

CO-PROCESADOR DE PACOTES DE VOZ PARA INTEGRAÇÃO DE VOZ E DADOS EM
AMBIENTE DE REDES LOCAIS TIPO ETHERNET

ALEXANDRE MOREIRA DE MORAES

CO-PROCESSADOR DE PACOTES DE VOZ PARA INTEGRAÇÃO DE VOZ E DADOS
EM AMBIENTE DE REDES LOCAIS DO TIPO ETHERNET

Dissertação apresentada ao curso de Mestrado
em Engenharia Elétrica na Universidade Federal
da Paraíba, em cumprimento às exigências
para obtenção do grau de Mestre.

Área de Concentração: Rede de Computadores

William Ferreira Giozza
orientador

João Marques de Carvalho
co-orientador

Campina Grande - Pb

Junho / 1992



M827c Moraes, Alexandre Moreira de
Co-processor de pacotes de voz para integracao de voz e dados em ambiente de redes locais do tipo ethernet / Alexandre Moreira de Moraes. - Campina Grande, 1992. 103 f. : il.

Dissertacao (Mestrado em Engenharia Eletrica) - Universidade Federal da Paraiba, Centro de Ciencias e Tecnologia.

1. Redes de Computadores 2. Interface de Comunicacao 3. Integracao de Voz 4. Dissertacao I. Giozza, William Ferreira, Dr. II. Carvalho, Joao Marques, Dr. III. Universidade Federal da Paraiba - Campina Grande (PB) IV. Título

CDU 004.7(043)

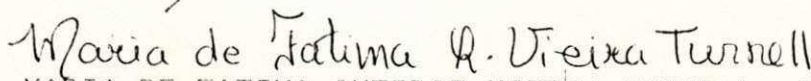
CO-PROCESSADOR DE PACOTES DE VOZ PARA INTEGRAÇÃO
DE VOZ E DADOS EM AMBIENTE DE REDES LOCAIS

ALEXANDRE MOREIRA DE MORAES

DISSERTAÇÃO APROVADA EM 02.06.1992

WILLIAM FERREIRA GIOZZA, Dr. Ing., UFPE
Orientador


JOÃO MARQUES DE CARVALHO, Ph.D., UFPE
Orientador


MARIA DE FATIMA QUEIROZ VIEIRA TURNELL, Ph.D., UFPE
Componente da Banca


ANTONIO CARLOS CAVALCANTI, Dr., UFPE
Componente da Banca

CAMPINA GRANDE - PB
JUNHO - 1992

LISTA DE ABREVIACÖES

- AC - Alternate Current
- ASCII - American Standard Code for Information
Interchange
- ASIC - Application Specific Integrated Circuit
- CIF - Caltech Interchanged Format
- CPPV - Co-Processador de Pacotes de Voz
- CRC - C6digo de Redund6ncia Ciclica
- CSMA/CD - Carrier Sense Multiple Access/Collision
Detection
- DC - Direct Current
- DMA - Direct Memory Access
- DPCM - Differential Pulse Code Modulation
- E/S - Entrada/Saída
- EPUSP - Escola Politécnica da Universidade de São Paulo
- ES2 - European Silicon Structures
- ICI - Interface de Comunicação Integrada
- IEEE - International Electrical Electronic Engineering
- LPC - Linear Predictive Coding
- LSI - Laboratorio de Sub-sistemas Integráveis
- PCM - Pulse Code Modulation
- PMU - Projeto Multi-Usuario
- RDSI - Rede Digital de Serviços Integrados
- RIO - Read Input/Output
- RX - "Buffer" Intermediário de Recepção
- TX - "Buffer" Intermediário de Transmissão
- WIO - Write Input/Output

S U M A R I O

CAPITULO I - Introdução.....	1
CAPITULO II - Integração de voz/dados em redes locais	
2 Introdução.....	3
2.1 Tipos de Trafego.....	5
2.2 Protocolos.....	10
2.2.1 CSMA/CD.....	10
2.2.2 Anel Lógico com Passagem Virtual de Ficha.....	11
2.2.3 Conexão de voz.....	14
2.3 Conclusões.....	18
CAPITULO III - Interface de Comunicação Integrada (ICID)	
3 Introdução.....	17
3.1 O telefone.....	17
3.2 A interface de comunicação analógica.....	18
3.3 A placa controladora de rede.....	20
3.4 Conclusões.....	23
CAPITULO IV - Co-Processador de Pacotes de Voz (CPPV)	
4 Introdução.....	26
4.1 Arquitetura do CPPV.....	26
4.1.1 Controle do telefone.....	27

4.1.1.1	Funcionalidade.....	27
4.1.1.2	Geração dos sinais acústicos.....	28
4.1.1.3	Identificador do número discado.....	31
4.1.1.4	Gerador de interrupções.....	34
4.1.2	Gerenciador dos "buffers" intermediários.....	36
4.1.2.1	Funcionalidade.....	36
4.1.2.2	Conversor analógico-digital.....	38
4.1.2.3	Conversor digital-analógico.....	39
4.1.2.4	"Buffer" intermediário para transmissão.....	40
4.1.2.5	"Buffer" intermediário para recepção.....	41
4.1.2.6	Dimensionamento dos "buffers" intermediários.....	41
4.1.2.7	Memória compartilhada.....	43
4.1.2.8	Gerador de endereços.....	43
4.2	Implementação do CPPV.....	46
4.3	Ambiente de "software" da ICI.....	52
4.4	Conclusão.....	55

CAPITULO V

5	Conclusões.....	56
---	-----------------	----

REFERENCIAS BIBLIOGRAFICAS.....	60
---------------------------------	----

APENDICE A

Esquemas elétricos do CPPV implementados
no SOLO 1400.....83

APENDICE B

"Layout" do ASIC CPPV.....83
Pinagem do ASIC CPPV.....84

APENDICE C

Diagramas de tempo.....95

APENDICE D

SOLO 1400.....98

RESUMO

Esta dissertação apresenta a arquitetura de uma interface de comunicação para integração de voz e dados em ambiente de redes locais do tipo Ethernet (CSMA/CD). Os circuitos de implementação da interface de comunicação integrada, baseada essencialmente num ASIC co-processador de pacotes de voz, responsável pelo processamento e controle do tráfego de voz, são descritos. Os protocolos de acesso ao meio e de conexão de voz também são descritos, e a arquitetura de um "software" básico para o gerenciamento da interface é proposto.

ABSTRACT

This work presents an architecture for a voice/data integrated local area network based on Ethernet protocol (CSMA/CD). The hardware of the integrated communications interface conceived around an ASIC, responsible for processing and control of the voice traffic, is described. Medium access control and voice connection protocol are also described. A software environment to control the interface is proposed.

CAPÍTULO I - INTRODUÇÃO

A integração de voz e dados em ambiente de redes locais é um tema de pesquisa relativamente novo, pois há pouco tempo a separação entre as aplicações de voz e dados incentivava a manutenção de redes independentes para suportar estes serviços de comunicação. Tradicionalmente a comunicação de voz tem sido suportada pela rede pública de telefonia, enquanto a comunicação de dados é mantida por redes especialmente projetadas para este fim. Além disso, as características eminentemente diferentes dos sinais de voz e dados impuseram limitações tecnológicas à integração de serviços. Entretanto, tais limitações vem sendo superadas aos poucos com o avanço da tecnologia de computadores e de comunicação e trabalhos têm sido desenvolvidos indicando possíveis soluções para que os serviços de voz e dados sejam integrados [1, 2, 3, 4].

Uma interface de comunicação integrando voz e dados em redes locais do tipo Ethernet (CSMA/CD) [5] foi proposta anteriormente em [6]. A rede local em questão é configurada em barramento e existe um telefone acoplado à algumas estações ligadas à rede.

Para haver uma comunicação de voz, basta que um dos usuários que esteja utilizando uma das estações (que dispõe de telefone), retire o fone do gancho e disque o número do usuário com quem ele deseja se comunicar (o outro usuário também deve estar presente no ambiente da rede). Este tipo de aplicação é

previsto como um serviço auxiliar de telefonia para um grupo de usuários de uma rede local não sobrecarregada em termos de tráfego de dados (Figura 1.1).

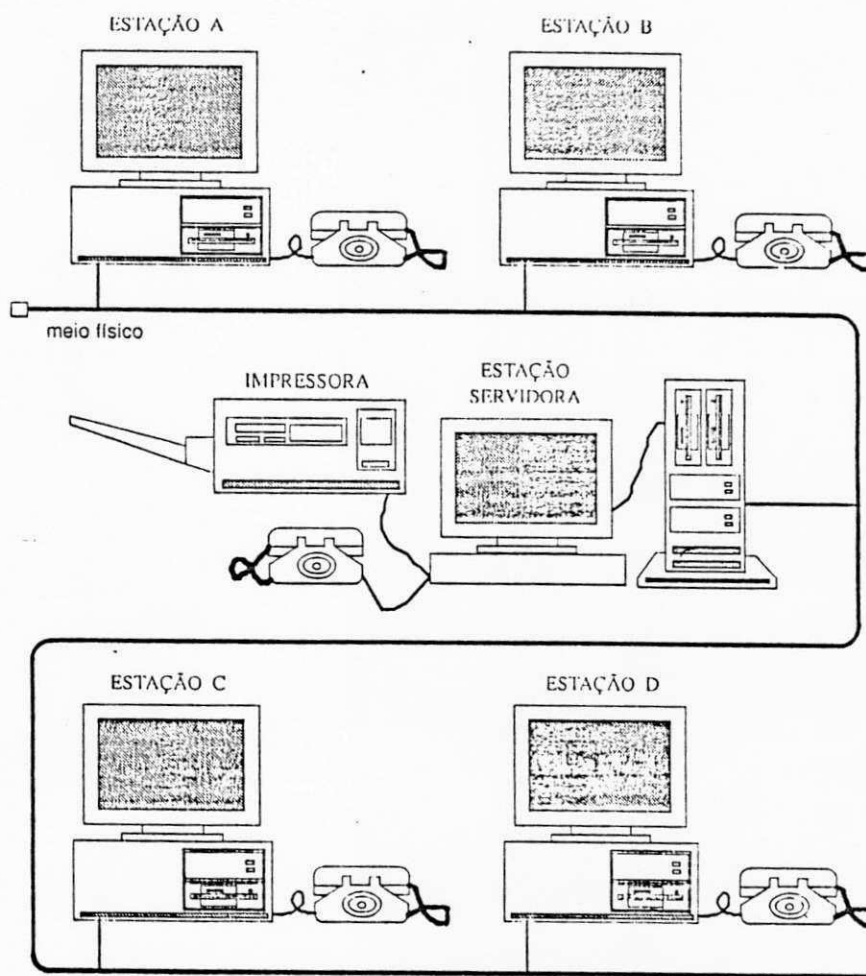


Figura 1.1. Rede local com integração de voz e dados.

Um ASIC (circuito integrado de aplicação específica), com as funções básicas de controle dos sinais do telefone, foi desenvolvido a fim de simplificar a implementação da interface de co-

municação proposta [7]. A limitação em termos de funções integradas por este ASIC pioneiro deveu-se as restrições quanto a área disponível para a sua realização prática (8 mm², via Projeto Multi-Usuario - PMU). Entretanto, atualmente, com a maior disponibilidade de área de silício (via o projeto de cooperação internacional CAPES/COFECUB e o programa CMP francês), é possível se conceber a integração de todas as funções de processamento dos protocolos de voz em uma única pastilha, simplificando bastante a implementação da interface de comunicação integrada (ICI) proposta.

Este trabalho apresenta todo o processo do desenvolvimento da arquitetura da ICI, concebida em torno de um ASIC co-processador de pacotes de voz (CPPV), responsável pelo processamento dos pacotes de voz e pelo controle do tráfego de voz na rede local.

No Capítulo II, são apresentadas as diferentes classes de tráfego existentes nas redes de comunicação e os protocolos de acesso ao meio físico para as estações envolvidas com o tráfego de dados (CSMA/CD), e para as estações envolvidas com o tráfego de voz e dados (Anel lógico com passagem virtual de ficha).

No Capítulo III, os componentes de "hardware" necessários para suportar a implementação dos protocolos apresentados no capítulo anterior, são detalhados. Estes componentes são o telefone convencional, a placa da interface de comunicação integrada e a

placa controladora de rede.

No Capítulo IV a arquitetura, a funcionalidade e o projeto e implementação de um circuito integrado de aplicação específica (ASIC) é detalhada. O ASIC foi realizado em uma ferramenta informatizada para projetos com células padrão, o SOLO 1400, da empresa europeia ES2. Este circuito integrado é o responsável pelo controle da placa da interface de comunicação integrada e é o assunto principal desta dissertação de Mestrado.

A seguir, algumas conclusões são tiradas, e sugestões são apresentadas para a continuação do trabalho.

O Apêndice A que se segue, é composto pelos esquemas elétricos utilizados no projeto e implementação do ASIC, feitos através do pacote SOLO 1400. O Apêndice B mostra o "layout" e o diagrama de pinos do circuito integrado. O Apêndice C ilustra os diagramas de tempo a serem seguidos pelos pinos do ASIC CPPV e, finalmente, o Apêndice D apresenta os conceitos básicos da ferramenta informatizada, SOLO 1400, que foi utilizada para a implementação do projeto do circuito integrado.

CAPÍTULO II - INTEGRAÇÃO DE VOZ/DADOS EM REDES CSMA/CD

2 Introdução

O interesse em estudar e propor a integração dos serviços de voz e dados em redes locais surgiu, primeiramente, da percepção que voz é um serviço de comunicação amplamente utilizado em escritórios e que as redes são componentes indispensáveis no esforço de automação das atividades deste ambiente. Assim, pode-se antever um escritório do futuro, onde uma rede local suporte uma vasta gama de serviços, tais como: transmissão de voz, "fac-simile", vídeo, transferência de arquivos, etc. Outro incentivo para esta integração é o esforço internacional que se tem observado, na tentativa de elaborar um padrão para Redes Locais de Serviços Integrados (RDSI) [8, 9], que serão redes digitais oferecendo uma grande capacidade de serviços de voz e dados baseados em canais de 64 Kbits/s. Finalmente, há a expectativa de que o desenvolvimento tecnológico acabe por consolidar as aplicações de voz e dados. Por exemplo, espera-se que técnicas de inteligência artificial possam, eventualmente, permitir que os computadores aceitem e interpretem comandos de voz em conversações pessoa-máquina ou que mensagens contendo voz, vídeo e texto, possam ser editadas, armazenadas e transmitidas, como é feito hoje com o correio eletrônico [10].

2.1 Tipos de Tráfego

As informações de voz e dados podem ser imaginadas como dois tipos distintos de tráfegos na rede de comunicação. Cada um

destes tipos possui características particulares que implicam em uma série de requisitos, que devem ser atendidos da melhor forma possível, para que a integração possa surgir como uma solução verdadeiramente eficaz à necessidade de comunicação de voz e dados. O projeto, em separado, de redes que suportem individualmente cada um destes tráfegos é um problema relativamente simples, haja visto o grande número de redes espalhadas pelo mundo que suportam voz ou dados separadamente. A dificuldade de se projetar uma rede que suporte satisfatoriamente os dois tráfegos reside exatamente no equacionamento do problema de satisfazer, de forma adequada, os requisitos individuais de cada tráfego e ainda manter aceitáveis aspectos como: eficiência, custo, qualidade dos serviços, etc.

Vamos estudar estes requisitos examinando as características dos sinais de voz e dados, localizando-os em uma das classes gerais de tráfegos existentes nas redes de comunicação. Atualmente são consideradas três classes de tráfego, embora outras classes possam surgir futuramente [11]. Em seguida, dentro de cada classe, suas características serão extraídas para que possamos perceber seus requisitos.

CLASSE I: refere-se aos sinais que necessitam de processamento em tempo real, entre eles voz e vídeo. Para todos os propósitos, vídeo pode ser considerado similar à voz, exceto pela banda de passagem requerida. A banda de passagem requerida

para o tráfego de voz, depende da qualidade desejada da voz reconstituída na saída. Códigos de baixas taxas tem sido desenvolvidos de modo a reduzir a banda de passagem requerida [12]. O tráfego de voz costuma ser tratado como um tráfego contínuo. Entretanto, foi mostrado que em uma conversação típica, este tipo de tráfego exibe uma característica "on-off" [13], isto porque a voz é um sinal gerado em tempo real pessoa-a-pessoa e que, devido a sua natureza conversacional, encontra-se ativo em um único sentido a cada instante. Além disso, mesmo durante o tempo de atividade em um sentido da comunicação, podemos perceber a existência de intervalos de silêncio entre sentenças e sílabas, o que resulta na ocupação da capacidade de transmissão em apenas 40% do tempo, aproximadamente [1]. Quanto à degradação, devido à riqueza de redundância naturalmente encontrada na voz humana, uma certa quantidade é tolerada (por exemplo: ruído, compressão, etc). Evidentemente, a quantidade exata de degradação depende do esquema de codificação usado. Esquemas como "Pulse Code Modulation" (PCMD) e o "Differential Pulse Code Modulation" (DPCMD) são relativamente protegidos contra erros e toleram algo em torno de 1% de erro [14]. Já os esquemas de codificação a baixas taxas como o "Linear Predictive Coding" (LPC) por exemplo, não são tão protegidos contra erros e, mecanismos adicionais de controle de erros são necessários. Outro requisito de grande importância para o tráfego de voz é o atraso fim-a-fim do fluxo de informação.

Este atraso está associado ao tempo de digitalização da voz, processamentos necessários na transmissão e recepção e o tempo de transmissão da mensagem da fonte ao destino. A quantidade exata de atraso fim-a-fim tolerável, é objetivo de debate entre pesquisadores. Em [15] e [16] por exemplo, podemos encontrar que atrasos de até 300 ms são tolerados. Já em [17], encontramos que o atraso máximo permitido está entre 100 e 500 ms. Parece que este intervalo é bastante aceitável, pois não foi encontrada nenhuma referência a atrasos fim-a-fim com valores fora deste intervalo. Juntamente com a preocupação e solução para este requisito, vem o requisito da garantia da continuidade da saída de voz. Geralmente, no projeto de cada rede, esquemas de garantia de continuidade para o fluxo de voz são implementados. Estes esquemas encarregam-se de disciplinar o tráfego, de modo que cada usuário participante de uma conversação receba novas informações de voz a intervalos regulares.

CLASSE II: consiste do tráfego de dados gerado em uma conversação pessoa-máquina feita entre um terminal de dados e um computador remoto. A esta classe pertencem os tráfegos conhecidos como "interactive data" (videotexto, por exemplo). A comunicação nesta classe é caracteristicamente assimétrica, constituindo-se de volumes intermitentes de informações, separados por intervalos de silêncio. Não sendo um tráfego eminentemente de tratamento em tempo real, as mensagens de dados desta classe podem tolerar

pequenos atrasos de transmissão. Para se ter uma idéia, um usuário pode esperar alguns segundos por uma resposta a um comando, porém esperas da ordem de minutos são inadmissíveis. Outra característica deste tipo de tráfego é a baixa utilização da banda de passagem que lhe é destinada. Em [18] encontramos que o fator ativo (uma medida da porção da faixa de passagem utilizada pelo tipo de tráfego) está em torno de 5%. Como um requisito fundamental deste tipo de tráfego, está a necessidade de lhe atribuir um alto grau de confiabilidade quanto a erros de transmissão, dada a sua baixa tolerabilidade a ocorrência de erros.

CLASSE III: é formada pelo tráfego de dados gerado nas conversações máquina-máquina. A esta classe pertencem os tráfegos conhecidos como "bulk data". Um exemplo que poderia ser citado para esta classe seria a transferência de arquivos entre usuários. As mensagens deste tipo de tráfego se caracterizam por serem tipicamente unidirecionais e longas, requerendo grande faixa de passagem. O fator ativo para este tipo de tráfego é maior que 50% [19]. O atraso de transmissão tolerado é substancialmente maior que o atraso tolerado nas mensagens da CLASSE II. Entretanto, os requisitos de integridade da mensagem original são mais restritivos ainda e a taxa de erro deve ser mantida muito baixa (menor que 10^{-7}) [19].

2.2 Protocolos

Foi necessário estudar alguns protocolos para compatibilizar as necessidades impostas pelo tipo de tráfego da classe I, apresentada anteriormente na seção 2.1.

2.2.1 CSMA/CD

O protocolo "Carrier Sense Multiple Access with Collision Detection" ou CSMA/CD é um meio simples e eficiente de determinar como uma estação transmite informações em um meio comum que está compartilhado com outras estações [5].

O termo "Carrier Sense" significa que alguma estação desejando transmitir "escuta" primeiro. Quando o canal está ocupado (isto é, alguma outra estação está transmitindo), a estação espera até que o canal fique limpo antes de transmitir ("escutando antes de falar").

O termo "Multiple Access" significa que alguma estação desejando transmitir pode fazê-lo. Nenhum controlador central é necessário para decidir quem está habilitado a transmitir e em que ordem.

O termo "Collision Detection" significa que quando o canal está desocupado (nenhuma outra estação está transmitindo), uma estação pode iniciar a transmissão. Entretanto, é possível que duas ou mais estações iniciem a transmissão simultaneamente, causando uma "colisão". No evento de uma colisão, as estações que transmitem continuarão transmitindo por um período fixo de tempo

para garantir que todas as estações que transmitem detectem a colisão. Isto é conhecido como reforço de colisão "jamming". Após este reforço, as estações param de transmitir e esperam um período de tempo aleatório antes de uma nova tentativa de transmissão. A quantidade de tempo aleatório de espera cresce com o número de colisões sucessivas. Deste modo as colisões podem ser resolvidas se um grande número de estações estão colidindo.

Há três vantagens significativas a favor do protocolo CSMA/CD. A primeira é que o protocolo CSMA/CD é uma tecnologia comprovada. Uma rede CSMA/CD, Ethernet, tem sido usada pela Xerox desde 1975. A rede Ethernet é tão bem entendida e aceita que o IEEE adotou-a (com pequenas mudanças) como o padrão IEEE 802.3 10Base5 (10 Mbps, banda básica, 500 metros por segmento) [5]. Confiança é a segunda vantagem do protocolo 802.3. Este método de acesso ao meio possibilita à rede trabalhar sem controle central ou chaveamento. Com isso, se uma das estações apresentar defeito, o resto da rede pode continuar operando. Finalmente, desde que redes CSMA/CD têm distribuição geográfica livre, elas permitem fácil expansão. Novos nós podem ser adicionados a qualquer tempo sem a reinicialização completa da rede.

2.2.2 Anel lógico com passagem virtual de ficha

Para garantir o fluxo contínuo do tráfego dos pacotes de octetos de voz em ambiente de rede local CSMA/CD, as estações envolvidas com este tráfego são organizadas em um anel lógico.

onde uma ficha circula virtualmente pelo anel, atribuindo a estação que possui a ficha a maior prioridade no acesso ao meio, como ilustra a Figura 2.1 [13].

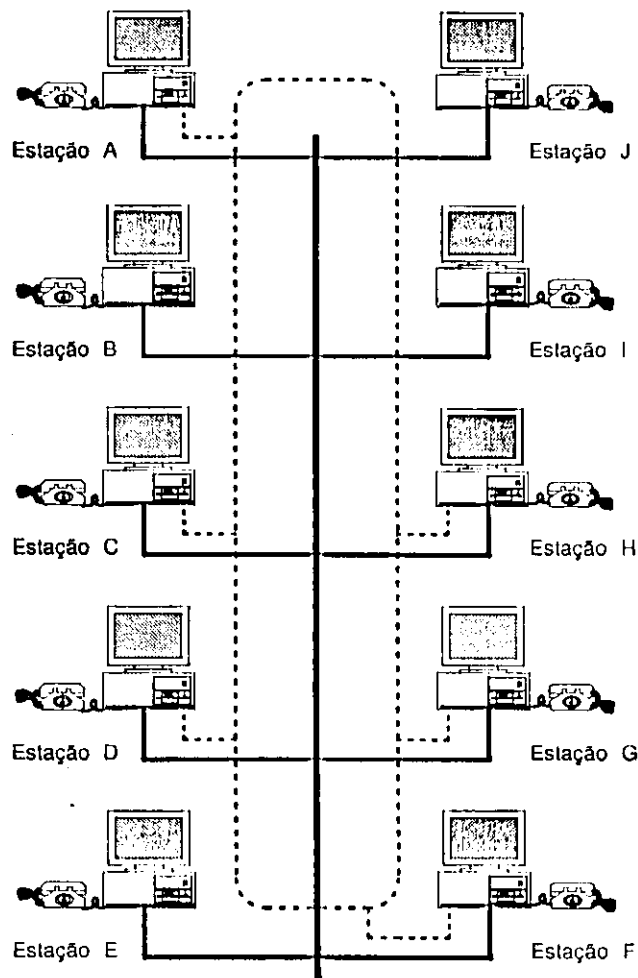


Figura 2.1. Anel lógico com passagem virtual de ficha.

Todas as outras estações da rede seguem usando o protocolo CSMA/CD para a aquisição do meio. A estação com a posse da ficha transmite com prioridade máxima, isto é, em caso de colisão com uma estação sem ficha, a estação que possui a ficha

desabilita seu tempo de "backoff" e volta imediatamente a tentar a transmissão, enquanto que a estação sem a ficha usa normalmente o tempo de "backoff".

A passagem da ficha é feita sem que seja necessária a transmissão de um pacote específico. Todas as estações escutam o meio continuamente e, sempre que o meio transitar entre os estados de ocupado para livre, a ficha é implicitamente passada. Para que isto possa acontecer, todas as estações mantêm sempre informações atualizadas sobre duas variáveis de estado:

- m, o número de estações no anel lógico;
- r, o número de vezes que o meio transitará entre os estados de ocupado para livre, antes que esta estação tenha a posse da ficha.

No fim de cada período de transmissão todas as estações decrementam o valor de r. A estação tem a posse da ficha quando r é igual a zero.

O número de estações que poderão fazer parte do anel lógico é limitado pela parcela da capacidade do meio atribuída ao tráfego de voz. Quando uma estação deseja se inserir no anel lógico (sempre que vai iniciar uma conversação de voz), um pedido de inserção deve ser emitido para todas as estações do anel lógico ("broadcast"), usando o protocolo CSMA/CD. Este pedido, quando recebido pela estação que tem a posse da ficha, é respondido por um quadro de resposta de inserção, caso haja

capacidade suficiente para suportar a carga imposta por esta nova estação (o valor de m). A resposta de inserção transportará os novos valores de r e m , e todas as outras estações que participam do anel lógico atualizarão esses valores.

Quando uma estação deseja retirar-se do anel lógico, um quadro "pedido de retirada" é transmitido, de modo a que todas as estações decrementem o valor de m que possuem e todas as estações com o valor de r superior ao da estação que se retira do anel lógico, também decrementam o valor de r que possuem.

2.2.3 Conexão de voz

Para que uma estação A possa se comunicar com uma estação B, é necessário que seja estabelecida uma conexão entre ambas. A Figura 2.2 ilustra a situação.

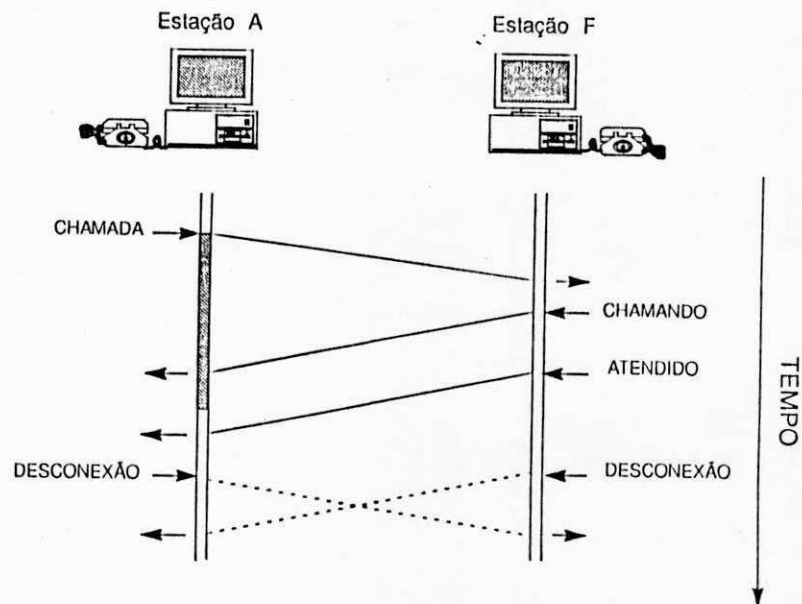


Figura 2.2. Fluxo dos quadros para conexão de voz.

