COMM_REG IP SPEC

Table of Contents

Introduction

The COMM_REG module is to receive writing signals and change internal registers, and to generate setting bits for other modules, or to read out data for daisy chain or SPI interface. For fault register bits, excepting for writing from interface, inputs also come from fault inputs.

Feature

Key features of the COMM_REG module are:

- support writing registers
- support MTP writing and can load data from MTP
- read-only bits can be read out by communication interface
- fault registers can be set by fault inputs, and cleared by writing via interface

Register Definition

Register Map

	Covered by MTP_CRC?	Covered by CONF_CRC?
SPF_CONF		Y
SF_CONF		Y
LF_SM_CONF		Y
SPF_FUNC		
SF_FUNC		
LF_SM_FUNC		
TM		Y
TRIM	Y	Y
FLT_SUM		
FLT_BIT		

Table1 block class1

Name	Add	D7	D6	D5	D4	D3	D2	D1	D0	Default	Block
DEV_ADD1	0x0000	TOP_DEV	ADD<6:0>							80	SPF_CONF
DEV_ADD2	0x0001		NUM<6:0>							00	SPF_CONF
COMM_CONF1	0x0002	CB_STL<4:0>							I2C_MAS_EN	08	SPF_CONF
COMM_CONF2	0x0003	COMN_TX_DIS	COMS_TX_DIS	STACK_RESP_CMD<5:0>						00	SPF_CONF
COMM_TO	0x0004	LCTO_SEL<1:0>		LONG<2:0>			SHORT<2:0>			BB	SF_CONF
COW_DET	0x0005					C_OW_TDIS<3:0>				00	LF_SM_CONF
GAP_CMP1	0x0006	GPIO_GAP_THR<2:0>			CELL_GAP_THR<4:0>					00	LF_SM_CONF
GAP_CMP2	0x0007				CELL_GAP_DEGL<4:0>					07	LF_SM_CONF
ADC_CONF1	0x0008	DLPF_FC<2:0>				ADC_CLK<1:0>		ADC_MODE<2:0>		09	SPF_CONF

Name	Add	D7	D6	D5	D4	D3	D2	D1	D0	Default	Block
ADC_CONF2	0x0009	CH_TOP_STL<3:0>				CH_STL<3:0>				55	SPF_CONF
ADC_CONF3	0x000A	CH_BOT_STL<3:0>					ADC_CHP_EN	CH_DT<1:0>		55	SPF_CONF
ADC_CONF4	0x000B	GPIO7_REF_SEL	GPIO6_REF_SEL	GPIO5_REF_SEL	GPIO4_REF_SEL	GPIO3_REF_SEL	GPIO2_REF_SEL	GPIO1_REF_SEL	GPIO0_REF_SEL	00	SPF_CONF
ADC_CONF5	0x000C					GPIO11_REF_SEL	GPIO10_REF_SEL	GPIO9_REF_SEL	GPIO8_REF_SEL	00	SPF_CONF
MON_CONF	0x000D	MON_WAKE_PERIOD<5:0>							MON_WAKE_EN	45	SF_CONF
OVUV_OTUT_CONF1	0x000E	OVUV_OTUT_EN			OVUV_ DEGL<4:0>					80	LF_SM_CONF
OVUV_OTUT_CONF2	0x000F		OV_THR<6:0>							60	LF_SM_CONF
OVUV_OTUT_CONF3	0x0010		UV_THR <6:0>							60	LF_SM_CONF
OVUV_OTUT_CONF4	0x0011	OT_ PACK_THR<4:0>					UT_ PACK_THR<2:0>			7E	LF_SM_CONF
OVUV_OTUT_CONF5	0x0012	OT_ PCB_THR<4:0>					UT_ PCB_THR<2:0>			7E	LF_SM_CONF
OVUV_OTUT_CONF6	0x0013	GPIO7_THR_SEL	GPIO6_THR_SEL	GPIO5_THR_SEL	GPIO4_THR_SEL	GPIO3_THR_SEL	GPIO2_THR_SEL	GPIO1_THR_SEL	GPIO0_THR_SEL	00	LF_SM_CONF
OVUV_OTUT_CONF7	0x0014					GPIO11_THR_SEL	GPIO10_THR_SEL	GPIO9_THR_SEL	GPIO8_THR_SEL	00	LF_SM_CONF
CB_CONF1	0x0015	CB_ROT_PACK_THR<4:0>					CB_DUTY<2:0>			7F	LF_SM_CONF
CB_CONF2	0x0016	CB_ROT_PCB_THR<4:0>					CB_PERIOD<2:0>			78	LF_SM_CONF
CB_CONF3	0x0017	CB_JOT_THR<3:0>				ROT_PAUSE_EN	ADC_PAUSE_EN	JOT_PAUSE_EN	FLT_STOP_EN	00	LF_SM_CONF
TWARN_CONF	0x0018						TWARN_THR<2:0>			03	LF_SM_CONF
GPIO_PUPD_EN1	0x0019	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	00	SPF_CONF
GPIO_PUPD_EN2	0x001A					GPIO11	GPIO10	GPIO9	GPIO8	00	SPF_CONF
GPIO_ASDIN_EN1	0x001B	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	00	SPF_CONF
GPIO_ASDIN_EN2	0x001C					GPIO11	GPIO10	GPIO9	GPIO8	00	SPF_CONF
GPIO_WEAK_PUPD_EN1	0x001D	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	00	SPF_CONF
GPIO_WEAK_PUPD_EN2	0x001E					GPIO11	GPIO10	GPIO9	GPIO8	00	SPF_CONF
FLT_SUM_MSK	0x0100				COW_FLT_MSK	SYS_FLT_MSK	GAP_FLT_MSK	OTUT_FLT_MSK	OVUV_FLT_MSK	00	SF_CONF
COW_FLT_MSK1	0x0101	C7_MSK	C6_MSK	C5_MSK	C4_MSK	C3_MSK	C2_MSK	C1_MSK	C0_MSK	00	LF_SM_CONF
COW_FLT_MSK2	0x0102	C15_MSK	C14_MSK	C13_MSK	C12_MSK	C11_MSK	C10_MSK	C9_MSK	C8_MSK	00	LF_SM_CONF
COW_FLT_MSK3	0x0103						C18_MSK	C17_MSK	C16_MSK	00	LF_SM_CONF
OV_FLT_MSK1	0x0104	CH8_MSK	CH7_MSK	CH6_MSK	CH5_MSK	CH4_MSK	CH3_MSK	CH2_MSK	CH1_MSK	00	SF_CONF
OV_FLT_MSK2	0x0105	CH16_MSK	CH15_MSK	CH14_MSK	CH13_MSK	CH12_MSK	CH11_MSK	CH10_MSK	CH9_MSK	00	SF_CONF
OV_FLT_MSK3	0x0106							CH18_MSK	CH17_MSK	00	SF_CONF
UV_FLT_MSK1	0x0107	CH8_MSK	CH7_MSK	CH6_MSK	CH5_MSK	CH4_MSK	CH3_MSK	CH2_MSK	CH1_MSK	00	SF_CONF
UV_FLT_MSK2	0x0108	CH16_MSK	CH15_MSK	CH14_MSK	CH13_MSK	CH12_MSK	CH11_MSK	CH10_MSK	CH9_MSK	00	SF_CONF
UV_FLT_MSK3	0x0109							CH18_MSK	CH17_MSK	00	SF_CONF
CELL_GAP_FLT_MS	0x010A	CH8_MSK	CH7_MSK	CH6_MSK	CH5_MSK	CH4_MSK	CH3_MSK	CH2_MSK	CH1_MSK	00	LF_SM_CONF

