

UNIVERSIDADE DO PORTO
FACULDADE DE ENGENHARIA

PRODEP - Medida 4.3/7/6/92/93

Estágio 3.10 - "Aquisição de Dados de Processos Industriais"

Vera Gomes Mata



H

68(047-2)/LCC/1992/11A^{TV}
13 10 - 25

REFERENCIA DO PROJECTO -
PRODEP - Medida 4.3/7.6/92/93

PARECER TÉCNICO

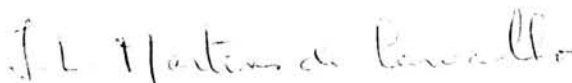
Acompanhei os trabalhos de estágio realizados no Instituto de Sistemas e Robótica pela aluna Vera Lúcia Gomes Mata, bem como apreciei os trabalhos produzidos e o relatório oportunamente apresentado.

Com base no acompanhamento referido, considero que aqueles trabalhos reflectem o trabalho efectivamente realizado, revelando empenhamento da estagiária.

Sou portanto de parecer que foram plenamente atingidos os objectivos inicialmente propostos.

Porto, 30 de Setembro de 1993

O Supervisor do Estágio no Instituto de Sistemas e Robótica



(Prof. Jorge Leite Martins de Carvalho)

O Supervisor do Estágio na Faculdade de Engenharia da Universidade do Porto



(Prof. Sebastião Feyo de Azevedo)

INSTITUTO DE SISTEMAS E ROBÓTICA
FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO
Outubro de 1992

AQUISIÇÃO DE DADOS DE PROCESSOS INDUSTRIAIS

Relatório de estágio integrado no Subprograma 4 do PRODEP, medida 4.3,
no Instituto de Sistemas e Robótica, Polo do Porto, de 1/7/92 a 30/9/92.

Executado por

Vera Lúcia Gomes Mata

sob a orientação de:

Professor Doutor **Jorge Martins de Carvalho**

Professor Doutor **Sebastião Feyo de Azevedo**

INSTITUTO DE SISTEMAS E ROBÓTICA
FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO
Outubro de 1992

AQUISIÇÃO DE DADOS DE PROCESSOS INDUSTRIAIS

Relatório de estágio integrado no Subprograma 4 do PRODEP, medida 4.3,
no Instituto de Sistemas e Robótica, Polo do Porto, de 1/7/92 a 30/9/92.

Executado por

Vera Lúcia Gomes Mata

sob a orientação de:

Professor Doutor **Jorge Martins de Carvalho**

Professor Doutor **Sebastião Feyo de Azevedo**

SUMÁRIO

A comunicação entre o computador e a interface foi estudado em pormenor e nesse sentido desenvolveram-se subrotinas com funções específicas de aquisição e envio de dados.

Esses programas foram escritos na linguagem Quick-Basic 4.50 e são facilmente integrados num programa principal, escrito nessa mesma linguagem, que faça o tratamento das variáveis envolvidas.

A descrição de cada subrotina encontra-se em no capítulo 'Aquisição e Envio de dados à escala laboratorial'.

Foram executadas rotinas para:

- aquisição de sinais analógicos;
- aquisição de sinais digitais;
- envio de sinais analógicos;
- envio de sinais digitais.

Cada um desses módulos foi tratado separadamente, embora eles possam ser conjugados e integrados em série num mesmo programa.

Para o referido procedimento é necessário a instalação de uma das seguintes placas, conforme descrito em cada subrotina:

- PCL-812PG com: 18 canais analógicos: 16 de entrada e 2 de saída,
 32 canais digitais: 16 de entrada e 16 de saída;
- PCL-726 com: 6 canais analógicos de saída,
 32 canais digitais: 16 de entrada e 16 de saída.

Ambas as placas estão preparadas para computadores IBM/XT/AT e compatíveis. Estão adaptadas a uma vasta gama de aplicações, tanto à escala laboratorial como industrial.

Cada placa possui uma série de elementos de contacto facilmente ajustáveis pelo utilizador, conforme se pode lêr na descrição pormenorizada que se encontra no capítulo seguinte, e que permitem um melhor ajuste das condições operatórias às condições físicas existentes.

A posição destes elementos, em caso de dúvida, pode ser verificada tanto por inspecção directamente na placa (o que poderá implicar ter de abrir o computador, caso a placa já esteja instalada) como pela utilização do programa de calibração - TESTPROG.EXE - que se encontra no conjunto de programas que acompanha a referida placa.

Depois de manipulação adequada é possível adaptá-las tanto a um SAD - Sistema de Aquisição de Dados -como a um SDA - Sistema de Envio de sinais- ou ainda num caso mais geral, integra-las conjuntamente num Controlador de Processos que manipule no máximo 16+8 canais analógicos (E+S) e 32+32 canais digitais (E+S).

Como exemplo de aplicação destas subrotinas, apresenta-se a descrição do programa PCLINOUT. Este programa tem capacidade de adquirir dados através de oito canais analógicos, no máximo, e de enviar sinais analógicos através de oito canais. Ambos os procedimentos são executados separadamente.

A aquisição e envio de dados é feita on-line. É um programa de fácil utilização e compreensão, que reflete no fundo a facilidade de manipulação destes tipos de interfaces.

INDICE

1.Aquisição e envio de dados à escala laboratorial	4
Descrição da interface PCL-812PG	4
Descrição da interface PCL-726	12
Subrotina HINPUT	19
Subrotina HINPUTNC	26
Subrotina HOUTPUT2	33
Subrotina HOUTPUT6	39
Subrotina DIGINOUT	43
2.Programa PCLINOUT	49

SECÇÃO	: Aquisição e Envio de dados à escala laboratorial
SUBSECÇÃO	: Descrição da Interface PCL-812PG, V1.0, 92.09.23
EQUIPAMENTO	: Interface PCL-812PG
AUTORES	: Vera Lúcia Gomes Mata* Sebastião Feyo de Azevedo**

1. Informação Geral

A Interface PCL-812 PG pode ser usada tanto para aquisição como para envio de dados, possuindo para o efeito dezoito canais analógicos: dezasseis de entrada e dois de saída e ainda dezasseis canais digitais de saída e dezasseis digitais de entrada.

É uma placa preparada para computadores IBM PC/XT/AT e compatíveis. Tem como características principais a sua multifuncionalidade e alta velocidade. Está adaptada a uma vasta gama de aplicações, tanto à escala laboratorial como industrial, onde se incluem a aquisição e envio de dados no controlo de processos e ensaios automáticos.

2. Especificações Gerais

Entradas Analógicas

Numero de Canais	16 canais
Resolução	12 bits
Gama de Entrada	+/- 5V, +/- 2.5V, +/- 1.25V, +/- 0.625V, +/- 0.3125V Gamas programáveis
Tipo de Conversão	Por aproximações sucessivas
Velocidade de Conversão	30 KHz no máximo
Linearidade	+/- 1 bit
Precisão	0.015%
Transferência de dados	Controlo programável, transferência interrompida e transferência por acesso directo à memória-DMA

* Aluna Finalista do Curso de Engenharia Química

** Professor Associado da FEUP

Saídas Analógicas

Numero de Canais	2 canais
Resolução	12 bits
Gamas de saída	0 a +5V ou 0 a +10V
Tensão de Referência	Interna: -5V (+/- 0.1V), -10V (+/- 0.2V) Externa:DC ou AC, máximo +/- 10V
Linearidade	+/- 0.5 bit

Entradas Digitais

Canal	16 bits
Nível	Compatível com TTL
Tensão de entrada	Menor: 0.8 V, no máximo Maior: 2.0 V, no mínimo
Carga de entrada	Menor: 0.4 mA, no máximo, a 0.5 V Maior: 0.05 mA, no máximo, a 2.7 V

Saídas Digitais

Canal	16 bits
Nível	Compatível com TTL
Tensão de saída	Menor: Debita 8 mA a 0.5 V, no máximo Maior: Fornece 0.4mA a 2.4 V, no mínimo

3. Instalação

Existem na placa diversos elementos de contacto chamados *Jumpers* (JP), cuja posição pode ser modificada sempre que se pretendam determinados requisitos.

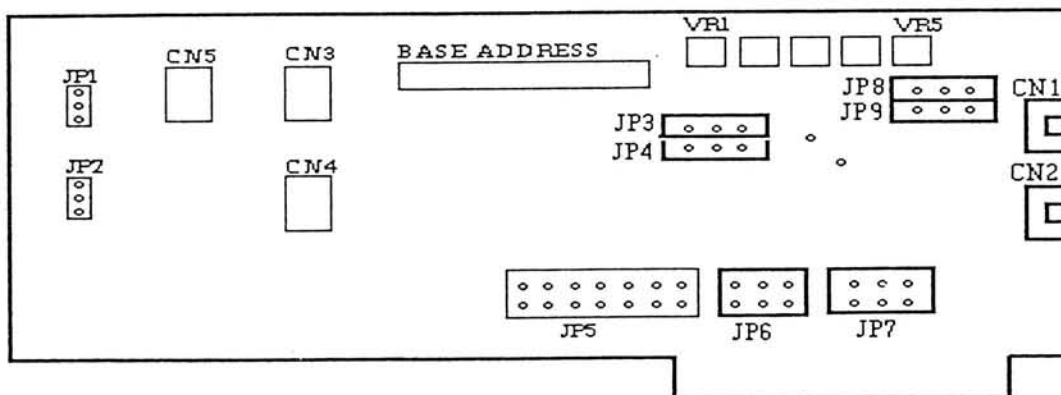


Fig. 1 - Diagrama esquemático da Interface PCL-812 PG

3.1. Elementos de contacto

Na placa PCL 812-PG existe um *dip-switch* e nove elementos de contacto. Vejamos como variam esses elementos:

3.1.1. Selecção do *Base Address*

Nome do Interruptor: SW1 (posições 1 a 5)

Muitos dispositivos periféricos e interfaces são controlados através das portas I/O. Estas portas são endereçadas usando o chamado *Base Address*.

O *Base Address* da porta I/O que estabelece contacto com a placa PCL 812-PG é seleccionável via um interruptor com 8 posições. Como esta placa requiere 16 endereços no espaço I/O, os endereços válidos situam-se na gama: HEX200 - HEX2FF. Pode-se, contudo, usar alguns destes endereços para outros dispositivos tais como COM1, COM2, impressora e Monitor.

O endereço estabelecido pela fábrica é de HEX220. No caso de ser necessário ajustar a um outro valor, a posição do interruptor está explicitada na tabela seguinte.

GAMA DE ENDEREÇOS E/S (HEX)	POSIÇÃO DO INTERRUPTOR					
	1 A8	2 A7	3 A6	4 A5	5 A4	6 A3
200-20F	0	0	0	0	0	X
210-21F	0	0	0	0	1	X
220-22F *	0	0	0	1	0	X
230-23F	0	0	0	1	1	X
⋮						
300-30F	1	0	0	0	0	X
⋮						
3F0-3FF	1	1	1	1	1	X

Nota: A9 é fixado em 1
ON = 0, OFF = 1
* - Especificação da fábrica
X - Não é usado

3.1.2. Selecção do estado de espera

Nome do interruptor: SW1 (posições 7 e 8)

Em alguns PC's de alta velocidade exista a necessidade de alterar o chamado *Estado de*

Espera para garantir estabilidade à transferência de dados.

O comprimento dos Estados de Espera podem ser seleccionados através das posições 7 e 8 no interruptor SW1. Esta placa pode ser configurada para um de quatro possíveis valores para a variável em questão (0,2,4 ou 6), conforme se encontra na tabela seguinte.

ESTADO DE ESPERA	POSICÃO DO INTERRUPTOR	
	7 A2	8 A1
0 *	0	0
2	0	1
4	1	0
6	1	1

* - Especificação da fábrica

3.1.3. Selecção da Fonte de Tensão de Referência dos Canais D/A

Nome dos Elementos de Contacto: JP3, JP4

A tensão de referência dos conversores D/A toma um dos valores -5 ou +5 volts gerada internamente, ou pode ainda ser uma voltagem vinda do exterior através do conector CN2, pino 17 ou 19. A fonte de tensão de referência do canal D/A 1(2) é seleccionada pelo JP3(4).

Ex: JP3 o o o :Referência do canal D/A 1 interna
 JP4 o o o :Referência do canal D/A 2 externa
 Int Ext

3.1.4. Selecção da referência interna D/A

Nome do elemento de contacto: JP8

NOTA: A referência interna é usada somente quando o(s) elemento(s) de contacto JP3 e/ou JP4 estiver(em) na posição INT.

A voltagem de referência interna pode ser de -5 ou -10 volts, seleccionável pelo elemento JP8.

Ex: JP8 10 5
 o o o :A voltagem de referência é -10 volts
 A gama de saída D/A é de 0 a +10volts

3.1.5. Selecção da voltagem máxima nas entradas A/D

Nome do elemento de contacto: JP9

A gama máxima de conversões A/D pode ser +/-5 volts ou +/-10 volts, seleccionável pelo JP9.

Se o JP9 é colocado na posição '5' as gamas analógicas de entrada possíveis são de +/-: 5, 2.5, 1.25, 0.625 e 0.3125 volts. Se o JP9 estiver na posição '10', as gamas respectivas são agora de +/-: 10, 5, 2.5, 1.25 e 0.625 volts.

