

## Digitaltechnik Labor 4

# Parkplatzzähler: Endlich endliche Automaten

Vorbereitung, Version 2

Jan Hoegen\*

Erstellt am: 11. Mai 2022

Betreuer: Prof. Dr. -Ing. Jan Bauer

Änderungen gegenüber der ursprünglichen Version wurden durch **rote Schrift** gekennzeichnet.

## Inhaltsverzeichnis

<b>1</b>	<b>Analyse der Parkplatzschaltung</b>	<b>2</b>
<b>2</b>	<b>Analyse des endlichen Automaten</b>	<b>2</b>
<b>3</b>	<b>Auswahl der Zustandscodierung</b>	<b>2</b>
<b>4</b>	<b>Aufbau der Zustandsübergangstabelle</b>	<b>2</b>
<b>5</b>	<b>Auswahl des Standardbausteins</b>	<b>4</b>
<b>6</b>	<b>Realisierung des Übergangsschaltnetzes</b>	<b>4</b>
<b>7</b>	<b>Realisierung des Ausgangsschaltnetzes</b>	<b>4</b>
<b>8</b>	<b>Vereinfachen der Full-NAND-Lösung</b>	<b>5</b>
<b>9</b>	<b>Simulation der Schaltung</b>	<b>5</b>

---

\*Matrikelnummer: 82358. E-Mail: [jan.hoegen@web.de](mailto:jan.hoegen@web.de)

## 1 Analyse der Parkplatzschaltung

Die Widerstände an den Tastern Count down und Count up sorgen dafür, dass zu jeder Zeit eine Spannung kleiner als  $V_{CC}$  an den Eingängen UP und DOWN des 74HC192 anliegt. Wird der Taster betätigt, so wird der entsprechende Eingang zwecklos geschaltet und die Spannung  $V_{CC}$  fällt vollständig am Widerstand ab. In diesem Fall liegt keine Spannung mehr am Eingang an.

Die beiden 74HC4511 sind ein BCD-zu-7-Segment-Codewandler. Abhängig der Eingangsvariablen A bis D werden die Ausgänge a bis g auf HIGH bzw. LOW gesetzt, sodass die 7 Segmente der Anzeige eine Dezimalzahl 0 bis 9 anzeigen. Dabei beschreibt das Datenblatt eine feste Codiertabelle, in der jedem Eingangszustand eine Ziffer zugeordnet wird.

Der 74HC192 addiert jedes Mal, wenn der UP-Eingang eine positive Taktflanke aufweist, zu seinem Aktuellen Zustand die Binärzahl 0001 dazu und speichert das Ergebnis als seinen neuen Zustand. Ähnlich verhält es sich mit dem DOWN-Eingang, jedoch wird hier mit der Binärzahl 0001 subtrahiert. Dabei können die Ausgangsvariablen  $Q_0$  bis  $Q_3$  Binärzahlen von 0000 bis 1001 anzeigen. Dies entspricht den Dezimalzahlen 0 bis 9.<sup>1</sup> Führt das Ergebnis der beiden Summanden zu einer Zahl, die größer als dieser Zahlenbereich ist, so wird der Carry-Ausgang auf HIGH gesetzt und **der Zählerzustand wird auf 0000 gesetzt**. Erzeugt die Rechnung jedoch einen Underflow, **also wird von 0000 subtrahiert**, so wird borrow auf HIGH gesetzt **und der Zustand wird auf 1001 – im Dezimalsystem eine 9 – gesetzt**. Auf die übrigen Pins des IC wird an dieser Stelle nicht näher eingegangen.

Der erste 74HC192 führt diese Funktion genau dann aus, wenn der Taster Count up bzw. Count down nach vorherigem Drücken wieder losgelassen wird. Der zweite 74HC192 erhält sein Eingangssignal UP, wenn das Ergebnis des Ersten größer als 1001 ist, bzw für den Eingang DOWN bei einem Underflow.

## 2 Analyse des endlichen Automaten

Das Zustandsdiagramm zeigt einen Moore-Automaten. Denn die Ausgangsvariablen sind lediglich vom Speicherzustand abhängig, nicht auf direkte Weise mit den Eingangsvariablen.

Es werden zwei D-Flip-Flops benötigt, um die Zustände zu realisieren.

## 3 Auswahl der Zustandscodierung

Da das Ergebnis der Aufgabe 0 der dritten Laborvorbereitung  $P_T = 0$  lautete, ist die Zustandskodierung aus Tabelle 1 zu verwenden.

## 4 Aufbau der Zustandsübergangstabelle

Nun wird die Zustandsübergangstabelle in Tabelle 2 vervollständigt.

---

<sup>1</sup>Der BCD-Codewandler ist ebenfalls für die Zahlen 0 bis 9 definiert, also den Binärzahlen 0000 bis 1001.

Tabelle 1: Zustandskodierung der FSM

Zustand	$Q_1$	$Q_0$	$Dn$	$Up$
idle	0	0	1	1
forw	0	1	1	0
back	1	0	0	1
wait	1	1	1	1

Tabelle 2: Zustandsübergangstabelle

		n				n+1	
Nr.	Übergang	$S_2$	$S_1$	$Q_1$	$Q_0$	$Q_1$	$Q_0$
1	idle → idle	0	0	0	0	0	0
2	forw → forw	0	0	0	1	0	1
3	back → back	0	0	1	0	1	0
4	wait → idle	0	0	1	1	0	0
5	idle → forw	0	1	0	0	0	1
6	forw → forw	0	1	0	1	0	1
7	back → X	0	1	1	0	X	X
8	wait → wait	0	1	1	1	1	1
9	idle → back	1	0	0	0	1	0
10	forw → X	1	0	0	1	X	X
11	back → back	1	0	1	0	1	0
12	wait → wait	1	0	1	1	1	1
13	idle → X	1	1	0	0	X	X
14	forw → wait	1	1	0	1	1	1
15	back → wait	1	1	1	0	1	1
16	wait → wait	1	1	1	1	1	1

## 5 Auswahl des Standardbausteins

Da das Ergebnis der Aufgabe 0 der dritten Laborvorbereitung  $P_M = 0$  lautete, ist das ASN und das ÜSN in Full-NAND zu realisieren.

## 6 Realisierung des Übergangsschaltnetzes

Das Übergangsschaltnetz wird mit Hilfe der Tabelle 2 in Full-NAND erstellt.

Tabelle 3: KV-Tafel für  $Q_0^+$

		$S_2S_1$			
$Q_1Q_0$		00	01	11	10
00	0		1	X	0
01		1	1	1	X
11	0		1	1	1
10	0		X	1	0

Tabelle 4: KV-Tafel für  $Q_1^+$

		$S_2S_1$			
$Q_1Q_0$		00	01	11	10
00	0	0	0	X	1
01	0	0	0	1	X
11	0		1	1	1
10		1	X	1	1

Für  $Q_0^+$  erhält man nach Umformen der disjunktiven Minimalform in Full-NAND:

$$Q_0^+ = \overline{S_1} \vee (\overline{Q_0} \wedge \overline{Q_1}) \vee (S_2 \wedge Q_0)$$

$$Q_0^+ = \overline{\overline{S_1} \wedge (\overline{Q_0} \wedge \overline{Q_1}) \wedge (S_2 \wedge Q_0)}$$

Gleiches Vorgehen für  $Q_1^+$ :

$$Q_1^+ = \overline{S_2} \vee (\overline{Q_0} \wedge Q_1) \vee (S_1 \wedge Q_1)$$

$$Q_1^+ = \overline{\overline{S_2} \wedge (\overline{Q_0} \wedge Q_1) \wedge (S_1 \wedge Q_1)}$$

## 7 Realisierung des Ausgangsschaltnetzes

Als Nächstes wird das Ausgangsschaltnetz in Full-NAND dargestellt. Dazu wird die Wahrheitstabelle in 1 genutzt.

Tabelle 5: KV-Tafel für  $Up$

		$Q_1$	
$Q_0$		0	1
0		1	1
1	0		1

Tabelle 6: KV-Tafel für  $Dn$

		$Q_1$	
$Q_0$		0	1
0		1	0
1		1	1

Schließlich werden auch die DMF von UP und DN durch Full-NAND realisiert.  
DMF des Ausgangs Up:

$$Up = \overline{Q_0} \vee Q_1$$

$$Up = \overline{Q_0 \wedge \overline{Q_1}}$$

Dn-Ausgang als disjunktive Minimalform in Full-NAND:

$$Dn = Q_0 \vee \overline{Q_1}$$

$$Dn = \overline{\overline{Q_0} \wedge Q_1}$$

## 8 Vereinfachen der Full-NAND-Lösung

Die Terme  $\overline{\overline{Q_0} \wedge Q_1}$  und  $\overline{Q_0 \wedge \overline{Q_1}}$  des ASN kommen bereits im Übergangsschaltnetz vor. Daher können hier zwei Gatter eingespart werden. Es sind somit insgesamt 12 NAND-Gatter notwendig.

## 9 Simulation der Schaltung

Die aufgebaute Schaltung ist Abbildung 1 zu entnehmen. Die Auswertung der Schaltung erzeugt die Wahrheitstabelle 7. Sie stimmt mit Tabelle 2 und dem Zustandsdiagramm aus der Laboranleitung überein. Es wurde also ein korrektes Schaltungsdesign verwendet. Außerdem wurde die Anforderung **von** maximal 12 NAND-Gatter erfüllt. **Taster und Reset arbeiten wie erwartet.**

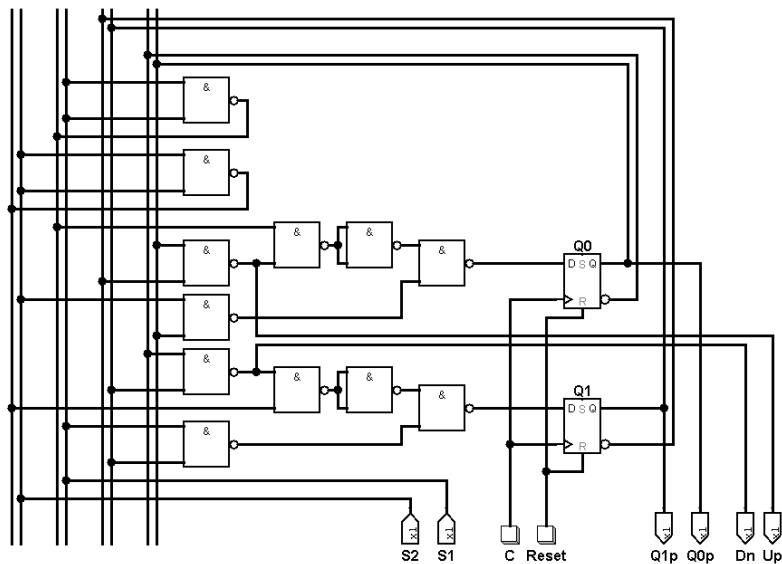


Abbildung 1: Simulation der Schaltung

Tabelle 7: Wahrheitstabelle der simulierten Schaltung

State	$S_2$	$S_1$	$Q_1$	$Q_0$	$Q_0^+$	$Q_0^+$	$Dn$	$Up$
idle $\rightarrow$ idle	0	0	0	0	0	0	1	1
idle $\rightarrow$ forw	0	1	0	0	0	1	1	0
idle $\rightarrow$ back	1	0	0	0	1	0	0	1
idle $\rightarrow$ X	1	1	0	0	1	1	1	1
back $\rightarrow$ back	0	0	1	0	1	0	0	1
back $\rightarrow$ X	0	1	1	0	1	1	1	1
back $\rightarrow$ back	1	0	1	0	1	0	0	1
back $\rightarrow$ wait	1	1	1	0	1	1	1	1
forw $\rightarrow$ forw	0	0	0	1	0	1	1	0
forw $\rightarrow$ forw	0	1	0	1	0	1	1	0
forw $\rightarrow$ X	1	0	0	1	1	1	1	1
forw $\rightarrow$ wait	1	1	0	1	1	1	1	1
wait $\rightarrow$ idle	0	0	1	1	0	0	1	1
wait $\rightarrow$ wait	0	1	1	1	1	1	1	1
wait $\rightarrow$ wait	1	0	1	1	1	1	1	1
wait $\rightarrow$ wait	1	1	1	1	1	1	1	1