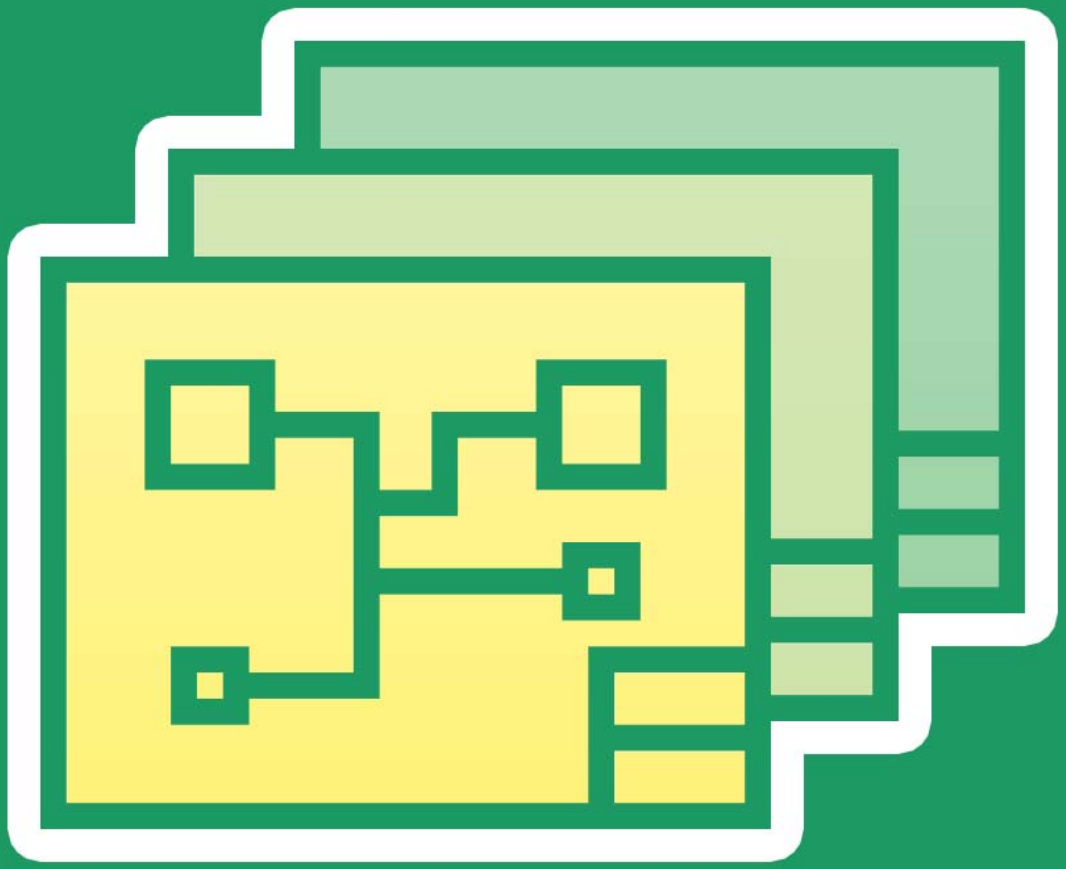
Altium

Electronic Rule Checking for Schematics



Christian Keller

Applications Engineer

ELECTRONIC RULE CHECKING FOR SCHEMATICS

INTRODUCTION

This technical paper covers a relatively undervalued function in the PCB design process giving you a leg up toward getting right-first-time designs. Many designers and companies put all their efforts in having a proper layout of the PCB and, nowadays, checking in real time the surrounding mechanical conditions.

But what if the schematic already contains errors? Usually there is a human driven review of the design but in times of rising design complexity and shortened timelines, it's more and more common for mistakes to sneak through. The electronic rule check (ERC) function in professional PCB design software can help you in finding and eliminating mistakes in the schematic by checking some basic rules and the 'grammar' underlying the design.

HOW DOES ERC HELP?

Answering this question is quite easy: by configuring the rules to perform checks that are specific to your design, ERC identifies problems so you can correct them early in the design. Plus, setting up and running ERC checks takes very little time…actually a fraction of the time you might spend manually checking the design. This lets you spend your time designing, not re-checking work.

One way to use ERC is to split up the preferences for this check in two general regions that cover the general 'grammar' of schematic design and a connection matrix defining which elements are allowed to be connected in which way, as shown in Figure 1.

