


UM SISTEMA DE POSICIONAMENTO ELETRÔNICO PARA DRAGAGEM,

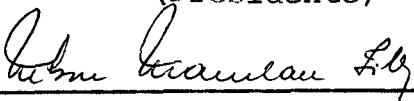
BASEADO EM MICROPROCESSADOR

Pedro Pereira Brando

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE  
PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE  
JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO  
GRAU DE MESTRE EM CIÊNCIAS (M.Sc.).

Aprovada por:

  
\_\_\_\_\_  
Prof. Guilherme Chagas Rodrigues  
(Presidente)

  
\_\_\_\_\_  
Prof. Nelson Maculan Filho

  
\_\_\_\_\_  
Prof. Ivan da Costa Marques

RIO DE JANEIRO, RJ - BRASIL

DEZEMBRO DE 1978

BRANDO, PEDRO PEREIRA

Um Sistema de Posicionamento Eletrônico para dragagem, baseado em microprocessador. [Rio de Janeiro] 1978

VIII, 130 pág. 29,7cm (COPPE-UFRJ, M.Sc., Engenharia de Sistemas e Computação, 1972)

Tese - Universidade Federal do Rio de Janeiro

1. Computadores I. COPPE/UFRJ

Pedro Pereira Brando nasceu no Rio de Janeiro, em 12 de novembro de 1943. Formou-se em Engenharia Civil na Escola de Engenharia da Universidade Federal do Rio de Janeiro, em 1970.

Chefe do Setor de Desenvolvimento da Divisão de Processamento de Dados da Companhia Brasileira de Dragagem, empresa na qual trabalha desde 1974.

Lecionou disciplina Computação I no Instituto de Matemática da UFRJ e na Escola de Química da UFRJ.

**DEDICATÓRIA**

A esposa e filho(s), que suportaram, sem hesitar, o sacrifício que o dia-a-dia deste trabalho exigiu, a eles dedico a alegria dos primeiros resultados.

**AGRADECIMENTOS**

A todos, amigos e colaboradores, que tornaram possível a realização deste trabalho, e, em especial, a

GUILHERME CHAGAS RODRIGUES, professor orientador desta tese;

MARCO ANTONIO DE PAIVA, Diretor Técnico da C.B.D.;

GERALDO CLAYTON TEIXEIRA, chefe da Divisão de Processamento de Dados da C.B.D.;

LEOPOLDO SPÍNOLA BITTENCOURT, engenheiro de planejamento da Divisão de Dragagem da C.B.D.;

ANTONIO GERALDO NEVES DA CUNHA, engenheiro eletrônico da Divisão de Processamento de Dados da C.B.D.;

ARNO SPECHT, engenheiro supervisor da Draga MACAPÁ (C.B.D.).;

TARCÍSIO NEVES DA CUNHA, chefe do Setor Técnico da Divisão de Processamento de Dados da C.B.D.;

RAIMUNDO AMORA RAMOS, companheiro de estudos, do programa de Engenharia de Sistemas e Computação - COPPE.

**SUMÁRIO**

O Sistema de Posicionamento Eletrônico - SPE - destina-se a posicionar dragas tipo autotransportadoras.

Este sistema de Tempo-Real resultou de um estudo realizado para determinar as causas da inoperância dos sistemas de posicionamento - MDS - importados pela Companhia Brasileira de Dragagem.

Constituiu objetivo mais importante deste projeto, dar ao SPE robustez e operacionalidade, qualidades que faltaram ao seu antecessor.

O SPE apresenta, como novidade, um controlador implementado com microprocessador, acrescido de um software especialmente desenvolvido para operá-lo.

Um enfoque especial foi dado ao cálculo do posicionamento, tendo em vista a limitada capacidade de cálculo numérico do microprocessador adotado.

**SUMMARY**

The Electronic Positioning System - SPE - is intended for position-fixing of hopper dredges.

This Real-Time System results of studies on the causes of the failures observed in the positioning system - MDS - imported by the Companhia Brasileira de Dragagem.

The SPE presents, as a special feature, a controller based on a microprocessor, with an orientend software specially designed to operate it.

The most important goal of the project is to provide the SPE with robustness and handling facilities, qualities that were not present in the previous system.

The positioning calculation has been specially considered, because of the limited capacity for numerical calculation of the microprocessor, adopted.

## Í N D I C E

**CAPÍTULO I: INTRODUÇÃO**

	Pág.
1.1 - NATUREZA DO TEXTO .....	01
1.2 - MOTIVAÇÃO E VIABILIDADE PARA O EMPREENDIMENTO .....	02
1.3 - PROJETO DO SISTEMA DE POSICIONAMENTO ELETRÔNICO (PSPE) ...	08
1.4 - OBJETIVOS E LIMITAÇÕES DO PROJETO PSPE .....	09

**CAPÍTULO II: DEFINIÇÃO DO SISTEMA SPE**

2.1 - PROPÓSITO E USO DO SISTEMA SPE .....	13
2.2 - FILOSOFIA DO SISTEMA SPE .....	14
2.3 - DESCRIÇÃO DO HARDWARE DO SISTEMA SPE .....	18
2.4 - FUNCIONAMENTO DO SISTEMA SPE .....	29
2.5 - TRIANGULAÇÃO COM O RADAR MRS MOTOROLA .....	34
2.6 - POSICIONAMENTO COM O SISTEMA SPE .....	39
2.7 - OPERAÇÃO DO SISTEMA SPE .....	42

**CAPÍTULO III: DETALHAMENTO DO SOFTWARE**

3.1 - ESTRUTURA DO SOFTWARE .....	48
3.2 - PACOTES DE SUBROTINAS .....	49
3.3 - PROGRAMA PRINCIPAL .....	73

**CAPÍTULO IV: IMPLEMENTAÇÃO DO SOFTWARE**

	Pág.
4.1 - METODOLOGIA DO POSICIONAMENTO .....	103
4.2 - ESCOLHA DE UMA ARITMÉTICA ADEQUADA .....	106
4.3 - ARITMÉTICA SPE .....	111
4.4 - ALGORITMO EXPEDITO DE PLOTAGEM .....	115
4.5 - RECURSOS UTILIZADOS NO DESENVOLVIMENTO DO PROJETO .....	117

**CAPÍTULO V: ANÁLISE DOS RESULTADOS EXPERIMENTAIS**

5.1 - FASE EXPERIMENTAL .....	119
5.2 - FASE DE OPERAÇÃO .....	122

**CAPÍTULO VI: CONCLUSÕES**

6.1 - APERFEIÇOAMENTOS .....	128
6.2 - EXTENSÕES .....	129
6.3 - INDUSTRIALIZAÇÃO .....	130



## CAPÍTULO I

### INTRODUÇÃO

#### 1.1 - NATUREZA DO TEXTO

O presente trabalho constitui, em síntese, o histórico de uma experiência. Experiência vivida pela Universidade na busca de uma solução (SPE) para um problema tecnológico (posicionamento eletrônico de dragas autotransportadoras), encaminhado a ela pela Companhia Brasileira de Dragagem (CBD).

Dentre as diversas fases dessa experiência, abordamos: no capítulo I, levantamento e estudo do problema; nos capítulos II, III e IV, projeto e execução; e, no capítulo V, implantação da solução.

Para a exposição do assunto, utilizamos tanto o enfoque acadêmico, quanto o profissional, uma vez que o fio condutor dessa experiência foi a integração Universidade-Empresa.

## 1.2 - MOTIVAÇÃO E VIABILIDADE PARA O EMPREENDIMENTO

O advento dos minicomputadores, propiciado pelas novas conquistas da tecnologia eletrônica digital, ao início dos anos 70, viabilizou técnica e economicamente uma série de aplicações independentes (STAND-ALONE) de processamento de dados, aplicações que, até então, tornavam-se proibitivas de serem realizadas com os computadores tradicionais, tendo em vista o elevado custo desse investimento e os cuidados exigidos para instalação, operação e manutenção desses "complexos sistemas eletrônicos". O posicionamento eletrônico de embarcações em TEMPO-REAL veio a se constituir em uma das aplicações viabilizadas por tal tecnologia. Dentro dessa filosofia, novos sistemas de posicionamento eletrônico foram, então, lançados no mercado internacional, para aplicações em navegação e para serviço de posicionamento propriamente dito, com características de hardware bem semelhante entre si, tornando obsoletos os primeiros sistemas de posicionamento existentes, de tecnologia ultrapassada, que, utilizando nos seus controladores hardwares não inteligentes, eram menos versáteis.

O posicionamento eletrônico de embarcações em tempo-real constitui-se, em síntese, no conhecimento da posição de um móvel flutuante sobre a lâmina d'água a cada instante, através de dispositivo eletrônico, obtendo-se com isso a caracterização de sua trajetória de navegação ou tendência de deslocamento, conforme o tipo de aplicação que se tenha: navegação ou posicionamento, respectivamente. Quando empregado no auxílio à navegação, o posicionamento eletrônico se satisfaz com precisões bem modestas na determinação de uma posição (da ordem de 500 metros), ao passo que, para fins de posicionamento propriamente dito, a precisão deve ser bem mais apurada, tanto assim que, dependendo do sistema, ela pode atingir até 3 metros, o que, considerando as condições de trabalho e

as dimensões do objeto posicionado, significa grande exatidão. Nesse último caso (operações de posicionamento) tais sistemas permitem executar tarefas como: levantamentos batimétricos; locação de plataformas petrolíferas; posicionamento de dragas; ou quaisquer outros serviços marítimos ou fluviais em que se deseje conhecer a posição de um móvel flutuante, com precisão e dinâmica, inatingíveis por métodos hidrográficos convencionais. O posicionamento de dragas é a aplicação objeto do nosso estudo.

Na execução de obras de dragagem em que se opera com dragas tipo auto-transportadoras, um sistema de posicionamento eletrônico de tempo-real é peça indispensável como recurso complementar desse equipamento. De fato, tal tipo de draga difere das demais por ser dotada de propulsão própria, operando sempre em movimento sobre a lâmina d'água, quando está dragando. Portanto é necessário que seu condutor, o draguista, conheça precisamente a posição em que o equipamento está a cada instante, uma vez que, nas condições normais de serviço, essas dragas, quase sempre, estão às voltas com a possibilidade de encalhar ou mesmo danificar-se em algum alto-fundo. Por outro lado, o posicionamento oferece também ao draguista meios para conduzir a draga durante a operação, exatamente sobre a locação da obra, evitando dragagens por excesso (sobredragagem), normalmente não remuneradas pelo contratante e, por vezes, até prejudiciais tecnicamente à obra.

Por volta de 1973, a crescente demanda de serviços, imposta pela política governamental dos CORREDORES DE EXPORTAÇÃO, passou a exigir da Companhia Brasileira de Dragagem (CBD) uma maior eficiência operativa. Visando atender a essa demanda e, também, à execução de projetos de engenharia portuária mais sofisticados, a Companhia comprou 3 (três) sistemas de posicionamento eletrônico MDS (Mini-Ranger Dredge System - Motorola)

para equipar suas dragas auto-transportadoras. A compra de tais sistemas, baseados em minicomputador, resultou de uma escolha ousada, porém criteriosa, para a qual muito contribuiu a experiência adquirida pela CBD no arrendamento de dragas auto-transportadoras estrangeiras. Essas dragas operaram sempre equipadas com sistemas de posicionamento e a Companhia teve oportunidade de examinar diversos deles, a maioria dos quais controlados por hardwares não inteligentes. Eram, de um modo geral, sistemas de pouca flexibilidade.

Um sistema de posicionamento eletrônico para dragagem, baseado em minicomputador, tem, genericamente, a seguinte configuração: controlador e periféricos. O controlador, o coração do sistema, constitui-se, em geral, de um minicomputador de OEM. Os periféricos, desempenhando três funções básicas (controle, entrada e saída), são, respectivamente: painel, radar e plotador. O funcionamento desses sistemas de tempo-real consiste na execução de ciclos periódicos de aproximadamente um segundo de duração, nos quais são desempenhadas as seguintes tarefas: medição, pelo radar, das distâncias da embarcação às estações de terra; determinação, pelo minicomputador, da posição geográfica correspondente, por meio de cálculo de triangulação e marcação dessa posição, em escala adequada, no plotador. Assim, durante a fase de operação, os sucessivos e contínuos ciclos que o sistema executa, automaticamente vão deixando riscada na carta do plotador a trajetória que está sendo desenvolvida pela draga. Esse recurso permite ao draguista a aferição instantânea da trajetória em relação às trilhas e aos off-sets previamente estabelecidos, para caracterizar pontos demarcatórios da zona a ser dragada ou altos-fundos perigosos para o equipamento. Em nosso caso particular, isto é, SISTEMA MDS-MOTOROLA, o minicomputador empregado pelo fabricante era o INTERDATA-70, e os periféricos eram: radar MRS-MOTOROLA e plotador HOUSTON 6650. Havia ainda um TELETYPE para carga do módulo objeto do programa de posicionamento no minicomputador.

A bem da verdade, não se satisfez a CBD, ao longo daqueles anos, com o rendimento obtido na utilização dos seus sistemas de posicionamento MDS-MOTOROLA. Tanto assim que, por ordem da Diretoria Técnica, já vinham sendo realizados, desde 1975, estudos para levantamento das razões pelas quais tais sistemas haviam se tornado inoperantes, e qual seria uma possível solução para o problema. Ao cabo desses estudos, em meados de 1976, a Diretoria Técnica da CBD tinha em mãos um relatório, no qual estavam arroladas as principais razões do fracasso na "experiência MDS" e, como alternativa para superar o problema, as especificações preliminares de um anteprojeto (nacional), definido com apoio do NCE - Núcleo de Computação Eletrônica da UFRJ, denominado Projeto do Sistema de Posicionamento Eletrônico (PSPE).

Para fazer a análise das falhas de operação do equipamento, o anteprojeto tomou por base os diários dos técnicos de manutenção e as principais queixas apresentadas por engenheiros supervisores de draga que operaram com o MDS. A conclusão a que se chegou foi que, embora tivesse havido algumas paralizações do equipamento por operação imprópria, quase todas as panes eram devidas ao hardware, e se localizavam, em sua maioria, no minicomputador INTERDATA-70. A explicação dada para isso era a falta de robustez que ele apresentava em relação aos demais componentes do sistema (painel, radar e plotador), visto que neles não se registraram ocorrências que comprometessem por muito tempo o funcionamento do MDS. De certa forma, tal anomalia faz sentido, por que o INTERDATA é um minicomputador de aplicação convencional, e ali estava sendo utilizado fora das condições ambientais para as quais fora projetado, em que pese a tropicalização dos seus circuitos, realizada pela Motorola. Tomemos por exemplo o caso de corrosão que um dos MDS sofreu, no processador INTERDATA-70, quando

instalado a bordo da draga "Siete", em Santos: ficou de tal forma inutilizado que foi impossível recuperá-lo no país. Foi, em resumo, o que sucedeu a esses sistemas de posicionamento, onde a operacionalidade e manutenção dos minicomputadores a bordo deixaram a desejar, fazendo dos MDS uma constante fonte de problemas para as suas dragas.

Ainda nesse relatório (1976), precedendo a recomendação do PSPE, ficou constatado que as falhas por operação imprópria dos equipamentos eram, até certo ponto, justificáveis, uma vez que o MDS era de um manuseio extremamente complicado para sua finalidade mas que esse problema seria contornável com um intensivo programa de treinamento em operação. Porém, quanto às partes do hardware, não havia solução, senão a de substituir o controlador (INTERDATA-70), pois decididamente não havia como adequá-lo àquelas condições de trabalho.

Paralelamente, a procura, no mercado internacional, de uma alternativa segura para substituição dos MDS não logrou sucesso. De fato, o re-exame desses diversos sistemas de posicionamento, cuja maioria da CBD já havia manipulado, rememorou vários e velhos inconvenientes, muitos dos quais os mesmos apresentados pelo MDS, o que permitiu concluir definitivamente que não estava ali a solução do problema. Na realidade, tudo isso se inseria num único contexto: o da filosofia de projeto, isto é, esses equipamentos tinham sido todos, sem exceção, projetados visando aos mercados europeu e americano, cujas condições climáticas e sócio-econômicas sabemos diferentes das nossas. Restou à CBD coragem para enfrentar o problema dentro de nossas fronteiras tecnológicas, sem temer pelo arrojado da idéia, porém sem perder de vista a máxima adequação da solução ao problema como um todo (PSPE).

O projeto PSPE preconizava, como solução, a substituição do minicomputador (INTERDATA-70) por um microcomputador. Na justificativa desse projeto nenhuma consideração do tipo Benefício-Custo foi feita, como seria usual em qualquer episódio desse tipo, pois os "micros" substituíram os "minis" em diversas aplicações independentes (STAND-ALONE) do processamento eletrônico de dados, devido ao seu baixo custo-performance. A verdade é que, em se tratando de posicionamento de dragas, o preço desses equipamentos tornou-se absolutamente desprezível, diante do seu benefício. Para se ter uma idéia, basta lembrar que o valor de uma draga autotransportadora é da ordem de Cr\$ 300.000.000,00 (trezentos milhões de cruzeiros) e seu custo operacional mensal, cerca de Cr\$ ..... Cr\$ 12.000.000,00 (doze milhões de cruzeiros).

Bem assimilado pela Diretoria da CBD, o projeto PSPE, ainda que representasse um certo risco, justificava o empreendimento, pois seu custo estimado era pouco significativo em relação ao investimento MDS. Por outro lado, a aquisição de novos sistemas de posicionamento, no mercado internacional, além de tecnicamente não recomendável, representava um vultoso investimento num momento em que as importações brasileiras sofriam restrições na área governamental, acrescidas ainda de um rigoroso controle exercido pela CAPRE e CACEX nesse tipo de operação. Por outro lado, o projeto PSPE, de perspectivas mais amplas, não limitando a aplicação do equipamento somente à dragagem, delineava, ainda, a possibilidade de sua extensão às operações de sondagem, visando à automação do levantamento batimétrico há muito perseguido pela CBD.

O projeto teve sua execução iniciada em JULHO de 1976, ocasião em que se tinha, na CBD, a notícia do lançamento, pela Motorola, de um novo MDS: o MRDP - Automatic Positioning System baseado, como o nosso SPE, em microprocessador, o que nos deu,

mais uma vez, a certeza de estarmos trilhando o caminho certo na busca da solução do problema. Para auxiliar a CBD no empreendimento, foi estabelecido um CONVÊNIO bem amplo com o NCE-UFRJ, buscando um assessoramento a nível de hardware e software que permitisse o detalhamento e execução do projeto. Posteriormente, foi contratada a montagem do protótipo de hardware do controlador a uma firma particular. Não obstante os atrasos e problemas ocorridos na entrega dos serviços por parte dessa última, o protótipo estava fazendo o seu primeiro teste experimental no dia 20/06/78, a bordo da Draga Macapã, no futuro porto de Sepetiba, com pleno êxito.

### 1.3 - PROJETO DO SISTEMA DE POSICIONAMENTO ELETRÔNICO (PSPE)

Fique claro, desde já, que o SPE não é, em princípio, um sistema de posicionamento eletrônico original em sua concepção. Muito pelo contrário, foi nosso intuito, quando da sua definição, nos valermos, ao máximo, da experiência adquirida pela CBD com o uso de diversos sistemas de posicionamento eletrônico ao longo desses anos, a fim de tornar o SPE o mais adequado possível à nossa realidade. Pelo fato de a idéia do projeto ter-se originado, conforme vimos, da tentativa de recuperarmos o sistema MDS, é inegável que coube a este, dentre os diversos sistemas manipulados, a maior contribuição ao SPE.

O uso do MDS na CBD mostrou que, em sua concepção, o sistema atendia, de certa forma, às suas necessidades em posicionamento eletrônico. Por essa razão, quando, no relatório de 1976, abordamos o problema junto à diretoria, tivemos o cuidado de sugerir como solução para recuperá-lo, a substituição do INTERDATA-70 por um microcomputador. De certo modo, não deixava de ser um eufemismo, pois, para sermos claros, deveríamos estar, na verdade, propondo a definição de um novo sistema de posicionamento (SPE), aproveitando apenas sua filosofia e seus periféricos, cuja robustez e operacionalidade já mais haviam comprometido o seu funcionamento.



O painel de controle (MDS) não tinha como ser aproveitado, em razão de sua identidade lógica com o MDS. Assim, passaram a integrar o sistema SPE o plotador HOUSTON 6650 e o radar MRS-MOTOROLA. Nessa estratégia, além do aproveitamento pela qualidade (precisão, operacionalidade e robustez) desses periféricos, havia a conveniência de manter-se no SPE, através da adoção do radar MRS-MOTOROLA, o mesmo método de triangulação por interseção circular do sistema MDS. Essa era, a nosso ver, a maior vantagem que ele levava sobre seus concorrentes que, em geral, usavam outros tipos de radar, cuja tecnologia implicava em método de interseção hiperbólica, menos prático portanto.

Em decorrência da diretriz exposta acima, na definição do sistema de posicionamento (SPE), para completar o hardware, ocupou-se apenas de especificar, para a montagem do projeto PSPE, um controlador baseado em microprocessador, cuja característica mais importante, deveria ser a de suportar as condições de trabalho a bordo das dragas, e um novo painel, a êle associado, para controle do sistema, equipado com display e teclado, tendo em vista maiores facilidades de operação. Ficaria para o software, a tarefa principal do projeto, uma vez que, totalmente reformulado, a nível de análise e programação, deveria ser ele o suporte de toda a criatividade que no SPE incorporamos, em nome de uma maior adequação às novas necessidades de trabalho.

#### 1.4 - OBJETIVOS E LIMITAÇÕES DO PSPE

Para se ter uma idéia mais concreta dos objetivos a serem atingidos com a definição do SPE, traçaremos a seguir um paralelo entre ele e o seu antecessor, o MDS, segundo os pontos-de-vista mais importantes do problema: robustez e operacionalidade.

As condições ambientais a bordo de uma draga em operação, são em geral, uma resultante de diversos fatores agentes da corrosão que lá normalmente atuam: marisia, emanações gasosas da dragagem, umidade e calor. Daí porque a robustez de um equipamento de posicionamento constitui fator decisivo na determinação de sua performance e vida útil. Conforme vimos no MDS, o hardware do controlador (minicomputador INTERDATA-70), pela sua tecnologia de circuito integrado, configurava, devido ao elevado número de placas, um volume muito grande de "eletrônica", propiciando à corrosão uma grande área de ataque, não havendo pois, proteção por tropicalização, que bastasse para protegê-lo.

Já o controlador do SPE, um microcomputador adequadamente dimensionado para a aplicação, pelo emprego de uma tecnologia bem mais moderna, LSI (LARGE SCALE INTEGRATED), reduziu drasticamente o volume do hardware e, conseqüentemente, sua dissipação calorífica, propiciando seu encapsulamento praticamente hermético. Sua estrutura compacta permitiu, definitivamente, evitar a corrosão em seus circuitos. Também podemos esperar uma maior MBTF (MINIMAL TIME BETWEEN FAILS) como decorrência da redução do número de seus componentes. Por outro lado, o caráter modular de sua estrutura e a característica PLUG-IN dos seus componentes tornarão a manutenção do hardware bastante simplificada, podendo mesmo ser executada no campo.

A operacionalidade foi sem dúvida o segundo maior problema apresentado pelo MDS. De fato, não foram poucas as oportunidades em que o sistema deixou de funcionar, por operação indevida, pois o pessoal de bordo, mesmo com a ajuda do engenheiro supervisor da draga, não conseguia recuperar o sistema, em caso de erro, durante a operação, ou mesmo, dar a partida nele, sendo o técnico de manutenção, inutilmente, várias

vezes, chamado a intervir. Não caberia tão somente, ao despreparo da tripulação da draga, em lidar com um equipamento de processamento eletrônico de dados, a culpa pelas ocorrências, mas também, à própria complexidade dos procedimentos de operação do MDS, que entre outras coisas, exigia do operador o conhecimento de aritmética binária e hexadecimal para manuseio e leitura de chaves e luzes no painel do minicomputador, o que, convenhamos, não é coisa das mais práticas.

Para se ter uma idéia, a operação MDS continha dois procedimentos básicos: o de partida e o de inicialização do sistema. O de PARTIDA compreendia a carga do programa de posicionamento em módulo objeto por Teletipo, carga essa que se iniciava com a prévia introdução dos cinquenta bytes do programa bootstrap de Teletipo, na memória do INTERDATA, por meio das chaves binárias. Dada a baixa confiabilidade da leitura da fita de papel no Teletipo e equívocos cometidos na carga do programa bootstrap, este procedimento era repetido diversas vezes até ser bem sucedido. O procedimento da INICIALIZAÇÃO não menos complicado, requeria do operador a leitura ou memorização de cerca de trinta passos, a serem executados no painel do MDS após a carga do programa de posicionamento. Compreendia ele o acionamento de chaves seletoras e decimais para introdução dos dados básicos da triangulação. Resumindo: seu maior problema era não haver um dispositivo visual para o usuário confirmar os valores "inputados", o que só era possível ao final de todo o procedimento, quando o próprio sistema entrando ou não em funcionamento, permitiria tirar essa conclusão.

Já o SPE, beneficiado pelo modernismo de sua tecnologia, prescinde de um procedimento de PARTIDA, uma vez que todo o seu software está gravado em memória EPROM (Erasible Programmable Read Only Mememory), ficando ele imediatamente disponível para o usuário, ao simples

pressionar da tecla "LIGAR", não havendo, pois, que se fazer a carga do software. Por outro lado, seu painel, equipado com display e teclado, permite, juntamente com o software, introduzir no SPE uma filosofia conversacional tal, que dispensa o usuário da memorização de passos no procedimento de INICIALIZAÇÃO do sistema, através da exibição de perguntas e mensagens no display, facilitando, inclusive, a recuperação do sistema no caso de erros.

O SPE, se comparado a outros sistemas de posicionamento eletrônico recentemente lançados, a exemplo do MRDP - Automatic Positioning System - Motorola, já aludido, tem certas limitações, chegando mesmo a ser um produto bem menos sofisticado. Na verdade, a diferença de enfoque dos respectivos projetos explica tudo: no SPE, conforme vimos, nenhuma feição (FEATURE) foi incorporada ao sistema que não tivesse a sua justificativa em termos da aplicação, ao passo que, nesse outro sistema, onde já estão normalmente incorporados as mesmas vantagens de robustez e operacionalidade do SPE, a configuração em aberto do hardware, pode, opcionalmente, receber periféricos tais como: sonar de profundidade, indicador de trilha, relógio de tempo-real, sensor de temperatura, fita magnética para aquisição de dados, etc, tendo em vista atender às demais aplicações do posicionamento eletrônico. Não obstante, recordemos que, conforme nossos estudos mostraram, o projeto do SPE pode ser estendido, visando também a outras aplicações do posicionamento eletrônico, em nosso caso, a sondagem batimétrica.

## CAPÍTULO II

### DEFINIÇÕES

#### 2.1 - PROPÓSITO E USO DO SISTEMA

O SPE é um sistema de posicionamento eletrônico de tempo real, destinado a posicionar dragas tipo autotransportadoras, durante a execução de serviços de dragagem, podendo, também, dependendo das condições geográficas do local da obra, fazer o acompanhamento da draga, quando em navegação de ida e volta, à área de "despejo de material".

O SPE, pela sua característica conversacional, não exige maior preparação por parte do usuário para sua utilização, uma vez que é programado para interagir com êle através de uma série de mensagens adequadas, prevenindo-o quanto a erros de procedimento ou cessão de dados ilegais, sendo também de fácil instalação e manutenção a bordo da embarcação.

O SPE, por ter sua metodologia de posicionamento baseada no processo de triangulação, requer unicamente do usuário, que ele planeje, devidamente, a tarefa de operação, escolhendo as bases da triangulação e definindo área de posicionamento, de tal modo que esses dados configurem uma "boa geometria" para a obra. Caso contrário, o sistema os rejeitará, antes mesmo de iniciar a operação.

## 2.2 - FILOSOFIA DO SISTEMA SPE

De acordo com os objetivos do projeto PSPE, o sistema assumiria dois compromissos básicos: ser robusto e operacional. Essa decisão implicaria, de imediato, em ter-se nele um controlador com tais qualidades, uma vez que o controlador não só constitui a parte mais delicada do hardware, como também determina, com auxílio do software, as facilidades operacionais que o sistema venha a ter.

Reunindo toda a inteligência de um sistema de posicionamento eletrônico, o controlador tem a finalidade de ser o responsável pelo seu funcionamento, cuidando, basicamente, entre outras, das seguintes principais tarefas: aquisição de dados, edição de mensagens, controle dos periféricos do sistema (ploter e radar), cálculo do posicionamento e gerência do tempo real. Por se tratar de uma aplicação em tempo real, o controlador deve, também, desempenhar certa performance, a fim de não comprometer o tempo de resposta especificado para o sistema, definindo-se esse tempo de resposta, parâmetro fundamental do sistema, como o intervalo de tempo decorrido entre duas determinações consecutivas de posições.

O SPE, em seu primeiro desenho, proposto pelo ante-projeto de 1975, seguia a tendência da tecnologia, a qual já tivemos a oportunidade de abordar: apresentava, como novidade, um controlador para o sistema, baseado em microprocessador. Indo mais além, sugeria a adoção do microprocessador INTEL 8008, tendo em vista, principalmente, a disponibilidade de recursos nacionais, a nível de software e hardware, para seu manuseio, uma vez que o NCE-UFRJ acabara de concluir com sucesso, o projeto de um "Terminal Inteligente", utilizando aquele microprocessador. No entanto, colocava-se, para uma posterior e mais apurada análise, a seguinte questão: que performance esperar desse novo microprocessador, sabendo-se que ele tinha menor poderio computacional que um mini como o INTERDATA-70 ?

Um estudo detalhado de um processador deve ser feito, se é nosso objetivo com ele implementarmos um controlador de um sistema de posicionamento, examinando-se a arquitetura de seu hardware, com vistas ao software, no intuito de verificar se ela é capaz de executar as tarefas para ele previstas, dentro da performance desejada. Para tanto, é de capital importância avaliarmos seu desempenho provável, principalmente, na execução de operações aritméticas de ponto flutuante, pois o cálculo do posicionamento, sendo indiscutivelmente a maior das tarefas do controlador, exige-as em quantidade. Esse fato, não sendo devidamente considerado, poderá comprometer totalmente o tempo de resposta do sistema.

Por ocasião da fase de detalhamento do PSPE, como meta de projeto, foi estabelecido o tempo de resposta máximo de 1.5 segundos para o sistema SPE. Esse intervalo de tempo foi estimado, levando-se em consideração dois fatores básicos: velocidade máxima da draga em operação (4m/s) e erro médio provável na determinação do posicionamento com o radar MRS(+ 6m). Objetivava ele garantir, no mínimo, uma determinação de posição por círculo de incerteza do posicionamento (raio = erro), ficando claro que performances melhores que essa eram almeçadas, pois dariam ao sistema a condição de efetuar determinações redundantes dentro desse círculo, o que proporcionaria maior confiabilidade no traçado gráfico do posicionamento no plotador.

