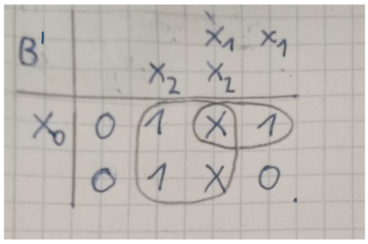
HWP II – Versuch 6

Tristan Sopauschke, Hans Schreiter

Aufgabe 4.1 – Würfel mit Zähler

Seien x_2, x_1, x_0 die drei LSBs des Zustandes des Digitalzählers, so sind die Wertetabellen und Ansteuerfunktionen für die Würfelanzeige die folgenden:

D’	C’	B’	A’	x_2	x_1	x_0
1	0	0	0	0	0	0
0	1	0	0	0	0	1
1	1	0	0	0	1	0
0	1	1	0	0	1	1
1	1	1	0	1	0	0
0	1	1	1	1	0	1
X	X	X	X	1	1	0
X	X	X	X	1	1	1



$$D' = \neg x_0$$
$$C' = \neg(\neg x_2 \wedge \neg x_1 \wedge \neg x_0)$$
$$B' = \neg x_2 \wedge x_1 \wedge x_0 \vee x_2 \wedge \neg x_1 \wedge \neg x_0 \vee x_2 \wedge \neg x_1 \wedge x_0 = x_2 \vee x_1 \wedge x_0$$
$$A' = x_2 \wedge x_0$$

Da die Würfelanzeige low-aktiv ist, müssen die Ansteuerfunktionen noch einmal invertiert werden. Die tatsächlichen Funktionen sind demnach:

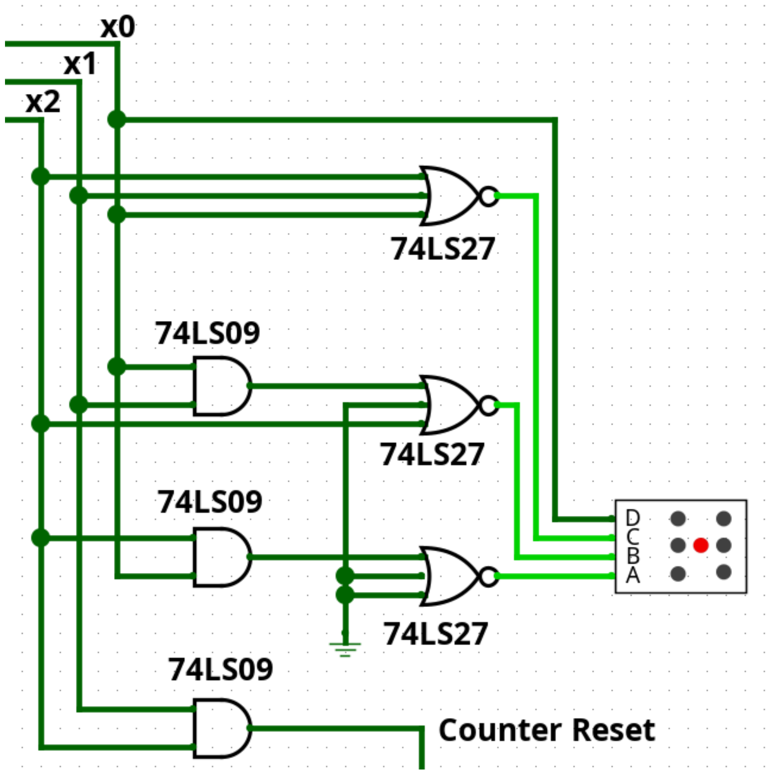
$$D = x_0$$
$$C = \neg x_2 \wedge \neg x_1 \wedge \neg x_0$$
$$B = \neg(x_2 \vee x_1 \wedge x_0)$$
$$A = \neg(x_2 \wedge x_0)$$
$$R = x_2 \wedge x_1$$

