SC8800 Device Specifications

Version: 1.0

HW-SC8800-DS-0005

2006-6-6

www.spreadtrum.com

IMPORTANT NOTICE

COPYRIGHT NOTICE

TRADEMARKS

Spreadtrum Communications, Inc. and Spreadtrum Communications, Inc.'s products are exclusively owned by Spreadtrum Communications, Inc. References to other companies and their products use trademarks owned by the respective companies and are for reference purpose only.

WARRANTY DISCLAIMER

Spreadtrum Communications, Inc. makes no representations or warranties, either express or implied, by or with respect to anything in this document, and shall not be liable for any implied warranties of merchantability or fitness for a particular purpose or for any indirect, special or consequential damages.

CONFIDENTIALITY

The information contained herein (including any attachments) is confidential. The recipient hereby acknowledges the confidentiality of this document, and except for the specific purpose, this document shall not be disclosed to any third party.

Revision History

Version	Data	Owner	Note
0.1	2006/4/3	Jin Ji	First draft.
0.2	2006/4/4	Jin Ji	Changed 1. The reset state on the pins. 2. The strapping pin table.
0.3	2006/4/14	Jin Ji	Changed according to Guo Wei's comments: 1. TX DAC common mode voltage to 1.4 V; 2. AFC, AGC and APC DAC output range to 0.2 to 1.7/2.1 V.
0.4	5/27/06	Jin Ji	Changed the doc according to the review feedback, except SIM clock calculation.
0.5	6/2/06	Jin Ji	Interchanged PLL M and N.
1.0	6/6/06	Jin Ji	Changed ARM IRQ input [16] to syscnt_irq | PBINT | CHINT.

Table of Contents

List of Figures

List of Tables

1 Overview

SC8800 is a TD-SCDMA and GSM/GPRS dual-mode baseband (BB) chip for user terminals. This document specifies the functions and features of SC8800.

1.1 SC8800 Features

SC8800 is Spreadtrum's 1st-generation TD-SCDMA/GSM/GPRS ASIC solution chip. It inherits most of the features from SC6600 GSM/GPRS series (with improvements), with the addition of TD-SCDMA dedicated functions.

- Low power and high-performance device in a 0.18 µm mixed signal CMOS technology.
  - External supply voltages: battery 3.6 V (typical), backup battery (optional) 3.0 V (typical) and RTC power supply 1.8 V (typical).
  - Internal supply voltages: analog 2.8 V (typical), digital I/O 3.0 V or 1.8 V (typical) and digital core 1.8 V (typical).
- 13 x 13 mm^2 289-ball 0.8 ball pitch LFBGA package.
- Compatible with TD-SCDMA LCR standard in 3GPP Release 1999.
- Dedicated TD-SCDMA signals processing engine for signal conditioning, joint detection, Viterbi decoding and Turbo decoding.
- Compatible with GSM/GPRS Release 1999, DCS1800 and PCS1900 recommendations.
- Dedicated GSM/GPRS signal processing engine for equalization, channel encoding/decoding for all traffic and control channels, GMSK modulation and encryption/decryption (A5/1 and A5/2, GEA 1 and GEA 2 algorithms).
- System timing:
  - Low swing 26/13 MHz master clock input.
  - Programmable TDMA timing with 1/8 chip resolution in TD-SCDMA mode or 1/4 bit resolution in GSM mode.
  - Time tracking in power saving mode.
- Embedded microcontroller (MCU) for protocol stack and peripheral control:
  - ARM926EJ®, 32-bit RISC processor.
  - Separate 16K I-Cache and 16K D-Cache.
  - Separate instruction and data AMBA AHB bus interfaces.
  - Support the 32-bit ARM and 16-bit Thumb instruction sets.
  - Support Java extention intruction sets for efficient execution of Java byte codes.
  - External memory interface supports both SDRAM and Flash/SRAM.
  - JTAG for test and In-Circuit Emulation.
- Embedded DSP core for all TD-SCDMA and GSM/GPRS specific signal processing tasks:
  - Teak®, 16-bit fixed point DSP core.
  - On-chip program RAM.
  - On-chip data RAM.
- Complete in-phase and quadrature (I/Q) component interface between the Digital Signal Processor (DSP) and RF module.
- Dedicated RF serial control interface and parallel control signals.
- Complete voice band codec:
  - Audio signal conversion between microphone/earphone and DSP.
  - Second set converters for auxiliary microphone/speaker.
  - Support of the Digital Audio Interface (DAI).
  - Stereo audio output.
  - Integrated microphone bias.

- Five[1] auxiliary analog inputs to a 10-bit analog-to-digital converter (ADC) for measurement purposes.
- Three auxiliary analog outputs from digital-to-analog converters (DAC) for control purposes (AFC, RF power ramping control and one spare).
- SIM card interface with 3.0 and 1.8 V SIM card support.
- 3.0 V as well as 1.8 V memory and I/O support.
- Larger keypad, up to 5 x 6.
- Serial/parallel interfaces:
  - Three UART, up to 460 k baud rate.
  - IrDA.
  - Two-wire serial interface.
  - Three-wire serial interface.
- On-chip PLL for programmable microcontroller and DSP clocks.
- Real time clock (RTC) and alarm running on a 32.768 kHz crystal.
- Power management with LDO for on-chip as well as off-chip circuitry.

ARM926EJ® is a registered trademark of Advanced RISC Machines Limited.
Teak® is a registered trademark of CEVA, Inc.

1.2 Applications

SC8800 provides a single-chip baseband solution to TD-SCDMA/GSM dual-mode wireless telephone handsets and data modems conforming to the TD-SCDMA standard (3GPP) and GSM/GPRS (Release 1999, GSM850, GSM900, DCS1800 and PCS1900, quad-band).

1.3 General Description

SC8800 is a highly integrated mixed signal baseband processor for TD-SCDMA and GSM/GPRS applications. It consists of an embedded 32-bit microcontroller and an embedded 16-bit DSP core and integrated many TD-SCDMA/GSM/GPRS-specific hardware accelerators and analog functions to simplify the system designs.

The embedded MCU runs a real-time operation system (RTOS), performs the system control functions according to the GSM protocol stack, and services all peripheral components including man-machine interface.

The embedded DSP provides data processing power for many TD-SCDMA and GSM/GPRS-specific physical layer signal processing, including power measurement of the serving and neighboring cells, frequency estimation and timing track, channel estimation/equalization, baseband data interleaving/deinterleaving, burst building, speech encoding/decoding, etc., and controls the operation of ASIC hardware accelerators.

The system timing control provides clocks for the system and controls the mode of the system operations. The system timing control consists of the following parts:

- A master clock input, which is the system clock for normal GSM operations.
- A phase locked loop (PLL), which is used to generate clocks with higher frequencies.
- A real time clock (RTC), which is used to count the seconds, minutes and hours and to maintain the system timing during the power-saving (sleep) mode.
- A system timer, which generates the TD-SCDMA and GSM-specific timing bases.
- A system state machine, which controls the system's normal operation and power-saving (sleep) modes.

[1] SC8800A1 has 6 inputs, and SC8800A2 has 5 inputs.

The baseband RF interface is a complete interface between an RF module and the on-chip digital signal processor (DSP). It consists of the following major parts:

- A transmit path, which converts a bit stream to analog quadrature (I/Q) signals to an RF transceiver.
- A receive path, which converts analog quadrature (I/Q) signals from an RF transceiver to digital signals.
- An event scheduler, which controls the timing of the baseband Tx and Rx actions with the eighth-chip or quarter-bit accuracy.

The RF interface also consists of the following parts:

- An RF power amplifier control, which controls the timing of an RF power amplifier ON/OFF as well as its output power.
- An RF serial port, which is used to program control registers inside the RF module.
- RF parallel controls, which form a group of control signals to control the ON/OFF of the RF active components and passive couplers/switches.

For the TD-SCDMA mode, accelerators are used for joint-detection, Viterbi decoding and turbo decoding.

For the GSM/GPRS mde, the Viterbi channel equalizer (VE) is a hardware accelerator for recovering the information bits from the baseband digital I/Q data.

The baseband channel encoder/decoder performs encoding and decoding functions for all GSM/GPRS traffic and control channels. It consists of the following major parts:

- A convolution coder (CC), which performs the convolution encoding with 1/2, 1/3 or 1/6 rate.
- A cyclic redundancy coder/decoder (CRC), which generates/checks parity bits for some of the information bits.
- A Viterbi decoder (VA), which performs the decoding for the convolution codes with both hard and soft bits provided from the Viterbi channel equalizer (VE).
- A FIRE decoder, which decodes the FIRE codes for FACCH/SACCH and some other control channels.

The voice band interface is a complete interface between a microphone/earphone and the DSP, and it consists of the following major parts:

- An uplink path, which converts the analog signal from a microphone to digital signals, performs filtering and outputs the data to the DSP.
- A downlink path, which converts the digital signals from the DSP to analog signals to drive an earphone or a speaker.
- A Digital Audio Interface (DAI), which is primarily used for Full Type Approval (FTA).

The peripheral control section provides Man-Machine-Interface (MMI) for keypad, LCD, etc. and serial data ports.

The auxiliary Analog-to-Digital Converter (ADC) module provides five input channels[1] that can be used for battery management or other applications.

The auxiliary Digital-to-Analog Converter (DAC) blocks supports Automatic Frequency Control (AFC), Automatic Gain Control (AGC) and RF power ramping.

The integrated analog baseband processor provides ADCs and DACs for the baseband and voice band up/down links, power management circuits, all auxiliary drivers, PLL, a band-gap reference, and LDO regulators.

[1] SC8800A1 has 6 inputs, and SC8800A2 has 5 inputs.

Proprietary architectures and algorithms were developed for low power ASIC design and power management. Unique techniques are used for noise/offset calibration and cancellation.

1.4 Block Diagram

Figure 1 shows the chip-level functional block diagram of SC8800. This chip architecture is based on two processor subsystems, an MCU and a DSP subsystem, and other functional blocks are connected to one or both of the buses and provide various hardware accelerations and interfaces to other components in the system.

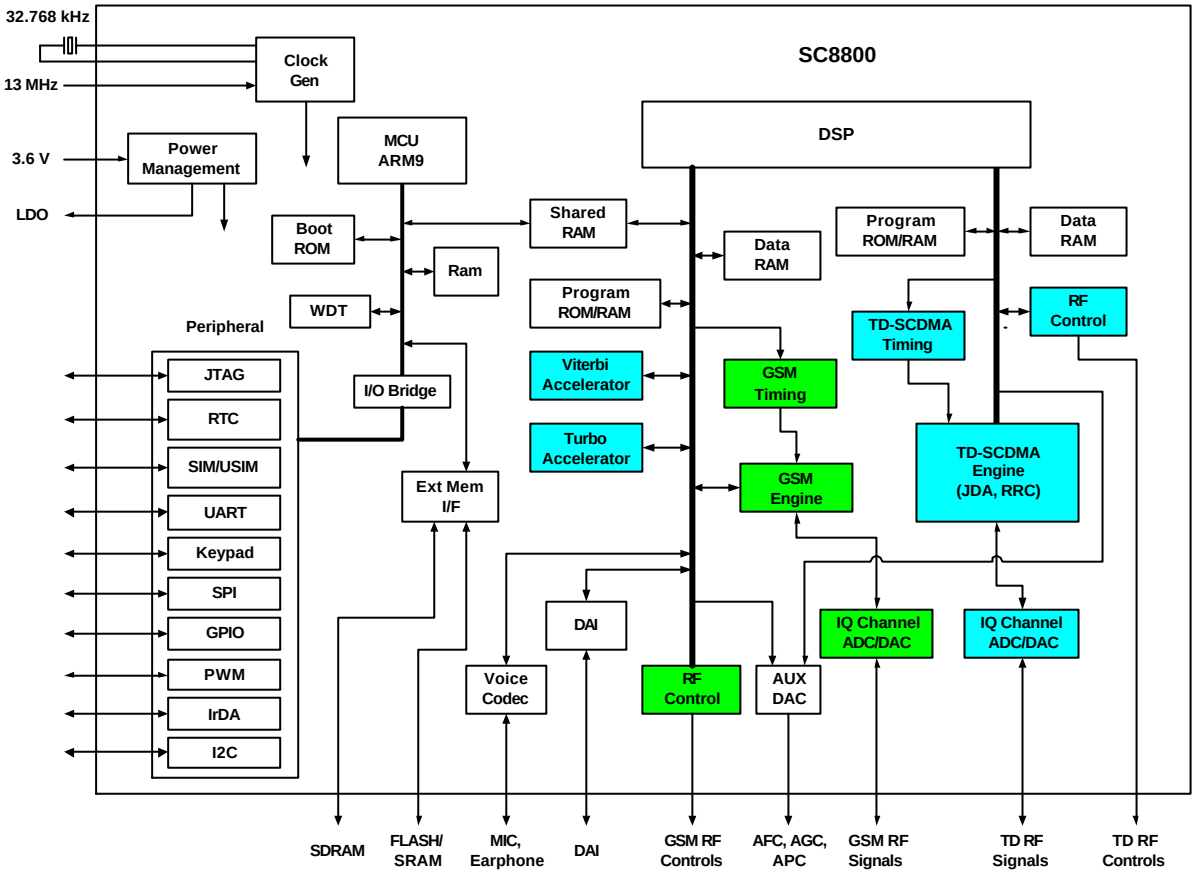


Figure 1: SC8800 chip-level functional block diagram.

2 Pin Information

This section describes SC8800 external pins. It should be noted that some of the pins have more than one function for different applications. In that case, the function can be selected through control registers.

2.1 Pin Type Description

Table 1 explains the pin type symbols used in the pin lists.

Table 1: I/O pin type description.

Symbol	Type Description
I	Digital input
O	Digital output
O/T	Digital output with tri-state option
P	Power supply
G	Ground
VBAT	Battery power supply input
VDD	Power supply for core
VIO	VDDIO, power supply for I/O
VMEM	Power supply for Flash and SRAM memories
VDRAM	Power supply for SDRAM
VSIM	Power supply for SIM interface
AI	Analog input
AO	Analog output
AV	Analog VDD
AG	Analog ground

2.2 Pin Groups

The following table lists the pins by function groups.

Table 2: Pin Description by Groups.

Pins	Description
RTC pins	
RTC32KO	real time clock output
RTC32KI	real time clock input
Clock input, Crystal enable	
MCLKI	master clock input
XTLEN	Crystal enable
SIM pins	
SIMCLK	SIM clock
SIMDA1	SIM data1
SIMRST1	SIM reset1
SIMDA2	SIM data2
SIMRST2	SIM reset2

Pins	Description
SRAM/Flash pins	
EMBH	SRAM byte high
EMBL	SRAM byte low
EMCSN[3:0]	SRAM CS signal, connecting to the chip select pins.
EMOEN	SRAM output enable, connecting to the output enables of memory devices to control data bus drivers
EMWEN	SRAM write enable, connecting to the write enables of memory devices.
EMADV	SRAM/Flash interface, address valid
EMA[24:0]	SRAM address bus, signals the address requested for memory accesses.
EMD[15:0]	SRAM data bus.
CLKSMEM	SRAM/Flash clock
SDRAM pins	
DMBH	SDRAM byte high
DMBL	SDRAM byte low
DMCKE	SDRAM clock enable
DMBA[1:0]	SDRAM bank select
DMCAS	Connect to the column address strobe (CAS) pins for all banks of SDRAM
DMRAS	Connect to the row address strobe (RAS) pins for all banks of SDRAM
DMACSN[3:0] CLKDMEM	SDRAM chip select signal SDRAM clock
DMWEN DMA[12:0]	SDRAM write enable signal SDRAM address bus
EMD[15:0]	SDRAM data bus
RF SPI pins	
RFSDA	RF serial port data
RFSCK	RF serial port clock
RFSEN [3:1]	RF serial port latch enable
RF TIMING CONTROL pins	
RFCTL [0:15]	RF Timing control
UART2/SPI2 pins	
U2CSN3	UART select
U2CTSN	UART clear-to-send
U2DSRN	UART data-set-ready
U2DTRN	UART data-terminal-ready
U2RTSN	UART request-to-send
U2RXD	UART receive
U2TXD	UART transmit
ARM JTAG pins	
MTCK	ARM JTAG clock
MTDI	ARM JTAG data input
MTDO	ARM JTAG data output
MTMS	ARM JTAG mode select
MTRSTN	ARM JTAG reset, active low

Pins	Description
UART0/SPI0 pins	
U0CSN3	UART select
U0CTSN	UART clear-to-send
U0DSRN	UART data-set-ready
U0DTRN	UART data-terminal-ready
U0RTSN	UART request-to-send
U0RXD	UART receive
U0TXD	UART transmit
UART1/SPI1 pins	
U1CSN3	UART select
U1CTSN	UART clear-to-send
U1DSRN	UART data-set-ready
U1DTRN	UART data-terminal-ready
U1RTSN	UART request-to-send
U1RXD	UART receive
U1TXD	UART transmit
I2C pins	
SCL	I2C clock
SCA	I2C data
PWM pins	
PWMA	Pulse width modulation channel A
PWMB	Pulse width modulation channel B
KEYPAD	
KEYIN[4:0]	Keypad 5 column inputs
KEYOUT[5:0]	Keypad 6 row outputs
TD-SCDMA Rx/Tx pins	
TDRXREF	TD-SCDMA Rx internal reference, connect to a cap
TDRXQP	TD-SCDMA analog differential inputs Q positive signal from RF module to SC8800
TDRXQN	TD-SCDMA analog differential inputs Q negative signal from RF module to SC8800
TDRXIP	TD-SCDMA analog differential inputs I positive signal from RF module to SC8800
TDRXIN	TD-SCDMA analog differential inputs I negative signal from RF module to SC8800
TDTXQP	TD-SCDMA analog differential outputs Q positive signal from SC8800 to RF module
TDTXQN	TD-SCDMA analog differential outputs Q negative signal from SC8800 to RF module
TDTXIP	TD-SCDMA analog differential outputs I positive signal from SC8800 to RF module
TDTXIN	TD-SCDMA analog differential outputs I negative signal from SC8800 to RF module
GSM Rx/Tx pins	
TXREF1	GSM Tx internal reference, connect to a cap
RXREF1	GSM Rx internal reference, connect to a cap
TXQP	GSM analog differential Q positive signal to RF module (Rx/Tx shared)
TXQN	GSM analog differential Q negative signal to RF modul (Rx/Tx shared)
TXIN	GSM analog differential I negative signal to RF module (Rx/Tx shared)
TXIP	GSM analog differential I positive signal to RF module (Rx/Tx shared)

Pins	Description
APC AFC pins	
APC1	RF analog power control, control DAC to convert digital data to analog control waveform
APC2	RF analog power control, control DAC to convert digital data to analog control waveform
AFCOUT	Analog frequency control output
MIC/AUX MIC pins	
MICP	MIC plus
MICN	MIC minus
AUXMICP	Aux MIC plus
AUXMICN	Aux MIC minus
AUX ADC pins	
DACOUT	Aux DAC output
Speaker/aux Speaker pins	
AUXSPP	Aux Speaker, plus
EARP	Earphone plus
HEAD_P_LP	Headphone Left Plus
HEAD_P_LN	Headphone Left Minus
HEAD_P_RP	Headphone Right Plus
HEAD_P_RN	Headphone Right Minus
Others	
RFLNA	RF LNA
RFMIXER	RF mixer
RFPAHG PAGSM	RF PA high gain PA GSM

2.3 Pin List by Numbers

Table 3 lists SC8800 pins by numbers.

Table 3: List of SC8800 pins by ball numbers.

No.	Ball No.	Pin Name	Type	Description	Reset State
1	B1	AVDDVBO	P	External voice Band VDD	NA
2	D4	AUXSPP	AO	Aux Speaker, plus	NA
3	D3	HEAD_P_RP	AO	Headphone Right Plus	NA
4	B2	HEAD_P_RN	AO	Headphone Right Minus	NA
5	C2	HEAD_P_LN	AO	Headphone Left Minus	NA
6	E4	DGND3V	G	Mixed GND	NA
7	C1	DVDD3V	P	Mixed VDD	NA
8	D2	AVDDPLL	P	PLL VDD	NA
9	F5	PLLPF	AI	PLL low-pass filter	NA

No.	Ball No.	Pin Name	Type	Description	Reset State
10	F4	AGNDPLL	G	PLL GND	NA
11	D1	RTCVSS	G	External RTC GND	NA
12	E3	RTC32KO	AO	RTC output	NA
13	G4	RTC32KI	AI	RTC input	NA
14	E2	RTCVDD	P	External RTC VDD	NA
15	F3	MCLKI	AI	Master clock input	NA
16	H4	VLCD	P	External LCD VDD	NA
17	H3	SIMCLK	O	SIM clock	0
18	E1	SIMDA 1	I	SIM data 1	Z
19	F2	VSIM	P	External SIM VDD	NA
20	G2	SIMRST1	O	SIM reset 1	0
21	F1	SIMDA2	I	SIM data 2	Z
22	G3	SIMRST2	O	SIM reset 2	0
23	T4	VMEM	P	External SRAM VDD	NA
24	G1	EMBH	O	SRAM byte high	0, input
25	J4	VSSIO	G	IO VSS	NA
26	J2	EMBL	O	SRAM byte low	0, input
27	H2	EMCSN0	O	SRAM CS 0	1, input
28	G5	VBAT	P	External Battery In	NA
29	J5	VSS	G	VSS	NA
30	H1	EMCSN1	O	SRAM CS 1	1, input
31	J3	EMCSN2	O	SRAM CS 2	1, input
32	K3	EMCSN3	O	SRAM CS 3	1, input
33	K4	EMOEN	O	SRAM OE	1, input
34	J1	EMWEN	O	SRAM WE	1, input
35	K2	EMADV	O	SRAM/Flash Interface	0, input
36	F15	SPVDD_L	P	SP VDD	NA
37	L1	EMA0	O	SRAM address	0, input
38	L2	EMA1	O	SRAM address	0, input
39	L3	EMA2	O	SRAM address	0, input
40	K1	EMA3	O	SRAM address	1, input
41	K5	VSSIO	G	IO VSS	NA
42	T4	VMEM	P	External SRAM VDD	NA
43	M1	EMA4	O	SRAM address	0, input
44	M2	EMA5	O	SRAM address	1, input
45	M3	EMA6	O	SRAM address	1, input
46	M5	VSS	P	VSS	NA
47	N1	EMA7	O	SRAM address	0, input
48	M4	EMA8	O	SRAM address	0, input
49	P1	EMA9	O	SRAM address	0, input
50	N2	EMA10	O	SRAM address	1, input
51	G5	VBAT	P	External Battery In	NA
52	N3	EMA11	O	SRAM address	0, input
53	P2	EMA12	O	SRAM address	1, input
54	N4	EMA13	O	SRAM address	1, input
55	R1	CLKSMEM	O	SRAM/Flash clock	NA
56	P3	EMA14	O	SRAM address	1, input
57	R2	EMA15	O	SRAM address	1, input
58	T1	EMA16	O	SRAM address	1, input
59	T4	VMEM	P	External SRAM VDD	NA

No.	Ball No.	Pin Name	Type	Description	Reset State
60	P5	VSSIO	G	IO VSS	NA
61	R3	EMA17	O	SRAM address	1, input
62	T2	EMA18	O	SRAM address	0, input
63	T3	EMA19	O	SRAM address	0, input
64	U1	EMA20	O	SRAM address	0, input
65	R4	EMA21	O	SRAM address	1, input
66	R5	EMA22	O	SRAM address	0, input
67	V2	EMA23	O	SRAM address	1, input
68	V1	EMA24	O	SRAM address	1, input
69	U2	EMD0	I/O	SRAM data	Z
70	W1	EMD1	I/O	SRAM data	Z
71	F15	VDD	P	Core Power	NA
72	R7	VSS	G	VSS	NA
73	W2	EMD2	I/O	SRAM data	Z
74	V3	EMD3	I/O	SRAM data	Z
75	U3	EMD4	I/O	SRAM data	Z
76	G5	VBAT	P	External Battery In	NA
77	U4	EMD5	I/O	SRAM data	Z
78	W3	EMD6	I/O	SRAM data	Z
79	T4	VMEM	P	External SRAM VDD	NA
80	U5	EMD7	I/O	SRAM data	Z
81	W4	EMD8	I/O	SRAM data	Z
82	V4	EMD9	I/O	SRAM data	Z
83	T6	VSSIO	G	IO VSS	NA
84	V6	EMD10	I/O	SRAM data	Z
85	V5	EMD11	I/O	SRAM data	Z
86	W5	EMD12	I/O	SRAM data	Z
87	U6	EMD13	I/O	SRAM data	Z
88	U7	EMD14	I/O	SRAM data	Z
89	W6	EMD15	I/O	SRAM data	Z
90	R8	VDRAM	P	External SDRAM VDD	NA
91	T7	DMBH	O	SDRAM byte high	0
92	U8	DMBL	O	SDRAM byte low	0
93	V7	DMCKE	O	SDRAM clock enable	0
94	W7	DMBA0	O	SDRAM bank0	0
95	R9	VSS	G	VSS	NA
96	V9	DMBA1	O	SDRAM bank1	0
97	G5	VBAT	P	External Battery In	NA
98	T9	DMCAS	O	SDRAM CAS	0
99	W8	DMRAS	O	SDRAM RAS	0
100	V8	DMACSN0	O	SDRAM CS	1
101	U9	DMACSN1	O	SDRAM CS	1
102	W9	DMACSN2	O	SDRAM CS	1
103	W10	DMACSN3	O	SDRAM CS	1
104	V10	CLKDMEM	O	SDRAM clock	NA
105	T10	DMWEN	O	SDRAM WE	0
106	R10	VSSIO	G	IO VSS	NA
107	U10	DMA0	O	SDRAM address	0
108	W11	DMA1	O	SDRAM address	0

No.	Ball No.	Pin Name	Type	Description	Reset State
109	V11	DMA2	O	SDRAM address	0
110	U11	DMA3	O	SDRAM address	0
111	R8	VDRAM	P	External SDRAM VDD	NA
112	W12	DMA4	O	SDRAM address	0
113	V12	DMA5	O	SDRAM address	0
114	U12	DMA6	O	SDRAM address	0
115	F15	VDD	P	Core Power	NA
116	T12	DMA7	O	SDRAM address	0
117	R12	VSS	G	VSS	NA
118	W13	DMA8	O	SDRAM address	0
119	U13	DMA9	O	SDRAM address	0
120	V13	DMA10	O	SDRAM address	0
121	W14	DMA11	O	SDRAM address	0
122	V14	DMA12	O	SDRAM address	0
123	G5	VBAT	P	External Battery In	NA
124	W15	DMD0	I/O	SDRAM data	Z
125	U14	DMD1	I/O	SDRAM data	Z
126	V15	DMD2	I/O	SDRAM data	Z
127	U15	DMD3	I/O	SDRAM data	Z
128	T14	DMD4	I/O	SDRAM data	Z
129	W16	DMD5	I/O	SDRAM data	Z
130	R14	VSSIO	G	IO VSS	NA
131	V16	DMD6	I/O	SDRAM data	Z
132	U16	DMD7	I/O	SDRAM data	Z
133	W17	DMD8	I/O	SDRAM data	Z
134	R8	VDRAM	P	External SDRAM VDD	NA
135	T15	DMD9	I/O	SDRAM data	Z
136	V17	DMD10	I/O	SDRAM data	Z
137	W18	DMD11	I/O	SDRAM data	Z
138	T16	DMD12	I/O	SDRAM data	Z
139	U17	DMD13	I/O	SDRAM data	Z
140	R15	VSS	G	VSS	NA
141	W19	DMD14	I/O	SDRAM data	Z
142	V18	DMD15	I/O	SDRAM data	Z
143	V19	PBINT	I	Power putton	Z
144	T17	RSTN	A	System reset	NA
145	U18	XTLEN	O	Xtal enable	1
146	G5	VBAT	P	External Battery In	NA
147	U19	CLKRTC	O	RTC clock	NA
148	R17	RFSDA	O	RF SPI data	Z
149	T18	RFSCK	O	RF SPI clock	Z
150	R16	RFSEN1	O	RF SPI enable	Z
151	T19	RFSEN2	O	RF SPI enable	Z
152	P16	RFSEN3	O	RF SPI enable	Z
153	P17	RFCTL0	O	RF control	0, input
154	R18	RFCTL1	O	RF control	0, input
155	R19	RFCTL2	O	RF control	0, input
156	N15	VSSIO	G	IO VSS	NA
157	N16	RFCTL3	O	RF control	0, input
158	F15	VDD	P	Core Power	NA

No.	Ball No.	Pin Name	Type	Description	Reset State
159	P19	RFCTL4	O	RF control	0, input
160	P18	RFCTL5	O	RF control	0, input
161	N17	RFCTL6	O	RF control	0, input
162	L15	VSS	G	VSS	NA
163	N18	RFCTL7	O	RF control	0, input
164	N19	RFCTL8	O	RF control	0, input
165	C17	VDDIO	P	IO VDD	NA
166	M17	U2CSN3	O	UART2/SPI2	0
167	M16	U2CTSN	I/O	UART2/SPI2	Z
168	M19	U2DSRN	I/O	UART2/SPI2	Z
169	M18	U2DTRN	O	UART2/SPI2	1
170	L16	U2RTSN	O	UART2/SPI2	1
171	L18	RFCTL9	O	RF control	0, input
172	L19	RFCTL10	O	RF control	0, input
173	L17	RFCTL11	O	RF control	0, input
174	K16	RFCTL12	O	RF control	0, input
175	K17	RFCTL13	O	RF control	0, input
176	K19	RFCTL14	O	RF control	0, input
177	K18	RFCTL15	O	RF control	0, input
178	J17	RFLNA	O	RF LNA	0
179	J16	RFMIXER	O	RF mixer	0
180	G5	VBAT	P	Extrernal Battery In	NA
181	J19	RFPAHG	O	RF PA high gain	0
182	H15	VSSIO	G	IO VSS	NA
183	J18	MTCK	I	ARM JTAG	0
184	H18	MTDI	I	ARM JTAG	0
185	H19	MTDO	O	ARM JTAG	Z
186	H17	MTMS	I	ARM JTAG	1
187	G15	VSS	G	VSS	NA
188	H16	MTRSTN	I	ARM JTAG	0
189	F15	VDD	P	Core Power	NA
190	G19	U0CSN3	O	UART0/SPI0	1
191	G18	U0CTSN	I/O	UART0/SPI0	Z
192	G17	U0DSRN	I/O	UART0/SPI0	Z
193	F16	U0DTRN	O	UART0/SPI0	1
194	F19	U0RTSN	O	UART0/SPI0	1
195	F18	U0RXD	I	UART0/SPI0	Z
196	F17	U0TXD	I/O	UART0/SPI0	1
197	E17	SCL	I/O	I2C	Z
198	D19	SDA	I/O	I2C	Z
199	E19	U1CSN3	O	UART1/SPI1	1
200	E18	U1CTSN	I/O	UART1/SPI1	Z
201	D17	U1DSRN	I/O	UART1/SPI1	Z
202	D18	U1DTRN	O	UART1/SPI1	1
203	C19	U1RTSN	O	UART1/SPI1	1
204	C18	U1RXD	I	UART1/SPI1	Z
205	G15	VSS	G	VSS	NA
206	B19	U1TXD	I/O	UART1/SPI1	1
207	G5	VBAT	P	External Battery In	NA
208	B18	PWMA	O	PWM	0

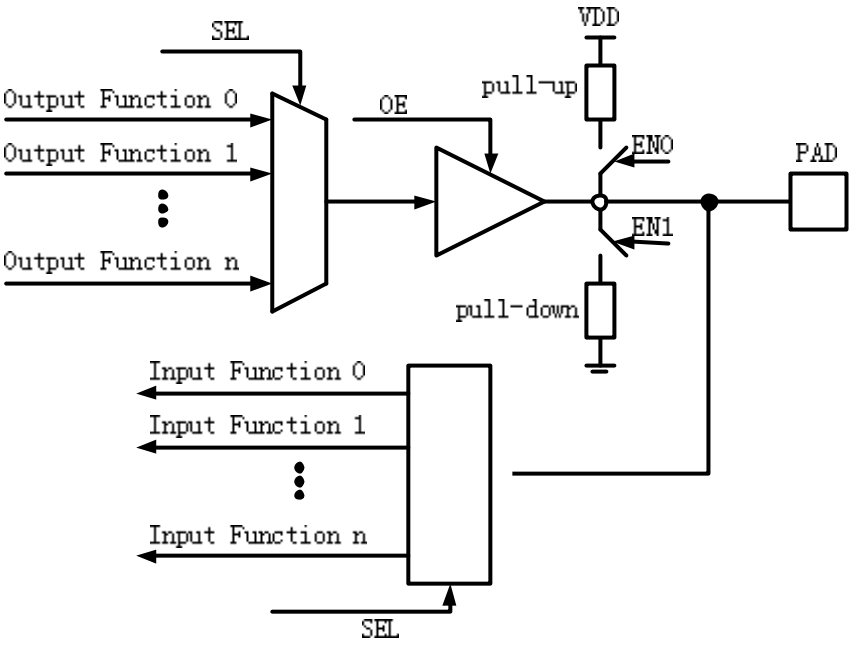
No.	Ball No.	Pin Name	Type	Description	Reset State
209	A19	PWMB	O	PWM	0
210	D16	VSSIO	G	IO VSS	NA
211	C17	VDDIO	P	IO VDD	NA
212	A18	U2RXD	I	UART2/SPI2	Z
213	B17	U2TXD	I/O	UART2/SPI2	1
214	D15	KEYIN0	I	Keypad	Z
215	C16	KEYIN1	I	Keypad	Z
216	A17	KEYIN2	I	Keypad	Z
217	B16	KEYIN3	I	Keypad	Z
218	C15	KEYIN4	I	Keypad	Z
219	F15	VDD	P	Core Power	NA
220	A16	KEYOUT0	O	Keypad	0, input
221	E14	VSS	G	VSS	NA
222	B15	KEYOUT1	O	Keypad	0, input
223	D14	KEYOUT2	O	Keypad	0
224	C14	KEYOUT3	O	Keypad	0
225	A15	KEYOUT4	O	Keypad	0
226	B14	KEYOUT5	O	Keypad	0
227	D13	LDOEN	O	LDO enable	0
228	E13	VSSIO	G	IO VSS	NA
229	G5	VBAT	P	External Battery In	NA
230	A14	TDRXREF	AI	TD-SCDMA Tx internal ref, connect to a cap	NA
231	A13	TDRXQP	AI	TD-SCDMA Rx Q positive	NA
232	B13	TDRXQN	AI	TD-SCDMA Rx Q negative	NA
233	B12	TDRXIN	AI	TD-SCDMA Rx I negative	NA
234	C13	TDRXIP	AI	TD-SCDMA Rx I positive	NA
235	C12	TDTXIN	AO	TD-SCDMA Tx I negative	NA
236	D12	TDTXIP	AO	TD-SCDMA Tx I positive	NA
237	A12	TDTXQP	AO	TD-SCDMA Tx Q positive	NA
238	B11	TDTXQN	AO	TD-SCDMA Tx Q negative	NA
239	A11	TXREF1	AI	GSM Tx internal ref, connect to a cap	NA
240	C11	TXQP	AI/O	GSM Tx Q positive	NA
241	D11	TXQN	AI/O	GSM Tx Q negative	NA
242	A10	TXIN	AI/O	GSM Tx I negative	NA
243	B10	TXIP	AI/O	GSM Tx I positive	NA
244	C10	AVDDBB	P	Analog BB VDD	NA
245	A9	PSUB	G	Substrate ground	NA
246	E11	RXREF1	AI	GSM Rx internal ref, connect to a cap	NA
247	E10	VBATA	P	Analog VDD	NA

No.	Ball No.	Pin Name	Type	Description	Reset State
248	B9	VDRV	-	Do not use	NA
249	A8	ISENSE	-	Do not use	NA
250	C9	VCHG	-	Charger input	NA
251	A7	ANATEST[1]	-	Do not use	NA
252	D10	AUXREF	AI	Aux reference, connect to a cap	NA
253	D9	APC2	AO	APC Control	NA
254	B7	APC1	AO	APC Control	NA
255	B8	AGNDAUX	G	Aux GND	NA
256	E9	AFCOUT	AO	AFC output	NA
257	C8	PAGSM	AO	PA GSM	NA
258	A6	AVDDAUX	P	AUX VDD	NA
259	D8	ADCI4	AI	ADC input PIN 4	NA
260	C7	PRODTEST[2]	-	Do not use	NA
261	B6	ADCI3	AI	ADC input PIN 3	NA
262	E8	ADCI2	AI	ADC input PIN 2	NA
263	D7	ADCI1	AI	ADC input PIN 1	NA
264	C6	ADCI0	AI	ADC input PIN 0	NA
265	A5	DACOUT	AO	Aux DAC output	NA
266	C5	LDO2	P	LDO for External Device	NA
267	E7	LDO1	P	LDO for External Device	NA
268	E10	VBATA	P	Analog VDD	NA
269	B5	VR_BG	AI	V ref, connect cap	NA
270	D6	MICBIAS	AI	Microphone bias	NA
271	A4	VBREF1	AI	Voice band ref, connect cap	NA
272	A3	AUXMICP	AI	Aux MIC plus	NA
273	D5	MICP	AI	MIC plus	NA
274	C4	MICN	AI	MIC minus	NA
275	B4	AUXMICN	AI	Aux MIC minus	NA
276	A2	AVDDVB	P	Voice band VDD	NA
277	E5	AGNDVB	G	Voice band GND	NA
278	B3	HEAD_P_LP	AO	Headphone Left Plus	NA
279	C3	EARP	AO	Earphone plus	NA
280	A1	AGNDVBO	G	External voice band GND	NA
281	J9	VSS1	G	VSS	NA
282	J10	VSS2	G	VSS	NA
283	J11	VSS3	G	VSS	NA
284	K9	VSS4	G	VSS	NA
285	K10	VSS5	G	VSS	NA
286	K11	VSS6	G	VSS	NA
287	L9	VSS7	G	VSS	NA
288	L10	VSS8	G	VSS	NA
289	L11	VSS9	G	VSS	NA

[1] This is PROG on SC8800A1, reserved.
[2] This is ADCI5 on SC8800A1, the 6th input to auxiliary ADC.

2.4 Multiplexed Pins

SC8800 uses programmable pin multiplexing so that pin count can be reduced. Multiple signals are connected to a multiplexer that connects to an I/O pin. Some I/O pins have more signals than others. Figure 2 illustrates how a multiplexed pin is configured by address location and bits to select a particular function. Referring to the control registers for control of the pin function selection.



No.	Pin Name	2nd Function Description	Type
193	U0DTRN	SSCTL1	I
194	U0RTSN	SSDR	I
195	U0RXD	SSDX	O
196	U0TXD	SSRST	I

Table 5: List of the 3rd pin functions.

No.	Pin Name	3rd Function Description	Type
153	RFCTL0	DSP2 GPIO 0	I/O
154	RFCTL1	DSP2 GPIO 1	I/O
155	RFCTL2	DSP2 GPIO 2	I/O
157	RFCTL3	DSP2 GPIO 3	I/O
159	RFCTL4	DSP2 GPIO 4	I/O
160	RFCTL5	DSP2 GPIO 5	I/O
161	RFCTL6	DSP2 GPIO 6	I/O
163	RFCTL7	DSP2 GPIO 7	I/O
164	RFCTL8	DSP2 GPIO 8	I/O
171	RFCTL9	DSP2 GPIO 9	I/O
172	RFCTL[0	DSP2 GPIO 10	I/O
173	RFCTL11	DSP2 GPIO 11	I/O
174	RFCTL12	DSP2 GPIO 12	I/O
175	RFCTL13	DSP2 GPIO 13	I/O
176	RFCTL14	DSP2 GPIO 14	I/O
177	RFCTL15	DSP2 GPIO 15	I/O

Table 6: List of the 4th pin functions.

No.	Pin Name	4th Function Description	Type
17	SIMCLK	GPIO 0	I/O
18	SIMDA 1	GPIO 1	I/O
20	SIMRST1	GPIO 2	I/O
21	SIMDA2	GPIO 3	I/O
22	SIMRST2	GPIO 4	I/O
31	EMCSN2	SCL [1]	I/O
32	EMCSN3	GPIO 6	I/O
35	EMADV	GPIO 7	I/O
66	EMA22	GPIO 8	I/O
67	EMA23	GPIO 9	I/O
68	EMA24	GPIO 10	I/O
91	DMBH	GPIO 11	I/O
92	DMBL	GPIO 12	I/O
93	DMCKE	GPIO 13	I/O
94	DMBA0	GPIO 14	I/O
96	DMBA1	GPIO 15	I/O
98	DMCAS	GPIO 16	I/O
99	DMRAS	GPIO 17	I/O

[1] This is GPIO 5 on SC8800A1.

