

# Simulação com Qspice

Ricardo Kerschbaumer

[1. Introdução](#)

[2. Introdução ao Qspice](#)

[3. Interface do Usuário](#)

[4. Captura de Esquemáticos](#)

[5. Configuração de Simulações](#)

[6. Inclusão de Sub-Circuitos](#)

[7. Adição de novos componentes ao Qspice](#)

[8. Inclusão de códigos em C++ ou Verilog no Qspice](#)

[9. Análise de Resultados](#)

[10. Recursos Adicionais](#)

## 1. Introdução

### História e Evolução do SPICE

SPICE (Simulation Program with Integrated Circuit Emphasis) é um simulador de circuitos desenvolvido na Universidade da Califórnia, Berkeley, no início dos anos 1970. O projeto original foi iniciado por Laurence Nagel, sob a orientação do professor Donald Pederson. A primeira versão, SPICE1, foi lançada em 1973, com o objetivo de fornecer uma ferramenta eficiente para a análise de circuitos integrados.

O SPICE revolucionou a forma de trabalho de engenheiros e projetistas, possibilitando a simulação precisa de circuitos analógicos e digitais antes da fabricação física. Com o sucesso inicial, o SPICE2 foi lançado em 1975, trazendo melhorias significativas em precisão e funcionalidades. Em 1989, a versão SPICE3, desenvolvida por Thomas Quarles, introduziu uma interface baseada em código-fonte C, facilitando sua portabilidade e expansão.

### Aplicações do SPICE

O SPICE é amplamente utilizado em diversas áreas da engenharia elétrica e eletrônica, incluindo:

- **Projeto de Circuitos Integrados (IC):** Simulação de desempenho e análise de viabilidade de ICs.
- **Sistemas de Comunicação:** Análise de circuitos de rádio-frequência (RF) e micro-ondas.
- **Eletrônica de Potência:** Estudo de conversores e inversores.
- **Automotivo:** Simulação de sistemas eletrônicos embarcados.

- **Indústria de Consumo:** Desenvolvimento de dispositivos eletrônicos, como smartphones e eletrodomésticos.

## Implementações do SPICE

Ao longo dos anos, diversas implementações do SPICE foram desenvolvidas, tanto comerciais quanto gratuitas, cada uma com suas próprias características e vantagens.

Cada implementação possui características próprias, com ênfase em determinados setores ou sistemas operacionais específicos.

### Algumas Implementações Comerciais

1. **HSPICE:** Desenvolvido pela Synopsys, é uma versão altamente precisa e rápida do SPICE, amplamente utilizada na indústria de semicondutores para a simulação de circuitos analógicos e digitais.
2. **PSPICE:** Uma implementação da Cadence Design Systems, PSPICE é conhecida por sua facilidade de uso e integração com ferramentas de projeto de PCB. É amplamente utilizada no ensino e na indústria.

### Algumas Implementações Gratuitas

1. **Ngspice:** Uma das implementações de código aberto mais conhecidas, Ngspice é derivada do SPICE3 e oferece uma vasta gama de funcionalidades para simulação de circuitos analógicos e digitais.
2. **Qspice:** Desenvolvido pela Qorvo, Qspice é uma versão moderna que oferece recursos avançados e uma interface amigável, sendo adequada para aplicações educacionais e de pesquisa.
3. **XSPICE:** Uma extensão do SPICE3, XSPICE adiciona capacidades para simulação de sistemas mistos (analógicos e digitais), sendo útil para simulações mais complexas.
4. **LTspice:** Desenvolvido pela Linear Technology (agora parte da Analog Devices), LTspice é uma das versões mais populares devido à sua eficiência, robustez e disponibilidade gratuita. É muito utilizado para simulações de circuitos de potência e analógicos.

O SPICE e suas variações continuam sendo ferramentas essenciais na simulação de circuitos eletrônicos, com uma rica história de desenvolvimento e uma ampla gama de aplicações.

As diversas implementações, tanto comerciais quanto gratuitas, oferecem opções flexíveis para atender às necessidades de engenheiros, pesquisadores e estudantes.

Este manual focará em como utilizar o Qspice para a simulação de circuitos, fornecendo uma abordagem prática e detalhada para tirar o máximo proveito desta poderosa ferramenta.

## Funcionamento do SPICE

## Aplicação de Linha de Comando e Arquivos de Texto

Originalmente, o SPICE foi projetado para ser utilizado via linha de comando, com circuitos descritos por arquivos de texto. Esses arquivos de texto, conhecidos como netlists, contêm informações detalhadas sobre os componentes do circuito e suas conexões. O processo de simulação em SPICE envolve várias etapas:

### 1. Criação da Netlist:

- A netlist é um arquivo de texto que descreve o circuito a ser simulado. Cada linha da netlist especifica um componente e suas conexões. A netlist também inclui diretivas de controle de simulação.
- **Exemplo de Netlist:**

```
* Exemplo de Divisor de Tensão
V1 N001 0 DC 10V
R1 N001 N002 1k
R2 N002 0 1k
.TRAN 1us 1ms
.END
```

Nesta netlist, `v1` é uma fonte de tensão DC de 10V conectada ao nó N001 e ao terra (0). `R1` e `R2` são resistores conectados em série.

### 1. Execução da Simulação:

- A simulação é iniciada executando o comando SPICE na linha de comando, especificando a netlist como entrada. O SPICE processa a netlist, resolve as equações do circuito e gera os resultados.
- **Comando de Simulação:**

```
spice3 <nome_da_netlist>
```

### 3. Análise dos Resultados:

- Os resultados da simulação são geralmente salvos em arquivos de texto ou binários, que podem ser analisados posteriormente. As saídas típicas incluem tensões nos nós, correntes nos componentes e respostas de frequência.
- Estes resultados podem ser analisados graficamente usando as ferramentas apropriadas.

## Implementações com Interface Gráfica

Com a evolução das tecnologias de software, várias implementações do SPICE incorporaram interfaces gráficas de usuário (GUI) para facilitar a captura de esquemáticos e a visualização dos resultados da simulação. Essas interfaces gráficas tornam o SPICE mais

acessível e intuitivo para engenheiros e estudantes. Vamos explorar algumas dessas funcionalidades:

### 1. Captura de Esquemáticos:

- As GUIs modernas permitem a criação de circuitos utilizando um editor gráfico. Os usuários podem selecionar componentes a partir de bibliotecas, posicioná-los no diagrama e conectá-los utilizando o mouse. Isso elimina a necessidade de escrever netlists manualmente.
- **Exemplos de Ferramentas:**
  - **LTspice:** Possui um editor gráfico intuitivo onde os usuários podem arrastar e soltar componentes no esquemático.
  - **Qspice:** Oferece uma interface amigável para a captura de esquemáticos, com ferramentas de conexão rápidas e opções de configuração de componentes.

### 2. Configuração de Simulações:

- As GUIs facilitam a configuração de simulações. Os usuários podem definir tipos de análises (DC, AC, Transient) e ajustar parâmetros através de menus e caixas de diálogo.
- **Exemplo em LTspice:**
  - Configurar uma simulação transiente através de um painel onde o usuário especifica o tempo de passo e o tempo final da simulação.

### 3. Plotagem dos Sinais de Saída:

- Uma das grandes vantagens das GUIs é a capacidade de visualizar os resultados da simulação de forma gráfica. As ferramentas de plotagem integradas permitem que os usuários analisem facilmente tensões, correntes e outros parâmetros ao longo do tempo ou frequência.
- **Exemplos:**
  - **LTspice:** Permite a plotagem de formas de onda diretamente na interface, com opções de zoom, cursores e medição.
  - **Qspice:** Oferece gráficos interativos para a visualização dos resultados, permitindo a análise detalhada dos sinais de saída.

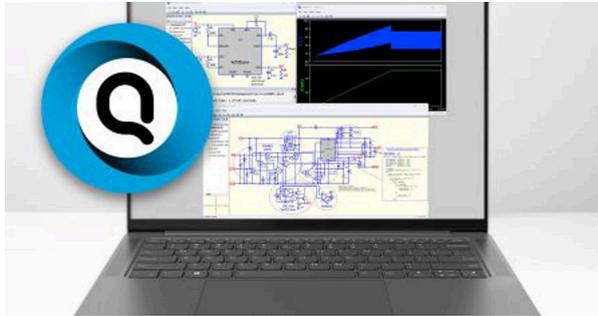
A transição do SPICE baseado em linha de comando para implementações modernas com interfaces gráficas aumentou significativamente a acessibilidade e usabilidade da ferramenta. A captura de esquemáticos através de GUIs e a visualização gráfica dos resultados de simulação permitem uma análise mais intuitiva e eficiente dos circuitos eletrônicos.

No restante deste manual, focaremos no uso do Qspice, demonstrando como aproveitar suas capacidades gráficas para a simulação e análise de circuitos eletrônicos.

OS capítulos serão devidos em introdução ao Qspice, interface de usuário, captura de esquemáticos, configuração da simulação, inclusão de sub-circuitos, adição de novos

componentes, inclusão de códigos em C++ e Verilog, Análise de resultados e recursos adicionais

## 2. Introdução ao Qspice



O Qspice é uma implementação moderna do simulador SPICE, desenvolvida pela Qorvo, uma empresa líder em soluções de RF e semicondutores. Qspice foi projetado para oferecer uma interface amigável e funcionalidades avançadas que facilitam a simulação e análise de circuitos eletrônicos, tanto analógicos quanto digitais.

### Destaques do Qspice

- **Interface Gráfica Intuitiva:** Qspice fornece uma interface gráfica de usuário (GUI) que simplifica a captura de esquemáticos, permitindo que os usuários desenhem e conectem componentes facilmente utilizando um rico conjunto de teclas de atalho.
- **Ferramentas de Simulação Avançadas:** Com suporte para análises DC, AC e Transient, dentre outras a Qspice oferece uma ampla gama de opções para a simulação precisa de circuitos.
- **Visualização de Resultados:** Gráficos interativos que permitem a plotagem de formas de onda e a análise detalhada dos resultados da simulação.
- **Biblioteca de Componentes:** Inclui uma vasta biblioteca de componentes padrão, bem como a capacidade de adicionar modelos personalizados.
- **Integração de Códigos C++ e Verilog:** O Qspice permite integrar na simulação blocos contendo códigos em C++ ou Verilog.
- **Criação de Sub-Circuitos:** No Qspice é possível encapsular parte de um circuito em um bloco hierárquico, permitindo o reaproveitamento deste circuito e uma melhor organização do diagrama.

### Criador do Qspice

O Qspice foi desenvolvido por Mike Engelhardt, um renomado engenheiro conhecido por suas contribuições significativas ao campo da simulação de circuitos eletrônicos. Mike Engelhardt é o criador do LTspice, um dos simuladores SPICE mais populares e amplamente utilizados no mundo. Mike Engelhardt trouxe sua vasta experiência e inovação para o desenvolvimento do Qspice, elevando o padrão das ferramentas de simulação disponíveis no mercado.

## Características da simulação

É importante destacar que diferente de alguns simuladores que permitem a simulação em tempo real, onde é possível interagir com o circuito durante a execução, manipulando chaves e botões, o Qspice, assim como as outras ferramentas baseadas em spice, realiza a simulação de maneira estática. Isso significa que todas as condições iniciais e configurações do circuito devem ser definidas antes de iniciar a simulação. Durante a execução, o Qspice processa o circuito conforme as especificações fornecidas, gerando os resultados ao final da simulação sem permitir intervenções dinâmicas. Essa abordagem, embora limita a interatividade durante a simulação, garante maior precisão e controle sobre os parâmetros analisados, facilitando a obtenção de resultados consistentes e repetíveis.

## Instalando o Qspice

### Requisitos do Sistema

O Qspice não é um software pesado e assim roda na maioria dos computadores, porém, para a simulação de circuitos grandes e complexos é necessário uma grande capacidade de processamento, ou então a simulação pode ficar demorada.

Estes são os requisitos de sistemas:

- Windows 11 ou uma edição de 64-bit do Windows 10
- 4 GB RAM (mínimo, 16 GB recomendado)
- 100 MB de espaço em disco para a instalação
- Pelo menos 16 GB de espaço em disco para os dados de simulação

### Download e Instalação

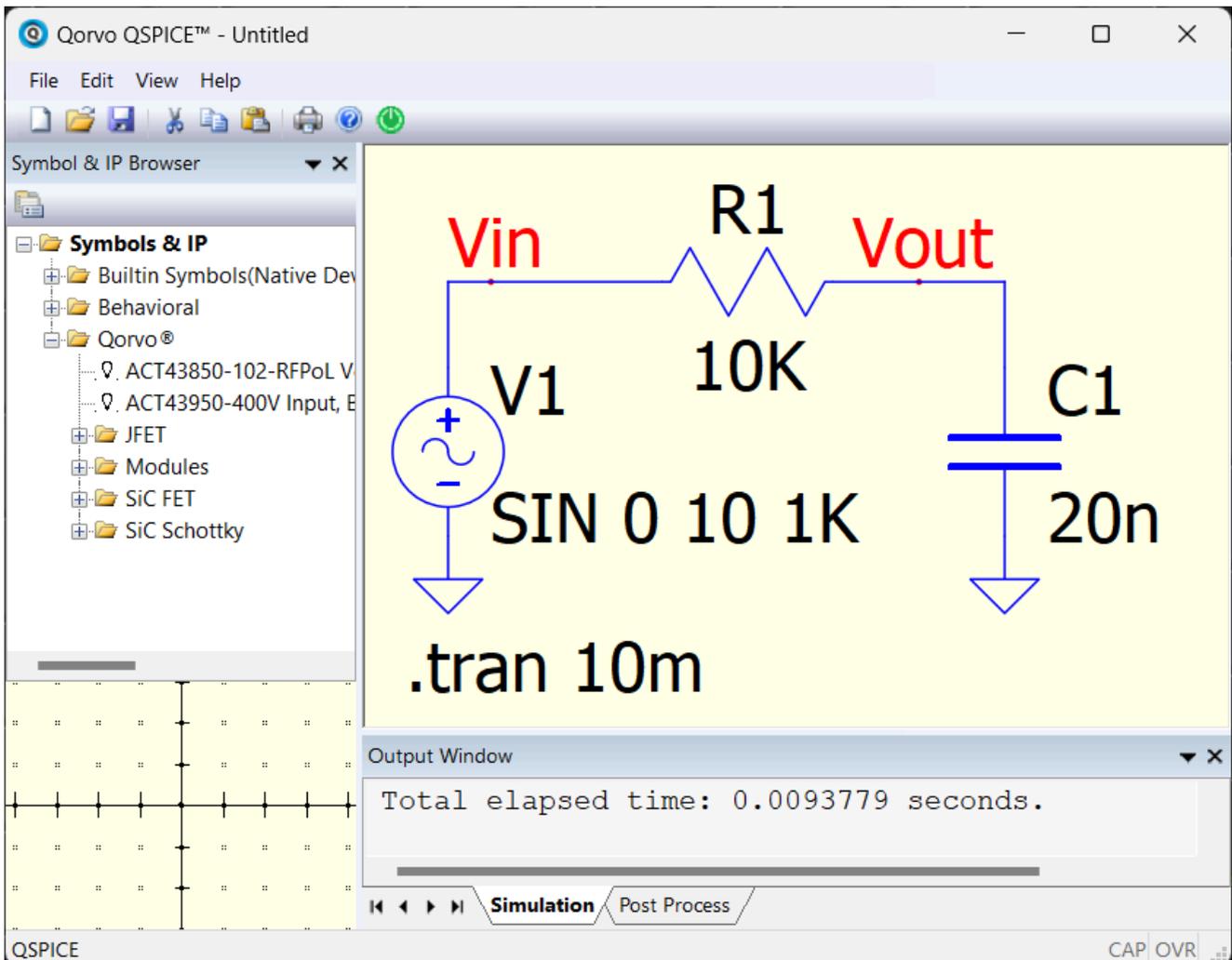
Como se trata de uma ferramenta gratuita, ela pode ser baixada diretamente da internet no link: <https://www.qspice.com/> ou na página da Qorvo no link: [QSPICE™ Simulator - Qorvo](#)

O processo de instalação segue o padrão do Windows, sem complicações.

## 3. Interface do Usuário

O Qspice possui uma interface enxuta, parte de sua filosofia de trabalho que é utilizar o mínimo de janelas e botões, focando em teclas de atalho. Desta forma a captura dos esquemáticos e a configuração da simulação se tornam mais ágeis. Porém, esta abordagem não é fácil para quem está iniciando, pois leva um tempo para se acostumar com as teclas de atalho e a sintaxe das diretivas de simulação.

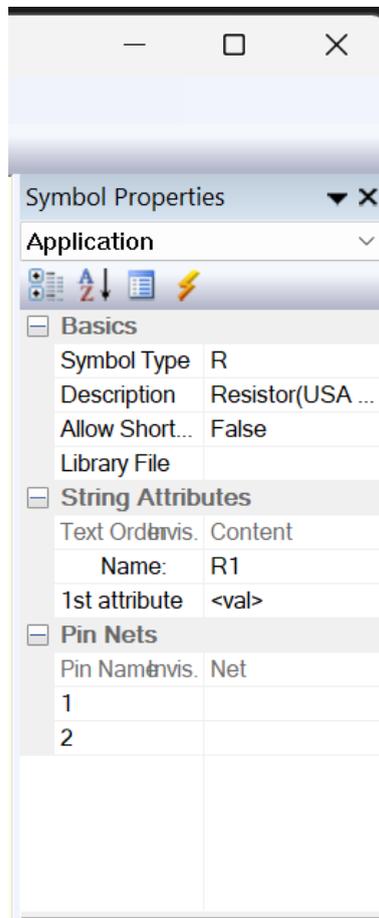
A imagem a seguir mostra a interface principal do Qspice.



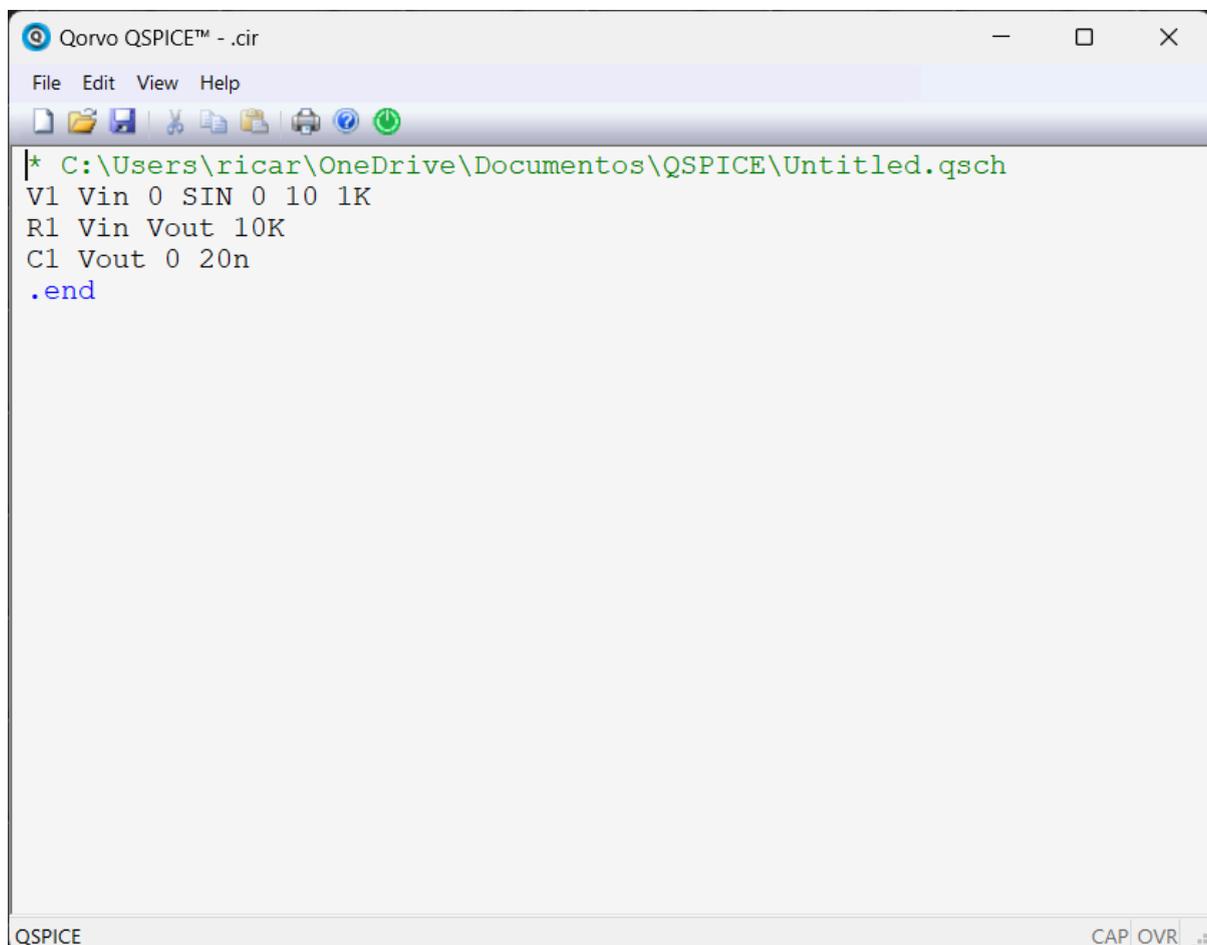
Esta janela possui as seguintes áreas:

- Na parte superior temos a barra de menus e ferramentas, com as funções básicas de manipulação de arquivos e controle da simulação.
- Na área central da janela temos o espaço onde o diagrama é desenhado.
- Na lateral esquerda temos uma árvore de componentes onde estão localizados os símbolos e modelos dos componentes a serem utilizados na construção do circuito a ser simulado.
- Na parte inferior da janela, temos a área de saída, onde são apresentadas as informações da simulação, inclusive as informações sobre os erros encontrados no circuito.

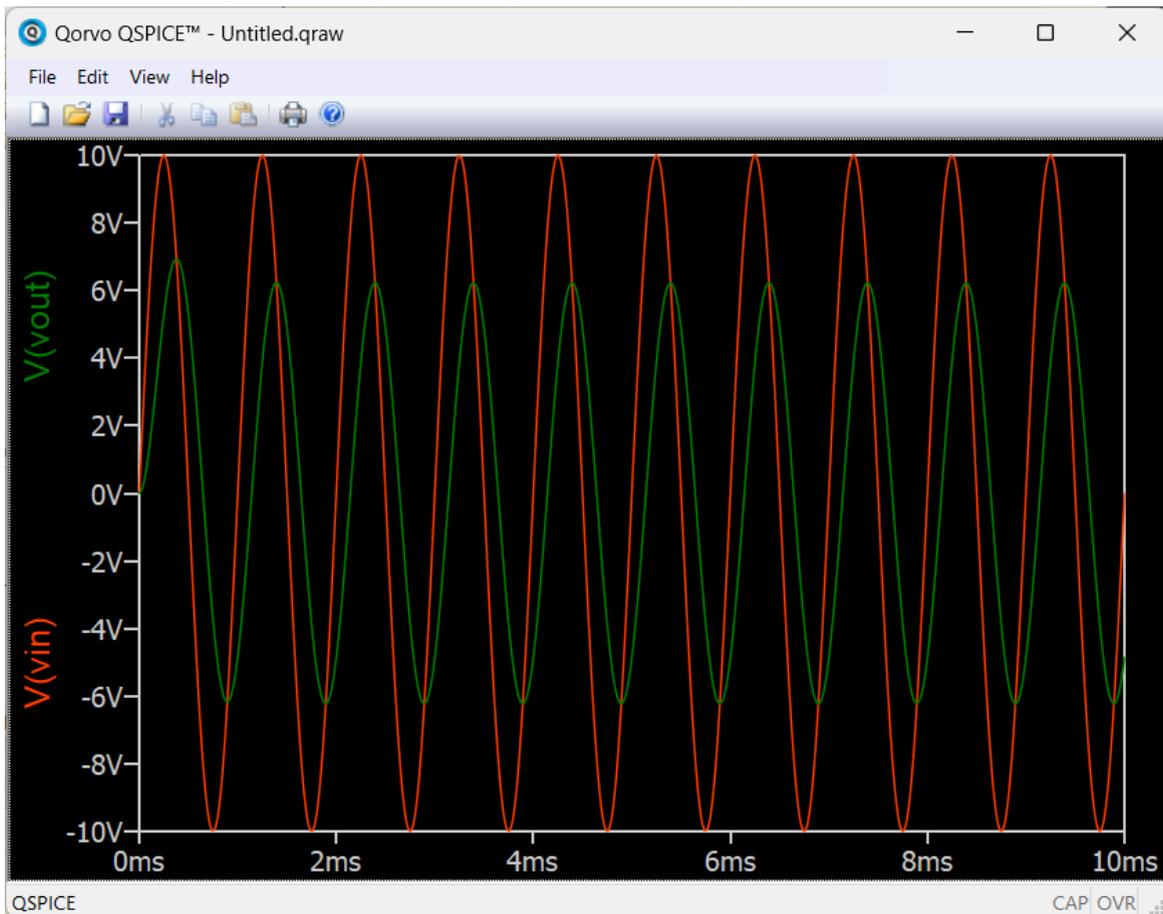
Quando selecionada a opção de visualização das propriedades de um determinado componente, uma nova área é aberta no lado direito da janela principal, como mostrado a seguir.



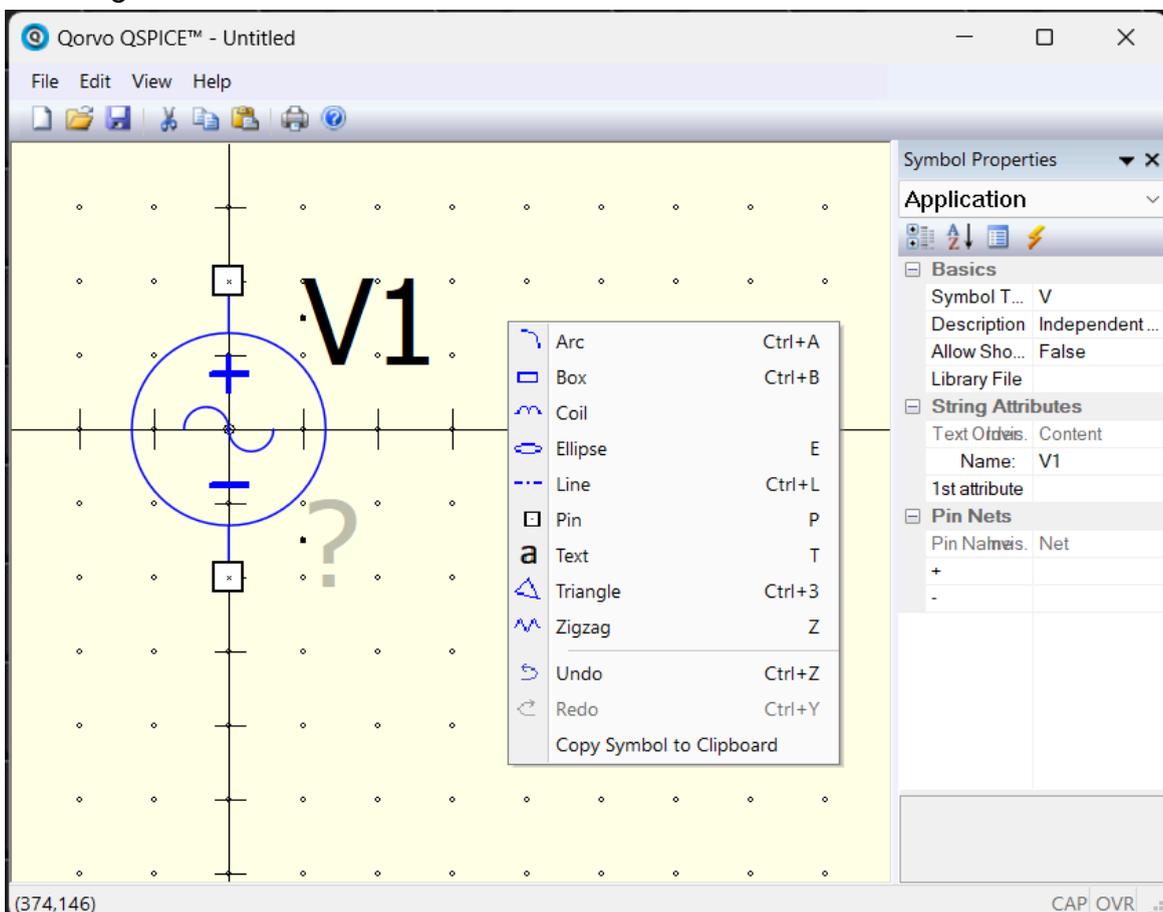
Existem ainda janelas auxiliares que são abertas para apresentar informações específicas. A imagem a seguir apresenta a imagem da janela que apresenta a Netlist.



Após a execução da simulação é apresentada uma janela onde são mostrados os resultados da simulação. A imagem a seguir mostra esta janela.



Existe também no Qspice um editor de símbolos, que permite criar novos componentes. A imagem a seguir mostra sua interface.



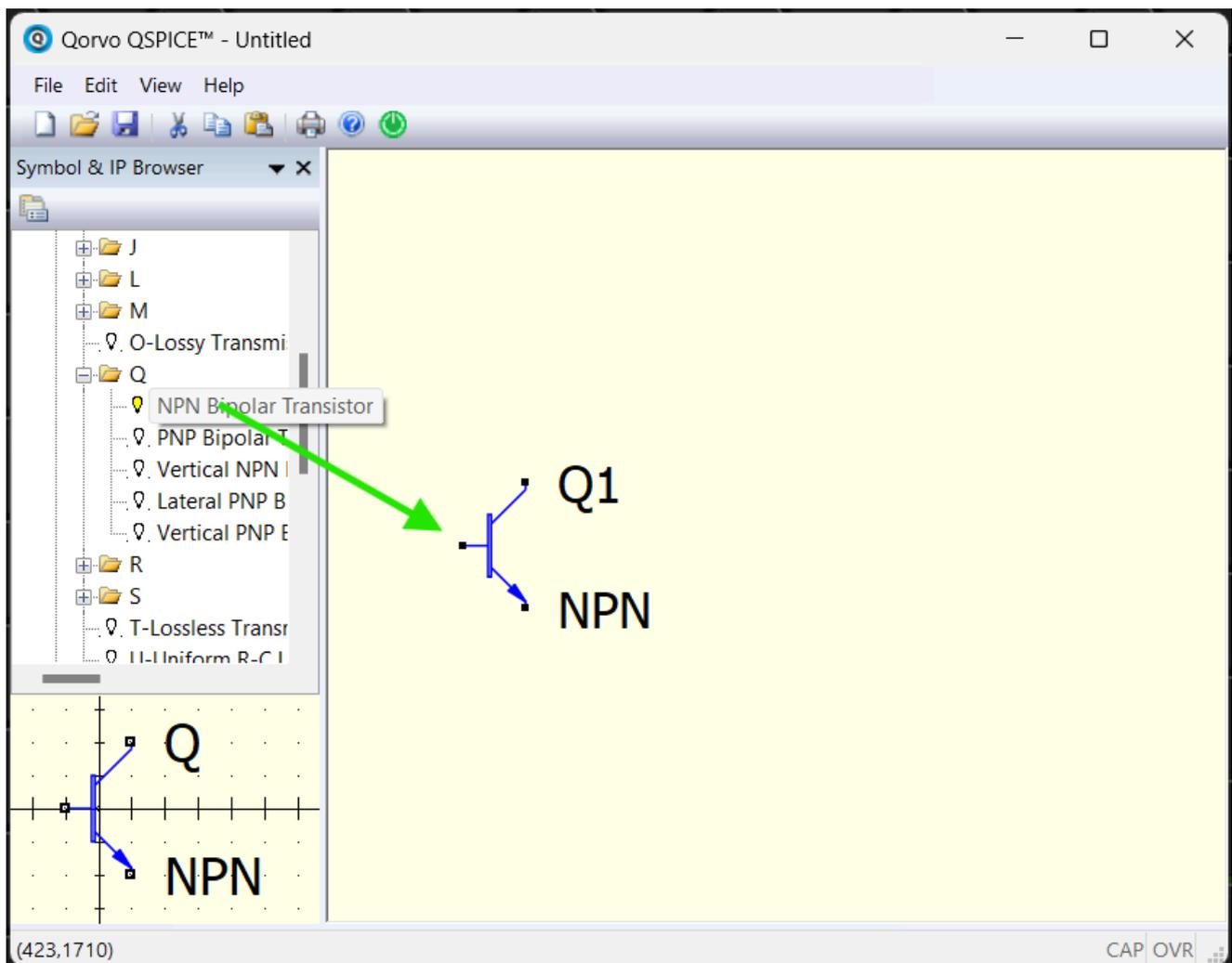
Existem ainda outras janelas com informações específicas, como valores RMS, gráficos de FFT, etc. que não serão apresentadas neste momento.

## 4. Captura de Esquemáticos

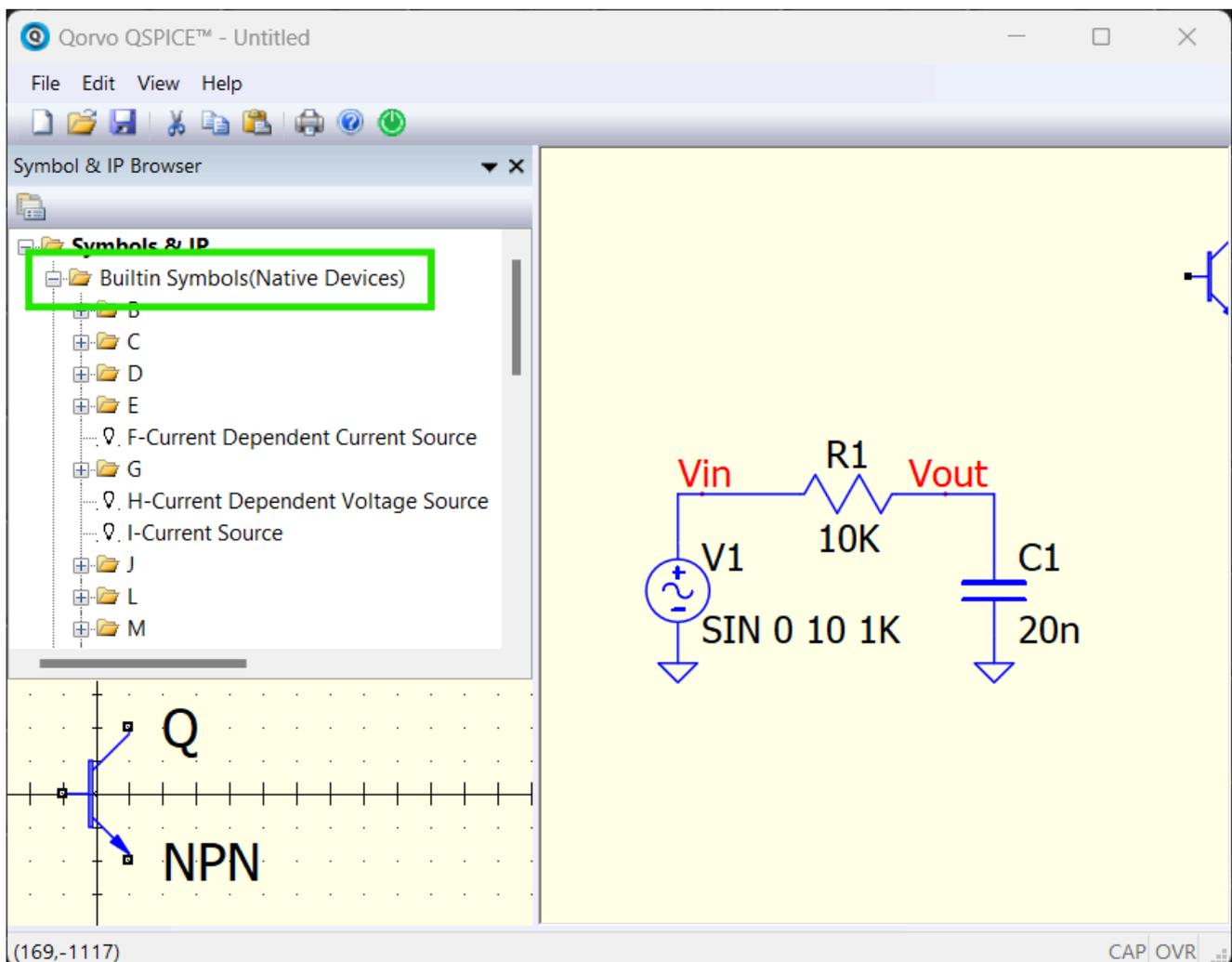
O primeiro passo para realizar uma simulação é a captura do esquemático, que se dá através do desenho do circuito. O Qspice fornece um conjunto de símbolos que podem ser utilizados neste processo. Assim, a captura do esquemático inicia pela adição dos componentes.

### Adicionando Componentes

A adição de componentes pode ser realizada principalmente de duas maneiras, clicando e arrastando o símbolo da árvore de componentes localizada na esquerda da janela principal, ou clicando na tecla de atalho correspondente. Na imagem a seguir é inserido um transistor.



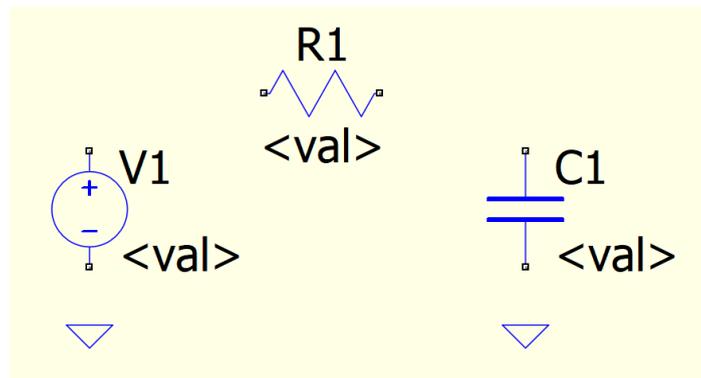
Para inserir um transistor utilizando teclas de atalho basta clicar a tecla "q" no teclado. O uso de teclas de atalho é muito recomendado, pois agiliza muito a montagem do circuito. É importante destacar que os símbolos dos componentes básicos estão localizados na pasta "Builtin Symbols(Native Devices)", como destacado na imagem a seguir.



Algumas informações importantes:

- A tecla de atalho para um determinado componente é a letra que dá nome à pasta onde o componente se encontra.
- Para componentes com mais de um modelo (por exemplo transistor PNP, NPN etc) basta clicar mais de uma vez a tecla de atalho.
- Para rotacionar um componente basta clicar "Ctrl + r".
- Para apagar um componente basta apontá-lo com o mouse (sem clicar) e apertar a tecla "Delete".
- Para mover um componente basta segurá-lo com o mouse e arrastá-lo.
- Para selecionar mais de um componente é necessário segurar a tecla "Shift" enquanto faz a seleção com o mouse.
- Para que a simulação funcione é obrigatório a inserção do terra (gnd), para isso use a tecla de atalho "g".

Como exercício vamos inserir os componentes mostrados na figura a seguir:

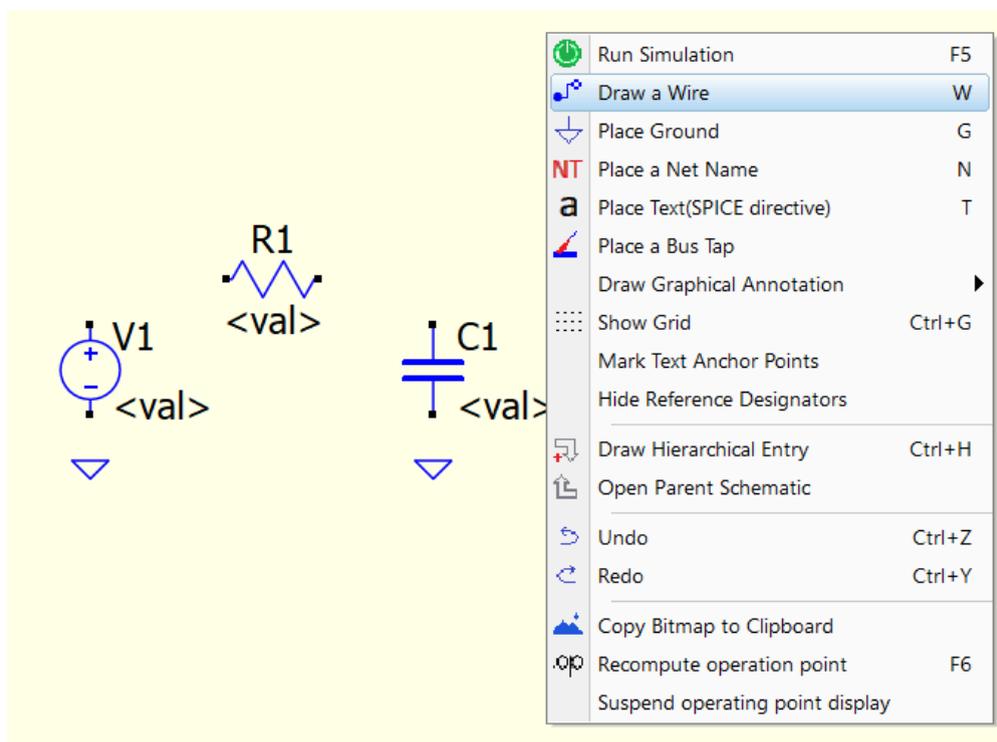


Para isso siga os seguintes passos:

1. Clique na tecla "v" e posicione a fonte V1.
2. Clique na tecla "r" para selecionar um resistor.
3. Clique "Contro+r" para rotacionar o componente e posicione o resistor R1.
4. Clique na tecla "c" e posicione o capacitor C1.
5. E finalmente clique na tecla "g" e posicione os dois símbolos de terra.

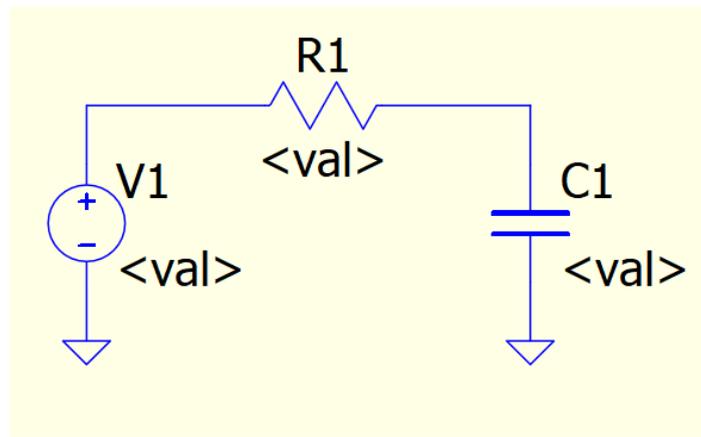
## Conectando Componentes

Para realizar a conexão dos componentes é utilizado um fio (wire). A ferramenta de conexão Wire pode ser selecionada clicando com o botão direito do mouse em uma parte vazia da janela e selecionando "Draw a Wire", como na imagem a seguir.



Ou simplesmente clicando na tecla de atalho "w".

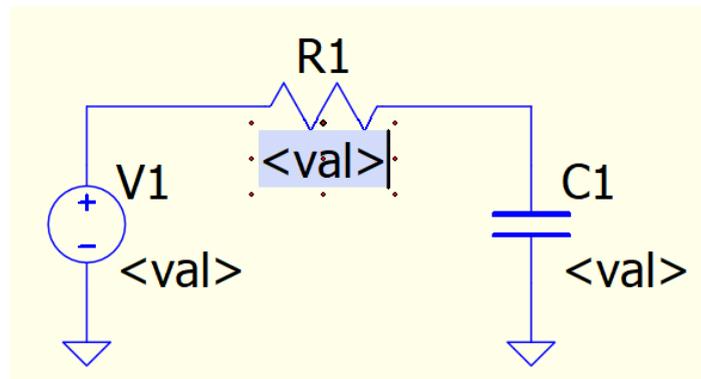
Para exercitar faça a conexão dos componentes como indicado na imagem a seguir.



## Parametrizando os Componentes

Para que a simulação funcione corretamente e reflita o comportamento do circuito real é necessário que cada componente seja parametrizado corretamente. Componentes simples como resistores e capacitores podem ser parametrizados simplesmente ajustando seus valores. Mas para componentes mais complexos, como transistores ou circuitos integrados é necessário um modelo.