Constituindo-se, sem dúvida, em dado preliminar do problema, o tempo de resposta do MDS (cerca de 0.5 segundos), para ser melhor avaliado, obrigava-nos a levar em conta algumas características fundamentais de hardware do INTERDATA, que por sinal, bem o habilitavam para o tipo de tarefa: ciclo de máquina adequado; acesso a 16 bits de memória; divisão e multiplicação para 32 bits; e um "set de instruções" equivalente

ao do sistema IBM 360. Donde se concluía, de imediato, mesmo sem uma contabilização mais rigorosa dos tempos prováveis de execução das operações em ponto flutuante, que, dificilmente, poderíamos atingir o tempo de resposta planejado (máximo = 1,5s) usando o INTEL 8008 na sua implementação, uma vez que, pelas suas características de hardware: ciclo de máquina de 12 usec, apenas 48 instruções e acesso a 8 bits, não parecia ser o 8008 uma UCP adequada para o serviço. E mesmo contando com uma otimização no desempenho do software, seria, ainda, certamente necessário para seu aproveitamento, complementar o hardware do controlador com circuitos de ponto flutuante, o que estava fora de cogitação.

Devido ao grau de amadurecimento, por nós, já atingido àque la altura, quer na implementação de software, quer no estudo do hardware do microprocessador INTEL 8008 (e demais produtos paralelos da linha daquele fabricante), optou-se, tendo em vista esse problema de performance, pela substituição do 8008, por um microprocessador mais poderoso e compatível, lançado no mercado, pelo próprio fabricante, como seu sucedâneo. Esse processador, INTEL 8080, pelas novas características de seu hardware: ciclo de máquina de 2 usec, isto é, seis vezes mais rápido, e um conjunto mais amplo de instruções, totalizando 78, embora não fosse, ainda, o ideal a ser escolhido para aplicação, poderia dar, sem dúvida, ao sistema, a performance desejada, dependendo da eficiência do software.

Assim sendo, uma maior atenção foi dada ao software no sentido de otimizá-lo para um melhor desempenho nos cálculos do posicionamento (operações de ponto flutuante). Desse esforço resultou a concepção de uma aritmética inteira especial que é capaz de executar, bem mais rapidamente, operações de ponto flutuante em um domínio numérico restrito, valendo-se artificialmente, de operações de aritmética de ponto fixo. Sua



implementação, bem mais simples que a da de ponto flutuante, se limitações introduzisse, estariam elas já de antemão com pensadas pela garantia de um desempenho satisfatório do sis tema.

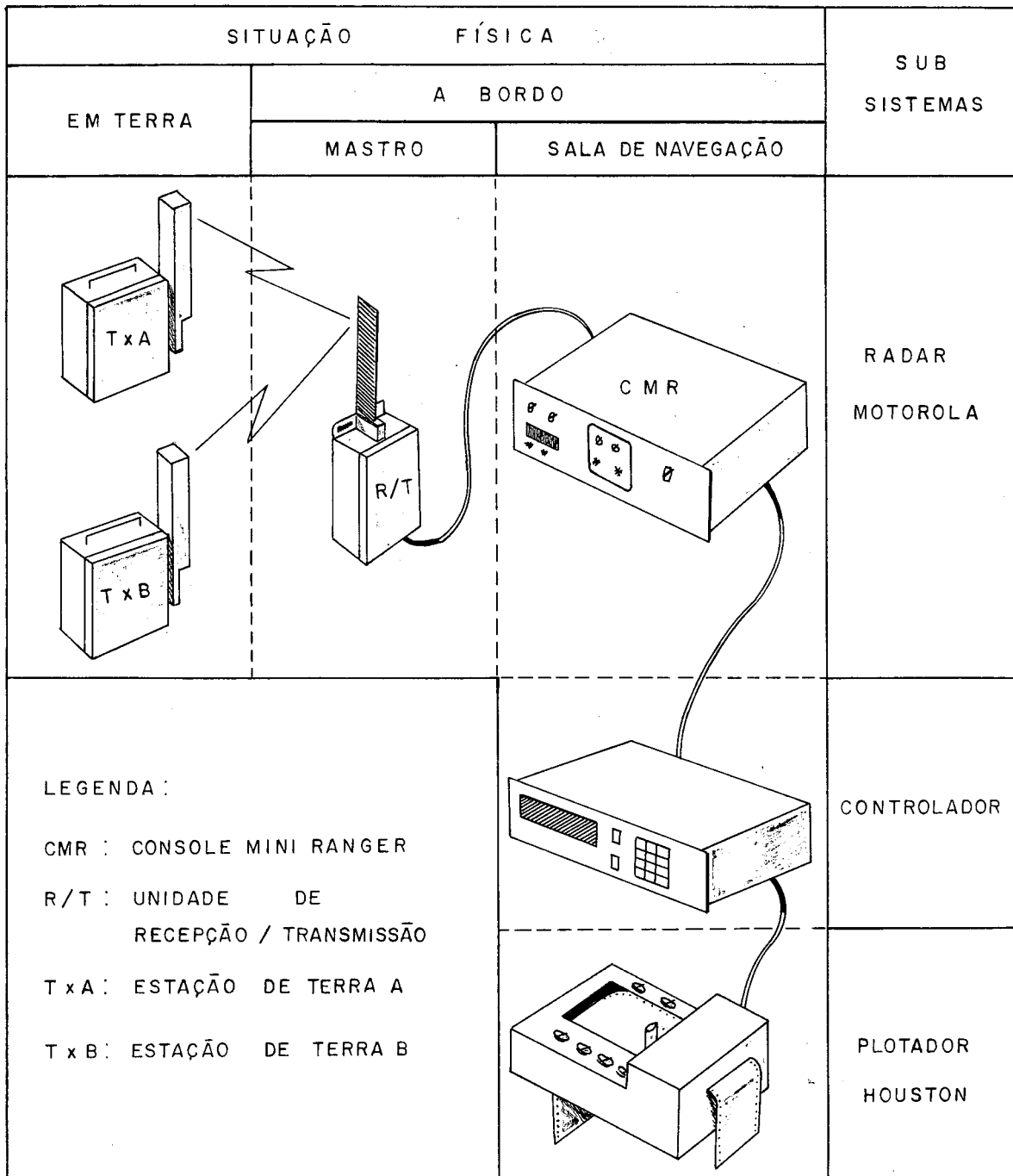
Fixado o processador (INTEL 8080), a arquitetura do controlado r começava a ganhar forma, na definição de um microcomputado r que a comporia a partir do dimensionamento de memória, inter faces internas e interfaces externas. Memória bastante para armazenar todo o software e prover ainda espaço para dado s. Interfaces internas para controlar os periféricos do painel de controle do sistema. Interfaces externas para se comunicar com os demais componentes do sistema: ploter e ra dar.

Completando o controlador do sistema, foi concebido um painel de controle para ser monitorado pelo software. A fim de facili tar a operação do sistema, e tendo em vista os objetivos de conversacionalidade do projeto, escolheu-se seus periférico s: display alfanumérico para dispositivo de exibição visual de mensagem e teclado numérico para a entrada de dados. Disp ãe ainda o painel, além do botão liga/desliga, de um outro, para interrupção do sistema. Essa interrupção, de caráter imedi ato, devolve o controle do sistema ao operador em qualquer etapa do processamento.

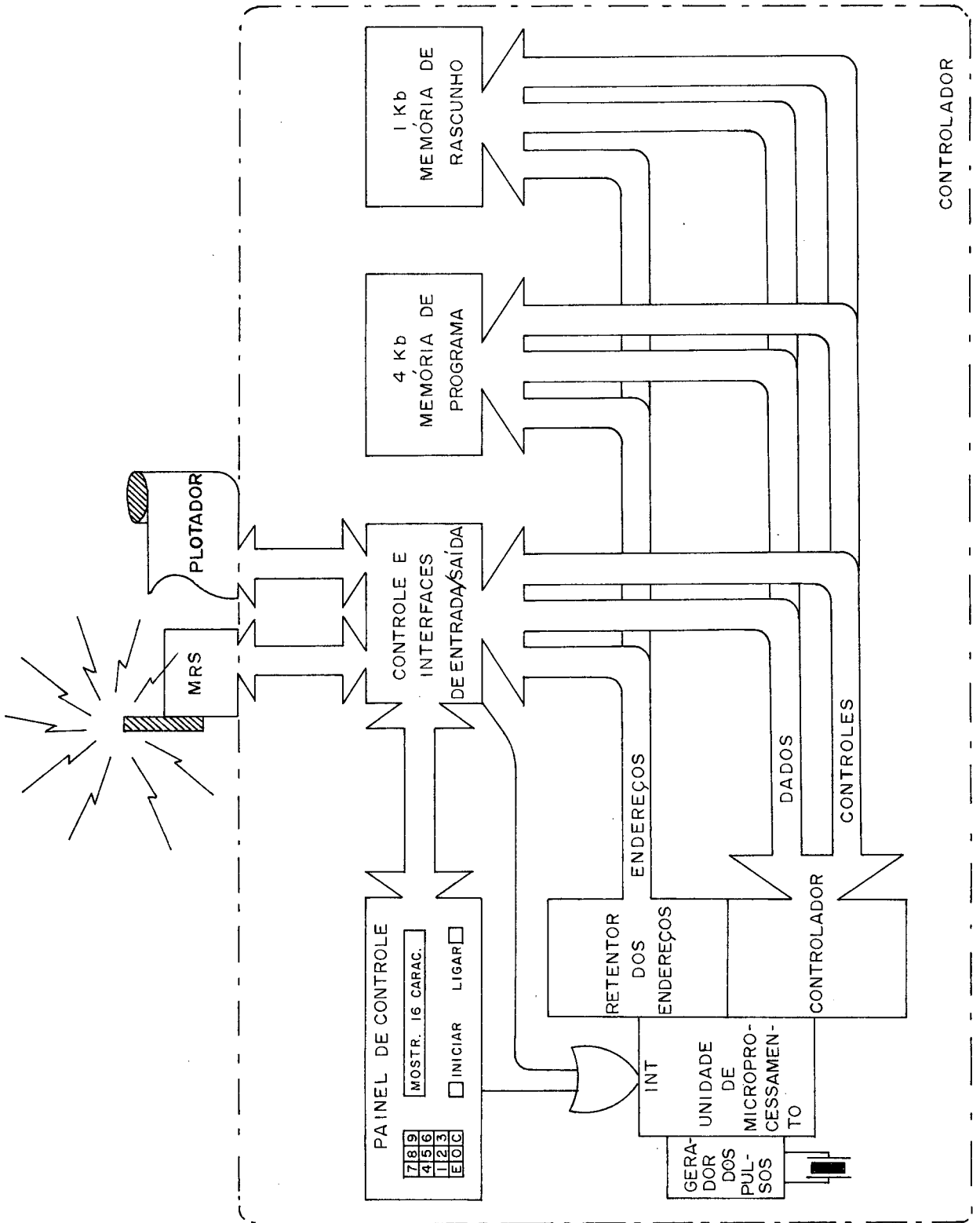
Desse consenso de idéias e opções, resultou uma especificaç ão mais formal de um controlador, para implementação e monta gem por terceiros. Esse controlador, juntamente com um software totalmente projetado em função dele, daria ao sis tema SPE as características desejadas: robustez e operacional idade.

### 2.3 - DESCRIÇÃO DE HARDWARE DO SISTEMA SPE.

O Sistema de Posicionamento Eletrônico SPE compreende os componentes mostrados na figura abaixo: CONTROLADOR, RADAR MOTOROLA e PLOTADOR HOUSTON.



A interligação funcional desses componentes é mostrada pelo diagrama abaixo:



## C O N T R O L A D O R

O controlador do sistema SPE consiste de um microcomputador, baseado no microprocessador 8080A da INTEL, compreendendo, do ponto-de-vista funcional, as seguintes partes: microcomputador propriamente dito e painel de controle.

### **a) MICROCOMPUTADOR.**

O microcomputador, centro do controle do sistema, foi implementado da maneira mais simples e compacta possível, com o auxílio de uma arquitetura pré-fabricada pela INTEL. Esta é constituída de uma série de micro-estruturas eletrônicas em configuração aberta, e denominada SBC 80/10 (single board computer). Esse processador, praticamente, supriu as necessidades básicas de suporte computacional do controlador, em termos de memória e entrada/saída, ou seja: Uup, 1Kb de RAM, 4 Kb de ROM e interfaces de E/S paralela para os periféricos internos e externos ao controlador (display e teclado; plotter e radar).

### **UNIDADE CENTRAL DE PROCESSAMENTO.**

A UCP consiste do microprocessador 8080A, do gerador de clock (8224) e do controlador do sistema (8228). Sendo ela o núcleo do microcomputador, executa todas as funções do processamento e fornece uma referência de tempo estável a todos os outros circuitos do sistema, gerando, também, todos os sinais de controle e endereços necessários ao acesso à memória e às partes de E/S. A UCP é capaz de executar qualquer uma das setenta e oito instruções do 8080A, e responder às requisições de interrupção, oriundas de acesso direto à memória, ou de 'WAIT' para memórias e periféricos, cujo tempo de acesso seja maior do que o ciclo de tempo dela.

O microprocessador 8080A contém seis registradores gerais de oito bits e um acumulador. Eles podem ser endereçados individualmente ou em pares, tornando-se operadores de precisão simples ou dupla. O 8080A possui, também, um contador de programa de 16 bits que permite o endereçamento direto até 64K bytes de memória, e uma pilha (stack) externa, que pode ser definida em qualquer ponto do espaço endereçável dessa memória. Essa pilha permite, de forma praticamente ilimitada, o encadeamento de subrotinas, bem como operações de salva/restaura do contexto da máquina. A filosofia de barras adotada no 8080A (16 linhas de endereço e oito linhas bidirecionais de dados) é usada para facilitar a interface com a memória e entrada/saída.

O conjunto de instruções do 8080A permite uma programação eficiente. O grupo de instruções envolvendo o acumulador inclui operações aritméticas e lógicas com endereçamento direto, indireto (através dos registradores) e imediato. O grupo de MOVE, LOAD e STORE permite que se troquem dados de 8 ou 16 bits entre a memória, os registradores e o acumulador, utilizando qualquer um dos modos de endereçamento. As instruções de JUMP incondicional, condicional ou computado possibilitam desvios a diferentes porções de um programa. O manuseio de subrotinas é efetuado por meio das instruções de CALL e RETURN condicional ou incondicional. As instruções de RESTART são utilizadas na operação de interrupção. O 8080A possui, ainda, instruções que permitem incrementar e decrementar a memória, os registradores gerais, o acumulador, os registradores duplos e o apontador da pilha.

## MEMÓRIA

A memória tipo ler/escrever, RAMs estáticas 8111 da INTEL, totalizando 4 Kb, compreende o espaço endereçável de  $3C00_{16}$  a  $3FFF_{16}$ . Tem por finalidade permitir o armazenamento de dados e servir de rascunho para os cálculos. A memória do tipo "leitura somente", EPROMs 8708 da INTEL, apagáveis por ultra-violeta, com cerca de 4Kb, endereçável de  $0000_{16}$  a  $0FFF_{16}$ , destina-se ao armazenamento permanente de todo o software. O tempo de ambas as memórias é da ordem do ciclo da 8080A, não necessitando elas, portanto, de estados de espera (requisição de WAIT).

## INTERFACES

As interfaces visam atender tanto aos periféricos do painel (internos ao controlador), quanto aos do sistema (externos ao controlador), por meio de linhas paralelas, provenientes de três integrados de interfaceamento tipo PPI 8255 (programmable peripheral interface).

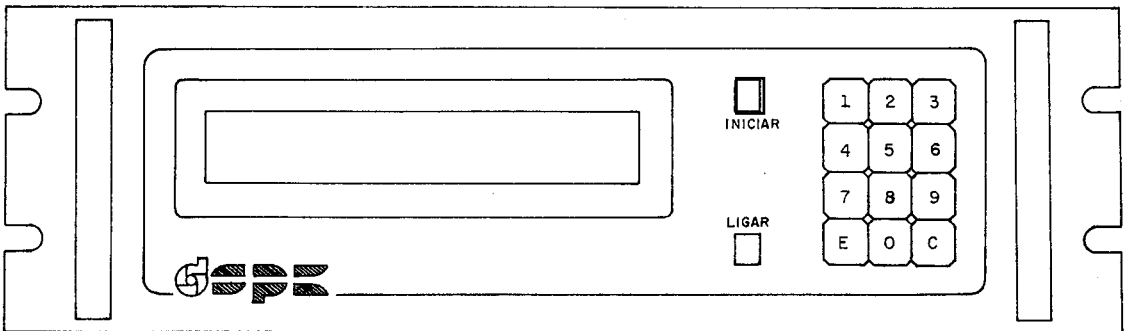
A 8255, por ser programável, é de altíssima flexibilidade, pois seu modo de operação, determinado por programação prévia, faz com que ela se adapte aos mais diferentes tipos de protocolo que um periférico possa exigir.

## INTERRUPÇÃO

Um único nível de interrupção é disponível ao sistema. Quando uma requisição de interrupção é reconhecida pela UCP, o componente 8228 força a execução de uma instrução de RESET 7. O processador, respondendo, suspende o programa em andamento, e passa a executar a rotina de serviço da interrupção, no endereço  $0038_{16}$  da memória, independentemente da origem da interrupção, externa ou interna.

## b) PAINEL DE CONTROLE

O painel de controle do microcomputador tem por finalidade acionar o controlador e permitir a operação do sistema, de um modo conversacional, dispondo, para isso, dos seguintes elementos: botão liga/desliga, botão de interrupção, teclado numérico e display alfanumérico.



### **BOTÃO LIGA/DESLIGA**

Uma vez acionado, liga o controlador e comanda um RESET para o microcomputador.

### **BOTÃO DE INTERRUPTÃO**

Permite ao operador provocar uma interrupção (externa) na Uup do controlador, assumindo o controle do sistema em qualquer etapa do processamento.

### **TECLADO NUMÉRICO**

O teclado DIGIPONTO possui ao todo doze teclas. Dez delas correspondem aos dígitos 0 a 9, sendo as duas restantes, teclas de controle, isto é, ENTRA (enter) e APAGA (clear). O teclado é interfaceado à UCP, através da 8255. Seu "debouncing" é feito por software.

## DISPLAY

O display utilizado é o SSD1000-0041 da Burroughs. Possui 16 posições para exibição de caracteres alfanuméricos. Cada caracter é exibido numa matriz de 5 x 7 pontos, com espaçamento de duas colunas entre os caracteres. Os caracteres são exibidos colocando-se o código binário respectivo de 6 bits nas linhas de entrada de dados, um a um, para cada posição do display. Um gerador de caracteres fornece até 64 tipos de caracteres diferentes. Quando da edição de uma mensagem, o primeiro caracter entra na posição mais a esquerda do display e os caracteres subsequentes entram sequencialmente no próximo espaço disponível à direita. O código do caracter deve ser colocado nas linhas de entrada durante 33 usec aproximadamente. Os caracteres vão sendo apresentados até o 16º, indo o 17º para o lugar do primeiro à esquerda. O display pode ser apagado a qualquer tempo.

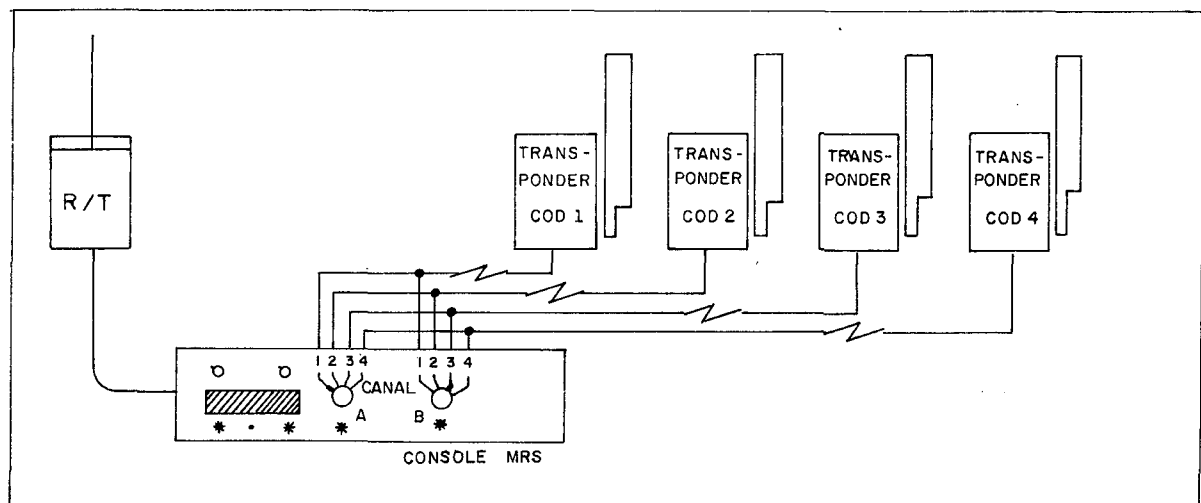
A interface do display com a UCP é feita através da mesma 8255 que faz a interface com o teclado. São utilizadas oito linhas para a comunicação entre display e UCP, das quais seis são as linhas de dados e duas são para controle (Clear e Data Present). A linha Data Present é utilizada para indicar ao display que deve retirar o código de um caracter enviado nas linhas de dados. A linha Clear é utilizada para limpar o display (Clear Display).



## R A D A R   M R S   M O T O R O L A

O Radar MRS Motorola é um sistema de posicionamento circular fabricado pela MOTOROLA. Consiste de uma unidade Receptora/Transmissora ligada a um console digital, colocados na unidade móvel que se deseja posicionar, e de dois a quatro Transponders de Radar instalados em locais de coordenadas conhecidas (Estações de Terra). O console indica, de forma alternada e quase contínua, a distância da unidade móvel a duas dessas estações fixas, previamente selecionadas.

O MRS opera com o princípio básico do radar de pulsos. A unidade de Recepção/Transmissão (R/T), que é montada a bordo do móvel, é usada para interrogar os dois transponders de radar. O tempo decorrido, entre o sinal de interrogação transmitido e cada um dos sinais de resposta recebidos pela R/T, é usado para determinar a distância a cada um dos transponders. A distância, medida em metros, e a indicação do transponder a ela correspondente são exibidos pelo Console MRS. Estas mesmas informações são ainda, disponíveis, no formato BCD paralelo, através de um conector no painel traseiro do Console.



O sistema admite apenas dois pontos de referência: CANAL A e CANAL B. O operador do sistema deve escolher, então, qual transponder será colocado no CANAL A e qual no CANAL B. Esta escolha é feita girando-se, no console, o botão de cada canal para a posição que indica o código do transponder selecionado. A unidade R/T faz uma interrogação seletiva dos dois transponders selecionados, comutando os seus códigos de interrogação.

O ciclo de uma medição do MRS se inicia por meio de um sinal modulador, emitido pelo console, que ativa o transmissor da R/T, fazendo-o irradiar um sinal de RF (rádio frequência), codificado, em sua antena. Ao mesmo tempo, o transmissor passa um sinal de controle para o console, que faz disparar os contadores de distância. Enquanto isso, esse sinal interrogador, é processado, e uma resposta é gerada pelo transponder selecionado. O sinal de resposta, recebido na antena do R/T, é passado ao console, como sinal de parada para os contadores de distância, encerrando o ciclo da medição.

#### **UNIDADE RECEPÇÃO/TRANSMISSÃO (R/T)**

O Receptor/Transmissor consiste de uma unidade RF (rádio frequência) e um processador de vídeo. A unidade RF contém um receptor e um transmissor na banda C, uma fonte de potência e circuitos associados. A função do processador de vídeo é transmitir o sinal de modulação do console para o transmissor e o sinal do receptor para o console.

#### **CONSOLE MRS**

O console faz todo o processamento dos dados. Esta unidade consiste de um circuito codificador-decodificador, um circuito para contagem das distâncias, um comutador de canal, um display numérico e uma fonte de potência. A fonte de potência fornece as voltagens de operação para o console e para o R/T.

O cálculo de uma leitura começa, a partir da conclusão do quinto ciclo de medição de um canal. O número armazenado no contador de distâncias é dividido por 5, e o resultado é convertido em metros e exibido pelo display numérico. Simultaneamente, acende-se o indicador do canal correspondente. Essas cinco medidas têm a finalidade de diminuir o desvio médio da distância. A informação de distância e o canal ao qual pertence essa distância são também fornecidos no formato BCD, para acoplamento em computadores ou impressoras. O comutador de canal controla a interrogação alternada e seqüencial dos canais A e B. A velocidade com que as medidas são apresentadas pode ser regulada de uma por segundo até uma em cada 10 segundos, por meio de um botão no painel frontal, quando em operação off-line. No modo remoto, a velocidade é fixa, compreendendo a apresentação de cada leitura em 5.7 ms.

## **TRANSPONDER**

O transponder consiste de uma unidade RT (receptor/transmissor) na banda C, e de um decodificador de pulsos, para o reconhecimento do código dos pulsos de interrogação que chegam até ele.

Os transponders de RADAR são instalados em locais conhecidos e servem como ponto de referência. Os transponders recebem as interrogações codificadas oriundas do R/T. Essas interrogações são detectadas, amplificadas e decodificadas pelos transponders e, dependendo do espaçamento dos pulsos de interrogação (código para o qual o transponder foi ajustado), resulta uma resposta do transponder. As frequências de interrogação e de resposta estão fora da banda passante normal dos receptores de RADAR.

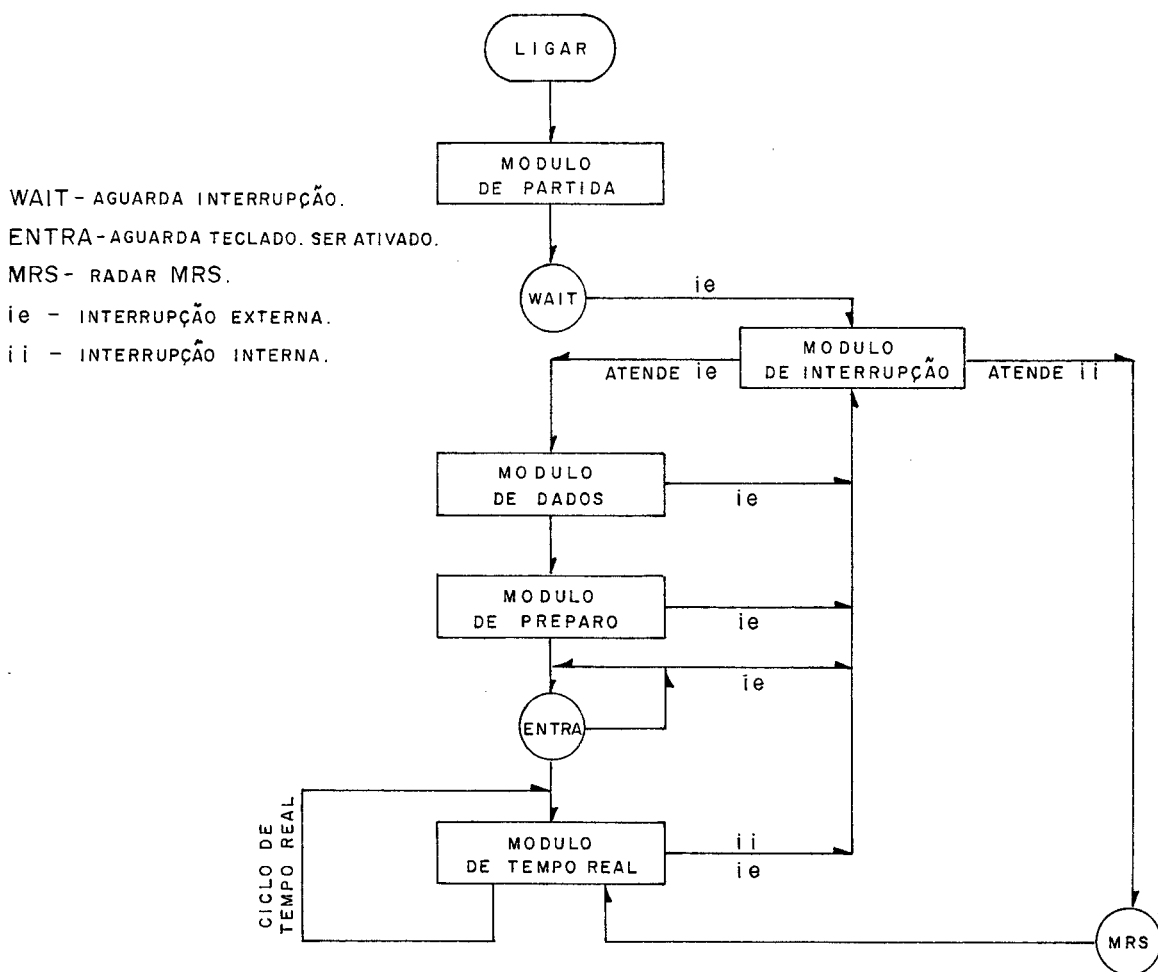
P L O T A D O R   H O U S T O N   6 6 5 0

O plotador HOUSTON 6650 é um plotter incremental bi-direcional, capaz de operar com diferentes tipos de sinais. O plotador utiliza motores incrementais bi-direcionais para os eixos da pena e do papel, capazes de produzirem incrementos de 0.1mm (STEP). Os motores executam um step de deslocamento quando a entrada percebe uma troca de informação de estado. O movimento da pena e o do papel são independentes. A combinação desses movimentos permite o deslocamento relativo da pena sobre o papel em oito direções fundamentais. A velocidade do plotador é de 18000 incrementos por minuto. O plotador executa, também, movimentos de 'pena ao alto' (PEN UP) e de 'pena a baixo' (PEN DOWN).

O plotador, após ser ligado, pode ser colocado em dois modos de operação: on-line ou off-line. A operação off-line permite ao usuário, através de botões de comando, inicializar (colocar na origem) a posição da pena sobre a carta do plotador, antes do início da plotagem. No modo on-line (remoto), o plotador é acionado por pulsos de comando, enviados em sua entrada, por seis linhas de dados compreendendo: PEN UP, PEN DOWN, X+, X-, Y+ e Y-. O status (BUSY, NOT-BUSY) do plotador não é fornecido durante a execução do comando. Cada comando, exceto PEN-UP/DOWN, tem a duração aproximada de 300 ms. Comandos enviados em espaçamento inferior a esse tempo ou inconsistentes (exemplo, Y+ e Y-) são ignorados.

## 2.4 - FUNCIONAMENTO DO SISTEMA SPE

O funcionamento do sistema SPE é regido por cinco módulos de software que compõem o seu programa de posicionamento (programa principal). Esses módulos são: DE PARTIDA, DE INTERRUPTÃO, DE DADOS, DE PREPARO e DE TEMPO REAL. A passagem do controle do sistema a cada um desses módulos se faz, em sua sequência lógica, na seguinte ordem: PARTIDA, DADOS, PREPARO e TEMPO REAL. O módulo de INTERRUPTÃO auxilia o sistema na transferência do controle entre módulos. O diagrama abaixo, esquematiza o funcionamento do SPE através dos módulos.



Em seguida iniciaremos o detalhamento do funcionamento do sistema sob controle de cada um desses módulos de software.

