

# Prototipagem e Montagem de Placa de Circuito Impresso

Aula 07 - Roteiro Pr tico n  04 -

Desenvolvimento da PCI: ARES (Design PCI)

# Apresentação

Esta aula é uma continuação da construção do projeto da PCI utilizando as ferramentas do software Proteus. O projeto do esquemático do circuito foi desenvolvido nas aulas anteriores com o uso da ferramenta ISIS Proteus.

Nesta aula, você irá desenvolver o design na Placa de Circuito Impresso. As etapas de posicionamento dos componentes, criação das trilhas e do roteamento, inserção da malha de terra, visualização final do desenho da placa e geração dos arquivos para fabricação serão realizadas nesta aula com o auxílio da ferramenta ARES Proteus.


## Objetivos

Ao final desta aula, você será capaz de:

- Criar o layout da placa de circuito impresso.
- Definir o posicionamento dos componentes e as trilhas do circuito.
- Criar os arquivos para fabricação seguindo formato padrão *Gerber*.

## ARES Proteus: Design PCI

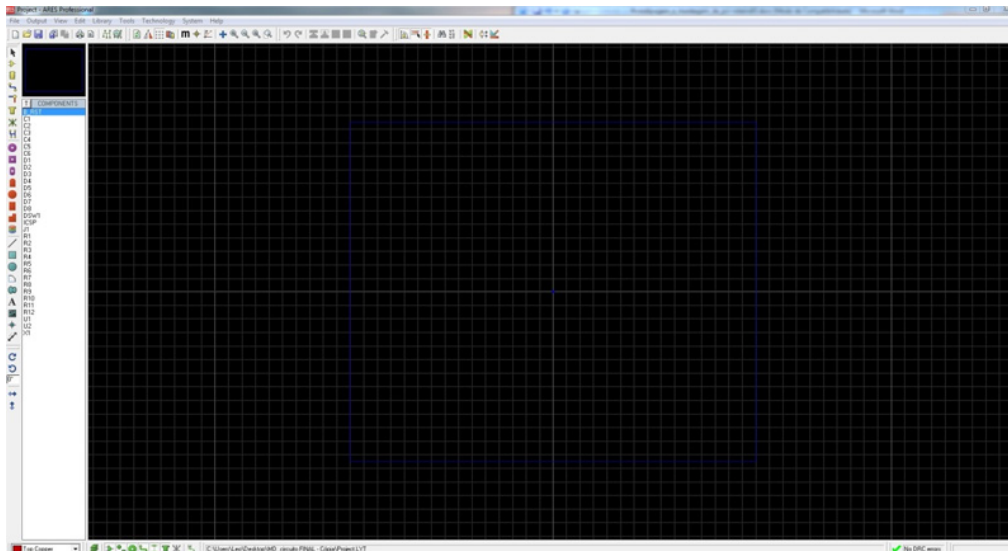
Vamos trabalhar com o ARES. Como já vimos, ele é o responsável por criar o projeto físico da placa, criar as trilhas e definir as regras de roteamento. Para abrir o

ARES, basta clicar (dentro do ISIS) no botão  automaticamente um outro

aplicativo será aberto. Inicialmente, pode aparecer uma janela perguntando qual o tipo de *layout* que devemos escolher, basta selecionar *DEFAULT*. A **Figura 01** mostra a tela inicial do ARES.

Caso o ARES não tenha o encapsulamento de algum componente, será solicitado que você indique qual tipo de encapsulamento deve-se utilizar.



**Figura 01** - Tela inicial do ARES



Ao construir o esquemático do circuito no ISIS usando os componentes é gerado o 'netlist' do circuito, que é responsável por descrever a conectividade de um circuito eletrônico. Ao iniciar o ARES, já se tem boa parte das informações para construção da placa final: em particular, comumente, os encapsulamentos já estão relacionados aos símbolos do esquemático, o que possibilita ao ARES pré-selecionar estes para inserção na área de design do layout da placa. Isso nos trás uma importante distinção a ser feita: a diferença entre componente e encapsulamento.

Um componente no Proteus é uma instância de um modelo eletrônico de dispositivo ou componente eletrônico cujo símbolo/desenho não representa necessariamente o modelo físico. É tanto, que no ISIS, devido ao modelo matemático do circuito gerado, é possível fazer a simulação do mesmo, como vimos anteriormente, podendo-se testar se a construção dele está correta.

Um encapsulamento é a representação física de um componente eletrônico de um fabricante específico e/ou modelo específico, com as mesmas dimensões e características. Esses modelos físicos ficam armazenados na biblioteca do ARES.

