

ARNALDO DE ALBUQUERQUE ARAÚJO

SISTEMA DE AQUISIÇÃO DE DADOS  
PARA UM CAMPO DE TESTES DE  
TURBINAS EÓLICAS

Dissertação apresentada à Coordenação dos Cursos de Pós-Graduação em Engenharia Elétrica da Universidade Federal da Paraíba, em cumprimento às exigências para obtenção do Grau de Mestre em Engenharia Elétrica.

ÁREA DE CONCENTRAÇÃO: **Processamento da Informação**

ORIENTADOR: **Joberto Sérgio Barbosa Martins**

CAMPINA GRANDE - PB

JANEIRO - 1981

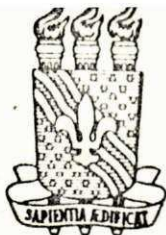


A663s      Araújo, Arnaldo de Albuquerque.  
              Sistemas de aquisição de dados para um campo de testes  
              de turbinas eólicas / Arnaldo de Albuquerque Araújo. -  
              Campina Grande, 1981.  
              95 f.

              Dissertação (Mestrado em Engenharia Elétrica) -  
              Universidade Federal da Paraíba, Centro de Ciências e  
              Tecnologia, 1981.  
              "Orientação: Prof. Dr. Joberto Sérgio Barbosa Martins".  
              Referências.

              1. Turbinas Eólicas. 2. Energia Eólica. 3. Sistema de  
              Aquisição de Dados. 4. Engenharia Elétrica - Dissertação.  
              I. Martins, Joberto Sérgio Barbosa. II. Universidade  
              Federal da Paraíba - Campina Grande (PB). III. Título

CDU 621.548.4(043)



CPqEE/CCT-UFPB

COORDENAÇÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

CENTRO DE CIÊNCIAS E TECNOLOGIA

UNIVERSIDADE FEDERAL DA PARAÍBA

Parecer final do julgamento da dissertação de mestrado

de ARNALDO DE ALBUQUERQUE ARAÚJO

Título: "Sistemas de Aquisição de Dados para um Campo  
de Testes de Turbinas Eólicas".

Conceito obtido: *APROVADO*

COMISSÃO EXAMINADORA

Prof. Joberto Sérgio B. Martins - M.Sc. *J. S. B. Martins*

- Presidente -

Prof. Gurdip Singh Deep - Ph.D. *Gurdip Singh Deep*

Prof. José Homero Feitosa Cavalcanti - M.Sc. *J. Homero*

Prof. Ivan Rocha Neto - Ph.D. *Ivan Rocha Neto*

Campina Grande, 24 de janeiro de 1981.

## AGRADECIMENTOS

Aos que, de uma forma ou de outra, contribuíram para o desenvolvimento deste trabalho e particularmente, aos professores:

- Dr. G. S. Deep, pela sua verdadeira orientação acadêmica,
- Dr. IVAN ROCHA NETO, pela sugestão deste trabalho,
- JOBERTO SÉRGIO B. MARTINS, MSc e MISAEL ELIAS DE MORAIS, MSc, pela orientação deste trabalho,

e também pelo apoio, incentivo e amizade demonstrados.

A minha noiva, a meus pais, pela sua compreensão e estímulo.

Ao CNPq, pela doação de um terminal de vídeo que possibilitou o desenvolvimento do software deste sistema de aquisição de dados.



## R E S U M O

O projeto da automação da aquisição de dados de experimentos e equipamentos é um tópico que tem recebido atenção considerável atualmente e muitos sistemas de aquisição de dados têm sido desenvolvidos com base em microprocessadores.

O objetivo deste trabalho é descrever o projeto de um sistema de aquisição de dados baseado em microprocessador, dedicado e modular. A modularidade aplica-se tanto ao "hardware" quanto ao "software" e permite que modificações no sistema proposto, que se façam necessárias, sejam realizadas sem maiores problemas. O projeto da unidade de controle do sistema foi baseado no microprocessador MC6800 da MOTOROLA.

O referido sistema de aquisição de dados foi desenvolvido para ser aplicado no campo de testes de turbinas eólicas do Núcleo de Energia Eólica do Centro de Ciências e Tecnologia da UFPb, Campus de Campina Grande. Este sistema tem como finalidades coletar e processar parâmetros sinóticos, elétricos e mecânicos, imprimir relatórios técnicos com os resultados dos testes e armazenar dados num meio de armazenamento de massa.

## A B S T R A C T

The design of automatic data acquisition systems for experiments and equipments is a subject that has received considerable attention recently, and many data-acquisition systems have been designed based on microprocessors.

The purpose of this report is to describe a dedicated and modular microprocessor-based data-acquisition system. The modularity applies to hardware, as well as to software, and allows the required changes on the proposed system to be executed without great problems. The system control unit was built based on MOTOROLA 6800.

The present data-acquisition system was built for use at the wind turbine test site of the Núcleo de Energia Eólica do Centro de Ciências e Tecnologia da UFPb, Campus de Campina Grande.

The objectives of this system are to acquire and process wind, electrical and mechanical parameters, to print technical reports of the measurement results and to store data in a mass storage medium.

# Í N D I C E

		PÁGINA
<b>CAPÍTULO I</b>	- INTRODUÇÃO	1
1.1	- MICROPROCESSADORES EM GERAL	5
1.2	- DESCRIÇÃO DO SISTEMA PROPOSTO	9
<b>CAPÍTULO II</b>	- "HARDWARE"	12
2.1	- ESTRUTURA DO "HARDWARE"	12
2.2	- UNIDADE CENTRAL DE PROCESSAMENTO	14
2.2.1	- BREVE DESCRIÇÃO DO MICROPROCESSADOR MC6800 da MOTOROLA - MPU	14
2.2.1.1	- SINAIS DA MPU	16
2.2.1.2	- REGISTRADORES DA MPU	18
2.3	- MEMÓRIAS	19
2.4	- INTERFACES DE AQUISIÇÃO DE DADOS	22
2.5	- ENTRADA/SAÍDA	26
<b>CAPÍTULO III</b>	- "SOFTWARE"	30
3.1	- ESTRUTURAÇÃO DO "SOFTWARE"	30
3.2	- PROGRAMA DE ATENDIMENTO AO SINAL RESET - PARES	33
3.3	- PROGRAMA DE CONTROLE - PC	36
3.4	- SUBROTINA PARA TRANSMISSÃO DE TABELAS - SRT/OUTTAB	37
3.5	- SUBROTINA PARA RECEPÇÃO DE CHARACTER - SRT/INICT	41
3.6	- SUBROTINA PARA TRANSMISSÃO DE UM CHARACTER - SRT/OUTICT	41
3.7	- PROGRAMA DE ATENDIMENTO À INTERRUPÇÃO NMI - PANMI	41



## ÍNDICE DAS FIGURAS

	PÁGINA
<b>FIGURAS</b>	
FIG. 2.1 - DIAGRAMA DE BLOCOS DO <b>SADNE</b>	13
FIG. 2.2 - CARTÃO DA <b>CPU</b>	15
FIG. 2.3.1 - MAPA DE MEMÓRIA	20
FIG. 2.3.2 - SELEÇÃO DE ENDEREÇOS	21
FIG. 2.3.3 - CARTÃO DE MEMÓRIA <b>PROM</b>	23
FIG. 2.3.4 - CARTÃO DE MEMÓRIA <b>RAM</b>	24
FIG. 2.4 - CARTÃO DE INTERFACE PARA AQUISIÇÃO DE DADOS	25
FIG. 2.5 - CARTÃO DE E/S	28
FIG. 3.1.1 - SOFTWARE EM DIAGRAMA DE BLOCOS	31
FIG. 3.1.2 - ESTRUTURAÇÃO DO <b>SOFTWARE</b>	32
FIG. 3.2.1 - SEQUÊNCIA DA INTERRUPTÃO <b>RESET</b>	34
FIG. 3.2.2 - FLUXOGRAMA DO PROGRAMA DE ATENDIMENTO AO SINAL <b>RESET - PARES</b>	35
FIG. 3.3 - FLUXOGRAMA DO PROGRAMA DE CONTROLE - <b>PC</b>	38
FIG. 3.4 - FLUXOGRAMA DA SUBROTINA PARA TRANSMISSÃO DE TABELAS - <b>SRT/OUTTAB</b>	40
FIG. 3.5 - FLUXOGRAMA DA SUBROTINA PARA RECEPÇÃO DE CARACTER - <b>SRT/INICT</b>	42
FIG. 3.6 - FLUXOGRAMA DA SUBROTINA PARA TRANSMISSÃO DE UM CARACTER - <b>SRT/OUTICT</b>	43
FIG. 3.7.1 - SEQUÊNCIA DA INTERRUPTÃO <b>NMI</b>	44
FIG. 3.7.2 - FLUXOGRAMA DO PROGRAMA DE ATENDIMENTO À INTERRUPTÃO - <b>NMI - PANMI</b>	46
FIG. 3.8.1 - FLUXOGRAMA DO PROGRAMA DO <b>RTC</b>	47
FIG. 3.8.2 - FLUXOGRAMA DO PROGRAMA <b>DSPLAY</b>	51
FIG. 3.8.3 - FLUXOGRAMA DO TESTE DE ANO BISSEXTO - <b>TABIS</b>	51



## FIGURAS

FIG. 3.9.1	- FLUXOGRAMA DO PROGRAMA DE ENTRADA DE PARÂMETROS - <b>PEP</b>	57
FIG. 3.9.2	- FLUXOGRAMA DA SUBROTINA QUE RECEBE PARÂMETROS - <b>SRT/INPAR</b>	65
FIG. 3.9.3	- FLUXOGRAMA DA SUBROTINA QUE PROCESSA E ARMAZENA PARÂMETROS RECEBIDOS - <b>SRT/STOC</b>	67
FIG. 3.10	- FLUXOGRAMA DO PROGRAMA DE RESET DOS PARÂMETROS - <b>PRP</b>	69
FIG. 3.11.1	- FLUXOGRAMA DO PROGRAMA DE STATUS DOS PARÂMETROS - <b>PSP</b>	71
FIG. 3.11.2	- FLUXOGRAMA DA SUBROTINA QUE ENVIA PARÂMETROS - <b>SRT/SENDP</b>	75
FIG. 3.11.3	- FLUXOGRAMA DA SUBROTINA QUE ENVIA VETORES DE ATIVIDADE - <b>SRT/SENDVA</b>	76
FIG. 3.12.1	- FLUXOGRAMA DO GERENTE DA AQUISIÇÃO DE DADOS - <b>GAD</b>	78
FIG. 3.12.2	- FLUXOGRAMA DO PROGRAMA DE AQUISIÇÃO DE DADOS - <b>PAD</b>	82
FIG. 3.13.1	- FLUXOGRAMA DO GERENTE DO PROCESSAMENTO DA INFORMAÇÃO - <b>GPI</b>	87
FIG. 3.13.2	- FLUXOGRAMA DO PROGRAMA DE PROCESSAMENTO DA INFORMAÇÃO DO ANEMÔMETRO - <b>PPAN</b>	92

## LISTA DAS PRINCIPAIS ABREVIações UTILIZADAS NESTE TRABALHO

### ABREVIações

<b>ACIA</b>	- ASYNCHRONOUS COMMUNICATIONS INTERFACE ADAPTER
<b>ACC</b>	- ACUMULADOR
<b>A/D</b>	- ANALÓGICO/DIGITAL
<b>ALU</b>	- ARITHMETIC LOGIC UNIT
<b>ASCII</b>	- AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE
<b>AUTO</b>	- MODO DE FUNCIONAMENTO AUTOMÁTICO
<b>BCD</b>	- BINARY CODE FOR DECIMAL
<b>CCR</b>	- CONDITION CODE REGISTER
<b>CI</b>	- CIRCUITO INTEGRADO
<b>CPO</b>	- MODO DE FUNCIONAMENTO CONTROLADO PELO OPERADOR
<b>CPU</b>	- CENTRAL PROCESSING UNIT
<b>CS</b>	- CHIP SELECT
<b>D/A</b>	- DIGITAL/ANALÓGICO
<b>DSPLAY</b>	- PROGRAMA QUE ATUALIZA OS DISPLAYS DO RTC
<b>E/S</b>	- ENTRADA/SAÍDA
<b>GAD</b>	- GERENTE DA AQUISIÇÃO DE DADOS
<b>GPI</b>	- GERENTE DO PROCESSAMENTO DA INFORMAÇÃO
<b>IRQ</b>	- INTERRUPT REQUEST
<b>MPU</b>	- MICROPROCESSING UNIT
<b>NMI</b>	- NON-MAKABLE INTERRUPT
<b>PAK-7</b>	- PROGRAMA DE ARMAZENAMENTO EM FITA CASSETE
<b>PANMI</b>	- PROGRAMA DE ATENDIMENTO À INTERRUPTÃO <u>NMI</u>
<b>PARES</b>	- PROGRAMA DE ATENDIMENTO AO SINAL <u>RESET</u>
<b>PEP</b>	- PROGRAMA DE ENTRADA DE PARÂMETROS
<b>PC</b>	- PROGRAMA DE CONTROLE

## ABREVIACOES

PIA	- PERIPHERAL INTERFACE ADAPTER
PIR	- PROGRAMA DE IMPRESSO DE RELATRIOS
PLK-7	- PROGRAMA DE LEITURA DE FITA CASSETE
PROM	- PROGRAMMABLE READ ONLY MEMORY
PRP	- PROGRAMA DE RESET DOS PARMETROS
PRTC	- PROGRAMA DO RTC
PSP	- PROGRAMA DE STATUS DOS PARMETROS
RAM	- RANDOM ACCESS MEMORY
ROM	- READ ONLY MEMORY
RS	- REGISTER SELECT
RTC	- REAL TIME CLOCK
R/W	- READ/WRITE
SADNE	- SISTEMA DE AQUISIO DE DADOS DO NCLEO DE ENERGIA ELICA DA UFPb.
SP	- STACK POINTER
SRT/INICT	- SUBROTINA PARA RECEPO DE CHARACTER
SRT/INPAR	- SUBROTINA QUE RECEBE PARMETROS
SRT/OUTICT	- SUBROTINA PARA TRANSMISSO DE UM CHARACTER
SRT/OUTTAB	- SUBROTINA PARA TRANSMISSO DE TABELAS
SRT/STOC	- SUBROTINA QUE ARMAZENA E PROCESSA PARMETROS RECEBIDOS
TTL	- TRANSISTOR TRANSISTOR LOGIC
TTY	- TELETYPEWRITER
VATB <sub>m</sub>	- VETOR DE ATIVIDADE DA BASE DE TURBINA "m"
VATTB	- VETOR DE ATIVIDADE DAS TORRES E BASES
VATT <sub>n</sub>	- VETOR DE ATIVIDADE DA TORRE "n"
VCO	- VOLTAGE CONTROLLED OSCILATOR
VMA	- VALID MEMORY ADDRESS

ã minha pobre região nordestina



## CAPÍTULO I

### INTRODUÇÃO

A automação da aquisição de dados de laboratórios, experimentos e equipamentos é um tópico que recebe considerável atenção atualmente e que tem se desenvolvido rapidamente. Mais e mais organizações voltam-se para controle e monitoramento por computadores.

O aparecimento dos microprocessadores no início da década de 70 tornou possível dedicar-se, a baixo custo, um computador a certos experimentos. Sistemas de aquisição de dados baseados em microprocessadores encontram uma gama incrível de aplicações. A complexidade destes sistemas varia consideravelmente, embora a maioria possua métodos e técnicas comuns que requerem considerações especiais.

Uma outra vantagem dos sistemas, baseados em microprocessadores, de aquisição de dados e de controle de processos e experimentos encontra-se no fato de que os mesmos podem ser conectados a redes locais de computadores, tendo assim acesso a plotadores, grandes bancos de dados e outros recursos que são economicamente inviáveis para pequenos sistemas dedicados.

Em sistemas típicos de aquisição de dados ou de controle de processos e experimentos, os dados são adquiridos através de um ou mais conversores analógico/digital (A/D) e um multiplexador. A aquisição dos dados pode ser cíclica ou em resposta a pedidos de interrupção gerados pelos equipamentos ou ainda em resposta a distúrbios no processo. Os dados resultantes de entrada são geralmente convertidos para as unidades apropriadas e formatadas. Os dados podem então ser impressos ou armazenados para processamento posterior. A natureza do experimento é que dita o volume e taxas

de dados necessários.

No caso de sistemas de controle de processos, sinais de controle (analógicos ou digitais) apropriados devem ser gerados e aplicados ao processo através de conversores digital/analógico (D/A) e distribuidores ou demultiplexadores. A natureza do processo é que dita as taxas de respostas e a complexidade dos cálculos de malha fechada.

Nos dias atuais a "crise" energética assume um caráter profundo, exigindo um rápido esforço no sentido de definirmos uma estrutura de oferta, compatível com nossos recursos naturais e capacidade tecnológica, possibilitando o rompimento num "único processo de dois laços de dependência: o tecnológico e o energético".

O procedimento estabelecido para a derivação das prioridades dos programas da área de energia encobre duas dificuldades. A primeira diz respeito ao julgamento conjunto das fontes, envolvendo as atuais de grande porte (eletricidade, petróleo e carvão) e as demais fundamentalmente de origem solar. Em outro sentido, há uma segunda quando se engloba os impactos sobre a estrutura energética nacional, impedindo, até certo ponto, vir à tona as prioridades regionais.

Unidades eólicas datam dos primórdios da civilização, e o quadro encontrado antes da recente crise de energia apresentava em um grande número de países milhares de unidades em operação (por exemplo, nos EEUU mais de 150.000 unidades). No entanto, a ênfase seguida era profundamente distinta da atual fase. Hoje deve-se procurar fontes de energia, a qual não precise ser desativada após a extensão da rede de fornecimento de energia elétrica, seja devido à interligação das unidades eólicas ao sistema elétrico, seja devido ao emprego em aplicações específicas justificadas com estudos de custo-benéfico.

Numa avaliação efetuada, ao nível do "Solar Group" do "US Dept of Energy, U.S.A"., tomando por base sete tecnologias solares, a energia eólica logrou a seguinte classificação:

em 1985: aquecimento solar de imóveis, biomassa, vento.

em 2000: aquecimento solar de imóveis, vento.

em 2020: vento, fotovoltaica.



Evidentemente reconhece-se a não identificação de tal classificação com os interesses da matriz brasileira, no entanto, a posição da energia eólica seria reforçada no nosso caso, em detrimento do aquecimento solar de imóveis, embora que subjetivamente tudo indica que no termo médio (2000) a biomassa tenha um papel mais relevante que o vento.

O potencial das aplicações do aproveitamento eólico é muito vasto, e pode ser resumido em:

- a) usos mecânicos: bombeamento d'água, ar comprimido, refrigeração, aquecimento por atrito, e
- b) usos elétricos: distribuição de eletricidade, carregamento de bateria, processos eletrolíticos, aquecimento por resistência.

De uma maneira geral o vento pode ser integrado em usos por três caminhos:

- a) em sistemas comportando armazenamento, do tipo bateria, bombeamento, ar comprimido, hidrogênio, etc. O custo do armazenamento ditaria a atratividade do sistema. Aonde o armazenamento se constituir em uma peça fundamental ao uso, o custo tenderia ser competitivo,
- b) em sistemas comportando unidades geradoras de reserva, entretanto no planejamento das unidades principais, mas cujas tarefas seriam em momentos de picos (uso em energia de ponta), e
- c) em sistemas comportando unidades que não entram no planejamento das unidades principais, mas cujas tarefas seriam o fornecimento de energia na medida em que haja disponibilidade de vento, poupando-se combustíveis (uso em energia de base).

A prática dos vários programas internacionais nos EEUU, Canadá, Suécia e na experiência iniciada, há cerca de três anos, no Núcleo de Energia (NERG) da UFPb, indica que o caminho a seguir deverá consagrar recursos em programas P&D que articulem as áreas de meteorologia, tecnologia e economia da energia, além de permitir a implantação de unidades eólicas em centrais de ensaios em campo

aberto e em condições de operações reais. De outra maneira, há necessidade de um programa coordenado onde as três áreas indicadas se articulem aos níveis da pesquisa, do desenvolvimento e da operação dos sistemas eólicos.

A indicação dos programas levou em consideração, além destes, a característica fundamental da relação existente entre a produção de energia e o tipo de usuário.

Particularmente, no caso do custo das unidades eólicas, o efeito da "curva de aprendizagem" tem sido ressaltado, enfatizando-se a tendência à redução do custo de unidades menores e de maior número de fabricação (1).

Este trabalho tem como objetivo o desenvolvimento de um sistema de aquisição de dados, baseado em microprocessador, para aplicação no campo de testes de turbinas eólicas do NERG, em Campina Grande, Pb.

O Sistema de Aquisição de Dados do NERG (SADNE) teve seu projeto baseado no microprocessador MC6800 da família MC6800 da MOTOROLA, (2,3,4). O MC6800 foi escolhido para este sistema por: ser um microprocessador de uso geral, ser adequado a interrupções, existir um sistema de desenvolvimento da MOTOROLA (EXORCISER) e documentação em nosso Laboratório de Sistemas Digitais (DEE/CCT/UFPb), etc.

O referido sistema de aquisição de dados tem como finalidades coletar e processar parâmetros sinóticos, elétricos e mecânicos, imprimir relatórios técnicos com os resultados dos testes e armazenar dados num meio de armazenamento de massa. As especificações do SADNE foram estabelecidas por pesquisadores do NERG e do Departamento de Ciências Atmosféricas (DCA/CCT/UFPb).

Os periféricos do sistema são uma TTY, máquina IBM adaptada, utilizada para comunicação entre o operador e o sistema de aquisição de dados, além da impressão de relatórios; um gravador convencional com interface apropriada, para armazenamento da grande massa de dados adquiridos; um RTC programado pelo operador do sistema para sinalizar a MPU para a aquisição de dados.

Seguem-se nos próximos itens uma revisão sobre microprocessadores em geral: sua história, seus termos básicos, suas aplicações, etc., e a descrição do sistema proposto. O desenvolvimento



to do sistema de aquisição de dados foi dividido em 2 partes distintas: o "hardware" e o "software" que são descritos nos capítulos II e III, respectivamente. Por último, os comentários, críticas, sugestões e enriquecimentos do sistema são apresentados no capítulo IV.

## 1.1 - MICROPROCESSADORES EM GERAL

O que causou o súbito "boom" do microprocessador no início da década de 70 ? O desenvolvimento da técnica de integração em larga escala (LSI) alcançou na época um nível de sofisticação tal que tornou-se possível realizar um crescente número de funções computacionais a baixo custo. Com o advento dos circuitos LSI programáveis, o primeiro destes circuitos foi a calculadora, o usuário tornou-se capaz de implementar mudanças em ou acrescentar novas características a um projeto simplesmente mudando um programa em uma memória ROM. Nada de mudanças dispendiosas de máscaras, além da economia gerada pelo armazenamento de sequências de programas em memórias, em vez de implementar estas sequências com "gates" e "flip-flops".

Os microprocessadores atuais representam a revolução causada pelo aparecimento dos processadores LSI que mudaram por completo o desenvolvimento de projetos com computadores. Os microprocessadores de hoje possuem capacidade computacional equiparável a de computadores pequenos e já começam a aparecer poderosos processadores e componentes LSI que excedem em muito os requerimentos das aplicações atuais dos microprocessadores.

Construídos com tecnologias bipolar e MOS cada vez mais aperfeiçoadas, estes rápidos e complexos componentes encontram aplicações também no coração de sistemas baseados em minicomputadores, fomentando mais e mais aplicações e projetos de equipamentos. Os benefícios da tecnologia LSI aplicados a grandes computadores pre dizem uma nova era de controle por computador com alto desempenho e baixo custo.

A Intel Corp. foi a primeira a comercializar o microprocessador e cabe-lhe o crédito pela façanha da utilização do conceito do microprocessador, embora muito se deva a outras companhias e



indivíduos que contribuíram de alguma forma para o desenvolvimento da tecnologia de integração em larga escala. Enquanto isso, na General Electric Co. alguém pensava num circuito LSI que pudesse ser programado pelo usuário. A GE então desenvolveu uma unidade lógica básica (BLU) distribuída em 8 CIs que podia ser usada, sem mudanças, com diferentes programas em vários projetos diferentes de terminais - basicamente o que se faz com os microprocessadores.

Uma dificuldade existente para os projetistas é a natureza dinâmica do desenvolvimento dos microprocessadores. Um engenheiro pode rapidamente perder-se na floresta de novos componentes, equipamentos para testes, suportes de projetos e toda literatura que acompanham este desenvolvimento. Um engenheiro de microprocessadores deve compreender bem: a organização, a operação e o desempenho da MPU do processador; o controle de entrada/saída; a organização e operação de RAMs, ROMs e EPROMs, mais alguns tipos de circuitos de interface como os conversores A/D e D/A; a operação de equipamentos periféricos; a operação de sistemas com interrupção prioritária multi-nível; e a operação de várias famílias lógicas como TTL, p-MOS, n-MOS e C-MOS. Para o projeto de "firmware", este mesmo engenheiro deve também possuir conhecimentos de programação. O projetista deve conhecer bem o "software" para programação em linguagem de máquina e em linguagem assembler, e também microprogramação.

Veja a seguir a definição de alguns termos básicos sobre os microprocessadores:

- a) Unidade de Microprocessamento (MPU): é formada por um grupo de registradores e lógica que formam a unidade lógica aritmética (ALU) e outro grupo de registradores com uma lógica de decodificação associada que formam a unidade de controle. A maioria das MPUs é implementada em um único CI, onde os registradores internos têm capacidade de armazenamento igual ao tamanho de palavra da MPU.
- b) Registrador: é formado por elementos lógicos ("gates", "flip-flops", "shift registers") que associados armazenam números de 4, 8 ou 16 bits. Eles são essencialmente para armazenamento temporário, onde seus conteúdos

dos mudam geralmente entre um ciclo de instrução e o próximo.

- c) Acumulador: é um registrador que realiza operações lógicas/aritméticas com seu próprio conteúdo e um operando e substitui seu conteúdo pelo resultado obtido.
- d) Contador de Programa: é um registrador cujo conteúdo corresponde ao endereço de memória da próxima instrução a ser executada. A contagem geralmente é incrementada de 1 à medida que cada instrução é executada, uma vez que as instruções são normalmente armazenadas sequencialmente.
- e) Registrador de Instrução: é o local de armazenamento do código binário que representa a instrução a ser executada. Geralmente esta instrução representa o conteúdo do endereço designado pelo contador de programa. Entretanto, tanto o conteúdo do registrador de instrução quanto o do contador de programa podem mudar de acordo com os resultados da computação. Claro que isto representa uma das idéias-chaves da computação de programas armazenados: as instruções, como também os dados, quando operadas podem determinar operações subsequentes de acordo com os resultados.
- f) Registrador de Indexação: algumas memórias são organizadas através de indexação. O endereço da próxima instrução pode ser encontrado somando-se os conteúdos do contador de programa e do registrador de indexação. Incrementando-se este registrador de 1 causa-se um desvio do processador para outra parte da memória.
- g) Apontador da Pilha: este registrador é utilizado quando o processador deve atender a um pedido de interrupção gerado por algum periférico. A MPU suspende temporariamente suas operações atuais e antes de executar a tarefa pedida pela interrupção, armazena os conteúdos de seus registradores internos numa parte da memória denominada de pilha. Este armazenamento é feito sempre no topo da pilha com a informação lá existente tendo sido antes empurrada para baixo. A pilha é portanto uma memória do tipo último a entrar/



/primeiro a sair ("last-in/first-out"). O apontador da pilha contém o endereço da próxima locação de memória não utilizada na pilha.

- h) "flag": geralmente é um "flip-flop" onde seu estado lógico indica algum aspecto do estado da MPU. Por exemplo, um "flag" de "carry" é ativado quando uma operação aritmética produz um "carry".

Os microprocessadores causaram um tremendo impacto no vasto espectro da aplicação da eletrônica. Num futuro próximo a utilização e aplicação dos microprocessadores parecem ser limitadas apenas pela imaginação do projetista. Cada vez mais sistemas são projetados, onde a ferramenta básica é o microprocessador, o qual pode ser o simples de 4 bits, o ubíquo de 8 bits ou o de 16 bits.

Seguem-se algumas aplicações dos microprocessadores em:

- . Sistemas de aquisição de dados para:
  - . controle de processos industriais
  - . monitoramento de experimentos
  - . equipamentos de laboratórios
  - . monitoramento do meio ambiente
  - . controle da poluição
  - . controle de esteiras de transporte
  - . controle de balanças industriais
  - . controle de tráfego, etc.
- . Equipamentos inteligentes para comunicação digital:
  - . multiplexadores
  - . conversores de códigos
  - . detetores de erro
  - . controladores de entrada/saída, etc.
- . Produtos comerciais e de consumo:
  - . jogos
  - . brinquedos
  - . balanças
  - . registradoras
  - . receptores de TV inteligentes
  - . computadores personalizados



- . transmissão de TV por cabo
  - . controle de estoque
  - . terminais de venda
  - . terminais inteligentes, etc.
- . Automóveis (controle da poluição e economia de combustível):
- . controle da taxa ar/combustível
  - . controle do tempo de ignição
  - . controle da taxa de exaustão dos gases, etc.
- . Instrumentos inteligentes:
- . analisadores lógicos
  - . testadores de CIs
  - . multitesters
  - . analisadores de assinaturas
  - . instrumentos "virtuais", etc.

## 1.2 - DESCRIÇÃO DO SISTEMA PROPOSTO

O campo de testes de turbinas eólicas do Núcleo de Energia Eólica, em Campina Grande, é composto de 5 torres micro-meteorológicas e 3 bases para turbinas eólicas do tipo Darrieus.

Em cada torre micro-meteorológica estão afixados 6 anemômetros e 1 sensor de direção de vento.

Em cada base para turbinas estão afixados os seguintes tipos de instrumentos: voltímetro, amperímetro, wattímetro, tacômetro e torquímetro.

Todos os instrumentos utilizados nos testes terão as suas saídas com um máximo de 4 dígitos decimais representados digitalmente em código binário ou BCD com um máximo de 16 bits por instrumento.

A informação digital adquirida nos instrumentos pelo sistema de aquisição de dados deve ser processada pela CPU do sistema para que se obtenha a representação real do valor da grandeza medida.

Os anemômetros giram com o vento. Um imã colocado no seu eixo gera um pulso toda vez que passa em frente a um sensor. Assim, num intervalo de tempo T a informação produzida é número.

de pulsos contados por um conjunto de contadores. A cada intervalo de tempo  $T$  o sistema deve ler um número que varia entre 0000 e 9999. Durante o processamento da informação adquirida, a CPU deve traduzir o número de pulsos de cada anemômetro para velocidade média no intervalo  $T$ . Em seguida, devem ser calculados para cada anemômetro:

- a) a velocidade média no intervalo de tempo  $t = NT$ , onde  $N$  é o número de medições,
- b) o desvio padrão,
- c) o momento centrado de terceira ordem e
- d) a potência média no intervalo de tempo  $t = NT$ .

Os indicadores de direção do vento estão sempre seguindo a direção do vento. No seu eixo existem 16 sensores que indicam a direção do vento naquele instante. A representação é feita em código "1 out of 16" onde os  $360^{\circ}$  são divididos em 16 sensores representados diretamente pelo código mencionado. Durante o processamento, a CPU deve traduzir o código do sensor de direção do vento para a direção em graus.

O NERG executará numa primeira etapa dois tipos de testes utilizando o sistema de aquisição de dados. Um dos testes consiste no estudo dos regimes naturais do vento, no qual são realizadas medições periódicas de parâmetros em intervalos pré-fixados. O outro teste consiste em determinar as características de performance de uma turbina Darrieus.

Em geral, apenas uma ou duas torres funcionarão simultaneamente. O funcionamento conjunto das cinco torres ocorrerá durante uma experiência específica e completa para levantamento das características de uma turbina.

Os tipos de testes e a aleatoriedade da sequência na qual eles são realizados, a qual é por especificação a critério do operador do sistema, impõem uma característica de modularidade no projeto do "hardware" e do "software" do sistema. Em termos práticos, isto implica em que cada torre e cada base para turbina, seja tratada independentemente, possibilitando um critério para utilização das mesmas definido pelo operador do sistema.



O intervalo de tempo T, entre as aquisições de dados e o número de aquisições N a serem realizadas são também programados pelo operador do sistema e assumem os seguintes valores:

T = 1, 2, 4 e 30 segundos

1, 2, 5 e 10 minutos

N = 1, 10 e 100

A densidade do ar, um parâmetro que depende das condições atmosféricas durante a experiência, deve ser fornecida ao sistema pelo operador para o cálculo da potência média do vento.

Após o processamento dos dados registrados, o sistema deve imprimir um relatório de pesquisa com o formato apresentado no apêndice V. Todos os dados registrados brutos devem ser armazenados em fita cassete.

O sistema foi implementado tendo em vista estas especificações. Possíveis modificações nestas: como na operacionalidade do sistema, relatórios, instrumentos, etc., serão permitidas devido à flexibilidade oferecida pelos sistemas baseados em microprocessador.



## CAPÍTULO II

### "HARDWARE"

#### 2.1 - ESTRUTURA DO SISTEMA

Escolheu-se uma arquitetura para o sistema de aquisição de dados que facilitasse a manutenção e a operação do sistema e que tornasse possível a implementação de alguma expansão futura que se faça necessária. Para tanto, optou-se por uma arquitetura com característica modular.

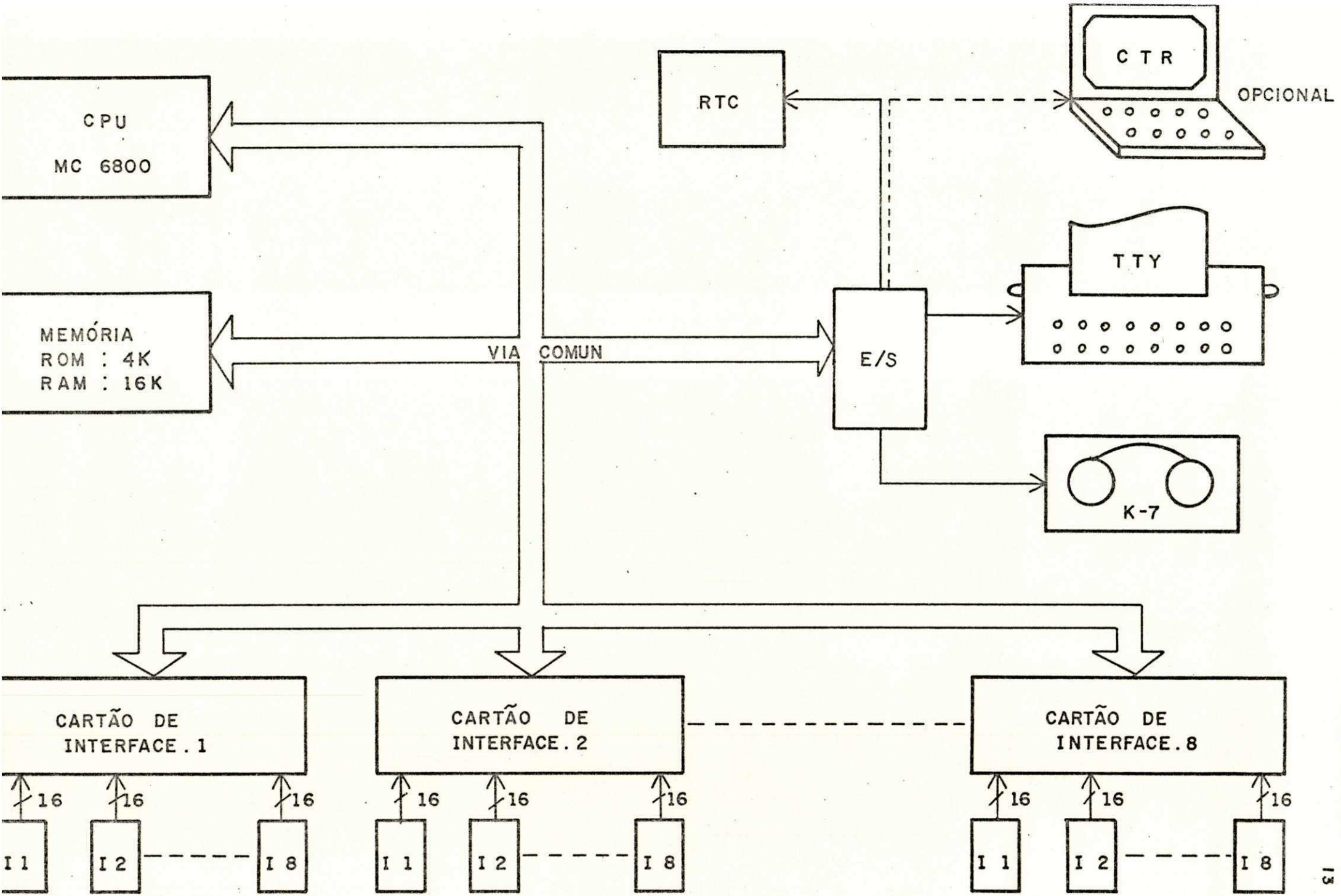
A modularidade caracteriza-se pelo fato de que:

- a) a CPU, a memória do sistema, as interfaces para E/S e as interfaces para instrumentos foram implementadas em cartões diferentes.
- b) o projeto dos cartões de interface para instrumentos permite a conexão de qualquer dos instrumentos do sistema em qualquer uma das 8 posições do cartão e
- c) qualquer um dos cartões do sistema pode ser conectado ao armário que contém o sistema de aquisição de dados sem qualquer ordem de conexão.

