

FPGA Modul XC6SLX25



FPGA Modul zur Anwendung im Hobby- und Prototypenbereich

Eigenschaften:

- | | |
|---|---|
| <ul style="list-style-type: none">• Xilinx Spartan 6 (XC6SLX25-2FGG484C)• 78 Nutzer-IOs• On-Board 50MHz Oszillator• 2 LEDs, 1 Taster• Rastermaß: 2.54 mm• Maße: 51 x 41 mm | <ul style="list-style-type: none">• SPI Flash zur Speicherung von Konfigurationen und Nutzerdaten• 6-Pin JTAG Header• Spannungsversorgung 3.3V• IO-Spannungen der FPGA-Bänke individuell einstellbar |
|---|---|

Versorgungsspannungen:

Zum Betrieb des Moduls ist eine Spannung von 3.3V notwendig. Diese ist an einem der Versorgungspins über die Pinleiste bereitzustellen (siehe Pinbelegungen). Bei korrektem Betrieb leuchtet LED1.

Die Betriebsspannungen der drei FPGA-Bänke können über die Pinleiste in Abhängigkeit ihrer Anwendung individuell bereitgestellt werden. Folgende Pins sind dafür vorgesehen:

- SV1, pin 2 = Bank 1
- SV2, pin 1 = Bank 3
- SV3, pin 1 = Bank 2

Sämtliche Pins der jeweiligen Leiste sind mit FPGA-IOs der entsprechenden Bänke verbunden. Für eine gewünschte IO-Spannung von 3.3V kann der Versorgungspin der FPGA-Bank im einfachsten Fall mit dem direkt benachbarten Versorgungspin des Moduls verbunden werden. Alle anderen Spannungen müssen separat bereitgestellt werden.

On-Board Schnittstellen:

Das Modul ist bestückt mit 2 Nutzer-LEDs (active high) und einem Taster (active low). Zur synchronen Ablaufsteuerung wurde ein 50MHz Oszillator integriert. Folgende FPGA Pins wurden hierfür verwendet:

LED1	-	B1
LED2	-	C1
Taster	-	A2
50MHz	-	Y12

Flash Speicher:

Der angebundene 32MBit SPI Flash Speicher (Nymonyx M25P32) kann neben der Ablage von Programmdateien auch zur Speicherung von Nutzerdaten verwendet werden. Folgende FPGA-Pins wurden hierfür verwendet:

D (Serial Data Input)	-	AB20 (FPGA-Pin)
Q (Serial Data Output)	-	AA20
C (Serial Clock)	-	Y21
\overline{S} (Chip Select)	-	T5

Die Flash-Signale \overline{W} und \overline{HOLD} sind permanent auf high gelegt.

Konfiguration:

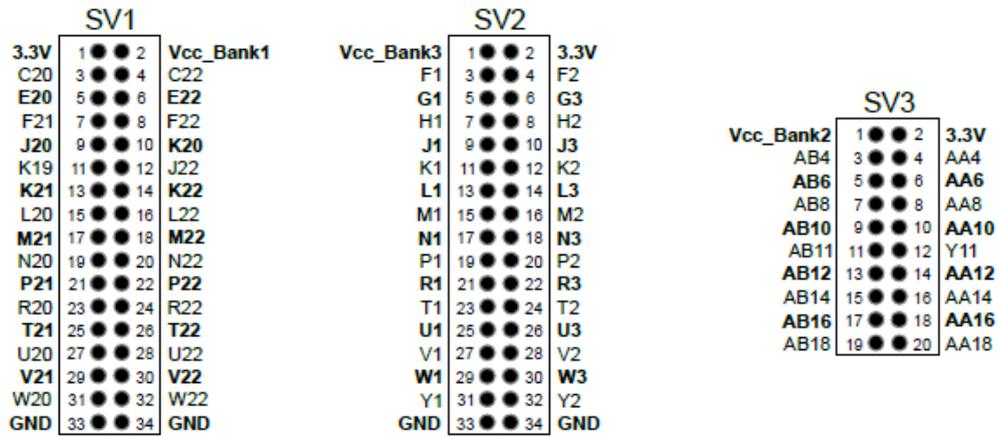
Die Parametrierung des Moduls erfolgt über einen 6-Pin JTAG-Header, mit dem Konfigurationsdaten zu Testzwecken direkt in das FPGA oder zum automatischen Laden nach Systemstart in den angebundene SPI Flash übertragen werden können. Die Konfiguration kann z.B. mittels des Xilinx iMPACT Tools erfolgen.

Eine erfolgreiche Konfiguration wird durch die leuchtende DONE LED angezeigt.

Das Speichern der Konfigurationsdaten im Flash erfolgt durch indirekte SPI Programmierung. Eine detaillierte Anleitung befindet sich im Anhang B.