A relação entre a gama analógica de entrada, a máxima tensão de entrada e o ganho do amplificador é dada pela seguinte equação:

$$\text{Gama de entrada} = \frac{\text{Tensão máxima (JP9)}}{\text{Ganho}}$$

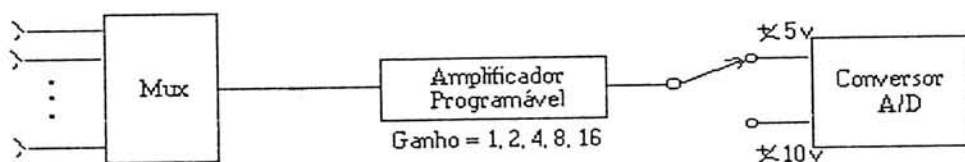


Fig.2 - Selecção da gama de entrada no conversor A/D.

Ex: JP9 10 5
 0 0 o :Gama máxima de entrada no conversor A/D é de +/- 10 volts.

NOTA: Esta gama necessita de um bom fornecimento de energia.

3.1.6. Selecção do canal DMA (JP6, JP7)

Nome do elemento de contacto: JP6, JP7

O modo DMA permite executar transferência de dados a alta velocidade na PCL-812 PG. Este tipo de transferência move os dados A/D do dispositivo de Hardware da interface directamente para a memória do PC,

sem necessidade de arranque da CPU.

As posições possíveis são:

	JP6	JP7
Nenhum DMA	x	x
DMA 1	1	1
DMA 3	3	3

3.1.7. Resistências Variáveis

Nome dos elementos: VRi

Existem cinco resistências variáveis (VR) na placa PCL-812 PG que permitem um melhor ajuste dos valores transferidos aos valores reais.

A posição dessas resistências é indicada na figura 1.

As funções de cada uma são:

- VR1 - Ajuste do ganho do canal D/A #1
- VR2 - Ajuste do ganho do canal D/A #2
- VR3 - Ajuste do ganho dos canais A/D
- VR4 - Ajuste do desvio (offset) do amplificador programável
- VR5 - Ajuste do desvio (offset).

3.2. Conectores

A PCL-812 PG está equipada com 3 conectores de 20 pinos acessíveis no fundo da placa e 2 outros conectores também de 20 pinos acessíveis nos bordos desta.

Apresentam-se em seguida os diagramas representativos do alinhamento dos pinos referentes aos conectores CN1 a CN4. São estes dois módulos que possibilitam a transferência de dados de e para o exterior.

A/D	1	2	A.GND
A/D	3	4	A.GND
A/D	5	6	A.GND
A/D	7	8	A.GND
A/D	9	10	A.GND
A/D	11	12	A.GND
A/D	13	14	A.GND
A/D	15	16	A.GND
A/D	17	18	A.GND
A/D	19	20	A.GND

CN1: 10 Entradas Analógicas

A/D	1	2	A.GND
A/D	3	4	A.GND
A/D	5	6	A.GND
A/D	7	8	A.GND
A/D	9	10	A.GND
A/D	11	12	A.GND
D/A	13	14	A.GND
D/A	15	16	A.GND
V.REF	17	18	A.GND
V.REF	19	20	A.GND

CN2: 6 Entradas Analógicas
2 Saídas Analógicas

D/I	1	2	D/I
D/I	3	4	D/I
D/I	5	6	D/I
D/I	7	8	D/I
D/I	9	10	D/I
D/I	11	12	D/I
D/I	13	14	D/I
D/I	15	16	D/I
D.GND	17	18	D.GND
+5V	19	20	+12V

CN3: 16 Entradas Digitais

D/O	1	2	D/O
D/O	3	4	D/O
D/O	5	6	D/O
D/O	7	8	D/O
D/O	9	10	D/O
D/O	11	12	D/O
D/O	13	14	D/O
D/O	15	16	D/O
D.GND	17	18	D.GND
+5V	19	20	+12V

CN4: 16 Saídas Digitais

4. Programação

Existem duas possibilidades de incorporar as funções da PCL-812PG num programa de aplicação:

Um dos métodos é lêr e escrever as instruções da porta I/O, directamente no programa de aplicação.

O outro modo, e de certa forma mais simples, é usando as rotinas existentes no Software da interface, as chamadas DRIVER ROUTINES. Estas estão preparadas para executar todas as funções de manuseamento da porta I/O da interface.

5. Referências

- [1] -"PCL-812PG ENHANCED MULTI-LAB CARD-USER'S MANUAL".
Maio, 1990.

SECÇÃO	: Aquisição e Envio de dados à escala laboratorial
SUBSECÇÃO	: Descrição da Interface PCL-726, V1.0, 1992.09.23
EQUIPAMENTO	: Interface PCL-726
AUTORES	: Vera Lúcia Gomes Mata* Sebastião Feyo de Azevedo**

1. Informação Geral

A Interface PCL-726 é exclusivamente usada para aquisição e envio de dados, possuindo para o efeito seis canais analógicos de saída, 16 canais digitais de entrada e 16 digitais de saída.

Cada canal analógico pode ser estabelecido individualmente com uma das seguintes gamas de saída: 0 a 5V, 0 a 10V, +/- 5V, +/- 10V ou 4 a 20 mA.

Projectada para o envio de dados em ambientes acidentados, a PCL-726 é uma solução ideal e económica para aplicações industriais que requerem múltiplos canais analógicos de saída de tensão e/ou corrente.

2. Especificações Gerais

Saídas Analógicas

Numero de Canais	6 canais
Resolução	12 bits
Gama de Saída	0 a +5V (Unipolar) 0 a +10V (Unipolar) +/- 5V (Bipolar) +/- 10V (Bipolar) 4 a 20mA (Corrente)
Tensão de Referência	
-Interna:	-5V (+/- 0.05V) -10V (+/- 0.05V)
-Externa:	DC ou AC, máximo +/- 10V
Linearidade	+/- 0.5 bit

* Aluna Finalista do Curso de Engenharia Química

** Professor Associado da FEUP

Precisão	+/- 0.012% da gama total
Estado Inicial	Todos os canais estão a 0V, tanto no modo Unipolar como no modo Bipolar

Entradas Digitais

Canal	16 bits
Nível	Compatível com TTL
Tensão de entrada	Menor: 0.8 V, no máximo Maior: 2.0 V, no mínimo
Carga de entrada	Menor: -0.4 mA, no máximo, a 0.5 V Maior: 0.05 mA, no máximo, a 2.7 V

Saídas Digitais

Canal	16 bits
Nível	Compatível com TTL
Tensão de saída	Menor: Debita 8 mA a 0.5 V, no máximo Maior: Fornece 0.4mA a 2.4 V, no mínimo

3. Instalação

A placa da interface PCL-726 possui um conjunto de elementos de contacto e interruptores facilmente ajustados pelo utilizador. Estes, uma vez devidamente seleccionados, permitirão que haja uma melhor correspondência entre as gamas de valores de saída digitais e as gamas analógicas realmente desejadas.

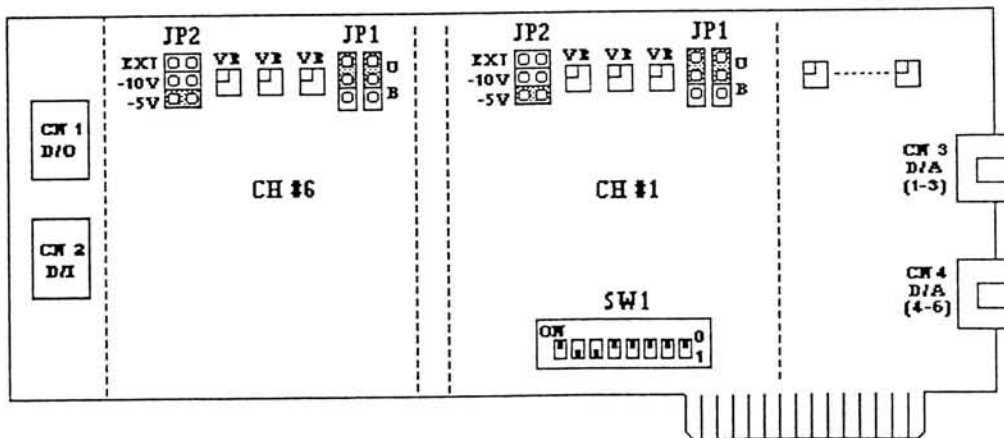


Figura 1. Diagrama esquemático da placa PCL 726.

A função de cada um dos elementos e a forma como eles se podem alterar são explicadas em seguida.

3.1. Elementos de Contacto (Jumpers, JP)

Na placa PCL 726 existe um dip-switch e sete elementos de contacto, os chamados Jumpers.

3.1.1. Selecção do Base Address

Nome do interruptor: SW1 (posições 1 a 5)

Muitos dispositivos periféricos e interfaces são controlados através das portas I/O. Estas portas são endereçadas usando o chamado *Base Address*.

O Base Address da porta I/O que estabelece contacto com a placa PCL 726 é seleccionável via um interruptor com 8 posições. Como esta placa requiere 16 endereços no espaço I/O, os endereços válidos situam-se na gama: HEX200 - HEX2F0. Pode-se, contudo, usar alguns destes endereços para outros dispositivos tais como COM1, COM2, impressora e Monitor.

O endereço estabelecido pela fábrica é de HEX2C0. No caso de ser necessário ajustar a um outro valor, a posição do interruptor está explicitada na tabela seguinte.

GAMA DE ENDEREÇOS E/S (HEX)	POSIÇÃO DO INTERRUPTOR					
	1 A8	2 A7	3 A6	4 A5	5 A4	6 A3
200-20F	0	0	0	0	0	X
210-21F	0	0	0	0	1	X
220-22F *	0	0	0	1	0	X
230-23F	0	0	0	1	1	X
⋮						
300-30F	1	0	0	0	0	X
⋮						
3F0-3FF	1	1	1	1	1	X

Nota: A9 é fixado em 1

ON = 0, OFF = 1

* - Especificação da fábrica

X - Não é usado

3.1.2. Selecção do estado de espera

Nome do interruptor: SW1 (posições 7 e 8)

Alguns PC's de alta velocidade podem requerir que os chamados *Estados de Espera* sejam inseridos de modo a efectuar-se uma transferência de dados suficientemente estável.

O comprimento dos Estados de Espera podem ser seleccionados através das posições 7 e 8 no interruptor SW1. Esta placa pode ser configurada para um de quatro possíveis valores para a variável em questão (0,2,4 ou 6), conforme se encontra na tabela seguinte.

ESTADO DE ESPERA	POSIÇÃO DO INTERRUPTOR	
	7 A2	8 A1
0 *	0	0
2	0	1
4	1	0
6	1	1

* - Especificação da fábrica

3.1.3. Selecção da Tensão de Referência e Modo

Nome dos Elementos de Contacto: JP1, JP2 (para cada um dos canais)

Existem dois *Jumpers* para selecção da fonte e valor da tensão de referência e modo de saída, em cada canal D/A.

A posição do elemento JP1 permite seleccionar uma referência interna de -5 ou -10 volts possibilitando ainda a utilização de uma tensão de referência externa.

Algumas combinações típicas são tabeladas em seguida.

Gama de Saída	JP2	JP1
0 a +5V Unipolar		
0 a +10V Unipolar		
+/- 5V Bipolar		
+/- 10V Bipolar		
4 a 20mA		
Ref.Ext (Atenuador programável)		

3.2. Conectores

A PCL 726 está equipada com quatro conectores de vinte pinos cada. Dois destes permitem efectuar conversões digital-analógicas - CN3 e CN4 - e os dois restantes são utilizados para saídas e entradas digitais - CN1 e CN2.

Os dois diagramas seguintes ilustram a disposição dos pinos em cada um dos conectores laterais que possibilitam a saída de sinais digitais para o exterior.

D/I	1	2	D/I
D/I	3	4	D/I
D/I	5	6	D/I
D/I	7	8	D/I
D/I	9	10	D/I
D/I	11	12	D/I
D/I	13	14	D/I
D/I	15	16	D/I
D.GND	17	18	D.GND
+ 5V	19	20	+12V

CN1: 16 Entradas Digitais

D/O	1	2	D/O
D/O	3	4	D/O
D/O	5	6	D/O
D/O	7	8	D/O
D/O	9	10	D/O
D/O	11	12	D/O
D/O	13	14	D/O
D/O	15	16	D/O
D.GND	17	18	D.GND
+ 5V	19	20	+12V

CN2: 16 Saídas Digitais

-5 Ref.Out	1	2	-10V Ref.Out
-5 Ref.Out	3	4	-10V Ref.Out
D/A #1 V.Out	5	6	D/A #1 I.Out
D/A #1 Ref.In	7	8	A.Gnd
D/A #2 V.Out	9	10	D/A #2 I.Out
D/A #2 Ref.In	11	12	A.Gnd
D/A #3 V.Out	13	14	D/A #3 I.Out
D/A #3 Ref.In	15	16	A.Gnd
A.Gnd	17	18	A.Gnd
Não ligado	19	20	Não ligado

CN3: Saídas D/A (Ch#1 a Ch#3)

-5 Ref.Out	1	2	D/A #4 I.Out
-5 Ref.Out	3	4	A.Gnd
D/A #4 V.Out	5	6	D/A #5 I.Out
D/A #4 Ref.In	7	8	A.Gnd
D/A #5 V.Out	9	10	D/A #6 I.Out
D/A #5 Ref.In	11	12	A.Gnd
D/A #6 V.Out	13	14	A.Gnd
D/A #6 Ref.In	15	16	A.Gnd
A.Gnd	17	18	+5V
Não ligado	19	20	+12V

CN4: Saídas D/A (Ch#4 a Ch#6)

4. Programação

As funções D/A da PCL 726 podem ser executadas bastando para tal a escrita directa de instruções I/O para os registos D/A.

Aquando da programação dos canais D/A é necessário ter em atenção que o byte mais significativo é enviado primeiro. Depois do byte mais baixo ser enviado, os dois valores são adicionados e passam para o conversor D/A ao mesmo tempo.