Após o usuário da estação A retirar o fone do gancho, disparar o processo de inserção no anel lógico, ser aceito e ouvir o tom de discar indicando a existência de linha livre para si, ele está habilitado para discar o número que identifica a estação do usuário B. Imediatamente após a discagem, a Interface de Comunicação Integrada (ICI) da estação do usuário A gera e transmite o quadro "CHAMADA" através da rede.

Se o usuário B não estiver envolvido com uma outra conexão de voz, a ICI da estação do usuário B, ao receber o quadro "CHAMADA", dispara o seu processo de inserção no anel lógico, gera localmente o sinal de "campainha chamando", e transmite um quadro "CHAMANDO", através da rede, para o usuário A. Se após um certo tempo pré-determinado, o quadro "CHAMANDO" transmitido pela estação B não chegar ao destino, a ICI da estação do usuário A gera localmente o tom de telefone ocupado, avisando ao usuário A que a conexão não foi possível. Se, por outro lado, o quadro "CHAMANDO" chegar dentro do intervalo de tempo tolerado, a ICI da estação do usuário A gera localmente o tom de telefone chamando, indicando a este usuário que o pedido de conexão poderá ser aceito (vai depender agora da disponibilidade do usuário B).

Quando o usuário B atende o pedido de conexão com a retirada do fone do gancho, ele faz com que a sua ICI gere e envie o quadro "ATENDIDO" através da rede, desligue automati-

camente o sinal de telefone chamando, e fique pronto para o fluxo dos pacotes de octetos de voz. O quadro "ATENDIDO", ao ser recebido e reconhecido pela ICI da estação do usuário A, faz com que a sua ICI desligue o sinal local de telefone chamando e fique pronto para iniciar o fluxo dos pacotes de octetos de voz.

Se o usuário A (chamador - inicializador da comunicação) recolocar o fone no gancho, a sua ICI gera e envia um quadro "DESCONEXÃO" para o usuário B (chamado), e interrompe o tráfego dos pacotes de octetos de voz, desfazendo assim a conexão. Quando o usuário chamado recebe um quadro "DESCONEXÃO", a ICI da estação deste usuário interrompe o tráfego dos pacotes de voz e gera localmente o tom de telefone ocupado.

2.3 Conclusões

Observando a importância da integração de voz em um ambiente de rede local, e que redes do tipo Ethernet CSMA/CD são bastante populares e têm características propícias para suportar o tipo de tráfego que uma comunicação de voz necessita, foram introduzidos dois protocolos (um de acesso facilitado ao meio de transmissão para as estações envolvidas com o tráfego de voz e o outro para a conexão de voz), de modo que redes Ethernet CSMA/CD possam ser usadas para a integração de voz e dados.

CAPÍTULO III - INTERFACE DE COMUNICAÇÃO INTEGRADA (ICI)

3 Introdução

Para que os protocolos de acesso ao meio físico, das estações envolvidas com o tráfego de voz (Anel lógico com passagem virtual de ficha), e o de conexão de voz possam ser implementados, algum "hardware" mínimo deve ser implementado. Este capítulo apresenta os componentes necessários para que a integração de voz e dados em um ambiente de rede local do tipo Ethernet possa ser implementada.

3.1 O telefone

O aparelho telefônico é um dispositivo de comunicação bidirecional, que pode ser entendido como sendo constituído por um modelo DC e um modelo AC [20].

O modelo DC pode ser entendido como sendo apenas uma resistência que tem seu valor modificado pela condição em que o aparelho se encontra.

Quando o fone está em repouso no gancho, o modelo apresenta o valor da resistência de um circuito aberto.

Quando o fone é retirado do gancho, a resistência apresenta um valor de algumas dezenas de Kohms (não há um valor padrão).

Quando o disco é levado pelo dedo do usuário até o bloqueio, a resistência apresenta-se como um curto.

Quando o disco é liberado e começa o retorno para sua posição de descanso, a resistência fica variando bruscamente en-

tre os valores de curto e aberto, e assim se repete pelo mesmo número de vezes do número discado pelo usuário. Há uma padronização do tempo em que o disco do telefone abre e fecha sua chave interna. A chave permanece aberta por 33 ms e fechada por 55 ms.

O modelo AC pode ser entendido como sendo uma impedância de 600 ohms e é constituído por um conversor de dois para quatro fios (bobina híbrida), cuja função é permitir que os sinais elétricos de voz, gerados pela cápsula transmissora do telefone, possam compartilhar do mesmo par de condutores que conduzem os sinais elétricos de voz gerados pelo interlocutor, e que serão aplicados na capsula receptora do telefone.

3.2 A interface de comunicação integrada (ICI)

A interface de comunicação integrada é uma placa que, uma vez encaixada em um dos "slots" de expansão de uma estação hospedeira, permite que o usuário desta estação possa comunicar-se com o usuário de uma outra estação que também tenha uma placa de interface de comunicação integrada (Figura 3.1)

Conectada a esta placa fica um telefone do tipo convencional, utilizado nas residências. A placa da interface de comunicação integrada (ICI) funciona como uma pequena central telefônica localizada em cada estação, gerando os sinais acústicos que são aplicados no telefone para indicar ao usuário o andamento do processo de conexão de voz, gerenciando os sinais elétricos que vêm do telefone, e gerenciando o fluxo dos pacotes de octetos de

voz que são gerados pelo telefone e recebidos por ele.

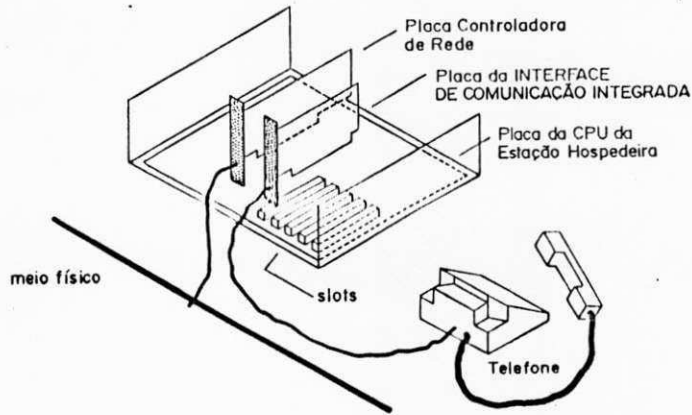


Figura 3.1. Estação com integração de voz e dados.

Desta forma, a ICI age através de uma interface analógica (Figura 3.2) e faz a adequação dos sinais elétricos analógicos gerados pelo e para o telefone.

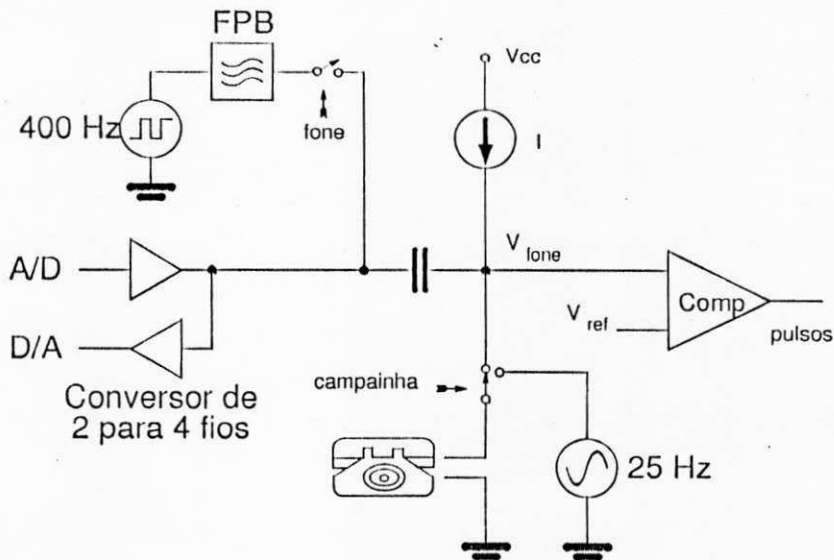


Figura 3.2. Interface analógica.

Os sinais elétricos analógicos gerados pelo telefone, são transformados em sinais cujos níveis elétricos são aceitáveis pelos circuitos que vão processá-los (transformação dos pulsos de corrente gerados pelo disco do telefone em pulsos digitais, e o tratamento do sinal de voz gerado pela cápsula transmissora do telefone, aplicado no conversor analógico-digital).

Os sinais de voz decodificada pelo conversor digital-analógico, acústicos de linha livre para discar, telefone chamando e telefone ocupado, e de disparo da campainha são sinais elétricos analógicos gerados para o telefone.

A placa da ICI é composta por circuitos analógicos e digitais que realizam as funções de uma central telefônica. Um gerador de trens de pulsos digitais de 400 Hz tem sua saída aplicada a um filtro passa-baixa para retirar algumas das componentes de alta frequência do trem de pulsos. Este sinal filtrado e devidamente amplificado e modulado em amplitude é o que dá origem aos sinais de linha livre para discar, de telefone chamando e de telefone ocupado, para poderem ser aplicados na cápsula receptora do telefone. A razão da filtragem é a de tornar o sinal acústico menos irritante para o ouvido do usuário. A modulação é feita através de uma chave analógica, disparada digitalmente e o sinal modulado acompanha a forma de onda digital do sinal que dispara a chave.

Um outro sinal digital também modula em amplitude o sinal

senoidal de baixa frequência (25 Hz) que dispara a campainha do telefone. Este sinal digital é aplicado na entrada de um "buffer" para conseguir potência suficiente para disparar um relé que permite que a tensão senoidal passe para a campainha.

Uma fonte de corrente constante força a passagem de uma corrente de 20 mA pelo circuito do telefone. Como foi visto na Seção 3.1, o circuito DC do telefone se comporta como uma resistência que varia dependendo da posição do fone (no gancho ou fora dele), e do disco (no bloqueio ou liberado de volta para a sua posição inicial). Mantendo uma corrente constante sobre esta "resistência variável", a queda de tensão sobre o telefone também vai variar. Usa-se um comparador com uma tensão de referência para gerar o trem de pulsos digital que vai permitir identificar se o fone foi retirado do gancho, qual o número discado e se o fone foi colocado novamente no gancho.

E, finalmente, um circuito conversor de dois para quatro fios é usado para permitir que o sinal de voz analógico gerado pela cápsula transmissora do telefone, possa ser aplicado na entrada do conversor analógico-digital, e que o sinal analógico de voz, reconvertido pelo conversor digital-analógico, possa ser aplicado na cápsula receptora do telefone.

O controle geral da ICI é feito por um circuito integrado de aplicação específica (ASIC). O circuito integrado é um Co-Processador de Pacotes de Voz (CPPV), cujo projeto e implementação

são o assunto principal desta dissertação de Mestrado.

A arquitetura e funcionalidade do CPPV são apresentadas com mais detalhes no próximo capítulo.

3.3 A placa controladora de rede

Os componentes básicos do "hardware" de uma rede local são as placas controladoras de rede, cabos, unidades de acesso ou repetidores. Dependendo do tipo e complexidade, a rede pode incluir também pontes, roteadores e concentradores [21].

Cada estação de trabalho conectada fisicamente à rede possui uma placa responsável pelo gerenciamento e controle do fluxo dos pacotes de dados. Estas placas contêm lógica suficiente para saber quando estão prontas para enviar um quadro e para reconhecer quando os quadros recebidos são destinados para si. Em conjunto com o programa de suporte, as placas controladoras de rede apresentam sete passos principais durante o processo de envio e recepção de um quadro. Quando um quadro vai ser enviado, os seguintes passos são apresentados na ordem mostrada; quando um quadro vai ser recebido, os passos devem ser vistos na ordem reversa:

1. Transferência do dado. Os dados são transferidos da memória do PC para a placa controladora de rede via DMA, memória compartilhada ou E/S programada.

2. Armazenamento. Enquanto estão sendo processados na placa controladora de rede, os dados são mantidos em um "buffer".

3. Formação do quadro. A placa controladora de rede tem que "quebrar" os dados (ou na recepção, juntá-los) em pedaços menores. A maioria das redes usa um tamanho entre 1 e 4 kbytes. Um cabeçalho é prefixado ao pacote de dados; uma cauda também é sufixada ao pacote de dado. Neste ponto, um quadro completo e pronto para transmitir está criado. (Na recepção, o cabeçalho e a cauda são removidos neste passo).

4. Acesso ao cabo. Em uma rede CSMA/CD, a placa controladora de rede garante que a linha está quieta antes de transmitir seus quadros. (O acesso ao cabo é irrelevante na recepção).

5. Conversões paralela e série. Os bytes de dados no "buffer" são enviados ou recebidos através dos cabos de maneira serial, com um bit seguindo o próximo. A placa controladora de rede faz esta conversão antes da transmissão (ou depois da recepção).

6. Codificação e decodificação. Todas as placas controladoras de rede usa a codificação Manchester, uma técnica que tem a vantagem de incorporar a informação de sincronização dentro do pacote de dados usando períodos de bits. No lugar da representação do 0 como a ausência de eletricidade e do 1 como a sua presença, os 0's e 1's são representados por mudanças na polaridade do sinal transmitido.

7. Envio e recepção dos impulsos. Os dados do quadro eletricamente codificados em impulsos são amplificados e enviados

através do cabo. (Na recepção, os impulsos são manipulados no passo da decodificação).

A execução de todos esses passos leva apenas uma fração de segundos. Enquanto você está lendo sobre estes passos, milhões de quadros teriam sido enviados através da rede.

As placas controladoras de rede e o programa de suporte reconhecem e manipulam erros, que ocorrem quando interferências elétricas, colisões, ou mal funcionamento do equipamento corrompem alguma porção de um quadro. Erros são normalmente detectados através do uso de um dado de código de redundância cíclica (CRC), interno ao quadro. O CRC é checado na recepção; se o valor calculado para o CRC não casar com o valor do CRC do quadro, o receptor não reconhece o quadro, o que significa que o receptor pede que o quadro errado seja retransmitido.

Os recursos da placa controladora de rede são usados pela placa da interface de comunicação integrada e seu programa suporte, para que os quadros de dados dos protocolos apresentados no Capítulo II e os quadros de octetos de voz, possam ser enviados e recebidos através do cabo da rede local.

A Figura 3.3 ilustra, em diagrama de blocos, os elementos que compõem uma estação com integração de voz e dados. Nesta figura são mostradas as partes integrantes da estação hospedeira, da placa controladora de rede e da ICI.

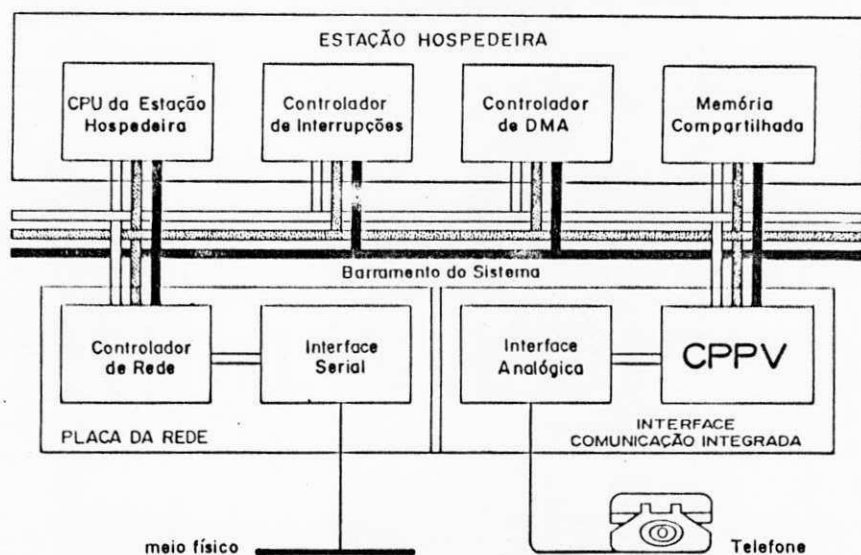


Figura 3.3. Estação hospedeira, placa da rede e ICI.

3.4 Conclusões

O conjunto de placas controladora de rede e interface de comunicação integrada, em conjunto com um telefone convencional, possibilita a integração de voz e dados em um ambiente de rede local de computadores em barramento, do tipo Ethernet. Para tal fim, estes três elementos devem trabalhar em harmonia. A placa da ICI funciona como uma central telefônica para o telefone e, ao mesmo tempo, interage com a placa controladora de rede na geração e recepção dos pacotes de octetos de voz.

CAPÍTULO IV - CO-PROCESSADOR DE PACOTES DE VOZ (CPPV)

4 Introdução

Neste capítulo, apresentamos uma visão mais detalhada da arquitetura do Co-Processador de Pacotes de Voz, mostrando os blocos internos do CPPV e sua funcionalidade. Também é apresentada a ferramenta informatizada que possibilitou o projeto do CPPV e as etapas do projeto usando esta ferramenta.

4.1 Arquitetura do CPPV.

A arquitetura proposta para o Co-Processador de Pacotes de Voz é ilustrada na Figura 4.1.

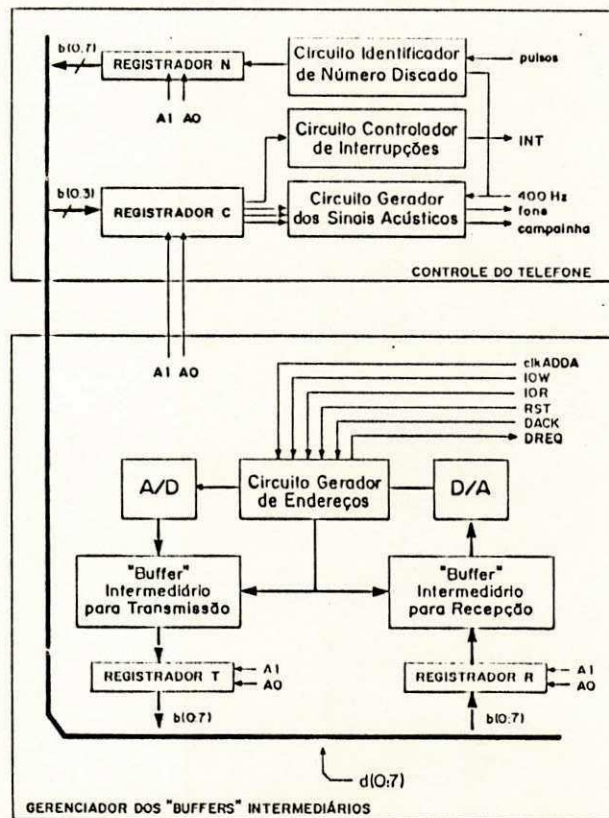


Figura 4.1. Arquitetura do CPPV.

O circuito pode ser dividido em dois blocos principais. O bloco de Controle do Telefone e o bloco Gerenciador dos "Buffers" Intermediários:

O Bloco de Controle do Telefone é o responsável:

- pelo gerenciamento dos sinais acústicos aplicados na cápsula receptora do telefone;
- pela identificação do número discado pelo usuário;
- e pela geração dos pedidos de interrupção à CPU da estação hospedeira.

O bloco Gerenciador dos "Buffers" Intermediários cuida da codificação, da decodificação e do fluxo dos blocos de octetos de voz entre o CPPV e a Memória Compartilhada.

O fluxo dos octetos de voz e de códigos de controle é feito através de uma porta que pode ser acessada pela CPU da estação hospedeira.

4.1.1 Controle do telefone

4.1.1.1 Funcionalidade

Quando o fone é retirado do gancho, a Interface Analógica sinaliza para o CPPV a ocorrência. O CPPV gera então um pedido de interrupção à CPU da estação hospedeira. Esta interrupção, quando atendida, ativa o programa gerenciador da Interface de Comunicação Integrada, residente na memória da estação hospedeira. Este programa por sua vez, inicia a fase de inserção da estação no anel lógico, programando o Controlador de Rede para

gerar e receber os quadros necessários. A estação que estiver de posse do anel lógico gera e envia o quadro de resposta ao pedido de inserção no anel lógico (Seção 2.2.2) e a informação contida neste quadro de resposta (inserção aceita ou não) é escrita na porta do CPPV. O bloco que se encarrega da Geração dos Sinais Acústicos recebe esta informação e ativa, na Interface Analógica, o sinal acústico apropriado (linha livre ou linha ocupada).

Caso a estação não consiga se inserir no anel lógico, o usuário ouvindo o sinal de linha ocupada na cápsula receptora do telefone, deve colocar o fone no gancho e fazer outra tentativa posteriormente. Quando o fone é recolocado no gancho, a interface volta ao estado de repouso, desligando o sinal de linha ocupada. Caso contrário, o usuário ouve o sinal de linha livre, e está habilitado a discar o número que identifica o seu interlocutor.

O número discado é codificado pelo bloco Identificador do Número Discado e fica disponível na porta do CPPV. Um novo pedido de interrupção é, então, gerado e o programa residente é novamente ativado para iniciar o processo que identifica o endereço do telefone chamado. O quadro de sinalização "CHAMADA" é o primeiro a ser enviado pela rede e um "timer" é inicializado.

Quando o quadro "CHAMADA" é recebido e reconhecido pelo usuário chamado, o programa residente na estação deste usuário é ativado, e o processo de inserção no anel lógico é iniciado. Caso a estação do usuário chamado não consiga se inserir no anel

lógico e enviar o quadro "CHAMANDO" para o usuário inicializador da comunicação, o "timer" do usuário inicializador da comunicação "estoura" e o programa residente desta estação escreve na porta do CPPV o código para que o sinal de linha ocupada seja ativado. Caso contrario, um "timer" é inicializado e o quadro "CHAMANDO" é enviado pela estação do usuário chamado. A informação contida no quadro "CHAMANDO" é escrita na porta do CPPV do usuário chamador, fazendo com que o sinal de telefone chamando seja ativado.

Quando o usuário chamado retira o fone do gancho antes que o "timer" "estoure", o sinal de telefone chamando é desligado e o "timer" é desabilitado, a interface fica pronta para o fluxo dos pacotes de octetos de voz, e o quadro "ATENDIDO" é enviado pela rede. Quando o quadro "ATENDIDO" é recebido e reconhecido pela interface de comunicação do usuário inicializador da comunicação, a informação contida neste quadro é escrita na porta do CPPV, fazendo com que seja iniciado o fluxo dos pacotes de octetos de voz. Caso contrário, o "timer" "estoura", fazendo com que a estação do usuário chamado envie o quadro "DESCONEXÃO" para o usuário chamador, desligue o sinal de campainha chamando e inicie o processo de saída do anel lógico.

4.1.1.2 Geração dos sinais acústicos

No sistema telefônico tradicional, as centrais telefônicas fornecem a sinalização acústica necessária para os telefones

envolvidos com a tentativa de estabelecimento de uma conexão de voz, a fim de manter os usuários envolvidos informados do progresso da conexão que está sendo tentada. Esta sinalização acústica é basicamente constituída pelos quatro sinais apresentados anteriormente no Capítulo III:

Cada um destes sinais possui características de intermitência diferentes, de modo que não possam ser confundidos. A Figura 4.2 ilustra as intermitências dos sinais acústicos.

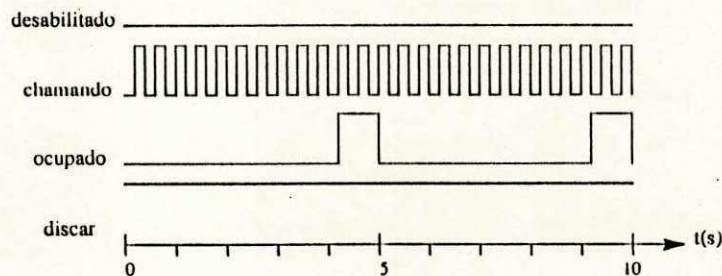


Figura 4.2. Intermitências dos sinais acústicos.

Para a geração destes sinais, um sinal de relógio de 400 Hz é aplicado na entrada do circuito Gerador dos Sinais Acusticos, onde sofre uma divisão por 200, isto é $1/(10 \cdot 10 \cdot 2)$, para gerar o sinal de telefone ocupado. O sinal de telefone ocupado é novamente dividido por 10 para gerar o sinal de telefone chamando, como ilustra o diagrama de blocos da Figura 4.3.

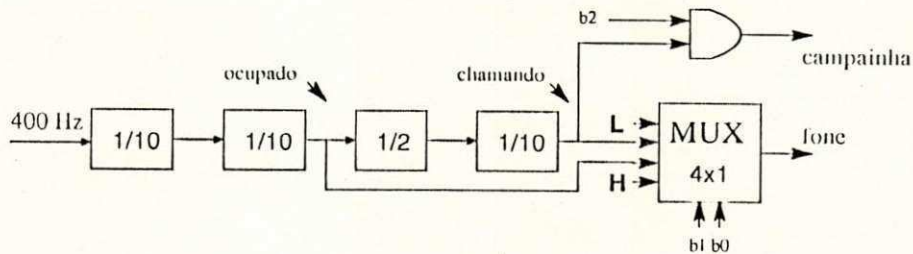


Figura 4.3. Circuito gerador dos sinais acústicos.

Tanto o sinal de telefone ocupado, quanto o sinal de telefone chamando, vão ser usados na Interface de Comunicação Integrada (ICII) como uma janela para a passagem do sinal analógico de 400 Hz. O sinal de telefone chamando também é usado como ativador da campainha do telefone do usuário chamado.

4.1.1.3 Identificador do número discado

O bloco identificador de número discado é o responsável pela geração do número de dois algarismos (ver Seção 4.1.2.5) que identifica o usuário interlocutor. Este bloco é composto basicamente por dois contadores de quatro bits cada, um responsável pela geração da unidade e o outro responsável pela dezena do número discado (Figuras 4.4a e 4.4b).

No processo de inicialização, os dois contadores recebem o valor FF16 (hexadecimal) através das entradas de "preset" (quando é dado um "reset" geral na interface de comunicação integrada). Este conteúdo será usado pelo bloco gerador de inter-

rupções apresentado posteriormente.

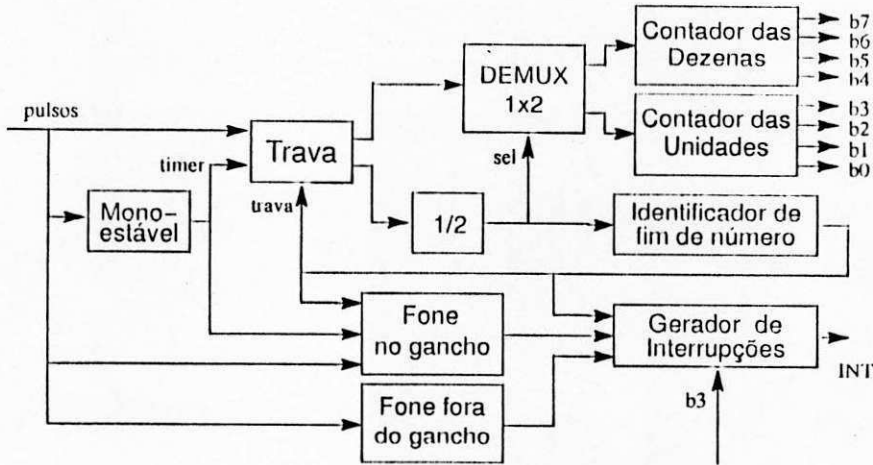


Figura 4.4a. Identificador de número discado.

