DTP2_PROJ_2

School of Engineering

Audio Codec Settings

WM8731 auf dem DE2-115 Board

Clocking Concept, Audio Data Rates

Register settings to select application scenarios

Codec Control Interface

I2C Master Block

Control-IF Driver

WM8731 Audio Codec

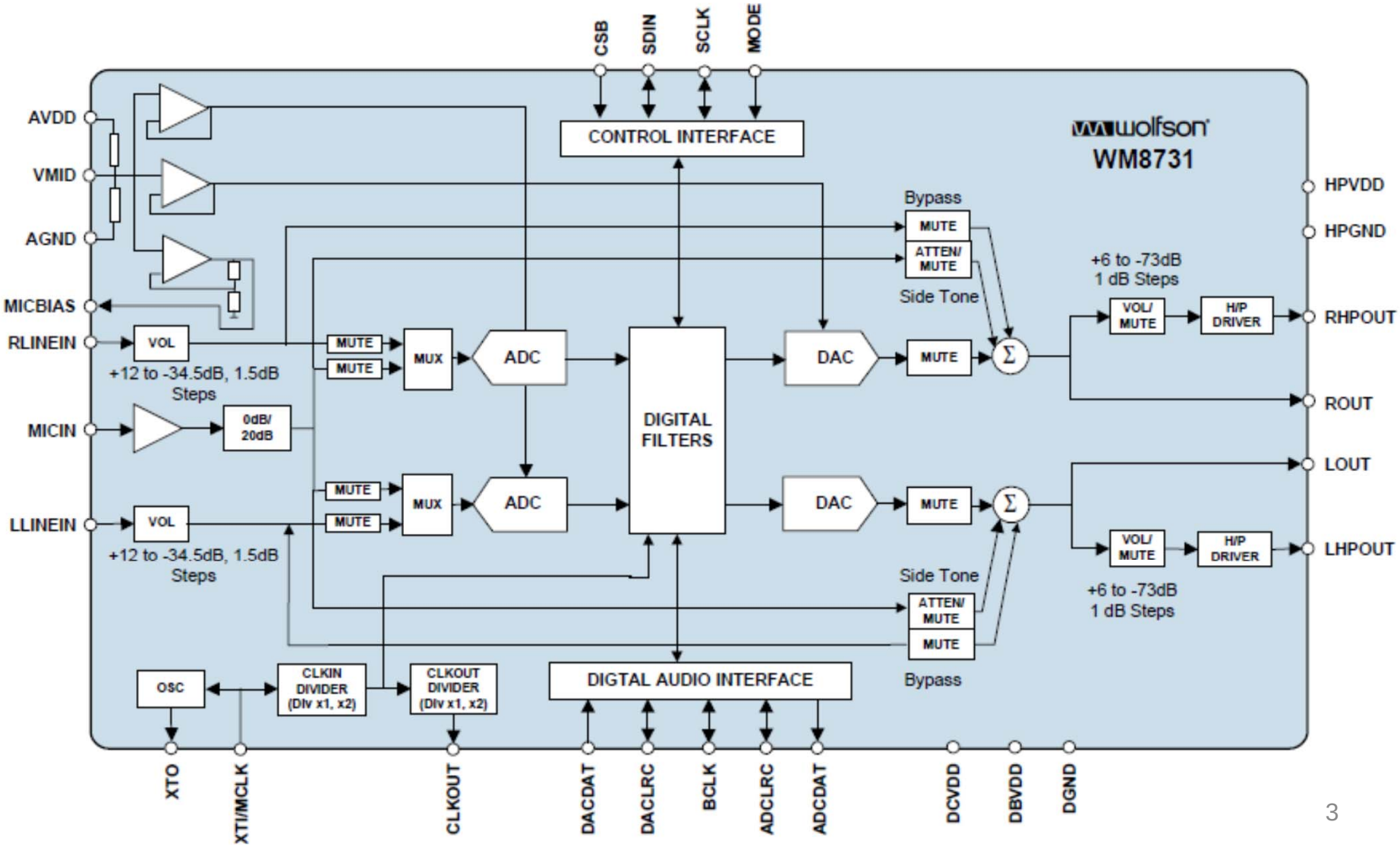
School of Engineering

Sampling Frequency:	8kHz – 96kHz
ADC SNR:	90dB (‘A’ weighted)
DAC SNR:	100dB (‘A’ weighted)
Headphone Amplifier:	2 x 50 mW
Preis:	SFr. 1.61 (1000 Stück)
Datenblatt:	OLAT SW3

