

UNIVERSIDADE DO MINHO

ESCOLA DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA TÊXTIL



MESTRADO EM ENGENHARIA TÊXTIL

**“MEDIÇÃO E ANÁLISE DE PARÂMETROS EM MÁQUINA DE
COSTURA INDUSTRIAL”**

Anexos

Mestrando: Helder Manuel Teixeira Carvalho

Orientadores: Doutor Fernando Nunes Ferreira

Doutor João Monteiro

Guimarães

Janeiro 1997

Anexos

I. Descrição dos módulos de software desenvolvidos.....	4
I.1. Estruturas de dados.....	4
I.1.1. Definição de tipos – type data info.vi	4
I.1.2. Variáveis globais.....	5
I.1.2.1. Variáveis globais de configuração: globals config.vi.....	5
I.1.2.2. Variáveis globais de dados (sinais): globals data.vi.....	7
I.2. Subrotinas ou VI's (Virtual Instruments)	8
I.2.1. VI's de configuração.....	8
I.2.1.1. Calculate fs.vi	8
I.2.1.2. Check config canais.vi.....	9
I.2.1.3. Configurar.vi	10
I.2.2. VI's de aquisição	10
I.2.2.1. Adquirir.vi	10
I.2.2.2. Adquirir osciloscópio.vi.....	11
I.2.2.3. Aquisição valores.vi	11
I.2.2.4. Osciloscópio.vi	12
I.2.2.5. Teste de saturação.vi	13
I.2.2.6. Trigger countdown.....	13
I.2.3. VI's de controlo de hardware externo	14
I.2.3.1. Filter countdown.vi	14
I.2.3.2. Programar ganhos.vi	14
I.2.3.3. Programar um ganho.vi	14
I.2.3.4. Programar switches.vi	15
I.2.4. VI's de visualização gráfica.....	17
I.2.4.1. Alteração de cor de fundo.vi.....	17
I.2.4.2. Alteração de grelhas	18
I.2.4.3. Build graphs.vi.....	18
I.2.4.4. Janela de gráficos.vi.....	20
I.2.5. VI's de escrita e leitura de ficheiros	22
I.2.5.1. Build info sinal.vi	22
I.2.5.2. Build info sinal cluster.vi.....	23
I.2.5.3. Build picos from table	24
I.2.5.4. Build picos table.vi.....	25

I.2.5.5. Display file.vi	26
I.2.5.6. Read cal file.vi	26
I.2.5.7. Read config.vi.....	27
I.2.5.8. Read data file.vi.....	27
I.2.5.9. Read peak file	28
I.2.5.10. Write cal file.vi.....	29
I.2.5.11. Write config.vi.....	30
I.2.5.12. Write data file.vi.....	30
I.2.5.13. Write peak file.vi.....	31
I.2.6. VI's de manipulação de sinais.....	32
I.2.6.1. Alteração info sinal.vi	32
I.2.6.2. Display info sinal.vi.....	33
I.2.6.3. Marcação de registos.vi.....	33
I.2.7. VI's de calibração de sensores	34
I.2.7.1. Avaliar calibração.vi	34
I.2.7.2. Calibração de sensores.vi	35
I.2.8. VI's de análise.....	35
I.2.8.1. Calcula picos.vi	35
I.2.8.2. Derivada média móvel.vi	36
I.2.8.3. Detecção de picos.vi	37
I.2.8.4. Painel de análise.vi	38
I.2.8.5. Resposta em frequência de filtros IIR.vi.....	39
I.2.8.6. Ver picos.vi.....	41
I.2.9. VI's de cálculo diverso	42
I.2.9.1. Calc velocidade.vi	42
I.2.9.2. Converter para binário.vi.....	42
I.2.9.3. Escalar ou reescalar vectores.vi	43
I.2.10. VI's de impressão.....	44
I.2.10.1. Imprimir calibração.vi.....	45
I.2.10.2. Imprimir gráficos.vi	46
I.2.10.3. Painel impressão cal.vi.....	46
I.2.10.4. Painel de impressão gráficos.vi.....	47
I.2.11. O painel principal: Main.vi.....	48
II. Especificações técnicas	49

II.1. Características técnicas da placa de aquisição LAB-PC+.....	50
II.2. Esquemático da placa de condicionamento tipo “piezoelétricos”	55
II.3. Esquemático da placa de condicionamento tipo “extensómetros”	56
II.4. Características técnicas dos sensores	57
II.4.1. Sensores de força piezoelétricos	57
II.4.2. Sensores de força extensiométricos	58
III. Fotografias	62
IV. Índice de tabelas.....	66

I. Descrição dos módulos de software desenvolvidos

Esta secção contém uma descrição das subrotinas ou VI's (Virtual Instruments) que foram desenvolvidos neste trabalho e que se encontram na livreria "Project.lib", tendo como objectivo constituir uma referência para futuros desenvolvimentos por parte do autor ou de terceiros.

As VI's apresentadas usam um leque variado de outras VI's que fazem parte integrante do LABVIEW Base Package e LABVIEW Advanced Analysis Toolpack. Estas não são descritas neste apêndice, para mais informação deverão ser consultados os manuais do LABVIEW e/ou LABVIEW Advanced Analysis Toolpack.

Grande parte das VI's servem como meras subrotinas, não sendo o seu painel visível durante a sua execução.

Nesta secção utilizar-se à a seguinte convenção:

- ***Nomes de definições de tipo, controlos ou indicadores apresentam-se em bold italic;***
- **Nomes de ficheiros ou de VI's apresentam-se em bold;**
- *Tipos, como cluster, double, text, apresentam-se em itálico e na língua inglesa.(Alguns termos técnicos, relacionados com o hardware, são também apresentados em itálico)*

É de notar que alguns controlos e indicadores de VI's servem como variáveis de entrada e saídas quando as VI's são chamadas por outras VI's.

I.1. Estruturas de dados

I.1.1. Definição de tipos – type data info.vi

A definição de tipos permite a partir de uma definição inicial de um controlo criar noutras VI's controlos do mesmo tipo. Uma alteração ao tipo produz uma alteração automática dos controlos baseados nessa definição de tipo. Assim, este mecanismo é particularmente útil para estruturas de dados complexas (como clusters), utilizadas em múltiplas VI's, e que sejam susceptíveis de serem alteradas frequentemente.

Neste momento, o único tipo de dados definido é o do cluster de informação sobre sinais ***data info type***, definida em ***type data info.vi***. Este tipo de estrutura repete-se em várias VI's e é susceptível de sofrer alterações em futuras versões do software.

O tipo ***data info type*** é um *cluster* com a seguinte estrutura:

Tipo <i>data info type</i> – Elementos do <i>cluster</i>		
Nome	<i>Tipo</i>	Descrição
<i>Data/hora</i>	<i>string</i>	Data e hora de aquisição do sinal
<i>Velocidade</i>	<i>double</i>	Velocidade de costura na aquisição

Tipo data info type – Elementos do cluster		
Nome	Tipo	Descrição
Ganho placa	<i>unsigned byte</i>	Ganho configurado na placa LAB-PC+ aquando da aquisição
Canal	<i>unsigned byte</i>	Canal amostrado
Nome	<i>string</i>	Nome do sinal ou do canal amostrado
m	<i>double</i>	Parâmetro m de calibração
b	<i>double</i>	Parâmetro b de calibração
Freq. amostragem	<i>double</i>	Frequência de amostragem do sinal
Tipo de sinal	<i>unsigned byte</i>	Tipo de sinal (tempo, frequência, derivada, etc.)
Janela	<i>unsigned byte</i>	Tipo de janela utilizada para cálculo do espectro (caso de sinais no domínio das frequências)
Comprimento	<i>unsigned long</i>	Comprimento de registo: Número de amostras contidas no sinal
Ganho externo	<i>double</i>	Ganho externo configurado aquando da aquisição
Zero placa	<i>word</i>	Parâmetro zero placa da calibração

Tabela 1: Elementos do cluster que forma o tipo *data type info*

I.1.2. Variáveis globais

A utilização de variáveis globais tornou-se necessária dado a constante utilização de dois tipos de informação: Sinais e informação de configuração.

Assim, foram criadas duas VI's com variáveis globais, que se passam a descrever.

I.1.2.1. Variáveis globais de configuração: *globals config.vi*

A configuração do software utiliza as variáveis listadas na tabela que se segue. Embora os seus nomes não coincidam, por razões várias, com os nomes que se utilizam no painel de configuração, existe uma correspondência entre os controlos aí presentes e as variáveis globais, que é também apresentada na tabela.

Variáveis globais de configuração			
Nome	Tipo	Descrição	Correspondência com painel de configuração
slot	<i>word</i>	Nº de device da placa de aquisição, configurável com	Igual a Device

Variáveis globais de configuração			
Nome	Tipo	Descrição	Correspondência com painel de configuração
		“NI-Daq Configuration Utility”	
gain	<i>unsigned byte</i>	Ganho da placa de aquisição	Corresponde a Ganho placa
dir trabalho	<i>file path</i>	Directório ou pasta de trabalho	igual a Pasta de trabalho
record length	<i>unsigned long</i>	Nº de amostras de uma aquisição	Igual a Comprimento de registo
sample frequency	<i>double</i>	Frequência de amostragem	Igual a Frequência de amostragem/canal
channels to sample	<i>string</i>	String com lista de canais a amostrar	Corresponde a Canais a amostrar
contagem/tempo	<i>boolean</i>	Escolha de preparação de tempo por tempo ou contagem de pontos	Corresponde a Preparação da aquisição por:Pontos/Tempo
temporização	<i>unsigned word</i>	Tempo ou nº de pontos antes de aquisição	Igual a Valor (de tempo ou pontos)
trigger timeout	<i>single</i>	Timeout de aquisição	Igual a Timeout da aquisição
config não salva	<i>boolean</i>	Sinalização de que configuração não foi gravada depois de uma alteração	-
sinais não salvos	<i>boolean</i>	Não utilizado nesta versão do software	-
config dos canais	<i>array de clusters</i>	Array com clusters de configuração dos canais. Um cluster para cada canal. Cluster definido na próxima tabela	Igual a Configuração dos canais

Tabela 2: Variáveis globais de configuração

O cluster de configuração dos canais contém os seguintes elementos:

Elementos do cluster de configuração de um canal			
Nome	Tipo	Descrição	Correspondência com painel de configuração
<i>nome</i>	<i>string</i>	Nome atribuído ao canal	Igual a Nome
<i>m</i>	<i>double</i>	Parâmetro <i>m</i> da calibração	Igual a <i>m</i>
<i>b</i>	<i>double</i>	Parâmetro <i>b</i> da calibração	Igual a <i>b</i>
zero Placa	<i>word</i>	Zero ajustado na placa externa	Igual a Ajuste de zero na placa
ganho ext	<i>double</i>	Ganho na placa externa	Igual a Ganho
placa	<i>unsigned byte</i>	Endereço de placa	igual a Placa
saída	<i>unsigned byte</i>	Endereço da saída na placa	igual a Saída
Tipo	<i>unsigned word enum {Piezoeléctricos, Extensómetros}</i>	Tipo de placa de condicionamento	igual a Tipo

Tabela 3: Cluster de configuração de um canal

I.1.2.2. Variáveis globais de dados (sinais): *globals data.vi*

Os sinais que se encontram nos bancos de registo de entrada e permanente são mantidos neste painel de variáveis globais.

Cada um dos bancos de registo está dividido em dois controlos (duas variáveis):

- Um controlo contém a informação sobre cada sinal (***Data info in*** e ***Data info perm***);
- O segundo controlo contém os vectores (*arrays*) de pontos, ou seja, os sinais propriamente ditos (***Data arrays in*** e ***Data arrays perm***).

O controlo respeitante à informação sobre sinais é um *array* de *clusters* do tipo que foi definido anteriormente (tipo ***data info type***, ver I.1.1). Cada banco de registos possui um *array* destes *clusters*, com informação sobre cada sinal aí contido (***Data info in*** e ***Data info perm***). O software mantém o *array* de sinais sincronizado com o *array* de *clusters* de informação.

O controlo que contém os sinais é um *array* de *clusters*. É necessário tecer aqui algumas considerações:

- Um sinal é um *array* de amostras (a amostra tem o tipo *double*). A agregação de vários sinais obriga assim a construir uma matriz, com um sinal em cada linha.

- Uma definição alternativa para esta variável será introduzir um sinal (um *array*, portanto) num *cluster*, do qual o sinal é o único elemento. Para guardar múltiplos sinais, é construído um *array* destes *clusters*.

Esta última definição é vantajosa em muitas operações (especialmente na construção de gráficos) pela sua maior simplicidade, e foi por isso adoptada.

Assim, o painel de variáveis globais de dados inclui para os sinais dois *arrays* de *clusters*: **Data arrays perm** e **Data arrays in**.

I.2. Subrotinas ou VI's (Virtual Instruments)

I.2.1. VI's de configuração

I.2.1.1. Calculate fs.vi

Descrição: Ajusta o valor pretendido para a frequência de amostragem de cada canal conforme a limitação da placa LAB-PC+, conforme o número de canais a amostrar e calcula a frequência real que se pode obter com a resolução dos contadores da placa e configuração de aquisição actual.

Esta VI faz a inicialização do clock da placa para obter a frequência real. Caso haja erro nesta inicialização, este retorna em "error out" e a frequência ajustada é -1.

A frequência ajustada retornada é por canal, mas a frequência máxima permitida refere-se ao "channel clock".

A frequência máxima permitida é ajustada conforme o ganho utilizado. Ela é de 83,33 KHz para um ganho de 1, 20 KHz para um ganho de 100 e 65 KHz para os restantes ganhos.

