

3-BIT VOLLADDIERER

MIT EINZELNEM EINGABE-DATENBUS

Simon C. Leemann, Abteilung für Physik

Versuch: Digitale Elektronik

November 1998

Zusammenfassung

In diesem Bericht soll aufgezeigt werden, wie man unter Verwendung der CMOS-ICs MC14008B und MC14042B (zwei Stück) einen einfachen 3bit-Addierer mit individuellem Eingabe-Datenbus und einem Ausgabe-Datenbus bauen kann. Ebenso sollte verständlich werden, weshalb man ausgehend von dieser Schaltung ebenso einen 4bit-Volladdierer des selben Typs herstellen könnte. Für den einfachen Nachbau empfiehlt sich Fig. 11 (letzte Seite), wo auch gleich alle Pin-Anschlüsse und ihre Belegung angegeben ist.

Problemstellung

Im Versuch „Digitale Elektronik“ soll man den Umgang mit digitalen Schaltkreisen lernen. Nachdem einige Begriffe und Methoden kennengelernt wurden, sollte man einfache Schaltungen bauen und ihre Funktionsweise nachvollziehen.

Im Anschluss daran baut man in Absprache mit dem Assistenten eine eigene Schaltung. Ich habe dafür den 3-bit Volladdierer gewählt, wobei der Assistent vorgab, dass ich eine sequentielle Schaltung baue mit einem einzigen Eingabe-Datenbus. Der Addierer addiert dabei zwei 3-bit Binärzahlen und gibt eine 4-bit Binärzahl als Resultat aus. Der einzelne Eingabe-Datenbus entspricht dabei der Vorstellung, dass man einen Addierer als Bestandteil eines komplexeren Schaltelements (bspw. Mikroprozessor) verwendet und darauf angewiesen ist, dass der Input an einer einzelnen Schnittstelle zu erfolgen hat.

Problemlösung

Unter einem Halbaddierer, versteht man eine logische Einheit, die zwei Zahlen addiert und das Resultat ausgibt. Im Gegensatz dazu ist ein Volladdierer in der Lage, einen Übertrag zuzuzählen und anschliessend einen (allfälligen) Übertrag mit dem Resultat auszugeben. Dies ermöglicht es, einen Kaskade von einzelnen Volladdierern aufzubauen, die sowohl mehrstellige Zahlen addiert als auch ein mehrstelliges Resultat ausgibt.

Grundsätzlich ist die Addition zweier Bits durch die Wahrheitstafel in Figur 1 dargestellt. Es ist zu beachten, dass die beiden Summanden mit A bzw. B, die Summe mit S und der Übertrag mit U bezeichnet werden.

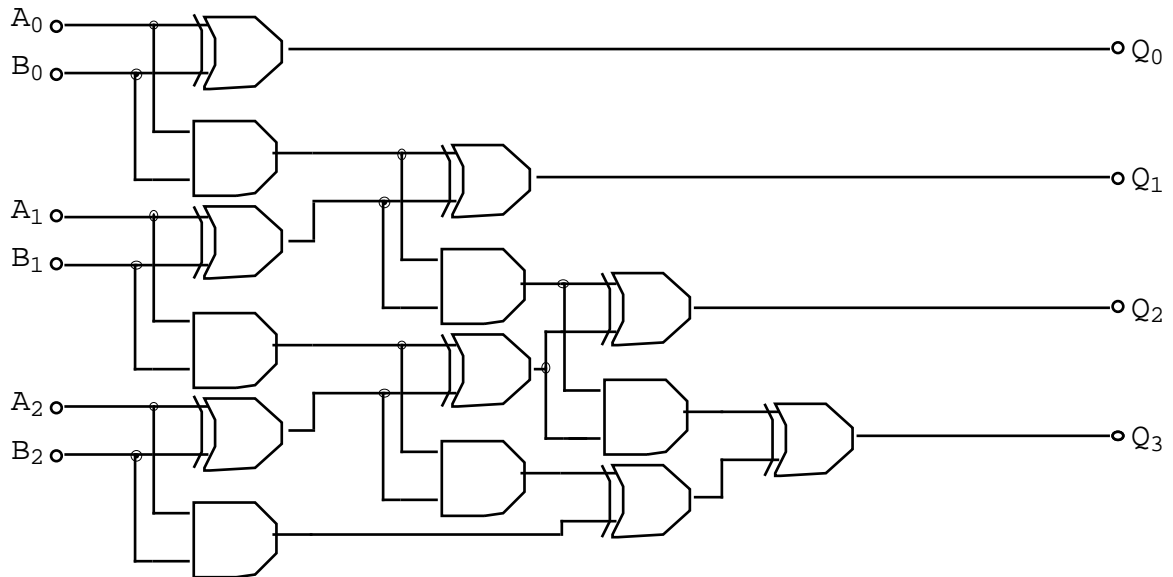
A	B	U	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Figur 1: Die Wahrheitstafel der Addition zweier Bits .