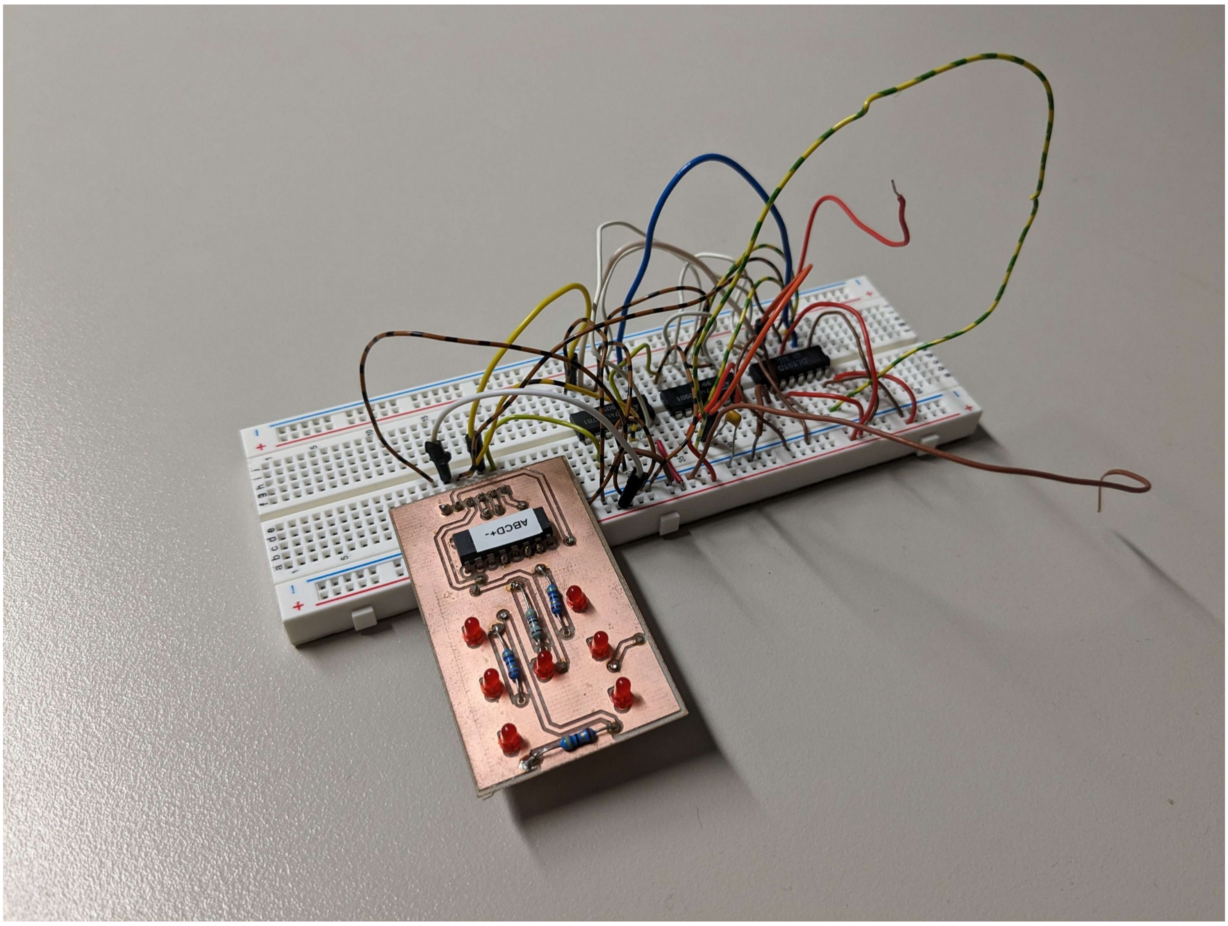
Zusätzlich muss eine Funktion für den Reset gefunden werden. Der Reset und damit Übergang zurück in Zustand 000 soll in dem Moment stattfinden, in dem der Zustand 110 betreten werden würde. Somit ergibt sich die oben aufgeführte Funktion R, deren Ausgang an den “CLR” Pin des von uns verwendeten DL193D gelegt wird.

Durch geschicktes Umrechnen und ein gutes Stück ausprobieren, ist es dann unverhofft tatsächlich möglich dieses Schaltwerk mit nur zwei (der uns zur Verfügung stehenden) ICs zu implementieren. Dazu wird jeweils ein 74LS27 (3x 3in NOR) und ein 74LS09 (4x 2in AND) genutzt.
Hier noch einmal die Funktionen unter expliziter Nutzung dieser Gatter:

$$D = \underline{x_0}$$
$$C = \neg x_2 \wedge \neg x_1 \wedge \neg x_0 = \neg(x_2 \vee x_1 \vee x_0) = \underline{\downarrow(x_2, x_1, x_0)}$$
$$B = \neg(x_2 \vee x_1 \wedge x_0) = \underline{\downarrow(x_2, (x_1 \wedge x_0), 0)}$$
$$A = \neg(x_2 \wedge x_0) = \underline{\downarrow(x_2 \wedge x_0, 0, 0)}$$
$$R = \underline{x_2 \wedge x_1}$$

Der Schaltplan sieht dann in etwa so aus.
Durch Nutzen von Logisim lässt sich die Korrektheit von diesem auch direkt verifizieren.

