



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA  
**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

**INPE-13030-PRE/8307**

**DESENVOLVIMENTO DO SOFTWARE PARA O BANCO DE  
HOMOLOGAÇÃO DE PCD'S PARA PLATAFORMA IBM-PC**

Valério Ramalho Nobre\*

\*Bolsista UFRN

Relatório Final de Projeto de Iniciação Científica (PIBIC/CNPq/INPE), orientado pelo  
Dr. Marcos Aurélio Ferreira dos Santos



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA  
**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

## **DESENVOLVIMENTO DO SOFTWARE PARA O BANCO DE HOMOLOGAÇÃO DE PCD'S PARA PLATAFORMA IBM-PC**

**RELATÓRIO FINAL DE PROJETO DE INICIAÇÃO CIENTÍFICA  
(PIBIC/CNPq/INPE)**

Valério Ramalho Nobre (UFRN, Bolsista PIBIC/CNPq)  
E-mail: [valerionobre@hotmail.com](mailto:valerionobre@hotmail.com)

Marcos Aurelio Ferreira dos Santos (INPE/CRN, Orientador)  
E-mail: [aurelio@crn2.inpe.br](mailto:aurelio@crn2.inpe.br)

### **COLABORADORES**

Manoel Jozeane (INPE/CRN)  
Roberto Moraes (INPE/CRN)  
João Gualberto (INPE/CRN)

julho de 2005

## RESUMO

O projeto em questão visa à atualização do sistema do Banco de Homologação de transmissores para o uso com o Satélite de Coleta de Dados brasileiro (SCD). Os transmissores a serem homologados por esse sistema serão as PCDs (Plataforma de Coleta de Dados). O sistema atual usa instrumentos físicos para as medições, o software que o controla está implementado em linguagem Basic e funciona em uma plataforma *Apple II Plus*, assim, além de utilizar um equipamento obsoleto, o banco de homologação atual apresenta falhas com uma boa frequência. O projeto objetiva desenvolver um sistema usando a tecnologia PXI (*PCI eXtensions for Instrumentation*), para a aquisição e digitalização dos dados das PCDs, comunicando-se com uma plataforma IBM-PC que executará um software desenvolvido na linguagem de programação visual LabView. O barramento PXI é composto por três módulos o MXI-4, para comunicação com o PC, o *downconverter*, para o condicionamento do sinal de saída da PCD, e o digitalizador, responsável por transformar o sinal analógico em digital. Assim, nesse sistema, a instrumentação física é substituída pela instrumentação virtual do LabView, a plataforma *Apple* deu lugar a uma IBM-PC, que provê melhor desempenho e maior facilidade de manutenção ao sistema, além do sistema PXI que oferece alta confiabilidade, robustez e bom desempenho.

### Palavras-chave

Homologação

Transmissor

Satélite

Instrumentação virtual

# SUMÁRIO

<b>CAPÍTULO 1 - INTRODUÇÃO</b> .....	1
<b>CAPÍTULO 2 - FUNDAMENTAÇÃO TEÓRICA</b> .....	3
2.1. Plataforma de Coleta de Dados (PCD) .....	3
2.2. Banco de Homologação .....	4
2.3. <i>PCI eXtensions for Instrumentation</i> (PXI) .....	12
2.4. LabView .....	17
<b>CAPÍTULO 3 - ARQUITETURA PROPOSTA PARA O BANCO DE HOMOLOGAÇÃO</b> .....	23
<b>CAPÍTULO 4 - RESULTADOS</b> .....	24
4.1. Instalação dos Equipamentos .....	24
4.2. Testes Iniciais .....	24
4.3. NiTuner e NI-SCOPE .....	24
4.4. Sincronismo com a PCD .....	25
4.5. Problemas com Espaço .....	25
4.6. Aquisição Completa .....	26
4.7. Gravação e Leitura em Arquivo .....	29
4.8. Medições de Parâmetros .....	31
<b>CAPÍTULO 5 - CONCLUSÃO</b> .....	33
<b>APÊNDICE A – Instalação e Configuração dos Equipamentos</b> .....	34
<b>APÊNDICE B – Código dos Principais Programas Desenvolvidos</b> .....	40
<b>REFERÊNCIAS</b> .....	42

## LISTA DE FIGURAS

Figura 1: Foto de uma PCD em campo com seus sensores .....	3
Figura 2: Foto do banco de homologação atual do INPE/CRN .....	4
Figura 3: Diagrama esquemático do banco de homologação [2] .....	5
Figura 4: Sinais de teste do banco de homologação .....	7
Figura 5: Arquitetura de comunicação entre módulos PXI [3] .....	13
Figura 6: Especificação da arquitetura PXI.....	15
Figura 7: Chassis PXI e módulos usados .....	17
Figura 8: (a) Painel Frontal e (b) Diagrama de Blocos de um programa em LabView.....	18
Figura 9: Diretório com VIs do ni5660 .....	19
Figura 10: Diretório com VIs do NiTuner .....	19
Figura 11: Diretório com VIs do NI-SCOPE .....	20
Figura 12: Diretório com VIs de configuração do NI-SCOPE .....	20
Figura 13: Diretórios com VIs de aquisição do NI-SCOPE, (a) em alto nível e (b) baixo nível.....	21
Figura 14: Diretório com VIs de configuração de triggers do NI-SCOPE .....	22
Figura 15: Diagrama esquemático do banco de homologação proposto .....	23
Figura 16: Foto dos equipamentos a serem usados no novo banco de homologação .....	23
Figura 17: Parte do software de aquisição de dados da parte do digitalizador.....	27
Figura 18: Parte do software de aquisição de dados da parte do <i>downconverter</i> .....	28
Figura 19: Parte do software de aquisição de dados responsável pela gravação dos dados em arquivo.....	30
Figura 20: Parte do software responsável pela leitura dos dados no arquivo.....	30
Figura 21: <i>Extract single tone information.vi</i> .....	31

## CAPÍTULO 1 - INTRODUÇÃO

O projeto em questão visa o desenvolvimento para um software em plataforma IBM-PC para o banco de homologação de Plataformas de Coleta de Dados (PCDs) para o Instituto Nacional de Pesquisas Espaciais – Centro Regional de Natal (INPE-CRN).

O Centro Regional do INPE, com sede em Natal, foi estabelecido por volta de 1970, quando o INPE chamava-se Comissão Nacional de Pesquisas Espaciais (CNAE) e assinou um convênio com o Governo do Estado e Universidade Federal do RN, visando estabelecer um núcleo de apoio aos lançamentos de foguetes e balões operados na Barreira do Inferno.

Após algum tempo de trabalho, o INPE/CRN passou a desenvolver equipamentos destinados à conexão com satélites, entre outros, montou uma Plataforma de Coleta de Dados (PCD) para operar com o Sistema CLS/ARGOS e já em 1983 a primeira PCD brasileira era testada e homologado nos laboratórios da CNES – França. De lá para cá a atividade de coleta de dados via satélite tem tomado dimensões nacionais, hoje com 350 unidades operando em todo o Brasil, inclusive operando com os satélites brasileiros.

Após está capacitado a montar PCDs, o INPE/CRN recebeu a outorga para realizar as homologações para o os transmissores do sistema ARGOS que fosse atuar no Brasil, o próximo passo foi o desenvolvimento de um sistema capaz de homologar esses dispositivos no próprio INPE. E assim, por volta de 1986, foram adquiridos os equipamentos necessários para a construção do banco de homologação, fazendo com que o INPE-CRN seja o único local no Brasil autorizado a fazer a homologação de transmissores para que estes possam trabalhar com os satélites da NOAA (*National Oceanic & Atmospheric Administration*). O sistema foi montado com os vários equipamentos necessários e um computador Apple II com um programa na linguagem **Basic** para a realização dos cálculos necessários. O sistema do INPE é capaz de fazer a homologação tanto de transmissores que trabalharão no Sistema de Coleta de Dados (SCD) brasileiro, como dos do sistema ARGOS, bastando para isso pequenos ajustes de configuração, já que os testes de desempenho realizados são os mesmos.

A presença desse banco de homologação no Brasil é de fundamental importância visto a utilidade das PCDs e caso ele não existisse o local mais próximo para se fazer tal procedimento seria na França, como foi feito com os primeiros protótipos construídos no INPE.

Porém com o passar do tempo, as PCDs evoluíram e o banco de homologação não acompanhou esse desenvolvimento, tornando-se um sistema ultrapassado e de difícil manutenção. Outro problema surgiu quando, as PCDs mais modernas não disponibilizavam em sua interface determinados sinais necessários para a operação do banco de homologação. Para não parar as homologações no INPE-CRN, um receptor francês do banco de homologação de transmissores chamados COSPAS/SARSAT , que é usado em aeronaves para o rastreamento em caso de pane na mesma, foi adaptado para que o banco do INPE pudesse homologar esses tipos de PCDs mais modernas.

Portanto, esse projeto tem o objetivo de desenvolvimento de um software para o banco de homologação em uma plataforma *desktop* PC, que através de uma interface usando a tecnologia PXI, adquire os dados da PCD a ser testada e faz a avaliação de todos os parâmetros necessários à mesma.

Como objetivos secundários do projeto, pode-se destacar o estudo da tecnologia PXI e da linguagem de programação LabView.

## CAPÍTULO 2 - FUNDAMENTAÇÃO TEÓRICA

### 2.1. Plataforma de Coleta de Dados (PCD)

Plataforma de Coleta de Dados (PCD) é um dispositivo eletrônico com capacidade de transmitir para um satélite, parâmetros ambientais captados pelo uso de sensores.

A necessidade de obter informações regulares, colhidas em locais remotos ou regiões vastas, como, por exemplo, dados meteorológicos (temperatura, pressão, direção e velocidade do vento, umidade, etc.), para aplicações em previsão de tempo por especialistas, levou à instalação de aparelhos de registros dessas informações. Na figura 1, temos a foto de uma PCD em campo com alguns sensores acoplados:



Figura 1: Foto de uma PCD em campo com seus sensores

Para a coleta destes registros, havia a necessidade de visitas periódicas aos locais ou ainda a manutenção de pessoas residentes aos locais que fizessem as leituras e as transmitissem de alguma forma a uma central de tratamento, tornando estes métodos dispendiosos e pouco eficientes.

Diante dessas necessidades de aumentar a eficiência na coleta dos dados, de aperfeiçoar a qualidade destes dados, bem como de automatizar a coleta para um tratamento mais imediato e com isso aumentar a eficiência nas previsões de fenômenos da natureza, surgiram as Plataformas de Coleta de Dados Automáticas, que inicialmente utilizavam transmissores de rádio, para comunicação com uma base receptora. No entanto, para regiões vastas, o emprego de transmissores de rádio era inviabilizado pelas potências necessárias para transmitir as informações a uma distância razoável, seja pelo tamanho das antenas ou pela vida útil das baterias que alimentariam estes transmissores.

Com o surgimento dos satélites artificiais passou-se a utilizá-los como "elos" ou "pontes" entre as PCDs e as estações receptoras, podendo então utilizar-se transmissores bem menores e de potências bastante reduzidas.

Essas PCDs têm suporte para até 32 sensores em grupos de 4 através de uma entrada do tipo serial. A transmissão é realizada a uma frequência de 401,65 MHz e sua potência de transmissão é ajustável de 30 a 33 dbm. O período de recorrência, que é o intervalo de tempo entre o início de duas transmissões sucessivas, pode ser programado em passos de 5 segundos até o limite de 500 segundos. Quanto à temperatura de operação para esse equipamento, pode variar de -20°C a +55°C.

Portanto, as modernas PCDs para uso com satélites são equipamentos extremamente eficientes para instalação em locais remotos que se valem de células solares para recarga de suas baterias, e que funcionam da seguinte forma: as informações dos sensores são coletadas e transmitidas pela PCD diretamente ao satélite que as retransmite para a estação receptora, e esta processa os dados e os envia para os usuários da informação através de um dos diversos meios de comunicação existentes (fax, telefone, internet, etc).