No.	Pin Name	4th Function Description	Type
100	DMACSN0	GPIO 18	I/O
101	DMACSN1	GPIO 19	I/O
102	DMACSN2	GPIO 20	I/O
103	DMACSN3	GPIO 21	I/O
104	CLKDMEM	GPIO 22	I/O
105	DMWEN	GPIO 23	I/O
107	DMA0	GPIO 24[1]	I/O
108	DMA1	GPIO 25[2]	I/O
109	DMA2	GPIO 26	I/O
110	DMA3	GPIO 27	I/O
112	DMA4	GPIO 28	I/O
113	DMA5	GPIO 29	I/O
114	DMA6	GPIO 30	I/O
116	DMA7	GPIO 31	I/O
118	DMA8	GPIO 32	I/O
119	DMA9	GPIO 33	I/O
120	DMA10	GPIO 34	I/O
121	DMA11	GPIO 35	I/O
122	DMA12	GPIO 36	I/O
124	DMD0	GPIO 37	I/O
125	DMD1	GPIO 38	I/O
126	DMD2	GPIO 39	I/O
127	DMD3	GPIO 40	I/O
128	DMD4	GPIO 41	I/O
129	DMD5	GPIO 42	I/O
131	DMD6	GPIO 43	I/O
132	DMD7	GPIO 44	I/O
133	DMD8	GPIO 45	I/O
135	DMD9	GPIO 46	I/O
136	DMD10	GPIO 47	I/O
137	DMD11	GPIO 48	I/O
138	DMD12	GPIO 49	I/O
139	DMD13	GPIO 50	I/O
141	DMD14	GPIO 51	I/O
142	DMD15	GPIO 52	I/O
143	PBINT	GPIO 53	I/O
144	RSTN	GPIO 54	I/O
145	XTLEN	GPIO 55	I/O
147	CLKRTC	GPIO 116	I/O
148	RFSDA	GPIO 56	I/O
149	RFSCK	GPIO 57	I/O
150	RFSEN1	GPIO 58	I/O
151	RFSEN2	GPIO 59	I/O
152	RFSEN3	GPIO 60	I/O
153	RFCTL0	GPIO 61	I/O
154	RFCTL1	GPIO 62	I/O
155	RFCTL2	GPIO 63	I/O
157	RFCTL3	GPIO 64	I/O

[1] This is not available on SC8800A2. GPIO [24] is used for minute interrupt.
[2] This is not available on SC8800A2. GPIO [25] is used for charger interrupt (CHINT).

No.	Pin Name	4th Function Description	Type
159	RFCTL4	GPIO 65	I/O
160	RFCTL5	GPIO 66	I/O
161	RFCTL6	GPIO 67	I/O
163	RFCTL7	GPIO 68	I/O
164	RFCTL8	GPIO 69	I/O
166	U2CSN3	GPIO 117	I/O
167	U2CTSN	GPIO 118	I/O
168	U2DSRN	GPIO 119	I/O
169	U2DTRN	GPIO 120	I/O
170	U2RTSN	GPIO 121	I/O
171	RFCTL9	GPIO 70	I/O
173	RFCTL11	GPIO 71	I/O
174	RFCTL12	GPIO 72	I/O
175	RFCTL13	GPIO 73	I/O
176	RFCTL14	GPIO 74	I/O
177	RFCTL15	GPIO 75	I/O
178	RFLNA	GPIO 76	I/O
179	RFMIXER	GPIO 77	I/O
181	RFPAHG	GPIO 78	I/O
183	MTCK	GPIO 79	I/O
184	MTDI	GPIO 80	I/O
185	MTDO	GPIO 81	I/O
186	MTMS	GPIO 82	I/O
188	MTRSTN	GPIO 83	I/O
190	U0CSN3	GPIO 84	I/O
191	U0CTSN	GPIO 85	I/O
192	U0DSRN	GPIO 86	I/O
193	U0DTRN	GPIO 87	I/O
194	U0RTSN	GPIO 88	I/O
195	U0RXD	GPIO 89	I/O
196	U0TXD	GPIO 90	I/O
197	SCL	GPIO 91	I/O
198	SDA	GPIO 92	I/O
199	U1CSN3	GPIO 93	I/O
200	U1CTSN	GPIO 94	I/O
201	U1DSRN	GPIO 95	I/O
202	U1DTRN	GPIO 96	I/O
203	U1RTSN	GPIO 97	I/O
204	U1RXD	GPIO 98	I/O
206	U1TXD	GPIO 99	I/O
208	PWMA	GPIO 100	I/O
209	PWMB	GPIO 101	I/O
212	U2RXD	GPIO 102	I/O
213	U2TXD	GPIO 103	I/O
214	KEYIN0	GPIO 104	I/O
215	KEYIN1	GPIO 105	I/O
216	KEYIN2	GPIO 106	I/O
217	KEYIN3	GPIO 107	I/O
218	KEYIN4	GPIO 108	I/O
220	KEYOUT0	GPIO 109	I/O

No.	Pin Name	4th Function Description	Type
222	KEYOUT1	GPIO 110	I/O
223	KEYOUT2	GPIO 111	I/O
224	KEYOUT3	GPIO 112	I/O
225	KEYOUT4	GPIO 113	I/O
226	KEYOUT5	GPIO 114	I/O
227	LDOEN	GPIO 115	I/O

2.5 I/O Buffer Selection

With each I/O pad, there is a register for setting the driving strength of the output buffer, and enabling of the internal pull-up and pull-down resisters. Figure 2 shows the programmable attributes of the I/O pads. Refer to the control registers for setting up the driving strength and pull-up/down values at each I/O pad.

2.5.1 Digital Pin Output Driving Strength

A digital pin's output driving strength is programmable as follows.

Table 7: Output driving strength of a digital pin.

Programming value	Maximum driving current (mA)
00	2
01	6
10	12
11	24

3 Electrical Specifications

3.1 DC Specifications

3.1.1 Absolute Maximum Ratings

The functionality of SC8800 is subject to the absolute maximum/minimum values listed in Table 8. Do not exceed these parameters or the part may be damaged permanently. Operation at absolute maximum ratings is not guaranteed.

Table 8: Absolute maximum ratings of SC8800

Symbol	Parameter	Min	Max	Unit
VBAT	Supply voltage from a battery	3.3	4.2	V
V_{DI}	Input voltage on any digital input	-0.5	3.6	V
V_{AI}	Input voltage on any analog input	-0.5	3.3	V
I_{AI2}	DC drive current for EARP, EARN, AUXSPP and AUXSPN		60	mA
$V_{max, ESD}$	Maxium ESD stress voltage, Huamn Body Model, any pin to any supply pin, either polarity or any pin to all non-supply pins together, either polarity. Three stresses maximum.		2,000	V
$I_{max, DC}$	Maximum DC Input current for any non-supply pin		5	mA
T_a	Ambient temperature	-45	+95	°C
$T_{storage}$	Storage temperature	-65	+125	°C

3.1.2 Recommended Operating Conditions

SC8800 is recommended to operate under the conditions list in Table 9.

Table 9: Recommended operating conditions.

Symbol	Parameter	Min	Typical	Max	Unit
VBAT	Supply voltage from a battery	3.3		4.2	V
$T_{junction}$	Junction temperature	-20		+125	°C
$T_{ambient}$	Ambient operating temperature	-30		+70	°C

3.1.3 Thermal Characteristics

The thermal characteristics are as shown in Table 10.

Table 10: Thermal characteristics.

Symbol	Parameter	Condition	Value	Unit
Theta JA	Junction-to-Ambient thermal resistance	Air flow: 0 m/sec	40	DegC/watt

3.1.4 DC Characteristics

SC8800 is fabricated with a 0.18 µm CMOS technology. The typical core voltage (VDD) is 1.8 V and the I/O supply (VDDIO) is typically at 3 V. The analog circuits are typically powered at 2.8 V. The core and analog power supplies are provided by the on-chip LDOs. The external battery can be connected directly to pins VBATA and VBAT. If not specified, VBAT means both. The power pins should be connected with a decoupling capacitor to ground (VSS, VSSIO).

For the following table, T_{amb} = -40 to +85 °C, VSS = 0 V (ground), VBAT = 3.6 V, and all voltages are measured with respect to VSS, unless otherwise specified.

Table 11: DC characteristics.

Symbol	Parameter	Conditions	Min	Typical	Max	Unit
$I_{standby\ (tot)}$	Total standby current		–	1.0	–	mA
P_{avg}	Average active power consumption	VDD = 1.8 V AVDD = 2.8 V	–	100	–	mW
Digital supply voltage: pins VDDIO						
VDDIO	Digital supply voltage		2.7	3.0	3.3	V
Digital supply voltage: internal VDD						
VDD	Digital core supply voltage		1.6	1.8	2.0	V
Digital input						
V_{IL}	Input voltage LOW-level		0	–	0.3 Vpad	V
V_{IH}	Input voltage HIGH-level		0.7 Vpad	–	Vpad	V
I_{LI}	Input leakage current		–	±2	–	µA
Digital output						
V_{OL}	Output voltage LOW-level	At I_{sink} = 2,4,6,10 mA (programmable)	–	–	0.3 Vpad	V
V_{OH}	Output voltage HIGH-level	At I_{source} = 2,4,6,10 mA (programmable)	0.7 Vpad	–	–	V
Analog supply voltage: pins VBATA						
VBATA	Analog supply voltage		3.3	3.6	4.2	V
I_{VBATA}	Analog supply current	AVDD = 2.8 V All analog blocks are active, but no load at EARP, EARN, AUXSPP and AUXSPN	–	5	–	mA
Analog supply voltage: internal AVDD						
AVDD	Analog core supply voltage		2.5	2.8	3.1	V
RTC supply: pin RTCVDD						
RTCVDD	Supply voltage			1.8		V
I_{RTCVDD}	Supply current			20		µA

Note:

- Vpad means the power supply voltage at the corresponding pad.

3.2 AC Specifications

A pin's AC characteristics include input and output capacitance, which determine loading for external drivers or other load analysis. The AC characteristics also include a de-rating factor, which indicates how much faster or slower the AC timings get with different loads.

Table 12: Standard input, output and I/O pin AC characteristics.

Parameters	Symbol	Minimum	Typical	Maximum	Units
Input capacitance, all standard input and IO pins	C_{in}			3.5	pF
Output capacitance, all standard output and IO pins	C_{load}			30	pF
Output de-rating falling edge on all standard output and I/O pins, from 30 pF load	T_{dr}		0.166		ns/pF

Note:

1. The AC specifications are tested with a 30 pF load as indicated in Figure 3:

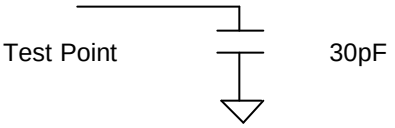


Figure 3: Test circuit of an I/O pin.

2. The output capacitance and de-rating falling edge are measured under the condition of maximum driving strength: 24 mA @ 3 V.

For the following tables, T_{amb} = -40 to +85 °C, VSS = 0 V (ground), VBAT = 3.6 V, and all voltages are measured with respect to VSS, unless otherwise specified.

Table 13: AC characteristics.

Symbol	Parameter	Conditions	Min	Typical	Max	Unit
f_{mclk}	Master clock frequency		–	13 or 26	–	MHz
f_{rtcclk}	Real-time clock frequency		–	32.768	–	kHz
Digital input						
C_I	Input capacitance		–	3	–	pF
Digital output						
T_R	Output rise time	Output load = 20 pF	2	–	5	ns
T_F	Output fall time	Output load = 20 pF	2	–	5	ns
Master clock input						
V_{mclk}	Master clock amplitude	AC coupling		–	0.9	V

Symbol	Parameter	Conditions	Min	Typical	Max	Unit
δ_{mclk}	Master clock duty cycle		40	–	60	%

3.2.1 Master Clock MCLKI Timing

The master clock is input from pin MCLKI.

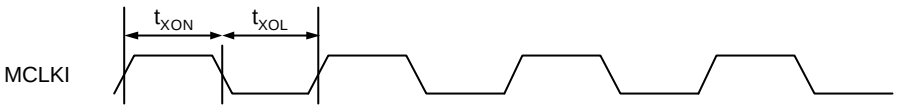


Figure 4: MCLKI Timing Parameters.

Table 14: MCLKI Timing Parameters.

Symbol	Parameter	Min	Typical	Max	Unit
t_{XOH}	MCLKI logic high	–	38.46	–	ns
t_{XOL}	MCLKI logic low	–	38.46	–	ns

3.2.2 SDRAM Interface Timing

Table 15: SDRAM interface AC specifications

Description	Symbol	Min	Max	Units
CLK DMEM period	clk	20		ns
Control signals setup time to CLK DMEM rise	See note	7 + $T_{clkdelay}$		ns
Control signals hold time to CLK DMEM rise		1		ns
EMD write data input setup time from CLK DMEM rise	Tds	3		ns
EMD read data input hold time from CLK DMEM rise	Tdh	1		ns

Note:

- $T_{clkdelay}$ is the value set in the delay control register.

3.2.3 SRAM/Flash

Table 16: SRAM/Flash interface asynchronous AC timing

Parameter	Description	Min	Unit
T_{rd_pulse}	Read cycle time	$N*T_{cycle}$ - T_{oe_delay} (note 1)	ns
T_{wr_pulse}	Write Cycle time	$N*T_{cycle}$ - T_{we_delay} (note 2)	
T_{b_delay}	EMBH/EMBL delay time	1 + N (note 3)	
T_{adr_delay}	Address delay time	1 (note 3)	

Parameter	Description	Min	Unit
T_{oe_delay}	EMOEN delay time	1 + N (note 3)	
T_{we_delay}	EMWEN delay time	1 + N (note 3)	
T_{cs_delay}	EMCSN delay time	1 + N (note 3)	
T_{cs_hd}	EMCSN hold time	1	ns
T_{we_hd}	EMWEN hold time	1	ns
T_{oe_hd}	EMOEN hold time	1	ns
T_{adr_hd}	Address hold time	T_{cycle}*Hold_set + 1 (see note 4)	
T_{dat_hd}	Data hold time	T_{cycle}*Hold_set + 1	
T_{we_hd}	WE hold time	1	ns

Note:

1. Tcycle means the period of one cycle of ARM clock, N represents the value of read width that has been set in global registers.
2. N represents the value of write-width that has been set in global registers.
3. N respectively represents the value of delay cell in global register.
4. Hold_set represents if the hold enable bit is set in SRAM/Flash controller register.

Write operation:

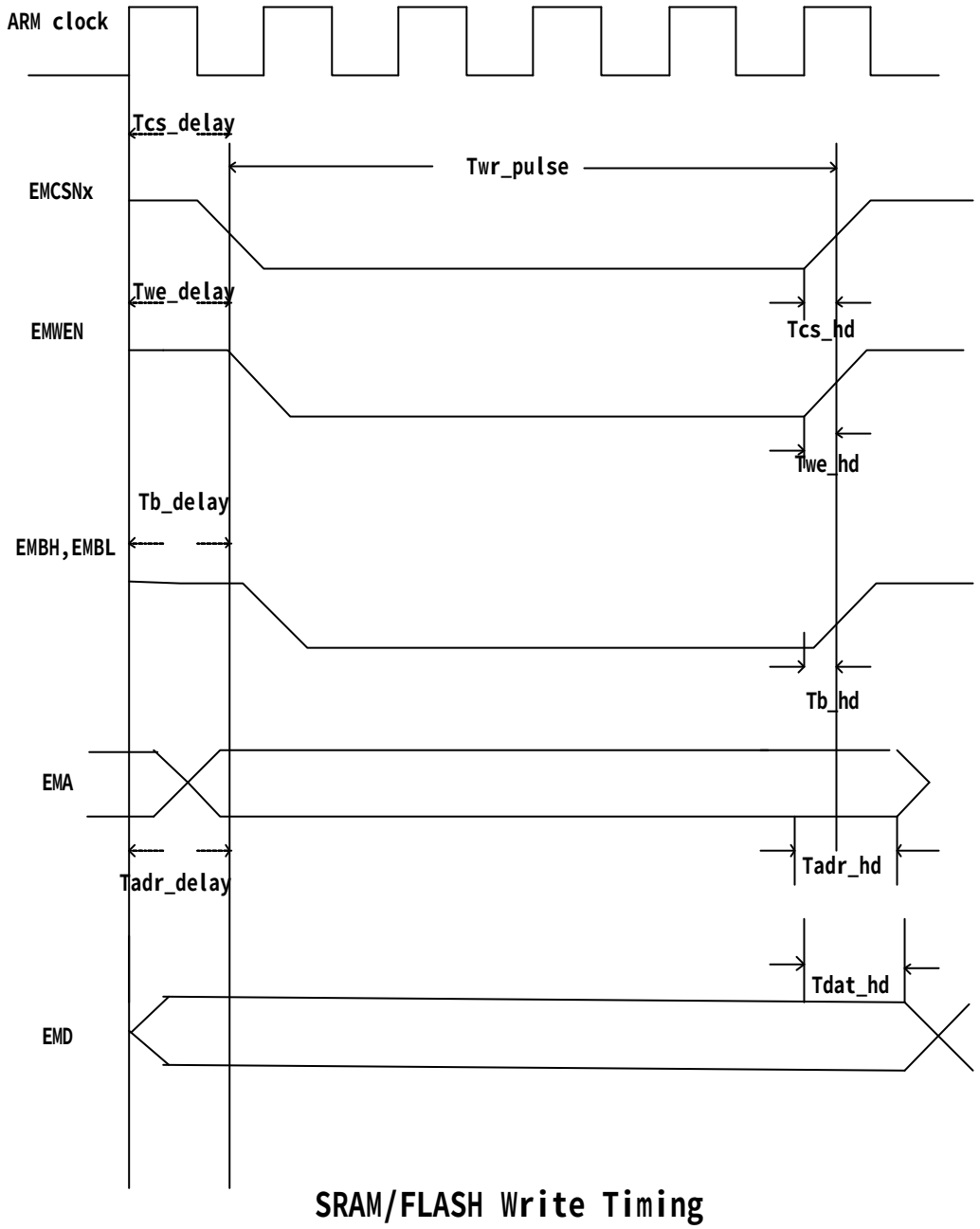


Figure 5: SRAM/FLASH write timing

Read operation

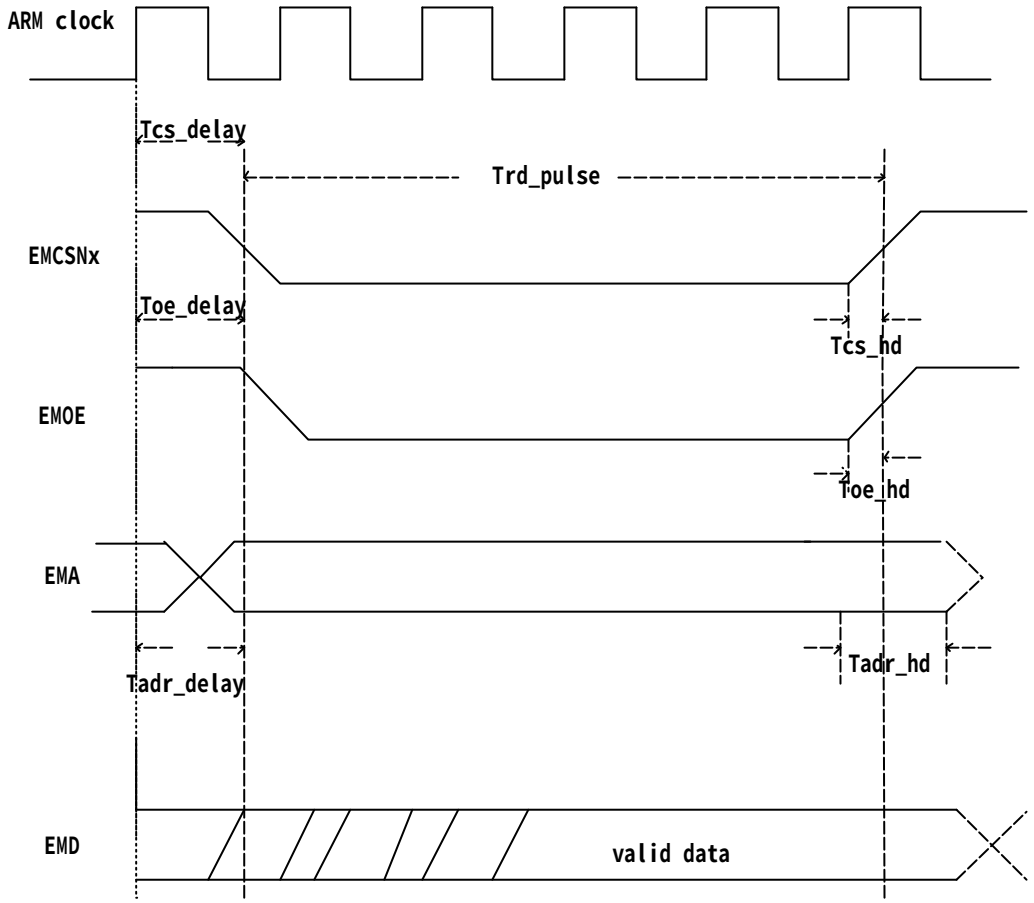


Figure 6: SRAM/FLASH read timing

3.2.4 UART Timing

The following is the UART timing diagram.

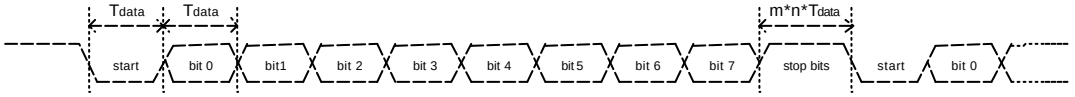


Figure 7: UART timing

Table 17: UART Timing Parameter

Symbol	Parameter	Min	Max	Unit
T_{data}	Bit Width	T	–	ns

Note:

- T: 1/(ARM clock)
- m, n: register programmable integer

The UART port can support up to 960 Kbit/sec data rate.

3.2.5 SPI Timing

The following is the SPI timing diagram.

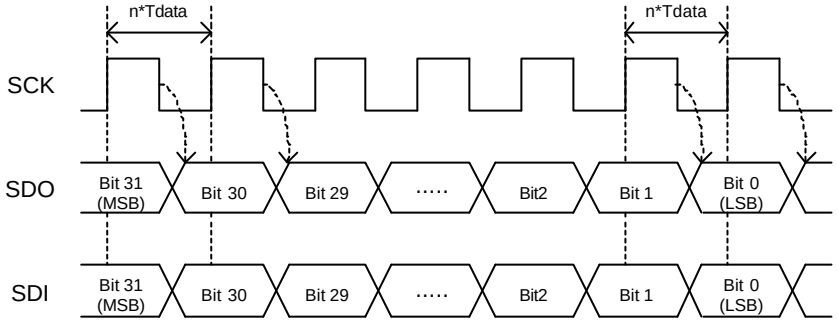


Figure 8: SPI timing

Table 18: SPI Timing Parameter

Symbol	Parameter	Min	Max	Unit
T_{data}	Bit Width	T	–	ns

Note:

- T: 1/(ARM clock)
- n: register programmable integer, n = 4, 5, 6, 7 ….

SPI in SC8800 can support up to ARM_CLK/4 data rate.

3.2.6 I2C Timing

The following are the I2C read and write waveforms. The data rate of I2C can be up to 400 Kbit/sec.

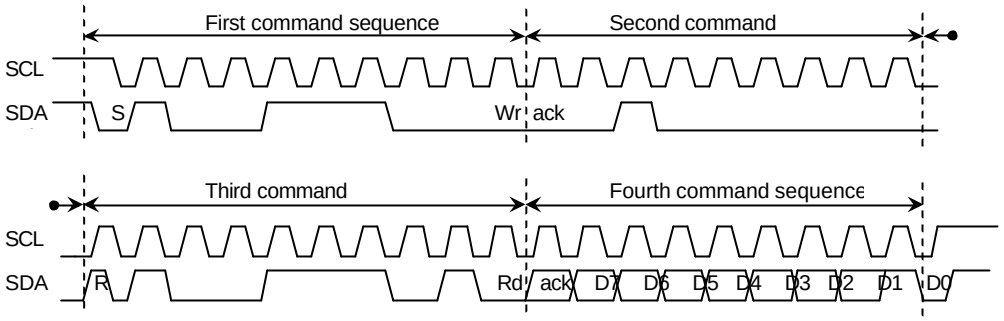


Figure 9: I2C timing

3.2.7 Three-Wire Serial Interface Timing

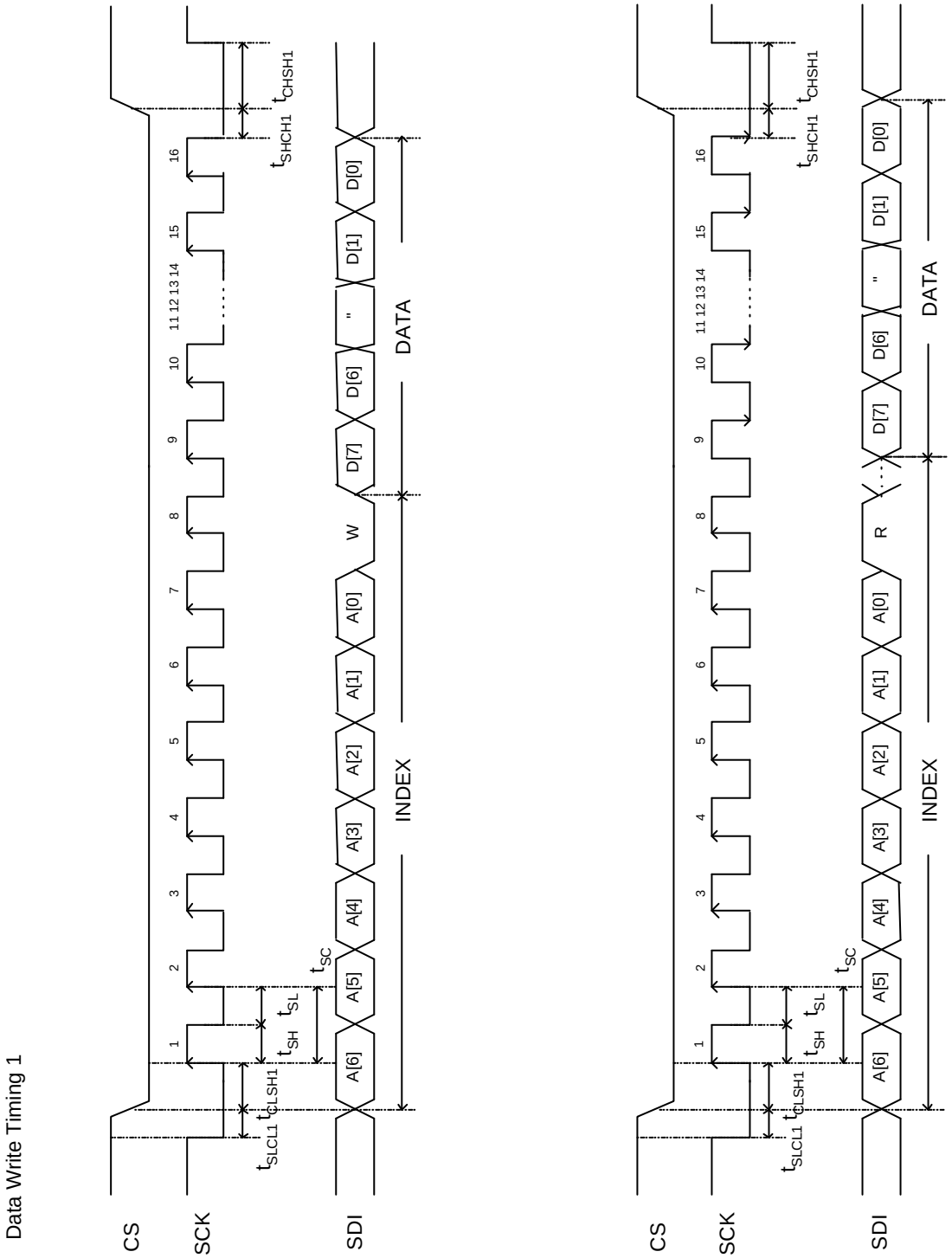


Figure 10: Three-wire serial interface timing parameters (mode 1).

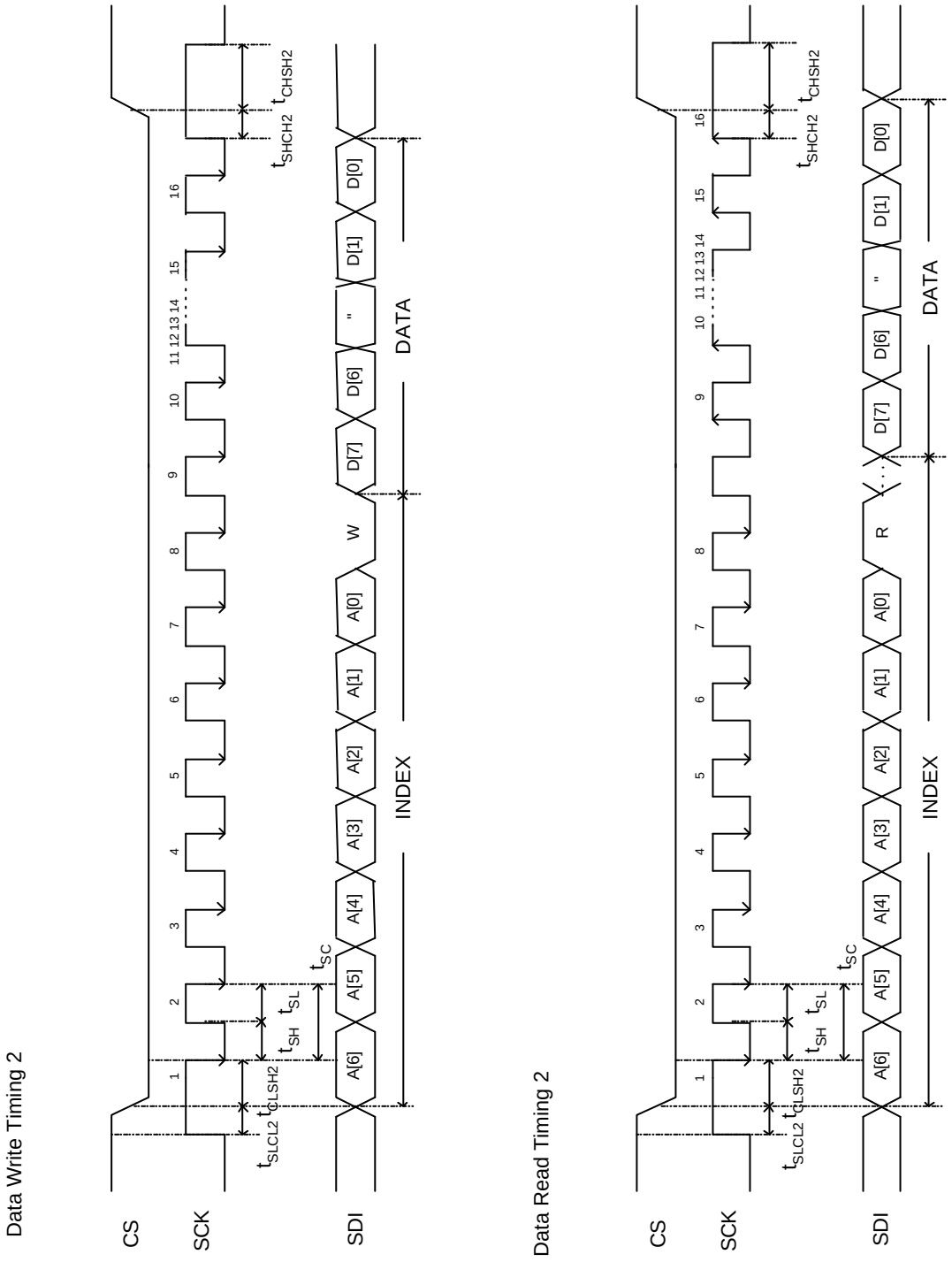


Figure 11: Three-wire serial interface timing parameters (mode 2).

Table 19: Three-wire serial interface timing parameters

Symbol	Parameter	Min	Max	Unit
t_{CLSL1}	Chip select low to SCK low	T	–	ns
t_{CLSH1}	Chip select low to SCK high	T	–	ns
t_{CLSL2}	Chip select low to SCK low	T	–	ns
t_{CLSH2}	Chip select low to SCK high	T	–	ns
t_{SH}	SCK high pulse width	N*T/2	–	ns
t_{SL}	SCK low pulse width	N*T/2	–	ns
t_{SC}	SCK cycle time	N*T	–	ns
t_{SHCH1}	SCK high to chip select high	T	–	ns
t_{CHSH1}	Chip select high to SCK high	T	–	ns
t_{SHCH2}	SCK high to chip select high	T	–	ns
t_{CHSH2}	Chip select high to SCK high	T	–	ns

Note:

1. T: the MCU clock cycle time
2. N: register programmable

3.2.8 JTAG Timing

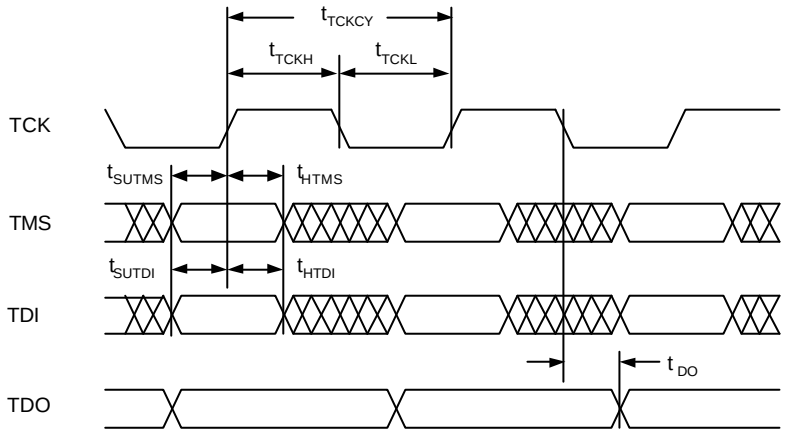


Figure 12: JTAG Interface Timing

Table 20: JTAG Interface Timing

Symbol	Parameter	Min	Typical	Max	Units
t_{TCKCY}	TCK period	?	–	–	ns
t_{TCKH}	TCK pulse width high	?	–	–	ns
t_{TCKL}	TCK pulse width low	?	–	–	ns
t_{SUTMS}	TMS Input Set-up Time	?	–	–	ns
t_{HTMS}	TMS Input Hold Time	?	–	–	ns
t_{SUTDI}	TDI Input Set-up Time	?	–	–	ns
t_{HTDI}	TDI Input Hold Time	?	–	–	ns
t_{DO}	TDO Data Output Delay	–	–	?	ns

3.3 Performance Specifications

This section specifies the performance of some major SC8800 modules.

3.3.1 TD-SCDMA Rx ADC

Table 21: Rx ADC AC characteristics

Parameters	Symbol	Note	Min	Typical	Max	Unit
Input Span			0.6	1.5		Vpp
Input Signal Range			0		2.8	V
Input Common Mode Voltage				1.45		
3 dB Cutoff Frequency (LPF)				2		MHz

3.3.2 TD-SCDMA Tx DAC

Table 22: Tx DAC AC characteristics

Parameters	Symbol	Note	Min	Typical	Max	Unit
Output Common Mode Voltage				1.45		V
Differential Output Voltage				400 600 800 1000		mVpp
3dB Cutoff Frequency (LPF)				1.2MHz		MHz
Filter Attenuation @1M (LPF)				1		dB
Filter Attenuation @1.2MHz (LPF)				3		dB
Filter Attenuation @5.12MHz (LPF)				28		dB
Passband Gain Flatness (LPF)				0.5		dB

3.3.3 GSM Baseband

3.3.3.1 Baseband Transmit Path

The outputs of GMSK modulator are converted to analog levels by 10-bit D/A Converters (DAC), and then passed to an analog Butterworth low pass filter. Each analog part has special offset cancellation.

Table 23: Performance of the baseband transmit path.

Parameter	Conditions/Comments	Min	Typical	Max	Unit
Transmission buffer length			160		bits
GMSK modulator					
Input data word rate			270.833		kHz
I/Q interpolation output size			10		bits
DAC					
Number of Channels			2		
Resolution			10		bits
GMSK phase trajectory error					

RMS error	Including analog filter			1.5	°
Peak error				7	°
LP analog filter					
Cutoff frequency (3 dB)			150 ± 25%		kHz
Group delay				9	µsec
I/Q mismatch					
I/Q amplitude mismatch				0.2	dB
I/Q phase mismatch				0.1	°
Maximum differential output voltage	TX_B [1:0] = 00		±0.6		V
	TX_B [1:0] = 10		±0.9		
	TX_B [1:0] = 01		±1.2		
	TX_B [1:0] = 11		±1.5		
Output common mode voltage			1.3		V
DC offset				10	mV
Minimum load resistance					
Single-ended		100			kΩ
Differential		50			kΩ
Maximum load Capacitance					
Single-ended				50	pF
Differential				50	pF
Relative output power	Complies with GSM spectral mask				
	0 – 100 kHz	-3		0	dB
	200 kHz			-32	
	250 kHz			-35	
	400 kHz			-63	
	600 kHz			-71	
	1200 kHz			-71	
	> 1800 kHz			-71	

3.3.3.2 Baseband Receive Path

The differential I/Q signal from the RF module is first amplified by a programmable gain amplifier (PGA). After the amplification, the signal is modulated to single-bit words by S-? modulator at a frequency much higher than the Nyquist rate. Then the signal passes through a digital filter. Each analog part has special offset cancellation.

Table 24: Performance of the baseband receive path.

Parameter	Conditions/Comments	Min	Typical	Max	Unit
PGA					
Input common mode signal range	2 dB steps controlled by 4 digital bits	0.6		1.8	V
Input gain control range		0		18	dB
Gain resolution			2		dB
Gain control linearity				±1	dB
Maximum analog input range after PGA	Differential			±1.8	V
Input sampling frequency			6.5		MHz
Analog input antialiasing LPF					
3-dB single sided bandwidth			500		kHz

Parameter	Conditions/Comments	Min	Typical	Max	Unit
Output data					
Word rate			270.833		kHz
Dynamic range		66			dB
Signal to (noise + distortion) ratio			60		dB
Gain mismatch between I/Q channels				±0.15	dB
Absolute group delay			20		µs
Group delay between I/Q channels (0 kHz – 100 kHz)				5	ns

3.3.4 Voice Band

3.3.4.1 Voice Band Uplink ADC

The performance of the voice band uplink ADC is as shown in Table 25.

Table 25: Performance of the voice band ADC

Parameter	Conditions/Comments	Min	Typical	Max	Units
Input common mode range		0.6		1.8	V
PGA gain range		0		33	dB
PGA gain step size			3		dB
Maximum analog input range after PGA	Differential			±1.5	V
Digital audio output sample rate			8		kHz
ADC resolution		13			bits
ADC signal to (noise + distortion) ratio			62		dB
Digital output word			16		bits

3.3.4.2 Voice Band Downlink DAC

The performance of the voice band downlink DAC is as shown in Table 26.

Table 26: Performance of the voice band DAC

Parameter	Conditions/Comments	Min	Typical	Max	Units
Digital audio input sample rate			8		kHz
Input digital word			16		bits
DAC resolution			13		bits
DAC signal to noise plus distortion ratio			62		dB
Analog audio output					
Low stop band rejection (< 100 Hz)		20			dB
Low passband edge (0.5 dB down from passband level)		120		150	Hz
High pass band edge (0.5 dB down from passband level)		3.4		3.7	kHz
High stop band rejection (> 4500 Hz)		35			dB
Volume Control (for each of normal and auxiliary output)					
Gain range		-6		27	dB

Parameter	Conditions/Comments	Min	Typical	Max	Units
Gain step size			3		dB
Side tone amplifier					
Gain range		-24		-6	dB
Gain step size			6		dB
Minimum load resistance					
Single ended		16			Ω
Differential		16			Ω
Output amplifier					
Maximum differential output voltage			±1.4		V
Maximum single-ended output voltage			1.4		V
Output common mode voltage Earpiece mute switch attenuation		 -40	1.1		V dB

3.3.5 Phase-Locked Loop (PLL)

The phase-locked loop (PLL) is used for frequency synthesizers to generate different clocks. The VCO clock is divided by a programmable "M", and the divided clock is phase-locked to an external reference clock divided by a programmable "N". The result is the output clock of the PLL at a frequency M/N times the reference clock frequency. The values for M and N can each range from 2 to 256.

fout = fin * PLLMN[11:0] / PLLMN[23:16]

Table 27: Performance of the phase-locked loop (PLL only)

Parameter	Conditions/Comments	Min	Typical	Max	Unit
Input reference			13		MHz
Frequency Range		92	300	400	MHz
Long term jitter	All other circuits on			1	ns/s
Short term jitter	All other circuits on			150	ps/cycle
Settling time	External filter capacitor 500 pF		100	200	µs
Sleep mode	Yes				

3.3.6 RTC Oscillator

SC8800 contains a RTC oscillator, for a specific crystal at 32.768 KHz. The 32.768 KHz crystal is connected with RTC32KI and RTC32KO.