### SOB CONTROLE DO MÓDULO DE INTERRUPÇÃO

O sistema percebe uma interrupção através de uma transição positiva na linha INT do microprocessador 8080. Em resposta, a UCP suspende a execução do processamento, através de um RESET-7, passando o controle do sistema para o Módulo de Interrupção. Essa interrupção (de um único nível) é assíncrona, podendo acontecer em qualquer etapa do processamento.

O atendimento à interrupção se inicia, desprezando o endereço de retorno armazenado na STACK e desligando a CHAVE DE LEITURA do MRS. Em seguida, é analisada a procedência da interrupção, isto é, se interna ou externa. Se interna, o controle é desviado para a rotina MRS em seu ponto de entrada MRSRD, promovendo a leitura do dado e devolvendo o controle ao Módulo de Tempo Real. Se externa, recupera o sistema, isto é, coloca ponteiro da stack no topo, limpa display, cessa atividade do MRS e desvia para iniciar/re-iniciar a operação do sistema no Módulo de Dados.

### SOB CONTROLE DO MÓDULO DE PARTIDA

O ativar do botão liga/desliga, no painel do controlador do sistema produz no 8080 um POWER ON/RESET. A tensão de alimentação (POWER ON) faz com que o 8080 comece a operar imediatamente. O RESET restaura (zera) o conteúdo dos seus registradores: contador de programa, apontador da stack e registros de trabalho. O sistema começa a funcionar, executando as instruções do módulo de partida, armazenadas a partir da posição zero da sua memória.

O sistema inicializa o topo da STACK na posição  $4000_{16}$  da memória, e, através da programação das três PPI 8255, as interfaces com os periféricos. A programação se dá por meio da colocação da palavra de controle adequada ao modo de operação

de cada uma delas. Um sinal de Reset é enviado para o MRS, para evitar que ele comece a funcionar pelo simples fato de ter sido ligado. São estabelecidos valores iniciais (DEFAULTS) para as variáveis básicas do posicionamento.

O sistema atinge, assim, a condição de pronto, exibindo mensagem de aviso no display (SISTEMA NO AR!) e liberando a interrupção. Logo após, entra em estado de espera (HALT), aguardando a chegada de uma interrupção externa.

#### SOB CONTROLE DO MÓDULO DE DADOS

O sistema passa ao controle do módulo de dados após reconhecer uma interrupção externa. O processo conversacional de aquisição de dados tem, então, início. O usuário interage com o sistema, através de uma série de perguntas que lhe são apresentadas no display. O usuário vai respondendo no teclado a essas perguntas, passo a passo, com a digitação do dado correspondente. A digitação dos dados é controlada pelas teclas ENTRA e APAGA. À medida que digitados, os dados vão sendo codificados e armazenados na Área de Respostas, alocada em memória volátil pelo programa. O processo é cíclico, até que seja respondida a última pergunta, quando o sistema passa ao Módulo de Preparo, por meio de um desvio incondicional.

#### SOB CONTROLE DO MÓDULO DE PREPARO

Ao preparar o sistema para operar em Tempo Real, o módulo efetua a crítica dos dados básicos introduzidos e calcula as VARIÁVEIS ESTÁTICAS auxiliares do cálculo de posicionamento.

Inicialmente, o sistema faz a redução das coordenadas dos pontos A, B, START e STOP, de modo a minimizá-las em valor (positivo). À medida que faz a redução, detecta erro, se dela resultou variável reduzida  $>32767m$ . Verifica, também, se nos demais dados introduzidos, isto é, alturas, faixa e escala, foi encontrado algum valor superior a esse limite (2 Bytes). Em seguida, não havendo erro, um teste global dos dados é feito no sentido de se verificar sua validade do ponto-de-vista geométrico da triangulação. Se a geometria é satisfatória, o sistema prossegue calculando o restante das VARIÁVEIS ESTÁTICAS. A ocorrência de erros nesta fase faz com que o sistema aguarde (HALT) a intervenção do operador, que se dará por meio de uma interrupção externa (BOTÃO DE INTERRUPTÃO). Concluindo a preparação, o sistema inicializa as variáveis de controle da plotagem e desvia para o módulo de Tempo Real assim que o operador aprontar o plotador.

### SOB CONTROLE DO MÓDULO DE TEMPO REAL

O sistema apresenta no display o caracter de prontidão (\*), avisando que entrou em Tempo Real. O processo cíclico que vai caracterizá-lo é: obtenção dos RANGES das distâncias A e B, cálculo das coordenadas da posição e plotagem.

O ciclo de Tempo Real se inicia quando o sistema ativa o RADAR MRS, para a obtenção dos RANGES das distâncias. São realizadas, ao todo, dezesseis (16) leituras para cada canal do MRS (A e B). Para cada leitura, o controlador liga a CHAVE DE LEITURA do MRS e se deixa interromper (interrupção interna) por ela para recuperar (MRSRD) a leitura. Tão logo seja recuperada, a CHAVE DE LEITURA é desligada e o sistema ignora novas leituras do MRS, até que conclua o tratamento dessa.



O tratamento de uma leitura consiste: na verificação de sua validade, na acumulação do seu valor e na incrementação do seu contador, conforme seu canal (A ou B). Concluído o processo, isto é, feitas as 16 leituras para os canais, o sistema faz o MRS cessar a sua operação e passa à determinação da posição. Durante esse processo, o sistema pode enviar ao display mensagens de erro por falta de dado, ou por dado inválido.

O cálculo da posição começa com a determinação do RANGE da distância para cada canal, partindo das leituras acumuladas e tirando a média. Determinados os RANGES (A e B) o sistema analisa a validade da posição, isto é, se atende aos requisitos geométricos estabelecidos para o posicionamento. Se houver erro no posicionamento o sistema avisa com mensagem de erro adequada, e abandona a posição determinada, desviando para re-iniciar o ciclo.

A plotagem da posição é feita determinando-se as coordenadas da posição validada no sistema de coordenadas do canal, por meio de rotação e translação de sistemas. Antes da posição ser plotada, é examinada a sua plotabilidade. São plotáveis, posições cujas coordenadas caibam na escala da plotagem. Se a posição não for plotável, o sistema envia mensagem de erro e abandona a posição, desviando para re-iniciar o ciclo.

A plotagem da posição consiste no envio de comandos incrementais para o plotador, de modo a conduzir sua pena à posição correspondente àquela determinada pela escala, na carta do plotador. Feita a plotagem, o sistema volta ao início do ciclo para fazer nova determinação de posição. O sistema pode ser interrompido externamente pelo usuário em qualquer etapa do processo.

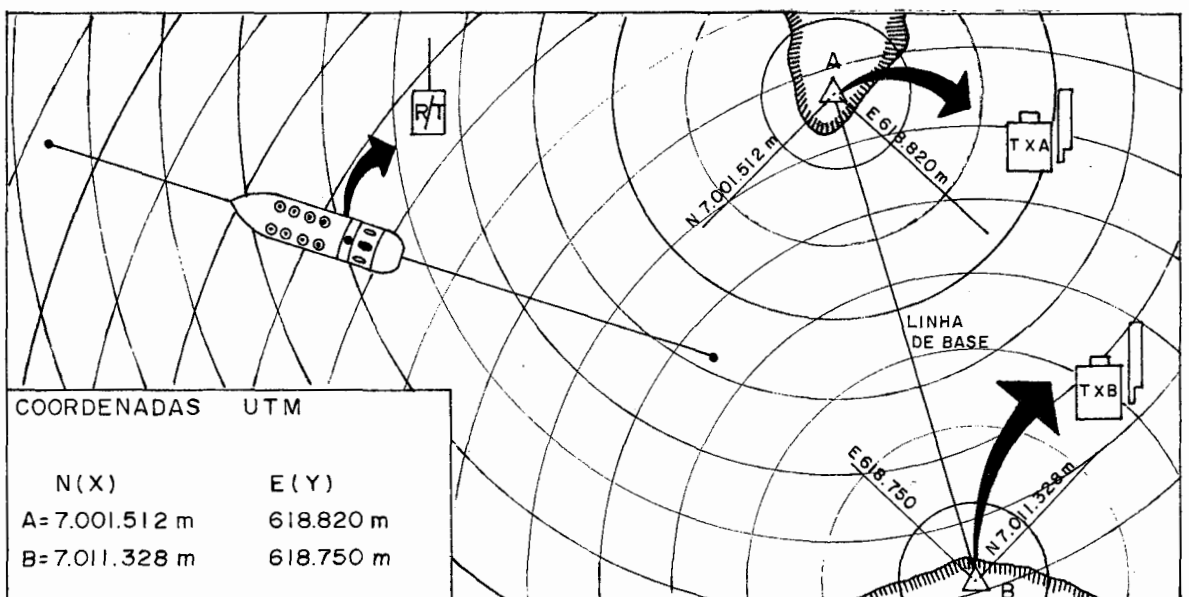
## 2.5 - A TRIANGULAÇÃO COM O RADAR MRS-MOTOROLA

O processo de triangulação consiste, basicamente, na determinação das coordenadas de um ponto, a partir das coordenadas conhecidas de dois outros pontos. Por compreender distâncias relativamente pequenas ( $< 50$  km), o seu emprego, em topografia ou em hidrografia, não leva em consideração a esfericidade da terra, isto é, o elipsóide, como faz a Geodesia. Assim sendo, a metodologia de cálculo desse processo de triangulação é uma simples resolução de problemas de triângulo em geometria plana: opera-se com o comprimento da linha de base (formada por dois pontos conhecidos) e a medida de dois de seus ângulos. A determinação desses ângulos, por instrumentos, apresenta duas modalidades de operação: intersecção "a vante" e intersecção "a ré", conforme seja feita com a ocupação, ou não, do ponto a determinar.

A hidrografia clássica utiliza-se de recursos óticos convencionais para determinar um ponto na triangulação de posicionamento. Opera por intersecção "a vante", pois, via de regra, o ponto (o objeto) a ser posicionado está flutuando sobre a lâmina d'água, não permitindo sua ocupação por instrumentos. Em virtude da movimentação constante das massas de água, quer por efeito de gravidade, vento ou maré, não é possível, na maioria dos casos, efetuar-se, em hidrografia, medições angulares, com a mesma precisão que na triangulação topográfica. Daí resulta que as normas de engenharia sejam bem mais tolerantes no que concerne ao erro cometido nessas determinações: cerca de 3 a 10 metros, conforme as condições locais da obra. Hoje em dia, os avanços da tecnologia aproximaram, por melhoria de acuidade na medição, os já conhecidos radares (desde a 2a. guerra) às fronteiras dessas tolerâncias, o que habilitou, de imediato, o seu emprego na triangulação hidrográfica.

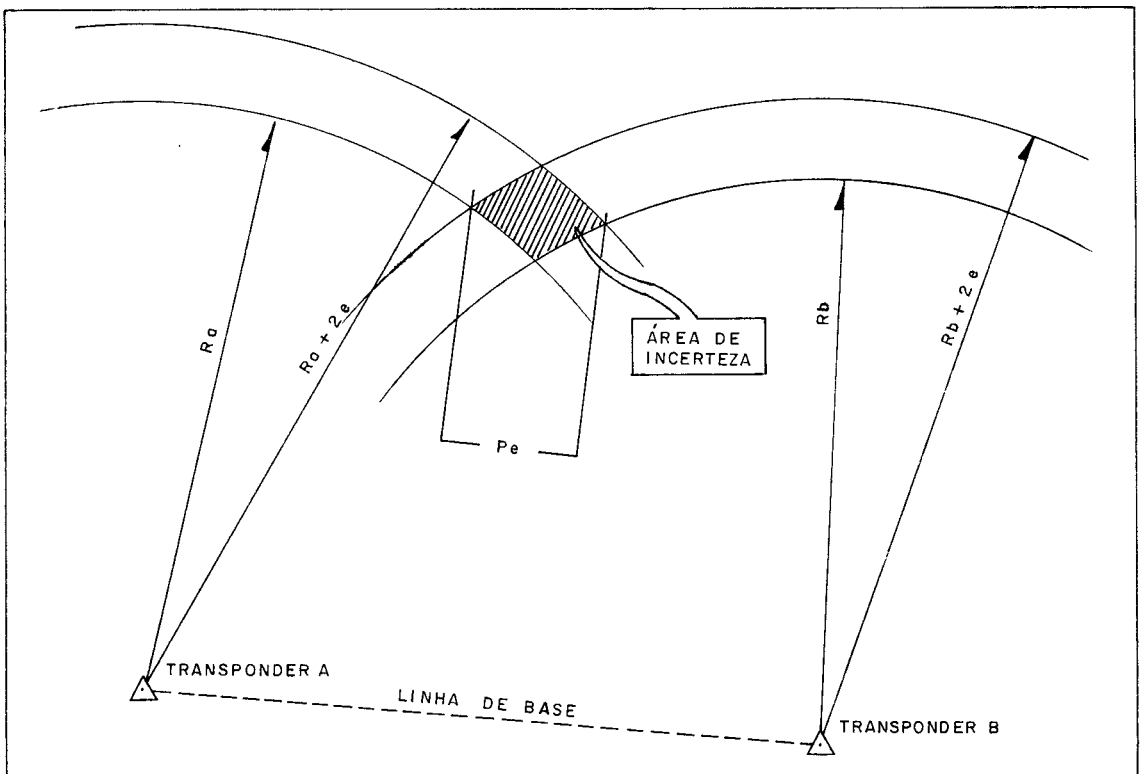
O posicionamento por meio do radar é um caso particular da triangulação plana conhecido como trilateração. É uma situação em que se conhece os dois outros lados do triângulo, ao invés de dois dos seus ângulos. Nesse caso, o radar, ocupando a própria posição a determinar, faz medições "a ré", sobre os pontos extremos de coordenadas conhecidas da linha de base. Tais medições são, em geral, "distâncias" ou "diferenças entre distâncias", do que resulta, um método de posicionamento por intersecção circular ou hiperbólica, respectivamente, conforme a tecnologia empregada no radar. O método hiperbólico requer pelo menos duas linhas de base.

O radar MRS-Mini Ranger System Motorola utiliza o posicionamento circular, uma vez que é capaz de fornecer diretamente as distâncias do ponto às estações da terra (extremos da linha de base). Nesse caso, a determinação da posição consiste no cálculo da intersecção de duas circunferências, cujos raios são as respectivas distâncias à posição. O usuário pode se valer de qualquer sistema cartesiano de coordenadas para essa determinação, porém o mais usado é o sistema UTM (Universal Transversal Mercator). A figura abaixo é um caso típico da triangulação de posicionamento com o MRS.

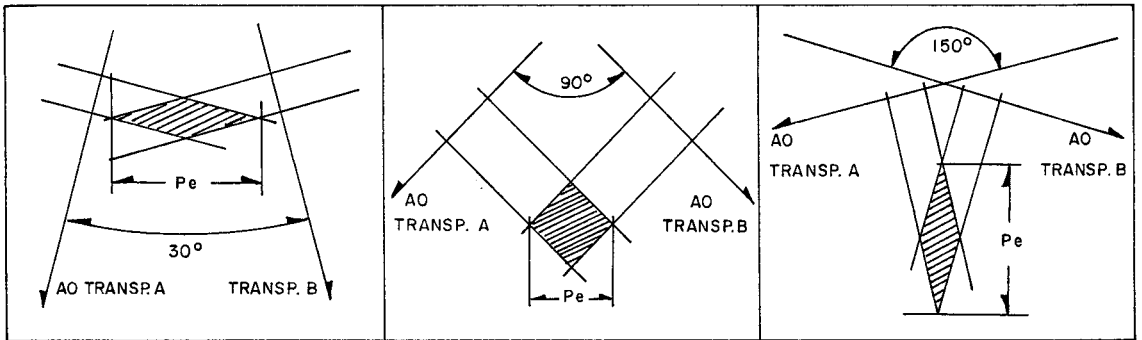


A propagação dos sinais no sistema MRS, por ser feita em frequências de radar, sofre interferência das condições atmosféricas locais. Os efeitos de atenuação, refração e reflexão do sinal são formas dessas interferências. O alcance máximo do sistema da ordem de 37000m, pode ser atingido em condições atmosféricas ideais e na ausência de obstáculos entre as antenas de Bordo e do Transponder. O erro cometido na medição pelo MRS é absoluto, isto é, não depende da distância medida. Esse erro ( $e$ ) é inerente ao processo de medição, e segundo o fabricante, é da ordem de  $\pm 3m$ .

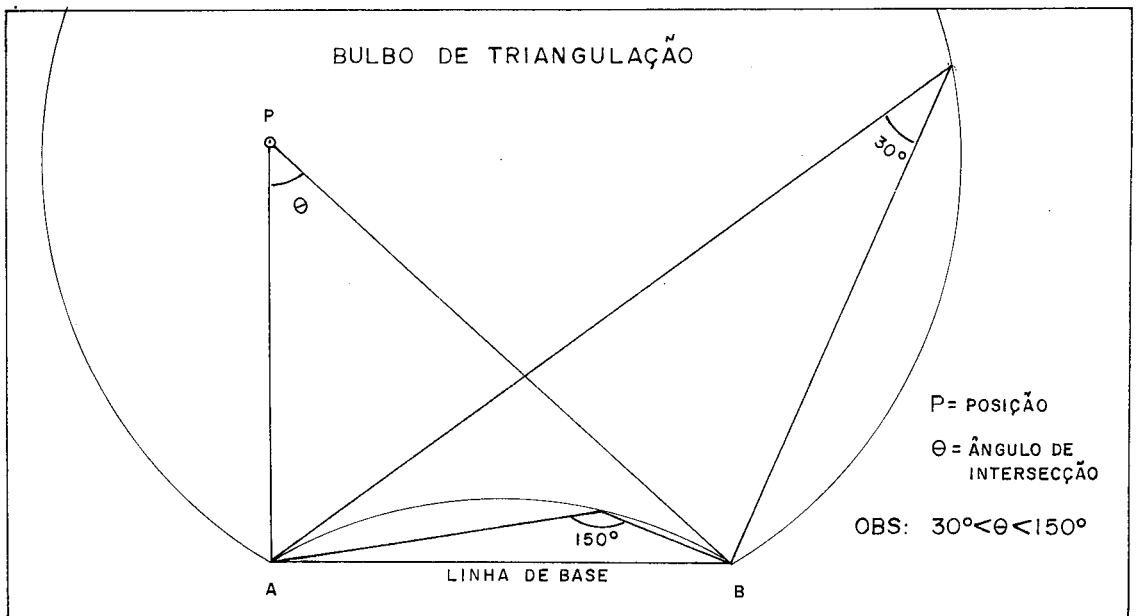
Em virtude do erro ( $e$ ), surge uma imprecisão no posicionamento da embarcação pelo MRS. Essa imprecisão corresponde à área do paralelogramo hachurada na figura que se segue, área na qual a unidade móvel pode de fato se encontrar.



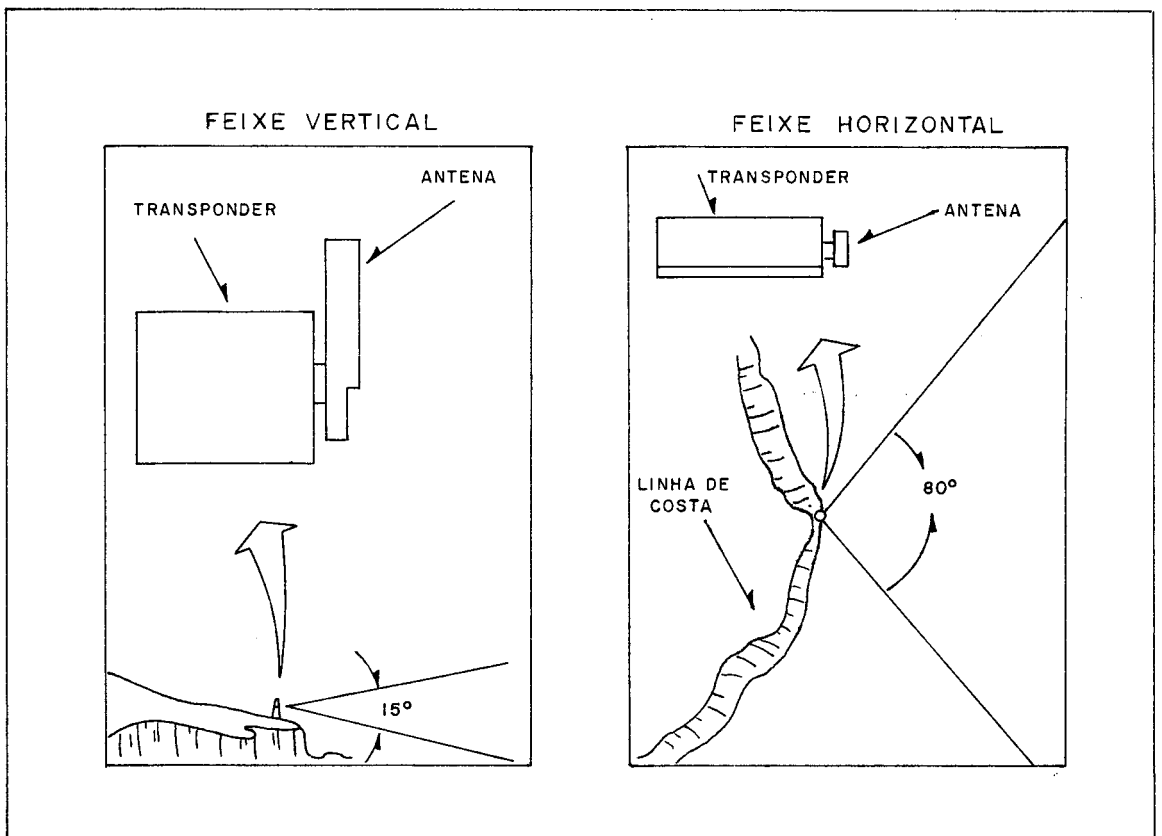
O erro de posição ( $P_e$ ) é máximo ao longo das diagonais do paralelogramo. O erro é variável em função do ângulo de intersecção dos arcos de circunferência. As diagonais crescem à medida que esses ângulos se afastam do ângulo reto ( $90^\circ$ ), sendo os ângulos de  $30^\circ$  e  $150^\circ$  limitações práticas para essas intersecções, conforme estabelece a triangulação clássica.



O traçado, a partir da linha de base, dos arcos dos segmentos capazes dos ângulos de  $30^\circ$  e  $150^\circ$  graus, permite delimitar a zona de bom posicionamento (Bulbo de Triangulação). Esta zona deve envolver toda a área a ser posicionada.



A natureza direcional da propagação dos sinais de radar requer cuidados especiais na instalação do MRS, no que concerne à escolha dos locais e posições das antenas. A instalação dos Transponders deve ser feita em locais livres de obstáculos, e com direcionamento adequado em suas antenas. O direcionamento da antena deve ser tal que permita ao transponder visualizar toda a área a ser posicionada, com seus feixes vertical ( $15^\circ$ ) e horizontal ( $80^\circ$ ) de recepção/transmissão de sinais. A instalação da estação de bordo (R/T) não apresenta esses problemas por possuir antena omnidirecional, sendo apenas recomendada a sua colocação no topo do mastro da embarcação, no intuito de facilitar o seu alcance (livre de obstáculos) e evitar interferências de reflexões provenientes da própria estrutura da embarcação.



## **2.6 - POSICIONAMENTO COM O SISTEMA SPE**

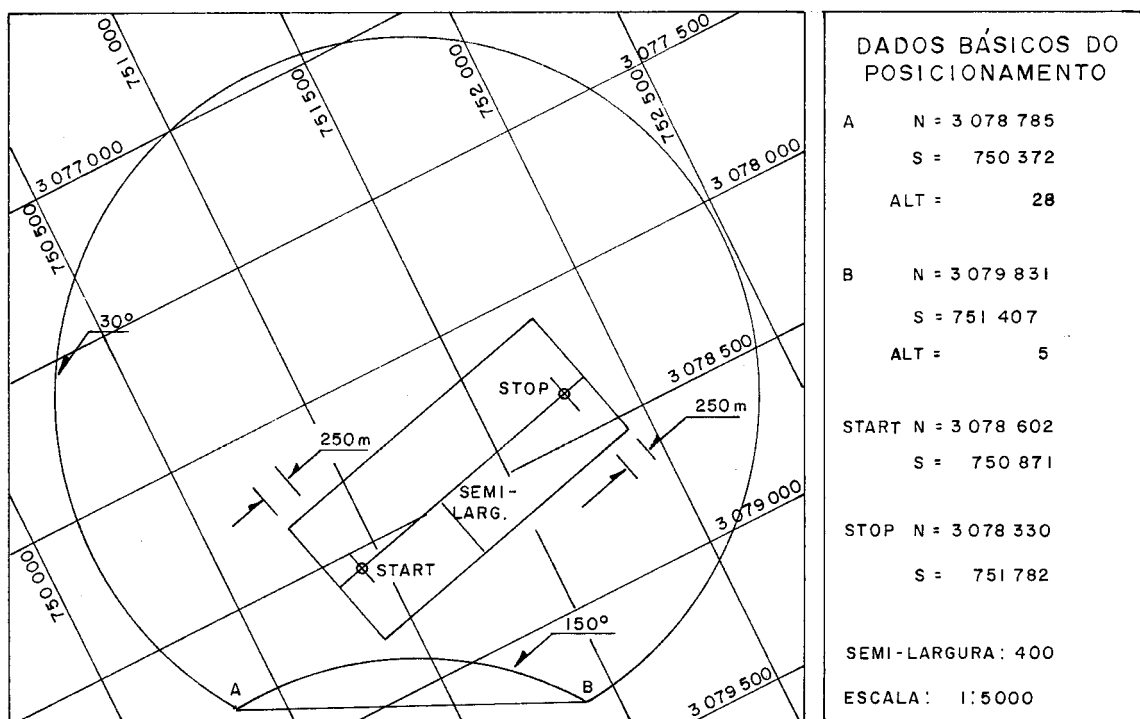
O sistema SPE registra, de modo praticamente contínuo, a posição da draga na carta do plotador. O registro dessa trajetória de deslocamento permite sua aferição em relação aos pontos da obra que se deseje atingir com o equipamento. Para isso, o sistema deve ser informado sobre qual região da obra se fará o posicionamento - CANAL DE POSICIONAMENTO - e ter essa região devidamente caracterizada na carta do plotador antes do início da operação - PRÉ-PLOTAGEM.

O Canal de Posicionamento é uma região de forma retangular, definida a partir de um eixo (de posicionamento) formado pelos pontos, inicial e final, de coordenadas conhecidas, da área a ser posicionada. O comprimento desse canal é igual à distância entre os extremos do eixo de posicionamento acrescida de 250m para cada lado. Sua largura é estabelecida pelo usuário, de modo a cobrir lateralmente, com as margens, toda a área objeto do posicionamento.

A pré-plotagem consiste na caracterização por meio de desenho, em escala adequada, na carta do plotador, dos pontos extremos do eixo de posicionamento e dos detalhes geométricos da obra. Os pontos extremos do eixo de posicionamento são marcados ao longo do eixo X da carta, e de preferência no centro da mesma. A escala selecionada deve ser tal que toda a área lateral do canal de posicionamento possa estar contida dentro da área útil da carta. Os detalhes geométricos da obra estabelecem trilhas e offset a serem aferidos. Tais detalhes formados por pontos imaginários ou bóias, delimitam: canais de acesso, bacias de evolução, jazidas, alto-fundos perigosos, curvas isobatimétricas, etc.

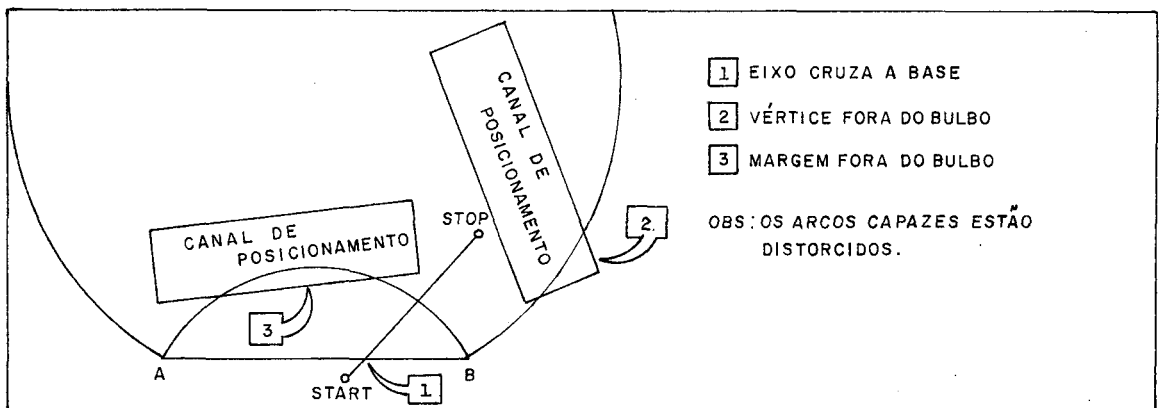
O conjunto DADOS BÁSICOS DO POSICIONAMENTO compreende todos os dados de que o sistema necessita para operar. Tais dados definem, a partir do canal de posicionamento, a geometria da triangulação que vamos executar, e o fator de escala adequada que se empregará em sua plotagem. Esses dados compreendem: coordenadas, alturas, faixa de posicionamento e escala. As coordenadas, expressas em metros, num sistema cartesiano arbitrado pelo usuário, são as coordenadas X/Y dos pontos inicial (START) e final (STOP) do eixo de posicionamento. As alturas são as das antenas dos transponders e da R/T de bordo, também em metros. A faixa (semi-largura do canal) define a área a ser coberta pelo posicionamento de ambos os lados do seu eixo. A escala é definida pelo seu denominador: 1/1000, 1/2000...

Os dados devem satisfazer inicialmente as seguintes condições geométricas, tendo em vista certas limitações do sistema de ordem aritmética: linha de base compreendida entre ... 1024m e 16383m, comprimento do eixo de posicionamento não inferior a 511m, range das distâncias entre 360m e 32767m. A figura que se segue ilustra os conceitos até aqui apresentados.

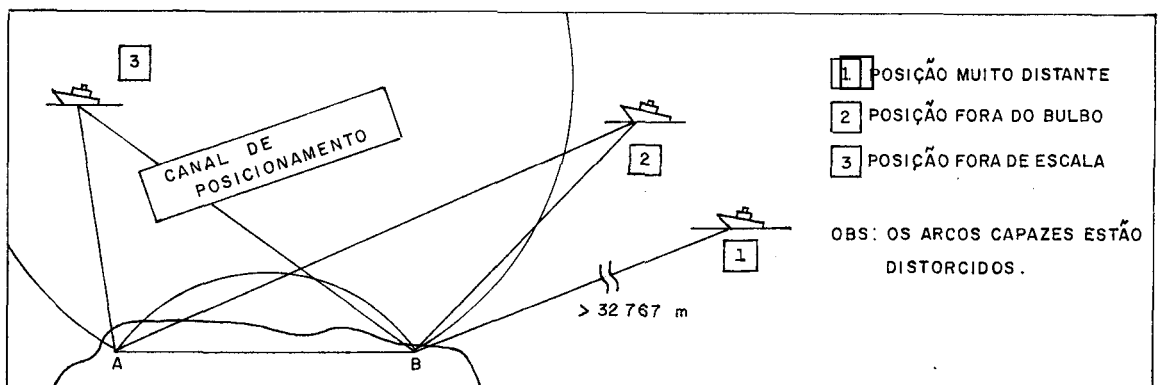




A geometria da triangulação, configurada pelos dados básicos do posicionamento, é verificada pelo sistema no sentido de se garantir que todo o canal de posicionamento esteja dentro do Bulbo de Triangulação. O sistema detecta três situações de erro em sua seqüência lógica de testes. A primeira situação é o cruzar da linha de base pelo eixo de posicionamento. A segunda é a pertinência de um dos vértices do canal de posicionamento (do retângulo) ao Bulbo. A última é a secância de uma das margens do canal ao arco dos ângulos de 150° do Bulbo.



