

# Einleitung zur den EDA Tools

## Labor Digitales Design

## Inhalt

1 Ziel .....	1
2 Einführung .....	2
3 Kombinatorische Logikschaltung .....	3
3.1 Entwurf .....	3
4 Schaltungs-Editor .....	5
4.1 Komponenten .....	5
4.2 Signale .....	5
4.3 Aufgaben .....	6
4.4 Layout .....	7
5 Simulation .....	8
5.1 Testbench .....	8
5.2 Simulator .....	8
5.3 Anzeige .....	10
5.4 Störung .....	12
6 Checkout .....	13
Glossar .....	14

## 1 Ziel

Dieses erste Labor dient dazu, die Entwicklungswerkzeuge der automatisierten Entwicklung in der Elektronik [Electronic Design Automation \(EDA\)](#) bedienen zu lernen. Es werden das Schaltungsentwicklungstool **Mentor HDL-Designer** und der Schaltungs-Simulator **Mentor Modelsim** benutzt, sowie die Verbindung zwischen den beiden. Als Beispiel wird eine einfache Logikschaltung mit Invertern, UND- und ODER-Gattern erstellt und simuliert.



Abbildung 1 - Mentor HDL Designer



Abbildung 2 - Mentor ModelSim



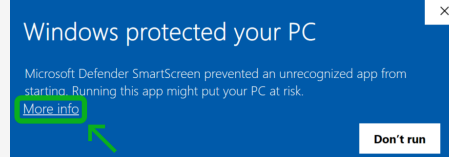
## 2 Einführung

Befolgen Sie die folgenden Schritte, um die Labordateien herunterzuladen und einzurichten:

- Laden Sie die für das Labor erforderlichen Dateien über folgenden [Link \(https://github.com/heis-synd-did/did-labs/archive/refs/heads/main.zip\)](https://github.com/heis-synd-did/did-labs/archive/refs/heads/main.zip) herunter.
- Entpacken Sie das Archiv auf Ihrem Netzlaufwerk: \\filer01.hevs.ch\FS\_COURSES\102.1-DiD-102.2-CAR\<firstname.lastname>.
- Führen Sie die Datei **did-labs/did-labs.bat** aus.



Wenn eine Sicherheitsnachricht erscheint (*nur beim ersten Ausführen*), klicken Sie auf **More Info** und **Run anyways**



- Nachdem HDL Designer geöffnet wurde, klicken Sie auf die Registerkarte **Project**.

