

## RTL Analysis for Complex FPGA designs using a Grey Cell Methodology to Improve QoR

*This paper demonstrates how to solve the pressing issue of inter-block RTL analysis for complex designs using Blue Pearl's Grey Cell methodology.*

### Evolving Market Needs

As designs are getting more complex, we are seeing two major issues that customers have to deal with in the RTL analysis space.

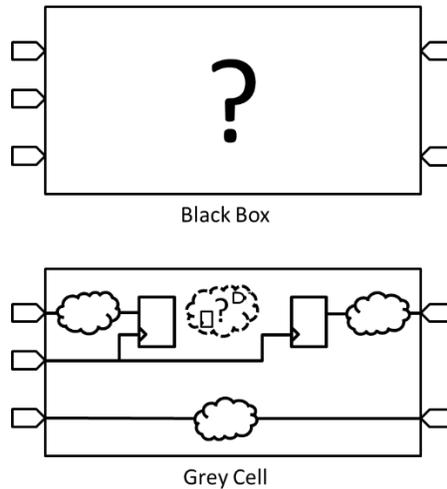
The first issue has to do with sheer design size and the associated data volume. As designs go through several iterations before they stabilize, it is inefficient to be dragging along every piece of design information all the time. It thus becomes incumbent upon EDA providers to alleviate the pain with new methodologies which can efficiently handle the data volume.

The second issue has to do with the increased use of purchased semiconductor Intellectual Property (IP) blocks. On average, every design brings together 50 to 100 different IPs that need to work well together. In most cases, designers who are using the IPs either do not know the inside of the IP well or do not have the information, which then adds tremendous difficulty during integration and verification because inter-IP issues are becoming predominant. It befits EDA providers to make inter-block RTL analysis possible.

By analogy, the first issue can be compared to the need for a hierarchical approach and the second issue to the increased importance of parasitic extraction as most delays reside outside the gates. In this article, we will show how Blue Pearl enables a methodology that relieves both issues with the **Grey Cell** methodology.

### What is a Grey Cell?

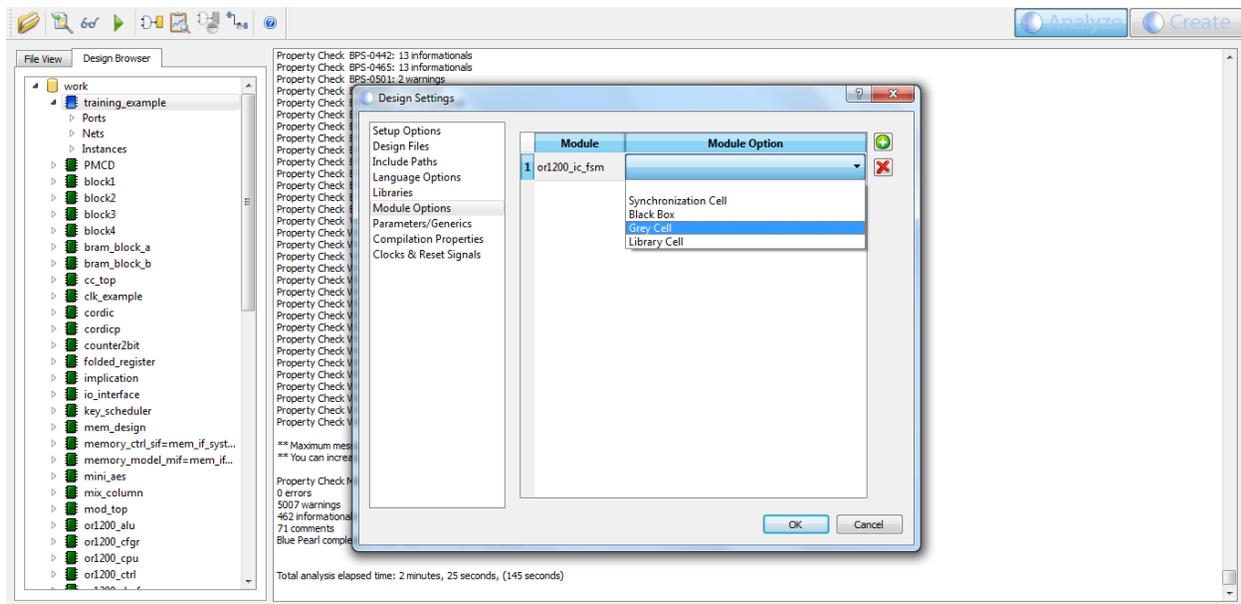
A Grey Cell, as depicted in **Figure 1** below, is a representation of a module which excludes all register-to-register logic. It contains only all logic from each input up to and including the nearest register, and all logic from each output back to and including the nearest register. A grey cell differs from a black box in that a black box has no logic inside. Grey cells enable the analysis of module-to-module connections while making abstraction of the details and/or preserving the trade secrets of the original IP provider.