Também durante a operação de posicionamento da draga o sistema detecta situações de erro. Tais situações em número de três provêm de uma seqüência lógica de testes. A primeira diz respeito às distâncias da draga às estações de terra, que devem estar dentro dos limites das distâncias aceitas pelo sistema. A segunda é a constatação da posição da draga fora do Bulbo de Triangulação. A última é a determinação de uma posição dentro do Bulbo que não pode ser plotada em face da escala adotada.



## 2.7 - OPERAÇÃO DO SISTEMA SPE

A operação do sistema SPE se faz por estágios bem definidos. Cada estágio é uma situação bem característica em que o sistema se encontra. O estágio possui mensagens e procedimentos próprios de operação que o usuário deve conhecer: PARTIDA DO SISTEMA, INICIALIZAÇÃO DO SISTEMA e TEMPO REAL. A evolução do sistema entre estágios se faz na ordem lógica e natural - PARTIDA, INICIALIZAÇÃO e TEMPO REAL. O estágio de PARTIDA só é executado uma vez, salvo o sistema tenha sido desligado. O estágio de INICIALIZAÇÃO só pode ser atingido mediante interrupção externa solicitada pelo operador. O estágio de TEMPO REAL é sempre uma consequência da conclusão do estágio de INICIALIZAÇÃO.

Os recursos disponíveis para a operação do sistema são os do seu PAINEL DE CONTROLE, já descritos anteriormente - display, teclado, botão liga/desliga e botão de interrupção. No display o sistema apresentará todos os informes que o operador precisa ter: mensagens de advertências/erros, perguntas e ecos de dados introduzidos. Pelo teclado o usuário faz registrar no sistema os dados que queira introduzir ou alterar. Os botões permitem ligar, desligar ou interromper o sistema em qualquer estágio da operação.

### ESTÁGIO DE PARTIDA

O ativar do botão liga/desliga coloca o sistema no ESTÁGIO DE PARTIDA. O controlador do SPE responde instantaneamente que está pronto para operar, exibindo, no painel de controle, a mensagem 'SISTEMA NO AR!'. A não apresentação dessa mensagem significa que a partida do sistema não se completou.

Uma vez pronto, o sistema pode passar ao estágio seguinte, de INICIALIZAÇÃO, em prosseguimento ao curso da operação, mediante o pressionar do BOTÃO DE INTERRUPÇÃO pelo operador.

### ESTÁGIO DE INICIALIZAÇÃO

O estágio de INICIALIZAÇÃO introduz e critica os dados básicos do posicionamento. A feição característica inicial do sistema neste estágio é a exibição, no painel, da mensagem correspondente à primeira pergunta da introdução de dados - 'XA ?'. O operador deverá responder a essa pergunta e a mais outras doze, a serem apresentadas a ele pelo sistema, passo a passo, com o dado correspondente. Respondida a última dessa série de treze perguntas, o sistema efetua a crítica dos dados introduzidos. O resultado da crítica determina a atitude a ser tomada pelo operador em continuação ao procedimento de operação.

Se houver inconsistência nos dados, uma mensagem de erro adequada, entre seis tipos diferentes, é apresentada no display pelo sistema. O sistema permanece no estado de espera (HALT), aguardando a intervenção do operador no sentido de recuperá-lo. A forma dessa recuperação é única, não dependendo do tipo de erro. O erro de consistência nos dados é inapelável para o sistema, obrigando o operador a retornar ao início do estágio, para reintroduzir os dados. Isto pode ser feito por meio do BOTÃO DE INTERRUPÇÃO.

Não havendo erro, a mensagem 'PLOTTER ORIGEM!' é apresentada pelo sistema. Concluindo a seqüência de operação do estágio, o operador deve colocar o plotador na origem, isto é, com a pena no ponto START do CANAL de POSICIONAMENTO. A passagem do sistema ao estágio seguinte, de TEMPO REAL, é feita pressionando a tecla ENTRA. Para isso, o plotador deve estar ligado (READY).

As perguntas se apresentam no display do sistema na forma: texto e resposta corrente. O texto, a pergunta em si, é uma seqüência de 9 caracteres alfabéticos seguidos de '?'. A resposta corrente - uma seqüência de seis dígitos - é o valor da resposta naquele instante para o sistema. Cada resposta contém pelo menos, um valor inicial (DEFAULT) que o sistema assume para ela. Cabe ao operador, diante de uma pergunta, validar ou alterar seu valor corrente.

Para alterar uma resposta corrente, basta ao operador pressionar a tecla APAGA. Em decorrência, o sistema apaga no display a resposta corrente e substitui o ? da pergunta por =, avisando ao operador que as teclas numéricas estão liberadas para a DIGITAÇÃO do DADO. Efetuada a digitação, o operador deve validar a sua resposta. Para validar uma resposta dada, ou uma existente que não se queira alterar, basta ao operador pressionar a tecla ENTRA. O sistema responde apresentando a próxima pergunta, desde que não tenha sido esta a última.

A digitação do dado consiste no registro dos seus seis algarismos no teclado numérico. À medida que o operador registra um dígito, o sistema o apresenta no display. Se no decorrer da digitação, foi teclado algum algarismo erradamente, o operador deve parar a digitação, a fim de introduzir todos os dígitos novamente. Para isso basta pressionar a tecla APAGA removendo os dígitos introduzidos até então. O sistema não valida respostas cujos dígitos (6) não foram completamente teclados.

As treze perguntas exibidas pelo sistema, seus valores DEFAULTS e seus significados, são apresentados no quadro abaixo em sua seqüência natural.

XA	?	000 000	coordenada X do Transponder A.
YA	?	000 000	coordenada Y do Transponder A.
XB	?	000 000	coordenada X do Transponder B.
YB	?	000 000	coordenada Y do Transponder B.
XSTRT	?	000 000	coordenada X do START do canal.
YSTRT	?	000 000	coordenada Y do START do canal.
XSTOP	?	000 000	coordenada X do ponto STOP.
YSTOP	?	000 000	coordenada Y do ponto STOP.
FAIXA	?	000 250	semi-largura do canal de posicionamento.
ALT R/T	?	000 030	altura da R/T.
ALT A	?	000 000	altura do Transponder A.
ALT B	?	000 000	altura do Transponder B.
ESCALA	?	000 002	unidades de milhar do denominador da escala do plotador.

As seis mensagens de erro que a crítica dos dados fornece, são apresentadas e comentadas abaixo:

**COORDN INVALIDA!**

as coordenadas configuram distâncias superiores a ..... 32767m, ou qualquer um dos demais dados excede esse valor;

**BASE FORA LIMIT!**

o comprimento da linha de base foge aos limites estabelecidos para o sistema;

**CANAL FORA TRG!**

um dos vértices do CANAL de posicionamento não pertence ao bulbo de triangulação;

**CANAL CRUZA BL!**

o eixo de posicionamento cruza a linha de base;

**MARGEM FORA TRG!**

a margem do canal de posicionamento é secante à circunferência do segmento capaz do ângulo de 150°;

**CANAL PEQUENO!**

o comprimento do eixo START-STOP do canal de posicionamento é inferior ao limite estabelecido.

## ESTÁGIO DE TEMPO REAL

O estágio de Tempo Real é, para o operador do sistema SPE, um estágio de observação. O operador intervém apenas quando o sistema solicitar recuperação ou atenção, em caso de erro. O caracter de prontidão - asterisco (\*) - é exibido sempre na posição primeira do display, seguido de brancos, para indicar que o sistema opera em Tempo Real sem erro pendente. A detecção de qualquer condição anormal no sistema, faz com que se ja apresentada no display, ao lado do caracter de prontidão, mensagem de erro correspondente. O sistema volta automaticamente a exibir apenas o caracter de 'prontidão' (\*) assim que percebe a recuperação do erro. A recuperação pode ser reconhecida, uma vez que ele nunca pára de executar seu ciclo de Tempo Real. Durante o ciclo o operador pode interromper o funcionamento do sistema a qualquer instante pelo Botão de Interrupção. O sistema responde evoluindo para o estágio de INICIALIZAÇÃO.

As mensagens de erro apresentadas pelo sistema e seu significado são mostrados abaixo:

XPONDERS FORA!

indica que ambos os transponders (A e B) não estão respondendo;

FORA XPNDR A! (ou B!)

indica que um dos transponders (A ou B) está fora;

LEITURA MRS!

leitura MRS apresentou dígito BCD inválido;

PLOTTER FORA!

indica que o plotador está desligado (NOT-READY);

RANGE INVALIDO!

a distância medida em um dos canais do MRS está fora dos limites estabelecidos;

**POSICIONAMENTO!**

indica que a posição se encontra fora do Bulbo de Triangulação;

**FORA DE ESCALA!**

a posição não pode ser plotada por estar fora da área útil da carta do plotador.

A recuperação do sistema, nos casos de erro, só requer a intervenção do operador para aqueles erros relacionados com o mau funcionamento de seus periféricos. São eles: 'XPONDERS FORA!', 'FORA XPNDR A ou B!' e 'LEITURA MRS'. Tal intervenção se faz sempre junto ao periférico. As demais mensagens de erro devem apenas despertar no operador uma maior atenção do ponto de vista náutico.

## CAPÍTULO III

### DETALHAMENTO DO SOFTWARE

#### 3.1 - ESTRUTURA DO SOFTWARE

O software do sistema SPE, escrito em linguagem ASSEMBLER 8080 INTEL, compreende: programa principal e pacotes de subrotinas. O programa principal está sub-dividido em cinco (5) módulos (programas independentes). As subrotinas, num total de vinte e quatro (24), permitem trinta e quatro (34) pontos de entrada, e estão agrupadas em dois pacotes (PACKAGES). Os módulos recorrem-se das subrotinas no decorrer do processamento. O programa principal possui duas tabelas de referências externas de pontos de entrada, uma para cada pacote. As tabelas promovem a ligação (LINK) entre os módulos e as subrotinas, em tempo de chamada.

#### ESQUEMA DE APRESENTAÇÃO

\* PACOTE DE SUBROTINAS

PACOTE IOPACK

PACOTE MTPACK

\* PROGRAMA PRINCIPAL

MÓDULO DE INTERRUPÇÃO

MÓDULO DE PARTIDA

MÓDULO DE DADOS

MÓDULO DE PREPARO

MÓDULO DE TEMPO REAL



### 3.2 - PACOTES DE SUBROTINAS

As subrotinas do sistema SPE estão grupadas conforme a finalidade, em dois pacotes de subrotinas: IOPACK e MTPACK. Esses pacotes de subrotinas atendem, respectivamente, às necessidades de ENTRADA/SAÍDA e de CÁLCULOS do sistema. Cada pacote em si é auto-suficiente, pois, nenhuma de suas rotinas chama qualquer outra externa a ele.

A comunicação, isto é, a passagem de parâmetros entre os módulos do programa principal e as subrotinas dos pacotes, é feita via registradores (simples ou duplos) do 8080 ou via variáveis definidas na memória. A chamada das subrotinas é feita pela instrução CALL.

Cada pacote constitui uma unidade independente para o compilador ASSEMBLER. Apenas o IOPACK se utiliza de referências externas, que por sua vez já estão resolvidas.

Nas descrições que se seguem, as variáveis serão referenciadas por seus nomes simbólicos, estabelecidos na programação. Quanto aos registradores, serão mencionados pelos respectivos símbolos, precedidos da letra X (XA é o registrador A).

P A C O T E I O P A C K

O pacote de subrotinas IOPACK reúne todas as subrotinas que, direta ou indiretamente, executam funções de entrada e saída no sistema SPE. Elas provêem o "serviço" dos periféricos (PLOTADOR, RADAR, DISPLAY e TECLADO) ou executam tarefas mais complexas de E/S. Essas últimas recorrem-se das anteriores. O pacote possui quatorze (14) pontos de entrada.

Tais rotinas são:

SUBROTINA TTY  
 SUBROTINA PLTR  
 SUBROTINA MRS  
 SUBROTINA KBIN  
 SUBROTINA DPQT  
 SUBROTINA DSPLY  
 SUBROTINA MENSA  
 SUBROTINA PLOTA

SUBROTINA TTY

## DESCRICHÃO:

É a rotina de serviço do TELETYPE (TTY). Possui três (3) pontos de entrada:

TYIN (Teletype input)

Lê um caracter no teclado do Teletype.

TYOUT (Teletype output)

Imprime um caracter no Teletype sem efetuar controle do carro.

TYCLR (Teletype clear)

Inicializa nova linha no Teletype, fazendo CARRIAGE RETURN (CR) e saltando de linha (LF).

PARÂMETROS: (só para TYOUT)

XC contém caracter a ser impresso.

RETORNO: (só para TYIN)  
XA contém caracter lido no teclado.

SUBROTINAS CHAMADAS:  
Nenhuma.

ALGORITMO:  
Elementar.

### SUBROTINA PLTTR

#### DESCRIÇÃO:

É a rotina de serviço do plotador HOUSTON. Testa status ou comanda a execução de um step pelo plotador, conforme parâmetro recebido. O status indica se o plotador está ligado (READY). O step compreende a execução, pelo plotador, de um deslocamento de 0,1mm em qualquer uma das suas oito direções fundamentais de movimento. Durante a execução do step, a rotina faz o processador entrar em delay até a sua conclusão, quando, então, retorna ao ponto de chamada.

#### PARÂMETROS:

Comanda um step no plotador para  $XA \neq 0$ . Caso contrário, testa status.

#### RETORNO:

Em caso de teste de status, ( $XA=0$ ) retorna com CARRY=1 para indicar plotter desligado (NOT READY).

SUBROTINAS CHAMADAS:  
Nenhuma.

## ALGORITMO:

Elementar.

## OBSERVAÇÕES:

A rotina não leva em consideração o status do plotador ao comandar um STEP. O padrão de bits do parâmetro que seleciona o step é o seguinte:

01 = X+	10 = PENA ABAIXA
02 = X-	20 = PENA SOBE
04 = Y+	
08 = Y-	

SUBROTINA MRS

## DESCRIÇÃO:

É a rotina de serviço do radar MRS-Motorola. Destina-se ao controle e aquisição de dados pelo radar. Contém três pontos de entrada (ENTRY-POINT) para finalidades específicas:

## MRSGO

Coloca em operação o MRS, fazendo com que ele comece a interrogar as estações de terra ininterruptamente, e retorna ao ponto de chamada.

## MRSRD

Esse ponto de entrada somente é atingido via rotina de serviço da interrupção. Lê e testa a validade de cada um dos cinco dígitos decimais da medição apresentada pelo console, mediante solicitação de interrupção ao sistema. Os dígitos vêm codificados em BCD, (binary coded decimal), e, se válidos, são armazenados na área de "leituras do MRS". O teste de validade decimal feito, dígito a dígito,

pelo procedimento BCDCK, faz retornar a condição de erro ou não-erro, no CARRY. A detecção de qualquer dígito errado faz a rotina abortar o dado. O retorno ao programa principal (Módulo de Tempo Real) é feito mediante desvio (JMP) para pontos adequados desse programa, conforme tenha ou não havido erro na leitura (WRNRD ou DATA).

#### MRSOF

Faz cessar imediatamente o interrogar das estações de terra pelo console MRS. Desabilita a interrupção na interface (8255≠1). Desliga a CHAVE DE LEITURA do MRS. Restaura o BIT DE STROBE corrente, por meio de uma leitura sem efeito (DUMMY), para prevenir interrupção pendente. Retorna ao ponto de chamada.

PARÂMETROS: (só para MRSRD)  
XHL contém o endereço da área de leitura para recepção do dado.

RETORNO: (só para MRSRD)  
XA contém o código do canal lido.

SUBROTINAS CHAMADAS:  
Nenhuma.

ALGORITMO:  
Elementar.

OBSERVAÇÕES:  
A chamada da rotina pelo ponto de entrada MRSRD é feito, excepcionalmente, pela instrução JMP.

SUBROTINA DPOUT

## DESCRIÇÃO:

É a rotina de serviço do display SELF-SCAN BURROUGHS. Possui dois pontos de entrada:

DPOUT (display output)

Exibe um caracter no display, considerando a posição corrente apontada pelo seu cursor interno.

CLSSC (display clear)

Executa um limpa-display e faz o cursor apontar para a primeira posição física do display (Home).

PARÂMETROS: (só para DPOUT)

XC contém o caracter a ser exibido, codificado em ASCII.

## RETORNO:

Nenhum.

## SUBROTINAS CHAMADAS:

Nenhuma.

## ALGORITMO:

Elementar.

## OBSERVAÇÕES:

A rotina trabalha internamente com o código ASCII-64 (Sub-Set) compatível com o display.

SUBROTINA DSPLY

## DESCRIÇÃO:

Exibe cadeia de caracteres codificados em ASCII no display BURROUGHS. Possui dois pontos de entrada:

**DSPLY**

Exibe a cadeia de caracteres a partir da próxima posição disponível do display, indicada pelo seu cursor circular.

**PRINT**

Após executar um limpa-display (HOME), exibe a cadeia de caracteres a partir da primeira posição física.

**PARÂMETROS:**

XHL contém o endereço da cadeia de caracteres da memória a ser exibida e XB o número de caracteres do seu comprimento (máximo = 16).

**RETORNO:**

XHL contém o endereço final + 1 da cadeia de caracteres que foi exibida.

**SUBROTINAS CHAMADAS:**

DPOUT (DPOUT, CLSSC).

**ALGORITMO:**

Elementar.

**OBSERVAÇÃO:**

A rotina também opera com o TTY, desde que sejam feitas as seguintes substituições:

CALL DPOUT por CALL TYOUT

CALL CLSSC por CALL TYCLR

**SUBROTINA MENSA****DESCRIÇÃO:**

ESCREVE ou APAGA, no display, mensagem correspondente ao estado de erro ou não-erro do sistema,

durante a fase de Tempo Real. A nova mensagem é cotejada com a que está sendo exibida no display (mensagem corrente), antes da execução de qualquer uma das duas funções. A função APAGA só é executada para mensagens iguais; a função ESCREVE, para mensagens diferentes.

PARÂMETROS:

XA contém nova mensagem em código ESCREVE/APAGA.

RETORNO:

Nenhum.

SUBROTINAS CHAMADAS:

DSPLY (DSPLY, PRINT).

ALGORITMO:

P1 - Analisa se a condição é APAGA.  
 ENTÃO: Compara se mensagens iguais,  
 ENTÃO: APAGA mensagem.  
 SENÃO: Compara se mensagens diferentes,  
 ENTÃO: ESCREVE mensagem.

P2 - Termina.

OBSERVAÇÕES:

A rotina MENSA permite ao sistema a substituição do display por qualquer dispositivo de cópia dura (teletipo), uma vez que é capaz de evitar a contínua e desnecessária repetição de mensagens redundantes, concernentes ao estado de erro do sistema.



SUBROTINA KBIN

## DESCRIÇÃO:

É a rotina de serviço do teclado numérico. Lê um caracter do teclado e o codifica em ASCII. Provê o "debouncing" para o teclado.

## PARÂMETROS:

Nenhum.

## RETORNO:

XA contém o caracter lido no teclado, em código ASCII.

## SUBROTINAS CHAMADAS:

Nenhuma.

## ALGORITMO:

- P1 - Salva o contexto da máquina.
- P2 - Testa se teclado ativo.  
ENTÃO: Desvia para P4 (debouncing).
- P3 - Vã para P2 (aguarda toque).
- P4 - Provê o debouncing do toque.
- P5 - Codifica em ASCII o caracter lido.
- P6 - Restaura o contexto da máquina.
- P7 - Termina.

## SUBROTINAS CHAMADAS:

Nenhuma.

## OBSERVAÇÕES:

O teclado possui duas teclas de controle APAGA e ENTRA que recebem, na conversão, os seguintes códigos ASCII: 0D e 7F. Tais códigos foram escolhidos visando à compatibilidade com o teclado do Teletype (CR, LF).

**SUBROTINA PLOTA****DESCRIÇÃO:**

Plota uma posição definida por suas coordenadas PLOTX e PLOTY medidas em steps, calculando as projeções (X/Y) do deslocamento a partir das variáveis OLDX e OLDY de controle da posição corrente da pena.

**PARÂMETROS:**

As variáveis inteiras PLOTX e PLOTY do programa principal contêm as coordenadas da posição a ser plotada em steps, por ocasião da chamada da rotina.

**RETORNO:**

Atualiza as variáveis de controle (OLDX e OLDY) da posição corrente da pena, com a nova posição (PLOTX e PLOTY).

**SUBROTINAS CHAMADAS:**

PLTTR.

**ALGORITMO:**

P1 - Calcula a componente Y do vetor deslocamento, determinando seu módulo e sentido. Atualiza projeção Y da posição corrente da pena (OLDY).

P2 - Calcula a componente X do vetor deslocamento, determinando seu módulo e sentido. Atualiza projeção X da posição corrente da pena (OLDX).

P3 - Seleciona a direção fundamental de movimento (step) a ser inicialmente utilizada na plotagem.

P4 - Desenvolve traçado, step a step, por meio de algoritmo EXPEDITO e termina.

**OBSERVAÇÃO:**

Ver algoritmo EXPEDITO no capítulo IV.

P A C O T E M T P A C K

O pacote de rotinas MTPACK agrupa todas as rotinas que auxiliam ou participam do cálculo NUMÉRICO do posicionamento no sistema SPE. Participam executando operações aritméticas, operações binárias, comparações e cálculo de funções. Auxiliam fazendo a manipulação de operandos, isolados ou em listas de variáveis. O pacote opera com quantidades inteiras (32 bits) e fracionárias (16 bits por 16 bits). As operações tem por base um acumulador aritmético fictício (FAC). Variáveis auxiliares servem de rascunho para os cálculos. Um apontador ajuda na manipulação de listas de parâmetros. O acumulador, as variáveis auxiliares e o apontador têm as suas alocações de memória providas pelo pacote. Ele possui vinte (20) pontos de entrada e suas rotinas são:

SUBROTINA LDSTO  
SUBROTINA CR  
SUBROTINA ADSUB  
SUBROTINA MP  
SUBROTINA DV  
SUBROTINA DIFAC  
SUBROTINA SLFAC  
SUBROTINA SRFAC  
SUBROTINA CPFAC  
SUBROTINA ZRFAC  
SUBROTINA SQFAC  
SUBROTINA TRANS  
SUBROTINA SOLVE  
SUBROTINA MIN  
SUBROTINA DCBIN  
SUBROTINA GET

**SUBROTINA LDSTO****DESCRIÇÃO:**

É a rotina básica do pacote, pois permite a comunicação entre o usuário e a unidade aritmética fictícia. Executa as funções de carga (LOAD) e armazenamento (STORE) do acumulador (FAC). Possui dois pontos de entrada:

LD

carrega o FAC com o conteúdo do operando.

ST

armazena o conteúdo do FAC no operando.

**PARÂMETROS:** (para ambas as entradas)

XHL aponta para o operando.

**RETORNO:**

Nenhum.

**SUBROTINAS CHAMADAS:**

Nenhuma.

**ALGORITMO:**

Elementar.

**SUBROTINA CR****DESCRIÇÃO:**

Compara o conteúdo do acumulador com o do operando. O resultado da comparação aciona os FLIP-FLOPS de condição, do mesmo modo que o faz a instrução "CMP" do microprocessador.

**PARÂMETROS:**

XHL aponta para o operando.

## RETORNO:

Os flip-flops CARRY e ZERO indicam:

se (FAC) > (OPR) Z = 0 e C = 0

se (FAC) < (OPR) Z = 0 e C = 1

se (FAC) = (OPR) Z = 1 e C = 0

## SUBROTINAS CHAMADAS:

Nenhuma.

## ALGORITMO:

Compara byte a byte, da ordem mais alta dos campos para a mais baixa.

## OBSERVAÇÃO:

Não destrói o conteúdo do acumulador.

SUBROTINA ADSUB

## DESCRIÇÃO:

Soma ou subtrai o conteúdo do operando do conteúdo do acumulador. Deixa o resultado dessas operações no acumulador (FAC), antes de retornar ao ponto de chamada. Opera com números inteiros ou fracionários. Possui dois pontos de entrada:

AD

Efetua a adição.

SB

Efetua a subtração.

PARÂMETROS: (para ambas as entradas)

XHL aponta para o operando.

## RETORNO:

FAC contém a soma ou a diferença conforme a chamada.

## SUBROTINAS CHAMADAS:

Nenhuma.

## ALGORITMO:

Adiciona ou subtrai byte a byte, da ordem mais baixa para mais alta, utilizando CARRY ou BORROW respectivamente.

**SUBROTINA MP**

## DESCRIÇÃO:

Multiplica o conteúdo do acumulador pelo do operando. O acumulador e o operando devem ser números fracionários. O resultado da operação é um inteiro (32 bits) que é deixado no acumulador. A parte fracionária do resultado é desprezada.

## PARÂMETROS:

XHL aponta para o operando (multiplicando).

## RETORNO:

FAC contém o produto.

## SUBROTINAS CHAMADAS:

ZRFAC

## ALGORITMO:

P1 - Compara sinal do multiplicando com o do multiplicador para determinar o sinal do resultado.

P2 - Toma o módulo do multiplicador e separa sua parte inteira da fracionária.

- P3 - Toma o módulo do multiplicando e separa sua parte inteira da fracionária.
- P4 - Multiplica parte fracionária do multiplicador por parte inteira do multiplicando.
- P5 - Multiplica parte inteira do multiplicador por parte inteira do multiplicando.
- P6 - Multiplica parte inteira do multiplicador por parte fracionária do multiplicando.
- P7 - Acumula esses produtos parciais no acumulador.
- P8 - Atribui sinal ao resultado.

**OBSERVAÇÃO:**

Os produtos parciais P5, P6 e P7 são calculados com uso da rotina clássica: multiplicação a dois passos com operando de 2 bytes.

**SUBROTINA DV**

**DESCRIÇÃO:**

Divide inteiro contido no acumulador por fracionário contido no operando, através de operação inteira com 48 bits. O resultado é um quociente fracionário.

**PARÂMETROS:**

XHL aponta para o operando.

**RETORNO:**

FAC contém o quociente fracionário.

**SUBROTINAS CHAMADAS:**

LDSTO (ST, LD + 6), CPFAC (CPFAC, CPFAC + 3),  
CR (CR + 3), ZRFAC (ZRFAC + 3).

## ALGORITMO:

- P1 - Se divisão por zero, termina.
- P2 - Determina sinal do quociente.
- P3 - Aloca módulo do divisor na parte mais baixa da área DVSR (48 bits).
- P4 - Aloca módulo do dividendo na parte mais alta de DVDN2 (48 bits).
- P5 - Zera DVDN1 (48 bits) e os 16 bits mais baixos de DVDN2.
- P6 - Executa 65 vezes (algoritmo clássico da divisão):
  - 6.1 - Subtrai DVSR de DVDN1.
  - 6.2 - Se resultado positivo:
    - ENTÃO: guarda resultado em DVDN1, e seta bit a entrar em DVDN2.
    - SENÃO: Reseta bit a entrar em DVDN2.
  - 6.3 - Shifta DVDN1 e DVDN2 para esquerda.
- P7 - Atribui sinal ao resultado.
- P8 - Termina movendo a parte mais significativa para o FAC.

## OBSERVAÇÃO:

A execução do algoritmo clássico da divisão em looping de 65 vezes, em uma operação com 48 bits, permite a obtenção da parte fracionária.

SUBROTINA DIFAC

## DESCRIÇÃO:

Decrementa ou incrementa o conteúdo do acumulador. O valor desse incremento ou decremento é enviado como parâmetro.



Possui dois pontos de entrada:

DCFAC

decrementa.

INFAC

incrementa.

PARÂMETROS: (para ambas as entradas)

XA contém inteiro sem sinal a decrementar ou incrementar, variando entre 1 a 255.

RETORNO:

FAC incrementado ou decrementado conforme valor de XA.

SUBROTINAS CHAMADAS:

Nenhuma.

ALGORITMO:

Elementar.

### SUBROTINA SLFAC

DESCRIÇÃO:

Shifta o conteúdo do acumulador 1 bit para a esquerda, fazendo entrar um zero a direita. Não leva em consideração o bit de sinal do campo.

PARÂMETROS:

Nenhum.

RETORNO:

FAC shiftado 1 bit a esquerda.

SUBROTINAS CHAMADAS:

Nenhuma.

**ALGORITMO:**

Shifta, byte a byte, 1 bit a esquerda propagando o CARRY OUT BIT.

**SUBROTINA SRFAC****DESCRIÇÃO:**

Shifta o conteúdo do acumulador 1 bit para direita, fazendo entrar um zero a esquerda. Não leva em consideração o sinal do campo.

**PARÂMETROS:**

Nenhum.

**RETORNO:**

FAC shiftado 1 bit a direita.

**SUBROTINAS CHAMADAS:**

Nenhuma.

**ALGORITMO:**

Shifta, byte a byte, 1 bit a direita propagando o CARRY OUT BIT.

**SUBROTINA CPFAC****DESCRIÇÃO:**

Complementa a dois o FAC.

**PARÂMETROS:**

Nenhum.

**RETORNO:**

FAC com o sinal trocado.

## SUBROTINAS CHAMADAS:

Nenhuma.

## ALGORITMO:

Elementar.

**SUBROTINA ZRFAC**

## DESCRIÇÃO:

Zera o conteúdo do acumulador ou de qualquer outro campo. Esse campo é definido pelo seu endereço inicial de memória e pelo seu comprimento em bytes. Possui dois pontos de entrada:

ZRFAC

zera o acumulador.

ZERAR

zera campo de memória.

## PARÂMETROS:

(só para ZERAR)

XHL contém o endereço inicial do campo.

XC contém o contador do campo.

## RETORNO:

FAC zerado (ZRFAC).

## SUBROTINAS CHAMADAS:

Nenhuma.

## ALGORITMO:

Elementar.

**SUBROTINA SQFAC****DESCRIÇÃO:**

Extrai a raiz quadrada de um número inteiro no acumulador, obtendo, como resultado, um fracionário (no acumulador). Utiliza o algoritmo clássico raiz quadrada "por lápis e papel".

**PARÂMETROS:**

Nenhum.

**RETORNO:**

FAC contém a raiz em forma de fracionário.