## 2.2. Banco de Homologação

O banco de homologação é um sistema formado por vários equipamentos que deve atender a determinados requisitos, como precisão, sensibilidade, *range* etc. A foto do banco existente atualmente no INPE/CRN está mostrada a seguir:



Figura 2: Foto do banco de homologação atual do INPE/CRN

Na imagem temos o conjunto de equipamentos utilizado e à esquerda a câmara térmica onde a PCD é colocada durante os testes. O sistema que foi originalmente montado possuía basicamente a estrutura mostrada na figura abaixo:

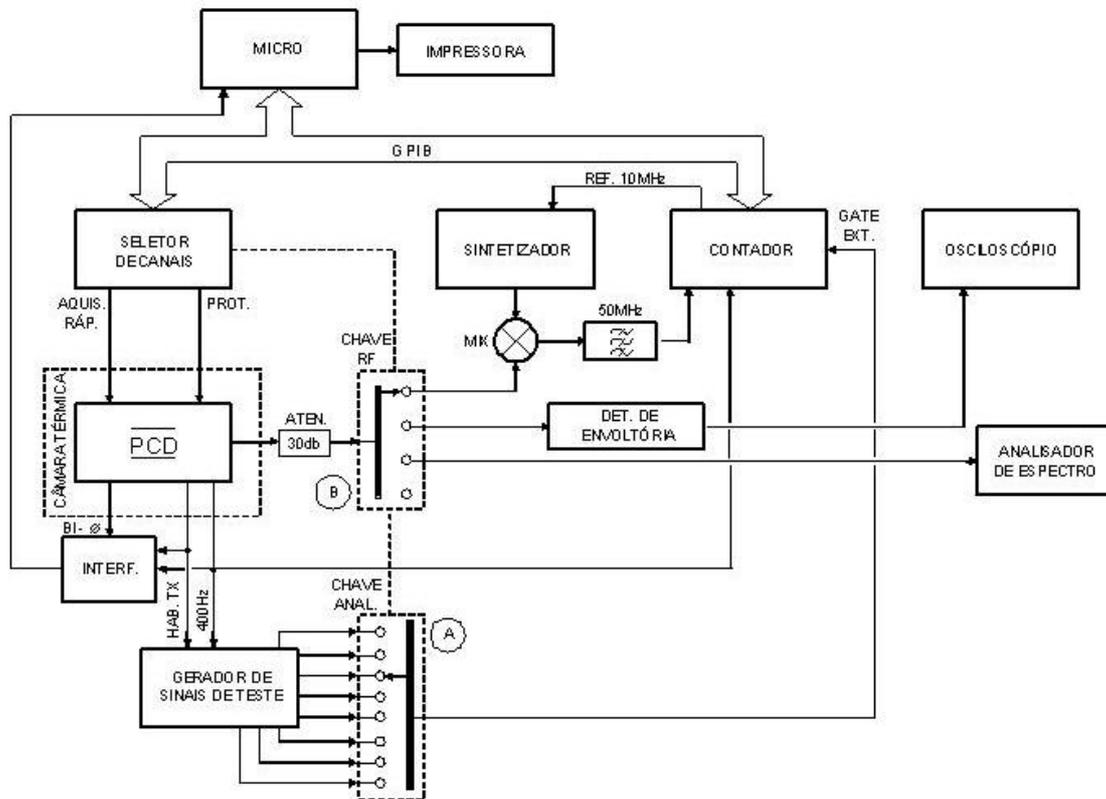


Figura 3: Diagrama esquemático do banco de homologação [2]

A PCD a ser testada é colocada em uma câmara térmica, cuja foto se encontra à direita da figura 2 e está indicado no diagrama da figura 3. Nesta serão simuladas as temperaturas requisitadas no processo de homologação. Os terminais da PCD são ligados em uma interface da câmara que liga os conectores internos à câmara aos externos à mesma. Essa interface serve para levar a alimentação da PCD sob teste e retornar os sinais necessários como RF, Habilita Tx, 400 Hz e BI-φ. O sinal Habilita Tx serve para indicar quando a transmissão está ocorrendo, o 400Hz é a frequência do sinal que está sendo modulado e o BI-φ é a sequência de bits da mensagem que será modulada e transmitida. Os dois primeiros servem para sincronização do banco com os momentos de transmissão.

Esse sistema é formado por um microcomputador que tem a função de fazer os cálculos necessários, como será visto mais à frente, ligado a algum dispositivo de saída como uma impressora, por exemplo. Este micro por sua vez se comunica através de uma

interface GPIB (*General Purpose Interface Bus*), que é um padrão para comunicação de equipamentos, com dois instrumentos que também suportam essa tecnologia, são eles o seletor de canais e o contador de frequência.

A linha tracejada que está ligada ao seletor de canais indica as partes que pertencem ao mesmo, no caso a chave RF e a chave analógica. A primeira tem a função de selecionar que equipamento receberá o sinal RF atenuado que sai da PCD, isso dependerá do teste que está sendo feito em determinado momento. A segunda chave, a analógica, é responsável por demultiplexar um dos sinais de teste que são usados durante o processo.

O gerador de sinais de testes é programado para gerar sinais que representam as janelas de tempo necessárias ao processo de homologação como será mostrado mais adiante. Ele faz uso de alguns sinais da PCD para realizar sua sincronização.

Quando a chave seletora é posicionada na posição mais acima, segundo a figura 3, o sinal RF passa por um *mixer* e é realizado o batimento do sinal com o sinal de frequência 401,70MHz vindo do sintetizador, que por sua vez está sincronizado com o contador conforme visto na figura 3. Após este processo, temos o mesmo sinal com a frequência da portadora abaixada, e então o mesmo passa por um filtro passa-baixas de 50 MHz e em seguida vai para o contador que tem a função de medir a frequência do sinal quando o sinal de *gate* externo for ativado, isto é no momento das janelas, e por último o sinal é enviado via GPIB para o micro.

Caso a chave RF esteja na segunda posição o sinal RF passará por um detector de envoltória e então para um osciloscópio com o intuito de serem verificados parâmetros referentes à potência do sinal transmitido.

Já na terceira posição da chave RF, o sinal é encaminhado a um analisador de espectro para a verificação de grandeza de sinais indesejados na transmissão.

Abaixo do bloco da PCD, no diagrama, temos uma interface que recebe os sinais BI- $\phi$ , Habilita TX e 400Hz, esta passa os dados ao micro para a verificação dos bits a serem modulados e transmitidos.

### 2.2.1. Testes Necessários à Homologação

Visto como se dá o funcionamento esquemático do banco de homologação, seguiremos descrevendo os procedimentos de verificação de desempenho das PCDs durante sua homologação, segundo o sistema ARGOS.

Tais verificações devem ser realizadas em três valores de temperaturas  $-20^{\circ}$ , temperatura ambiente e  $+50^{\circ}$ , sendo esse o papel da câmara térmica anteriormente citada, gerar essas condições de temperatura. Inicialmente os testes são feitos sob temperatura ambiente. Após o sistema do banco de homologação está ligado e a PCD está configurada para a realização dos testes segundo [2], a PCD deve ser ligada e um tempo de uma hora deve ser esperado antes do início das medições para que se tenha a estabilização completa dos circuitos.

A maioria das características a serem avaliadas devem ser medidas em determinados instantes da transmissão, para isso existem algumas janelas de tempo ao longo da mesma que são referenciadas nas medições. A seguir temos o esquema desse padrão.

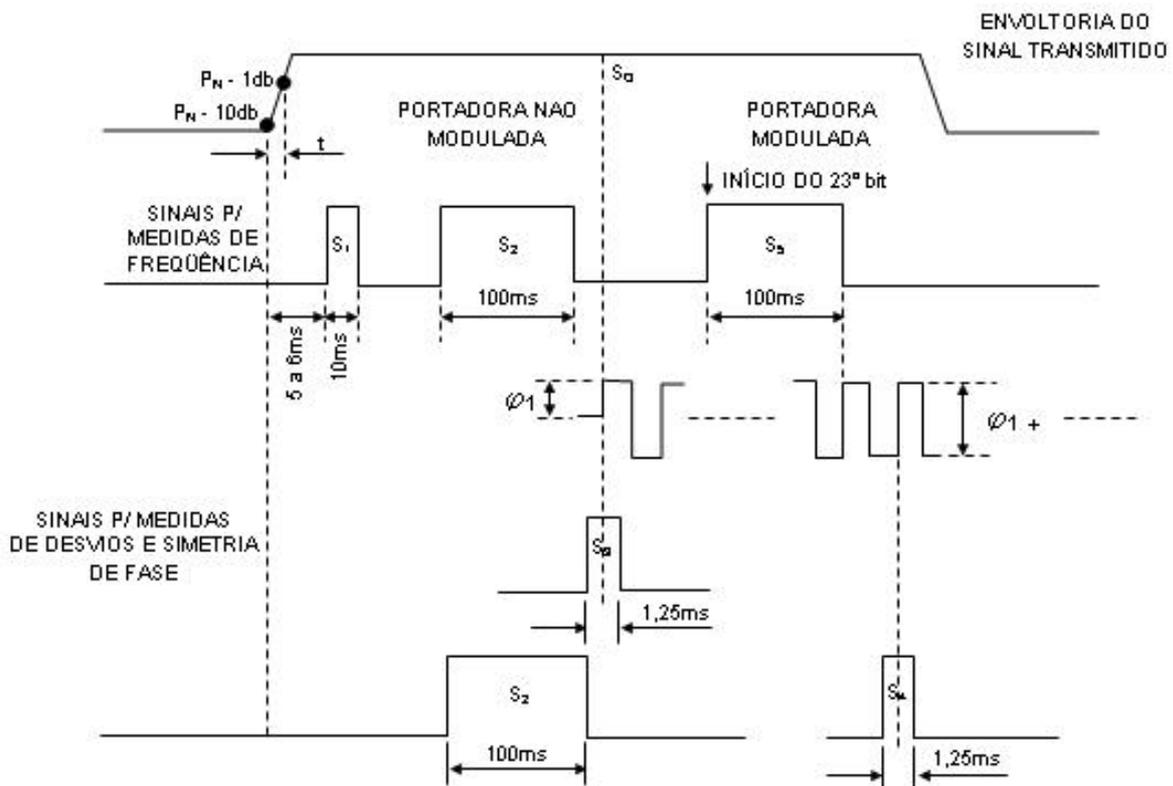


Figura 4: Sinais de teste do banco de homologação

O primeiro grupo de características a ser verificado é a potência de transmissão. Começamos pelo nível de saída do transmissor ou potência nominal, que é a potência do sinal RF que sai no terminal de transmissão. Esse valor deve ser medido na janela  $S_2$ , segundo a figura 4, com uma precisão de  $\pm 0,3\text{dB}$ , mas o valor aceitável depende da antena. Outro teste é sobre a estabilidade da potência, a variação desta não deve

ultrapassar 1dB, a precisão usada é de  $\pm 0,2\text{dB}$  e a o tempo da transmissão é a janela  $S_0$ . Os últimos deste grupo são o tempo de subida e de descida da potência, sendo  $P_n$  a potência nominal, esse tempo é o tempo computado entre os valores de potência  $P_n - 1\text{dB}$  e  $P_n - 10\text{dB}$ , o valor correto para estes é de no máximo 5ms e a medição usa uma precisão de  $\pm 0,2\text{dB}$ .

A partir daqui, assumiremos a seguinte nomenclatura:  $f_i^{(j)}$  é uma medição de frequência que se refere à janela  $j$  de acordo com a figura 4 durante a transmissão  $i$  de um conjunto de transmissões realizadas.

Outra categoria de parâmetros a ser analisado diz respeito às frequências de transmissão. O estabelecimento da frequência é calculado com o desvio médio quadrático entre as janelas  $S_1$  e  $S_2$ , em dez transmissões sucessivas e com precisão de  $\pm 10^{-8}$  e um valor válido para esse parâmetro deve ser menor ou igual a  $10^{-7}$ . A fórmula usada é mostrada a seguir:

$$\sigma_1 = \left\{ \frac{1}{10} \sum_{i=1}^{10} \left( \frac{f_i^{(1)} - f_i^{(2)}}{f_i^{(2)}} \right)^2 \right\}^{\frac{1}{2}} \quad (1)$$

Como próxima característica, temos a frequência média de transmissão que é a média da frequência em  $S_2$  em dez transmissões sucessivas. A precisão deve ser de  $\pm 10^{-7}$ , o limite especificação mínima é de  $401,65\text{MHz} \pm 4\text{kHz}$  e a o modo do cálculo está em 2:

$$f_0 = \overline{f^{(2)}} = \frac{1}{10} \sum_{i=1}^{10} f_i^{(2)} \quad (2)$$

O próximo quesito é a estabilidade da frequência durante a transmissão, que deve usar uma precisão de  $\pm 0,1\text{Hz}$  e seu valor deve ser menor ou igual a 2Hz. Essa medida é feita com a média da diferença entre a janela  $S_2$  e  $S_5$  em dez transmissões sucessivas. Como na equação:

$$\overline{\Delta f_0} = \frac{1}{10} \sum_{i=1}^{10} (f_i^{(2)} - f_i^{(5)}) \quad (3)$$

Ainda nas medições de frequência, temos a estabilidade de frequência em curto termo, que trata da estabilidade desta em um período de 100 ms e essa não deve exceder  $2 \cdot 10^{-9}$ . A precisão usada nesse caso é de  $\pm 10^{-10}$  e esta é medida com o seguinte cálculo:

$$\sigma_2 = \sigma\left(\frac{\Delta f^{(2)}}{f^{(2)}}\right) = \left\{ \frac{1}{20} \sum_{i=1}^{10} \left( \frac{f_i^{(2)} - f_i^{(5)} - \overline{\Delta f_0}}{f_i^{(2)}} \right)^2 \right\}^{\frac{1}{2}} \quad (4)$$

A próxima medição é semelhante à anterior, no entanto o período em avaliação é de 15 minutos e o número de amostras realizadas é **n**. O resultado desse teste é o coeficiente de inclinação de uma reta que não pode ultrapassar a taxa de 2Hz/min. As aferições têm precisão de 0,1Hz. As **n** amostra, na fórmula abaixo, devem ocorrer todas dentro de um intervalo de 20 minutos.

$$a = \frac{\sum_{i=1}^n (t_i - \bar{t}) \cdot (f_i^{(2)} - \overline{f^{(2)}})}{\sum_{i=1}^n (t_i - \bar{t})^2} \quad (5)$$

sendo:

$$\overline{f^{(2)}} = \frac{1}{n} \sum_{i=1}^n f_i^{(2)} \quad \text{e} \quad \bar{t} = \frac{1}{n} \sum_{i=1}^n t_i \quad (6)$$

Se as **n** medidas forem realizadas em intervalos de tempo regulares,  $T_R$ , teremos:

$$t_i = i \times T_R \quad (7)$$

E a fórmula do coeficiente fica simplificada a 8:

$$a = \frac{6}{T_R \cdot n \cdot (n^2 - 1)} \cdot \sum_{i=1}^n f_i^{(2)} \cdot (2 \cdot i - n + 1) \quad (8)$$

Continuando essas medições de estabilidade de frequência temos a medição de estabilidade de longo termo, em que o período avaliado é de duas horas. Essa medição deve usar os valores armazenados do teste de “frequência média de transmissão”, anteriormente calculados, com intervalos entre eles de duas horas.