Audio Codec WM8731

Block Diagram

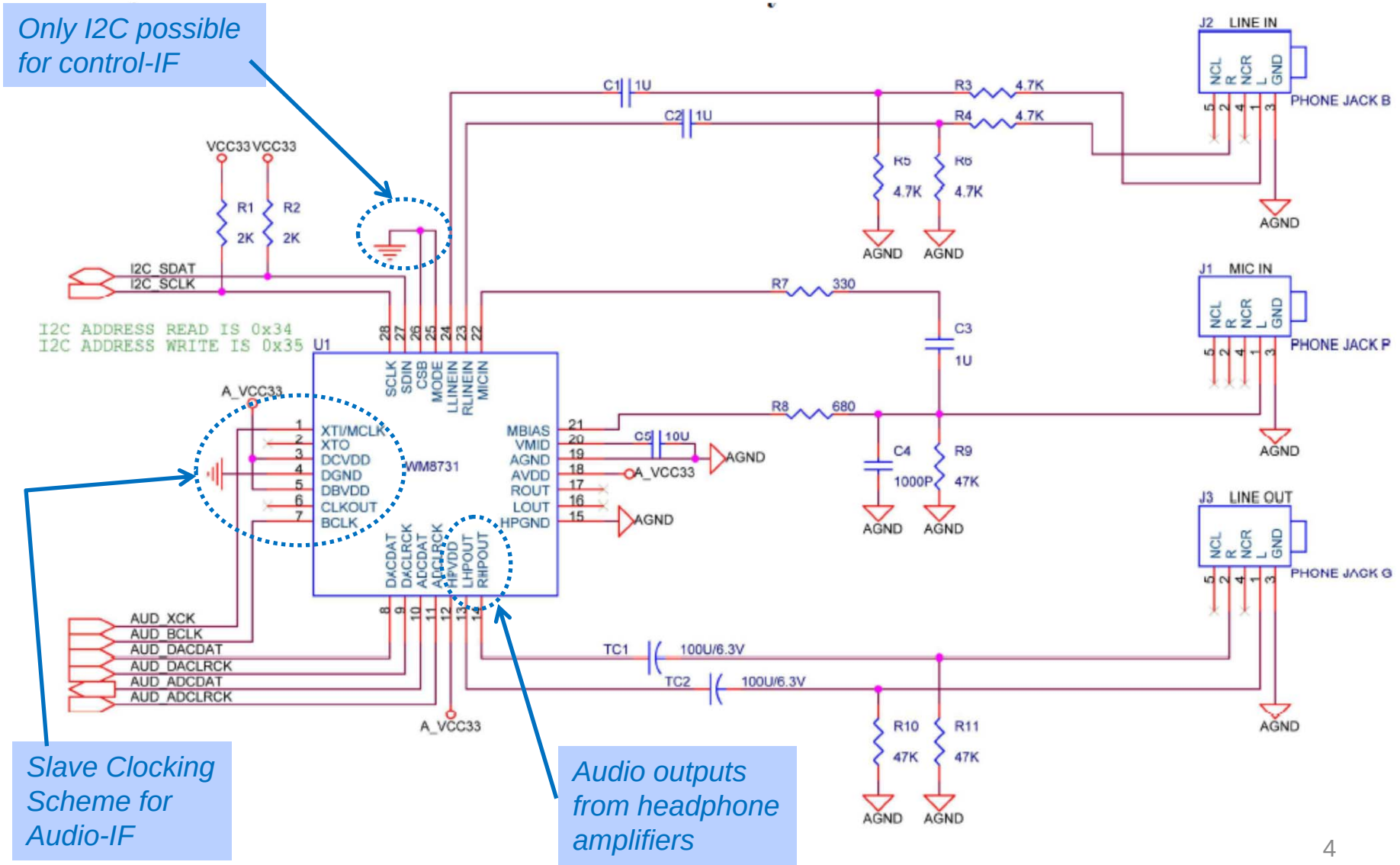
School of Engineering



Audio Codec Settings

WM8731 on the DE2-115 Board

School of Engineering



Audio Codec Settings

WM8731 in the DE2-115 Board

School of Engineering

Clocking Concept

- **FPGA-System-Clock**

Alle Blöcke in FPGA (ausser Taktteiler) werden mit dem clk_12M5 getriggert

(synchrones Design mit «single clock-domain»).

- **MCLK:**

Der Master Clock für den Audio Codec (MCLK) ist gleichzeitig auch der FPGA System Clock (clk_12M)

Bmk: *Die I2S Schnittstelle des WM8731 wird in Slave Mode betrieben.*
Das BCLK Signal der I2S Schnittstelle wird ebenfalls im FPGA erzeugt (von clk_12M abgeleitet).

Audio Abtastrate

Verschiedene Abtastraten möglich, unteres Beispiel «Normal Mode 48kHz»

(für andere Abtastraten siehe WM8731 Datenblatt Abschnitt Audio Data Sampling Rates, Ausschnitt unten)

SAMPLING RATE		MCLK FREQUENCY	SAMPLE RATE REGISTER SETTINGS					DIGITAL FILTER TYPE
ADC	DAC							
kHz	kHz	MHz	BOSR	SR3	SR2	SR1	SR0	
48	48	12.288	0 (256fs)	0	0	0	0	1

Table 18 Normal Mode Sample Rate Look-up Table

Audio Codec Settings

Register Map

School of Engineering

- **R0 :** Left Line In [8] LRINBOTH; [7] LINMUTE; [4:0] LINVOL
- **R1 :** Right Line In [8] RLINBOTH; [7] RINMUTE; [4:0] RINVOL
- **R2 :** Left Headphone Out …(siehe Tabelle)
- **R3 :** Right Headphone Out
- **R4 :** Analog Audio Control Path
- **R5 :** Digital Audio Path Control
- **R6 :** Power Down Control
- **R7 :** Digital Audio Interface Format
- **R8 :** Sampling Control (sample rate)
- **R9 :** Active Control (activate digital audio interface)
- **R10 - 14 :** Reserved_A ; _B; _C; _D; _E
- **R15 :** Reset

Audio Codec Settings

Register Map

School of Engineering

- *There are 11 registers with 16 bits per register (7 bit address + 9 bits of data).*

REGISTER	B 15	B 14	B 13	B 12	B 11	B 10	B 9	B8	B7	B6	B5	B4	B3	B2	B1	B0
R0 (00h)	0	0	0	0	0	0	0	LRIN BOTH	LIN MUTE	0	0	LINVOL				
R1 (02h)	0	0	0	0	0	0	1	RLIN BOTH	RIN MUTE	0	0	RINVOL				
R2 (02h)	0	0	0	0	0	1	0	LRHP BOTH	LZCEN	LHPVOL						
R3 (06h)	0	0	0	0	0	1	1	RLHP BOTH	RZCEN	RHPVOL						
R4 (08h)	0	0	0	0	1	0	0	0	SIDEATT		SIDETONE	DAC SEL	BY PASS	INSEL	MUTE MIC	MIC BOOST
R5 (0Ah)	0	0	0	0	1	0	1	0	0	0	0	HPOR	DAC MU	DEEMPH		ADC HPD
R6 (0Ch)	0	0	0	0	1	1	0	0	PWR OFF	CLK OUTPD	OSCPD	OUTPD	DACPD	ADCPD	MICPD	LINEINPD
R7 (0Eh)	0	0	0	0	1	1	1	0	BCLK INV	MS	LR SWAP	LRP	IWL		FORMAT	
R8 (10h)	0	0	0	1	0	0	0	0	CLKO DIV2	CLKI DIV2	SR				BOSR	USB/ NORM
R9 (12h)	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	ACTIVE
R15(1Eh)	0	0	0	1	1	1	1	RESET								
	ADDRESS							DATA								