Esta VI é utilizada no painel de configuração para ajustar a frequência de amostragem/canal ao valor mais próximo possível daquele que é pretendido pelo utilizador.

Painel visível em execução: Não.

Variáveis de entrada "calculate fs.vi"		
Nome	Tipo	Descrição
slot	<i>unsigned byte</i>	Nº de device da placa para a qual se ajusta a frequência de amostragem
canais a adquirir	<i>string</i>	lista de canais a amostrar
frequência pretendida/canal	<i>double</i>	Frequência de amostragem por canal que se pretende utilizar
ganho	<i>unsigned byte</i>	Ganho a utilizar no amplificador de entrada da placa de aquisição

Tabela 4: Variáveis de entrada de "calculate fs.vi"

Saídas de “calculate fs.vi”		
Nome	Tipo	Descrição
frequência ajustada/canal	<i>double</i>	Frequência de amostragem por canal mais próxima da pretendida
máxima frequência permitida	<i>double</i>	Máxima frequência de amostragem de todos os canais permitida para o ganho pretendido
error out	<i>cluster de 3 elementos</i>	Cluster de erro do LABVIEW. Indica erro na programação da placa. A programação é feita para obter dos drivers a frequência de amostragem ajustada.

Tabela 5: Saídas de “calculate fs.vi”

I.2.1.2. Check config canais.vi

Descrição: Esta VI verifica se uma determinada configuração para os canais está correcta. As seguintes condições levam à indicação de erro:

- Canais configurados com endereço de placa 0: O endereço de placa 0 é reservado para operações de I/O (desactivação de sinal de leitura), não devendo ser utilizado.
- Canais com endereço externo de placa/saída iguais;
- Canais configurados com uma saída de placa que não existe (saída 2 numa placa de piezoeléctricos).

Retorna erro numa variável do tipo *boolean* e activa uma mensagem de erro em caso de alguma destas condições se verificar.

Painel visível em execução: Não.

Variáveis de entrada de “check config canais.vi”		
Nome	Tipo	Descrição
Configuração dos canais	<i>array de clusters de configuração de um canal (ver Tabela 3Tabela 3)</i>	Data e hora de aquisição do sinal

Tabela 6: Variáveis de entrada de “check config canais.vi”

Saídas de “check config canais.vi”		
Nome	Tipo	Descrição
erro	<i>boolean</i>	TRUE se configuração incorrecta

Tabela 7: Saídas de “check config canais.vi”

I.2.1.3. Configurar.vi

Descrição: Painel de configuração do software. Não tem variáveis de entrada nem saídas, pois funciona autonomamente, gravando a nova configuração ou alterando-a sem gravar, caso em que sinaliza a alteração da configuração na variável global **config não salva**.

Comentários: Esta VI começa por inicializar os seus controlos com os parâmetros de configuração actuais. Entra depois num ciclo *while* em que corrige constantemente os valores dos controlos para valores efectivamente utilizáveis (limitação de comprimento ed registo, ajuste de frequência de amostragem, etc.). Os valores de configuração permitidos são, no caso dos parâmetros respeitantes à placa de aquisição, valores para a placa LAB-PC+. O ciclo *while* acaba quando é premido um dos três botões.

Painel visível em execução: Sim.

I.2.2. VI's de aquisição

I.2.2.1. Adquirir.vi

Descrição: Adquire sinais conforme parâmetros definidos na configuração (a que acede através das variáveis globais de **globals config.vi**). Retorna **TRUE** em **cancelled?** se a operação não for cancelada pelo utilizador. Programa ganhos externos e comuta os switches das placas de condicionamento (passa-altos desligados). Chama **trigger countdown.vi** para a temporização/contagem de pontos anterior à sensibilização da placa ao sinal de sincronismo. Esta VI executa as aquisições sempre com *hardware trigger*, isto é, a placa de aquisição inicia a operação de aquisição num flanco ascendente do sinal de sincronismo. Os sinais são fornecidos numa matriz (valores digitais lidos pela placa), sendo também retornado um *array* de clusters de informação sobre os sinais. Não tem variáveis de entrada.

Painel visível em execução: Não.

Saídas de “adquirir.vi”		
Nome	Tipo	Descrição
data	<i>array 2D de word</i>	Sinais adquiridos, em foram de matriz. Valores digitais lidos pela placa, não escalados a valores de força.
error out	<i>error cluster</i>	Cluster de erro do LABVIEW. Indica erro na operação de aquisição.
cancelled?	<i>boolean</i>	TRUE se utilizador cancelar operação.
Acquired Data info	<i>array de clusters</i>	Informação sobre sinais adquiridos, com o

Saídas de “adquirir.vi”		
Nome	Tipo	Descrição
	<i>tipo data info type</i>	tipo definido para a informação sobre um sinal

Tabela 8: Saídas de “adquirir.vi”

I.2.2.2. Adquirir osciloscópio.vi

Descrição: VI para aquisição de sinais a ser utilizada no painel de osciloscópio. Ao contrário de **adquirir.vi**, os parâmetros de aquisição são passados à subVI em vez de serem utilizados os da configuração. Outra diferença reside no facto de não serem programados os ganhos externos nem comutados os filtros passa-alto das placas de condicionamento, e de poder ou não ser utilizado o sinal de sincronismo para iniciar as aquisições.

Painel visível em execução: Não.

Variáveis de entrada de “adquirir osciloscópio.vi”		
Nome	Tipo	Descrição
Canais a adquirir	<i>string</i>	Lista de canais a adquirir
Fs/canal	<i>double</i>	Frequência de amostragem/canal
Número de amostras/canal	<i>unsigned word</i>	Número de amostras a adquirir por canal
Hardware Trigger	<i>boolean</i>	Determina se é ou não utilizado o sinal de sincronismo para o início de aquisições
ganho placa	<i>unsigned byte</i>	Ganho para o amplificador de entrada da placa de aquisição

Tabela 9: Variáveis de entrada de “adquirir osciloscópio.vi”

Saídas de “adquirir osciloscópio.vi”		
Nome	Tipo	Descrição
error out	<i>error cluster</i>	Cluster de erro do LABVIEW. Indica erro na operação de aquisição
sinais	<i>array 2D de word</i>	Matriz com os sinais adquiridos

Tabela 10: Saídas de “adquirir osciloscópio.vi”

I.2.2.3. Aquis n valores.vi

Descrição: Vi que faz a leitura de n pontos espaçados de um determinado intervalo de tempo. Esta VI foi inicialmente utilizada na calibração de sensores. Testes mostraram que os valores fornecidos pela VI de aquisição utilizada nesta rotina (**AI Single**

Scan.VI) diferem ligeiramente dos valores obtidos numa aquisição de sinal através de **adquirir.vi** ou **adquirir osciloscópio.vi**, pelo que **aquis n valores.vi** foi substituída no painel de calibração de sensores por **adquirir osciloscópio.vi**. Garante-se assim que as leituras durante a calibração são efectuadas nas mesmas condições em que são efectuadas no osciloscópio ou na aquisição normal. Esta VI não é utilizada em nenhum painel do software.

Painel visível em execução: Não.

Variáveis de entrada de “aquis n valores.vi”		
Nome	Tipo	Descrição
Número de leituras	<i>word</i>	Número de leituras a efectuar
Intervalo entre amostras	<i>double</i>	Intervalo entre cada duas leituras, em segundos
Canal	<i>byte</i>	Canal da placa de aquisição em que devem ser efectuadas as leituras.

Tabela 11: Variáveis de entrada de “aquis n valores.vi”

Saídas de “aquis n valores.vi”		
Nome	Tipo	Descrição
Vector de valores adquiridos	<i>array de word</i>	Array de valores lidos (leituras digitais efectuadas pela placa)

Tabela 12: Saídas de “aquis n valores.vi”

I.2.2.4. Osciloscópio.vi

Descrição: Painel de osciloscópio (ver no documento principal: *III.8.O painel de osciloscópio*). Não tem variáveis de entrada nem saídas.

Comentários: Esta VI está constantemente a efectuar as seguintes operações:

- Correção de valores de parâmetros de aquisição, bem como fixação de limites mínimos e máximos de valor de frequência de amostragem conforme número de canais amostrados e comprimento de registo. O limite máximo depende do canal visualizado: A placa LAB-PC+ tem aquisição sequencial, por isso quanto maior for o índice do canal visualizados, menor será a frequência máxima possível . O limite mínimo é fixado de foram a evitar que uma aquisição ultrapasse um tempo de 5 segundos;
- Cálculo e indicação do tempo de aquisição conforme a configuração actual da aquisição neste painel;
- Programação de passa-altos e ganhos das placas de condicionamento;
- Aquisição.

A ligação dos filtros passa-alto ocorre sem temporização dos filtros para que seja possível uma rápida e contínua actualização do gráfico.

Caso seja premido algum dos botões **Configurar**, os parâmetros de aquisição configurados neste painel são armazenados nas respectivas variáveis globais de configuração.

Painel visível em execução: Sim.

I.2.2.5. Teste de saturação.vi

Descrição: Percorre os sinais obtidos numa aquisição para verificar se algum dos valores aí contidos está no limite da gama de codificação da placa (-2047 a 2047). Este facto pode significar que o sinal sai for a dos limites de codificação (satura o conversor analógico-digital), o que pode influenciar os sinais obtidos nos restantes canais. No caso dessa ocorrência, é visualizada uma mensagem de aviso.

Esta VI tem um indicador (**sinal saturado**) em que fica a indicação do primeiro canal em que foi detectada uma saturação. Este indicador não é utilizado e não está, por isso, ligado. Futuros utilizadores poderão rapidamente ligar o indicador no conector da VI e utilizar esta indicação.

Painel visível em execução: Não.

Variáveis de entrada de “teste de saturação.vi”		
Nome	Tipo	Descrição
data in	<i>array 2D de word</i>	Sinais adquiridos

Tabela 13: Variáveis de entrada de “teste de saturação.vi”

Saídas de “teste de saturação.vi”		
Nome	Tipo	Descrição
saturado	<i>boolean</i>	TRUE se detectar saturação

Tabela 14: Saídas de “teste de saturação.vi”

I.2.2.6. Trigger countdown

Descrição: Vi que faz uma contagem decrescente de pontos dados pela máquina de costura ou de tempo (conforme configuração do software), visualizando a mesma. Para a contagem de pontos, faz leituras o sinal de sincronismo (que deve ser ligado ao canal 0). Não tem variáveis de entrada, pois o tipo de contagem decrescente e o seu valor encontram-se nas respectivas variáveis globais.

Painel visível em execução: Sim.

Saídas de “trigger countdown.vi”		
Nome	Tipo	Descrição
<i>cancelled?</i>	<i>boolean</i>	TRUE se aquisição for cancelada pelo utilizador. Permite à rotina de aquisição adquirir.vi não efectuar as restantes operações de aquisição.

Tabela 15: Saídas de “trigger countdown.vi”

I.2.3. VI's de controlo de hardware externo

I.2.3.1. Filter countdown.vi

Descrição: Executa e visualiza uma temporização de 10 segundos. É utilizada em programar switches.vi para temporização com objectivo de estabilizar sinal aquando da ligação de filtros passa-alto.

Não tem entradas nem saídas.

Painel visível em execução: Sim.

I.2.3.2. Programar ganhos.vi

Descrição: Programa os ganhos de todos os canais em que vai ser efectuada uma aquisição (informação contida em variável global **channels to sample**). Utiliza a VI **programar um ganho.vi** para efectuar esta operação para cada canal. Não tem variáveis de entrada.

Esta VI é utilizada na aquisição normal no painel principal. Os ganhos passados a **programar um ganho.vi** são obtidos da variável global de configuração de canais.

Painel visível em execução: Não.

Saídas de “programar ganhos.vi”		
Nome	Tipo	Descrição
<i>error out</i>	<i>error cluster</i>	Cluster de erro do LABVIEW. Indica erro nas operações I/O paralelo necessárias para comunicação com as placas de condicionamento

Tabela 16: Saídas de “programar ganhos”

I.2.3.3. Programar um ganho.vi

Descrição: Programa um canal com o ganho que lhe é passado como variável de entrada. Esta VI é utilizada pelo painel de osciloscópio e pela rotina **programar ganhos.vi**. Os endereços das placas/saídas do condicionador externo para cada canal são fornecidas pela variável global de configuração dos canais (**config dos canais**).

A programação do ganho de um canal segue os seguintes passos (conforme descrito no documento principal em *II.3.3.Interface digital no barramento - endereçamento e comunicação com placas externas*):

- Configuração das portas da placa LAB-PC+ para escrita de dados
- Escrita de ganho na porta 1 (porta de dados);
- Escrita de endereço de placa e código de função na porta 0 (porta de endereços);
- Escrita de 0 nos 4 bits de endereço de placa para desactivar sinal de leitura de dados.

Salienta-se mais uma vez aqui a necessidade de manter os 4 bits correspondentes ao código de função, pois a não o fazer, e devido ao atraso no sinal de leitura que chega da PAL, os dados que se encontram no porto de dados passariam para outro canal antes de haver desactivação definitiva do sinal de leitura.

Painel visível em execução: Não.

Variáveis de entrada de “programar um ganho.vi”		
Nome	Tipo	Descrição
<i>canal</i>	<i>unsigned byte</i>	Canal para o qual deve ser programado o ganho.
<i>ganho</i>	<i>double</i>	Ganho a programar. Limites: 0 a 9,96 em passo de 10/256. O ganho máximo é de exactamente $10 \cdot 255 / 256$.