$$\overline{\Delta f^{(2)}} = \overline{f^{(2)}}(t) - \overline{f^{(2)}}(t + 2h) \quad (9)$$

$$E = \frac{\overline{\Delta f^{(2)}}}{\overline{f^{(2)}}(t)} \quad (10)$$

Um segundo conjunto de testes de performance é sobre o sinal de modulação. Esses são realizados sobre o sinal modulador da transmissão na modulação em fase. Os primeiros parâmetros avaliados são o desvio e a excursão de fase, que testam se o chaveamento da fase está ocorrendo corretamente. Para realizar essas medidas são usados pares de transmissões sucessivas. Na primeira delas, se armazena o valor da frequência  $f^{(3)}$ , na janela  $S_3$ , e da segunda pegam-se as frequências  $f^{(2)}$  e  $f^{(4)}$ , nas janelas  $S_2$  e  $S_4$ , respectivamente. Dados esses valores, são calculados os o desvio de fase ( $\varphi_1$ ) e a excursão de fase ( $\varphi_1 + \varphi_2$ ), da seguinte forma:

$$\varphi_1 = 2\pi\tau_{S_3}|f^{(3)} - f^{(2)}| \quad (11)$$

$$\varphi_1 + \varphi_2 = 2\pi\tau_{S_4}|f^{(4)} - f^{(2)}| \quad (12)$$

Para efeitos de avaliação desses parâmetros, pegam-se dez amostras de cada em dez pares de transmissões sucessivas, e então se calcula a média dos mesmos,  $\overline{\Phi_1}$  e  $\overline{\Phi_1 + \Phi_2}$ , da seguinte maneira:

$$\overline{\Phi_1} = \frac{1}{10} \sum_{i=1}^{10} \varphi_1(i) \quad (13)$$

$$\overline{\Phi_1 + \Phi_2} = \frac{1}{10} \sum_{i=1}^{10} (\varphi_1 + \varphi_2)(i) \quad (14)$$

Esses valores são registrados com precisão de  $\pm 0,01$  radianos. A excursão de fase ( $\overline{\Phi_1 + \Phi_2}$ ) deve ser menor que 2,4 radianos e o desvio de fase ( $\overline{\Phi_1}$ ) estar no intervalo de  $1,1 \pm 0,1$  radianos.

Outro critério relacionado ao sinal de modulação sob análise é a simetria de fase, que como o próprio nome sugere, mede o grau de simetria entre os níveis do sinal modulador. Esse usa os valores de medidos para desvio e excursão de fase, mostrados anteriormente, através da seguinte equação:

$$S = \frac{2 \cdot \overline{\Phi_1} - \overline{(\Phi_1 + \Phi_2)}}{\overline{\Phi_1 + \Phi_2}} \quad (15)$$

Essa medida usa precisão de  $\pm 1\%$  e somente valores menores que 4% são admissíveis.

Por último nesse conjunto de parâmetros temos o tempo de estabelecimento de modulação,  $t_r$ . Este trata do tempo que o sinal de modulação leva para ir de 10 a 90% de sua amplitude total. A precisão da medida deve ser de  $\pm 25\mu\text{s}$  e obedecer a:

$$100 \mu\text{s} < t_r < 250 \mu\text{s}$$

A próxima categoria a ser analisada é sobre o formato e estrutura da mensagem. Para isso o sinal enviado pela PCD precisa ser demodulado e então, retira-se a mensagem binária. A seguir temos os pontos a serem analisados.

- Duração da portadora pura: é o intervalo de tempo,  $T_1$ , entre o início da transmissão e o primeiro bit do sinal modulado. Usa-se uma precisão de  $\pm 0,5\text{ms}$  e o espera-se um valor de  $160\text{ms} \pm 2,5\text{ms}$ .
- Duração da portadora modulada: é o intervalo de tempo,  $T_2$ , que separa o primeiro bit da modulação e o fim da transmissão.
- Duração da transmissão: é o período de tempo total da transmissão, assim:  $T_3 = T_1 + T_2$ . É medido com uma precisão também de  $\pm 0,5\text{ms}$  e os valores admissíveis são de  $360\text{ms} \pm 5\text{ms}$  a  $920\text{ms} \pm 12\text{ms}$  com passos de  $80\text{ms} \pm 1\text{ms}$ , dependendo do número de sensores configurados na programação.
- Período de recorrência: refere-se ao tempo que separa o início de duas transmissões sucessivas com precisão de  $\pm 2\%$  e o valor para a homologação é  $T_R \pm 10\%$ , isto é, aceita-se um valor 10% a mais ou a menos que o que foi configurado.
- Sincronização de bits: verifica a presença de quinze bits “1” no sinal demodulado, a partir do momento d-o início do sinal de modulação.
- Sincronização do formato: confere a presença da seqüência de bits da chamada sincronização de formato que é composta por 8 bits sendo eles da seguinte forma: “00010111”.
- Inicialização: é simplesmente um bit “1” após a sincronização do formato.
- Número de blocos de 32 bits: possui o código binário do número de sensores usados. São quatro bits, sendo um deles para paridade.
- Identificação: Esse campo é codificado em 20 bits, sendo 14 para a identificação da PCD, de acordo com o programado [2], e 6 para verificação de erro nesses bits.

- Frequência de bit: é o cálculo de quantos bits por segundo foram transmitidos na mensagem. A precisão usada é de  $\pm 0,4\text{Hz}$ .

O próximo conjunto de parâmetros a se analisar é o espectro de transmissão. Esta deve ser feita com a PCD em transmissão contínua, para tanto se substitui o módulo PROCESSADOR pelo módulo TESTE [2]. Em seguida, são feitas as medições em duas condições,  $401,65\text{MHz} \pm 80\text{kHz}$  e toda a faixa, das relações entre as raiais parasitas e a portadora. Os valores devem ser registrados com uma precisão de  $\pm 1\text{dB}$ . Os valores esperados são menores que  $-40\text{dB}$  para o primeiro caso e de para toda a faixa são aplicadas as regulamentações do país e internacionais.

O último conjunto de parâmetros para a homologação da PCD é a verificação da proteção contra transmissões contínuas, para isto são forçadas transmissões contínuas da PCD. A homologação exige apenas que exista algum mecanismo de proteção.

### **2.3. *PCI eXtensions for Instrumentation (PXI)***

O PXI é um padrão aberto mantido e promovido pela *PXI Systems Alliance* (PXISA) [3].

A especificação do PXI o define como uma plataforma robusta para instrumentação e automação. O PXI é impulsionado pela alta velocidade do barramento PCI, que é um padrão para desktops, e por projetos de hardware em geral. Com isso o PXI agrega todos os benefícios do PCI numa arquitetura com características mecânicas, elétricas e de software adequadas a ambientes de testes, instrumentação, aquisição de dados e industriais.

A especificação do PXI 2.0 é baseada na especificação do CompactPCI que define novas características ao PCI oferecendo uma maior integridade mecânica além de facilidade de instalação e remoção dos componentes. Essa especificação do PXI exige ainda testes de condições de operação, testes de compatibilidade eletromagnética (EMC) e ventilação, além das especificações mecânicas do CompactPCI, tudo isso visando fácil integração e garantir interoperabilidade de equipamentos de diferentes fabricantes.

O PXI oferece as mesmas características elétricas do PCI inclusive taxas de transferência de dados de  $132\text{MB/s}$  e a funcionalidade Plug-and-Play. Mas o benefício mais marcante que o PXI leva do PCI, talvez seja a importância que o PCI tem no mercado de computadores pessoais (PC) que faz com que existam mais de 800

fornecedores dessa tecnologia e assim uma vasta disponibilidade de hardwares, drivers, firmwares, sistemas operacionais e aplicações de softwares, todos estes podendo ser adaptados aos sistemas PXI com uma boa relação custo/benefício. Em relação ao CompactPCI, o PXI pode oferecer quase de duas vezes mais *slots* de periféricos por segmento, se comparado aos sistemas PCI de desktops. PXI ainda possui também a capacidade de múltiplos-segmentos o que aumenta ainda mais a disponibilidade de entrada e saída desses sistemas. Além disso, o PXI traz características desejáveis em sistemas de instrumentação como barramento de disparo (*trigger bus*) e clock de referência para sincronização de mais de uma placa, barramento *Star Trigger* para temporização precisa, e barramentos locais para comunicação de periféricos adjacentes sem afetar a banda do PCI e com níveis de tensão mais altos. Para exemplificar os modos de comunicação entre os módulos em um chassis PXI, que é onde eles ficam instalados, temos a figura:

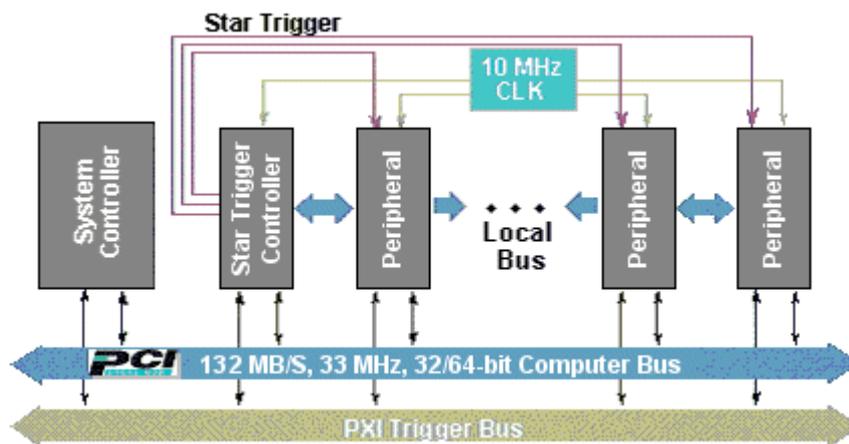


Figura 5: Arquitetura de comunicação entre módulos PXI [3]

O barramento de disparo ou, *trigger bus*, é composto por 8 linhas que possuem um uso bastante flexível. Podem ser usadas para sincronização de módulos periféricos PXI, controle de seqüência de operação de módulos, acontecimento de eventos etc. O atraso nesse barramento é da ordem de dezenas de nano segundos (ns).

O clock de referência tem uma freqüência de 10MHz e chega a todas as placas do sistema, serve para se obter um bom sincronismo na operação dessas, já que esse clock tem uma boa precisão, dando suporte para protocolos de disparo sofisticados. A forma de distribuição desse clock segue uma gama de restrições o que provê um baixo escorregamento (*skew*).

O barramento de *Star Trigger* é um conjunto de linhas dedicadas que saem do *slot 2* do chassis e leva a cada um dos outros módulos. Esse meio de comunicação fornece uma sincronização bastante precisa, devendo ter um atraso máximo, segundo a especificação, de 1ns entre dois sinais enviados por diferentes placas e de 5ns entre os momentos de envio e de recebimento de um sinal.

O barramento local PXI usa a topologia *Daisy-chain* que conecta cada *slot* periférico com os *slots* adjacentes a ele, à esquerda e à direita. Cada barramento local desses tem 13 linhas de largura que pode transmitir sinais analógicos entre as placas e fornece uma comunicação lateral (*side-band*) de alta velocidade, e que não afeta a largura de banda PCI. Os sinais que trafegam no barramento local variam de sinais TTL (5V) de alta velocidade até sinais analógicos de 42Volts. O barramento local do slot mais a esquerda (*slot 2*) é usado para o *Star Trigger*, conforme a figura 5.

O *slot 1* é reservado para o controlador do sistema, ou, no caso de controle via PC, para a interface de comunicação com o PC, como veremos a frente. Outro *slot* que pode ter um uso especial é o 2. Nele pode-se colocar periféricos comuns, como nos outros (com exceção do 1), ou ainda instalar módulos com características de sincronização. Esse geralmente tem uma base de tempo de 10MHz mais precisa que o do chassis, assim existe uma configuração em que essa referência interna ao módulo seja usada pelo barramento do chassis. A placa instalada no *slot 2* é também a única que tem a possibilidade de fazer uso das linhas de *Star Trigger*, desde que esta suporte esse recurso.

A especificação do PXI também define requisitos para o nível de software para *frameworks* padrões como Windows NT e 98, que possuem milhões de usuários e que fazem uso de milhares de aplicações. Isso facilita a integração de sistemas e a compatibilidade entre diferentes fornecedores. Ainda na especificação de softwares, existe a obrigatoriedade, por parte dos fornecedores, de cederem os *drivers* apropriados para seus periféricos PXI eliminando a necessidade de desenvolvimento desses *drivers* por parte do usuário final.

Com as características mecânicas, elétricas e de software do PXI, os desenvolvedores de sistemas de instrumentação que já estejam familiarizados com aplicações projetadas para *desktops* PCs podem direcionar seus esforços para o uso de um sistema PXI com um pequeno aumento de custo. A seguir temos a abrangência do que a especificação PXI define.

