

VU Advanced Digital Design

Homework 1

Bernhard Urban

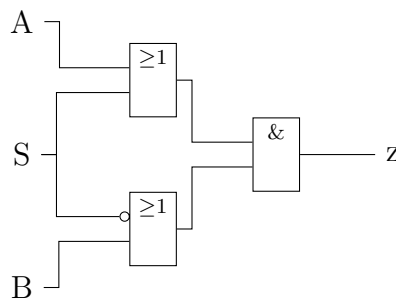
Matr.Nr.: 0725771

lewurm@gmail.com

21. Mai 2011

1 Hazards

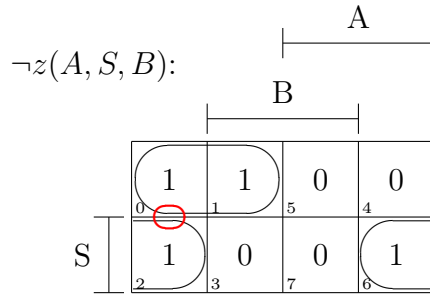
Folgende Schaltung ist gegeben:



1.1 KV-Diagramm

Mittels KV-Diagramm lassen sich die Static-0 Hazards bestimmen ($\neg z$ ist gesucht!)

1 Hazards



Der Übergang in dem ein Static-0 Hazard auftreten kann, ist markiert. Alternativ kann das ganze auch etwas formaler aus dem Minterm ermittelt werden:

$$\neg Z = (\neg S \wedge \neg A) \vee (S \wedge \neg B) \tag{1}$$

Daraus folgt mit einer Belegung von $A = 0$ und $B = 0$

$$\neg Z = \neg S \vee S \tag{2}$$

$$Z = S \wedge \neg S \tag{3}$$

was einem Static-0 Hazard entspricht.

1.2 Kung values

Zuerst ist die Wahrheitstabelle für das OR-Gatter und das NOT-Gatter mit neunwertiger Logik zu bestimmen:

OR-Gatter

≥ 1	0	1	↓	↑	S1	S0	D+	D-	*
0	0	1	↓	↑	S1	S0	D+	D-	*
1	1	1	1	1	1	1	1	1	1
↓	↓	1	↓	S1	S1	D-	S1	D-	*
↑	↑	1	S1	↑	S1	D+	D+	S1	*
S1	S1	1	S1	S1	S1	S1	S1	S1	*
S0	S0	1	D-	D+	S1	S0	D+	D-	*
D+	D+	1	S1	D+	S1	D+	D+	S1	*
D-	D-	1	D-	S1	S1	D-	S1	D-	*
*	*	1	*	*	*	*	*	*	*

NOT-Gatter

\neg	0	1	↓	↑	S1	S0	D+	D-	*
	1	0	↑	↓	S0	S1	D-	D+	*

Anhand der Tabellen (inkl. des AND-Gatter auf den Folien) ist nun beispielhaft zu zeigen, welche Eingangswerte zu $D+$ bzw. $D-$ führen können:

$$D+ : A = D+, \quad S = S1, \quad B = D+$$

$$D- : A = \downarrow, \quad S = \downarrow, \quad B = \downarrow$$

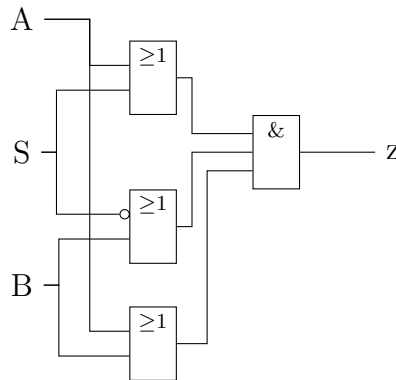
1.3 Erweiterte Schaltung

Um sogenannte “Single Input Change“-Glitches zu vermeiden, muss der Minterm wie folgt erweitert werden:

$$\neg Z = (\neg S \wedge \neg A) \vee (S \wedge \neg B) \vee (\neg A \wedge \neg B) \quad (4)$$

$$Z = (S \vee A) \wedge (\neg S \vee B) \wedge (A \vee B) \quad (5)$$

Oder als Schaltung:



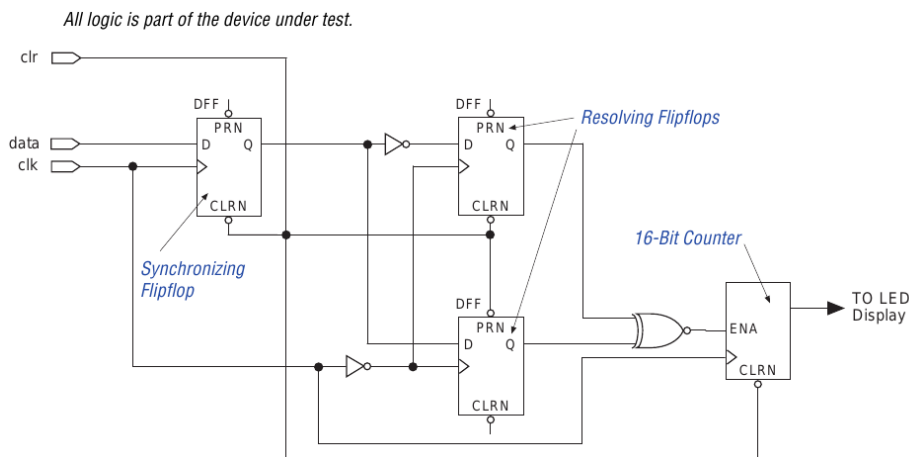
Durch ein “Multiple Input Change” kann jedoch noch immer ein Static-0 Hazard entstehen, zum Beispiel bei diesem Übergang:

$$A = 0, \quad S = 1, \quad B = 0$$

$$\Rightarrow A = 0, \quad S = 0, \quad B = 1$$

Vergleiche dazu auch den Übergang im KV-Diagramm von der Zelle 2 auf 1 über 3.

2 Metastability Measurement / Synchronizer



2.1 Measurement Circuit

Die Schaltung besteht prinzipiell aus einem Synchronizing FF, zwei Resolving FFs wobei eines einen Inverter vorgeschaltet hat, einem XNOR-Gatter und einem 16-Bit Counter.

Die asynchrone Leitung wird nun als Eingang an das Synchronizing FF angelegt. Dessen Ausgang wird mit einem der Resolving FF verbunden und ausserdem mit einem Inverter dessen Ausgang mit dem anderen Resolving FF verbunden ist. Die Ausgänge dieser Resolving Ausgänge werden nun an ein XNOR-Gatter angeschlossen. In einer metastabilenfreien Welt wäre das Ergebnis dieses Gatters natürlich immer logisch 0. Im Falle von logisch 1, das heisst ein metastabiler Zustand wurde erkannt, wird der dahinterliegende Counter inkrementiert.

Alle Elemente sind mit der selben Clock verbunden, die Resolving FFs werden jedoch von einer fallenden Flanke getriggert. Dadurch erreicht man, dass man die sogenannte *settling time* t_{MET} des zu messenden Flip-Flops durch die Clock High Time Δt eingestellt werden kann.

2.2 Uncertainties

Verhalten Das Verhalten der Resolving FFs kann produktionsbedingt unterschiedlich sein, d.h. trotz selben Inputs haben die Resolving FFs verschiedene Ausgabewerte, dadurch könnte ein Upset nicht gezählt werden. Dieser Effekt könnte durch anderen Einflussfaktoren wie zum Beispiel Temperatur zusätzlich verstärkt werden.

Inverter Dieser erzeugt ein zusätzliches Delay das nicht in die Berechnung der Parameter miteinbezogen wird.

Clock-Skew Die Flanke der Clockleitung erreicht die bei den Resolving FFs nie gleichzeitig \Rightarrow wenn der Skew zu groß ist könnte der Counter schon zählen, weil einer der Resolving FFs den neuen Wert noch nicht übernommen hat.

Creeping Durch undefinierte Pegeln (Creeping) am Ausgang der Resolving FFs, könnte es sein dass der Counter falsch zählt.

2.3 Clock Booster

Im Prinzip wie ein üblicher 2-Stage Synchronizer, jedoch werden die FF-Elementen mit der doppelten Frequenz betrieben. Dadurch erreicht man, dass die asynchronen Daten trotz den zwei FF-Elementen in einem Taktzyklus für die eigentliche Schaltung zur Verfügung steht. Dies ist bei zeitkritischen asynchronen Signalen, wie zum Beispiel Interruptleitungen von Vorteil. Durch die doppelte Frequenz verringert sich aber der MTBU,

3 Synchronizer

wie folgende Beziehung zeigt:

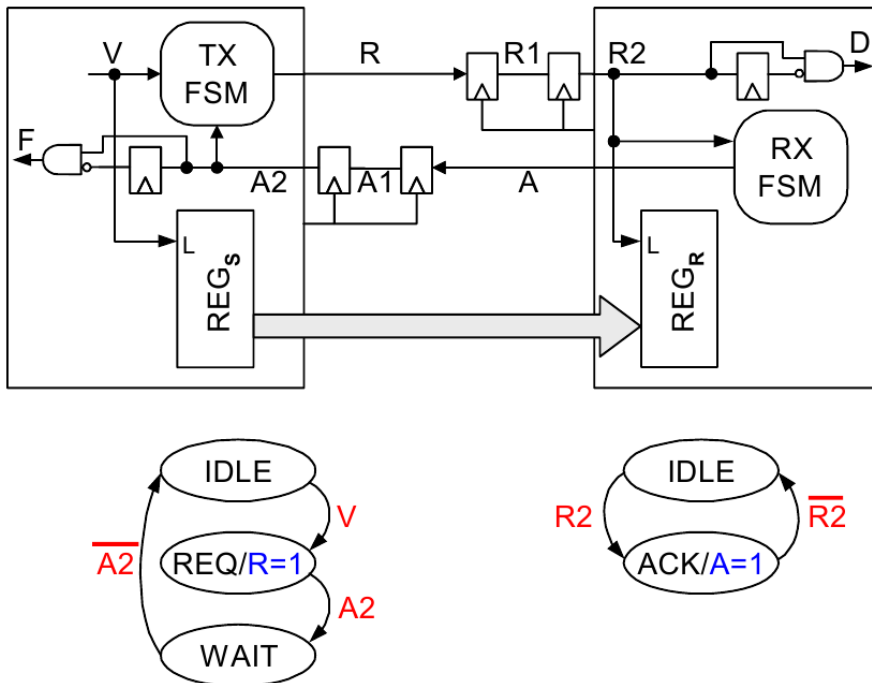
$$MTBU_{CB} = \frac{e^{\frac{t_{res}}{\tau C}}}{\lambda_{dat} \cdot T_0 \cdot 2 \cdot f_{clk}} \quad (6)$$

$$MTBU_{2-Stage-Sync} = \frac{e^{\frac{t_{res}}{\tau C}}}{\lambda_{dat} \cdot T_0 \cdot f_{clk}} \quad (7)$$

$$\Rightarrow MTBU_{CB} = \frac{MTBU_{2-Stage-Sync}}{2} \quad (8)$$

3 Synchronizer

Die essentielle Aussage des Papers von Ginosar: man sollte nicht allzu kreativ sein :-)
Die Wahl fällt daher eindeutig auf den “Two-Flop Synchronizer” (in der Pushvariante), hier abgebildet:



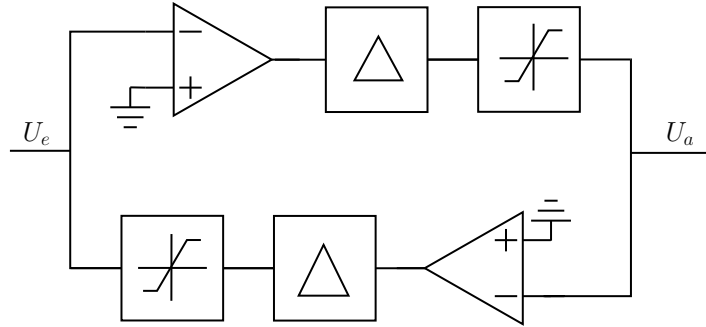
Die wesentliche Idee dabei ist folgende: Der Sender kann dem Empfänger einen Request schicken bzw. der Empfänger den Erhalt mit einem ACK bestätigen kann. Auf Basis dessen kann ein “Four-Phase Handshake Protocol” verwirklicht werden. Der Datentransfer findet über ein Register statt. Da der Empfänger und der Sender asynchron zueinander sind, werden das Request und ACK Signal synchronisiert; bei der vorgestellten Schaltung im Paper passiert das mit jeweils zwei Flip-Flops (R1 & R2 bzw. A1 & A2).

Das Protokoll wird mittels je einer FSM gesteuert: Durch das Signal V wird der Sender getriggert und setzt R=1 während die Daten bei REG_S anliegen. Dadurch wird der Empfänger aufgefordert sein Register REG_L mit den Daten zu füllen. Danach signalisiert er dem Sender mit A=1, dass die Daten erfolgreich “gelatcht” wurden. Der Sender

bestätigt das mit dem Setzen von $R=0$; der Empfänger setzt schlussendlich $A=0$ und beide Statemaschinen befinden sich wieder im initialen Zustand.

4 Metastability Modeling

Mithilfe folgender Schaltung soll Metastabilität modelliert werden:



Es ist nun das Verhalten dieser Schaltung zu untersuchen, wenn an U_e ein kurzer Impuls eingespeist wird. Die Dauer des Impulses am Einganges wird mit T_I , die Verzögerung des Pure Delays mit T_Δ und das Delay des Integrators mit T_R bezeichnet. Ich gehe davon aus, bevor I angelegt wird, ausreichend lange logisch 0 angelegt wurde $\Rightarrow U_a$ entspricht deshalb logisch 1 (stabil).

Je nach Dauer von T_I sind nun verschiedene Szenarien in Betracht zu ziehen:

1. $T_I < \frac{T_R}{2}$: Der Puls ist zu kurz \Rightarrow der Ausgang des Integrators wird nicht in den negativen Spannungsbereich kommen. Der Pegel ist zwar kurzfristig unzulässig, wird aber schnell wieder vom darauffolgenden Komparator "gerichtet".
2. $\frac{T_R}{2} < T_I < T_R$: U_a wird eine negative Spannung annehmen, durch die Rückführung wird dieser kurze negative Puls jedoch vollständig vom zweiten Integrator eliminiert.
3. $T_R < T_I < 2 \cdot T_\Delta$: Der Puls ist lange genug um den Integrator voll zum Aussteuern zu bringen, wird aber von den Delay Elementen so verzögert dass die Schaltung zum Oszillieren beginnt.
4. $2 \cdot T_\Delta < T_I < 2 \cdot T_\Delta + T_R$: Die Schaltung schwingt zwar anfangs ein wenig, U_a nimmt dann schlussendlich aber logisch 0 an. Das Verhalten ist ähnlich wie im Fall (2), aber "invertiert".
5. $2 \cdot T_\Delta + T_R < T_I$: Hier verhält sich die Schaltung wie es von einem Latch zu erwarten ist.

Wie man sieht, kann die Schaltung zwar kurzfristig undefinierte Zustände erreichen, diese werden aber durch die invertierenden Komparatoren wieder eliminiert. D.h. das Modell ist meiner Meinung nach nicht geeignet. Einzig zur Modellierung von Oszillation, verursacht durch Metastabilität, wäre es unter Umständen geeignet.