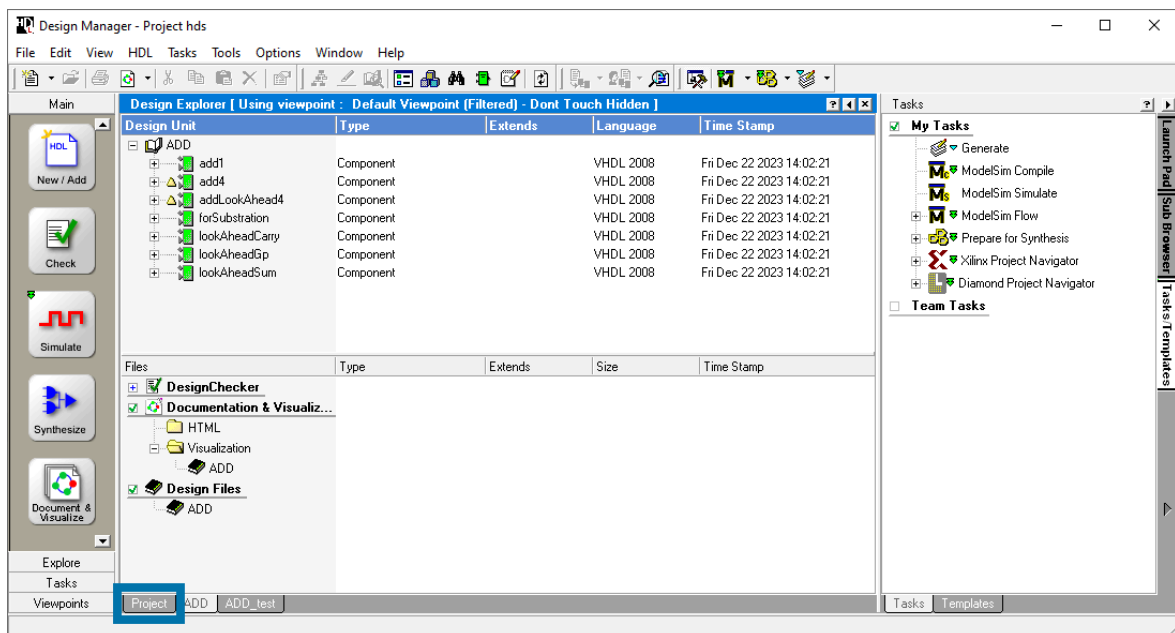


Abbildung 4 - HDL Designer

- Doppelklicken Sie auf die für das Labor relevanten Bibliotheken, um sie zu öffnen. Jedes Labor besitzt 2 Bibliotheken welche mit der Abkürzung des Labors einhergeht. In diesem Labo „IND“ sind es die Bibliotheken **IND** und **IND\_test**.



Die Dateien dürfen nicht auf den Laborcomputern belassen werden. Diese müssen entweder auf einem USB-Stick oder auf den \\filer01\... gespeichert werden. Teilen Sie ausserdem die Dateien mit Ihrem Laborpartner damit dieser die Arbeit fortsetzen kann.



### 3 Kombinatorische Logikschaltung

Kommen wir nun zur eigentlichen Schaltung die realisiert werden muss. Ein Unternehmen hat beschlossen, jeden seiner Einkäufe nach strengen Regeln zu überprüfen. Ein Artikel darf nur dann gekauft werden, wenn mindestens eine der folgenden Bedingungen erfüllt ist:

- **Bedingung A:** die Auftragsbücher des Unternehmens sind voll, die Lieferzeit des Materials ist kurz und die Lagerbestände des Unternehmens sind niedrig
- **Bedingung B:** die Auftragsbücher sind nicht voll, aber der Preis und die Lagerbestände des Unternehmens sind niedrig
- **Bedingung C:** der Preis ist hoch, aber die Auftragsbücher sind voll und die Lagerbestände des Unternehmens sind niedrig
- **Bedingung D:** die Lagerbestände des Unternehmens sind niedrig und die Lieferzeit ist kurz
- **Bedingung E:** die Lieferzeit ist lang, aber der Preis ist niedrig



Ergänzen Sie die Wahrheitstabelle Tabelle 1 laut Pflichtenheft.



Leiten Sie von der Tabelle eine Schaltung ab welche aus den Grundgattern UND, ODER sowie Inverter besteht.

#### 3.1 Entwurf

Öffnen Sie die Schaltung unter **IND**  $\Rightarrow$  **selection**. Die Eingangs- und Ausgangssignale sind bereits definiert, ebenso wie die Zwischenbedingung A (**condition A**).

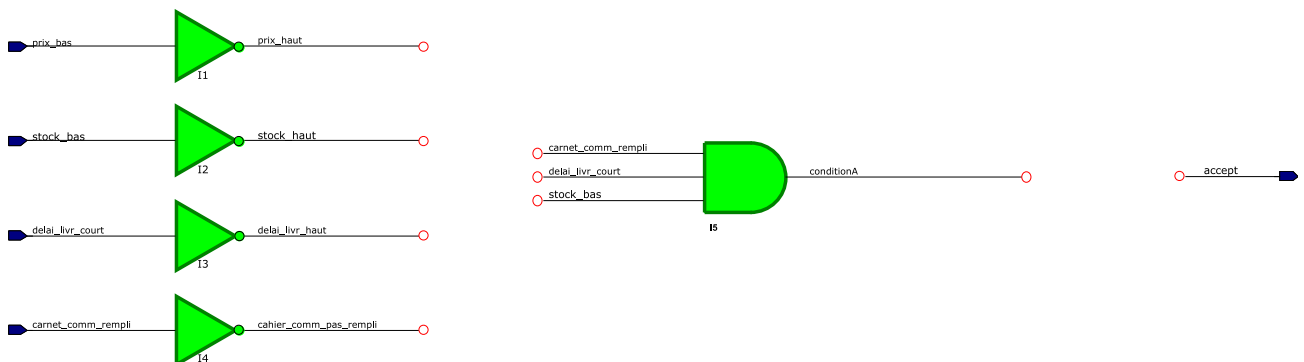
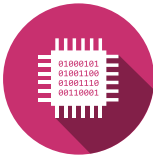