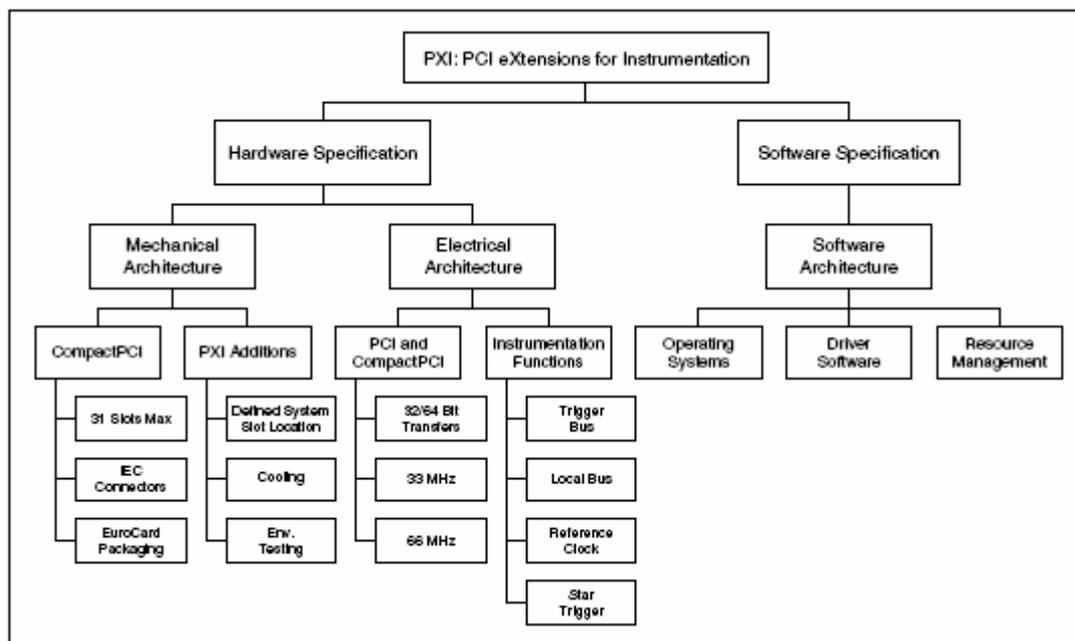


Figura 6: Especificação da arquitetura PXI

Uma vez que a arquitetura PXI é compatível com o barramento PCI, e por sua vez com plataformas IBM-PC que é onde se desenvolve o banco de homologação, optou-se por utilizar essa tecnologia.

Para fazer a aquisição dos dados seria necessário realizar um abaixamento na frequência de saída da PCD, para isso é utilizado um *downconverter*. Dessa forma foi adquirido um módulo analisador de sinais o NI PXI-5660 – *2.7GHz RF Signal Analyzer* da *National Instruments*, esse é composto de dois módulos o NI 5600 *RF Downconverter Module* e NI 5620 *IF Digitizer Module*.

O primeiro deles tem a função de realizar o abaixamento da frequência e então ser possível a amostragem do sinal da PCD com taxas de amostragem menores [10]. Já o segundo módulo tem como função, como o próprio nome sugere, fazer a digitalização do sinal da PCD para em seguida este ser enviado para o computador. Esses módulos podem ser configurados em conjunto ou individualmente. Abaixo estão algumas das principais características do analisador de sinal:

- Faixa de frequência de entrada de 9kHz a 2.7GHz
- 20MHz de largura de banda em tempo real
- Base de tempo de 10MHz
  - $\pm 20$  ppb de estabilidade
  - $\pm 50$  ppb de precisão

- >80dB de range dinâmico livre de espúrios
- Sinal de entrada de -130 a +30 dBm
- Digitalizador de 14 bits de resolução e taxa de amostragem de até 64MS/s
- 32MB de memória onboard

Para efetuar a comunicação entre o barramento PXI e o computador onde os dados seriam analisados, utilizou-se uma interface de comunicação MXI-4 (Measurement eXtensions for Instrumentation), e dentre as combinações possíveis optou-se pelo par NI PCI-PXI8331, também da *National Instruments*, que faz a comunicação entre um PC e o chassis PXI via cabo de cobre. A seguir algumas características dessa interface:

- Alcance de 20 metros com cabo de cobre
- Suporta topologia *Daisy Chain* e estrela
- Taxa de transmissão de 78MB/s sustentado e até 132MB/s de pico
- Checagem de erros e retransmissão
- Suporta padrão PCI 5V e 3.3V

Para a acomodação desses módulos escolheu-se um chassis PXI, tendo como requisito apenas a quantidade de *slots* necessário para acomodar os módulos, que nesse caso seriam cinco, um para o MXI-4, três para o módulo *downconverter* e um para o digitalizador. Assim escolheu-se um chassis de 8 *slots* de propósito geral o NI PXI-1042 que é do mesmo fabricante dos outros componentes. As principais características desse equipamento são:

- Trabalha com módulos PXI 3U e CompactPCI
- Inibição de operação e monitoramento da voltagem remotos
- Referência de 10MHz com exatidão de  $\pm 25$  ppm (partes por milhão)
- Fonte de alimentação AC removível de alta performance

Na figura 7, temos a imagem do *chassis* com todos os módulos acima comentados:



Figura 7: Chassis PXI e módulos usados

Os módulos presentes são da esquerda para a direita: o MXI-4, PXI-5600 e PXI-5620.

#### 2.4. LabView

O LabView é um ambiente de programação gráfica, a sua linguagem de programação é conhecida como linguagem G. Esse ambiente é formado por duas interfaces, o painel frontal, ou *Front Panel*, e o diagrama de blocos, ou *Block Diagram*. No painel frontal, são inseridos elementos de interface com o usuário, como botões, campos de texto, termômetros etc. Para cada elemento neste painel, há um correspondente no diagrama de blocos. Já no diagrama de blocos, tem-se a programação propriamente dita na linguagem G. Na figura 8, temos um exemplo de painel frontal (a) e de um diagrama de blocos (b).

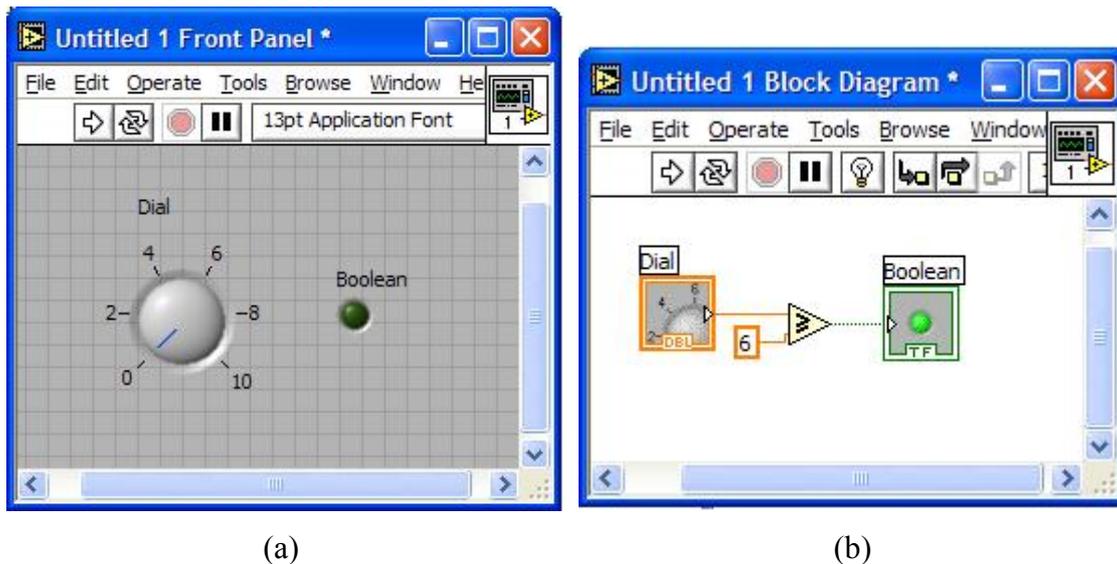


Figura 8: (a) Painel Frontal e (b) Diagrama de Blocos de um programa em LabView

Um programa na linguagem G é construído com a união de blocos por “fios” (*wires*). Os blocos em questão são operações básicas como operações aritméticas, comparações etc, podem ser elementos presentes no painel frontal, ou ainda sub-VIs (*Virtual Instrument*) que são blocos que realizam tarefas mais complexas, usando outros blocos mais simples e/ou chamadas à *drivers*. Os fios têm a função de ligar esses blocos e suas cores dependem do tipo de dado que aquele fio transporta, laranja para número de ponto flutuante, azul para inteiros, verde para booleanos e assim por diante.

O fluxo de execução do programa nessa linguagem segue da esquerda para a direita e cada bloco só será executado quando todos os fios que estão em suas entradas estiverem com os dados disponíveis, até que todos os blocos tenham sido executados. Através do painel frontal, os dados são recebidos e enviados pelo programa para o usuário.

Para o acesso a equipamentos externos, no nosso caso os módulos PXI, são fornecidos juntos com seus *drivers* conjuntos de sub-VIs que utilizam esses *drivers*. A seguir serão citados os conjuntos referentes aos módulos PXI usados.

#### 2.4.1. ni5660

Visto que o analisador de sinais (NI PIX-5660) é composto por dois módulos, o digitalizador (NI PXI-5600) e o *downconverter* (NI PXI-5620), esse produto acompanha um conjunto de VIs que são usados para a configuração e uso desses módulos conjuntamente. Por exemplo, um sub-VI de inicialização do ni5660 é formado pelo

sub-VI de inicialização individual do digitalizador ligado ao do *downconverter*. A figura 9 mostra alguns desses VIs.

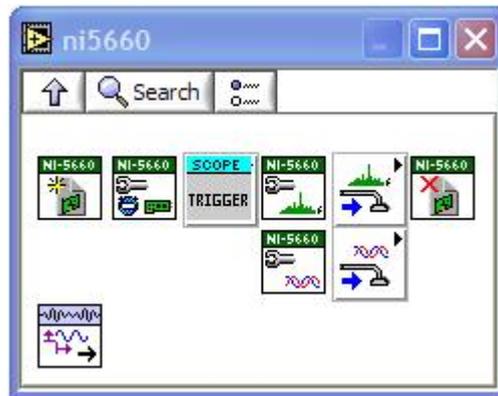


Figura 9: Diretório com VIs do ni5660

#### 2.4.2. NiTuner

Um dos pacotes do qual o ni5660 faz uso é o NiTuner, este contém VIs que fazem a configuração do módulo *downconverter* (NI PXI-5600). Lá estão presentes blocos para inicialização e finalização do dispositivo, configuração de atenuação, configuração de *clocks*, etc. Além de serem usados no ni5660 estes VIs podem ser colocados num programa diretamente. Abaixo eles estão mostrados na figura 10.

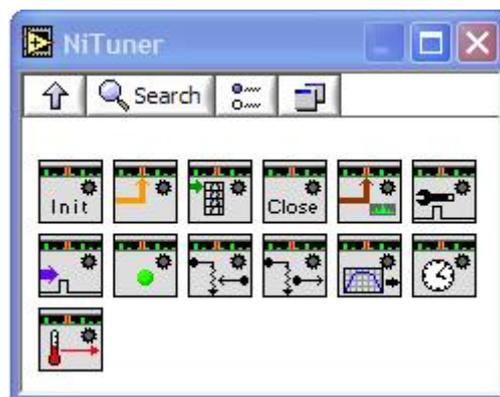


Figura 10: Diretório com VIs do NiTuner

#### 2.4.3. NI-SCOPE

O segundo conjunto de VIs utilizados no ni5660 é o NI-SCOPE. Estes são responsáveis pelo acesso aos *drivers* para a placa PXI digitalizadora (NI PXI-5620). Esse dispositivo apresenta muito mais opções de configuração e modos de uso que o

*downconverter*. Na figura 11, vemos janela com os blocos e subpastas para os modelos de digitalizadores.



Figura 11: Diretório com VIs do NI-SCOPE

Na parte de configuração, teremos a configuração vertical, responsável por *range* de entrada do dispositivo, para uma melhor quantização dos dados, se o sinal de entrada possui nível DC, etc. O VI de configuração horizontal permite a escolha de taxa de aquisição dos dados, quantidade de amostras por registro, o número de registros a se adquirir dentre outros. Existe também o VI de configuração automática que através da análise do sinal de entrada autoconfigura o digitalizador para gravar aquele sinal. Na figura a seguir temos esses VIs mostrados.

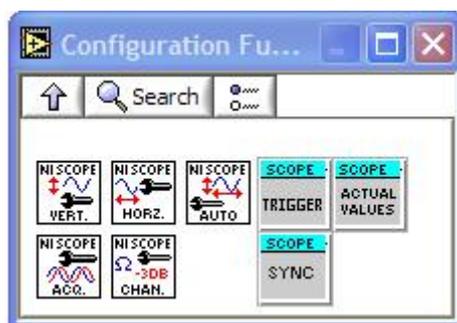


Figura 12: Diretório com VIs de configuração do NI-SCOPE

Outro diretório do NI-SCOPE que vale ser ressaltado é o de funções de aquisição que são mostradas a seguir.

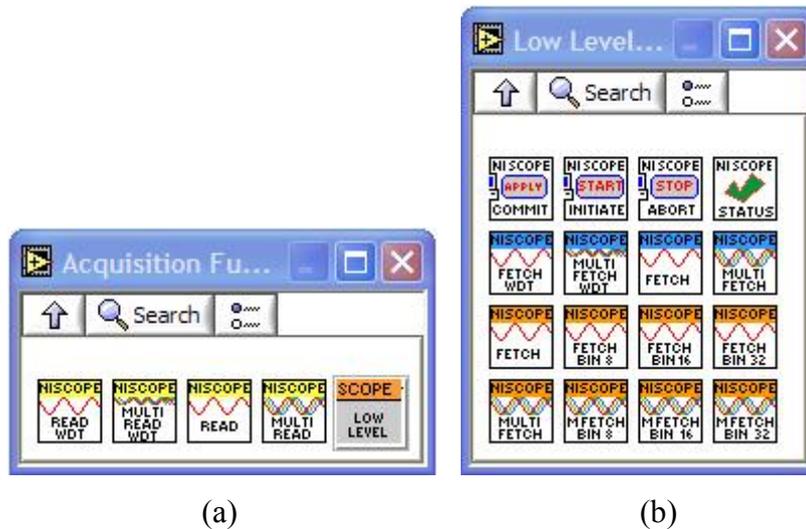


Figura 13: Diretórios com VIs de aquisição do NI-SCOPE, (a) em alto nível e (b) baixo nível.