Aus Figur 1 ist sofort ersichtlich, dass $U = (A \wedge B) = \text{AND}(A,B)$ und $S = [(A \wedge \neg B) \vee (\neg A \wedge B)] = \text{XOR}(A,B)$. Beide Gatter sind als Bestandteile von CMOS-ICs zu finden (ich verwende in dieser Arbeit ausschliesslich CMOS-Bauteile mit +10 V DC Speisung).

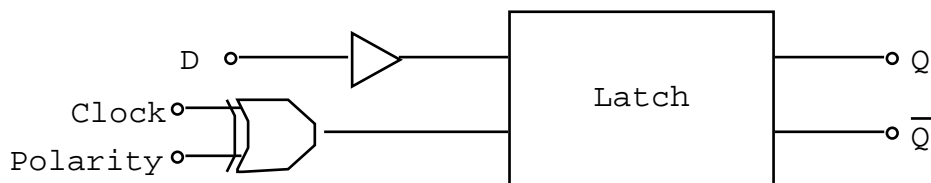
Wie bereits angetönt, können durch eine Kaskadenschaltung mehrstellige Zahlen summiert werden. Dazu wird das Übertragsausgang des 1. Volladdierers (d.h. derjenige des 2^0 -bits) an den Übertragseingang des Addierers für das nächste Bit angeschlossen; der Übertragsausgang dieses Addierers wird wiederum an den nächsten Übertragseingang angeschlossen usw. Schliesslich bekommt man mit allen Summenbits und dem letzten Übertragsausgang das Resultat. Dieses Prinzip ist in Figur 8 ersichtlich.

Die Schaltung für einen 3-bit Volladdierer kann rein kombinatorisch, d.h. durch Verkettung logischer Elemente gelöst werden. Dabei ist man aber angewiesen auf zwei getrennte Eingabe-Datenbusse. Diese Schaltung ist in Figur 2 dargestellt.



Figur 2: Das Schaltschema für einen kombinatorischen „3bit-Volladdierer“ (allerdings ohne Übertragseingang!).

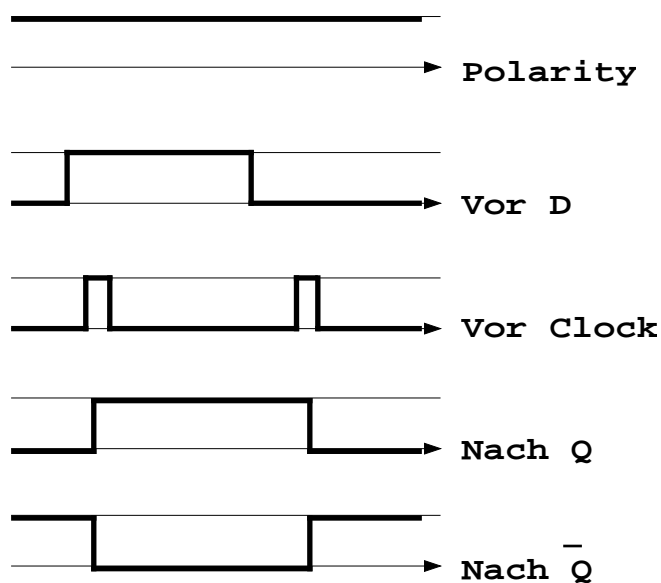
Im Gegensatz zu der oben genannten kombinatorischen Schaltung, soll bei dieser Arbeit eine sequentielle Schaltung mit einem einzigen Eingabe-Datenbus gebaut werden. Das heisst, dass die beiden Summanden nacheinander über denselben Bus eingegeben werden und über zwei Speicherelemente (Latches) zum Addierer gelangen. Für das Beschreiben der Latches muss ein Clocksignal benutzt werden: Wird ein Clock-Signal an eine Latch gesendet, so nimmt diese die Daten an ihrem Eingang auf, speichert sie und stellt sie (bis zum nächsten Clock-Signal) an ihrem Ausgang zur Verfügung. In Figur 3 ist das Blockdiagramm für eine Latch skizziert. Die Latch ist aus praktischen Gründen noch mit einem Polarity-Eingang ausgestattet, damit man bestimmen kann, ob der Clock-Impuls High oder Low ist. Die Wahrheitstafel für die Latch, insbesondere mit verschiedener Polarity, ist in Figur 4 dargestellt. In Figur 5 und Figur 6 sind zwei Zeitdiagramme für eine Latch angegeben.



