

UMA PROPOSTA DE BAIXO CUSTO PARA A INTERCONEXÃO DE EQUIPAMENTOS

Marcello Thiry Comicholi da Costa
Bacharel em Ciências da Computação, UFSC, 1990.
Aluno de Mestrado em Engenharia Elétrica, UFSC.

Elizabeth Sueli Specialski
M.Sc. em Computação, UFRGS, 1981.

Laboratório de Integração de Software e Hardware
Departamento de Ciências Estatísticas e da Computação
Universidade Federal de Santa Catarina
Caixa Postal 476 - Florianópolis - SC
CEP 88049 - Brasil

RESUMO

Um problema encontrado frequentemente em universidades e pequenas empresas é o alto custo para a interligação de diversos equipamentos. Apesar das diversas soluções encontradas no mercado, quase todas necessitam de hardware adicional, causando uma elevação nos custos finais. Este artigo aborda uma proposta de interconexão para a redução destes gastos evitando a aquisição de novos equipamentos e procurando atender as diversas necessidades dos usuários.

ABSTRACT

High expense of general devices interconnection is a problem frequently found at universities and small companies. Although general solutions are available on the market, almost all of them need additional hardware, raising the final prices. The approach presented in this paper aims the reducing of the final interconnection costs avoiding the necessity of new equipment purchase, while supporting the users demand.

1. INTRODUÇÃO

Ambientes comumente encontrados em universidades e empresas informatizadas são um centro de processamento de dados contendo um ou mais mainframes, um conjunto de microcomputadores espalhados em diversos departamentos e laboratórios e outros recursos caros alocados a locais específicos (impressoras de alta velocidade, discos rígidos de grande capacidade). Quando se pretende interconectar estes ambientes, surgem alguns problemas como custos elevados das redes locais e, aplicações e serviços oferecidos não atendem a todas as necessidades dos usuários. As soluções adotadas são, muitas vezes, imediatistas e resolvem o problema a curto prazo. Desta forma, após um determinado intervalo de tempo se terá investido em diversos softwares e hardwares que na maioria das vezes não são compatíveis entre si, pois, apesar dos esforços das organizações internacionais responsáveis pela padronização, poucos fabricantes tem se preocupado com a conectividade.

Este artigo descreve um trabalho que pretende preencher algumas lacunas deixadas pelos problemas citados anteriormente, trazendo uma proposta para a interconexão de PC's, através de um mecanismo desenvolvido totalmente em software. A escolha dos microcomputadores do tipo IBM-PC foi devido à sua grande disponibilidade geral, sendo ainda muito utilizado em diversos setores empresariais e acadêmicos.

Desenvolveu-se uma rede local em software que funciona sob o ambiente DOS, em background e simulando multiprocessamento, tornando o sistema totalmente transparente ao usuário. A rede SoftNET foi desenvolvida para locais que não necessitam de altas taxas de transmissão e não possuem alto tráfego de informações. O sistema permite a utilização de recursos de rede, sem alterar o desempenho das estações conectadas.

O objetivo principal consiste em permitir uma comunicação de forma eficiente, mas sem elevar os custos, permitindo o compartilhamento de recursos caros de hardware, como impressoras e discos rígidos, através da troca de informações e arquivos entre os usuários do sistema.

2. A REDE LOCAL SoftNET

A SoftNET é uma rede local que pode interconectar diversos microcomputadores do tipo IBM-PC, oferecendo serviços de transferência de arquivos, troca de mensagens e informações de status. Entre estas informações está incluído um comando de diretório remoto.

2.1. CARACTERÍSTICAS

Estruturada com uma topologia em barramento, a rede local SoftNET utiliza o par trançado como meio de transmissão. Na primeira versão projetou-se uma pequena placa que permitia também a utilização de cabo coaxial, porém a idéia foi abandonada devido a sua inviabilidade.

A conexão da estação com o meio físico é feita através de um conector DB25 que deve ser colocado na placa serial RS232 do microcomputador. Os pinos Rx e Tx são curto-circuitados para permitir a topologia em barramento. Para uma maior proteção da serial, utiliza-se ainda o pino 7 (terra de sinal). Desta forma, necessita-se de apenas dois fios para a interligação. A figura 1 mostra um modo simples de proteção com a utilização de dois diodos. Eles evitam que o pino Tx assumam valores de tensão não desejáveis.