**Figure 1: Grey cell vs. Black Box**

When a module is declared as a Grey Cell, Blue Pearl will disable some checks e.g. dangling nets, since the module does not contain the complete RTL representation. In general, IP providers will specify to their customers when they are providing a Grey Cell model. However, in case a designer sees superfluous messages for a purchased IP, the designer should confirm with the IP provider if indeed a Grey Cell model.

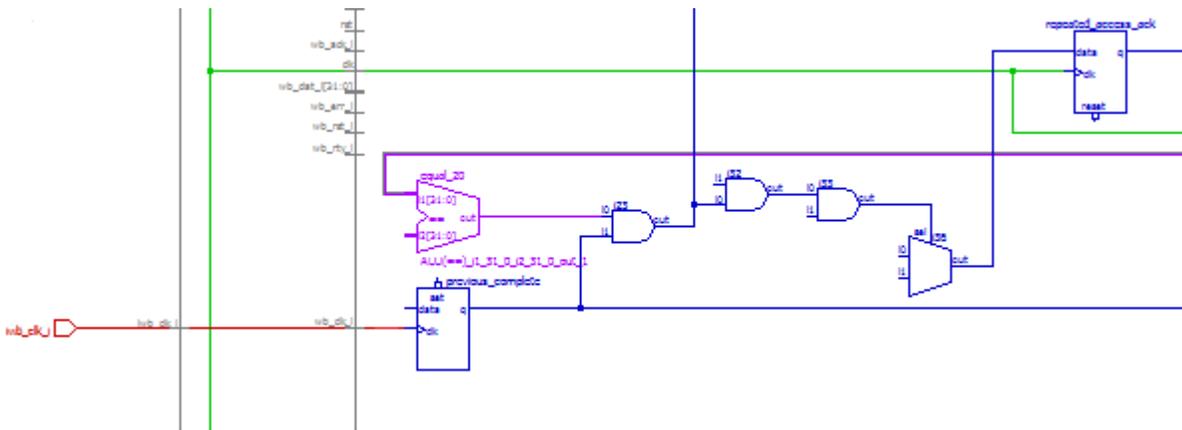
It is straightforward to define a module as a Grey Cell using the Blue Pearl Software Suite. The designer has to select the module and then declare it as a Grey Cell in the Module Options menu, as shown in **Figure 2** below.



