

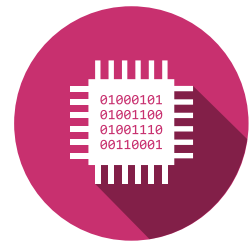
Verkabelter Multiplizierer

Inhaltsverzeichnis

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	4

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_3	a_2	a_1	a_0
				$\times b_3$	b_2	b_1	b_0
				$b_0 \cdot a_3$	$b_0 \cdot a_2$	$b_0 \cdot a_1$	$b_0 \cdot a_0$
			$b_1 \cdot a_3$	$b_1 \cdot a_2$	$b_1 \cdot a_1$	$b_1 \cdot a_0$	
		$b_2 \cdot a_3$	$b_2 \cdot a_2$	$b_2 \cdot a_1$	$b_2 \cdot a_0$		
	$b_3 \cdot a_3$	$b_3 \cdot a_2$	$b_3 \cdot a_1$	$b_3 \cdot a_0$			
p_7	p_6	p_5	p_4	p_3	p_2	p_1	p_0

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.

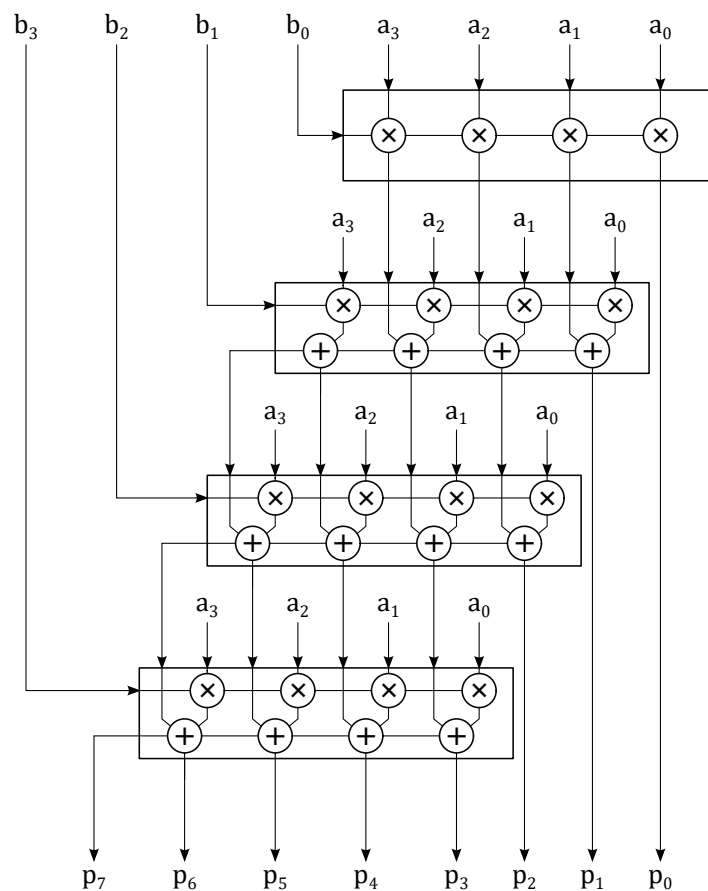
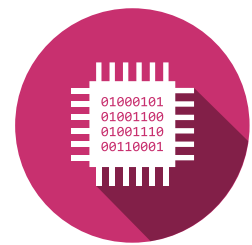
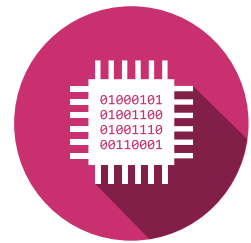


Abbildung 2: Architektur des Multiplizierers

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.

$$\begin{array}{r}
 \begin{array}{cccc}
 & & a_3 & a_2 & a_1 & a_0 \\
 & & \times b_3 & b_2 & b_1 & b_0 \\
 \hline
 & & \overline{b_0 \cdot a_3} & b_0 \cdot a_2 & b_0 \cdot a_1 & b_0 \cdot a_0 \\
 & 1 & \overline{b_1 \cdot a_3} & b_1 \cdot a_2 & b_1 \cdot a_1 & b_1 \cdot a_0 \\
 & \overline{b_2 \cdot a_3} & \overline{b_2 \cdot a_2} & b_2 \cdot a_1 & b_2 \cdot a_0 & \\
 1 & b_3 \cdot a_3 & \overline{b_3 \cdot a_2} & \overline{b_3 \cdot a_1} & \overline{b_3 \cdot a_0} & \\
 \hline
 p_7 & p_6 & p_5 & p_4 & p_3 & p_2 & p_1 & p_0
 \end{array}
 \end{array}$$

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.