Pinbelegungen:

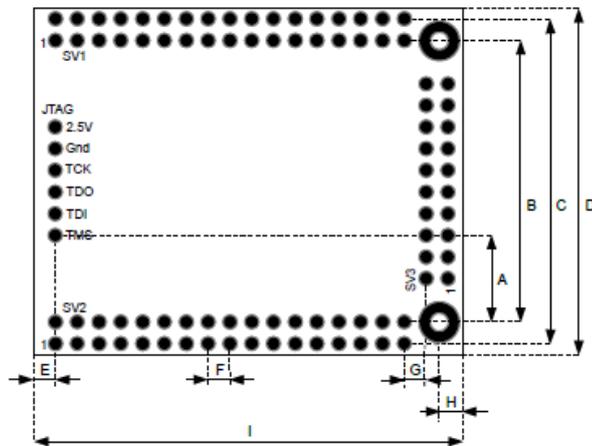
Folgende FPGA-Pins wurden an die Pinleisten geführt:



Die detaillierten Pinbezeichnungen sind im Anhang A aufgeführt:

Abmessungen:

Symbol	Measure (mm)
A	10.16
B	33.02
C	38.10
D	41.15
E	2.16
F	2.54
G	2.54
H	3.82
I	51.18



Anhang A:
PinOut
SV1

Header		FPGA	
Pin #	Pin #	Bank	Pin Description
1	3.3V		
2	Vcc_Bank1		
3	C20	1	IO_L32P_1
4	C22	1	IO_L32N_1
5	E20	1	IO_L35P_1
6	E22	1	IO_L35N_1
7	F21	1	IO_L37P_1
8	F22	1	IO_L37N_1
9	J20	1	IO_L43P_GCLK5_1
10	K20	1	IO_L40P_GCLK11_1
11	K19	1	IO_L40N_GCLK10_1
12	J22	1	IO_L43N_GCLK4_1
13	K21	1	IO_L44P_1
14	K22	1	IO_L44N_1
15	L20	1	IO_L45P_1
16	L22	1	IO_L45N_1
17	M21	1	IO_L46P_1
18	M22	1	IO_L46N_1
19	N20	1	IO_L47P_1
20	N22	1	IO_L47N_1
21	P21	1	IO_L48P_1
22	P22	1	IO_L48N_1
23	R20	1	IO_L49P_1
24	R22	1	IO_L49N_1
25	T21	1	IO_L50P_1
26	T22	1	IO_L50N_1
27	U20	1	IO_L51P_1
28	U22	1	IO_L51N_1
29	V21	1	IO_L52P_1
30	V22	1	IO_L52N_1
31	W20	1	IO_L60P_1
32	W22	1	IO_L60N_1
33	GND		
34	GND		

SV2

Header		FPGA	
Pin #	Pin #	Bank	Pin Description
1	Vcc_Bank3		
2	3.3V		
3	F1	3	IO_L50N_3
4	F2	3	IO_L50P_3
5	G1	3	IO_L48N_3
6	G3	3	IO_L48P_3
7	H1	3	IO_L47N_3
8	H2	3	IO_L47P_3
9	J1	3	IO_L41N_GCLK26_3
10	J3	3	IO_L41P_GCLK27_3
11	K1	3	IO_L40N_3
12	K2	3	IO_L40P_3
13	L1	3	IO_L39N_3
14	L3	3	IO_L39P_3
15	M1	3	IO_L38N_3
16	M2	3	IO_L38P_3
17	N1	3	IO_L37N_3
18	N3	3	IO_L37P_3
19	P1	3	IO_L36N_3
20	P2	3	IO_L36P_3
21	R1	3	IO_L35N_3
22	R3	3	IO_L35P_3
23	T1	3	IO_L34N_3
24	T2	3	IO_L34P_3
25	U1	3	IO_L33N_3
26	U3	3	IO_L33P_3
27	V1	3	IO_L32N_3
28	V2	3	IO_L32P_3
29	W1	3	IO_L2N_3
30	W3	3	IO_L2P_3
31	Y1	3	IO_L1N_VREF_3
32	Y2	3	IO_L1P_3
33	GND		
34	GND		

SV3

Header		FPGA	
Pin #	Pin #	Bank	Pin Description
1	Vcc_Bank2		
2	3.3V		
3	AB4	2	IO_L57N_2
4	AA4	2	IO_L57P_2
5	AB6	2	IO_L49N_2
6	AA6	2	IO_L49P_2
7	AB8	2	IO_L45N_2
8	AA8	2	IO_L45P_2
9	AB10	2	IO_L41N_VREF_2
10	AA10	2	IO_L41P_2
11	AB11	2	IO_L32N_GCLK28_2
12	Y11	2	IO_L32P_GCLK29_2
13	AB12	2	IO_L31N_GCLK30_2
14	AA12	2	IO_L31P_GCLK31_2
15	AB14	2	IO_L16N_VREF_2
16	AA14	2	IO_L16P_2
17	AB16	2	IO_L19N_2
18	AA16	2	IO_L19P_2
19	AB18	2	IO_L14N_2