Name	Add	D7	D6	D5	D4	D3	D2	D1	D0	Default	Block
K1											
CELL_GAP_FLT_MSK2	0x010B	CH16_MSK	CH15_MSK	CH14_MSK	CH13_MSK	CH12_MSK	CH11_MSK	CH10_MSK	CH9_MSK	00	LF_SM_CONF
CELL_GAP_FLT_MSK3	0x010C							CH18_MSK	CH17_MSK	00	LF_SM_CONF
GPIO_GAP_FLT_MSK1	0x010D	GPIO7_MSK	GPIO6_MSK	GPIO5_MSK	GPIO4_MSK	GPIO3_MSK	GPIO2_MSK	GPIO1_MSK	GPIO0_MSK	00	LF_SM_CONF
GPIO_GAP_FLT_MSK2	0x010E					GPIO11_MSK	GPIO10_MSK	GPIO9_MSK	GPIO8_MSK	00	LF_SM_CONF
OT_FLT_MSK1	0x010F	GPIO7_MSK	GPIO6_MSK	GPIO5_MSK	GPIO4_MSK	GPIO3_MSK	GPIO2_MSK	GPIO1_MSK	GPIO0_MSK	00	SF_CONF
OT_FLT_MSK2	0x0110					GPIO11_MSK	GPIO10_MSK	GPIO9_MSK	GPIO8_MSK	00	SF_CONF
UT_FLT_MSK1	0x0111	GPIO7_MSK	GPIO6_MSK	GPIO5_MSK	GPIO4_MSK	GPIO3_MSK	GPIO2_MSK	GPIO1_MSK	GPIO0_MSK	00	SF_CONF
UT_FLT_MSK2	0x0112					GPIO11_MSK	GPIO10_MSK	GPIO9_MSK	GPIO8_MSK	00	SF_CONF
SYS_FLT_MSK1	0x0113					TBYTE_FAST_MSK	TBYTE_TO_MSK	FCOMM_FLT_MSK	FR_CRC_MSK	00	LF_SM_CONF
SYS_FLT_MSK2	0x0114	TWARN_MSK	AGND_OW_MSK	CONF_CRC_MSK	MTP_CRC_MSK	CP_OV_MSK	CP_UV_MSK	VAA_OV_MSK	VAA_UV_MSK	00	LF_SM_CONF
SYS_FLT_MSK3	0x0115			VDD_OV_MSK	VDD_UV_MSK	CLK_256K_OKB_MSK	VDD_OKB_MSK	VAA_OKB_MSK	LCTO_SD_MSK	00	SF_CONF
SYS_FLT_MSK4	0x0116	CB_CONF_FLT_MSK	SCTO_MSK	LCTO_MSK		RX_FIFO_OF_MSK	TX_FIFO_OF_MSK	TX_FIFO_UF_MSK	CMP_FLT_MSK	00	SF_CONF
SYS_FLT_MSK5	0x0117						HBFAST_MSK	HBTO_MSK	FLT_TONE_DET_MSK	00	SF_CONF
MTP_ADR_MANU	0x1FFB		ADR_MANU<6:0>							X	SPF_FUNC
MTP_DIN_MANU	0x1FFC	DIN_MANU<7:0>								X	SPF_FUNC
MTP_DOUT	0x1FFD	DOUT<7:0>								X	SPF_FUNC
MTP_TEST	0x1FFE	CLEN	CS_MANU	MANU	SRL	MRGN	RD_MANU	WR_MANU	HVEN	X	SPF_FUNC
MTP_CTRL	0x1FFF				ECED	WR_NVM			BUSY	X	SPF_FUNC
TM_REGn (n=1-10)	0x0800-0x0809									X	TM
TRIM_ADC	0x1000									X	TRIM
…		…								X	TRIM
TRIM_ANA										X	TRIM
XY_LOT										X	TRIM
TRIM_CRC	0x107F									X	TRIM
CONF_ CRC_H	0x2000	CRC<15:8>								00	LF_SM_FUNC
CONF_ CRC_L	0x2001	CRC<7:0>								00	LF_SM_FUNC
CTRL1	0x2002	SRSTB	DIR_SEL		WAKE_TONE_GEN	STA_TONE_GEN	SD_TONE_GEN	TO_SD	TO_SLEEP	80	SPF_FUNC
CTRL2	0x2003						CMP_BIST_GO	ADD_W_EN	SPI_DIR	00	SF_FUNC
ADC_CTRL	0x2004	MON_WAKE_GO					FREEZE	ADC_SGLE_GO	ADC_CNTI_GO	00	SPF_FUNC
CB_CTRL	0x2005						CB_MANU	CB_PAUSE	CB_GO	00	LF_SM_FUNC
CBFET_EN1	0x2006	CH8	CH7	CH6	CH5	CH4	CH3	CH2	CH1	00	LF_SM_FUNC
CBFET_EN2	0x2007	CH16	CH15	CH14	CH13	CH12	CH11	CH10	CH9	00	LF_SM_FUNC

Name	Add	D7	D6	D5	D4	D3	D2	D1	D0	Default	Block
CBFET_EN3	0x2008							CH18	CH17	00	LF_SM_FUNC
CB_TO_CHn_H (n=1-18)	0x2009	UNIT						TO<9:8>		83	LF_SM_FUNC
CB_TO_CHn_L (n=1-18)	0x200A	TO<7:0>								FF	LF_SM_FUNC
GPIO_PUPD1	0x202D	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	00	SF_FUNC
GPIO_ PUPD2	0x202E					GPIO11	GPIO10	GPIO9	GPIO8	00	SF_FUNC
DIAG_CTRL	0x2100								C_OW_DET_GO	00	LF_SM_FUNC
GPIO_WEAK_PUPD 1	0x2101	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	00	SF_FUNC
GPIO_WEAK_PUPD 2	0x2102					GPIO11	GPIO10	GPIO9	GPIO8	00	SF_FUNC
CS_EN_MANU1	0x2103	CS7_EN	CS6_EN	CS5_EN	CS4_EN	CS3_EN	CS2_EN	CS1_EN	CS0_EN	00	SPF_FUNC
CS_EN_MANU2	0x2104	CS15_EN	CS14_EN	CS13_EN	CS12_EN	CS11_EN	CS10_EN	CS9_EN	CS8_EN	00	SPF_FUNC
CS_EN_MANU3	0x2105						CS18_EN	CS17_EN	CS16_EN	00	SPF_FUNC
I2C_MAS_CTRL	0x2200	STOP	RX	SR	ACK				TX	00	SF_FUNC
I2C_TR	0x2201	DATA<7:0>								00	SF_FUNC
I2C_RD	0x2202	DATA<7:0>								00	SF_FUNC
RR_CNT_H	0x3FFE	CNT<15:8>								00	LF_SM_FUNC
RR_CNT_L	0x3FFF	CNT<7:0>								00	LF_SM_FUNC
CHn_LFP_H (n=1-18)	0x4000									00	SPF_FUNC
CHn_ LFP_L (n=1-18)	0x4001									00	SPF_FUNC
…											
AUX_CHn_ LFP_H (n=1-18)	0x4024									00	LF_SM_FUNC
AUX_CHn_ LFP_L (n=1-18)	0x4025									00	LF_SM_FUNC
…											
CHn_H (n=1-18)	0x4048									00	LF_SM_FUNC
CHn_L (n=1-18)	0x4049									00	LF_SM_FUNC
…											
AUX_CHn_H (n=1-18)	0x406C									00	SF_FUNC
AUX_CHn_L (n=1-18)	0x404D									00	SF_FUNC
…											
GPIOn_H (n=0-11)	0x4090									00	SPF_FUNC
GPIOn_L (n=0-11)	0x4091									00	SPF_FUNC
…											
AUX_GPIOn_H (n=0-11)	0x40A8									00	LF_SM_FUNC

Name	Add	D7	D6	D5	D4	D3	D2	D1	D0	Default	Block
AUX_GPIOn_L (n=0-11)	0x40A9									00	LF_SM_FUNC
…											
VPTAT_H	0x40C0									00	LF_SM_FUNC
VPTAT_L	0x40C1									00	LF_SM_FUNC
BG_H	0x40C2									00	LF_SM_FUNC
BG_L	0x40C3									00	LF_SM_FUNC
BG2_H	0x40C4									00	LF_SM_FUNC
BG2_L	0x40C5									00	LF_SM_FUNC
FR_CNT_H	0x5000	CNT<15:8>								00	LF_SM_FUNC
FR_CNT_L	0x5001	CNT<7:0>								00	LF_SM_FUNC
GPIO_DIN1	0x5002	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	00	SF_FUNC
GPIO_DIN2	0x5003					GPIO11	GPIO10	GPIO9	GPIO8	00	SF_FUNC
CB_CH_EN_FULL_DUTY1	0x5004	CH8	CH7	CH6	CH5	CH4	CH3	CH2	CH1	00	SF_FUNC
CB_CH_EN_FULL_DUTY2	0x5005	CH16	CH15	CH14	CH13	CH12	CH11	CH10	CH9	00	SF_FUNC
CB_CH_EN_FULL_DUTY3	0x5006							CH18	CH17	00	SF_FUNC
CB_ODD_CNT_H	0x5007	CNT<15:8>								00	SF_FUNC
CB_ODD_CNT_L	0x5008	CNT<7:0>								00	SF_FUNC
CB_EVEN_CNT_H	0x5009	CNT<15:8>								00	SF_FUNC
CB_EVEN_CNT_L	0x500A	CNT<7:0>								00	SF_FUNC
FLT_SUM	0x5100				COW_FLT	SYS_FLT	GAP_FLT	OTUT_FLT	OVUV_FLT	00	FLT_SUM
COW_FLT1	0x5101	C7	C6	C5	C4	C3	C2	C1	C0	00	FLT_BIT
COW_FLT2	0x5102	C15	C14	C13	C12	C11	C10	C9	C8	00	FLT_BIT
COW_FLT3	0x5103						C18	C17	C16	00	FLT_BIT
OV_FLT1	0x5104	CH8	CH7	CH6	CH5	CH4	CH3	CH2	CH1	00	FLT_BIT
OV_FLT2	0x5105	CH16	CH15	CH14	CH13	CH12	CH11	CH10	CH9	00	FLT_BIT
OV_FLT3	0x5106							CH18	CH17	00	FLT_BIT
UV_FLT1	0x5107	CH8	CH7	CH6	CH5	CH4	CH3	CH2	CH1	00	FLT_BIT
UV_FLT2	0x5108	CH16	CH15	CH14	CH13	CH12	CH11	CH10	CH9	00	FLT_BIT
UV_FLT3	0x5109							CH18	CH17	00	FLT_BIT
CELL_GAP_FLT1	0x510A	CH8	CH7	CH6	CH5	CH4	CH3	CH2	CH1	00	FLT_BIT
CELL_GAP_FLT2	0x510B	CH16	CH15	CH14	CH13	CH12	CH11	CH10	CH9	00	FLT_BIT
CELL_GAP_FLT3	0x510C							CH18	CH17	00	FLT_BIT
GPIO_GAP_FLT1	0x510D	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	00	FLT_BIT
GPIO_GAP_FLT2	0x510E					GPIO11	GPIO10	GPIO9	GPIO8	00	FLT_BIT
OT_FLT1	0x510F	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	00	FLT_BIT
OT_FLT2	0x5110					GPIO11	GPIO10	GPIO9	GPIO8	00	FLT_BIT

Name	Add	D7	D6	D5	D4	D3	D2	D1	D0	Default	Block
UT_FLT1	0x5111	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	00	FLT_BIT
UT_FLT2	0x5112					GPIO11	GPIO10	GPIO9	GPIO8	00	FLT_BIT
SYS_FLT1	0x5113					TBYTE_FAST	TBYTE_TO	FCOMM_FLT	FR_CRC	00	FLT_BIT
SYS_FLT2	0x5114	TWARN	AGND_OW	CONF_CRC	MTP_CRC	CP_OV	CP_UV	VAA_OV	VAA_UV	00	FLT_BIT
SYS_FLT3	0x5115			VDD_OV	VDD_UV	CLK_256K_OKB	VDD_OKB	VAA_OKB	LCTO_SD	00	FLT_BIT
SYS_FLT4	0x5116	CB_CONF_FLT	SCTO	LCTO		RX_FIFO_OF	TX_FIFO_OF	TX_FIFO_UF	CMP_FLT	00	FLT_BIT
SYS_FLT5	0x5117						HBFAST	HBTO	FLT_TONE_DET	00	FLT_BIT
TM_KEY	0x6000	KEY<7:0>								00	SF_FUNC

Table2 Register Map2

Functional Details

Block Diagram

The following diagram shows the COMM_REG architecture and internal modules and connections.

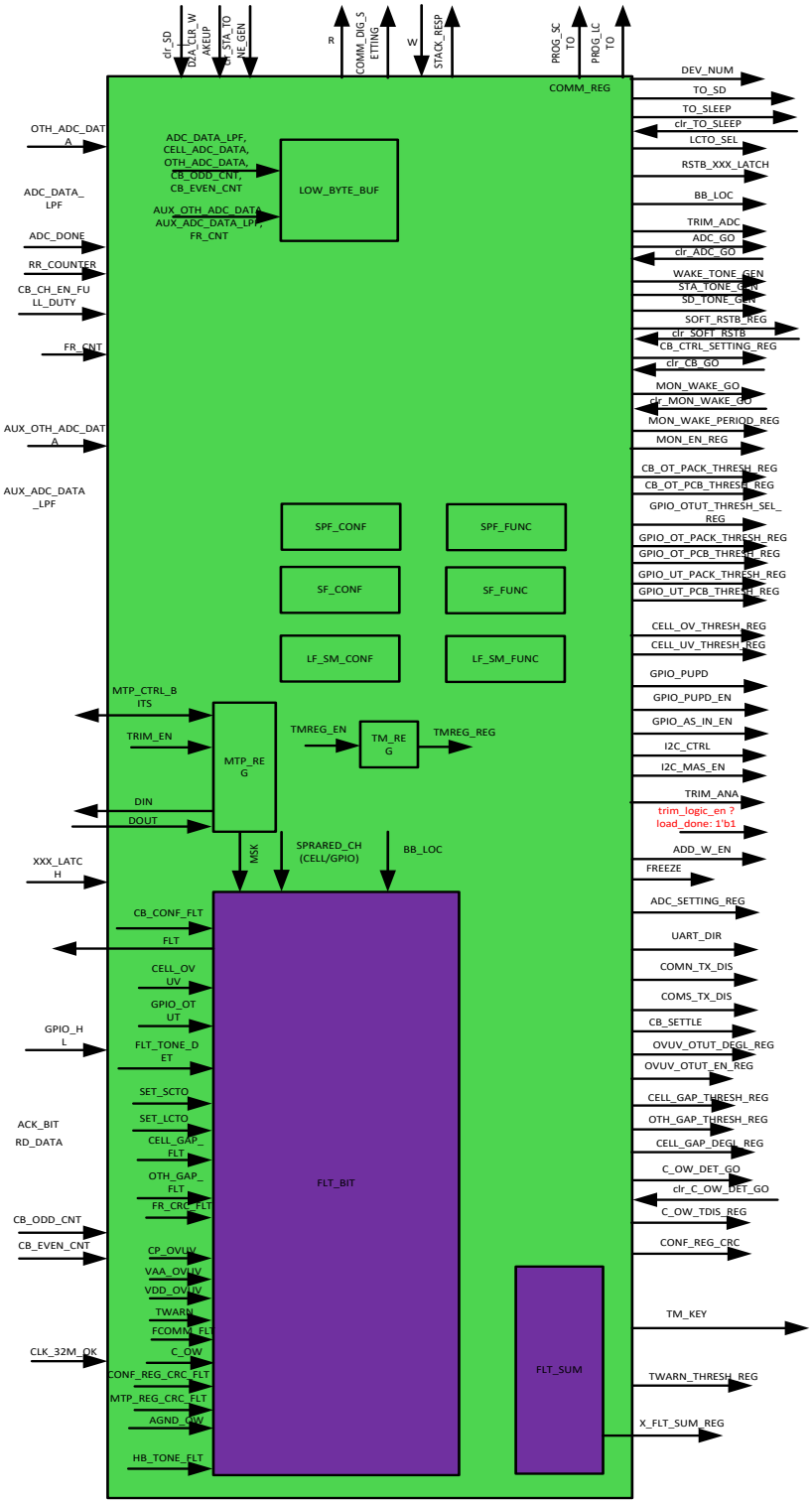

Figure1 COMM_REG diagram

Module input/output list

Name	Dir	Width	Description	duration
DEV_ADD	O	7	Device address	Level(CLK_REG domain)
COMN_TX_DIS	O	1	N port transmission disable	Level(CLK_REG domain)
COMS_TX_DIS	O	1	S port transmission disable	Level(CLK_REG domain)
SPI_DIR_REG	O	1	1: spi replace N port; 0: spi replace S port	Level(CLK_REG domain)
WAKE_TONE_GEN	O	1	Wake tone generation	Level(CLK_REG domain)
FREEZE	O	1	ADC results freeze for reading	Level(CLK_REG domain)
DIR_SEL	O	1	Indicate the receiving data direction 1: rx from N port; 0: rx from S port	Level(CLK_REG domain)
STA_TONE_GEN	O	1	STA tone generation	Level(CLK_REG domain)
TO_SD	O	1	For analog	Level(CLK_REG domain)
TO_SLEEP	O	1	For analog	Level(CLK_REG domain)
read_data	O	8	Read data for daisy chain or SPI interface	Level(CLK_REG domain)
reg0000~reg001E	O	8*31	Register bits for CONF_CRC	Level(CLK_REG domain)
reg0100~reg0117	O	8*24	Register bits for CONF_CRC	Level(CLK_REG domain)
D2A_TOP_DEV	O	1	Top device in system	Level(CLK_REG domain)
D2A_LCTO_SEL01	O	1	For analog	Level(CLK_REG domain)
D2A_LCTO_SEL10	O	1	For analog	Level(CLK_REG domain)
PROG_LCTO	O	3	Program bits for Long timeout	Level(CLK_REG domain)
PROG_SCTO	O	3	Program bits for Long timeout	Level(CLK_REG domain)
DLPF_FC_REG	O	3	Digital low pass filter stage setting	Level(CLK_REG domain)
SD_TONE_GEN	O	1	Shut down tone generation	Level(CLK_REG domain)
ADC_SGLE_GO	O	1	ADC single go	Level(CLK_REG domain)
ADC_CNTI_GO	O	1	ADC continuous go	Level(CLK_REG domain)
MON_EN_REG	O	1	Monitor enable bit	Level(CLK_REG domain)
MON_WAKE_GO	O	1	Monitor wake go	Level(CLK_REG domain)
ADR_MANU	O	7	Manual address setting, for MTP_TOP	Level(CLK_REG domain)
CLEN	O	1	For MTP interface	Level(CLK_REG domain)
CS_MANU	O	1	Manual CS setting, for MTP_TOP	Level(CLK_REG domain)
MANU	O	1	MANU mode for ICELL test	Level(CLK_REG domain)
SRL	O	1	For MTP_TOP	Level(CLK_REG domain)
MRGN	O	1	For MTP_TOP	Level(CLK_REG domain)
RD_MANU	O	1	Manual RD setting, for MTP_TOP	Level(CLK_REG domain)
WR_MANU	O	1	Manual WR setting, for MTP_TOP	Level(CLK_REG domain)
HVEN	O	1	For MTP_TOP	Level(CLK_REG domain)
wr_nvm	O	1	Write MTP	Level(CLK_REG domain)
DIN	O	1	DATA from MTP	Level(CLK_REG domain)
GainErr_code_comm	O	12	TRIM_ADC	Level(CLK_REG domain)
ADC_VCM_GAIN_EN	O	1	TRIM_ADC	Level(CLK_REG domain)
ADC_TEMP_COM_EN	O	1	TRIM_ADC	Level(CLK_REG domain)

trim_logic_en	O	1	For TRIM_LOGIC	Level(CLK_REG domain)
GainErr_code_Cell1~18	O	7*18	TRIM_ADC	Level(CLK_REG domain)
GainErr_code_GPIO	O	12	TRIM_ADC	Level(CLK_REG domain)
GainErr_code_VPTAT	O	12	TRIM_ADC	Level(CLK_REG domain)
ADC_TEMP_code_H	O	8	TRIM_ADC	Level(CLK_REG domain)
ADC_TEMP_code_L	O	8	TRIM_ADC	Level(CLK_REG domain)
Vos_code_comm	O	8	TRIM_ADC	Level(CLK_REG domain)
Vos_code_cell1~18	O	6*18	TRIM_ADC	Level(CLK_REG domain)
Vos_code_GPIO	O	8	TRIM_ADC	Level(CLK_REG domain)
ALG_GAINERR1	O	8	TRIM_ADC	Level(CLK_REG domain)
Vcm_code_comm	O	7	TRIM_ADC	Level(CLK_REG domain)
GainErr_code_B_comm	O	12	TRIM_ADC	Level(CLK_REG domain)
AUX_VCM_GAIN_EN	O	1	TRIM_ADC	Level(CLK_REG domain)
AUX_TEMP_COM_EN	O	1	TRIM_ADC	Level(CLK_REG domain)
GainErr_code_B_Cell1~18	O	7*18	TRIM_ADC	Level(CLK_REG domain)
GainErr_code_B_GPIO	O	12	TRIM_ADC	Level(CLK_REG domain)
GainErr_code_B_VPTAT	O	12	TRIM_ADC	Level(CLK_REG domain)
AUX_TEMP_code_B_H	O	8	TRIM_ADC	Level(CLK_REG domain)
AUX_TEMP_code_B_L	O	8	TRIM_ADC	Level(CLK_REG domain)
Vos_code_B_comm	O	8	TRIM_ADC	Level(CLK_REG domain)
Vos_code_B_cell1~18	O	6*18	TRIM_ADC	Level(CLK_REG domain)
Vos_code_B_GPIO	O	8	TRIM_ADC	Level(CLK_REG domain)
ALG_GAINERR2	O	8	TRIM_ADC	Level(CLK_REG domain)
Vcm_code_B_comm	O	7	TRIM_ADC	Level(CLK_REG domain)
Reg1000~107F	O	8*128	For MTP_CRC	Level(CLK_REG domain)
CONF_CRC	O	16	For CONF_CRC check	Level(CLK_REG domain)
TM_REG1~10	O	8*10	Regs for test mode	Level(CLK_REG domain)
CELL_OV_THRESH_REG	O	7	For OVUV_OTUT_CMP	Level(CLK_REG domain)
CELL_UV_THRESH_REG	O	7	For OVUV_OTUT_CMP	Level(CLK_REG domain)
GPIO_OTUT_THRESH_SEL_REG	O	12	For OVUV_OTUT_CMP	Level(CLK_REG domain)
GPIO_OT_PACK_THRESH_REG	O	5	For OVUV_OTUT_CMP	Level(CLK_REG domain)
GPIO_OT_PCB_THRESH_REG	O	5	For OVUV_OTUT_CMP	Level(CLK_REG domain)
GPIO_UT_PACK_THRESH_REG	O	3	For OVUV_OTUT_CMP	Level(CLK_REG domain)
GPIO_UT_PCB_THRESH_REG	O	3	For OVUV_OTUT_CMP	Level(CLK_REG domain)
CB_OT_PACK_THRESH_REG	O	5	For OVUV_OTUT_CMP	Level(CLK_REG domain)
CB_OT_PCB_THRESH_REG	O	5	For OVUV_OTUT_CMP	Level(CLK_REG domain)
TWARN_THRESH_REG	O	3	For OVUV_OTUT_CMP	Level(CLK_REG domain)
GPIO_PUPD	O	12	For analog	Level(CLK_REG domain)
GPIO_PUPD_EN	O	12	For analog	Level(CLK_REG domain)
GPIO_AS_IN_EN	O	12	For analog	Level(CLK_REG domain)
WEAK_PUPD_EN	O	12	For analog	Level(CLK_REG domain)
WEAK_PUPD	O	12	For analog	Level(CLK_REG domain)

CS_EN_MANU	O	19	For analog	Level(CLK_REG domain)
OVUV_OTUT_EN_REG	O	1	For OVUV_OTUT_CMP	Level(CLK_REG domain)
OVUV_DEGL_REG	O	5	For OVUV_OTUT_CMP	Level(CLK_REG domain)
MON_WAKE_PERIOD_REG	O	6	For CYC_WAKE	Level(CLK_REG domain)
CELL_GAP_THRESH_REG	O	5	For GAP_CMP	Level(CLK_REG domain)
OTH_GAP_THRESH_REG	O	3	For GAP_CMP	Level(CLK_REG domain)
CELL_CAP_DEGL_REG	O	5	For GAP_CMP	Level(CLK_REG domain)
CH_TOP_STL_REG	O	4	ADC_SETTING	Level(CLK_REG domain)
CH_STL_REG	O	4	ADC_SETTING	Level(CLK_REG domain)
CH_BOT_STL_REG	O	4	ADC_SETTING	Level(CLK_REG domain)
ADC_MODE_REG	O	2	ADC_SETTING	Level(CLK_REG domain)
ADC_CLK_SET_REG	O	2	ADC_SETTING	Level(CLK_REG domain)
ADC_CHP_EN_REG	O	1	ADC_SETTING	Level(CLK_REG domain)
CH_DT_REG	O	2	ADC_SETTING	Level(CLK_REG domain)
STACK_RESPONSE	O	6	Determine byte interval time in response frames	Level(CLK_REG domain)
SOFT_RSTB_REG	O	1	0:Soft reset 1:not reset	Level(CLK_REG domain)
ADD_W_EN	O	1	Indicating address identify frame	Level(CLK_REG domain)
DEV_NUM	O	7	Device number from current device to the top device	Level(CLK_REG domain)
I2C_MAC_EN	O	1	I2c master enable	Level(CLK_REG domain)
i2c_sr	O	1	For I2C_MAS to generate a restart condition	Level(CLK_REG domain)
i2c_stop	O	1	For I2C_MAS to generate a stop condition	Level(CLK_REG domain)
i2c_tx	O	1	For I2C_MAS to send data	Level(CLK_REG domain)
i2c_tx	O	1	For I2C_MAS to receive data	Level(CLK_REG domain)
i2c_tx_data	O	8	For I2C_MAS prepare data to send	Level(CLK_REG domain)
C_OW_DET_GO	O	1	For C_OW_CTRL	Cleared by clr_C_OW_DET_GO
C_OW_TDIS_REG	O	4	For C_OW_CTRL	Level(CLK_REG domain)
BIST_GO	O	1	For CMP_BIST_CTRL	Cleared by clr_BIST_GO
CB_GO	O	1	To start CB_CTRL and load CB_Setting_REG	Cleared by clr_CB_GO
CB_MANU_PAUSE	O	1	pause CB_CTRL	Level(CLK_REG domain)
CB_SETTLE	O	5	For RECLK_COMP: CB settle time	Level(CLK_REG domain)
CB_GO	O	1	To start CB_CTRL and load CB_Setting_REG	Cleared by clr_CB_GO
CB_PERIOD_REG	O	3	in automatic mode, indicate odd/even covert time, 5s-30min, 8steps refer to competitor spec	Level(CLK_REG domain)
CB_DUTY_REG	O	3	duty of internal PWM, shall cover 12.5%-100%, 8steps period is 200ms	Level(CLK_REG domain)
CB_MANUAL_REG	O	1	CB mode select, 1: manual , 0:automatic	Level(CLK_REG domain)
JOT_EN_REG	O	1	enable JOT to pause CB_CTRL	Level(CLK_REG domain)
ADC_PAUSE_EN_REG	O	1	enable ADC_EN to pause CB_CTRL	Level(CLK_REG domain)
GPIO_CBOT_EN_REG	O	1	enable GPIO_CBOT to pause CB_CTRL	Level(CLK_REG domain)

