



KiCad Avanzado

Contenido:

- Procedimientos y conceptos avanzados sobre KiCad.

Autores: Diego Brengi - Noelia Scotti - Diego Alamon

Profundizando editor de esquemáticos



KiCad Trabajo grupal



SCH

Como cada hoja del esquemático se almacena en un archivo de texto diferente, podemos desarrollar el esquemático entre varias personas sin conflictos. Además podemos repartir otras tareas como:

- Diseño de símbolos.
- Manejo de bibliotecas.
- Lista de materiales.
- Diagrama jerárquico.

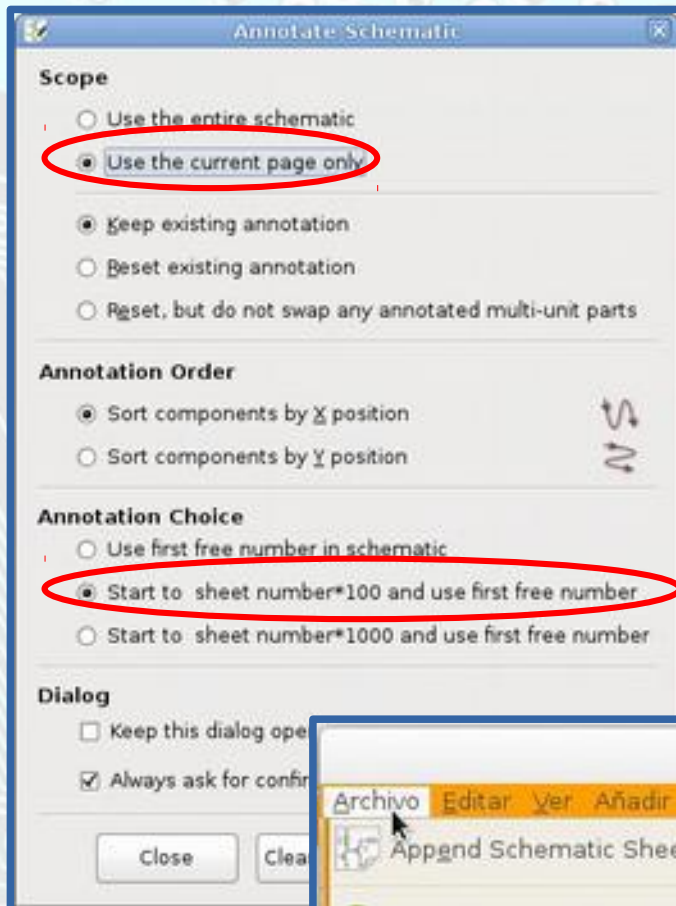
En KiCad **podemos trabajar en grupo** para realizar ciertas fases del desarrollo.

PCB

El PCB es un único archivo, por lo tanto es imposible actualmente trabajar en simultáneo entre varias personas. Pero si es posible distribuir las siguientes tareas:

- Diseño de stack-up.
- Cálculos de impedancia.
- Diseño de footprints.
- Diseño de modelos 3D.
- Ruteo de PCB.

KiCad Trabajo grupal

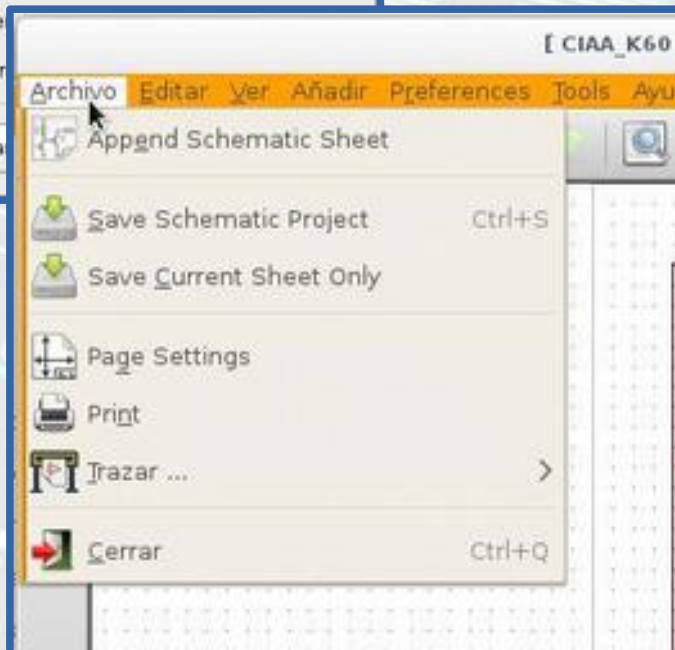


Claves para el trabajo grupal

- Diseño con múltiples hojas (jerárquico o plano).
- Repartir las tareas desde el principio del proyecto.
- Designar un coordinador que gestione librerías, conflictos y hojas complicadas como las de una CPU.
- Reglas de diseño claras desde el principio (tamaños de textos, identificación de pares diferenciales, ubicación de pines jerárquicos, campos de información de los componentes, etc.)

Cuidados que hay que tener

- No agregar hojas a mitad de proyecto, si fuera necesario, lo debe hacer el coordinador teniendo cuidado de que no se modifique la numeración de las hojas actuales.
- Guardar, editar y subir al repositorio solamente nuestros archivos. Si editamos involuntariamente archivos del compañero, deben ser reemplazados por la versión que esté en el repositorio remoto.
- Numerar los componentes con el esquema number*100.
- No renumerar nunca el proyecto completo.



