UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE FÍSICA E QUÍMICA DE SÃO CARLOS

DESENVOLVIMENTO DE UM
ANALISADOR DE ALTURA
DE PULSOS-
EDSON DOS SANTOS MOREIRA

Dissertação apresentada ao Ins-
tituto de Física e Química de
São Carlos, USP, para a obten-
ção do Título de Mestre em Fís-
ica Aplicada.

Orientador: Prof. Dr. Jan Frans Willem Slaets

Departamento de Física e Ciência dos Materiais

BIBLIOTECA DO INSTITUTO DE FÍSICA E QUÍMICA DE SÃO CARLOS - USP
FÍSICA

124

MEMBROS DA COMISSÃO JULGADORA DA DISSERTAÇÃO DE MESTRADO DE _____

Edson dos Santos Moreira

APRESENTADA AO INSTITUTO DE FÍSICA E QUÍMICA DE SÃO CARLOS, DA
UNIVERSIDADE DE SÃO PAULO, EM 12 DE julho DE 1984.

COMISSÃO JULGADORA:

Dr. Jan F.W. Slaets

- Orientador

Dr. Aldo Félix Craievich

Dr. Odelar Leite Linhares

BIBLIOTECA DO INSTITUTO DE FÍSICA E QUÍMICA DE SÃO CARLOS - USP
FÍSICA

12/7

A meus
Pais e Irmãos

AGRADECIMENTOS

- . A Jan Slaets, paciente amigo e Conselheiro, pela confiança cre
ditada.
- . A Ulysses Thomas Borelli Jr., pela ajuda no desenvolvimento do
"Software".
- . A Carlos Antonio Ruggiero e Álvaro Garcia Neto, pelos conselhos
e pela companhia nos momentos difíceis.
- . A Lício Onofre Baptista de Almeida pela ajuda no desenvolvimento
dos sistemas analógicos.
- . A Ailton Batista Alves e José Augusto Lopes da Rocha, pela aju
da nas montagens e documentação.
- . A Valentin Obac Roda pelos conselhos na elaboração da dissertação
ção.
- . Ao Professor Aldo Craievich, José Ramon Beltram e Dimas Rober-
to Vollet, pela ajuda na adequação do projeto à sua aplicação
no laboratório de Raios-X.
- . A Nêuri Lúcia Mendonça Antunes, pela paciência e dedicação no
encaminhamento dos assuntos burocráticos.
- . A Maria Cristina Olivio pela datilografia deste trabalho.

1987 - 1988 - 1989 - 1990 - 1991 - 1992 - 1993 - 1994 - 1995 - 1996 - 1997 - 1998 - 1999 - 2000 - 2001 - 2002 - 2003 - 2004 - 2005 - 2006 - 2007 - 2008 - 2009 - 2010 - 2011 - 2012 - 2013 - 2014 - 2015 - 2016 - 2017 - 2018 - 2019 - 2020 - 2021 - 2022 - 2023 - 2024 - 2025

Este trabalho foi realizado com o apoio
do CNPq, FINEP e SUBIN

ÍNDICE

Lista de figuras	i
Resumo	iii
Abstract	iv
Introdução	1

CAPÍTULO I CONSIDERAÇÕES À TÉCNICA DE ANÁLISE DE ALTURA DE PULSOS

1.1 - Introdução: A importância do Analisador	3
1.2 - Os Detetores	5
1.2.1 - Evolução dos Detetores de Raios-X	5
1.2.2 - O Detetor Proporcional Sensível à Posição	6
1.3 - O Sistema de Aquisição	10

CAPÍTULO II A ELETRÔNICA DO SISTEMA

2.1 - Introdução	13
2.2 - Módulo Controlador Central	16
2.3 - Módulo de Entrada de Sinal	19
2.4 - Módulo de Operação e Monitoração	28
2.5 - Módulo de Comunicação	30
2.6 - A Implementação	32

CAPÍTULO III A PROGRAMAÇÃO DO SISTEMA

3.1 - Introdução	34
3.2 - O Sistema de Desenvolvimento de Programas	34
3.3 - O Sistema Operacional	35
3.4 - O Programa de Controle do Analisador	36
3.4.1 - Rotina de Fundo: "Display"	38

3.4.2 - Rotina Coleta	40
3.4.3 - Rotina de Transmissão de Dados	40
3.5 - O Programa de Recebimento de Dados	43
 <i>CAPÍTULO IV MEDIDAS DE EFICIÊNCIA DO SISTEMA</i>	
4.1 - Introdução	47
4.2 - Teste de Linearidade	50
4.3 - A Determinação do Raio de Giro de uma Amostra .	56
 <i>CAPÍTULO V CONCLUSÕES E POSSÍVEIS MELHORAMENTOS DO SISTEMA</i>	
 <i>APÊNDICE I DIAGRAMAS ELÉTRICOS DOS CIRCUITOS MONTADOS</i>	 65
<i>APÊNDICE II LISTAGEM DO PROGRAMA DE CONTROLE DO AAP</i>	73
<i>APÊNDICE III LISTAGEM DO PROGRAMA DE RECEBIMENTO DE DADOS</i>	103
 <i>REFERÊNCIAS BIBLIOGRÁFICAS</i>	 106

LISTA DE FIGURAS

1 - Diagrama Básico de um Analisador de Altura de Pulsos	4
2 - Esquema do detetor	7
3 - Instrumentação Necessária para Uso do Detetor	8
4 - Tratamento dos Sinais nos Diversos Estágios	9
5 - Sistema Completo de Aquisição de Curvas	11
6 - Sistema Montado	14
7 - Detalhe da Placa Montada	14
8 - Diagrama de Blocos do Aparelho	15
9 - Módulo Controlador Central (Kit)	17
10 - Módulo Controlador Central (expansão)	18
11 - Módulo de Entrada de Sinal	21
12 - Diagrama de tempo do Módulo de Entrada de Sinais	23
12-A Determinação do Tempo Morto do Sistema	27
13 - Módulo de Operação e Monitoração	29
14 - Módulo de Comunicação	31
15 - Diagrama de Fluxo do Programa de Controle do AAP	37
16 - Fluxograma da Rotina Display	39
17 - Fluxograma da Rotina de Coleta de Dados	41
18 - Fluxograma da Rotina de Transmissão de Dados	42
19 - Programa de Recebimento de Dados	44
20 - Curva de Espalhamento - AAP Northern	48
21 - Curva de Espalhamento - AAP L.I.E.	49

22 - Curva de Linearidade - AAP Northern	52
23 - Curva de Linearidade - AAP L.I.E.	53
24 - Curva de Sensibilidade do Detetor	54
25 - Curva LOG[I(s)] Vs $(N-NO)^2$ - Northern	59
26 - Curva LOG[I(s)] Vs $(N-NO)^2$ - L.I.E.	60

RESUMO

Neste trabalho descrevemos o Desenvolvimento de um Analisador de Altura de Pulsos. Este aparelho é essencial no estudo de sinais oriundos de sensores que detetam fenômenos físicos e codificam as informações na amplitude dos pulsos que fornecem na saída.

O sistema compõe-se de um Módulo de Entrada de Sinais conectado a um Módulo de Controle baseado no microprocessador 8085A capaz de memorizar pulsos com até 1 μ S de largura em 256 canais com resolução melhor que 20 mV. Um Módulo de Comunicação dotado de interface serial é usado para transferência de dados para outros dispositivos através do protocolo RS232c. O Módulo de Operação e Monitoração munido de teclado hexadecimal e saída analógica possibilita a visualização das curvas coletadas num monitor XY.

A arquitetura do aparelho e os programas desenvolvidos para este sistema de baixo custo, foram otimizados de forma a produzir um tempo morto típico de aproximadamente 100 μ S.

Como aplicação ele foi utilizado para levantamento de curvas no Laboratório de espalhamento de raios-X a baixo ângulo deste Departamento.

O desempenho do aparelho foi testado através de comparações entre seus dados e os obtidos através de um similar, um PHA Northern modelo NS633, e concluiu-se pela sua eficiência.

ABSTRACT

This work describes the development of a Pulse Height Analyzer. This equipment is essential to analyse data coming from detectors producing information codified in pulse amplitudes.

The system developed consist of a Signal Input Module conected to a Controller Module based on a 8085A microprocessor capable to memorize pulses up to 1 us in 256 channels with a resolution better than 20 mV. A Communication Module with a serial interface is used for data transfer to a host computer using RS232c protocol. The Monitoring and Operation Module consist of a hexadecimal Keybord, a 6 digit 7-segment display and a XY analog output enabling real time visualization of data on a XY monitor.

The hardware and the software designed for this low cost system were optimized to obtain a typical dead time of approximately 100 us.

As application, this device was used to adquire curves at the Small Angle X-ray Scattering Laboratory in this Department.

The apparatus performance was tested by comparing its data with a Northern Pulge Height Analyzer model NS633 output, with favorable results.

INTRODUÇÃO

O projeto, objeto desta dissertação, foi desenvolver um equipamento capaz de fazer análise de sinais oriundos de um detector proporcional sensível a posição [1]. Tais detectores encontram aplicações em análises de espalhamento de baixo ângulo de fibras musculares e tecido nervoso, em medicina nuclear [2], em cristalografia [3], espectroscopia Mossbauer [4], física nuclear [5], etc.

Entre o detector e o Analisador existe um Conversor tempo-em-amplitude que condiciona o sinal do detector numa forma adequada à análise do AAP.

O sinal na saída do Conversor Tempo-em-Amplitude, ou seja, na entrada do AAP, tem as seguintes características:

- Ocorrência aleatória
- Frequência média máxima: 8 KHz
- Altura de Pulsos variando entre 0 e 10 volts
- Largura dos Pulsos: 1 μ S

As características iniciais requeridas do AAP foram:

- 256 canais (8 bits)
- Sistema de visualização das curvas para permitir monitoração da operação
- Sistema de comunicação com o Computador Central do Instituto a fim de agilizar o processo de coleta de dados.

Tendo definidos aqueles parâmetros, e tratando-se do desenvolvimento de um protótipo, optou-se pela utilização de um

kit SDK-85 por sua disponibilidade no momento e pelo fato de que uma parte do sistema, aquela relativa à Unidade Central de Processamento já estar pronta, o que economizaria tempo. Outro fator importante foi sua área de expansão que poderia facilmente alojar os outros Módulos que comporiam o AAP. Decidiu-se também utilizar um conversor A/D integrado de 8 bits e para a visualização das curvas foram usados dois conversores D/A de 8 bits e um tubo de raios catódicos com deflexão eletrostática.

Construiu-se assim um sistema relativamente barato, de fácil manutenção e operação, inclusive com a possibilidade de se programar o tempo de coleta, fazendo com que o sistema de aquisição pare automaticamente depois de um tempo pré-determinado de funcionamento.

CAPÍTULO I

CONSIDERAÇÕES À TÉCNICA DE ANÁLISE DE ALTURA DE PULSOS

1.1 - Introdução: A importância do analisador

A detecção, identificação e espectroscopia de partículas carregadas e fótons energizados (raios-X, raios gama) constituem parte essencial dos campos da física nuclear e da física de partículas, tão bem quanto de numerosos campos que fazem uso da radioatividade (traçadores radiográficos médicos, ciências forenses e inspeção industrial).

O Analisador Multicanal ou Analisador de Altura de Pulsos (AAP) como é conhecido devido a sua principal aplicação, é um instrumento muito importante na realização desta tarefa, e é fabricado por várias indústrias de equipamentos no exterior. A idéia é simples, como mostra a figura 1a. Os sinais a ser analisados são pulsos de amplitude variável que, na entrada do AAP são injetados num circuito detetor de pico. Ligado ao detetor está um conversor Analógico/Digital (CA/D), que gera um endereço relacionado com a altura alcançada pelo pulso na entrada. O conteúdo da posição de memória (canal) selecionada é então incrementado de um, indicando que mais um pulso foi detetado com aquela amplitude ou, o que é equivalente, naquele canal. Como resultado temos um gráfico que é um histograma de altura de pulsos, figura 1b, representando a frequência de ocorrência de pulsos em relação aos canais de chegada.

A grande utilidade dos AAP's reside no fato de que muitos detetores usados nas áreas mencionadas acima fornecem, como saída, pulsos com amplitudes proporcionais à energia da radia-

(a)

(b)

Fig.1 - Diagrama básico de um Analisador de Altura de pulsos.

ção detetada. Assim, um Analisador de Altura de Pulsos converte a saída dos detetores num espectro de energias!

1.2 - Os detetores

Existe uma grande variedade de detetores que fornecem sinais que, analisados por um AAP, podem gerar espectros de energia, de tempo [6] e de posição [7,8,9]. A presente aplicação refere-se a esta última classe de detetores.

1.2.1 - Evolução dos detetores de Raios-X

Para a maioria das experiências em espalhamento de raios-X o tempo de exposição é o fator limitante mais crítico, principalmente pelos problemas de estabilização do feixe incidente [10], ou mesmo no caso de amostras biológicas que são instáveis no decorrer do tempo. Nos últimos anos, esforços têm sido feitos para baixar aquele tempo.

Primariamente a detecção de raios-X foi feita através de cintiladores que, controlados por motores, faziam a varredura da região angular desejada passo a passo [11]. Este método, além de exigir grande tempo de exposição, devido ao fato de as medidas serem feitas ponto a ponto, dificultava a análise de fenômenos cinéticos.

Outro método usado, é a detecção por filme. As limitações dos filmes são bem conhecidas [12]: Por um lado os problemas de escalamento de dados e por outro, longas exposições ainda são necessárias devido ao fato de que um mínimo número de fótons são necessários para obter dados acima da máscara química. Problemas

como o da revelação e leitura dos filmes também devem ser levados em consideração.

A solução para estes problemas foi encontrada nos contadores proporcionais sensíveis à posição de chegada das partículas. Com este método, o tempo de exposição é enormemente reduzido e ele apresenta bem menos problemas que a detecção por filme. Presentemente no Laboratório de Baixo Ângulo, este tipo de detector é usado na análise de espalhamento.

1.2.2 - O detector proporcional sensível à posição

Estes detectores consistem simplesmente de uma câmara geralmente cilíndrica, com poucos centímetros de tamanho, com um fino fio esticado axialmente. São preenchidas com algum gás ou mistura de gases. Num dos lados da câmara localiza-se uma janela, feita de Mylar Aluminizada, transparente à radiação. O fio central é polarizado num potencial positivo de modo que elétrons atraídos em sua direção causam ionização adicional, resultando num sinal maior. Este efeito de multiplicação de cargas tornam os contadores proporcionais úteis em baixas energias de radiação.

A figura 2 mostra esquema do detector. É um PSD 110 da Tennelec [13]. Um par de pré amplificadores especialmente projetados são acoplados capacitivamente aos terminais do fio fora da câmara. Quando um raio-X entrando na câmara é detectado, um sinal aparece na saída de ambos pré amplificadores.

$$V_A = -E(1 - \exp \frac{-2t}{RCx})$$
$$V_B = -E(1 - \exp \frac{-2t}{RC(l-x)})$$

Fig. 2 - Esquema do detetor.

R : Resistência total do fio central

L : Comprimento do fio

A amplitude do sinal gerado nos terminais A e B é proporcional à energia (ou comprimento de onda) do raio-X e o tempo de subida do sinal é proporcional à distância entre o ponto de chegada e o terminal correspondente. A diferença entre os tempos de subida dos 2 sinais indica em que lugar do detetor o raio-X foi detetado. A figura 3 mostra um conjunto de aparelhos necessários para determinar a posição de chegada de um raio-X no detetor e transformá-la numa informação adequada à análise de um AAP.

Para medir a diferença dos tempos de subida dos pré-amplificadores de saída de cada lado do detetor, dois canais de temporização e um Conversor Tempo-em-Amplitude (CTA) [14] são usados. Cada canal de temporização consiste de um amplificador linear formatador ("shaping") e um detetor "crossover" também conhecido como "crossover pickoff". Um canal temporizador é designado "comece" e o outro "pare". Note, no diagrama da figura 4, que o cruzamento ("crossover") da linha de base do amplificador formatador ocorre quando do pico da saída do pré amplificador. O detetor "crossover" por sua vez gera um pulso rápido quando esta condição ocorre. Para um dado evento ionizante, dois sinais são gerados, e

Fig.3 - Instrumentação necessária para uso do detetor.

Fig.4 - Tratamento dos sinais nos diversos estágios.

o tempo entre a ocorrência destes sinais indica em que posição do fio o evento foi detetado. Os pulsos "comece" e "pare" são as entradas do CTA.

Um canal de energia é usado para rejeitar sinais fora de uma banda selecionável de energias (ou comprimentos de onda). Eletronicamente sinais de ambos pré amplificadores são somados, amplificados e formatados em pulsos bipolares. Aqueles pulsos cuja amplitude é menor ou maior que valores numa faixa predefinida são rejeitados pelo analisador discriminador. Sinais aceitos pelo Canal Energia abrem a porta do CTA para que este aceite sinais dos canais de Temporização. Sinais rejeitados pelo Canal de Energia fecham a porta do CTA, fazendo com que as informações dos canais de temporização sejam rejeitados.

A monitoração da informação à saída do TAC é feita pelo AAP. O Analisador Multicanal gera um histograma de contagem Vs voltagem (número do canal). Desde que, nesta aplicação, posição é convertida em tempo, e tempo é convertido em voltagem, o histograma é um espectro de Contagem Vs distância de um dos lados do detetor tomado como referência.