**Figure 2: How to define a Grey Cell**

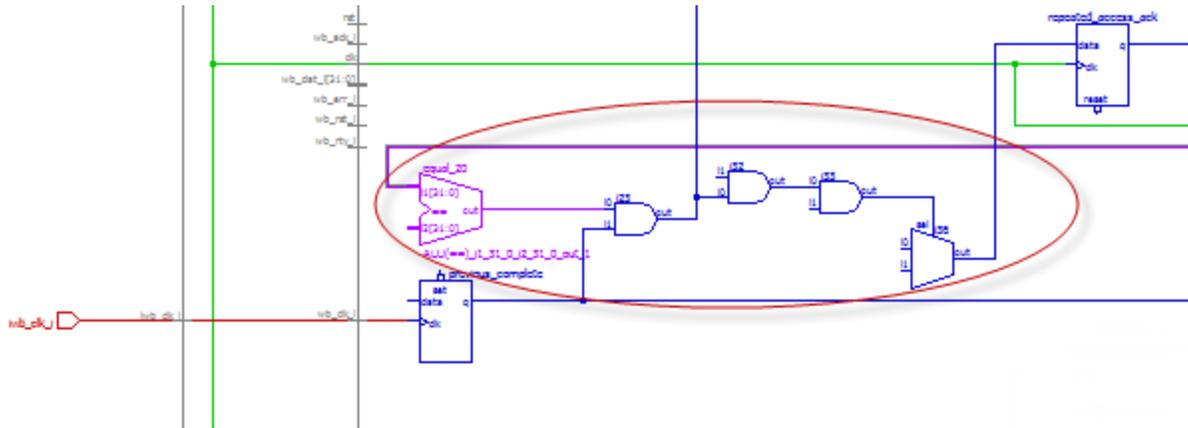
## Using Blue Pearl Software Suite to Handle Massive Designs

To illustrate the leading issue described in the first section of this article, let's assume that all the cells shown in **Figure 3** are part of the same module. In this portion of the design, the Blue Pearl Software Suite has detected a clock domain crossing (CDC) starting at the flip-flop labeled "previous\_complete" and ending at the flip-flop labeled "repeated\_access\_ack". Notice that the path is completely contained within the module. The designer would thus take the necessary action to fix any CDC issues within the module.



**Figure 3: CDC Detected during RTL Analysis**

As this module is being integrated within a bigger design, one may not want to keep all the details within the module. If the module is defined as a Grey Cell, all the elements within the red circle (see **Figure 4** below) will NOT be included in the analysis thus enabling the designer to analyze bigger designs a lot faster. Let's note that the number of elements in the circle can be quite significant, meaning one can abstract a lot of design details as low level modules pass the RTL analysis checks.



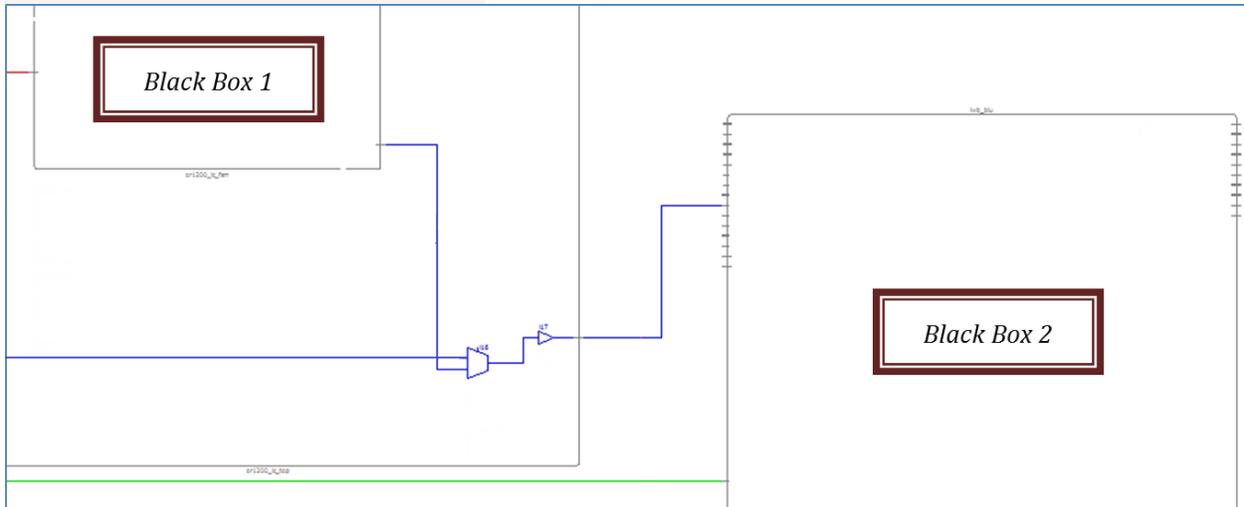
**Figure 4: Module has fewer details when defined as Grey Cell**

When a module is declared as a Grey Cell, the Blue Pearl Software Suite will eliminate all register-to-register logic from the internal model, thus saving analysis effort and run time.