Table 28: Performance of the 32.768-KHz oscillator

Description	Symbol	Min	Typical	Max	Units
Crystal Frequency			32.768		KHz
Equivalent series resistance Drive Level		6		65 1	KOhm uW
Internal power supply	VRTC		1.8		V
Input High Voltage RTC32XI		0.8 VRTC		VRTC	V
Input Low Voltage RTC32XI		0		0.2 VRTC	V
Input Leakage, RTC32XI				1	uA
Input Capacitance RTC32XI/RTC32XO			18	25	pF
Amplifier Stabilization Time		2		10	s

Parasitic Resistance RTC32XI/RTC32XO to any node		20			MOhm
Parasitic Capacitance, RTC32XI/RTC32XO			5		pF

3.3.7 Automatic Frequency Control (AFC) DAC

Parts of the AFC are a 10-bit D/A converter and a programmable gain output driver. Special offset cancellation is applied in the AFC DAC.

Table 29: Performance of the AFC DAC.

Parameter	Conditions/Comments	Min	Typical	Max	Unit
Resolution			10		bits
Integral non-linearity			±3		LSB
Differential non-linearity			±2		LSB
Settling time			4		µs
Gain range		1.5		2	x
Minimum output voltage			0.2		V
Maximum output voltage	Gain = 1.5		1.7		V
Maximum output voltage	Gain = 2		2.1		V
Minimum load resistance			50		kΩ
Maximum load capacitance			50		pF

3.3.8 AGC DAC

Table 30: Performance of the AGC DAC

Parameter	Conditions/Comments	Min	Typical	Max	Units
Resolution			10		bits
Integral non-linearity			±3		LSB
Differential non-linearity			±2		LSB
Settling time			4		µs
Gain range		1.5		2	x
Minimum output voltage			0.2		V
Maximum output voltage	Gain = 1.5		1.7		V
Maximum output voltage	Gain = 2		2.1		V
Minimum load resistance			50		kΩ
Maximum load capacitance			50		PF

3.3.9 RF Power Ramping Control DAC

The RF power amplifier control (APC1 and APC2) consists of a 10-bit D/A converter and a programmable-gain output driver. Special offset cancellation is applied in the DAC.

Table 31: Performance of the RF power ramping control DAC

Parameter	Conditions/Comments	Min	Typical	Max	Units
Resolution			10		bits
Integral non-linearity			±3		LSB

Parameter	Conditions/Comments	Min	Typical	Max	Units
Differential non-linearity			±2		LSB
Settling time			4		µs
Gain range		1.5		2	x
Minimum output voltage			0.2		V
Maximum output voltage	Gain = 1.5		1.7		V
Maximum output voltage	Gain = 2		2.1		V
Minimum load resistance			50		kΩ
Maximum load capacitance			50		PF

3.3.10 Auxiliary Analog-to-Digital Converter (ADC)

The auxiliary ADC is a 10-bit successive approximation A/D converter. The ADC has five input channels. Special offset cancellation is applied in the ADC.

Table 32: Performance of the auxiliary ADC

Parameter	Conditions/Comments	Min	Typical	Max	Unit
Resolution			10		bits
Integral non-linearity			±3		LSB
Differential non-linearity			±2		LSB
Number of input channels			5		
Analog input range		0.1		1.3	V
Conversion time				15	µs
Power up settling time				10	µs

3.3.11 LDO

SC8800 integrated low-voltage and low quiescent current low dropout regulators (LDO) for power supply and management purposes. Some extra LDO regulators are designed for external use. All LDOs have their own bypass control signals. External tantalum or MLCC ceramic capacitors are recommended to use with the LDOs.

Table 33: Performance of the integrated LDO regulators

Parameter	Conditions/Comments	Min	Typical	Max	Unit
LDO1 (RF)					
Output voltage			2.8		V
Output current				100	mA
Output capacitor	Required for stability and ripple rejection	10		50	µF
LDO2 (LCM)					
Output voltage			2.8		V
Output current				50	mA
Output capacitor	Required for stability and ripple rejection	1		50	µF
LDO_VB	Internal use only				
Output voltage			2.8		V
Output current				10	mA
Output capacitor	Required for stability and	1		50	µF

Parameter	Conditions/Comments	Min	Typical	Max	Unit
	ripple rejection				
LDO_VBO					
Output voltage			2.8		V
Output current				70	mA
Output capacitor	Required for stability and ripple rejection	10		50	µF
LDO_BB	Internal use only				
Output voltage			2.8		V
Output current				10	mA
Output capacitor	Required for stability and ripple rejection	1		50	µF
LDO_AUX	Internal use only				
Output voltage			2.8		V
Output current				10	mA
Output capacitor	Required for stability and ripple rejection	1		50	µF
LDO_PLL	Internal use only				
Output voltage			1.8/1.7/1.6/1.5		V
Output current				10	mA
Output capacitor	Required for stability and ripple rejection	1		50	µF
LDO_RTC	Internal use only				
Output voltage			1.8/1.7/1.6/1.5		V
Output current				10	mA
Output capacitor	Required for stability and ripple rejection	1		50	µF
LDO for digital core					
Output voltage			1.8/1.7/1.6/1.5		V
Output current				160	mA
Output capacitor	Required for stability and ripple rejection	10		50	µF
LDO for VDDIO					
Output voltage			3.0/2.8/2.07/1.8		V
Output current				100	mA
Output capacitor	Required for stability and ripple rejection	10		50	µF
LDO for VMEM					
Output voltage Output current			3.0/2.8/2.07/1.8	 110	V mA
Output capacitor	Required for stability and ripple rejection	10		50	µF
LDO for VSDRAM Output voltage			3.0/2.8/2.07/1.8		V
Output current				100	mA
Output capacitor **LDO for LCD**	Required for stability and ripple rejection	10		50	µF
Output voltage			3.0/2.8/2.07/1.8		V
Output current				100	mA
Output capacitor	Required for stability and ripple rejection	1		50	µF

Parameter	Conditions/Comments	Min	Typical	Max	Unit
LDO for VSIM					
Output voltage			3.0/2.8/2.07/1.8		V
Output current				10	mA
Output capacitor	Required for stability and ripple rejection	1		50	µF

4 Functional Description

This section provides detailed information about SC8800 functions, including description, control registers and software interface.

SC8800 has two general-purpose processors, ARM926EJ MCU and Teak DSP. In general, the MCU and DSP control the operation of various ASIC hardware modules. The MCU or DSP prepares the data that needs the ASIC to process and sets up some control registers that tells the ASIC how to process the data. Therefore, the software control interface consists of data memories and control registers. The software interface sections define the organization of the data memories and the meanings of control register bits.

4.1 ARM Memory Map

On the ARM side, the address bus is 32-bit wide and the address map is presented in this section. In order to support multiple boot modes, the ARM memory space can be remapped after reset.

Table 34: SC8800 ARM memory address map after reset

Address Range	Memory Sector	Sub-Sector
0x0000_0000 – 0x0FFF_FFFF	External Flash/SRAM Space (default)	This Address space can be remapped, which is controlled by remap control register Remap[1:0]: 0x00: default to external flash/sram. 0x01: swap address space with internal rom. 0x10: swap address space with external sdram. **0x11: default to external flash/sram.** SMCS0: 0x0000_0000 – 0x03FF_FFFF SMCS1: 0x0400_0000 – 0x07FF_FFFF SMCS2: 0x0800_0000 – 0x0BFF_FFFF SMCS3: 0x0C00_0000 – 0x0FFF_FFFF
0x1000_0000 – 0x1FFF_FFFF	Internal Shared Memory Space	0x1000_0000 – 0x1000_0FFF is used. 2048*16
0x2000_0000 – 0x2FFF_FFFF	Internal ROM Space (default)	This Address space can be remapped, which is controlled by remap control register Remap[1:0]: 0x00: default to Internal rom. 0x01: swap address space with external flash/sram. 0x10: default to Internal rom. 0x11: default to Internal rom. 0x2000_0000 – 0x2000_0FFF is used. 1024*32
0x3000_0000 – 0x3FFF_FFFF	**Internal RAM space**	0x3000_0000 – 0x3000_3FFF is used. 4096*32
0x4000_0000 – 0x1FFF_FFFF	ARM External SDRAM Memory Memory Space	This Address space can be remapped, which is controlled by remap control register Remap[1:0]:

Address Range	Memory Sector	Sub-Sector
		0x00: default to external sdram. 0x01: default to external sdram. 0x10: swap address space with external flash/sram. **0x11: default to external sdram.** DMCS0: 0x4000_0000 – 0x43FF_FFFF DMCS1: 0x4400_0000 – 0x47FF_FFFF DMCS2: 0x4800_0000 – 0x4BFF_FFFF DMCS3: 0x4C00_0000 – 0x4FFF_FFFF
0x7000_0000 – 0x7FFF_FFFF	AHB Control Reg space :	0x7000_0000: ARM9 Flash or SRAM External Memory Registers. 0x7010_0000: Interrupt Control Registers. 0x7020_0000: DMA Control Registers. 0x7030_0000: ARM9 SDRAM Control Registers.
0x8000_0000 – 0x8FFF_FFFF	Peripheral Address Space	
	0x8000_0000	System Watch Dog
	0x8100_0000	Timer/Counter
	0x8200_0000	RemPause
	0x8200_3000	VB
	0x8300_0000	Serial0
	0x8400_0000	Serial1
	0x8500_0000	SIM
	0x8600_0000	I2C
	0x8700_0000	KPD
	0x8800_0000	PWM
	0x8900_0000	RTC
	0x8A00_0000	GPIO
	0x8B00_0000	GLB
	0x8C00_0000	CHIPPIN
	0x8D00_0000	GEA
	0x8E00_0000	Serial2
	0x8F00_0000	RTC Timer/Counter

It should be noted that:

1. The re-map (address space swapping) among the external flash/sram, sdram and internal ROM is controlled by strapping values on power up to remap [1:0] or write value to remap [1:0] when system has powered up.
2. For internal shared memory, only 16-bit write and 8- or 16-bit read are supported. All Other memory accesses support byte, half word and word write/read.
3. Page mode and Burst mode are supported for the flash memory read.

4.2 External Flash or SRAM Memory Interface

4.2.1 FLASH Interface Features

- Only support 16-bit-width FLASH.
- Support up to 1G Bytes FLASH, that is, up to 29 memory address pins, and HADDR [29:0].
- Support up to 4 CS.
- Support optional asynchronous normal-mode read/write operation.
- Support optional asynchronous 4/8/16/32 page-mode read operation.
- Support optional synchronous 4/8/16/32/continuous burst-mode read operation.
- Support optional wrap or increase access in burst mode. We support real wrap mode burst. That is, one wrap burst is implemented by one external memory burst stead of divided into two bursts.
- Support asynchronous normal-mode read operation without ADV-latched address.
- Support asynchronous normal-mode read operation with ADV-latched address.
- Support asynchronous page-mode read operation without ADV-latched address.
- Support asynchronous page-mode read operation with ADV-latched address.
- Support synchronous single-mode read operation without ADV-latched address.
- Support synchronous single-mode read operation with ADV-latched address.
- Support synchronous burst-mode read operation with address latched by CLK.
- Not support synchronous burst-mode read operation with address latched by ADV.
- In burst mode, support optional WAIT/RDY control. If WAIT/RDY is used, it acts as a valid signal for read data. Otherwise, fixed wait time is used. It's important that, in other modes, memory controller always uses fixed wait time even if WAIT/RDY is enabled, and in continuous burst mode and increase 4/8/16/32 burst mode, WAIT/RDY must be used, and only in wrap 4/8/16/32 burst mode, two ways are selected by software.
- In burst mode, support an optional WAIT/RDY active watchdog to prevent system from dead lock by WAIT/RDY.
- In burst mode, the active polarity of WAIT/RDY is configurable.
- In burst mode, only support WAIT/RDY active one cycle before data. Not support active with data.
- In burst mode, support rising edge, falling edge or configurable of CLK.
- In burst mode, first access latency is configurable from 1 to 7.
- In burst mode, only support one data hold cycle. Not support two data hold cycles.
- In burst mode, only support linear mode. Not support interleave mode.
- In burst mode, 4/8/16/32/continuous mode can be selected by software.
- In 4/8/16/32 burst mode, wrap mode and increase mode can be select by software.
- Not support burst suspend.
- Support asynchronous write operation without ADV-latched address.
- Support asynchronous write operation with ADV-latched address.
- Support WP output controlled by software.
- The idle polarity of CLK for each CS can be configured independently.
- Support configurable read data valid time under normal mode.
- Support configurable first read data valid time under page mode.
- Support configurable following read data valid time under page mode.
- Support configurable first read data valid time under burst mode.
- Support configurable following read data valid time under burst mode.
- Support configurable read hold time.
- Support configurable write pulse width.
- Support configurable write hold time.
- Support configurable read-to-read turn around time.
- Support configurable read-to-write turn around time.
- Support configurable write-to-read turn around time.
- Support configurable write-to-write turn around time.
- Memory clock is optional HCLK or HCLK/2.

- Memory clock can be selected between clk_dmem_out and clk_smem_out. Clk_dmem_out is HCLK and clk_smem_out is HCLK or HCLK/2. That is, we support that FLASH uses HCLK/2 and pSRAM uses HCLK.
- Support dynamic changing memory controller configuration register and FLASH/pSRAM configuration register when program is running in FLASH and pSRAM.

4.2.2 SRAM Interface Features

- Only support 16-bit-width SRAM.
- Support up to 1G Bytes SRAM, that is, up to 29 memory address pins, and HADDR [29:0].
- Support up to 4 CS.
- Support asynchronous read operation.
- Support asynchronous write operation.
- Configurable timing parameters are the same as FLASH.

4.2.3 pSRAM Interface Features

- Device List:
  - ✓ CellularRAM, synchronous burst and asynchronous, provided by Micron, Spansion (AMD), Infineon, etc.
  - ✓ CosmoRAM, rev 2, synchronous burst and asynchronous, provided by Toshiba, NEC, Spansion (Fujitsu), etc.
  - ✓ UtRAM, synchronous burst and asynchronous, provided by Samsung, Hynix, etc.
  - ✓ Spansion synchronous burst pSRAM type 4.
  - ✓ Spansion asynchronous pSRAM from type 1 to type 7.
  - ✓ CosmoRAM, rev 3, Address/Data Multiplexed pSRAM, provided by NEC.
  - ✓ Renesas and Cypress??? No data sheet for CellularRAM.
- Only support 16-bit-width pSRAM.
- Support up to 1G Bytes pSRAM, that is, up to 29 memory address pins, and HADDR [29:0].
- Support up to 4 CS.
- Support asynchronous normal-mode read/write operation.
- Support asynchronous 4/8/16/32 page-mode read operation.
- Support asynchronous normal-mode write and synchronous burst-mode read.
- Support synchronous 4/8/16/32/128/256/512/continuous burst-mode read/write operation. In these burst length, 128/256/512 is for full-page burst mode in Samsung UtRAM and Spansion pSRAM type 4.
- Support synchronous single-mode read/write operation.
- For CellularRAM, support Configuration Registers reads and writes by an optional output pin CRE. There are two ways to configure CR by software. One is that, software visits CR by reads and writes Memory Controller Control Register slave directly, and CRE is controlled by memory controller automatically, and CR loading data comes from HWDATA. The other is that, CRE is controlled just like a GPIO pin, and software can set/clear this pin directly, and then software can visit CR by reads and writes Memory Controller slave directly, and CR loading data comes from HADDR. Solution one is recommended and solution two is just a backup. On pSRAM bus, only asynchronous read/write is used to access Configuration Registers.
- The active polarity of CRE is configurable.
- Not support multiplexed address/data pSRAM.
- In the burst mode, initial access latency can be configured as variable or fixed. If using variable mode, WAIT/RDY must be used.
- In the burst mode, support separate read WAIT/RDY enable and write WAIT/RDY enable.
- Other features are the same as FLASH.

4.2.4 External Flash and SRAM Control Registers

Base Address: 0x7000_0000

Address	Signal	Bit Pos	Default	Description
0x0000	ExtMemCtl0	[31:0]	**Read/Write**	
		[0] [1]	0x1 0x1	Read to read turn around ctl Read to write turn around ctl.
		[2]	0x1	Write to read turn around ctl
		[3]	0x1	Write to write turn around ctl
		[7:4]	0x1	First read data valid time under burst/page mode.
		[11:8]	0x7	Read data valid time under normal mode
		[15:12]	0x7	Write pulse width
		[16]	0x1	Read hold time
		[17]	0x1	Write hold time
		[19:18	0x3	Second and following read data valid time under page mode.
		[21]	0x0	Page mode flag
		[22]	0x0	Burst mode flag
		[26]	0x0	flash
		[27]	0x1	Not use buffer in the ext mem ctl module
		[28]	0	Bus error indicator
		[29]	0x0	Write protection error
		[30]	0	Write protect
0x0004	ExtMemCtl1	[31:0]	**Read/Write**	Per bit definition and default value are the same as ExtMemCtl0
0x0008	ExtMemCtl2	[31:0]	**Read/Write**	Per bit definition and default value are the same as ExtMemCtl0
0x000C	ExtMemCtl3	[31:0]	**Read/Write**	Per bit definition and default value are the same as ExtMemCtl0

Note:

1. For all the ExtMemCtl registers, the bit field definition is exactly the same.
2. Correct programming of ExtMemCtl registers based on the system configuration is needed to make the system work properly.
3. ExtMemCtl0 tp ExtMemCtl3 correspond to EMCSN0 to EMCSN3 separately.
4. Please refer to flash/sram/sdram data sheet for more info on read write timing.

4.3 Bus Monitors Control Registers

Base Address: 0x7000_0000

Address	Signal	Bit Pos	Default	Description
0x0028	ADDR0	[31:0]	**Read/Write**	
0x002C	ADDR0_MSK	[31:0]	**Read/Write**	

0x0030	RDATA0	[31:0]	**Read/Write**	
0x0034	RDATA0_MSK	[31:0]	**Read/Write**	
0x0038	WDATA0	[31:0]	**Read/Write**	
0x003C	WDATA0_MSK	[31:0]	**Read/Write**	
0x0040	ADDR1	[31:0]	**Read/Write**	
0x0044	ADDR1_MSK	[31:0]	**Read/Write**	
0x0048	RDATA1	[31:0]	**Read/Write**	
0x004C	RDATA1_MSK	[31:0]	**Read/Write**	
0x0050	WDATA1	[31:0]	**Read/Write**	
0x0054	WDATA1_MSK	[31:0]	**Read/Write**	
0x0058	ADDR2	[31:0]	**Read/Write**	
0x005C	ADDR2_MSK	[31:0]	**Read/Write**	
0x0060	RDATA2	[31:0]	**Read/Write**	
0x0064	RDATA2_MSK	[31:0]	**Read/Write**	
0x0068	WDATA2	[31:0]	**Read/Write**	
0x006C	WDATA2_MSK	[31:0]	**Read/Write**	
0x0070	ADDR3	[31:0]	**Read/Write**	
0x0074	ADDR3_MSK	[31:0]	**Read/Write**	
0x0078	RDATA3	[31:0]	**Read/Write**	
0x007C	RDATA3_MSK	[31:0]	**Read/Write**	
0x0080	WDATA3	[31:0]	**Read/Write**	
0x0084	WDATA3_MSK	[31:0]	**Read/Write**	
0x0088	ADDR4	[31:0]	**Read/Write**	
0x008C	ADDR4_MSK	[31:0]	**Read/Write**	
0x0090	RDATA4	[31:0]	**Read/Write**	
0x0094	RDATA4_MSK	[31:0]	**Read/Write**	
0x0098	WDATA4	[31:0]	**Read/Write**	
0x009C	WDATA4_MSK	[31:0]	**Read/Write**	
0x00A0	ADDR5	[31:0]	**Read/Write**	
0x00A4	ADDR5_MSK	[31:0]	**Read/Write**	
0x00A8	RDATA5	[31:0]	**Read/Write**	
0x00AC	RDATA5_MSK	[31:0]	**Read/Write**	
0x00B0	WDATA5	[31:0]	**Read/Write**	
0x00B4	WDATA5_MSK	[31:0]	**Read/Write**	
0x00B8	ADDR6	[31:0]	**Read/Write**	
0x00BC	ADDR6_MSK	[31:0]	**Read/Write**	
0x00C0	RDATA6	[31:0]	**Read/Write**	
0x00C4	RDATA6_MSK	[31:0]	**Read/Write**	
0x00C8	WDATA6	[31:0]	**Read/Write**	
0x00CC	WDATA6_MSK	[31:0]	**Read/Write**	
0x00D0	ADDR7	[31:0]	**Read/Write**	
0x00D4	ADDR7_MSK	[31:0]	**Read/Write**	
0x00D8	RDATA7	[31:0]	**Read/Write**	
0x00DC	RDATA7_MSK	[31:0]	**Read/Write**	

0x00E0	WDATA7	[31:0]	**Read/Write**	
0x00E4	WDATA7_MSK	[31:0]	**Read/Write**	
0x00E8	SIZE	[31:0]	**Read/Write**	
		[1:0]		Monitor 0
		[5:4]		Monitor 1
		[9:8]		Monitor 2
		[13:12]		Monitor 3
		[17:16]		Monitor 4
		[21:20]		Monitor 5
		[25:24]		Monitor 6
		[29:28]		Monitor 7
0x00EC	SIZE_MSK	[31:0]	**Read/Write**	
		[1:0]		Monitor 0
		[5:4]		Monitor 1
		[9:8]		Monitor 2
		[13:12]		Monitor 3
		[17:16]		Monitor 4
		[21:20]		Monitor 5
		[25:24] [29:28]		Monitor 6 Monitor 7
0x00F0	CONTROL	[31:0]	**Read/Write**	
		[1:0]		Monitor 0 W/R Select 2'b00: No monitor 2'b01: Monitor write data 2'b10: Monitor read data 2'b11: No monitor
		[3:2]		Monitor 1 W/R Select
		[5:4]		Monitor 2 W/R Select
		[7:6]		Monitor 3 W/R Select
		[9:8] [11:10]		Monitor 4 W/R Select Monitor 5 W/R Select
		[13:12] [15:14]		Monitor 6 W/R Select Monitor 7 W/R Select
		[16]		Monitors Interrupt Enable 1'b0: Disable 1'b1: Enable
0x00F4	INT STATUS	[31:0] [0]	**Read Only**	 Mon_irq0
		[1]		Mon_irq1
		[2]		Mon_irq2
		[3]		Mon_irq3
		[4]		Mon_irq4
		[5]		Mon_irq5
		[6]		Mon_irq6
		[7]		Mon_irq7
0x00F8	INT CLEAR	[31:0]	**Write Only**	
		[0]		Write 1 to clear mon_irq0

		[1]		Write 1 to clear mon_irq1
		[2]		Write 1 to clear mon_irq2
		[3]		Write 1 to clear mon_irq3
		[4]		Write 1 to clear mon_irq4
		[5]		Write 1 to clear mon_irq5
		[6]		Write 1 to clear mon_irq6
		[7]		Write 1 to clear mon_irq7

4.4 MCU Interrupt Controller

The MCU interrupt controller collects interrupt trigger signals, applies mask on them and generates one of the MCU interrupt signals, FIQ or IRQ (Figure 13). The interrupt triggers can be programmed to be either level sensitive or edge sensitive, triggered either at high-level or at low-level in level-sensitive case, either at positive edge or at negative edge in edge-triggered mode.

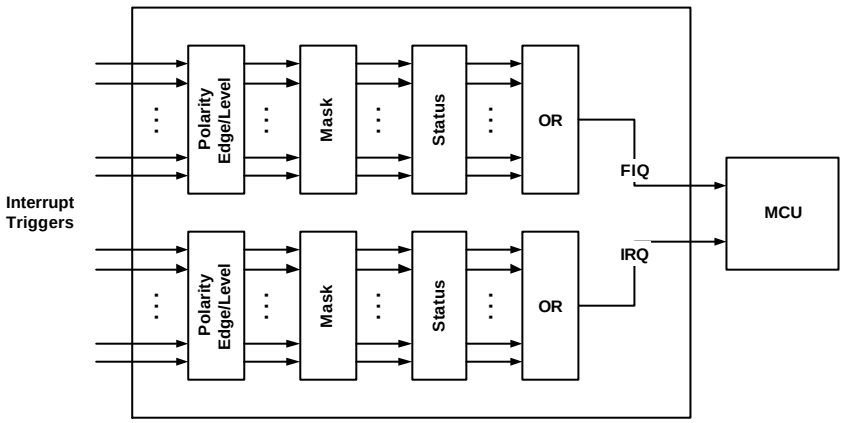


Figure 13: MCU interrupt controller.

On SC8800A1, clock-independent interrupts include the following.

1. Keypad
2. CHINT
3. Power Button
4. ALARM0
5. System Counter
6. RTC timer 0/1

On SC8800A2, two more clock-independent interrupts were added.

1. GPIO
2. RTC Minute interrupt (through GPIO [24])

4.4.1 Interrupt Control Registers

Base Address: 0x7010_0000

Address	Signal	Bit Pos	Default	Description

0x0000	IRQ Status	[31:0]	0 Read Only	Masked IRQ Status.
	INTCTrtc1	0	0	RTC Timer 1 interrupt
	INTCTrtc2	1	0	RTC Timer 2 interrupt
	COMMRX	2	0	Int from ARM comm. rx
	COMMTX	3	0	Int from ARM comm. tx
	INTCNT1	4	0	Timer 1 interrupt
	INTCNT2	5	0	Timer 2 interrupt
	gpio_irq	6	0	Interrupt from GPIO
	rtc_irq	7	0	Interrupt from real time clock
	kpd_irq	8	0	Interrupt from keypad
	i2c_irq	9	0	Interrupt from I2C
	sim_irq	10	0	Interrupt from sim
	ser_irq0	11	0	Serial 0 interrupt
	ser_irq1	12	0	Serial 1 interrupt
	dsp_irq	13	0	Interrupt from dsp
	adc_irq	14	0	Adc done interrupt
	gea_poll_int	15	0	Gea done interrupt
	syscnt_irq | PBINT | CHINT	16	0	Interrupt from system counter or power button or charger input
	wdg_irq	17	0	Interrupt from Watchdog
	ser_irq2	18	0	Serial 2 interrupt
	spk_irq	19	0	Speaker interrupt
	dma_irq	20	0	DMA interrupt
	vbc_irq	21	0	VBC interrupt
	mon_irq0	22	0	Bus monitor 0 interrupt
	mon_irq1 mon_irq2	23 24	0 0	Bus monitor 1 interrupt Bus monitor 2 interrupt
	mon_irq3 mon_irq4	25 26	0 0	Bus monitor 3 interrupt Bus monitor 4 interrupt
	mon_irq5	27	0	Bus monitor 5 interrupt
	mon_irq6	28	0	Bus monitor 6 interrupt
	mon_irq7	29	0	Bus monitor 7 interrupt
	dsp_frq	30	0	DSP FRQ
	Software int	31	0	Reserved for Software interrupt
0x0004	FIQ Status	[31:0]	0 Read Only	Masked FIQ Status
0x0008	IRQ Raw Status	[31:0]	- Read only	Raw interrupt status from different interrupt sources.
0x000C	Interrupt Select	[31:0]	0 Read/Write	Select bits for the corresponding output interrupt type between IRQ and FIQ. 1'b0: IRQ 1'b1: FIQ
0x0010	Interrupt Enable	[31:0]	0 Read/Write	Enable bits for the corresponding interrupt sources.
0x0014	Interrupt Enable Clear	[31:0]	- Write only	Write 1 to this register with corresponding bits to disable the corresponding interrupts.

0x0018	Software Interrupt	[31]	0 Read/Write	Write to this register will issue a software interrupt.
0x001C	Software Interrupt Clear	[31]	0 Write only	Write 1 to this register with corresponding bits to disable the corresponding interrupts.
0x0020	Protect Bit	[0]	0 Read/Write	This register can only be accessed in privilege mode. 1'b0: Free access to interrupt control registers 1'b1: Interrupt control registers can only be accessed in privilege mode.
0x0034	Default Vector Address	[31:0]	0 Read/Write	Default Vector Address.

4.5 DMA Control Registers

Base Address: 0x7020_0000

Address	Signal	Bit Pos	Default	Description
0x0000	General Control Register	[31:0]	**Read/Write**	
		[31:16]	0	**control the interval between the data transfer burst. The unit is bus clk**
		[15:2]	0	**reserved.**
		[1]		**error abort enable.**
		[0]		**pause_req.**
0x0004	Channel 0 to 7 interrupt enable register	[31:0]	0 **Read/Write**	.
		[7:0]		**channel0 to channel7 interrupt enable.**
0x0020	Channel 0 config	[15:0]		
		[0]	Read/Write 0	**channel enable (channel data transfer start)**
		[2:1]	Read-only 0	**data transfer size. 00: byte. 01: halfword. 10: word.**
		[3]	Read-only 0	**destination data address increase in data transfer.**
		[4]	Read-only 0	**source address increase in data transfer.**
		[5]	Read/Write 0	**hardware reqest enable**
		[7]	Read/Write	**link list pointer enable**

Address	Signal	Bit Pos	Default	Description
			0	
		[10]	Read/Write 0	channel busy.
		[11]	Read/Write 0	data transfer done
		[12]	Read/Write 0	data transfer error.
		[15:13]	Read/Write 0	channel data transfer priority.
0x0024	Channel 0 size ctl	[31:0]		
		[11:0]	Read/Write 0	total data transfer size.
		[15]	Read/Write 0	Infinit data transfer enable.
		[24:16]	Read/Write 0	burst size.
		[26:25]	Write-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[27]	Write-only 0	destination data address increase in data transfer.
		[28]	Write-only 0	source address increase in data transfer.
0x0028	Channel 0 source address	[31:0]	**Read/Write**	source address
0x002C	Channel 0 destination address	[31:0]	**Read/Write**	destination address
0x0030	Channel 0 link list pointer register	[31:0]	**Read/Write**	Link list pointer address
0x0040	Channel 1 config	[15:0]		
		[0]	Read/Write 0	channel enable (channel data transfer start)
		[2:1]	Read-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[3]	Read-only 0	destination data address increase in data transfer.
		[4]	Read-only 0	source address increase in data transfer.
		[5]	Read/Write 0	hardware reqest enable
		[7]	Read/Write 0	link list pointer enable
		[10]	Read/Write 0	channel busy.
		[11]	Read/Write 0	data transfer done

Address	Signal	Bit Pos	Default	Description
		[12]	Read/Write 0	data transfer error.
		[15:13]	Read/Write 0	channel data transfer priority.
0x0044	Channel 1 size ctl	[31:0]		
		[11:0]	Read/Write 0	total data transfer size.
		[15]	Read/Write 0	Infinit data transfer enable.
		[24:16]	Read/Write 0	burst size.
		[26:25]	Write-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[27] [28]	Write-only 0 Write-only 0	destination data address increase in data transfer. source address increase in data transfer.
0x0048	Channel 1 source address	[31:0]	**Read/Write**	source address
0x004C	Channel 1 destination address	[31:0]	**Read/Write**	destination address
0x0050	Channel 1 link list pointer register	[31:0]	**Read/Write**	Link list pointer address
0x0060	Channel 2 config	[15:0]		
		[0]	Read/Write 0	channel enable (channel data transfer start)
		[2:1]	Read-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[3]	Read-only 0	destination data address increase in data transfer.
		[4]	Read-only 0	source address increase in data transfer.
		[5]	Read/Write 0	hardware reqest enable
		[7]	Read/Write 0	link list pointer enable
		[10]	Read/Write 0	channel busy.
		[11]	Read/Write 0	data transfer done
		[12]	Read/Write 0	data transfer error.
		[15:13]	Read/Write 0	channel data transfer priority.

Address	Signal	Bit Pos	Default	Description
0x0064	Channel 2 size ctl	[31:0]		
		[11:0]	Read/Write 0	total data transfer size.
		[15]	Read/Write 0	Infinit data transfer enable.
		[24:16]	Read/Write 0	burst size.
		[26:25]	Write-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[27]	Write-only 0	destination data address increase in data transfer.
		[28]	Write-only 0	source address increase in data transfer.
0x0068	Channel 2 source address	[31:0]	**Read/Write**	source address
0x006C	Channel 2 destination address	[31:0]	**Read/Write**	destination address
0x0070	Channel 2 link list pointer register	[31:0]	**Read/Write**	Link list pointer address
0x0080	Channel 3 config	[15:0]		
		[0]	Read/Write 0	channel enable (channel data transfer start)
		[2:1]	Read-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[3]	Read-only 0	destination data address increase in data transfer.
		[4]	Read-only 0	source address increase in data transfer.
		[5]	Read/Write 0	hardware reqest enable
		[7]	Read/Write 0	link list pointer enable
		[10]	Read/Write 0	channel busy.
		[11]	Read/Write 0	data transfer done
		[12]	Read/Write 0	data transfer error.
		[15:13]	Read/Write 0	channel data transfer priority.
0x0084	Channel 3 size ctl	[31:0]		
		[11:0]	Read/Write 0	total data transfer size.
		[15]	Read/Write 0	Infinit data transfer enable.

Address	Signal	Bit Pos	Default	Description
		[24:16]	Read/Write 0	burst size.
		[26:25]	Write-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[27]	Write-only 0	destination data address increase in data transfer.
		[28]	Write-only 0	source address increase in data transfer.
0x0088	Channel 3 source address	[31:0]	**Read/Write**	source address
0x008C	Channel 3 destination address	[31:0]	**Read/Write**	destination address
0x0090	Channel 3 link list pointer register	[31:0]	**Read/Write**	Link list pointer address
0x00A0	Channel 4 config	[15:0]		
		[0]	Read/Write 0	channel enable (channel data transfer start)
		[2:1]	Read-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[3]	Read-only 0	destination data address increase in data transfer.
		[4]	Read-only 0	source address increase in data transfer.
		[5]	Read/Write 0	hardware reqest enable
		[7]	Read/Write 0	link list pointer enable
		[10]	Read/Write 0	channel busy.
		[11]	Read/Write 0	data transfer done
		[12]	Read/Write 0	data transfer error.
		[15:13]	Read/Write 0	channel data transfer priority.
0x00A4	Channel 4 size ctl	[31:0]		
		[11:0]	Read/Write 0	total data transfer size.
		[15]	Read/Write 0	Infinit data transfer enable.
		[24:16]	Read/Write 0	burst size.
		[26:25]	Write-only 0	data transfer size. 00: byte. 01: halfword. 10: word.

Address	Signal	Bit Pos	Default	Description
		[27]	Write-only 0	destination data address increase in data transfer.
		[28]	Write-only 0	source address increase in data transfer.
0x00A8	Channel 4 source address	[31:0]	**Read/Write**	source address
0x00AC	Channel 4 destination address	[31:0]	**Read/Write**	destination address
0x00B0	Channel 4 link list pointer register	[31:0]	**Read/Write**	Link list pointer address
0x00C0	Channel 5 config	[15:0]		
		[0]	Read/Write 0	channel enable (channel data transfer start)
		[2:1]	Read-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[3]	Read-only 0	destination data address increase in data transfer.
		[4] [5] [7]	Read-only 0 Read/Write 0 Read/Write 0	source address increase in data transfer. hardware reqest enable link list pointer enable
		[10]	Read/Write 0	channel busy.
		[11]	Read/Write 0	data transfer done
		[12]	Read/Write 0	data transfer error.
		[15:13]	Read/Write 0	channel data transfer priority.
0x00C4	Channel 5 size ctl	[31:0]		
		[11:0]	Read/Write 0	total data transfer size.
		[15]	Read/Write 0	Infinit data transfer enable.
		[24:16]	Read/Write 0	burst size.
		[26:25]	Write-only 0	data transfer size. 00: byte. 01: halfword. 10: word.
		[27]	Write-only 0	destination data address increase in data transfer.
		[28]	Write-only 0	source address increase in data transfer.

Address	Signal	Bit Pos	Default	Description
0x00C8	Channel 5 source address	[31:0]	**Read/Write**	source address
0x00CC	Channel 5 destination address	[31:0]	**Read/Write**	destination address
0x00D0	Channel 5 link list pointer register	[31:0]	**Read/Write**	**Link list pointer address**
0x00E0	Channel 6 config	[15:0]		
		[0]	Read/Write 0	**channel enable (channel data transfer start)**
		[2:1]	Read-only 0	**data transfer size. 00: byte. 01: halfword. 10: word.**
		[3]	Read-only 0	**destination data address increase in data transfer.**
		[4]	Read-only 0	**source address increase in data transfer.**
		[5]	Read/Write 0	**hardware reqest enable**
		[7]	Read/Write 0	**link list pointer enable**
		[10]	Read/Write 0	**channel busy.**
		[11]	Read/Write 0	**data transfer done**
		[12]	Read/Write 0	**data transfer error.**
		[15:13]	Read/Write 0	**channel data transfer priority.**
0x0044	Channel 6 size ctl	[31:0]		
		[11:0]	Read/Write 0	**total data transfer size.**
		[15]	Read/Write 0	**Infinit data transfer enable.**
		[24:16]	Read/Write 0	**burst size.**
		[26:25]	Write-only 0	**data transfer size. 00: byte. 01: halfword. 10: word.**
		[27]	Write-only 0	**destination data address increase in data transfer.**
		[28]	Write-only 0	**source address increase in data transfer.**
0x00E8	Channel 6 source address	[31:0]	**Read/Write**	source address
0x00EC	Channel 6	[31:0]	**Read/Write**	destination address

Address	Signal	Bit Pos	Default	Description
	destination address			
0x00F0	Channel 6 link list pointer register	[31:0]	**Read/Write**	**Link list pointer address**
0x0100	Channel 7 config	[15:0]		
		[0]	Read/Write 0	**channel enable (channel data transfer start)**
		[2:1]	Read-only 0	**data transfer size. 00: byte. 01: halfword. 10: word.**
		[3]	Read-only 0	**destination data address increase in data transfer.**
		[4]	Read-only 0	**source address increase in data transfer.**
		[5]	Read/Write 0	**hardware reqest enable**
		[7]	Read/Write 0	**link list pointer enable**
		[10]	Read/Write 0	**channel busy.**
		[11]	Read/Write 0	**data transfer done**
		[12]	Read/Write 0	**data transfer error.**
		[15:13]	Read/Write 0	**channel data transfer priority.**
0x0044	Channel 7 size ctl	[31:0]		
		[11:0]	Read/Write 0	**total data transfer size.**
		[15]	Read/Write 0	**Infinit data transfer enable.**
		[24:16]	Read/Write 0	**burst size.**
		[26:25]	Write-only 0	**data transfer size. 00: byte. 01: halfword. 10: word.**
		[27]	Write-only 0	**destination data address increase in data transfer.**
		[28]	Write-only 0	**source address increase in data transfer.**
0x0108	Channel 7 source address	[31:0]	**Read/Write**	source address
0x010C	Channel 7 destination address	[31:0]	**Read/Write**	destination address
0x0110	Channel 7 link list pointer register	[31:0]	**Read/Write**	**Link list pointer address**

4.6 External SDRAM Memory Interface

4.6.1 SDRAM Interface Features

- Support 1/2/4/8 banks. For 8-bank mode, the maximum CS number is 2 instead of 4.
- Support 16 (2 DQMs) and 32 (4 DQMs) bit width.
- Support 1/2/4/8/16/full-page burst. For full-page burst, alternative bank access MUST be disabled.
- Support from 16Mbits to 8Gbits (No data sheet for 1G/2G/4G/8G bit SDRAM), that is, HADDR [29:0] is used. Support width-configurable row address from 11bits to 13bits, support width-configurable column address from 8bits to 12bits.
- Support 1/2/3/4 CAS latency for SDRAM configuration. That is, support configurable 1/2/3/4/5/6/7 read back data latency.
- Support real wrap mode burst. That is, one wrap burst is implemented by one SDRAM burst stead of divided into two SDRAM bursts.
- Only support sequential type.
- Support AUTO-REFRESH. And this function can be disabled.
- Support self-refresh and support two ways to enter this state – software invoking and hardware invoking. If software writes a self-refresh command or a hardware signal is set to 1, SDRAM enters self-refresh state. If software writes a wake-up command or this signal is cleared to 0, SDRAM exits self-refresh state. Designer Note, all banks must be idle when entering self-refresh, and tXSR must be met when exiting self-refresh. So for hardware invoking, maybe a output done signal is required.
- Only support big endian in SDRAM interface.
- Support up to 4 CS. That is, support up to 4-die stacked SDRAM. And the mapping from CS to AHB is configurable, see Misc. Features for detailed.
- Not support CONCURRENT AUTO PRECHARGE.
- Not support SDRAM POWER DOWN AND DEEP POWER DOWN mode.
- Not support CLOCK SUSPEND mode.
- Not support BURST READ / SINGLE WRITE mode.
- Support configurable mode register and extended mode registers.
- Support configurable auto refresh interval time.
- Support configurable PRECHARGE command period tRP.
- Support configurable ACTIVE to READ/WRITE delay tRCD.
- Support configurable WRITE recovery time tWR.
- Support configurable AUTO REFRESH command period tRFC/tRC/tARFC.
- Support configurable exit SELF REFRESH to ACTIVE command time tXSR/tSREX/tSRFX.
- Support configurable LOAD MODE REGISTER command to ACTIVE or REFRESH command delay tMRD.
- Support an auto-refresh counter as a watchdog for SDRAM sleep.
- Support the following software direct command list for SDRAM:
  - ✓ Initialize (Don't provide this command because it's not decided which mode registers are used)
  - ✓ Pre-charge all banks (one command for one chip or for all chips)
  - ✓ Auto-refresh (one command for one chip or for all chips)
  - ✓ Load mode register and extended mode register (one command for one chip or for all chips)
  - ✓ Self-refresh (one command for one chip or for all chips)
  - ✓ Resume (one command for one chip or for all chips)
- Support two options for software commands, one command for one chip, and one command for all chips. The option select depends on LDO driving strength.
- Support two options for auto-refresh routine, chip by chip refresh mode, and all chips at the same time mode. The option select depends on LDO driving strength.

