

ulm university universität
uulm

Grundpraktikum der Elektrotechnik

Versuch 12: Würfeln mit Eccles & Jordan

Institut für Mikroelektronik
Universität Ulm



Grundpraktikum Elektrotechnik

Versuch 12: Würfeln mit Eccles & Jordan

Inhaltsverzeichnis

1	Einführung	1
1.1	Sequentielle Logik	1
1.2	Latches	2
1.3	Flip-Flops	3
1.4	Zähler und Register	4
1.5	Linear rückgekoppelte Schieberegister	4
1.6	DVB-Scrambler	5
1.7	GPS C/A-Code	6
1.8	Digital-Board	9
2	Vorbereitende Aufgaben	10
2.1	Latches und Flip-Flops	10
2.2	XOR mit 6 Eingängen	11
2.3	Asynchrone Zähler	11
2.4	Synchrone Zähler	12
2.5	Schieberegister	13
2.6	Linear rückgekoppeltes Schieberegister	13
3	Versuchsdurchführung	14
3.1	Latches und Flip-Flops	14
3.2	8 Bit-Asynchrone Zähler	15
3.3	4 Bit-Synchrone Zähler	17
3.4	Schieberegister	17
3.5	DVB-S Scrambler	18
3.6	GPS C/A-Code Generator	18

1 Einführung

Latches und Flip-Flops sind Schaltungen mit zwei stabilen Zuständen und können daher verwendet werden, um die Datenmenge von 1 Bit zu speichern. Als Speicherzelle bilden sie die Grundlage der sequentiellen Logik und sind somit unverzichtbar für die heutige Digitaltechnik. Das erste Latch wurde schon lange bevor es Transistoren oder integrierte Schaltungen gab im Jahre 1918 von den beiden englischen Physikern William Henry Eccles und Frank Wilfred Jordan erfunden und patentiert und bestand aus einer bistabilen Schaltung aus zwei Vakuumröhren.

Der grundlegende Aufbau von Speichern aus Logikgattern soll zu Beginn des Versuches betrachtet werden, bevor diese Speicher mit zusätzlichen Gattern zu komplexeren Schaltungen kombiniert werden. So werden Sie im Laufe des Versuchs einen Asynchron-, einen Synchron- und einen Ringzähler, einen Frequenzteiler, einen Addierer sowie ein linear rückgekoppeltes Schieberegister aufbauen und untersuchen.

Die theoretischen Grundlagen für diesen Versuch sind im Skript der Vorlesung *Digitale Schaltungen* (Kapitel 5: Sequentielle Logik) beschrieben. Das für diesen Versuch relevante Wissen wird im folgenden kurz dargestellt, sodass der Versuch erfolgreich durchgeführt werden kann, ohne die Vorlesung gehört zu haben.

1.1 Sequentielle Logik

Im vorhergehenden Versuch über kombinatorische Logik haben Sie Schaltnetze betrachtet, deren Ausgänge ausschließlich von der momentanen Belegung der Eingänge abhängen. Die vorhergehenden Eingangszustände haben daher keine Wirkung auf den Ausgang, somit sind diese Schaltungen nicht in der Lage, Werte zu speichern. Zeitliche Effekte haben Sie nur hinsichtlich der Gatterlaufzeit untersucht, die zu Verzögerungen und auch unerwünschten Zwischenzuständen z.B. durch einen Glitch führen kann.

Durch Rückkoppelungen in der Schaltung kann eine Speicherung erzielt werden, diesen Typ Schaltungen nennt man sequentielle Logik oder auch Schaltwerke, die Gegenstand dieses Versuches sind. Die Ausgänge dieser Schaltungen sind somit nicht nur von der momentanen Belegung der Eingänge, sondern zusätzlich auch noch von vorhergehenden Zuständen abhängig.

Bei sequentieller Logik kann zwischen asynchronen und synchronen Schaltwerken unterschieden werden. Bei asynchronen Schaltwerken bewirkt eine Änderung der Eingänge eine direkte Veränderung der Ausgänge, die Gatterlaufzeiten bestimmen hierbei die Dauer, bis ein stabiles Ergebnis am Ausgang anliegt. Dies ermöglicht schnelle Schaltungen, da jedes Gatter das Signal nur um seine eigene Laufzeit verzögert. Durch unterschiedliche Laufzeiten in verschiedenen Gattern können allerdings unerwünschte Zwischenzustände auftreten wie Sie im Lauf des Versuches beispielsweise beim Asynchronzähler feststellen werden. Bei einem synchronen Schaltwerk können unerwünschte Zwischenzustände vermieden werden, indem die Speicherzellen mit einem zentralen Taktsignal versorgt werden. Dies erfordert jedoch in der Regel einen höheren Schaltungsaufwand als bei asynchronen Schaltwerken. Zudem kann der Takt maximal so schnell wie die langsamste Teilschaltung gewählt werden, weswegen synchrone Schaltungen typischerweise langsamer sind.

1.2 Latches

Die einfachste Speicherschaltung ist das so genannte RS-Latch (benannt nach den Eingängen *Reset* und *Set*) und kann beispielsweise aus zwei NOR-Gattern, wie in Abbildung 1a dargestellt, aufgebaut werden.

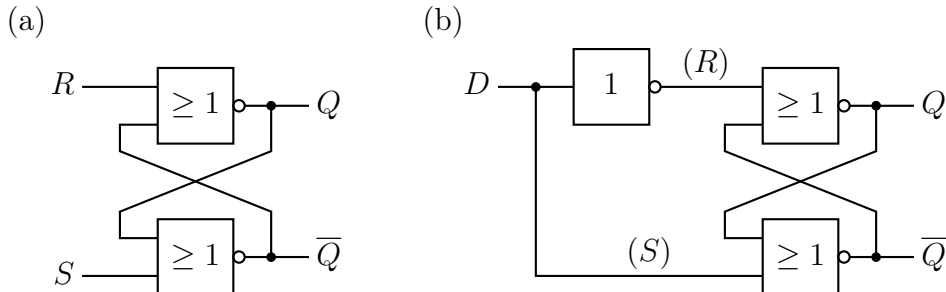


Abbildung 1: (a) Ungetaktetes RS-Latch und (b) ungetaktetes D-Latch.

Eine logische 1 an S , während R auf 0 gesetzt wird, setzt den Ausgang Q auf 1 und den negierten Ausgang \bar{Q} auf 0. Dagegen setzt $S = 0$ und $R = 1$ den Ausgang Q auf 0. Liegt an beiden Eingängen eine logische 0 an, so wird der aktuelle Zustand gespeichert. Die Eingangskombination, bei der sowohl an S als auch an R eine 1 anliegt sollte vermieden werden, da in diesem Fall an beiden Ausgängen eine 0 anliegt, was der Definition von Q und \bar{Q} widerspricht. Dies ist insbesondere problematisch, wenn anschließend R und S gleichzeitig von 1 auf 0 gesetzt werden. Je nach Laufzeit der beiden NOR-Gatter kann der Ausgang einige Zeit oszillieren, bis schließlich ein stabiler Zustand erreicht wird, dabei ist der Zustand von Q und \bar{Q} anschließend nicht bekannt.

Die zu vermeidende Eingangskombination des RS-Latches wird beim D-Latch (das D steht für delay) aus Abbildung 1b vermieden, indem das Eingangssignal *Data* (D) mit S und das invertierte Signal \bar{D} mit R verbunden wird. Allerdings geht hiermit auch der Zustand zum Speichern verloren.

Das D-Latch kann mit einem zusätzlichen Takteingang C (*Clock*) versehen werden, wie in Abbildung 2a dargestellt. Das getaktete D-Latch speichert bei $C = 0$ den Zustand und

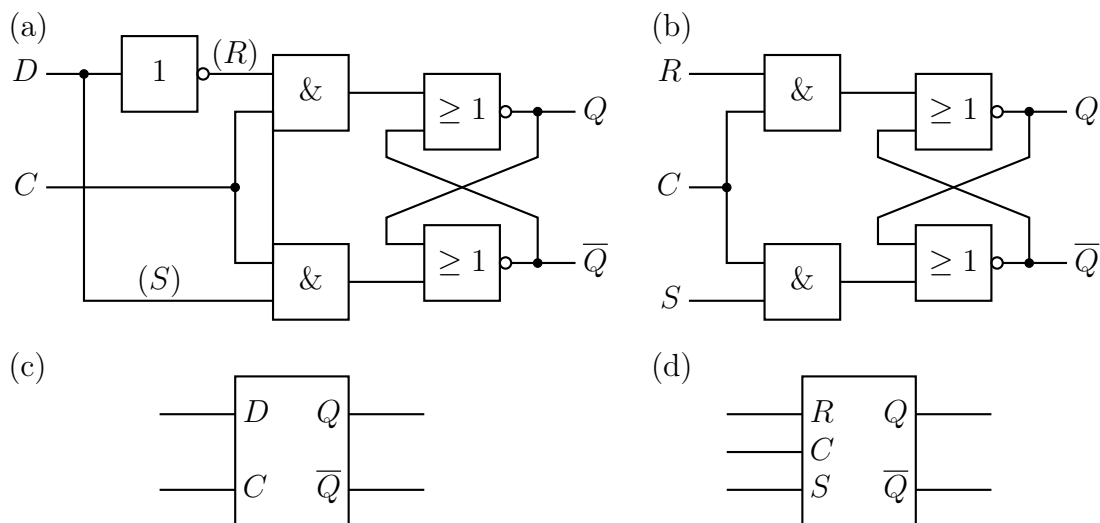


Abbildung 2: (a) Getaktetes D-Latch und (b) getaktetes RS-Latch mit den jeweiligen Symbolen in (c) bzw. (d).

folgt bei $C = 1$ dem Wert von D . Somit ist die Speichermöglichkeit wiederhergestellt, ohne dass der unerwünschte Eingangszustand auftreten kann.