**SUBROTINAS CHAMADAS:**

LDSTO (LD, ST), ADSUB (AD, SB)

SLFAC (SLFAC, SHFTL), CPFAC,

DIFAC (INFAC, DCFAC) ZRFAC.

**ALGORITMO:**

P1 - Cria radicando de 8 bytes. Abriga em sua parte baixa (4 bytes) o argumento dado (FAC). Zera a parte alta.

P2 - Admite, inicialmente, como raiz, o valor um.

P3 - Executa 32 vezes:

3.1 - Shifta dois bits à esquerda o radicando.

3.2 - Shifta um bit à esquerda a raiz e subtrai um. Atualiza valor da raiz.

3.3 - Repete operação 3.2 sem atualizar.

3.4 - Subtrai da parte alta do radicando o valor temporário obtido no item anterior.

3.5 - Se diferença positiva:

ENTÃO: Substitui parte alta do radicando pela diferença e soma um à raiz.

3.6 - Volta para 3.1.

P4 - Ajusta bit de mais baixa ordem da raiz.

P5 - Termina.

### SUBROTINA TRANS

#### DESCRIÇÃO:

Calcula o valor da função expressa por:

$$((A-B) * (C-D) + (E-F) * (G-H)) / I$$

onde: A, B, C, D, E, F, G, H e I são fornecidos por lista de parâmetros.

#### PARÂMETROS:

XHL aponta lista de parâmetros.

#### RETORNO:

FAC contém o valor da função na forma fracionária.

#### SUBROTINAS CHAMADAS:

LDSTO (ST), ADSUB (AD), DV

GET (GET, UPDAT), SOLVE.

#### ALGORITMO:

P1 - Calcula  $(A-B) * (C-D)$

P2 - Calcula  $(E-F) * (G-H)$

P3 - Soma as parcelas de P1 e P2.

P4 - Divide resultado por I.

P5 - Termina.

#### OBSERVAÇÃO:

Todos os parâmetros são fracionários.

SUBROTINA SOLVE

## DESCRIÇÃO:

Calcula o valor da função expressa por ...

$(A-B) * (C-D)$

onde: A, B, C e D são fornecidos por lista de parâmetros.

## PARÂMETROS:

XHL aponta lista de parâmetros.

## RETORNO:

FAC contém o valor da função na forma de inteiro.

## SUBROTINAS CHAMADAS:

LDSTO (LD, ST), ADSUB (SB), GET.

## ALGORITMO:

P1 - Calcula  $(A-B)$ .

P2 - Calcula  $(C-D)$ .

P3 - Multiplica as parcelas de P1 e P2.

P4 - Termina.

## OBSERVAÇÃO:

Todos os parâmetros são inteiros.

SUBROTINA MIN

## DESCRIÇÃO:

Calcula o valor mínimo entre quatro de uma lista de valores, enviados em lista de parâmetros. Destrói conteúdo do acumulador.

## PARÂMETROS:

XHL aponta lista de parâmetros.

## RETORNO:

FAC contém valor mínimo calculado.

## SUBROTINAS CHAMADAS:

LDSTO (LD, ST), CR, GET.

## ALGORITMO:

Opera com inteiros ou fracionários.

SUBROTINA DCBIN

## DESCRIÇÃO:

É a rotina auxiliar da conversão decimal-binário. Deve ser chamada pelo usuário em forma de "looping", considerando os n dígitos decimais do número a ser convertido. Destrói o conteúdo do acumulador.

## PARÂMETROS:

XHL aponta campo binário.

XA contém o dígito decimal.

## RETORNO:

$(OPR) = (OPR) * 10 + XA.$

## SUBROTINAS CHAMADAS:

LDSTO (LD, ST), SLFAC,

DIFAC (INFAC).

## ALGORITMO:

Efetua multiplicação por 10, utilizando soma e shifts.

**SUBROTINA GET****DESCRIÇÃO:**

Inicializa ponteiro de lista de parâmetros. Recupera parâmetro da lista, atualizando, automaticamente, o ponteiro. Possui dois pontos de entrada:

UPDAT

inicializa ponteiro de lista.

GET

recupera parâmetro da lista.

**PARÂMETROS:**

XHL aponta lista de parâmetros.

**RETORNO:**

Nenhum.

**SUBROTINAS CHAMADAS:**

Nenhuma.

**ALGORITMO:**

Elementar.



### 3.3 - PROGRAMA PRINCIPAL

O programa principal do sistema SPE constitui, para a compilação, uma unidade completa (deck único). É composto de cinco (5) módulos, sendo cada módulo uma rotina independente. Durante o funcionamento do sistema, os módulos se comunicam entre si ou diretamente, por meio de desvios incondicionais (JUMPS), ou indiretamente, por meio de interrupção. Recursos comuns aos módulos, supridos pelo programa, facilitam a integração entre eles. Esses recursos compreendem: tabelas de Referências Externas a subrotinas, Listas de Variáveis, Constantes e Alocações de Variáveis.

As tabelas de referências externas a subrotinas, uma para cada pacote, relacionam os endereços de alocações de memória dos pontos de entrada de todas as subrotinas utilizadas no sistema: 14 pontos de entrada no IOPACK e 20 no MTPACK.

As listas de variáveis, usadas, exclusivamente, por subrotinas do MTPACK, permitem a passagem de uma série de variáveis, de uma única vez, como parâmetros.

As constantes definem, entre outros, os seguintes valores: mensagens de PERGUNTAS; mensagens de ADVERTÊNCIAS/ERROS, da fase de consistência; códigos ESCREVE/APAGA de controle das mensagens de erro de Tempo Real; constantes numéricas inteiras e fracionárias.

A alocação de variáveis abrange, entre outras, variáveis básicas do posicionamento, variáveis estáticas do posicionamento, variáveis dinâmicas do posicionamento, variáveis de controle da plotagem, variáveis de controle e aquisição de dados, variáveis auxiliares da consistência geométrica.

Os módulos do programa executam, como já foi dito antes, funções específicas: de controle da interrupção, de partida do sistema, de aquisição de dados, de consistência de dados e de gerência do Tempo Real. Esses MÓDULOS são: DE INTERRUPTÃO, DE PARTIDA, DE DADOS, DE PREPARO e DE TEMPO REAL.

MÓDULO DE INTERRUPÇÃO (INTRP)

## DESCRIÇÃO:

Faz o tratamento da interrupção no sistema. Controla o fluxo de dados do MRS, prevenindo o OVERRUN. Analisa a procedência da interrupção (interna ou externa) e faz o seu tratamento.

## SUBROTINAS CHAMADAS:

MRS (MRSRD, MRSOF), CLSSC.

## ALGORITMO COMENTADO:

**P1 - Ajusta ponteiro da STACK.**

É desprezado o endereço de retorno armazenado pela interrupção no stack, tenha sido ela interna ou externa.

**P2 - Desliga chave de leitura MRS.**

Desabilita interrupção na interface MRS (8255 1). Desliga CHAVE DE LEITURA do MRS.

**P3 - Analisa procedência da interrupção, desviando se for interna.**

Lê o bit STROBE do 8255 1 e verifica se está ligado. Se o bit está ligado significa que chegou leitura (READ FLAG) do MRS. Nesse caso, desvia então para rotina MRSRD para ler e tratar o dado.

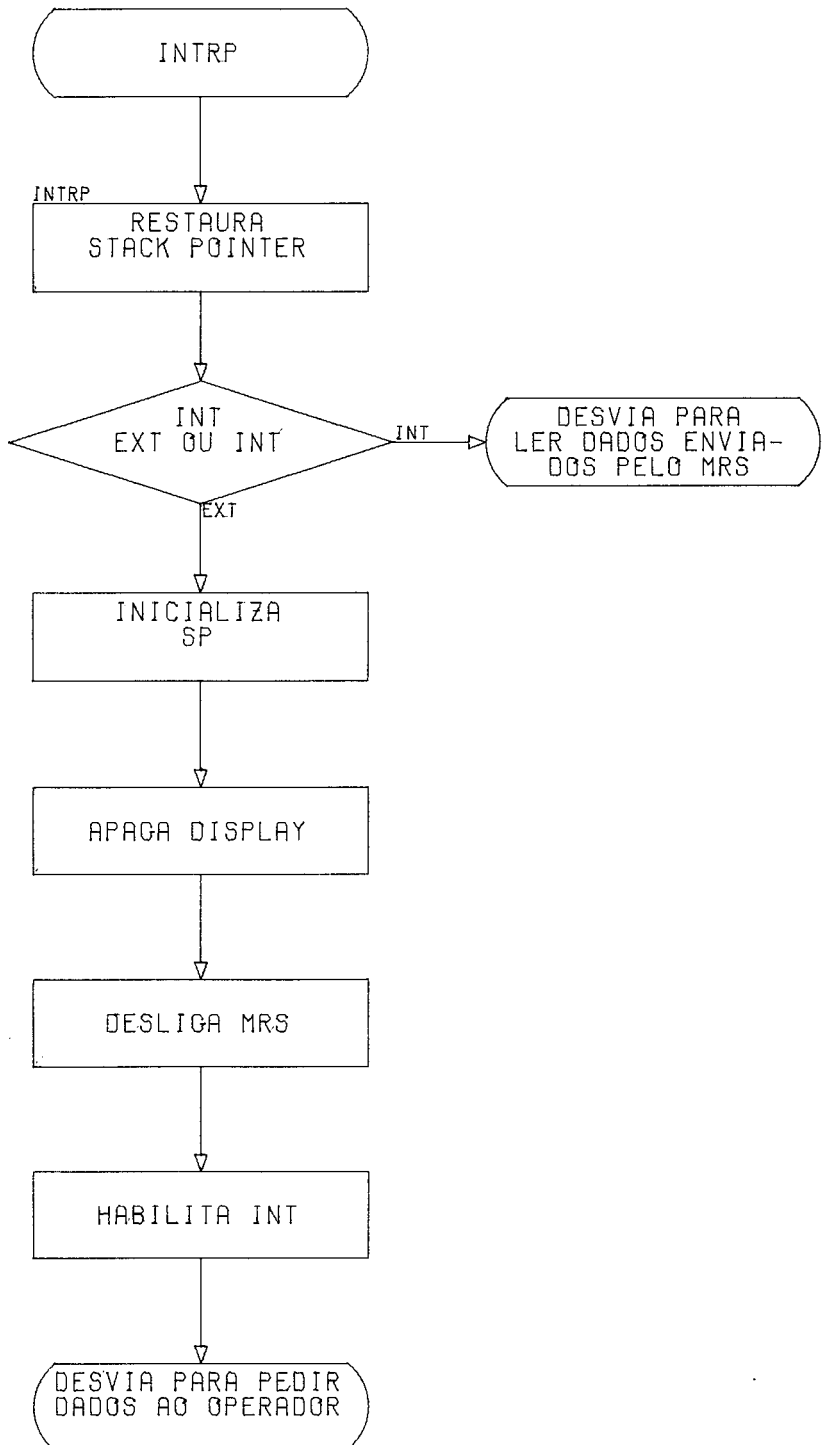
**P4 - Recupera o sistema em atendimento à interrupção externa.**

Restaura o topo da STACK ( $4000_{16}$ ). Limpa o display. Cessa a atividade de leitura do MRS.

**P5 - Desvia para re-iniciar a operação.**

Desvia para o módulo INCLZ.

DIAGRAMA DE BLOCOS  
MÓDULO DE INTERRUPÇÃO



MÓDULO DE PARTIDA (START)

## DESCRIÇÃO:

Inicializa o controlador tão logo este é ligado. A inicialização se dá em duas etapas. Na primeira etapa, o ligar do controlador gera um "reset" automático (POWER-ON/RESET) no microcomputador, abrangendo: registradores, linhas de dados, linhas de endereços e interfaces periféricas PPI. Em seguida, numa segunda etapa, por meio de software, uma série de preparações são executadas: alocação de stack, inicialização das interfaces de E/S, resete do MRS e inicialização de memória RAM.

## SUBROTINAS CHAMADAS:

MRS (MRSOF), ZRFAC (ZERAR), DSPLY.

## ALGORITMO COMENTADO:

**P1 - Inicializa microprocessador.**

Desabilita interrupção e define o topo da stack na posição  $4000_{16}$  da memória.

**P2 - Inicializa entrada e saída.**

Programa os PPI-8255, colocando em cada um deles a palavra de controle adequada:

8255 nº 1-controle = 8EH;

8255 nº 2-controle = B7H;

8255 nº 3-controle = 99H.

**P3 - Reseta o MRS.**

Chama subrotina MRSOF para resetar o MRS.

**P4 - Coloca valores DEFAULTS na área de respostas.**

Move zeros ( $30_{16}$ ) para os 66 bytes da área de resposta, utilizando a subrotina ZERAR e atribui valores DEFAULTS.

**P5 - Avisa operador.**

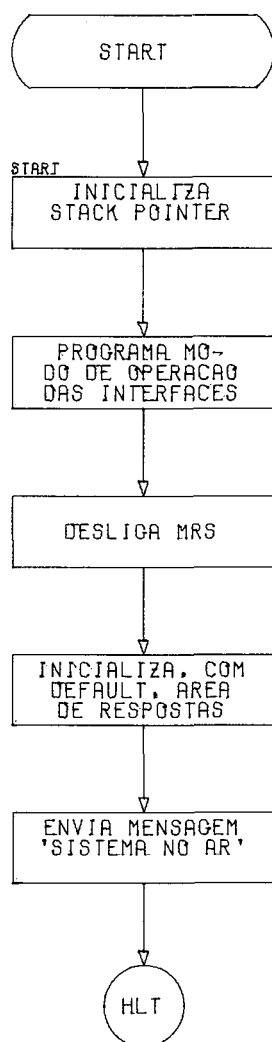
Envia, pelo display, mensagem ao operador (SISTEMA NO AR!), dando condição de pronto (READY) ao sistema.

**P6 - Habilita interrupção e entra em espera (HLT).**

Reabilita a interrupção, até então inibida, a fim de permitir ao operador tirar o microcomputador de HLT e dar início ao processamento do MÓDULO DADOS.

## DIAGRAMA DE BLOCOS

## MÓDULO DE PARTIDA



MÓDULO DE DADOS (DADOS)

## DESCRIÇÃO:

Tem por finalidade a aquisição dos dados básicos do posicionamento. Para isso, apresenta no display a seqüência de 13 perguntas aludidas anteriormente, às quais o usuário responde com o dado numérico correspondente, uma de cada vez. Os dados adquiridos são armazenados na memória em código ASCII, num campo denominado ÁREA DE RESPOSTAS, para posterior processamento pelo MÓDULO DE PREPARO. Todo esse procedimento de digitação pode ser sus- tado a qualquer instante, por meio de interrupção externa, fazendo a rotina voltar ao seu início, isto é, exibir a primeira pergunta da série. Mantêm-se válidos todos os dados já introduzidos até então.

## SUBROTINAS CHAMADAS:

DSPLY, KBIN, DPOUT, CLSSC, MOVE.

## ALGORITMO COMENTADO:

**P1 - Inicializa apontador de perguntas e de respostas.**

Os apontadores da área de perguntas (PTPER) e de respostas (PTRES) devem, ao início da rotina, estar apontando, nesses campos, respectivamente, a la pergunta e a la resposta.

**P2 - Exibe no display pergunta e resposta corrente apontadas.**

Move para buffer do display a pergunta apontada. Atualiza o apontador PTPER que passa a apontar a próxima pergunta. Move para buffer do display a resposta corrente (da la vez vale DEFAULT), concatenando-a com a pergunta. Exibe no display o conteúdo de seu buffer.

**P3 - Altera ou valida o valor numérico dessa resposta.**

Testa se é para VALIDAR ou ALTERAR a resposta corrente exibida. Em caso de ALTERAÇÃO, o buffer do display é modificado. Recebe um caracter '=' em lugar de '?', e a parte concernente ao valor numérico (resposta) é apagada com 'brancos'. Nessas condições o buffer é re-exibido no display, ficando o teclado liberado para o usuário digitar a nova resposta. À medida que o usuário vai teclando os dígitos, eles vão sendo apresentados no display, e, simultaneamente, introduzidos no buffer do display, em seu sub-campo RESPONSTA. Uma vez encerrada a digitação, segue para o passo seguinte, como se fosse VALIDAR esta resposta.

**P4 - Move resposta alterada ou validada para a área de respostas.**

A parte numérica do buffer do display é movida para área de respostas, sendo automaticamente, atualizado o apontador PTRES que passa a apontar a próxima resposta.

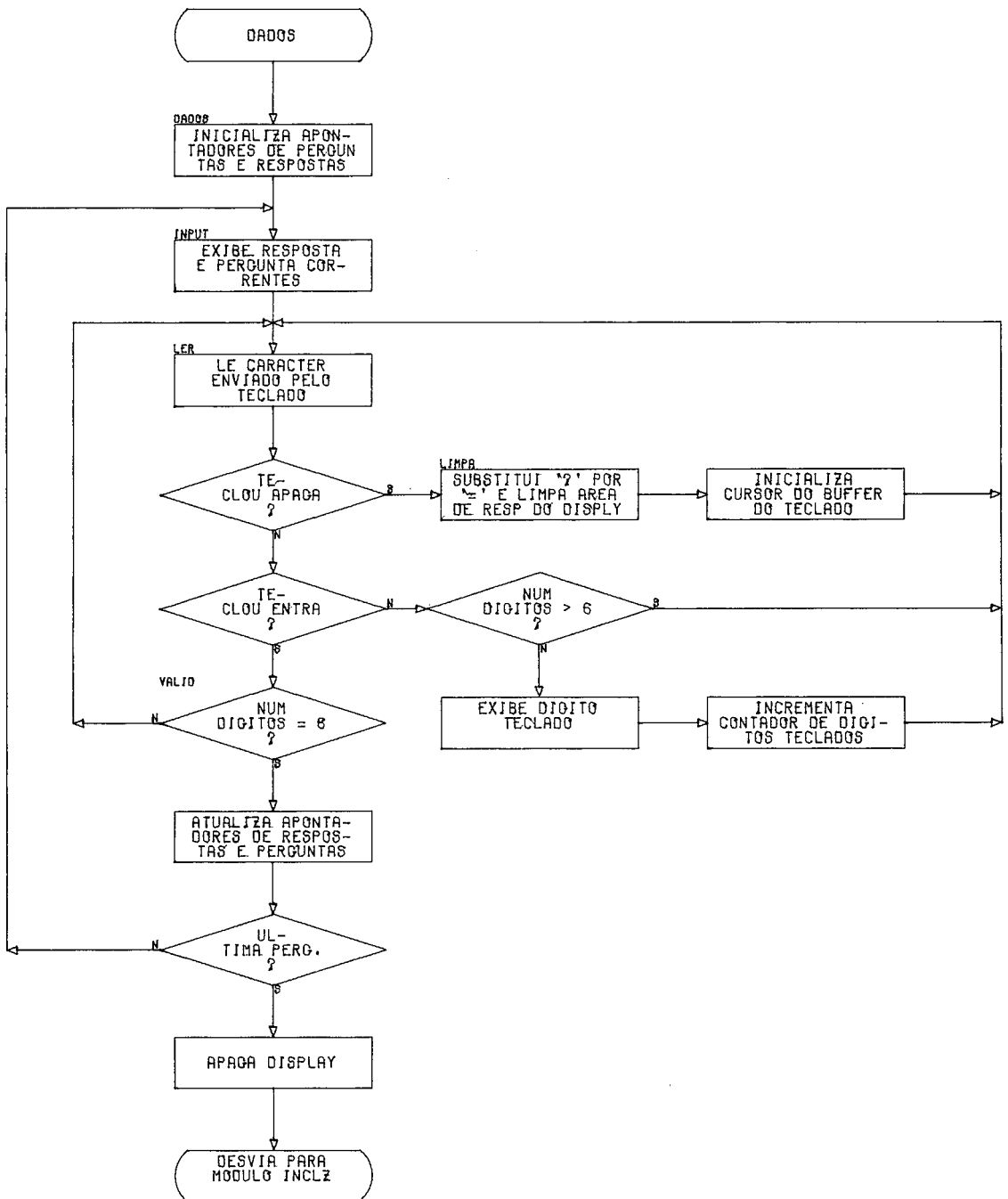
**P5 - Se não apresentou ainda as treze perguntas, volta para P2.**

Compara o endereço contido no apontador PTPER com o endereço final da área de perguntas. Se for menor, volta para P2 a fim de apresentar a próxima pergunta. Caso contrário, vai ao passo seguinte.

**P6 - Termina processamento do MÓDULO DADOS, desviando para o MÓDULO DE PREPARO.**

Apaga o display e desvia para INCLZ, ponto de entrada do módulo DE PREPARO.

DIAGRAMA DE BLOCOS  
 MODULO DE DADOS





MÓDULO DE PREPARO (INCLZ)

## DESCRIÇÃO:

É o módulo de crítica. Faz a consistência individual e conjunta dos dados básicos do posicionamento. Calcula, também, variáveis estáticas do posicionamento, a partir desses dados. Precedendo a consistência individual, o módulo efetua as reduções das coordenadas dos pontos A, B, START, STOP, de modo a minimizá-las em valor (positivo). Só então verifica se essas coordenadas reduzidas e os demais dados excedem a 32.767 (2 bytes). A consistência conjunta dos dados básicos, feita à medida que vão sendo calculadas as variáveis estáticas, compreende a verificação de comprimentos (linha de base e canal) e o teste de validade da geometria de triangulação, configurada por esses dados. Qualquer erro detectado nesta etapa faz o sistema enviar mensagem específica e abortar os dados.

Se os dados passarem pela crítica, o sistema prossegue calculando as demais variáveis estáticas do posicionamento e inicializando a plotagem, para entrar no MÓDULO DE TEMPO REAL.

## SUBROTINAS CHAMADAS:

LDSTO (LD, ST), ADSUB (AD, SUB), DV,  
 DIFAC (INFAC), SQFAC, CR, MP,  
 MIN, SLFAC, DCBIN, SRFAC,  
 TRANS, SOLVE, GET (UPDAT).

## ALGORITMO COMENTADO:

**Pl - Converte a binário os dados referentes às variáveis básicas do posicionamento.**

Inicializa apontador da área de resposta (RESPTS), fazendo-o apontar para a primeira

resposta. Prepara contador com o número de respostas (13) a serem convertidas. Executa ciclos de conversão regidos por esse contador, até varrer todo o campo das respostas. Cada conversão consiste em tomar os seis dígitos ASCII componentes de uma resposta, transformá-los em binário, e alocá-los na variável numérica que lhes corresponde, na ÁREA DE VARIÁVEIS BÁSICAS DO POSICIONAMENTO.

**P2 - Calcula parâmetros de redução para o sistema de coordenadas XY do usuário tal que minimize seus valores (positivos).**

As coordenadas dos pontos A, B, START e STOP são examinadas em cada um dos eixos (X, Y), independentemente, no sentido de se determinar o seu valor mínimo. Esses valores mínimos de cada eixo (XR e YR) são os parâmetros de redução do sistema. A obtenção de um mínimo é feita chamando-se a função MIN e fornecendo-se a ela uma lista de VARIÁVEIS, como parâmetros. Para a obtenção do mínimo de X (XR), a lista fornecida (XLIST) contém: XA, XB, XSTART, XSTOP. Para a obtenção do mínimo de Y (YR), a lista fornecida (YLIST) contém: YA, YB, YSTART e YSTOP.

**P3 - Efetua redução das coordenadas XY ao novo sistema, usando os parâmetros de redução. Simultaneamente, converte cada variável reduzida a fracionário, testando também a sua validade. Se variável reduzida > 32767, então:**  
prepara mensagem 'COORDN INVLDA!' e desvia para "PROCEDIMENTO DE ERRO".

As variáveis de cada um dos eixos, reunidas nas listas (XLIST e YLIST), são tratadas pelo procedimento REDUC em duas chamadas independentes. O procedimento REDUC, através da sub-rotina GET, recupera e trata cada variável da lista de parâmetros fornecida. O tratamento de uma variável consiste em dela subtrair o parâmetro de redução do seu eixo e, em seguida, usando shift à esquerda, convertê-la a fracionário. Durante a conversão, testa se a variável reduzida (já fracionário) é  $> 32767$ . Caso seja, abandona todo o processo de conversão em andamento e trata o erro.

**P4 - Converte a fracionário as demais variáveis básicas do posicionamento.**

Forma uma lista de parâmetros (VLIST) com as demais variáveis básicas do posicionamento (ZA, ZB, ALT, SLARG). Define um redutor de eixo, fictício, igual a zero ( $VR = 0$ ). Chama procedimento REDUC para fazer a conversão a fracionário dessas variáveis, uma vez que a redução não terá efeito.

**P5 - Calcula comprimento da linha de base da triangulação, testando-o quanto aos limites de validade. Se está dentro dos limites,**

então:

cria variáveis: quadrado e dobro da base, variáveis estáticas do cálculo do posicionamento;

senão:

prepara mensagem 'BASE FORA LIMIT!' e desvia para "PROCEDIMENTO DE ERRO".

O quadrado da linha de base (BASE 2) é determinado através de lista de parâmetros referente a esse cálculo (LIS01) e utilizando-se a função TRANS. Extraíndo-se a raiz quadrada desse valor, através da função SQFAC, obtêm-se o comprimento da linha de base (BASE). A base é testada quanto aos limites de validade (1024m e 16384m). Se está fora dos limites, então: prepara mensagem e trata o erro; se não, calcula dobro da base (BASED).

**P6 - Calcula comprimento do canal, testando-o quanto ao comprimento mínimo. Se não atende à condição, então:**

prepara mensagem 'CANAL PEQNO!' e desvia para "PROCEDIMENTO DE ERRO".

O comprimento do canal (LNGTH) é determinado através de lista de parâmetros referente a esse cálculo (LIS02), utilizando-se a função TRANS e extraíndo-se a raiz quadrada. Esse valor é testado quanto ao mínimo (512m). Se está fora do limite, prepara mensagem e trata o erro.

**P7 - Calcula coordenadas dos pontos extremos do canal, no sistema de coordenadas da linha de base.**

A partir das coordenadas XY reduzidas, determina as coordenadas dos extremos do canal (START e STOP), no sistema da linha de base (XLSTA, YLSTA, XLSTO, YLSTO). Tais coordenadas são calculadas usando a função TRANS com as listas de variáveis respectivas (LIS05, LIS06, LIS07, LIS08, LIS09, LIS10).

**P8 - Determina o lado da linha de base em que se dará a triangulação, isto é, à esquerda ou à direita dela.**

Testa o sinal da coordenada Y, do ponto START, calculada (YLSTR). Define a variável SIDE, lado da triangulação, em relação ao sentido AB da linha de base, como: à esquerda (0) para YLSTR negativo, e à direita (-1) para YLSTR positivo.

**P9 - Verifica se canal cruza a linha de base. Se cruza, então:**

prepara mensagem 'CANAL CRUZA BL!' e desvia para "PROCEDIMENTO DE ERRO".

Compara sinal da coordenada Y, do ponto STOP, calculada (YLSTO) com o lado da triangulação (SIDE). Se sinais diferentes, canal cruza a linha de base, então: trata o erro.

**P10- Determina parâmetros definidores (raio e centro) dos arcos de contigência da boa geometria do posicionamento.**

São calculados os raios e as coordenadas dos centros dos arcos de contigência da BOA GEOMETRIA (segmentos capazes de  $30^\circ$  a  $150^\circ$ ). No cálculo dos raios, é levada em consideração a FAIXA e a FOLGA de manobra do canal de posicionamento. O cálculo das coordenadas dos centros é feito no sistema da linha de base. Desse modo, a coordenada Y do centro do arco de  $30^\circ$  é sempre positiva, e a do de  $150^\circ$  é sempre negativa.

**P11- Testa a consistência global da geometria do posicionamento. Se inconsistente, então:**

prepara mensagem 'CANAL FORA TRG!' ou 'MARGEM FORA TRG!' conforme o caso, e desvia para "PROCEDIMENTO DE ERRO".

Calcula distâncias entre os centros dos arcos (30° e 150°) e os extremos do canal (START, STOP). Compara cada par de distâncias com o respectivo raio, para determinar se as extremidades do canal estão ou não fora do BULBO de TRIANGULAÇÃO. Se pelo menos uma está fora, então prepara mensagem 'CANAL FORA TRG' e desvia para tratar erro. De outra forma, prossegue, verificando se as margens do canal cortam o arco de 150°. Em caso afirmativo, prepara mensagem 'MARGEM FORA TRG' e desvia para tratar erro.

**P12- Calcula demais variáveis estáticas que participam do cálculo do posicionamento.**

São elas as variáveis utilizadas para o cálculo de correção por projeção horizontal do RANGE médio (AKA2 e AKB2) e para o cálculo da transformação de sistema de coordenadas: linha de base-plotador.

**P13- Calcula fator de escala e define limites X/Y para a plotagem.**

A partir da escala "por mil" fornecida pelo usuário, determina o fator de escala (metro-step). Em função desse fator, calcula o máximo e mínimo de X e Y (XMAX, XMIN, YMAX, YMIN) cobertos pela área útil do plotador.

**P14- Solicita ao operador que estabeleça a origem do plotador.**

Inicializa com zero as variáveis de controle da posição da pena (OLDX e OLDY).

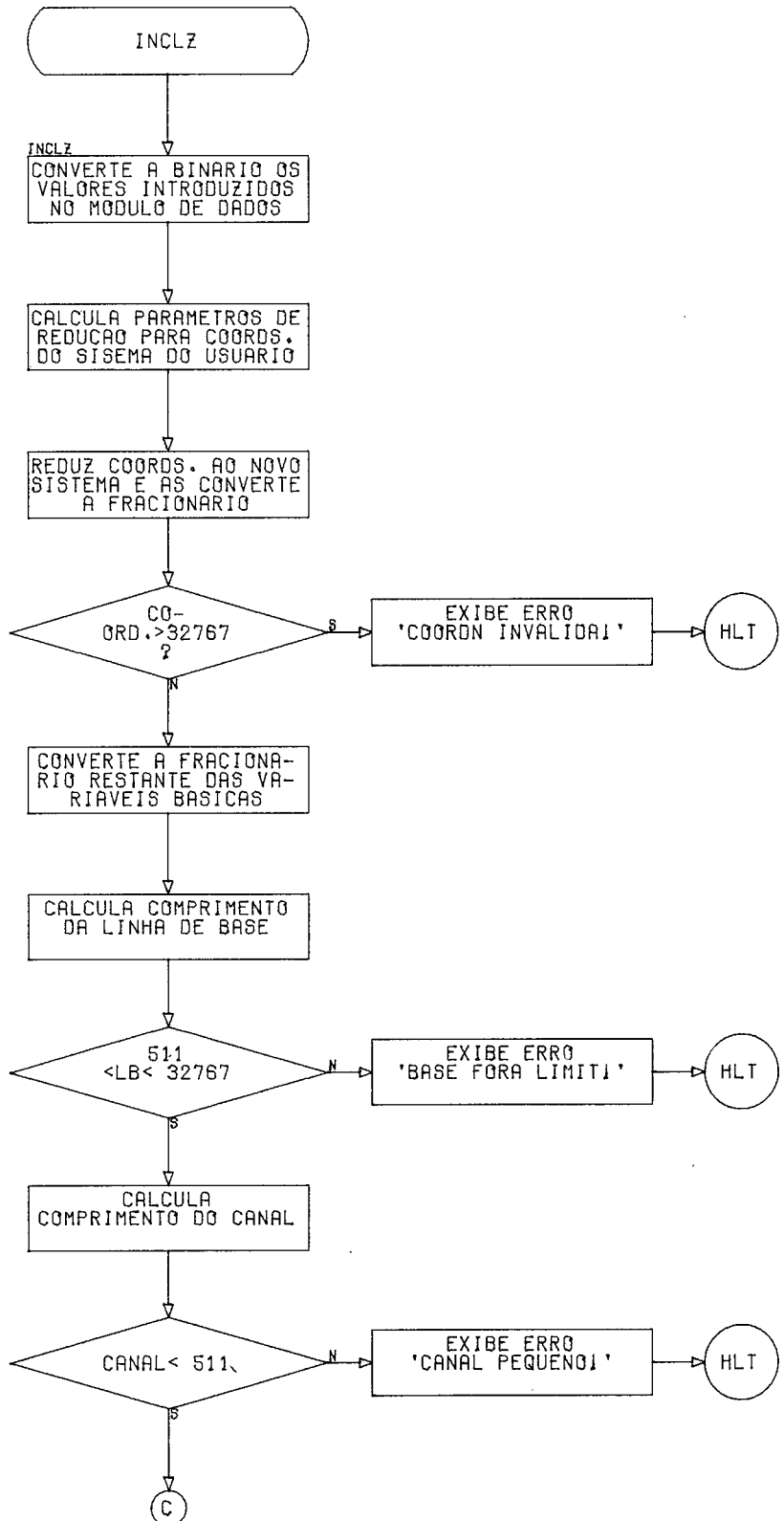
Exibe mensagem pedindo ao operador que coloque a pena do plotador na origem ('PLOTTER ORIGEM!') e o deixe pronto (READY). A origem da plotagem é o ponto START do canal.