- Support different SDRAM size. In detail, we make bank width, row width and column width configurable. That is, bank – 0/1/2/3 bits, row – 11/12/13 bits, column – 8/9/10/11/12 bits. The particular width for each type of SDRAM is decided and configured by software. The following list is a summary for current SDRAM, and this just is a reference for software. REMEMBER, more types besides this list are supported only if their bank width, row width and column width are in above lists.
  - ✓ 16M (1M*16) 2 banks row length = 11 column length = 8
  - ✓ 64M (4M*16) 4 banks row length = 12 column length = 8
  - ✓ 128M (8M*16) 4 banks row length = 12 column length = 9
  - ✓ 256M (16M*16) 4 banks row length = 13 column length = 9
  - ✓ 512M (32M*16) 4 banks row length = 13 column length = 10
  - ✓ 1G (64M*16) 4 banks row length = 13 column length = 11
  - ✓ 2G (128M*16) 4 banks row length = 13 column length = 12
  - ✓ 64M (2M*32) 4 banks row length = 11 column length = 8
  - ✓ 128M (4M*32) 4 banks row length = 12 column length = 8
  - ✓ 256M (8M*32) 4 banks row length = 13 (or 12) column length = 8 (or 9)
  - ✓ 512M (16M*32) 4 banks row length = 13 column length = 9
  - ✓ 1G (32M*32) 4 banks row length = 13 column length = 10
  - ✓ 2G (64M*32) 4 banks row length = 13 column length = 11
  - ✓ 4G (128M*32) 4 banks row length = 13 column length = 12
- Support that SELF-REFRESH is issued separately for different SDRAM chips.
- SDRAM clock is HCLK.

4.6.2 Control Registers

Base Address: 0x7030_0000

Address	Signal	Bit Pos	Default	Description
0x0000	ExtMemCtl0	[31:0]	**Read/Write**	
		[3:0]		Refresh period time
		[7:4] [10:8]		Precharge period time Active to read/write time
		[12:11]		Write recovery time
		[14:13]		Cas latency. Can be set to 2 or 3.
		[17:15]		burst length control. Set to burst of 8 under normal condition (0x011). Set to full page under dcam write mode (0x111).
		[18]		Control the mapping of sdram address to AHB address: 0: (row_address, bank_address, col_address) map to AHB address (MSB to bit1). 1: (bank address, row_address, col_address) map to AHB address (MSB to bit1).
		[20:19]		the size of the SDRAM. Can be: 00: 4Mx16, 01:8Mx16, 10:16Mx16.
		[28]	0	Bus error indicator
		[29]	0x0	Write protection error
		[30]	0	Write protect

		[31]	0x0	SDRAM flag
0x0004	ExtMemCtl1	[31:0]	**Read/Write**	Per bit definition and default value are the same as ExtMemCtl0
0x0008	ExtMemCtl2	[31:0]	**Read/Write**	Per bit definition and default value are the same as ExtMemCtl0
0x000C	ExtMemCtl3	[31:0]	**Read/Write**	Per bit definition and default value are the same as ExtMemCtl0
0x0020	ExtMemINI	[31:0]	**Read/Write**	External memory initialization control
		[4:0]	0	Extended mode register load value
		[13:11]	0	Refreash interval control
		[21:14]	0	Refreash prescale value
		[22]	0	Initial mode register request
		[23]	0	Load mode register request
		[24]	0	Load extended mode register request
		[25]	0	Suspension request
		[26]	0	Resume request
		[29:27]	0	Indicate the ExtMemCtlN number used in initial process.
		[31]	0	Force buffer flush
0x0024	ExtMem_msctl	[31:0]	**Read/Write**	External memory misc control
		[0]	Read only	Reflex the EMCKE output vaule
		[1]	0	Ext mem controller idle bit
		[2]	0	reserved
		[3]	0	Internal mem access error report enable
		[4]	0	Mem controller watch dog enable
		[5]	0	Flag for sdram ini done
		[7]	0	**Must be tied to be '1' after initialization.**
		[11:7]	0	Reserved
		[15:12]	0	ROM address remap base address

4.7 Watchdog Timer

SC8800 provides a watchdog timer (WDT) that can reset the MCU subsystem if the MCU SW does not reload the timer. The timer has 32 bits.

The counter is controlled by RTC clock. With RTC clock at 32.768 kHz, 32 bits provide up to 131,072 seconds or more than 36 hours of count time.

The watchdog works in one of the following modes:

- The counter counts down to 0 and then sends an interrupt to the MCU.
- The counter counts down to 0 and then reset the MCU subsystem.

The watchdog after HW reset is disabled by default.

The control registers in WDT are controlled by a lock and only a specific value can unlock the register write enable.

4.7.1 Watchdog Registers

Watch dog Base Address: 0x8000_0000

Address	Signal	Bit Pos	Default	Description
0x0000	Load	[31:0]	0xFFFF_FFFF **Read/Write**	Watch dog Load
0x0004	Value	[31:0]	0xFFFF_FFFF **Read**	Current value for watch dog counter
0x0008	Control	[31:0]	0 **Read/Write**	Watch dog control register
		[0]	0	Watch dog interrupt out enable
		[1]	0	Watch dog reset out enable
0x000C	Interrupt Clear	[31:0]	- **Write**	Write any value into this register clear the watch dog interrupt
0x0010	Raw interrupt	[0]	0 **Read**	Raw watch dog interrupt
0x0014	Masked interrupt	[0]	0 **Read**	Masked watch dog interrupt
0x0C00	Lock Register	[31:0]	- **Write**	Write 0x1ACCE551: unlock Write others: lock
		[0]	0 **Read**	1'b0: unlocked 1'b1: locked

4.8 Timers

SC8800 provides several timers at the MCU side.

4.8.1 General-Purpose Timers

The MCU sub-system provides two general-purpose timers. These two timers can be controlled individually.

Each timer has a 23-bit counter and the stop count (one shot mode) or period count (periodical mode) can be set to a control register. The frequency of the count clock is 32.768 kHz (RTC clock)[1].

At the end of the count, the timer can generate an interrupt to the MCU.

4.8.2 Timer Control Registers

Timer Base Address: 0x8100_0000

Address	Signal	Bit Pos	Default	Description

[1] On SC8800A1, the clock is APB clock.

Address	Signal	Bit Pos	Default	Description
0x0000	Timer0_load	[15:0]	0 **Read/Write**	Write to this register will reload the timer with the new value.
0x0004	Timer0_value	[15:0]	0 **Read Only**	Return the current timer value.
0x0008	Timer0_control	[15:0]	**Write Only**	Bit [3:2]: 00: prescale timer clk (apb clk) by 1. 01: prescale timer clk (apb clk) by 16. 10: prescale timer clk (apb clk) by 256. 11: undefined. Bit 6: 1: select periodic mode 0: select free running mode Bit 7: enable bit of this timer.
	Timer_int_status	[15:0]	0 **Read Only**	Bit0: Timer0 Interrupt Status Bit1: Timer1 Interrupt Status
0x000C	Timer0_clear	[15:0]	0 **Write Only**	Write to this register will clear the interrupt generated by this timer.
0x0020	Timer1_load	[15:0]	0 **Read/Write**	Write to this register will reload the timer with the new value.
0x0024	Timer1_value	[15:0]	0 **Read Only**	Return the current timer value.
0x0028	Timer1_control	[15:0]	0 **Write Only**	Bit [3:2]: 00: prescale timer clk (apb clk) by 1. 01: prescale timer clk (apb clk) by 16. 10: prescale timer clk (apb clk) by 256. 11: undefined. Bit 6: 1: select periodic mode 0: select free running mode Bit 7: enable bit of this timer.
	Timer_int_status	[15:0]	0 **Read Only**	Bit0: Timer0 Interrupt Status Bit1: Timer1 Interrupt Status
0x002C	Timer1_clear	[15:0]	0 **Write Only**	Write to this register will clear the interrupt generated by this timer.

4.9 RTC Timer Control Registers

RTC Timer Base Address: 0x8F00_0000

Address	Signal	Bit Pos	Default	Description
0x0000	Timer0_load	[15:0]	0 **Read/Write**	Write to this register will reload the timer with the new value.
0x0004	Timer0_value	[15:0]	0 **Read Only**	Return the current timer value.
0x0008	Timer0_control	[15:0]	0 **Write Only**	Bit 5: **enable bit of wake output**. Bit 6: 1: select periodic mode 0: select free running mode Bit 7: enable bit of this timer.
	Timer_int_status	[15:0]	0 **Read Only**	Bit0: RTC Timer0 Interrupt Status Bit1: RTC Timer1 Interrupt Status
0x000C	Timer0_clear	[15:0]	0 **Write Only**	Write to this register will clear the interrupt generated by this timer.
0x0020	Timer1_load	[15:0]	0 **Read/Write**	Write to this register will reload the timer with the new value.
0x0024	Timer1_value	[15:0]	0 **Read Only**	Return the current timer value.
0x0028	Timer1_control	[15:0]	0 **Write Only**	Bit 5: **enable bit of wake output**. Bit 6: 1: select periodic mode 0: select free running mode Bit 7: enable bit of this timer.
	Timer_int_status	[15:0]	0 **Read Only**	Bit0: RTC Timer0 Interrupt Status Bit1: RTC Timer1 Interrupt Status
0x002C	Timer1_clear	[15:0]	0 **Write Only**	Write to this register will clear the interrupt generated by this timer.

4.10 Remap and Pause Control Registers

Remap Base Address: 0x8200_0000

Address	Signal	Bit Pos	Default	Description
0x0000	Pause_en	[15:0]	0 **Write Only**	Write to this register will pause the ARM cpu, and ARM cpu will come out of pause state when any interrupt received.
0x0020	Remap	[15:0]	Read/Write	
		[1:0]	power strapped in, default 0	Remap memory space. See 1.1.1

4.11 MCU Peripherals

Many peripheral control functions are provided for easy system integration.

Most of the peripherals are controlled by the MCU, including data ports, SIM interface, keypad interface, etc. The data ports are provided for data transfers with other devices, and they include both serial ports (UART and 2-wire serial interface) and a parallel port.

The MCU control registers are listed with the functional modules. Only 32 bits read and write are supported for control register access.

4.12 UART Controller

SC8800 provides three UART ports, UART0 is on the MCU side, and UART1 and UART2 can be used on either the MCU or the DSP side.

4.12.1 Overview

SC8800's universal asynchronous receiver/transmitter (UART) serial interfaces are compatible to National 16550 or Intel 8251 asynchronous serial interface. Following are the features the UART interfaces provide:

- Full-duplex operation.
- Hardware flow control support.
- Support full modem hand-shaking signals.
- 128 byte of Rx FIFO and 128 byte of Tx FIFO.

Each UART in SC8800 shares the same design and has an independent baud rate generator, which is used as a clock reference for data recovery. Each of them also has a 128-byte Rx FIFO and a 128-byte Tx FIFO to decrease MCU interrupt load during the data receiving or transmitting. In the text that follows, only the register and signal names of the UART0 are used. All of these references apply to UART1 and UART2 as well.

4.12.2 Functional Description

Each UART includes the pins as shown in the following table.

Table 35: List of UART pins.

Pin Name	Signal	I/O	Description
U0TXD	Transmit data	O	Transmit data output.
U0RXD	Receive data	I	Receive data input.
U0RTSN	Ready to send	O	**When hardware flow control is disabled, programming the UART_CTL0 bit 6 directly controls this output. That is, this pin is the inverse of the UART_CTL0 bit 6. Hardware flow control is disabled by default. If receive hardware flow control is enabled through writing the UART_CTL1 bit 7 to 1, ready to send signal will be auto generated in hardware. In this case, U0RTSN remains high as long as the number of unread bytes in the Rx FIFO is greater than the configured threshold value, which is specified in the UART_CTL1 bit 0 to bit 6.**
U0CTSN	Clear to send	I	**The value of this input signal is indicated by the UART_STS0 register bit 9 after logical inverse. When the U0CTSN input changes value, an interrupt is generated towards the MCU. If hardware flow control is enabled and the U0CTSN input pin is low, the UART immediately stops data transmission on the U0TXD output pin after it completes the current data byte transmission.**
U0TRN	Data terminal ready	O	**This output is directly controlled by programming the corresponding UART_CTL0 register bit 8. Please note that the output is the inverse of the register bit programmed.**
U0DTSN	Data set ready	I	**The value of this input signal is indicated by the UART_STS0 status register bit 8. When DSR input changes value, an interrupt is generated towards the MCU.**

Note: If Ring Indication and Carrier Detect signals are required for this interface, they can be supplied by using two GPIO pins.

The size of Tx FIFO for the transmission is 128 bytes. It uses a FIFO pointer to monitor the number of data in the Tx FIFO that have not yet been transmitted. The MCU can read the UART_STS1 bits [15:8] to access the pointer.

The UART also uses a configurable Tx interrupt threshold that can be configured by writing to the UART_CTL2 register bits [14:8]. This field specifies the number of empty spaces that must be available in the Tx FIFO before issuing an interrupt. Whenever the number of empty spaces exceeds this value, an interrupt is generated to MCU (UART_STS0 Tx FIFO empty bit, bit 0). The interrupt bit in the UART_STS0 Tx FIFO empty bit is automatically cleared when the number of empty bytes in the Tx FIFO equals or falls below the specified Tx FIFO threshold.

The size of Rx FIFO for receiving data is 128 bytes. It uses a pointer to monitor the number of characters in the Rx FIFO that have not yet been read by the MCU. The MCU can access the pointer by reading the UART_STS1 bits [7:0].

The UART also uses a configurable Rx interrupt threshold that can be configured by writing to the UART_CTL2 register bits [6:0]. This field specifies the number of unread data that must be available in the Rx FIFO before issuing an interrupt. Whenever the number of unread data exceeds this value, an interrupt is generated to the MCU. The interrupt is cleared by MCU reading the data out.

The UART interface also uses a receive timeout threshold, and the threshold can be configured by the UART_CTL1 bits [13:9]. Using the receive timeout interrupt prevents deadlocks, which occur when unread data are present in the Rx FIFO, but are not enough to generate a normal Rx interrupt. An Rx FIFO full interrupt (UART_STS0 bit 0) is always issued when the configured Rx timeout threshold time has elapsed and unread data are present in the Rx FIFO. When the CPU reads the last data in the Rx FIFO, the timer is reset and the interrupt is cleared automatically. Setting the timeout threshold to zero disables the timeout function.

When receiving data, UART also generates parity error/framing error if error condition appears. Also, if the receive FIFO over runs, an error interrupt is generated. Write to corresponding interrupt clear bits will clear these interrupts.

4.12.3 6-Wire UART Interface

This UART port is UART 0, used only by the MCU. It consists of the following pins.

- U0TXD, data output.
- U0RXD, data input.
- U0DTRN, active low.
- U0RTSN, active low.
- U0DSRN, active low.
- U0CTSN, active low.

The hardware flow control pins are also shared with JTAG pins, i.e., when JTAG is used, only Rx and Tx pins are available, reducing to a 2-wire UART port.

4.12.4 2-Wire UART Interface

This UART is UART 1 and can be used by either the MCU or DSP and consists of the following pins.

- U1TXD, data output.
- U1RXD, data input.

There is no hardware flow control for this UART interface.

4.12.5 UART0, UART1, UART2 Registers

ARM UART0 Base Address: 0x8300_0000

ARM UART1 Base Address: 0x8400_0000

ARM UART2 Base Address: 0x8E00_0000

DSP1 UART Base Address: 0xD000

DSP2 UART Base Address: 0x7C00

DSP Address	ARM Address	Signal	Bit Pos	Default	Description
0x00	0x0000	UART_TXD	[7:0]	N/A **Write only**	Write data to this address initiates a character transmission through tx fifo.
			[15:8]	Unused	

DSP Address	ARM Address	Signal	Bit Pos	Default	Description
0x02	0x0004	UART_RXD	[7:0]	N/A **Read only**	Reading this register retrieves the next data byte from the rx fifo.
			[15:8]	Unused	
0x04	0x0008	UART_STS0	[15:0]	**Read only**	
			[0]	0	Row Rx_fifo_full. This bit is set when the number of rx fifo data bytes is larger than the rx interrupt watermark value. Auto cleared when the condition disappears.
			[1]	0	Row Tx_fifo_empty This bit is set when the number of tx fifo data bytes is less than the tx interrupt watermark value. Auto cleared when the condition disappears.
			[2]	0	Row parity_error
			[3]	0	Row framing_error
			[4]	0	Row rxf_overrun
			[5]	0	Row dsr_change
			[6]	0	Row cts_change
			[7]	0	Row break_detect
			[8]	0	DSR: Data set ready
			[9]	0	CTS: clear to send
			[10]	0	RTS: Request to send
			[11]	0	RXD: rx data in
			[12]	0	TXD: tx data out
			[13]	0	Rx_tout: This bit is set when receive time is out. Auto cleared when the condition disappears.
			[15:14]	Unused	
0x06	0x000C	UART_STS1	[7:0]	0	Rx_fifo_cnt
			[15:8]	0	Tx_fifo_cnt
0x08	0x0010	UART_IEN	[7:0]	**Read/Write**	
			[0]	0	Rx_fifo_full interrupt enable
			[1]	0	Tx_fifo_empty interrupt enable
			[2]	0	Parity_error interrupt enable
			[3]	0	Framing_error interrupt enable
			[4]	0	Rxf_overrun interrupt enable
			[5]	0	Dsr_change interrupt enable
			[6]	0	Cts_change interrupt enable
			[7]	0	Break_detect interrupt enable
			[12:8]	Unused	
			[13]	0	Rx_tout interrupt enable
			[15:14]	Unused	
0x0A	0x0014	UART_ICLR	[7:0]	**Write Only**	
			[0]	0	Unused
			[1]	0	Unused
			[2]	0	Write “1” Clear parity_error interrupt

DSP Address	ARM Address	Signal	Bit Pos	Default	Description
			[3]	0	Write “1” Clear framing_error interrupt
			[4]	0	Write “1” Clear rx_fifo_overrun interrupt
			[5]	0	Write “1” Clear dsr_change interrupt
			[6]	0	Write “1” Clear cts_change interrupt
			[7]	0	Write “1” Clear break_detect interrupt
			[12:8]	Unused	
			[13]	0	Write “1” Clear Rx_tout interrupt
			[15:14]	Unused	
0x0C	0x0018	UART_CTL0	[8:0]		
			[0]	0	Odd_Parity 0: even parity, 1: odd parity.
			[1]	0	Parity_enable 0: parity disabled, 1: parity enabled.
			[3:2]	0x3	Byte_length, data byte length. 0: 5 bits, 1: 6 bits, 2: 7 bits, 3: 8 bits.
			[5:4]	0x3	Stop_Bit_Number 0: unused, 1: 1stop bit, 2: 1.5 stop bits, 3: 2 stop bits.
			[6]	0	Ready To Send When receive hardware flow control is not set, this bit controls the output of rtsn. When rhwf is set, this bit is controlled by the rx fifo level.
			[7]	0	Send Break When tx fifo is empty and tx is idle, setting this bit forces the tx data output low. Need to be cleared by mcu.
			[8]	0	Data Terminal Ready This bit controls the dtrn output
			[9]	0	Ir_tx_iv : Irda TX polarity inverse
			[10]	0	Ir_rx_iv : Irda RX polarity inverse
			[11]	0	Ir_tx_en : Irda TX enable
			[12]	0	Ir_dplx : Irda TX/RX enable
			[13]	0	Ir_wctl : set “1”, increase pulse width for one clock.
			[15:14]	Unused	
0x0E	0x001C	UART_CTL1	[13:0]		
			[6:0]	0	Receive_Hardware_Flow_Control_Threshold When rhwf is enabled, if the number of unread bytes in the rx fifo is greater than the receive hardware flow control threshold value, the rtsn is set to high to stop the remote tx. Rts bit in the control register is also cleared at this time.
			[7]	0	Receive_Hardware_Flow_Control_Enable
			[8]	0	Transmit_Hardware_Flow_Control_Enable

DSP Address	ARM Address	Signal	Bit Pos	Default	Description
			[13:9]	0	Receive_Timeout_Value
			[14]	0	UART self Loop: Tx input to Rx
			[15]	0	DMA_en: enable uart fifo control by DAM
0x10	0x0020	UART_CTL2	[14:0]		
			[6:0]	0	Receive_Int_WaterMark
			[7]	Unused	
			[14:8]	0	Transmit_Int_WaterMark
			[15]	Unused	
0x12	0x0024	UART_CKD0	[15:0]	0x054A	Clock divider bit 0 to 15
0x14	0x0028	UART_CKD1	[4:0]	0	Clock divider bit 16 to 20
			[15:5]	Unused	
0x16	0x002C	UART_STS2	[15:0]	**Read only**	Masked STS0 register, masked by IEN reg.
			[0]	0	Masked Rx_fifo_full This bit is set when the number of rx fifo data bytes is larger than the rx interrupt watermark value or when receive time is out. Auto cleared when the condition disappears.
			[1]	0	Masked Tx_fifo_empty This bit is set when the number of tx fifo data bytes is less than the tx interrupt watermark value. Auto cleared when the condition disappears.
			[2]	0	Masked Parity_error
			[3]	0	Masked Framing_error
			[4]	0	Masked Rxf_overrun
			[5]	0	Masked Dsr_change
			[6]	0	Masked Cts_change
			[7]	0	Masked Break_detect
			[12:8]	Unused	
			[13]	0	Masked Rx_tout interrupt
			[15:14]	Unused	
0x18		UART_WAIT_STATE	[15:0]		
			[3:0]	1	DSP bus wait cycle
			[15:4]	Unused	

4.13 SPI Interface

SC8800 provides up to 3 SPI interface ports, which share the pins with UART and IrDA.

Table 36: Organization of the serial ports

	Serial port 0	Serial port 1	Serial port 2
User	MCU	Shared	Shared
Function 1	UART 0	UART 1	UART 2
Function 2	SPI 0	SPI 1	SPI 2
Function 3			IrDA

The signals are mapped to the pins as shown in the following table.

Table 37: Pin mapping for the serial ports.

UART Pins	SPI Pins
UART_TXD	SPI_DO
UART_RXD	SPI_DI
UART_DTRN	SPI_CLK
UART_RTSN	SPI_CSN0
UART_DSRN	SPI_CSN1
UART_CTSN	SPI_CSN2
	SPI_CSN3

4.13.1 Serial Port Interface Control Registers

ARM SPI0 Base Address: 0x8300_0000

ARM SPI1 Base Address: 0x8400_0000

ARM SPI2 Base Address: 0x8E00_0000

DSP1 SPI Base Address: 0xD000

DSP2 SPI Base Address: 0x7C00

DSP Address	ARM Address	Signal	Bit Pos	Default	Description
0x00	0x0000	spi_txd	[31:0]	N/A **Write only**	Write data to this address initiates a character transmission through tx fifo.
		spi_rxd	[31:0]	N/A **Read only**	Read data from rx fifo.
0x02	0x0004	spi_sts0	[15:0]	**Read only**	
			[0]	0	Row Rx_fifo_full_tout. This bit is set when the number of rx fifo data bytes is larger than the rx interrupt watermark value or when receive time is out. Auto cleared when the condition disappears.
			[1]	0	Row Tx_fifo_empty This bit is set when the number of tx fifo data bytes is less than the tx interrupt watermark value. Auto cleared when the condition disappears.
			[2]	0	Row rxf_overrun

DSP Address	ARM Address	Signal	Bit Pos	Default	Description
			[3]	0	Data bus busy
			[4]	0	RXD: rx data in
			[5]	0	TXD: tx data out
			[31:6]	Unused	
0x04	0x0008	spi_sts1	[13:0]	**Read only**	
			[5:0]	0	Rx_fifo_cnt
			[7:6]		
			[13:8]	0	Tx_fifo_cnt
			[31:14]		
0x06	0x000C	spi_ien	[7:0]	**Read/Write**	
			[0]	0	Rx_fifo_full_tout interrupt enable
			[1]	0	Tx_fifo_empty interrupt enable
			[2]	0	Rxf_overrun interrupt enable
			[31:3]	Unused	
0x08	0x0010	spi_ctl0	[15:0]		
			[0]	0	Write “1” Enable Rx data shift in at clock neg-edge
			[1]	0	Write “1” Enable Tx data shift out at clock neg-edge
			[6:2]	0	Transmit data bit number, “1”: 1bit per word, “0”: 32 bit per word
			[7]	0	Write “1” enable data transmit/receive from LSB.
			[11:8]	0	Four bit chip select. There are totally 4 chip selects for SPI
			[14:12]	Unused	
			[15]	Write only	“1” Receive data only.
			[31:16]	Unused	
0x0A	0x0014	Spi_ctl1			
			[4:0]	0	Receive Data full threshold
			[9:5]	0	Transmit Data empty threshold
			[13:10]	Unused	
			[14]	0	SPI self Loop: Tx input to Rx
			[15]	0	DMA_en
0x0C	0x0018	Spi_ctl2	[15:0]	0	
			[4:0]	0	2-wire mode, w/r control bit position
			[5]	0	“1” enable 2-wire mode
			[6]	0	2-wire mode, read command polarity
			[7]	0	S8 mode (LCD S8)
			[9:8]	0	S8 cd maps to csn number
			[10]	0	2-wire Melody timing 2, csn high mode enable
			[11]	0	2-wire Melody timing 1, csn high mode enable

DSP Address	ARM Address	Signal	Bit Pos	Default	Description
			[15:12]	0	Csn high bit enable, “0x0001” enable csn0 high.
0x0E	0x001C	Spi_clkd	[15:0]	0x0003	Clock divider bit 0 to 15
0x10	0x0020	Spi_sts2	[15:0]	**Read only**	Masked STS0 register, masked by IEN reg.
			[0]	0	Masked Rx_fifo_full_tout. This bit is set when the number of rx fifo data bytes is larger than the rx interrupt watermark value or when receive time is out. Auto cleared when the condition disappears.
			[1]	0	Masked Tx_fifo_empty This bit is set when the number of tx fifo data bytes is less than the tx interrupt watermark value. Auto cleared when the condition disappears.
			[2]	0	Masked rxf_overrun
			[15:3]	Unused	
0x12	0x0024	SPI_ICLR	[2:0]	**Write Only**	
			[0]	0	Unused
			[1]	0	Unused
			[2]	0	Write “1” Clear rx over run interrupt
			[15:3]	Unused	
0x14	0x0028	SPI_ctl3	[15:8]		
			[7:0]	Unused	
			[15:8]	0	Transmit data interval, programmable from 0 to 255 clock cycles.

4.14 SIM Card Interface

The SIM card interface is implemented according to the GSM SIM card standard and supports T = 0 protocol only. The SIM card interface is to transfer data from/to a SIM card in an asynchronous fashion in half duplex mode through a bi-directional I/O pin.

The SIM interface consists of the following dedicated pins.

- SIMCLK, SIM clock.
- SIMDA, SIM data.
- VSIM, SIM power supply.
- SIMRST, SIM reset.

4.14.1 SIM Functional Description

The Features for the SIM card controller in SC8800 include:

- Automatic activation and deactivation sequence.
- Programmable generation of clock to the SIM card.
- Programmable transmission baud rate.

- Support for T = 0 asynchronous protocol type.
- One 16 byte Tx FIFO and one 16 byte Rx FIFO.
- Parity checking and error handling.

4.14.2 SIM Clock and Baud Rate Control

The SIM card clock is generated and sent from the SIM card controller to the SIM card. The SIM_CTL1 register bits [10:8] decide the generated SIM clock frequency.

Table 38: SIM clock control.

Clock mode, 0x8500_001C [10:8]	SIM clock frequency
0	MCU clock / 2
1	MCU clock / 4
2	MCU clock / 8
3	MCU clock / 16
4	MCU clock / 32
5	MCU clock / 64
6	MCU clock / 128
7	MCU clock / 256

The baud rate that the SIM card controller uses in Tx and Rx is controlled through programming SIM_CLK_DVD register.

The clock output to the SIM card needs to match the baud rate that the SIM card controller uses so that the Rx and Tx can be done correctly.

4.14.3 SIM Activation and Deactivation

SC8800 controls the activation and deactivation process with the SIM control registers. The following are brief list of commands for the activation and deactivation functions.

- Program active_deactiva_en bit to 1 to enable the activation/deactivation function.
- Program do_act bit to 1 to start the activation sequence.
- (Program do_deact bit to 1 to start the deactivation sequence.)
- By programming the auto_active_deactive to 1, the activation will be executed when a card is inserted and the deactivation will be executed when the card is removed.

4.14.4 Activation and Deactivation

The SIM card controller is implemented with the following modules: SIM card activation/deactivation control, SIM card Rx/Tx control, Rx and Tx FIFO and related FIFO controls, SIM card I/O control, and SIM card control registers.

The activation/deactivation control module controls the activation/deactivation sequence of the SIM interface. After sending an activate command (writing 1 to SIM_CTL0 register bit 11) or a deactivate command (writing 1 to SIM_CTL0 register bit 12), the activation sequence or deactivation sequence will be sent to the SIM card. With the auto_active_deactive bit in SM_CTL0 register (bit 14) enabled, the activate/deactivate will be done automatically when card insertion/removal happens.

4.14.5 SIM Data Tx and Rx

The Tx will start when the tx_enable is 1. There is data in the FIFO and the last Rx is completed. The tx_int_mark sets the condition under which the tx_int will happen. When the empty entry in the tx_fifo is less than tx_int mark, tx_empty int will be set.

The Rx will start when the rx_enable is 1, when the last Tx is done and there is data coming in from the data input. The rx_int_mark sets the condition under which the rx_int will happen. When the data entry in the rx_fifo is greater than rx_int mark, rx_full int will be set.

The bit_convention decides the MSB and LSB in the Tx/Rx serial data, logic_level decides if the high electrical level represents the logic 1 in Tx/Rx serial data, and odd_parity bit decides if using odd or even parity in Tx/Rx data.

In the SIM interface module, a timer is also designed to check if the receive portion is idle for a certain period of time and generate an interrupt when it happens. The watch_dog_count_limit defines the idle period in data bit streaming. Watch_dog_repeat_en decide if the timer start right after the interrupt is acknowledged (the interrupt is cleared). Write 1 to watch_dog_trigger start the timer counting. Any activity in the Rx portion will reset the timer counting.

4.14.6 Rx/Tx Control

The Rx/Tx control module controls the transmitting and receiving data to/from the SIM card. The SIM card controller receives/transmits data according to the data-level and bit-ordering that are specified in the Logic_level (bit 1) and Bit_convention (bit 0) bits in the SIM_CTL0 register. The data format is:
10 bits per character, 1 start bit + 8 data bits + 1 parity bit.

The SIM card controller includes a 16 byte Tx FIFO for data transmission. It uses a pointer to monitor the number of data in the Tx FIFO that have not yet been transmitted. The MCU can access the pointer by reading the SIM_STS1 register bit 4 to bit 0.

The SIM card controller also uses a configurable Tx interrupt threshold that can be configured by writing to the SIM_CTL1 register bit 4 to 7. This field specifies the number of empty characters that should be available in the Tx FIFO before issuing an interrupt. Whenever the number of empty characters exceeds this value, an interrupt is generated to the MCU. This interrupt is automatically cleared when the number of empty bytes in the Tx FIFO equals or falls below the specified Tx interrupt threshold value.

The SIM card controller also includes a 16 byte Rx FIFO for data receiving. It uses a pointer to monitor the number of bytes in the Rx FIFO that have not yet been read by the MCU. The MCU reads the SIM_STS1 register bit 0 to 4 to access the pointer.
The SIM card controller also uses a configurable Rx interrupt threshold. Which can be configured by writing to the SIM_CTL1 register bit 0 to 3. This field specifies the number of unread data that should be available in the Rx FIFO before issuing an interrupt. Whenever the number of unread data exceeds this value, an interrupt is generated to the MCU. This interrupt is automatically cleared when the number of unread data in the Rx FIFO equals or falls below the specified Rx interrupt threshold value.

4.14.7 Retransmission

The SIM card controller supports retransmission upon detecting an error condition. The SIM card controller checks the I/O line from the SIM card 11 bits after the start bits leading edge. If the detected I/O is a zero (error ACK), it assumes an error occurred and retransmits the byte. If the error ACK signal is repeated for the programmable number of times specified in the SIM_SHE register, the SIM card interface sets the bad Tx parity error bit in the status register (SIM_STS0 bit 3) and issues an interrupt.

For the transmission from the SIM card controller to the SIM card, when the SIM card controller detects a parity error following the transmission of a data byte, it performs the following sequence:

- The SIM card interface retransmits the data.
- If the retransmission succeeds, it ignores the initial failure.
- If the retransmission fails for the number of times specified in the programmable SIM_SHE register, the SIM card interface sets the bad parity bit interrupt the SIM_STS0 register, issues an interrupt to the MCU, and stops retransmission of the byte.

4.14.8 SIM Power Supply

VSIM is the power supply to the SIM card. During the activation/deactivation process, VSIM output is automatically controlled. The on/off of VSIM can also be controlled by directly programming the SIM_CTL0 register bit 7.

4.14.9 Unresponsive Card Detection

The SIM interface can detect an unresponsive card by means of a watchdog timer function, which determines the maximum allowable time that a data byte should take to arrive from the SIM card. The MCU can configure the watchdog timer by programming the SIM_WDT register. The watchdog timer can function in two modes: auto mode and single mode. In the auto mode, the watchdog timer is continuously enabled. In the single mode, the watchdog timer disables itself when a data byte is received from the card, or after it has timed out.

4.14.10 SIM Interface Watchdog

SC8800 provides two separate watchdogs.

- One watchdog is used to report Rx idle for a preprogrammed time.
- The other watchdog is for time out when RX retransmission is on error.

4.14.11 SIM Interface Separate Timing Control

SC8800 provides separate control registers and counters for Rx and Tx. This provides more timing control flexibility.

4.14.12 SIM Connection

It is recommended to connect the SIM card as follows.

- Connect a 2.2 uF capacitor at VSIM.
- Connect a 10 kO resistor from SIMDA pin to VSIM.
- The driving strength setting on SIMRST, SIMCLK and SIMDA should be 00 (the default).

4.14.13 SIM Card Interface Registers

SIM Base Address: 0x8500_0000

Note1: Although there is no register change in the SC8800, the error handling function need to be tested and by enable the watchdog in SC8800, RX retransmission hang for certain SIM card which lack of support of error retransmission should not happen.

Address	Signal	Bit Pos	Default	Description

Address	Signal	Bit Pos	Default	Description
0x0000	SIM_TX	[7:0]	N/A **Write Only**	Writing to this reg will send data to tx fifo and then the data get transmitted.
0x0004	SIM_RX	[7:0]	N/A **Read only**	Read from this address retrieve data from rx fifo.
0x0008	SIM_STS0	[15:0]	**Read/Write**	
	Rx_fifo_full	[0]	0	rx fifo data number bigger than rx_int_mark
	Tx_fifo_empty	[1]	1	tx fifo data number bigger than tx_int_mark
	Rx parity error	[2]	0	Int status bit for rx parity error
	Tx parity error	[3]	0	Int status bit for tx parity error
	Rx_tout/Unresp_card	[4]	0	Int status bit for rx time out or card unresponsive
	Card_in	[5]	0	Int status bit for card inserted
	Card_out	[6]	0	Int status bit for card removed
	Early answer to reset	[7]	0	Int status bit for early answer to reset
	Active done	[8]	0	Int status bit to show activation is done
	Unused	[15:9]		
0x000C	SIM_STS1	[15:0]	**Read/Write**	
	Rx_fifo_count	[4:0]	0	Rx fifo data count, bit4 used for debug only.
	Tx_fifo_count	[9:5]	0	Tx fifo data count, bit9 used for debug only.
	Card_inserted	[10]	0	Reflect of card in input pin
	Sim_data	[11]	0	Reflect of sim data io pin
	Sim_active_status	[12]	0	Activation status, 1: activated. 0: not activated.
	Active_on	[13]	0	Busy in activation process
	Deactive_on	[14]	0	Busy in deactivation process
	Unused	[15:7]		
0x0010	SIM_IE	[15:0]	**Read/Write**	
	Rx_full_ie	[0]	0	Enable bit for rx full int
	Tx_empty_ie	[1]	0	Enable bit for tx empty int
	Rx_parity_err_ie	[2]	0	Enable bit for rx_parity_error
	Tx_parity_err_ie	[3]	0	Enable bit for tx_parity_error
	Unresp_card_ie	[4]	0	Enable bit for unresp_card
	Card_in_ie	[5]	0	Enable bit for card_in
	Card_out_ie	[6]	0	Enable bit for card_out
	Early_atr_ie Active_done_ie	[7] [8]	0 0	Enable bit for early answer to reset Enable bit for active done
	Unused	[15:8]		
0x0014	SIM_ICLR	[8:0]	**Write Only**	
	Rx_full_iclr	[0]	0	Int clear bit for rx full int
	Tx_empty_iclr	[1]	0	Int clear bit for tx empty int
	Rx_parity_err_iclr	[2]	0	Int clear bit for rx_parity_error

Address	Signal	Bit Pos	Default	Description
	Tx_parity_err_iclr	[3]	0	Int clear bit for tx_parity_error
	Unresp_card_iclr	[4]	0	Int clear bit for unresp_card
	Card_in_iclr	[5]	0	Int clear bit for card_in
	Card_out_iclr	[6]	0	Int clear bit for card_out
	Early_atr_iclr	[7]	0	Int clear bit for early answer to reset
	Active_done_iclr	[8]	0	Int clear bit bit for active done
	Unused	[15:9]		
0x0018	SIM_CTL0	[15:0]	**Read/Write**	
	Bit_convention	[0]	0	Bit convention: 0: MSB (bit 7) transmitted first 1: LSB (bit 0) transmitted first
	Logic_level	[1]	0	0: high logic level represet " 0" 1: high logic level represet " 1"
	Odd_parity	[2]	0	0: Even Parity 1: Odd Parity
	Loopback_mode	[3]	0	Transmit data looped back to receive.
	Rx_fifo_Reset	[4]	0	Reset the rx fifo
	Tx_fifo_reset	[5]	0	Reset the tx fifo
	Sim_reset	[6]	0	Reset the sim card module
	Power_enable	[7]	0	Enable the Power supply to sim card
	Card_out_latched	[8]	0	This bit will be 1 if card_out happened.
	Rx_enable	[9]	0	Enable the sim rx
	Tx_enable	[10]	0	Enable the sim tx
	do_act	[11]	0	
	do_deact	[12]	0	
	Active_deactive_en	[13]	0	Enable the active/deactive procedure
	Auto_active_deactive	[14]	0	Enable the auto start of active/deactive when card_in or card_out happen.
	Tx_data_out_low	[15]	0	Force the tx data to low logic level.
0x001C	SIM_CTL1	[15:0]	**Read/Write**	
	Rx_int_mark Tx_int_mark	[3:0] [7:4]	0 0	Receive_Int_WaterMark Tranmit_Int_WaterMark
	Clock_mode	[10:8]	0	Select the sim clk speed. The sim clk is divided From the arm bus clock: 0: APB CLK/2 1: APB CLK/4 2: APB CLK/8 3: APB CLK/16 4: APB CLK/32 5: APB CLK/64 6: APB CLK/128 7: APB CLK/256
	Clock_enable	[11]	0	Enable for the sim clk output
	Clock_polarity	[12]	0	Set the polarity of the sim_clk signal when the sim clk is stopped.
	Ad_speed_ctl	[15:13]	0	Control the speed of the active/deactive sequence.

