

Praktikum Schaltungsintegration (PSI)

Versuch Nr.: SI2
Versuchstitel: Dynamisches Verhalten von CMOS-Invertern und -Gattern
Versuchskonzept: C. Schimpfle / F. Aschauer / D. Kohlert

I. Versuchsziel

In den Versuchen SI0 bis SI3 sollen Sie den Umgang mit einem sehr leistungsfähigen, in der Industrie eingesetzten EDA (Electronic Design Automation)-Software-Paket kennen lernen. Sie werden den Einsatz der Programme anhand von vollständigen Entwurfsabläufen (Design Flows) für Full-Custom- und Standardzell-entwürfe üben.

II. Versuchskurzbeschreibung

In diesem Versuch lernen Sie den vollständigen Entwurfsablauf von der Schaltplaneingabe bis zur Post-Layout-Analyse kennen. Um hierarchisches Design einzuüben, entwerfen Sie zunächst ein 4-fach-NAND-Gatter als Standardzelle. Diese kombinieren Sie danach mit bereits existierenden Standardzellen zu einem (etwas) komplexeren Funktionsblock. Diesen und weitere bereits vorhandene Funktionsblöcke untersuchen Sie im Hinblick auf ihr Zeitverhalten.

III. Der Entwurfsablauf (Design Flow) mit Cadence-EDA-Programmen

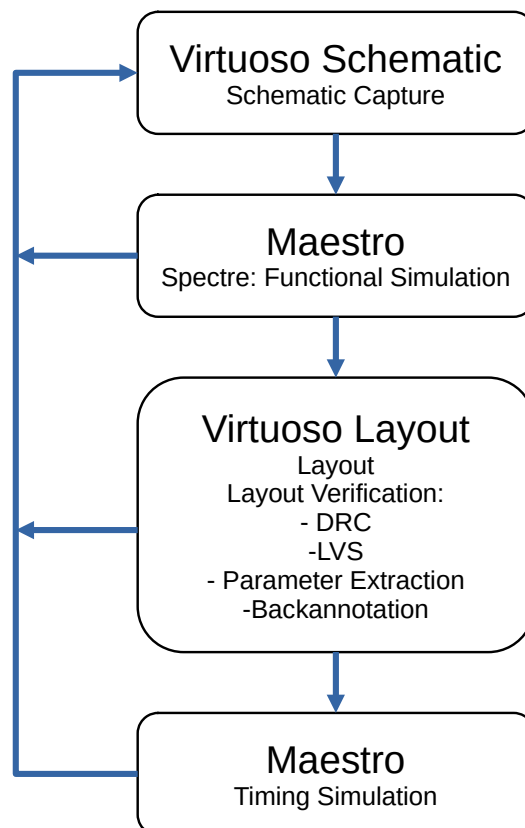


Abb. 1: Schaltplan-basierter Entwurfsablauf

Abb. 1 zeigt einen Schaltplan-basierten Entwurfsablauf mit Cadence. Er umfasst die folgenden Schritte:

- Schaltplaneingabe (Schematic Capture) mit **VIRTUOSO**. Sie können mit diesem Programm auch einer beliebigen Ansammlung von verknüpften Komponenten ein Symbol Ihrer Wahl zuordnen, um diesen Funktionsblock dann in einer höheren Hierarchieebene wiederzuverwenden.
- Funktionale Simulation mit **SPECTRE**. Hier wird überprüft, ob die Schaltung die gewünschte Funktionalität aufweist. An eine erfolgreiche Simulation schließt sich üblicherweise die Testvektor-Erzeugung an, also die Festlegung derjenigen Signale (Stimuli), mit denen die produzierten Schaltungen getestet werden (vgl. auch Praktikum PTT).
- Layouterstellung mit **VIRTUOSO XL**. In diesem Schritt werden die Geometriedaten erzeugt, mit denen die Masken für die Produktion der integrierten Schaltung hergestellt werden. Je nach Hierarchieebene werden einzelne Polygone editiert (unterste Ebene) und damit z.B. Standardzellen entworfen, oder es werden (Standard-)Zellen platziert und verdrahtet, sodass komplexere Funktionsblöcke entstehen. Das Programm führt auch wichtige Tests und Kontrollen zur *Layoutverifizierung* durch: die Überprüfung, ob alle geometrischen Entwurfsregeln eingehalten wurden (**Design Rule Check, DRC**), die Überprüfung, ob das Layout topologisch mit dem Schaltplan übereinstimmt (**Layout Versus Schematic, LVS**) und schließlich auch die Extraktion von Komponentenparametern für die folgende (Analog-)Simulation des Zeitverhaltens.
- Post-Layout-Simulation mit **SPECTRE**. Mit den extrahierten Parametern für die Transistoren und parasitären Kapazitäten wird eine analoge (SPICE-)Simulation durchgeführt, um das (Zeit-)Verhalten der Schaltung zu überprüfen.