OnBoard

Signal	Pin #	Bank	Pin Description
50MHz	A18	0	IO_L65N_0

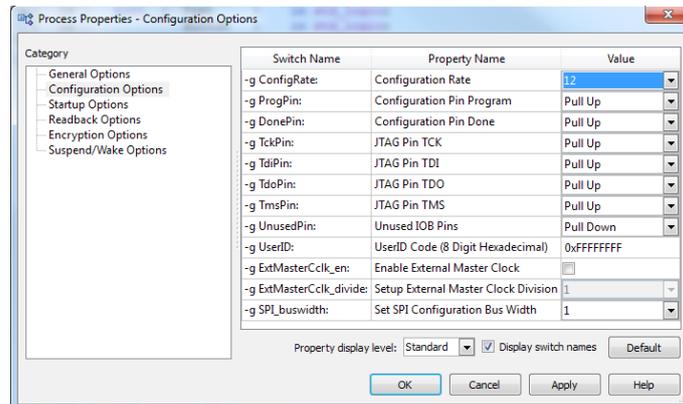
Flash

D	AB20	2	IO_L3N_MOSI_2
Q	AA20	2	IO_L3P_DIN_2
C	Y21	2	IO_L1P_CCLK_2
\bar{S}	T5	2	IO_L65N_CSO_B_2

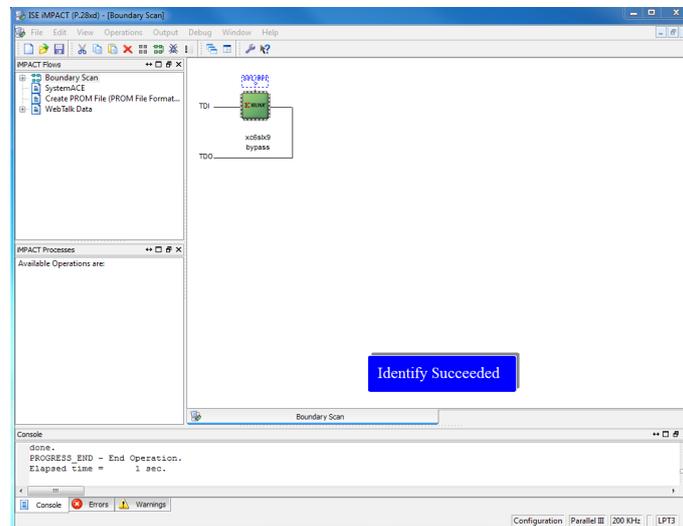
Anhang B:

SPI Flash Konfiguration (Xilinx iMPACT)

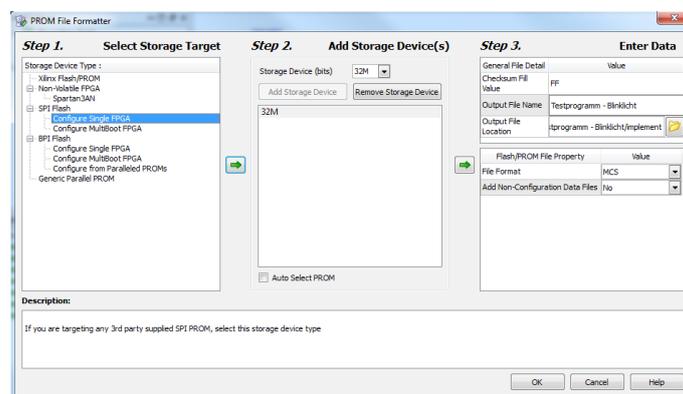
- legen sie bei Nutzung des Xilinx ISE Project Navigator vor Erzeugung der bit-Datei die Konfigurationsrate für den SPI Flash PROM fest: Generate Programming File -> Process Properties -> Configuration Options -> Configuration Rate -> 12