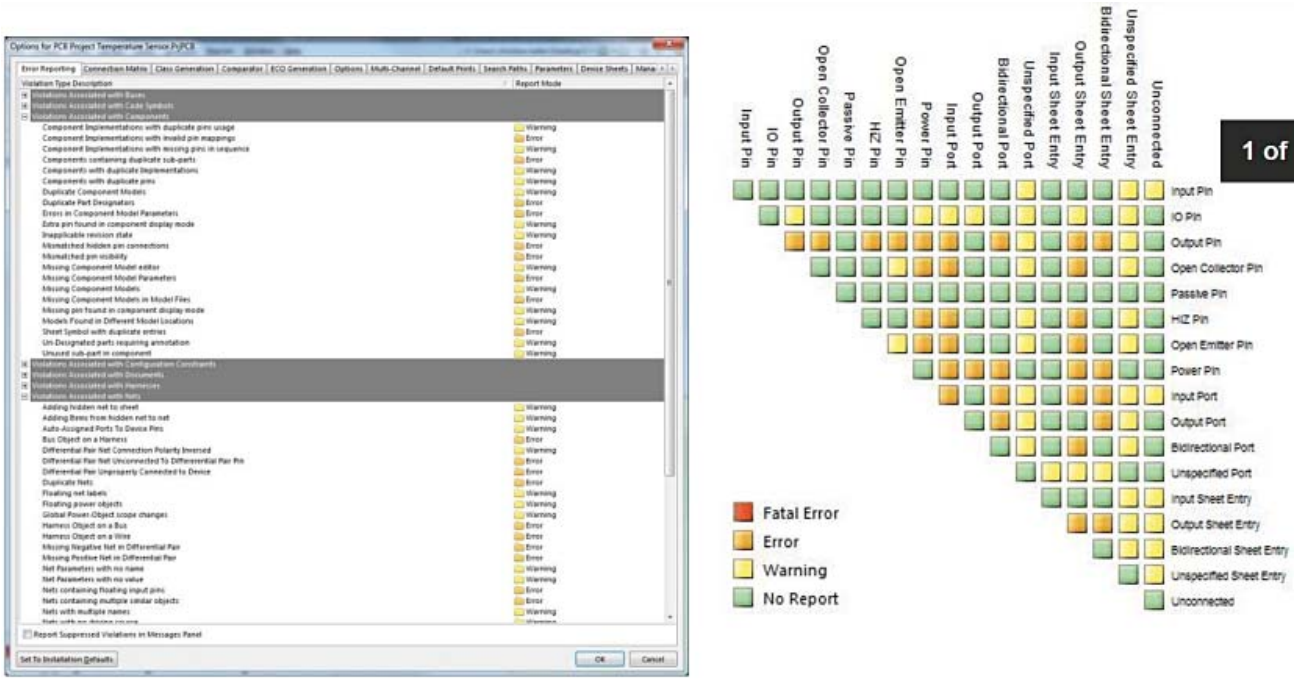


Figure 1: The ERC is split into 'grammar' checking and checking to ensure all connections are legal.

The 'grammar' area covers different settings regarding the usage of busses, components, documents, harnesses, nets, parameters etc.

THE GRAMMAR OF SCHEMATICS

As an example for violations in 'grammar', a floating net-label can be used (see Figure 2). It must be mentioned that issues like this one are not always as obvious as in this example, especially when dealing with imported designs.

This means that the area of grammar covers all checks regarding how to draw a proper schematic in the way the design tool expects it to be done.

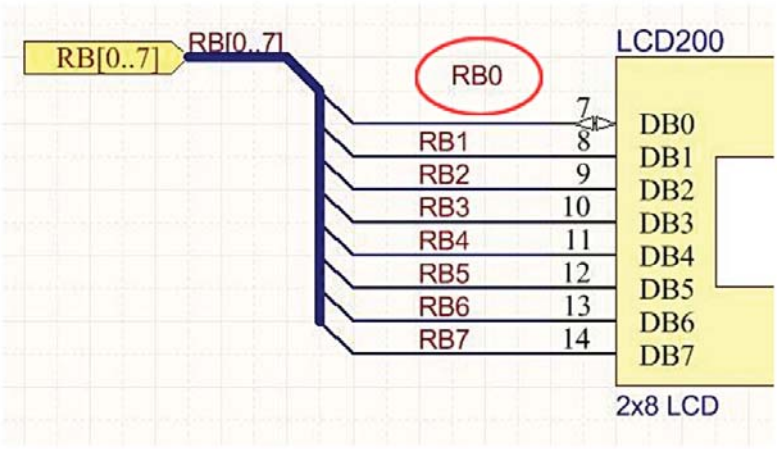


Figure 2: A floating net label is an example of a 'grammar' problem.

CONNECTIVITY CHECK

Even if everything has been drawn correctly, there are some remaining checks based on the connection matrix. This way, it is quite easy to check nets that may be spread over various schematic pages and moving through multiple hierarchical levels.

A direct connection is easy to check. It gets more complicated when multiple outputs are connected to one input, which is normally not allowed. But if the outputs are controlled by an internal enable procedure within the component it might be an acceptable design. Another example for a case where an unwanted warning is created can be seen in Figure 3.

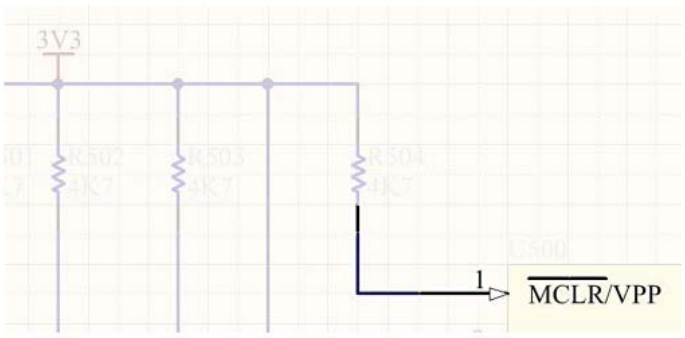


Figure 3: An input connected to a non-driven resistor creates an ERC warning.

An input pin being connected to the passive pin of a resistor will create an ERC warning because the input is not driven. How can we solve this? One way could be to teach the system the behavior of the resistor by attaching a behavioral model. But imagine the work if this has to be done for the full schematic for even most complex components.

Instead, you can override the system by placing so called "NoERC" elements and the system will not create further warnings. No ERC symbols can also be used for unconnected outputs to reduce error messaging.

CONCLUSION

This technical paper should have given you an idea of what you can expect from a state-of-the-art design tool when it comes to an automated rules checking of the schematic.

Altium Designer can support this way of working with a wide range of configurable check routines that can be used either to check the schematic after it has been created or by using the online ERC and batch checks when creating your production data.