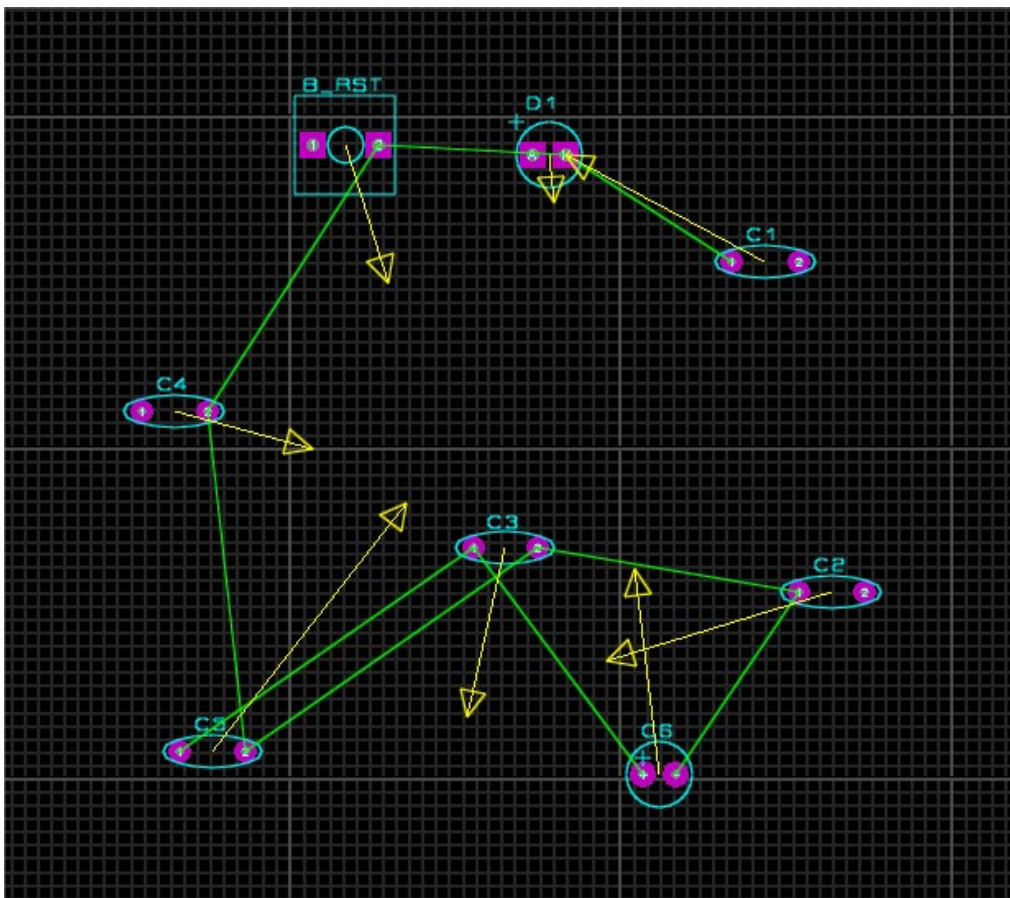
No ARES, ao ativar o 'Componente Mode', , estará disponível os mesmos componentes que foram utilizados no ISIS para montar o circuito, trazendo com eles as informações de conectividade, que poderão ser visualizadas no modo gráfico do ARES por meio de fios verdes à medida que os componentes são inseridos na área de desenho da placa. Por outro lado, no 'Package Mode', , acessa-se apenas às instâncias físicas, não existindo a obrigatoriedade de inserção dos mesmos componentes que foram utilizados no circuito implementado no ISIS, podendo-se fazer, inclusive, outras conexões. Portanto, quando se trabalha com um layout obtido a partir de um esquemático, é mais comum se utilizar exclusivamente o modo componente.

Portanto, observe que vários recursos são parecidos com os do ISIS, mas com uma observação: aqui não podemos criar componente, apenas usá-lo ou criar um novo encapsulamento.

## Adicionando Componente

Veja que no modo de componente aparecem todos os nossos componentes criados no ISIS, são eles que devemos adicionar ao nosso projeto. Para isso, basta clicar no componente, clicar uma vez na tela que o componente irá aparecer e, em seguida, escolhemos o local, posicionamos o componente e clicamos novamente. À medida que os componentes são adicionados vão aparecendo também as conexões entre eles identificadas por um fio verde, aparecem também o chamado vetor de força (em amarelo), o qual serve para indicar um possível local daquele componente, a **Figura 02** mostra alguns componentes adicionados.