As funções de aquisição, na figura 13(a), fazem uso das funções de baixo nível, contidas em 13(b). Uma digitalização é iniciada no momento em que o fluxo de execução atinge o bloco de baixo nível `niScope Initiate Acquisition.vi`, a partir daí o digitalizador começa a digitalizar e armazenar os dados em sua memória *onboard*. Como está implementada em forma de um *buffer* circular, após algum tempo na memória, o dado é sobrescrito e não poderá mais ser recuperado. Quando a execução atinge um VI com a função de *fetch*, isto é, pegar os dados, ele espera por um determinado *trigger* ou disparo, cujo tipo e fonte dependem da configuração feita anteriormente. Quando esse disparo é dado, os dados são enviados ao controlador ou PC via o barramento PCI, na forma com que o VI de *fetch* trabalha, ou seja, com o tamanho e formato determinado pelo VI, 8, 16 ou 32 bits etc.

Como último ponto indispensável a se comentar sobre o NI-SCOPE, temos os, já citados, *triggers* ou disparos. Conforme já dito estes têm a função de enviar o sinal para que aos dados adquiridos sejam enviados ao controlador do sistema. Existem vários tipos destes, como sinais elétricos ou via software. Os de sinais físicos precisam de uma configuração que indique de que fonte esse sinal virá, se por uma linha de *Star Trigger*, pelo barramento de *Trigger*, por um canal de aquisição etc. Abaixo temos esses VIs para configuração dos disparos.

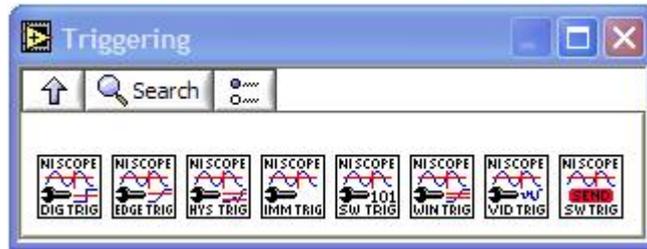


Figura 14: Diretório com VIs de configuração de triggers do NI-SCOPE

É importante salientar que nem todas as características configuradas pelos VIs do NI-SCOPE são necessariamente suportadas pelo *driver* de um determinado modelo de digitalizador.

#### 2.4.4. Spectral Measurement Toolkit

Por último dos pacotes de VIs fornecidos para o LabView, temos um *kit* de ferramentas para medições dos sinais adquiridos. Esse conjunto possui funções para medições de características do espectro de sinais como potências, busca por frequências de maior contribuição etc.

## CAPÍTULO 3 - ARQUITETURA PROPOSTA PARA O BANCO DE HOMOLOGAÇÃO

Dada a viabilidade do sistema, o diagrama abaixo mostra como será a estrutura do novo banco de homologação do INPE/CRN.

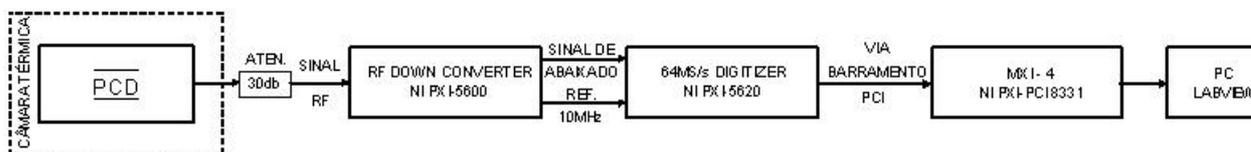


Figura 15: Diagrama esquemático do banco de homologação proposto

Primeiramente, vale ressaltar a simplicidade do novo projeto comparativamente ao anterior. O funcionamento se dá da seguinte forma, a PCD é colocada na câmara térmica do mesmo modo que no sistema anterior, inclusive usando a mesma câmara, que é a única parte do sistema antigo a ser reutilizada, e os sinais da plataforma são colocados para fora da câmara via a interface já comentada. O sinal enviado pela plataforma passa por um atenuador de 30dB e o sinal vai para o módulo *downconverter*, a frequência da portadora do sinal é abaixada para 15 MHz e então o novo sinal enviado para o digitalizador. Além do sinal a se analisar, o *clock* de referência é compartilhado entre os dois módulos. Via o *backplane* ou barramento PCI do chassis, os sinais são enviados para o módulos MXI-4 que, por sua vez, se comunicaria com o PC que estaria executando um programa em LabView.

No PC, devem ser analisados todos os parâmetros necessários à homologação do transmissor. Assim o software deve ser responsável pela aquisição dos dados, abertura de janelas, medições de valores e por fim testar se estes satisfazem as especificações. Abaixo temos a imagem do conjunto a ser utilizado nessa nova proposta:



Figura 16: Foto dos equipamentos a serem usados no novo banco de homologação

## CAPÍTULO 4 - RESULTADOS

### 4.1. Instalação dos Equipamentos

Com a chegada dos equipamentos, módulos e chassis PXI e LabView, uma vez que os módulos já vieram montados no chassis, deu-se uma etapa de configuração de software no PC, com instalação do ambiente de programação LabView 7.1, *drivers* e configuração dos mesmos para os equipamentos a serem usados. Esse processo de instalação está mostrado no apêndice A desse documento.

### 4.2. Testes Iniciais

Como forma de familiarização com o equipamento e seus VIs, em paralelo com o estudo dos blocos, houve o desenvolvimento de vários programas que realizavam a exibição da FFT de um sinal de entrada, programas de aquisição simples mostrando o sinal adquirido, entre outros, sempre seguindo exemplos disponíveis e os VIs do `ni5660`.

O sinal utilizado com entrada era de um gerador de sinais capaz de produzir senóides com em frequências de 100kHz a 3,2GHz. Um detalhe percebido durante os testes iniciais foi que quando a frequência de entrada no *downconverter* era abaixo de 15 MHz, o que ocorria era um *upconversion*, visto que esse módulo sempre centra a frequência da portadora para 15 MHz. Assim devido a algumas saturações nos circuitos desse módulo, o sinal de saída tem um nível de potência alto o suficiente para que quando o fluxo do programa chegue até os VIs do digitalizador seja gerado o seguinte erro **“One or more channels has an ADC overload.”**. A solução para estes casos foi trabalhar com um sinal de entrada na faixa de -15dBm.

### 4.3. NiTuner e NI-SCOPE

Ao final dos testes iniciais, não se conseguiu um resultado satisfatório, isto é, a recuperação do sinal amostrado no tempo. Como os VIs do `ni5660` são para tarefas mais específicas como exibição de FFT, modulação e demodulação etc, então mudamos o enfoque para um nível mais baixo e passamos a utilização dos VIs do NiTuner e NI-SCOPE, para obtermos assim maior flexibilidade na programação dos periféricos.

#### 4.4. Sincronismo com a PCD

Como mostrado anteriormente, o banco de homologação percebia o início da transmissão através de alguns sinais da PCD diferentes do sinal RF. E a nova proposta devido à evolução desses transmissores é que esses sinais não fossem mais usados. Para isso, necessitava-se de um mecanismo que percebesse o início de uma transmissão para só então começar o processo de homologação.

Para tal função poderia se usado o *edge trigger* que espera em uma entrada previamente configurada que um sinal ultrapasse determinado nível e quando esse evento ocorrer o disparo é detectado. Como o *driver* em uso era o que acompanhava o produto o NI-SCOPE 2.6.1, e em sua documentação constava que a placa digitalizadora em uso, NI PXI-5620, não suportava *edge trigger*, tentamos implementar esse mecanismo nos mesmos. Para tanto, pensou-se em fazer uma aquisição contínua que pegava porções de amostras e verificava se alguma delas havia ultrapassado o limiar e quando isso ocorresse seria feita a aquisição por um tempo equivalente a uma transmissão, por exemplo. Esse método foi tentado por falta de opção já que como se estava usando elementos de alto nível na implementação, provavelmente, alguns atrasos inviabilizariam o processo.

Antes do fim dessa solução proposta descobrimos através de fóruns na Internet, que existia um novo *driver* que dava suporte ao *edge trigger* no equipamento usado, era a versão 2.7 do NI-SCOPE. Com essa nova versão seria possível usar tal disparo, porém com uma limitação, a frequência de amostragem usada teria que ser a máxima do dispositivo, no nosso caso 64MS/s.

#### 4.5. Problemas com Espaço

Uma transmissão de uma PCD pode ter duração de 360ms a 920ms, dependendo do número de sensores nela colocados. Como nosso programa permite que se configure quanto tempo a aquisição deve durar, testamos se era possível adquirir os 920ms referentes a transmissão de maior duração. Verificou-se que este não foi possível, devido à limitação da memória no digitalizador em 64MB, uma vez que a taxa de aquisição de 64MS/s não poderia ser modificada e que cada amostra ocupava 16 bits. Vale salientar que o erro não era dado em tempo de aquisição, mas sim antes que a mesma começasse, este já era previsto.

Tentamos de várias formas contornar este problema. Primeiro colocou-se, ao invés de usar o VI de aquisição do NI-SCOPE *read*, dado que este poderia exigir mais

memória, um VI de baixo nível *fetch* de 8 bits, mesmo que depois da aquisição seja necessário alguns cálculos para colocar esse resultado em valores corretos, no nosso caso em volts. O problema não foi resolvido, a conclusão a que chegamos é que o dado de 8 bits refere-se ao tamanho do dado ao ser transferido para o controlador (PC) e não como o digitalizador armazena o dado.

Outra solução proposta habilitar a opção de pegar mais dados que a memória seja capaz de suportar, seguindo um exemplo que acompanha os VIs. Nesse caso tínhamos que passar os dados adquiridos em registro de  $n$  amostras. Vários valores de  $n$  e de registros foram testados, mas quando se tratavam de valores muito altos, sempre ocorria o erro de *buffer*, indicando que algum dado tinha sido sobrescrito antes de ser enviado para o controlador, isso ocorreu pelo uso do *buffer* circular. Esse problema ocorreu mesmo em se tratado dos VIs de baixo nível, *fetch*, anteriormente usados.

Como não se conseguiu achar uma solução e, para números de sensores menores, era possível adquirir todos os dados, já que conseguimos fazer aquisições de até aproximadamente 515ms, assumiu-se essa condição como satisfatória.

#### **4.6. Aquisição Completa**

Neste ponto em que estávamos, já era possível adquirir uma transmissão a partir do momento em que ela começasse e por um determinado tempo limitado, porém satisfatório. Até então a aquisição estava sendo feita somente usando o digitalizador, era necessário agora inserir a parte do uso do módulo *downconverter*. E assim o fizemos. Por considerar esse uma parte de fundamental importância na construção do programa final, mostraremos com um pouco mais de detalhes esta.

Primeiro temos a parte do NI-SCOPE.

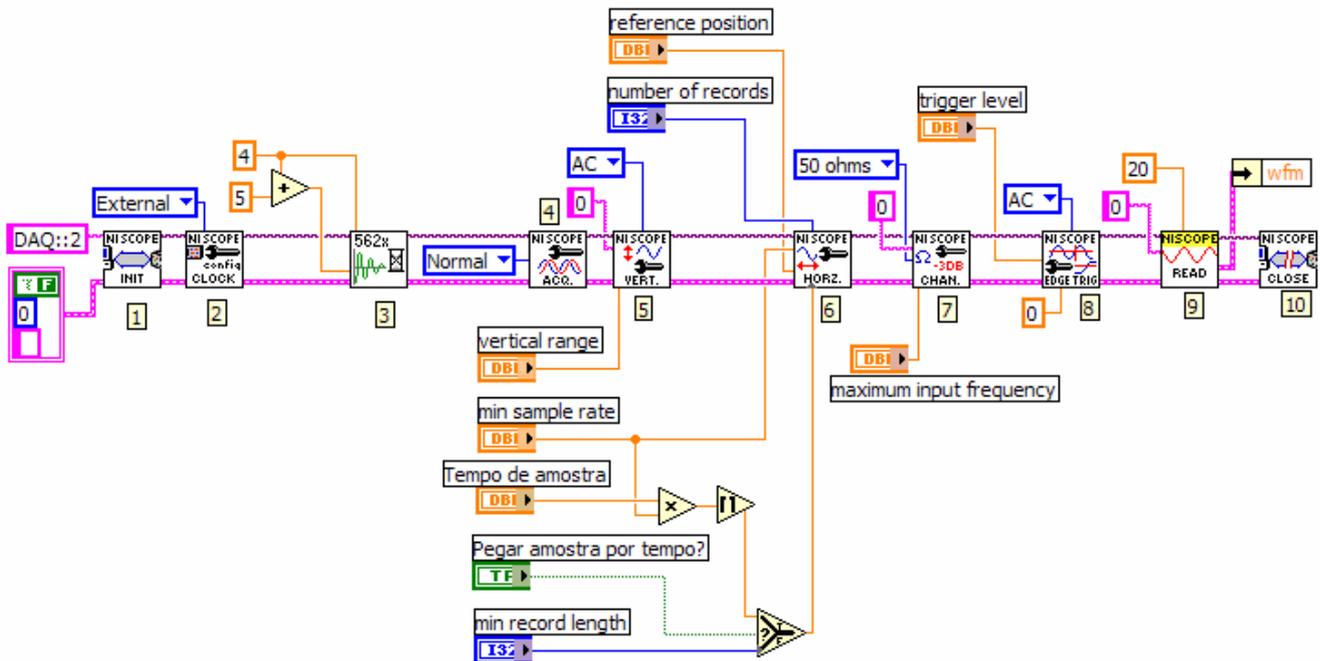


Figura 17: Parte do software de aquisição de dados da parte do digitalizador

Falaremos dos VIs na ordem em que aparecem numerados na figura 17.