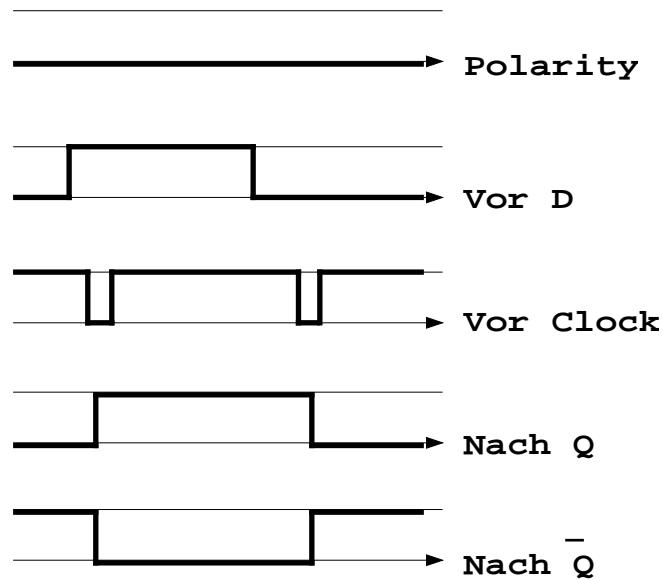
Figur 3: Blockdiagramm für eine Latch. Beachte, dass D der Dateneingang und Q der Datenausgang ist.

Clock	Polarity	Q
0	0	Data
1	0	Latch
0	1	Data
1	1	Latch

Figur 4: Wahrheitstafel einer Latch. Data ist kurz für Data Feed, also das Einlesen und Speichern der am Eingang anliegenden Daten. Latch bezeichnet das unveränderte Verhalten des Speicherelements (d.h. sie speichert immernoch die zuletzt eingelesenen Daten und stellt diese am Ausgang zur Verfügung).