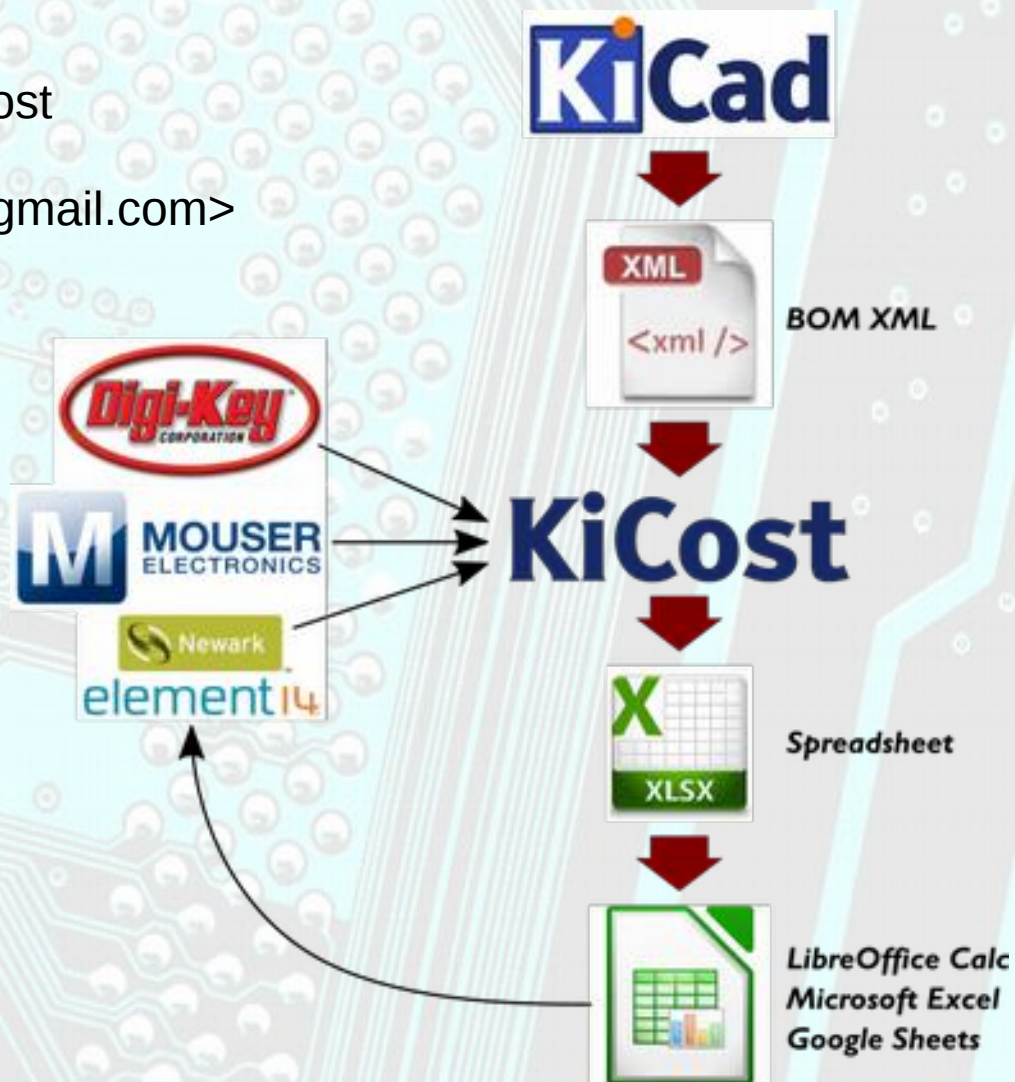
KiCost – Cálculo de costos

KiCost es un script Python para calcular costos de los componentes de diseño hecho en KiCad.

Licencia: MIT
Código: <https://github.com/xesscorp/KiCost>
Docum.: <https://kicost.readthedocs.io>
Autor: Dave Vandembout <devbisme@gmail.com>

Funcionamiento básico:

- 1) Ingresar en el esquemático los campos especiales de información.
- 2) Generar el netlist básico (xml).
- 3) Correr el script con el xml como entrada.
- 4) KiCost consultará las bases de datos de los proveedores y generará una planilla de cálculo.



KiCost – Cálculo de costos

Campos del esquemático que utiliza:

Desc: Descripción larga

Manf: Fabricante

Manf#: Código del fabricante

Digikey#: Código digikey

Mouser#: Código Mouser

kicost:pricing: Colocar precios propios con formato
1:\$1.50; 10:\$1.00; 25:\$0.90; 100:\$0.75

Digikey#

Manf

Manf#

</

KiCost – Cálculo de costos

Ejercicio2cese\$ **kicost -i ej2.xml**

No 'newark#' or 'manf#' field: cannot lookup part ['F1', 'F2'] at newark

No 'digikey#' or 'manf#' field: cannot lookup part ['F1', 'F2'] at digikey

No 'mouser#' or 'manf#' field: cannot lookup part ['F1', 'F2'] at mouser

Part ['R1', 'R2'] not found at newark

Progress: 100%

Exception TypeError: TypeError("'NoneType' object is not callable",) in ignored

Ejercicio2cese\$

| 3 | A | B | C | D | E | F | G | H | I | M | S | Y |
|----|------------------|---------------------|--------------|-----------|----------|-------|-----|-------------|---------|----------|---------|--------|
| 1 | | | | | | | | Board Qty: | 10 | | | |
| 2 | | | | | | | | Total Cost: | \$68,53 | \$72,93 | \$75,88 | \$0,00 |
| 3 | | | | | | | | Unit Cost: | \$6,85 | | | |
| 4 | | | | | | | | | | | | |
| 5 | Global Part Info | | | | | | | | | Digi-Key | Mouser | Newark |
| 6 | Refs | Value | Desc | Footprin | Manf | Manf# | Qty | Unit\$ | Ext\$ | Ext\$ | Ext\$ | Ext\$ |
| 7 | XA1 | Conn_Poncho2P_2x_20 | ej2:Conn | Harwin | M50-350 | | 10 | \$3,17 | \$31,70 | \$34,22 | \$31,70 | |
| 8 | J2 | CAN | ej2:CON | AMP | 640456-3 | | 10 | \$0,12 | \$1,16 | \$1,19 | \$1,16 | |
| 9 | R1,R2 | 60 | Resistors | Vishay Da | CRCW120 | | 20 | \$0,08 | \$1,67 | \$1,67 | \$9,02 | |
| 10 | U1 | SP3232ECN | | Exar | SP3232E | | 10 | \$0,91 | \$9,07 | \$9,36 | \$9,07 | |
| 11 | U2 | AMIS42665TJAA1RG | ej2:SOIC | ON | AMIS426 | | 10 | \$0,97 | \$9,67 | \$10,64 | \$9,67 | |
| 12 | F1,F2 | FIDUCIAL | ej2:Fiducial | 1mm | | | 20 | \$0,00 | \$0,00 | | | |
| 13 | J1 | DB9-F | ej2:DB9 | AMP | 1734354 | | 10 | \$1,19 | \$11,90 | \$12,44 | \$11,90 | |
| 14 | C1-C6 | 100nF | Capacitor | Yageo | CC1206K | | 60 | \$0,06 | \$3,36 | \$3,41 | \$3,36 | |
| 15 | | | | | | | | | | | | |

Ver el instructivo: <https://www.youtube.com/watch?v=AeccxROpDfY>

Ventajas:

- Considera precios según las cantidades.
- Marca el proveedor de menor precio.
- Informa stock existente.
- Permite armar la lista para hacer directamente el pedido.
- Permite incluir otros campos arbitrarios a la planilla de reporte.

Desventajas:

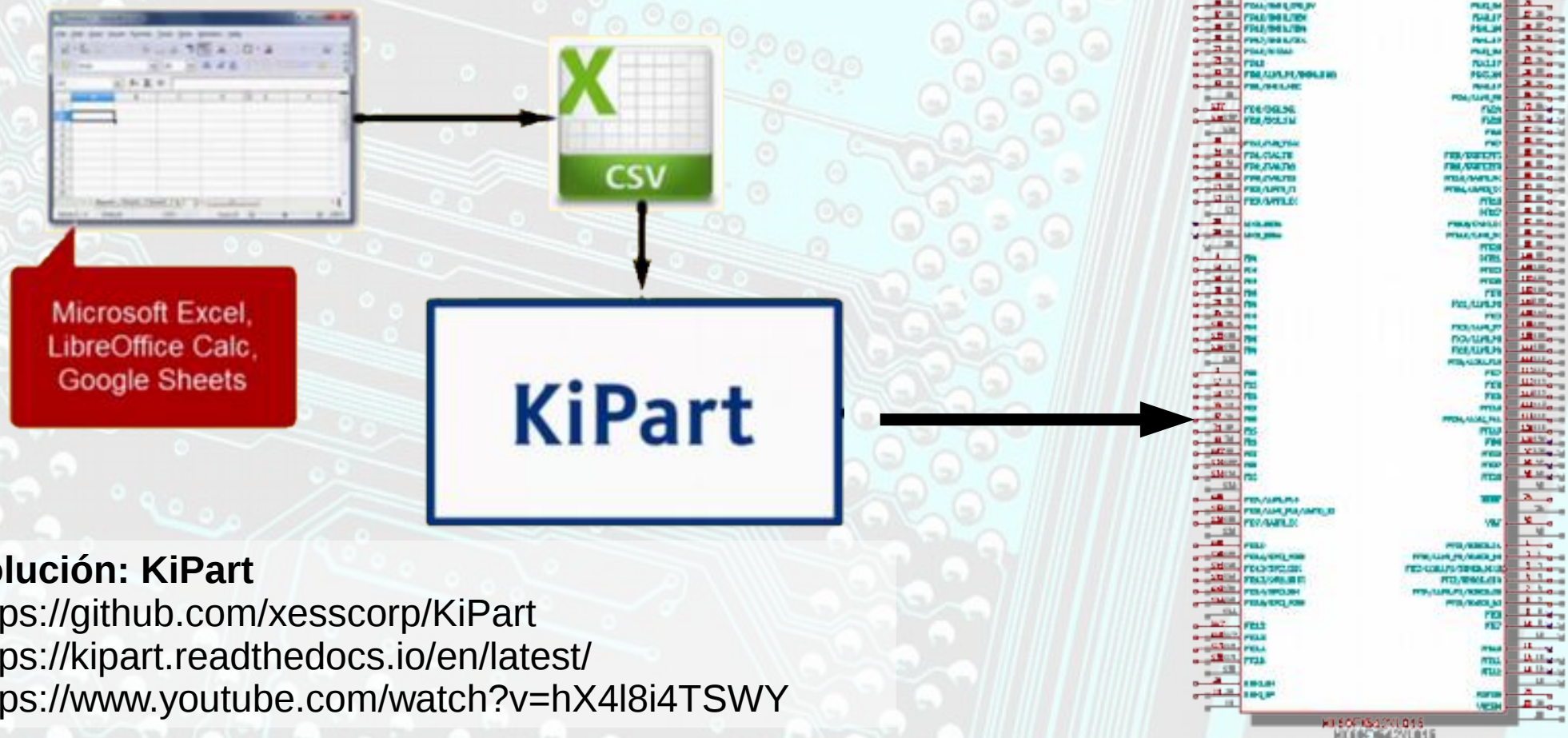
- No siempre es eficiente en encontrar los proveedores.
- Necesita exacta concordancia con los campos de descripción, valor y footprint para agrupar componentes.
- No permite seleccionar los nombre de los campos a considerar.
- Parece trabajar mejor con Digikey.

Ver el instructivo: <https://www.youtube.com/watch?v=AeccxROpDfY>

KiPart para componentes de muchos pines

Problemática

- Crear gráficamente componentes con una gran cantidad de pines.
- Reedición de componentes trabajosa.
- Pines con muchas funciones (textos largos en el componente).
- Se quiere mostrar solo las funciones usadas para ese pin.



Solución: KiPart

<https://github.com/xesscorp/KiPart>

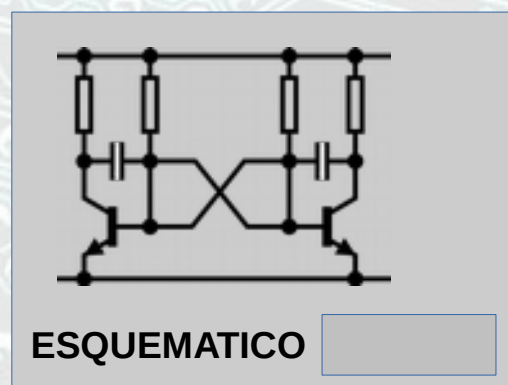
<https://kipart.readthedocs.io/en/latest/>

<https://www.youtube.com/watch?v=hX4l8i4TSWY>

KiField para editar, reorganizar o corregir campos del esquemático

Problemática

- En diseños de muchos componentes, resulta complejo llevar en forma ordenada y consistente los campos adicionales de información como por ejemplo la descripción larga de los componentes o los código de producto.
- En KiCad es muy laborioso reorganizar o borrar los campos de información de los componentes, y requiere recorrer uno por uno los componentes del esquemático.



KiField extrae los campos a una planilla

KiField inserta los campos desde una planilla



Se editan los campos cómodamente en una planilla de cálculo.

Solución: KiField

<https://github.com/xesscorp/kifield>

<https://kifield.readthedocs.io/en/latest/>

Tip: KiField No esta internacionalizado, tiene problemas al manejar acentos y comillas dobles.

Profundizando editor de footprints



Opciones avanzadas del editor de módulos

En el siguiente video se ejemplifica como fácilmente se puede:

- Realizar dibujos y editarlos.
- Importar un archivo DXF.
- Copiar, alinear y distribuir pads.
- Enumerar pads fácilmente.

KiCad module editor demo

<https://vimeo.com/99235812>

The logo for KiCad Module Editor, featuring the word "KiCad" in a blue sans-serif font with a small orange dot above the 'i', followed by "Module Editor" in a blue sans-serif font.

Profundizando Pcbnew

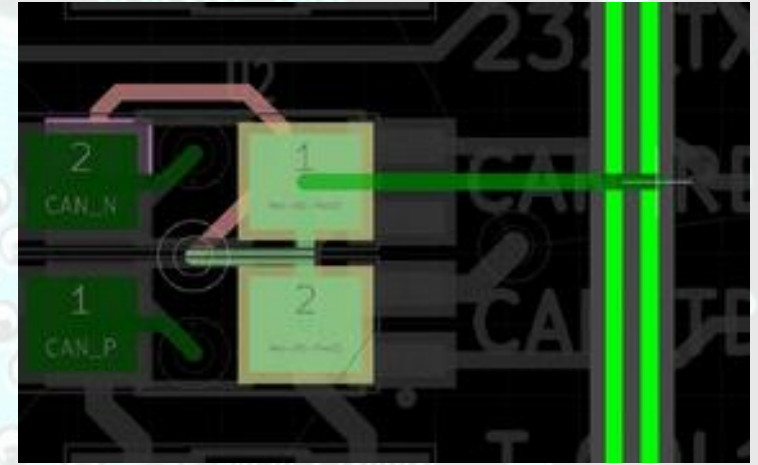
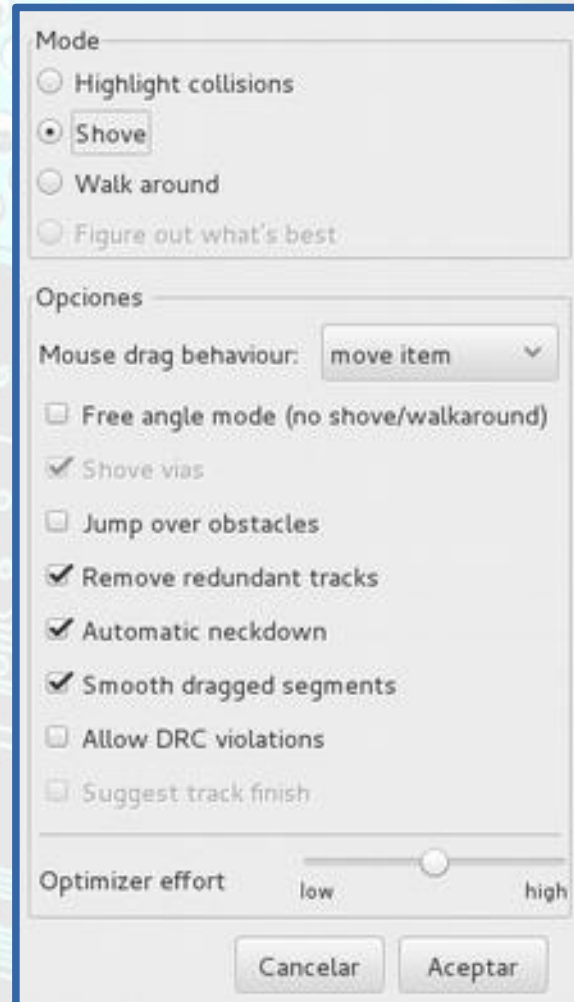