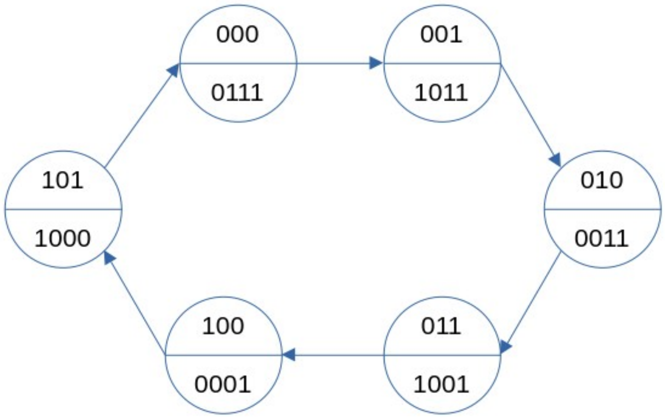




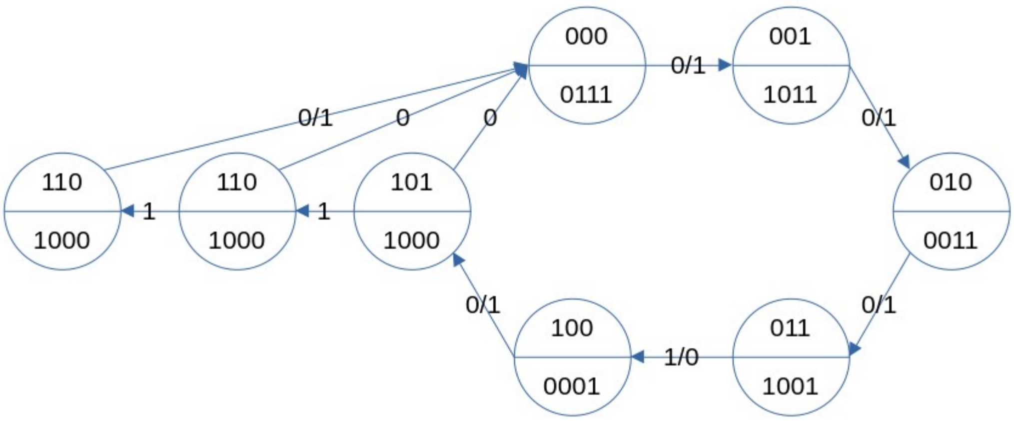
Aufgabe 5.1 - Würfel mit GAL

Zunächst muss der Automat, den wir zuvor mit dem Zählerchip realisiert haben, nun vernünftig formalisiert, ein wenig erweitert und anschließend eine Schaltwerksynthese durchgeführt werden.

Der Automat ist zunächst nichts weiter als ein simpler Modulo-6-Zähler, dessen Zuständen jeweils die passenden Bitmuster für die zugehörigen Augenzahlen als Ausgabe zugeordnet werden. Der resultierende Moore-Automat sieht folgendermaßen aus:



Nun soll bei Knopfdruck die Augenzahl 6 mit höherer Wahrscheinlichkeit auftreten. Eine Möglichkeit, dies zu realisieren, ist dem Automaten zwei weitere Zustände hinzuzufügen, die ebenfalls das Bitmuster für die Augenzahl 6 ausgeben. Diese neuen Zustände werden nur dann durchlaufen, wenn die Eingabe gleich 1 (Knopf gedrückt) ist.



Nun geht es an die Schaltwerksynthese. Da wir dem Automaten zwei neue Zustände hinzugefügt haben, müssen wir neben den Übergangsfunktionen auch neue Ausgabefunktionen berechnen.

x	$\mathbf{q_2}$	$\mathbf{q_1}$	$\mathbf{q_0}$	$\mathbf{q_2^+}$	$\mathbf{q_1^+}$	$\mathbf{q_0^+}$	**D**	**C**	**B**	**A**
0	0	0	0	0	0	1	0	1	1	1
0	0	0	1	0	1	0	1	0	1	1
0	0	1	0	0	1	1	0	0	1	1
0	0	1	1	1	0	0	1	0	0	1
0	1	0	0	1	0	1	0	0	0	1
0	1	0	1	0	0	0	1	0	0	0
0	1	1	0	0	0	0	1	0	0	0
0	1	1	1	0	0	0	1	0	0	0
1	0	0	0	0	0	1				
1	0	0	1	0	1	0				
1	0	1	0	0	1	1				
1	0	1	1	1	0	0				
1	1	0	0	1	0	1			~	
1	1	0	1	1	1	0				
1	1	1	0	1	1	1				
1	1	1	1	0	0	0				

Da wir mit einem Moore-Automaten arbeiten, sind die Ausgabefunktionen D, C, B, A per Definition unabhängig vom Eingabesymbol x.