1. É o bloco de inicialização, cria uma nova sessão para o instrumento dado este que deve passar por todos os VI do NI-SCOPE que farão uso da placa.
2. Configura o *clock* da placa para receber uma referência externa, nesse caso do *downconverter*, pois ele possui uma base de tempo mais precisa que a do chassis.
3. Serve para esperar que a referência da placa entre em sincronismo com a externa. Também conhecido como *PLL Lock*. Esse bloco foi acrescentado depois, mais a frente isso será comentado.
4. Configura a aquisição como normal, em oposição ao modo DDC (*digital downconverter*).
5. Diz que canal do digitalizador está sendo usado (canal 0, o único), o acoplamento realizado, AC, e o valor do *range* de amplitude do sinal que é configurado no painel frontal. Vale lembrar que o NI PX-5620 só suporta um *range* máximo de 2V e o acoplamento AC.
6. Faz a configuração dos chamados parâmetros horizontais da placa. O valor da posição de referência é o valor que indica, dado que ocorreu o *trigger* num momento, quantos por cento das amostras serão pegadas antes do momento do

disparo e as restantes são pegas depois do mesmo. O número de registros (*records*) sempre é usado em 1 visto que os dados a se adquirir estão continuamente localizados. A taxa de amostragem (*sample rate*) também aparece como entrada para este bloco, devido a problemas já comentados sempre ela é usada em 64MS/s. A outra entrada configura quantos amostras devem ser adquiridas, mas colocou-se uma opção de se informar quantas amostras se deseja ou quanto tempo de aquisição se quer.

7. Faz configurações do canal de aquisição, escolhe-se o canal a se configurar, nesse caso o 0, coloca-se a impedância de entrada e a frequência máxima de entrada, para a configuração do filtro interno.
8. Este VI indica que o módulo esperará por um *edge trigger*, e o configura para esperá-lo no canal 0 (não está mostrado, mas é a opção *default*), *delay* igual a zero e acoplamento AC com o *trigger*.
9. Esse é o VI de leitura, como seu nome já sugere, aí estão sendo colocados o número do canal a onde será feita a aquisição e o tempo de *timeout*, para esperar pelo disparo antes que ele reporte erro. Na saída temos um fio com um tipo de dado do LabView chamado *waveform*.
10. Por último, temos o VI que finaliza a sessão para o instrumento.

Agora temos a seqüência de configuração do *downconverter*.

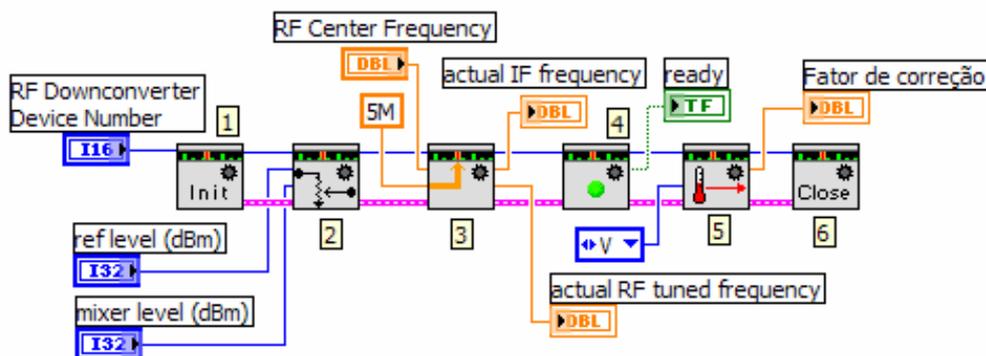


Figura 18: Parte do software de aquisição de dados da parte do *downconverter*

1. Da mesma forma que no caso anterior, este é o responsável por abrir a sessão do equipamento recebendo simplesmente o número atribuído ao módulo *downconverter* no programa de configuração (MAX – *Measurement and Automation eXplorer*).

2. Esse VI é responsável pela configuração dos atenuadores dessa placa. São indicados o nível de referência e o nível de tensão do *mixer*, com esses valores os atenuadores internos são configurados. Esses valores devem obedecer à certos limites, e se o primeiro for colocado com um valor acima de 1000, a opção de autoatenuação é ativada.
3. Configura a frequência central da portadora do sinal de entrada e o *span* desejado. Essa frequência escolhida como central, ou uma aproximação dela, é a que será abaixada para 15MHz pelo processo.
4. Esse bloco tem a função de indicar se os circuitos estão suficientemente sincronizado com a frequência desejada e todos os PLLs foram feitos.
5. É responsável por gerar um fator de correção para a onda resultante do processo de abaixamento da frequência de acordo com a temperatura do sistema. O fator de correção deve multiplicar cada amostra da onda resultante.
6. Serve para encerrar a sessão do *downconverter*.

Além dessas partes o programa possui outras partes para exibição de valores, interface com o usuário, que não serão detalhadas.

#### **4.7. Gravação e Leitura em Arquivo**

Visto que a quantidade de dados a ser manipulado nesse projeto é bastante grande, devido ao alto valor para taxa de amostragem, optou-se por armazenar os dados adquiridos num arquivo antes de fazer seu processamento e possivelmente recuperar somente uma a parte de dados a ser analisada num determinado momento, evitando assim a sobrecarga desnecessária do sistema PC.

Existem três modos principais de se gravar dados em um arquivo, são eles texto, *datalog* ou binário. O modo texto é o mais dispendioso e é útil quando se quer usar os dados gravados em outras aplicações como no Microsoft Excel, por exemplo. Então tentamos usar o modo *datalog* que grava um tipo de dado do LabView que se queira, como uma *waveform*, em um arquivo. O funcionamento foi satisfatório até que tentamos gravar uma quantidade grande de dados, pois o PC acusou falta de memória mesmo quando a memória virtual foi colocada a valores por volta de 2GB. Tentamos ainda fragmentar a *waveform* em várias e gravar um arquivo com várias *waveforms*, mas o problema persistiu.

A solução foi passar para a gravação em arquivo binário, que apesar de mais complicada é bem mais eficiente que as outras em termos de velocidade e de espaço ocupado. Nesse caso sim tivemos sucesso. Na figura a seguir, temos a seqüência para a configuração dessa gravação.

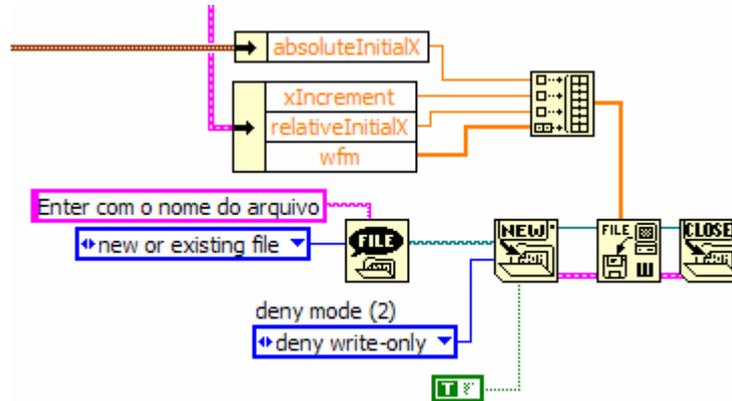


Figura 19: Parte do software de aquisição de dados responsável pela gravação dos dados em arquivo

A seqüência de operações é simples nesse caso, primeiro um bloco responsável por exibir um *prompt* para escolha do nome do arquivo é executado. Em seguida, o mesmo é aberto ou criado, o dado, nesse caso um vetor de *double* é gravado em tal arquivo e por último o arquivo é fechado.

Foi desenvolvido um outro programa para a leitura e processamento dos dados, para que fosse possível fechar a aplicação que adquire os dados, pois isto usa muita memória o que deixa o processamento mais lento. A estrutura da leitura lembra a da escrita como vemos abaixo.

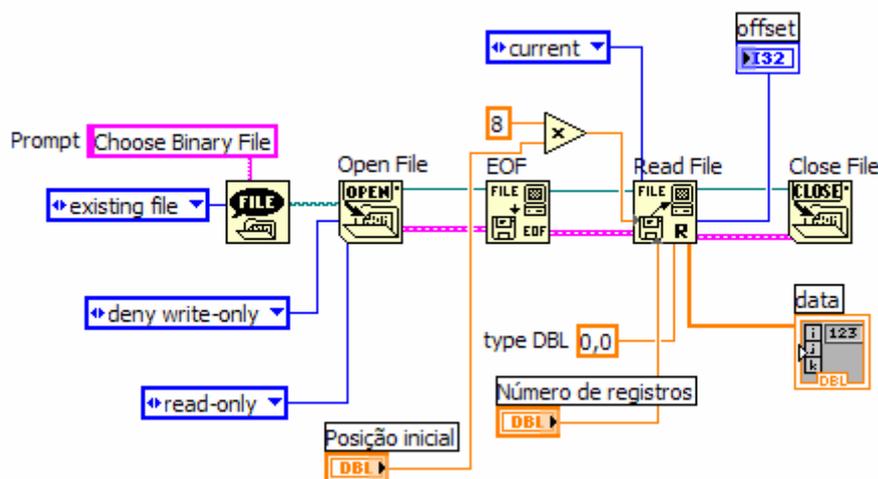


Figura 20: Parte do software responsável pela leitura dos dados no arquivo

Escolhe-se o arquivo através do *prompt* e o mesmo é aberto em seguida. É colocado um caracter de EOF (*end-of-file*) no arquivo aberto, nesse caso foram deixados os valores padrões e o mesmo foi colocado no fim do arquivo. Este serve como indicação de onde está o fim do arquivo. O bloco de leitura recebe a posição inicial do arquivo onde os dados começam a ser recuperados, nesse caso, graças a opção mostrada *current*, o a posição inicial refere-se a o ponto onde a marca do arquivo se encontra, como ele já foi aberto essa marca está no início da área de dados. É indicado também nesse bloco, o número de registros a serem recuperados e o tipo do vetor a ser recebido. O dado é recuperado e então o arquivo é fechado.

Apesar de na imagem existir um bloco de controle a posição inicial e número de registros, esses não são colocados diretamente pelo usuário no nosso programa, eles são calculados dado o intervalo de tempo do sinal transmitido o qual se deseja analisar.

Esse mecanismo é importante por duas coisas, primeiro somente parte do dado armazenado no arquivo é carregado para a memória o que exige menos recursos do PC e é possível fazer a seleção das janelas que devem ser abertas no processo de homologação como mostrado anteriormente.

#### 4.8. Medições de Parâmetros

Dado que a aquisição dos dados já poderia ser feita, passamos a testar os modos de realizar medições na *waveform* adquirida.

Durante esse processo estávamos obtendo erros na medição de frequência que não eram aceitáveis, percebeu-se então que não estávamos esperando que a referência do *downconverter*, que é usado como fonte de referência externa para o digitalizador, casasse com a deste, e assim eram obtidos resultados errados. Tudo se resolveu com o uso do bloco 3 da figura 17, já que este faz com que o fluxo espere que o casamento das referências ocorra.

Prosseguindo com a medição da frequência, entre outros blocos com essa função, achamos um que tem essa função mostrado na figura 21:



Figura 21: *Extract single tone information.vi*

Este recebe um tipo de dado *waveform* construída com elementos da paleta de *waveforms*, assim a partir do vetor recebido deve ser feito um dado deste tipo. Informa-se ainda uma região em que a frequência da portadora deverá estar contida, quanto menor essa região mais rápida se dá a busca. Como retorno temos a frequência da portadora. Os resultados obtidos foram próximos por volta de 16,649.997MHz para uma entrada de 401,65MHz. Lembrando que como indicado quando executado o programa de aquisição, a frequência de 400MHz é abaixada para 15MHz e a frequência usada 401,65MHz aparece como 16,65MHz.

Por último, foram feitos alguns programas para o cálculo da potência do sinal adquirido, porém os resultados não foram muito expressivos.

## **CAPÍTULO 5 - CONCLUSÃO**

Dada a importância das PCDs para a previsão dados meteorológicos, e, conseqüentemente, para o INPE, e os requisitos para a homologação dos transmissores tecnologicamente mais evoluídos, o desenvolvimento do software do banco de homologação tem um valor bastante representativo.