O seguinte programa escrito em Quick Basic 4.5 exemplifica como programar o canal D/A #1 para gerar uma voltagem de +3 volts.

' JP1: Unipolar, JP2: -5 volts.

' Gama de saída D/A é então de 0 a +5 volts.

,

V = 3

x.value% = (V/5*4096) ' Para Unipolar

' x.value% = ((V/5*2048) + 2048 ' Para Bipolar (-5 a +5 volts)

IF x.value% > 4095 OR x.value% < 0 then PRINT " Valor incorrecto !!!": END

```
hi.byte% = x.value% \ 256
lo.byte% = x.value% MOD 256
OUT &H2C0 + 0, hi.byte%
OUT &H2C0 + 1, lo.byte%
END
```

5. Referências

[1] - "PCL 726 USER'S MANUAL". Maio 1989.

SECÇÃO	: Aquisição e Envio de dados à escala laboratorial
SUBSECÇÃO	: Subrotina HINPUT - Aquisição de dados através de canais consecutivos, V1.0, 1992.07.23
EQUIPAMENTO	: Interface PCL-812PG
SISTEMA OPERATIVO	: MS-DOS, LINGUAGEM : Quick-Basic 4.5
AUTORES	: Vera Lúcia Gomes Mata* Sebastião Feyo de Azevedo**

1. Objectivo

Aquisição de dados à escala laboratorial através da utilização da interface PCL 812PG.

2. Especificações (Quick-Basic 4.5)

```

SUB HINPUT (ncanais, pcanal, ppin(), er%, amp())
  'Integer ncanais, pcanal, er
  'Double precision ppin()
  ' COMMON SHARED Ind  'Interface só é inicializada uma vez

```

3. Descrição

Esta subrotina tem a possibilidade de adquirir dados simultâneamente através de um numero máximo de dezasseis canais.

Utiliza para o efeito a interface PCL 812PG ENHANCED MULTI-LAB CARD desenvolvida pela Advantech Co., Ltd, para computadores IBM PC/XT/AT e compatíveis.

Trata-se de uma interface de 12 bits com dezasseis canais de entrada e com uma

* Aluna finalista do curso de Engenharia Química da FEUP

** Professor Associado da FEUP

amplitude de entrada programável. A placa recebe o valor de cada propriedade em volts. O computador converte para valor digital na gama 0 a 2^{12} bits e por fim é calculado o valor percentual correspondente à gama analógica utilizada.

Para mais informação sobre a sua instalação, utilização e programação aconselha-se a consulta da subsecção referente à descrição da interface.

4. Parâmetros

Parâmetros de Entrada

ncanais - Variável INTEIRA, numero total de canais consecutivos ligados [1-16]

pcanal - Variável INTEIRA, primeiro canal ligado[0-15]

amp() - Variável inteira, Código correspondente à amplitude de entrada do canal; dimensão ncanais. Os códigos permitidos (e as correspondentes gamas fisicas) são:

Gama(volts): ±	0.625	1.25	2.5	5	10	(JP9=-10)
Gama(volts): +	0.312	0.625	1.25	2.5	5	(JP9= -5)
Código(=amp(i))	4	3	2	1	0	

Parâmetros de Saída

ppin() - Variável indexada, em PRECISÃO DUPLA,; dimensão ncanais. Serve como variável de saída do valor da propriedade lida, em percentagem.

er% - Variável INTEIRA, indicador de erro. Os valores possíveis encontram-se na alinea 7 - Indicadores.

5. Rotinas Auxiliares

A comunicação com a interface estabelece-se utilizando as chamadas *Driver Routines*. Neste trabalho chamaram-se as rotinas de: Inicialização, Estabelecimento da gama de canais, Estabelecimento da amplitude de entrada e Conversão Analógico-Digital.

6. Rotinas a definir pelo utilizador

O utilizador não necessita de definir qualquer rotina.

7. Indicadores

A variável indicadora de erro, *er%*, assume os seguintes valores à saída da subrotina HINPUT:

- er%* = 0 : Execução bem sucedida
- er%* = 1 : Falha na inicialização
- er%* = 3 : Gama do endereço da porta E/S errada
- er%* = 7 : Gama de canais de entrada A/D violada
- er%* = 9 : Falha no equipamento de conversão A/D. Dados não disponíveis.

8. Exemplo

Pretende-se lêr as temperaturas de 4 aquecedores. A gama de temperaturas possível é de 100 - 1000°C e o sinal eléctrico do sistema de detecção-transmissão é, após amplificação, de ± 5 volts.

O objectivo é lêr temperaturas e detectar situações em que os valores estejam a menos de 20% dos limites físicos recomendados (isto é, entre 20 e 80% da gama de funcionamento). Utilizam-se os canais 0, 1, 2 e 3.

```
DECLARE SUB HINPUT (ncanais!, pcanal!,ppin(),er%,amp())
COMMON SHARED Ind
DIM c(4), ppin(4), amp(4)
```

```
OPEN dados.dat FOR INPUT AS #1
```

```

INPUT #1, lixo$
INPUT #1, ncanais
INPUT #1, lixo$
INPUT #1, pcanal
INPUT #1, lixo$
FOR i=pcanal TO pcanal+ncanais-1
INPUT #1, amp(i)
INPUT #1, ppmin(i)
INPUT #1, ppmax(i)
NEXT i

```

inicio:

```

CALL HINPUT (ncanais, pcanal, ppin(), er%, amp())
IF er% <> 0 THEN
LOCATE 10,10
PRINT USING"ER%=##";er%
END IF

FOR i=1 TO ncanais
pp=pcanal+i-1
y(i)=ppmin(i)+ppin(i)*(ppmax(i)-ppmin(i))/100
LOCATE 10+i, 10
IF ppin(i) < 20 OR ppin(i) > 80 then
BEEP
LOCATE 20,10
PRINT"CUIDADO.A propriedade está fora dos limites admissíveis"
END IF

PRINT USING"ppin(##)=###.##%; valor = ##.##";pp;ppin(i);y(i)

NEXT i
GOTO inicio
END

```

O ficheiro **dados.dat** é, neste caso:

```

ncanais
4
pcanal
0
amp(i), ppmin(i),ppmax(i)
0, 100, 1000
0, 100, 1000

```

0, 100, 1000
0, 100, 1000

9. Palavras - Chave

Sistema de Aquisição e Envio de Dados
Interface
Conversão Analógico-Digital

10. Referências

- [1] - "Pcl - 812 ENHANCED MULTI-LAB CARD - USER'S MANUAL".
Maio de 1990.
- [2] - "Process Dynamics and control"; D. Seborg, T. Edgar, D. Mellichamp;
John Wiley & Sons; 1989

===== HINPUT =====

Subrotina utilizada para aquisição de NCANAIS consecutivos na PCL812PG

SUB HINPUT (ncanais!, pcanal!, ppin(), ER%, amp())

Os parâmetros são:

ncanais : Numero Total de Canais Consecutivos (1-16)
pcanal : Primeiro Canal (0-15)
ppin() : Valor Percentual da Propriedade Lida
ER : Indicador de Erro
amp() : Código Correspondente à Amplitude do Canal

Para o funcionamento correcto desta Subrotina ,
necessário colocar os pinos da placa PCL 812-PG
nas seguintes posições:

BASE ADDRESS - HEX 220
JP1 - INTERNAL
JP5 - X (Nulo)
JP6, JP7 - X (Nulo)

DIM dat%(20), ary1%(20), ary2%(20)

ind = ind + 1

IF ind > 1 THEN GOTO CAD

----- Inicialização da placa -----

port% = &H220: '	Porta-Adereço I/O
dat%(0) = port%	
dat%(1) = 3: '	Seleccção de IRQ 3
dat%(2) = 1: '	Seleccção de DMA 1
ER% = 0: '	Indicador de Erro
fun% = 0: '	Função 0

CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), ER%)

IF ER% <> 0 THEN
EXIT SUB

END IF

'----- Estabelecimento dos Canais de Entrada -----

dat%(0) = pcanal: ' Primeiro Canal
dat%(1) = pcanal + ncanais - 1: ' Ultimo Canal
fun% = 1: ' Função 1

CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), ER%)

IF ER% <> 0 THEN
EXIT SUB
END IF

'----- Estabelecimento das Gamas de Entrada dos Canais -----

FOR i = pcanal TO ncanais - 1
dat%(0) = i: ' Canal i
pp = i - pcanal + 1
dat%(1) = amp(i): ' Código da Amplitude do Canal i
fun% = 23: ' Função 23

CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), ER%)

IF ER% <> 0 THEN
EXIT SUB
END IF

NEXT i

CAD:

'----- Conversão Analógico-Digital -----

fun% = 3: ' Função 3
FOR j = 1 TO ncanais

CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), ER%)

IF ER% <> 0 THEN
EXIT SUB
END IF

ppin(dat%(1)) = dat%(0)
ppin(dat%(1)) = ppin(dat%(1)) / 4095 * 100 ' 0 % < ppin(j) < 100 %

NEXT j

'-----

END SUB

SECÇÃO	: Aquisição e Envio de dados à escala laboratorial
SUBSECÇÃO	: Subrotina HINPUTNC - Aquisição de dados através de canais não consecutivos, V1.0, 1992.07.30
EQUIPAMENTO	: Interface PCL-812PG
SISTEMA OPERATIVO	: MS-DOS, LINGUAGEM : Quick-Basic 4.5
AUTORES	: Vera Lúcia Gomes Mata* Sebastião Feyo de Azevedo**

1. Objectivo

Aquisição de dados à escala laboratorial através da utilização da interface PCL 812PG.

2. Especificações (Quick-Basic 4.5)

```
SUB HINPUTNC (ncanais, c(), ppin(), er%, amp())
'Integer ncanais, er, c()
'Double precision ppin()
'COMMON SHARED Ind 'Placa inicializada uma só vez
```

3. Descrição

Esta subrotina tem a possibilidade de adquirir dados simultaneamente através de um numero máximo de dezasseis canais, não necessariamente consecutivos.

Utiliza para o efeito a interface PCL 812PG ENHANCED MULTI-LAB CARD desenvolvida pela Advantech Co., Ltd, para computadores IBM PC/XT/AT e compatíveis.

Trata-se de uma interface de 12 bits com dezasseis canais de entrada e com uma amplitude de entrada programável. A placa recebe o valor de

* Aluna Finalista do Curso de Engenharia Química

** Professor Associado da FEUP

cada propriedade em volts. O computador converte para valor digital na gama 0 a 2^{12} bits e por fim é calculado o valor percentual correspondente à gama analógica utilizada.

Para mais informação sobre a sua instalação, utilização e programação aconselha-se a consulta da subsecção referente à descrição da interface.

4. Parâmetros

Parâmetros de Entrada

ncanais - Variável INTEIRA, numero total de canais ligados[1-16]

c() - Variável INTEIRA, de dimensão *ncanais*, vector dos canais pretendidos para aquisição, por ordem crescente destes [0-15]

amp() - Variável INTEIRA, código correspondente à amplitude de entrada do canal, de dimensão *ncanais*. Os códigos permitidos (e as correspondentes gamas físicas) são:

Gama(volts): ±	0.625	1.25	2.5	5	10	(JP9=-10)
Gama(volts): +	0.312	0.625	1.25	2.5	5	(JP9= -5)
Código(=amp(i))	4	3	2	1	0	

Parâmetros de Saída

ppin() - Variável indexada, em PRECISÃO DUPLA, dimensão *ncanais*. Serve como variável de saída do valor da propriedade lida, em percentagem.

er% - Variável INTEIRA, indicador de erro. Os valores possíveis encontram-se na alinea 7 - Indicadores.

5. Rotinas Auxiliares

A comunicação com a interface estabelece-se utilizando as chamadas *Driver Routines*. Neste trabalho chamaram-se as rotinas de: Inicialização, Estabelecimento da gama de canais, Estabelecimento da amplitude de entrada e Conversão Analógico-Digital.

6. Rotinas a definir pelo utilizador

O utilizador não necessita de definir qualquer rotina.

7. Indicadores

A variável indicadora de erro, *er%*, assume os seguintes valores à saída da subrotina HINPUT:

- er%* = 0 : Execução bem sucedida
- er%* = 1 : Falha na inicialização
- er%* = 3 : Gama do endereço da porta E/S errada
- er%* = 7 : Gama de canais de entrada A/D violada
- er%* = 9 : Falha no equipamento de conversão A/D. Dados não disponíveis.

8. Exemplo

Pretende-se lêr as temperaturas de 4 aquecedores. A gama de temperaturas possível é de 100 - 1000°C e o sinal eléctrico do sistema de detecção-transmissão é, após amplificação, de ± 5 volts.