Es ergeben sich (mit KV-Diagramm vereinfacht) die folgenden Übergangsfunktionen in DNF:

$q_2^+ = \neg q_2 \wedge q_1 \wedge q_0 \;\vee\; q_2 \wedge \neg q_1 \wedge x \;\vee\; q_2 \wedge \neg q_1 \wedge \neg q_0 \;\vee\; q_2 \wedge \neg q_0 \wedge x$

$q_1^+ = \neg q_1 \wedge q_0 \wedge x \;\vee\; \neg q_2 \wedge \neg q_1 \wedge q_0 \;\vee\; q_1 \wedge \neg q_0 \wedge x \;\vee\; \neg q_2 \wedge q_1 \wedge \neg q_0$

$q_0^+ = \neg q_0 \wedge x \;\vee\; \neg q_2 \wedge \neg q_0 \;\vee\; \neg q_1 \wedge \neg q_0$

Und die Ausgabefunktionen:

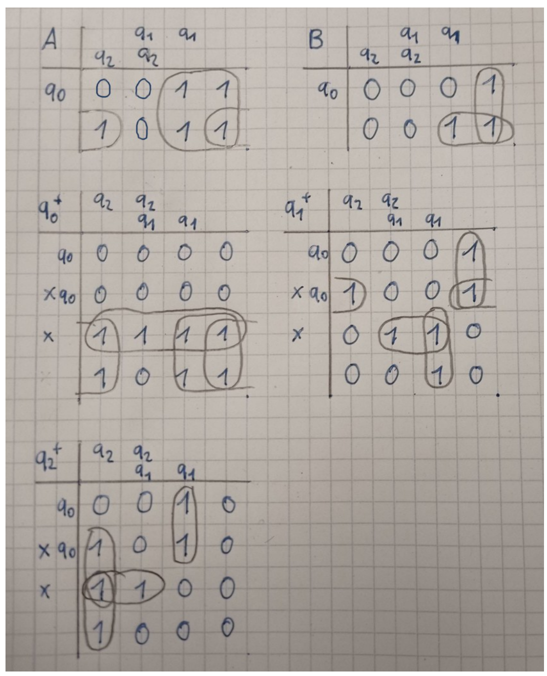
$D = q_0 \vee q_2 \wedge q_1 \wedge \neg q_0$

$C = \neg q_2 \wedge \neg q_1 \wedge \neg q_0$

$B = \neg q_2 \wedge \neg q_1 \;\vee\; \neg q_2 \wedge \neg q_0$

$A = \neg q_2 \;\vee\; \neg q_1 \wedge \neg q_0$

Zur Vereinfachung haben wir teils die nebenstehenden KV-Diagramme genutzt.