A arquitetura do SADNE, que teve seu projeto baseado na família M6800 da MOTOROLA, (2,3), é mostrada de maneira simplificada, numa organização de "via comum" ("common bus"), na figura 2.1.

O sistema de microcomputador é composto pelos seguintes blocos:

- a) um cartão da unidade central de processamento - CPU





- ("Central Processing Unit"),
- b) um cartão de memórias programáveis PROM ("Programmable Ready Only Memory"),
- c) dois cartões de memórias de acesso aleatório RAM ("Random Access Memory"),
- d) um cartão de interface para dispositivos de E/S (Entrada/Saída) que são uma TTY ("Teletypewriter"), a qual é uma máquina IBM adaptada; um terminal de vídeo (opcional); um gravador de fita cassete e os "displays" do relógio RTC ("Real Time Clock"), o qual foi implementado em "software",
- e) oito cartões de interface para conexão dos vários instrumentos ao sistema de aquisição de dados e
- f) fonte de alimentação que fornece  $\pm 5$  Vdc e  $\pm 12$  Vdc.

## 2.2 - UNIDADE CENTRAL DE PROCESSAMENTO

A figura 2.2 mostra o cartão da CPU em diagrama de blocos, o qual compõe-se de:

- a) microprocessador MC6800 da MOTOROLA - MPU ("Microprocessing Unit"),
- b) circuito de "clock" de 1 MHz,
- c) circuito de "clock" de 1 Hz,
- d) "buffers" da via de endereços e sinais de controle,
- e) "transceiver" da via de dados e
- f) circuito para "reset" automático/manual.

A figura 1 do apêndice I mostra a implementação da CPU.

### 2.2.1 - BREVE DESCRIÇÃO DO MICROPROCESSADOR MC6800 DA MOTOROLA-MPU

O MC6800 é um microprocessador de 8 bits que é compatível com TTL e requer apenas uma única fonte de alimentação de 5 volts. Tem capacidade de endereçamento de 65.356 locações de memória com sua via de endereços de 16 bits. Sua via de dados é bi-direcional

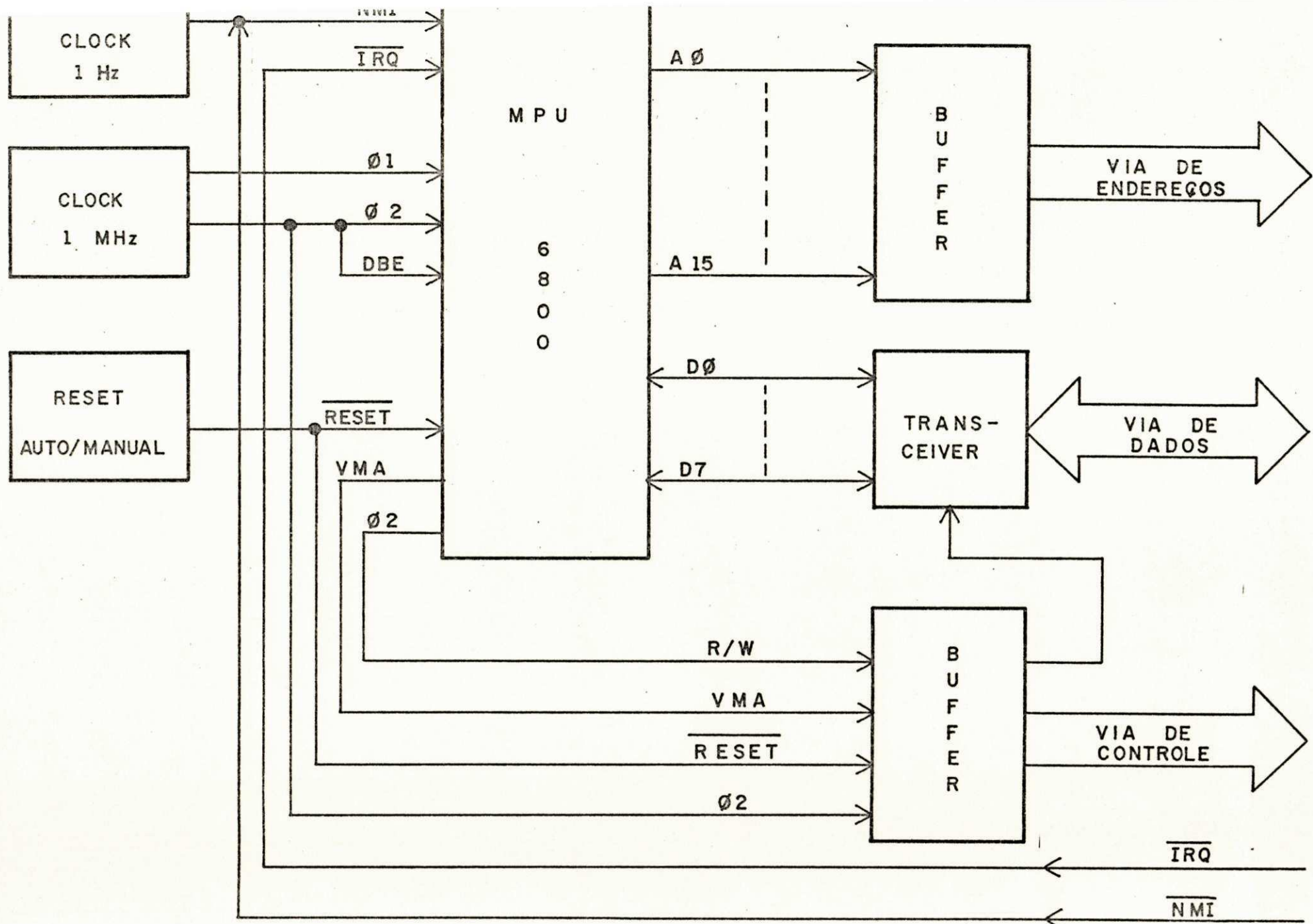


FIG. 2.2- CARTÃO DA CPU.



e tem controle de "three-state". Possui um conjunto de 72 instruções diferentes que inclui instruções para: operações aritméticas binárias e decimais, operações lógicas, operações de deslocamento e giro, operações de armazenamento e carregamento, operações de pulso e desvio condicionais e incondicionais e manipulações com a pilha, o indexador e interrupções. Possui seis registradores internos e 4 vetores para atendimento de interrupções que podem ser geradas por "hardware" e "software".

A organização interna do MC6800, a descrição de seus pinos e seu conjunto de instruções são apresentados no apêndice II.

#### 2.2.1.1 - SINAIS DA MPU

A MPU requer que alguns sinais de controle e temporização sejam providenciados para que possa funcionar a contento. Também, a própria MPU fornece alguns sinais, que juntamente com os gerados externamente, servem para determinar o estado da mesma. O SADNE, sendo um sistema dedicado, não utiliza todos os sinais existentes na MPU. Segue-se uma descrição sucinta dos sinais utilizados neste sistema.

- a) VIA DE DADOS (D0 a D7) - São usados oito pinos para esta via bi-direcional que transfere dados de e para as memórias e periféricos. Tem controle de "three-state" e capacidade para alimentar uma carga padrão TTL. Devido ao grande porte do sistema, torna-se necessário o uso de "transceivers" para aumentar a capacidade de alimentação ("fanout") da via de dados. Utilizou-se o CI 8216 (5), quatro "transceivers" bi-direcionais para dados, para tal. O controle de "three-state" dos "transceivers" é feito por um "AND" lógico dos sinais  $\overline{\text{O2}}$ , VMA e R/W.
- b) VIA DE ENDEREÇOS (A0 a A7) - São usados 16 pinos para esta via uni-direcional. Como a via de dados, esta via tem controle de "three-state" e capacidade para alimentar uma carga padrão TTL. Assim, torna-se necessário o uso de "buffers" para a via de endereços e optou-se pelo uso do CI 8T95 ("hex buffers") (6), para este fim, devido a sua grande capacidade de alimentação e sua alta velocidade.

- c) SISTEMA DE "CLOCK" ( $\emptyset 1$  e  $\emptyset 2$ ) - 2 pinos são usados para um "clock" de duas fases não superpostas que é gerado por um oscilador a cristal com frequência de 1MHz. Este "clock" foi implementado com o CI 74124 (8), dois VCOs ("Voltage-Controlled Oscillators"). A fase  $\emptyset 2$  é também aplicada às entradas de habilitação ("enable") ou de seleção ("chip select") dos demais componentes do sistema (memórias, PIAs e ACIAs) com a finalidade de garantir que os mesmos sejam habilitados apenas quando a via de endereços e o sinal VMA estiverem ativos.
- d)  $\overline{\text{RESET}}$  - Esta entrada é utilizada para "reset" e inicialização da MPU nas condições de energização inicial do sistema ("start-up") ou de energização após uma quebra do fornecimento da energia do sistema ("power down condition") ou ainda, para reinicialização do sistema, sempre que desejado pelo operador, através de uma chave externa.

Para a inicialização do sistema, a linha  $\overline{\text{RESET}}$  deve ser mantida no estado lógico baixo, por pelo menos 8 ciclos do "clock" do sistema, depois que a tensão de alimentação atinge sua tensão mínima de operação (4,75 Vdc). O circuito para "reset" automático foi implementado com o temporizador LM555 (7) que garante este atraso mínimo necessário para a inicialização do sistema.

- e) READ/WRITE (R/W) - Este sinal refere-se à via de dados e define se a MPU está no modo de leitura ou modo de escrita em cada ciclo. O estado normal deste sinal é o nível lógico alto (modo de leitura).
- f) VALID MEMORY ADDRESS (VMA) - Este sinal indica aos componentes do sistema que existe um endereço válido na via de endereços. Quando no estado lógico baixo inibe a transferência de dados.
- g) DATA BUS ENABLE (DBE) - É o sinal de controle de "three-state" da via de dados da MPU. Normalmente é uma derivação de fase 2 do "clock" ( $\emptyset 2$ ).
- h)  $\overline{\text{INTERRUPT REQUEST}}$  ( $\overline{\text{IRQ}}$ ) - Este sinal é gerado pelas



PIAs, ACIAs ou algum outro dispositivo externo com a finalidade de notificar a MPU um pedido de serviço. Este pedido só será atendido se o bit da máscara de interrupção do registrador do código de condição - CCR ("Condition Code Register") não estiver ativo.

- i) NON-MASKABLE INTERRUPT (NMI) - Difere do sinal  $\overline{IRQ}$  no sentido de que o bit da máscara de interrupção do CCR não afeta a MPU. Esta interrupção é sempre atendida pela MPU. Neste sistema, este sinal é gerado por um "clock" de 1 Hz para atualização do RTC, que foi implementado em "software". Assim, a cada segundo, a MPU atende a uma interrupção não mascarável prioritária, gerada por um circuito oscilador ( $f = 1\text{Hz}$ ) implementado com o temporizador LM555 (7).

#### 2.2.1.2 - REGISTRADORES DA MPU

A MPU possui seis registradores internos para utilização por parte do usuário, dos quais três são de 2 bytes (16 bits) e os outros três são registradores de 8 bits. Segue-se uma rápida descrição destes registradores:

- a) CONTADOR DE PROGRAMA - PC ("Program Counter") - O contador de programa é um registrador de 2 bytes (16 bits) que aponta para o endereço do programa atual.
- b) APONTADOR DA PILHA - SP ("Stack Pointer") - O apontador da pilha é o registrador, também de 2 bytes, que contém o endereço da próxima locação de memória livre de uma pilha externa. Esta pilha é normalmente uma porção da memória RAM do sistema e pode ter o número de locações (endereços) que sejam necessários.
- c) REGISTRADOR DE INDEXAÇÃO - IX ("Index Register") - É o último dos registradores de 16 bits e é utilizado para armazenar dados ou um endereço (de 16 bits) de memória para aplicações de endereçamento de memórias do modo indexado.
- d) ACUMULADORES (ACCA e ACCB) - A MPU possui 2 acumuladores (registradores) de 8 bits que são usados para armazenar operandos e resultados fornecidos pela unidade

lógica aritmética - ALU - ("Arithmetic Logic Unit").

- e) REGISTRADOR DO CÓDIGO DE CONDIÇÕES - CCR ("Condition Code Register") - Este registrador indica os resultados de uma operação realizada pela ALU. O CCR é um registrador de 8 bits onde o bit 3 indica se a operação resultou um número negativo (N), o bit 2 indica se este resultado foi zero (Z), o bit 1 indica ocorrência de "overflow" (V), o bit 0 indica propagação de "carry" (C) e o bit 5 indica propagação do meio "carry" (H). Estes bits do CCR são utilizados como condições de teste para aplicações com instruções de desvio ou pulo condicional. O bit 4 é o bit da máscara de interrupção (I). Os bits 6 e 7, não utilizados, são uns.

## 2.3 - MEMÓRIAS

A figura 2.3.1 mostra o mapa de memórias, as quais foram distribuídas do modo abaixo:

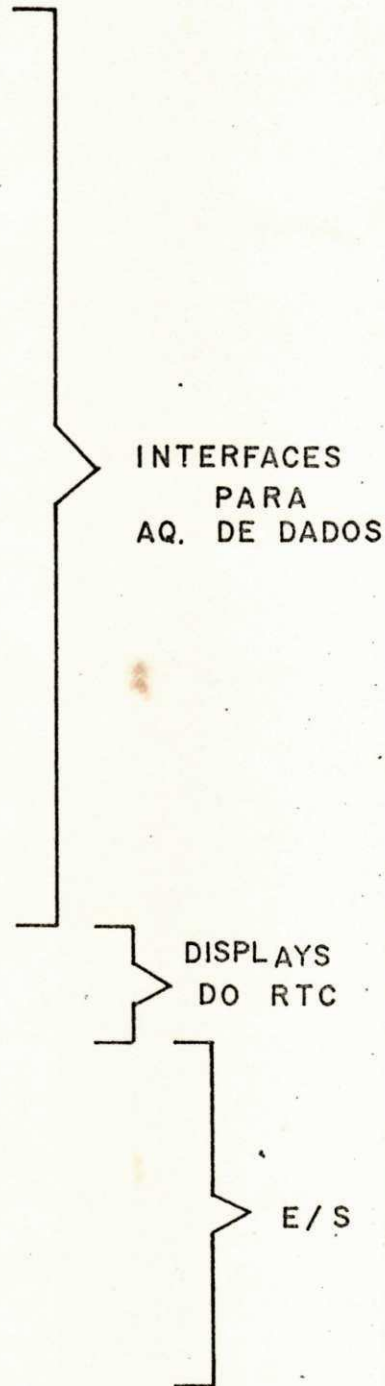
- PROM: 4K bytes de memória destinados ao monitor básico do sistema, programas de aquisição e processamento de dados, implementação em "software" do RTC, monitor do gravador de fita cassete, oito locações de memória para os vetores de atendimento a interrupções e armazenamento de tabelas que contêm mensagens de erro e identificação de programas, perguntas e tabelas de conversão de unidades.
- RAM: 16K bytes de memória destinados para armazenamento da grande massa de dados adquiridos, dos vetores de atividade do sistema e dos parâmetros definidos pelo operador e da informação processada.
- E/S: 44 locações de memória para endereços dos componentes (PIAs e ACIAs) utilizados nas interfaces para periféricos e interfaces para aquisição de dados.

A figura 2.3.2 mostra como foram distribuídos os 16 bits da via de endereços para selecionar estes componentes do sistema com um "hardware" mínimo necessário.

O cartão de memórias PROM pode ser visto em diagrama de



FFFF	PROM
F000	
8203	PIA 8
8200	
8103	PIA 7
8100	
8083	PIA 6
8080	
8043	PIA 5
8040	
8023	PIA 4
8020	
8013	PIA 3
8010	
800B	PIA 2
8008	
8007	PIA 1
8004	
4801	PIA 10
4800	
4401	PIA 9
4400	
4201	ACIA 2
4200	
4101	ACIA 1
4100	
3FFF	
	RAM
0000	



NOTA 1. ENDEREÇ  
EM HEXADEC

BITS DE ENDER.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PROM	1	1	1	1	X	X	X	X	X	X	X	X	X	X	X	X
PIA 8	1	0	0	0	0	0	1	0	0	0	0	0	0	0	X	X
PIA 7	1	0	0	0	0	0	0	1	0	0	0	0	0	0	X	X
PIA 6	1	0	0	0	0	0	0	0	1	0	0	0	0	0	X	X
PIA 5	1	0	0	0	0	0	0	0	0	1	0	0	0	0	X	X
PIA 4	1	0	0	0	0	0	0	0	0	0	1	0	0	0	X	X
PIA 3	1	0	0	0	0	0	0	0	0	0	0	1	0	0	X	X
PIA 2	1	0	0	0	0	0	0	0	0	0	0	0	1	0	X	X
PIA 1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	X	X
PIA 10	0	1	0	0	1	0	0	0	0	0	0	0	0	0	X	X
PIA 9	0	1	0	0	0	1	0	0	0	0	0	0	0	0	X	X
ACIA 2	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	X
ACIA 1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	X
RAM	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X

NOTA 1. X = NÃO IMPORTA (DON'T CARE)



blocos na figura 2.3.3. O decodificador de endereços foi implementado com o CI 74155 (8), dois decodificadores/demultiplexadores de 2 linhas para 4 linhas. Utilizou-se a memória programável 2708 (9), que é estática, tem organização de 1024 palavras de 8 bits e que além de ser compatível com a família M6800, pode ser programada pelo EXORciser, sistema de desenvolvimento de programas da MOTOROLA existente em nosso Laboratório de Sistemas Digitais. A implementação deste cartão pode ser vista na figura 2 do apêndice I.

A figura 2.3.4 mostra, em diagrama de blocos, um dos dois cartões de memórias RAM, no qual foram distribuídos 8K bytes de memória. Para implementação do decodificador de endereços, utilizaram-se o CI 7442 (8), decodificador de 4 linhas para 10 linhas (1 de 10) e o CI 7430 (8), gate NAND positivo de 8 entradas. O primeiro bloco de 8K bytes de memórias RAM é selecionado quando o bit A13 da via de endereços é igual a zero e o segundo bloco é selecionado para A13 igual a um. Utilizou-se a memória estática 2114 (5), com organização de 1024 palavras de 4 bits, a qual satisfaz aos requerimentos de tempo de acesso da MPU. A implementação do cartão 1 de memória RAM (8K bytes) é vista na figura 3 do apêndice I.

## 2.4 - INTERFACES DE AQUISIÇÃO DE DADOS

Cada cartão de interface para aquisição de dados, em número de oito, foi projetado para conectar 8 instrumentos com 16 bits de saída cada ao sistema. Com o objetivo de minimizar os custos de implementação destes cartões, os 8 instrumentos têm suas saídas multiplexadas na entrada de uma PIA ("Peripheral Interface Adapter", veja item 2.5 para definição da PIA) MC6820 (3), a qual por sua vez está conectada à MPU. Utilizou-se o multiplexador 74151 (8) (8 para 1) para tal.

A figura 2.4 mostra um destes cartões em diagrama de blocos. O controle de seleção de dados nos multiplexadores é feito por "software" através de três pinos da PIA9 (fig. 2.5). Estes sinais encontram-se disponíveis na via comum do sistema. Para executar uma instrução de leitura de um instrumento qualquer, a MPU deve antes selecionar o endereço nos multiplexadores que

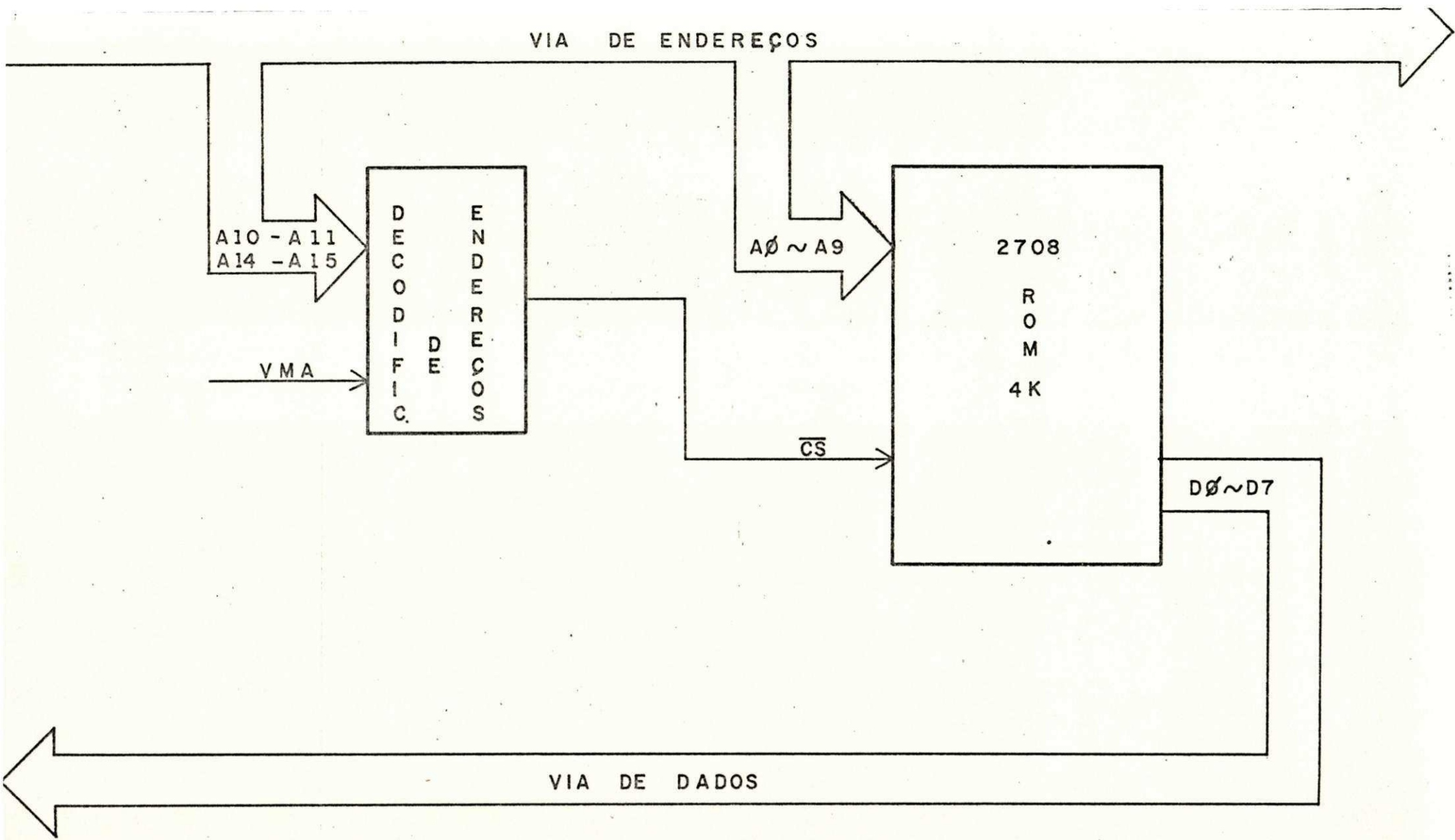
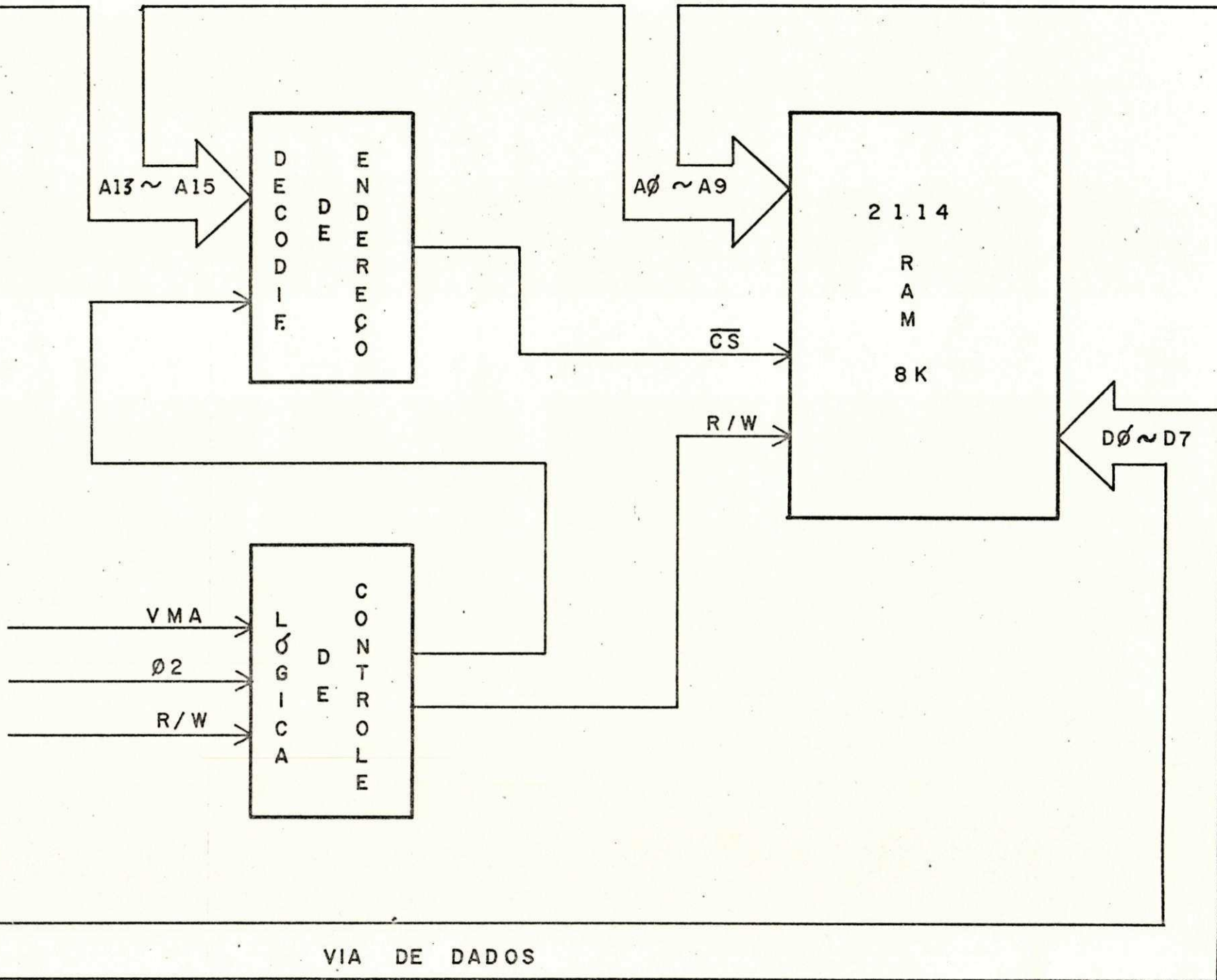


FIG. 2.3.3 - - CARTÃO DE MEMÓRIA PROM.



VIA DE ENDEREÇOS



VIA DE DADOS

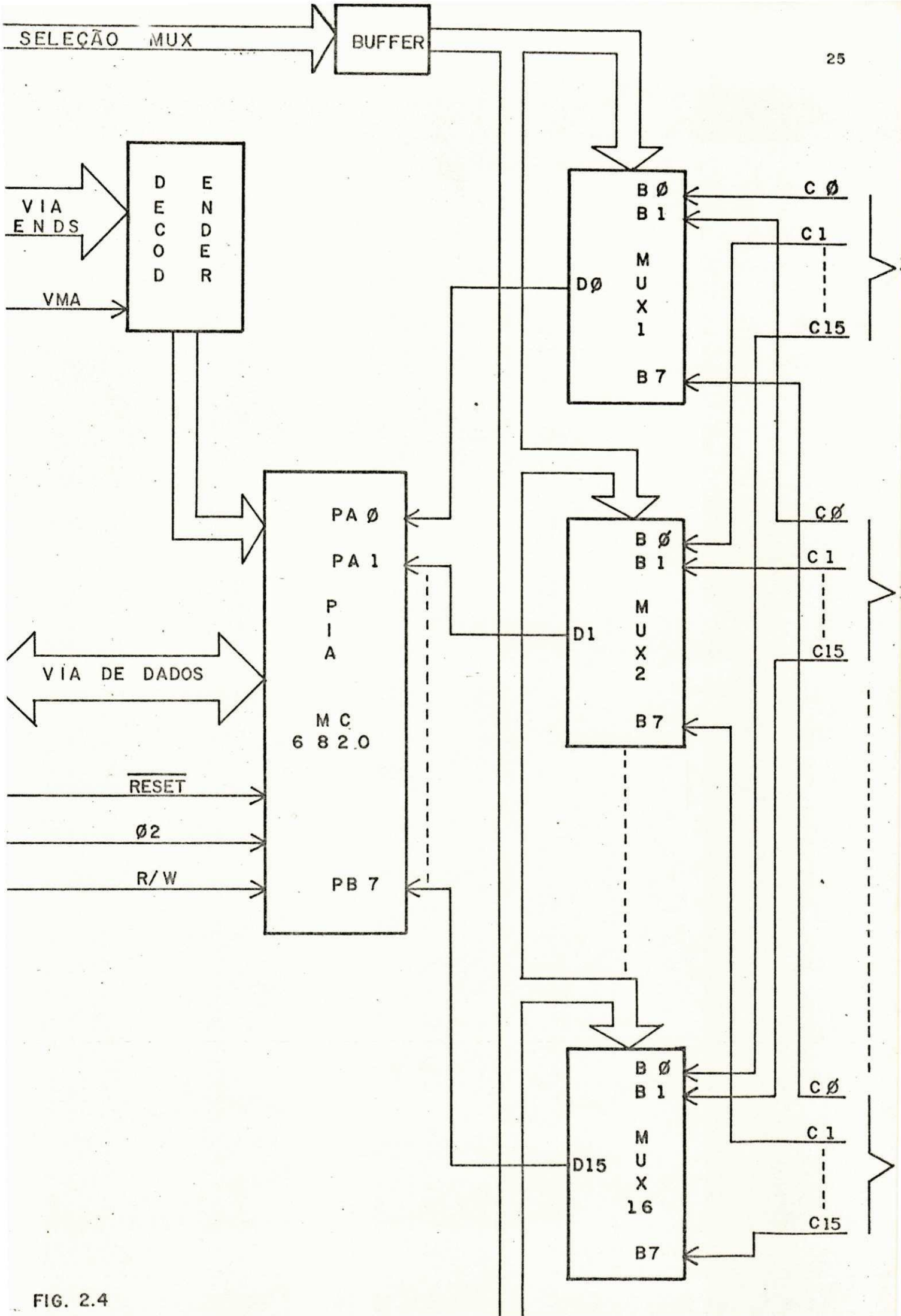


FIG. 2.4



corresponde a este instrumento (veja fig. 4 do apêndice I para implementação de um destes cartões).

Os instrumentos podem ser conectados à interface em qualquer uma das 8 posições existentes, mas devido à estrutura do "software" esta ordem uma vez estabelecida, não mais deve mudar. Assim, foi sugerido que nas torres esta ordem seja como segue-se: as 6 primeiras posições para anemômetros e a 7a. posição para o diretor de vento (ainda não foi definido o instrumento para a 8a. posição).

## 2.5 - ENTRADA/SAÍDA

Os componentes para interface de Entrada/Saída da família M6800 da MOTOROLA são a PIA ("Peripheral Interface Adapter") MC6829 (2), uma interface em paralelo, e a ACIA ("Asynchronous Communication Interface Adapter") MC6850 (2), uma interface em série assíncrona. Estes componentes permitem que uma variedade de periféricos sejam conectados à MPU com um mínimo de lógica adicional e uma programação relativamente simples. Veja no apêndice III as descrições dos pinos e arquiteturas internas da PIA e da ACIA.

A PIA MC6820 prover um meio universal flexível para conexão de periféricos em paralelo à MPU. Internamente, a PIA é dividida em duas configurações simétricas independentes de registradores. Cada metade possui três registradores: um registrador de saída de dados, um registrador de controle e um registrador de direção de dados.

O fluxo de dados entre a MPU e a PIA é realizado através da própria via de dados do sistema. A direção deste fluxo é controlada pela MPU através da entrada R/W da PIA. A parte da PIA em conexão com a MPU inclui também três linhas de seleção de pastilha, CS0, CS1 e  $\overline{\text{CS2}}$ , para a seleção de uma determinada PIA. Existem duas entradas de endereço, RS0 e RS1, que usadas juntamente com o bit 2 do registrador de controle da PIA, selecionam registradores específicos internos da PIA. A MPU pode ler ou escrever nos registradores internos da PIA, selecionando-a através da via de endereços do sistema usando estas cinco linhas de entrada da PIA e o sinal R/W. Cada PIA é vista pe

la MPU como sendo quatro locações de endereços que são tratadas da mesma maneira que uma memória RAM qualquer. A MPU também fornece à PIA um sinal de temporização através da sua entrada de habilitação ("Enable"). Este pulso de habilitação é usado para condicionar a circuitaria interna de controle de interrupção da PIA. Como toda transferência de dados ocorre durante a fase  $\phi_2$  do "clock", este pulso é normalmente o sinal  $\phi_2$ .

A parte da PIA em conexão com o periférico inclui duas vias de dados bi-direcionais de 8 bits cada (PA0 a PA7 e PB0 a PB7) e quatro linhas de controle de interrupção (CA1, CA2, CB1 e CB2).

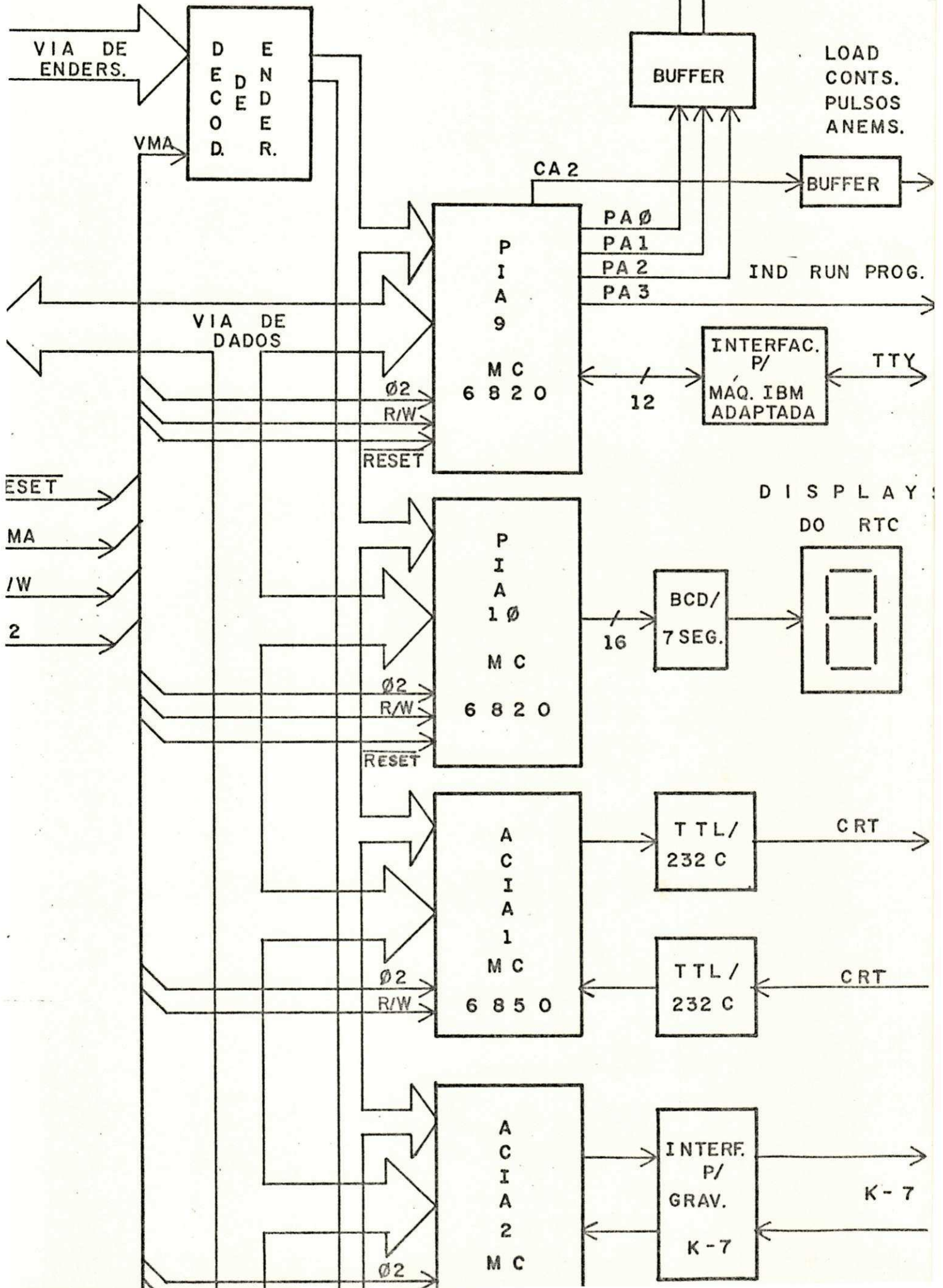
A ACIA MC6850 prover meios eficientes para conexão de periféricos (Modems, terminais de vídeo, TTYs, etc) que requerem uma formatação dos dados em série e assíncrona. Possui quatro registradores internos: registrador de controle, registrador de "status", registrador para transmissão de dados e registrador para recepção de dados.

Como acontece com a PIA, o fluxo de dados entre a MPU e a ACIA é realizado através da via de dados do sistema e tem sua direção controlada pela MPU através da entrada R/W da ACIA. Existem três linhas de seleção (CS0, CS1 e  $\overline{CS2}$ ) para endereçamento de uma ACIA particular e uma linha adicional de endereço para selecionar registradores internos específicos da ACIA. Cada ACIA é vista pela MPU como sendo duas locações de memória que são tratadas da mesma maneira que uma memória RAM qualquer. O sinal  $\phi_2$  é usado para temporização da ACIA também.