1.3 - O Sistema de Aquisição

A figura 5 mostra o sistema de aquisição montado para levantar curvas de espalhamento de raios-X a partir de detetores proporcionais sensíveis à posição. É composto - além do detetor e da eletrônica para amplificar e formatar os sinais oriundos dele e fazer a conversão tempo-em-amplitude, do Analisador de Altura de pulsos, de um monitor X-Y (tubo de raios catódicos) e utiliza o computador central do Instituto, um VAX 11-780 para armaze

Fig.5 - Sistema completo de aquisição de curvas.

namento das curvas obtidas.

Continuamente a UCP busca, no Módulo de Operação e Monitoração, informações sobre a escala desejada na colocação da curva no vídeo. Através deste módulo também se pode interromper a rotina de fundo para análise quantitativa e informações sobre o tempo restante da coleta de dados. Neste caso é aberta uma linha de conversação com o computador central.

CAPÍTULO II

A ELETRÔNICA DO SISTEMA

2.1 - INTRODUÇÃO

O analisador Multicanal ou analisador de Altura de Pulsos — AAP, visto nas figuras 6 e 7, foi montado num Kit SDK 85 da Intel [15] que contém um circuito básico, a UCP, e uma área para expansão. Para a exibição das curvas em tempo real é utilizado um tubo de raios catódicos, aqui chamado Monitor XY. A figura 8 mostra esquema completo do aparelho.

Neste capítulo descreveremos o circuito existente no Kit e os circuitos desenvolvidos para o AAP.

O módulo Controlador Central é composto de uma UCP que, através de rotinas armazenadas na memória de programas, controlam as atividades do Sistema. Na inicialização, imediatamente após o acionamento da alimentação, o controle é feito pelo programa Monitor. Depois, via teclado, é acionado o programa de controle do AAP. Durante o tempo programado no Relógio, a UCP deve fazer leituras contínuas da memória de dados e, através de portas de saída e conversores D/A, fazer o "refrescamento" da imagem na tela do Monitor XY. O recebimento de um pulso pelo Módulo de Entrada de Sinal deve interromper por alguns microssegundos esta tarefa pois agora a UCP deve fazer o processamento adequado do novo dado. Findo o atendimento àquele dado, deve voltar ao trabalho de mostrar a curva, agora já ligeiramente alterada.

Continuamente a UCP busca, no Módulo de Operação e Monitoração, informações sobre a escala desejada na colocação da curva no vídeo. Através deste módulo também se pode interromper a

Fig.6 - Sistema montado destacando Monitor XY, placa com circuitos de controle e terminal VAX.

Fig.7 - Detalhe da placa montada.

Fig.8 - Diagrama de blocos do aparelho.

rotina de fundo para análise quantitativa e informações sobre o tempo restante da coleta de dados. Neste caso é aberta uma linha de conversação com o computador central.

2.2 - Módulo Controlador Central

Este módulo é basicamente composto pelo circuito oferecido junto com Kit ao qual foi acrescentado memória de programa e de dados suficientes para as necessidades do AAP.

A figura 9 mostra o diagrama relativo a parte do Kit. O processador usado é um 8085A [16] de 8 bits de dados, 16 de endereços e trabalhando com "clock" de 3.072 MHz. O programa Monitor está gravado numa ROM de 2 Kbytes, a 8355 que possui também 3 portas de entrada/saída. Para dados, existem 256 bytes de memória RAM contidos no 8155. Este integrado possui também 3 portas de e/s e um Relógio. Na verdade existe circuito impresso para dois 8355 e dois 8155, mas usamos apenas um 8355 e os dois 8155 (o 8155 opcional será referenciado como 8155-expansão). Também nesta parte ficam os circuitos de demultiplexação do duto de dados e endereços e os alimentadores/isoladores para a ligação com a área de expansão, ligação esta feita através dos conectores J1, J2 e J4. São gerados aqui, os habilitadores de pastilha ("chip select") para os periféricos residentes nesta área. Ocupam os 32 Kbytes mais baixos do endereçamento.

A figura 10 mostra aquela porção do Módulo Controlador que foi montada na área de expansão. O circuito de geração de habilitadores de pastilha fornecem indicadores nos seguintes endereços para os seguintes dispositivos:

80000 a 93FF Memória RAM (existe circuito para 5

Fig.9 - Módulo controlador central.(Kit).

Módulo Controlador Central
(Área de Expansão)

Fig.10 - Módulo controlador central (expansão).

Kbytes)

A000	a	AFFF	Relógio (8253)
B000	a	BFFF	Porta de entrada de dados (8212)
C000	a	CFFF	Porta serial USART (8251A)
D000	a	DFFF	EPR0M contém programas controladores do AAP
E000	a	FFFF	Disponível
F000	a	FFFF	Disponível

O Relógio é alimentado da seguinte forma: Como ele não pode receber pulsos de frequência maior que 2MHz, o "clock" do sistema é primeiro dividido por 48 (30 em Hexadecimal) pelo Relógio do 8155-expansão, dando na saída frequência de 64 KHz que alimenta a entrada 1 do Relógio 8253. O Relógio 1 está programado para dividir a entrada por 64000 (FA00 hex.) gerando na saída uma base de tempo de um segundo que alimenta o Relógio 2 do 8253. Este Relógio é programado com o tempo estabelecido pelo operador e é lido pelo programa a cada ciclo de varredura da memória da rotina que mostra a curva no Monitor XY.

2.3 - O módulo de entrada de sinal

A peça principal deste módulo é um conversor Analógico/Digital, o AD574 de Analog Devices [17]. Seu método de conversão é o de aproximação sucessivas e o sinal a ser convertido deve permanecer estável na entrada durante o tempo de conversão que é, em média 25 microssegundos. Foi montado na configuração "Stand Alone" exigindo uma porta para acoplá-lo ao Módulo Controlador.

Nesta configuração faz conversão em 12 bits, apesar de serem utilizados apenas os 8 bits mais significativos.

Os pulsos na entrada do módulo chegam aleatoriamente, sem frequência fixa, mas em média numa velocidade nunca superior a 8 KHz. A duração dos pulsos é de 1 microssegundo e sua amplitude varia entre 0 e 10 Volts.

O fato de os pulsos terem apenas 1 microssegundo de duração e o conversor precisar de nível estável por 25 microssegundos, gerou a necessidade de se colocar, como se vê na figura 11, uma memória analógica na entrada do Módulo. Esta memória, também conhecida como Amostrador/Retentor ("Sample/Hold") é composta de uma chave analógica — que é aberta para amostragem do sinal, de um capacitor de retenção e de um amplificador seguidor na saída com alta impedância de entrada. Quando aberta a porta de controle, o sinal na entrada do Amostrador é carregado no capacitor. Quando fechada, a chave tem também alta impedância de modo que o capacitor não é descarregado nem via entrada, nem via saída, por um eventual leitor da memória.

O fato dos pulsos chegarem aleatoriamente gera o problema de se decidir quando abrir a porta de controle. Este problema foi resolvido usando o próprio pulso para disparar a abertura da porta. A entrada de sinais está ligada a um comparador que satura tão logo a entrada sai de zero Volts, ou seja, chega um pulso. No entanto, a saída do comparador não poderia ser usada diretamente para abrir a porta devido ao fato de que ele fecharia alguns nanossegundos depois da saída do pulso da entrada (devido ao pequeno atraso gerado pelo comparador), tempo suficiente para que o capacitor começasse a se descarregar via chave. Isto foi resolvido fazendo com que a subida do pulso na saída do comparador dis

Módulo de Entrada de Sinal

Fig.11 - Módulo de entrada de sinal.

parasse outro pulso de duração menor, 700 nanossegundos, para abrir a porta de controle.

A conversão é iniciada também com um pulso de 700 nanossegundos gerado a partir do pulso que abre a porta de controle. Durante o tempo de conversão, a entrada do AD574 não deve ser alterada e suas saídas ficam em alta impedância. O fim de conversão pode ser detetado monitorando-se o pino de Status do conversor. Este sinal é usado para ativar a leitura do conversor pela porta 8212 que também gera uma interrupção (RST 6.5) que aciona a rotina de coleta de dados para a memória central do AAP.

Um esquema de tempos bem definido é essencial para um bom desempenho do sistema. A seguir analisamos o diagrama de tempo do Módulo de Entrada de Sinal, figura 12. Ali temos mostrados, na primeira linha, 6 pulsos (P1 a P6) de amplitudes diferentes e a forma como eles são tratados. Note-se, na última linha, que dos 6, cinco chegam a entrar no Amostrador/Retentor e destes apenas 3 efetivamente são convertidos. Os tempos referenciados na figura como t_0 a t_7 são:

$t_0 = 1 \text{ uS}(\text{min})$	$t_4 = 25 \text{ uS}$
$t_1 = 1 \text{ uS}$	$t_5 = 100 \text{ nS}$
$t_2 = 700 \text{ nS}$	$t_6 = 33,5 \text{ uS}$
$t_3 = 700 \text{ nS}$	$t_7 = 575 \text{ nS}(\text{max})$

O esquema de rejeição de pulsos que porventura cheguem antes do fim do processamento do anterior, é comandado pelo pino de status do AD574 e de interrupção do 8212. A saída status age sobre o Monoastável 1 fazendo com que este não abra a porta do

Fig.12 - Diagrama de tempo do módulo de entrada de sinal

Amostrador/Retentor durante o tempo em que o conversor estiver ativo (área hachurada na terceira linha. Note que apenas P6 não passou por este critério). Já o pino Interrupção age sobre o Monoastável 2 que gera o pulso negativo de início de conversão, quarta linha. Por este critério não passaram P2 e P4. Veja, na última linha, que estes pulsos foram memorizados, mas não convertidos. Note também que o pulso que deveria ser gerado pelo monoastável 2 relativamente a P6 não o é, apesar do monoastável não estar inibido. É que seu disparador, monoastável 1 nem chegou a gerar seu pulso na saída, conforme visto acima.

O esforço seguinte é feito na determinação do tempo morto do sistema, ou seja, o tempo durante o qual pulsos chegando na entrada do Módulo de Entrada de Sinais não chegam a ser convertidos.

O circuito utilizado neste módulo gera uma interrupção na rotina de fundo toda vez que um dado fica disponível na porta de entrada (8212 usada no modo "interrupting input port"). Esta interrupção fica ativa até que o processador providencia a leitura da porta, sinalizado no diagrama da figura 12 pela linha "habilitação de pastilha para leitura". A leitura da porta de entrada é uma das primeiras tarefas da rotina de atendimento para coleta, livrando o Módulo para outra conversão. No entanto, não é permitido o encadeamento de interrupções de forma que este novo pulso convertido deve esperar o total processamento do pulso anterior para ser lido. Em outras palavras, quer se dizer que a tarefa de conversão pode ser sobreposta à de leitura. Como geralmente a conversão é mais curta, o tempo gasto na leitura e processamento do pulso é quem define o tempo morto do aparelho.

Devido a arquitetura de interrupções disponível no

Kit, o caminho que percorre o processador desde que recebe um sinal de interrupção até atendê-la é tortuoso. Quando da percepção, por "hardware" de um pulso no seu pino de interrupção, o processador espera o fim da instrução atualmente executada (que pode variar entre 4 e 18 estados - um estado num sistema de 3 MHz equivale a 330 nanossegundos), executa automaticamente uma instrução "call" de aproximadamente 6 estados e salta para o endereço reservado para o atendimento inicial - no caso da RST 6.5 o endereço é 0034 hexadecimal. Como o Kit é de uso geral e neste endereço temos o Programa Monitor, gravado em ROM, naquele endereço existe uma instrução de salto para uma posição reservada na RAM de serviço. Endereço 20C8 hex. Na inicialização, é carregado nesta posição, o endereço da rotina coleta, D41E hex juntamente com outra instrução de salto. São então que o processador começa a executar a rotina de atendimento de interrupção. O tempo de execução destas três instruções de "JUMP" é de 30 estados. As instruções no início da rotina de atendimento para coleta — vide item 3.4.2 e apêndice II, gastam outros 52 estados que somados aos 30 dos saltos e os 6 do "CALL" perfazem 88 estados. Dependendo da instrução que estiver sendo executada no momento da interrupção o tempo total deve variar entre o mínimo de 92 e o máximo de 106 estados, ou em média 99 estados que equivalem a 33 microssegundos.

O tempo de 33 microssegundos seria o tempo morto do sistema se a única tarefa da rotina coleta fosse ler o dado da porta. É necessário, no entanto, gerar um endereço, fazer uma leitura, uma atualização e um armazenamento na memória, sendo acrescentado algum tempo a mais àquele. Uma olhada na rotina em questão mostra que ela pode terminar em três pontos distintos:

1 - Quando o conversor fornece 00 ou FF. Um dos gran

des problemas enfrentado nas primeiras versões do aparelho foi a observação de um crescimento demasiado rápido na contagem dos canais inicial e final. Verificou-se que a aberração era causada pela alta sensibilidade do circuito comparador da entrada, que disparava a conversão de todos os sinais, mesmo os que estavam abaixo da transição 00000000 para 00000001. O resultado desta conversão era sempre 00. Já o defeito do canal 256 era devido ao fato de os amplificadores que fornecem os pulsos para o AAP geram tensões superiores a 10 volts (canal 256). Se a curva sendo captada contivesse contagens significativas próximas ao referido canal, o aparelho apresentava a alteração no último canal.

Dada a dificuldade de se ajustar tanto o ganho do circuito que gera os sinais para o AAP, quanto a sensibilidade do comparador, resolveu-se eliminar os dois canais, o que em absoluto altera as características da curva. Detetando um destes valores a rotina retorna ao ponto de interrupção sem fazer nada, gastando mais 133 estados (já se levou em conta, neste cálculo o tempo gasto na recuperação dos registradores no fim da rotina). Somados aos 99 iniciais perfazem o tempo de 76 microssegundos.

2 - A segunda situação, que é a típica do sistema é quando chega pulsos em qualquer outro canal cujo conteúdo não tenha ultrapassado 65535 contagens (a soma normal é feita em 16 bits). Neste caso devem ser somados aos 99 estados iniciais mais

O tempo morto varia entre 76 e 113 us dependendo da contagem acumulada no canal acessado

Fig. 12-A - Determinação do tempo morto do sistema.

206, o que dá 100 microssegundos.

- 3 - Este é o caso extremo quando torna-se necessário utilizar o terceiro byte da memória para a atualização do conteúdo do canal. Aí deve-se somar mais 245 estados dando 113 microssegundos.

A figura 12-A mostra diagrama utilizado na determinação do tempo morto.

2.4 - Módulo de operação e monitoração

A monitoração e operação do sistema é feita, parte pelas facilidades oferecidas pelo Kit SDK e parte por meios criados especialmente para o projeto. A figura 13 ilustra este Módulo. Os meios de comunicação Operador/Máquina são os seguintes:

- Um monitor XY. É um tubo de raios catódicos com deflexão eletrostática dotado de controle de brilho, controle de escala, e controle de posicionamento da curva na tela.

A memória de dados é lida ciclicamente e os dados transferidos para o Monitor via portas de saída (8155) e conversores D/A (1408), um para cada eixo. O controle desta transferência é feito pela rotina Display (vide item 3.4.1). Oito bits são escolhidos dentro da palavra de 24, e são injetados nos conversores D/A que geram uma tensão equivalente ao seu valor, tensão esta que faz a deflexão do feixe de elétrons do Monitor. A amostragem dos dados para exibição na tela começa pelos oito bits menos significativos. A rotina display faz leituras periódicas da tecla ESCALA, e quando esta está pressionada, a cada leitura os dados na tela do Monitor aparecem divididos por 2. O conteúdo da memória, porém, não é al-

Fig.13 - Módulo de operação e monitoração.

terado. Os circuitos de controle do Monitor estão na Área de expansão.

- um conjunto Teclado/Display. O teclado é hexadecimal contendo, além das teclas dos 16 dígitos, tecla para inicialização (RESET), carga de programas (SUBST MEM), execução de programas (GO e EXEC), etc. Uma descrição mais detalhada das funções destas teclas encontram-se no capítulo III.

Os circuitos de controle do teclado e display estão na área do Kit (circuito impresso). O controle é feito por um 8279 ("Keyboard/display Controller") e o acionamento de uma tecla resulta na geração, por aquele dispositivo de uma interrupção (RST 5.5) que, por sua vez executa uma rotina de leitura de teclado. O display é de 7 segmentos com seis dígitos. Pode ser acionado via rotinas do Programa Monitor e é útil na troca de mensagens (erro, andamento da execução de programas, etc) e na conferência dos dados que entram pelo teclado (endereço de programa a ser executado, opção de funcionamento, tempo programado, etc).

2.5 - Módulo de Comunicação

Este módulo compõe-se basicamente de uma interface de Comunicação Serial [18] usando o protocolo padrão RS232c, adotado pela maioria dos computadores. É implementado através de uma USART ("Universal Synchronous/Asynchronous Receiver/Transmitter") 8251A que está programada com as seguintes opções:

- 9600 bauds
- 8 bits de dados
- 2 bits de parada
- Relógio 64x velocidade de Transmissão.

Módulo de Comunicação
(Área de Expansão)

Fig.14 - Módulo de comunicação.

Como visto na figura 14, a ligação com o VAX é feita através de alimentadores e receptores de linha.

A fim de otimizar a utilização das linhas do VAX, apenas uma linha foi destinada para servir tanto o AAP quando da transmissão de dados, quanto ao terminal VAX, que tem as funções normais de um terminal (editar, compilar e executar programas, imprimir arquivos, acessar periféricos em geral, etc). Aliás, a alocação de uma linha apenas para o AAP seria um despropósito, dentro do quadro de demanda atual, já que a aquisição de uma curva pode levar horas e a linha só é utilizada na transferência final dos dados para armazenamento em disco no VAX, tarefa de aproximadamente 5 segundos. O compartilhamento da linha entre o AAP e o terminal VAX foi possível tomando alguns cuidados no programa de recebimento de dados (apêndice III) que é rodado no VAX (por exemplo, os dados são passados de uma vez e arquivados. Finda a transferência, o arquivo é lido e só aí os dados são colocados na tela do terminal VAX).