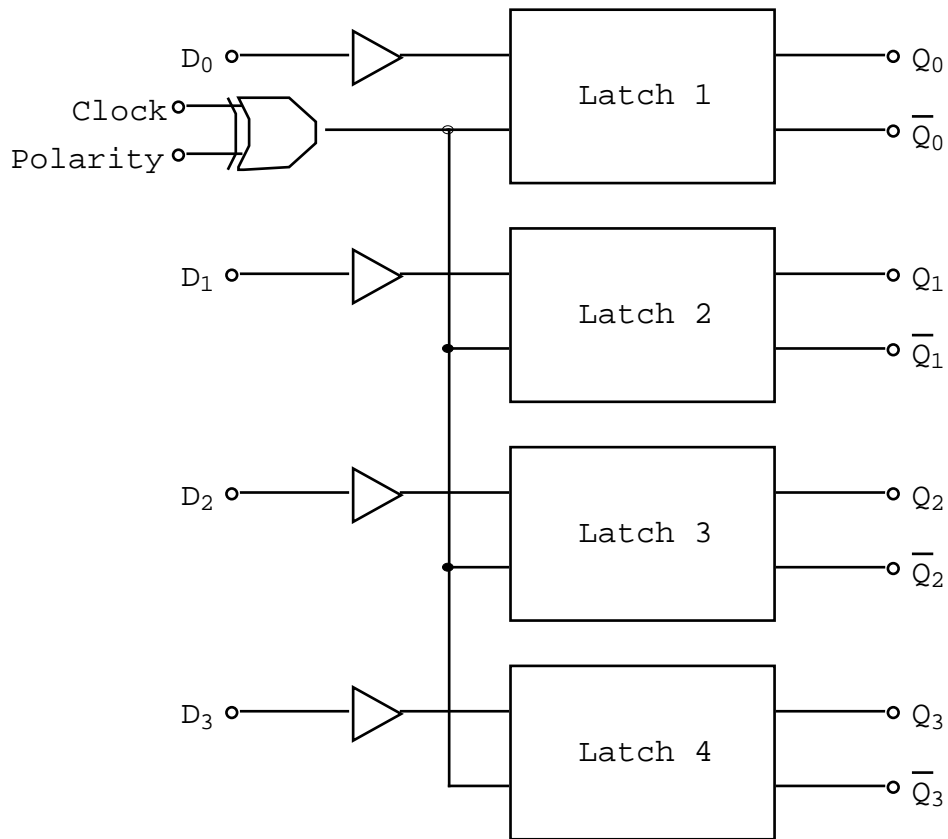


Figur 5: Zeitdiagramm für eine Latch. In diesem Beispiel ist die Polarity auf High gestellt, d.h. das Clocksigna ist High gem. Wahrheitstafel in Fig. 4. Zu Beginn speichert die Latch ein Low. Am Dateneingang wird ein High angelegt und die Clock getriggert. Dadurch nimmt die Latch das High auf und stellt es am Ausgang zur Verfügung. Der Dateneingang wird danach auf Low umgestellt und die Clock wieder getriggert. Dadurch nimmt die Latch das Low an und stellt es an ihrem Ausgang zur Verfügung.



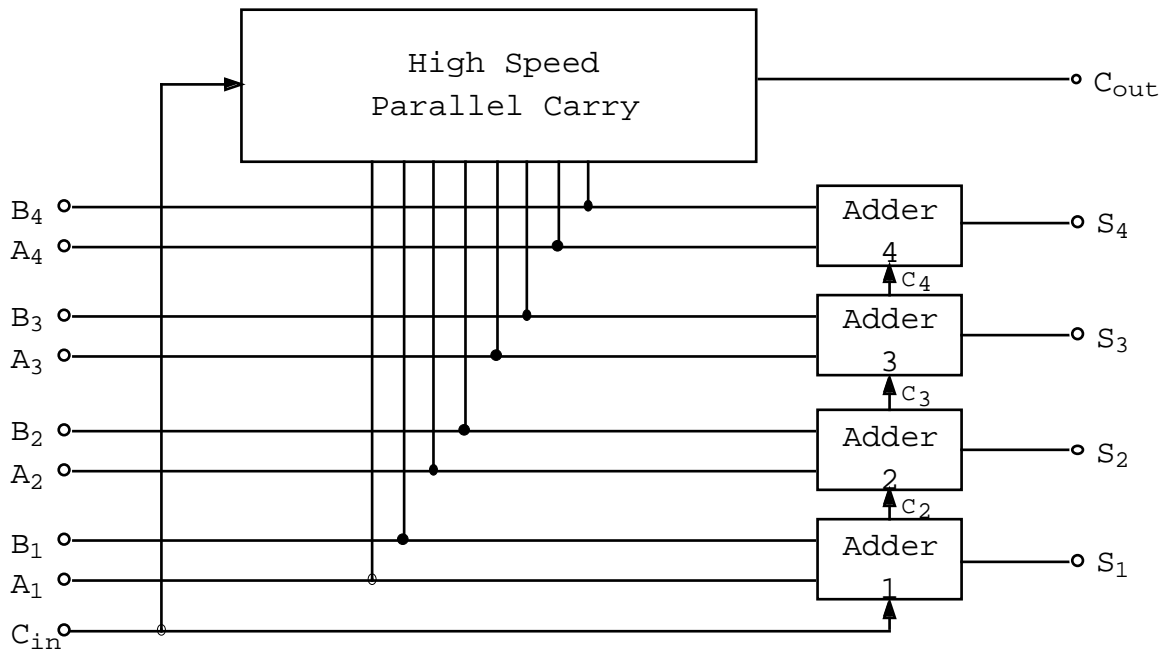
Figur 6: Dasselbe Beispiel wie in Fig. 5 mit Ausnahme, dass hier die Polarity auf Low gestellt ist, d.h. das Clocksignal ist Low.