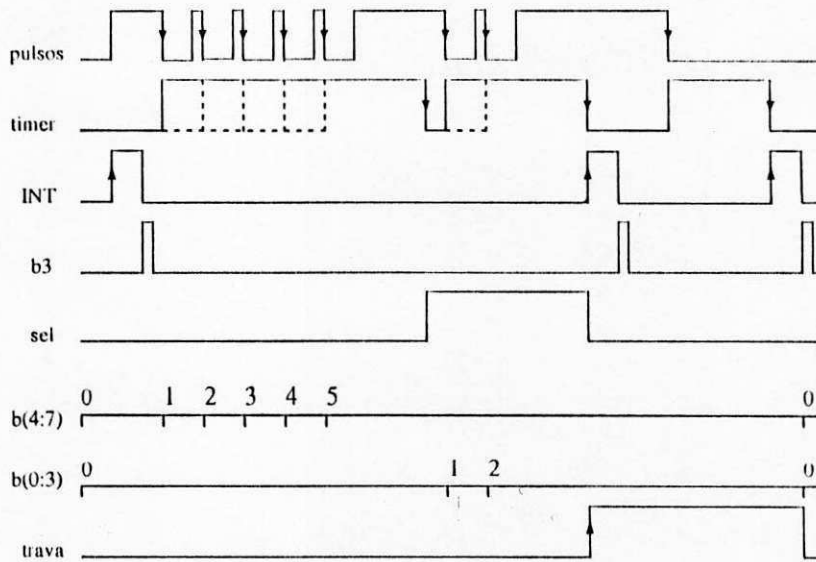


Figura 4.4b. Diagrama de tempo quando o número 52 for discado.

Os pulsos oriundos do disco do telefone chegam ao bloco identificador do número discado, e a cada borda de descida, vão incrementando o contador das dezenas e, simultaneamente, disparando (e redisparando) um circuito monoestável, cuja largura de

pulso é igual a soma da largura de um pulso gerado pelo disco do telefone (100 ms - Seção 3.1) com um período de tempo que possibilite a separação entre um número discado e o outro (400 ms [20]). O pulso gerado pelo monoestável permanece em nível lógico alto enquanto o disco do telefone permanecer gerando pulsos, pois a cada borda de descida de um pulso do disco do telefone, o monoestável é redispelado e assim tende a permanecer até que o tempo de duração do seu pulso acabe. Como consequência, após a borda de descida do último pulso do algarismo que representa a dezena, a saída do monoestável permanece em nível lógico alto durante o tempo de separação dos dígitos discados e, não ocorrendo outra borda de descida de mais um pulso do disco do telefone para redispelar o monoestável, o pulso do monoestável cai, fazendo com que o demultiplexador desvie o caminho da passagem dos pulsos do disco do telefone para o contador das unidades. Com isto tem-se o valor do algarismo que representa a dezena do número discado. O processo se repete para o valor do algarismo que representa a unidade do número discado e, na segunda descida do pulso de saída do monoestável, tem-se o número completo que identifica o usuário interlocutor e o demultiplexador volta a garantir que o caminho dos pulsos gerados pelo disco do telefone seja para o contador de dezenas. Esta segunda descida do pulso do monoestável também ativa um circuito que trava a passagem dos pulsos gerados pelo disco do telefone,

impedindo que a discagem de um novo algarismo atrapalhe o desempenho do bloco identificador de número discado. Uma terceira função da segunda borda de descida do pulso do monoestável é gerar um pedido de interrupção para o controlador de interrupções da estação hospedeira, sinalizando para o mesmo que um número completo encontra-se disponível nas saídas dos dois contadores.

Quando o fone é colocado no gancho, o circuito de trava é desbloqueado para que uma nova identificação de número discado possa ser efetivada quando o fone for novamente retirado do gancho.

4.1.1.4 Gerador de interrupções

O bloco gerador de interrupções gerencia os pedidos de interrupção que a placa interface de comunicação integrada faz a estação hospedeira. Quando o usuário retira o fone do gancho, o primeiro pedido de interrupção é enviado, através de uma das oito linhas de entrada de interrupção presentes no controlador de interrupções da estação hospedeira e escolhida pelo usuário. Este primeiro pedido interrompe o andamento do programa em execução na estação hospedeira e desvia para a rotina de tratamento da interrupção, que realiza duas ações básicas:

- ligar uma variável de estado, indicando que o pedido de interrupção que chegou foi o primeiro (o usuário retirou o fone do gancho). A utilização de uma variável de estado foi adotada para manter o número de linhas de interrupções utilizadas pela

interface de comunicação integrada a apenas uma. Outros dispositivos periféricos usam as outras linhas presentes no controlador de interrupções para realizarem suas tarefas (como, por exemplo, o teclado, a unidade de discos flexível, o controlador de rede, etc.);

- gerar um pulso de "reset" de pequena duração e enviá-lo para o bloco gerador de interrupções. Este pulso é gerado quando a rotina de tratamento da interrupção escreve, em sequência, duas palavras de controle (0816 e 0016) na porta do CPPV. O nível lógico presente no quarto "bit" alterna de ligado para desligado, gerando, deste modo, o pulso desejado na entrada de "clear" do "flip-flop" interno ao bloco gerador de interrupções que mantém a linha de pedido de interrupção ligada.

Após estas ações, a rotina de tratamento da interrupção devolve à CPU da estação hospedeira o programa que havia sido interrompido.

Quando o usuário acaba de discar os dois dígitos que formam o número que identifica o usuário interlocutor, um segundo pedido de interrupção é gerado fazendo com que o programa em execução na estação hospedeira seja novamente interrompido e a CPU passe a executar os comandos da rotina de tratamento da interrupção. Uma operação de leitura é feita na porta do CPPV onde está presente o conteúdo do registrador que guarda o valor do número discado. Se este valor for FF16, isto significa que nenhum

numero foi discado, e que o usuário colocou o fone no gancho. Se o valor for FX_{16} (onde X pode ser qualquer valor decimal), isto significa que o usuário discou errado o algarismo da unidade do número desejado, e depois colocou o fone no gancho. Em ambos os casos acima a variável de estado é desligada, indicando que o fone foi colocado no gancho. Um pulso de "reset" é novamente enviado para desativar a linha de pedido de interrupção e a rotina de tratamento da interrupção devolve à CPU da estação hospedeira o programa que havia sido interrompido.

Caso o conteúdo lido do registrador que contém o valor do número discado esteja entre 00_{16} e 99_{16} , a rotina de tratamento da interrupção vai testar se este conteúdo é um endereço de interlocutor válido e, em caso afirmativo, este endereço vai indicar o endereço de destino dos quadros que vão fluir através da rede.

4.1.2 Gerenciador dos "buffers" intermediários

4.1.2.1 Funcionalidade

O bloco Gerenciador dos "Buffers" Intermediários controla a geração e a recepção dos pacotes de octetos de voz. Quando ativado, este bloco executa quatro passos para desempenhar sua função (Figuras 4.5a e 4.5b).

1. Encher o "buffer" Intermediário de Transmissão (Tx) com as amostras codificadas pelo Conversor Analógico-Digital e esvaziar o "buffer" intermediário de recepção (Rx), enviando os

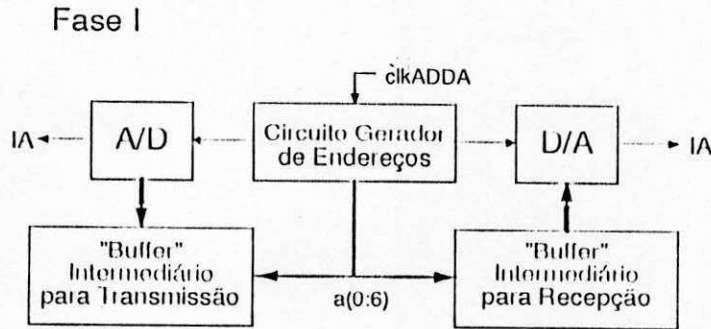


Figura 4.5a. Gerenciador dos "buffers" intermediários.

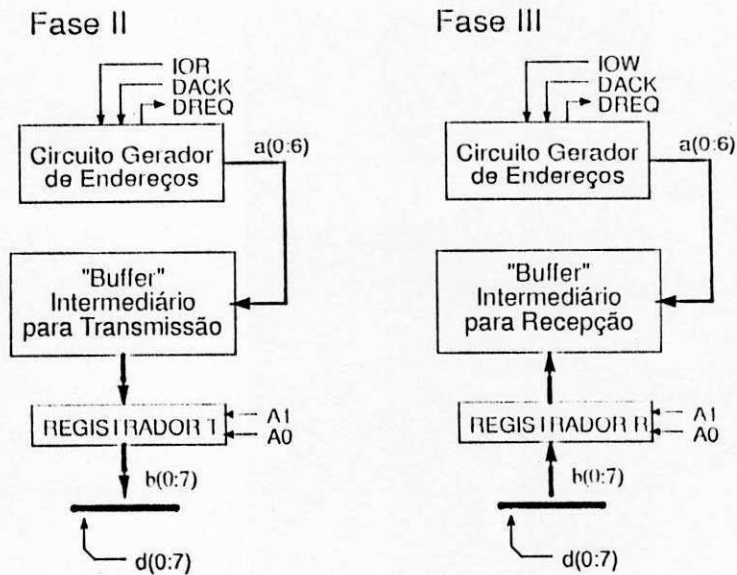


Figura 4.5b. Gerenciador dos "buffers" intermediários.

octetos de voz para o Conversor Digital-Analógico (estas duas ações são realizadas simultaneamente);

2. Transferir o bloco de octetos de voz de TX, através do registrador T, para o "buffer" de transmissão da Memória Compartilhada, via DMA;

3. Transferir o bloco de octetos de voz do "buffer" de recepção da Memória Compartilhada para RX, através do registrador R, via DMA;

4. Voltar para o passo 1, enquanto a conexão existir.

A capacidade dos "buffers" intermediários TX e RX é de 128 octetos de voz. Logo, a uma taxa de conversão de 64 Kbits/s, o tempo necessário para a execução do passo 1 é de 16 ms. O tempo requerido para a execução dos passos 2 e 3 é quase equivalente ao tempo de conversão de um ou dois octetos de voz (depende da velocidade de transferência do Controlador de DMA disponível na estação hospedeira). Considerando que no processo de "playout" (nome dado à técnica de retirada dos quadros de octetos de voz da fila de "buffers" de recepção a uma taxa constante, para manter a continuidade da conversação [25]), existe a possibilidade de se perder quadros inteiros de octetos de voz, perder um ou dois octetos de voz não irá prejudicar a qualidade da voz reproduzida.

Este ciclo continua até que um dos usuários recoloque o fone no gancho. Neste instante, o bloco Gerenciador dos "Buffers" Intermediários é desativado e o fluxo de quadros específicos para a desconexão de voz e para a retirada da estação do anel lógico é iniciado, como descrito anteriormente.

4.1.2.2 Conversor Analógico-Digital

O conversor analógico-digital utilizado no projeto do ASIC Co-Processador de Pacotes de Voz, é uma célula padrão

fechada que faz parte de uma das bibliotecas disponíveis no pacote SOLO 1400 (Figura 4.6).

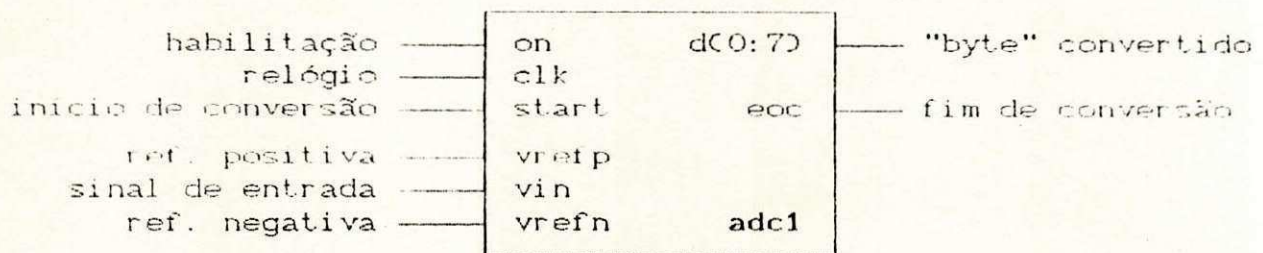


Figura 4.6. Conversor analógico-digital

O Conversor Analógico-Digital do SOLO 1400 é do tipo aproximação sucessiva e é ativado pela entrada início de conversão ("start"). A conversão começa na borda de descida do sinal "start". O valor analógico de entrada é amostrado, num processo que continua por dois ciclos de relógio. A conversão leva 14 ciclos de relógio, desde o início da amostragem e, acabado este tempo, o sinal de fim de conversão (EOC) vai para o nível lógico alto (Figura 4.7).

As linhas EOC e d(0:7) são levadas ao nível lógico baixo quando o sinal "start" vai para o nível lógico alto. As linhas d(0:7) ficam inválidas no início da conversão, e tornam-se válidas quando EOC vai para o nível lógico alto.

4.1.2.3 Conversor Digital-Analógico

O conversor digital-analógico utilizado no projeto do ASIC Co-Processador de Pacotes de Voz, é uma outra célula padrão fechada que também faz parte de uma das bibliotecas disponíveis no pacote SOLO 1400 (Figura 4.8).

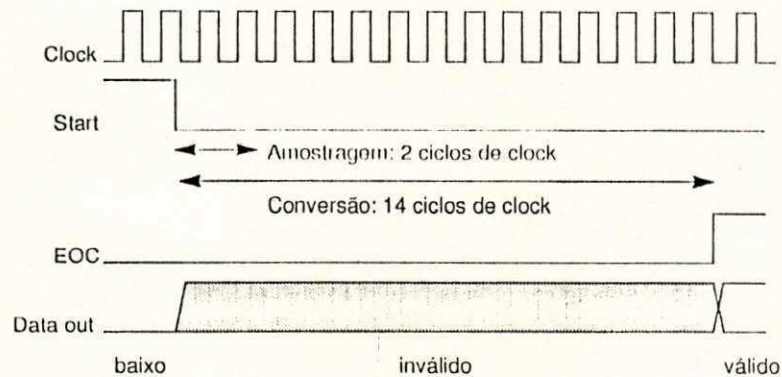


Figura 4.7. Temporização do Conv. Analógico-Digital

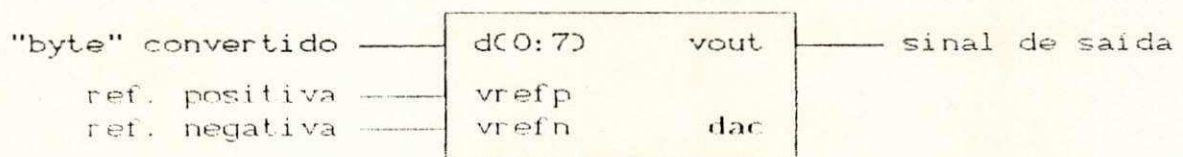


Figura 4.8. Conversor digital-analógico

O Conversor digital-analógico do SOLO 1400 é composto internamente por duas cadeias de resistores de polisilício, formando dois divisores resistivos escalonados, cada cadeia controlada por um conjunto de quatro chaves analógicas. A metade mais significativa do octeto a ser convertido dispara o primeiro conjunto de quatro chaves e fornece uma tensão de entrada para o segundo divisor resistivo. A metade menos significativa do octeto a ser convertido dispara o segundo conjunto de quatro chaves e fornece na saída do segundo divisor resistivo o valor do "byte" convertido.

4.1.2.4 "Buffer" intermediário para transmissão

O bloco "Buffer" Intermediário para Transmissão é usado para o armazenamento temporário dos blocos de octetos de voz que são gerados pelo conversor analógico-digital e depois enviados, via DMA, para a fila de "buffers" de transmissão, alocada em uma área da memória compartilhada pelo CPPV e a CPU da estação hospedeira. Este bloco é composto por uma memória RAM de 128 octetos de oito "bits" e um circuito que gerencia os sinais de controle desta memória RAM (habilitação de memória - ME, habilitação para a escrita - WE e habilitação para a saída de três estados - OED).

4.1.2.5 "Buffer" intermediário para recepção

O bloco "Buffer" Intermediário para Recepção é usado para o armazenamento temporário dos blocos de octetos de voz que são enviados, via DMA, da fila de "buffers" de recepção, alocada em uma área da memória compartilhada pelo CPPV e a CPU da estação hospedeira, para o conversor digital-analógico. O bloco "buffer" intermediário para recepção também é composto por uma memória RAM de 128 palavras de oito "bits" e um circuito que gerencia os sinais de controle desta memória RAM (habilitação de memória - ME, habilitação para a escrita - WE e habilitação para a saída de três estados - OED).

4.1.2.6 Dimensionamento dos "buffers" intermediários

O dimensionamento dos "buffers" intermediários (128 posi-

ções de memória) foi feito levando-se em consideração os seguintes fatores:

- o atraso fim-a-fim não deve ser superior a 200 ms [15];
- deseja-se utilizar 10% da banda disponível em uma rede

tipo Ethernet, para o tráfego de voz.

A Tabela 4.1 ilustra o formato dos quadros IEEE 802.3.

Pr	Início	Dest	Font	Tam	Dados	CRC	Fim	campos
62	2	48	48	16	1024	32	0	bits

Tabela 4.1. Formato dos quadros IEEE 802.3.

O conversor analógico-digital codifica a uma taxa de 64 kbits/s. O campo de dados contém 1024 bits (128 octetos). O conversor A/D leva 16 ms para converter os 128 octetos de voz. A banda de 64 kbits/s aumenta para 77 kbits/s, isto é, $(1024 + 208) \text{ bits} / 16 \text{ ms}$). Deseja-se utilizar 10% da banda total disponível, ou seja 1 Mbit/s. $1 \text{ Mbit/s} / 77 \text{ kbits/s}$ proporcionam 12 conversações simultâneas ("half-duplex"). As 12 conversações ocupam 924 kbits/s $(12 \cdot 77 \text{ kbits/s})$. Os 76 kbits/s restantes $(1 \text{ Mbit/s} - 924 \text{ Kbits/s})$ são usados para detectar colisões ("jam" - sinal de reforço de colisão), sentir a portadora e outras tarefas do protocolo. O limite do atraso fim-a-fim é respeitado, pois $12 \text{ conversações} \cdot 16 \text{ ms} = 192 \text{ ms}$, que é menor que 200 ms. Estes fatos também explicam a necessidade de apenas dois dígitos.

4.1.2.7 Memória compartilhada

Os pacotes de octetos de voz que são gerados pelo CPPV são armazenados temporariamente no "buffer" intermediário de transmissão (TX). Quando este "buffer" fica preenchido com as 128 amostras de voz, ele é descarregado, via DMA, para uma fila encadeada de "buffers" que estão "esperando a vez" de serem transmitidos, via o meio físico da rede, para a estação de destino. Esta fila de "buffers" se localiza em uma determinada área da memória da estação hospedeira.

Do mesmo modo, os pacotes de octetos de voz que chegam, via o meio físico da rede, de uma outra estação, são colocados em uma fila de "buffers" de recepção, que ficam "esperando a vez" de serem descarregados, via DMA, para o "Buffer" Intermediário de Recepção (RX), onde serão expostos ao Conversor Digital-Analógico para a decodificação. Esta fila de recepção também está localizada em uma determinada área da memória da estação hospedeira.

Como tanto o CPPV quanto a placa Controladora de Rede podem acessar estas mesmas áreas de memória de madeira compartilhada, Chamamos estas áreas de Memória Compartilhada.

4.1.2.6 Gerador de endereços

O bloco Gerador de Endereços é o responsável pelo fornecimento dos endereços para os "buffers" intermediários de transmissão (Tx) e recepção (Rx). Este bloco é basicamente um contador composto de sete "flip-flops", que geram os endereços

de forma sequencial, de 0016 a 7F16, cobrindo todas as 128 posições de memória dos dois "buffers" intermediários.

O sinal de relógio que incrementa o Gerador de Endereços (ClkADDA) é derivado do sinal de relógio que entra no Conversor Analógico-Digital. No final de cada conversão (sinal EOC - fim de conversão, gerado pelo Conversor Analógico-Digital), estes endereços vão sendo incrementados de 0016 a 7F16, e a cada posição de memória dos "buffers" intermediários apontada pelo Gerador de Endereços, Tx vai sendo preenchido com um novo "byte" gerado pelo Conversor Analógico-Digital e, simultaneamente, Rx vai sendo esvaziado para um "latch" que tem sua saída de dados ligada a entrada do Conversor Digital-Analógico. Quando a contagem atinge 7F16, Tx está totalmente preenchido com 128 novas amostras de voz, e todas as 128 amostras de voz Rx foram passadas para o Conversor Digital-Analógico (Figura 4.9).

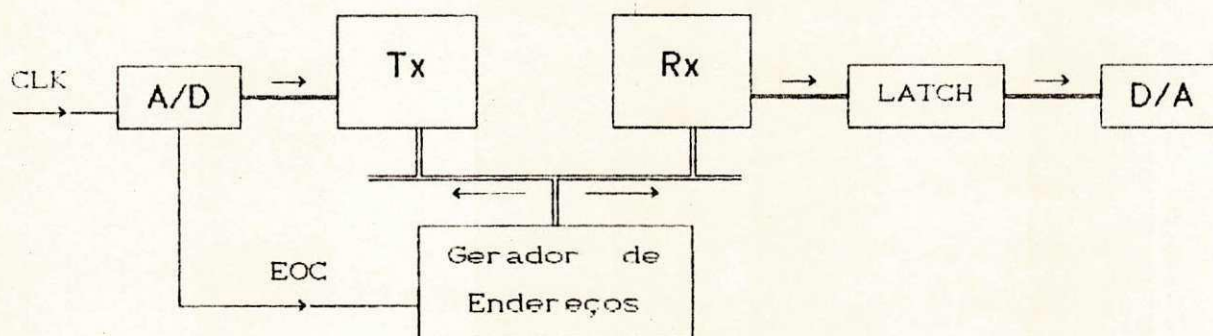


Figura 4.9. Primeira etapa

Depois que esta primeira etapa fica concluída, o sinal de relógio que incrementa o contador do Gerador de Endereços não é mais derivado do sinal de relógio do Conversor Analógico-Digital.

Inicia-se agora a fase de transferência dos blocos de octetos de voz de Tx, para a cauda da fila dos "buffers" de transmissão da Memória Compartilhada, e do bloco de octetos de voz que está na cabeça da fila dos "buffers" de recepção da Memória Compartilhada para Rx. Estas duas transferências são realizadas via acesso direto a memória (DMA), e os sinais RIO (lê de um dispositivo de entrada/saída) e WIO (escreve em um dispositivo de entrada/saída), gerados pelo Controlador de DMA da CPU da estação hospedeira, são usados como relógio para o contador do Gerador de Endereços.

A segunda etapa então, é caracterizada pela transferência do bloco de octetos de voz de Tx, para a cauda da fila dos "buffers" de transmissão da Memória Compartilhada, usando o sinal RIO como relógio que incrementa as posições de memória de Tx, para que o conteúdo de Tx possa ser transferido para a Memória Compartilhada, como ilustra a Figura 4.10.

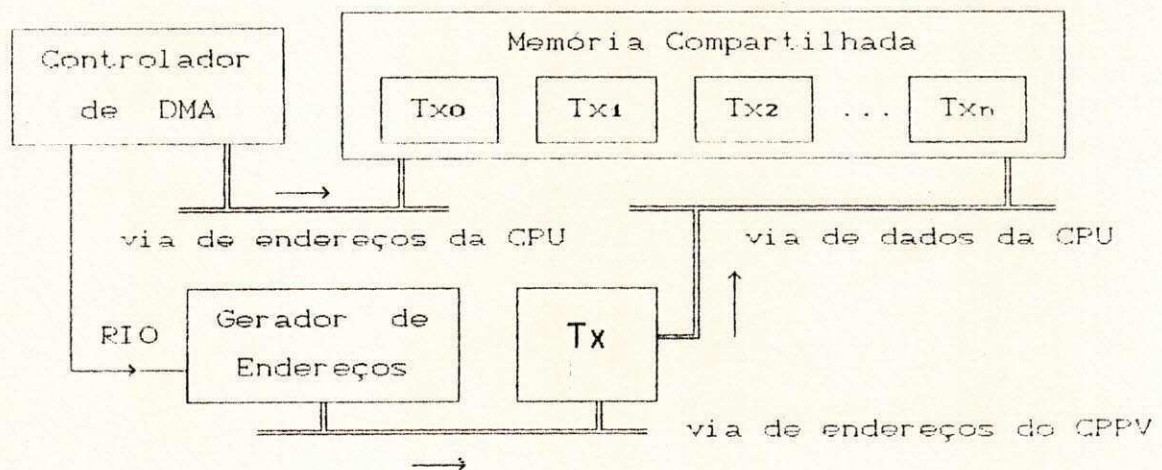


Figura 4.10. Segunda etapa

E, finalmente, a terceira etapa é caracterizada pela transferência do bloco de octetos da cabeça da fila dos "buffers" de recepção da Memória Compartilhada, para Rx, usando o sinal WIO como relógio que incrementa as posições de memória de Rx, para que Rx possa receber o bloco de octetos de voz da Memória Compartilhada, como ilustra a Figura 4.11.

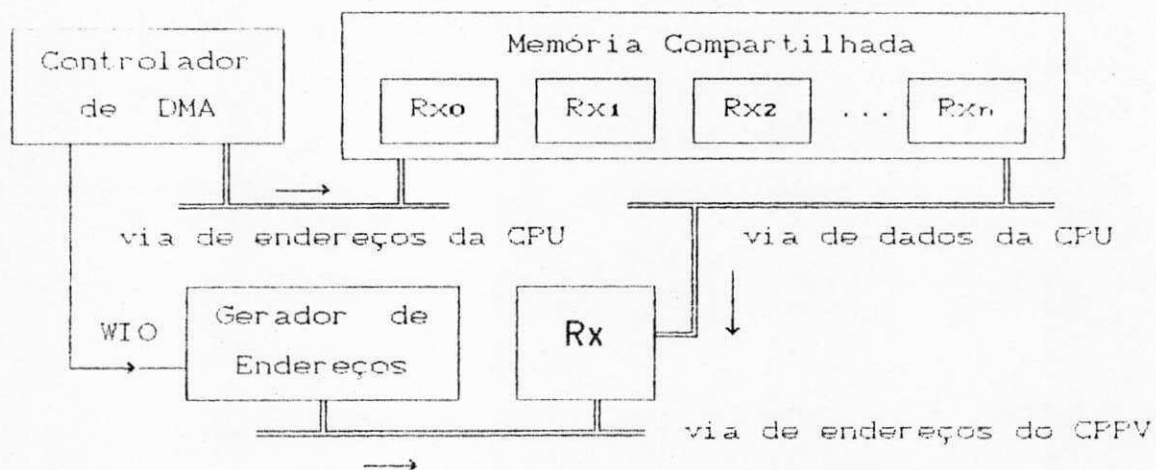


Figura 4.11. Terceira etapa

4.2 Implementação do CPPV

O projeto do Co-Processador de Pacotes de Voz foi desenvolvido numa estação de trabalho SUN 3, instalada no Laboratório de Sub-sistemas Integráveis (LSI) da Escola Politécnica da Universidade de São Paulo (EPUSP).

O programa utilizado no projeto foi o pacote SOLO 1400 da empresa europeia European Silicon Structures (ES2) para projetos de com células padrão ("standard cell"), descrito na seção anterior.

Utilizando a metodologia de projeto "top-down", a arqui-

tetura do circuito do CPPV foi dividida em vários níveis hierárquicos, onde blocos menores entram na formação de blocos maiores, até o projeto atingir um nível superior, formado pelo circuito completo e os "pads" de alimentação, entrada, saída e bidirecionais (Apendice A).

A maioria dos blocos foi projetada utilizando um programa de captura esquemática do pacote SOLO 1400, o DRAFT. Através deste programa pode-se montar um circuito com células padrão presentes nas bibliotecas de células do SOLO 1400, com células já construídas pelo projetista, e com células geradas por um gerador automático parametrizado, o GENERATE. As memórias RAM que formam os "buffers" intermediários para transmissão e recepção são geradas pelo programa GENERATE. A listagem abaixo mostra as características das RAMs geradas para o projeto do CPPV.