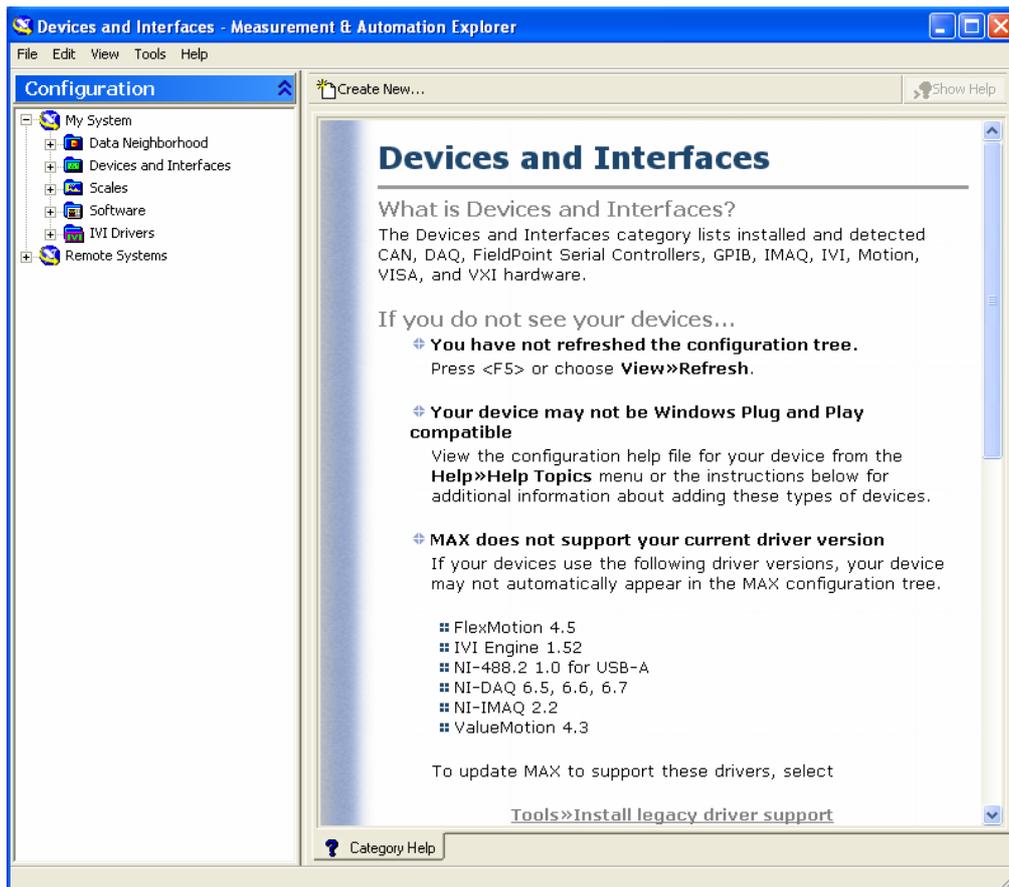
Portanto, o trabalho, até aqui desenvolvido, necessita de uma continuação para que se conclua o mesmo, sejam realizados os devidos testes no seu funcionamento, possível resolução de alguns problemas encontrados, para que, posteriormente, o banco de homologação atual possa ser substituído por outro com a nova arquitetura proposta.

## APÊNDICE A – Instalação e Configuração dos Equipamentos

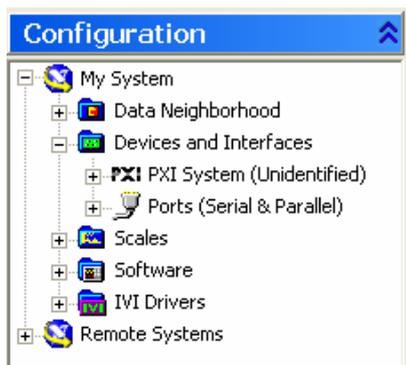
1. Instalação do ambiente de programação LabView 7.1 completo.
  - Durante a instalação são solicitados os CDs contendo os *drivers* para o LabView, estes são DEVICE DRIVER REFERENCE CD 1 e 2.
  - Quando requisitado pelos *drivers* a serem instalados, acrescenta-se a opção MODULAR INSTRUMENTS.

OBS.: Após essa etapa, os CDs que acompanham o NI MXI-4 e o chassis NI PXI-1042 não possuem mais nenhum software que ainda não tenha sido instalado com uma versão mais atualizada que os que foram instalados na etapa 1.

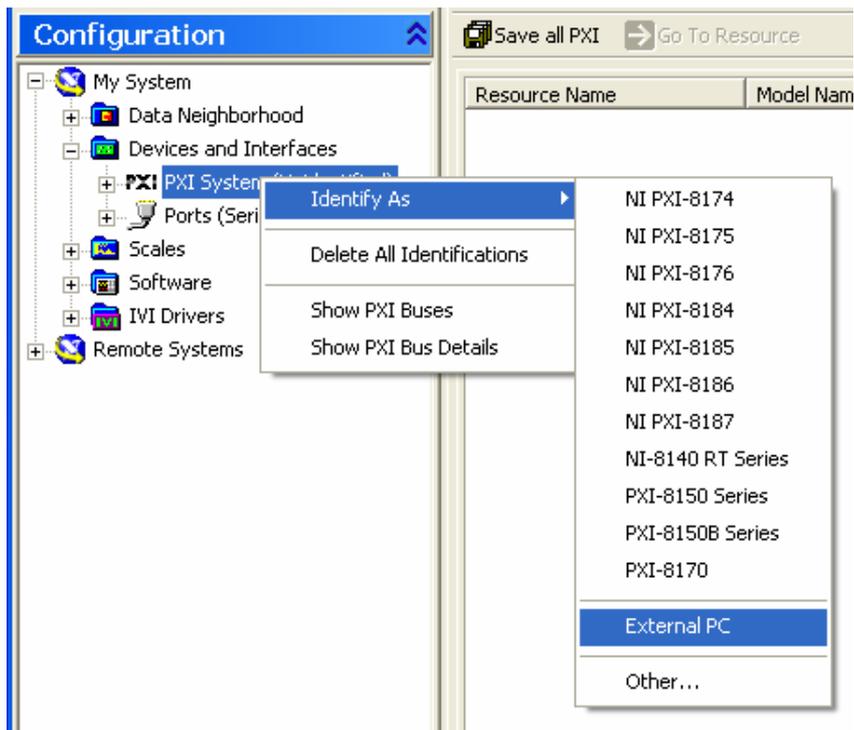
2. Instalação do NI-RFSA CD. São dois CDs que contêm os *drivers* para os módulos do analisador de sinais.
  - Nos softwares a serem instalados com esse CD, está o *Measurement and Automation eXplorer (MAX) 3.1*, que deve ser instalado, dado que a versão que acompanha o LabView é mais desatualizada.
3. Instalação da versão mais nova do NI-SCOPE a 2.7, pois esta possui suporte a características que serão necessárias ao desenvolvimento do software do banco de homologação. Esse *driver* pode ser conseguido no site na *National Instruments* [11] e também estão gravados em CDs internamente no INPE/CRN.
4. Instalação do kit de ferramentas *Spectral Measurements Toolkit*.
5. Em seguida, coloca-se a placa PCI do par de placas do MXI-4 num *slot* vazio do PC e liga-se o computador. Quando iniciar o Windows XP ele deverá reconhecer um novo hardware automaticamente, então mande-se procurar o respectivo *driver*.
6. Executa-se o MAX versão 3.1.



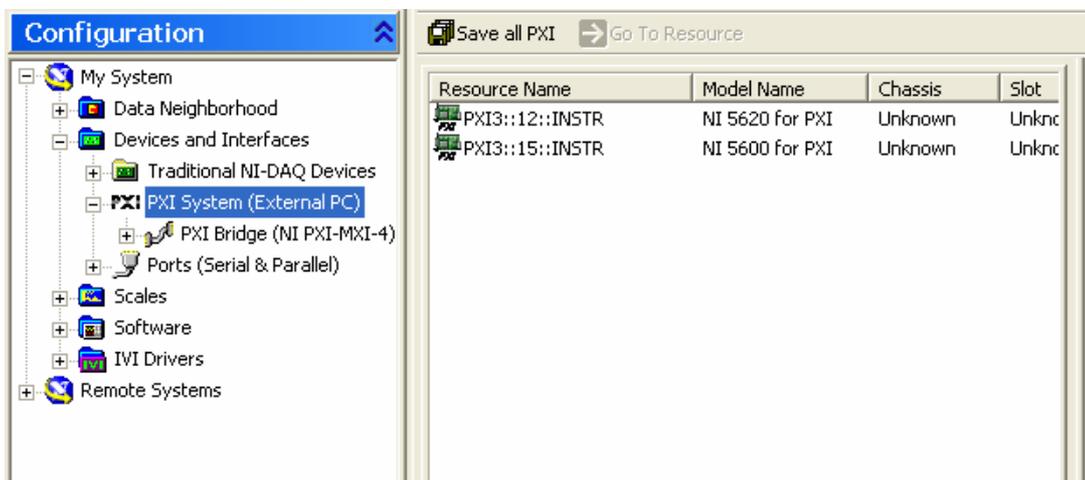
7. Em seguida, dá um duplo *click* em **Devices and Interfaces**.



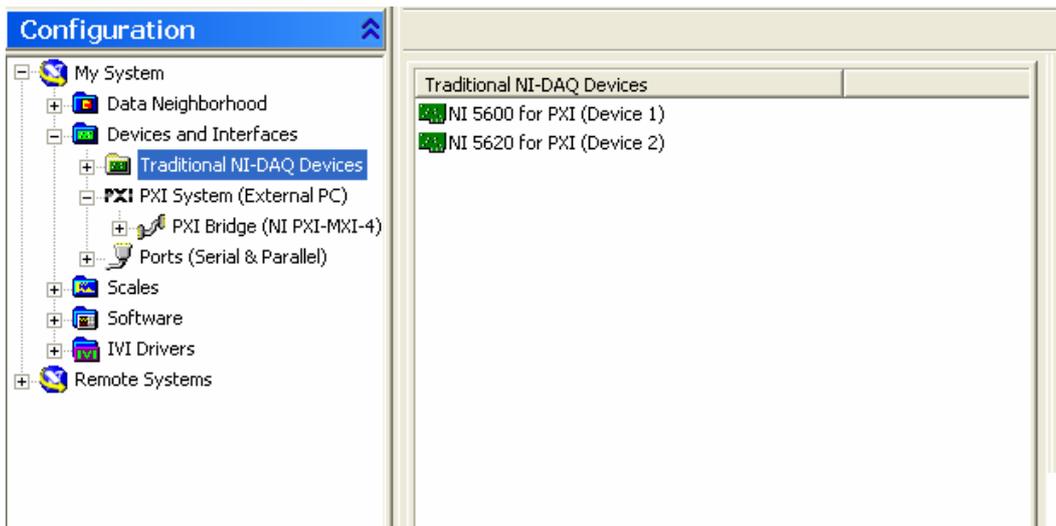
8. *Click* com o botão direito do *mouse* em **PXI System (Unidentified)** e selciona **Identify As -> External PC**.



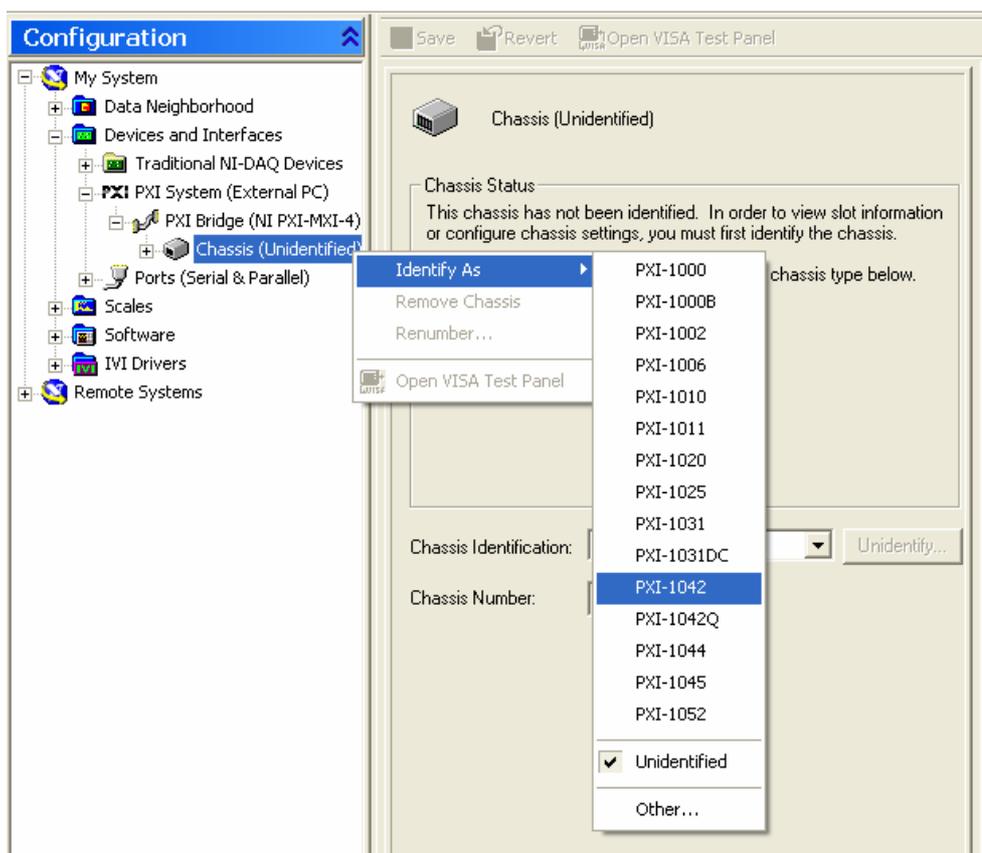
9. Então deve aparecer uma pasta **Tradicional NI-DAQ Devices**, caso isto não ocorra aperte o botão F5.



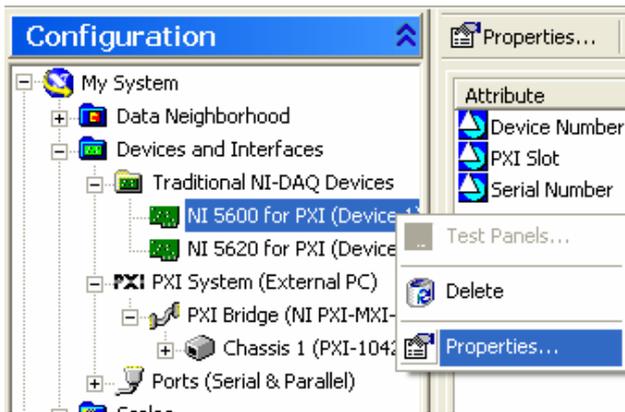
10. Selecione essa nova pasta e na janela ao lado deve aparecer os dispositivos instalados, anote o número do mesmo indicado ao lado de **Device**, pois este será usado na programação.