In meiner Schaltung verwende ich zwei CMOS-IC MC14042B um beide Summanden zwischenzuspeichern. Dieser IC ist eine Quad Transparent Latch, also ein Speicherchip mit 4 Dateneingängen, einer gemeinsamen Clock und Polarity sowie 4 Datenausgängen. Das Blockdiagramm ist in Figur 7 aufgezeigt.



Figur 7: Blockdiagramm für MC14042B.

In meiner Schaltung verwende ich einen einzigen IC für das ganze Addierwerk. Der CMOS-IC MC14008B ist ein 4-bit Volladdierer. Das Blockdiagramm dafür ist in Figur 8 zu finden. Beachte, dass für die Summation der Überträge die High Speed Parallel Carry Unit zuständig ist. Mittels Übertragseingang C_{in} und Übertragsausgang C_{out} kann dieser Chip selbst in eine Addierer-Kaskade eingebaut werden.



Figur 8: Blockdiagramm für MC14008B.

Die Wahrheitstafel für einen einzelnen Addierer (in Fig. 8 mit Adder bezeichnet) des ICs ist in Figur 9 abgebildet. Beachte, dass hier nur 2 Bits sowie ein Übertragsbit summiert werden. Die Ausgabe der Summe bräuchte also S_i als erstes und C_{i+1} als zweites Bit.

C_i	A_i	B_i	C_{i+1}	S_i
0	0	0	0	0
0	1	0	0	1
0	0	1	0	1
0	1	1	1	0
1	0	0	0	1
1	1	0	1	0
1	0	1	1	0
1	1	1	1	1

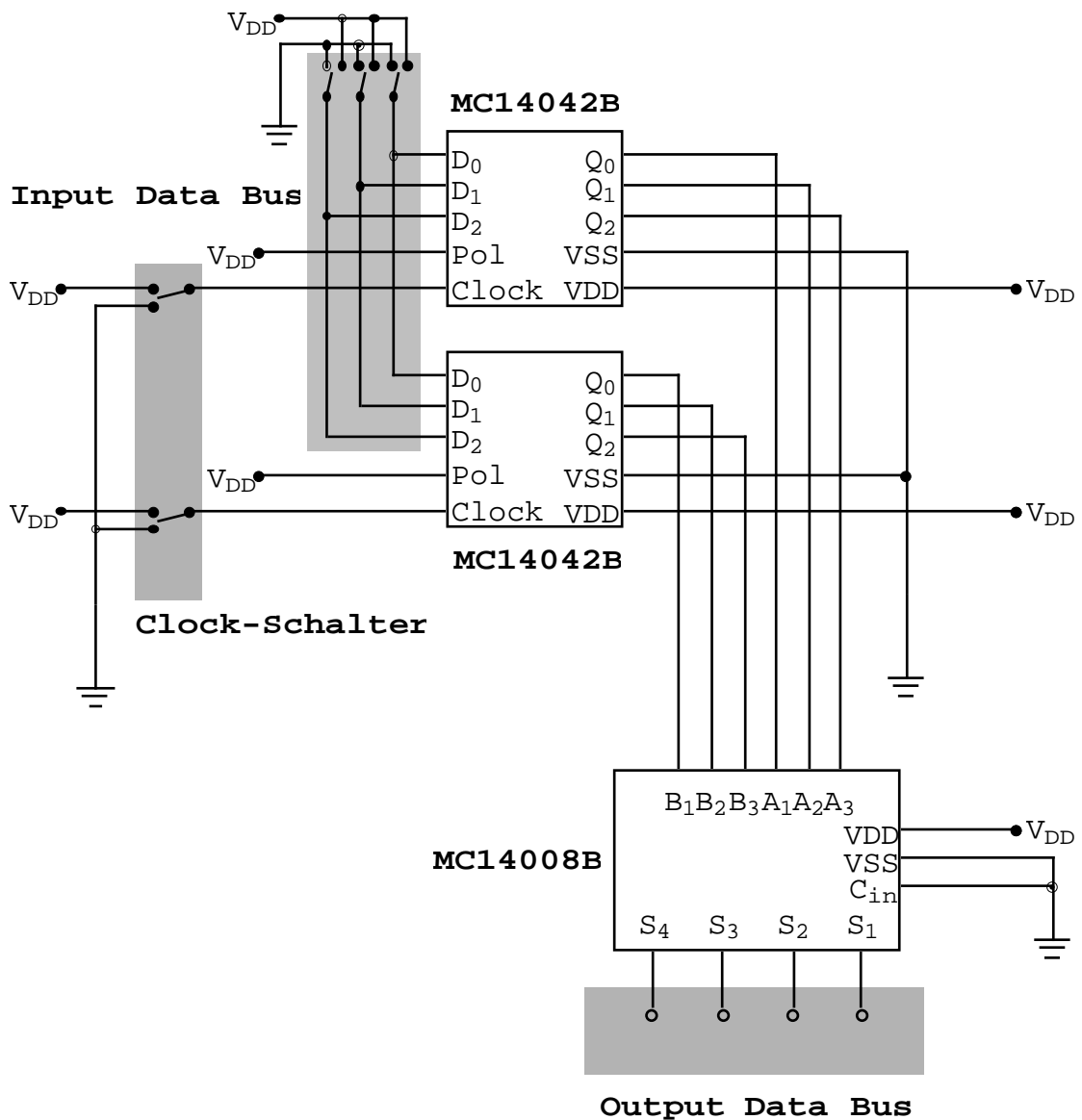
Figur 9: Wahrheitstafel für einen einzelnen Addierer des MC14008B.

