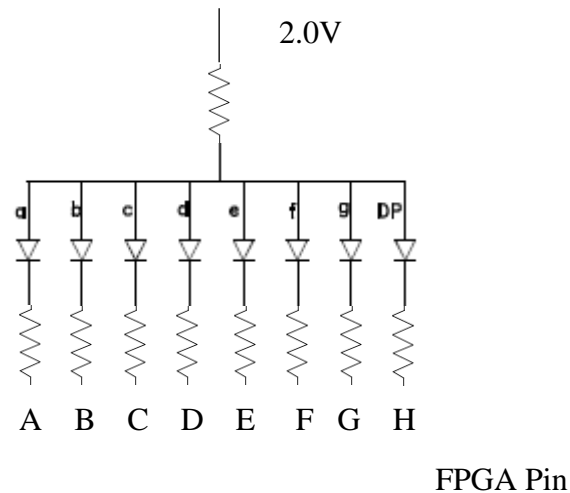


Name: Xuanjiao Zhu Nr.3038674

Sieben Segmentanzeige

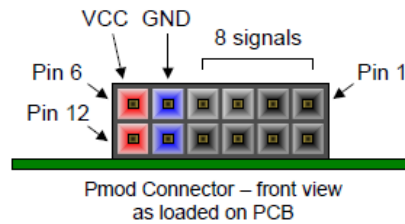
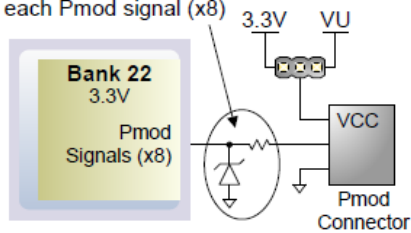
1. Suchen Sie im „7-Seg Manual “ das elektrische Schaltbild und geben Sie dieses an.
Zeichnen Sie zusätzlich den Anschluss für die Versorgungsspannung, die Vorwiderstände und die Anschlüsse zum FPGA ein



2. Die LEDs sollen durch logische Zustände geschaltet werden. Geben Sie die Zugehörigkeit von logischem Pegel und LED Zustand an. Beachten Sie die „CommonAnode“ Verschaltung.

Logischem Pegel	LED Zustand
0	ON
1	OFF

ESD/protection circuit on each Pmod signal (x8)



Digilent produces a large collection of accessory boards that can attach to the Pmod and VHDC expansion connectors to add ready-made functions like A/D's, D/A's, motor drivers, sensors, cameras and other functions. See www.digilentinc.com for more information.

Pmod A		Pmod B		Pmod C		Pmod D	
Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
JA1	AD11	JB1	AE9	JC1	AL11	JD1	AN14
JA2	AD9	JB2	AC8	JC2	AJ10	JD2	AN13
JA3	AM13	JB3	AB10	JC3	AK9	JD3	AP12
JA4	AM12	JB4	AC9	JC4	AF9	JD4	AL10
JA7	AD10	JB7	AF8	JC7	AK11	JD7	AP14
JA8	AE8	JB8	AB8	JC8	AC10	JD8	AN12
JA9	AF10	JB9	AA10	JC9	AJ9	JD9	AM11
JA10	AJ11	JB10	AA9	JC10	AA8	JD10	AK8

Pmod port pinouts.