A parte da ACIA em conexão com o periférico inclui duas linhas para dados seriais (transmissão e recepção) e três linhas de controle. Duas entradas para "clock" são disponíveis para fornecer taxas de "clock" individuais para o receptor e o transmissor da ACIA.

A figura 2.5 mostra o cartão de Entrada/Saída do sistema de aquisição de dados. Para efetuar a comunicação entre o operador e o sistema, utiliza-se uma TTY que é uma máquina IBM adaptada para tal. Esta máquina é conectada à MPU em paralelo através de 12 bits da PIA9 (fig. 2.5). A impressão de relatórios





é também uma função da máquina IBM adaptada.

Prevendo-se uma possível aquisição de um terminal de vídeo para o SADNE, existe no cartão de E/S uma ACIA (ACIA2) para conexão do mesmo à MPU. O terminal de vídeo funcionará a uma velocidade de transmissão de 300 bauds.

Utiliza-se um gravador de fita cassete para armazenamento da grande massa de dados adquiridos e não processados (é interesse do NERG arquivar esta massa de dados brutos para futuro levantamento meteorológico da região). O mesmo está conectado à MPU através da ACIA1 (fig. 2.5) e funciona a uma velocidade de transmissão, também de 300 bauds.

O RTC, implementado em "software" e que serve tanto para sinalizar a MPU para a aquisição de dados como para fornecer horário e data de início e término de aquisição de dados e impressão de relatórios, fornece também ao operador, através de "displays" conectados à MPU através da PIA10 (fig. 2.5), o horário com precisão de minutos.

A linha de controle CA2 da PIA9 (fig. 2.5) é utilizada para enviar o comando "LOAD" que controla as interfaces dos anemômetros. Para cada anemômetro existe uma interface que foi implementada com um conjunto de contadores que contam os pulsos vindos dos anemômetros durante um intervalo de tempo determinado. O comando "LOAD" transfere essa informação dos contadores de pulsos para um conjunto de "latches" e em seguida executa o "reset" dos contadores de pulsos, preparando-os desta maneira para nova contagem de pulsos. O sistema de aquisição de dados adquire nos "latches" a informação sobre os anemômetros a ser armazenada e processada.



## CAPÍTULO III

### "SOFTWARE"

#### 3.1 - ESTRUTURAÇÃO DO "SOFTWARE"

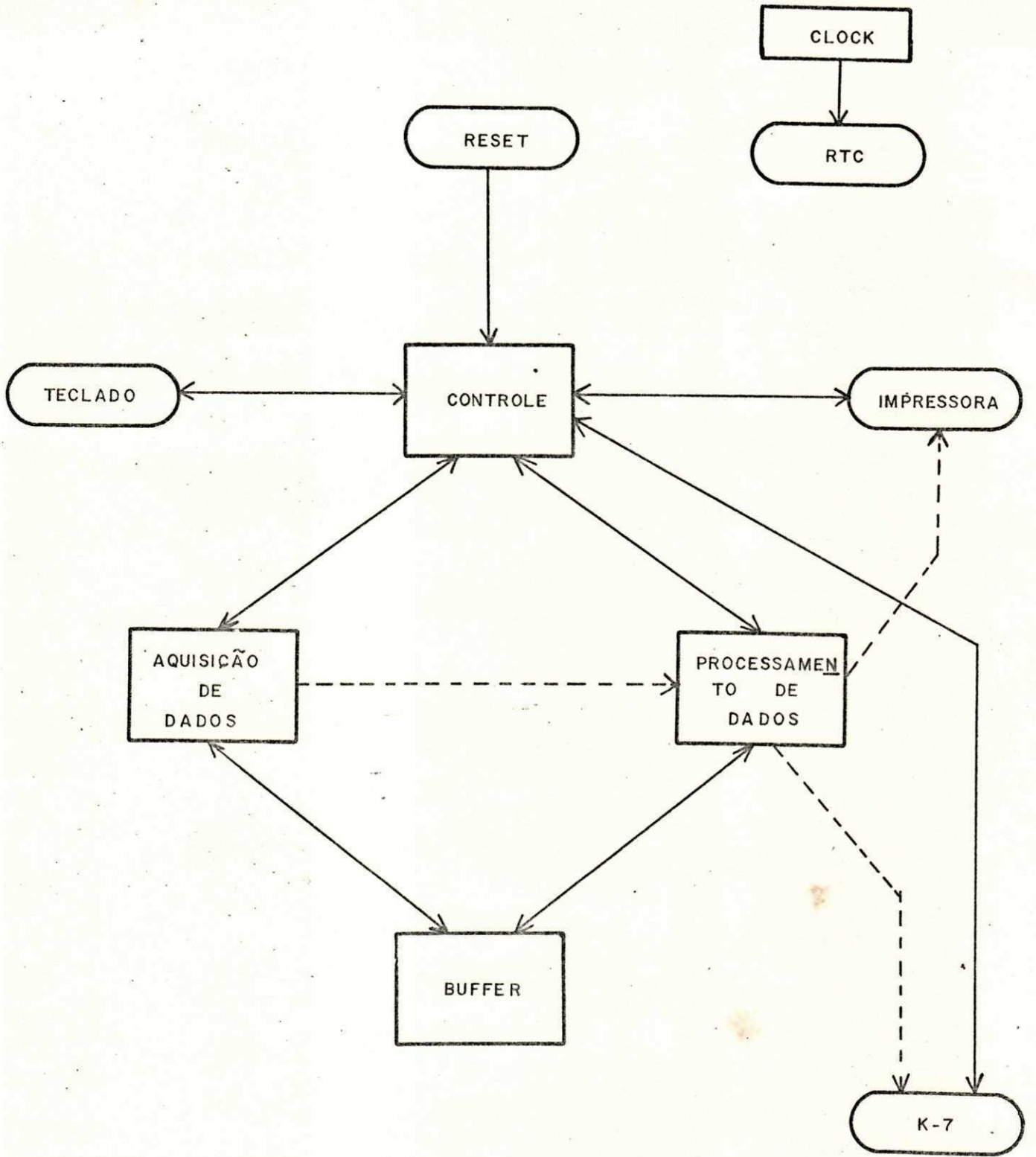
Para este sistema de aquisição de dados baseado em microprocessador, desenvolveu-se um "software" básico, um conjunto de programas coordenados por um programa de controle com a finalidade de coordenar a MPU durante a execução das diferentes tarefas ("TASKS") do sistema. A figura 3.1.1 mostra o "software", estruturado numa maneira simplificada, em diagrama de blocos, (10) e a figura 3.1.2 mostra o "software" mais detalhadamente.

A primeira tarefa do "software" é a inicialização do sistema, a qual é executada pelo Programa de Atendimento ao Sinal  $\overline{\text{RESET}}$  (PARES), que em seguida entrega o comando do sistema ao Programa de Controle (PC). O PC é quem coordena a execução das diversas tarefas pelos outros programas.

O Programa de Entrada de Parâmetros (PEP) executa a função de interagir ou conversar com o operador, requisitando informações necessárias ao bom funcionamento do sistema. Através deste programa, o operador define os parâmetros do sistema por meio de uma linguagem de simples utilização.

O Programa de "Reset" dos Parâmetros (PRP) tem como função "limpar" as posições de memória onde são armazenados os parâmetros fornecidos pelo operador através do PEP. O Programa de "Status" dos Parâmetros (PSP), tem como finalidade fornecer ao operador uma listagem do estado atual destes parâmetros.

O Gerente da Aquisição de Dados (GAD) é responsável pela



———— MODO CPO  
- - - - MODO AUTO



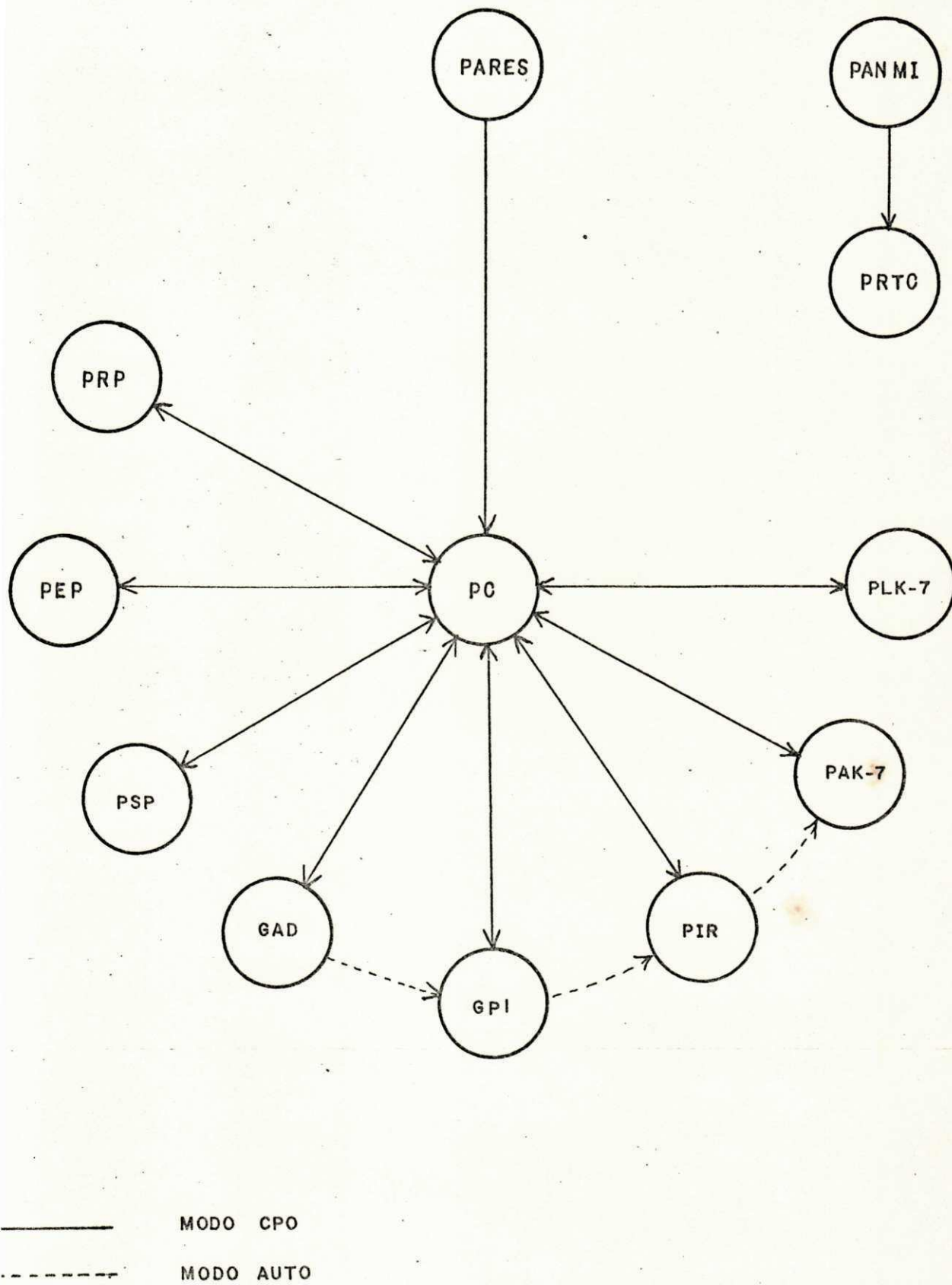


FIG. 3.1.2 - Estruturação do "Software"

coordenação e supervisão da aquisição de dados. Os dados adquiridos dos instrumentos especificados (das torres ou bases selecionadas) são armazenados sequencialmente.

O Gerente do Processamento da Informação (GPI) coordena o processamento destes dados adquiridos e monta uma tabela com o resultado do processamento, a qual é então formatada e impressa pelo Programa de Impressão de Relatórios (PIR).

O Programa de Armazenamento em Fita Cassete (PAK-7) transporta a grande massa de dados adquiridos para fita cassete, onde o arquivo pode ser mantido para consultas futuras. O Programa de Leitura de Fita Cassete (PLK-7) tem como função carregar este arquivo na memória do sistema.

O Programa de Atendimento à Interrupção NMI (PANMI) é responsável pela atualização do RTC, no sentido de que está encarregado de chamar o Programa do RTC (PRTC), a cada segundo, para que o mesmo atualize seu horário e calendário de tempo real.

Segue-se uma descrição mais detalhada destes programas e de algumas subrotinas comuns a estes programas.

### 3.2 - PROGRAMA DE ATENDIMENTO AO SINAL RESET - PARES

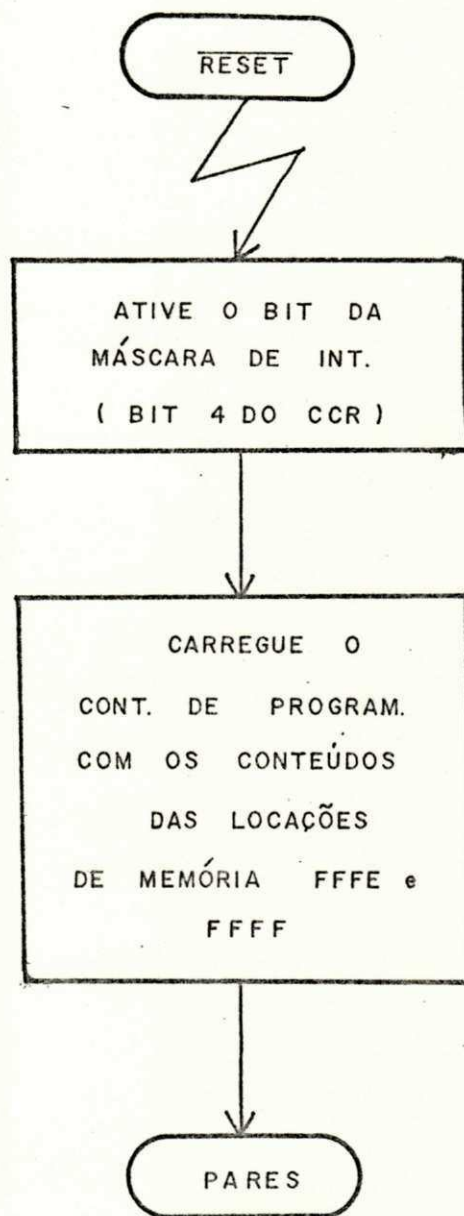
Este programa é responsável pela execução da inicialização do sistema quando da condição de energização do mesmo ou da sua reinicialização, que pode acontecer a qualquer instante após sua energização, a pedido externo do operador (Veja fig. 3.2.2).

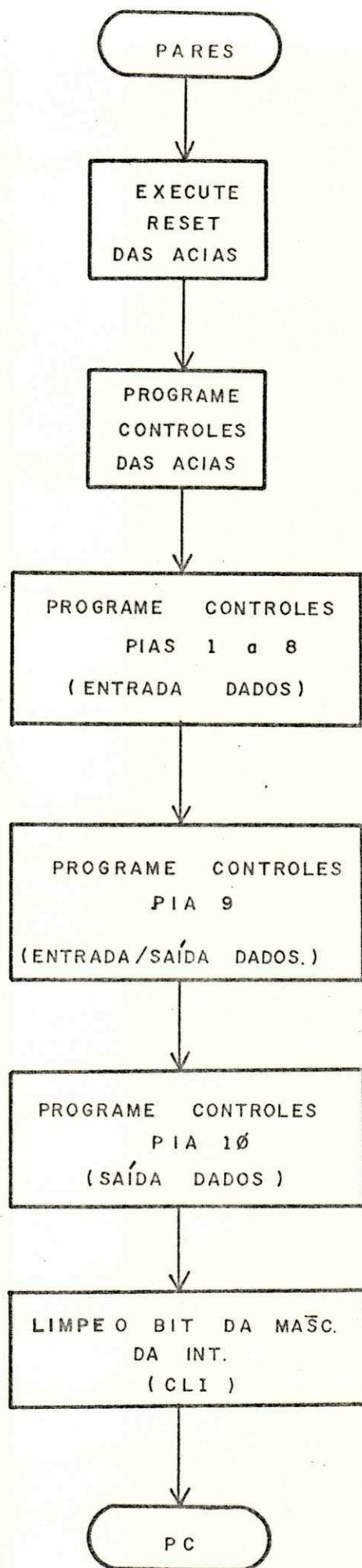
Esta inicialização consiste em:

- a) preparar ou programar os dispositivos de interface de entrada/saída (PIAs e ACIAs) para que funcionem de acordo com os requisitos deste sistema de aquisição de dados;
- b) carregar o apontador da pilha com seu valor inicial e
- c) limpar o bit da máscara de interrupção (bit 4 do CCR).

Ao detetar o sinal RESET a MPU executa a sequência vista na figura 3.2.1 onde o primeiro passo é ativar o bit da máscara de



FIG 3 2 1 - Sequência da Interrupção RESET

FIG. 3.2.2 - Fluxograma do Programa de Atendimento ao Sinal RESET



interrupção com a finalidade de evitar que alguma interrupção  $\overline{IRQ}$  aconteça durante o processo da inicialização do sistema. Em seguida, a MPU carrega o Contador de Programa com os conteúdos (endereço do programa de inicialização - PARES) das locações de memória FFFE e FFFF (endereços do vetor de atendimento da MPU ao sinal  $\overline{RESET}$ ) e passa então a executar o programa de inicialização do sistema, que é o PARES.

Observe-se que o "reset" das ACIAs é executado em "software" porque, ao contrário da PIAs, as ACIAs não possuem entrada para o "reset" por "hardware".

Ao terminar a inicialização do sistema, o PARES entrega o comando do sistema ao Programa de Controle.

### 3.3 - PROGRAMA DE CONTROLE - PC

O programa de controle é quem disciplina e coordena a execução das diferentes tarefas do sistema pelos outros programas existentes. O PC assume o comando do sistema nas seguintes condições:

- a) após a inicialização do sistema quando da sua energização inicial ou após a sua energização depois de uma falta de energia,
- b) após uma reinicialização do sistema que pode acontecer a qualquer instante sempre que o operador desejar e,
- c) após o término da execução de qualquer das tarefas do sistema, se o mesmo encontra-se no modo de funcionamento CPO (controlado pelo operador).

Existem dois modos de funcionamento deste sistema (veja linhas tracejadas das figuras 3.1.1 e 3.1.2):

- a) modo de funcionamento controlado pelo operador (CPO) e
- b) modo de funcionamento automático (AUTO).

No modo de funcionamento CPO, o sistema só executa uma tarefa qualquer quando a mesma é requisitada pelo operador ao programa de controle, que volta a assumir o comando logo após o término da execução da tarefa requisitada.

No modo de funcionamento AUTO, alguns programas podem chamar outro programa. Estes programas são o GAD, que ao terminar a

aquisição de dados chama o GPI, que ao terminar o processamento da informação chama o PIR, que ao terminar a impressão do relatório chama o PAK-7, que ao terminar o armazenamento dos dados brutos em fita cassete devolve o comando do sistema ao PC.

O Programa de Controle (veja fig. 3.3) começa por transmitir uma mensagem de identificação, através da SRT/OUTTAB (veja item 3.4), que indica ao operador que está pronto para atender a alguma requisição de serviço para o qual o sistema foi projetado. Enquanto nesta situação, o PC simplesmente realiza o "polling" do teclado através da SRT/INICT (veja item 3.5). Ao identificar uma requisição de serviço, o PC chama o programa responsável pela execução desta tarefa.

Segue-se a descrição de três subrotinas que são responsáveis pela transmissão/recepção de caracteres, ou seja, são responsáveis pela comunicação entre o sistema de aquisição de dados e o operador através de terminal de vídeo.

#### 3.4 - SUBROTINA PARA TRANSMISSÃO DE TABELAS - SRT/OUTTAB

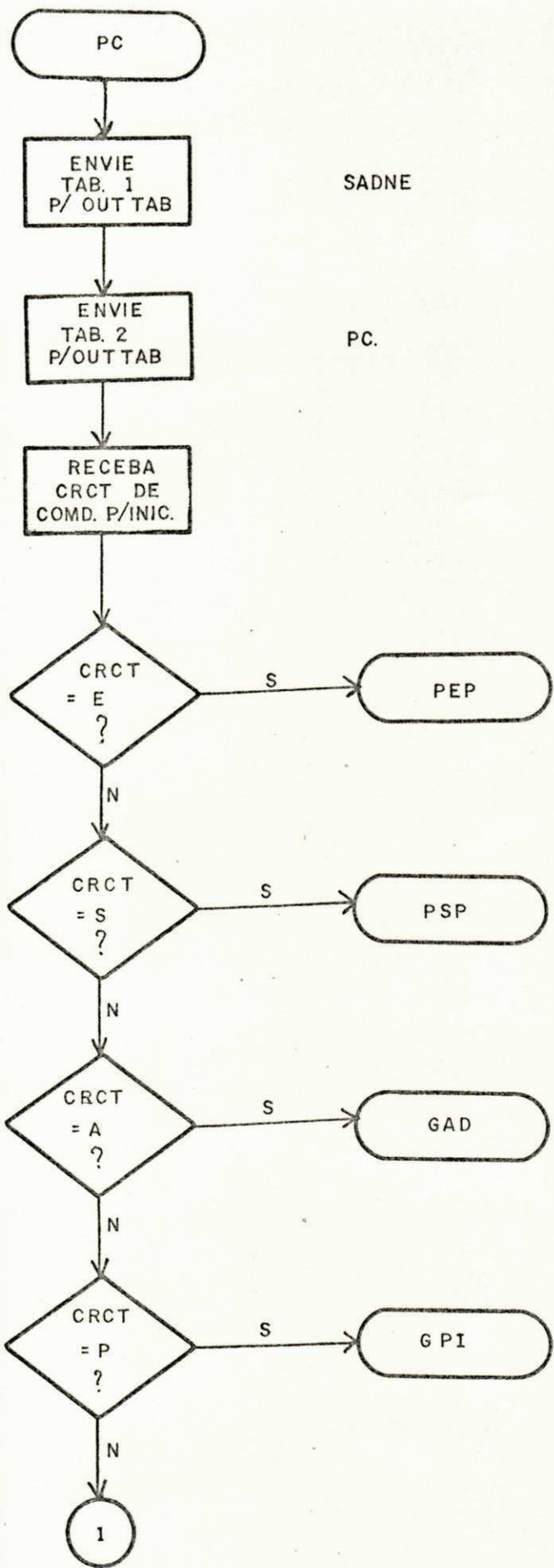
Esta subrotina é responsável pela transmissão de tabelas que contêm:

- a) mensagens de identificação dos programas PC, PEP, PSP e PIR,
- b) mensagens de erro,
- c) perguntas que são feitas ao operador pelo PEP.

Os caracteres que compõe as tabelas são armazenados em código ASCII ("American Standard Code for Information Interchange", 1968), código normalmente utilizado para teclados, e o último caractere de cada tabela é um "flag" que indica à SRT/OUTTAB o fim da tabela.

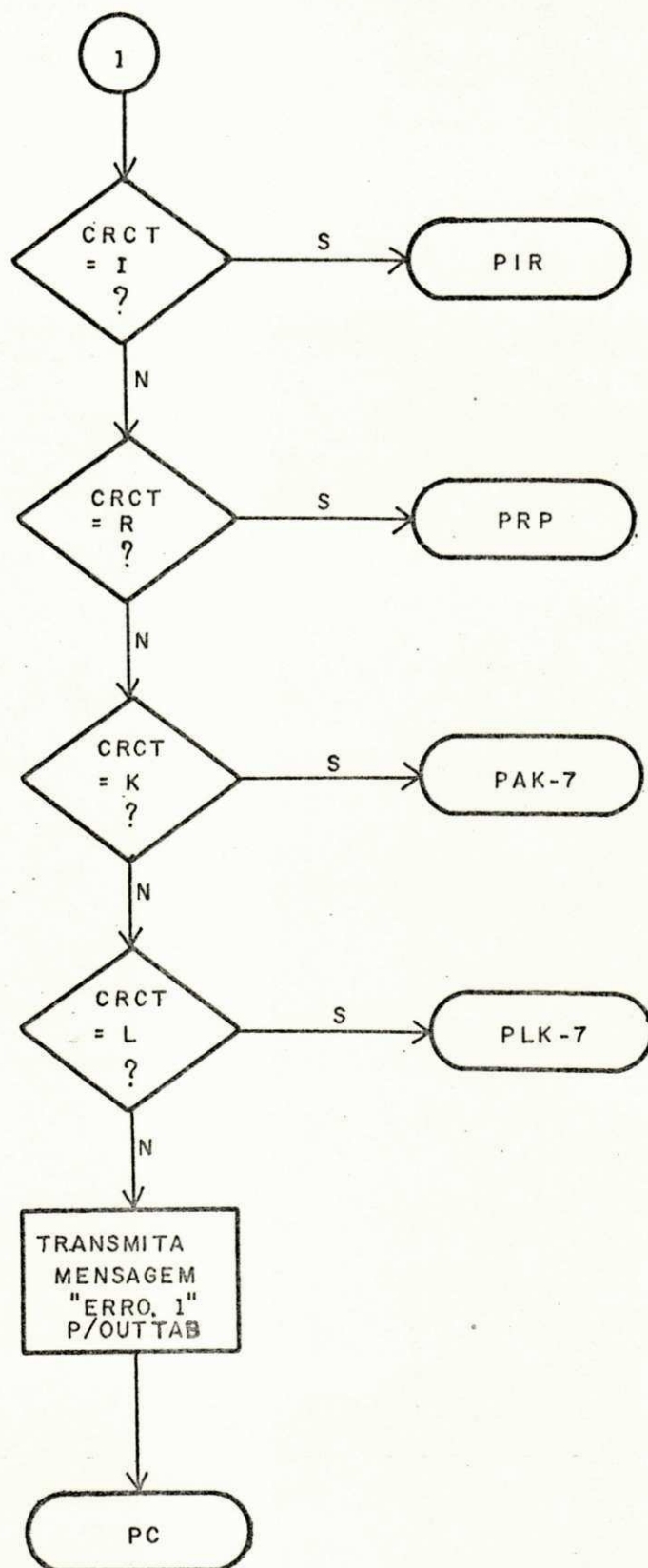
A SRT/OUTTAB utiliza indexação para retirar cada caractere da tabela e armazená-lo no acumulador A para então transmiti-lo (veja fig. 3.4).





SADNE

PC.





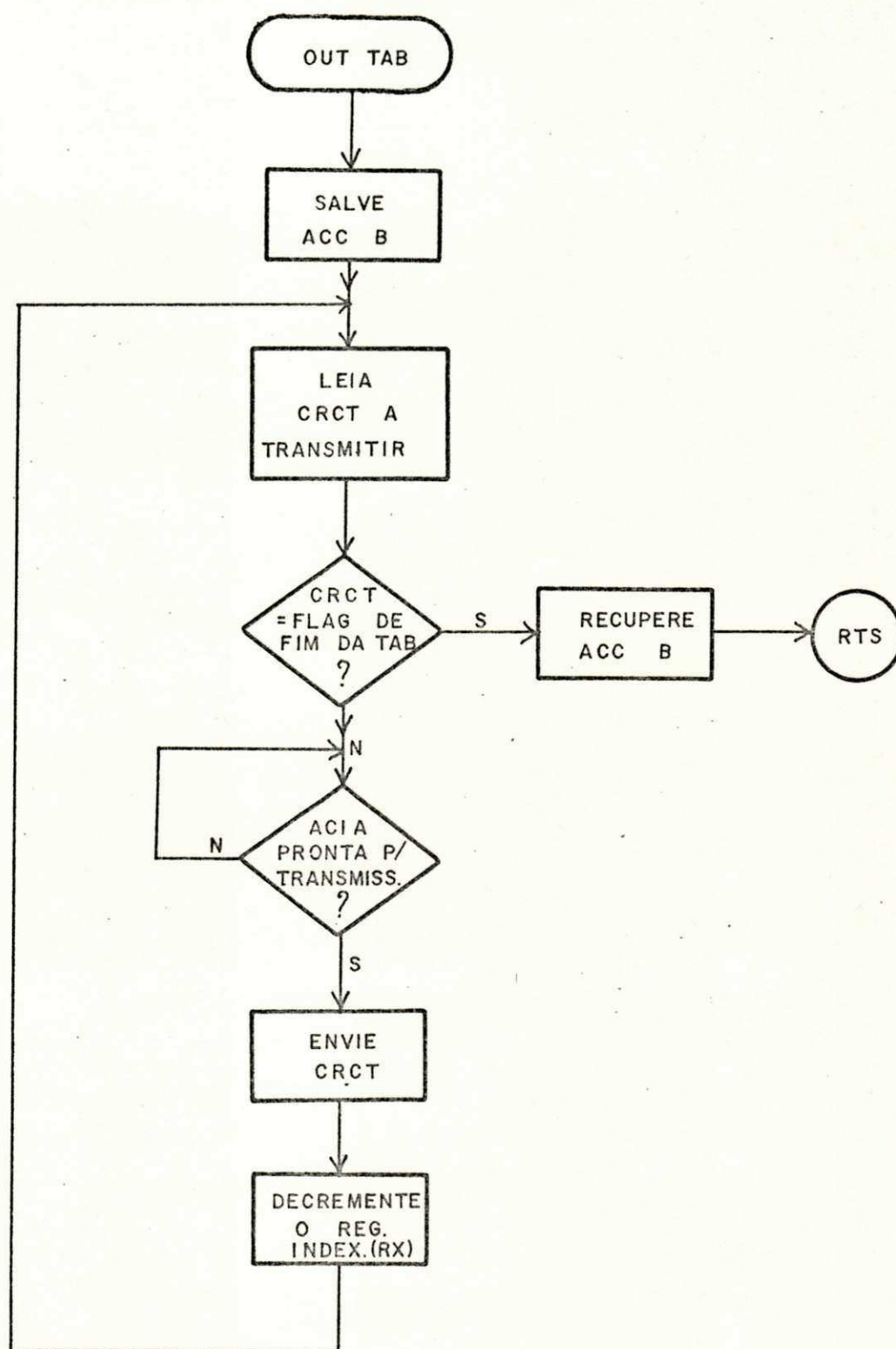


FIG. 3.4 - Fluxograma da Subrotina Para Transmissão de um Tab

### 3.5 - SUBROTINA PARA RECEPÇÃO DE CARACTER - SRT/INICT

Esta subrotina é responsável pela recepção de caracteres enviados pelo operador através de terminal de vídeo.

A SRT/INICT realiza o "polling" da ACIA e com a chegada de um caracter (em código ASCII) armazena este no acumulador A. Os próximos passos são limpar o bit de paridade do caracter recebido e transmiti-lo de volta (transmissão de eco) pela SRT/OUTICT (veja fig. 3.5).

### 3.6 - SUBROTINA PARA TRANSMISSÃO DE UM CARACTER - SRT/OUTICT

Esta subrotina é responsável pela transmissão de um caracter armazenado em código ASCII no acumulador A.

A SRT/OUTICT (veja fig. 3.6) é utilizada para transmissão do eco do caracter para que o mesmo possa aparecer na tela do terminar de vídeo e também quando se faz necessário a transmissão de apenas um caracter em alguns programas do sistema, sem que para isso seja preciso utilizar a SRT/OUTTAB.

### 3.7 - PROGRAMA DE ATENDIMENTO À INTERRUPTÃO $\overline{\text{NMI}}$ - PANMI

O programa de atendimento à Interrupção  $\overline{\text{NMI}}$  é responsável pela atualização do RTC, implementado em "software", no sentido de que está encarregado de chamar o Programa do RTC - PRTC, a cada segundo, para que o PRTC execute a atualização do horário e calendário de tempo real. Quando do período de aquisição de dados, o PANMI também é responsável pela contagem do tempo de aquisição de dados.

A MPU ao detetar um nível lógico zero na linha  $\overline{\text{NMI}}$ , antes de executar o PANMI, executa a sequência vista na figura 3.7.1. Neste processo de interrupção, a MPU primeiro conclui a execução da instrução em que trabalhava antes de ser interrompida e "salva" na pilha seu estado atual (a MPU armazena na pilha conteúdos do Registrador do Código de Condições, dos acumuladores A e B, do Registrador de Indexação e do Contador de Programa). A MPU não procede desta maneira no processo de "reset" por ser este um processo de inicialização ou de reinicialização onde não interessa à MPU "salvar" seu estado atual.



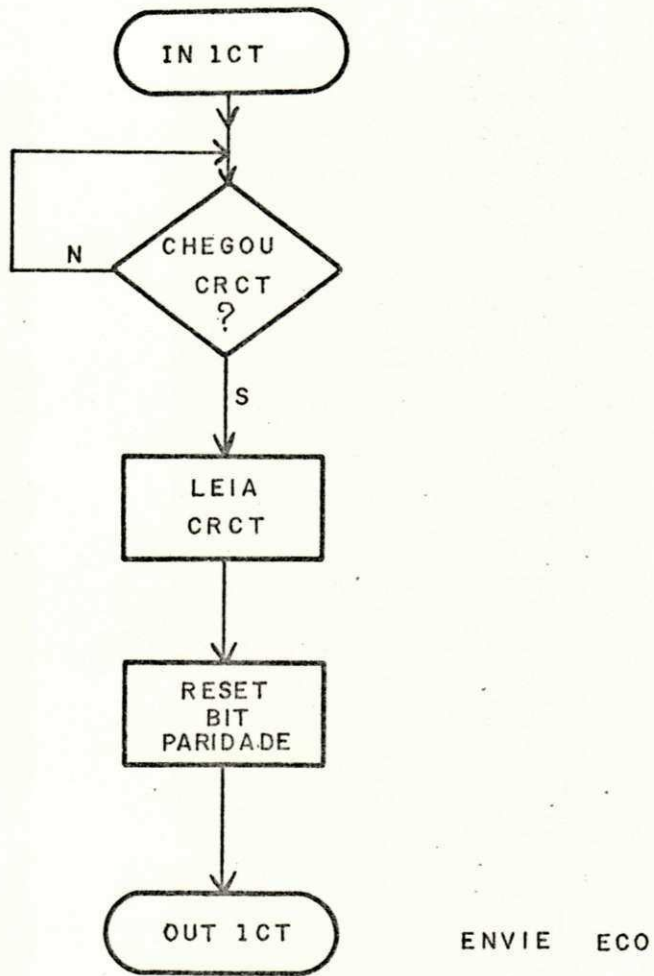


FIG. 3.5 - Fluxograma da Subrotina Para Recepção de Caracter

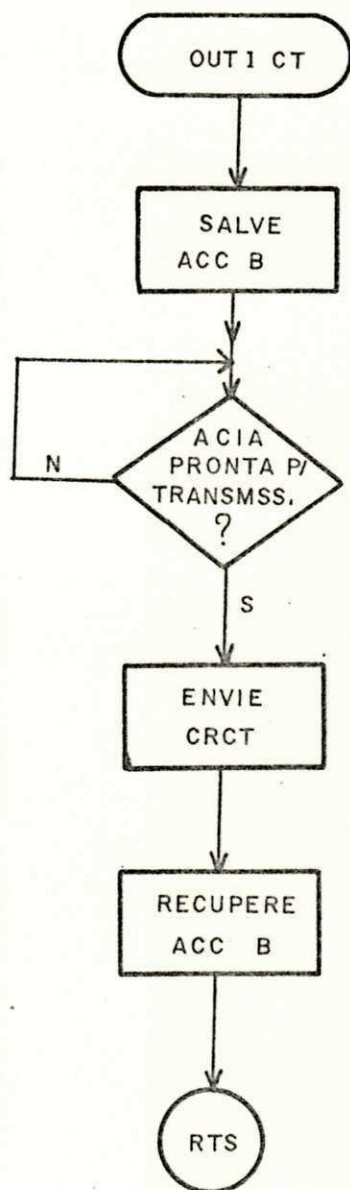
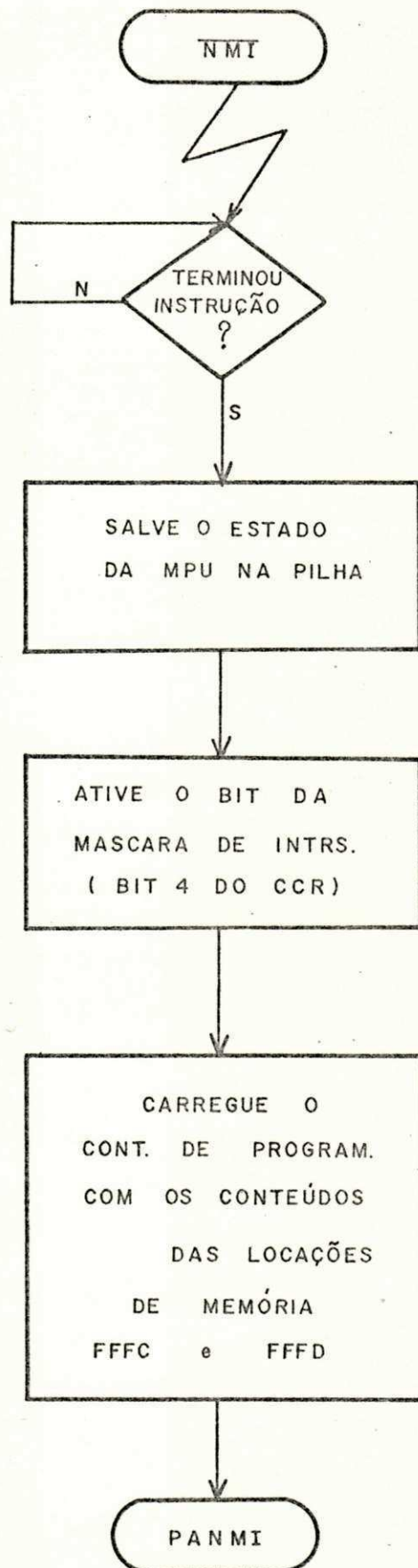


FIG. 3.6 - Fluxograma da Subrotina Para Transmissão de Um Caracter





Em seguida, como no processo de interrupção  $\overline{\text{RESET}}$ , a MPU ativa o bit da máscara de interrupção e carrega o Contador de Programa com os conteúdos (endereço do PANMI) das locações de memória FFFC e FFFD (endereço do vetor de atendimento da MPU à interrupção  $\overline{\text{NMI}}$ ) e passa então a executar o Programa de Atendimento à Interrupção  $\overline{\text{NMI}}$  - PANMI (fig. 3.7.2).

