ALBERT-LUDWIGS-UNIVERSITÄT FREIBURG

INSTITUT FÜR INFORMATIK

Prof. Dr. Bernd Becker
Dipl. Inf. Thomas Eschbach

Freiburg, 30. April 2002

Übungsblatt 02 zur Vorlesung

Technische Informatik II

Aufgabe 1

Unter der Webadresse

http://mopo.informatik.uni-freiburg.de/umfrage/frageb_mopo.html

finden Sie einen Fragebogen zum Mopo-Projekt, d.h. zu den über die Fakultät erworbenen Notebooks bzw. dem FunkLAN.

Ihre Teilnahme an der Umfrage wird anhand Ihrer Matrikelnummer protokolliert, d.h. jeder (*) von Ihnen füllt diesen Fragebogen bitte einzeln aus (keine Gruppenabgabe). Hinweis: Die von Ihnen zu gebenden Antworten werden unabängig von Ihrer Matrikelnummer ausgewertet. Ihre Antworten sind daher anonym. Antworten Sie bitte nach bestem Wissen und Gewissen, damit unsere Auswertung einen sinnvollen Beitrag für den weiteren Erfolg des Mopo-Projektes liefern kann.

(*): Sollten Sie weder ein Notebook innerhalb des Projektes erstanden haben, noch das FunkLAN nutzen/genutzt haben, d.h. in keinem Fall eine Teilnahme bei uns vorliegen, sind Sie von dieser Umfrage freigestellt.

Nehmen Sie diese Projekt, das wir ehrenamtlich für Sie betreiben, bitte ernst. Im Falle einer versuchten Manipulation der Umfrageresultate behalten wir uns die Option auf einen Punkteabzug vor.

Aufgabe 2

Punkte (2.5, 3.5)

Zur Berechnung der Funktion $f = a \oplus b \oplus c \oplus d$ kann die Realisierung aus Abbildung 1 verwendet werden. Die Verzögerungszeiten für die Grundgatter sind für die Bausteinfamilie FAST in Tabelle 1 angegeben. Vernachlässigen Sie in dieser Aufgabe die kapazitiven Lasten. Die Anstiegs- und Abfallzeiten an den primären Eingängen sind kleiner als $\delta = 2.5\,ns$. Weiterhin sind die Anstiegs- und Abfallzeiten an den Ausgängen eines Gatters kleiner als δ, falls die Anstiegs- und Abfallzeiten an den Eingängen des Gatters kleiner als δ sind. Alle primären Eingänge schalten zum Zeitpunkt t_0 auf die neuen logischen Werte. Bis zu welchem Zeitpunkt liegt an Signal f mindestens der alte logische Wert an und ab welchem

Zeitpunkt liegt sicher der neue logische Wert an, wenn:

1.) ein $\oplus$-Gatter durch die Realisierung aus Abbildung 2 zusammengesetzt wird.
2.) ein $\oplus$-Gatter durch die Realisierung aus Abbildung 3 zusammengesetzt wird.

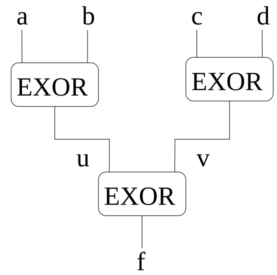

Abbildung 1: Realisierung der $\oplus$-Funktion mit 4 Eingängen

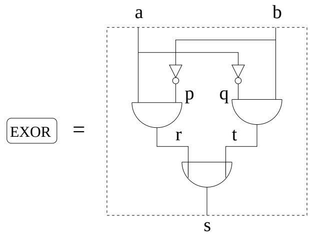

Abbildung 2: AND/OR Realisierung eines $\oplus$-Gatters

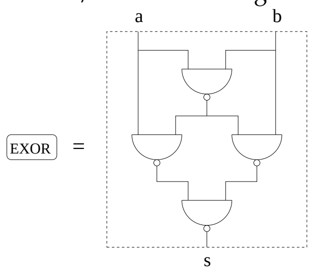

Abbildung 3: NAND Realisierung eines $\oplus$-Gatters

	NAND 74F00		NOT 74F04		AND 74F08		OR 74F32	
	min	max	min	max	min	max	min	max
t_{PLH}	2.4	6.0	2.4	6.0	3.0	6.6	3.0	6.6
t_{PHL}	1.5	5.3	1.5	5.3	2.5	6.3	3.0	6.3