O objectivo é lêr temperaturas e detectar situações em que os valores estejam a menos de 20% dos limites físicos recomendados (isto é, entre 20 e 80% da gama de funcionamento). Os canais utilizados são: 0, 4, 7 e 15.

```
DECLARE SUB HINPUT (ncanais!, pcanal!,ppin(),er%,amp())
COMMON SHARED Ind
DIM ppin(4), amp(4), c(4)
```

```

OPEN dados2.dat FOR INPUT AS #1
  INPUT #1, lixo$
  INPUT #1, ncanais
  INPUT #1, lixo$
  FOR i=1 TO ncanais
    INPUT #1, c(i), amp(c(i)) ppmin(c(i)), ppmax(c(i))
  NEXT i

inicio:

CALL HINPUT (ncanais, c(), ppin(), er%, amp())
  IF er% <> 0 THEN
    LOCATE 10,10
    PRINT USING"ER%=##";er%: END
  END IF

FOR i=1 TO ncanais
  y(i)=ppmin(c(i))+ppin(c(i))*(ppmax(c(i))-ppmin(c(i)))/100
  LOCATE 10+i, 10
  IF ppin(c(i)) < 20 OR ppin(i) > 80 then
    BEEP
    LOCATE 20,10
    PRINT"CUIDADO.A propriedade está fora dos limites admissíveis"
  END IF

PRINT USING"ppin(##)=###.##%; valor = ##.##";c(i);ppin(c(i));y(i)

NEXT i
GOTO inicio
END

```

O ficheiro de dados **dados2.dat** é:

```

ncanais
4
c(i), amp(c(i)), ppmin(c(i)), ppmax(c(i))
0, 0, 100, 1000
4, 0, 100, 1000
7, 0, 100, 1000
15, 0, 100, 1000

```

9. Palavras - Chave

Sistema de Aquisição e Envio de Dados
Interface
Conversão Analógico-Digital

10. Referências

- [1] - "Pcl - 812 ENHANCED MULTI-LAB CARD - USER'S MANUAL". Maio de 1990.
- [2] - "Process Dynamics and control"; D. Seborg, T. Edgar, D. Mellichamp; John Wiley & Sons; 1989

===== HINPUTNC - Canais Não Consecutivos =====

Subrotina utilizada para aquisição de NCANAIS não consecutivos.
Os parâmetros são:

ncanais : Numero Total de Canais Não Consecutivos (1-16)
c() : Numero do Canal (0-15)
ppin() : Valor Percentual da Propriedade Lida
ER : Indicador de Erro
amp() : Código Correspondente à Amplitude do Canal

Para o funcionamento correcto desta Subrotina ,
necessário colocar os pinos da placa PCL 812-PG
nas seguintes posições:

BASE ADDRESS - HEX 220
JP1 - INTERNAL
JP5 - X (Nulo)
JP6, JP7 - X (Nulo)

SUB HINPUTNC (n, c(), ppin(), er%, amp())

REDIM dat%(20), ary1%(20), ary2%(20)

ind = ind + 1

IF ind > 1 THEN GOTO CAD

----- Inicialização da placa -----

port% = &H220: ' Porta-Adereço I/O
dat%(0) = port%
dat%(1) = 3: ' Selecção de IRQ 3
dat%(2) = 1: ' Selecção de DMA 1
er% = 0: ' Indicador de Erro
fun% = 0: ' Função 0

CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)

IF er% <> 0 THEN
EXIT SUB

END IF

----- Estabelecimento dos Canais de Entrada -----

dat%(0) = c(1): ' Primeiro Canal
dat%(1) = c(n): ' Ultimo Canal
fun% = 1: ' Função 1

```
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
IF er% <> 0 THEN  
    EXIT SUB  
END IF
```

```
'----- Estabelecimento das Gamas de Entrada dos Canais -----'
```

```
FOR i = 1 TO n  
    dat%(0) = c(i): '          Canal i  
    pp = c(i)  
    dat%(1) = amp(pp): '      Código da Amplitude do Canal i  
    fun% = 23: '              Função 23
```

```
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
IF er% <> 0 THEN  
    EXIT SUB  
END IF
```

```
NEXT i
```

```
CAD:
```

```
'----- Conversão Analógico-Digital -----'
```

```
dat%(1) = 0  
fun% = 3: '          Função 3  
FOR j = 1 TO n
```

```
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
IF er% <> 0 THEN  
    EXIT SUB  
END IF
```

```
ppin(dat%(1)) = dat%(0)  
ppin(dat%(1)) = ppin(dat%(1)) / 4095 * 100 ' 0 % < ppin(j) < 100 %
```

```
NEXT j
```

```
'-----'
```

```
END SUB
```

SECÇÃO	: Aquisição e Envio de dados à escala laboratorial.
SUBSECÇÃO	: Subrotina HOUTPUT2 - Envio de sinais analógicos através de dois canais consecutivos. V1.0, 1992.08.02
EQUIPAMENTO	: Interface PCL-812PG
SISTEMA OPERATIVO	: MS-DOS, LINGUAGEM : Quick-Basic 4.5
AUTORES	: Vera Lúcia Gomes Mata* Sebastião Feyo de Azevedo**

1. Objectivo

Envio de sinais analógicos à escala laboratorial através da utilização da interface PCL 812PG.

2. Especificações (Quick-Basic 4.5)

```
SUB HOUTPUT2 ( ppout(), er%)
  'Integer er
  'Double precision ppout()
```

3. Descrição

Esta subrotina tem a possibilidade de enviar sinais analógicos através de dois canais.

Utiliza para o efeito a interface PCL 812PG ENHANCED MULTI-LAB CARD desenvolvida pela Advantech Co., Ltd, para computadores IBM PC/XT/AT e compatíveis.

Trata-se de uma interface de 12 bits com dois canais de saída e com uma

* Aluna Finalista do Curso de Engenharia Química

** Professor Associado da FEUP

amplitude de saída de 0 a +5 volts ou de 0 a +10 volts (dependendo da posição do JP8.

Para mais informação sobre a sua instalação, utilização e programação aconselha-se a consulta da subsecção referente à descrição da interface.

4. Parâmetros

Parâmetros de Entrada

ppout() - Variável indexada, em PRECISÃO DUPLA, de dimensão 2. Serve como variável de entrada do valor analógico que se deseja gerar, em percentagem.

Parâmetros de Saída

er% - Variável INTEIRA, indicador de erro. Os valores possíveis encontram-se na alinea 7 - Indicadores.

5. Rotinas Auxiliares

A comunicação com a interface estabelece-se utilizando as chamadas *Driver Routines*. Neste trabalho chamaram-se as rotinas de: Inicialização e Conversão Digital-Analógica com saída para dois canais.

O modo de aceder a essas rotinas encontra-se descrita no manual do utilizador.

6. Rotinas a definir pelo utilizador

O utilizador não necessita de definir qualquer rotina.

7. Indicadores

A variável indicadora de erro, `er%`, assume os seguintes valores à saída da subrotina HINPUT:

```
er% = 0 : Execução bem sucedida
er% = 1 : Falha na inicialização
er% = 14 : Dados D/A inválidos (não pertence a 0-4095)
er% = 15 : Numero de canal D/A inválido (diferente de 1 ou 2)
```

8. Exemplo

O programa exemplo seguinte envia sinais aleatórios para os canais D/A #1 e #2, de 5 em 5 segundos.

```
DECLARE SUB HOUTPUT (ppout(),er%)
DIM ppout(2)

inicio:

t0 = TIMER
DO
LOOP WHILE (TIMER-t0) < 5

    ppout(1) = RND * 100           'Valores na gama 0-100%
    ppout(2) = RND * 50 + 25      'Valores na gama 25-75%

CALL HOUTPUT ( ppout(), er%)
    IF er% <> 0 THEN
        LOCATE 10,10
        PRINT USING "ER%=##";er%
    END IF

LOCATE 10,10: PRINT USING "CANAL D/A #1 =>###.##%";ppout(1)
LOCATE 12,10: PRINT USING "CANAL D/A #2 0>###.##%";ppout(2)

GOTO inicio
END
```


9. Palavras - Chave

Sistema de Aquisição e Envio de Dados
Interface
Conversão Digital-Analógica

10. Referências

- [1] - "Pcl - 812 ENHANCED MULTI-LAB CARD - USER'S MANUAL".
Maio de 1990.
- [2] - "Process Dynamics and control"; D. Seborg, T. Edgar,
D. Mellichamp; John Wiley & Sons; 1989

===== HOUTPUT2 =====

Subrotina HOUTPUT2: Envio de dados pelos 2 canais da PCL-812PG

Para o funcionamento correcto desta Subrotina é necessário colocar os pinos da placa PCL 812-PG nas seguintes posições:

BASE ADDRESS - HEX 220
JP8, JP9 - INTERNAL

SUB HOUTPUT2 (ppout(), er%)

REDIM ary1%(20), ary2%(20)

REDIM dat%(20)

port% = &H220

dat%(0) = port%

dat%(1) = 3

dat%(2) = 1

er% = 0

fun% = 0

CALL pci812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)

IF er% <> 0 THEN
EXIT SUB

END IF

dat%(0) = INT(ppout(1) * 4095 / 100)

dat%(1) = INT(ppout(2) * 4095 / 100)

fun% = 16

CALL pci812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)

IF er% <> 0 THEN
EXIT SUB

END IF

END SUB

SECÇÃO : Aquisição e Envio de dados à escala laboratorial.

SUBSECÇÃO : Subrotina HOUTPUT6 - Envio de sinais analógicos
através de seis canais consecutivos.
V1.0, 1992.08.15

EQUIPAMENTO : Interface PCL-726

SISTEMA OPERATIVO : MS-DOS, LINGUAGEM : Quick-Basic 4.5

AUTORES : Vera Lúcia Gomes Mata*
Sebastião Feyo de Azevedo**

1. Objectivo

Envio de sinais analógicos à escala laboratorial através da utilização da interface PCL 726.

2. Especificações (Quick-Basic 4.5)

```
SUB HOUTPUT6 ( ppout()  
'Double precision ppout()
```

3. Descrição

Esta subrotina tem a possibilidade de enviar sinais analógicos através de seis canais.

Utiliza para o efeito a interface PCL 726 ENHANCED MULTI-LAB CARD desenvolvida pela Advantech Co., Ltd, para computadores IBM PC/XT/AT e compatíveis.

Trata-se de uma interface de 12 bits com seis canais de saída e com uma

* Aluna finalista do curso de Engenharia Química da FEUP

** Professor Associado da FEUP

amplitude de saída variável (dependendo da posição dos elementos de contacto da placa: JP1 e JP2, para cada canal).

Para mais informação sobre a sua instalação, utilização e programação aconselha-se a consulta da subsecção referente à descrição da interface.

4. Parâmetros

Parâmetros de Entrada

ppout() - Variável indexada, em PRECISÃO DUPLA, de dimensão 6. Serve como variável de entrada do valor analógico que se deseja gerar, em percentagem.

5. Rotinas Auxiliares

A comunicação com a interface estabelece-se sem necessidade de utilização de qualquer rotina auxiliar.

6. Rotinas a definir pelo utilizador

O utilizador não necessita de definir qualquer rotina.

7. Exemplo

O programa exemplo seguinte envia sinais analógicos para os canais D/A #1 a #6 consoante o pretendido pelo utilizador.

É um programa simples que permite saber a posição dos elementos de contacto JP1 e JP2 (vêr subsecção correspondente à descrição desta placa).

```
DECLARE SUB HOUTPUT (ppout(),er%)
```

```
DIM ppout(6)
```

```
inicio:
```

```
FOR i=1 TO 6
```

```
    PRINT USING "ppout(#) [0-100%, -1 para terminar=";i;
```

```
    INPUTppout(i)
```

```
    IF ppout(i)=-1 then END
```

```
NEXT i
```

```
CALL HOUTPUT6 ( ppout())
```

```
GOTO inicio
```

```
END
```

Por exemplo, se para o canal #1 os elementos de contacto estiverem na posição JP1 = B e JP2 = -10, o valor de tensão medido, inpondo ao programa o seguinte valor de sinal de saída, é:

```
ppout(1) = 0      tensão = -10V  
ppout(1) = 100   tensão = +10V
```

9. Palavras - Chave

Sistema de Aquisição e Envio de Dados
Interface
Conversão Analógico-Digital

10. Referências

[1] - "Pcl - 726 - USER'S MANUAL".
Maio de 1990.

[2] - "Process Dynamics and control"; D. Seborg, T. Edgar, D. Mellichamp;
John Wiley & Sons; 1989

===== HOUTPUT6 =====

Subrotina HOUTPUT6 : Envio de dados pelos 6 canais da PCL726

Para o funcionamento correcto desta Subrotina
, necess rio colocar os pinos da placa PCL 726
nas seguintes posições:

SW1 &H2C0 (base address)
JP1 U ou B (para cada um dos canais)
JP2 (-5) ou (-10) (para cada um dos canais)

SUB HOUTPUT6 (ppout())

'DIM dat%(4), ARY1%(10), ARY2%(10)

io.port% = &H2C0

FOR i = 1 TO 6

x.value% = ppout(i) * 4096 / 100

hi.byte% = x.value% \ 256

lo.byte% = x.value% MOD 256

OUT io.port% + (i - 1) * 2, hi.byte%

OUT io.port% + (i - 1) * 2 + 1, lo.byte%

NEXT i

=====

END SUB

SECÇÃO	: Aquisição e Envio de dados à escala laboratorial
SUBSECÇÃO	: Subrotina DIGINOUT - Aquisição e Envio de dados através dos canais digitais, V1.0, 1992.08.29
EQUIPAMENTO	: Interface PCL-812PG
SISTEMA OPERATIVO	: MS-DOS, LINGUAGEM : Quick-Basic 4.5
AUTORES	: Vera Lúcia Gomes Mata* Sebastião Feyo de Azevedo**

1. Objectivo

Aquisição e Envio de dados à escala laboratorial através da utilização da interface PCL 812PG.

2. Especificações (Quick-Basic 4.5)

```
SUB DIGINOUT (varinout$, Valor%,er%)  
  'Integer Valor, er
```

3. Descrição

Esta subrotina tem a possibilidade de adquirir dados através de um numero máximo de dezasseis canais digitais de entrada e dezasseis canais digitais de saída.

Utiliza para o efeito a interface PCL 812PG ENHANCED MULTI-LAB CARD desenvolvida pela Advantech Co., Ltd, para computadores IBM PC/XT/AT e compatíveis.

Para mais informação sobre a sua instalação, utilização e programação aconselha-se a consulta da subsecção referente à descrição da interface.