Para ajustar os valores dos componentes basta dar um clique duplo sobre a palavra <val> e digitar o valor, como na imagem a seguir.

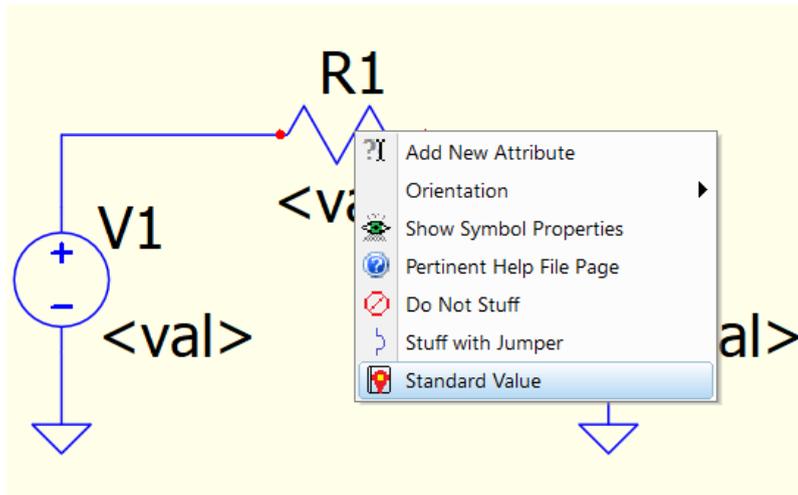


Para facilitar a manipulação de valores muito grandes ou muito pequenos o Qspice permite o uso dos sufixos apresentados na tabela a seguir.

Sufixo	Multiplicador
f	1e-15
p	1e-12
n	1e-9
u(or μ)	1e-6
mil	25.4e-6
m	1e-3
K	1e+3
Meg	1e+6
G	1e+9

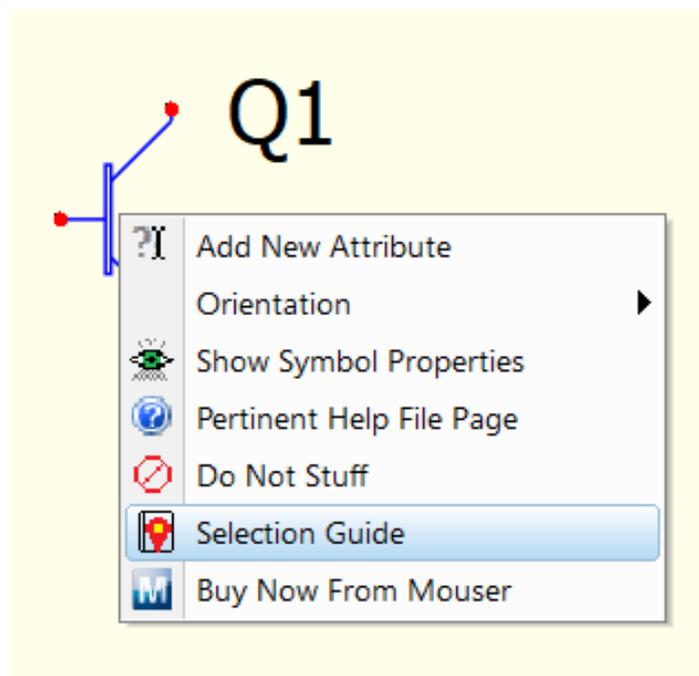
Sufixo	Multiplicador
T	1e+12

Para seleccionar valores comerciais para os componentes é necessário clicar com o botão direito sobre o componente e seleccionar a opção "Standard Value", como na imagem a seguir.

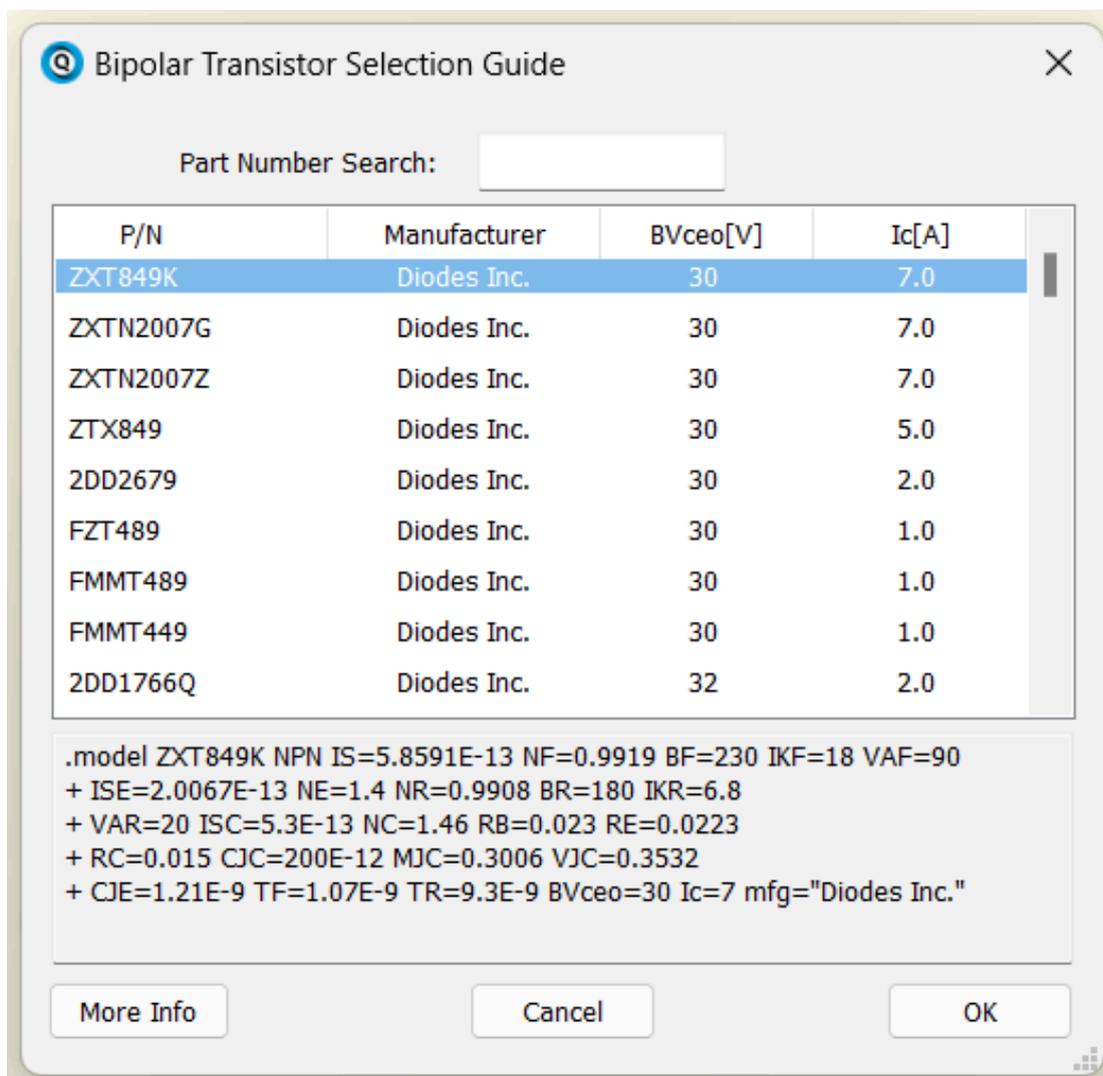


Como mencionado, para componentes mais complexos é necessário um modelo. O Qspice possui uma biblioteca com diversos modelos de componentes comerciais. Além disso é possível adicionar modelos de componentes personalizados de forma bastante fácil. Isso será abordado mais a frente neste material.

Para seleccionar um modelo de simulação para um componente é necessário clicar sobre o componente e seleccionar a opção "Selection Guide". A figura a seguir mostra como abrir o guia de seleção para escolher o modelo de simulação de um transistor.

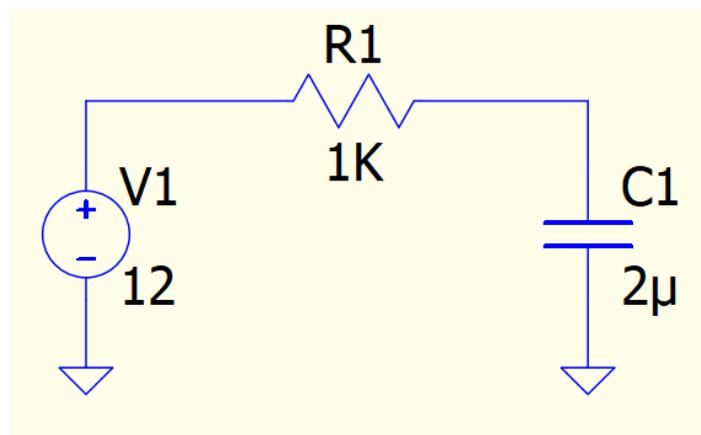


A imagem a seguir mostra o guia de seleção de modelo de simulação.



Observe na parte inferior da janela quantos parâmetros de simulação estão associados ao modelo de transistor selecionado. Esta modelagem permite que o Qspice simule todas as características do componente.

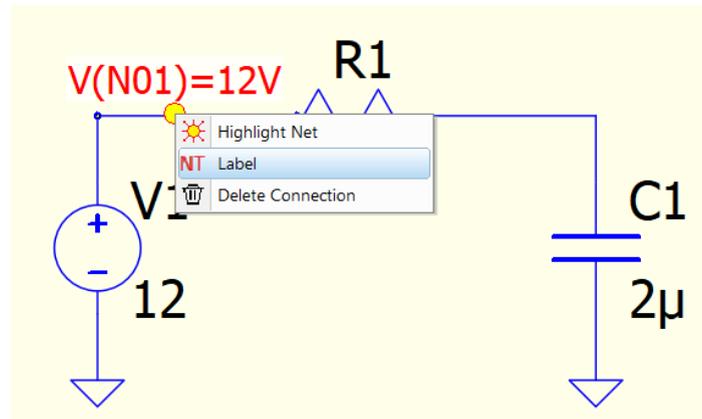
Para exercitar ajuste os valores dos componentes como mostrado na imagem a seguir.



O Qspice atribui automaticamente os nomes dos componentes, que podem ser editados se necessário.

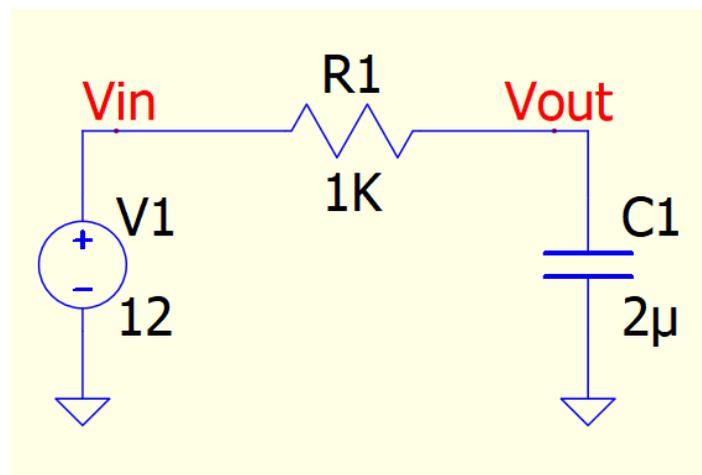
## Ajustes no Circuito

Também é possível visualizar e alterar os nomes dos nós do circuito. Para isso basta clicar com o botão direito do mouse sobre o fio e selecionar a opção "Label", como mostrado na imagem a seguir. Com um clique duplo sobre o nome é possível editá-lo.



A tecla de atalho "n" também pode ser utilizada para dar nomes aos nós do circuito.

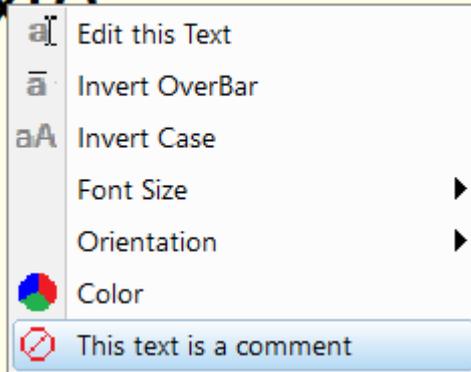
Como exercício ajuste os nomes dos nós do circuito como na imagem a seguir.



Para facilitar a compreensão do circuito é interessante adicionar textos, comentários etc. Para isso pode-se utilizar a tecla de atalho "T". Porém para adicionar texto é necessário tomar algumas providências. Todos os textos adicionados ao circuito são por padrão interpretados pelo simulador como comandos de simulação, assim, os textos informativos devem ser atribuídos como comentário. Para isso siga os seguintes passos.

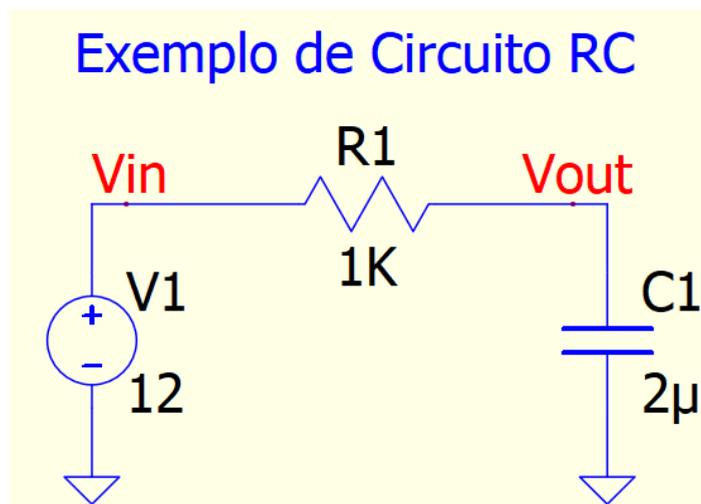
1. Clique na tecla de atalho "t".
2. Digite o texto desejado.
3. Posicione o texto no circuito clicando com o mouse.
4. Clique com o botão direito do mouse sobre o texto.
5. Selecione a opção "This text is a comment". Como mostrado na imagem a seguir.

texto

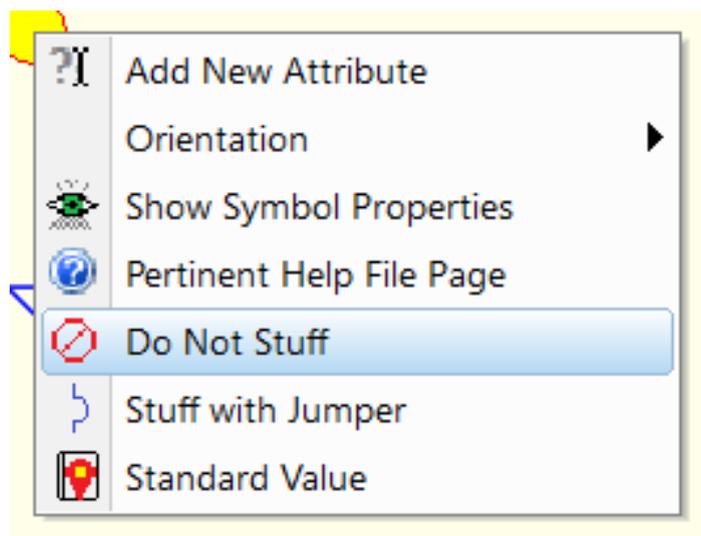


O texto ficará na cor azul, indicando que é um comentário.

Como exercício insira um texto no circuito, como na imagem a seguir.



Também é possível desativar um componente ou substituí-lo por um fio temporariamente para realizar testes durante a simulação. Para isso basta clicar com o botão direito do mouse sobre o componente e selecionar uma das opções, "Do Not Stuff" para desativar o componente ou "Stuff with Jumper" para substituí-lo temporariamente por um fio. Veja a figura a seguir.



## Salvando e Carregando Simulações

Agora que finalizamos nosso circuito é possível salvá-lo. Para isso basta acessar a opção "Save" no menu "File", ou clicar no ícone correspondente.

Para abrir simulações salvas, basta acessar a opção "Open" no menu "File", ou clicar no ícone correspondente.

## Simulando o Circuito

O Qspice fornece diversos modos de simulação, que serão abordado na próxima seção. Mas para ilustrar o processo de simulação do circuito faremos aqui uma simulação simples de transiente.

Antes de realizar uma simulação é necessário configurar o simulador, isso é feito adicionando diretivas de texto ao circuito. Como exemplo vamos adicionar as seguintes linhas

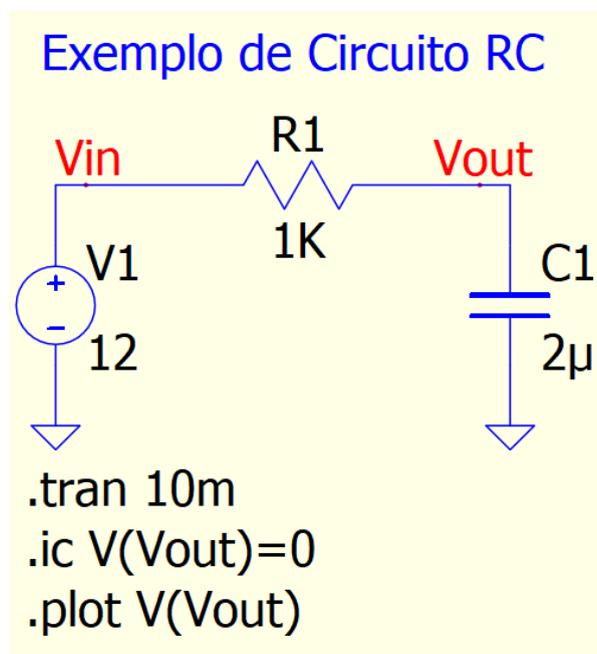
### Diretivas

```
.tran 10m  
.ic V(out)=0  
.plot V(out)
```

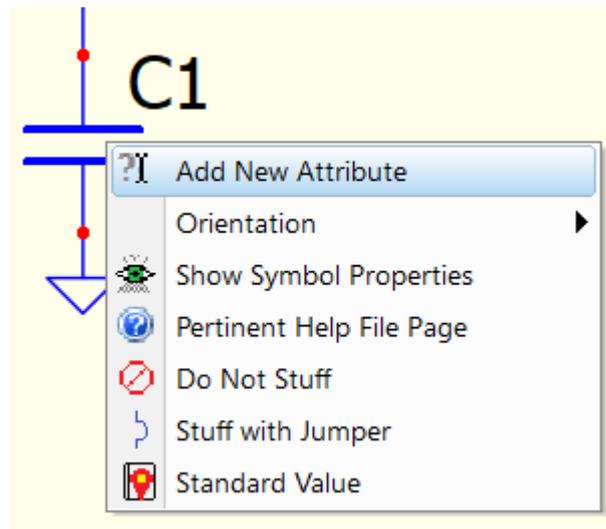
Essas diretivas instruem o simulador a realizar uma simulação transiente de 10 ms, com uma tensão inicial de 0 V no nó Vout, e a plotar o gráfico da tensão neste nó.

Para inserir estas diretivas basta clicar a tecla de atalho "t" digitar o texto e posicionar no circuito.

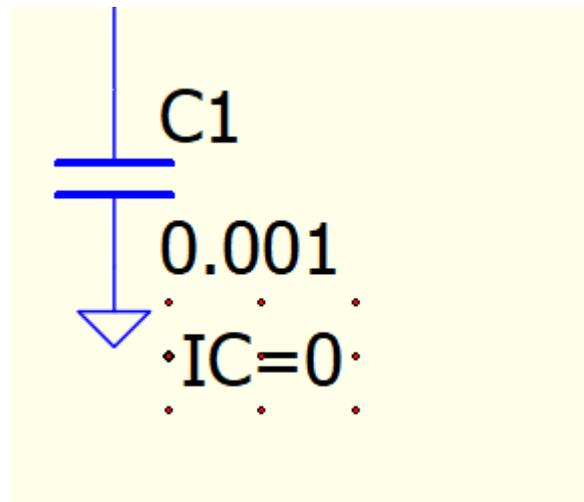
Como exercício insira as diretivas no circuito, como mostrado na imagem a seguir.



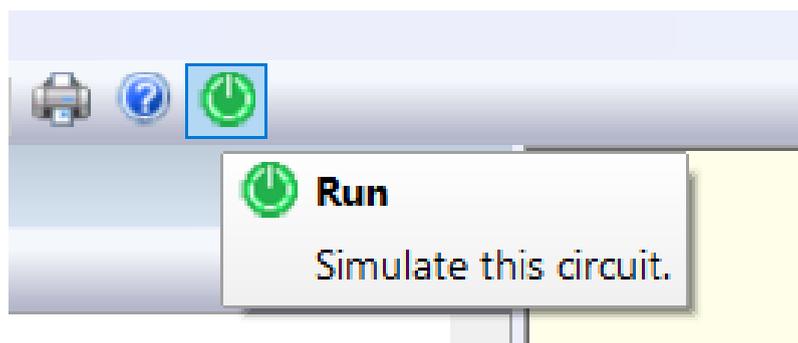
Também é possível definir a condução inicial de um componente clicando com o botão direito sobre ele e selecionado "Add New Attribute", como na figura.



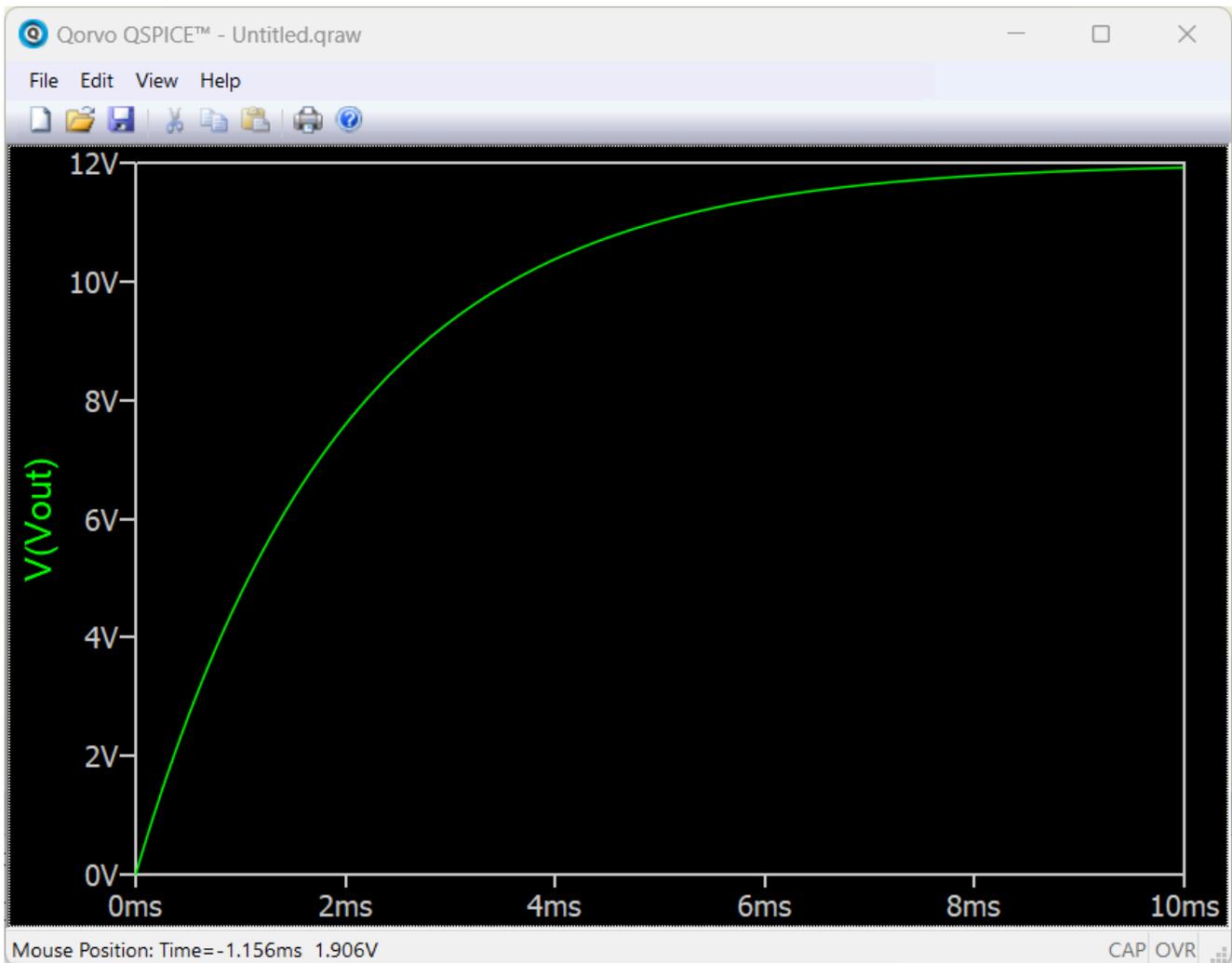
Será adicionado um campo de texto. Neste campo deve-se digitar "IC=" e o valor desejado, como na figura a seguir.



Agora com a simulação configurada, para executar a simulação basta clicar no botão "Run" como mostrado na imagem a seguir, ou usar a tecla de atalho "F5"



O resultado da simulação é apresentado em uma nova janela, como na imagem a seguir.



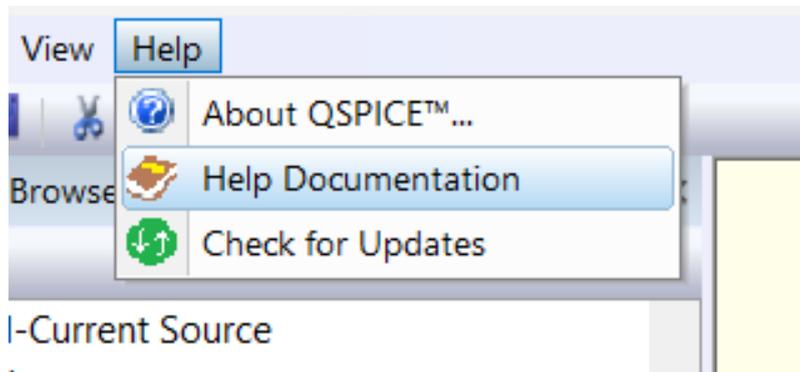
Estas janelas podem ser ajustadas lado a lado para facilitar a operação e visualização.

A título de informação, a imagem a seguir apresenta a netlist gerada para o circuito anterior.

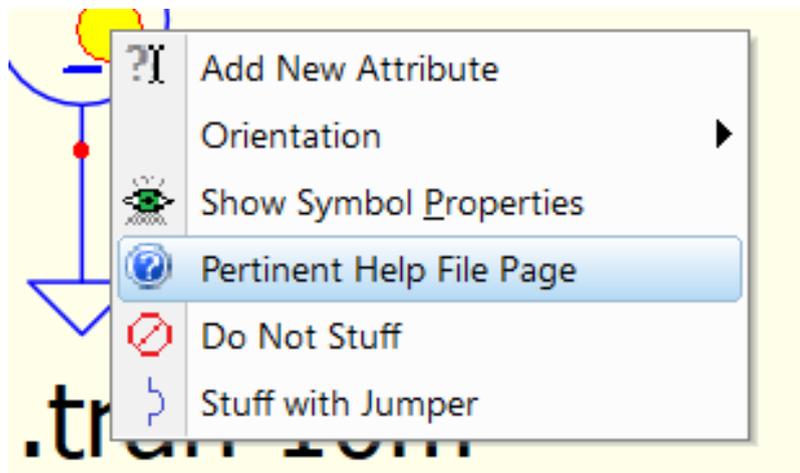
```
Qorvo QSPICE™ - Tutorial.cir
File Edit View Help
* C:\Users\ricar\OneDrive\Área de Trabalho\Tutorial.qsch
V1 Vin 0 12
R1 Vin Vout 1K
C1 Vout 0 2µ
.tran 10m
.ic V(Vout)=0
.plot V(Vout)
.end
```

## Obtendo Ajuda

O Qspice possui uma extensa documentação que ajuda bastante a resolver problemas e tirar dúvidas. Para acessar esta documentação basta clicar em "Help Documentation", no menu "Help", como na imagem a seguir.



Também é possível obter a ajuda sobre um componente específico clicando sobre ele com o botão direito e selecionando "Pertinent Help File Page", como mostrado na imagem a seguir.



## Teclas de Atalho

Como pode ser observado nos exemplos anteriores a interface do Qspice é desenhada para facilitar a operação utilizando teclas de atalho. A tabela a seguir mostra todos os atalhos da interface de captura de diagramas esquemáticos.

Comando	Descrição em Inglês	Descrição em Português
Espaço	Zoom to fit	Ajustar zoom para caber na tela
B <sup>1</sup>	Behavioral source	Fonte comportamental
C <sup>1</sup>	Capacitor	Capacitor
D <sup>1</sup>	Diode	Diodo
E <sup>1</sup>	E-source	Fonte controlada por tensão
F	F-source	Fonte controlada por corrente
G <sup>2</sup>	Ground, G-source	Terra, fonte controlada por condutância

<b>Comando</b>	<b>Descrição em Inglês</b>	<b>Descrição em Português</b>
H	H-source	Fonte controlada por corrente de transcondutância
I	Current Source	Fonte de corrente
J <sup>1</sup>	JFET	Transistor de Efeito de Campo (JFET)
L <sup>1</sup>	Indutor	Indutor
M <sup>1</sup>	MOSFET	Transistor de Efeito de Campo de Metal-Óxido-Semicondutor (MOSFET)
N	Place a net label	Colocar uma etiqueta de rede
Q <sup>1</sup>	Bipolar Transistor	Transistor Bipolar
R <sup>1</sup>	Resistor	Resistor
S <sup>1</sup>	Voltage Controlled Switch	Chave controlada por tensão
T <sup>3</sup>	Place Text	Inserir texto
V <sup>1</sup>	Voltage Source	Fonte de tensão
W	Start a wire	Iniciar uma conexão
Y	Piezoelectric Crystal	Cristal piezoelétrico
Z <sup>1</sup>	MESFET	Transistor MESFET
Ctrl+A	Draw an arc (graphical annotation)	Desenhar um arco (anotação gráfica)
Ctrl+B	Draw a box (graphical annotation)	Desenhar uma caixa (anotação gráfica)
Ctrl+C	Copy selected object(s) to clipboard	Copiar objeto(s) selecionado(s) para a área de transferência
Ctrl+F	Find	Localizar
Ctrl+G	Toggle display of grid dots	Alternar exibição dos pontos da grade
Ctrl+L	Draw a line (graphical annotation)	Desenhar uma linha (anotação gráfica)
Ctrl+M	Mirror selected object(s)	Espelhar objeto(s) selecionado(s)
Ctrl+R	Rotate selected object(s)	Rotacionar objeto(s) selecionado(s)
Alt+Ctrl+R	Rotate in 45° increments	Rotacionar em incrementos de 45°
Ctrl+V	Paste; whether CAD objects, bitmap, or text	Colar; seja objetos CAD, bitmap, ou texto
Ctrl+X	Cut	Recortar
Ctrl+Y	Redo	Refazer
Ctrl+Z	Undo	Desfazer
Ctrl+3	Draw a triangle (graphical annotation)	Desenhar um triângulo (anotação gráfica)

Comando	Descrição em Inglês	Descrição em Português
;	Toggle a text graphic's comment (or component's stuff) status	Alternar o status de comentário de um gráfico de texto (ou informações de um componente)
F2	Toggle visibility of the Symbol and IP Browser pane	Alternar visibilidade do painel de Símbolos e Navegador de IP
F3	Toggle visibility of the Symbol Properties pane	Alternar visibilidade do painel de Propriedades do Símbolo
F4	Toggle visibility of the output console	Alternar visibilidade do console de saída
F5	Run the simulation	Executar a simulação

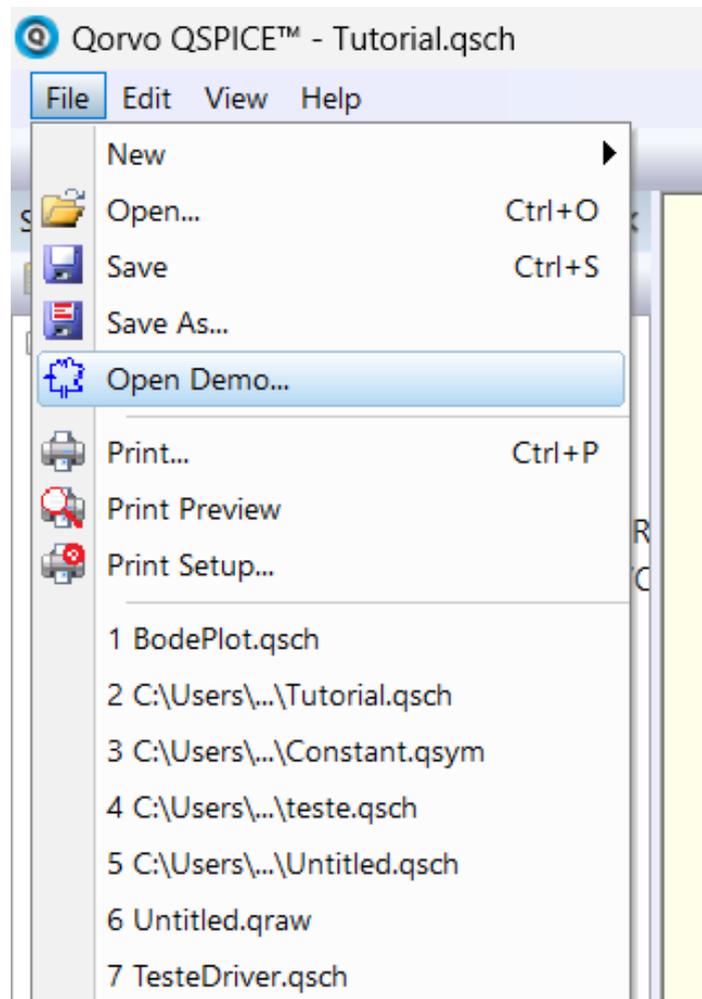
<sup>1</sup> Pressionar repetidamente a tecla alterna entre diferentes versões do símbolo.

<sup>2</sup> Pressionar repetidamente a tecla 'G' alterna entre diferentes versões do símbolo de terra e, em seguida, dos símbolos de fonte de condutância.

<sup>3</sup> A tecla ponto, '.', é aceita como sinônimo de 'T'.

## Exemplos

Para conhecer um pouco mais sobre o Qspice é possível abrir vários exemplos. Para acessar os exemplos basta clicar em "Open Demo" no menu "File", como apresentado na imagem.



## 5. Configuração de Simulações

O Qspice oferece uma variedade de tipos de simulação, adequadas tanto para projetos analógicos quanto para projetos de sinal misto (mixed-signal). Para que se possa realizar uma simulação é necessário primeiro configurar os parâmetros desta simulação. A simulação da seção anterior é um exemplo de simulação de transiente.

### Tipos de Simulação

Os principais tipos de simulação que o Qspice suporta incluem:

1. **Ponto de operação (Operation Point Analysis: .op):** Usada para encontrar o ponto de operação estático do circuito, analisando como as correntes e tensões DC se distribuem no circuito.
2. **Simulação de varredura DC (DC Sweep: .dc):** Similar a simulação DC, porém permite variar um ou mais parâmetros do circuito (como valores de fontes) e observar como essas variações afetam o comportamento geral.
3. **Simulação AC (AC Analysis: .ac):** Focada na análise de resposta em frequência dos circuitos. Ideal para verificar o ganho e a fase em diferentes frequências de sinais alternados.
4. **Simulação Transiente (Non-Linear Transient Analysis: .tran):** Permite a análise do comportamento do circuito ao longo do tempo, útil para observar como as variáveis do circuito, como tensões e correntes, mudam em resposta a um estímulo aplicado.

Existem ainda outros tipos de simulação que não serão abordados neste material.

### Ponto de operação (Operation Point Analysis: .op)

Esta análise calcula o ponto de operação DC do circuito, que é o estado estável de tensão e corrente em cada nó e componente do circuito, sem considerar a variação de sinais temporais. É útil para verificar se o circuito está funcionando corretamente antes de iniciar outras simulações, como transiente ou AC.

Para realizar este tipo de análise, a diretiva ".op" deve ser adicionada ao circuito a ser simulado. Esta diretiva não necessita de parâmetros adicionais.

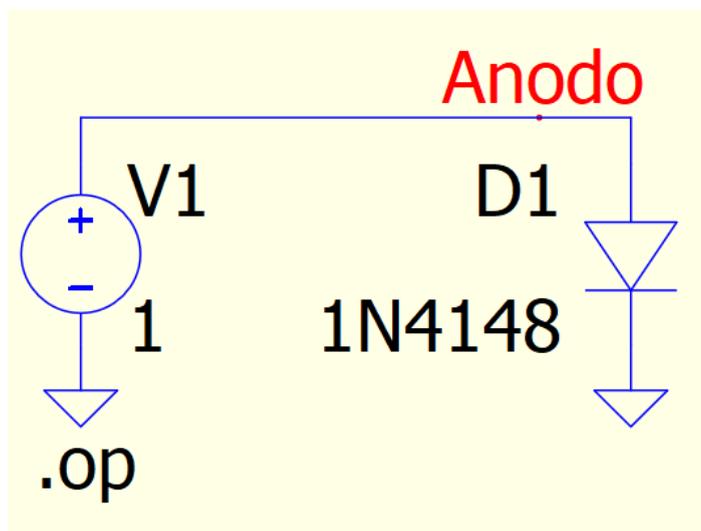
#### Diretiva

```
.op
```

**Importante:** Na simulação DC todos os capacitores são substituídos por circuitos abertos e todos os indutores são substituídos por curto circuitos.

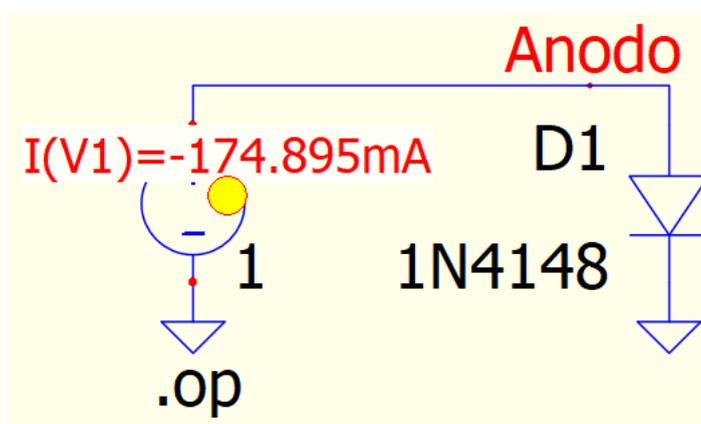
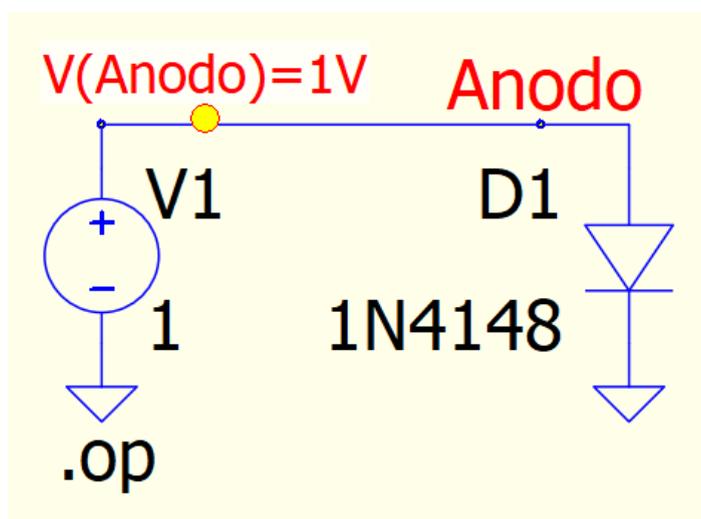
Para exercitar este tipo de simulação construa o circuito da imagem a seguir utilizando as técnicas explicadas na seção anterior. Não esqueça de ajustar a tensão da fonte para 1 V,

escolher o modelo do diodo como 1N4148 e ajustar no nome do nó superior do circuito para Anodo. Adicione também a diretiva ".op", na forma de texto.

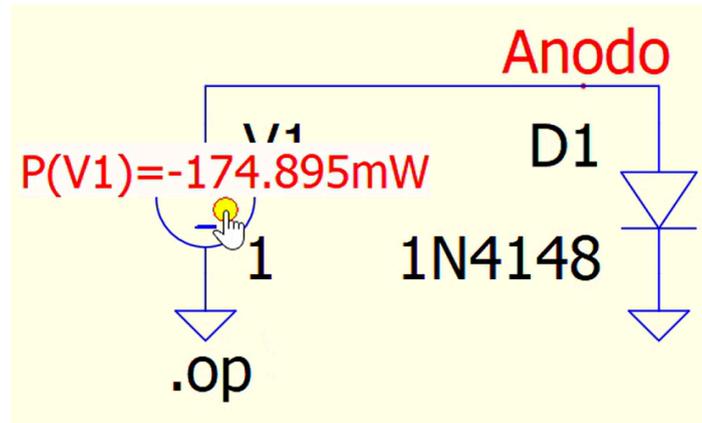


A diretiva ".op" configura o simulador para fazer a simulação DC. Agora, clique no botão "Run" para rodar a simulação, como explicado na seção anterior.

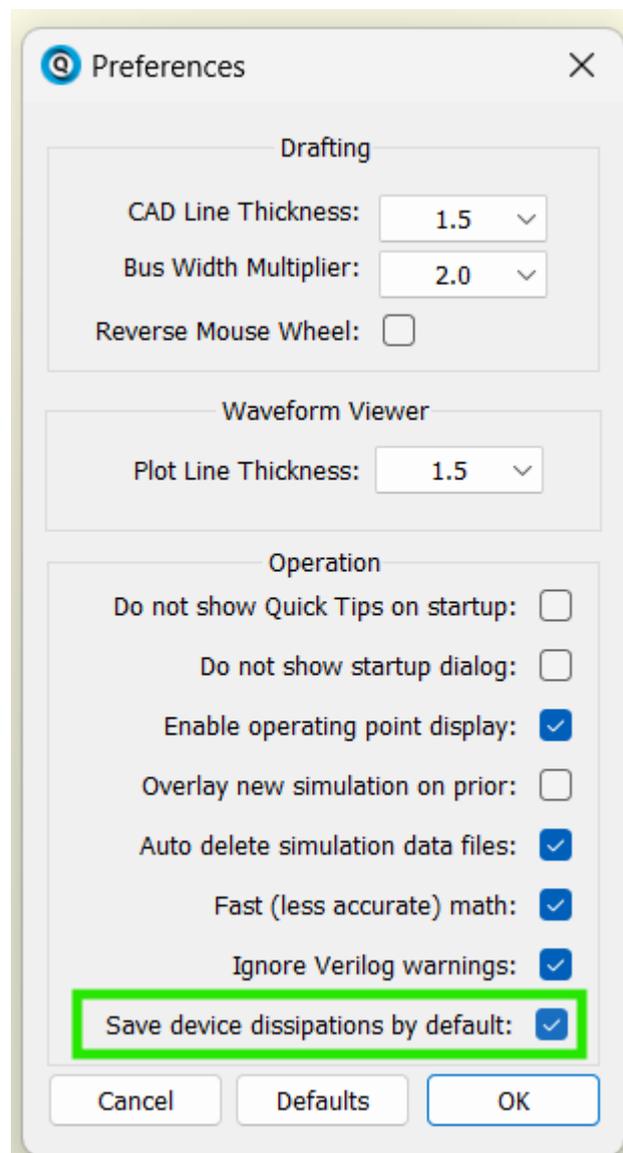
Para este tipo de simulação não faz muito sentido visualizar os resultados em forma de gráfico. A forma mais prática de visualizar as tensões nos nós e as correntes sobre os componentes é simplesmente posicionar o ponteiro do mouse sobre o nó ou o componente desejado. Veja as imagens a seguir.



