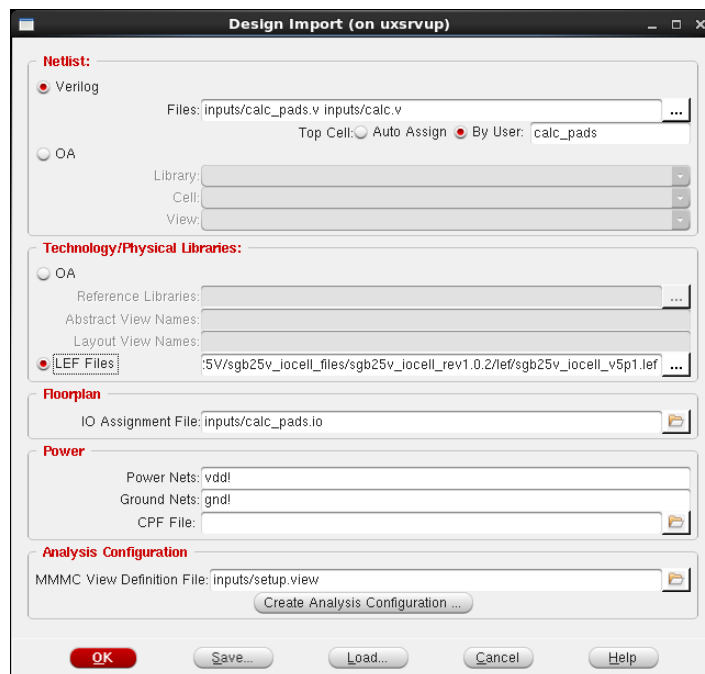


# Chipentwurf

## Übungsblatt 5

31. Januar 2019

1. Nachdem unser Chip erfolgreich synthetisiert wurde soll nun das Layout erstellt werden. Dazu verwenden wir das Place&Route-Tool *Cadence Innovus*. Im Folgenden wird der Arbeitsablauf exemplarisch dargestellt. Die einzelnen Schritte müssen ggf. an Ihr Design angepasst werden.
2. Wechseln sie in das Verzeichnis `layout` innerhalb Ihres Projektverzeichnisses und starten Sie Innovus durch den Befehl `innovus`.
3. Öffnen Sie den Dialog zum Design-Import (`File` → `Import Design...`):



Füllen Sie die Felder entsprechend der Abbildung aus. Gehen Sie sicher dass Sie alle verwendeten Dateien kennen und eine Vorstellung von Ihrem Inhalt haben.

4. Spezifizieren Sie den Floorplan (`Floorplan` → `Specify Floorplan...`). Denken Sie an die Power-Ringe und die Verdrahtung der IOs. Achten Sie darauf dass der Padframe lückenlos durch IO-Filler geschlossen werden kann.
5. Erstellen Sie Power-Rings (`Power` → `Power Planning` → `Add Ring...`).
6. Erstellen Sie Power-Stripes (`Power` → `Power Planning` → `Add Stripe...`).
7. Deklarieren Sie die globalen Netze. Die benötigten Befehle sind in der Datei `inputs/global_nets.tcl` für Sie vorbereitet.
8. Bei den Versorgungsnetzen handelt es sich um sog. Special Nets. Starten Sie das Special Routing (`Route` → `Special Route...`). Sie können die Standardeinstellungen verwenden.
9. Anschließend platzieren Sie das Design mit den Befehl `placeDesign`.
10. Sie können das Zeitverhalten in Bezug auf die Setup-Zeiten analysieren durch `timeDesign -preCTS`. Um die Hold-Zeiten zu überprüfen verwenden sie zusätzlich den Parameter `-hold`.
11. Jetzt soll der Clock Tree synthetisiert werden. Wir verwenden die Clock Tree Engine CCOpt.
  - a) Zuerst muss aus den bereits geladenen Constraints eine Clock Tree Spezifikation erstellt werden:
 

```
create_ccopt_clock_tree_spec -file <file name>
```
  - b) Bisher wurde das Clock-Netz als ideal angenommen und für den Takt keine Verzögerung berechnet. Damit ist jetzt Schluss. Schalten Sie durch die folgenden Befehle den Analysis-View um und sorgen Sie dafür dass die Clock propagiert wird:
 

```
set_analysis_view -setup {view_wst_postCTS view_typ_postCTS view_bst_postCTS}
-hold {view_bst_postCTS}
set_interactive_constraint_modes [all_constraint_modes -active]
set_propagated_clock [all_clocks]
```
  - c) Lassen Sie nun den Clock-Tree durch `ccopt_design` implementieren.
12. Analysieren Sie erneut das Zeitverhalten. Denken Sie daran, dass Sie sich nun in dem Zustand `postCTS` befinden. Für den Fall dass Sie Probleme feststellen, können Sie das Design mit `optDesign -postCTS` in Bezug auf die Setup-Zeit optimieren. Um in Bezug auf Hold-Zeiten zu optimieren verwenden sie zusätzlich den Parameter `-hold`. Konnten Sie alle Probleme beheben?
13. Nun können Sie das Design routen (`Route` → `NanoRoute` → `Route...`). Auch hier sind die Standardeinstellungen akzeptabel.
14. Führen Sie einige Überprüfungen durch:
 

```
Verify → Verify Geometry...
Verify → Verify DRC...
Verify → Verify Connectivity...
```

15. Abschließend exportieren Sie

- Das Layout als DEF
- Die Netzliste als Verilog
- Das Zeitverhalten als SDF