* Aluna Finalista do Curso de Engenharia Química

** Professor Associado da FEUP

4. Parâmetros

Parâmetros de Entrada

varinout\$ - Variável ALFANUMÉRICA, tem dois estados possíveis: "IN" - Entrada de valores digitais
"OUT" - Saída de valores digitais.

Valor% - Variável INTEIRA, é variável de entrada no caso de varinout\$="OUT"; toma valores de 0 a 65535.

Parâmetros de Saída

Valor% - Variável INTEIRA, é variável de saída no caso de varinout\$="IN"; toma valores de 0 a 65535.

er% - Variável INTEIRA, indicador de erro. Os valores possíveis encontram-se na alinea 7 - Indicadores.

5. Rotinas Auxiliares

A comunicação com a interface estabelece-se utilizando as chamadas *Driver Routines*. Neste trabalho chamaram-se as rotinas de: Leitura de valores digitais e a de Escrita de valores digitais..

6. Rotinas a definir pelo utilizador

O utilizador não necessita de definir qualquer rotina.

7. Indicadores

A variável indicadora de erro, er%, assume os seguintes valores à saída da subrotina DIGINOUT:

er% = 0 : Execução bem sucedida
er% = 1 : Falha na inicialização
er% = 13 : Dado de saída inválido (fora da gama 0-65535)
er% = 51 : Variável varinout\$ incorrecta

8. Exemplo

Pretende-se lêr as temperaturas de 4 aquecedores. A gama de temperaturas possível é de 100 - 1000°C e o sinal eléctrico do sistema de detecção-transmissão é, após amplificação, de ± 5 volts.

O objectivo é lêr temperaturas e detectar situações em que os valores estejam a menos de 20% dos limites físicos recomendados (isto é, entre 20 e 80% da gama de funcionamento).

```
DECLARE SUB DIGINOUT (varinout$, Valor%, er%)  
  
varinout$="OUT"  
  
inicio:  
  
INPUT"Valor decimal a ser enviado [0- 65535],-1 para terminar";Valor%  
  
IF valor% =-1 THEN END  
  
CALL DIGINOUT (varinout$, Valor%, er%)  
    IF er% <> 0 THEN  
        LOCATE 10,10  
        PRINT USING"ER%=##";er%: END  
    END IF  
  
GOTO inicio  
END
```

9. Palavras - Chave

Sistema de Aquisição e Envio de Dados
Interface
Canais Digitais de Entrada/Saída

10. Referências

[1] - "Pcl -812 ENHANCED MULTI-LAB CARD - USER'S MANUAL". Maio de 1990.

[2] - "Process Dynamics and control"; D. Seborg, T. Edgar, D. Mellichamp; John Wiley & Sons; 1989

'===== DIGINOUT ====='

' Subrotina DIGINOUT: Aquisição e envio de sinais digitais pela PCL-812PG

'=====

SUB DIGINOUT (varinout\$, Valor%, er%)

'-----

Os parâmetros são:

varinout\$: Variável indicadora de entrada ou saída de valores digitais

Valor% : Variável de entrada ou saída; valor entre 0-65535.

er :^aIndicador de erro

'-----

'----- Inicialização -----'

DIM dat%(4), ary1%(10000), ary2%(10000)

port% = &H220

dat%(0) = port%

dat%(1) = 3

dat%(2) = 3

er% = 0

fun% = 0

CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)

IF er% <> 0 THEN
EXIT SUB

END IF

'-----

'----- Escolha do procedimento IN/OUT -----'

SELECT CASE UCASE\$(varinout\$)

CASE "IN"

GOTO DIN

CASE "OUT"

GOTO DOUT

CASE ELSE

er% = 51: EXIT SUB

END SELECT

'-----

'----- Leitura de Valores Digitais -----'

```
DIN:  
fun% = 22  
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
    IF er% <> 0 THEN  
        EXIT SUB
```

```
END IF
```

```
IF dat%(1) < 128 THEN Valor% = dat%(0) + dat%(1) * 256  
IF dat%(1) > 127 THEN Valor% = dat%(1) * 256! - 65536! + dat%(0)  
EXIT SUB
```

'-----'

'----- Escrita de Valores digitais -----'

```
DOUT:  
fun% = 21  
D = INT(Valor% / 256)  
dat%(1) = D  
dat%(0) = Valor% - D * 256!  
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
    IF er% <> 0 THEN  
        EXIT SUB
```

```
    END IF  
EXIT SUB
```

'-----'

```
END SUB
```

Programa PCLINOUT

São descritas em seguida, algumas das características deste programa .

a) Menu de parâmetros de configuração

Neste ponto do programa, o utilizador define os parâmetros referentes a cada canal.

Assim, após a inicialização do programa e escolhendo no menu principal aquisição ou envio de dados, surge no monitor o menu referente à definição dos parâmetros de configuração para cada um dos canais.

Existem sete opções distintas, conforme se vê no fluxograma correspondente a este programa. Destas, só as opções 6 e 7 é que não dizem respeito ao ficheiro de parâmetros de configuração.

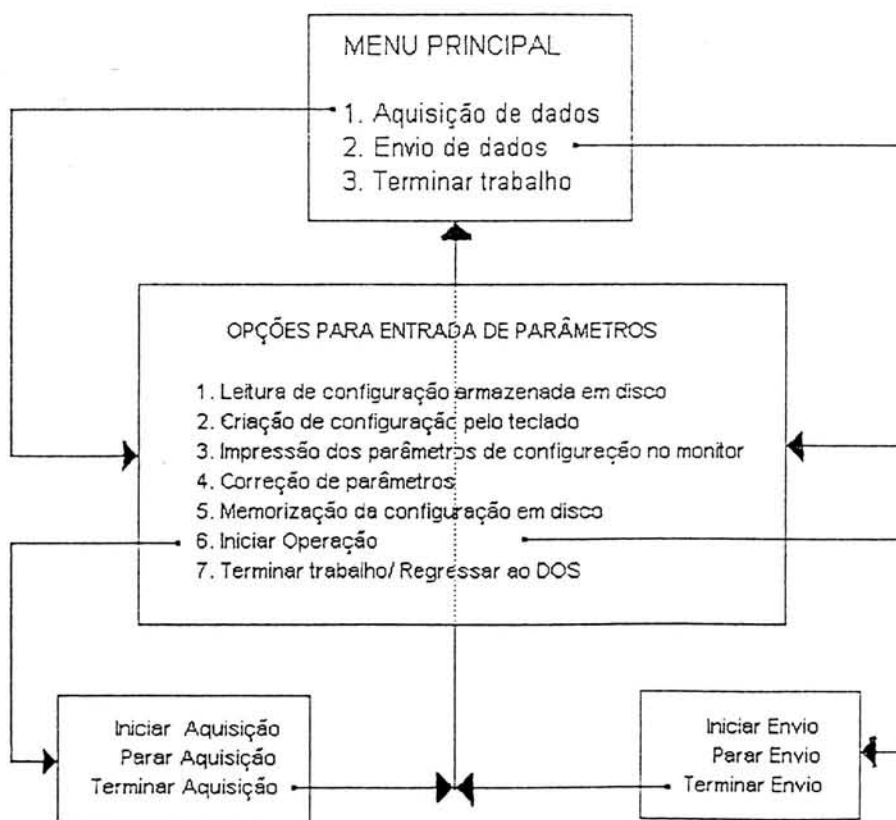


Figura 1: Diagrama esquemático do funcionamento do programa PCLINOUT.

Para iniciar a operação de aquisição ou envio de dados (opção 6) é necessário que o ficheiro de parâmetros esteja na RAM do computador. Este ficheiro contém informações referentes ao sistema e ao(s) canal(is) da(s) interface(s) ligada(s).

Quando se desejar terminar trabalho e regressar ao ambiente DOS, escolhe-se a opção 7.

A opção 1 permite a leitura de um ficheiro de parâmetros já existente em disco enquanto que a opção 5 possibilita a gravação desse mesmo ficheiro em disco, sempre que o utilizador o pretender.

NOTA: Estes ficheiros encontram-se sempre no mesmo local do programa principal, não sendo possível a sua gravação ou leitura noutra local.

A opção 2 anula o ficheiro anterior que se encontra na RAM, inicializando todos os valores, que constam do ficheiro, em zero permitindo ao utilizador a introdução de novos parâmetros.

Caso se pretenda alterar apenas o ficheiro já residente, escolhe-se a opção 4.

A opção 3 possibilita a visualização do ficheiro de parâmetros no monitor, não se efectuando qualquer alteração.

b) *Aquisição de dados*

Os parâmetros de configuração são os seguintes:

- Numero do canal [0-7]
- Estado do canal [ON/OFF]
- Nome da propriedade a medir
- Parâmetros de calibração (polinómio de 2º grau)
- Código da amplitude de entrada [0-4]

Escolhendo a opção 6, dentro do ambiente de Aquisição, o utilizador inicia esta operação. Após a escolha desta não é possível alterar nenhum dos parâmetros definidos anteriormente.

A aquisição efectuar-se-á através da interface PCL-812PG. Esta placa possui 16 canais de entrada, embora neste programa só seja possível a utilização dos oito primeiros.

O contacto efectua-se chamando a subrotina **HINPUTNC** descrita anteriormente.

O programa apresenta de imediato os canais que estão ligados. A inicialização da aquisição através destes canais faz-se quando se premir a tecla ENTER na posição 'iniciar aquisição'. No monitor aparece de imediato o nome e o valor da propriedade medida em percentagem da gama total de entrada correspondente a cada canal assim como o valor da propriedade física correspondente.

Com a ajuda das teclas do cursor, a aquisição pode parar temporariamente se se colocar o cursor na posição 'Parar Aquisição' premindo na tecla 'ENTER'. Este procedimento tem como função uma melhor visualização momentânea dos parâmetros ou quando não se pretender adquirir valores durante um certo periodo de tempo.

Uma vez na opção descrita, restam duas hipóteses: reiniciar a aquisição premindo novamente 'ENTER' na posição 'Iniciar Aquisição' ou volta-se ao menu inicial executando 'Terminar Aquisição'.

c) *Envio de dados:*

Os parâmetros de configuração são os seguintes:

- Numero do canal [0-7]
- Estado do canal [ON/OFF]
- Modo Unipolar, Bipolar ou Débito de corrente [U/B/C/-]
- Tensão de referência interna [5/10]
- Valor da propriedade, em percentagem da gama total de saída.

Escolhendo a opção 6 efectua-se o envio de sinais analógicos pelos canais ligados. Este envio é contínuo e efectua-se através das subrotinas **HOUPUT2** e **HOUTPUT6**.

O procedimento é equivalente ao descrito na alinea anterior para Aquisição. No entanto, caso algum valor tenha de ser alterado, proceder-se-á da seguinte forma:

- 1º Opção 'Terminar Envio'
- 2º Opção 'Envio de dados' (Menu Principal)
- 3º Opção 'Corrigir parâmetros' (Menu de configuração)
- 4º Opção 'Iniciar operação' (Menu de configuração)
- 5º Opção 'Iniciar Envio'.

NOTA: Ter sempre atenção ao parâmetro erro referente às subrotinas de aquisição e/ou envio de dados. Se este valor fôr diferente de zero, aconselha-se a consulta da secção correspondente a descrição das subrotinas de entrada e saída de sinais analógicos.

** Programa PCLINOUT **

'Objectivo: Aquisição remota de dados usando a interface PCL 812-PG

' Programa executado por:

' - Vera Lúcia Gomes Mata

' Sob a Orientação de:

' - Sebastião Feyo de Azevedo

-----declaração das subrotinas-----

DECLARE SUB adcaixa ()
DECLARE SUB adfim ()
DECLARE SUB menu (opcao%)
DECLARE SUB admenu (opcao%)
DECLARE SUB adparam (opcao%)
DECLARE SUB adpasave ()
DECLARE SUB adpload ()
DECLARE SUB adespera (aguardat#)
DECLARE SUB monitbest ()
DECLARE SUB monitconfig ()
DECLARE SUB moniterro ()
DECLARE SUB adacquis ()
DECLARE SUB HINPUTNC (n, c(), ppin(), er%, amp())
DECLARE SUB HOUTPUT2 (ppout(), er%)
DECLARE SUB HOUTPUT6 (ppoute())
DECLARE SUB adcalc (volt(), valor())
DECLARE SUB adsave ()
DECLARE SUB dacalc ()
DECLARE SUB dapaload ()
DECLARE SUB dapasave ()
DECLARE SUB daenviof ()
DECLARE SUB daespera ()
DECLARE SUB dadummy (col1%)
DECLARE SUB daparam (opcao%)

'-----declaração das variaveis comuns a todo o programa-----

```
COMMON SHARED propri$, calib() AS SINGLE, estado$()
COMMON SHARED ndrive AS INTEGER
COMMON SHARED ppin(), c(), n, amp(), er%, volt(), valor(), var, ce(), ind
COMMON SHARED a(), amp$, stat%, dum$, indi, flag%, ampl(), variavel, ae()
COMMON SHARED modubc$, est$, vref(), ppr(), ppout(), ppoute(), g$
```

'-----

'-----determinação da melhor configuração do monitor-----

```
TYPE parametros
  Colors AS INTEGER
  atribs AS INTEGER
  xpix AS INTEGER 'nº de pixeis do monitor na horizontal
  ypix AS INTEGER 'nº de pixeis do monitor na vertical
  tcol AS INTEGER 'nº de colunas
  trow AS INTEGER 'nº de linhas
END TYPE
```

' Screen mode para os vários monitores

```
CONST VGA = 12
CONST MCGA = 11
CONST EGA256 = 9
CONST EGA64 = 8
CONST MONO = 10
CONST HERC = 3
CONST CGA = 2
COMMON SHARED mode AS INTEGER, p AS parametros
'mode- melhor sreen mode obtido
```