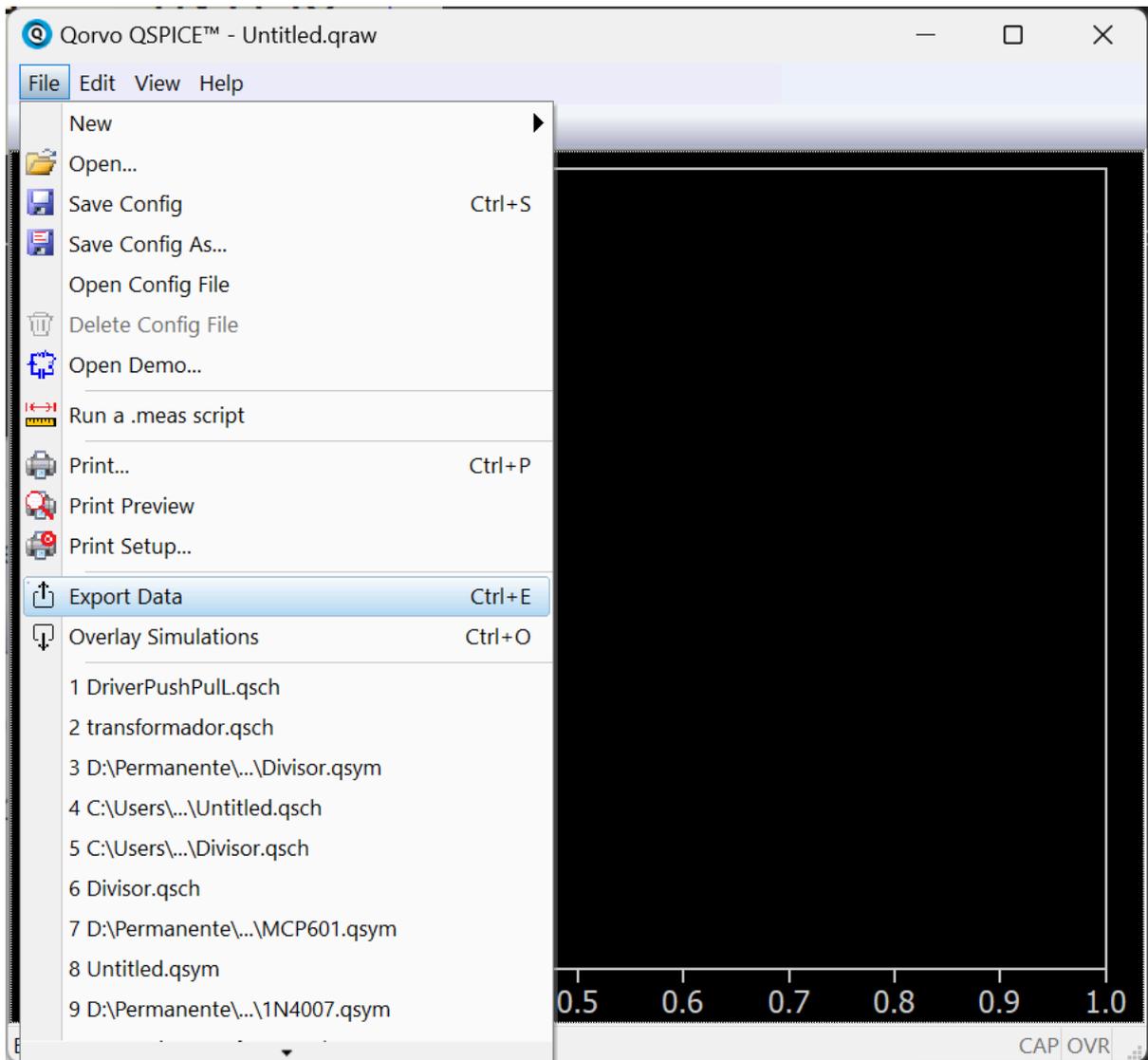
Para verificar os valores de potência nos componentes é necessário segurar a tecla Ctrl enquanto aponta o componente com o mouse. Veja a imagem a seguir.



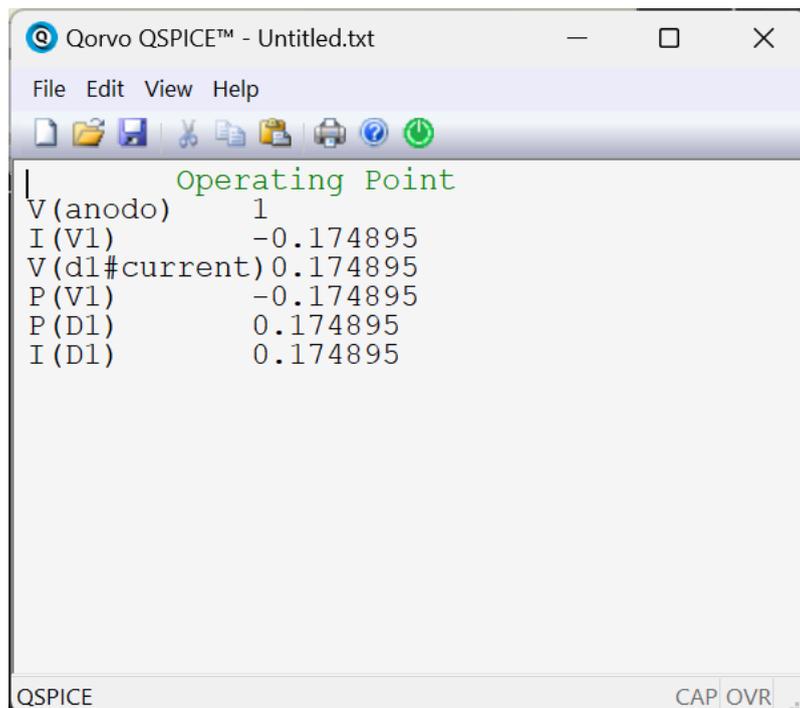
**Importante** o Qspice não salva a potência dissipada pelos componentes com sua configurações padrão, para isso é necessário ativar a opção "Save device dissipations by default", como destacado na imagem a seguir.



Os dados da simulação podem ser exportados para ser utilizados em outras aplicações, para isso é necessário selecionar a opção "Export Data" no menu "File" da janela de visualização dos gráficos, como na imagem a seguir.



Será apresentado um arquivo de texto contendo os resultados da simulação, como na imagem a seguir.



Este tipo de simulação não é comumente utilizado de maneira isolada, pois ele é feito como o primeiro passo de todas as outras análises.

Se um ".op" for invocado sozinho, os dados não são apresentados em um gráfico 2D, mas simplesmente como uma lista de tensões nos nós e correntes nos dispositivos. Os dados são plotados como uma linha reta contra um eixo horizontal sem unidades. No entanto, se incluído com um comando ".step", é possível criar um gráfico 2D bastante interessante.

#### Diretiva

```
.step
```

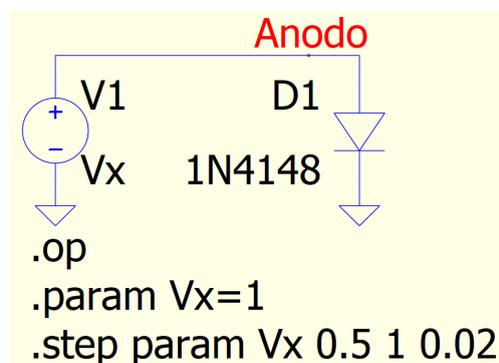
O comando .step é usado para executar a simulação múltiplas vezes, variando um ou mais parâmetros. Para utilizar este comando é necessário primeiro declarar os parâmetros que se deseja variar. Isso é realizado com o comando ".param"

#### Diretiva

```
.param
```

O comando ".param" permite definir parâmetros no formato "nome=valor", por exemplo, ".param Freq = 10K" que define um parâmetro chamado "Freq" com o valor de 10K. Este parâmetro pode ser utilizado em qualquer ponto do circuito.

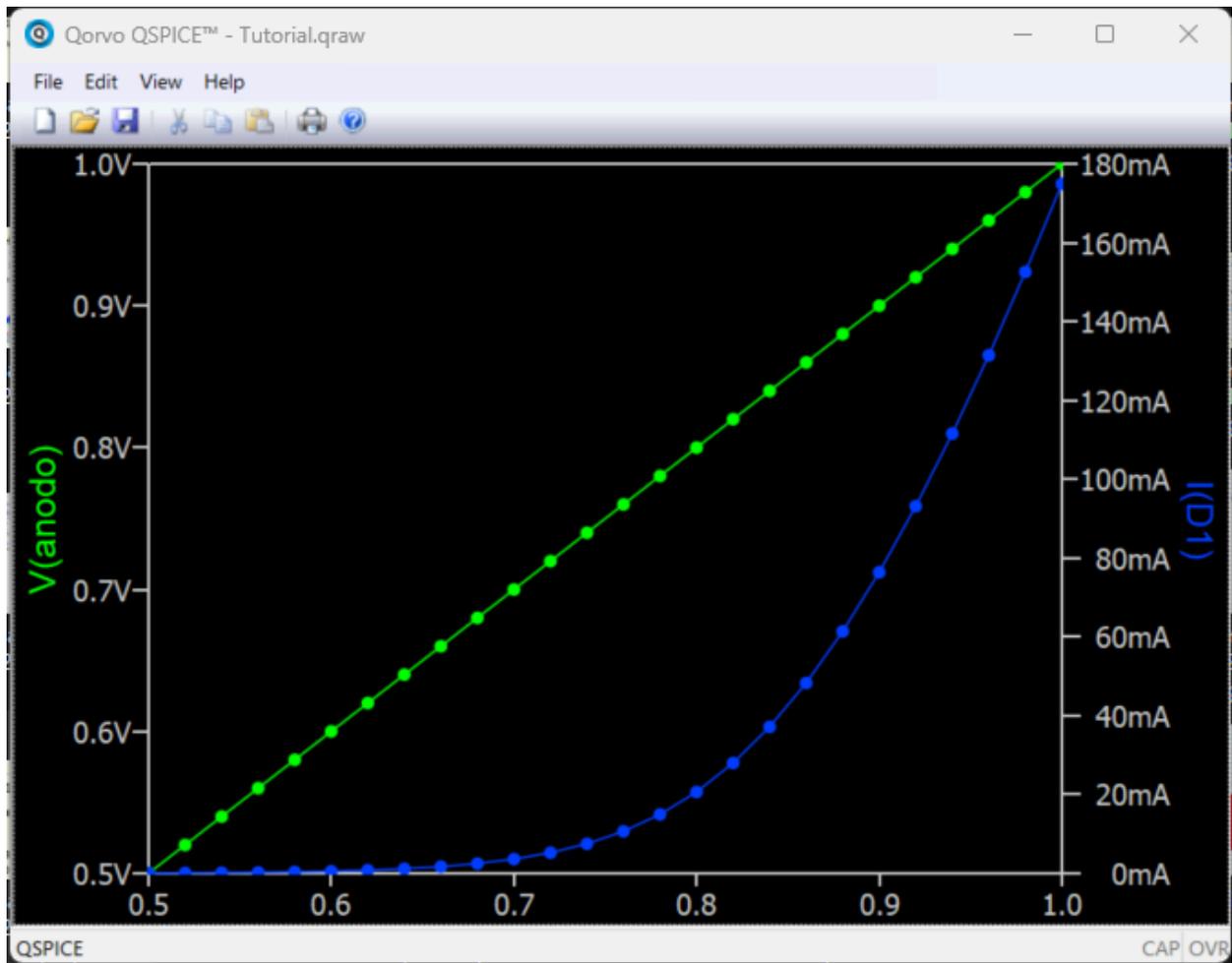
Para exercitar os comandos ".step" e ".param" vamos agora alterar o circuito do exercício anterior para que a tensão da fonte V1 varia de 0.5 V a 1 V em intervalor de 0,02 V. Veja na imagem a seguir como fica.



Neste exercício é criado um parâmetro de nome "Vx", com valor 1. Este parâmetro é atribuído a tensão da fonte V1. Em seguida este parâmetro é variado de 0,5 V a 1 V pelo comando ".step".

Após executar a simulação é possível visualizar o gráfico da tensão clicando sobre o nó "Anodo" e o gráfico da corrente, clicando sobre o diodo.

A imagem a seguir mostra estes gráficos.



É possível observar a evolução das várias simulações executadas com o comando ".step" na janela de saída do simulador, na parte inferior da janela principal. Como mostrado na figura a seguir.

```

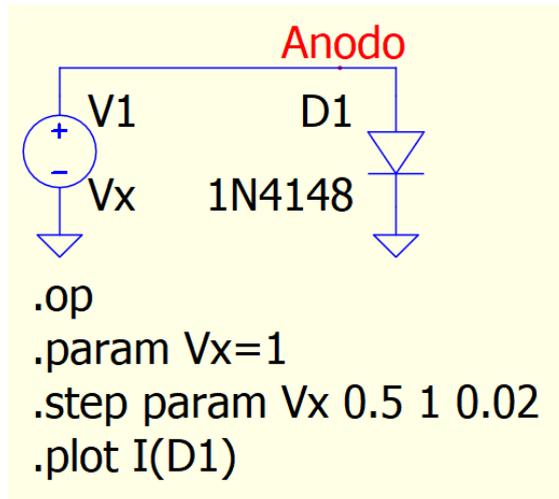
Output Window
1 of 26 steps: .step vx=0.5
2 of 26 steps: .step vx=0.52
3 of 26 steps: .step vx=0.54
4 of 26 steps: .step vx=0.56
5 of 26 steps: .step vx=0.58
6 of 26 steps: .step vx=0.6
7 of 26 steps: .step vx=0.62
8 of 26 steps: .step vx=0.64
9 of 26 steps: .step vx=0.66
10 of 26 steps: .step vx=0.68
11 of 26 steps: .step vx=0.7
12 of 26 steps: .step vx=0.72
13 of 26 steps: .step vx=0.74
14 of 26 steps: .step vx=0.76
15 of 26 steps: .step vx=0.78
16 of 26 steps: .step vx=0.8
17 of 26 steps: .step vx=0.82
18 of 26 steps: .step vx=0.84
  
```

Quando se deseja plotar automaticamente algum sinal é possível adicionar o comando ".plot".

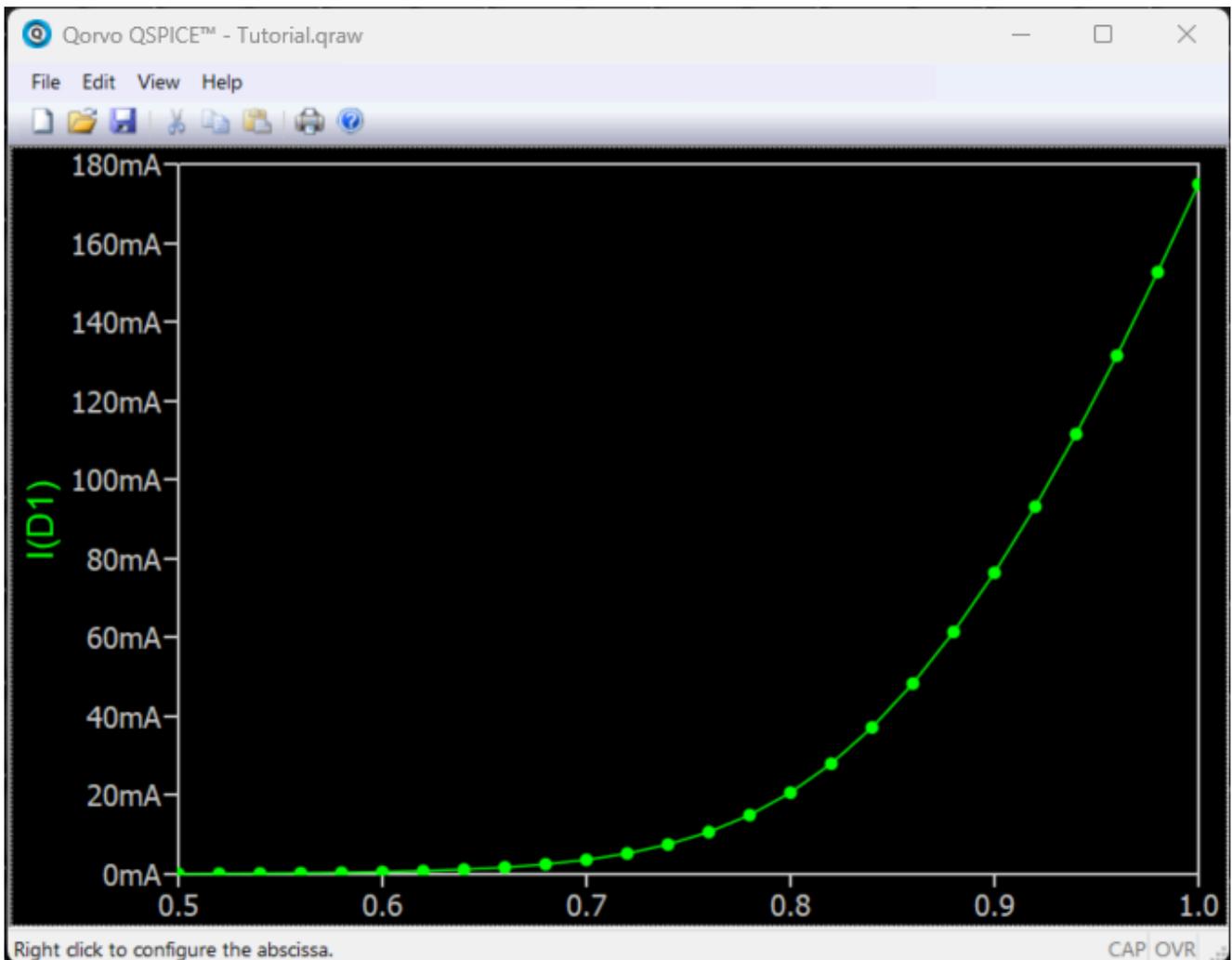
## Diretiva

.plot

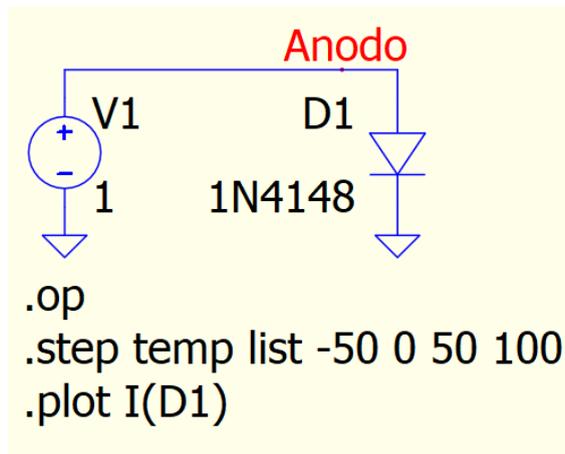
O comando ".plot" instrui o simulador a plotar no gráfico um ou mais sinais. Veja no exemplo da imagem a seguir como plotar a corrente no diodo D1.



A imagem a seguir mostra o resultado desta simulação.

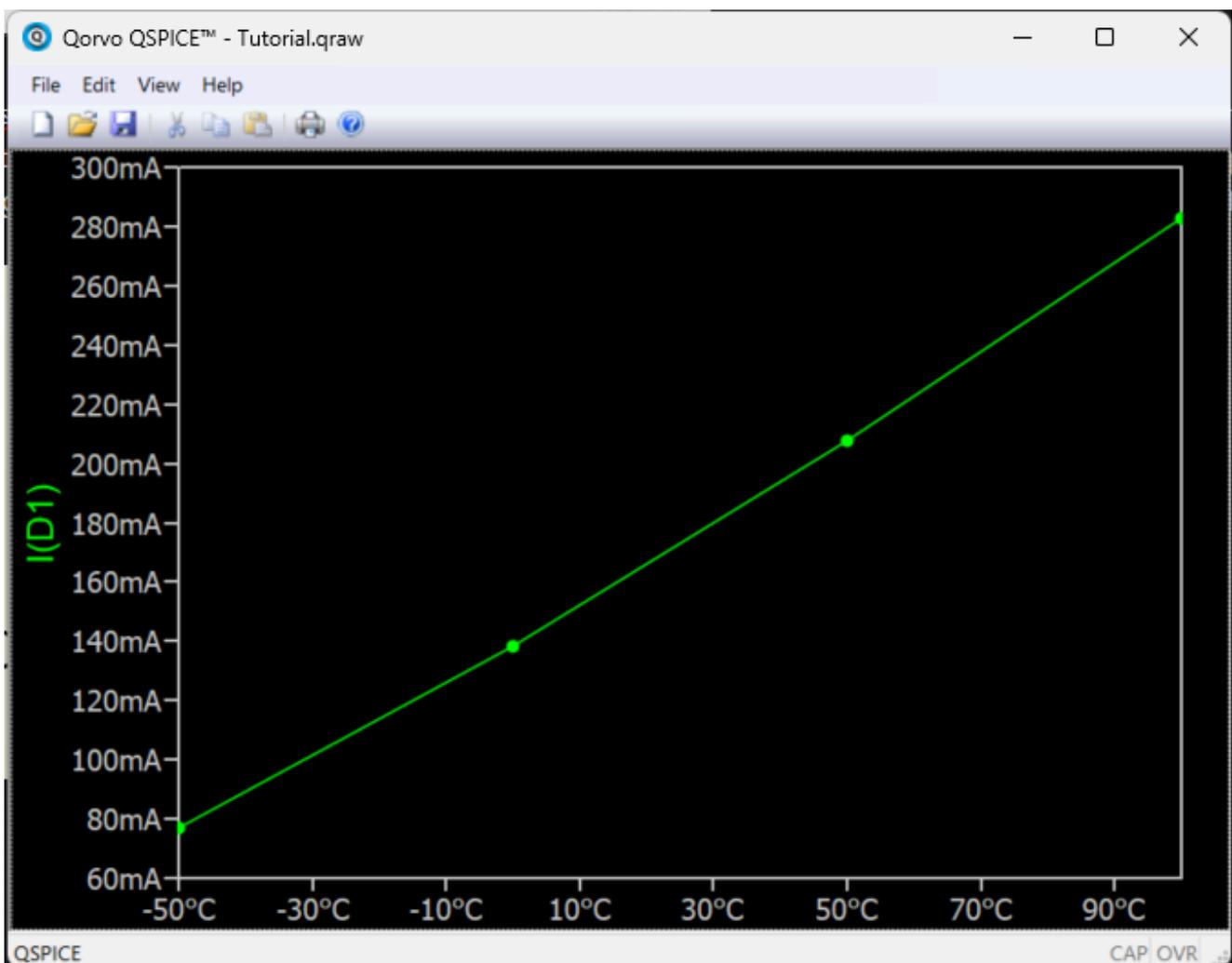


É também possível utilizar o comando ".step" com uma lista de valores. Para ilustrar este conceito, no próximo exemplo vamos variar um parâmetro muito importante, a temperatura. O Qspice utiliza um parâmetro interno de nome "temp" para realizar a simulação. O valor padrão deste parâmetro é 27 °C. A imagem a seguir mostra o circuito sendo simulado em quatro diferentes temperaturas, -50 °C, 0 °C, 50 °C e 100 °C.



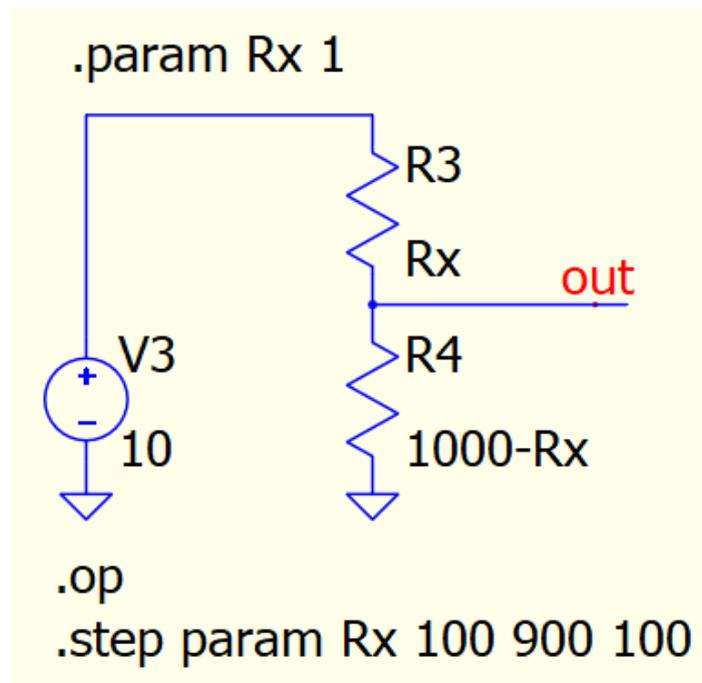
Observe que nesta simulação a tensão da fonte é constante, 1 V.

O gráfico com a variação da corrente no diodo D1 em função da temperatura é mostrado na imagem a seguir.

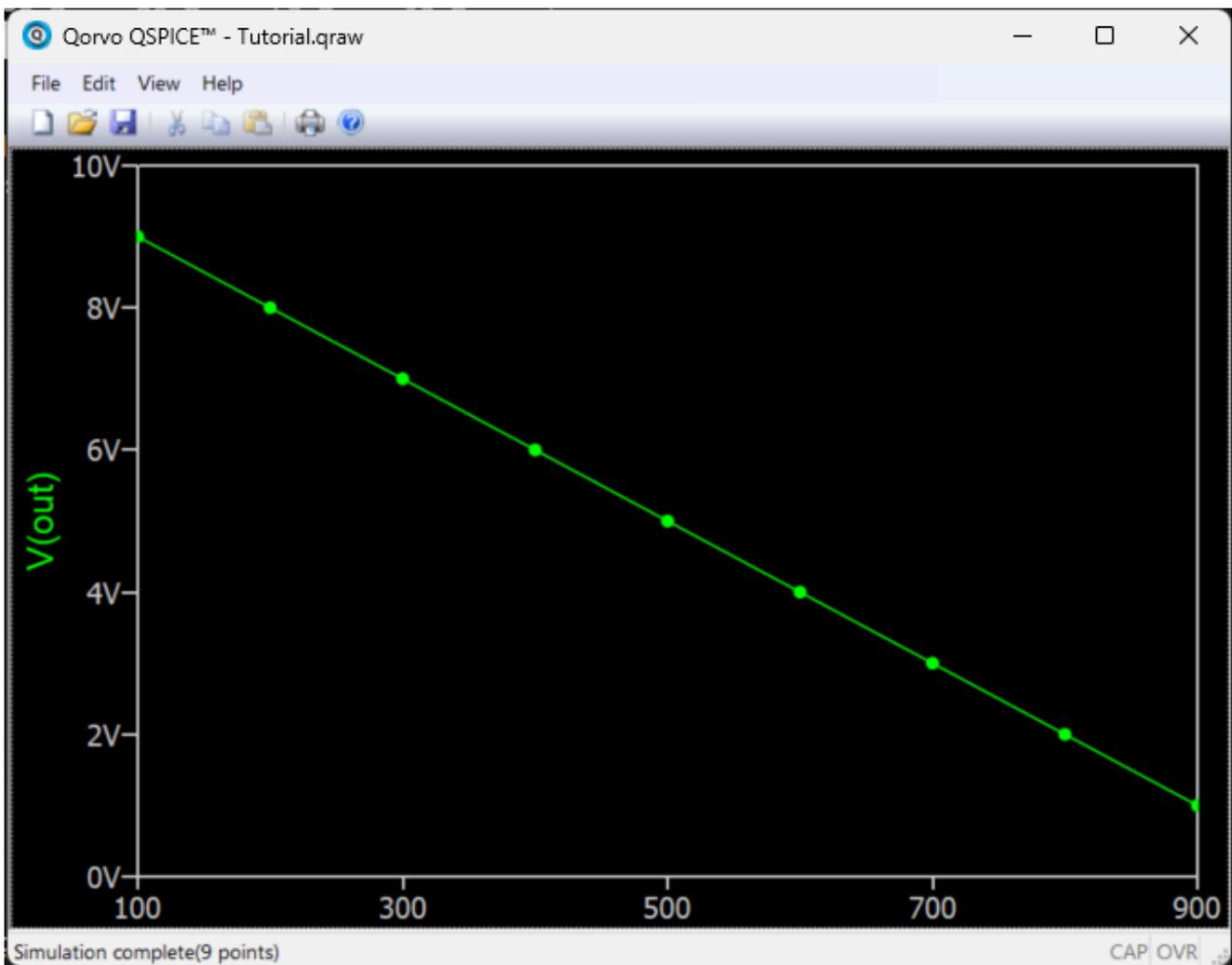


Com o comando ".step" é possível variar o valor de qualquer componente e ver o resultado da simulação do circuito. Veja na imagem a seguir a simulação de um divisor de tensão com

vários valores para os resistores.



A imagem a seguir mostra a tensão da saída em função da variação dos resistores.



Porém, se o que se deseja é varia apenas a tensão ou a corrente das fontes e a temperatura, existe uma forma mais eficiente faze-lo, a simulação de varredura DC, que será apresentada a seguir.

Este material irá apresentar apenas as funcionalidades básicas de algumas das principais funções ou diretivas do simulador.

É importante acessar os arquivos de ajuda correspondentes para maiores informações.

## Simulação de varredura DC (DC Sweep: .dc)

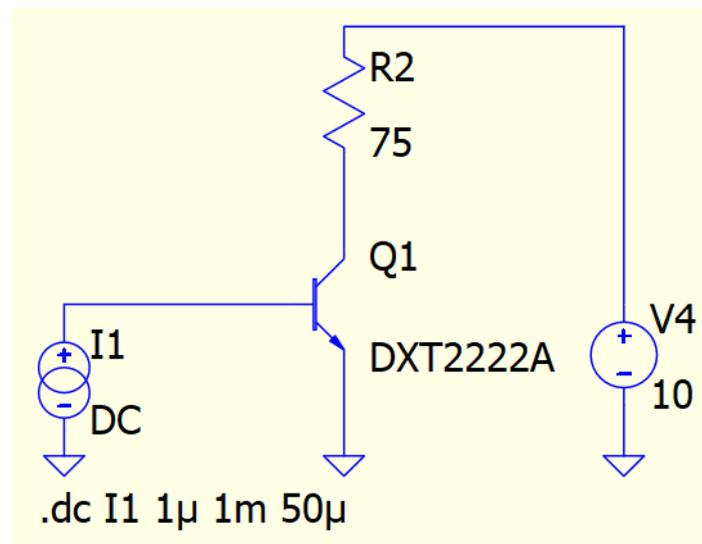
O comando de varredura DC, ".dc", plota o ponto de operação DC de um circuito enquanto varia uma fonte de tensão, uma fonte de corrente ou a temperatura do circuito. Até seis fontes podem ser varridas.

O comando é usado principalmente para traçar curvas de transistores.

### Diretiva

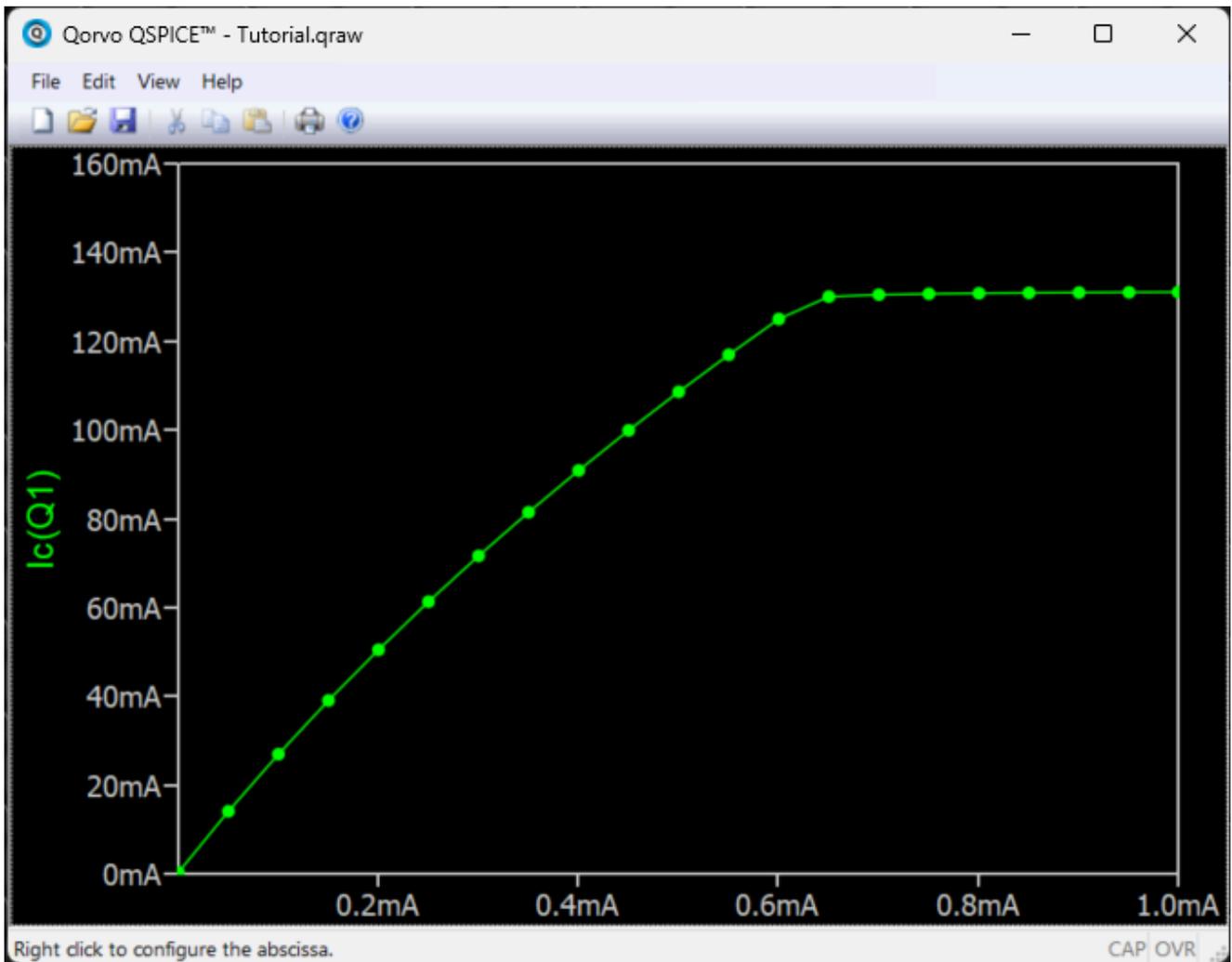
.dc

Como exemplo, vamos analisar a corrente do coletor de um transistor em função da variação da corrente de sua base. A imagem a seguir mostra este circuito.

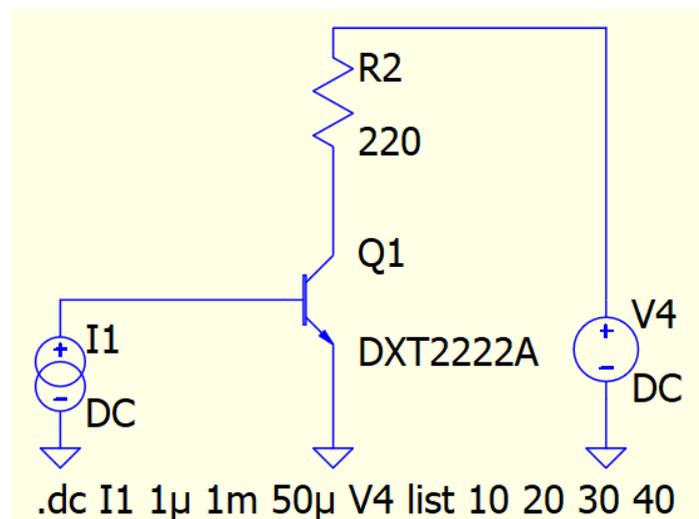


Observe que o valor da fonte de corrente é variado pelo comando ".dc" de 1 uA a 1 mA em intervalos de 50 uA.

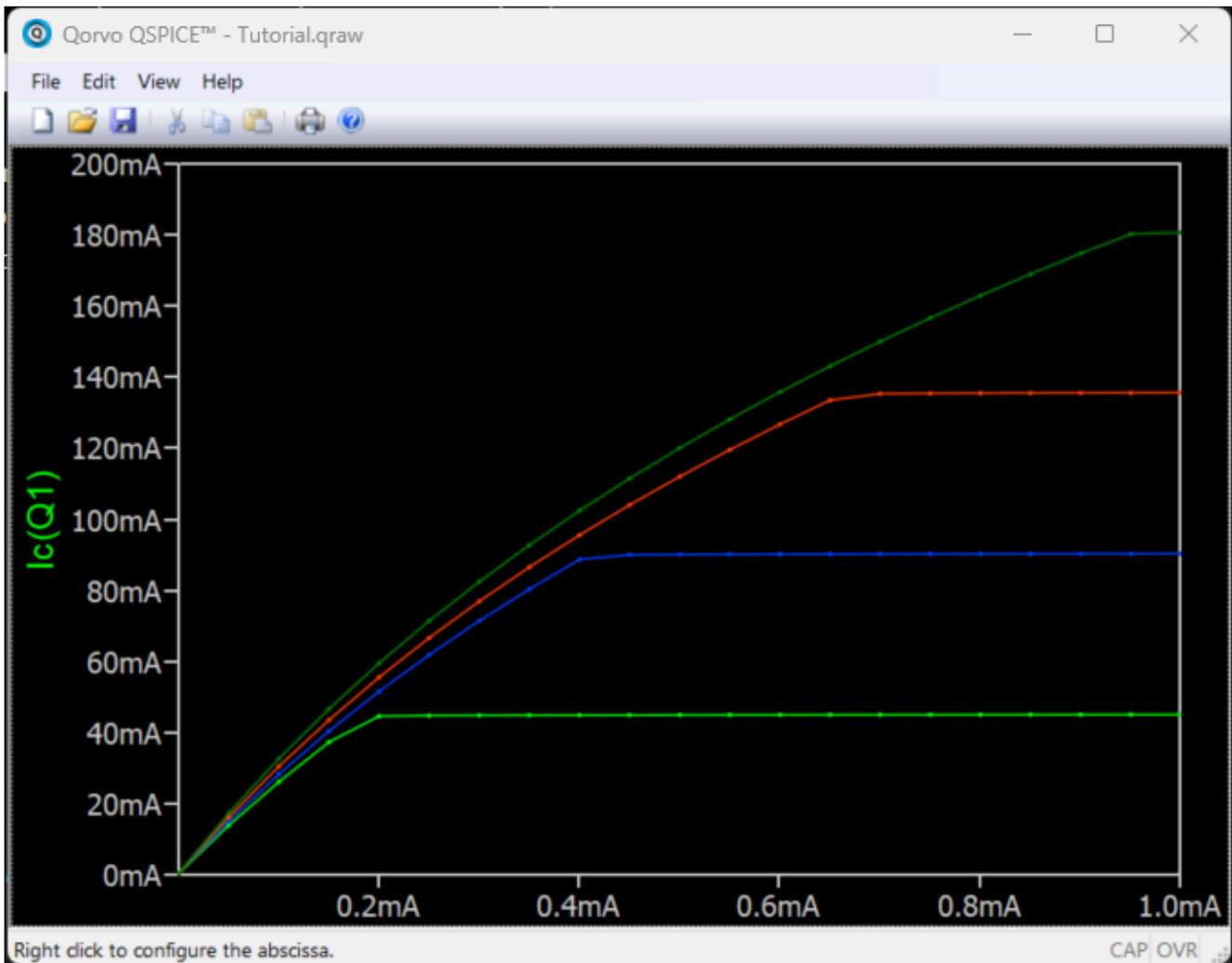
A corrente de coletor é mostrada no gráfico da figura a seguir.



É possível variar vários parâmetros da simulação ao mesmo tempo. Na imagem a seguir é apresentada uma simulação onde a fonte de corrente I1 tem a mesma variação do exemplo anterior, mas a fonte de tensão V4 possui quatro valores diferentes, 10 V, 20 V, 30 V e 40V. Neste caso a simulação plota várias curvas, relacionadas as várias combinações dos parâmetros.



Assim, na imagem a seguir é apresentada a corrente do coletor em função da corrente da base para quatro diferentes tensões da fonte de tensão.



## Simulação AC (AC Analysis: .ac)

Neste tipo de simulação a resposta no domínio da frequência é calculada. O domínio da frequência implica em um circuito linear, uma vez que as frequências são independentes apenas em um sistema linear. Portanto, um circuito não linear é primeiro linearizado.

O procedimento consiste em:

1. encontrar o ponto de operação DC do circuito
2. expandir os componentes não lineares como uma série de Taylor
3. manter apenas o termo de primeira ordem, descartando o termo de ordem zero e os termos de ordem superior.

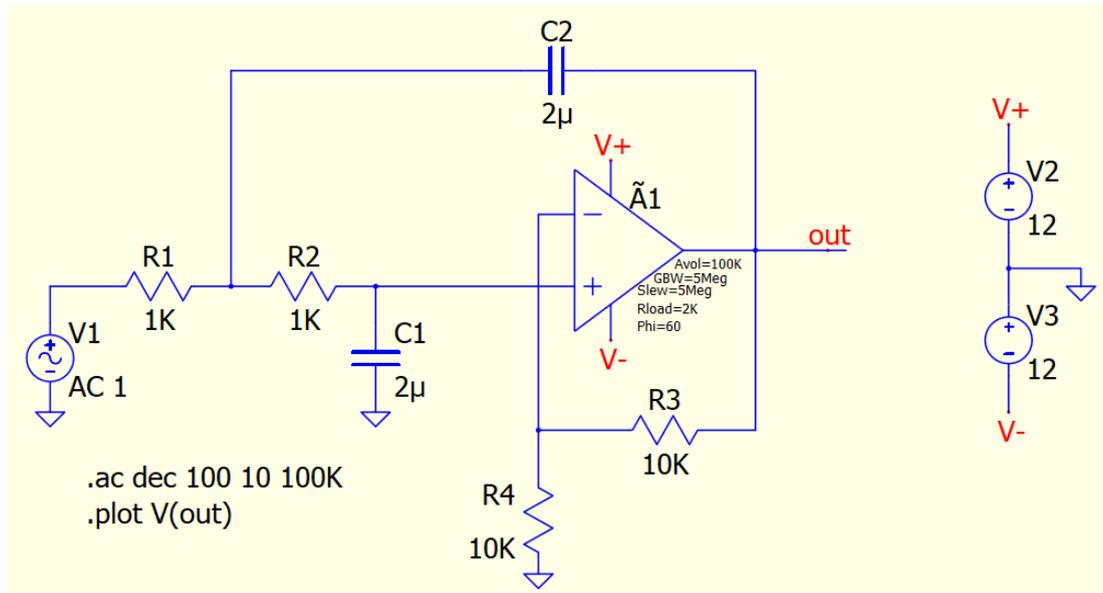
A técnica é eficaz para circuitos de tempo contínuo, ou seja, circuitos sem chaveamento.

