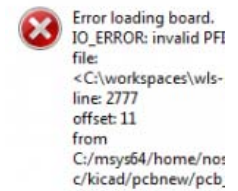


list loading produces “No footprint defined for element” error. What is worse PCB design saved cannot be reopened again as *pcbnew* gives fatal error on board loading.

To fix it I opened .kicad_pcb original file in text editor at line 2777 as error message instructs. The section was pointing out to quartz component as follows:



PCB broken by misspelled footprint

```
(module crystal_hc-49/smd (layer F.Cu) (tedit 53650385) (tstamp 53634F85)
  (at 95.885 58.42)
  (descr "Crystal, HC-49/SMD")
  (tags QUARTZ)
  ...
```

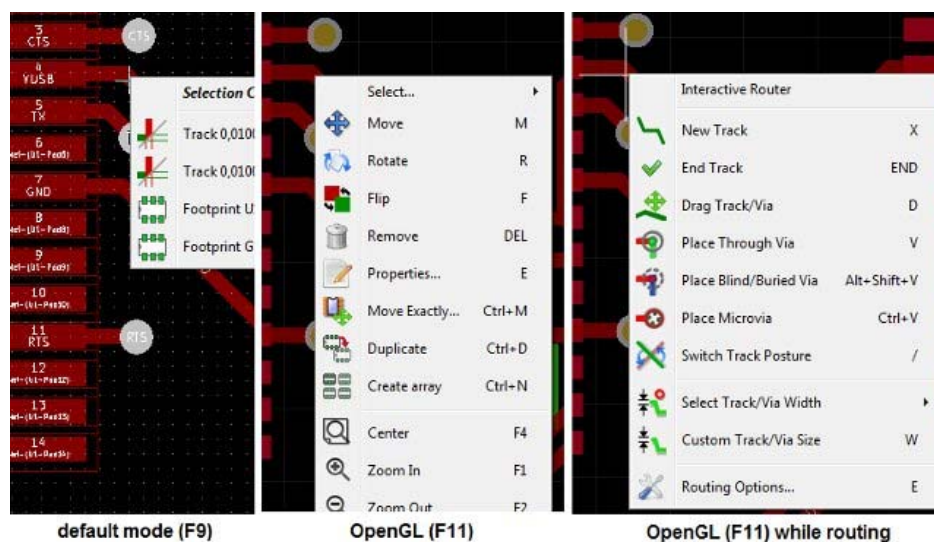
Saving the board deisng misnamed the module as shown below. Added semicolon was probable source of parsing error on board loading.

```
(module smd: (layer F.Cu) (tedit 53650385) (tstamp 53634F85)
```

When I manually fixed module name to “hc-49_smd” it and tried to open again I succeeded. To avoid further problems I have reviewed and renamed all my footprints having slash in the name.

Interactive router shortcuts

Most shortcuts work same way in all modes, like X to start routing, M to move objects or V to place vias. Other interactive router specific were harder to find. I knew some of them from videos yet I could not find ultimate list of all possibilities in documentation. [Chapter 9](#) of pcbnew documentation did not answer my question, neither shortcut list (invoked by Shift+? in *pcbnew*) did as it does not change across different modes. What I obviously overlooked was chapter on context menus that could shed some light and help to understand that **context menu changes on canvas view mode change**. After switching from default mode (F9) to OpenGL (F11) or Cairo (F12) modes, that activates interactive router, right mouse click brings different options and their shortcuts for quick keystrokes. Pay attention that menu changes after you start routing process (pressing X) and gets back when skipped routing (ESC pressed).