Address	Signal	Bit Pos	Default	Description
				0: slowest, 7: fastest.
0x0020	SIM_CK_dvd	[15:0]	**Read/Write**	
	Sim_clock_dividor		0x5952	Clock divider bit 0 to 15
0x0024	SIM_SHE	[15:0]		
	Rx_retrx_lmt	[3:0]	0	Rx retransmit limit
	Tx_retrx_lmt	[7:4]	0	Tx retransmit limit
	Unused	[15:8]		
0x0028	SIM_TGC	[15:0]	**Read/Write**	
	Guard_time	[7:0]	0x30	Times in bit unit between the consecutive byte during data transmission to the SIM card.
	Turnaround_guard_time	[15:8]	0x30	Times in bit unit between the rx and tx.
0x002C	SIM_WDT	[15:0]	**Read/Write**	
	Watch_dog_trigger	[0]	0	Trigger the start of watch dog timer
	Watch_dog_repeat_en	[1]	1	Enable the timer repeat mode When this bit set 1, the watch dog will continuously check the if the receive is idle longer than wathch dog counter defined time. If it is, unresponsive card interrupt will be triggered. When this bit set 0, the above monitor action will be done only one time.
	Watch_dog_count_limit	[15:2]	0x200	Watch dog timer limit
0x0030	SIM_INT_M	[15:0]	**Read/Write**	
	Rx_full_masked	[0]	0	Masked int bit for rx full int
	Tx_empty_masked	[1]	0	Masked int bit for tx empty int
	Rx_parity_err_masked	[2]	0	Masked int bit for rx_parity_error
	Tx_parity_err_masked	[3]	0	Masked int bit for tx_parity_error
	Unresp_card_masked	[4]	0	Masked int bit for unresp_card
	Card_in_masked	[5]	0	Masked int bit for card_in
	Card_out_masked	[6]	0	Masked int bit for card_out
	Early_atr_masked	[7]	0	Masked int bit for early answer to reset
	Active_done_masked	[8]	0	Masked int bit for active done
	Unused	[15:8]		
0x0034	SIM_CTL2 Sim_sel	[15:0] [0]	**Read/Write** 0	 1'b0: sim card 0 1'b1: sim card 1

4.15 Two-Wire Serial Interface

This serial port is on the MCU side and consists of the following pins.

- SDA, data.
- SCL, clock.

These two pins are also shared with a 3-wire serial port.

4.15.1 Features

The two-wire serial interface on SC8800 is a two-wire, bi-directional serial bus that provides a simple and efficient method of data exchange between devices. It is most suitable for applications requiring occasional communication over a short distance between many devices.

In general the two-wire interface defines 3 transmission speeds:

- Normal: 100 kbps
- Fast: 400 kbps
- High speed: 3.5 Mbps

Only normal and fast modes are supported on SC8800. Another limitation is that only single master operations are supported on SC8800.

The main features include the following.

- Software programmable clock frequency
- Software programmable acknowledge bit
- Interrupt driven data-transfers
- Start/Stop/Repeated Start/Acknowledge generation
- Supports Clock Stretching/Wait state generation
- Single Master Operation
- 8 word buffer mode support

4.15.2 Two-Wire Interface Connection

The two-wire serial interface uses a serial data line (SDA) and a serial clock line (SCL) for data transfers. All devices connected to these two signals must have open drain or open collector outputs. Both lines must be pulled-up to VCC by resistors. To reduce the number of system components, SC8800 provides the pull-up resistors on-chip.

4.15.3 Clock Divider 0 & 1 Registers

This register is used to scale the SCL clock line. Due to the structure of the two-wire serial interface, the controller uses a 4*SCL clock signal internally. The clock divider register must be programmed to this 4*SCL bit rate. Change the value of the clock register only when the '2ws_en' bit is cleared. Otherwise, the register change will be ignored. 2ws_dvdr0 is the lower 16 bits of the divider and 2ws_dvdr1 is the higher 10 bit (bit 0 ~ bit 9 map to clock divider bit 16 ~ bit 25).

Example: CLK_I = 32 MHz, and the desired SCL = 100 KHz.

$$\text{Prescale} = \frac{32\ \text{MHz}}{4 \times 100\ \text{kHz}} = 80 = 0\text{x}50$$

Therefore, the reset value is 0x50.

4.15.4 Operation

4.15.4.1 System Configuration

The two-wire serial system uses a serial data line (SDA) and a serial clock line (SCL) for data transfers. All devices connected to these two signals must have open drain or open collector outputs. The logic AND function is exercised on both lines with external pull-up resistors.

The two-wire serial controller implemented here is a single master device; therefore it starts generating a clock as soon as it is enabled. The user should program the clock to the desired value before starting any transfers.

Data is transmitted synchronously to SCL on the SDA line on a byte-by-byte basis. Each data byte is 8 bits long. There is one SCL clock pulse for each data bit with the MSB being transmitted first. There is an acknowledge bit following each transferred byte. Each bit is sampled during the high period of SCL; therefore the SDA line may be changed only during the low period of SCL and must be held stable during the high period of SCL. A transition on the SDA line while SCL is high is interpreted as a command (START or STOP command).

4.15.4.2 Two-wire Serial Protocol

Normally, a standard communication consists of four parts:

1. START signal generation
2. Slave address transfer
3. Data transfer
4. STOP signal generation

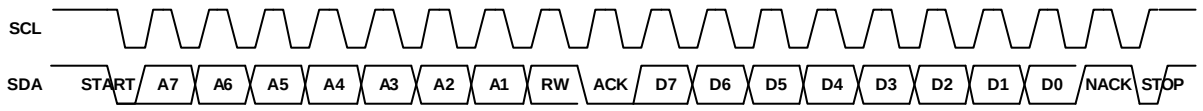


Figure 14: Two-wire serial interface protocol.

4.15.4.3 START signal

When the bus is free/idle, that is no master device is engaging the bus (both SCL and SDA lines are high), a master can initiate a transfer by sending a START signal. A START signal is defined as a high-to-low transition of SDA while SCL is high. The START signal denotes the beginning of a new data transfer. A repeated START is a START signal without first generating a STOP signal. The master uses this method to communicate with another slave or the same slave in a different transfer direction (e.g. writing to device to reading from device) without releasing the bus.

The controller generates a START signal when the start bit in the 2ws_command Register (0x86000004) is set and the read or write bit is set. Depending on the current status of the SCL line a START or Repeated START is generated.

4.15.4.4 Slave Address Transfer

The first byte of data transferred by the master immediately after the START signal is the slave address. This is a seven-bit calling address followed by a RW bit. The RW bit signals the slave data transfer direction. No two slaves in the system can have the same address. Only the slave with an address that

matches the one transmitted by the master will respond by returning an acknowledge bit by pulling the SDA low at the 9th SCL clock cycle.

The controller treats a Slave Address Transfer as any other write action. Store the slave device's address in the 2ws_command register and set the write bit. The controller will then transfer the slave address on the bus.

4.15.4.5 Data Transfer

Once successful slave addressing is achieved, the data transfer can proceed on a byte-by-byte basis in the direction specified by the RW bit sent by the master. Each transferred byte is followed by an acknowledge bit on the 9th SCL clock cycle. If the slave signals a No Acknowledge (NACK), the master can generate a STOP signal to abort the data transfer or generate a repeated START signal and start a new transfer cycle.

If the master, as the receiving device, does not acknowledge the slave, the slave releases the SDA line for the master to generate a STOP or repeated START signal.

For writing data to a slave, store the data to transmit in the 2ws_command register and set the write bit. For reading data from a slave, set the read bit. When the transfer is done, an interrupt is generated to the MCU. The 2ws_command bits 8 to 15 contain valid data. The user may issue a new write or read command at this time.

4.15.4.6 STOP signal

The master can terminate the communication by generating a STOP signal. A STOP signal is defined as a low-to-high transition of SDA while SCL is at logical '1'.

4.15.5 Arbitration Procedure

Since the two-wire serial controller supports single master configurations only, no Arbitration logic is added to the controller. Only clock synchronization is supported since slave devices can use this mechanism for clock stretching.

4.15.5.1 Clock Synchronization

Since the logical AND function is performed on the signals, a high to low transition on SCL or SDA affect all devices connected to the bus. The SCL clock signal can be synchronized between multiple masters using this feature. Each device starts counting its SCL low period when the current master drives SCL low. Once a device's clock has gone low, it holds the SCL line low until the clock high state is reached.

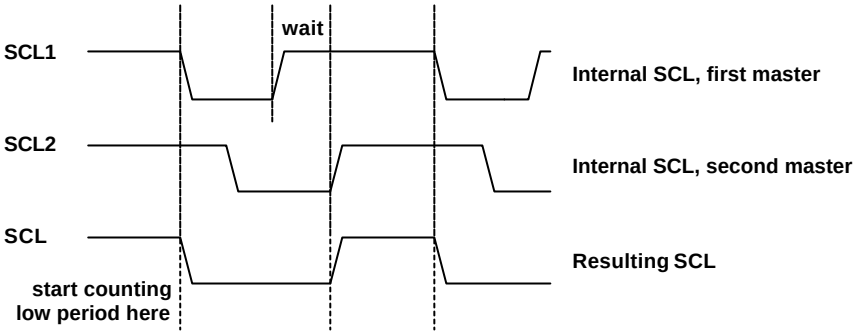


Figure 15: Clock synchronization on the two-wire serial bus.

4.15.5.2 Clock Stretching

Slave devices can use the clock synchronization mechanism to slow down the transfer bit rate. After the master has driven SCL low the slave can drive SCL low for the required period and then release it. If the slave's SCL low period is greater than the master's SCL low period, the resulting SCL bus signal low period is stretched, thus inserting wait-states.

4.15.6 Programming Examples

4.15.6.1 Example 1

Write 1 byte of data to a slave.

Slave address = 0x51 (b"1010001")
Data to write = 0xAC

Two-wire Serial Sequence:
1) Generate start command
2) Write slave address + write bit
3) Receive acknowledge from slave
4) Write data
5) Receive acknowledge from slave
6) Generate stop command

Commands:
1) Write 0xA225 to 0x86000004 (address + start bit + write cmd + iclr bit).
-- Wait for interrupt --
2) Read 2ws_ack bit from 0x86000004, should be '0'.
Write 0xAC15 to 0x86000004 (data + stop bit + write cmd + iclr bit)
-- Wait for interrupt --
3) Read 2ws_ack bit from 0x86000004, should be '0'.

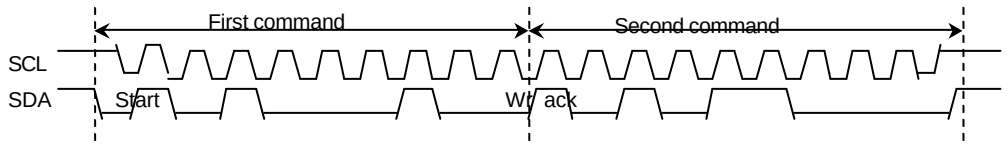


Figure 16: Two-wire serial port write example.

4.15.6.2 Example 2

Read a byte of data from a peripheral memory device.

Slave address = 100_1110
Memory location to read from = 0x20

Two-wire Serial sequence:
1) Generate start signal
2) Write slave address + write bit
3) Receive acknowledge from slave
4) Write memory location
5) Receive acknowledge from slave
6) Generate repeated start signal
7) Write slave address + read bit
8) Receive acknowledge from slave
9) Read byte from slave
10) Write no acknowledge (NACK) to slave, indicating end of transfer
11) Generate stop signal

Commands:
1) Write 0x9C25 to 0x86000004 (slave address (W bit) + start bit + write bit + iclr bit).
-- Wait for interrupt --
2) Read 2ws_ack bit from 0x86000004, should be '0'.
   Write 0x2005 to 0x86000004 (memoery address + write bit + iclr bit).
-- Wait for interrupt --
3) Read 2ws_ack bit from 0x86000004, should be '0'.
   Write 0x9D25 to 0x86000004 (slave address(R bit) + start bit + write bit + iclr bit).
-- Wait for interrupt --
4) Write 0x001B to 0x86000004 (stop bit + read bit + tx_ack + iclr bit).
5) Read 0x86000004 to get the read data D7 to D0;

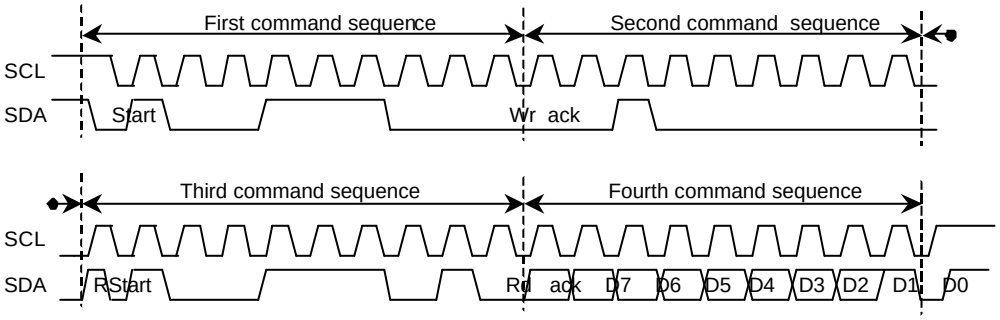


Figure 17: Two-wire serial port read example.

4.15.7 Two Wire Serial Port Control Registers

Two Wire Serial Port Base Address: 0x8600_0000

Address	Signal	Bit Pos	Default	Description

Address	Signal	Bit Pos	Default	Description
0x0000	2ws_control	[15:0]	**Read/Write**	
	2ws_int	[0]	0	2ws interrupt
	2ws_ack	[1]	0	2ws received ack value
	2ws_busy	[2]	0	2ws data line value
	2ws_ie	[3]	0	2ws interrupt enable
	2ws_en	[4]	0	2ws module enable
	2ws_cmdbuf_en	[5]	0	Enable the command buffer mode
	2ws_cmd_exec	[6]	0	Start to exec the command in the command buffer
	2ws_st_cmdbuf	[9:7]	0	The state of 2ws command buffer state machine.
	2ws_cmdbuf_wptr	[9:7]	0	2ws command buffer write pointer
0x0004	2ws_command	[15:0]	**Read/Write**	
	Int_ack	[0]	0	2ws interrupt clear bit
	Tx_ack	[1]	0	2ws transmit ack that need to be send
	Cmd_write	[2]	0	2ws write command
	Cmd_read	[3]	0	2ws read command
	Cmd_stop	[4]	0	2ws stop command
	Cmd_start	[5]	0	2ws start command
	2ws_ack	[6]	0	2ws received ack value
	2ws_busy	[7]	0	2ws busy in exec commands
	2ws_data	[15:8]	0	2ws data received or data need to be transmitted
0x0008	2ws_dividor0	[15:0]	**Read/Write**	2ws clock dividor [15:0], default is 0x40
0x000C	2ws_dividor1	[15:0]	**Read/Write**	2ws clock dividor [25:16], default is 0x0
0x0010	2ws_rst	[15:0]	**Read/Write**	Write with bit 0 set to 1 will reset the 2ws module.
0x0014	2ws_cmd_buf	[15:0]	**Read/Write**	Write command to this register will save the command to the command buffer. Read from this register after the commands are all finished will return the results of the corresponding commands. The format of the command and the return result is the same as i2c_command register.

4.16 Keypad Interface

SC8800 provides a keypad interface that supports up to a keypad array of 5 x 6, or 40 keys and it consists of the following pins.

- KEYOUT [5:0], 6 row outputs.
- KEYIN [4:0], 4 column inputs.

No external pull-up resistors are needed when the internal pull-up at the input pads are enabled, see

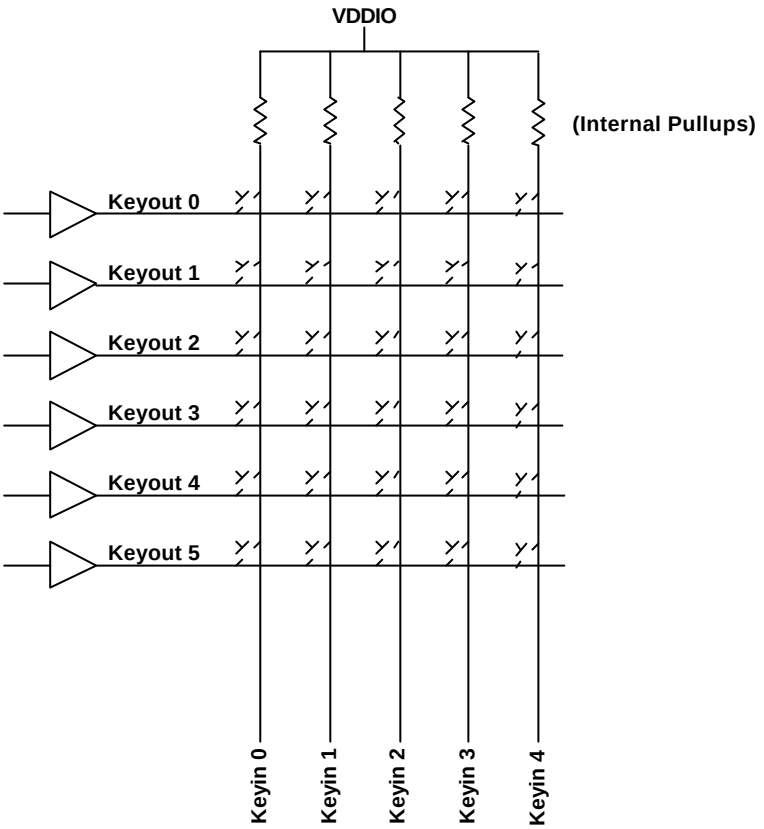


Figure 18: Keypad connection.

The keypad controller finds the key pressed by periodically scan through the key matrix. Programming the control register, which is clk_divider and clk_divider1, controls the scan speed.

The keypad control logic provides two modes of operation, non-time-out mode and time-out mode.

4.16.1 Non-Time-Out Mode

In this mode, the keypad scan circuit will periodically scan through all the keys and if a key is pressed, the scan circuit will stay at this key until the key is released or the keypad interrupt is cleared. After that, it will scan the next key.

4.16.2 Time-Out Mode

In this mode, the keypad scan circuit will periodically scan through all the keys and if a key is pressed, the scan circuit will stay at this key for a preset period of time or until the keypad interrupt is cleared and then continue to scan the next key.

Thus, in Time-Out Mode, the software can detect 2 keys pressed at the same time while in Non-Time-Out Mode only one key press can be detected unless the software can clear the interrupt fast enough.

If a key is pressed, and the interrupt is cleared, the interrupt will be set again when the scan circuit goes back to that key. It does need more than 1 interrupt to check the key release.

The mode select bit is at register at 0x8700_0004 bit 1, and the time period before scan to the next key is set by register at 0x8700_0014, bits 15:6. The time unit is base on the scan speed while the scan speed is defined by register at 0x8700_0010 bits 15:0, and 0x8700_0014 bit 5:0.

4.16.3 Sleep Mode

During sleep, all keypad inputs are OR'ed together to generate a wakeup trigger whenever a key is pressed.

4.16.4 Keypad Registers

Keypad Base Address: 0x8700_0000

Note: power_button_int and charger_int is new in this module for sc8800.

Address	Signal	Bit Pos	Default	Description
0x0000	Keypad_sts	[15:0]	**Read only**	
	Keypad_int	[0]		Keypad interrupt
	Time_out	[1]		Time out interrupt
	Row_count	[4:2]		Row counter
	Column_count	[7:5]		Column counter
	Unused	[15:8]		
0x0004	Keypad_ctl	[15:0]	**Read/Write**	
	Keypad_ie	[0]	0	Keypad interrupt enable
	Keypad_timeout	[1]	0	Keypad timeout interrupt enable
	Keypad_enable	[2]	0	Keypad enable
	Read_iclr_en	[3]	0	When this bit is 1, keypad interrupt will be cleared by read the keypad status register.
	Row4_en	[4]	0	Enable bit for row 4
	Row5_en	[5]	0	Enable bit for row 5
	Row6_en	[6]	0	Reserved
	Row7_en	[7]	0	Reserved
	Col3_en	[8]	0	Enable bit for column 3
	Col4_en Charger_ien	[9] [10]	0 0	Enable bit for column 4 Charger interrupt enable
	Pbutton_ien	[11]	0	P Button interrupt enable
0x0008	Keypad_iclr	[15:0]	**Write only**	
	Keypad_int_iclr	[0]	0	Write 1 to this bit will clear the keypad int status bit.
	Keypad_toint_iclr Swdg_int_clr	[1] [8]	0	Write 1 to this bit will clear the keypad tout int status bit. Write 1 to this bit will clear the swdg_int.
	Spk_int_iclr	[9]	0	Write 1 to this bit will clear the speaker int status bit.
0x000C	Keypad_polarity	[12:8]	0	

Address	Signal	Bit Pos	Default	Description
	Row_polarity	[0:7]	0	Internal row output xor with this value to generate row output. This register is used to control the row output polarity.
	Column_polarity	[12:8]	0	Column input xor with this value to generate the interbal column input. This register is used to control the column input polarity.
0x0010	Clk_dividor	[15:0]	**Read/Write**	
	Clk_dividor	[15:0]	0	Clock dividor [15:0]
0x0014	Clk_dividor1	[15:0]	**Read/Write**	
	Clk_dividor1	[5:0]	0	Clock dividor [21:16]
	Tcnt_reg	[15:6]	0	Time out counter value
0x0018	Masked_int	[15:0]	**Read only**	
	Masked_kpd_int	[0]	0	Keypad interrupt masked by the interrupt enable bit.
	Masked_tout_int	[1]	0	Keypad timeout interrupt masked by the time out interrupt enable bit.
	Charger_int	[2]	0	This bit reflects the charger_int input value. Please note the charger_int input share the pin with the zbus address 2. Chip pin control register GPIO_ZA[5:4} need to be 0 to select charger_int. It is selected by default.
	pbutton_int	[3]	0	This bit reflects the pbutton_int input value. Please note the pbutton_int input share the pin with the zbus address 3. Chip pin control register GPIO_ZA[5:4} need to be 0 to select charger_int. It is selected by default.
	swdg_int	[4]		swdg_int

4.17 32-bit System Counter

The system counter is a free-running counter and it counts whenever the RTC clock is running. The counter gets reset to zero at the hardware reset.

When the system counter count equals to the value of SYST_ALM register (system counter alarm, at 0x8B00_0040), an interrupt is generated. The interrupt length is one cycle of the RTC clock, or about 30.5 µs at 32.768 kHz. Before sent to the MCU, the rising edge of this interrupt is latched by RTC clock, the interrupt status shows on register sys_ctl bit [1]. The MCU is able to clear it through register sys_ctl bit [3].

4.17.1 32 bit System Counter Registers

System Counter Base Address: 0x8700_0000

Address	Signal	Bit Pos	Default	Description

Address	Signal	Bit Pos	Default	Description
0x001C	System_count0	[15:0]	**Read only**	The lower 16 bit of the system counter
0x0020	System_count1	[15:0]	**Read only**	Then higher 16 bit of the system counter
0x0024	Syst Control	[15:0]		
		[0]	**Read/Write**	Interrupt enable
		[1]	**Read only**	Interrupt status

Please note:

- The tick of the system counter is selected by bit 14 of the global register GEN1 (address 0x8B00_0018). Please refer to the global register section.
- Some reordering is done for this 32-bit system counter and thus this need to the tested as a new module.

4.18 Pulse Width Modulation Outputs

SC8800 provides two PWM outputs that can be used to control a ringer or buzzer or LCD lights. Up to 5 PWM outputs can be supported.

SC8800 generates up to 5 PWM output signals from the MCU side, PWM_A, PWM_B, PWM_C, PWM_D, and PWM_E. PWM_A is the combination of two identical but separately programmable PWM circuits, PWM0 and PWM1. The outputs from the two circuits are toggled at a rate of the PWM0 pre-scaled clock rate. By only enable the PWM0 or PWM1, the PWM_A can also be the PWM0 or PWM1 output. PWM_B ~ PWM_E are the output directly from the PWM 2 ~ 5. Figure 19 shows the organization of PWM module.

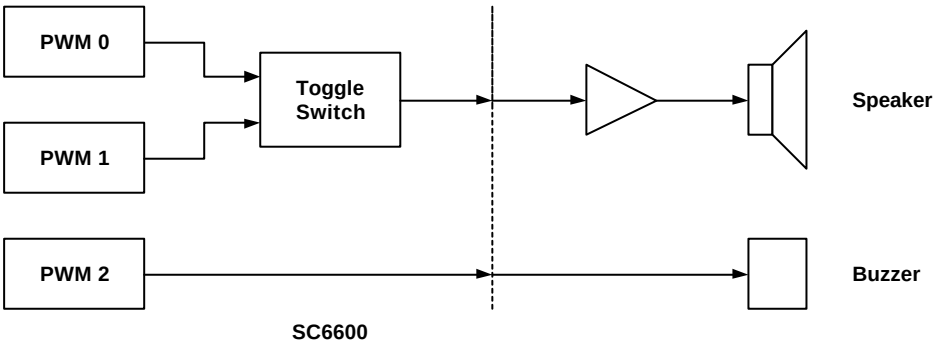


Figure 19: PWM module block diagram.

The operation of each of the PWM circuits is as described below.

The PWM uses the MCU peripheral bus clock to generate two pulse trains:

- A low frequency pulse train programmable to around 200 – 500 Hz. This is the Tone signal.
- A high frequency pulse train around 40 kHz with a variable duty cycle. This is the Mod signal.

The two pulse trains are mixed to produce the PWM output. The frequency of this output is the frequency of the Tone; the volume is dependent on the duty cycle of the Mod. The frequency of the Tone and the duty cycle of the Mod are programmable so that complex tones can be generated.

A PWM functional block diagram is shown in Figure 20.

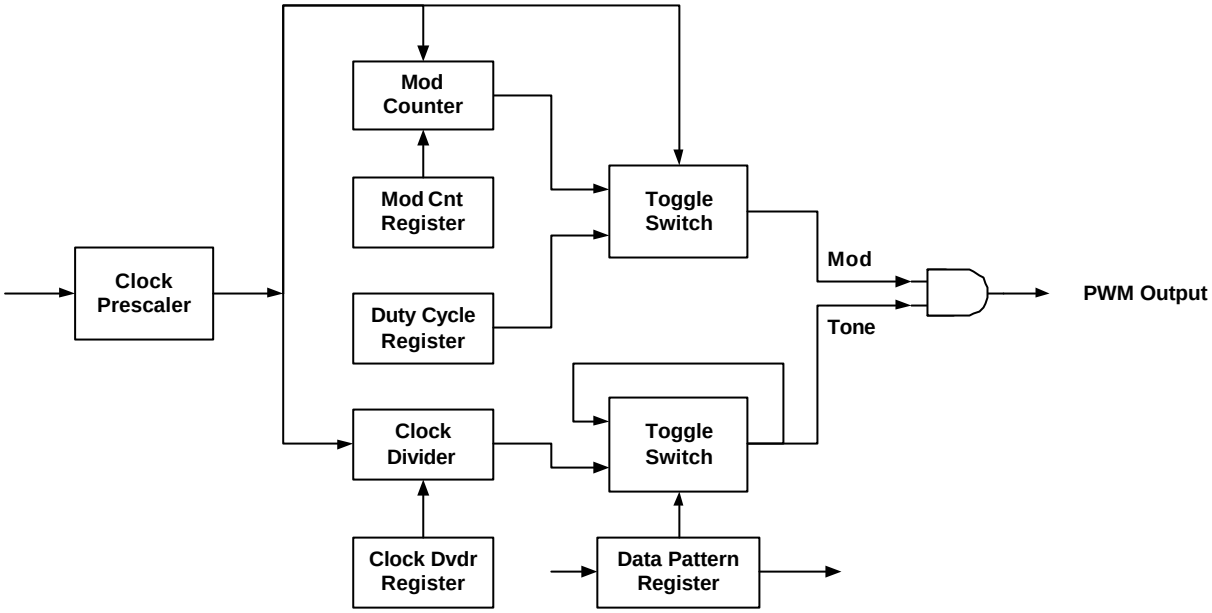


Figure 20: PWM module functional block diagram.

The Mod counter circuit is an 8-bit counter that is clocked around 4 MHz, which is the frequency of pre-scaled clock. The counter counts to the value set in the counter register and is then reset to 0. The default value of the counter register is decimal 100. The duty cycle register is a programmable 8-bit register. The duty cycle for the Mod is written to this register. The comparator function compares the counter output with the value of the duty cycle register. The output from the comparator is determined by the following rule:

If Counter Value <= Duty Cycle Register, Comparator Output = 1, else Comparator output = 0.

The output from the comparator is the Mod signal. The duty cycle of the Mod signal is programmable by writing to the duty cycle register and to the Mod counter register. If the Mod counter register default value is used (decimal 100) the frequency of the Mod signal is pre-scaled clock /100.

The Clock Divider Register is a programmable, 10-bit register. The divider circuit divides down the pre-scaled clock input by the value of the register to yield an output signal. The output from the divider is used to clock the Data Pattern Shift Register.

The Data Pattern Shift Register is a 32-bit serial shift register. The contents of the programmable Data Pattern Latch register are downloaded to the Data Pattern Shift Register to set the initial value of the shift register.

The output from the shift register is the time pulse train, Tone. This output is also fed back to the register input so that the same series of 32 bits are continuously cycled through the register. The data pattern is cycled at a frequency of (pre-scaled clock)/(clock divider)/32.

The Mod and Tone pulse trains are ANDed to generate the output from the PWM circuit.

4.18.1 PWM Registers

PWM Base Address: 0x8800_0000

Address	Signal	Bit Pos	Default	Description
0x0000	Pwm0_prescale	[15:0]	**Read/Write**	
	Clk_prescaler	[7:0]		Pwm 0 Clock prescaler
	Pwm_en0	[8]		Pwm 0 enable
0x0004	Pwm0_cnt	[15:0]	**Read/Write**	
	Cnt_reg0	[7:0]		Pwm 0 mod counter
	Duty_reg0	[15:8]		Pwm 0 mod duty cycle
0x0008	Pwm0_dvd	[15:0]	**Read/Write**	
	Dvd_reg0	[15:0]	0	Pwm 0 tone dividor
0x000C	Pattern_reg0_15_0	[15:0]	**Read/Write,** default 0	Pwm 0 pattern register low bits
0x0010	Pattern_reg0_31_16	[15:0]	**Read/Write,** default 0	Pwm0 pattern register high bits
0x0014	Pwm1_prescale Clk_prescaler	[15:0] [7:0]	**Read/Write**	Pwm 1 Clock prescaler
	Pwm_en1	[8]		Pwm 1 enable
0x0018	Pwm1_cnt	[15:0]	**Read/Write**	
	Cnt_reg1	[7:0]		Pwm 1 mod counter
	Duty_reg1	[15:8]		Pwm 1 mod duty cycle
0x001C	Pwm1_dvd	[15:0]	**Read/Write**	
	Dvd_reg1	[15:0]	0	Pwm 1 tone dividor
0x0020	Pattern_reg1_15_0	[15:0]	**Read/Write,** default 0	Pwm 1 pattern register low bits
0x0024	Pattern_reg1_31_16	[15:0]	**Read/Write,** default 0	Pwm 1 pattern register high bits
0x0028	Pwm2_prescale	[15:0]	**Read/Write**	
	Clk_prescaler	[7:0]		Pwm 2 Clock prescaler
	Pwm_en2	[8]		Pwm 2 enable
0x002C	Pwm2_cnt	[15:0]	**Read/Write**	
	Cnt_reg2	[7:0]		Pwm 2 mod counter
	Duty_reg2	[15:8]		Pwm 2 mod duty cycle
0x0030	Pwm2_dvd	[15:0]	**Read/Write**	
	Dvd_reg2	[15:0]	0	Pwm 2 tone dividor
0x0034	Pattern_reg2_15_0	[15:0]	**Read/Write,**	Pwm 2 pattern register low bits

Address	Signal	Bit Pos	Default	Description
			default 0	
0x0038	Pattern_reg2_31_16	[15:0]	**Read/Write, default 0**	Pwm 2 pattern register high bits

Please note:

- There are 3 PWM modules instantiation. Only 2 output is going out of the chip. The first output is a combination of PWM0 and/or PWM1. The second output is PWM3.

4.19 Real Time Clock

The MCU sub-system provides real time clock support. The real time clock (RTC) is used to count seconds, minutes, hours and days. Together with the MCU software, a real-time calendar can be implemented. The RTC is also used in sleep mode to reduce system power consumption.
The real time clock is generated from an external 32.768 kHz crystal with a separate power supply, e.g., a button battery.

The initial values of the counters are random and the MCU can set the counters through the control registers. At the intervals of seconds, minutes, hours or days, the RTC block can generate interrupt signals to the MCU, if enabled. This feature can be used as general-purpose timers. The minute interrupt is connected to GPIO [24][1].

The RTC block also provides an alarm function. The MCU can set values in the alarm registers. When the preset values are reached, an interrupt will be generated and sent to the MCU.

The day counter has 16 bits and therefore can count for 65,536 days.

4.19.1 Real Timer Counter Registers

RTC Base Address: 0x8900_0000

Address	Signal	Bit Pos	Default	Description
0x0000		[15:0]	**Read only**	
	Second_count	[5:0]	N/A	Current second count
0x0004		[15:0]	**Read only**	
	Minute_count	[5:0]	N/A	Current minute count
0x0008		[15:0]	**Read only**	
	Hour_count	[4:0]]	N/A	Current hour count
0x000C		[15:0]	**Read_only**	
	Day_count	[7:0]	N/A	Current hour count
0x0010		[15:0]	**Write only**	
	Second_update	[5:0]	0	Write to update second count

[1] This only on SC8800A2.

Address	Signal	Bit Pos	Default	Description
0x0014		[15:0]	**Write only**	
	Minute_update	[5:0]		Write to update minute count
0x0018		[15:0]	**Write only**	
	Hour_update	[4:0]	N/A	Write to update hour count
0x001C		[15:0]	**Write only**	
	Day_update	[7:0]	N/A	Write to update day count
0x0020		[15:0]	**Read/Write**	
	Second_alm	[5:0]	N/A	Alarm second count
0x0024		[15:0]	**Read/Write**	
	Minute_alm	[5:0]	N/A	Alarm minute count
0x0028		[15:0]	**Read/Write**	
	Hour_alm	[4:0]	N/A	Alarm hour count
0x002C		[15:0]	**Read/Write**	
	Day_alm	[4:0]	N/A	Alarm day count
0x0030	Rtc_control	[15:0]	**Read/Write**	
	Sec_ie	[0]	0	Sec int enable
	Min_ie	[1]	0	Min int enable
	Hour_ie	[2]	0	Hour int enable
	Day_ie	[3]	0	Day int enable
	Alarm_ie	[4]	0	Alarm int enable
	Hour_format_sel	[5]	0	Hour format select 0: The read back hour count is formatted as 0 to 24. 1: The read back hour count is formatted as 0 to 12, and bit 4 of Hour_count represent AM (0) or PM (1).
	Rtc_enable	[6]	0	Rtc module enable
0x0034	Rtc_int	[15:0]	**Read only**	
	Sec_int	[0]	0	Second interrupt
	Min_int	[1]	0	Minute interrupt
	Hour_int	[2]	0	Hour interrupt
	Day_int	[3]	0	Day interrupt
	Alm_int	[4]	0	Alarm interrupt
			0	
0x0038	Rtc_int_clr	[15:0]	**Write Only**	
	Sec_int_clr	[0]	0	Second int is cleared by write 1 to this bit
	Min_int_clr	[1]	0	Minute int is cleared by write 1 to this bit
	Hour_int_clr	[2]	0	Hour int is cleared by write 1 to this bit
	Day_int_clr	[3]	0	Day int is cleared by write 1 to this bit
	Alm_int_clr	[4]	0	Alarm int is cleared by write 1 to this bit

Address	Signal	Bit Pos	Default	Description
0x003C	Rtc_int_masked	[15:0]	**Read/Write**	
	Sec_int_masked	[0]	0	
	Min_int_masked	[1]	0	
	Hour_int_masked	[2]	0	
	Day_int_masked	[3]	0	
	Alm_int_masked	[4]	0	

4.20 MCU GPIO

SC8800 provides up to 116 GPIO pins from the MCU side, GPIO [116:0]. However, many of the pins are multiplexed with other functions and system design trade-off must be exercised on selecting them.

All the GPIO pins can be programmed to be either input or output. When in input mode, they can be programmed to trigger interrupt to the MCU.

4.20.1 GPIO Control Registers

ARM GPIO Register Base Address: 0x8A00_0000

GPIO pin15~00 Register Base Address: 0x8A00_0000
GPIO pin31~16 Register Base Address: 0x8A00_0080[1,2]
GPIO pin47~32 Register Base Address: 0x8A00_0100
GPIO pin63~48 Register Base Address: 0x8A00_0180
GPIO pin79~64 Register Base Address: 0x8A00_0200
GPIO pin95~80 Register Base Address: 0x8A00_0280
GPIO pin111~96 Register Base Address: 0x8A00_0300
GPIO pin116~112 Register Base Address: 0x8A00_0380

Address	Signal	Bit Pos	Default	Description
0x00	GPIODATA	**[15:0]**	0	**GPIO data register**
0x04	GPIODMSK	**[15:0]**	0	GPIO data mask register, GPIO pin can be read and write if the mask bit is “1”
0x08	GPIODIR	**[15:0]**	0	**“ 1” configure pin to be output** “0” configure pin to be input
0x0C	GPIOIS	**[15:0]**	0	Interrupt sense register. “1” detect levels, “0” detect edges
0x10	GPIOIBE	**[15:0]**	0	Interrupt both edges register. “1” both edges trigger an interrupt, “0” interrupt generation event is controlled by GPIOIEN
0x14	GPIOIEV	**[15:0]**	0	Interrupt event register, “1” rising edges or high levels trigger interrupts, “0” falling edges or low levels trigger interrupts.

[1] GPIO [24] is used for minute interrupt on SC8800A2.
[2] GPIO [25] is used for charger interrupt on SC8800A2.