Ein Programmpaket wie das soeben beschriebene wird erst benutzbar, wenn es mit passenden Technologiedaten "gefüttert" wird. In diesem Praktikum benutzen wir die Skywater 150nm Design Rules.

IV. Versuchsdurchführung

IV.1 Schaltplaneingabe für das 4-fach-NAND-Gatter

Schaltplan 4-fach-NAND-Gatter zeichnen:

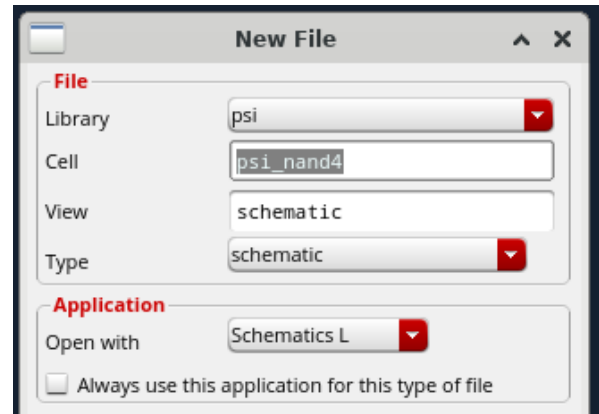
Virtuoso aufrufen

Im Virtuoso-Command-Window: File/New/Cellview

Fenster erscheint:

Ausfüllen wie gezeigt

"OK"



Zeichenfenster erscheint

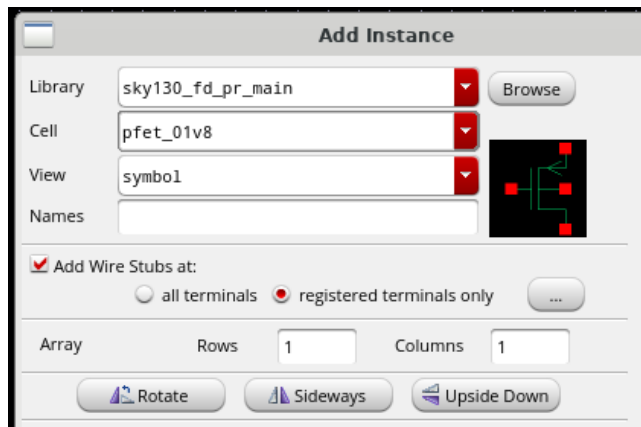
Transistor aus Library holen:

Keyboard Shortcut: "i":

Fenster erscheint:

Ausfüllen wie gezeigt:

p-FET aus Library sky130_fd_pr_main wird eingefügt:



Symbol hängt am Cursor, kann mit LMB platziert werden.

Unterhalb erscheint ein Fenster mit Transistor-Eigenschaften, können so bleiben.

Auf gleiche Weise n-FET einfügen

Kopieren von Bauteilen: Keyboard Shortcut: "c":

Bauelement verschieben:

Mauszeiger auf Bauelement bis gelbes Rechteck erscheint, mit gedrücktem LMB verschieben

Bauelemente verdrahten:

Keyboard Shortcut: "w"

Externen Anschluss (Pins) platzieren:

Keyboard Shortcut: "p", Fenster ist selbsterklärend

Erstellen Sie die Schaltung eines 4-fach-NAND

Platzieren Sie alle Inputs, Outputs und auch Versorgungen als Pins

Versorgungsanschlüsse sind "Input", Achtung auf Ausgang, muss "Output" sein

Um Probleme mit Keywords (in, out, vdd, vss usw.) zu vermeiden, sollten Anschlussnamen gewählt werden, die sicher keine Keywords sind.

Vorschlag: Alle Namen beginnen mit "n_" ("n" für node)

Mit "Check and Save" werden Fehler in der Schaltung erkannt, gelegentlich durchführen, auf alle Fälle zum Schluss !

IV.2 Funktionale Simulation

IV.2.1 Vorüberlegungen

Die Simulation soll zunächst nur die funktionale Korrektheit des Gatters nachweisen, Untersuchungen des Laufzeitverhaltens werden später durchgeführt, wenn die Einflüsse der parasitären Kapazitäten im Layout mit berücksichtigt werden können. Es wird daher hier mit dem unbelasteten Gatter simuliert.