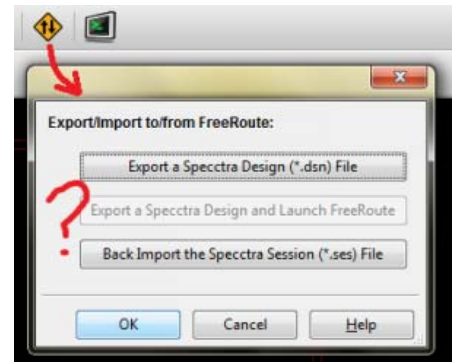
Push and shove can save half of time spent on manual routing. Granted. Couple hours of intense work later and

what I am missing is ability to *interactively drag components* i.e. moving component in a way that keeps routes connected to pads pushing things around. Having partially routed board makes you realizing better components placement; moving them though disconnects traces that must be redrawn... no problem for couple discrete elements, pain in the ass for tens to hundreds connections of single IC. I keep fingers crossed CERN team place it on enhancements list 😞

Freerouter

Fully automated routing is *passé* among professional EEs. For hobby work it is nice accelerator, especially combined with push-and-shove. Imagine this: for sensitive fragments of PCB (like switching DC/DCs or microwave parts) you do the manual routing job and then spawn Freerouting tool to wire the rest of it more carelessly.

The new Kicad still teases with that option but www.freerouting.net is down so Kicad bundle gives you grayed out button instead. Porting my [offline freerouter](#) for from old to new Kicad would be failure as with the advent of JDK8 old JARs stopped working anyway.



Why there is freerouter shortcut in pcbnew then? The answer is in the Kicad [source code](#) — *pcbnew* will enable middle button when it detects `freeroute.jar` in Kicad binary directory. Since freerouter source code was GPLed, mirrored and modified, now it can be still compiled and run as standalone co-working Java application.

To enable Freerouter in Kicad:

1. Download Freerouter from this mirror — <https://github.com/freerouting/freerouting>
2. Extract `freerouting-master.zip/binaries/freerouting.jar`
3. Copy JAR to Kicad bin directory and rename it to `freeroute.jar`
4. Restart Kicad to enable freerouter middle button in pcbnew. Voila!



Freerouter enabled

This entry was posted in [Electronics](#), [Software](#). Bookmark the [permalink](#).

19 Responses to Kicad upgrade: push & shove and freerouting revisited

Roger says:

2015/11/03 at 17:25

Hi Andy,

I'm using KiCad to do a project design and I used your trick to enable Freerouter. The middle button is working but when I try to use the application a Java error appears:

java.lang.Error: illegal character ...

Could you help me please?

Thanks!

[Reply](#)

andy says:

2015/11/04 at 12:54

Roger, without more details I am just guessing. What was the whole error description: is this illegal character on you path? Was there the whole stack trace printed out? Did you try manually running freerouter (opening command line window, going to directory of freerouting.jar

and running “java -jar freerouting.jar”)?

[Reply](#)

Quark66 says:

2016/02/09 at 10:53

Hi, my experience is: I run

java -jar freeroute.jar -de MyScheme.dsn (-de as input param, router open it immediately) everyth. OK. But if I try to use national char set in filename (not good, my fault), than the same error. Example

java -jar freeroute.jar -de MůjVýkres.dsn

[Reply](#)

Roger says:

2015/11/04 at 13:52

Hi! Thanks for your reply. I changed all the possible strange characters on the path and nothing changed. Here is the hole thing when I tried to run it through CMD:

```
C:\Program Files\KiCad\bin>java -jar freeroute.jar
HelpSet helpset/en/Help.hs not found.
help_broker is null
java.lang.Error: Illegal character
at designformats.specetra.SpecetraFileScanner.next_token(SpecetraFileScanner.java:1406)
at designformats.specetra.ScopeKeyword.skip_scope(ScopeKeyword.java:49)
at designformats.specetra.Component.read_place_scope(Component.java:353)
at designformats.specetra.Component.read_scope(Component.java:79)
at designformats.specetra.Component.read_scope(Component.java:45)
at designformats.specetra.ScopeKeyword.read_scope(ScopeKeyword.java:110)
at designformats.specetra.ScopeKeyword.read_scope(ScopeKeyword.java:110)
at designformats.specetra.DsnFile.read(DsnFile.java:85)
at interactive.BoardHandling.import_design(BoardHandling.java:1044)
at gui.BoardFrame.read(BoardFrame.java:180)
at gui.MainApplication.create_board_frame(MainApplication.java:301)
at gui.MainApplication.open_board_design_action(MainApplication.java:235)
at gui.MainApplication.access$200(MainApplication.java:31)
at gui.MainApplication$4.actionPerformed(MainApplication.java:171)
at javax.swing.AbstractButton.fireActionPerformed(Unknown Source)
at javax.swing.AbstractButton$Handler.actionPerformed(Unknown Source)
at javax.swing.DefaultButtonModel.fireActionPerformed(Unknown Source)
at javax.swing.DefaultButtonModel.setPressed(Unknown Source)
at javax.swing.plaf.basic.BasicButtonListener.mouseReleased(Unknown Source)
at java.awt.AWTEventMulticaster.mouseReleased(Unknown Source)
at java.awt.Component.processMouseEvent(Unknown Source)
at javax.swing.JComponent.processMouseEvent(Unknown Source)
at java.awt.Component.processEvent(Unknown Source)
at java.awt.Container.processEvent(Unknown Source)
at java.awt.Component.dispatchEventImpl(Unknown Source)
at java.awt.Container.dispatchEventImpl(Unknown Source)
at java.awt.Component.dispatchEvent(Unknown Source)
at java.awt.LightweightDispatcher.retargetMouseEvent(Unknown Source)
at java.awt.LightweightDispatcher.processMouseEvent(Unknown Source)
at java.awt.LightweightDispatcher.dispatchEvent(Unknown Source)
at java.awt.Container.dispatchEventImpl(Unknown Source)
at java.awt.Window.dispatchEventImpl(Unknown Source)
at java.awt.Component.dispatchEvent(Unknown Source)
at java.awt.EventQueue.dispatchEventImpl(Unknown Source)
at java.awt.EventQueue.access$500(Unknown Source)
at java.awt.EventQueue$3.run(Unknown Source)
at java.awt.EventQueue$3.run(Unknown Source)
at java.security.AccessController.doPrivileged(Native Method)
at java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(Unknown Source)
at java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(Unknown Source)
at java.awt.EventQueue$4.run(Unknown Source)
at java.awt.EventQueue$4.run(Unknown Source)
at java.security.AccessController.doPrivileged(Native Method)
at java.security.ProtectionDomain$JavaSecurityAccessImpl.doIntersectionPrivilege(Unknown Source)
at java.awt.EventQueue.dispatchEvent(Unknown Source)
at java.awt.EventDispatchThread.pumpOneEventForFilters(Unknown Source)
at java.awt.EventDispatchThread.pumpEventsForFilter(Unknown Source)
at java.awt.EventDispatchThread.pumpEventsForHierarchy(Unknown Source)
at java.awt.EventDispatchThread.pumpEvents(Unknown Source)
at java.awt.EventDispatchThread.pumpEvents(Unknown Source)
at java.awt.EventDispatchThread.run(Unknown Source)
```

[Reply](#)

andy *says:*

2015/11/04 at 16:51

How can you see that error when “trying to run” freerouter? Stack trace shows you are “opening DSN file” which is next step *after* running freerouting user interface... I feel misguided. Precise wording is critical when asking for help, dude 😊

I cannot help with Freerouting defects like this one around DSN parsing. Same error was seen here. I tried given DSN file and my freerouter also failed with the same stack trace as yours. In contrast design files I created and exported to DSN can be opened/consumed by freerouter. Reach out to someone with freerouter source code knowledge, maybe maintainer of mirrored code could help.

[Reply](#)

Roger *says:*

2015/11/05 at 23:41

OK, you're right, the problem appears when loading the DSN file. I'm sorry. So, it must be some difference between the given design, mine and yours, but which is these difference?

Thanks for your webpage and for your help!

Greetings from Spain!