### 3.8 - IMPLEMENTAÇÃO DO RTC EM SOFTWARE

Resolveu-se implementar o RTC ("Real Time Clock") em "software" por ser uma tarefa facilmente executada pelo microprocessador e também, por ser este um sistema dedicado, tal tarefa não influi na sua capacidade computacional (veja fig. 3.8.1).

O RTC tem precisão de segundos, calendário válido até o ano de 1999 e fornece ao operador, através de "displays" localizados no armário que contém o SADNE, hora e minuto. Se necessário o operador pode pedir ao sistema através do PSP (item 3.11), hora (com precisão de segundos) e data.

Quando da energização do sistema, na sua inicialização ou depois de uma falta de energia o operador deve "acertar" o RTC através do PEP (item 3.9). O RTC passa então a ser atualizado a cada segundo por meio de uma interrupção  $\overline{\text{NMI}}$  gerada em "hardware". A cada minuto, o RTC atualiza o horário dos "displays" através do Programa DISPLAY (fig. 3.8.2) e retorna ao PANMI.

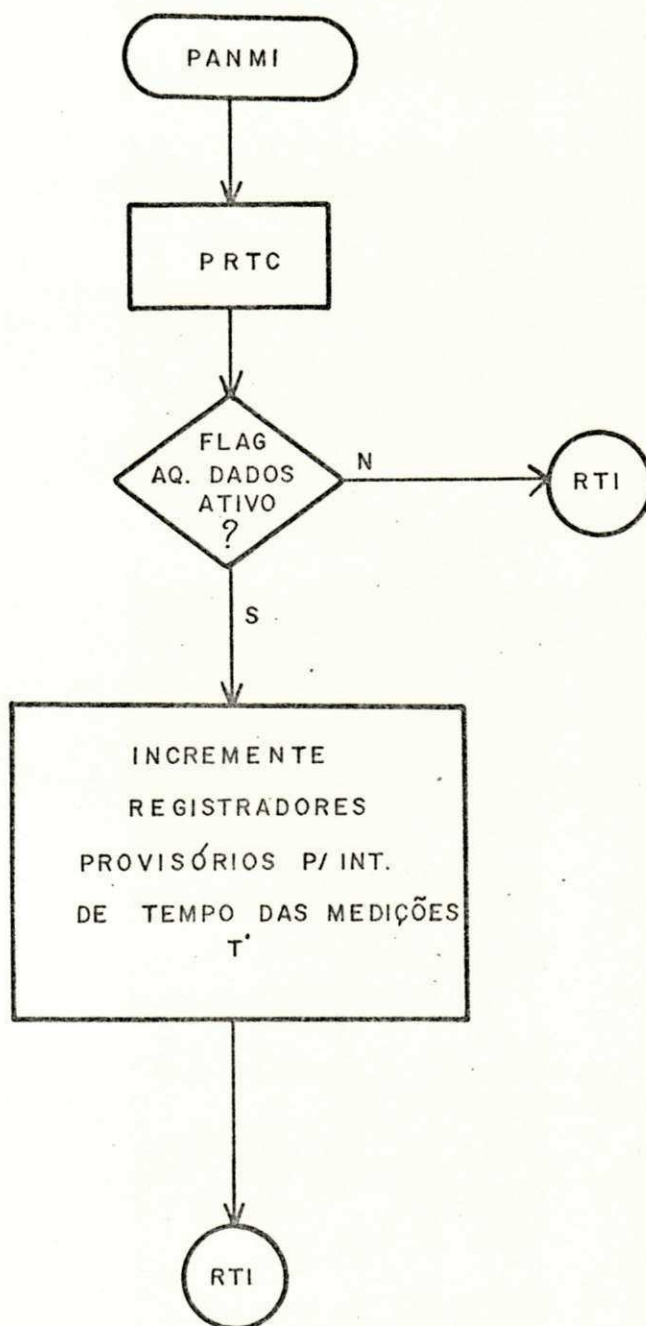
O fluxograma da implementação do RTC - PRTC (fig. 3.8.1) é bastante claro e dispensa maiores comentários. A implementação do teste para verificar se o ano é bissexto é mostrada no fluxograma da figura 3.8.3.

### 3.9 - PROGRAMA DE ENTRADA DE PARÂMETROS - PEP

O Programa de Entrada de Parâmetros executa a função de interagir ou conversar com o operador requisitando informações necesárias ao bom funcionamento do sistema. O PEP está dividido em duas etapas:

- a) a primeira na qual faz uma série de perguntas ao operador que deve respondê-las todas quando da condição de inicialização ou energização do sistema, ou a apenas al





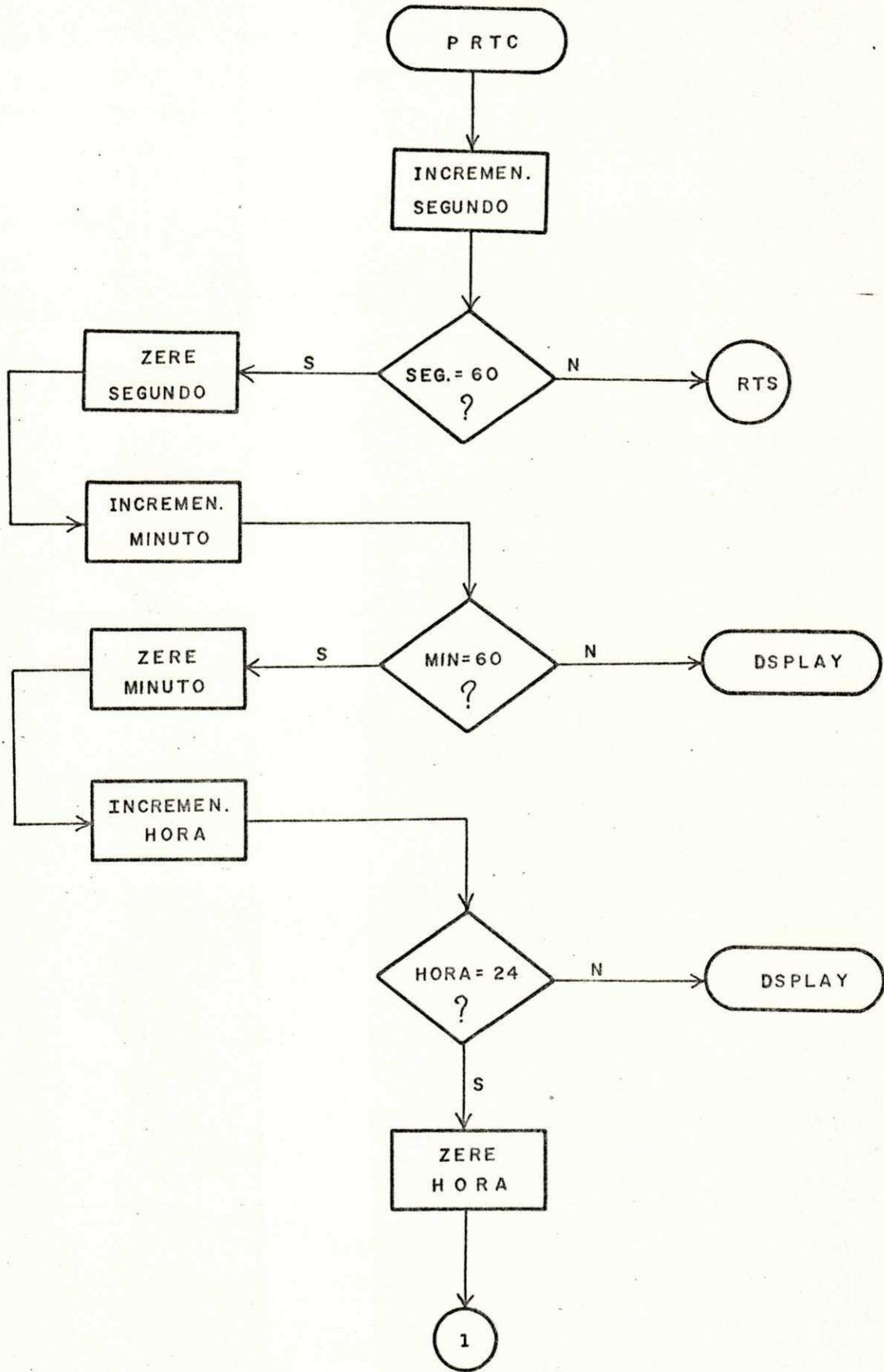
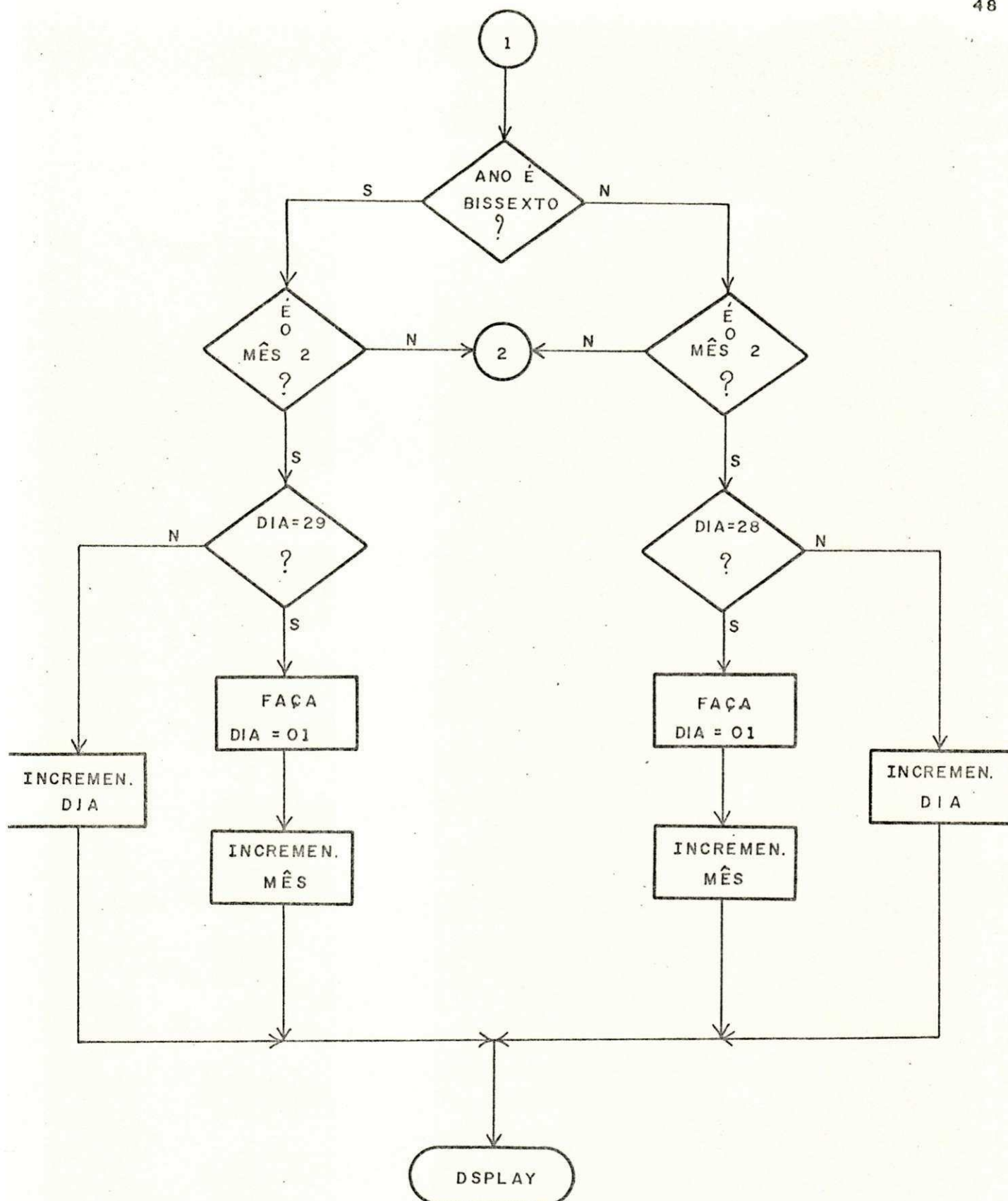
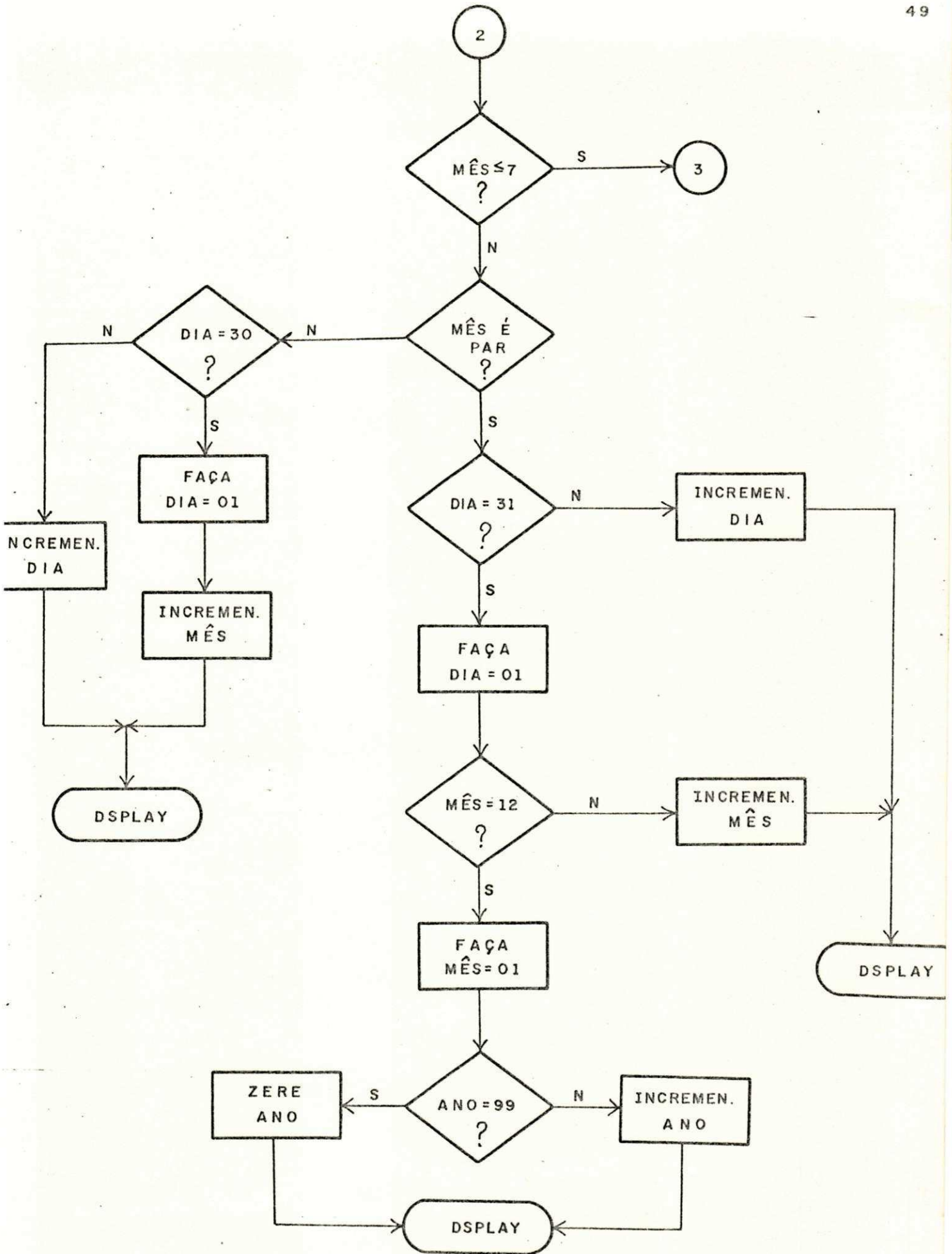
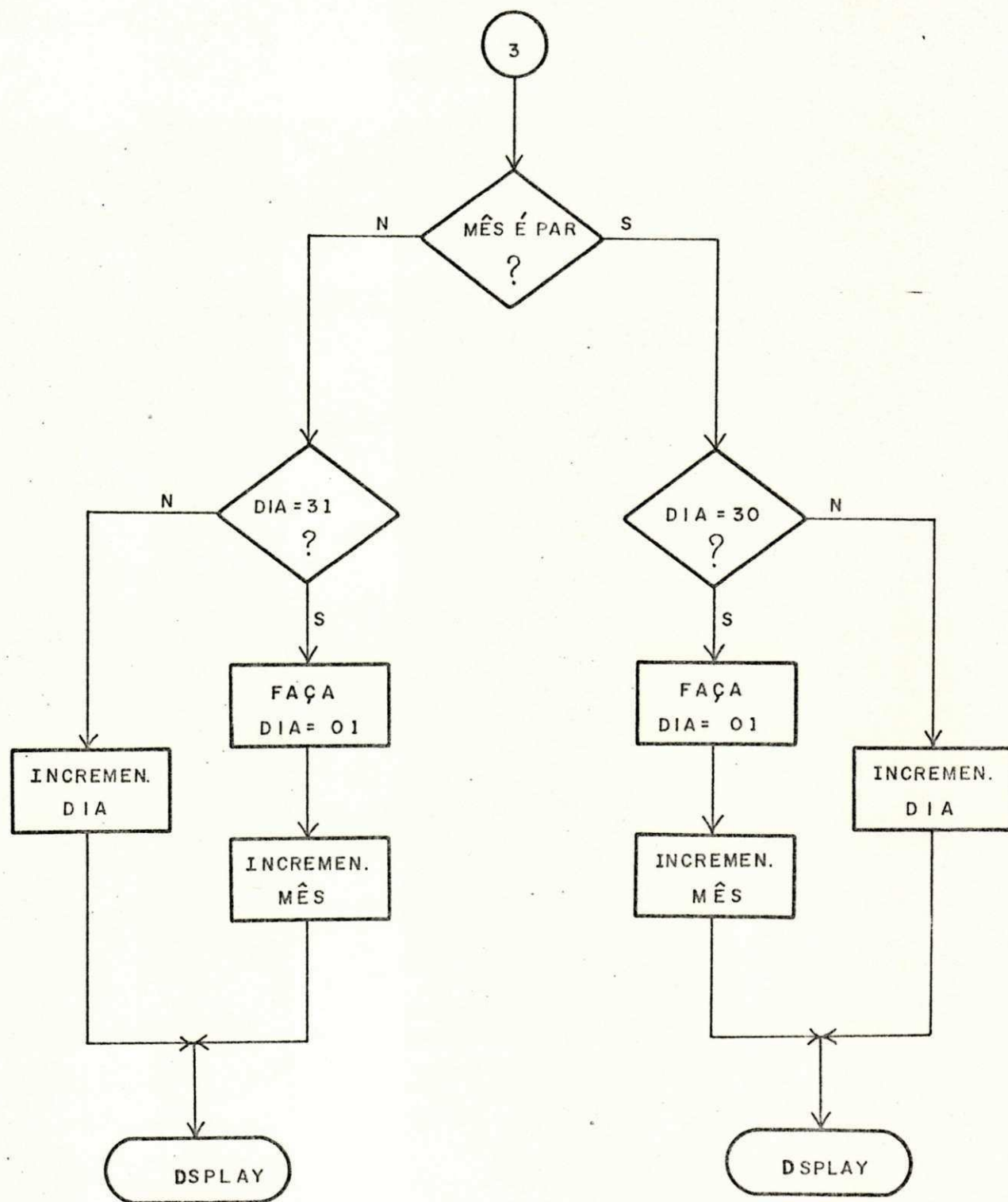


FIG. 2.0.1. 57



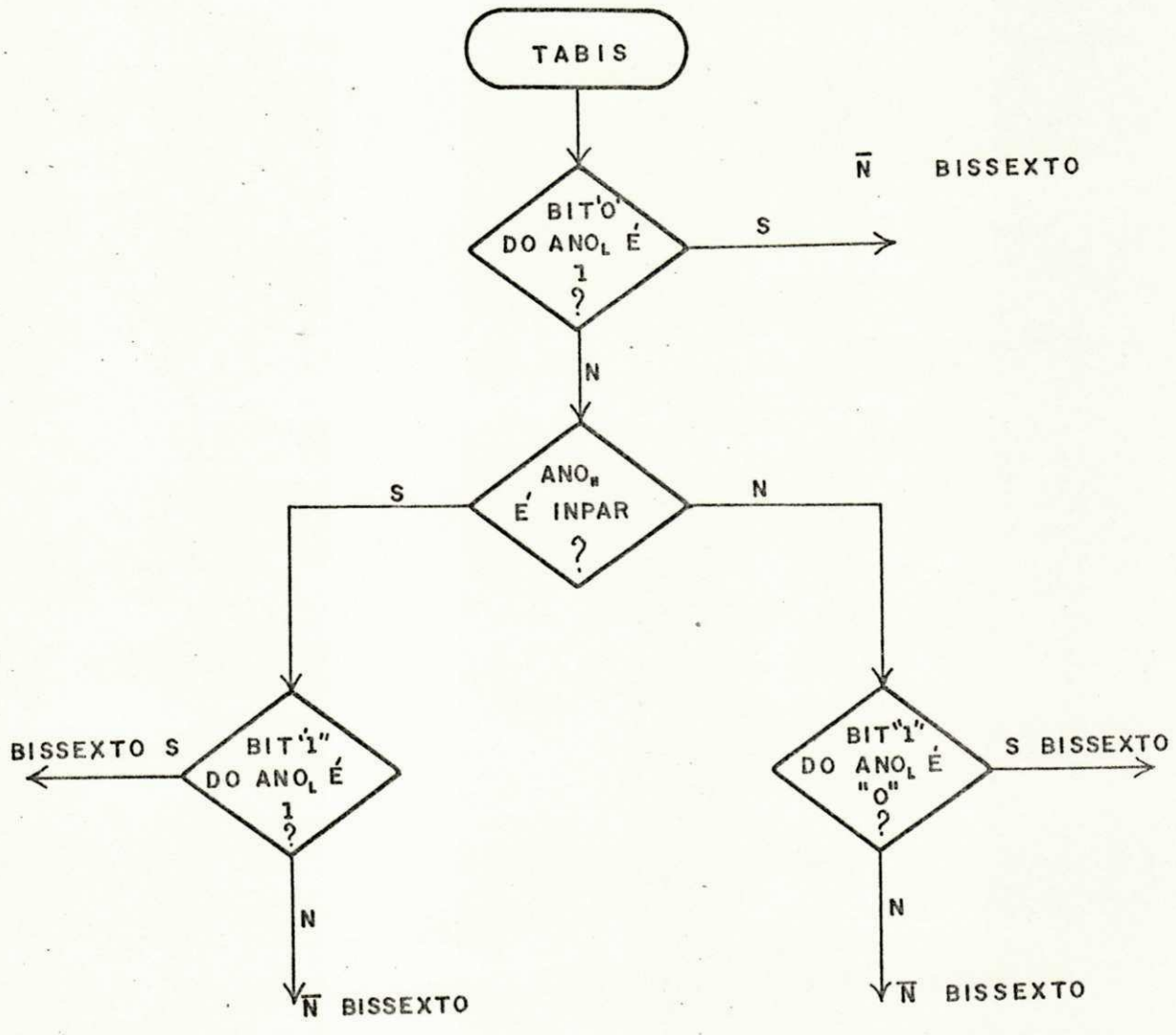






UFPP/BIBLIOTECA/PRAI

FIG. 3.8.1 - Fluxograma do PRTC





gumas, se desejar, antes de uma nova série de aquisições de dados e

- b) a segunda, de definição da atividade das torres e bases de turbinas, na qual é o próprio operador quem define que torres ou bases quer selecionar para a aquisição de dados e que instrumentos destas ou daquelas sofrerão a aquisição de dados.

A série de perguntas, feita pelo PEP ao operador na primeira parte do programa, refere-se :

- a) ao intervalo de tempo de aquisição de dados,  
 b) ao número de aquisições de dados,  
 c) ao acerto do horário e calendário do RTC,  
 d) ao modo de funcionamento do sistema e  
 e) à densidade do ar, parâmetro esse utilizado no cálculo da potência do vento.

Veja abaixo uma listagem das perguntas executadas na primeira etapa do PEP:

SADNE-PEP

INT MEDIÇÕES T = XX:XX(CR)

No MEDIÇÕES N = XXXX(CR)

DENSIDADE AR = X,XX(CR)

HORA: XX:XX:XX(CR)

DATA: XX/XX/XX(CR)

MODO AUTOMÁTICO? X

Nesta listagem, observe-se que:

- a) o grifo indica a informação "teclada" pelo operador,  
 b) os "Xs" representam o número de dígitos decimais com os quais o operador pode responder às perguntas feitas e  
 c) o "CR", embora "teclado" pelo operador, aparece entre parêntesis por ser uma tecla de controle da máquina.

Para o intervalo de tempo T de medições ou aquisição de dados foram reservados quatro dígitos decimais, dois para minutos e

dois para segundos, fixando assim, o intervalo de tempo máximo de medições em 99 minutos e 60 segundos, ou seja, 100 minutos.

Para o número de medições N também foram reservados quatro dígitos, sendo portanto, o número máximo de medições igual a 9999 medições.

Para a densidade do ar foram reservados três dígitos decimais, sendo um inteiro e duas casas decimais. As perguntas HORA e DATA são destinadas para correção ou acerto do horário e calendário do RTC. Para cada uma destas perguntas foram reservados seis dígitos decimais.

A pergunta sobre o modo de funcionamento do sistema deve ser respondida com um S (sim) ou um N (não). A tecla S corresponde ao modo automático de funcionamento (modo AUTO) e a tecla N corresponde ao modo de funcionamento controlado pelo operador (modo CPO).

Na segunda etapa do PEP, o operador define para o sistema a atividade das torres e bases. Entenda-se por atividade a indicação de quais torres ou bases foram selecionadas para aquisição de dados e que instrumentos destas torres ou bases sofrerão aquisição de dados.

Para armazenamento desta informação foram reservadas nove locações de memórias, as quais foram denominadas de vetores de atividade das cinco torres e das três bases de turbinas do sistema de aquisição de dados.

A primeira destas nove locações de memória foi denominada de Vetor de Atividade das Torres e Bases (VATTB), onde os cinco primeiros bits desta locação de memória representam a atividade das cinco torres e os outros 3 bits representam a atividade das 3 bases de turbinas. A informação é armazenada em binário onde "1" representa torre ou base selecionada e "0" representa a não seleção da torre ou base.

As próximas cinco locações de memória foram escolhidas para armazenar a atividade dos instrumentos das cinco torres e foram denominadas de Vetores de Atividade dos Instrumentos das Torres (VATT1 a VATT5). Como em cada torre podem ser conectados até 8 instrumentos, cada bit destas locações de memória representa a atividade de um instrumento. A informação é armazenada em binário onde "1" representa instrumento ativo ou instrumento que sofrerá medições e "0"



representa instrumento desativado.

As três últimas locações de memória foram escolhidas para armazenar a atividade dos instrumentos das três bases de turbinas e foram denominadas de Vetores de Atividade dos Instrumentos das Bases (VATB1 a VATB3). Como acontece com as torres, em cada base podem ser conectados até 8 instrumentos e cada bit destas locações de memória representa a atividade de um instrumento. A informação também é armazenada em binário e "1" representa instrumento ativo e "0" instrumento desativado.

Na sua primeira etapa (veja fig. 3.9.1), o PEP começa por transmitir uma mensagem de identificação e logo em seguida faz a primeira pergunta ao operador. Feita a pergunta, o PEP executa o "polling" do teclado através da SRT/INPAR (veja item 3.9.1) que armazena a informação "teclada" pelo operador no "buffer" temporário para parâmetros. Em seguida, o PEP processa a informação recebida (de ASCII para BCD) e armazena-a em local apropriado no "buffer" permanente para parâmetros. Estes processamento e armazenamento são executados pela SRT/STOC (veja item 3.9.2). O PEP faz então nova pergunta e este procedimento repete-se até que todas as perguntas sejam feitas.

Terminada esta primeira etapa, o PEP inicia a segunda etapa transmitindo uma mensagem de identificação para logo em seguida transmitir um caracter que indica ao operador que o mesmo pode começar a definir a atividade das torres e bases. O PEP executa então o "polling" do teclado e armazena a informação "teclada" pelo operador nos vetores de atividade apropriados.

O PEP "entende" alguns comandos que desempenham funções de terminadas neste programa. Veja descrição destes comandos abaixo:

- a) comando "C" - este comando é utilizado para cancelar-se uma resposta já existente ou para cancelar-se uma resposta na qual o operador observa que está cometendo erro.
- b) comando "CR" - este comando indica fim da informação ou fim da resposta e é utilizado também para indicar ao PEP que não se deseja responder a uma pergunta feita.
- c) comando "X" - com este comando o PEP devolve o comando do sistema ao PC.



Estes três comandos são utilizados nas duas etapas do PEP. Os próximos comandos são utilizados na segunda etapa apenas.

- d) comando "Tn" - este comando indica ao PEP que a torre "n" ("n" varia de 1 a 5) foi selecionada para a aquisição de dados.
- e) comando "Bm" - este comando indica do PEP que a base "m" ("m" varia de 1 a 3) foi selecionada para aquisição de dados.
- f) comando "D" - este comando indica ao PEP que todos os instrumentos de uma determinada torre ou base foram selecionados para sofrer aquisição de dados. Escolheu-se o modo "default" para a atividade dos instrumentos como sendo todos ativos. Desta maneira, o operador deve comunicar ao PEP se algum instrumento de uma determinada torre ou base não vai sofrer aquisição de dados.
- g) comando "/" - este comando indica ao PEP que o operador deseja desativar algum instrumento de uma determinada torre ou base.

Para uma melhor compreensão da utilização destes comandos veja a seguir uma listagem de um exemplo da definição de parâmetros e atividade do sistema, quando da condição de inicialização do sistema.

SADNE-PEP

INT MEDIÇÕES T = 10:15(CR)

No MEDIÇÕES N = 100(CR)

DENSIDADE AR = 1,12(CR)

HORA: 10:25:00C

HORA: 10:35:00(CR)

MODO AUTOMÁTICO? S

DEFINA TORRES E BASES

\* TID

\* T2/3,4,5 C

\* T2/3,4,6 (CR)

\* B1D

\* X

SADNE-PC.

Neste exemplo, o sistema executará uma série de 100 aquisições de dados em intervalos de 10 minutos e 15 segundos. Ao acertar o horário do RTC, o operador notou que cometeu um erro e utilizou o comando "C" para cancelar sua resposta. O modo de funcionamento escolhido para o sistema foi o modo automático (AUTO). Na segunda etapa, o operador selecionou as torres 1 e 2 e a base 1 para sofrer aquisição de dados, sendo que a torre 1 e a base 1 estão c/ todos seus instrumentos ativos e a torre 2 tem os instrumentos 3, 4 e 6 desativados. Com o comando "X", o operador chamou o Programa de Controle.

### 3.9.1 - SUBROTINA QUE RECEBE PARÂMETROS - SRT/INPAR

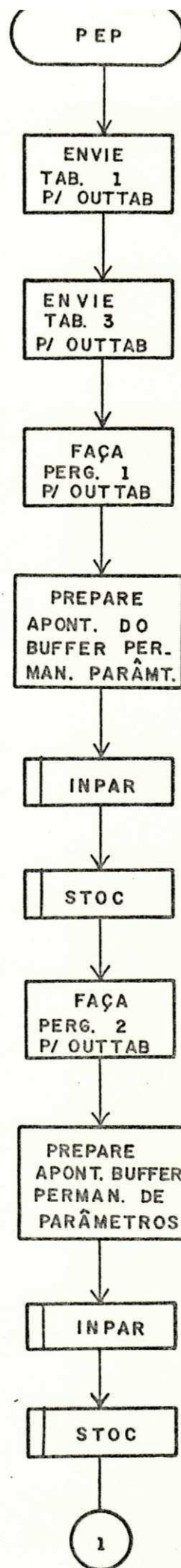
Esta subrotina está encarregada de receber a informação "teclada" pelo operador na primeira etapa do PEP e armazená-la no "buffer" temporário para parâmetros, o qual tem capacidade máxima de seis locações de memória (veja fig. 3.9.2).

A SRT/INPAR executa o "polling" do teclado e testa a validade do caracter recebido, que pode ser alfa-numérico. Esta subrotina transmite mensagens de erros quando detecta erro na formatação das respostas ou quando detecta a utilização de comandos em local não apropriado.

Os caracteres utilizados para conexão nas respostas (":", "/", e ",") são simplesmente ignorados pela SRT/INPAR.

### 3.9.2 - SUBROTINA QUE ARMAZENA E PROCESSA PARÂMETROS RECEBIDOS - SRT/STOC.

Esta subrotina está encarregada de processar a informação, "teclada" pelo operador e armazenada no buffer temporário para parâmetros, de ASCII para BCD, para em seguida armazená-la em local apropriado no "buffer" permanente para parâmetros (veja fig.(3.9.3).



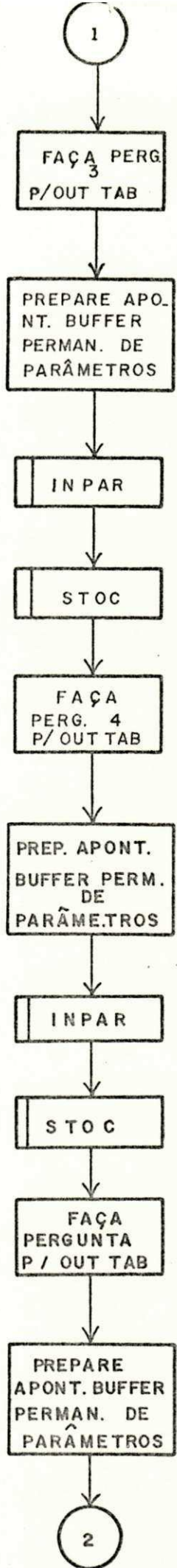
SADNE

PEP

INT MEDIÇÕES T = ?

Nº MEDIÇÕES N = ?

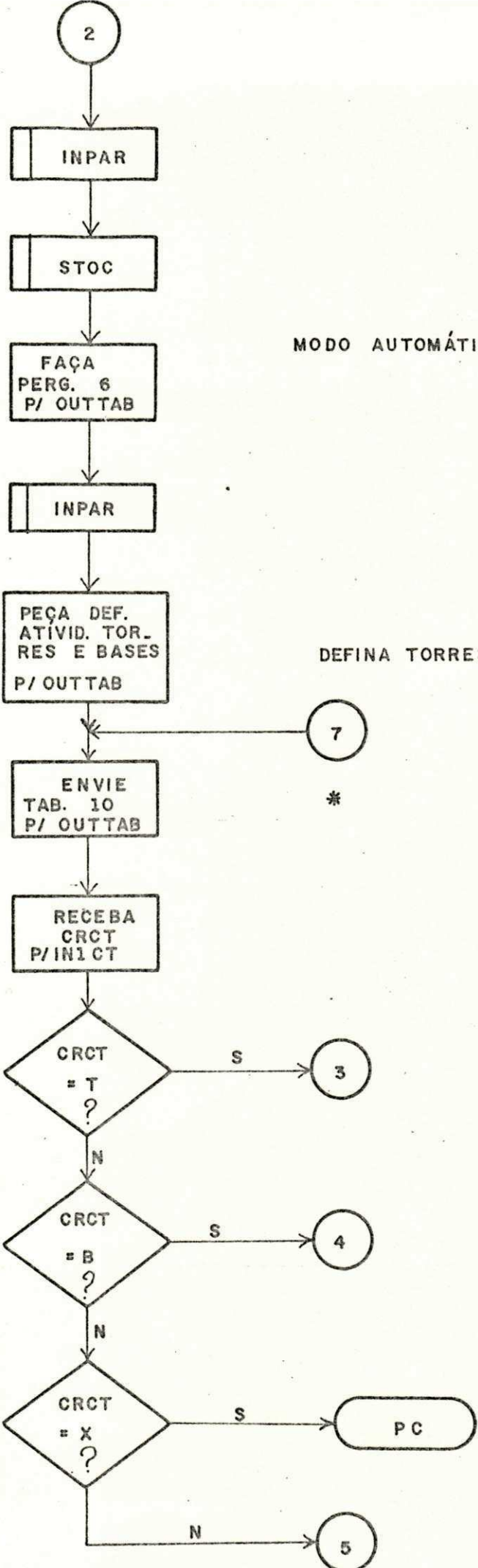




DENSIDADE AR=?

HORA: ?

DATA: ?



MODO AUTOMÁTICO ?