Data Sheet for Block RAM128TX

Generator	: RAM v1.2.8	Words	: 128
Date/Time	: Fri Nov 1 1991 13:44:17	Bits per word	: 8
		Rows	: 32
Dimensions	: 1652.40 x 1121.60 μm	Columns	: 32
Area	: 1.85 sq. mm		
Rotation	: None	Memory Enable	: Inverted
Technology	: ECPD15	Write Enable	: Inverted
		Output Enable	: Inverted

Process Type : Industrial

Parameter	Description	Min	Typ	Max	
tacc	- Access time from ME	4.20	8.92	17.49	nS
macc	- Load dependent delay	0.14	0.29	0.56	nS/pF
tmpw	- ME pulse width	1.91	4.06	7.97	nS
tmed	- Output disable time from ME	2.45	5.23	10.25	nS
toee	- Output enable time from OE	1.87	3.97	7.79	nS
tpre	- Minimum precharge time	1.22	2.60	5.10	nS
twpw	- Write pulse width	1.95	3.93	7.70	nS
tws	- Data setup to write end	0.37	1.95	3.62	nS
twdh	- Data hold to write end	0.00	0.00	0.00	nS
twrh	- Write hold time	2.05	4.37	9.55	nS
twod	- Output disable time from WE	2.66	5.65	11.07	nS
toed	- Output disable time from OE	2.39	5.08	9.95	nS
trds	- Read pulse setup time	0.00	0.00	0.00	nS
trdh	- Read pulse hold time	0.00	0.00	0.00	nS
tads	- Address setup time	0.00	0.00	0.00	nS
tadh	- Address hold time	0.83	1.75	3.45	nS
pcon	- Power consumption	----	----	2.38	mW/MHz

Process Type : Military

Parameter	Description	Min	Typ	Max	
tacc	- Access time from ME	3.67	8.92	21.12	nS

macc - Load dependent delay	0.12	0.29	0.68	nS/pF
tmpw -ME pulse width	1.67	4.06	9.63	nS
tmed - Output disable time from ME	2.15	5.23	12.39	nS
toee - Output enable time from OE	1.54	3.97	9.41	nS
tpre - Minimum precharge time	1.07	2.60	6.16	nS
twpw - Write pulse width	1.52	3.93	9.30	nS
twds - Data setup to write end	0.76	1.95	4.37	nS
twdh - Data hold to write end	0.00	0.00	0.00	nS
twrh - Write hold time	1.80	4.37	10.34	nS
twod - Output disable time from WE	2.33	5.65	13.38	nS
toed - Output disable time from OE	2.09	5.08	12.03	nS
trds - Read pulse setup time	0.00	0.00	0.00	nS
trdh - Read pulse hold time	0.00	0.00	0.00	nS
tads - Address setup time	0.00	0.00	0.00	nS
tadh - Address hold time	0.73	1.76	4.17	nS
pcon - Power consumption	----	----	2.38	mW/MHz

Ao final de cada etapa do projeto, o bloco implementado passou por uma bateria de testes de simulação lógica, onde vetores de teste específicos foram aplicados, no intuito de provar o bloco nas condições mais genéricas. A empresa que projetou o pacote SOLO 1400 garante o funcionamento elétrico do circuito nas mesmas condições de teste a que o circuito foi submetido durante a etapa de simulações lógicas [23].

Por exemplo, uma metodologia de teste aplicada nas memórias RAM que compõem os "buffers" intermediários para transmissão e recepção, proposta pelo próprio fabricante do pacote SOLO 1400, foi a de escrever os padrões 10101010z e 01010101z alternadamente nas posições da RAM, e depois ler estas posições e comparar com o que foi escrito. Depois escrever os padrões 01010101z e 10101010z alternadamente nas posições da RAM, novamente ler estas posições, e comparar com o que foi escrito. Esta metodologia não testa todas as condições possíveis de trabalho das RAMs pois o ideal seria que se escrevesse todas as 2^8 possibilidades de padrão de "bits" em cada posição de memória e depois as lesse, mas isso acarretaria na geração de um número gigantesco de vetores de teste. Porém, esta metodologia permite que cada "bit" formador de cada palavra possa assumir os dois níveis lógicos alternadamente, e assim garantir o bom funcionamento destas memórias.

Para o teste de contadores, um vetor de teste necessário seria a geração de um número de pulsos de relógio maior que o valor máximo de contagem, para permitir que o contador atinja o seu máximo valor de contagem e recomece a contagem do início. Por exemplo, um vetor de teste para o gerador de endereços, composto por um contador até 127 (7 bits de endereço), seria a geração de um trem de pulsos com um número de pulsos maior que 128, aplicados na entrada de relógio do contador.

Para o teste de blocos compostos por lógica combinacional, como por exemplo, o bloco identificador do número discado, o processo de criação de vetores de teste exige bom senso por parte do projetista, pois não havendo nenhuma organização estrutural por parte do circuito a ser testado, seria necessário um grande número de vetores de teste para garantir que todas as partes do circuito trabalhem de acordo com o projetado. O que foi feito para testar o bloco identificador do número discado, foi a criação de oito vetores de teste organizados em quatro pares de trens de pulso simulando os pulsos gerados pelo disco do telefone, onde o primeiro par de trens de pulso é composto por um número par e outro ímpar de pulsos, o segundo par é composto por um número ímpar e outro par de pulsos, o terceiro par é composto por dois números pares de pulsos e o quarto par é composto por dois números ímpares de pulsos. Esta simulação representa a retirada do fone do gancho, a discagem de um número e a colocação do fone no gancho, esta ação se repetindo para quatro números diferentes.

Depois que todo o circuito foi projetado, e seus blocos devidamente simulados, a seqüência de programas que o pacote SOLO 1400 exige para dar continuidade ao projeto, foi executada. Como resultado final, é gerado um arquivo no formato CIF com o conjunto de mascaras necessárias para as várias etapas dos processos químicos que vão atuar internamente, e sobre a superfície da pas-

tilha de silício, permitindo a implementação física do ASIC. O ASIC do CPPV será então encapsulado e devolvido para que o projetista possa fazer os testes elétricos e lógicos em um ambiente real de testes.

O ASIC Co-Processador de Pacotes de Voz ficou com as seguintes características físicas:

Pads	33;
Array area (mm)	$3.44 \times 3.01 = 10.34 \text{ mm}^2$;
Active chip area (mm)	$4.65 \times 4.29 = 19.94 \text{ mm}^2$;
Die size (mm)	$4.86 \times 4.50 = 21.86 \text{ mm}^2$.

O diagrama de pinos e o leiaute das mascaras superpostas podem ser observados no Apêndice B.

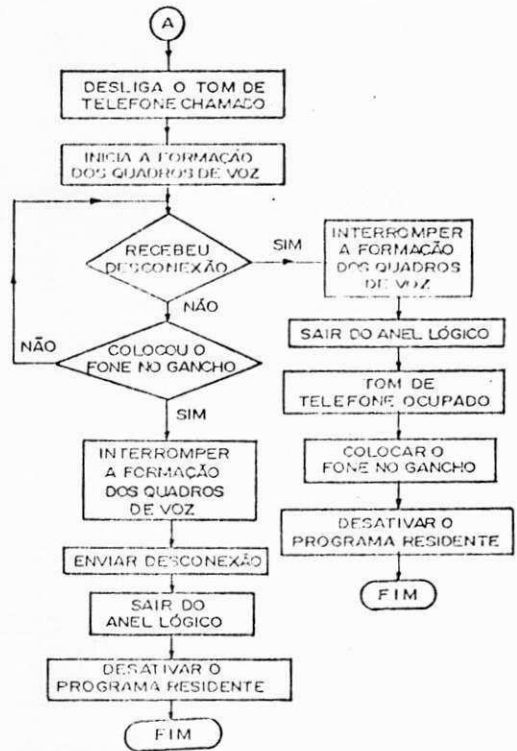
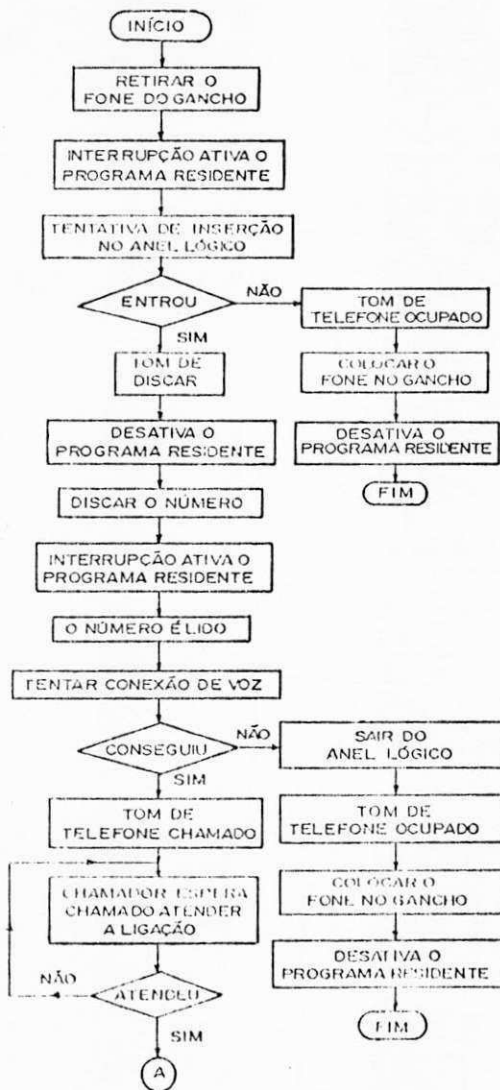
4.3 Ambiente de "software" da ICI

O projeto do ASIC CPPV foi desenvolvido levando em consideração a arquitetura da estação hospedeira e os sinais que a CPU desta estação gera e recebe.

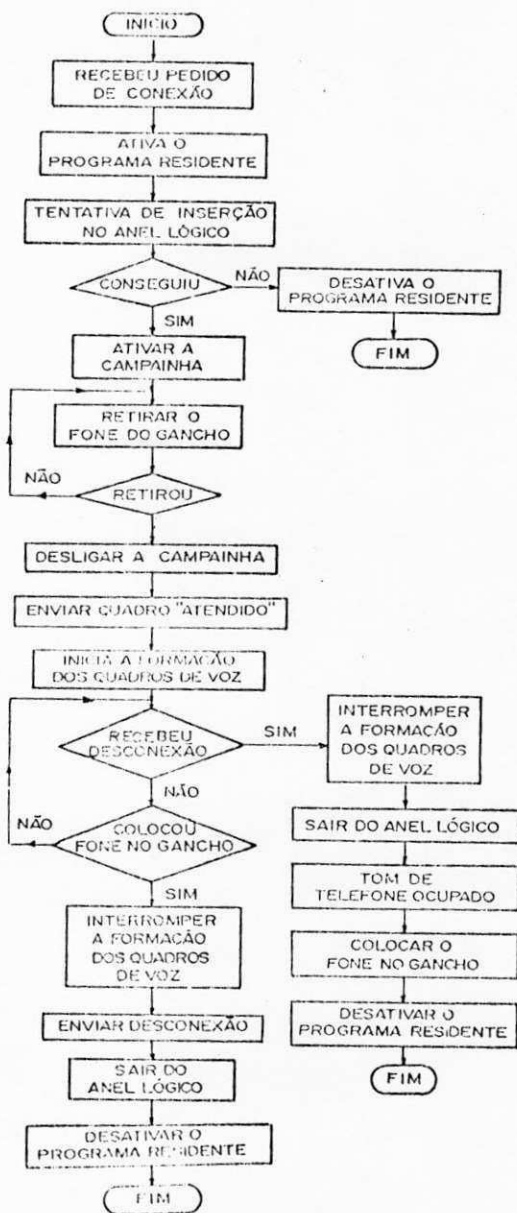
Portanto, a pessoa que vier a desenvolver o programa para gerenciar as operações do CPPV deve ter noções sobre a arquitetura interna da estação hospedeira que vai abrigar a ICI.

Esta pessoa deve seguir um roteiro mínimo durante a elaboração do programa gerenciador, para que o mesmo possa realmente usar toda a potencialidade da arquitetura do ASIC proposto. Os fluxogramas das páginas que se seguem mostram este roteiro mínimo.

PROCESSO DE COMUNICAÇÃO DE VOZ
VISTO PELO LADO DO USUÁRIO CHAMADO



PROCESSO DE COMUNICAÇÃO DE VOZ
VISTO PELO LADO DO USUARIO CHAMADO



4.4 Conclusão

O arquivo no formato CIF com o conjunto de máscaras do projeto do CPPV foi gravado em fita magnética para posterior envio para a empresa responsável pela fundição do ASIC. A forma de contactar esta empresa poderá ser feita através do Programa Multi-Usuário (PMU), que envolve a divisão de uma determinada área de silício entre universidades e organizações de pesquisa, ou através de um projeto francês similar (CMP), por intermédio do programa de cooperação internacional CAPES/COFECUB 85-88.

CAPÍTULO V - CONCLUSÕES

A mudança provocada pela invenção do telefone foi uma das maiores desta época. Hoje não se consegue pensar em um escritório, uma fábrica ou uma instituição qualquer, onde a transmissão de voz não tenha um papel fundamental. A integração dos serviços de voz e dados em uma rede de computadores vem permitir um grande número de aplicações antes inexecutáveis. Chamadas pelo nome do usuário, chamadas via diretório de nomes, recados selecionados deixados para usuários específicos, atendimento selecionado de usuários, e discagem repetida são apenas algumas das aplicações que se pode citar.

A integração de voz e dados em uma rede de computadores não é, no entanto, um assunto trivial. Soluções para vários tipos de problemas devem ser encontradas, e nem sempre é possível a implementação de todas, tendo-se que se contentar com soluções de compromisso. Os problemas começam desde a simples digitalização dos sinais de voz, até a implementação das aplicações propriamente ditas. Contudo, vários sistemas tem surgido mostrando a viabilidade de suas implementações e deixando já antever sua grande aplicabilidade em um futuro próximo.

Dentro desta filosofia é que foi projetado e implementado um circuito integrado, cuja aplicação específica é a de gerenciar uma placa de interface de comunicação, que integra o tráfego de voz gerado e recebido por um telefone convencional, em uma rede de computadores configurada em barramento (Ethernet).

O uso de um ASIC que integre o maior número possível de circuitos se justifica quando a demanda para o uso deste ASIC é da ordem de grandeza de milhares de unidades. Esta é a visão comercial da utilização da integração de sistemas em pastilhas de silício. Mas em um ambiente de pesquisa, a produção de alguns protótipos (algumas dezenas) que utilizam o ASIC pode ser realizada, através de programas como o PMU brasileiro, seu similar francês (CMP) ou o PMU líbero-americano. Este fato justifica o uso de ASICs para implementação de protótipos a baixo custo.

Durante o Estágio Integrado, necessário para a conclusão do curso de Graduação na Universidade Federal da Paraíba (UFPb), o projeto de um circuito integrado de aplicação específica (ASIC) foi implementado no Laboratório de Sub-Sistemas Integráveis (LEI) da Escola Politécnica da Universidade de São Paulo (EPUSP). O pacote Solo 1400 da empresa European Silicon Structures (ESS) havia sido instalado a pouco tempo em algumas estações de trabalho SUN do referido laboratório, e poucas pessoas, na época, estavam se iniciando no uso desta ferramenta informatizada para projetos de CAD para microeletrônica. Porém, os elementos de biblioteca utilizados no primeiro circuito integrado eram elementos básicos de pouca complexidade, e os recursos oferecidos pelo pacote não foram suficientemente explorados.

A experiência adquirida durante a fase de implementação do primeiro ASIC foi usada posteriormente, durante a fase do pro-

jeto do ASIC do CPPV, objeto desta dissertação. Baseado nesta experiência, uma arquitetura inicial foi definida para a implementação do projeto, mas nesta fase, algumas modificações foram realizadas tendo em vista o uso de novos recursos e elementos de biblioteca mais poderosos do Solo 1400. Como consequência, um maior aprendizado da ferramenta foi adquirido e projetos que venham a ser realizados posteriormente terão maior grau de eficiência (Área de silício ocupada e tempo de projeto).

O projeto foi concluído e está gravado em fita magnética para posterior envio para a empresa ES2, onde será difundido.

Como sugestões para a continuidade do trabalho, propomos a realização de testes das partes integrantes do circuito integrado (quando a implementação física do ASIC estiver concluída e disponível). A placa da interface de comunicação integrada deve ser montada para receber o CPPV para os testes dos circuitos internos isoladamente, e depois, dois protótipos devem ser colocados em estações de uma rede Ethernet (como por exemplo uma rede Novell). Esta rede vem com a documentação necessária para a utilização dos seus recursos para o tráfego de pacotes de dados.

Como um projeto posterior, sugerimos a integração de outros elementos ao CPPV, como um microprocessador dedicado, mais memória e canais independentes de DMA, para que a ICI possa livrar ao máximo as tarefas que seriam hoje realizadas pela CPU da estação hospedeira. A integração de um microprocessador ao

ASIC do CPUV daria à ICI uma maior autonomia no processo de operação, pois seu programa supervisor rodaria neste microprocessador e a CPU da estação hospedeira só seria interrompida nos instantes em que fosse essencialmente necessário. Este projeto implicaria em um estudo bem mais profundo sobre arquitetura e integração de sistemas digitais, e abriria um estudo sobre a aplicação desta ICI em outras topologias de rede, como redes em anel, visto que teríamos o programa de suporte da ICI rodando na própria ICI, e este programa poderia se adaptar a topologia de rede em que a interface de comunicação integrada estivesse localizada.

Uma vez que estes protótipos tenham sido testados e seu desempenho levantado, poder-se-ia pensar em produzi-los em escala comercial, atendendo entre outras áreas, escritórios que estão em fase de automação de suas funções, ou em ambientes de pesquisa que envolva um certo número de pesquisadores que necessitem de uma maior interação.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] - HARRINGTON, E. A., "Voice /data integration using circuit switched networks", IEEE Transaction on Communication, Vol. 28, No. 6, Jun., 1980.
- [2] - MAXEMCHUIK, N. F., "A variation on CSMA/CD that yields movable slots in integrated voice/data local network", The Bell System Technical Journal, Vol. 61, No. 7, Set., 1982.
- [3] - CARVALHO, J. M. e MOURA, J. A. B., Arquitetura de uma interface para rede local sob tráfego de voz e dados. Anais do 3° SBRC , Rio de Janeiro, RJ, Abril, 1985.
- [4] - SOARES, L. F. G. e outros, Sistema de telefonia para um ambiente de rede local com voz e dados integrados, Anais do 6° SBRC, Belo Horizonte, MG, Março, 1988.
- [5] - GIOZZA, W. F., de ARAÚJO, J. F. M., MOURA, J. A. B e SAUVE, J. P - Redes Locais de Computadores - Tecnologia e Aplicações. McGraw-Hill, 1986.
- [6] - CARVALHO, J. M. e GIOZZA, W. F., Interface de comunicação com integração de voz/dados. Anais do 5° SBT, Campinas, São Paulo, Set., 1987.
- [7] - SILVA, I. S., GIORDANO, O. e GIOZZA, W. F., Circuito integrado para interface de comunicação com integração de voz e dados em redes locais, Anais do 5° SBMicro, Campinas, SP, Julho, 1990.
- [8] - DORROS, I., "ISDN IEEE Commun. Mag., Vol. 19, Mar., 1981.

- [9] - TANH. W., "ISDN - New vistas in information processing". IEEE Trans. on Commun. Mag., Vol. 24, Nov., 1986.
- [10] - CHEN, T. M. e MESSERSCHIMMIT, D. G., "Integrated voice/data switching". IEEE Commun. Mag., Vol. 26, Jun., 1988.
- [11] - ROSS, M., "Circuit versus packet switching". Fundamentals of Digital Switching. (J. MacDonalds, ed), NY: Plenum, 1983.
- [12] - HEGGESTAD, H. M., McAULEY, R. J. e TIERNEY, J., "Practical considerations for speech digitizing at rates from 64.0 to 0.6 Kbits/s". Global Telecommunications Conference, Miami, 1982, B 3.8.1 - B 3.8.8.
- [13] - BRADY, P. T., "A statistical analysis of on-off patterns in 16 conversations", Bell System Technical Journal, Vol. 48, Jan., 1968.
- [14] - FLANAGAN, J. L., SCHUEDER, M. R., ATAL, B. S., CORCHIERE, R. E., JAYANT, N. S. e TRIBOLET, J. M., "Speech coding", IEEE Transactions on Communications, Vol. COM-27, No. 4, Apr., 1979.
- [15] - KLEMMER, E. T., "Subjective evaluation delay in telephone conversations", Bell System Technical Journal, Vol. 46, 1967.
- [16] - FORGIE, J. W., "Speech transmission in packet-switching store and forward networks", AFIPS National Computer Conference 1975, Vol. 44, p.137-142.

[17] - TURNER, J., "Design of an integrated services packet networks", IEEE J. on Sel. Areas in Commun., Vol. SAC-4, Nov., 1986.

[18] - GOPAL, P. M., "Voice transmission on local area network", University of Waterloo, CCNG REPORT T-131, Apr., 1985.

[19] - TSUKUDA, K. Y., YUKIMATSU, K. e OHNISHI, H., "Integrated services data switching network", Sixth Data Communications Symposium, Nov., 1979.

[20] - CPA - T - Requisitos mínimos de interfaces com equipamentos terminais. Sistema de Documentação Telebrás, Série Engenharia. CPqD - Campinas - SP. 220-250-701 (Padrão). Emissão 03, Out. 1988.

[21] - RYAN, R., MARSHALL, G., BEACH, R. and KERMAN, S., "Local network architecture proposed for work stations", Electronics, Aug., 1981.

[22] - Solo 1400 - User Guide, ES2, 1990.

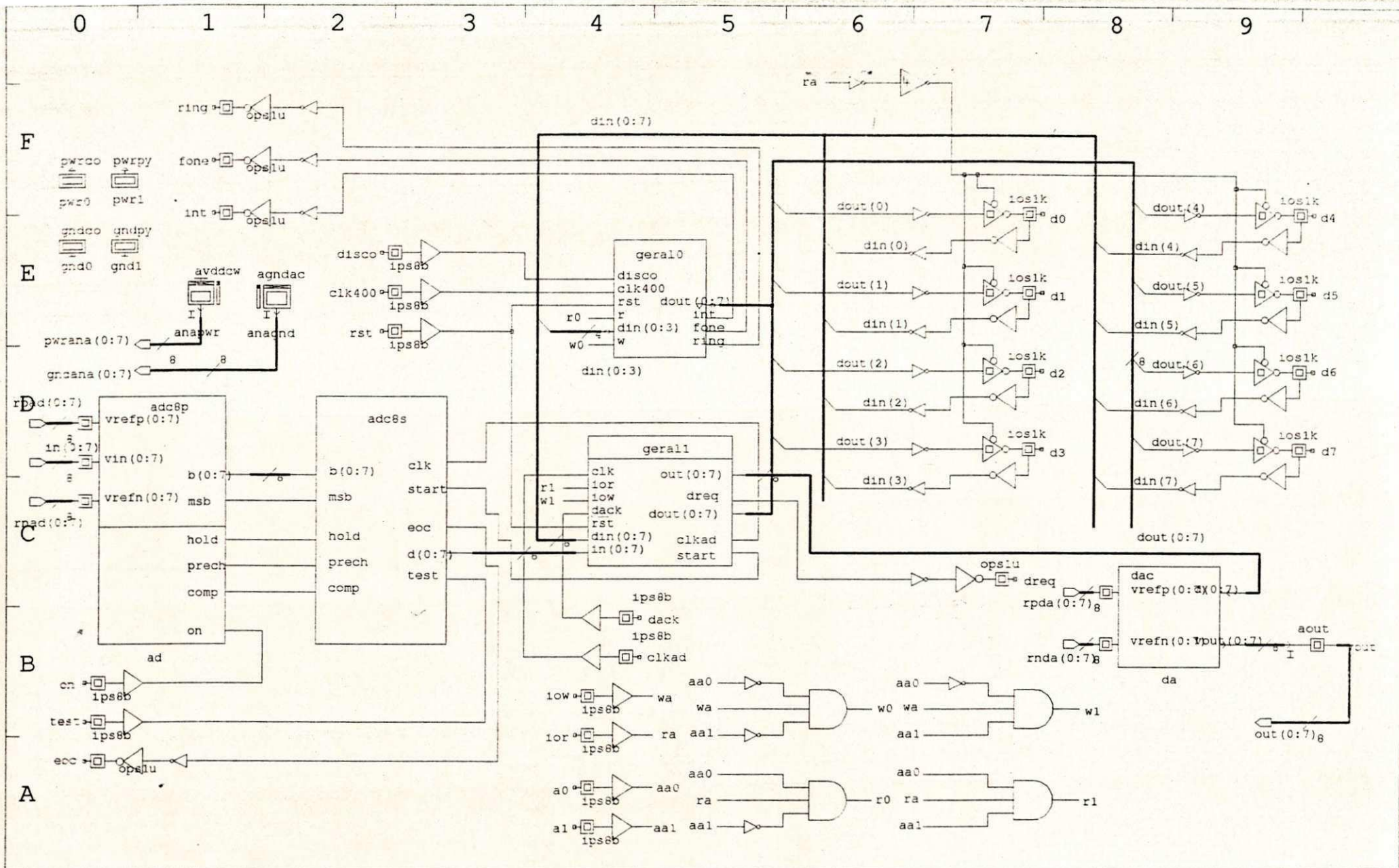
[23] - Solo 1400 - Simulation, ES2, 1990.

[24] - MORAES, A. M. e GIOZZA, W. F., "Co-Processador de Pacotes de Voz para redes locais, Anais do 6° SBCCI, Jaguariúna, SP, Outubro, 1990.

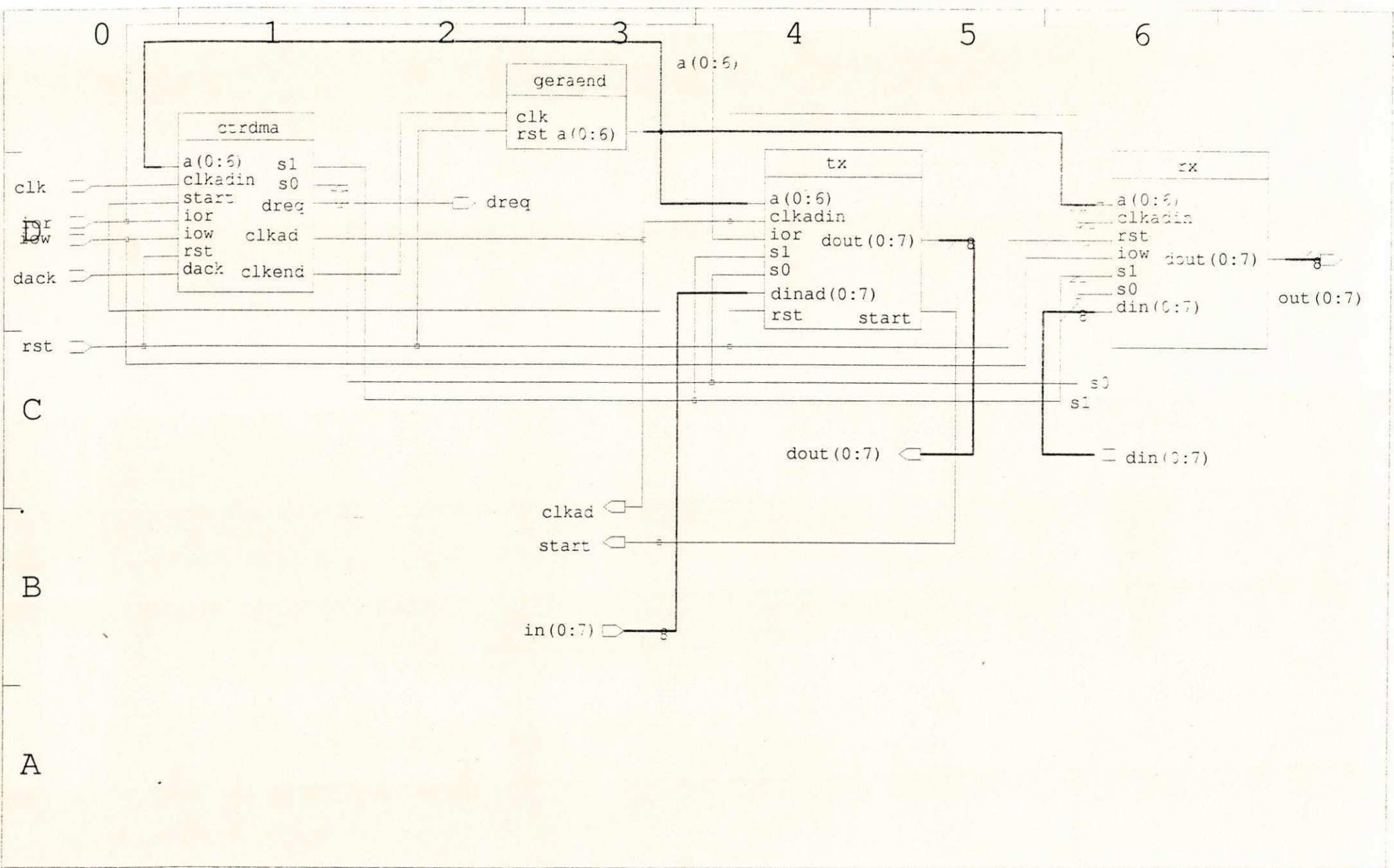
[25] - MONTGOMERY, W. A., "Techniques for packet voice synchronization", IEEE Journal on Selected Areas in Communications, Vol. SAC-1, No. 6, Dec., 1983.

