

# Verkabelter Multiplizierer

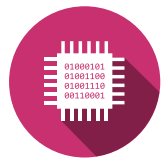
## Labor Digital Design

## Inhalt

1 Ziel .....	1
2 Multiplizierer für natürliche Zahlen .....	2
2.1 Algorithmus .....	2
2.2 Analyse .....	2
2.3 Schaltung .....	2
2.4 Erstellung .....	3
3 Multiplizierer für Arithmetische Zahlen .....	4
3.1 Algorithmus .....	4
3.2 Analyse .....	4
3.3 Erstellung .....	4
4 Analyse .....	5

## 1 | Ziel

In diesem Labor wird der Entwurf von iterativen arithmetischen Schaltungen anhand von kombinatorischen Logikgattern geübt. Das Labor zeigt die Realisierungstechnik von Multiplizierern für natürliche wie auch für ganze Zahlen.



## 2 | Multiplizierer für natürliche Zahlen

### 2.1 Algorithmus

Abbildung 1 stellt den Algorithmus zur Multiplikation von 2 Zahlen von je 4 Ziffern dar. Das Produkt ist gegeben durch die Summe von Teilprodukten. Die Teilprodukte werden erstellt durch die Multiplikation von einer der Zahlen durch eine Ziffer der anderen Zahl.

				a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>
				× b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>
				b <sub>0</sub> *a <sub>3</sub>	b <sub>0</sub> *a <sub>2</sub>	b <sub>0</sub> *a <sub>1</sub>	b <sub>0</sub> *a <sub>0</sub>
			b <sub>1</sub> *a <sub>3</sub>	b <sub>1</sub> *a <sub>2</sub>	b <sub>1</sub> *a <sub>1</sub>	b <sub>1</sub> *a <sub>0</sub>	
		b <sub>2</sub> *a <sub>3</sub>	b <sub>2</sub> *a <sub>2</sub>	b <sub>2</sub> *a <sub>1</sub>	b <sub>2</sub> *a <sub>0</sub>		
	b <sub>3</sub> *a <sub>3</sub>	b <sub>3</sub> *a <sub>2</sub>	b <sub>3</sub> *a <sub>1</sub>	b <sub>3</sub> *a <sub>0</sub>			
p <sub>7</sub>	p <sub>6</sub>	p <sub>5</sub>	p <sub>4</sub>	p <sub>3</sub>	p <sub>2</sub>	p <sub>1</sub>	p <sub>0</sub>

Abbildung 1: Multiplikationsalgorithmus

### 2.2 Analyse

Für die Multiplikation von 2 mit 4 Bits codierten natürlichen Zahlen (unsigned), bestimmen Sie den Binärwert des grösstmöglichen Resultates. Schliessen Sie daraus die Anzahl benötigter Bits für das Produkt von 2 natürlichen Zahlen, welche mit  $n_1$ , respektiv mit  $n_2$  Bits codiert sind.

### 2.3 Schaltung

Abbildung 2 zeigt die Schaltung eines Multiplizierers, welcher nach dem oben angegebenen Algorithmus arbeitet.