DEFINA TORRES E BASES

\*

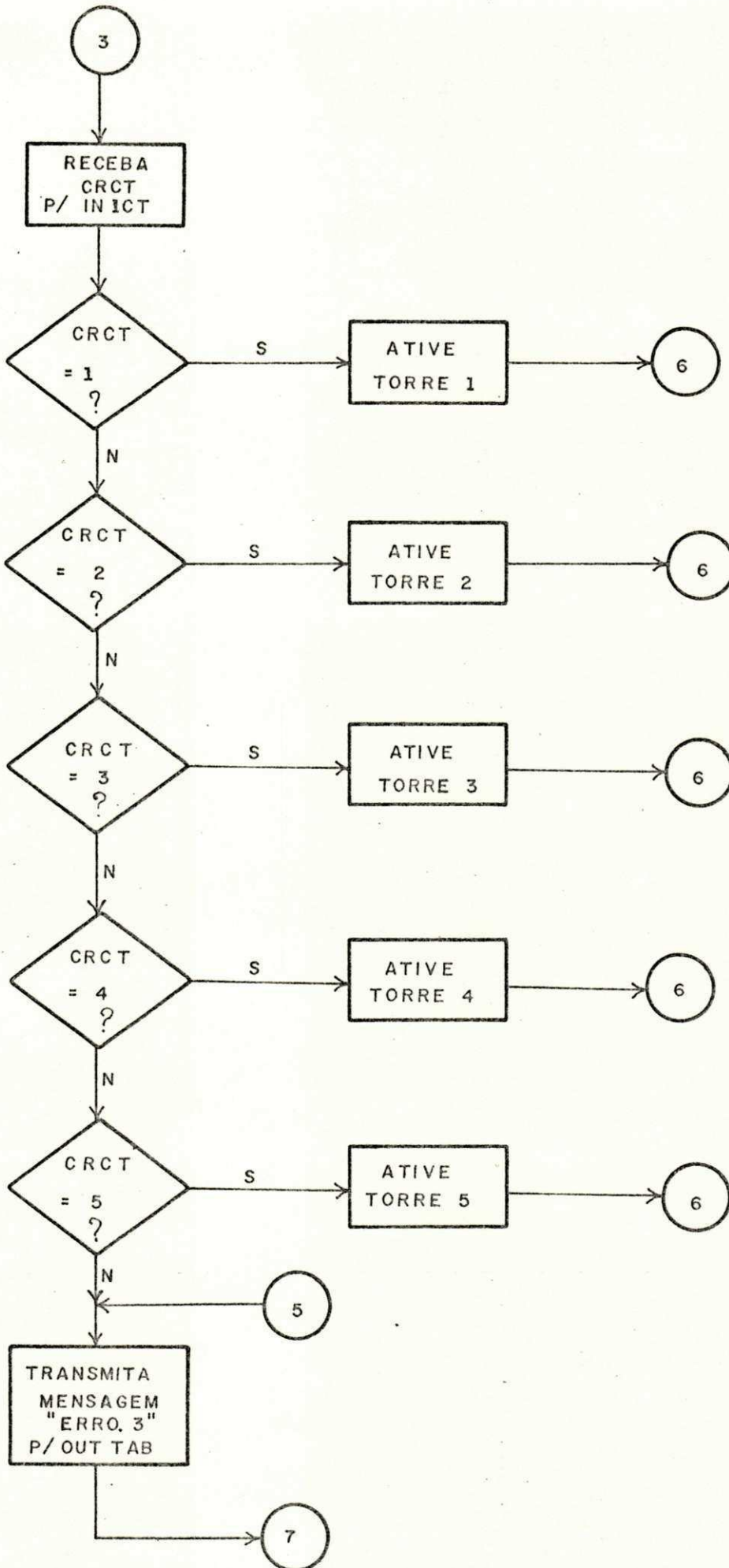
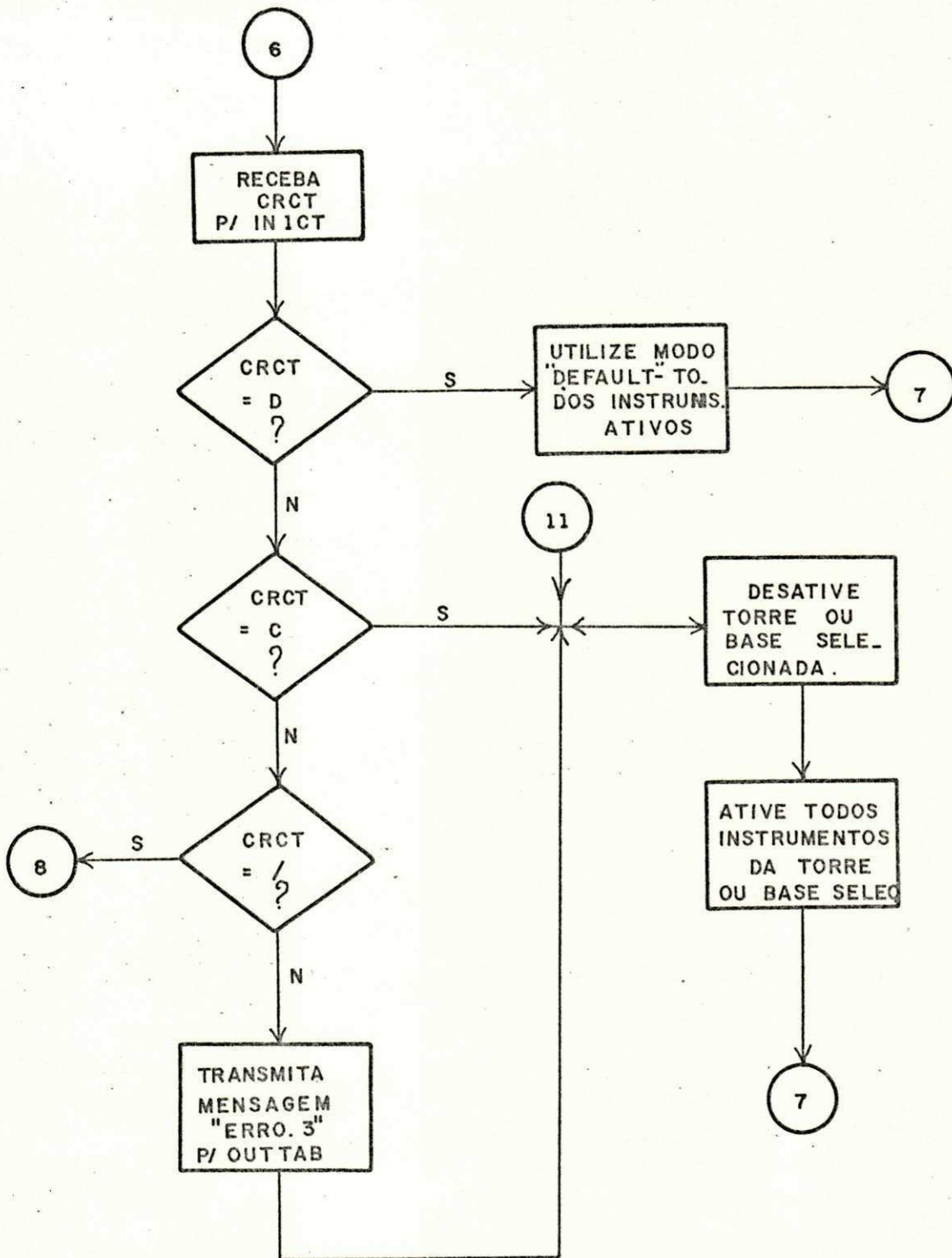
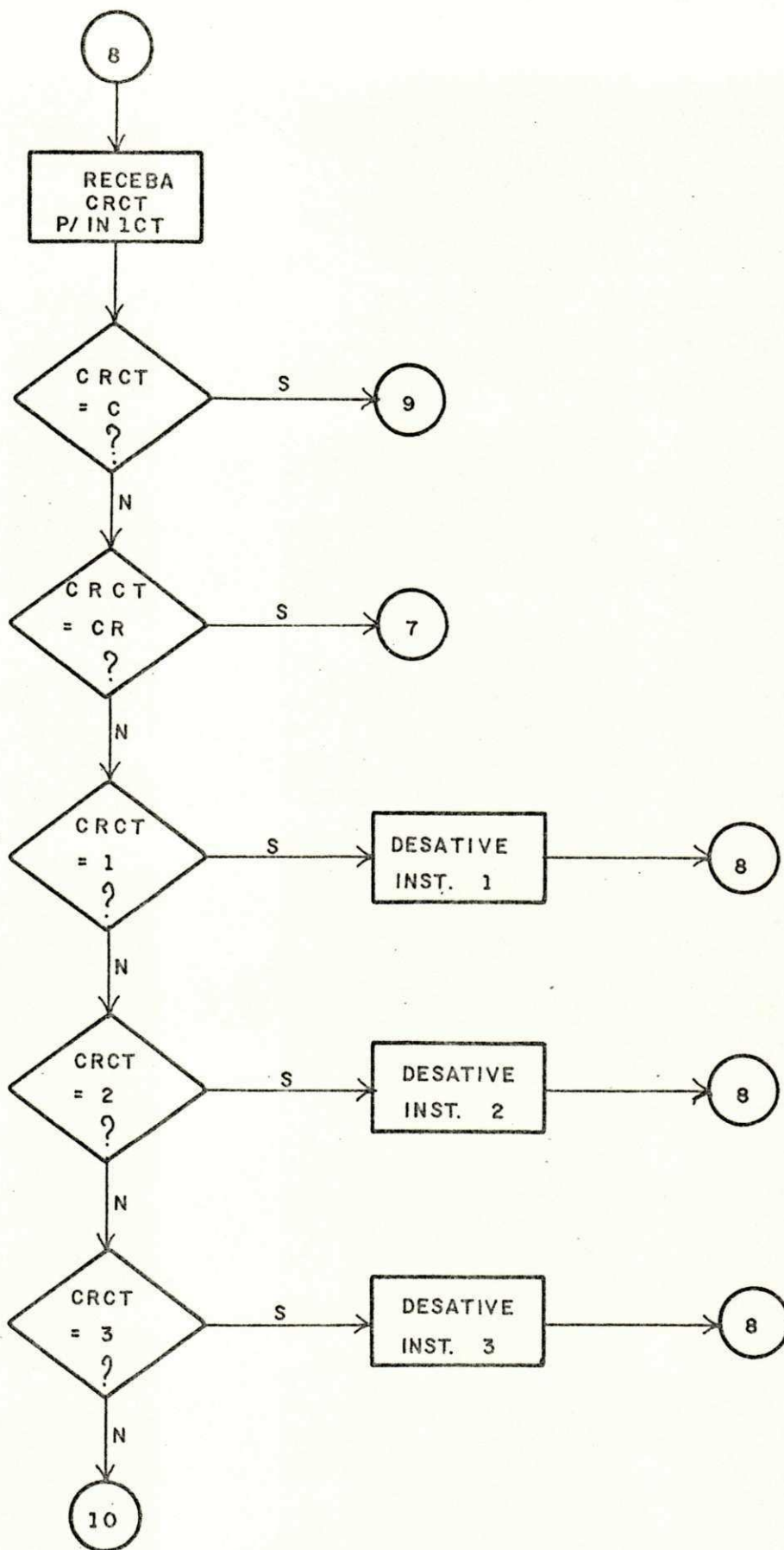
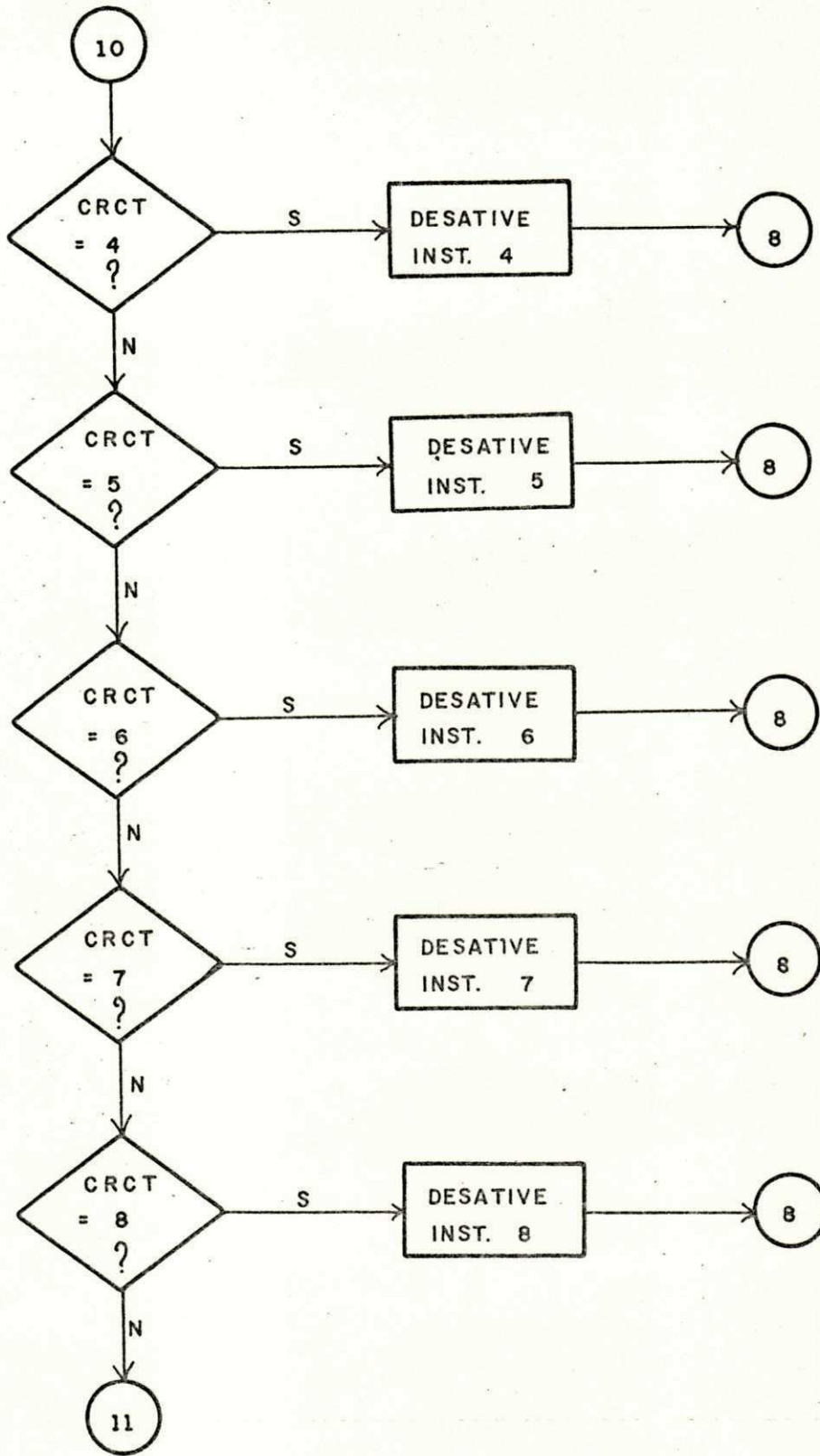


FIG. 3.9.1 - Fluxograma do DFD

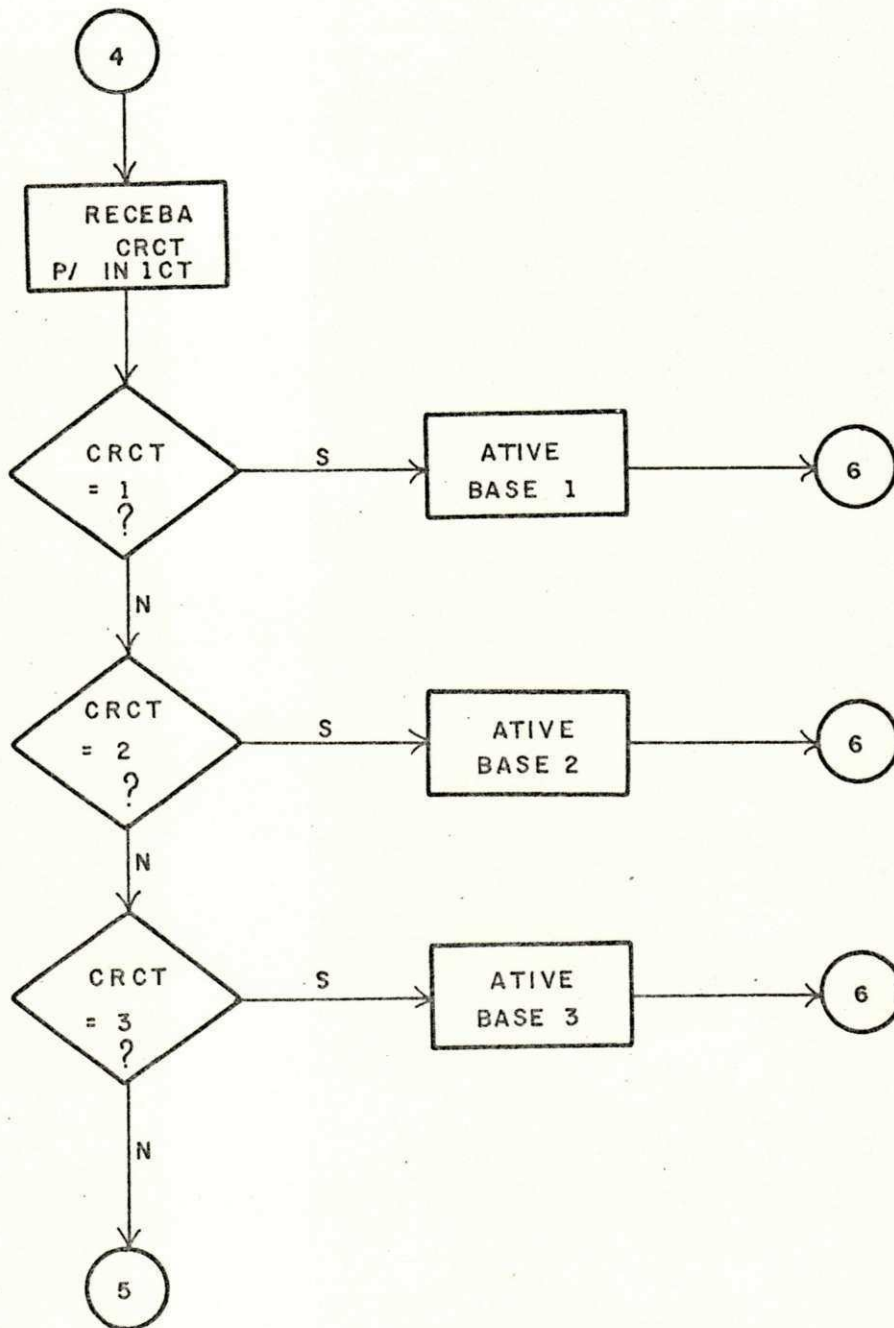


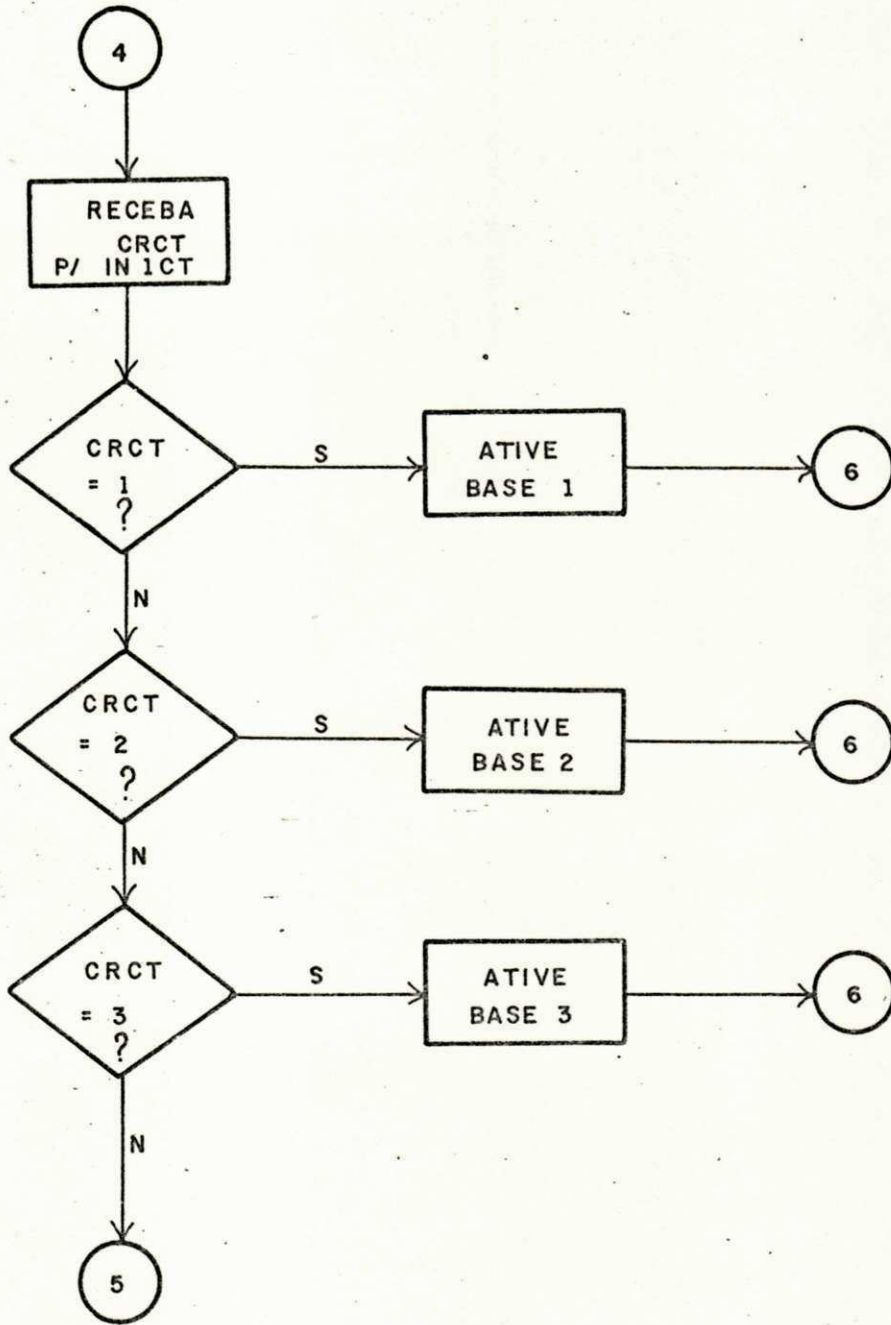


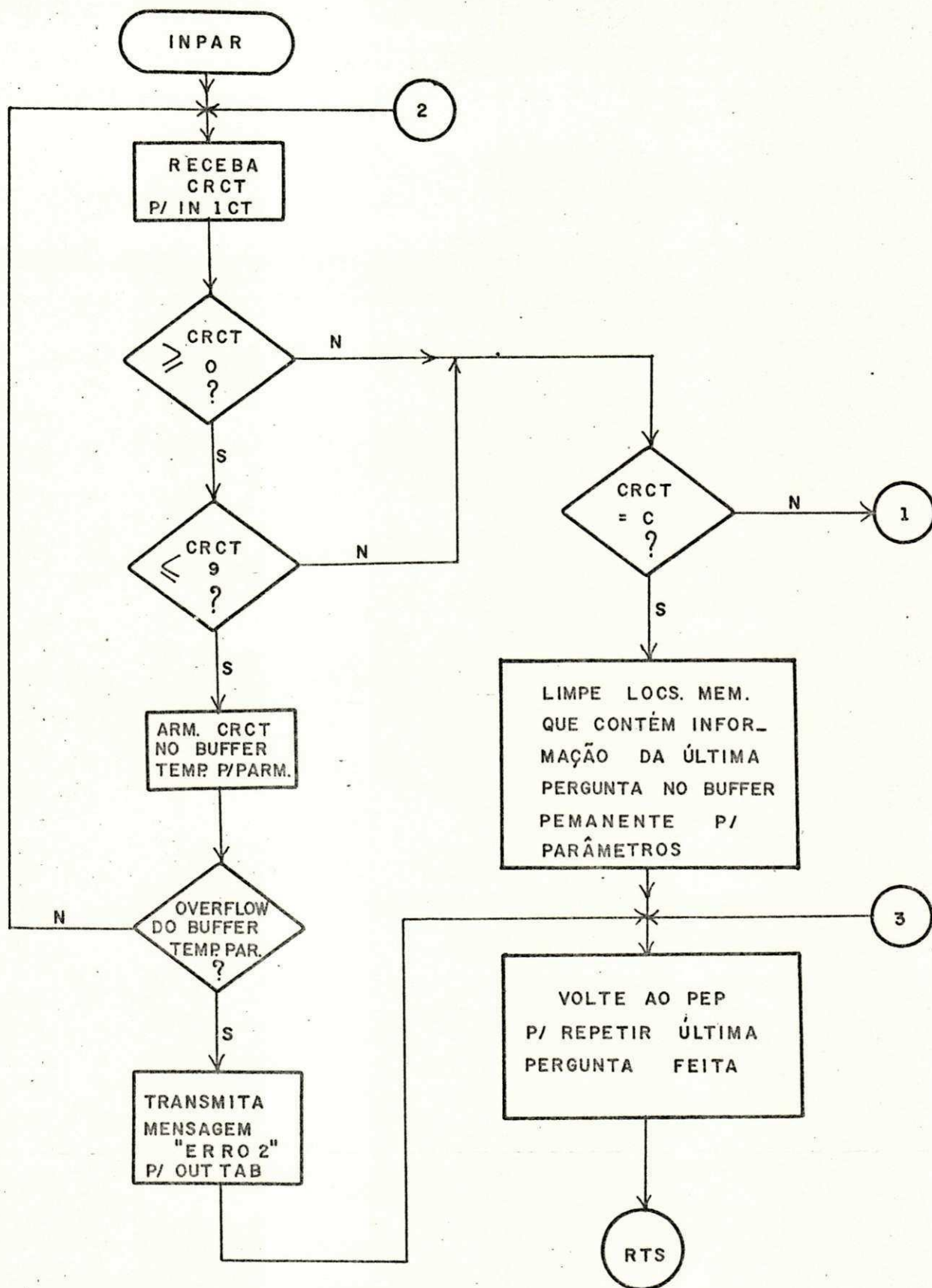














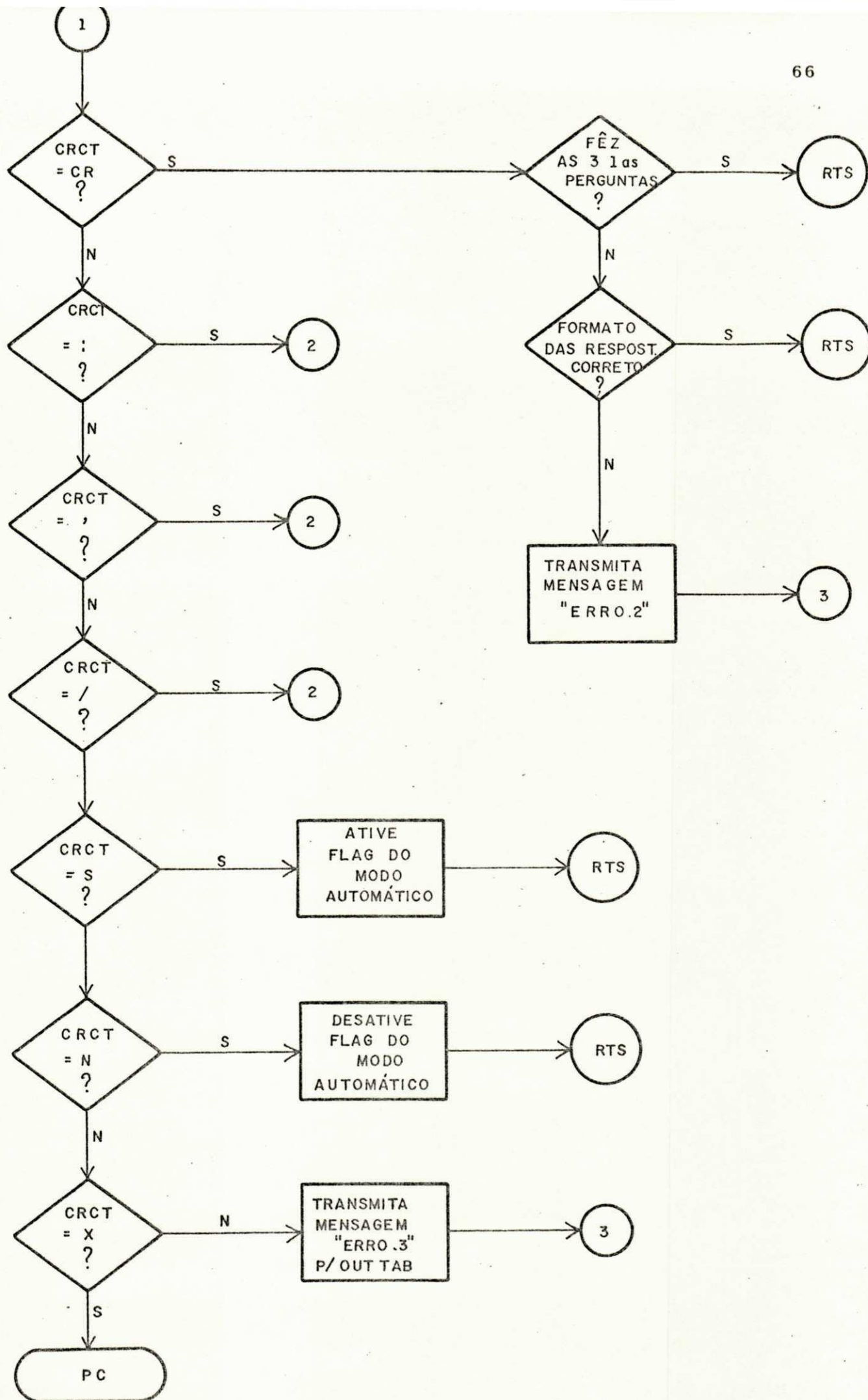


FIG. 3.9.2 - Fluxograma da SRT/INPAR

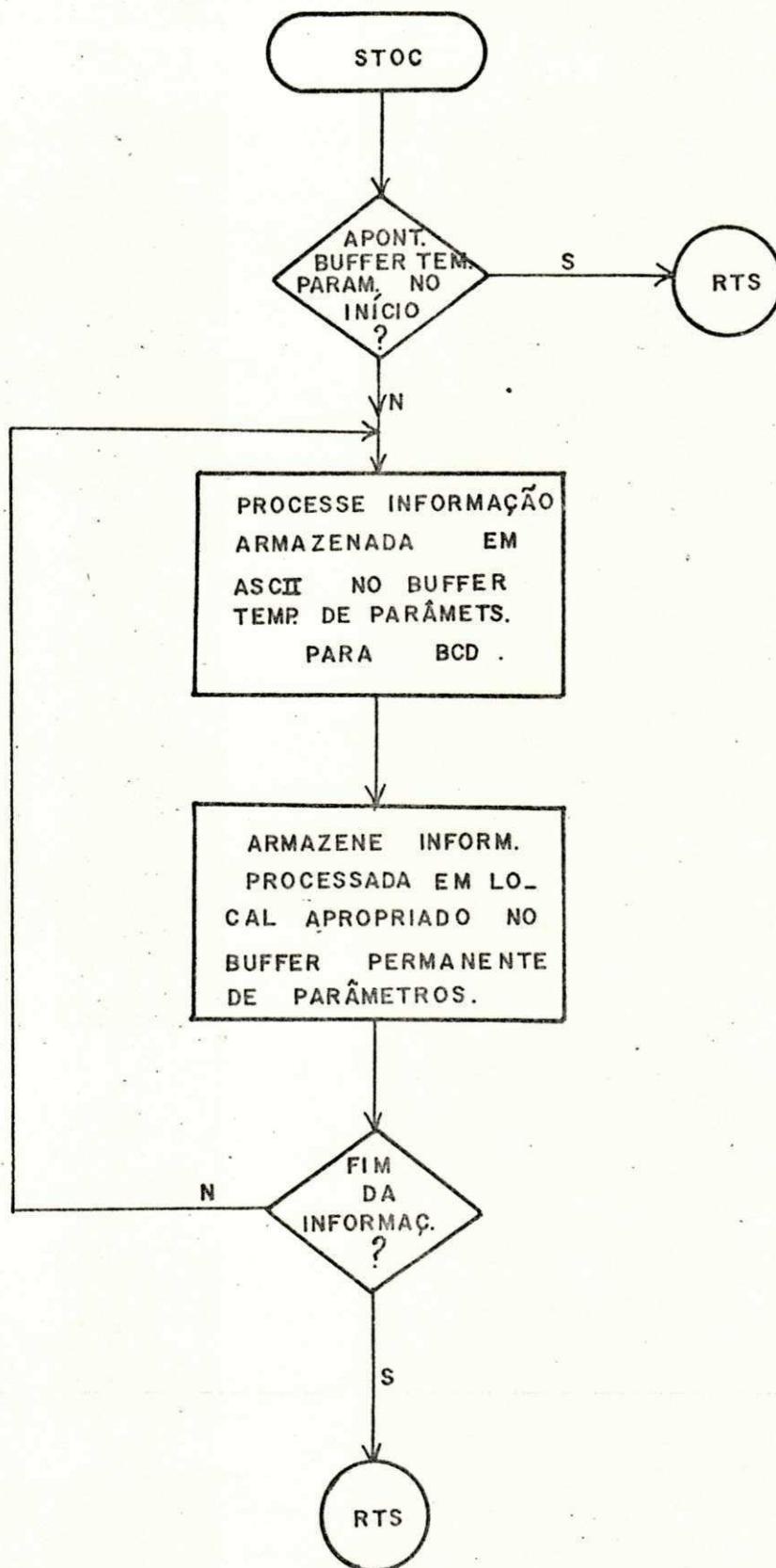


FIG. 3.9.3 - Fluxograma da Subrotina que Processa e Armazena Parâ

### 3.10 - PROGRAMA DE "RESET" DOS PARÂMETROS - PRP

A função deste programa (veja fig. 3.10) é "limpar" as posições de memória onde são armazenados os parâmetros e vetores de atividade, necessários para o funcionamento adequado deste sistema, fornecidos pelo operador através do PEP (item 3.9).

Entenda-se por "limpar" as posições de memória a ação de:

- a) tornar zeros os bits das locações de memória destinadas ao "buffer" permanente de parâmetros e ao vetor de atividade das torres e bases (VATTB) e
- b) tornar uns todos os bits das locações de memória destinadas aos vetores de atividade dos instrumentos das torres e bases (VATT1 a VATT5 e VATB1 a VATB3).

Esta tarefa não é executada pelo PARES (item 3.2) pelo simples fato de que sempre que o operador pedisse um "reset" externo, para realizar nova série de aquisições de dados teria que definir todos os parâmetros e vetores de atividade novamente.

Deste modo, o operador pode pedir o estado atual dos parâmetros e vetores de atividade através do PSP (veja item 3.11) e fazer apenas modificações desejadas para realização de nova série de aquisições de dados.

Assim, para a realização de uma série de aquisições de dados, as seguintes sequências devem ser observadas:

- a) quando da inicialização ou energização do sistema, deve-se primeiro requisitar o PRP, para então definir-se os parâmetros do PEP;
- b) quando da reinicialização do sistema após o "reset" externo, deve-se primeiro requisitar o PSP, para então modificar-se os parâmetros desejados através do PEP.

### 3.11 - PROGRAMA DE "STATUS" DOS PARÂMETROS - PSP

Tem como finalidade fornecer ao operador, quando necessário, o estado atual dos parâmetros e vetores de atividade do sistema de aquisição de dados (veja fig. 3.11.1).

Este programa simplesmente transmite tabelas que definem os



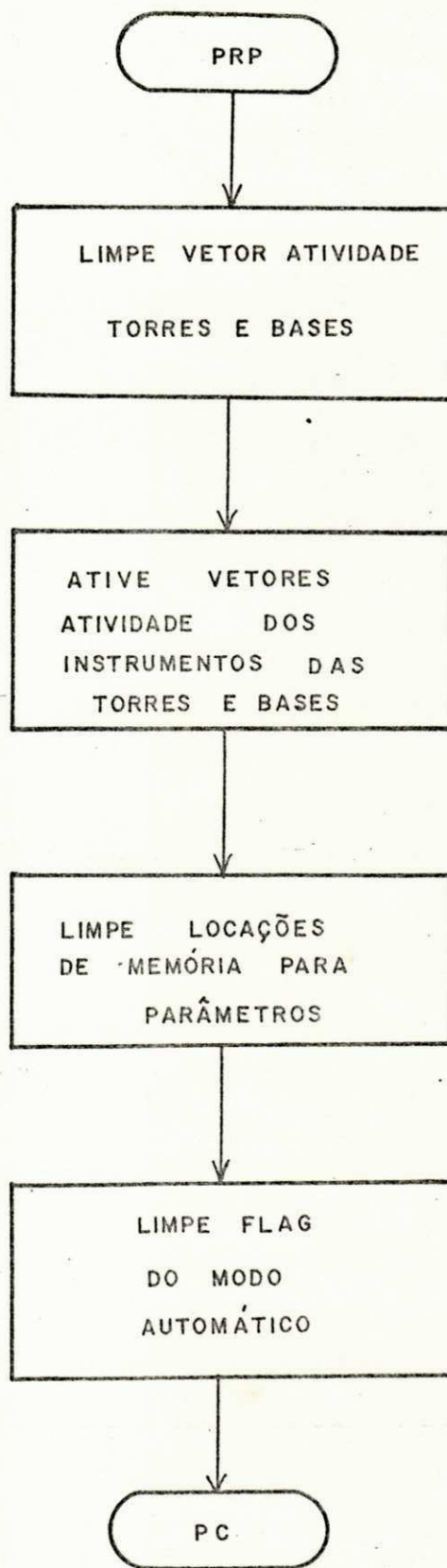


FIG. 3.10 - Fluxograma do Programa de "Reset" dos Parâmetros (PRP).

parâmetros e os vetores de atividade. Após a transmissão de cada tabela que define um parâmetro, o PSP chama a SRT/SENDP (subrotina que envia parâmetros, veja fig. 3.11.2), a qual processa a informação armazenada em BCD para ASCII e a transmite pela SRT/OUTICT.

