

# Laboratório de Controle de Sistemas

**Profa. Grace S. Deaecto**

Faculdade de Engenharia Mecânica / UNICAMP  
13083-860, Campinas, SP, Brasil.  
[grace@fem.unicamp.br](mailto:grace@fem.unicamp.br)

Primeiro Semestre de 2023

- 1 Experimento 1
  - Objetivo
  - Conceitos fundamentais
  - Roteiro

## Experimento 1

Introdução ao LabVIEW e à placa de aquisição de dados USB 6001

# Objetivo

O objetivo deste experimento é apresentar o equipamento e o software que serão utilizados ao longo do curso, mais especificamente, a **placa NI-USB 6001** e o **software LabVIEW**, ambos da *National Instruments*. Em linhas gerais são objetivos desta aula :

- Aprender os elementos fundamentais da placa de aquisição de dados :
  - leitura e escrita de sinais analógicos
- Aprender os elementos fundamentais da programação em LabVIEW :
  - utilização do diagrama de blocos, painel frontal, paletas de funções e controle
  - execução de loops
- Utilizar o LabVIEW para geração e aquisição de dados.

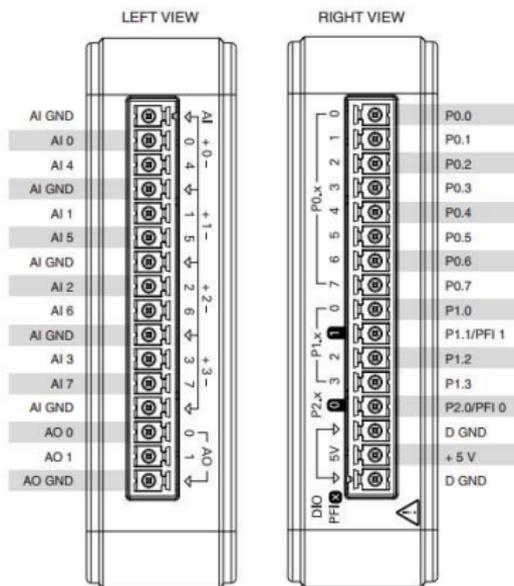
# Placa NI-USB 6001

- Esta é uma placa de aquisição de dados multifuncional de uso geral com :
  - 8 entradas analógicas
  - 2 saídas analógicas
  - 2 portas de entrada e saída digitais e um contador
    - 1 porta de entrada e saída com 8 bits
    - 1 porta de entrada e saída com 4 bits
    - contador de 32 bits



# Placa NI-USB 6001

A figura abaixo apresenta as conexões da placa.



# Software LabVIEW

- O LabVIEW é o nome tanto de uma linguagem de programação como do programa que interpreta esta linguagem.
- A linguagem foi idealizada para **aplicações de controle e interface com equipamentos de medida e processamento de sinais**.
- A programação é feita utilizando elementos gráficos (diagrama de blocos) ao invés de linhas de código.
- A execução de cada bloco é orientada pelo **fluxo de informações**.
- O LabVIEW está associado a um **sistema de aquisição de dados** (conversores AD, DA, contadores) denominado DAQ (Data Acquisition).

# Software LabVIEW

- O ambiente de trabalho do LabVIEW é composto de **duas telas**, **uma de execução**, e outra de interface com o usuário chamada de **painel frontal**.
- Para cada elemento de supervisão ou acionamento o programa cria automaticamente botões, mostradores, traçadores de gráficos e outras formas de instrumentos reais no painel frontal, gerando um **ambiente de instrumentação virtual**.
- Por este motivo se diz que cada programa em LabVIEW é um instrumento virtual e sua extensão é **.vi**.

# Roteiro : Exercício 1 - Introdução ao LabView

## Para realizar no LabVIEW

- Faça um programa que converta uma temperatura de grau Fahrenheit (F) para grau Celsius (C), ou seja

$$C = \frac{F - 32}{1.8}$$

Utilize diferentes tipos de botões de controle e indicadores.

- Faça um loop de forma que a conversão aconteça instantaneamente e visualize o histórico de medidas utilizando o “Waveform Chart” e o “Waveform Graph”.

## Roteiro : Exercício 2 -Aquisição de dados

- Faça um programa em LabView que envie um sinal analógico de onda quadrada para o pino AO.0 e leia este mesmo sinal no pino AI.0 da placa NI-USB 6001.

### **Procedimento : Configuração do Canal**

- Utilizando o bloco “Basic Function Generator.vi” , gere um sinal de onda quadrada com frequência de 0.5 Hz, amplitude e offset de 0.5. Em sampling info considere frequência  $F_s = 500$  Hz e número de amostras da forma de onda  $\#s = 1$ .
- Utilizando o bloco “DAQmx Create Virtual Channel.vi” configure o tipo de medida (analógica/digital, Entrada/Saída) e indique o pino desejado para escrita ou leitura.
- Conecte o bloco anterior em “DAQmx Start Task.vi” para a inicialização da geração ou medida do sinal e o bloco “DAQmx Stop Task.vi” para o término da aquisição.

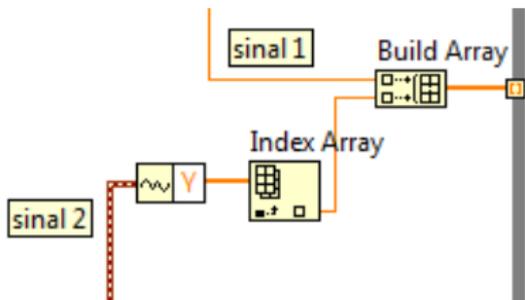
# Roteiro

- Geralmente deseja-se fazer a aquisição e processamento de dados de forma contínua e, portanto, as tarefas relacionadas são colocadas dentro de um loop. Entretanto, como a configuração do canal deve ser realizada somente uma vez, ela fica localizada fora do loop.
- Dentro do loop, faça a conexão correspondente para a escrita ou leitura do sinal utilizando os blocos “DAQmx Write.vi” ou “DAQmx Read.vi”, respectivamente.
- Conecte um visualizador gráfico para o acompanhamento da aquisição.
- Para o armazenamento dos sinais de onda gerada e adquirida utilize o bloco “Write Delimited Spreadsheet.vi”. Indique o endereço correto para o armazenamento do arquivo e o tipo de dado a ser armazenado (2D matriz ou 1D vetor). Na opção “format” escreva o código “%.;%f”.

# Roteiro

## Para a conversão de dados

- Se o sinal for do tipo “waveform”, ele apresenta várias informações diferentes ( $t_0$ ,  $dt$ ,  $Y$ ). Para obter uma informação específica podemos utilizar o bloco “Get Waveform Components”.
- No caso em questão a informação desejada está no vetor  $Y$ , de forma que, para armazená-lo em arquivo devemos utilizar o bloco “Index Array” como apresentado na figura a seguir.





# Roteiro

## Para realizar no Matlab

- Selecione o diretório onde se encontra o arquivo armazenado.
- Carregue o arquivo utilizando o comando load
- Trace o gráfico dos sinais desejados. Note que a frequência de amostragem escolhida foi de 500 Hz e, portanto, as amostras são obtidas a cada 2 ms, assim

`load nome.txt`

$$t = 0 : 0.002 : (size(nome, 1) - 1) \times 1/500$$

O comando

`plot(t,nome)`

traça o gráfico desejado.