Table 29 Mapping of Program Registers

Audio Codec Settings

Example: W8731_ADC_DAC_M12DB_48K

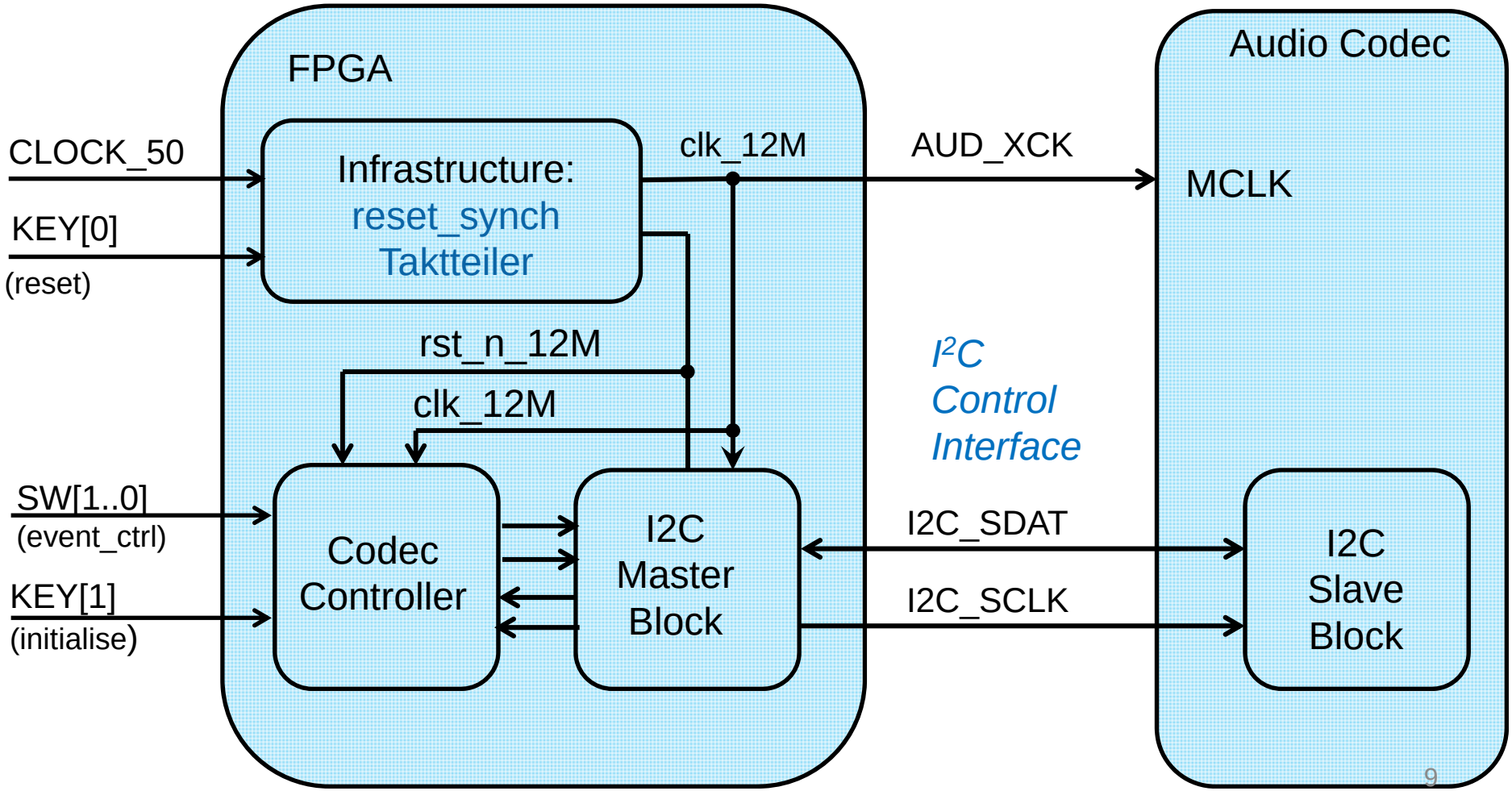
School of Engineering

Register	Address Bits B[15:9]	Data Bits B[8:0]	Beschreibung
R0 LEFT_LINE_IN	*"0000000" = 0x00*	*"000001111" = 0x00F*	No mute, no both, volume -12 dB
R1 RIGHT_LINE_IN	*"0000001"* = 0x01	*"000001111" = 0x00F*	No mute, no both, volume -12 dB
R2 LEFT_HP_OUT	*"0000010"* = 0x02	"001101101" = 0x06D	No zero-cross, no both, volume -12dB
R3 RIGHT_HP_OUT	*"0000011"* = 0x03	"001101101" = 0x06D	No zero-cross, no both, volume -12dB
R4 ANALOG_AP	*"0000100"* = 0x04	"000010010" = 0x012	Mute microphone, select DAC, disable bypass, disable sidetone,…
R5 DIGITAL_AP	*"0000101"* = 0x05	"000000000" = 0x000	Disable DAC mute, disable de-emphasis, …
R6 POWER_DOWN	*"0000110"* = 0x06	"000000000" = 0x000	No power down mode selected
R7 DIGITAL_AI	*"0000111"* = 0x07	"000000000" = 0x000	Audio-IF I2S Mode, audio data 16 bits wide, …
R8 SAMPLING	*"0001000"* = 0x08	"000000000" = 0x000	Normal mode, MCLK 12,28MHz, sample rate 48kHz (both ADC and DAC), SR=[0011], …
R9 DIGITAL_ACTIVATE	*"0000111"* = 0x09	"000000001" = 0x001	Activate Audio Digital Interface
R10 – R14 RESERVED A-E	0x0A - 0x0E	"000000000" = 0x000	No need to write
R15 RESET	*"0001111"* = 0x0F	"000000000" = 0x000	Only write if SW-reset required