Auch das RS-Latch kann getaktet werden, die Schaltung hierfür ist in Abbildung 2b dargestellt. Es unterscheidet sich von der ungetakteten Variante dadurch, dass es eine anliegende 1 an S oder R nur während der Taktphase ($C = 1$) übernimmt, während es bei $C = 0$ den Inhalt unabhängig von R und S speichert. Allerdings kann auch bei dieser Variante der unerwünschte Eingangszustand $S = R = C = 1$ auftreten.

Getaktete Latches sind transparent, das heißt während des anliegenden Taktsignals sind mehrere Zustandsänderungen am Ausgang möglich, man bezeichnet dies auch als zustandsgesteuert.

1.3 Flip-Flops

Aus zwei getakteten Latches lässt sich ein Flip-Flop realisieren, dass nur eine Änderung des Ausgangs pro Taktsignal ermöglicht. Die Realisierungen eines RS- bzw. D-Flip-Flops aus den jeweiligen getakteten Latches und ihre Symbole sind in Abbildung 3 dargestellt.

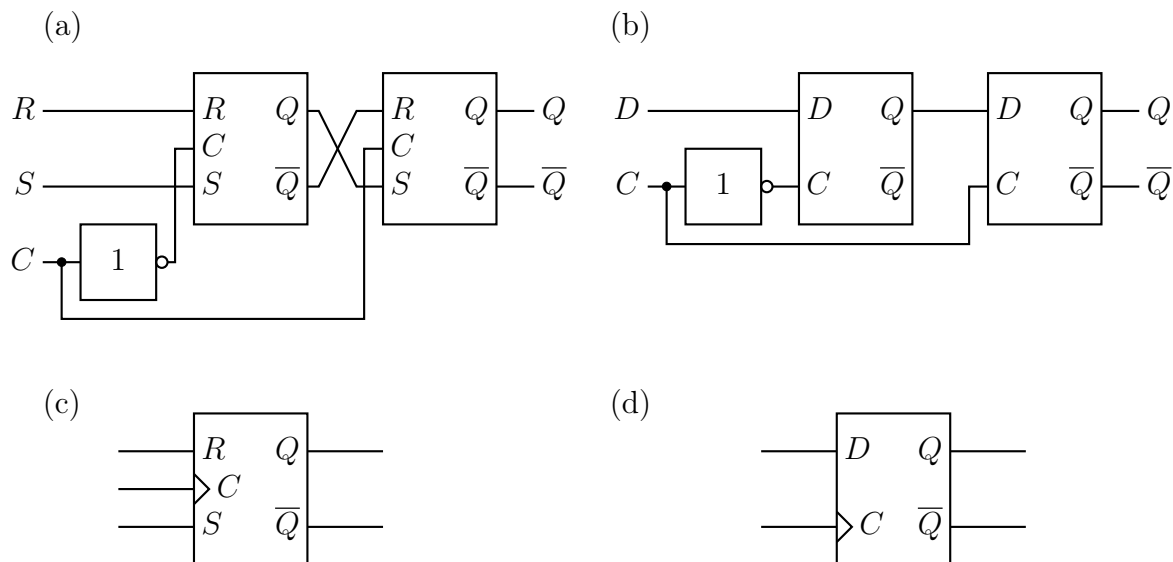


Abbildung 3: (a) RS-Flip-Flop und (b) D-Flip-Flop mit positiv Flankensteuerung und den jeweiligen Symbolen in (c) und (d).

Das vordere Master-Latch ist mit dem invertierten Taktsignal verbunden und ist somit während $C = 0$ transparent. Das dahinter geschaltete Slave-Latch ist dagegen bei $C = 1$ transparent und übernimmt somit bei einer steigenden Taktflanke den Ausgangswert des Master-Latches, dass zu diesem Zeitpunkt keine Ausgangsänderung erfahren kann. Somit ist das Flip-Flop positiv flankengesteuert, da nur zu diesem Zeitpunkt ein neuer Zustand am Ausgang erscheinen kann. Dementsprechend kann ein negativ flankengesteuertes Flip-Flop realisiert werden, wenn das Slave-Latch anstatt des Master-Latches mit dem invertierten Taktsignal verbunden wird. In den Symbolen der Flip-Flops in Abbildung 3c und 3d wird die Flankensteuerung durch das Dreieck am Takteingang angezeigt, ein negativ flankengesteuertes Flip-Flop würde durch einen zusätzlichen Kreis auf den negierten Takteingang hinweisen.

Beim RS-Flip-Flop kann erneut der unerwünschte Eingangszustand auftreten, beim D-Flip-Flop wird dieser wie schon bei den D-Latches vermieden. Durch eine Rückkopplung

von \overline{Q} auf D kann aus dem D-Flip-Flop ein Toggle-Flip-Flop oder auch T-Flip-Flop realisiert werden, was bei jedem anliegenden Taktsignal die Ausgänge Q und \overline{Q} tauscht. Dies wird z.B. für Asynchrone Zähler oder Frequenzteiler verwendet.

Speicherelemente lassen sich somit nach der Art der Steuerung unterscheiden. Bei ungetakteten Latches wird der Zustand nur von den Setz- und Rücksetzeingängen gesteuert während bei getakteten Speichern der Zeitpunkt der Informationsübernahme durch ein Taktsignal vorgegeben wird. Getaktete Latches übernehmen die Information dabei während eines anliegenden Taktpegels und sind in dieser Zeit transparent (Zustandssteuerung), während beim Flip-Flop die Information durch einen Zustandswechsel bzw. eine Flanke des Taktsignals übernommen wird (Flankensteuerung).

1.4 Zähler und Register

Auch bei Zählern unterscheidet man zwischen synchronen und asynchronen Schaltungen. Bei einem Synchronzähler werden alle Flip-Flops mit dem gleichen Takt versorgt, die richtigen Eingangssignale der Flip-Flops werden mittels Gatterlogik aus dem momentanen Zustand erzeugt. Dabei beschränken die Gatterlaufzeiten der Logik die maximal mögliche Taktrate, da die korrekten Eingangssignale rechtzeitig vor dem nächsten Takt anliegen müssen. Dagegen werden beim Asynchronzähler die Taktsignale innerhalb des Schaltwerks selbst generiert, eine zusätzliche Gatterlogik ist daher je nach Zähler nicht nötig. Die einfachste Realisierung eines Asynchronzählers lässt sich durch die Serienschaltung von den oben genannten Toggle-Flip-Flops erzielen, indem der Ausgang \overline{Q} jedes Flip-Flops mit dem C -Eingang des Flip-Flops für das nächsthöhere Bit verbunden wird. Sowohl Synchron- als auch Asynchronzähler werden Sie im Lauf des Versuches noch näher kennenlernen.

Register sind eine Zusammenschaltung mehrerer Flip-Flops mit einem gemeinsamen Takt und einem Satz von Ein- und Ausgängen. Eine Spezialform sind Schieberegister, bei denen die Flip-Flops seriell verschaltet sind, d.h. der Eingang D jedes Flip-Flops ist mit dem Ausgang Q des vorangehenden Flip-Flops verbunden. So werden bei jedem Taktsignal die gespeicherten Werte eine Zelle weitergeschoben. Die Ein- und Ausgabe der Werte kann je nach Bauart des Registers seriell, parallel oder beides zugleich erfolgen, ebenso gibt es Schieberegister mit umschaltbarer Schieberichtung.