[Reply](#)

andy *says:*

2015/11/06 at 01:31

Is your DSN file produced by Kicad or you exported it from some other tool? What I see is that my DSN files are ascii (text) while the LPC18XX_43XX_SCH.DSN file from link I mentined before is binary and is not a DSN file at all. In fact it is zip file (sic!) with binary files mentioning Orcad inside, probably full blown project, far from Specctra design file. So first of all check if your file is really DSN. Do your homework and create some other simple Kicad project, export to DSN and see if Freerouter can open it.

[Reply](#)

Denis Bisson *says:*

2015/11/22 at 18:34

Sir, you made my day!

Thank you very much regarding the little procedure to make “Freerouter” working.

I get used in the past 20 years to used PADS and SPECTRA and recently, for personal projects at home began to use KiCAD. I was tired of routing many traces manually and was unable to find something a low-cost autorouter. I heard about this “Freerouter” but I though it requires to run Linux until I read your page. I followed your procedure and it is autorouting in the backgrand as I am writing this message.

Thank very much again!

Dennis.

[Reply](#)

andy *says:*

2015/11/22 at 21:15

you are welcome 😊

[Reply](#)

Robert *says:*

2015/12/01 at 22:03

Great page, really, really helpful.

Thanks

[Reply](#)

siyad *says:*

2016/02/02 at 07:09

hi

i was try to autorouting in kicad,if i click “lauch free router via java webstart” option

it will shows an error

Unable to load resource: <http://freerouting.net/java/freeroute.jnlp>

how to fix it?

[Reply](#)

andy *says:*

2016/02/02 at 18:57

Button labeled this way was in old version of kicad and support for online freerouter was removed long time ago. I mentioned the about this “disapearance” in first paragraph of article, so read carefully referred topics.

If you want to use autorouting you have two options: stay with old kicad and read my old article on using offline copy of freerouter, or migrate to latest kicad and follow this article.

[Reply](#)**Tom** says:2016/02/29 at 02:05

I'm trying this with Linux Mint 17.3 Mate. I have installed version 4.0.2-4 +6225~38~ubuntu~14.04.1-stable release build, but, Autoroute is not working.

I have renamed the 3.8MB jar file. I'm not sure where the kicad binary file is located, but, I put a copy of freeroute.jar in /usr/bin and /usr/share, restarted kicad, but, it auroroute doesn't run.

Is this fix strictly for Windows or am I missing something? Thanks

[Reply](#)**andy** says:2016/02/29 at 14:00

I have only tested on my Win box where I do my uC works. It is more about *library loading* than operating system you run on. I have already said that [...] *when it detects freeroute.jar in Kicad binary directory [...]*, so you have to make sure that it is exactly where kicad executable is. I did not try in on Linux though (neither studied source code of Kicad) so I cannot prove it is only way or whether it works on specific linux distro or not.

In short: find you kicad executable and place JAR next to it, restart kicad and try again.

[Reply](#)**Saulo** says:2016/03/28 at 22:25

hello friend.

I have the same problem. Ubuntu 10.15. You got a solution?

[Reply](#)**Gary Levin** says:2016/03/09 at 06:03

Hi Andy,

Thanks tremendously.

I was able to follow your instructions and get the autorouter working.

Starting to bring up my tool chain.

Always had my boards done by PCB bureaus. Now I need to do them myself as a freelancer, developer. Kicad may just fit the bill.

Gary

[Reply](#)**andy** says:2016/03/09 at 16:22

Gary, glad to hear my scribble did the job 🙄 On local market I still find SW job offerings way more lucrative than HW, this is why I keep EE works as a hobby only. I keep fingers crossed for you though!

[Reply](#)**Damiano** says:2016/06/01 at 17:18

I am pointing out that FreeRouting is not dead, I am actually fixing it and making it more reliable.

It is on github called Engidea/FreeRoutingNew and I have a cheap web at <http://www.freerouting.eu>

If you want to do free angle routing it is still the only one available...

[Reply](#)

Pingback: [KiCad Libraries – Hilo90mhz](#)

[amichalec.net](#) :: homepage

Andrzej Michalec, Copyrights © 2006-2016

loading