Überlegen Sie sich für die Simulation Stimuli-Verläufe zum testen des NAND (4 Quellen), wobei alle möglichen Eingangsbitkombinationen abgedeckt werden sollen. Aufgrund der zu erwartenden Gatterlaufzeiten soll eine Pulsweite von 500ps nicht unterschritten werden. Die Versorgungsspannung für diese Technologie liegt bei 1,8V. Dokumentieren Sie die Signalverläufe !

IV.2.2 Setup Simulation

Quellen angeben

Im Virtuoso-Command Window:

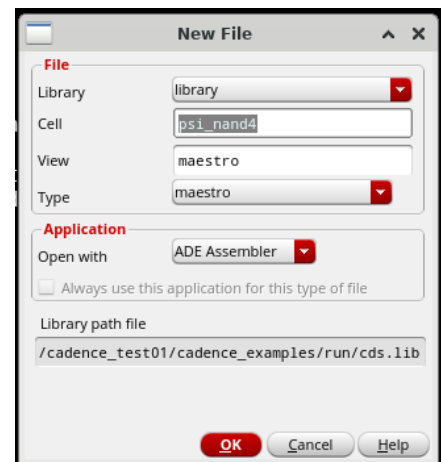
Start/New/Cellview

Nebstehendes Fenster erscheint

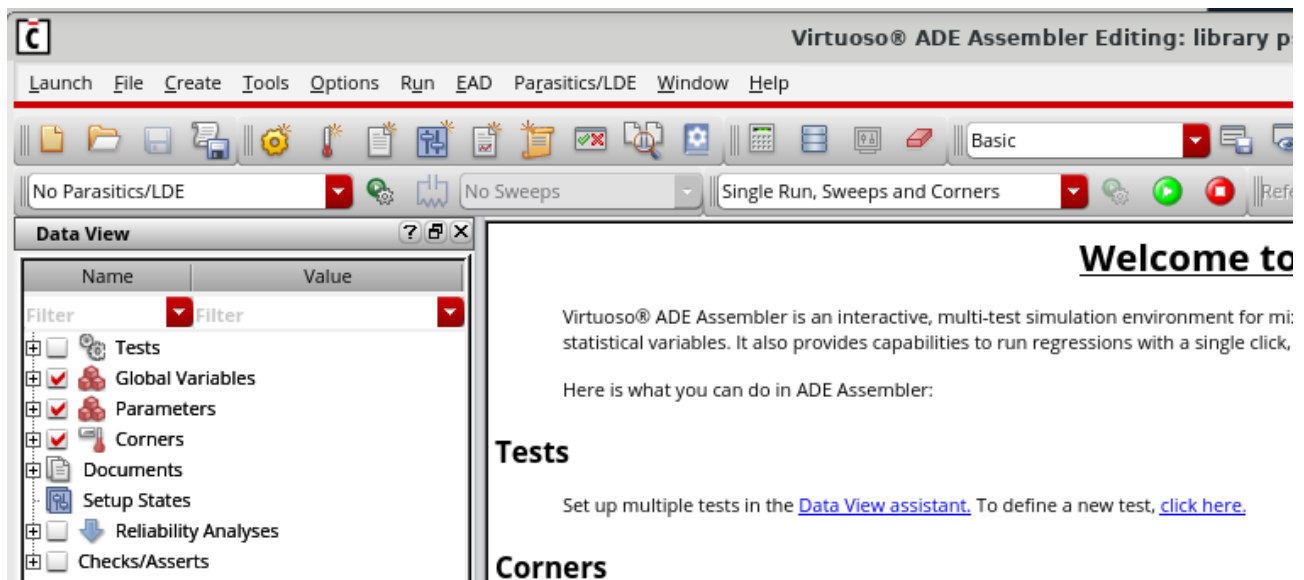
Type "Maestro" angeben

Cell: name der zu simulierenden Zelle (Hier: psi_nand4)