Tabela 17: Variáveis de entrada de “programar um ganho.vi”

Saídas de “programar um ganho.vi”		
Nome	Tipo	Descrição
<i>error out</i>	<i>error cluster</i>	Cluster de erro do LABVIEW. Indica erro nas operações I/O paralelo necessárias para comunicação com as placas de condicionamento

Tabela 18: Saídas de “programar um ganho.vi”

I.2.3.4. Programar switches.vi

Descrição: Faz a programação dos *multiplexers analógicos* existentes nas placas, que determinam os seguintes modos de funcionamento das placas:

- Ligação ou bypass a filtros passa-alto (placas tipo extensómetros e tipo piezoeléctricos);
- Escolha de ligação à saída de detector de pico ou da saída normal do condicionador (placa tipo piezoeléctricos);
- Curto-circuito da saída do detector de pico e amplificador de carga para fazer reset ao mesmo (placa tipo piezoeléctricos).

Esta VI tem dois modos de execução:

- Um primeiro utilizado em aquisições normais, em que se faz a escolha da ligação ou bypass ao filtro passa-alto;
- Um segundo modo utilizado especificamente para calibração de sensores, em que o funcionamento da placa de condicionamento tipo piezoeléctrico difere do da placa tipo extensómetros.

A programação dos *multiplexers* (*switches*) não é feita individualmente para cada canal, mas em bloco, para todos os canais. Esta operação é igual à da programação de um ganho, só que é indicado um endereço de saída da placa não existente (2 no caso de piezoeléctricos, 4 no caso de extensómetros), é indicado um código significando “função de programação de switches”. A palavra de dados contém em vez de um valor de ganho o estado pretendido para os switches.

Os seguintes passos constituem a programação de um estado dos switches:

- Configuração das portas da placa LAB-PC+ para escrita de dados;
- Escrita da palavra de controlo com estado dos switches no porto de dados;
- Escrita da palavra de endereço/função da placa com função de programação de switches;
- Escrita de 0 nos 4 bits de endereço de placa para desactivar sinal de leitura de dados.

Os dois modos de execução desta VI constituem-se pelos seguintes passos:

Modo de aquisição normal:

- Construção de palavras de controlo de *switches* e de endereço/função para cada placa diferente encontrada na configuração de canais. A palavra de controlo tem em atenção se deve ou não ser utilizado filtro passa-alto;
- Programação de todas as placas.

Modo de calibração – Placa tipo extensómetros:

- A placa a que pertence o canal em que se efectua a calibração é programada com ou sem filtro passa-alto conforme parâmetro de entrada **Passa-alto**. Dependendo do mesmo parâmetro, pode haver lugar a uma temporização para estabilização do filtro passa-alto.

Modo de calibração – Placa tipo piezoeléctricos:

- Os detectores de pico são ligados à saídas da placa. As suas saídas e as dos amplificadores de carga são curto-circuitados para fazer reset. Conforme o parâmetro de entrada **Passa-alto**, são ligados ou desligados os filtros passa-alto, podendo haver lugar a uma temporização.
- Depois de um tempo de espera para descarga do detector de pico e temporização para estabilização dos filtros passa-alto, é aberto o curto-circuito à sua saída – o detector de pico fica pronto a funcionar.

Painel visível em execução: Não.

Variáveis de entrada de “programar switches.vi”		
Nome	Tipo	Descrição
Modo de calibração	<i>boolean</i>	Escolha de modo de calibração (TRUE) ou modo de aquisição normal
canal a calibrar	<i>unsigned byte</i>	Canal sobre o qual vai ser efectuada a calibração. Ignorado se Modo de calibração =FALSE
Passa-alto	<i>unsigned word enum</i> {S/filtro, Filtro s/Estabilização, Filtro c/estabilização}	Desligar filtro passa-alto (s/filtro), ligar filtro e não temporizar (Filtro s/estabilização), ligar filtro e temporizar (Filtro c/estabilização).

Tabela 19: Variáveis de entrada de “programar switches.vi”

Saídas de “programar switches.vi”		
Nome	Tipo	Descrição
error out	<i>error cluster</i>	Cluster de erro do LABVIEW. Indica erro nas operações I/O paralelo necessárias para comunicação com as placas de condicionamento

Tabela 20: Saídas de “programar switches.vi”

I.2.4. VI's de visualização gráfica

I.2.4.1. Alteração de cor de fundo.vi

Descrição: Esta VI funciona como uma caixa de diálogo para escolha de um valor de cor num controlo próprio, que será de pois utilizado para a cor de fundo do gráfico no menu principal.

Painel visível em execução: Sim.

Variáveis de entrada de “alteração de cor de fundo.vi”		
Nome	Tipo	Descrição
cor inicial	<i>long</i>	Valor inicial a dar ao controlo de cor (cor actual do gráfico)

Tabela 21: Variáveis de entrada de “alteração de cor de fundo.vi”

Saídas de “alteração de cor de fundo.vi”		
Nome	Tipo	Descrição
<i>cor escolhida</i>	<i>long</i>	Cor escolhida pelo utilizador

Tabela 22: Saídas de “alteração de cor de fundo.vi”

I.2.4.2. Alteração de grelhas

Descrição: Esta VI é semelhante a **alteração de cor de fundo.vi**, mas possuindo dois controlos de cor, destinados à escolha da cor de linhas para formar uma grelha no gráfico.

Painel visível em execução: Sim.

Variáveis de entrada de “alteração de grelhas.vi”		
Nome	Tipo	Descrição
<i>cor inicial x</i>	<i>long</i>	Cor inicial das linhas que demarcam valores do eixo dos xx
<i>cor inicial y</i>	<i>long</i>	Cor inicial das linhas que demarcam valores do eixo dos yy

Tabela 23: Variáveis de entrada de “alteração de grelhas.vi”

Saídas de “alteração de grelhas.vi”		
Nome	Tipo	Descrição
<i>cor escolhida x</i>	<i>long</i>	Cor escolhida pelo utilizador para as linhas que demarcam valores do eixo dos xx
<i>cor escolhida y</i>	<i>long</i>	Cor escolhida pelo utilizador para as linhas que demarcam valores do eixo dos yy

Tabela 24: Saídas de “alteração de grelhas.vi”

I.2.4.3. Build graphs.vi

Descrição: A Esta VI faz a construção de todos os dados necessários para a visualização de um gráfico, fornecendo o gráfico propriamente dito, a legenda e um título que descreve as escalas utilizadas. A construção do gráfico depende do tipo de escala seleccionada de momento, dos tipos dos sinais que se encontram nos registos, de se pretender fazer a visualização dos registos de entrada ou dos registos permanentes e, no caso de se fazer a visualização dos registos permanentes, da selecção destes registos.

O gráfico é fornecido como um *array* de clusters de gráfico. Cada cluster de gráfico contém os seguintes elementos:

Construção de um <i>cluster</i> de gráfico	
Elemento	Tipo
Array de pontos	<i>array de double</i>
Ponto inicial para escala x	<i>double</i>
Incremento para escala x	<i>double</i>

Tabela 25: Construção de um *cluster* de gráfico

Só no painel de calibração não é utilizado este tipo de construção de gráfico, mas são utilizados gráficos x-y.

A escolha dos registos que serão visualizados depende do tipo de escala x que está seleccionada e dos tipos dos sinais que se encontram nos registos. As seguintes tabelas mostram os tipos de sinais e como se relaciona a visualização de um determinado tipo de sinal com a escala seleccionada:

Tipo	Descrição
0	Sinais no tempo
1	Espectro de potências de um sinal
2	Módulos do espectro do sinal
3	Fase do espectro do sinal
4	Derivada de um sinal temporal
5	Vector de valores discretos

Tabela 26: Tipos de sinais

Escala x	Tipos de sinais visualizados
Tempo	0,4
Ângulo	0,4
Frequência	1,2,3
Frequência normalizada	1,2,3
Pontos discretos	5

Tabela 27: Visualização de sinais nas diversas escalas

Painel visível em execução: Não.

Variáveis de entrada de “build graphs.vi”		
Nome	Tipo	Descrição
permanentes	<i>boolean</i>	Escolha de visualização de registos permanentes (TRUE) ou de registos de entrada (FALSE)
sel_reg.perm	<i>array de long</i>	Lista de índices de registos permanentes seleccionados. Ignorado se permanentes =FALSE.
EscalaX	<i>unsigned word enum {Tempo, Ângulo, Freqüência, Freq.normalizada, Pontos discretos}</i>	Tipo de escala. Este controlo é um <i>enumerated ring control</i> , pelo que a ele deve ser ligado um controlo com exactamente os mesmos elementos na enumeração (caso do controlo de escala no painel principal e no painel de análise).

Tabela 28: Variáveis de entrada de “build graphs.vi”

Saídas de “build graphs.vi”		
Nome	Tipo	Descrição
gráfico	<i>array de clusters de gráfico</i>	Dados para visualização de gráfico. Formato do <i>cluster</i> conforme Tabela 25.
Plot names	<i>array de string</i>	Lista de nomes de sinais, para uso como legenda
Título	<i>string</i>	Título de gráfico: Descrição de unidades de escalas x e y.

Tabela 29: Saídas de “bulid graphs.vi”

I.2.4.4. Janela de gráficos.vi

Descrição: Permite abrir uma janela extra com um gráfico. São passados à VI os dados a visualizar (os sinais) bem como alguns atributos do gráfico. Assim, é possível criar uma cópia exacta de um determinado gráfico e mantê-la numa janela separada.

Não tem saídas.

Painel visível em execução: Sim.

Variáveis de entrada de “janela de gráficos.vi”		
Nome	Tipo	Descrição
gráfico	<i>array de clusters</i>	Dados para construção do gráfico. Formato

Variáveis de entrada de “janela de gráficos.vi”		
Nome	Tipo	Descrição
	<i>de gráfico</i>	do cluster conforme Tabela 25.
graph control	<i>cluster de atributos de gráfico (10 elementos)</i>	<i>Ver próxima tabela</i>

Tabela 30: Variáveis de entrada de “alteração de cor de fundo.vi”

Atributos de gráfico utilizados		
Nome	Tipo	Descrição
Legend Visible	<i>boolean</i>	Determina se a legenda está visível
Palette visible	<i>boolean</i>	Determina se a paleta de controlo do gráfico está visível
Plot Area Colors	<i>cluster de 2 elementos</i>	<i>Cluster de cores da área do gráfico</i>
Grid Colors	<i>cluster de 2 elementos</i>	<i>Cluster de cores das linhas de grelha</i>
X Scale Info	<i>cluster de 8 elementos</i>	<i>Cluster de informação sobre escala x</i>
Y Scale Info	<i>cluster de 8 elementos</i>	<i>Cluster de informação sobre escala y</i>
Cursor List	<i>array de clusters de 10 elementos</i>	<i>Array de clusters com informação sobre os cursores</i>
plot names array	<i>array de string</i>	Nomes dos gráficos para legenda
plot info array	<i>array de clusters de 8 elementos</i>	<i>Array de clusters com informação sobre cada traçado no gráfico</i>
cursor info array	<i>array de clusters de 10 elementos</i>	<i>Array de clusters com informação sobre cursores.</i>

Tabela 31: Atributos de gráfico utilizados na janela de gráficos

Comentários: A utilização de uma segunda lista de cursores, construída explicitamente, (como é visível em **main.vi**), além da utilização do atributo **Cursor List** por via directa, tornou-se necessário dado a utilização deste último não ter funcionado correctamente. Este facto atribui-se a um possível erro do LABVIEW.

Para mais informações sobre os atributos de gráficos, deve ser consultada a documentação do LABVIEW.

I.2.5. VI's de escrita e leitura de ficheiros

As VI's descritas nesta secção relacionam-se com a escrita e leitura de ficheiros de vários tipos, bem como com a manipulação de dados necessária para estas operações.

Os formatos de ficheiros não são descritos, podendo ser consultados no documento principal em *III.11. Formatos de ficheiros*.

Todos os ficheiros utilizados são ficheiros de texto, por simplicidade e mais fácil exportação/importação de dados. É necessário ter em atenção que qualquer edição a estes ficheiros deve respeitar o formato definido, caso contrário os dados não podem ser recuperados.

Os campos contidos nos ficheiros são sujeitos a uma conversão de texto para o formato do campo destinatário na sua leitura. Se o utilizador inadvertidamente editar o ficheiro deixando o campo incorrecto, o valor resultante será nulo. Assim, por exemplo, o valor 0,345a será convertido para 0.

I.2.5.1. Build info sinal.vi

Descrição: Constrói informação sobre sinal para um único sinal em formato texto a partir de *cluster* tipo **data info type**. É utilizada para escrita de ficheiros de sinais (**write data file.vi**). O texto gerado por esta VI inclui o cabeçalho com o número do sinal (ver no documento principal: *III.11. Formatos de ficheiros*).

Em futuras versões de ficheiros, é necessário acrescentar eventuais novos parâmetros a serem gravados.

Painel visível em execução: Não.

Variáveis de entrada de "build info sinal.vi"		
Nome	Tipo	Descrição
<i>info reg</i>	<i>data info type</i>	Cluster de informação sobre um sinal
<i>numero sinal</i>	<i>unsigned word</i>	Número do sinal a ser aplicado no cabeçalho da informação sobre o sinal.

Tabela 32: Variáveis de entrada de "build info sinal.vi"

Saídas de "build info sinal.vi"		
Nome	Tipo	Descrição
<i>info text</i>	<i>string</i>	Texto com informação sobre sinal a escrever no ficheiro de sinais

Tabela 33: Saídas de "build info sinal.vi"

I.2.5.2. Build info sinal cluster.vi

Descrição: Esta VI constrói, a partir de um *array* de linhas de texto lido de um ficheiro de sinais, um *array de clusters* de informação sobre sinais (tipo ***data info type***), sendo utilizada em read data file.vi.