Após a transmissão de cada tabela que define um vetor de atividade, o PSP chama a SRT/SENDVA (subrotina que envia Vetores de Atividade, veja fig. 3.11.3), a qual processa a informação armazenada em binário para ASCII e a transmite pela SRT/OUTICT.

Veja abaixo um exemplo da listagem sobre o estado do sistema que seria fornecida pelo PSP, se requisitada pelo operador após a definição dos parâmetros e vetores de atividade do sistema vista no exemplo do item 3.9:

SADNE-STATUS

INT MEDIÇÕES T = 10:15

No MEDIÇÕES N = 0100

DENSIDADE AR = 01,12

HORA: 10:40:00

DATA: 10/01/81

MODO FUNCIONAMENTO: AUTO

ATIVIDADE TORRES E BASES

VATTB: 00100011

VATT1: 11111111

VATT2: 11010011

VATT3: 11111111

VATT4: 11111111

VATT5: 11111111

VATB1: 11111111

VATB2: 11111111

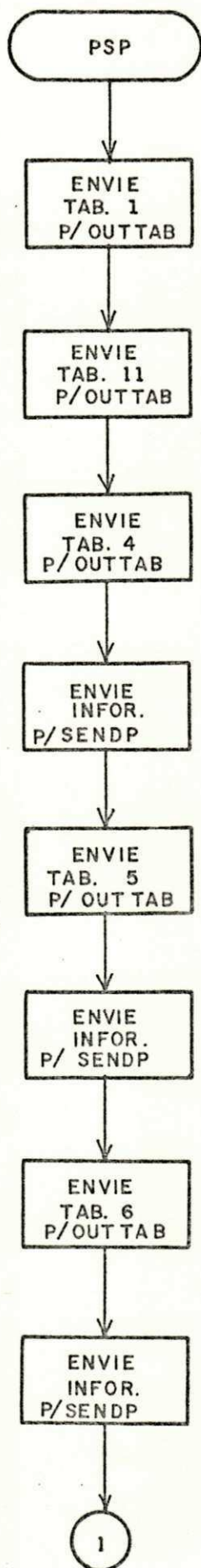
VATB3: 11111111

Nota 1.

1 = ativo

0 = inativo

Na listagem, HORA e DATA indicam o horário e a data da impressão desta listagem.



SADNE-

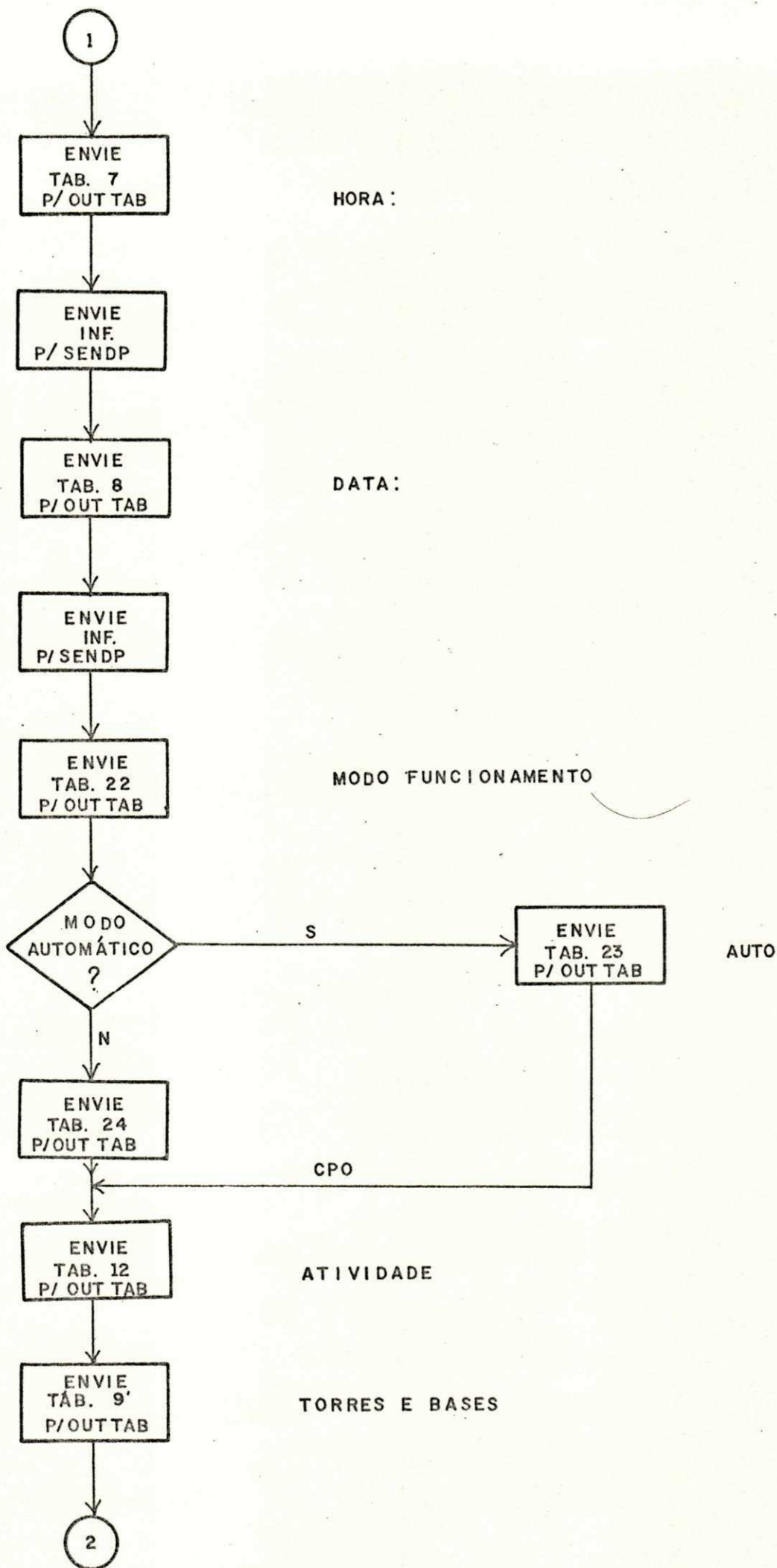
STATUS

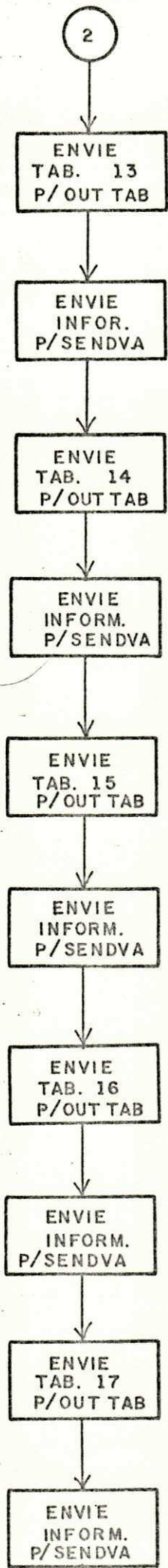
INT MEDIÇÕES T =

Nº MEDIÇÕES N =

DENSIDADE AR =







VATTB:

VATT1:

VATT2:

VATT3:

VATT 4:

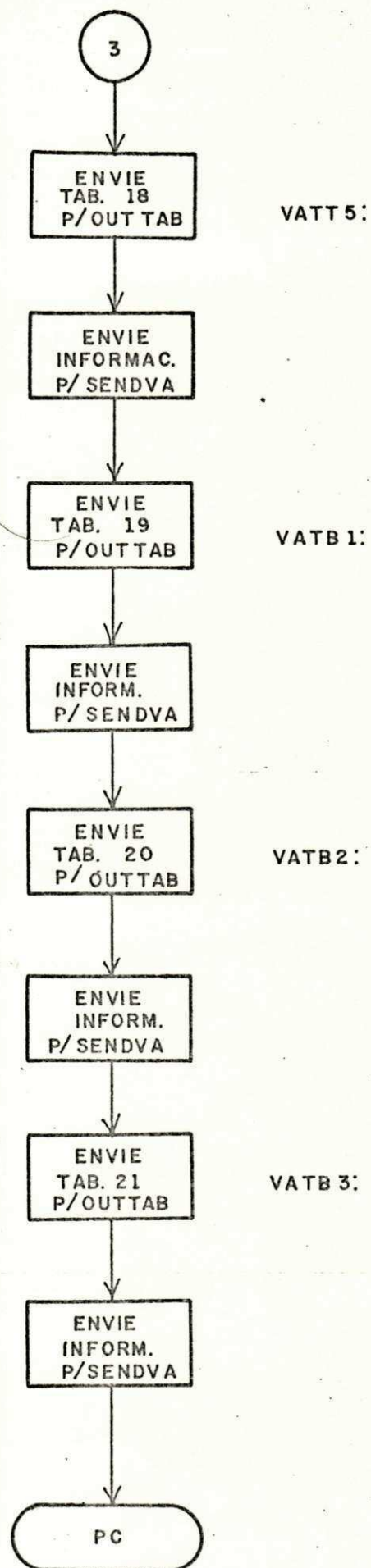
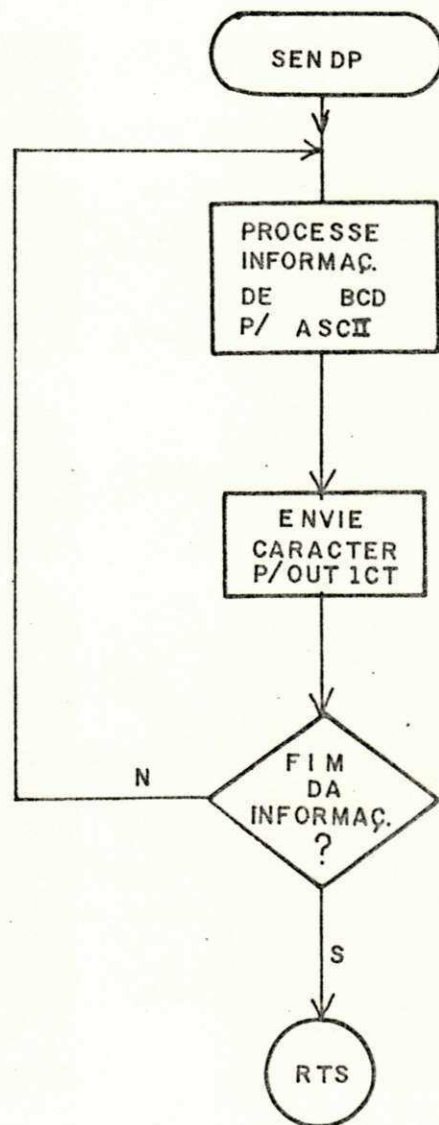


FIG 3 11 1 - Fluxograma do PSP





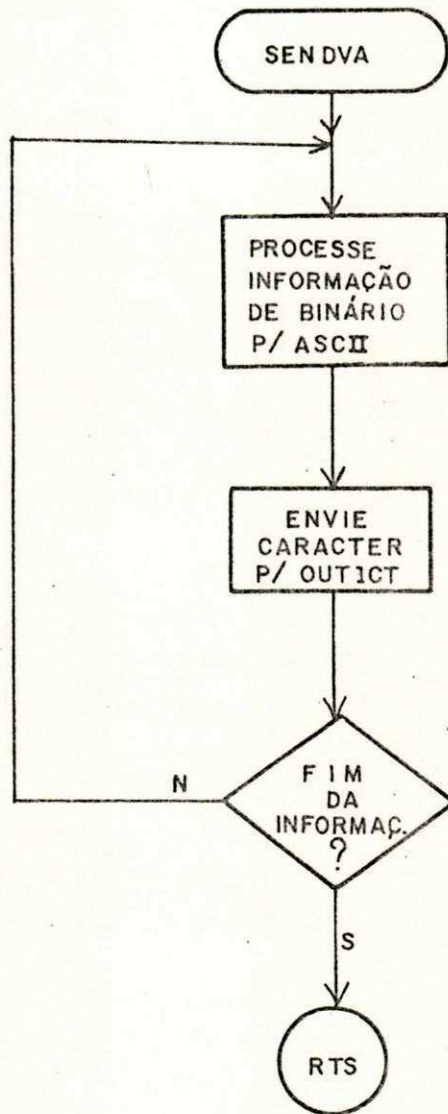


FIG. 3.11.3 - Fluxograma da Subrotina que Envia Vetores de Atividade (SRT/SENDVA)

### 3.12 - GERENTE DA AQUISIÇÃO DE DADOS - GAD

Este programa foi denominado de Gerente da Aquisição de Da dos pelo fato do mesmo ser responsável pela coordenação e supervi são da aquisição de dados (veja fig. 3.12.1).

São tarefas do GAD:

- a) preparar os registradores provisórios para medição do tempo da aquisição de dados,
- b) preparar os registradores provisórios para contagem do número de aquisições de dados e executar esta contagem,
- c) preparar o apontador do "buffer" da aquisição de dados,
- d) ativar e desativar o "flag" da aquisição de dados, o qual sinaliza o PANMI para que o mesmo execute a contagem do tempo da aquisição de dados,
- e) ler e armazenar o horário e a data do início e do término de uma série de aquisições de dados, informação que deve constar no relatório fornecido pelo sistema ao operador,
- f) sinalizar as interfaces dos anemômetros para a aquisição de dados e
- g) verificar qual o modo de funcionamento do sistema no final da aquisição de dados para chamar o programa apropriado.

#### 3.12.1 - PROGRAMA DE AQUISIÇÃO DE DADOS - PAD

O Programa de Aquisição de dados, embora seja uma subrotina, foi assim denominado por executar uma importante tarefa do sistema de aquisição de dados.

O PAD executa as funções de aquisitar dados e armazená-los no "buffer" da aquisição de dados, que é sequencial. Neste "buffer" os dados adquiridos são armazenados um após o outro, independentemente do tipo de instrumento e de a qual torre ou base este pertence.

Assim, as medições de um mesmo instrumento, realizadas durante uma série de aquisições de dados, estão armazenadas a distância.



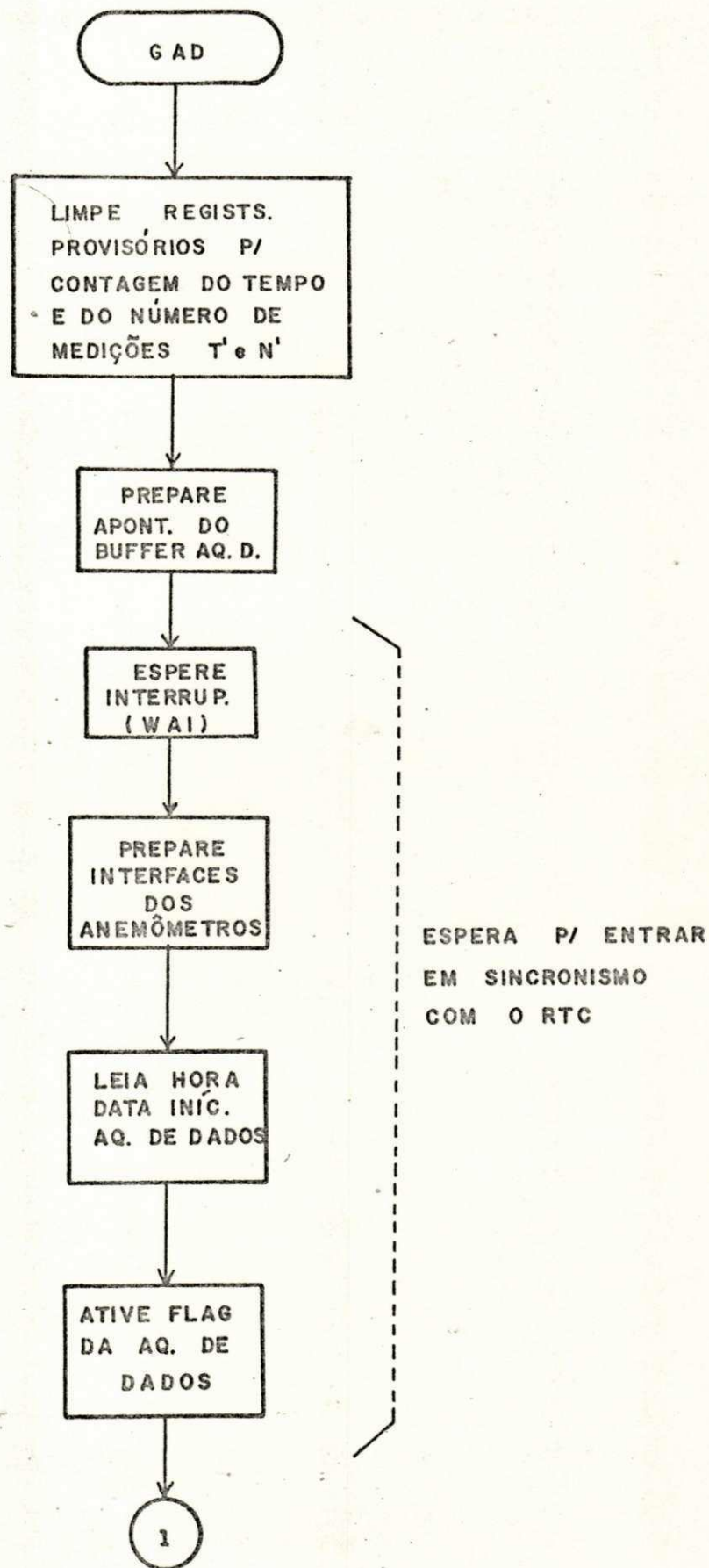


FIG. 3.12.1 - Fluxograma do Gerente da Aquisição de Dados (GAD)  
(Pág. 1 de 2)

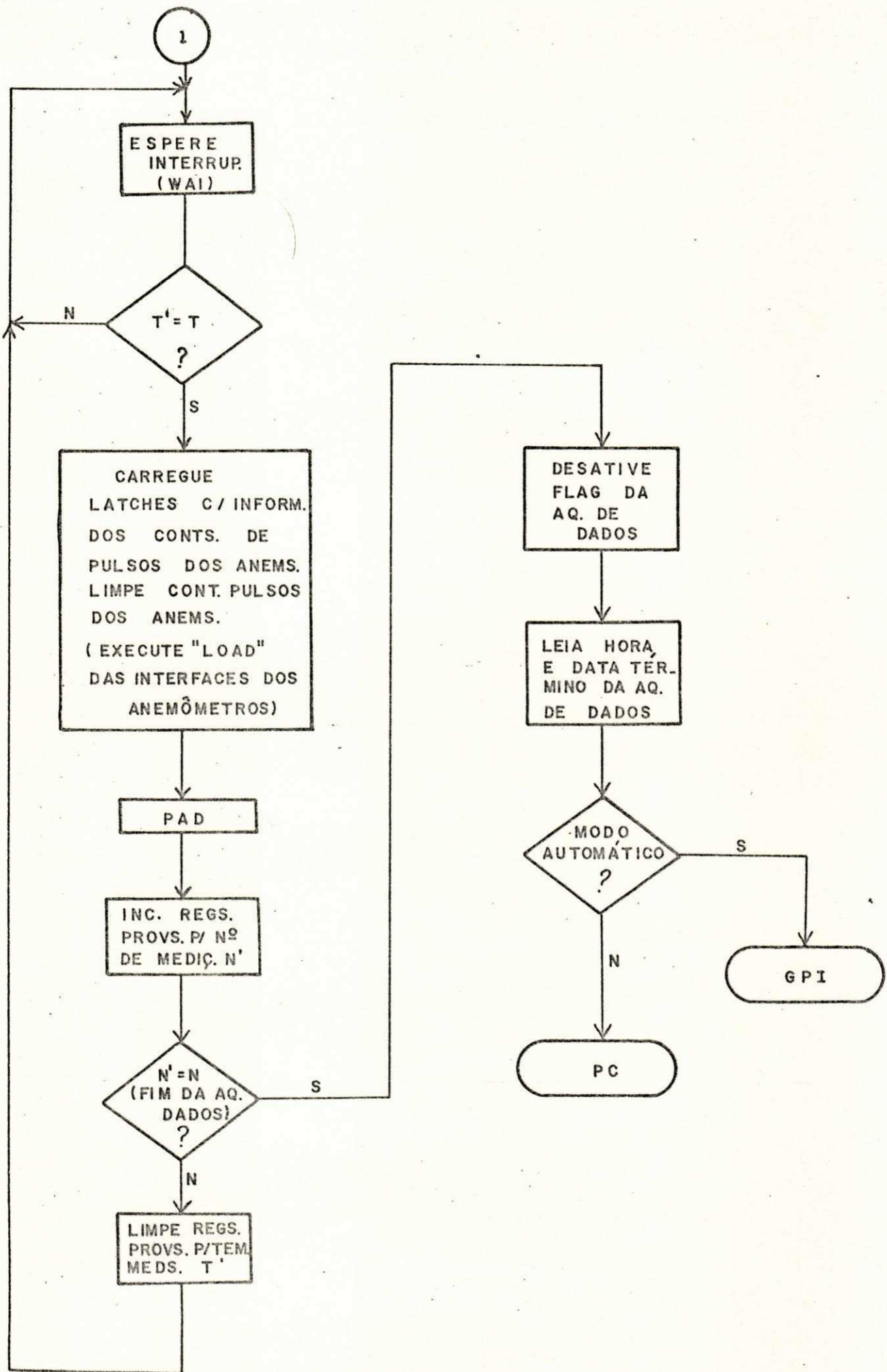


FIG. 3.12.1 - Fluxograma do GAD  
(Pág. 2 de 2)

cias fixas e bem determinadas umas das outras. Para um determinado número  $N$  de medições, forma-se uma sequência de armazenamento que depende da atividade das torres e bases do sistema e que se repete  $N$  vezes.

Veja a seguir como são armazenados os dados adquiridos numa série de aquisições de dados onde o número de medições  $N = 3$  e os vetores de atividade são assim definidos:

VATTB = 00000111

VATT1 = 00000111

VATT2 = 00000001

VATT3 = 00001100

Neste exemplo, a abreviação utilizada ( $ANiTn$ ) significa Anemômetro "i" da torre "n" (i varia entre 1 e 6 e n varia entre 1 e 5).

#### ARMAZENAMENTO NO "BUFFER" SEQUENCIAL

AN1T1	}	N = 1
AN2T1		
AN3T1		
AN1T2		
AN3T3		
AN4T3		
AN1T1	}	N = 2
AN2T1		
AN3T1		
AN1T2		
AN3T3		
AN4T3		
AN1T1	}	N = 3
AN2T1		
AN3T1		
AN1T2		
AN3T3		
AN4T3		



Observe-se a formação da sequência de armazenamento e a existência da distância fixa entre os armazenamentos de uma medição de um instrumento qualquer e da próxima medição do mesmo instrumento. Como estão previstos 2 bytes para armazenamento de cada medição, para o exemplo visto, esta distância entre armazenamentos de medições de um mesmo instrumento é de  $6 \times 2 = 12$  locações de memória porque existem 6 instrumentos ativos.

Existe também uma distância de armazenamento das medições dos instrumentos de uma determinada torre ou base ativa e a próxima torre ou base. No exemplo dado, a distância de armazenamento entre a torre 1 a torre 2 é de  $3 \times 2 = 6$  locações de memória (existem 3 instrumentos ativos na torre 1), já a distância de armazenamento entre as torres 2 e 3 é de  $1 \times 2 = 2$  (existe apenas um instrumento ativo na torre 2). Estas distâncias de armazenamento das medições são utilizadas pelo Gerente do Processamento da Informação (item 3.13) para calcular o apontador do "buffer" da aquisição de dados para o seu processamento.

No PAD (veja fig. 3.12.2) existe um teste para detetar o final do "buffer" da aquisição de dados com o objetivo de evitar que se perca informação. Este "buffer" foi dimensionado para armazenar até 100 medições de 64 instrumentos de 16 bits, possuindo assim 12.800 locações de memória.

### 3.13 - GERENTE DO PROCESSAMENTO DA INFORMAÇÃO - GPI

O Gerente do Processamento da Informação (Fig. 3.13.1) é responsável pela coordenação do processamento dos dados adquiridos pelo PAD.

Uma função do GPI é calcular a distância entre os armazenamentos de uma medição de um instrumento e da próxima medição do mesmo instrumento e as distâncias de armazenamento entre cada torre ou base e a próxima (veja exemplo do item 3.12.1). O objetivo deste procedimento é calcular o apontador do "buffer" da aquisição de dados para que os dados sejam retirados do mesmo, para que possam ser processados.

É também função do GPI a verificação da atividade das torres e bases e seus instrumentos para chamar os programas de proces

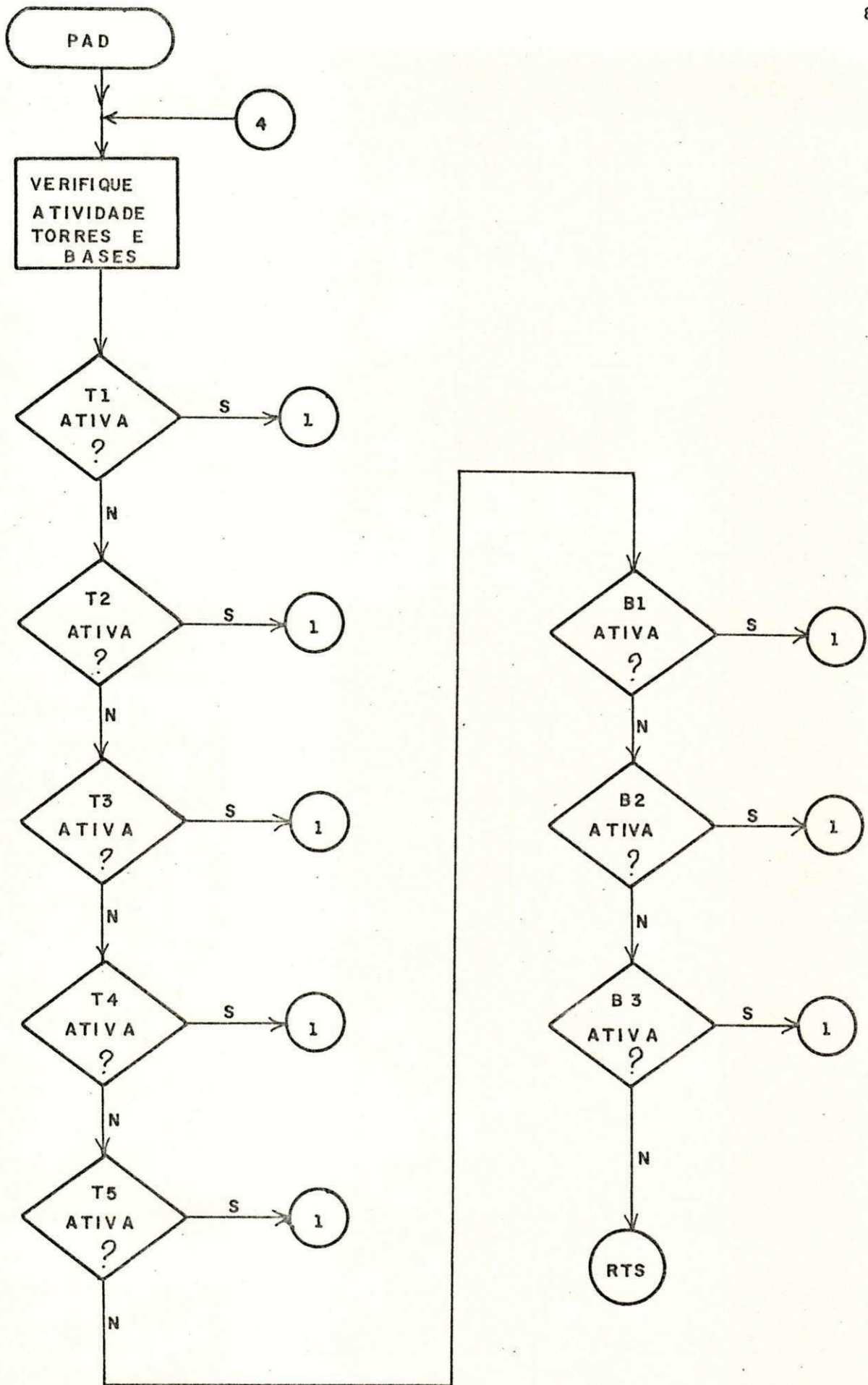


FIG. 3.12.2 - Fluxograma do Programa de Aquisição de Dados (PAD)  
(Pág. 1 de 4)

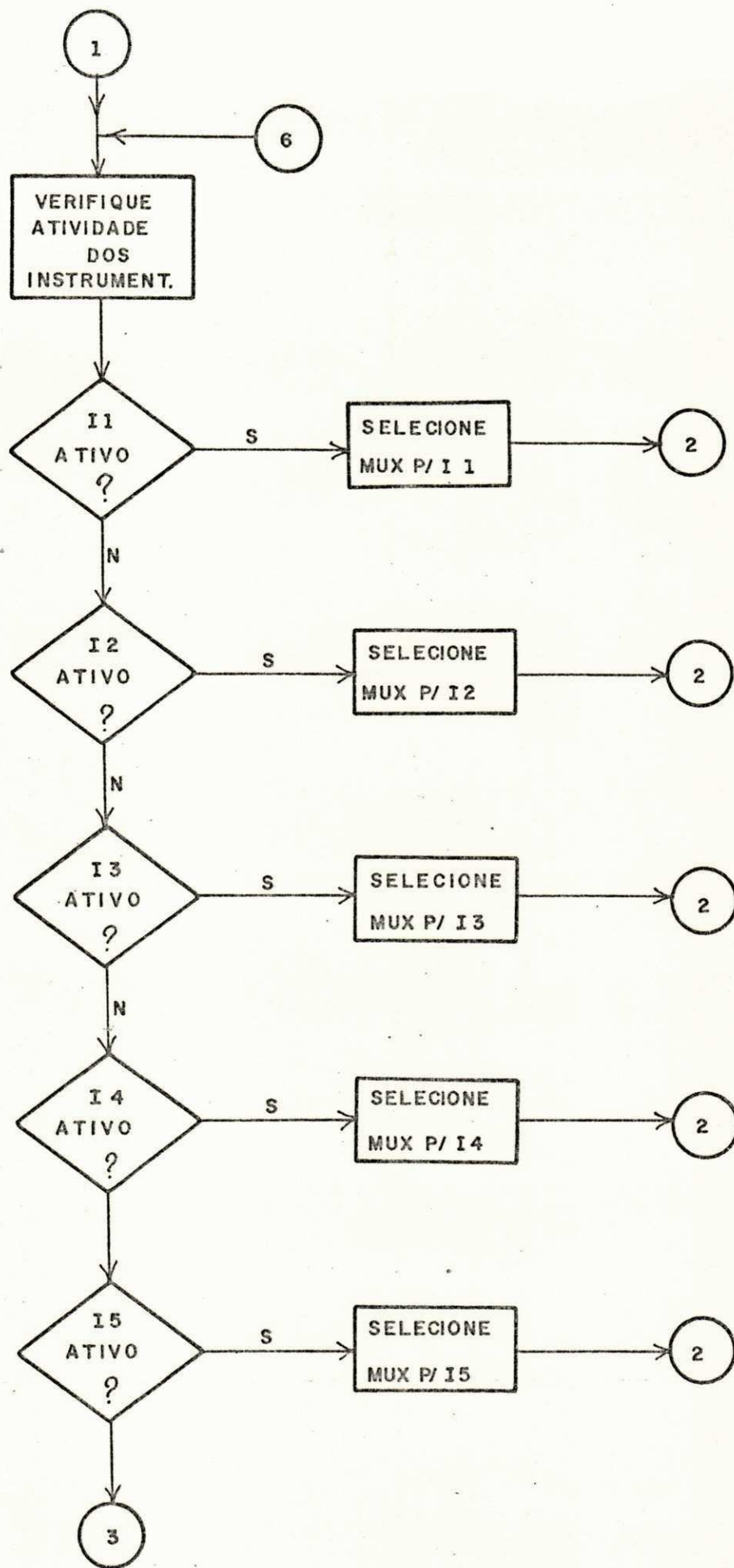


FIG. 3.12.2 - Fluxograma do PAD  
(Pág. 2 de 4)



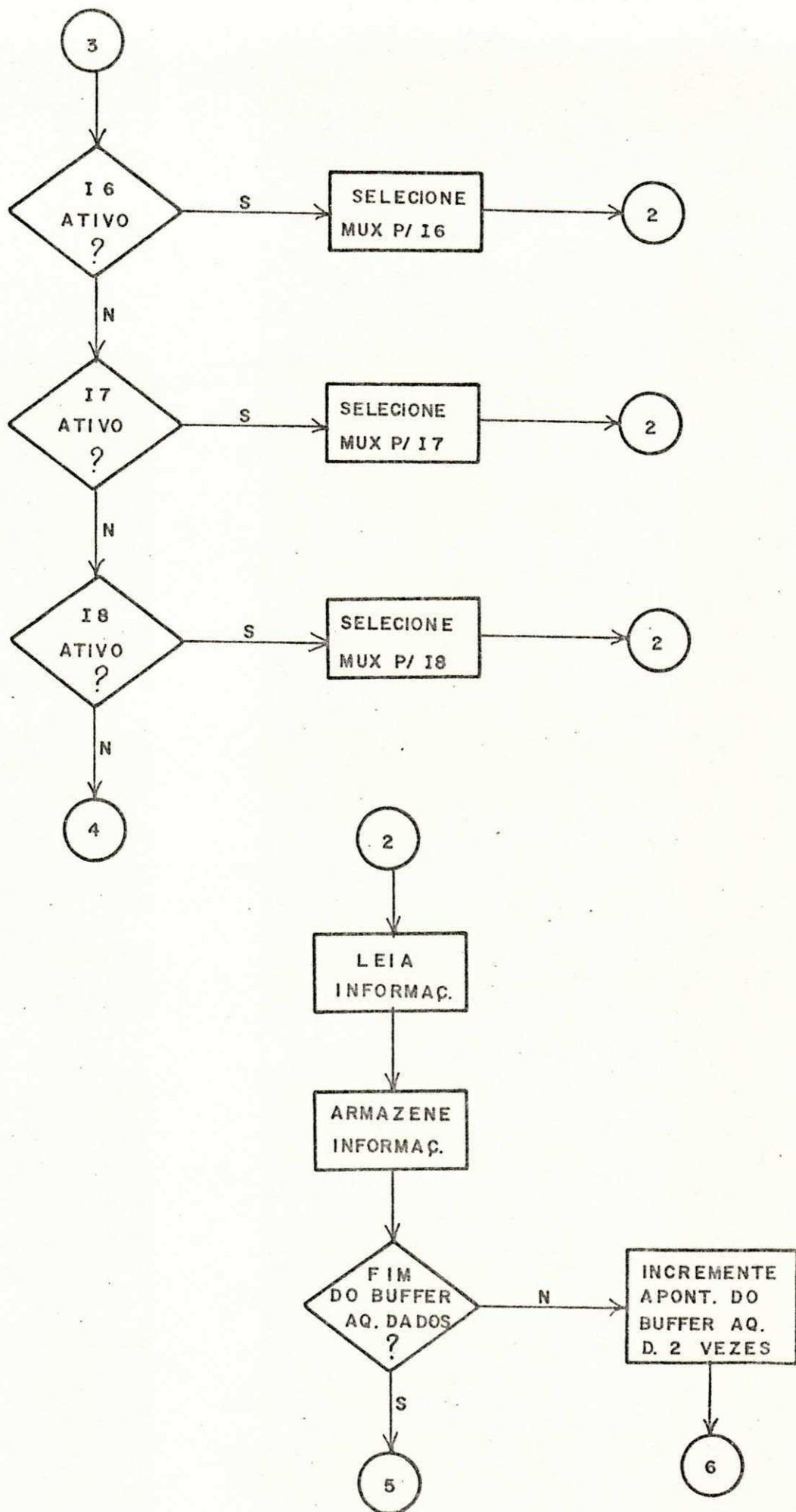


FIG. 3.12.2 - Fluxograma do PAD  
(Pág. 3 de 4)

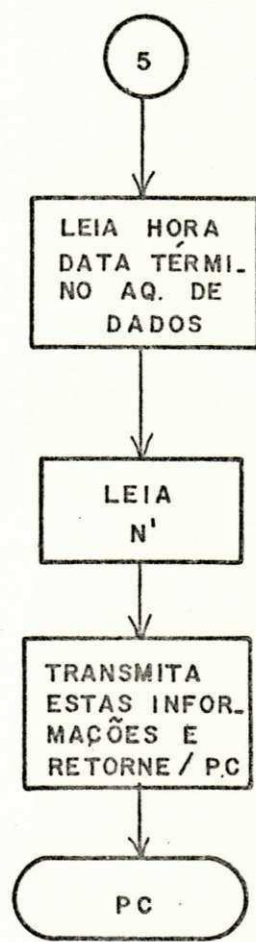


FIG. 3.12.2 - Fluxograma do PAD  
(Pág. 4 de 4)

samento específicos para cada tipo de instrumento.

Os programas de processamento são responsáveis pela conversão dos dados adquiridos para unidades reais e pelo armazenamento e formatação dos dados processados para a impressão posterior do relatório técnico.

Em alguns casos cálculos adicionais precisam ser realizados. Por exemplo, para os anemômetros devem ser calculados: a velocidade média do vento, o desvio padrão, o momento centrado de terceira ordem e a potência média do vento. Estes cálculos envolvem operações de adição, subtração, multiplicação, divisão, extração de raízes quadrada e terceira ordens, tudo com precisão de duas casas decimais. Como o microprocessador não possui instruções específicas para realizar tais cálculos, rotinas devem ser desenvolvidas. O desenvolvimento destas rotinas não faz parte deste trabalho. No caso dos instrumentos das bases de turbinas, o NERG ainda não especificou que processamento deveria ser realizado para os dados destes instrumentos, no entanto, a estrutura modular do "software" permite que estes programas sejam acrescentados posteriormente sem maiores dificuldades.