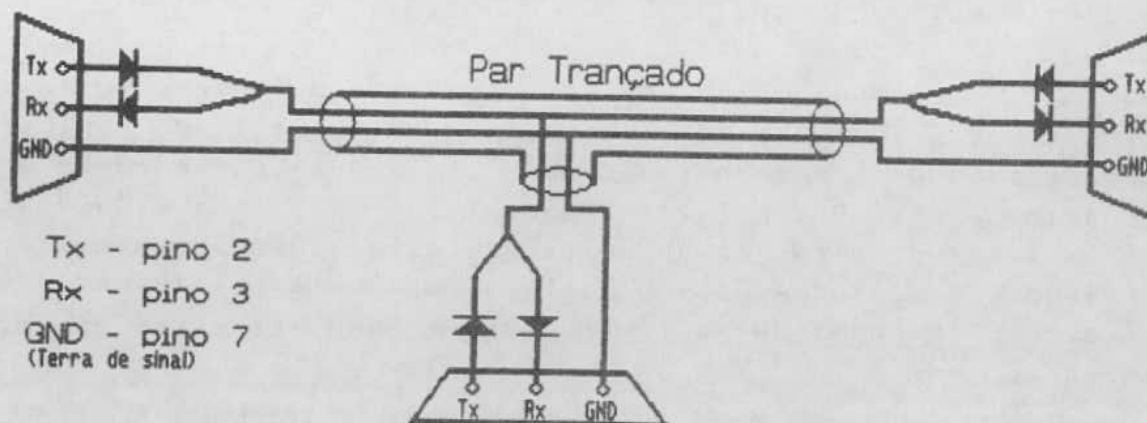


Fig. 1 - Utilização de diodos para proteção

A SoftNET possui endereçamento e velocidade configuráveis, podendo-se ainda escolher qual placa serial a ser

utilizada, já que alguns micros possuem mais de uma saída serial padrão RS232. A faixa de valores permitidos para a velocidade de transmissão está entre 7.680 e 115.200 bps. Esta limitação é dada pela própria placa serial.

O endereçamento de uma estação é feito por meio de um subconjunto do alfabeto. As letras que compõe este subconjunto estão compreendidas entre "F" e "Y". A rede reserva a letra "Z" para endereçar estações que querem se conectar à SoftNET. As letras de "A" a "E" estão reservadas para os dispositivos locais à estação. É ainda permitido um endereço de difusão (para todas as estações), o qual é representado pelo símbolo "*".

Para conectar uma nova estação, a rede envia um quadro de difusão com endereço origem "Z". O endereço com o qual a estação está tentando se conectar é armazenado em um campo de dados. Um campo do quadro contém o comando de conexão. As outras estações verificam o endereço recebido e se alguma já estiver conectada com aquele endereço é enviada uma resposta negativa, caso contrário a estação começa a fazer parte da rede.

O mecanismo de acesso ao meio, segue a filosofia das técnicas de controle distribuído em barramento, baseando-se no método CSMA-CD. Algumas modificações foram feitas para que o algoritmo funcionasse sob o DOS que originalmente foi projetado para ser um sistema operacional monousuário.

2.2. PROTOCOLO DE COMUNICAÇÃO

Um sistema de comunicação todo implementado por software gera um grande problema de overhead. A SoftNET foi projetada para reduzir ao máximo este problema.

As informações trocadas pelas estações devem obedecer ao formato apresentado na figura 2, onde cada campo, excluindo o campo de dados (DAD), possui o tamanho de um byte.

Os campos SYN e STX sincronizam a chegada de um novo quadro. Somente após o reconhecimento destes dois ca-

racteres, a estação inicia o procedimento de recepção de um quadro.

SYN	STX	ORG	DST	COM	LOT	HIT	DAD	ETX	CRC
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

- SYN - Sinaliza a chegada de um novo quadro.
- STX - Indicador de início de quadro.
- ORG - Endereço da estação origem.
- DST - Endereço da estação destino.
- COM - Comando a ser executado.
- LOT - Parte baixa do tamanho de DAD.
- HIT - Parte alta do tamanho de DAD.
- DAD - Contém a informação (área de dados).
- ETX - Indicador de final de quadro.
- CRC - Controle de erros (check-sum).