APENDICE A

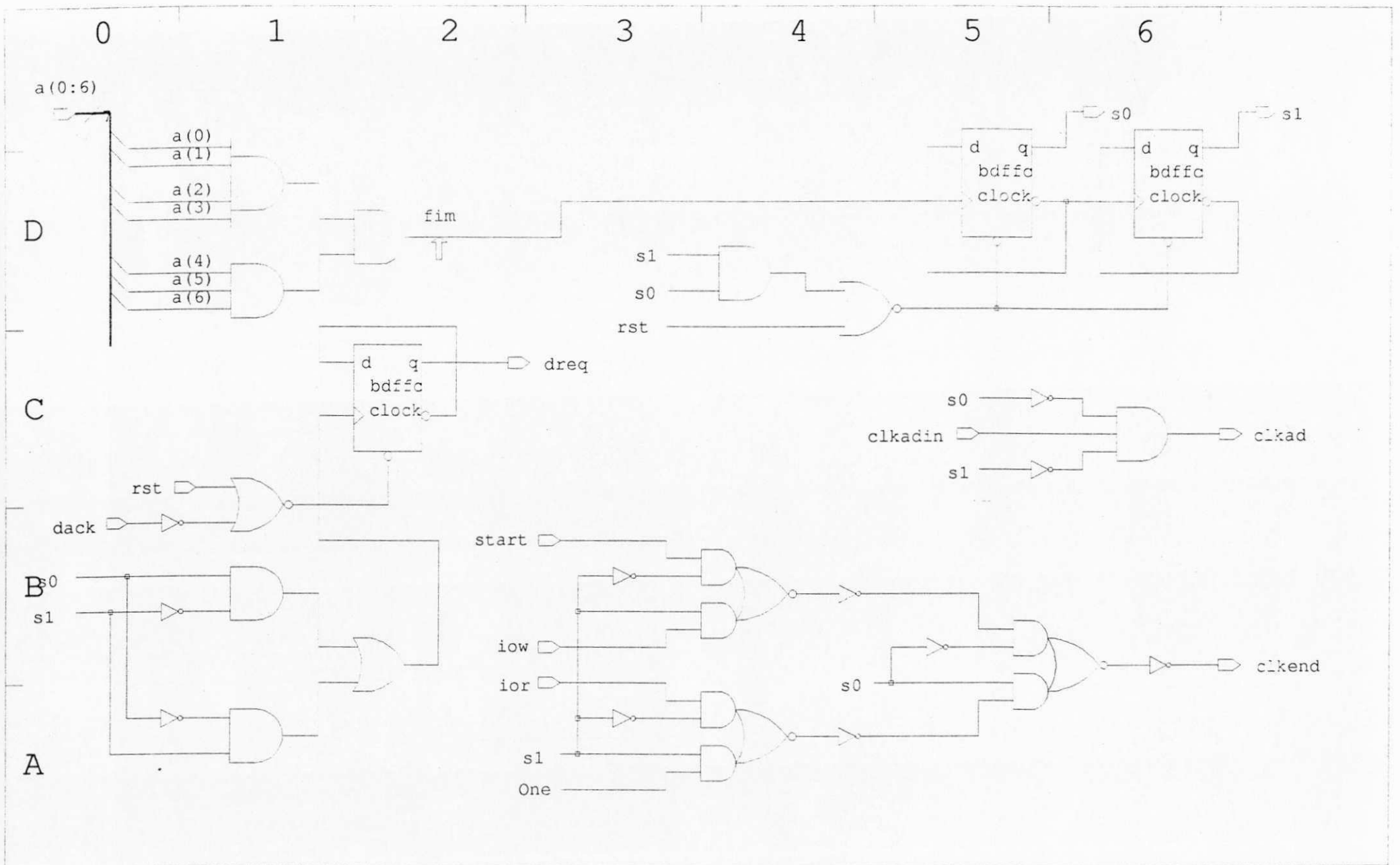
ESQUEMAS DOS CIRCUITOS ELETRICOS

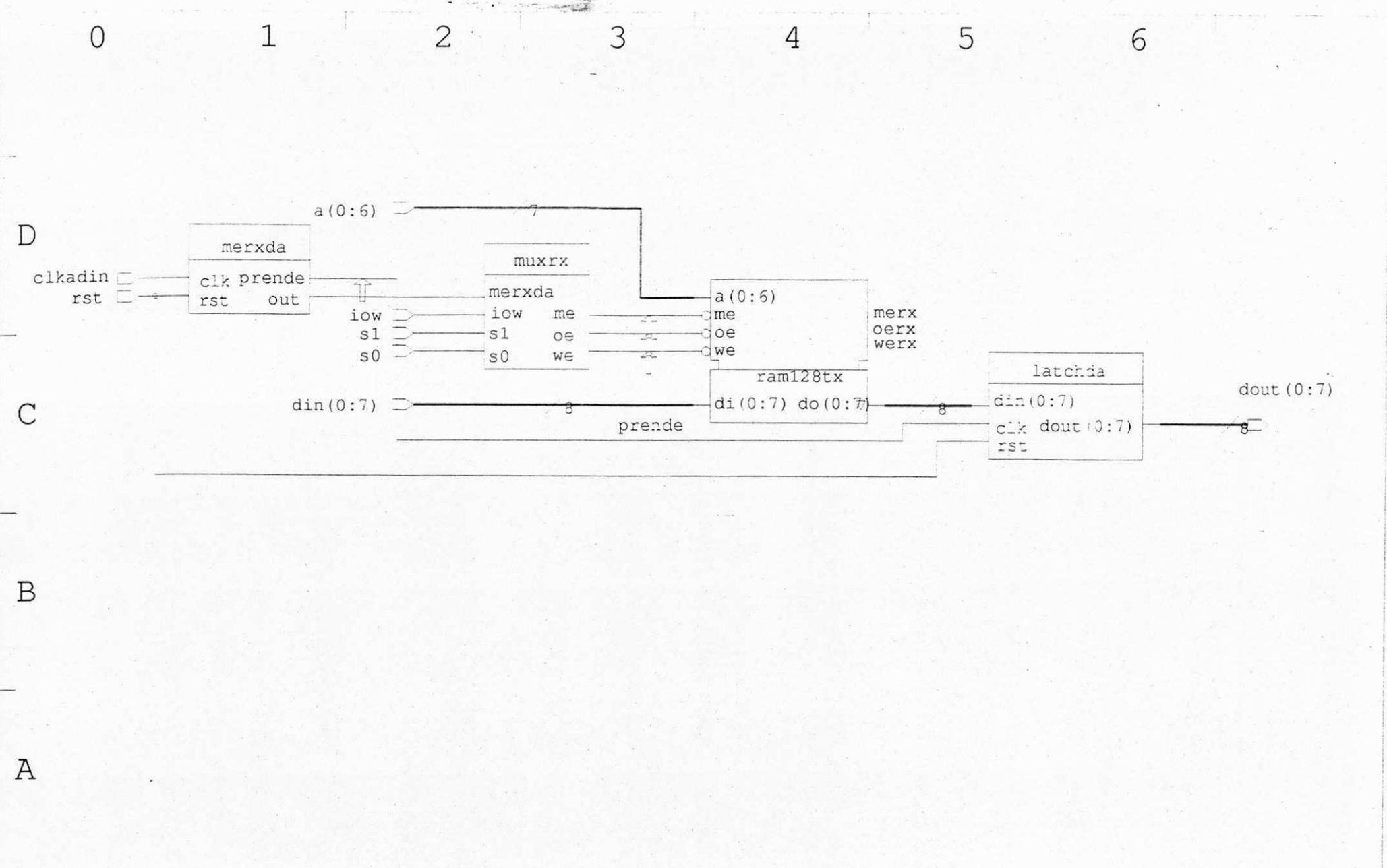


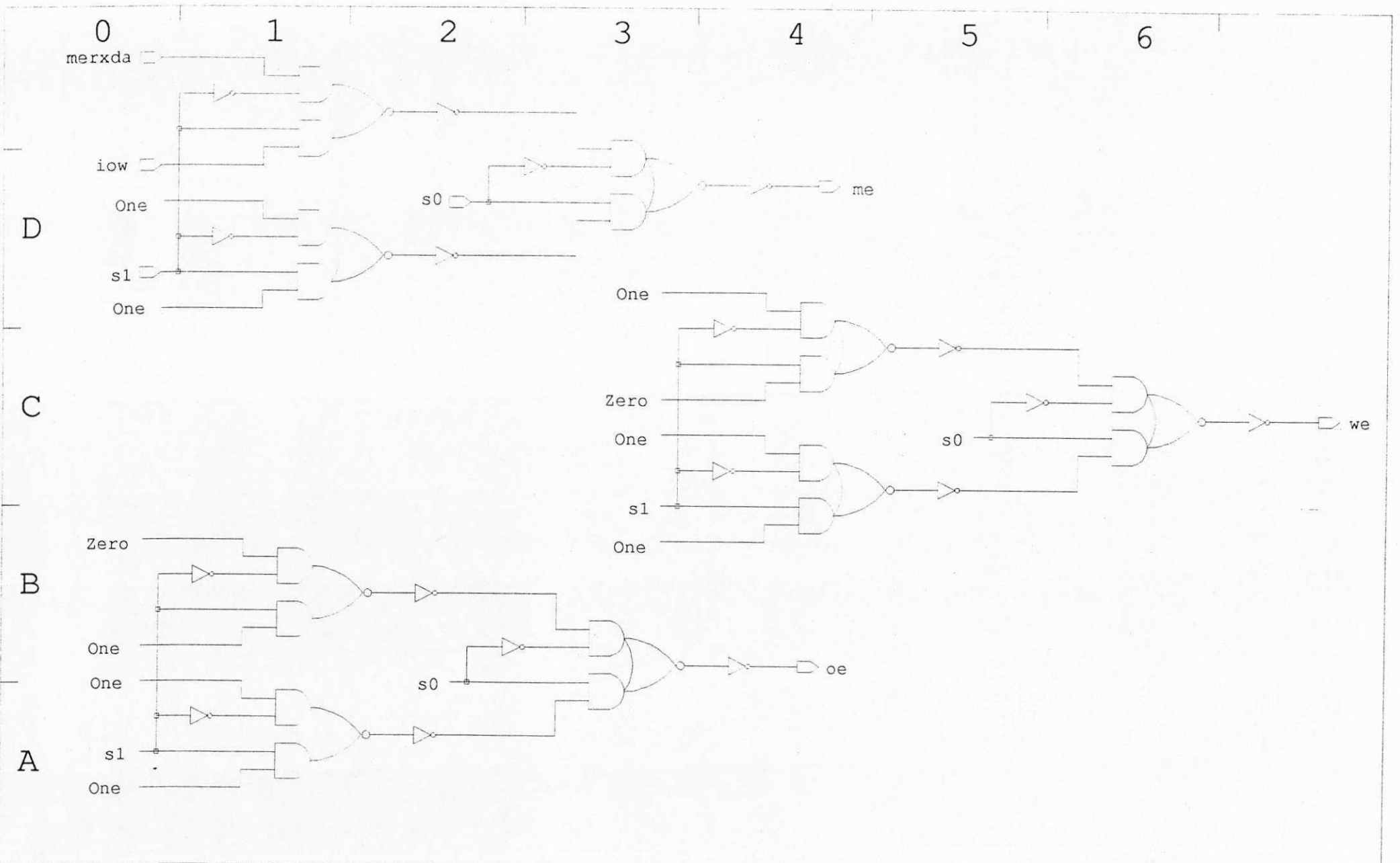
ES2: Solo version 3.1.2 Draft version 6.4.0	Designer	Alexandre
Sheet s1 of part cppv (1 sheet)	Date	Fri Nov 8 1991

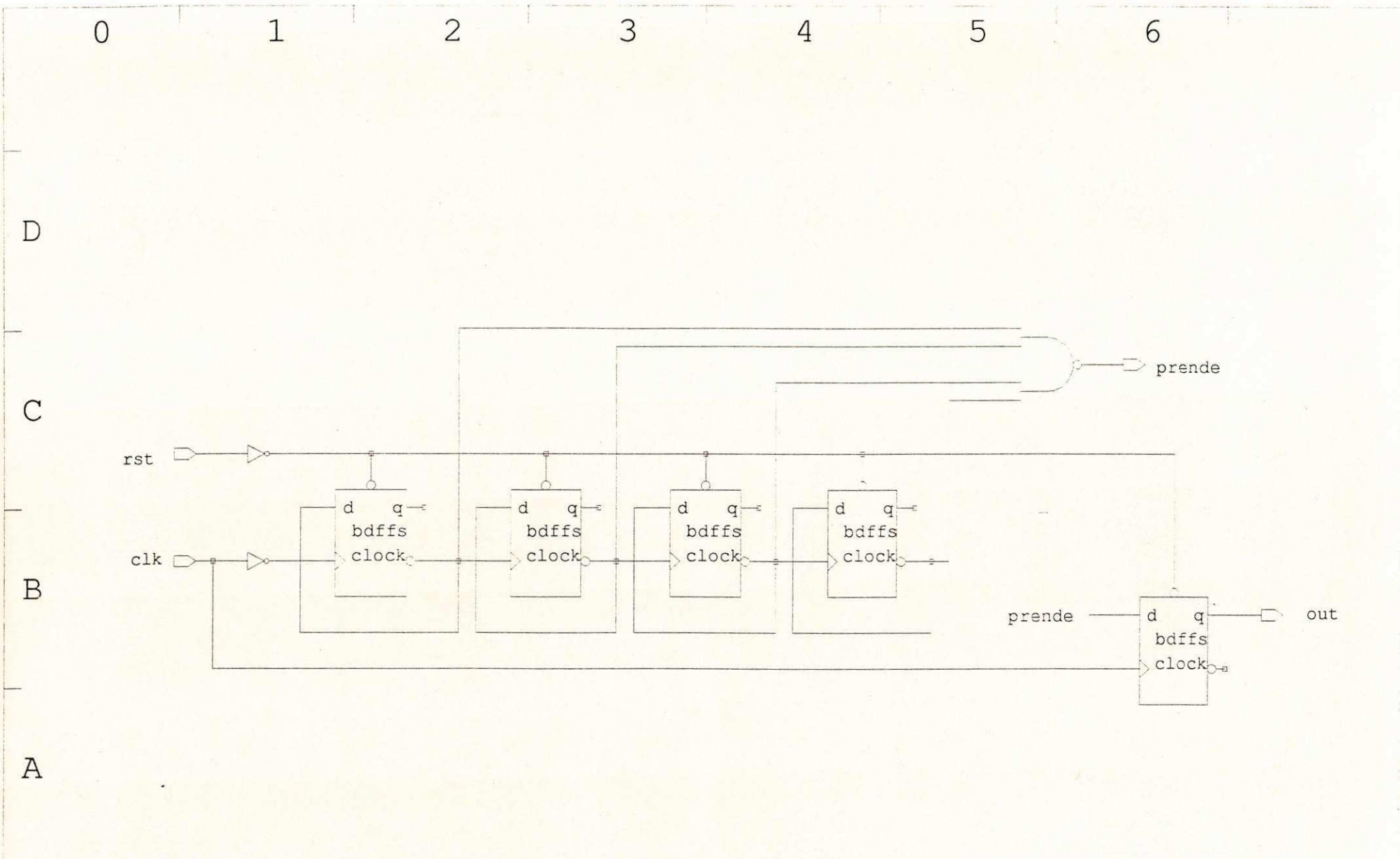


ES2: Solo version 3.1.2 Draft version 6.4.0		Designer	Alexandre
Sheet s1 of part gerall (1 sheet)		Date	Fri Nov 8 1991



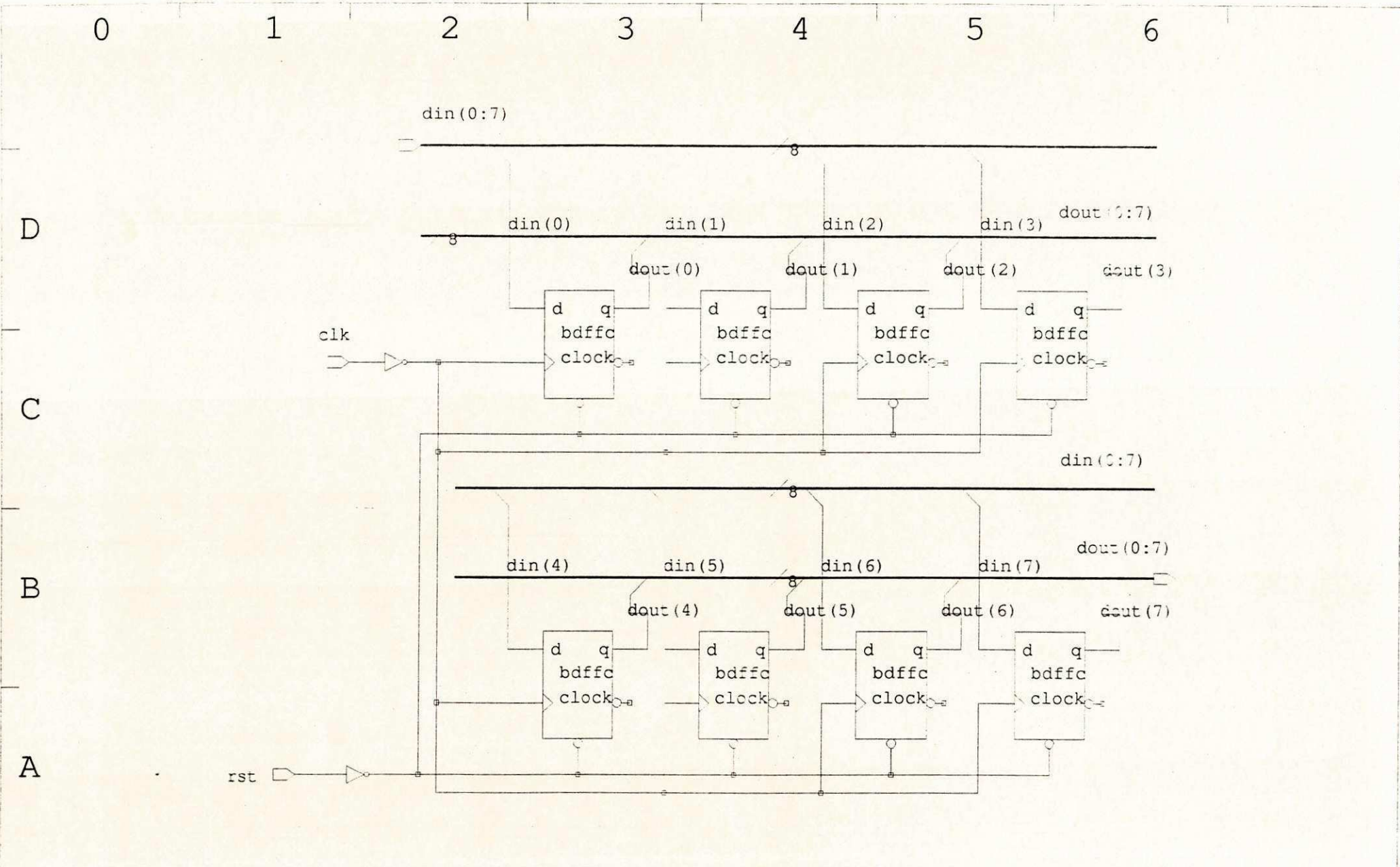




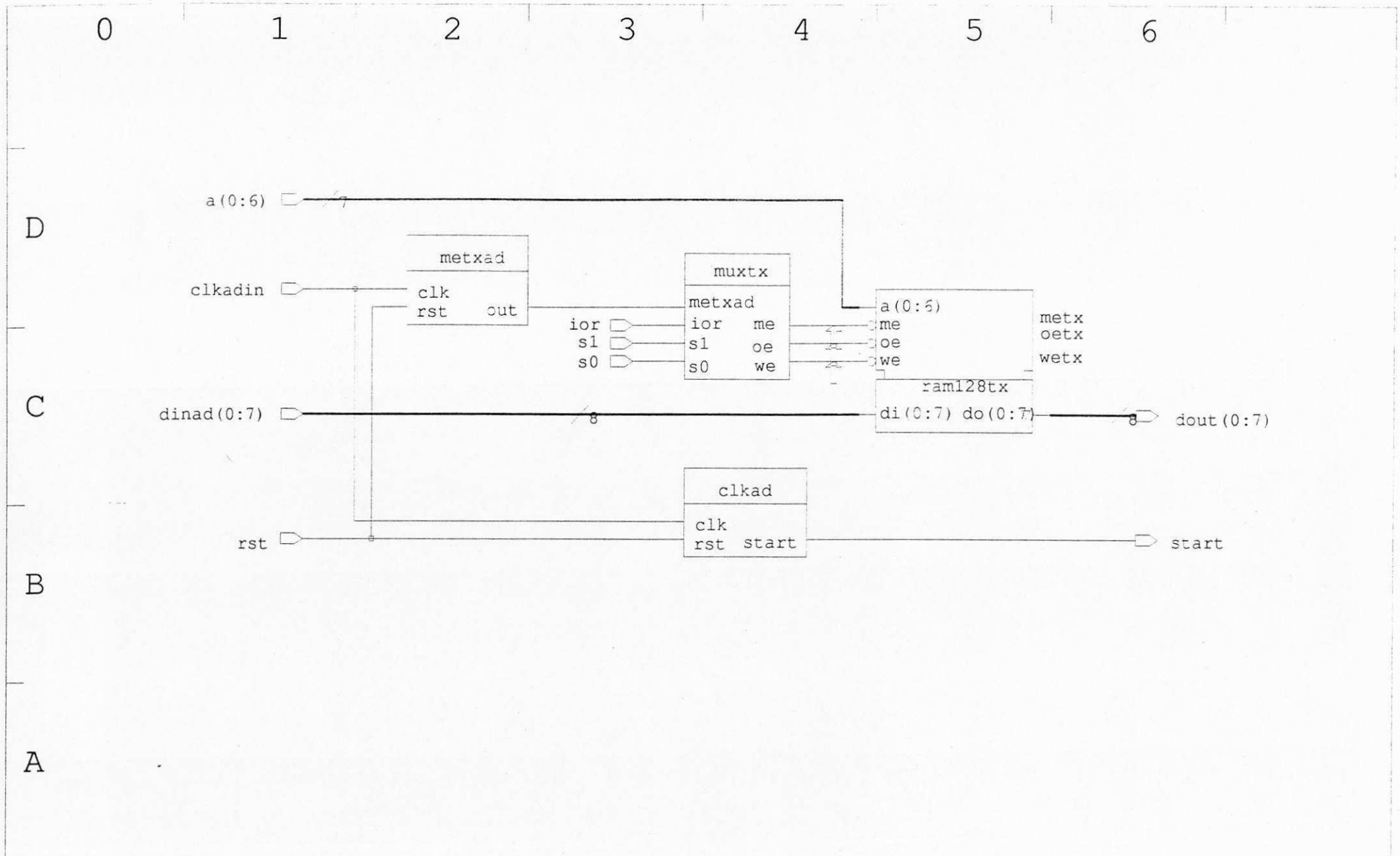


ES2: Solo version 3.1.2 Draft version 6.4.0	Designer Alexandre
Sheet s1 of part merxda (1 sheet)	Date Fri Nov 8 1991

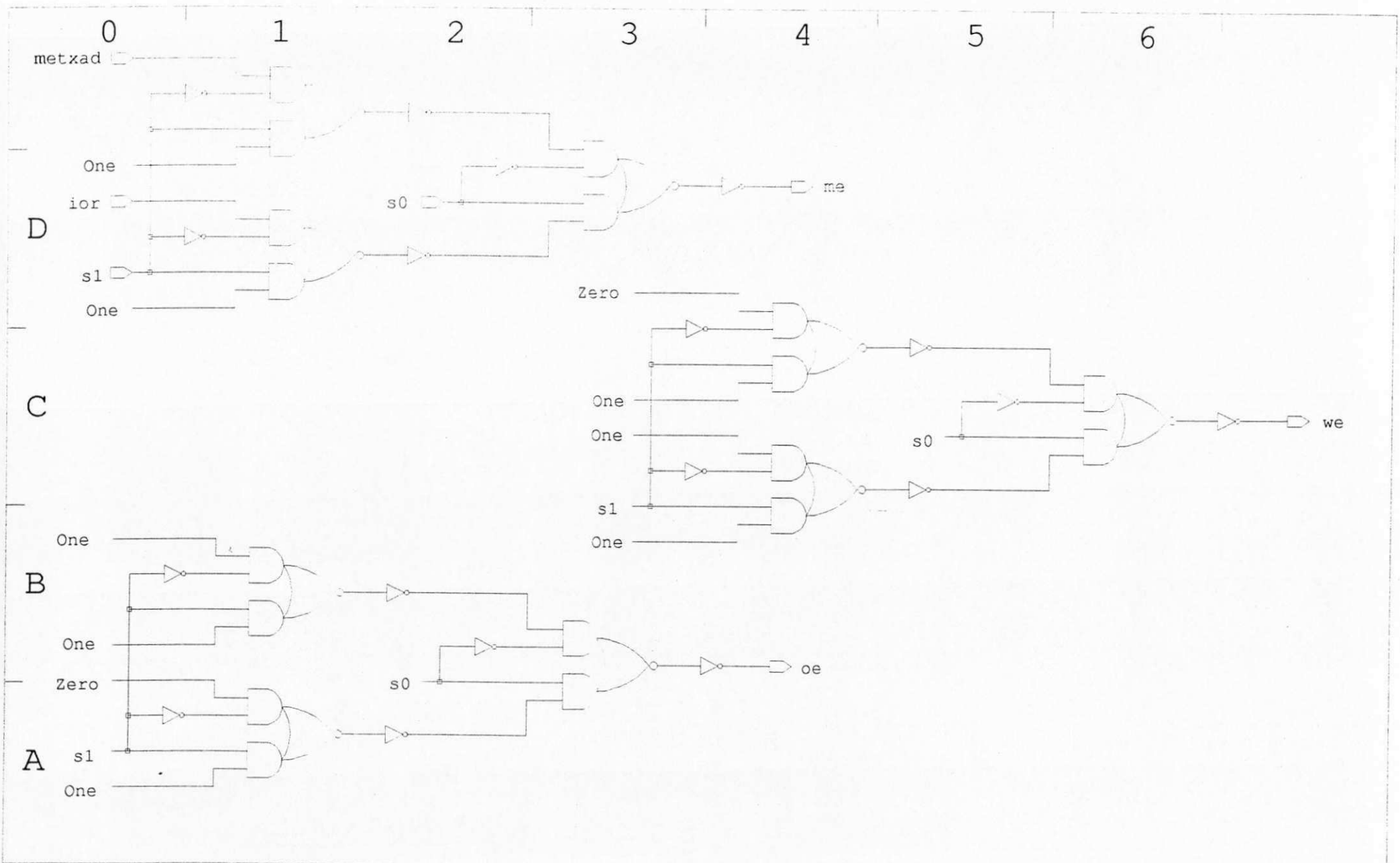
24



ES2: Solo version 3.1.2 Draft version 6.4.0	Designer	Alexandre
Sheet s1 of part latchda (1 sheet)	Date	Fri Nov 8 1991



