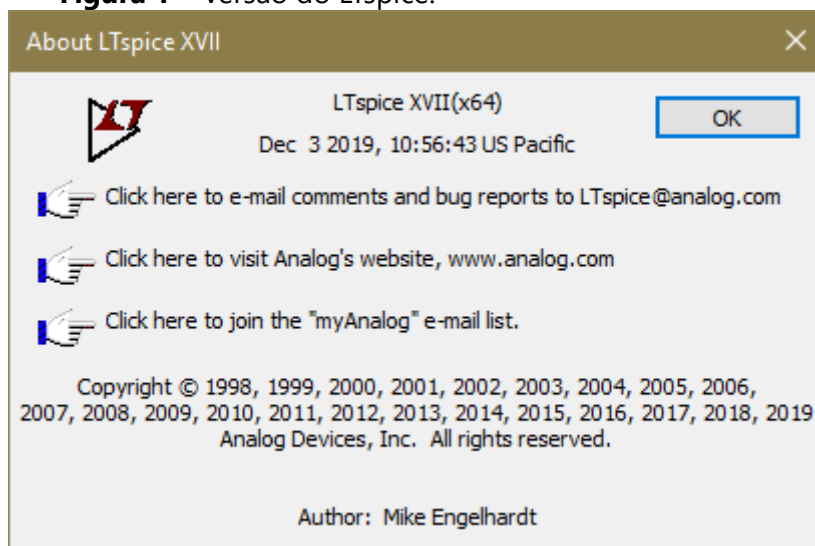


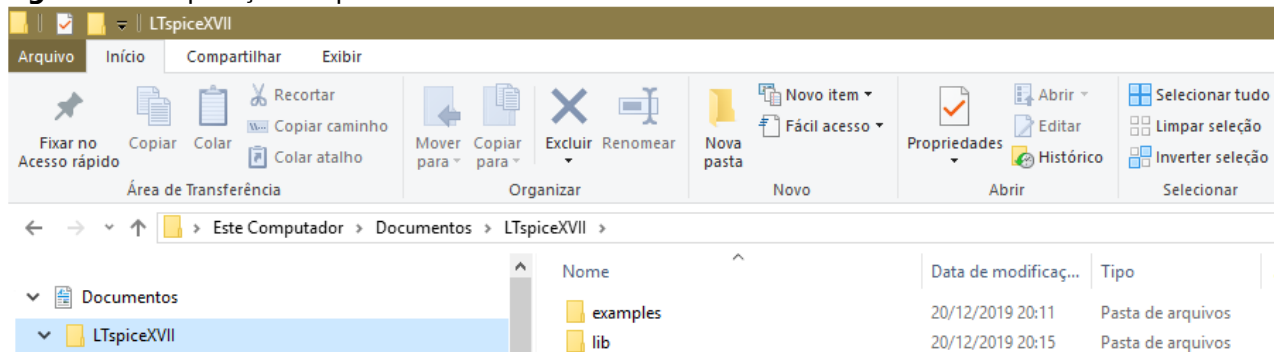
Nesse terceiro artigo sobre LTspice, vamos ver como é organizada a biblioteca de componentes. A versão que estou usando como referência é mostrada a seguir.

Figura 1 – Versão do LTspice.



Nessa versão para Windows, o LTspice é instalado numa pasta recomendada ou outra que você escolher, mas, na pasta [Documentos] é criado uma estrutura de biblioteca e exemplos para trabalho, conforme mostrado a seguir.

Figura 2 – Disposição de pastas.