1.5 Linear rückgekoppelte Schieberegister

Heutzutage werden in vielen technischen Anwendungen sogenannte Pseudo-Zufallssequenzen benötigt. Der Mobilfunk-Standard GSM setzt sie beispielsweise zur Verschlüsselung ein, das digitale Fernsehen DVB verwendet sie zur Energieverwischung im Signal und dem Satellitennavigationssystem GPS ermöglichen sie die Bestimmung der Signallaufzeiten von den Satelliten zur Erde. Diese Pseudo-Zufallssequenzen, auch Pseudo-Noise-Sequenzen (PN-Sequenzen) genannt, ähneln echten Zufallssequenzen und lassen sich extrem einfach und effizient mit linear rückgekoppelten Schieberegistern (Linear Feedback Shift Register, LFSR) erzeugen. Diese bestehen aus einem Schieberegister der Länge n und der XOR-Verknüpfung einer bestimmten Kombination von Flip-Flop-Ausgängen, deren Ergebnis als Eingangssignal für das Schieberegister verwendet wird. Zusätzlich müssen die Werte zur Vorinitialisierung des Schieberegisters bekannt sein. Ein LFSR mit n Bit kann dabei eine periodische Pseudo-Zufallssequenz mit einer maximalen Periodenlänge

von $2^n - 1$ erzeugen. Abbildung 4 zeigt unterschiedliche Darstellungen desselben linear rückgekoppelten Schieberegisters der Länge 3, das am Ausgang eine PN-Sequenz liefert. In diesem Fall werden der Ausgang des ersten und des letzten Flip-Flops XOR-verknüpft auf den Eingang zurückgeführt. Oft wird anstelle der XOR-Verknüpfung die gleichbedeutende binäre Addition (ohne Übertrag) verwendet, d. h.

$$0 + 0 = 0 \quad 0 + 1 = 1 \quad 1 + 0 = 1 \quad 1 + 1 = 0.$$

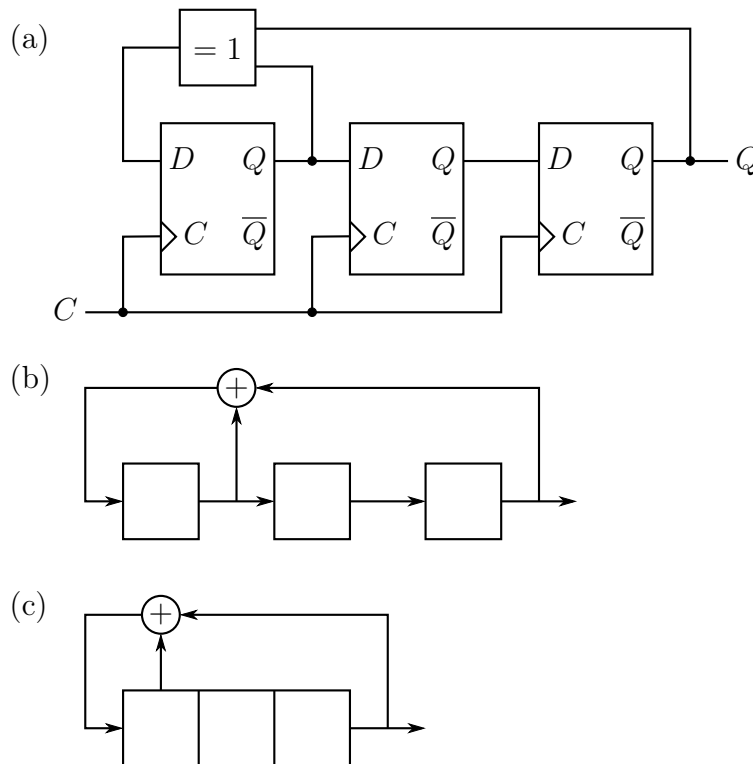


Abbildung 4: Unterschiedliche Darstellungen desselben linear rückgekoppelten Schieberegisters der Länge 3.

1.6 DVB-Scrambler

In Abbildung 5 ist der Scrambler (Verwürfler) des digitalen Fernsehens nach dem DVB-Standard (identisch bei DVB-S, DVB-T und DVB-C) dargestellt. Dieser besteht aus einem Schieberegister der Länge 15, die Ausgänge der letzten beiden Flip-Flops werden XOR-verknüpft auf den Eingang zurückgeführt. Ein zusätzliches AND-Gatter wird verwendet, um die Verwürfelung ein- und ausschalten zu können. Der zu übertragende Datenstrom wird Hilfe eines XOR-Gatter mit der PN-Sequenz verwürfelt. Auf der Empfängerseite wird das selbe Schieberegister zur Erzeugung der Daten für die Decodierung eingesetzt. Dabei muss gewährleistet sein, dass die Schieberegister auf Sende- und Empfängerseite gleich initialisiert werden und synchron arbeiten.

Die Verwürfelung wird eingesetzt, um Gleichanteile aus dem Signal zu minimieren, die z.B. bei einem komplett schwarzen oder weißen Bild auftreten würden. Durch die Übertragung quasi „zufälliger“ Daten wird das Signal über die gesamte Bandbreite des Übertragungskanal recht gleichmäßig verteilt, somit wird die Bandbreite besser ausgenutzt

und Energiespitzen minimiert, was Interferenzen durch Intermodulation zwischen benachbarten Signalbändern reduziert.

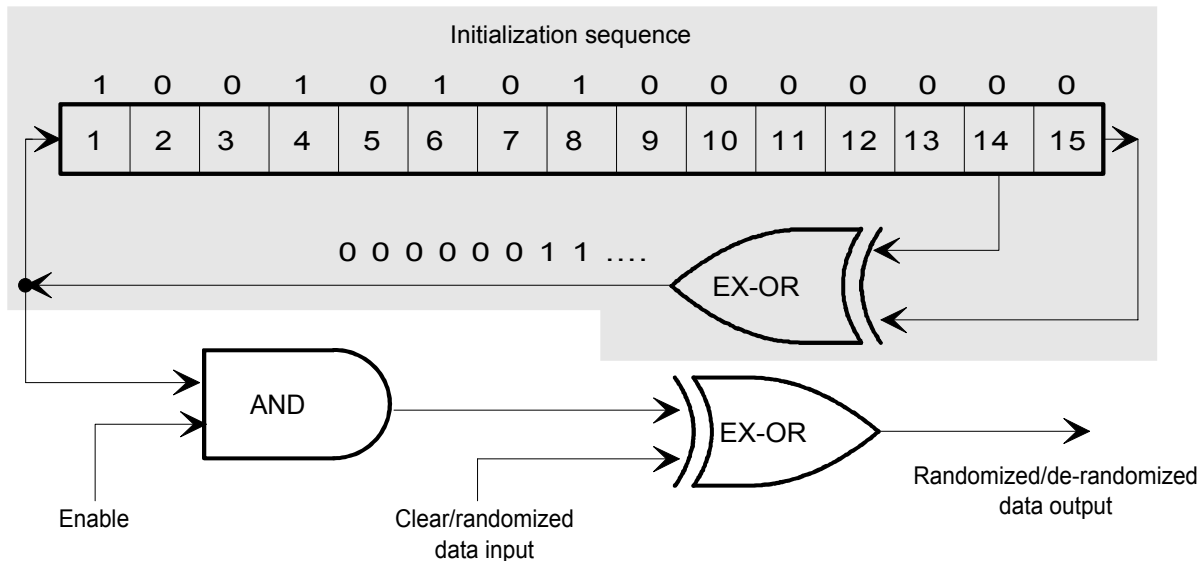


Abbildung 5: Scrambler/Descrambler von DVB-S.
Quelle: DVB-S-Standard (ETSI EN 300 421 V1.1.2)

1.7 GPS C/A-Code

Die Satelliten des *Global Positioning System* (GPS) senden zwei Datensignale aus, den frei empfangbaren C/A-Code (*coarse/acquisition*) für zivile und den verschlüsselten P/Y-Code (*precision/encrypted*) für militärische Anwendungen.

Der C/A-Code-Generator des GPS-Systems ist in Abbildung 6 dargestellt. Für den Versuch benötigen Sie nur den grau hinterlegten Teil des Code-Generators. Dieser besteht aus zwei linear rückgekoppelten Schieberegistern der Länge 10 (G1-Shift-Register und G2-Shift-Register; vgl. Abbildung 7 und 8). Das erste liefert die G1-Sequenz, die bei jedem Satelliten identisch ist. Im zweiten Register G2 wählt die Phase-Select-Logic für jeden Satelliten eine von 32 fest zugewiesenen Kombinationen aus zwei Flip-Flop-Ausgängen aus, die XOR-verknüpft für die Erzeugung der Sequenz $G2_i$ eingesetzt werden. Die XOR Verknüpfung aus der G1-Sequenz und der $G2_i$ -Sequenz ergeben das Ausgangssignal G_i , diese C/A-Code-Sequenz ist für jeden Satelliten unterschiedlich.

Jeder Satellit sendet nun die selbe Nachricht, bestehend aus einem sehr genauen Zeitsignal und weiteren Informationen (unter anderem Bahndaten der Satelliten) aus und codiert sie mit seinem eigenen C/A-Code. Auf Empfängerseite werden ebenfalls alle 32 möglichen C/A-Codes erzeugt, die mit den empfangenen Daten abgeglichen werden. Somit können die empfangenen Satelliten klar identifiziert werden.

Für eine erfolgreiche GPS-Ortung werden mindestens 4 Satelliten benötigt. Einer dient als Referenz-Zeitsignal, aus den Laufzeitunterschieden zu den anderen 3 Satelliten kann durch Triangulation die Position bestimmt werden. Die Genauigkeit liegt für zivile Anwendungen bei einigen Metern.

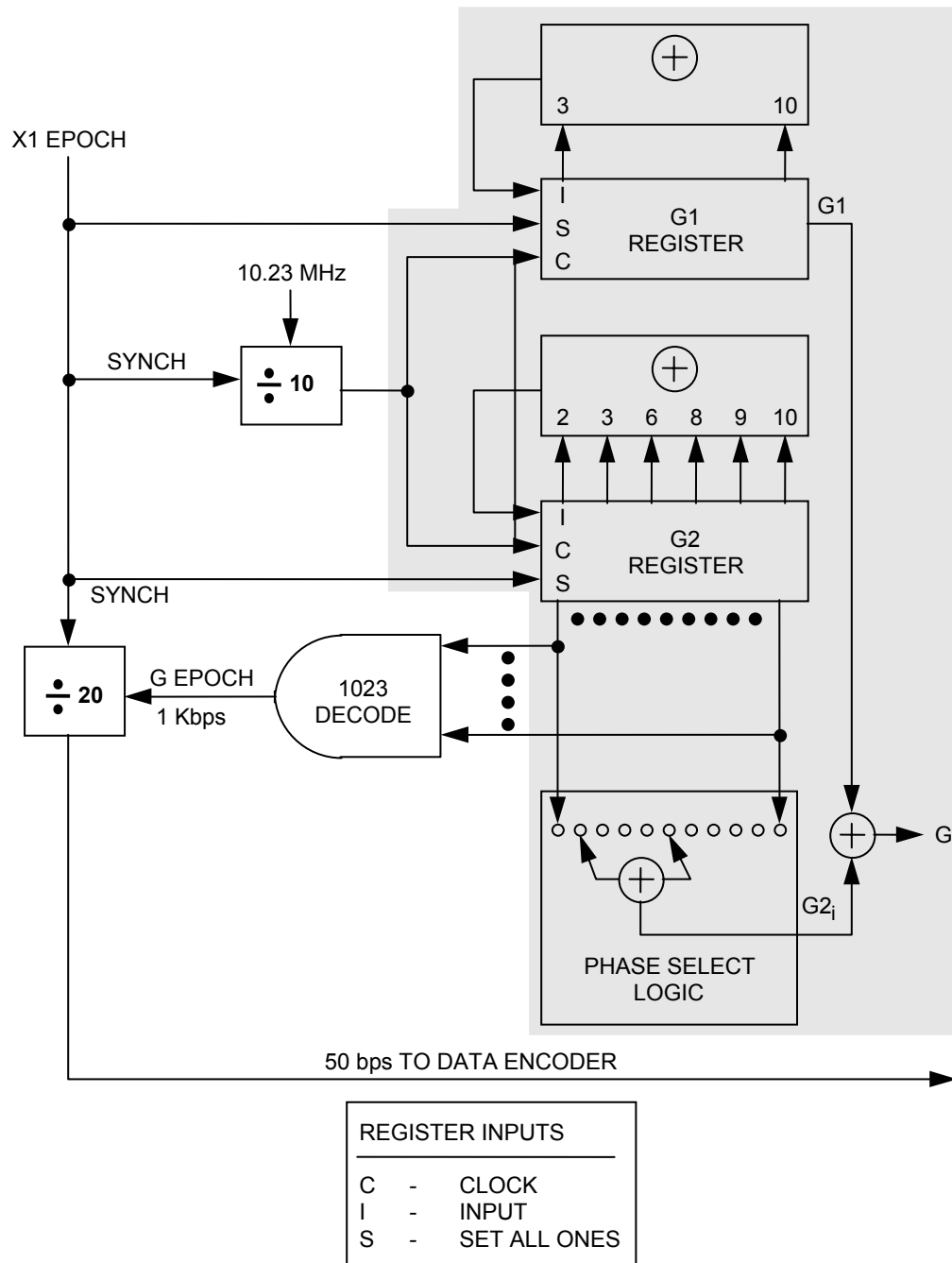


Abbildung 6: C/A-Code-Generator des GPS-Systems.
Quelle: GPS-Standard (GPS JPO IS-GPS-200D)

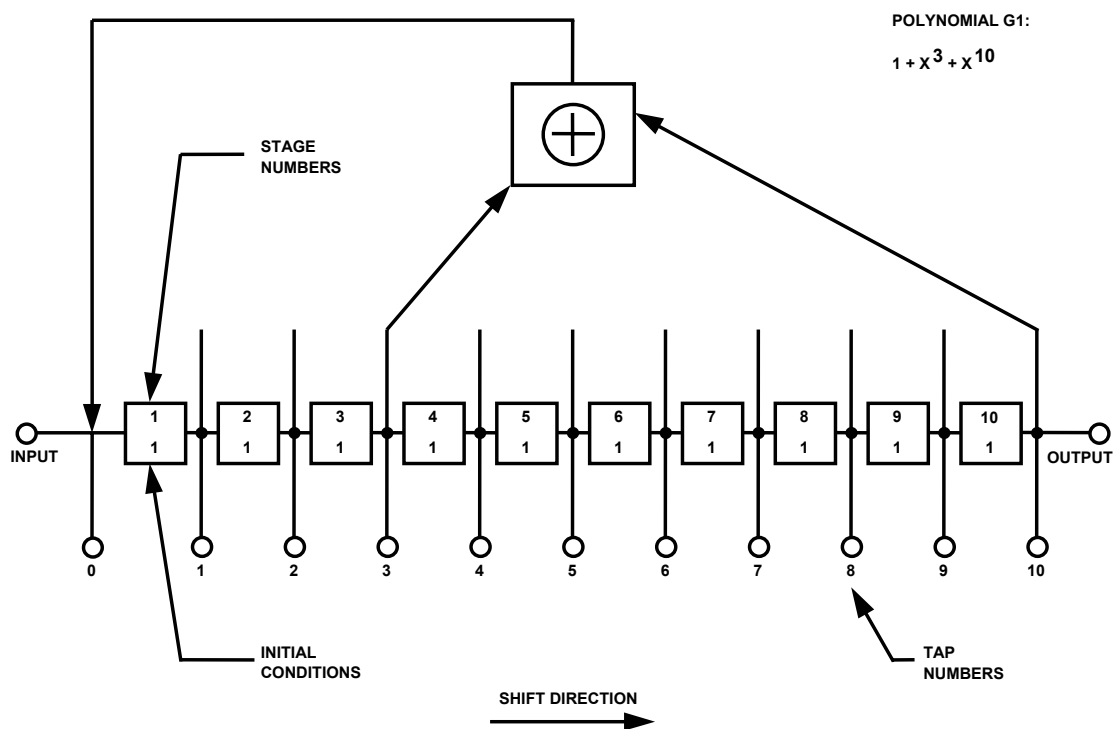


Abbildung 7: G1-Shift-Register des GPS-Systems.
 Quelle: GPS-Standard (GPS JPO IS-GPS-200D)

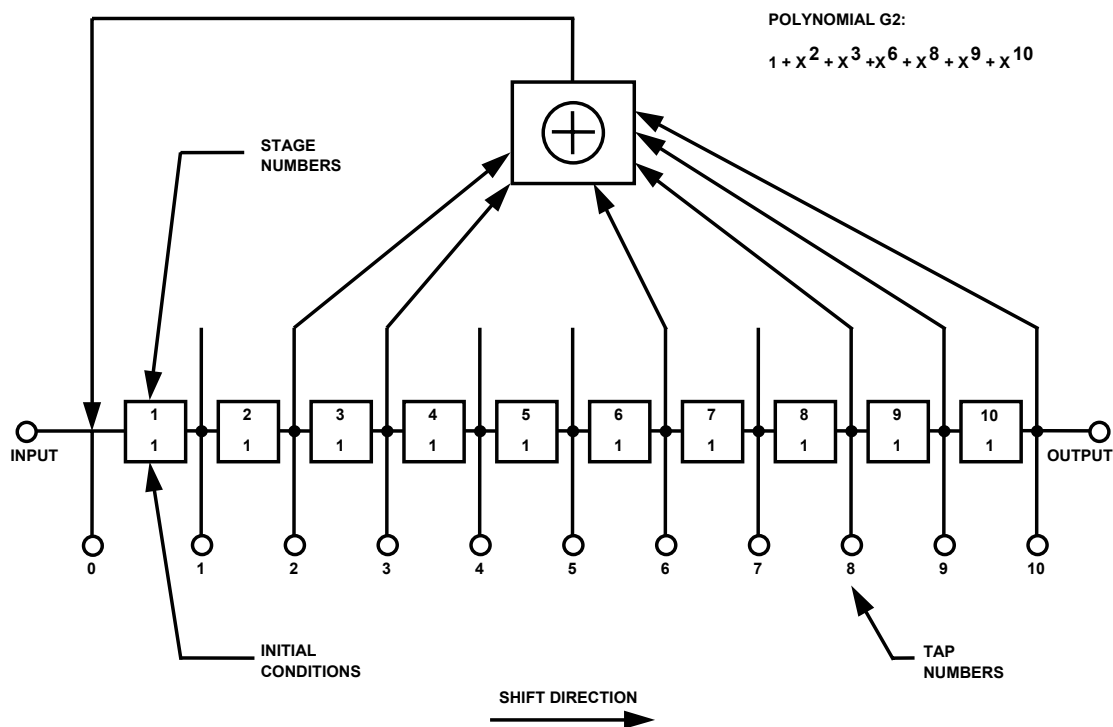


Abbildung 8: G2-Shift-Register des GPS-Systems.
 Quelle: GPS-Standard (GPS JPO IS-GPS-200D)

1.8 Digital-Board

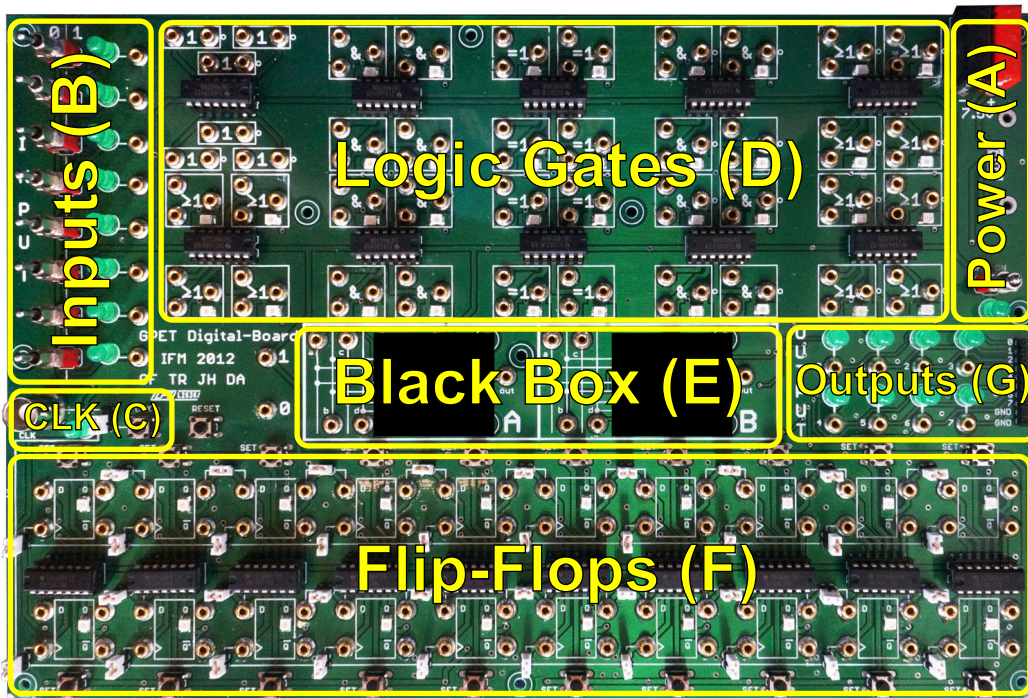


Abbildung 9: *Digital-Board* zur Realisierung von Schaltnetzen und Schaltwerken.

Das speziell für die digitalen Versuche entworfene *Digital-Board* kennen Sie bereits aus dem vorhergehenden Versuch, es ist in Abbildung 9 dargestellt.

Zusätzlich zu den bereits verwendeten Komponenten befinden sich im Bereich (F) 20 D-Flip-Flops. Jedes Flip-Flop besitzt einen Daten- (D) und einen Takt-Eingang (C), sowie die Ausgänge Q und \bar{Q} , der aktuelle Wert des Ausgangs Q wird mit einer orangen LED angezeigt. Es gibt einen zentralen *RESET*-Taster, der den Speicherinhalt aller Flip-Flops auf 0 setzt. Dieser sollte nach dem Anschalten der Versorgungsspannung einmal gedrückt werden, um die Speicher zu initialisieren. Ebenso kann der Inhalt jedes Flip-Flops per individuellem *SET*-Taster auf 1 gesetzt werden. Für den Einsatz als Schieberegister kann außerdem der Ausgang Q jedes Flip-Flops mit dem Eingang D des benachbarten Speichers mit den horizontalen Jumpern verbunden werden. Alternativ zum eigenen Takteingang kann jedes Flip-Flop mittels der vertikalen Jumper auch mit einem zentralen Taktsignal verbunden werden. Dies erleichtert die Realisierung synchroner Schaltungen. Das zentrale Taktsignal kann im Bereich (C) wahlweise mit dem Taster erzeugt werden, dieser muss dafür per Jumper mit der Takt-Leitung verbunden werden. Alternativ kann mit diesem Jumper auch der externe Takteingang, der über eine BNC-Buchse realisiert ist, gewählt werden. Das Taktsignal darf hierbei ausschließlich von dem im Oszilloskop eingebauten Frequenzgenerator, nicht vom separaten Frequenzgenerator kommen.

Alle Gatter sind bereits korrekt mit der Versorgungsspannung (5 V) und dem Ground-Potenzial verbunden. Um die einzelnen Gatter zu komplexeren Schaltungen zu kombinieren stehen ihnen neben den Jumpern auch Laborkabel (2 mm-Stecker, blau und gelb) zur Verfügung. Damit können Sie die zu messenden Ausgangswerte auch auf die Ausgänge 0-7 im Bereich (G) legen, die sich über die Steckerleiste auch mit den digitalen Oszilloskop-Eingängen verbinden lässt. Hierfür liegt ein gesondertes Flachbandkabel bereit.

2 Vorbereitende Aufgaben

Bitte bearbeiten Sie die Vorbereitungsfragen sorgfältig vor dem Praktikum, da die Ergebnisse für eine erfolgreiche Bearbeitung der Versuche notwendig sind. Die Antworten können dabei direkt in diese Anleitung eingetragen werden und sind als Bestandteil des Protokolls am Versuchsende abzugeben. Sollten Sie noch offene Fragen haben, können Sie diese Ihrem Tutor während dem Kolloquium stellen.

2.1 Latches und Flip-Flops

Im Kapitel 1.2 und 1.3 haben Sie 6 verschiedene Latches und Flip-Flops kennengelernt. Füllen Sie für diese 6 Speichertypen die vorgefertigten Wahrheitstabellen in Tabelle 1 aus. Markieren Sie hierbei das Auftreten der unerwünschten Eingangskombination. Vervollständigen Sie mit Hilfe dieser Wahrheitstabellen anschließend im Timing-Diagramm in Abbildung 10 die positiven Ausgangssignale Q der genannten Speicher für die gegebenen Eingangssignale C , D , S , und R . Gehen Sie dabei davon aus, dass die Taktrate deutlich langsamer ist als die Gatterlaufzeit, sodass der Effekt der Gatterlaufzeit vernachlässigt werden kann.

Tabelle 1: Füllen Sie die Wahrheitstabellen für die Latches und Flip-Flops aus.