FLT_STOP_EN_REG	O	N/A	enable FLT_WAKE to stop CB_CTRL, stop CB_EN, and wait for another CB_GO when FLT_WAKE is "L"	Level(CLK_REG domain)
CBFET_EN_REG	O	18	CB_EN Input of 18 channels	Level(CLK_REG domain)
CB_TWARN_THRESH_REG	O	4	after CB_GO,CB_CTRL output CB_TWARN_THRESH to analog, don't need other operation	Level(CLK_REG domain)
CB_UNIT_REG	O	18	unit of CB_TO_THRESH_REG	Level(CLK_REG domain)
CB_TO_THRESH_REG1-18	O	10*18	CB threshold time about each channel	Level(CLK_REG domain)
X_FLT_SUM_REG	O	1*5	Fault sum regs for FLT_LOGIC	Level(CLK_REG domain)
X_FLT_REG	O	1*143	Fault bit regs for FLT_LOGIC	Level(CLK_REG domain)
TM_KEY	O	8	For TM_KEY_CHECK	Level(CLK_REG domain)
CLK_REG_SC	I	1	Scan-mux result of 8MHz clock from CLK_32M	8MHz
resetb_CLK	I	1	Asynchronous reset signal(synchronously released)	
CLK_8M_256K_SC	I	1	When CLK_32M is OK, use CLK_8M; otherwise use CLK_256K; after scanmux	8MHz or 256kHz
resetb_CLK_OUT	I	1	Asynchronous reset signal by CLK_OUT(synchronously released)	
CLK_I2C_SC	I	1	For I2C_CTRL(change CLK_REG to CLK_I2C domain)	
resetb_SR_CLK_I2C	I	1	Asynchronous reset signal (synchronously released) with soft reset for I2C_CTRL	
SOFT_RSTB_32M	I	1	Soft reset from RSTGEN	Level(CLK_REG domain)
SOFT_RSTB	I	1	CLK_SLOW synced SOFT_RSTB_REG from u_SOFT_RSTB_sync	Level(CLK_SLOW domain)
pulse_2ms	I	1		
ini_addr	I	16	Initial address for writing	Level(CLK_REG domain)
reg_addr	I	16	Register address for reading	Level(CLK_REG domain)
neg_rx_en	I	1		
bytes	I	4	Bytes number for successive writing	Level(CLK_REG domain)
wr_data	I	128	Received data buffer	Level(CLK_REG domain)
wr_update	I	1	Write update time	Level(CLK_REG domain)
SPI_EN	I	1	1:used as bridge 0:used as AFE	Async(constant)
rx_en_n	I	1	Receive data from N port	Level(CLK_32M domain)
rx_en_s	I	1	Receive data from S port	Level(CLK_32M domain)
dev_addr_dlv	I	1	Device address identify delivery	Level(CLK_REG domain)
dev_addr0	I	8	Device address	Level(CLK_REG domain)
state_tx_bps	I	1	tx_state is STATE_BYPASS	Level(CLK_REG domain)
state_rx_bps	I	1	state is STATE_BYPASS	Level(CLK_REG domain)
state_rx_init	I	1	state is STATE_INIT	Level(CLK_REG domain)
state_rx_cur_addr	I	1	state is STATE_CUR_ADR	Level(CLK_REG domain)
FRAME_DONE	O	1	A complete frame is received.	Level(CLK_REG domain)
FR_CRC_FLT	O	1	Frame CRC fault	Level(CLK_REG domain)
TWARN	O	1	Fault input from u_OVUV_OTUT_CMP	Level(CLK_SLOW domain)

load_done	O	1	MTP load done	Level(CLK_REG domain)
DEV_ADD_LATCH	O	7	Latched device address	constant
DEV_NUM_LATCH	O	7	Latched device number in the rest of the daisy chain	constant
DIR_SEL_LATCH	O	1	Latched direction seletion 1:data from N to S 0:data from S to N	constant
TOP_DEV_LATCH	O	1	Latched top device information	constant
clr_ADC_GO	O	1	Signal to clear 4 kinds of ADC_GO	
clr_MON_WAKE_GO	O	1	Signal to clear MON_WAKE_GO	1 CLK_SLOW
clr_C_OW_DET_GO	O	1	Signal to clear C_OW_DET_GO	1 CLK_SLOW
RR_COUNTER	I	16	Round-robin number, RR_COUNTER is frozen when FREEZE_DLY is detected, is cleared by ADC_GO_DLY is high .	
pos_HBFAST	I	1	HB too fast condition is detected	1 CLK_256K
pos_HBTO	I	1	HB too slow(timeout) condition is detected	1 CLK_256K
clr_TO_SLEEP	I	1	Signal to clear TO_SLEEP	2~3 CLK_SLOW
clr_WAKE_TONE_GEN	I	1	Pulse for module u_COMM_REG to clear WAKE_TONE_GEN	1 CLK_256K
clr_STA_TONE_GEN	I	1	Pulse for module u_COMM_REG to clear STA_TONE_GEN	1 CLK_256K
clr_SD_TONE_GEN	I	1	Pulse for module u_COMM_REG to clear SD_TONE_GEN	1 CLK_256K
MISS	I	1		
ORDER	I	1		
SYNCT	I	1		
SYNCD	I	1		
BIT	I	1		
RR	I	1		
SOFB	I	1		
IERR	I	1		
TXDIS	I	1		
SOF	I	1		
UNEXP_C	I	1		
UNEXP_R	I	1		
CRC	I	1		
CONFL	I	1		
pos_TBYTE_FAST	I	1	fault flag: receiving data is too fast	4 CLK_32M
pos_TBYTE_TO	I	1	fault flag: receiving data is too slow	4 CLK_32M
FCOMM_FLT_IN	I	1	Communication fault received from last device	Level(CLK_REG domain)
i2c_rx_data	I	8	I2c received data	Level(CLK_I2C domain)
i2c_ack_out	I	1	I2c acknowledge bit	Level(CLK_REG domain)
DOUT	I	8	Data loaded from u_MTP	async
READ	I	1	READ pulse from u_MTP_TOP	Level(CLK_MTP domain)
idle_2h	I	1	Mark the time for shadow registers to update with DOUT	8 CLK_MTP
clrb_wr_mtp	I	1	Signal to clear WR_MTP	Level(CLK_MTP domain)

ADR	I	7	Address for MTP_REG	Level(CLK_MTP domain)
WR_MTP	I	1	MTP writing is on-going	
LCTO_SD_LATCH	I	1	Analog latched LCTO_SD fault	Async(constant)
CLK_256K_OKB_LATCH	I	1	Analog latched CLK_256K not OK fault	Async(constant)
VDD_OKB_LATCH	I	1	Analog latched VDD not OK fault	Async(constant)
VAA_OKB_LATCH	I	1	Analog latched VAA not OK fault	Async(constant)
VDD_OV_LATCH	I	1	Analog latched VDD OV fault	Async(constant)
VDD_UV_LATCH	I	1	Analog latched VDD UV fault	Async(constant)
GPIO_HL	I	12	From analog, read-only	Async
CELL_UV	I	18	Cell UV fault	Level(CLK_SLOW domain)
CELL_OV	I	18	Cell OV fault	Level(CLK_SLOW domain)
GPIO_UT	I	12	GPIO UT fault	Level(CLK_SLOW domain)
GPIO_OT	I	12	GPIO OT fault	Level(CLK_SLOW domain)
VAA_OV	I	1	VAA OV fault from analog	Async
VAA_UV	I	1	VAA UV fault from analog	Async
VDD_OV	I	1	VDD OV fault from analog	Async
VDD_UV	I	1	VDD UV fault from analog	Async
CP_OV	I	1	CP OV fault from analog	Async
CP_UV	I	1	CP UV fault from analog	Async
AGND_OW	I	1	AGND OW fault from analog	Async
SET_SCTO	I	1	Short timeout	Level(CLK_REG domain)
SET_LCTO	I	1	Long timeout	Level(CLK_REG domain)
TRIM_EN	I	1	Trim enable	Level(CLK_REG domain)
TMREG_EN	I	1	Test mode reg enable	Level(CLK_REG domain)
tx_add_reg_addr	I	1	tx register address adds bytes end pulse	1 CLK_32M
tx_state_addr	I	1	tx_state is STATE_ADDR	Level(CLK_REG domain)
ADC_CH1~18	I	16*18	Trimmed ADC_CH data	Level(CLK_REG domain)
ADC_GPIO1~12	I	16*12	Trimmed ADC_GPIO data	Level(CLK_REG domain)
ADC_VPTAT	I	1	Trimmed ADC_VPTAT data	Level(CLK_REG domain)
ADC_VBG	I	1	Trimmed ADC_VBG data	Level(CLK_REG domain)
ADC_VBG2	I	1	Trimmed ADC_VBG2 data	Level(CLK_REG domain)
ADC_LPF_CH1~18	I	16*18	Trimmed ADC_CH data after LPF	Level(CLK_REG domain)
AUX_CH1~18	I	16*18	Trimmed AUX_CH data	Level(CLK_REG domain)
AUX_GPIO1~12	I	16*12	Trimmed AUX_GPIO data	Level(CLK_REG domain)
AUX_VPTAT	I	1	Trimmed AUX_VPTAT data	Level(CLK_REG domain)
AUX_VBG	I	1	Trimmed AUX_VBG data	Level(CLK_REG domain)
AUX_VBG2	I	1	Trimmed AUX_VBG2 data	Level(CLK_REG domain)
AUX_LPF_CH1~18	I	16*18	Trimmed AUX_CH data after LPF	Level(CLK_REG domain)
CB_ODD_CNT	I	16	odd/even is same in manual mode, is different in automatic mode, counter	Level(CLK_CB domain)