Abbildung 5 - Bestehende Schaltung zum Komplettieren



Auftrags- bücher voll				Bedingung					Akzeptiert
				A	B	C	D	E	

Tabelle 1 - Wahrheitstabelle



## 4 | Schaltungs-Editor

Dieser Abschnitt soll Sie durch die Erstellung einer digitalen Schaltung führen.

Mit Hilfe Invertern, UND- und ODER-Gattern zeichnen Sie die Schaltung, welche Sie aus der Wahrheitstabelle Tabelle 1 herausgelesen haben.



Detaillierte Informationen wie man HDL-Designer sowie ModelSim benutzt finden Sie zusätzlich im Dokument **DiD-EDA-UserGuide-en.pdf** auf Cyberlearn [1].

### 4.1 Komponenten

Komponenten sind logische Gatter, Konstante Signale wie logic\_0 / logic\_1, Multiplexer, Flip-Flops ... sowie die eigens erstellten Blöcke.

- Fügen Sie eine Komponente hinzu, indem Sie auf die Schaltfläche **Komponente hinzufügen** klicken.

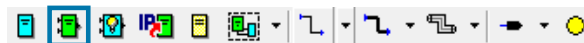


Abbildung 6 - Komponente hinzufügen

- Suchen Sie die gewünschte Komponente in der Bibliothek **gates**.
- Wählen Sie die Komponente aus und ziehen Sie sie in die Schaltung.



Fügen Sie alle für Ihren Schaltkreis erforderlichen Komponenten hinzu.

### 4.2 Signale

Signale dienen dazu, die Eingänge und Ausgänge der verschiedenen Blöcke miteinander zu verbinden. Sie können ein Bit (**std\_ulogic**) oder mehrere Bits (**std\_ulogic\_vector**, **signed**, **unsigned**) repräsentieren. Die Signale können durch [?] Linien oder durch Ihre Namen verbunden werden.

#### 4.2.1 Hinzufügen eines Signals

- Fügen Sie ein Signal hinzu, indem Sie auf den Knopf **Add Signal** klicken.



Abbildung 7 - Signal hinzufügen

- Nach dem Klicken auf die Schaltfläche kann ein neues Signal erstellt werden, indem Sie auf den Schaltplan klicken.
- Um ein neues Signal zu bestätigen, verbinden Sie einen Eingang und einen Ausgang miteinander oder doppelklicken Sie, um die Erstellung des Signals abzuschliessen.
- Mit der **ESC** Taste können Sie die Erstellung eines Signals abbrechen.



### 4.2.2 Ändern des Signalnamens

Doppelklicken Sie auf ein Signal, um seinen Namen zu ändern. Der Name des Signals kann im **Textfeld** geändert werden.

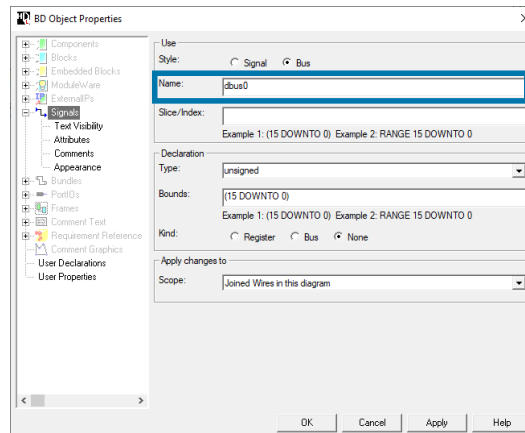


Abbildung 8 - Eigenschaften von Signalen und Bussen bearbeiten



Signale mit dem gleichen Namen sind miteinander verbunden! Signale mit unterschiedlichen Namen müssen über Buffer miteinander verbunden werden.