### Diretiva

.ac

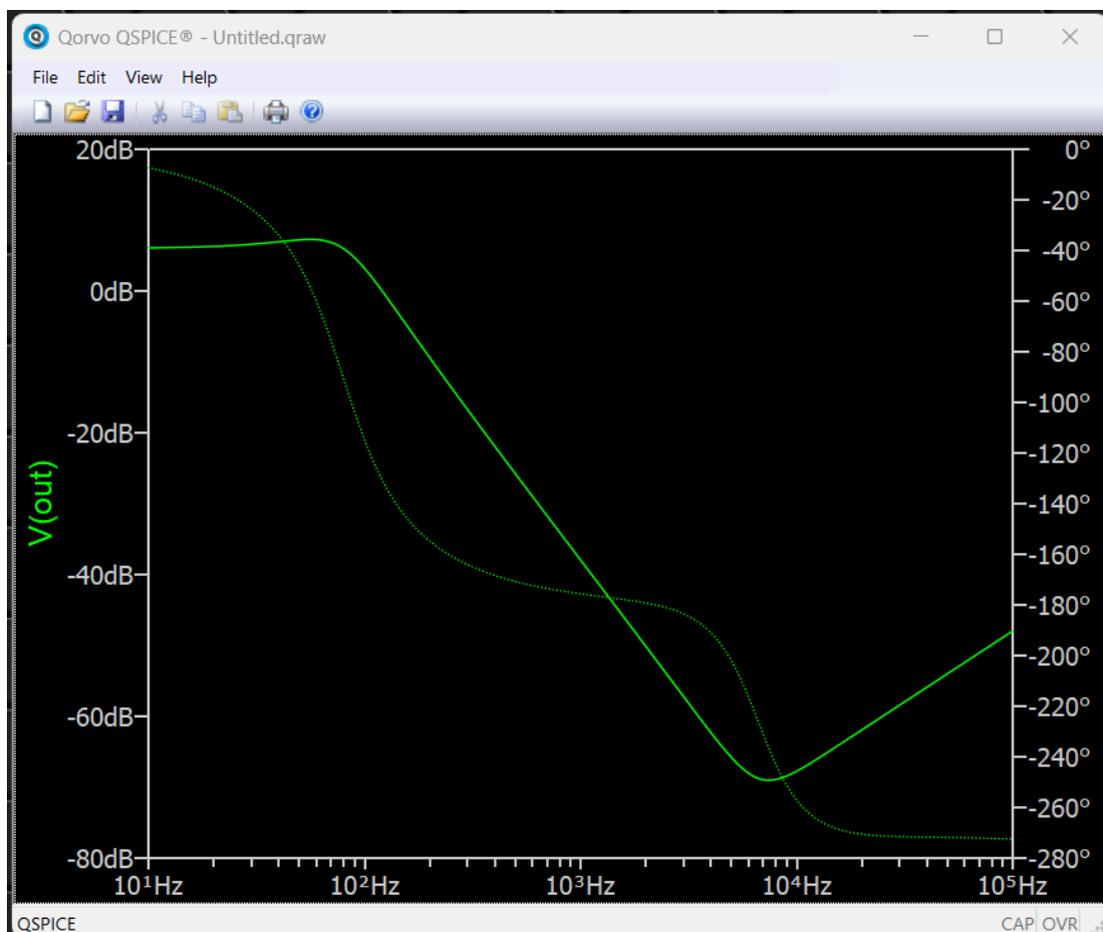
Este tipo de simulação é muito importante pois permite analisar por exemplo a resposta em frequência de um filtro, plotando seu ganho e fase.

Como exemplo veja na imagem a seguir o circuito de um filtro passa baixas de segunda ordem.



Neste exemplo temos uma simulação AC com 100 pontos por década, com frequência inicial de 10 Hz e frequência final de 100 KHz. É importante destacar que o sinal de entrada a ser aplicado ao circuito deve vir de uma fonte com valor "AC 1", como na fonte V5 do exemplo.

O resultado é apresentado na imagem a seguir.



Observe que o simulador plota ao mesmo tempo a amplitude e a fase do sinal de saída.

A saída da simulação AC é tipicamente apresentada em dB (decibéis), uma unidade logarítmica utilizada para expressar a razão entre dois valores, como potência, tensão ou corrente. O ganho em dB é uma medida da amplificação ou atenuação de um sinal à medida que ele passa por um circuito.

O cálculo do ganho em dB para tensão ou corrente é dado pela fórmula:

$$Ganho(dB) = 20 \times \log_{10} \left( \frac{V_{saída}}{V_{entrada}} \right)$$

Para potência, a fórmula é:

$$Ganho(dB) = 10 \times \log_{10} \left( \frac{P_{saída}}{P_{entrada}} \right)$$

Um ganho positivo em dB indica amplificação, enquanto um valor negativo indica atenuação. Por exemplo, um ganho de 20 dB significa que o sinal de saída é 10 vezes maior que o de entrada, em termos de tensão (ou corrente), enquanto um ganho de -3 dB indicaria que o sinal de saída é aproximadamente 70% do sinal de entrada.

A apresentação em dB é preferida em análises de circuitos porque facilita a visualização de amplificações e atenuações em larga escala, especialmente em análises de resposta em frequência, onde as variações de ganho podem ser expressas em uma escala logarítmica que é mais intuitiva e manejável para comparações de múltiplas frequências.

### Valores comuns para referência

Ganho em dB	Ganho em Tensão
-60	0.001
-55	0.002
-50	0.003
-45	0.006
-40	0.010
-35	0.018
-30	0.032
-25	0.056
-20	0.100
-15	0.178
-10	0.316
-5	0.562
0	1.000
5	1.778
10	3.162

Ganho em dB	Ganho em Tensão
15	5.623
20	10.000

A fase por sua vez é apresentada diretamente em graus.

## Simulação Transiente (Non-Linear Transient Analysis: .tran)

A análise transiente não linear é o que leva os engenheiros a usarem o SPICE. Ela simplesmente calcula o comportamento do circuito como se você o construísse e o energizasse.

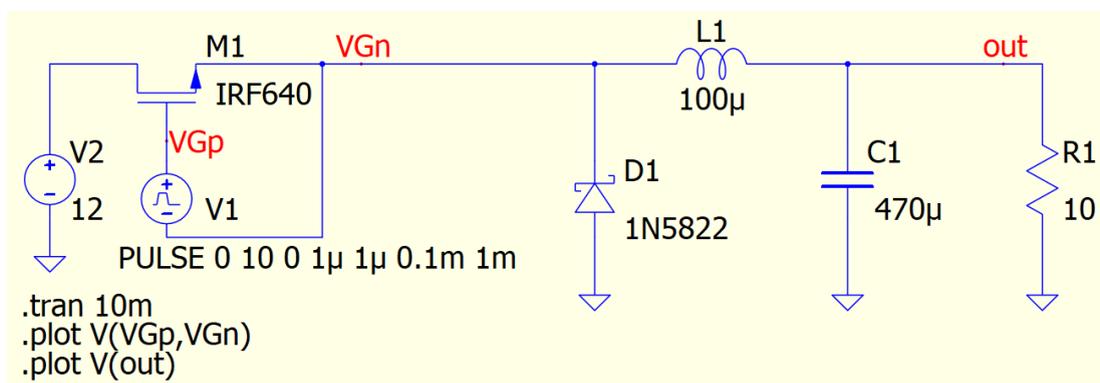
Esta análise, representada pela diretiva ".tran" no Qspice, é uma das ferramentas mais importantes e amplamente utilizadas em simulações de circuitos eletrônicos. Ela permite que os engenheiros analisem como um circuito responde a estímulos variáveis ao longo do tempo. Essa capacidade é crucial para entender como os componentes interagem dinamicamente e como o circuito se estabiliza ou reage a mudanças abruptas, como a aplicação de uma tensão ou corrente.

### Diretiva

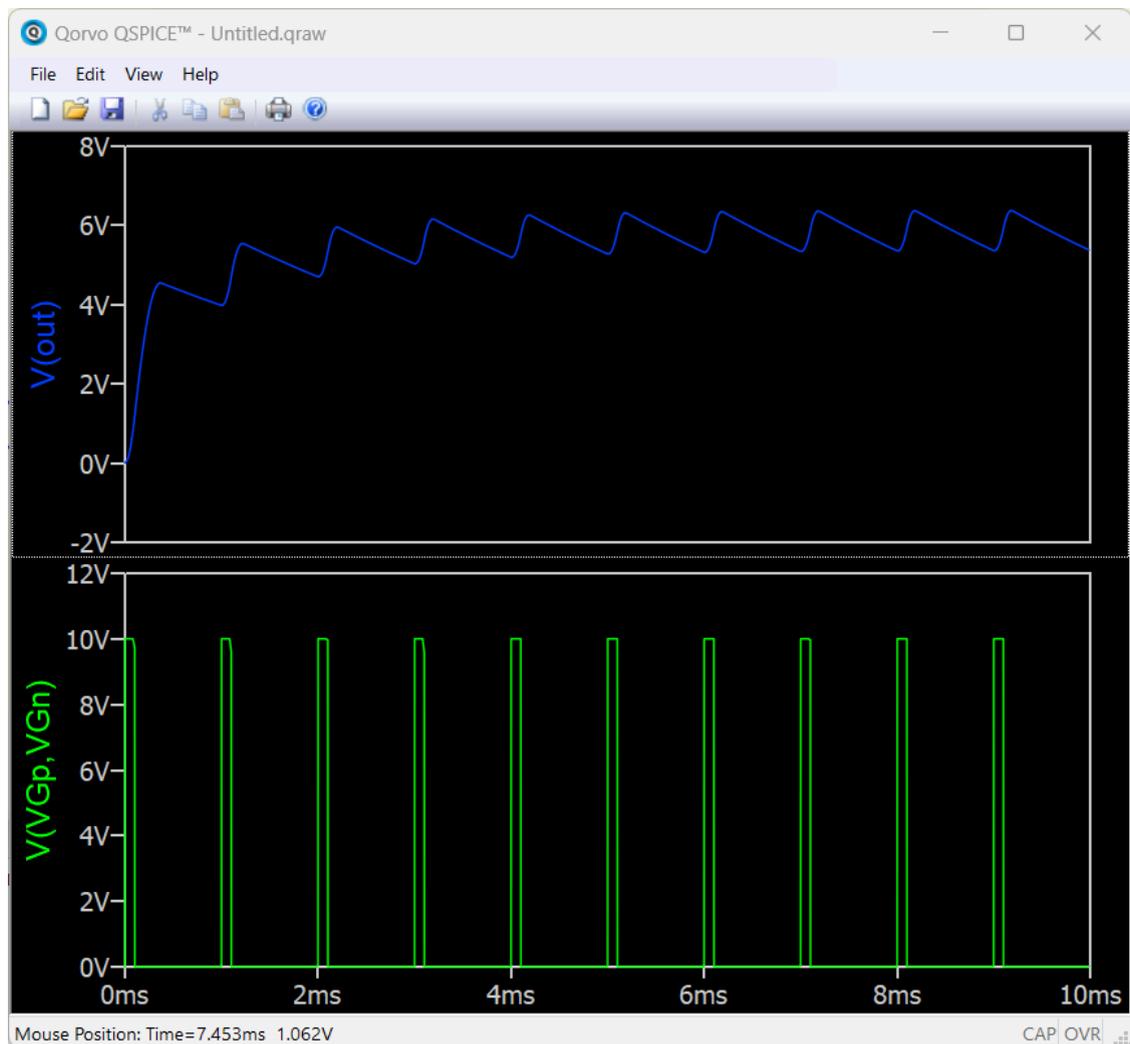
.tran

A análise transiente calcula as variáveis de estado do circuito (como tensões e correntes) em função do tempo, considerando os efeitos não lineares dos componentes, como diodos, transistores e indutores saturáveis. Ao simular o circuito ao longo do tempo, o Qspice resolve uma série de equações diferenciais que modelam o comportamento dos componentes. Isso permite observar, por exemplo, como um capacitor carrega e descarrega, como um transistor chaveia ou como um circuito ressonante responde a uma excitação.

O comando ".tran" aceita diversos parâmetros. A forma mais simples é inserir o tempo desejado de simulação, por exemplo, ".tran 100m" para simular o circuito por 100 ms. Como exemplo vamos simular o circuito da imagem a seguir.



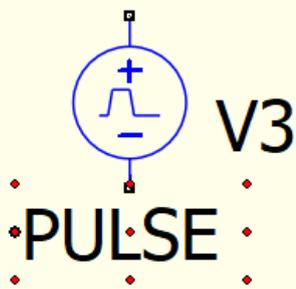
Neste exemplo o circuito é simulado por um período de 10 ms. O resultado plota duas curvas, uma com a tensão na saída e outra com a tensão diferencial entre os nós VGp e VGn. O resultado é apresentado na imagem a seguir.



Observe que foram utilizados dois comandos ".plot", assim a janela de saída mostra dois gráficos separados.

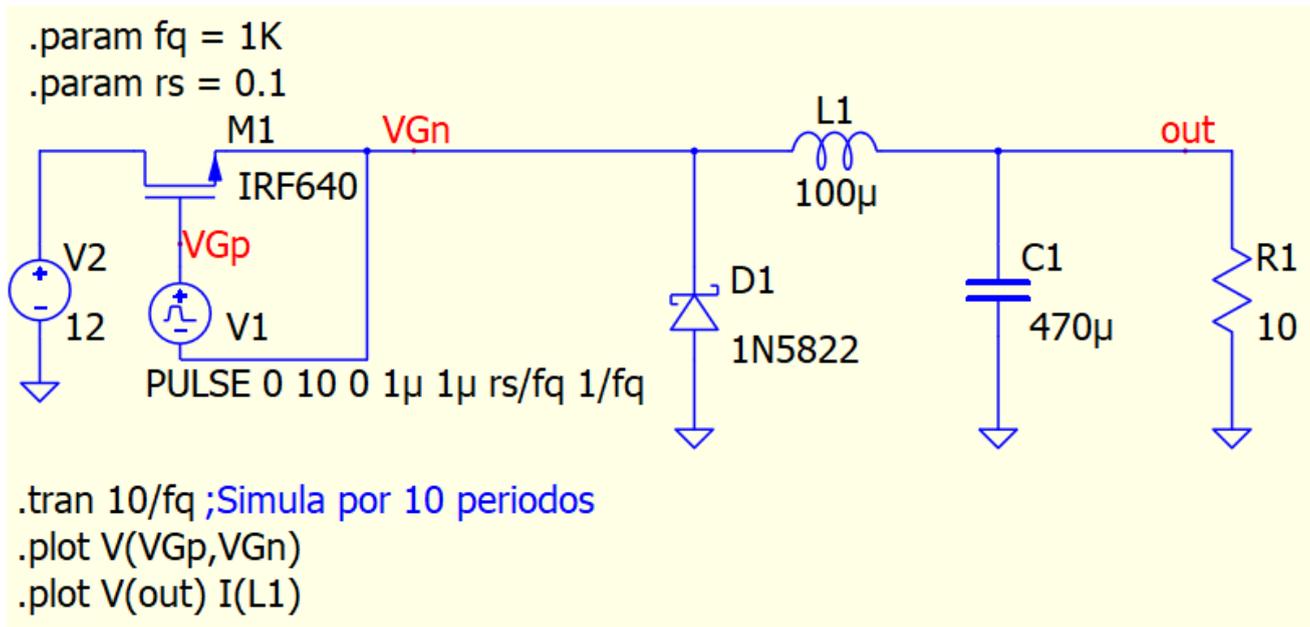
Neste exemplo também é utilizado um tipo diferente de fonte. A fonte V1 é do tipo PULSE, e está ajustada para fornecer pulsos de 10 V, com tempo de subida e descida de 1  $\mu$ s, tempo ligado de 0,1 ms e período total de 1 ms.

O Qspice possui um sistema de ajuda que facilita a configuração dos diversos componentes. Observe na imagem a seguir que quando o usuário está digitando os parâmetros da fonte é apresentado em azul um texto mostrando o nome e a posição de cada um dos parâmetros.



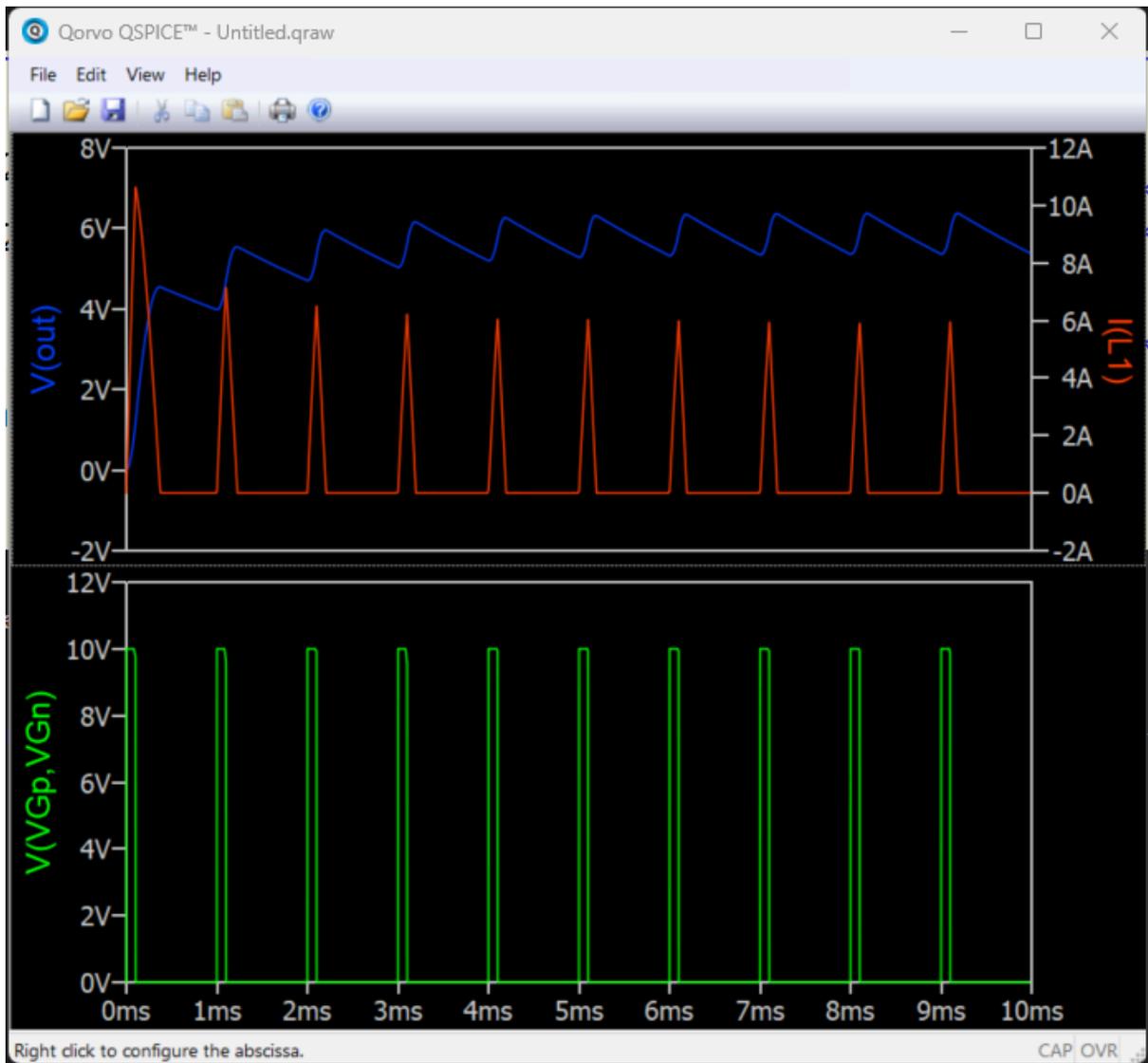
`PULSE <VALUE1> VALUE2 TDELAY TRISE [TFALL] [TON] [TPERIOD] [NCYCLES]`

Uma forma interessante de descrever os valores dos componentes é utilizar o comando ".param" apresentado anteriormente. Veja na imagem a seguir o exemplo anterior alterado para utilizar o comando ".param".



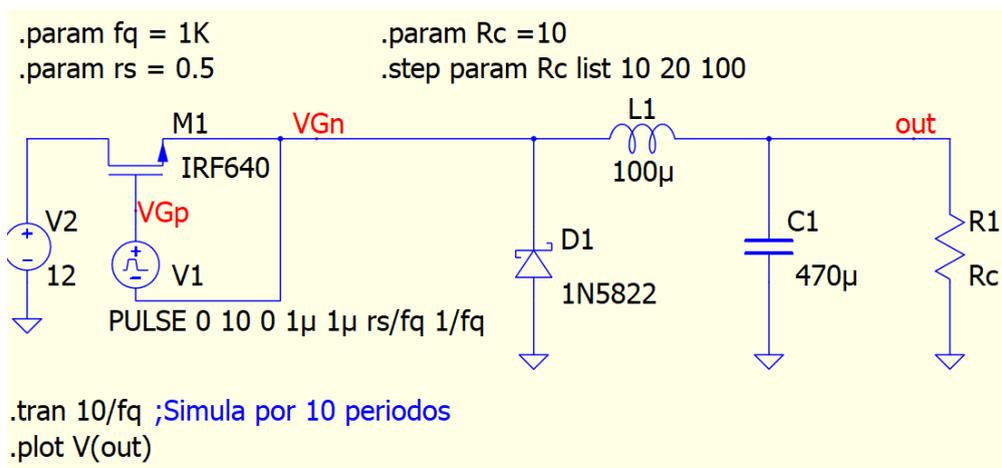
Observe que foram criados dois parâmetros, "fq" para armazenar a frequência do PWM (Fonte de pulsos) e "rs" para a razão cíclica. Estes dois valores são então utilizados para configurar o comando ".tran" e a fonte de pulsos.

Foi também adicionada uma segunda curva ao segundo comando ".plot". Veja na imagem a seguir como fica a saída.



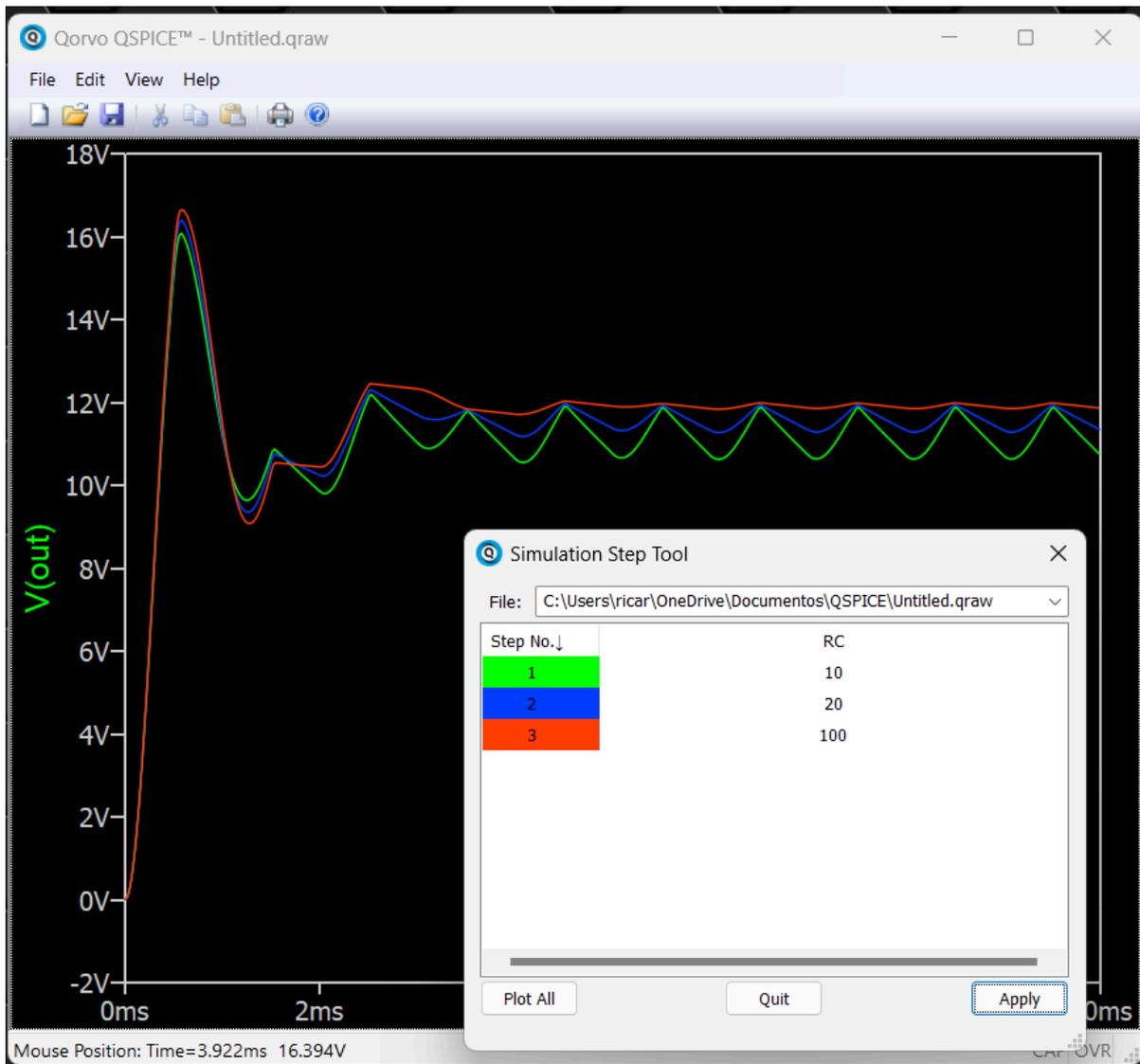
A curvas são adicionadas a janela de saída na ordem inversa dos comandos ".plot" no circuito.

É possível também combinar a simulação de transiente com o comando ".step", apresentado anteriormente. Na imagem a seguir o parâmetro "Rc" é adicionado para armazenar o valor do resistor de carga. São então realizadas três simulações com três valores diferentes para Rc.



Na imagem a seguir são apresentados os três gráficos para a tensão de saída. Com a janela de saída selecionada, se for pressionada a tecla "F6", é apresentada uma janela com

os valores da simulação relacionados a cada cor do gráfico.



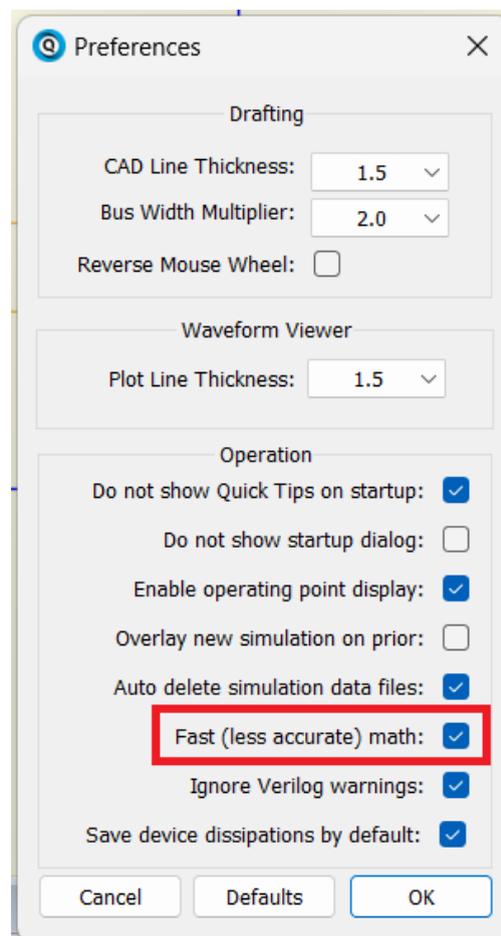
**Importante:** Neste tipo de análise é possível que a simulação não convirja. Neste caso é apresentada a mensagem "Fatal error: Timestep too small", como na imagem a seguir.

```
Output Window
C:\Users\ricar\OneDrive\Área de Trabalho\exemplo.qsch
Fatal error: Timestep too small(1.69854e-15) at t=0.0135874
```

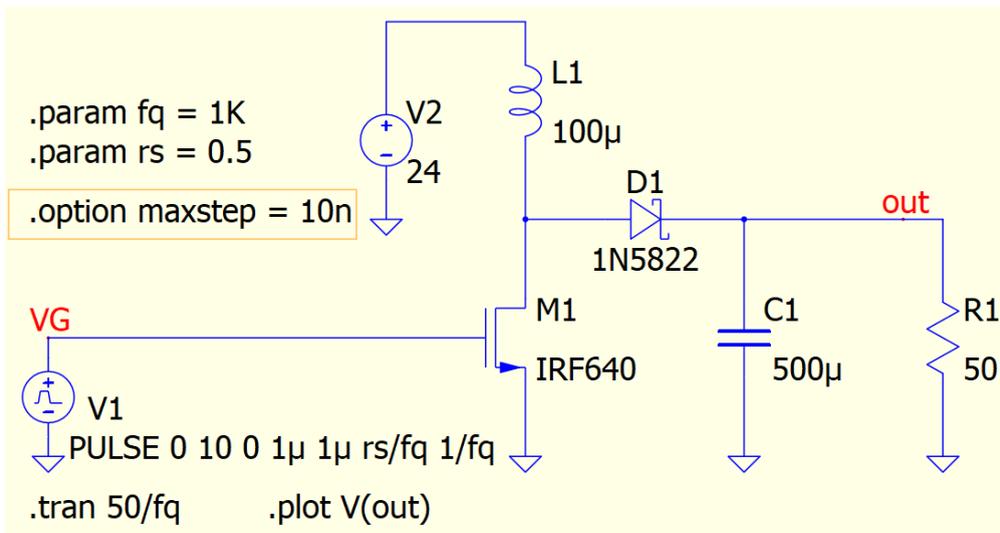
Esse problema geralmente está associado a circuitos que têm não-linearidades fortes. Componentes como diodos e transistores em certas configurações podem causar mudanças abruptas de estado, que exigem passos de tempo muito pequenos para capturar corretamente o comportamento do circuito. Elementos de circuito mal condicionados como resistores, indutores ou capacitores com valores extremamente altos ou baixos podem resultar em um comportamento que exige passos de tempo muito pequenos. Outra causa são circuitos de chaveamento que alternam rapidamente entre estados, neste caso, o simulador pode ter dificuldades para acompanhar essas mudanças rápidas com passos de tempo maiores.

Para resolver este tipo de problema existem algumas abordagens possíveis.

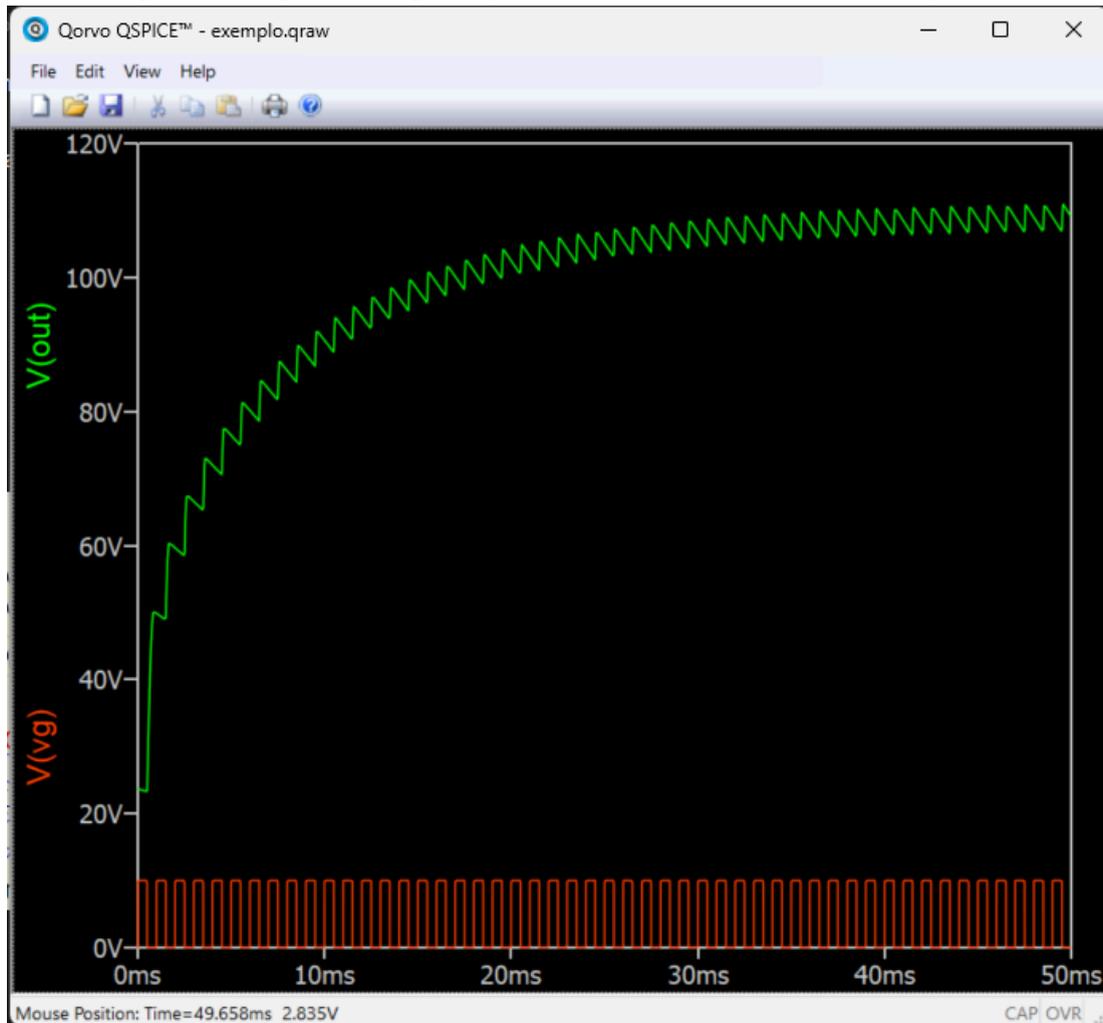
1. Verificar o circuito buscando componentes ou configurações com as características citadas anteriormente.
2. Diminuir os passos de tempo da simulação. Isso pode ser feito adicionando a diretiva ".option maxstep = x", onde x é o intervalo de tempo desejado. É importante ter em mente que quanto menor o tempo estipulado maior é a precisão da simulação, porém, o tempo necessário para executar a simulação, o esforço computacional e o espaço em disco necessário para salvar os resultados são também muito maiores. O ideal é encontrar o equilíbrio entre precisão e esforço computacional.
3. A opção ".option CSHUNT = x", onde x é um valor de capacitância, adiciona um capacitor entre cada nó do circuito e o GND, isso pode ajudar a simular circuitos mal comportados, porém é importante considerar as consequências da adição deste capacitor no comportamento do circuito. Por exemplo, ".option CSHUNT = 1p" Adiciona um capacitor de 1 pF.
4. Outra opção que pode ajudar nestes casos é acessar as preferências do Qspice e desabilitar a opção "Fast (less accurate) math". Esta opção permite escolhes entre variáveis de ponto flutuante de 64 ou 80 bits durante o processo de simulação. Veja a imagem a seguir.



A imagem a seguir apresenta um circuito que para ser simulado necessita de um "step" menor que o padrão do Qspice. Observe o comando ".option maxstep" ajustando o intervalo de tempo para 10 ns.



A título de ilustração a imagem a seguir apresenta a saída do simulador para este circuito.

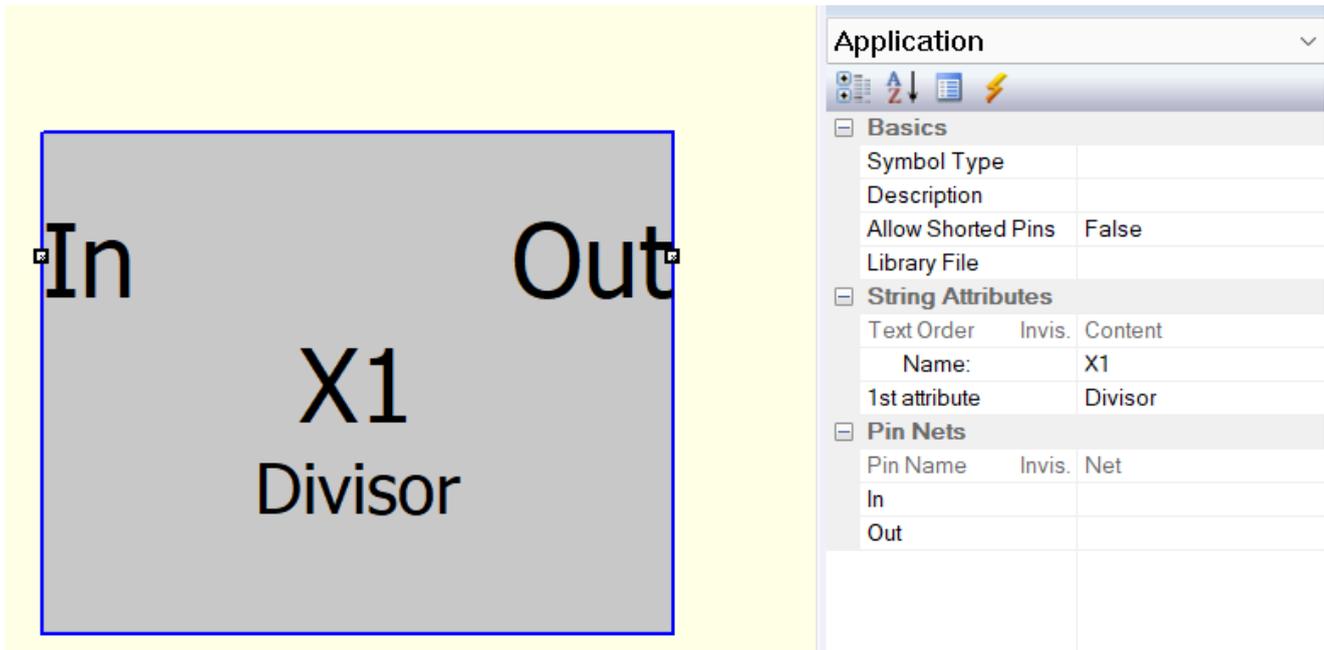


Esta seção apresentou alguns dos principais tipos de simulação realizados pelo Qspice. Existem ainda outros tipos, e os arquivos de ajuda do Qspice fornecem informações sobre elas.

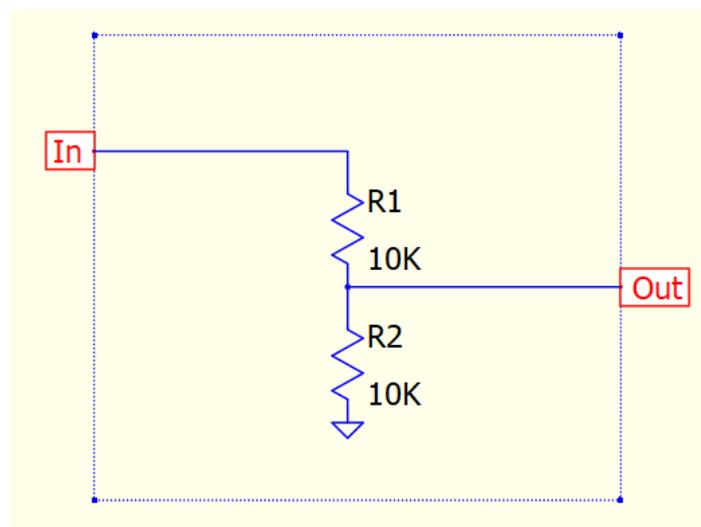
## 6. Inclusão de Sub-Circuitos

### Sub-Circuitos ou Hierarquia de Esquemáticos

O Qspice suporta a captura de esquemáticos hierárquicos, ou seja, é possível inserir um esquemático dentro de outro. A ideia é que um símbolo represente outro arquivo de esquemático no mesmo diretório e com o mesmo nome do símbolo. Os pinos do símbolo são chamados de portas, onde a posição e a ordem dos pinos não é significativa, mas o nome da porta é. Esses nomes de portas se referem aos nomes das redes do esquemático apontado pelo símbolo. O Tipo de Símbolo, mostrado nas propriedades, ficará em branco para uma entrada hierárquica, já que o símbolo não é um dispositivo SPICE. Veja um exemplo na imagem a seguir.



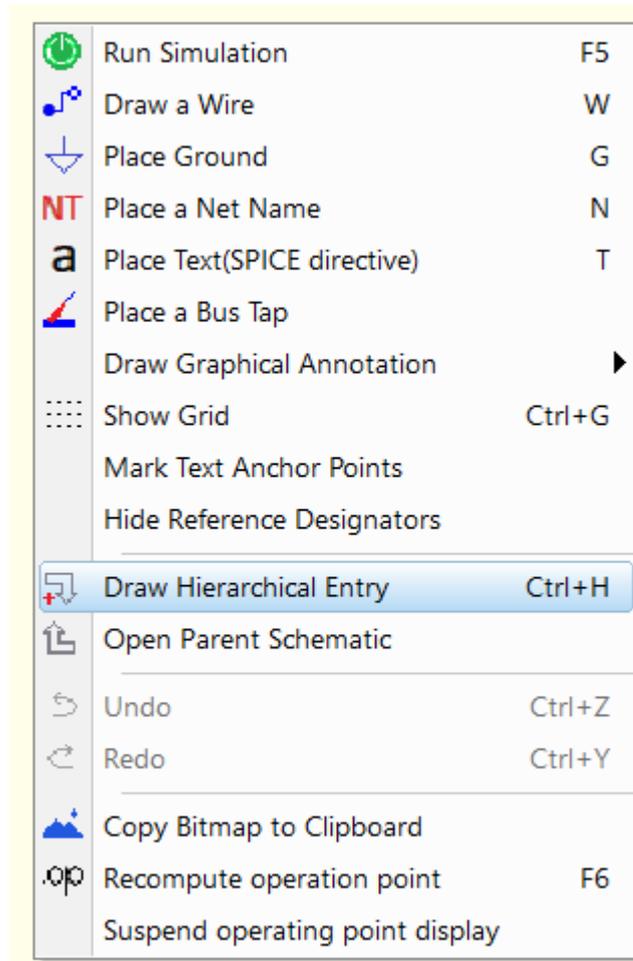
A forma mais fácil de desenhar projetos hierárquicos é deixando o Qspice fazer a maior parte do trabalho para você. Tanto as abordagens de cima para baixo quanto de baixo para cima são permitidas. Como exemplo vamos construir um sub-circuito muito simples contendo um divisor de tensão como o da imagem a seguir.



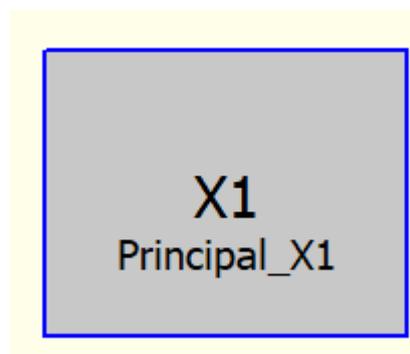
## Abordagem de Cima para Baixo (Top-Down)

1. Salve seu esquemático de nível superior em um diretório que você deseja usar para seu design hierárquico.

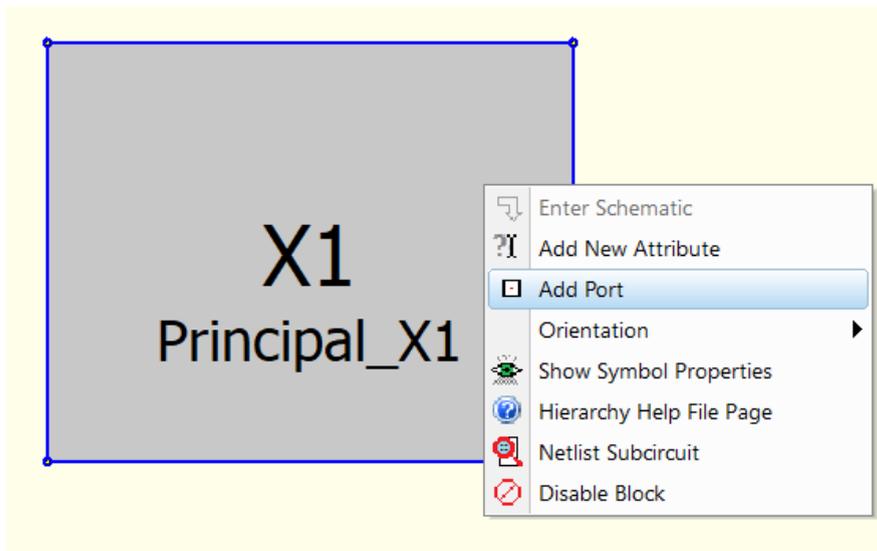
2. Clique com o botão direito em uma área vazia da tela e clique em "Draw Hierarchical Entry", como na imagem a seguir.



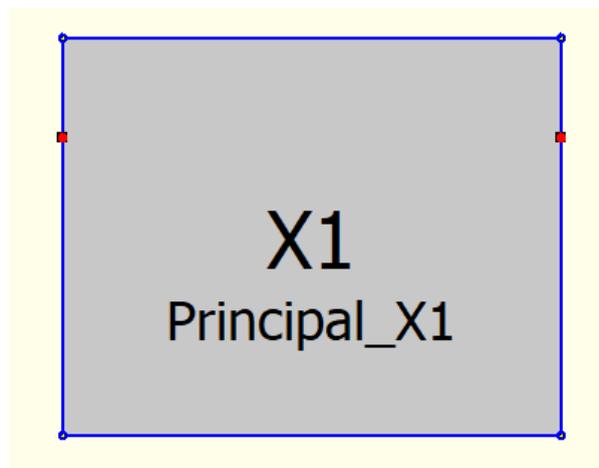
3. Arraste e desenhe uma caixa grande o suficiente para os nomes das suas portas e o nome do esquemático. Não se preocupe. Você pode editar o tamanho da caixa mais tarde para ajustá-la, veja a imagem a seguir.