2.6 - A implementação

A placa do SDK-85 possui uma área para expansão onde foram montados os circuitos dedicados ao AAP. Estes circuitos foram implementados usando a técnica de "Wire-Wrap". Esta técnica é muito adequada ao desenvolvimento de protótipos de circuitos digitais porque, além da rapidez na montagem dos circuitos, oferece grande facilidade nas possíveis alterações.

Alguns cuidados foram tomados, principalmente quanto a eliminação de ruídos [19,20], que foi fonte de maior parte dos problemas encontrados durante o desenvolvimento do projeto. Todo circuito integrado foi desacoplado com capacitor de 0.1 uF, para evitar ruídos devido aos chaveamentos dos diversos dispositivos.

Outro fator importante considerado foi a implementação de um efetivo plano de terra evitando-se a ligação dos vários terras em cascata. O procedimento ideal é ligar cada terra de dispositivo no terra da fonte.

As alimentações dos circuitos digitais foram separadas das alimentações dos circuitos digitais.

Cuidado foi também tomado no perfeito casamento de impedâncias entre o AAP e a fonte dos sinais a fim de evitar reflexão.

CAPÍTULO III

A PROGRAMAÇÃO DO SISTEMA

3.1 - INTRODUÇÃO

Na descrição da parte eletrônica do sistema fizemos referência à parte provida pelo Kit da Intel e à parte desenvolvida neste projeto e montada na área de expansão do Kit. Neste capítulo apresentaremos brevemente o programa Monitor fornecido juntamente com o Kit, o programa de controle do analisador de Altura de pulsos, gravado em EPROM e mapeado na área de expansão - apêndice II, e o Programa de Recebimento de Dados armazenado em disco no Computador Central, um VAX 11, apêndice III.*

3.2 - O Sistema de desenvolvimento de programas

Para o desenvolvimento dos programas, isto é, sua edição, montagem e depuração, o AAP foi ligado ao VAX 11 para podermos utilizar suas facilidades. Por outro lado, foi utilizado no AAP um monitor alternativo [21] com a finalidade de tornar a comunicação mais eficiente. Os programas eram editados e montados no VAX. Depois, através de um "LOADER" eles eram carregados no Micro. A gravação de versões já depuradas em EPROM eram feitas por um módulo acoplado ao micro.

No estágio final de desenvolvimento foi utilizado o MICROLIE [22] realizando as mesmas funções do VAX, com a diferença de que ele já possui um módulo de gravação de EPROM o que versatiza sobremaneira esta tarefa.

Os programas todos foram escritos em linguagem assembler 8085. Grande esforço foi feito no sentido de se otimizar os

programas, principalmente em termos de tempo. Neste sentido foram essenciais os resultados obtidos com algumas instruções mais fortes providas pelo 8085 mas não divulgadas pela Intel [23].

3.3 - O sistema Operacional

A placa é provida de um programa monitor de 2 Kbytes que se encarrega de tomar as decisões tão logo o sistema é alimentado. Este programa contém rotinas de inicialização dos periféricos do Kit, de tratamento do teclado, além de algumas rotinas utilitárias tais como: gerar atrasos, colocar endereços ou dados no display, possibilitar depuração de programas, emitir mensagens de erro, carregar programas na memória, etc.

O teclado do SDK é composto por 16 teclas que formam o conjunto hexadecimal e mais algumas que acionam rotinas residentes no Monitor. Estas teclas são:

RESET - Dispara procedimento de inicialização. Coloca o contador de programa em 0000. Quando o sistema é alimentado é gerado um RESET interno.

G0 - Tecla que possibilita a transferência de controle do Monitor para um endereço qualquer dentro do espaço de endereçamento do Micro. O acionamento da tecla G0 seguido de um endereço e de EXEC faz com que o contador de programa assumo o novo endereço e o micro passa a executar as instruções contidas a partir daquele endereço.

SINGLE STEP - É um modo alternativo de transferência de controle do sistema e é utilizado para depurar programas em desenvolvimento. Ao contrário da tecla G0, a cada instrução executada, o controle volta para o monitor que fornece ao operador meios

de verificar os efeitos causados pela instrução, por exemplo através da tecla EXAMINE REG, e continuar a execução através da tecla NEXT.

SUBST MEM - Esta tecla permite o armazenamento de programas na memória RAM, a partir de um endereço estabelecido.

VECT INTR - Está ligada diretamente no pino RST 7.5 da UCP. Como as descritas acima, esta tecla aciona uma rotina. A particularidade é que esta rotina não está definida no monitor e pode ser colocada pelo usuário na região de memória que melhor lhe convenha.

Dos 4 pinos de interrupção possíveis de ser utilizados diretamente pelo usuário (exclua-se o pino INTR que exige um controlador de interrupção associado a ele), dois são utilizados pelo sistema SDK. São o TRAP, usado para controle do SINGLE STEP e RST 5.5 utilizado pelo controlador de teclado e display. Os outros dois, RST 6.5 e RST 7.5 são utilizados pelo programa do analisador de altura de pulsos. RST 6.5 é acionado eletronicamente pela porta de entrada de dados toda vez que o conversor A/D tem um dado pronto para enviar para a Unidade Central de Processamento. RST 7.5, que está ligada a tecla VECT INTR é utilizada pelo operador para transferir curvas coletadas para o VAX.

3.4 - O programa de controle do analisador

O programa de controle do AAP, visto no diagrama de fluxo da figura 15, é acionado via teclado (tecla G0). Começa com a inicialização de variáveis, programação dos periféricos utilizados pelo sistema, e tratamento inicial das interrupções. Segue a isto a rotina OPÇÃO que espera que o operador decida sobre uma

Fig.15 - Diagrama de fluxo do programa controlador AAP.

das duas alternativas disponíveis de funcionamento do aparelho. A parece no display OPTE.

Teclando "D" - aparece DISP, estará decidindo pela opção "Apenas Display". Haverá um salto para uma rotina que varre a memória de dados colocando seu conteúdo no Monitor XY. Nenhuma alteração no conteúdo da memória ocorre nesta opção.

Por outro lado, teclando "C" - aparece DIFR, o salto será para uma rotina cuja base é a mesma da opção anterior. A diferença é que antes de entrar na rotina propriamente dita é ativa da a interrupção de coleta de dados (RST 6.5). Assim, a rotina "Display" roda em fundo e é interrompida toda vez que um dado está pronto na saída do conversor A/D. Tanto numa quanto noutra opção, o programa de fundo também poderá ser interrompido para enviar dados para o VAX, teclando-se VECT INTR (aparece CARR).

Uma vez terminada a transferência de dados para o VAX, o micro entra novamente em estado de opção. Isto possibilita que coletas sejam interrompidas para análise quantitativa parcial de dados e depois continuadas.

3.4.1 - Rotina de Fundo: "Display"

Esta rotina, figura 16, foi desenvolvida originalmente por Ulisses Borelli Thomas Jr. e já foi utilizada em outros sistemas [24]. É aqui utilizada com pequenas modificações. Seu trabalho é varrer sequencialmente a memória de acumulação fazendo leituras e colocando o conteúdo nas portas de saída para serem mostradas pelo Monitor XY. Como para cada canal é reservado uma palavra de 24 bits e o conversor D/A utilizado na saída é de 8 bits, uma característica importante desta rotina é que ela permite ao operador escolher os 8 bits mais significativos da palavra através

Fig.16 - Fluxograma da rotina display.

da chave "ESCALA" colocada no painel.

Na opção "COLETA" a rotina DISPLAY começa com a programação do tempo prē determinado de coleta de dados. A cada varredura da memória a rotina lê o temporizador e findo aquele tempo ela se coloca em modo "Apenas Display". A inscrição no Display de 7 segmentos no painel do aparelho muda de DIFR para DISP.

3.4.2 - Rotina Coleta

A rotina coleta esquematizada no diagrama de fluxo da figura 17, é acionada via interrupção RST 6.5. Começa com a desativação de todas as interrupções, salvamento de Status e registradores e leitura da porta ligada ao conversor A/D. Aqui o par de registradores H-L é carregado com a posição inicial (canal 1) da memória de acumulação. O par D-E contém o deslocamento que somado à posição inicial, fornecerá o endereço efetivo do canal. O valor fornecido pelo conversor é transferido para o registrador E e D recebe 00. Como para cada canal são reservados 4 bytes, D-E é multiplicado por 4 - através de 2 "shifts" à esquerda a fim de que contenha o real deslocamento a ser somado com H-L. Uma vez obtido o endereço efetivo do canal, seu conteúdo é lido, incrementado de 1 (um) e novamente armazenado.

Para efeito de simplicidade foram reservados 4 bytes para cada canal. Sō são utilizados, no entanto, 3 bytes podendo acumular até 16.777.216 eventos por canal. Também a detecção de 00 e FF são desconsiderados, sendo que os canais relativos a eles são zerados.

3.4.3 - Rotina de Transmissão de Dados

Como declarado anteriormente, esta rotina — figura

Fig.17 - Fluxograma da rotina de coleta de dados.

Fig.18 - Fluxograma da rotina de transmissão de dados.

18, é acionada pelo operador através da tecla VECT INTR. Antes de ser acionada, no entanto, deve ser rodado o programa de Recebimento de Dados no VAX. Dependendo do estado da variável FLAG1, a rotina decide se tem que fazer ou não tratamento de tempo. Se FLAG1 for zero, significa que o tempo programado já esgotou ou o temporizador não foi acionado (acontece quando da opção Apenas Display). Por outro lado, se FLAG1 for diferente de zero, a rotina deve guardar o tempo decorrido até a interrupção para, depois da transmissão, e se for optado novamente por "Coleta", poder reprogramar o relógio, descontando o tempo de parada. Feito isto são programados o relógio de 8155 (expansão) e a USART 8251. A transmissão é feita em 9600 bauds.

O primeiro dado transmitido é o tempo transcorrido. Depois são enviados os dados em linhas de 4 bytes que são lidos da memória e convertidos em código ASCII. Cada linha é terminada por <CR> e <LF>. No fim é enviado o código padrão de fechamento de arquivo: CNTRL Z. A "conversação" entre os dois sistemas — VAX e AAP — pode ser sincronizada através dos códigos de parada XOFF, ou de continue XON que são interpretados pelo AAP.

No fim da rotina o programa fica à espera da opção do operador que deve escolher "D" caso a coleta esteja terminada ou "C" se quiser continuá-la. Neste caso a rotina providencia a reprogramação do relógio com o tempo que restava no momento da interrupção.

3.5 - O programa de recebimento de dados

Este programa, apêndice III, esquematizado na figura 19 é escrito em FORTRAN [25] e rodado no VAX 11. Providencia o re

Fig.19 - Programa de recebimento de dados.

cebimento dos dados coletados e enviados pela rotina de transmissão rodada no AAP. Como visto anteriormente, a qualquer momento pode-se fazer uma transferência para o VAX.

Os dados são processados e armazenados no AAP no formato binário inteiro. A rotina de transmissão converte-os para ASCII hexadecimal antes de serem enviados. O programa de recebimento os armazena no formato binário novamente — variáveis A, B, C. Depois é feita uma conversão para inteiro através de EQUIVALENCE entre aquelas variáveis e outras (TEMPRO, TEMRES, DADO) tipo inteiro.

A fim de otimizar o aproveitamento das linhas do Computador Central foi utilizada apenas uma linha para ligar o AAP e o terminal-VAX ao VAX. Para acionar o programa de recebimento, deve-se primeiro conectar o terminal-VAX ao VAX. Executando o programa aparece na tela do terminal um cabeçalho e uma indagação sobre o nome do arquivo que se quer abrir para guardar os dados coletados. Uma vez definido o nome, a chave comutadora deve ser chaveada para AAP e acionado o programa de transmissão de dados — através de VECT INTR. A conversação está entabulada.

O programa de recebimento abre um arquivo temporário, que pode ser tornado permanente no final do processo. Nele são gravados os dados recebidos. Uma vez terminada a transmissão, — sinalizada pelo aparecimento de OPTE no display do Kit, a chave comutadora deve voltar à posição terminal-VAX. Neste ponto o programa de recebimento está esperando um caracter de controle (C) para continuar sua tarefa. Esta consiste em ler o arquivo e colocar na tela do vídeo as informações recebidas: tempo programado, tempo restante e as contagens de cada canal. Depois de mostrados os conteúdos dos 256 canais o operador deve informar se quer ou

não que o arquivo fique permanente no disco.

CAPÍTULO IV

MEDIDAS DE EFICIÊNCIA DO SISTEMA

4.1 - INTRODUÇÃO

Vamos aqui mostrar alguns testes feitos com o AAP que nos garante seu desempenho na execução das tarefas para as quais ele foi projetado. As comparações foram feitas em relação aos resultados obtidos por um AAP da Northern Scientific, Inc. modelo NS633 [26]. As figuras 20 e 21 mostram duas curvas de espalhamento de carvão vítreo exposto à radiação de cobre nas mesmas condições — 2400 segundos, 40 Kvolts, 35 mA, e obtidas com os dois aparelhos. O corte abrupto no centro das curvas é devido à colocação de um "beam stop" na região próxima do ângulo zero para proteção do detetor. Ressalte-se aqui que o aparelho da Northern referido acima, não possui, ao contrário do AAP-LIE, controle de tempo, tarefa que deve ser realizada manualmente pelo operador.

Análises diretas das curvas de difração coletadas não pode nos fornecer dados comparativos relevantes sobre o desempenho dos dois aparelhos, principalmente por dois motivos:

- 1 - O fato de os pulsos chegarem à entrada do AAP aleatoriamente e que cada aparelho tem um tempo morto (tempo gasto na conversão e processamento de cada pulso, durante o qual sinais que chegam são rejeitados) causa uma leve diferença na contagem de cada canal e isto torna as coordenadas levemente diferentes para um mesmo canal, e esta diferença varia com a variação da velocidade média de entrada de sinal (a velocidade varia, por exemplo, com a potência do feixe de raio-X incidente).

Fig.21 - Curva de espalhamento de carvão vítreo obtida
com AAP - L.I.E.

2 - O segundo motivo é a diferença na resolução de posição entre os dois sistemas e está relacionado com a "largura" da curva. A observação atenta das curvas 20 e 21 revela que a curva relativa ao AAP-NORTHERN é mais expandida que a do AAP-LIE. A região angular relativa aos canais cobertos por L nas figuras deve ser a mesma pois é delimitada por dois parâmetros geométricos que são o "beam stop" por um lado e o fim do detetor por outro (a montagem foi feita propositalmente para forçar esta situação). No entanto, o número de canais cobertos por L do LIE é diferente do número de canais cobertos por L do NORTHERN. Esta resolução de posição é um dado que leva em conta o modo como são feitos os acoplamentos entre os diversos módulos envolvidos na captação do pulso-detetor, amplificadores, conversor tempo-em-amplitude e conversor A/D, e define exatamente a relação entre determinada região no espaço dos canais no AAP e seu equivalente no espaço das posições no detetor.

Dois métodos alternativos foram utilizados para comprovar o desempenho do AAP-LIE: O primeiro, através da obtenção da curva de linearidade do detetor; o segundo, analiticamente, através do processamento dos dados colhidos para as curvas das figuras 20 e 21, determinando o raio de giro das partículas bombardeadas por raios-X. No segundo caso queremos demonstrar que o cálculo de um parâmetro que só depende da estrutura da amostra (e não do equipamento), coincide quando feito com os dados de ambos aparelhos.

4.2 - Teste de Linearidade

O teste de linearidade é feito deslocando o feixe in

cidente de raio-X, sem amostra, e verificando a correspondência linear no deslocamento do canal sendo afetado no AAP. Esta verificação nos garante que a informação contida nos pulsos fornecidos pelo detetor não são distorcidos nas várias transformações por que passa.

Aqui, o método usado para se conseguir a curva de linearidade do sistema foi o seguinte:

- Posiciona-se o detetor de tal forma que recebe o feixe de raios-X de suas extremidades;

- Regula-se a fonte de raios-X de forma a dar um feixe direto (sem amostra) com intensidade suficientemente baixa (25 kvolts, 10 ma, com filtro) a fim de não danificar o detetor.

- Determina-se um intervalo e faz-se a varredura passo a passo deslocando o detetor até a outra extremidade, fazendo coletas iguais (100 segundos) em cada ponto. Em cada parada anota-se o número do canal que atingiu o pico de contagem e a posição do detetor.

Aproveitamos também para levantar a curva de sensibilidade do detetor através do registro do número de contagens em cada pico.

A figura 22 mostra o gráfico de linearidade obtido com o sistema NORTHERN. Na figura 23 temos a mesma curva obtida com o sistema LIE. A curva de sensibilidade do detetor, figura 24, foi obtida com o sistema LIE. Destas curvas tiramos algumas informações bastante interessantes:

- 1 - Região de sensibilidade do detetor. A região de sensibilidade útil do detetor, é de 75mm como se vê na figura 24,

Fig.22 - Curva de linearidade obtida com AAP - Northern.

Fig.23 - Curva de linearidade obtida com AAP - L.I.E.

Fig.24 - Curva de sensibilidade do detector.

e que está de acordo com a especificação do fabricante.

2 - Dos 75 mm úteis do detetor, o AAP-LIE consegue captar sinais de 72 mm, enquanto o AAP-NORTHERN cobre 60 mm. Como os dois sistemas trabalham com 256 canais, percebemos que, para uma mesma região angular, o NORTHERN tem maior resolução que o LIE. No entanto, o LIE consegue cobrir praticamente todo espaço útil do detetor, enquanto o NORTHERN só cobre 60 mm.

