ALBERT-LUDWIGS-UNIVERSITÄT FREIBURG
INSTITUT FÜR INFORMATIK

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

Freiburg, 25. Juni 2002

Übungsblatt 08 zur Vorlesung

Technische Informatik II

Aufgabe 1
Punkte (4)

Erstellen Sie eine Timing Analyse der in Abbildung 1 dargestellten Schaltung. Die Schaltung kann zur Verlängerung von Kontrollsignalen eingesetzt werden. Betrachten Sie dazu Abbildung 2 und ergänzen Sie den Verlauf des Signales $/oe$. Berechnen Sie alle Zeitpunkte t, $t \geq 0$, an denen das Signal $/oe$ frühestens oder spätestens steigt oder fällt. Berechnen Sie daraus die minimale Zeit τ, sodass das Signal $/oe$ mindestens zu einem Zeitpunkt inerhalb von zwei Takten inaktiv wird. Verwenden Sie für Ihre Berechnung die Verzögerungszeit des Flipflops 74F74 aus dem Datenblatt I.

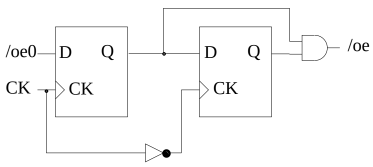

Abbildung 1: Erzeugung verlängerter Kontrollsignale.

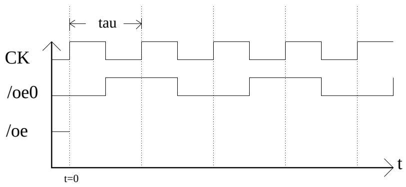

Abbildung 2: Timing-Analyse

Aufgabe 2

Punkte (6)

Aus der Vorlesung wissen Sie, dass das Signal $/PCload$ zum Laden des Befehlszählers vom internen Datenbus DI aus benötigt wird. Geben Sie die PAL-Gleichung für $/PCload$ an (Register-PAL $20R8$).

Hinweis: Der Test, ob der Akkumulator gleich Null ist, der für bedingte JUMP-Befehle benötigt wird, soll hierbei mit zwei getrennten kombinatorischen PALs der Baureihe $20L8$ realisiert werden (wie in der Vorlesung angedeutet, Signale ZH und ZL).

Aufgabe 3

Punkte (5)

Betrachten Sie die 32-Bit Conditional-Sum ALU aus Abbildung 3, die anstelle der bisherigen ALU im RE-TI-II Rechner eingesetzt werden soll. Sie besteht im Wesentlichen aus einem 16-Bit Multiplexer und drei 16-Bit ALUs nach dem *Carry-Lookahead* Prinzip (Abbildung 4). Bestimmen Sie die maximale Verzögerungszeit dieser ALU.

Hinweis: Abbildung 5 zeigt das Timing-Diagramm der 16-Bit ALUs. Bestimmen Sie zunächst die einzelnen Verzögerungen u_1 bis u_5, bevor Sie die Gesamtverzögerung der 32-Bit ALU berechnen. Zum Zeitpunkt t_0 seien alle Operandenbits und der Eingangsübertrag gültig, die *Function Select* Bits seien ausreichend früher stabil. Der Multiplexer hat eine maximale Verzögerung von $7\,ns$ von Dateneingängen zu Ausgängen und $11\,ns$ von Select-Eingängen zu Ausgängen, die übrigen Verzögerungszeiten sind im *Datenblatt 2* aufgelistet.

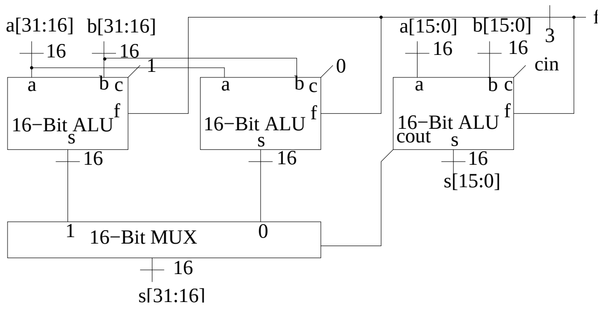

Abbildung 3: 32-Bit Conditional-Sum ALU

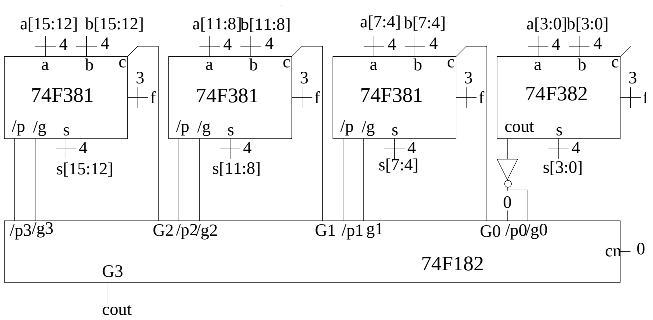

Abbildung 4: 16-Bit *Carry-Lookahead* ALU

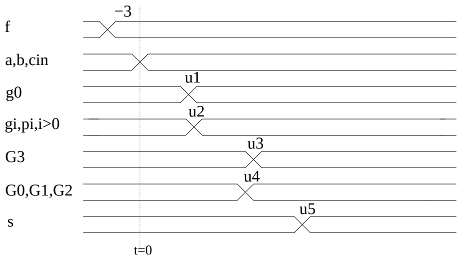

Abbildung 5: Timing-Diagramm der 16-Bit *Carry-Lookahead* ALU

Aufgabe 4

Punkte (5)

Zu einer Funktion $f : \mathbb{B}^k \to \mathbb{B}$, die von den Variablen $X_1, \ldots, X_k$ abhänge, sei die *Supportmenge* $supp(f)$ die Untermenge der Menge $\{X_1, \ldots, X_k\}$ von f mit folgender Eingenschaft:

$$\begin{aligned} X_i \in supp(f) \quad :\Longleftrightarrow \quad & \exists a_1, \ldots a_{i-1}, a_{i+1}, \ldots a_k \in \mathbb{B} \\ & f(a_1, \ldots a_{i-1}, 0, a_{i+1}, \ldots a_k) \\ & \neq f(a_1, \ldots a_{i-1}, 1, a_{i+1}, \ldots a_k) \end{aligned}$$

für ein $i \in \{1, \ldots k\}$.
Beispiel: Für die Funktion $f : \mathbb{B}^3 \to \mathbb{B}$ mit $f(a,b,c) = a \wedge c$ gilt $supp(f) = \{a, c\}$.
Die Variable b ist also für die Funktion f „nicht wesentlich“.
Die Additionsfunktion $+_n$ kann man als $n+1$ Funktionen

$$\begin{aligned} s_n : \mathbb{B}^{2n+1} &\to \mathbb{B} \\ s_{n-1} : \mathbb{B}^{2n+1} &\to \mathbb{B} \\ &\vdots \\ s_1 : \mathbb{B}^{2n+1} &\to \mathbb{B} \\ s_0 : \mathbb{B}^{2n+1} &\to \mathbb{B} \end{aligned}$$

auffassen. Berechnen Sie für jede Funktion s_i die Supportmenge (als Untermenge von $\{a_{n-1}, \ldots, a_0, b_{n-1}, \ldots, b_0, c\}$).

Abgabe : Montag, den 1. Juli 2002 bis 17.00 Uhr