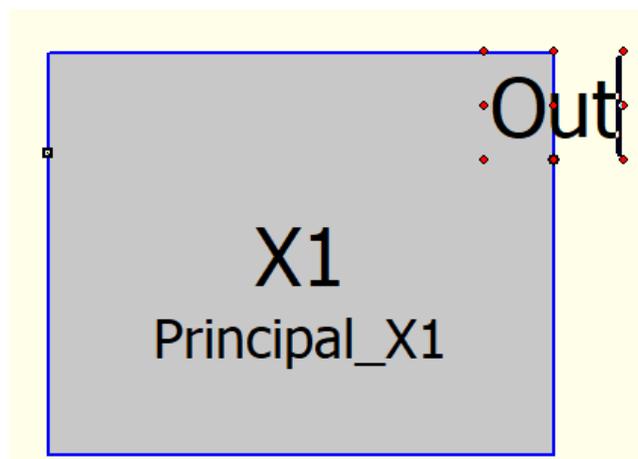
4. Adicione as portas que você deseja conectar do esquemático de nível superior para o nível inferior com o comando do menu do botão direito "Add Port".



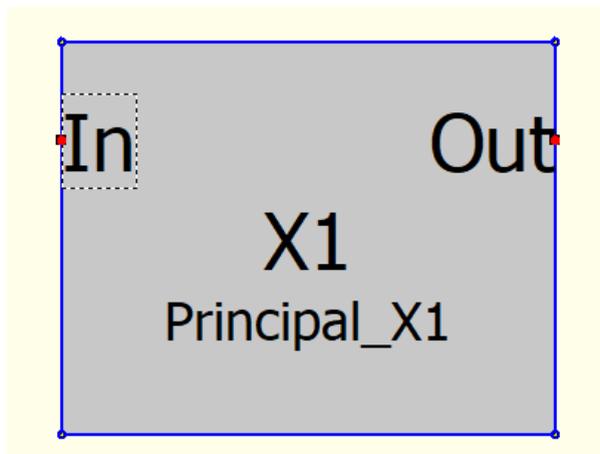
Neste exemplo são duas portas.



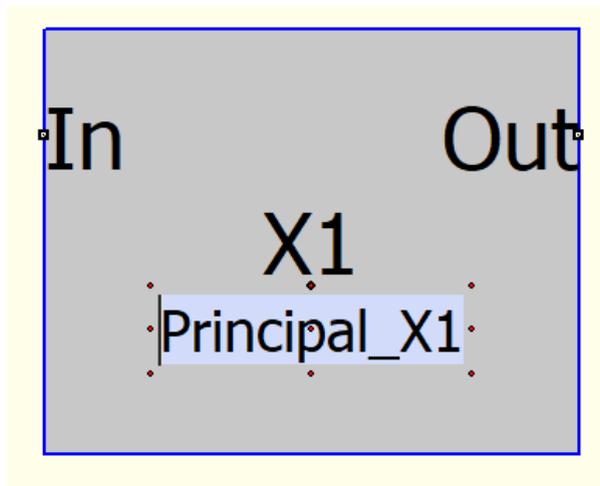
5. Edite os nomes e a justificação do texto das portas conforme sua preferência. Para ajustar o nome de um clique duplo sobre o ponto vermelho que representa a porta. Veja a imagem a seguir.



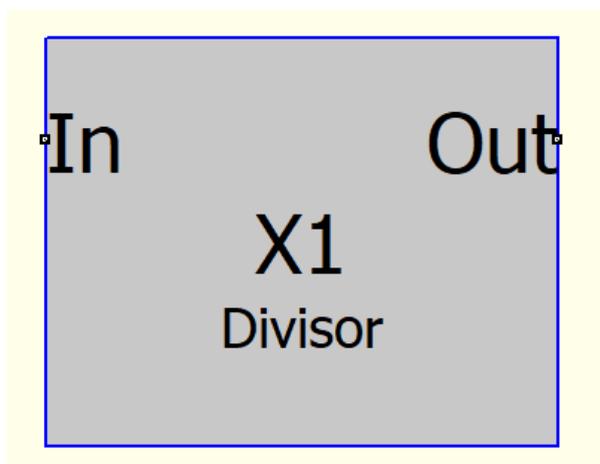
Para ajustar a posição do texto clique sobre os pontos vermelhos que aparecem ao redor do texto durante a edição. Veja a figura a seguir.



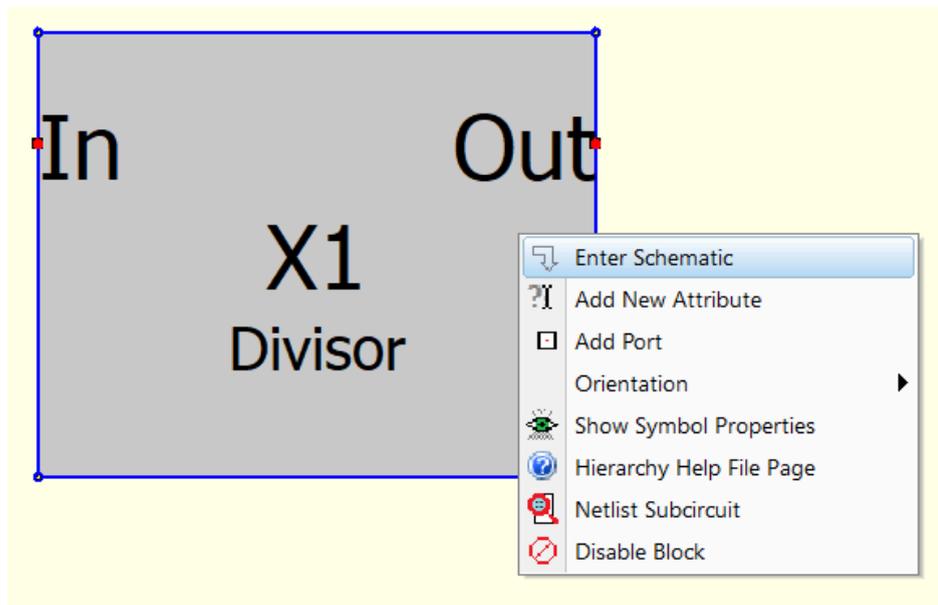
6. Com um duplo clique edite o nome do sub-circuito, como nas imagens a seguir.



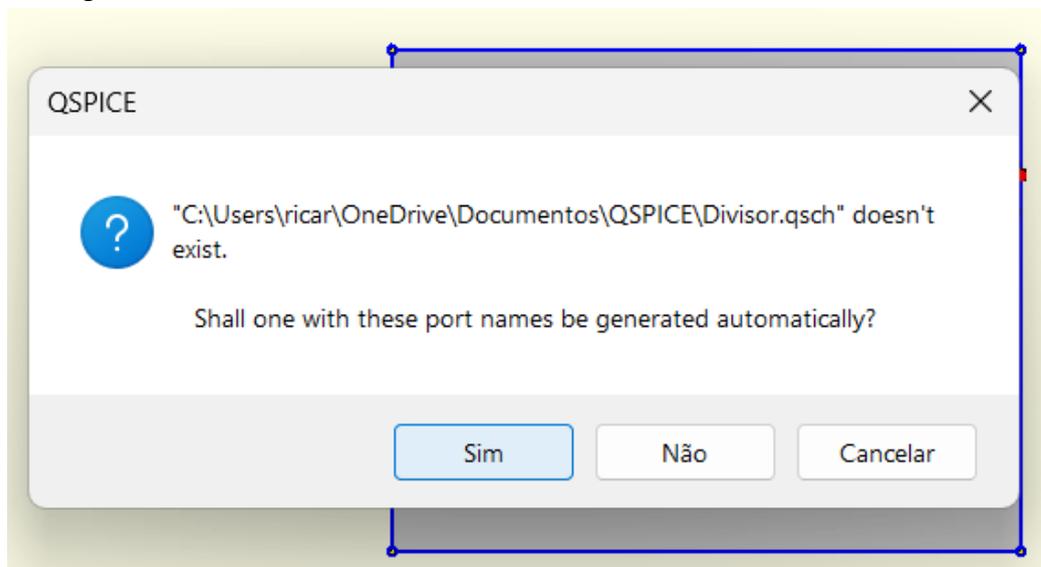
O nome que for inserido neste campo será o nome dado ao arquivo no momento de salvar o sub-circuito.



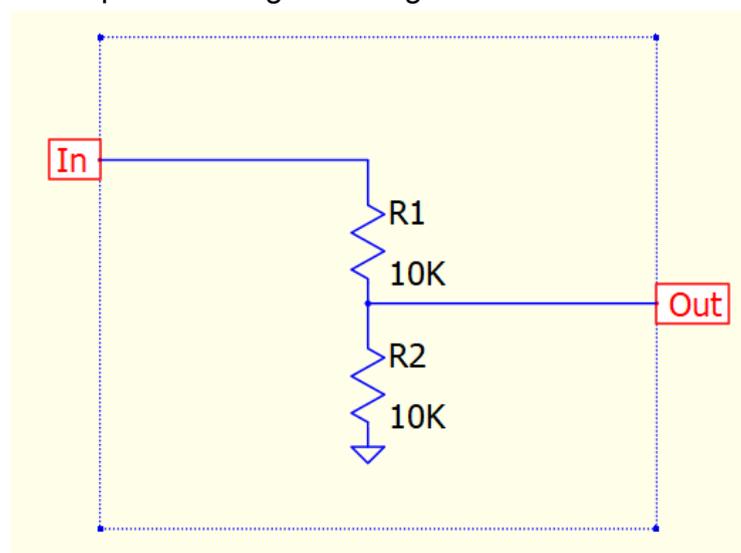
7. Clique com o botão direito no bloco e clique "Enter Schematic", como na imagem a seguir.

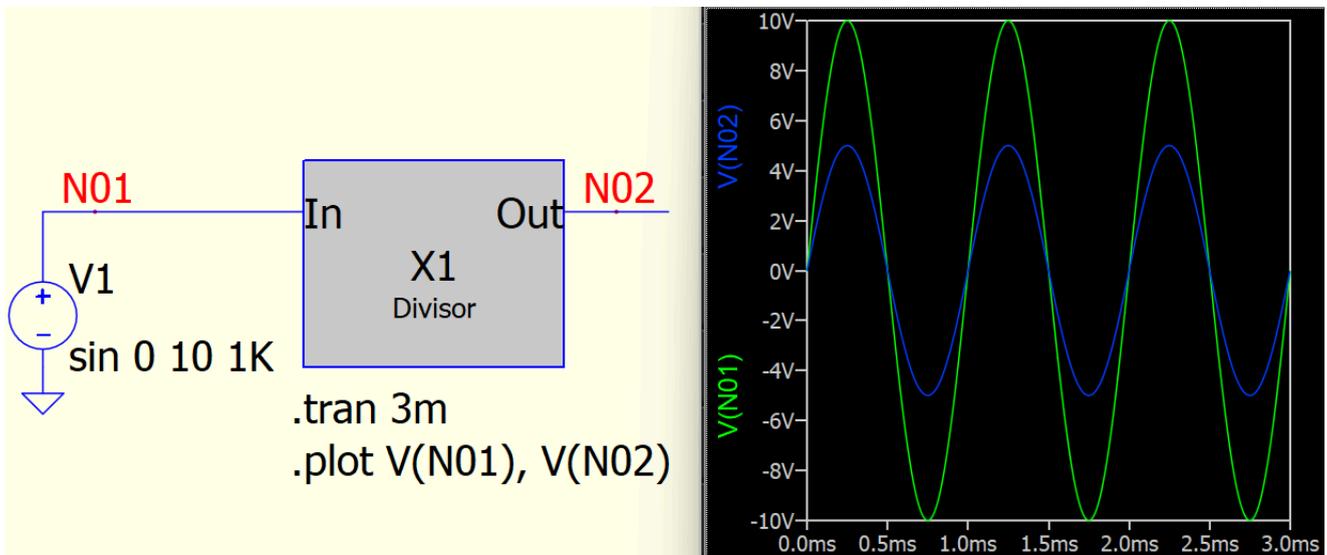


8. Selecione "Sim" para permitir que o Qspice crie um esquemático em branco com esses nomes de rede anotados como portas, este arquivo irá conter o sub-circuito. Veja a imagem a seguir.



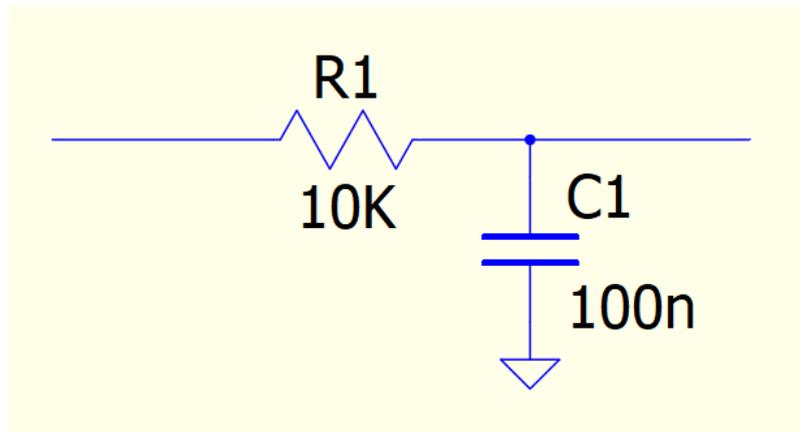
9. Agora, basta editar e salvar o sub-circuito como necessário, e então realizar a simulação desejada. Veja um exemplo nas imagens a seguir.



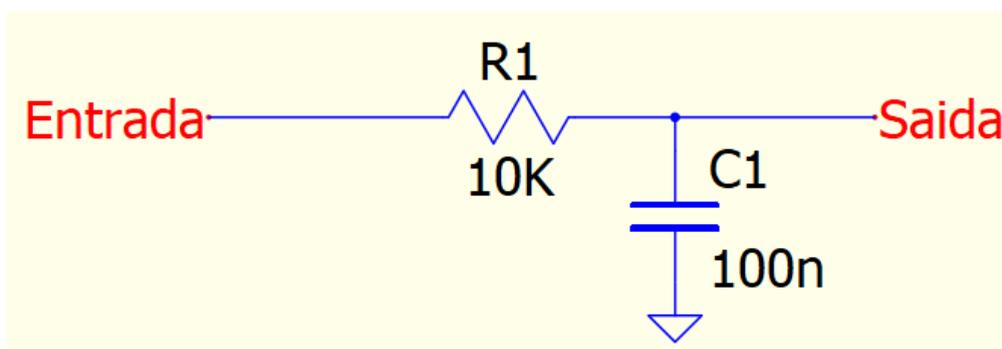


## Abordagem de Baixo para Cima (Bottom-Up)

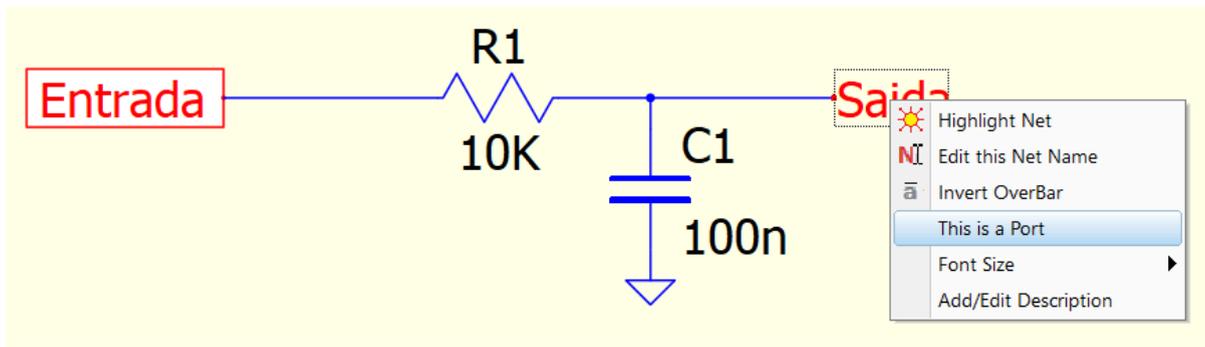
1. Salve seu esquemático de nível inferior em um diretório que você deseja usar para seu design hierárquico. Como por exemplo o circuito da imagem a seguir.



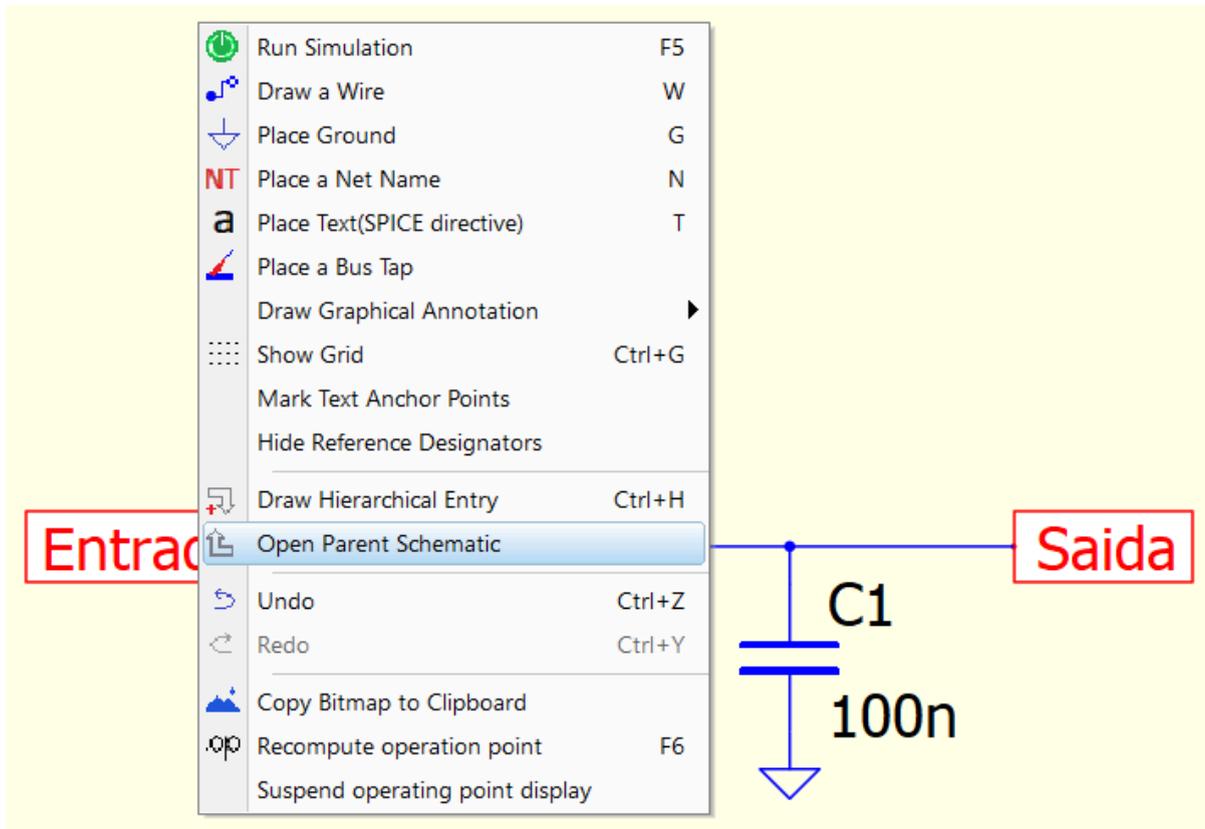
2. Rotule as redes que deseja usar como portas na camada de hierarquia, como na imagem a seguir.



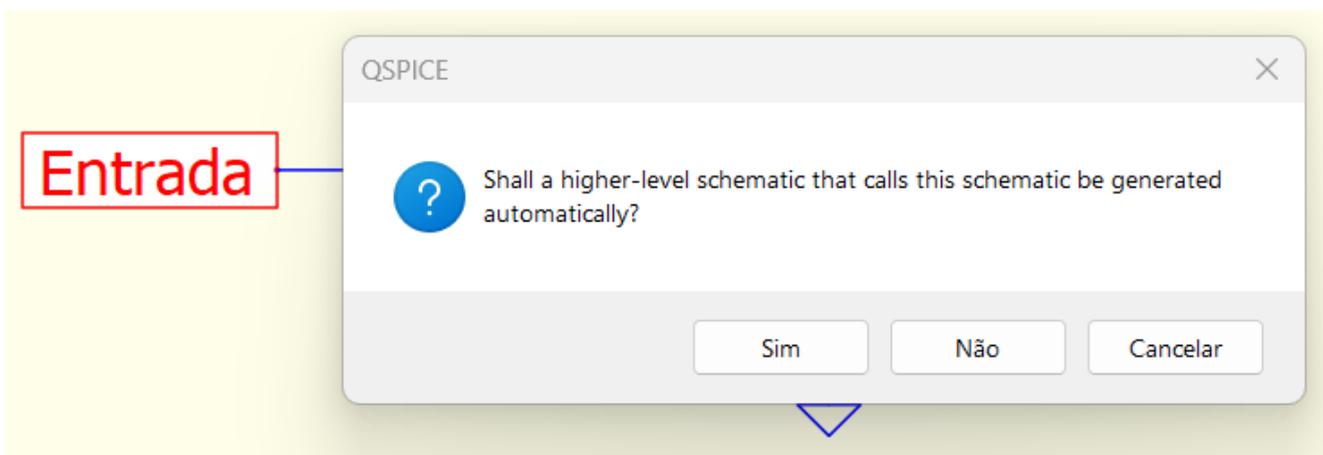
3. Clique com o botão direito em cada uma dessas redes e marque "This is a Port", como na imagem a seguir.



4. Clique com o botão direito em uma área vazia do circuito e clique em "Open Parent Schematic", como na imagem a seguir.



5. Selecione "Sim" quando perguntado se "Deve ser gerado automaticamente um esquemático de nível superior que chame este esquemático?", como na imagem a seguir.



6. Se necessário, você pode reajustar o tamanho da caixa e as localizações das portas. A imagem a seguir mostra um exemplo.

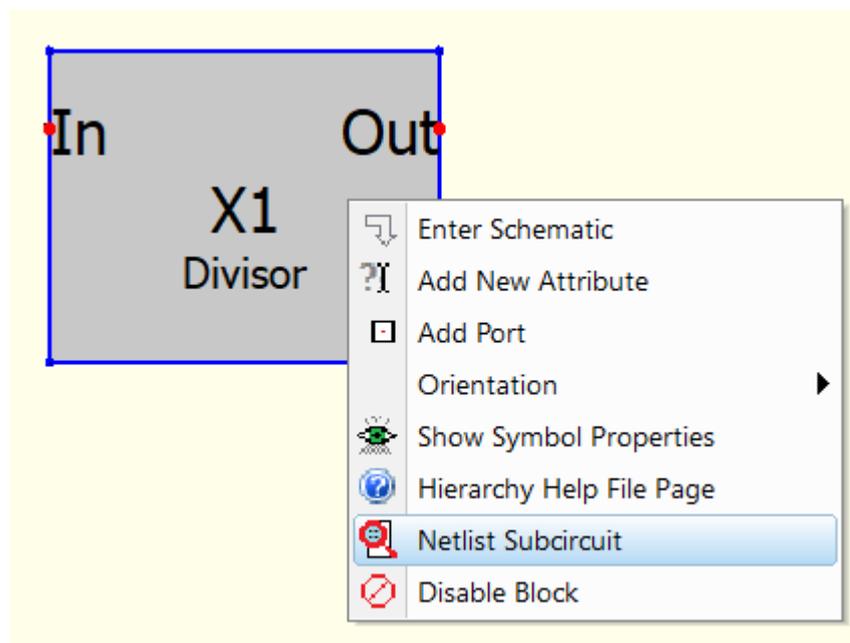


7. Agora, é só finalizar o circuito e realizar as simulações desejadas.

## Reaproveitando Sub-circuitos

Para reaproveitar um sub-circuito em outra simulação a forma mais fácil é copiar o arquivo do sub-circuito para a pasta onde a simulação atual está salva, e então copiar o bloco criado para o sub-circuito da simulação original onde o sub-circuito foi criado, e colar este bloco na simulação atual.

Outra forma de reaproveitar um sub-circuito é utilizar utilizar sua "Netlist" para criar um novo componente. A Netlist de um sub-circuito pode ser obtida clicando com o botão direito no bloco do sub-circuito e então clicando em "Netlist Subcircuit", como mostrado na imagem a seguir.



Como gerar um novo componente a partir de uma Netlist é o assunto do próximo capítulo.

## 7. Adição de novos componentes ao Qspice

Uma das principais limitações dos simuladores de circuitos em geral é a disponibilidade de componentes. Muitas vezes é necessário utilizar um componente genérico para simular nossos circuitos porque o componente que estamos utilizando não está disponível no simulador. Para contornar esta limitação a maioria dos simuladores permite adicionar, ou importar novos componentes. O Qspice não é diferente, ele permite que se importe componentes externos a partir de modelos SPICE genéricos.

## Como são modelados os componentes

No SPICE, os componentes eletrônicos são modelados matematicamente através de equações que descrevem seu comportamento físico. Cada componente, como resistores, capacitores, transistores e diodos, possui um modelo matemático específico que define como ele reage a variações de tensão e corrente no circuito. Além disso, modelos podem ser ajustados para diferentes condições operacionais, como temperatura e variações de processo de fabricação.

Esses modelos são expressos através de diretivas como `.model` ou `.subckt`, que permitem definir características detalhadas do componente ou criar sub-circuitos para dispositivos mais complexos.

Muitos fabricantes de componentes eletrônicos fornecem modelos SPICE de seus componentes, mas encontrá-los nem sempre é fácil. A seguir temos uma lista com alguns locais onde são disponibilizados modelos SPICE. A formatação varia com o fabricante e o simulador utilizado, assim alguns modelos precisam ser adaptados. Alguns fornecedores disponibilizam bibliotecas de componentes com a extensão `.lib`, que contém vários componentes, e também podem ser utilizadas.

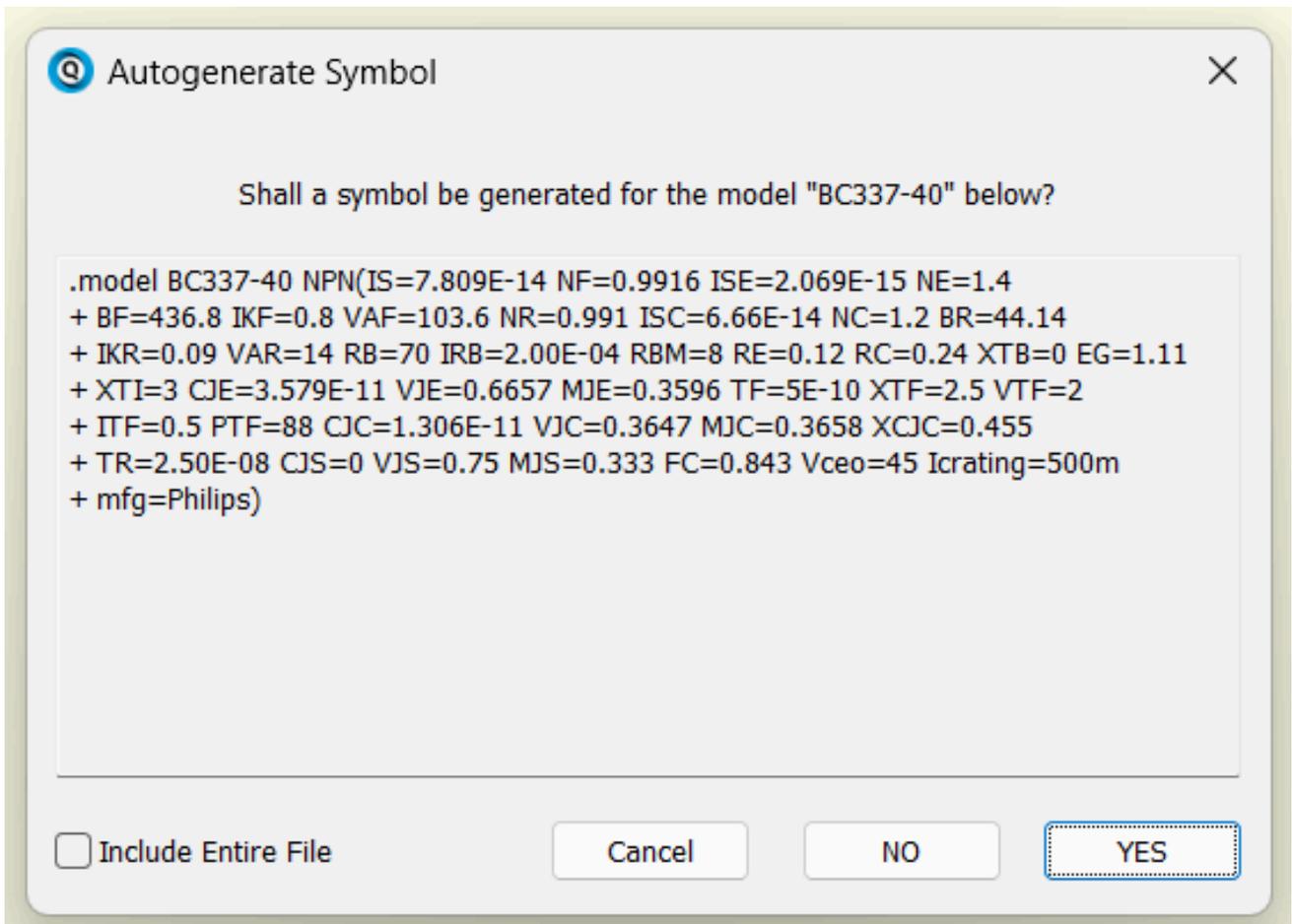
Um exemplo de site que disponibiliza modelos SPICE é: <https://github.com/kicad-spice-library/KiCad-Spice-Library>

## Importando componentes simples

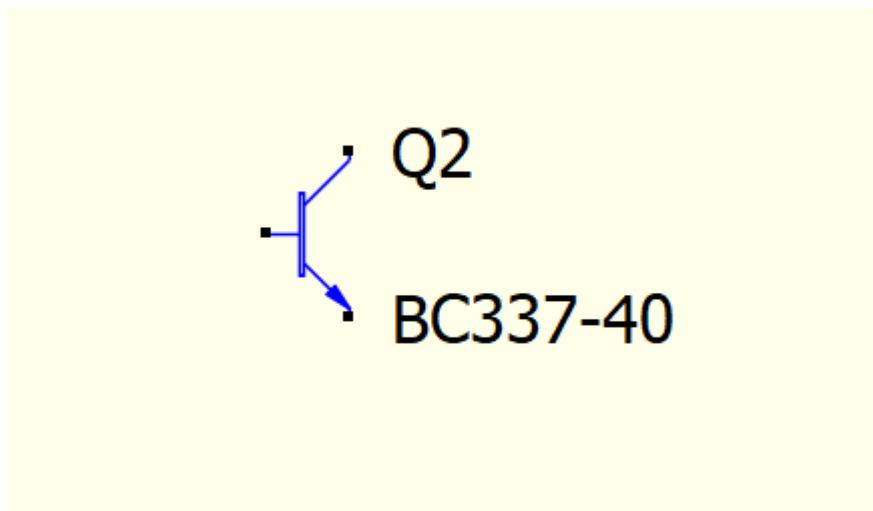
Para componentes mais simples como transistores e diodos o Qspice já possui símbolos nativos, assim basta fornecer o modelo SPICE (`.model`). Como exemplo vamos utilizar o transistor BC337, cujo modelo é apresentado a seguir.

```
.model BC337-40 NPN(IS=7.809E-14 NF=0.9916 ISE=2.069E-15 NE=1.4 BF=436.8
IKF=0.8 VAF=103.6 NR=0.991 ISC=6.66E-14 NC=1.2 BR=44.14 IKR=0.09 VAR=14
RB=70 IRB=2.00E-04 RBM=8 RE=0.12 RC=0.24 XTB=0 EG=1.11 XTI=3 CJE=3.579E-11
VJE=0.6657 MJE=0.3596 TF=5E-10 XTF=2.5 VTF=2 ITF=0.5 PTF=88 CJC=1.306E-11
VJC=0.3647 MJC=0.3658 XCJC=0.455 TR=2.50E-08 CJS=0 VJS=0.75 MJS=0.333
FC=0.843 Vceo=45 Icrating=500m mfg=Philips)
```

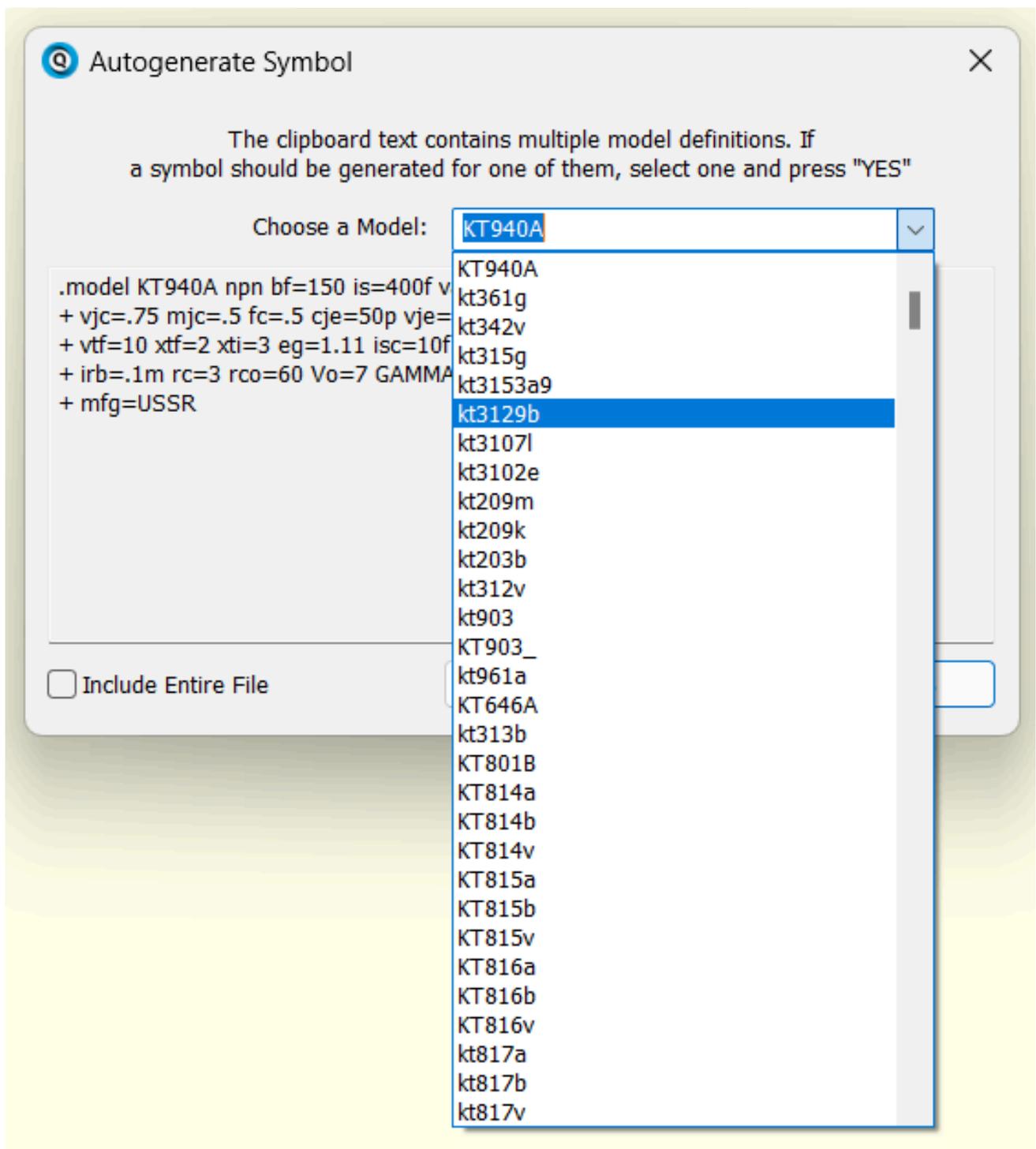
O Qspice foi desenvolvido para facilitar a importação de componentes, assim para importar o BC337 basta copiar o texto do modelo e colar na janela principal do simulador. A janela da imagem a seguir é apresentada.



Basta clicar "YES" e o Qspice vai associar um símbolo ao componente e adiciona-lo ao circuito, como na imagem a seguir.



Também é possível colar diretamente o arquivo de texto que contém o modelo do componente. Caso seja adicionado um arquivo que contém mais de um componente, a janela apresentada pelo Qspice é diferente, agora é possível escolher qual componente do arquivo queremos importar. Veja um exemplo na imagem a seguir.



## Importando componentes complexos

Também é possível importar modelos de componentes mais complexos, como circuitos integrados, por exemplo, que são compostos por vários componentes. Os fabricantes dos circuitos em muitos casos, disponibilizam modelos SPICE de seus componentes. O processo de importação do modelo de um componente mais complexo no Qspice é feito através da importação da "Netlist" completa do componente.

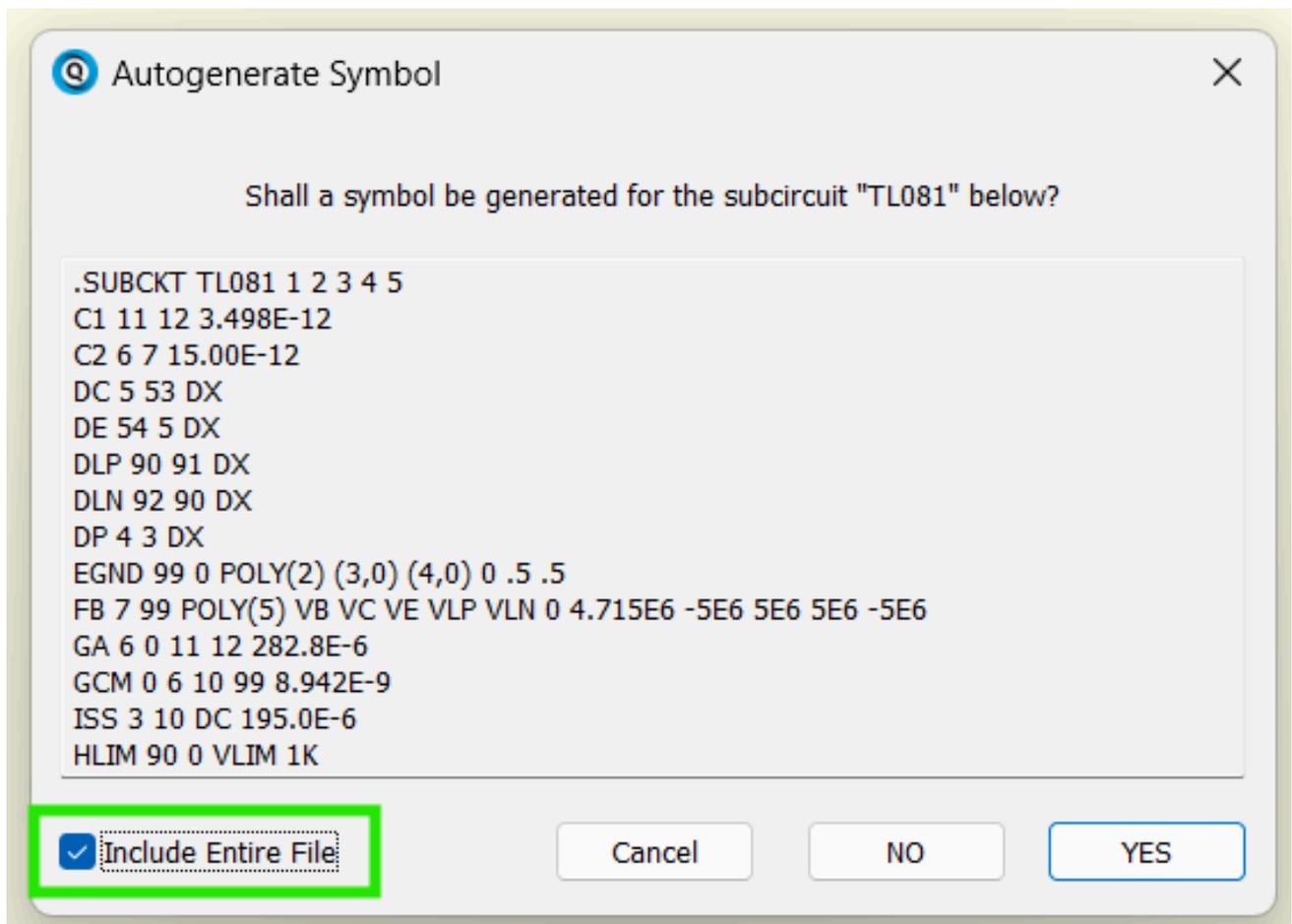
O SPICE utiliza a diretiva `.subckt` para isso. O processo é semelhante, basta colar o texto ou o arquivo com a descrição do componente na janela principal do Qspice. Como exemplo vamos utilizar o amplificador operacional TL081, cujo modelo é apresentado a seguir.

```

* TL081 OPERATIONAL AMPLIFIER "MACROMODEL" SUBCIRCUIT
* CREATED USING PARTS RELEASE 4.01 ON 06/16/89 AT 13:08
* (REV N/A) SUPPLY VOLTAGE: +/-15V
* CONNECTIONS: NON-INVERTING INPUT
* | INVERTING INPUT
* | | POSITIVE POWER SUPPLY
* | | | NEGATIVE POWER SUPPLY
* | | | | OUTPUT
* | | | | |
.SUBCKT TL081 1 2 3 4 5
*
C1 11 12 3.498E-12
C2 6 7 15.00E-12
DC 5 53 DX
DE 54 5 DX
DLP 90 91 DX
DLN 92 90 DX
DP 4 3 DX
EGND 99 0 POLY(2) (3,0) (4,0) 0 .5 .5
FB 7 99 POLY(5) VB VC VE VLP VLN 0 4.715E6 -5E6 5E6 5E6 -5E6
GA 6 0 11 12 282.8E-6
GCM 0 6 10 99 8.942E-9
ISS 3 10 DC 195.0E-6
HLIM 90 0 VLIM 1K
J1 11 2 10 JX
J2 12 1 10 JX
R2 6 9 100.0E3
RD1 4 11 3.536E3
RD2 4 12 3.536E3
R01 8 5 150
R02 7 99 150
RP 3 4 2.143E3
RSS 10 99 1.026E6
VB 9 0 DC 0
VC 3 53 DC 2.200
VE 54 4 DC 2.200
VLIM 7 8 DC 0
VLP 91 0 DC 25
VLN 0 92 DC 25
.MODEL DX D(IS=800.0E-18)
.MODEL JX PJF(IS=15.00E-12 BETA=270.1E-6 VTO=-1)
.ENDS

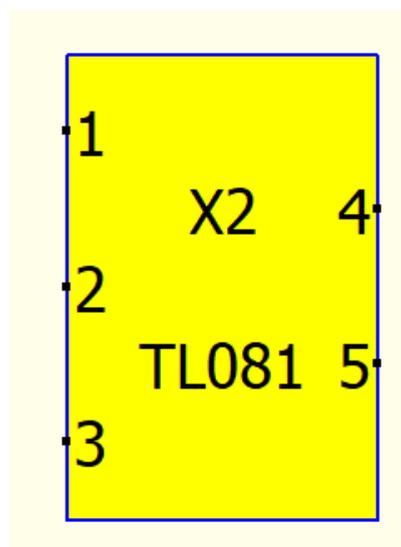
```

Quando colamos este modelo no Qspice a janela da imagem a seguir é apresentada.



Para modelos mais complexos, que podem conter sub-modelos é importante selecionar a opção "Include Entire File", como destacado em verde na figura anterior.