Esta construção tem em atenção a versão do ficheiro que foi lido. Nesta VI existe uma estrutura *case* que contém casos para todas as versões de ficheiros existentes (de momento são duas). Futuras versões podem ser incluídas rapidamente, copiando o caso da versão anterior e fazendo as alterações necessárias.

A VI começa por retirar das linhas que lhe são fornecidas aquelas que contêm somente comentários, reconhecidas por se iniciarem com um asterisco (*). Aqui é necessário ter em atenção que campos tais como o nome de um sinal podem também começar com esse caracter, sendo estes casos verificados.

De seguida é feita uma comparação do número de linhas que sobrou desta “filtragem” com o número de linhas que deveria existir para a versão do ficheiro e tendo em conta que são múltiplos sinais (divisão inteira do número de linhas pelo número de elementos da informação sobre um sinal para a versão em questão tem que ter resto=0). Também aqui é fácil acrescentar um caso para novas versões.

Finalmente é feita a conversão e é construído o *array*.

Painel visível em execução: Não.

Variáveis de entrada de “build info sinal cluster.vi”		
Nome	Tipo	Descrição
<i>info text array in</i>	<i>array de string</i>	<i>Array de strings</i> que contém em cada elemento uma linha da secção de informação sobre sinais contida no ficheiro.
<i>file_version</i>	<i>unsigned byte</i>	Versão do ficheiro

Tabela 34: Variáveis de entrada de “build info sinal cluster.vi”

Saídas de “build info sinal cluster.vi”		
Nome	Tipo	Descrição
<i>info clusters</i>	<i>array de clusters data info type</i>	Informação sobre sinais construída
<i>n_sinais</i>	<i>unsigned word</i>	Número de <i>clusters</i> de informação construídos
<i>erro</i>	<i>unsigned byte</i>	0: Sem erro, 1:erro (Número de linhas incorrecto)

Tabela 35: Saídas de “build info sinal cluster.vi”

I.2.5.3. Build picos from table

Descrição: O painel de análise, na seu subpainel de detecção de picos, utiliza um *array* de clusters para guardar os valores de picos, e que tem a seguinte estrutura (cada cluster descreve um período do sinal, em que podem ter sido analisados vários picos):

Elementos do <i>cluster</i> que contém os picos de um período de sinal analisado		
Nome	Tipo	Descrição
nome	<i>string</i>	Nome do sinal em que foi(foram) calculado(s) o(s) pico(s) (normalmente com índice do período em questão)
velocidade	<i>double</i>	Velocidade de costura na aquisição do sinal analisado
picos	<i>array de cluster de três elementos</i>	Picos detectados nesse período do sinal

Tabela 36: Cluster que contém picos detectados num período do sinal

Cada período do sinal é descrito por um *cluster* deste tipo. O painel de análise mantém em memória um *array* destes *clusters*. Em cada período do sinal podem ter sido analisados vários picos, assim sendo cada *cluster* deste tipo contém um *array* de *clusters* que descrevem um pico individualmente, e que têm a seguinte estrutura:

Cluster de pico individual		
Nome	Tipo	Descrição
Início	<i>double</i>	Início do pico, em milisegundos
Fim	<i>double</i>	Fim do pico, em milisegundos (= -1 para picos absolutos)
Valor	<i>double</i>	Valor do pico, na unidade do sinal original

Tabela 37: Cluster para pico individual

A VI **build picos from table.vi** transforma uma tabela (um tipo de controlo do LABVIEW que é do tipo *array 2D de string*) num *array* de clusters que descrevem os picos da foram que se apresentou.

Esta VI é utilizada para reconstruir os *clusters* de picos após alterações efectuadas pelo utilizador na tabela, e para reconstruir essa estrutura na sequência da leitura de um ficheiro de picos.

Painel visível em execução: Não.

Variáveis de entrada de “build picos from table.vi”		
Nome	Tipo	Descrição
Table in	array 2D de string	Tabela de picos

Tabela 38: Variáveis de entrada de “build picos from table.vi”

Saídas de “build picos from table.vi”		
Nome	Tipo	Descrição
Picos out	array de clusters do tipo que se definiu nas tabelas 36 e 37)	Estrutura de dados utilizada para guardar resultado da análise de picos.

Tabela 39: Saídas de “build picos from table.vi”

A utilização desta estrutura, algo complexa, para guardar os resultados da detecção de picos tornou mais simples a manipulação desta informação.

I.2.5.4. Build picos table.vi

Descrição: Esta VI tem a função inversa da que cumpre build picos from table, transformando as estruturas de picos numa tabela de picos.

Painel visível em execução: Não.

Variáveis de entrada de “build picos table.vi”		
Nome	Tipo	Descrição
Picos in	array de clusters do tipo que se definiu (ver tabelas 36 e 37)	Estrutura de dados utilizada para guardar resultado da análise de picos.

Tabela 40: Variáveis de entrada de “build picos table.vi”

Saídas de “build picos table.vi”		
Nome	Tipo	Descrição
Tabela de picos	array 2D de string	Tabela de picos

Tabela 41: Saídas de “build picos table.vi”

I.2.5.5. Display file.vi

Descrição: Painel para visualização do caminho e conteúdo dum ficheiro. Utilizado pela VI **read data file.vi** para previsão de um ficheiro. Não tem saídas.

Painel visível em execução: Sim.

Variáveis de entrada de “display file.vi”		
Nome	Tipo	Descrição
<i>file text</i>	<i>string</i>	Conteúdo do ficheiro
<i>path info</i>	<i>file path</i>	Caminho do ficheiro

Tabela 42: Variáveis de entrada de “display file.vi”

I.2.5.6. Read cal file.vi

Descrição: Faz a leitura de um ficheiro de calibração, de extensão “.cal (ver no documento principal: *III.11. Formatos de ficheiros*). Confirma a correcta leitura do ficheiro ou faz um aviso de erro. Visualiza as condições em que a calibração foi efectuada. Não tem entradas.

Painel visível em execução: Não.

Saídas de “read cal file.vi”		
Nome	Tipo	Descrição
<i>Pesos</i>	<i>array de double</i>	Array com pesos aplicados para calibração
<i>Leituras</i>	<i>array de word</i>	Leituras digitais, para os valores de peso de Pesos
<i>erro</i>	<i>long</i>	Código de erro standard do LABVIEW. Um erro no formato do ficheiro não se reflecte aqui, mas na visualização de uma mensagem de erro
<i>canal</i>	<i>unsigned byte</i>	Nº do canal em que estava a ser efectuada a calibração aquando da escrita do ficheiro
<i>Zero placa</i>	<i>word</i>	Valor disponível aquando da escrita do ficheiro de zero ajustado no canal da placa externa

Tabela 43: Saídas de “read cal file.vi”

I.2.5.7. Read config.vi

Descrição: Faz a leitura do ficheiro de configuração (“**sewtest.cfg**”). Caso haja erro de leitura ou formato incorrecto do ficheiro, é sugerida a criação de um ficheiro novo. Não tem entradas. Retorna código de erro standard do Labview à excepção de:

- Código 4: Ficheiro de configuração corrupto, novo ficheiro não foi criado
- Código -1: Novo ficheiro de configuração foi criado, mas não lido

A leitura da configuração é feita de modo a armazenar os valores lidos directamente nas variáveis globais de configuração.

Painel visível em execução: Não.

Saídas de “read config.vi”		
Nome	Tipo	Descrição
<i>erro</i>	<i>long</i>	Erro de leitura, conforme descrito

Tabela 44: Saídas de “read config.vi”

I.2.5.8. Read data file.vi

Descrição: Esta VI faz a leitura de ficheiros de sinais (extensão “**.sew**”). A sua execução processa-se em várias fases:

- A escolha do ficheiro através de uma caixa de diálogo standard do sistema operativo;
- A verificação do tipo de ficheiro (completo, simples ou estranho), e a adaptação do painel a esse tipo. Nesta fase a parte inicial do ficheiro pode ser visualizada em **display file.vi**, no caso de ficheiros completos ou simples aparecem no painel os comentários que foram inseridos pelo utilizador quando gravou o ficheiro;
- A leitura do ficheiro depois de confirmada pelo utilizador.

Por sua vez, a leitura do ficheiro processa-se nos seguintes passos:

- A determinação da versão do ficheiro (em ficheiros completos) tinha já sido efectuada quando da fase de escolha de um ficheiro;
- É lida a informação sobre sinais e é convertida num *array* de *clusters* do tipo **data info type** através de **build info sinal cluster.vi** (caso de ficheiros completos);
- São lidos e convertidos os *arrays* de valores que constituem os sinais;
- É verificada a correcção dos dados.

No caso de não se tratar de um ficheiro completo, são preenchidos provisoriamente alguns campos da informação sobre sinais e o utilizador é avisado do facto.

Esta VI não tem entradas.

Painel visível em execução: Sim.

Saídas de “read data file.vi”		
Nome	Tipo	Descrição
Info array	<i>array de clusters</i> <i>tipo data info</i> <i>type</i>	Array com informação sobre sinais lidos
Data array	<i>array 2D de</i> <i>double</i>	Sinais lidos do ficheiro
erro	<i>long</i>	Código de erro standard do LABVIEW, à excepção de código 4: Erro de formato
cancelled	<i>boolean</i>	Indica se leitura de ficheiro foi cancelada (TRUE)

Tabela 45: Saídas de “read data file.vi”

I.2.5.9. Read peak file

Descrição: Esta VI faz a leitura de um ficheiro de picos (extensão “.pco”). O ficheiro de picos é um ficheiro de texto (no formato descrito no documento principal em *III.11. Formatos de ficheiros*), em que existe, à semelhança de ficheiros de sinais, uma secção de comentários e uma secção com dados, neste caso em vez de sinais valores de picos. Cada linha de dados tem o seguinte formato:

Nome	Velocidade	Início	Fim	Valor	Início	Fim	Valor	...
------	------------	--------	-----	-------	--------	-----	-------	-----

Figura 1: Formato de linha de dados de picos

Cada uma destas linhas corresponde à análise de um período do sinal. O nome do sinal aparece em primeiro lugar, com um sufixo que corresponde ao número do período descrito.

A seguir vêm os picos propriamente ditos, com indicação de instante de início e fim, e valor.

Os campos são separados por caracteres TAB.

A VI **read peak file.vi** faz a leitura destes ficheiros. À semelhança da abertura de ficheiros de sinais, a abertura de um ficheiro de picos processa-se nas seguintes fases:

- A escolha do ficheiro através de uma caixa de diálogo standard do sistema operativo;
- A visualização dos comentários contidos no ficheiro escolhido;
- A leitura do ficheiro depois de confirmada pelo utilizador.

Caso o ficheiro escolhido pelo utilizador não tiver o formato de um ficheiro de picos, é visualizada uma mensagem de erro.

A VI não tem entradas.

Painel visível em execução: Sim.

Saídas de “read peak file.vi”		
Nome	Tipo	Descrição
<i>picos</i>	<i>array 2D de string</i>	Tabela de picos em formato matriz de <i>string</i> . Pode ser convertida para uma estrutura de dados sobre picos por build picos from table.vi
<i>cancelled</i>	<i>boolean</i>	Indica se a operação foi cancelada (TRUE)
<i>erro</i>	<i>long</i>	Código de erro standard do LABVIEW

Tabela 46: Saídas de “read peak file.vi”

I.2.5.10. Write cal file.vi

Descrição: Faz a escrita de um ficheiro de calibração (extensão “.cal”). Ver também **read cal file.vi**.

Painel visível em execução: Não.

Variáveis de entrada de “write cal file.vi”		
Nome	Tipo	Descrição
<i>Pesos</i>	<i>array de double</i>	Array com pesos aplicados para calibração
<i>Leituras</i>	<i>array de word</i>	Leituras digitais, para os valores de peso de <i>Pesos</i>
<i>canal</i>	<i>unsigned byte</i>	Nº do canal em que estava a ser efectuada a calibração
<i>Zero placa</i>	<i>word</i>	Valor de zero ajustado no canal da placa externa

Tabela 47: Variáveis de entrada de “write cal file.vi”

Saídas de “write cal file.vi”		
Nome	Tipo	Descrição
<i>erro</i>	<i>long</i>	Código de erro standard do LABVIEW.

Tabela 48: Saídas de “write cal file.vi”

I.2.5.11. Write config.vi

Descrição: Faz a escrita do ficheiro de configuração (“**sewtest.cfg**”). Os valores da configuração são lidos directamente das respectivas variáveis globais. É também possível criar um ficheiro de configuração novo, com valores de defeito. Caso exista já um ficheiro de configuração, são mantidos neste os comentários iniciais (linhas começadas por asterisco até ao primeiro parâmetro).

Painel visível em execução: Não.

Variáveis de entrada de “write config.vi”		
Nome	Tipo	Descrição
<i>write default file</i>	<i>boolean</i>	Quando TRUE, a VI cria um ficheiro de configuração novo, com valores de defeito.

Tabela 49: Variáveis de entrada de “write config.vi”

Saídas de “write config.vi”		
Nome	Tipo	Descrição
<i>erro</i>	<i>long</i>	Código de erro standard do LABVIEW.

Tabela 50: Saídas de “write config.vi”

I.2.5.12. Write data file.vi

Descrição: Faz a gravação de um ficheiro de sinais. Ver também **read data file.vi**.

O ficheiro pode ser um ficheiro simples ou completo, e pode conter todos os sinais ou somente os sinais que estão seleccionados no controlo do banco de registos permanente. Esta opção é feita no painel da própria VI. Após confirmação da gravação do ficheiro o seu nome é escolhido através de uma caixa de diálogo standard do sistema operativo.