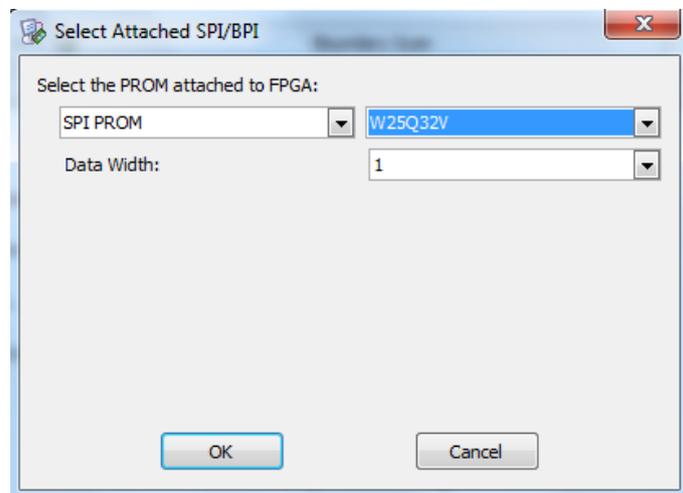
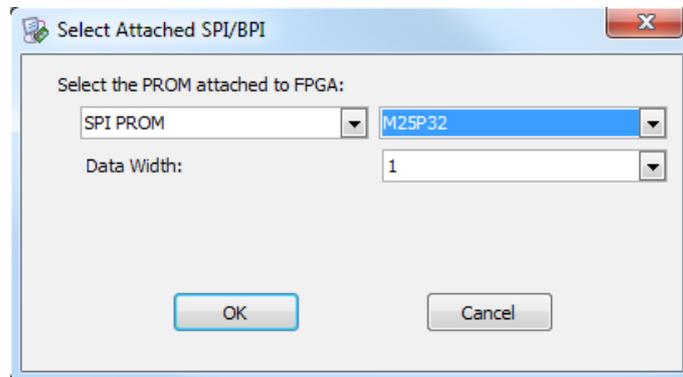
- verbinden sie das Modul über ein JTAG Kabel mit dem Rechner und stellen sie die Spannungsversorgung her
- starten sie Xilinx iMPACT und wählen sie den Boundary Scan Modus
- das Xilinx Spartan 6 FPGA sollte erkannt werden



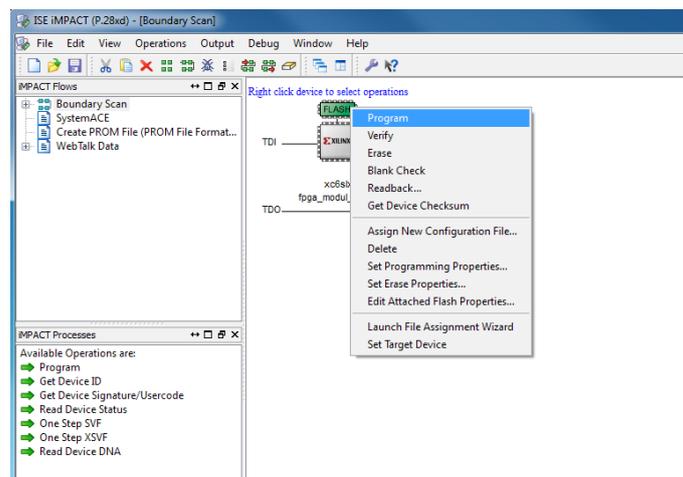
- erzeugen sie aus der bit-Datei eine mcs-Konfigurationsdatei für den Flash Speicher; wählen sie im Auswahldialog die Option „Configure Single FPGA“ und als Speichergröße 32MBit



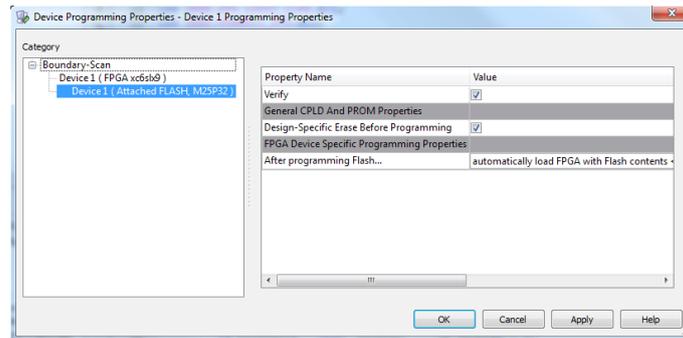
- wählen sie nach einem Rechtsklick auf das FPGA-Icon die Option „Add SPI/BPI Flash ...“ und selektieren sie im folgenden Dialog die zu speichernde mcs-Datei
- wählen sie den SPI Flash aus (SPI PROM - M25P32) oder (W25Q32V).



- nach einem Rechtsklick auf das Flash-Icon können sie über die Option „Program“ die Übertragung vornehmen



- wählen Sie für die Übertragung die Einstellungen „Verify“ und „Erase before Programming“



- nach einer korrekten Übertragung wird das Programm jedes Mal automatisch in das FPGA geladen, sobald die Spannungsversorgung hergestellt wird