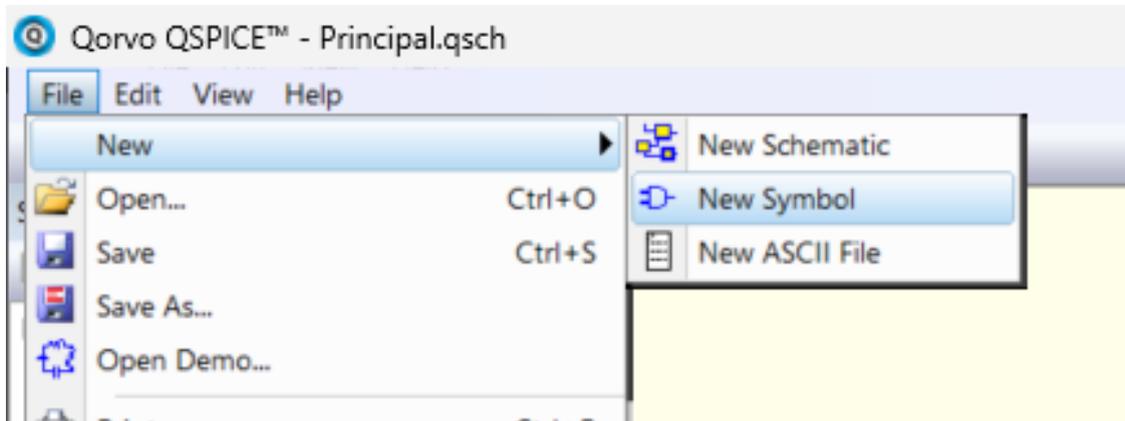
A principal diferença entre importar componentes simples e componentes complexos é que para componentes complexos o Qspice não consegue criar um símbolo adequado. A imagem a seguir mostra o símbolo criado para o TL081.



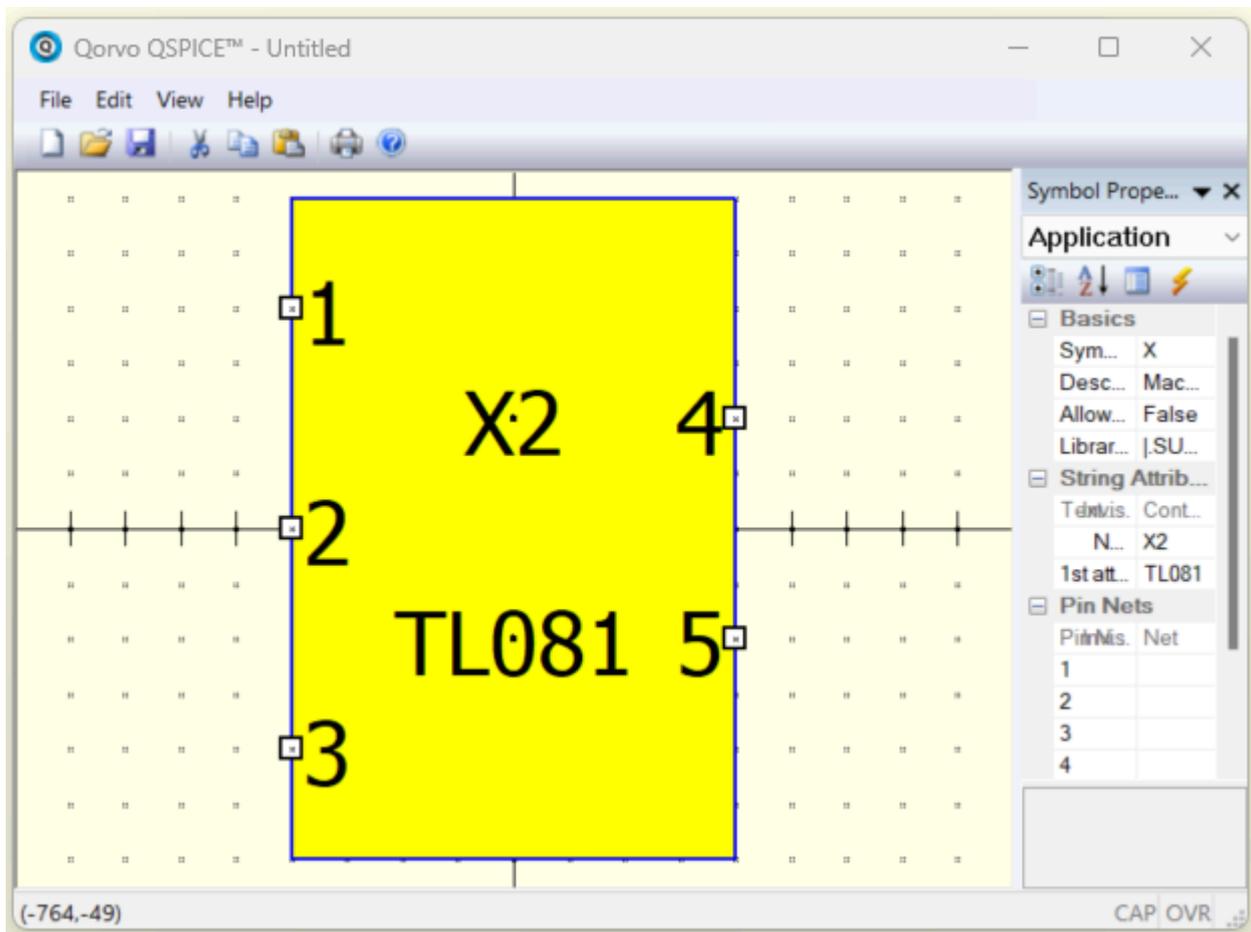
Apesar de ser possível utilizar o componente desta forma, é muito mais apropriado criar um símbolo para ele.

## Criando um símbolo para um componente

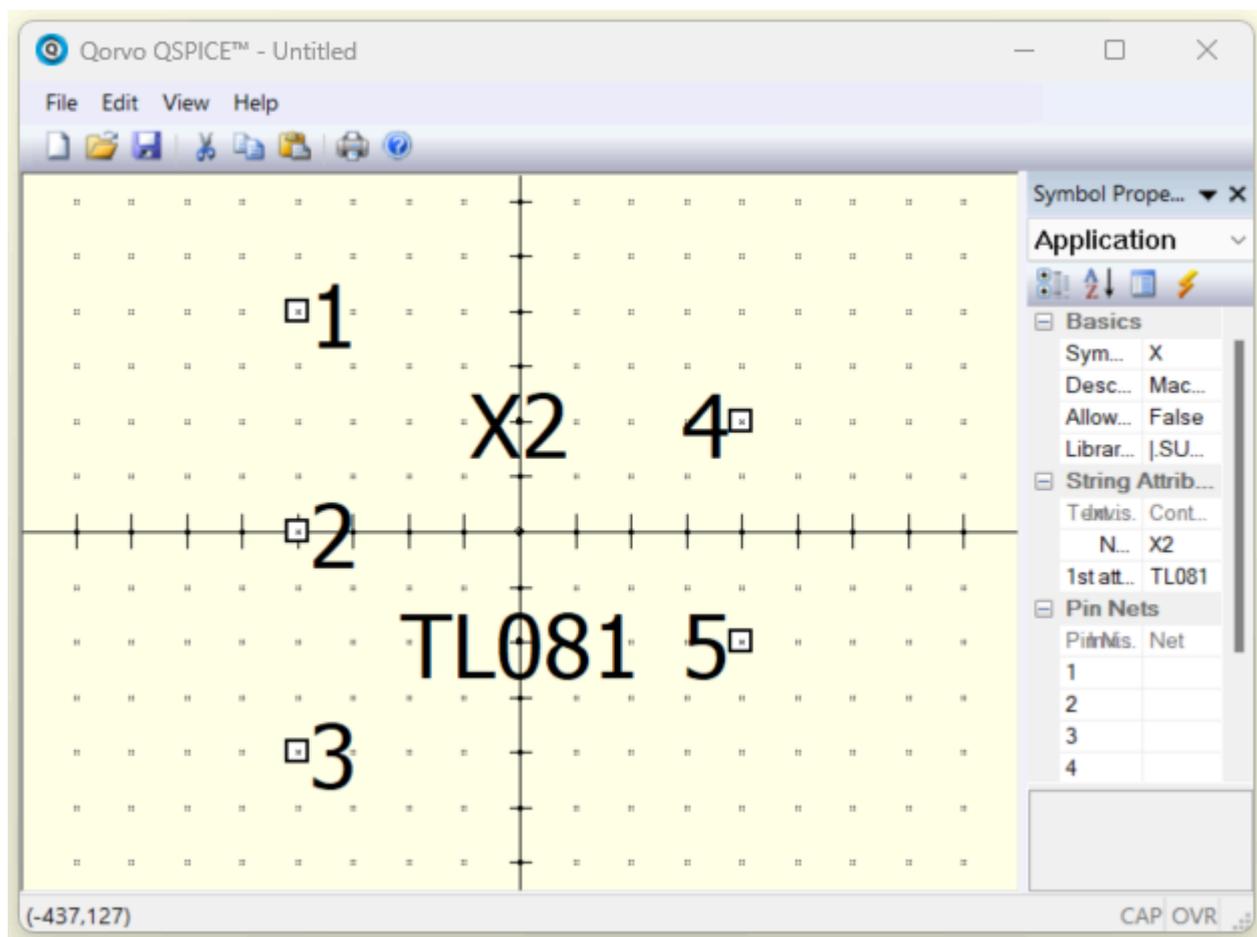
O Qspice possui um editor de símbolos, para criar um novo símbolo basta clicar em "File", "New" e "New Symbol", como na imagem a seguir.



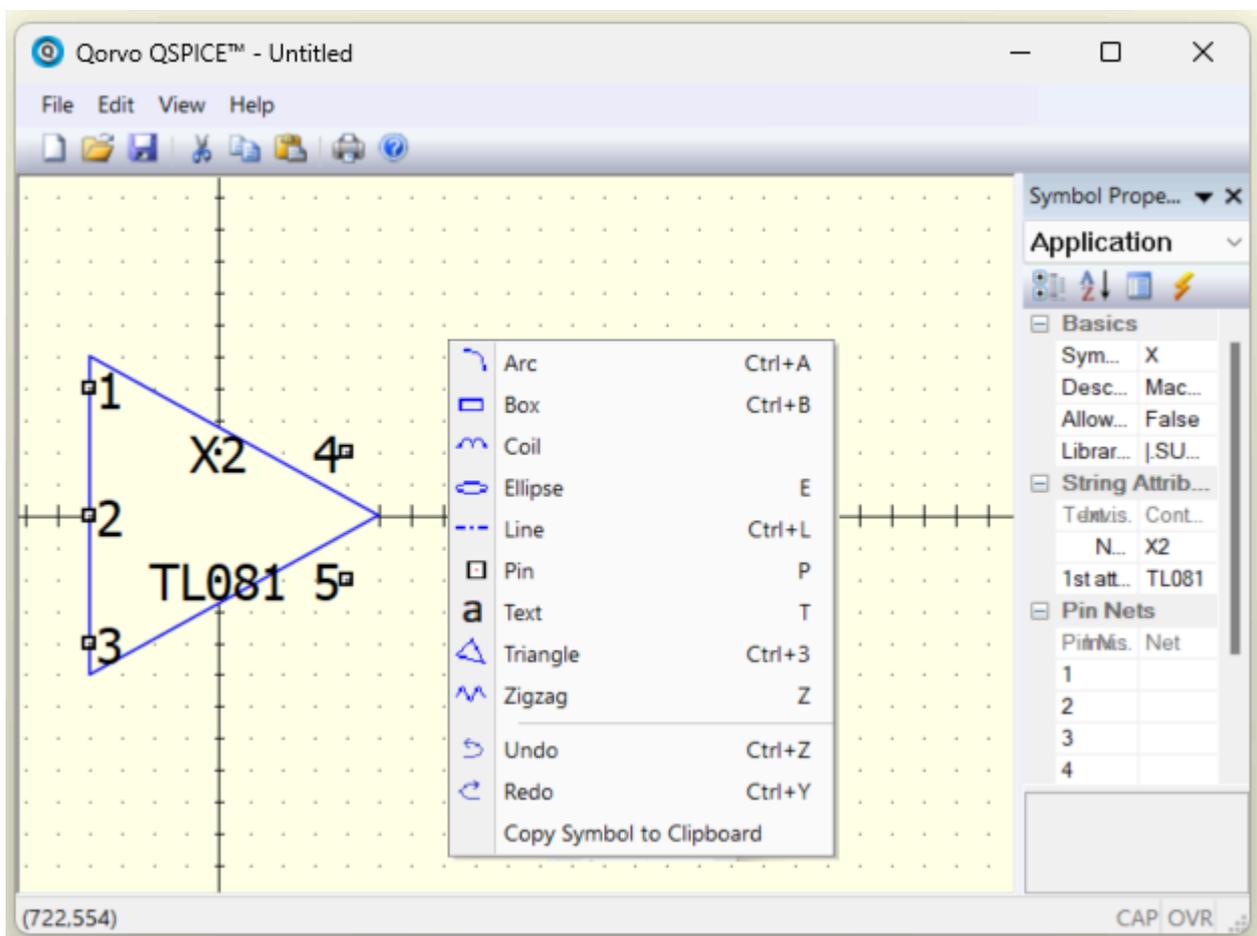
Para editar o símbolo copie e cole o componente que se deseja editar no centro do editor de símbolos, como na imagem a seguir.



Selecione e delete o retângulo amarelo, como na imagem a seguir.



Agora, utilizando as ferramentas de desenho disponíveis no menu do botão direito do mouse desenhe um triângulo, como na imagem a seguir.



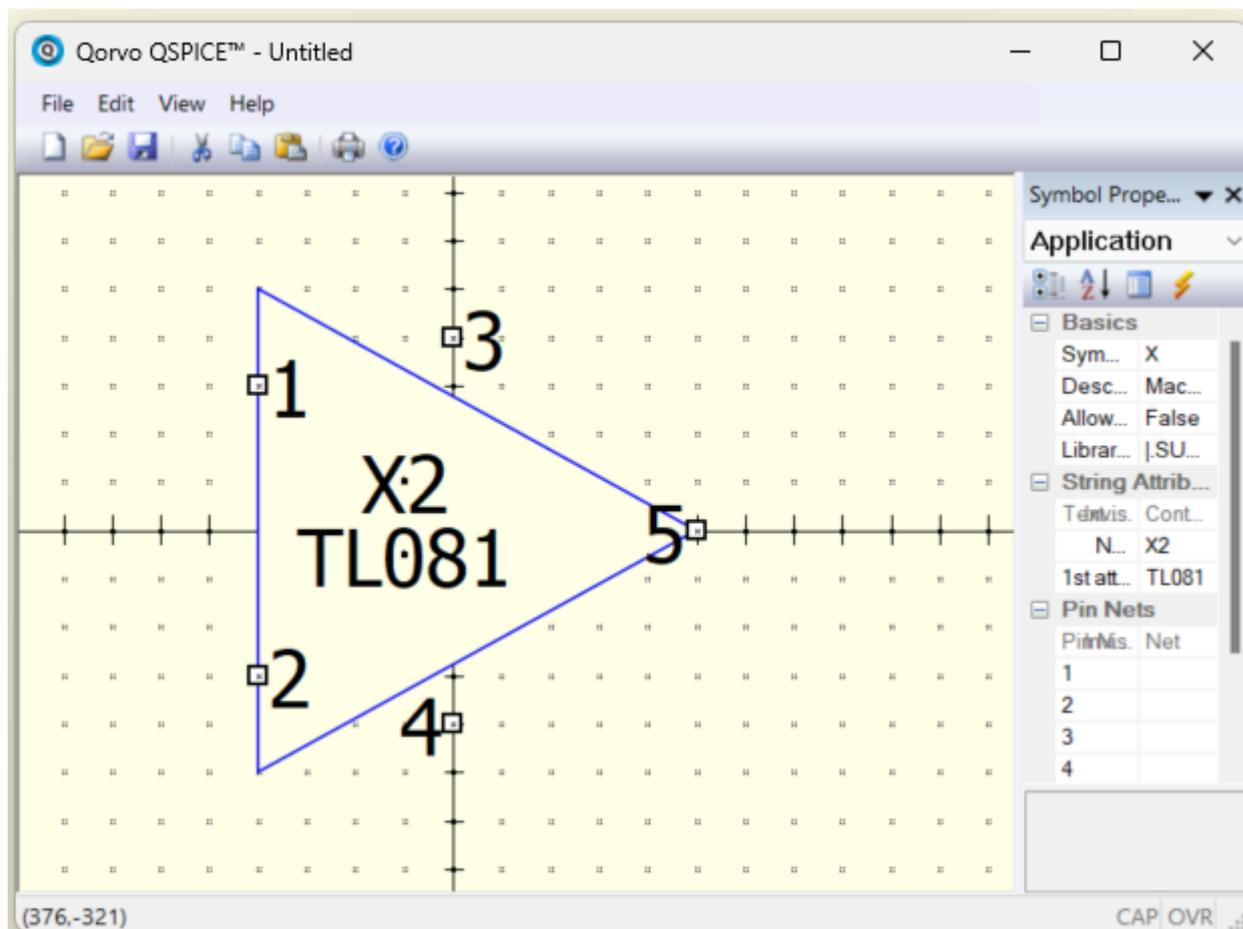
Agora, é necessário posicionar os pontos de conexão. **Importante, não altere o nome dos pontos de conexão, ou o modelo não vai funcionar.** Para saber qual a função de cada ponto de conexão é necessário observar os comentários no texto do modelo, como a seguir.

```
* CONNECTIONS:  NON-INVERTING INPUT
*                |  INVERTING INPUT
*                | |  POSITIVE POWER SUPPLY
*                | | |  NEGATIVE POWER SUPPLY
*                | | | |  OUTPUT
*                | | | | |
.SUBCKT TL081    1 2 3 4 5
```

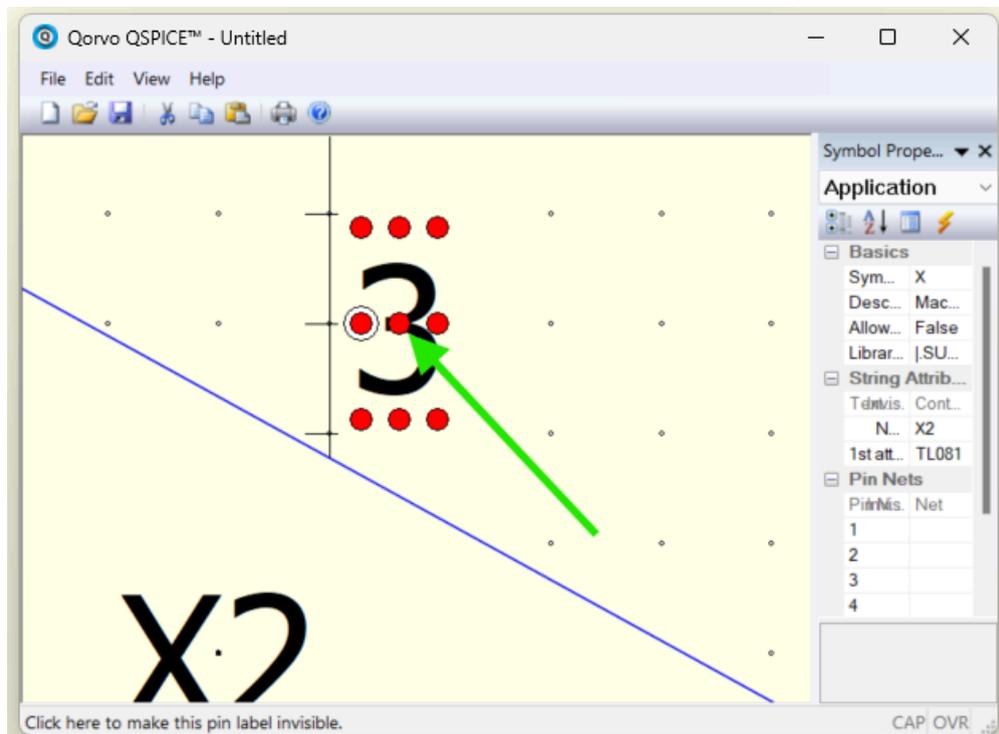
Para nosso TL081 os pinos são os seguintes:

1. Entrada não inversora (+).
2. Entrada inversora (-).
3. Alimentação positiva (V+).
4. Alimentação Negativa (V-).
5. Saída.

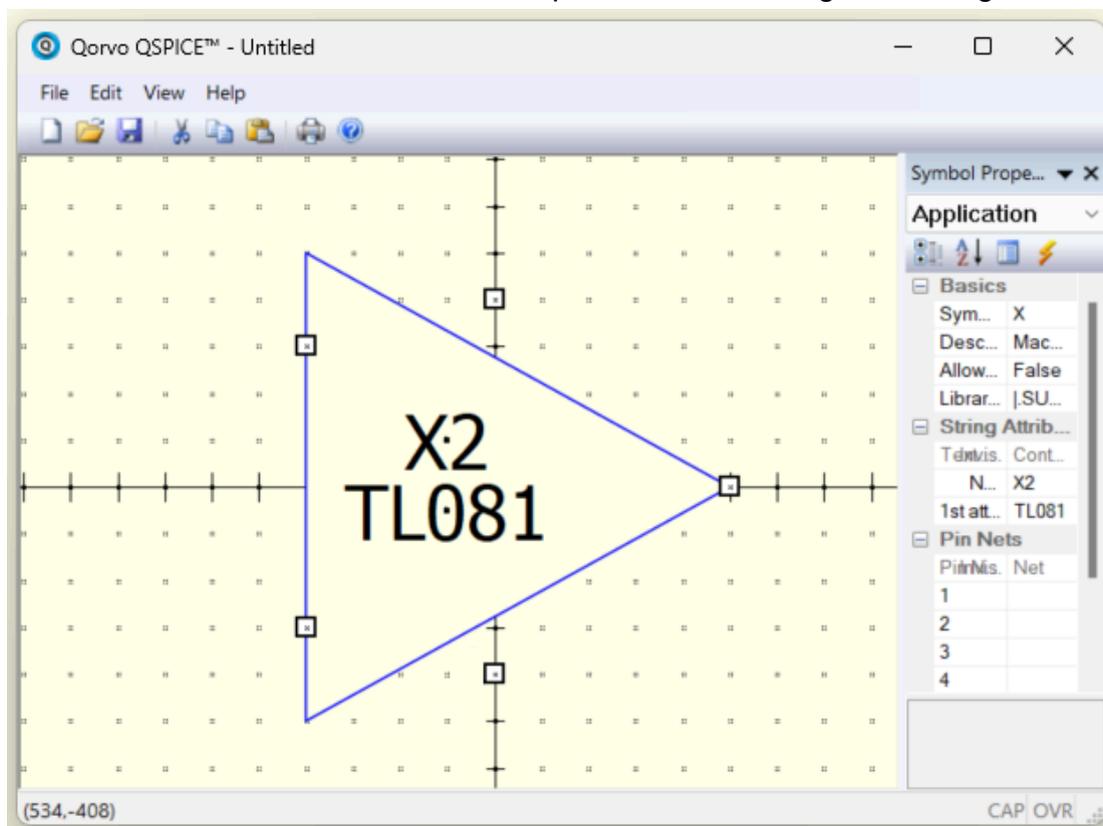
Com estas informações é possível posicionar os pontos de conexão adequadamente. Também é possível posicionar o nome do componente e outros textos, como na imagem a seguir.



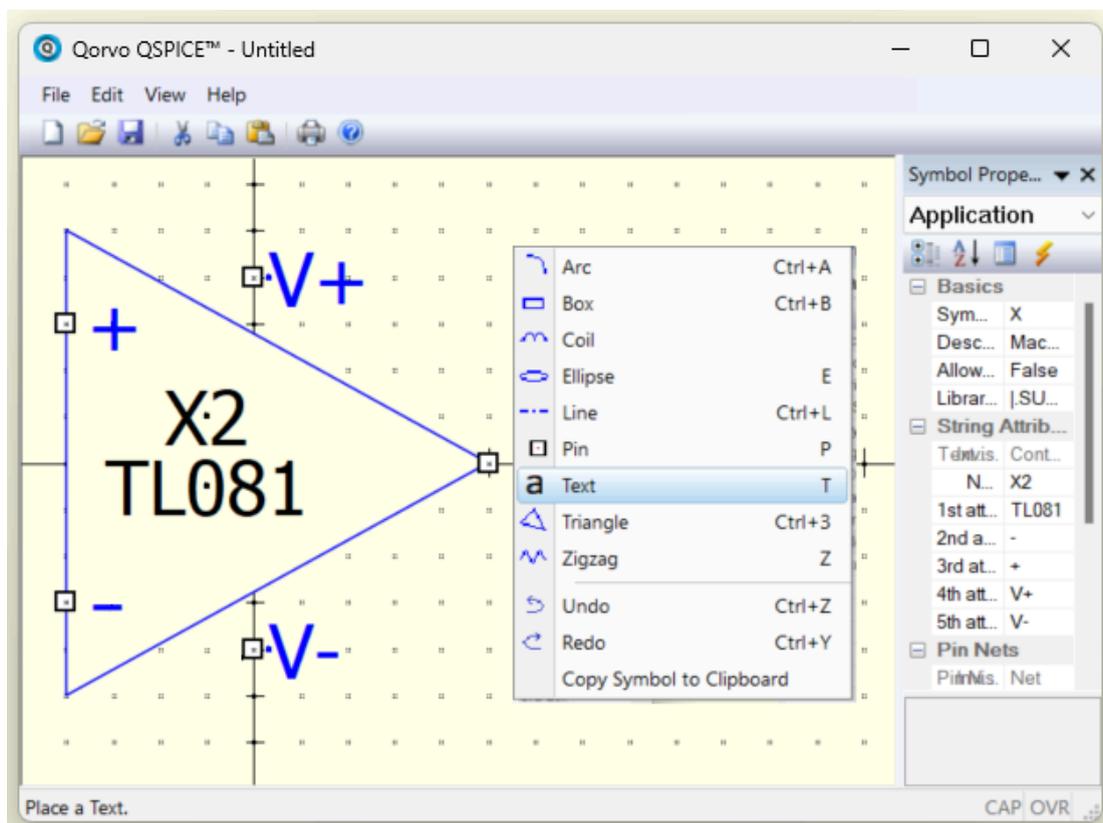
Para facilitar a utilização do componente é necessário nomear os pontos de conexão adequadamente. Como não é permitido mudar o nome, o que é recomendado pelo criador do Qspice é que se oculte o nome dos pontos de conexão e depois se adicione textos próximos a estes pontos. Para ocultar o nome basta clicar para editar o texto e então clicar no ponto vermelho que aparece no centro do texto, veja a flecha verde na imagem a seguir.



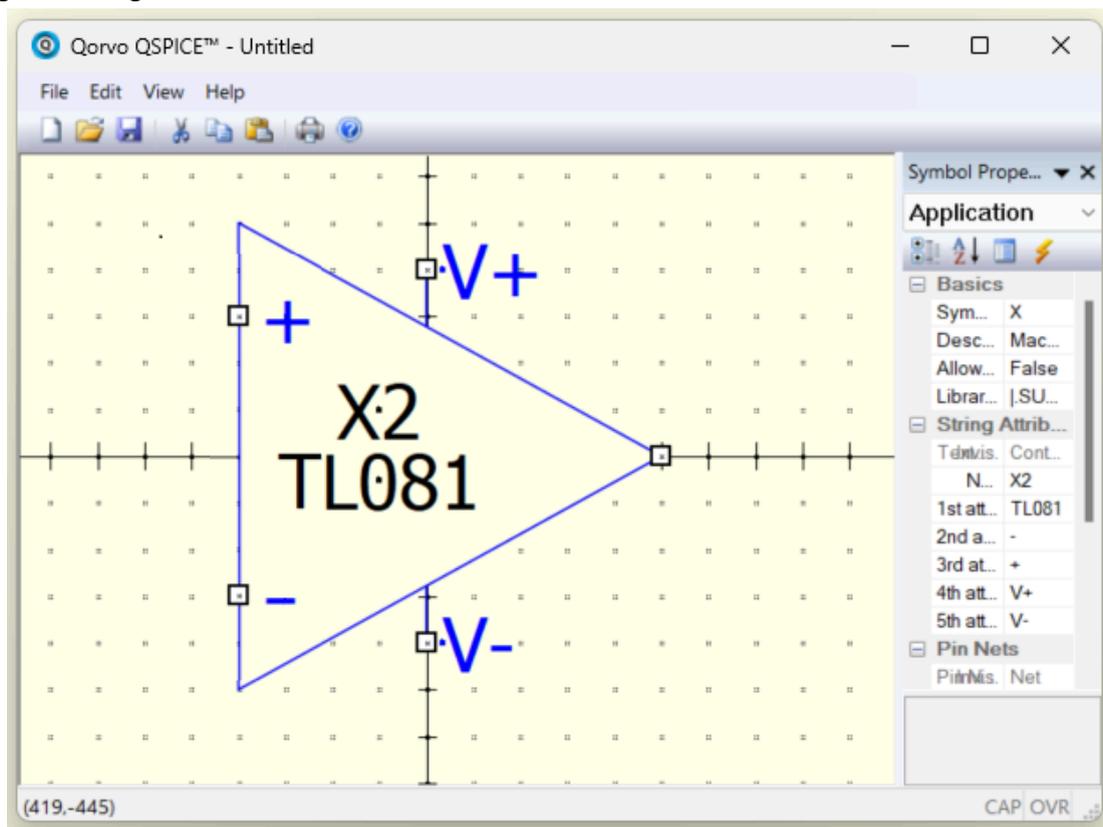
Ocultando todos os nomes temos o símbolo apresentado na imagem a a seguir.



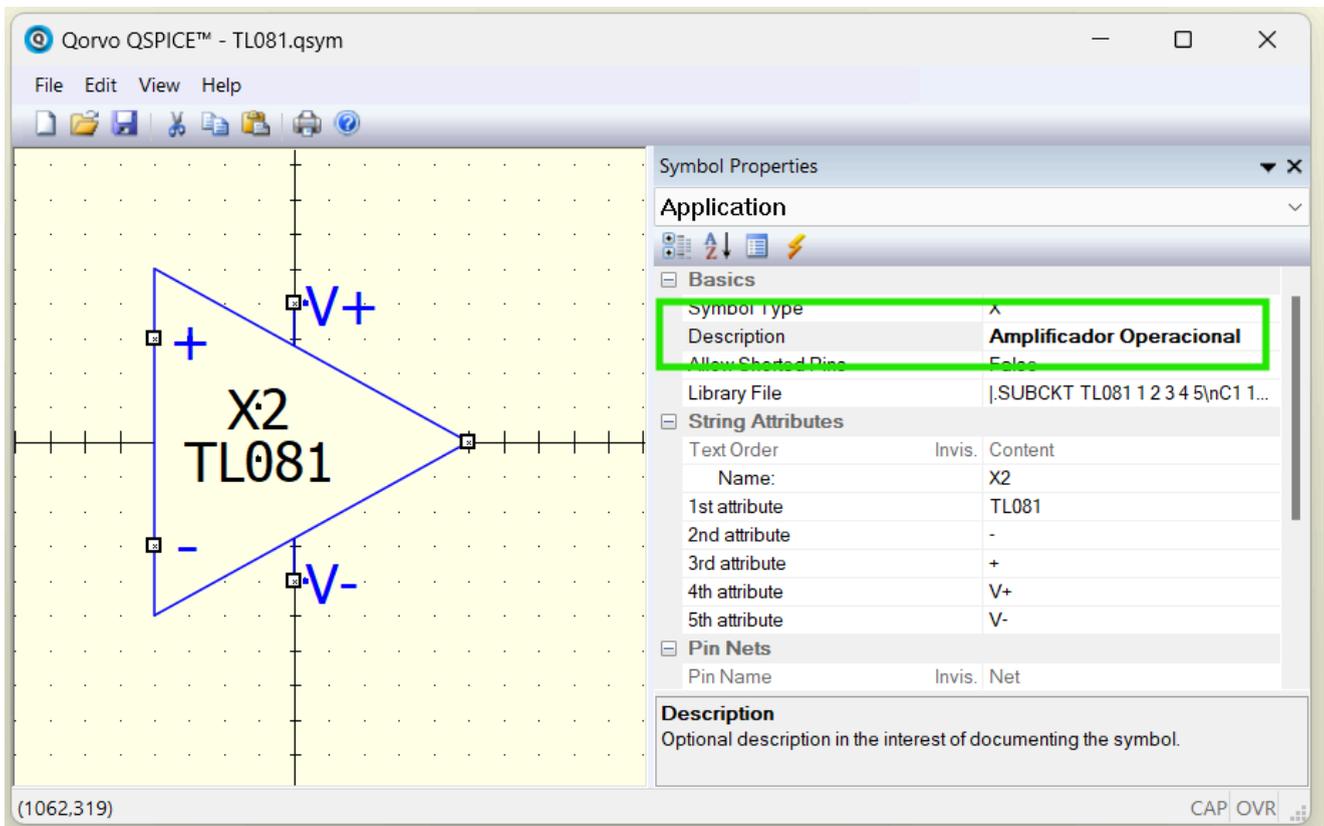
Agora, para adicionar textos é necessário clicar com o botão direito e selecionar "Text". É importante transformar todos os textos em comentários (texto azul). Veja a imagem a seguir.



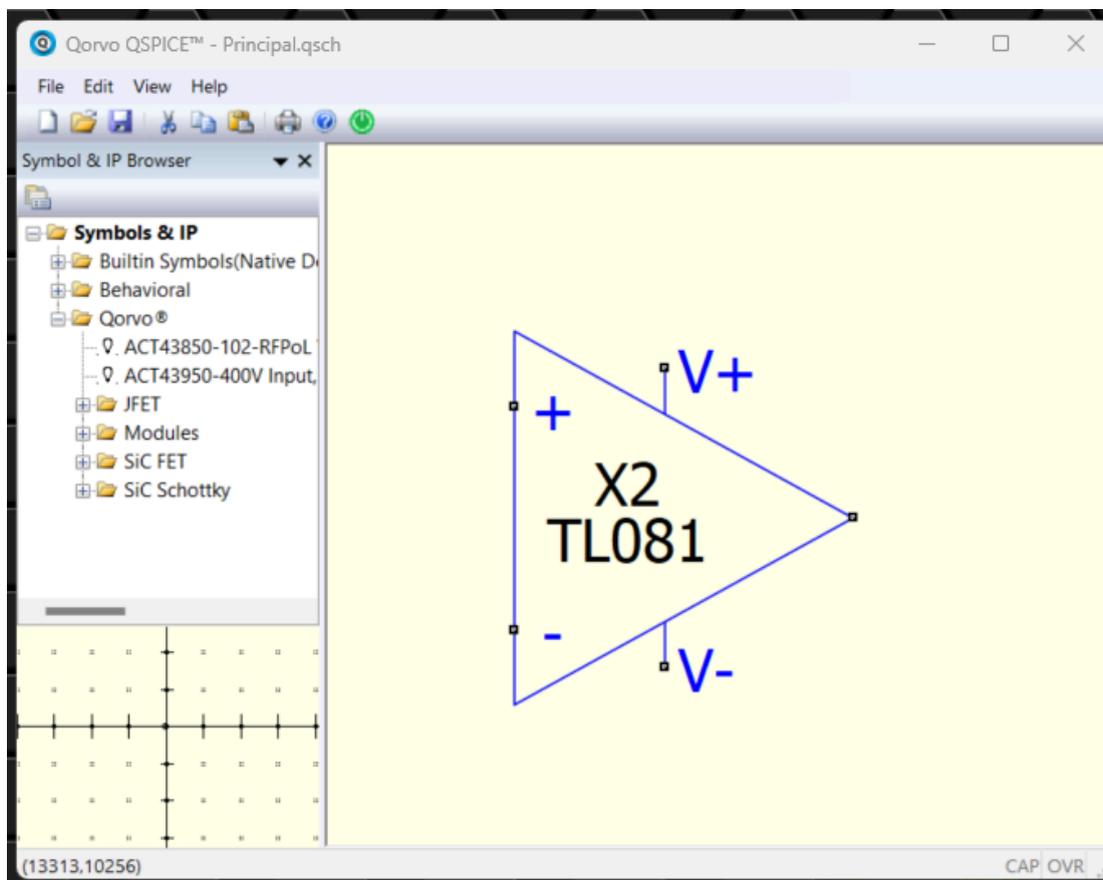
Após ajustar os textos e fazer as alterações necessárias ao símbolo o resultado fica como na imagem a seguir.



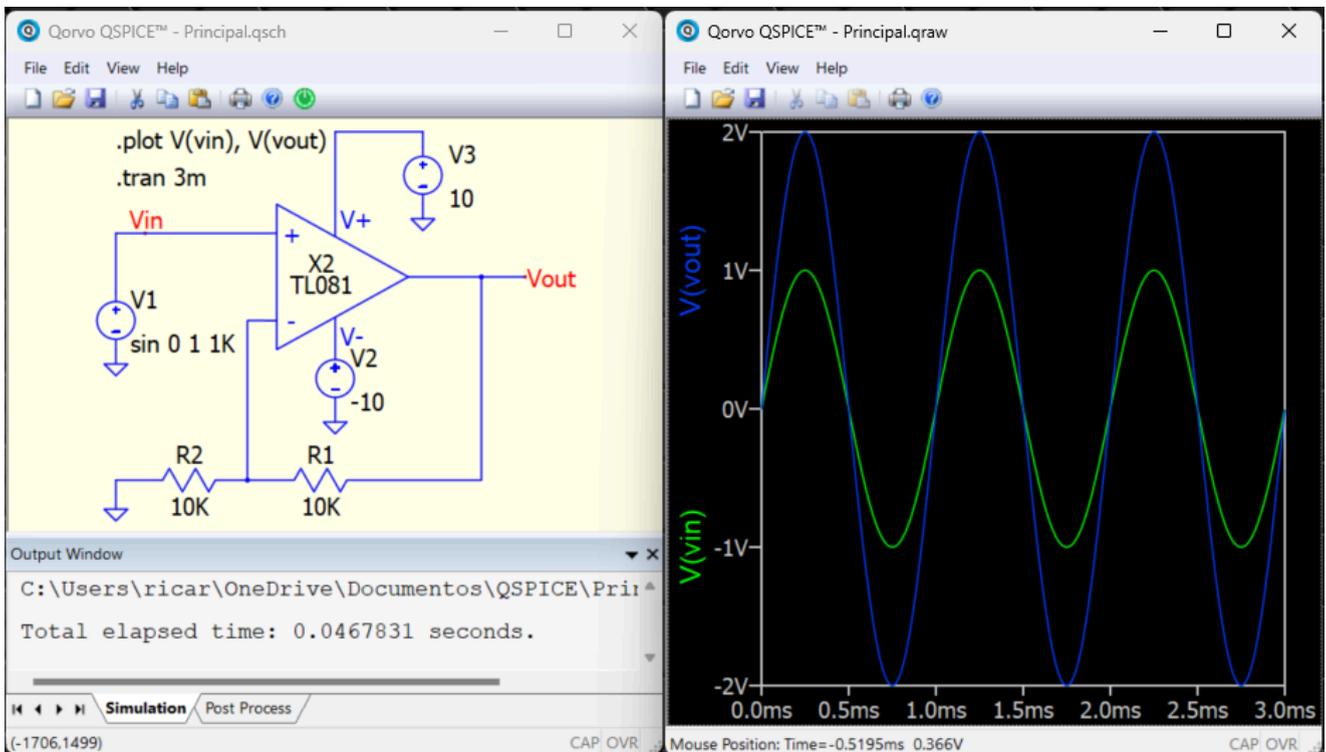
O último passo é ajustar a descrição do componente. Isso é feito editando o texto no "Symbol Properties", como destacado em verde na imagem a seguir.



Agora, para utilizar o símbolo em nossa simulação deve-se dar um clique em uma área vazia do editor de símbolos e clicar "Ctrl - C" para copiar o novo símbolo. Então é só colá-lo no simulador, como na imagem a seguir.



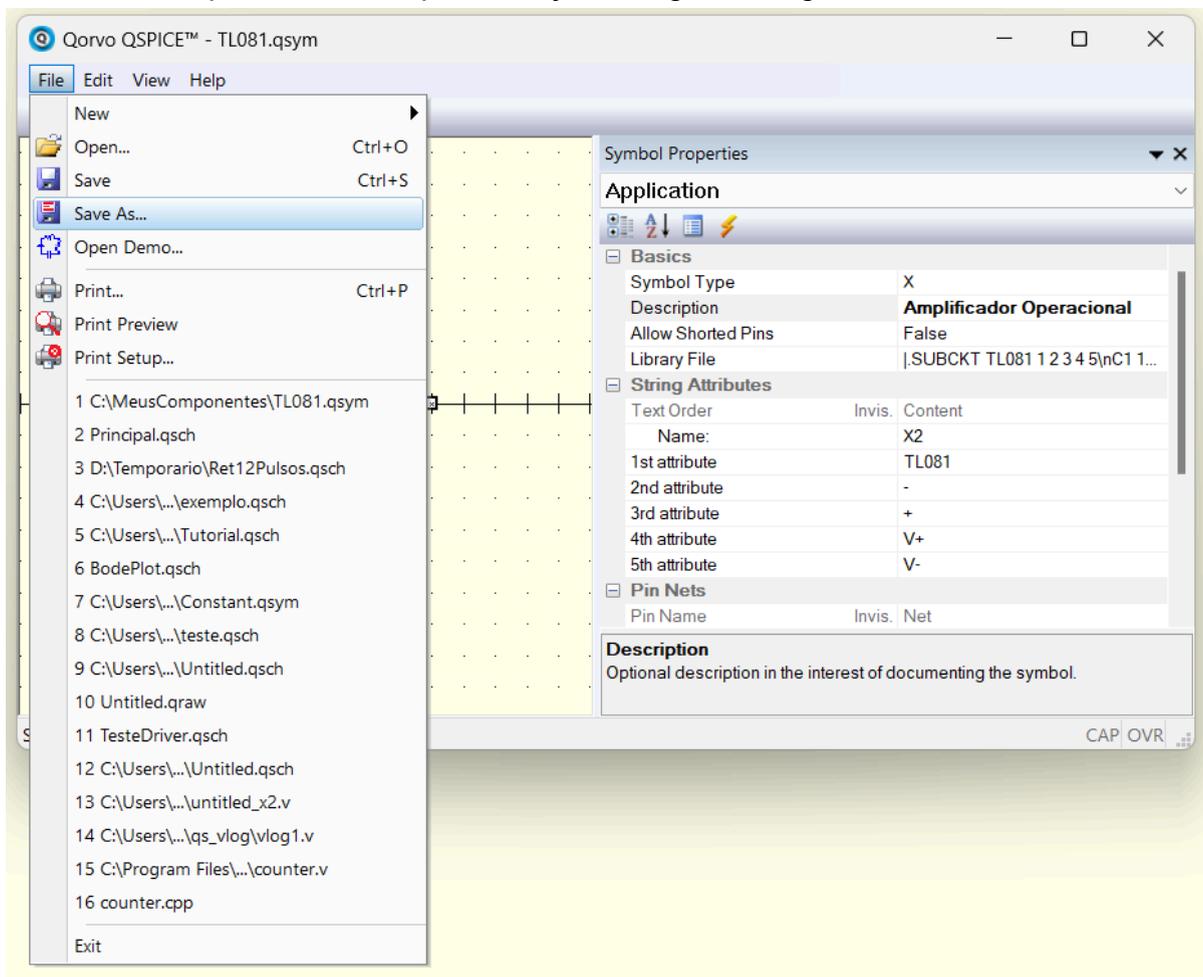
A imagem a seguir mostra o TL081 recém criado sendo utilizado na simulação de um amplificador não inversor.