3 - A resolução de posição é a seguinte para os dois sistemas:

$$RESLIE = \frac{\Delta \text{ POSIÇÃO}}{\Delta \text{ CANAL}} = \frac{72}{243} = 0,298 \text{ mm/CANAL}$$

$$RESNOR = \frac{\Delta \text{ POSIÇÃO}}{\Delta \text{ CANAL}} = \frac{60}{248} = 0,241 \text{ mm/CANAL}$$

4 - Está claro que a disparidade na resolução está relacionada com o ajuste da amplitude dos sinais e do fundo de escala do conversor analógico/digital. O AAP-LIE cobre quase exatamente a região do detetor porque os 10 volts de fundo de escala de seu conversor praticamente coincide com os valores oriundos do detetor que identificam os canais superiores (perto de 256). Já no caso NORTHERN o seu fundo de escala coincide com os valores de pulsos gerados perto da região dos 60 mm. Pulsos acima destas amplitudes não são processados pelo analisador.

5 - Outro dado importante é que a resolução do detetor é 0.200 mm (especificada pelo fabricante). Ou seja, o sistema ótimo seria aquele que pudesse cobrir os 75 mm com resolução de 0.200 mm/canal, o que somaria 375 canais. Sempre que se quiser, no

sistema AAP-LIE, aumentar a resolução de posição, pode-se fazer aumentando o ganho do amplificador na entrada do Módulo de Entrada de Sinal, perdendo com isso espaço útil do detetor. No entanto, o limite de 0.200 mm/canal não deve ser ultrapassado pois a partir daí não se verificará melhoria na resolução da curva.

4.3 - A determinação do raio de giro de uma amostra

Este teste se presta de maneira muito boa para ratificar a eficiência do aparelho tendo em vista que o raio de giro é um parâmetro estrutural da amostra calculável a partir dos dados obtidos. Vem da lei de Guinier [27].

A intensidade de espalhamento para um sistema de partículas em solução é dada por:

$$I(\vec{s}) = (\rho - \rho_0)^2 \left\{ \int \sigma(r) e^{-2\pi i \vec{s} \cdot \vec{r}} d\vec{r} \right\}^2 \quad (1)$$

onde:

$I(\vec{s})$ = intensidade espalhada pelo sistema de partículas

ρ = densidade eletrônica das partículas

ρ_0 = densidade do solvente

$\sigma(\vec{r})$ = função da forma $\sigma(\vec{r}) = 0$ fora da partícula e

$\sigma(\vec{r}) = 1$ dentro da partícula

$\vec{s}$ = vetor posição no espaço recíproco

$\vec{r}$ = vetor posição no espaço real

Para valores de $\vec{s} \rightarrow 0$, temos uma aproximação conhecida

como lei de Guinier. Vamos colocar a origem das coordenadas no centro de gravidade da partícula e analisemos o eixo x. Temos

$$\int \sigma(x) x dx = 0 \quad (2) \text{ pois a origem das coordenadas está no centro de gravidade.}$$

Para valores de $s_x \rightarrow 0$ pode-se escrever em série a componente exponencial da equação 1.

$$e^{-2\pi i \vec{s}_x \cdot \vec{X}} = 1 - 2\pi i s_x \cdot X + 2\pi^2 s_x^2 X^2 + \dots \quad (3)$$

onde $\vec{s}_x$ = componente de $\vec{s}$ na direção x.

Substituindo (3) em (1) temos:

$$I(s_x) = (\rho - \rho_0)^2 V^2 e^{-4\pi^2 s_x^2 R_x^2} \quad (4)$$

onde foi definido

$$R_x^2 = \frac{1}{V} \int x^2 \sigma(x) dx \quad (5)$$

e R_x representa a componente do raio de giro da partícula segundo a direção x.

Considerando os três eixos e lembrando que se trata de um conjunto de partículas totalmente ao acaso, pode-se escrever:

$$I(s) = N (\rho - \rho_0)^2 V^2 e^{-\frac{4}{3}\pi^2 s^2 R_G^2} \quad (6)$$

que é a lei de Guinier onde

N = número médio de partículas irradiadas

V = volume de partículas

R_G = Raio de giro da partícula definido como

$$R_G = \left[\frac{\int r^2 \rho(r) dr}{n} \right]^{1/2} \quad (7)$$

r = módulo do vetor posição r com origem no centro de massa da partícula

n = número de elétrons da partícula

$\rho(r)$ = Densidade eletrônica da partícula

Aplicando logaritmo à equação (6) temos

$$\text{Log} I(s) = \underbrace{\log N (\rho - \rho_0)^2 v^2}_{\text{constante}} - \underbrace{\frac{4}{3} \pi^2 R_G^2 \log e}_{\text{constante}} \cdot s^2$$

Donde se deduz que se graficarmos $\text{LOG}[I(S)]$ em função de s^2 (onde S é proporcional a $N - N_0$) teremos, na região em que é válida a lei de Guinier, uma reta. E mais, a raiz quadrada da inclinação desta reta é proporcional ao raio de giro da partícula.

As figuras 25 e 26 mostram os gráficos de $\text{LOG}[I(S)]$ Vs $(N - N_0)^{**2}$ para as curvas de espalhamento mostradas nas figuras 20 e 21 respectivamente. Para obtermos aquelas curvas, consideramos apenas dados de uma região angular equivalente a 1 cm a partir da borda da depressão causada pelo "beam stop" em ambas as curvas.

Pela exposição feita acima, o fato das curvas serem paralelas, ou terem o mesmo coeficiente angular, nos garante que o raio de giro é o mesmo nas curvas obtidas pelos dois sistemas.

Fig.25 - Curva $\text{LOG}(I(S))$ Vs $(N-N_0)^2$, dados AAP - Northern.

Fig.26 - Curva $\text{LOG}(I(S))$ Vs $(N-N_0)^2$, dados AAP - L.I.E.

CAPÍTULO V

CONCLUSÕES E POSSÍVEIS MELHORAMENTOS DO SISTEMA

Dentro do contexto em que se inseriu a proposta de execução do projeto, acreditamos que ele tenha alcançado plenamente seus objetivos. A exploração dos recursos da arquitetura eletrônica e das facilidades de programação dos subsistemas utilizados foi feita de forma conjunta resultando num sistema eficiente, versátil e de fácil operação. As medidas e comparações relatadas no capítulo IV mostram o desempenho perfeito do aparelho na aplicação direta para a qual ele foi projetado. Os seis meses de operação por que já passou até aqui, mostram sua confiabilidade.

O convívio com o problema, a experiência adquirida durante a constante busca de perfeição do funcionamento do aparelho, e a preocupação em torná-lo útil em toda gama de aplicações que podem ser realizadas com um AAP, nos mostrou, porém, alguns pontos onde se esgotam os recursos da arquitetura utilizada (Processador 8085, Kit SDK, etc) e mesmo do modo como estão interligados os vários subsistemas para formar o Sistema de Levantamento de Curvas de Espalhamento de Raios-X a Baixo ângulo (detetor, amplificadores, Conversor Tempo-em-Amplitude, conversor A/D, etc).

Neste capítulo final, enumeramos alguns limites no desempenho do AAP-LIE e tentaremos propor soluções para contorná-los. Estes limites são basicamente três: Tempo de processamento de pulso que, se na aplicação atual não compromete é importante para certo tipo de aplicações [28]; acoplamento Detetor/Analisador; e facilidades na operação. Com relação a isto, alguns pontos podem ser discutidos:

1 - Uma melhoria substancial seria a substituição do Kit SDK-85 por um Módulo Controlador Central próprio, com eletrônica e programação totalmente dedicados. Os recursos do processador não podem ser utilizados em sua totalidade devido à estrutura imposta ao sistema destinado a uso geral. Algumas vantagens são diretas:

- Melhor uso do sistema de interrupções, deixando a tarefa de processamento de pulsos mais rápida. Poder-se-ia economizar dois ou três "jump's" existentes no início do processo de atendimento de interrupção, se o sistema montado fosse ainda o 8085. No entanto, um processador com estrutura de interrupções melhor, como Z80, poderia ser utilizado para economizar também o tempo gasto no salvamento dos registradores no início da rotina de atendimento e no seu rearmazenamento no fim desta, caso em que o ganho seria substancial tendo em vista que aquela tarefa é feita por 4 instruções de 16 estados ao invés das 8 instruções de 96 estados gastos no 8085. O tempo morto típico poderia passar dos 100 atuais para 66 microssegundos, praticamente sem salto tecnológico significativo.

- Possibilidade de interfaceamento Homem/Máquina muito mais fácil. O sistema de acionamento do aparelho pode ser simplificado trocando sequências de comando, por exemplo GO D000 NEST pelo acionamento de uma única tecla. Isto pode ser feito implementando um Controlador de interrupções para expandir o número de interrupções disponíveis (No caso do 8085 este número sobe de 4 para 12 diretamente).

2 - Como mencionado no capítulo II, a resolução do detetor é de 200 micrometros, o que dá nos seus 75 mm de compri-

mento útil, 375 canais. A resolução do AAP-LIE é 0.298 mm/canal, da maneira como foi acoplado ao detetor, de forma a poder cobrir os 75 mm do detetor. O conversor utilizado é de 12 bits embora apenas 8 sejam lidos para agilizar o processo de transferência de dados (como o processador é de 8 bits, é necessário apenas um ciclo de leitura para tal transferência). Um esquema para melhor aproveitamento do detetor seria utilizar mais um bit do conversor, o que implicaria na disponibilidade de 512 canais. A manipulação adequada da amplificação dos pulsos na entrada do conversor poderia fazer com que os 375 canais iniciais do AAP cobrisse os 75 mm do detetor. É claro que agora a leitura teria que ser feita em dois ciclos para cada pulso, aumentando o tempo de leitura, mas não significativamente.

3 - Vimos ainda que o tempo de conversão é de 25 microssegundos e o tempo leitura e processamento do sinal é de aproximadamente 100 microssegundos, dos quais 33 são de leitura. Mais, os tempos de conversão e processamento são superpostos, de forma que a velocidade de conversão não está influenciando no tempo de processamento do pulso. As modificações sugeridas no item 1 só alteram o tempo computado como leitura de forma que só se conseguirmos diminuir drasticamente o tempo de processamento é que o tempo de conversão poderá ser um fator limitante. Estas reduções de tempo poderiam ser conseguidas usando um processador de 16 bits, e assim, daríamos um salto tecnológico podendo aí serem melhorados praticamente todos os blocos em que foi dividido o AAP.

4 - Analisando agora o caminho do pulso desde o detetor até a entrada do AAP verificamos o seguinte: Dois pulsos saem do detetor, um de cada lado. São amplificados e conformados e che

gam no conversor Tempo-em-Amplitude que gera um pulso cuja altura depende do tempo entre os pulsos, ou seja da distância de uma das extremidades onde chegou o raio-X. Este pulso é enviado para o AAP e logo na entrada há um conversor A/D que converte a amplitude do pulso num endereço de memória. A crítica que se faz aqui é que estes dois passos poderiam ser feitos num sô, ou seja, a conversão de tempo/amplitude e amplitude/endereço poderia ser substituída por tempo/endereço, economizando o tempo da Conversão A/D e barateando o custo do aparelho.

5 - Outra "fraqueza" do sistema desenvolvido é que ele depende de um sistema externo (no caso um Vax11) para armazenamento definitivo dos dados colhidos. Porém o VAX 11 pode sofrer quedas imprevisíveis e as vezes demoradas, o que pode paralisar completamente o escoamento de dados e ser fatal em casos de análise de fenômenos cinéticos. O L.I.E. desenvolveu um sistema Modular de aquisição e processamento de dados, o MICROlie, que contém todas as facilidades necessárias para o armazenamento e posterior processamento de dados que agora está sendo feita pelo VAX.E, mais é montado sobre um duto padrão, o S100 [29] que permite vários Môdulos inteligentes, controlarem processos diferentes e se comunicarem entre si quando necessário.

Com algumas poucas modificações, o AAP-LIE poderia ser transformado num módulo do MICROlie, tornando-o muito mais simples de operar (através de comandos de alto nível) e mais seguro pois não dependeria de sistema remoto de armazenamento, mas com opção de usá-lo, se necessário. A montagem do MICROlie também seria interessante na medida que abriria a possibilidade de utilizãlo para controlar outros sistemas dentro do Laboratório..

APÊNDICE I

DIAGRAMAS ELÉTRICOS DOS CIRCUITOS MONTADOS

UNIVERSIDADE DE SÃO PAULO	USP - IFOSC - DECM
LABORATÓRIO DE INSTRUMENTAÇÃO ELETRÔNICA	
MÓDULO DE CONTROLE CENTRAL (EXPANS.)	FL 1/1
PROJETO: EDSON S. MOREIRA	
DESENHO: JOSÉ AUGUSTO	SÃO CARLOS, JUNHO, 1984

UNIVERSIDADE DE SÃO PAULO	USP - IEQSC - DECM
LABORATÓRIO DE INSTRUMENTAÇÃO ELETRÔNICA	
MÓDULO DE OPERAÇÃO (EXPANSÃO)	
PROJETO: EDSON DOS S. MOREIRA	
DESENHO: JOSE AUGUSTO	SÃO CARLOS, JUNHO, 1984

APÊNDICE II

LISTAGEM DO PROGRAMA DE CONTROLE DO AAP

```

;*****
;
;                                PROGRAMA MULTICANAL
;
;                                Edson dos Santos Moreira
;                                18.11.83
;*****
0000
0000
LIST A,X
ORG 0000H

D000
D000
;*****
;    1 -      BIBLIOTECA DE MACROS
;*****
D000
D000
;    1.1 -   MACRO QUE SALVA OS REGISTRADORES NA FILHA

PUSH_REG     MACRO
              PUSH    PSW
              PUSH    B           ;
              PUSH    D           ; MACRO DE SALVAMENTO DE REGISTRADORES
              PUSH    H           ;
              ENDM

D000
;    1.2 -   MACRO QUE REARMAZENA OS REGISTRADORES

POP_REG       MACRO
              POP     H
              POP     D           ;
              POP     B           ; MACRO DE RECUPERACAO DE REGISTRADORES
              POP     PSW        ;
              ENDM

```

```

;*****

D000
;      2 - INICIALIZACAO
D000
;*****

D000
D000
D000 F3      DI
D001 31 FF 93 LXI    SP,STK_ADR      ;Define STACK POINTER
D004 3E 03     MVI    A,03H      ; Inicializacao das Portas da 8155-1
D006 D3 20     OUT    PORT_A0    ; Portas A e B --> saida
                                   ; Porta C --> entrada
                                   ; Sendo: A --> eixo X do Osciloscópio
                                   ;      B --> eixo Y do Osciloscópio
                                   ;      C --> chave A/M e sincron. timer

D008
D008
D008 AF      XRA    A
D009 32 00 20 STA    SHIFT
D00C 32 03 20 STA    CONTX
D00F 32 04 20 STA    TEMP1
D012 32 05 20 STA    TEMP2
D015 32 09 20 STA    CONT_VAR_M
D018 32 0A 20 STA    CONT_VAR_A
D01B 32 07 20 STA    TEMP5
D01E 32 0B 20 STA    BYTE_DSFLD
D021
D021 3E 15     MVI    A,15H
D023 32 06 20 STA    DELAY
D026 3E 09     MVI    A,09H
D028 32 08 20 STA    CONT_ROTATE
D02B 32 60 20 STA    FLAG1
D02E
D02E 21 CB 20 LXI    H,20CBH      ; *** INTERRUPTAO 6.5 ***
D031 36 C3     MVI    M,0C3H      ; JMF
D033 23        INX    H
D034 EB       XCHG
D035 21 1E D1 LXI    H,COLETA      ; COLETA
D038 D9       SHLX
D039
D039 21 CE 20 LXI    H,20CEH      ; *** INTERRUPTAO 7.5 ***
D03C 36 C3     MVI    M,0C3H      ; JMF
D03E 23        INX    H
D03F EB       XCHG
D040 21 50 D1 LXI    H,CARREGA      ; CARREGA
D043 D9       SHLX
D044
;      2.1 - OPCAO - ESPERA QUE O OPERADOR DEFINA O MODO DE OPERACAO DO SISTEI

D044
D044 3E 00     MVI    A,0H
D046 06 00     MVI    B,0H
D048 21 3F D4 LXI    H,OPTE      ; Mensagem 'OPTE'
D04B CD B7 02 CALL    MON$_OUTPT    ; no display
D04E 3E 0E     MVI    A,0EH      ; Habilita interrupcao 5.5

```

D050 30
D051 FB
D052 CD E7 02
D055 FE 0C
D057 CA 83 D0
D05A FE 0D
D05C CA 62 D0
D05F CD 15 02

SIM
EI
CALL MON\$_RKBD ; Espera entrada de opcao no teclado
CPI 0CH ; 'C' e' coleta
JZ VARR_COL
CPI 0DH ; 'D' e' display sem coleta
JZ VARR_SO
CALL MON\$_ERRO

```

D062
D062
D062
D062
D062 2A 20 20
D065 22 50 28
D068 2A 21 20
D06B 22 51 28
D06E
D06E AF
D06F 32 60 20
D072
D072 3E 00
D074 06 01
D076 21 47 D4
D079 CD B7 02
D07C
D07C 3E 0B
D07E 30
D07F FB
D080
D080 C3 01 D1