## 4.3 Aufgaben

Erstellen Sie Ihre Schaltung klar und lesbar.



- Benennen Sie jedes Signal im Schaltkreis.
- Stellen Sie die Anzeigeeinstellungen für die Signalnamen ein.
- Machen Sie die Schaltung lesbar, indem Sie einige Signale nach Namen und nicht mit einem Draht miteinander verbinden.



Es ist wichtig, den Namen eines Signals immer anzuzeigen. Eine falsche Manipulation ändert den Namen teilweise automatisch und dies ist ohne den Signalnamen nicht erkennbar.



## 4.4 Layout

Die Schaltung, wenn sie ordnungsgemäss ausgefüllt ist, kann gedruckt und/oder als PDF exportiert werden.

Zuerst:



Füllen Sie den verfügbaren Rahmen aus.



Führen Sie alle erforderlichen Änderungen durch, um diesen Schaltplan so lesbar wie möglich zu gestalten.

Für das Erstellen des PDF oder des Druckes. Gehen Sie zu **Datei** ⇒ **Seite einrichten** und Verwenden Sie die folgende Konfiguration:

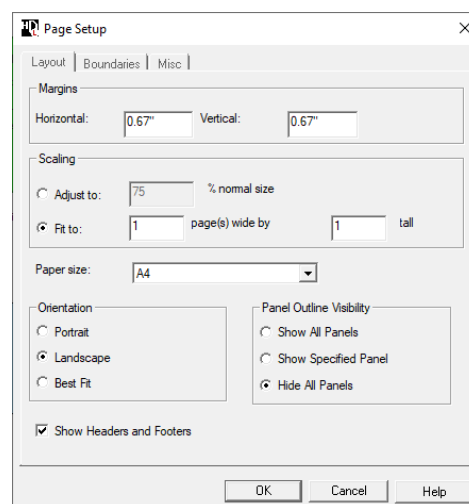


Abbildung 9 - Seitenkonfiguration

- Drucken Sie den Schaltplan auf einem physischen Drucker oder wählen Sie **Adobe PDF** oder **Microsoft print to PDF** aus, um die Seite als **PDF** zu exportieren.



Exportieren Sie Ihren Schaltplan als PDF.



## 5 | Simulation

Um sicherzustellen, dass die Schaltung korrekt funktioniert, wird eine Simulation durchgeführt, die alle möglichen Kombinationen der Eingangssignale testet.

### 5.1 Testbench

Die Simulation befindet sich unter **IND\_test**  $\Rightarrow$  **IND\_tb**.

Der **Testbench** (Teststand) besteht aus zwei Teilen:

- Der zu testende Schaltkreis (**DUT - Device Under Test**), der zuvor entwickelten Schaltung
- Der Tester, der die Eingangssignale für den DUT generiert und die Ausgänge überprüft.

