

# Praktikum "Grundlagen Technische Informatik"

Hochschule Osnabrück  
Fakultät Ingenieurwissenschaften und Informatik  
Labor für Digital- und Mikroprozessortechnik  
Prof. Dr.-Ing. M. Weinhardt, Dipl.-Inf. (FH) R. Höckmann

WS20/21

## Versuch 1: Simulation einfacher Logik-Gatter und Normalformen einfacher Schaltungen

In diesem Versuch wird der prinzipielle Aufbau digitaler Schaltungen thematisiert und es werden die Normalformen einfacher Schaltungen geübt. Gleichzeitig erfolgt eine Einführung in das Simulationsprogramm „Logisim-Evolution“.

### Vorbereitung (Vor dem Praktikum)

- Installieren Sie Logisim-Evolution gemäß der separaten Anleitung „Einrichten der Entwicklungsumgebung“ auf Ihrem Rechner
- Stellen Sie die Wahrheitstabellen für ein NAND-, ein NOR- und ein XOR-Gatter auf. Eine Vorlage dafür finden Sie im Anhang
- Ermitteln Sie aus der Wahrheitstabelle des XOR-Gatters die disjunktive Normalform und formen Sie diese so um, dass sie mit einer Schaltung realisiert werden kann, welche nur aus NAND- und Inverter-Gattern besteht. (**Lösungen ohne Herleitung werden nicht akzeptiert!**)

Verwenden Sie für die Ausarbeitungen die vorgegebenen Deckblätter. Ergänzen Sie darauf Ihre Namen und die Gruppenbezeichnung. Nummerieren Sie die Teilaufgaben entsprechend der Aufgabenstellung. Saubere handschriftliche Lösungen sind zulässig. Achten Sie bitte auf die Lesbarkeit und verwenden Sie ein Lineal.

### Aufgabe 1: Simulation von Logik-Gattern mit Logisim-Evolution

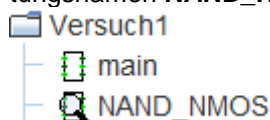
Logisim-Evolution ist eine Software zur Simulation digitaler Schaltungen für den Einsatz in der Lehre. Für die Entwicklung realer Schaltungen werden andere Simulationsprogramme eingesetzt.





- a) Starten Sie Logisim-Evolution. Beim ersten Start sollten Sie folgende Einstellungen im Menü (File → Preferences → International) setzen: Gate shape: IEC und Language: Deutsch / German. Zunächst wird eine einfache Schaltung bestehend aus zwei Eingängen, einem NAND-Gatter und einem Ausgang aufgebaut. Während der Erstellung einer Schaltung verwenden Sie das Bearbeitungswerkzeug (↻). Fügen Sie der Schaltung zwei Eingangs-Pins (I↻), ein NAND-Gatter (⋈, Rubrik: Gatter), sowie einen Ausgangs-Pin (↻1) hinzu. Durch Doppelklick auf die Bauteile in der Schaltung können Sie ihnen einen Namen zuweisen (Eingänge: „X0“ und „X1“, Ausgang: „Y“). Verbinden Sie die Bauteile durch Klicken und Ziehen mit der Maus miteinander, so dass sich die folgende Schaltung ergibt:

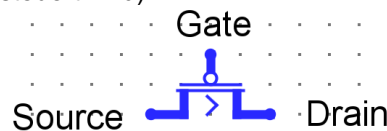


Um die Eingangswerte zu ändern, wählen Sie das Hand-Werkzeug (☞) und klicken damit auf die Eingangs-Pins. Vergleichen Sie für alle vier möglichen Eingangskombinationen den Ausgangswert der Schaltung mit dem von Ihnen erwarteten Wert gemäß der Wahrheitstabelle des NAND-Gatters. Speichern Sie das Projekt unter dem Dateinamen **Versuch1.circ** ab (Datei → Speichern).


- b) In der Vorlesung wurden mehrere Varianten zur Realisierung eines NAND-Gatters vorgestellt. Zunächst soll das Gatter in NMOS-Technologie aufgebaut werden. Fügen Sie dazu dem Projekt eine neue Schaltung hinzu (Projekt → Schaltung hinzufügen ...). Verwenden Sie als Schaltungsnamen **NAND\_NMOS**:



Die mit der Lupe () markierte Schaltung ist die jeweils im Bearbeitungsfenster dargestellte Schaltung. Der Wechsel erfolgt durch Doppelklick. Bauen Sie nun die Schaltung auf. Neben zwei Eingangs- und einem Ausgangs-Pin benötigen Sie dazu die Bauteile Pull-Widerstand (, Rubrik Verdrahtung), Masse (, Rubrik: Verdrahtung) und zwei Transistoren (, Rubrik: Verdrahtung). Stellen Sie für den Pull-Widerstand die benötigte Pull-Richtung ein, indem Sie den Widerstand anklicken und dann in der Attributtabelle unten links den entsprechenden Wert auswählen. Dort stellen Sie für die Transistoren auch den benötigten Typ (P oder N), die Ausrichtung (d.h. die Lage des Drain-Ausgangs) sowie die Position des Gate-Eingangs ein (Attribut „Position der Eingänge“). Eine Besonderheit von Logisim-Evolution: Der am Source-Eingang anliegende Wert wird nur in der Pfeilrichtung zum Drain-Ausgang übertragen (natürlich nur, wenn der Drain-Eingang angesteuert wird):

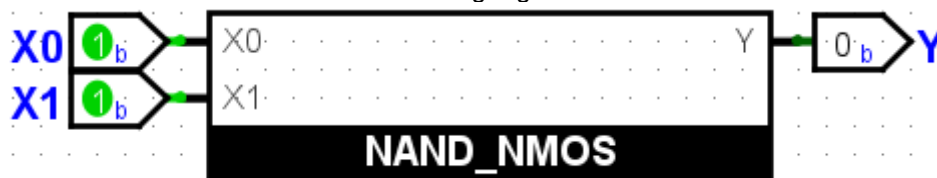


Nachdem Sie alle Bauteile miteinander verbunden und die Pins beschriftet haben (Eingänge: X0 und X1, Ausgang: Y), müssen Sie für alle vier möglichen Eingangskombinationen den Ausgangswert kontrollieren.


**Ausarbeitung:** Fügen Sie Ihre Namen sowie das Datum der Versuchsdurchführung als Text () in die Schaltung **NAND\_NMOS** ein und exportieren Sie die Schaltung als Bild (Datei→Bild exportieren). Das Bild fügen Sie in Ihre Ausarbeitung ein. Beispiel:

William Shockley  
John Bardeen  
Walter Brattain  
23.12.1947

- c) Nun soll in der Schaltung **main** das Bibliotheks-NAND-Gatter durch das selbst erstellte NAND-Gatter ersetzt werden. Aktivieren Sie dazu durch Doppelklick die Schaltung **main**. Löschen Sie dort die Instanz des NAND-Gatters sowie die Verbindungsleitungen. Klicken Sie nun auf den Namen der Schaltung **NAND\_NMOS** und ziehen Sie eine neue Instanz davon mit der Maus ins Bearbeitungsfenster. An der Zielposition müssen Sie noch einmal klicken. Verbinden Sie nun die Pins mit dem Gatter und testen Sie alle Eingangskombinationen wie zuvor:



**Ausarbeitung:** Wie zuvor: Ihre Namen und das Datum in die Schaltung einfügen und exportiertes Bild in die Ausarbeitung einfügen.

- d) Erstellen Sie nun eine weitere Schaltung (Name: **NAND\_CMOS**). In dieser bauen Sie nun das NAND-Gatter in CMOS-Logik auf. Dazu brauchen Sie als neues Bauteil die Versorgungsspannung (, Rubrik: Verdrahtung).

**Ausarbeitung:** Wie zuvor: Ihre Namen und das Datum in die Schaltung einfügen und exportiertes Bild in die Ausarbeitung einfügen.

- e) Erstellen Sie in einer neuen Schaltung **NOR\_CMOS** ein NOR-Gatter in CMOS-Technik.

**Ausarbeitung:** Wie zuvor: Ihre Namen und das Datum in die Schaltung einfügen und exportiertes Bild in die Ausarbeitung einfügen.

- f) Erstellen Sie in einer neuen Schaltung **INV\_CMOS** ein Inverter-Gatter in CMOS-Technik.

**Ausarbeitung:** Wie zuvor: Ihre Namen und das Datum in die Schaltung einfügen und exportiertes Bild in die Ausarbeitung einfügen.

- g) Erstellen Sie in einer neuen Schaltung **XOR\_CMOS** ein XOR-Gatter. Verwenden Sie dabei nur die von Ihnen erzeugten Gatter **INV\_CMOS** und **NAND\_CMOS**. Die dafür notwendigen Gleichungen haben Sie bereits in der Vorbereitung aufgestellt.

**Ausarbeitung:** Wie zuvor: Ihre Namen und das Datum in die Schaltung einfügen und exportiertes Bild in die Ausarbeitung einfügen.

## Aufgabe 2: Normalformen einfacher Schaltungen

In dieser Aufgabe werden aus den Wahrheitstabellen boolescher Funktionen die gefragten Normalformen bestimmt.

Die folgende Tabelle enthält mehrere kombinatorische Schaltungen mit drei Eingängen (X2-X0) und jeweils einem Ausgang (Y4–Y0). Erfragen Sie, welche Schaltung Sie analysieren sollen.