(a) ung. RS-Latch				(b) ung. D-Latch			(c) get. RS-Latch				
R	S	Q	\overline{Q}	D	Q	\overline{Q}	C	R	S	Q	\overline{Q}
0	0			0			0	d	d		
0	1			1			1	0	0		
1	0						1	0	1		
1	1						1	1	0		
							1	1	1		

(d) get. D-Latch				(e) RS-Flip-Flop						(f) D-Flip-Flop			
C	D	Q	\overline{Q}	C	R	S	Q	\overline{Q}		C	D	Q	\overline{Q}
0	d			↑	0	0				↑	0		
1	0			↑	0	1				↑	1		
1	1			↑	1	0				sonst	d		
				↑	1	1							
				sonst	d	d							

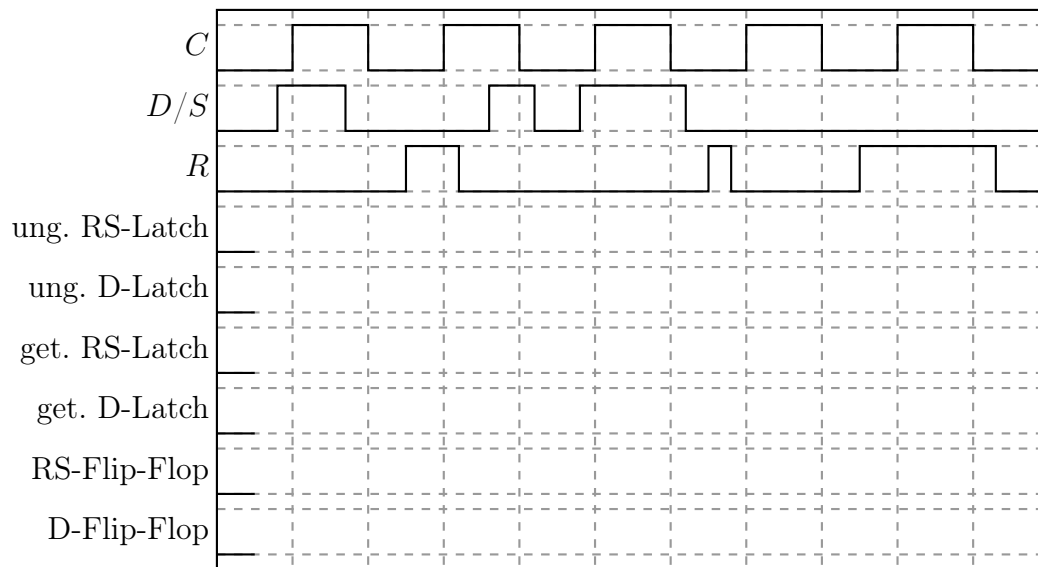


Abbildung 10: Vervollständigen Sie das Timing-Diagramm der 6 Speicher.

2.2 XOR mit 6 Eingängen

Wie können Sie ein XOR-Gatter mit 6 Eingängen aus einfachen XOR-Gattern mit 2 Eingängen erstellen? Zeichnen Sie das Blockschaltbild in Abbildung 11 ein. Achten Sie dabei auf eine Realisierung mit einer möglichst geringen Gatteranzahl und Gatterlaufzeit.



Abbildung 11: Zeichnen Sie hier das Blockschaltbild eines 6-fach XOR mit möglichst geringer Gatteranzahl und Gatterlaufzeit.

2.3 Asynchrone Zähler

Ergänzen Sie die D-Flip-Flops im Blockschaltbild in Abbildung 12 zu einem 4 Bit-Asynchrone Zähler. Was sind die Vor- und Nachteile eines Asynchrone Zählers gegenüber einem Synchronzähler?

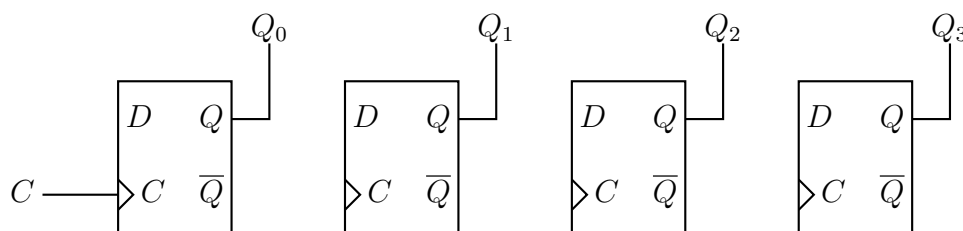


Abbildung 12: Vervollständigen Sie das Blockschaltbild zu einem Asynchronzähler

Vorteile: _____

Nachteile: _____

2.4 Synchronzähler

Entwerfen Sie einen 4 Bit-Synchronzähler mit der Zählreihenfolge $0 \rightarrow 3 \rightarrow 6 \rightarrow 9 \rightarrow 12 \rightarrow 0 \rightarrow \dots$. Füllen Sie für den Entwurf der Zählerlogik zunächst Tabelle 2 aus, welche die Zählstände, die Ausgänge Q_0 bis Q_3 und die jeweils benötigten Eingangswerte D_0 bis D_3 enthält. Ermitteln Sie daraus die vereinfachten logischen Eingangsfunktionen für D_0 bis D_3 mit Hilfe der KV-Diagramme in Abbildung 13. Die logische Funktion D_3 lässt sich auch mit einem einzelnen Gatter realisieren, wenn die richtigen Terme im KV-Diagramm ausgewählt werden und anschließend eine weitere Vereinfachung durchgeführt wird. Finden Sie diese Lösung?

Tabelle 2: Füllen Sie die Wahrheitstabelle für den Synchronzählers aus.

Zählerstand	Q_3	Q_2	Q_1	Q_0	D_3	D_2	D_1	D_0
0								
3								
6								
9								
12								

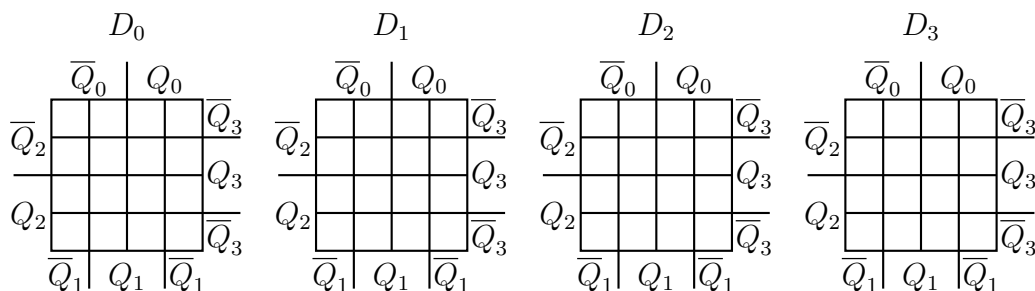


Abbildung 13: Vervollständigen Sie das KV-Diagramm des 4 Bit Synchronzählers

$D_0 =$ _____

 $D_1 =$ _____

 $D_2 =$ _____

 $D_3 =$ _____

2.5 Schieberegister

Ergänzen Sie die D-Flip-Flops im Blockschaltbild in Abbildung 14 zu einem 4 Bit Schieberegistregister mit serielllem Eingang und parallelen und seriellen Ausgängen.

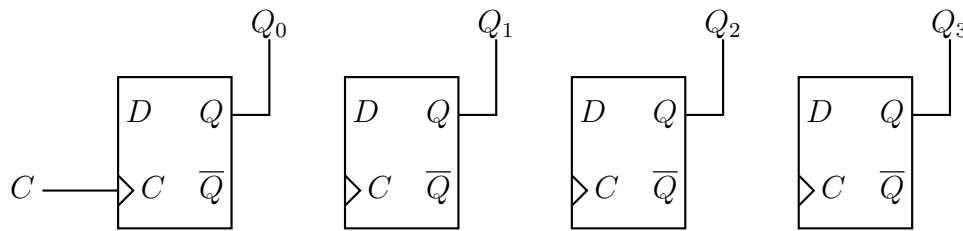


Abbildung 14: Vervollständigen Sie das Blockschaltbild zu einem 4 Bit Schieberegister.

2.6 Linear rückgekoppeltes Schieberegister

Das linear rückgekoppelte Schieberegister aus Abbildung 4 sei mit den Werten 100 vor-initialisiert. Geben Sie für die ersten 9 Takte den Inhalt des Schieberegisters sowie den Ausgangs- und Eingangswert in Tabelle 3 an.

Tabelle 3: Inhalte des linear rückgekoppelten Schieberegisters.

Takt	Inhalt	Ausgang	Eingang
0	100	0	1
1			
2			
3			
4			
5			
6			
7			
8			
9			

3 Versuchsdurchführung

Wichtige Hinweise

Bitte lesen Sie die folgenden Hinweise vor Versuchsbeginn gründlich durch. Eine Missachtung führt zum sofortigen Ausschluss aus dem Versuch.

- Verbinden Sie keine weiteren Kabel außer der Versorgungsspannung oder einem externen Taktsignal vom Oszilloskop mit dem *Digital-Board*, da dies das *Digital-Board* zerstören könnte.
- Das *Digital-Board* muss an CH2 der Spannungsquelle angeschlossen werden. Die Eingangsspannung beträgt 7,5 V, die Strombegrenzung muss auf maximal 0,5 A eingestellt werden. Überprüfen Sie diese Werte unbedingt vor Einschalten der Quelle. Durch einen Spannungsregler auf dem *Digital-Board* wird die interne Versorgungsspannung von 5 V für die Logikbausteine generiert.
- Bitte beachten Sie, dass der Schalter für die Spannungsversorgung stets ausgeschaltet ist, wenn Sie mit den Laborkabeln oder Jumpern neue Schaltungen aufbauen.
- Das extern eingespeiste Taktsignal darf ausschließlich vom Oszilloskop kommen, schließen Sie keinesfalls den separaten Frequenzgenerator an das *Digital-Board* an.
- Verwenden Sie zum Aufbauen der Schaltungen ausschließlich die bereitgestellten blauen und gelben Laborkabel sowie Jumper.