Fig. 2 - Estrutura de um quadro na rede local SoftNET

Cada quadro é delimitado por dois caracteres especiais, os quais indicam o início de um novo quadro (STX) e o final do mesmo (ETX).

Os dois campos de endereço, ORG e DST, representam o endereço da estação origem e o endereço da estação destino respectivamente. Nesta versão, a SoftNET apresenta um endereçamento feito através de letras do alfabeto, permitindo que estes campos contenham qualquer letra do alfabeto de "F" a "Y", ou ainda, um endereço de difusão (para todas as estações) representado pelo símbolo "*".

O campo COM é reservado para controle da rede e contém os comandos a serem realizados na estação destino.

Os dois campos LOT e HIT são combinados de forma a identificar o tamanho da informação em bytes, a qual está contida no campo de dados (DAD). Obviamente este último campo possui tamanho variável.

O último campo (CRC) é reservado para controle de erros. A técnica utilizada é a paridade longitudinal, que apesar de ser simples, é funcional para os propósitos estabelecidos.

3. SERVIÇOS APRESENTADOS PELA SoftNET

Uma das aplicações comuns às redes locais é o compartilhamento de periféricos caros entre diversos micros. A SoftNET não permite um compartilhamento direto de programas armazenados no disco rígido de uma determinada estação, porém permite um serviço de transferência de arquivos.

Na maioria das redes existem dois tipos de equipamentos: os usuários que funcionam como estações de trabalho e os equipamentos servidores que são responsáveis pela prestação de serviços à rede, tais como gerência de unidades de memória de massa e impressoras. A SoftNET não faz este tipo de distinção, sendo permitido que qualquer estação trabalhe tanto como usuária como estação servidora. A responsabilidade fica a cargo do usuário.

Na versão inicial da SoftNET foram implementados alguns serviços para a manipulação de arquivos e mensagens entre as estações. Algumas alterações foram propostas para versão posterior, tais como serviços de impressão e correio eletrônico. Com estes recursos implementados, a SoftNET deixará de ser apenas um recurso didático, tornando-se uma ferramenta de suporte ao desenvolvimento de sistemas.

3.1. SISTEMA OPERACIONAL DE REDE

A tarefa principal do Sistema Operacional de Rede (SOR) implementado para a SoftNET é permitir o compartilhamento de um disco rígido a nível de arquivos.

O SOR foi obtido a partir da agregação de módulos de extensão ao sistema operacional nativo (DOS), garantindo-se compatibilidade dos softwares já existentes ao novo ambiente. Os comandos de rede foram, desta forma, anexados ao DOS. Esta alteração procura ser a mais transparente possível ao usuário.

Os comandos convencionais COPY e DIR foram estendidos para atenderem também a chamados remotos, sem que, com isto, perdessem suas antigas funções. Para garantir a padronização procurou-se manter a sintaxe inalterada, inclusive para comandos de rede.

Novos comandos foram criados para satisfazer algumas necessidades. O comando MSG permite a transferência de mensagens, sendo um comando válido somente quando a rede estiver instalada. Para verificar o nome com o qual a estação foi conectada, bem como sua alteração, foi implementado o comando USER. Dois outros comandos estão sendo propostos para a próxima versão: os comandos TYPE e DEL, que são de fácil implementação, pois possuem características semelhantes aos já oferecidos.

. COPY: Comando já oferecido pelo DOS para realizar cópias locais. Com a utilização da SoftNET passa a ser permitido também a transferência de arquivos entre estações.

```
Ex: C>copy a:arquivo.pas [comando local]
    A>copy nomes.dad j: [comando remoto]
    B>copy r:c:codigos.dta [comando remoto]
```

. DIR: Permite a listagem dos arquivos existentes em uma estação remota ou de um drive local.

```
Ex: C>dir b:*.* [comando local]
    A>dir g:nomes.* [comando remoto]
    B>dir h:d:teste??.0?? [comando remoto]
```

. MSG: Está direcionado à troca de mensagens entre estações. Este comando permite mensagens de difusão, sendo o único com este tipo de endereçamento. O escopo para este comando obedece a seguinte sintaxe:

MSG "mensagem a transmitir" (estação destino):

Para o envio de mensagens de difusão, utiliza-se o caracter "*" no campo estação destino.

```
Ex: C>msg "Bom dia estação h:" h:
    C>msg "O horário de funcionamento acaba às
6:00" *:
```

USER: Permite fazer uma verificação do endereço com o qual sua estação foi conectada. Ainda através deste comando pode-se tentar modificar o endereço atual da estação. Para tal, o comando apresenta a seguinte sintaxe:

USER [Estação]

Caso seja especificado a estação, a rede tentará a conexão com o novo endereço, tendo como resultado desta operação, uma mensagem de status.

