

---

INTRODUÇÃO À INSTRUMENTAÇÃO BIOMÉDICA

INICIANDO COM O LABVIEW

---

OBS: Neste manual de Introdução ao LabView, os textos entre colchetes e em negritos são os comandos de programação.

**Passo 1:** Execute o LabView [**Iniciar >> Programs >> National Instrument >> LabView** ]

1. Coloque a janela “Painel frontal” (janela cinza) ao lado da janela “Diagrama de Blocos” (janela branca): [**windows>>tile left and right**]; Observe que as duas janelas têm os mesmos cabeçalhos.
2. Conhecendo as “Paletas” de trabalho:
  - Tools Pallets [**View>>Tools Pallet**], presente em ambas as janelas;
  - Controls Pallets [**View>> Controls Pallet**] na janela “Painel de Controle”;
  - Functions Pallets [**View>>Functions Pallet**] na janela “Diagrama de Bloco”;

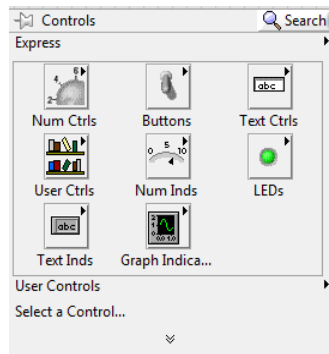
DICA: Uma forma prática de acessar “Controls Pallets” and “Functions Pallets” é clicando nas janelas com botão da direita do mouse. Verifique!

Salvando o projeto: [**arquivo>>save>>IIB>>lab01-a.vi**]. Se o arquivo IIB ainda não existe, crie-o no diretório C:\.

- **Conhecendo o LabView:** Para facilitar e acelerar o seu aprendizado com o LabView, deixe a janela “Context Help” [**Help >> Show context Help**] sempre aberta. Ao passar o mouse pelas VIs do LabView, automaticamente as informações da respectiva VI é mostrada.

- **Faça um tour por várias funções do LabView.**

Dica: Uma maneira fácil de encontrar uma VI ou função do LabView é usando a opção Search no canto direito superior da janela de funções (ver figura abaixo).



- **Pensando como LabView:** Você já deve ter notificado que as funções do LabView estão ordenadas por classes:

- **Structures:** todas as funções necessárias para estruturar uma programação. Ex: sequência, for, while, if, etc.
- **Numeric:** Todas as funções necessárias para operacionalizar os dados. Ex: soma, subtração, multiplicação; divisão, geração de números aleatórios, etc.
- **Boolean:** Funções para operação lógicas. Ex: and, or, not, etc. **String:** Funções para manipular textos. Ex: caixa de texto, tamanho do arquivo, etc.
- **Array:** Funções para manipulação matricial dos dados: Ex: criação de vetores, matrizes, estruturação da matriz, etc.

**Atenção:** Ao manipular com o LabView, você deve observar que as ligações entre as funções dessas classes estão diferenciadas por cores. **Números:** amarelo; **string:** rosa; **booleano:** verde, etc. No caso das classes numéricas, o tipo de dados é diferenciado pelo tipo da ligação: **valor unitário:** linha fina; **vetor:** linha intermediária; **matriz:** linha dupla.

## Passo 2: Programando em LabView

- **Gerando e plotando números aleatórios:**

- Insira a função “**Random Number**” na janela “**Diagrama de Bloco**”, [Functions » Numeric » Random Number (0–1)]. Conecte a saída do gerador ao gráfico. Para isto use a ferramenta “conect wire” da **Tools Palette**;
- Crie um indicador usando o botão direito do mouse e clicando sobre a saída da função geradora. Verifique que um indicador foi automaticamente adicionado na janela de “**Painel Frontal**”.
- Execute o programa, clicando na seta branca no canto superior esquerdo da janela. Coloque um “**While Loop**” nesta operação geradora de números aleatórios. A função geradora deverá ficar dentro do loop. [Functions»Structures » **While Loop**]. O loop gerará números aleatórios até que um comando FALSO ou VERDADEIRO seja instruído.
- Conecte o botão “**Stop**” no terminal do “**While Loop**”.
- Execute o programa.