ok

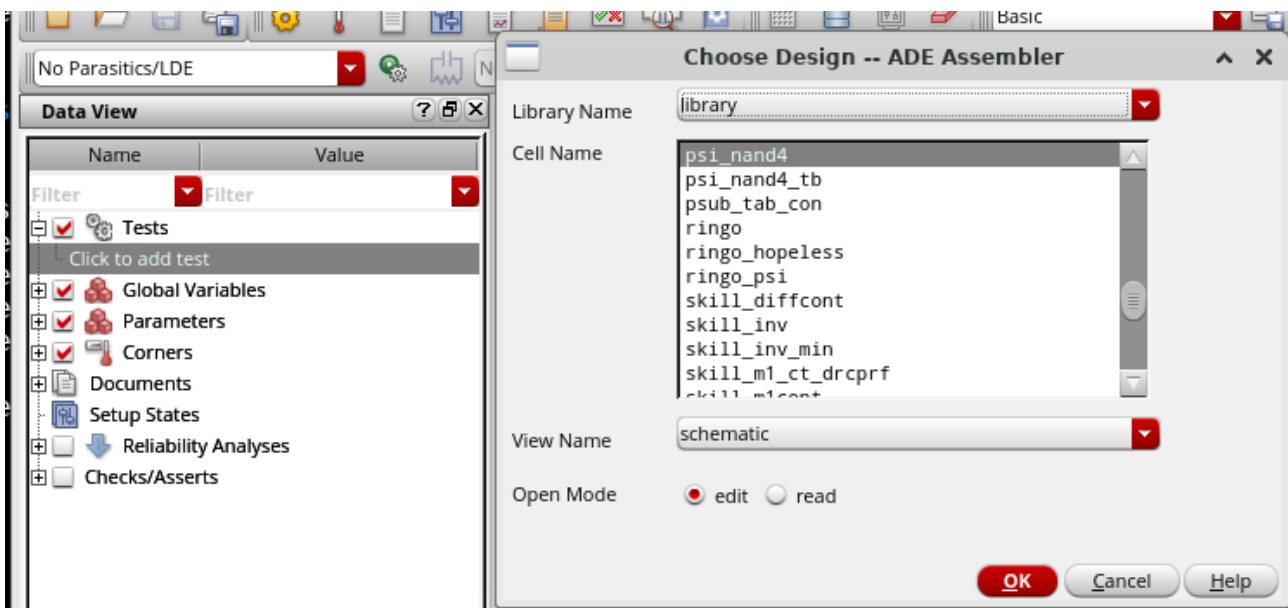
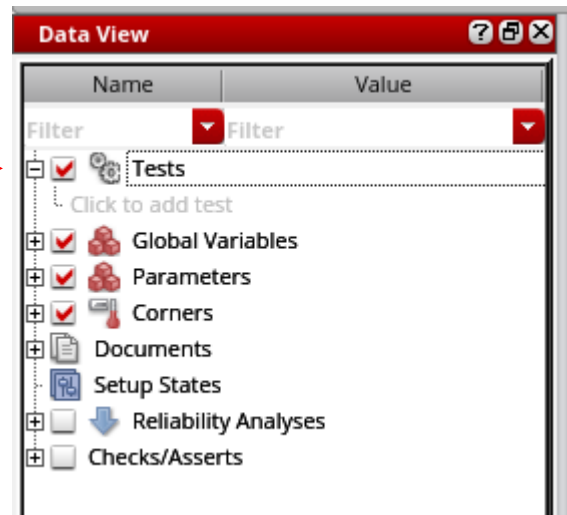


Fenster erscheint:



Neuen Test erstellen:

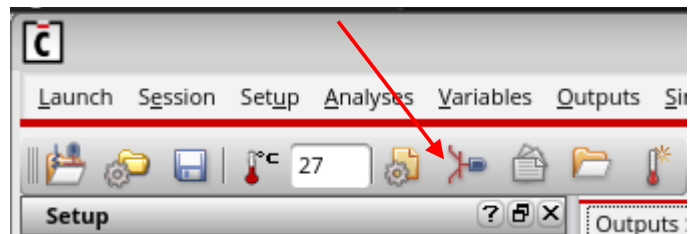
Es können für jede Zelle beliebig viele, unterschiedliche Tests angelegt werden, Hier ist aber ein Test mit Transientenanalyse ausreichend.



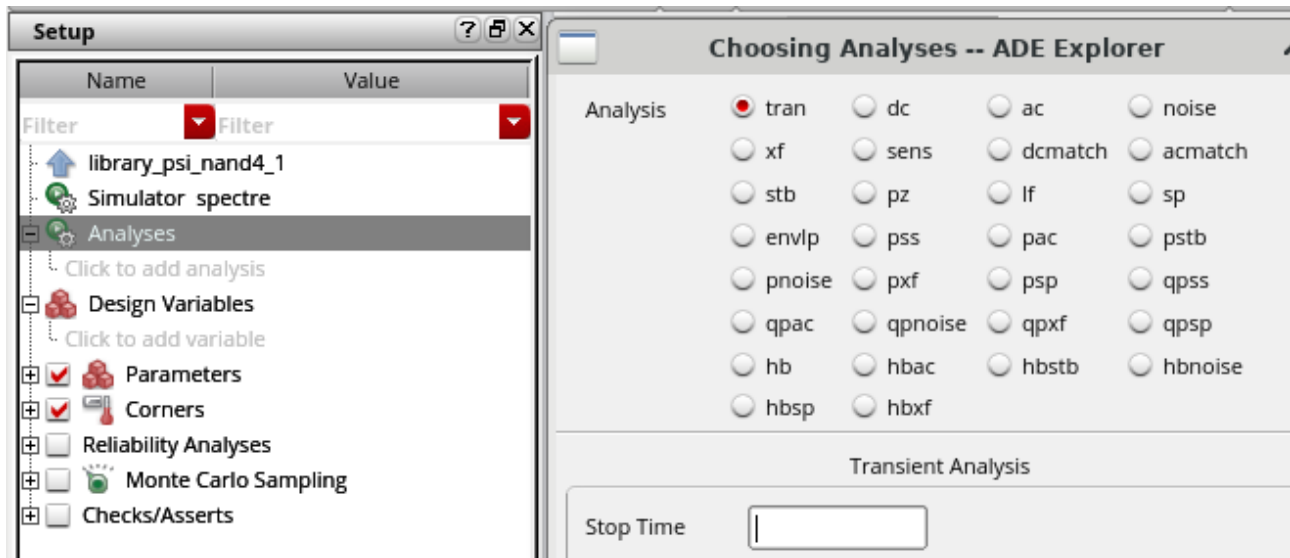
"Click to add test" -> Auswahlfenster erscheint, gewünschte Zelle auswählen