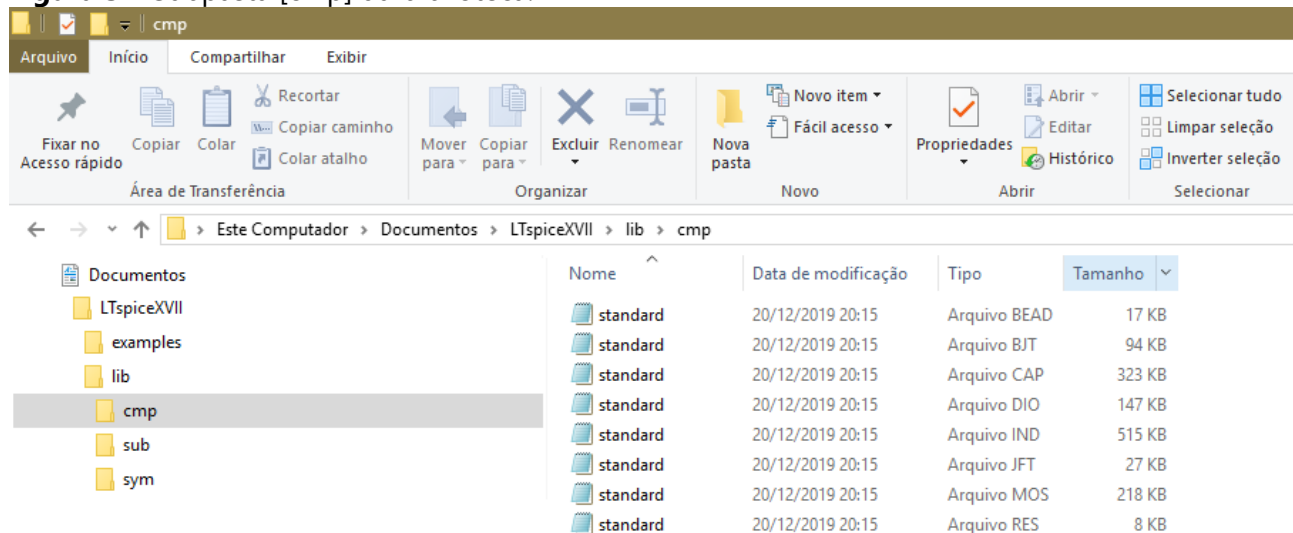


O foco é na pasta [lib], onde componentes, símbolos e modelos spice são organizados. Nesta pasta encontraremos uma subpasta, nomeada [cmp], e seus arquivos. Observe que temos oito arquivos com o nome 'standard', mas, cada um tem uma extensão diferente.

Quando selecionamos um resistor e vamos escolher um específico na lista, o arquivo 'standard.res' é a base de dados. Para capacitor é o arquivo 'standard.cap'. Para indutor é o arquivo 'standard.ind'. Para diodo é o arquivo 'standard.dio'. Para transistor bipolar é o arquivo 'standard.bjt'. Para transistor de efeito de campo é o arquivo 'standard.jft'. Para transistor MOSFET é o arquivo 'standard.mos', e para alguns componentes da Würth Elektronik é o arquivo 'standard.bead'.