CALL monitbest 'determinacao da melhor placa grafica

CALL monitconfig 'obtenção de parametros do monitor para a placa grafica encontrada

'-----

'-----dimensionamento e declaração das variaveis-----

```
DIM propri$(0 TO 7)
DIM gravar$(0 TO 7), calib(0 TO 7, 3) AS SINGLE, estado$(0 TO 7), amp(0 TO 7), c(0
TO 8), ppin(0 TO 7)
DIM a(0 TO 8), amp$(0 TO 8), volt(0 TO 7), valor(0 TO 7)
DIM ampl(0 TO 7), ppout(8), ppoute(0 TO 8), ce(0 TO 8), ae(0 TO 8)

DIM modubc$(8), est$(8), vref(8), ppr(8)
DIM array(8000)
```

'-----apresentação-----

```
TIMER ON
CLS : SCREEN mode
LOCATE 6, 30: PRINT " **PROGRAMA PCLINOUT **"
LOCATE 8, 13: PRINT " Aquisicao remota de dados com interfaces PCL"
x1% = 10 * p.xpix / p.tcol: x2% = p.xpix - x1%
y1% = 4 * p.ypix / p.trow: y2% = 9 * p.ypix / p.trow
LINE (x1%, y1%)-(x2%, y2%), , B
LINE (x1% + 2, y1% + 2)-(x2% - 2, y2% - 2), , B
LOCATE 15, 20: PRINT " VERA LÚCIA GOMES MATA"
LOCATE 16, 20: PRINT " "
LOCATE 17, 20: PRINT " SEBASTIO FEYO de AZEVEDO"
LOCATE 23, 20: PRINT " DEPARTAMENTO de ENG. QUIMICA";
LOCATE 24, 20: PRINT "FACULDADE de ENGENHARIA da UNIVERSIDADE do
PORTO";
LOCATE 25, 20: PRINT " PORTO, 1992";

ti# = TIMER
DO
c$ = INKEY$
tf# = TIMER - ti#
IF c$ <> "" THEN tf# = 10
LOOP WHILE tf# < 9
CLS
```

```
'-----numero de drives do computador-----
LOCATE 10, 30: PRINT "  Numero de drives do computador"
LOCATE 14, 30: PRINT "    1- Um drive (A)"
LOCATE 16, 30: PRINT "    2- Dois drives (A e B)"
```

```
DO
  ndrив$ = INPUT$(1)
LOOP UNTIL ndrив$ = "1" OR ndrив$ = "2"
ndrive = VAL(ndriv$)
'-----
```

```
'-----inicialização de algumas variáveis-----
rfich$ = "pcl"
rdrive$ = "a"
rdirect$ = ""
FOR i = 0 TO 7
  estado$(i) = "OFF"
  est$(i + 1) = "OFF"
NEXT i
'-----
```

inicio:

```
'-----inicialização de variáveis-----
indi = 0
stat% = 1

FOR i = 0 TO 7
  flag4 = 0
NEXT i
'-----
```

```
'----- escolha do procedimento -----
DO
  CALL menu(opcao%)
  SELECT CASE opcao%
    CASE 1 'Aquisição de dados
      GOTO aq
```

```

CASE 2 'Envio de dados
  GOTO env
CASE 3 'Terminar programa
  SYSTEM
  END
END SELECT
LOOP

```

```

'----- introducao de parametros para aquisiçao -----
aq:
DO
CALL admenu(opcao%)
SELECT CASE opcao%
  CASE 1 'leitura de parametros do disco
    CALL adpload
  CASE 2 'introdução de parametros pelo teclado
    FOR i = 0 TO 7
      propri$(i) = ""
      estado$(i) = "OFF"
      calib(i, 1) = 0: calib(i, 2) = 0: calib(i, 3) = 0
      amp(i) = 1
    NEXT i
    CALL adparam(opcao%)
  CASE 3 'visualização dos parametros no monitor
    CALL adparam(opcao%)
  CASE 4 'correção de parametros
    CALL adparam(opcao%)
  CASE 5 'gravação de parametros em disco
    CALL adpasave
  CASE 7 'terminar programa
    SYSTEM
  END
END SELECT
LOOP UNTIL opcao% = 6 'iniciar aquisiçao" de dados
'-----

```

```
'----- definição de variáveis para aquisição -----  
n = 0   'nº de canais ligados  
variavel = 0  
'Escolha da variável 'Temporização' máxima
```

```
FOR i = 0 TO 7  
  IF estado$(i) = "ON" THEN  
    n = n + 1  
  END IF  
NEXT i
```

```
'-----
```

```
'-----aquisição de dados-----  
ON ERROR GOTO erro2
```

```
iniaoq:  
  var = 1  
  CALL adcaixa 'apresentação da estrutura que aparece no visor
```

```
DO  
  CALL adacquis  
  IF stat% = 3 THEN CALL adfim  
  IF variavel = 1 THEN  
    variavel = 0  
    GOTO inicio  
  END IF  
LOOP  
END
```

```
'-----
```

```
'----- Introdução de parâmetros para Envio de Dados -----
```

```
env:  
DO  
CALL admenu(opcao%)  
SELECT CASE opcao%  
  CASE 1 'leitura de parametros do disco  
    CALL dapaload  
  CASE 2 'introdução de parametros pelo teclado
```

```

FOR i = 1 TO 8
  modubc$(i) = "U"
  modubc$(1) = "-": modubc$(2) = "-"
  est$(i) = "OFF"
  vref(i) = 5
  ppout(i) = 50

NEXT i

      CALL daparam(opcao%)
CASE 3 'visualização dos parametros no monitor
      CALL daparam(opcao%)
CASE 4 'correção de parametros
      CALL daparam(opcao%)
CASE 5 'gravação de parametros em disco
      CALL dapasave
CASE 7 'terminar programa
      SYSTEM
      END
END SELECT
LOOP UNTIL opcao% = 6 'iniciar envio de dados
'-----

'-----envio de dados-----
ON ERROR GOTO erro2

inienv:
var = 2
CALL adcaixa 'apresentação da estrutura que aparece no visor
DO
CALL daenviof
IF stat% = 3 THEN CALL adfim
IF variavel = 1 THEN
  variavel = 0
  GOTO inicio
END IF
LOOP

```

END

dados:

```
DATA " Iniciar Aquisição"," Parar Aquisição","Terminar Aquisição"  
DATA " Iniciar Envio  "," Parar Envio  ","Terminar Envio  "
```

-----erros-----

```
moniterro: CALL moniterro  
RESUME
```

```
EGA256erro: mode = EGA64  
RESUME NEXT
```

```
erro0:  
CLS : LOCATE 12, 20  
PRINT "Ocorreu um erro na leitura do ficheiro"  
c$ = INPUT$(1)  
CLS  
RESUME
```

```
erro1:  
CLS : LOCATE 12, 20  
SELECT CASE ERR  
CASE 53  
PRINT "File not found"  
CASE 71  
PRINT "Drive not ready."  
CASE 76  
PRINT "Path not fond."  
CASE 70  
PRINT "Permission denied."  
CASE ELSE  
PRINT "Ocoreu um erro no programa."  
END SELECT  
LOCATE 25, 20: PRINT "Prima qualquer tecla para continuar";  
c$ = INPUT$(1)  
RESUME inicio:
```

```

erro2:
  CLS : LOCATE 12, 20
  SELECT CASE ERR
    CASE 24, 25
      PRINT "Ocorreu um erro na impressora"
      impri$ = "OFF"
    CASE 53, 70, 71, 76
      PRINT "Ocorreu um erro na gravacao dos resultados"
      FOR i = 0 TO 7
        gravar$(i) = "OFF"
      NEXT i
    CASE ELSE
      PRINT "Ocorreu um erro no programa"
  END SELECT
  ti = TIMER
  DO
    c$ = INKEY$
    tf = TIMER - ti
    IF c$ <> "" THEN tf = 4
    LOOP WHILE tf < 3

    FOR i = 0 TO 7
      clinha(i) = 1
    NEXT i
  CLS
  RESUME iniaq

```

```

===== ADAQUIS =====
'Esta subrotina controla os parâmetros necessários à subrotina de aquisição
'HINPUTNC e estabelece contacto com esta.
=====

```

```

SUB adaquis

  indic = 0
  indi = indi + 1
  IF indi > 1 THEN GOTO a
  FOR i = 0 TO 7
    IF estado$(i) = "on" OR estado$(i) = "ON" THEN
      indic = indic + 1
      c(indic) = i
    END IF
  NEXT i

```



```

FOR i = 0 TO 7: LOCATE i * 2 + 5, 65: PRINT "OFF"
      LOCATE 2 * i + 5, 8: PRINT i
NEXT i
COLOR 7
FOR i = 1 TO n
  LOCATE 2 * c(i) + 5, 65: PRINT " ON"
      LOCATE 2 * c(i) + 5, 6: PRINT USING "## \- \"; c(i); propri$(c(i))

NEXT i
RESTORE dados
FOR i = 1 TO 3
  LOCATE 25 + i, 5
  IF i = 1 THEN COLOR 15 ELSE COLOR 7
  READ dummy$
  PRINT dummy$
NEXT i
LOCATE 25, 1
flag% = 1
stat% = 0
a: dum$ = INKEY$

  IF dum$ <> "" THEN
    IF LEFT$(dum$, 1) = CHR$(0) AND RIGHTS$(dum$, 1) = CHR$(72) THEN flag%
= flag% - 1
    IF LEFT$(dum$, 1) = CHR$(0) AND RIGHTS$(dum$, 1) = CHR$(80) THEN flag%
= flag% + 1
    IF dum$ = CHR$(13) THEN stat% = flag%
    IF flag% < 1 THEN flag% = 3
    IF flag% > 3 THEN flag% = 1
    IF stat% = 3 THEN dum$ = "fim"

RESTORE dados

FOR i% = 1 TO 3
  LOCATE 25 + i%, 5
  IF flag% = i% THEN COLOR 15 ELSE COLOR 7
  READ dummy$
  PRINT dummy$
NEXT i%
COLOR 7
END IF

```

```
IF stat% = 1 THEN
```

```
CALL HINPUTNC(n!, c(), ppin(), er%, amp())
```

```
    LOCATE 23, 41: PRINT er%
```

```
CALL adcalc(volt(), valor())
```

```
END IF
```

```
=====
```

```
END SUB
```

```
===== ADCAIXA =====
```

```
'A subrotina desenha a estrutura com que os dados vão sendo visualizados durante a  
aquisição.
```

```
=====
```

```
SUB adcaixa
```

```
    CLS
```

```
    SCREEN 12
```

```
    COLOR 10
```

```
SELECT CASE var
```

```
    CASE 1
```

```
        LOCATE 13, 8
```

```
        PRINT " ESTA SUBROTINA FAZ A AQUISIÇÃO DE PROPRIEDADES  
EXPERIMENTAIS"
```

```
        LOCATE 16, 28
```

```
        PRINT "A ESCALA LABORATORIAL"
```

```
    CASE 2
```

```
        LOCATE 13, 15
```

```
        PRINT " ESTA SUBROTINA FAZ O ENVIO DE PROPRIEDADES "
```

```
        LOCATE 16, 28
```

```
        PRINT "A ESCALA LABORATORIAL"
```

```
END SELECT
```

```
    COLOR 2
```

```
    LOCATE 28, 25
```

```
    PRINT "Pressione qualquer tecla ..."
```

```
    DO
```

```
    dum$ = INKEY$
```

```
    LOOP UNTIL dum$ <> ""
```

```
caixa:
```

```
    CLS
```

```
    COLOR 10
```

```

LINE (0, 0)-(639, 335), , B
LINE (2, 2)-(637, 333), , B
LINE (2, 42)-(637, 42)
LINE (230, 335)-(409, 380), , B
LINE (232, 337)-(407, 378), , B
LINE (0, 385)-(200, 459), , B
COLOR 2
LOCATE 2, 7
PRINT "CANAL"
LOCATE 2, 23
PRINT "TENSAO (V)"
LOCATE 2, 42
PRINT "PROPRIEDADE "
LOCATE 2, 66
PRINT "ESTADO"
LOCATE 23, 34
PRINT "ERRO :"
LOCATE 27, 28
PRINT "Use as setas para escolher"
LOCATE 28, 30
PRINT "e ENTER para confirmar."