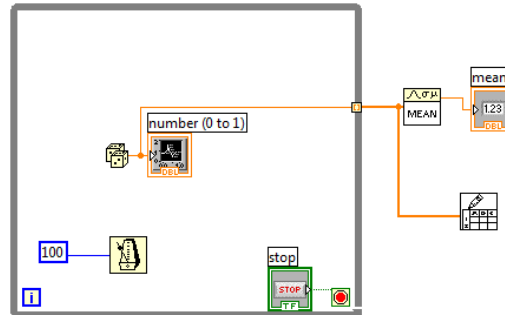
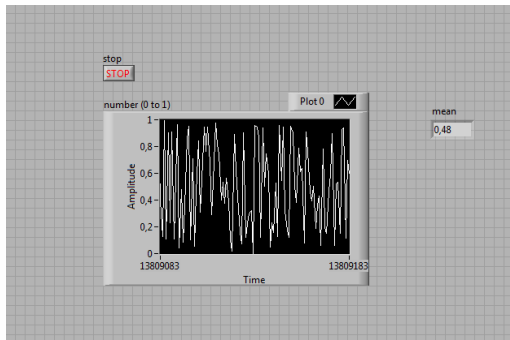
- Verifique que a ação mecânica do botão **“Stop”** poderá ser modificada usando a opção **[Mechanical Action]** clicando com o botão direito do mouse sobre o botão **“Stop”**. Teste as opções e descubra o que acontece.
- Mude o indicador de número aleatórios para **“Waveform Chart”** selecionando **[Controls » Graph » Waveform Chart]** no **Painel de Controle**. Não se esqueça de usar o botão direito do mouse para executar estas funções. Este ambiente gráfico mostra um ponto por vez. Coloque o título **“Random Plot”** no gráfico usando o comando **“Text Edit”** da **“Tools palette”**.
- Mude a amplitude da escala vertical do gráfico para [Min= -1 e Max= +1] clicando nos valores extremos (após colocar o número pressione **“enter”**).
- Veja esta montagem do **“Diagrama de Bloco”** e **“Painel Frontal”** na figura abaixo.

#### - Adicionando um Temporizador à VI

- Na janela **“Diagrama de Bloco”** insira a função **Wait Until Next msMultiple** dentro do **“While Loop”** **[Functions»Time & Dialog»Wait Until Next msMultiple]**.
- Configure o temporizador: com a ferramenta **“Wire”**, clique com o mouse direito sobre o terminal esquerdo do temporizador e selecione **“Create Control”** e coloque um botão de controle variando entre 1 e 500 ms. Escolha o modelo do botão usando a opção clicando sobre a caixa controle de tempo (mouse direito) **[Replace>>Numeric]**. Configure usando a opção **propriedades** do botão.
- Execute a função e observe que a frequência de amostragem dos dados mudou. Portanto você poderá usar este comando temporizador para controlar frequência de aquisição de dados e/ou controle de acesso a equipamentos.

#### - Adicionando análises e arquivo I/O à VI.

- Faça a média dos valores randômicos: adicione a VI **“mean.vi”** no **“Diagrama de Bloco”**, fora do **“While Loop”** **[Functions » Mathematics » ProbabilityandStatistics » Mean.vi]**;
  - Se a conexão entre o loop e a VI **mean.vi** falhar, mude a condição do quadrado amarelo sobre o loop para **“Enable Indexing”**. Use o mouse direito para realizar esta tarefa.
  - Adicione um indicador na saída da VI **mean.vi**. Faça isto clicando com o mouse direito no lado direito da VI **mean.vi** e selecione **“Create Indicator”** .
  - Salvando os dados em um arquivo: adicione **[Functions » Programming » File I/O» Write To Spreadsheet File.vi]** no diagrama de bloco fora do Loop While. Conecte a saída do loop na entrada **ID** do **“Write To Spreadsheet File”**.
  - Depois que salvar os dados, abra os mesmo usando um programa estatístico (Origin, Excel, etc). Os dados estão distribuídos em linha ou coluna? Se estiverem em linha, mude o status booleano **“Transpose”** na VI salvar para **“T (True)”**. Após salvar os dados, verifique-os novamente através do software estatístico.
- Veja nas imagens abaixo o painel frontal e o diagrama de blocos do exemplo descrito acima.