Leeres Fenster erscheint

Quellen angeben: Transistorsymbol klicken:



Analyse-Modus wählen:

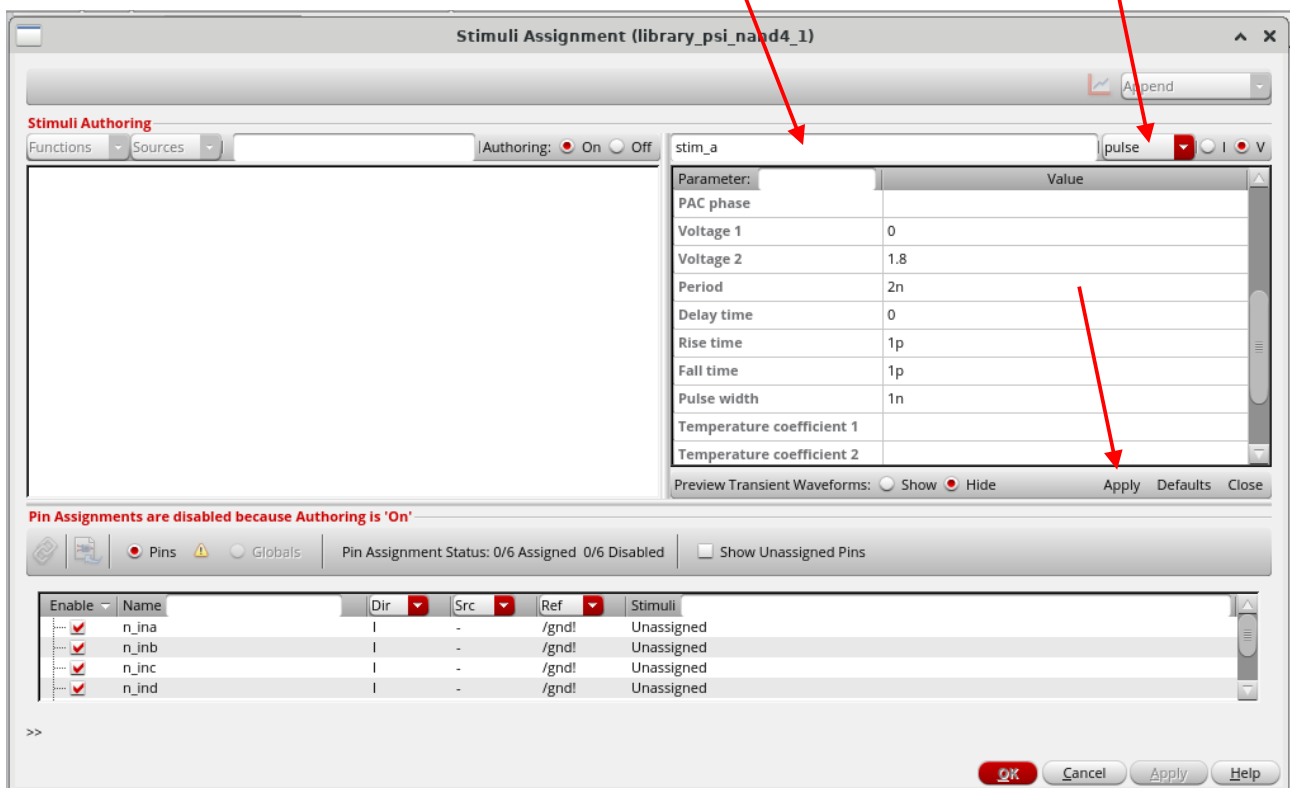


Hier: Transienten-Analyse auswählen
Stop-Time angeben (z.B. 1n **NICHT ns !!**)