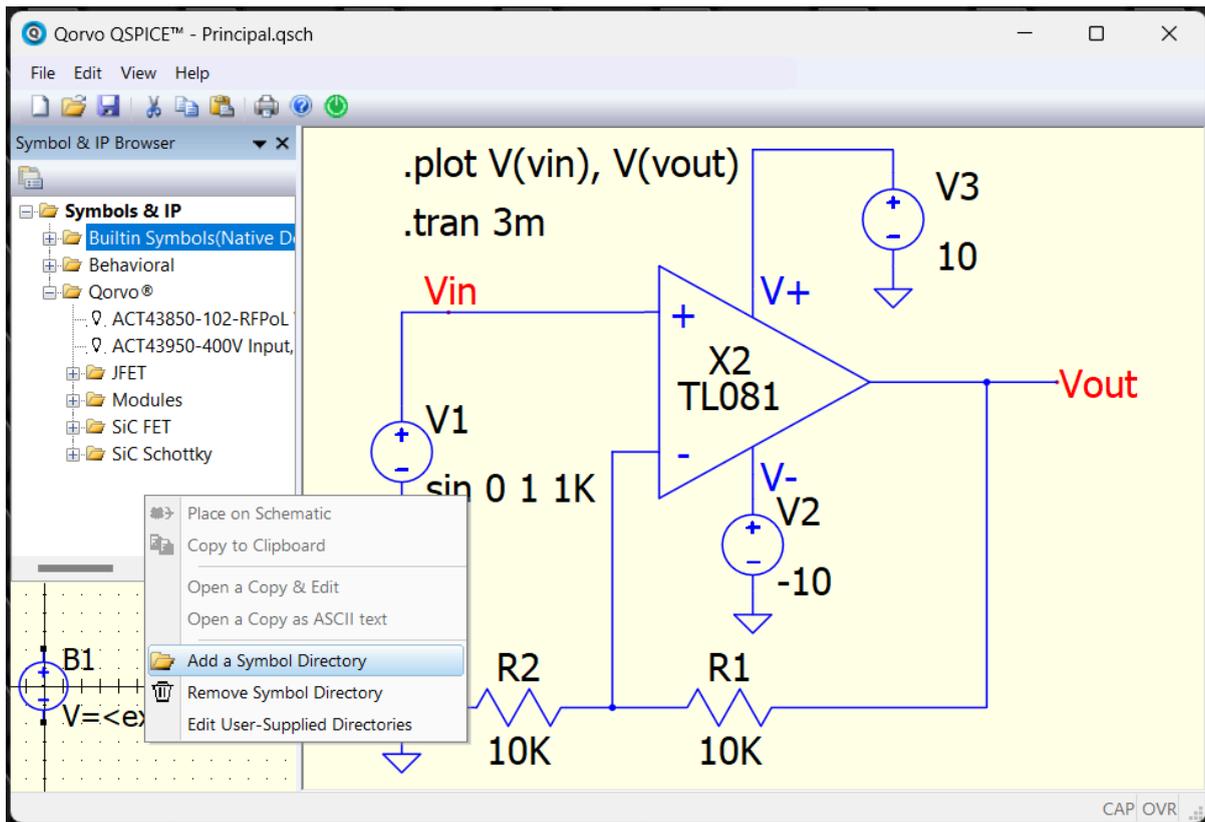
## Criando uma biblioteca de componentes

O Qspice permite que se crie uma pasta com os símbolos de componentes que criamos, assim é possível utilizá-los em outras simulações mais facilmente.

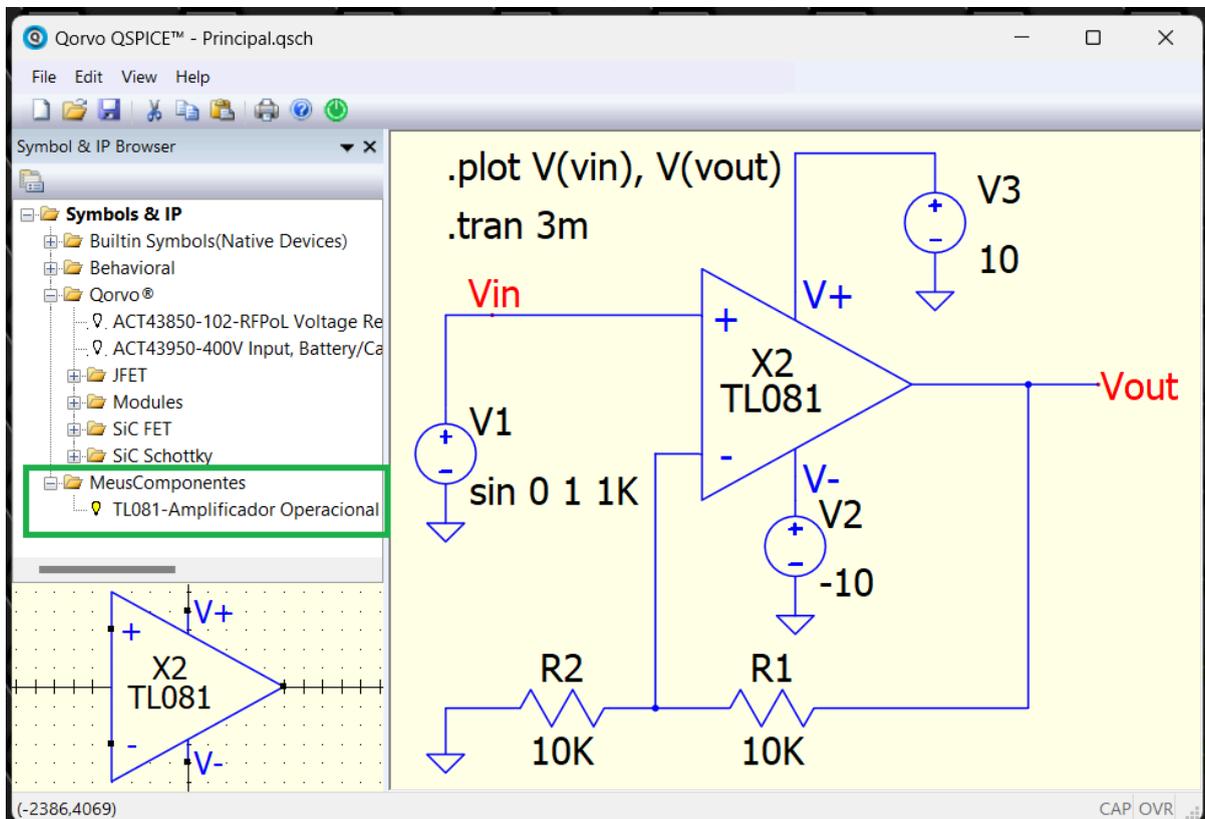
Para isso, basta criar uma pasta em seu computador e então, a partir do editor de símbolos, salvar o novo componente nesta pasta. Veja a imagem a seguir.



Agora, no Qspice, clique com o botão direito do mouse em uma área vazia da janela "Symbol & IP", como na imagem a seguir.



Agora, adicione a pasta na qual você salvou o símbolo anteriormente. A pasta é mostrada no Qspice com nosso componente, veja a imagem a a seguir.



Assim é possível criar quantas bibliotecas personalizadas for necessário. É importante destacar que os sub-circuitos criados no capítulo anterior também podem ser exportados como "Netlist" e então adicionados a biblioteca como componentes importados.

Este é apenas um das várias formas de adicionar componentes e bibliotecas a uma simulação do Qspice, existem ainda outras que não serão abordadas aqui.

## 8. Inclusão de códigos em C++ ou Verilog no Qspice

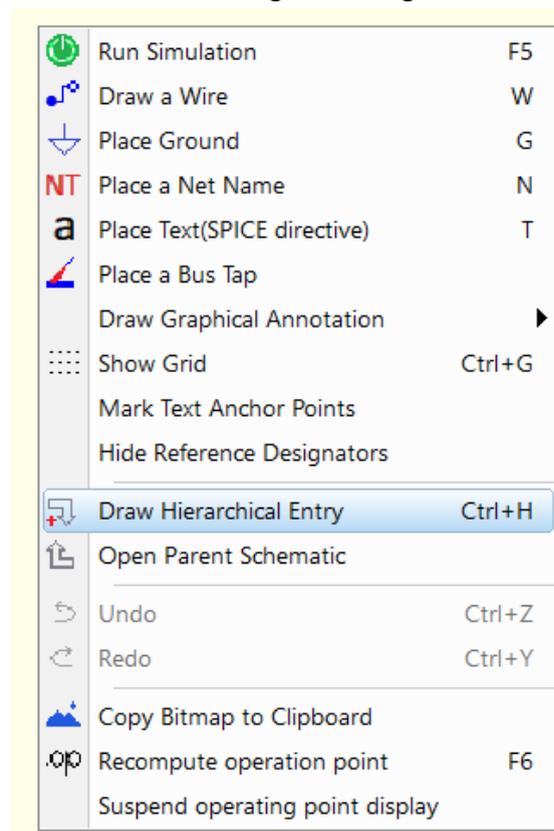
Integrar códigos em C++ ou Verilog é o principal diferencial do Qspice, se comparado a outros simuladores baseados em Spice. Esta funcionalidade é especialmente útil para projetos que envolvem controle digital ou processamento de sinal. O Verilog, como linguagem de descrição de hardware, permite que o comportamento de circuitos digitais complexos, como FPGAs ou ASICs, seja simulado diretamente dentro do ambiente de simulação. Já o C++ permite que algoritmos e modelos matemáticos sejam implementados para gerar ou processar sinais simulados, estendendo as capacidades do Qspice além da simples simulação de componentes passivos e ativos.

Ao permitir essa inclusão, o Qspice se torna uma ferramenta poderosa para o desenvolvimento de sistemas embarcados e eletrônica digital, facilitando a simulação de projetos que exigem a interação entre software e hardware. Dessa forma, é possível testar e depurar projetos digitais e algoritmos sem a necessidade de dispor do hardware físico logo nas primeiras fases de desenvolvimento.

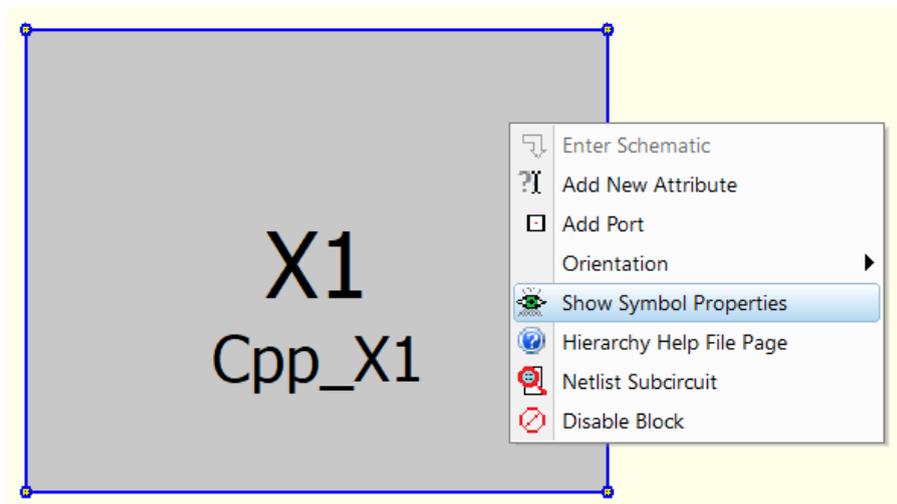
### Códigos em C++ no Qspice

Para inserir um trecho de código em C++ em uma simulação do QSPICE, siga os seguintes passos:

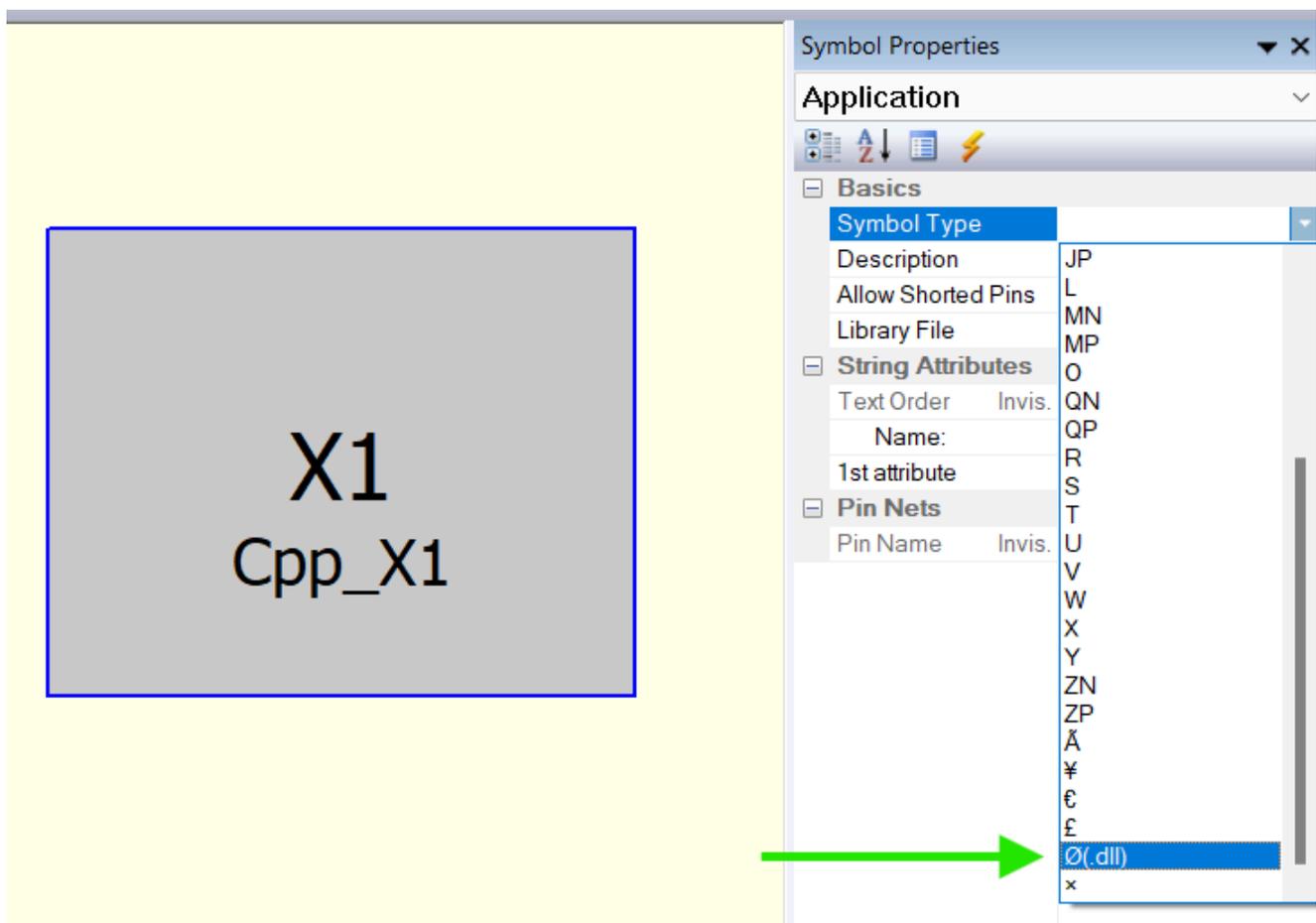
1. Criar uma caixa para conter o código, usando o botão direito e clicando em "Draw Hierarchical Entry", como mostrado na imagem a seguir.



2. Agora, é necessário configurar esta caixa para conter um código em C++. para isso, clique com o botão direito sobre a caixa, e selecione "Show Symbol Properties", para abrir a aba de propriedades se ainda não estiver aberta, veja a imagem a seguir.

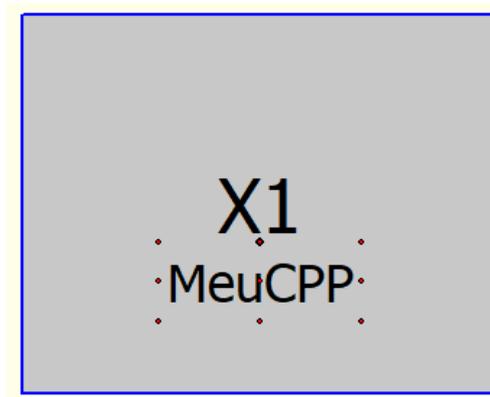


3. Agora, com a caixa selecionada mude a propriedade "Symbol Type" para ".dll" como indicado pela seta verde na imagem a seguir.

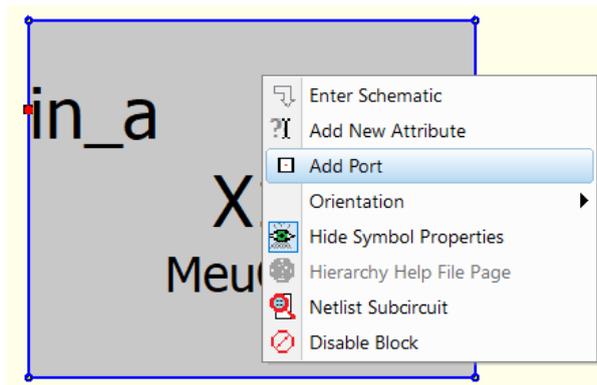


O código em C++ será compilado para um DLL e ligado a simulação.

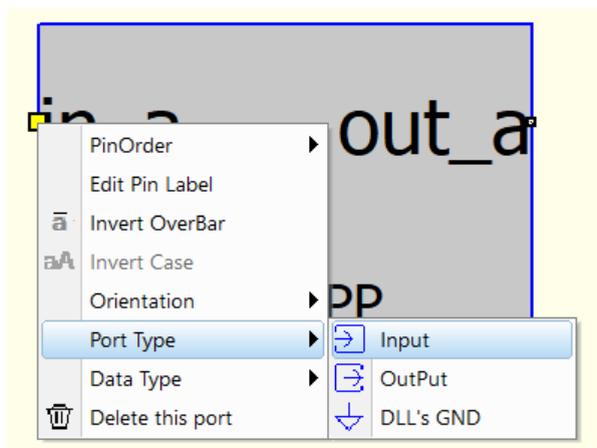
4. Agora, é necessário mudar o nome do bloco para um nome válido de função em C++, como na imagem a seguir.



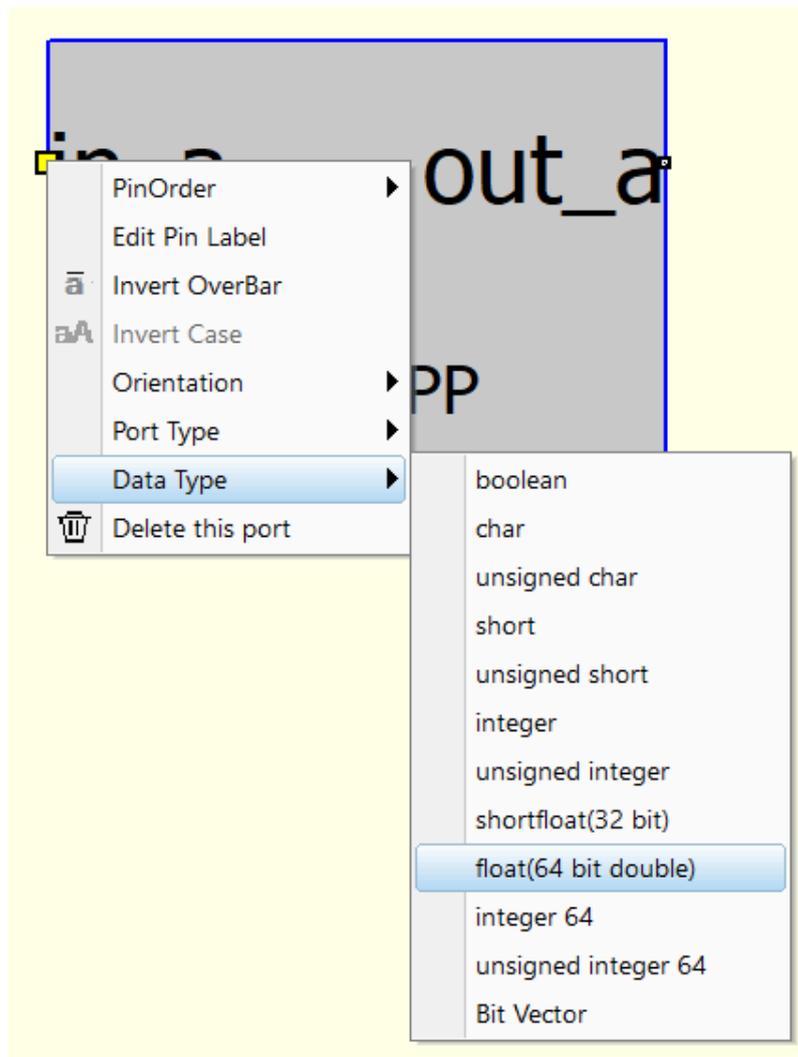
5. Agora, é necessário adicionar as portas de entrada e saída do nosso bloco de código. Para isso clique com o botão direito e selecione "Add Port", como na imagem a seguir. De nome as portas e ajuste os textos como desejado.



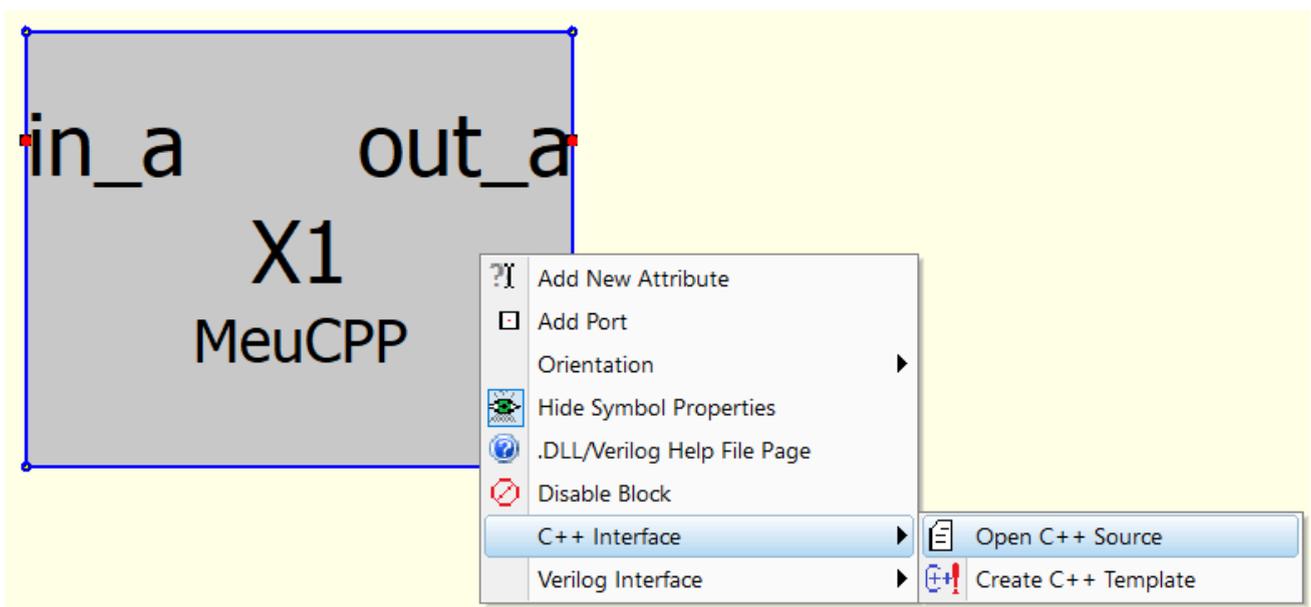
6. Agora, é necessário definir a direção de cada uma das portas. Clique com o botão direito sobre a porta, selecione "Port Type" e selecione "Input" ou "OutPut", como na imagem a seguir.



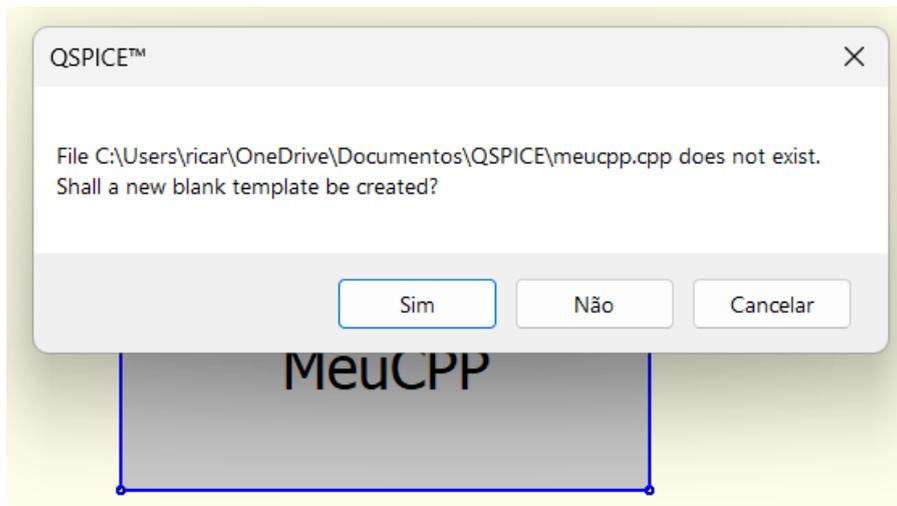
7. O próximo passo é definir também o tipo de dado de cada porta. Para isso clique com o botão direito sobre a porta, selecione "Data Type", selecione então o tipo de dado, geralmente será "float". Veja a imagem a seguir.



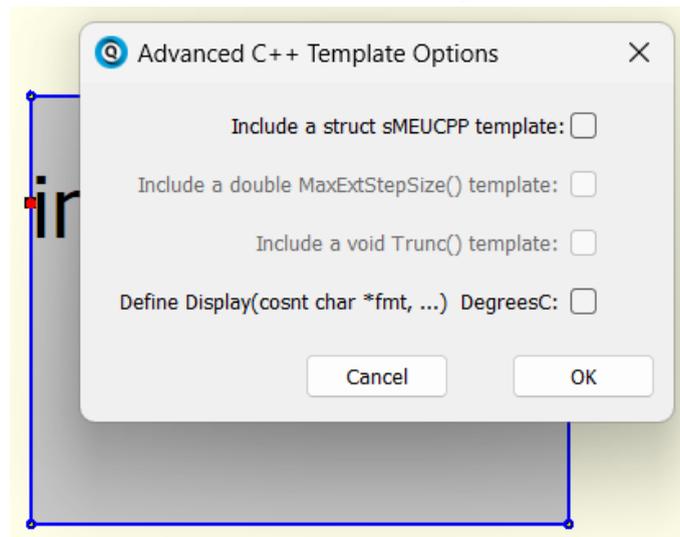
8. Agora, é possível gerar o fonte C++. Para isso, clique com o botão direito sobre o bloco e selecione "C++ Interface" e clique em "Open C++ Source", como na imagem a seguir.



Será aberta uma janela como a da imagem a seguir pedindo se é para criar o arquivo C++, que ainda não existe, clique "Sim".



Será aberta outra janela com propriedades mais avançadas, clique "OK".



Um editor com o código em C++ é então exibido, como na imagem a seguir.

A screenshot of a code editor window titled "Qorvo QSPICE™ - meucpp.cpp". The window has a menu bar with "File", "Edit", "View", and "Help", and a toolbar with various icons. The code content is as follows:

```
// Automatically generated C++ file on Mon Sep 30 10:05:35 2024
//
// To build with Digital Mars C++ Compiler:
//
// dmc -mn -WD meucpp.cpp kernel32.lib

union uData
{
    bool b;
    char c;
    unsigned char uc;
    short s;
    unsigned short us;
    int i;
    unsigned int ui;
    float f;
    double d;
    long long int i64;
    unsigned long long int ui64;
    char *str;
    unsigned char *bytes;
};
```

The status bar at the bottom left shows "Line: 1 Col: 1" and at the bottom right shows "CAP OVR".

9. Agora é possível implementar nosso código em C++. Como primeiro exemplo vamos fazer algo simples, onde a saída será a entrada multiplicada por ela mesmo. Para isso localize no final do código C++ a linha "// Implement module evaluation code here:". Abaixo desta linha adicione o código "out\_a = in\_a \* in\_a;". Use os nomes de entrada e saída que você criou. Veja a imagem a seguir.

```
extern "C" __declspec(dllexport) void meucpp(void **opaque, double t, union {
{
    double in_a = data[0].d; // input
    double &out_a = data[1].d; // output

    // Implement module evaluation code here:
    out_a = in_a * in_a;
}
}
```

Line: 38 Col: 4 CAP OVR

10. Agora, para compilar o código C++, clique com o botão direito na janela de código e selecione "Compile DLL". Se tudo estiver correto seu código será compilado. Veja a imagem a seguir.

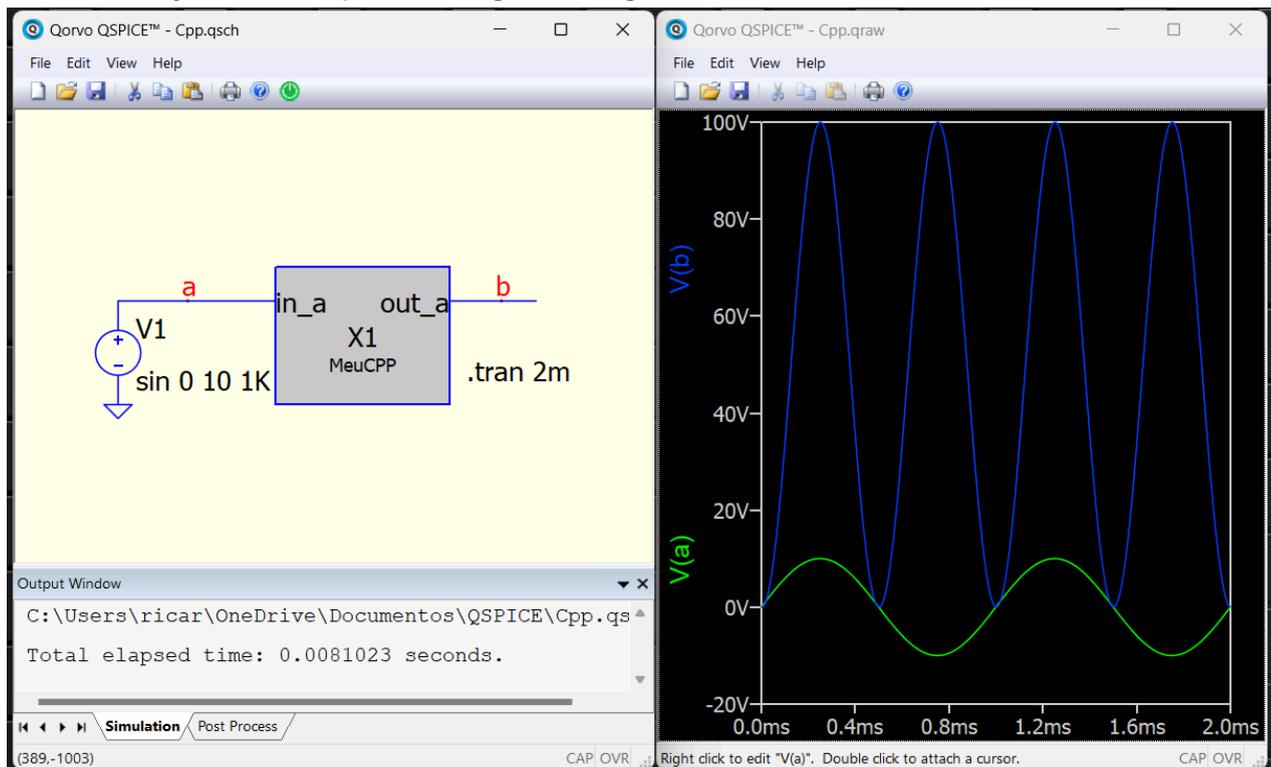
```
extern "C" __declspec(dllexport) void meucpp(void **opaque, double t, union {
{
    double in_a = data[0].d; // input
    double &out_a = data[1].d; // output

    // Implement module evaluation code here:
    out_a = in_a * in_a;
}
}
```

Compile DLL & Run	F5
Compile DLL	F6
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Select All	Ctrl+A

Compile this file to create a .DLL CAP OVR

11. Agora é possível fechar a janela de código e finalizar a simulação, utilizando seu código em C++. Veja um exemplo na imagem a seguir.



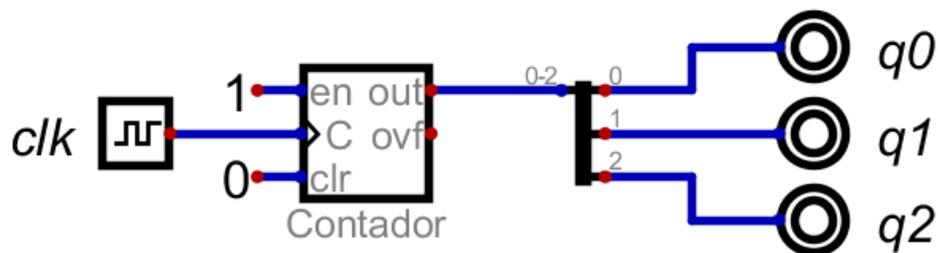
É importante ter em mente que o Qspice define vários parâmetros de interface entre o bloco de código C++ e o restante do circuito. Veja a imagem a seguir.

Name	Description	Units	Default
COUT	Output capacitance	C	0.
MAXTimestep	Maximum time between evaluations	s	Infinite
REF	Logic threshold for inputs declared as boolean	V	.5
ROUT	Output impedance	$\Omega$	1000.
VHIGH	Logic high level for outputs declared as boolean	V	1.
VLSB	ADC and DAC incremental voltage for an LSB change for multi-bit binary data types	V	1.

A impedância de saída, por exemplo é de 1000 Ohms, o que pode interferir no funcionamento do circuito.

## Códigos em Verilog no Qspice

Como exemplo de código em Verilog no Qspice vamos implementar o contador de 3 bits da imagem a seguir.



O código em Verilog para este circuito é o seguinte:

```

module DIG_Counter_Nbit
#(
    parameter Bits = 2
)
(
    output [(Bits-1):0] out,
    output ovf,
    input C,
    input en,
    input clr
);
    reg [(Bits-1):0] count;

    always @ (posedge C) begin
        if (clr)
            count <= 'h0;
        else if (en)
            count <= count + 1'b1;
    end

```

```

    assign out = count;
    assign ovf = en? &count : 1'b0;

    initial begin
        count = 'h0;
    end
endmodule

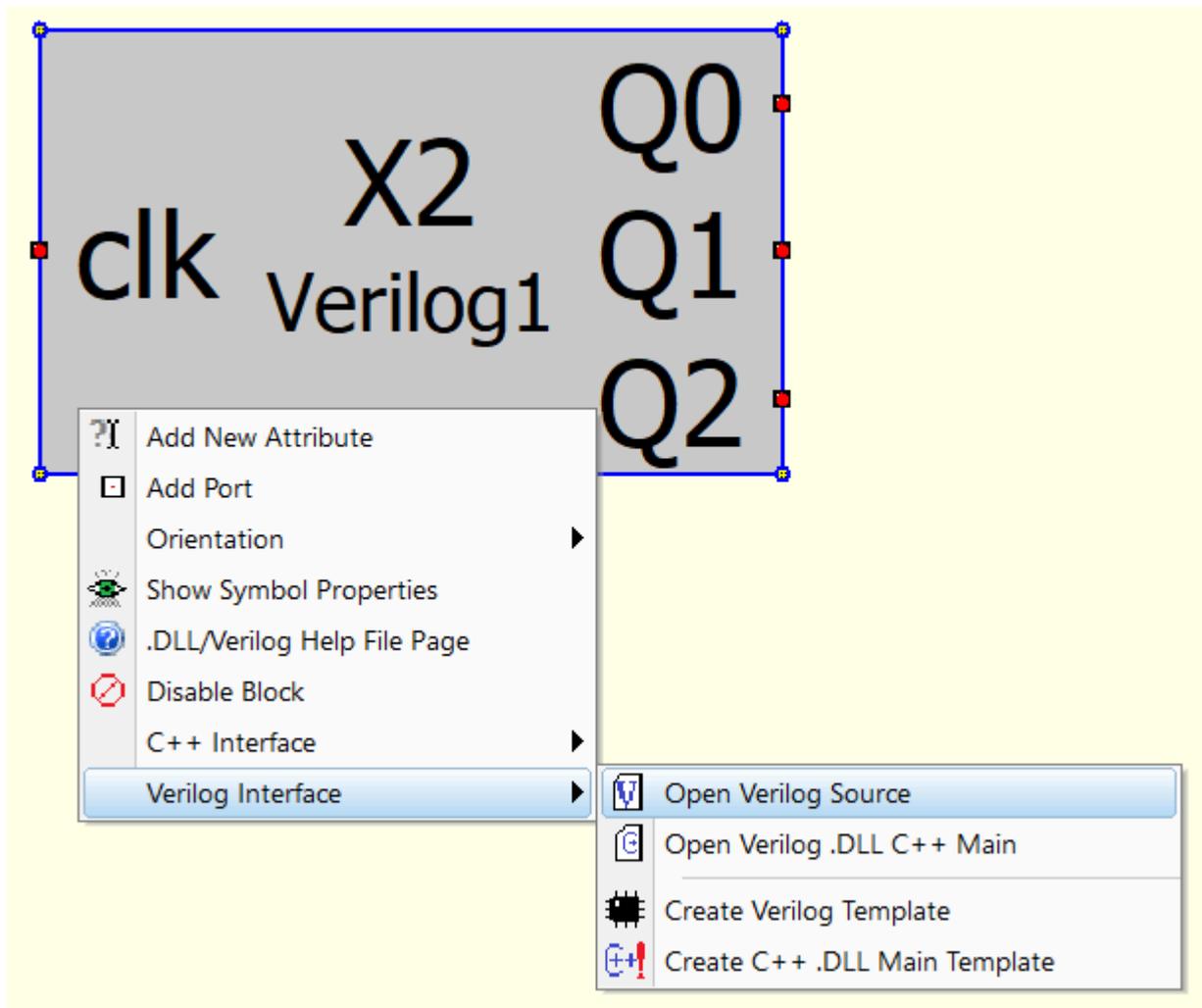
module cont (
    input clk,
    output q0,
    output q1,
    output q2
);
    wire [2:0] s0;
    DIG_Counter_Nbit #(
        .Bits(3)
    )
    DIG_Counter_Nbit_i0 (
        .en( 1'b1 ),
        .C( clk ),
        .clr( 1'b0 ),
        .out( s0 )
    );
    assign q0 = s0[0];
    assign q1 = s0[1];
    assign q2 = s0[2];
endmodule

```

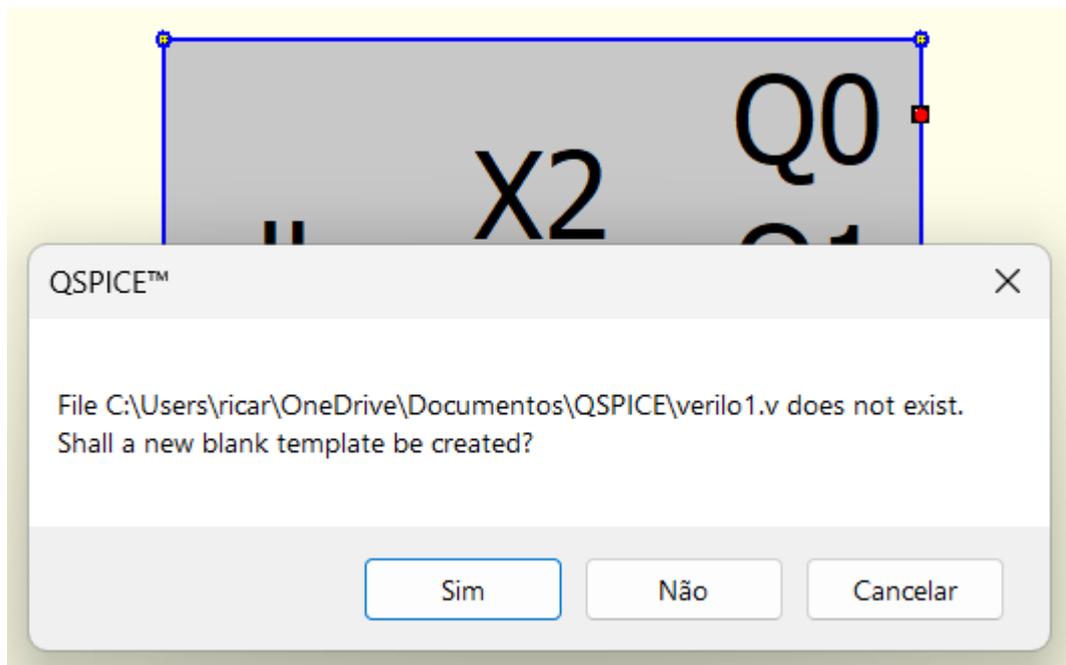
A criação de um bloco de código Verilog é muito semelhante a criação de um bloco de código C++, isso porque o Verilog vai ser convertido em C++ e compilado para um DLL para então ser simulado.

Assim, para criar um bloco com Verilog siga os passos de 1 a 7 da criação do bloco C++. É importante dar atenção aos tipos de dados das entradas e saídas, sinais digitais por exemplo devem ser declarados como "Boolean".

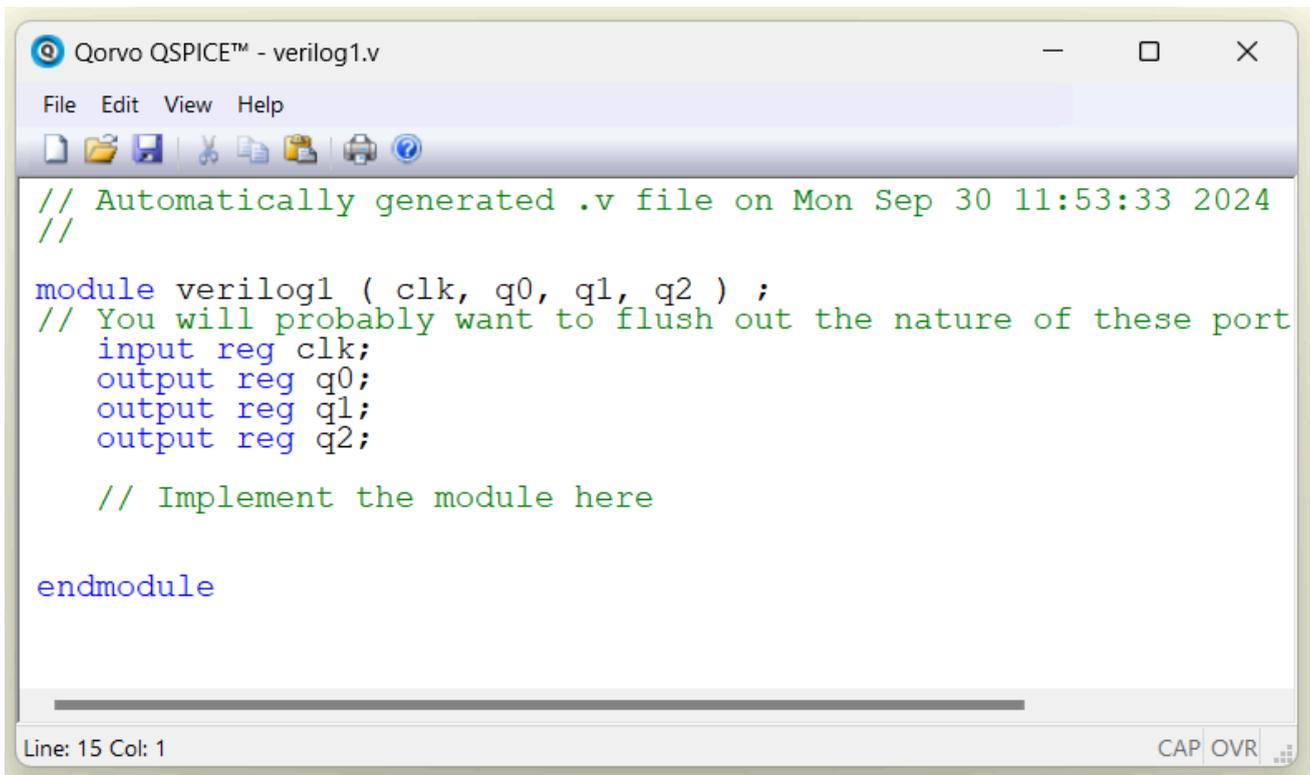
Com o bloco criado e as entradas e saídas definidas é possível criar o arquivo para conter o código Verilog. Para isso clique com o botão direito sobre o bloco e selecione "Open Verilog Source", como no exemplo da imagem a seguir.



Uma janela é aberta pedindo se é para criar um novo arquivo Verilog, clique "Sim", como na imagem a seguir.