**Figura 02** - Adicionando componentes



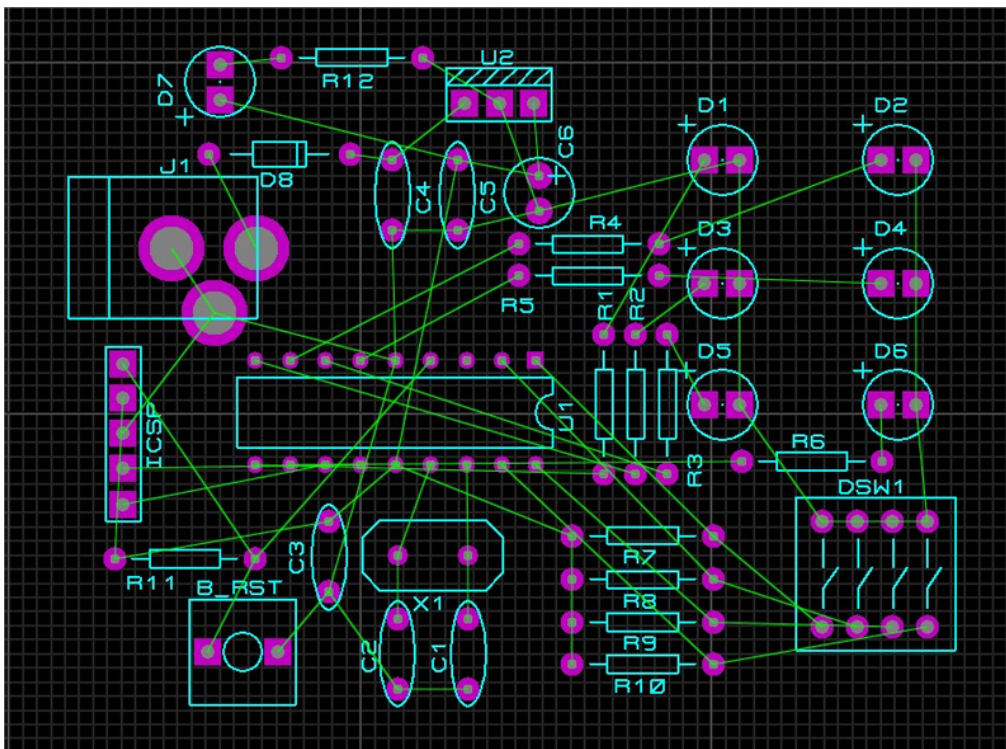
Raramente os projetistas usam o recurso de vetor de força, caso queiram, podem desabilitá-lo. Para isso, basta ir em View => Layers e desmarcar a opção Force Vectors.

Agora, vamos posicionar todos os componentes, inicialmente, vamos separá-lo por módulos, igual como fizemos no ISIS. Para o microcontrolador vamos colocá-lo no centro do circuito, pois esse componente liga-se a todos os outros módulos. A **Figura 03** mostra todos os componentes posicionados para facilitar a identificação dos módulos. Vejam também que a opção vetor de força foi desabilitada.

Se um componente for deletado no ARES, ele somente será excluído da placa, mas irá continuar no projeto. Caso queiram modificar os componentes, devemos fazer isso no ISIS. Os comandos são simples: mover, rotacionar, deletar, dentre outros.


Vamos posicionar nossos componentes da forma como mostra a **Figura 04**. Esse é o padrão de ordem dos componentes escolhido pelo projetista. Caso queira fazer outras versões, fique à vontade.

**Figura 04** - Circuito com os componentes na posição final



Lembre-se de que o local onde colocamos cada componente corresponde à posição física que ele será soldado. Para o posicionamento dos componentes, algumas regras básicas devem ser seguidas: colocar chaves e botões de forma acessível; não colocar gerador de ruído (regulador) próximo ao relógio do circuito; posicionar o gravador na borda externa; o capacitor de filtro do microcontrolador deve ser colocado próximo ao microcontrolador e afastar componentes que trabalham em temperaturas maiores. Todas essas regras foram seguidas para esse nosso projeto, inclusive a posição de cada Led para o semáforo. Caso seja necessário, podem ser criados pontos de fixação para apoiar a placa.



O próximo passo é criar a borda externa, ela será os limites da nossa placa. Para isso, vamos em 2D Graphics BoX Mode , no *layer* que fica do lado esquerdo. Na parte inferior, escolhemos *Board Edge*, conforme mostrado na **Figura 05** e criamos um retângulo cobrindo todos os componentes. Esse retângulo pode ser redimensionado posteriormente. Os outros *Layers* que aparecem são específicos, os mais usados são: TOP (parte de cima da placa), BOTTOM (parte de baixo da placa) e BOARD EDGE (corte externo da placa). Tanto para o TOP quanto para o BOTTOM existem outras opções como COPPER (cobre da placa), SILK (nomenclatura da placa) e MASK (usado para criar a máscara de solda). Algumas dessas opções serão usadas mais adiante nesse curso.