### Using Blue Pearl Software to Handle Designs with a Large Number of IPs

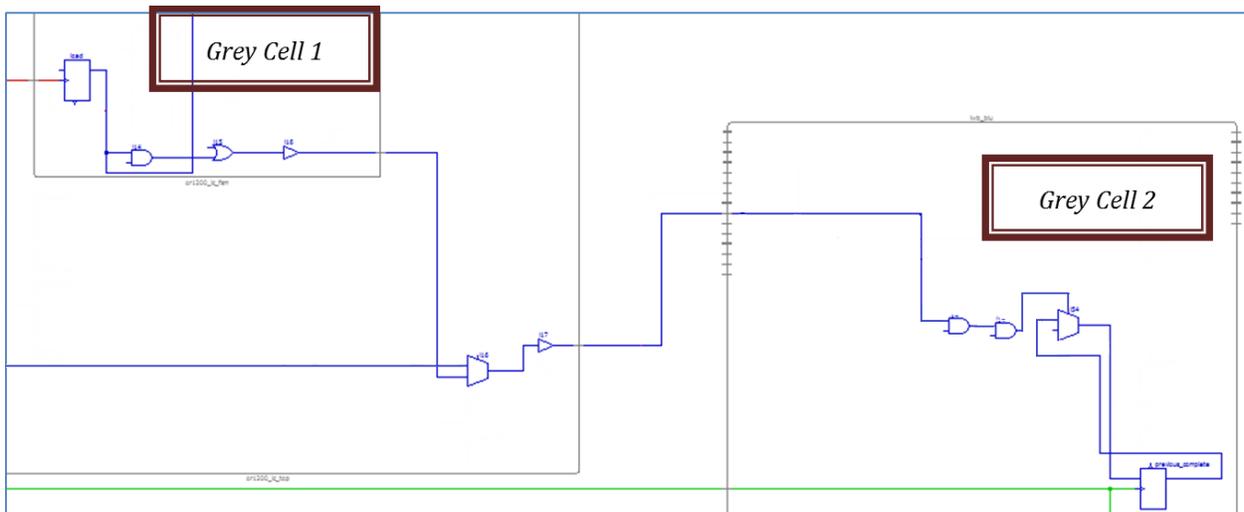
To illustrate the second issue described in the first section of this article, let's assume we have a design that contains numerous purchased IPs. The industry is rapidly demanding that IP sub-systems work well together to reduce time spent on integration and verification. As an example, a Memory IP sub-system would include the memory controller, the PHY and the Verification IP.

During chip assembly, if a designer considers the IPs to be black box (see **Figure 1** above) the risk of having field failures is greatly increased. This is due to the fact that designs now have 10 to 50 clocks that drive these IPs, and many CDCs occur between the IPs. In the example below, CDC analysis with black boxes does not show any violation (see **Figure 5 below**) between the blocks.



**Figure 5: Case where IPs are Black-Boxed**

When these IPs are defined as Grey Cells instead, the added details as compared to a Black Box, is sufficient to really find those troublesome issues that can crop up in the field. In this particular case, a CDC violation (**see Figure 6 below**) as indicated by the color coded diagram below is found. This should help the designer take the corrective action early in the design cycle.



**Figure 6: Analysis results using Grey Cell rather than Black Box**

### **Grey Cell enablement using the Blue Pearl Software**

With the Grey Cell methodology, IP providers can be assured that the implementation of their design know-how is protected while it provides designers with a sophisticated method to catch issues before tape out. Designers can now catch those dreadful issues early in the flow while doing RTL analysis on bigger designs.