```

```

;*****
;      3 -      ATENDIMENTO DE OPCAO DISPLAY SEM COLETA
;*****

VARR_SO:
LHLD PRIME
SHLD TMP      ; ARMAZENA TEMPO DECORRIDO ATE O FIM DA COLETA
LHLD SEGU
SHLD TMP+1

XRA A
STA FLAG1

MVI A,00H
MVI B,01H      ; MENSAGEM 'DISP'
LXI H,OPTE+8    ;      NO DISPLAY
CALL NON$_OUTPT

MVI A,0BH      ; MASCARA TODAS AS INTERRUPCOES, MENOS A 7.5
SIM
EI

JMP DSPL1

```



```

D083
D083
D083
D083 3E 00
D085 06 01
D087 21 43 D4
D08A CD B7 02
D08D
D08D 11 FF FF
D090 CD F1 05
D093 CD F1 05
D096 CD F1 05
D099
D099 CD 68 D2
D09C

;*****
;      4 -  ATENDIMENTO DE OPCAO COLETA
;*****

VARR_COL:
MVI  A,00H
MVI  B,01H      ; Mensagem 'DIFr'
LXI  H,OPTE+4    ;      no display
CALL MON$_OUTPT

LXI  D,OFFFH
CALL MON$_DELAY
CALL MON$_DELAY
CALL MON$_DELAY

CALL TECLADO

; 4.1 -  TRATAMENTO DOS DADOS QUE ENTRAM PELO TECLADO
;
; - ESTAO ARMazenADOS NAS POSICOES CLEAR...CLEAR+5
;
; CORRESPONDENTES AOS TRES BYTES DO TEMPO DE COLETA
;
; ESTA ROTINA FORMATA OS DADOS PARA SEREM USADOS PE
;
; ROTINA DE CONVERSAO.
;

D09C
D09C 21 0F 28
D09F CD 57 D2
D0A2 FE 07
D0A4 D2 15 02
D0A7 32 22 28
D0AA CD 57 D2
D0AD 32 21 28
D0B0 CD 57 D2
D0B3 32 20 28
D0B6 CD 92 D3

LXI  H,BUFF-1
CALL FORMAT
CPI  07H      ; VERIFICA SE E' > QUE 70000
JNC  MON$_ERRO ; SE FOR.... ERRO
STA  ADRS+2   ; SE NAO, ARMAZENA
CALL FORMAT
STA  ADRS+1
CALL FORMAT
STA  ADRS
CALL CNV_BCDTOBIN_W

; ESTA ROTINA CONVERTE O DADO BCD QUE ESTA' EM
; ADRS EM BINARIO E COLOCA O RESULTADO EM ADRD

D0B9 3E 01
D0BB 06 00
D0BD 21 4B D4
D0C0 CD B7 02
D0C3
D0C3 3E 00
D0C5 06 01
D0C7 21 43 D4

MVI  A,01H
MVI  B,00H
LXI  H,OPTE+12 ; LIMPA CAMPO DE DADOS DO DISPLAY
CALL MON$_OUTPT

MVI  A,00H
MVI  B,01H      ; MENSAGEM 'DIFr'
LXI  H,OPTE+4    ;      NO DISPLAY

CALL MON$_OUTPT

D0CA CD B7 02
D0CD
D0CD 11 30 28
D0D0 ED
D0D1 11 40 28
D0D4 D9
D0D5
D0D5 CD 56 D3
D0D8

LXI  D,ADRD      ; PASSA O CONTEUDO DE ADRD PARA HL
LHLX
LXI  D,TEMPO     ; PASSA CONTEUDO DE HL PARA TEMPO
SHLX

CALL RELOGIO      ; CHAMA SUBROTINA DE PROGRAMACAO DE TEMPO

```

4.2 - LIMPEZA DA MEMORIA A SER UTILIZADA

D0D8		LXI	H, MEM_DSPL_I
D0DB	21 00 80	LOOP:	
D0DB		MVI	A, 00H
D0DB	3E 00	MOV	M, A
D0DD	77	INX	H
D0DE	23	MVI	A, 84H
D0DF	3E 84	CMP	H
D0E1	BC	JNZ	LOOP
D0E2	C2 DB D0		
D0E5			
D0E5	3E 08	MVI	A, 08H
D0E7	30	SIM	
D0E8	FB	EI	

```
*****
DOE9
;      5 -   ROTINA DISPLAY
;
;
;      Versao original por U. T. BORELLI
;
DOE9
*****

;      5.1 -   LEITURA E TRATAMENTO DO TIMER

DOE9
DOE9      DSPL:
DOE9 3E 4F      MVI    A,4FH    ; CONTROL WORD DO 8253
;              ; - COUNTER-1
;              ; - COUNTER LATCHING OPERATION

DOEB D3 A3      OUT    0A3H

DOED
DOED DB A1      IN     0A1H
DOEF 32 20 20   STA    PRIME
DOF2 DB A1      IN     0A1H
DOF4 32 21 20   STA    SEGU
DOF7 2A 20 20   LHLD   PRIME
DOFA 01 00 00   LXI    B,00H
DOFD 08         SBH
DOFE CA 62 D0   JZ     VARR_50
D101

;      5.2 -   LEITURA DE DADO NA MEMORIA E ENVIO PARA O OSCILOSCOPIO

D101
D101      DSPL1:
D101 11 00 80   LXI    D,MEM_DSPL_I    ;(D-E)= END. NA MEM.
D104 3A 00 20   LDA    SHIFT        ;SHIFT USADO CRIAR END. NA SHIFT TABLE
D107 07         RLC
D108 07         RLC                ;CRIA O "LOW BYTE"
D109 4F         MOV    C,A
D10A 06 D0      MVI    B,HDAT_ADR    ;"HIGH BYTE" DO END. NA "SHIFT
;              ;TABLE".

D10C      NEXT:
D10C 69         MOV    L,C
D10D 60         MOV    H,B
D10E E9         PCHL                ;"PROG. COUNT."= ENDEREÇO
;              ;NA "SHIFT TABLE"

D10F
D10F      OSCILOS:
D10F 3A 03 20   LDA    CONTX        ;ROTINA PARA ENVIAR
D112 D3 22      OUT    PORT_CO      ;DADOS AO OSCILO-
D114 3A 0B 20   LDA    BYTE_DSPLD   ;COPIO
D117 D3 21      OUT    PORT_BO
D119 3A 03 20   LDA    CONTX        ;ATUALIZA CONTADOR
D11C 3C         INR    A
D11D 32 03 20   STA    CONTX
D120 CA 29 D1   JZ     GETSHIFT     ;A CADA VARRED. DO
;              ;"DISPLAY",TOMA UM
;              ;NOVO SHIFT.
;              ;ATUALIZA END. PROXIMA

D123 EB         XCHG
D124 28 04      AND     04H
D126 C3 0C D1   JMP     NEXT
```

D129
D129

D129
D129
D129 DB 23
D12B E6 07
D12D 32 04 20
D130 3A 06 20
D133 3D
D134 32 06 20
D137 C2 4F D1

D13A 3E 15
D13C 32 06 20
D13F

D13F
D13F 3A 04 20
D142 E6 02
D144 C2 67 D1
D147
D147 3A 04 20
D14A E6 01
D14C C2 A3 D1
D14F

D14F
D14F
D14F 3A 04 20
D152 E6 04
D154 C2 B8 D1
D157 3A 09 20
D15A FE F0
D15C DA 61 D1
D15F 3E 00
D161
D161 32 00 20
D164 C3 E9 D0
D167

D167
D167
D167 3A 04 20
D16A E6 04
D16C C2 8E D1
D16F 3A 09 20
D172 32 07 20
D175 FE 11
D177 DA 82 D1
D17A 3E 00
D17C 32 09 20
D17F C3 47 D1
D182
D182 FE 00
D184 CA 47 D1
D187 3D
D188 32 09 20
D18B C3 47 D1

; 5.3 - ESCOLHA DO DESLOCAMENTO ADEQUADO
;
DENTRO DA PALAVRA DE 24 BITS

GETSHIFT:
IN PORT_D0 ;LE A PORTA D0 = SITUACAO
ANI 07H ;DA CHAVE A/M E BOTOES.
STA TEMP1 ;RESERVA CODIGO LIDO.
LDA DELAY ;OTIMIZA O TEMPO DE
DCR A ;ATENDIMENTO DOS
STA DELAY ;BOTOES.
JNZ TEST_A_M

MVI A,15H ;REINICIALIZA DELAY
STA DELAY

; 5.4 - TESTA CHAVE DE ESCALA

LDA TEMP1
ANI 02H ;TESTA BOTAO '-'.
JNZ SBT_VAR_M ;SIM, SALTA PARA SUBTRAIR
TEST_BIT_0;
LDA TEMP1 ;NAO, TESTA BOTAO '+'.
ANI 01H
JNZ ADD_VAR_M ;SIM, SALTA PARA SOMAR.

; 5.5 - TESTA CHAVE DE AUTOMATICO/MANUAL

TEST_A_M:
LDA TEMP1 ;NAO, TESTA CHAVE A/M.
ANI 04H
JNZ CRIA_SHIFT ;AUTOMATICO, CRIAR SHIFT AUTOM.
LDA CONT_VAR_M ;MANUAL, CRIAR SHIFT MANUAL
CPI 0F0H ;TESTA SE O CONT_VAR_M >= F0H,
JC POOL ;SIM, FAZER SHIFT = 00.
MVI A,00H ;NAO, FAZER SHIFT = CONT_VAR_M.
POOL:
STA SHIFT
JMP DSPL

; 5.6 - ATENDE CHAVE DE DIMINUICAO DE ESCALA

SBT_VAR_M:
LDA TEMP1
ANI 04H
JNZ AUTOM ;SALTA, ESTA EM AUTOM.
LDA CONT_VAR_M ;MANUAL.
STA TEMP5 ;RESERVA O CONTADOR DE VARREDURA MANUAL.
CPI 11H
JC TEST_Z_DCR ;CONT_VAR_M < 11H, E = 0? DECREM.
MVI A,00H ;NAO, ZERO O CONT_VAR_M
STA CONT_VAR_M
JMP TEST_BIT_0 ;VAI TESTAR O BOTAO '+'.
TEST_Z_DCR:
CPI 00H ;SE FOR 0, VAI TESTAR
JZ TEST_BIT_0 ;BOTAO '+'.
DCR A ;NAO, DECREMENTA O
STA CONT_VAR_M ;CONT_VAR_M.
JMP TEST_BIT_0 ;E VAI TESTAR '+'.
;

D18E 3A 07 20
D191 32 09 20
D194 FE F0
D196 CA 47 D1
D199 3D
D19A 32 09 20

D19D 32 07 20
D1A0 C3 47 D1
D1A3

D1A3
D1A3

D1A3 3A 07 20
D1A6 32 09 20
D1A9 FE 10
D1AB CA 4F D1
D1AE 3C
D1AF 32 09 20
D1B2 32 07 20
D1B5 C3 4F D1
D1B8
D1B8 3A 0A 20
D1BB A7
D1BC CA E9 D0
D1BF 3D
D1C0 32 05 20

D1C3 3A 05 20
D1C6 E6 FF
D1C8 CA E9 D0
D1CB 07
D1CC 32 05 20
D1CF 3A 08 20
D1D2 3D
D1D3 32 08 20
D1D6 D2 C3 D1

D1D9
D1D9 E5
D1DA 21 08 20
D1DD 3A 08 20
D1E0 E6 0F
D1E2 32 08 20
D1E5 3A 09 20
D1EB 86
D1E9 FE F0
D1EB DA F3 D1
D1EE 3E 00
D1F0 C3 FC D1
D1F3
D1F3 E6 1F
D1F5 FE 11
D1F7 DA FC D1
D1FA 3E 10
D1FC
D1FC 32 00 20
D1FF 3E 09
D201 32 08 20
D204 E1

AUTOM:
LDA TEMP5 ;TESTA SE CONT_VAR_M
STA CONT_VAR_M ;E IGUAL A FOH.
CPI OFOH
JZ TEST_BIT_0 ;SIM, TESTAR BOTAO '+'
DCR A ;NAO, DECREMENTAR O
STA CONT_VAR_M ;CONT_VAR_M

STA TEMP5
JMP TEST_BIT_0 ;TESTAR BOTAO '+',

; 5.7 - ATENDE CHAVE DE AUMENTO DE ESCALA

ADD_VAR_M:

LDA TEMP5
STA CONT_VAR_M ;TESTA SE CONT_VAR_M E
CPI 10H ;IGUAL A 10H
JZ TEST_A_M ;SIM, TESTAR: AUTOMATICO OU MANUAL?
INR A ;NAO, INCREMENTAR E TESTAR;
STA CONT_VAR_M
STA TEMP5
JMP TEST_A_M ;AUTOMATICO OU MANUAL?
CRIA_SHIFT:
LDA CONT_VAR_A
ANA A
JZ DSPL ;SE NAO FEZ COLETA, VOLTA PARA DSPL
DCR A ;** PARA AJUSTAR O SHIFT **
STA TEMP2 ;TESTA SE O CONTADOR DE VARREDURAS
NEW_ROTATE: ;NAO, CRIAR O SHIFT ADEQUADO
LDA TEMP2
ANI OFFH ;CONT. DE VARRED. AUTOMATICO = 0?
JZ DSPL ;SIM, VAI PARA DISPLAY (ANTES ERA : SHIFT_NMB)
RLC ;NAO,
STA TEMP2
LDA CONT_ROTATE
DCR A ;DECREMENTA O CONTADOR DE ROTATES
STA CONT_ROTATE
JNC NEW_ROTATE ;SE HA CARRY, CRIAR O SHIFT ADEQUADO
;NAO, FAZER NOVA PROCURA DO SHIFT ADEQUADO

SHIFT_NMB:

PUSH H ;SALVA O REGISTRADOR H
LXI H, CONT_ROTATE ;VAI CRIAR UM SHIFT
LDA CONT_ROTATE ;BASEADO NO CONTEUDO
ANI OFH ;DO CONT_ROTATE E DO CONT_VAR_M
STA CONT_ROTATE
LDA CONT_VAR_M
ADD H
CPI OFOH ;COMPARA COM FOH. SE FOR >= FAZ SHIFT = 00
JC CMP_11 ;SE NAO, VAI COMPARAR COM 11H
MVI A, 00H
JMP REINIC
CMP_11:
ANI 1FH ;COMPARA COM 11H. SE FOR < ,
CPI 11H ;REINICIALIZA COM O SHIFT CALCULADO.
JC REINIC
MVI A, 10H ; SE FOR >= , PEGAR O TERCEIRO BYTE
REINIC:
STA SHIFT
MVI A, 09H ;NAO, JA TEM O SHIFT ADEQUADO. REINICIALIZA
STA CONT_ROTATE ;O CONTADOR DE ROTATES
POP H ;RESTAURA O REGISTRADOR H

D205 C3 E9 D0

D208 FE F0

D20A DA 1F D2

D20D C6 10

D20F 32 00 20

D212 FE 11

D214 DA E9 D0

D217 3E 00

D219 32 00 20

D21C C3 E9 D0

D21F

D21F 3E 10

D221 32 00 20

D224 C3 E9 D0

D227

D227

D227

D227

D000

D000 C3 43 D0

D003 00

D004 C3 4A D0

D007 00

D008 C3 53 D0

D00B 00

D00C C3 5D D0

D00F 00

D010 C3 68 D0

D013 00

D014 C3 74 D0

D017 00

D018 C3 82 D0

D01B 00

D01C C3 8F D0

D01F 00

D020 C3 9B D0

D023 00

D024 C3 A3 D0

D027 00

D028 C3 B1 D0

D02B 00

D02C C3 C0 D0

D02F 00

D030 C3 D0 D0

D033 00

D034 C3 E1 D0

D037 00

D038 C3 F2 D0

D03B 00

D03C C3 02 D1

D03F 00

D040 C3 11 D1

JMP DSPL

AJUST_16:

CPI 0F0H

JC SHIFT_10

ADI 10H

STA SHIFT

CPI 11H

JC DSPL

MVI A,00H

STA SHIFT

JMP DSPL

SHIFT_10:

MVI A,10H

STA SHIFT

JMP DSPL

VAI PARA A ROTINA DISPLAY.

PEGAR OS BITS 16 A 23 DA MEMORIA.

SE CONT_VAR_M < F0H, FAZER SHIFT=10H

CONT_VAR_M >= F0H, FAZER

CONT_VAR_M + 10H = SHIFT

COMPARA COM 11H

CONT_VAR_M < 11H, VAI PARA DISPLAY

NAD, FAZER SHIFT = 0 E

VOLTAR PARA A ROTINA DISPLAY

FAZER SHIFT = 10H

5.8 - TABELA DE DESLOCAMENTOS

ORG 0D000H

JMP AJST0

NOP

JMP AJST1

NOP

JMP AJST2

NOP

JMP AJST3

NOP

JMP AJST4

NOP

JMP AJST5

NOP

JMP AJST6

NOP

JMP AJST7

NOP

JMP AJST8

NOP

JMP AJST9

NOP

JMP AJST10

NOP

JMP AJST11

NOP

JMP AJST12

NOP

JMP AJST13

NOP

JMP AJST14

NOP

JMP AJST15

NOP

JMP AJST16

5.9 - AJUSTES - APONTADOS PELA TABELA DE DESLOCAMENTOS -
QUE POR SUA VEZ FOI ACESSADA PELA ROTINA GETSHIFT

D043
D043

D043 1A
D044 32 0B 20
D047 C3 0F D1

D04A ED
D04B 10
D04C 7D
D04D 32 0B 20
D050 C3 0F D1

D053 ED
D054 10
D055 10
D056 7D
D057 32 0B 20
D05A C3 0F D1

D05D ED
D05E 10
D05F 10
D060 10
D061 7D
D062 32 0B 20
D065 C3 0F D1

D068 ED
D069 10
D06A 10
D06B 10
D06C 10
D06D 7D
D06E 32 0B 20
D071 C3 0F D1

D074 D5
D075 ED
D076 EB
D077 18
D078 18
D079 18
D07A 7A
D07B 32 0B 20
D07E D1
D07F C3 0F D1

D082 D5
D083 ED
D084 EB
D085 18
D086 18
D087 7A
D088 32 0B 20
D08B D1
D08C C3 0F D1

AJST0: ;PEGA BITS 0 A 7 DA MEM.