Der MC14008B entspricht in der Funktionsweise aber exakt einer kombinatorischen Addierschaltung wie in Fig. 2 gezeigt.

Zur Eingabe der Summanden habe ich einfache Schalter verwendet, die den Vorteil haben, dass man die eingegebenen Ziffern auch im Nachhinein sofort ablesen kann. Die Clocksignale habe ich ebenso mit Schaltern eingegeben, wobei sich hier eindeutig Tastenschalter (die nicht einrasten!) anbieten, da nur ein kurzer Puls auf den Clockeingang nötig ist.

Es folgt nun in Figur 10 das Schaltschema für den ganzen 3-bit Volladdierer mit

einzelnen Eingabe-Datenbus. Beachte, dass die Angaben zu den Ein-/Ausgang der ICs denjenigen in den obigen Blockdiagrammen entsprechen.

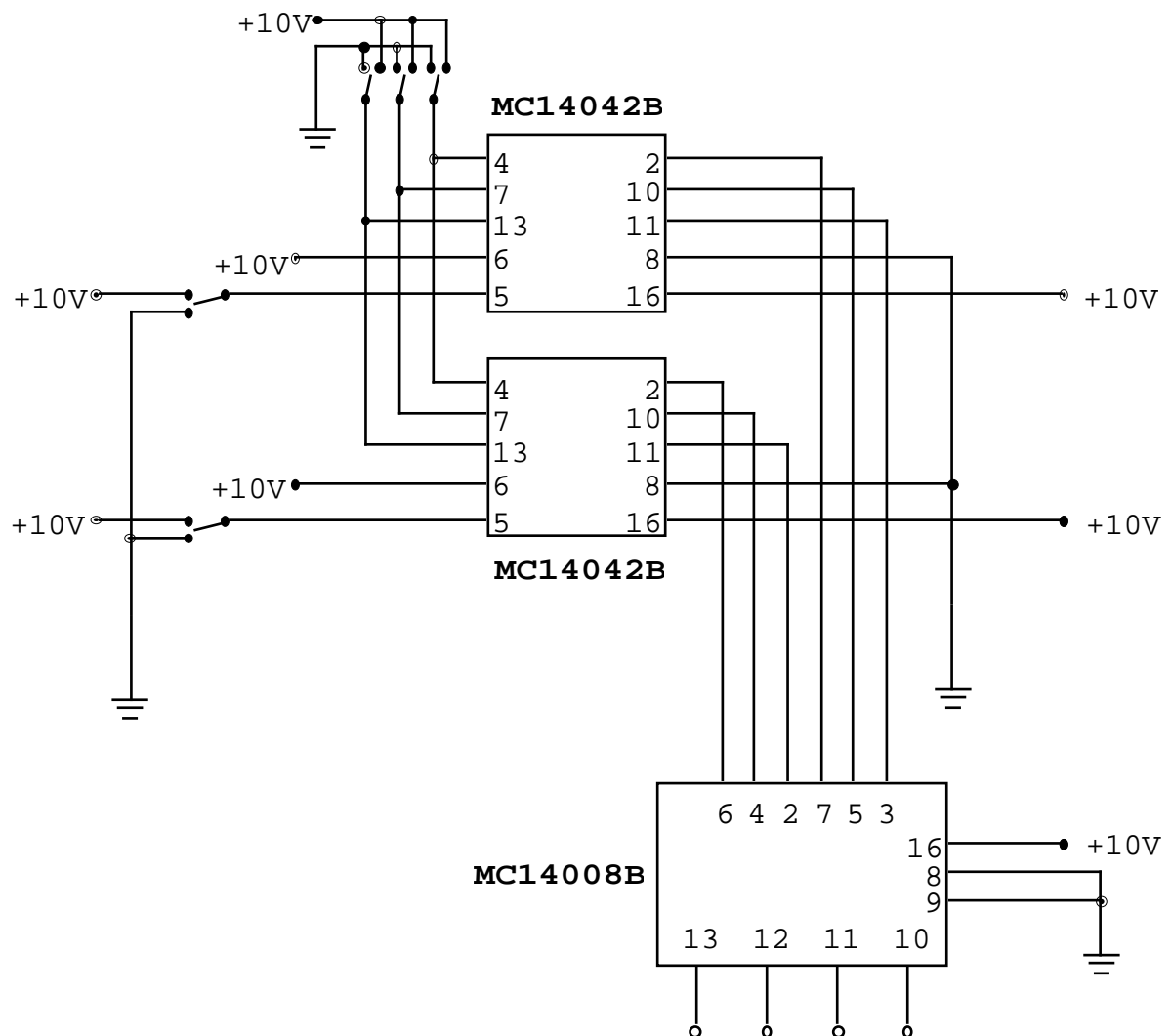


Figur 10: Schaltschema des 3-bit Volladdierers mit individuellem Eingabe-Datenbus. Die Einfärbung und Beschriftung dient lediglich zur Veranschaulichung der besonderen Funktionseinheiten. Beachte, dass V_{DD} und V_{SS} die Betriebsspannungsanschlüsse der ICs sind. Offensichtlich ist auch, dass jeweils nur drei Eingänge der Latch-ICs und des Addierers benutzt wurden; dies ist natürlich darauf zurückzuführen, dass nur ein 3bit-Volladdierer gebaut wurde. Es wäre also ein Leichtes anhand der verwendeten Bauelemente auf 4bit zu erweitern; dazu müsste aber noch das Übertragsausgang des Addierers als fünftes Summenbit verwendet werden.