Ex: A>user [informa o nome da estação]
B>user H [tenta alterar o nome da estação]

4. IMPLEMENTAÇÃO

A rede local SoftNET foi implementada com uma metodologia baseada em paralelismo. Apesar de seu ambiente de funcionamento (DOS) ser monousuário, a implementação simula multiprocessamento através das interrupções do PC. Esta característica permite transparência ao usuário.

A implementação pode ser dividida em duas partes distintas. Um programa de iniciação, denominado módulo transiente, é executado pelo usuário na conexão da estação à rede ou na desinstalação da rede, sendo descartado posteriormente (fig. 3.a). O outro segmento é o módulo que fica residente em memória, sendo responsável pelo tratamento dos comandos de rede e por todo o controle de acesso ao meio (fig. 3.b).

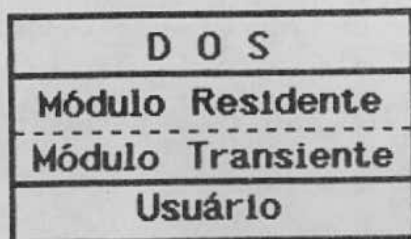


Fig. 3a - Memória durante a instalação

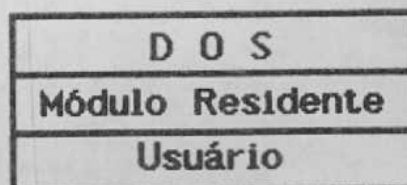


Fig. 3b - Memória após a instalação

Ambos os módulos foram implementados em linguagem assembly 8086/8088, devido às suas características de flexibilidade no tratamento de interrupções e pela garantia de velocidade na execução das rotinas que funcionam em background. Uma versão em linguagem C foi proposta, tendo em vista a maior facilidade de alteração e acompanhamento do funcionamento da rede.

4.1. MÓDULO TRANSIENTE

Este módulo é responsável pela instalação da rede local SoftNET na estação e pela conexão desta à rede. Estes dois processos são realizados através de apenas um comando, o qual deve ser invocado do sistema operacional pelo usuário.

Na instalação a rede deve ser chamada com os seguintes parâmetros: nome da estação, número da serial e fator de velocidade. O nome da estação, como já foi visto, é uma letra de "F" a "Y" (totalizando um máximo de 20 estações). O número da serial pode assumir apenas dois valores. Se o parâmetro contiver o valor "1" a serial assumida será a COM1. Caso o valor seja "2" será assumida a COM2. O fator de velocidade é um dígito hexadecimal entre "1" e "F". A velocidade correspondente para cada dígito pode ser encontrada na fig. 4.

FATOR	1	2	3	4	5
VELOCIDADE (bps)	115200	57600	38400	28800	23040
FATOR	6	7	8	9	A
VELOCIDADE (bps)	19200	16457	14400	12800	11520
FATOR	B	C	D	E	F
VELOCIDADE (bps)	10472	9600	8861	8228	7680

Fig. 4 - Correspondência entre o fator de velocidade e a velocidade real

Ex. 1: C>sn g 1 c

Ex. 2: A>sn t 2 2

No primeiro exemplo, o módulo transiente fará a conexão à rede com o nome de estação "g", assumindo a serial COM1 e com velocidade de 9600 bps. Analogamente o segundo exemplo mostra uma conexão com o nome de estação "t", assumindo a serial COM2 e velocidade de 57600 bps.

O módulo transiente realiza ainda a desinstalação da rede, liberando toda a memória ocupada pela rede e restaurando os vetores de interrupção. Para desinstalar a rede o usuário deve proceder como na instalação, passando apenas o parâmetro "/u".

4.2. MÓDULO RESIDENTE EM MEMÓRIA

As principais rotinas da rede que farão o tratamento das interrupções, de erro crítico, dos comandos de rede e de todo o protocolo de acesso ao meio estão contidas em um segmento de programa que após a instalação da rede fica residente em memória. Desta forma, a estação fica liberada e pronta para executar qualquer comando do sistema operacional, inclusive comandos de rede.