Codec Control Interface

School of Engineering

I2C Bus, Two-Wire Serial Interface: I2C_SDAT (data), I2C_SCLK (clock)



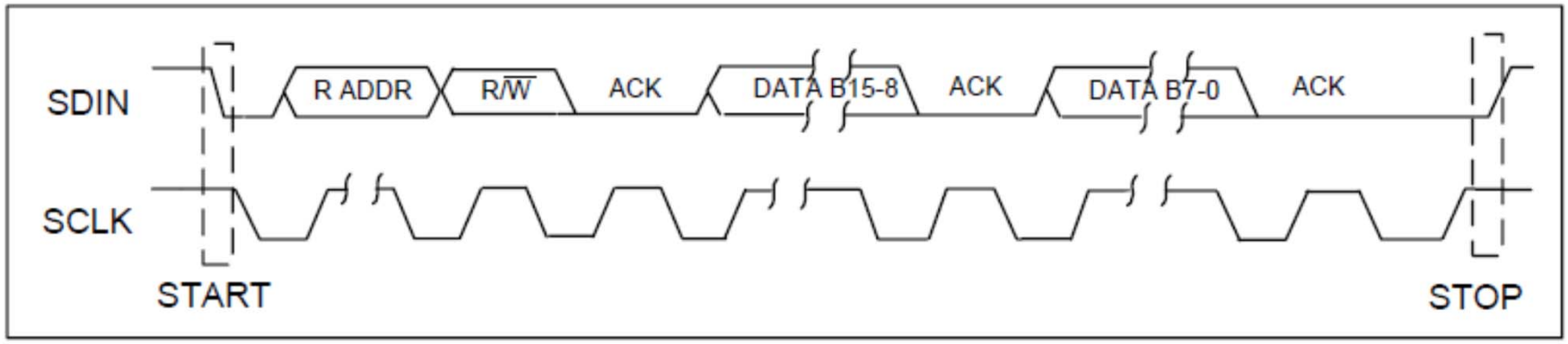
Codec Control Interface

WM8731 - Datenblatt Auszug

School of Engineering

2-WIRE SERIAL CONTROL MODE

The WM8731/L supports a 2-wire MPU serial interface. The device operates as a slave device only. The WM8731/L has one of two slave addresses that are selected by setting the state of pin 15, (CSB).



B[15:9] are Control Address Bits

B[8:0] are Control Data Bits

ADDRESS
0011010

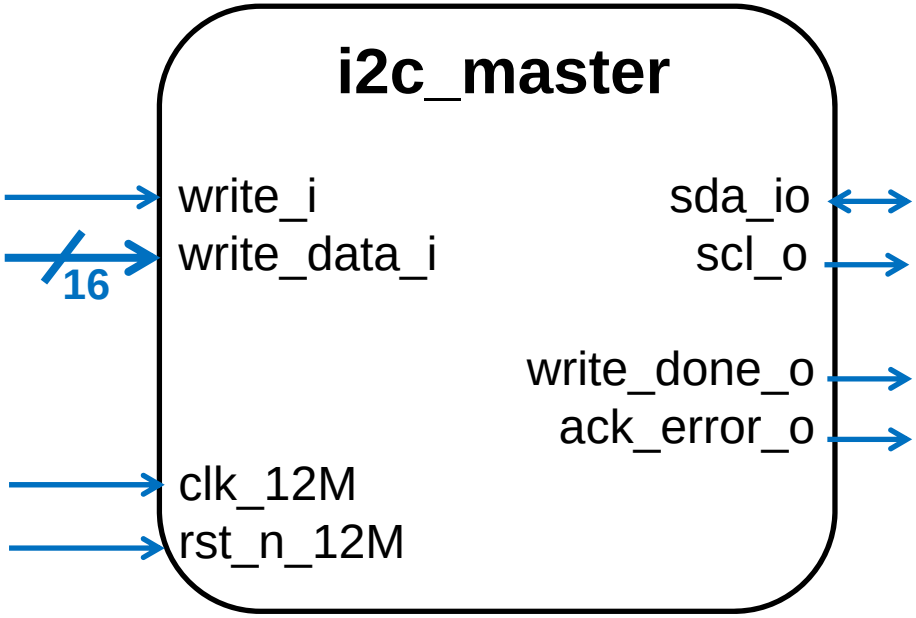
I^2C Slave device address

Codec Control Interface

School of Engineering

I2C Master Block

Block wird zu Verfügung gestellt: i2c_master.vhd



Aufpassen:

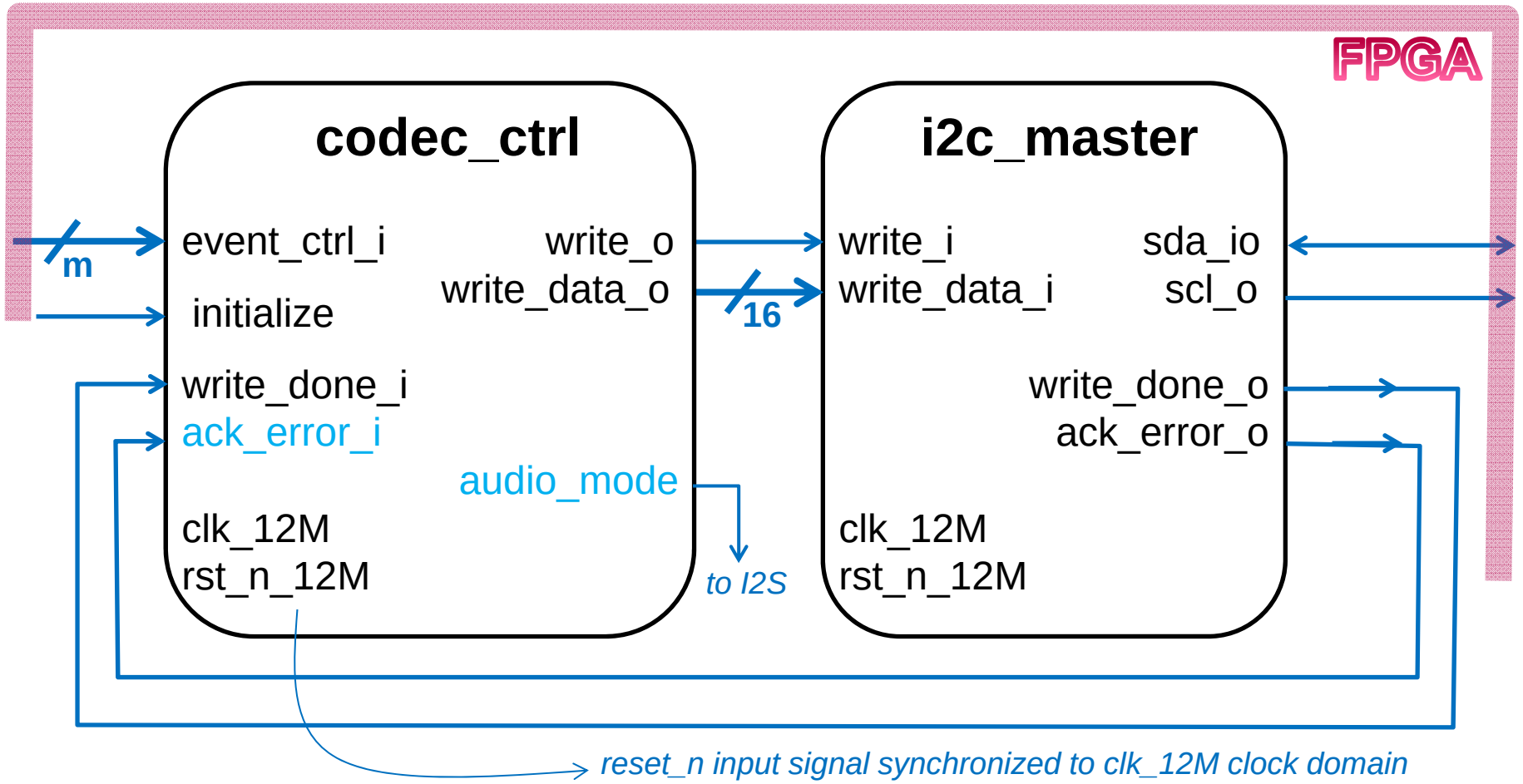
- *SDA is an INOUT Port;*
- *Write_data contains*
  7-bits register-address
  9-bits register-value

Für Integration im Top-Level Block:

- *Component Declaration*
- *Signale für Verbindungen*
- *Instantiation*

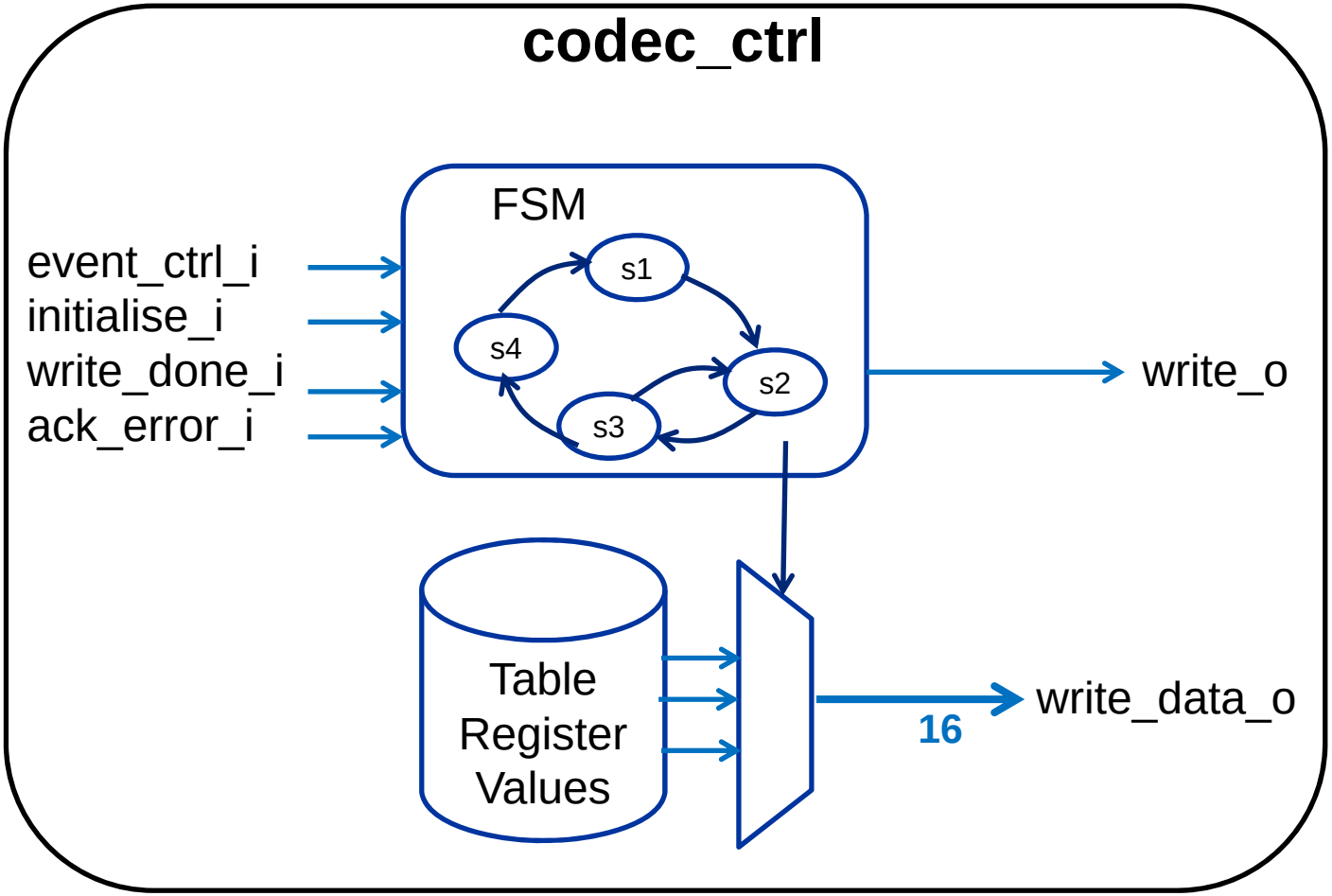
Codec Control Interface

School of Engineering



Codec Control Interface

School of Engineering



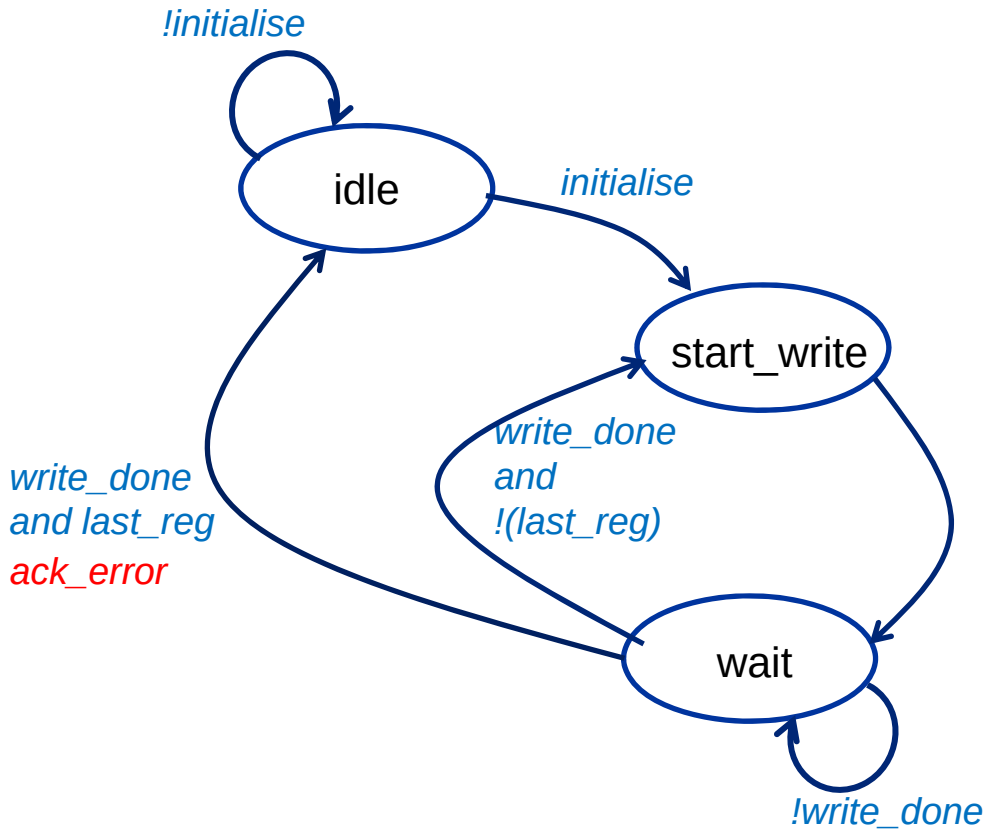
Example event_ctrl_i (via button or switches):