11. Para configurar o chassis, no nível abaixo de **PXI Bridge (NI PXI-MXI-4)** na estrutura de pastas, aperte como botão direito do *mouse* em **Chassis (Unidentified)** e selecione **Identify As -> PXI-1042**.



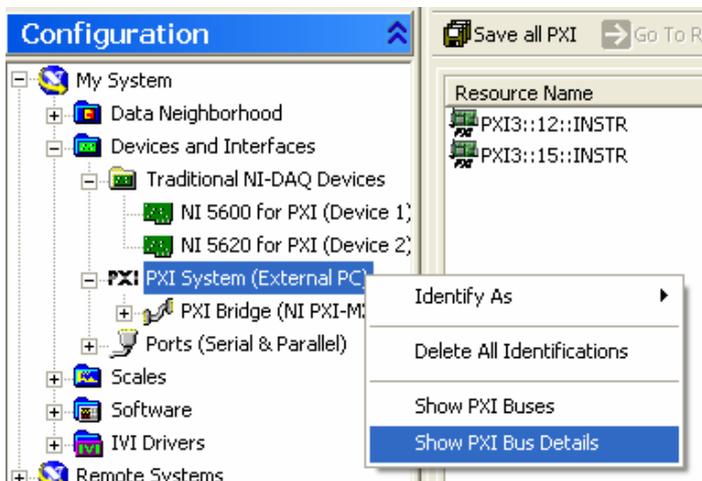
12. Para testar se os módulos estão funcionando corretamente, deve-se selecionar cada um deles no diretório abaixo de **Traditional NI-DAQ Devices**, aperta como botão direito e em seguida seleciona **Properties...**



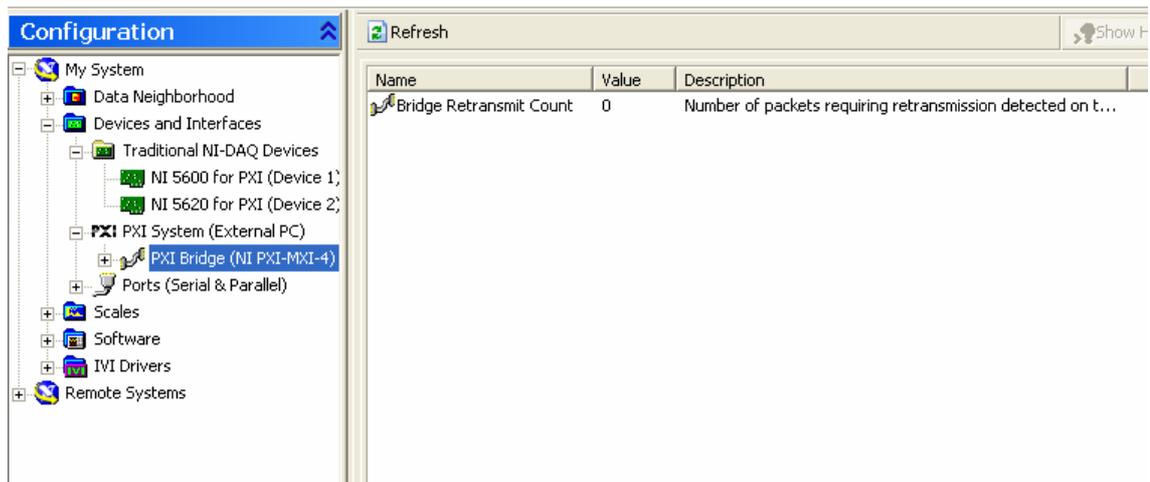
13. Na janela de propriedades aperta-se no botão **Test Resources**, e o teste para o módulo selecionado será feito.



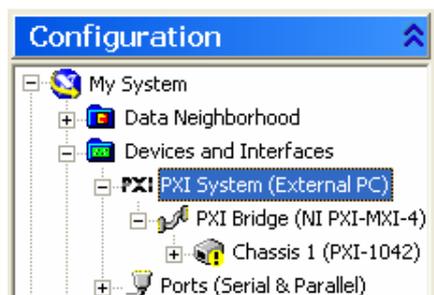
14. Para testar a eficiência do link MXI-4, deve-se apertar como o botão direito em **PXI System (External PC)** e selecionar **Show PXI Bus Details**.



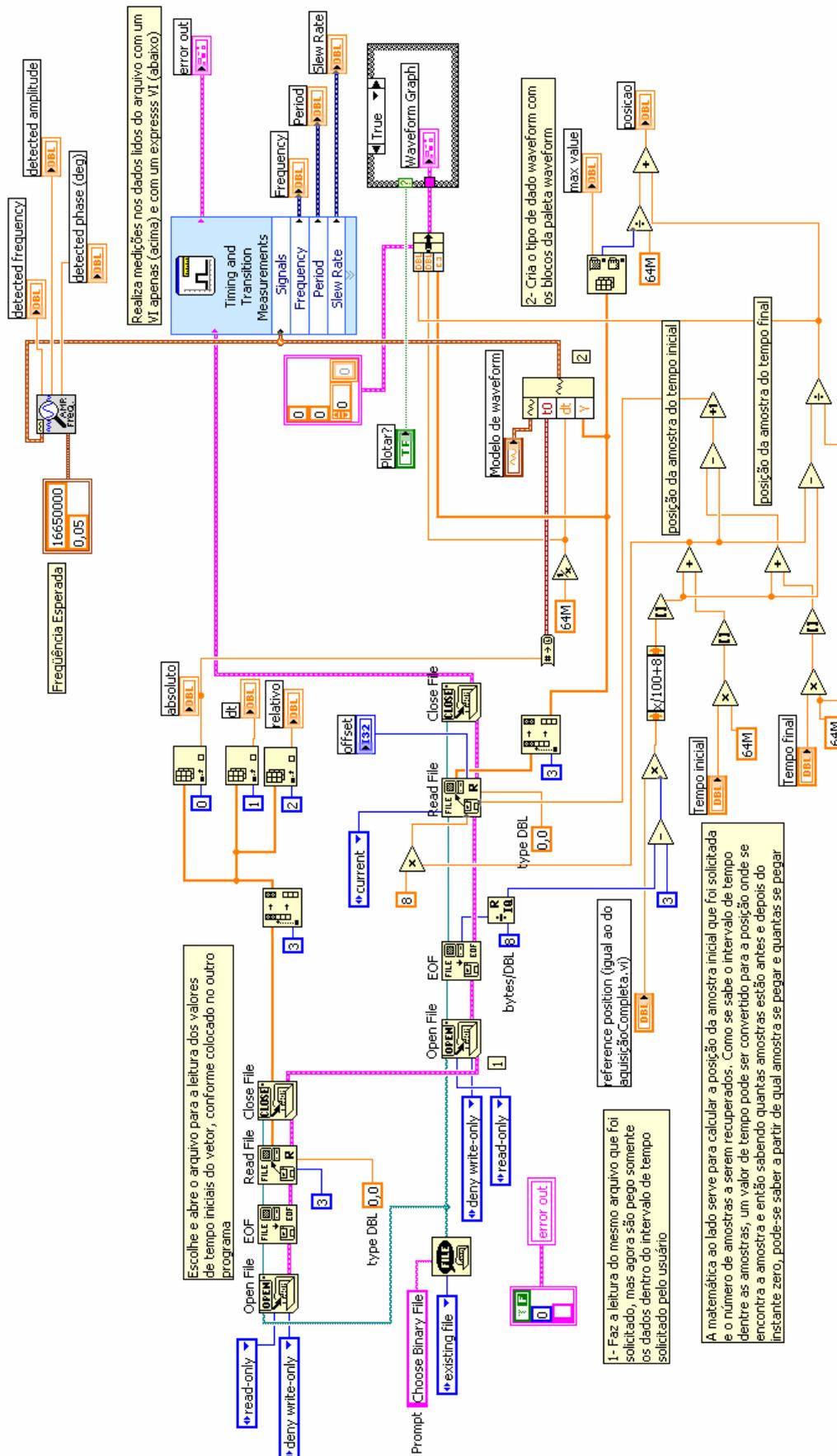
15. E selecionando o mesmo **PXI System (External PC)**, é possível ver o número de retransmissões realizadas pelo MXI-4. O normal é que seja zero.

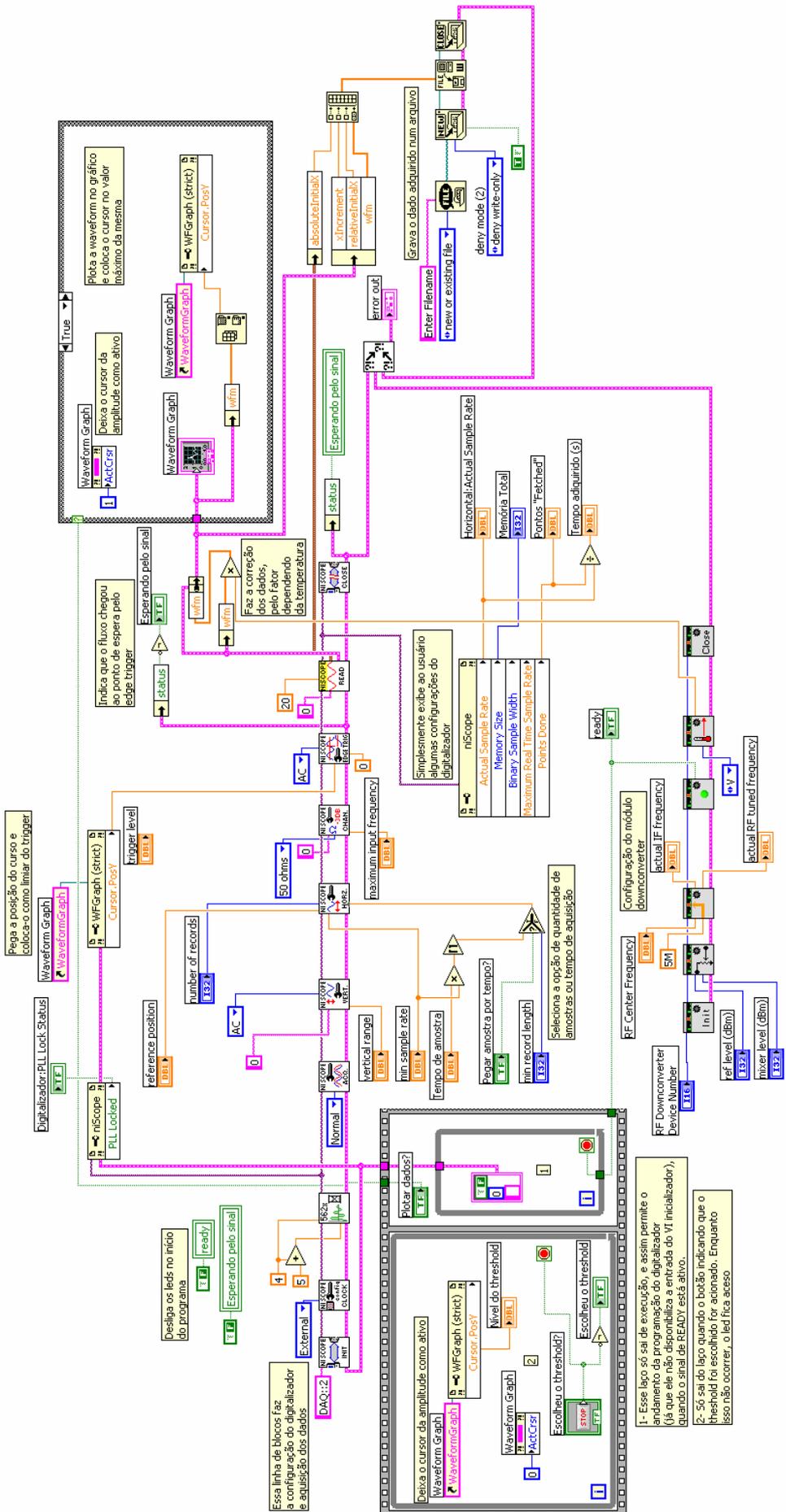


16. Quando há problemas com o *link* de comunicação MXI-4 ao lado da identificação do chassis aparecerá um símbolo de exclamação amarelo e preto, isto pode ocorrer se o chassis não estiver ligado, por exemplo.



## APÊNDICE B – Código dos Principais Programas Desenvolvidos





- 1- Esse laço só sai de execução, e assim permite o refinamento da programação do digitalizador (é que ele não é iniciado do VI inicializador), quando o sinal de READY está ativo.
- 2- 50 sai do laço quando o botão indicando que o threshold foi escolhido for acionado. Enquanto isso não ocorrer, o led fica aceso.

## REFERÊNCIAS

- [1] CNES, ARGOS Plataform Transmitter Terminals – General Specifications and Certification, julho 1998
- [2] Dubut, J. e Aquino, C. D., Plataforma de Coleta de Dados para o Sistema ARGOS – Manual Técnico, setembro 1983
- [3] PXI Systems Alliance, <http://www.pxisa.org>
- [4] National Instruments, <http://www.ni.com>
- [5] VI, Function & How-To help, LabView 7.1
- [6] LabView UserManual 7.pdf
- [7] NI 5660 RF Vector Signal Analyzer – Getting Started Guide
- [8] PXI – NI PXI-1042 Series User Manual
- [9] MXI-4 Series User Manual
- [10] Carlson , A . Bruce, Communications Systems, McGraw-Hill
- [11] <http://digital.ni.com/softlib.nsf/websearch/2BB24B1FE891AE8F86256EF50054700E>