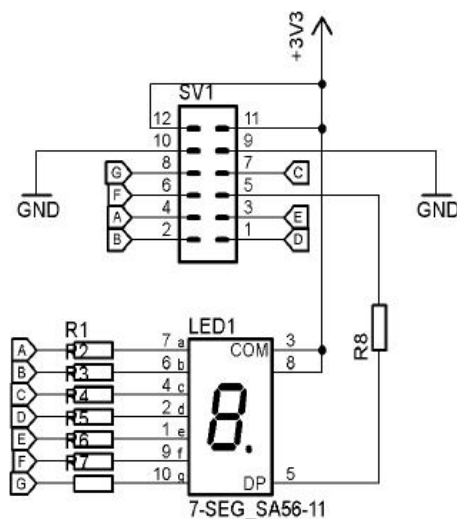


Abbildung 1: Schaltplan der Sieben-Segmentanzeige

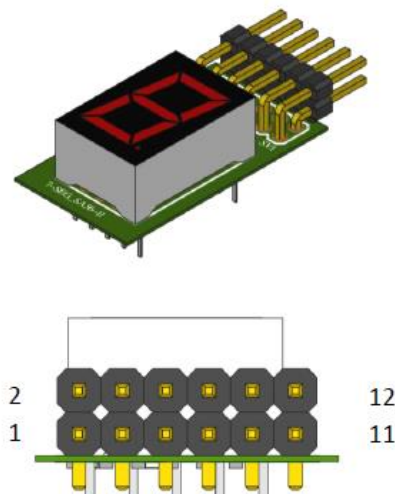


Abbildung 2: Ansichten der Sieben-Segmentanzeige mit Pinbelegung

3. Erstellen Sie anhand von Tabelle 1 das Pin-Mapping für zwei SiebenSegmentanzeigen an PMOD-A und PMOD-B

	LED Char	Pin am SV1	Pin am PMOD	Name am PMOD	FPGA Pin
PMOD-A	A	4	2	JA2	AD9
	B	2	1	JA1	AD11
	C	7	10	JA10	AJ11
	D	1	7	JA7	AD10
	E	3	8	JA8	AE8
	F	6	3	JA3	AM13
	G	8	4	JA4	AM12
PMOD_B	A	4	2	JB2	AC8
	B	2	1	JB1	AE9
	C	7	10	JB10	AA9
	D	1	7	JB7	AF8
	E	3	8	JB8	AB8
	F	6	3	JB3	AB10
	G	8	4	JB4	AC9

4. Zahlen und Buchstaben können durch eine Kombination der LEDs A bis G dargestellt werden. Erstellen Sie eine Tabelle nach dem Schema von Tabelle 2 für alle darstellbaren Zeichen.

	A	B	C	D	E	F	G
AUS							
0	0 (LED ON)	0	0	0	0	0	1
1	1 (LED OFF)	0	0	1	1	1	1
2	0	0	1	0	0	1	0
3	0	0	0	0	1	1	0
4	1	0	0	1	1	0	0
5	0	1	0	0	1	0	0
6	0	1	0	0	0	0	0
7	0	0	0	1	1	1	1
8	0	0	0	0	0	0	0
9	0	0	0	0	1	0	0
A	0	0	0	1	0	0	0
B	1	1	0	0	0	0	0
C	0	1	1	0	0	0	1
D	1	0	0	0	0	1	0
E	0	1	1	0	0	0	0
F	0	1	1	1	0	0	0

5. Geben Sie den Verilog-Code für einen ALIAS „ZERO “ an, welcher das Mapping auf den LED-Bit-Vektor (A bis G) für das Zeichen „0 “ realisiert.

```
parameter 0 0'b00000001
```

6. Der Controller soll Anhand der Eingangs-Schalterstellung dem Ausgang einen zugehörigen Wert liefern. Welche Verilog-Konstrukte stehen Ihnen allgemein zur Verfügung um Entscheidungswege zu realisieren? Welche Struktur eignet sich für den genannten Fall am besten?

- 1) If else and Case structure
- 2) Multiplexer structure

```
.....  
if(!reset_n) begin  
    out0<=LEER_OUT;  
    out1<=LEER_OUT;  
end else begin  
    case(in)  
        ZERO: begin  
            out0<=ZERO_OUT;  
            out1<=LEER_OUT;  
            end  
  
        ONE: begin  
            out0<=ONE_OUT;  
            out1<=LEER_OUT;  
            end  
  
        .....  
  
        default: begin  
            out0<=LEER_OUT;  
            out1<=LEER_OUT;  
            end  
    endcase  
end  
end
```