Stimuli-Assignment-Fenster erscheint, IO's erscheinen bereits im unteren Fenster:
Wenn nicht, Transistorsymbol nochmal klicken
Quellen zuweisen: "Authoring"-Modus muss angewählt sein
Namen der Quellen nicht gleich Signalnamen

Name der Quelle (reinschreiben):

Quelle:



Art der Quelle: Hier "pulse", alle üblichen Spice-Quellen sind verfügbar

Es sollen Rechtecksignale mit minimaler Pulsbreite von 500ps, Risetime = Falltime = 1p und 50% Duty Cycle verwendet werden.

Voltage 1: Low-Pegel

Voltage 2: High-Pegel (1.8V)

Period: Periodendauer

Delay Time: Wartezeit zu Beginn der Simulation (0ps hier)

Rise Time: Anstiegszeit des Signals (z.B. 1ps)

Fall Time: Abfallzeit des Signals (z.B. 1ps)

Pulse Width: Pulsbreite

Wenn alle Eingaben gemacht : "Apply" drücken, Angaben werden übernommen

Angaben für alle Quellen machen, DC-Quellen nicht vergessen !

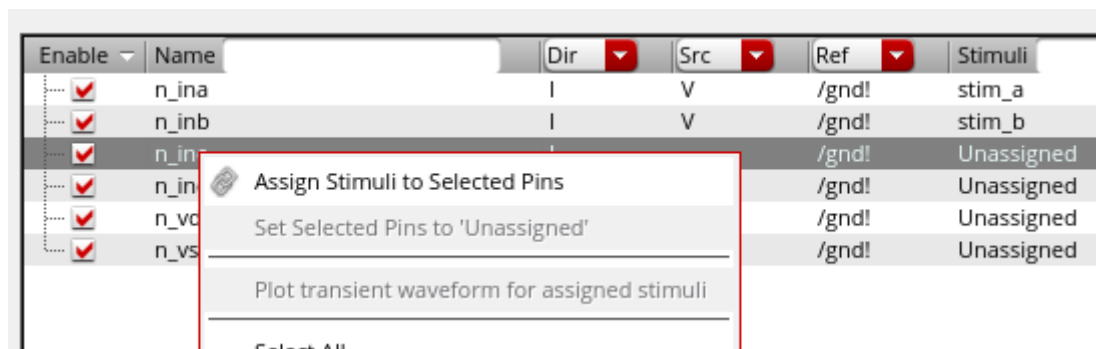
Wenn alle Quellen beschrieben: "Authoring"-Modus ausschalten

Zuordnen der Stimuli:

Signal im Authoring-Fenster anwählen



Rechtsklick auf Signalnamen (Assign) realisiert Zuordnung



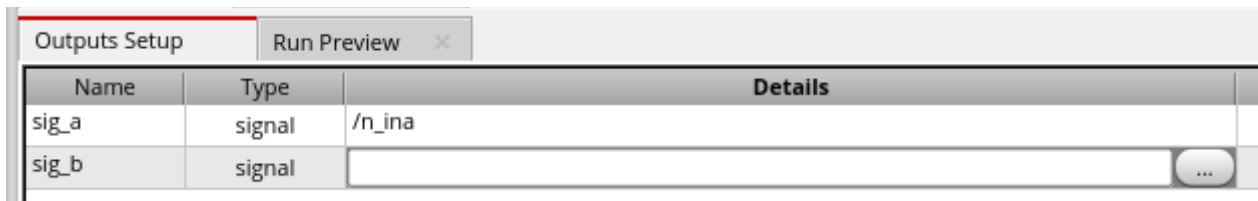
Wenn alles erledigt: Apply/ok

Signale plotten:

Signale auswählen:

Hauptmenü: Outputs/Add/Signal

Fenster erscheint:



Beliebiger Name

Hier reinklicken

Button erscheint, anklicken, Schematic erscheint

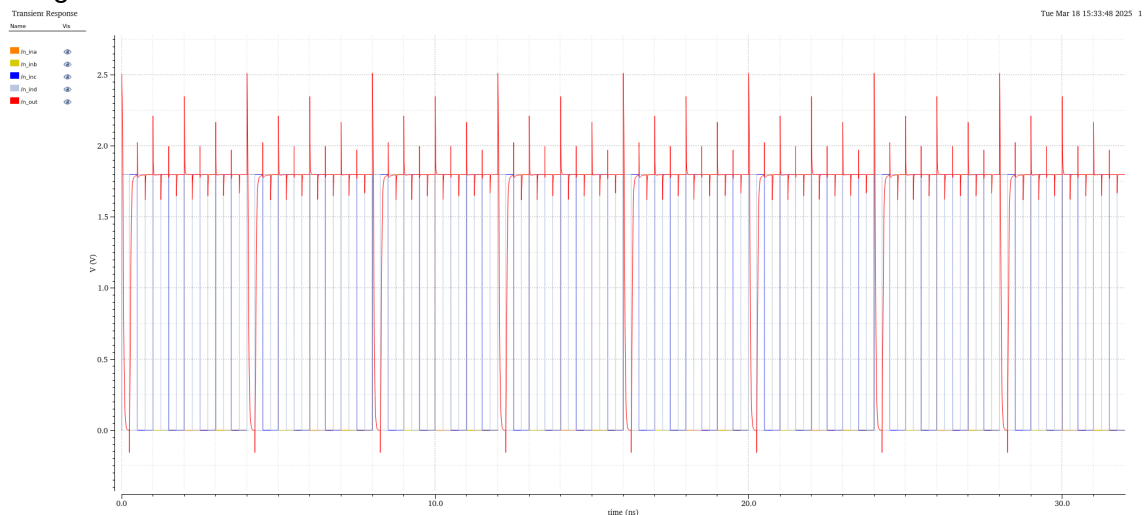
Das zu plottende Signal kann im Schematic ausgewählt werden.

IV.2.3 Durchführung der Simulation, Darstellung der Signale

Wenn alle gewünschten Signale zugeordnet:

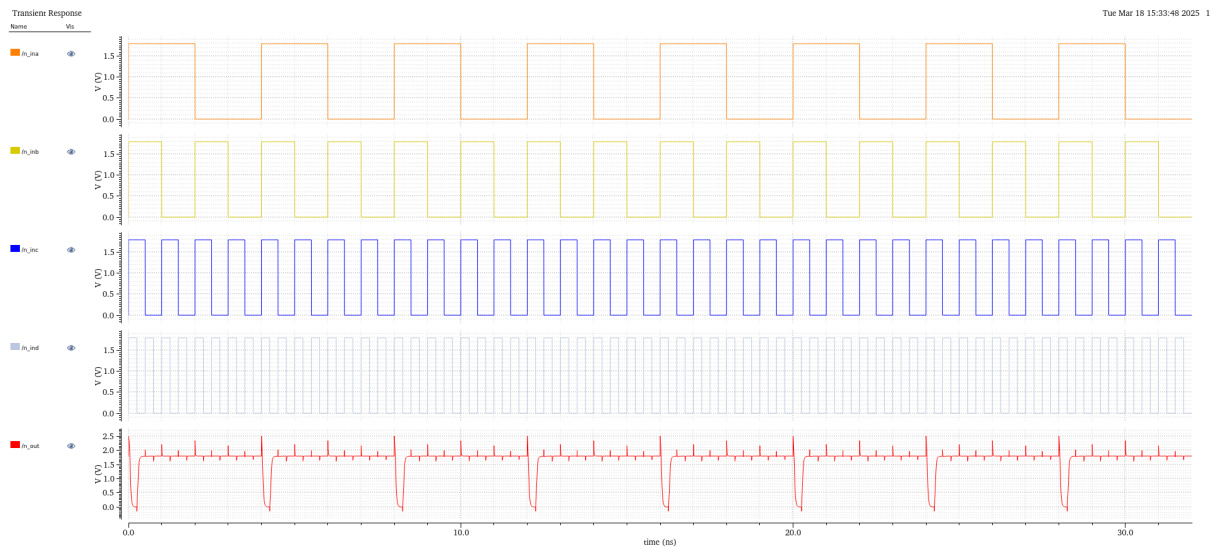
Hauptmenü: Simulation/Netlist and run

Signalfenster öffnet sich:



Signale einzeln untereinander darstellen:

Hauptmenü des Plotfensters : Graph / Split Current strip / Trace



Waveform-Fenster in File oder Clipboard abspeichern: File/save Image
In Protokollfile eintragen

Messungen im Zeitbereich: Measurements / Transient
Anstiegszeiten, Abfallzeiten, Delays verfügbar

Signalnamen rechtsklicken: Menü erscheint: Measurement / Time Domain / Delay

Weisen Sie anhand des Plots nach, dass das 4-fach-NAND korrekt funktioniert, Plot in Protokoll !
Wenn fertig, alle Simulationsfenster schliessen