Para a VI receber informação sobre os sinais que estão seleccionados, é-lhe fornecido o controlo de registos permanentes que existe no painel principal **main.vi**. Este controlo é um *array* de clusters com o seguinte formato:

Cluster de controlo de banco de registos		
Nome	Tipo	Descrição
<i>nome</i>	<i>string</i>	Nome do sinal

Cluster de controlo de banco de registos		
Nome	Tipo	Descrição
<i>tipo</i>	<i>unsigned long</i>	Tipo do sinal
<i>selected</i>	<i>boolean</i>	Sinal seleccionado quando TRUE

Tabela 51: Cluster de controlo de banco de registos

O formato do ficheiro está descrito no documento principal em *III.11. Formatos de ficheiros*. A conversão de clusters de informação sobre sinal (*data info type*) em texto é feita por **build info sinal.vi**.

Painel visível em execução: Sim.

Variáveis de entrada de “write data file.vi”		
Nome	Tipo	Descrição
<i>reg control array</i>	<i>array de clusters do tipo definida na Tabela 51</i>	Array com clusters de controlo do banco de registos permanente, utilizado para determinar quais os registos seleccionados.

Tabela 52: Variáveis de entrada de “write data file.vi”

Saídas de “write data file.vi”		
Nome	Tipo	Descrição
<i>erro</i>	<i>long</i>	Código de erro standard do LABVIEW.

Tabela 53: Saídas de “write data file.vi”

I.2.5.13. Write peak file.vi

Descrição: Faz a gravação de um ficheiro de picos. Ver também **read peak file.vi**.

No painel desta VI é possível, à semelhança de **write data file.vi**, introduzir comentários que serão visualizados quando o utilizador abrir o ficheiro com **read peak file.vi**. Estes comentários aparecem na primeira secção do mesmo.

O formato de um ficheiro de picos está descrito no documento principal em *III.11. Formatos de ficheiros*.

Painel visível em execução: Sim.

Variáveis de entrada de “write peak file.vi”		
Nome	Tipo	Descrição
<i>tabela de picos</i>	<i>array 2D de string</i>	Tabela de picos fornecida por build picos table.vi e visualizada em ver picos.vi

Tabela 54: Variáveis de entrada de “write peak file.vi”

Saídas de “write peak file.vi”		
Nome	Tipo	Descrição
<i>erro</i>	<i>long</i>	Código de erro standard do LABVIEW

Tabela 55: Saídas de “write peak file.vi”

I.2.6. VI's de manipulação de sinais

I.2.6.1. Alteração info sinal.vi

Descrição: Serve como interface ao utilizador para se proceder a eventuais alterações de informação sobre sinais.

Embora os controlos deste painel se relacionem com o tipo **data info type**, não foram baseados nele. Deve ser evitada a introdução de valores errados nos controlos, pelo que se criou uma imagem de **data info type** com controlos devidamente protegidos. Assim, por exemplo, o utilizador escolhe para o ganho da placa de aquisição uma de diversas opções de um *ring control*. Este não fornece o valor escolhido, mas o seu índice na lista de opções. O tipo **data info type** utiliza, naturalmente, não este índice, mas o valor.

Diversos controlos são protegidos de uma forma igual ou semelhante. A VI faz, pois, a conversão de valores contidos na informação sobre sinais para os controlos acessíveis ao utilizador e vice-versa. A alteração do tipo **data info type** obriga à alteração desta VI.

Painel visível em execução: Sim.

Variáveis de entrada de “Alteração info sinal.vi”		
Nome	Tipo	Descrição
<i>first index</i>	<i>word</i>	Nº de sinal cuja informação deve aparecer quando o painel abrir

Tabela 56: Variáveis de entrada de “Alteração info sinal.vi”

Saídas de “Alteração info sinal.vi”		
Nome	Tipo	Descrição
<i>cancelled</i>	<i>boolean</i>	Indica se as alterações foram canceladas. Importante, pois uma alteração de valores de ganhos ou parâmetros de calibração implica uma reescalagem dos valores de força.

Tabela 57: Saídas de “Alteração info sinal.vi”

I.2.6.2. Display info sinal.vi

Descrição: Esta VI é um painel para visualização, em texto, da informação sobre um sinal. Não é utilizada actualmente, pois foi substituída por **Alteração info sinal.vi**. Pode ser modificada para outro fim. Não tem saídas.

Painel visível em execução: Sim.

Variáveis de entrada de “display info sinal.vi”		
Nome	Tipo	Descrição
<i>numero sinal</i>	<i>unsigned word</i>	Índice do sinal cuja informação vai ser visualizada
<i>nr aquisição</i>	<i>unsigned word</i>	Número de aquisição na qual foi obtido o sinal
<i>infoin</i>	<i>string</i>	Texto a ser visualizado (obtido através de build info sinal.vi)

Tabela 58: Variáveis de entrada de “display info sinal.vi”

I.2.6.3. Marcação de registos.vi

Descrição: A VI constitui uma caixa de diálogo com um controlo tipo *multiple selection list box* através da qual é possível seleccionar rapidamente múltiplos registos. Foi criada para facilitar a operação de selecção de registos no painel principal.

Sugere-se a utilização de um controlo tipo *multiple selection list box* directamente no painel principal em futuras versões (a indicação de tipo de sinal tornou-se desnecessária com a introdução de prefixos no nome do sinal), o que tornaria supérflua a utilização desta VI.

Painel visível em execução: Sim.

Variáveis de entrada de “marcação de registos.vi”		
Nome	Tipo	Descrição
<i>Registo permanente</i>	<i>array de clusters de controlo de banco de registos, conforme Tabela 51.</i>	Controlo de banco de registos com a selecção actual de registos

Tabela 59: Variáveis de entrada de “marcação de registos.vi”

Saídas de “marcação de registos.vi”		
Nome	Tipo	Descrição
Novas selecções	<i>array de clusters de controlo de banco de registos, conforme Tabela 51.</i>	Controlo de banco de registos com a nova selecção de registos

Tabela 60: Saídas de “alteração de cor de fundo.vi”

I.2.7. VI's de calibração de sensores

I.2.7.1. Avaliar calibração.vi

Descrição: A partir de um *array* de pesos (valores x) e um *array* de leituras para esses pesos (valores y) calcula uma interpolação linear pelo método dos mínimos quadrados. É utilizada por **calibração de sensores.vi**.

Fornecerá não só os valores m e b da recta, como também o erro médio quadrático da interpolação e um *array de clusters de gráfico* com um gráfico x-y resultante dos pontos fornecidos e da sua interpolação.

O gráfico é fornecido como um *array de clusters* de 2 elementos. Cada *cluster* nesse *array* corresponde a um traçado. Os elementos do *cluster* são *arrays*, em que um *array* contém os valores x do traçado, o outro os valores y.

Painel visível em execução: Não.

Variáveis de entrada de “avaliar calibração.vi”		
Nome	Tipo	Descrição
Pesos	<i>array de double</i>	Array com pesos aplicados no processo de calibração (valores x para a interpolação)
Leituras	<i>array de word</i>	Array com leituras obtidas para os pesos aplicados (valores y para a interpolação)

Tabela 61: Variáveis de entrada de “avaliar calibração.vi”

Saídas de “avaliar calibração.vi”		
Nome	Tipo	Descrição
Calibração (1) e curva ajustada (2)	<i>array de cluster de gráfico x-y conforme descrito.</i>	Gráfico de pontos obtidos na calibração e da recta ajustada a esses pontos

Tabela 62: Saídas de “avaliar calibração.vi”

I.2.7.2. Calibração de sensores.vi

Descrição: Esta VI constitui o painel de calibração de sensores, cujo funcionamento e utilização está descrito no documento principal em III.9. O *painel de calibração de sensores*.

O seu funcionamento consiste numa verificação e ajuste constante dos parâmetros de aquisição e na reacção à activação dos diversos botões, que provocam determinadas acções.

No final, verifica se todos os parâmetros de calibração de um sensor foram obtidos, visualizando mensagens de aviso caso esta condição não se verifique.

Os factores de calibração retornados não estão normalizados, isto é, terão que ser recalculados segundo o método descrito no documento principal em III.9.2. *Factores de calibração e fórmulas de conversão*.

Não tem entradas.

Painel visível em execução: Sim.

Saídas de “calibração de sensores.vi.vi”		
Nome	Tipo	Descrição
<i>m*</i>	<i>double</i>	Parâmetro m da calibração, não normalizado
<i>b*</i>	<i>double</i>	Parâmetro b da calibração, não normalizado
Zero placa*	<i>word</i>	Zero ajustado na placa externa, não normalizado
canal	<i>unsigned byte</i>	Canal em qual foi efectuada a calibração de um sensor
calibração aceite	<i>boolean</i>	TRUE se a calibração foi aceite

Tabela 63: Saídas de “calibração de sensores.vi”

I.2.8. VI's de análise

I.2.8.1. Calcula picos.vi

Descrição: Calcula as amplitudes de picos relativos ou absolutos a partir de cursores colocados na localização dos mesmos.

A VI tem como entrada o índice, nos registos permanentes, do sinal sobre o qual é efectuada o cálculo, e um *array* com posições de cursores nesse sinal. Estes cursores são obtidos no processo de detecção e definição de picos (ver no documento principal: III.10.2. *Detecção e cálculo de picos*).

A partir das posições destes cursores, o software procura a amplitude do sinal nesse ponto (caso do pico absoluto) ou a diferença de amplitudes entre dois pontos consecutivos (caso do pico relativo).

A VI fornece os picos calculados numa estrutura definida nas tabelas 36 e 37. Esta estrutura pode depois ser convertida para uma tabela (*array 2D de string*) com **build picos table.vi**.

Painel visível em execução: Não.

Variáveis de entrada de “calcula picos.vi”		
Nome	Tipo	Descrição
<i>Tipo de pico</i>	<i>boolean</i>	Pico absoluto (FALSE) ou pico relativo (TRUE)
<i>Cursor Indexes in</i>	<i>array de long</i>	Posições (índices) de cursores colocados sobre o sinal a analisar
<i>Índice de sinal</i>	<i>long</i>	Índice do sinal a analisar no banco de registos permanentes

Tabela 64: Variáveis de entrada de “calcula picos.vi”

Saídas de “calcula picos.vi”		
Nome	Tipo	Descrição
<i>Picos</i>	<i>array de clusters</i> do tipo que se definiu anteriormente (ver tabelas 36 e 37)	Estrutura de dados utilizada para guardar resultado da análise de picos.

Tabela 65: Saídas de “calcula picos.vi”

I.2.8.2. Derivada média móvel.vi

Descrição: Calcula a derivada de um sinal segundo o algoritmo descrito no documento principal em IV.3. *Detecção de picos*. Fornece como resultado um sinal que em cada ponto representa uma média de derivadas na vizinhança.

Restrições:

- *k* deve ser maior que *step*, maior que 1 e par; Se $k < step$, então $step=1$, se *k* ímpar, então $k=k-1$;
- *dt* deve ser maior que 0, senão é utilizado $dt=1$

Painel visível em execução: Não.

Variáveis de entrada de “derivada média móvel.vi”		
Nome	Tipo	Descrição
X	<i>array de double</i>	Sinal sobre qual vai ser efectuado o cálculo da derivada
dt	<i>double</i>	Intervalo de amostragem, $dt=1/f_a$, em que f_a é a frequência de amostragem
k	<i>unsigned long</i>	Número de amostras na vizinhança consideradas para o cálculo da derivada;
step	<i>unsigned long</i>	Distanciamento entre amostras para cálculo da diferença

Tabela 66: Variáveis de entrada de “derivada média móvel.vi”

Saídas de “derivada média móvel.vi”		
Nome	Tipo	Descrição
dX/dt	<i>array de double</i>	Sinal derivada do sinal original
error	<i>long</i>	error=1: $k < step$ error=2: $dt=0$ Em ambos os casos o cálculo efectua-se, nas condições descritas nas restrições

Tabela 67: Saídas de “derivada média móvel.vi”

I.2.8.3. Detecção de picos.vi

Descrição: Esta VI faz a detecção automática de picos relativos ou absolutos, fornecendo um vector de cursores que estão colocados sobre os locais onde a VI encontrou picos.

A VI determina as localizações de picos que serão fornecidos a **calcula picos.vi** para cálculo das amplitudes. As funções de detecção da localização e cálculo das amplitudes encontram-se separadas pois o resultado da detecção é ainda sujeito a uma fase de verificação e correcção por parte do utilizador antes de serem calculadas as amplitudes propriamente ditas.

A VI recebe um *array* de cursores que são interpretados em pares. Cada par delimita uma zona em que deve ser procurado um pico. Se o número de cursores neste array for ímpar, a VI não executa e é visualizada uma mensagem de erro.

A detecção de um pico absoluto consiste na procura de um máximo de amplitude na zona de detecção definida por um par de cursores. A detecção de um pico relativo baseia-se na procura de dois zeros da derivada do sinal, conforme no documento principal em IV.3. *Detecção de picos*.

Painel visível em execução: Não.

Variáveis de entrada de “detecção de picos.vi”		
Nome	Tipo	Descrição
<i>senal</i>	<i>array de double</i>	Sinal em que vai ser feita a procura de picos
<i>derivada</i>	<i>array de double</i>	Derivada do sinal, calculada com derivada média móvel.vi
<i>Tipo de pico</i>	<i>boolean</i>	Pico absoluto (FALSE) ou pico relativo (TRUE)
<i>Cursor indexes in</i>	<i>array de long</i>	Índices dos cursores que delimitam zonas de detecção
<i>k</i>	<i>unsigned long</i>	Parâmetro <i>k</i> com o qual foi calculada a derivada

Tabela 68: Variáveis de entrada de “detecção de picos.vi”

Saídas de “detecção de picos.vi”		
Nome	Tipo	Descrição
<i>Cursor List out</i>	<i>array de clusters de cursor (ver documentação LABVIEW)</i>	Lista de cursores colocados sobre os locais dos picos detectados. O formato do <i>cluster</i> é definido pela documentação do LABVIEW
<i>error</i>	<i>boolean</i>	TRUE se o número de cursores de delimitação de zonas de detecção for nulo ou ímpar

Tabela 69: Saídas de “detecção de picos.vi”

I.2.8.4. Painel de análise.vi

Descrição: Esta VI constitui o painel de análise. O seu funcionamento e os métodos de cálculo a ele subjacentes estão descritos no documento principal em III.10. O painel de análise e IV. As diversas técnicas de cálculo utilizadas. Os únicos parâmetros de entrada e saída utilizados relacionam-se com a indicação da selecção de sinais no banco de registos permanentes, e destinam-se a manter sincronizadas o controlo de registos permanentes do painel principal com o do painel de análise.

