esses anos de estudo.

AGRADECIMENTOS

Ao nosso orientador, pela confiança depositada e pelo auxílio recebido em todas as etapas desse trabalho.

Aos Profs. Vilson Mognon e Afonso Ferreira Miguel, pela co-orientação e boa vontade em ceder um pouco de sua experiência e conhecimento.

Ao Prof. Luiz Augusto de Paula Lima Junior, pelas orientações referentes ao gerenciamento desse trabalho.

Ao amigo Carlos Eduardo Fusinato Magnani, pelas dúvidas esclarecidas, pelas dicas valiosas e pelas horas de conversa.

Aos nossos familiares, pela compreensão nos momentos de ausência, pelo apoio, carinho, paciência e dedicação.

EPÍGRAFE

Cem vezes todos os dias
lembro a mim mesmo que minha vida, interior e exterior,
depende dos trabalho de outros homens, vivos ou mortos,
e que devo esforçar-me a fim de devolver
na mesma medida que recebi.

Albert Einstein.

LISTA DE ILUSTRAÇÕES

Figura 1: Processo de quantização.	22
Figura 2: Conversor mais preciso.	23
Figura 3: Erro de quantização.	24
Figura 4: Diagrama de blocos do conversor A/D-SAR.	25
Figura 5: Fluxograma da lógica de controle.	26
Figura 6: Saída típica de um conversor D/A.	28
Figura 7: Símbolo lógico da porta inversora.	31
Figura 8: Resultado da simulação da porta inversora.	31
Figura 9: Símbolo lógico da porta NAND.	32
Figura 10: Resultado da simulação da porta NAND.	32
Figura 11: Símbolo lógico da porta AND.	33
Figura 12: Resultado da simulação da porta AND.	34
Figura 13: Símbolo lógico da porta NOR.	35
Figura 14: Resultado da simulação da porta NOR.	35
Figura 15: Símbolo lógico da porta OR.	36
Figura 16: Resultado da simulação da porta OR.	36
Figura 17: Símbolo lógico da porta NAND3.	37
Figura 18: Resultado da simulação da porta NAND3.	38
Figura 19: Símbolo lógico do flip-flop D.	39
Figura 20: Resultado da simulação do flip-flop D.	39
Figura 21: Esquema lógico do shift register.	40
Figura 22: Resultado da simulação do shift register.	41
Figura 23: Esquema lógico do registrador de aproximações sucessivas.	42
Figura 24: Fluxograma da lógica de controle.	43
Figura 25: Resultado obtido a partir do esquemático construído.	44
Figura 26: Símbolo lógico do conversor D/A R-2R.	44
Figura 27: Resultado obtido a partir do esquemático construído.	45
Figura 28: Símbolo lógico do amplificador operacional.	46
Figura 29: Resultado obtido a partir do esquemático construído.	46

LISTA DE TABELAS

Tabela 1: Sensibilidade de um conversor.....	23
Tabela 2: Tabela verdade do comparador (níveis lógicos).....	29
Tabela 3: Tabela verdade da porta inversora.....	30
Tabela 4: Tabela verdade da porta NAND.....	32
Tabela 5: Tabela verdade da porta AND.....	33
Tabela 6: Tabela verdade da porta NOR.....	34
Tabela 7: Tabela verdade da porta OR.....	36
Tabela 8: Tabela verdade da porta NAND3.....	37
Tabela 9: Tabela verdade do flip-flop D.....	39
Tabela 10: Tabela verdade do flip-flop D.....	40
Tabela 11: Tabela verdade do comparador.....	45
Tabela 12: Tabela verdade do comparador (níveis lógicos).....	45

RESUMO

Esse documento é o relatório técnico final do projeto de um circuito integrado desenvolvido como projeto de conclusão de curso e com o objetivo de aprender os fundamentos de projeto de circuito integrados e produzir material de base para o Laboratório de Microeletrônica. O resultado final do trabalho foi a modelagem, a simulação e a criação do layout de um conversor analógico para digital. Isto foi alcançado com a utilização de ferramentas para descrição de hardware em VHDL, para simulação eletrônica com SPICE e para representação física do circuito. O documento é constituído por um breve histórico da microeletrônica, das tecnologias envolvidas, as motivações e justificativas e o detalhamento do projeto.

SUMÁRIO

1	BREVE HISTÓRIA DA MICROELETRÔNICA.....	12
2	A MICROELETRÔNICA NO BRASIL.....	14
3	TECNOLOGIAS DE MICROELETRÔNICA.....	16
3.1	EDA.....	16
3.2	VHDL.....	17
3.3	SPICE.....	18
3.4	GDSII.....	19
4	MOTIVAÇÃO E JUSTIFICATIVAS.....	20
5	DETALHAMENTO DO PROJETO.....	22
5.1	INTRODUÇÃO.....	22
5.2	TRABALHO DESENVOLVIDO.....	24
5.2.1	Lógica de controle.....	25
5.2.2	Conversor Digital para Analógico com rede R-2R.....	27
5.2.3	Amplificador operacional (Comparador).....	28
6	PROCEDIMENTOS DE TESTE E RESULTADOS OBTIDOS.....	30
6.1	CIRCUITOS BÁSICOS.....	30
6.1.1	Porta inversora (NOT).....	30
6.1.2	Porta NAND.....	31
6.1.3	Porta AND.....	33
6.1.4	Porta NOR.....	34
6.1.5	Porta OR.....	35
6.1.6	Porta NAND3.....	37
6.1.7	Flip-flop tipo D com clear assíncrono.....	38
6.1.8	Shift Register.....	40
6.2	CONVERSOR A/D-SAR.....	41
6.2.1	Lógica de controle.....	42
6.2.2	Conversor D/A R-2R.....	44
6.2.3	Amplificador operacional (Comparador).....	45
7	CONCLUSÃO.....	47
	REFERÊNCIAS BIBLIOGRÁFICAS.....	49
	ANEXO A.....	51

PORTA LÓGICA INVERSORA (NOT)	51
Símbolo lógico e diagrama eletrônico	51
Código SPICE	52
Resultado da simulação SPICE	54
Layout.....	55
PORTA LÓGICA NAND	56
Símbolo lógico e diagrama eletrônico	56
Código SPICE	57
Resultado da simulação SPICE	60
Layout.....	61
PORTA LÓGICA AND	62
Símbolo lógico e diagrama eletrônico	62
Código SPICE	63
Resultado da simulação SPICE	66
Layout.....	67
PORTA LÓGICA NOR	68
Símbolo lógico e diagrama eletrônico	68
Código SPICE	69
Resultado da simulação SPICE	72
Layout.....	73
PORTA LÓGICA OR	74
Símbolo lógico e diagrama eletrônico	74
Código SPICE	75
Resultado da simulação SPICE	78
Layout.....	79
PORTA LÓGICA NAND 3 ENTRADAS	80
Símbolo lógico e diagrama eletrônico	80
Código SPICE	81
Resultado da simulação SPICE	84
Layout.....	85
FLIP-FLOP TIPO D COM CLEAR ASSÍNCRONO	86
Símbolo e diagrama lógico	86
Código SPICE	87
Resultado da simulação SPICE	90
Layout.....	91

SHIFT-REGISTER 8 BITS	92
Diagrama lógico	92
Código SPICE	93
Resultado da simulação SPICE	97
Layout	98
LÓGICA DE CONTROLE SAR.....	99
Diagrama lógico	99
Código SPICE	100
Resultado da simulação SPICE	105
Layout.....	106
CONVERSOR DIGITAL ANALÓGICO R2R (DAC).....	107
Diagrama eletrônico	107
Código SPICE	108
Resultado da simulação SPICE	110
Layout.....	111
AMPLIFICADOR OPERACIONAL.....	112
Símbolo lógico e diagrama eletrônico	112
Código SPICE	113
Resultado da simulação SPICE	115
Layout.....	116
CONVERSOR ANALÓGICO P/ DIGITAL.....	117
Layout do circuito integrado.....	117

1 BREVE HISTÓRIA DA MICROELETRÔNICA

O transistor foi originalmente inventado por John Bardeen e Walter Brattain em 1947 no *Bell Laboratories*. Avanços foram feitos rapidamente por William Shockley e a produção do primeiro circuito integrado por Jack Kilby na *Texas Instruments* começou em 1958 e, subsequentemente, por Robert Noyce na *Fairchild Inc.* O campo da microeletrônica havia começado seu avanço triunfante.

Antes da criação do transistor, circuitos eletrônicos eram limitados a projetos simples compostos quase que inteiramente por válvulas termiônicas. Eletrônicos eram utilizados quase exclusivamente na difusão e recepção de rádio e televisão. As válvulas foram usadas em larga escala nesse campo até os anos 1960. Mesmo antes do transistor violar a fortaleza da válvula, ela era utilizada como um dispositivo discreto e em circuitos simples em nichos de mercados tecnológicos, como eletrônica médica (aparelhos auditivos, por exemplo) e em equipamentos militares e tecnologia espacial. Isto proporcionou os meios de financiar o primeiro transistor útil. Mais tarde, o Japão foi o primeiro a começar a utilizar semicondutores em aplicações comerciais, tal como rádios portáteis, gravadores de fita e eletrônicos de entretenimento. A produção em massa de transistores havia começado, e uma correspondente e drástica diminuição nos preços de produtos eletrônicos foi observada.

A primeira calculadora de mão acessível foi criada por volta de 1972. Uma invasão de eletrônicos nos escritórios e administrações logo aconteceu. Ambientes onde rudimentares e centralizados processamentos de dados eram realizados por grandes computadores centrais foram especialmente abertos para novos aparelhos eletrônicos. Informática e aparelhos digitais relacionados desempenharam um papel menor comparado ao rádio e à televisão, mas que rapidamente mudou com a invenção e ampla aplicação da tecnologia MOS e subsequentemente CMOS. Essa tecnologia forma a base dos circuitos altamente integrados (Large Scale Integration – LSI).

Um dos maiores acontecimentos em 1972 foi a invenção do microprocessador na Intel. A Intel tinha menos que uma dúzia de funcionários naquela época. Gordon Moore, diretor do departamento de Pesquisa e Desenvolvimento da Fairchild Semiconductor Corp. e mais tarde presidente e diretor executivo da Intel profetizou, naquele tempo, em 1965, o que hoje é conhecida como a tão falada Lei de Moore: “*O número de transistores em um chip dobrará a cada ano!*”. Ao criar sua hoje tão famosa relação, Moore disse que sua declaração deveria ser válida pelos próximos 10 anos. Em 1975, Moore novamente repetiu sua afirmação, mas com mais precisão, depois de a tecnologia de semicondutores ter alcançado

uma certa maturidade: “*O número de transistores em cada chip dobrará a cada 18 meses!*”; o que notavelmente ainda descreve a situação atual. Não está claro se este crescimento exponencial continuará a persistir, pode persistir por mais tempo ou pode em breve começar a fraquejar. Em ambos os casos, nos próximos anos não haverá um declínio nesta tendência, pelo menos os limites físicos não serão ainda alcançados no futuro próximo.

2 A MICROELETRÔNICA NO BRASIL

Desde a década de 1950 algumas universidades brasileiras, como o ITA, a partir de 1953, e o IFUSP, no início dos anos 1960, tiveram atividades em semicondutores e dispositivos. Sempre acompanhamos de perto o desenvolvimento da área e o país contribuiu de alguma forma com o desenvolvimento da mesma.

Já bem no início da história dos dispositivos semicondutores, em meados dos anos 1960, a Philco instala fábrica de diodos e transistores em São Paulo. Isto é, apenas 10 anos após a liberação e disseminação da tecnologia pela Bell Laboratories, o Brasil iniciou atividades industriais de microeletrônica. Esta atividade motivou a criação do Laboratório de Microeletrônica (LME) da Escola Politécnica da Universidade de São Paulo (EPUSP), que começou a operar em 1970. Este laboratório pioneiro foi responsável pelo desenvolvimento de várias tecnologias de microeletrônica, pela formação de um número considerável de profissionais e por várias iniciativas acadêmicas e industriais, incluindo o desenvolvimento do primeiro circuito integrado no país em 1971 e o projeto e construção do primeiro circuito integrado com tecnologia MOS na América Latina em 1973.

Nos anos seguintes, novos laboratórios de microeletrônica surgem em outras universidades espalhadas pelo país, a citar, na UNICAMP, na UFRGS, na UFPE e um novo laboratório na EPUSP, o Laboratório de Sistemas Integrados (LSI). Com atuação mais ampla que apenas microeletrônica, o LSI realizou pesquisa de alto nível nas atividades de obtenção e caracterização de silicetos, processos de plasma para deposição de filmes finos e de corrosão e processos de limpeza e oxidação de silício. Em seguida, o grupo concentrou esforços no desenvolvimento de micro-sensores de silício.

Os cinco grupos acima constituem os grupos universitários com instalações completas de microeletrônica de silício. Além destes, existem grupos que atuam em temas específicos associados a processos de microeletrônica, como por exemplo o grupo de Engenharia Elétrica da UnB e o Departamento de Física do ITA.

Esse é um breve histórico das iniciativas acadêmicas. No entanto, essas atividades não fazem muito sentido se não forem acompanhadas por aplicações industriais. A primeira iniciativa industrial no ramo da microeletrônica no Brasil foi uma fábrica da Philco em 1966 destinada a produção de diodos e transistores, conforme citado. De lá pra cá, a fábrica fechou parceria para ampliação de seu parque tecnológico e expansão de seu setor de atuação, foi transferida para o interior de Minas Gerais e foi adquirida por outros grupos até o descontinuação total de todas as suas atividades em 2000. Outra iniciativa no setor foi a

empresa Transit, também no interior de Minas Gerais, criada em 1974, tendo sido fechada ainda no final dos anos 1970. Empresas internacionais também se interessaram em se instalar no Brasil, como a Icotron, do grupo Siemens, com fábrica em Gravataí-RS, e a Semicron, com fábrica em Cotia-SP, ambas com uma linha completa de difusão de diodos e tiristores de potência. Esta última é a única entre as citadas que ainda continua com sua linha completa de fabricação. A Icotron doou seus equipamentos de difusão para a UFRGS em 1998. Outras empresas tiveram fábricas de montagem e encapsulamento de componentes semicondutores no país, entre elas Texas Instruments, Fairchild, Philips e Sanyo. No entanto, fecharam suas operações de produção de componentes semicondutores no início da década de 1990 com a ampla e súbita abertura do nosso mercado.

É possível observar que desde a década de 1960 tivemos atividades de microeletrônica e que havia um bom desenvolvimento tecnológico na segunda metade da década de 1970. Inclusive, este estágio era superior ao dos países hoje conhecidos como tigres asiáticos, afirma o Prof. Dr. Jacobus W. Swart, professor titular da Universidade Estadual de Campinas, especialista em materiais e componentes semicondutores, desde maio de 2007 licenciado para ocupar o cargo de Diretor do CTI, Centro de Tecnologia da Informação Renato Archer, naquela mesma cidade. *“Porém, durante as duas décadas subseqüentes a área passou por um estágio de estagnação, constituindo um paradoxo para a retórica de política de reserva de mercado de informática.”* - mais uma vez afirma o Prof. Jacobus Swart - *“Investiu-se recursos volumosos em uma fábrica de máscaras enquanto que as indústrias clientes não se instalaram conforme planejado”*.

Hoje o país apresenta um déficit comercial de vários bilhões de dólares anuais em componentes eletrônicos. Finalmente, a formação de recursos humanos capaz de projetar circuitos integrados e a instalação de uma ou mais fábricas de circuitos integrados é uma necessidade real para equilibrar a balança comercial e promover o desenvolvimento econômico do país.

3 TECNOLOGIAS DE MICROELETRÔNICA

Um sistema eletrônico pode ser uma placa de circuito impresso ou um circuito integrado. Um circuito integrado pode ser um sistema em um chip (SoC), um circuito integrado de aplicação específica (ASIC) ou uma matriz de portas programáveis em campo (FPGA).

O primeiro circuito integrado, chamado de oscilador de deslocamento de fase, foi criado por Jack Kilby da Texas Instruments em 12 de setembro de 1958 e consistia em apenas um transistor, um capacitor e três resistores. Os primeiros microprocessadores foram projetados manualmente com a ajuda de grandes bancadas de corte e uma grande dose de esforço de elaboração. Hoje, um circuito integrado típico pode chegar facilmente a conter várias centenas de milhões de transistores e quilômetros de fios de interconexões.

Como é possível projetar circuitos com tantos milhões de transistores? É o que procuramos demonstrar nesta seção.

3.1 EDA

Essa capacidade de integração em altíssima escala (VLSI, do inglês very large-scale integration) foi permitida pelo uso moderno de muitas tecnologias e aplicações de automatização de projeto eletrônico (EDA, do inglês Electronic Design Automation). Um projeto de circuito com centenas de milhões de transistores sem o auxílio de computadores é inconcebível. O rápido desenvolvimento da eletrônica de semicondutores só foi possível porque os computadores construídos de semicondutores se tornaram mais eficientes e rápidos, que novamente tornaram possível o desenvolvimento de ainda mais eficientes circuitos integrados. A tecnologia de semicondutores e a automatização de projetos eletrônicos estão intimamente interligadas e dependentes uma da outra. Progresso em EDA leva a progresso em tecnologia de semicondutores e vice versa.

EDA compreende um conjunto de ferramentas de software e hardware de co-projeto, síntese, verificação e teste que verifica o projeto lógico e leva-o aos estágios de verificação e síntese ao nível de transistores para eventualmente produzir-se um projeto físico em representação geométrica. EDA, portanto, pode ser vista como uma coleção de ferramentas de automatização de projeto e automatização de testes que automatiza tarefas de projeto e teste, respectivamente.

A história de EDA começa no início da década de 1960 depois da introdução dos

circuitos integrados por J. Kilby em 1958, já mencionado. Nesse estágio muitíssimo inicial, o projeto lógico e físico desses circuitos, sobretudo, era criado à mão paralelamente. Projeto lógico construído de placas de circuitos que imitavam o projeto físico do circuito integrado era construído para simular e verificar se o circuito integrado funcionaria conforme esperado antes da fabricação. Durante esse período, pesquisa e desenvolvimento em EDA era típico de grandes corporações, como Bell Labs, Hewlett Packard, IBM, Intel e Tektronix.

O primeiro marco fundamental em EDA veio na forma de programas para simulação de circuito e verificação de layout. Várias linguagens de simulação proprietárias e modelos foram propostos. Os modelos SPICE (comumente referidas como simulação SPICE nos dias de hoje) foram usados para verificar se o projeto lógico especificado no nível de transistores se comportaria conforme as especificações funcionais. Isso removeu a necessidade de construir placas de circuito com fios.

Conforme podemos observar, atualmente nenhuma área em um ambiente industrial consegue seguir em frente sem substancial auxílio de computadores.

3.2 VHDL

VHDL é uma linguagem de descrição de hardware. Ela descreve o comportamento de um circuito ou sistema eletrônico, a partir do qual o circuito físico pode ser obtido.

VHDL representa VHSIC Hardware Description Language. VHSIC, por sua vez, é um acrônimo para Very High Speed Integrated Circuits, uma iniciativa financiada pelo Departamento de Defesa dos Estados Unidos da América na década de 1980 que levou à criação da VHDL. A VHDL foi criada visando simulação, modelagem e documentação, mas acabou recebendo mais tarde a possibilidade de síntese (transformação em portas lógicas), com o objetivo de se automatizar o projeto de circuitos.

Sua primeira versão foi VHDL 87, posteriormente atualizada para VHDL 93. VHDL foi a original e a primeira linguagem de descrição de hardware a ser padronizada pelo Instituto de Engenheiros Elétricos e Eletrônicos (IEEE).

A VHDL é destinada para a **síntese e simulação de circuitos digitais**. A motivação fundamental para utilizar VHDL é que VHDL é um padrão, linguagem independente de tecnologia e fornecedor e, conseqüentemente, facilmente portátil e reutilizável.

As duas principais aplicações imediatas de VHDL estão no campo de Programmable Logic Devices (PLDs) e Field Programmable Gate Arrays (FPGAs) e no campo de ASICs (Application Specific Integrated Circuits). Depois que o código VHDL foi

escrito, ele pode ser usado tanto para implementar o circuito em um dispositivo programável (da Altera, por exemplo) ou podem ser submetidas a uma *foundry* para a fabricação de um *chip*. Atualmente, muitos *chips* comerciais complexos (microcontroladores, por exemplo) são projetados usando essa abordagem.

Uma nota final sobre VHDL é que, contrariamente aos programas de computador comuns, que são seqüenciais, suas declarações são inerentemente concorrentes (paralelas). Por esse motivo, VHDL é normalmente referida como um código ao invés de um programa.

3.3 SPICE

Simulação de circuito é uma parte importante de qualquer processo de design. Ao simular seus circuitos, é possível detectar erros no início do processo e evitar dispendioso e demorado reprojetado. Você pode também facilmente trocar componentes para avaliar projetos com uma enorme variedade de materiais.

Simulation Program with Integrated Circuit Emphasis (Programa de Simulação com ênfase em Circuitos Integrados), ou SPICE, tem sido utilizado por mais de 30 anos para prever com precisão o comportamento de circuitos eletrônicos. A implementação original do SPICE foi desenvolvida no campus da Universidade da Califórnia em Berkeley nos anos 1960. SPICE foi desenvolvido em grande parte como um derivado do CANCER (Computer Analysis of Non-linear Circuits Excluding Radiation), também desenvolvido pela mesma universidade.

Ao longo dos anos, as muitas revisões do SPICE viram melhorias tanto em precisão como em velocidade. Além destas melhorias, as adições à linguagem têm permitido a simulação e modelagem de circuitos integrados mais complexos, incluindo MOSFETs.

A primeira versão amplamente utilizada do SPICE foi anunciada em Waterloo, Canadá, em 1973. Pouco tempo depois SPICE foi aprovado por quase todas as principais instituições de engenharia na América do Norte. SPICE evoluiu para o nível acadêmico e para a indústria para simulação de **circuitos analógicos e de sinal misto**. Ao longo dos anos algoritmos de simulação adicionais, modelos de componentes, correções de erros e recursos foram adicionados ao programa. Ainda hoje SPICE é o simulador de circuitos mais utilizado no mundo.

3.4 GDSII

Projeto físico refere-se a todos os passos de síntese que converte uma representação de circuito em termos de transistores em uma representação geométrica em termos de polígonos e seu formato. A representação geométrica, também chamada de **layout**, é usada para projetar máscaras e então manufaturar um chip.

O formato GDSII, abreviação de Graphics Database System (sistema de banco de dados gráfico), é um formato de arquivo de dados que é o padrão industrial de fato para troca de dados de layout de um circuito integrado. É um formato de arquivo binário para representar formas geométricas planares, rótulos de texto e outras informações sobre o layout em formato hierárquico. Os dados podem ser usados para reconstruir toda ou parte da arte (layout) de um circuito integrado, para ser usada em compartilhamento de layouts, transferência de layout entre diferentes ferramentas ou fabricação de máscaras.

Inicialmente, GDSII foi desenvolvido para controlar a plotagem de máscaras de circuito integrado. Apesar de seu limitado conjunto de recursos e baixa densidade de dados, ele se tornou o formato convencional da indústria para transferir dados do layout do circuito integrado entre ferramentas de projeto de diferentes fornecedores, todas elas operadas com formatos de dados proprietários. Ele foi originalmente desenvolvido pela *Calma Company* em meados dos anos 1970 para seu programa de desenvolvimento de layout Graphics Data System. Hoje o formato é propriedade da *Cadence Design Systems*.

Os objetos contidos em um arquivo GDSII são agrupados atribuindo-os atributos numéricos incluindo o número da camada, posição e tipo do dado.

Arquivos GDSII são, normalmente, o produto final do ciclo de projeto de circuitos integrados.

4 MOTIVAÇÃO E JUSTIFICATIVAS

Nas seções anteriores, conhecemos um pouco da história da microeletrônica e o grande desenvolvimento pelo qual passou nesse curto espaço de tempo. Fomos capazes de perceber a importância da microeletrônica no cenário mundial, o que o seu desenvolvimento representa para o Brasil e, portanto, seu papel significativo como ferramenta de crescimento econômico para um país.

O desenvolvimento geral da eletrônica requer o uso de circuitos integrados de aplicação específica em grande escala. Para o desenvolvimento de microssistemas completos necessita-se tanto da disponibilidade das técnicas de microfabricação como da capacitação em projeto de CI, já que os microssistemas são compostos por *chips* contendo sensores ou atuadores integrados com o circuito de controle ou processamento de sinal.

A área e mercado de microeletrônica e microssistemas cresceram enormemente, participando direta ou indiretamente de todas as atividades humanas. Adicionalmente, é graças a ela que todas as outras áreas do conhecimento humano conseguem avançar. A área é também extremamente multidisciplinar, envolvendo conhecimentos e profissionais de engenharia eletrônica, engenharia e ciência de materiais, física, química, biologia, medicina e ciência da computação. Até mesmo ciências humanas, tais como economia, sociologia, história e educação tem muito a ver com a área, tendo em vista as enormes consequências destas tecnologias sobre a economia, defesa, segurança, empregos, vida social, educação, saúde, etc.

Em vista do papel desempenhado pela microeletrônica no cenário atual, entendemos a importância de conhecer as técnicas de projeto de circuito integrados como ferramenta para nosso desenvolvimento e aperfeiçoamento profissional. Este o motivo principal de escolhermos a microeletrônica. Nossas perspectivas futuras são desenvolvermo-nos, de fato, como projetistas de circuitos integrados e aprimorar conhecimentos através de projetos de pós-graduação (mestrado e doutorado) para atuar no setor industrial ou acadêmico e auxiliar no desenvolvimento tecnológico do país.

Adicionalmente, desde meados de 2008 a PUCPR inaugurou um laboratório de microeletrônica, iniciando suas atividades com o ano letivo de 2009. O laboratório montado conta com recursos de ponta no que se refere a projeto de circuitos integrados, possuindo ferramentas para desenvolvimento de circuitos digitais utilizando VHDL e outras linguagens de descrição de hardware suportadas pelo ambiente QUARTUS II da ALTERA, simulação SPICE através do ambiente HSPICE da SYNOPSIS para simulação de circuitos analógicos e

de sinal misto e um conjunto de 12 ferramentas MENTOR GRAPHICS para projeto lógico e físico de circuitos integrados. No entanto, o referido laboratório, com exceção do presente projeto, ainda não rendeu frutos. No início oficial de suas atividades em 2009, seu primeiro projeto foi a construção de um microprocessador RISC de 8 bits, ainda inacabado e suspenso desde o início do ano corrente. É de grande importância para os cursos de Engenharia de Computação, Elétrica e Eletrônica, inicialmente, o desenvolvimento do presente projeto, porque mostrará à PUCPR o potencial educacional, social e econômico que a microeletrônica representa. Investimentos no Laboratório de Microeletrônica possibilitarão a inserção da PUCPR no cenário da microeletrônica, onde estão presentes as maiores e mais importantes instituições de pesquisa do país. Posteriormente, novos grupos de graduação e pós-graduação poderão beneficiar-se com os resultados obtidos pelo grupo de microeletrônica, devido ao seu caráter extremamente multidisciplinar, a citar, o grupo de pós-graduação em engenharia biomédica e seu laboratório de engenharia de reabilitação, entre outros grupos. Com iniciativas e incentivo à microeletrônica, a PUCPR poderá pouco a pouco consolidar-se como uma grande potência em microeletrônica. Para tal, é preciso que ratifiquemos a importância e o potencial deste laboratório. Acreditamos, sinceramente, que este projeto cumpre este papel e com este objetivo foi ele idealizado por alunos e orientador.

Finalmente, um terceiro aspecto que justifica a escolha da equipe pela microeletrônica é a variedade de disciplinas estudadas no curso de Engenharia de Computação que ela envolve. Para realizarmos o presente projeto foram empregados os conhecimentos de disciplinas como lógica matemática, eletricidade básica, circuitos elétricos, sistemas digitais, microprocessadores, eletrônica e arquitetura de computadores, além de outras disciplinas do qual os conhecimentos servem de base para as já mencionadas ou foram requeridos em grau de menor relevância, a citar, cálculo diferencial e integral, física, técnicas de programação, sistemas operacionais, resolução de problemas em engenharia e engenharia de software. Adicionalmente, conhecimentos em tecnologias de microeletrônica (linguagens de descrição de hardware, simulação de circuitos e representação física de um CI) foram necessários. No total foram diretamente empregados o conhecimento de 15 disciplinas, além de outras 15 indiretamente e de 3 tecnologias específicas ao desenvolvimento de circuitos integrados não abrangidas pelo curso, que exigiu fosse buscado extracurricularmente com grande esforço por parte dos integrantes.

Estes constituem os principais motivos à execução deste projeto. Passemos, então, ao detalhamento do mesmo.

5 DETALHAMENTO DO PROJETO

5.1 INTRODUÇÃO

Os sinais podem ser de duas naturezas: analógicos ou digitais.

Nos **sinais analógicos**, os níveis variam sobre uma faixa contínua, assumindo infinitos valores. Podemos citar, por exemplo, a temperatura e a voz. Nos **sinais digitais**, os códigos são construídos a partir de um número finito de símbolos, isto é, a variável que carrega a informação varia discretamente com o tempo.

O processo de transformação de uma amostra analógica contínua em uma forma digital é chamado de **quantização**. A figura abaixo mostra o processo de conversão. No eixo X temos a tensão de entrada analógica e no eixo Y o código digital correspondente.

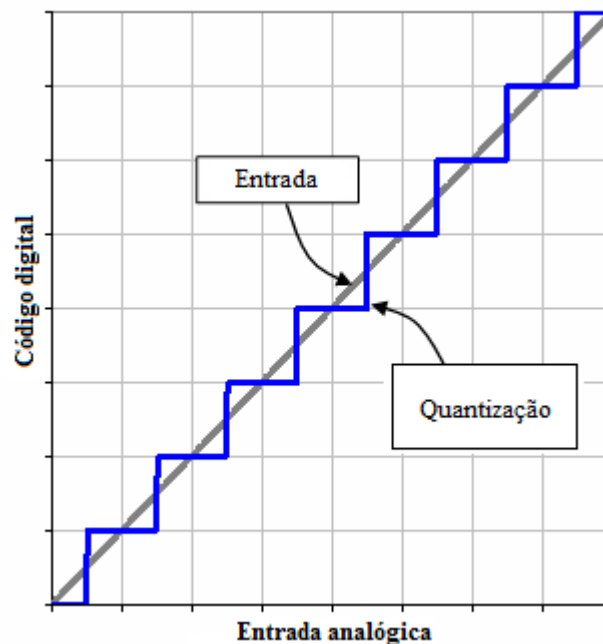


Figura 1: Processo de quantização.

Quanto maior a quantidade de níveis digitais mais preciso será o conversor, obedecendo a seguinte equação:

$$Q = \frac{V_{cc}}{2^n}, \text{ onde:}$$

Q (quantum): é tamanho da quantidade analógica que corresponde a um nível de saída;

n: é número de bits do conversor.

Fica evidente que quanto menor Q, ou seja, quanto maior o número de bits do

conversor, maior resolução (ou maior sensibilidade) terá o conversor. Considerando uma tensão analógica $V_{cc} = 5V$, temos:

n	Q	nº de níveis	Níveis
1	2,50 V	2	Nível 0: 0,00 a 2,50 V Nível 1: 2,51 a 5,00 V
2	1,25 V	4	Nível 0: 0,00 a 1,25 V Nível 1: 1,26 a 2,50 V Nível 2: 2,51 a 3,75 V Nível 3: 3,76 a 5,00 V
3	0,625 V	8	Nível 0: 0,000 a 0,625 V Nível 1: 0,626 a 1,250 V Nível 2: 1,251 a 1,875 V Nível 3: 1,876 a 2,500 V Nível 4: 2,501 a 3,125 V Nível 5: 3,126 a 3,750 V Nível 6: 3,751 a 4,375 V Nível 7: 4,376 a 5,000 V
(...)	(...)	(...)	(...)
8	0,01953125	256	(...)

Tabela 1: Sensibilidade de um conversor.

Quando comparada com a figura 1, a imagem abaixo evidencia essa constatação.

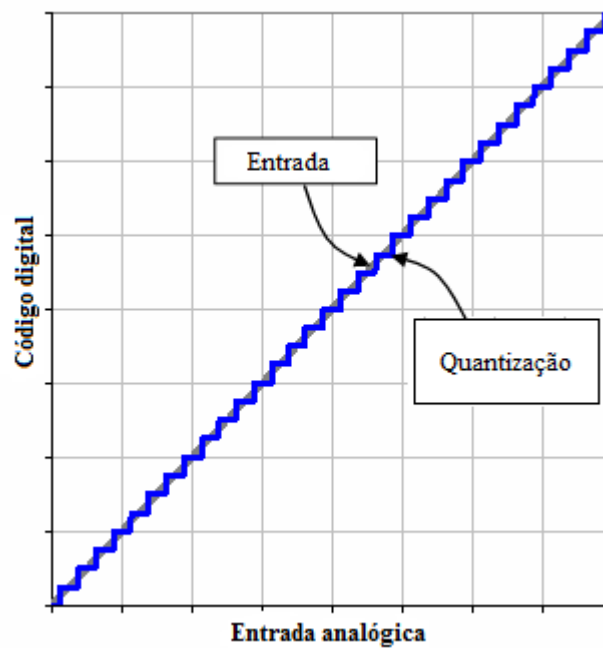


Figura 2: Conversor mais preciso.