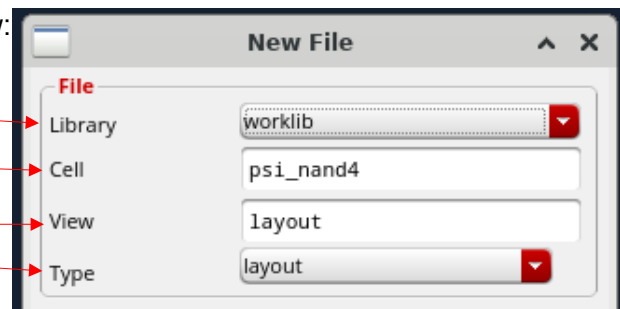
IV.3 Layout

Im Virtuoso-Command-Window: File/New/Cellview:

Working Library

Name (wie Schematic)

Typ Layout



Schematic-Window und leeres Layout-Window öffnen sich:

Layout mit vorhandenen FET's erzeugen:

Button ganz links unten



oder: Connectivity/Generate/All from Source, ok

Layout mit FET's erscheint:

FET's günstig platzieren, Verdrahten mit Rectangles und Wires. Benutzen Sie zur Spannungsversorgung die selbst gelayouteten Stromschienen, evtl auch mehrere nebeneinander !

In der mitgelieferten Standardzell-Bibliothek liegt der Abstand der li1-Lagen der oberen und der unteren Stromschiene bei 4µm, das soll eingehalten werden.

Die Stromschienen müssen innerhalb der Zellenbegrenzung Liegen !

Mit Hilfe des Befehls /Edit/Flip lassen sich Transistoren spiegeln, so dass sich Drain/Source-Gebiete auf dem gleichen Potential unmittelbar gegenüberliegen. Weiterhin sollen FET's mit Gates, die verbunden werden sollen, so platziert werden, dass die Gates genau übereinanderliegen. Damit können sie mit der "wire"-Funktion einfach verbunden werden.

Externe Anschlüsse:

Ein- und Ausgänge der Schaltung benötigen einen "Pin-Designator", das ist ein Rechteck der Ebene "Metal1 pin", das auf der Anschlussfläche platziert wird. Zusätzlich wird ein "Metal1 drw" Fleck in der obersten Hierarchiestufe benötigt, nur dieser Fleck kann mit einem Label versehen werden.

Label erstellen:

Annahme: Anschluss liegt auf Metal1

Benötigt wird: Pin-Designator = Ebene Metal1 pin, zusätzlich Fleck auf met1 drw platziert

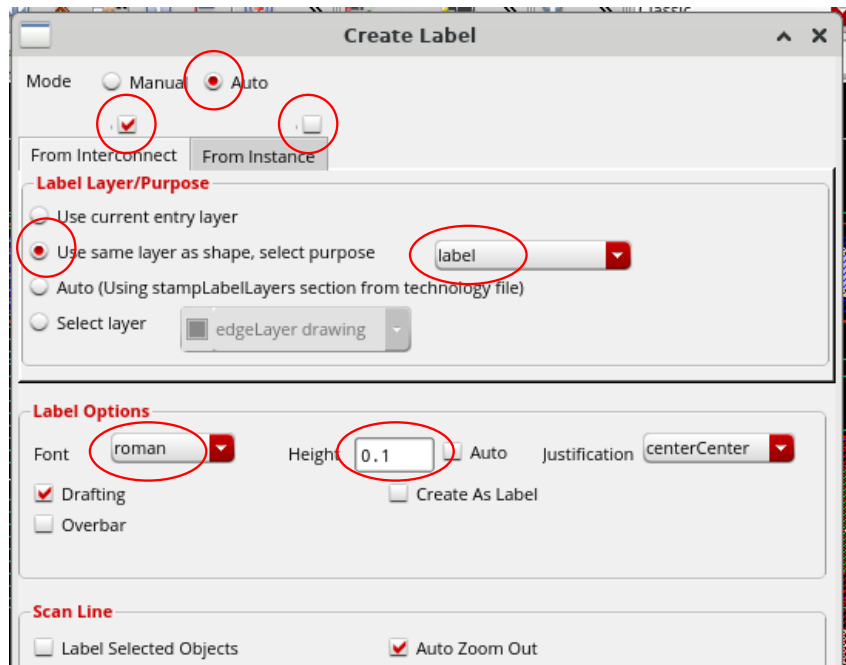
Label:

nur met1 drw aktivieren

Shortcut **L**, F3

Nebenstehendes Fenster erscheint:

Ausfüllen wie gezeigt:



Wenn alles erledigt: DRC durchführen, evtl. Fehler beseitigen

Weiterer Check: Connectivity/Check against source
Liefert aber kryptische Fehlermeldungen

Die nächsten Punkte fehlen noch im Skript:

Nächste Schritte: Pegasus DRC
Pegasus LVS (Layout versus schematic)
Pegasus Parasitics Extraction
Spectre Timing Simulation