**Ausarbeitung:** Geben Sie an, welche Schaltung Sie analysieren. Bestimmen Sie die konjunktive Normalform und die disjunktive Normalform dieser Schaltung.

X2	X1	X0	Y4	Y3	Y2	Y1	Y0
0	0	0	1	0	0	1	1
0	0	1	0	0	0	0	0
0	1	0	0	0	0	1	0
0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	0
1	0	1	1	0	1	0	0
1	1	0	0	1	1	1	0
1	1	1	1	1	1	0	1

**Anhang: Deckblatt für die Ausarbeitung zu Versuch 1**

Teilnehmer	Gruppe Nr.:
	A3

**Wahrheitstabellen für die Vorbereitung****NAND:**

$X1$	$X0$	$\overline{X1 \wedge X0}$
0	0	
0	1	
1	0	
1	1	

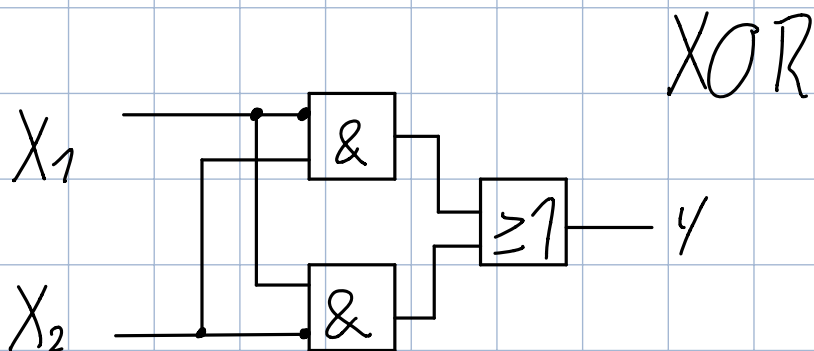
**NOR:**

$X1$	$X0$	$\overline{X1 \vee X0}$
0	0	
0	1	
1	0	
1	1	

**XOR:**

$X1$	$X0$	$X1 \oplus X0$
0	0	
0	1	
1	0	
1	1	

# Vorarbeit



$$(\overline{X_1} \wedge X_2) \vee (X_1 \wedge \overline{X_2}) = Y$$

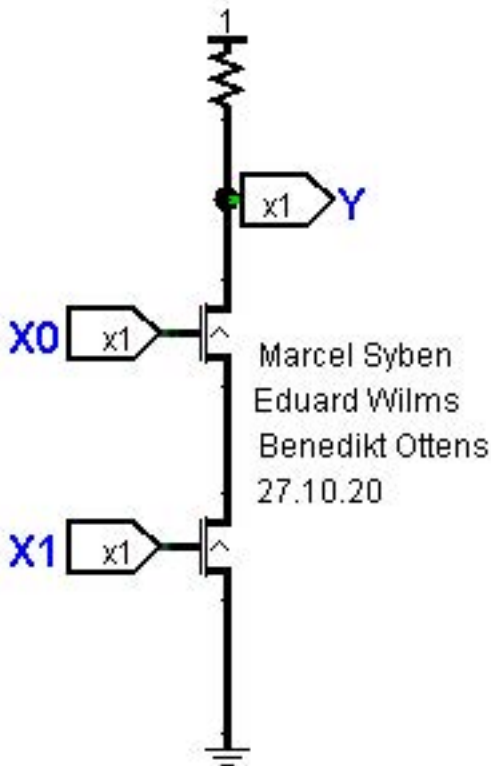
$$\overline{(\overline{X_1} \wedge X_2) \wedge (X_1 \wedge \overline{X_2})} = Y$$

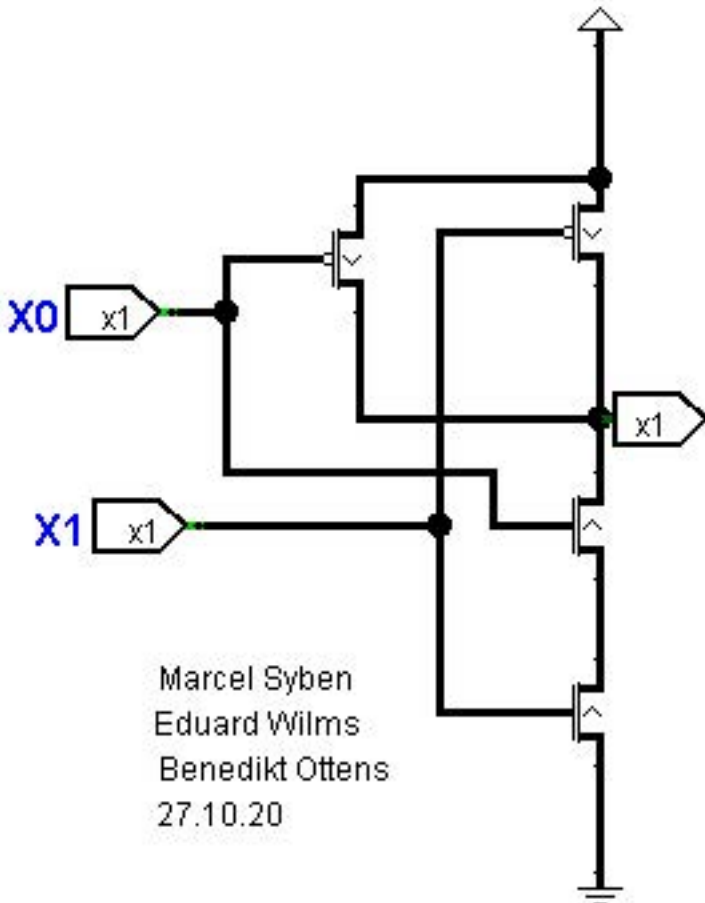
2.