A diferença entre o valor analógico atual e o valor digital quantizado é chamada de **erro de quantização**. A imagem abaixo ilustra esse problema.

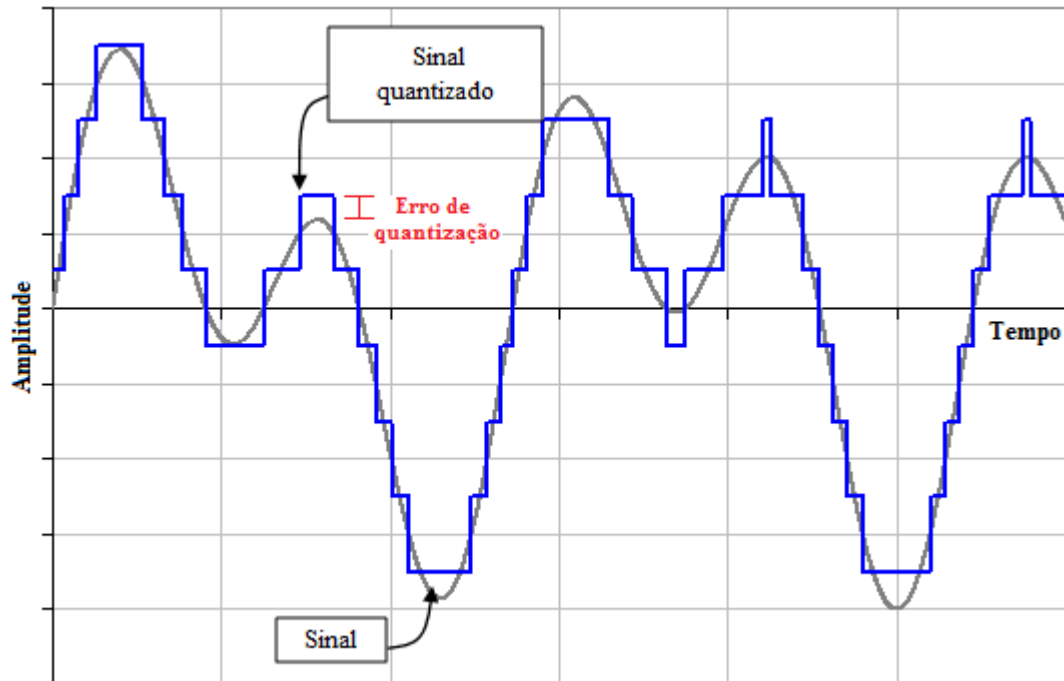


Figura 3: Erro de quantização.

Esse problema não é passível de ser eliminado completamente, visto que sempre haverá perda de informação ao se converter uma mensagem com infinitos níveis de tensão a uma codificação com níveis limitados. É possível, apenas, minimizar o erro ao máximo para faixas toleráveis adequadas à aplicação. Devido a isto, devemos estar conscientes de que sempre haverá discrepâncias entre o valor analógico medido e o valor digital quantizado.

5.2 TRABALHO DESENVOLVIDO

No decorrer da disciplina, construímos um conversor analógico para digital (A/D) por aproximações sucessivas (SAR, do inglês Successive Approximation Register).

Para que tenhamos uma visão geral do que isto representa, apresentamos a seguir um diagrama de blocos que ilustra a arquitetura interna do circuito integrado desenvolvido.

Um A/D-SAR é constituído, basicamente, por três blocos. A lógica de controle é o mais significativo dos 3, e, além desta, é utilizado um conversor digital para analógico (D/A) e um comparador.

É na lógica de controle que é construída a estratégia de aproximações sucessivas, isto

é, o registrador de aproximações sucessivas (SAR). O registrador armazena cada um dos resultados intermediários gerados pelas sucessivas aproximações e, ao final da conversão, o resultado final da mesma. A cada nova aproximação gerada pela lógica de controle, a informação binária armazenada no registrador é submetida ao D/A para que possa ser comparada com a entrada analógica que se deseja converter. Baseado no resultado da comparação, a lógica de controle chega ao valor binário mais próximo para a entrada analógica correspondente.

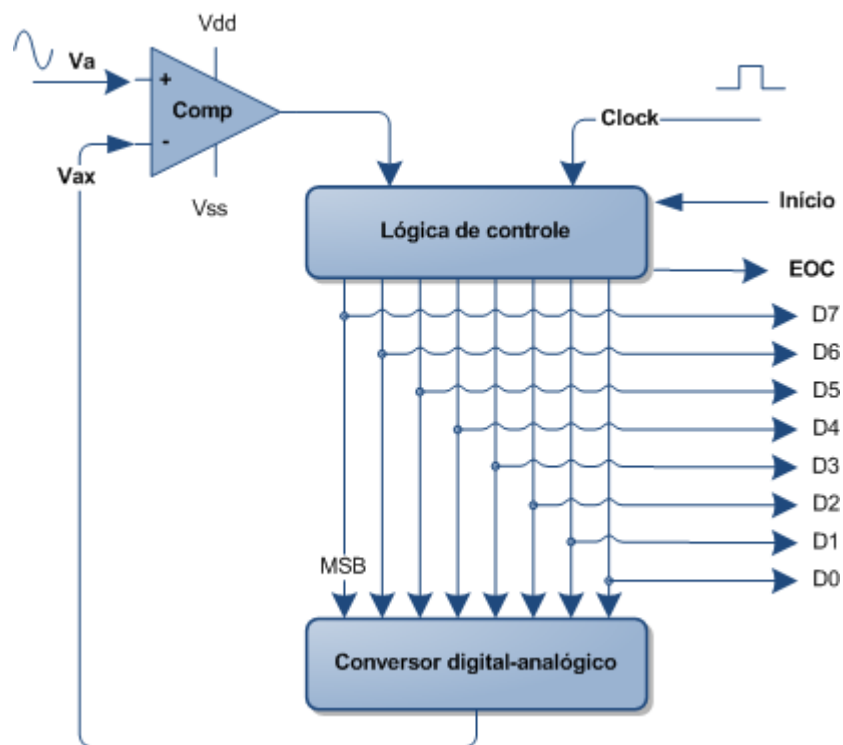


Figura 4: Diagrama de blocos do conversor A/D-SAR.

Nas próximas subseções abordaremos em maiores detalhes o papel e a estrutura de cada um desses blocos.

5.2.1 Lógica de controle

A lógica de controle compõe a parte digital do A/D-SAR. É neste bloco que o registrador de aproximações sucessivas é implementado. Por esse motivo, podemos dizer que este bloco é o núcleo do A/D-SAR. Os demais blocos, que também representam outras porções de hardware, são auxiliares para a execução da função implementada pela lógica de controle.

A lógica de controle é composta, principalmente, por um registrador de deslocamento - *shift register* - que fará com que apenas um bit por vez esteja selecionado para comparação, e por um registrador, que armazenará o resultado final da conversão.

Apresentamos, logo abaixo, um fluxograma que ilustra o seu funcionamento.

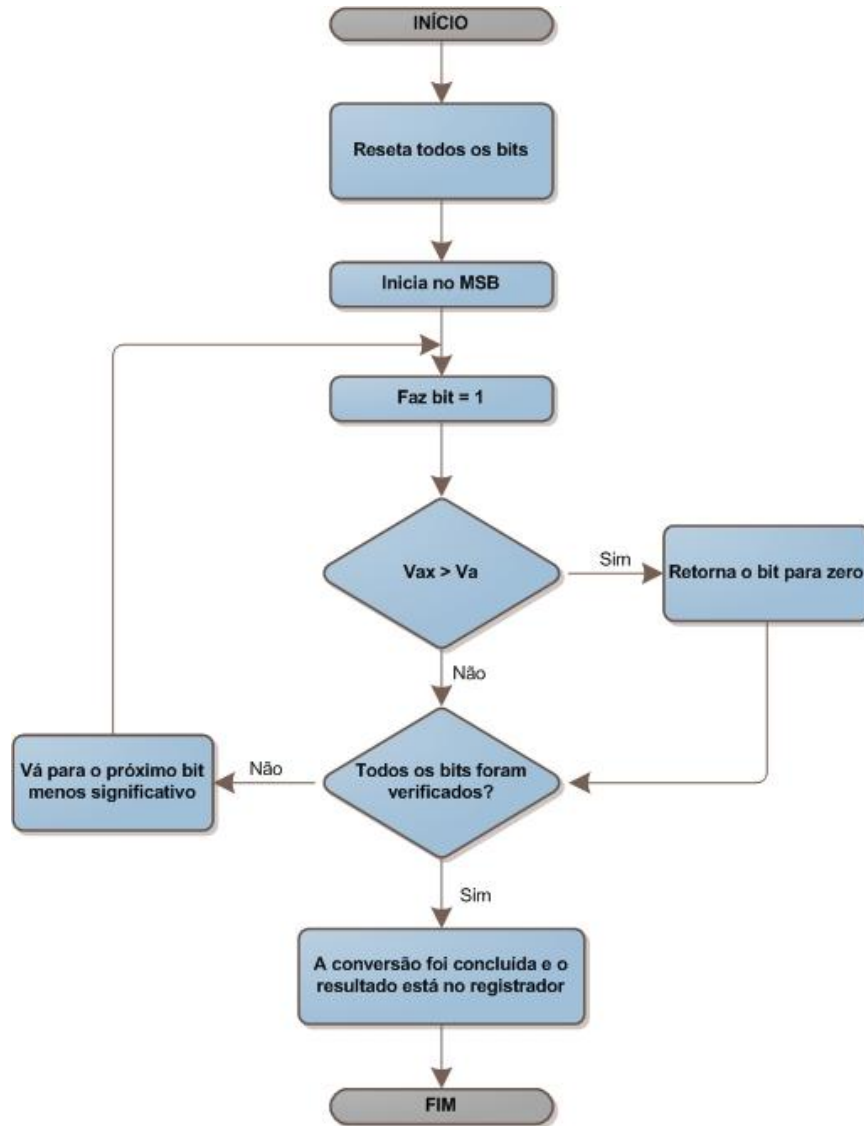


Figura 5: Fluxograma da lógica de controle.

Ao sinal de início da conversão, o *shift register* e o registrador são zerados. O processo de aproximação começa testando o bit mais significativo, e apenas este, que é então ajustado para nível lógico um (1). Com esta configuração, o valor contido no registrador é enviado para comparação. Caso este valor (V_{ax}) represente uma tensão maior que a tensão de entrada (V_a), este bit volta a ser zero (0); caso contrário, permanece um (1). Se todos os bits foram verificados, então a conversão foi concluída e o número binário armazenado no

registrador é a representação digital da entrada analógica; caso contrário, o próximo bit mais significativo deve ser testado, até que todos tenham sido verificados. Isto é, todos os 8 bits do registrador passam pela mesma verificação. Ou seja, são selecionados pelo *shift register*, ajustados para nível lógico um, comparados com o sinal de entrada e ajustados definitivamente conforme o resultado da comparação.

A lógica de controle foi projetada e construída utilizando as seguintes tecnologias:

- a) Linguagem de descrição de hardware VHDL;
- b) Simulação SPICE;
- c) GDSII.

E os seguintes recursos:

- a) Ambiente de desenvolvimento QUARTUS II da ALTERA;
- b) Ambiente de desenvolvimento DSCH do pacote MICROWIND;
- c) Ambiente de desenvolvimento MICROWIND;
- d) Ambiente de desenvolvimento HSPICE da SYNOPSIS.

Os esquemas lógicos e eletrônicos, os códigos, bem como os resultados obtidos para todos os componentes criados podem ser encontrados no anexo A.

5.2.2 Conversor Digital para Analógico com rede R-2R

O conversor digital para analógico é um dos elementos que constitui, juntamente com o amplificador operacional, a parte analógica do A/D-SAR. Ele é um bloco de interfaceamento entre a parte digital e a analógica, mais especificamente entre a lógica de controle e o comparador, respectivamente. Vejamos abaixo.

Cada vez que um bit do registrador é selecionado para comparação pelo *shift register* dentro da lógica de controle, um valor intermediário do que será o resultado final da conversão é enviado para comparação com a entrada analógica. A função de comparação é exercida pelo amplificador operacional.

No entanto, como a entrada é analógica e a informação que deve ser comparada é digital, é preciso que esta última seja convertida em analógica antes da comparação. Essa tarefa é executada por esse bloco, o conversor D/A. Deste modo, sua função é adequar os números binários gerados pelas sucessivas aproximações de tal modo que possa ser comparado com o sinal analógico que se deseja converter.

A imagem abaixo ilustra a saída de um conversor D/A em função da palavra binária de entrada.

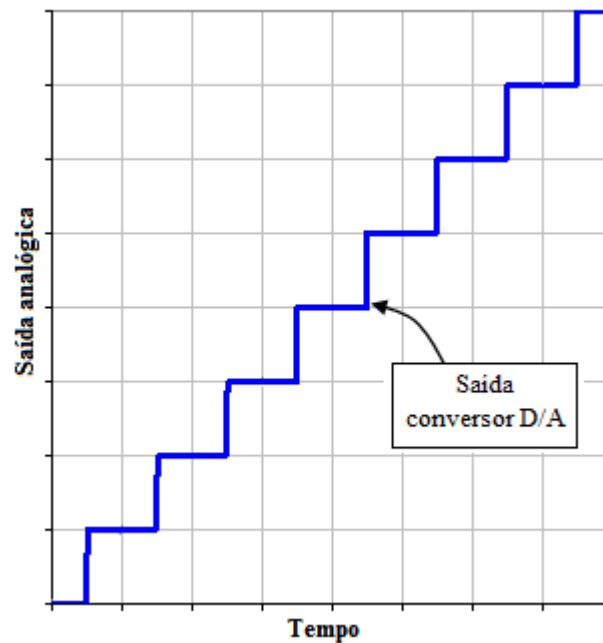


Figura 6: Saída típica de um conversor D/A.

O conversor digital para analógico foi projetado e construído utilizando as seguintes tecnologias:

- a) Simulação SPICE;
- b) GDSII.

E os seguintes recursos:

- a) Ambiente de desenvolvimento HSPICE da SYNOPSYS;
- b) Ambiente de desenvolvimento MICROWIND.

O esquema eletrônico, o código, bem como os resultados obtidos para o componente criado pode ser encontrado no anexo A.

5.2.3 Amplificador operacional (Comparador)

O amplificador operacional é o outro elemento que constitui a parte analógica do A/D-SAR. Ele é encarregado de comparar os números binários gerados pela lógica de controle com a entrada analógica, que é o sinal que queremos converter.

Conforme vimos, cada vez que um bit do registrador é selecionado para comparação pelo *shift register* dentro da lógica de controle, um novo valor é enviado para comparação com a entrada analógica. O resultado da comparação (nível lógico 0 ou 1) é enviado à lógica

de controle que, então, armazena essa informação no bit adequado.

A tabela verdade que ilustra seu funcionamento pode ser observada abaixo.

Entrada	Nível lógico
	Saída
Analógica > Digital	1
Analógica < Digital	0

Tabela 2: Tabela verdade do comparador (níveis lógicos).

O amplificador operacional foi projetado e construído utilizando as seguintes tecnologias:

- a) Simulação SPICE;
- b) GDSII.

E os seguintes recursos:

- a) Ambiente de desenvolvimento HSPICE da SYNOPSYS;
- b) Ambiente de desenvolvimento MICROWIND.

O esquema eletrônico, o código, bem como os resultados obtidos para o componente criado pode ser encontrado no anexo A.

6 PROCEDIMENTOS DE TESTE E RESULTADOS OBTIDOS

Em eletrônica, o elemento mais simples que existe é o transistor. É a partir de combinações entre transistores que surgem as portas lógicas e memórias mais simples que, por sua vez, combinadas dão origem a funções mais complexas e assim por diante. Deste modo, para que chegássemos ao resultado final deste projeto, que aqui apresentamos, foi necessário que construíssemos cada componente, do mais simples ao mais complexo, transistor a transistor. De maneira a representar esse fluxo de trabalho, este capítulo está dividido em duas subseções: uma que traz as verificações e apresenta os resultados obtidos para esses elementos básicos, que serviram de matéria-prima para a construção do conversor A/D-SAR; e outra que apresenta as verificações e os resultados para os três blocos principais apresentados no capítulo anterior, que formam o circuito integrado projetado.

6.1 CIRCUITOS BÁSICOS

Os seguintes componentes foram criados para a construção do conversor analógico para digital.

6.1.1 Porta inversora (NOT)

A porta NOT realiza a função lógica “NÃO”. Na prática, a saída é o inverso da entrada aplicada.

A tabela verdade que representa esta função pode ser observada abaixo.

Nível lógico	
Entrada	Saída
0	1
1	0

Tabela 3: Tabela verdade da porta inversora.

O símbolo lógico para a NOT é apresentado abaixo.

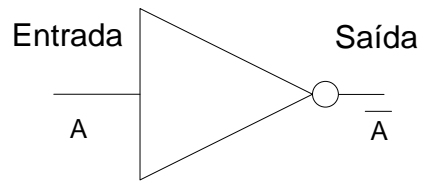


Figura 7: Símbolo lógico da porta inversora.

Para verificação do seu funcionamento, conforme a tabela verdade foi aplicado um sinal à sua entrada e observada a inversão esperada na saída.

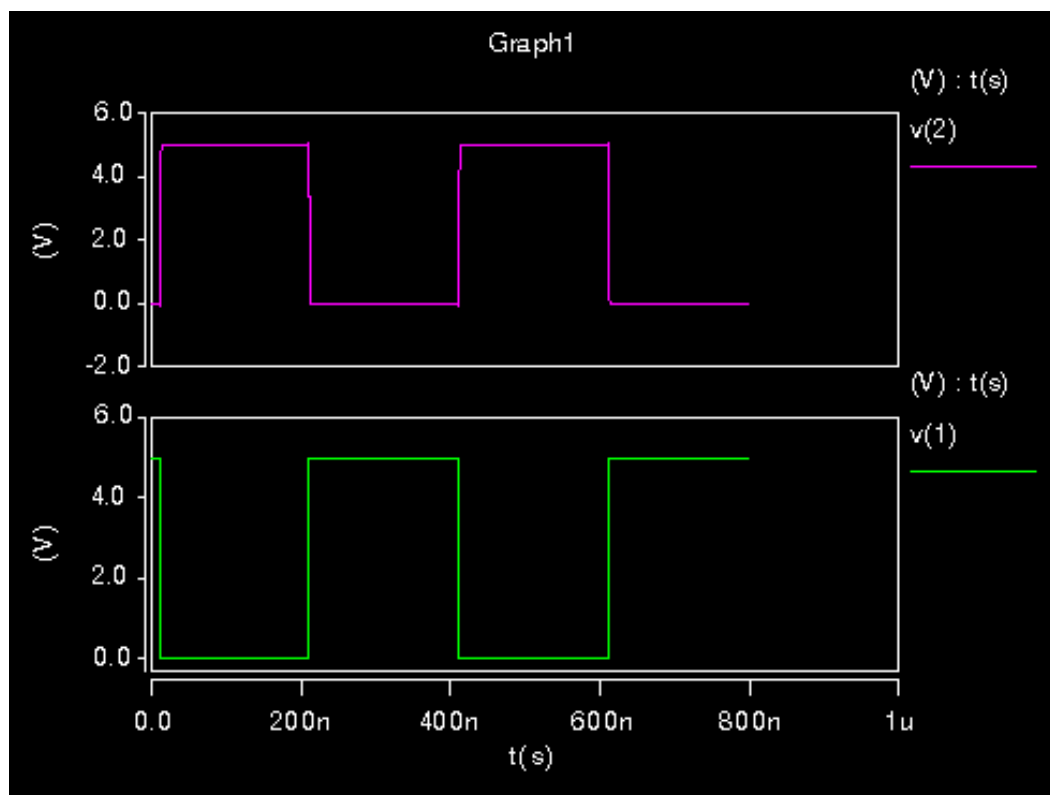


Figura 8: Resultado da simulação da porta inversora.

O esquema lógico e o eletrônico, os códigos, bem como os resultados obtidos também estão disponíveis no anexo A.

6.1.2 Porta NAND

A porta NAND realiza a função lógica “não-E”. Na prática, a saída só será “0” (zero) quando as duas entradas forem “1” (um).

A tabela verdade que representa esta função pode ser observada abaixo.

Nível lógico		
Entrada A	Entrada B	Saída
0	0	1
0	1	1
1	0	1
1	1	0

Tabela 4: Tabela verdade da porta NAND.

O símbolo lógico para a NAND é apresentado abaixo.

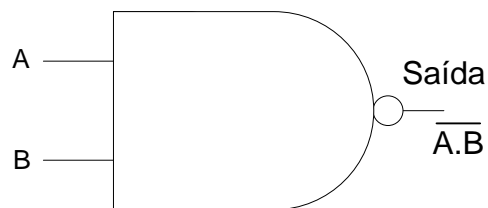


Figura 9: Símbolo lógico da porta NAND.

Para verificação do seu funcionamento, conforme a tabela verdade foi aplicado um sinal a cada uma de suas entradas e observada a função esperada na saída.

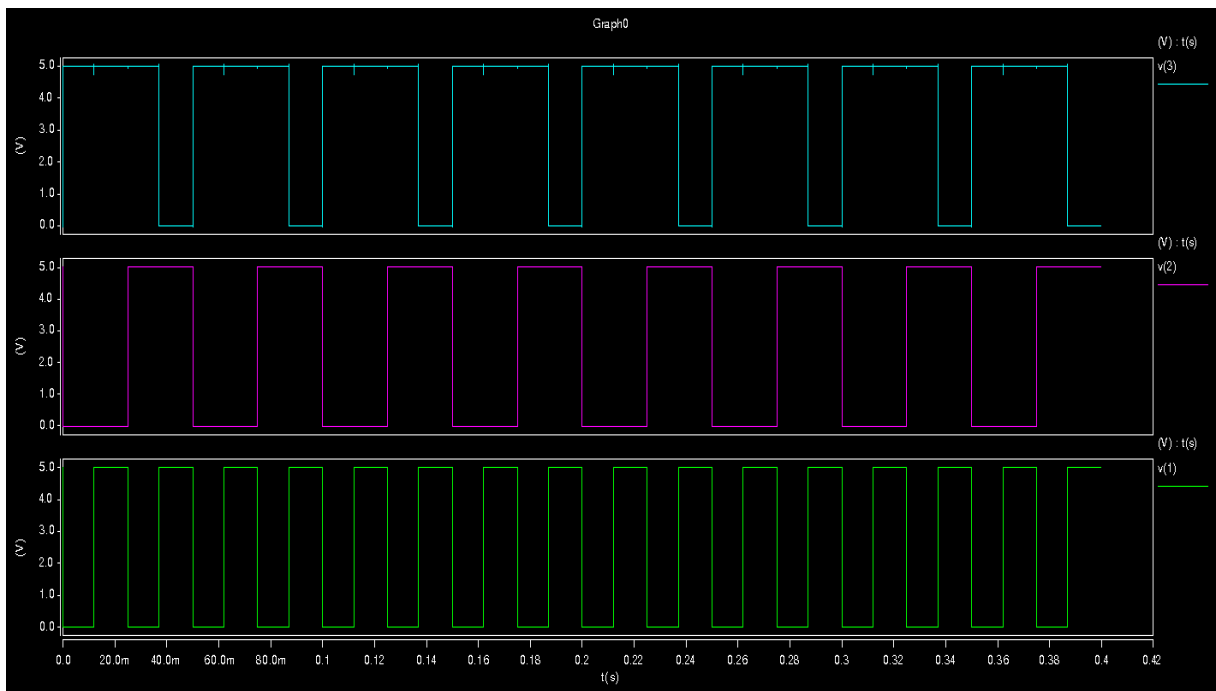


Figura 10: Resultado da simulação da porta NAND.

O esquema lógico e o eletrônico, os códigos, bem como os resultados obtidos também estão disponíveis no anexo A.

6.1.3 Porta AND

A porta AND realiza a função lógica “ E ”. Na prática, a saída só será “1” (um) quando as duas entradas forem “1” (um).

A tabela verdade que representa esta função pode ser observada abaixo.

Nível lógico		
Entrada A	Entrada B	Saída
0	0	0
0	1	0
1	0	0
1	1	1

Tabela 5: Tabela verdade da porta AND.

O símbolo lógico para a AND é apresentado abaixo.

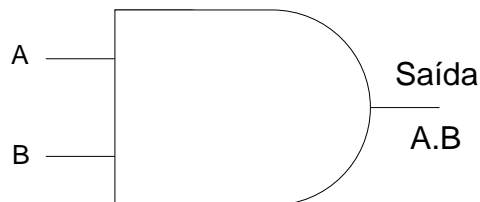


Figura 11: Símbolo lógico da porta AND.

Para verificação do seu funcionamento, conforme a tabela verdade foi aplicado um sinal a cada uma de suas entradas e observada a função esperada na saída.

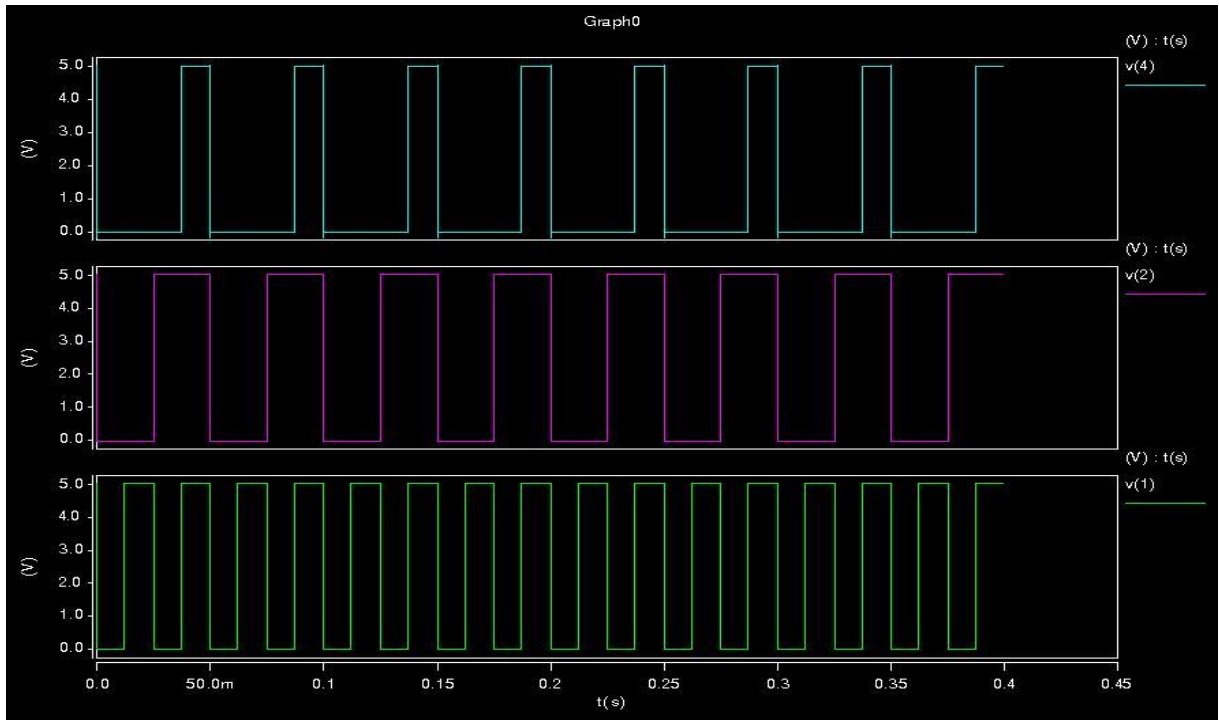


Figura 12: Resultado da simulação da porta AND.

O esquema lógico e o eletrônico, os códigos, bem como os resultados obtidos também estão disponíveis no anexo A.

6.1.4 Porta NOR

A porta NOR realiza a função lógica “*não-OU*”. Na prática, a saída só será “1” (um) quando as duas entradas forem “0” (zero).

A tabela verdade que representa esta função pode ser observada abaixo.

Nível lógico		
Entrada A	Entrada B	Saída
0	0	1
0	1	0
1	0	0
1	1	0

Tabela 6: Tabela verdade da porta NOR.

O símbolo lógico para a NOR é apresentado abaixo.

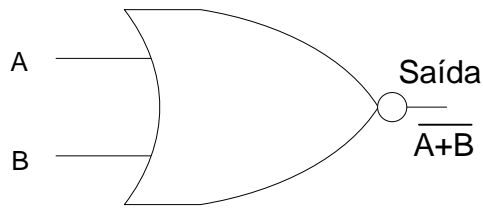


Figura 13: Símbolo lógico da porta NOR.

Para verificação do seu funcionamento, conforme a tabela verdade foi aplicado um sinal a cada uma de suas entradas e observada a função esperada na saída.

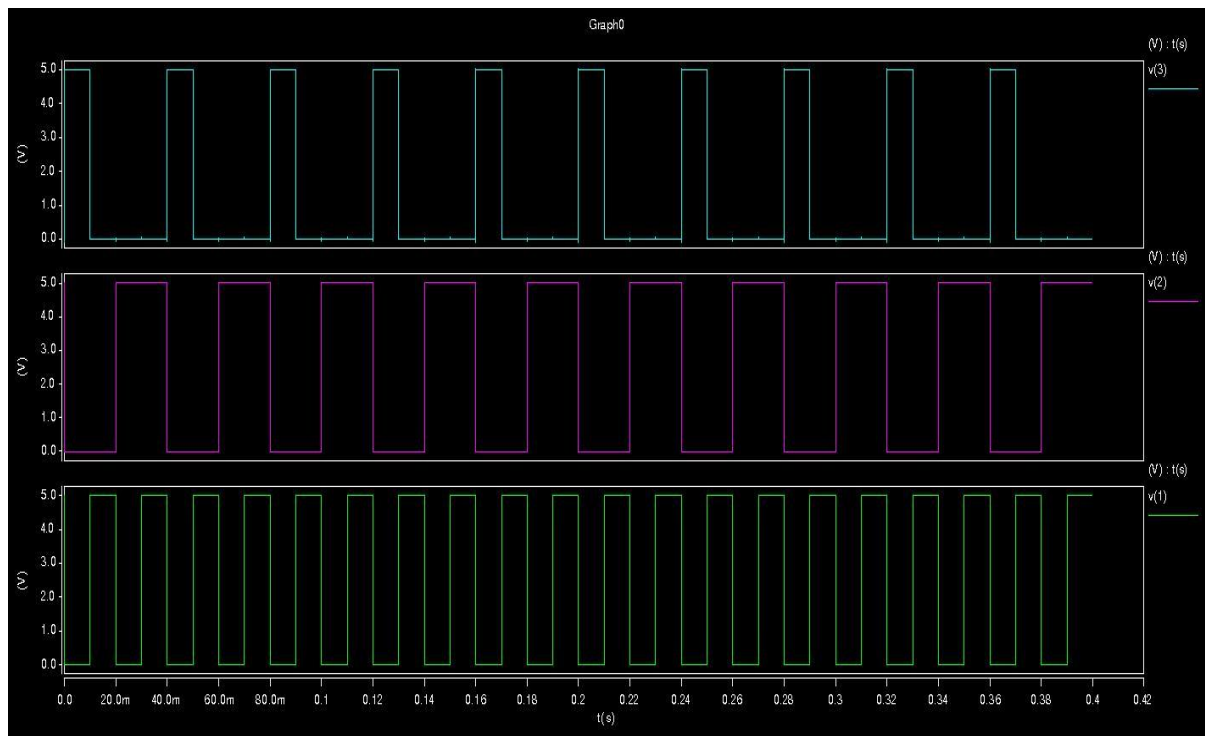


Figura 14: Resultado da simulação da porta NOR.

O esquema lógico e o eletrônico, os códigos, bem como os resultados obtidos também estão disponíveis no anexo A.

6.1.5 Porta OR

A porta OR realiza a função lógica “OU”. Na prática, a saída só será “0” (zero) quando as duas entradas forem “0” (zero).

A tabela verdade que representa esta função pode ser observada abaixo.

Nível lógico		
Entrada A	Entrada B	Saída
0	0	0
0	1	1
1	0	1
1	1	1

Tabela 7: Tabela verdade da porta OR.

O símbolo lógico para a OR é apresentado abaixo.

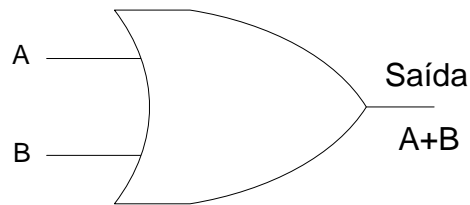


Figura 15: Símbolo lógico da porta OR.

Para verificação do seu funcionamento, conforme a tabela verdade foi aplicado um sinal a cada uma de suas entradas e observada a função esperada na saída.