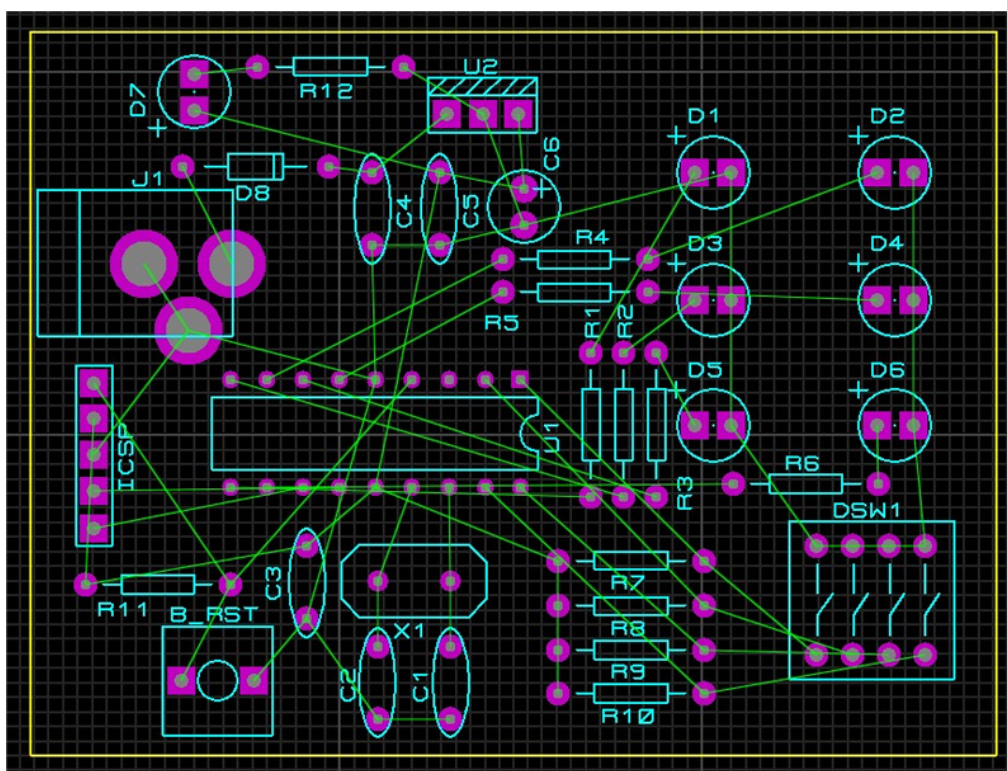
**Figura 05** - *Layer* que podemos trabalhar no ARES



Após adicionar a borda externa da placa, o circuito deve ficar como mostrado na **Figura 06**.




**Figura 06** - Circuito com os componentes e a borda externa

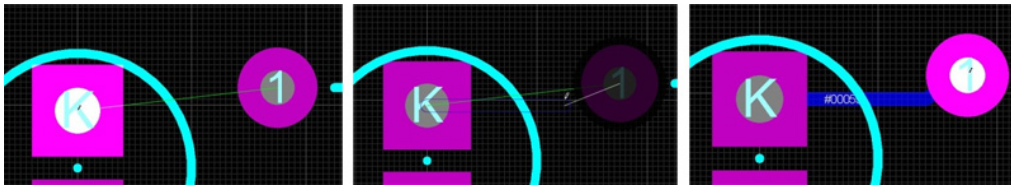


## Criando as trilhas

Vamos criar as trilhas do nosso circuito. A regra básica é que para tensões de alimentação as trilhas devem ser maiores; para sinais, podemos usar o tipo de trilha padrão. Para atingir uma boa experiência em roteamento de circuito deve-se praticar bastante. Nesse curso, criaremos somente as trilhas do tipo padrão e, mesmo assim, faremos as mais próximas, o restante será feito automaticamente. Para aperfeiçoar a técnica, vocês podem praticar futuramente. Um roteamento manual de uma placa de simples face pode demorar dias, dependendo do tamanho e da quantidade de componentes.

Para criar as trilhas, vamos clicar no *Track Mode* , em seguida, clicamos no pino do componente que queremos criar a trilha e arrastar até o próximo pino do outro componente. A **Figura 07** mostra esse passo a passo. Nesse ponto, o *scroll* do mouse ajuda bastante quando queremos aplicar um zoom na tela que estamos trabalhando.