Sempre que for enviado um comando para ser executado pelo sistema operacional, o módulo residente intercepta e analisa a linha de comando do DOS através da interrupção de software 21h. Caracterizado um comando de rede, a linha de comando é copiada para um buffer local e uma linha nula será devolvida ao DOS após o processamento do comando. Caso contrário, a linha será imediatamente devolvida ao sistema operacional, que ficará responsável pela sua análise e posterior execução.

Um comando de rede tem sua análise de sintaxe, controle de erros e execução feitas neste módulo. Para executar qualquer comando remoto, a SoftNET possui quatro processos principais que são responsáveis pela recepção, tratamento e transmissão dos quadros e ainda por toda a temporização do sistema (fig. 5).

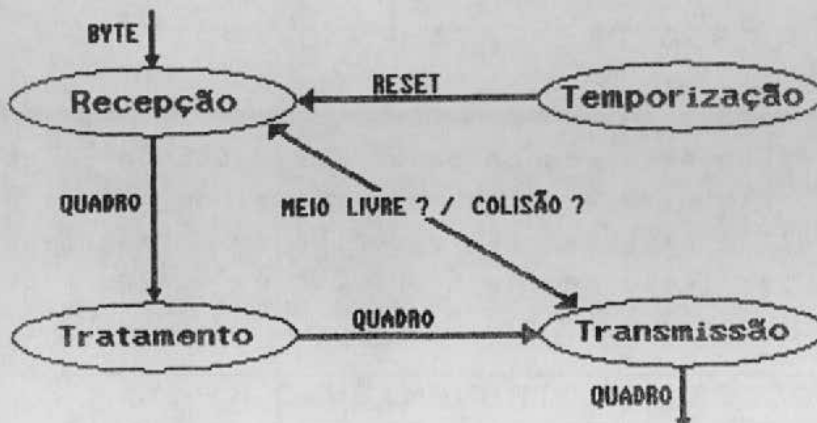


Fig. 5 - Interação do principais processo da SoftNET

4.2.1. PROCESSO DE RECEPÇÃO (P1)

Este processo é disparado pela chegada de um byte na entrada serial do microcomputador. A serial do PC gera uma interrupção, fazendo com que o quadro seja montado byte a byte. De acordo com cada byte recebido, P1 atualiza o contador de recepção. Quando o quadro estiver completo, P1 analisa seu destino e seu comando, caso não seja um comando de resposta e seja destinado à sua estação, é sinalizado à P4 o envio de uma resposta. A seguir o quadro é enviado para P3 que fará o tratamento adequado. Nesta etapa são detectadas a ocorrência de colisões e de erros de transmissão.

4.2.2. PROCESSO DE TEMPORIZAÇÃO (P2)

Processo disparado pela interrupção de CLOCK do PC (Timer Ticks). Faz a temporização da chegada de um quadro. Caso P1 não tenha recebido um quadro em um tempo n pré-estabelecido, P2 envia à P1 um pedido de inicialização de seus apontadores e retorno ao seu estado inicial, ou seja, à espera de um novo quadro. Se um quadro completo estiver pronto para ser tratado é chamado o processo P3. É ainda neste processo que é feita a verificação da variável que indica a transmissão de uma resposta. Este processo é fundamental para o funcionamento da rede, pois toda a comunicação é feita pela passagem de variáveis, deixando a cargo deste processo a ativação dos demais.

4.2.3. PROCESSO DE TRATAMENTO (P3)

Ativado pela temporização (P2). Quando P1 está com um quadro completo, ele seta uma variável de quadro pronto. Esta variável é testada pelo processo de temporização. O tratamento a ser feito vem no campo COM do quadro recebido.

4.2.4. PROCESSO DE TRANSMISSÃO (P4)

Disparado por P1, tem a função de colocar um quadro no meio de transmissão. P4 faz todos os testes necessá-

rios antes de tentar enviar o quadro. Verifica se P1 está recebendo algum quadro e em caso negativo, inicia sua transmissão. Detectando o meio ocupado, P4 aguardará um tempo aleatório antes de uma nova tentativa. Durante a transmissão, P4 se comunicará com P1 para verificar se ocorreu alguma colisão. Neste caso será aguardado um tempo aleatório antes de uma nova transmissão. De acordo com o padrão estabelecido para o protocolo CSMA/CD serão feitas até 16 tentativas de retransmissão.