Address	Signal	Bit Pos	Default	Description
0x18	GPIOIE	**[15:0]**	0	Interrupt mask register, "1" corresponding pin is not masked. "0" corresponding pin interrupt is masked
0x1C	GPIORIS	**[15:0]**	0 Read only	Row interrupt status, reflect the status of interrupts trigger conditions detection on pins (prior to masking). "1" interrupt condition met "0" condition not met
0x20	GPIOMIS	**[15:0]**	0 Ready only	Masked interrupt status, "1" Interrupt active "0" interrupt not active
0x24	GPIOIC	**[15:0]**	0 Write only	Interrupt clear, "1" clears edge detection interrupt. "0" has no effect.
0x40	GPIOITCR	**[0]**	0	Integration test control register, "1" In integration test mode "0" Normal mode
		[15:1]		Reserved
0x44	GPIOITOP1	**[15:0]**	0 Write only	Integration test input read/set register. In integration test mode, writes specify the value to be driven on the GPIOMIS lines. Read GPIOMIS checks interrupt active.
0x48	GPIOITOP2	**[0]**	0 Write only	Integration test input read/set register. In integration test mode, reads return the value of GPIOINTR.
		[15:1]		Reserved

4.21 Global Registers

ARM Global Register Base Address: 0x8B00_0000

Address	Signal	Bit Pos	Default	Description
0x8B00_0000	ADCC	**[6:0]**	**0x00**	adc_ctl: It changes when adc_ctl is zero, only the data with all zeros or one bit one, other bits are zeros, can be written into this register. If all input bits are one, this register is set to zero.
		[7]		adcal_bypass
		[8]	0x00	test_ctl, bit8: test_en, enable ADC test.
		[13:9]	0x00	**auxadcs1_a:** **1: the relative source voltage is through a resistance network.** 0: otherwise
		[20:14]	0x00	auxadcs_a: Must be **one-hot** code to select a source voltage
		[31:21]		
0x8B00_0004	ADCR	[9:0]	adc_rslt Read only	adc_rslt
		[15:10]		

Address	Signal	Bit Pos	Default	Description
0x8B00_0008	GEN0	[0]	0	clksmem_div_sel 1'b0: clksmem = clk_ahb/2 1'b1: clksmem = clk_ahb
		[1]	1	mcu_dsp1_rst
		[2]	1	timers_eb²: enable timers
		[3]	1	sim_eb: enable sim
		[4]	1	i2c_eb: enable i2c
		[5]	1	**pwm_b_eb: enable pwm_b**
		[6]	1	pwm_a_eb: enable pwm_a
		[7]	1	rtc_eb:
		[8]	1	kpd_eb
		[9]	0	opllbpsp
		[10]	1	mcu_dsp2_rst
		[11]	0	mcu_soft_rst, ARM set this bit, after delay 15 clocks, this bit changes to low
		[12]	1	timers_rtc_eb
		[13]	0	dsp_start_0
		[14]	0	mcu_sleep_en, cleared by signal mcu_wakeup_pls
		[15]	0	Sys_pd: system power down
		[19:16]	0	clk_smem_dly_sel
		[23:20]	0	clk_dmem_dly_sel
		[31:24]	0	
0x8B00_000C	PCTL	[0]	pad_dsp_ctl[0]	clk_all_en. When this bit is 1, all clock gatings are disabled. Default is 0
		[4:1]	pad_dsp_ctl[4:1]	clk_arm9_div[3:0]. MCU clock divider. If clk_arm9_div = N, the MCU clock frequency is the PLL output frequency divided by (N+1). Default is 4'b0100
		[8:5]	pad_dsp_ctl[8:5]	clk_dsp1_div[3:0]. Teak1 clock divider. If clk_dsp1_div = N, the DSP clock frequency is the PLL output frequency divided by (N+1). Default is 4'b0011
		[9]	pad_dsp_ctl[9]	tk1_test_mode (testmodep). This is connected to all Teak1 related clock-gating cells. When this bit is 1, all those clock gatings are disabled. Default is 0

Address	Signal	Bit Pos	Default	Description
		[15:10]	pad_dsp_ctl[15:10]	tk1_strapping[5:0], Teak1 strap bit. [0] = INTP. Internal program mode. Default = 1. [1] = TESETP. Test mode. Default = 0 [2] = BOOTP. Boot mode, i.e., start from address 0xFFFE. Default = 1 [3] = ZRPDWRPOL. Z R/W polarity. Default = 1. Active low. [4] = ZSTRBPOL. Z strobe polarity. Default = 1. Active low. [5] = XSTRBPOL. X strobe polarity. Default = 1. Active low.
		[19:16]	pad_dsp_ctl[19:16]	clk_dsp2_div[3:0]. Teak2 clock divider. If clk_dsp2_div = N, the DSP clock frequency is the PLL output frequency divided by (N+1). Default is 4'b0011
		[20]	pad_dsp_ctl[20]	tk2_test_mode (testmodep). This is connected to all Teak2 related clock-gating cells. When this bit is 1, all those clock gatings are disabled. Default is 0
		[26:21]	pad_dsp_ctl[26:21]	tk2_strapping[5:0], Teak2 strap bit. [0] = INTP. Internal program mode. Default = 1. [1] = TESETP. Test mode. Default = 0 [2] = BOOTP. Boot mode, i.e., start from address 0xFFFE. Default = 1 [3] = ZRPDWRPOL. Z R/W polarity. Default = 1. Active low. [4] = ZSTRBPOL. Z strobe polarity. Default = 1. Active low. [5] = XSTRBPOL. X strobe polarity. Default = 1. Active low.
		[27]	pad_dsp_ctl[27]	VINITHI Default is 0
		[31:28]		
0x8B00_0010	INT	[0]	0 Read only	dsp_irq: DSP set this bit by signal dsp_irq_pls
		[1]	0 Read only	dsp_frq: DSP set this bit by signal dsp_frq_pls
		[2]	0	mcu_irq: ARM set this bit, after delay 15 clocks, the bit changes to low
		[3]	0	mcu_frq: ARM set this bit, after delay 15 clocks, the bit changes to low
		[4]	0 Read only	adc_irq: The bit is set by signal ad adc_irq_pls
		[5]	0 Read only	vbc_irq: The bit is set by signal vbc_dsp_int
		[15:6]		Reserved

Address	Signal	Bit Pos	Default	Description
0x8B00_0014	ICLR	[0]	Write only	Write “1” Clear DSP_irq
		[1]		Write “1” Clear DSP_frq
		[3:2]		
		[4]		Write “1” Clear adc_irq (internal control register)
		[5]		Write “1” Clear vbc_irq (internal control register)
		[15:6]		Reserved
0x8B00_0018	GEN1	[3:0]	0x0	clk_mcu_dly_sel
		[7:4]	0x0	
		[8]	1	gea_eb[2]
		[9]	0	pllmn_we: pll_mn register write enable
		[10]	1	**dcam_eb**
		[11]	1	rempause_eb[2]
		[12]	0	testmodep_mcu[2]: gate clock disable
		[13]	0	syst_en
		[14]	0	syst_cksel 1’b0: RTC clock 1’b1: 6.5MHz clock
		[15]	0	CLK_26MHZ_EN[2], indicating that the input clock is 26 MHz.
		[19:16]	0x0	clk_dsp_dly_sel
		[20]	ReadOnly	Arm_boot_md0
		[21]	ReadOnly	Arm_boot_md1
		[22]	1	Serclk_eb0
		[23]	1	Serclk_eb1
		[24]	1	Serclk_eb2
		[25]	1	wdg_enable
		[26] [27]	1 1	dma_eb gpio_eb
		[28]	1	wdg_eb
		[30:29]	0	Analog_out_sel
		[31]	0	pdbg
0x8B00_001C				
0x8B00_0020	HWRST[3]	[15:0]	0	The register value do not change on MCU reset, it can only be cleared by DSP reset ext_rst_b.
0x8B00_0024	PLL_MN	[31:0]	0x0041_0600	PLL frequency control [23:16] N [11:0] M PLL frequency = input frequency * M / N DSP can change this register when GEN1[9], pllmn_we high
0x8B00_0028	LDO_CTL	[0]	0	ldo_bp18
		[1]	0	ldo_bp18_rst
		[2]	0	ldo_bprf0: For LDO1 (i.e. VDDRF0 LDO)
		[3]	0	ldo_bprf0_rst: For LDO1 (i.e. VDDRF0 LDO)

Address	Signal	Bit Pos	Default	Description
		[4]	0	ldo_bprf1: For LDO2 (i.e. VDDRF1 LDO)
		[5]	0	ldo_bprf1_rst: For LDO2 (i.e. VDDRF1 LDO)
		[6]	0	ldo_bpio
		[7]	0	ldo_bpio_rst
		[8]	0	ldo_bpsmem
		[9]	0	ldo_bpsmem_rst
		[10]	0	ldo_vref27
		[11]	0	ldo_vref27_rst
		[12]	0	ldo_bppll
		[13]	0	ldo_bppll_rst
		[14]	0	ldo_bpvbo: **should be operated simultaneously with bit[16] ldo_bpvb.**
		[15]	0	ldo_bpvbo_rst: **should be operated simultaneously with bit[17] ldo_bpvb_rst.**
		[16]	0	ldo_bpvb: **should be operated simultaneously with bit[14] ldo_bpvb0.**
		[17]	0	ldo_bpvb_rst: **should be operated simultaneously with bit[15] ldo_bpvb0_rst.**
		[18]	0	ldo_bpd3v: **should be operated simultaneously with bit[20] and bit[22].**
		[19]	0	ldo_bpd3v_rst: **should be operated simultaneously with bit[21] and bit[23].**
		[20]	0	ldo_bpvbb: **should be operated simultaneously with bit[18] and bit[22].**
		[21]	0	ldo_bpvbb_rst: **should be operated simultaneously with bit[19] and bit[23].**
		[22]	0	ldo_bpaux: **should be operated simultaneously with bit[18] and bit[20].**
		[23]	0	ldo_bpaux_rst: **should be operated simultaneously with bit[19] and bit[21].**
		[24]	0	ldo_bpsim
		[25]	0	ldo_bpsim_rst
		[26]	0	fsm_ldorfbp_en: For LDO1 (i.e. VDDRF0 LDO)
		[27]	0	fsm_ldoauxbp_en
		[28]	0	fsm_ldod3vbp_en
		[29]	0	fsm_afcpd_en
		[30]	0	fsm_xtlen_en
		[31]	0	auto_b_en: For VDDCORE
0x8B00_002C	GEN2	[0]		

Address	Signal	Bit Pos	Default	Description
		[3:1]	0	ext_eoen_dly_sel
		[6:4]	0	ext_ewen_dly_sel
		[7]	0	Arm_boot_en
		[9:8]	0	Ser_sel0: “0” uart “1” Irda “2” SPI
		[11:10]	0	Ser_sel1: “0” uart “1” Irda “2” SPI
		[13:12]	0	Ser_sel2: “0” uart “1” Irda “2” SPI
		[14]	1	uart_mux_sel1
		[15]	1	uart_mux_sel2
		[18:16]	0	ext_eblw_dly_sel
		[21:19]	0	ext_ebup_dly_sel
		[24:22]	0	ext_eadv_dly_sel
		[31:25]	0	
0x8B00_0030	TK_BOOT_ADDR	[31:0]		
		[15:0]	0x0000	Reserved
		[31:16]	0x0000	Teak_boot_addr
0x8B00_0034	STC_DSP_STSTE	[15:0]	Read only	Teak2 State
0x8B00_0038	LDO_CTL2	[0]	0	ldo_core_b0
		[1]	0	ldo_core_b0_rst
		[2]	0	ldo_core_b1
		[3]	0	ldo_core_b1_rst
		[4]	0	ldo_rtc_b0
		[5]	0	ldo_rtc_b0_rst
		[6]	0	ldo_rtc_b1
		[7]	0	ldo_rtc_b1_rst
		[8]	0	ldo_sim_b0
		[9]	0	ldo_sim_b0_rst
		[10]	0	ldo_sim_b1
		[11]	0	ldo_sim_b1_rst
		[12]	0	slp_core_b0
		[13]	0	slp_core_b0_rst
		[14]	0	slp_core_b1
		[15]	0	slp_core_b1_rst
		[16]	0	swdg_md_sel2_rst
		[17]	0	swdg_md_sel2_set
		[18]	0	swdg_md_sel1_rst
		[19]	0	swdg_md_sel1_set
		[20]	0	swdg_md_sel0_rst
		[21]	0	swdg_md_sel0_set
		[22]	0	swdg_rst_en_rst
		[23]	0	swdg_rst_en_set
		[24]	0	swdg_int_en_rst
		[25]	0	swdg_int_en_set
		[26]	0	rtc_gsm_mode_rst
		[27]	0	rtc_gsm_mode_set
		[28]	0	ldo_bpdmem
		[29]	0	ldo_bpdmem_rst
0x8B00_003C	ANATST_CTL	[0]	0	start_en1u
		[1]	0	start_en1u _rst

Address	Signal	Bit Pos	Default	Description
		[2]	0	start_en2u
		[3]	0	start_en2u _rst
		[4]	0	start_en3u
		[5]	0	start_en3u _rst
		[6]	0	start_en6u
		[7]	0	start_en6u _rst
		[13:8]		
		[14]	0	recharge
		[15]	Read only	standby
0x8B00_0040	SYST_ALM	[31:0]	0x0000FFFF	syst_alm
0x8B00_0044	BUSCLK_ALM	[2:0]	0	Bus_clk_dvdr bit2~0
		[3]	0	arm_vb_acc
		[4]	0	arm_vb_anaon, enable VB clock for analog. this bit is active when arm_vb_acc is high
		[5]	0	arm_vb_adcon, enable VB ADC digital path. this bit is active when arm_vb_acc is high
		[6]	0	arm_vb_da1on, enable VB DAC0 digital path. this bit is active when arm_vb_acc is high
		[7]	0	arm_vb_da0on, enable VB DAC1 digital path. this bit is active when arm_vb_acc is high
		[31:8]		
0x8B00_0048	SYST_CK_NUM	[16:0]	0	Syst_ck_num
		[31:17]		
0x8B00_004C	A9_CSTRI_EN	[3:0]	0	a9_cstri_en
0x8B00_0050	LDO_CTL3	[31:0]		
		[0]	0	ldo_bplcd_rst
		[1]	0	ldo_bplcd_set
		[2]	0	ldo_lcd_b0_rst
		[3]	0	ldo_lcd_b0_set
		[4]	0	ldo_lcd_b1_rst
		[5]	0	ldo_lcd_b1_set
		[31:6]		
0x8B00_0054	VBCLK	[31:0]		
		[3:0]	4'd6	clk_vbc_div
		[7:4]	4'd10	clk_vbana_div
		[11:8]	4'd2	clk_vbsys_div
		[12] [31:13]	0	clk_vbana_sel

Frequency of Voice Band Codec:
The clock for the digital part of Voice Band Codec is

$$f_{vbc13} = \frac{f_{pll}}{\text{clk\_vbc\_div} * \text{clk\_vbsys\_div}}$$

The clock for the analog part of Voice Band Codec is

$$f_{VBCCLK} = \begin{cases} \dfrac{f_{pll}}{\text{clk\_vbc\_div} * \text{clk\_vbana\_div}} & (\text{if clk\_vbana\_sel is } 1) \\ 2\ \text{MHz} & (\text{if clk\_vbana\_sel is } 0) \end{cases}$$

4.22 Chip Pin Control

4.22.1 Pin Function Selection

The control bits for pad function selections are listed in the following table. A selection value of 0 selects the 1st pin function, a value of 1 selects the 2nd function, a value of 2 selects the 3rd function and a value of 3 selects the 4th function (if applicable). Some of the functions are for test and debug purposes only and therefore not described in this document.

4.22.2 Pad Driving Strength

The driving strength of digital I/O pads can be programmed through control registers from 2X to 10X. For a 6X driver, the maximum load is 20 pF for 50 MHz operations. There are two control bits and the setting are as follows.

Table 39: SC8800 I/O driving strength control.

Control Bits	Driving Strength
00	2X
01	6X
10	12X
11	24X

4.22.3 Pad Weak Pull-Up

The digital I/O pads provide a built-in weak pull-up (WPU) resister that can be activated or deactivated at the control of the MCU. When activated, the weak pull-up resistance is about 40 kΩ.

4.22.4 Chip Pin Control Registers

ARM Pin Register Base Address: 0x8C00_0000

The control bits for pad function selections are listed in the following table. A selection value of 0 selects the 1st pin function, a value of 1 selects the 2nd function, a value of 2 selects the 3rd function and a value of 3 selects the 4th function (if applicable). Please refer to the pin function list for the meanings of the functions. These registers are written only.

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0000	Dsppda_reg	Dsppda_sel	[5:4]	0	DSP data and address select: 0:DSP1 data and ddress 1:DSP2 data and ddress 2: 3:
		Dsppda_drv	[3:2]	0	DSP data and address driver

Address	Register	Signal	Bit Pos	Default	Description
		Dsppda_wpu	[1]	0	DSP data and address wpu: 0: Disable 1: Enable
		Dsppda_wpd	[0]	0	DSP data and address wpd: 0: Disable 1: Enable
0x8C00_0004	Dspdzxo_reg	Dspdzxo_sel	[5:4]	0	DSP X/Z odd address select: 0:DSP1 X/Z odd address 1:DSP2 X/Z odd address 2: 3:
		Dspdzxo_drv	[3:2]	0	DSP X/Z odd address drive
		Dspdzxo_wpu	[1]	0	DSP X/Z odd address wpu: 0: Disable 1: Enable
		Dspdzxo_wpd	[0]	0	DSP X/Z odd address wpd: 0: Disable 1: Enable
0x8C00_0008	Dspdzxe_reg	Dspdzxe_sel	[5:4]	0	DSP X/Z even address select: 0:DSP1 X/Z even address 1:DSP2 X/Z even address 2: 3:
		Dspdzxe_drv	[3:2]	0	DSP X/Z even address drive
		Dspdzxe_wpu	[1]	0	DSP X/Z even address wpu: 0: Disable 1: Enable
		Dspdzxe_wpd	[0]	0	DSP X/Z even address wpd: 0: Disable 1: Enable
0x8C00_000C	Dspxerd_reg	Dspxerd_sel	[5:4]	0	Dspxerd select: 0:DSP1 X-even RD 1:DSP2 X-even RD 2: 3:
		Dspxerd_drv	[3:2]	0	Dspxerd drive
		Dspxerd_wpu	[1]	0	Dspxerd wpu: 0: Disable 1: Enable
		Dspxerd_wpd	[0]	0	Dspxerd wpd: 0: Disable 1: Enable
0x8C00_0010	Dspxewr_reg	Dspxewr_sel	[5:4]	0	Dspxewr select: 0:DSP1 X-even WR 1:DSP2 X-even WR 2: 3:
		Dspxewr_drv	[3:2]	0	Dspxewr drive
		Dspxewr_wpu	[1]	0	Dspxewr wpu: 0: Disable 1: Enable
		Dspxewr_wpd	[0]	0	Dspxewr wpd: 0: Disable 1: Enable
0x8C00_0014	Dspxord_reg	Dspxord_sel	[5:4]	0	Dspxord select: 0:DSP1 X-odd RD 1:DSP2 X-odd RD 2: 3:
		Dspxord_drv	[3:2]	0	Dspxord drive
		Dspxord_wpu	[1]	0	Dspxord wpu: 0: Disable 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0018	Dspxowr_reg	Dspxord_wpd	[0]	0	Dspxordwpd: 0: Disable 1: Enable
		Dspxowr_sel	[5:4]	0	Dspxowr select: 0:DSP1 X-odd WR 1:DSP2 X-odd WR 2: 3:
		Dspxowr_drv	[3:2]	0	Dspxowrdrive
		Dspxowr_wpu	[1]	0	Dspxowr wpu: 0: Disable 1: Enable
		Dspxowr_wpd	[0]	0	Dspxowr wpd: 0: Disable 1: Enable
0x8C00_001C	Dspyerd_reg	Dspyerd_sel	[5:4]	0	Dspyerd select: 0:DSP1 Y-even RD 1:DSP2 Y-even RD 2: 3:
		Dspyerd_drv	[3:2]	0	Dspyerd drive
		Dspyerd_wpu	[1]	0	Dspyerdwpu: 0: Disable 1: Enable
		Dspyerd_wpd	[0]	0	Dspyerdwpd: 0: Disable 1: Enable
0x8C00_0020	Dspyewr_reg	Dspyewr_sel	[5:4]	0	Dspyewr select: 0:DSP1 Y-even WR 1:DSP2 Y-even WR 2: 3:
		Dspyewr_drv	[3:2]	0	Dspyewr drive
		Dspyewr_wpu	[1]	0	Dspyewr wpu: 0: Disable 1: Enable
		Dspyewr_wpd	[0]	0	Dspyewr wpd: 0: Disable 1: Enable
0x8C00_0024	Dspyord_reg	Dspyord_sel	[5:4]	0	Dspyord select: 0:DSP1 Y-odd RD 1:DSP2 Y-odd RD 2: 3:
		Dspyord_drv	[3:2]	0	Dspyord drive
		Dspyord_wpu	[1]	0	Dspyordwpu: 0: Disable 1: Enable
		Dspyord_wpd	[0]	0	Dspyordwpd: 0: Disable 1: Enable
0x8C00_0028	Dspyowr_reg	Dspyowr_sel	[5:4]	0	Dspyowr select: 0:DSP1 Y-odd WR 1:DSP2 Y-odd WR 2: 3:
		Dspyowr_drv	[3:2]	0	Dspyowr drive
		Dspyowr_wpu	[1]	0	Dspyowr wpu: 0: Disable 1: Enable
		Dspyowr_wpd	[0]	0	Dspyowr wpd: 0: Disable 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_002C	Dspzerd_reg	Dspzerd_sel	[5:4]	0	Dspzerd select: 0:DSP1 Z-even RD 1:DSP2 Z-even RD 2: 3:
		Dspzerd_drv	[3:2]	0	Dspzerd drive
		Dspzerd_wpu	[1]	0	Dspzerd wpu: 0: Disable 1: Enable
		Dspzerd_wpd	[0]	0	Dspzerd wpd: 0: Disable 1: Enable
0x8C00_0030	Dspzewr_reg	Dspzewr_sel	[5:4]	0	Dspzewr select: 0:DSP1 Z-even WR 1:DSP2 Z-even WR 2: 3:
		Dspzewr_drv	[3:2]	0	Dspzewr drive
		Dspzewr_wpu	[1]	0	Dspzewr wpu: 0: Disable 1: Enable
		Dspzewr_wpd	[0]	0	Dspzewr wpd: 0: Disable 1: Enable
0x8C00_0034	Dspzord_reg	Dspzord_sel	[5:4]	0	Dspzord select: 0:DSP1 Z-odd RD 1:DSP2 Z-odd RD 2: 3:
		Dspzord_drv	[3:2]	0	Dspzord drive
		Dspzord_wpu	[1]	0	Dspzord wpu: 0: Disable 1: Enable
		Dspzord_wpd	[0]	0	Dspzord wpd: 0: Disable 1: Enable
0x8C00_0038	Dspzowr_reg	Dspzowr_sel	[5:4]	0	Dspzowr select: 0:DSP1 Z-odd WR 1:DSP2 Z-odd WR 2: 3:
		Dspzowr_drv	[3:2]	0	Dspzowr drive
		Dspzowr_wpu	[1]	0	Dspzowr wpu: 0: Disable 1: Enable
		Dspzowr_wpd	[0]	0	Dspzowr wpd: 0: Disable 1: Enable
0x8C00_003C	Dspprd_reg	Dspprd_sel	[5:4]	0	Dspprd select: 0: DSP1_PRD 1: 2: 3:
		Dspprd_drv	[3:2]	0	Dspprd drive
		Dspprd_wpu	[1]	0	Dspprd wpu: 0: Disable 1: Enable
		Dspprd_wpd	[0]	0	Dspprd wpd: 0: Disable 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0040	Dsppwr_reg	Dsppwr_sel	[5:4]	0	Dsppwr select 0:DSP1_PWR 1: 2: 3:
		Dsppwr_drv	[3:2]	0	Dsppwr drive
		Dsppwr_wpu	[1]	0	Dsppwr wpu: 0: Disable 1: Enable
		Dsppwr_wpd	[0]	0	Dsppwr wpd: 0: Disble 1: Enable
0x8C00_0044	Dsp2pd15_reg	Dsp2pd15_sel	[5:4]	0	Dsp2pd15 select 0: DSP2PD15 1: 2: 3:
		Dsp2pd15_drv	[3:2]	0	Dsp2pd15 drive
		Dsp2pd15_wpu	[1]	0	Dsp2pd15 wpu: 0: Disable 1: Enable
		Dsp2pd15_wpd	[0]	0	Dsp2pd15 wpd: 0: Disble 1: Enable
0x8C00_0048	Dsp2prd_reg	Dsp2prd_sel	[5:4]	0	Dsp2prd select 0: DSP2PRD 1: 2: 3:
		Dsp2prd_drv	[3:2]	0	Dsp2prd drive
		Dsp2prd_wpu	[1]	0	Dsp2prd wpu: 0: Disable 1: Enable
		Dsp2prd_wpd	[0]	0	Dsp2prd wpd: 0: Disble 1: Enable
0x8C00_004C	Dsp2pwr_reg	Dsp2pwr_sel	[5:4]	0	Dsp2pwr select 0:DSP2PWR 1: 2: 3:
		Dsp2pwr_drv	[3:2]	0	Dsp2pwr drive
		Dsp2pwr_wpu	[1]	0	Dsp2pwr wpu: 0: Disable 1: Enable
		Dsp2pwr_wpd	[0]	0	Dsp2pwr wpd: 0: Disble 1: Enable
0x8C00_01C8	Ema0_reg	Ema0_sel	[5:4]	Write-only 0	Ema0 select 0: EMA0 1: 2: 3:
		Ema0_drv	[3:2]	Write-only 0	Ema0 Drive
		Ema0_wpu	[1]	Write-only 0	Ema0 wpu: 0: Disable 1: Enable
		Ema0_wpd	[0]	Write-only 0	Ema0 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_01CC	Ema1_reg	Ema1_sel	[5:4]	Write-only 0	Ema1 select 0: Ema1 1: 2: 3:
		Ema1_drv	[3:2]	Write-only 0	Ema1 drive
		Ema1_wpu	[1]	Write-only 0	Ema1 wpu: 0: Disable 1: Enable
		Ema1_wpd	[0]	Write-only 0	Ema1 wpd: 0: Disble 1: Enable
0x8C00_01D0	Ema2_reg	Ema2_sel	[5:4]	Write-only 0	Ema2 select 0: Ema2 1: 2: 3:
		Ema2_drv	[3:2]	Write-only 0	Ema2 Drive
		Ema2_wpu	[1]	Write-only 0	Ema2 wpu: 0: Disable 1: Enable
		Ema2_wpd	[0]	Write-only 0	Ema2 wpd: 0: Disble 1: Enable
0x8C00_01D4	Ema3_reg	Ema3_sel	[5:4]	Write-only 0	Ema3 select 0: Ema3 1: 2: 3:
		Ema3_drv	[3:2]	Write-only 0	Ema3 drive
		Ema3_wpu	[1]	Write-only 0	Ema3 wpu: 0: Disable 1: Enable
		Ema3_wpd	[0]	Write-only 0	Ema3 wpd: 0: Disble 1: Enable
0x8C00_01D8	Ema4_reg	Ema4_sel	[5:4]	Write-only 0	Ema4 select 0: Ema4 1: 2: 3:
		Ema4_drv	[3:2]	Write-only 0	Ema4 drive
		Ema4_wpu	[1]	Write-only 0	Ema4 wpu: 0: Disable 1: Enable
		Ema4_wpd	[0]	Write-only 0	Ema4 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_01DC	Ema5_reg	Ema5_sel	[5:4]	Write-only 0	Ema5 select 0: Ema5 1: 2: 3:
		Ema5_drv	[3:2]	Write-only 0	Ema5 drive
		Ema5_wpu	[1]	Write-only 0	Ema5 wpu: 0: Disable 1: Enable
		Ema5_wpd	[0]	Write-only 0	Ema5 wpd: 0: Disble 1: Enable
0x8C00_01E0	Ema6_reg	Ema6_sel	[5:4]	Write-only 0	Ema6 select 0: Ema6 1: 2: 3:
		Ema6_drv	[3:2]	Write-only 0	Ema6 drive
		Ema6_wpu	[1]	Write-only 0	Ema6 wpu: 0: Disable 1: Enable
		Ema6_wpd	[0]	Write-only 0	Ema6 wpd: 0: Disble 1: Enable
0x8C00_01E4	Ema7_reg	Ema7_sel	[5:4]	Write-only 0	Ema7 select 0: Ema7 1: 2: 3:
		Ema7_drv	[3:2]	Write-only 0	Ema7 drive
		Ema7_wpu	[1]	Write-only 0	Ema7 wpu: 0: Disable 1: Enable
		Ema7_wpd	[0]	Write-only 0	Ema7 wpd: 0: Disble 1: Enable
0x8C00_01E8	Ema8_reg	Ema8_sel	[5:4]	Write-only 0	Ema8 select 0: Ema8 1: 2: 3:
		Ema8_drv	[3:2]	Write-only 0	Ema8 drive
		Ema8_wpu	[1]	Write-only 0	Ema8 wpu: 0: Disable 1: Enable
		Ema8_wpd	[0]	Write-only 0	Ema8 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_01EC	Ema9_reg	Ema9_sel	[5:4]	Write-only 0	Ema9 select 0: Ema9 1: 2: 3:
		Ema9_drv	[3:2]	Write-only 0	Ema9 drive
		Ema9_wpu	[1]	Write-only 0	Ema9 wpu: 0: Disable 1: Enable
		Ema9_wpd	[0]	Write-only 0	Ema9 wpd: 0: Disble 1: Enable
0x8C00_01F0	Ema10_reg	Ema10_sel	[5:4]	Write-only 0	Ema10 select 0: Ema10 1: 2: 3:
		Ema10_drv	[3:2]	Write-only 0	Ema10 drive
		Ema10_wpu	[1]	Write-only 0	Ema10 wpu: 0: Disable 1: Enable
		Ema10_wpd	[0]	Write-only 0	Ema10 wpd: 0: Disble 1: Enable
0x8C00_01F4	Ema11_reg	Ema11_sel	[5:4]	Write-only 0	Ema11 select 0: Ema11 1: 2: 3:
		Ema11_drv	[3:2]	Write-only 0	Ema11 drive
		Ema11_wpu	[1]	Write-only 0	Ema11 wpu: 0: Disable 1: Enable
		Ema11_wpd	[0]	Write-only 0	Ema11 wpd: 0: Disble 1: Enable
0x8C00_01F8	Ema12_reg	Ema12_sel	[5:4]	Write-only 0	Ema12 select 0: Ema12 1: 2: 3:
		Ema12_drv	[3:2]	Write-only 0	Ema12 drive
		Ema12_wpu	[1]	Write-only 0	Ema12 wpu: 0: Disable 1: Enable
		Ema12_wpd	[0]	Write-only 0	Ema12 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_01FC	Ema13_reg	Ema13_sel	[5:4]	Write-only 0	Ema13 Select 0: Ema13 1: 2: 3:
		Ema13_drv	[3:2]	Write-only 0	Ema13 drive
		Ema13_wpu	[1]	Write-only 0	Ema13 wpu: 0: Disable 1: Enable
		Ema13_wpd	[0]	Write-only 0	Ema13 wpd: 0: Disble 1: Enable
0x8C00_0200	Ema14_reg	Ema14_sel	[5:4]	Write-only 0	Ema14 select 0: Ema14 1: 2: 3:
		Ema14_drv	[3:2]	Write-only 0	Ema14 drive
		Ema14_wpu	[1]	Write-only 0	Ema14 wpu: 0: Disable 1: Enable
		Ema14_wpd	[0]	Write-only 0	Ema14 wpd: 0: Disble 1: Enable
0x8C00_0204	Ema15_reg	Ema15_sel	[5:4]	Write-only 0	Ema15 select 0: Ema15 1: 2: 3:
		Ema15_drv	[3:2]	Write-only 0	Ema15 drive
		Ema15_wpu	[1]	Write-only 0	Ema15 wpu: 0: Disable 1: Enable
		Ema15_wpd	[0]	Write-only 0	Ema15 wpd: 0: Disble 1: Enable
0x8C00_0208	Ema16_reg	Ema16_sel	[5:4]	Write-only 0	Ema16 select 0: Ema16 1: 2: 3:
		Ema16_drv	[3:2]	Write-only 0	Ema16 drive
		Ema16_wpu	[1]	Write-only 0	Ema16 wpu: 0: Disable 1: Enable
		Ema16_wpd	[0]	Write-only 0	Ema16 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_020C	Ema17_reg	Ema17_sel	[5:4]	Write-only 0	Ema17 select 0: Ema17 1: 2: 3:
		Ema17_drv	[3:2]	Write-only 0	Ema17 drive
		Ema17_wpu	[1]	Write-only 0	Ema17 wpu: 0: Disable 1: Enable
		Ema17_wpd	[0]	Write-only 0	Ema17 wpd: 0: Disble 1: Enable
0x8C00_0210	Ema18_reg	Ema18_sel	[5:4]	Write-only 0	Ema18 select 0: Ema18 1: 2: 3:
		Ema18_drv	[3:2]	Write-only 0	Ema18 drive
		Ema18_wpu	[1]	Write-only 0	Ema18 wpu: 0: Disable 1: Enable
		Ema18_wpd	[0]	Write-only 0	Ema18 wpd: 0: Disble 1: Enable
0x8C00_0214	Ema19_reg	Ema19_sel	[5:4]	Write-only 0	Ema19 select 0: Ema19 1: 2: 3:
		Ema19_drv	[3:2]	Write-only 0	Ema19 drive
		Ema19_wpu	[1]	Write-only 0	Ema19 wpu: 0: Disable 1: Enable
		Ema19_wpd	[0]	Write-only 0	Ema19 wpd: 0: Disble 1: Enable
0x8C00_0218	Ema20_reg	Ema20_sel	[5:4]	Write-only 0	Ema20 select 0: Ema20 1: 2: 3:
		Ema20_drv	[3:2]	Write-only 0	Ema20 drive
		Ema20_wpu	[1]	Write-only 0	Ema20 wpu: 0: Disable 1: Enable
		Ema20_wpd	[0]	Write-only 0	Ema20 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_021C	Ema21_reg	Ema21_sel	[5:4]	Write-only 0	Ema21 select 0:Ema21 1: 2: 3:
		Ema21_drv	[3:2]	Write-only 0	Ema21 drive
		Ema21_wpu	[1]	Write-only 0	Ema21 wpu: 0: Disable 1: Enable
		Ema21_wpd	[0]	Write-only 0	Ema21 wpd: 0: Disble 1: Enable
0x8C00_0220	Ema22_reg	Ema22_sel	[5:4]	Write-only 0	Ema22 select 0: Ema22 1: 2: 3: GPIO8
		Ema22_drv	[3:2]	Write-only 0	Ema22 drive
		Ema22_wpu	[1]	Write-only 0	Ema22 wpu: 0: Disable 1: Enable
		Ema22_wpd	[0]	Write-only 0	Ema22 wpd: 0: Disble 1: Enable
0x8C00_0224	Ema23_reg	Ema23_sel	[5:4]	Write-only 0	Ema23 select 0: Ema23 1: 2: 3: GPIO9
		Ema23_drv	[3:2]	Write-only 0	Ema23 drive
		Ema23_wpu	[1]	Write-only 0	Ema23 wpu: 0: Disable 1: Enable
		Ema23_wpd	[0]	Write-only 0	Ema23 wpd: 0: Disble 1: Enable
0x8C00_0228	Ema24_reg	Ema24_sel	[5:4]	Write-only 0	Ema24 select 0: Ema24 1: 2: 3: GPIO10
		Ema24_drv	[3:2]	Write-only 0	Ema24 drive
		Ema24_wpu	[1]	Write-only 0	Ema24 wpu: 0: Disable 1: Enable
		Ema24_wpd	[0]	Write-only 0	Ema24 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_022C	Emd0_reg	Emd0_sel	[5:4]	Write-only 0	Emd0 select 0: Emd0 1: 2: 3:
		Emd0_drv	[3:2]	Write-only 0	Emd0 drive
		Emd0_wpu	[1]	Write-only 0	Emd0 wpu: 0: Disable 1: Enable
		Emd0_wpd	[0]	Write-only 0	Emd0 wpd: 0: Disble 1: Enable
0x8C00_0230	Emd1_reg	Emd1_sel	[5:4]	Write-only 0	Emd1 select 0: Emd1 1: 2: 3:
		Emd1_drv	[3:2]	Write-only 0	Emd1 drive
		Emd1_wpu	[1]	Write-only 0	Emd1 wpu: 0: Disable 1: Enable
		Emd1_wpd	[0]	Write-only 0	Emd1 wpd: 0: Disble 1: Enable
0x8C00_0234	Emd2_reg	Emd2_sel	[5:4]	Write-only 0	Emd2 select 0:Emd2 1: 2: 3:
		Emd2_drv	[3:2]	Write-only 0	Emd2 drive
		Emd2_wpu	[1]	Write-only 0	Emd2 wpu: 0: Disable 1: Enable
		Emd2_wpd	[0]	Write-only 0	Emd2 wpd: 0: Disble 1: Enable
0x8C00_0238	Emd3_reg	Emd3_sel	[5:4]	Write-only 0	Emd3 select 0: Emd3 1: 2: 3:
		Emd3_drv	[3:2]	Write-only 0	Emd3 drive
		Emd3_wpu	[1]	Write-only 0	Emd3 wpu: 0: Disable 1: Enable
		Emd3_wpd	[0]	Write-only 0	Emd3 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_023C	Emd4_reg	Emd4_sel	[5:4]	Write-only 0	Emd4 select 0: Emd4 1: 2: 3:
		Emd4_drv	[3:2]	Write-only 0	Emd4 drive
		Emd4_wpu	[1]	Write-only 0	Emd4 wpu: 0: Disable 1: Enable
		Emd4_wpd	[0]	Write-only 0	Emd4 wpd: 0: Disble 1: Enable
0x8C00_0240	Emd5_reg	Emd5_sel	[5:4]	Write-only 0	Emd5 select 0: Emd5 1: 2: 3:
		Emd5_drv	[3:2]	Write-only 0	Emd5 drive
		Emd5_wpu	[1]	Write-only 0	Emd5 wpu: 0: Disable 1: Enable
		Emd5_wpd	[0]	Write-only 0	Emd5 wpd: 0: Disble 1: Enable
0x8C00_0244	Emd6_reg	Emd6_sel	[5:4]	Write-only 0	Emd6 select 0: Emd6 1: 2: 3:
		Emd6_drv	[3:2]	Write-only 0	Emd6 drive
		Emd6_wpu	[1]	Write-only 0	Emd6 wpu: 0: Disable 1: Enable
		Emd6_wpd	[0]	Write-only 0	Emd6 wpd: 0: Disble 1: Enable
0x8C00_0248	Emd7_reg	Emd7_sel	[5:4]	Write-only 0	Emd7 select 0: Emd7 1: 2: 3:
		Emd7_drv	[3:2]	Write-only 0	Emd7 drive
		Emd7_wpu	[1]	Write-only 0	Emd7 wpu: 0: Disable 1: Enable
		Emd7_wpd	[0]	Write-only 0	Emd7 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_024C	Emd8_reg	Emd8_sel	[5:4]	Write-only 0	Emd8 select 0: Emd8 1: 2: 3:
		Emd8_drv	[3:2]	Write-only 0	Emd8 drive
		Emd8_wpu	[1]	Write-only 0	Emd8 wpu: 0: Disable 1: Enable
		Emd8_wpd	[0]	Write-only 0	Emd8 wpd: 0: Disble 1: Enable
0x8C00_0250	Emd9_reg	Emd9_sel	[5:4]	Write-only 0	Emd9 select 0: Emd9 1: 2: 3:
		Emd9_drv	[3:2]	Write-only 0	Emd9 drive
		Emd9_wpu	[1]	Write-only 0	Emd9 wpu: 0: Disable 1: Enable
		Emd9_wpd	[0]	Write-only 0	Emd9 wpd: 0: Disble 1: Enable
0x8C00_0254	Emd10_reg	Emd10_sel	[5:4]	Write-only 0	Emd10 select 0: Emd10 1: 2: 3:
		Emd10_drv	[3:2]	Write-only 0	Emd10 drive
		Emd10_wpu	[1]	Write-only 0	Emd10 wpu: 0: Disable 1: Enable
		Emd10_wpd	[0]	Write-only 0	Emd10 wpd: 0: Disble 1: Enable
0x8C00_0258	Emd11_reg	Emd11_sel	[5:4]	Write-only 0	Emd11 select 0: Emd11 1: 2: 3:
		Emd11_drv	[3:2]	Write-only 0	Emd11 drive
		Emd11_wpu	[1]	Write-only 0	Emd11 wpu: 0: Disable 1: Enable
		Emd11_wpd	[0]	Write-only 0	Emd11 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_025C	Emd12_reg	Emd12_sel	[5:4]	Write-only 0	Emd12 select 0: Emd12 1: 2: 3:
		Emd12_drv	[3:2]	Write-only 0	Emd12 drive
		Emd12_wpu	[1]	Write-only 0	Emd12 wpu: 0: Disable 1: Enable
		Emd12_wpd	[0]	Write-only 0	Emd12 wpd: 0: Disble 1: Enable
0x8C00_0260	Emd13_reg	Emd13_sel	[5:4]	Write-only 0	Emd13 select 0: Emd13 1: 2: 3:
		Emd13_drv	[3:2]	Write-only 0	Emd13 drive
		Emd13_wpu	[1]	Write-only 0	Emd13 wpu: 0: Disable 1: Enable
		Emd13_wpd	[0]	Write-only 0	Emd13 wpd: 0: Disble 1: Enable
0x8C00_0264	Emd14_reg	Emd14_sel	[5:4]	Write-only 0	Emd14 select 0: Emd14 1: 2: 3:
		Emd14_drv	[3:2]	Write-only 0	Emd14 drive
		Emd14_wpu	[1]	Write-only 0	Emd14 wpu: 0: Disable 1: Enable
		Emd14_wpd	[0]	Write-only 0	Emd14 wpd: 0: Disble 1: Enable
0x8C00_0268	Emd15_reg	Emd15_sel	[5:4]	Write-only 0	Emd15 select 0: Emd15 1: 2: 3:
		Emd15_drv	[3:2]	Write-only 0	Emd15 drive
		Emd15_wpu	[1]	Write-only 0	Emd15 wpu: 0: Disable 1: Enable
		Emd15_wpd	[0]	Write-only 0	Emd15 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_026C	Embh_reg	Embh_sel	[5:4]	Write-only 0	Embh select 0: Embh 1: 2: 3:
		Embh_drv	[3:2]	Write-only 0	Embh drive
		Embh_wpu	[1]	Write-only 0	Embh wpu: 0: Disable 1: Enable
		Embh_wpd	[0]	Write-only 0	Embh wpd: 0: Disble 1: Enable
0x8C00_0270	Embl_reg	Embl_sel	[5:4]	Write-only 0	Embl select 0: Embl 1: 2: 3:
		Embl_drv	[3:2]	Write-only 0	Embl drive
		Embl_wpu	[1]	Write-only 0	Embl wpu: 0: Disable 1: Enable
		Embl_wpd	[0]	Write-only 0	Embl wpd: 0: Disble 1: Enable
0x8C00_0274	Emcsn0_reg	Emcsn0_sel	[5:4]	Write-only 0	Emcsn0 select 0: Emcs0 1: 2: 3:
		Emcsn0_drv	[3:2]	Write-only 0	Emcsn0 drive
		Emcsn0_wpu	[1]	Write-only 0	Emcsn0 wpu: 0: Disable 1: Enable
		Emcsn0_wpd	[0]	Write-only 0	Emcsn0 wpd: 0: Disble 1: Enable
0x8C00_0278	Emcsn1_reg	Emcsn1_sel	[5:4]	Write-only 0	Emcsn1 select 0: Emcsn1 1: 2: 3:
		Emcsn1_drv	[3:2]	Write-only 0	Emcsn1 drive
		Emcsn1_wpu	[1]	Write-only 0	Emcsn1 wpu: 0: Disable 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
		Emcsn1_wpd	[0]	Write-only 0	Emcsn1 wpd: 0: Disble 1: Enable
0x8C00_027C	Emcsn2_reg	Emcsn2_sel	[5:4]	Write-only 0	Emcsn2 select 0: Emcsn2 1: 2: 3: SCL [1]
		Emcsn2_drv	[3:2]	Write-only 0	Emcsn2 drive
		Emcsn2_wpu	[1]	Write-only 0	Emcsn2 wpu: 0: Disable 1: Enable
		Emcsn2_wpd	[0]	Write-only 0	Emcsn2 wpd: 0: Disble 1: Enable
0x8C00_0280	Emcsn3_reg	Emcsn3_sel	[5:4]	Write-only 0	Emcsn3 select 0: Emcsn3 1: 2: 3: GPIO6
		Emcsn3_drv	[3:2]	Write-only 0	Emcsn3 drive
		Emcsn3_wpu	[1]	Write-only 0	Emcsn3 wpu: 0: Disable 1: Enable
		Emcsn3_wpd	[0]	Write-only 0	Emcsn3 wpd: 0: Disble 1: Enable
0x8C00_0284	Emoen_reg	Emoen_sel	[5:4]	Write-only 0	Emoen select 0: Emoen 1: 2: 3:
		Emoen_drv	[3:2]	Write-only 0	Emoen drive
		Emoen_wpu	[1]	Write-only 0	Emoen wpu: 0: Disable 1: Enable
		Emoen_wpd	[0]	Write-only 0	Emoen wpd: 0: Disble 1: Enable
0x8C00_0288	Emwen_reg	Emwen_sel	[5:4]	Write-only 0	Emwen select 0: Emwen 1: 2: 3:
		Emwen_drv	[3:2]	Write-only 0	Emwen drive

[1] This is GPIO 5 on SC8800A1.

Address	Register	Signal	Bit Pos	Default	Description
		Emwen_wpu	[1]	Write-only 0	Emwen wpu: 0: Disable 1: Enable
		Emwen_wpd	[0]	Write-only 0	Emwen wpd: 0: Disble 1: Enable
0x8C00_028C	Emadv_reg	Emadv_sel	[5:4]	0	Emadv select 0: Emadv 1: 2: 3: GPIO7
		Emadv_drv	[3:2]	0	Emadv drive
		Emadv_wpu	[1]	0	Emadv wpu: 0: Disable 1: Enable
		Emadv_wpd	[0]	0	Emadv wpd: 0: Disble 1: Enable
0x8C00_0290	Dmbh_reg	Dmbh_sel	[5:4]	0	Dmbh select 0: Dmbh 1: 2: 3: GPIO11
		Dmbh_drv	[3:2]	0	Dmbh Drive
		Dmbh_wpu	[1]	0	Dmbh wpu: 0: Disable 1: Enable
		Dmbh_wpd	[0]	0	Dmbh wpd: 0: Disble 1: Enable
0x8C00_0294	Dmbl_reg	Dmbl_sel	[5:4]	0	Dmbl select 0: Dmbl 1: 2: 3: GPIO12
		Dmbl_drv	[3:2]	0	Dmbl drive
		Dmbl_wpu	[1]	0	Dmbl wpu: 0: Disable 1: Enable
		Dmbl_wpd	[0]	0	Dmbl wpd: 0: Disble 1: Enable
0x8C00_0298	Dmcke_reg	Dmcke_sel	[5:4]	0	Dmcke select 0: Dmcke 1: 2: 3: GPIO13
		Dmcke_drv	[3:2]	0	Dmcke drive
		Dmcke_wpu	[1]	0	Dmcke wpu: 0: Disable 1: Enable
		Dmcke_wpd	[0]	0	Dmcke wpd: 0: Disble 1: Enable
0x8C00_029C	Dmcsn0_reg	Dmcsn0_sel	[5:4]	0	Dmcsn0 select 0: Dmcsn0 1: 2: 3: GPIO18
		Dmcsn0_drv	[3:2]	0	Dmcsn0 drive

Address	Register	Signal	Bit Pos	Default	Description
		Dmcsn0_wpu	[1]	0	Dmcsn0 wpu: 0: Disable 1: Enable
		Dmcsn0_wpd	[0]	0	Dmcsn0 wpd: 0: Disble 1: Enable
0x8C00_02A0	Dmcsn1_reg	Dmcsn1_sel	[5:4]	0	Dmcsn1 select 0: Dmcsn1 1: 2: 3: GPIO19
		Dmcsn1_drv	[3:2]	0	Dmcsn1 drive
		Dmcsn1_wpu	[1]	0	Dmcsn1 wpu: 0: Disable 1: Enable
		Dmcsn1_wpd	[0]	0	Dmcsn1 wpd: 0: Disble 1: Enable
0x8C00_02A4	Dmcsn2_reg	Dmcsn2_sel	[5:4]	0	Dmcsn2 select 0: Dmcsn2 1: 2: 3: GPIO20
		Dmcsn2_drv	[3:2]	0	Dmcsn2 drive
		Dmcsn2_wpu	[1]	0	Dmcsn2 wpu: 0: Disable 1: Enable
		Dmcsn2_wpd	[0]	0	Dmcsn2 wpd: 0: Disble 1: Enable
0x8C00_02A8	Dmcsn3_reg	Dmcsn3_sel	[5:4]	0	Dmcsn3 select 0: Dmcsn3 1: 2: 3: GPIO21
		Dmcsn3_drv	[3:2]	0	Dmcsn3 drive
		Dmcsn3_wpu	[1]	0	Dmcsn3 wpu: 0: Disable 1: Enable
		Dmcsn3_wpd	[0]	0	Dmcsn3 wpd: 0: Disble 1: Enable
0x8C00_02AC	Dmwen_reg	Dmwen_sel	[5:4]	0	Dmwen select 0: Dmwen 1: 2: 3: GPIO23
		Dmwen_drv	[3:2]	0	Dmwen drive
		Dmwen_wpu	[1]	0	Dmwen wpu: 0: Disable 1: Enable
		Dmwen_wpd	[0]	0	Dmwen wpd: 0: Disble 1: Enable
0x8C00_02B0	Dma0_reg	Dma0_sel	[5:4]	0	Dma0 select 0: Dma0 1: 2: 3: GPIO24[1]
		Dma0_drv	[3:2]	0	Dma0 drive

[1] GPIO [24] is not available on SC8800A2. It is used for minute interrupt.

Address	Register	Signal	Bit Pos	Default	Description
		Dma0_wpu	[1]	0	Dma0 wpu: 0: Disable 1: Enable
		Dma0_wpd	[0]	0	Dma0 wpd: 0: Disble 1: Enable
0x8C00_02B4	Dma1_reg	Dma1_sel	[5:4]	0	Dma1 select 0: Dma1 1: 2: 3: GPIO25[1]
		Dma1_drv	[3:2]	0	Dma1 drive
		Dma1_wpu	[1]	0	Dma1 wpu: 0: Disable 1: Enable
		Dma1_wpd	[0]	0	Dma1 wpd: 0: Disble 1: Enable
0x8C00_02B8	Dma2_reg	Dma2_sel	[5:4]	0	Dma2 select 0: Dma2 1: 2: 3: GPIO26
		Dma2_drv	[3:2]	0	Dma2 drive
		Dma2_wpu	[1]	0	Dma2 wpu: 0: Disable 1: Enable
		Dma2_wpd	[0]	0	Dma2 wpd: 0: Disble 1: Enable
0x8C00_02BC	Dma3_reg	Dma3_sel	[5:4]	0	Dma3 select 0: Dma3 1: 2: 3: GPIO27
		Dma3_drv	[3:2]	0	Dma3 drive
		Dma3_wpu	[1]	0	Dma3 wpu: 0: Disable 1: Enable
		Dma3_wpd	[0]	0	Dma3 wpd: 0: Disble 1: Enable
0x8C00_02C0	Dma4_reg	Dma4_sel	[5:4]	0	Dma4 select 0: Dma4 1: 2: 3: GPIO28
		Dma4_drv	[3:2]	0	Dma4 drive
		Dma4_wpu	[1]	0	Dma4 wpu: 0: Disable 1: Enable
		Dma4_wpd	[0]	0	Dma4 wpd: 0: Disble 1: Enable
0x8C00_02C4	Dma5_reg	Dma5_sel	[5:4]	0	Dma5 select 0:Dma5 1: 2: 3: GPIO29
		Dma5_drv	[3:2]	0	Dma5 drive

[1] GPIO [25] is not available on SC8800A2. It is used for charger interrupt.

Address	Register	Signal	Bit Pos	Default	Description
		Dma5_wpu	[1]	0	Dma5 wpu: 0: Disable 1: Enable
		Dma5_wpd	[0]	0	Dma5 wpd: 0: Disble 1: Enable
0x8C00_02C8	Dma6_reg	Dma6_sel	[5:4]	0	Dma6 select 0: Dma6 1: 2: 3: GPIO30
0x8C00_02CC	Dma7_reg	Dma6_drv	[3:2]	0	Dma6 drive
		Dma6_wpu	[1]	0	Dma6 wpu: 0: Disable 1: Enable
		Dma6_wpd	[0]	0	Dma6 wpd: 0: Disble 1: Enable
		Dma7_sel	[5:4]	0	Dma7 select 0: Dma7 1: 2: 3: GPIO31
		Dma7_drv	[3:2]	0	Dma7 drive
		Dma7_wpu	[1]	0	Dma7 wpu: 0: Disable 1: Enable
		Dma7_wpd	[0]	0	Dma7 wpd: 0: Disble 1: Enable
0x8C00_02D0	Dma8_reg	Dma8_sel	[5:4]	0	Dma8 select 0: Dma8 1: 2: 3: GPIO32
		Dma8_drv	[3:2]	0	Dma8 drive
		Dma8_wpu	[1]	0	Dma8 wpu: 0: Disable 1: Enable
		Dma8_wpd	[0]	0	Dma8 wpd: 0: Disble 1: Enable
0x8C00_02D4	Dma9_reg	Dma9_sel	[5:4]	0	Dma9 select 0: Dma9 1: 2: 3: GPIO33
		Dma9_drv	[3:2]	0	Dma9 drive
		Dma9_wpu	[1]	0	Dma9 wpu: 0: Disable 1: Enable
		Dma9_wpd	[0]	0	Dma9 wpd: 0: Disble 1: Enable
0x8C00_02D8	Dma10_reg	Dma10_sel	[5:4]	0	Dma10 select 0: Dma10 1: 2: 3: GPIO34
		Dma10_drv	[3:2]	0	Dma10 drive
		Dma10_wpu	[1]	0	Dma10 wpu: 0: Disable 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_02DC	Dma11_reg	Dma10_wpd	[0]	0	Dma10 wpd: 0: Disble 1: Enable
		Dma11_sel	[5:4]	0	Dma11 select 0: Dma11 1: 2: 3: GPIO35
		Dma11_drv	[3:2]	0	Dma11 drive
		Dma11_wpu	[1]	0	Dma11 wpu: 0: Disable 1: Enable
		Dma11_wpd	[0]	0	Dma11 wpd: 0: Disble 1: Enable
0x8C00_02E0	Dma12_reg	Dma12_sel	[5:4]	0	Dma12 select 0: Dma12 1: 2: 3: GPIO36
		Dma12_drv	[3:2]	0	Dma12 drive
		Dma12_wpu	[1]	0	Dma12 wpu: 0: Disable 1: Enable
		Dma12_wpd	[0]	0	Dma12 wpd: 0: Disble 1: Enable
0x8C00_02E4	Dmd0_reg	Dmd0_sel	[5:4]	0	Dmd0 select 0: Dmd0 1: 2: 3: GPIO37
		Dmd0_drv	[3:2]	0	Dmd0 drive
		Dmd0_wpu	[1]	0	Dmd0 wpu: 0: Disable 1: Enable
		Dmd0_wpd	[0]	0	Dmd0 wpd: 0: Disble 1: Enable
0x8C00_02E8	Dmd1_reg	Dmd1_sel	[5:4]	0	Dmd1 select 0: Dmd1 1: 2: 3: GPIO38
		Dmd1_drv	[3:2]	0	Dmd1 drive
		Dmd1_wpu	[1]	0	Dmd1 wpu: 0: Disable 1: Enable
		Dmd1_wpd	[0]	0	Dmd1 wpd: 0: Disble 1: Enable
0x8C00_02EC	Dmd2_reg	Dmd2_sel	[5:4]	0	Dmd2 select 0: Dmd2 1: 2: 3: GPIO39
		Dmd2_drv	[3:2]	0	Dmd2 drive
		Dmd2_wpu	[1]	0	Dmd2 wpu: 0: Disable 1: Enable
		Dmd2_wpd	[0]	0	Dmd2 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_02F0	Dmd3_reg	Dmd3_sel	[5:4]	0	Dmd3 select 0: Dmd3 1: 2: 3: GPIO40
		Dmd3_drv	[3:2]	0	Dmd3 drive
		Dmd3_wpu	[1]	0	Dmd3 wpu: 0: Disable 1: Enable
		Dmd3_wpd	[0]	0	Dmd3 wpd: 0: Disble 1: Enable
0x8C00_02F4	Dmd4_reg	Dmd4_sel	[5:4]	0	Dmd4 select 0: Dmd4 1: 2: 3: GPIO41
		Dmd4_drv	[3:2]	0	Dmd4 drive
		Dmd4_wpu	[1]	0	Dmd4 wpu: 0: Disable 1: Enable
		Dmd4_wpd	[0]	0	Dmd4 wpd: 0: Disble 1: Enable
0x8C00_02F8	Dmd5_reg	Dmd5_sel	[5:4]	0	Dmd5 select 0: Dmd5 1: 2: 3: GPIO42
		Dmd5_drv	[3:2]	0	Dmd5 drive
		Dmd5_wpu	[1]	0	Dmd5 wpu: 0: Disable 1: Enable
		Dmd5_wpd	[0]	0	Dmd5 wpd: 0: Disble 1: Enable
0x8C00_02FC	Dmd6_reg	Dmd6_sel	[5:4]	0	Dmd6 select 0: Dmd6 1: 2: 3: GPIO43
		Dmd6_drv	[3:2]	0	Dmd6 drive
		Dmd6_wpu	[1]	0	Dmd6 wpu: 0: Disable 1: Enable
		Dmd6_wpd	[0]	0	Dmd6 wpd: 0: Disble 1: Enable
0x8C00_0300	Dmd7_reg	Dmd7_sel	[5:4]	0	Dmd7 select 0: Dmd7 1: 2: 3: GPIO44
		Dmd7_drv	[3:2]	0	Dmd7 drive
		Dmd7_wpu	[1]	0	Dmd7 wpu: 0: Disable 1: Enable
		Dmd7_wpd	[0]	0	Dmd7 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0304	Dmd8_reg	Dmd8_sel	[5:4]	0	Dmd8 select 0: Dmd8 1: 2: 3: GPIO45
		Dmd8_drv	[3:2]	0	Dmd8 drive
		Dmd8_wpu	[1]	0	Dmd8 wpu: 0: Disable 1: Enable
		Dmd8_wpd	[0]	0	Dmd8 wpd: 0: Disble 1: Enable
0x8C00_0308	Dmd9_reg	Dmd9_sel	[5:4]	0	Dmd9 select 0: Dmd9 1: 2: 3: GPIO46
		Dmd9_drv	[3:2]	0	Dmd9 drive
		Dmd9_wpu	[1]	0	Dmd9 wpu: 0: Disable 1: Enable
		Dmd9_wpd	[0]	0	Dmd9 wpd: 0: Disble 1: Enable
0x8C00_030C	Dmd10_reg	Dmd10_sel	[5:4]	0	Dmd10 select 0: Dmd10 1: 2: 3: GPIO47
		Dmd10_drv	[3:2]	0	Dmd10 drive
		Dmd10_wpu	[1]	0	Dmd10 wpu: 0: Disable 1: Enable
		Dmd10_wpd	[0]	0	Dmd10 wpd: 0: Disble 1: Enable
0x8C00_0310	Dmd11_reg	Dmd11_sel	[5:4]	0	Dmd11 select 0: Dmd11 1: 2: 3: GPIO48
		Dmd11_drv	[3:2]	0	Dmd11 drive
		Dmd11_wpu	[1]	0	Dmd11 wpu: 0: Disable 1: Enable
		Dmd11_wpd	[0]	0	Dmd11 wpd: 0: Disble 1: Enable
0x8C00_0314	Dmd12_reg	Dmd12_sel	[5:4]	0	Dmd12 select 0: Dmd12 1: 2: 3: GPIO49
		Dmd12_drv	[3:2]	0	Dmd12 drive
		Dmd12_wpu	[1]	0	Dmd12 wpu: 0: Disable 1: Enable
		Dmd12_wpd	[0]	0	Dmd12 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0318	Dmd13_reg	Dmd13_sel	[5:4]	0	Dmd13 select 0: Dmd13 1: 2: 3: GPIO50
		Dmd13_drv	[3:2]	0	Dmd13 drive
		Dmd13_wpu	[1]	0	Dmd13 wpu: 0: Disable 1: Enable
		Dmd13_wpd	[0]	0	Dmd13 wpd: 0: Disble 1: Enable
0x8C00_031C	Dmd14_reg	Dmd14_sel	[5:4]	0	Dmd14 select 0: Dmd14 1: 2: 3: GPIO51
		Dmd14_drv	[3:2]	0	Dmd14 drive
		Dmd14_wpu	[1]	0	Dmd14 wpu: 0: Disable 1: Enable
		Dmd14_wpd	[0]	0	Dmd14 wpd: 0: Disble 1: Enable
0x8C00_0320	Dmd15_reg	Dmd15_sel	[5:4]	0	Dmd15 select 0: Dmd15 1: 2: 3: GPIO52
		Dmd15_drv	[3:2]	0	Dmd15 drive
		Dmd15_wpu	[1]	0	Dmd15 wpu: 0: Disable 1: Enable
		Dmd15_wpd	[0]	0	Dmd15 wpd: 0: Disble 1: Enable
0x8C00_0324	Dmba0_reg	Dmba0_sel	[5:4]	0	Dmba0 select 0: Dmba0 1: 2: 3: GPIO14
		Dmba0_drv	[3:2]	0	Dmba0 drive
		Dmba0_wpu	[1]	0	Dmba0 wpu: 0: Disable 1: Enable
		Dmba0_wpd	[0]	0	Dmba0 wpd: 0: Disble 1: Enable
0x8C00_0328	Dmba1_reg	Dmba1_sel	[5:4]	0	Dmba1 select 0: Dmba1 1: 2: 3: GPIO15
		Dmba1_drv	[3:2]	0	Dmba1 drive
		Dmba1_wpu	[1]	0	Dmba1 wpu: 0: Disable 1: Enable
		Dmba1_wpd	[0]	0	Dmba1 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_032C	Dmcas_reg	Dmcas_sel	[5:4]	0	Dmcas select 0: Dmcas 1: 2: 3: GPIO16
		Dmcas_drv	[3:2]	0	Dmcas drive
		Dmcas_wpu	[1]	0	Dmcas wpu: 0: Disable 1: Enable
		Dmcas_wpd	[0]	0	Dmcas wpd: 0: Disble 1: Enable
0x8C00_0330	Dmras_reg	Dmras_sel	[5:4]	0	Dmras select 0: Dmras 1: 2: 3: GPIO17
		Dmras_drv	[3:2]	0	Dmras drive
		Dmras_wpu	[1]	0	Dmras wpu: 0: Disable 1: Enable
		Dmras_wpd	[0]	0	Dmras wpd: 0: Disble 1: Enable
0x8C00_0334	Rfsda_reg	Rfsda_sel	[5:4]	0	Rfsda select 0: Rfsda 1: 2: 3: GPIO56
		Rfsda_drv	[3:2]	0	Rfsda drive
		Rfsda_wpu	[1]	0	Rfsda wpu: 0: Disable 1: Enable
		Rfsda_wpd	[0]	0	Rfsda wpd: 0: Disble 1: Enable
0x8C00_0338	Rfsck_reg	Rfsck_sel	[5:4]	0	Rfsck select 0: Rfsck 1: 2: 3: GPIO57
		Rfsck_drv	[3:2]	0	Rfsck drive
		Rfsck_wpu	[1]	0	Rfsck wpu: 0: Disable 1: Enable
		Rfsck_wpd	[0]	0	Rfsck wpd: 0: Disble 1: Enable
0x8C00_033C	Rfsen1_reg	Rfsen1_sel	[5:4]	0	Rfsen1 select 0: Rfsen1 1: 2: 3: GPIO58
		Rfsen1_drv	[3:2]	0	Rfsen1 drive
		Rfsen1_wpu	[1]	0	Rfsen1wpu: 0: Disable 1: Enable
		Rfsen1_wpd	[0]	0	Rfsen1wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0340	Rfsen2_reg	Rfsen2_sel	[5:4]	0	Rfsen2 select 0: Rfsen2 1: 2: 3: GPIO59
		Rfsen2_drv	[3:2]	0	Rfsen2 drive
		Rfsen2_wpu	[1]	0	Rfsen2 wpu: 0: Disable 1: Enable
		Rfsen2_wpd	[0]	0	Rfsen2 wpd: 0: Disble 1: Enable
0x8C00_0344	Rfsen3_reg	Rfsen3_sel	[5:4]	0	Rfsen3 select 0: Rfsen3 1: 2: 3: GPIO60
		Rfsen3_drv	[3:2]	0	Rfsen3 Drive
		Rfsen3_wpu	[1]	0	Rfsen3 wpu: 0: Disable 1: Enable
		Rfsen3_wpd	[0]	0	Rfsen3 wpd: 0: Disble 1: Enable
0x8C00_0348	Rfctl0_reg	Rfctl0_sel	[5:4]	0	Rfctl0 select 0: Rfctl0 1: DSP1 GPIO0 2: DSP2 GPIO0 3: GPIO61
		Rfctl0_drv	[3:2]	0	Rfctl0 Drive
		Rfctl0_wpu	[1]	0	Rfctl0 Wpu: 0: Disable 1: Enable
		Rfctl0_wpd	[0]	0	Rfctl0 wpd: 0: Disble 1: Enable
0x8C00_034C	Rfctl1_reg	Rfctl1_sel	[5:4]	0	Rfctl1 select 0:Rfctl1 1: DSP1 GPIO1 2: DSP2 GPIO1 3: GPIO62
		Rfctl1_drv	[3:2]	0	Rfctl1 Drive
		Rfctl1_wpu	[1]	0	Rfctl1 wpu: 0: Disable 1: Enable
		Rfctl1_wpd	[0]	0	Rfctl1 wpd: 0: Disble 1: Enable
0x8C00_0350	Rfctl2_reg	Rfctl2_sel	[5:4]	0	Rfctl2 select 0: Rfctl2 1: DSP1 GPIO2 2: DSP2 GPIO2 3: GPIO63
		Rfctl2_drv	[3:2]	0	Rfctl2 drive
		Rfctl2_wpu	[1]	0	Rfctl2 wpu: 0: Disable 1: Enable
		Rfctl2_wpd	[0]	0	Rfctl2 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0354	Rfctl3_reg	Rfctl3_sel	[5:4]	0	Rfctl3 select 0: Rfctl3 1: DSP1 GPIO3 2: DSP2 GPIO3 3: GPIO64
		Rfctl3_drv	[3:2]	0	Rfctl3 drive
		Rfctl3_wpu	[1]	0	Rfctl3 wpu: 0: Disable 1: Enable
		Rfctl3_wpd	[0]	0	Rfctl3 wpd: 0: Disble 1: Enable
0x8C00_0358	Rfctl4_reg	Rfctl4_sel	[5:4]	0	Rfctl4 select 0: Rfctl4 1: DSP1 GPIO4 2: DSP2 GPIO4 3: GPIO65
		Rfctl4_drv	[3:2]	0	Rfctl4 drive
		Rfctl4_wpu	[1]	0	Rfctl4 wpu: 0: Disable 1: Enable
		Rfctl4_wpd	[0]	0	Rfctl4 wpd: 0: Disble 1: Enable
0x8C00_035C	Rfctl5_reg	Rfctl5_sel	[5:4]	0	Rfctl5 select 0: Rfctl5 1: DSP1 GPIO5 2: DSP2 GPIO5 3: GPIO66
		Rfctl5_drv	[3:2]	0	Rfctl5 drive
		Rfctl5_wpu	[1]	0	Rfctl5 wpu: 0: Disable 1: Enable
		Rfctl5_wpd	[0]	0	Rfctl5 wpd: 0: Disble 1: Enable
0x8C00_0360	Rfctl6_reg	Rfctl6_sel	[5:4]	0	Rfctl6 select 0: Rfctl6 1: DSP1 GPIO6 2: DSP2 GPIO6 3: GPIO67
		Rfctl6_drv	[3:2]	0	Rfctl6 drive
		Rfctl6_wpu	[1]	0	Rfctl6 wpu: 0: Disable 1: Enable
		Rfctl6_wpd	[0]	0	Rfctl6 wpd: 0: Disble 1: Enable
0x8C00_04B0	Rfctl7_reg	Rfctl7_sel	[5:4]	0	Rfctl7 select 0: Rfctl7 1: DSP1 GPIO7 2: DSP2 GPIO7 3: GPIO68
		Rfctl7_drv	[3:2]	0	Rfctl7 drive
		Rfctl7_wpu	[1]	0	Rfctl7 wpu: 0: Disable 1: Enable
		Rfctl7_wpd	[0]	0	Rfctl7 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0364	Rfctl8_reg	Rfctl8_sel	[5:4]	0	Rfctl8 select 0: Rfctl8 1: DSP1 GPIO8 2: DSP2 GPIO8 3: GPIO69
		Rfctl8_drv	[3:2]	0	Rfctl8 drive
		Rfctl8_wpu	[1]	0	Rfctl8 wpu: 0: Disable 1: Enable
		Rfctl8_wpd	[0]	0	Rfctl8 wpd: 0: Disble 1: Enable
0x8C00_0368	Rfctl9_reg	Rfctl9_sel	[5:4]	0	Rfctl9 select 0: Rfctl9 1: DSP1 GPIO9 2: DSP2 GPIO9 3: GPIO70
		Rfctl9_drv	[3:2]	0	Rfctl9 drive
		Rfctl9_wpu	[1]	0	Rfctl9 wpu: 0: Disable 1: Enable
		Rfctl9_wpd	[0]	0	Rfctl9 wpd: 0: Disble 1: Enable
0x8C00_036C	Rfctl10_reg	Rfctl10_sel	[5:4]	0	Rfctl10 select 0: Rfctl10 1: DSP1 GPIO10 2: DSP2 GPIO10 3:
		Rfctl10_drv	[3:2]	0	Rfctl10 drive
		Rfctl10_wpu	[1]	0	Rfctl10 wpu: 0: Disable 1: Enable
		Rfctl10_wpd	[0]	0	Rfctl10 wpd: 0: Disble 1: Enable
0x8C00_0370	Rfctl11_reg	Rfctl11_sel	[5:4]	0	Rfctl11 select 0: Rfctl11 1: DSP1 GPIO11 2: DSP2 GPIO11 3: GPIO71
		Rfctl11_drv	[3:2]	0	Rfctl11 drive
		Rfctl11_wpu	[1]	0	Rfctl11 wpu: 0: Disable 1: Enable
		Rfctl11_wpd	[0]	0	Rfctl11 wpd: 0: Disble 1: Enable
0x8C00_0374	Rfctl12_reg	Rfctl12_sel	[5:4]	0	Rfctl12 select 0: Rfctl12 1: DSP1 GPIO12 2: DSP2 GPIO12 3: GPIO72
		Rfctl12_drv	[3:2]	0	Rfctl12 drive
		Rfctl12_wpu	[1]	0	Rfctl12 wpu: 0: Disable 1: Enable
		Rfctl12_wpd	[0]	0	Rfctl12 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0378	Rfctl13_reg	Rfctl13_sel	[5:4]	0	Rfctl13 select 0: Rfctl13 1: DSP1 GPIO13 2: DSP2 GPIO13 3: GPIO73
		Rfctl13_drv	[3:2]	0	Rfctl13 drive
		Rfctl13_wpu	[1]	0	Rfctl13 wpu: 0: Disable 1: Enable
		Rfctl13_wpd	[0]	0	Rfctl13 wpd: 0: Disble 1: Enable
0x8C00_037C	Rfctl14_reg	Rfctl14_sel	[5:4]	0	Rfctl14 select 0: Rfctl14 1: DSP1 GPIO14 2: DSP1 GPIO14 3: GPIO74
		Rfctl14_drv	[3:2]	0	Rfctl14 drive
		Rfctl14_wpu	[1]	0	Rrfctl14 wpu: 0: Disable 1: Enable
		Rfctl14_wpd	[0]	0	Rfctl14 wpd: 0: Disble 1: Enable
0x8C00_0380	Rfctl15_reg	Rfctl15_sel	[5:4]	0	Rfctl15 select 0: Rfctl15 1: DSP1 GPIO15 2: DSP2 GPIO15 3: GPIO75
		Rfctl15_drv	[3:2]	0	Rfctl15 drive
		Rfctl15_wpu	[1]	0	Rfctl15 wpu: 0: Disable 1: Enable
		Rfctl15_wpd	[0]	0	Rfctl15 wpd: 0: Disble 1: Enable
0x8C00_0384	Rflna_reg	Rflna_sel	[5:4]	0	Rflna select 0: Rflna 1: DSP2_ABORT_N 2: 3: GPIO76
		Rflna_drv	[3:2]	0	Rflna drive
		Rflna_wpu	[1]	0	Rflna wpu: 0: Disable 1: Enable
		Rflna_wpd	[0]	0	Rflna wpd: 0: Disble 1: Enable
0x8C00_0388	Rfmixer_reg	Rfmixer_sel	[5:4]	0	Rfmixer select 0: Rfmixer 1: DSP2_STOP 2: 3: GPIO77
		Rfmixer_drv	[3:2]	0	Rfmixer drive
		Rfmixer_wpu	[1]	0	Rfmixer wpu: 0: Disable 1: Enable
		Rfmixer_wpd	[0]	0	Rfmixer wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_038C	Rfpahg_reg	Rfpahg_sel	[5:4]	0	Rfpahg select 0: Rfpahg 1: DSP2_CONT_N 2: 3: GPIO78
		Rfpahg_drv	[3:2]	0	Rfpahg drive
		Rfpahg_wpu	[1]	0	Rfpahg wpu: 0: Disable 1: Enable
		Rfpahg_wpd	[0]	0	Rfpahg wpd: 0: Disble 1: Enable
0x8C00_0390	Keyin0_reg	Keyin0_sel	[5:4]	0	Keyin0 select 0: Keyin0 1: 2: RXID 3: GPIO104
		Keyin0_drv	[3:2]	0	Keyin0 drive
		Keyin0_wpu	[1]	0	Keyin0 wpu: 0: Disable 1: Enable
		Keyin0_wpd	[0]	0	Keyin0 wpd: 0: Disble 1: Enable
0x8C00_0394	Keyin1_reg	Keyin1_sel	[5:4]	0	Keyin1 select 0: Keyin1 1: 2: 3: GPIO105
		Keyin1_drv	[3:2]	0	Keyin1 drive
		Keyin1_wpu	[1]	0	Keyin1 wpu: 0: Disable 1: Enable
		Keyin1_wpd	[0]	0	Keyin1 wpd: 0: Disble 1: Enable
0x8C00_0398	Keyin2_reg	Keyin2_sel	[5:4]	0	Keyin2 select 0: Keyin2 1: 2: RXQD 3: GPIO106
		Keyin2_drv	[3:2]	0	Keyin2 drive
		Keyin2_wpu	[1]	0	Keyin2 wpu: 0: Disable 1: Enable
		Keyin2_wpd	[0]	0	Keyin2 wpd: 0: Disble 1: Enable
0x8C00_039C	Keyin3_reg	Keyin3_sel	[5:4]	0	Keyin3 select 0: Keyin3 1: 2: VADOUT 3: GPIO107
		Keyin3_drv	[3:2]	0	Keyin3 drive
		Keyin3_wpu	[1]	0	Keyin3 wpu: 0: Disable 1: Enable
		Keyin3_wpd	[0]	0	Keyin3 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_03A0	Keyin4_reg	Keyin4_sel	[5:4]	0	Keyin4 select 0: Keyin4 1: 2: VDACD0 3: GPIO108
		Keyin4_drv	[3:2]	0	Keyin4 drive
		Keyin4_wpu	[1]	0	Keyin4 wpu: 0: Disable 1: Enable
		Keyin4_wpd	[0]	0	Keyin4 wpd: 0: Disble 1: Enable
0x8C00_03A4	Keyout0_reg	Keyout0_sel	[5:4]	0	Keyout0 select 0: Keyout0 1: 2: 3: GPIO109
		Keyout0_drv	[3:2]	0	Keyout0 drive
		Keyout0_wpu	[1]	0	Keyout0 wpu: 0: Disable 1: Enable
		Keyout0_wpd	[0]	0	Keyout0 wpd: 0: Disble 1: Enable
0x8C00_03A8	Keyout1_reg	Keyout1_sel	[5:4]	0	Keyout1 select 0: Keyout1 1: 2: VDACD1 3: GPIO110
		Keyout1_drv	[3:2]	0	Keyout1 drive
		Keyout1_wpu	[1]	0	Keyout1 wpu: 0: Disable 1: Enable
		Keyout1_wpd	[0]	0	Keyout1 wpd: 0: Disble 1: Enable
0x8C00_03AC	Keyout2_reg	Keyout2_sel	[5:4]	0	Keyout2 select 0: Keyout2 1: 2: VBCLK13 3: GPIO111
		Keyout2_drv	[3:2]	0	Keyout2 drive
		Keyout2_wpu	[1]	0	Keyout2 wpu: 0: Disable 1: Enable
		Keyout2_wpd	[0]	0	Keyout2 wpd: 0: Disble 1: Enable
0x8C00_03B0	Keyout3_reg	Keyout3_sel	[5:4]	0	Keyout3 select 0: Keyout3 1: 2: VBCLKADC 3: GPIO112
		Keyout3_drv	[3:2]	0	Keyout3 drive
		Keyout3_wpu	[1]	0	Keyout3 wpu: 0: Disable 1: Enable
		Keyout3_wpd	[0]	0	Keyout3 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_03B4	Keyout4_reg	Keyout4_sel	[5:4]	0	Keyout4 select 0: Keyout4 1: 2: 3: GPIO113
		Keyout4_drv	[3:2]	0	Keyout4 drive
		Keyout4_wpu	[1]	0	Keyout4 wpu: 0: Disable 1: Enable
		Keyout4_wpd	[0]	0	Keyout4 wpd: 0: Disble 1: Enable
0x8C00_03B8	Keyout5_reg	Keyout5_sel	[5:4]	0	Keyout5 select 0: Keyout5 1: 2: 3: GPIO114
		Keyout5_drv	[3:2]	0	Keyout5 drive
		Keyout5_wpu	[1]	0	Keyout5 wpu: 0: Disable 1: Enable
		Keyout5_wpd	[0]	0	Keyout5 wpd: 0: Disble 1: Enable
0x8C00_03BC	U0csn3_reg	U0csn3_sel	[5:4]	0	U0csn3 select 0: U0csn3 1: 2: 3: GPIO84
		U0csn3_drv	[3:2]	0	U0csn3 drive
		U0csn3_wpu	[1]	0	U0csn3 wpu: 0: Disable 1: Enable
		U0csn3_wpd	[0]	0	U0csn3 wpd: 0: Disble 1: Enable
0x8C00_03C0	U0ctsn_reg	U0ctsn_sel	[5:4]	0	U0ctsn select 0: U0ctsn 1: SSCLK 2: 3: GPIO85
		U0ctsn_drv	[3:2]	0	U0ctsn drive
		U0ctsn_wpu	[1]	0	U0ctsn wpu: 0: Disable 1: Enable
		U0ctsn_wpd	[0]	0	U0ctsn wpd: 0: Disble 1: Enable
0x8C00_03C4	U0dsrn_reg	U0dsrn_sel	[5:4]	0	U0dsrn select 0: U0dsrn 1: SSCTL0 2: 3: GPIO86
		U0dsrn_drv	[3:2]	0	U0dsrn drive
		U0dsrn_wpu	[1]	0	U0dsrn wpu: 0: Disable 1: Enable
		U0dsrn_wpd	[0]	0	U0dsrn wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_03C8	U0dtrn_reg	U0dtrn_sel	[5:4]	0	U0dtrn select 0: U0dtrn 1: SSCTL1 2: 3: GPIO87
		U0dtrn_drv	[3:2]	0	U0dtrn drive
		U0dtrn_wpu	[1]	0	U0dtrn wpu: 0: Disable 1: Enable
		U0dtrn_wpd	[0]	0	U0dtrn wpd: 0: Disble 1: Enable
0x8C00_03CC	U0rtsn_reg	U0rtsn_sel	[5:4]	0	U0rtsn select 0: U0rtsn 1: SSDR 2: 3: GPIO88
		U0rtsn_drv	[3:2]	0	U0rtsn drive
		U0rtsn_wpu	[1]	0	U0rtsn wpu: 0: Disable 1: Enable
		U0rtsn_wpd	[0]	0	U0rtsn wpd: 0: Disble 1: Enable
0x8C00_03D0	U0rxd_reg	U0rxd_sel	[5:4]	0	U0rxd select 0: U0rxd 1: SSDX 2: 3: GPIO89
		U0rxd_drv	[3:2]	0	U0rxd drive
		U0rxd_wpu	[1]	0	U0rxd wpu: 0: Disable 1: Enable
		U0rxd_wpd	[0]	0	U0rxd wpd: 0: Disble 1: Enable
0x8C00_03D4	U0txd_reg	U0txd_sel	[5:4]	0	U0txd select 0: U0txd 1: SSRST 2: 3: GPIO90
		U0txd_drv	[3:2]	0	U0txd drive
		U0txd_wpu	[1]	0	U0txd wpu: 0: Disable 1: Enable
		U0txd_wpd	[0]	0	U0txd wpd: 0: Disble 1: Enable
0x8C00_03D8	U1csn3_reg	U1csn3_sel	[5:4]	0	U1csn3 select 0: U1csn3 1: RXID 2: 3: GPIO93
		U1csn3_drv	[3:2]	0	U1csn3 drive
		U1csn3_wpu	[1]	0	U1csn3 wpu: 0: Disable 1: Enable
		U1csn3_wpd	[0]	0	U1csn3 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_03DC	U1ctsn_reg	U1ctsn_sel	[5:4]	0	U1ctsn select 0: U1ctsn 1: RXQD 2: 3: GPIO94
		U1ctsn_drv	[3:2]	0	U1ctsn drive
		U1ctsn_wpu	[1]	0	U1ctsn wpu: 0: Disable 1: Enable
		U1ctsn_wpd	[0]	0	U1ctsn wpd: 0: Disble 1: Enable
0x8C00_03E0	U1dsrn_reg	U1dsrn_sel	[5:4]	0	U1dsrn select 0: U1dsrn 1: VADOUT 2: 3: GPIO95
		U1dsrn_drv	[3:2]	0	U1dsrn drive
		U1dsrn_wpu	[1]	0	U1dsrn wpu: 0: Disable 1: Enable
		U1dsrn_wpd	[0]	0	U1dsrn wpd: 0: Disble 1: Enable
0x8C00_03E4	U1dtrn_reg	U1dtrn_sel	[5:4]	0	U1dtrn select 0: U1dtrn 1: VDACD0 2: 3: GPIO96
		U1dtrn_drv	[3:2]	0	U1dtrn drive
		U1dtrn_wpu	[1]	0	U1dtrn wpu: 0: Disable 1: Enable
		U1dtrn_wpd	[0]	0	U1dtrn wpd: 0: Disble 1: Enable
0x8C00_03E8	U1rtsn_reg	U1rtsn_sel	[5:4]	0	U1rtsn select 0: U1rtsn 1: VDACD1 2: 3: GPIO97
		U1rtsn_drv	[3:2]	0	U1rtsn drive
		U1rtsn_wpu	[1]	0	U1rtsn wpu: 0: Disable 1: Enable
		U1rtsn_wpd	[0]	0	U1rtsn wpd: 0: Disble 1: Enable
0x8C00_03EC	U1rxd_reg	U1rxd_sel	[5:4]	0	U1rxd select 0: U1rxd 1: VBCLK13 2: 3: GPIO98
		U1rxd_drv	[3:2]	0	U1rxd drive
		U1rxd_wpu	[1]	0	U1rxd wpu: 0: Disable 1: Enable
		U1rxd_wpd	[0]	0	U1rxd wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_03F0	U1txd_reg	U1txd_sel	[5:4]	0	U1txd select 0: U1txd 1: VBCLKADC 2: 3: GPIO99
		U1txd_drv	[3:2]	0	U1txd drive
		U1txd_wpu	[1]	0	U1txd wpu: 0: Disable 1: Enable
		U1txd_wpd	[0]	0	U1txd wpd: 0: Disble 1: Enable
0x8C00_03F4	U2csn3_reg	U2csn3_sel	[5:4]	0	U2csn3 select 0: U2csn3 1: D1TCK 2: D2TCK 3: GPIO117
		U2csn3_drv	[3:2]	0	U2csn3 drive
		U2csn3_wpu	[1]	0	U2csn3 wpu: 0: Disable 1: Enable
		U2csn3_wpd	[0]	0	U2csn3 wpd: 0: Disble 1: Enable
0x8C00_03F8	U2ctsn_reg	U2ctsn_sel	[5:4]	0	U2ctsn select 0: U2ctsn 1: D1TDI 2: D2TDI 3: GPIO118
		U2ctsn_drv	[3:2]	0	U2ctsn drive
		U2ctsn_wpu	[1]	0	U2ctsn wpu: 0: Disable 1: Enable
		U2ctsn_wpd	[0]	0	U2ctsn wpd: 0: Disble 1: Enable
0x8C00_03FC	U2dsm_reg	U2dsrn_sel	[5:4]	0	U2dsrn select 0: U2dsrn 1: D1TDO 2: D2TDO 3: GPIO119
		U2dsrn_drv	[3:2]	0	U2dsrn drive
		U2dsrn_wpu	[1]	0	U2dsrn wpu: 0: Disable 1: Enable
		U2dsrn_wpd	[0]	0	U2dsrn wpd: 0: Disble 1: Enable
0x8C00_0400	U2dtrn_reg	U2dtrn_sel	[5:4]	0	U2dtrn select 0: U2dtrn 1: D1TMS 2: D2TMS 3: GPIO120
		U2dtrn_drv	[3:2]	0	U2dtrn drive
		U2dtrn_wpu	[1]	0	U2dtrn wpu: 0: Disable 1: Enable
		U2dtrn_wpd	[0]	0	U2dtrn wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0404	U2rtsn_reg	U2rtsn_sel	[5:4]	0	U2rtsn select 0: U2rtsn 1: D1TINT 2: D2TINT 3: GPIO121
		U2rtsn_drv	[3:2]	0	U2rtsn drive
		U2rtsn_wpu	[1]	0	U2rtsn wpu: 0: Disable 1: Enable
		U2rtsn_wpd	[0]	0	U2rtsn wpd: 0: Disble 1: Enable
0x8C00_0408	U2rxd_reg	U2rxd_sel	[5:4]	0	U2rxd select 0: U2rxd 1: 2: 3: GPIO102
		U2rxd_drv	[3:2]	0	U2rxd drive
		U2rxd_wpu	[1]	0	U2rxd wpu: 0: Disable 1: Enable
		U2rxd_wpd	[0]	0	U2rxd wpd: 0: Disble 1: Enable
0x8C00_040C	U2txd_reg	U2txd_sel	[5:4]	0	U2txd select 0: U2txd 1: 2: 3: GPIO103
		U2txd_drv	[3:2]	0	U2txd drive
		U2txd_wpu	[1]	0	U2txd wpu: 0: Disable 1: Enable
		U2txd_wpd	[0]	0	U2txd wpd: 0: Disble 1: Enable
0x8C00_0410	Simclk_reg	Simclk_sel	[5:4]	0	Simclk select 0: Simclk 1: 2: 3: GPIO0
		Simclk_drv	[3:2]	0	Simclk drive
		Simclk_wpu	[1]	0	Simclk wpu: 0: Disable 1: Enable
		Simclk_wpd	[0]	0	Simclk wpd: 0: Disble 1: Enable
0x8C00_0414	Simda1_reg	Simda1_sel	[5:4]	0	Simda1 select 0: Simda1 1: 2: 3: GPIO1
		Simda1_drv	[3:2]	0	Simda1 drive
		Simda1_wpu	[1]	0	Simda1 wpu: 0: Disable 1: Enable
		Simda1_wpd	[0]	0	Simda1 wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0418	Simrst1_reg	Simrst1_sel	[5:4]	0	Simrst1 select 0: Simrst1 1: 2: 3: GPIO2
		Simrst1_drv	[3:2]	0	Simrst1 drive
		Simrst1_wpu	[1]	0	Simrst1 wpu: 0: Disable 1: Enable
		Simrst1_wpd	[0]	0	Simrst1 wpd: 0: Disble 1: Enable
0x8C00_041C	Simda2_reg	Simda2_sel	[5:4]	0	Simda2 select 0: Simda1 1: 2: 3: GPIO3
		Simda2_drv	[3:2]	0	Simda2 drive
		Simda2_wpu	[1]	0	Simda2 wpu: 0: Disable 1: Enable
		Simda2_wpd	[0]	0	Simda2 wpd: 0: Disble 1: Enable
0x8C00_0420	Simrst2_reg	Simrst2_sel	[5:4]	0	Simrst2 select 0: Simrst2 1: 2: 3: GPIO4
		Simrst2_drv	[3:2]	0	Simrst2 drive
		Simrst2_wpu	[1]	0	Simrst2 wpu: 0: Disable 1: Enable
		Simrst2_wpd	[0]	0	Simrst2 wpd: 0: Disble 1: Enable
0x8C00_0424	Clksmem_reg	Clksmem_sel	[5:4]	0	Clksmem select 0: Clksmem 1: Clk_dsp1 2: Clk_dsp2 3:
		Clksmem_drv	[3:2]	0	Clksmem drive
		Clksmem_wpu	[1]	0	Clksmem wpu: 0: Disable 1: Enable
		Clksmem_wpd	[0]	0	Clksmem wpd: 0: Disble 1: Enable
0x8C00_0428	Clkdmem_reg	Clkdmem_sel	[5:4]	0	Clkdmem select 0: Clkdmem 1: Clkapb 2: CLKARM9 3: GPIO22
		Clkdmem_drv	[3:2]	0	Clkdmem drive
		Clkdmem_wpu	[1]	0	Clkdmem wpu: 0: Disable 1: Enable
		Clkdmem_wpd	[0]	0	Clkdmem wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_042C	Clkrtc_reg	Clkrtc_sel	[5:4]	0	Clkrtc select 0: Clkrtc 1: Clk_qbc 2: Clk_jda 3: GPIO116
		Clkrtc_drv	[3:2]	0	Clkrtc drive
		Clkrtc_wpu	[1]	0	Clkrtc wpu: 0: Disable 1: Enable
		Clkrtc_wpd	[0]	0	Clkrtc wpd: 0: Disble 1: Enable
0x8C00_0430	Clkpll_reg	Clkpll_sel	[5:4]	0	Clkpll select 0: Clkpll 1: CLKDSP1 2: CLKDSP2 3:
		Clkpll_drv	[3:2]	0	Clkpll drive
		Clkpll_wpu	[1]	0	Clkpll wpu: 0: Disable 1: Enable
		Clkpll_wpd	[0]	0	Clkpll wpd: 0: Disble 1: Enable
0x8C00_0434	Clkdsp1_reg	Clkdsp1_sel	[5:4]	0	Clkdsp1 select 0: Clkdsp1 1: 2: 3:
Note: These two registers:clkdsp1_reg and clkdsp2_reg appear at RTL code,	but disappear at pin_list document	Clkdsp1_drv	[3:2]	0	Clkdsp1 drive
		Clkdsp1_wpu	[1]	0	Clkdsp1 wpu: 0: Disable 1: Enable
		Clkdsp1_wpd	[0]	0	Clkdsp1 wpd: 0: Disble 1: Enable
0x8C00_0438	Clkdsp2_reg	Clkdsp2_sel	[5:4]	0	Clkdsp2 select 0: Clkdsp2 1: 2: 3:
		Clkdsp2_drv	[3:2]	0	Clkdsp2 drive
		Clkdsp2_wpu	[1]	0	Clkdsp2 wpu: 0: Disable 1: Enable
		Clkdsp2_wpd	[0]	0	Clkdsp2 wpd: 0: Disble 1: Enable
0x8C00_043C	Mtck_reg	Mtck_sel	[5:4]	0	Mtck select 0: Mtck 1: D1tck 2: D2tck 3: GPIO79
		Mtck_drv	[3:2]	0	Mtck drive
		Mtck_wpu	[1]	0	Mtck wpu: 0: Disable 1: Enable
		Mtck_wpd	[0]	0	Mtck wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0440	Mtdi_reg	Mtdi_sel	[5:4]	0	Mtdi select 0: Mtdi 1: D1tdi 2: d2tdi 3: GPIO80
		Mtdi_drv	[3:2]	0	Mtdi drive
		Mtdi_wpu	[1]	0	Mtdi wpu: 0: Disable 1: Enable
		Mtdi_wpd	[0]	0	Mtdi wpd: 0: Disble 1: Enable
0x8C00_0444	Mtdo_reg	Mtdo_sel	[5:4]	0	Mtdo select 0: Mtdo 1: D1tdo 2: D2tdo 3: GPIO81
		Mtdo_drv	[3:2]	0	Mtdo drive
		Mtdo_wpu	[1]	0	Mtdo wpu: 0: Disable 1: Enable
		Mtdo_wpd	[0]	0	Mtdo wpd: 0: Disble 1: Enable
0x8C00_0448	Mtms_reg	Mtms_sel	[5:4]	0	Mtms select 0: Mtms 1: D1tms 2: D2tms 3: GPIO82
		Mtms_drv	[3:2]	0	Mtms drive
		Mtms_wpu	[1]	0	Mtms wpu: 0: Disable 1: Enable
		Mtms_wpd	[0]	0	Mtms wpd: 0: Disble 1: Enable
0x8C00_044C	Mtrstn_reg	Mtrstn_sel	[5:4]	0	Mtrstn select 0: Mtrstn 1: D1tint 2: D2tint 3: GPIO83
		Mtrstn_drv	[3:2]	0	Mtrstn drive
		Mtrstn_wpu	[1]	0	Mtrstn wpu: 0: Disable 1: Enable
		Mtrstn_wpd	[0]	0	Mtrstn wpd: 0: Disble 1: Enable
0x8C00_0450	D1tck_reg	D1tck_sel	[5:4]	0	D1tck select 0: D1tck 1: 2: 3:
		D1tck_drv	[3:2]	0	D1tck drive
		D1tck_wpu	[1]	0	D1tck wpu: 0: Disable 1: Enable
		D1tck_wpd	[0]	0	D1tck wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0454	D1tdi_reg	D1tdi_sel	[5:4]	0	D1tdi select 0: D1tdi 1: 2: 3:
		D1tdi_drv	[3:2]	0	D1tdi drive
		D1tdi_wpu	[1]	0	D1tdi wpu: 0: Disable 1: Enable
		D1tdi_wpd	[0]	0	D1tdi wpd: 0: Disble 1: Enable
0x8C00_0458	D1tdo_reg	D1tdo_sel	[5:4]	0	D1tdo select 0: D1tdo 1: 2: 3:
		D1tdo_drv	[3:2]	0	D1tdo drive
		D1tdo_wpu	[1]	0	D1tdo wpu: 0: Disable 1: Enable
		D1tdo_wpd	[0]	0	D1tdo wpd: 0: Disble 1: Enable
0x8C00_045C	D1tint_reg	D1tint_sel	[5:4]	0	D1tint select 0: D1tint 1: 2: 3:
		D1tint_drv	[3:2]	0	D1tint drive
		D1tint_wpu	[1]	0	D1tint wpu: 0: Disable 1: Enable
		D1tint_wpd	[0]	0	D1tint wpd: 0: Disble 1: Enable
0x8C00_0460	D1tms_reg	D1tms_sel	[5:4]	0	D1tms select 0: D1tms 1: 2: 3:
		D1tms_drv	[3:2]	0	D1tms drive
		D1tms_wpu	[1]	0	D1tms wpu: 0: Disable 1: Enable
		D1tms_wpd	[0]	0	D1tms wpd: 0: Disble 1: Enable
0x8C00_0464	D2tck_reg	D2tck_sel	[5:4]	0	D2tck select 0: D2tck 1: 2: 3:
		D2tck_drv	[3:2]	0	D2tck drive
		D2tck_wpu	[1]	0	D2tck wpu: 0: Disable 1: Enable
		D2tck_wpd	[0]	0	D2tck wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0468	D2tdi_reg	D2tdi_sel	[5:4]	0	D2tdi select 0: D2tdi 1: 2: 3:
		D2tdi_drv	[3:2]	0	D2tdi drive
		D2tdi_wpu	[1]	0	D2tdi wpu: 0: Disable 1: Enable
		D2tdi_wpd	[0]	0	D2tdi wpd: 0: Disble 1: Enable
0x8C00_046C	D2tdo_reg	D2tdo_sel	[5:4]	0	D2tdo select 0: D2tdo 1: 2: 3:
		D2tdo_drv	[3:2]	0	D2tdo drive
		D2tdo_wpu	[1]	0	D2tdo wpu: 0: Disable 1: Enable
		D2tdo_wpd	[0]	0	D2tdo wpd: 0: Disble 1: Enable
0x8C00_0470	D2tint_reg	D2tint_sel	[5:4]	0	D2tint select 0: D2tint 1: 2: 3:
		D2tint_drv	[3:2]	0	D2tint drive
		D2tint_wpu	[1]	0	D2tint wpu: 0: Disable 1: Enable
		D2tint_wpd	[0]	0	D2tint wpd: 0: Disble 1: Enable
0x8C00_0474	D2tms_reg	D2tms_sel	[5:4]	0	D2tms select 0: D2tms 1: 2: 3:
		D2tms_drv	[3:2]	0	D2tms drive
		D2tms_wpu	[1]	0	D2tms wpu: 0: Disable 1: Enable
		D2tms_wpd	[0]	0	D2tms wpd: 0: Disble 1: Enable
0x8C00_0478	Scl_reg	Scl_sel	[5:4]	0	Scl select 0: Scl 1: 2: 3: GPIO91
		Scl_drv	[3:2]	0	Scl drive
		Scl_wpu	[1]	0	Scl wpu: 0: Disable 1: Enable
		Scl_wpd	[0]	0	Scl wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_047C	Sda_reg	Sda_sel	[5:4]	0	Sda select 0: Sda 1: 2: 3: GPIO92
		Sda_drv	[3:2]	0	Sda drive
		Sda_wpu	[1]	0	Sda wpu: 0: Disable 1: Enable
		Sda_wpd	[0]	0	Sda wpd: 0: Disble 1: Enable
0x8C00_0480	Pwma_reg	Pwma_sel	[5:4]	0	Pwma select 0: Pwma 1: 2: 3: GPIO100
		Pwma_drv	[3:2]	0	Pwma drive
		Pwma_wpu	[1]	0	Pwma wpu: 0: Disable 1: Enable
		Pwma_wpd	[0]	0	Pwma wpd: 0: Disble 1: Enable
0x8C00_0484	Pwmb_reg	Pwmb_sel	[5:4]	0	Pwmb select 0: Pwmb 1: 2: 3: GPIO101
		Pwmb_drv	[3:2]	0	Pwmb drive
		Pwmb_wpu	[1]	0	Pwmb wpu: 0: Disable 1: Enable
		Pwmb_wpd	[0]	0	Pwmb wpd: 0: Disble 1: Enable
0x8C00_0488	Xtlen_reg	Xtlen_sel	[5:4]	0	Xtlen select 0: Xtlen 1: 2: 3: GPIO55
		Xtlen_drv	[3:2]	0	Xtlen drive
		Xtlen_wpu	[1]	0	Xtlen wpu: 0: Disable 1: Enable
		Xtlen_wpd	[0]	0	Xtlen wpd: 0: Disble 1: Enable
0x8C00_048C	Idoen_reg	Idoen_sel	[5:4]	0	Idoen select 0: Idoen 1: 2: 3: GPIO115
		Idoen_drv	[3:2]	0	Idoen drive
		Idoen_wpu	[1]	0	Idoen wpu: 0: Disable 1: Enable
		Idoen_wpd	[0]	0	Idoen wpd: 0: Disble 1: Enable