			of odd group	
CB_EVEN_CNT	I	16	counter of even group	Level(CLK_CB domain)
CB_EN_FULL_DUTY	I	18	output CB_CH_EN with full duty	Level(CLK_CB domain)
FR_CNT	I	16	Frame counter	Level(CLK_REG domain)
CMP_FLT	I	16	Compare fault from u_CMP_BIST_CTRL	Level(CLK_REG domain)
RX_FIFO_OF	I	1	TX FIFO overflow, when TX FIFO is full and daisy chain still write to TX FIFO	1 CLK_REG_SC
TX_FIFO_OF	I	1	RXFIFO overflow, when RXFIFO is full and host still write to RXFIFO	1 CLK_REG_SC
TX_FIFO_UF	I	1	TXFIFO underflow, when FIFO is empty and read from FIFO	1 CLK_REG_SC
CONF_REG_CRC_FLT	I	1	Configure registers CRC fault	Level(CLK_SLOW domain)
MTP_REG_CRC_FLT	I	1	MTP registers CRC fault	Level(CLK_SLOW domain)
FLT_TONE_DET	I	1	Fault tone detected by analog	async
clr_BIST_GO	I	1	Signal to reset BIST_GO	
clr_CB_GO	I	1	output to clear CB_GO	>1 CLK_CB_SC
CELL_GAP_FLT	I	18	Over gap threshold flag of CELL1-CELL18	Level(CLK_SLOW domain)
OTH_GAP_FLT	I	12	Over gap threshold flag of other channel	Level(CLK_SLOW domain)
C_OW_FLT	I	19	Over range flags	Level(CLK_REG domain)
ECED	I	1	MTP data corrected flag by ECC algorithm	aysnc
CB_CONF_FLT	I	1	>2 consecutive channels turn on in CBFET_EN	1 CLK_CB_SC
BUSY	I	1	MTP write on-going flag by u_MTP	async
neg_response	I	1	Negative edge of response	1 CLK_REG
SCAN_CLK	I	1	CLK in scan mode	Up to 10MHz
SCAN_MODE	I	1	SCAN_MODE	Level(CLK_REG domain)
SCAN_RSTB	I	1	Async resetb in scan mode	Level(CLK_32M domain)
Scan_enable	I	1	1:shift mode 0:capture mode or function mode	4 CLK_32M

Clock Domain

The clock for COMM_REG is CLK_REG_SC and CLK_8M_256K_SC.
For u_FLT_BIT and u_FLT_SUM, both CLK_REG_SC and CLK_8M_256K_SC are used.
For other sub-modules, only CLK_REG_SC is used.

COMM_REG function description

1 Control bits

TO_SD bit is defined in reg2002[1]. TO_SD is synchronized in module u_TO_SD_sync by CLK_SLOW, and module u_TO_SD_sync output D2A_TO_SD for analog. Analog shutdown digital part, thus TO_SD and D2A_TO_SD are both pulses.(HWR001_COMM_REG)

TO_SLEEP bit is defined in reg2002[0]. TO_SLEEP is synchronized in module u_TO_SLEEP_sync by CLK_SLOW, and module u_TO_SLEEP_sync output D2A_TO_SLEEP for analog, and then generate clr_TO_SLEEP for u_COMM_REG to clear TO_SLEEP, thus TO_SLEEP is a pulse.(HWR002_COMM_REG)

PROG_SCTO[2:0] is defined in reg0004[2:0]. (HWR003_COMM_REG)

PROG_LCTO[2:0] is defined in reg0004[5:3]. (HWR004_COMM_REG)

LCTO_SEL[1:0] is defined in reg0004[7:6]. (HWR005_COMM_REG)

ADC_SGLE_GO is defined in reg2004[1], ADC_CNTI_GO is defined in reg2004[0]. Both ADC_SGLE_GO and ADC_CNTI_GO are cleared by clr_ADC_GO. (HWR011_COMM_REG)

MON_WAKE_GO is defined in reg2004[7]. MON_WAKE_GO is cleared by clr_MON_WAKE_GO. (HWR012_COMM_REG)

MON_EN_REG is defined in reg000D[0]. (HWR050_COMM_REG)

FREEZE is defined in reg2004[2]. (HWR013_COMM_REG)

DLPF_FC[2:0] is defined in reg0008[7:5]. (HWR015_COMM_REG)

STACK_RESPONSE[5:0] is defined in reg0003[5:0]. (HWR019_COMM_REG)

COMN_TX_DIS is defined in reg0003[7], COMN_TX_DIS is defined in reg0003[6]. (HWR054_COMM_REG)

SD_TONE_GEN is defined in reg2002[2], use SPI_DIR as SD_TONE_DIR. SPI_DIR is defined in reg2003[0]. (HWR051_COMM_REG) SD_TONE_GEN is cleared by clr_SD_TONE_GEN. (HWR020_COMM_REG)

WAKE_TONE_GEN is defined in reg2002[4], WAKE_TONE_GEN reset to 0 when clr_WAKE_TONE_GEN is high. (HWR052_COMM_REG)

STA_TONE_GEN is defined in reg2002[3], STA_TONE_GEN reset to 0 when clr_STA_TONE_GEN is high. (HWR053_COMM_REG)

All fault bits have corresponding mask bits. Mask bits are defined in reg0100~0117. (HWR036_COMM_REG)

MTP_CTRL_BITS are defined in reg1FFB~1FFF. They are writable only when TRIM_EN high. (HWR043_COMM_REG)

I2C_CTRL bits include i2c_stop, i2c_rx, i2c_sr, i2c_tx. These 4 bits are defined in reg2200. I2C_MAS_EN is defined in reg0002[0]. I2C_CTRL include i2c_tx_data[7:0], which is defined in reg2201. I2c_rd_data[7:0] is readable from address 0x2202. I2c_ack_out is readable from bit 4 of address 2200. (HWR047_COMM_REG)

SOFT_RSTB_REG is defined in reg2002[7]. SOFT_RSTB_REG is default high. When written 0 via COMM_CTRL, SOFT_RSTB_REG is low. SOFT_RSTB_REG reset to high when clr_SOFT_RSTB high. (HWR049_COMM_REG)

2 Trim bits

TRIM_ADC bits are defined in reg1000~reg105F. (HWR008_COMM_REG)

TRIM_ANA bits are defined in reg1060~reg107A. (HWR042_COMM_REG)