**Figura 07** - Criando trilhas



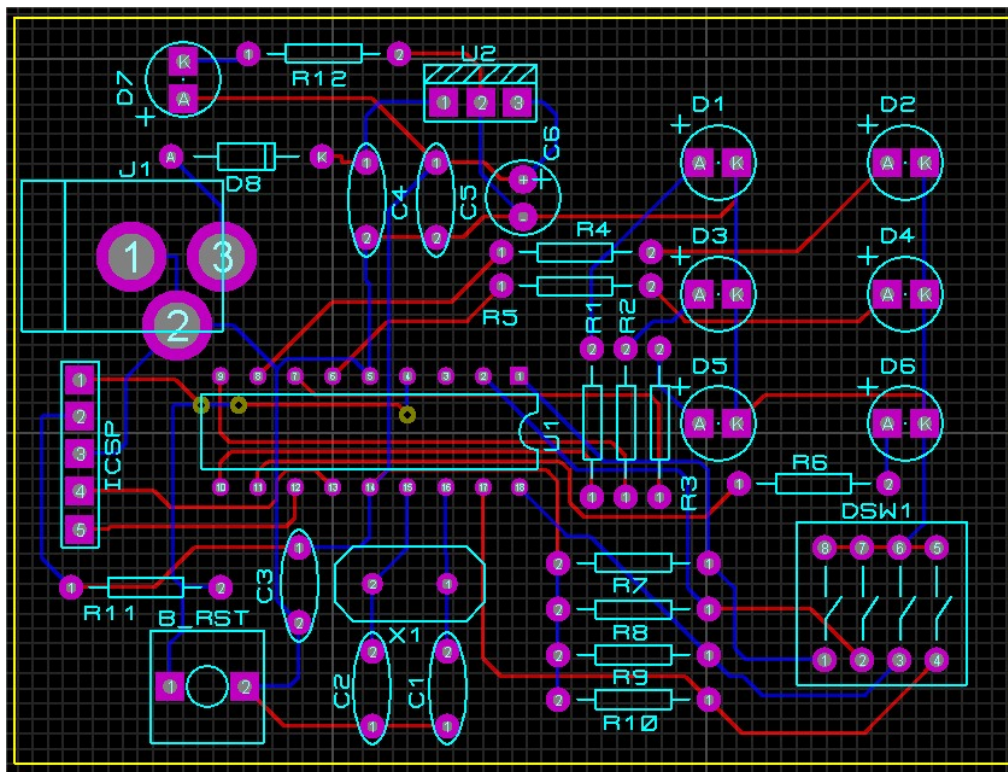
## Dica

Em larga escala, quanto menor for o tamanho da placa, mais barato será sua fabricação. Outros fatores que também influenciam na fabricação de placas é a quantidade de *layer* e o tipo de material que será utilizado.

Conforme explicado anteriormente, vamos criar as trilhas mais próximas, use toda a placa para criar várias trilhas, elas devem ficar na cor azul representando que são trilhas do BOTTOM, pois os componentes estão no TOP e são soldados no BOTTOM. Se a trilha ficar na cor vermelha é porque o *layer* selecionado está incorreto, para desfazer, basta apagar a trilha errada, selecionar o layer correto e continuar.

Para criar o roteamento automático, vamos em Tools => Auto Router... e clicamos em Begin Routing. Observe que o roteamento automático será aplicado, usando trilhas no BOTTOM e no TOP, mas as trilhas que foram criadas manualmente irão permanecer intactas. Ao final, o circuito deve estar parecido com o mostrado na **Figura 08**.

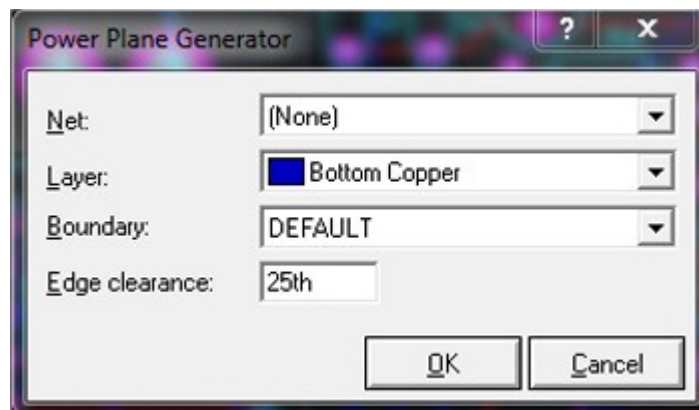
**Figura 08** - Circuito com roteamento automático



Em seguida, vamos aplicar a malha de terra no nosso circuito. Ela serve para economizar material e proteger a placa contra descargas estáticas. A descarga estática geralmente acontece no manuseio da placa, principalmente quando tocamos. Se na placa existir uma maior quantidade de cobre conectado ao seu neutro, essa descarga irá para o sinal neutro da placa, o que causa um dano menor.

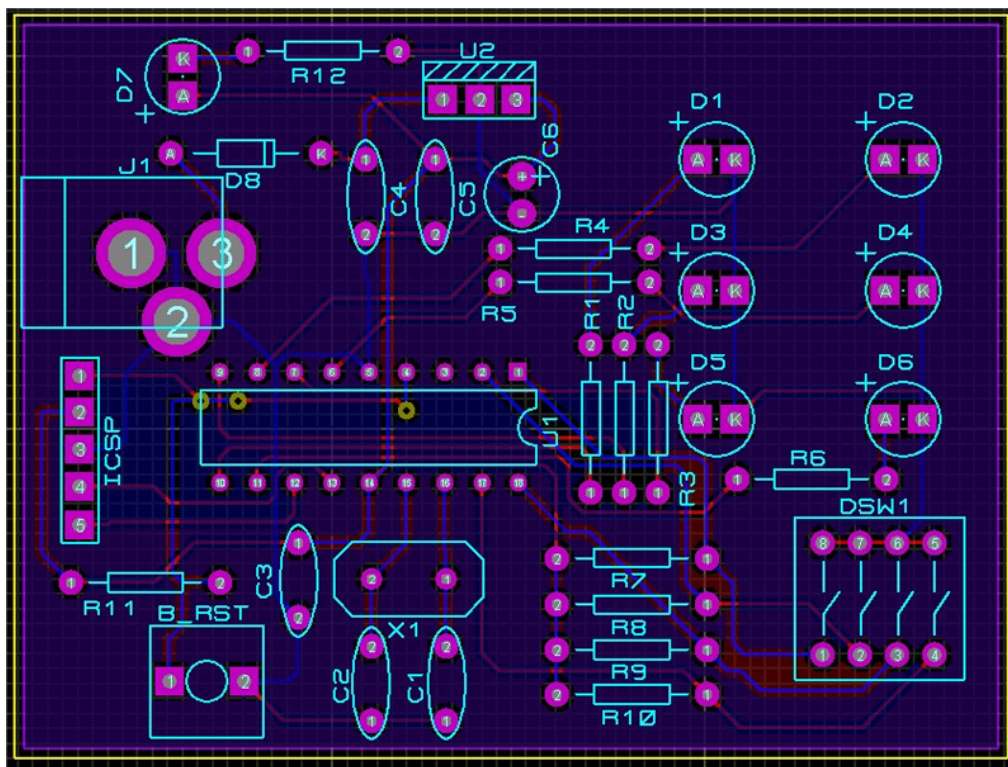
Apesar da nossa placa não ter aterramento, vamos chamar de malha de terra e fazemos a ligação com o sinal neutro ou GND da placa. Para isso, vamos em Tools => Power Plane Generator, uma janela (**Figura 09**) será mostrada.

**Figura 09** - Definição de malha de terra



Na opção Net, escolhemos GND = POWER e, na opção Layer, escolhemos BOTTOM. Se necessário, podemos fazer também no TOP, quando apertamos em OK a malha será aplicada. A **Figura 10** mostra o resultado da placa com TOP e BOTTOM e malha de terra nos dois lados.

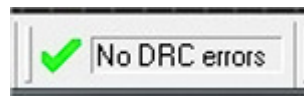
**Figura 10** - Circuito com dois *layers*



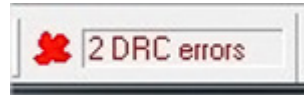
Se o circuito não tiver erro, uma mensagem no canto inferior direito do ARES terá uma indicação igual à **Figura 11**. Se existir erro, a mensagem será igual à **Figura 12**, ela informa a quantidade de erro encontrada e mostra no circuito o local do erro com um círculo vermelho.



**Figura 11** - Circuito sem erros



**Figura 12** - Circuito com erros

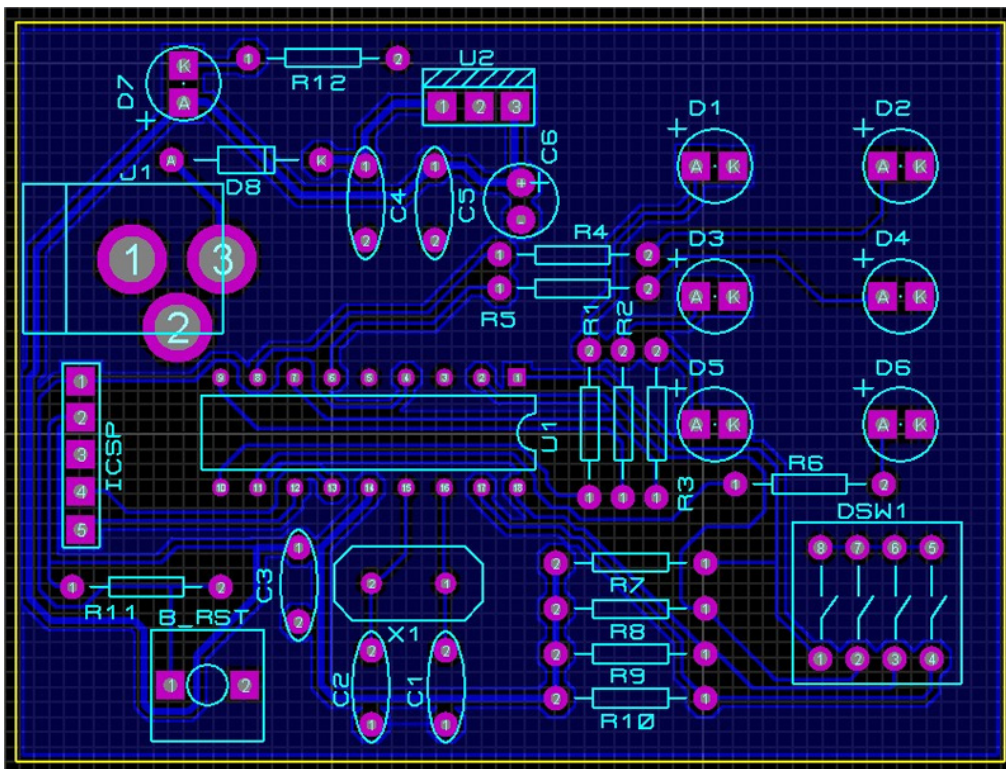


## Circuito final da placa com as trilhas

Agora, o circuito está finalizado, o desafio é montar o mesmo circuito usando o roteamento somente no layer do BOTTOM. A **Figura 13** mostra o circuito apenas com o BOTTOM, compare-o com a **Figura 08** e veja a diferença. O circuito da **Figura 13** é o que será usado a partir de agora.

Observe que as trilhas de alimentação tem uma espessura maior do que as trilhas de sinal, conforme já foi explicado nesta aula.

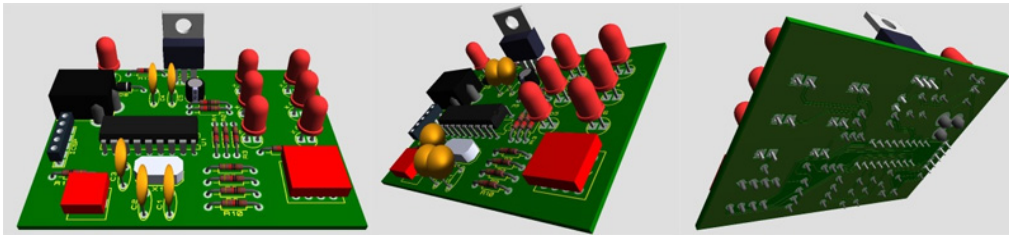
**Figura 13** - Circuito com um *layer*



## Imagem em 3D

O ARES do Proteus tem um recurso muito interessante, ele permite visualizar em 3D como será a placa. Para isso, vamos em Output => 3D Visualization, basta clicar na tela e começar a rodar com o mouse que a placa será movimentada. A **Figura 14** mostra a movimentação da placa em 3D. Para sair da visualização, basta clicar no botão ESC.

**Figura 14** - Visualização 3D da placa de circuito impresso com apenas um *layer*

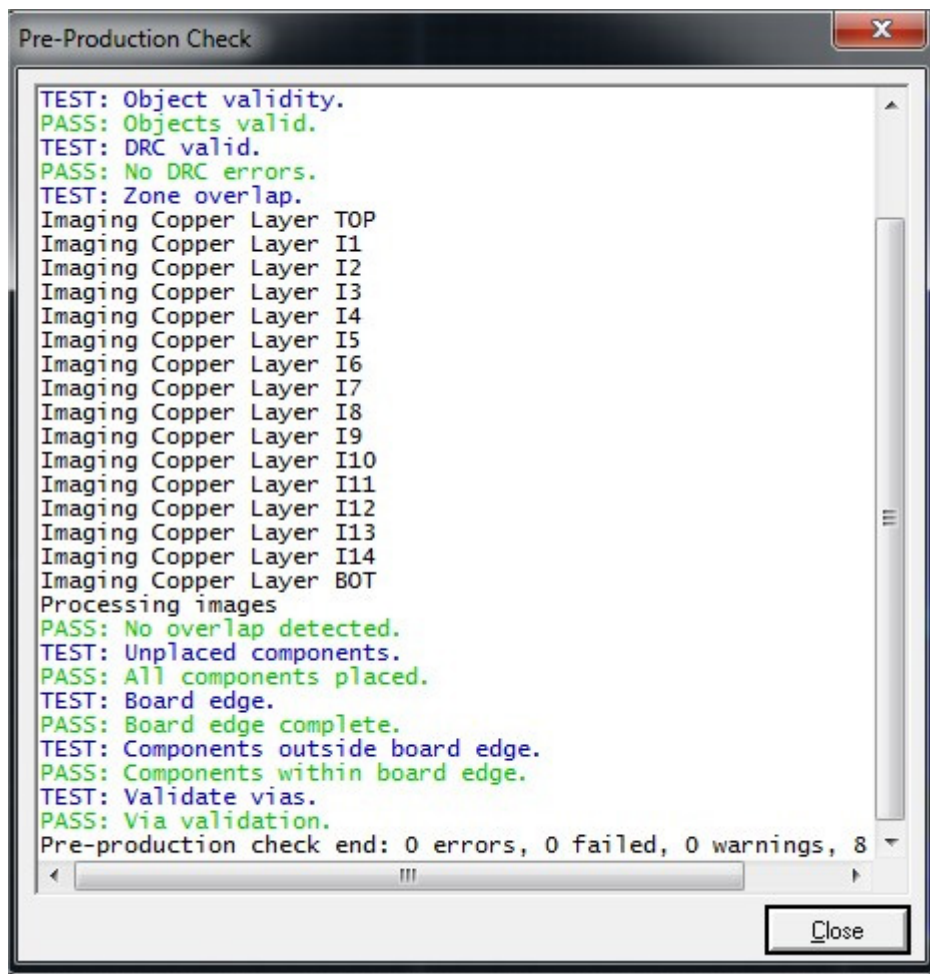


## Criando arquivos *gerber*

O último passo, nesta aula, será gerar os arquivos *gerber*. Esses arquivos são necessários para a produção da placa, seja ela para fins comerciais ou de prototipagem. Como nossa placa possui somente o *layer* do BOTTOM e os componentes estão no *layer* do TOP, vamos gerar somente os arquivos necessários para esse fim.

Para gerar esses arquivos, vamos em Output → Gerber/Excellon Output... uma janela irá aparecer pedindo para buscar por erros, desse modo, podemos clicar em Yes. Após a checagem, aparecerá um relatório, se não tiver nenhum erro, podemos continuar. A **Figura 15** mostra a janela de relatório da checagem por erros.

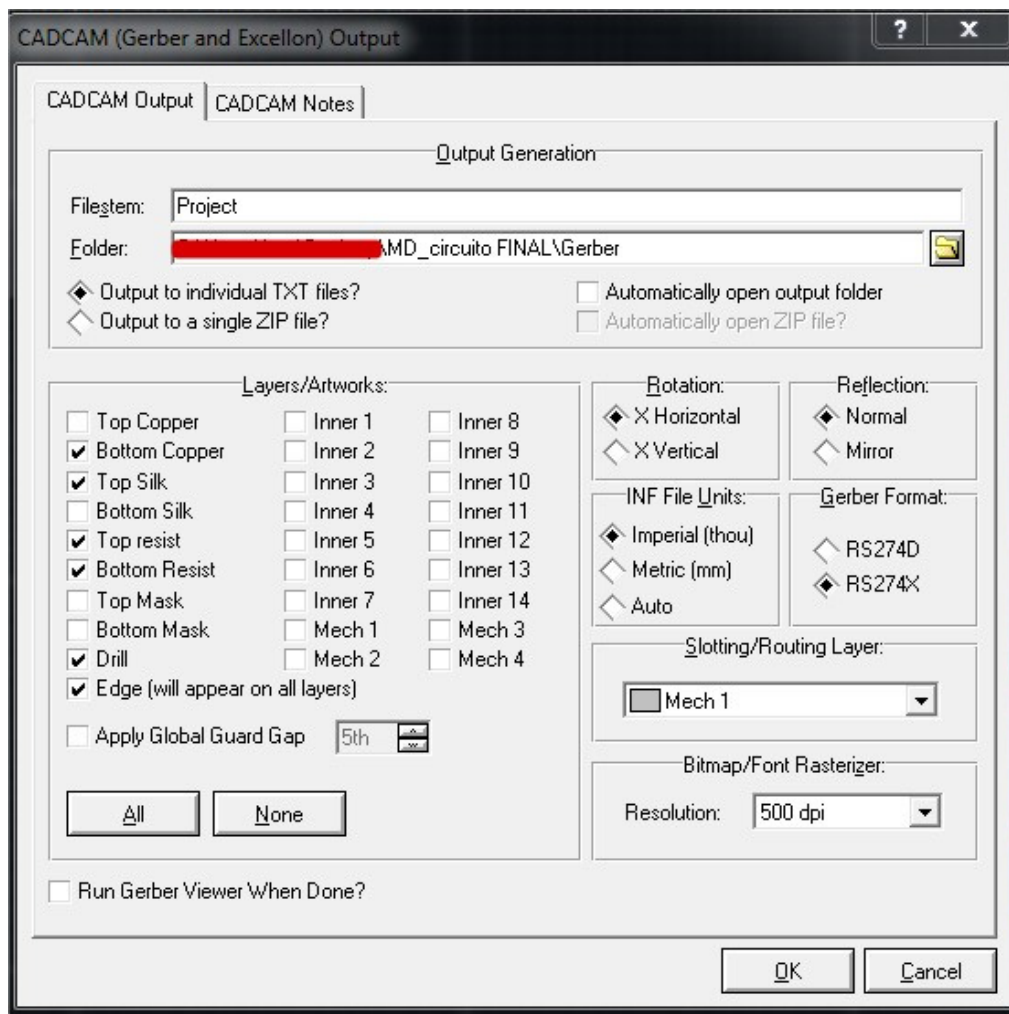
**Figura 15** - Checagem por erros



Quando clicamos em Close, a janela de configuração do *gerber* será aberta. Em *Layers/Artworks*, habilitamos: Bottom Copper, Top Silk, Bottom Resist, Drill e Edge. Se nossa placa fosse de dupla face, a configuração seria diferente. As outras configurações da janela não são modificadas, apenas observe que, em Gerber Format, o padrão selecionado é RS274X, esse é o padrão utilizado atualmente por grande parte das empresas de criação de placas. Por fim, em Folder, escolhemos o local de destino dos arquivos *gerber*. A **Figura 16** mostra a configuração do *gerber*.



**Figura 16** - Configuração do *gerber*



Após a criação do arquivo *gerber*, finalizamos o nosso projeto no Proteus. Esses arquivos são necessários para as próximas aulas.

Lembre-se, nesta aula, foi apresentado o básico para usar o Proteus, mas ele possui muitos outros recursos que podem ser explorados futuramente.

Enfim, o projeto da Placa de Circuito Impresso foi finalizado. O projeto da PCB iniciou com a criação do esquemático do circuito nas aulas anteriores, onde foi possível inserir todos os componentes, fazer as ligações e simular. Nesta aula, foram realizadas todas as etapas de design da placa, incluindo posicionamento dos componentes, criação das trilhas, criação da malha de terra e exportação dos arquivos Gerber. A ferramenta ARES do Proteus foi utilizada para projetar o design da placa. Contudo, independente do software, as etapas de design de PCB seguem a mesma sequência realizada nesta aula.