Zuletzt wird das Ergebnis auf den Ausgängen S_1 - S_4 des Addierers ausgegeben. Man ist natürlich versucht dieses Ergebnis in einer gut lesbaren Form auszugeben. Ich habe dies mit den zur Verfügung stehenden BCD-Anzeigen (7bit-Segmentanzeige) umgesetzt. Ich möchte aber darauf hinweisen, dass diese Anzeigen für meine Arbeit nicht vollständig befriedigen, da nur Zahlen zwischen 0 und 9 angezeigt werden können. Sicherlich könnte man dieses Problem mit einer umfangreicheren Anzeige einfach lösen. Zwecks Konzentration auf die Problemstellung sehe ich allerdings von der zugehörigen Dokumentation ab.

Ergebnisse

Unter Verwendung der beschriebenen Elemente lässt sich auf einfache Art und Weise ein 3bit-Volladdierer mit individuellem Eingabe-Datenbus aufbauen. Die Summe auf dem Ausgabe-Datenbus kann mit Hilfe von Anzeigen in einer simplen Darstellung angezeigt werden. Zuletzt möchte ich auf Figur 11 hinweisen, wo der interessierte Bastler ein abgewandeltes Schaltschema mit Pin-Anschlüssen aller ICs findet und mit dessen Hilfe er in kürzester Zeit meine Schaltung nachbauen kann. Für zusätzlich Interessierte und aufbauende Schaltungen sei auf das Motorola™ CMOS Logic Data Buch hingewiesen, wo man alle relevanten Daten zu den benutzten ICs findet. Einige der Darstellungen dieses Berichts entstammen diesem Manual und weitere sind dort zu finden.



Figur 11: Schaltschema mit Pin-Anschlüssen aller Bauteile.