Tabelle 1: Verzögerungszeiten von FAST Bausteinen in ns

Aufgabe 3
Punkte (1, 4)

Betrachten Sie die hierarchische Realisierung eines Addierers in Abbildung 4. Die Verzögerungszeiten für die Gatter der Bausteinfamilie FAST sind in Tabelle 1 angegeben. Nehmen Sie für diese Aufgabe bezüglich der kapazitiven Lasten und der Anstiegs- und Abfallzeiten dieselben Vereinbarungen wie in Aufgabe 1 an. Alle primären Eingänge schalten zum Zeitpunkt t_0 auf die neuen logischen Werte.

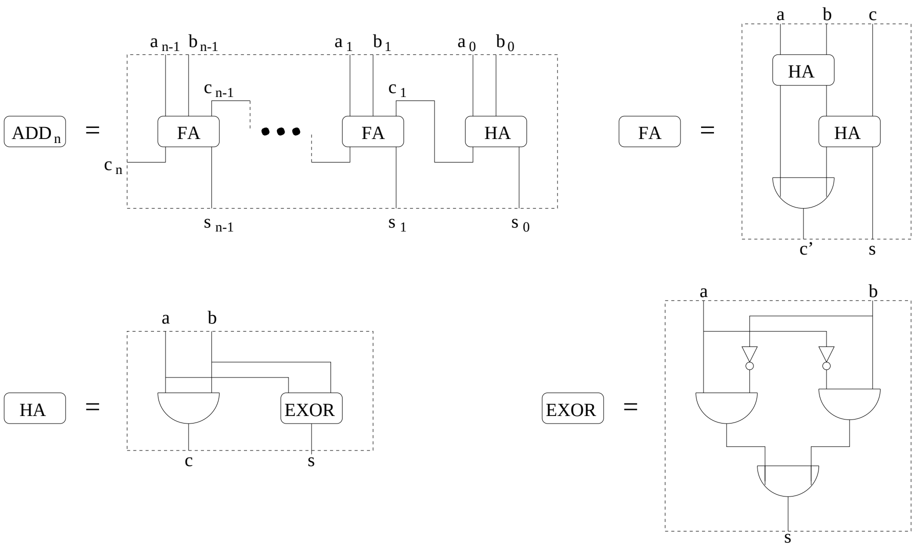

Abbildung 4: Hierarchische Beschreibung eines Carry-Ripple Addierers

Bis zu welchem Zeitpunkt liegt an Signal s_0 mindestens der alte logische Wert an und ab welchem Zeitpunkt liegt sicher der neue logische Wert an?

Aufgabe 4

Punkte (4)

Für die Realisierung einer n-Bit ALU ist ein modifizierter Multiplexer nötig, der folgende Funktion berechnet:

$$MMUX_r : \{0,1\}^{n \cdot r+k} \rightarrow \{0,1\}^n \text{ mit } MMUX_r(d^0, \ldots, d^{r-1}, s) = d^{\langle s \rangle}$$

$$\forall d^i \in \{0,1\}^n, i \in \{0, \ldots, r-1\}, s \in \{0,1\}^k \text{ und } r = 2^k.$$

Realisieren Sie den Multiplexer $MMUX_r$ und bestimmen Sie die asymptotischen Kosten und die Tiefe Ihres Schaltkreises.

Aufgabe 5

Punkte (2,3)

Um einen schnellen und kostengünstigen Multiplizierer für zwei n-Bit Binärzahlen entwerfen zu können, müssen die einzelnen Partialprodukte in geeigneter Art und Weise addiert werden. Aus diesem Grund ist in der Vorlesung der sogenannte *Wallace-Tree Multiplizierer* vorgestellt worden, eine *baumartige* Lösung auf Basis von Carry-Save Addierern, 4-zu-2 Reduktionszellen und Carry-Lookahead Addierern.

1.) Geben Sie einen solchen *baumartigen* Schaltkreis an, der zwei 8-Bit Binärzahlen multipliziert.
2.) Erläutern Sie detailliert die Funktionsweise des Schaltkreises anhand der Multiplikation der Zahlen $a = 173$ und $b = 247$, indem Sie explizit die auftretenden Zwischenergebnisse in Ihre Realisierung aus der ersten Teilaufgabe eintragen.

Abgabe : Montag, den 13. Mai 2002 bis 17.00 Uhr