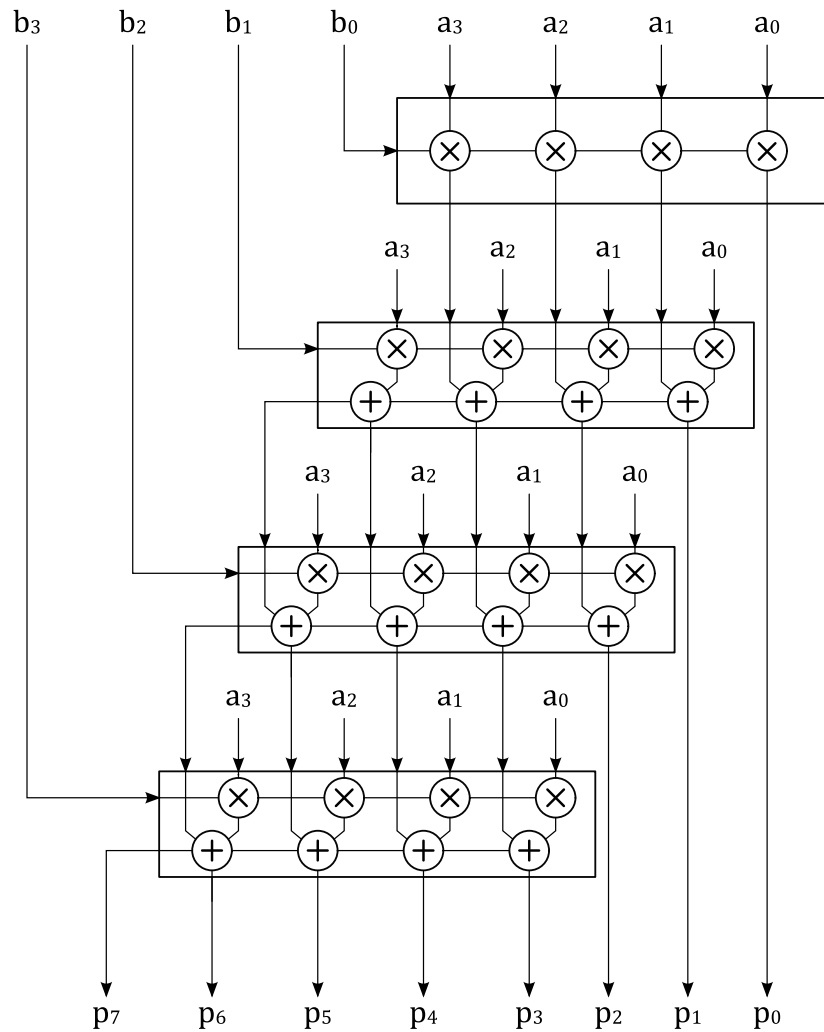
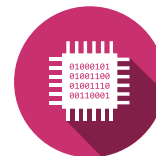


Abbildung 2: Architektur des Multiplizierers

## 2.4 Erstellung

Mit Hilfe von INV, UND, ODER und XOR Gattern, ergänzen Sie das hierarchische Schema des Multiplizierers der Abbildung 2 und überprüfen Sie seine Funktionalität.



## 3 | Multiplizierer für Arithmetische Zahlen

### 3.1 Algorithmus

Abbildung 3 stellt den Algorithmus von Baugh-Wooley zur Multiplikation von zwei im Zweier-Komplement codierten arithmetischen Zahlen (signed) mit derselben Anzahl an Bits dar.

				a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>
				× b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>
			1	<u>b<sub>0</sub>*a<sub>3</sub></u>	b <sub>0</sub> *a <sub>2</sub>	b <sub>0</sub> *a <sub>1</sub>	b <sub>0</sub> *a <sub>0</sub>
		<u>b<sub>1</sub>*a<sub>3</sub></u>	b <sub>1</sub> *a <sub>2</sub>	b <sub>1</sub> *a <sub>1</sub>	b <sub>1</sub> *a <sub>0</sub>		
	<u>b<sub>2</sub>*a<sub>3</sub></u>	b <sub>2</sub> *a <sub>2</sub>	b <sub>2</sub> *a <sub>1</sub>	b <sub>2</sub> *a <sub>0</sub>			
1	b <sub>3</sub> *a <sub>3</sub>	<u>b<sub>3</sub>*a<sub>2</sub></u>	<u>b<sub>3</sub>*a<sub>1</sub></u>	<u>b<sub>3</sub>*a<sub>0</sub></u>			
p <sub>7</sub>	p <sub>6</sub>	p <sub>5</sub>	p <sub>4</sub>	p <sub>3</sub>	p <sub>2</sub>	p <sub>1</sub>	p <sub>0</sub>

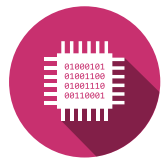
Abbildung 3: Multiplikationsalgorithmus für Zahlen im Zweier-Komplement

### 3.2 Analyse

Für die Multiplikation von 2 mit 4 Bits codierten ganzen Zahlen, bestimmen Sie den minimalen und den maximalen Wert des Resultates. Schliessen Sie daraus die Anzahl benötigter Bits für das Produkt von 2 natürlichen Zahlen, welche mit  $n_1$ , respektiv mit  $n_2$  Bits codiert sind.

### 3.3 Erstellung

Ergänzen Sie das hierarchische Schema des Multiplizierers der Abbildung 2 mit Hilfe von kombinatorischen Logikgattern und überprüfen Sie seine Funktionalität.



## 4 | Analyse

Unter der Annahme, dass alle Logikgatter dieselbe Verzögerung von 1 ns vorweisen, bestimmen Sie die maximale Berechnungsverzögerung der erstellten Operatoren.

Schlagen Sie eine andere Struktur vor, um die Geschwindigkeit dieser Operatoren zu vergrößern.