```

```

=====
END SUB

```

```

===== ADCALC =====

```

```

' Esta subrotina calcula, para cada canal, os valores da propriedade lida
' e da tensão, enviando estes valores para o ecran.
=====

```

```

SUB adcalc (volt(), valor())

```

```

FOR i = 1 TO n
a(c(i)) = ppin(c(i)) / 100
ampl(c(i)) = 2 ^ amp(c(i))
volt(c(i)) = 10 / ampl(c(i)) * (a(c(i)) + .5) - 5 / ampl(c(i))
valor(c(i)) = calib(c(i), 1) + calib(c(i), 2) * volt(c(i)) + calib(c(i), 3) * volt(c(i)) ^ 2

```

```

LOCATE 5 + 2 * c(i), 25: PRINT USING "##.###"; volt(c(i))
LOCATE 5 + 2 * c(i), 45: PRINT USING "####.#"; valor(c(i))
NEXT i

```

```

=====
END SUB

```

```

===== adespera =====
' Nesta subrotina o computador faz uma pausa de, aguardat, segundos.
=====
'
SUB adespera (aguardat#)
    tinic# = TIMER
    DO
        LOOP WHILE (TIMER - tinic#) < aguardat#
=====
END SUB

```

```

===== ADFIM =====
'Subrotina que finaliza o programa de aquisição de dados.
=====
SUB adfim

CLS
COLOR 2
LOCATE 16, 30
PRINT "Retorno ao Menu Principal ..."
COLOR 10
LOCATE 28, 31
PRINT "Confirma (S/N) ?"
DO
    dum$ = INKEY$
    IF dum$ = "S" OR dum$ = "s" THEN
        variavel = 1
        EXIT SUB
    END IF
LOOP UNTIL dum$ = "N" OR dum$ = "n"
dum$ = "." : indi = 0
CALL adcaixa
END SUB

```

```

===== admenu =====
' A subrotina imprime no visor as opções para entrada de parâmetros e
' detecta qual a opcao escolhida.
=====
'
SUB admenu (opcao%)
CLS : SCREEN 0
LOCATE 6, 20: PRINT "OPÇÕES PARA ENTRADA DE PARÂMETROS"
LOCATE 10, 20: PRINT "1- Leitura de configuração armazenada em disco"
LOCATE 11, 20: PRINT "2- Criação da configuração pelo teclado"
LOCATE 12, 20: PRINT "3- Impressão dos parâmetros de configuração no visor"
LOCATE 13, 20: PRINT "4- Correção de parâmetros"
LOCATE 14, 20: PRINT "5- Memorização de configuração em disco"
LOCATE 15, 20: PRINT "6- Iniciar operação"
LOCATE 16, 20: PRINT "7- Terminar trabalho / Regressar ao DOS"
DO
  opcao$ = INKEY$
  opcao% = VAL(opcao$)
LOOP WHILE opcao% < 1 OR opcao% > 7
=====
END SUB

```

```

===== adpload =====
' A subrotina lê do disco os parâmetros de configuração.
=====
'
SUB adpload
CLS : SCREEN 0
DO
  LOCATE 12, 47: PRINT " ";
  LOCATE 12, 30: PRINT "Nome do ficheiro: "; : INPUT fich$
LOOP UNTIL fich$ = "" OR LEN(fich$) < 13
IF fich$ = "" THEN EXIT SUB
bsl% = INSTR(1, fich$, ":")
IF bsl% > 0 THEN fich$ = RIGHT$(fich$, LEN(fich$) - bsl%)
DO
  bsl% = INSTR(1, fich$, "\")
  IF bsl% > 0 THEN fich$ = RIGHT$(fich$, LEN(fich$) - bsl%)
LOOP WHILE bsl% <> 0

```

```

ON ERROR GOTO erro1
OPEN fich$ FOR INPUT AS #1
INPUT #1, lixo$
FOR i = 0 TO 7
    INPUT #1, estado$(i)
    INPUT #1, propri$(i)
    INPUT #1, calib(i, 1)
    INPUT #1, calib(i, 2)
    INPUT #1, calib(i, 3)
    INPUT #1, amp(i)
NEXT i

CLOSE #1

```

```

=====
END SUB

```

```

===== ADPARAM =====
'Esta subrotina permite os parâmetros necessários à subrotina de aquisição
'HINPUTNC e estabelece contacto com esta.
=====

```

```

SUB adparam (opcao%)
DO
CLS : SCREEN 0
LOCATE 2, 35: PRINT "PARÂMETROS"
LOCATE 3, 1: PRINT "-----"
----"
LOCATE 4, 1: PRINT "Canal    0    1    2    3    4    5    6    7"
LOCATE 5, 1: PRINT "Gama/Código"
LOCATE 6, 1: PRINT "-----"
----"
LOCATE 7, 1: PRINT "Estado"
LOCATE 8, 1: PRINT "Propri."
LOCATE 9, 1: PRINT "-----"
----"
LOCATE 10, 3: PRINT "a"
LOCATE 11, 3: PRINT "b"
LOCATE 12, 3: PRINT "c"
LOCATE 13, 1: PRINT "-----"
----"

```

```

FOR i = 0 TO 7
  LOCATE 7, i * 9 + 12: PRINT estado$(i)
  LOCATE 8, i * 9 + 12: PRINT propri$(i)
  LOCATE 10, i * 9 + 12: PRINT calib(i, 1)
  LOCATE 11, i * 9 + 12: PRINT calib(i, 2)
  LOCATE 12, i * 9 + 12: PRINT calib(i, 3)
  LOCATE 5, i * 9 + 12: PRINT amp(i)

NEXT i

SELECT CASE opcao%
  CASE 2
    COLOR 0, 15
    LOCATE 25, 3: PRINT "[ESC] - Introduzir Novos Parâmetros
[T]erminar ";
    COLOR 7, 0
  CASE 3
    COLOR 0, 15
    LOCATE 25, 20: PRINT "Prima qualquer tecla para continuar";
    COLOR 7, 0
    c$ = INPUT$(1)
    EXIT SUB
  CASE 4
    COLOR 0, 7
    LOCATE 25, 3: PRINT "[ESC] - Introduzir Novos Parâmetros
[T]erminar ";
    COLOR 7, 0
END SELECT
DO
  opc$ = INPUT$(1)
  opc$ = UCASE$(opc$)
LOOP UNTIL opc$ = CHR$(27) OR opc$ = "T"
SELECT CASE opc$
  CASE CHR$(27)
    LOCATE 25, 1: PRINT " ";
    LOCATE 16, 1, 1: PRINT "Canal: ";
  DO
  DO
  canal1$ = INKEY$
  LOOP WHILE canal1$ = ""
  LOOP UNTIL canal1$ = CHR$(13) OR canal1$ = CHR$(13) OR (ASC(canal1$) >=
48 AND ASC(canal1$) <= 55)
  canal = VAL(canal1$)

```

```

IF canal1$ <> CHR$(13) THEN
LOCATE 16, 1, 0: PRINT "Canal"; canal; : PRINT ":";
DO
LOCATE 16, 11: PRINT "- Estado[ON/OFF]      "; : INPUT estado1$
estado1$ = UCASE$(estado1$)
LOOP UNTIL estado1$ = "ON" OR estado1$ = "OFF" OR estado1$ = ""
IF estado1$ <> "" THEN estado$(canal) = estado1$
LOCATE 17, 11: PRINT "- Propriedade      "; : INPUT propri1$
IF propri1$ <> "" THEN propri$(canal) = LEFT$(propri1$, 8)
DO
LOCATE 18, 11: PRINT "- Amplitude [0-4]      "; : INPUT ; amp1$
LOOP UNTIL amp1$ = "0" OR amp1$ = "1" OR amp1$ = "2" OR amp1$ = "3" OR
amp1$ = "4"
IF amp1$ <> "" THEN amp$(canal) = VAL(amp1$)
COLOR 15, 0
LOCATE 20, 1: PRINT "Calibraç,,:o: "; propri$(canal); " = a + b*V + c*V^2"
COLOR 7, 0
LOCATE 21, 11: PRINT "- a      "; : INPUT ; calib1$
IF calib1$ <> "" THEN calib$(canal, 1) = VAL(calib1$)
LOCATE 22, 11: PRINT "- b      "; : INPUT ; calib1$
IF calib1$ <> "" THEN calib$(canal, 2) = VAL(calib1$)
LOCATE 23, 11: PRINT "- c      "; : INPUT ; calib1$
IF calib1$ <> "" THEN calib$(canal, 3) = VAL(calib1$)
END IF
END SELECT
LOOP UNTIL opc$ = "T"
'=====
END SUB

```

```

'===== adpasave =====
' A subrotina grava em disco os parâmetros de configuração.
'=====
'

```

```

SUB adpasave
CLS : SCREEN 0

```

```

DO
LOCATE 12, 47: PRINT "      ";
LOCATE 12, 30: PRINT "Nome do ficheiro:"; : INPUT fich$
LOOP UNTIL fich$ = "" OR LEN(fich$) < 13
IF fich$ = "" THEN EXIT SUB

```



```

bsl% = INSTR(1, fich$, ":")
IF bsl% > 0 THEN fich$ = RIGHTS$(fich$, LEN(fich$) - bsl%)
DO
  bsl% = INSTR(1, fich$, "\")
  IF bsl% > 0 THEN fich$ = RIGHTS$(fich$, LEN(fich$) - bsl%)
LOOP WHILE bsl% <> 0

ON ERROR GOTO erro1
OPEN fich$ FOR OUTPUT AS #1
PRINT #1, "Parametros referentes a aquisição para o programa PCLINOUT"
FOR i = 0 TO 7
  PRINT #1, estado$(i)
  PRINT #1, propri$(i)
  PRINT #1, calib(i, 1)
  PRINT #1, calib(i, 2)
  PRINT #1, calib(i, 3)
  PRINT #1, amp(i)
NEXT i

CLOSE #1

'=====
END SUB

'===== DACALC =====
'Calculo dos valores de tensao enviados para o exterior      ||
'=====
'
SUB dacalc

FOR i = 1 TO n
  SELECT CASE modubc$(ce(i))
    CASE "-", "u", "U", "c", "C"
      ae(i) = vref(ce(i)) * ppout(ce(i)) / 100
    CASE "b", "B"
      ae(i) = 2 * vref(ce(i)) * (ppout(ce(i))) / 100 - vref(ce(i))
  END SELECT
  LOCATE 3 + 2 * ce(i), 25: PRINT USING "##.###"; ae(i)
  LOCATE 3 + 2 * ce(i), 45: PRINT USING "#####.# %"; ppout(ce(i))
NEXT i
'=====

END SUB

```

'===== DADUMMY =====

SUB dadummy (col1%)

RESTORE dados

FOR i% = 1 TO 3

IF flag% = i% THEN COLOR 15 ELSE COLOR 7

READ dum\$

LOCATE 25 + i%, 5

PRINT dum\$

NEXT i%

COLOR col1%

'=====

END SUB

'===== DAENVIOF =====

'Esta subrotina controla os parâmetros necessários às subrotinas de envio
'HOUTPUT2 e HOUTPUT6 e estabelece contacto com as interfaces PCL812PG e
PCL726

'=====

SUB daenviof

indic = 0

indi = indi + 1

IF indi > 1 THEN GOTO ae

n = 0

FOR i = 1 TO 8

IF est\$(i) = "on" OR est\$(i) = "ON" THEN

n = n + 1

indic = indic + 1

ce(indic) = i

END IF

NEXT i

FOR i = 1 TO 8: LOCATE i * 2 + 3, 65: PRINT "OFF"

LOCATE 2 * i + 3, 8: PRINT i

NEXT i

COLOR 7

FOR i = 1 TO n

LOCATE 2 * ce(i) + 3, 65: PRINT " ON"

LOCATE 2 * ce(i) + 3, 8: PRINT ce(i)

```

NEXT i
RESTORE dados
FOR i = 1 TO 6
  LOCATE 25 + i - 3, 5
  IF i = 4 THEN COLOR 15 ELSE COLOR 7
  READ dummy$
  IF i > 3 THEN PRINT dummy$
NEXT i
LOCATE 25, 1
flag% = 1
stat% = 0
tempo2 = TIMER - 100
ae: dum$ = INKEYS
IF dum$ <> "" THEN
  IF LEFT$(dum$, 1) = CHR$(0) AND RIGHTS(dum$, 1) = CHR$(72) THEN flag%
= flag% - 1
  IF LEFT$(dum$, 1) = CHR$(0) AND RIGHTS(dum$, 1) = CHR$(80) THEN flag%
= flag% + 1
  IF dum$ = CHR$(13) THEN stat% = flag%
  IF flag% < 1 THEN flag% = 3
  IF flag% > 3 THEN flag% = 1
  IF stat% = 3 THEN dum$ = "fim"

RESTORE dados

FOR i% = 1 TO 6
  LOCATE 25 + i% - 3, 5
  IF flag% = i% - 3 THEN COLOR 15 ELSE COLOR 7
  READ dummy$
  IF i% > 3 THEN PRINT dummy$
NEXT i%
COLOR 7
END IF

IF stat% = 1 THEN
CALL dacalc
CALL HOUTPUT2(ppout(), er%)
FOR i = 3 TO 8
ppoute(i - 2) = ppout(i)
NEXT i
CALL HOUTPUT6(ppoute())
  LOCATE 23, 41: PRINT er%
END IF
=====
END SUB

```

```

===== DAESPERA =====
'
SUB daespera
  LOCATE 28, 28
  PRINT "Pressione qualquer tecla ..."
  DO
    dum$ = INKEY$
  LOOP UNTIL dum$ <> ""
=====
END SUB

```

```

===== ADPALOAD =====
' A subrotina lê do disco os parâmetros de configuração.
=====
'
SUB dapaload

CLS : SCREEN 0
DO
  LOCATE 12, 47: PRINT " ";
  LOCATE 12, 30: PRINT "Nome do ficheiro:"; : INPUT fich$
LOOP UNTIL fich$ = "" OR LEN(fich$) < 13
IF fich$ = "" THEN EXIT SUB
bsl% = INSTR(1, fich$, ":")
IF bsl% > 0 THEN fich$ = RIGHT$(fich$, LEN(fich$) - bsl%)
DO
  bsl% = INSTR(1, fich$, "\")
  IF bsl% > 0 THEN fich$ = RIGHT$(fich$, LEN(fich$) - bsl%)
LOOP WHILE bsl% <> 0

ON ERROR GOTO erro1
OPEN fich$ FOR INPUT AS #2
INPUT #2, lixo$
FOR i = 1 TO 8
  modubc$(1) = "-": modubc$(2) = "-"
  INPUT #2, modubc$(i)
  INPUT #2, est$(i)
  INPUT #2, vref(i)
  INPUT #2, ppout(i)

```

NEXT i

CLOSE #2

=====

END SUB

SUB daparam (opcao%)