LDAX D
STA BYTE_DSPLD
JMP OSCILOS

AJST1: ;PEGA BITS 1 A 8 DA MEM.

LHLX
SHLR
MOV A,L
STA BYTE_DSPLD
JMP OSCILOS

AJST2: ;PEGA BIT 2 A 9 DA MEM.

LHLX
SHLR
SHLR
MOV A,L
STA BYTE_DSPLD
JMP OSCILOS

AJST3: ;PEGA BIT 3 A 10 DA MEM.

LHLX
SHLR
SHLR
SHLR
MOV A,L
STA BYTE_DSPLD
JMP OSCILOS

AJST4: ;PEGA BIT 4 A 11 DA MEM.

LHLX
SHLR
SHLR
SHLR
SHLR
MOV A,L
STA BYTE_DSPLD
JMP OSCILOS

AJST5: ;PEGA BIT 5 A 12 DA MEM.

PUSH D
LHLX
XCHG
SDEL
SDEL
SDEL
MOV A,D
STA BYTE_DSPLD
POP D
JMP OSCILOS

AJST6: ;PEGA BIT 6 A 13 DA MEM.

PUSH D
LHLX
XCHG
SDEL
SDEL
MOV A,D
STA BYTE_DSPLD
POP D
JMP OSCILOS

AJST7: ;PEGA BIT 7 A 14 DA MEM.

D08F D5
D090 ED
D091 EB
D092 18
D093 7A
D094 32 0B 20
D097 D1
D098 C3 0F D1

D09B ED
D09C 7C
D09D 32 0B 20
D0A0 C3 0F D1

D0A3 D5
D0A4 EB
D0A5 28 01
D0A7 ED
D0A8 10
D0A9 7D
D0AA 32 0B 20
D0AD D1
D0AE C3 0F D1

D0B1 D5
D0B2 EB
D0B3 28 01
D0B5 ED
D0B6 10
D0B7 10
D0B8 7D
D0B9 32 0B 20
D0BC D1
D0BD C3 0F D1

D0C0 D5
D0C1 EB
D0C2 28 01
D0C4 ED
D0C5 10
D0C6 10
D0C7 10
D0C8 7D

D0C9 32 0B 20
D0CC
D0CC D1
D0CD C3 0F D1

D0D0 D5
D0D1 EB
D0D2 28 01
D0D4 ED
D0D5 10
D0D6 10
D0D7 10
D0D8 10
D0D9 7D
D0DA 32 0B 20
D0DD D1
D0DE C3 0F D1

PUSH D
LHLX
XCHG
SDEL
MOV A,D
STA BYTE_DSPLD
POP D
JMP OSCIOS

AJST8:
LHLX
MOV A,H
STA BYTE_DSPLD
JMP OSCIOS

AJST9:
PUSH D
XCHG
AND 01H
LHLX
SHLR
MOV A,L
STA BYTE_DSPLD
POP D
JMP OSCIOS

AJST10:
PUSH D
XCHG
AND 01H
LHLX
SHLR
SHLR
MOV A,L
STA BYTE_DSPLD
POP D
JMP OSCIOS

AJST11:
PUSH D
XCHG
AND 01H
LHLX
SHLR
SHLR
SHLR
MOV A,L

STA BYTE_DSPLD

POP D
JMP OSCIOS

AJST12:
PUSH D
XCHG
AND 01H
LHLX
SHLR
SHLR
SHLR
SHLR
MOV A,L
STA BYTE_DSPLD
POP D
JMP OSCIOS

;PEGA BIT 8 A 15 DA MEM.,
;OU SEJA, O BYTE CENTRAL

;PEGA BIT 9 A 16 DA MEM.

;PEGA BIT 10 A 17 DA MEM.

;PEGA BIT 11 A 18 DA MEM.

;PEGA BIT 12 A 19 DA MEM.

DOE1 D5
DOE2 EB
DOE3 28 01
DOE5 ED
DOE6 EB
DOE7 18
DOE8 18
DOE9 18
DOEA 7A
DOEB 32 0B 20
DOEE D1
DOEF C3 0F D1

DOF2 D5
DOF3 EB
DOF4 28 01
DOF6 ED
DOF7 EB
DOF8 18
DOF9 18
DOFA 7A
DOFB 32 0B 20
DOFE D1
DOFF C3 0F D1

D102 D5
D103 EB
D104 28 01
D106 ED
D107 EB
D108 18
D109 7A
D10A 32 0B 20
D10D D1
D10E C3 0F D1

D111 D5
D112 EB
D113 28 01
D115 ED
D116 7C
D117 32 0B 20
D11A D1
D11B C3 0F D1

AJST13:
PUSH D
XCHG
AND 01H
LHLX
XCHG
SDEL
SDEL
SDEL
MOV A,D
STA BYTE_DSPLD
POP D
JMP OSCILOS

AJST14:
PUSH D
XCHG
AND 01H
LHLX
XCHG
SDEL
SDEL
MOV A,D
STA BYTE_DSPLD
POP D
JMP OSCILOS

AJST15:
PUSH D
XCHG
AND 01H
LHLX
XCHG
SDEL
MOV A,D
STA BYTE_DSPLD
POP D
JMP OSCILOS

AJST16:
PUSH D
XCHG
AND 01H
LHLX
MOV A,H
STA BYTE_DSPLD
POP D
JMP OSCILOS

;PEGA BIT 13 A 20 DA MEM.

;PEGA BIT 14 A 21 DA MEM.

;PEGA BIT 15 A 22 DA MEM.

;PEGA BIT 16 A 23 DA MEM.,
;OU SEJA, BYTE MAIS
;SIGNIFICATIVO DA MEM.

```

*****
D11E
;      6 -      ATENDIMENTO DE INTERRUPTAO PARA COLETA DE DADOS
D11E
*****
D11E
D11E
; ESTA ROTINA E' ACIONADA PELA INTERRUPTAO 6.5
; LE DADOS DA PORTA LIGADA AO CONVERSOR. O DADO GERA UM ENDEREÇO QUE DEVE SER INCR
; PARA CADA CANAL ESTA' RESERVADA UMA PALAVRA DE 4 BYTES, EMBORA SO' 3 SEJAM UTILIZ.

D11E
D11E      COLETA:
D11E F3      DI
D11F      PUSH_REG      ;REGISTRADORES
D11F F5      +          PUSH    PSW
D120 C5      +          PUSH    B      ;
D121 D5      +          PUSH    D      ; MACRO DE SALVAMENTO DE REGISTRADORES
D122 E5      +          PUSH    H      ;
D123
D123 DB B0      IN      PORT_DAT      ; LE O DADO. O DADO E' LIDO LOGO NA ENTRADA DA ROTINA POIS ISTO
; LIBERA O CONVERSOR PARA A PROXIMA CONVERSÃO ENQUANTO O DADO
; ATUAL E' PROCESSADO.

D125
D125 01 01 00      LXI      B,01H      ; PREPARA PAR BC COM CONSTANTE 0001 PARA A SOMA EM 16 BITS
D128 21 00 80      LXI      H,MEM_DSPL_I      ; PREPARA HL COM ENDEREÇO INICIAL DOS CANAIS
D12B 16 00      MVI      D,00H      ; PREPARA REGIS. D PARA FORMAR, COM REGIS. E, DADO DE
; 16 BITS

D12D
D12D FE FF      CPI      OFFH      ; ESTAS 4 INSTRUÇÕES SÃO NECESSÁRIAS DEVIDO AO FATO DO CONVERSOR
D12F CA 43 D1      JZ      RETOR      ; CONVERTER QUALQUER TENSÃO ACIMA DE 10V COMO UMA PALAVRA FF
D132 FE 00      CPI      00H      ; TAMBÉM, DEVIDO A SENSIBILIDADE DO CIRCUITO DE DISPARO E A RESO
D134 CA 43 D1      JZ      RETOR      ; LUCAO DO CONVERSOR, PODE GERAR CONTAGENS FALSAS NO CANAL ZERO.

D137
D137 B7      ORA      A      ; RESETA CARRY FLAG. NECESSARIO, POIS SDEL FAZ SHIFT COM CARRY
D138
D138 5F      MOV      E,A      ; O DADO LIDO ESTA EM E.
D139 18      SDEL      ; MULTIPLICA POR 2
D13A 18      SDEL      ; MULTIPLICA POR 2. EM DE TEM-SE O DESLOCAMENTO A PARTIR
; DO ENDEREÇO INICIAL DOS CANAIS.
D13B 19      DAD      D      ; AO ENDEREÇO INICIAL E' SOMADO O DESLOCAMENTO. EM HL
; TEM-SE O ENDEREÇO REAL DO CANAL A SER ALTERADO.

D13C EB      XCHG
D13D ED      LHLX      ; BUSCA NA MEMORIA A CONTAGEM ATUAL DO CANAL
D13E 09      DAD      B      ; ATUALIZA SOMANDO UM
D13F D9      SHLX      ; REARMAZENA
D140
D140 DA 49 D1      JC      TERC_BYTE      ; SE ESTOUROU 16 BITS, ATUALIZAR O TERCEIRO BYTE
D143
D143      RETOR:
D143      POP_REG      ; SE NAO, REARMAZENAR REGISTRADORES
D143 E1      +          POP      H
D144 D1      +          POP      D      ;
D145 C1      +          POP      B      ; MACRO DE RECUPERAÇÃO DE REGISTRADORES
D146 F1      +          POP      PSW      ;
D147 FB      EI      ; HABILITA INTERRUPTOES
D148 C9      RET      ; RETORNA AO PONTO DA INTERRUPTAO

```

```
D149
D149          TERC_BYTE:
D149 13          INX    D          ; FAZ DE APONTAR
D14A 13          INX    D          ;          TERCEIRO BYTE
D14B EB          XCHG
D14C 34          INR     H          ; INCREMENTA CONTAGEM DO CANAL
D14D
D14D C3 43 D1    JMP RETOR
```

```

;*****
D150
;      7 -  ROTINA DE ATENDIMENTO DE INTERRUPTO PARA TRANSMISSAO DE DADOS
D150
;*****
D150
D150
D150      CARREGA:
D150 F3      DI
D151      PUSH_REG

D151 F5      +      PUSH    PSW
D152 C5      +      PUSH    B          ;
D153 D5      +      PUSH    D          ; MACRO DE SALVAMENTO DE REGISTRADORES
D154 E5      +      PUSH    H          ;
D155
D155 3A 60 20      LDA     FLAG1
D156 FE 00      CPI      0H          ; FLAG1 SINALIZA SE O PROGRAMA FOI INTERROMPIDO
; NA OPCAO COLETA OU DISPLAY,
D15A CA 69 D1      JZ      CARR      ; SE VEIO DA DISPLAY NAO E' NECESSARIO SALVAR
; CONTEUDO DO TIMER

D15D
D15D 2A 20 20      LHLD    PRIME
D160 22 50 28      SHLD    TMP          ; ARMAZENA TEMPO DECORRIDO ATE A INTERRUPTO
D163 2A 21 20      LHLD    SEGU
D166 22 51 28      SHLD    TMP+1
D169
D169      CARR:
D169 3E 00      MVI      A,00H
D16B 06 01      MVI      B,01H          ; COLOCA 'CARR.'
D16D 21 4D D4      LXI     H,DATA      ; NO DISPLAY
D170 CD E7 02      CALL    MON$_OUTPT
D173
D173 3E 4F      MVI      A,4FH          ; PARA O TIMER (8155)
D175 D3 28      OUT      PORT_COMMAND_EXP_RAM
D177
D177 3E 05      MVI      A,05H          ; VELOCIDADE LS BYTE 9600BAUD
D179 D3 2C      OUT      PORTLOW_TIM_EXP_RAM
D17B 3E 40      MVI      A,40H          ; VELOCIDADE HS BYTE ,E PROGRAMA DO TIMER (ONDA,
; CONTINUA E QUADRADA),
D17D D3 2D      OUT      PORTHIGH_TIM_EXP_RAM
D17F 3E CF      MVI      A,0CFH          ; STSRT O TIMER (8155),
D181 D3 28      OUT      PORT_COMMAND_EXP_RAM
D183
D183 3E EF      MVI      A,0EFH          ; CARREGA MODE INSTRUCTION WORD
; - 64X
; - NOPARITY
; - 2 STOPBITS
; - EIGHT BIT LENGTH

D185 D3 C0      OUT      COM_STAT_51
D187 3E 37      MVI      A,37H          ; CARREGA COMMAND WORD
; - ENABLE Rx,Tx
; - RESET ERROR FLAGS

D189 D3 C0      OUT      COM_STAT_51
D18B
D18B
D18B 4F      COMECO:
XRA      A

```

D18C 32 30 20	STA	CONT_CAR	
D18F 2A 30 28	LHLD	ADRD	;
D192 7C	MOV	A,H	;
D193 E5	PUSH	H	; TRANSMITE TEMPO PROGRAMADO
D194 21 00 C0	LXI	H, TXRX_51	;
D197 CD FD D2	CALL	HEX_ASC	;
D19A E1	POP	H	
D19B 7D	MOV	A,L	
D19C 21 00 C0	LXI	H, TXRX_51	
D19F CD FD D2	CALL	HEX_ASC	
D1A2 CD 49 D3	CALL	CRET	
D1A5			
D1A5 2A 50 28	LHLD	TMP	;
D1A8 7C	MOV	A,H	;
D1A9 E5	PUSH	H	; TRANSMITE TEMPO TRANSCORRIDO
D1AA 21 00 C0	LXI	H, TXRX_51	;
D1AD CD FD D2	CALL	HEX_ASC	;
D1B0 E1	POP	H	
D1B1 7D	MOV	A,L	
D1B2 21 00 C0	LXI	H, TXRX_51	
D1B5 CD FD D2	CALL	HEX_ASC	
D1B8 CD 49 D3	CALL	CRET	
D1BB			
D1BB 11 00 80	LXI	D, MEM_DSPL_I	
D1BE 21 00 C0	LXI	H, TXRX_51	
D1C1			
D1C1	DADO:		
D1C1 1A	LDAX	D	;
D1C2 CD FD D2	CALL	HEX_ASC	;
D1C5 3A 30 20	LDA	CONT_CAR	;
D1C8 3C	INR	A	;
D1C9 32 30 20	STA	CONT_CAR	;
D1CC FE 04	CPI	04	; Transmite dados (4 por linha)
D1CE CC 49 D3	CZ	CRET	;
D1D1 13	INX	D	;
D1D2 7A	MOV	A,D	;
D1D3 FE 84	CPI	MEM_FIM	
D1D5 C2 C1 D1	JNZ	DADO	
D1D8			
D1D8 AF	XRA	A	
D1D9 32 30 20	STA	CONT_CAR	
D1DC CD 40 D3	CALL	SINAL_FIM	; Transmite fechamento de arquivo ('Z')
D1DF			
D1DF 3E 00	MVI	A, 00H	
D1E1 06 00	MVI	B, 00H	; Mensagem 'OPTE'
D1E3 21 3F D4	LXI	H, OPTE	; no display
D1E6 CD B7 02	CALL	MON\$_OUTPT	
D1E9			
D1E9 3E 0E	MVI	A, 0EH	
D1EB 30	SIM		
D1EC FB	EI		
D1ED CD E7 02	CALL	MON\$_RDKBD	; Espera decisao para continuar
D1F0 FE 0C	CPI	0CH	
D1F2 CA FD D1	JZ	RET_DIFR	; 'C' e' retorno com coleta
D1F5 FE 0D	CPI	0DH	
D1F7 CA 17 D2	JZ	RET_DISP	; 'D' e' retorno so' com display
D1FA CD 15 02	CALL	MON\$_ERRO	
D1FB			

; 7.1 - ROTINA DE REARMAZENAMENTO DE PARAMETROS PARA A VOLTA COM C

```

D1FD      RET_DIFR:
D1FD
D1FD 3E 00      MVI    A,00H
D1FF 06 00      MVI    B,00H      ; MENSAGEM 'DIFR.'
D201 21 43 D4    LXI    H,OPTE+4      ; NO DISPLAY
D204 CD B7 02    CALL   MON$_OUTPT
D207
D207
D207 11 50 28    LXI    D,TMP      ; PASSA TMP PARA HL
D20A ED          LHLX
D20B 11 40 28    LXI    D,TEMPO      ; PASSA HL PARA TEMPO
D20E D9          SHLX
D20F CD 56 D3    CALL   RELOGIO      ; CHAMA SUBROTINA DE PROGRAMACAO DE TEMPO
D212
D212 3E 18      MVI    A,18H      ; PREPARA ACUMULADOR COM MASCARA PARA INSTRUCAO SIM
                                ; 'RESETANDO' POSSIVEIS INTERRUPTOS PENDENTES
D214 C3 23 D2    JMP     RETORNO
D217
D217

```

; 7.2 - ROTINA DE REARMAZENAMENTO NO CASO DE VOLTA SEM COLETA

```

D217      RET_DISP:
D217      MVI    A,00H
D219      MVI    B,00H      ; MENSAGEM 'DISP.'
D21B 21 47 D4    LXI    H,OPTE+8      ; NO DISPLAY
D21E CD B7 02    CALL   MON$_OUTPT
D221
D221 3E 1B      MVI    A,1BH      ; PREPARA ACUMULADOR COM MASCARA DE 'SIH'
                                ; 'RESETANDO' E DEIXANDO DISPONIVEL APENAS A 7.5
D223
D223      RETORNO:
D223
D223 30          SIM
D224 3E 40      MVI    A,40H      ; RESET INTERNO DA 8251
D226 D3 C0      OUT     COM_STAT_51
D228            POP_REG
D228 E1          +      POP     H
D229 D1          +      POP     D
D22A C1          +      POP     B      ; MACRO DE RECUPERACAO DE REGISTRADORES
D22B F1          +      POP     PSW
D22C FB
D22D C9          RET