Comentários: O diagrama de blocos desta VI consiste numa única estrutura de sequência (*sequence*) com três fases (*frames*): Inicialização, funcionamento e finalização. As duas primeiras têm como função manter sincronizados os controlos de registos permanentes dos painéis principal e de análise. Para tal, a VI recebe o banco de registos permanentes do painel principal, inicializando a *list box* de selecção de registos permanentes do painel de análise com os mesmos valores. No final é

devolvida uma lista de registos seleccionados para que estes sejam mantidos com as mesmas selecções no painel principal

A fase de funcionamento (*frame* 1) está dividida nas quatro partes que se seguem:

- Uma estrutura *case* para adaptar o painel quando muda o tipo de análise a efectuar;
- Uma estrutura *case* para redesenhar o gráfico sempre que sejam alteradas as selecções de registos ou o tipo de escala x ;
- Uma estrutura *case* para as acções de análise propriamente ditas;
- Uma estrutura *case* para reagir à actuação de botões comuns a todos os tipos de análise.

Painel visível em execução: Sim.

Variáveis de entrada de “Painel de análise.vi”		
Nome	Tipo	Descrição
Controlo reg per	<i>array de clusters de controlo de banco de registos, conforme Tabela 51.</i>	Controlo de banco de registos com a selecção actual de registos permanentes

Tabela 70: Variáveis de entrada de “Painel de análise.vi”

Saídas de “Painel de análise.vi”		
Nome	Tipo	Descrição
Registos marcados	<i>array de long</i>	Lista de registos permanentes seleccionados pelo utilizador no painel de análise. Utilizado para que o painel principal actualize o estado das selecções de registos permanentes

Tabela 71: Saídas de “Painel de análise.vi”

I.2.8.5. Resposta em frequência de filtros IIR.vi

Descrição: Simula a resposta em frequência de um filtro IIR conforme descrito no documento principal em IV.5.3. *Simulação da resposta de filtros*. A geração de um impulso é feita pela VI **Impulse Pattern.vi**, contida nos exemplos do LABVIEW, e utiliza 1024 pontos.

Painel visível em execução: Não.

Variáveis de entrada de “Resposta em frequência de filtros IIR.vi”		
Nome	Tipo	Descrição
Ordem	<i>long</i>	Ordem do filtro/número de coeficientes utilizados pelo filtro digital no cálculo de uma filtragem
Tipo filtragem	<i>unsigned word enum</i> { <i>Passa-baixo, Passa-alto, Passa-banda, Rej.banda</i> }	Tipo de filtragem a simular. Controlo tipo <i>ring control</i>
Tipo de filtro	<i>unsigned word enum</i> { <i>Butterworth, Chebyshev, Chebyshev II, Elíptico, Bessel</i> }	Tipo de filtro a simular. Controlo tipo <i>ring control</i>
Freq.inferior	<i>double</i>	Frequência de corte inferior, em Hz
Freq.superior	<i>double</i>	Frequência de corte superior, em Hz. Ignorado se tipo de filtragem for passa-alto ou passa-baixo
Ripple	<i>double</i>	Ripple máximo na banda passante, em dB (nem sempre aplicável, dependendo de tipo de filtro.Ver documento principal: <i>IV.5. Filtragem digital</i>)
Atenuação	<i>double</i>	Atenuação mínima na banda de corte, em dB (nem sempre aplicável, dependendo de tipo de Ver documento principal: <i>IV.5. Filtragem digital</i>)
Resposta de filtro	<i>boolean</i>	Resposta a simular: Em amplitude (TRUE) ou em fase (FALSE)
Ind.reg.permanente	<i>unsigned long</i>	Índice do primeiro registo permanente seleccionado. A simulação basear-se-à na frequência de amostragem desse sinal

Tabela 72: Variáveis de entrada de “Resposta em frequência de filtros IIR.vi”

Saídas de “Resposta em frequência de filtros IIR.vi”		
Nome	Tipo	Descrição
Gráfico	<i>array de cluster de gráfico</i>	O cluster de gráfico tem a estrutura descrita na Tabela 25
Título	<i>string</i>	Título para o gráfico: Indicação de unidades

Saídas de “Resposta em frequência de filtros IIR.vi”		
Nome	Tipo	Descrição
		de escalas x e y

Tabela 73: Saídas de “Resposta em frequência de filtros IIR.vi”

I.2.8.6. Ver picos.vi

Descrição: Esta VI constitui o painel de visualização e edição de picos (ver documento principal: *III.10.2. Detecção e cálculo de picos*), onde é possível apagar picos da tabela e abrir/gravar ficheiros de picos.

Comentários: Esta VI utiliza **build picos table.vi** e **build picos from table.vi**.

Painel visível em execução: Sim.

Variáveis de entrada de “Ver picos.vi”		
Nome	Tipo	Descrição
Picos in	<i>array de clusters</i> do tipo que se definiu anteriormente (ver tabelas 36 e 37)	Estrutura de dados utilizada para guardar resultado da análise de picos. É fornecida à VI para se fazer a visualização da tabela de picos.

Tabela 74: Variáveis de entrada de “Ver picos.vi”

Saídas de “Ver picos.vi”		
Nome	Tipo	Descrição
Picos out	<i>array de clusters</i> do tipo que se definiu anteriormente (ver tabelas 36 e 37)	Estrutura de dados utilizada para guardar resultado da análise de picos. Reflecte conteúdo da tabela de picos após execução de Ver picos.vi
Tabela de picos	<i>array 2D de string</i>	Tabela de picos em forma de matriz de strings. Contém a mesma informação que picos out , portanto também actualizada após execução de Ver picos.vi .

Tabela 75: Saídas de “Ver picos.vi”

I.2.9. VI's de cálculo diverso

I.2.9.1. Calc velocidade.vi

Descrição: Faz o cálculo da velocidade de costura média durante uma aquisição, conforme descrito no documento principal em IV.2. *Cálculo da velocidade de costura*. Para tal, é necessário que tenha havido durante a aquisição no mínimo uma rotação completa da máquina.

Comentários: É imperativo para um correcto funcionamento desta VI que a placa de aquisição esteja configurada para iniciar as aquisições no flanco ascendente do sinal de sincronismo, pois o algoritmo conta rotações completas da máquina a partir da detecção desse flanco.

Caso haja necessidade de iniciar as aquisições em qualquer ponto, a VI tem de ser modificada de modo a procurar o primeiro flanco do sinal de sincronismo e contar períodos e amostras a partir desse ponto. (Esta modificação pode tornar-se necessária para, por exemplo, indicar a velocidade de costura no painel de osciloscópio, em que as aquisições podem ser feitas sem *hardware trigger*).

A VI dispõe de um indicador de precisão de cálculo, não utilizado e por isso não conectado de momento.

Painel visível em execução: Não.

Variáveis de entrada de "calc velocidade.vi"		
Nome	Tipo	Descrição
<i>senal sincro</i>	<i>array de word</i>	Sinal de sincronismo, em valores digitais lidos pela placa
<i>fs</i>	<i>double</i>	Frequência de amostragem do sinal de sincronismo.

Tabela 76: Variáveis de entrada de "calc velocidade.vi"

Saídas de "calc velocidade.vi"		
Nome	Tipo	Descrição
<i>velocidade</i>	<i>double</i>	Velocidade média de costura calculada

Tabela 77: Saídas de "calc velocidade.vi"

I.2.9.2. Converter para binário.vi

Descrição: Reconverte sinais já convertidos em valores de força para valores digitais lidos da placa a partir da informação sobre os sinais (parâmetros de calibração e ganhos). Ver documento principal: III.9.2. *Factores de calibração e fórmulas de conversão*.

Painel visível em execução: Não.

Variáveis de entrada de “converter para binário.vi”		
Nome	Tipo	Descrição
Info reg in	<i>array de clusters do tipo data info type</i>	Informação sobre os sinais a serem reconvertidos
data array in	<i>array 2D de double</i>	Matriz que contém os sinais a serem reconvertidos

Tabela 78: Variáveis de entrada de “converter para binário.vi”

Saídas de “converter para binário.vi”		
Nome	Tipo	Descrição
data array out	<i>array 2D de word</i>	Sinais convertidos para valores digitais

Tabela 79: Saídas de “converter para binário.vi”

I.2.9.3. Escalar ou reescalar vectores.vi

Descrição: Converte sinais de valores digitais para valores de força, ou recalcula essa conversão caso se tenham alterado os factores de calibração e/ou de ganhos na informação sobre determinado sinal (ver documento principal: *III.9.2. Factores de calibração e fórmulas de conversão*).

No segundo caso, é feita uma reconversão para digital, para de seguida ser feita a conversão para valores de força com base nos novos factores de calibração e/ou ganhos. Para tal, é necessário fornecer à VI tanto a informação actualizada sobre o sinal como a informação que existia antes da modificação).

A VI aceita sinais em formato de matriz ou em formato de array de clusters. Este último é o formato utilizado no banco de registos permanente (ver *I.1.2.2 Variáveis globais de dados (sinais): globals data.vi*).

O formato de matriz apropria-se para a conversão de valores digitais em valores de força logo após a aquisição, pois é nesse formato que as VI's de aquisição fornecem os sinais.

O segundo formato apropria-se para reconverter sinais que já se encontravam nos bancos de registos.

Painel visível em execução: Não.

Variáveis de entrada de “escalar ou reescalar vectores.vi”		
Nome	Tipo	Descrição
Info reg in	<i>array de clusters do tipo data info</i>	Informação sobre os sinais a serem reconvertidos depois de alterações

Variáveis de entrada de “escalar ou reescalar vectores.vi”		
Nome	Tipo	Descrição
	<i>type</i>	efectuadas pelo utilizador
<i>old info reg in</i>	<i>array de clusters do tipo data info type</i>	Informação sobre os sinais a serem reconvertidos antes de alterações efectuadas pelo utilizador
<i>data array in</i>	<i>array de double</i>	Sinais, em formato de matriz
<i>data cluster array in</i>	<i>array de cluster de array de double</i>	Sinais, em formato de <i>array de cluster</i> (formato utilizado nas variáveis globais de dados)
<i>input binary</i>	<i>boolean</i>	TRUE: Sinal de entrada está em valores digitais. FALSE: Sinal já escalado em valores de força

Tabela 80: Variáveis de entrada de “escalar ou reescalar vectores.vi”

Saídas de “escalar ou reescalar vectores.vi”		
Nome	Tipo	Descrição
<i>data array out</i>	<i>array 2D de double</i>	Resultado da conversão ou reconversão de <i>data array in</i>
<i>data cluster array out</i>	<i>array de cluster de array de double</i>	Resultado da conversão ou reconversão de <i>data cluster array in</i>

Tabela 81: Saídas de “escalar ou reescalar vectores.vi”

I.2.10. VI's de impressão

A impressão de gráficos necessita, para cada caso, de duas VI's:

- Uma VI cujo painel é visível, e em que o utilizador configura o gráfico da forma que o deseja ver impresso;
- Uma segunda VI cujo painel não é visível, e que é uma imagem da primeira. Esta VI é executada quando o utilizador confirma a impressão, e tem o atributo “Print panel when VI completes execution” (VI Setup-Execution Options). Assim, após a sua execução, que consiste simplesmente em configurar o painel à imagem do painel que o utilizador configurou, é feita a sua impressão.

O software tem, assim, 4 VI's de impressão, 2 para cada caso (impressão de gráficos de sinais e impressão de gráficos de calibração)

I.2.10.1. Imprimir calibração.vi

Descrição: Painel em que é feita a configuração do aspecto com o qual um gráfico de calibração deve ser impresso, e em que o utilizador pode acrescentar comentários a figurarem na impressão. Ao confirmar a operação, é executada uma VI de impressão (**painel impressão cal.vi**) cujo painel é uma imagem do painel da presente VI.

Não tem saídas.

Painel visível em execução: Sim.

Variáveis de entrada de “imprimir calibração.vi”		
Nome	Tipo	Descrição
<i>data in</i>	<i>cluster de 6 elementos</i>	Cluster com diversas informações e resultados da calibração. Ver próxima tabela
<i>graph control</i>	<i>cluster de 10 elementos segundo formato em Tabela 31</i>	Atributos do gráfico da VI chamadora. O gráfico desta VI é configurado com estes atributos
<i>vectores</i>	<i>array de clusters de 2 elementos</i>	Gráfico de calibração, tipo gráfico x-y. Cada cluster contém um array de abcissas e outro de ordenadas. Cada cluster é um traçado.