$$(\overline{X_2} \wedge X_1 \wedge X_0) \vee (X_2 \wedge \overline{X_1} \wedge \overline{X_0}) \vee (X_2 \wedge \overline{X_1} \wedge X_0) \vee$$

$$(X_2 \wedge X_1 \wedge \overline{X_0}) \vee (X_2 \wedge X_1 \wedge X_0)$$

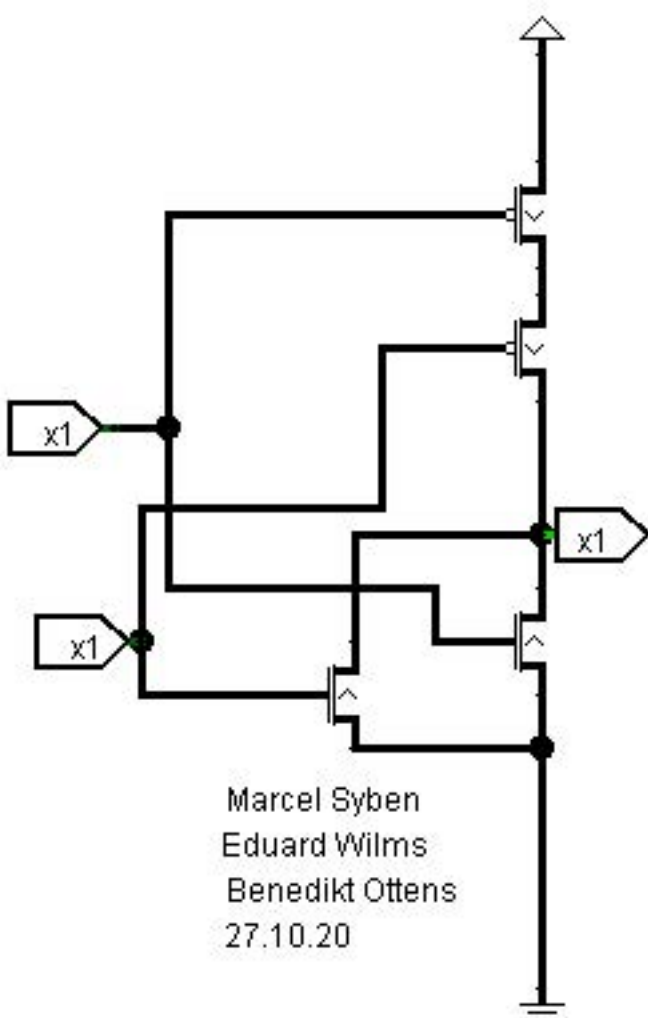
$$(X_2 \vee X_1 \vee X_0) \wedge (X_2 \vee X_1 \vee \overline{X_0}) \wedge (X_2 \vee \overline{X_1} \vee X_0)$$





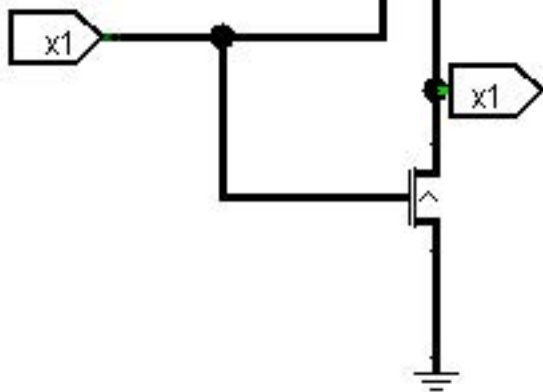
Marcel Syben  
Eduard Wilms  
Benedikt Ottens  
27.10.20





Marcel Syben  
Eduard Wilms  
Benedikt Ottens  
27.10.20

Marcel Syben  
Eduard Wilms  
Benedikt Ottens  
27.10.20



Marcel Syben  
Eduard Wilms  
Benedikt Ottens  
27.10.20