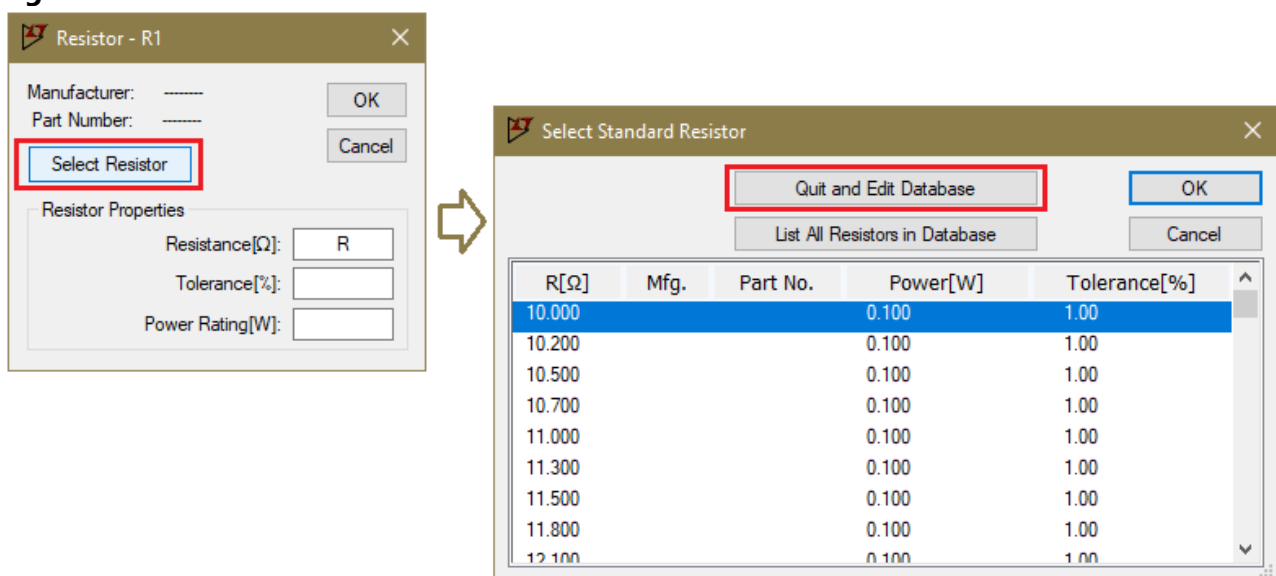
Esses arquivos estão dispostos como mostrado a seguir. Observe que as extensões dos arquivos informam que componentes estão contidos nos mesmos.

Figura 3 – Subpasta [cmp] da biblioteca.



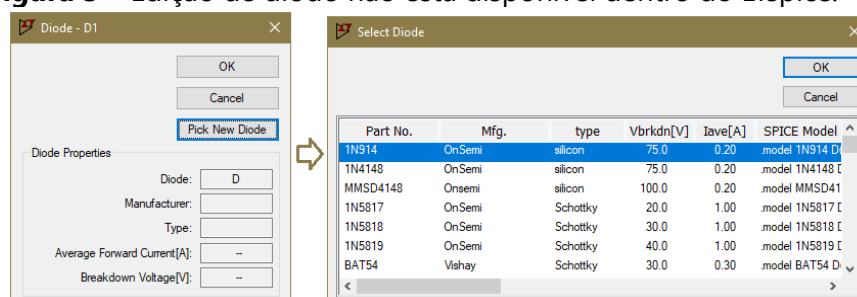
No LTspice, quando estiver trabalhando é possível editar alguns desses arquivos diretamente. Eles são acessados da seguinte forma.

Figura 4 – Editar a base de dados de resistores.



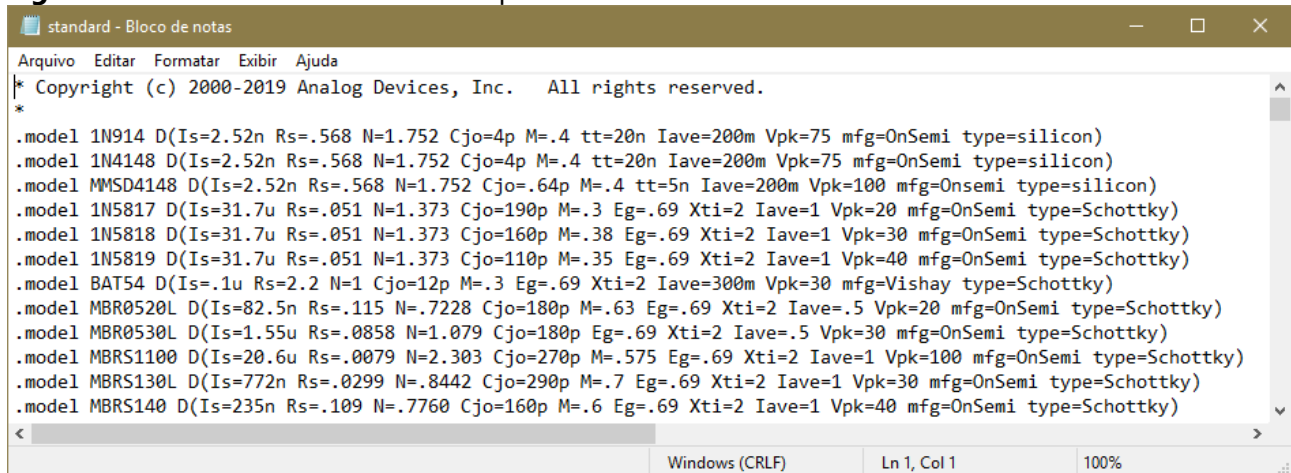
Outros componentes não são editados dentro do LTspice, mas, acessando o arquivo 'standard.?', por exemplo, o arquivo 'standard.dio'. Observe que dentro do LTspice não temos a opção para editar a base de dados.