5. CONCLUSÕES

Implementou-se uma rede local que utiliza o conceito de transparência, permitindo ao usuário trabalhar com diversas aplicações concomitantemente com os processos residentes em memória, que funcionam em background.

Uma parte do vetor de interrupção do microcomputador é alterada para a segurança da rede e do próprio usuário. Isto fornece aos usuários a possibilidade de desviar as interrupções do PC para suas próprias rotinas. Por exemplo, um programa que usa o clock pode ser carregado, pois a SoftNET após realizar o tratamento desta interrupção passa a chamada para a antiga posição no vetor de interrupção. Entretanto, alguns programas que também ficam residentes em memória podem não tomar esta precaução, fazendo com que seja conveniente instalar a SoftNET somente após o carregamento destes.

Sendo uma rede em software, a SoftNET não tem como controlar a configuração da saída serial pelos usuários. Portanto, após a rede ter sido instalada, não devem ser carregados outros programas que alterem a configuração da saída serial (RS232) usada pela rede.

A SoftNET foi projetada para controle de um tráfego irregular de mensagens e o seu desempenho depende de como ela será utilizada.

Quanto à compatibilidade da rede local SoftNET pode-se dizer que esta funciona sob qualquer sistema operacional compatível com a versão MS/DOS 2.0 ou superior, tornando-a bastante versátil quanto ao ambiente de utilização.

Como a rede foi implementada em software, a especificação original do CSMA/CD teve que ser um pouco alterada para que a performance da rede não fosse prejudicada. Todo o controle de erros e colisões é feito pela estação que originou a conversação. Desta forma a estação que trabalha em background não sofre interferência, permanecendo o conceito de transparência.

A principal vantagem da rede aqui apresentada está no seu baixo custo final, na simplicidade de instalação e de sua utilização aliadas ao fato de que ela oferece os principais serviços utilizados em um ambiente restrito.

7. REFERÊNCIAS

- [1] TANENBAUM, A. S.; COMPUTER NETWORKS; Prentice-Hall, 1988.
- [2] GIOZZA, W. F. e outros; REDES LOCAIS DE COMPUTADORES - TECNOLOGIA E APLICAÇÕES; McGraw-Hill, São Paulo, 1986.
- [3] LAGES, N. A. C. e NOGUEIRA, J. M. S.; INTRODUÇÃO AOS SISTEMAS DISTRIBUÍDOS; UNICAMP, Campinas, 1986.
- [4] THIRY C. C., M. e IBAGY, S. R.; SoftNET, UMA REDE LOCAL DE BAIXO CUSTO; TRABALHO DE CONCLUSÃO DO CURSO DE GRADUAÇÃO EM COMPUTAÇÃO, UFSC, 1990.
- [5] MOURA, J. A. B. e outros; REDES LOCAIS DE COMPUTADORES - PROTOCOLOS DE ALTO NÍVEL E AVALIAÇÃO DE DESEMPENHO; McGraw-Hill, São Paulo, 1986.
- [6] HOGAN, T.; THE PROGRAMMER'S PC SOURCEBOOK; Microsoft Press.
- [7] SANTOS, A. S.; MANUAL DE REFERÊNCIA TÉCNICA - PCPAQ; Microtec, São Paulo, 1986.
- [8] RECTOR R. e ALEX Y G.; THE 8086 BOOK (INCLUDES THE 8088); Osborne/McGraw-Hill, Berkeley, CA, 1980.

- [9] NORTON, P.; LINGUAGEM ASSEMBLY PARA IBM PC; Campus, Rio de Janeiro, 1988.
- [10] SIEMENS AG; SAB 8086/8088 - SAB 80186/80188 - PROGRAMMER'S REFERENCE MANUAL; Siemens, 1985.
- [11] MICROSOFT; MICROSOFT MS-DOS 3.3 - GUIA DO USUÁRIO E REFERÊNCIA DO USUÁRIO; Microsoft, 1988.
- [12] IBM Corporation; TECHNICAL REFERENCE: PERSONAL COMPUTER XT (SOFTWARE AND HARDWARE); IBM, 1983.
- [13] INTEL Corporation; MICROPROCESSOR AND PERIPHERAL HANDBOOK; Intel, Santa Clara, CA, 1987.