```

```

*****
D22E
;      8 -  BIBLIOTECA DE SUBROTINAS
D22E
*****
D22E
D22E
;      8.1 - ROTINA QUE COLOCA, NO CAMPO DE ENDEREÇOS DO DISPLAY,
;              OS DADOS APONTADOS POR BUFF
D22E      ENDER;
D22E 21 10 28      LXI      H,BUFF
D231 AF            XRA      A
D232 06 00         MVI      B,0H
D234 CD B7 02      CALL     MON$_OUTPT
D237 C9            RET
D238

;      8.2 - ROTINA QUE COLOCA, NO CAMPO DE DADOS DO DISPLAY,
;              OS CODIGOS APONTADOS POR BUFF+4
D238      DADOS;
D238 21 14 28      LXI      H,BUFF+4
D23B 3E 01         MVI      A,1H
D23D 06 00         MVI      B,0H
D23F CD B7 02      CALL     MON$_OUTPT
D242 C9            RET
D243

;      8.3 - ROTINA QUE ESPERA O DADO, ARMAZENA NO "BUFFER"
;              E MOSTRA NO DISPLAY.
D243      DIGITO;
D243 CD E7 02      CALL     MON$_RDKBD
D246 FE 11         CPI      11H          ; PARA SAIR TECLE "NEXT" (next e' 11h)
D248 CA FA D2      JZ       SAI
D24B FE 13         CPI      13H          ; PARA LIMPAR DISPLAY TECLE "SUBST MEM"
D24D CA 68 D2      JZ       TECLADO
D250 32 15 28      STA      BUFF+5      ;COLOCA DIGITO LIDO NA POSICAO MAIS A
D253 CD 38 D2      CALL     DADOS        ; DIREITA DO DISPLAY
D256 C9            RET
D257

;      8.4 - ESTA ROTINA FORMATA OS DADOS QUE CHEGAM
;              ( CADA TECLA GERA UM BYTE )
D257      FORMAT;
D257 23            INX      H
D258 7E            MOV      A,M          ; PEGA O DIGITO MAIS SIGNIFICATIVO
D259 0F            RRC
D25A 0F            RRC
D25B 0F            RRC
D25C 0F            RRC
D25D 47            MOV      B,A
D25E 23            INX      H
D25F 7E            MOV      A,M          ; PEGA DIGITO MENOS SIGNIFICATIVO

D260 B0            ORA      B
D261 FE 15         CPI      15H
D263 C2 67 D2      JNZ      VOLTA
D266 AF            XRA      A
D267              VOLTA;
D267 C9            RET

```

D269

; 8.5 - RECEBIMENTO DO TEMPO PROGRAMADO DO TECLADO
;
- O TEMPO ENTRA EM DECIMAL -

D269

D268

TECLADO:

D268

D268 21 10 28

LXI H,BUFF

; BUFF APONTA O "BUFFER" DE CODIGOS PARA O DISPLAY

D26B 3E 06

MVI A,06H

D26D 06 15

MVI B,15H

; 15H E' O CODIGO DE BRANCO PARA O DISPLAY

D26F

ALI:

D26F 70

MOV M,B

;

D270 23

INX H

; PREENCHE O BUFFER COM BRANCOS

D271 3D

DCR A

;

D272 C2 6F D2

JNZ ALI

D275 CD 2E D2

CALL ENDER

; LIMPA CAMPO DE ENDEREÇOS

D278 CD 38 D2

CALL DADOS

; LIMPA CAMPO DE DADOS

D27B 3E 0E

MVI A,0EH

; DESMASCARAMENTO DA INTERRUPCAO 5.5

D27D 30

SIM

D27E FB

EI

D27F

D27F

PRIM:

D27F CD 43 D2

CALL DIGITO

D282

SEG:

D282 3A 15 28

LDA BUFF+5

D285 32 14 28

STA BUFF+4

D288 CD 43 D2

CALL DIGITO

D28B

TERC:

D28B 3A 14 28

LDA BUFF+4

D28E 32 13 28

STA BUFF+3

D291 3A 15 28

LDA BUFF+5

D294 32 14 28

STA BUFF+4

D297 CD 43 D2

CALL DIGITO

D29A CD 2E D2

CALL ENDER

D29D

QUART:

D29D 3A 13 28

LDA BUFF+3

D2A0 32 12 28

STA BUFF+2

D2A3 3A 14 28

LDA BUFF+4

D2A6 32 13 28

STA BUFF+3

D2A9 3A 15 28

LDA BUFF+5

D2AC 32 14 28

STA BUFF+4

D2AF CD 43 D2

CALL DIGITO

D2B2 CD 2E D2

CALL ENDER

D2B5

QUINT:

D2B5 3A 12 28

LDA BUFF+2

D2B8 32 11 28

STA BUFF+1

D2BB 3A 13 28

LDA BUFF+3

D2BE 32 12 28

STA BUFF+2

D2C1 3A 14 28

LDA BUFF+4

D2C4 32 13 28

STA BUFF+3

D2C7 3A 15 28

LDA BUFF+5

D2CA 32 14 28

STA BUFF+4

D2CD CD 43 D2

CALL DIGITO

D2D0 CD 2E D2

CALL ENDER

D2D3

SEXT:

D2D3 3A 11 28

LDA BUFF+1

D2D6 32 10 28

STA BUFF

D2D9 3A 12 28

LDA BUFF+2

D2DC 32 11 28

STA BUFF+1

D2DF 3A 13 28

LDA BUFF+3


```
D2E2 32 12 28      STA  BUFF+2
D2E5 3A 14 28      LDA  BUFF+4
D2E8 32 13 28      STA  BUFF+3
D2EB 3A 15 28      LDA  BUFF+5
D2EE 32 14 28      STA  BUFF+4
D2F1 CD 43 D2      CALL DIGITO
D2F4 CD 2E D2      CALL  ENDER
D2F7
D2F7 C3 7F D2      JMP  PRIM
D2FA
D2FA E1            POP  H
D2FB F3            DI
D2FC C9            RET
D2FD
```

!PARA COMPENSAR SAIDA FORCADA DA ROTINA DIGITO

```
D2FD
D2FD 47
D2FE 0F
D2FF 0F
D300 0F
D301 0F
D302 CD 0A D3
D305 78
D306 CD 0A D3
D309 C9
D30A
```

; 8.6 - ROTINA DE CONVERSAO HEX-ASCII

```
HEX_ASC:
MOV  B,A
RRC
RRC
RRC
RRC
CALL CONV
MOV  A,B
CALL CONV
RET
```

```
D30A
D30A E6 0F
D30C FE 0A
D30E DA 13 D3
D311 C6 07
D313
D313 C6 30
D315 4F
D316 CD 1D D3
D319 CD 29 D3
D31C C9
D31D
```

; 8.7 - ESTA ROTINA CONVERTE DIGITO

```
CONV:
ANI  0FH
CPI  0AH ; VERIFICA SE E' MENOR QUE 0A
JC   NUM ; SIM, ENTAO NUMERO, SOMA 30
ADI  07H ; NAO, LETRA, SOMA 37
NUM:
ADI  30H
MOV  C,A
CALL FLUX ; VERIFICA FLUXO DE DADOS
; NA USART
CALL VAI_VAX ; SE TUDO BEM, MANDA O ASCII
RET
```

; 8.8 - ROTINA DE MONITORAMENTO DE FLUXO DE DADOS

```
D31D
D31D DB C0
D31F E6 02
D321 C8
D322 7E
D323 FE 13
D325 CC 32 D3
D328 C9
D329
```

```
; NA USART.
FLUX:
IN   COM_STAT_51 ; LE STATUS
ANI  02H         ; VERIFICA SE TEM DADO VINDO
; DE FORA
RZ   ; SE NAO, VOLTA
MOV  A,H         ; SE TIVER, LE
CPI  13H         ; VE SE E' 'S
CZ   VEM_VAX     ; SE SIM, ESPERA 'Q
RET
```

; 8.9 - ROTINA QUE ENVIA DADO PARA FORA

```
D329
D329 DB C0
D32B E6 01
D32D CA 29 D3
D330 71
D331 C9
```

```
VAI_VAX:
IN   COM_STAT_51 ; LE STATUS DA USART
ANI  01H         ; VERIFICA SE JA' ENVIOU
; DADO ANTERIOR
JZ   VAI_VAX     ; SE NAO, ESPERA
MOV  M,C         ; SE SIM, ENVIA
RET
```

```

; 8.10 - ROTINA QUE ESPERA ^Q PARA CONTINUAR TRANSMISSAO
VEM_VAX:
D332      IN      COM_STAT_51      ; LE STATUS
D332  DB  C0
D334  E6  02      ANI      02H      ; VERIFICA SE CHEGOU DADO
D336  CA  32 D3   JZ      VEM_VAX   ; NAO, ESPERA
D339  7E          MOV      A,M      ; SIM, LE
D33A  FE  11      CPI      11H      ; E' ^Q ?
D33C  C2  32 D3   JNZ     VEM_VAX   ; NAO, ESPERE
D33F  C9          RET              ; SIM RETORNE
D340

; 8.11 - ROTINA QUE ENVIA UM ^Z PARA FECHAR ARQUIVO
SINAL_FIM:
D340      MVI      C,1AH
D340  OE  1A
D342  CD  1D D3   CALL     FLUX
D345  CD  29 D3   CALL     VAI_VAX
D348  C9          RET
D349

; 8.12 - ROTINA QUE ENVIA CARRIAGE RETURN E LINE FEED
; - SEPARADORES -
D349      CRET:
D349  OE  0D      MVI      C,0DH
D34B  CD  1D D3   CALL     FLUX
D34E  CD  29 D3   CALL     VAI_VAX
D351  AF          XRA      A
D352  32  30 20   STA      CONT_CAR
D355  C9          RET
D356
D356

; 8.13 - ROTINA DE PROGRAMACAO DO TEMPO DE COLETA
RELOGIO:
D356
D356
D356

; 8.13.1 - PROGRAMACAO DO 8155, ESTE RECEBE O CLOCK DO SISTEMA - 3
; E DIVIDE-O POR 48 GERANDO UM CLOCK DE 64Khz Para o 8253
D356      MVI      A,030H      ; LSB DO TIMER COUNT (8155)
D356  3E  30      OUT      2CH
D358  D3  2C      MVI      A,40H      ; - MSB DO TIMER COUNTER: 00
D35A  3E  40      ; - MODO: ONDA QUADRADA CONTINUA
D35C  D3  2D      OUT      2DH
D35E  3E  CF      MVI      A,0CFH      ; - START TIMER DA 8155
; - disable interrupt port
; - todas portas out
D360  D3  28      OUT      28H
D362

; 8.13.2 - PROGRAMACAO DO 8253 - CONTADOR 0, RECEBE 64K E GERA 1h
D362      MVI      A,3EH      ; CONTROL WORD DA 8253
D362  3E  3E      ; - COUNTER-0
; - LOAD LSB E MSB
; - MODE 3 , BINARY
D364  D3  A3      OUT      0A3H
D366  3E  2B      MVI      A,2BH      ; LSB          F/DAR 1 SEG NA SAIDA
D368  D3  A0      OUT      0A0H
D36A  3E  FA      MVI      A,0FAH      ; MSB
D36C  D3  A0      OUT      0A0H
D36E

; 8.13.3 - SINCRONISMO DO TIMER, E' NECESSARIO PELO FATO DO 8253 S
; CARREGAR O CONTADOR APOS UM PULSO COMPLETO DO CLOCK, O
; FINO C5 DA PORTA C ESTA' LIGADO NA SAIDA DO CONTADOR ZERO

```

```

D36E
D36E
D36E DB 23
D370 E6 20
D372 C2 6E D3
D375

; 8.13.4 - PROGRAMACAO DO 8253 - CONTADOR 1, SEU CLOCK E' A SAIDA
; - OS DADOS ESTAO EM TEMPO E TEMPO+1

D375
D375 3E 70
MVI A,70H ; CONTROL WORD DO 8253
; - COUNTER-1
; - LOAD LSB E MSB
; - MODE 0, BINARY

D377 D3 A3
D379 3A 40 28
D37C D3 A1
D37E 3A 41 28
D381 D3 A1
D383
D383
D383
D383 DB 23
D385 E6 20
D387 CA 83 D3
D38A
D38A

D38A
D38A DB 23
D38C E6 20
D38E C2 8A D3
D391 C9
D392
D392
D392

ZERO:
IN PORT_DO
ANI 20H ; MONITORA ESTADO DO CLOCK
JZ ZERO

UM:
IN PORT_DO ; AQUI PASSOU DE ZERO PARA UM
ANI 20H
JNZ UM ; QUANDO SAIR E' PORQUE VOLTOU A ZERO
RET

```

; 8.14 - ROTINA DE CONVERSAD BCD PARA BINARIO

```

D392
D392
D392
D392 F5 + PUSH PSW
D393 C5 + PUSH B
D394 D5 + PUSH D
D395 E5 + PUSH H
; MACRO DE SALVAMENTO DE REGISTRADORES
;

D396 3A 20 28 LDA ADRS
D399 E6 0F ANI 0FH
D39B 32 00 28 STA 2800H
D39E AF XRA A
D39F 32 01 28 STA 2801H
D3A2 3A 22 28 LDA ADRS+2
D3A5 E6 0F ANI 0FH
D3A7 0E 04 MVI C,04H
D3A9 CD 09 D4 CALL L1
D3AC DA E0 D3 JC L4
D3AF 3A 21 28 LDA ADRS+1
D3B2 47 MOV B,A
D3B3 E6 F0 ANI 0F0H
D3B5 0F RRC
D3B6 0F RRC
D3B7 0F RRC
D3B8 0F RRC

```

D3B9 0E 03	MVI	C,03H	;	ROTINA DE CONVERSAO BCD - BINARIO
D3BB CD 09 D4	CALL	L1	;	Dado a ser convertido esta em ADRS
D3BE DA E0 D3	JC	L4	;	Resultado e' colocado em ADRD
D3C1 78	MOV	A,B	;	
D3C2 E6 0F	ANI	0FH	;	Versao orisinal (macro) P/ video por
D3C4 0E 02	MVI	C,02H	;	C. A. RUGGIERO
D3C6 CD 09 D4	CALL	L1	;	
D3C9 DA E0 D3	JC	L4	;	
D3CC 3A 20 28	LDA	ADRS	;	
D3CF E6 F0	ANI	0F0H		
D3D1 0F	RRC			
D3D2 0F	RRC			
D3D3 0F	RRC			
D3D4 0F	RRC			
D3D5 0E 01	MVI	C,01H		
D3D7 CD 09 D4	CALL	L1		
D3DA D2 E3 D3	JNC	L3		
D3DD C3 E0 D3	JMP	L4		
D3E0 CD 15 02	L4:	CALL	0215H	
D3E3 3A 22 28	L3:	LDA	ADRS+2	
D3E6 E6 F0	ANI	0F0H		
D3EB CA 00 D4	JZ	L7		
D3EB FE F0	CPI	0F0H		
D3ED C2 FD D3	JNZ	L10		
D3F0 2A 00 28	LHLD	2800H		
D3F3 7D	MOV	A,L		
D3F4 2F	CMA			
D3F5 6F	MOV	L,A		
D3F6 7C	MOV	A,H		
D3F7 2F	CMA			
D3F8 67	MOV	H,A		
D3F9 23	INX	H		
D3FA C3 03 D4	JMP	L9		
D3FD CD 15 02	L10:	CALL	0215H	
D400 2A 00 28	L7:	LHLD	2800H	
D403 22 30 28	L9:	SHLD	ADRD	
D406 C3 3A D4	JMP	L11		
D409 5F	L1:	MOV	E,A	
D40A AF	XRA	A		
D40B 57	MOV	D,A		
D40C 6B	L12:	MOV	L,E	
D40D 62	MOV	H,D		
D40E 18	SDEL			
D40F 7A	MOV	A,D		
D410 B7	ORA	A		
D411 DA 32 07	JC	SRET	;	*
D414 18	SDEL			
D415 7A	MOV	A,D		
D416 B7	ORA	A		
D417 DA 32 07	JC	SRET	;	*
D41A 19	DAD	D		
D41B 7C	MOV	A,H		
D41C B7	ORA	A		
D41D DA 32 07	JC	SRET	;	*
D420 EB	XCHG			
D421 18	SDEL			
D422 7A	MOV	A,D		
D423 B7	ORA	A		
D424 DA 32 07	JC	SRET	;	*