X-Y and lot information are readable from address 0x107C~107D. These bits are writable only when TRIM_EN high. (HWR048_COMM_REG)

3 Setting bits

ADC_SETTING_REG bits are defined in reg0008~reg000C. (HWR010_COMM_REG)

MON_WAKE_PERIOD_REG[5:0] is defined in reg000D[7:2]. (HWR031_COMM_REG)

CELL_OV_THRESH_REG[6:0] is defined in reg000F[6:0], CELL_UV_THRESH_REG[6:0] is defined in reg0010[6:0]. (HWR041_COMM_REG)

GPIO_OT_PACK_THRESH_REG[4:0] is defined in reg0011[7:3], GPIO_UT_PACK_THRESH_REG[2:0] is defined in reg0011[2:0], GPIO_OT_PCB_THRESH_REG[4:0] is defined in reg0012[7:3], GPIO_UT_PCB_THRESH_REG[2:0] is defined in reg0012[2:0]. (HWR044_COMM_REG)

GPIO_PUPD[11:0] is defined in reg202D~202E, GPIO_PUPD_EN[11:0] is defined in reg0019~001A, GPIO_AS_IN_EN[11:0] is defined in reg001B~001C. (HWR045_COMM_REG)

GPIO_HL[11:0] is readable from address 0x5002~5003. (HWR046_COMM_REG)

OVUV_DEGL_REG[4:0] is defined in reg000E[4:0]. (HWR056_COMM_REG)

OVUV_OTUT_EN_REG is defined in reg000E[7]. (HWR057_COMM_REG)

4 ADC results

ADC_DATA_LPF are readable from address 0x4000~0x4023. (HWR009_COMM_REG)

CELL_ADC_DATA are readable from address 0x4048~0x406B, OTH_ADC_DATA are readable from address 0x4090~0x40A7, 0x40C0~0x40C5. (HWR014_COMM_REG)

RR_COUNTER is readable from address 0x3FFE~0x3FFF. (HWR016_COMM_REG, HWR018_COMM_REG)

5 COMM_DIG_SETTING

TOP_DEV is defined in reg0000[7], DIR_SEL is defined in reg2002[6], DEV_ADD[6:0] is defined in reg0000[6:0], DEV_NUM[6:0] is defined in reg0001[6:0]. (HWR021_COMM_REG, HWR024_COMM_REG)

Default values of COMM_DIG_SETTING registers are from XXX_LATCH. DEV_ADD cannot be changed when ADD_W_EN is low. (HWR023_COMM_REG)

ADD_W_EN is defined in reg2003[1]. ADD_W_EN is high when address identify starts. ADD_W_EN is cleared by writing reg2003[1] to 0.(HWR021_COMM_REG, HWR022_COMM_REG)

DEV_NUM[6:0] indicates the device number from current device to the top device. DEV_NUM[6:0] adds by step 1 when receiving device response when ADD_W_EN high(in address identify response stage). DEV_NUM[6:0] reset to 0 when address identify starts. (HWR024_COMM_REG)

6 CB_CTRL_SETTING_REG

CBFET_EN_REG1~18 are defined in reg2006~2008. CB_GO is defined in reg2005[0]. CB_GO is high when written high, is low when clr_CB_GO high. CB_MANUAL_REG is defined in reg2005[2]. (HWR025_COMM_REG)

CB_TO_THRESH_REG1~18[9:0] and CB_UNIT_REG[17:0] are defined in reg2009~202C. (HWR026_COMM_REG)

CB_TWARN_THRESH_REG[3:0] is defined in reg0017[7:4]. (HWR027_COMM_REG)

JOT_EN_REG is defined in reg0017[1]. (HWR028_COMM_REG)

GPIO_CBOT_EN_REG is defined in reg0017[3]. (HWR029_COMM_REG)

CB_OT_PACK_THRSH_REG[4:0] is defined in reg0015[7:3], CB_OT_PCB_THRSH_REG[4:0] is defined in reg0016[7:3]. GPIO_OTUT_THRESH_SEL_REG[11:0] is defined in reg0013~0014. (HWR030_COMM_REG)

CB_MANU_PAUSE is defined in reg2005[1]. (HWR033_COMM_REG)

ADC_PAUSE_EN_REG is defined in reg0017[2]. (HWR034_COMM_REG)

FLT_STOP_EN_REG is defined in reg0017[0]. (HWR035_COMM_REG)

CB_PERIOD_REG[2:0] is defined in reg0016[2:0]. (HWR037_COMM_REG)

CB_DUTY_REG[2:0] is defined in reg0015[2:0]. (HWR038_COMM_REG)

CB_ODD_CNT[15:0] and CB_EVEN_CNT[15:0] are readable from address 0x5007~500A. (HWR039_COMM_REG)

CB_CH_EN_FULL_DUTY[17:0] are readable from address 0x5004~5006. (HWR040_COMM_REG)

CB_SETTLE[4:0] is defined in reg0002[7:3]. (HWR055_COMM_REG)

7 Fault regs

Fault regs include FLT_BIT registers and FLT_SUM registers. FLT_BIT registers are located in addresses 0x5101~5117, FLT_SUM registers are located in address 0x5100.

SCTO flag bit is defined in reg5116[6], it is set by SET_SCTO. (HWR001_FLT_REG)

LCTO flag bit is defined in reg5116[5], when it is set by SET_LCTO or by LCTO_LATCH. (HWR002_FLT_REG)

LCTO_SD, VAA_OKB, VDD_OKB, CLK_256K_OKB, VDD_OV, VDD_UV flag bits are defined in address 0x5115. They support being only set by corresponding input XXX_LATCH. FLT_BIT output RST_XXX_LATCH when register bits is written to 1. (HWR003_FLT_REG)

FLT_TONE_DET_REG is recorded in reg5117[0]. FLT_TONE_DET_REG shall be high when FLT_TONE_DET is high. (HWR004_FLT_REG)

GPIO_OTUT (including GPIO_OT[11:0] and GPIO_UT[11:0]) are recorded in FLT_REG. OT_FLT[11:0] recording GPIO_OT faults, are in address 0x510F~5110, UT_FLT[11:0] recording GPIO_UT fault, are in address 0x5111~5112. (HWR005_FLT_REG)

CELL_OVUV (including CELL_OV[17:0] and CELL_UV[17:0]) are recorded in FLT_REG. OV_FLT[17:0] recording CELL_OV faults, are in address 0x5104~5106, UV_FLT[17:0] recording CELL_UV faults, are in address 0x5107~5109. (HWR006_FLT_REG)

Every fault bit has its corresponding mask bit. All fault bits can not be set high when corresponding mask bit(from reg0100~0117) is 1. (HWR009_FLT_REG)

COW[18:0] are fault flag from u_C_OW_CTRL, are in address 0x5101~5103. (HWR010_FLT_REG)

All fault bits can not be written to 1 via COMM_CTRL. When corresponding mask bit is 0, all fault bits canl be cleared to 0 when written 0 via COMM_CTRL. (HWR011_FLT_REG)

CB_CONF_FLT is recorded in reg5116[7]. (HWR012_FLT_REG)

RX_FIFO_OF_REG bit set by RX_FIFO_OF is recorded in reg5116[3], TX_FIFO_OF_REG bit set by TX_FIFO_OF is recorded in reg5116[2]. (HWR013_FLT_REG)

TX_FIFO_UF_REG bit set by TX_FIFO_UF is recorded in reg5116[1]. (HWR014_FLT_REG)