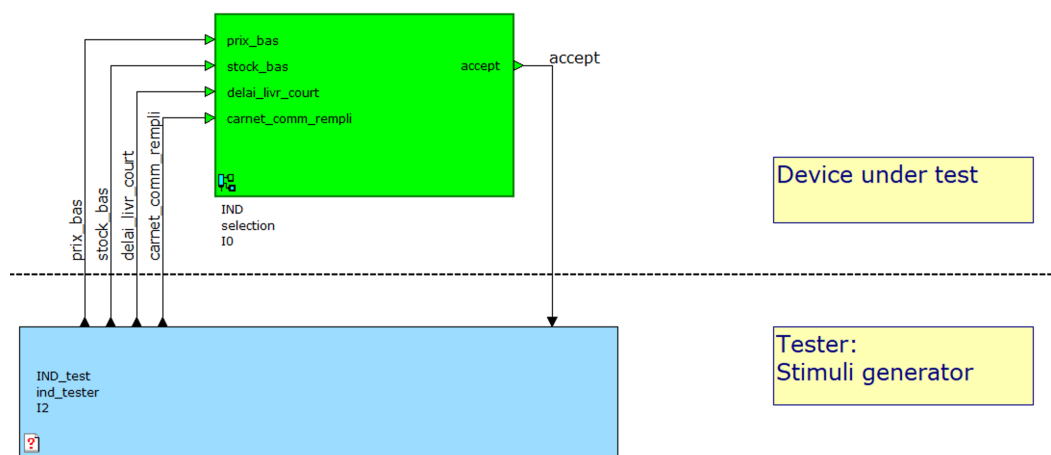


Abbildung 10 - Generate VHDL

### 5.2 Simulator

Klicken Sie auf die Schaltfläche **Generate and run entire ModelSim flow (Through Components)**, um die Simulation zu starten.

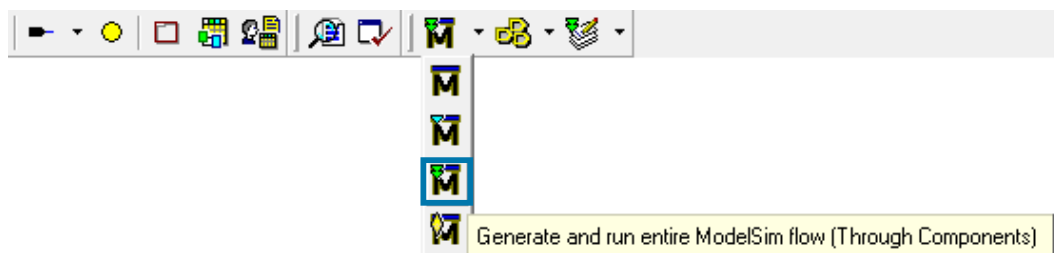


Abbildung 11 - Generate VHDL



Wenn die Simulation gestartet wird, darf kein Block in der Schaltung ausgewählt sein.

- Wenn die Generierung und Kompilierung erfolgreich sind, wird das folgende Fenster angezeigt:

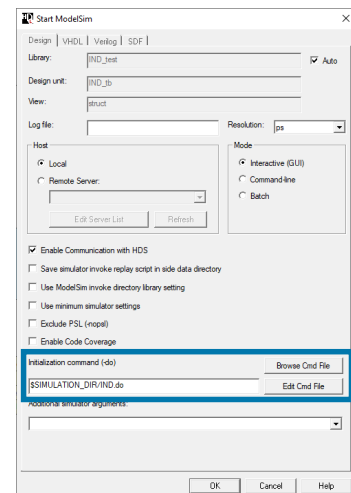




- Geben Sie die **Initialisierungs-Befehlsdatei (\*.do)** ein, in diesem Fall **\$SIMULATION\_DIR/IND.do**. Diese Datei enthält die Informationen der anzuzeigenden Signale.
- Klicken Sie auf **OK**.



Die **\*.do**-Dateien befinden sich im Verzeichnis **Simulation**, das sich im Stammverzeichnis von **did-labs** befindet.



Wenn nichts passiert, wenn die Simulation gestartet wird, überprüfen Sie das Log-Fenster. Falls das Log-Fenster nicht angezeigt wird:

- Klicken Sie mit der rechten Maustaste auf das **Log Window**.
- Klicken Sie auf **Maximize**.

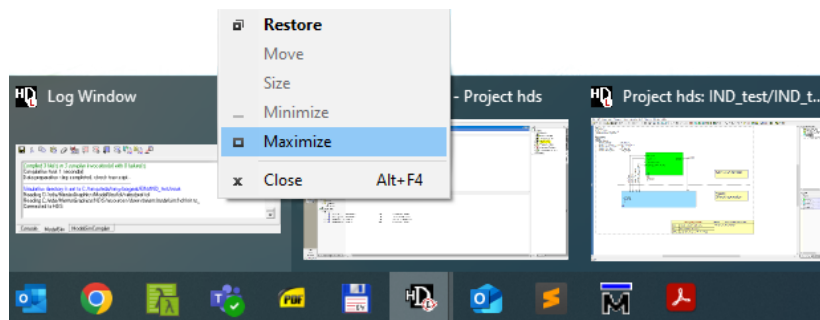


Abbildung 12 - Log-Fenster anzeigen

Dieses Fenster zeigt Probleme im Zusammenhang mit der Kompilierung der Schaltung an, die behoben werden müssen, um die Simulation zu starten.