Ruteo interactivo

En el modo GAL, tenemos opciones para configurar el ruteo interactivo.

Básicamente podemos elegir tres modos:

- **Resaltar colisiones**
- **Empujar**
- **Rodear.**



Actualmente KiCad no posee campos para ingresar la información detallada (espesores y materiales) del stack-up. Por eso es conveniente colocar esta información en alguna parte del diseño. Utilizaremos la capa de dibujos para anotar el stackup.



Stack-up de la CIAA-FSL fabricada por Mayer S.A. (4 Layers)

Impedancia Controlada

Si tenemos que seleccionar nosotros el stackup y realizar los cálculos, tendremos que conocer



Tipos de Materiales

- Copper Foils
- Cores / Laminates
- Prepregs

Fabricantes de materiales

- Isola
- Nouya
- Shengyi Technology
- Rogers

Interpretación de hojas de datos de materiales

- Permitividad del dieléctrico.
- Espesor final del material luego del prensado.
- Contenido de resina.
- Tg (glass transition temperature)

Procesos de fabricación

Pueden utilizarse cores/laminates comprados o pueden armarse a partir de prepregs y foils de cobre.

Impedancia Controlada: Ejemplo de cálculo

Ejemplo de cálculo de impedancia para la CIAA-FSL



STACKUP CIAA-FSL

L1 SIGNAL: 1 oz (0,035 mm)
PREPREG: 2x7628AT05: 0,346 mm
L2 GND: 1 oz (0,035 mm)
LAMINATE: 4x7628M: 0,76 mm
L3 PWR: 1 oz (0,035 mm)
PREPREG: 2x7628AT05: 0,346 mm
L4 SIGNAL: 1 oz (0,035 mm)
TOTAL: 1,6 mm (Fabricante Isola)

Ernesto Mayer S.A. pasó el stack-up típico que usaban para PCBs de cuatro capas y en base a eso se hicieron los cálculos con el PCB Calculator.

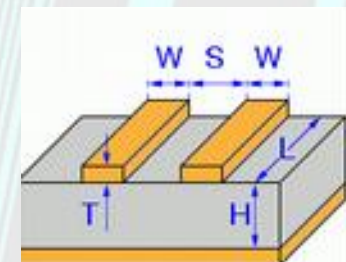


USB $Z_{0\text{diff}} = 90 \text{ ohm} \pm 15\%$
 $W?$, $S?$

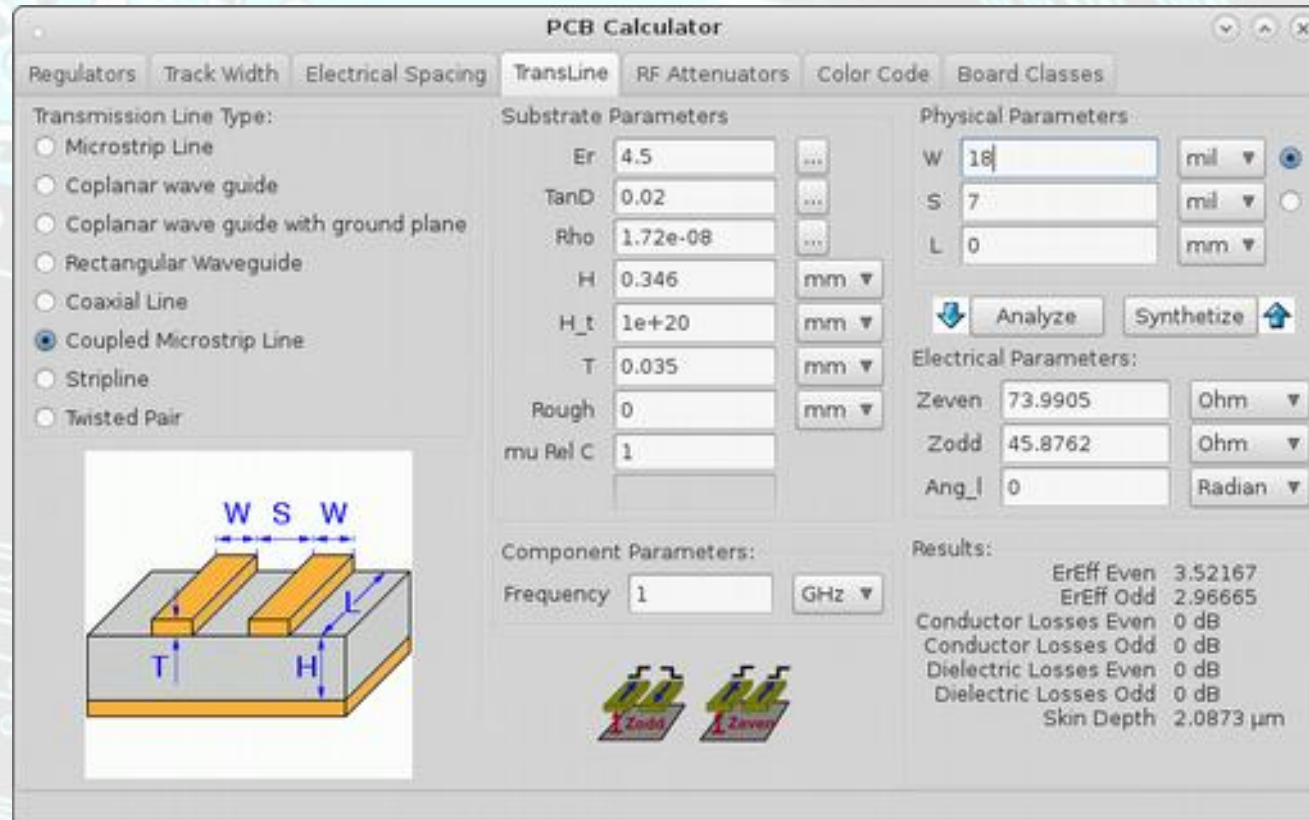
Pares
diferenciales
a calcular



Ethernet $Z_{0\text{diff}} = 100 \text{ ohm}$
 $W?$, $S?$



Impedancia Controlada: PCB Calculator



The screenshot shows the 'PCB Calculator' application window with the 'TransLine' tab selected. The interface is divided into several sections:

- Transmission Line Type:** A list of radio buttons including Microstrip Line, Coplanar wave guide, Coplanar wave guide with ground plane, Rectangular Waveguide, Coaxial Line, **Coupled Microstrip Line** (selected), Stripline, and Twisted Pair.
- Substrate Parameters:** Fields for Er (4.5), TanD (0.02), Rho (1.72e-08), H (0.346 mm), H_t (1e+20 mm), T (0.035 mm), Rough (0 mm), and mu Rel C (1).
- Physical Parameters:** Fields for W (18 mil), S (7 mil), and L (0 mm). Below these are 'Analyze' and 'Synthesize' buttons.
- Electrical Parameters:** Fields for Zeven (73.9905 Ohm), Zodd (45.8762 Ohm), and Ang_l (0 Radian).
- Results:** A table showing calculated values: ErEff Even (3.52167), ErEff Odd (2.96665), Conductor Losses Even (0 dB), Conductor Losses Odd (0 dB), Dielectric Losses Even (0 dB), Dielectric Losses Odd (0 dB), and Skin Depth (2.0873 μm).
- Component Parameters:** A field for Frequency (1 GHz).
- Diagram:** A 3D schematic of a coupled microstrip line with dimensions W, S, W, T, and H labeled.

Herramientas

- **Regulators:** Calculadora de resistencias para reguladores.
- **Track Width:** Calculadora para el ancho de pistas en fc. de la corriente máxima.
- **Electrical Spacing:** Separaciones mínimas para lograr determinada aislación entre pistas.
- **TransLine:** Calculadora de impedancia característica (Z_0) para líneas de Tx.
- **RF-Attenuators:** Cálculo de atenuadores de RF.
- **Color Code:** Código de colores de resistencias.
- **Board Classes:** Clasificación de PCBs según sus dimensiones mínimas.

Impedancia Controlada: Microstrip Diferencial

**Coupled Microstrip Line
= Microstrip Diferencial**

PCB Calculator

Regulators Track Width Electrical Spacing **TransLine** RF Attenuators Color Code Board Classes

Transmission Line Types

- ☐ Coplanar wave guide
- ☐ Coplanar wave guide with ground plane
- ☐ Rectangular Waveguide
- ☐ Coaxial Line
- ☒ **Coupled Microstrip Line**
- ☐ Stripline
- ☐ Twisted Pair

Substrate Parameters

- Er 4.5
- TanD 0.02
- Rho 1.72e-08
- H 0.346 mm
- H_t 1e+20 mm
- T 0.035 mm
- Rough 0 mm
- mu Rel C 1

Physical Parameters

- W 18 mil
- S 7 mil
- L 0 mm

Analyze Synthetize

Electrical Parameters:

- Zeven 73.9905 Ohm
- Zodd 45.8762 Ohm
- Ang_l 0 Radian

Component Parameters:

- Frequency 1 GHz

Results:

- ErEff Even 3.52167
- ErEff Odd 2.96665
- Conductor Losses Even 0 dB
- Conductor Losses Odd 0 dB
- Dielectric Losses Even 0 dB
- Dielectric Losses Odd 0 dB
- Skin Depth 2.0873 μm

TransLine: Qué datos necesitamos para el cálculo de un Microstrip Diferencial?

- Er: Permitividad del dieléctrico. Sacar de la hoja de datos del material.
- H: Altura del dieléctrico.
- T: Espesor de la capa de cobre en la que se ruteará el par diferencial.
- W: Ancho de las pistas del par diferencial. Es lo que queremos calcular.
- S: Distancia entre las pistas del par diferencial. Podemos utilizar el clearance que indique el fabricante.
- Zeven/Zodd: Impedancia en modo par y en modo impar. $Z_{\text{odd}} = Z_{\text{diff}} / 2$. Zeven estimar un valor inicial de 65 ohm si se quiere usar la opción Synthetize.
- Los valores de frecuencia y longitud de la línea “no afectan significativamente” ya que no estamos haciendo un cálculo de pérdidas sino de impedancia.

Impedancia Controlada: Ejemplo de cálculo

Ejemplo de cálculo de impedancia para la CIAA-FSL
Par diferencial USB: $Z_{diff} = 90 \text{ ohm}$

1) Replicar el cálculo realizado en la CIAA-FSL.

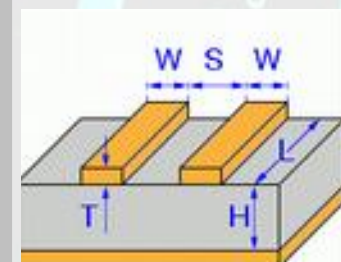
1

Datos a ingresar en la calculadora

- Tipo de línea = Coupled Microstrip Line
- $\epsilon_r = 4,5$
- $H = 0,346 \text{ mm}$
- $T = 0,035 \text{ mm}$
- W = Seleccionar el radio button superior para que al sintetizar se calcule W .
- $S = 7 \text{ mils}$
- $Z_{odd} = Z_{diff} / 2 = 45 \text{ ohm}$
- $Z_{even} = 65 \text{ ohm}$
- Los demás valores dejarlos como vienen por default.

STACKUP CIAA-FSL

L1 SIGNAL: 1 oz (0,035 mm)
PREPREG: 2x7628AT05: 0,346 mm
L2 GND: 1 oz (0,035 mm)
LAMINATE: 4x7628M: 0,76 mm
L3 PWR: 1 oz (0,035 mm)
PREPREG: 2x7628AT05: 0,346 mm
L4 SIGNAL: 1 oz (0,035 mm)
TOTAL: 1,6 mm (Fabricante Isola)

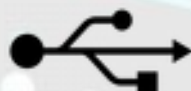


2

Presionar el botón
Synthetize

3

Redondear
resultado
 $W = 18 \text{ mils}$
 $S = 7 \text{ mils}$



4

Presionar el
botón
Analyze

5

Calcular Z_{diff}
final
 $Z_{diff} = 91,75 \text{ ohm}$

90 ohm $\pm 15\%$
Cumple!

Pcbnew: Ruteo de pares diferenciales

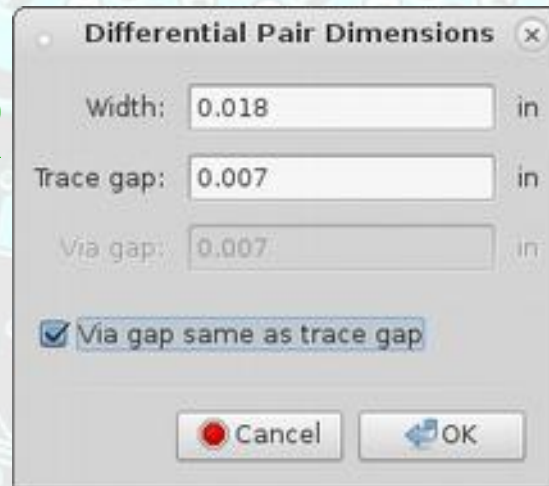
Dimensions →

Differential Pair Dimensions

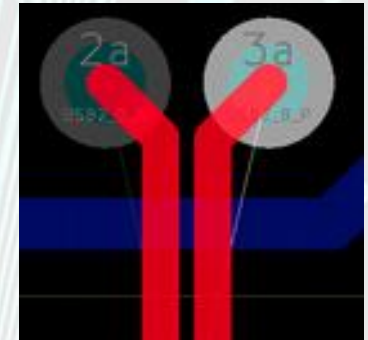
Route → Differential Pair

1 Setear en el SCH los nombres de las nets:
USB_P
USB_N
ETH_TX+
ETH_TX-

2



3



- 1) Verificar que en el esquemático las nets tengan nombres con el sufijo **_P** y **_N** ó los símbolos **+** y **-** para que KiCad las reconozca. Por ejemplo, USB_P y USB_N.
- 2) Antes de rutear cada par, se debe configurar el ancho de las pistas y la separación entre las mismas. Ir a Dimensions → Differential Pair Dimensions.
- 3) Rutear el par diferencial. Route → Differential Pair.

Pcbnew: Ruteo de pares diferenciales

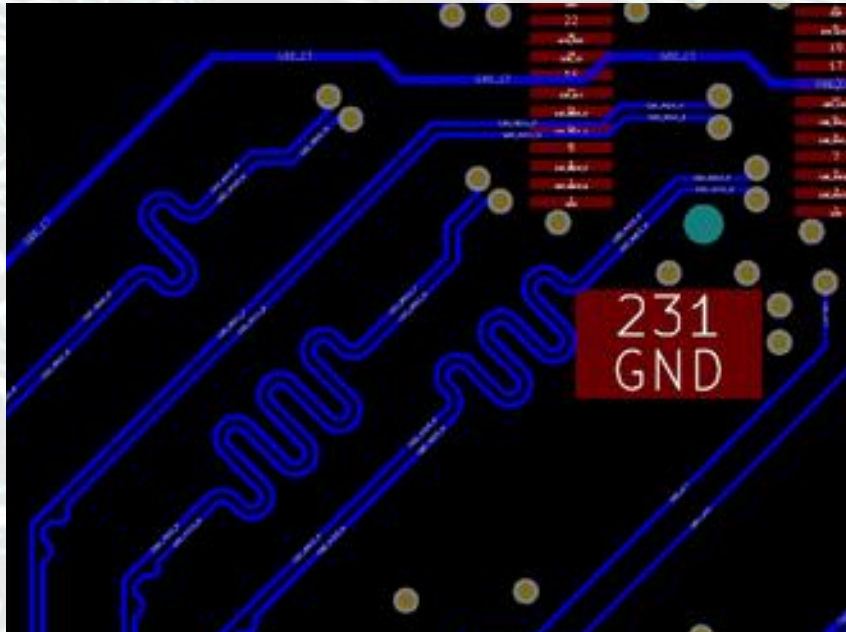
Ecualización de longitud

Luego de trazados los pares diferenciales, debemos ecualizar su longitud de acuerdo a la especificación de la interfaz.

Por ejemplo, para Gigabit Ethernet la máxima diferencia de longitud entre pares es de 30 mils.

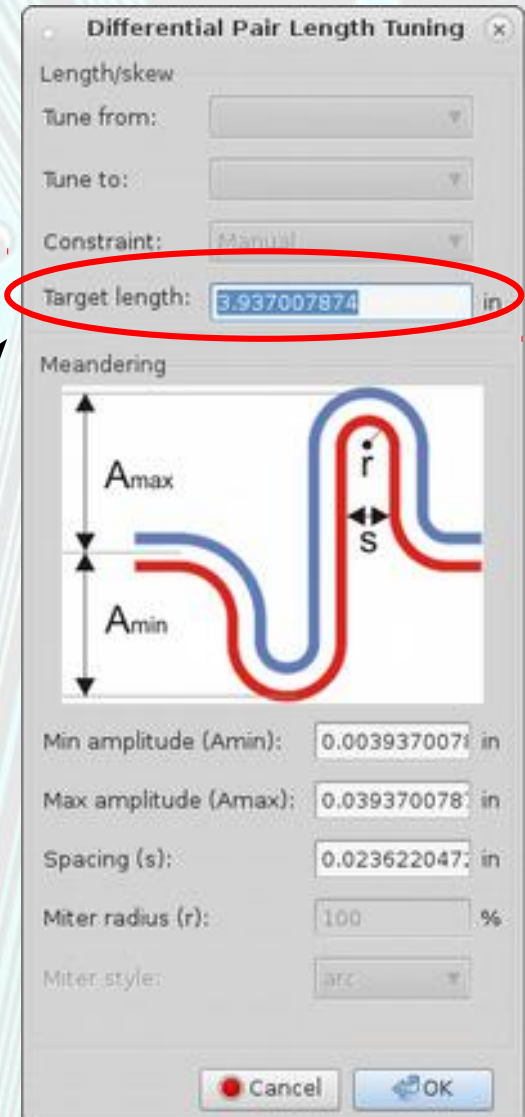
- 1) Ir a Route → Tune Differential Pair Length.
- 2) Clickear sobre el par diferencial y presionar la tecla “L”.
- 3) Configurar la longitud y la forma de las onditas (esto último sólo si fuera necesario).

Route → Tune Differential Pair Length



Target length

Obtener las longitudes de los pares diferenciales a ecualizar e identificar el más largo. Esta longitud será la que utilizemos para ecualizar.



Pcbnew: Ruteo de pares diferenciales

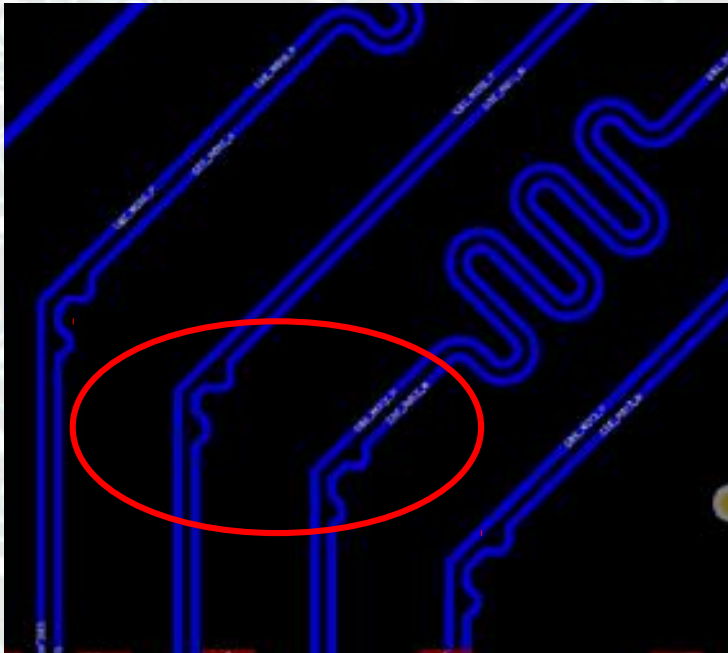
Ecualización de fase

Luego de trazados y ecualizados los pares diferenciales, debemos ecualizar su fase de acuerdo a la especificación de la interfaz.

Por ejemplo, para Gigabit Ethernet la tolerancia máxima de ecualización intra-par es de 5 mils.