Führen Sie die folgenden Aufgaben gemäß Vorgaben durch und stellen Sie Ihre Ergebnisse der Aufgaben 3.2 – 3.6 in einem sauber angefertigten Protokoll dar. (Aufgabenteil 3.1 bearbeiten Sie direkt in der Anleitung).

3.1 Latches und Flip-Flops

Zu Beginn des Versuches sollen einfache Speicherzellen aus den grundlegenden Logikgattern aufgebaut werden, die Sie bereits im vorherigen Versuch kennengelernt haben.

3.1.1 Ungetaktete Latches

Bauen Sie ein ungetaktetes RS-Latch und ein ungetaktetes D-Latch nach Abbildung 1 aus NOR und NOT Gattern auf. Verwenden Sie zwei Schalter im Bereich (B) des *Digital-Boards* zur Erzeugung der Signale D/S und R und verbinden Sie die positiven Ausgänge Q mit den LEDs im Bereich (G). Versuchen Sie, die in Abbildung 15 dargestellte Signalfolge nachzuvollziehen und ergänzen Sie die Ausgangssignale Q der Latches. Die Verzögerungen durch Gatterlaufzeiten müssen nicht berücksichtigt werden. Das Timing ist ebenfalls nicht relevant, solange die vorgegebene Abfolge der Signale eingehalten wird.

3.1.2 Getaktete Latches

Erweitern Sie die beiden Latches nun mit Hilfe von AND-Gattern um einen Takteingang (vgl. Abbildung 2), den Sie mit einem weiteren Schalter verbinden. Vervollständigen Sie das Timing-Diagramm in Abbildung 16.

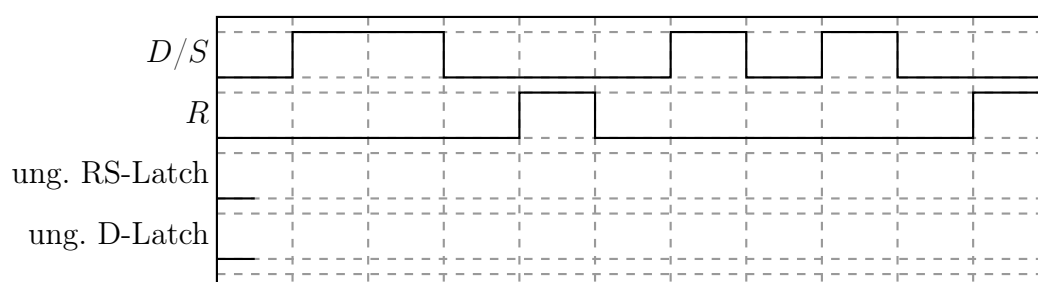


Abbildung 15: Vervollständigen Sie das Timing-Diagramm der ungetakteten Latches.

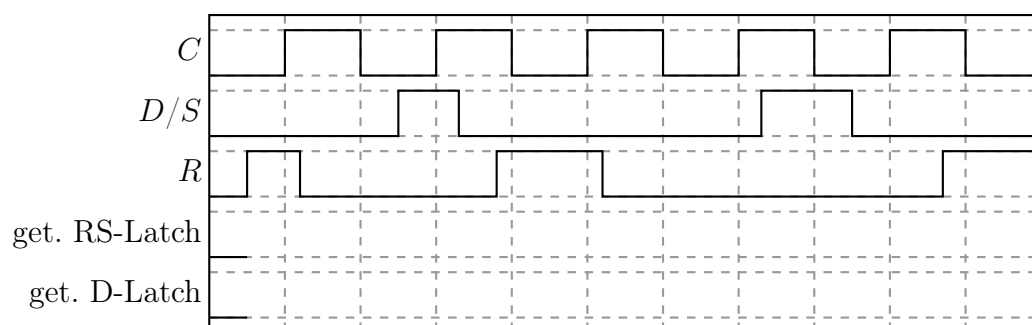


Abbildung 16: Vervollständigen Sie das Timing-Diagramm der getakteten Latches.

3.1.3 D-Flip-Flop

Realisieren Sie ein positiv flankengesteuertes D-Flip-Flop, wie in Abbildung 3b dargestellt aus den Einfachgattern NOR, AND und NOT. Die Eingangssignale sollen auch hier durch Schalter realisiert werden, der Ausgang soll auf eine LED im Bereich (G) gelegt werden. Tragen Sie die ermittelten Ausgangssignale des Flip-Flops in Abbildung 17 ein.

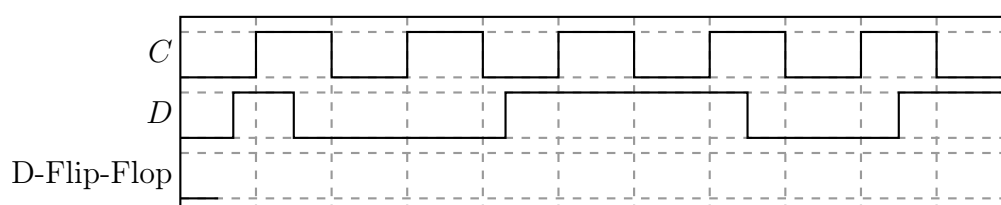


Abbildung 17: Vervollständigen Sie das Timing-Diagramm des D-Flip-Flops.

3.2 8 Bit-Asynchrnzähler

Wie Sie feststellen konnten, erfordert der Aufbau eines einzelnen Flip-Flops aus Logikgattern einen erheblichen Schaltungsaufwand. Um auch komplexer Schaltwerke realisieren zu können, verwenden wir ab jetzt die integrierten D-Flip-Flops im Bereich (F) des *Digital-Boards*. Entfernen Sie für diesen Versuch zunächst alle Jumper. Sie benötigen lediglich den Jumper für die Wahl des CLK -Tasters als Takteingangs und eine Verbindung des ersten Flip-Flops des Zählers mit dem zentralen Taktsignal.

3.2.1 Aufbau

Bauen Sie einen 8 Bit-Asynchrone Zähler nach dem Prinzip aus Vorbereitungsfrage 2.3 auf. Verwenden Sie den *CLK*-Taster zur Erzeugung eines Taktpulses und überprüfen Sie die Funktionalität. Um die höherwertigen Bits zu überprüfen können Sie den Zähler mit den *SET*-Tastern auch vorinitialisieren. Zeigen Sie anschließend den Aufbau Ihrem Tutor.

3.2.2 Zwischenwerte

Betrachten Sie nun die Ausgänge des Zählers auf dem Oszilloskop. Setzen Sie zunächst das Oszilloskop mittels **Default Setup** zurück. Verbinden Sie anschließend die Flip-Flops mit den Ausgängen 0-7 des im Bereich (G) und schließen Sie die Digitaleingänge des Oszilloskops an die Stiftleiste an. Aktivieren Sie im die Eingänge D_0 - D_7 im **Digital** Menü und gruppieren Sie diese als Hexadezimal-Bus. Achten Sie darauf, dass die richtigen Schwellwerte für die digitalen Eingänge (CMOS, 2,5 V) eingestellt sind. Stellen Sie zuletzt den **Trigger** auf einen geeigneten Eingang ein und wählen Sie die wechselnde Flanke, um sowohl bei steigender als auch bei fallender Flanke ein Triggerevent auszulösen. Der Triggermodus sollte mittels **Mode/Coupling** auf Normal gestellt werden, da im Auto-Modus auch bei fehlendem Triggerimpuls eine Aktualisierung durchgeführt wird.

Schauen Sie sich den Übergang vom Zählerstand $0111\,1111_{\text{bin}}$ (127_{dec} bzw. $7F_{\text{hex}}$) zum Zählerstand $1000\,0000_{\text{bin}}$ (128_{dec} bzw. 80_{hex}) an. Tragen Sie die Zwischenzustände des Asynchrone Zählers in das Timing-Diagramm in Abbildung 18 ein und nehmen Sie einen Screenshot vom Oszilloskop in Ihr Protokoll mit auf. Was ist der Grund für diese Zwischenzustände?