### Passo 3 - Exercício:

1) Mude o “**While Loop**” por um “**For Loop**”. Assim como você aprendeu no curso de programação, o “**For Loop**” necessita do número de loops a serem executados.

Portanto no número **N**, localizado no canto superior esquerdo do loop, adicione um controle. Faça isto usando o botão direito do mouse. Agora você não irá precisar mais do botão “**Stop**”. Selecione-o com um clique e apague. Pronto! Agora você tem um programa com o número de pontos a ser adquirido já predeterminado.

2) Salve também o vetor tempo. Lembre que numa aquisição de dados, geralmente você tem o sinal em função do tempo, portanto você salva as duas informações. Faça isto concatenando a variação do tempo e a amplitude do sinal aleatório. Use a função [Functions»Array»Build Array.vi]. A variação do tempo você obtém subtraindo a saída de dois temporizadores: Um externo e outro interno ao Loop. Agora você deve ligar este sinal na entrada 2D da **VI** que salva. Mude a opção “**transpose**” na VI salvar para “**false**”, pois agora os dados já foram colocados no formato coluna pela VI “**Build Array**”.

3) Configure o formato dos dados para **flutuante** com apenas duas casas decimais. Use a opção “**format**” na VI salvar. Ex: %.f2

4) Faça uma multiplicação entre dois números usando 4 estruturas de sequência: “**Flat Sequence Structure**”. Insira uma sequência com 4 frame no diagrama de bloco. Insira dois controles e um indicador numérico no painel frontal. No diagrama de bloco, coloque a VI do controle 1 no primeiro frame, a VI do controle 2 no segundo frame; insira o operador de multiplicação no terceiro frame e a VI do indicador numérico no quarto frame da sequência. Como ligar estas três VIs ao operador multiplicação, se elas estão em estruturas diferentes? Para isto, você deve inserir **sequência local** usando o botão direito do mouse sobre a **borda da sequência**. Ligue a saída do comando desejado no conector (quadrado) que aparece na borda interna da sequência. Na sequência seguinte, este conector dará a opção de saída, que deverá ser usada como entrada na sua programação.

5) Transforme sua função numa VI com entrada e saída: Com o botão direito do mouse, no ícone **LabView**, no canto direito da tela **Painel de Controle**, mude para opção **[show conector]**. No mesmo ícone, escolha o padrão na opção **[Patterns]**. A cada quadrinho do ícone você pode atribuir um parâmetro de entrada e/ou de saída. Para isto, clique primeiro em um dos quadrinhos e em seguida em um **controle** ou **indicador** que você deseja associar. O padrão do LabView é: **controle** para os quadrinhos da direita e **indicador** para os quadrinhos da esquerda. Salve o projeto e feixe as janelas.

6) Crie um outro projeto em branco, salve como **lab01-b.vi**: No diagrama de bloco, adicione a **VI** que você criou **[Functions»select a VI»lab01-a.vi]**. Entre com os controles e os indicadores que você atribuiu. Faça isto usando apenas o mouse direito.

Fazer um tour pelos VIs já desenvolvidos pelo Labview também irá acelerar o seu aprendizado. Para isto, use a opção **VI from Template**, ao iniciar o LabView.

Outros exemplos de VIs com nível de programação mais avançada e generalizada poderá ser encontrado usando a opção **[Help » Find examples]**. Existem diversos vídeos, sobre o LabView, no [www.youtube.com](http://www.youtube.com) e na página na National Instruments [www.ni.com](http://www.ni.com).