- 1) Ir a Route → Tune Differential Pair Skew/Phase.
- 2) Clickear sobre el par diferencial y presionar la tecla “L”.
- 3) Configurar el desvío deseado y la forma de las onditas (esto último sólo si fuera necesario).

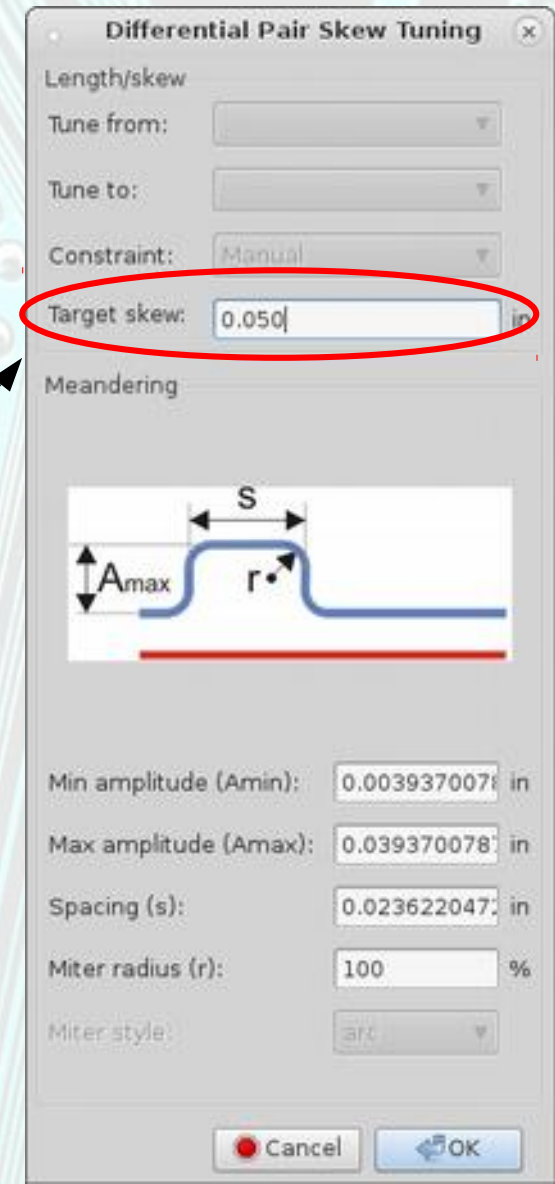
Route → Tune Differential Pair Skew/Phase



Target skew

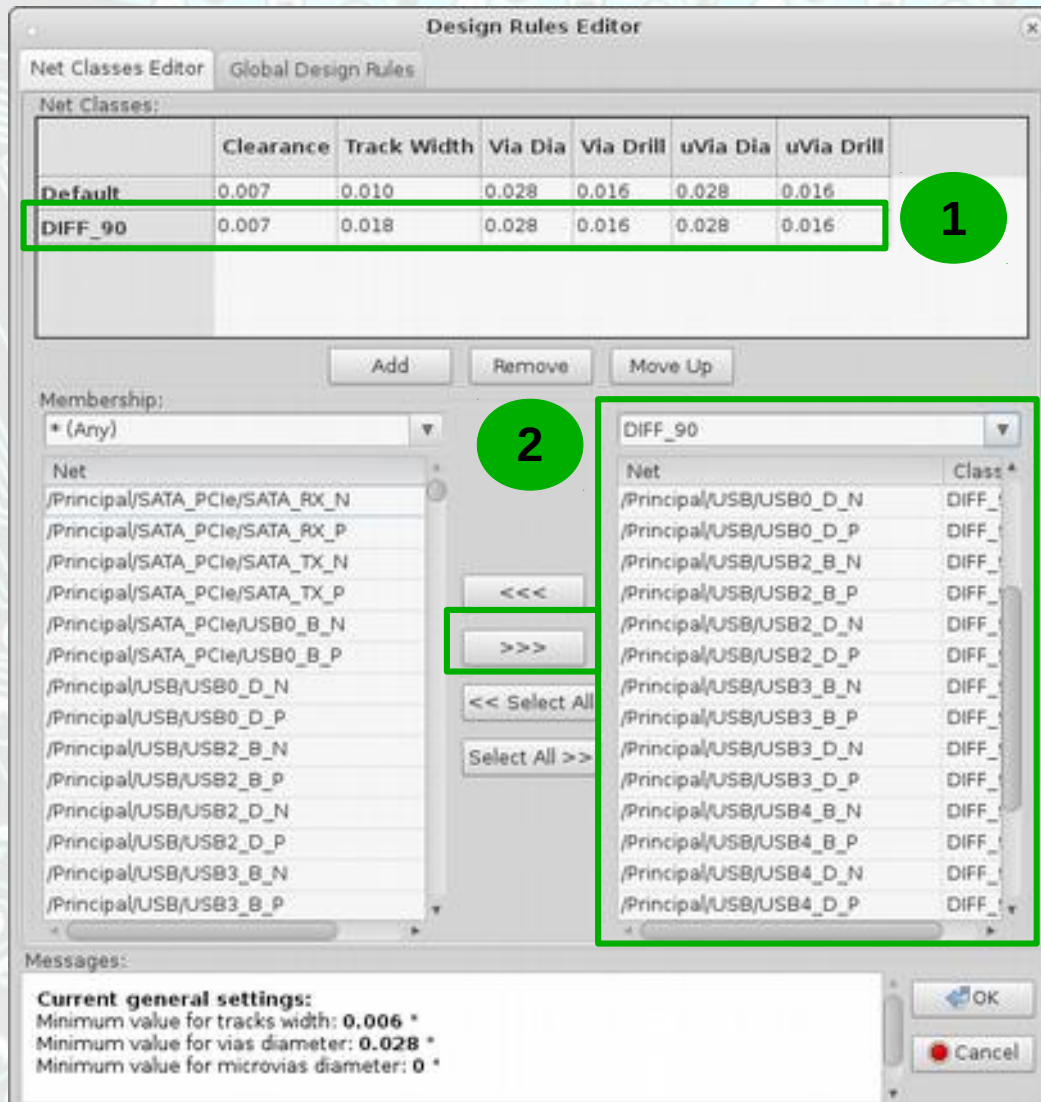
Obtener la diferencia de longitud entre las líneas del par diferencial. Esta diferencia será la que utilizemos para ecualizar la fase.

Tip: Ubicar las onditas en donde las líneas se separan más, por ejemplo en las curvas.



Pcbnew: Ruteo de pares diferenciales IV

Design Rules → Design Rules



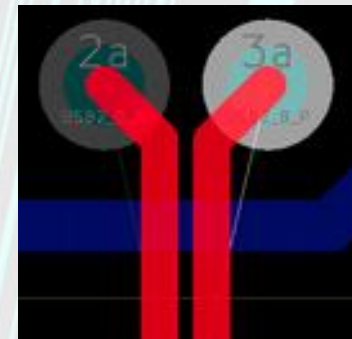
Seteo de reglas específicas para pares diferenciales.

Si queremos que al ejecutar el DRC, sean chequeados los pares diferenciales, hay que crear una Net Class y asignarle las nets correspondientes.

1) Crear la Net Class con los valores de clearance y ancho de pista correspondientes.