Figura 5 – Edição de diodo não está disponível dentro do LTspice.



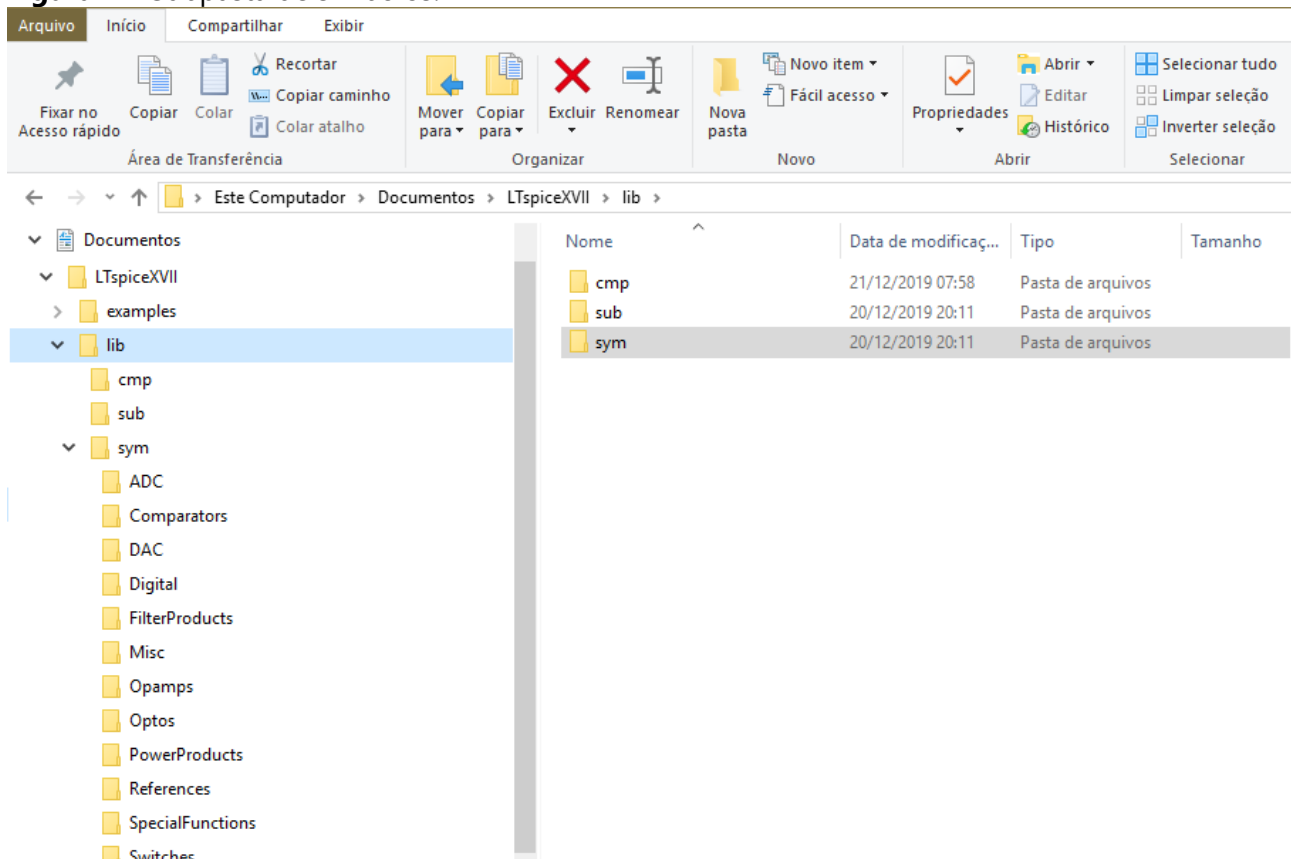
Seguindo com o exemplo do arquivo 'standard.dio', podemos editá-lo abrindo o arquivo e modificando valores ou inserindo novos modelos spice de diodos.