Tabela 82: Variáveis de entrada de “imprimir calibração.vi”

Estrutura do cluster <i>data in</i>		
Nome	Tipo	Descrição
<i>m in</i>	<i>double</i>	Parâmetro m^* obtido no processo de calibração
<i>b in</i>	<i>double</i>	Parâmetro b^* obtido no processo de calibração
<i>emq in</i>	<i>double</i>	Erro médio quadrático da interpolação linear feita no processo de calibração
<i>Zero placa</i>	<i>double</i>	Parâmetro <i>Zero placa</i> do processo de calibração
<i>canal in</i>	<i>word</i>	Nº do canal em que foi feita a calibração
<i>nome in</i>	<i>string</i>	Nome do canal em que foi feita a calibração

Tabela 83: Estrutura do cluster *data in*

I.2.10.2. Imprimir gráficos.vi

Descrição: Painel em que é feita a configuração do aspecto com o qual um gráfico de sinais deve ser impresso, e em que o utilizador pode acrescentar comentários, um título e descrições de escalas a figurarem na impressão. Ao confirmar a operação, é executada uma VI de impressão (**painel impressão gráficos.vi**) cujo painel é uma imagem do painel da presente VI.

Não tem saídas.

Painel visível em execução: Sim.

Variáveis de entrada de “imprimir gráficos.vi”		
Nome	Tipo	Descrição
gráfico	<i>array de cluster de gráfico</i>	Gráfico a imprimir. Cada traçado é um cluster, estrutura conforme <i>Tabela 25</i>
graph control	<i>cluster de 10 elementos segundo Tabela 31</i>	Atributos do gráfico da VI chamadora. O gráfico desta VI com estes atributos

Tabela 84: Variáveis de entrada de “imprimir gráficos.vi”

I.2.10.3. Painel impressão cal.vi

Descrição: VI que efectua a impressão de um gráfico de calibração. Esta VI é chamada por **imprimir calibração.vi**. O seu painel é configurado através das variáveis de entrada, e no final da execução, a opção “Print panel when VI completes execution” (VI Setup-Execution Options) obriga o painel a ser enviado para a impressora.

Não tem saídas.

Painel visível em execução: Não.

Variáveis de entrada de “painel impressão cal.vi”		
Nome	Tipo	Descrição
data in	<i>cluster de 6 elementos</i>	Cluster com diversas informações e resultados da calibração. Ver <i>Tabela 83</i>
vectores	<i>array de clusters de 2 elementos</i>	Gráfico de calibração, tipo gráfico x-y. Cada cluster contém um array de abcissas e outro de ordenadas. Cada cluster é um traçado.
graph control cal	<i>cluster de 3 elementos</i>	Atributos do gráfico da VI chamadora. O gráfico desta VI é configurado com estes atributos. Ver próxima tabela
comments	<i>string</i>	Comentários que o utilizador acrescentou em imprimir calibração.vi

Tabela 85: Variáveis de entrada de “painel impressão cal.vi”

Estrutura do cluster <i>graph control cal</i> ¹		
Nome	Tipo	Descrição
X Scale Info	<i>cluster de 8 elementos</i>	Informação sobre escala x. Consultar documentação LABVIEW
Y Scale Info	<i>cluster de 8 elementos</i>	Informação sobre escala y. Consultar documentação LABVIEW
Plot info array	<i>array de cluster de 8 elementos</i>	Informação sobre cada traçado do gráfico. (<i>Plot Info</i>). Consultar documentação LABVIEW

Tabela 86: Estrutura do cluster *graph control cal*

I.2.10.4. Painel de impressão gráficos.vi

Descrição: VI que efectua a impressão de um gráfico de sinais. Esta VI é chamada por **imprimir gráficos.vi**. O seu painel é configurado através das variáveis de entrada, e no final da execução, a opção “Print panel when VI completes execution” (VI Setup-Execution Options) obriga o painel a ser enviado para a impressora.

Não tem saídas.

Painel visível em execução: Não.

Variáveis de entrada de “painel impressão gráficos.vi”		
Nome	Tipo	Descrição
gráfico	<i>array de clusters de gráfico</i>	Gráfico a imprimir. Cada traçado é um cluster, estrutura conforme <i>Tabela 25</i>
comments	<i>string</i>	Comentários acrescentados pelo utilizador
título	<i>string</i>	Título do gráfico
nomey	<i>string</i>	Nome para escala y
nomex	<i>string</i>	Nome para escala x
graph control	<i>cluster de 10 elementos segundo Tabela 31</i>	Atributos do gráfico da VI chamadora. O gráfico desta VI com estes atributos

Tabela 87: Variáveis de entrada de “painel impressão gráficos.vi”

¹ O cluster *graph control cal*, utilizado nesta VI é mais simples que o cluster *graph control* que existe nas VI's associadas à impressão e/ou visualização de gráficos de sinais, pois na impressão de calibração existem menos atributos que o utilizador influencia directamente.

I.2.11. O painel principal: Main.vi

Descrição: Esta VI é a VI fundamental de todo o software, permitindo efectuar um grande número de operações, e funcionando também como elo de ligação entre as restantes VI's.

Quando ela é aberta, todas as restantes VI's são carregadas para a memória, pois esta VI está no topo da hierarquia.

Não tem saídas nem entradas.

Comentários: Main.vi é constituída por uma estrutura de sequência com duas frames:

- Uma primeira de inicialização;
- Uma segunda de execução.

A segunda *frame* consiste num ciclo *while* que executa enquanto que o utilizador não escolher a opção "Fim" do menu principal.

No ciclo *while* encontramos três estruturas *case*, com as seguintes funções:

- O processamento das opções do menu principal (*case* na parte superior)
- O processamento da actuação dos botões (*case* central);
- O processamento das opções do menu gráfico (*case* na parte inferior).

Painel visível em execução: Sim.

II. Especificações técnicas

II.1. Características técnicas da placa de aquisição LAB-PC+

II.2. Esquemático da placa de condicionamento tipo “piezoelétricos”

II.3. Esquemático da placa de condicionamento tipo “extensómetros”