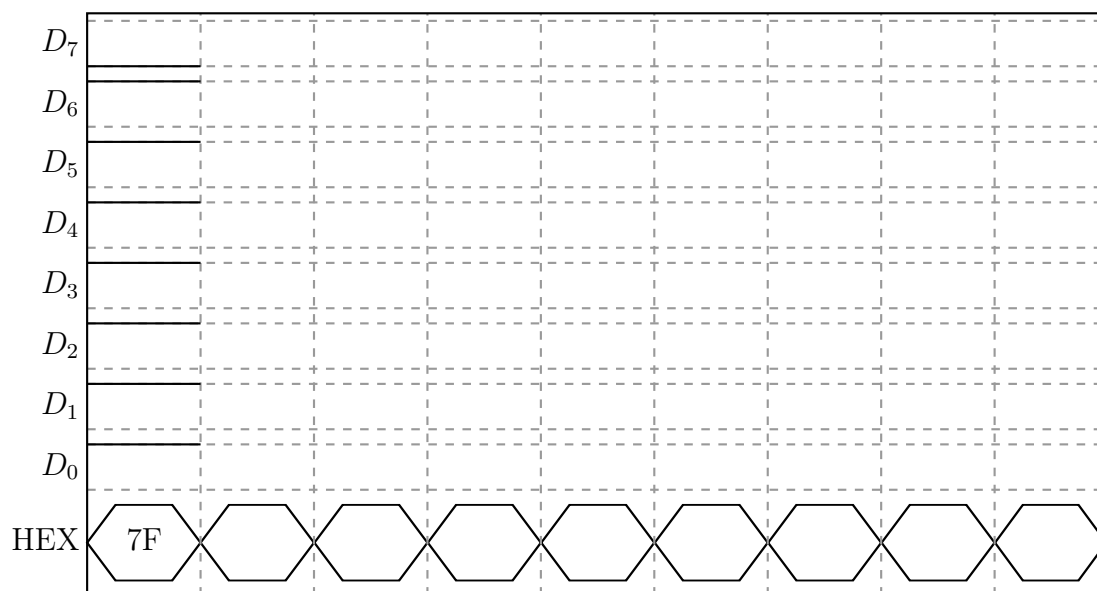


Abbildung 18: Timing-Diagramm der Zwischenwerte des Asynchrone Zählers

3.2.3 Gatterlaufzeit

Bestimmen Sie Zeit bis ein neuer (korrekter) Zählerstand an den Ausgängen anliegt und errechnen Sie daraus die Laufzeitverzögerung eines D-Flip-Flops.

3.3 4 Bit-Synchronzähler

Setzen Sie für diesen Versuch alle vertikalen Jumper auf das *Digital-Board*. Damit werden alle Flip-Flops mit dem zentralen Taktsignal versorgt. Verwenden Sie weiterhin den *CLK*-Taster als Takteingang.

3.3.1 Aufbau

Bauen Sie nun einen 4 Bit-Synchronzähler mit der Zählreihenfolge $0 \rightarrow 3 \rightarrow 6 \rightarrow 9 \rightarrow 12 \rightarrow 0 \rightarrow \dots$ auf und überprüfen Sie die Funktionalität. Verwenden Sie dazu die Ergebnisse von Vorbereitungsfrage 2.4. Zeigen Sie den Aufbau Ihrem Tutor.

3.3.2 Übergänge

Betrachten Sie auch hier die vier Bits des Zählers auf dem Oszilloskop. Schauen Sie sich die Übergänge der Zählerstände an. Im Gegensatz zum Asynchronzähler sollten die Übergänge beim Synchronzähler gleichzeitig stattfinden. Dennoch werden Sie bei mehrfacher Wiederholung des Versuches feststellen, dass die Übergänge unter Umständen zu minimal unterschiedlichen Zeitpunkten stattfinden. Dies ist auf verschiedene Ursachen zurückzuführen. Neben leicht unterschiedlichen Laufzeiten der einzelnen D-Flip-Flops hängt dies primär mit der Datenerfassung der digitalen Eingänge des Oszilloskops zusammen. Ein limitierender Faktor ist die minimale Zeitauflösung des Oszilloskops. Was ist die maximale Zeitliche Auflösung des Oszilloskops, oder anders ausgedrückt der minimale Laufzeitunterschied den Sie beobachten können?

Zudem stellt der digitale Eingang das Signal stark idealisiert dar, da ein einfacher Schwellwertvergleich vorgenommen wird. Liegt das Eingangssignal unterhalb des eingestellten Referenzpegels (bei CMOS 2,5 V), so wird eine 0 erkannt, oberhalb eine 1. Tatsächlich findet dagegen ein annähernd exponentieller Lade- bzw. Entladevorgang statt, die Zeitkonstante wird dabei primär von den parasitären Kapazitäten des folgenden Gatters sowie der Eingangskapazität des Oszilloskopeingangs bestimmt.

3.4 Schieberegister

3.4.1 Ringzähler

Bauen Sie ein 8 Bit-Schieberegister auf, indem Sie die horizontalen Jumper verwenden und verbinden Sie den Ausgang des letzten Flip-Flops mit dem Eingang des ersten (8 Bit-Ringzähler). Überprüfen Sie die Funktionalität, indem Sie ein beliebiges Bitmuster „durchschieben“. Zeigen Sie den Aufbau Ihrem Tutor.

3.4.2 Frequenzteiler

Der von Ihnen aufgebaute Ringzähler kann, abhängig vom Anfangszustand des Schieberegisters, auch als Frequenzteiler genutzt werden. Betreiben Sie den Ringzähler hierzu mit dem Frequenzgenerator aus dem Oszilloskop. Der Frequenzgenerator wird mit der Taste Wave Gen aktiviert. Stellen Sie die Form auf Rechteck, die untere Spannung auf 0 V und die obere Spannung auf 5 V ein, der Arbeitszyklus sollte 50 % betragen.

Wenn Sie die richtigen Werte eingestellt haben verbinden Sie den Ausgang *Gen. Out* mit dem BNC-Eingang des *Digital-Boards* und wählen Sie den BNC-Eingang mit dem Jumper aus. Mehrfaches Drücken der Taste Wave Gen aktiviert und deaktiviert das

Taktsignal. Stellen Sie nun eine Frequenz von 10 Hz ein und stellen Sie diesen Takt sowie den Ausgang des letzten Flip-Flops auf dem Oszilloskop dar. Untersuchen Sie die fünf Anfangszustände aus Tabelle 4 im Hinblick auf die Frequenzteilung. Messen Sie hierbei die Frequenz des Ausgangssignals und leiten Sie daraus die Frequenzteilung sowie den Arbeitszyklus ab.

Tabelle 4: Messergebnisse des Frequenzteilers.

Anfangszustand	Ausgangsfrequenz	Frequenzteilung	Arbeitszyklus
11110000			
11001100			
10101010			
10000000			
10001000			

3.5 DVB-S Scrambler

Bauen Sie das linear rückgekoppelte Schieberegister des DVB-S Scramblers auf, das im grau hinterlegten Teil in Abbildung 5 dargestellt ist. Wie kann das Scrambling auf der Empfängerseite rückgängig gemacht werden? Wie lautet das Boolesche Gesetz für diesen Vorgang? (Tipp: Erinnern Sie sich an den vorherigen Versuch)

Ermitteln Sie die ersten 20 Werte der PN-Sequenz. Achten Sie auf eine korrekte Vorinitialisierung des Registers, die in Abbildung 5 oberhalb jeder Zelle angegeben ist.

3.6 GPS C/A-Code Generator

Bauen Sie den C/A-Code-Generator des GPS-Systems auf, das im grau hinterlegten Teil in Abbildung 6 dargestellt ist. Beginnen Sie mit dem G1-Shift-Register nach Abbildung 7 und dem G2-Shift-Register nach Abbildung 8. Schalten Sie als letztes die Verknüpfungen zur Generierung der G_i -Sequenz gemäß Abbildung 6. In Tabelle 5 finden Sie die entsprechenden Werte für die Phase-Select-Logic der Satelliten 5, 13 und 17. Ermitteln Sie für diese Satelliten jeweils die ersten 10 Werte der PN-Sequenz. Beide Register müssen dabei jeweils mit Einsen vorinitialisiert werden.

Tabelle 5: GPS C/A Code

Satellit	Phase-Select-Logic	PN-Sequenz
5	$1 \oplus 9$	
13	$6 \oplus 7$	
17	$1 \oplus 4$	

Evaluation

Auf der Moodleseite gibt es zu jedem Versuch eine Umfrage zur Evaluation. Bitte helfen Sie die Qualität des Praktikums zu verbessern, indem Sie uns Rückmeldung geben wie Sie die Versuchsdurchführung empfunden haben!

Loggen Sie Sich beide einzeln vom Praktikumsrechner aus ein und beantworten Sie die Umfrage noch direkt nach der Versuchsdurchführung bevor Sie das Praktikum verlassen!