DO

CLS : SCREEN 0

LOCATE 1, 1: PRINT " |<- PCL 812-PG ->|<----- PCL 726 -----
----->|"

LOCATE 2, 1: PRINT "-----
----"

LOCATE 3, 1: PRINT "Canal 1 2 3(1) 4(2) 5(3) 6(4) 7(5) 8(6)"

LOCATE 5, 1: PRINT "-----
----"

'propriedade

LOCATE 6, 1: PRINT "Estado"

LOCATE 7, 1: PRINT "Modo U/B/C/- "

LOCATE 8, 1: PRINT "V.Ref. "

LOCATE 9, 1: PRINT "Propr.,%"

LOCATE 10, 1: PRINT "-----
----"

modubc\$(1) = "-": modubc\$(2) = "-"

FOR i = 1 TO 8

 j = i - 1

 LOCATE 7, j * 9 + 14: PRINT UCASE\$(modubc\$(i))

 LOCATE 6, j * 9 + 13: PRINT est\$(i)

 LOCATE 8, j * 9 + 13: PRINT USING "##"; vref(i)

 LOCATE 9, j * 9 + 12: PRINT ppout(i)

NEXT i

SELECT CASE opcao%

 CASE 2

 COLOR 0, 15

 LOCATE 25, 3: PRINT "[ESC] - Introduzir Novos Parfmetros

[T]erminar ";

 COLOR 7, 0

 CASE 3

 COLOR 0, 15

```

LOCATE 25, 20: PRINT "Prima qualquer tecla para continuar";
COLOR 7, 0
c$ = INPUT$(1)
EXIT SUB
CASE 4
COLOR 0, 7
LOCATE 25, 3: PRINT "[ESC] - Introduzir Novos Parfmetros
[T]erminar ";
COLOR 7, 0
END SELECT
DO
opc$ = INPUT$(1)
opc$ = UCASE$(opc$)
LOOP UNTIL opc$ = CHR$(27) OR opc$ = "T"
SELECT CASE opc$
CASE CHR$(27)
LOCATE 25, 1: PRINT " ";
LOCATE 16, 1, 1: PRINT "Canal: ";
DO
DO
canal1$ = INKEY$
LOOP WHILE canal1$ = ""
LOOP UNTIL canal1$ = CHR$(13) OR canal1$ = CHR$(13) OR (ASC(canal1$) >=
49 AND ASC(canal1$) <= 56)
canal = VAL(canal1$)
IF canal1$ <> CHR$(13) THEN
LOCATE 16, 1, 0: PRINT "Canal"; canal; : PRINT ":";
DO
LOCATE 16, 11: PRINT "- Estado[ON/OFF] "; : INPUT estado1$
estado1$ = UCASE$(estado1$)
LOOP UNTIL estado1$ = "ON" OR estado1$ = "OFF" OR estado1$ = ""
IF estado1$ <> "" THEN est$(canal) = estado1$
LOCATE 17, 11: PRINT "- Modo U/B/C/- "; : INPUT modubc$
IF modubc$ <> "" THEN modubc$(canal) = modubc$
LOCATE 18, 11: PRINT "- Tens.,o de Ref.[5/10] "; : INPUT vref$
IF vref$ <> "" THEN vref$(canal) = VAL(vref$)
LOCATE 19, 11: PRINT "- Propriedade, % "; : INPUT ppr$
IF ppr$ <> "" THEN ppout(canal) = VAL(ppr$)
END IF
END SELECT
LOOP UNTIL opc$ = "T"

END SUB

```

```

===== adpasave =====
' A subrotina grava em disco os parâmetros de configuração.
'
SUB dapasave

CLS : SCREEN 0

DO
  LOCATE 12, 47: PRINT " ";
  LOCATE 12, 30: PRINT "Nome do ficheiro:"; : INPUT fich$
LOOP UNTIL fich$ = "" OR LEN(fich$) < 13
IF fich$ = "" THEN EXIT SUB

bsl% = INSTR(1, fich$, ":")
IF bsl% > 0 THEN fich$ = RIGHT$(fich$, LEN(fich$) - bsl%)
DO
  bsl% = INSTR(1, fich$, "\")
  IF bsl% > 0 THEN fich$ = RIGHT$(fich$, LEN(fich$) - bsl%)
LOOP WHILE bsl% <> 0

ON ERROR GOTO erro1
OPEN fich$ FOR OUTPUT AS #2
PRINT #2, "Parâmetros para envio pelas interfaces PCL 812-PG e PCL 726"
modubc$(1) = "-": modubc$(2) = "-"
FOR i = 1 TO 8
  PRINT #2, modubc$(i)
  PRINT #2, est$(i)
  PRINT #2, vref(i)
  PRINT #2, ppout(i)
NEXT i

CLOSE #2

=====

END SUB

```

```

===== HINPUTNC - Canais Não Consecutivos =====
Subrotina utilizada para aquisição de NCANAIS não consecutivos.
Os parâmetros são:

ncanais : Numero Total de Canais Não Consecutivos (1-16)
c()      : Numero do Canal (0-15)
ppin()   : Valor Percentual da Propriedade Lida
ER       : Indicador de Erro
amp()    : Código Correspondente à Amplitude do Canal

Para o funcionamento correcto desta Subrotina é
necessário colocar os pinos da placa PCL 812-PG
nas seguintes posições:

BASE ADDRESS - HEX 220
JP1          - INTERNAL
JP5          - X (Nulo)
JP6, JP7    - X (Nulo)
=====

```

```

SUB HINPUTNC (n, c(), ppin(), er%, amp())

```

```

REDIM dat%(20), ary1%(20), ary2%(20)

```

```

ind = ind + 1

```

```

IF ind > 1 THEN GOTO CAD

```

```

----- Inicialização da placa -----

```

```

port% = &H220: '          Porta-Adereço I/O

```

```

dat%(0) = port%

```

```

dat%(1) = 3: '          Selecção de IRQ 3

```

```

dat%(2) = 1: '          Selecção de DMA 1

```

```

er% = 0: '          Indicador de Erro

```

```

fun% = 0: '          Função 0

```

```

CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)

```

```

IF er% <> 0 THEN

```

```

    EXIT SUB

```

```

END IF

```


'----- Estabelecimento dos Canais de Entrada -----

```
dat%(0) = c(1): '           Primeiro Canal
dat%(1) = c(n): '         Ultimo Canal
fun% = 1: '               Função 1
```

```
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
IF er% <> 0 THEN
    EXIT SUB
END IF
```

'----- Estabelecimento das Gamas de Entrada dos Canais -----

```
FOR i = 1 TO n
dat%(0) = c(i): '           Canal i
pp = c(i)
dat%(1) = amp(pp): '       Código da Amplitude do Canal i
fun% = 23: '               Função 23
```

```
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
IF er% <> 0 THEN
    EXIT SUB
END IF
```

```
NEXT i
```

```
CAD:
```

'----- Conversão Analógico-Digital -----

```
dat%(1) = 0
fun% = 3: '               Função 3
FOR j = 1 TO n
```

```
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
IF er% <> 0 THEN
    EXIT SUB
END IF
```

```
ppin(dat%(1)) = dat%(0)
ppin(dat%(1)) = ppin(dat%(1)) / 4095 * 100 ' 0 % < ppin(j) < 100 %
```

```
NEXT j
```

```
-----
END SUB
```

```
===== HOUTPUT2 =====
' Subrotina HOUTPUT2: Envio de dados pelos 2 canais da PCL-812PG
=====
```

```
' Para o funcionamento correcto desta Subrotina é
' necessário colocar os pinos da placa PCL 812-PG
' nas seguintes posições:
```

```
' BASE ADDRESS - HEX 220
' JP8, JP9 - INTERNAL
'
```

```
-----
SUB HOUTPUT2 (ppout(), er%)
```

```
REDIM ary1%(20), ary2%(20)
```

```
REDIM dat%(20)
```

```
port% = &H220
dat%(0) = port%
dat%(1) = 3
dat%(2) = 1
er% = 0
fun% = 0
```

```
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
IF er% <> 0 THEN
    EXIT SUB
END IF
```

```
dat%(0) = INT(ppout(1) * 4095 / 100)
dat%(1) = INT(ppout(2) * 4095 / 100)
```

```
fun% = 16
```

```
CALL pcl812(fun%, SEG dat%(0), SEG ary1%(0), SEG ary2%(0), er%)
```

```
IF er% <> 0 THEN  
    EXIT SUB
```

```
END IF
```

```
END SUB
```

```
===== HOUTPUT6 =====  
' Subrotina HOUTPUT6 : Envio de dados pelos 6 canais da PCL726  
=====
```

```
'  
' Para o funcionamento correcto desta Subrotina  
' é necessário colocar os pinos da placa PCL 726  
' nas seguintes posições:  
'  
' SW1 &H2C0 (base address)  
' JP1 U ou B (para cada um dos canais)  
' JP2 (-5) ou (-10) (para cada um dos canais)  
'  
'  
-----
```

```
SUB HOUTPUT6 (ppout())
```

```
'DIM dat%(4), ARY1%(10), ARY2%(10)
```

```
io.port% = &H2C0
```

```
FOR i = 1 TO 6
```

```
x.value% = ppout(i) * 4096 / 100
```

```
hi.byte% = x.value% \ 256
lo.byte% = x.value% MOD 256
```

```
OUT io.port% + (i - 1) * 2, hi.byte%
OUT io.port% + (i - 1) * 2 + 1, lo.byte%
```

```
NEXT i
```

```
=====
END SUB
```

```
===== MENU =====
```

```
SUB menu (opcao%)
```

```
CLS : SCREEN 0
```

```
LOCATE 6, 20: PRINT "      MENU PRINCIPAL      "
```

```
LOCATE 10, 20: PRINT "  1 - Aquisição de Dados"
```

```
LOCATE 11, 20: PRINT "  2 - Envio de Dados "
```

```
LOCATE 15, 20: PRINT "  3 - Terminar trabalho "
```

```
DO
```

```
  opcao$ = INKEY$
```

```
  opcao% = VAL(opcao$)
```

```
LOOP WHILE opcao% < 1 OR opcao% > 3
```

```
=====
END SUB
```

```
===== monitbest =====
```

```
' Esta subrotina encontra a melhor placa grafica que o computador possui.
' Começa por assumir o melhor screen mode e verifica se ele funciona. Em
' caso de erro, passa para o screen mode seguinte, fazendo isto continua-
' mente at, encontrar algo que funcione.
```

```
=====
SUB monitbest
```

```
  ' Assume o melhor screen mode
```

```
  mode = VGA
```

```
  ON ERROR GOTO moniterro
```

```
  'testa o screen mode at, algum funcionar
```

```
  SCREEN mode
```

```

' se a placa , EGA, verifica se possui mais de 64K
ON ERROR GOTO EGA256erro
IF mode = EGA256 THEN SCREEN 8, , 1
ON ERROR GOTO 0 'em caso de ocorrer outro erro qualquer abandona o programa
    'apresentando a mensagem do erro que ocorreu
'regresso ao modo de texto
SCREEN 0, , 0
WIDTH 80, 25

```

```

=====
END SUB

```

```

===== MONITCONFIG =====

```

```

' Esta subrotina atribui valores aos vários parâmetros ligados ao
' ecran conforme o modo encontrado na subrotina MODE.
=====

```

```

SUB monitconfig
SELECT CASE mode
CASE 2 'monitor CGA
    p.Colors = 0
    p.atrifs = 2
    p.xpix = 639
    p.ypix = 199
    p.tcol = 80
    p.trow = 25
CASE 3 'monitor Hercules
    p.Colors = 0
    p.atrifs = 2
    p.xpix = 720
    p.ypix = 348
    p.tcol = 80
    p.trow = 25
CASE 8 'monitor EGA 64K
    p.Colors = 16
    p.atrifs = 16
    p.xpix = 639
    p.ypix = 199
    p.tcol = 80
    p.trow = 25
CASE 9 ' monitor EGA 256K
    p.Colors = 64

```

```

p.atrifs = 16
p.xpix = 639
p.ypix = 349
p.tcol = 80
p.trow = 25
CASE 10 ' monitor monocrom tico EGA ou VGA
p.Colors = 0
p.atrifs = 2
p.xpix = 319
p.ypix = 199
p.tcol = 80
p.trow = 25
CASE 11 ' monitor MCGA
p.Colors = 216
p.atrifs = 2
p.xpix = 639
p.ypix = 479
p.tcol = 80
p.trow = 30
CASE 12 ' monitor VGA
p.Colors = 216
p.atrifs = 16
p.xpix = 639
p.ypix = 479
p.tcol = 80
p.trow = 30
CASE ELSE 'n,,o tem placa gr fica
p.Colors = 16
p.atrifs = 16
p.xpix = 0
p.ypix = 0
p.tcol = 80
p.trow = 25
CLS : SCREEN 0
LOCATE 12, 10: PRINT "O programa não pode ser executado, porque não existe
nenhuma placa gr fica"
c$ = INPUT$(1)
SYSTEM
EXIT SUB
END SELECT
=====
END SUB

```

DEFINT A-Z

```
'===== moniterro ====='  
' Esta subrotina , chamada aquando a ocorrência de um erro na instrução  
' screen, na subrotina que testa o tipo de monitor e placa. Conforme o  
' valor da variavel mode assume-se um outro screen mode para essa variavel.  
'=====
```

SUB moniterro

SELECT CASE mode

CASE VGA

mode = MCGA

CASE MCGA

mode = EGA256

CASE EGA256

mode = CGA

CASE CGA

mode = MONO

CASE MONO

mode = HERC

CASE ELSE

PRINT "Placa grafica n,,o encontrada"

END

END SELECT

```
'=====
```

END SUB



FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

BIBLIOTECA



0000101480