II.4. Características técnicas dos sensores

II.4.1. Sensores de força piezoelétricos

II.4.2. Sensores de força extensiométricos

III. Fotografias

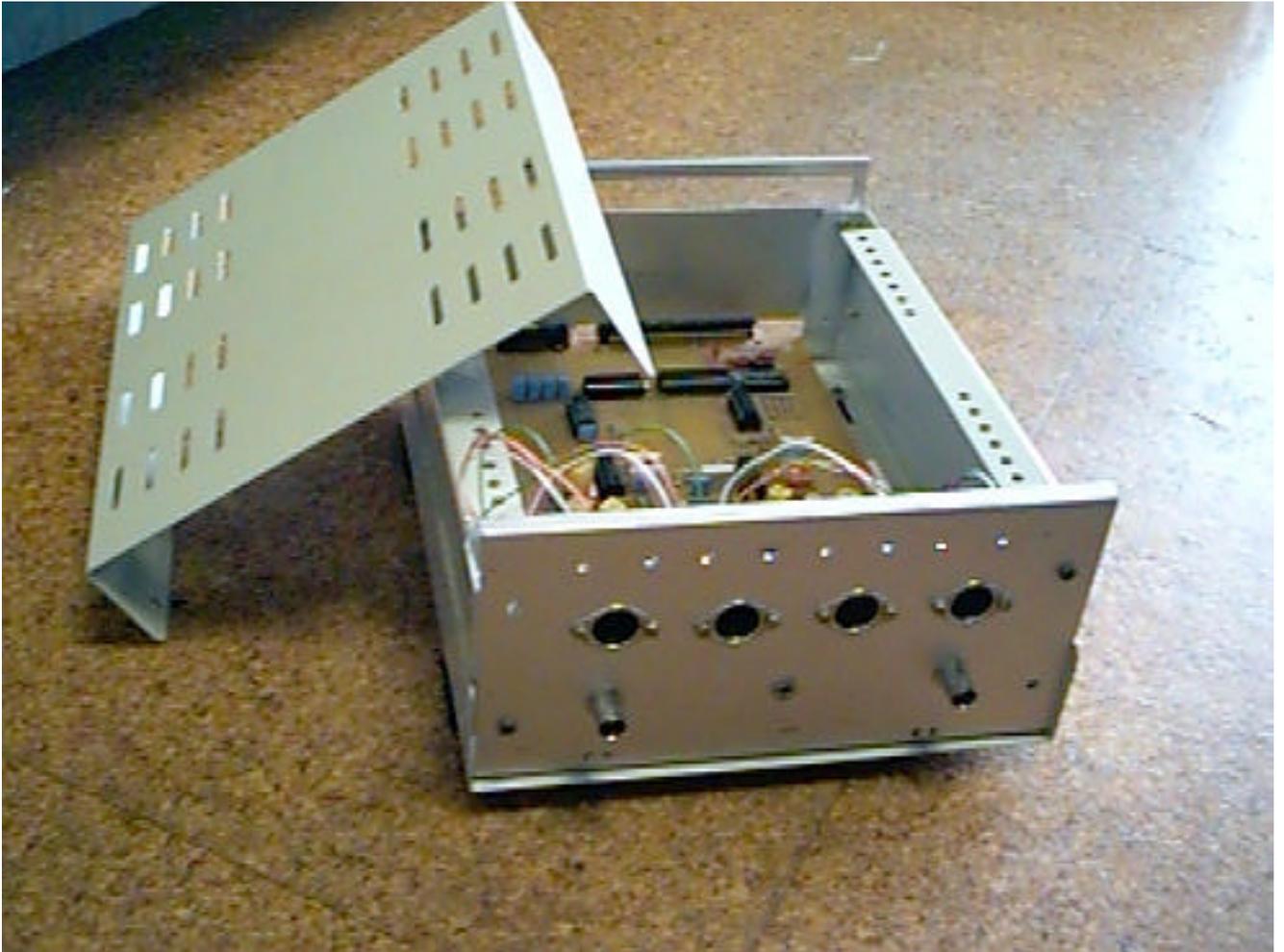


Fig. 1: Suporte físico para as placas de condicionamento

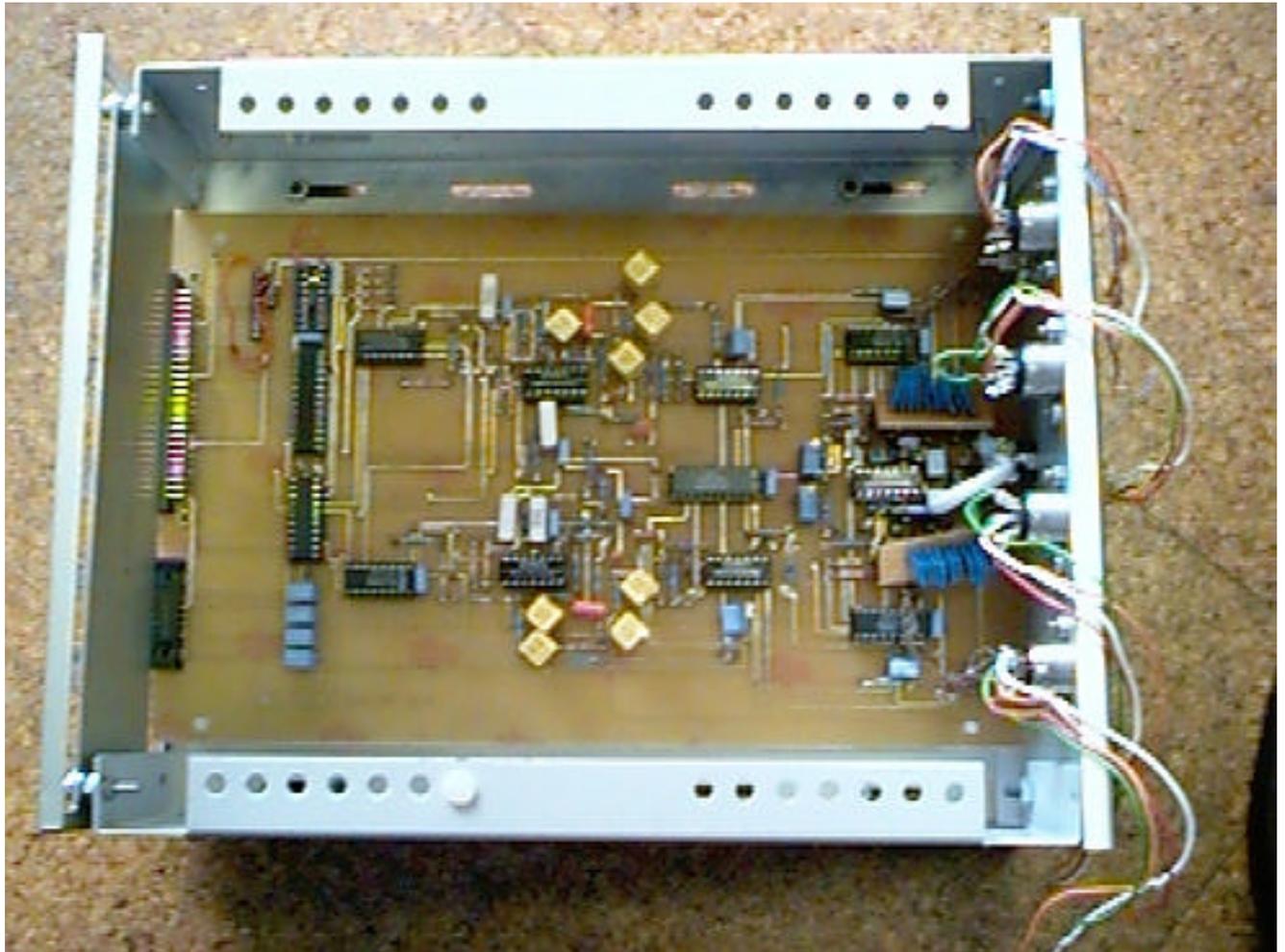


Fig. 2: Placa tipo piezoelétricos

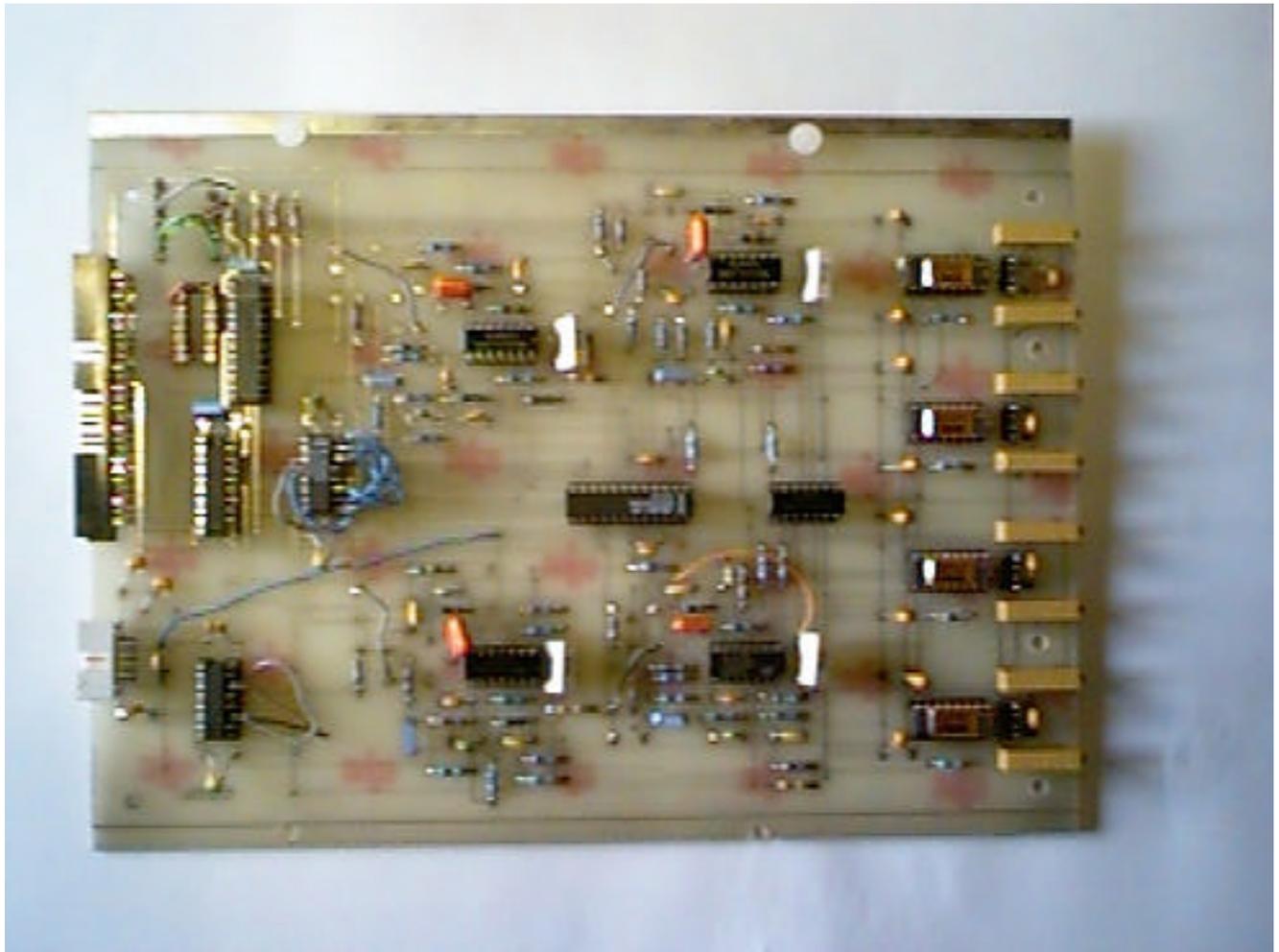


Fig. 3 : Placa tipo extensómetros

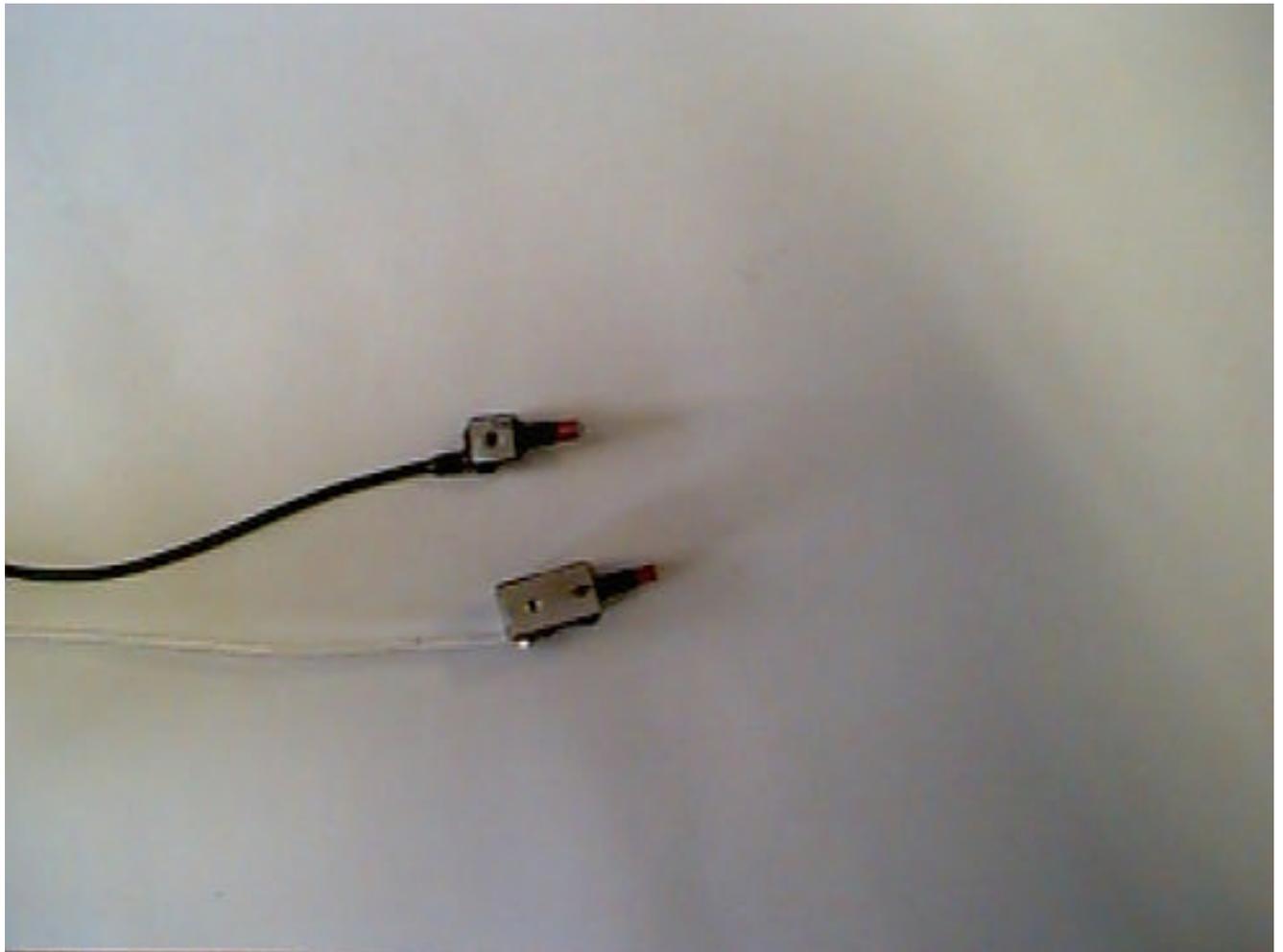


Fig. 4: Sensores extensiométricos para medição de tensão de linha

IV. Índice de tabelas

Tabela 1: Elementos do cluster que forma o tipo <i>data type info</i>	5
Tabela 2: Variáveis globais de configuração	6
Tabela 3: Cluster de configuração de um canal.....	7
Tabela 4: Variáveis de entrada de “calculate fs.vi”	8
Tabela 5: Saídas de “calculate fs.vi”	9
Tabela 6: Variáveis de entrada de “check config canais.vi”	9
Tabela 7: Saídas de “check config canais.vi”.....	10
Tabela 8: Saídas de “adquiri.vi”	11
Tabela 9: Variáveis de entrada de “adquirir osciloscópio.vi”	11
Tabela 10: Saídas de “adquirir osciloscópio.vi”	11
Tabela 11: Variáveis de entrada de “aquis n valores.vi”	12
Tabela 12: Saídas de “aquis n valores.vi”	12
Tabela 13: Variáveis de entrada de “teste de saturação.vi”	13
Tabela 14: Saídas de “teste de saturação.vi”	13
Tabela 15: Saídas de “trigger countdown.vi”	14
Tabela 16: Saídas de “programar ganhos”	14
Tabela 17: Variáveis de entrada de “programar um ganho.vi”	15
Tabela 18: Saídas de “programar um ganho.vi”	15
Tabela 19: Variáveis de entrada de “programar switches.vi”	17
Tabela 20: Saídas de “programar switches.vi”	17
Tabela 21: Variáveis de entrada de “alteração de cor de fundo.vi”	17
Tabela 22: Saídas de “alteração de cor de fundo.vi”	18
Tabela 23: Variáveis de entrada de “alteração de grelhas.vi”	18
Tabela 24: Saídas de “alteração de grelhas.vi”	18
Tabela 25: Construção de um <i>cluster de gráfico</i>	19
Tabela 26: Tipos de sinais	19
Tabela 27: Visualização de sinais nas diversas escalas	19
Tabela 28: Variáveis de entrada de “build graphs.vi”	20
Tabela 29: Saídas de “bulid graphs.vi”	20
Tabela 30: Variáveis de entrada de “alteração de cor de fundo.vi”	21
Tabela 31: Atributos de gráfico utilizados na janela de gráficos.....	21
Tabela 32: Variáveis de entrada de “build info sinal.vi”	22
Tabela 33: Saídas de “build info sinal.vi”	22

Tabela 34: Variáveis de entrada de “build info sinal cluster.vi”	23
Tabela 35: Saídas de “build info sinal cluster.vi”	23
Tabela 36: Cluster que contém picos detectados num período do sinal.....	24
Tabela 37: Cluster para pico individual.....	24
Tabela 38: Variáveis de entrada de “build picos from table.vi”	25
Tabela 39: Saídas de “build picos from table.vi”	25
Tabela 40: Variáveis de entrada de “build picos table.vi”	25
Tabela 41: Saídas de “build picos table.vi”	25
Tabela 42: Variáveis de entrada de “display file.vi”	26
Tabela 43: Saídas de “read cal file.vi”	26
Tabela 44: Saídas de “read config.vi”	27
Tabela 45: Saídas de “read data file.vi”	28
Tabela 46: Saídas de “read peak file.vi”	29
Tabela 47: Variáveis de entrada de “write cal file.vi”	29
Tabela 48: Saídas de “write cal file.vi”	29
Tabela 49: Variáveis de entrada de “write config.vi”	30
Tabela 50: Saídas de “write config.vi”	30
Tabela 51: Cluster de controlo de banco de registos	31
Tabela 52: Variáveis de entrada de “write data file.vi”	31
Tabela 53: Saídas de “write data file.vi”	31
Tabela 54: Variáveis de entrada de “write peak file.vi”	32
Tabela 55: Saídas de “write peak file.vi”	32
Tabela 56: Variáveis de entrada de “Alteração info sinal.vi”	32
Tabela 57: Saídas de “Alteração info sinal.vi”	32
Tabela 58: Variáveis de entrada de “display info sinal.vi”	33
Tabela 59: Variáveis de entrada de “marcação de registos.vi”	33
Tabela 60: Saídas de “alteração de cor de fundo.vi”	34
Tabela 61: Variáveis de entrada de “avaliar calibração.vi”	34
Tabela 62: Saídas de “avaliar calibração.vi”	35
Tabela 63: Saídas de “calibração de sensores.vi”	35
Tabela 64: Variáveis de entrada de “calcula picos.vi”	36
Tabela 65: Saídas de “calcula picos.vi”	36
Tabela 66: Variáveis de entrada de “derivada média móvel.vi”	37
Tabela 67: Saídas de “derivada média móvel.vi”	37
Tabela 68: Variáveis de entrada de “detecção de picos.vi”	38
Tabela 69: Saídas de “detecção de picos.vi”	38

Tabela 70: Variáveis de entrada de “Painel d análise.vi”	39
Tabela 71: Saídas de “Painel de análise.vi”	39
Tabela 72: Variáveis de entrada de “Resposta em frequência de filtros IIR.vi”	40
Tabela 73: Saídas de “Resposta em frequência de filtros IIR.vi”	41
Tabela 74: Variáveis de entrada de “Ver picos.vi”	41
Tabela 75: Saídas de “Ver picos.vi”	41
Tabela 76: Variáveis de entrada de “calc velocidade.vi”	42
Tabela 77: Saídas de “calc velocidade.vi”	42
Tabela 78: Variáveis de entrada de “converter para binário.vi”	43
Tabela 79: Saídas de “converter para binário.vi”	43
Tabela 80: Variáveis de entrada de “escalar ou reescalar vectores.vi”	44
Tabela 81: Saídas de “escalar ou reescalar vectores.vi”	44
Tabela 82: Variáveis de entrada de “imprimir calibração.vi”	45
Tabela 83: Estrutura do cluster <i>data in</i>	45
Tabela 84: Variáveis de entrada de “imprimir gráficos.vi”	46
Tabela 85: Variáveis de entrada de “painel impressão cal.vi”	47
Tabela 86: Estrutura do cluster <i>graph control cal</i>	47
Tabela 87: Variáveis de entrada de “painel impressão gráficos.vi”	47