Aguarda em looping o comando ENTRA a ser acionado no teclado pelo operador, em conclusão da tarefa.

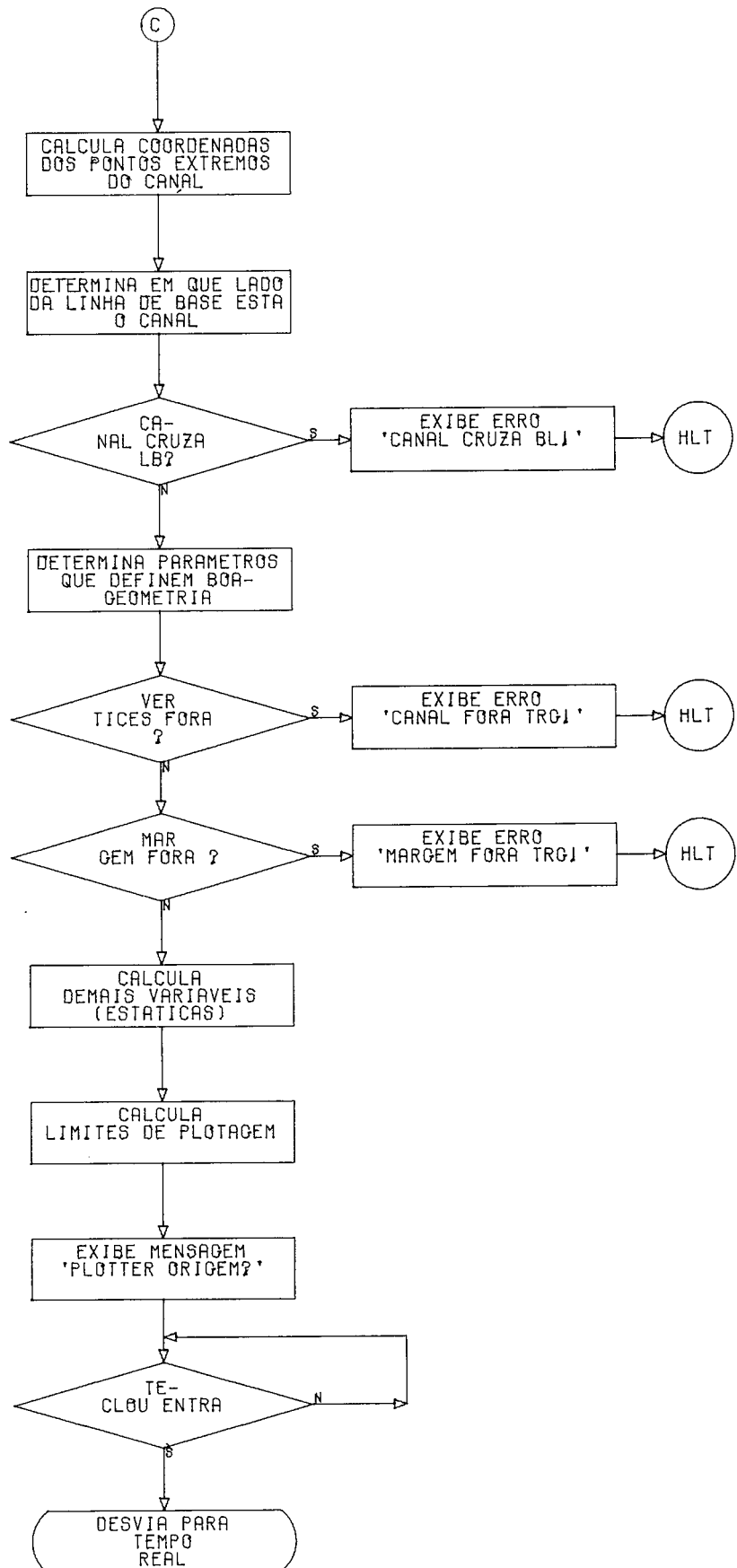
**P15- Termina, desviando para MÓDULO TREAL.**

Executa um desvio para o ponto de entrada do módulo de Tempo Real.

DIAGRAMA DE BLOCOS  
 MODULO DE PREPARO







## MÓDULO DE TEMPO REAL

### DESCRIÇÃO:

É o módulo de gerência do Tempo Real. Inicia, controla e encerra a execução dos ciclos de Tempo Real. Cada ciclo compreende: leitura, cálculo e plotagem de uma posição.

A leitura das distâncias no MRS é feita sob controle da interrupção (interna), que o programa habilita ou desabilita, por meio de uma CHAVE DE LEITURA. O MRS é ativado antes do início das leituras e desativado depois de efetuar dezesseis (16) leituras para cada canal (A e B) do MRS. A ocorrência de erros durante a leitura faz com que o programa desligue o MRS, emita mensagem de erro adequada e desvie para re-iniciar a leitura.

A determinação da posição, pelo módulo, consiste no cálculo de suas coordenadas XY, por meio de transformação de sistema. Durante o cálculo, o módulo testa a validade geométrica da posição e sua plotabilidade. Havendo erro, ele abandona a posição e re-inicia o ciclo, para determinar nova posição. Se a posição testada é consistente, o módulo faz sua plotagem encerrando o ciclo. O módulo mantém o sistema em ciclos até que o operador o faça parar por interrupção.

### SUBROTINAS CHAMADAS:

MRS (MRSOF, MRSGO), MENSA, PLOTA,  
ADSUB (AD, SB), GET (GET, UPDAT), ZRFAC,  
SRFAC, SQFAC, CPFAC, DV, MP, CR,  
DCBIN, TRANS.

## ALGORITMO COMENTADO:

PONTO DE ENTRADA TREAL:

**P1 - Restaura estado (status) de erro corrente no sistema e apresenta no display caracter de prontidão do Tempo Real (\*).**

A variável de controle da mensagem corrente (MSGCR) aponta, pelo seu valor, qual a mensagem de erro que o sistema está exibindo, devendo ser zerada para indicar que não há erro pendente no sistema. O caracter exibido na primeira posição do display (\*) informa ao operador que o sistema está em Tempo Real. Sempre que não houver erro pendente, as quinze seguintes posições do display se apresentarão em branco, sendo essa a configuração do display quando o sistema está em operação normal.

**P2 - FAIL: define RANGES médios como inexistentes.**

Os RANGES médios (LA e LB) são tornados negativos para indicar que o sistema não sabe, naquele instante, sua posição anterior.

**P3 - NEWS: executa um retardo (delay) antes de re-iniciar a operação de leitura pelo radar MRS.**

O console MRS não aceita ser re-comandado em menos de 350ms após ter sido desligado. Esse delay se faz necessário em casos de erro (FAIL), porque o tempo decorrido entre o desativar do console (MRSOF) e seu re-ativar (MRSGO) é menor que o habitual (1.5s), uma vez que não houve o processamento completo do cálculo da posição e a conseqüente plotagem.

- P4 - Inicializa com zero as variáveis de controle e as contadoras de erro da leitura dos canais A e B do MRS. Zera, também, a variável contadora de posições inconsistentes detectadas.**

O controle da leitura em cada um dos canais A e B do radar MRS é feito por meio de uma variável de controle (ACNTR e BCNTR). A variável de controle conta o número de leituras para o canal (A ou B) e as acumula. A variável contadora de posições inconsistentes contabiliza as posições determinadas com erro pelo radar.

- P5 - Inicializa a variável de sincronização da leitura, fazendo canal de leitura solicitado igual a A.**

Como a leitura do MRS se altera entre um e outro canal (A e B), é preciso inicializar o canal que está sendo solicitado (CANSO).

- P6 - Comanda início da leitura dos RANGES A e B pelo radar MRS.**

A rotina de serviço do MRS é chamada, em seu ponto de entrada MRSGO, para iniciar as atividades do console MRS.

- P7 - WAIT: prepara a interface MRS para receber a leitura e aguarda em looping a chegada dessa leitura via interrupção.**

Habilita interrupção na interface 8255 #1 e liga CHAVE DE LEITURA. Através dessa chave, a interface pode sentir a leitura pronta (READ FLAG) no MRS. Aguarda em looping, por 5.7ms, até a leitura se consumir via interrupção (interna) do sistema.

**P8 - Chegando a interrupção,  
então:**

o controle do sistema passa para o módulo de interrupção (INTRP) que promove a recuperação do dado apresentado pelo MRS.

A presença da leitura na interface do MRS se faz sentir por 1 bit denominado IBF (input buffer full), provocado pelo READ FLAG. A presença desse bit, aliada ao fato da CHAVE DE LEITURA estar ligada, gera uma interrupção na interface. A interrupção é tratada pelo módulo INTRP (de interrupção), conduzindo o processador ao ponto de entrada MRSRD na rotina de serviço do radar MRS.

**P9 - Inicia a recuperação do sistema tendo em vista a não chegada da leitura.**

Desliga a CHAVE DE LEITURA e desabilita interrupção na 8255 #1, para evitar interrupção pendente posterior, uma vez que o console MRS continua fazendo leituras.

**P10- Trata o erro, identificando o canal falto e incrementando seu contador de falhas. Alterna o canal a ser solicitado.**

Com base no valor da variável de controle do canal solicitado (CANSO), identifica o canal MRS, cuja leitura estaria sendo solicitada. O contador de falhas do canal (NOA ou NOB) é incrementado de um. Alterna (A por B ou B por A) o canal a ser solicitado na variável de controle (CANSO).

**P11- Testa se o contador de falhas do canal, anteriormente incrementado, já saturou; então:**

edita mensagem de erro 'FORA CANAL' ( A ou B).

Vá para passo seguinte (STOP:)

senão:

ignora o erro e aguarda apresentação de nova leitura pelo MRS, em P7 (WAIT:).

São toleradas até 16 falhas por canal, na tentativa de se determinar uma nova posição. Ultrapassado, por qualquer um dos canais, esse limite, o sistema avisa do erro, especificando inclusive o canal. No passo seguinte prossegue na recuperação do sistema, examinando se o contador de erro do canal em questão saturou. Caso isso tenha acontecido, faz a recuperação do sistema no passo seguinte, P12 (STOP:). Se não, ignora o erro, aguardando próxima leitura pelo MRS.

**P12- STOP: cessa a atividade do console MRS e desvia para recuperar sistema em P2 (FAIL:).**

Desativa o console MRS, chamando a subrotina de serviço do radar, no seu ponto de entrada MRSOF. Desvia para recuperar o sistema, abandonando as leituras feitas, até então, pelo radar.

**PONTO DE ENTRADA WRNRD:**

**P13- Erro na leitura do radar MRS: exhibe mensagem 'LEITURA MRS', desvia para P12 (STOP:).**

A rotina MRSRD ao recuperar a leitura apresentada pelo radar MRS na interface, testa a validade dos seus cinco dígitos BCD. Havendo dígito inválido, retorna neste ponto de entrada do módulo para tratar o erro.

PONTO DE ENTRADA DATA:P14- Trata leitura consistente apresentada pelo MRS.

A rotina MRSRD retorna neste ponto ao recuperar uma leitura MRS consistente. O tratamento dessa leitura consiste em remover, se houver, mensagem de leitura errada pendente ('LEITURA MRS'), e identificar o canal (ID) lido.

P15- Exerce controle de sincronismo (A/B) sobre a leitura MRS. Se canal lido não é igual ao canal solicitado, então:

desvia para aguardar próxima leitura em P7 (WAIT:).

Como a leitura MRS é produzida de forma contínua e alternada (CANAL A/CANAL B) pelo console MRS, é preciso compararmos sempre a identidade (ID) do canal lido com a do que queremos ler (CANSO). Se os canais não coincidem, devemos esperar por uma próxima leitura (SINCRONIZAÇÃO IMEDIATA).

P16- Troca canal solicitado. Testa o contador de leituras do canal lido. Se já saturou, então:

desvia para aguardar leitura do outro canal em P7 (WAIT:).

Faz canal solicitado (CANSO) igual ao próximo (A/B). Testa contador de leituras do canal lido (ACNTR/BCNTR) para verificar se já fez as 16 leituras previstas. Se já o fez, despreza leitura recebida para esse canal.

P17- Converte a binário leitura decimal do MRS.

Converte os cinco dígitos decimais da

leitura MRS que a rotina MRSRD armazenou no buffer (MRSBF). O binário resultante dessa conversão é armazenado em variável inteira.

**P18- Examina se a posição anterior foi determinada com êxito,**

então:

calcula desvio da leitura em relação a leitura média correspondente na posição anterior;

senão:

desvia para acumular a leitura em P20.

O valor da leitura média (LA ou LB) da posição anterior, se negativo, indica que ela não foi determinada com êxito. Nesse caso, a leitura não é testada quanto à sua variação máxima tolerada (20 metros), sendo simplesmente aceita.

**P19- Testa o desvio da leitura calculado. Se maior que 20 metros,**

então:

incrementa contador de posições mal de terminadas. Testa se contador já saturou,

então:

abandona a posição desviando para P12 (STOP:);

senão:

abandona a leitura, desviando para P7 (WAIT:).

O sistema admite até um máximo de 16 leituras com desvio excessivo ( 20m) independentemente do canal MRS em que ela ocorreu. Tal se sucedendo, abandona a posição que estava sendo determinada, desprezando todas as leituras para ela já realizadas. Não sendo o caso, apenas a leitura é desprezada.



P20- Incrementa contador e acumula a leitura do canal MRS.

A variável de controle da leitura (ACNTR ou BCNTR) para o canal MRS é atualizada no seu contador e acumulador de leituras.

P21- Testa se já atingiu o número de leituras desejadas para cada canal. Se não for o caso,

então:

desvia para aguardar próxima leitura em P7 (WAIT:).

A determinação de uma posição só estará completa, quando forem feitas 16 leituras para cada canal do MRS.

P22- Posição determinada: Desativa MRS.

De posse das leituras para os canais A e B, o sistema desativa o MRS (MRSOF) e passa ao cálculo do posicionamento propriamente dito.

P23- Calcula o RANGE das estações A e B. Restaura os contadores de leituras respectivos. Paralelamente ao cálculo da média, testa a validade desses RANGES, quanto aos limites estabelecidos. Se inconsistentes,

então:

prepara mensagem 'RANGE INVLD!' e desvia para P3 (NEWPS:);

senão:

apaga mensagem de erro de range, pendente no sistema, se houver.

O RANGE, para ser válido, deve estar compreendido entre 256m e 32767m. Os RANGES (RA e RB) são armazenados em forma de fracionários, apropriada para o cálculo de posicionamento.

**P24- Calcula a projeção horizontal dos RANGES.**

Devido à diferença de altura entre as estações de terra e a antena de bordo, se faz necessário corrigir o RANGE das estações (RA e RB).

**P25- Testa se a posição determinada pelos RANGES de A e de B está dentro do BULBO DE TRIANGULAÇÃO. Se está fora, então:**

edita mensagem de erro 'POSICIONAMENTO!' e desvia para P3 (NEWPS:);

senão:

apaga, se houver, mensagem de erro 'POSICIONAMENTO!' pendente no sistema.

Com bases nos ranges (RA e RB), determina o ângulo de intersecção de suas circunferências se fez fora dos limites estabelecidos para a triangulação. Utiliza o procedimento PERTN, que calcula e compara o referido ângulo com os limites estabelecidos (30° e 150°) para o BULBO.

**P26- Calcula as coordenadas X/Y da posição no sistema de coordenadas da linha de base.**

Calcula a intersecção das circunferências definidas pelos raios (RA e RB), determinando as coordenadas da posição (XP e YP). Leva em consideração o lado (SIDE) da triangulação na hora de decidir do sinal da ordenada (YP).

**P27- Calcula coordenadas X/Y da posição no sistema CANAL.**

As coordenadas da posição (XP e YP), determinadas no "sistema da LINHA DE BASE", são transformadas para o "sistema do CANAL",

por meio de rotação e translação. O cálculo de transformação usa a função TRANS para determinação dessas novas coordenadas (CHANN e OFFSET).

**P28- Converte as coordenadas X/Y da posição no sistema CANAL, em steps, por meio do fator de escala metro-step. Paralelamente, testa a sua plotabilidade em termos dos eixos X e Y do plotador. Se não for plotável,**

**então:**

**edita mensagem 'FORA DE ESCALA!' e desvia para P3 (NEWPS:);**

**senão:**

**apaga, se houver, mensagem 'FORA DE ESCALA!' pendente no sistema.**

A plotabilidade de uma posição, implica que suas coordenadas em step (PLOTX e PLOTY) estejam dentro dos limites estabelecidos para cada um dos eixos - X/YMAX e X/YMIN. O sistema despreza a posição não plotável e parte para de terminação de nova posição (NEWPS), editando mensagem de erro.

**P29- Examina status do plotador antes de iniciar a plotagem da posição. Se não estiver ligado,**

**então:**

**edita mensagem 'PLOTER FORA!' e desvia para P3 (NEWPS:);**

**senão:**

**apaga, se houver, mensagem 'PLOTER FORA!' pendente no sistema.**

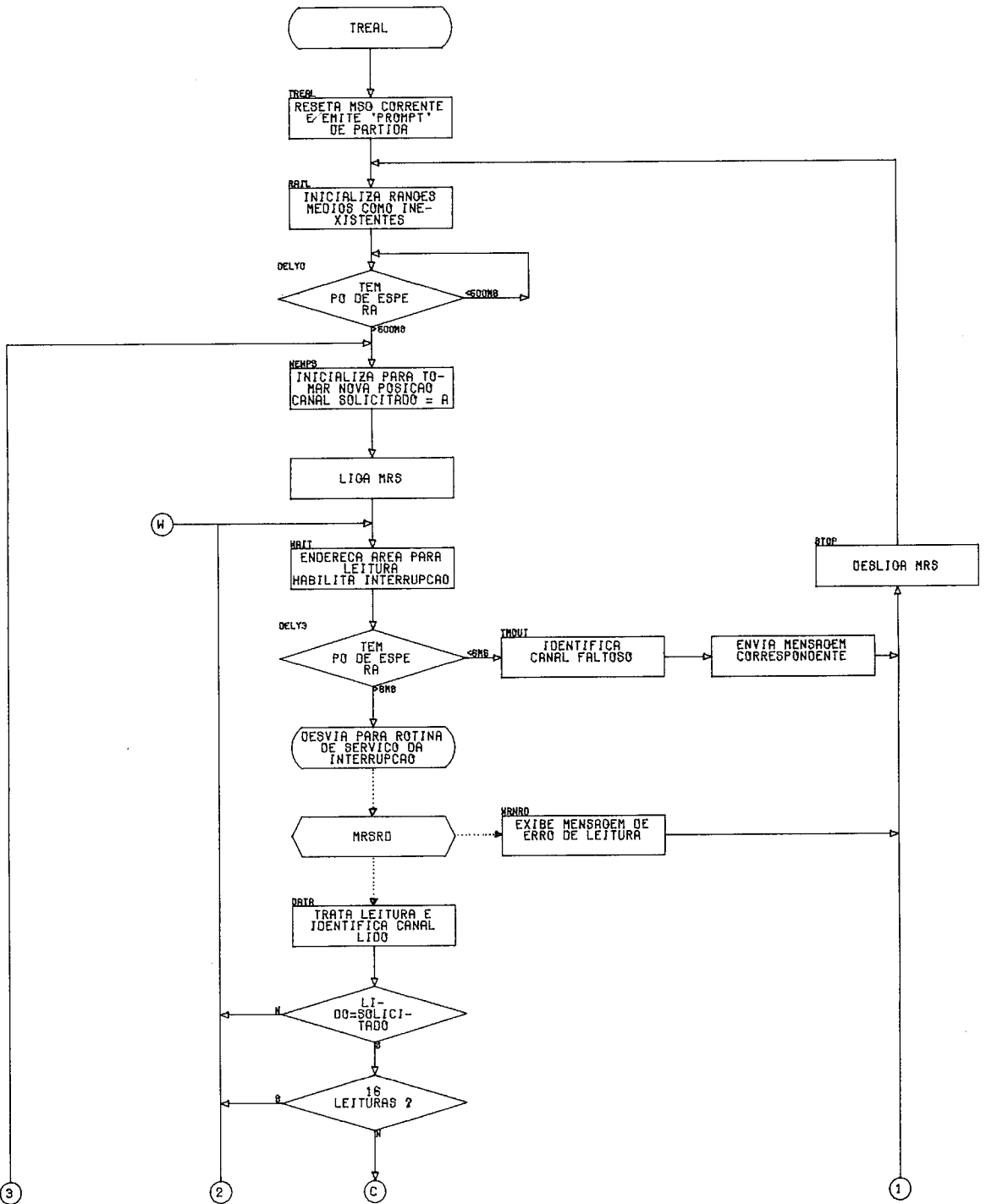
**P30- Plota a posição definida pelas suas coordenadas em steps.**

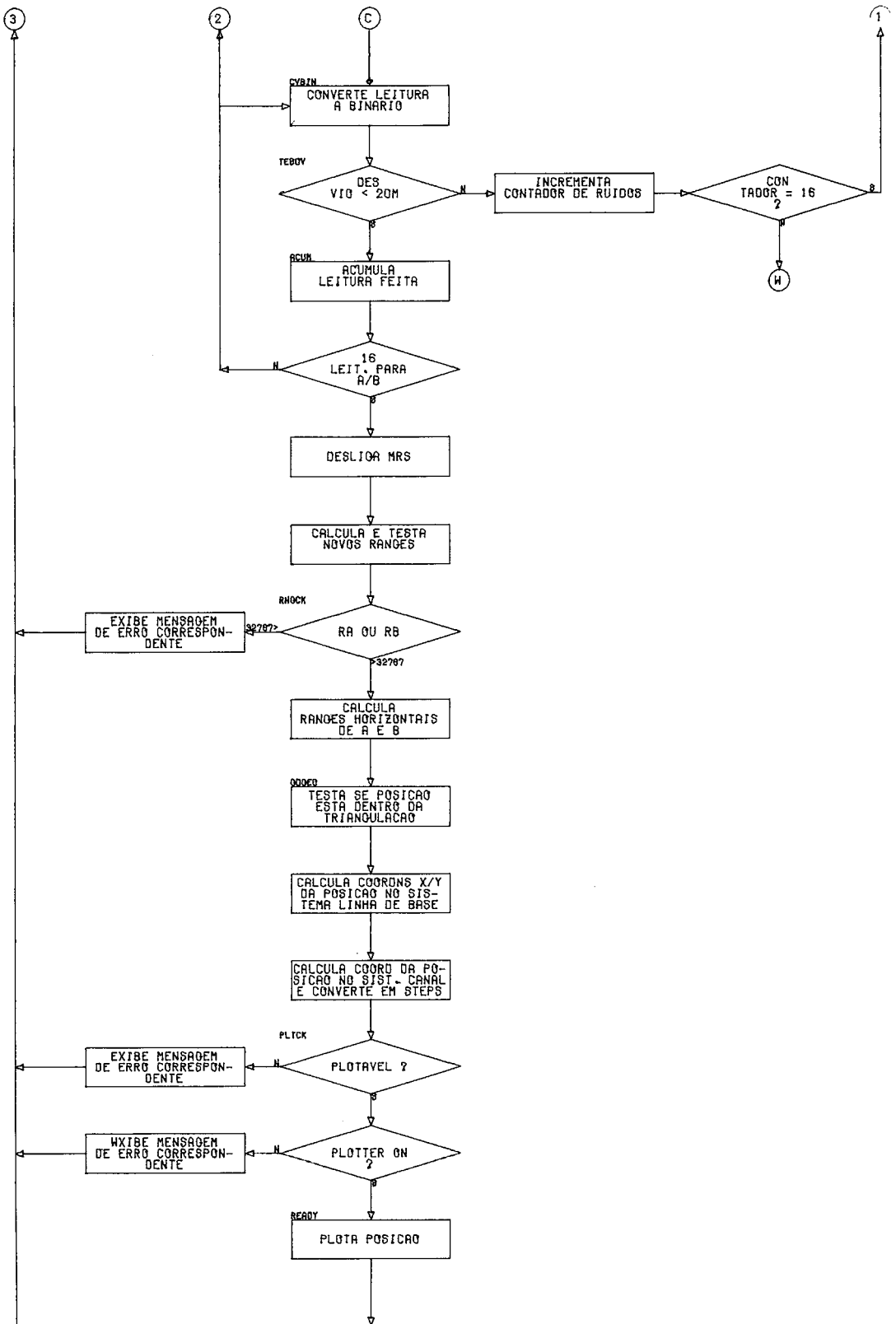
Através da subrotina PLOTA, são enviados incrementais para o plotador, de modo a conduzir a pena à posição correspondente àquela determinada na carta do plotador. As variáveis de controle da posição da pena (OLDX e OLDY) são atualizadas com os valores da posição plotada (PLOTX e PLOTY).

**P31- Volta para P3.**

Encerra o ciclo de tempo real, passando à determinação de nova posição, em P3 (NEWPS).

DIAGRAMA DE BLOCOS  
 MÓDULO DE TEMPO REAL





## CAPÍTULO IV

### IMPLEMENTAÇÃO DO SOFTWARE

#### 4.1 - METODOLOGIA DO POSICIONAMENTO

Compreende a determinação da posição de um móvel e sua representação em escala adequada na carta do plotador, partindo dos seguintes dados:

- a) coordenadas ortogonais das estações de terra (A e B), no sistema do usuário;
- b) coordenadas ortogonais dos pontos inicial e final (START e STOP) do eixo do canal de posicionamento, no sistema do usuário;
- c) altura das antenas das estações de terra e de bordo;
- d) ranges (RA e RB) da posição a ser determinada em relação às estações de terra A e B;
- e) SIDE, lado da Linha de Base (vetor AB) em que se faz o posicionamento (esquerdo ou direito).

As etapas desse cálculo se baseiam em transformações de sistemas de coordenadas. Três sistemas ortogonais de referência são usados:

- Sistema do Usuário;
- Sistema da Linha de Base;
- Sistema do Canal de Posicionamento.

O sistema do usuário é aquele em que o problema é proposto, isto é, LOCAL ou UTM, dependendo da escolha. O sistema da Linha de Base é formado, a partir da mesma, considerando seu eixo X positivo como o vetor AB e seu eixo Y positivo como o vetor normal a ela, passando por A. O sistema canal de

posicionamento é formado, a partir do mesmo, considerando seu eixo X positivo como o vetor STARTSTOP, e seu eixo Y positivo como o vetor normal a ele, passando por START.

A primeira etapa do cálculo é a determinação das coordenadas da posição ( $X'p$  e  $Y'p$ ), no sistema da Linha de Base. As fórmulas de intersecção de circunferência resolvem o problema:

$$X'p = \frac{BASE}{2} + \frac{RAH2 - RBH2}{2 \cdot BASE}$$

$$Y'p = RAH2 - X'p$$

o sinal de  $Y'p$  depende do lado da Linha de Base.

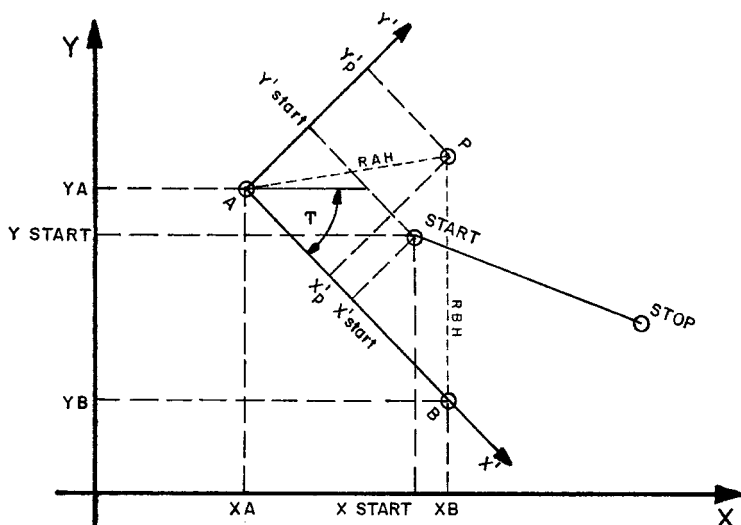
onde:

BASE = comprimento da Linha de Base;

RAH2 = quadrado da projeção de RA, no plano horizontal;

RBH2 = quadrado da projeção de RB, no plano horizontal.

Nosso objetivo é obter as coordenadas da posição ( $XCHAN$  e  $YOFSET$ ), no sistema do Canal de Posicionamento. Para isso, numa etapa intermediária, determinamos as coordenadas do ponto START do Canal de Posicionamento, no sistema da Linha de Base ( $X'start$  e  $Y'start$ ).