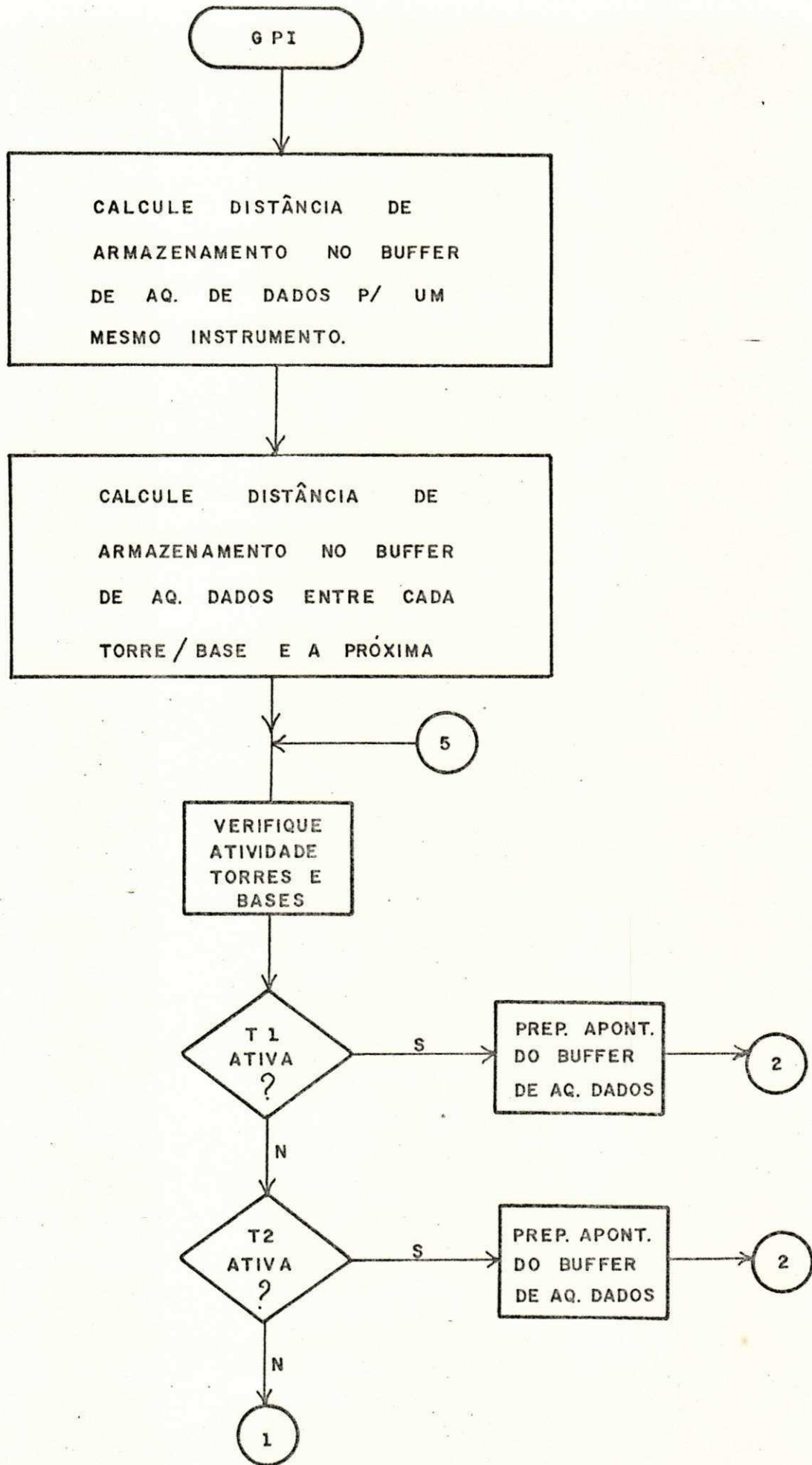


FIG. 3.13.1 - Fluxograma do Gerente do Processamento da Informação (GP (Pág. 1 de 5)

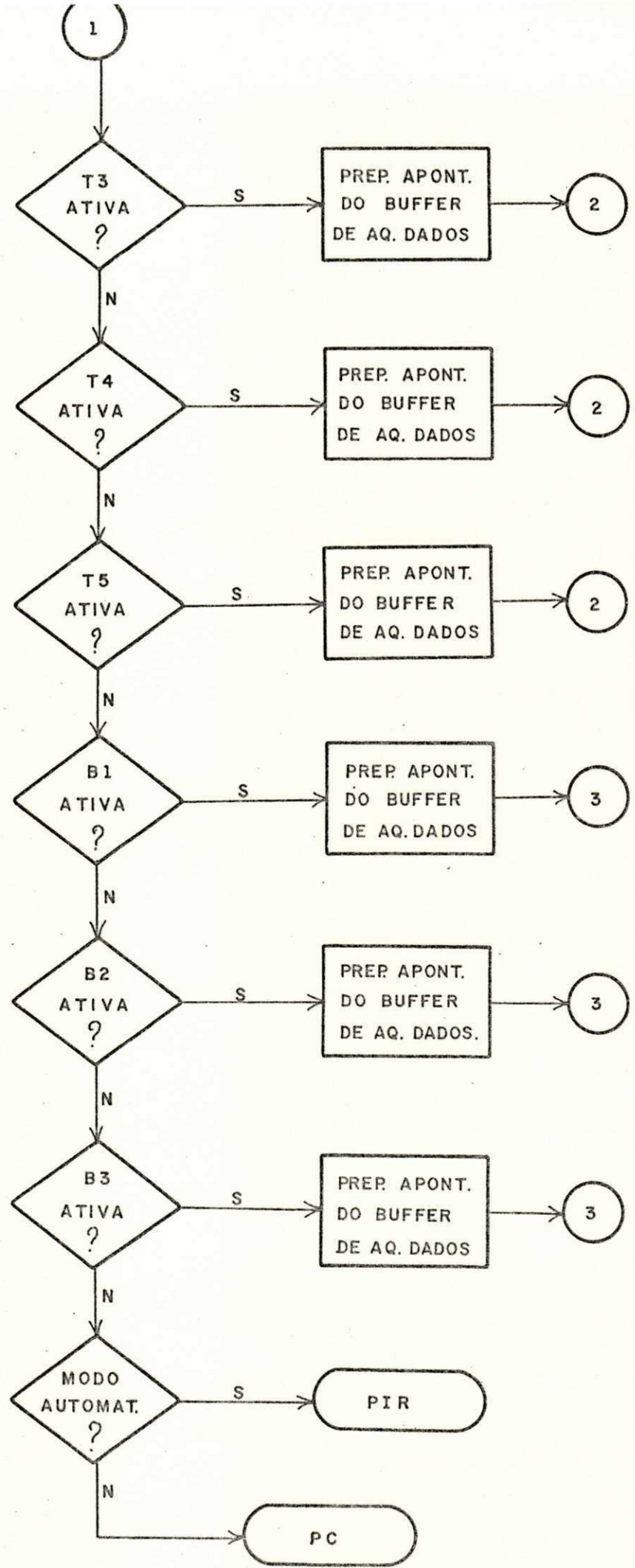


FIG. 3.13.1 - Fluxograma do GPI  
(Pág. 2 de 5)

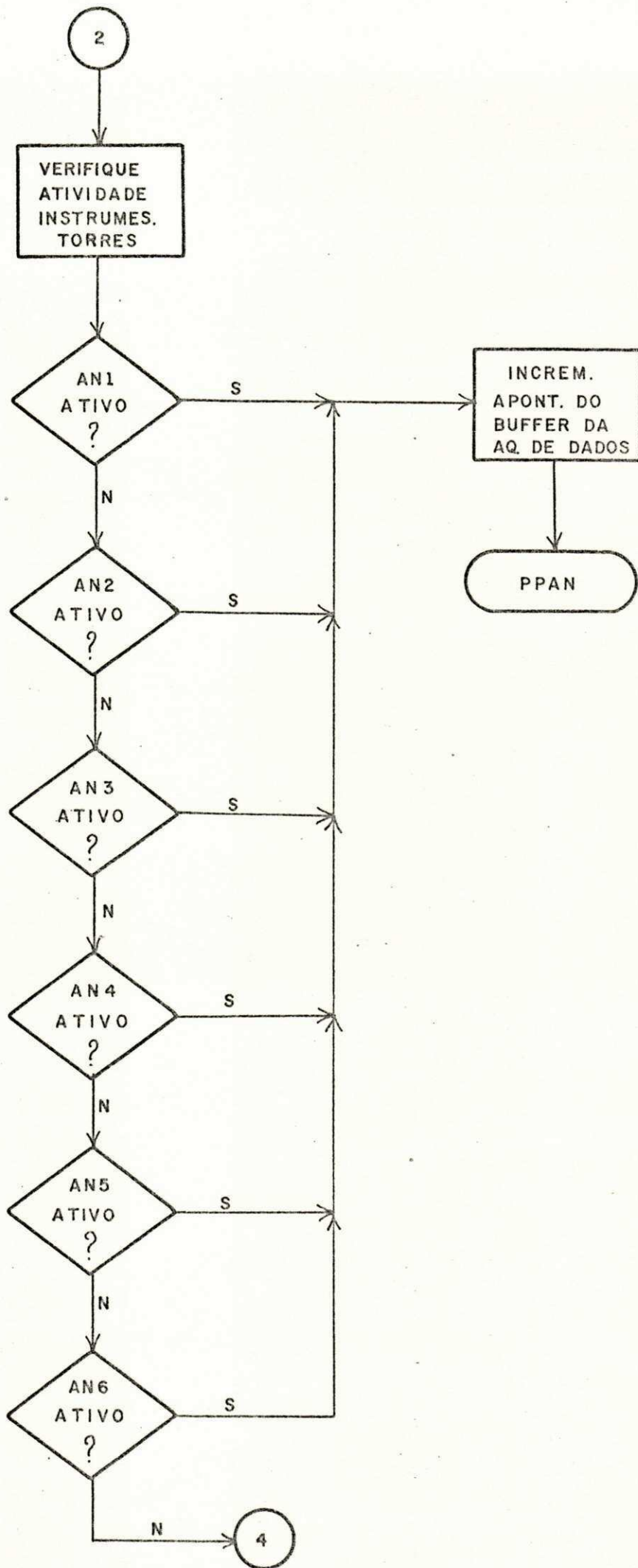


FIG. 3.13.1 - Fluxograma do GPI  
(Pág. 3 de 5)



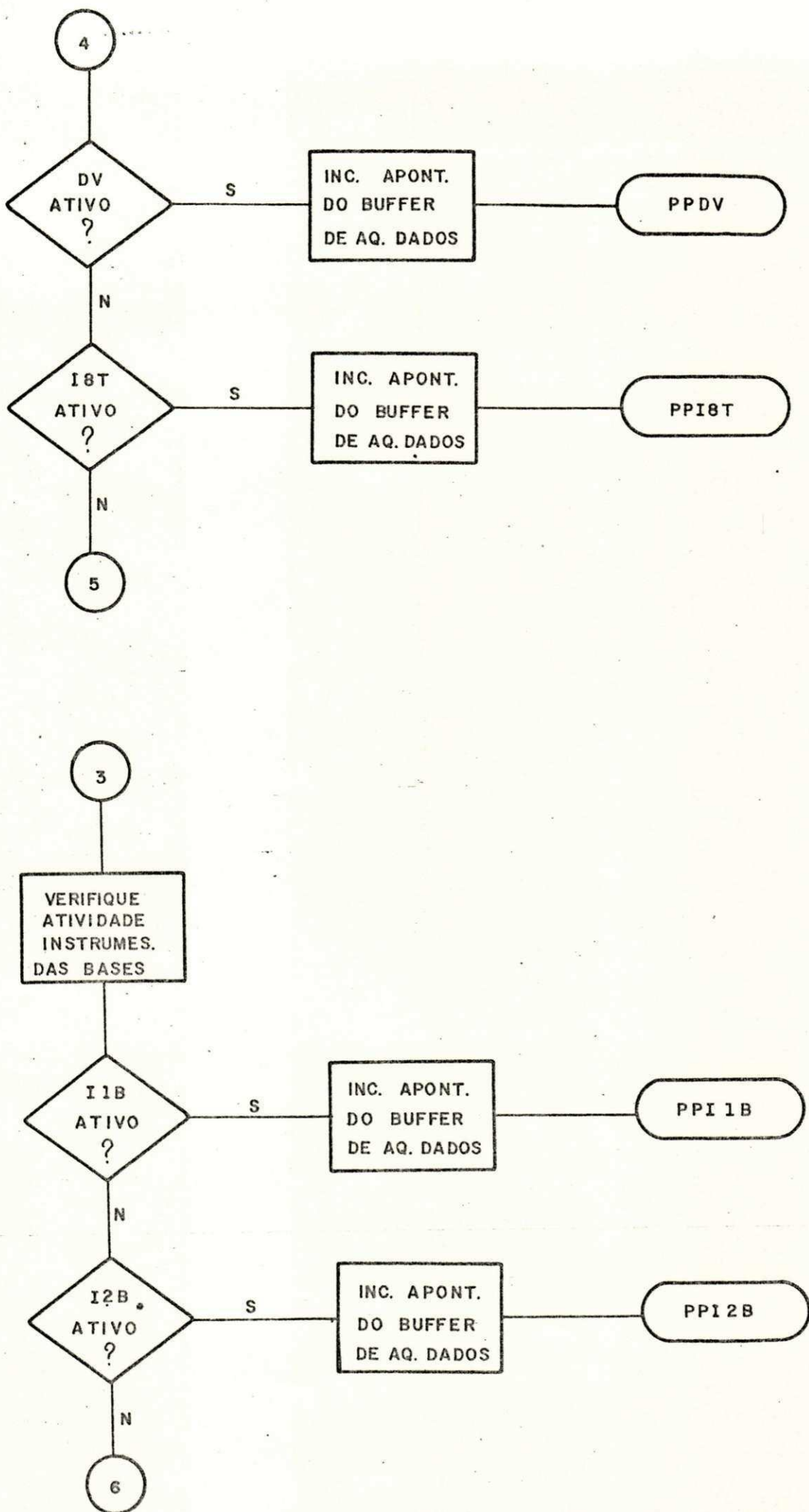


FIG. 3.13.1 - Fluxograma do GPI  
(Pág. 4 de 5)

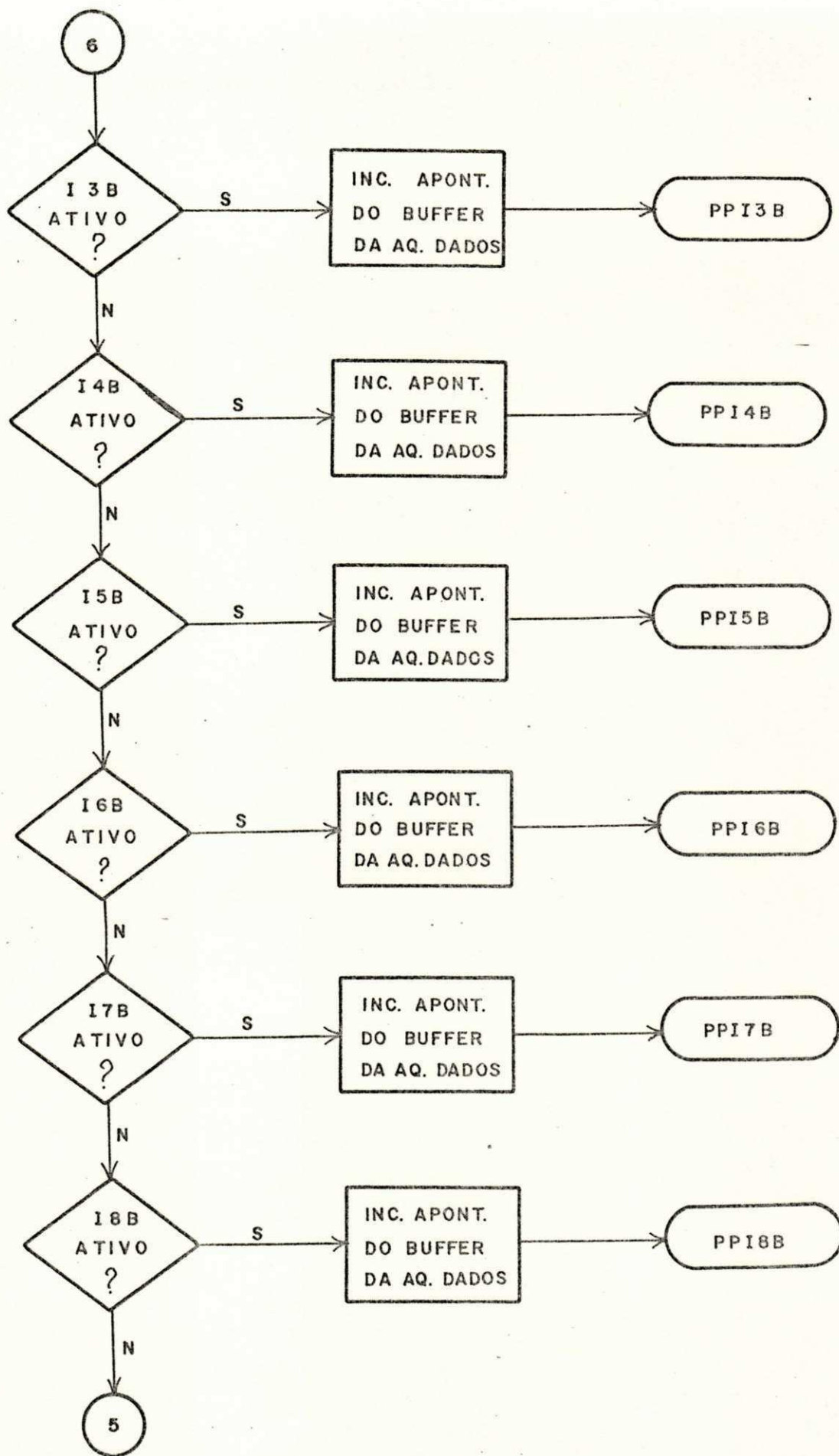


FIG. 3.13.1 - Fluxograma do GPI  
(Pág. 5 de 5)

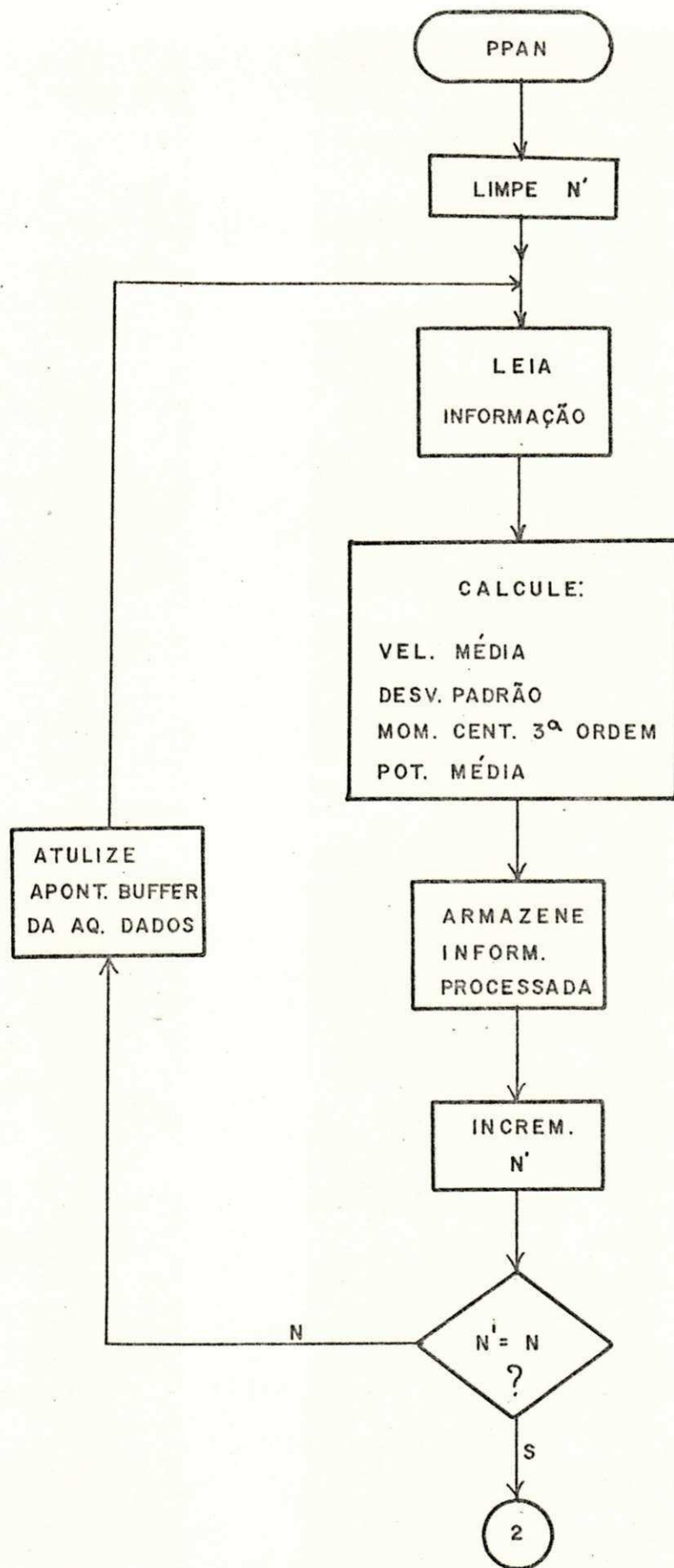


FIG. 3.13.2 - Fluxograma do Programa de Processamento da Informação do Anemômetro (PPAN)



## CAPÍTULO IV

### CONCLUSÃO

O sistema de aquisição de dados, relatado neste trabalho, foi projetado com base nas especificações estabelecidas pelo NERG. No entanto, devido à flexibilidade oferecida pelos microprocessadores em geral e à característica modular do projeto deste sistema em especial, modificações nas especificações e na operacionalidade do SADNE são permitidas.

Devido à utilização da via comum ("parallel-wired bus"), o cartão da CPU e demais cartões funcionais podem ser conectados ao armário que contém o sistema sem qualquer ordem de conexão, permitindo também um número destes cartões limitado apenas pela capacidade ("fanout") de alimentação dos "buffers" do sistema.

Com relação aos cartões de interface para aquisição de dados, o número de instrumentos por cartão pode ser facilmente duplicado, bastando para isso acrescentar-se mais 16 multiplexadores do mesmo tipo utilizado (8 linhas para 1) e também mais uma linha para seleção de endereços nos multiplexadores. Claro que a resolução permitida para os instrumentos continua a mesma (4 dígitos decimais ou 16 bits). Quanto ao número destes cartões, sua expansão encontra-se limitada pela disponibilidade de locações de memória e capacidade de alimentação dos "buffers" utilizados.

A expansão da memória do sistema pode ser realizada com modificações nos decodificadores de endereços, muito embora isto provavelmente não se faça necessário devido a já grande quantidade de memórias existentes no sistema.

A alocação de novos periféricos é permitida e prevendo-se a aquisição de um terminal de vídeo para o sistema, já existe no cartão de E/S uma interface apropriada para conexão do mesmo ao sistema de aquisição de dados.

Os vários cartões ou módulos que compõem o sistema do

microcomputador estão montados em placas de "wire-wrap" e encontram-se em fase de testes. Por ser o "hardware" um tópico mais conhecido e também mais simples de testar-se, resolveu-se dar mais ênfase à depuração dos programas de "software".

A estrutura também modular do "software", onde um programa de controle coordena a execução das diversas tarefas do sistema por outros programas, permite que novos programas sejam introduzidos, uma vez que os mesmos são escritos em "módulos" independentes um do outro.

O intervalo de tempo T entre as aquisições de dados e o número de aquisições N a serem realizadas, os quais são programados pelo operador através do Programa de Entrada de Parâmetros (PEP), excedem em muito as especificações propostas. A estrutura do PEP permite que T varie entre 1 segundo e 99 minutos e 60 segundos (100 minutos) e que N varie entre 1 e 9999 vezes.

O PEP poderia ter a estrutura de sua primeira etapa modificada da seguinte maneira: após realizar-se uma pergunta, fornecer-se ao operador o conteúdo atual da mesma e colocar-se à disposição do operador para modificações, se necessárias. Resolveu-se utilizar o sistema do Programa de "Status" dos Parâmetros (PSP) para fornecer ao operador o estado atual e vetores de atividade do sistema, apenas com o intuito de economizar-se memória, uma vez que o mesmo PSP é utilizado no relatório técnico fornecido ao operador pelo Programa de Impressão de Relatórios (PIR) (veja formato do relatório técnico no apêndice V).

A utilização dos Vetores de Atividade mostrou-se bastante prática e econômica. O número de cartões de interface de aquisição de dados e também o número de instrumentos por cartão poderiam ser duplicados, bastando para isso, em termos de "software", utilizarem-se Vetores de Atividade de 2 bytes ("high" e "low").

A existência do Gerente de Aquisição de Dados (GAD) permite que modificações na operacionalidade da aquisição de dados sejam realizadas sem ser preciso modificar-se o Programa de Aquisição de Dados (PAD). O PAD apresentou um bom desempenho, mesmo para aquisições de dados com intervalos de tempo de 1 segundo. Graças ao GAD que prepara as interfaces dos instrumentos para a aquisição de dados e também à maneira simples como os dados são



armazenados no "buffer" da aquisição de dados, o PAD tem condições de realizar aquisições de dados em todos os 64 instrumentos de todas as 5 torres e 3 bases de turbinas, executando apenas uma média de 600 instruções (entre deslocamentos, seleção de endereços para os multiplexadores dos cartões de interfaces de aquisição, leituras e armazenamentos de dados com 2 bytes) com um número médio de 5 ciclos da MPU por instrução, gastando um tempo de 3.000 microsegundos que mostra-se insignificante comparado com 1 segundo.

Sugere-se que este sistema de aquisição de dados baseado em microprocessador, dedicado e "stand-alone" seja futuramente conectado ao PDP/11 existente no Campus II da UFPb, permitindo assim que um pesquisador situado no campo de testes de turbinas, onde este sistema será utilizado, possa utilizar toda a capacidade computacional e recursos oferecidos pelo PDP/11, enquanto o SADNE realiza outras tarefas.

Usuários de sistemas baseados em microprocessadores deveriam exigir um sistema de diagnose em "software" com a finalidade de detectar falhas nos instrumentos e auxiliar a manutenção. Sistemas de diagnose executam ajustes em amplificadores operacionais e calibração de conversores A/D, testam periféricos e instrumentos conectados ao sistema, etc. A inserção de um sistema de diagnose neste sistema de aquisição de dados seria justificável e de grande ajuda para a manutenção do mesmo.



APÊNDICE I

IMPLEMENTAÇÃO DO "HARDWARE" DO SISTEMA

A P Ê N D I C E    I I

DESCRIÇÃO DOS PINOS, ORGANIZAÇÃO INTERNA E CONJUNTO DE INSTRU-  
ÇÕES DO MICROPROCESSADOR MC6800

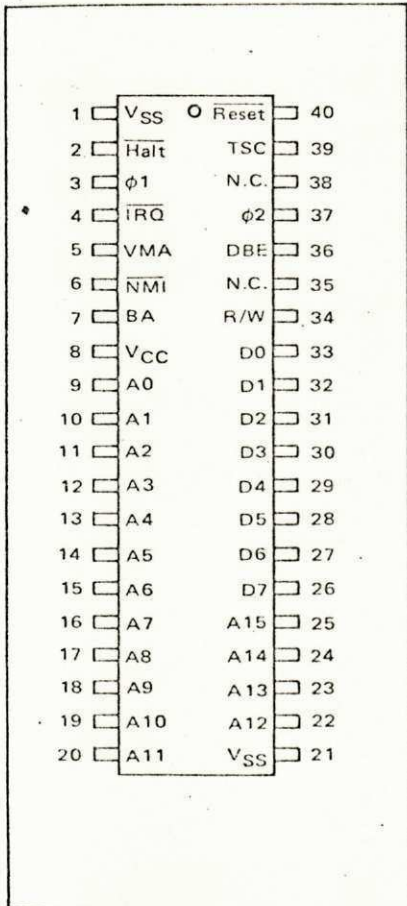
TABLE 6 - CONDITION CODE REGISTER MANIPULATION INSTRUCTIONS

OPERATIONS	MNEMONIC	IMPLIED			BOOLEAN OPERATION	COND. CODE REG.							
		OP	~	#		5	4	3	2	1	0		
						H	I	N	Z	V	C		
Clear Carry	CLC	0C	2	1	0 → C	•	•	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0 → I	•	R	•	•	•	•	•	•
Clear Overflow	CLV	0A	2	1	0 → V	•	•	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1 → C	•	•	•	•	•	•	•	S
Set Interrupt Mask	SEI	0F	2	1	1 → I	•	S	•	•	•	•	•	•
Set Overflow	SEV	0B	2	1	1 → V	•	•	•	•	•	•	S	•
Accmltr A → CCR	TAP	06	2	1	A → CCR	(12)						•	
CCR → Accmltr A	TPA	07	2	1	CCR → A	•	•	•	•	•	•	•	•

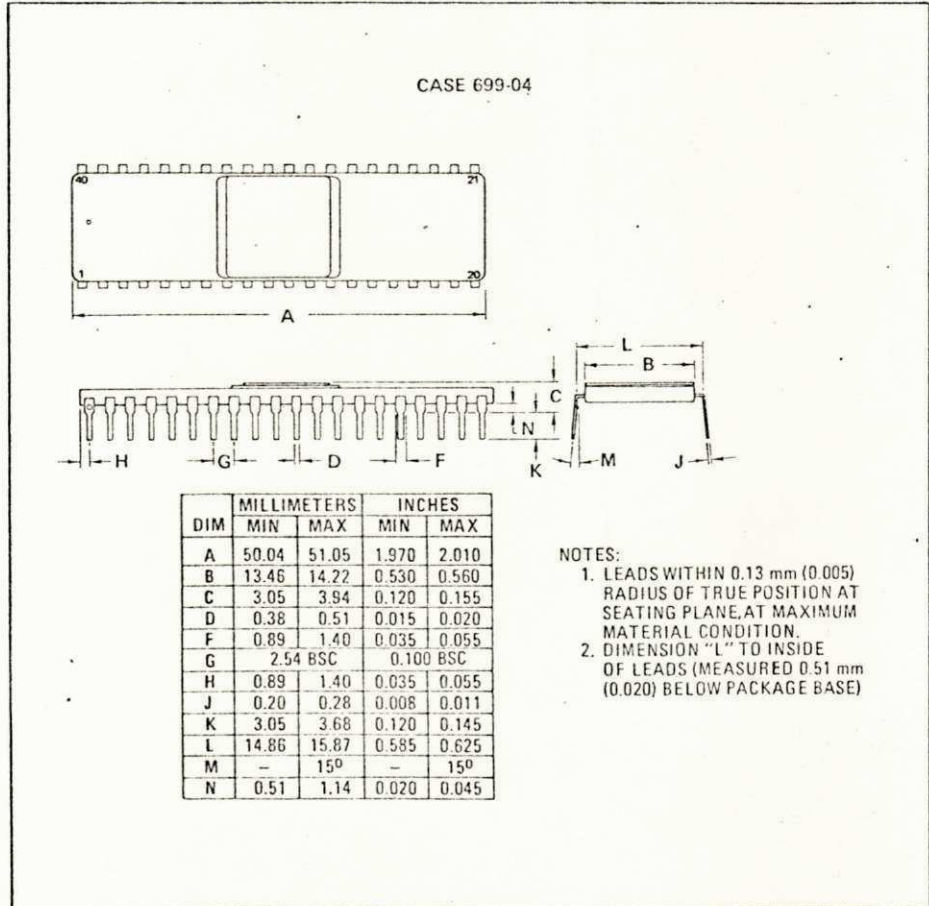
CONDITION CODE REGISTER NOTES:

- (Bit set if test is true and cleared otherwise)
- 1 (Bit V) Test: Result = 10000000?
  - 2 (Bit C) Test: Result = 00000000?
  - 3 (Bit C) Test: Decimal value of most significant BCD Character greater than nine? (Not cleared if previously set.)
  - 4 (Bit V) Test: Operand = 10000000 prior to execution?
  - 5 (Bit V) Test: Operand = 01111111 prior to execution?
  - 6 (Bit V) Test: Set equal to result of N⊙C after shift has occurred.
  - 7 (Bit N) Test: Sign bit of most significant (MS) byte = 1?
  - 8 (Bit V) Test: 2's complement overflow from subtraction of MS bytes?
  - 9 (Bit N) Test: Result less than zero? (Bit 15 = 1)
  - 10 (All) Load Condition Code Register from Stack. (See Special Operations)
  - 11 (Bit I) Set when interrupt occurs. If previously set, a Non-Maskable Interrupt is required to exit the wait state.
  - 12 (All) Set according to the contents of Accumulator A.