Address	Register	Signal	Bit Pos	Default	Description
0x8C00_0490	Pbint_reg	Pbint_sel	[5:4]	0	Pbint select 0: Pbint 1: 2: 3: GPIO53
		Pbint_drv	[3:2]	0	Pbint drive
		Pbint_wpu	[1]	0	Pbint wpu: 0: Disable 1: Enable
		Pbint_wpd	[0]	0	Pbint wpd: 0: Disble 1: Enable
0x8C00_0494	Rstn_reg	Rstn_sel	[5:4]	0	Rstn select 0: Rstn 1: 2: 3: GPIO54
		Rstn_drv	[3:2]	0	Rstn drive
		Rstn_wpu	[1]	0	Rstn wpu: 0: Disable 1: Enable
		Rstn_wpd	[0]	0	Rstn wpd: 0: Disble 1: Enable
0x8C00_0498	Reserve_reg				
0x8C00_049C	Reserve_reg				
0x8C00_04A0	Reserve_reg				
0x8C00_04A4	Reserve_reg				
0x8C00_04A8 0x8C00_04AC	Reserve_reg Reserve_reg				

4.23 GEA Accelerator

4.23.1 GPRS Encryption Algorithm (GEA)

GEA algorithms are options for the GPRS channels and SC8800 supports both GEA 1 and GEA 2. The GEA accelerator is controlled by the MCU. The accelerator generates the cipher bit steam, and the MCU performs the final XOR of the bit steams.

4.23.2 GEA Control Registers

GEA interface Register Base Address: 0x8D00_0000

Address	Signal	Bit Pos	Default	Description
0x8D00_0000 to 0x8D00_0C7F	KEYSTREAM TABLE	**[15:0]**	0	**3 – 1600 bytes of keystream depending on WORD_LEN**
0x8D00_0C80	GEA MODE	**[1:0]**	0	0: Encryption Disable 1: GEA I encryption 2: GEA II encryption **3: reserved**
	Reserved	**[15:2]**		

Address	Signal	Bit Pos	Default	Description
0x8D00_0C84	WORD_LEN	**[9:0]**	0	**Specify how many words (2 bytes) of keystream need to be generated. Max 800**
	Reserved	**[14:10]**		
	DIR	**[15]**	0	**Single bit DIRECTION**
0x8D00_0C88	Kc[15:0]	**[15:0]**	0	**First word of Kc**
0x8D00_0C8C	Kc[31:16]	**[15:0]**	0	**Second word of Kc**
0x8D00_0C90	Kc[47:32]	**[15:0]**	0	**Third word of Kc**
0x8D00_0C94	Kc[63:48]	**[15:0]**	0	**Fourth word of Kc**
0x8D00_0C98	INPUT[15:0]	**[15:0]**	0	**First word of INPUT**
0x8D00_0C9C	INPUT[31:16]	**[15:0]**	0	**Second word of INPUT**
0x8D00_0CA0	GEA_START		Read only	**Read this location to start GEA keystream generation**
0x8D00_0CA4	POLL_REG	**[0]**		**This is the polling register, when high, indicating GEA keystream is ready. Software is responsible for clearing it by writing 0 into this bit.**

4.24 Voice Band Codec Register

DSP Base Address: 0xBA00

DSP control VB by default. From DSP side, please set 0xBD4A = 0x00F0, turn on VB clock, before write/read VB control register.

ARM Base Address: 0x8200_3000

From ARM side, please set 0x8B000044 = 0x0F8, enable ARM access VB, turn on VB clock, before write/read VB control register.

ARM Address Offset	DSP Address	Register Name	Signal Name	Bits	Default	Description
0x0000	0xBA00	VBCTRL0				Voice band control
			VDACPG0	[3:0]	0	DL0 PGA gains, -6 to 27 dB in 3 dB steps. 0000: -6 dB 0001: -3 dB 0010: 0 dB 0011: 3 dB … 1010: 24 dB 1011: 27 dB
				[7:4]	0	Reserved

ARM Address Offset	DSP Address	Register Name	Signal Name	Bits	Default	Description
			VADPG	[11:8]	0	UL PGA gain, 0 to 36 dB in 3 dB steps. 0000: 0 dB 0001: 3 dB 0010: 6 dB 0011: 9 dB … 1011: 33dB 1100: 36 dB
			VSTPG0	[15:12]	0	DL0 Side tone gain, -30 to 0 dB in 3 dB steps. 0000: -30 dB 0001: -27 dB … 1010: 0 dB … 1111: Sleep mode
0x0004	0xBA01	VBCTL1	VDACPG1	[3:0]	0	DL1 PGA gains, -6 to 27 dB in 3 dB steps. 0000: -6 dB 0001: -3 dB 0010: 0 dB 0011: 3 dB … 1010: 24 dB 1011: 27 dB
				[11:4]	0	Reserved
			VSTPG1	[15:12]	0	DL1 Side tone gain, -30 to 0 dB in 3 dB steps. 0000: -30 dB 0001: -27 dB … 1010: 0 dB … 1111: Sleep mode
0x0008	0xBA02	VBAUX				Voice band aux control
			VADMUX	[0]	0	Input select 0: MIC input 1: Auxiliary input
			VDACOUTS	[1]	0	Output select 0: Earphone output 1: Auxiliary output
			STEREOEN	[2]	0	Stereo output enable
				[3]	0	Reserved
			MICBIAS	[5:4]	0	Microphone bias select 00: 01: 10: 11:
				[15:6]	0	Reserved

ARM Address Offset	DSP Address	Register Name	Signal Name	Bits	Default	Description
0x0010	0xBA04	VBDA0	VBDA0	[15:0]	0	Voice band DA0 input data from DSP
0x0014	0xBA05	VBDA1	VBDA1	[15:0]	0	Voice band DA1 input data from DSP
0x0018	0xBA06	VBAD	VBAD	[15:0]	0	Voice band AD output data to DSP
0x001C	0xBA07	VBBUFFSIZE	VBADBUFFSIZE	[7:0]	0	Voice band AD data buffer size, 0~159 maps to buffer size 1~160
			VBDABUFFSIZE	[15:8]	0	Voice band DA0, DA1 data buffer size, 0~159 maps to buffer size 1~160
0x0020	0xBA08	VBTEST				Voice band test control
			VBDLOOP	[2:0]	0	Voice band digital loop: 00: Normal Mode 01: Loop1 TX sigma-delta to RX sinc4 in 02: Loop2 TX LP to RX LP in 03: Loop3 TX HP to RX HP in 04: Loop4 TX DSP in to RX out 05: Loop5 RX sinc4 out to TX sinc4 in 06: Loop6 RX LP out to TX LP in 07: Loop7 RX HP out to TX HP in
			DATESTSEL	[3]	0	VB Test select, “1/0” sel DL1, DL0
			VBDAION	[4]	0	“0” DAI block is in power down “1” DAI block is active
			VBDAICLK	[5]	0	Writing 1 to this bit starts the VBDAICLK (104kHz DAI clock) on reception of the first sample. This bit is automatically reset to 0 by VBDAIRST after reception of the last sample.
			VBDAIMD	[6:7]	0	DAI mode selection: “00”: Normal operation “01”: Test of speech decoder / DTX functions (downlink) “10”: Test of speech encoder / DTX functions (uplink) “11”: Test of acoustic devices and A/D & D/A
			VBENABLE	[8]	0	Initial value of this bit is low, means after power on and system reset, voice band is disabled and DSP can program coefficient registers and RAM. Set this bit to high will enable voice band. This bit must be set to 0 when DSP writing data to coefficient RAM.

ARM Address Offset	DSP Address	Register Name	Signal Name	Bits	Default	Description
			VBRXSAT_RSTEN	[9]	0	If rx analog AD unstable, enable VB reset if 1
			VBRXHP_BYPASS	[10]	0	Bypass rx high pass filter
			VBTXHP_BYPASS0	[11]	0	Bypass tx high pass filter for DL0
			VBTXHP_BYPASS1	[12]	0	Bypass tx high pass filter for DL1
			RAMSW_NUMB	[13]	0	“1” DSP access data buffer 1, “0” DSP access data buffer 0.
			RAMSW_EN	[14]	0	“1” enable DSP read VB ADC, write DAC0, DAC1 data buffer when VBENABLE bit low.
			VBDAIMODCTL	[15]	0	“0” DAI mode input from connector, “1” DAI mode set by DSP through register VBTEST bit7~6
0x0024	0xBA09	VBADBUFFDTA	ADBUFF_?	[8:0]	0	2’s compliment number added to VBADBUFFSIZE, control AD data buffer change for next block.
				[15:9]		Reserved
0x0028	0xBA0A	VBANAP				Voice band analog control
			PDVADC	[0]	1	Power down Voice band ADC digital/analog
			PDVDAC0	[1]	1	Power down Voice band DAC DL0 digital/analog
			PDVDAC1	[2]	1	Power down Voice band DAC DL1 digital/analog
				[3]	0	Reserved
			ALOOP1	[4]	0	Voice band Analog loop 1, MIC PGA output to earphone PGA input.
			ALOOP2	[5]	0	Voice band Analog loop 2, ADC sigma-delta to analog filter
			VBCALOOP	[6]	0	Analog calibration loop
				[15:7]	0	Reserved
0x0024	0xBA0B	VBDABUFFDTA	DABUFF_?	[8:0]	0	2’s compliment number added to VBDABUFFSIZE, control DA0,1 data buffer change for next block.
				[11:9]		Reserved
			VBADDMA_EN	[12]	0	“1” enable ADC data buffer read by DMA.
			VBDA0DMA_EN	[13]	0	“1” enable DAC0 data buffer read by DMA.
			VBDA1DMA_EN	[14]	0	“1” enable DAC1 data buffer read by DMA.
				[15]		Reserved
0x0030	0xBA0C	VBANACS				Voice band analog control

ARM Address Offset	DSP Address	Register Name	Signal Name	Bits	Default	Description
			VADOSCP	[5:0]	0	Waiting 1~63 1MHz cycle then set VOSCRST when analog VADSOC signal high
				[7:6]	0	Reserved
			VBCALP	[14:8]	0	Analog calibration period, 0~127 1Mhz clock cycle, resolution 8ms, delay 0~1s.
			VBCALEN	[15]	0	Enable Analog calibration
0x0034	0xBA0D	VBADCNT	VBADCNT	[7:0]	0 Read only	ADC data buffer counter
				[15:0]		Reserved
0x0038	0xBA0E	VBSTATUS				Voice band status
			DAISSRSTN	[0]	0 Read only	DAI SSRST_n input from SS
				[3:1]	0	Reserved
			DAIMODE	[5:4]	0 Read only	When VBDAIMODCTL="0", DSP read connector input value. When VBDAIMODCTL="1", DSP read VBTEST bit7~6 value.
				[15:6]	0	Reserved
0x003C	0xBA0F	VBDACNT	VBDA0CNT	[7:0]	0 Read only	DAC0 data buffer counter
			VBDA1CNT	[15:8]	0 Read only	DAC1 data buffer counter
0x0040	0xBA10	DACOEF0		[15:0]	0	Digital DAC coefficient 0 bit15~0
0x0044	0xBA11	DACOEF1		[15:0]	0	Digital DAC coefficient 1 bit15~0
0x0048	0xBA12	DACOEF2		[15:0]	0	Digital DAC coefficient 2 bit15~0
0x004C	0xBA13	DACOEFTOP				DAC coefficients, upper bits
			DACOEF0TOP	[3:0]	0	Digital DAC coefficient 0 bit19~16
			DACOEF1TOP	[7:4]	0	Digital DAC coefficient 1 bit19~16
			DACOEF2TOP	[11:8]	0	Digital DAC coefficient 2 bit19~16
				[15:12]	0	Reserved
0x0200	0xBA80	VOICE_IIR_COEF		[15:0]	X	Voice band IIR filter coefficient 0
		COEFRAM		[15:0]	X	…
0x02FC	**0xBAFF**	**COEFRAM 127**		[15:0]	X	Coefficient 128 (total of 128 data)

4.25 JTAG Interface

The JTAG interface can be used either by the MCU or by the DSP. The power-on default setting is for the MCU. The MCU can then write a control register to switch the JTAG interface to the DSP side.

The JTAG interface consists of the following pins.

- MTDO, test data output.
- MTDI, test data input.
- MTCK, test clock.
- MTMS, test mode.
- MTRST_N, test reset, active low. When used on the DSP side, it functions as DTINT, or test interrupt.

When not used for JTAG, these pins can be used for UART 0 hardware flow control or as GPIO pins.

4.26 Strapping Pins

The strapping pins are used for setting up the operation modes of the chip. The basic mechanism is to use the hardware reset to latch the high or low values at the pins.

4.26.1 List of Strapping Pins

The following table lists the strapping pins on SC8800.

Table 40: List of strapping pins

Pin Name	Default Pin Value	Strapping Function	Default Function Value	SW Control	Pull-up To	Comments
RFCTL0	Low	TEST_PIN_SEL [0]	0	No	VIO	No rst at IE
RFCTL1	Low	TEST_PIN_SEL [1]	0	No	VIO	No rst at IE
RFCTL2	Low	TEST_BIST_EN	0	No	VIO	
RFCTL3	Low	LDO_DMEM_B0	1	No	VIO	Default to 1.8 V
RFCTL6	Low	LDO_DMEM_B1	1	No	VIO	Default to 1.8 V
RFCTL4	Low	ANALOG_OUT_SEL_MF [0]	0	No	VIO	
RFCTL5	Low	ANALOG_OUT_SEL_MF [1]	0	No	VIO	
RFCTL7	Low	LDO_IO_B0	0	No	VIO	
RFCTL8	Low	LDO_IO_B1	0	No	VIO	
RFCTL9	Low	LDO_SMEM_B0	0	No	VIO	
RFCTL10	Low	LDO_SMEM_B1	0	No	VIO	No rst at IE
RFCTL11	Low	PROD_DRV [0]	0	No	VIO	
RFCTL12	Low	PROD_DRV [1]	0	No	VIO	
RFCTL14	Low	ARM_BOOT_MD0	0	No	VIO	
RFCTL15	Low	ARM_BOOT_MD1	0	No	VIO	
PRODTEST	-	JTAG_PIN_EN	-	No	VBAT	This pin can be connected with pull-down or left open for normal operation.
ANATEST	-	TEST_ANALOG_EN	-	No	VBAT	This pin can be connected with pull-down or left open for normal

Pin Name	Default Pin Value	Strapping Function	Default Function Value	SW Control	Pull-up To	Comments
						operation.
KEYOUT [0]	Low	REMAP [0]	0	Yes	VIO	See Table 34
KEYOUT [1]	Low	REMAP [1]	0	Yes	VIO	See Table 34

4.26.2 Boot Mode Control

The following table lists the ARM boot modes and the control through the strapping pins.

Table 41: Boot mode control

Boot mode	Strapping	Sub mode	SC8800A1	SC8800A2
FLASH_BOOT	Keyout1: pwd Keyout0: pwd			(Default)
ROM_BOOT	Keyout1: pwd Keyout0: pwu	Rfctl15: pwd Rfctl14: pwd	UART	UART (Default)
		Rfctl15: pwd Rfctl14: pwu	NAND8_256B_3CYCLE	NAND8_256B_3CYCLE
		Rfctl15: pwu Rfctl14: pwd	NAND16_256B_3CYCLE NAND16_256B_4CYCLE	NAND8_2048B_4CYCLE NAND8_2048B_5CYCLE
		Rfctl15: pwu Rfctl14: pwu	NAND8_256B_4CYCLE	NAND8_256B_4CYCLE
SDRAM BOOT	Keyout1: pwu Keyout0: pwd			
FLASH BOOT	Keyout1: pwu Keyout0: pwu			

4.26.3 Control of Strapping Pins

There are several important aspects of the control of the strapping pins.
Only hardware reset can set the strapping pins. Software reset should not change the strapping controls.
Some strapping values can be changed by software.

5 Package Outline

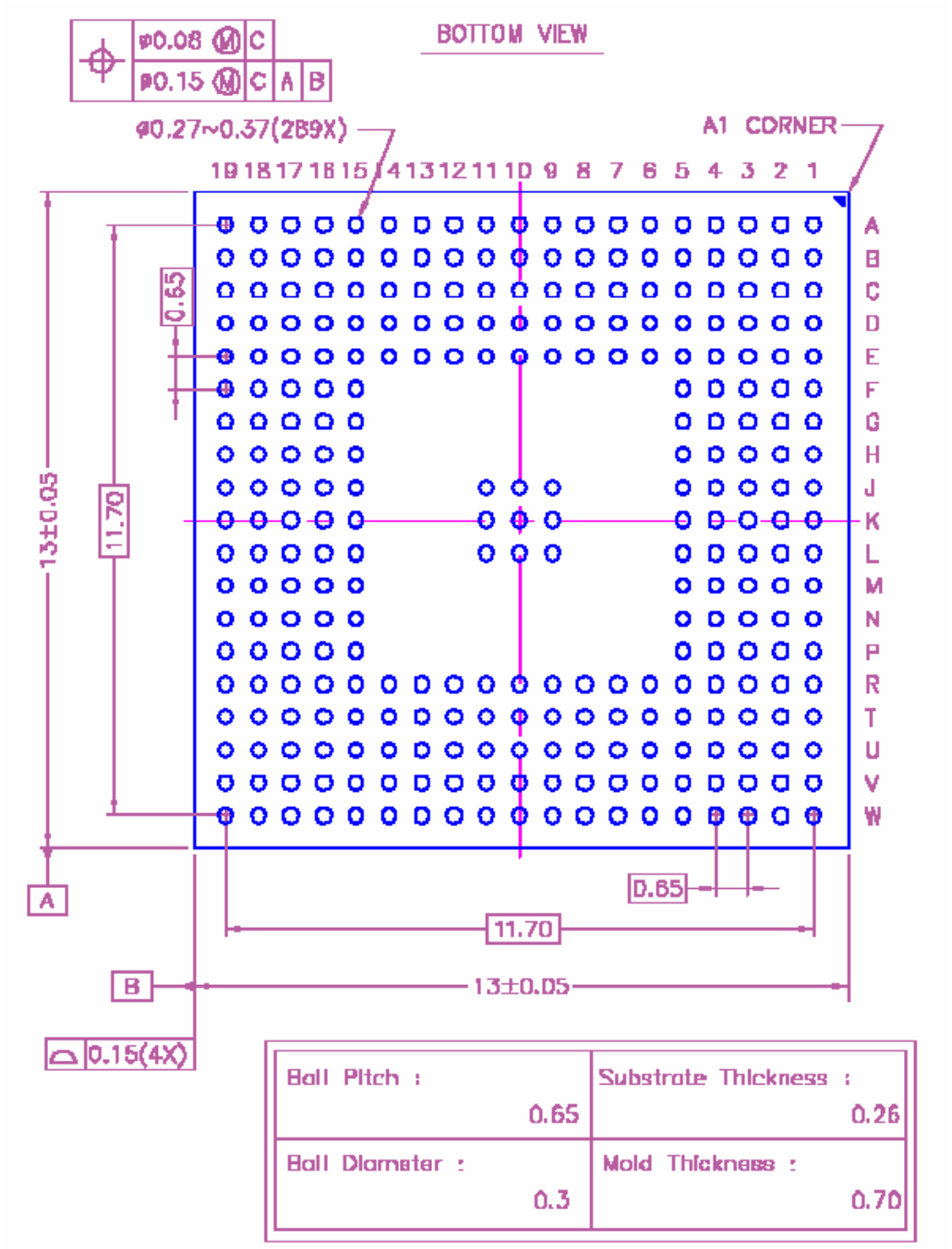


Figure 21: SC8800 289-ball package mechanical drawing.

6 Reflow Profile

This profile is designed for use with Sn63 or Sn62 and can serve as a general guideline in establishing a reflow profile.

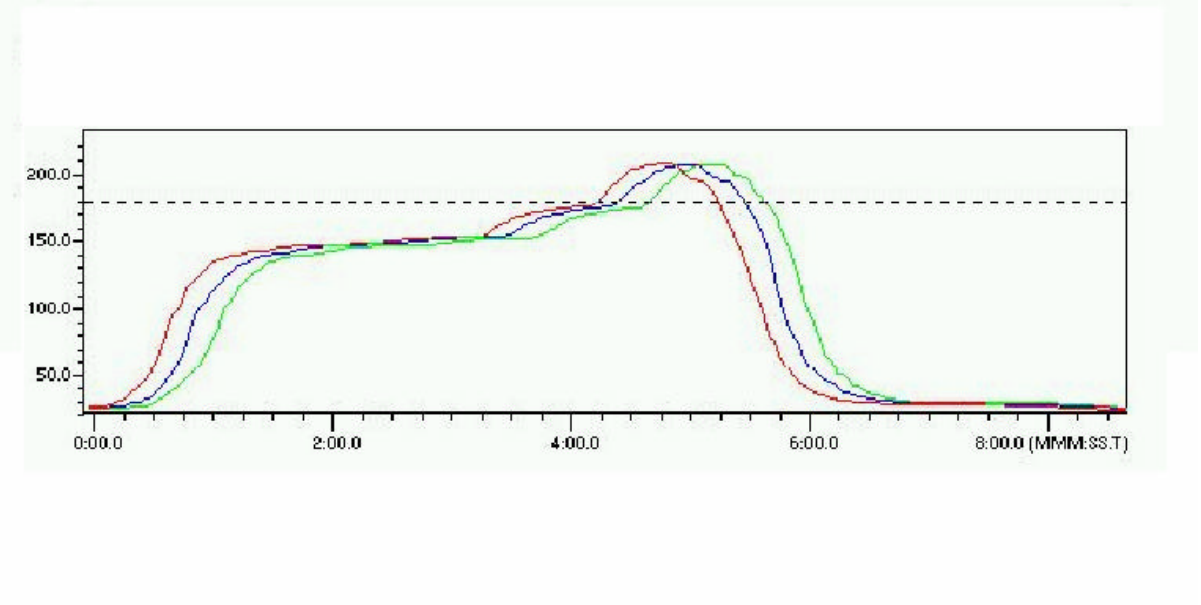


Figure 22: Reflow profile recommended for 63/37 Sn/Pb solder paste or Cu lead frame.

Heat-up @ 1 ~ 3 ºC per sec to 140 ºC
Preheat @ 140 ~ 150 ºC for 120 ~ 160 sec
Ramp @ 2 ~ 3 ºC per sec to peak temperature (220 ~ 225 ºC), temperature over 183 ºC for 45 ~ 75 sec
Cool down to room temperature @ 4 ~ 2 ºC per sec to avoid undesired intermatalic compound layer.