Considerando T como o ângulo de rotação entre os sistemas do



usuário e o da Linha de Base, podemos, através das fórmulas de transformação de coordenadas, escrever:

$$X'_{start} = (X_{START} - X_A) \text{COST} + (Y_{START} - Y_A) \text{SENT}$$

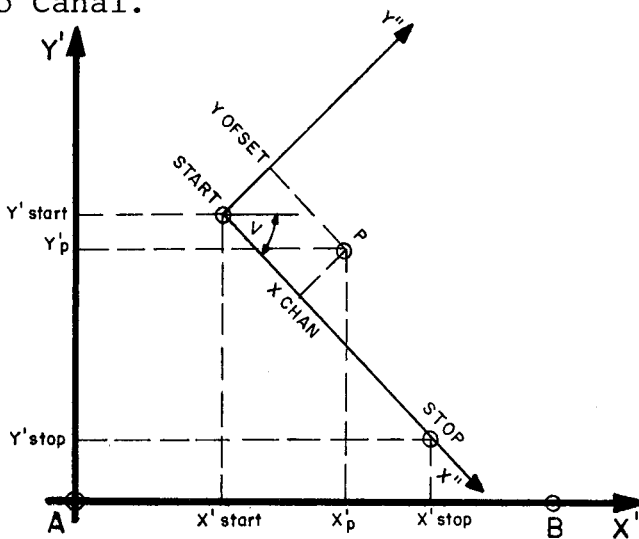
$$Y'_{start} = (X_{START} - X_A) \text{SINT} - (Y_{START} - Y_A) \text{COST}$$

onde:

COST = co-seno do ângulo T;

SENT = seno do ângulo T.

Finalmente, considerando agora V como o ângulo de rotação entre os sistemas da Linha de Base e do Canal de Posicionamento, efetuaremos a transformação das coordenadas de P para o sistema do Canal.



$$X_{CHAN} = (X'_{p} - X'_{start}) \text{COSV} + (Y'_{p} - Y'_{start}) \text{SENV}$$

$$Y_{OFFSET} = (X'_{p} - X'_{start}) \text{SENV} - (Y'_{p} - Y'_{start}) \text{COSV}$$

onde:

COSV = co-seno do ângulo V;

SENV = seno do ângulo V.

Através das coordenadas da posição no sistema Canal de Posicionamento ( $X_{CHAN}$  e  $Y_{OFFSET}$ ) e considerando os eixos X e Y do plotador coincidentes com os deste sistema, obtemos  $PLOTX$  e  $PLOTY$ , após aplicarmos, sobre  $X_{CHAN}$  e  $Y_{OFFSET}$ , o fator de escala apropriado à plotagem.

Conforme já tivemos oportunidade de mostrar (cf. 2.5), o erro do posicionamento depende do ângulo sob o qual se faz a intersecção dos ranges (RA e RB), e da precisão dos mesmos. No item 4.3 ilustraremos esse comportamento.

## **4.2 - ESCOLHA DE UMA ARITMÉTICA ADEQUADA**

A metodologia do Posicionamento, apresentada no item anterior, pedia, pela natureza de suas fórmulas, que utilizássemos uma aritmética de ponto flutuante, como forma de implementá-la. Tanto, que assim o fez a Motorola em seu MDS, tirando partido das adequadas facilidades do processador INTERDATA-70. Em nosso caso, SPE, havia restrições para implementações dessa natureza (processador 8080), o que nos levou a estudar detalhadamente diversas alternativas. A precisão (perda de significativos), o tempo de resposta e o espaço de memória total disponível (4Kbytes SBC 80/10) foram os dados básicos para esse estudo. Baseados nesses dados investigamos as seguintes aritméticas: de Ponto Flutuante, de Racionais e de Ponto Fixo. Vejamos, separadamente, cada uma delas.

### **a) Aritmética de Ponto Flutuante**

Opera com números reais, representados de forma finita, dentro de certa magnitude (EXPOENTE) e precisão (MANTISSA). É implementada usualmente em dois níveis de precisão: R\*4 e R\*8. A modalidade R\*4 representa o número real com quatro bytes, sendo um deles para EXPOENTE e os outros três para MANTISSA (7 significativos decimais). Já a modalidade R\*8, mais precisa, utiliza seis bytes para MANTISSA (14 significativos decimais). Essa última é a que mais convém à aplicação, uma vez que as fórmulas de solução do problema (cf. 4.1) não apresentam boa estabilidade numérica, diante dos dados que, na prática, são fornecidos a elas.

**i ) Vantagens.**

Obviamente, é o tipo de aritmética mais adequada ao problema.

**ii) Desvantagens.**

Ciclo de máquina não muito rápido, ausência de instruções adequadas e acesso apenas à 8 bits (BYTE) de memória são as maiores limitações à implementação dessa aritmética no 8080. Caso haja tal implementação, as possíveis consequências serão:

- baixa performance nos cálculos, durante a fase de Tempo Real, o que poderia comprometer o tempo de resposta do sistema;
- consumo elevado de memória das rotinas, o que iria exigir uma expansão de SBC 80/10.

**b) Aritmética de Racionais**

É a aritmética intermediária entre a de Ponto Fixo e a de Ponto Flutuante no que concerne a um equilíbrio entre a precisão (dígitos significativos) e a dinâmica de range. Também é chamada Aritmética Sem Ponto (No Point). Os números racionais ( $p/q$ ) são representados por um par de inteiros ( $p, q$ ), alocados em posições de memória não necessariamente contíguas. O inteiro  $q$  é sempre diferente de zero. O racional é considerado negativo, se  $p$  ou  $q$  o for ( $-p/q$  equivale a  $p/-q$ ). Nessa aritmética as quatro operações fundamentais são implementadas através de apenas duas: soma e multiplicação. Devido a limitações na representação dos inteiros  $p$  e  $q$  (uma, duas... palavras de memória), após cada operação deve-se efetuar a normalização do resultado. Com isso evita-se o rápido crescimento do numerador e do denominador do resultado. A normalização é conseguida através do cálculo do MDC - Máximo Divisor Comum - entre o numerador e denominador.

**i ) Vantagens.**

É de implementação mais fácil que a da de ponto flutuante. Permite melhor desempenho nos cálculos.

**ii) Desvantagens.**

Trata-se de uma aritmética pouco conhecida entre nós, cujo sucesso na aplicação, fundamentalmente, depende:

- de um estudo cuidadoso das fórmulas empregadas para se avaliar a possibilidade de crescimento do numerador e do denominador das quantidades calculadas.
- de características de CPU que o 8080 não possui: multiplicação de palavra simples (16 bits), soma e subtração de palavra dupla (32 bits).

**c) Aritmética de Ponto Fixo**

Aritmética originalmente presente em qualquer processador, variando apenas a forma de implementação. Em nosso caso, 8080, dispomos de adição, subtração e complemento a 1, para células de 8 bits (byte). A aritmética de ponto fixo é usualmente aplicada em operações entre inteiros - Aritmética Inteira. Nesse caso, o ponto se encontra (fixo) a direita dos bits da palavra e sua posição é um invariante em relação às quatro operações. A concepção da posição do ponto entre qualquer par de bits da palavra nos leva a uma outra modalidade de representação - números fracionários. Regras elementares permitem determinar-se facilmente a posição do ponto após ser efetuada qualquer uma das quatro operações - Aritmética Fracionária.

Para um estudo de implementação via Aritmética de Ponto Fixo, deve ser feita a reformulação das equações de transformação de sistemas (ver: 4.1), com o objetivo de eliminar a dependência de senos e co-senos. Assim, consideremos as expressões de definição do seno e co-seno dos ângulos V e de T:

$$\begin{aligned} \text{COST} &= (\text{XB} - \text{XA}) / \text{BASE} \\ \text{SENT} &= (\text{YB} - \text{YA}) / \text{BASE} \\ \text{COSV} &= \text{CONSTA} / \text{LENGTH} \\ \text{SENV} &= \text{CONSTB} / \text{LENGTH} \end{aligned}$$

o que, finalmente, nos permite escrever:

$$\begin{aligned} \text{X'start} &= ((\text{XSTART} - \text{XA}) (\text{XB} - \text{XA}) + (\text{YSTART} - \text{YA}) (\text{YB} - \text{YA}) / \text{BASE} \\ \text{Y'start} &= ((\text{YSTART} - \text{YA}) (\text{XB} - \text{XA}) - (\text{XSTART} - \text{XA}) (\text{YB} - \text{YA}) / \text{BASE} \\ \text{XCHAN} &= ((\text{X'p} - \text{X'start}) \text{CONSTA} + (\text{Y'p} - \text{Y'start}) \text{CONSTB}) / \text{LENGTH} \\ \text{YOFSET} &= ((\text{Y'p} - \text{Y'start}) \text{CONSTA} - (\text{X'p} - \text{X'start}) \text{CONSTB}) / \text{LENGTH} \end{aligned}$$

onde:

$$\begin{aligned} \text{CONSTA} &= ((\text{XSTOP} - \text{XSTART}) (\text{XB} - \text{XA}) + (\text{YSTOP} - \text{YSTART}) (\text{YB} - \text{YA})) / \text{BASE} \\ \text{CONSTB} &= ((\text{YSTOP} - \text{YSTART}) (\text{XB} - \text{XA}) + (\text{XSTOP} - \text{XSTART}) (\text{YB} - \text{YA})) / \text{BASE} \end{aligned}$$

e LENGTH é o comprimento do Canal de Posicionamento.

Paralelamente a essa reformulação, devem ser introduzidas limitações nos dados básicos do problema (posicionamento). Tais limitações compreendem o estabelecimento de valores superiores e inferiores para as seguintes variáveis: ranges (entre 150m e 32000m), comprimento da Linha de Base (entre 1024m a 16484m) e comprimento do Canal de Posicionamento (a partir de 512m até o que o Bulbo suportar). Devem ainda, as coordenadas XY dos pontos A, B, START e STOP, configurarem, após transformação de minimização, valores inferiores a 32767m. O estabelecimento dessa limitação teve como base o valor máximo 32767. - inteiro de 2 bytes.

Essa modalidade de aritmética (ponto fixo) foi a que maior atenção recebeu de nossa parte. Daí por que apresentaremos, separadamente, os estudos que fizemos para os Inteiros e Fracionários.

### - Aritmética Inteira.

Escolhemos como base desse estudo o inteiro  $I^*4$ . O  $I^*4$  é a modalidade de representação em que o Inteiro é alocado em quatro bytes (32 bits) de memória, na forma complemento a 2. O sinal do número é dado pelo bit de mais alta ordem - positivo se o bit for zero. Essa modalidade permite uma variação de valores desde  $-2^{31}$  até  $+2^{31}-1$ . A Aritmética Inteira nos obrigou a fazer um estudo detalhado da perda de significativos nas fórmulas. Foi dada maior ênfase às fórmulas que continham divisão e raiz quadrada, também àquelas que se utilizavam desses resultados (propagação de erro). O cálculo de  $Y^p$  foi o que se mostrou mais crítico de todos, conforme resultados apresentados pelo SSA - Sistema de Simulação Aritmética (cf.cap.4.3). A obtenção de um valor satisfatório para ele só se tornou possível com a correção (recuperação de significativos) do radicando de sua fórmula de definição.

#### i ) Vantagens.

O desenvolvimento de rotinas de aritmética inteira é de fácil implementação em qualquer processador.

#### ii) Desvantagens.

Tempo de resposta e memória crescem em função da necessidade de recuperar significativos e filtrar dados.

Introduz nos dados básicos do posicionamento as já aludidas limitações.

Obs.: A modalidade  $I^*6$ , um inteiro de maior capacidade, chegou a ser cogitada, como forma de se evitar a perda de significativos. Porém a perspectiva do tempo de resposta vir aumentar muito, desencontrajou a idéia.

### - Aritmética Fracionária.

Como decorrência da preocupação com a perda de significativos, proveniente das operações de divisão e de raiz quadrada

na aritmética inteira, buscou-se uma nova representação - o número fracionário. Um número que definimos com dois bytes em sua parte inteira e dois outros em sua parte fracionária ( $I*2.F*2$ ), com as mesmas características do  $I*4$  no que concerne à definição de sinal. O primeiro passo dado no estabelecimento desta aritmética foi a proposição de novas operações de raiz quadrada e divisão. Operações que, dando como resultado fracionário, evitam o colapso de significativos da aritmética inteira.

#### i ) Vantagens.

Precisão satisfatória dentro dos limites estabelecidos para os dados, não necessitando de recuperação de significativos.

#### ii) Desvantagens.

Introduz limitações aos dados básicos do posicionamento.

A aritmética por nós escolhida é resultante da combinação das aritméticas Inteira e Fracionária: a essa combinação de nominamos ARITMÉTICA SPE.

### 4.3 - ARITMÉTICA SPE

#### I M P L E M E N T A Ç Ã O

A aritmética SPE opera com dois tipos de número: inteiro e fracionário. O número inteiro implementado é da modalidade  $I*4$ , o fracionário,  $I*2.F*2$ . Das operações aritméticas, somente a soma e subtração utilizam operandos idênticos (inteiro com inteiro, fracionário com fracionário). A divisão e multiplicação, por combinarem operandos de um modo especial, tem a capacidade de transformar quantidades: na multiplicação, fracionário por fracionário, que resulta inteiro; na divisão, inteiro por fracionário, que resulta fracionário.

Essa aritmética especial possui um acumulador - FAC - que é a base de operação para todos os cálculos. O FAC é sempre um dos operandos nesses cálculos. Através do MTPACK, essa aritmética se comporta como uma unidade de aritmética fictícia do HARDWARE. Exceptuando as rotinas LDSTO, DCBIN e GET, os demais pontos de entrada do pacote representam as operações possíveis dessa aritmética. A análise de cada uma dessas rotinas fornece maiores detalhes sobre as operações.

As variáveis e constantes, utilizadas nos cálculos do SPE, classificam-se, quanto à natureza, em inteiras e fracionárias. A natureza é atribuída a elas em função de características próprias, ou de implicações que venham a ter nas expressões. Tais implicações compreendem o exame da perda de significativos, acarretada por elas nos cálculos. A atribuição de valores a variáveis é feita pela rotina ST (instrução STORE) devendo ser observada apenas a correspondência em tipo, entre o valor (FAC) e a variável. Através da rotina LD (instrução LOAD) podemos carregar, no acumulador (FAC), valores de constantes ou de variáveis.

As expressões calculadas no SPE compreendem as de resolução das fórmulas da metodologia (posicionamento) e das de consistência de dados (filtragem numérica). Essas expressões por envolverem, de um modo geral, muitos operandos, foram implementadas com auxílio de uma rotina padrão: TRANS. A TRANS resolve a expressão mais completa (geral) do cálculo, isto é, uma transformação de coordenadas. Essa rotina foi desenvolvida para receber seus parâmetros, via Lista de Variáveis. Através dela, evitou-se o desenvolvimento, operando a operando, da maioria das expressões do cálculo. Isto foi possível a partir do estabelecimento de uma lista de variáveis adequada ao cálculo de cada uma delas.



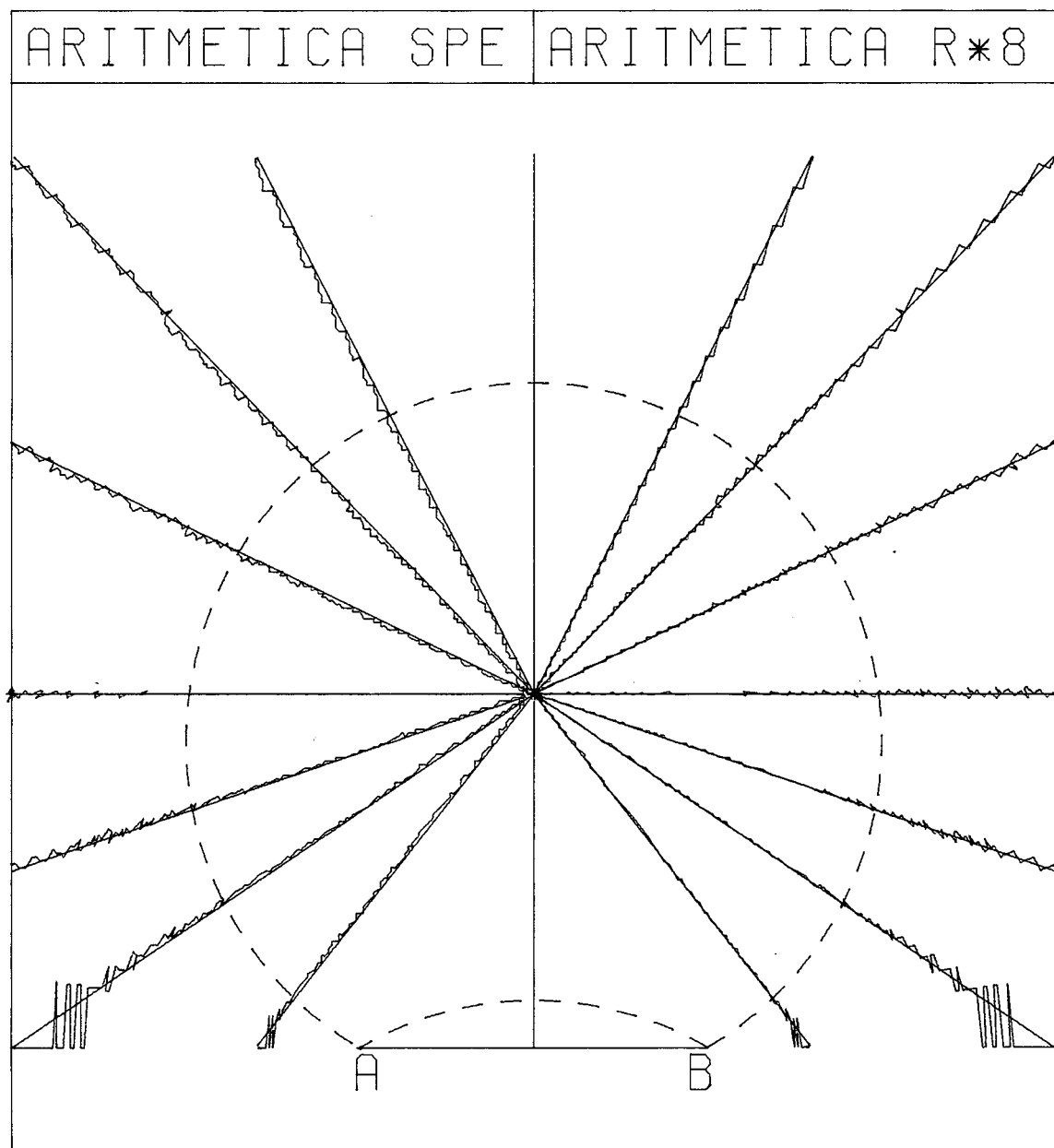
## S I M U L A Ç Ã O

A simulação dos cálculos do SPE constituiu fator decisivo na escolha da aritmética do sistema. Através da simulação, analisou-se o comportamento do SPE diante dos diversos tipos de aritmética estudados. Essa simulação compreendia a execução dos cálculos do Módulo de Preparo e do Módulo de Tempo Real, em condições idênticas àsquelas do sistema. O SSA - Sistema de Simulação Aritmética, foi a principal ferramenta que utilizamos nessa fase.

O SSA é um sistema de informação escrito em linguagem FORTRAN IV, desenvolvido e implementado no sistema IBM/3. Compreende cerca de 10 (dez) programas, 18 (dezoito) subrotinas e dois (2) arquivos. O SSA simula operações utilizando as aritméticas: R\*4, R\*8 e SPE. A aritmética inteira (I\*4) teve sua implementação descontinuada, tão logo o SSA mostrou que seus primeiros resultados eram pouco satisfatórios.

Para utilizar SSA, o usuário deve primeiramente prover os dados básicos da triangulação e, em seguida, fornecer os rumos para o posicionamento. Os dados da triangulação, tal qual no SPE, são imediatamente criticados pelo sistema através de programa que simula o Módulo de Preparo. Se a geometria for satisfatória, o SSA solicita então, ao usuário, os rumos do posicionamento. A partir dos rumos fornecidos, o SSA calcula os ranges hipotéticos A e B, para prosseguir com a simulação. Esses rumos podem ser segmentos de reta ou arcos de círculo, que o sistema discretiza, em função do incremento do plotador Houston (0.1mm). O SSA apresenta os resultados da simulação em relatórios ou em plotagens. A plotagem é feita em plotter de precisão: Calcomp 936. Essa plotagem, conforme veremos mais adiante, apresenta certa rugosidade em seu traçado, devido ao truncamento dos ranges - RA e RB são inteiros.

A figura abaixo apresenta o resultado de uma simulação pl<sub>o</sub>tada. Nela podemos observar o comportamento da aritmética SPE e da aritmética R\*8, em diversas situações de posicionamento em relação ao Bulbo de Triangulação.



#### 4.4 - ALGORITMO EXPEDITO DE PLOTAGEM

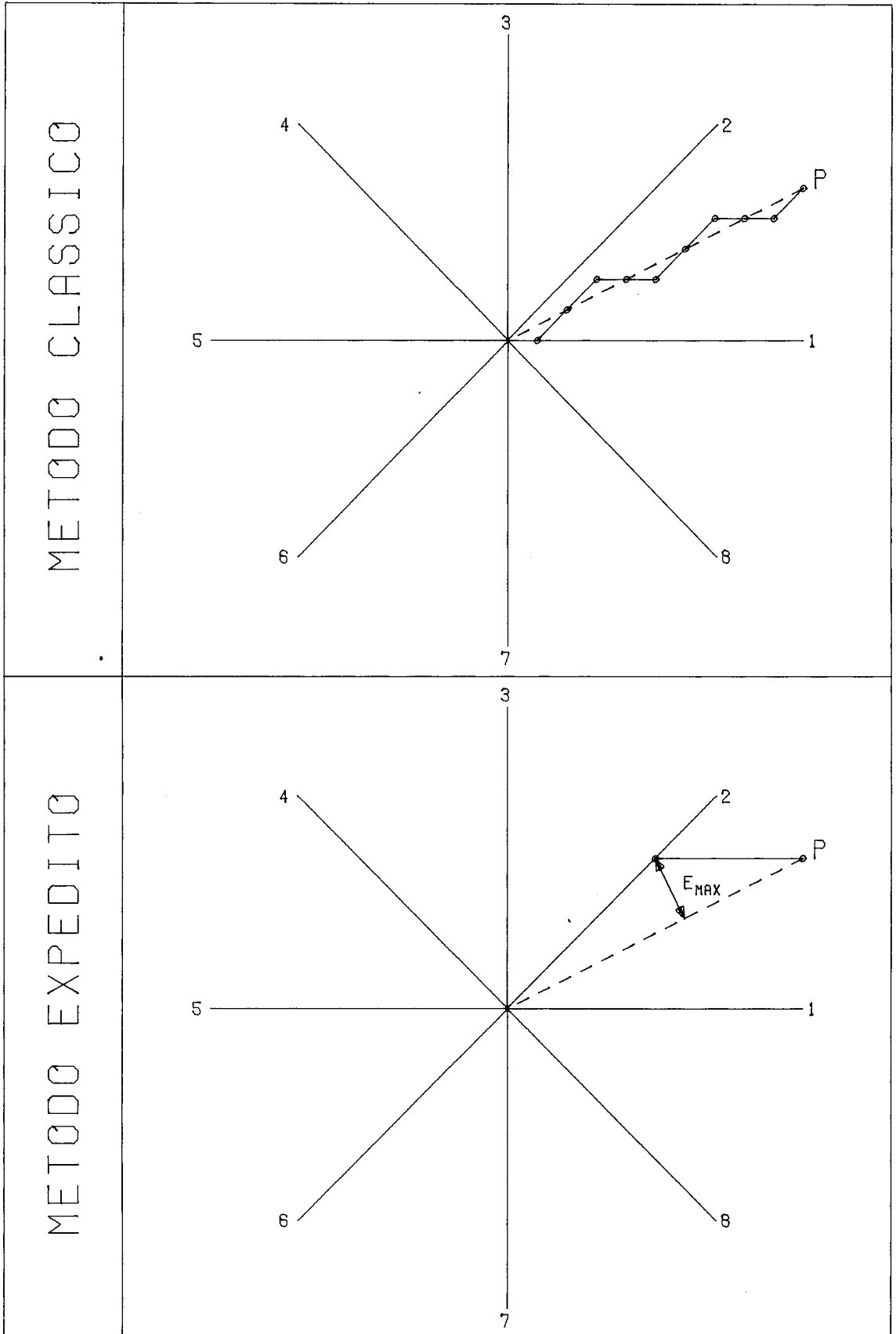
Conforme já vimos, o plotador HOUSTON (como qualquer plotador incremental) só é capaz de efetuar deslocamentos em certas direções (fundamentais). Tais direções (oito, ao todo) são obtidas mercê da combinação dos incrementos elementares do papel e da pena. Assim sendo, qualquer deslocamento, para ser por ele efetuado, precisa ser previamente vetorizado. A vetorização consiste na decomposição desse deslocamento em uma série de vetores-incremento, o que, considerando a representação do deslocamento, pode ser feito com cálculo mínimo ou erro mínimo.

O processo clássico de vetorização consiste, em resumo, na determinação de um par de vetores incrementos (direções fundamentais) que envolvem a direção do deslocamento pedido, e na geração de uma combinação desses incrementos, de modo que se obtenha um trajeto que resulte o mais aproximado possível do real, isto é, com erro mínimo. Já o processo EXPEDITO apenas seleciona o par de vetores e determina seus módulos, de modo que a resultante seja o deslocamento: opera com cálculo mínimo.

O primeiro é um processo preciso, porém lento, e requer alguma sofisticação para sua implementação. Quanto ao segundo, seu algoritmo é uma mera obtenção de componentes, o que torna sua implementação imediata. É um processo rápido, embora produza resultados não muito precisos.

Ocorre que, em se tratando de pequenos deslocamentos (1, 2, 3, 4... incrementos), o erro produzido pelo segundo processo torna-se desprezível, por ser visualmente pouco perceptível. No caso particular do SPE, o ciclo de Tempo-Real registra deslocamentos inferiores a 10m (draga em operação), isto é, 1, 2, 3, 4... incrementos, dependendo da escala de plotagem. Por essa razão, ignoramos o aspecto impreciso do segundo processo e adotamo-lo. Da mesma forma o fez a Motorola em seu MDS.

Mostraremos, a seguir, que representação de deslocamento teríamos, se submetêssemos os dois métodos a uma mesma situação de plotagem: o ponto "P" de coordenadas (10;5), medidas em incrementos.



#### 4.5 - RECURSOS UTILIZADOS NO DESENVOLVIMENTO DO PROJETO

Por ocasião da fase de detalhamento do PSPE, já dispúnhamos de algumas rotinas de SOFTWARE, desenvolvidas no simulador do NCE: SOS/PTI. Este simulador é um cross-software capaz de desenvolver SOFTWARE para microprocessador da linha INTEL (8008 e 8080). Ele oferece ao usuário: editor de texto, compilador ASSEMBLER, compilador PL/TI, link-editor, biblioteca, depurador e recursos de um sistema de grande porte (B-6700). As rotinas de que já dispúnhamos eram, entre outras, as de cálculo (as mais elementares) e as de aquisição de dados por teclados (DADOS).

Ao decidirmos pela adoção do processador 8080 (cf. 2.2), aceitamos também, ao pressentirmos dificuldades com o emprêgo apenas da simulação, a oferta da firma montadora do protótipo, que colocou à nossa disposição o seu sistema INTELLEC. Trata-se de um sistema próprio para desenvolvimento integrado de hardware e software para microprocessador da linha INTEL. Em seu modelo MDS-80, o usuário dispõe dos seguintes recursos: editor de texto, compilador ASSEMBLER e depurador de Tempo-Real, (ICES), residentes em seu microcomputador. Sua filosofia de desenvolvimento permite ao usuário desenvolver, além do SOFTWARE, os vários estágios de HARDWARE do protótipo. Com os recursos do INTELLEC, tornou-se oportuno desistirmos da simulação e optarmos por uma nova metodologia de trabalho: depuração de Tempo Real.

A depuração de Tempo-Real possibilitou, em curto prazo, a integração do SOFTWARE com o HARDWARE, facilitando o desenvolvimento das rotinas de entrada/saída (IOPACK) e os dos módulos de manuseio da interrupção (TREAL e INTRP). Isso seria extremamente difícil de ser feito com o cross-software.

Se, por um lado, essa opção nos trouxe vantagens, como as anteriormente citadas, trouxe-nos também desvantagens: ao emigrarmos de um sistema de grande porte (residência do cross-software) para um microcomputador, perdemos na versatilidade e velocidade dos seus periféricos de entrada/saída, o que prejudicou sensivelmente o rendimento dos trabalhos, a ponto de, em certa etapa do projeto, utilizarmos uma leitora serial de cartões 2760 da HEWLETT-PACKARD.

Concluída a montagem do projeto, entramos na fase dos testes simulados de laboratório. Para essa etapa, contamos com um valioso recurso: um simulador de radar. Este dispositivo é capaz de fornecer ao console MRS distâncias simuladas. Apesar da limitação que apresentava (incapacidade para fornecer distâncias diferentes para os canais A e B) pudemos, com ele, simular operações de posicionamento suficientes para dar ao protótipo condições de ser testado no campo.

## CAPÍTULO V

### ANÁLISE DOS RESULTADOS EXPERIMENTAIS

#### 5.1 - FASE EXPERIMENTAL

O protótipo SPE realizou seu teste preliminar a 20/06/78, a bordo da draga MACAPÁ no Porto de Sepetiba. Após ter sido aprovado, retornou ao Laboratório para ajustes finais de software e de hardware. Iniciou-se, então, a fase experimental: o protótipo foi instalado a bordo daquela draga e a equipe, preparada para operar com ele. Durante dois meses, engenheiro supervisor, comandante e mestres de pequena cabotagem (responsáveis pela condução da draga) atuaram junto ao protótipo, observando-o, manipulando-o, e recebendo, ao mesmo tempo, orientação e treinamento adequados. Nesse período executou posicionamento da draga em operação de abertura do canal de acesso e da bacia de evolução do porto, tendo oportunidade de combinar diversas situações geométricas de triangulação.