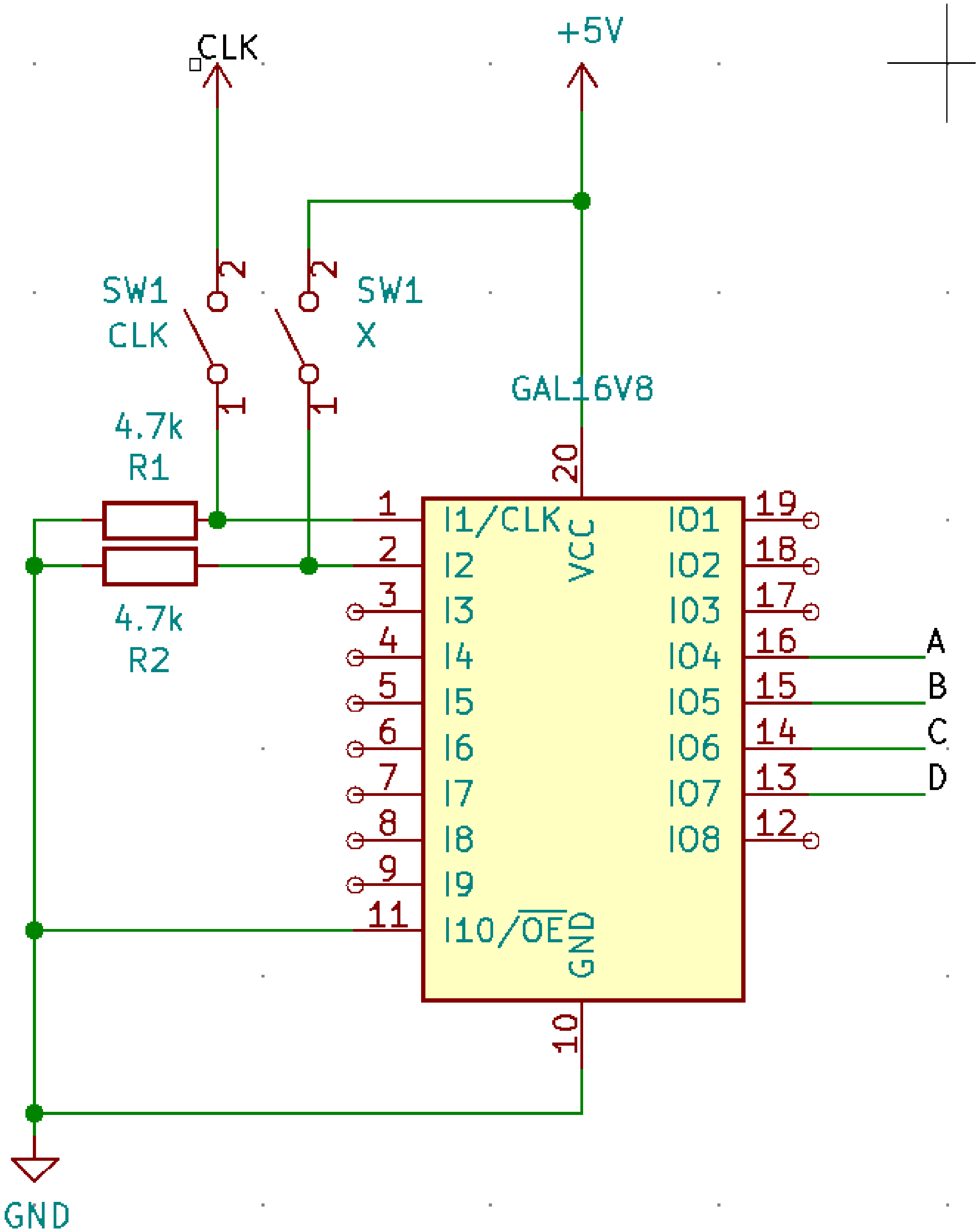


Diese DNFs können dem G16V8 nun direkt übergeben und einprogrammiert werden. Der Code:

```
/* Info */
Name Wuerfel ;
/* -- etc... -- */

/* Input pins */
PIN 1 = clock ;
PIN 2 = x ; /* Der Taster */

/* Output pins */
PIN 19 = q0 ;
PIN 18 = q1 ;
PIN 17 = q2 ;

PIN 16 = A ;
PIN 15 = B ;
PIN 14 = C ;
PIN 13 = D ;

/* Uebergangsfunktionen */
q0.d = (x & !q0) # (!q2 & !q0) # (!q1 & !q0)  ;
q1.d = (!q1 & q0 & x) # (!q2 & !q1 & q0) # (q1 & !q0 & x) #
      (!q2 & q1 & !q0)  ;
q2.d = (!q2 & q1 & q0) # (q2 & !q1 & x) # (q2 & !q1 & !q0) #
      (q2 & !q1 & !q0) # (q2 & !q0 & x) ;

/* Ausgabefunktionen */
D = q0 # (q2 & q1 & !q0) ;
C = !q2 & !q1 & !q0 ;
B = (!q2 & !q1) # (!q2 & !q0) ;
A = !q2 # (!q1 & !q0) ;
```

Der GAL wird dann laut Datenblatt und entsprechend der einprogrammierten Pinverlegung an V_{CC}, Ground und die Bildanzeige angeschlossen.

Um vernünftig mit der Schaltung zu interagieren, bauen wir nun noch zwei Schalter mit Pull-Down Widerstand vor die Eingabepins 1 (CLK) und 2 (x).
Unser Schaltplan:



Und aufgebaut sieht es in etwa so aus: