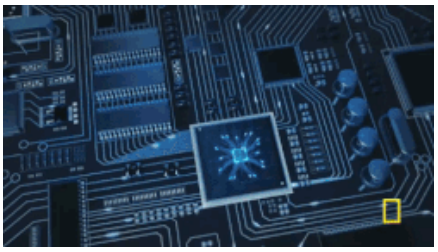
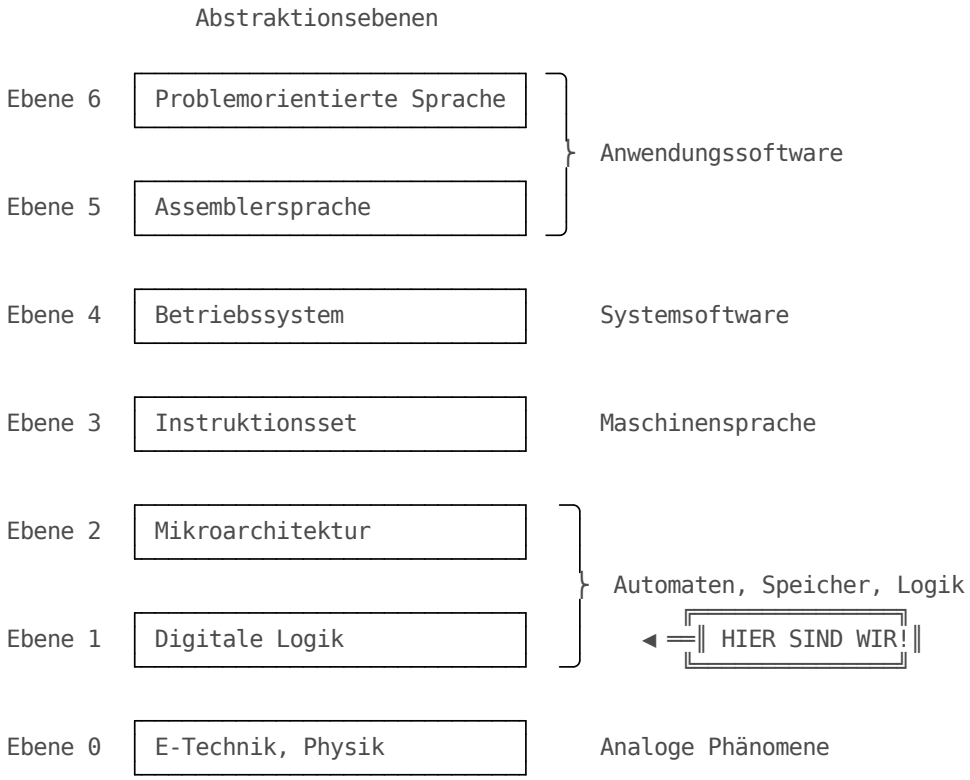
Standardschaltnetze

Parameter	Kursinformationen
Veranstaltung:	`Digitale Systeme / Eingebettete Systeme`
Semester:	`Wintersemester 2025/26`
Hochschule:	`Technische Universität Freiberg`
Inhalte:	`Multiplexer, Demultiplexer, Encoder, Decoder und programmierbare Logikbausteine`
Link auf GitHub:	https://github.com/TUBAF-IfI-LiaScript/VL_EingebetteteSysteme/blob/master/05_Standardschaltnetze.md
Autoren:	Sebastian Zug & André Dietrich & Fabian Bär



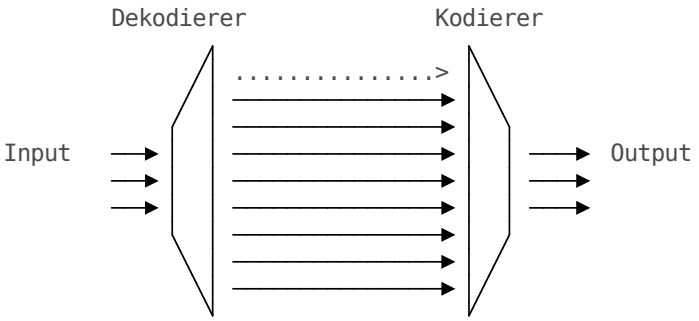
Fragen an die Veranstaltung

- Unterscheiden Sie Multiplexer und Demultiplexer.
- Wie lassen sich mit einem Multiplexer beliebige Wahrheitstafeln abbilden? Welche Grenzen hat dieser Ansatz?
- Welche Aufgaben realisieren Dekoder?
- Nennen Sie Anwendungsbeispiele für einen Multiplexer.
- Was ist der Unterschied zwischen einem Kodierer und einem Prioritätskodierer?
- Wie funktioniert die Adressierung bei einem Speicherbaustein mittels Dekoder?
- Welche Rolle spielen Tri-State-Buffer in Bus-Systemen?
- Erklären Sie die Funktionsweise eines 7-Segment-Displays mit vorgeschaltetem Dekoder.



Dekodierer / Kodierer

In der digitalen Elektronik ist ein Binärdecoder eine kombinatorische Logikschaltung, die binäre Informationen von den n codierten Eingängen in maximal $k = 2^n$ eindeutige Ausgänge umwandelt. Sie werden zum Beispiel für die Ansteuerung von Siebensegmentanzeigen und als Adressdecoder für Speicher und Port-mapped I/O genutzt.



So kann z.B. ein abgewandeltes NOT Gatter als 1:2-Binärdecoder mit 1-Eingang und 2-Ausgänge klassifiziert werden, da er mit einem Eingang A zwei Ausgänge A und $\overline{A}$ hat.

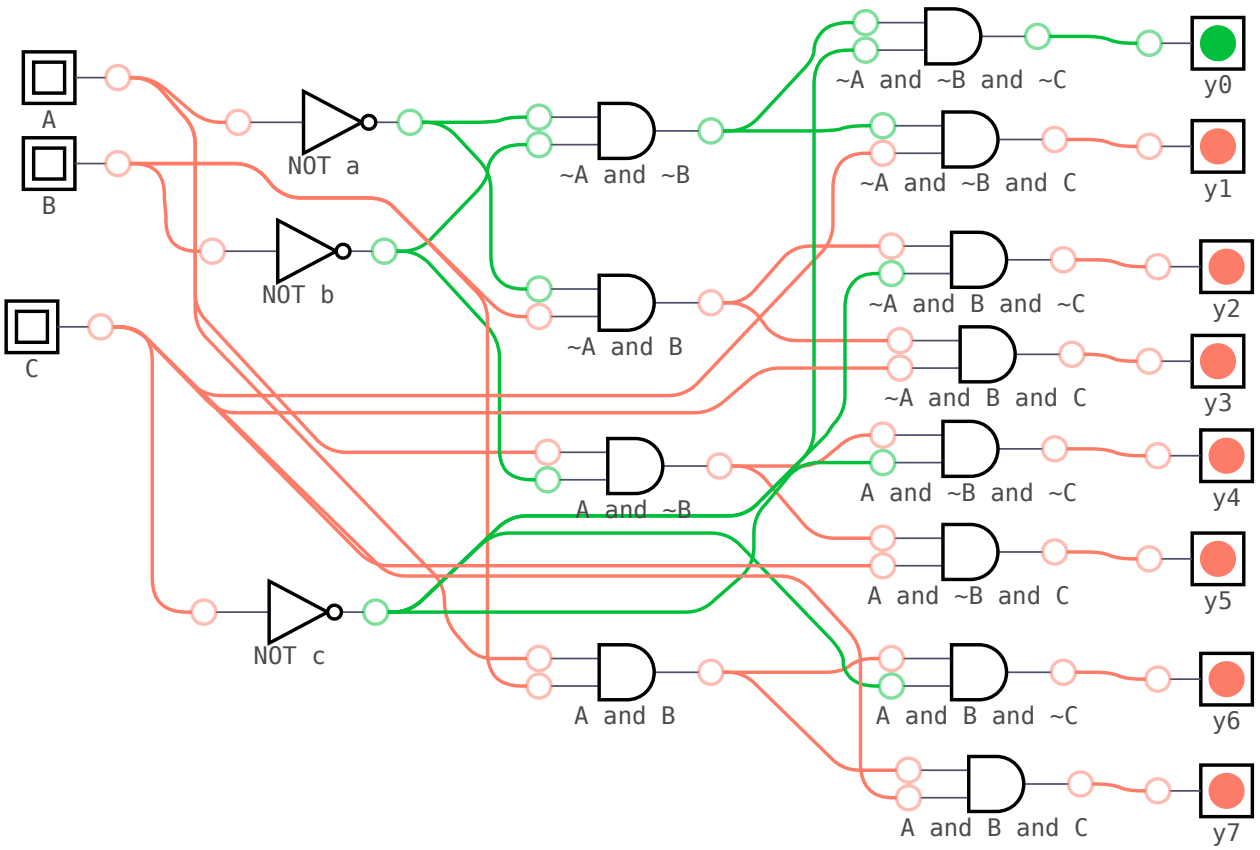
n-zu-k Dekodierer

3-8 Dekodierer

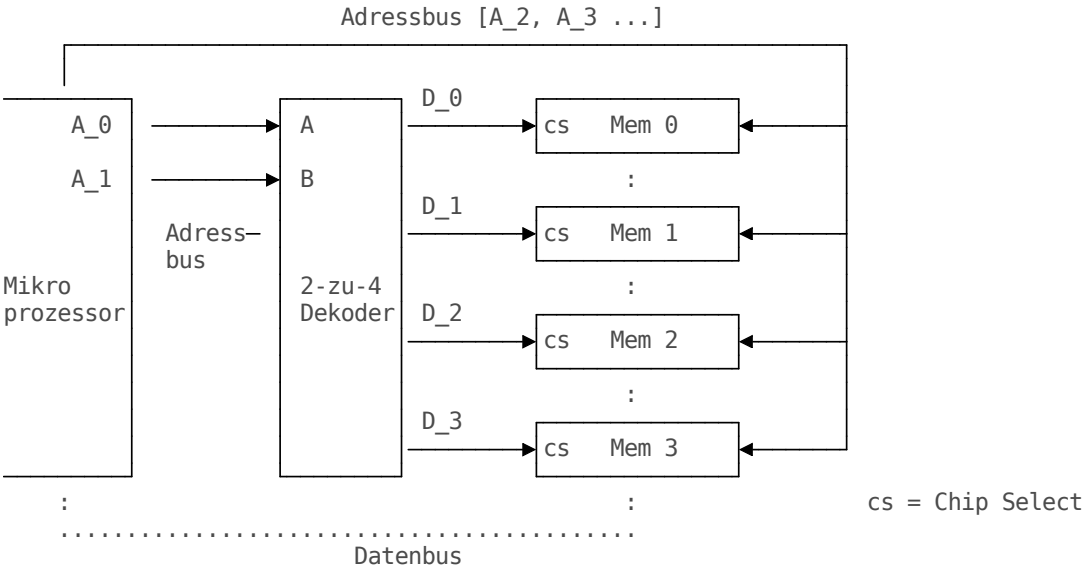
A	B	C	y_0	y_1	y_2	y_3	y_4	...
0	0	0	1	0	0	0	0	...
0	0	1	0	1	0	0	0	...
0	1	0	0	0	1	0	0	...
0	1	1	0	0	0	1	0	...
1	0	0	0	0	0	0	1	...
1	0	1	0	0	0	0	0	...
1	1	0	0	0	0	0	0	...
1	1	1	0	0	0	0	0	...

Für jede Eingangskombination wird genau 1 Ausgang aktiviert.

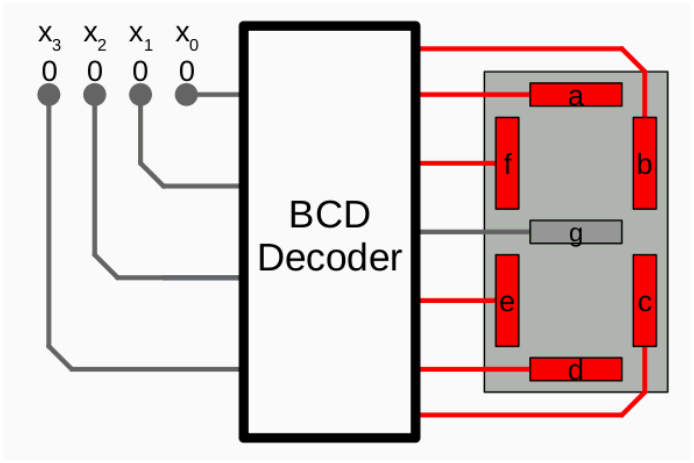
$$
\begin{aligned}
y_0 &= \overline{A} \cdot \overline{B} \cdot \overline{C} \\
y_1 &= \overline{A} \cdot \overline{B} \cdot C \\
y_2 &= \overline{A} \cdot B \cdot \overline{C} \\
y_3 &= \overline{A} \cdot B \cdot C
\end{aligned}
$$

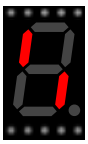


Adressdekoder



BCD Dekoder für 7 Segmentanzeige





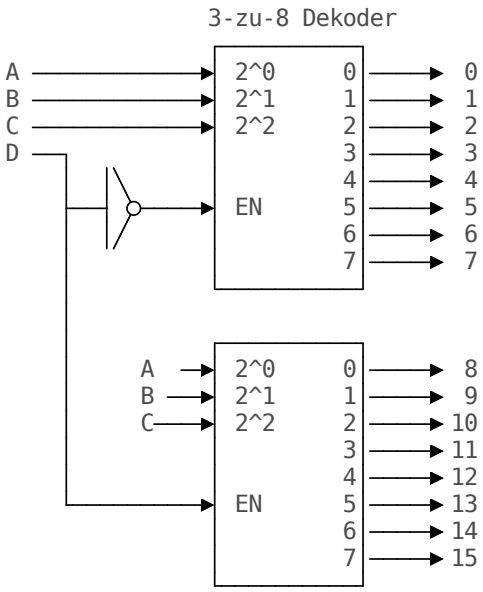
sevenSegmentDisplay.ino

```
const int PINS[] = {0, 1, 2, 3, 4, 5, 6};
const int PAUSE = 1000;
byte segDigits[10][8] = {
  { 1,1,1,1,1,1,1,0 },  // = 0
  { 0,1,1,0,0,0,0,0 },  // = 1
  { 1,1,0,1,1,1,0,1 },  // = 2
  { 1,1,1,1,1,0,0,1 },  // = 3
  { 0,1,1,0,0,0,1,1 },  // = 4rtragen Sie die Funktion in ein Karnaugh
    -Veitch-Di
};
void setup() {
  for(int i=0;i<8;i++){
    pinMode(PINS[i], OUTPUT);
  }
}

void loop() {
  for (int count = 0; count <=9; ++count) {
    for(int i=0; i<10; i++) {
      digitalWrite(PINS[i], segDigits[count][i]);
    }
    delay(PAUSE);
  }
}
```

```
Sketch uses 1086 bytes (3%) of program storage space. Maximum is 32256 bytes.
Global variables use 103 bytes (5%) of dynamic memory, leaving 1945 bytes for local
variables. Maximum is 2048 bytes.


Sketch uses 1086 bytes (3%) of program storage space. Maximum is 32256 bytes.
Global variables use 103 bytes (5%) of dynamic memory, leaving 1945 bytes for local
variables. Maximum is 2048 bytes.
```

Verschaltung von Dekodern

Realisierung eines 4-zu-16 Dekoders auf der Basis von zwei 3-zu-8 Dekodern



.

n-zu-k Kodierer

- n Ausgänge y_0, y_1, ... ,y_{n-1}
- $k = 2^n$ Eingänge x_0, x_1, ... ,x_{k-1}
- nur genau eine Eingangsleitung darf auf 1 sein: $x_i = 1, xj <> i = 0$

Jeder Eingangsleitung ist genau eine Kombination der möglichen Belegungen der Ausgangsleitungen zugeordnet, z.B. ihre binäre Repräsentation.

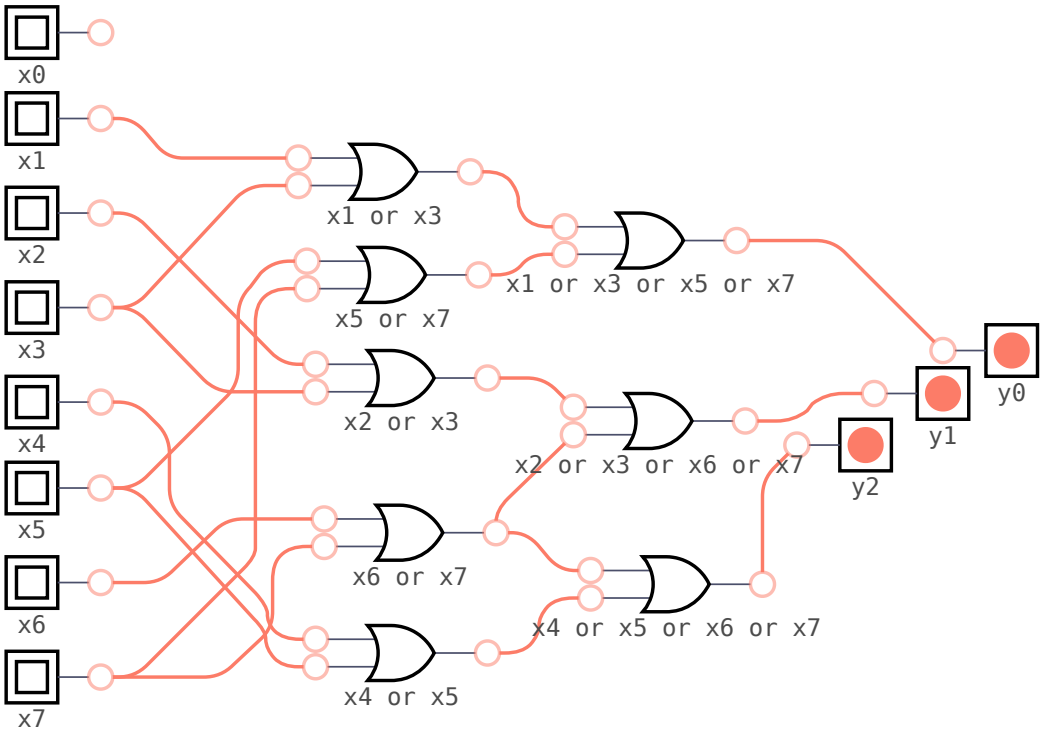
8-3 Kodierer

x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	y_2	y_1	y_0
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

$$y_0 = x_1 + x_3 + x_5 + x_7$$
$$y_1 = x_2 + x_3 + x_6 + x_7$$
$$y_2 = x_4 + x_5 + x_6 + x_7$$

Achtung: Die Wahrheitstafel ist unvollständig !

1. Falsche Ausgangszustände sind möglich!
2. Was passiert wenn alle Pegel 0 sind?



Prioritätsencoder

Abhilfe schafft der Prioritätsencoder. Hier wird eine explizite Auswahl für verschiedene Eingangskonfigurationen getroffen.

x_3	x_2	x_1	x_0	y_1	y_0
0	0	0	0	0	0
0	0	0	1	0	0
0	0	1	X	0	1
0	1	X	X	1	0
1	X	X	X	1	1

$$y_0 = \overline{x}_3\overline{x}_2x_1 + x_3$$
$$y_1 = \overline{x}_3x_2$$

Und im echten Leben? Kommen noch einige Spezialeingänge / -ausgänge dazu Link.

Input									Output				
E_{in}	D7	D6	D5	D4	D3	D2	D1	D0	GS	Q2	Q1	Q0	E_{out}
0	X	X	X	X	X	X	X	X	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	1
1	1	X	X	X	X	X	X	X	1	1	1	1	0
1	0	1	X	X	X	X	X	X	1	1	1	0	0
1	0	0	1	X	X	X	X	X	1	1	0	1	0
1	0	0	0	1	X	X	X	X	1	1	0	0	0
1	0	0	0	0	1	X	X	X	1	0	1	1	0
1	0	0	0	0	0	1	X	X	1	0	1	0	0
1	0	0	0	0	0	0	1	X	1	0	0	1	0
1	0	0	0	0	0	0	0	1	1	0	0	0	0

X = Don't Care

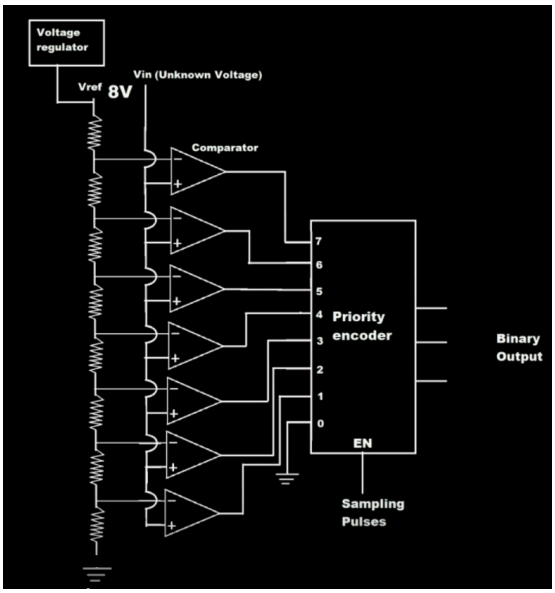
MC14532B - 8-Bit Priority Encoder [1]

> Aufgabe: Vereinfachen Sie die Funktionen für Q_2!

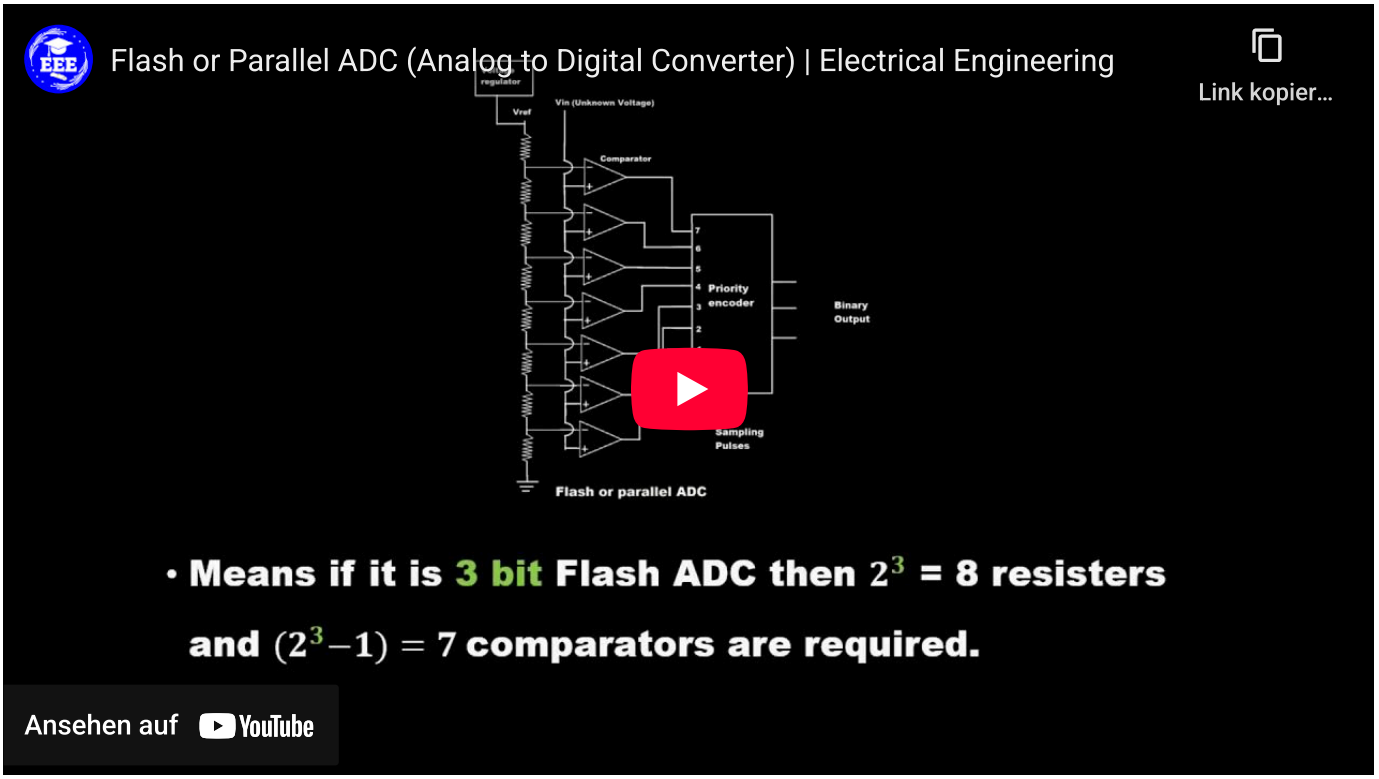
$$Q_2 = D_4 + D_5 + D_6 + D_7$$

Analog Digitalwandler

Wir werden dem Encoder bei der Diskussion der Peripherie eines Mikrocontrollers sehr häufig wiederbegegnen. Das Video zeigt einen Anwendungsfall - das Mapping der Ergebnisse eines Analog-Digital-Wandlers auf eine binäre Ausgabe.



[2]

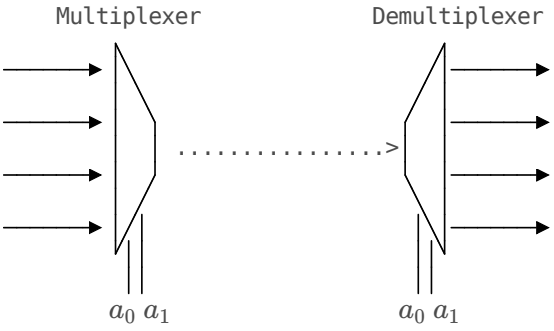


[1] Datenblatt MC14532B - 8-Bit Priority Encoder, Firma ON Semiconductor

[2] Autor: Always Be Positive, Youtube Video - Flash or Parallel ADC (Analog to Digital Converter)

Multiplexer / Demultiplexer

Eine Mulitplexerschaltung bildet analoge oder digitale Eingangssignale auf einen Kommunikationskanal ab, der Demultiplexer übernimmt die Abbildung auf n Ausgangsleitungen.



Multiplexer	Dekoder
mehrere Eingänge, ein Ausgang	mehrere Eingänge, mehrere Ausgänge
Steuerleitungen konfigurierereren die Weiterleitung	das Mapping wird durch die interne Logik bestimmt
bildet den Pegel einer Eingangsleitung auf die Ausgangsleitung ab	wandelt den binären Code in einen unären Code um == aktiviert eine Ausgangsleitung

Multiplexer

Generelle Konfiguration eines 1-aus-k Multiplexer:

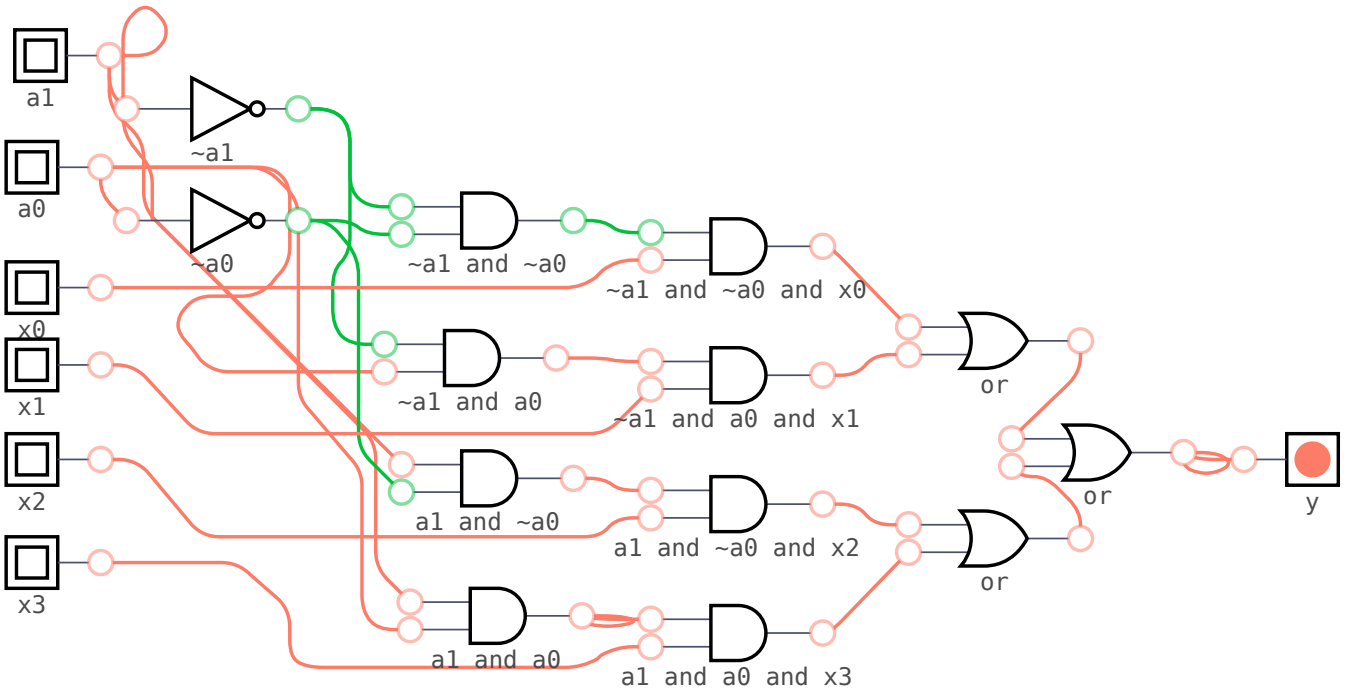
- n Steuerleitungen: $s_{n-1}, ..., s_1, s_0$
- $k = 2^n$ Eingänge: $x_0, x_1, ..., x_{k-1}$
- ein Ausgang: y
- $y = x_i$ für $(s_{n-1}, ..., s_1, s_0)_2 = i$

Beispiel: 1-zu-4 Multiplexer

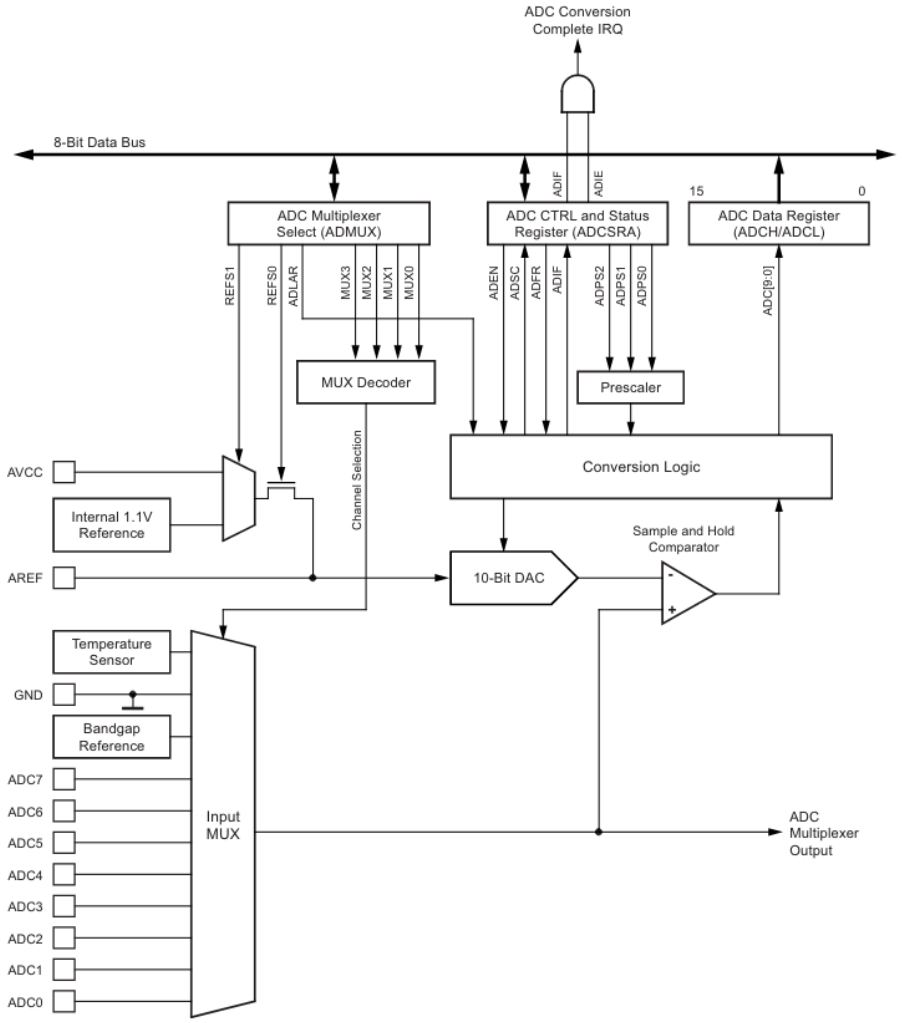
Eine Wahrheitstabelle mit 4 Eingangsvariablen und 2 Steuersignalen würde eine entsprechende Größe aufweisen. Allerdings kann die Funktion auch sehr einfach hergeleitet werden. Ein Inputsignal wird nur dann durchgeleitet, wenn die zugehörige Kombination von Steuersignalen anliegt.

a_1	a_2	**Signal**
0	0	x_0
0	1	x_1
1	0	x_2
1	1	x_3

$y = \overline{a_1} \cdot \overline{a_2} \cdot x_0 + \overline{a_1} \cdot a_2 \cdot x_1 + a_1 \cdot \overline{a_2} \cdot x_2 + a_1 \cdot a_2 \cdot x_3$



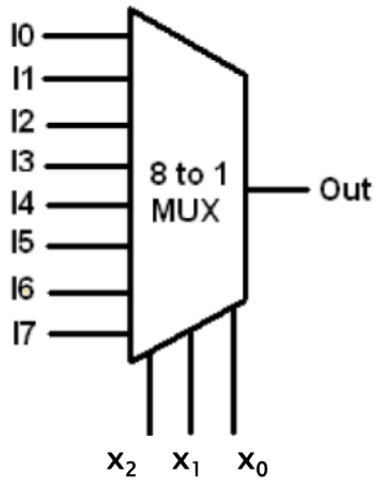
Anwendung in Microcontrollern



Analog Digitalwandler [3]

Multiplexer als universelle boolesche Funktionsrepräsentation

x_2	x_1	x_0	y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



[3] Firma Microchip - ATmega328P8-bit AVR Microcontroller Datasheet Link

Demultiplexer

Generelle Konfiguration eines 1-aus-k Multiplexer:

- n Steuerleitungen: $s_{n-1}, ..., s_1, s_0$
- $k = 2^n$ Ausgänge: $y_0, y_1, ..., y_{k-1}$
- ein Eingang: x
- $y = x_i$ für $(s_{n-1}, ..., s_1, s_0)_2 = i$

Beispiel: 2 Bit Adresse → 4 Ausgänge

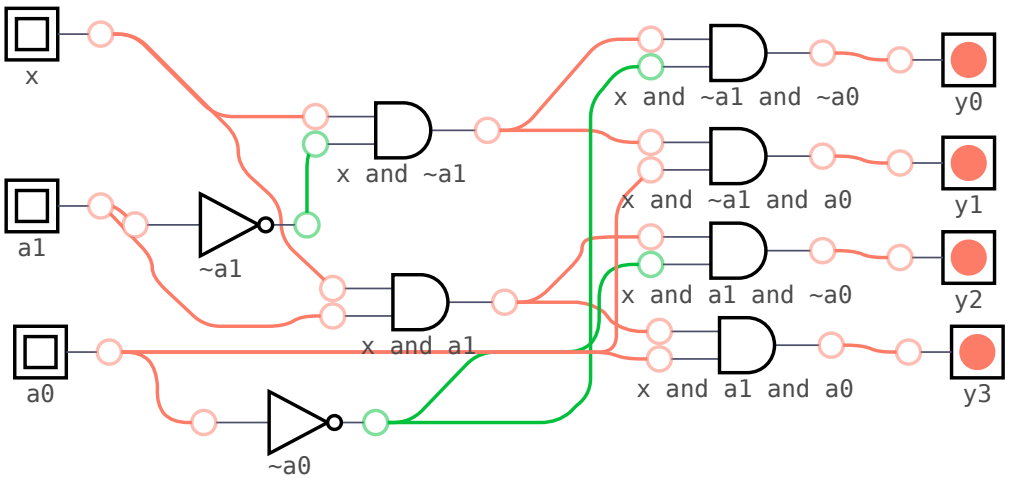
a_0	a_1	x	y_0	y_1	y_2
0	0	0			
0	0	1	1		
0	1	0			
0	1	1		1	
1	0	0			
1	0	1			1
1	1	0			
1	1	1			

$y_0 = x \cdot \overline{a_1} \cdot \overline{a_0}$

$y_1 = x \cdot \overline{a_1} \cdot a_0$

$y_2 = x \cdot a_1 \cdot \overline{a_0}$

$y_3 = x \cdot a_1 \cdot a_0$



Komperatoren

... siehe Hausaufgabe

Ausblick

Nunmehr können boolesche Funktionen als Schaltnetze abbilden? Was fehlt für den Rechner? Zwei Dinge ... Speicher und arithmetische Operationen.

Hausaufgaben

1. Entwerfen Sie einen Komperator, der zwei zweistellige Zahlen vergleicht. Definieren Sie dazu zunächst einen Ein-Bit Komperator und nutzen sie diesen als Grundlage für die Zwei-Bit-Variante
2. Entwickeln Sie ein Schaltnetz, dass die Teilbarkeit durch drei von einer 4-stelligen binären Zahl prüft. Stellen Sie dazu eine Wahrheitstafel auf, minimieren Sie den Ausdruck soweit wie möglich und skizzieren Sie die Verdrahtung der Gatter.

Wer möchte sich mit dem NE555 beschäftigen?