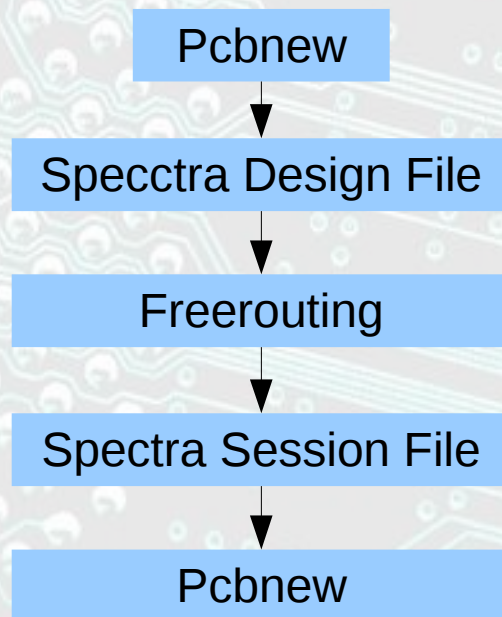
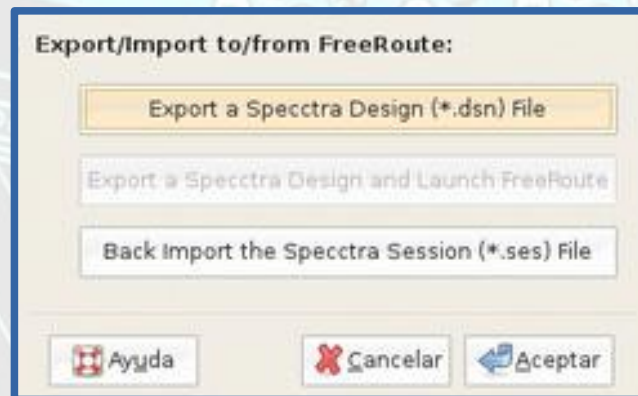
2) Seleccionar la net class del lado derecho y asignarle las nets.

3) Ejecutar el DRC para probar que los pares diferenciales quedaron bien trazados. Tools → DRC.



Auto Router en Kicad

Freerouting es un programa java externo a Kicad que realiza autorouteo.

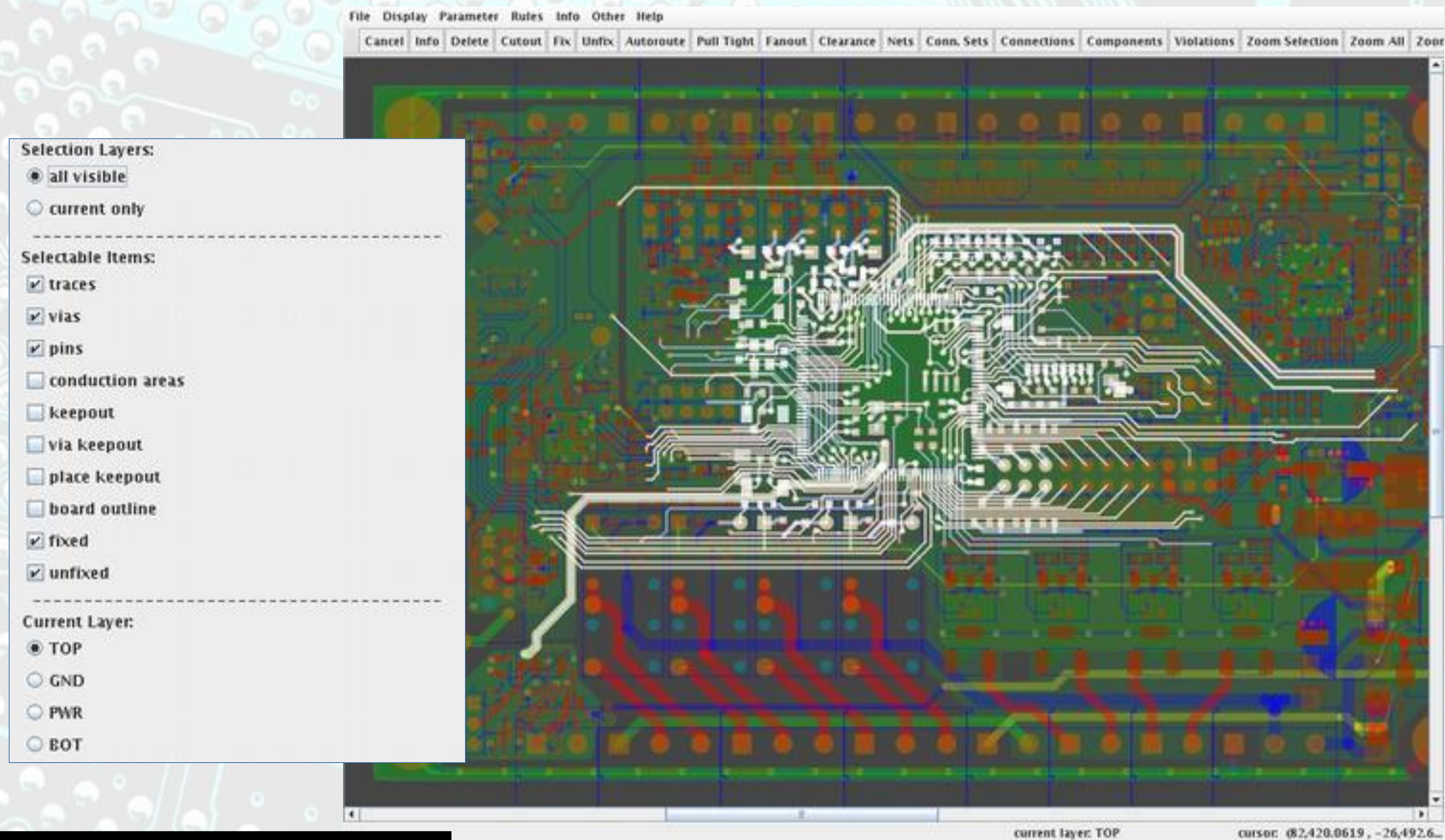


<https://github.com/nikrophT/FreeRouting>

Su autor lo ha publicado con licencia GPL, pero ante un reclamo de la empresa donde trabajaba (esta jubilado actualmente) optó por quitar el código. Sin embargo está disponible ya que se han realizado copias.

Auto router en KiCad

Captura de pantalla de Freerouting



```
$ java -jar FreeRouting.jar
```


Adaptación: Diego Brengi - djavier@ieee.org

Carátula principal:

Foto titulada “Circuit” de Yuri Samoilov bajo licencia CC-BY disponible en <https://www.flickr.com/photos/yusamoilov/14011462899/>

Fondo de la presentación:

Foto titulada “computer motherboard tracks” de Creativity103 bajo licencia CC-BY disponible en: https://www.flickr.com/photos/creative_stock/5228433146/

Las imágenes de clipart se tomaron de: <https://openclipart.org/>

Los demás logos corresponden a proyectos de Software Libre u Open Source. Consultar cada licencia en particular.

Todas las capturas de pantalla fueron realizadas por los autores y están bajo la misma licencia que esta presentación.

El resto de las imágenes se cita la fuente debajo de cada una.