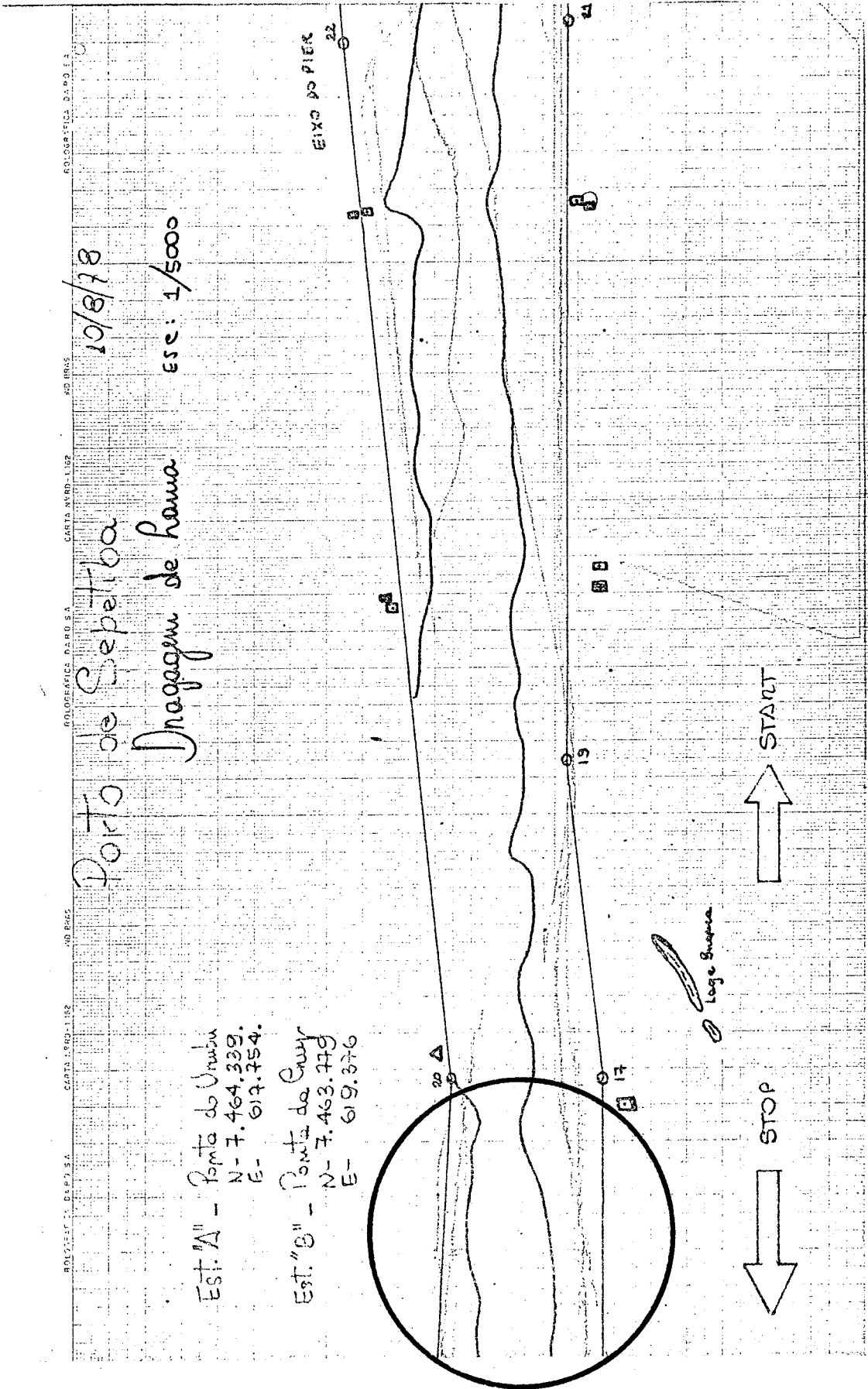
Os casos que se seguem exemplificam duas situações extremas de operação:

##### A. SITUAÇÃO NORMAL

Em condições normais de operação, o sistema apresentou traçado satisfatório, representando, na plotagem, com boa resolução, as trajetórias. Na página que se segue, pode-se observar, nitidamente, a firmeza do posicionamento.

Tal firmeza já não acontece na região assinalada. Embora à primeira vista isto possa parecer resolução deficiente, nada mais é que uma situação de geometria não muito favorável: aproximação excessiva do BULBO DE TRIANGULAÇÃO (cf. 2.6.).

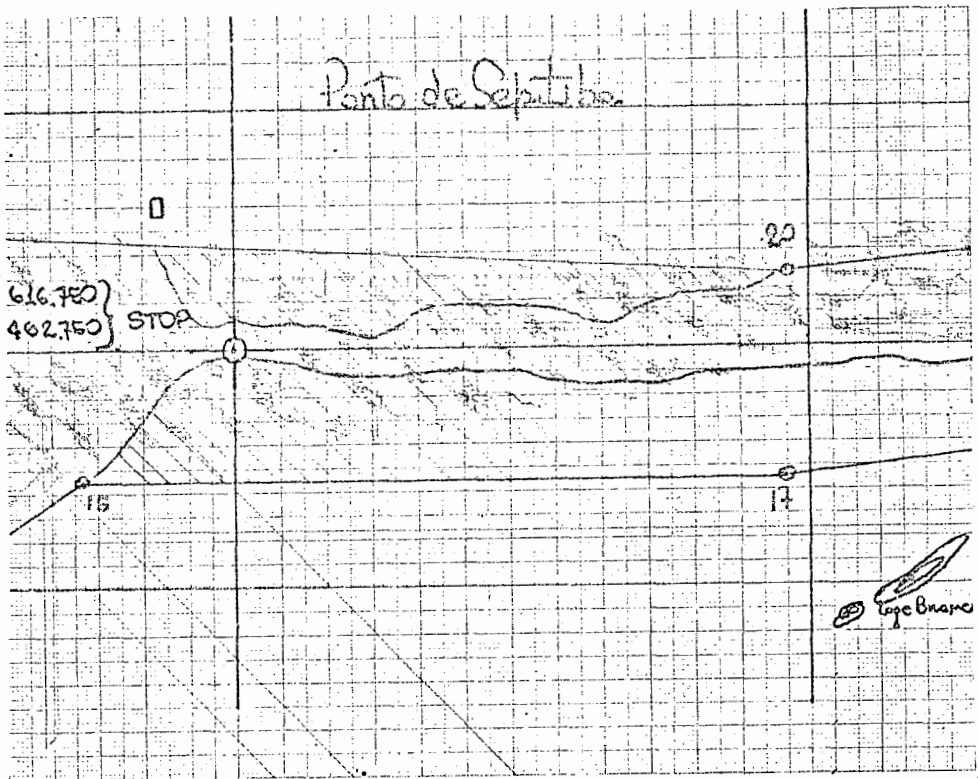
FIGURA REFERENTE À SITUAÇÃO "A".





## B. OPERANDO COM INTERFERÊNCIA DE RADAR

A figura abaixo caracteriza uma situação típica de interferência de radar. Nela podemos observar que o traçado do plotador, anteriormente claro, já não se repete. O aspecto hachuriado desse traçado é devido à variações anormais nos RANGES (A e B), produzida por interferências de radar resultando numa indefinição do posicionamento.



Essa condição anormal de operação decorre, em geral, da não observância dos cuidados prescritos para instalação do sistema. Mas podem surgir, também, em decorrência de fatores ambientais. O caso aqui ilustrado foi devido à instalação incorreta da antena de bordo.

A natureza dos serviços prestados pelo equipamento, nesta ocasião, foi a mesma daqueles prestados pelo seu antecessor - o MDS, em condições idênticas de dificuldades técnica e submetido às mesmas intempéries.

## 5.2 - FASE DE OPERAÇÃO

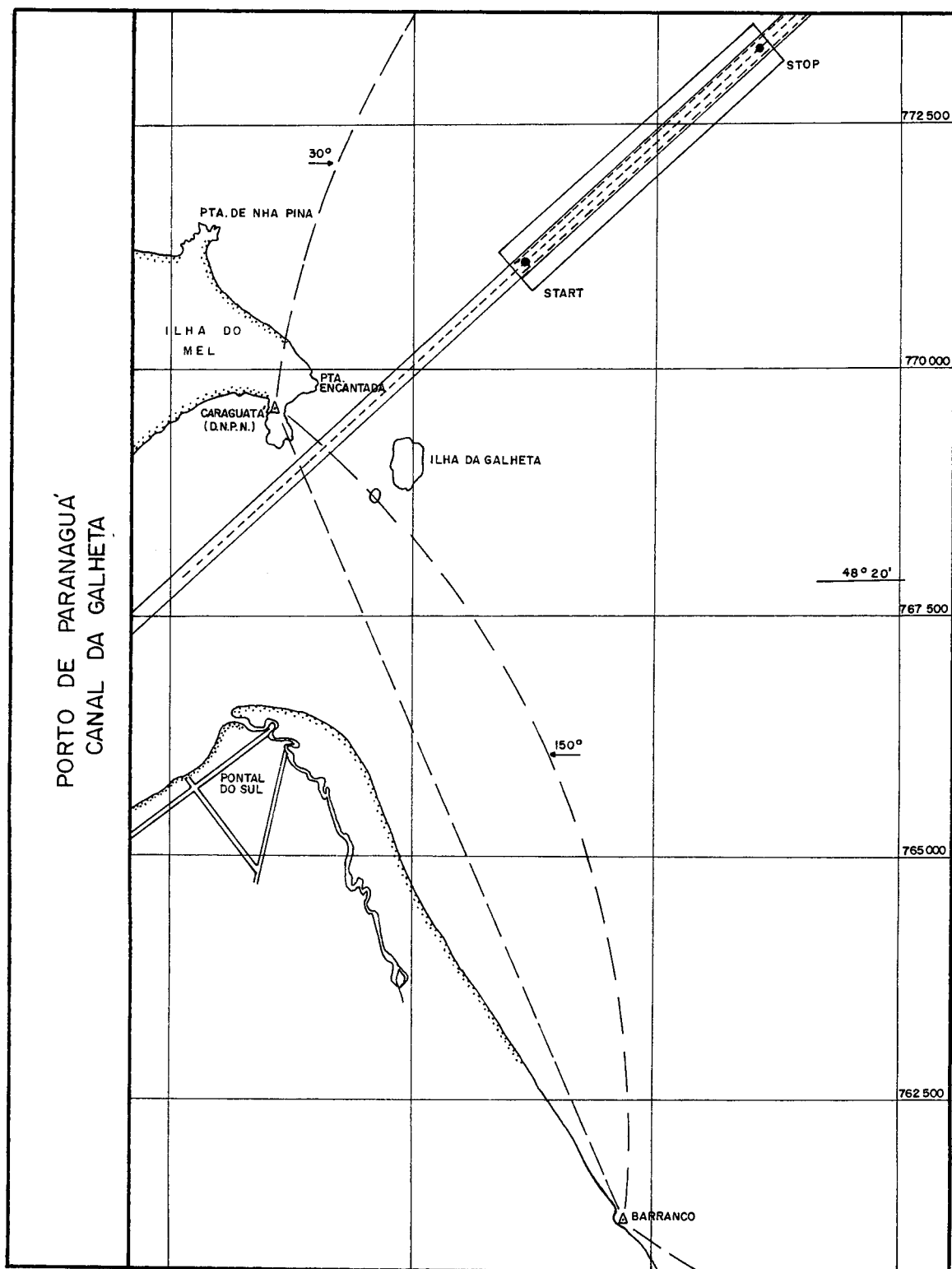
Concluída a fase de instalação, o protótipo SPE entrou em regime normal de funcionamento, incorporando-se à rotina de serviços da Draga Macapã (CBD). No momento, participa de operação no Porto de Paranaguá, onde realiza dragagem de precisão, para acerto dos taludes do Canal da Galheta.

Aberto em 1974 e representando hoje a única via de acesso ao interior do porto, esse canal operou normalmente até meados de 1978, quando, então, a acomodação natural dos seus taludes provocou assoreamento em determinados trechos, impossibilitando, praticamente, a navegação de grande porte. Nessa ocasião, convocada pela PORTOBRÁS, a Companhia Brasileira de Dragagem (CBD) enviou a draga Macapã para desobstruí-lo.

Para tanto, foi necessário, em primeiro lugar, planejar a triangulação. Foram estabelecidas estações de terra nos pontos: CARAGUATÁ (estação A) e BARRANCO (estação B). Foram definidas características do canal de posicionamento (pontos: START e STOP; SEMILARG) e a escala de plotagem. O sistema SPE recebeu, então, os seguintes dados para operar:

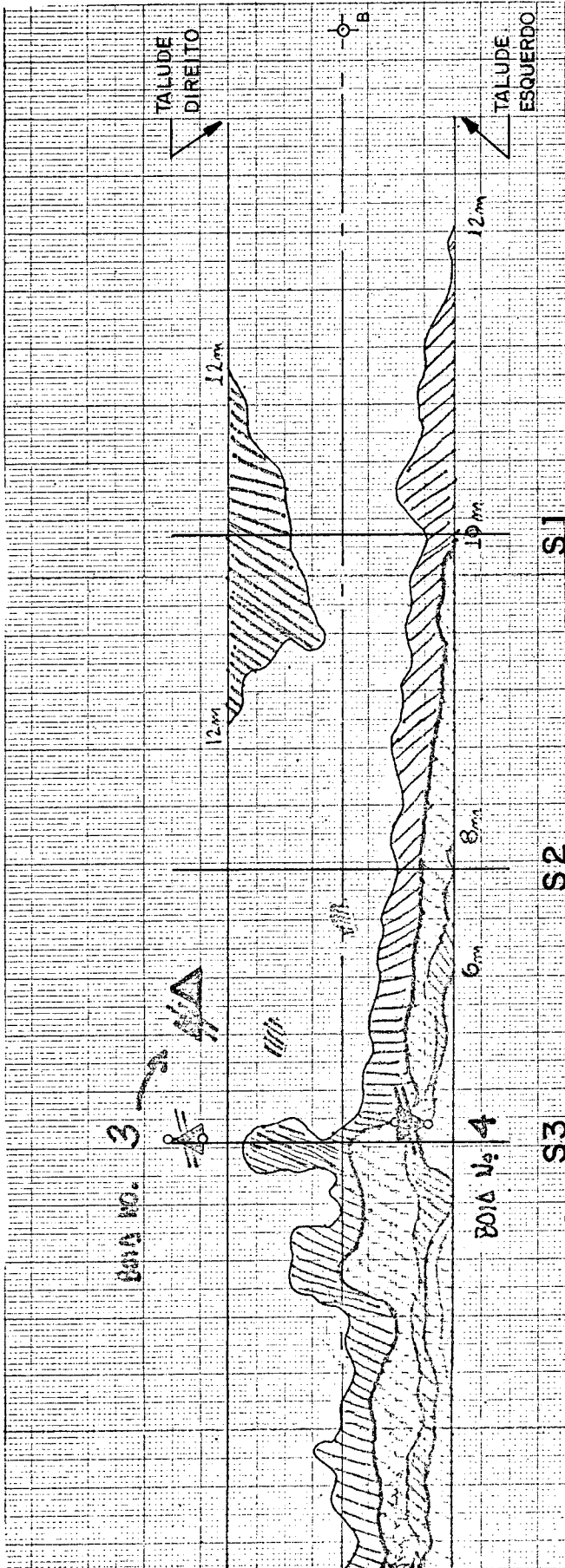
XA = 769621	XB = 761252	XSTART = 771015	SLARG = 000250
YA = 168918	YB = 165295	YSTART = 166442	ESCALA = 000005
ZA = 000050	Zb = 000008	XSTOP = 773364	ALT = 000030
		YSTOP = 163848	

A seguir, o croqui que apresenta a triangulação de posicionamento realizada nessa operação: nele podemos observar que, na localização das estações de terra A e B, procurou-se levar em conta condições de geometria e visibilidade do local da dragagem.



As páginas posteriores ilustram etapas dessa operação. Nelas apresenta mos três cartas (trechos de cartas) características da estratégia do po sicionamento: o plano de trabalho e sua evolução.

Para facilitar a análise das mesmas, acrescentamos, ao final deste capí tulo, perfis de secções transversais de pontos importantes do canal, com as duas linhas de sondagem correspondentes: INICIAL e ATUAL.

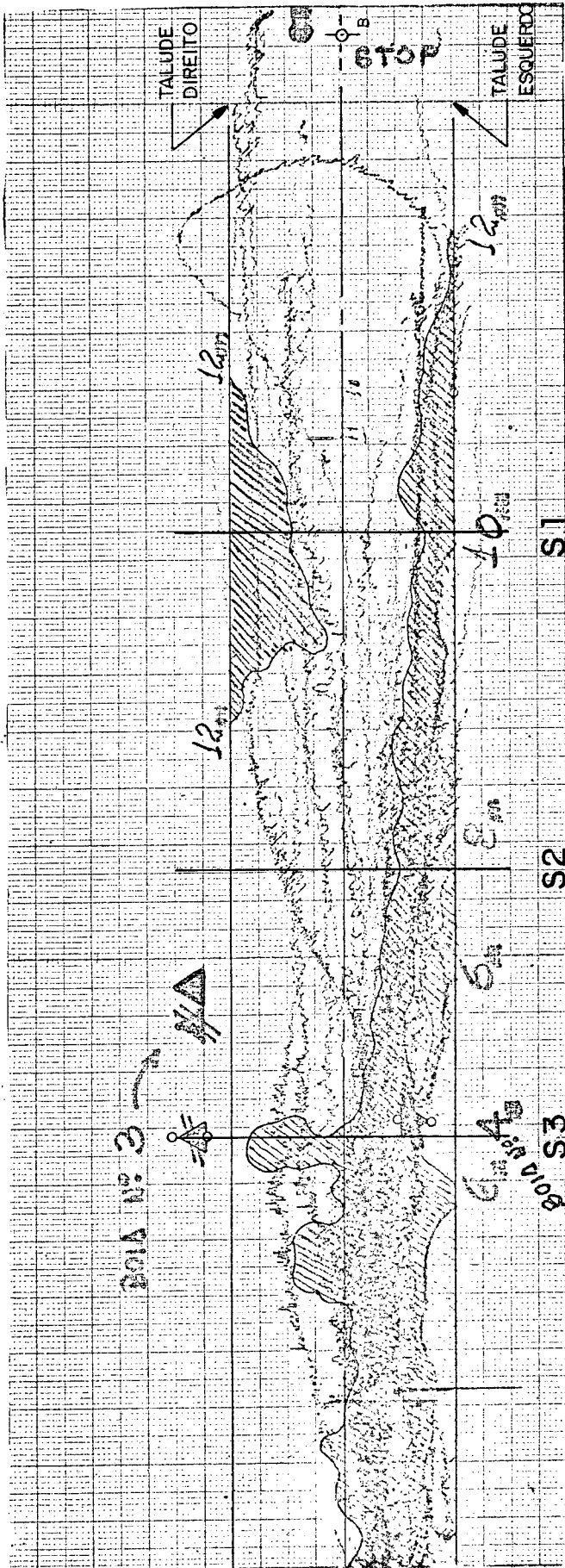


### PLANO DE TRABALHO

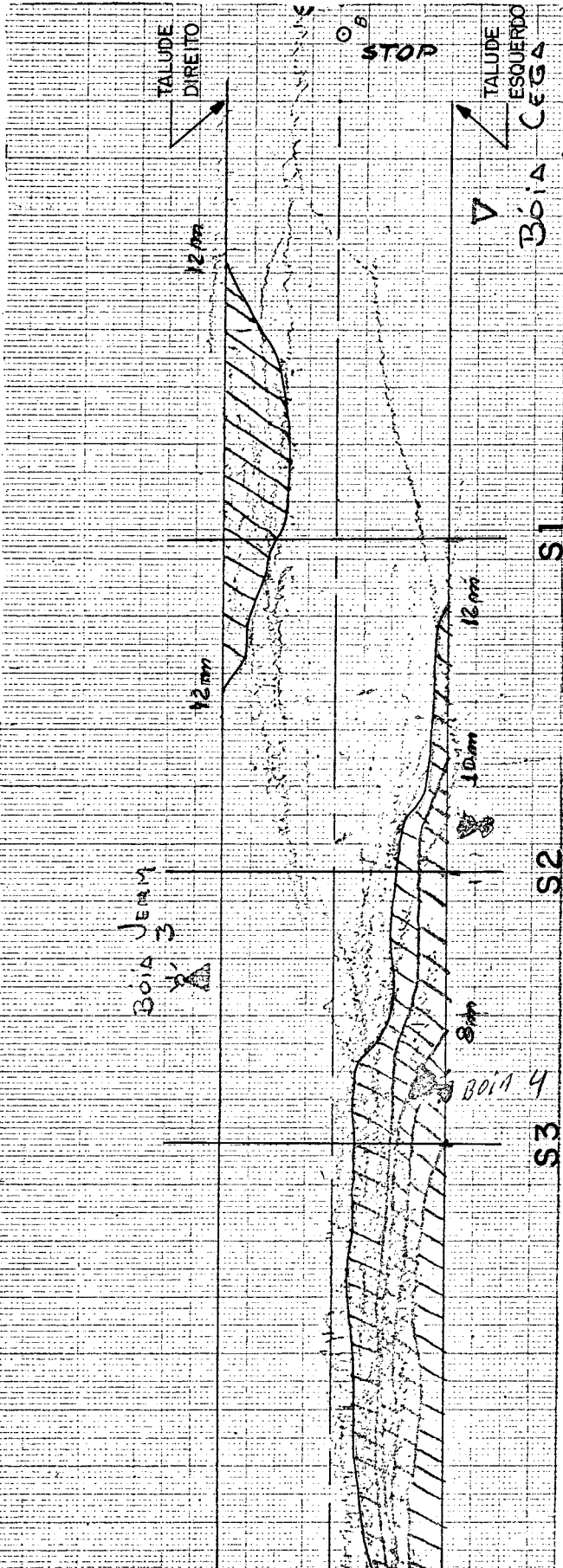
A carta do plotador mostra, com hachuriado, regiões do canal onde se concentrará o ataque do equipamento.

As profundidades registradas em forma de "linhas de igual" dão a prioridade da dragagem.

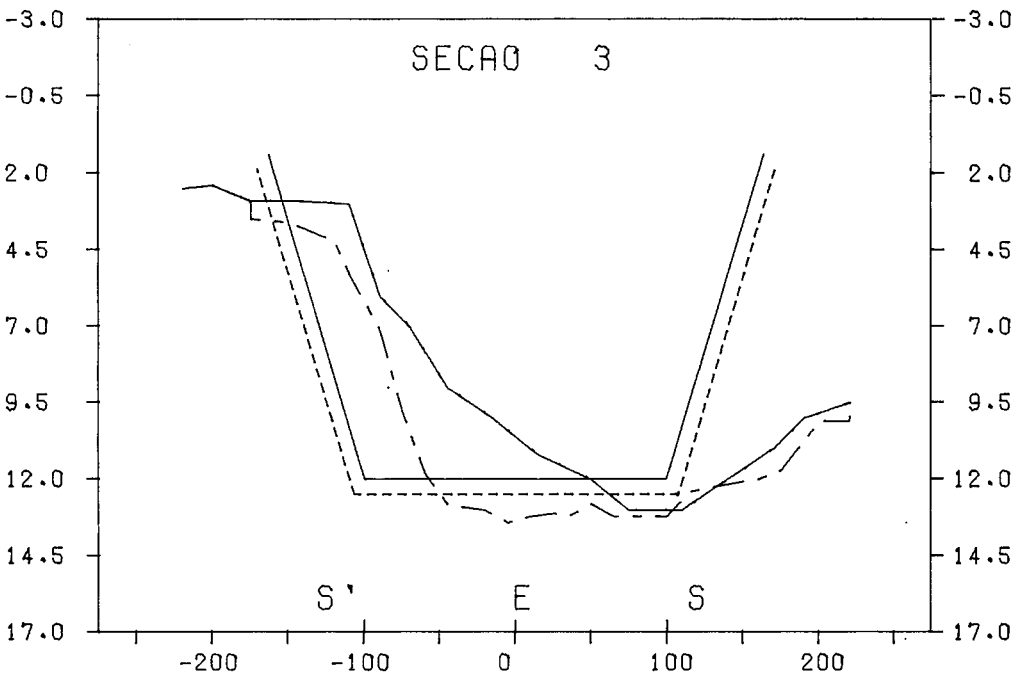
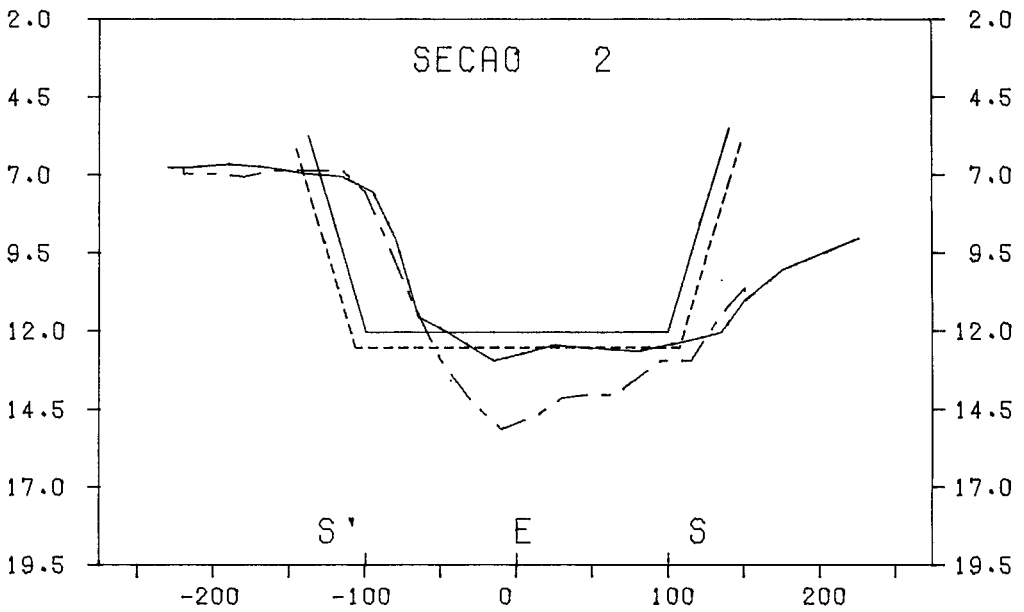
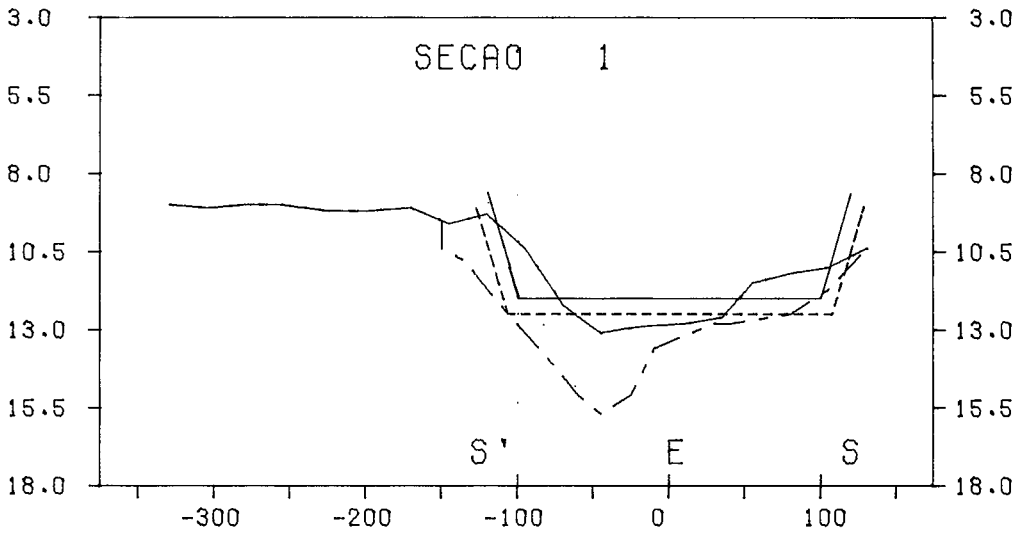
A "linha de 12m" delimita a profundidade objeto dessa dragagem.



Esta 2a. carta pode dar uma idéia do início da obra. Através do registro realizado pelo plotador, podemos verificar as trajetórias da draga, em ataque simultâneo aos taludes esquerdo e direito.



Nesta 3a. Carta, podemos ver o andamento da dragagem, através do recuo das linhas de igual, mostrando já ter sido removida grande parte do material.



## CAPÍTULO VI

### CONCLUSÕES

#### 6.1 - APERFEIÇOAMENTO

O comportamento do sistema SPE, em sua fase preliminar, resultou satisfatório. Não obstante, sugerimos alguns aperfeiçoamentos de SOFTWARE e HARDWARE que, uma vez introduzidos no protótipo, melhorariam sua operação e manutenção:

##### DIGITAÇÃO COM CAMPO PROTEGIDO

Das treze respostas digitadas pelo operador, durante a fase de inicialização, cinco delas (ZA, ZB, ALT, ESCALA, SLARG) não requerem, em seu registro, mais do que três dígitos decimais para defini-las. Assim, o operador é obrigado a preencher seus respectivos campos com, pelo menos, três zeros à esquerda.

Introduzindo-se, no sistema, a digitação com campo protegido, essa perda de tempo não mais aconteceria, uma vez que tais zeros necessários já teriam sido previamente lançados no campo.

##### NOVA ROTINA PLOTA

Face ao método de atualização da pena empregado, a rotina de plotagem (PLOTA) foi desenvolvida sem a previsão de testar o status do Plotter a cada comando que lhe era enviado. Em decorrência disso, o sistema apresenta um grande inconveniente de operação: o plotador não pode ser desligado durante a operação do sistema, sem prejuízo da omissão de alguns steps.



Assim, uma nova rotina PLOTA que testasse o status do plotador a cada step, e, simultaneamente, fizesse a atualização das variáveis de controle do Plotador, resolveria o problema.

#### PROM DE DIAGNÓSTICO DO SISTEMA

Rotinas de auto-diagnóstico poderiam ser incorporadas ao sistema através memória "PROM", com objetivo de testarem processador e periféricos.

Em caso de pane, tais rotinas permitiriam ao operador obter, no próprio local, um diagnóstico expedito do defeito.

#### UMA ARQUITETURA ADEQUADA PARA O CONTROLADOR

Na implementação do HARDWARE do controlador foi utilizada uma estrutura pronta da INTEL (SBC 80/10). Trata-se de uma arquitetura superdimensionada para a aplicação e, como tal, de manutenção cara.

O desenvolvimento de uma arquitetura nacional, mais adequada ao controlador, permitiria uma manutenção mais fácil e barata do sistema - a nível de componente.

## 6.2 - EXTENSÃO

A BATIMETRIA consiste na determinação, ponto a ponto, de profundidades do relevo hidrográfico. Entre nós essa determinação é feita aleatoriamente, face à inexistência de controle de posicionamento. Consequentemente, o modelo digital hidrográfico obtido apresenta a inconveniência de ser irregular.

Sendo o SPE um sistema de controle de posicionamento, seu emprego permitiria ao sondador uma nova metodologia de trabalho: sondagem batimétrica automática pré-determinada.

Tal método de trabalho faz o levantamento hidrográfico do relevo, em secções pré-determinadas, e, automaticamente, o registro, ponto a ponto, dos dados de posição e profundidade de cada secção.

É evidente que o emprego do SPE na batimetria automática vai requerer implementações no SOFTWARE e no HARDWARE: trata-se de aplicação mais sofisticada.

Das implementações de HARDWARE, as mais significativas são os interfaceamentos dos seguintes periféricos: unidade de leitura e gravação K-7, ecobatímetro digital e unidade de vídeo-gráfico.

Quanto às de SOFTWARE, além das rotinas de controle desses novos periféricos, uma nova rotina de posicionamento se faz necessária, considerando a maior flexibilidade exigida nesta aplicação.

### 6.3 - INDUSTRIALIZAÇÃO

O fato do protótipo operar, ininterruptamente, há seis meses, acrescido das facilidades encontradas pela equipe de operação no seu manuseio, leva-nos a concluir que os objetivos preconizados no projeto PSPE - robustez e operacionalidade - foram atingidos.

Como saldo favorável, temos ainda: a performance do tempo de resposta, que é inferior àquela prevista inicialmente, e a resolução do traçado, que superou a expectativa, considerando que não se lançou mão de recursos de aritmética de ponto flutuante.

Corroborando as conclusões aqui apresentadas, a diretoria da C.B.D. deu por aprovado o protótipo e iniciou preparativos para a confecção de unidades industriais, a fim de equipar, definitivamente, suas dragas autotransportadoras.