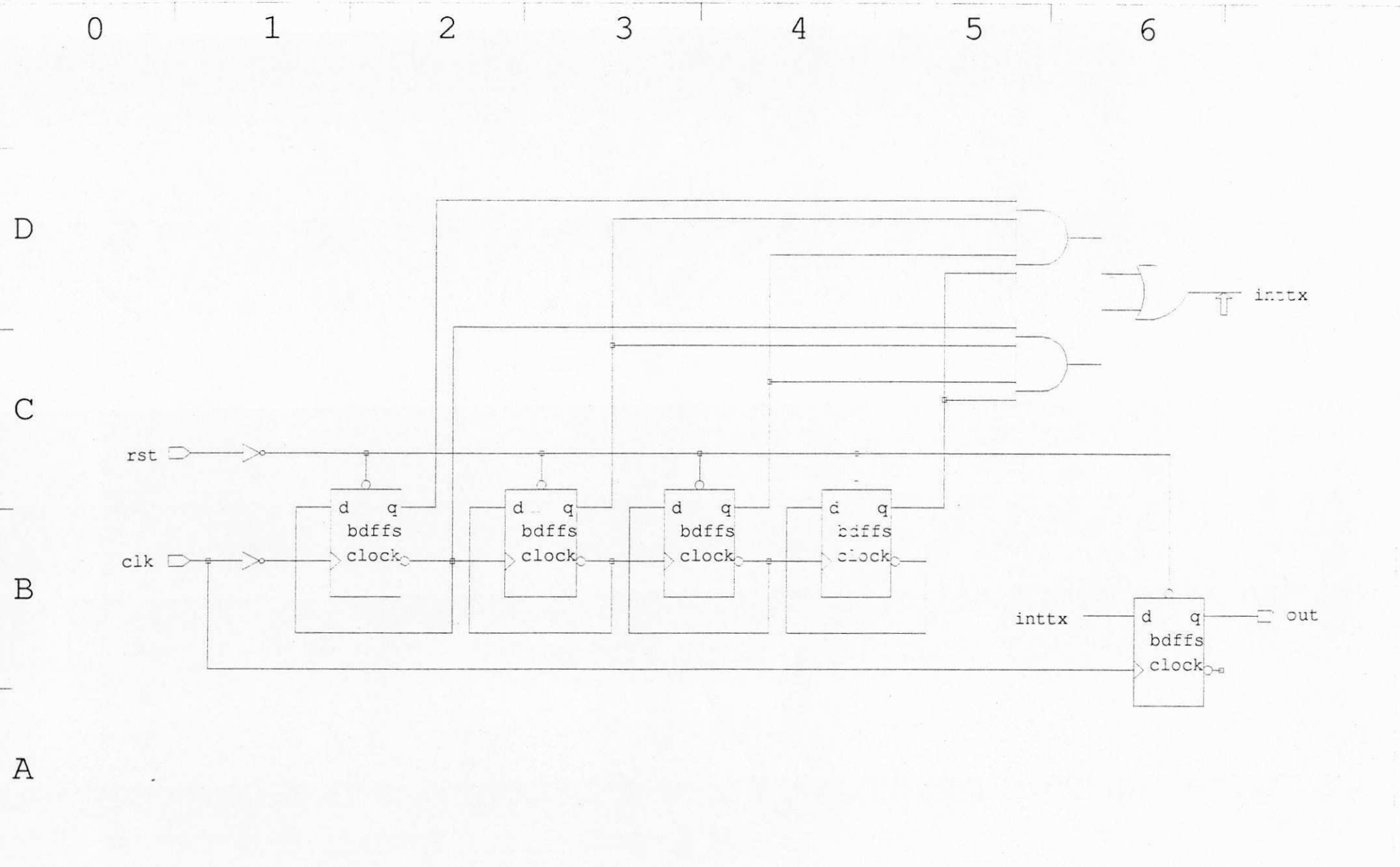
ES2: Solo version 3.1.2 Draft version 6.4.0	Designer	Alexandre
Sheet s1 of part tx (1 sheet)	Date	Fri Nov 8 1991



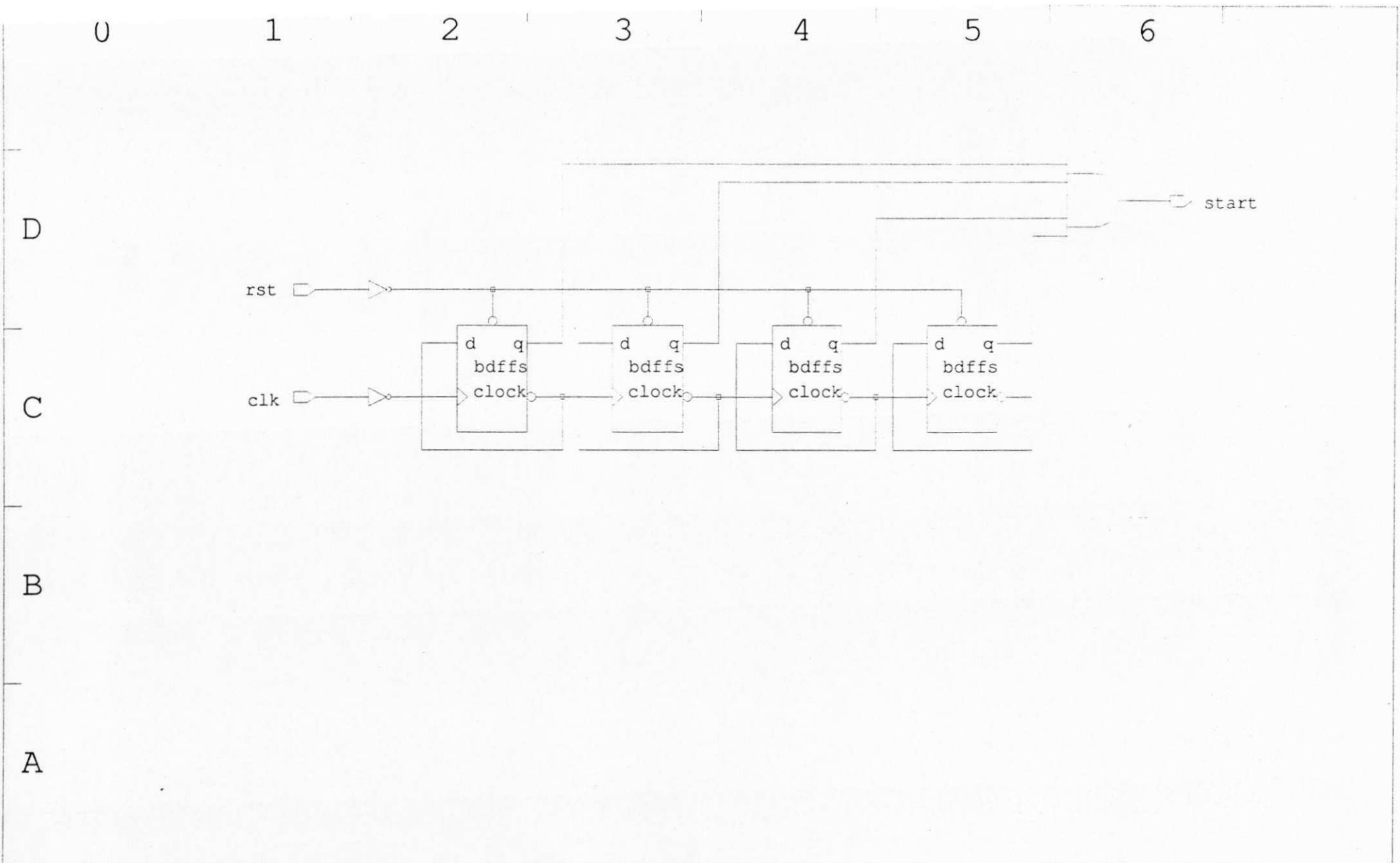
ES2: Solo version 3.1.2 Draft version 6.4.0

Sheet s1 of part muxtx (1 sheet)

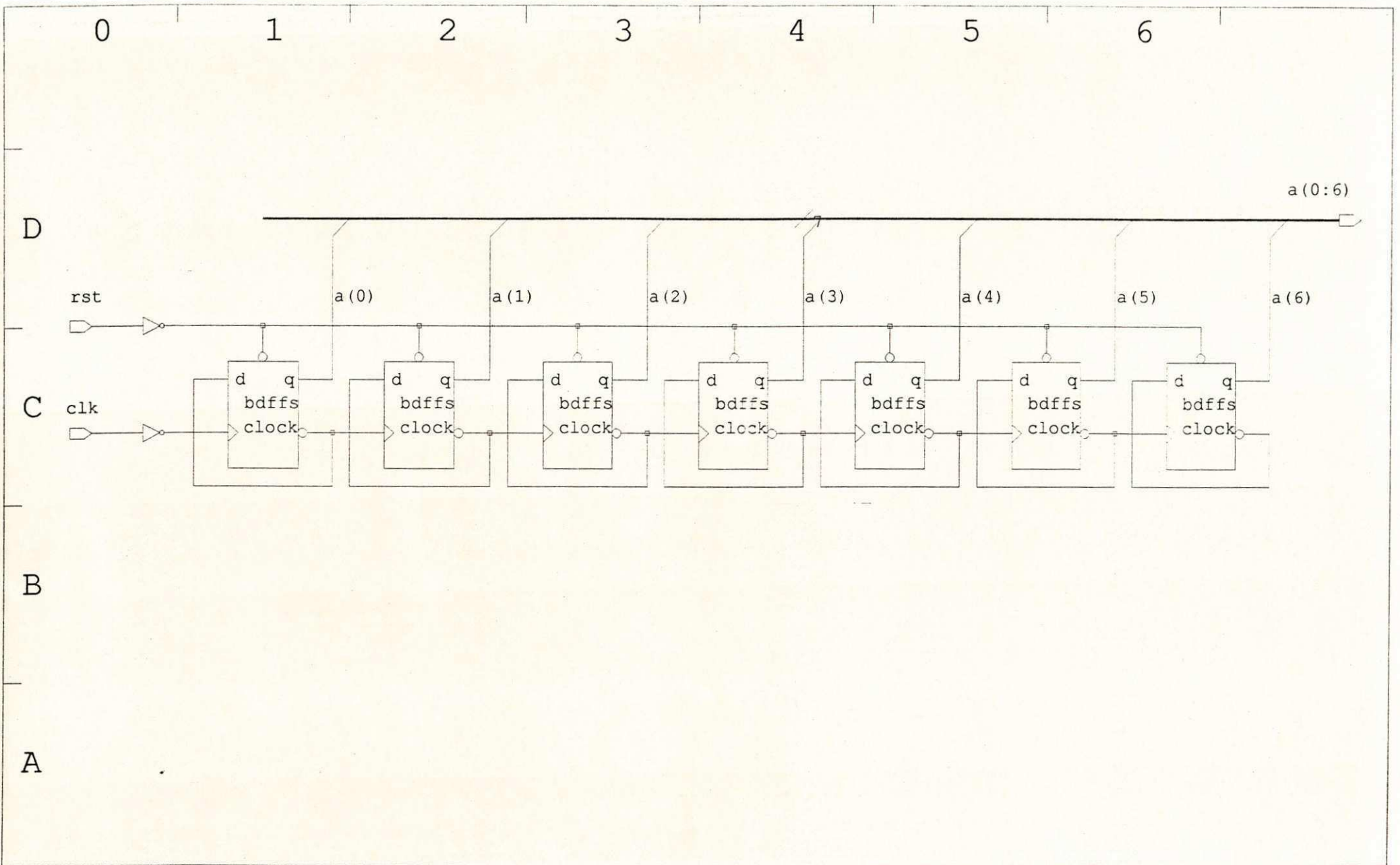
Designer	Alexandre
Date	Fri Nov 8 1991



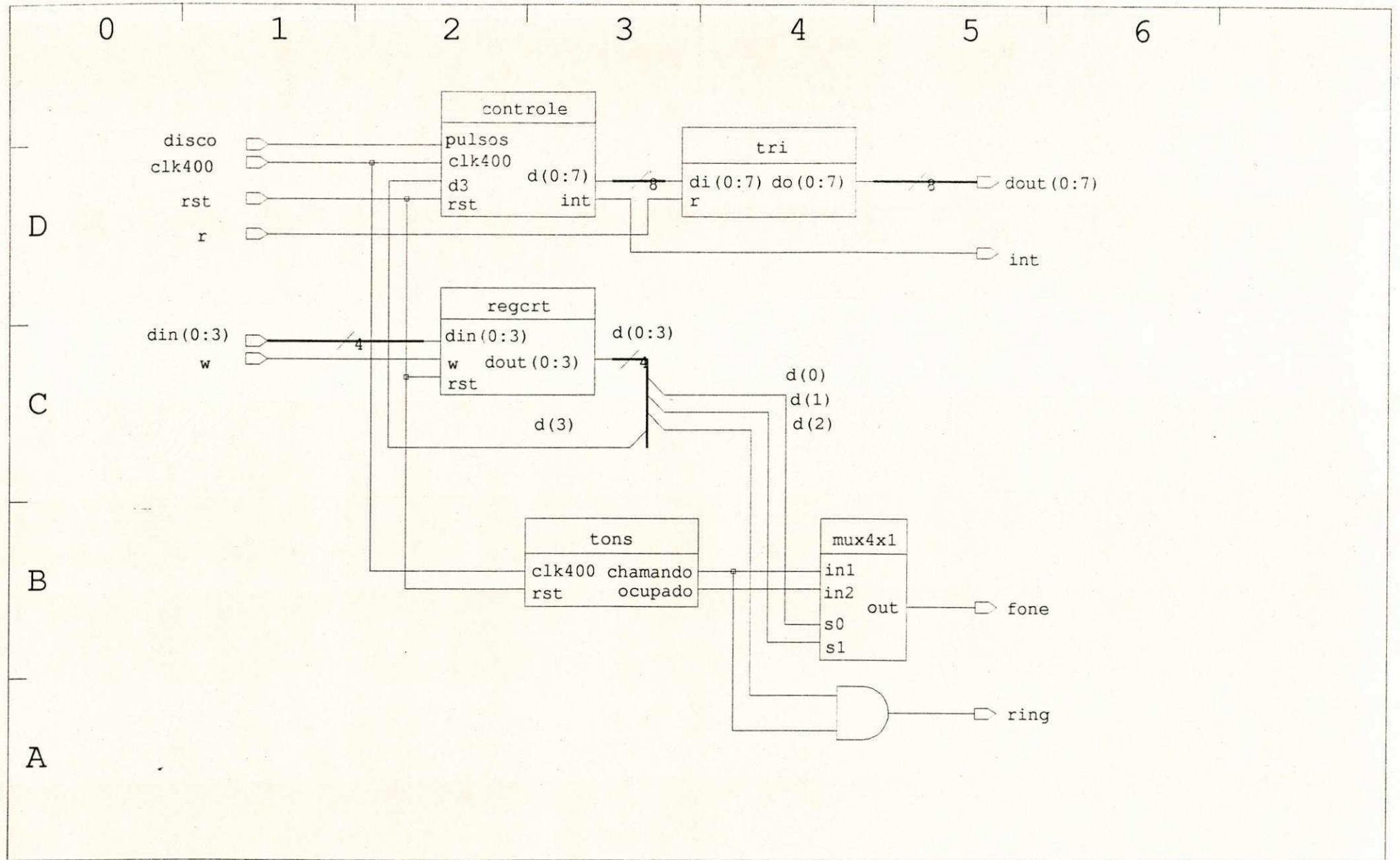
ES2: Solo version 3.1.2 Draft version 6.4.0	Designer Alexandre
Sheet s1 of part metxad (1 sheet)	Date Fri Nov 8 1991



ES2: Solo version 3.1.2 Draft version 6.4.0	Designer	Alexandre
Sheet s1 of part clkad (1 sheet)	Date	Fri Nov 8 1991



ES2: Solo version 3.1.2 Draft version 6.4.0	Designer	Alexandre
Sheet s1 of part geraend (1 sheet)	Date	Fri Nov 8 1991



ES2: Solo version 3.1.2 Draft version 6.4.0		Designer	Alexandre
Sheet s1 of part geral0 (1 sheet)		Date	Fri Nov 8 1991

77

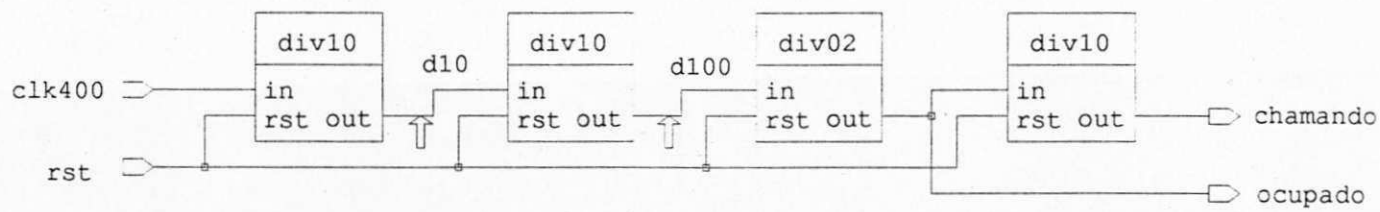
D

C

B

A

0 1 2 3 4 5 6



ES2: Solo version 3.1.2 Draft version 6.4.0

Designer Alexandre

Sheet s1 of part tons (1 sheet)

Date

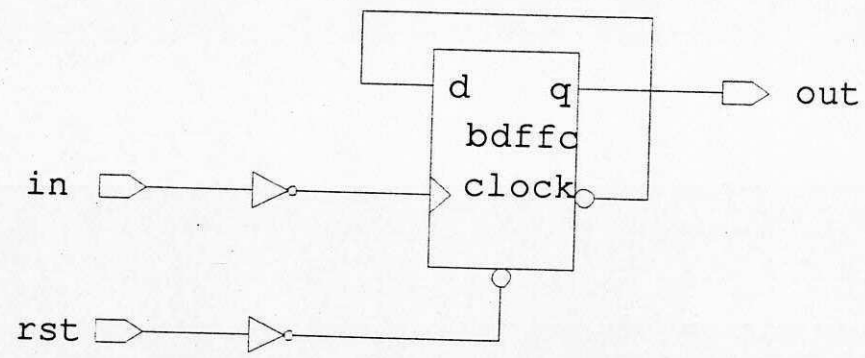
Fri Nov 8 1991

0 1 2 3 4

C

B

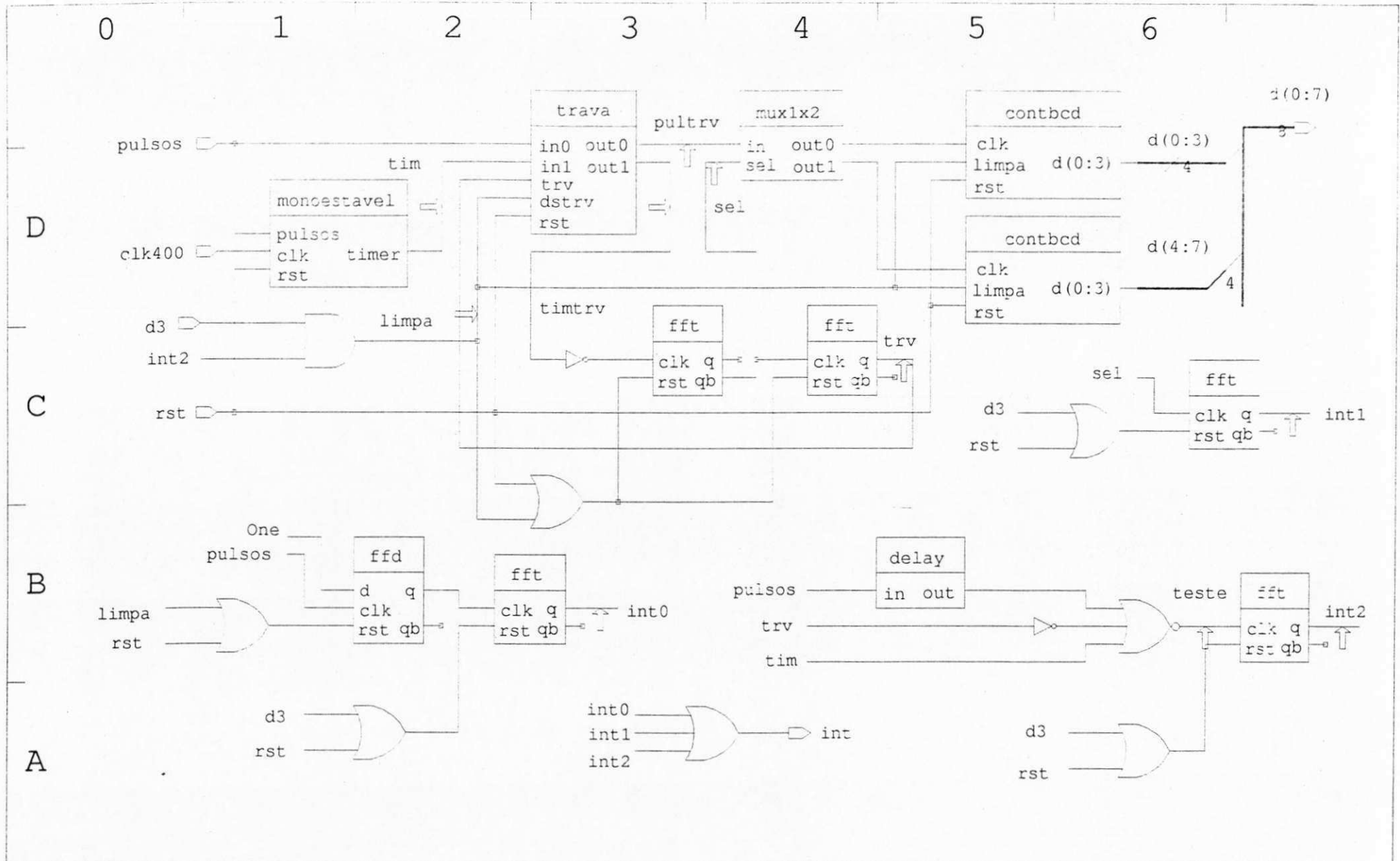
A



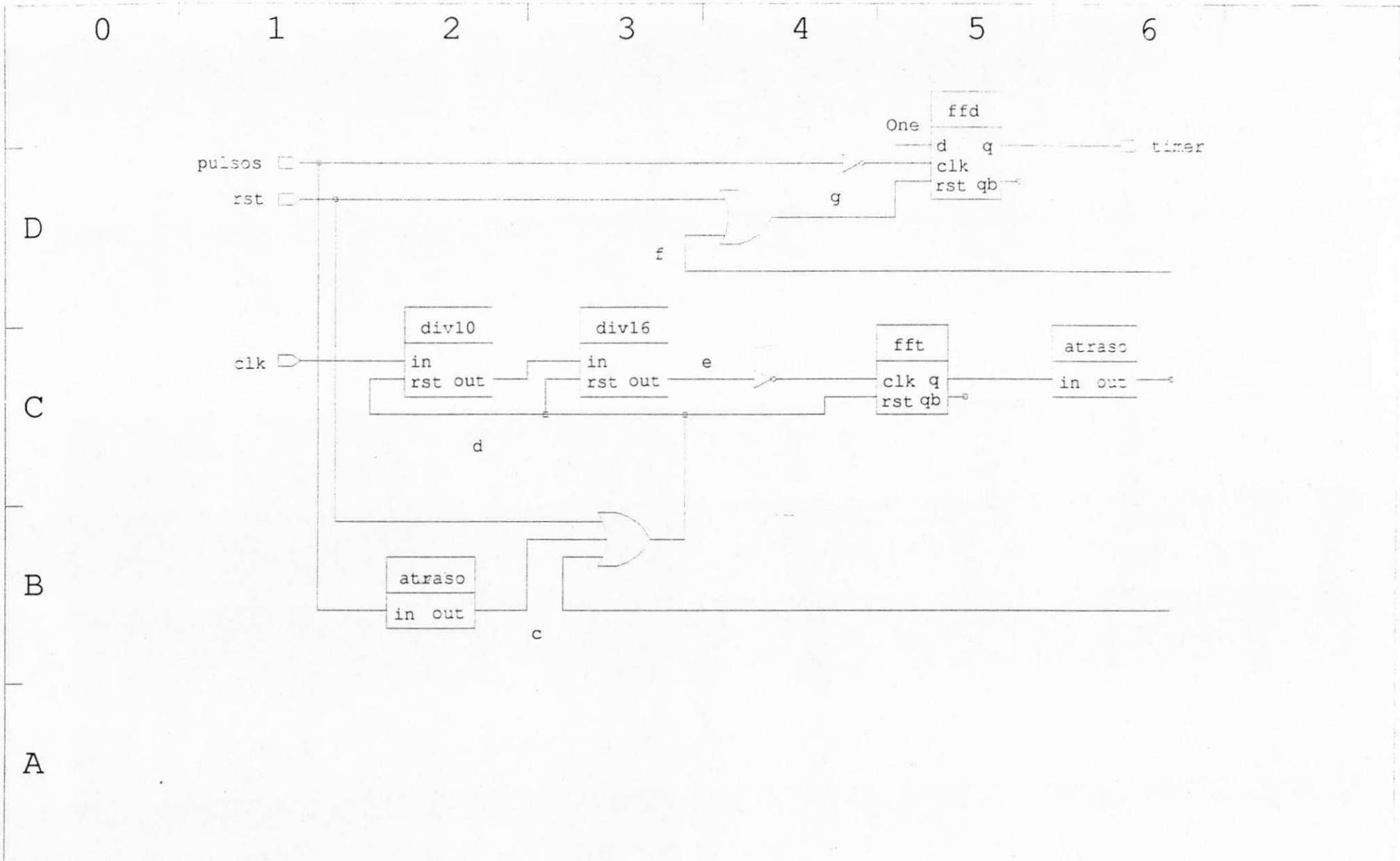
ES2: Solo version 3.1.2 Draft version 6.4.0

Sheet s1 of part div02 (1 sheet)

Designer	Alexandre
Date	Fri Nov 8 1991



ES2: Solo version 3.1.2 Draft version 6.4.0		Designer	Alexandre
Sheet s1 of part controle (1 sheet)		Date	Fri Nov 8 1991



ES2: Solo version 3.1.2 Draft version 6.4.0	Designer	Alexandre
Sheet s1 of part monoestavel (2 sheet)	Date	Fri Nov 8 1991

0

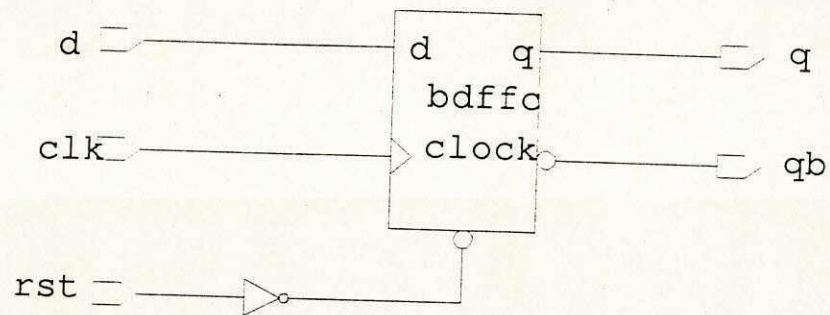
1

2

3

4

C



B

A

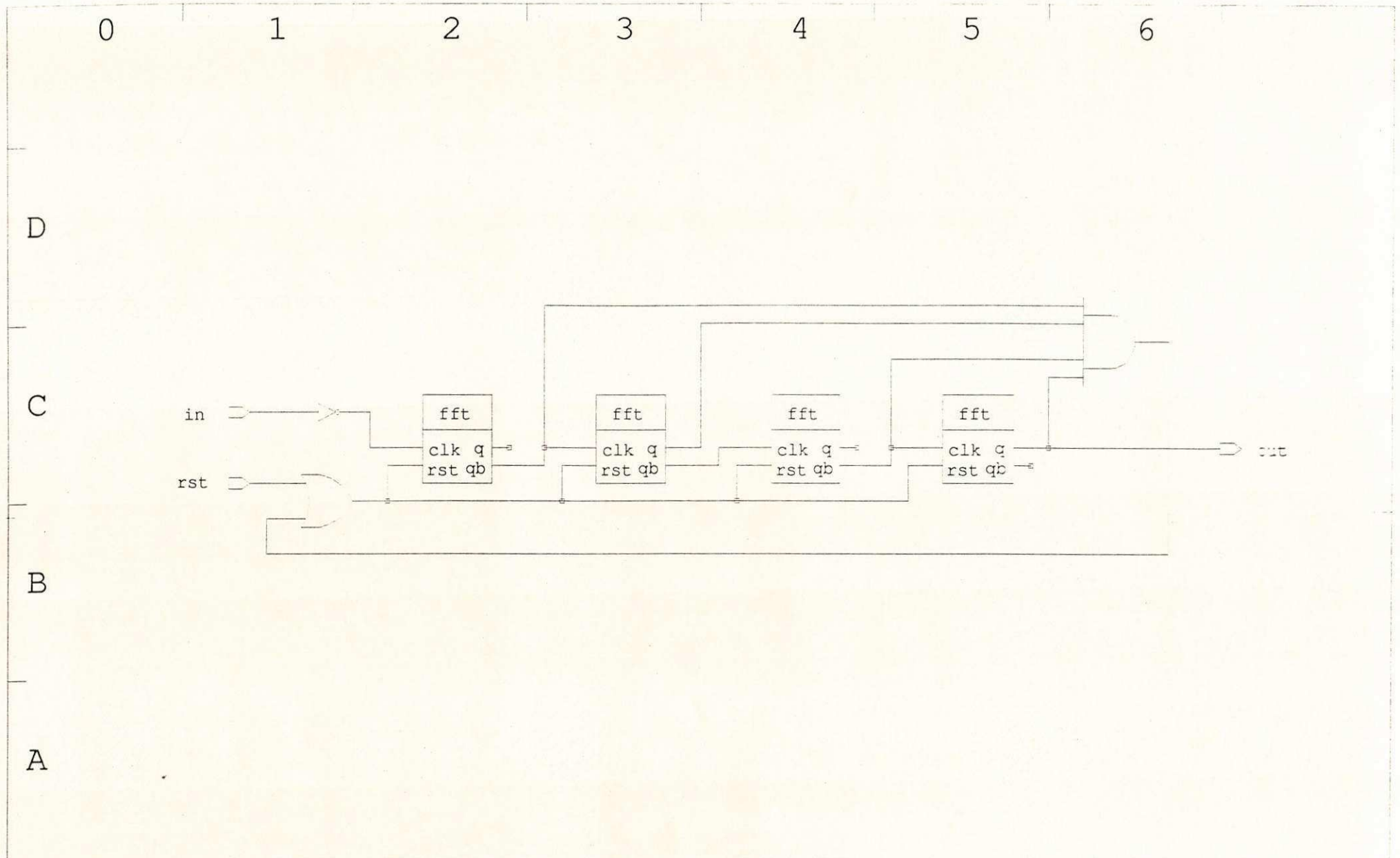
81

ES2: Solo version 3.1.2 Draft version 6.4.0

Sheet s1 of part ffd (1 sheet)

Designer Alexandre

Date Fri Nov 8 1991



ES2: Solo version 3.1.2 Draft version 6.4.0

Designer Alexandre

Sheet s1 of part div10 (1 sheet)

Date Fri Nov 8 1991

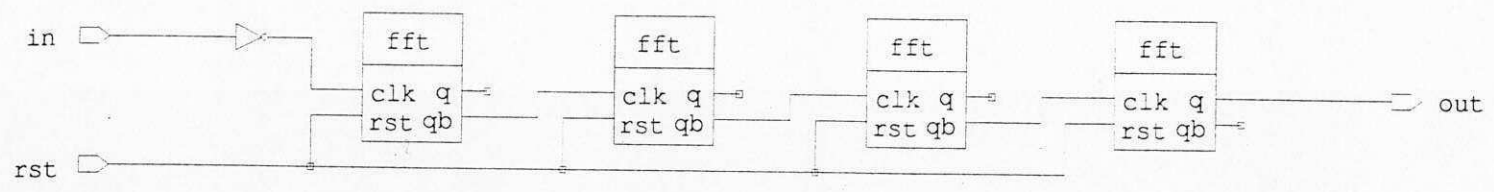
0 1 2 3 4 5 6

D

C

B

A



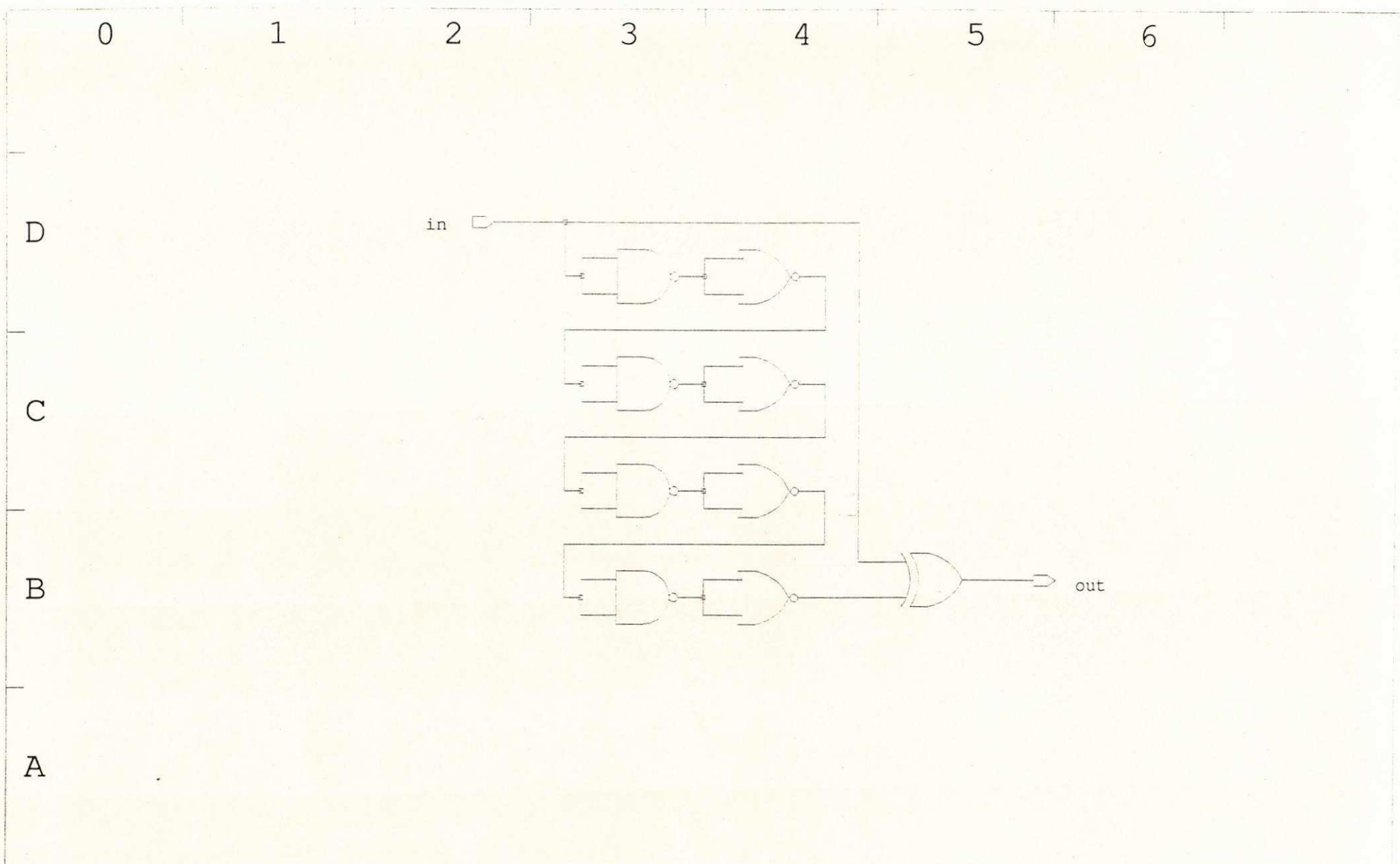
83

ES2: Solo version 3.1.2 Draft version 6.4.0

Sheet s1 of part div16 (1 sheet)

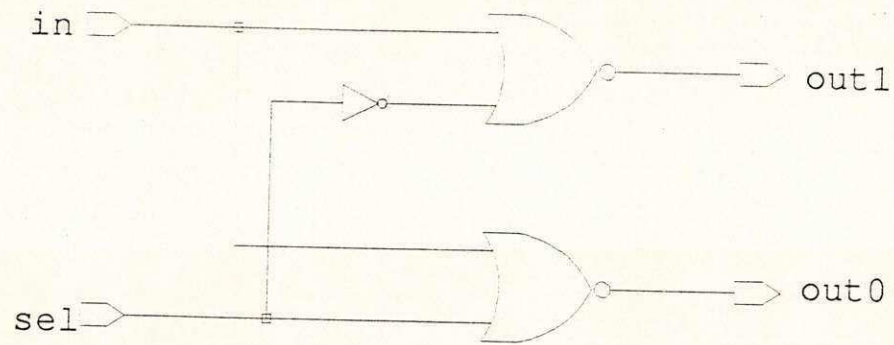
Designer Alexandre

Date Fri Nov 8 1991



ES2: Solo version 3.1.2 Draft version 6.4.0	Designer	Alexandre
Sheet s1 of part atraso (1 sheet)	Date	Fri Nov 8 1991

C



B

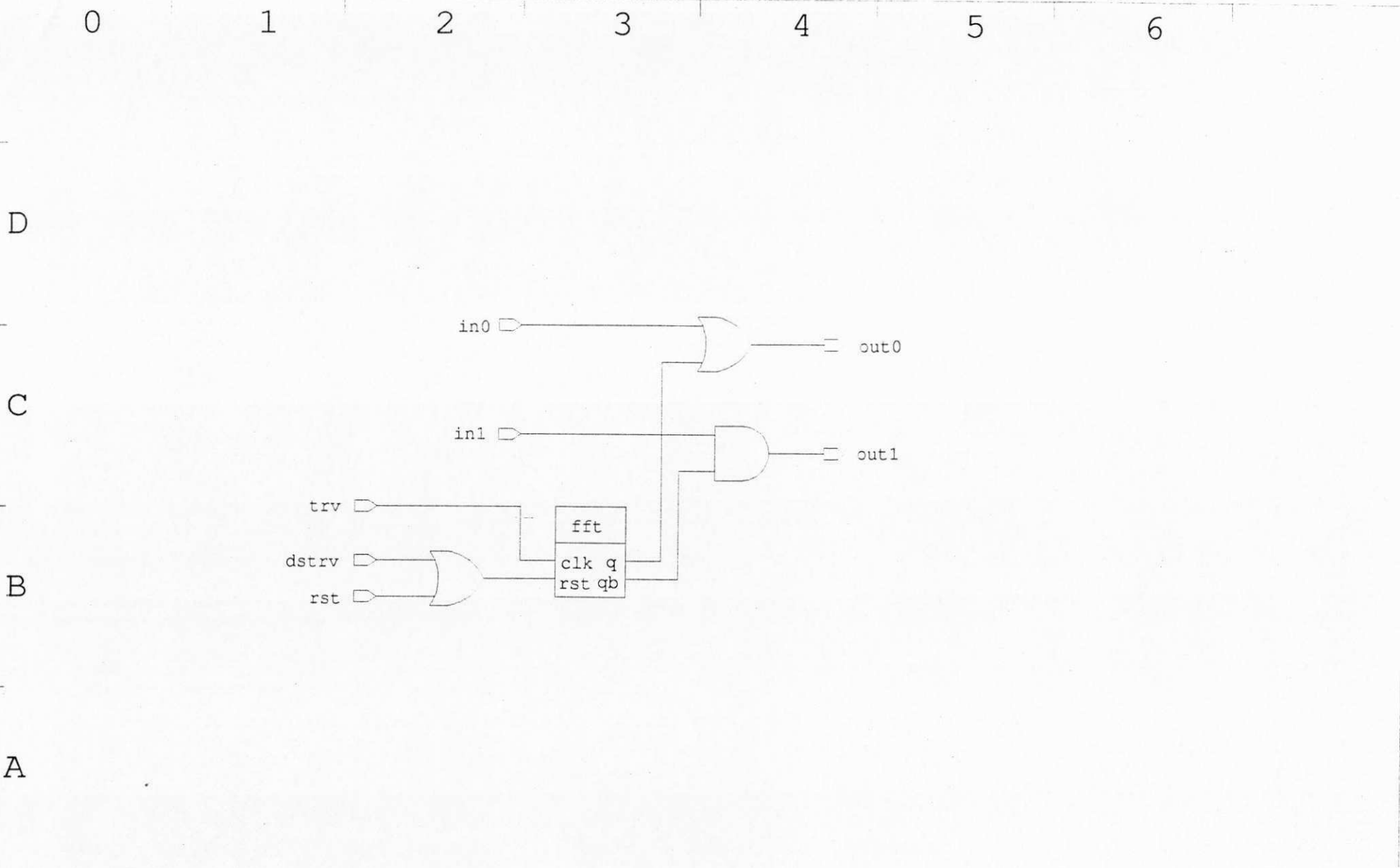
A

ES2: Solo version 3.1.2 Draft version 6.4.0

Sheet s1 of part mux1x2 (1 sheet)

Designer Alexandre

Date Fri Nov 8 1991

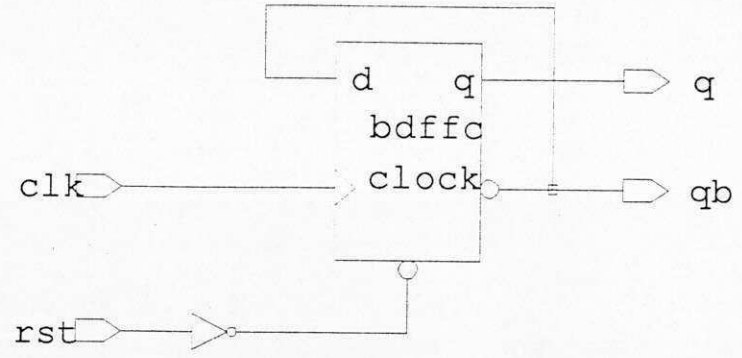


C

B

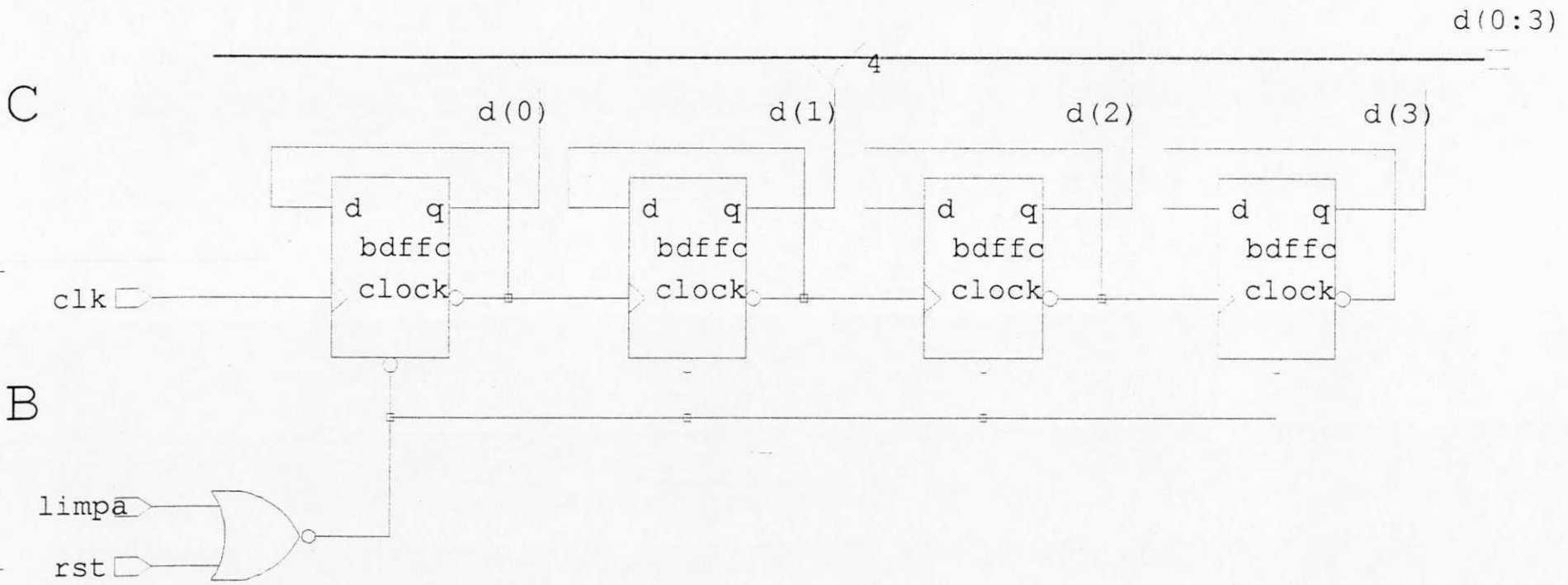
A

0 1 2 3 4

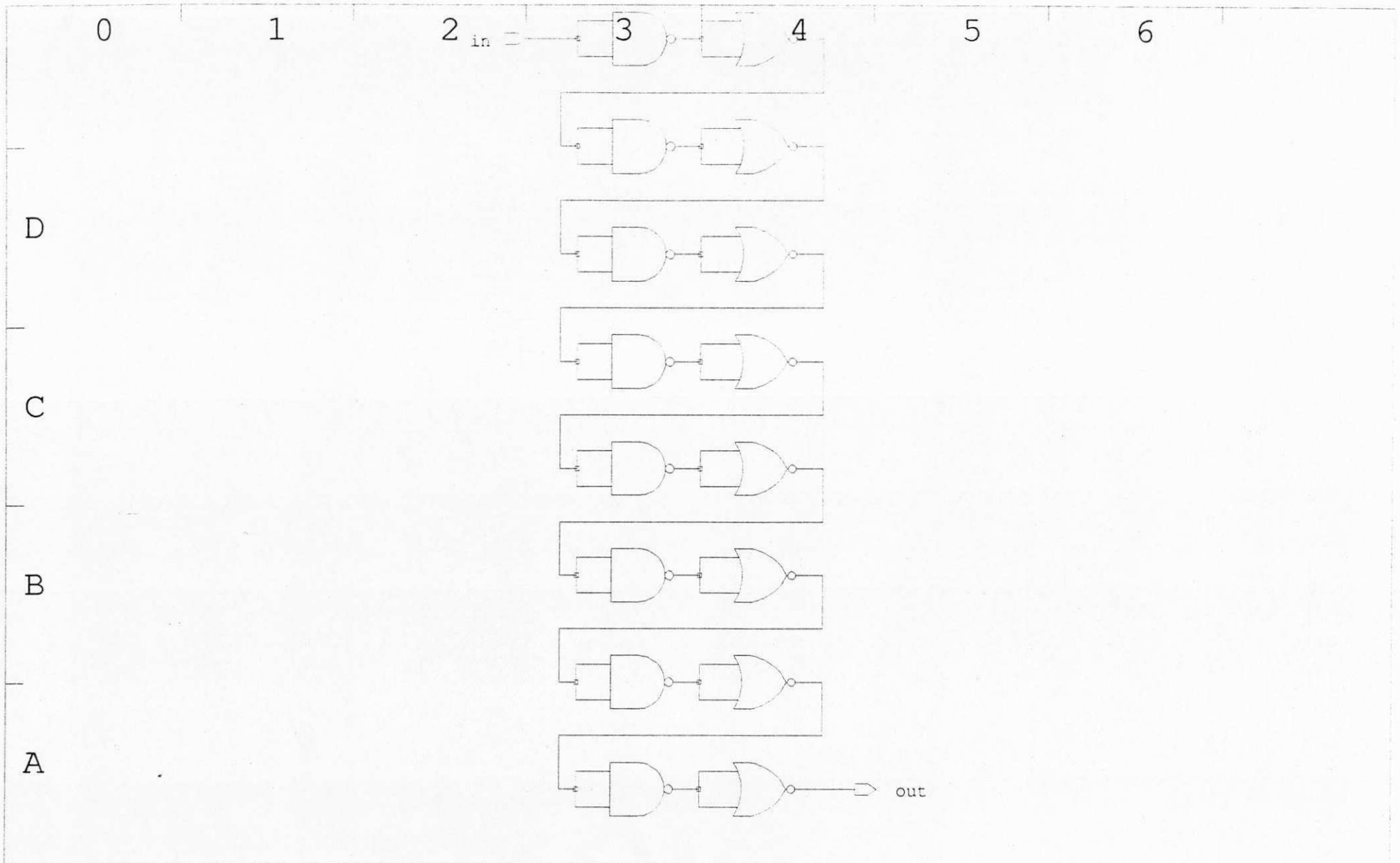


ES2: Solo version 3.1.2 Draft version 6.4.0	Designer	Alexandre
Sheet s1 of part fft (1 sheet)	Date	Fri Nov 8 1991

0 1 2 3 4



88



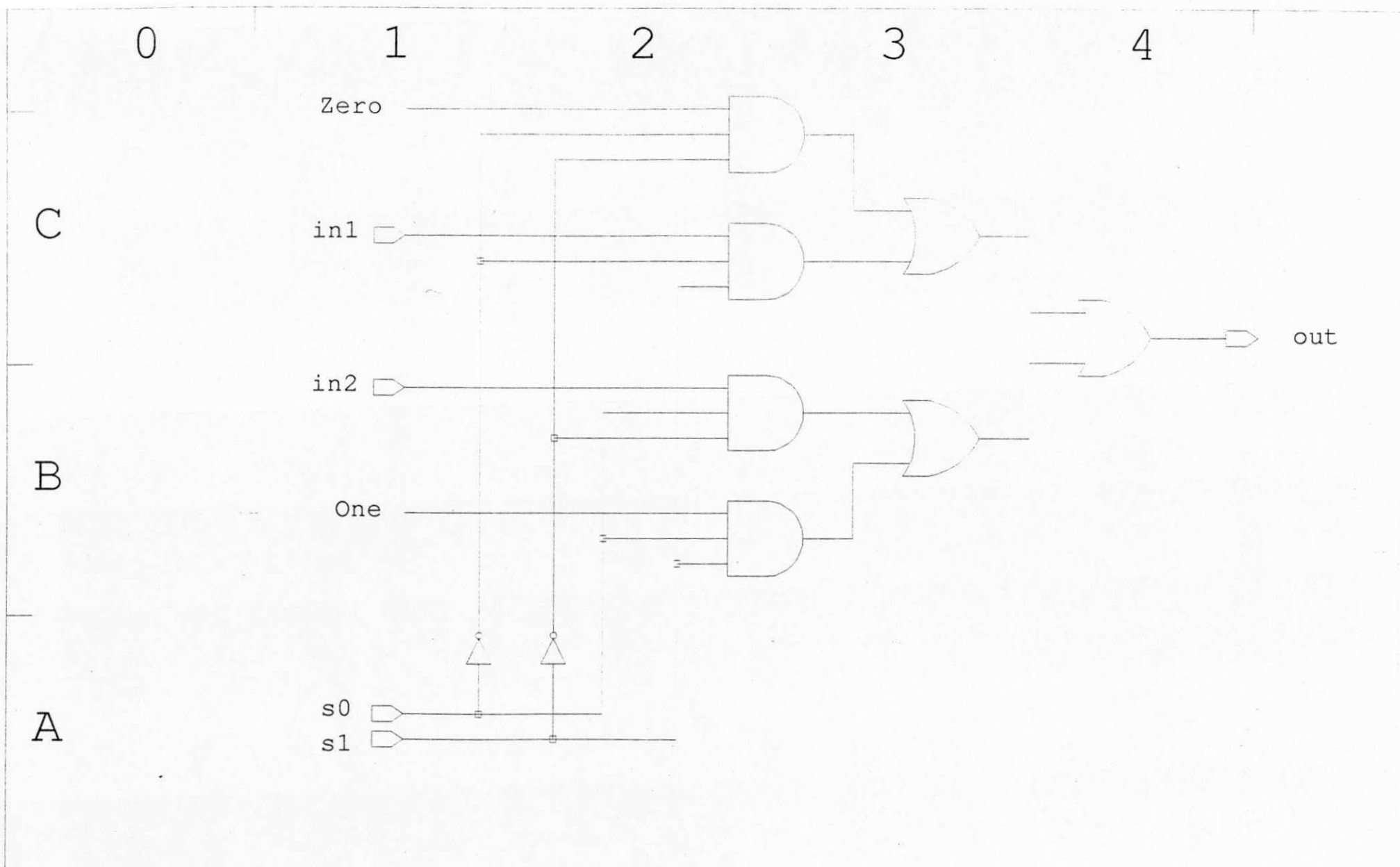
ES2: Solo version 3.1.2 Draft version 6.4.0

Designer Alexandre

Sheet s1 of part delay (1 sheet)

Date

Fri Nov 8 1991



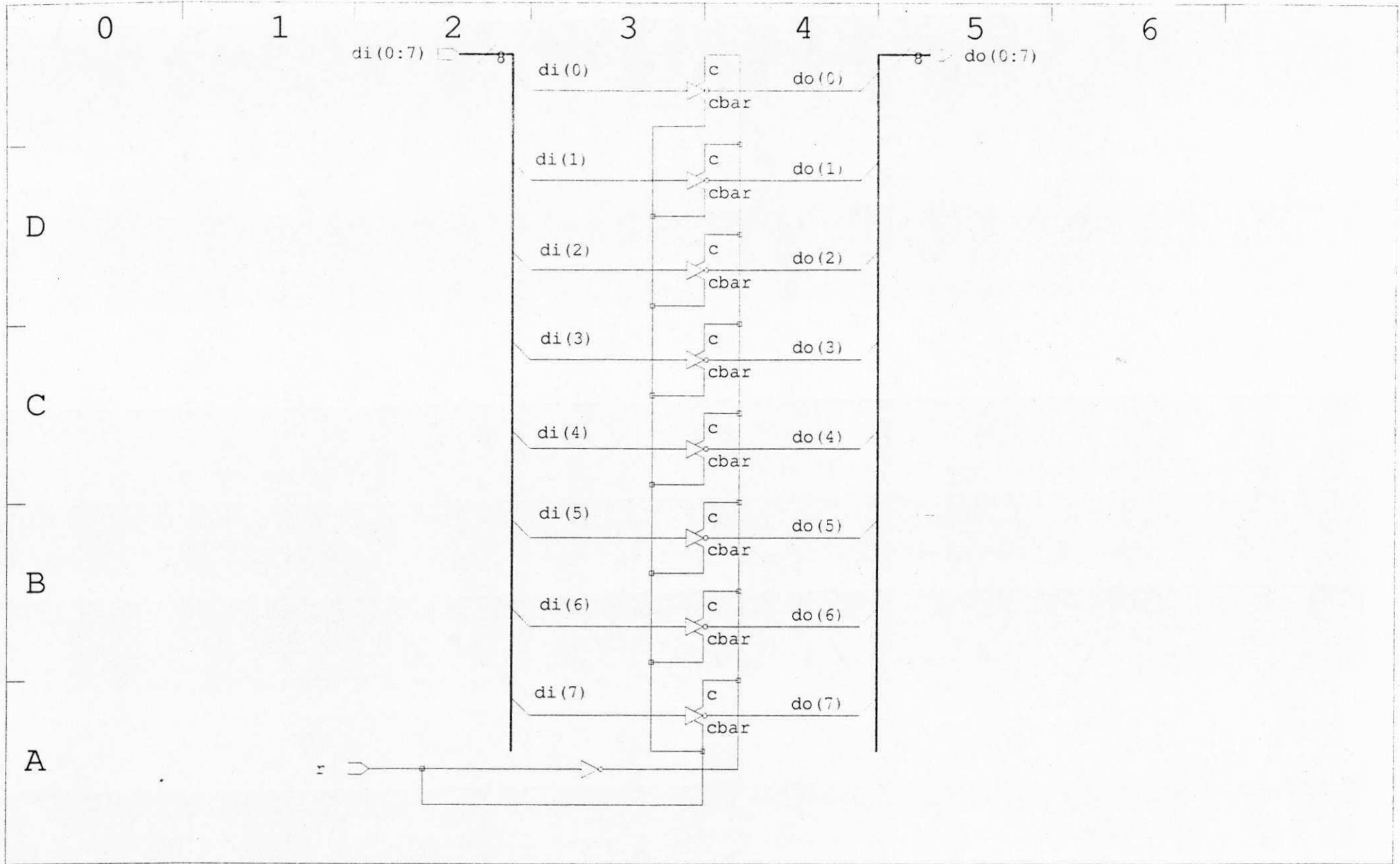
ES2: Solo version 3.1.2 Draft version 6.4.0

Designer Alexandre

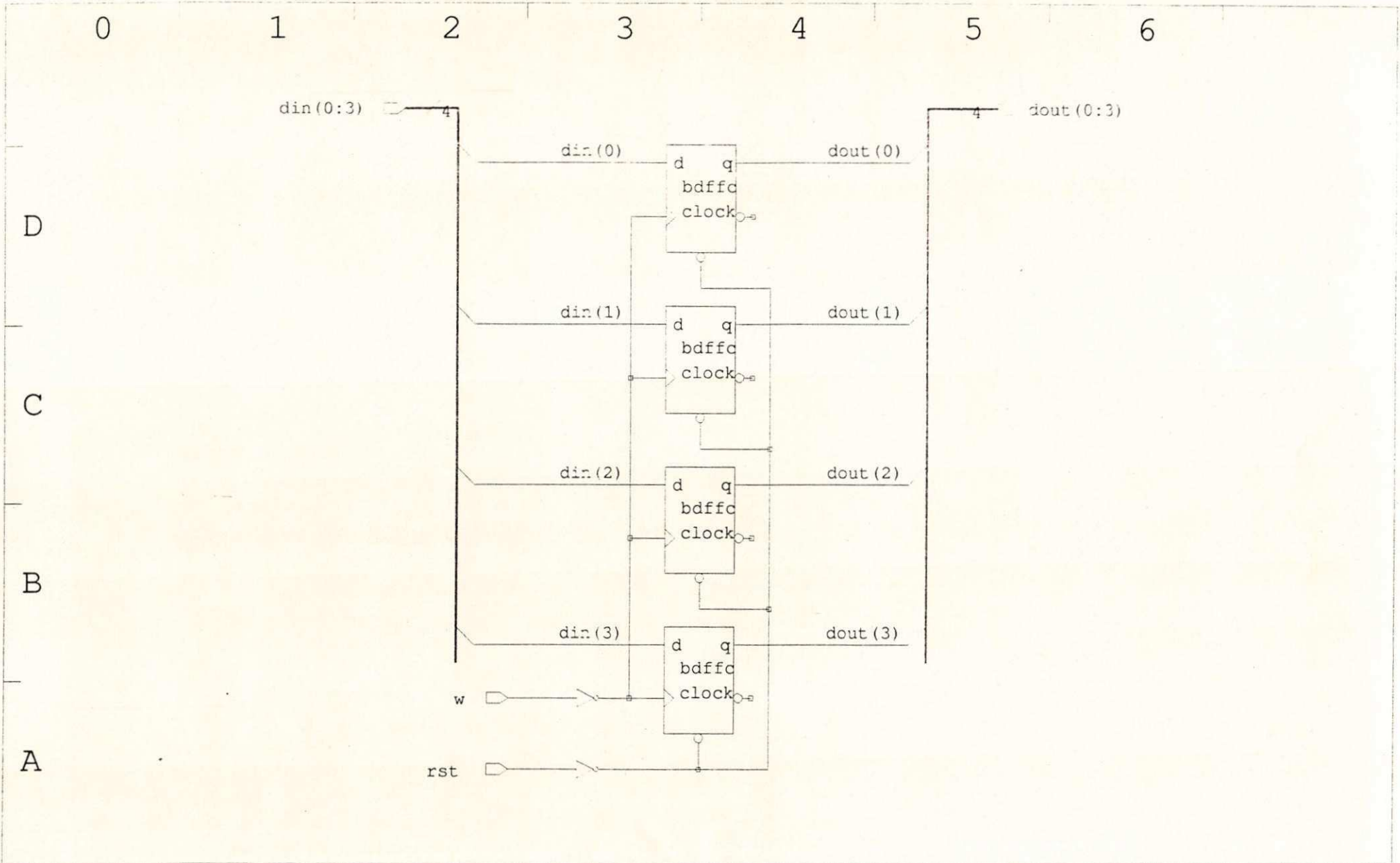
Sheet s1 of part mux4x1 (1 sheet)

Date Fri Nov 8 1991

91

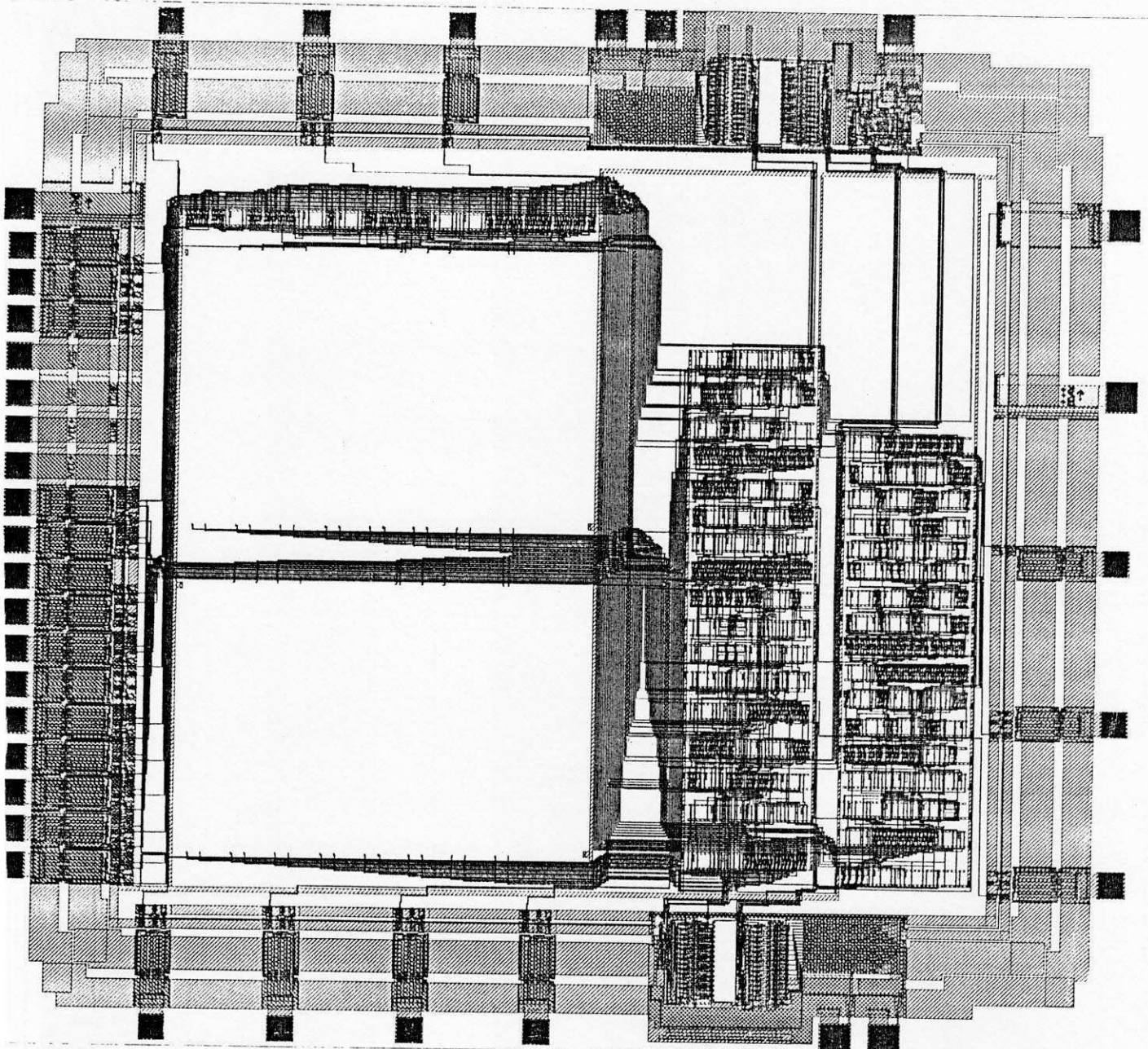


ES2: Solo version 3.1.2 Draft version 6.4.0	Designer	Alexandre
Sheet s1 of part tri (1 sheet)	Date	Fri Nov 8 1991

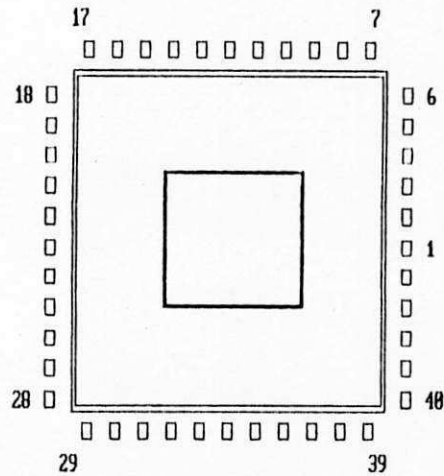


ES2: Solo version 3.1.2 Draft version 6.4.0		Designer	Alexandre
Sheet s1 of part regcrt (1 sheet)		Date	Fri Nov 8 1991

"LAYOUT" DO ASIC CPPV



PINAGEM DO ASIC CPPV



Cavidade onde repousa o "chip"

- | | |
|----------------------------|---------------------------------|
| 1.int - interrupção | 23.dreq - pedido de DMA |
| 2.nc | 24.dack - reconhecimento de DMA |
| 3.anagnd - GND | 25.d2 - dado |
| 4.anapwr - Vcc | 26.cxad - relógio do A/D |
| 5.nc | 27.d3 - dado |
| 6.vout | 28.d5 - dado |
| 7.nc | 29.d4 - dado |
| 8.nc | 30.d7 - dado |
| 9.ad - entrada do A/D | 31.d6 - dado |
| 10.fonc - sinais acústicos | 32.nc |
| 11.rst - "reset" | 33.nc |
| 12.ring - campainha | 34.a1 - endereço |
| 13.eoc - fim de conversão | 35.iow - escreve na E/S |
| 14.test - teste do A/D | 36.iow - lê da E/S |
| 15.on - liga o A/D | 37.a0 - endereço |
| 16.gnd1 - GND | 38.da - saída o D/A |
| 17.nc | 39.nc |
| 18.gnd0 - GND | 40.disco - pulsos do disco |
| 19.pwr0 - Vcc | 41.nc |
| 20.pwr1 - Vcc | 42.clk400 - relógio de 400 Hz |
| 21.d0 - dado | 43.nc |
| 22.d1 - dado | 44.nc |

DAGRAMAS DE TEMPO

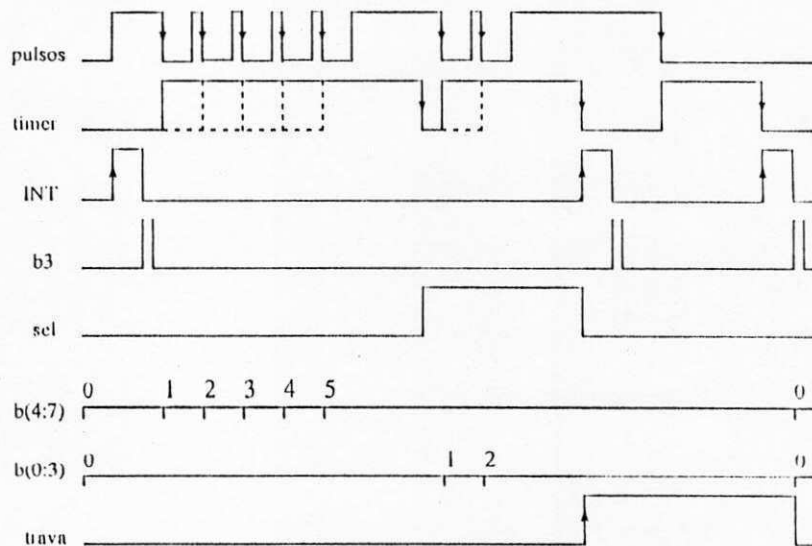
Os diagramas de tempo mostrados nas figuras que se seguem servem para auxiliar no teste de funcionamento do ASIC CPPV.

Reportando-se à pinagem do CPPV (Apêndice B), tem-se:

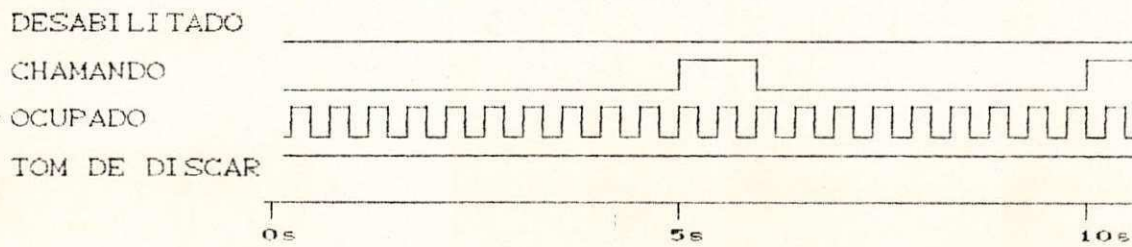
A1 e A0 - são usados para endereçar um dos quatro registradores internos do CPPV, conforme o quadro abaixo (Figura 4.1):

A1	A0	REGISTRADOR
0	0	N
0	1	C
1	0	T
1	1	R

Os sinais internos e externos envolvidos com o Bloco de Controle do Telefone são DISCO (pulsos), timer, INT, B3 do registrador C, sel, CLK400 e trava. Eles podem ser vistos agindo em conjunto através do diagrama de tempo da Figura abaixo; usando um exemplo que foi utilizado na Seção 4.1.1.3.

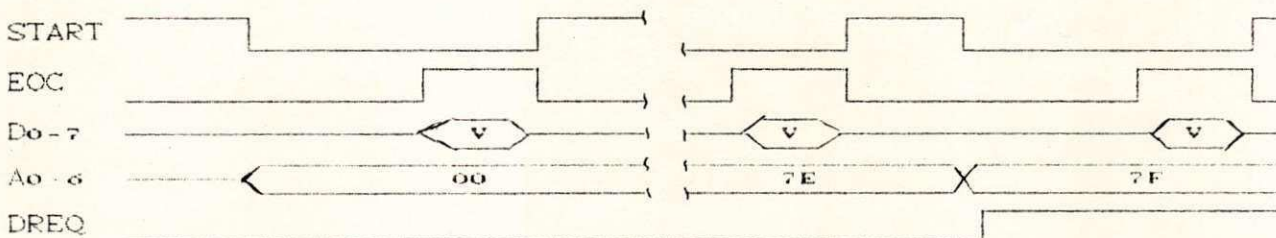


Os sinais relacionados com o Bloco Gerador de Sinais Acusticos são FONE, RING, CLK400, e os bits B2 B1 B0 do registrador C. Os bits B1 B0 selecionam o sinal de saída pelo pino FONE e o bit B2 habilita a saída do pino RING (Figura 4.3). O diagrama de tempo abaixo ilustra as intermitências dos sinais acústicos envolvidos com o bloco acima citado.

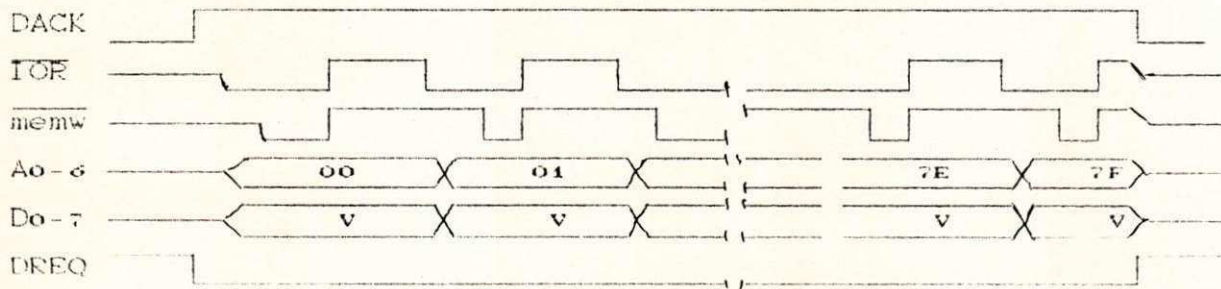


O Bloco Gerenciador dos "Buffers" Intermediários foi dividido em três FASES de funcionamento (Seção 4.1.2.1).

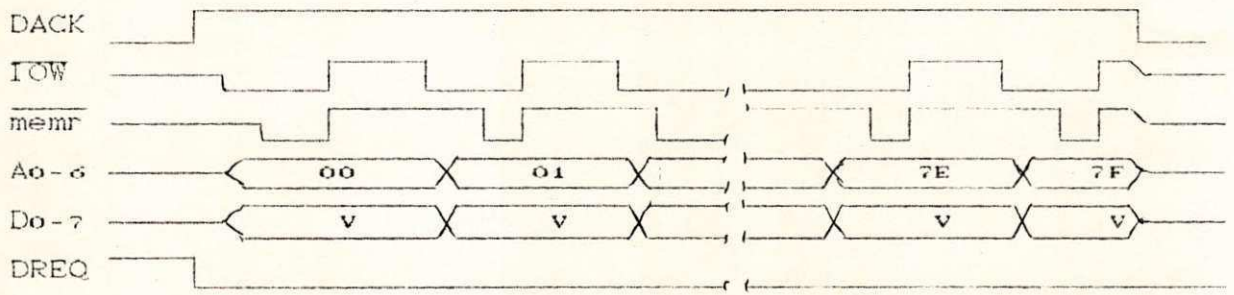
Na FASE I os sinais envolvidos com este bloco são ANAPWR, ANAGND, AD, EOC, TEST, ON e CKAD, todos envolvidos com o Conversor Analógico-Digital, e DREQ. O diagrama de tempo da Figura 4.7 ilustra a sequência de funcionamento deste conversor. O diagrama de tempo abaixo ilustra o processo durante a FASE I:



Na FASE II, os sinais envolvidos são DREQ, DACK, \overline{IOR} , A0-6 e D0-7 (barramento do sistema). O diagrama de tempo abaixo descreve como esta fase funciona:



Na FASE III, os sinais envolvidos são DREQ, DACK, \overline{IOW} , A0-6 e D0-7 (barramento do sistema). O diagrama de tempo abaixo descreve como esta fase funciona:



Solo 1400

O pacote Solo 1400 é um conjunto de programas produzido pela empresa europeia ES2 para projetos de circuitos integrados do tipo células padrões ou "standard cell". Este pacote roda em estações de trabalho SUN, VAX ou PC (numa versão menos poderosa). O Solo 1400 é composto por uma série de programas que, quando executados numa determinada sequência, permitem ao projetista implementar seu circuito integrado, com a confiabilidade desejada para o mesmo, pois cada etapa do projeto só fica concluída quando um resultado previamente planejado é obtido [22].

As etapas para a implementação de um projeto de um circuito integrado utilizando o Solo 1400 são mostradas no diagrama de fluxo da Figura D.1.

Porém, antes de rodar os programas do Solo 1400, é necessário definir um nome e indicar o tipo do processo para a parte do projeto que será implementada [22].

O tipo do processo está relacionado com a unidade padrão de comprimento com que todo o "layout" é parametrizado. Esta unidade padrão é o λ (lambda). No caso deste projeto, o processo escolhido foi o ECPD15, onde o λ equivale a $1.5 \mu\text{m}$ (micrômetros).

A tecnologia avança para que este parâmetro diminua cada vez mais, pois isto implica em redução na área de silício utilizada, isto é, economia de dinheiro. Logo deve-se trabalhar com o processo que apresentar o menor valor de λ .

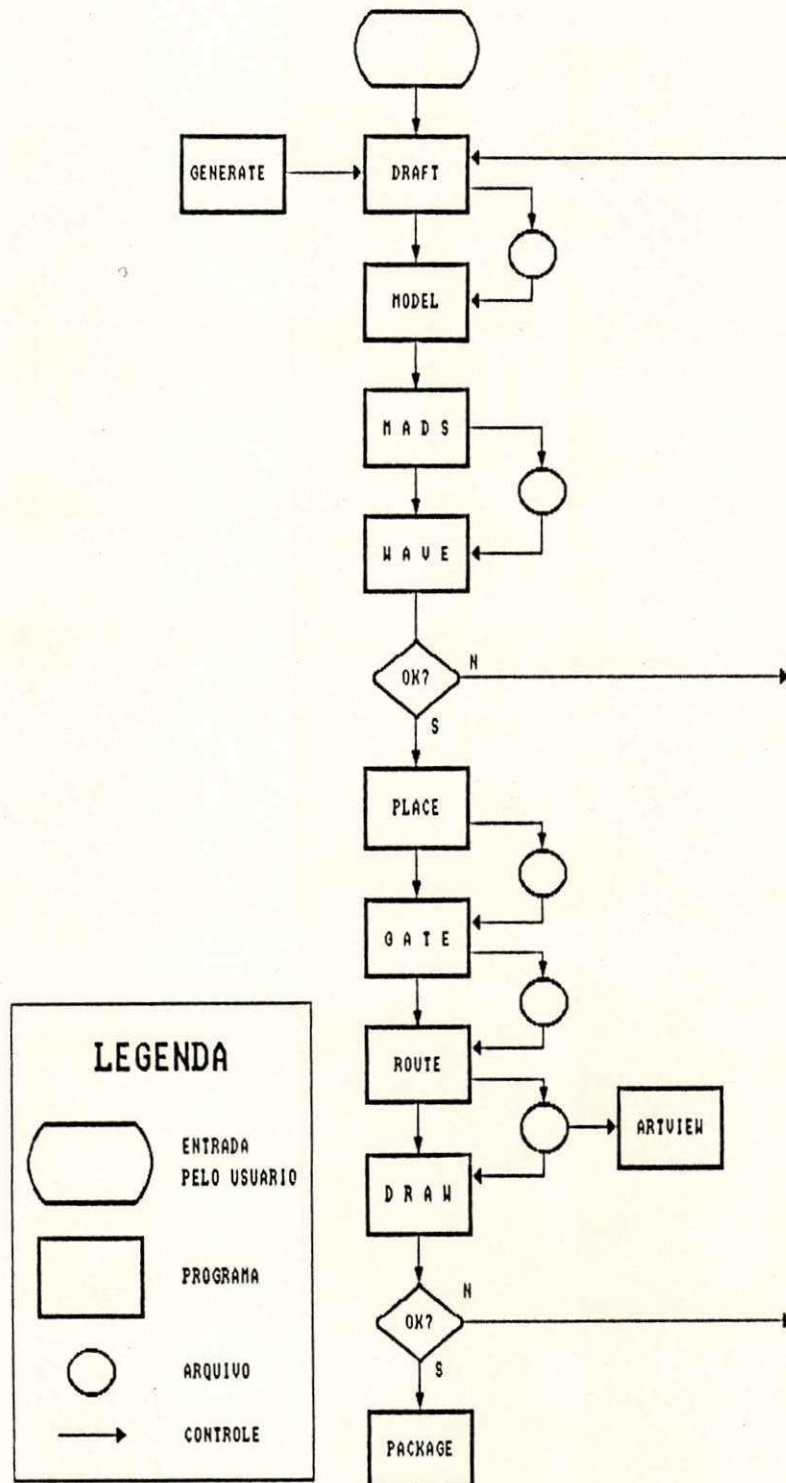


Figura D.1. Seqüência a ser seguida em um proj. com o Solo 1400.

Os passos principais para um projeto com células padrão, utilizando o Solo 1400, são:

- Chamar o programa "DRAFT" que faz a edição e a extração dos circuitos lógicos. Este programa roda em uma janela própria e sua interação com o usuário se faz através do teclado, e de "menus" acessados pelo "mouse". O projetista monta seu circuito lógico desejado utilizando os elementos básicos das bibliotecas disponíveis no SOLO 1400 (portas E, OU, "flip-flops", etc). Uma vez concluída esta montagem, o projetista passa seu circuito lógico por um verificador de ligações para checar se existe alguma ligação faltando ou suspeita. Se esta verificação der negativa, o circuito lógico é fechado em um bloco, onde apenas aparecem as entradas e saídas do circuito (uma caixa preta). Este bloco recebe um nome, passa a fazer parte de uma biblioteca criada pelo projetista e pode ser usado como um elemento de um circuito lógico de nível hierárquico maior. Neste ponto, quando o projetista já tem a parte do projeto que será implementada pronta e checada, dois arquivos são gerados: um que guarda informações do desenho esquemático do circuito lógico e outro que contém a descrição das ligações elétricas deste circuito (arquivo formato MODEL);

- Caso o projetista precise de blocos que apresentem uma certa regularidade estrutural, como memórias RAM, ROM e PLAs, o programa GENERATE gera estes blocos automaticamente sendo preciso

apenas que o projetista entre com os dados pedidos pelo programa via teclado (por exemplo, para a geração de uma memória RAM, é pedido o número de bits por palavra, o número de palavras, o nível lógico que vai ativar as entradas que controlam a memória, etc). Uma vez gerados, estes blocos vão fazer parte da biblioteca do projetista para uso posterior;

- O próximo passo é passar o arquivo formato MODEL pelo compilador MODEL, que vai gerar outros arquivos que irão auxiliar as fases posteriores do projeto;

- Para a simulação lógica do circuito em questão, escreve-se em um editor de textos um arquivo ASCII que contenha um padrão de sinais que serão aplicados nas entradas do circuito lógico;

- O programa MADS é o responsável pela simulação do circuito lógico e reporta ao projetista se houve alguma violação na temporização e em que instante de tempo esta violação ocorreu;

- Para a visualização das formas de onda geradas pelo programa simulador, roda-se o programa WAVE e em uma janela própria, tem-se uma visão geral ou por partes do período de simulação. Pode-se fazer outras simulações do circuito lógico. Para isso, basta fazer a modificação necessária no arquivo de entrada do simulador, rodar novamente o programa MADS, e rodar o programa WAVE. Caso o resultado das simulações não agrade, pode-se modificar a estrutura do circuito lógico através do

programa DRAFT, e repetir todas as etapas citadas acima;

- Passa-se agora para a parte do projeto físico propriamente dito, quando se roda quatro programas em sequência: PLACE que faz a disposição automática das células lógicas nas linhas e colunas que vão formar um "array" na parte central da pastilha, GATE, que coloca as portas lógicas nestas linhas e colunas, e ROUTE que é o responsável pelo roteamento entre as portas lógicas, entre as linhas de células e entre as colunas que compõem a pastilha. Este roteamento é feito em dois níveis de metal, sendo que no nível inferior tem-se trilhas de alumínio dispostas na direção horizontal e no nível superior tem-se as trilhas de alumínio dispostas na direção vertical. Finalmente, roda-se o programa DRAW que gera o arquivo em um formato padrão de descrição das máscaras (formato CIF);

- Para a visualização das máscaras que irão fazer parte do "layout" do projeto, roda-se o programa ARTVIEW. Este programa roda em uma janela própria e mostra as várias máscaras necessárias em um processo de fabricação de um circuito integrado, sendo que cada máscara é mostrada em uma cor padronizada (por exemplo, vermelho para o polisilício, verde para difusões do tipo N, azul marinho para a primeira camada de alumínio, etc). Este programa oferece facilidades como, por exemplo, mostrar apenas algumas máscaras, dar "zoom" em determinadas áreas, observar o circuito lógico não a nível do

portas lógicas mas a níveis de hierarquia, entre outras;

- Para fechar o projeto, roda-se o programa PACKAGE que oferece os tipos de encapsulamentos que podem ser usados para conter a pastilha nas dimensões em que ela se apresenta, além de facilitar a ligação dos "pads" de entrada e saída nos pinos do circuito integrado.