Fehler können in **Rot**, **Grün** oder **Schwarz** angezeigt werden.



Starten Sie das Simulationsprogramm mit der Datei **\$SIMULATION\_DIR/IND.do**.

### 5.2.1 Simulation starten

Die Simulation wird durch den Tester-Block mit Werten „stimuliert“.



Öffnen Sie die Datei der Testers und bestimmen Sie die erforderliche Simulationszeit.



- Geben Sie die Simulationszeit in das **Textfeld** ein.
- Klicken Sie auf die Schaltfläche **Start simulation**.
- Mit der Schaltfläche **Reset simulation** können Sie die Simulation zurücksetzen.



Starten Sie die Simulation und überprüfen Sie, ob die Ein- und Ausgangssignale korrekt sind.

### 5.3 Anzeige

Die verschiedenen Elemente des Simulationsprogrammes **Mentor ModelSim** sind in Abbildung 13 dargestellt:

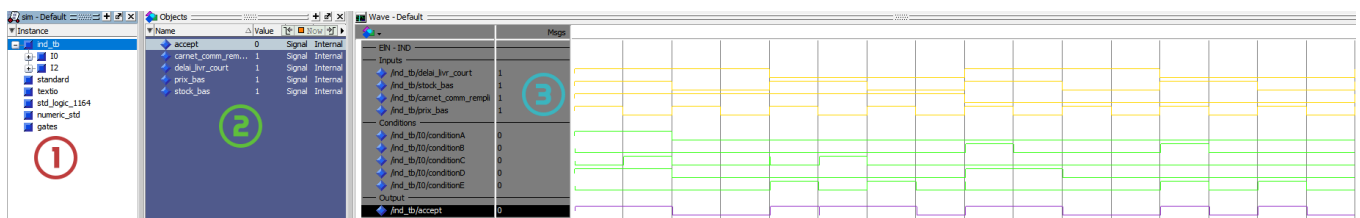


Abbildung 13 - Simulation mit nur Bedingung A

1. Hierarchie der Schaltung, jedes einzelnen Blockes Fenster 1. Durch Auswahl eines Blocks werden die zugehörigen Signale im Fenster 2 angezeigt.
2. Zeigt die verfügbaren Signale im ausgewählten Block an. Sie können in Fenster 3 gezogen werden (Drag&Drop), um sie anzuzeigen.
3. Das **Wave** Simulationsfenster mit den Signalen der Schaltung.

#### 5.3.1 Hinzufügen von Signalen

Das Hinzufügen von Signalen erfolgt wie zuvor beschrieben.

Nach dem Hinzufügen von Signalen ist ein Neustart der Simulation erforderlich, um sie anzuzeigen.



Fügen Sie die Signale der verbleibenden 4 Bedingungen in das Simulationsfenster ein.

#### 5.3.2 Zeitachse

Durch einen Rechtsklick auf die Zeitachse  $\Rightarrow$  **Grid, Timeline & Cursor control** können Sie die Zeitachse und das Zeitmarkierungsraster ändern.



Ändern Sie die Zeitachse so, dass sie in Nanosekunden angezeigt wird.



### 5.3.3 Funktionskontrolle

Um die Funktionalität des Schaltkreises zu überprüfen, müssen die Signale miteinander korreliert werden.

In einigen Labors kann das **Transcript**-Fenster am unteren Rand von **ModelSim** auch Informationen über automatisierte Tests anzeigen.



Überprüfen Sie, ob Ihre Simulation mit Ihrer Wahrheitstabelle Tabelle 1 übereinstimmt.