Figura 6 – Editando diretamente o arquivo 'standard.dio'.



Todos os componentes têm símbolos e são encontrados na pasta [sym], e em suas subpastas, como por exemplo, [Digital], [ADC], [Optos] e outras como pode ser visto a seguir. A extensão dos arquivos de símbolos é *.asy.

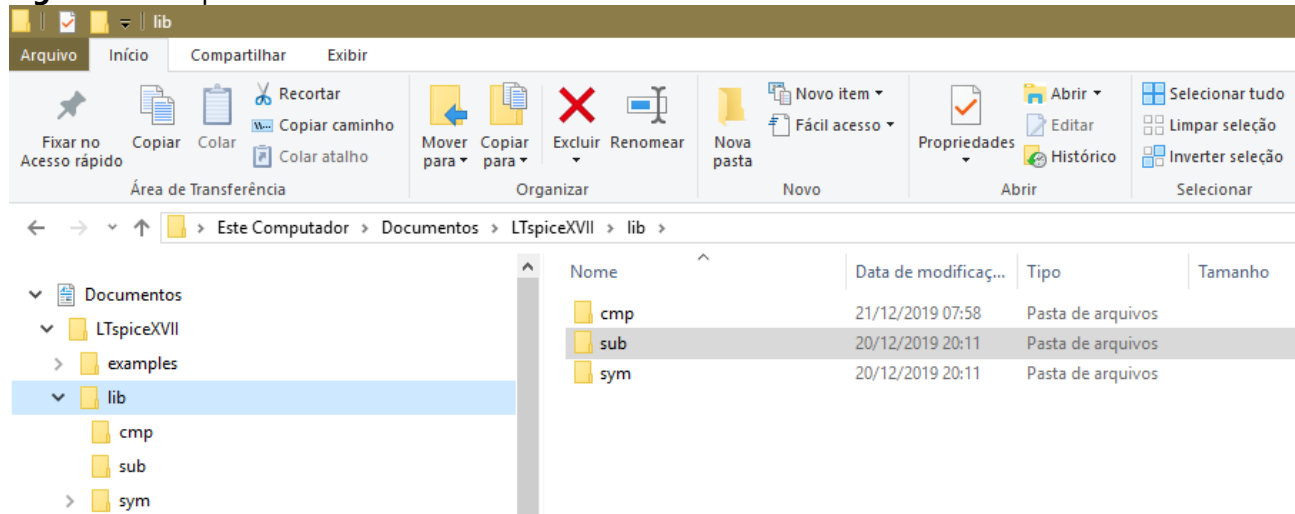
Figura 7 – Subpasta de símbolos.