D427	0D			DCR	C	
D428	C2 0C D4			JNZ	L12	
D42B	2A 00 2B			LHLD	2800H	
D42E	19			DAD	D	
D42F	7C			MOV	A,H	
D430	E7			ORA	A	
D431	DA 32 07			JC	SRET	/*
D434	22 00 2B			SHLD	2800H	
D437	C3 1C 06			JMP	FRET	
D43A			L11:	POP_REG		
D43A	E1	+	POP	H		
D43B	D1	+	POP	D		;
D43C	C1	+	POP	B		; MACRO DE RECUPERACAO DE REGISTRADORES
D43D	F1	+	POP	PSW		;
D43E	C9			RET		

```

*****
D43F
; 9 - AREA DE DADOS
D43F
*****
D43F
0000 TXRX_51 EQU 00000H
8000 MEM_DSPL_I EQU 08000H ;ENDERECO INICIAL DA MEMORIA
93FF STK_ADR EQU 093FFH ;ENDERECO DO STACK POINTER.
2000 SHIFT EQU 02000H
2003 CONTX EQU 02003H
2004 TEMP1 EQU 02004H
2005 TEMP2 EQU 02005H
2006 DELAY EQU 02006H
2007 TEMPS EQU 02007H
2008 CONT_ROTATE EQU 02008H
2009 CONT_VAR_M EQU 02009H
200A CONT_VAR_A EQU 0200AH
200B BYTE_DSPLD EQU 0200BH
2020 PRIME EQU 02020H
2021 SEGU EQU 02021H
2030 CONT_CAR EQU 02030H
2060 FLAG1 EQU 02060H
2810 BUFF EQU 02810H
2820 ADRS EQU 02820H
2830 ADDR EQU 02830H
2840 TEMPO EQU 02840H
2850 TMP EQU 02850H
0732 SRET EQU 0732H ;MONITOR V2.2
061C FRET EQU 061CH ;MONITOR V2.2
0215 MON$_ERRO EQU 0215H
02B7 MON$_OUTPT EQU 02B7H ;OUTPUT TO 8085 DISPLAY (SEE MANUAL)
02E7 MON$_RDKBD EQU 02E7H ;READ INTO REG A FROM HEX KEYPAD (RST 5.5)
0363 MON$_UPDAD EQU 0363H ;UPDATE ADDR_FIELD FROM REG D_E
036E MON$_UPDDT EQU 036EH ;UPDATE DATA_FIELD FROM REG A
05F1 MON$_DELAY EQU 05F1H ;DELAY TIME SET IN REG D_E
0020 PORT_A0 EQU 020H ;DE CONTROLE
0021 PORT_B0 EQU 021H ;PORTA DE SAIDA PARA O OSCILOSCOPIO - EIXO Y
0022 PORT_C0 EQU 022H ;IDEM.-EIXO X.
0023 PORT_D0 EQU 023H ;PORTA DE ENTRADA DA CHAVE A/M E DOS BOTOES '+', E '-'
0028 PORT_COMMAND_EXP_RAM EQU 028H
002C PORTLOW_TIM_EXP_RAM EQU 2CH
002D PORTHIGH_TIM_EXP_RAM EQU 2DH
0084 MEM_FIM EQU 084H
00B0 PORT_DAT EQU 0B0H ;ENTRADA DE DADOS
00C0 COM_STAT_51 EQU 0C0H
00D0 HDAT_ADR EQU 0D0H ;***HIGH BYTE ADDR. SHIFT TABLE
D43F
D43F
D43F 00 12 07 0E OPTES: DB 00H,12H,07H,0EH,0DH,13H,0FH,14H,0DH,13H,05H,12H,15H,15H
D443 00 13 0F 14
D447 00 13 05 12
D44B 15 15
D44D 0C 0A 14 14 DATA: DB 0CH,0AH,14H,14H
D451
D451 END

```

CROSS REFERENCE

LABEL	VALUE	REFERENCE
ADD_VAR_M	D1A3	330 -377
ADDRD	2830	215 790 1235 -1306
ADRS	2820	197 199 201 1183 1188 1193 1208 1219
	2820	-1305
AJST0	D043	447 -485
AJST1	D04A	449 -489
AJST10	D0B1	467 -564
AJST11	D0C0	469 -575
AJST12	D0D0	471 -588
AJST13	D0E1	473 -601
AJST14	D0F2	475 -614
AJST15	D102	477 -626
AJST16	D111	479 -637
AJST2	D053	451 -495
AJST3	D05D	453 -502
AJST4	D068	455 -510
AJST5	D074	457 -519
AJST6	D082	459 -530
AJST7	D08F	461 -540
AJST8	D09B	463 -549
AJST9	D0A3	465 -554
AJUST_16	D208	-426
ALI	D26F	-970 974
AUTOM	D18E	351 -365
BUFF	2810	193 916 925 939 967 984 985 988
	2810	989 990 991 995 996 997 998 999
	2810	1000 1004 1005 1006 1007 1008 1009 1010
	2810	1011 1015 1016 1017 1018 1019 1020 1021
	2810	1022 1023 1024 -1304
BYTE_DSPLD	200B	78 295 487 493 500 508 517 527
	200B	537 546 552 561 572 584 598 611
	200B	623 634 643 -1299
CARR	D169	751 -759
CARREGA	D150	97 -740
CMP_11	D1F3	412 -415
CNV_BCDTOBIN	D392	202 -1177
COLETA	D11E	90 -669
COMECO	D18B	-787
COM_STAT_51	00C0	781 785 883 1063 1074 1083 -1326
CONTX	2003	72 293 297 299 -1291
CONT_CAR	2030	789 818 820 829 1105 -1302
CONT_ROTATE	2008	83 398 400 405 406 408 423 -1296
CONT_VAR_A	200A	76 387 -1298
CONT_VAR_M	2009	75 338 352 357 363 367 371 379
	2009	383 409 -1297
CONV	D30A	1041 1043 -1047
CRET	D349	799 810 822 -1100
CRIA_SHIFT	D1B8	337 -386
DADO	D1C1	-815 826
1		
DADOS	D238	-924 940 976
DATA	D44D	762 -1334

DELAY	2006	81	315	317	320	-1294			
DIGITO	D243	-933	982	986	992	1001	1012	1025	
DSFL	D0E9	-261	344	389	395	425	432	435	439
DSPL1	D101	153	-278						
ENDER	D22E	-915	975	993	1002	1013	1026		
FLAG1	2060	84	142	748	-1303				
FLUX	D31D	1055	-1062	1094	1102				
FORMAT	D257	194	198	200	-945				
FRET	061C	1267	-1310						
GETSHIFT	D129	300	-311						
HDAT_ADR	00D0	284	-1327						
HEX_ASC	D2FD	794	798	805	809	817	-1035		
L1	D409	1191	1201	1206	1215	-1237			
L10	D3FD	1223	-1233						
L11	D43A	1236	-1268						
L12	D40C	-1240	1260						
L3	D3E3	1216	-1219						
L4	D3E0	1192	1202	1207	1217	-1218			
L7	D400	1221	-1234						
L9	D403	1232	-1235						
LOOP	D0DB	-225	231						
MEMORY	H 0000	0							
MEM_DSPL_I	8000	224	279	682	812	-1288			
MEM_FIH	0084	825	-1324						
MON\$_DELAY	05F1	180	181	182	-1316				
MON\$_ERRD	0215	114	196	845	-1311				
MON\$_OUTPT	02B7	105	147	177	208	213	763	835	854
	02B7	874	919	928	-1312				
MON\$_RDKBD	02E7	109	840	934	-1313				
MON\$_UPDAD	0363	-1314							
MON\$_UPDDT	036E	-1315							
NARG	0000	0							
NEW_ROTATE	D1C3	-392	401						
NEXT	D10C	-286	305						
NUM	D313	1050	-1052						
OPTE	D43F	104	146	176	207	212	834	853	873
	D43F	-1330							
OSCILOS	D10F	-292	488	494	501	509	518	529	539
	D10F	548	553	563	574	587	600	613	625
	D10F	636	645						
POOL	D161	340	-342						
FORTHIGH_TIM	002B	772	-1323						
PORTLOW_TIM	002C	769	-1322						
PORT_A0	0020	63	-1317						
PORT_B0	0021	296	-1318						
PORT_C0	0022	294	-1319						
PORT_COMMAND	0028	766	774	-1321					
PORT_ID0	0023	312	1143	1162	1168	-1320			
PORT_DAT	00B0	677	-1325						
PRIM	D27F	-981	1028						
PRIME	2020	136	268	271	754	-1300			

QUART	D29D	-994							
QUINT	D2B5	-1003							
REINIC	D1FC	414	418	-420					
RELOGIO	D356	220	861	-1111					
RETOR	D143	687	689	-706	721				
RETORNO	D223	865	-879						
RET_DIFR	D1FD	842	-849						
RET_DISP	D217	844	-870						
SAI	D2FA	936	-1029						
SBT_VAR_M	D167	326	-348						
SEG	D2B2	-983							
SEGU	2021	138	270	756	-1301				
SEXT	D2D3	-1014							
SHIFT	2000	71	280	343	421	430	434	438	-1290
SHIFT_10	D21F	428	-436						
SHIFT_NMB	D1D9	-403							
SINAL_FIM	D340	830	-1092						
SINC	D36E	-1142	1145						
SRET	0732	1245	1249	1253	1258	1265	-1309		
STACK	S 0000	0							
STK_ADR	93FF	61	-1289						
TECLADO	D268	184	938	-965					
TEMP1	2004	73	314	324	328	335	349	-1292	
TEMP2	2005	74	391	393	397	-1293			
TEMP5	2007	77	353	366	372	378	384	-1295	
TEMPO	2840	217	859	1155	1157	-1307			
TERC	D28B	-987							
TERC_BYTE	D149	704	-715						
TEST_A_M	D14F	318	-334	381	385				
TEST_BIT_0	D147	-327	358	361	364	369	373		
TEST_Z_DCR	D182	355	-359						
THP	2850	137	139	755	757	801	857	-1308	
TXRX_51	C000	793	797	804	808	813	-1287		
UM	D38A	-1167	1170						
VAI_VAX	D329	1057	-1073	1077	1095	1103			
VARR_COL	D083	111	-173						
VARR_SO	D062	113	-135	274					
VEH_VAX	D332	1069	-1082	1085	1088				
VOLTA	D267	957	-959						
ZERO	D383	-1161	1164						

1

4

APÊNDICE III

LISTAGEM DO PROGRAMA DE RECEBIMENTO DE DADOS

```
C
C      PROGRAMA CO(LETA)
C
C
C      PROGRAMA DE RECEBIMENTO DE DADOS DAS INTERFACES SERIAIS
C      DO ANALIZADOR DE ALTURA DE PULSOS
C      Edson LIE Julho 83
C
C      CHARACTER NOME*40, GUARDA*1, CONTINUA*1
C      BYTE A(4), B(4), C(4)
C      INTEGER*4 TEMPRO, TEMRES, DADO, TELA(256)
C      EQUIVALENCE(A, TEMPRO), (B, TEMRES), (C, DADO)
C      CALL ASSIGN(1, 'tt:')
C
C      abertura do arquivo
C
C      CALL CLEAR
C      TYPE 199
199  FORMAT('1', 61X, 'L I E', //, 47X, 'Lab. de Instrumentacao
      1Eletronica', //, '-----', //,
      11X, 'ANALIZADOR', 66X, 'DFCH', //,
      17X, '256 CANAIS', 61X, 'USP', //,
      1'-----', //, 55X, 'S A X S', //,
      141X, 'Lab. de Difracao de R-X a Baixo Angulo', 10(//))
C      type 200
200  format(%, '$ Arquivo a ser criado: ')
C      accept 300, nome
300  format(a)
      open(UNIT=10, FILE=nome, STATUS='NEW')
C
C      leitura dos tempos programado e transcorrido
C
C      A(4)='00'X
C      A(3)='00'X
C      B(4)='00'X
C      B(3)='00'X
C      READ(1, 100) A(2), A(1)
100  FORMAT(2Z2)
      READ (1, 100) B(2), B(1)
C
C      LEITURA DOS DADOS COLETADOS
C
C      C(4)='00'X
C      DO 180 J=1, 257
C          READ(1, 140, END=185) C(1), C(2), C(3)
140  FORMAT(3Z2)
      WRITE(10, 165) DADO
165  FORMAT(1X, I8)
180  CONTINUE
185  ACCEPT 400, CONTINUA
400  FORMAT(A)
      IF (CONTINUA.EQ.'C') GO TO 399
      TYPE 500
500  FORMAT(1X, 'TECLE C PARA CONTINUAR')
      GO TO 185
399  REWIND 10
```

```
TYPE 110,TEMPRO
110  FORMAT('1',10('/),' Programados',i8,' ses.')
```

TYPE 130,TEMRES

```
130  FORMAT(3(/),40X,'Restam',i8,' ses.',10(/))
    PAUSE 'TECLE C PARA CONTINUAR'
    CALL  CLEAR
    READ(10,181)(TELA(J),J=1,256)
181  FORMAT(1X,i8)
    I=1
    DO 184 J=1,51
    TYPE 182,I,TELA(I),TELA(I+1),TELA(I+2),TELA(I+3),TELA(I+4)
182  FORMAT(1X,'CANALIS ',i3,'...',5(4X,i8))
    IF(J.EQ.22.OR.J.EQ.43)PAUSE 'TECLE C PARA CONTINUAR'
    I=I+5
184  CONTINUE
    TYPE 186,TELA(256)
186  FORMAT(1X,'CANALIS 256...',4X,i8)
190  TYPE 191
191  FORMAT(1X,/,/,',$', ' QUER GUARDAR O ARQUIVO? (S/N)')
    ACCEPT 192,GUARDA
192  FORMAT(A)
    IF(GUARDA.EQ.'S') close(unit=10,STATUS='SAVE')
    CLOSE(UNIT=10,STATUS='DELETE')
    CALL EXIT
    END
```

C

C ROTINA PARA ENVIAR COMANDOS

C VT52 PARA LIMPAR A TELA DE VIDEO.

C

```
SUBROUTINE CLEAR
PARAMETER ESCAPE = '33'o
OPEN (UNIT=99, FILE='SYS$COMMAND', STATUS='NEW',
1     CARRIAGE CONTROL='NONE', FORM='UNFORMATTED')
WRITE (99)     Escape, 'H', Escape, 'J'
RETURN
END
```

REFERÊNCIAS

- 1 - Borkowski, C.J.; Kopp, M.K. - New Type of Position - Sensitive Detectors of Ionizing Radiation Using Risetime Measurement, Rev. of Scientific Instruments, Vol. 39, No. 10, October, 1968.
- 2 - G. Charpak - Applications of Proportional Chambers to Solve Problems in Medicine and Biology - Wire chamber Conference - Viena, 1978.
- 3 - R. Hamlin et. ali, J. Appl. Cryst., 14 (1981)85
- 4 - M.J. Tricker et ali, Nucl. Instr. and Meth., 135 (1976)11
- 5 - J. Stahler and G. Presser, Nucl. Instr. and meth. 177 (1980)427
- 6 - Paul, D.A. - On the analyses of decay time coincidence J. of Physics E: Scientific Instruments, 1977, Vol. 10.
- 7 - Gabriel, A.; Dupont, Y. - A Position Sensitive Proportional Detector for X-ray Crystallography, The review of Scientific Instruments, Vol. 43, Number 11, 1972.
- 8 - Wijnaendts, R.W., Harink, H.C. - A position dependent Particle Counter Using Microchannel Plates, J. of Physics E: Scient. Instr., Vol. 9, 1976.
- 9 - Borso, C.S.; Danyluk, S.S. - Application of a directly exposed self Scanning Photodiode array as a linear position sensitive detector in a small-Angle X-ray Scattering instrument, Review Sci. Instrum., 51(12), dec. 1980.

- 10 - Slaets, J.; Craievich, A.F. - A Step Scanning Controller for the Elimination of Drift Effects in SAXS measurements, J. Physics E, Vol. 9, 1976.
- 11 - Slaets, J.; Moreira, E. - Um Controlador de Varredura de Passo Programável, Cienc. e Cultura 34(7), 1982.
- 12 - Buras, B.; Marr, G.V. - European Synchrotron Radiation Facility. Supp. III Instr. ESF France, 1979.
- 13 - Tennelec, PSD 1100 Instruction Manual, Tennelec Inc. Tennessee
- 14 - Tennelec, TC 861 Time to Amplitude Converter, idem
- 15 - Intel, SDK-85 MCS-85 System design Kit. Intel Co. Santa Clara, 1979.
- 16 - Intel, MCS-80/85 Family user's Manual, Intel Co., Santa Clara, 1979.
- 17 - Analog Devices, Data — Acquisition Databook 1982. Vol. 1, Norwood.
- 18 - Zuffo, J.A. - Microprocessadores: Dutos do sistema, Técnicas de interface..., Ed. Edgard Blucher, 1981.
- 19 - Morris, R.; Miller, J. - Projetos com Circuitos Integrados TTL, Ed. Guanabara Dois, Rio de Janeiro, 1978.
- 20 - Williamson, T. - Designing Microcontroller Systems for Electrically Noisy Environments, Ap. Notes Intel Co.
- 21 - Ruggiero, C. A. - Programa Monitor para o 8085A, Ciência e Cultura, 35(7), julho de 1983.

- 22 - Slaets, J. - Sistema Modular para Coleta de Dados e Controle de Processos, Ídem.
- 23 - Ruggiero, C.A. - Novas instruções do 8085A. Ciência e Cultura, 34(7), julho de 1982.
- 24 - Borelli, U.; Moreira, E.; Slaets, J. - Aquisição de Dados com Eliminação de Ruído Aleatório. Cien. e Cultura, 35(7), 1983.
- 25 - Digital, VAX-11 FORTRAN Language Reference Manual, Digital Equipment Co. Massachusetts, 1982.
- 26 - Northern, Pulse Height Analyzer — Instr. Manual Northern Scientific, Inc. Wisconsin, 1970.
- 27 - Guinier, A. - Small-angle scattering of X-ray. New York. John Wiley & Sons, Inc. London. Chapman & Hall, Ltd., 1955.
- 28 - Klein, P.; Meiners, D. - A Multichannel data acquisition system for Fast Analogue signals, J. Phys. E: Sci. Instr., Vol. 12, 1979.
- 29 - Elmsquist, K. A. & Fullmer, H. - Standard Specification for S-100 Bus Interface Devices, IEEE Computer, 28, 52, 1979.