PIN ASSIGNMENT



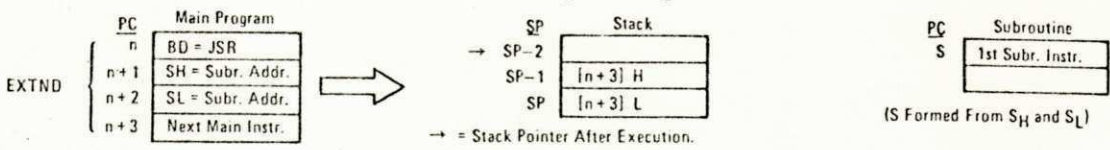
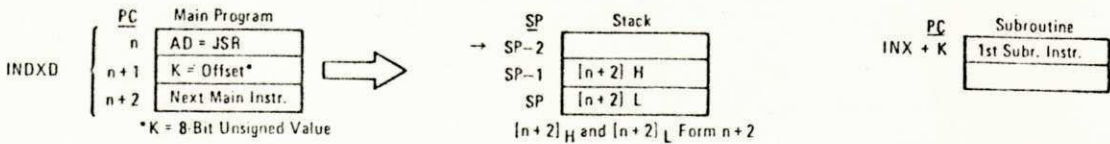
PACKAGE DIMENSIONS



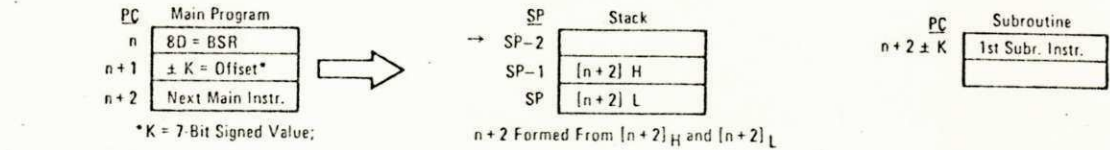


**SPECIAL OPERATIONS**

**JSR, JUMP TO SUBROUTINE:**



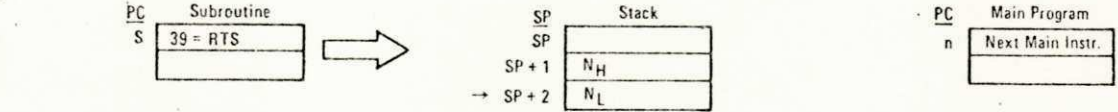
**BSR, BRANCH TO SUBROUTINE:**



**JMP, JUMP:**



**RTS, RETURN FROM SUBROUTINE:**



**RTI, RETURN FROM INTERRUPT:**

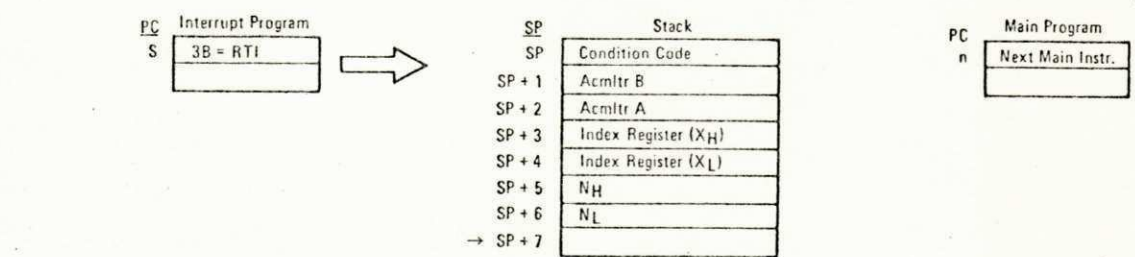


TABLE 4 - INDEX REGISTER AND STACK MANIPULATION INSTRUCTIONS

POINTER OPERATIONS	MNEMONIC	BOOLEAN/ARITHMETIC OPERATION															COND. CODE REG.							
		IMMED			DIRECT			INDEX			EXTND			IMPLIED			BOOLEAN/ARITHMETIC OPERATION	H	I	N	Z	V	C	
		OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#								
Compare Index Reg	CPX	8C	3	3	9C	4	2	AC	6	2	BC	5	3				$X_H - M, X_L - (M + 1)$	•	•	⑦	1	⑦	•	•
Decrement Index Reg	DEX													09	4	1	$X - 1 \rightarrow X$	•	•	•	•	•	•	•
Decrement Stack Ptr	DES													34	4	1	$SP - 1 \rightarrow SP$	•	•	•	•	•	•	•
Increment Index Reg	INX													08	4	1	$X + 1 \rightarrow X$	•	•	•	•	•	•	•
Increment Stack Ptr	INS													31	4	1	$SP + 1 \rightarrow SP$	•	•	•	•	•	•	•
Load Index Reg	LDX	CE	3	3	DE	4	2	EE	6	2	FE	5	3				$M \rightarrow X_H, (M + 1) \rightarrow X_L$	•	•	⑨	1	R	•	•
Load Stack Ptr	LDS	8E	3	3	9E	4	2	AE	6	2	BE	5	3				$M \rightarrow SP_H, (M + 1) \rightarrow SP_L$	•	•	⑨	1	R	•	•
Store Index Reg	STX				DF	5	2	EF	7	2	FF	6	3				$X_H \rightarrow M, X_L \rightarrow (M + 1)$	•	•	•	•	•	•	•
Store Stack Ptr	STS				9F	5	2	AF	7	2	BF	6	3				$SP_H \rightarrow M, SP_L \rightarrow (M + 1)$	•	•	⑨	1	R	•	•
Indx Reg → Stack Ptr	TXS													35	4	1	$X - 1 \rightarrow SP$	•	•	•	•	•	•	•
Stack Ptr → Indx Reg	TSX													30	4	1	$SP + 1 \rightarrow X$	•	•	•	•	•	•	•

TABLE 5 - JUMP AND BRANCH INSTRUCTIONS

OPERATIONS	MNEMONIC	COND. CODE REG.																				
		RELATIVE			INDEX			EXTND			IMPLIED			BRANCH TEST	H	I	N	Z	V	C		
		OP	~	#	OP	~	#	OP	~	#	OP	~	#									
Branch Always	BRA	20	4	2										None	•	•	•	•	•	•	•	•
Branch If Carry Clear	BCC	24	4	2										$C = 0$	•	•	•	•	•	•	•	•
Branch If Carry Set	BCS	25	4	2										$C = 1$	•	•	•	•	•	•	•	•
Branch If = Zero	BEQ	27	4	2										$Z = 1$	•	•	•	•	•	•	•	•
Branch If ≥ Zero	BGE	2C	4	2										$N \oplus V = 0$	•	•	•	•	•	•	•	•
Branch If > Zero	BGT	2E	4	2										$Z + (N \oplus V) = 0$	•	•	•	•	•	•	•	•
Branch If Higher	BHI	22	4	2										$C + Z = 0$	•	•	•	•	•	•	•	•
Branch If ≤ Zero	BLE	2F	4	2										$Z + (N \oplus V) = 1$	•	•	•	•	•	•	•	•
Branch If Lower Or Same	BLS	23	4	2										$C + Z = 1$	•	•	•	•	•	•	•	•
Branch If < Zero	BLT	2D	4	2										$N \oplus V = 1$	•	•	•	•	•	•	•	•
Branch If Minus	BMI	28	4	2										$N = 1$	•	•	•	•	•	•	•	•
Branch If Not Equal Zero	BNE	26	4	2										$Z = 0$	•	•	•	•	•	•	•	•
Branch If Overflow Clear	BVC	28	4	2										$V = 0$	•	•	•	•	•	•	•	•
Branch If Overflow Set	BVS	29	4	2										$V = 1$	•	•	•	•	•	•	•	•
Branch If Plus	BPL	2A	4	2										$N = 0$	•	•	•	•	•	•	•	•
Branch To Subroutine	BSR	8D	8	2											•	•	•	•	•	•	•	•
Jump	JMP				6E	4	2	7E	3	3				} See Special Operations	•	•	•	•	•	•	•	•
Jump To Subroutine	JSR				AD	8	2	8D	9	3						•	•	•	•	•	•	•
No Operation	NOP													Advances Prog. Cntr. Only	•	•	•	•	•	•	•	•
Return From Interrupt	RTI														•	•	•	•	•	•	•	•
Return From Subroutine	RTS														•	•	•	•	•	•	•	•
Software Interrupt	SWI													} See Special Operations	•	•	•	•	•	•	•	•
Wait for Interrupt	WAI															•	•	•	•	•	•	•





TABLE 3 - ACCUMULATOR AND MEMORY INSTRUCTIONS

OPERATIONS	MNEUMONIC	ADDRESSING MODES					BOOLEAN/ARITHMETIC OPERATION (All register labels refer to contents)	COND. CODE REG.										
		IMMED		DIRECT		INDEX		EXTND		IMPLIED		5	4	3	2	1	0	
		OP	~ #	OP	~ #	OP		~ #	OP	~ #	OP	~ #	H	I	N	Z	V	C
Add	ADDA	8B	2 2	9B	3 2	AB	5 2	BB	4 3			A + M → A	1	1	1	1	1	
	ADDB	CB	2 2	DB	3 2	EB	5 2	FB	4 3			B + M → B	1	1	1	1	1	
Add Acmltrs	ABA									1B	2 1	A + B → A	1	1	1	1	1	
Add with Carry	ADCA	89	2 2	99	3 2	A9	5 2	B9	4 3			A + M + C → A	1	1	1	1	1	
	ADCB	C9	2 2	D9	3 2	E9	5 2	F9	4 3			B + M + C → B	1	1	1	1	1	
And	ANDA	84	2 2	94	3 2	A4	5 2	B4	4 3			A · M → A	1	1	1	1	1	
	ANDB	C4	2 2	D4	3 2	E4	5 2	F4	4 3			B · M → B	1	1	1	1	1	
Bit Test	BITA	85	2 2	95	3 2	A5	5 2	B5	4 3			A · M	1	1	1	1	1	
	BITB	C5	2 2	D5	3 2	E5	5 2	F5	4 3			B · M	1	1	1	1	1	
Clear	CLR					6F	7 2	7F	6 3			00 → M	1	1	1	1	1	
	CLRA									4F	2 1	00 → A	1	1	1	1	1	
	CLRB									5F	2 1	00 → B	1	1	1	1	1	
Compare	CPA	81	2 2	91	3 2	A1	5 2	B1	4 3			A - M	1	1	1	1	1	
	CPB	C1	2 2	D1	3 2	E1	5 2	F1	4 3			B - M	1	1	1	1	1	
Compare Acmltrs	CBA									11	2 1	A - B	1	1	1	1	1	
Complement, 1's	COM					63	7 2	73	6 3			M̄ → M	1	1	1	1	1	
	COMA									43	2 1	Ā → A	1	1	1	1	1	
	COMB									53	2 1	B̄ → B	1	1	1	1	1	
Complement, 2's (Negate)	NEG					60	7 2	70	6 3			00 - M → M	1	1	1	1	1	
	NEGA									40	2 1	00 - A → A	1	1	1	1	1	
	NEGB									50	2 1	00 - B → B	1	1	1	1	1	
Decimal Adjust, A	DAA									19	2 1	Converts Binary Add. of BCD Characters into BCD Format	1	1	1	1	1	
Decrement	DEC					6A	7 2	7A	6 3			M - 1 → M	1	1	1	1	1	
	DECA									4A	2 1	A - 1 → A	1	1	1	1	1	
	DECB									5A	2 1	B - 1 → B	1	1	1	1	1	
Exclusive OR	EORA	88	2 2	98	3 2	A8	5 2	B8	4 3			A ⊕ M → A	1	1	1	1	1	
	EORB	C8	2 2	D8	3 2	E8	5 2	F8	4 3			B ⊕ M → B	1	1	1	1	1	
Increment	INC					6C	7 2	7C	6 3			M + 1 → M	1	1	1	1	1	
	INCA									4C	2 1	A + 1 → A	1	1	1	1	1	
	INCB									5C	2 1	B + 1 → B	1	1	1	1	1	
Load Acmltr	LDA	86	2 2	96	3 2	A6	5 2	B6	4 3			M → A	1	1	1	1	1	
	LDAB	C6	2 2	D6	3 2	E6	5 2	F6	4 3			M → B	1	1	1	1	1	
Or, Inclusive	ORA	8A	2 2	9A	3 2	AA	5 2	BA	4 3			A + M → A	1	1	1	1	1	
	ORB	CA	2 2	DA	3 2	EA	5 2	FA	4 3			B + M → B	1	1	1	1	1	
Push Data	PSHA									36	4 1	A → M <sub>SP</sub> , SP - 1 → SP	1	1	1	1	1	
	PSHB									37	4 1	B → M <sub>SP</sub> , SP - 1 → SP	1	1	1	1	1	
Pull Data	PULA									32	4 1	SP + 1 → SP, M <sub>SP</sub> → A	1	1	1	1	1	
	PULB									33	4 1	SP + 1 → SP, M <sub>SP</sub> → B	1	1	1	1	1	
Rotate Left	ROL					69	7 2	79	6 3			M	1	1	1	1	1	
	ROLA									49	2 1	A	1	1	1	1	1	
	ROLB									59	2 1	B	1	1	1	1	1	
Rotate Right	ROR					66	7 2	76	6 3			M	1	1	1	1	1	
	RORA									46	2 1	A	1	1	1	1	1	
	RORB									56	2 1	B	1	1	1	1	1	
Shift Left, Arithmetic	ASL					68	7 2	78	6 3			M	1	1	1	1	1	
	ASLA									48	2 1	A	1	1	1	1	1	
	ASLB									58	2 1	B	1	1	1	1	1	
Shift Right, Arithmetic	ASR					67	7 2	77	6 3			M	1	1	1	1	1	
	ASRA									47	2 1	A	1	1	1	1	1	
	ASRB									57	2 1	B	1	1	1	1	1	
Shift Right, Logic	LSR					64	7 2	74	6 3			M	1	1	1	1	1	
	LSRA									44	2 1	A	1	1	1	1	1	
	LSRB									54	2 1	B	1	1	1	1	1	
Store Acmltr.	STAA			97	4 2	A7	6 2	B7	5 3			A → M	1	1	1	1	1	
	STAB			D7	4 2	E7	6 2	F7	5 3			B → M	1	1	1	1	1	
Subtract	SUBA	80	2 2	90	3 2	A0	5 2	B0	4 3			A - M → A	1	1	1	1	1	
	SUBB	C0	2 2	D0	3 2	E0	5 2	F0	4 3			B - M → B	1	1	1	1	1	
Subtract Acmltrs.	SBA									10	2 1	A - B → A	1	1	1	1	1	
Subtr. with Carry	SBCA	82	2 2	92	3 2	A2	5 2	B2	4 3			A - M - C → A	1	1	1	1	1	
	SBCB	C2	2 2	D2	3 2	E2	5 2	F2	4 3			B - M - C → B	1	1	1	1	1	
Transfer Acmltrs	TAB									16	2 1	A → B	1	1	1	1	1	
	TBA									17	2 1	B → A	1	1	1	1	1	
Test, Zero or Minus	TST					6D	7 2	7D	6 3			M = 00	1	1	1	1	1	
	TSTA									4D	2 1	A = 00	1	1	1	1	1	
	TSTB									5D	2 1	B = 00	1	1	1	1	1	

LEGEND:

- OP Operation Code (Hexadecimal);
- ~ Number of MPU Cycles;
- # Number of Program Bytes;
- + Arithmetic Plus;
- Arithmetic Minus;
- Boolean AND;
- M<sub>SP</sub> Contents of memory location pointed to be Stack Pointer;

- + Boolean Inclusive OR;
- ⊕ Boolean Exclusive OR;
- M̄ Complement of M;
- Transfer Into;
- 0 Bit = Zero;
- 00 Byte = Zero;

CONDITION CODE SYMBOLS:

- H Half-carry from bit 3;
- I Interrupt mask
- N Negative (sign bit)
- Z Zero (byte)
- V Overflow, 2's complement
- C Carry from bit 7
- R Reset Always
- S Set Always
- 1 Test and set if true, cleared otherwise
- Not Affected

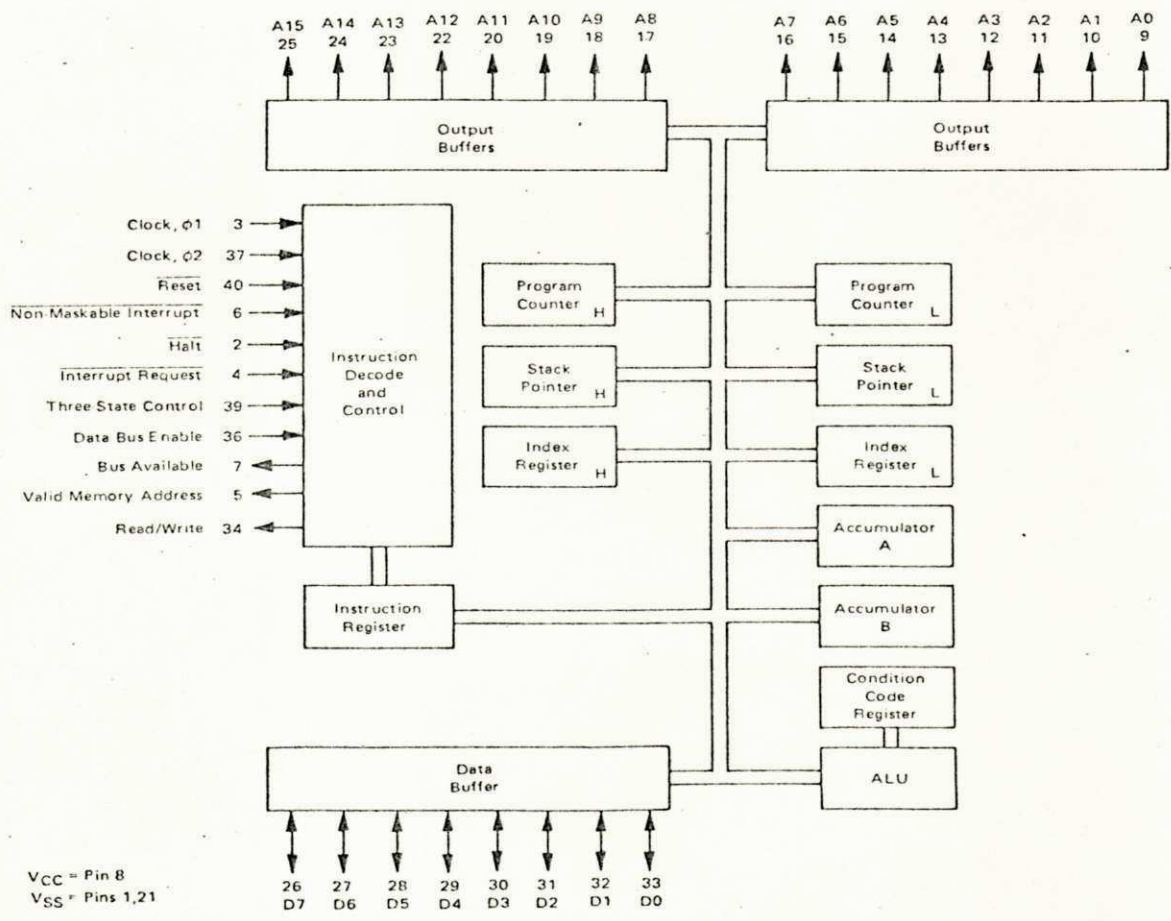
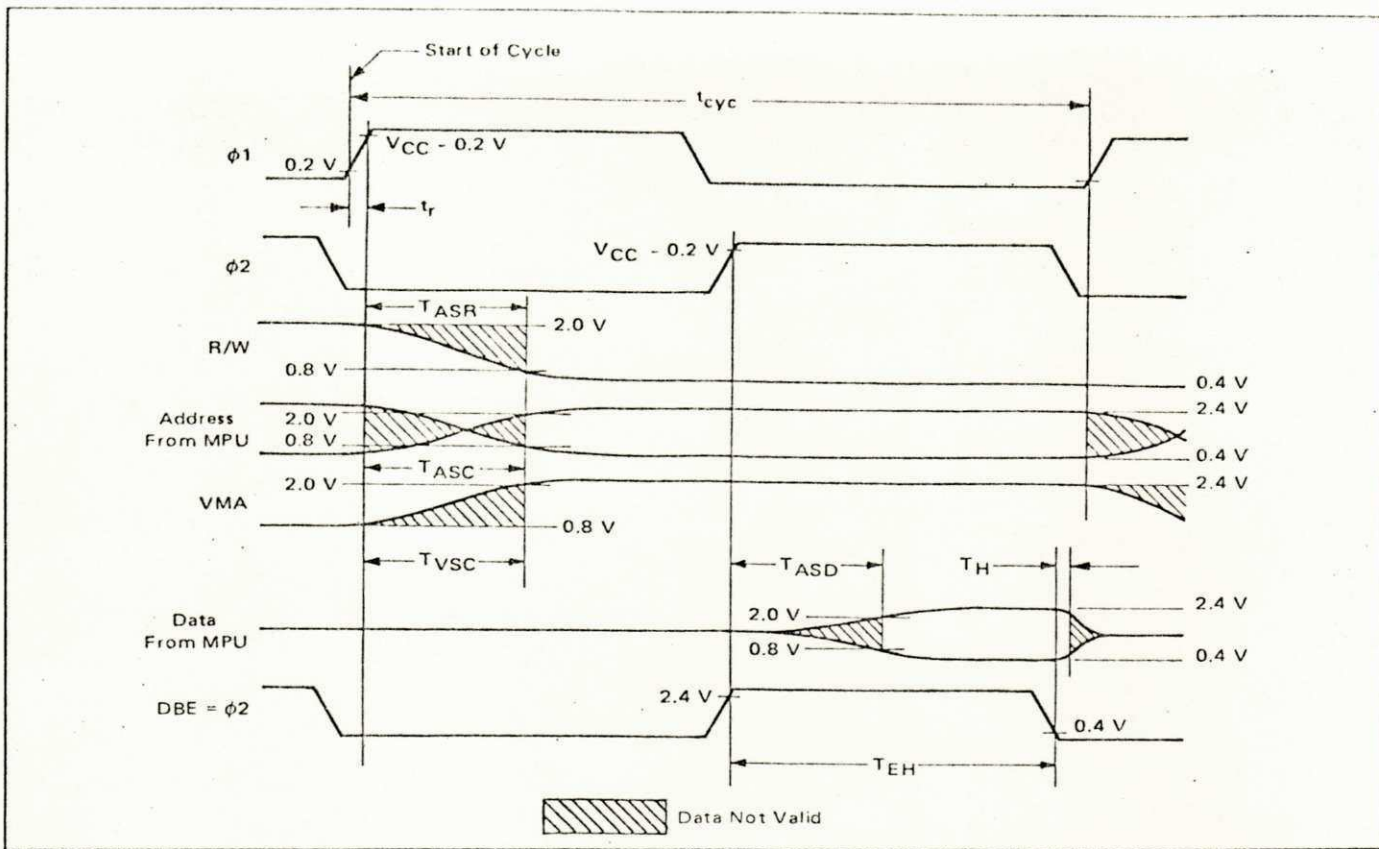
Note - Accumulator addressing mode instructions are included in the column for IMPLIED addressing



MOTOROLA Semiconductor Products Inc.



FIGURE 3 - WRITE DATA IN MEMORY OR PERIPHERALS

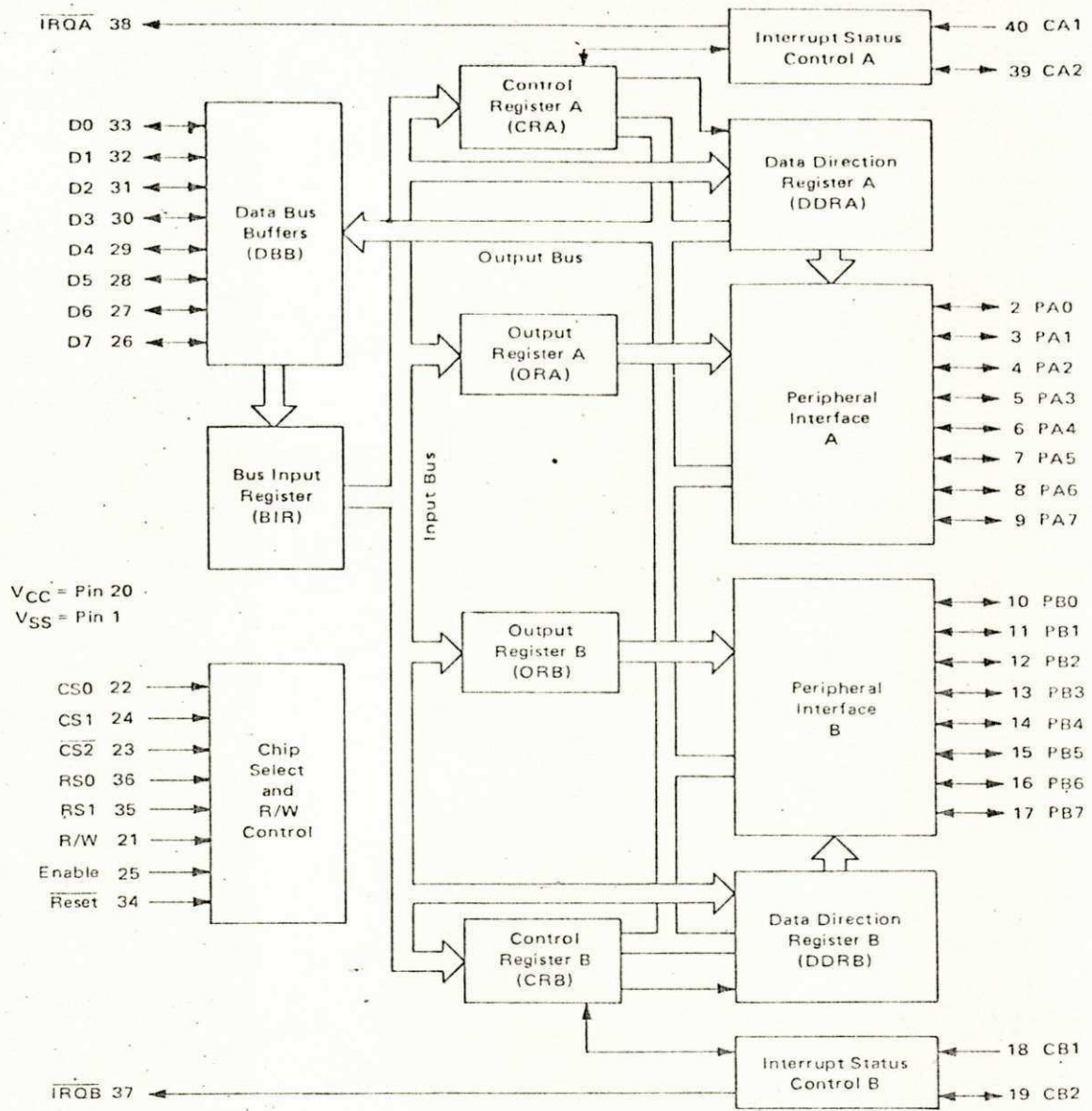


A P Ê N D I C E   I I I

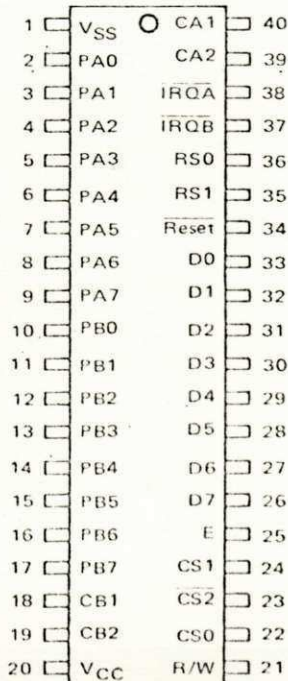
DESCRIÇÃO DOS PINOS E ARQUITETURA INTERNA DA PIA MC6820

DESCRIÇÃO DOS PINOS E ARQUITETURA INTERNA DA ACIA MC6850

EXPANDED BLOCK DIAGRAM

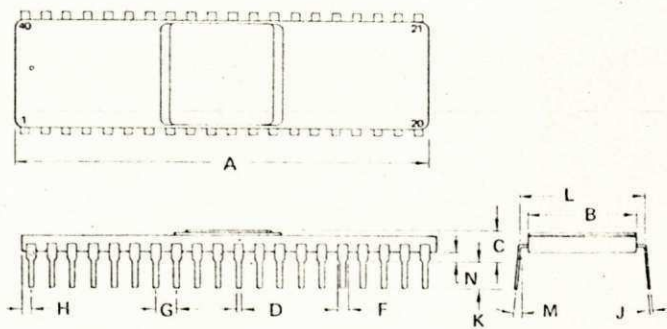


PIN ASSIGNMENT



PACKAGE DIMENSIONS

CASE 699-04



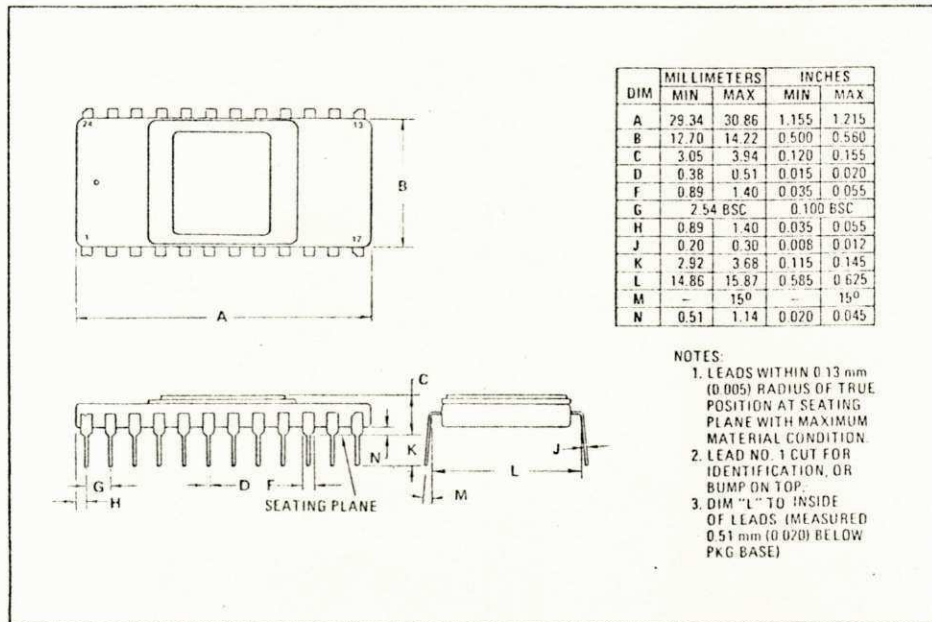
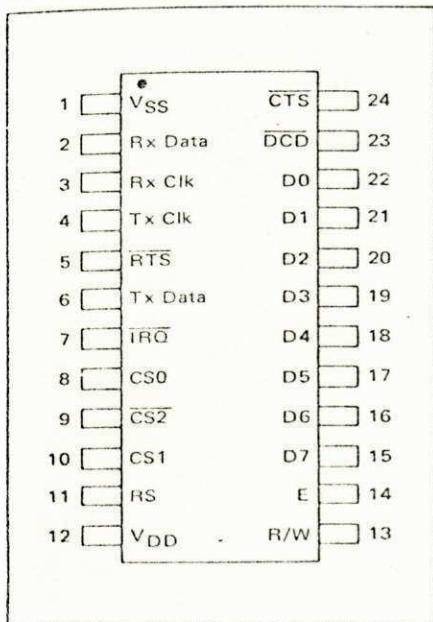
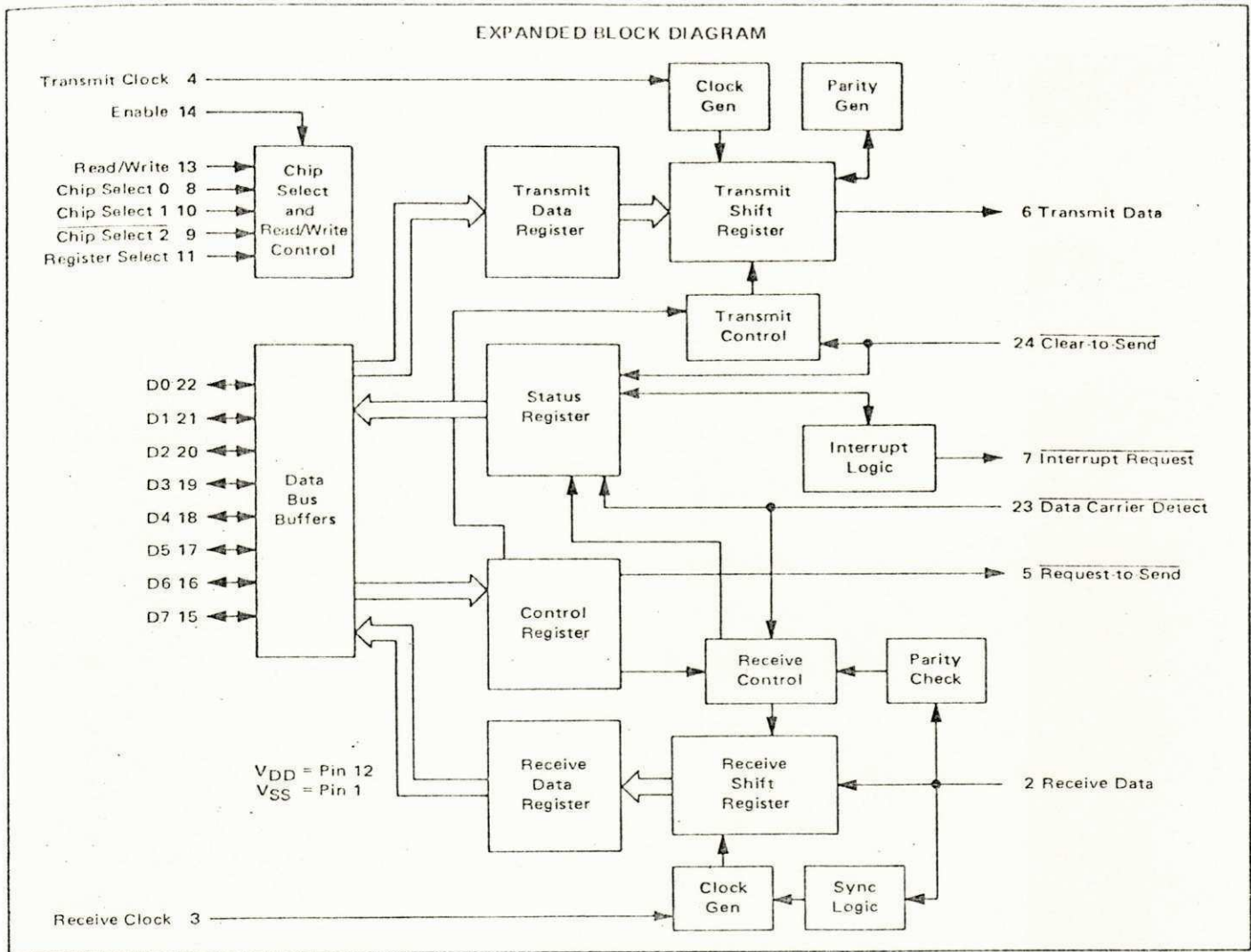
DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	50.04	51.05	1.970	2.010
B	13.46	14.22	0.530	0.560
C	3.05	3.94	0.120	0.155
D	0.38	0.51	0.015	0.020
F	0.89	1.40	0.035	0.055
G	2.54 BSC		0.100 BSC	
H	0.89	1.40	0.035	0.055
J	0.20	0.28	0.008	0.011
K	3.05	3.68	0.120	0.145
L	14.86	15.87	0.585	0.625
M	15°		15°	
N	0.51	1.14	0.020	0.045

NOTES:

- LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
- DIMENSION "L" TO INSIDE OF LEADS (MEASURED 0.51 mm (0.020) BELOW PACKAGE BASE)



EXPANDED BLOCK DIAGRAM



**A P Ê N D I C E   I V**

**DISTRIBUIÇÃO DOS PINOS DA VIA COMUM DO SISTEMA**

## VIA COMUM

Os sinais foram distribuidos da maneira vista abaixo:

SINAL	PINOS		SINAL
	A	B	
D0	1	23	+ 5 Volts
D2	2	24	+ 5 Volts
D4	3	25	+ 5 Volts
D6	4	26	D1
+ 12 Volts	5	27	D3
- 5 Volts	6	28	D5
RESET	7	29	D7
02	8	30	A15.VMA
A - SEL MUX	9	31	$\overline{\text{NMI}}$
B - SEL MUX	10	32	R/W
C - SEL MUX	11	33	$\overline{\text{TRQ}}$
A14	12	34	A15
A12	13	35	A13
A10	14	36	A11
A8	15	37	A9
A6	16	38	A7
A4	17	39	A5
A2	18	40	A3
A0	19	41	A1
GND	20	42	RESET MANUAL
GND	21	43	RESET MANUAL
GND	22	44	VMA



A P Ê N D I C E V

FORMATO DOS RELATÓRIOS

SADNE - RELATÓRIO DE PESQUISA

SADNE - STATUS

INT MEDIÇÕES T = 10:00

No MEDIÇÕES N = 0010

DENSIDADE AR = 01,12

HORA: 10:30:30

DATA: 10/01/81

MODO FUNCIONAMENTO: CPO

ATIVIDADE TORRES E BASES

VATTB: 00000001

VATT1: 00001111

VATT2: 11111111

VATT3: 11111111

VATT4: 11111111

VATT5: 11111111

VATB1: 11111111

VATB2: 11111111

VATB3: 11111111

HORÁRIO INIC TESTES: 08:50:00

HORÁRIO TERM TESTES: 10:30:00

DATA INIC TESTES: 10/01/81

DATA TERM TESTES: 10/01/81

SADNE - PROCÊSSAMENTO

TORRE 1 - VEL MED	DESV PAD	MOM 3a ORD	POT MED
AN1 - 3,47	0,31	1,82	25,82
AN2 - 3,80	0,22	1,52	34,21
AN3 - 3,95	0,18	1,48	30,30
AN4 - 3,50	0,25	1,39	32,40

FIM

## B I B L I O G R A F I A

- 01 - Araújo, T. S. de - Programa de Energia Eólica, Doc. elaborado para o Grupo de Assessoria de Energia do CNPq - jul. 1980.
- 02 - M6800 Microprocessor Applications Manual - MOTOROLA Semiconductor Products Inc., 1975.
- 03 - M6800 Systems Reference and Data Sheets - MOTOROLA Semiconductor Products Inc., 1975.
- 04 - M6800 Microprocessor Programming Manual - MOTOROLA Semiconductor Products Inc., 1975.
- 05 - Component Data Catalog - Intel Corporation, 1978.
- 06 - Logic - TTL Data Manual - Signetic Corporation, 1978.
- 07 - Linear Data Book - National Semiconductor Corporation, 1978.
- 08 - The TTL Data Book - Texas Instruments Inc., 2nd. Edition, 1976.
- 09 - The MOS Memory Data Book - Texas Instruments Inc., 1978.
- 10 - Mao, T. W. and Yeh, R.T. - Communication Port: A Language Concept for Concurrent Programming, IEEE Trans. on Software Engineering, Vol. 6, No 2, March 1980.
- 11 - Microprocessors - Electronics Book Series - McGraw Hill, 1975.
- 12 - Applying Microprocessors - Electronics Book Series - McGraw Hill, 1976.
- 13 - Hilburn, J. L. and Julich, P.N. - Microcomputers/Microprocessors Hardware, Software and Applications, Prentice Hall, 1976.
- 14 - Soucek, B. - Microprocessors and Microcomputers, Wiley, 1976.
- 15 - Amundson, R.M; Osecry, B.D. and Bennett, A. W. - A Hierarchical Network for Data Acquisition and Control, IEEE Trans. on IECI, Vol. 23, No. 1, Feb. 1976.
- 16 - Fullagar, D.; Bradshaw, P. and Evans, L. - Interfacing Data Converters and Microprocessors, Electronics, Dec. 9, 1976.



- 17 - Data acquisition Handbook - National Semiconductor Corporation, 1978.
- 18 - Special Issue on Automatic Testing Equipment - IEEE Trans. on Instrumentation and Measurement, Vol. 27, No. 2, June 1978.
- 19 - Morrisson, R. L. - Microcomputers Invade the Linear World, IEEE Spectrum, Vol. 15, No. 7, July 1978.
- 20 - Deep, G.S; Cavalcanti, J.H.F. e Araújo, A. de A. - Controle de Velocidade de um Motor dc por Microprocessador, Anais do 5º Sem. Integrado de Software e Hardware Nacionais, Julho 1978.
- 21 - Microprocessor User's Guide - PRO-LOG Corporation, 1979.
- 22 - Special Issue - An Applications Review: Technology 79 - IEEE Spectrum, Vol. 16, No. 1, Jan. 1979.
- 23 - Aylor, J. H; Parrish, E.A and McDonald, W.E. - A Modular Microcomputer Development System, IEEE Trans. on IECI, Vol. 26, No. 1, Feb. 1979.
- 24 - 1979 Electrical and Electronic Measurement and Test Instrument Conference - IEEE Trans. on Instrumentation and Measurement, Vol. 28, No. 4, Dec. 1979.
- 25 - Special Issue - An Applications Review: Technology 80 - IEEE Spectrum, Vol. 17, No. 1, Jan. 1980.
- 26 - Joint Special Issue on Microprocessors and Microcomputers - IEEE Trans. on Computers, Vol. 29, No. 2, Feb. 1980.
- 27 - Tyuluman, S.A; Schroder, D. C. and Austin, M.E. - A Microprocessor-Based System for the Monitoring and Control of a Solar Installation; IEEE Trans. on IECI, Vol. 27, No.1 Feb. 1980.
- 28 - Hayes, J.P. and McCluskey, E.J. - Testability Considerations in Microprocessor-Based Design, IEEE Computer Magazine, Vol. 13, No. 3, March 1980.
- 29 - Maia, J.B. de M. - Sistema de Aquisição de Dados para uma Central de Despacho (sistema CHESF), Tese de Mestrado, DEE/UFPb, Outubro de 1980.