Outros componentes podem ser adicionados no LTspice, mas, nem todos estão inclusos nos arquivos 'standard.?', na subpasta [cmp]. Então, podemos adicionar esses componentes na pasta [sub] e seus símbolos na pasta [sym]. Normalmente são nomeados com a própria referência do

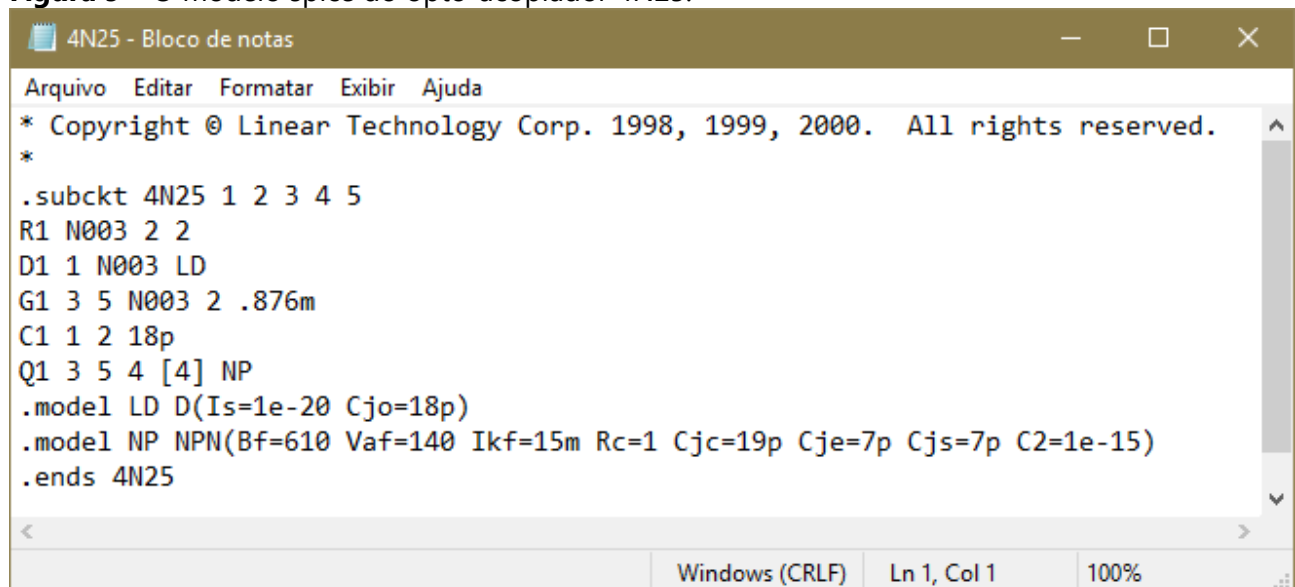
componente, como por exemplo, o opto-acoplador 4N25, cujo nome do arquivo é 4N25.sub ou 4N25.lib. Seu símbolo seria o arquivo 4N25.asy, encontrado na subpasta [sym][Optos].

Figura 8 – Subpasta subcircuitos com extensões .sub ou .lib.



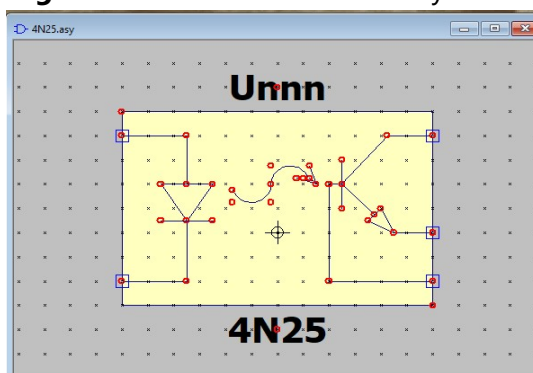
A extensão *.sub ou *.lib, são subcircuitos, isto é, são modelos spice em arquivos textos, que podem ser editados com o Bloco de Notas do Windows, e são disponibilizados pelos fabricantes de componentes. Um modelo spice é mostrado a seguir.

Figura 9 – O modelo spice do opto-acoplador 4N25.



O arquivo do símbolo do 4N25, é o 4N25.asy, e pode ser editado dentro do LTspice.]

Figura 10 – Símbolo do 4N25.asy.



Bem, essa é a organização da biblioteca de componentes do LTspice. Como foi visto é possível expandi-la seguindo a estrutura e disposições mostradas. Nos próximos artigos seguiremos com novas análises.