init_codec_default, set_analog_bypass, set_volume_0dB, …

Codec Control Interface

School of Engineering

Finite State Machine (Automaten) of codec_ctrl block

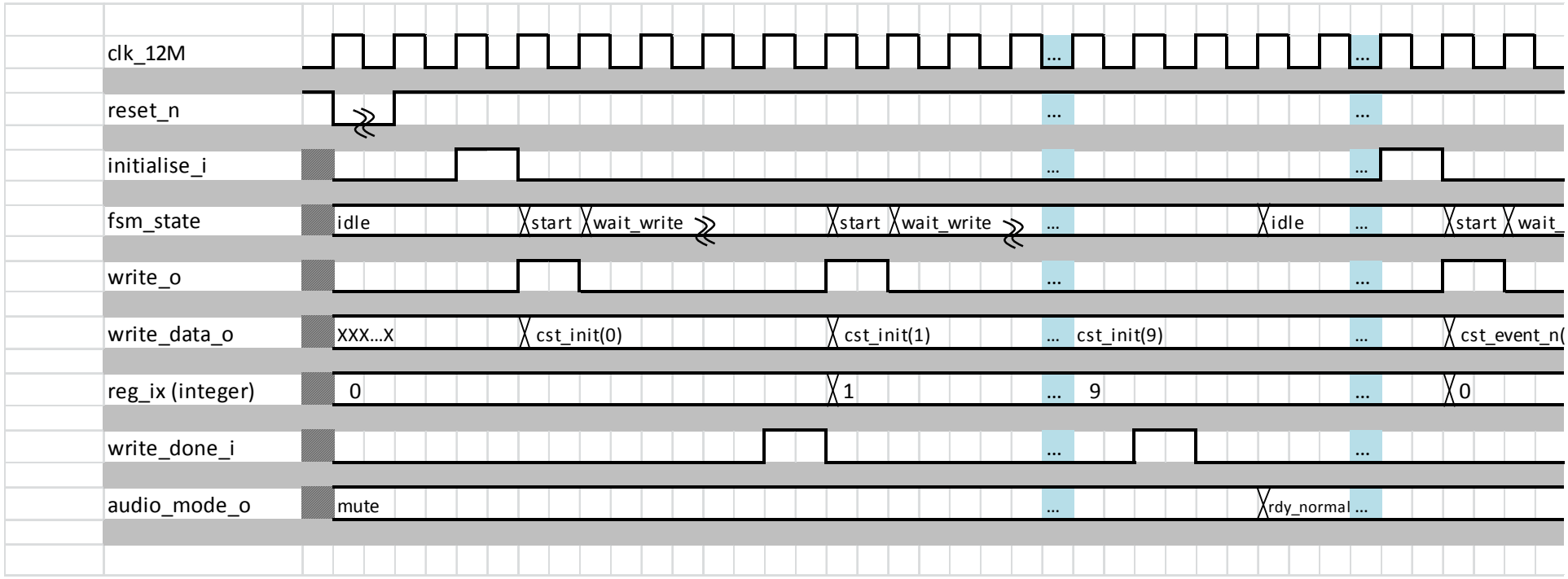


Obs.: Table of Register Values wird im Lab3-Material gegeben (als Package)

Codec Control Interface

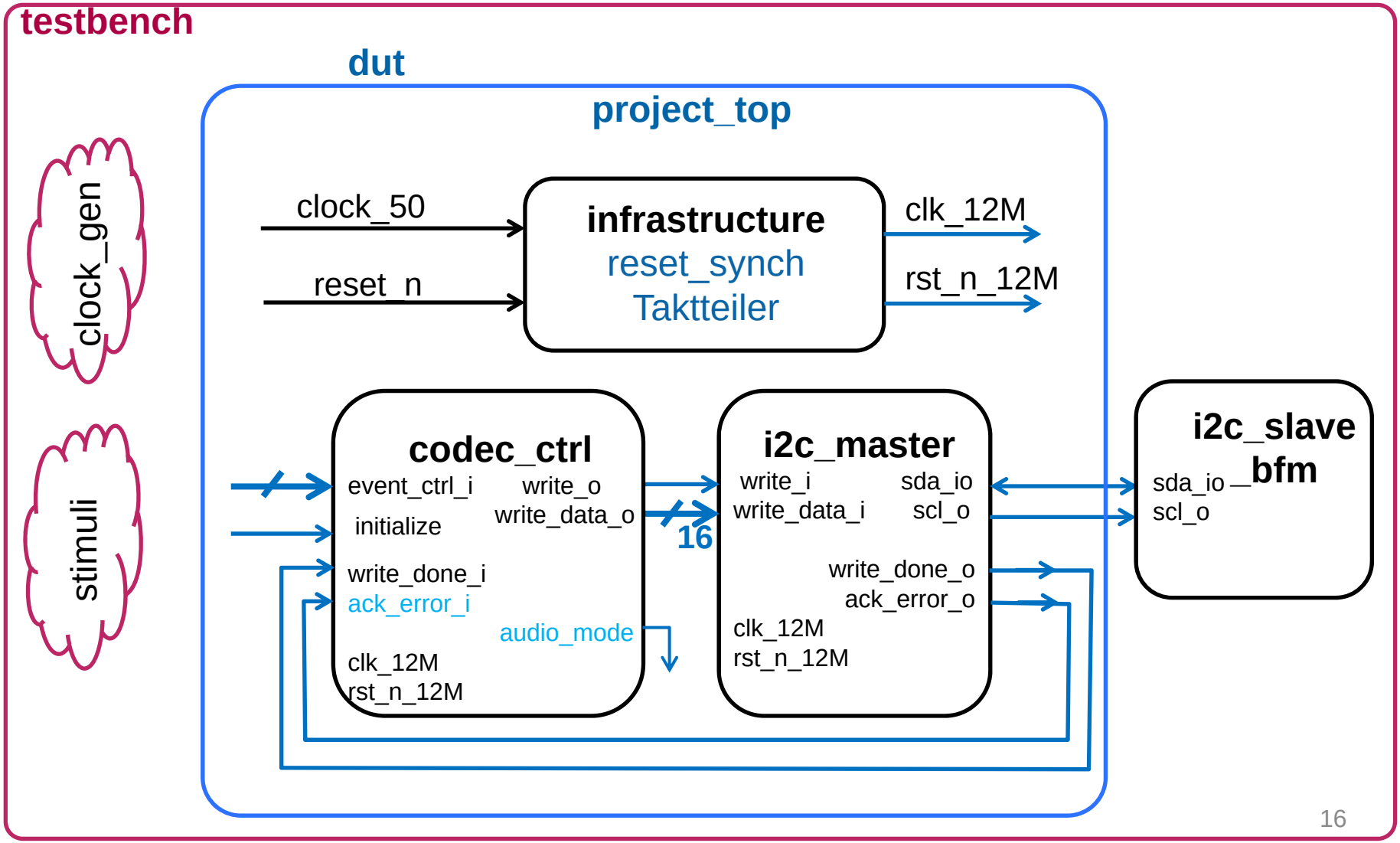
School of Engineering

Timing between codec_ctrl and i2c_master blocks



Top Level Testbench für Codec Control Interface

School of Engineering



Audio Codec Settings

Quiz: Which bits change for analog bypass mode?