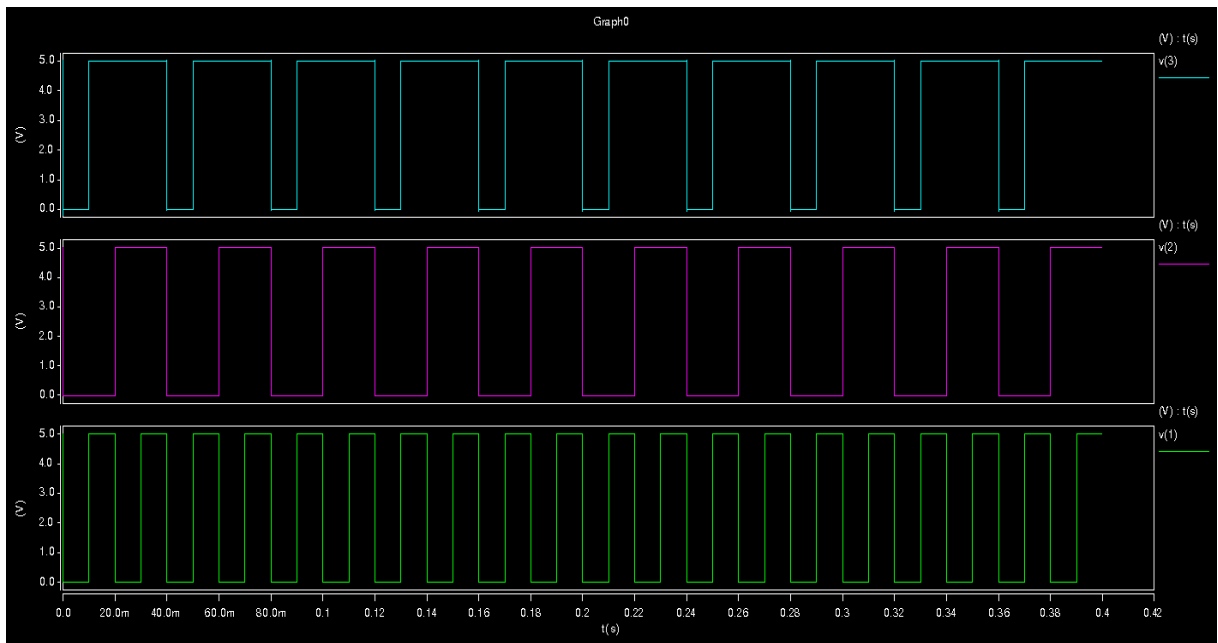


Figura 16: Resultado da simulação da porta OR.

O esquema lógico e o eletrônico, os códigos, bem como os resultados obtidos também estão disponíveis no anexo A.

6.1.6 Porta NAND3

A porta NAND3, assim como a NAND padrão de duas entradas já apresentada, também realiza a função lógica “*não-E*”, porém, agora com três entradas. Na prática, a saída só será “0” (zero) quando as três entradas forem “1” (um).

A tabela verdade que representa esta função pode ser observada abaixo.

Nível lógico			
Entrada A	Entrada B	Entrada C	Saída
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Tabela 8: Tabela verdade da porta NAND3.

O símbolo lógico para a NAND3 é apresentado abaixo.

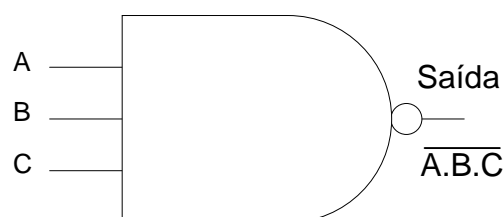


Figura 17: Símbolo lógico da porta NAND3.

Para verificação do seu funcionamento, conforme a tabela verdade foi aplicado um sinal a cada uma de suas entradas e observada a função esperada na saída.

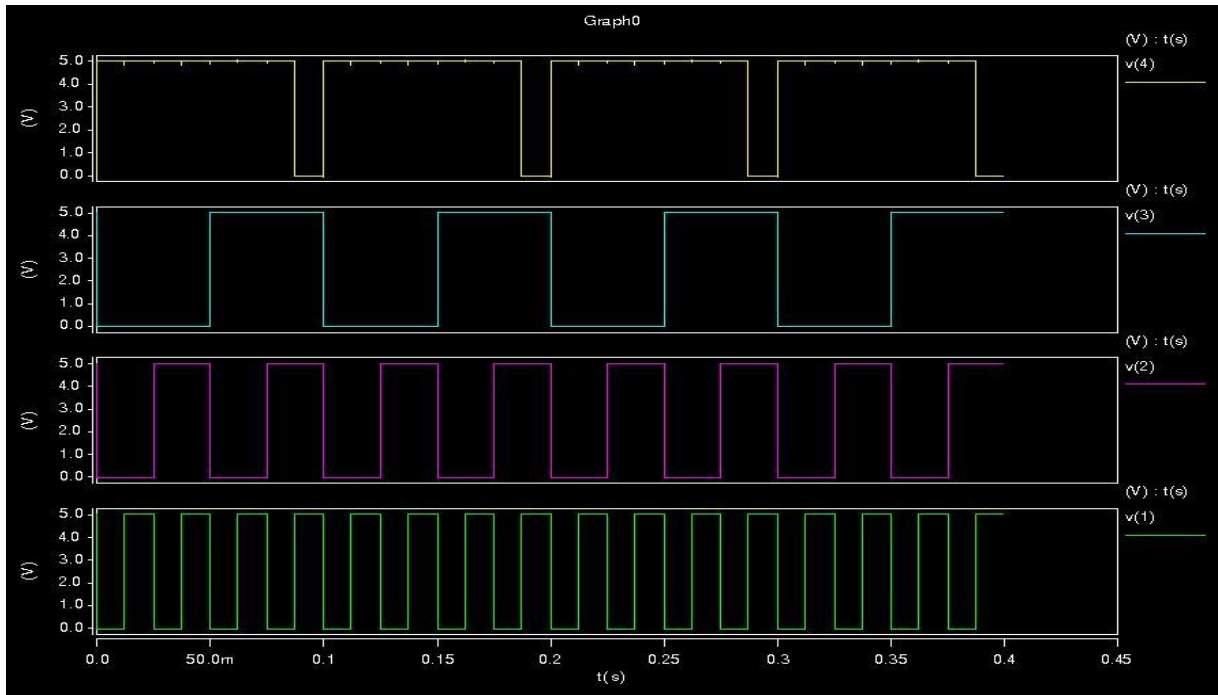


Figura 18: Resultado da simulação da porta NAND3.

O esquema lógico e o eletrônico, os códigos, bem como os resultados obtidos também estão disponíveis no anexo A.

6.1.7 Flip-flop tipo D com clear assíncrono

O elemento lógico básico que provê memória em muitos circuitos seqüenciais é o *flip-flop*. Ele realiza a função de memória de 1 bit.

O mais econômico e eficiente *flip-flop* em termos de número de transistores e área de silício é o *flip-flop D*. Também é possível projetar o circuito para que a saída do *flip-flop* mude na borda de subida (transição de 0 para 1) do sinal de controle *clock*. E ainda, que o *flip-flop* tenha uma ou mais entradas assíncronas, usadas para forçar o *flip-flop* para um estado particular independente do *clock*.

Na prática, um flip-flop D com clear assíncrono só funciona quando a entrada *clear* está em nível lógico “1”, caso contrário, o *flip-flop* está sempre em estado de *reset* (nível lógico “0”) independente dos sinais presentes nas entradas *clock* e *D*. Estando *clear* em “1”, a saída *Q* somente será afetada caso uma transição de “0” para “1” aconteça em *clock*. Ou seja, caso o sinal de *clock* seja contínuo ou mesmo uma borda descendente (transição de 1 para 0), a saída permanece inalterada. Isto comprova o efeito de memória exercido por tal componente. A tabela verdade que ilustra seu funcionamento pode ser observada abaixo.

Nível lógico				
Entrada			Saída	
Clock	Clear	D	Q	Q_bar
x	0	x	0	1
1	1	x	Não altera	Não altera
0	1	x	Não altera	Não altera
↑	1	0	0	1
↑	1	1	1	0

Tabela 9: Tabela verdade do flip-flop D.

O símbolo lógico para o flip-flop D é apresentado abaixo.

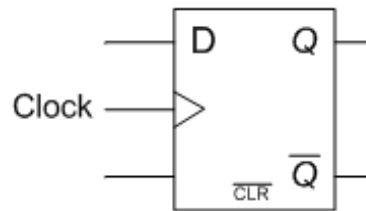


Figura 19: Símbolo lógico do flip-flop D.

Para verificação do seu funcionamento, foi aplicado um sinal a cada uma de suas entradas conforme a tabela verdade e observada a função esperada na saída.

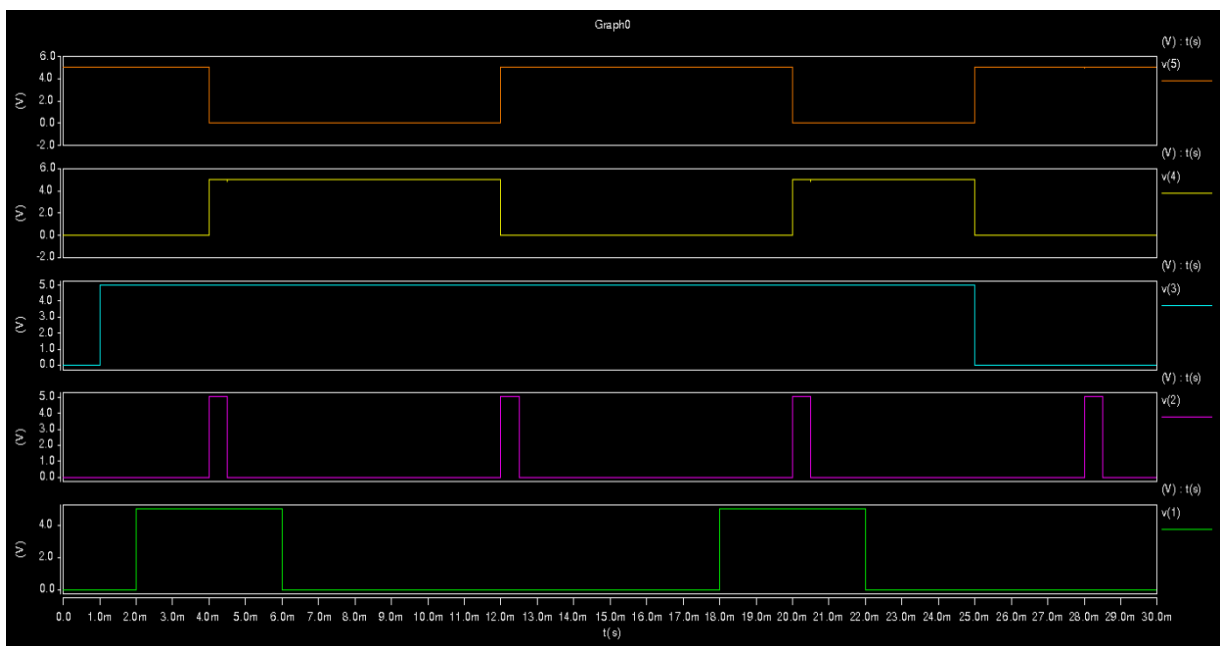


Figura 20: Resultado da simulação do flip-flop D.

O esquema lógico, os códigos, bem como os resultados obtidos estão disponíveis também no anexo A.

6.1.8 Shift Register

Um *shift register*, ou registrador de deslocamento, é uma cascata de *flip-flops* que compartilham o mesmo *clock* e tem cada um, exceto o último flip-flop, sua saída conectada à entrada do próximo na cadeia.

Esse arranjo resulta em um circuito que desloca em uma posição a sequência de bits armazenada, deslocando para dentro o dado presente em sua entrada e deslocando para fora o último bit na sequência, por meio de uma transição de *clock*.

A tabela verdade que ilustra seu funcionamento pode ser observada abaixo.

Nível lógico										
Entrada			Saída							
Clock	Clear	Data	Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7
x	0	x	0	0	0	0	0	0	0	0
1	1	x	Não altera	Não altera	Não altera	Não altera	Não altera	Não altera	Não altera	Não altera
0	1	x	Não altera	Não altera	Não altera	Não altera	Não altera	Não altera	Não altera	Não altera
↑	1	1	1	0	0	0	0	0	0	0
↑	1	0	0	1	0	0	0	0	0	0
↑	1	0	0	0	1	0	0	0	0	0
↑	1	0	0	0	0	1	0	0	0	0
↑	1	0	0	0	0	0	1	0	0	0
↑	1	0	0	0	0	0	0	1	0	0
↑	1	0	0	0	0	0	0	0	1	0
↑	1	0	0	0	0	0	0	0	0	1

Tabela 10: Tabela verdade do flip-flop D.

O esquema lógico para o registrador de deslocamento é apresentado abaixo.

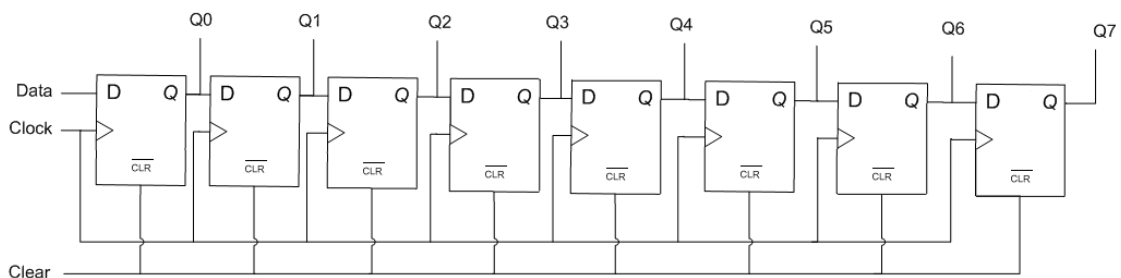


Figura 21: Esquema lógico do shift register.

Para verificação do seu funcionamento, foi aplicado um sinal a cada uma de suas entradas conforme a tabela verdade e observada a função esperada na saída.

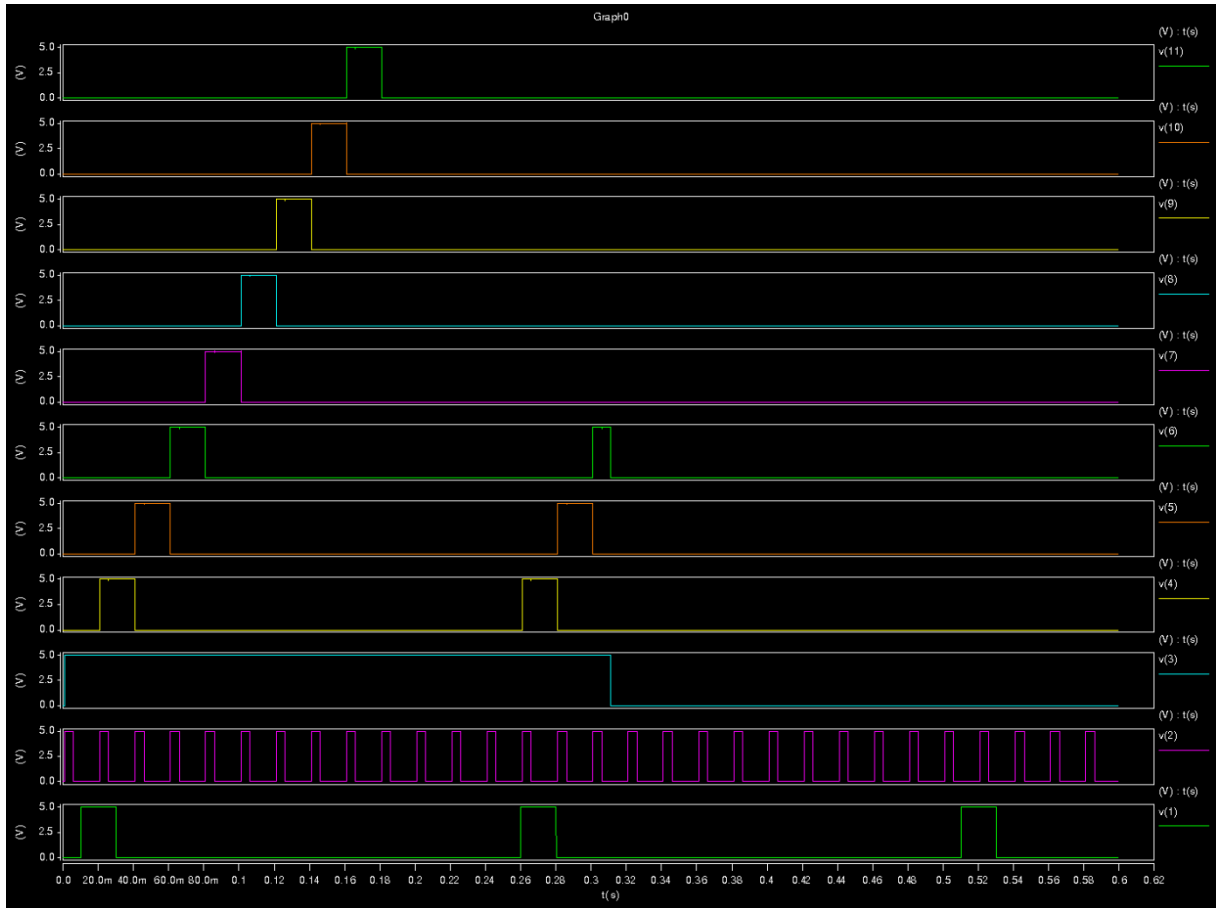


Figura 22: Resultado da simulação do shift register.

O esquema lógico, os códigos, bem como os resultados obtidos estão disponíveis também no anexo A.

6.2 CONVERSOR A/D-SAR

Nesta seção, apresentamos os testes e os resultados obtidos para o conversor analógico para digital por aproximações sucessivas.

Ao contrário da abordagem adotada na seção anterior, para esta não tratamos detalhadamente do funcionamento de cada um dos blocos, uma vez que esta explicação já foi dada no capítulo anterior. Apresentamos apenas aquele material necessário à facilitar a compreensão dos resultados.

6.2.1 Lógica de controle

O esquema lógico do registrador de aproximações sucessivas é apresentado abaixo.

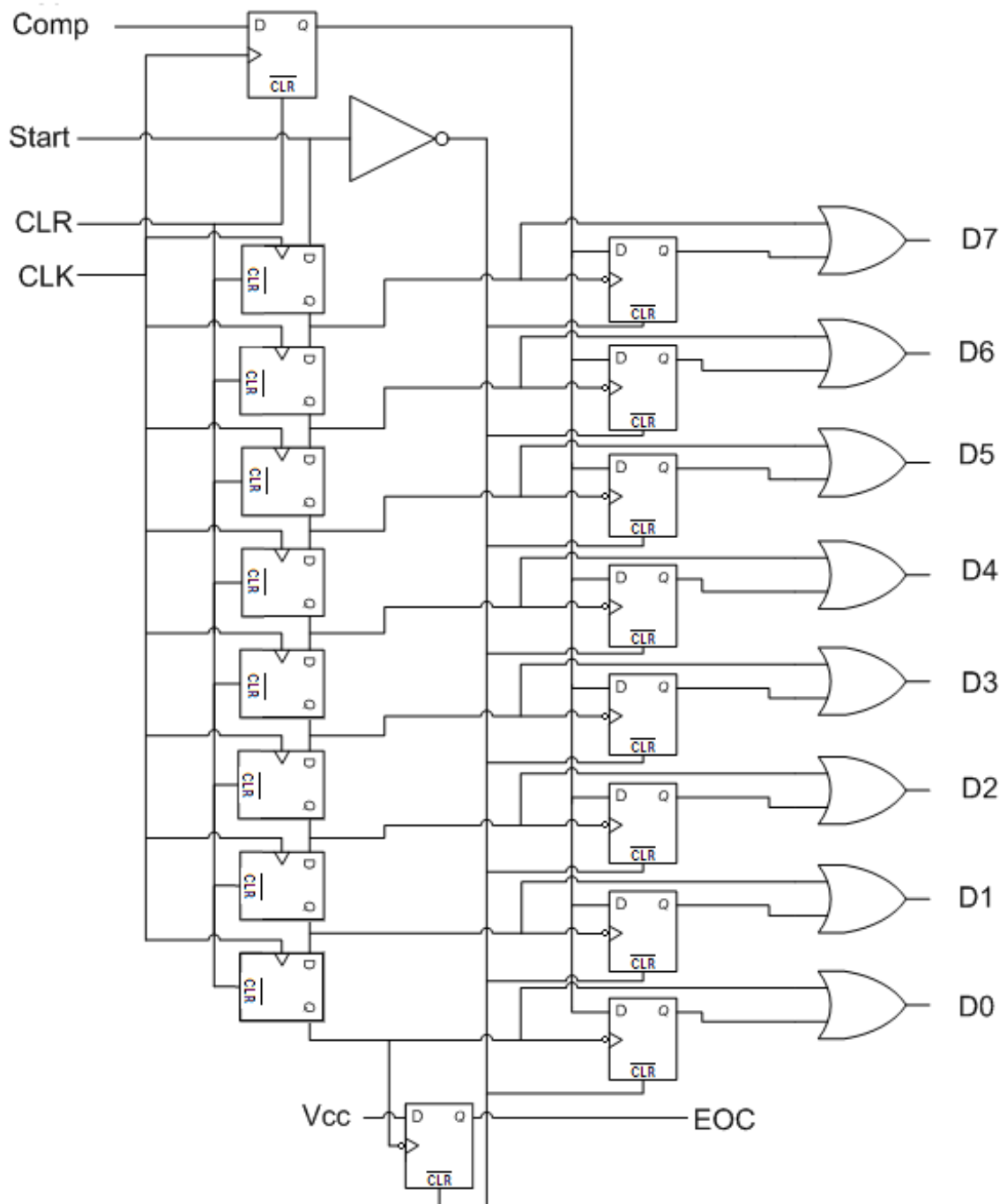


Figura 23: Esquema lógico do registrador de aproximações sucessivas (lógica de controle).

Para verificação do seu funcionamento, foi aplicado um sinal adequado a cada uma de suas entradas e observado o funcionamento tal qual o fluxograma abaixo.

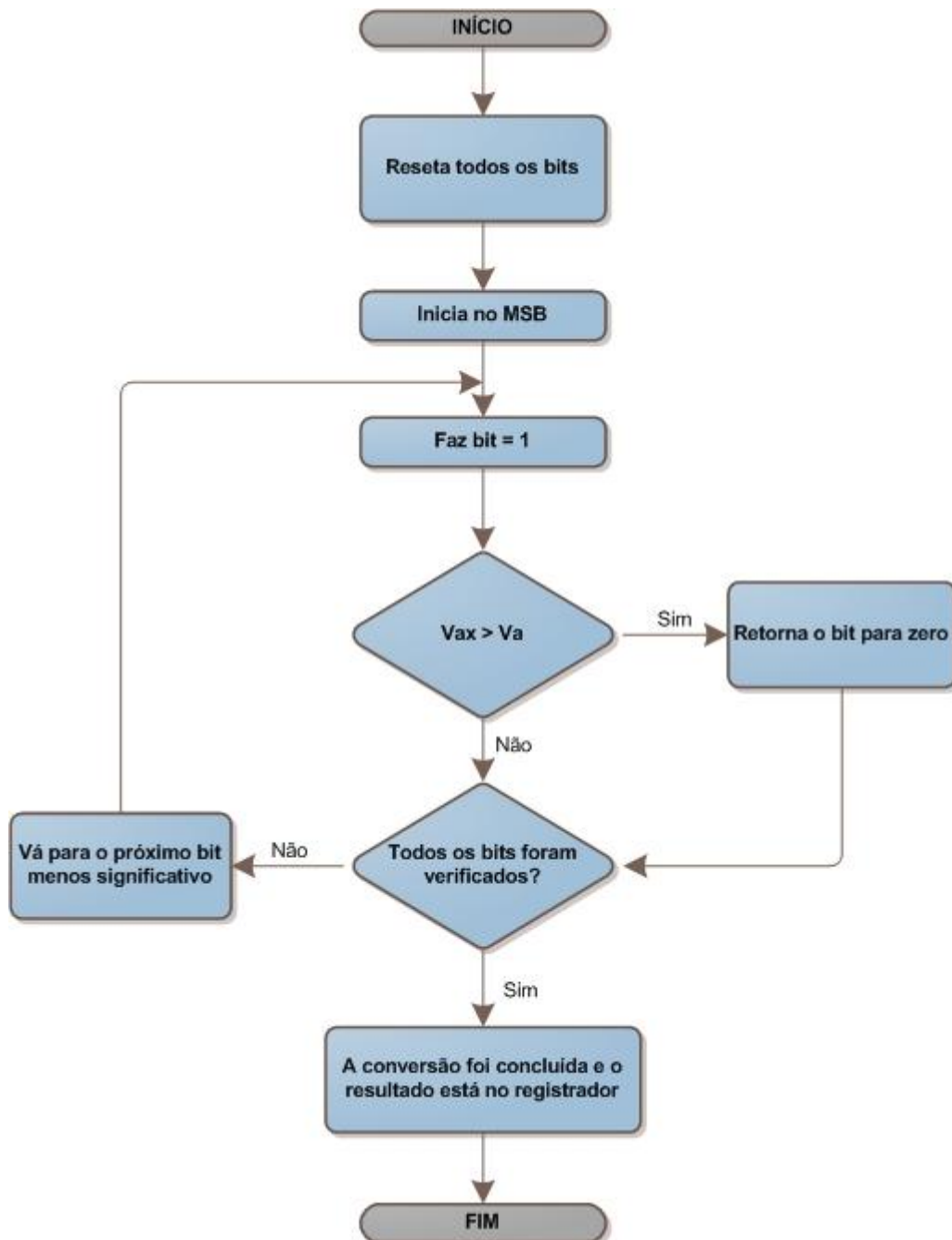


Figura 24: Fluxograma da lógica de controle.

O resultado da verificação abaixo comprova o funcionamento esperado.

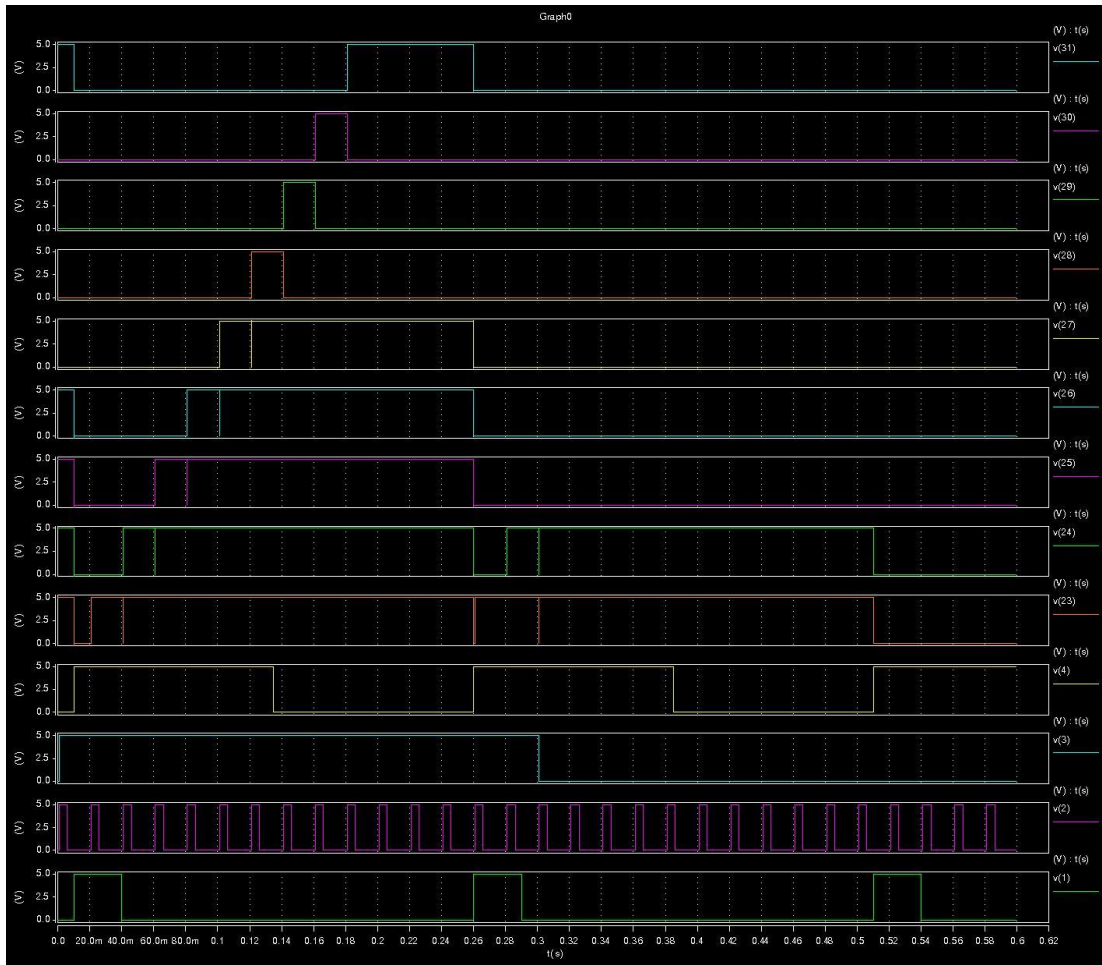


Figura 25: Resultado obtido a partir do esquemático construído.

O esquema lógico, os códigos, bem como os resultados obtidos estão disponíveis também no anexo A.

6.2.2 Conversor D/A R-2R

O símbolo lógico para o conversor digital para analógico é apresentado abaixo.

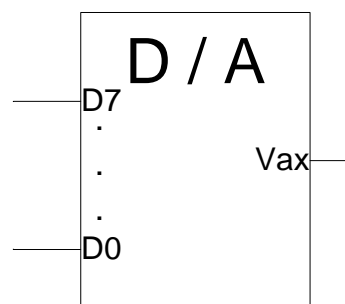


Figura 26: Símbolo lógico do conversor D/A R-2R.

Para verificação do seu funcionamento, foi aplicado um sinal adequado a cada uma de suas entradas e observado o funcionamento tal qual esperado.

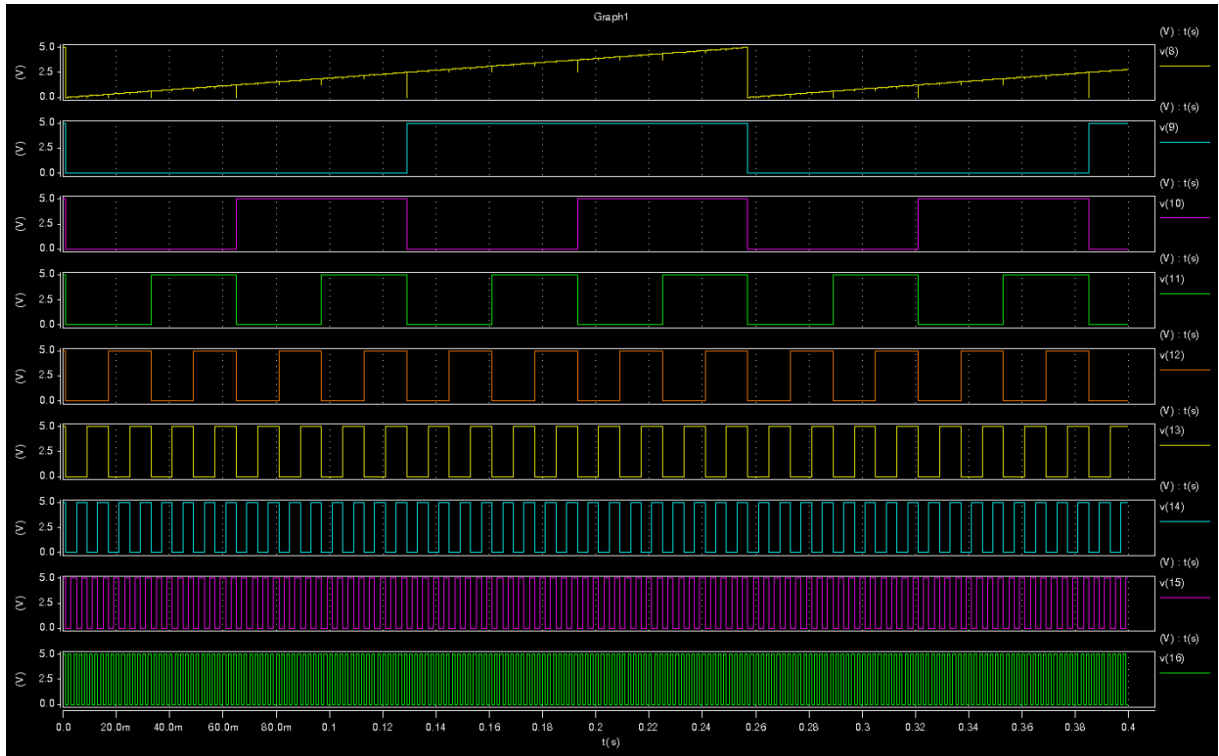


Figura 27: Resultado obtido a partir do esquemático construído.

O esquema eletrônico, o código, bem como os resultados obtidos estão disponíveis também no anexo A.

6.2.3 Amplificador operacional (Comparador)

A tabela verdade que ilustra seu funcionamento pode ser observada abaixo.

Entrada	Saída
$V_+ > V_-$	3,8 V
$V_- > V_+$	0 V

Tabela 11: Tabela verdade do comparador.

Ou seja:

Entrada	Nível lógico
	Saída
Analógica > Digital	1
Analógica < Digital	0

Tabela 12: Tabela verdade do comparador (níveis lógicos).

O símbolo lógico para o amplificador operacional é apresentado abaixo.

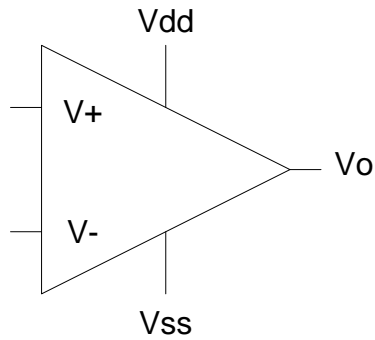


Figura 28: Símbolo lógico do amplificador operacional.

Para verificação do seu funcionamento, foi aplicado um sinal adequado a cada uma de suas entradas e observado o funcionamento tal qual esperado.

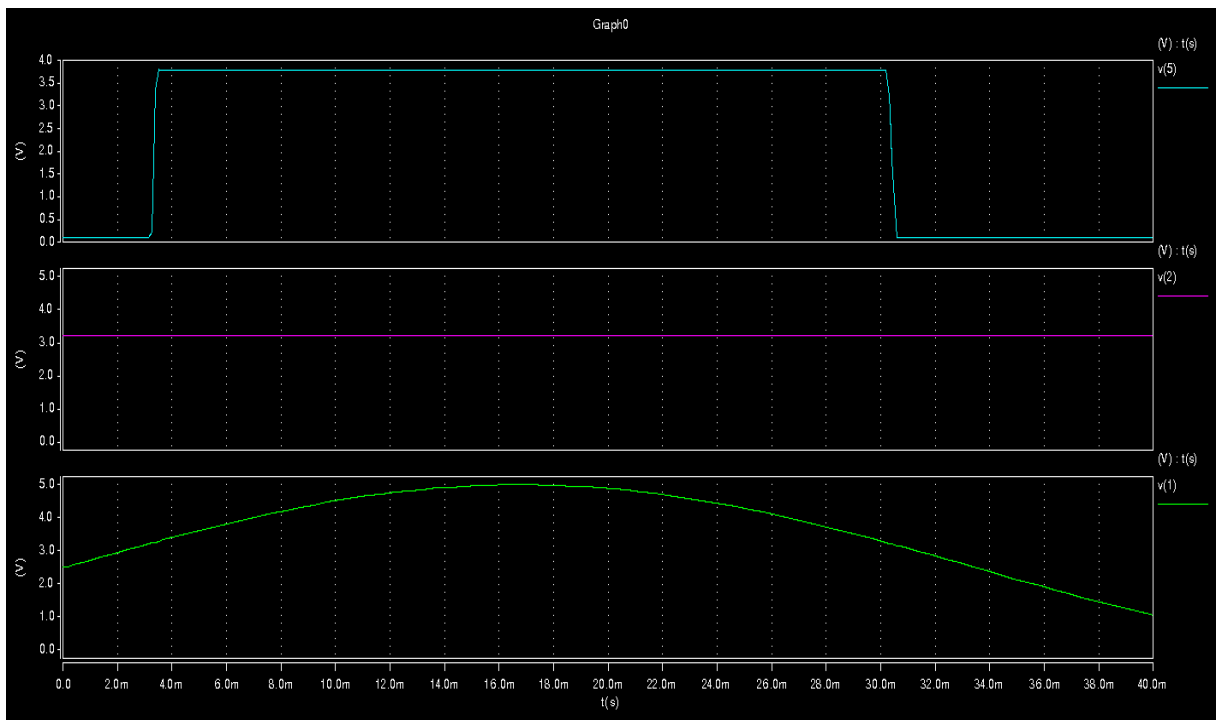


Figura 29: Resultado obtido a partir do esquemático construído.

O esquema eletrônico, o código, bem como os resultados obtidos estão disponíveis também no anexo A.

7 CONCLUSÃO

Durante a disciplina de Projeto Final desenvolvemos o projeto de um circuito integrado. Esta é a atividade principal exercida por uma Design House no meio industrial.

Para desenvolver esse trabalho, a equipe fez um estudo teórico a respeito de diversas técnicas de conversão de sinais analógicos em digitais e dos fundamentos de projeto de circuitos integrados. Uma vez definida a topologia do conversor, iniciamos com a descrição de hardware dos elementos digitais e, paralelamente, a modelagem eletrônica (simulação SPICE) dos elementos analógicos e dos elementos digitais elementares. Em cada uma dessas etapas, todos os testes e simulações que se fizeram necessários foram também realizados com o objetivo de confirmar os resultados esperados. Concluídas estas duas etapas, passamos à representação geométrica do circuito integrado. Como mencionado em momento oportuno, esta última etapa corresponde à criação do layout do circuito integrado, que será usado pela fábrica para a construção das máscaras em processo de fotolitografia para, então, serem utilizadas na fabricação do circuito integrado. Este é o objeto final do projeto de um circuito integrado. Concluídas as atividades de modelagem, passamos à verificação final que ratificou o funcionamento e validou o projeto realizado.

Durante a realização do projeto as principais dificuldades encontradas foram:

a) A inexperiência da equipe em projeto de circuitos integrados, por não se tratar de um curso com ênfase em projeto de circuitos integrados e por este trabalho ser, ainda, um projeto piloto para o Laboratório de Microeletrônica.

b) A indisponibilidade das ferramentas MENTOR GRAPHICS, um conjunto de 12 ferramentas para síntese e verificação do circuito integrado. Isso levou a um sério problema, pois foi preciso que buscássemos um conjunto bem menor com apenas duas ferramentas, Microwind e DSch. No entanto, com recursos muito limitados quando comparados com o conjunto MENTOR, a nossa produtividade caiu consideravelmente levando-nos até mesmo a vários retrabalhos dos componentes utilizados por ocasião da não compatibilidade entre as ferramentas utilizadas.

Agora, um destaque importante quanto ao produto do trabalho realizado é que todo esse material produzido ficará como base de dados para a realização de quaisquer novos projetos e atividades utilizando o Laboratório de Microeletrônica, já que a construção de qualquer circuito em microeletrônica faz uso das portas lógicas ou componentes analógicos construídos, até mesmo de um conversor analógico para digital. Ressaltamos a importância desse material para os alunos do curso de Engenharia Eletrônica, que na disciplina de

Sistemas Digitais serão direcionados para o projeto de portas lógicas, que compõem o universo das máquinas digitais (processadores, microcontroladores, conversores), para então terem condições de prosseguir em outras disciplinas de microeletrônica. Essa característica valoriza ainda mais a iniciativa de realizar um projeto pioneiro em microeletrônica na PUCPR.

Outro fator importante que valoriza o trabalho realizado foi a iniciativa da equipe em aventurar-se por um setor que não é o foco do curso de graduação que estudam e o esforço em buscar conhecer tecnologias não abordadas por ele. Por meio desse trabalho conseguimos atingir o objetivo pessoal de cada um com vistas ao prosseguimento da carreira, que era o de aprender os fundamentos das técnicas e das tecnologias de projeto de circuitos integrados independente do que viesse a ser desenvolvido para atingir esse fim.

Acreditamos, ainda, que o maior objetivo da disciplina de projeto final é avaliar a capacidade dos alunos em gerenciar, mas não só isso, de executar um projeto de hardware ou software e, através deste, avaliar o domínio do conhecimento amalhado durante os 5 anos de graduação, sem o que não seria possível executar tal projeto satisfatoriamente. Acreditamos que atingimos mais este objetivo.

Finalmente, por esses motivos, estamos certos de que nosso projeto cumpriu sua tarefa.

REFERÊNCIAS BIBLIOGRÁFICAS

MEAD, Carver; CONWAY, Lynn. **"Introduction to VLSI Systems"**. ISBN 0-20-104358-0, Addison-Wesley Pub, 1979.

ANALOG DEVICES, Ed. Daniel H. Sheingold. **"Analog-Digital Conversion Handbook"**. ISBN 0-13-032848-0, Prentice Hall, Englewood Cliffs, NJ, 1986, 670pp;

RAZAVI, Behzad. **"Design of Analog CMOS Integrated Circuits"**, ISBN 0-07-118839-8, McGRAW-HILL International Editions, Electrical Engineering Series, 2001, 684pp;

ANALOG DEVICES, Ed. Walt Jung. **"Op Amp Applications Handbook"**. ISBN 0-7506-7844-5, Newnes, 2006, 878pp;

BUCHANAN, James E. **"BiCMOS/CMOS Systems Design"**, ISBN 0-07-008712-1, McGRAW-HILL International Editions, 1991, 322pp;

CUTLER, Phillip. **"Circuitos Eletrônicos Lineares"**, McGRAW-HILL do Brasil, 1972, 386pp;

CLAYTON, George; WINDER, Steve. **"Operational Amplifiers"**, ISBN 0-75-064643-8, Fourth Ed, Newnes Ed., 2000, 383pp;

GRAEME, Jerald, **"Amplifier Applications of OP AMPS"**, ISBN 0-07-134642-2, McGRAW-HILL International Editions, 1999, 286pp;

LANCASTER, Don. **"Active Filter Cookbook"**, ISBN 978-0-7506-2986-7, Newnes Ed., 2008, 240pp;

PERTENCE JR, Antonio. **"Amplificadores Operacionais e Filtros Ativos"**, McGRAW-HILL do Brasil, 1988, 359pp;

FLOYD, Thomas L.; BUCHLA, David. **"Basic Operational Amplifiers and Linear Integrated Circuits"**. ISBN 0-13-082987-0, Prentice Hall, Inc., 1999, 593pp;

KLEITZ, William. **"Digital Electronics with VHDL"**. ISBN 0-13-171490-2, Pearson Education Inc., Prentice Hall, 2006, 938pp;

TOCCI, Ronald J.; WIDMER, Neal S. **“Sistemas Digitais: Princípios e Aplicações”**. ISBN 8-58-791820-6, Prentice Hall, 8 ed. 2000, 768pp;

SMITH, Kenneth C.; SEDRA, Adel S. **“Microeletrônica”**. ISBN 8-57-605022-6, Prentice Hall, 5 ed. 2007, 864pp;

BOYLESTAD, Robert L.; NASHELSKY, Louis. **“Dispositivos eletrônicos e teoria de circuitos”**. ISBN 8-58-791822-2, Prentice Hall, 8 ed. 2004, 696pp;

WILLIAMS, Jim. **“The Art and Science of Analog Circuit Design”**. ISBN 0-75-067062-2, Newnes, 1998;

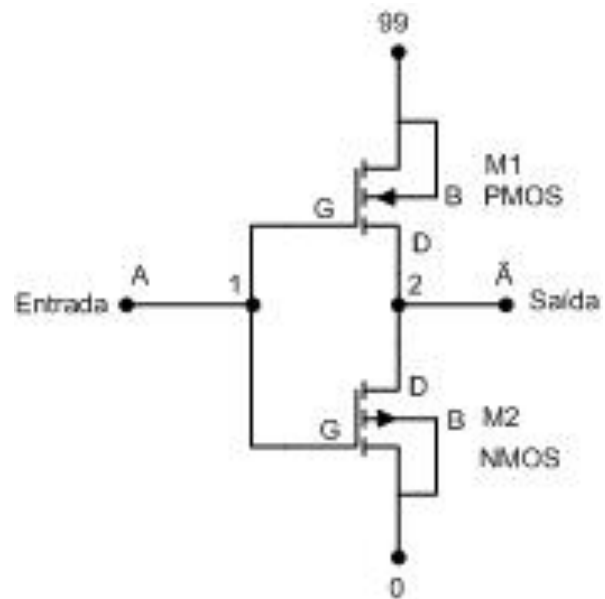
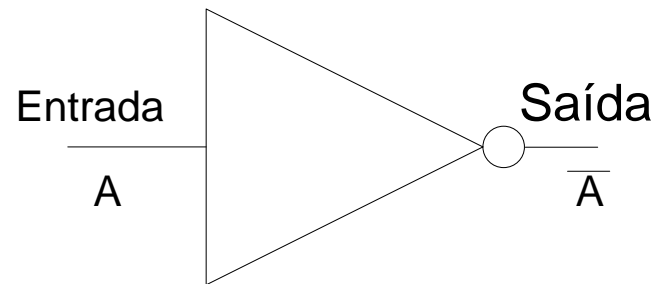
HASTINGS, Alan; HASTINGS, Roy Alan. **“The Art of Analog Layout”**. ISBN 0-13-146410-8, Prentice Hall, 2 ed. 2005;

JOHNS, David A.; MARTIN, Ken. **“Analog Integrated Circuit Design”**. ISBN 8-12-651778-6, Wiley India Pvt. Ltd., 2009.

ANEXO A

PORTA LÓGICA INVERSORA (NOT)

Símbolo lógico e diagrama eletrônico



PORTA LÓGICA INVERSORA (NOT)

Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
* Porta lógica NOT
*****

*Início do código

*ENTRADA
*Fonte de alimentação
Vdd 99 0 5V

Vf1 1 0 pulse 5V 0 (10n .1n .1n 200n 400n)

***** Main Circuit *****
x1 1 2 99 0 NOT

***** Sub - Circuits *****
.subckt NOT in out vdd vss
M1 out in vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M2 out in vss vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NOT

.MODEL NTRAN NMOS (
+LEVEL = 2
+UO = 608.3
+VTO = 825.3E-3
+NFS = 0.105E+12
+TPG = 1.000
+TOX = 40.00E-9
+NSUB = 7.755E+15
+UCRIT = 50.00E+3
+UEXP = 78.26E-3
+VMAX = 49.89E+3
+RSH = 50.15
+XJ = 450.0E-9
+LD = 112.1E-9
+DELTA = 3.714
+PB = 0.440

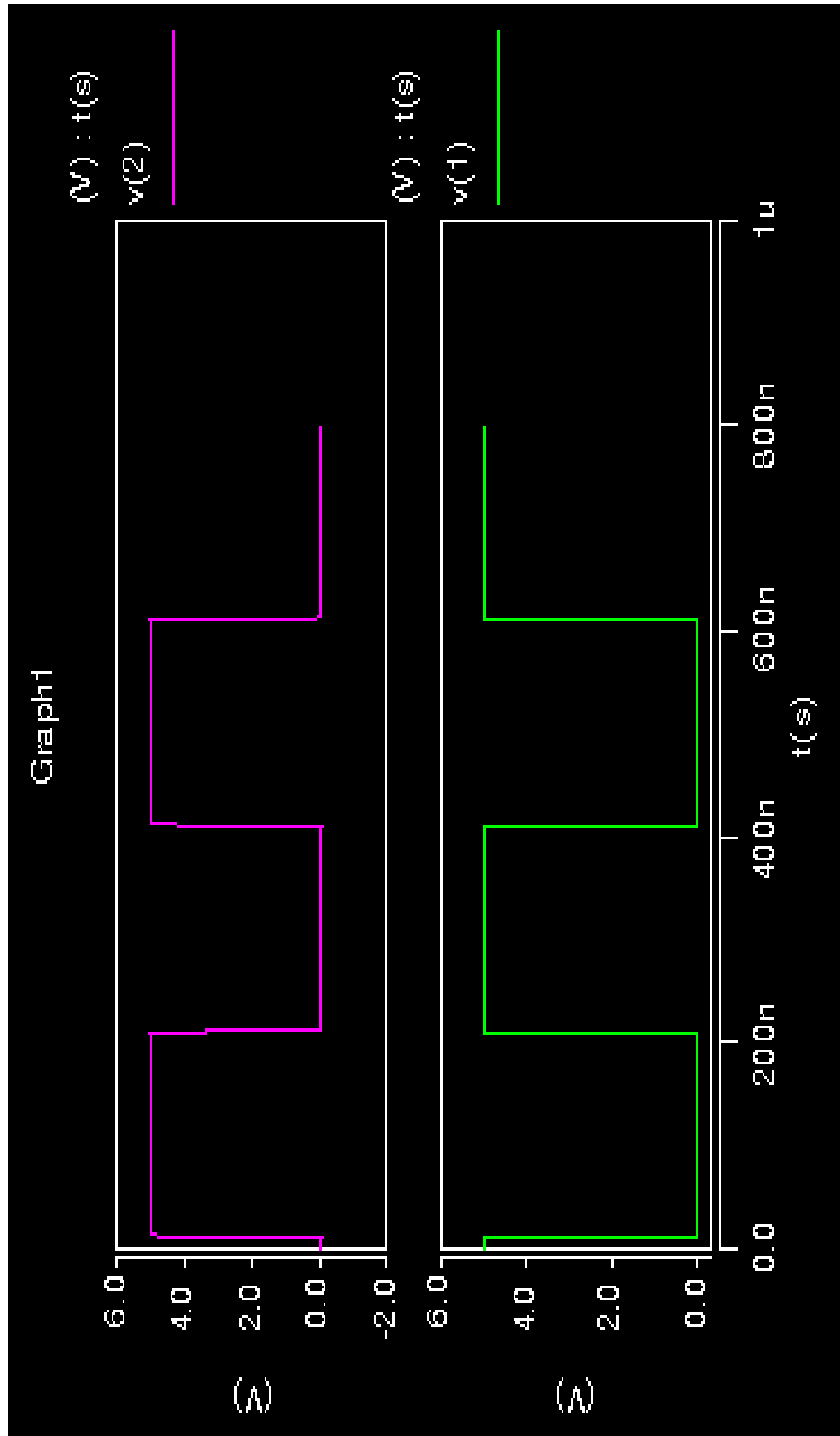
```

```
+JS = 10.00E-6
+NEFF = 3.358
+CJ = 323.1E-6
+MJ = 461.5E-3
+CJSW = 929.9E-12
+MJSW = 268.3E-3
+CGSO = 96.77E-12
+CGDO = 96.77E-12
+CGBO = 40.00E-12
+FC = 500.0E-3
+KF = 5.210E-28)

*
.MODEL PTRAN PMOS(
+ LEVEL = 2
+ UO = 205.1
+ VTO = -703.0E-3
+ NFS = 0.01E+12
+ TPG = -1.00
+ TOX = 40.00E-9
+ NSUB = 1.486E+16
+ UCRIT = 70.00E+3
+ UEXP = 184.2E-3
+ VMAX = 40.76E+3
+ RSH = 69.46
+ XJ = 450.0E-9
+ LD = 230.5E-9
+ DELTA = 1.843
+ PB = 0.958
+ JS = 10.00E-6
+ NEFF = 0.688
+ CJ = 804.9E-6
+ MJ = 525.0E-3
+ CJSW = 749.1E-12
+ MJSW = 495.4E-3
+ CGSO = 199.0E-12
+ CGDO = 199.0E-12
+ CGBO = 101.E-12
+ FC = 500.E-3
+ KF = 8.600E-30)

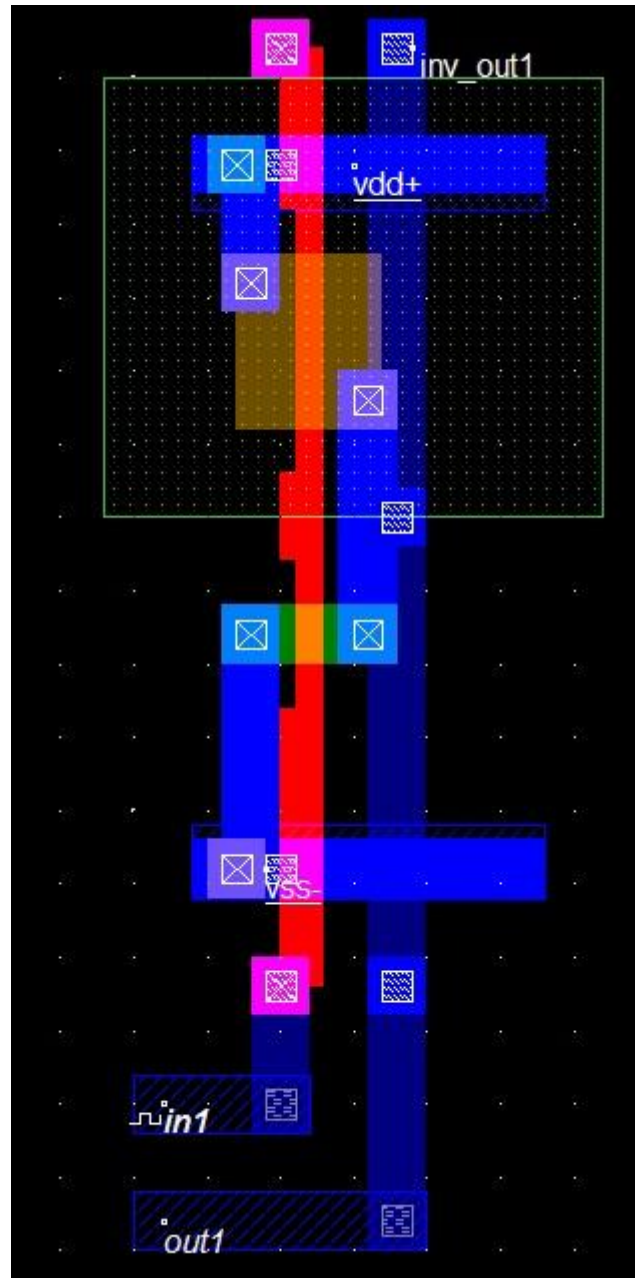
.tran 0.1n 800n
.options node nopage post
.end
```

PORTA LÓGICA INVERSORA (NOT)
Resultado da simulação SPICE

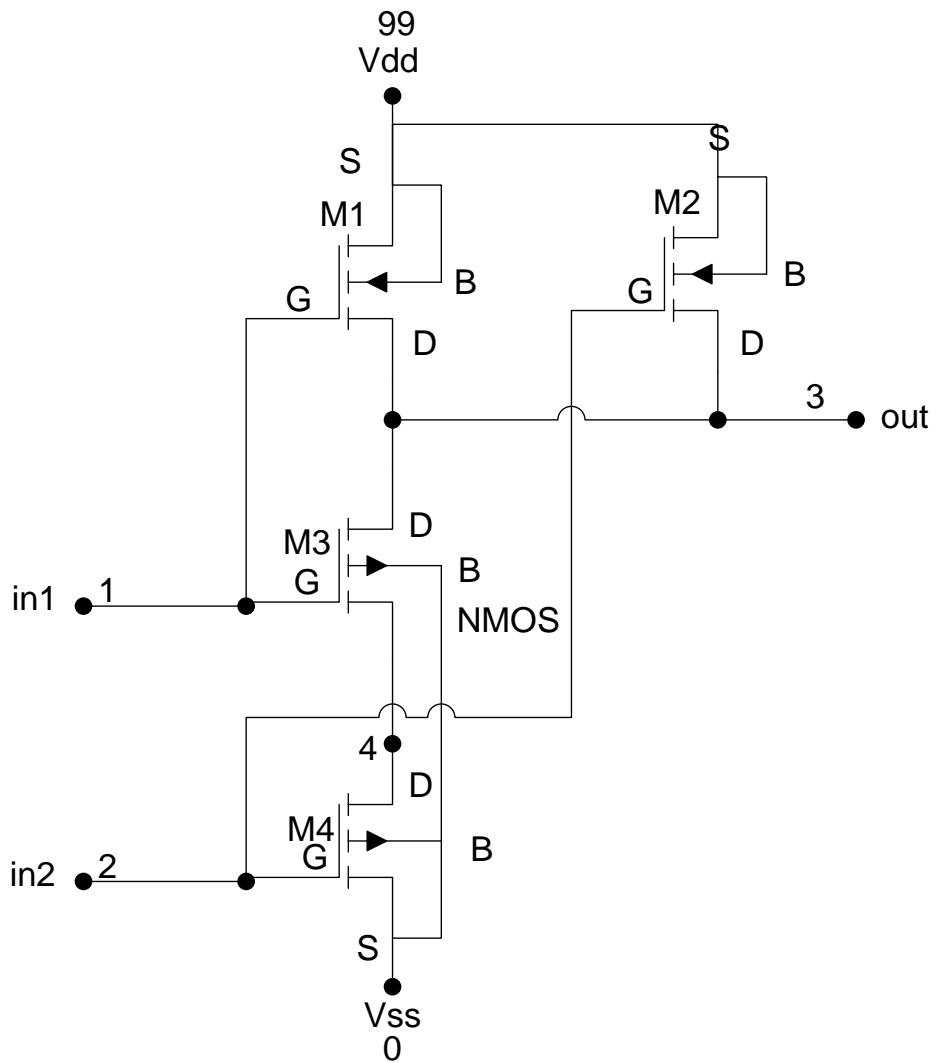
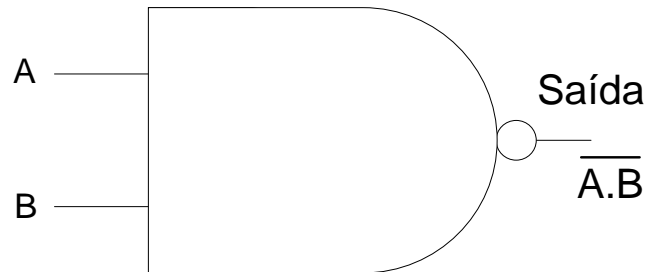


PORTA LÓGICA INVERSORA (NOT)

Layout



PORTA LÓGICA NAND
Símbolo lógico e diagrama eletrônico



PORTA LÓGICA NAND

Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
*Porta Nand - Entradas: 1 2 e saída 3
*****

*Inicio do código

*ENTRADA
*Fonte de alimentação
Vdd 99 0 5V

*Fontes para teste

Vf1 1 0 pulse 5V 0 (10n .1n .1n 12m 25m)
Vf2 2 0 pulse 5V 0 (10n .1n .1n 25m 50m)

***** Main Circuit *****

x1 1 2 3 99 0 NAND

***** 1 2 3 99 0
.subckt NAND in1 in2 out Vdd Vss
M1 out in1 Vdd Vdd PTRAN L=2.0U W=6U AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M2 out in2 Vdd Vdd PTRAN L=2.0U W=6U AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M3 out in1 4 Vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
M4 4 in2 Vss Vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NAND

***** Sub - Circuits *****
.subckt NOT in out vdd vss

M1 out in vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M2 out in vss vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NOT

```

```
.MODEL NTRAN NMOS (
```

```
+LEVEL = 2
```

```
+UO = 608.3
```

```
+VTO = 825.3E-3
```

```
+NFS = 0.105E+12
```

```
+TPG = 1.000
```

```
+TOX = 40.00E-9
```

```
+NSUB = 7.755E+15
```

```
+UCRIT = 50.00E+3
```

```
+UEXP = 78.26E-3
```

```
+VMAX = 49.89E+3
```

```
+RSH = 50.15
```

```
+XJ = 450.0E-9
```

```
+LD = 112.1E-9
```

```
+DELTA = 3.714
```

```
+PB = 0.440
```

```
+JS = 10.00E-6
```

```
+NEFF = 3.358
```

```
+CJ = 323.1E-6
```

```
+MJ = 461.5E-3
```

```
+CJSW = 929.9E-12
```

```
+MJSW = 268.3E-3
```

```
+CGSO = 96.77E-12
```

```
+CGDO = 96.77E-12
```

```
+CGBO = 40.00E-12
```

```
+FC = 500.0E-3
```

```
+KF = 5.210E-28)
```

```
*
```

```
.MODEL PTRAN PMOS(
```

```
+LEVEL = 2
```

```
+UO = 205.1
```

```
+VTO = -703.0E-3
```

```
+NFS = 0.01E+12
```

```
+TPG = -1.00
```

```
+TOX = 40.00E-9
```

```
+NSUB = 1.486E+16
```

```
+UCRIT = 70.00E+3
```

```
+UEXP = 184.2E-3
```

```
+VMAX = 40.76E+3
```

```
+RSH = 69.46
```

```
+XJ = 450.0E-9
```

```
+LD = 230.5E-9
```

```
+DELTA = 1.843
```

```
+PB = 0.958
```

```
+JS = 10.00E-6
```

```
+NEFF = 0.688
```

```
+CJ = 804.9E-6
```

```
+MJ = 525.0E-3
```

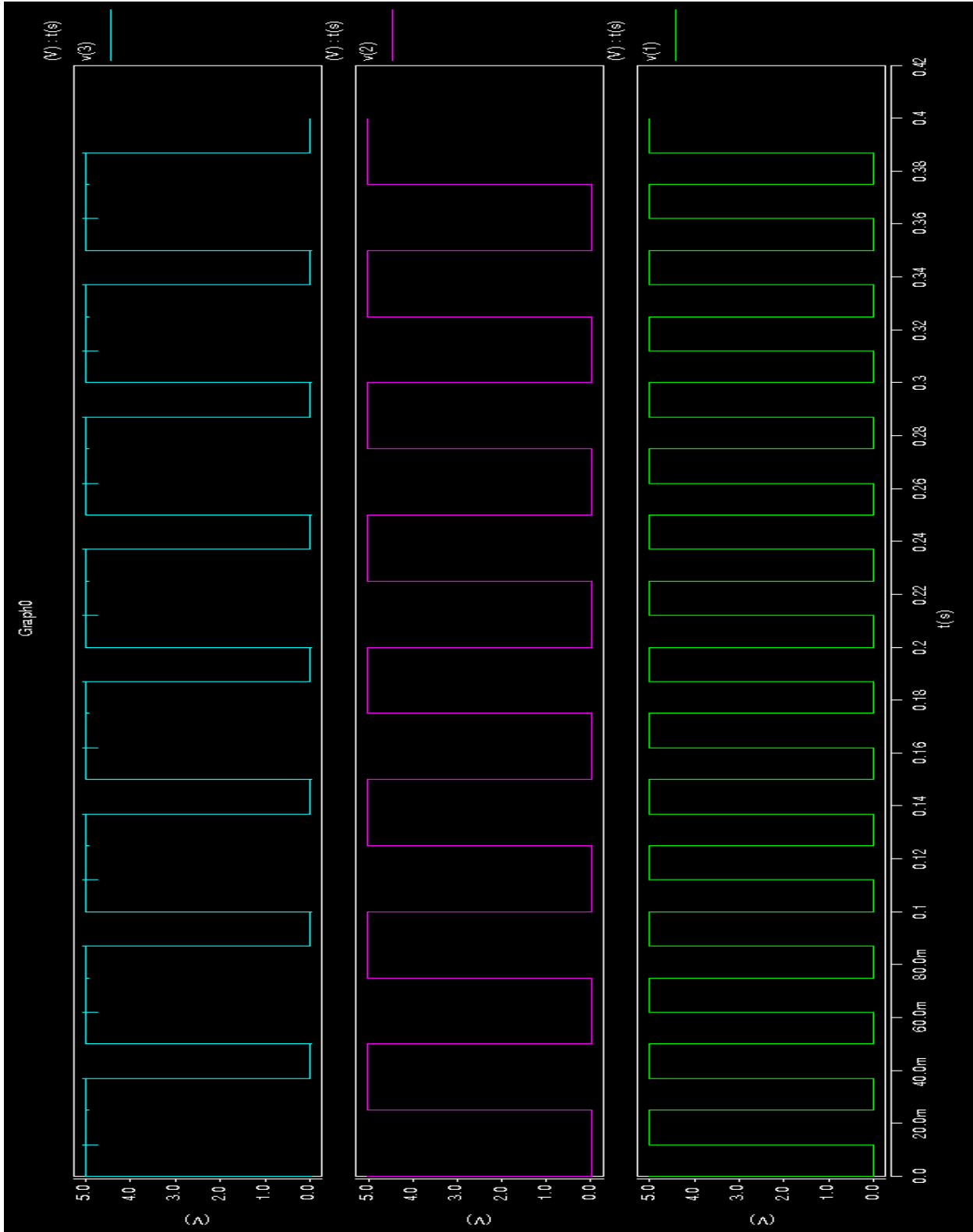
```
+ CJSW = 749.1E-12
+ MJSW = 495.4E-3
+ CGSO = 199.0E-12
+ CGDO = 199.0E-12
+ CGBO = 101.E-12
+ FC   = 500.E-3
+ KF   = 8.600E-30)

.tran 0.1ms 400ms
.options node nopage post

.end
```

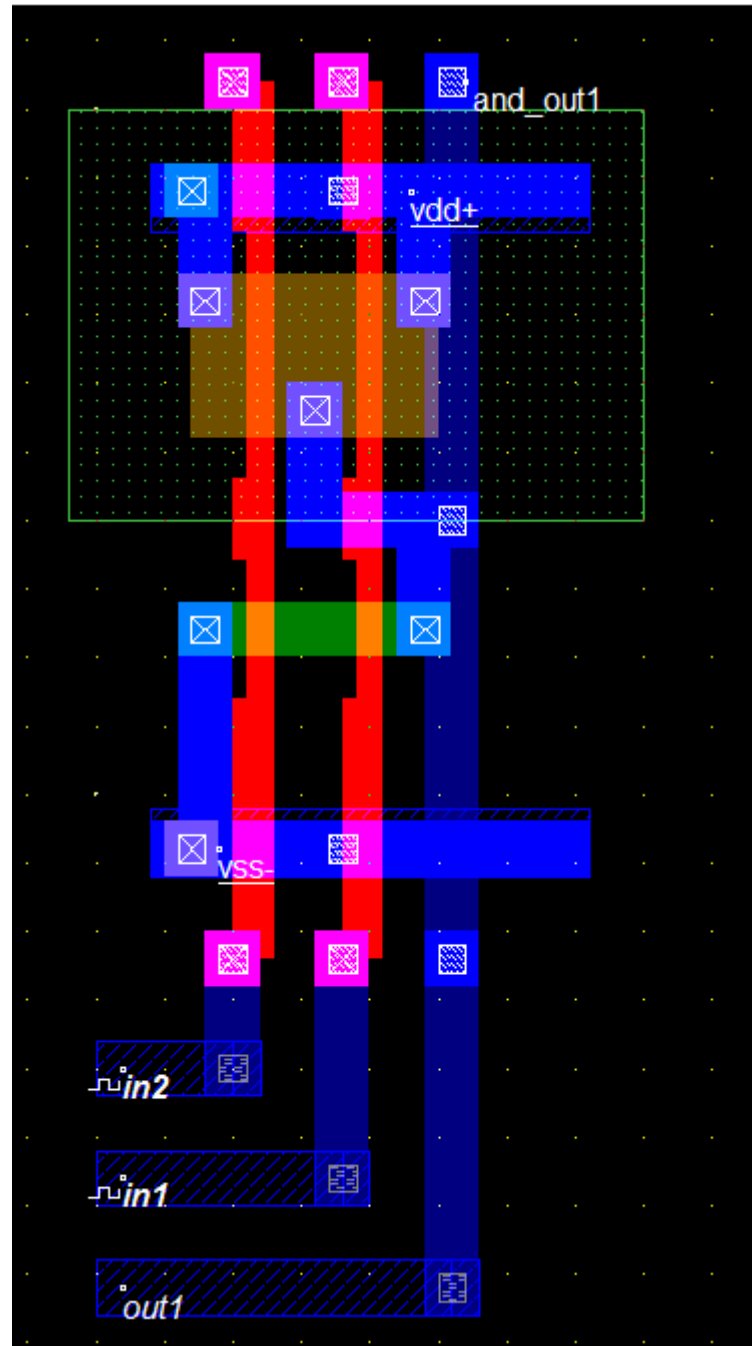
PORTA LÓGICA NAND

Resultado da simulação SPICE

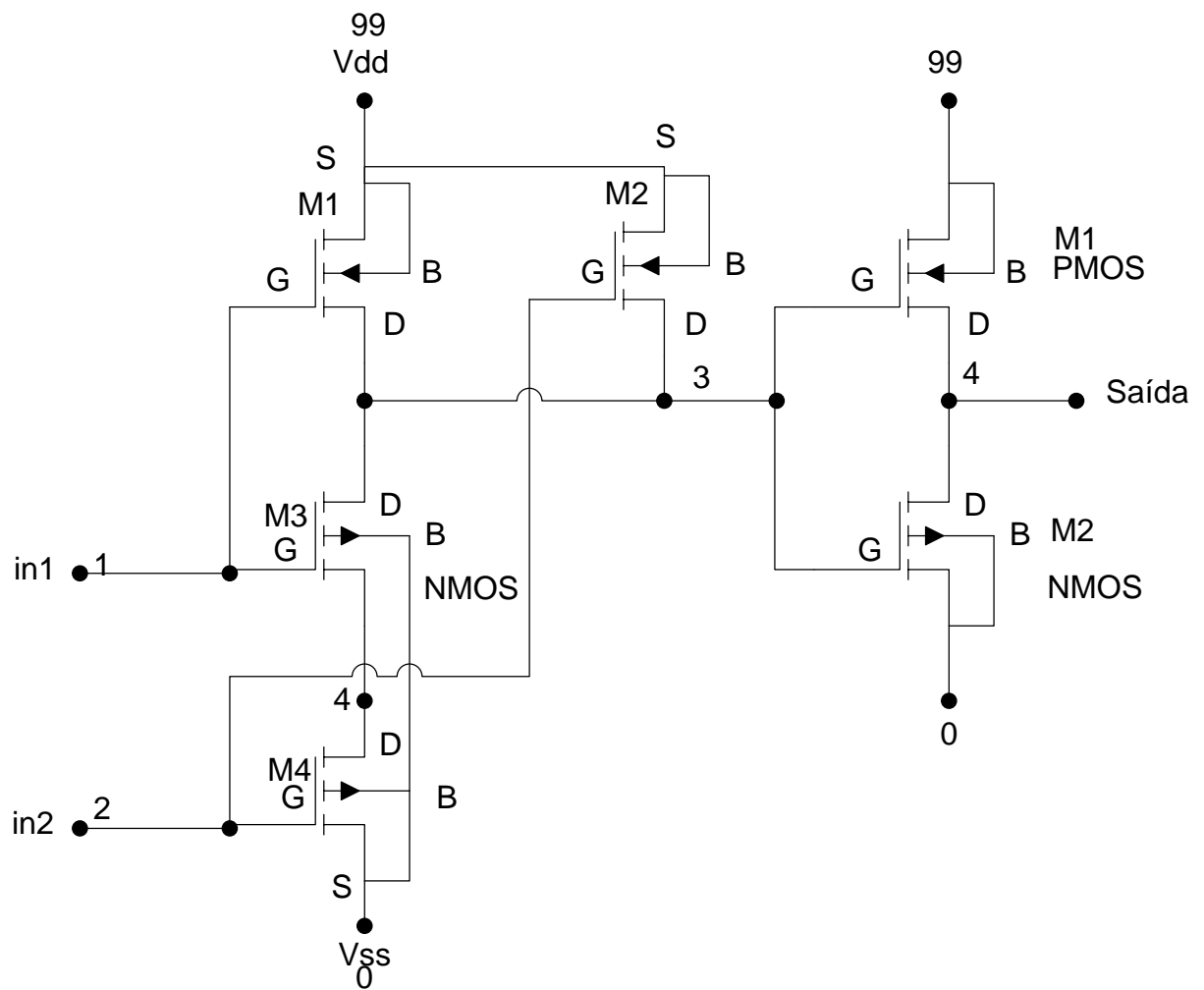
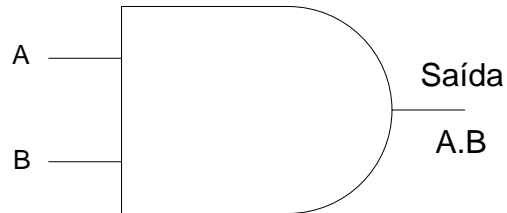


PORTA LÓGICA NAND

Layout



PORTA LÓGICA AND
Símbolo lógico e diagrama eletrônico



PORTA LÓGICA AND
Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
*Porta And - Entradas: 1 2 e saída 4
*****

*Inicio do código

*ENTRADA
*Fonte de alimentação
Vdd 99 0 5V

*Fontes para teste

Vf1 1 0 pulse 5V 0 (10n .1n .1n 12m 25m)
Vf2 2 0 pulse 5V 0 (10n .1n .1n 25m 50m)

***** Main Circuit *****

x1 1 2 3 99 0 NAND
x2 3 4 99 0 NOT

***** Sub - Circuits *****

***** 1 2 3 99 0
.subckt NAND in1 in2 out Vdd Vss
M1 out in1 Vdd Vdd PTRAN L=2.0U W=6U AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M2 out in2 Vdd Vdd PTRAN L=2.0U W=6U AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M3 out in1 4 Vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
M4 4 in2 Vss Vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NAND

.subckt NOT in out vdd vss
M1 out in vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M2 out in vss vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NOT

```



```
.MODEL NTRAN NMOS (
```

```
+LEVEL = 2
```

```
+UO = 608.3
```

```
+VTO = 825.3E-3
```

```
+NFS = 0.105E+12
```

```
+TPG = 1.000
```

```
+TOX = 40.00E-9
```

```
+NSUB = 7.755E+15
```

```
+UCRIT = 50.00E+3
```

```
+UEXP = 78.26E-3
```

```
+VMAX = 49.89E+3
```

```
+RSH = 50.15
```

```
+XJ = 450.0E-9
```

```
+LD = 112.1E-9
```

```
+DELTA = 3.714
```

```
+PB = 0.440
```

```
+JS = 10.00E-6
```

```
+NEFF = 3.358
```

```
+CJ = 323.1E-6
```

```
+MJ = 461.5E-3
```

```
+CJSW = 929.9E-12
```

```
+MJSW = 268.3E-3
```

```
+CGSO = 96.77E-12
```

```
+CGDO = 96.77E-12
```

```
+CGBO = 40.00E-12
```

```
+FC = 500.0E-3
```

```
+KF = 5.210E-28)
```

```
*
```

```
.MODEL PTRAN PMOS(
```

```
+LEVEL = 2
```

```
+UO = 205.1
```

```
+VTO = -703.0E-3
```

```
+NFS = 0.01E+12
```

```
+TPG = -1.00
```

```
+TOX = 40.00E-9
```

```
+NSUB = 1.486E+16
```

```
+UCRIT = 70.00E+3
```

```
+UEXP = 184.2E-3
```

```
+VMAX = 40.76E+3
```

```
+RSH = 69.46
```

```
+XJ = 450.0E-9
```

```
+LD = 230.5E-9
```

```
+DELTA = 1.843
```

```
+PB = 0.958
```

```
+JS = 10.00E-6
```

```
+NEFF = 0.688
```

```
+CJ = 804.9E-6
```

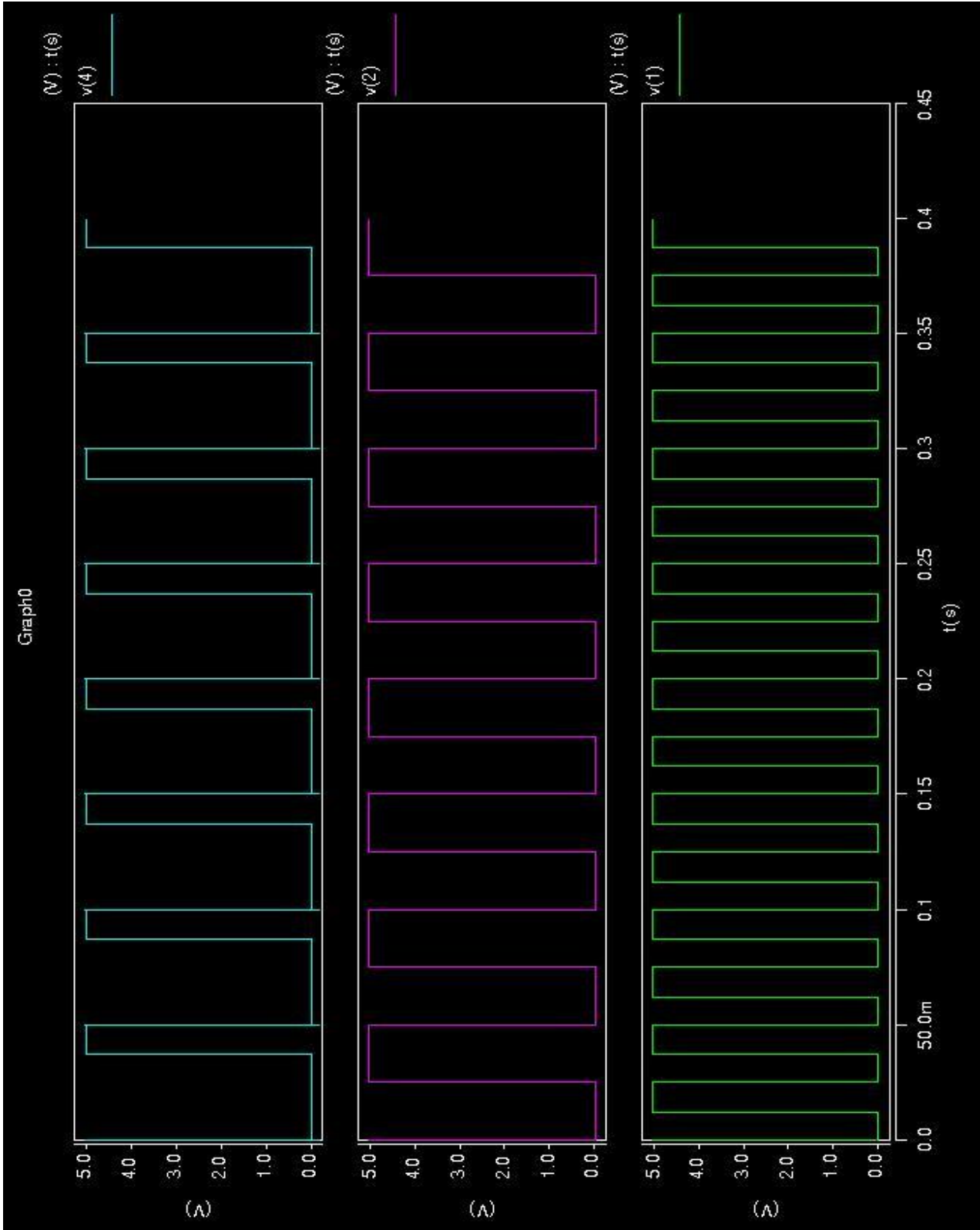
```
+MJ = 525.0E-3
```

```
+ CJSW = 749.1E-12
+ MJSW = 495.4E-3
+ CGSO = 199.0E-12
+ CGDO = 199.0E-12
+ CGBO = 101.E-12
+ FC   = 500.E-3
+ KF   = 8.600E-30)

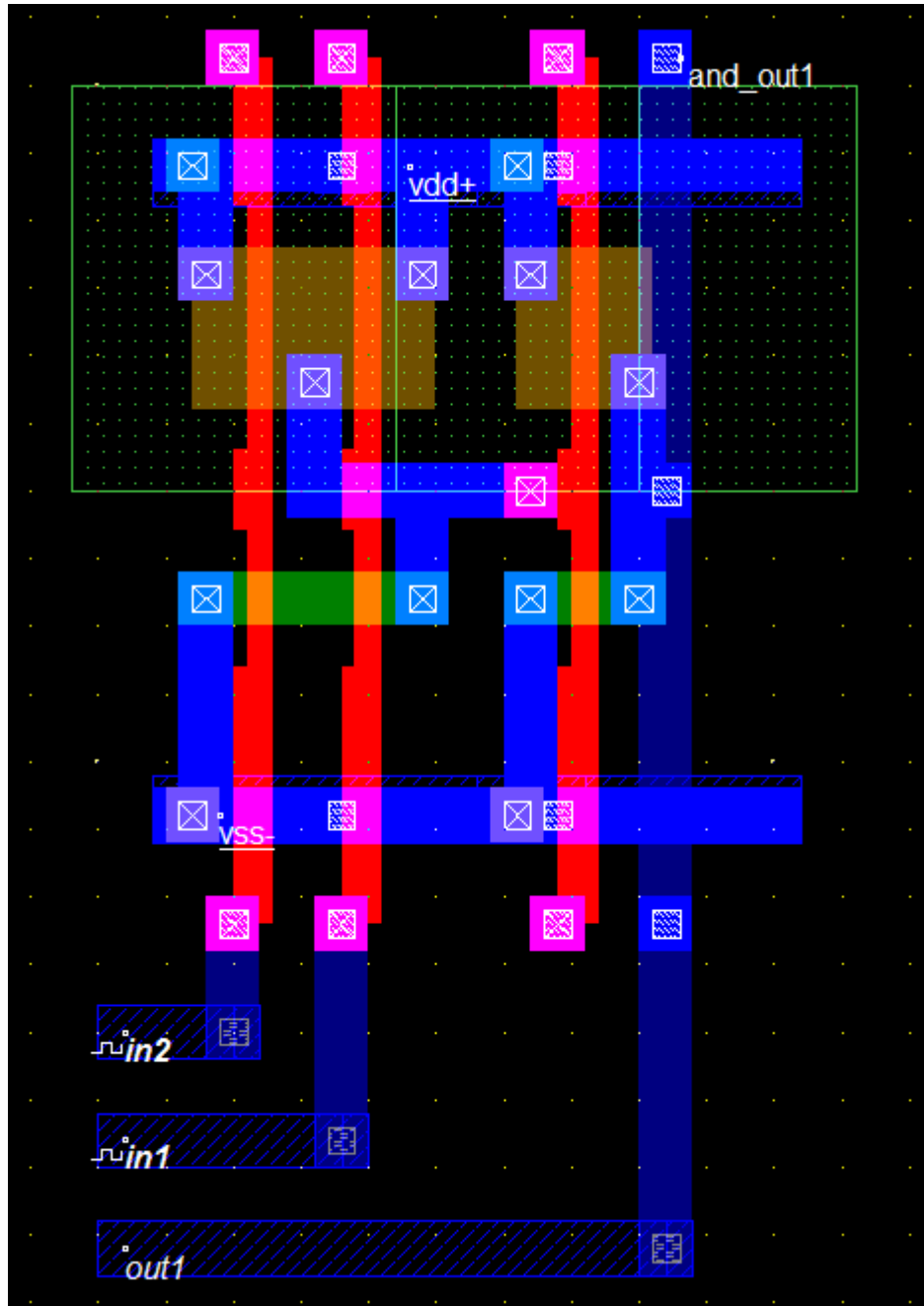
.tran 0.1ms 400ms
.options node nopage post

.end
```

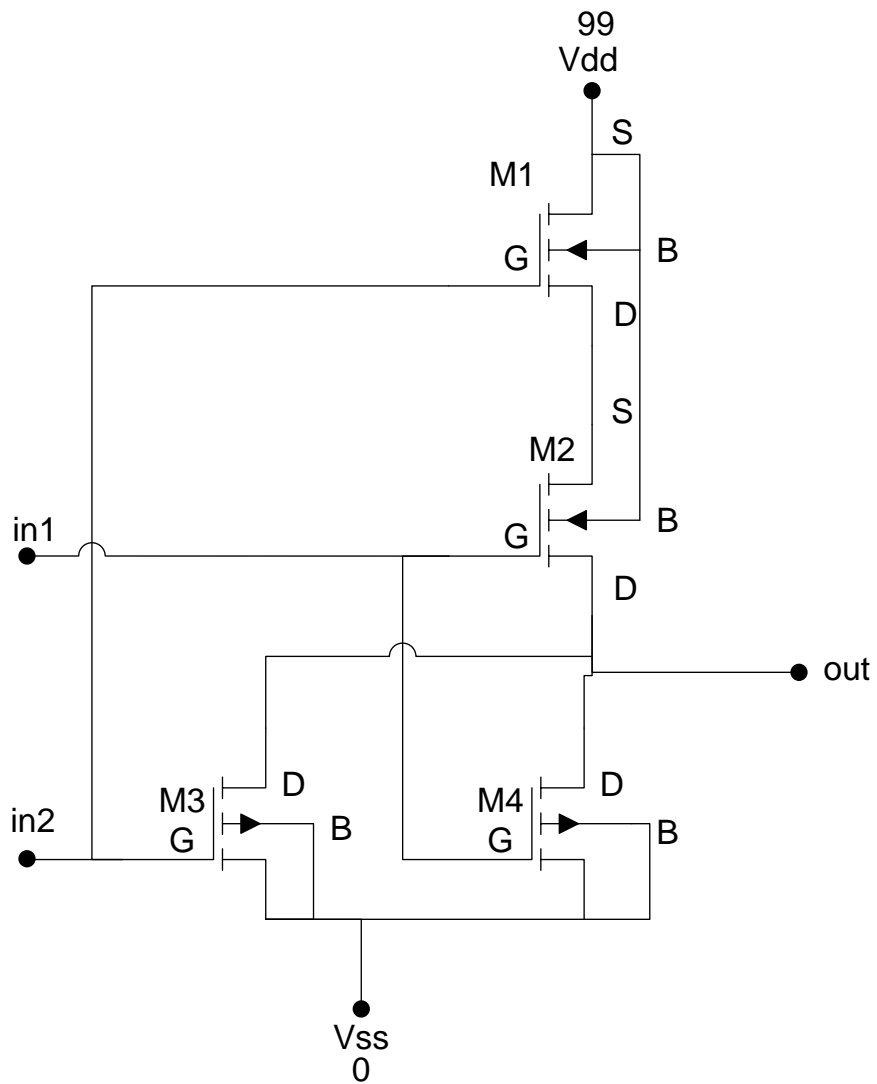
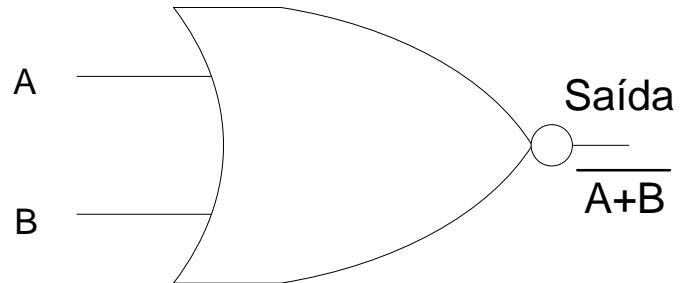
PORTA LÓGICA AND
Resultado da simulação SPICE



PORTA LÓGICA AND Layout



PORTA LÓGICA NOR
Símbolo lógico e diagrama eletrônico



PORTA LÓGICA NOR
Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
*Porta OR - Entradas: 1 2 e saída 3
*****

*Inicio do código

*ENTRADA
*Fonte de alimentação
Vdd 99 0 5V

*Fontes para teste
Vf1 1 0 pulse 5V 0 (10n .1n .1n 10m 20m)
Vf2 2 0 pulse 5V 0 (10n .1n .1n 20m 40m)

***** Main Circuit *****

x1 1 2 3 99 0 NOR

***** 1 2 3 99 0
.subckt NOR in1 in2 out Vdd Vss
*   D   G   S   B
M1   3   in1  Vdd  Vdd PTRAN  L=2.0U W=6U  AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M2   out  in2   3  Vdd PTRAN  L=2.0U W=6U  AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M3   out  in1  Vss  Vss NTRAN  L=2.0U W=3U  AD=75P  AS=75P  PD=32U
PS=32U NRD=0.33 NRS=0.33
M4   out  in2  Vss  Vss NTRAN  L=2.0U W=3U  AD=75P  AS=75P  PD=32U
PS=32U NRD=0.33 NRS=0.33

.ends NOR

***** Sub - Circuits *****
.subckt NOT in out vdd vss

M1   out  in  vdd  vdd PTRAN  L=2.0U W=6U  AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M2   out  in  vss  vss NTRAN  L=2.0U W=3U  AD=75P  AS=75P  PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NOT

```

```
.MODEL NTRAN NMOS (
```

```
+LEVEL = 2
```

```
+UO = 608.3
```

```
+VTO = 825.3E-3
```

```
+NFS = 0.105E+12
```

```
+TPG = 1.000
```

```
+TOX = 40.00E-9
```

```
+NSUB = 7.755E+15
```

```
+UCRIT = 50.00E+3
```

```
+UEXP = 78.26E-3
```

```
+VMAX = 49.89E+3
```

```
+RSH = 50.15
```

```
+XJ = 450.0E-9
```

```
+LD = 112.1E-9
```

```
+DELTA = 3.714
```

```
+PB = 0.440
```

```
+JS = 10.00E-6
```

```
+NEFF = 3.358
```

```
+CJ = 323.1E-6
```

```
+MJ = 461.5E-3
```

```
+CJSW = 929.9E-12
```

```
+MJSW = 268.3E-3
```

```
+CGSO = 96.77E-12
```

```
+CGDO = 96.77E-12
```

```
+CGBO = 40.00E-12
```

```
+FC = 500.0E-3
```

```
+KF = 5.210E-28)
```

```
*
```

```
.MODEL PTRAN PMOS(
```

```
+LEVEL = 2
```

```
+UO = 205.1
```

```
+VTO = -703.0E-3
```

```
+NFS = 0.01E+12
```

```
+TPG = -1.00
```

```
+TOX = 40.00E-9
```

```
+NSUB = 1.486E+16
```

```
+UCRIT = 70.00E+3
```

```
+UEXP = 184.2E-3
```

```
+VMAX = 40.76E+3
```

```
+RSH = 69.46
```

```
+XJ = 450.0E-9
```

```
+LD = 230.5E-9
```

```
+DELTA = 1.843
```

```
+PB = 0.958
```

```
+JS = 10.00E-6
```

```
+NEFF = 0.688
```

```
+CJ = 804.9E-6
```

```
+MJ = 525.0E-3
```

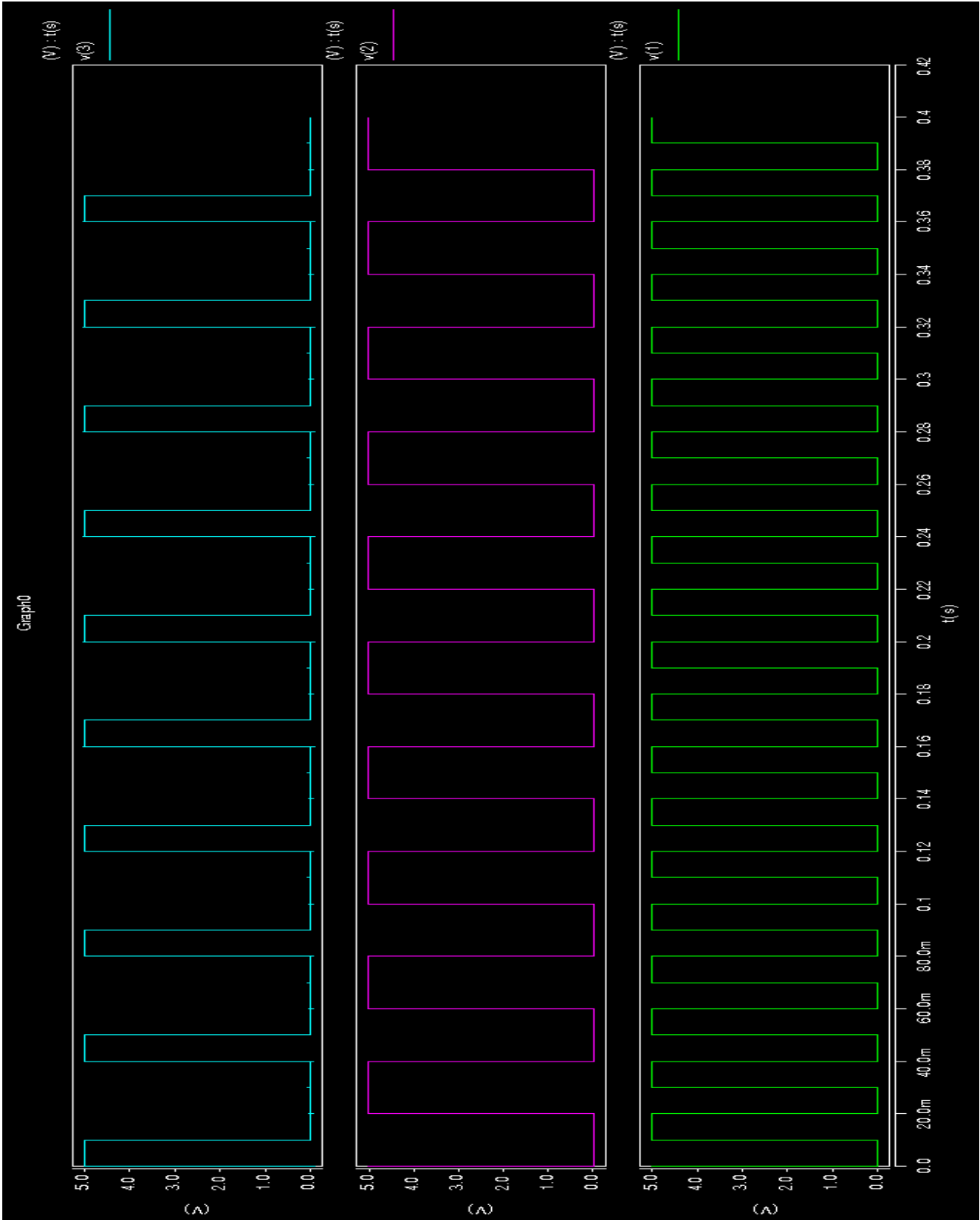
```
+ CJSW = 749.1E-12
+ MJSW = 495.4E-3
+ CGSO = 199.0E-12
+ CGDO = 199.0E-12
+ CGBO = 101.E-12
+ FC   = 500.E-3
+ KF   = 8.600E-30)

.tran 0.1ms 400ms
.options node nopage post

.end
```

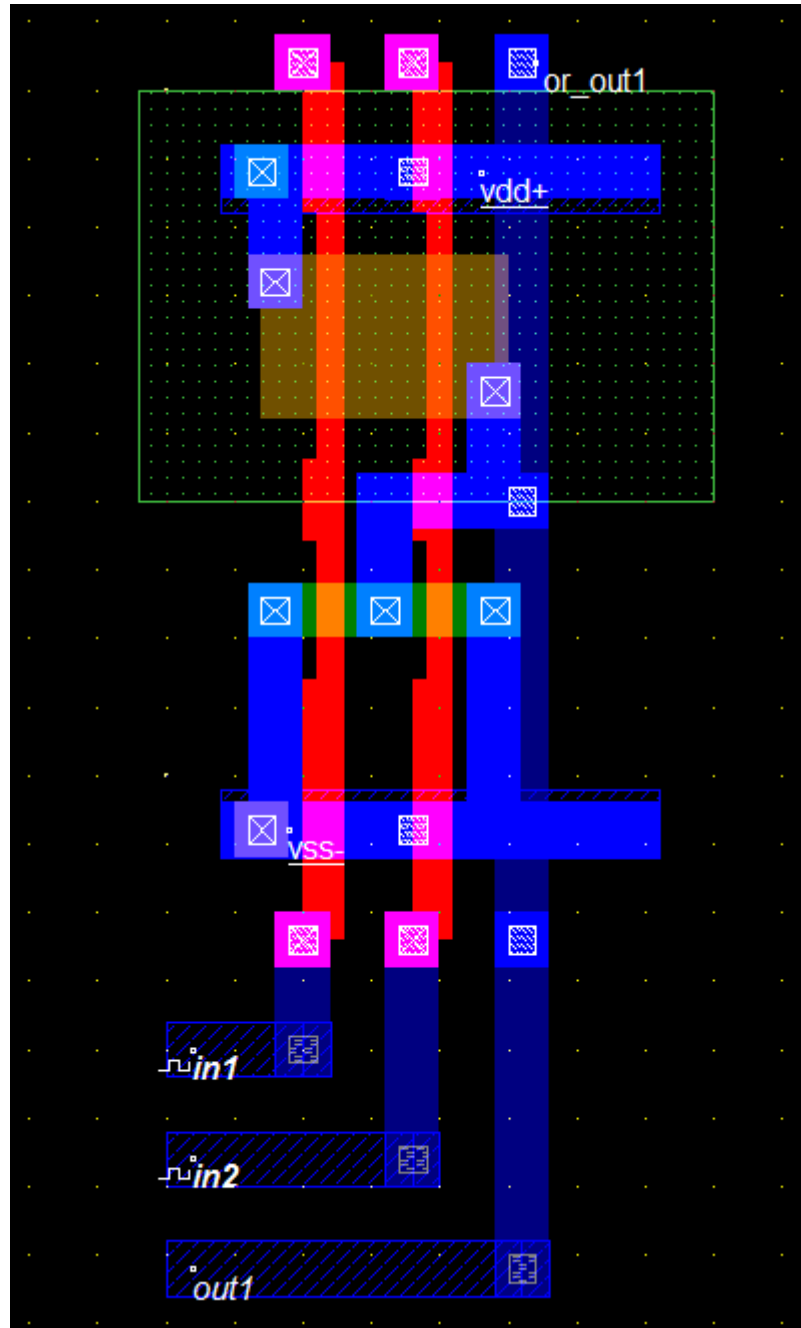

PORTA LÓGICA NOR

Resultado da simulação SPICE



PORTA LÓGICA NOR

Layout



PORTA LÓGICA OR

Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
* Porta OR - Entradas: 1 2 e saída 3
*****

*Início do código

*ENTRADA
*Fonte de alimentação
Vdd 99 0 5V

*Fontes para teste

Vf1 1 0 pulse 5V 0 (10n .1n .1n 10m 20m)
Vf2 2 0 pulse 5V 0 (10n .1n .1n 20m 40m)

***** Main Circuit *****
x1 1 2 3 99 0 OR

***** Sub - Circuits *****
.subckt OR in1 in2 out Vdd Vss
x1 in1 in2 3 Vdd Vss NOR
x2 3 out Vdd Vss NOT
.ends OR

***** 1 2 3 99 0
.subckt NOR in1 in2 out Vdd Vss
*   D   G   S   B
M1   3   in1   Vdd   Vdd PTRAN   L=2.0U W=6U   AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M2   out  in2   3   Vdd PTRAN   L=2.0U W=6U   AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M3   out  in1   Vss   Vss NTRAN   L=2.0U W=3U   AD=75P AS=75P
PD=32U PS=32U NRD=0.33 NRS=0.33
M4   out  in2   Vss   Vss NTRAN   L=2.0U W=3U   AD=75P AS=75P
PD=32U PS=32U NRD=0.33 NRS=0.33
.ends NOR

.subckt NOT in out vdd vss

```

```

M1 out in vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M2 out in vss vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NOT

```

```

.MODEL NTRAN NMOS (

```

```

+LEVEL = 2
+UO = 608.3
+VTO = 825.3E-3
+NFS = 0.105E+12
+TPG = 1.000
+TOX = 40.00E-9
+NSUB = 7.755E+15
+UCRIT = 50.00E+3
+UEXP = 78.26E-3
+VMAX = 49.89E+3
+RSH = 50.15
+XJ = 450.0E-9
+LD = 112.1E-9
+DELTA = 3.714
+PB = 0.440
+JS = 10.00E-6
+NEFF = 3.358
+CJ = 323.1E-6
+MJ = 461.5E-3
+CJSW = 929.9E-12
+MJSW = 268.3E-3
+CGSO = 96.77E-12
+CGDO = 96.77E-12
+CGBO = 40.00E-12
+FC = 500.0E-3
+KF = 5.210E-28)

```

```

*
```

```

.MODEL PTRAN PMOS(

```

```

+LEVEL = 2
+UO = 205.1
+VTO = -703.0E-3
+NFS = 0.01E+12
+TPG = -1.00
+TOX = 40.00E-9
+NSUB = 1.486E+16
+UCRIT = 70.00E+3
+UEXP = 184.2E-3
+VMAX = 40.76E+3
+RSH = 69.46
+XJ = 450.0E-9
+LD = 230.5E-9
+DELTA = 1.843

```

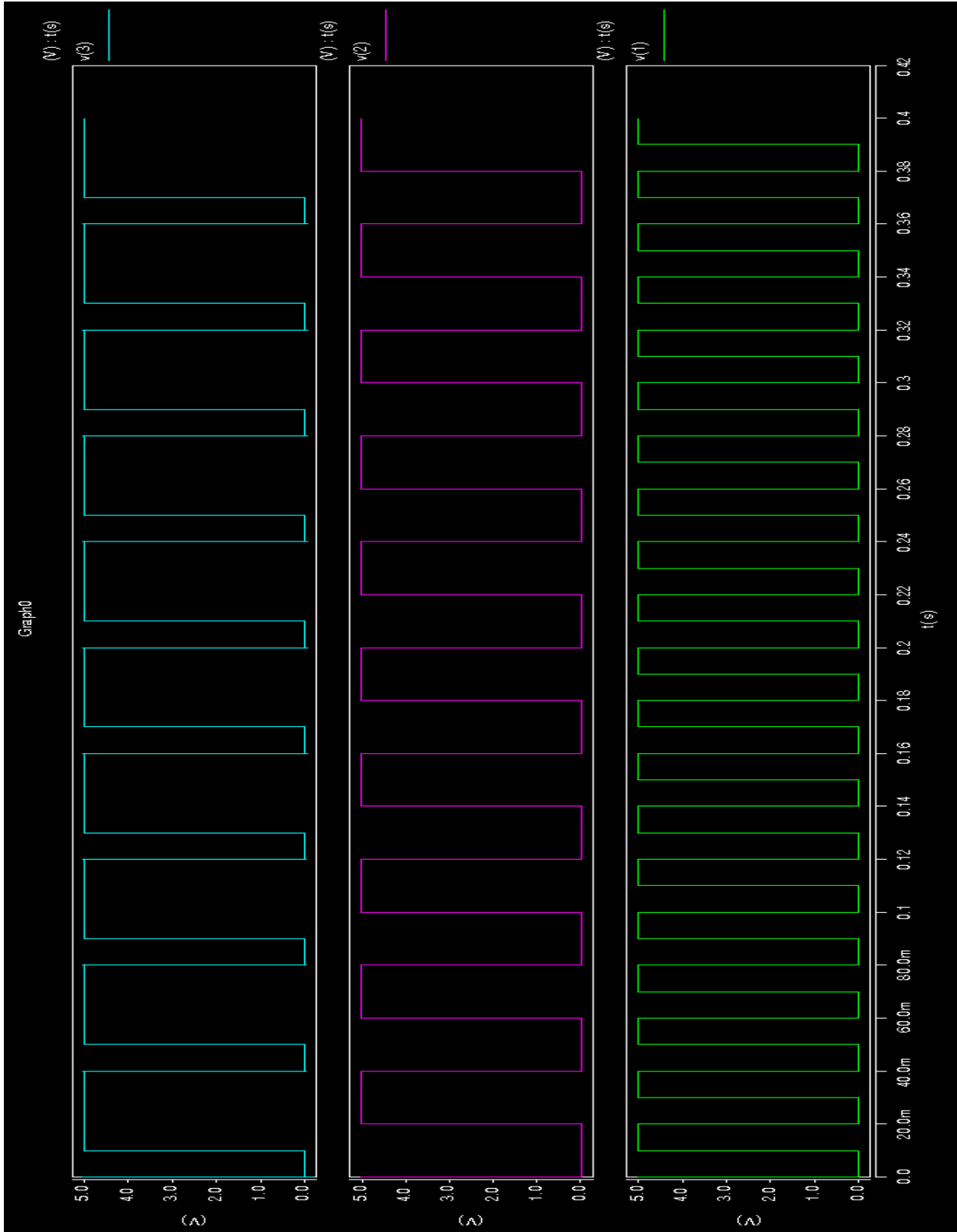
```
+ PB = 0.958
+ JS = 10.00E-6
+ NEFF = 0.688
+ CJ = 804.9E-6
+ MJ = 525.0E-3
+ CJSW = 749.1E-12
+ MJSW = 495.4E-3
+ CGSO = 199.0E-12
+ CGDO = 199.0E-12
+ CGBO = 101.E-12
+ FC = 500.E-3
+ KF = 8.600E-30)

.tran 0.1ms 400ms
.options node nopage post

.end
```

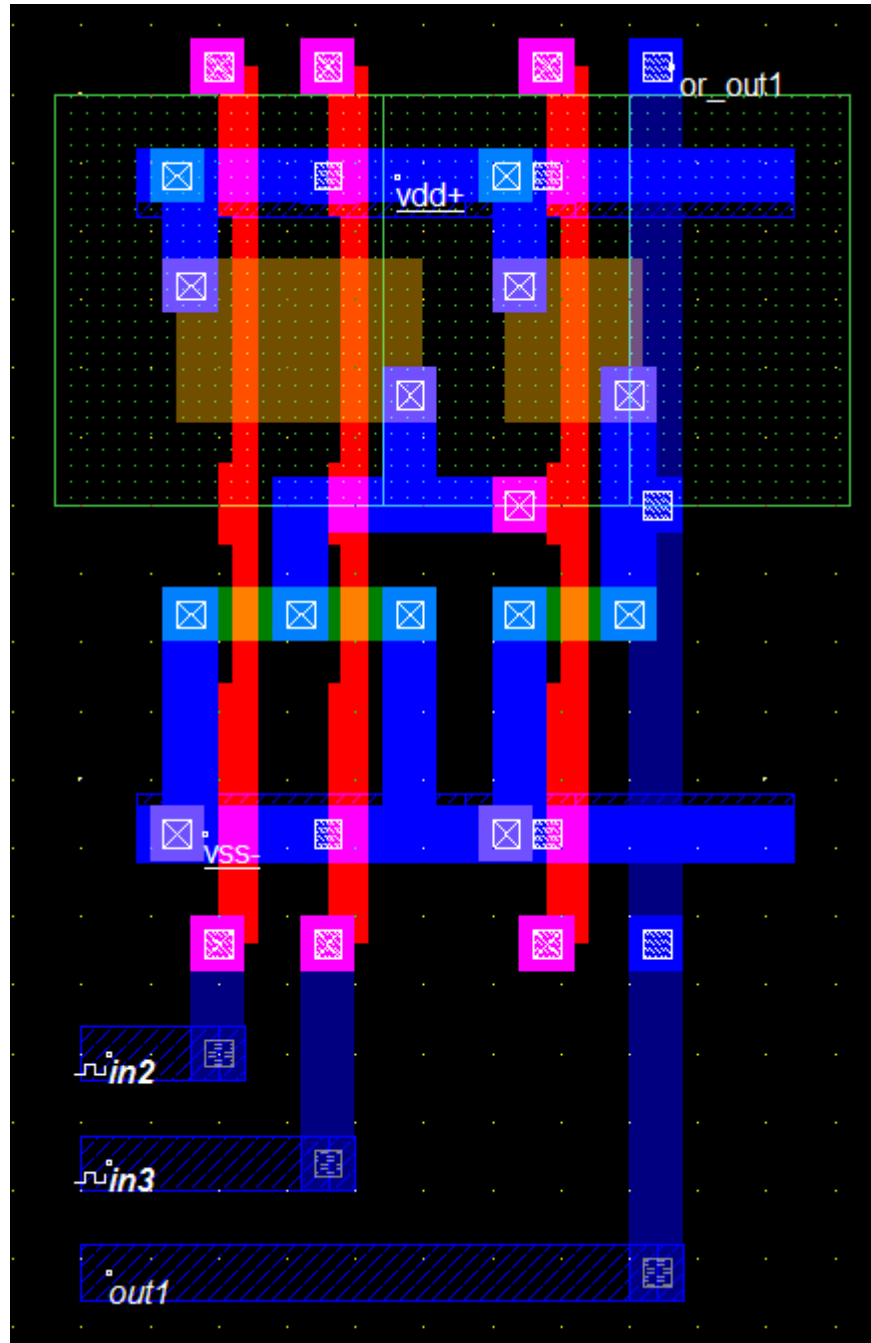
PORTA LÓGICA OR

Resultado da simulação SPICE



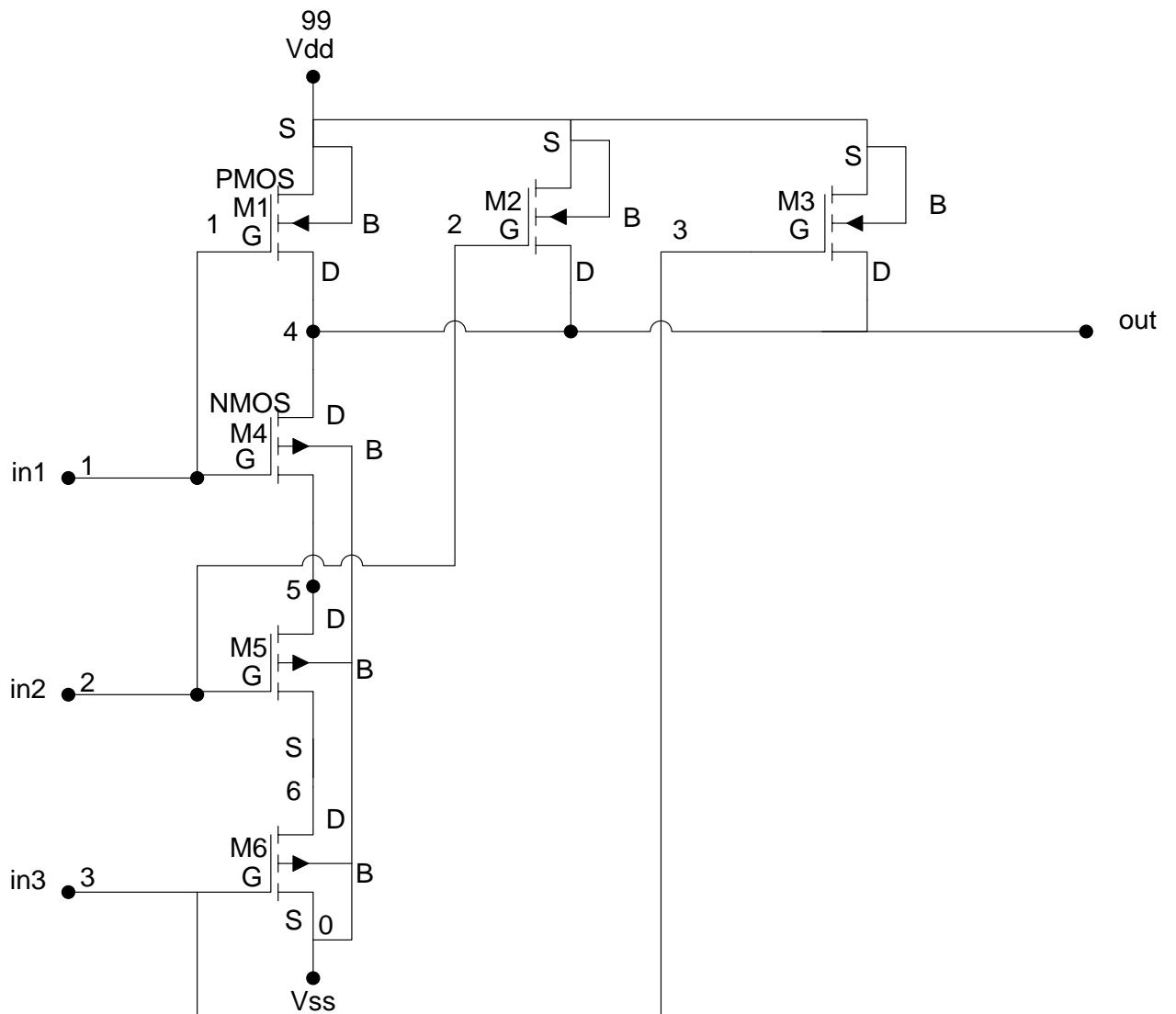
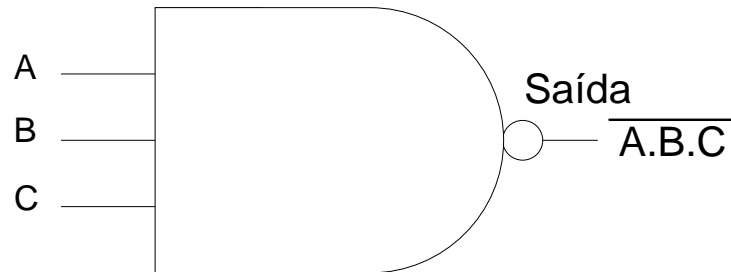
PORTA LÓGICA OR

Layout



PORTA LÓGICA NAND 3 ENTRADAS

Símbolo lógico e diagrama eletrônico



PORTA LÓGICA NAND 3 Entradas
Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
*Porta Nand - Entradas: 1 2 3 e saída 4
*****

*Inicio do código

*ENTRADA
*Fonte de alimentação
Vdd 99 0 5V

*Fontes para teste

Vf1 1 0 pulse 5V 0 (10n .1n .1n 12m 25m)
Vf2 2 0 pulse 5V 0 (10n .1n .1n 25m 50m)
Vf3 3 0 pulse 5V 0 (10n .1n .1n 50m 100m)

***** Main Circuit *****
x1 1 2 3 4 99 0 NAND3

***** Sub - Circuits *****
***** 1 2 3 4 99 0
.subckt NAND3 in1 in2 in3 out vdd vss
*   D   G   S   B
M1  out in1 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U PS=32U
NRD=0.33 NRS=0.33
M2  out in2 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U PS=32U
NRD=0.33 NRS=0.33
M3  out in3 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U PS=32U
NRD=0.33 NRS=0.33
M4  out in1 5 vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U PS=32U
NRD=0.33 NRS=0.33
M5  5 in2 6 vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U PS=32U
NRD=0.33 NRS=0.33
M6  6 in3 vss vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U PS=32U
NRD=0.33 NRS=0.33
.ends NAND3

.MODEL NTRAN NMOS (
+LEVEL = 2

```

```

+UO = 608.3
+VTO = 825.3E-3
+NFS = 0.105E+12
+TPG = 1.000
+TOX = 40.00E-9
+NSUB = 7.755E+15
+UCRIT = 50.00E+3
+UEXP = 78.26E-3
+VMAX = 49.89E+3
+RSH = 50.15
+XJ = 450.0E-9
+LD = 112.1E-9
+DELTA = 3.714
+PB = 0.440
+JS = 10.00E-6
+NEFF = 3.358
+CJ = 323.1E-6
+MJ = 461.5E-3
+CJSW = 929.9E-12
+MJSW = 268.3E-3
+CGSO = 96.77E-12
+CGDO = 96.77E-12
+CGBO = 40.00E-12
+FC = 500.0E-3
+KF = 5.210E-28)

```

*

```

.MODEL PTRAN PMOS(
+ LEVEL = 2
+ UO = 205.1
+ VTO = -703.0E-3
+ NFS = 0.01E+12
+ TPG = -1.00
+ TOX = 40.00E-9
+ NSUB = 1.486E+16
+ UCRIT = 70.00E+3
+ UEXP = 184.2E-3
+ VMAX = 40.76E+3
+ RSH = 69.46
+ XJ = 450.0E-9
+ LD = 230.5E-9
+ DELTA = 1.843
+ PB = 0.958
+ JS = 10.00E-6
+ NEFF = 0.688
+ CJ = 804.9E-6
+ MJ = 525.0E-3
+ CJSW = 749.1E-12
+ MJSW = 495.4E-3
+ CGSO = 199.0E-12

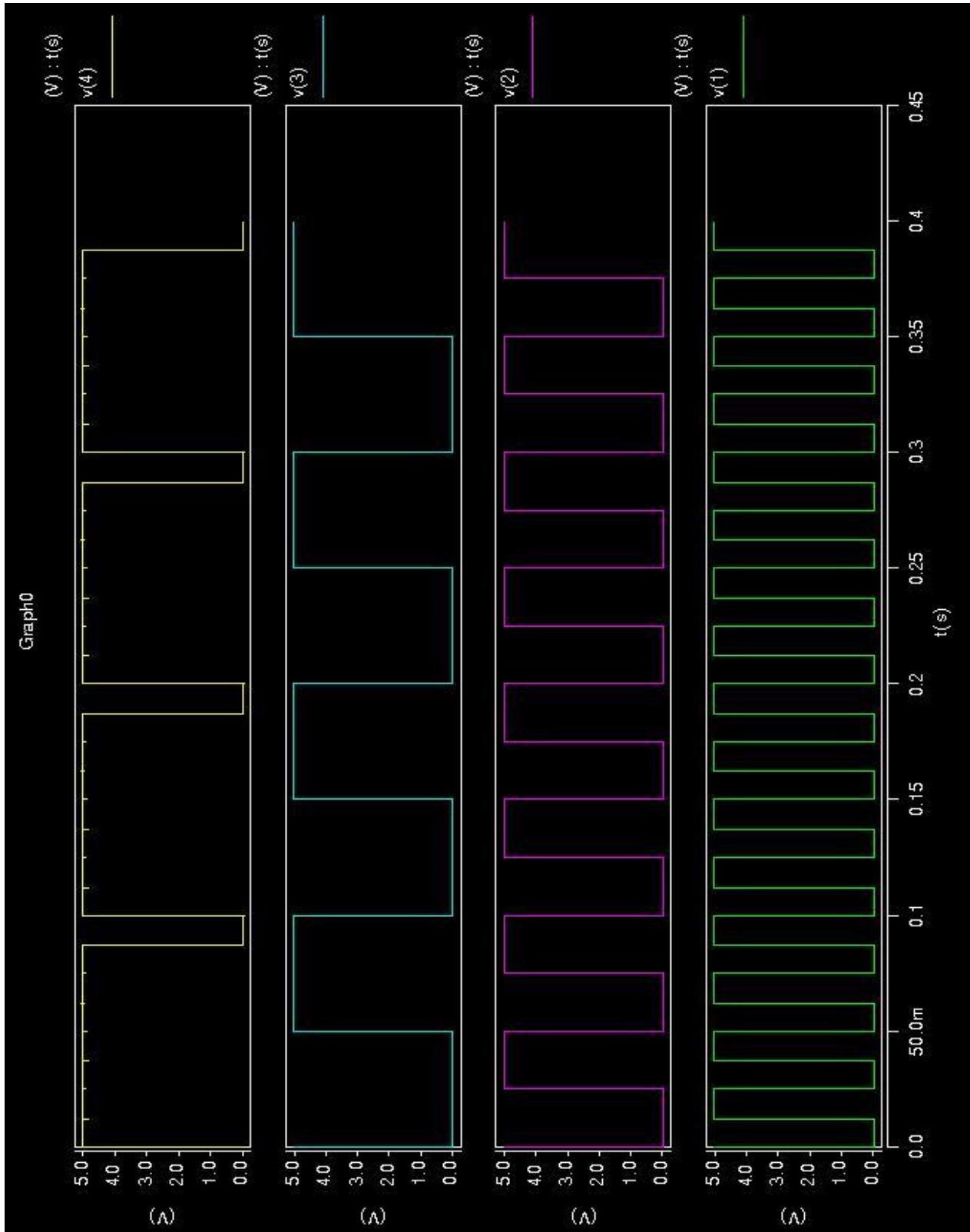
```

```
+ CGDO = 199.0E-12
+ CGBO = 101.E-12
+ FC   = 500.E-3
+ KF   = 8.600E-30)

.tran 0.1ms 400ms
.options node nopage post

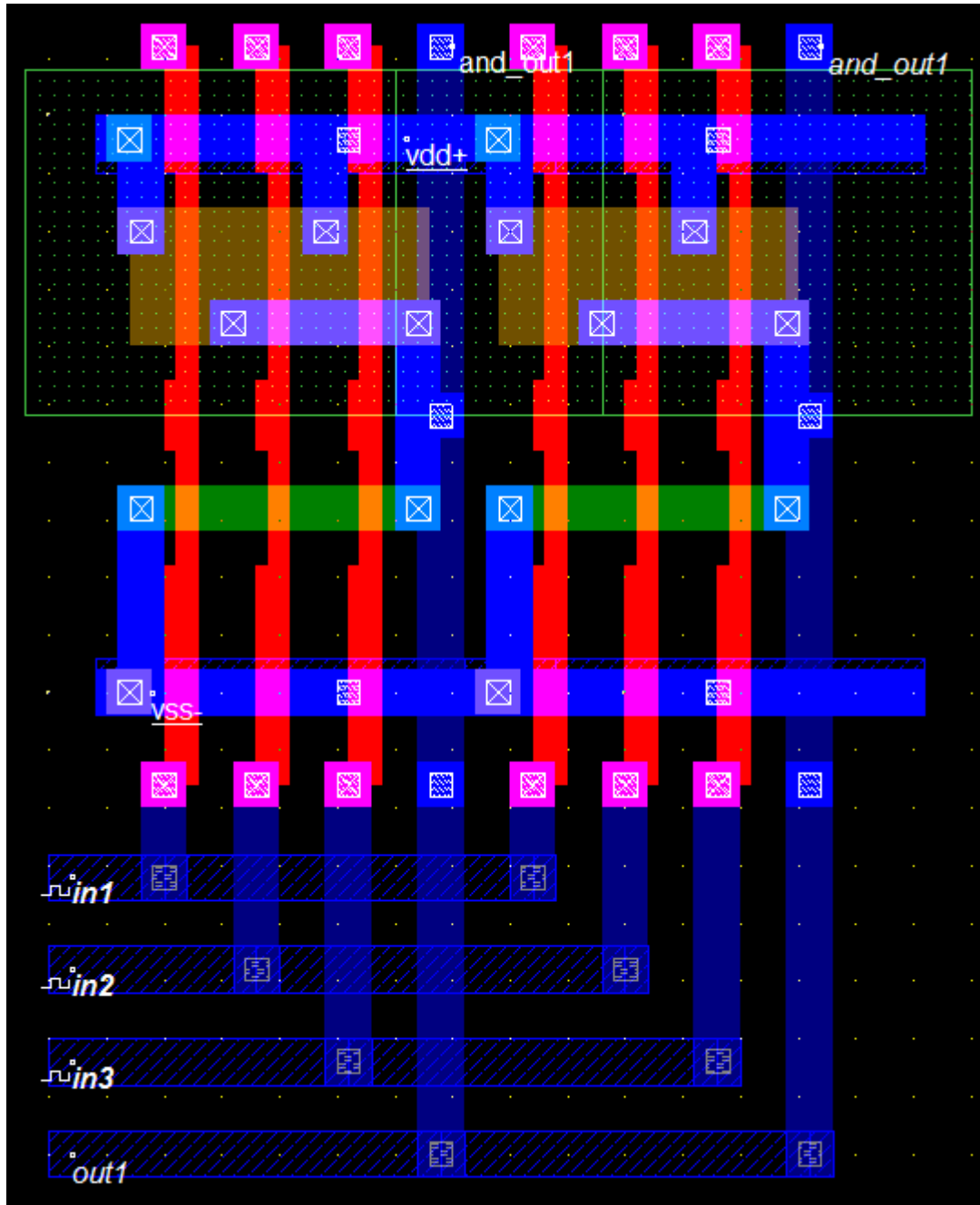
.end
```

PORTA LÓGICA NAND 3 Entradas
Resultado da simulação SPICE

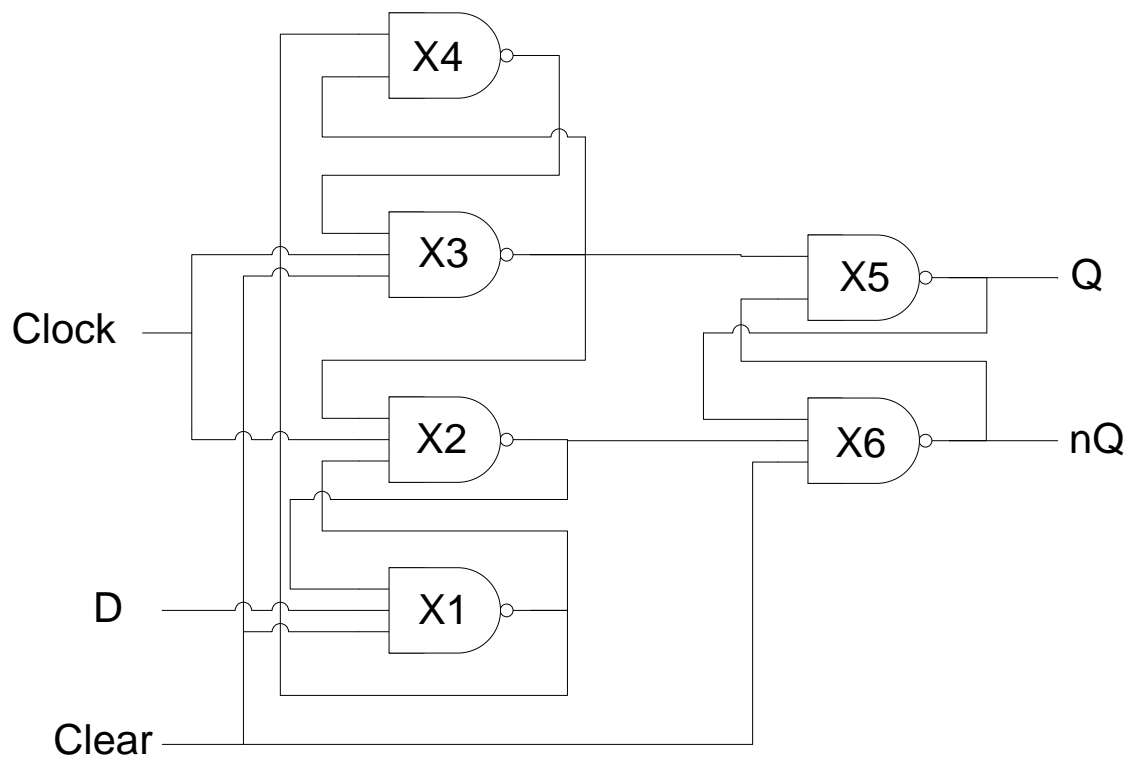
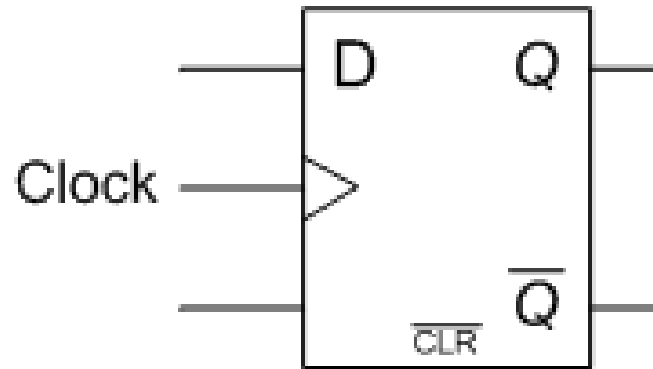


PORTA LÓGICA NAND 3 Entradas

Layout



FLIP-FLOP TIPO D COM CLEAR ASSÍNCRONO
Símbolo e diagrama lógico



FLIP-FLOP tipo D com Clear Assíncrono

Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
*****
* FlipFlop tipo D com reset assincrono - Entradas: DATA_IN CLOCK RESET OUT1 OUT2
como subcircuit
*****

*Inicio do código

*ENTRADA
*Fonte de alimentação
Vdd 99 0 5V

*Fontes para teste

V_DataIn 1 0 PULSE(0 5 2ms 0.1ns 0.1ns 4ms 30ms)
V_CLK 2 0 PULSE(0 5 4ms 0.1ns 0.1ns 0.5ms 8ms)
V_Clear 3 0 PULSE(0 5 1ms 0.1ns 0.1ns 8ms 60ms)

***** Main Circuit *****
x1 1 2 3 4 5 99 0 FFD_RA

***** Sub - Circuits *****
***** DataIN CLK RST OUT1(Q) OUT2(nQ) vdd vss
.subckt FFD_RA 1 5 2 8 9 99 0

x1 3 1 2 4 99 0 NAND3
x2 6 5 4 3 99 0 NAND3
x3 7 5 2 6 99 0 NAND3
x4 4 6 7 99 0 NAND
x5 6 9 8 99 0 NAND
x6 8 3 2 9 99 0 NAND3
.ends FFD_RA

*****1 2 3 99 0
.subckt NAND in1 in2 out vdd vss
M1 out in1 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M2 out in2 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P

```



```

PD=32U PS=32U NRD=0.33 NRS=0.33
M3  out  in1  4  vss NTRAN  L=2.0U W=3U  AD=75P  AS=75P  PD=32U
PS=32U NRD=0.33 NRS=0.33
M4  4  in2  vss  vss NTRAN  L=2.0U W=3U  AD=75P  AS=75P  PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NAND

***** 1 2 3 4 99 0
.subckt NAND3 in1 in2 in3 out vdd vss
*   D   G   S   B
M1  out  in1  vdd  vdd PTRAN  L=2.0U W=6U  AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M2  out  in2  vdd  vdd PTRAN  L=2.0U W=6U  AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M3  out  in3  vdd  vdd PTRAN  L=2.0U W=6U  AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M4  out  in1  5  vss NTRAN  L=2.0U W=3U  AD=75P  AS=75P  PD=32U
PS=32U NRD=0.33 NRS=0.33
M5  5  in2  6  vss NTRAN  L=2.0U W=3U  AD=75P  AS=75P  PD=32U
PS=32U NRD=0.33 NRS=0.33
M6  6  in3  vss  vss NTRAN  L=2.0U W=3U  AD=75P  AS=75P  PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NAND3

.tran 0.1m 30ms
.options node nopage post

.MODEL NTRAN NMOS (
+LEVEL = 2
+UO    = 608.3
+VTO   = 825.3E-3
+NFS    = 0.105E+12
+TPG    = 1.000
+TOX    = 40.00E-9
+NSUB   = 7.755E+15
+UCRIT  = 50.00E+3
+UEXP   = 78.26E-3
+VMAX   = 49.89E+3
+RSH    = 50.15
+XJ     = 450.0E-9
+LD     = 112.1E-9
+DELTA  = 3.714
+PB     = 0.440
+JS     = 10.00E-6
+NEFF   = 3.358
+CJ     = 323.1E-6
+MJ     = 461.5E-3
+CJSW   = 929.9E-12
+MJSW   = 268.3E-3
+CGSO   = 96.77E-12

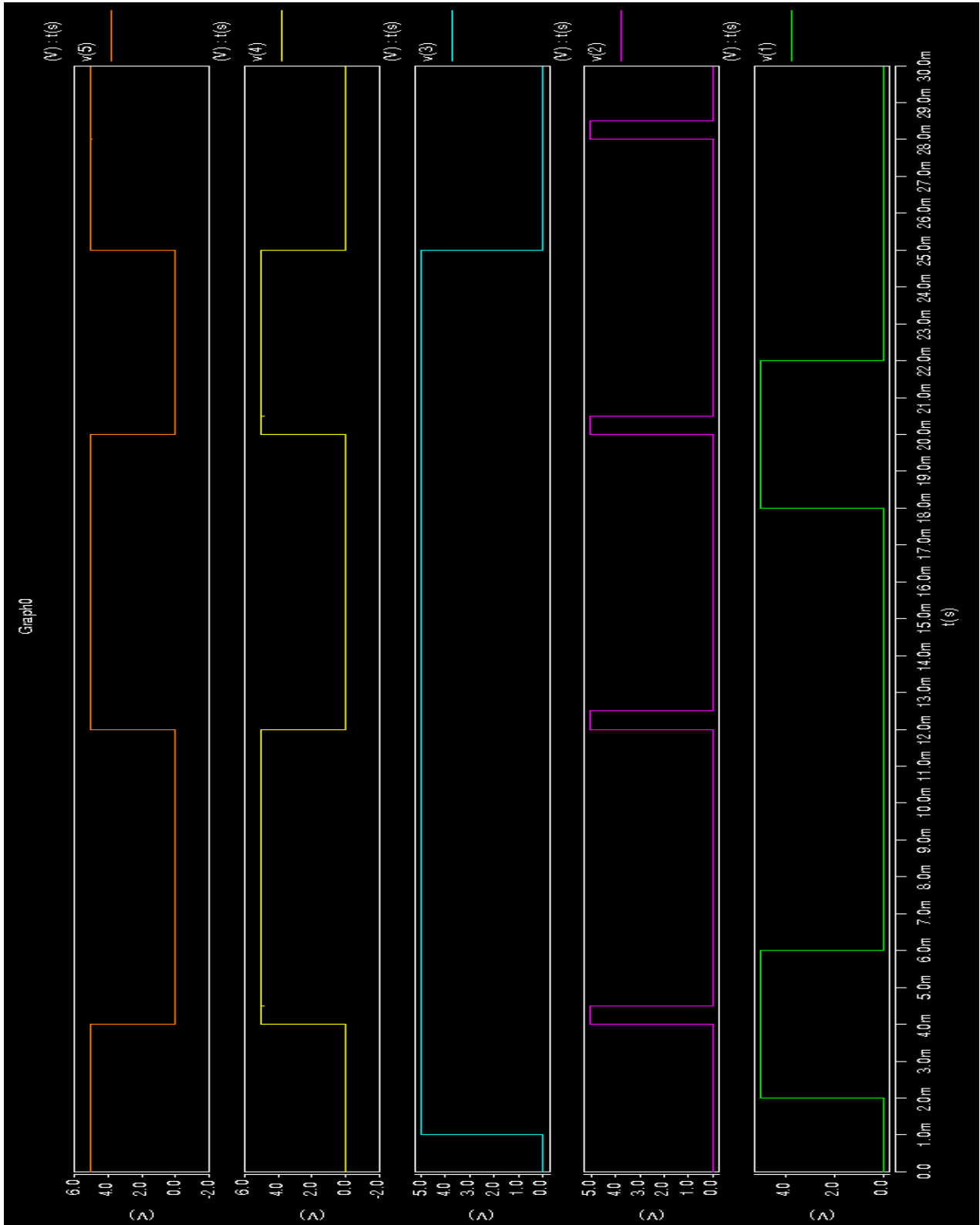
```

```
+CGDO = 96.77E-12
+CGBO = 40.00E-12
+FC   = 500.0E-3
+KF   = 5.210E-28)

*
.MODEL PTRAN PMOS(
+ LEVEL = 2
+ UO    = 205.1
+ VTO   = -703.0E-3
+ NFS   = 0.01E+12
+ TPG   = -1.00
+ TOX   = 40.00E-9
+ NSUB  = 1.486E+16
+ UCRIT = 70.00E+3
+ UEXP  = 184.2E-3
+ VMAX  = 40.76E+3
+ RSH   = 69.46
+ XJ    = 450.0E-9
+ LD    = 230.5E-9
+ DELTA = 1.843
+ PB    = 0.958
+ JS    = 10.00E-6
+ NEFF  = 0.688
+ CJ    = 804.9E-6
+ MJ    = 525.0E-3
+ CJSW  = 749.1E-12
+ MJSW  = 495.4E-3
+ CGSO  = 199.0E-12
+ CGDO  = 199.0E-12
+ CGBO  = 101.E-12
+ FC    = 500.E-3
+ KF    = 8.600E-30)

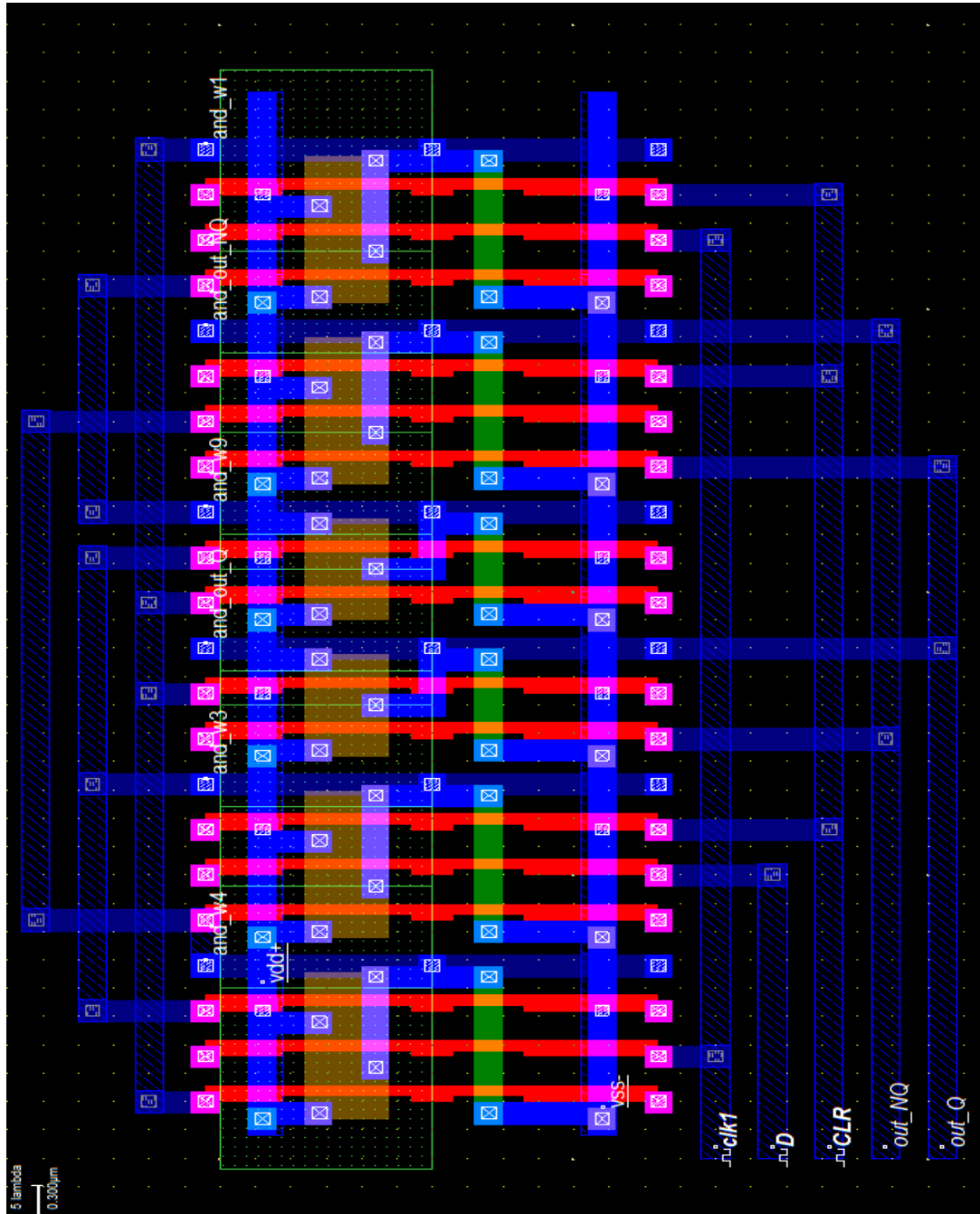
.end
```

FLIP-FLOP tipo D com Clear Assíncrono
Resultado da simulação SPICE



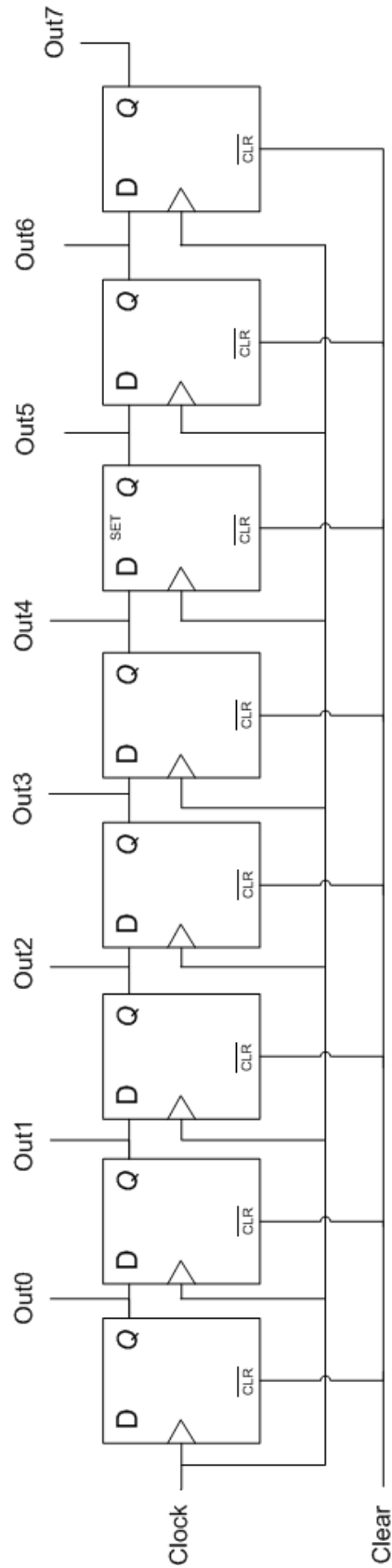
FLIP-FLOP tipo D com Clear Assíncrono

Layout



SHIFT-REGISTER 8 BITS

Diagrama lógico



SHIFT-REGISTER 8 BITS

Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
*Porta And - Entradas: 1 2 e saída 4
*****
* Shift Register 8 bits - Entradas: DATA_IN CLOCK RESET OUT1 OUT2
*****

*Inicio do código

*ENTRADA
*Fonte de alimentação
Vdd 99 0 5V

*Sinais de Entrada

V_DataIn 1 0 PULSE(0 5 10ms 0.1ms 0.1ms 20ms 250ms)
V_CLK 2 0 PULSE(0 5 1ms 0.1ns 0.1ns 5ms 20ms)
V_Clear 3 0 PULSE(0 5 1ms 0.1ms 0.1ms 300ms 600mS)

***** Main Circuit *****
* Data Clk Clr Out0 Out1 Out2 Out3 Out4 Out5 Out6 Out7 Vdd Vss Tipo
X1 1 2 3 4 5 6 7 8 9 10 11 99 0 SR8

***** Sub - Circuits *****

***** Data Clk Clr Out0 Out1 Out2 Out3 Out4 Out5 Out6 Out7 Vdd Vss
.subckt SR8 1 2 3 4 5 6 7 8 9 10 11 99 0

*Entradas - Data Clk Clr
*Saídas - Out0 Out1 Out2 Out3 Out4 Out5 Out6 Out7 Vdd Vss
x1 1 2 3 4 12 99 0 FFD_RA
x2 4 2 3 5 13 99 0 FFD_RA
x3 5 2 3 6 14 99 0 FFD_RA
x4 6 2 3 7 15 99 0 FFD_RA
x5 7 2 3 8 16 99 0 FFD_RA
x6 8 2 3 9 17 99 0 FFD_RA
x7 9 2 3 10 18 99 0 FFD_RA
x8 10 2 3 11 19 99 0 FFD_RA
.ends SR8

***** DataIN CLK RST OUT1(Q) OUT2(nQ) vdd vss

```

```

.subckt FFD_RA      1 5 2 8 9 99 0
x1 3 1 2 4 99 0 NAND3
x2 6 5 4 3 99 0 NAND3
x3 7 5 2 6 99 0 NAND3
x4 4 6 7 99 0 NAND
x5 6 9 8 99 0 NAND
x6 8 3 2 9 99 0 NAND3
.ends FFD_RA

***** 1 4 9 7 8 99 0
.subckt FFD data clk clear out1 out2 vdd vss
x1 3 5 6 vdd vss NAND
x2 6 clk 5 vdd vss NAND
x3 5 clk 3 2 vdd vss NAND3
x4 2 clear data 3 vdd vss NAND3
x5 5 out2 out1 vdd vss NAND
x6 out1 2 out2 vdd vss NAND
.ends FFD

***** 1 2 3 99 0
.subckt NAND in1 in2 out vdd vss

M1 out in1 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M2 out in2 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M3 out in1 4 vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
M4 4 in2 vss vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NAND

***** 1 2 3 4 99 0
.subckt NAND3 in1 in2 in3 out vdd vss
* D G S B
M1 out in1 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M2 out in2 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M3 out in3 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M4 out in1 5 vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
M5 5 in2 6 vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
M6 6 in3 vss vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NAND3

```

```
.MODEL NTRAN NMOS (
```

```
+LEVEL = 2
```

```
+UO = 608.3
```

```
+VTO = 825.3E-3
```

```
+NFS = 0.105E+12
```

```
+TPG = 1.000
```

```
+TOX = 40.00E-9
```

```
+NSUB = 7.755E+15
```

```
+UCRIT = 50.00E+3
```

```
+UEXP = 78.26E-3
```

```
+VMAX = 49.89E+3
```

```
+RSH = 50.15
```

```
+XJ = 450.0E-9
```

```
+LD = 112.1E-9
```

```
+DELTA = 3.714
```

```
+PB = 0.440
```

```
+JS = 10.00E-6
```

```
+NEFF = 3.358
```

```
+CJ = 323.1E-6
```

```
+MJ = 461.5E-3
```

```
+CJSW = 929.9E-12
```

```
+MJSW = 268.3E-3
```

```
+CGSO = 96.77E-12
```

```
+CGDO = 96.77E-12
```

```
+CGBO = 40.00E-12
```

```
+FC = 500.0E-3
```

```
+KF = 5.210E-28)
```

```
*
```

```
.MODEL PTRAN PMOS(
```

```
+LEVEL = 2
```

```
+UO = 205.1
```

```
+VTO = -703.0E-3
```

```
+NFS = 0.01E+12
```

```
+TPG = -1.00
```

```
+TOX = 40.00E-9
```

```
+NSUB = 1.486E+16
```

```
+UCRIT = 70.00E+3
```

```
+UEXP = 184.2E-3
```

```
+VMAX = 40.76E+3
```

```
+RSH = 69.46
```

```
+XJ = 450.0E-9
```

```
+LD = 230.5E-9
```

```
+DELTA = 1.843
```

```
+PB = 0.958
```

```
+JS = 10.00E-6
```

```
+NEFF = 0.688
```

```
+CJ = 804.9E-6
```

```
+MJ = 525.0E-3
```



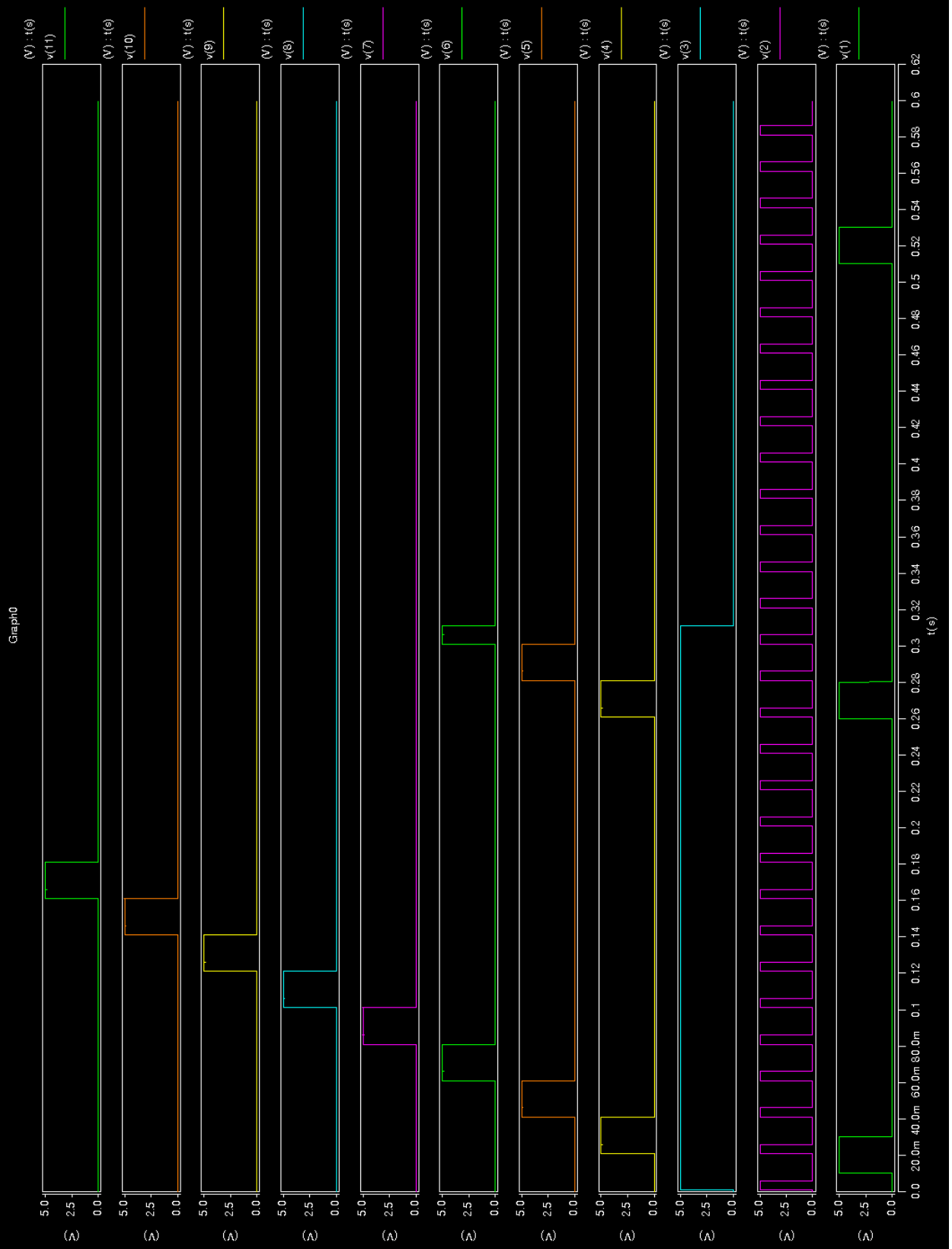
```
+ CJSW = 749.1E-12
+ MJSW = 495.4E-3
+ CGSO = 199.0E-12
+ CGDO = 199.0E-12
+ CGBO = 101.E-12
+ FC   = 500.E-3
+ KF   = 8.600E-30)

.tran 1ms 600m
.options node nopage post

.end
```

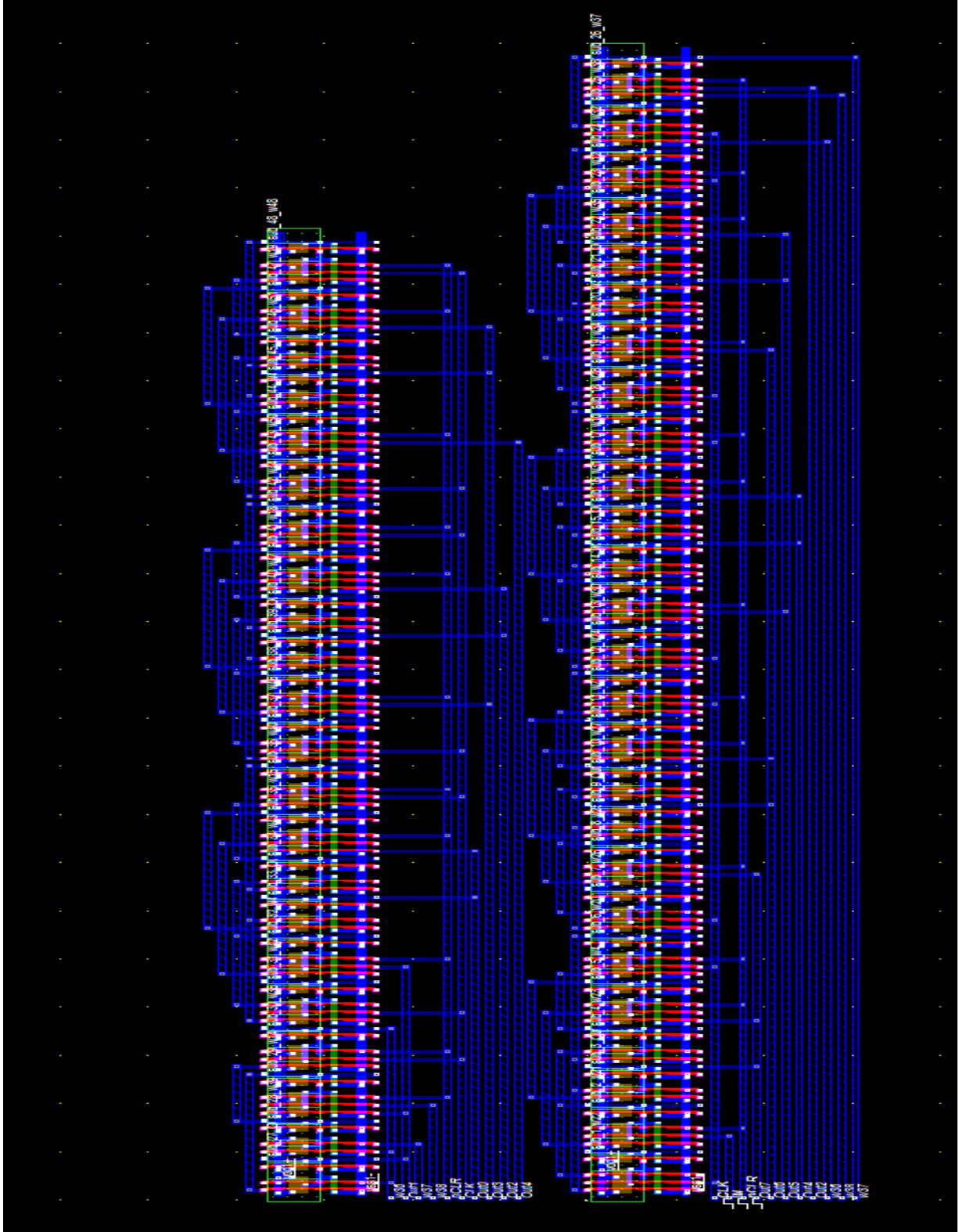
SHIFT-REGISTER 8 BITS

Resultado da simulação SPICE



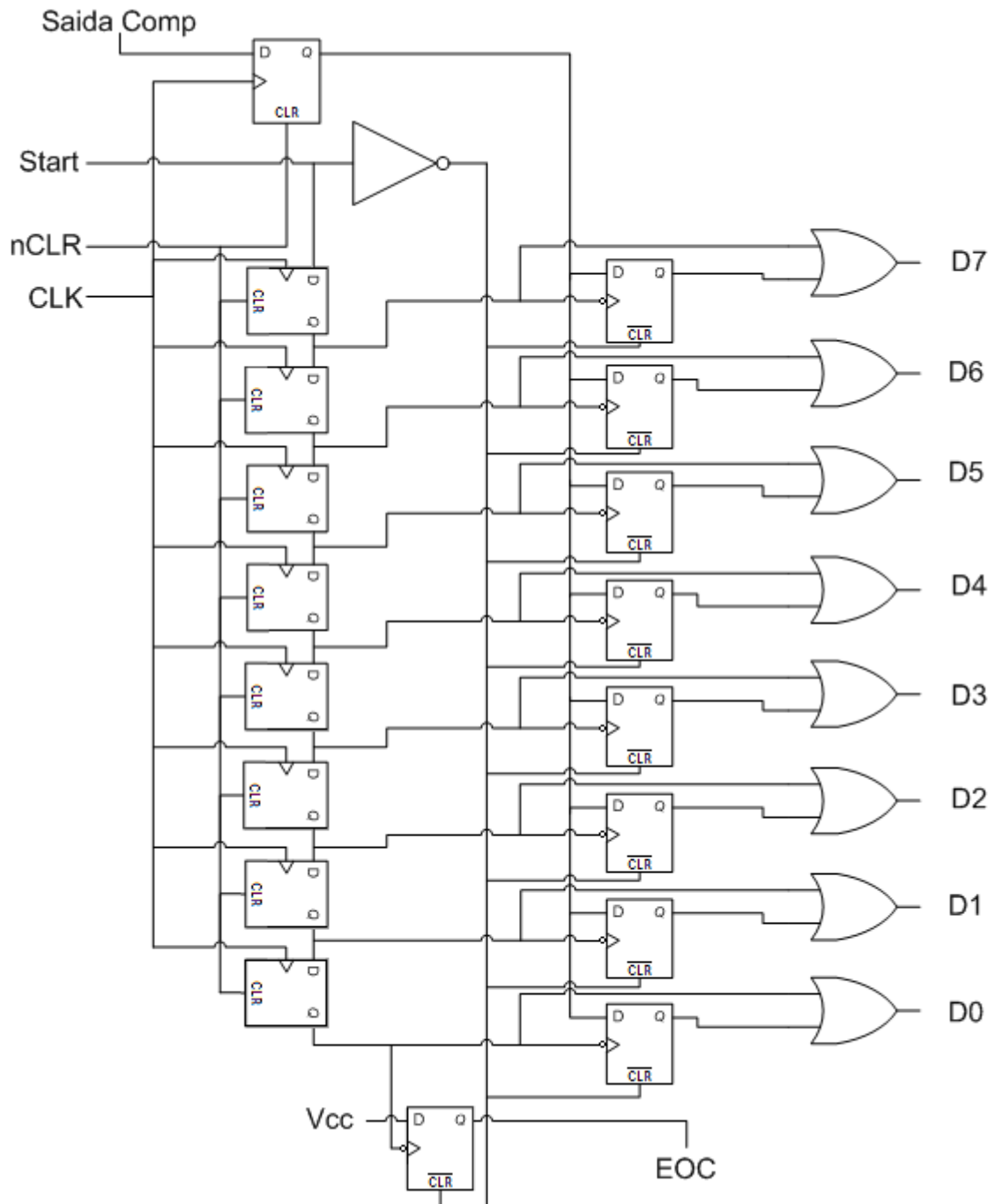
SHIFT-REGISTER 8 BITS

Layout



LÓGICA DE CONTROLE SAR

Diagrama lógico



LÓGICA DE CONTROLE SAR

Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
* Lógica de controle do conversor analógico digital SAR
*****
*Inicio do código

*ENTRADAS

*Fonte de alimentação
Vdd 99 0 5V

*Fontes para teste
V_Start 1 0 PULSE(0 5 10ms 1ns 1ns 30ms 250ms)
V_Comp 4 0 PULSE(0 5 10ms 1ns 1ns 125ms 250ms)
V_CLK 2 0 PULSE(0 5 1ms 1ns 1ns 5ms 20ms)
V_Clear 3 0 PULSE(0 5 1ms 1ns 1ns 300ms 600ms)

***** Main Circuit *****
*FFD data clk clear out1 out2 vdd vss tipo
x1 4 2 3 5 41 99 0 FFD

*NOT IN OUT Vdd Vss Tipo
x2 1 6 99 0 NOT

*SR Data Clk Clr Out0 Out1 Out2 Out3 Out4 Out5 Out6 Out7 Vdd Vss Tipo
x3 1 2 3 7 8 9 10 11 12 13 14 99 0 SR8

*nFFD data clk clear out1 out2 vdd vss tipo
x4 5 7 6 15 32 99 0 nFFD
x5 5 8 6 16 33 99 0 nFFD
x6 5 9 6 17 34 99 0 nFFD
x7 5 10 6 18 35 99 0 nFFD
x8 5 11 6 19 36 99 0 nFFD
x9 5 12 6 20 37 99 0 nFFD
x10 5 13 6 21 38 99 0 nFFD
x11 5 14 6 22 39 99 0 nFFD

*OR In1 In2 Out Vdd Vss Tipo
x12 7 15 23 99 0 OR
x13 8 16 24 99 0 OR
x14 9 17 25 99 0 OR

```

```

x15 10 18 26 99 0 OR
x16 11 19 27 99 0 OR
x17 12 20 28 99 0 OR
x18 13 21 29 99 0 OR
x19 14 22 30 99 0 OR

*nFFD data clk clear out1 out2 vdd vss tipo
x20 99 14 6 31 40 99 00 nFFD

***** Sub - Circuits *****
.subckt OR in1 in2 out Vdd Vss
x1 in1 in2 3 Vdd Vss NOR
x2 3 out Vdd Vss NOT
.ends OR

.subckt NOR in1 in2 out Vdd Vss
*   D   G   S   B
M1   3   in1   Vdd   Vdd PTRAN   L=2.0U W=6U   AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M2   out  in2   3   Vdd PTRAN   L=2.0U W=6U   AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M3   out  in1   Vss   Vss NTRAN   L=2.0U W=3U   AD=75P   AS=75P
PD=32U PS=32U NRD=0.33 NRS=0.33
M4   out  in2   Vss   Vss NTRAN   L=2.0U W=3U   AD=75P   AS=75P
PD=32U PS=32U NRD=0.33 NRS=0.33
.ends NOR

.subckt NOT in out vdd vss
M1   out  in   vdd   vdd PTRAN   L=2.0U W=6U   AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M2   out  in   vss   vss NTRAN   L=2.0U W=3U   AD=75P   AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NOT

***** Data Clk Clr Out0 Out1 Out2 Out3 Out4 Out5 Out6 Out7 Vdd Vss
.subckt SR8 1 2 3 4 5 6 7 8 9 10 11 99 0

*Entradas - Data Clk Clr
*Saídas - Out0 Out1 Out2 Out3 Out4 Out5 Out6 Out7 Vdd Vss

x1 1 2 3 4 12 99 0 FFD_RA
x2 4 2 3 5 13 99 0 FFD_RA
x3 5 2 3 6 14 99 0 FFD_RA
x4 6 2 3 7 15 99 0 FFD_RA
x5 7 2 3 8 16 99 0 FFD_RA
x6 8 2 3 9 17 99 0 FFD_RA
x7 9 2 3 10 18 99 0 FFD_RA
x8 10 2 3 11 19 99 0 FFD_RA
.ends SR8

```

```

***** DataIN CLK RST OUT1(Q) OUT2(nQ) vdd vss
.subckt FFD_RA      1  5  2  8  9  99  0
x1  3  1  2  4  99  0 NAND3
x2  6  5  4  3  99  0 NAND3
x3  7  5  2  6  99  0 NAND3

x4  4  6  7  99  0 NAND

x5  6  9  8  99  0 NAND
x6  8  3  2  9  99  0 NAND3
.ends FFD_RA

*****      1  4  9  7  8  99  0
.subckt nFFD data  clk  clear out1  out2  vdd  vss
x1 data  4 clear out1  out2  vdd  vss FFD_RA
x7 clk  4  vdd  vss NOT
.ends nFFD

*****      1  4  9  7  8  99  0
.subckt FFD  data  clk  clear out1  out2  vdd  vss
x1  3  5  6  vdd  vss NAND
x2  6  clk  5  vdd  vss NAND
x3  5  clk  3  2  vdd vss NAND3
x4  2  clear data  3  vdd vss NAND3
x5  5  out2 out1  vdd  vss NAND
x6 out1  2  out2  vdd  vss NAND
.ends FFD

*****      1  2  3  99  0
.subckt NAND in1 in2 out vdd vss

M1  out  in1  vdd  vdd PTRAN      L=2.0U W=6U  AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M2  out  in2  vdd  vdd PTRAN      L=2.0U W=6U  AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M3  out  in1  4  vss NTRAN      L=2.0U W=3U  AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
M4  4  in2  vss  vss NTRAN      L=2.0U W=3U  AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NAND

*****      1  2  3  4  99  0
.subckt NAND3 in1 in2 in3 out vdd vss
*   D   G   S   B
M1  out  in1  vdd  vdd PTRAN      L=2.0U W=6U  AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33
M2  out  in2  vdd  vdd PTRAN      L=2.0U W=6U  AD=150P AS=150P
PD=32U PS=32U NRD=0.33 NRS=0.33

```

```

M3 out in3 vdd vdd PTRAN L=2.0U W=6U AD=150P AS=150P PD=32U
PS=32U NRD=0.33 NRS=0.33
M4 out in1 5 vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
M5 5 in2 6 vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
M6 6 in3 vss vss NTRAN L=2.0U W=3U AD=75P AS=75P PD=32U
PS=32U NRD=0.33 NRS=0.33
.ends NAND3

```

```

.MODEL NTRAN NMOS (

```

```

+LEVEL = 2
+UO = 608.3
+VTO = 825.3E-3
+NFS = 0.105E+12
+TPG = 1.000
+TOX = 40.00E-9
+NSUB = 7.755E+15
+UCRIT = 50.00E+3
+UEXP = 78.26E-3
+VMAX = 49.89E+3
+RSH = 50.15
+XJ = 450.0E-9
+LD = 112.1E-9
+DELTA = 3.714
+PB = 0.440
+JS = 10.00E-6
+NEFF = 3.358
+CJ = 323.1E-6
+MJ = 461.5E-3
+CJSW = 929.9E-12
+MJSW = 268.3E-3
+CGSO = 96.77E-12
+CGDO = 96.77E-12
+CGBO = 40.00E-12
+FC = 500.0E-3
+KF = 5.210E-28)

```

```

*
```

```

.MODEL PTRAN PMOS(

```

```

+LEVEL = 2
+UO = 205.1
+VTO = -703.0E-3
+NFS = 0.01E+12
+TPG = -1.00
+TOX = 40.00E-9
+NSUB = 1.486E+16
+UCRIT = 70.00E+3
+UEXP = 184.2E-3
+VMAX = 40.76E+3

```

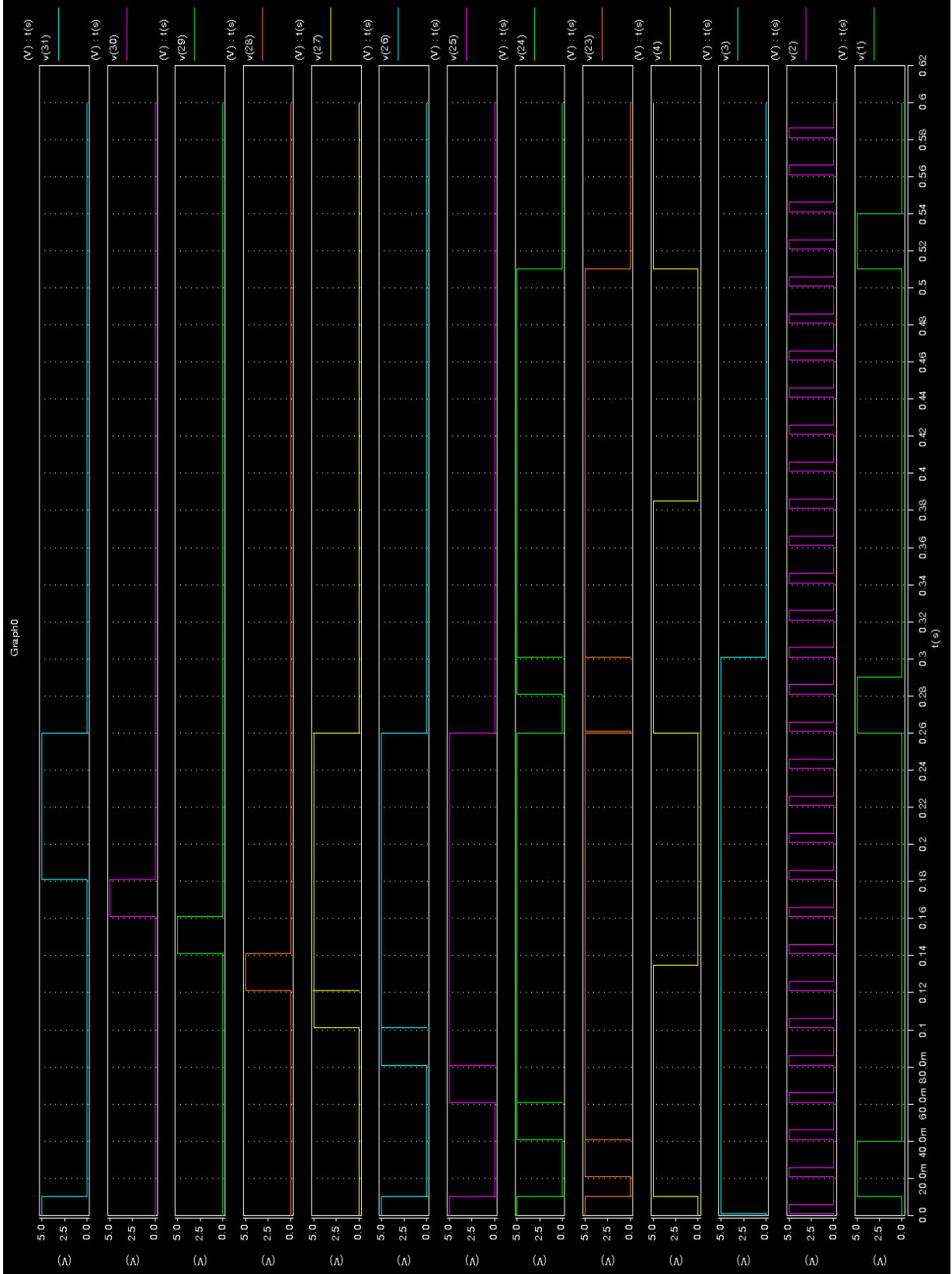


```
+ RSH = 69.46
+ XJ = 450.0E-9
+ LD = 230.5E-9
+ DELTA = 1.843
+ PB = 0.958
+ JS = 10.00E-6
+ NEFF = 0.688
+ CJ = 804.9E-6
+ MJ = 525.0E-3
+ CJSW = 749.1E-12
+ MJSW = 495.4E-3
+ CGSO = 199.0E-12
+ CGDO = 199.0E-12
+ CGBO = 101.E-12
+ FC = 500.E-3
+ KF = 8.600E-30)

.tran 1ms 600ms
.options node nopage post
.end
```

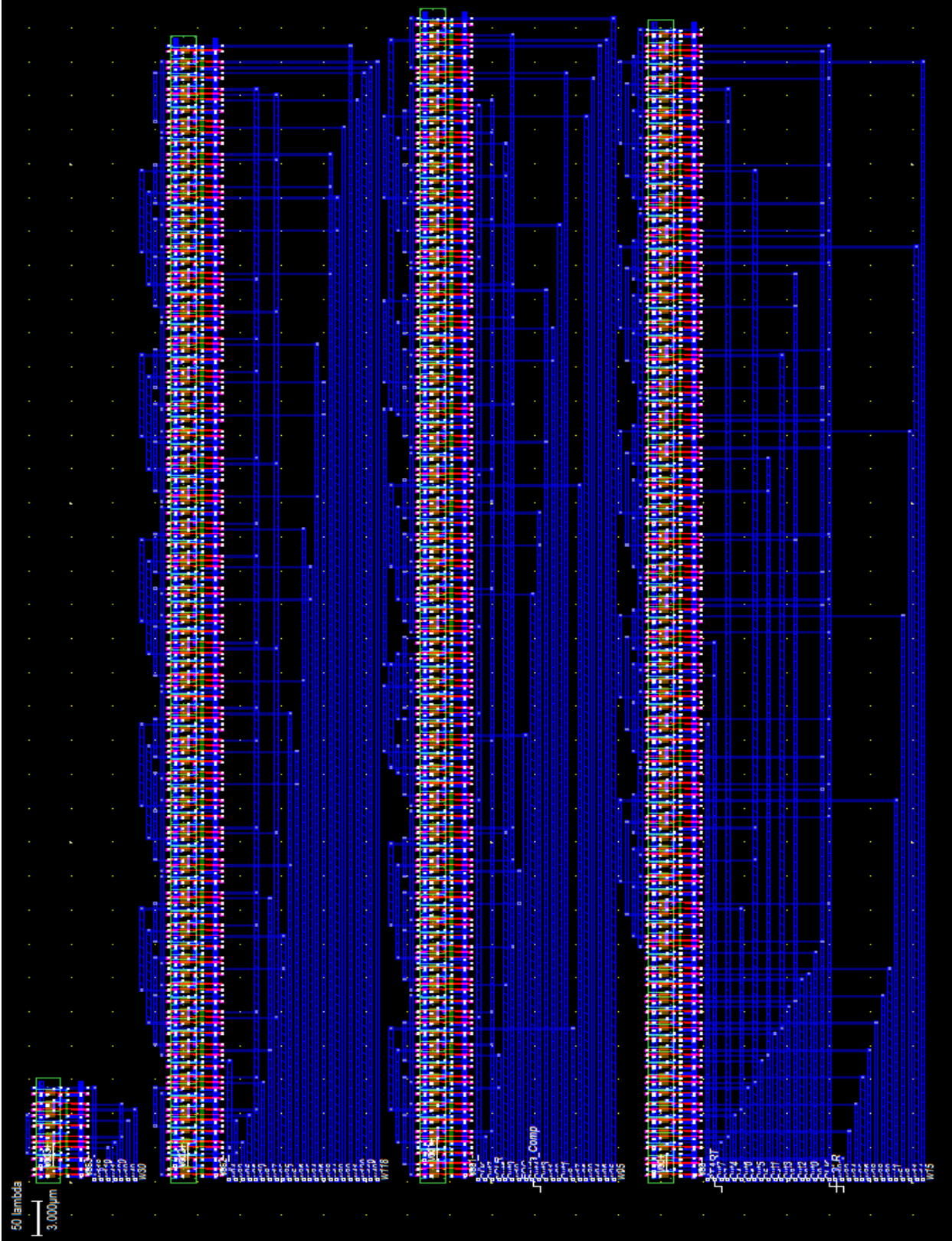
LÓGICA DE CONTROLE SAR

Resultado da simulação SPICE



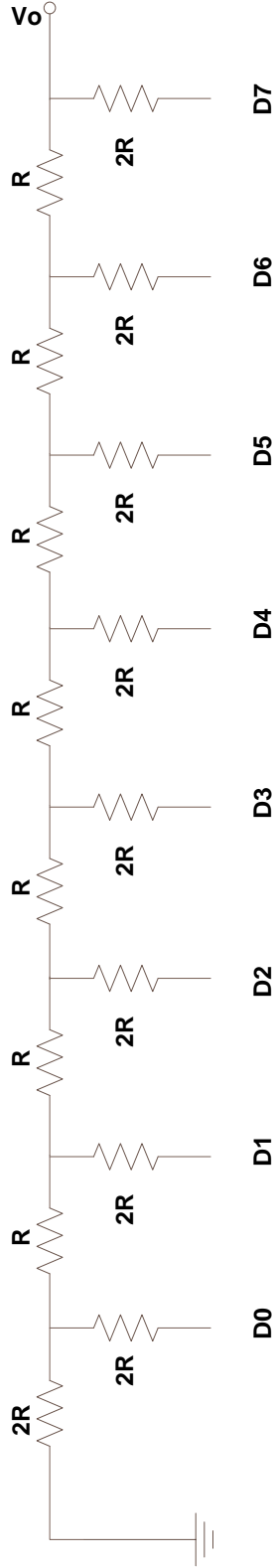
LÓGICA DE CONTROLE SAR

Layout



CONVERSOR DIGITAL ANALÓGICO R2R (DAC)

Diagrama eletrônico



CONVERSOR DIGITAL ANALÓGICO R2R (DAC)

Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
* Conversor Digital Analógico R2R
*****
*entradas nós
*v7 - 16 menos significativo
*v6 - 15
*v5 - 14
*v4 - 13
*v3 - 12
*v2 - 11
*v1 - 10
*v0 - 9 mais significativo
*saída nó 8

VIN7 16 0 PULSE(5 0 1mS 0.1nS 0.1nS 1mS 2mS)
VIN6 15 0 PULSE(5 0 1mS 0.1nS 0.1nS 2mS 4mS)
VIN5 14 0 PULSE(5 0 1mS 0.1nS 0.1nS 4mS 8mS)
VIN4 13 0 PULSE(5 0 1mS 0.1nS 0.1nS 8mS 16mS)
VIN3 12 0 PULSE(5 0 1mS 0.1nS 0.1nS 16mS 32mS)
VIN2 11 0 PULSE(5 0 1mS 0.1nS 0.1nS 32mS 64mS)
VIN1 10 0 PULSE(5 0 1mS 0.1nS 0.1nS 64mS 128mS)
VIN0 9 0 PULSE(5 0 1mS 0.1nS 0.1nS 128mS 256mS)

R1 1 0 2000

R2 2 1 1000
R3 3 2 1000
R4 4 3 1000
R5 5 4 1000
R6 6 5 1000
R7 7 6 1000
R8 8 7 1000

R10 16 1 2000
R11 15 2 2000
R12 14 3 2000
R13 13 4 2000
R14 12 5 2000
R15 11 6 2000
R16 10 7 2000

```

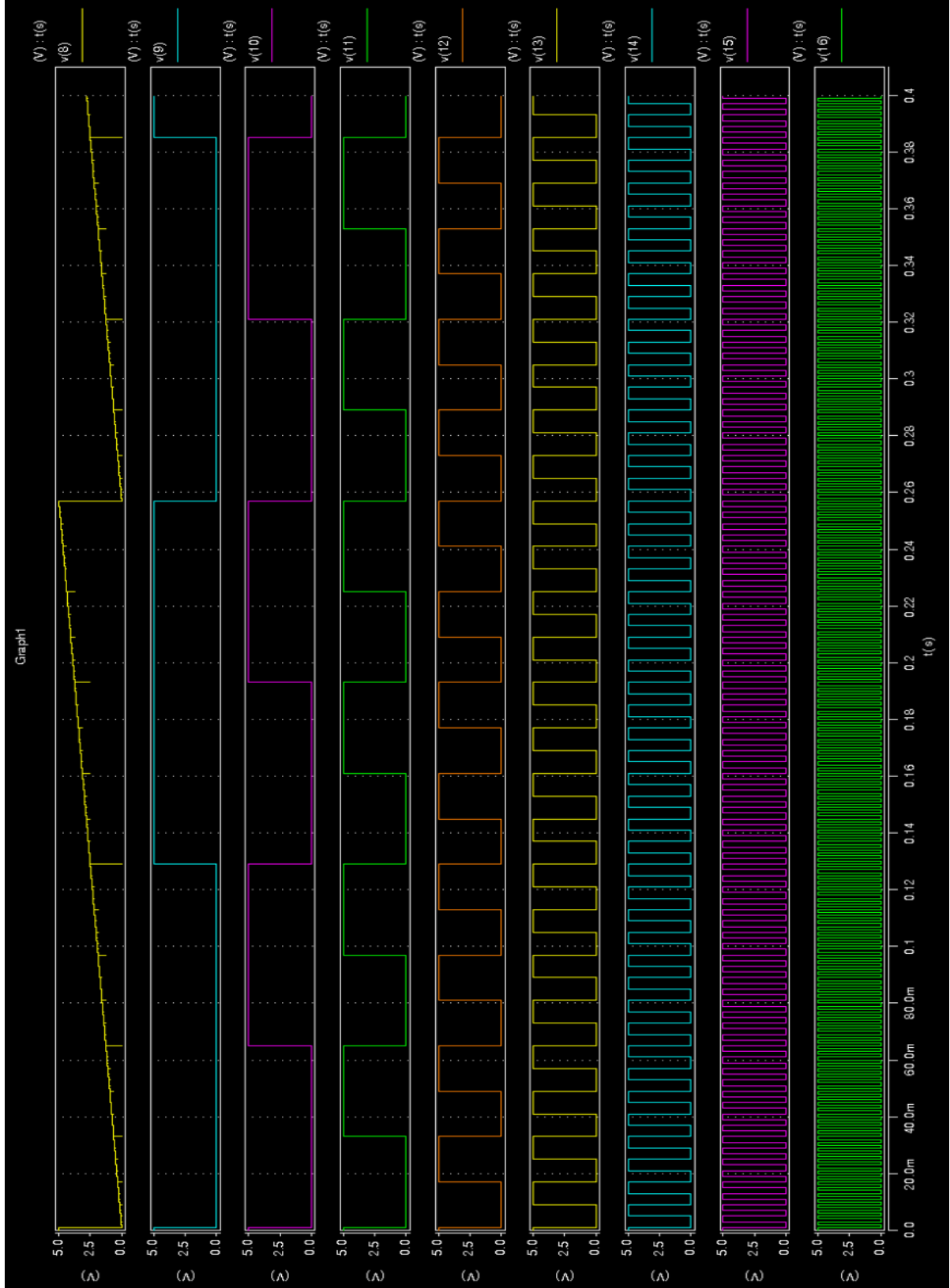
```
R17 9 8 2000
```

```
.tran 0.1ms 400ms
```

```
.options node nopage post
```

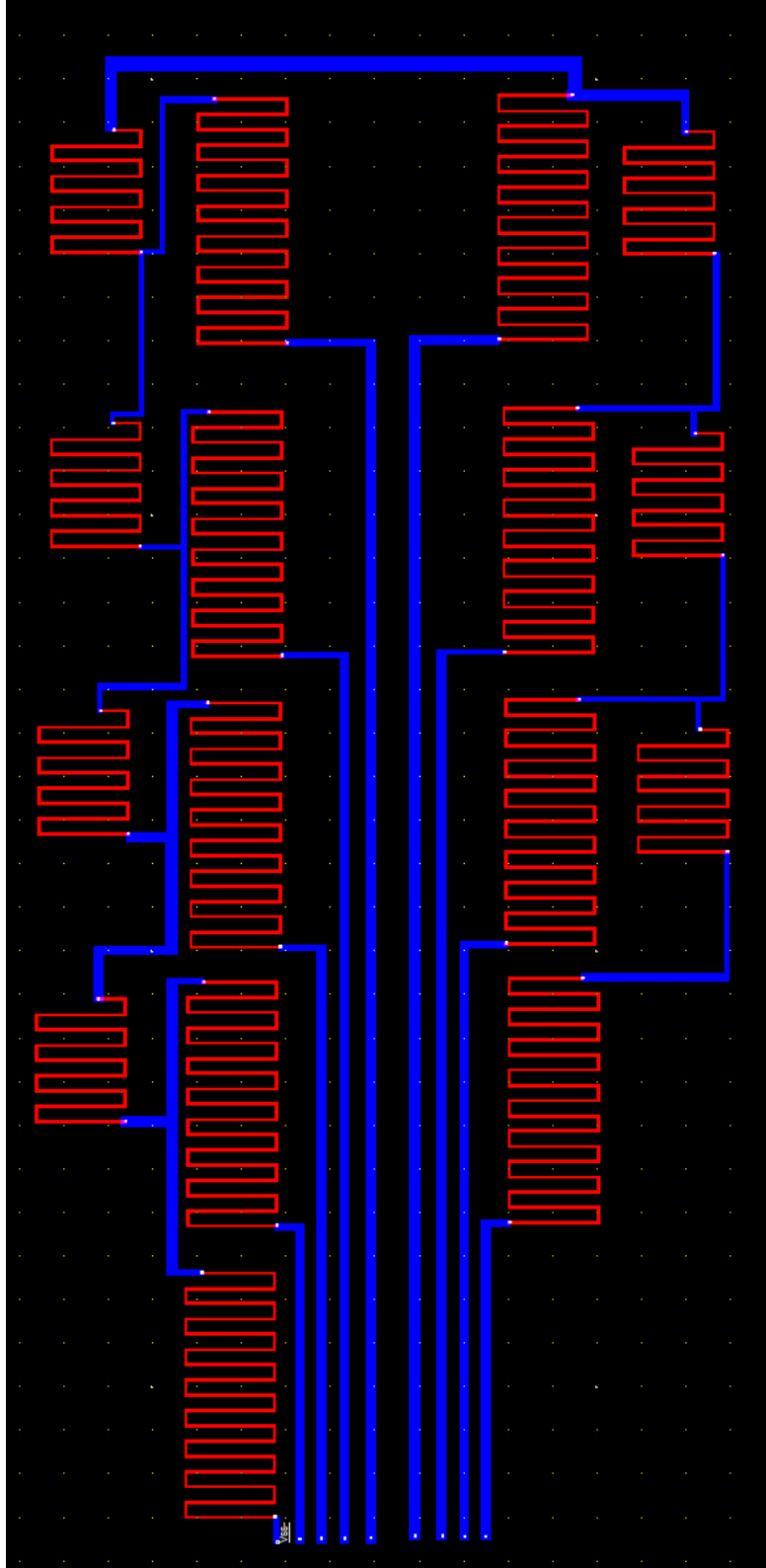
```
.end
```

CONVERSOR DIGITAL ANALÓGICO R2R (DAC) Resultado da simulação SPICE

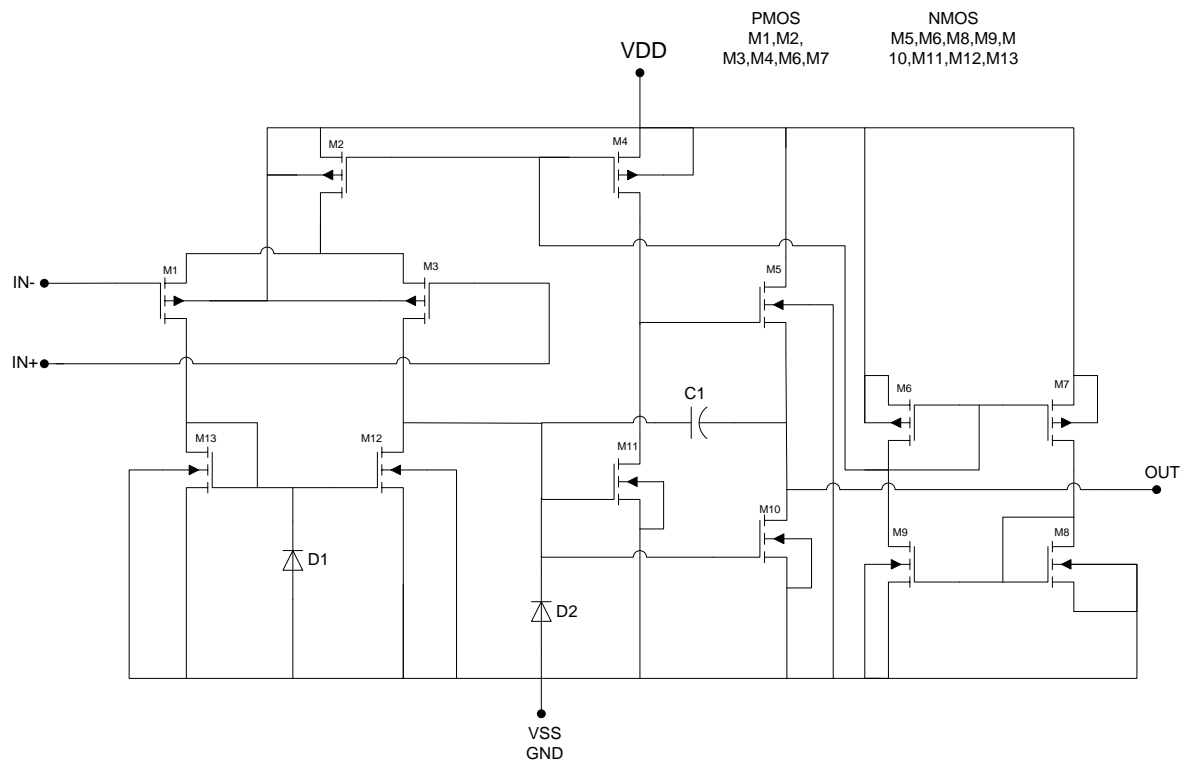
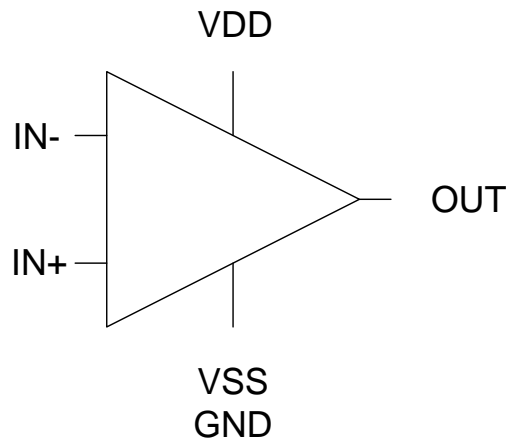


CONVERSOR DIGITAL ANALÓGICO R2R (DAC)

Layout



AMPLIFICADOR OPERACIONAL
Símbolo lógico e diagrama eletrônico



AMPLIFICADOR OPERACIONAL

Código SPICE

```

*****
* ENGENHARIA DE COMPUTAÇÃO - PUCPR
* Curitiba, 25 de novembro de 2010
*****
* Gustavo Hoeflich Damaso de Oliveira
* Valdir Antoniazzi Junior
*****
* Amp Op TLV 2324
*****

Vdd 99 0 5V

V1 1 0 sin(2.5 2.5 15 0.1m)
V2 2 0 3.23V

***CIRCUIT DESCRIPTION
* inP inN Vdd Vss Out Nome do componente
x1 1 2 99 0 5 TLV2324I

* CONNECTIONS: NON-INVERTING INPUT
*      | INVERTING INPUT
*      || POSITIVE POWER SUPPLY
*      ||| NEGATIVE POWER SUPPLY
*      |||| OUTPUT
*      |||||
.SUBCKT TLV2324I 1 2 3 4 5
*
C1 11 12 6.413E-12
C2 6 7 10.00E-12
DC 5 53 DX
DE 54 5 DX
DLP 90 91 DX
DLN 92 90 DX
DP 4 3 DX
EGND 99 0 POLY(2) (3,0) (4,0) 0 .5 .5
FB 7 99 POLY(5) VB VC VE VLP VLN 0 1.265E9 -2E9 2E9 2E9 -2E9
GA 6 0 11 12 4.964E-6
GCM 0 6 10 99 198.04E-12
ISS 3 10 DC 300.0E-9
HLIM 90 0 VLIM 1K
J1 11 2 10 JX
J2 12 1 10 JX
R2 6 9 100.0E3
RD1 60 11 201.5E3
RD2 60 12 201.5E3
RO1 8 5 50

```

```
RO2 7 99 50
RP 3 4 500.0E3
RSS 10 99 666.7E6
VAD 60 4 -.3
VB 9 0 DC 0
VC 3 53 DC 1.700
VE 54 4 DC .600
VLIM 7 8 DC 0
VLP 91 0 DC 30
VLN 0 92 DC 30
.ends TLV2324I

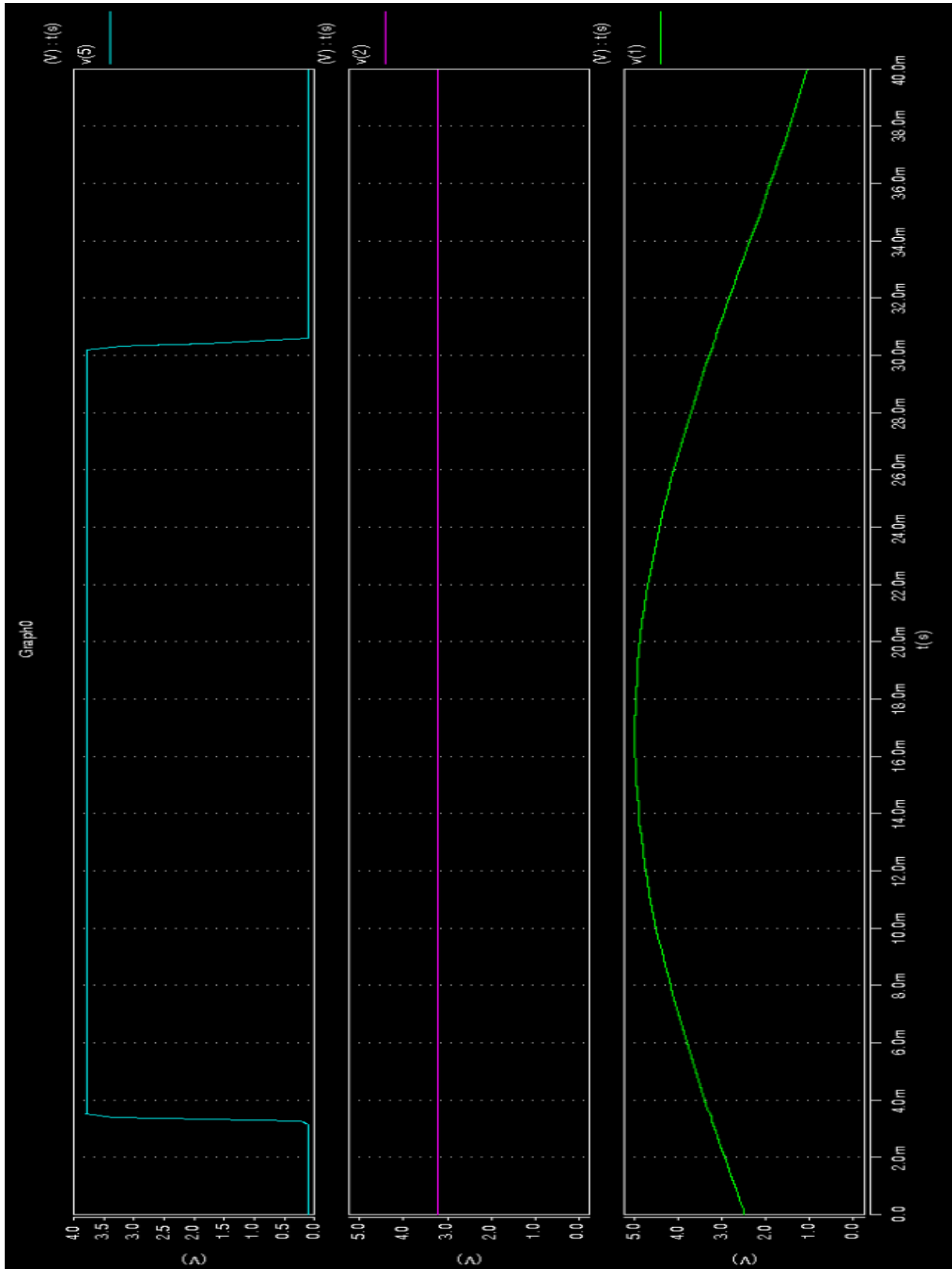
.tran 0.1m 40ms
.options node nopage post

.MODEL DX D(IS=800.0E-18)
.MODEL JX PJF(IS=300.0E-15 BETA=164.3E-6 VTO=.036)

.end
```

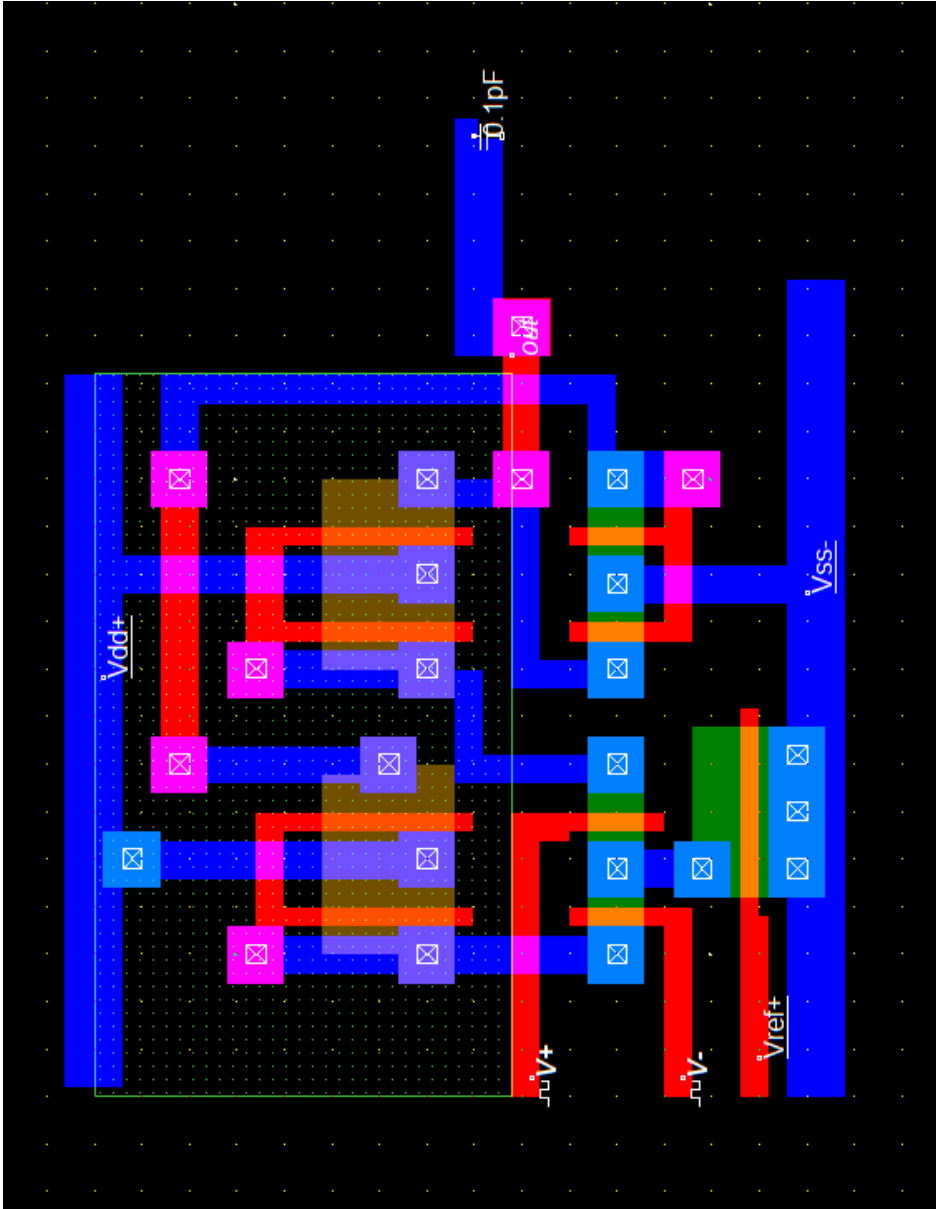
AMPLIFICADOR OPERACIONAL

Resultado da simulação SPICE



AMPLIFICADOR OPERACIONAL

Layout



CONVERSOR ANALÓGICO P/ DIGITAL
Layout do circuito integrado