A janela do editor de código é aberta contendo o modelo de código Verilog, como na imagem a seguir.



```
// Automatically generated .v file on Mon Sep 30 11:53:33 2024
//
module verilog1 ( clk, q0, q1, q2 ) ;
// You will probably want to flush out the nature of these port
input reg clk;
output reg q0;
output reg q1;
output reg q2;

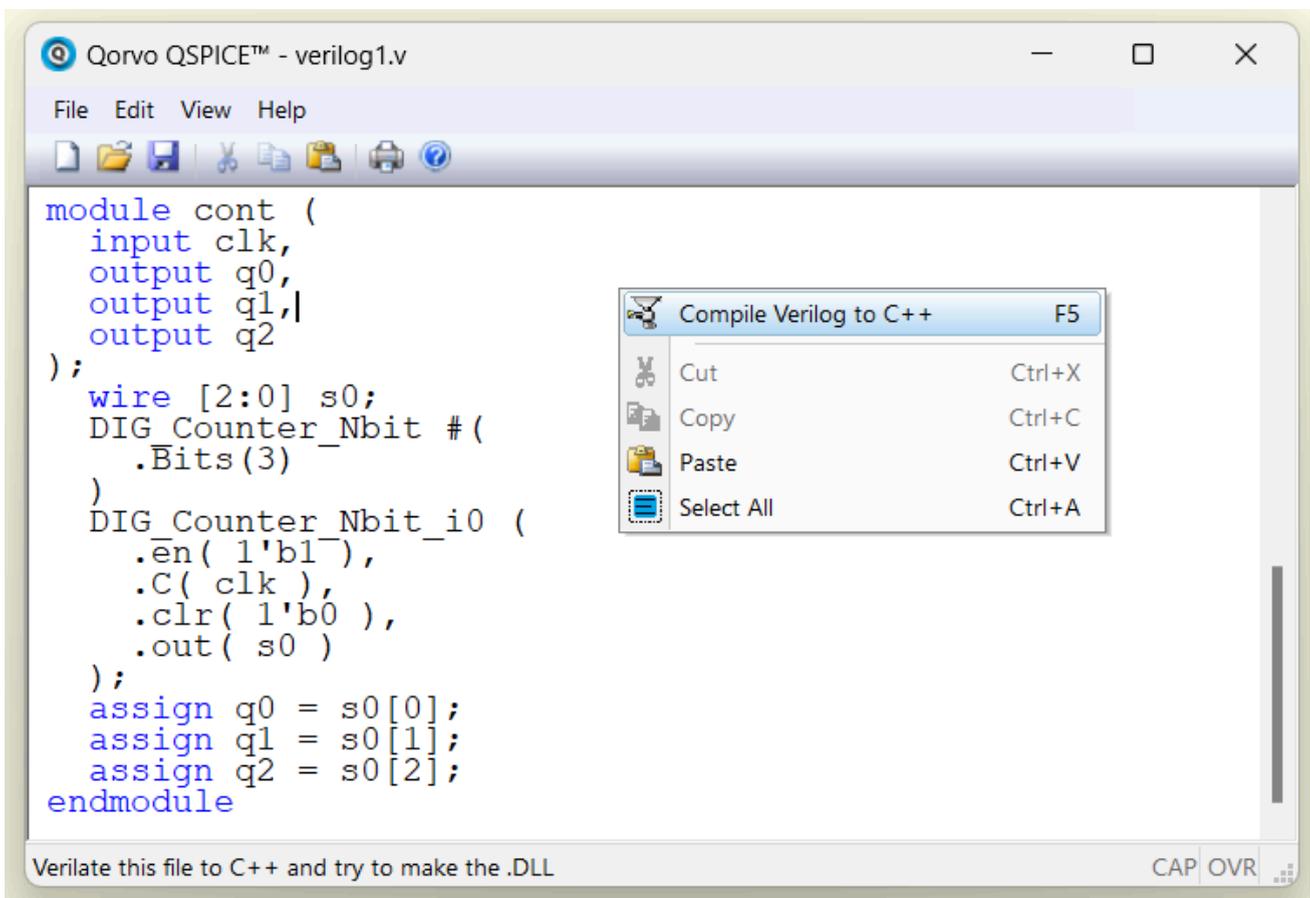
// Implement the module here

endmodule
```

Line: 15 Col: 1 CAP OVR

Observe que os nomes das entradas e saídas estão todos em minúsculo. Esta nomenclatura deve ser respeitada ou o código dará erro na compilação.

Após incluir seu código Verilog, basta clicar com o botão direito do mouse e selecionar "Compile Verilog to C++", como na imagem a seguir.



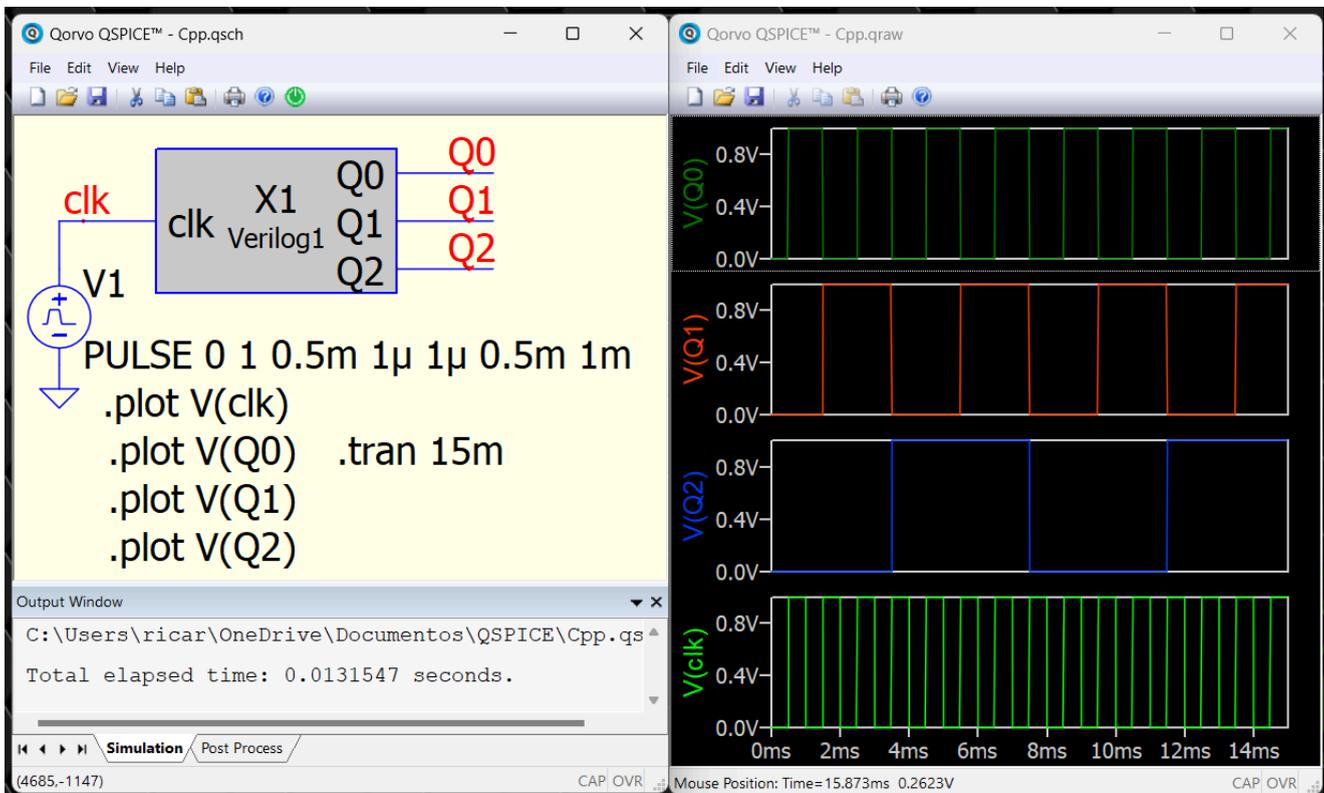
```
module cont (
input clk,
output q0,
output q1,
output q2
);
wire [2:0] s0;
DIG Counter_Nbit # (
.Bits(3)
)
DIG Counter_Nbit_i0 (
.en( 1'b1 ),
.C( clk ),
.clr( 1'b0 ),
.out( s0 )
);
assign q0 = s0[0];
assign q1 = s0[1];
assign q2 = s0[2];
endmodule
```

Verilate this file to C++ and try to make the .DLL CAP OVR

Context Menu:

- Compile Verilog to C++ F5
- Cut Ctrl+X
- Copy Ctrl+C
- Paste Ctrl+V
- Select All Ctrl+A

Se tudo deu certo seu código Verilog está pronto para ser simulado. Veja um exemplo na imagem a seguir.

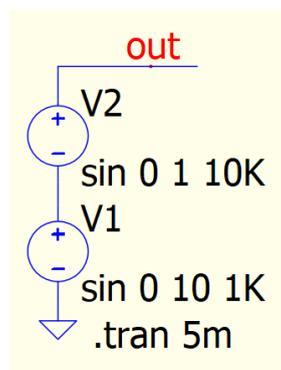


## 9. Análise de Resultados

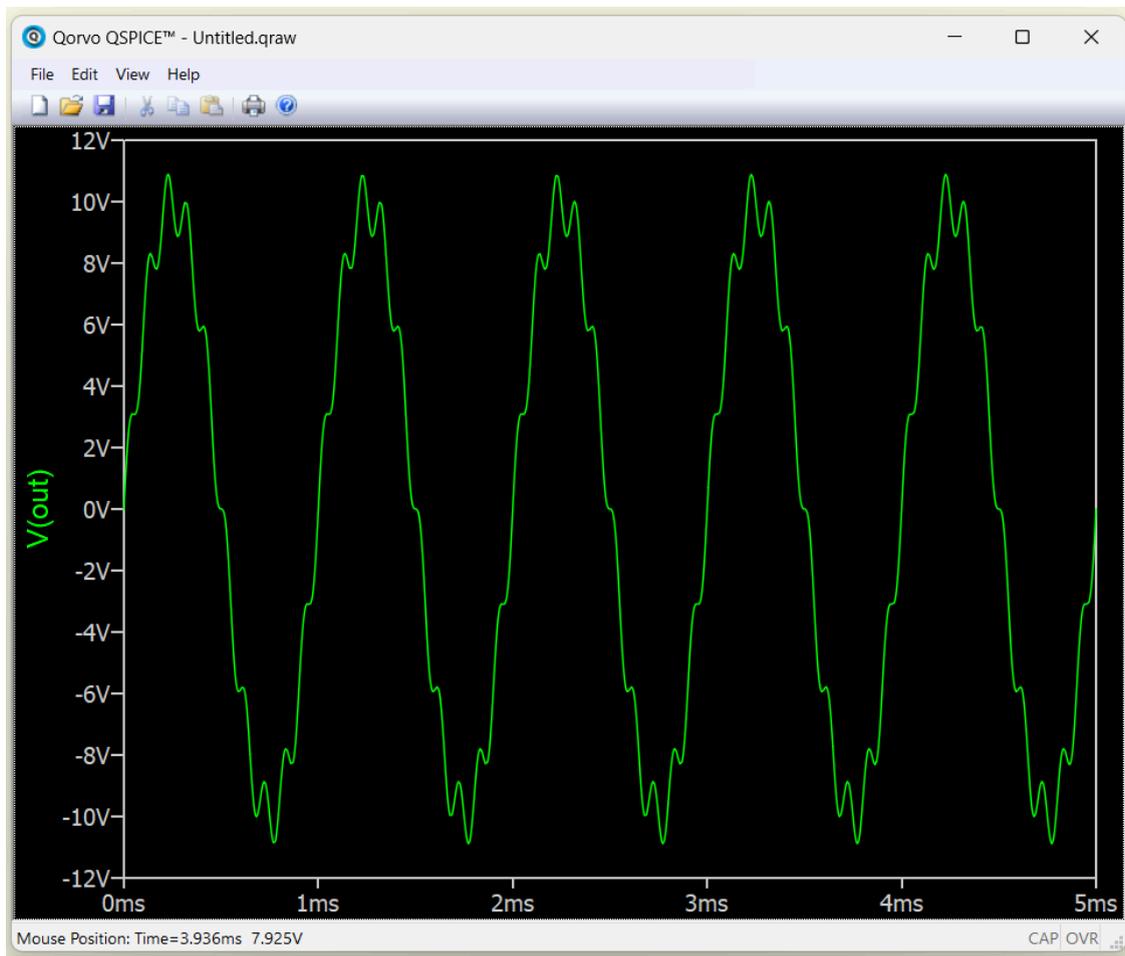
A análise de resultados é uma etapa fundamental no processo de simulação de circuitos. Após a execução de uma simulação, seja ela transiente, AC ou DC, a interpretação correta dos dados gerados permite que os projetistas avaliem o desempenho do circuito e identifiquem possíveis ajustes necessários. Compreender os resultados da simulação é crucial para validar o comportamento do projeto e garantir que ele atenda às especificações desejadas.

No Qspice, os resultados da simulação podem ser visualizados de várias formas, como gráficos de formas de onda, gráficos de resposta em frequência e listas de tensões e correntes em diferentes nós e componentes. Esses dados permitem que o engenheiro obtenha insights valiosos sobre como o circuito reage a diferentes estímulos, como alterações de tensão, corrente, frequência e até temperatura.

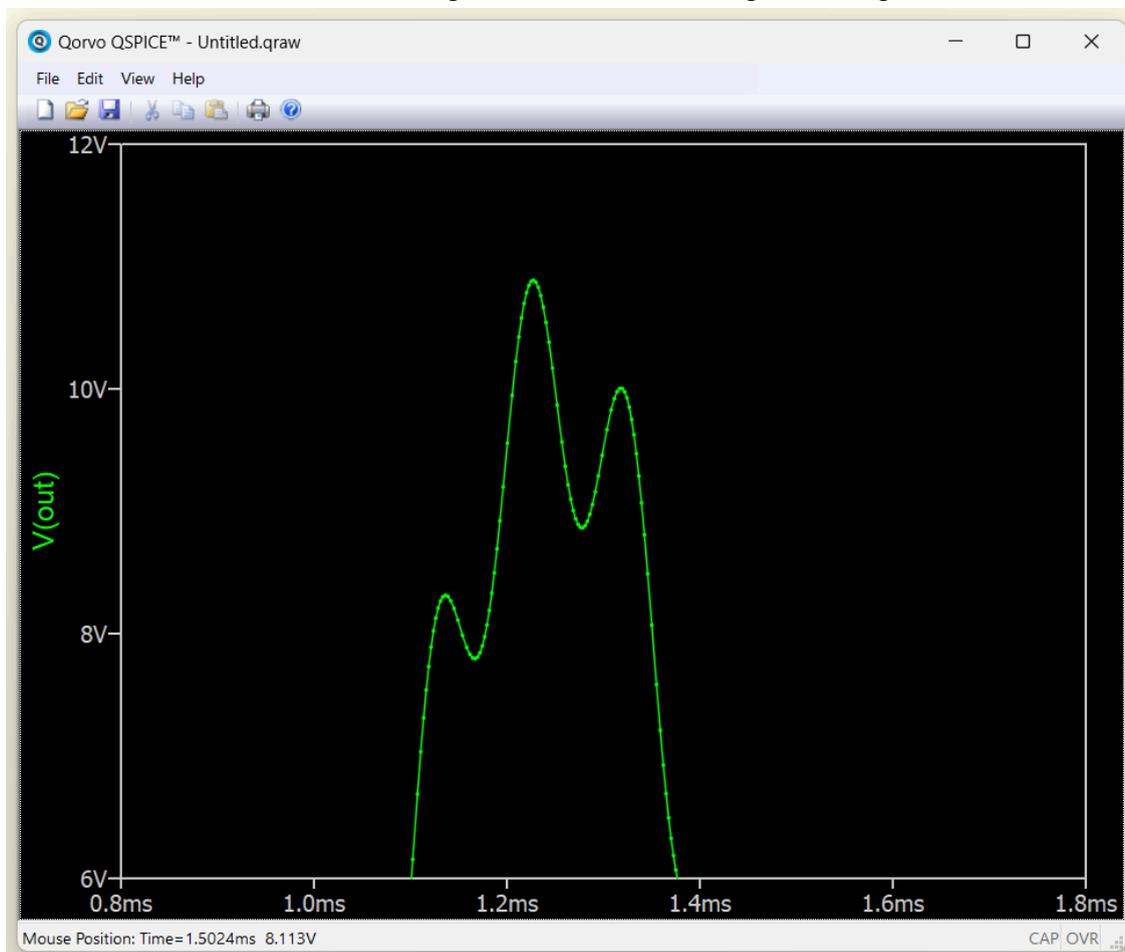
Para ilustrar as principais funcionalidades de análise de resultados do Qspice vamos utilizar inicialmente o circuito simples da imagem a seguir.



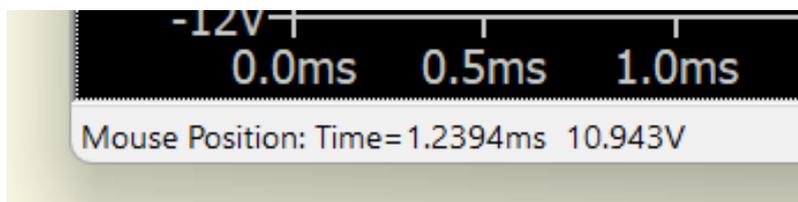
Para este circuito o resultado da simulação é apresentado na janela de gráficos apresentada na imagem a seguir.



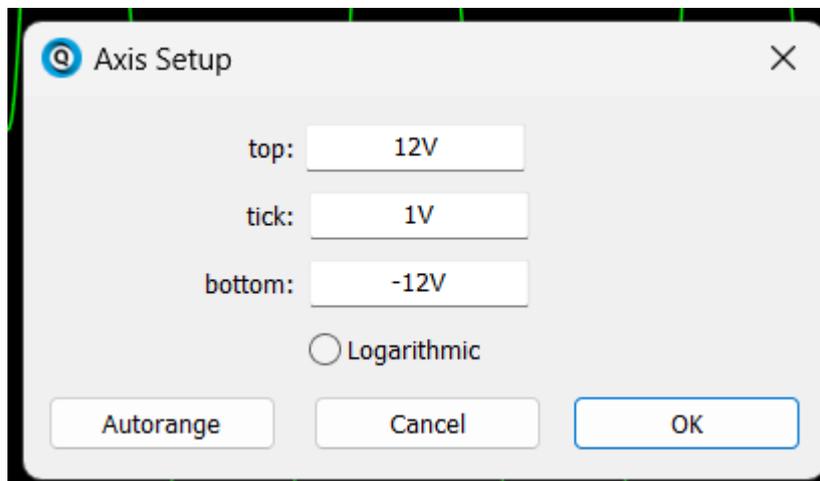
Vamos explorar um pouco mais esta janela. Desenhando uma área com o mouse, é possível fazer zoom em uma área do gráfico, como na imagem a seguir.



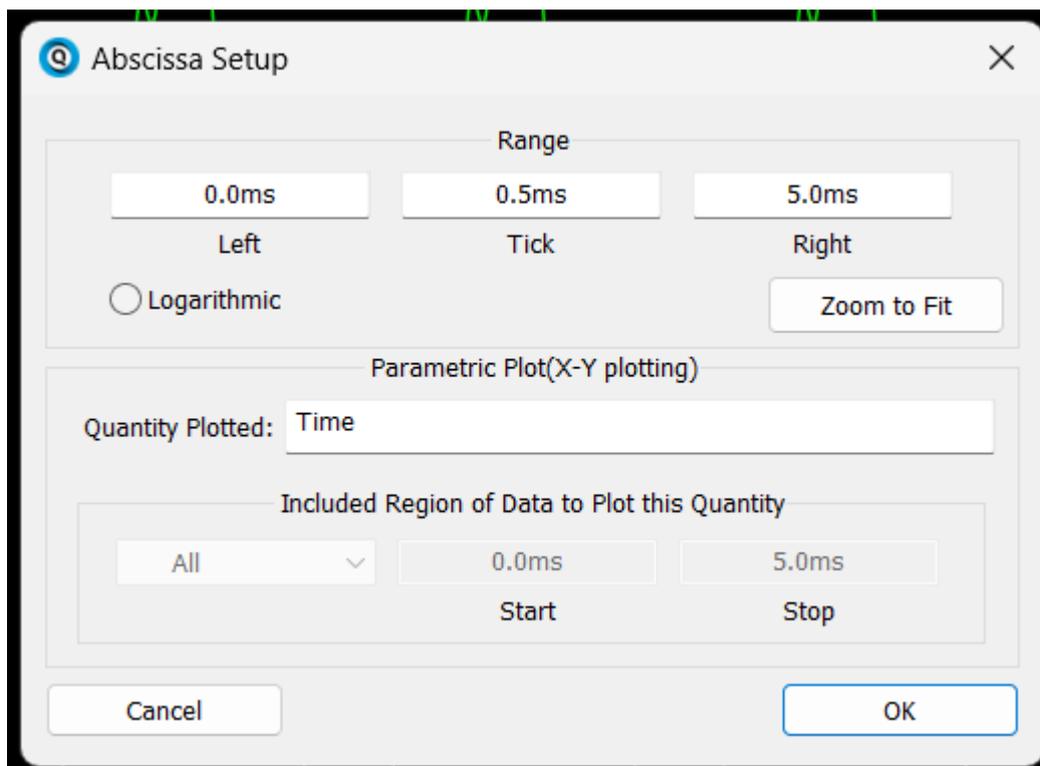
Se observarmos o parte inferior esquerda da janela, os valores da posição do mouse nos eixos é apresentada, veja a imagem a seguir. Esta informação é bastante útil para realizar medidas nos gráficos.



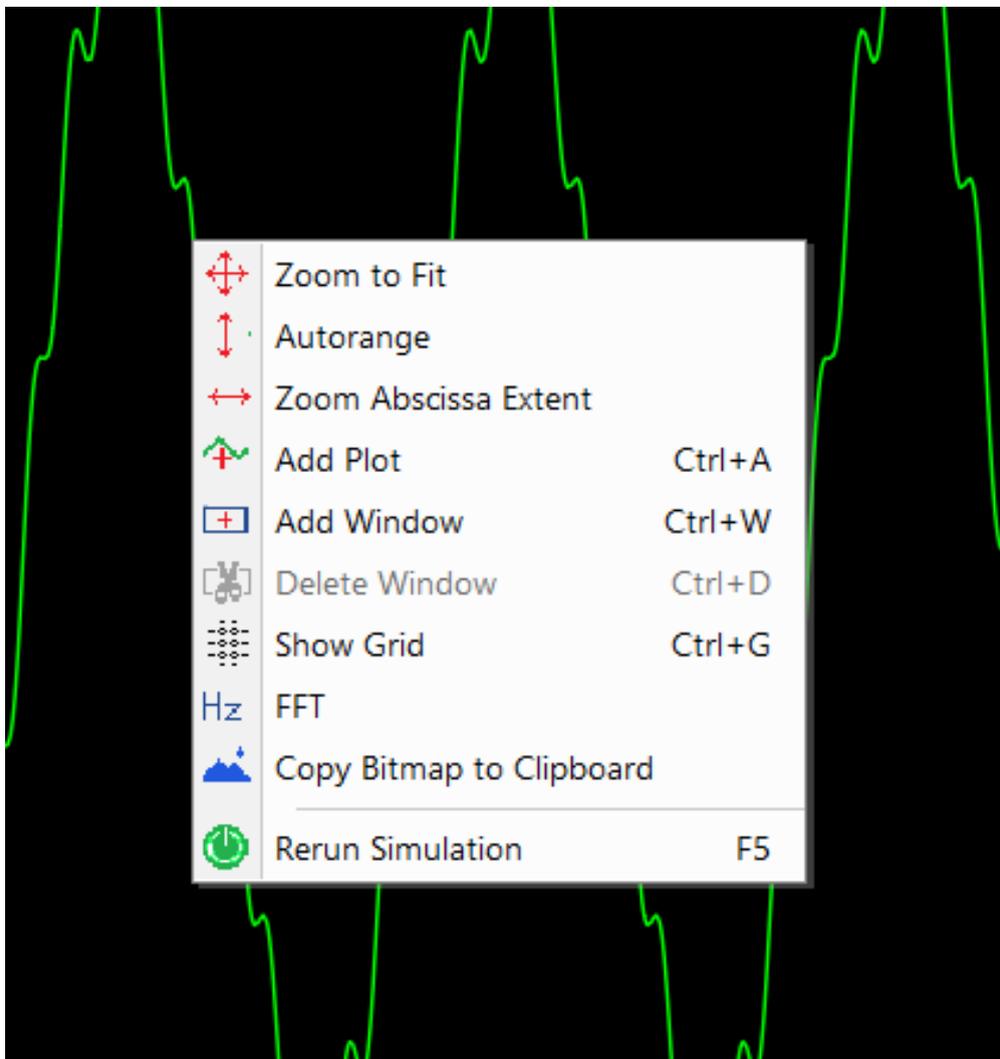
Clicando com o botão direito do mouse sobre a escala do eixo horizontal abre a janela de configuração deste eixo, como na imagem a seguir.



Também é possível clicar com o botão direito sobre a escala do eixo horizontal para abrir a janela de ajustes deste eixo. Veja a imagem a seguir.

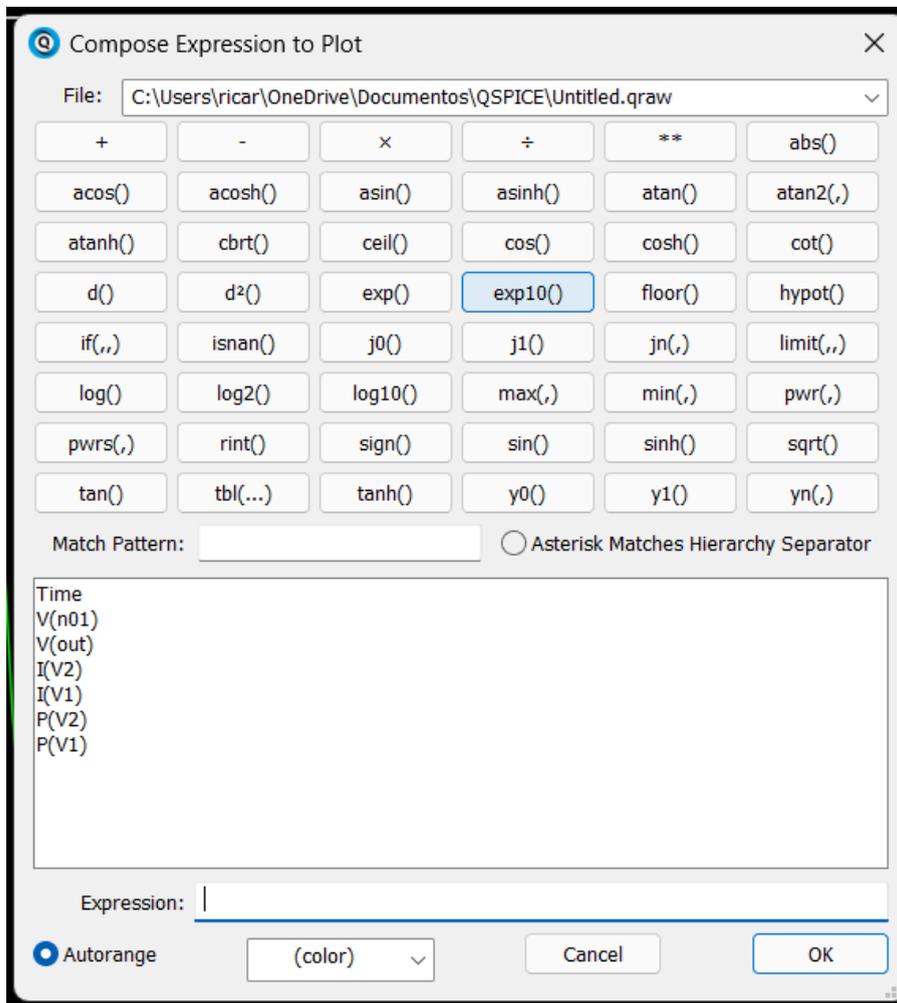


Clicar com o botão direito do mouse sobre a área do gráfico abre um menu com opções de ajuste dos gráficos, veja a imagem a seguir.



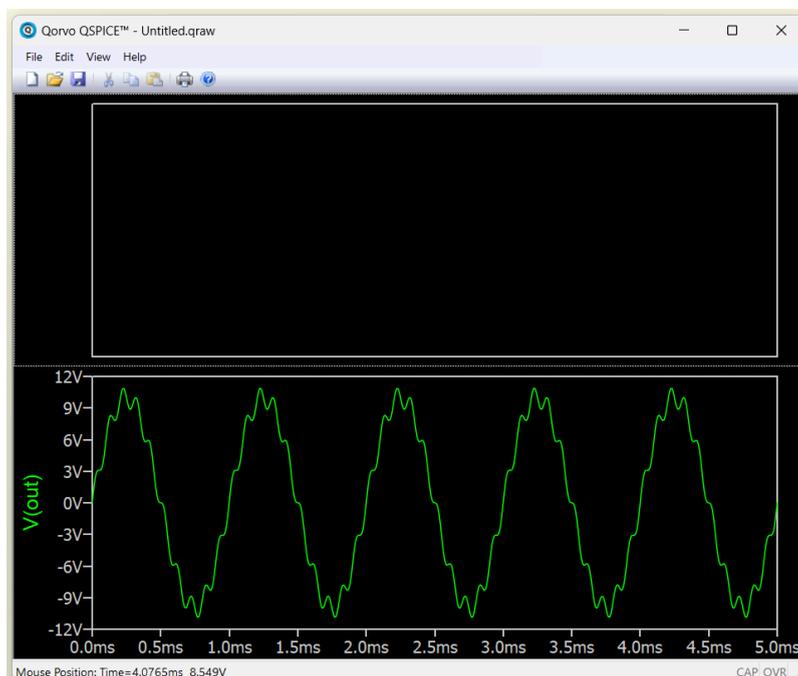
### A função de cada item é:

- **Zoom to Fit:** Ajusta automaticamente o zoom do gráfico para que todo o conteúdo plotado seja exibido dentro da janela. Isso é útil quando se deseja visualizar rapidamente todo o conjunto de dados da simulação.
- **Autorange:** Ajusta automaticamente os eixos do gráfico com base na amplitude dos sinais exibidos. Ele redefine a escala do gráfico para garantir que todas as formas de onda sejam visíveis na melhor escala possível.
- **Zoom Abscissa Extent:** Esta opção faz o zoom especificamente no eixo x (abscissa), focando no intervalo de tempo ou de frequência exibido. Ele expande ou contrai o eixo horizontal para mostrar a extensão completa dos dados ao longo do tempo.
- **Add Plot (Ctrl + A):** (O mesmo que clicar com o botão direito do mouse sobre a etiqueta com o nome do sinal no gráfico) Adiciona um novo gráfico à janela de saída. Isso permite que o usuário visualize múltiplas formas de onda em janelas separadas ou sobrepostas. Quando se seleciona esta opção a janela apresentada na figura a seguir é exibida.

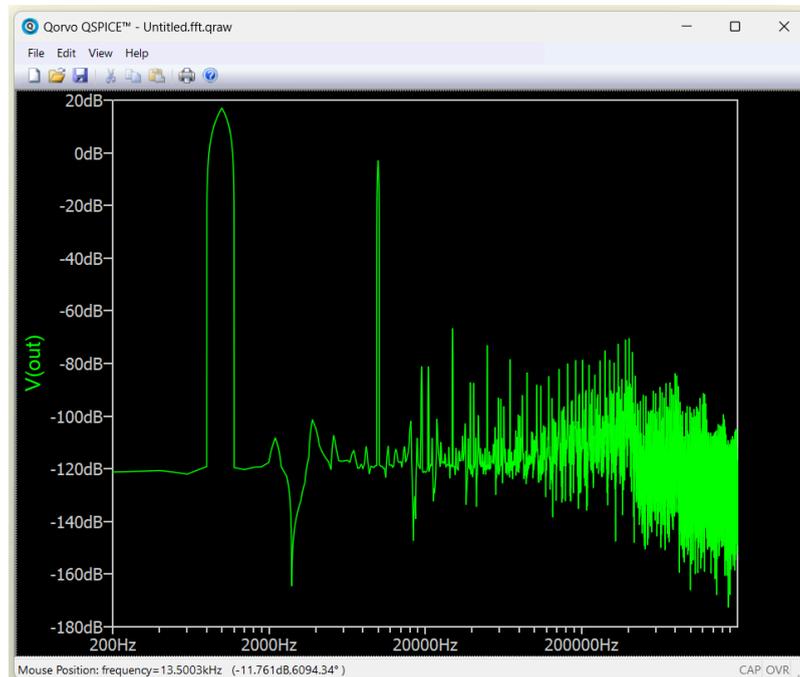


Aqui é possível selecionar qualquer sinal do circuito para ser plotado no gráfico. Além disso, é possível realizar dezenas de operações matemáticas entre estes sinais, o que permite buscar informações realmente complexas sobre o funcionamento de um circuito.

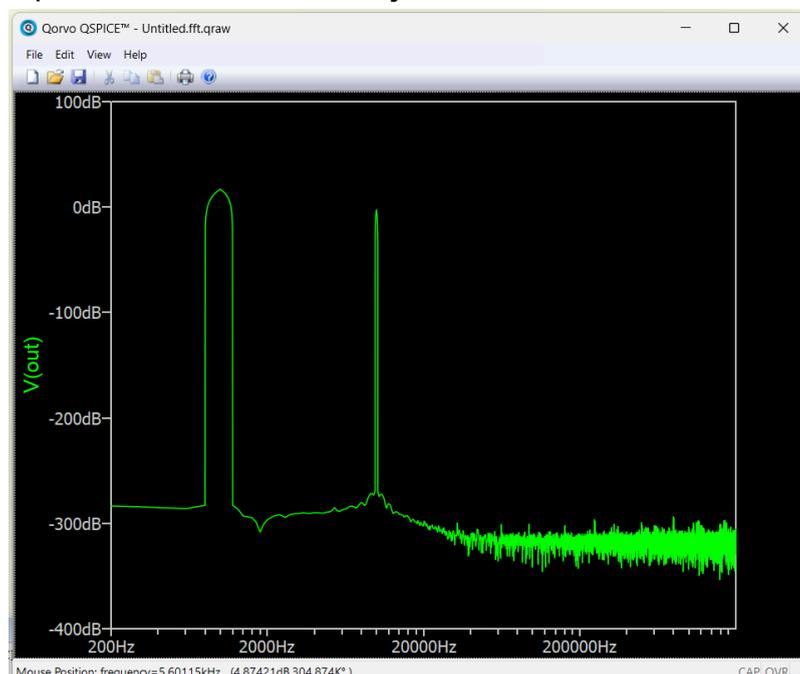
- **Add Window (Ctrl + W):** Adiciona uma nova janela de gráficos, onde o usuário pode plotar formas de onda adicionais. Isso ajuda a organizar a análise de sinais em diferentes janelas para facilitar a visualização. Veja um exemplo na imagem a seguir.



- **Delete Window (Ctrl + D):** Exclui a janela de gráfico atualmente selecionada. Isso é útil quando há muitas janelas abertas e você deseja remover algumas para limpar a área de trabalho.
- **Show Grid (Ctrl + G):** Exibe uma grade sobre o gráfico, facilitando a leitura e a interpretação dos valores de tempo e amplitude. A grade ajuda a alinhar visualmente os pontos de interesse com os eixos.
- **FFT (Fast Fourier Transform):** Realiza a Transformada Rápida de Fourier (FFT) no sinal atualmente exibido. Esta ferramenta é extremamente útil para converter um sinal do domínio do tempo para o domínio da frequência, permitindo a análise espectral. A imagem a seguir mostra a FFT da saída atual.

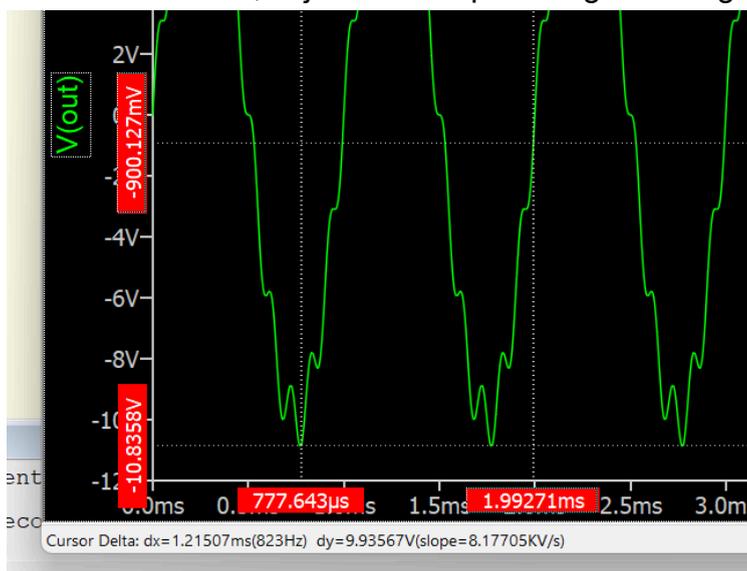


A FFT é bastante sensível aos parâmetros utilizados na simulação e no ajuste da ferramenta de o gráfico da imagem a seguir é a FFT da mesma saída só que com um MAXSTEP mais pequeno. Observe a diferença.



- **Copy Bitmap to Clipboard:** Copia a imagem do gráfico atual para a área de transferência, permitindo que o usuário cole o gráfico em outros documentos ou apresentações como uma imagem.
- **Rerun Simulation (F5):** Executa novamente a simulação com os mesmos parâmetros. Isso é útil se o usuário fizer alterações no circuito ou nos parâmetros de simulação e quiser atualizar o gráfico de saída com os novos resultados.

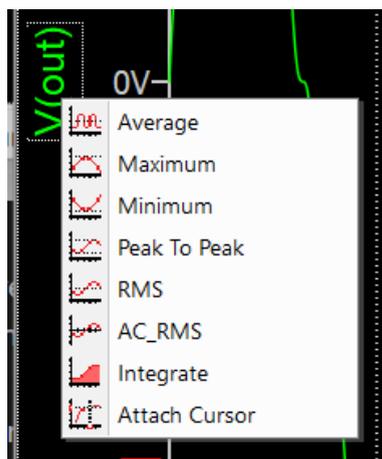
Também é possível atrelar um ou dois cursores a um sinal, para isso basta dar clique duplo sobre a etiqueta com o nome do sinal, veja um exemplo na figura a seguir.



Observe que as informações importantes relacionadas aos dois cursores são mostradas na parte inferior esquerda da janela do gráfico.

Também é possível dar um clique duplo sobre o cursor para definir seu valor.

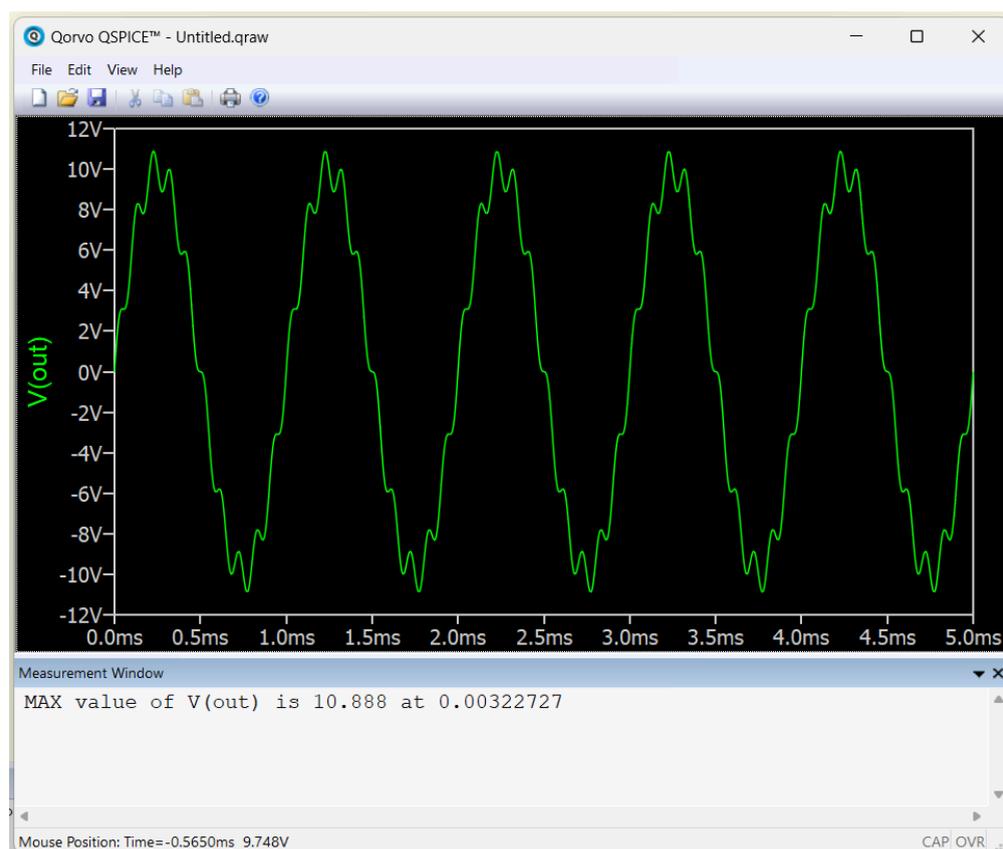
Segurando a tecla "CTRL" enquanto clica com o botão direito do mouse sobre a etiqueta com o nome de uma forma de onda irá abrir o menu exibido na imagem a seguir.



- **Average:** Calcula a média do sinal ao longo do intervalo de tempo ou do período selecionado. Isso fornece o valor médio (DC) da forma de onda, útil para identificar o deslocamento médio do sinal.
- **Maximum:** Determina o valor máximo atingido pela forma de onda ao longo do intervalo selecionado. Este valor representa o pico superior da forma de onda, sendo útil para análise de picos de tensão ou corrente em circuitos.

- **Minimum:** Calcula o valor mínimo da forma de onda no intervalo especificado. Esse valor representa o pico inferior da forma de onda, essencial para medir quedas de tensão ou corrente.
- **Peak To Peak:** Mede a diferença entre o valor máximo e o valor mínimo do sinal, ou seja, a amplitude total da oscilação do sinal. Isso é importante para medir a faixa completa de variação do sinal.
- **RMS (Root Mean Square):** Calcula o valor eficaz (RMS) da forma de onda. Este valor é especialmente útil em circuitos AC para determinar a potência dissipada, pois o valor RMS é equivalente ao valor de uma corrente ou tensão DC que produziria a mesma potência em uma resistência.
- **AC\_RMS:** Similar ao cálculo RMS, mas foca no valor RMS do componente AC do sinal, excluindo o valor médio DC. Isso é útil para analisar apenas as oscilações AC de um sinal que também tem um componente DC.
- **Integrate:** Integra o sinal ao longo do tempo. A integração de uma forma de onda pode ser usada para calcular a carga total transportada por uma corrente ou a energia total dissipada em um determinado período.
- **Attach Cursor:** (O mesmo que dar clique duplo sobre a etiqueta com o nome do sinal) Anexa um cursor ao gráfico para medir diretamente o valor do sinal em um ponto específico no tempo. Isso é útil para medições precisas em momentos ou intervalos específicos do gráfico.

Como os valores resultantes das funções deste menu não são formas de onda e sim valores numéricos, uma nova aba se abre na parte inferior da janela do gráfico para apresentar estes resultados. A imagem a seguir mostra o resultado da função "Maximum" na forma de onda de saída.



## 10. Recursos Adicionais

O Qspice é um simulador relativamente novo, e assim ainda não se tem tanto material disponível. A principal fonte de informações é a documentação do próprio simulador. A seguir estão alguns sites e vídeos que também podem ser úteis:

Site Oficial: <https://www.qorvo.com/design-hub/design-tools/interactive/qspice/>

Forum do Qspice: <https://forum.qorvo.com/c/qspice/9>

Play list Qspice: [https://youtube.com/playlist?list=PLJN-MF0N390OCfi4ZJBTomUMdu6BQLSGv&si=\\_I18TsSIOfqRfVAc](https://youtube.com/playlist?list=PLJN-MF0N390OCfi4ZJBTomUMdu6BQLSGv&si=_I18TsSIOfqRfVAc)

Repositório de Robert Dunn: <https://github.com/robdunn4/QSpice>

Repositório de KSKelvin: <https://github.com/KSKelvin-Github/Qspice/>

Repositório de modelos de componentes: <https://github.com/kicad-spice-library/KiCad-Spice-Library>

Repositório de Marcos Alonso: <https://github.com/marcosalonsoelectronics/website/tree/main/Qspice>

Play list Qspice: [https://youtube.com/playlist?list=PLq\\_MUr0hT\\_-kn89\\_adwe6QNpnLpZtJel-&si=IYeeojl\\_k7PeCPI8](https://youtube.com/playlist?list=PLq_MUr0hT_-kn89_adwe6QNpnLpZtJel-&si=IYeeojl_k7PeCPI8)