### 5.3.4 Speichern des Kontexts

Um die Simulation mit den vorherigen Einstellungen erneut öffnen zu können, können Sie die Konfiguration speichern, indem Sie im **Wave**-Fenster  $\Rightarrow$  **Ctrl + s** Drücken.

Die generierte **.do**-Datei kann im **Simulation**-Ordner des Projekts gespeichert werden.



Speichern Sie die Konfiguration der Simulation und überschreiben Sie die aktuelle Datei **IND.do**.

### 5.3.5 Drucken der Simulation

Wie beim Schaltplan können die Simulationssignale gedruckt oder als PDF über **File**  $\Rightarrow$  **Print...** exportiert werden.

Eine empfohlene Druckkonfiguration wird in Tabelle 2 dargestellt:

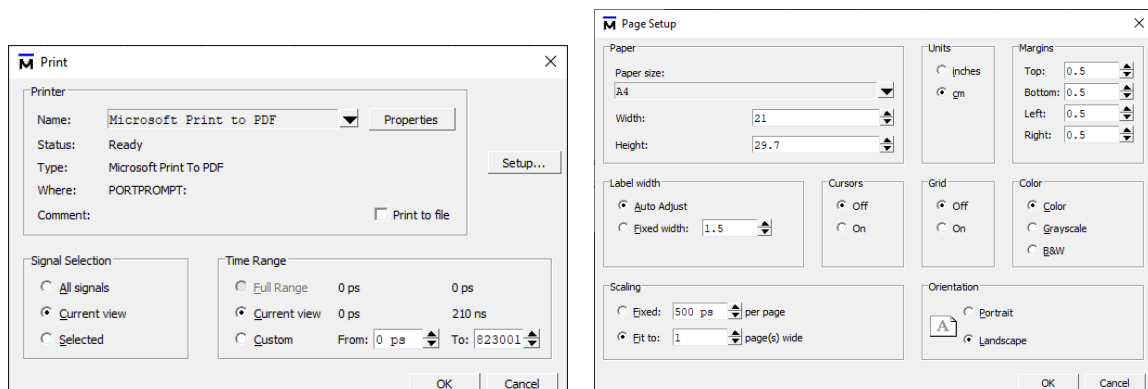


Tabelle 2 - Empfohlene Druckkonfiguration für die Simulation



Exportieren Sie das Ergebnis der Simulation als PDF.



## 5.4 Störung

Störungen (Glitches) können auf dem Ausgangssignal auftreten. Eine dieser Störungen wird in Abbildung 14 hervorgehoben.

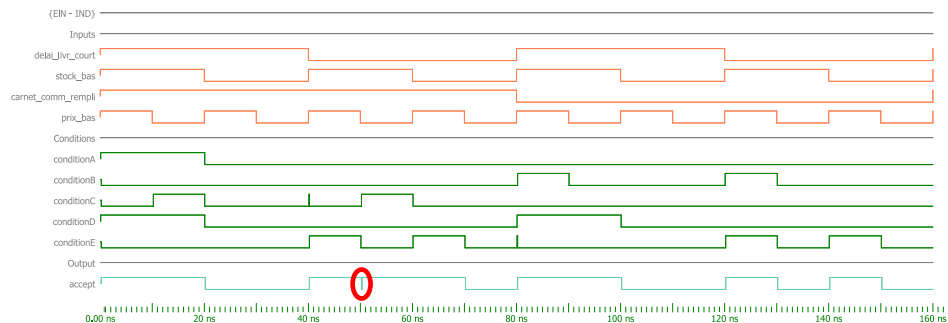


Abbildung 14 - Störung auf dem Ausgangssignal



- Finden Sie diese Störungen in Ihrer Simulation.
- Erklären Sie den Grund für diese Störungen.



## 6 | Checkout

Bevor Sie das Labor verlassen versichern Sie sich das sie folgende Tätigkeiten durchgeführt haben:



# Glossar

*EDA* – Electronic Design Automation [1](#)



# Literatur

- [1] A. A. Silvan, Zahno Rémy, Borgeat, „DiD EDA Userguide“. 2024.