School of Engineering

Register	Address Bits B[15:9]	Data Bits B[8:0]	Beschreibung
R0 LEFT_LINE_IN	*"0000000"* = *0x00*		
R1 RIGHT_LINE_IN	*"0000001"* = 0x01		
R2 LEFT_HP_OUT	*"0000010"* = 0x02		
R3 RIGHT_HP_OUT	*"0000011"* = 0x03		
R4 ANALOG_AP	*"0000100"* = 0x04		
R5 DIGITAL_AP	*"0000101"* = 0x05		
R6 POWER_DOWN	*"0000110"* = 0x06		
R7 DIGITAL_AI	*"0000111"* = 0x07		
R8 SAMPLING	*"0001000"* = 0x08		
R9 DIGITAL_ACTIVATE	*"0000111"* = 0x09		
~~*R10 – R14 RESERVED A-E*~~	~~0x0A - 0x0E~~	~~"000000000" = 0x000~~	~~No need to write~~
~~*R15 RESET*~~	~~*"0001111"* = 0x0F~~	~~"000000000" = 0x000~~	~~Only write if SW-reset required~~

Audio Codec Settings

Example: W8731_BYPASS_M12DB_8K

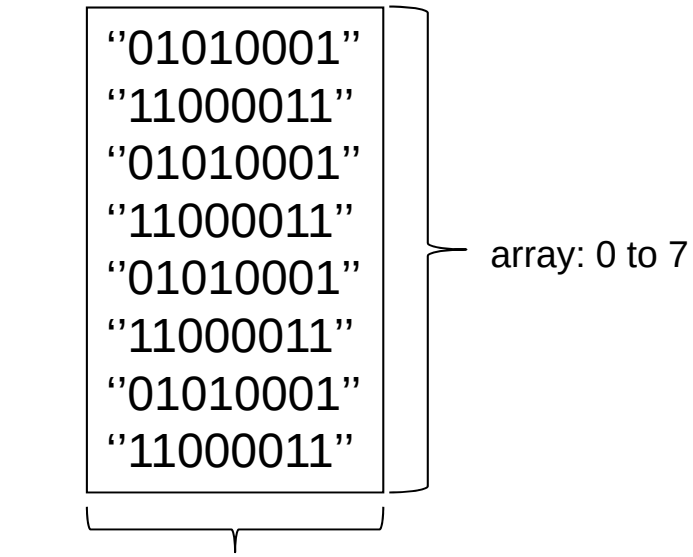
School of Engineering

Register	Address Bits B[15:9]	Data Bits B[8:0]	Beschreibung
R0 LEFT_LINE_IN	*"0000000" = 0x00*	*"000001111" = 0x00F*	No mute, no both, volume -12 dB
R1 RIGHT_LINE_IN	*"0000001"* = 0x01	*"000001111" = 0x00F*	No mute, no both, volume -12 dB
R2 LEFT_HP_OUT	*"0000010"* = 0x02	"001101101" = 0x06D	No zero-cross, no both, volume -12dB
R3 RIGHT_HP_OUT	*"0000011"* = 0x03	"001101101" = 0x06D	No zero-cross, no both, volume -12dB
R4 ANALOG_AP	*"0000100"* = 0x04	"000001010" = 0x00A	Mute microphone, disable DAC, enable bypass, disable sidetone,…
R5 DIGITAL_AP	*"0000101"* = 0x05	"000001000" = 0x008	Enable DAC mute, disable de-emphasis, …
R6 POWER_DOWN	*"0000110"* = 0x06	"000000000" = 0x000	No power down mode selected
R7 DIGITAL_AI	*"0000111"* = 0x07	"000000000" = 0x002	Audio-IF I2S Mode, audio data 16 bits wide, …
R8 SAMPLING	*"0001000"* = 0x08	"000000000" = 0x00C	Normal mode, MCLK 12,28MHz, sample rate 48kHz (both ADC and DAC), SR=[0011], …
R9 DIGITAL_ACTIVATE	*"0000111"* = 0x09	"000000001" = 0x001	Activate Audio Digital Interface
R10 – R14 RESERVED A-E	0x0A - 0x0E	"000000000" = 0x000	No need to write
R15 RESET	*"0001111"* = 0x0F	"000000000" = 0x000	Only write if SW-reset required

Arrays in VHDL

Typendefinierung eines Arrays

School of Engineering



Beispiel für Typendefinierung eines Arrays:

```
type codec_array is array(0 to 7) of std_logic_vector(7 downto 0);
```

Typen Deklaration erfolgt in Architektur zwischen *Architecture* und *Begin* oder im Package

Array mit Konstanten füllen

School of Engineering

```
Constant: C_W8731_ANALOG_BYPASS: codec_array :=
(
0 => “011001101”
1 => “111001101”,
2 => “111001101”,
3 => “111001101”,
4 => “011001101”,
5 => “011001101”,
6 => “011001101”,
7 => “011001101”
8 => “011001101”
9 => “011001101”
);
```

Achtung: Dateninhalt nur Beispiel

Auslesen eines Arrays

School of Engineering

In Entity:

```
output_data : out std_logic_vector(7 downto 0);
```

In Architektur:

```
output_data <= C_W8731_ANALOG_BYPASS(array_pointer);
```

Packages

VHDL Package

School of Engineering

Definition:

```
package reg_table_pkg is
.
.
type codec_array is array(0 to 9) of std_logic_vector(8 downto 0);
.
.
end package;
```

Benutzung in Entity Architektur Datei:

```
use work.reg_table_pkg.all;
```