Table 1−1. OMAP2423 Pin Characteristics

Pin					Pinout Control Register			Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
C5	O	Dedicated	V_{DDS1}	jtag_emu0	0x013C	3	jtag_rtck	Mode0	000	No	–	00
A11	IO	Dedicated	V_{DDS1}	sdrc_d12	0x0064	0	gpmc_a23	Mode0	000	PUPD	D	01
C22	I	Dedicated	V_{DDS1}	jtag_tms	0x0140	1	jtag_tdi	Mode0	000	PUPD	D	01
C24	O	Dedicated	V_{DDS1}	jtag_tms	0x0140	2	jtag_tdo	Mode0	000	No	–	00
W12	IO	Muxed	V_{DDS}	ssi1_wake	0x00E4	1	vlynq_clk	Protect	111	PUPD	U	11
Y15	IO	Muxed	V_{DDS}	vlynq_tx1	0x00E8	1	vlynq_tx0	Protect	111	PUPD	D	01
AC23	I	Dedicated	V_{DDA}	tv_rref	0x011C	0	tv_rref	Mode0	000	No	–	00
AD23	O	Dedicated	V_{DDA}	uart3_tx_irtx	0x0118	2	tv_cvbs	Mode0	000	No	–	00
AD20	IO	Muxed	V_{DDS}	sys_xtalout	0x0134	2	sys_clkreq	Protect	111	PUPD	U	11
AB26	I	Dedicated	V_{DDA}	uart3_tx_irtx	0x0118	3	tv_vref	Mode0	000	No	–	00
AB25	I	Dedicated	V_{DDS2}	jtag_emu0	0x013C	1	jtag_ntrst	Mode0	000	PUPD	D	01
AC22	IO	Muxed	V_{DDS2}	gpio_6	0x0138	3	jtag_emu1	Protect	111	PUPD	U	11
AD7	IO	Muxed	V_{DDS}	ssi1_wake	0x00E4	3	vlynq_rx0	Protect	111	PUPD	D	01
W10	IO	Muxed	V_{DDS}	ssi1_wake	0x00E4	2	vlynq_rx1	Protect	111	PUPD	D	01
AE10	IO	Muxed	V_{DDS}	vlynq_tx1	0x00E8	2	vlynq_nla	Protect	111	PUPD	U	11
E3	I	Dedicated	V_{DDS1}	jtag_emu0	0x013C	2	jtag_tck	Mode0	000	PUPD	D	01
M26	O	Dedicated	V_{DDS1}	sdrc_ba0	0x0034	3	sdrc_a9	Mode0	000	No	–	00
AC26	O	Dedicated	V_{DDS1}	sdrc_nras	0x00A4	0	sdrc_nras	Mode0	000	No	–	00
AE25	IO	Muxed	V_{DDS1}	sdrc_ncs0	0x00A0	3	sdrc_cke1	Protect	111	PUPD	U	11
B16	O	Dedicated	V_{DDS1}	gpmc_wait2	0x009C	3	sdrc_nclk	Mode0	000	No	–	00
A24	IO	Dedicated	V_{DDS1}	sdrc_dqs1	0x00B0	2	sdrc_dqs3	Mode0	000	PUPD	D	01

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin					Pinout Control Register			Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
D1	IO	Dedicated	V_{DDS1}	sdrc_d4	0x006C	1	gpmc_a14	Mode0	000	PUPD	D	01
D22	I	Dedicated	V_{DDS1}	jtag_tms	0x0140	0	jtag_tms	Mode0	000	PUPD	D	01
AA26	IO	Muxed	V_{DDS1}	sdrc_a14	0x0030	0	gpio_0	Protect	111	PUPD	D	01
W25	IO	Muxed	V_{DDS1}	sdrc_a14	0x0030	1	sdrc_a13	Protect	111	PUPD	D	01
P25	O	Dedicated	V_{DDS1}	sdrc_a4	0x003C	0	sdrc_a4	Mode0	000	No	−	00
L25	O	Dedicated	V_{DDS1}	sdrc_a4	0x003C	2	sdrc_a2	Mode0	000	No	−	00
H25	O	Dedicated	V_{DDS1}	sdrc_a0	0x0040	0	sdrc_a0	Mode0	000	No	−	00
T26	O	Dedicated	V_{DDS1}	sdrc_ba0	0x0034	1	sdrc_a11	Mode0	000	No	−	00
AF23	O	Dedicated	V_{DDS1}	sdrc_ba0	0x0034	0	sdrc_ba0	Mode0	000	No	−	00
Y25	IO	Muxed	V_{DDS1}	sdrc_ncs0	0x00A0	1	sdrc_ncs1	Protect	111	PUPD	U	11
AE23	O	Dedicated	V_{DDS1}	sdrc_nras	0x00A4	2	sdrc_nwe	Mode0	000	No	−	00
A15	IO	Dedicated	V_{DDS1}	gpmc_wait2	0x009C	2	sdrc_clk	Mode0	000	PUPD	−	00
B21	IO	Dedicated	V_{DDS1}	sdrc_d24	0x0058	1	sdrc_d23	Mode0	000	PUPD	D	01
D26	IO	Dedicated	V_{DDS1}	sdrc_stk_d16	0x0050	3	sdrc_d29	Mode0	000	PUPD	D	01
F1	IO	Dedicated	V_{DDS1}	sdrc_d0	0x0070	0	gpmc_a11	Mode0	000	PUPD	D	01
D2	IO	Dedicated	V_{DDS1}	sdrc_d8	0x0068	3	gpmc_a16	Mode0	000	PUPD	D	01
A3	IO	Dedicated	V_{DDS1}	sdrc_d4	0x006C	0	gpmc_a15	Mode0	000	PUPD	D	01
B3	IO	Dedicated	V_{DDS1}	sdrc_d8	0x0068	2	gpmc_a17	Mode0	000	PUPD	D	01
A7	IO	Dedicated	V_{DDS1}	sdrc_d16	0x0060	1	gpmc_a26	Mode0	000	PUPD	D	01
N25	O	Dedicated	V_{DDS1}	sdrc_a8	0x0038	3	sdrc_a5	Mode0	000	No	−	00
R25	O	Dedicated	V_{DDS1}	sdrc_a8	0x0038	1	sdrc_a7	Mode0	000	No	−	00

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin					Pinout Control Register			Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
H26	O	Dedicated	V_{DDS1}	sdrc_ba0	0x0034	2	sdrc_a10	Mode0	000	No	–	00
AC25	O	Dedicated	V_{DDS1}	sdrc_a14	0x0030	3	sdrc_ba1	Mode0	000	No	–	00
W26	IO	Muxed	V_{DDS1}	sdrc_a14	0x0030	2	sdrc_a12	Protect	111	PUPD	D	01
Y26	O	Dedicated	V_{DDS1}	sdrc_ncs0	0x00A0	0	sdrc_ncs0	Mode0	000	No	–	00
AF24	O	Dedicated	V_{DDS1}	sdrc_ncs0	0x00A0	2	sdrc_cke0	Mode0	000	No	–	00
AA25	O	Dedicated	V_{DDS1}	sdrc_nras	0x00A4	1	sdrc_ncas	Mode0	000	No	–	00
B19	IO	Dedicated	V_{DDS1}	sdrc_dqs1	0x00B0	1	sdrc_dqs2	Mode0	000	PUPD	D	01
D24	IO	Muxed	V_{DDS2}	mmc_dat3	0x00F8	0	mmc_dat3	Protect	111	PUPD	D	01
G20	IO	Muxed	V_{DDS2}	dss_d17	0x00C4	1	uart1_cts	Protect	111	PUPD	U	11
AE18	IO	Muxed	V_{DDS1}	sdrc_d0	0x0070	2	gpmc_a9	SBoot3	***	PUPD	U	11
B2	IO	Dedicated	V_{DDS1}	sdrc_d4	0x006C	2	gpmc_a13	Mode0	000	PUPD	D	01
A6	IO	Dedicated	V_{DDS1}	sdrc_d16	0x0060	3	gpmc_a24	Mode0	000	PUPD	D	01
L26	O	Dedicated	V_{DDS1}	sdrc_a4	0x003C	1	sdrc_a3	Mode0	000	No	–	00
M25	O	Dedicated	V_{DDS1}	sdrc_a4	0x003C	3	sdrc_a1	Mode0	000	No	–	00
T25	O	Dedicated	V_{DDS1}	sdrc_a8	0x0038	0	sdrc_a8	Mode0	000	No	–	00
E23	IO	Muxed	V_{DDS2}	mmc_dat_dir3	0x00FC	0	mmc_dat_dir3	Protect	111	PUPD	D	01
H20	IO	Muxed	V_{DDS2}	mmc_cmd	0x00F4	3	mmc_dat2	Protect	111	PUPD	D	01
G4	IO	Muxed	V_{DDS1}	gpmc_ncs0	0x008C	2	gpmc_ncs2	Protect	111	PUPD	U	11
D23	IO	Muxed	V_{DDS2}	mmc_dat3	0x00F8	2	mmc_dat_dir1	Protect	111	PUPD	D	01
D5	IO	Muxed	V_{DDS1}	sdrc_d0	0x0070	1	gpmc_a10	SBoot3	***	PUPD	U	11
E4	IO	Muxed	V_{DDS1}	gpmc_a7	0x0074	0	gpmc_a7	SBoot3	***	PUPD	U	11

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin				Pinout Control Register				Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
D6	IO	Muxed	V_{DDS1}	gpio_6	0x0138	0	gpio_6	Protect	111	PUPD	D	01
H8	IO	Muxed	V_{DDS1}	gpmc_ncs4	0x0090	0	gpmc_ncs4	Protect	111	PUPD	U	11
G23	IO	Muxed	V_{DDS2}	mmc_dat3	0x00F8	3	mmc_dat_dir2	Protect	111	PUPD	D	01
F23	IO	Muxed	V_{DDS2}	mmc_dat3	0x00F8	1	mmc_dat_dir0	Protect	111	PUPD	D	01
H17	IO	Muxed	V_{DDS2}	mmc_cmd	0x00F4	1	mmc_dat0	Protect	111	PUPD	D	01
AF9	IO	Muxed	V_{DDS1}	gpmc_a7	0x0074	1	gpmc_a6	SBoot3	***	PUPD	U	11
AE16	IO	Muxed	V_{DDS1}	gpmc_a7	0x0074	2	gpmc_a5	SBoot3	***	PUPD	U	11
A10	IO	Dedicated	V_{DDS1}	sdrc_d12	0x0064	2	gpmc_a21	Mode0	000	PUPD	D	01
B12	IO	Dedicated	V_{DDS1}	sdrc_d8	0x0068	0	gpmc_a19	Mode0	000	PUPD	D	01
J24	IO	Muxed	V_{DDS2}	mmc_dat_dir3	0x00FC	1	mmc_cmd_dir	Protect	111	PUPD	D	01
H23	IO	Muxed	V_{DDS2}	eac_bt_fs	0x00F0	3	mmc_clko	Protect	111	PUPD	D	01
AE15	O	Muxed	V_{DDS1}	gpmc_nbe1	0x0098	1	gpmc_nwp	Mode0	000	No	−	00
W8	IO	Muxed	V_{DDS1}	gpmc_a7	0x0074	3	gpmc_a4	SBoot3	***	PUPD	U	11
J7	IO	Muxed	V_{DDS1}	sdrc_d0	0x0070	3	gpmc_a8	SBoot3	***	PUPD	U	11
R26	O	Dedicated	V_{DDS1}	sdrc_a8	0x0038	2	sdrc_a6	Mode0	000	No	−	00
K3	IO	Muxed	V_{DDS1}	gpmc_ncs4	0x0090	1	gpmc_ncs5	Protect	111	PUPD	U	11
B11	IO	Dedicated	V_{DDS1}	sdrc_d12	0x0064	1	gpmc_a22	Mode0	000	PUPD	D	01
B7	IO	Dedicated	V_{DDS1}	sdrc_d12	0x0064	3	gpmc_a20	Mode0	000	PUPD	D	01
G19	IO	Muxed	V_{DDS2}	mmc_cmd	0x00F4	2	mmc_dat1	Protect	111	PUPD	D	01
C23	IO	Muxed	V_{DDS2}	mmc_dat_dir3	0x00FC	2	mmc_clki	Protect	111	PUPD	D	01
J23	IO	Muxed	V_{DDS2}	mmc_cmd	0x00F4	0	mmc_cmd	Protect	111	PUPD	D	01

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin					Pinout Control Register			Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
K19	IO	Muxed	V_{DDS2}	i2c2_sda	0x0114	0	i2c2_sda	Protect	111	PU	U	11
N1	O	Dedicated	V_{DDS1}	gpmc_nadv_ale	0x0094	1	gpmc_noe	Mode0	000	No	–	00
K20	IO	Muxed	V_{DDS2}	dss_d17	0x00C4	2	uart1_rts	Protect	111	PUPD	U	11
AF10	IO	Muxed	V_{DDS1}	gpmc_a3	0x0078	0	gpmc_a3	SBoot3	***	PUPD	U	11
W9	IO	Muxed	V_{DDS1}	gpmc_a3	0x0078	1	gpmc_a2	SBoot3	***	PUPD	U	11
F2	IO	Dedicated	V_{DDS1}	sdrc_d4	0x006C	3	gpmc_a12	Mode0	000	PUPD	D	01
B4	IO	Dedicated	V_{DDS1}	sdrc_d8	0x0068	1	gpmc_a18	Mode0	000	PUPD	D	01
B8	IO	Dedicated	V_{DDS1}	sdrc_d16	0x0060	2	gpmc_a25	Mode0	000	PUPD	D	01
L24	IO	Muxed	V_{DDS2}	usb0_rcv	0x0120	2	usb0_se0	Mode0	000	PUPD	–	00
L19	IO	Muxed	V_{DDS2}	mcbsp1_clkx	0x0110	3	i2c2_scl	Protect	111	PU	U	11
K23	IO	Muxed	V_{DDS2}	usb0_rcv	0x0120	0	usb0_rcv	Mode0	000	PUPD	–	00
M23	IO	Muxed	V_{DDS2}	tv_rref	0x011C	2	usb0_vp	Mode0	000	PUPD	–	00
L23	IO	Muxed	V_{DDS2}	tv_rref	0x011C	1	usb0_puen	Mode0	000	PUPD	–	00
M8	IO	Muxed	V_{DDS1}	gpmc_a3	0x0078	2	gpmc_a1	SBoot3	***	PUPD	U	11
P1	IO	Muxed	V_{DDS1}	gpmc_d2	0x0088	3	gpmc_clk	Protect	111	PUPD	D	01
H1	IO	Muxed	V_{DDS1}	gpmc_a3	0x0078	3	gpmc_d15	SBoot3	***	PUPD	U	11
H2	IO	Muxed	V_{DDS1}	gpmc_d14	0x007C	0	gpmc_d14	SBoot3	***	PUPD	U	11
M7	IO	Muxed	V_{DDS1}	gpmc_ncs4	0x0090	2	gpmc_ncs6	Protect	111	PUPD	U	11
H24	IO	Muxed	V_{DDS2}	uart3_tx_irtx	0x0118	1	uart3_rx_irrx	Protect	111	PUPD	U	11
G24	IO	Muxed	V_{DDS2}	uart3_tx_irtx	0x0118	0	uart3_tx_irtx	Protect	111	PUPD	U	11
J25	IO	Muxed	V_{DDS2}	usb0_rcv	0x0120	3	usb0_dat	Mode0	000	PUPD	–	00

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin				Pinout Control Register				Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
M24	IO	Muxed	V_{DDS2}	usb0_rcv	0x0120	1	usb0_txen	Mode0	000	PUPD	−	00
N23	IO	Muxed	V_{DDS2}	tv_rref	0x011C	3	usb0_vm	Mode0	000	PUPD	−	00
N8	O	Dedicated	V_{DDS1}	gpmc_nadv_ale	0x0094	0	gpmc_nadv_ale	Mode0	000	No	−	00
M1	O	Dedicated	V_{DDS1}	gpmc_nadv_ale	0x0094	2	gpmc_nwe	Mode0	000	No	−	00
P2	IO	Muxed	V_{DDS1}	gpmc_d14	0x007C	1	gpmc_d13	SBoot3	***	PUPD	U	11
U2	IO	Muxed	V_{DDS1}	gpmc_d14	0x007C	3	gpmc_d11	SBoot3	***	PUPD	U	11
R23	IO	Muxed	V_{DDS2}	mcbsp1_dx	0x010C	3	mcbsp1_fsx	Protect	111	PUPD	D	01
R19	IO	Dedicated	V_{DDS2}	mcbsp1_clkx	0x0110	2	i2c1_sda	Mode0	000	PU	U	11
K24	IO	Muxed	V_{DDS2}	i2c2_sda	0x0114	2	uart3_cts_rctx	Protect	111	PUPD	U	11
M20	IO	Muxed	V_{DDS2}	i2c2_sda	0x0114	3	uart3_rts_sd	Protect	111	PUPD	U	11
P3	IO	Muxed	V_{DDS1}	gpmc_ncs4	0x0090	3	gpmc_ncs7	Protect	111	PUPD	U	11
H12	IO	Muxed	V_{DDS2}	dss_d17	0x00C4	3	uart1_tx	Protect	111	PUPD	U	11
AE12	I	Dedicated	V_{DDS1}	gpmc_nbe1	0x0098	2	gpmc_wait0	Mode0	000	PUPD	U	11
AF19	O	Dedicated	V_{DDS1}	gpmc_ncs0	0x008C	0	gpmc_ncs0	Mode0	000	No	−	00
T1	IO	Muxed	V_{DDS1}	gpmc_d10	0x0080	0	gpmc_d10	SBoot3	***	PUPD	U	11
U1	IO	Muxed	V_{DDS1}	gpmc_d14	0x007C	2	gpmc_d12	SBoot3	***	PUPD	U	11
N2	IO	Muxed	V_{DDS1}	gpmc_wait2	0x009C	0	gpmc_wait2	Protect	111	PUPD	U	11
AA24	IO	Muxed	V_{DDS2}	spi2_somi	0x0108	1	spi2_cs0	Protect	111	PUPD	U	11
Y24	IO	Muxed	V_{DDS2}	spi2_somi	0x0108	2	mcbsp1_clkr	Protect	111	PUPD	D	01
P23	IO	Muxed	V_{DDS2}	mcbsp1_dx	0x010C	2	mcbsp_clks	Protect	111	PUPD	D	01
P24	IO	Dedicated	V_{DDS2}	mcbsp1_clkx	0x0110	1	i2c1_scl	Mode0	000	PU	U	11

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin					Pinout Control Register			Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
P20	IO	Muxed	V_{DDS2}	uart1_rx	0x00C8	1	mcbsp2_dr	Protect	111	PUPD	D	01
AB2	IO	Dedicated	V_{DDS1}	gpmc_d2	0x0088	2	gpmc_d0	Mode0	000	PUPD	U	11
AA2	IO	Dedicated	V_{DDS1}	gpmc_d2	0x0088	1	gpmc_d1	Mode0	000	PUPD	U	11
Y1	IO	Muxed	V_{DDS1}	gpmc_d10	0x0080	1	gpmc_d9	SBoot3	***	PUPD	U	11
V1	IO	Muxed	V_{DDS1}	gpmc_d10	0x0080	2	gpmc_d8	SBoot3	***	PUPD	U	11
AB24	IO	Muxed	V_{DDS2}	uart2_rts	0x00EC	1	uart2_tx	Protect	111	PUPD	U	11
W23	IO	Muxed	V_{DDS2}	spi1_simo	0x0100	3	spi1_cs1	Protect	111	PUPD	U	11
T23	IO	Muxed	V_{DDS2}	i2c2_sda	0x0114	1	hdq_sio	Protect	111	PUPD	U	11
T19	IO	Muxed	V_{DDS2}	mcbsp1_clkx	0x0110	0	mcbsp1_clkx	Protect	111	PUPD	D	01
T8	IO	Muxed	V_{DDS1}	gpmc_ncs0	0x008C	3	gpmc_ncs3	Protect	111	PUPD	U	11
AC2	IO	Dedicated	V_{DDS1}	gpmc_d2	0x0088	0	gpmc_d2	Mode0	000	PUPD	U	11
AB1	IO	Dedicated	V_{DDS1}	gpmc_d6	0x0084	3	gpmc_d3	Mode0	000	PUPD	U	11
AE20	IO	Muxed	V_{DDS1}	gpmc_nbe1	0x0098	3	gpmc_wait1	Protect	111	PUPD	U	11
AF14	IO	Muxed	V_{DDS1}	gpmc_ncs0	0x008C	1	gpmc_ncs1	Protect	111	PUPD	U	11
T4	IO	Muxed	V_{DDS1}	gpmc_wait2	0x009C	1	gpmc_wait3	Protect	111	PUPD	U	11
AC9	IO	Dedicated	V_{DDS}	dss_d1	0x00B4	0	dss_d1	Mode0	000	PUPD	D	01
AD12	IO	Muxed	V_{DDS}	dss_d9	0x00BC	0	dss_d9	Protect	111	PUPD	D	01
AD14	IO	Muxed	V_{DDS}	dss_d13	0x00C0	3	dss_d16	Protect	111	PUPD	D	01
AF15	IO	Muxed	V_{DDS}	cam_xclk	0x00DC	3	ssi1_rdy_tx	Mode0	000	PUPD	D	01
AE24	IO	Muxed	V_{DDS}	gpio_6	0x0138	2	gpio_125	Protect	111	PUPD	−	00
AD24	IO	Muxed	V_{DDS2}	uart2_rts	0x00EC	2	uart2_rx	Protect	111	PUPD	U	11

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin				Pinout Control Register				Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
U23	IO	Muxed	V_{DDS2}	mcbsp1_dx	0x010C	1	mcbsp1_dr	Protect	111	PUPD	D	01
R24	IO	Muxed	V_{DDS2}	mcbsp1_dx	0x010C	0	mcbsp1_dx	Protect	111	PUPD	D	01
R20	IO	Muxed	V_{DDS2}	spi2_somi	0x0108	3	mcbsp1_fsr	Protect	111	PUPD	D	01
T24	IO	Muxed	V_{DDS2}	uart1_rx	0x00C8	2	mcbsp2_clkx	Protect	111	PUPD	D	01
AF3	IO	Dedicated	V_{DDS1}	gpmc_d6	0x0084	2	gpmc_d4	Mode0	000	PUPD	U	11
AE3	IO	Dedicated	V_{DDS1}	gpmc_d6	0x0084	0	gpmc_d6	Mode0	000	PUPD	U	11
AF12	O	Muxed	V_{DDS1}	gpmc_nadv_ale	0x0094	3	gpmc_nbe0	Mode0	000	No	–	00
W3	IO	Muxed	V_{DDS}	gpio_121	0x0130	1	gpio_122	Protect	111	No	–	00
W7	IO	Muxed	V_{DDS}	eac_bt_fs	0x00F0	0	eac_bt_fs	Protect	111	PUPD	D	01
U3	O	Muxed	V_{DDS1}	gpmc_nbe1	0x0098	0	gpmc_nbe1	Mode0	000	No	–	00
AD10	IO	Dedicated	V_{DDS}	dss_d5	0x00B8	0	dss_d5	Mode0	000	PUPD	D	01
AD13	IO	Muxed	V_{DDS}	dss_d13	0x00C0	0	dss_d13	Protect	111	PUPD	D	01
Y13	IO	Muxed	V_{DDS}	dss_d17	0x00C4	0	dss_d17	Protect	111	PUPD	D	01
AD18	IO	Muxed	V_{DDS}	gpio_62	0x00E0	0	gpio_62	Protect	111	PUPD	U	11
AD16	IO	Muxed	V_{DDS}	eac_ac_sclk	0x0124	1	eac_ac_fs	Protect	111	PUPD	D	01
V23	IO	Muxed	V_{DDS2}	spi1_cs2	0x0104	0	spi1_cs2	Protect	111	PUPD	U	11
U24	IO	Muxed	V_{DDS2}	spi1_cs2	0x0104	3	spi2_simo	Protect	111	PUPD	D	01
V25	IO	Muxed	V_{DDS2}	spi2_somi	0x0108	0	spi2_somi	Protect	111	PUPD	D	01
V8	IO	Dedicated	V_{DDS1}	gpmc_d6	0x0084	1	gpmc_d5	Mode0	000	PUPD	U	11
AE5	IO	Dedicated	V_{DDS1}	gpmc_d10	0x0080	3	gpmc_d7	Mode0	000	PUPD	U	11
AD4	IO	Muxed	V_{DDS}	uart2_rts	0x00EC	3	eac_bt_sclk	Protect	111	PUPD	D	01

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin					Pinout Control Register			Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
AE6	IO	Muxed	V_{DDS}	gpio_121	0x0130	0	gpio_121	Protect	111	No	–	00
V19	IO	Muxed	V_{DDS2}	spi1_simo	0x0100	1	spi1_somi	Protect	111	PUPD	D	01
V24	IO	Muxed	V_{DDS2}	spi1_ncs2	0x0104	2	spi2_clk	Protect	111	PUPD	D	01
T20	IO	Muxed	V_{DDS2}	uart1_rx	0x00C8	0	uart1_rx	Protect	111	PUPD	U	11
V4	IO	Muxed	V_{DDS}	cam_d0	0x00D8	1	cam_hs	Protect	111	PUPD	D	01
Y4	IO	Muxed	V_{DDS}	cam_d0	0x00D8	0	cam_d0	Protect	111	PUPD	U	11
Y23	IO	Muxed	V_{DDS2}	mmc_dat_dir3	0x00FC	3	spi1_clk	Protect	111	PUPD	D	01
W24	IO	Muxed	V_{DDS2}	spi1_simo	0x0100	2	spi1_cs0	Protect	111	PUPD	U	11
P7	IO	Muxed	V_{DDS}	cam_d0	0x00D8	2	cam_vs	Protect	111	PUPD	D	01
U20	IO	Muxed	V_{DDS2}	spi1_cs2	0x0104	1	spi1_cs3	Protect	111	PUPD	U	11
W4	O	Muxed	V_{DDS}	cam_xclk	0x00DC	0	cam_xclk	Mode0	000	No	–	00
AB3	IO	Muxed	V_{DDS}	cam_d4	0x00D4	1	cam_d3	Protect	111	PUPD	D	01
AC12	IO	Muxed	V_{DDS}	dss_d9	0x00BC	1	dss_d10	Protect	111	PUPD	D	01
AC13	IO	Muxed	V_{DDS}	dss_d13	0x00C0	1	dss_d14	Protect	111	PUPD	D	01
AC15	IO	Muxed	V_{DDS}	cam_xclk	0x00DC	2	ssi1_flag_tx	Mode0	000	PUPD	–	00
AC16	IO	Muxed	V_{DDS}	gpio_62	0x00E0	3	ssi1_rdy_rx	Mode0	000	PUPD	–	00
Y17	IO	Muxed	V_{DDS}	eac_ac_mclk	0x0128	0	eac_ac_mclk	Protect	111	PUPD	D	01
AF22	IO	Muxed	V_{DDS}	eac_ac_sclk	0x0124	3	eac_ac_dout	Protect	111	PUPD	D	01
AC19	O	Dedicated	V_{DDS}	sys_xtalout	0x0134	0	sys_xtalout	Mode0	000	No	–	00
Y18	IO	Muxed	V_{DDS}	sys_xtalout	0x0134	1	gpio_36	Protect	111	No	–	00
Y19	IO	Muxed	V_{DDS2}	gpio_6	0x0138	1	gpio_124	Protect	111	PUPD	–	00

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin				Pinout Control Register				Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
AC24	IO	Muxed	V_{DDS2}	vlynq_tx1	0x00E8	3	uart2_cts	Protect	111	PUPD	U	11
Y3	IO	Muxed	V_{DDS}	cam_d4	0x00D4	3	cam_d1	Protect	111	PUPD	D	01
H10	IO	Muxed	V_{DDS2}	spi1_simo	0x0100	0	spi1_simo	Protect	111	PUPD	D	01
U7	IO	Muxed	V_{DDS}	cam_d4	0x00D4	2	cam_d2	Protect	111	PUPD	D	01
AD3	IO	Muxed	V_{DDS}	cam_d8	0x00D0	3	cam_d5	Protect	111	PUPD	D	01
AD6	IO	Muxed	V_{DDS}	cam_d0	0x00D8	3	cam_lclk	Protect	111	PUPD	D	01
AC7	IO	Muxed	V_{DDS}	dss_vsync	0x00CC	3	cam_d9	Protect	111	No	−	00
AF8	IO	Dedicated	V_{DDS}	dss_vsync	0x00CC	0	dss_vsync	Mode0	000	PUPD	D	01
AC8	IO	Dedicated	V_{DDS}	dss_d1	0x00B4	1	dss_d2	Mode0	000	PUPD	D	01
AC11	IO	Dedicated	V_{DDS}	dss_d5	0x00B8	2	dss_d7	Mode0	000	PUPD	D	01
AE13	IO	Muxed	V_{DDS}	dss_d9	0x00BC	3	dss_d12	Protect	111	PUPD	D	01
Y12	IO	Muxed	V_{DDS}	dss_d13	0x00C0	2	dss_d15	Protect	111	PUPD	D	01
W13	IO	Muxed	V_{DDS}	cam_xclk	0x00DC	1	ssi1_dat_tx	Mode0	000	PUPD	−	00
AF11	IO	Muxed	V_{DDS}	gpio_62	0x00E0	2	ssi1_flag_rx	Mode0	000	PUPD	−	00
AE19	IO	Muxed	V_{DDS}	sys_xtalout	0x0134	3	sys_clkout	Protect	111	PUPD	D	01
AD19	IO	Muxed	V_{DDS}	eac_ac_sclk	0x0124	2	eac_ac_din	Protect	111	PUPD	D	01
AE22	IO	Muxed	V_{DDS}	eac_ac_mclk	0x0128	1	eac_ac_rst	Protect	111	PUPD	D	01
Y20	I	Muxed	V_{DDS2}	sys_nirq	0x012C	0	sys_nirq	Protect	111	PUPD	U	11
V2	IO	Muxed	V_{DDS}	cam_d4	0x00D4	0	cam_d4	Protect	111	PUPD	D	01
W20	IO	Muxed	V_{DDS2}	uart2_rts	0x00EC	0	uart2_rts	Protect	111	PUPD	U	11
AA4	IO	Muxed	V_{DDS}	cam_d8	0x00D0	2	cam_d6	Protect	111	No	−	00

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin					Pinout Control Register			Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
AD5	IO	Muxed	V_{DDS}	eac_bt_fs	0x00F0	2	eac_bt_dout	Protect	111	PUPD	D	01
AF6	IO	Muxed	V_{DDS}	sys_nirq	0x012C	2	gpio_119	Protect	111	No	–	00
AF5	IO	Dedicated	V_{DDS}	uart1_rx	0x00C8	3	dss_pclk	Mode0	000	PUPD	D	01
AE8	IO	Muxed	V_{DDS}	dss_vsync	0x00CC	2	dss_acbias	Protect	111	PUPD	D	01
AD9	IO	Dedicated	V_{DDS}	dss_d1	0x00B4	3	dss_d4	Mode0	000	PUPD	D	01
AD11	IO	Muxed	V_{DDS}	dss_d5	0x00B8	3	dss_d8	Protect	111	PUPD	D	01
AE11	IO	Muxed	V_{DDS}	dss_d9	0x00BC	2	dss_d11	Protect	111	PUPD	D	01
W14	IO	Muxed	V_{DDS}	vlynq_tx1	0x00E8	0	vlynq_tx1	Protect	111	PUPD	D	01
W15	IO	Muxed	V_{DDS}	gpio_62	0x00E0	1	ssi1_dat_rx	Mode0	000	PUPD	–	00
AD15	IO	Muxed	V_{DDS}	ssi1_wake	0x00E4	0	ssi1_wake	Mode0	000	PUPD	–	00
W17	I	Dedicated	V_{DDS}	eac_ac_mclk	0x0128	2	sys_nrespwron	Mode0	000	No	–	00
AC18	IO	Muxed	V_{DDS}	eac_ac_sclk	0x0124	0	eac_ac_sclk	Protect	111	PUPD	D	01
AE2	IO	Dedicated	V_{DDS}	eac_ac_mclk	0x0128	3	sys_nreswarm	Mode0	000	PUPD	U	11
AD21	I	Dedicated	V_{DDS}	gpio_121	0x0130	2	sys_32k	Mode0	000	No	–	00
AC20	I	Dedicated	V_{DDS}	gpio_121	0x0130	3	sys_xtalin	Mode0	000	No	–	00
AB4	IO	Muxed	V_{DDS}	cam_d8	0x00D0	1	cam_d7	Protect	111	No	–	00
W16	O	Dedicated	V_{DDS2}	sys_nirq	0x012C	1	sys_nvmode	Mode0	000	No	–	00
N24	IO	Muxed	V_{DDS2}	jtag_emu0	0x013C	0	jtag_emu0	Protect	111	PUPD	U	11
U8	IO	Muxed	V_{DDS2}	eac_bt_fs	0x00F0	1	eac_bt_din	Protect	111	PUPD	D	01
AC6	IO	Muxed	V_{DDS}	cam_d8	0x00D0	0	cam_d8	Protect	111	No	–	00
AF4	IO	Muxed	V_{DDS}	sys_nirq	0x012C	3	gpio_120	Protect	111	No	–	00

Table 1−1. OMAP2423 Pin Characteristics (Continued)

Pin				Pinout Control Register				Control		Pull		
Ball 2423	Type	Mux Type	Power	Name Extension	Offset	Byte	Pin Name	Reset State	Reset Value	Type	Reset State	Reset Value
AD8	IO	Dedicated	V_{DDS}	dss_vsync	0x00CC	1	dss_hsync	Mode0	000	PUPD	U	11
Y9	IO	Dedicated	V_{DDS}	sdrc_dqs1	0x00B0	3	dss_d0	Mode0	000	PUPD	D	01
AE9	IO	Dedicated	V_{DDS}	dss_d1	0x00B4	2	dss_d3	Mode0	000	PUPD	D	01
W11	IO	Dedicated	V_{DDS}	dss_d5	0x00B8	1	dss_d6	Mode0	000	PUPD	D	01

Chapter 1

Introduction

This chapter introduces the features, supporting subsystems, and architecture of the OMAP2423 high-performance, multimedia application system-in-package (SIP) device.

Topic **Page**

1.1 Overview 1-2

1.2 OMAP2423 Description 1-4

1.3 Distinguishing Between the OMAP2423 and OMAP242x Devices and the Silicon Versions 1-11

1.4 Major Differences Between the OMAP2422 and OMAP2423 Devices 1-14

1.1 Overview

The OMAP2423 high-performance, multimedia application SIP processor is composed of the OMAP2420 device and two stacked mobile dual data rate (DDR) synchronous dynamic random access memory (SDRAM) modules. The OMAP2423 processor is based on the enhanced OMAP™ 2.0 architecture and integrated on the advanced 90-nm process technology from Texas Instruments, Inc. (TI).

The OMAP2423 processor delivers low-power enhanced video imaging, audio, and graphics processing to support these high-quality multimedia features:

- Streaming video
- 2D/3D electronic gaming
- Video conferencing
- Video recording
- Audio
- High-resolution, still-image capture (used in 2.5G and 3G wireless terminals and high-performance PDAs)

The following subsystems support the OMAP2423 processor:

- Microprocessor unit (MPU) subsystem based on the ARM1136JF-S microprocessor
- Digital signal processor (DSP) subsystem based on the industry leading TMS320C55x™ (C55x™) DSP, which supports a wide range of audio and video applications
- Imaging and video accelerator (IVA) subsystem with an MPU and imaging accelerators to enable high-end imaging and video applications
- GFX subsystem for 2D and 3D graphics acceleration to support display and gaming effects
- Camera subsystem that supports multiple formats and interfacing options connected to a wide variety of image sensors
- Display subsystem with a wide variety of features for multiple concurrent image manipulation as well as a programmable interface supporting a wide variety of displays. The display subsystem also supports NTSC/PAL video out.
- Level 3 (L3) and Level 4 (L4) interconnect that provides high-bandwidth data transfer for multiple initiators to the internal and external memory controllers and to on-chip peripherals

The OMAP2423 processor supports high-level operating systems, such as:

- Windows CE
- Symbian OS
- Linux
- Palm OS

TI and multiple third parties offer a rich and comprehensive set of video, still picture, and audio codecs that run on the DSP and IVA processors.

Security features integrated on the devices support applications designed for:

- Protection against malicious attacks
- M-commerce
- Content protection for recordable media (CPRM)
- Digital rights management (DRM)

High-security devices rely on a security scheme based on hardware mechanisms and a secure read-only memory (ROM) code, ensuring that only trusted code can access the secure resources. These resources are located in specific regions of memory as well as in peripherals, hardware cryptographic accelerators, and eFuse keys. General-purpose devices that do not use the security features can still use the cryptographic hardware accelerators.

The OMAP2423 processor also offers:

- Comprehensive power and clock-management scheme enabling high-performance, low-power operation, and ultralow-power standby features
- Connection to TI 2.5G or 3G modem chipsets for a complete terminal solution
- Core operating voltage range from 1.05 V to 1.3 V nominal and interface specified at 1.8 V nominal

1.2 OMAP2423 Description

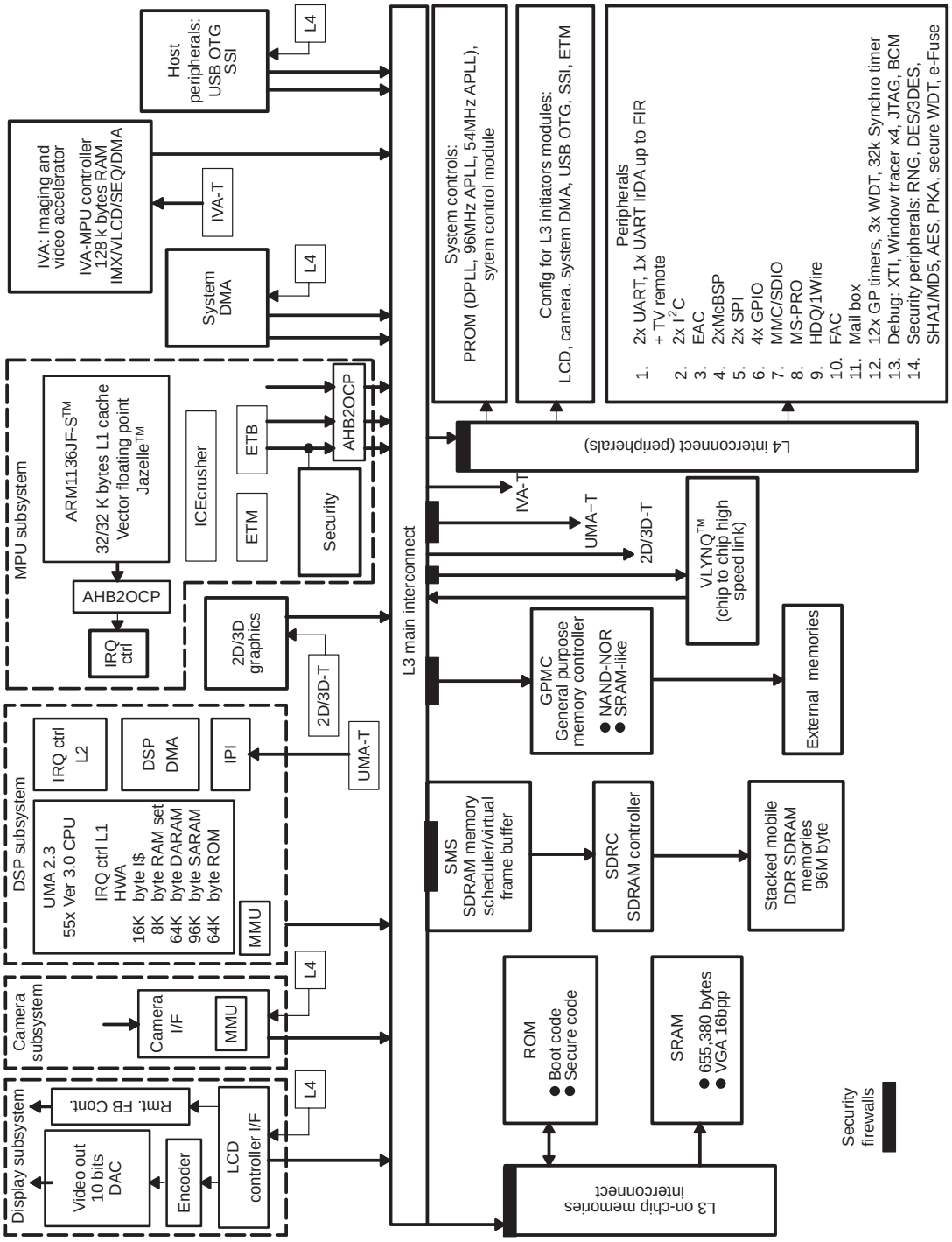
The OMAP2423 device is a stacked package version of the OMAP2420 device in which two mobile DDR SDRAM memories are stacked onto the OMAP2420 device.

The OMAP2423 processor is offered in the 447-ball 14 x 14mm, 0.5mm pitch package.

The OMAP2423 footprint is not compatible with the OMAP2420 footprint but is compatible with the OMAP2420 POP 768-Mb version. See the *OMAP2423 Data Manual* (literature number SWPS019B).

Figure 1−1 shows the OMAP2423 block diagram.

Figure 1−1. OMAP2423 Block Diagram



1.2.1 MPU Subsystem (Based on ARM1136JF-S)

The MPU general-purpose processor consists of the following:

- ❑ v6 instruction set architecture (ISA)
- ❑ Caches
  - ■ 32K-byte instruction and 32K-byte data; 4-way set associative memory management units (MMUs)
  - ■ 64-entry instruction and 64-entry data write buffer
- ❑ Vector floating-point processor
- ❑ Jazelle Java accelerator
- ❑ ICECrusher/ETB/ETM for emulation and trace

1.2.2 TMS320C55x DSP Subsystem

The general-purpose, fixed-point C55x DSP, designed for applications such as audio and video acceleration, includes these features:

- ❑ Revision 3.0 core with reduced power consumption and improved performance based on higher parallelism and new instructions
- ❑ 16K-byte, 2-way set associative instruction cache
- ❑ Two 4K-byte random access memory (RAM) sets
- ❑ 64K-byte, dual-access static random access memory (SRAM)
- ❑ 96K-byte, single-access SRAM
- ❑ 64K-byte ROM
- ❑ Multichannel direct memory access controller (dDMA):
  - ■ 1 read port, 1 write port
  - ■ 24 logical channels
  - ■ 32 hardware requests
  - ■ 128- x 64-bit FIFO depth
- ❑ Video decoder/encoder hardware accelerators

1.2.3 On-Chip Memory

The on-chip memory configuration offers secure and non-secure memory resources for general-purpose program and data storage:

- ❑ 96K-byte ROM
- ❑ 640K-byte, single-access SRAM (up to VGA 16 bpp)

1.2.4 Memory Interfaces

The OMAP2423 processor includes two memory interfaces. The first, the general-purpose memory controller (GPMC), is dedicated to external memory. The second, the SDRAM controller (SDRC), is dedicated to two stacked mobile DDR SDRAM memory modules.

- GPMC
  - NOR flash, NAND flash, SRAM, and PSRAM asynchronous and synchronous protocols
  - Flexible asynchronous protocol control for external ASIC or peripheral interfacing
  - 16-bit data, up to eight chip-selects, 128M-byte address bus, 1G-byte total address space
- SDRC
  - 1 x 32MB stacked mobile DDR SDRAM, 133 MHz
  - 1 x 64MB stacked mobile DDR SDRAM, 166 MHz
  - 32-bit data, two chip-selects

1.2.5 DMA Controllers

DMA controllers use system DMA (SDMA) configured for memory-to-memory, memory-to-peripheral, and peripheral-to-memory transfers:

- 1 read port, 1 write port
- 32 logical channels
- 256 x 32 bits FIFO depth

1.2.6 Multimedia Accelerators

The OMAP2423 processor uses the following multimedia accelerators for display and gaming effects and high-end imaging and video applications:

- 2D and 3D graphics accelerator (GFX)
  - Supports geometrical functions and rasterization
  - Minimizes external memory bandwidth and power with unique tile-based rendering
  - Can sustain 2M triangles per second and up to 400 equivalent M pixels per second (100M pixels, depth of complexity of 4)
  - Target screen size is HVGA at 30 fps with a depth complexity of 4
  - Supports application programming interfaces such as OpenGL ES and Direct3D Mobile
- IVA for large screens, high frame rates, and low power, for example, H.263, H26L, MPEG4, JPEG, MJPEG encode and decode
  - MPEG4 encode
  - MPEG4 decode

- VTC + recording (2 encode + 1 decode)
- H.263 video conferencing
- JPEG
  - RGB Bayer (CCD) requiring full image pipeline
  - YUV data (no image pipeline required)

- Camera interface
  - Up to 10 bits data in Bayer RGB and ITU-R BT.656 formats
  - CCP standard
  - Integrated DMA and MMU for high-bandwidth data transfer
- Display controller
  - Color and monochrome displays up to 2048 X 2048 X 24 bpp resolution
  - Picture-in-picture (overlay), color-space conversion, rotation, resizing support
  - NTSC/PAL video encoder with integrated DAC output
  - MeSSI standards

1.2.7 Security

The secure firmware resides in ROM and includes hardware security features that enable high-security devices and the following encryption/decryption accelerators:

- RNG
- DES/3DES
- SHA1/MD5
- AES
- PKA

1.2.8 Comprehensive Power Management

- Clock and reset generation and distribution
- Wake-up event management
- Low-voltage operation modes
- Advanced leakage-management techniques to achieve ultralow standby power

1.2.9 Peripherals

The OMAP2423 processor supports a comprehensive set of peripherals to provide flexible and high-speed interfacing and on-chip programming resources:

- Enhanced audio controller (EAC)

  Provides three programmable audio interfaces enabling low power, multiple audio source mixing, and sample rate conversion

- Universal asynchronous receiver/transmitter (UART)1/2

  Two general serial communication interfaces

- UART3

  UART + IrDA up to FIR + TV remote control interface

- Multichannel buffered serial port (McBSP)1/2

  Two general-purpose multichannel buffered serial interfaces

- I^2C1/2

  Two master/slave I^2C standard interfaces

- HDQ/1-wire

  Benchmark HDQ and Dallas Semiconductor 1-Wire protocol interfaces

- McSPI1/2

  Two multichannel serial-port interface (SPI) controllers

- Synchronous/serial interconnect (SSI) controller

  Dual-port, synchronous serial-interface controller

- Universal serial bus (USB) OTG

  Multiport USB 2.0 full/half-speed USB host/client controllers

- VLYNQ™

  High-speed serial communications interface

- MS-PRO

  MS or MS-Pro memory card interface

- MMC/SD

  Interface controller for MMC/SD/SDIO standards

- General-purpose (GP) timers

  12 GP timers

- Watchdog timer (WDT)

  Four watchdog timers

- 32-kHz timer

  32-kHz clock timer

- Quad general-purpose I/O controllers

  Four 32-bit GPIOs

- Mailbox

  MPU/DSP/IVA interprocessor communications

- Control

  I/O multiplexing and chip-configuration control

1.2.10 Debug Support

Comprehensive and concurrent JTAG-based, multiprocessor debug support uses the TI XDS560™, ARM Ltd. Multi-ICE/Real-ICE, or third parties, such as Lauterbach emulation tools. The ARM Ltd. embedded trace macrocell (ETM), in conjunction with an embedded 4K-byte trace buffer, supports the MPU trace.

Comprehensive and concurrent JTAG-based, multiprocessor debug support uses the TI XDS560™, ARM Ltd. Multi-ICE/Real-ICE, or third parties, such as Lauterbach emulation tools. For more information about these emulation/debugging environments, see the *Configuring JTAG Emulators for the OMAP24xx Software Development Platform* (TI Literature number SWPA037) and *Configuring XDS510 USB JTAG Emulator and Code Composer Studio for the OMAP2420 Device SDP* (TI Literature number SWPA057) application notes available through TI representatives.

The ARM Ltd. embedded trace macrocell (ETM), in conjunction with an embedded 4K-byte trace buffer, supports the MPU trace.

TI cTOOLS supports the DSP trace. Window tracers and external trace trigger modules allow trace of internal bus activity and communication to the software programmer for debug.

Contact your TI representative for emulation tools and an emulation technical reference manual.

1.3 Distinguishing Between the OMAP2423 and OMAP242x Devices and the Silicon Versions

You can retrieve the silicon type, silicon die, and silicon revision number on the OMAP2420 device using the software in three register read operations with the IDCODE_reg, the DIE_ID_reg, and the PRODUCTION_ID_reg in the TEST-Chip-level test access port (TAP) (mapped on the L4 interconnect port).

Table 2−1 and Table 2−2 describe the IDCODE_reg register.

See Table 2-8, *L4 Peripheral Memory Space Mapping*, in the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D) for the memory space address of the TEST-Chip-level TAP device.

Table 2−1. Device Identification Register (IDCODE_reg)

Register	Address	Size	JTAG Access	OCP Access	Reset Value
IDCODE_reg	0x48014204	32	R	R	NA

The silicon type can be read in the HAWKEYE bit field value of IDCODE_reg.

The silicon revision can be read in the VERSION bit field value of IDCODE_reg.

The VERSION is connected directly to the DESIGNREV bit field DIE_ID_reg[63:60] which contains the version number of the chip written through eFuse.

Table 2−2. IDCODE_reg Register Definition

Field	Bits	Value	Comment
IDCODE_reg [31:28]	VERSION	See Table 2−4	Revision number
IDCODE_reg [27:12]	HAWKEYE	See Table 2−3	Hawkeye number
IDCODE_reg [11:1]	TI_IDM	–†	Manufacturer identity
IDCODE_reg [0]	–	0x1	Always set to 1

† TI internal data

The Hawkeye number is hardcoded in the design.

Table 2−3. Hawkeye Number Value

Silicon Type	Field	Value
OMAP2420	IDCODE_reg [27:12]	0xB5D9
OMAP2423	IDCODE_reg [27:12]	0xB5D9

Note:

Because the OMAP2423 device is a stacked package version of the OMAP2420 device (in which two mobile DDR SDRAM memories are stacked onto the OMAP2420 die), the Hawkeye of the OMAP2423 device is also the same as that for the OMAP2420 device. Only the version number may differ.

Table 2−4. Revision Number Value

Revision	Field	Value
ES 1.0	IDCODE_reg [31:28]	0000
ES 2.0	IDCODE_reg [31:28]	0001
ES 2.05	IDCODE_reg [31:28]	0010
ES 2.10	IDCODE_reg [31:28]	0011
ES 2.1.1	IDCODE_reg [31:28]	0100
ES 2.2	IDCODE_reg [31:28]	0101

The IDCODE_reg value is 0x0B5D902F for OMAP2420 ES1.0.

The IDCODE_reg value is 0x1B5D902F for OMAP2420 ES2.0.

The IDCODE_reg value is 0x2B5D902F for OMAP2420 ES2.05.

The IDCODE_reg value is 0x3B5D902F for OMAP2420 ES2.10.

The IDCODE_reg value is 0x4B5D902F for OMAP2420 ES2.1.1.

The IDCODE_reg value is 0x5B5D902F for OMAP2420 ES2.2.

You can retrieve the silicon die with the DIE_ID_reg register, described in Table 2−5 to Table 2−9.

The value of the DIE_ID_reg register identifies uniquely each silicon die: each die ID is unique as it is created from the wafer, lot, and factory numbers, plus the die x/y coordinates in the wafer.

DIE_ID_reg register is a 128-bit register split into four 32-bit registers: DIE_ID_0, DIE_ID_1, DIE_ID_2, and DIE_ID_3.

Table 2−5. Die Identification Register (DIE_ID_reg)

Register	Address	Size	JTAG Access	OCP Access	Reset Value
DIE_ID_0	0x48014218	32	R	R	NA
DIE_ID_1	0x4801421C	32	R	R	NA
DIE_ID_2	0x48014220	32	R	R	NA
DIE_ID_3	0x48014224	32	R	R	NA

Table 2−6. DIE_ID_0 Register Definition

Field	Bits	Value	Comment
DIE_ID_reg [31:0]	Reserved	0x−−−−−−−−	Reserved for special identification

Table 2−7. DIE_ID_1 Register Definition

Field	Bits	Value	Comment
DIE_ID_reg [63:60]	DESIGNREV	See Table 2−4	Device revision
DIE_ID_reg [59:32]	Reserved	0x−−−−−−−−	Reserved for special identification

Table 2−8. DIE_ID_2 Register Definition

Field	Bits	Value	Comment
DIE_ID_reg [95:64]	Reserved	0x−−−−−−−−	Reserved for special identification

Table 2−9. DIE_ID_3 Register Definition

Field	Bits	Value	Comment
DIE_ID_reg [127:96]	Reserved	0x−−−−−−−−	Reserved for special identification

The DESIGNREV register field (DIE_ID_reg[63:60]) represents the design revision sent to the IDCODE_reg register.

You can retrieve the device type in one read register operation using the PRODUCTION_ID_reg register as described in Table 2−10 and Table 2−11.

Table 2−10. Production Identification Register (PRODUCTION_ID_reg)

Register	Address	Size	JTAG Access	OCP Access	Reset Value
PRODUCTION_ID_reg	0x48014208	32	R	R	NA

Table 2−11. Production Identification Register (PRODUCTION_ID_reg) Definition

Field	Bits	Value	Comment
PRODUCTION_ID_reg [31:20]	Reserved	–	Reserved
PRODUCTION_ID_reg [19:16]	DEVICE_TYPE	See Table 2−12	OMAP242x device type
PRODUCTION_ID_reg [15:0]	Reserved	–†	Reserved

† TI internal data

Depending on the DEVICE_TYPE register field PRODUCTION_ID_reg[19:16], you can retrieve the OMAP242x device type used.

Table 2−12 lists the different values for the DEVICE_TYPE register field.

Table 2−12. Device_Type Value

Device Type	Field	Device_Type Bit Value			
		19	18	17	16
OMAP2420/22/23 unprogrammed	PRODUCTION_ID_reg [19:16]	0	0	0	0
OMAP2420	PRODUCTION_ID_reg [19:16]	0	0	0	1
OMAP2420 POP	PRODUCTION_ID_reg [19:16]	0	0	1	0
OMAP2422 Mono (1x512Mb)	PRODUCTION_ID_reg [19:16]	0	1	0	0
OMAP2423	PRODUCTION_ID_reg [19:16]	1	0	0	0

Note:

This implementation is not available on all devices. To retrieve the device type on unprogrammed devices (PRODUCTION_ID_reg[19:16] = 0000), use the printed device reference.

1.4 Major Differences Between the OMAP2422 and OMAP2423 Devices

Although both the OMAP2422 and OMAP2423 devices are stacked memory versions of the OMAP2420 device, the devices differ as follows:

- Size of stacked memories
  - OMAP2422 device: 2 X 256Mb
  - OMAP2423 device: 512Mb + 256Mb

    The 512Mb DDR SDRAM memory is mapped to chip-select 0 (see Chapter 2 for details).
- The OMAP2423 device has no internal connection between the OMAP2420 die and the DDR die TempWARN (see Chapter 12 for details).
- The OMAP2423 device has dedicated DDR VDDQ supplies to the stacked mobile DDR SDRAM memories.
- Chip package (see Chapter 29 for details)
  - OMAP2422 device: 325 balls, 13x13 mm
  - OMAP2423 device: 447 balls, 14x14 mm
- Ball out (see Chapter 29 for details)
- The OMAP2423 ball out is compatible with the OMAP2420 POP 768Mb.

Chapter 2

Memory Mapping

This chapter describes the memory mapping of the OMAP2423 multimedia system-in-package (SIP) processor.

Topic **Page**

2.1 Introduction **2-2**

2.2 Memory Mapping **2-4**

2.1 Introduction

The OMAP2423 MPU can handle 4G bytes with its 32-bit address port. The system memory mapping provides two levels of granularity for space address allocation. The first level features 1G-byte granularity with four quarters (Q0 to Q3) of 1GB. Then, as second granularity level, system targets are mapped on 128M bytes inside each 1GB quarter.

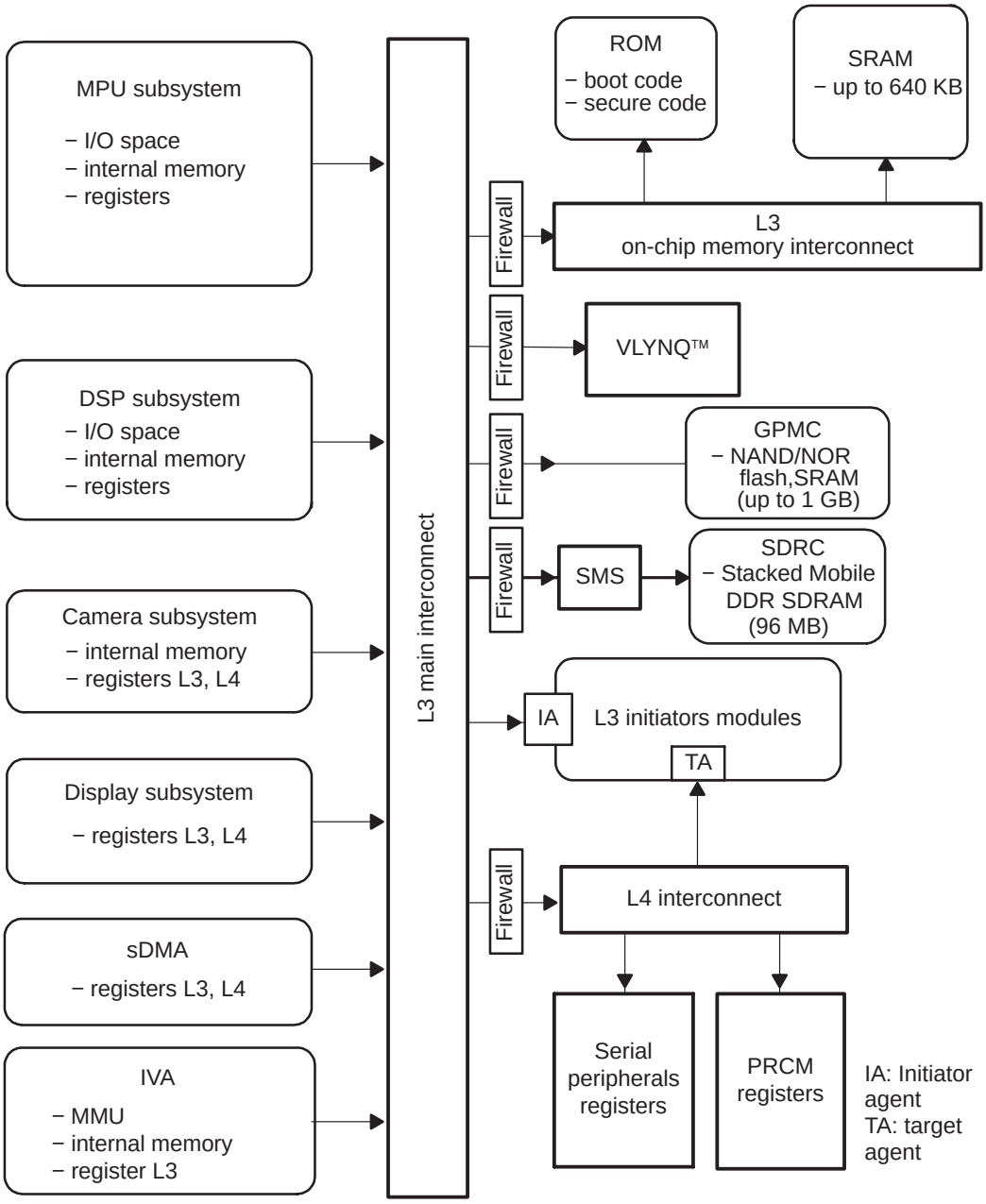
Memory map space is composed of memory space (GPMC, SDRC,…), register space (L3, L4 interconnect), and I/O space (DSP subsystem,…), which are shared among the OMAP2423 modules.

Each module/subsystem is connected to an interconnect port via a dedicated interface agent, which can be configured to tune the access depending on the characteristics of each module.

Internal accesses are conditioned by firewalls defined in Chapter 6 and in Table 6−7 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Figure 2−1 shows the memory type of each OMAP2423 module.

Figure 2−1. OMAP2423 Memory Organization and Type



2.2 Memory Mapping

2.2.1 Global Memory Space Mapping

This section provides a global view of the OMAP2423 memory mapping. The following memory spaces are detailed:

- Boot space

  The system boots in a 1M-byte boot space. This area is located either in the internal ROM or in the GPMC memory space.

  If the boot from internal ROM is selected, this 1M-byte memory space is redirected to the on-chip boot ROM memory address space [0x4000 0000 – 0x400F FFFF].

  If the boot from general-purpose memory controller (GPMC) is selected, this memory space is part of the GPMC memory space. At reset the 0x0000 0000 address is available on chip-select zero (GPMC.nCS0) for a memory size of 16M bytes.

- GPMC space

  Eight chip-selects (GPMCn.CS0 to GPMCn.CS7) are available on Q0 addressing space for the GPMC (general-purpose memory controller for NOR/NAND flash, SRAM memories). These chip-selects have programmable size and programmable base address in the total memory space of first quarter Q0 (1GB).

  With the GPMC, software can access NAND-flash, NOR-flash, synchronous-, and asynchronous-style protocol interfaces.

- SDRC space

  Two chip-selects (SDRC.nCS0 and SDRC.nCS1) are available on the third quarter Q2 addressing space for stacked mobile DDR SDRAM.

  The chip-select 0 memory space (selected by the SDRC.nCS0 signal) is dedicated to 512M-bit mobile DDR SDRAM1 and its base address is always mapped at 0x8000 0000.

  The chip-select 1 memory space (selected by the SDRC.nCS1 signal) is dedicated to 256M-bit mobile DDR SDRAM2 and its base address is programmable through SDRC_CS_CFG register fields CS1STARTLOW and CS1STARTHIGH. See Section 12.12.

  According to the SDRC_CS_CFG register, the chip-select 1 base address is 0xA000 0000 (CS1STARTLOW = 0x0 and CS1STARTHIGH = 0x4) at reset and ranges from 0x8400 0000 (CS1STARTLOW = 0x2 and CS1STARTHIGH = 0x0) to 0xBFFF FFFF (CS1STARTLOW = 0x3 and CS1STARTHIGH = 0x7) following the CS1STARTLOW and CS1STARTHIGH values programmed.

  The SDRC-SMS virtual memory space is a different memory space used to access the memory through the rotation engine (see Chapter 12).

Table 3−1 describes the memory space mapping.

Table 3−1. OMAP2423 Memory Space Mapping

Device Name		Start Address (hex)	End Address (hex)	Size	Description
GPMC/ boot space−Q0		0x0000 0000	0x3FFF FFFF	1GB 1 MB	8/16-bit Ex/R/W See Section 2.2.1 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM* version D (SWPU064D).
Internal boot ROM−Q1	Public	0x4000 0000	0x4000 7FFF	32KB	32-bit Ex/R
	Secure	0x4000 8000	0x4001 7FFF	64KB	32-bit Ex/R
	Reserved	0x4001 8000	0x400F FFFF	928KB	
Reserved		0x4010 0000	0x401F FFFF	1MB	
Internal SRAM−Q1	Secure	0x4020 0000	0x4020 F7FF	62KB	32-bit Ex/R/W
	Public	0x4020 F800	0x4020 FFFF	2KB	32-bit Ex/R/W
	Secure	0x4021 0000	0x4029 FFFF	576KB	32-bit Ex/R/W
Reserved (SRAM)		0x402A 0000	0x402F FFFF	384KB	
Reserved		0x4030 0000	0x47FF FFFF	125MB	
L4−peripherals−Q1		0x4800 0000	0x4FFF FFFF	128MB	See Section 2.2.3.3 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM* version D (SWPU064D).
GFX−Q1		0x5000 0000	0x5000 FFFF	64KB	
Reserved (GFX)		0x5001 0000	0x53FF FFFF	65472KB	
CMDWR−Q1 (emulation)		0x5400 0000	0x5400 FFFF	64KB	
Reserved (emulation)		0x5401 0000	0x57FF FFFF	65472KB	
DSP SS−Q1	dmemory	0x5800 0000	0x5802 7FFF	160KB	DARAM + SARAM
	Reserved	0x5802 8000	0x5803 FFFF	96KB	
	dIPI	0x5900 0000	0x5900 0FFF	4KB	
	dMMU	0x5A00 0000	0x5A00 0FFF	4KB	
	Reserved	0x5A00 1000	0x5BFF FFFF	~64MB	
IVA−Q1	IVA memory	0x5C00 0000	0x5C07 FFFF	512KB	See Table 2−9 and Table 2−10 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM* version D (SWPU064D).
	Reserved	0x5C08 0000	0x5CFF FFFF	15.5MB	
	iMMU	0x5D00 0000	0x5D00 0FFF	4KB	
	Reserved	0x5D00 1000	0x5FFF FFFF	48MB	
VLYNQ™−Q1		0x6000 0000	0x67FF FFFF	128MB	Remote memory space and configuration registers

Note: X = 0xA000 0000 @ reset, 0x8400 0000 ≤ X ≤ 0xBFFF FFFF

Table 3−1. OMAP2423 Memory Space Mapping (Continued)

Device Name		Start Address (hex)	End Address (hex)	Size	Description
L3−control registers−Q1		0x6800 0000	0x6FFF FFFF	128MB	See Table 2−7 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM* version D (SWPU064D).
SDRC−SMS Q1		0x7000 0000	0x7FFF FFFF	256MB	SDRC−SMS virtual memory space
SRDC/SMS−Q2	CS0	0x8000 0000	0x83FF FFFF	64MB	Stacked mobile DDR SDRAM1 memory space
SRDC/SMS−Q2	CS1	0x8400 0000			Stacked mobile DDR SDRAM2 programmable memory space
		X(1)	X+0x01FF FFFF	32MB	
			0xBFFF FFFF		
Not used−Q3		0xC000 0000	0xFFFF FFFF	1 GB	Future use

Note: X = 0xA000 0000 @ reset, 0x8400 0000 ≤ X ≤ 0xBFFF FFFF

2.2.2 DSP Subsystem Memory Space Mapping

See Section 2.2.2 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM* version D (SWPU064D).

2.2.3 L3 and L4 Interconnect Registers Memory Mapping

See Section 2.2.3 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM* version D (SWPU064D).

2.2.4 IVA Memory Space Mapping

See Section 2.2.4 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM* version D (SWPU064D).

Chapter 3

MPU Subsystem

This chapter describes the features, integration, and functions of the microprocessor unit (MPU) subsystem of the OMAP2423 multimedia system-in-package (SIP) processor. The chapter includes a programming model and register summary.

This chapter is equivalent to Chapter 3 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 4

DSP Subsystem

This chapter describes the features and integration of the DSP subsystem in the OMAP2423 multimedia system-in-package (SIP) processor and provides a programming model of the subsystem.

This chapter is equivalent to Chapter 4 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 5

Power Reset and Clock Management

This chapter describes the power, reset, and clock management (PRCM) module in the OMAP2423 multimedia system-in-package (SIP) processor.

Topic **Page**

5.1 Power Reset and Clock Manager Overview 5-2
5.2 Power Reset and Clock Manager Environment 5-2
5.3 Power Reset and Clock Manager Integration 5-2
5.4 Clock Manager Functional Description 5-2
5.5 Reset Manager Functional Description 5-8
5.6 Power Management Functional Description 5-8
5.7 PRCM Interrupts 5-8
5.8 Sleep and Wake-Up Processes 5-8
5.9 Emulation Support 5-8
5.10 Programming Model 5-8
5.11 Power Reset and Clock Manager Registers 5-8
5.12 Register Descriptions 5-8

5.1 Power Reset and Clock Manager Overview

See Section 5.1 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.2 Power Reset and Clock Manager Environment

See Section 5.2 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.3 Power Reset and Clock Manager Integration

See Section 5.3 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.4 Clock Manager Functional Description

The PRCM provides an efficient clock distribution associated with an aggressive idle scheme. This allows reducing dynamic consumption and offers a simplified and unified clock software control.

5.4.1 Clock Generation

See Section 5.4.1 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.4.2 Clock Distribution

See Section 5.4.2 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.4.3 Clock Configurations

Clock configurations are dependent of the core voltage and maximum clock frequencies may be not applicable to production. Please contact your TI representative to get the Data Manual performance addendum including validated values for production.

5.4.3.1 Scalable Clocks

Because of the frequency limitations of the stacked mobile DDR SDRAM, the OMAP2423 device can support four different clock configurations:

- ❑ Regular configurations

- Optimized for 133-MHz SDRAM speed grade. The DPLL max output clock is 532 MHz, the core L3 interconnect clock is 133 MHz, and the MPU clock is 266 MHz.
- 133-MHz-SDRAM speed grade and max MPU frequency. The DPLL max output clock is 660 MHz, the core L3 interconnect clock is 110 MHz, and the MPU clock is 330 MHz.

❑ Low frequency and boot configurations

- Boot clock. The DPLL clock, the core L3 interconnect clock, and the MPU clock frequency is a system clock frequency (12, 13, or 19.2 MHz).

Note:

The ROM code changes the DPLL output clock settings to 24 MHz. See Section 34.4.2.2, *Public and Secure Initialization*, for details.

- 32-kHz frequency configuration. The DPLL clock is 32 kHz, the core L3 interconnect clock is 32 kHz, and the MPU clock is 32 kHz.

No other configuration is sure to work. It is the responsibility of the software to set the correct clock configuration. Although the two stacked memories support different frequencies (166-MHz for the 512M-bit DDR and 133 MHz for the 256M-bit DDR), the OMAP2423 device must be configured with the minimum of the two speed grades, that is, the 133-MHz SDRAM speed grade.

Table 6−13 through Table 6−17 provide detailed views of the different configurations, including all of the modules supplied from the DPLL output. For each PRCM clock signal output, the frequency is indicated as well as the ratio with regards to the DPLL frequency.

The following legend applies to the tables:

❑ A low-power mode is described for each one of the six first configurations: clock ratios are unchanged, but the DPLL output clock is set to DPLL_CLKOUT x1 (as opposed to DPLL_CLKOUT x2 in normal mode). These low-power modes allow scaling down the chip voltage to 1.05 V, whereas 1.3 V is required in normal mode.

❑ A green box indicates that register programming is required and provides the related fields. The Divider/Sync column provides the value to write in the registers.

❑ The MPU subsystem interrupt controller, the DSP subsystem interrupt controller, and the IVA subsystem interrupt controller clocks are not listed. Their frequencies are automatically fixed when the related processors clocks are fixed.

❑ When a synchronizer is required between the interconnect and a subsystem (DSP or IVA), the related column is marked activated.

The DPLL setting requires particular attention. The DPLL_MULT and DPLL_DIV bit fields (respectively, CM_CLKSEL1_PLL[21:12] and CM_CLKSEL1_PLL[12:8]) allow selection of the DPLL frequency.

The default configuration uses the DPLL frequency x2 as the CORE_CLK frequency. The low-power mode uses CORE_CLK in its DPLL frequency x1 configuration, set via the CORE_CLK_SRC bit field in the CM_CLKSEL2_PLL register (CM_CLKSEL2_PLL[1:0]). For DPLL programming information, see Section 5.10.15.

Table 6−13. Regular OMAP2423 Clock Configuration – SDRAM 133 MHz

Configuration III SDRAM Speed Grade: 133 MHz						
DSP Synchronizer activated – IVA Synchronizer Activated						
			Normal		Low Power	
Module	**Clocks − Sync**	**Control Bits**	**Divider /Sync**	**Frequency (MHz)**	**Divider /Sync**	**Frequency (MHz)**
DPLL	DPLL settings	CM_CLKSEL1_PLL [21:8]	M, N	266	M, N	266
	CORE_CLK	CM_CLKSEL2_PLL[1:0]	DPLL x2	532	DPLL	266
MPU subsystem	MPU_FCLK	CM_CLKSEL_MPU[4:0]	2	266	2	133
DSP subsystem	DSP_FCLK	CM_CLKSEL_DSP[4:0]	3	177.3	3	88.67
	DSP_ICLK	CM_CLKSEL_DSP[6:5]	6	88.67	6	44.33
	Synchronizer	CM_CLKSEL_DSP[7]	activated	\	activated	\
IVA subsystem	IVA_FCLK	CM_CLKSEL_DSP[12:8]	3	177.3	3	88.67
	IVA_ICLK		auto	88.67	auto	44.33
	IVA_MPU_CLK		auto	88.67	auto	44.33
	Synchronizer	CM_CLKSEL_DSP[13]	activated	\	activated	\
GFX subsystem	GFX_FCLK	CM_CLKSEL_GFX[2:0]	8	66.5	8	33.25
SSI	SSI_SSR_FCLK	CM_CLKSEL1_CORE [24:20]	3	177.3	3	88.67
	SSI_SST_FCLK		auto	88.67	auto	44.33
L3 interconnect, interface clocks	CORE_L3_ICLK	CM_CLKSEL1_CORE [4:0]	4	133	4	66.5
	MPU_ICLK					

Table 6−13. Regular OMAP2423 Clock Configuration – SDRAM 133 MHz (Continued)

Configuration III SDRAM Speed Grade: 133 MHz						
DSP Synchronizer activated – IVA Synchronizer Activated						
			Normal		**Low Power**	
Module	**Clocks − Sync**	**Control Bits**	**Divider /Sync**	**Fre-quency (MHz)**	**Divider /Sync**	**Fre-quency (MHz)**
	GFX_ICLK					
	DSS_L3_CLK					
L4 interconnect, interface clocks	CORE_L4_ICLK	CM_CLKSEL1_CORE [6:5]	8	66.5	8	33.25
	WU_L4_ICLK					
	SSI_ICLK					
	DSS_L4_ICLK					
	CORE_L4_USB_ CLK	CM_CLKSEL1_CORE [27:25]	16	33.25	16	16.63

Table 6−14. Regular OMAP2423 Clock Configuration – SDRAM 133 MHz (Max MPU)

Configuration IV SDRAM Speed Grade: 133 MHz − Max MPU Speed						
DSP Synchronizer Bypassed – IVA Synchronizer Bypassed						
			Normal		**Low Power**	
Module	**Clocks − Sync**	**Control Bits**	**Divider /Sync**	**Fre-quency (MHz)**	**Divider /Sync**	**Fre-quency (MHz)**
DPLL	DPLL settings	CM_CLKSEL1_PLL [21:8]	M, N	330	M, N	330
	CORE_CLK	CM_CLKSEL2_PLL[1:0]	DPLL x2	660	DPLL	330
MPU subsystem	MPU_FCLK	CM_CLKSEL_MPU[4:0]	2	330	2	165
DSP subsystem	DSP_FCLK	CM_CLKSEL_DSP[4:0]	3	220	3	110
	DSP_ICLK	CM_CLKSEL_DSP[6:5]	6	110	6	55
	Synchronizer	CM_CLKSEL_DSP[7]	by-passed	\	by-passed	\
IVA subsystem	IVA_FCLK	CM_CLKSEL_DSP[12:8]	6	110	6	55
	IVA_ICLK		auto	55	auto	27.5
	IVA_MPU_CLK		auto	55	auto	27.5
	Synchronizer	CM_CLKSEL_DSP[13]	by-passed	\	by-passed	\
GFX subsystem	GFX_FCLK	CM_CLKSEL_GFX[2:0]	12	55	12	27.5

Table 6−14. Regular OMAP2423 Clock Configuration – SDRAM 133 MHz (Max MPU) (Continued)

Configuration IV SDRAM Speed Grade: 133 MHz − Max MPU Speed						
DSP Synchronizer Bypassed – IVA Synchronizer Bypassed						
			Normal		Low Power	
Module	**Clocks − Sync**	**Control Bits**	**Divider /Sync**	**Frequency (MHz)**	**Divider /Sync**	**Frequency (MHz)**
SSI	SSI_SSR_FCLK	CM_CLKSEL1_CORE [24:20]	3	220	3	110
	SSI_SST_FCLK		auto	110	auto	55
L3 interconnect, interface clocks	CORE_L3_ICLK	CM_CLKSEL1_CORE [4:0]	6	110	6	55
	MPU_ICLK					
	GFX_ICLK					
	DSS_L3_CLK					
L4 interconnect, interface clocks	CORE_L4_ICLK	CM_CLKSEL1_CORE [6:5]	12	55	12	27.5
	WU_L4_ICLK					
	SSI_ICLK					
	DSS_L4_ICLK					
	CORE_L4_USB_CLK	CM_CLKSEL1_CORE [27:25]	12	55	12	55

Table 6−17. Low Frequency and Boot OMAP2423 Clock Configurations

Configuration VII and VIII Miscellaneous (Boot and 32 kHz)						
DSP Synchronizer Bypassed – IVA Synchronizer Bypassed						
			Normal		Low Power	
Module	**Clocks − Sync**	**Control Bits**	**Divider /Sync**	**Frequency (MHz)**	**Divider /Sync**	**Frequency (kHz)**
DPLL	DPLL settings	CM_CLKSEL1_PLL [21:8]	M, N	19.2	M, N	32
	CORE_CLK	CM_CLKSEL2_PLL[1:0]	DPLL	19.2	DPLL	32
MPU subsystem	MPU_FCLK	CM_CLKSEL_MPU[4:0]	1	19.2	1	32
DSP subsystem	DSP_FCLK	CM_CLKSEL_DSP[4:0]	N/A	0	N/A	0
	DSP_ICLK	CM_CLKSEL_DSP[6:5]	N/A	0	N/A	0
	Synchronizer	CM_CLKSEL_DSP[7]	bypassed	\	bypassed	\

Table 6−17. Low Frequency and Boot OMAP2423 Clock Configurations (Continued)

Configuration VII and VIII Miscellaneous (Boot and 32 kHz)						
DSP Synchronizer Bypassed – IVA Synchronizer Bypassed						
			Normal		**Low Power**	
Module	**Clocks − Sync**	**Control Bits**	**Divider /Sync**	**Fre-quency (MHz)**	**Divider /Sync**	**Fre-quency (kHz)**
IVA subsystem	IVA_FCLK	CM_CLKSEL_DSP[12:8]	N/A	0	N/A	0
	IVA_ICLK		N/A	0	N/A	0
	IVA_MPU_CLK		N/A	0	N/A	0
	Synchronizer	CM_CLKSEL_DSP[13]	by-passed	\	by-passed	\
GFX subsystem	GFX_FCLK	CM_CLKSEL_GFX[2:0]	N/A	0	N/A	0
SSI	SSI_SSR_FCLK	CM_CLKSEL1_CORE [24:20]	1	19.2	N/A	0
	SSI_SST_FCLK		auto	19.2	N/A	0
L3 interconnect, interface clocks	CORE_L3_ICLK	CM_CLKSEL1_CORE [4:0]	1	19.2	1	32
	MPU_ICLK					
	GFX_ICLK					
	DSS_L3_CLK					
L4 interconnect, interface clocks	CORE_L4_ICLK	CM_CLKSEL1_CORE [6:5]	1	19.2	1	32
	WU_L4_ICLK					
	SSI_ICLK					
	DSS_L4_ICLK					
	CORE_L4_USB_CLK	CM_CLKSEL1_CORE [27:25]	1	19.2	0	N/A

The DPLL clock frequency can be 12, 13, or 19.2 MHz in boot configuration (Configuration VII).

5.4.3.2 Peripheral Clocks

See Section 5.4.3.2 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.5 Reset Manager Functional Description

See Section 5.5 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.6 Power Management Functional Description

See Section 5.6 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.7 PRCM Interrupts

See Section 5.7 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.8 Sleep and Wake-Up Processes

See Section 5.8 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.9 Emulation Support

See Section 5.9 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.10 Programming Model

See Section 5.10 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.11 Power Reset and Clock Management Registers

See Section 5.11 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

5.12 Register Descriptions

See Section 5.12 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 6

Internal Interconnect

This chapter describes the OMAP2423 multimedia system-in-package (SIP) processor internal interconnect.

This chapter is equivalent to Chapter 6 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 7

Interprocessor Communication

This chapter discusses the interprocessor communication (IPC) between the on-chip processors of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 7 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 8

System Control Module

This chapter describes the system control module on the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 8 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 9

Memory Management Units

This chapter describes the MMU devices of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 9 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 10

DMA

This chapter describes the generic DMA module used in the DSP subsystem (DDMA), camera subsystem (CamDMA), and system DMA (SDMA) of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 10 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 11

Interrupt Controller

This chapter describes the interrupt controller (INTC) module used in the MPU, DSP, and IVA subsystems of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 11 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 12

Memory Subsystem

This chapter describes the memory subsytem of the OMAP2423 multimedia system-in-package (SIP) device, including the general-purpose memory controller (GPMC), stacked DDR controller, and on-chip memory (OCM) subsystem.

Topic **Page**

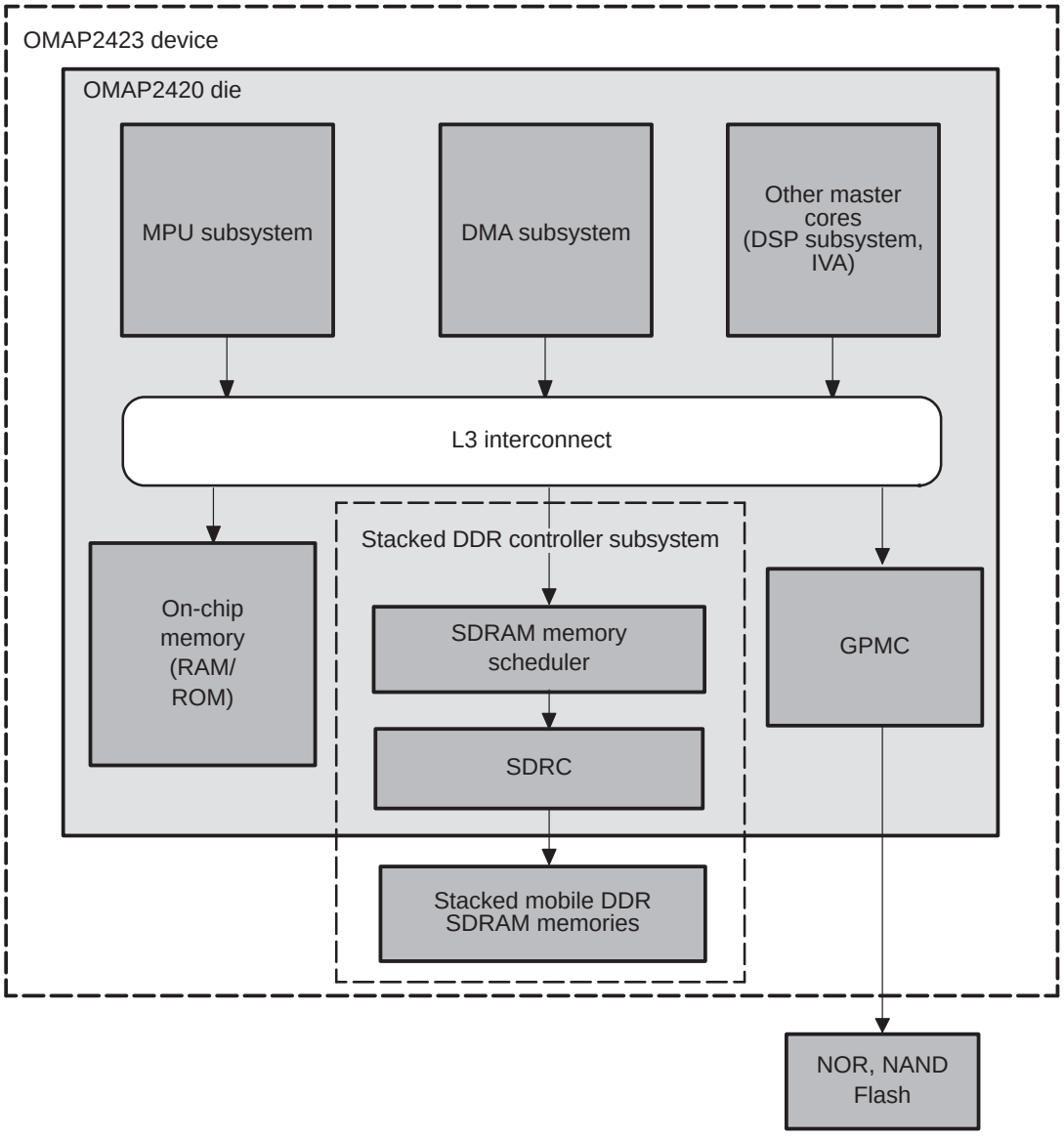
12.1 General-Purpose Memory Controller Overview 12-2
12.2 General-Purpose Memory Controller Environment 12-4
12.3 General-Purpose Memory Controller Integration 12-9
12.4 General-Purpose Memory Controller Functional Description ... 12-14
12.5 General-Purpose Memory Controller Programming Model 12-14
12.6 General-Purpose Memory Controller Registers 12-14
12.7 Stacked DDR Controller Subsystem Overview 12-15
12.8 Stacked DDR Controller Subsystem Environment 12-19
12.9 Stacked DDR Controller Subsystem Integration 12-21
12.10 Stacked DDR Controller Subsystem Functional Description 12-30
12.11 Stacked DDR Controller Subsystem Programming Model 12-45
12.12 Stacked DDR Controller Subsystem Registers 12-64
12.13 OCM Subsystem Overview 12-108
12.14 OCM Subsystem Integration 12-108
12.15 OCM Subsystem Functional Description 12-108

12.1 General-Purpose Memory Controller Overview

The general-purpose memory controller (GPMC) is the OMAP2423 unified memory controller dedicated to interfacing external memory devices such as:

- Asynchronous SRAM-like memories and ASIC devices
- Asynchronous, page mode, and synchronous burst NOR flash
- NAND flash

Figure 12−1. GPMC Environment



12.1.1 Feature List

The GPMC is the OMAP2423 16-bit external memory controller. The GPMC data access engine provides a flexible programming model to interface all known standard memories. The GPMC supports various accesses:

- Asynchronous read/write access
- Asynchronous read page access (4−8−16 Word16)
- Synchronous read access
- Synchronous read burst access without wrap capability (4−8−16 Word16)
- Synchronous read burst access with wrap capability (4−8−16 Word16)
- Synchronous write burst access
- Address and data multiplexed access

The GPMC can be interfaced with a wide range of external devices:

- Asynchronous or synchronous 8-bit-wide memory or ASIC device (without page/burst support)
- Asynchronous or synchronous 16-bit-wide memory or ASIC device (with page/burst support)
- Asynchronous or synchronous 16-bit-wide address and data multiplexed memory or ASIC device (with page/burst support)
- 8-bit and 16-bit NAND flash devices

The GPMC supports up to 8 chip-select regions of programmable size and base addresses (from 16M bytes to 128M bytes) in a total address space of 1G byte.

12.2 General-Purpose Memory Controller Environment

Figure 12−2 through Figure 12−4 depict different GPMC external connection options:

- GPMC to 16-bit non-multiplexed memory
- GPMC to 16-bit address/data multiplexed memory
- GPMC to 16-bit NAND device

Figure 12−2 illustrates connections between the GPMC module and a 16-bit non-multiplexed external device/memory.

Figure 12−2. GPMC to 16-Bit Non-Multiplexed Memory

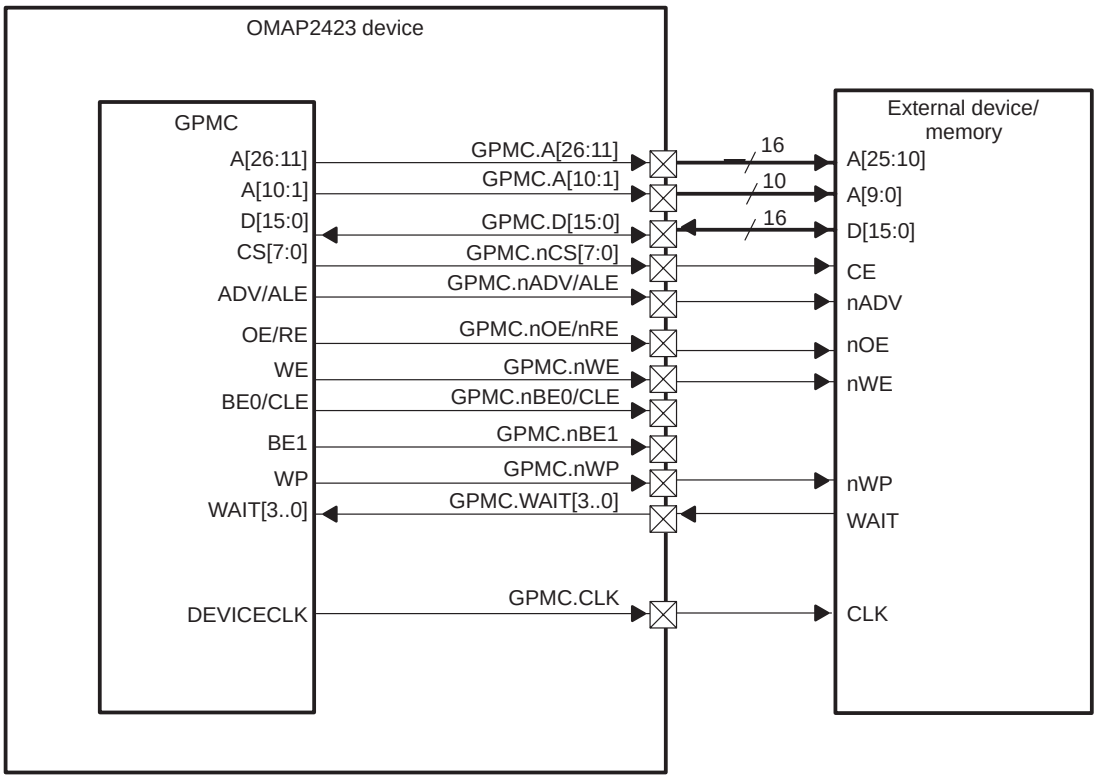
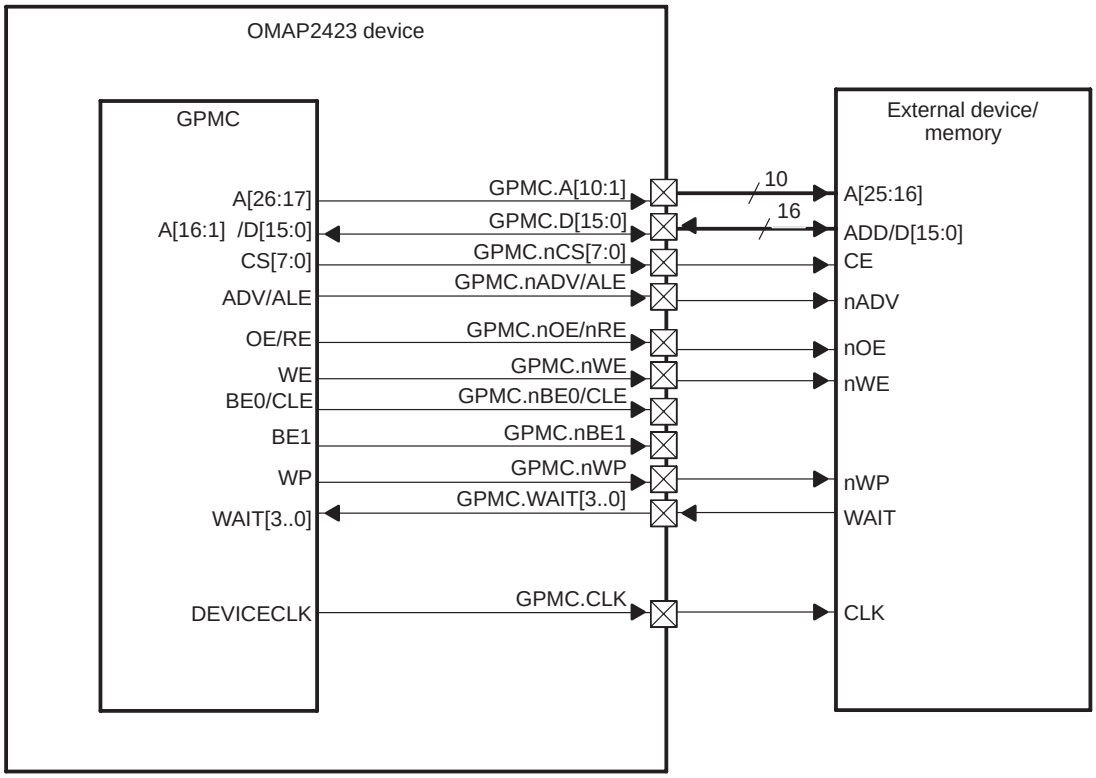


Figure 12−3 illustrates a connection between the GPMC module and a 16-bit address/data multiplexed external memory device.

Figure 12−3. GPMC to 16-Bit Address/Data Multiplexed Memory

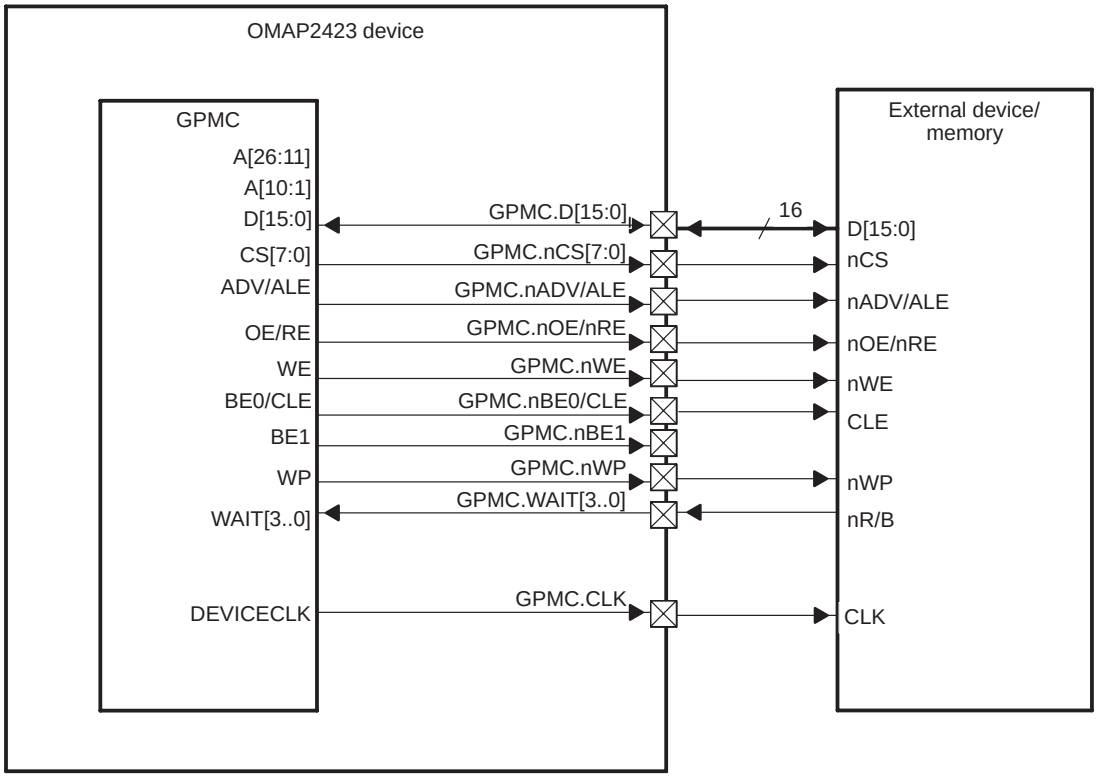


Note:

An 8-bit device must be connected to D[7:0] data bus of the GPMC. The OMAP2423 device does not provide the A0 byte address line required for random-byte-addressable, 8-bit-wide device interfacing (for both multiplexed and non-multiplexed protocol). This limits the use of 8-bit-wide device interfacing to byte alias access.

Figure 12−4 illustrates a connection between the GPMC and a 16-bit NAND device.

Figure 12−4. GPMC to 16-Bit NAND Device



Note:

An 8-bit NAND must be connected to D[7:0] data bus of the GPMC.

All GPMC controller subsystem I/O pins are listed in Table 13−1.

Table 13−1. GPMC I/O Description

Signal Name	I/O	Description
GPMC.A[26:11]	O	Address bus: A[26:11] in non-multiplexed mode
GPMC.A[10:1]	O	Address bus: A[26:17] in add/data multiplexed mode A[10:1] in non-multiplexed mode
GPMC.D[15:0]	I/O	Data bus: A[16:1]/D[15:0] in add data/multiplexed mode D[15:0] in non-multiplexed mode
GPMC.nCS[7:0]	O	Chip-selects (active low) Note: nCS[7:4]
GPMC.CLK	I/O	External clock
GPMC.nADV/ALE	O	Address valid (active low) if NOR protocol memory is selected Address latch enable (active high) if NAND protocol memory is selected
GPMC.nOE/nRE	O	Output enable (active low) if NOR protocol memory is selected Read enable (active low) if NAND protocol memory is selected

Table 13−1. GPMC I/O Description (Continued)

Signal Name	I/O	Description
GPMC.nWE	O	Write enable (active low)
GPMC.nBE0/CLE	O	Lower byte enable (active low) Command latch enable (active high) if NAND protocol memory is selected
GPMC.nBE1	O	Upper byte enable (active low)
GPMC.nWP	O	Write protect (active low)
GPMC.WAIT[3:0]	I	External wait signal for NOR and NAND protocol memories
GPMC.IO_DIR	O	Data bus signal direction control

Table 13−2 shows the usage of address and data SDRAM controller pins according to the connected external device.

Table 13−2. GPMC Pin Multiplexing Options

GPMC Pin	Non-multiplexed Address Data 16-Bit Device	Multiplexed Address Data 16-Bit Device	Non-multiplexed Address Data 16-Bit Device With LIMITEDADDRESS Bit Enabled	16-Bit NAND Device	8-Bit NAND Device
GPMC.A[26]	A25	Not used	Not used	Not used	Not used
GPMC.A[25]	A24	Not used	Not used	Not used	Not used
GPMC.A[24]	A23	Not used	Not used	Not used	Not used
GPMC.A[23]	A22	Not used	Not used	Not used	Not used
GPMC.A[22]	A21	Not used	Not used	Not used	Not used
GPMC.A[21]	A20	Not used	Not used	Not used	Not used
GPMC.A[20]	A19	Not used	Not used	Not used	Not used
GPMC.A[19]	A18	Not used	Not used	Not used	Not used
GPMC.A[18]	A17	Not used	Not used	Not used	Not used
GPMC.A[17]	A16	Not used	Not used	Not used	Not used
GPMC.A[16]	A15	Not used	Not used	Not used	Not used
GPMC.A[15]	A14	Not used	Not used	Not used	Not used
GPMC.A[14]	A13	Not used	Not used	Not used	Not used
GPMC.A[13]	A12	Not used	Not used	Not used	Not used
GPMC.A[12]	A11	Not used	Not used	Not used	Not used
GPMC.A[11]	A10	Not used	Not used	Not used	Not used
GPMC.A[10]	A9	A25	A9	Not used	Not used
GPMC.A[9]	A8	A24	A8	Not used	Not used
GPMC.A[8]	A7	A23	A7	Not used	Not used

Table 13−2. GPMC Pin Multiplexing Options (Continued)

GPMC Pin	Non-multiplexed Address Data 16-Bit Device	Multiplexed Address Data 16-Bit Device	Non-multiplexed Address Data 16-Bit Device With LIMITEDADDRESS Bit Enabled	16-Bit NAND Device	8-Bit NAND Device
GPMC.A[7]	A6	A22	A6	Not used	Not used
GPMC.A[6]	A5	A21	A5	Not used	Not used
GPMC.A[5]	A4	A20	A4	Not used	Not used
GPMC.A[4]	A3	A19	A3	Not used	Not used
GPMC.A[3]	A2	A18	A2	Not used	Not used
GPMC.A[2]	A1	A17	A1	Not used	Not used
GPMC.A[1]	A0	A16	A0	Not used	Not used
GPMC.D[15]	D15	A15/D15	D15	D15	Not used
GPMC.D[14]	D14	A14/D14	D14	D14	Not used
GPMC.D[13]	D13	A13/D13	D13	D13	Not used
GPMC.D[12]	D12	A12/D12	D12	D12	Not used
GPMC.D[11]	D11	A11/D11	D11	D11	Not used
GPMC.D[10]	D10	A10/D10	D10	D10	Not used
GPMC.D[9]	D9	A9/D9	D9	D9	Not used
GPMC.D[8]	D8	A8/D8	D8	D8	Not used
GPMC.D[7]	D7	A7/D7	D7	D7	D7
GPMC.D[6]	D6	A6/D6	D6	D6	D6
GPMC.D[5]	D5	A5/D5	D5	D5	D5
GPMC.D[4}	D4	A4/D4	D4	D4	D4
GPMC.D[3]	D3	A3/D3	D3	D3	D3
GPMC.D[2]	D2	A2/D2	D2	D2	D2
GPMC.D[1]	D1	A1/D1	D1	D1	D1
GPMC.D[0]	D0	A0/D0	D0	D0	D0

Note:

The OMAP2423 device does not provide the A0 byte address line required for random byte addressable 8-bit-wide device interfacing (for both multiplexed and non-multiplexed protocols). This limits the use of 8-bit-wide device interfacing to byte alias access.

12.3 General-Purpose Memory Controller Integration

12.3.1 Description

See Section 12.3.1 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

12.3.2 Clocking, Reset, and Power Management Scheme

See Section 12.3.2 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

12.3.3 GPMC I/O Configuration

Static multiplexing options exist for the GPMC depending on attached memories, application pin-count requirements, and interface configurations required. If a configuration does not require all GPMC pins, GPIOs or alternate function signals can also be multiplexed (see Table 12−4 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D)).

12.3.3.1 GPMC Address and Data Bus Usage

Depending on the GPMC configuration on each chip-select, the address and data bus lines that are not required for a particular access protocol are not updated (changed from current value) and are not sampled when input (input data bus).

- When GPMC_CONFIG.LIMITEDADDRESS = 1, non-multiplexed address/data NOR devices do not use the GPMC I/O: GPMC.A[26:11]. Only GPMC.A[10:0] address lines are used. This limits the memory support to only 2K-byte addressable memories.
- Multiplexed address and data NOR devices do not use GPMC I/O: GPMC.A[26:11]. The address is multiplexed on the data bus.
- 8-bit-wide NOR devices do not use GPMC I/O: GPMC.D[15:8].
- 16-bit-wide NAND devices do not use the GPMC I/O: GPMC.A[26:11] and GPMC.A [10:1].
- 8-bit-wide NAND devices do not use the GPMC I/O: GPMC.A[26:11], GPMC I/O: GPMC.A[10:1], or GPMC I/O: GPMC.D[15:8].

The OMAP2423 power-on reset (IC reset) sets the address and data lines to the default logical one value. This default value is a transparent value that allows OMAP device pin sharing (transparent multiplexing) between GPMC and alternate I/O functions. The alternate functions are selected instead of the GPMC function (assuming the GPMC function is not used). For example, alternate I/O functions can use [D15−D8] if GPMC use is restricted to 8-bit-wide device interfacing only.

You should never configure a chip-select or make any access with an incompatible sharing configuration that can corrupt the transparent reset value.

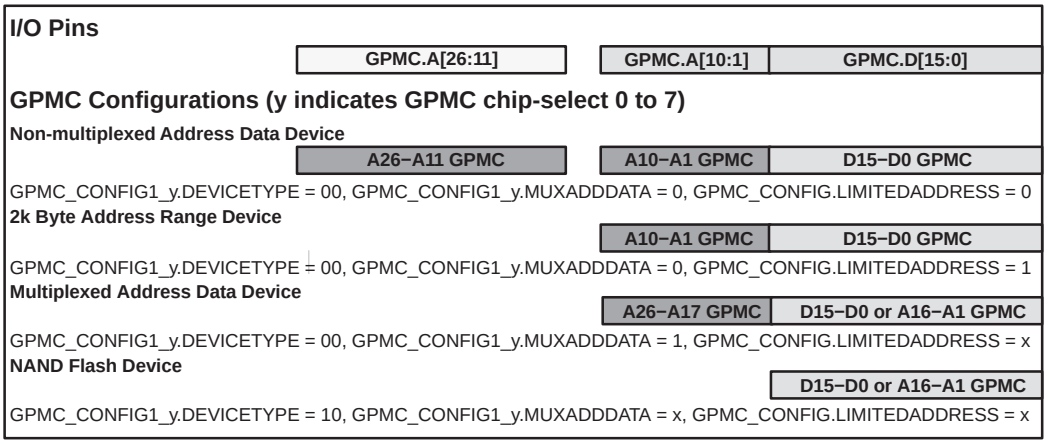
12.3.3.2 GPMC I/O Configuration Solutions

Figure 12−6 shows the different solutions for connecting SDRAM-like devices, NOR flash devices, asynchronous/synchronous devices, and NAND flash devices.

All these memories are connected via several chip-selects (denoted n and y) and share three I/O groups:

- GPMC.A[26:11]
  - Non-multiplexed address/data device (address [26:11]) when GPMC_CONFIG.LIMITEDADDRESS is not set.
- GPMC.A[10:1]
  - Non-multiplexed address/data device (address [10:1])
  - Multiplexed address/data device (address [26:17])
- GPMC.D[15:0]
  - Non-multiplexed address/data device (data [15:0])
  - Multiplexed address/data device (address [16:1] and data [15:0])

Figure 12−6. GPMC I/O Configuration Options in Default Pinout Mode 0



The OMAP2423 device supports 32-bit-wide external DDR/SDRAM interface only when the GPMC usage is restricted to address and data multiplexed protocol (random synchronous or asynchronous memory and NAND device type). These protocols leave free the GPMC.A[26:11] address lines.

12.3.3.3 GPMC I/O Configuration Settings (in Default Pinout Mode 0)

In this section, the symbol *y* signifies chip-select value (CSy).

The non-multiplexed address/data device, if not limited to 2K-byte address range, is selected by programming the following register fields:

- ❏ SDRC register SDRC_SHARING[11:9] (CS0MUXCFG) = 010
- ❏ SDRC register SDRC_SHARING[14:12] (CS1MUXCFG) = 010
- ❏ GMPC register GPMC_CONFIG1_y[11:10] (CSy DEVICETYPE) = 00
- ❏ GPMC_CONFIG1_y[9] (CSy MUXADDDATA) = 0
- ❏ GPMC_CONFIG[1] (LIMITEDADDRESS) = 0

GPMC_CONFIG1_y.DEVICETYPE = 0, GPMC_CONFIG1_y.MUXADDATA = 0 with 32-bit external DDR, and GPMC_CONFIG.LIMITEDADDRESS = 0 are not supported.

The non-multiplexed address/data device, if limited to 2K-byte address range, is selected by programming the following register fields:

- ❏ GMPC register GPMC_CONFIG1_y[11:10] (CSy DEVICETYPE) = 00
- ❏ GPMC_CONFIG1_y[9] (CSy MUXADDDATA) = 0
- ❏ GPMC_CONFIG[1] (LIMITEDADDRESS) = 1

Note:

The LIMITEDADDRESS field applies only to non-multiplexed address/data devices and has no effect on other device types (multiplexed address/data, NAND).

The multiplexed address/data device is selected by programming the following register fields:

- ❏ GMPC register GPMC_CONFIG1_y[11:10] (CSy DEVICETYPE) = 00
- ❏ GPMC_CONFIG1_y[9] (CSy MUXADDDATA) = 1

The NAND device is selected by programming the following register field:

- ❏ GMPC register GPMC_CONFIG1_y[11:10] (CSy DEVICETYPE) = 10

Note:

The CSy MUXADDDATA field applies only to DEVICETYPE = 00 and has no effect on NAND-like devices.

12.3.3.4 GPMC CS0 Default Configuration at IC Reset

To ensure correct external boot with a GPMC access from IC reset time on CS0, three external pins (SYS.BOOT[2:0]) are sampled to configure three GPMC register field reset values of GPMC_CONFIG1_0 register:

- GPMC_CONFIG1_0[22]: WAITREADMONITORING bit reset value
- GPMC_CONFIG1_0[13:12]: DEVICESIZE bits reset value
- GPMC_CONFIG1_0[9]: MUXADDDATA bit reset value

The SYS.BOOT[3] external pin determines internal or external boot.

Table 13−3. GPMC CS0 Default Configuration at IC Reset

SYS.BOOT[3:0]	Internal/ External Boot Code	Wait Monitoring†	Memory Interface Type†	GPMC_ CONFIG1_0_ WAITREADMONITORING Reset Value	GPMC_ CONFIG1_0_ DEVICESIZE Reset Value	GPMC_ CONFIG1_0_ MUXADDDATA Reset Value
0000	External‡	Active low	Non-multiplexed 16-bit	0x1	0x1	0x0
0001	External‡	Don't care	Non-multiplexed 16-bit	0x0	0x1	0x0
0010	External‡	Active low	Address/Data multiplexed 16-bit	0x1	0x1	0x1
0011	External‡	Don't care	Address/Data multiplexed 16-bit	0x0	0x1	0x1
1000	Internal	Active low	Non-multiplexed 16-bit	0x1	0x1	0x0
1001	Internal	Don't care	Non-multiplexed 16-bit	0x0	0x1	0x0
1010	Internal	Active low	Address/Data multiplexed 16-bit	0x1	0x1	0x1
1011	Internal	Don't care	Address/Data multiplexed 16-bit	0x0	0x1	0x1
1100	Internal	−	NAND 8-bit§	0x0	0x0	0x0
1101	Internal	−	NAND *16-bit§	0x0	0x1	0x0

† External boot memory must connect to GPMC.
‡ External direct boot mode only supported on emulation devices
§ DEVICETYPE is always 0x0 at reset and is programmed by internal ROM code to NAND (0x2) if necessary.

CAUTION

With internal boot code, the CS0 configuration can be modified before the first CS0 access. This modification into internal boot code is necessary for:

- **NAND devices attached to CS0**
- **Non-multiplexed 2K-byte address range device used to free GPMC[26:11] I/O and attached to CS0.**

12.3.4 Pin List and Pad Multiplexing With Other Functions

See Section 12.3.4 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

12.3.5 GPMC Register Summary

See Section 12.3.5 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

12.4 General-Purpose Memory Controller Functional Description

See Section 12.4 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

12.5 General-Purpose Memory Controller Programming Model

See Section 12.5 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

12.6 General-Purpose Memory Controller Registers

See Section 12.6 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

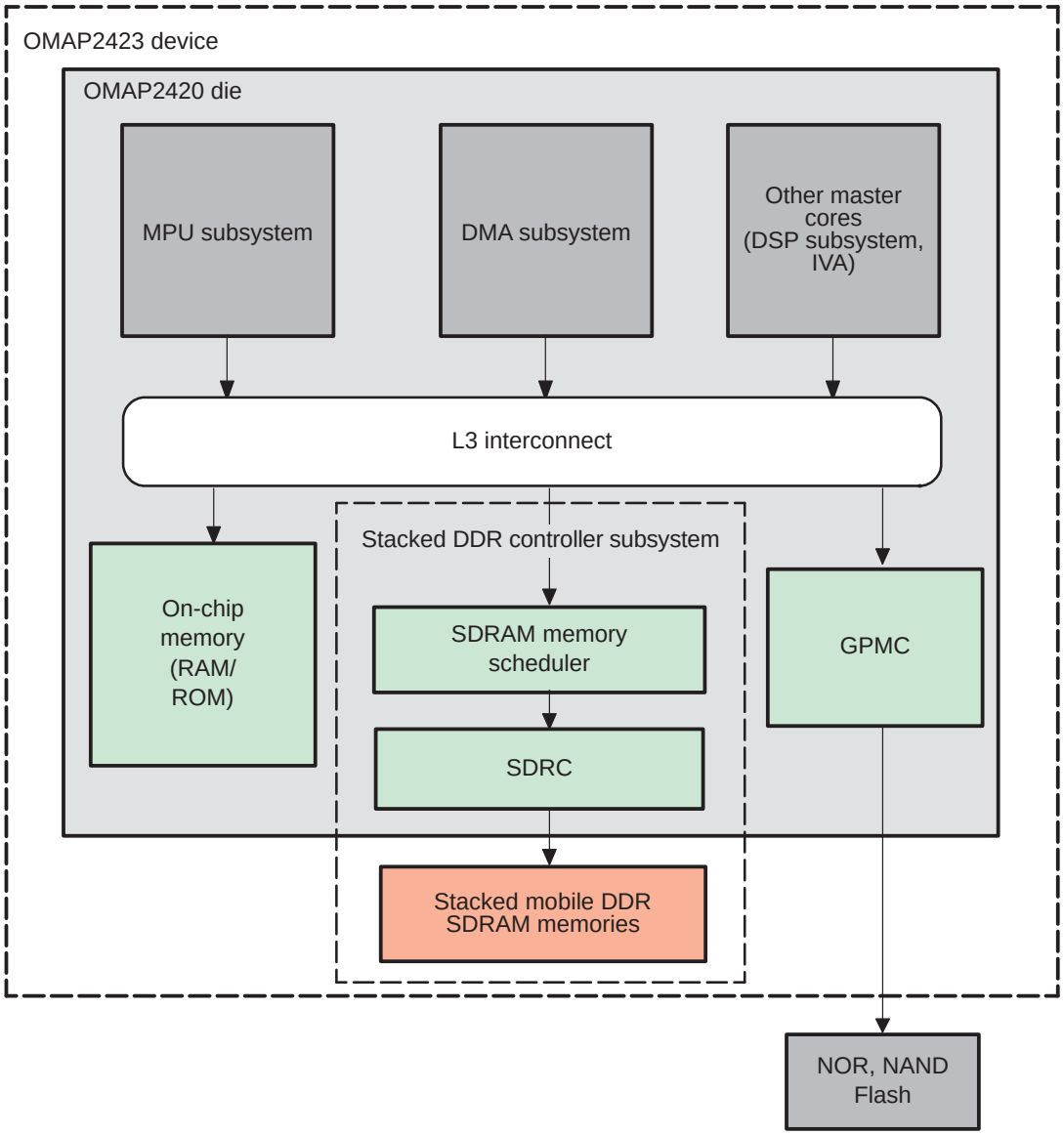
12.7 Stacked DDR Controller Subsystem Overview

The stacked DDR controller subsystem provides connectivity between the OMAP2420 application processor and the stacked mobile DDR (MDDR) SDRAM memories.

The stacked DDR controller subsystem provides a high performance interface to two embedded mobile DDR RAMs: one 512M-bit x32 memory mapped on SDRC CS0, and one 256M-bit x32 memory mapped on SDRC CS1. The subsystem comprises four submodules:

- The SDRAM memory scheduler (SMS), consisting of the scheduler, security firewall, and virtual rotated frame-buffer modules.
- The SDRAM controller (SDRC)
- The two stacked mobile DDR RAM

Figure 12−27. Stacked DDR Controller Subsystem Overview



12.7.1 Stacked DDR Controller Subsystem Features

The main features of the stacked DDR controller subsystem are:

- Security firewall:
  - Four memory regions
  - Independently programmable read/write access control
  - Independently programmable security control
- Virtual rotated frame-buffer module:
  - Minimizes SDRAM page-miss penalty when accessing rotated (that is, non-sequentially addressed) lines in a frame-buffer
  - Four independent contexts
  - Supports rotations of 0°, 90°, 180°, and 270°
  - Transparent to software applications
- Memory-access scheduler:
  - Optimizes latency and bandwidth usage between initiators
  - Three quality of service classes of initiator (QoS)
  - Eight request FIFO queues divided between the QoS classes
  - Programmable arbitration between and within the QoS classes
- SDRAM controller (SDRC):
  - Supports two independently configurable memory areas with separate chip-selects
  - Supports mobile DDR SDRAM memory
  - Supports 256M-bit and 512M-bit devices
  - Memory device organization:
    - Four-bank support
    - 32-bit data width
  - Fully pipe-lined on SDRAM interface
  - Burst Support
    - System burst support

      Incrementing burst support

      Wrapping burst support for critical word-first cache line refill
    - Memory burst support

      System burst for mobile DDR SDRAM: system burst translated into memory burst size of 4

      Read interrupt by read, write interrupt by write
  - Supports standard and enhanced low-power memory features

- Stacked mobile DDR RAM
  - 1 × 256M-bit mobile DDR RAM, 133 MHz
    - Memory organization

      2,097,152 words × 32 bits × 4 banks
  - 1 x 512M-bit mobile DDR RAM, 166 MHz
    - Memory organization

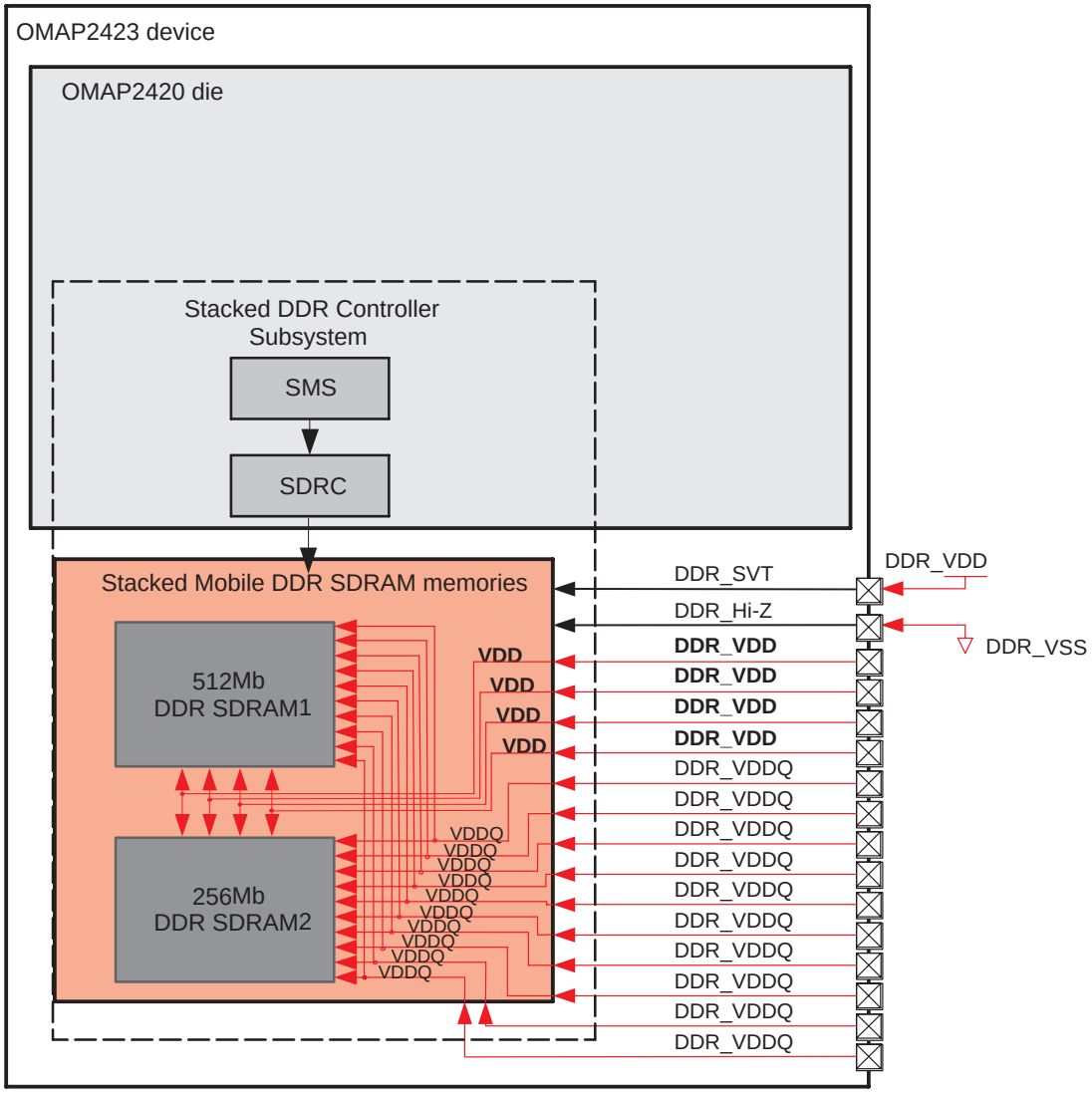
      4,194,304 words × 32 bits × 4 banks
  - Programmable partial array self-refresh
  - Programmable driver strength
  - Autorefresh and self-refresh
  - Auto temperature compensated self-refresh by built-in temperature sensor
- Power-management support

12.8 Stacked DDR Controller Subsystem Environment

12.8.1 Stacked DDR Subsystem Description

Figure 12−28 shows the stacked DDR controller subsystem interface.

Figure 12−28. Stacked DDR Controller Subsystem Connections to External Interface



The OMAP2423 device has two stacked mobile DDR SDRAM memories connected directly to the SDRAM controller; there is no other way to connect external memories on the SDRC interface.

Except for the pins listed on Figure 12−28, the SDRAM controller subsystem has no other interface with the external environment. Pins DDR_Hi-Z and DDR_SVT are used for special memory test features and must be tied to DDR_VSS and DDR_VDD, respectively. See Chapter 29 for details.

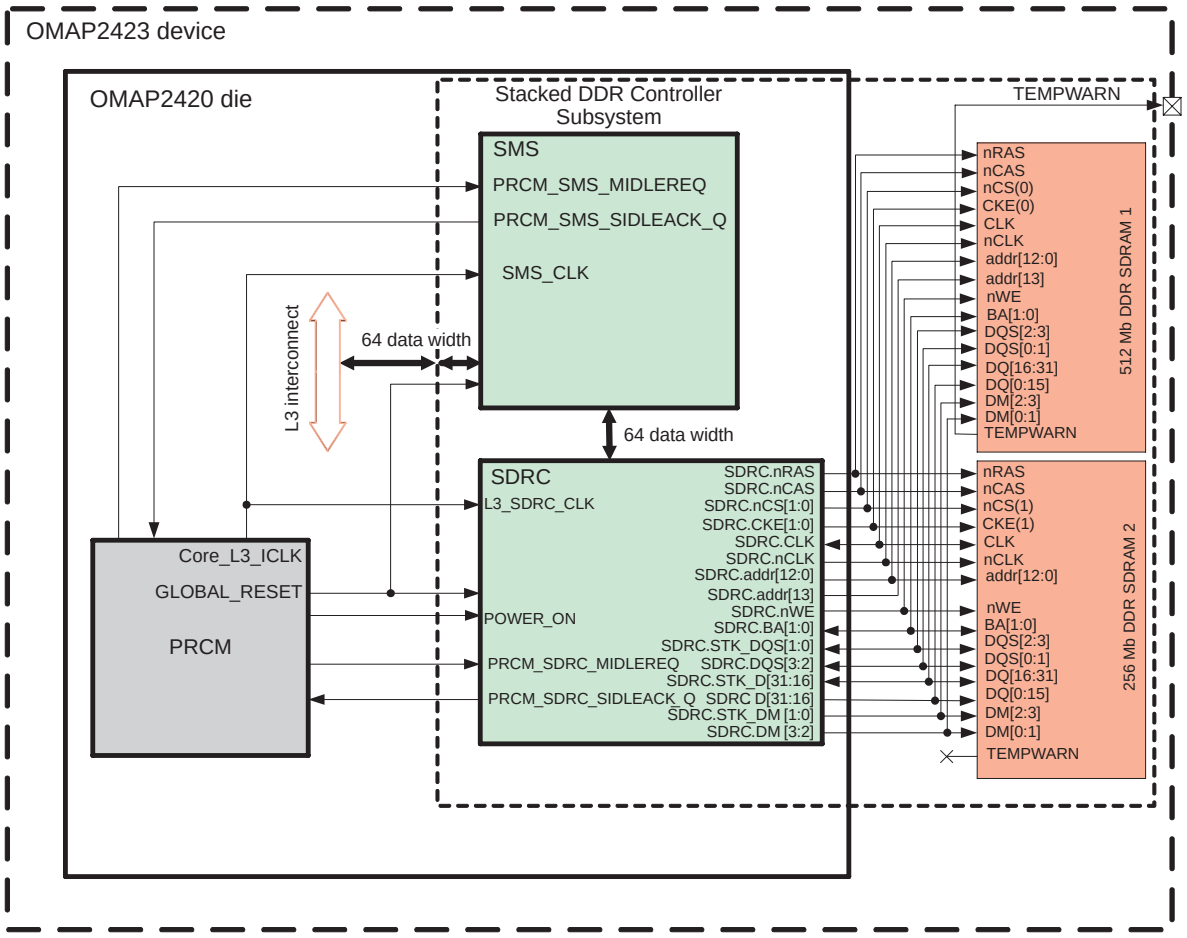
The DDR_VDD and DDR_VDDQ pins are specific to the memory supply. For more information, see the 512Mb × 32 DDR memory specification in Appendix B, the 256Mb × 32 DDR memory specification in Appendix C, and the *OMAP2423 Data Manual*.

12.9 Stacked DDR Controller Subsystem Integration

12.9.1 Description

Figure 12−33 shows how the SMS module, the SDRC module, and the stacked die memories (DDR SDRAM1 and DDR SDRAM 2) are integrated into the OMAP2423 device, how they are connected, and how they interact with the power reset and clock management module (PRCM).

Figure 12−33. Stacked DDR Controller Subsystem Integration to the OMAP2423 Processor



Note:

Because the SDRC and stacked DDR SDRAM have no aligned bit and byte mappings between them, the bit and byte swaps are transparent to software considerations. Bit swap is transparent to software when limited to one byte. Byte swap is transparent to software when the corresponding DM and DQS signals are also swapped.

12.9.2 Clocking, Reset, and Power Management Scheme

12.9.2.1 Clocking

SMS

Clock SMS_CLK comes internally from the PRCM module and runs at the L3 interconnect frequency. This clock is used as both interface and functional clock in the SMS module.

The SMS_CLK source is the PRCM CORE_L3_ICLK output; CORE_L3_ICLK belongs to the L3 interconnect clock domain.

SDRC

Clock L3_SDRC_CLK comes from the PRCM module and runs at the L3 interconnect frequency. This clock is used as both interface and functional clock in the SDRC module.

The L3_SDRC_CLK source is the PRCM CORE_L3_ICLK output; CORE_L3_ICLK belongs to the L3 interconnect clock domain.

At the PRCM level, the SDRC module can be configured to prevent/allow the whole domain to be shut down via the PRCM.CM_AUTOIDLE3_CORE[2] AUTOSDRC bit.

If AUTOSDRC is set 0, the SDRC does not permit the clock domain (CORE_L3_ICLK) to shut down. If set to 1, the SDRC accepts an idle request from PRCM as soon as the software shuts off the domain clock. The PRCM_SDRC_MIDLEREQ signal is then asserted and SDRC sends back PRCM_SDRC_SIDLEACK_Q according its SDRC.SDRC_SYSCONFIG[4:3] IDLE_MODE field setting.

Note:

The domain clock is shut down only if all other modules that receive the clock are able and ready to accept this request.

For details, see Chapter 5.

Stacked DDR SDRAM

The two stacked DDR SDRAM input clocks are the SDRC.CLK and SDRC.nCLK outputs of the SDRC module.

SDRC.CLK and SDRC.nCLK run at the L3 interconnect frequency.

The stacked DDR SDRAM memories operate correctly at a maximum frequency of 133 MHz; this implies that the maximum L3 interconnect frequency must not exceed 133 MHz to prevent functional problems with the stacked DDR SDRAM.

12.9.2.2 Hardware Reset

Global reset of the SDRAM controller occurs by activating the CORE_RST_N signal in the CORE_RST domain (see Chapter 5).

One global reset signal, GLOBAL_RESET, is qualified by the POWER_ON signal. This qualification determines whether the signal is a cold or warm reset:

- On a **cold** reset (that is, a power-on reset, when POWER_ON = 0 and GLOBAL_RESET is applied), all registers and state machines within the SDRC are asynchronously reset.
- On a **warm** reset (that is, any other system reset conditions under control of the chip top-level power manager; POWER_ON = 1 when GLOBAL_RESET is applied), the SDRC registers and FSM are not reset, but the stacked SDRAM memories can be optionally placed in self-refresh mode.

12.9.2.3 Software Reset

The SMS and SDRC modules can be reset under software control via the software reset bits:

- SMS.SMS_SYSCONFIG[1] SOFTRESET
- SDRC.SDRC_SYSCONFIG[1] SOFTRESET

The software reset has the same action as the hardware cold reset, that is, all the registers and the FSM are reset immediately and unconditionally.

12.9.2.4 Power Domain

The core power domain supplies power to the SDRAM controller. For details, see Chapter 5. Section 12.10.2 describes the power-saving features and idle modes.

12.9.3 Interfacing OMAP2420 and Stacked DDR SDRAM

Note:

The TEMPWARN pin has a dedicated output on the OMAP2423 ball out. You must connect TEMPWARN to a GPIO of the OMAP2423 device using the board.

You can use any GPIO to monitor TEMPWARN except GPIO.1, GPIO.2, GPIO.37, and GPIO.38, all of which must not be connected to the board because of internal connections (if connected to the board, they must only be used as debug pins).

Table 13−53 lists the details of the interconnect between the OMAP2420 processor and the stacked memories.

Table 13−53. Stacked DDR Controller Subsystem I/O Description

OMAP2420 Pin	Type	Description	DDR SDRAM1 Pin	Type	DDR SDRAM2 Pin	Type
SDRC.D[31]	I/O	Data I/O	DQ[0]	I/O	DQ[0]	I/O
SDRC.D[30]	I/O	Data I/O	DQ[7]	I/O	DQ[7]	I/O
SDRC.D[29]	I/O	Data I/O	DQ[2]	I/O	DQ[2]	I/O
SDRC.D[28]	I/O	Data I/O	DQ[5]	I/O	DQ[5]	I/O
SDRC.D[27]	I/O	Data I/O	DQ[3]	I/O	DQ[3]	I/O
SDRC.D[26]	I/O	Data I/O	DQ[6]	I/O	DQ[6]	I/O
SDRC.D[25]	I/O	Data I/O	DQ[1]	I/O	DQ[1]	I/O
SDRC.D[24]	I/O	Data I/O	DQ[4]	I/O	DQ[4]	I/O
SDRC.D[23]	I/O	Data I/O	DQ[10]	I/O	DQ[10]	I/O
SDRC.D[22]	I/O	Data I/O	DQ[8]	I/O	DQ[8]	I/O
SDRC.D[21]	I/O	Data I/O	DQ[11]	I/O	DQ[11]	I/O
SDRC.D[20]	I/O	Data I/O	DQ[9]	I/O	DQ[9]	I/O
SDRC.D[19]	I/O	Data I/O	DQ[13]	I/O	DQ[13]	I/O
SDRC.D[18]	I/O	Data I/O	DQ[12]	I/O	DQ[12]	I/O
SDRC.D[17]	I/O	Data I/O	DQ[15]	I/O	DQ[15]	I/O
SDRC.D[16]	I/O	Data I/O	DQ[14]	I/O	DQ[14]	I/O
SDRC.STK_D[31]	I/O	Data I/O	DQ[16]	I/O	DQ[16]	I/O
SDRC.STK_D[30]	I/O	Data I/O	DQ[17]	I/O	DQ[17]	I/O
SDRC.STK_D[29]	I/O	Data I/O	DQ[18]	I/O	DQ[18]	I/O
SDRC.STK_D[28]	I/O	Data I/O	DQ[19]	I/O	DQ[19]	I/O
SDRC.STK_D[27]	I/O	Data I/O	DQ[20]	I/O	DQ[20]	I/O
SDRC.STK_D[26]	I/O	Data I/O	DQ[21]	I/O	DQ[21]	I/O
SDRC.STK_D[25]	I/O	Data I/O	DQ[22]	I/O	DQ[22]	I/O
SDRC.STK_D[24]	I/O	Data I/O	DQ[23]	I/O	DQ[23]	I/O
SDRC.STK_D[23]	I/O	Data I/O	DQ[24]	I/O	DQ[24]	I/O
SDRC.STK_D[22]	I/O	Data I/O	DQ[25]	I/O	DQ[25]	I/O
SDRC.STK_D[21]	I/O	Data I/O	DQ[26]	I/O	DQ[26]	I/O
SDRC.STK_D[20]	I/O	Data I/O	DQ[27]	I/O	DQ[27]	I/O
SDRC.STK_D[19]	I/O	Data I/O	DQ[28]	I/O	DQ[28]	I/O
SDRC.STK_D[18]	I/O	Data I/O	DQ[29]	I/O	DQ[29]	I/O
SDRC.STK_D[17]	I/O	Data I/O	DQ[30]	I/O	DQ[30]	I/O

† For details about temperature sensing, see Section 12.10.2.2.

Table 13−53. Stacked DDR Controller Subsystem I/O Description (Continued)

OMAP2420 Pin	Type	Description	DDR SDRAM1 Pin	Type	DDR SDRAM2 Pin	Type
SDRC.STK_D[16]	I/O	Data I/O	DQ[31]	I/O	DQ[31]	I/O
SDRC.BA[1:0]	O	Bank address 1−0	BA[1:0]	I	BA[1:0]	I
SDRC.addr[12:0]	O	Address	addr[12:0]	I	addr[12:0]	I
SDRC.addr[13]	O	Address	addr[13]	I		
SDRC.nCS0	O	Chip-select 0 (active low) − select DDR SDRAM1	nCS0	I		
SDRC.nCS1	O	Chip-select 1 (active low) − select DDR SDRAM2			nCS1	I
SDRC.CLK	O	Clock	CK	I	CK	I
SDRC.nCLK	O	Clock invert	nCK	I	nCK	I
SDRC.CKE0	O	Clock enable for chip-select 0 (active high)	CKE (DDR SDRAM1)	I		
SDRC.CKE1	O	Clock enable for chip-select 1 (active high)			CKE (DDR SDRAM2)	I
SDRC.nRAS	O	Row address strobe (active low)	nRAS	I	nRAS	I
SRC.nCAS	O	Column address strobe (active low)	nCAS	I	nCAS	I
SDRC.nWE	O	Write enable (active low)	nWE	I	nWE	I
SDRC_DM3	O	Data mask	DM0	I	DM0	I
SDRC_DM2	O	Data mask	DM1	I	DM1	I
SDRC.STK_DM1	O	Data mask	DM2	I	DM2	I
SDRC.STK_DM0	O	Data mask	DM3	I	DM3	I
SDRC.DQS3	I/O	Data strobe	DQS0	I/O	DQS0	I/O
SDRC.DQS2	I/O	Data strobe	DQS1	I/O	DQS1	I/O
SDRC.STK_DQS1	I/O	Data strobe	DQS2	I/O	DQS2	I/O

† For details about temperature sensing, see Section 12.10.2.2.

Table 13−53. Stacked DDR Controller Subsystem I/O Description (Continued)

OMAP2420 Pin	Type	Description	DDR SDRAM1 Pin	Type	DDR SDRAM2 Pin	Type
SDRC.STK_DQS0	I/O	Data strobe	DQS3	I/O	DQS3	I/O
		Temperature sensing†	TEMPWARN	O	TEMPWARN (not used)	O

† For details about temperature sensing, see Section 12.10.2.2.

Note:

Because the SDRC and the stacked DDR SDRAM have no aligned bit and byte mappings between them, the bit and byte swaps are transparent to software considerations. Bit swap is transparent to software when limited to one byte. Byte swap is transparent for software when the corresponding DM and DQS signals are also swapped.

12.9.3.1 CS0, CS1 Memory Spaces

The SDRC is directly connected to two mobile DDR SDRAM, each chip-select controlling one mobile DDR SDRAM (CS0 controls the 512Mb x 32 memory; CS1 controls the 256Mb x 32 memory). The chip-select 0 (CS0) start address is hardware-fixed with a value of 0x8000 0000. The chip-select 1 (CS1) start address is programmable through the SDRC.SDRC_CS_CFG register.

See *Chip-Select Configuration* in Section 12.11.2.2.

12.9.3.2 AC Timing Control

The SDRC controller permits the configuration of the ac timing parameters controlling the DDR SDRAM transaction timings.

A totally independent, complete set of parameters exists for each chip-select and so for each DDR SDRAM.

In the case of the OMAP2423 device, the two chip-selects control two identical memories so the two sets of ac timing parameters are the same.

For details, see *AC Timing Parameters* in Section 12.11.2.2.

12.9.3.3 Address Multiplexing

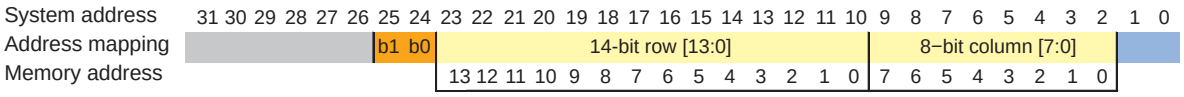
Table 13−53a and Figure 12−33a describe the address multiplexing scheme used to access the 512M-bit DDR SDRAM1 for read/write operations.

Figure 12−33a shows the 32-bit system address mapping and the relationship between the row and column addresses of the memory devices.

Table 13−53a. SDRC DDR SDRAM1 Address Multiplexing

Banks	Column Address	Row Address	Total Size (M bits)	Device Organization
BA1, BA0	A0−A7	A0−A13	512	16M words×32 bits

Figure 12−33a. SDRC DDR SDRAM1 System Address Multiplexing Scheme



This address multiplexing is chip-select dependent and programmed through ADDRMUX field SDRC.SDRC_MCFG[24:20]. See *Memory Configuration* in Section 12.11.2.2 for details.

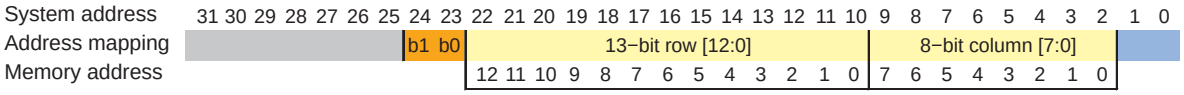
Table 13−53b and Figure 12−33b describe the address multiplexing scheme used to access the 256M-bit DDR SDRAM2 for read/write operations.

Figure 12−33b shows the 32-bit system address mapping and the relationship between the row and column addresses of the memory devices.

Table 13−53b. SDRC DDR SDRAM2 Address Multiplexing

Banks	Column Address	Row Address	Total Size (M bits)	Device Organization
BA1, BA0	A0−A7	A0−A12	256	8M words×32 bits

Figure 12−33b. SDRC DDR SDRAM2 System Address Multiplexing Scheme



12.9.4 Register Summary

Table 13−57 and Table 13−58 summarize the control registers for the SMS and SDRC modules, listing their names, type, width and physical addresses.

Table 13−57. SMS Register Summary

Register Name	Type	Register Width (Bits)	Physical Address
SMS_REVISION	R	32	0x6800 8000
SMS_SYSCONFIG	RW	32	0x6800 8010
SMS_SYSSTATUS	R	32	0x6800 8014
SMS_RG_START0 − SMS_RG_START3	RW	32	0x6800 8040− 0x6800 8070
SMS_RG_END0 − SMS_RG_END3	RW	32	0x6800 8044− 0x6800 8074
SMS_RG_ATT0 − SMS_RG_ATT3	RW	32	0x6800 8048− 0x6800 8078
SMS_RG_WRPERM0 − SMS_RG_WRPERM3	RW	32	0x6800 8080− 0x6800 80B0

Table 13−57. SMS Register Summary (Continued)

Register Name	Type	Register Width (Bits)	Physical Address
SMS_RG_RDPERM0 − SMS_RG_RDPERM3	RW	32	0x6800 8084− 0x6800 80B4
SMS_REGS_SECURITY	RW	32	0x6800 80C0
SMS_CLASS_ARBITER0	RW	32	0x6800 80D0
SMS_CLASS_ARBITER1	RW	32	0x6800 80D4
SMS_CLASS_ARBITER2	RW	32	0x6800 80D8
SMS_INTERCLASS_ARBITER	RW	32	0x6800 80E0
SMS_CLASS_ROTATION0 − SMS_CLASS_ROTATION2	RW	32	0x6800 80E4− 0x6800 80EC
SMS_ERR_ADDR	R	32	0x6800 80F0
SMS_ERR_TYPE	RW	32	0x6800 80F4
SMS_POW_CTRL	RW	32	0x6800 80F8
SMS_ROT_CONTROL0 − SMS_ROT_CONTROL3	RW	32	0x6800 8100− 0x6800 8130
SMS_ROT_SIZE__0_3	RW	32	0x6800 8104− 0x6800 8134
SMS_ROT_PHYSICAL_BA__0_3	RW	32	0x6800 8108− 0x6800 8138

Table 13−58. SDRC Register Summary

Register Name	Type	Register Width (Bits)	Physical Address
SDRC_REVISION	R	32	0x6800 9000
SDRC_SYSCONFIG	RW	32	0x6800 9010
SDRC_SYSSTATUS	R	32	0x6800 9014
SDRC_CS_CFG	RW	32	0x6800 9040
SDRC_SHARING	RW	32	0x6800 9044
SDRC_ERR_ADDR	RW	32	0x6800 9048
SDRC_ERR_TYPE	RW	32	0x6800 904C
SDRC_DLLA_CTRL	RW	32	0x6800 9060
SDRC_DLLA_STATUS	R	32	0x6800 9064
SDRC_DLLB_CTRL	RW	32	0x6800 9068
SDRC_DLLB_STATUS	R	32	0x6800 906C
SDRC_POWER	RW	32	0x6800 9070
SDRC_MCFG_0	RW	32	0x6800 9080

Table 13−58. SDRC Register Summary (Continued)

Register Name	Type	Register Width (Bits)	Physical Address
SDRC_MR_0	RW	32	0x6800 9084
SDRC_EMR2_0	RW	32	0x6800 908C
SDRC_DCDL1_CTRL	RW	32	0x6800 9094
SDRC_DCDL2_CTRL	RW	32	0x6800 9098
SDRC_ACTIM_CTRLA_0	RW	32	0x6800 909C
SDRC_ACTIM_CTRLB_0	RW	32	0x6800 90A0
SDRC_RFR_CTRL_0	RW	32	0x6800 90A4
SDRC_MANUAL_0	RW	32	0x6800 90A8
SDRC_MCFG_1	RW	32	0x6800 90B0
SDRC_MR_1	RW	32	0x6800 90B4
SDRC_EMR2_1	RW	32	0x6800 90BC
SDRC_ACTIM_CTRLA_1	RW	32	0x6800 90C4
SDRC_ACTIM_CTRLB_1	RW	32	0x6800 90C8
SDRC_RFR_CTRL_1	RW	32	0x6800 90D4
SDRC_MANUAL_1	RW	32	0x6800 90D8

12.10 Stacked DDR Controller Subsystem Functional Description

The stacked DDR controller subsystem is an integral part of the OMAP2423 device providing connectivity between the OMAP2420 application processor and two stacked DDR SDRAM memories.

This subsystem is mainly composed of four entities:

- The SDRAM memory scheduler (SMS)
- The SDRAM controller (SDRC)
- Two stacked mobile DDR RAMs

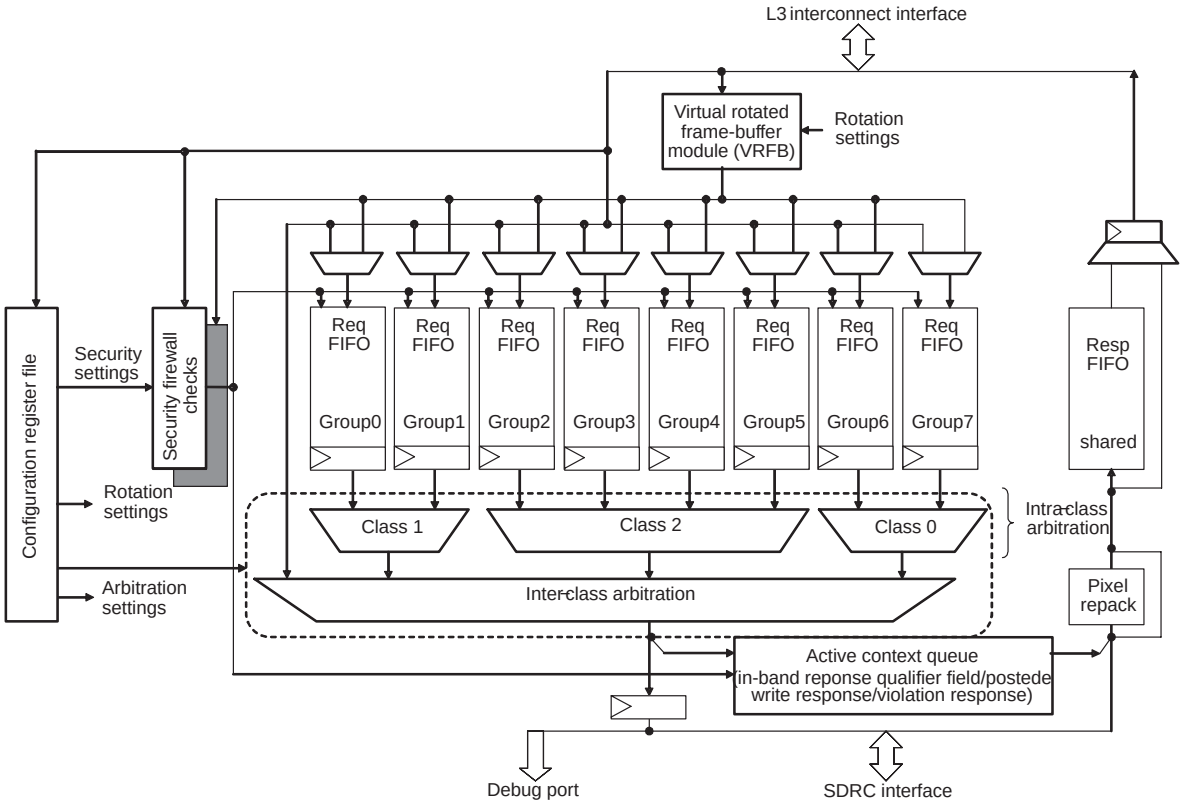
The SDRC is connected to the L3 interconnect through the SMS and also communicates with DDR SDRAM memory devices.

12.10.1 SDRAM Memory Scheduler

The SDRAM memory scheduler (SMS) optimizes the stacked mobile DDR SDRAM memories usage in order to provide the quality of service (QoS) level required by each of the initiators in the system. In addition, a security firewall allows control of the access protection to the memory and a virtual rotated frame-buffer module minimizes the SDRAM page-miss penalty when accessing rotated (i.e., nonsequentially addressed) lines in a graphics frame buffer.

Figure 12−35 shows the top-level diagram for the SMS.

Figure 12−35. SMS Top-Level Diagram



12.10.1.1 Memory Access Scheduling

When a memory transaction request arrives in the memory-access scheduler after being processed by the virtual rotated frame-buffer module and security firewall (see Section 12.10.1.4 and Section 12.10.1.5), the request is sent to one of eight deep first-in-first-out (FIFO) queues.

The allocation to a particular queue is fixed and depends on the source of the request. To prioritize concurrent transactions and optimize memory usage, each FIFO queue (and hence the memory-access request source) is categorized into one of three arbitration classes:

- ❑ Class 0 (highest priority): For bandwidth-sensitive devices with very severe real-time constraints.
- ❑ Class 1: For latency-sensitive devices where system performance degrades severely when the average memory-access latency increases.
- ❑ Class 2: For all other devices, possibly with high bandwidth requirements, but without being too latency-sensitive

Table 13−59 shows the mapping of FIFO queues and memory-access request sources to arbitration classes.

Table 13−59. Arbitration Class Allocation

Class	FIFO Queue	Source Device
0	6	Camera subsystem DMA
	7	Display subsystem
1	0	MPU subsystem CPU (instruction and data access)
	1	DSP subsystem CPU (instruction and data access)
		IVA subsystem CPU (instruction and data access)
2	2	System DMA (read)
		DSP subsystem DMA
	3	2D/3D graphics accelerator
	4	IVA subsystem video hardware accelerator
	5	System DMA (write)
		VLYNQ™
		USB

A two-level arbitration scheme determines which FIFO queue grants the next memory transaction.

Note:

In the register's description, a group represents a FIFO queue of requests from initiator.

12.10.1.2 Intra-Class Arbitration

The first level of arbitration is the intraclass arbitration. One FIFO queue is selected from each class using a least-recently-used algorithm. However, one FIFO queue in a class can be specified as high priority using the SMS.SMS_CLASS_ARBITER*c* register (where *c* is the class number: 0, 1 or 2).

A burst transaction request can be submitted to the arbitration stage as soon as either of the following occurs:

The first request of the burst is received

A complete burst is buffered in the FIFO queue.

The request can be configured on a per-FIFO queue basis via the SMS.SMS_CLASS_ARBITER*c* register.

It is also possible to use the SMS.SMS_CLASS_ARBITER*c* register to specify how many consecutive transactions (1, 2, 3, or 4) a FIFO queue is granted at a time. This can result in better memory performance because of the high probability that two consecutive transactions in a FIFO queue will access consecutive memory addresses in the same SDRAM page.

This setting is ignored for consecutive transactions split by the virtual rotated frame-buffer module. Instead, the SMS.SMS_CLASS_ROTATION*c* register specifies for the class the number of consecutive transactions granted at a time to a FIFO queue.

Note:

The granting of multiple consecutive transactions is propagated to the second level of arbitration.

12.10.1.3 Interclass Arbitration

The second level of arbitration, the interclass arbitration, decides from which of the three classes the next transaction should be granted. After class 0, which always has the highest priority, the relative priority of class 1 to class 2 is determined via programmable values in the SMS_INTERCLASS_ARBITER register, such that for a certain number of transaction class 1 has priority and for a certain number of transaction class 2 has priority.

Finally the selected transaction request is passed to the SDRC module.

Note:

A burst access is not split, even if the burst continues over the priority window. The priority window is ignored until the end of burst transmission to the SDRC.

There is one exception when the burst is split: when the FIFO (8 words of 64 bits) is not able to provide consecutive accesses.

If this occurs once, one idle is inserted by SMS.

If this occurs two consecutive times, another FIFO queue is selected and the burst is completed later on.

12.10.1.4 Security Firewall

Access permissions for the various initiators in the system can be defined for up to four memory regions. The start and end of the memory regions are specified with 64K-byte granularity via the SMS.SMS_RG_START*r* and SMS.SMS_RG_END*r* registers, respectively (whereas the number of the region: 0 to 3). Memory outside these regions is treated as a fifth region without security protection that initiators can access unconditionally.

To configure the protection attributes, write permission, and read permission of each memory region, you can use the SMS.SMS_RG_ATT*r*, SMS.SMS_RG_WRPERM*r,* and SMS.SMS_RG_RDPERM*r* registers, respectively.

Attempting an illegal access (for example, a write access to a write-protected region) indicates a security violation to the security module. The address where the security violation occurred and the type of violation can be read from the SMS.SMS_ERR_ADDR and SMS.SMS_ERR_TYPE registers, respectively.

All of these registers are only accessible in secure mode.

For more information on security, see Chapter 31.

12.10.1.5 Virtual Rotated Frame-Buffer (VRFB) Module

The VRFB module minimizes the stacked SDRAM page-crossing penalty when the image located in the frame buffer is accessed in different rotation view angles (0° view, 90° view, 180° view, and 270° view).

The VRFB allows this functionality by modifying incoming requests and virtual addresses generated by the initiators connected to the L3 interconnect (LCD controller, 3D engine, DSP, MPU…).

If the address of the request targets the virtual rotated frame-buffer address space, then the request is sent to the rotation engine. The SMS translates the virtual address into physical SDRAM addresses and reinserts a request or a multiple request in the SMS request path to the SDRC.

The 256M bytes VRFB virtual address space is split into four independently-configurable contexts, each with four views that can handle a frame buffer rotated by 0°, 90°, 180°, or 270°, as shown in Table 13−60.

Table 13−60. Virtual Rotated Frame-Buffer Module Address Space

Context	Offset Address Range	Rotation
0	0x70000000 – 0x70FFFFFF	0°
	0x71000000 – 0x71FFFFFF	90°
	0x72000000 – 0x72FFFFFF	180°
	0x73000000 – 0x73FFFFFF	270°
1	0x74000000 – 0x74FFFFFF	0°
	0x75000000 – 0x75FFFFFF	90°
	0x76000000 – 0x76FFFFFF	180°
	0x77000000 – 0x77FFFFFF	270°
2	0x78000000 – 0x78FFFFFF	0°
	0x79000000 – 0x79FFFFFF	90°
	0x7A000000 – 0x7AFFFFFF	180°
	0x7B000000 – 0x7BFFFFFF	270°
3	0x7C000000 – 0x7CFFFFFF	0°
	0x7D000000 – 0x7DFFFFFF	90°
	0x7E000000 – 0x7EFFFFFF	180°
	0x7F000000 – 0x7FFFFFFF	270°

For each of the four contexts, the buffer's physical base address, image height, width, and pixel size can be configured via three independent sets of registers, SMS.SMS_ROT_CONTROLx, SMS.SMS_ROT_SIZEx, and SMS.SMS_ROT_PHYSICAL_BAx (where x is the context number from 0 to 3).

The memory arrangement for the pixels of a frame buffer accessed through the rotation engine is page-based. A page is a rectangular array of pixels, the size of which must match the page size of the SDRAM component attached to the controller.

The page size is defined using the SMS.SMS_ROT_CONTROLx[10:8] PH and SMS.SMS_ROT_CONTROLx[6:4] PW fields.

The image height and width (IMAGEHEIGHT field SMS.SMS_ROTx_SIZE [26:16] and IMAGEWIDTH field SMS.SMS_ROTx_SIZE [10:0]) in bytes must be multiples of the page height and width, respectively.

A concrete example of configuration is shown in Section 12.11.1.2.

For more information about the VRFB module, see the *Using the OMAP2420 Device to Rotate and Display Images* (TI Literature number SWPA040) and *Understanding Image Rotation Setup on the OMAP2420 Device* (TI Literature number SWPA041) application notes available through TI representatives.

12.10.1.6 Register Security

SMS.SMS_REGS_SECURITY can be used for the following register groups to independently restrict permission to write to the SMS registers to secure access:

- VRFB context configuration registers (independently for each context)
- Memory-access scheduler arbitration registers

Writing to the security firewall memory region registers (including SMS.SMS_REGS_SECURITY) is always restricted to secure accesses.

If an illegal access is attempted (for example, a nonsecure write access to a register configured as secure), a security violation is indicated to the security module. The address where the security violation occurs and the type of violation can be read from the SMS.SMS_ERR_ADDR and SMS.SMS_ERR_TYPE registers.

For more information on security, see Chapter 31.

12.10.1.7 Module Power Saving

The SMS.SMS_SYSCONFIG register enables and disables the SMS module interconnect clock, autoidle, and power-saving mode. When this mode is enabled, all FIFO queues are empty, and there are no outstanding transactions in the SDRC module. After a delay programmable via the SMS.SMS_POW_CTRL register, the interconnect clock is disabled internally to the module (after a delay programmable via the SMS.SMS_POW_CTRL register), thus reducing power consumption.

When there is new activity on the interconnect interface, the interconnect clock is restarted without any latency penalty. After reset, this mode is disabled by default. Enabling this mode is recommended to reduce power consumption.

12.10.1.8 System Power Management

The SMS can be configured via the SMS.SMS_SYSCONFIG[4:3] SIDLEMODE field to be in one of the following idle modes:

- No-idle mode (SIDLE MODE = 0x1): the module never goes into idle state
- Force-idle mode (SIDLE MODE = 0x0): the module goes into idle state immediately after receiving the request from the PRCM module.
- Smart-idle mode (SIDLE MODE = 0x2): the module goes into idle state after receiving the request from the PRCM module and after all asserted output interrupts are acknowledged.

12.10.2 SDRAM Controller (SDRC)

The SDRAM controller (SDRC) module provides two configurable memory areas, each providing access to a stacked 256M-bit mobile DDR SDRAM. The row/column addressing scheme provides a four-bank support compliant with the two 256M-bit embedded memories.

12.10.2.1 Bank Tracking

The SDRC contains hardware for tracking open pages on a per-bank basis.

Four open pages are tracked per chip-select, for a total of eight open pages.

To efficiently pipeline, the SDRC includes a request look-ahead FIFO that analyses interconnect requests with respect to the status of the target bank.

The bank status can be:

- Bank open on another row
- Bank idle
- Bank open on the same row

The SDRC state machine generates the appropriate sequence of memory commands. All precharge and active commands are hidden whenever possible to optimize the memory bandwidth usage.

The look-ahead FIFO depth is 5×64-bit requests.

12.10.2.2 Refresh Management

Refresh management can be subdivided as follows:

- Self-refresh management
- Autorefresh management
- Temperature sensing management

Self-refresh is executed to place the stacked DDR SDRAM into an autonomous refresh mode, typically when the OMAP2423 device enters an idle mode and switches the stacked SDRAM clock off. In self-refresh, the SRAM itself supplies the row address generation required to refresh and retain data in the absence of any external clocking.

Self-refresh can be entered using any of the following methods:

- Manual software command (entry to self-refresh mode)
- Automatically on a warm reset event (if the SDRC is programmed to enter self-refresh on warm reset)
- Automatically on an idle request sent by the system power manager (if the SDRC is programmed to enter self-refresh on idle request)
- Automatically after a (programmable) period of inactivity on the interconnect interface if enabled via CLKCTRL field SDRC.SDRC_POWER[5:4] (CLKCTRL = 0x2)

Self-refresh can be exited as follows:

- Manual software command (exit from self-refresh mode). See Section 12.11.2.3.
- Automatically after receiving a read or write transaction to access memory

These self-refresh controls are shared by both chip-selects (but not by manual command registers that provide per-chip-select software control).

When Autorefresh is enabled, a programmable hardware counter within the SDRC generates a periodic event that triggers either a single refresh or a burst of consecutive refresh commands. This is the standard refresh mode used when the system is active, while the running applications regularly access the SDRAM.

An autorefresh command can also be applied using SDRC.SDRC_ MANUAL_x (see Section 12.11.2.3.). This method can be used to generate device-specific initialization sequence after power-up or after the memory device exits from a deep power down mode state.

The autorefresh request period is a user-controlled parameter programmed to meet the specification of the memory devices. Each time an autorefresh request is issued, the SDRC can service the autorefresh using one of the following commands:

- Single autorefresh command
- Burst of four autorefresh commands
- Burst of eight autorefresh commands

When a burst of four or eight is selected, the programmed period value is automatically scaled in hardware by 4 or 8. Consequently, the value to be programmed does not depend on the selected burst length.

Depending on the DDR SDRAM temperature, the autorefresh period must be adjusted to avoid the loss of data. To monitor temperature sensing, use the TEMPWARN pin of the stacked DDR SDRAM1. This output of the DDR SDRAM1 is connected to a GPIO of the OMAP2423 processor via the customer board so that an interrupt event can be triggered whenever a below-temperature crossover occurs.

For details, see Section12.11.2.6.

12.10.2.3 System Power Management

Unlike the SMS, the SDRC can only be configured in *Smart-Idle* mode via the IDLEMODE field SDRC.SDRC_SYSCONFIG[4:3] (IDLEMODE = 0x2). The module goes into idle state after having received the request from the PRCM module once all asserted output interrupts have been acknowledged.

The SDRC acknowledgement is conditioned by the internal activity of the SDRC.

12.10.2.4 Power Saving Features

The SDRC has three operating modes:

- Page opening/closure policy
- Low-power dynamic operation modes
- Static low-power modes

Page Opening/Closure Policy

The OMAP2423 device supports only one page policy (although the auto-precharge policy might be implemented in a future SDRC version). The SDRC.SDRC_POWER[0] PAGEPOLICY bit must be set to 1.

The SDRC tracks open pages and determines whether the current access is to an open page or a closed page. If the accessed page is open, the SDRC executes the access immediately. If the accessed page is closed the SDRC:

- Closes the current open page by issuing a PRECHARGE command to that bank. SDRC.SDRC_MANUAL_x[3:0] CMDCODE field is set to 0x1.
- Opens the accessed page by issuing an ACTIVE command to that bank
- Executes the access by issuing a READ or a WRITE command

Up to four pages can be opened simultaneously (one per bank). Pages remain open until one of the following events occurs:

- New read or write request to another page in the same bank
- Autorefresh request (a PRECHARGE_ALL command is issued first)
- Selfrefresh entry request (a PRECHARGE_ALL command is issued first)

Dynamic Low-Power Operating Modes

The dynamic low-power operating modes of the SDRC are designed to:

- Control the stacked DDR SDRAM clocks
- Control the internal clock gating of the SDRC when the interconnect interface is idle
- Control the self-refresh functionality

The SDRC.SRDC_POWER[2] PWDENA bit is used to activate the power-down mode of the stacked memory by pulling the relevant CKE signal at a low level each time the memory interface is not used.

When PWDENA bit is enabled, the SDRC still provides a free running clock to the stacked memory.

CKE signal is dynamically controlled based on the current memory command. There is a zero-latency penalty when this mode is enabled. This mechanism is independent from the SDRC.SDRC_POWER[5:4] CLKCTRL field.

The SDRC has three modes of autoclock behavior when the interconnect interface is idle (that is, there are no outstanding active transactions in process).

These modes are controlled via the SDRC.SDRC_POWER[5:4] CLKCTRL field and the SDRC.SDRC_POWER[23:8] AUTOCOUNT field:

- Mode 0: The autoclock gating feature is disabled. No internal clock gating is performed in the SDRC in response to the detection of an idle state on the interconnect interface.
- Mode 1: A 16-bit counter starts decrementing when an interconnect idle condition is detected. The counter start value is loaded from the AUTOCOUNT 16-bit field. When this counter times out, the internal clock gating within the SDRC is enabled. If an interconnect active command is received before the counter times out, or if an internal autorefresh request is issued, the procedure is aborted, the counter stops decrementing and is reloaded with the AUTOCOUNT value, and the request is serviced immediately.
- Mode 2: This mode is similar to mode 1, but prior to the internal clock gating, the SDRC places the SDRAM into self-refresh mode and turns the stacked SDRAM clock off. This is the lowest power mode.

To achieve maximum power saving, TI recommends to use both PWDENA and CLKCTRL-related features.

All the settings for the power saving features are common to the two chip-selects but only the accessed chip-select exits from self-refresh.

If the SDRC.SDRC_POWER[6] SRFRONIDLEREQ bit is enabled, the SDRC enters self-refresh mode on a hardware idle request from the PRCM. The memory clock is automatically switched off after which the SDRC sends an acknowledge back to the PRCM.

In this situation, it is possible for the power manager to cut off the SDRC clock. Therefore the SDRC waits for 1024 cycles (DLL relocking maximum time), before accessing the memory again when the system exits the idle state. This mechanism is independent of the SDRC.SDRC_POWER[5:4] CLKCTRL[5:4] field.

Static Low-Power Operating Modes

Software-driven controls for low-power operation modes include the capability to use SDRC.SDRC_MANUAL_x to put the memory in self-refresh or deep power-down mode.

- In self-refresh, each chip-select can be controlled independently. If both chip-selects are in self-refresh, the stacked DDR SDRAM clock can be switched off by setting the SDRC.SDRC_POWER[3] STKCLKDIS bit to 1.

  Another manual command must be used for the memory to exit self-refresh. It is the software's responsibility to ensure that the SDRAM clock is on (STKCLKDIS must be 0).

- In deep power-down mode, each chip-select is controlled independently. The stacked DDR SDRAM clock can be switched off if both chip-selects are in self-refresh or deep power-own mode, by setting the STKCLKDIS to 1.

Another manual command must be used for the memory to exit deep power-down mode. It is the software's responsibility to ensure that the SDRAM clock is on (STKCLKDIS must be 0).

After a memory exits deep power-down mode (all data are lost), a full-initialization sequence must be sent to the device before it can be used.

12.10.2.5 SDRC Power OFF Mode

In some applications, the SDRC power domain can be in the off mode while the stacked memory is in self-refresh mode.

When the SDRC in the off mode, be sure to prevent an unwanted exit from self-refresh when the context is restored. While in the off mode, the SDRC outputs that control the SDRAM are set to maintain self-refresh (CKE signal low).

When a reset occurs, the default reset state of the SDRC is power-down enable (PDE), which means that the CKE signal is held low.

When exiting the off mode:

- Power is restored to the SDRC.
- Software restores all registers except SDRC.SDRC_MR0, SDRC.SDRC_MR1, SDRC.SDRC_EMR2_0, and SDRC.SDRC_EMR2_1.
- Exit from self-refresh mode is achieved via the SDRC.SDRC_MANUAL_x[3:0] CMDCODE field. Because exit from the self-refresh field is unconditional (that is, the current state of the SDRC state machine is not considered), ensure that autorefresh is disabled.
- SDRC.SDRC_MR0, SDRC.SDRC_MR1, SDRC.SDRC_EMR2_0, and SDRC.SDRC_EMR2_1 registers can be restored to reflect the memory registers.
- The SDRAM is ready to use.

Once context is successfully restored, the software can reinitialize the SDRAM or return the SDRAM to self-refresh. When the device has successfully exited self-refresh, autorefresh is enabled again.

12.10.2.6 Digitally-Controlled Delay Line

The SDRC includes digitally-controlled delay technology for interfacing the two stacked high-speed mobile DDR memory components.

A DLL is a calibration module used to track voltage and temperature variations dynamically, as well as to compensate the silicon process dispersion (process voltage temperature (PVT) tracking).

The DLLs control several DCDL companion modules (digitally-controlled delay line) by providing an 8-bit value, continuously updated so that it always reflects a nominal 72° or 90° delay with respect to the memory interface clock frequency under all PVT conditions. For more information about the DLL and DCDL, see the *OMAP2420 DLL/DCDL Usage* (TI Literature number WMN_120_1) application note available through TI representatives.

DLL/DCDL Architecture

The DLL/DCDL architecture consists of:

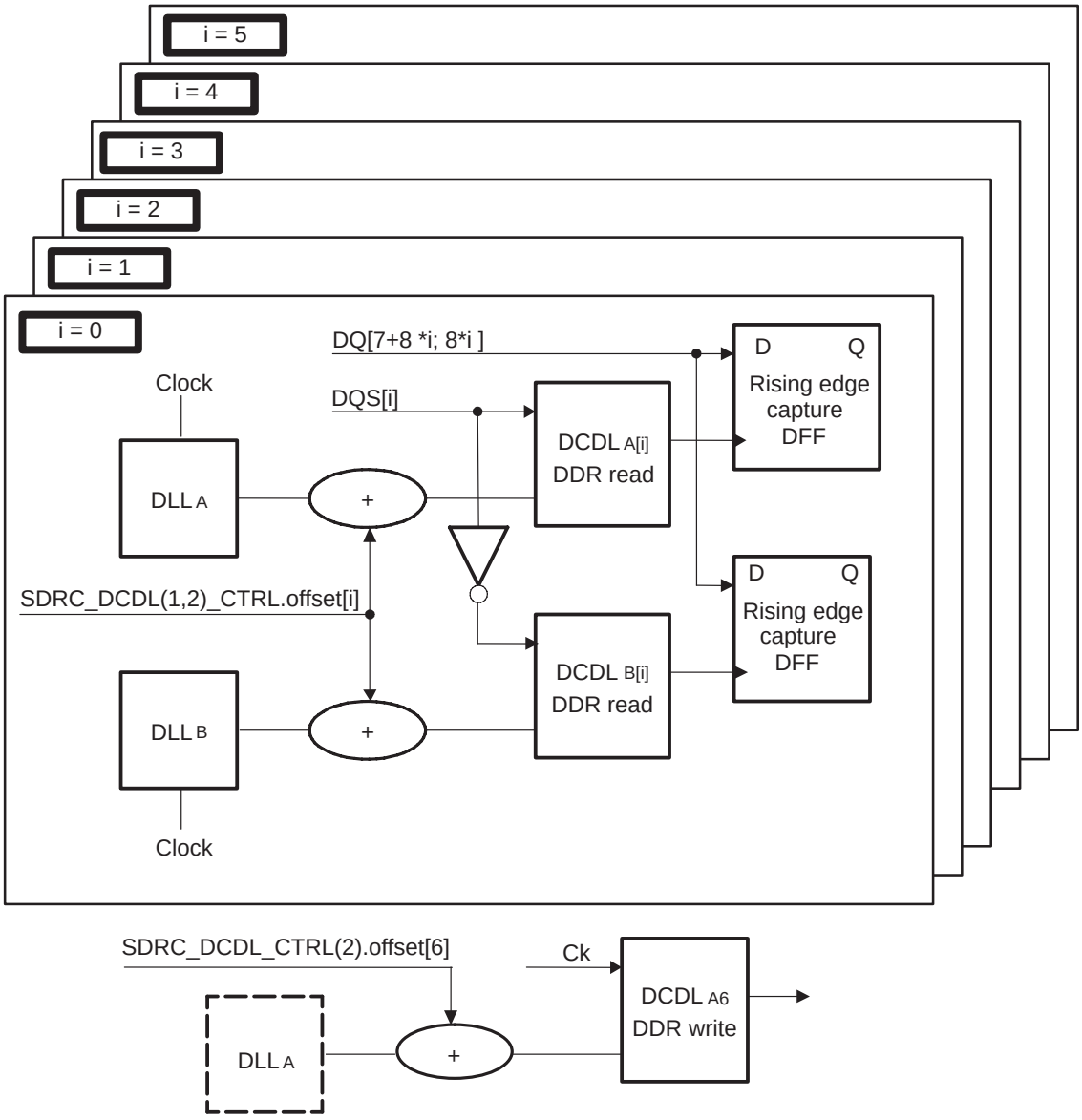
- Two DLLs: Two separate DLL elements are used in the SDRC, DLLA, and DLLB. Both DLLs are used to control the read timing (rising and falling clock edge); DLLA also controls the write timing.
- Twelve DCDLs dedicated to generating the six DQS in DDR read mode
- One DCDL dedicated to generating the DLL write clock.

Note:

The maximum SDRC operational frequency is 133 MHz, i.e., maximum operational frequency of the stacked memories.

Figure 12−36 shows the DLL/DCDL architecture.

Figure 12−36. DLL/DCDL Architecture



Note:

The OMAP2423 device uses only four DQS (i=2, i=3, i=4, and i=5) to access 32-bit data of stacked mobile DDR RAMs.

For the configuration of DLL/DCDL modules see *DLL/DCDL Configuration* in Section 12.11.2.2.

12.10.2.7 Mode Registers

The mode registers are used to define the specific mode of operation of the stacked mobile DDR SDRAM. There are two mode registers per chip select used to control the operation of the device.

The standard mode register, which is standard for all SDRAM devices, and the extended mode register, which is standard on all mobile DRAM devices.

Mode Register (MR)

This is a 13-bit register that controls the following parameters:

- CAS latency (CASL)
- Serial/interleaved mode (SIL)
- Burst length (BL)

The mode register is accessible via SDRC.SDRC_MR_*x* register (x is the CS area: 0 or 1). Writing to SDRC.SDRC_MR_*x* initiates an implicit Load Mode Register command qualified by BA1, BA0 = 0, 0.

Extended Mode Register 2 (EMR2)

This is a 13-bit register that controls the following parameters:

- Advanced temperature compensated self-refresh (ATCSR)
- Partial array self-refresh (PASR)
- Driver strength (DS)

Extended mode register 2 is accessible via SDRC.SDRC_EMR2_x register. Writing to SDRC.SDRC_EMR2_x initiates an implicit Load Mode Register command qualified by BA1, BA0 = 1, 0.

12.10.3 Stacked Mobile DDR RAM

The OMAP2423 device includes two stacked mobile DDR RAMs connected directly to the SDRC interface.

These stacked mobile DDR RAMs have the following characteristics:

- Elpida ECK2532CCCN-Y3
  - 1.8-V power supply
  - 1.8-V I/O power
  - 256Mb × 32 mobile DDR RAM
  - Four-bank operation
  - Differential clock input
  - MRS features

- CAS latency (3)
- Burst length (2, 4, 8)
- Burst sequence(sequential or interleave)

- EMRS features
  - Driver strength programmable
  - Automatic temperature compensated self-refresh
  - Partial array self-refresh (all banks, bank 0 & bank 1, or bank 0 only)

See the 256Mb × 32 DDR memory specification in Appendix C.

- Elpida ECK5132CACN
  - 1.8-V power supply
  - 1.8-V I/O power
  - 512Mb x 32 mobile DDR RAM
  - Four-bank operation
  - Differential clock output
  - MRS features
    - CAS latency (3)
    - Burst length (2, 4, 8, 16)
    - Burst sequence (sequential or interleave)
  - EMRS features
    - Driver strength programmable
    - Automatic temperature compensated self-refresh
    - Partial array self-refresh (all banks, bank 0 and bank 1, or bank 0 only)

See the 512Mb × 32 DDR memory specification in Appendix B.

12.11 Stacked DDR Controller Subsystem Programming Model

This section contains a few programming guides to help the user set up SMS and SDRC registers according to stacked mobile DDR RAM characteristics and required applications.

12.11.1 SMS Programming Model

12.11.1.1 Security Firewall Memory Region Configuration

Four memory regions can have access permissions set.

Memory region 0 is configured as follows:

- Configure the region location and size:
  - STARTADDRESS field SMS.SMS_RG_START0[30:16]: memory region 0 start (aligned on a 64K bytes boundary)
  - ENDADDRESS field SMS.SMS_RG_END0[30:16]: memory region 0 end (aligned on a 64K-byte boundary)

Note:

Memory region 0 can be disabled by setting the same value for fields START-ADDRESS and ENDADDRESS.

- Configure the region attributes:
  - REQPERM field SMS.SMS_RG_ATT0[7:4]: protection attributes (secure access only). REQPERM field allows setting the protection attributes to the relevant memory region. Setting this 4-bit field allows different types of accesses (secure, nonsecure, debug, nondebug access)

Example 12−1.

REQPERM = 0x2, only allows secure/nondebug access

REQPERM = 0x3, allows both secure/non debug and non secure/nondebug access.

- CONNIDVECTOR field SMS.SMS_RG_WRPERM0[15:0]: setting this field provides the list of all initiators that have write permission to memory region 0.
- CONNIDVECTOR field SMS.SMS_RG_RDPERM0[15:0]: setting this field provides the list of all initiators that have read permission to memory region 0

12.11.1.2 Virtual Rotated Frame-Buffer Context Configuration

Use of the rotation engine requires several initialization programming steps. Once a rotation engine context has been set up, the application is able to use it transparently, as if addressing a frame buffer object with a standard raster-based memory arrangement.

Step 1: Define the page configuration. This operation is only dependent upon the external memory device. The same page settings are then applied for any newly created rotated frame buffer.

The stacked mobile RAM device attached to the SDRC is 1024-byte pages. The page can be defined as a 16x64 array. Note that the page is not necessarily a square (32x32 would be also a suitable value). It is recommended that the longest side corresponds to the access direction requiring the maximum bandwidth.

In terms of register settings, in any context that uses that page size:

Page width = 2^{pw} bytes, where pw = 6

Page height = 2^{ph} bytes, where ph = 4

Step 2: The application needs to allocate the appropriate amount of memory in the SDRAM address space, as required by the size of the frame buffer object.

Example 12−2. Create a 400×300 frame buffer, 16 bits per pixel

- Pixel size = 2^{ps} bytes, where ps = 1
- Number of pages per line: 400x2/64 = 12.5, rounded up to 13
- Number of pages per column: 300/16 = 18.75, rounded up to 19

In terms of register settings, the image size parameters correspond to the enlarged image, and are programmed to:

- Image width = w pixels, where w = 13×64 /2 = 416
- Image height = h pixels, where h = 19×16 = 304

Note:

In this example, the values obtained are nonintegers and are rounded up to the closest integer value. This results in a loss of physical memory, generally negligible compared to the total size of the image.

In terms of memory allocation (in the physical memory), this corresponds to a 416×304×2 = 252 928-byte buffer.

The physical base address of this buffer must be aligned on a page boundary (in that example a 1024-byte boundary; that is, the 10 LSBs of the base address must be all zeros).

This buffer must be allocated as a contiguous memory segment.

All these parameters, once determined, must be loaded in the registers of the chosen context (there are four VRFB contexts with four independent sets of registers).

Example 12−3. Context 1

- Configure the physical base address of frame buffer: ex 2G bytes, start of CS0, and third quarter Q2

- SMS_ROT_PHYSICAL_BA1 [30:0] PHYSICALBA = 0x20000000

❑ Configure the image height and width:

- Image height (416): SMS_ROT_SIZE1 [IMAGEHEIGHT]: = 0x1A0
- Image width (304): SMS_ROT_SIZE1 [IMAGEWIDTH] = 0x130

❑ Configure the control parameters:

- Pixel size (ex: 2 bytes, 2^1 bytes): SMS_ROT_CONTROL1[PS] = 0x1
- Page size (ex: 1024 bytes = 64 bytes×16 bytes)
- Page height (16 bytes, 2^4 bytes): SMS_ROT_CONTROL1[PH] = 0x4
- Page width (64 bytes, 2^6 bytes): SMS_ROT_CONTROL1[PW] = 0x6

❑ The frame buffer just created is now ready to use by the different system initiators (MPU, sDMA, LCD controller, etc.)

From these modules perspective, the frame buffer object can be accessed at the following virtual rotated frame buffer address space memory locations:

❑ Context 1 0-degree view: 0x7400 0000

❑ Context 1 90-degree view: 0x7500 0000

❑ Context 1 180-degree view: 0x7600 0000

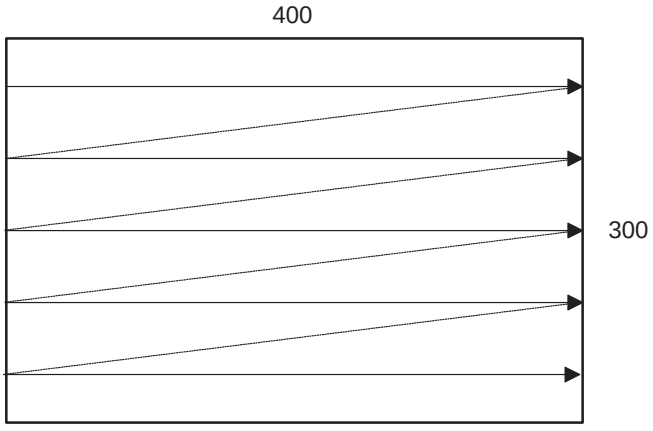
❑ Context 1 270-degree view: 0x7700 0000

The start address of the view corresponds to the logical origin (x=0, y=0) of the image. The image pitch parameter, commonly defined as the distance between two vertically-adjacent pixels, is fixed to 2048 pixels.

Example 12−4. Usage by the Application

In a system using a 400×300 LCD panel that has a native landscape orientation:

Figure 12−37. Natural Scan Order



Natural scan order

The display buffer is the one created in the example sequence.

When the application is running and using the portrait orientation for the display (typically a PDA-type application):

- ❏ The LCD controller accesses the frame buffer using the 0° view.
- ❏ The processor and other initiators such 2D DMA or 3D accelerators are using the 90° view.

When the application is running and using the landscape orientation for the display (typically a video record/player or gaming application):

- ❏ The LCD controller is still accessing the frame buffer using the 0° view.
- ❏ The processor and other initiators such 2D DMA or 3D accelerators are now also using the 0° view.

More information about manipulating rotated frame buffers (180, 270 degrees, mirroring) is available in Chapter 17, *Display Subsystem*.

For more information about the VRFB module, see the *Using the OMAP2420 Device to Rotate and Display Images* (TI Literature number SWPA040) and *Understanding Image Rotation Setup on the OMAP2420 Device* (TI Literature number SWPA041) application notes available through TI representatives.

12.11.1.3 Memory-Access Scheduler Configuration

The memory-access scheduler is configured as follows:

Register security settings:

- ❏ SMS.SMS_REGS_SECURITY
  - ■ VRFB context configuration registers independent for each context (SECUREARBITER bit SMS.SMS_REGS_SECURITY[5])
  - ■ Memory-access scheduler arbitration registers (SECUREROTCTXT bit SMS.SMS_REGS_SECURITY[x], where x is the context number from 0 to 3)

Configure the arbitration parameters for each of the three classes:

- ❏ SMS.SMS_CLASS_ARBITER0 – SMS.SMS_CLASS_ARBITER2
  - ■ One high-priority FIFO queue in the class (HIGHPRIOVECTOR field SMS.SMS_CLASS_ARBITERx[7:6], where x is the class group from 0 to 2)
  - ■ Number of consecutive transactions to perform (EXTENDED GRRANT field SMS.SMS_CLASS_ARBITERx[23:20], where x is the class group from 0 to 2)
  - ■ Burst transaction submitted for arbitration immediately or after the burst has been buffered (BURSTCOMPLETE field SMS.SMS_CLASS_ARBITERx[31:30], where x is the class group from 0 to 2)

- SMS.SMS_CLASS_ROTATION0 – SMS.SMS_CLASS_ROTATION2: Number of consecutive VRFB-split transactions to perform (NOF SERVICES field SMS.SMS_CLASS_ROTATION[4:0], where x is the class group from 0 to 2)

12.11.2 SDRAM Controller Programming Model

12.11.2.1 IP Revision

You can read the IP revision code in the REV field SDRC.SDRC_REVISION[7:0].

12.11.2.2 SDRC Configuration

Each stacked mobile DDR RAM must be configured independently, each controlled by a dedicated register set and chip-select (CS0 for DDR RAM1 and CS1 for DDR RAM2).

A number of parameters must be set before executing the initialization sequence and accessing these stacked memories.

Chip-Select Configuration

The total address space is divided into 128M-byte partitions. Each 128M-byte address space is also segmented into a 32M-byte address space. Each partition is a possible start address for CS1, except the partition occupied by CS0.

The CS0 start address is always located at 0x80000000 (with respect to the 32-bit interconnect address). Use the SDRC.SDRC_CS_CFG register to configure the CS1 start address. Two bitfields, CS1STARTLOW and CS1STARTHIGH, correspond to SDRC_CS_CFG[9:8] and SDRC_CS_CFG[3:0] and allow the value of the CS1 start address to be programmed with a minimum 32M-byte granularity.

Ensure that space 0 and 1 do not overlap each other.

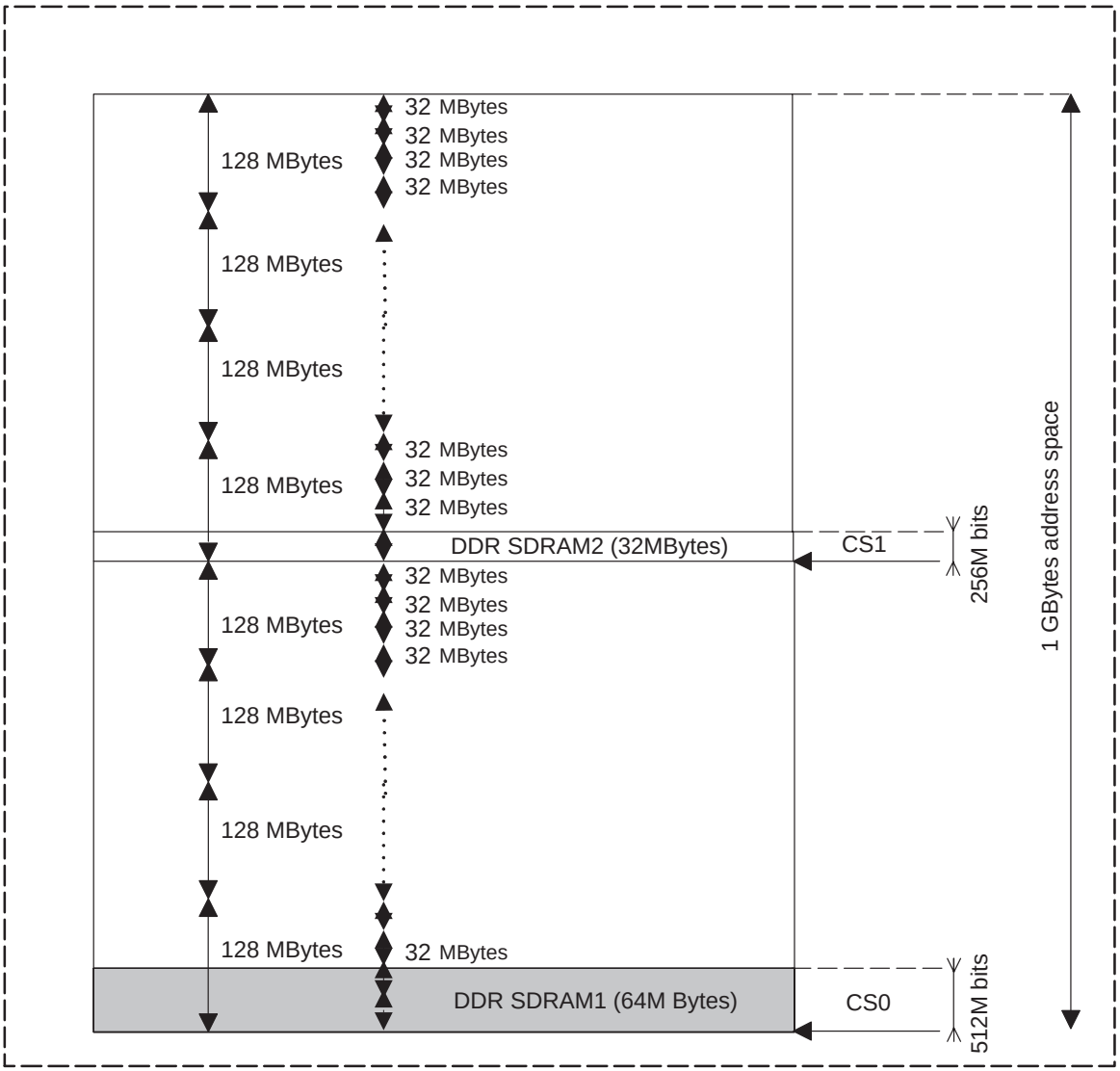
(CS1STARTLOW = 0x0 and CS1STARTHIGH = 0x0) or (CS1STARTLOW = 0x1 and CS1STARTHIGH = 0x0) are forbidden value settings.

Memory space 0 (DDR SDRAM1), selected by the SDRC module using the chip-select 0 (nCS0) output pin is always at offset 0 from the SDRAM memory space base address (0x80000000). The size of the area is programmable via the RAMSIZE field SDRC.SDRC_MCFG0[17:8] (for the OMAP2423 device, RAMSIZE = 0x20).

Memory space 1 (DDR SDRAM2), selected by the SDRC module using chip-select 1 (nCS1) output pin is located at reset at an offset 0x20000000 from the SDRAM memory space base address. The size of this area is programmable via the RAMSIZE field SDRC.SDRC_MCFG1[17:8] (for the OMAP2423 device, RAMSIZE = 0x10).

Figure 12−39 illustrates the memory space configuration for the OMAP2423 device and its two stacked MDDRs at power on reset.

Figure 12−39. CS0/CS1 Start Address Slots



To attach the two mobile DDR SDRAM devices on CS0 and CS1, use this programming model:

- ❑ Configure the SDRC.SDRC_CS_CFG register to place the CS1 start address:
  - ■ Set CS1STARTLOW = 0x0 (reset value)
  - ■ Set CS1STARTHIGH = 0x4 (reset value)

Note:

Set CS1STARTLOW = 0x2 and CS1STARTHIGH = 0x0 for a contiguous address space between the two chip-select regions.

- Configure SDRC.SDRC_SHARING register to access the two memories
  - Set CS0MUXCFG field (SDRC.SDRC_SHARING[11:9]) to 0x4 (or 0x5)
  - Set CS1MUXCFG field (SDRC.SDRC_SHARING[14:12]) to 0x4 (or 0x5)
  - Set SDRCTRISTATE bit (SDRC.SDRC_SHARING[8]) to 1

Memory Configuration

The memory configuration is defined on a per-chip-select basis via the SDRC.SDRC_MCFG_0 and SDRC.SDRC_MCFG_1 registers, which configure the SDRC interface to control the stacked DDR SRAM1 and stacked DDR SDRAM2.

Table 13−61 and Table 13−61a summarize the SDRC.SDRC_MCFG_0 and SDRC.SDRC_MCFG_1 settings used to control and access the two stacked mobile DDR RAMs.

Table 13−61. 512Mb DDR SDRAM1 Memory Configuration (CS0)

Bit Field	Register	Value	Comments
ADDRMUX	SDRC_MCFG_0[24:20]	0x1A	Address multiplexing scheme (see Section 12.9.3.3)
RAMSIZE	SDRC_MCFG_0[17:8]	0x20	The RAM size of each stacked memory is 512M bits.
B32NOT16	SDRC_MCFG_0[4]	0x1	The stacked devices are ×32 bits.
DEEPPD	SDRC_MCFG_0[3]	0x1	The memories support deep power down.
DDRTYPE	SDRC_MCFG_0[2]	0x0	Mobile DDR
RAMTYPE	SDRC_MCFG_0[1:0]	0x1	Double data rate SDRAM

Table 13−61a. 256Mb DDR SDRAM2 Memory Configuration (CS1)

Bit Field	Register	Value	Comments
ADDRMUX	SDRC_MCFG_1[24:20]	0x8	Address multiplexing scheme (see Section 12.9.3.3)
RAMSIZE	SDRC_MCFG_1[17:8]	0x10	The RAM size of each stacked memory is 256M bits.
B32NOT16	SDRC_MCFG_1[4]	0x1	The stacked devices are ×32 bits.
DEEPPD	SDRC_MCFG_1[3]	0x1	The memories support deep power down.
DDRTYPE	SDRC_MCFG_0[2] SDRC_MCFG_1[2]	0x0	Mobile DDR
RAMTYPE	SDRC_MCFG_0[1:0] SDRC_MCFG_1[1:0]	0x1	Double Data Rate SDRAM

AC Timing Parameters

Table 13−62 lists ac parameters that can be independently programmed in clock cycles for each of the two memories via registers SDRC.SDRC_ACTIM_CTRLA_0, SDRC.SDRC_ACTIM_CTRLA_1, SDRC.SDRC_ACTIM_CTRLB_0, and SDRC.SDRC_ACTIM_CTRLB_1.

Table 13−62. Programmable AC Parameters

SDRC AC Parameter	Description	Range (Clock Cycles)
tRFC	Autorefresh period	0−31
tRC	Row cycle time	0−31
tRAS	Row active time	0−15
tRP	Row precharge time	0−7
tRCD	Row to column delay time	0−7
tRRC	Active to active command period	0−7
tDPL/tWR/tCDLR	Data in to precharge command (write recovery time)	0−7
tDAL	Data in to active command	0−31
tXSR	Exit self-refresh to active time	0−255

Based on the stacked memories data sheets (see the memory specifications in Appendix B and Appendix C), Table 13−62a recommends ac parameter settings depending on TEMPWARN activity:

Table 13−62a. AC Parameters Recommended Values

SDRC AC Parameter	Recommended Value For TEMPWARN = 0		Recommended Value For TEMPWARN = 1	
	512Mb/CS0	256Mb/CS1	512Mb/CS0	256Mb/CS1
tRFC	15	15	17	17
tRC	9	12	11	14
tRAS	6	8	7	9
tRP	3	4	4	5
tRCD	4	4	5	5
tRRC	2	2	3	3
tDPL/tWR/tCDLR	2	2	3	3
tDAL	5	6	7	8
tXSR	16	16	18	18

Note:

SDRC ac parameters can be the same for the two memories. To ensure the same settings for the two memories, you must use the recommended values provided for the 256Mb x 32 memory.

For details on managing the ac parameter programming based on the temperature junction of the stacked DDR SDRAM, see Section 12.11.2.6.

Note:

All values are set for use with the SDRAM running at 133 MHz frequency and are subject to change depending on the silicon results.

For details regarding the value to set for each parameter, see the memory specification in Appendix C.

The following ac parameters cannot be programmed (hard coded):

Table 13−63. Non-Programmable AC Parameters

SDRC AC Parameter	Description	Range (Clock Cycles)
tCCD	Read/write command to read/write command	1
tCKED	CKE to clock disable or power down entry mode	1
tPED	CKE to clock enable or power down exit set up	1
tMRD	Last register command to active or refresh command	3
tDQM	Dqm to data mask during writes	0
tDQZ	Dqm to data high impedance during reads	2
tDQSS	Write command to first DQS latching transition	1
tRPRE	DQS read preamble	1

Note:

The SDRC only uses tXSR when exiting self-refresh mode, regardless of the first command issued.

The SDRC systematically inserts an autorefresh command before it serves the first request.

DLL/DCDL Configuration

The two DLLs must be configured using the SDRC.SDRC_DLLA_CTRL and SDRC.SDRC_DLLB_CTRL registers.

Note:

The stacked mobile DDR RAMs used in the OMAP2423 device operate at a maximum frequency of 133 MHz; the two DLLs must be configured in lock mode.

The DLLPHASE bit (SDRC.SDRC_DLLx_CTRL[1], where x is A or B) is used to setup the nominal delay tracked by the DLL. This delay must be set up for 90 degrees (25% of the clock period).

The DLL can be loaded with an 8-bit delay programmable value (DELAY field SDRC_DLLx_CTRL[15:8]) with a valid range of 0−239, where each step represents about 22 ps in nominal PVT conditions. This delay must be set with a granularity of 4, i.e., SDRC_DLLx_CTRL[9:8] must be set to 0b11.

The delay unit is loaded when a LOADDLL bit (SDRC.SDRC_DLLx_CTRL[2], where x is A or B) low-to-high transition occurs. The load capability can be used to shorten the locking time by setting an initial value near the expected locked state value.

Note that when loading a value into the DLL, the DLL must be enabled (SDRC.SDRC_DLLx_CTRL[0] ENADLL bit set to 1, where x is A or B) for the load to be effective, for at least two clock cycles.

To setup the controlled delay block for lock mode (normal mode):

- Write into the SDRC_DLLx_CTRL register, with ENADLL bit set, LOADDLL bit set, DLLPHASE field set to 0x77 to minimize the locking time.
- Write into the SDRC_DLLx_CTRL register, with LOADLL bit reset, to launch the tracking process.
- Wait for at least 480 L3 clock cycles to be sure the DLL is locked before accessing the memory.

Note that once locked, the DLL can be periodically disabled under software control, assuming this period is short compared with the voltage and temperature variations. When disabled, the DLL counter is frozen at its current value. When enabled again, the tracking resumes from that DLL counter value.

The two registers SDRC.SDRC_DCDL(1,2)_CTRL contain the following bit fields:

- Offset2: SDRC.DQS2 signal adjustment, read mode, nominally 0.
- Offset3: SDRC.DQS3 signal adjustment, read mode, nominally 0.
- Offset4: SDRC.STK_DQS0 signal adjustment, read mode, nominally 0.
- Offset5: SDRC.STK_DQS1 signal adjustment, read mode, nominally 0.
- Offset6: Data to DQS delay adjustment, write mode, nominally 0.

The DLL behavior can be monitored by the software using the SDRC DLL status registers (SDRC.SDRC_DLLA_STATUS and SDRC.SDRC_DLLB_STATUS). For each DLL, the current counter value is available (DLLCNT field SDRC.SDRC_DLLx_STATUS[15:8]), as well as two status bits: overflow (OVF bit SDRC.SDRC_DLLx_STATUS[0]) and underflow (UDF bit SDRC.SDRC_DLLx_STATUS[1]). Overflow is set when the DLL counter value is 255 during at least one SDRC clock cycle, and underflow is set when the DLL counter is 0 during at least one SDRC clock cycle.

These two status bits are both reset if the DLL is disabled (ENADLL bit is set to 0).

For more information about the DLL and DCDL, see the *OMAP2420 DLL/DCDL Usage* (TI Literature number WMN_120_1) application note available through TI representatives.

Programming Mode Registers

Mode Register (MR)

Following the memories characteristics, the MR programming model is as follows:

- ❑ Set CAS latency (CASL field SDRC.SDRC_MRx[6:4], where x is the CS area: 0 or 1) to 3
- ❑ Set burst sequence (SIL bit SDRC.SDRC_MRx[3]) to 0 (Serial mode)
- ❑ Set burst length (BL field SDRC.SDRC_MRx[2:0],) to 4 (SDRC limitation for DDR SDRAM)

The mode register is accessible via SDRC.SDRC_MR_*x*. Writing to SDRC.SDRC_MR_*x* initiates an implicit load mode register command that is qualified by BA1, BA0 = 0, 0.

Extended Mode Register 2 (EMR2)

This register controls functions specific to LOW POWER operation. These additional functions include drive strength (SDRC.SDRC_EMR2_x[6:5] DS field, where x is the CS area: 0 or 1), partial array self-refresh (SDRC.SDRC_EMR2_x[2:0] PASR field), and advanced temperature-compensated self-refresh (SDRC.SDRC_EMR2_x[9] ATCSR bit).

The OMAP2423 device has default values for the extended mode register 2 (if not programmed, the device operates with default values – PASR = 000 (all banks), DS = 00 (normal), ATCSR = 0 (enable))

To ensure the extended memory mode register is correctly set up:

- ❑ Set SDRC.SRDC_EMR2_0 to 0x000
- ❑ Set SDRC.SDRC_EMR2_1 to 0x000

Note:

Setting SDRC.SDRC_EMR2_x to 0x000 sets DS field in normal strength mode. This value is subject to change according silicon results.

Extended mode register 2 is accessible via SDRC.SDRC_EMR2_x. Writing to SDRC.SDRC_EMR2_x initiates an implicit Load Mode Register command qualified by BA1, BA0 = 1, 0.

12.11.2.3 Manual Software Commands

The manual command registers, SDRC.SDRC_MANUAL_x (where x is the CS area: 0 or 1), implement software-driven commands to set the CMDCODE field SDRC.SDRC_MANUAL_x[3:0]:

- ❑ No operation (NOP/INHIBIT) command (CMDCODE: 0x0)

  When the CMDCODE bits SDRC_MANUAL_X[3:0] are programmed with 0x0, the SDRC generates the NOP/INHIBIT command.

  Table 13−63a lists the status of the SDRC memory port signals for the NOP/INHIBIT command.

Table 13−63a. Status of the SDRC Memory Port Signals for No Operation Command

Command	nCS	NRAS	NCAS	NWE
INHIBIT	H	X	X	X
NOP	0	H	H	H

NOP does not initiate a new operation but is needed to complete operations requiring more than a single clock cycle, for example, autorefresh.

INHIBIT is also a NOP. nCS high disables the command decoder so that NRAS, NCAS, NWE and all the address inputs are ignored.

- Precharge all command (CMDCODE: 0x1)

  When the CMDCODE bits SDRC_MANUAL_X [3:0] are programmed with 0x1, SDRC generates the precharge all command.

  Table 13−63b lists the status of the SDRC memory port signals for the precharge all command.

Table 13−63b. Status of the SDRC Memory Port Signals for Precharge All Command

Command	nCS	NRAS	NCAS	NWE
Precharge all	L	L	H	L

All banks must satisfy the tRAS (min) requirement before executing the precharge all command.

During the precharge all command, address A10 remains high and bank information (BA0 and BA1) is Don't Care.

All banks can be precharged at the same time using the precharge all command.

At the end of tRP, after performing precharge to all of the banks, all banks are in idle state.

- Autorefresh command (CMDCODE: 0x2)

  When the CMDCODE bits SDRC_MANUAL_X[3:0] are programmed with 0x2, SDRC generates the autorefresh command.

  Table 13−63c lists the status of the SDRC memory port signals for the autorefresh command.

Table 13−63c. Status of the SDRC Memory Port Signals for Autorefresh Command

Command	nCS	NRAS	NCAS	NWE
Autorefresh	L	L	L	H

In addition to the signal status, the CKE signal is at logic high for autorefresh.

The autorefresh command can only be asserted with all banks in idle state when the device is not in power-down mode (CKE is high in the previous cycle).

The autorefresh command must be followed by NOPs until the autorefresh operation is complete. All banks are in the idle state at the end of the autorefresh operation.

- Enter deep power down (ENTER DPDM) command (CMDCODE: 0x3)

  When the CMDCODE bits SDRC_MANUAL_X[3:0] are programmed with 0x3, the SDRC executes the ENTER DPDM command, which is used for

low-power devices, that is, LPSDR or MDDR devices supporting the deep power-down mode).

Table 13−63d lists the status of the SDRC memory port signals for the deep power-down mode.

Table 13−63d. Status of the SDRC Memory Port Signals for Deep Power-Down Mode

Command	nCS	NRAS	NCAS	NWE	CKE
Enter DPMD	0	H	H	L	0

The device enters the deep power-down mode when nCS and NWE are held at logic low, NRAS and NCAS are high at the rising edge of the clock, and CKE is low.

- Exit deep power down (EXIT DPDM) command (CMDCODE: 0x4)

  When the CMDCODE bits SDRC_MANUAL_X [3:0] are programmed with 0x4, the SDRC executes the EXIT DPDM command. The device exits the deep power-down mode when the inhibit command is sampled with CKE held at logic high.

- Enter self-refresh command (CMDCODE: 0x5)

  The SDRC puts memory in self-refresh when the CMDCODE bits SDRC_MANUAL_X [3:0] are programmed with 0x5. The signal status is the same as that defined for the autorefresh command. In addition, CKE is held at logic low during the enter self-refresh command.

  The device enters the self-refresh mode from the all-banks idle state by asserting low on nCS, NRAS, NCAS, and CKE with high on NWE.

  When the device enters the self-refresh mode, CKE low is the only state that matters; all other inputs, including the clock, are ignored to remain in the self-refresh mode.

- Exit self-refresh command (CMDCODE: 0x6)

  When the CMDCODE bits SDRC_MANUAL_X[3:0] are programmed with 0x6, the SDRC executes the exit self-refresh command and, after meeting the tXSR timing parameter, executes one autorefresh command to adhere to the memory protocol.

  The device enters the exit self-refresh mode when CKE is detected high with the NOP command.

  To exit self-refresh, restart the external clock, assert CKE high, and then follow with NOPs for a minimum time of tXSR before the SDRAM reaches idle state to begin normal operation.

- Set CKE signal high command (CMDCODE: 0x7)

  When the CMDCODE bits SDRC_MANUAL_X [3:0] are programmed with 0x7, CKE is set high (for example, during DDR memory initialization).

- Set CKE signal low command (CMDCODE: 0x8)

  When the CMDCODE bits SDRC_MANUAL_X [3:0] are programmed with 0x8, CKE is set low. (for example, when placing memory in power-down mode).

12.11.2.4 Power Management

Clock Enable Management

The SDRC supports two autonomous clock enable signals (SDRC.CKE0, SDRC.CKE1), and each chip-select (CS0, CS1) has its own CKE signal. The power management of the CS0 memory is controlled by SDRC.CKE0. The power management of the CS1 memory is controlled by SDRC.CKE1. This allows the SDRAM memories associated with CS0 and CS1 to be independently placed in either power down (PD), deep power down (DPD), or self-refresh (SR) mode.

Stopping the Memories Clocks

One method of controlling the power efficiency in applications is to stop the clocks (SDRC.CLK and SDRC.nCLK) which control the stacked memories. The clocks can be stopped altogether, if no data accesses are in progress.

The SDRC power management register contains the bit STKCLKDIS (SDRC.SDRC_POWER[3]) that controls suspension of the memory clocks.

Writing 1 to STKCLKDIS bit freezes the clocks (logical 0 is applied). Writing 0 enables the clock.

Power Down and Deep Power Down Mode

In power down mode, the power consumption of the mobile DDR RAM is suppressed by deactivating its input initial circuit. No internal memory refresh operation occurs during the power down mode.

Enabling the PWDENA field SDRC.SDRC_POWER[2] activates the power down management of the two stacked memories (see Section 12.10.2.4 for functional details).

Deep power down mode achieves maximum power reduction by eliminating the power from the entire memory array. The programming model for deep power down is as follows. Before executing deep power down, all banks must be precharged or in idle state.

Deep Power Down Entry

Precharge all banks (CMDCODE: 0x1)

Enter deep power down mode (CMDCODE: 0x3)

Deep Power Down Exit

Exit deep power down mode (CMDCODE: 0x4)

The values of mode registers (SDRC.SDRC_MR_0, SDRC.SDRC_MR_1) and extended mode register (SDRC.SDRC_EMR_0, SDRC.SDRC_EMR_1) are retained upon exiting deep power down.

12.11.2.5 Refresh Management

Self-Refresh

The programming model for entering and exiting self-refresh mode is as follows:

Self-Refresh Entry

- ❑ PRECHARGE all banks (CMDCODE: 0x1)
- ❑ NOP (CMDCODE: 0x0)
- ❑ Enter self-refresh mode (CMDCODE: 0x5)

The software does not have to disable the autorefresh (ARE field SDRC.SDRC_RFR_CTRL_x[1:0], where x is the CS area: 0 or 1) before entering self-refresh. The autorefresh counter is reset in hardware so that an autorefresh cycle is automatically generated just following self-refresh exit and prior to any other command.

Self-Refresh Exit

- ❑ Exit self-refresh mode (CMDCODE: 0x6)
- ❑ Reconfigure SDRC registers as required
- ❑ Enable autorefresh by programming:
  - ■ ARCV field SDRC.SDRC_RFR_CTRL_x[23:8]
  - ■ ARE field (SDRC.SDRC_RFR_CTRL_x[1:0]) to the desired refresh burst

Autorefresh

The SDRAM refresh configuration register group controls refresh management in normal operation. This group contains two SDRC.SDRC_RFR_CTRL registers that are defined on a per-chip-select basis and contain the following fields:

- ❑ ARE field SDRC.SDRC_RFR_CTRL_x[1:0]
- ❑ ARCV field SDRC.SDRC_RFR_CTRL_x[23:8]

The ARE field enables and disables autorefresh. Autorefresh bursts of 1, 4, and 8 are programmed using this field. The autorefresh burst starts when the 16-bit autorefresh counter decrements to zero. The ARCV field loads the autorefresh counter with a 16-bit value. The ARCV is calculated using this formula:

Autorefresh counter value = (Memory refresh interval time/clock period) – margin (50 cycles)

The margin takes into account the fact that they may have an ongoing access when the counter expires, delaying the effective refresh sequence.

Note:

Memory refresh interval time is in time units.

Example 12−5.

In a normal operational mode using the DDR SDRAM at 133 MHz, ARCV is defined as follows:

$$ARCV = (7.8\,\mu s / 7.5\ ns) - 50 = 990 = 0x03DE$$

The value to be programmed is independent of the burst refresh configuration. If burst refresh is selected, the value is automatically scaled to the burst refresh size in hardware.

The autorefresh is enabled using the CMDCODE field of the relevant SDRC.SDRC_MANUAL_x register (CMDCODE: 0x2).

12.11.2.6 Temperature Sensing Management

The temperature warning signal (TEMPWARN), an output of the mobile DDR RAM, is connected to the TEMPWARN ball of the OMAP2423 processor. The connection allows monitoring of TEMPWARN activity for tuning ac parameters and the autorefresh duty cycle, depending on the stacked DDR SDRAM temperature junction.

TEMPWARN must be connected to a GPIO of the OMAP2423 processor via the customer board so that an interrupt event can be triggered to detect a temperature crossover.

Software can then adjust the autorefresh duty cycle and ac parameters for each DDR SDRAM. Table 13−63e and Table 13−63f show the 1-bit temperature sensing parameters for DDR SDRAM1 and DDR SDRAM2.

Table 13−63e. 1-Bit Temperature Sensing for DDR SDRAM1

Temp Warning Value	DDR Junction Temp Range	512Mb Stacked Memories Autorefresh Duty Cycle	ARCV Value
0	−20°C to +85°C	7.8 µs	990 (0x03DE)
1	+85°C to +105°C	1.95 µs	210 (0x00D2)

Table 13−63f. 1-Bit Temperature Sensing for DDR SDRAM2

Temp Warning Value	DDR Junction Temp Range	256Mb Stacked Memories Autorefresh Duty Cycle	ARCV Value
0	−20°C to +85°C	7.8 µs	990 (0x03DE)
1	+85°C to +105°C	1.95 µs	210 (0x00D2)

To monitor this temperature, the programming model is as follows:

- ❑ Program the GPIO connected to TEMPWARN to generate an interrupt request on a TEMPWARN event. The GPIO must be configured as an external input interrupt with a trigger feature (see Chapter 29 for details).
- ❑ Following the TEMPWARN activity (0 or 1), reconfigure the ac parameters of the two stacked mobile DDR SDRAM and SDRC.SDRC_RFR_CTRL_0, SDRC.SDRC_RFR_CTRL_1 with the good ARCV value

For more information on ac parameters to set depending TEMPWARN activity, see *AC Timing Parameters* in Section 12.11.2.2.

12.11.2.7 Error Management

All data transfers in the SDRC operate a system of full handshaking. A valid read or write request that is presented to the SDRC by the L3 interconnect sequencer will result in the SDRC acknowledging the transfer by raising a dedicated flag. Failure to do this within a defined temporal window constitutes an error. Errors can arise from the following sources:

- ❑ Transaction error while the memory is in deep power down mode.
- ❑ Interconnect transaction error.
- ❑ An illegal initiator access.
- ❑ Capture of the address of the last illegal access in the SDRC.SDRC_ERR_ADDR register

If an error occurs, the software error handler:

- ❑ Interrogates the ERRORVALID bit (SDRC.SDRC_ERR_TYPE[0]) to verify presence of an error.
- ❑ Interrogates the ERRORDPD bit (SDRC.SDRC_ERR_TYPE[1]) to ascertain the presence or not of a transaction error resulting from the device being in deep power down mode.
- ❑ Interrogates the ERRORMCMD field (SDRC.SDRC_ERR_TYPE[6:4]) to ascertain the presence or not of a transaction error resulting from an interconnect transaction error.
- ❑ Interrogates the ERRORCONNID field (SDRC.SDRC_ERR_TYPE[11:8]) to ascertain the presence or not of a transaction error resulting from an illegal access from an interconnect initiator.
- ❑ Interrogates the ERRORADD field (SDRC.SDRC_ERR_TYPE[3:2]) to ascertain the presence or not of a transaction error resulting from an illegal address
  - ■ Interconnect access to an address outside the memory space (ERRORADD = 0x0)
  - ■ Interconnect access to an address outside the register space (ERRORADD = 0x1)
  - ■ Interconnect access to a memory location which is not refreshed (ERRORADD = 0x3)

- Writes 0 to the **ERRORVALID** bit to clear the active error status.
- Executes error recovery from software

12.11.2.8 Basic Initialization Sequence For DDR SDRAM Operations

The mobile DDR RAM is initialized in the power-on sequence according to the following.

- To stabilize internal circuits, when power is applied, a 200µs or longer pause must precede any signal toggling.
- After the pause, all banks must be precharged using the precharge command.
  - SDRC.SDRC_MANUAL_0[3:0] CMDCODE field: 0x1
  - SDRC.SDRC_MANUAL_1[3:0] CMDCODE field: 0x1
- Once the precharge is completed, two or more autorefresh must be performed.
  - SDRC.SDRC_MANUAL_0[3:0] CMDCODE field: 0x2
  - SDRC.SDRC_MANUAL_1[3:0] CMDCODE field: 0x2
- Both the mode register and the extended mode register must be programmed:
  - Configure SDRC.SDRC_MR_0 and SDRC.SDRC_MR_1
  - Configure SDRC.SDRC_EMR2_0 and SDRC.SDRC_EMR2_1

After the mode register set cycle or the extended mode register set cycle, a pause (2 clocks minimum) must be satisfied.

Note:

Because the mode register powers up in a unknown state, it must be loaded prior to applying any operational command.

CKE and DM must be held high until the precharge command is issued to ensure data-bus High-Z.

12.12 Stacked DDR Controller Subsystem Registers

12.12.1 Stacked DDR Controller Subsystem Instance Summary

Table 13−64 shows the base address and address space for the SMS module instance.

Table 13−64. SMS Instance Summary

Module Name	Base Address	Size
SMS	0x6800 8000	64K bytes

Table 13−65 shows the base address and address space for the SDRC module instance.

Table 13−65. SDRC Instance Summary

Module Name	Base Address	Size
SDRC	0x6800 9000	64K bytes

12.12.2 Stacked DDR Controller Subsystem Register Summary

This section summarizes the stacked DDR controller subsystem module registers.

Table 13−66. SMS Register Summary

Register Name	Type	Register Width (Bits)	Physical Address
SMS_REVISION	R	32	0x6800 8000
SMS_SYSCONFIG	RW	32	0x6800 8010
SMS_SYSSTATUS	R	32	0x6800 8014
SMS_RG_START0 − SMS_RG_START3	RW	32	0x6800 8040− 0x6800 8070
SMS_RG_END0 − SMS_RG_END3	RW	32	0x6800 8044− 0x6800 8074
SMS_RG_ATT0 − SMS_RG_ATT3	RW	32	0x6800 8048− 0x6800 8078
SMS_RG_WRPERM0 − SMS_RG_WRPERM3	RW	32	0x6800 8080− 0x6800 80B0
SMS_RG_RDPERM0 − SMS_RG_RDPERM3	RW	32	0x6800 8084− 0x6800 80B4
SMS_REGS_SECURITY	RW	32	0x6800 80C0
SMS_CLASS_ARBITER0	RW	32	0x6800 80D0
SMS_CLASS_ARBITER1	RW	32	0x6800 80D4
SMS_CLASS_ARBITER2	RW	32	0x6800 80D8
SMS_INTERCLASS_ARBITER	RW	32	0x6800 80E0

Table 13−66. SMS Register Summary (Continued)

Register Name	Type	Register Width (Bits)	Physical Address
SMS_CLASS_ROTATION0 − SMS_CLASS_ROTATION2	RW	32	0x6800 80E4− 0x6800 80EC
SMS_ERR_ADDR	R	32	0x6800 80F0
SMS_ERR_TYPE	RW	32	0x6800 80F4
SMS_POW_CTRL	RW	32	0x6800 80F8
SMS_ROT_CONTROL0 − SMS_ROT_CONTROL3	RW	32	0x6800 8100− 0x6800 8130
SMS_ROT_SIZE__0_3	RW	32	0x6800 8104− 0x6800 8134
SMS_ROT_PHYSICAL_BA__0_3	RW	32	0x6800 8108− 0x6800 8138

Table 13−67. SDRC Register Summary

Register Name	Type	Register Width (Bits)	Physical Address
SDRC_REVISION	R	32	0x6800 9000
SDRC_SYSCONFIG	RW	32	0x6800 9010
SDRC_SYSSTATUS	R	32	0x6800 9014
SDRC_CS_CFG	RW	32	0x6800 9040
SDRC_SHARING	RW	32	0x6800 9044
SDRC_ERR_ADDR	RW	32	0x6800 9048
SDRC_ERR_TYPE	RW	32	0x6800 904C
SDRC_DLLA_CTRL	RW	32	0x6800 9060
SDRC_DLLA_STATUS	R	32	0x6800 9064
SDRC_DLLB_CTRL	RW	32	0x6800 9068
SDRC_DLLB_STATUS	R	32	0x6800 906C
SDRC_POWER	RW	32	0x6800 9070
SDRC_MCFG_0	RW	32	0x6800 9080
SDRC_MR_0	RW	32	0x6800 9084
SDRC_EMR2_0	RW	32	0x6800 908C
SDRC_DCDL1_CTRL	RW	32	0x6800 9094
SDRC_DCDL2_CTRL	RW	32	0x6800 9098
SDRC_ACTIM_CTRLA_0	RW	32	0x6800 909C
SDRC_ACTIM_CTRLB_0	RW	32	0x6800 90A0

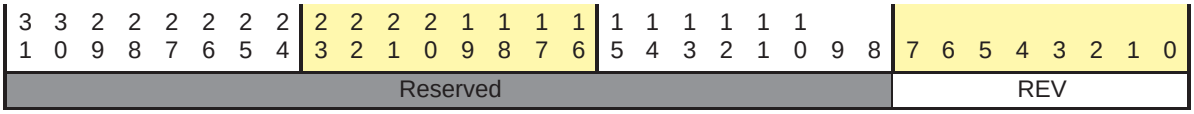
Table 13−67. SDRC Register Summary (Continued)

Register Name	Type	Register Width (Bits)	Physical Address
SDRC_RFR_CTRL_0	RW	32	0x6800 90A4
SDRC_MANUAL_0	RW	32	0x6800 90A8
SDRC_MCFG_1	RW	32	0x6800 90B0
SDRC_MR_1	RW	32	0x6800 90B4
SDRC_EMR2_1	RW	32	0x6800 90BC
SDRC_ACTIM_CTRLA_1	RW	32	0x6800 90C4
SDRC_ACTIM_CTRLB_1	RW	32	0x6800 90C8
SDRC_RFR_CTRL_1	RW	32	0x6800 90D4
SDRC_MANUAL_1	RW	32	0x6800 90D8

12.12.3 SMS Register Descriptions

Table 13−68. SMS_REVISION

Address Offset	0x0000		
Physical Address	0x6800 8000	**Instance**	SMS1
Description	This register contains the IP revision code		
Type	R		
Write Latency	Not relevant		

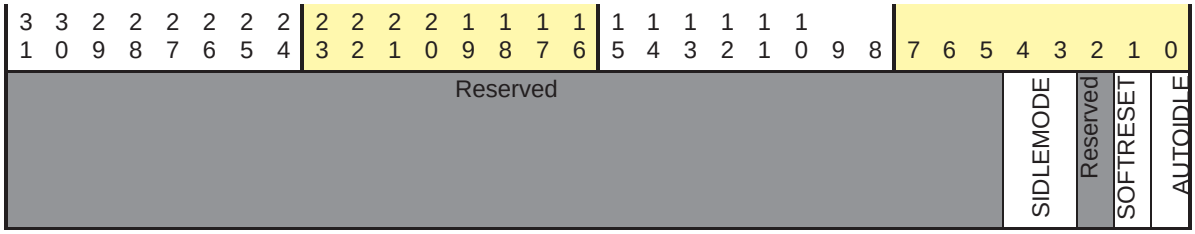


Bits	Field Name	Description	Type	Reset
31:8	Reserved	Reads returns 0	R	0x000000
7:0	REV	IP revision code [7:4] Major revision [3:0] Minor revision Examples: 0x10 for 1.0, 0x21 for 2.1	R	†

† Internal data

Table 13−69. SMS_SYSCONFIG

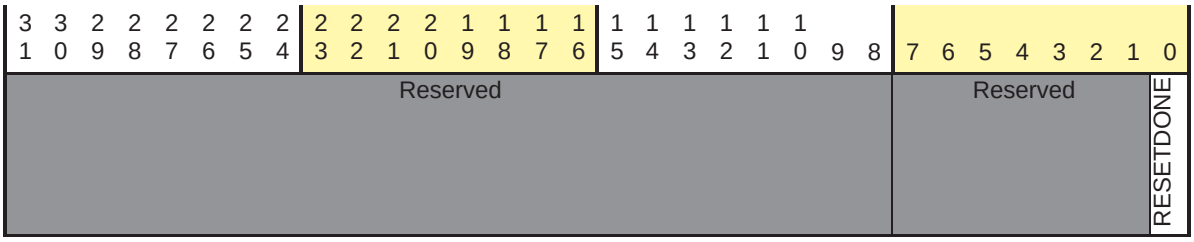
Address Offset	0x0010		
Physical Address	0x6800 8010	**Instance**	SMS1
Description	This register controls the various parameters of the OCP interface		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:5	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0000000
4:3	SIDLEMODE	Power Management Req/Ack Control	RW	0x0
		0x0: Force Idle—An idle request is acknowledged unconditionally		
		0x1: No Idle—An idle request is never acknowledged.		
		0x2: Smart Idle—Acknowledgement to an idle request is based on the internal activity of the module		
		0x3: Reserved—Do not use.		
2	Reserved	Write 0s for future compatibility. Reads returns 0s	RW	0
1	SOFTRESET	Software reset	RW	0
		0x0: Normal mode (no reset applied)		
		0x1: Software reset is activated		
0	AUTOIDLE	Internal OCP clock gating strategy	RW	0
		0x0: OCP clock is free-running		
		0x1: Automatic OCP clock gating strategy is applied, based on the OCP interface activity		

Table 13−70. SMS_SYSSTATUS

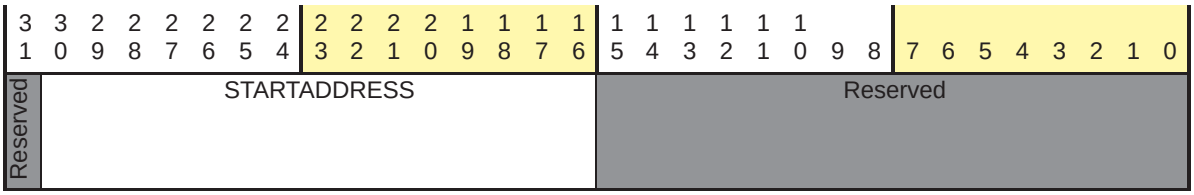
Address Offset	0x0014		
Physical Address	0x6800 8014	**Instance**	SMS1
Description	This register provides status about the module excluding interrupt status info		
Type	R		
Write Latency	Not relevant		



Bits	Field Name	Description	Type	Reset
31:8	Reserved	Reserved for Module-specific Status Information. Read returns 0s.	R	0x000000
7:1	Reserved	Reserved for OCP socket Status Information. Read returns 0s.	R	0x00
0	RESETDONE	Internal Reset Monitoring 0x0: Internal Module Reset is ongoing 0x1: Reset completed—The module is ready to be used	R	−

Table 13−71. SMS_RG_START0—SMS_RG_START3

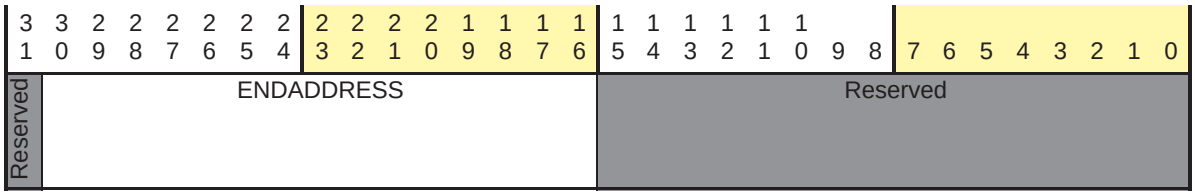
Address Offset	0x0040−0x0070 in 0x10 byte increments		
Physical Address	0x6800 8040−0x6800 8070	**Instance**	SMS1
Description	This register provides the region #i start address (lowest address inside the region), with a 64KB granularity.		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31	Reserved	Write 0s for future compatibility. Reads returns 0	RW	0
30:16	STARTADDRESS	Region #i start address (included in the region) Aligned on 64KB boundary. [15:0] must be written with 0s	RW	0x0000
15:0	Reserved	Write 0s for future compatibility. Reads returns 0	RW	0x0000

Table 13−72. SMS_RG_END0—SMS_RG_END3

Address Offset	0x0044−0x0074 in 0x10 byte increments
Physical Address	0x6800 8044−0x6800 8074 **Instance** SMS1
Description	This register provides the region #i end address (lowest address outside the region), with a 64KB granularity.
Type	RW
Write Latency	Immediate



Bits	Field Name	Description	Type	Reset
31	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0
30:16	ENDADDRESS	Region #i end address (not included in the region) Aligned on 64KB boundary. [15:0] must be written with 0s	RW	0x0000
15:0	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0000

Table 13−73. SMS_RG_ATT0—SMS_RG_ATT3

Address Offset	0x0048−0x0078 in 0x10 byte increments
Physical Address	0x6800 8048−0x6800 8078 **Instance** SMS1
Description	This register provides the protection attributes for region #i. Can only be programmed by a secure transaction
Type	RW
Write Latency	Immediate

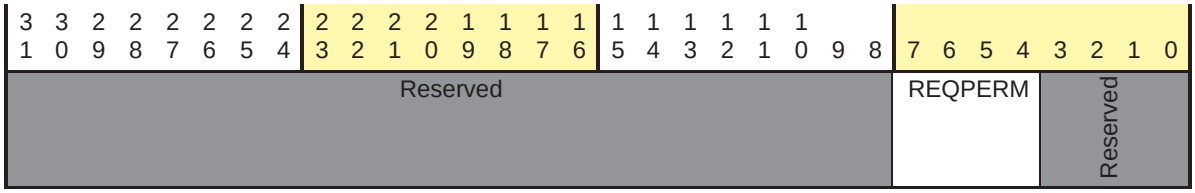
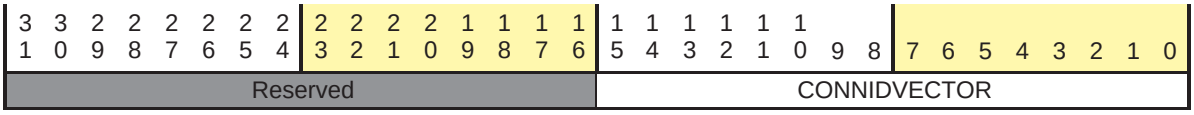


Table 13−73. SMS_RG_ATT0—SMS_RG_ATT3 (Continued)

Bits	Field Name	Description	Type	Reset
31:8	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x000000
7:4	REQPERM	Region access permission with respect to request qualifiers (secure transaction and debug transaction) ReqPerm[0] must be set to 1 if non secure/non debug accesses are permitted ReqPerm[1] must be set to 1 if secure/non debug accesses are permitted ReqPerm[2] must be set to 1 if non secure/debug accesses are permitted ReqPerm[3] must be set to 1 if secure/debug accesses are permitted	RW	0xF
3:0	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0

Table 13−74. SMS_RG_WRPERM0—SMS_RG_WRPERM3

Address Offset	0x0080−0x00B0 in 0x10 byte increments
Physical Address	0x6800 8080−0x6800 80B0 **Instance** SMS1
Description	This register provides the list of all initiators that have permission for writing to that memory region
Type	RW
Write Latency	Immediate



Bits	Field Name	Description	Type	Reset
31:16	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0000
15:0	CONNIDVECTOR	Vector providing the write permission of up to 16 initiators. Any non authorized access generates a security violation. For bit[k], k = 0 to 15: 0: Initiator (ConnID=k) has no write permission 1: Initiator (ConnID=k) has the write permission	RW	0x0000

Table 13−75. SMS_RG_RDPERM0—SMS_RG_RDPERM3

Address Offset	0x0084−0x00B4 in 0x10 byte increments
Physical Address	0x6800 8084−0x6800 80B4 **Instance** SMS1
Description	This register provides the list of all initiators that have permission for reading from that memory region
Type	RW
Write Latency	Immediate

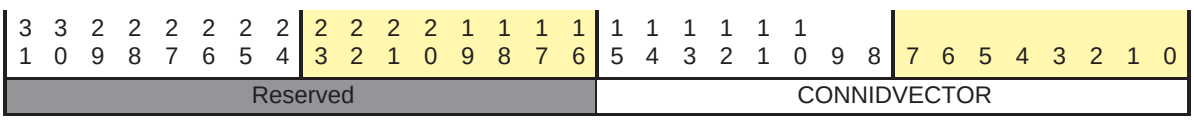
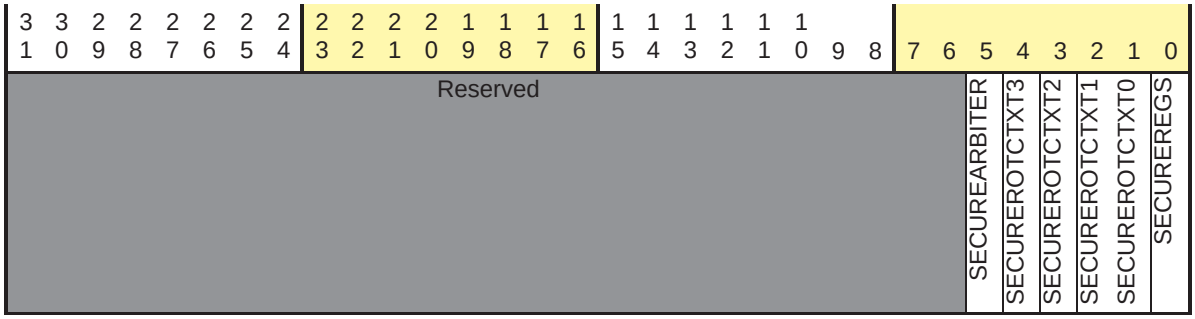


Table 13−75. SMS_RG_RDPERM0—SMS_RG_RDPERM3 (Continued)

Bits	Field Name	Description	Type	Reset
31:16	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0000
15:0	CONNIDVECTOR	Vector providing the read permission of up to 16 initiators. Any non authorized access generates a security violation. For bit[k], k = 0 to 15: 0: Initiator (ConnID=k) has no read permission 1: Initiator (ConnID=k) has the read permission	RW	0x0000

Table 13−76. SMS_REGS_SECURITY

Address Offset	0x00C0		
Physical Address	0x6800 80C0	**Instance**	SMS1
Description	This register provides the security level required to access all SMS registers. The security level is either register-specific, or common to a set of registers. Any write access to SMS_REG_SECURITY must be secure, else a violation exception is generated. Read is allowed unconditionally.		
Type	RW		
Write Latency	Immediate		



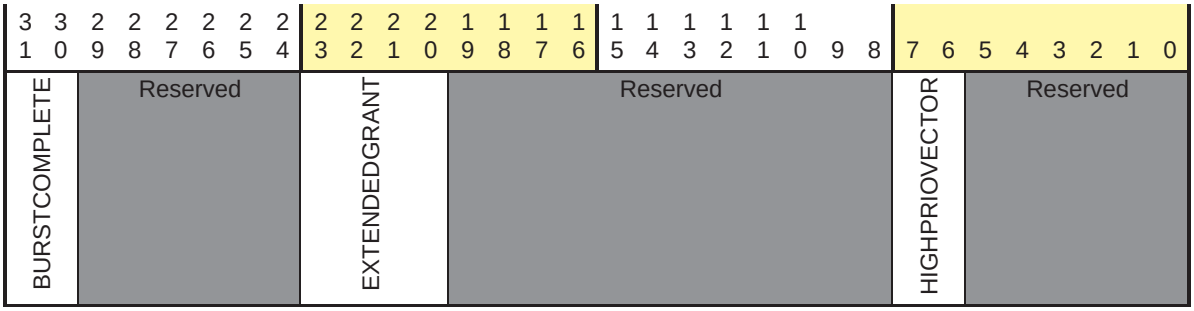
Bits	Field Name	Description	Type	Reset
31:6	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0000000
5	SECUREARBITER	Sets the security level to program the SMS group 2 arbitration control registers (reads are always allowed). 0x0: Any transaction is allowed 0x1: Secure transaction required	RW	0
4	SECURER-OTCTXT3	Sets the security level to program rotation engine context 3 (reads are always allowed). 0x0: Any transaction is allowed 0x1: Secure transaction required	RW	0
3	SECURER-OTCTXT2	Sets the security level to program rotation engine context 2 (reads are always allowed). 0x0: Any transaction is allowed 0x1: Secure transaction required	RW	0

Table 13−76. SMS_REGS_SECURITY (Continued)

Bits	Field Name	Description	Type	Reset
2	SECURER-OTCTXT1	Sets the security level to program rotation engine context 1 (reads are always allowed). 0x0: Any transaction is allowed 0x1: Secure transaction required	RW	0
1	SECURER-OTCTXT0	Sets the security level to program rotation engine context 0 (reads are always allowed). 0x0: Any transaction is allowed 0x1: Secure transaction required	RW	0
0	SECUREREGS	Sets the security level to program all group 1 registers (reads are always allowed). This bit is hardwired to 1. 0x1: Secure transaction required	R	1

Table 13−77. SMS_CLASS_ARBITER0

Address Offset	0x00D0		
Physical Address	0x6800 80D0	**Instance**	SMS1
Description	This register controls the arbitration parameters between the class #i request groups		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:30	BURST COMPLETE	Delayed service until burst request complete BurstComplete[k], k= 6 to 7 (BurstComplete[30] is for group number 6, BurstComplete[31] is for group number 7) 0x0: Group #k request to arbiter issued as soon as the first burst request is available 0x1: Group #k request to arbiter delayed until a complete burst transaction is buffered	RW	0x0
29:24	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00

Table 13−77. SMS_CLASS_ARBITER0 (Continued)

Bits	Field Name	Description	Type	Reset
23:20	EXTENDED-GRANT	Extended grant service inside a class Vector specifying the number of consecutive services a group can be granted consecutively. 2 bits per group ExtendedGrant[2*k+1,2*k], k = 6 to 7 (Extendend-Grant[21:20] is for group number 6, Extended-Grant[23:22] is for group number 7) 0x0: 1 service for group #k when granted 0x1: 2 services for group #k when granted 0x2: 3 services for group #k when granted 0x3: 4 services for group #k when granted	RW	0x0
19:8	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x000
7:6	HIGHPRIO VECTOR	High priority attribute inside a class Vector allocating a higher priority to one of the class members. A single group may be given this attribute at a time. HighPrioVector[k], k= 6 to 7 (HighPrioVector[6] is for group number 6, HighPrioVector[7] is for group number 7) 0x0: Group #k has standard priority (LRU based) 0x1: Group #k has the highest priority over all other class members	RW	0x0
5:0	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00

Table 13−78. SMS_CLASS_ARBITER1

Address Offset	0x00D4		
Physical Address	0x6800 80D4	**Instance**	SMS1
Description	This register controls the arbitration parameters between the class #i request groups		
Type	RW		
Write Latency	Immediate		

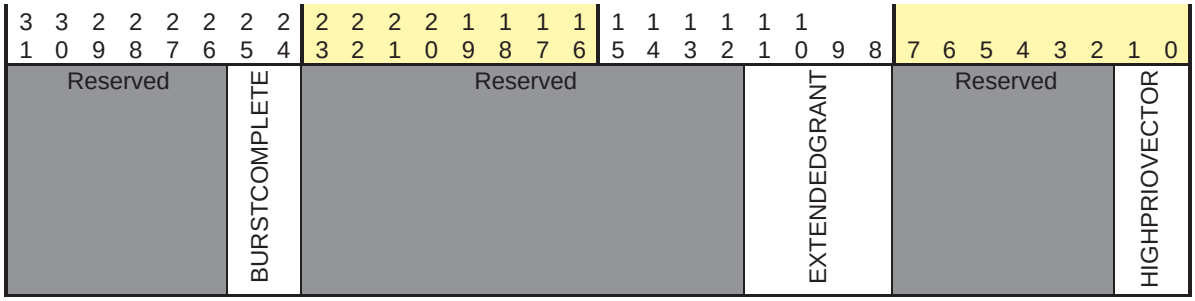
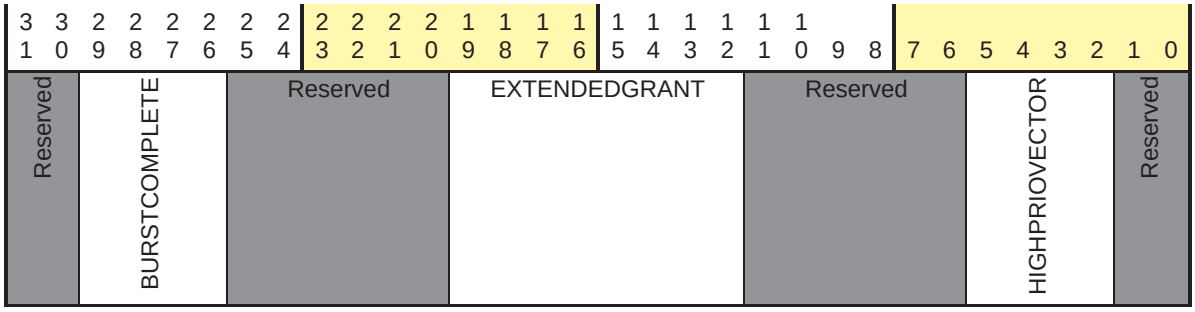


Table 13−78. SMS_CLASS_ARBITER1 (Continued)

Bits	Field Name	Description	Type	Reset
31:26	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00
25:24	BURST COMPLETE	Delayed service until burst request complete BurstComplete[k], k= 0 to 1 (BurstComplete[24] is for group number 0, BurstComplete[25] is for group number 1) 0x0: Group #k request to arbiter issued as soon as the first burst request is available 0x1: Group #k request to arbiter delayed until a complete burst transaction is buffered	RW	0x0
23:12	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x000
11:8	EXTENDED-GRANT	Extended grant service inside a class Vector specifying the number of consecutive services a group can be granted consecutively. 2 bits per group ExtendedGrant[2*k+1,2*k], k = 0 to 1 (Extendend-Grant[9:8] is for group number 0, ExtendedGrant[11:10] is for group number 1) 0x0: 1 service for group #k when granted 0x1: 2 services for group #k when granted 0x2: 3 services for group #k when granted 0x3: 4 services for group #k when granted	RW	0x0
7:2	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00
1:0	HIGHPRIO VECTOR	High priority attribute inside a class Vector allocating a higher priority to one of the class members. A single group may be given this attribute at a time. HighPrioVector[k], k= 0 to 1 (HighPrioVector[1] is for group number 1, HighPrioVector[0] is for group number 0) 0x0: Group #k has standard priority (LRU based) 0x1: Group #k has the highest priority over all other class members	RW	0x0

Table 13−79. SMS_CLASS_ARBITER2

Address Offset	0x00D8		
Physical Address	0x6800 80D8	**Instance**	SMS1
Description	This register controls the arbitration parameters between the class #i request groups		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:30	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0
29:26	BURST COMPLETE	Delayed service until burst request complete BurstComplete[k], k= 2 to 5 (BurstComplete[29] is for group number 5, BurstComplete[28] is for group number 4, BurstComplete[27] is for group number 3, BurstComplete[26] is for group number 2) 0x0: Group #k request to arbiter issued as soon as the first burst request is available 0x1: Group #k request to arbiter delayed until a complete burst transaction is buffered	RW	0x0
25:20	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00
19:12	EXTENDED-GRANT	Vector specifying the number of consecutive services a group can be granted consecutively. 2 bits per group ExtendedGrant[2*k+1,2*k], k = 2 to 5 (Extendend-Grant[19:18] is for group number 5, Extended-Grant[17:16] is for group number 4, Extendend-Grant[15:14] is for group number 3, Extended-Grant[13:12] is for group number 2) 0x0: 1 service for group #k when granted 0x1: 2 services for group #k when granted 0x2: 3 services for group #k when granted 0x3: 4 services for group #k when granted	RW	0x00
11:6	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00

Table 13−79. SMS_CLASS_ARBITER2 (Continued)

Bits	Field Name	Description	Type	Reset
5:2	HIGHPRIO VECTOR	Vector allocating a higher priority to one of the class members. A single group may be given this attribute at a time. HighPrioVector[k], k= 2 to 5 (HighPrioVector[5] is for group number 5, HighPrioVector[4] is for group number 4, HighPrioVector[3] is for group number 3, HighPrioVector[2] is for group number 2) 0x0: Group #k has standard priority (LRU based) 0x1: Group #k has the highest priority over all other class members	RW	0x0
1:0	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0

Table 13−80. SMS_INTERCLASS_ARBITER

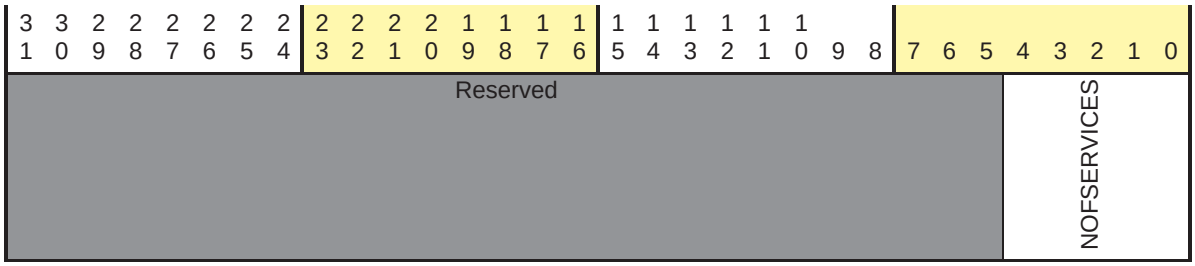
Address Offset	0x00E0		
Physical Address	0x6800 80E0	**Instance**	SMS1
Description	This register controls the PWM counter that defines the priority switching between class 1 and class 2		
Type	RW		
Write Latency	Immediate		

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	CLASS2PRIO	Reserved	CLASS1PRIO

Bits	Field Name	Description	Type	Reset
31:24	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00
23:16	CLASS2PRIO	Class 2 high priority window width (clock cycle count)	RW	0x40
15:8	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00
7:0	CLASS1PRIO	Class 1 high priority window width (clock cycle count)	RW	0x40

Table 13−81. SMS_CLASS_ROTATION0—SMS_CLASS_ROTATION2

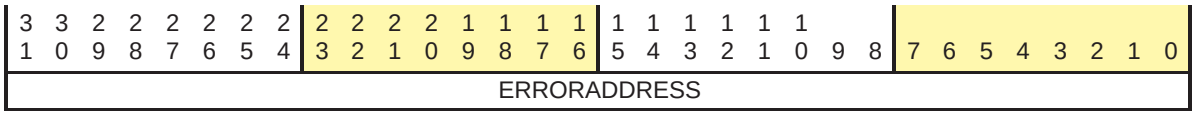
Address Offset	0x00E4−0x00EC in 0x4 byte increments
Physical Address	0x6800 80E4−0x6800 80EC **Instance** SMS1
Description	This register controls the number of consecutive services that can be allocated to a thread the transactions of which have been split by the rotation engine.
Type	RW
Write Latency	Immediate



Bits	Field Name	Description	Type	Reset
31:5	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0000000
4:0	NOFSERVICES	Number of RE split transactions serviced consecutively when the thread gets granted by the arbitration logic.	RW	0x00

Table 13−82. SMS_ERR_ADDR

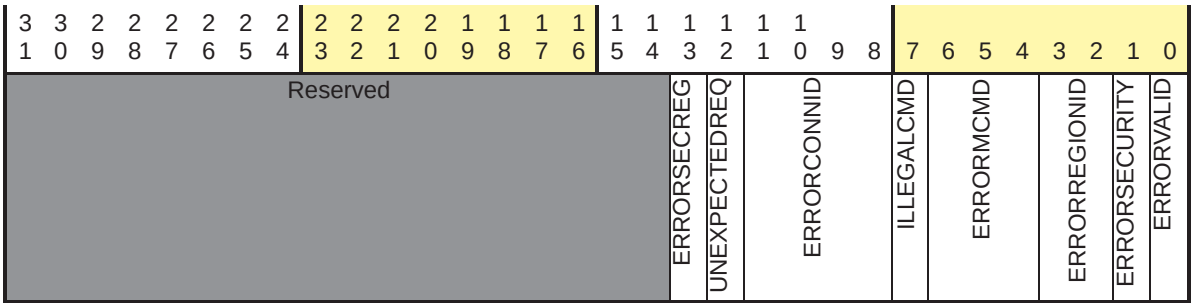
Address Offset	0x00F0
Physical address	0x6800 80F0 **Instance** SMS1
Description	This register captures the address of an access that has generated an error
Type	R
Write Latency	Not relevant



Bits	Field Name	Description	Type	Reset
31:0	ERRORADDRESS	Extended grant service inside a class (bit 31 is always 0)	R	0x00000000

Table 13−83. SMS_ERR_TYPE

Address Offset	0x00F4		
Physical Address	0x6800 80F4	**Instance**	SMS1
Description	This register provides additional information about the access that has generated the security error		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:14	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00000
13	ERRORSECREG	SMS security register accessed by non secure write transaction 0x0: No violation detected on secure registers 0x1: A non-secure write access has been detected to the security control registers. Such writes are not performed (reads are always allowed)	RW	0
12	UNEXPECTEDREQ	Unexpected request received during SMS idle state 0x0: No unexpected request received 0x1: A request has been received on the OCP interface while the SMS has been put in idle mode by the system power manager	RW	0
11:8	ERRORCONNID	Identifies the illegal access initiator OCP ConnID of the illegal access initiator: See the top level documentation for the device using the SDRC module.	R	0x0
7	ILLEGALCMD	Illegal command on the L3 interface 0x0: No illegal command received 0x1: Illegal command has been received	RW	0
6:4	ERRORMCMD	OCP command that caused the error	R	0x0
3:2	ERRORREGIONID	Region ID of the region that has been illegally accessed	R	0x0

Table 13−83. SMS_ERR_TYPE (Continued)

Bits	Field Name	Description	Type	Reset
1	ERRORSECURITY	Security violation error 0x0: No illegal command received 0x1: Security violation was detected	RW	0
0	ERRORVALID	Error validity status—Must be explicitly cleared with a write transaction 0x0: All error fields no longer valid 0x1: Error detected and logged in the other error fields	RW	0

Table 13−84. SMS_POW_CTRL

Address Offset	0x00F8		
Physical Address	0x6800 80F8	**Instance**	SMS1
Description	This register controls the SMS power management, in conjunction with the regular OCP socket registers		
Type	RW		
Write Latency	Immediate		

31–8	7–0
Reserved	IDLEDELAY

Bits	Field Name	Description	Type	Reset
31:8	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x000000
7:0	IDLEDELAY	Delay before auto-idle when no more traffic in the SMS 8-bit value, in L3 clock cycle unit	RW	0x80

Table 13−85. SMS_ROT_CONTROL0—SMS_ROT_CONTROL3

Address Offset	0x0100−0x0130 in 0x10 byte increments		
Physical Address	0x6800 8100−0x6800 8130	**Instance**	SMS1
Description	The control register configures the Virtual Rotated Frame Buffer module for the context #i.		
Type	RW		
Write Latency	Immediate		

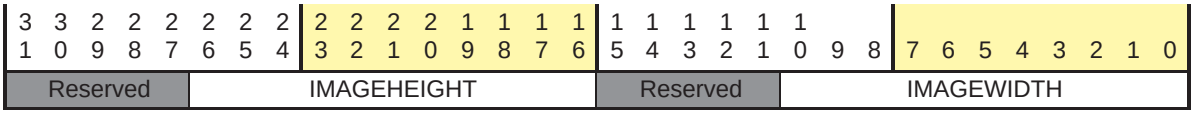
31–8	7	6–5	4	3–2	1–0
Reserved	PH	Reserved	PW	Reserved	PS

Table 13−85. SMS_ROT_CONTROL0—SMS_ROT_CONTROL3 (Continued)

Bits	Field Name	Description	Type	Reset
31:11	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x000000
10:8	PH	Exponent based 2 value, 2^{ph} indicates the page height in rows for the context #i	RW	0x0
7	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0
6:4	PW	Exponent based 2 value, 2^{pw} indicates the page width in bytes for the context #i	RW	0x0
3:2	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0
1:0	PS	Exponent based 2 value, 2^{ps} indicates the pixel size in bytes for the context #i. The value 3 is invalid.	RW	0x0

Table 13−86. SMS_ROT_SIZE0—SMS_ROT_SIZE3

Address Offset	0x0104−0x0134 in 0x10 byte increments
Physical Address	0x6800 8104−0x6800 8134 **Instance** SMS1
Description	The control register configures the bank organization for the context #i.
Type	RW
Write Latency	Immediate



Bits	Field Name	Description	Type	Reset
31:27	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00
26:16	IMAGEHEIGHT	Image height in pixels for the context #i	RW	0x000
15:11	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x00
10:0	IMAGEWIDTH	Image width in pixels for the context #i	RW	0x000

Table 13−87. SMS_ROT_PHYSICAL_BA0—SMS_ROT_PHYSICAL_BA3

Address Offset	0x0108−0x0138 in 0x10 byte increments
Physical Address	0x6800 8108−0x6800 8138 **Instance** SMS1
Description	The control register allows to configure the Physical Base Address for the context i.
Type	RW
Write Latency	Immediate

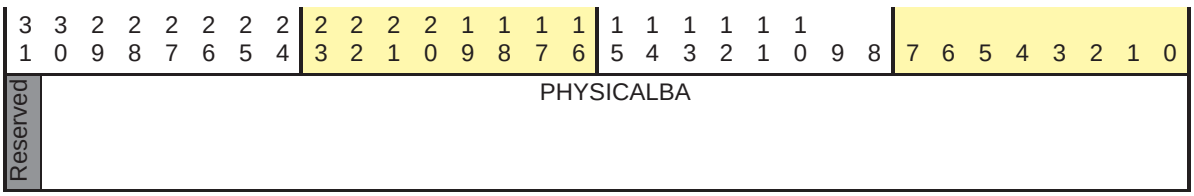


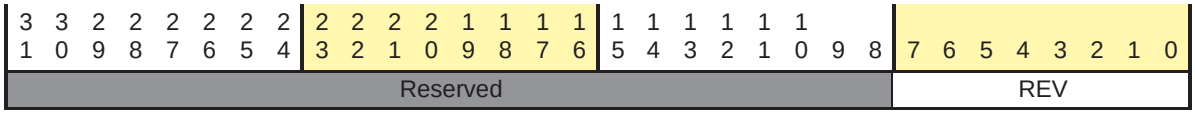
Table 13−87. SMS_ROT_PHYSICAL_BA0—SMS_ROT_PHYSICAL_BA3 (Continued)

Bits	Field Name	Description	Type	Reset
31	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0
30:0	PHYSICALBA	Physical base address of the frame buffer for the context #i in SDRAM	RW	0x00000000

12.12.4 SDRC Registers Descriptions

Table 13−88. SDRC_REVISION

Address Offset	0x0000		
Physical Address	0x6800 9000	**Instance**	SDRC1
Description	This register contains the IP revision code		
Type	R		
Write Latency	Not relevant		

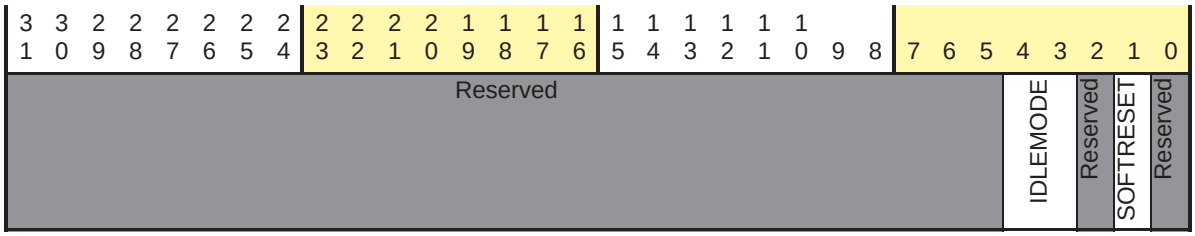


Bits	Field Name	Description	Type	Reset
31:8	Reserved	Reads returns 0	R	0x000000
7:0	REV	IP revision code [7:4] Major revision [3:0] Minor revision Examples: 0x10 for 1.0, 0x21 for 2.1	R	†

† TI internal data

Table 13−89. SDRC_SYSCONFIG

Address Offset	0x0010		
Physical Address	0x6800 9010	**Instance**	SDRC1
Description	This register controls the various parameters of the OCP interface		
Type	RW		
Write Latency	Immediate		



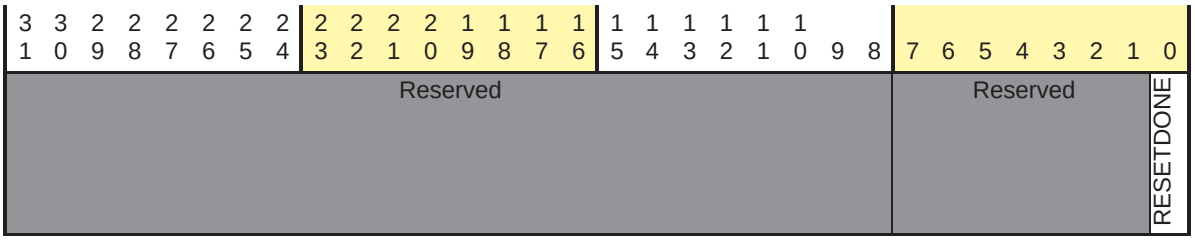
Bits	Field Name	Description	Type	Reset
31:5	Reserved	Write 0s for future compatibility Reads returns 0	RW	0x0000000

Table 13−89. SDRC_SYSCONFIG (Continued)

Bits	Field Name	Description	Type	Reset
4:3	IDLEMODE	Power Management Req/Ack Control 0x0: Reserved—Do not use. 0x1: Reserved—Do not use. 0x2: Smart Idle—Acknowledgement to an idle request is based on the internal activity of the module. Issued when the SDRC enters self-refresh 0x3: Reserved—Do not use.	RW	0x2
2	Reserved	Write 0s for future compatibility Read returns 0	RW	0
1	SOFTRESET	Software reset 0x0: Normal mode (no reset applied) 0x1: Software reset is activated	RW	0
0	Reserved	Write 0s for future compatibility Read returns 0	RW	0

Table 13−90. SDRC_SYSSTATUS

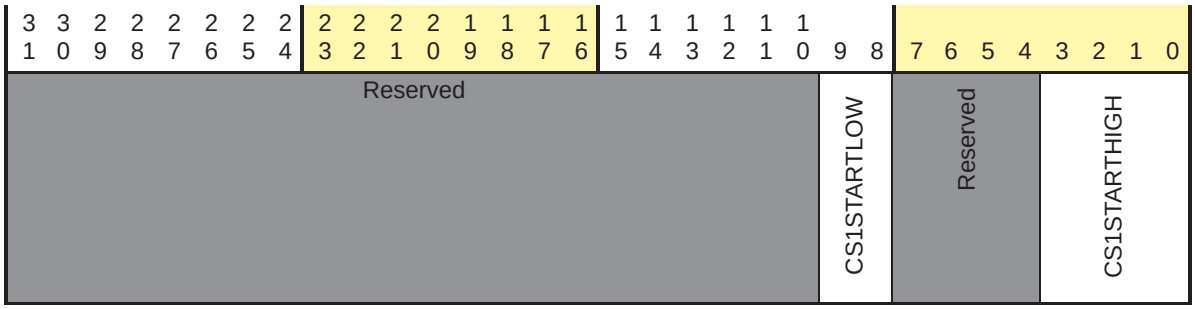
Address Offset	0x0014		
Physical Address	0x6800 9014	**Instance**	SDRC1
Description	This register provides status about the module excluding interrupt status info		
Type	R		
Write Latency	Not relevant		



Bits	Field Name	Description	Type	Reset
31:8	Reserved	Reserved for Module-specific Status Information Reads returns 0	R	0x000000
7:1	Reserved	Reserved for OCP socket Status Information Reads returns 0	R	0x00
0	RESETDONE	Internal Reset Monitoring 0x0: Internal Module Reset is ongoing 0x1: Reset completed—The module is ready to be used	R	−

Table 13−91. SDRC_CS_CFG

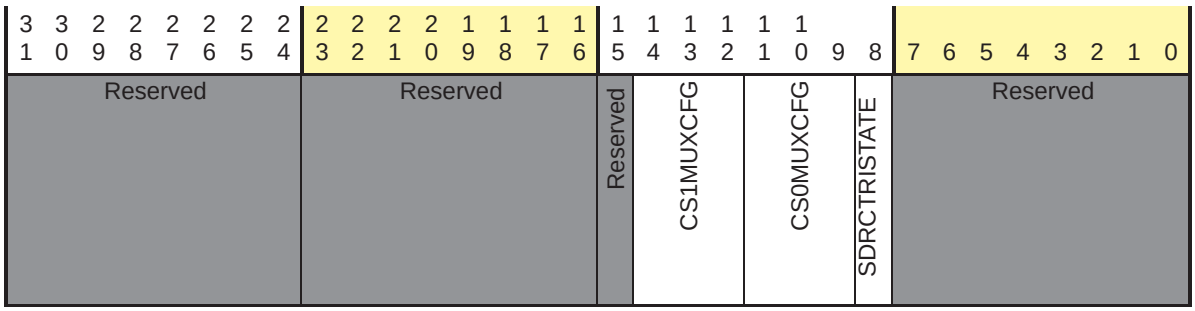
Address Offset	0x0040
Physical Address	0x6800 9040 **Instance** SDRC1
Description	This register configures the start address of CS1 address space. Must be aligned on a boundary that is a multiple of the size of the attached memory or of the next power of two if the memory size is not a power of two.
Type	RW



Bits	Field Name	Description	Type	Reset
31:10	Reserved	Write 0s for future compatibility Reads return 0	RW	0x000000
9:8	CS1STARTLOW	CS1 address space start address (lower add bits a1:a0) / 32MB unit	RW	0x0
7:4	Reserved	Write 0s for future compatibility Reads return 0	RW	0x0
3:0	CS1STARTHIGH	CS1 address space start address (upper add bits a5:a4:a3:a2) / 128MB unit	RW	0x4

Table 13−92. SDRC_SHARING

Address Offset	0x0044
Physical Address	0x6800 9044 **Instance** SDRC1
Description	This register controls the GPMC access with respect to the SDRC, in a pin-sharing configuration
Type	RW
Write Latency	Immediate



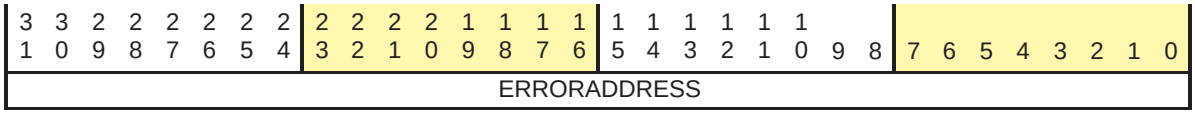
Bits	Field Name	Description	Type	Reset
31:24	Reserved	Write 0s for future compatibility Read returns 0	RW	0x00
23:16	Reserved	Write 0s for future compatibility Read returns 0	RW	0x00

Table 13−92. SDRC_SHARING (Continued)

Bits	Field Name	Description	Type	Reset
15	Reserved	Write 0s for future compatibility Read returns 0	RW	0
14:12	CS1MUXCFG	Identifies the SDRC pins used by CS1	RW	0x2
		0x0: Not used for the OMAP2423 device		
		0x1: Not used for the OMAP2423 device		
		0x2: Not used for the OMAP2423 device		
		0x3: Not used for the OMAP2423 device		
		0x4: 32-bit stacked DDR SDRAM on Data-lane[31:0]		
		0x5: 32-bit stacked DDR SDRAM on Data-lane[31:0]		
		0x6: Reserved for future use		
		0x7: Not used for the OMAP2423 device		
11:9	CS0MUXCFG	Identifies the SDRC pins used by CS0	RW	0x3
		0x0: Not used for the OMAP2423 device		
		0x1: Not used for the OMAP2423 device		
		0x2: Not used for the OMAP2423 device		
		0x3: Not used for the OMAP2423 device		
		0x4: 32-bit stacked DDR SDRAM on Data-lane[31:0]		
		0x5: 32-bit stacked DDR SDRAM on Data-lane[31:0]		
		0x6: reserved for future use		
		0x7: Not used for the OMAP2423 device		
8	SDRCTRISTATE	Static 3-state command for the SDRC I/O pads	RW	1
		0x0: All SDRC interface pins are set to high im-pedance		
		0x1: Normal mode: SDRC drives the I/O pads based on the memory traffic		
7:0	Reserved	Write 0s for future compatibility Read returns 0	RW	0x00

Table 13−93. SDRC_ERR_ADDR

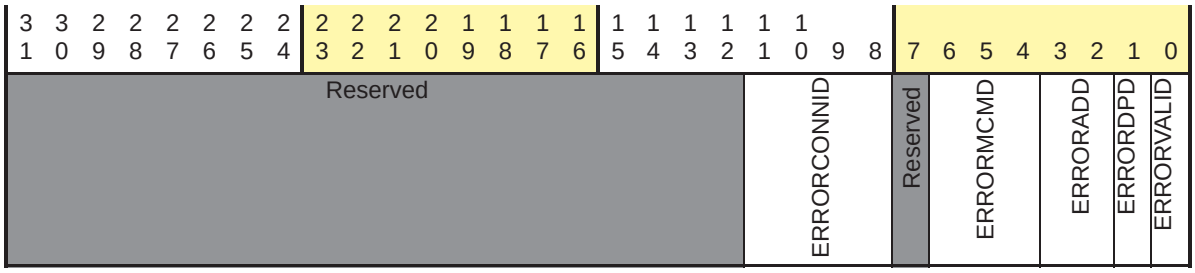
Address Offset	0x0048		
Physical Address	0x6800 9048	**Instance**	SDRC1
Description	This register captures the address of the last illegal access received on the OCP interface		
Type	R		
Write Latency	Not relevant		



Bits	Field Name	Description	Type	Reset
31:0	ERRORADDRESS	Address of illegal access	R	0x00000000

Table 13−94. SDRC_ERR_TYPE

Address Offset	0x004C		
Physical Address	0x6800 904C	**Instance**	SDRC1
Description	This register provides additional information about the last illegal access		
Type	RW		
Write Latency	Immediate		



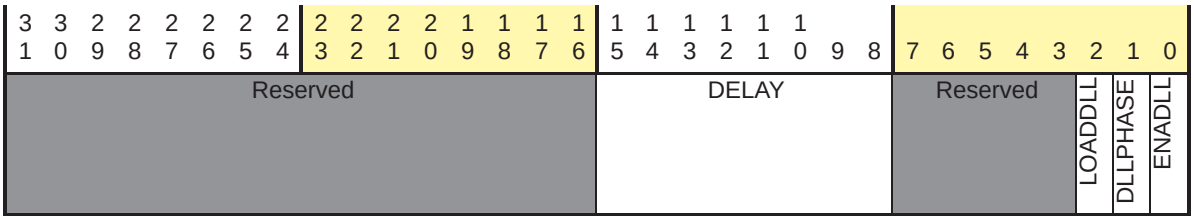
Bits	Field Name	Description	Type	Reset
31:12	Reserved	Write 0s for future compatibility Reads returns 0	RW	0x00000
11:8	ERRORCONNID	Identifies the illegal access initiator. OCP ConnID of the illegal access initiator: See the top level documentation for the device using the SDRC module.	R	0x0
7	Reserved	Write 0s for future compatibility Reads returns 0	RW	0
6:4	ERRORMCMD	System command of the transaction that caused the error (3-bit field)	R	0x0
3:2	ERRORADD	Flag that indicates access is to an illegal address 0x0: The System request was to an address outside the memory space (MAddrSpace =1) 0x1: The System request was to an address outside the register space (MAddrSpace =0) 0x2: No Err Add. Not an address error	R	0x2

Table 13−94. SDRC_ERR_TYPE (Continued)

Bits	Field Name	Description	Type	Reset
3:2 (cont'd)	ERRORADD (cont'd)	0x3: The PASR field was set to a value other than "All bank" The access was to a memory location that is not refreshed		
1	ERRORDPD	Transaction error while the memory is in deep power down mode 0x0: The memory was not in deep power down mode when the error occurred 0x1: The error is due to an unexpected access while the memory was in deep power down mode	R	0
0	ERRORVALID	Error validity status—Must be explicitly cleared with a write transaction 0x0: All error fields no longer valid 0x1: Error detected and logged in the other error fields	RW	0

Table 13−95. SDRC_DLLA_CTRL

Address Offset	0x0060		
Physical Address	0x6800 9060	**Instance**	SDRC1
Description	This register controls the SDRC DLL A resource, used for fine timing tuning on a double-data-rate interface		
Type	RW		
Write Latency	Immediate		



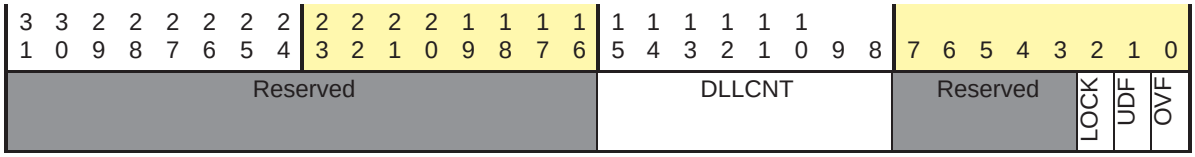
Bits	Field Name	Description	Type	Reset
31:16	Reserved	Write 0s for future compatibility Reads return zero	RW	0x0000
15:8	DELAY	8-bit delay; valid range 0..224 one step = 22ps ±8.5 ps Delay[1:0] must be set to 0x3	RW	0x00
7:3	Reserved	Write 0s for future compatibility Reads return 0	RW	0x00
2	LOADDLL	Loads delay field value into the DLL module, to initialize the tracking counter, and put the DLL in lock or unlock mode Delay field is loaded when a low to high transition occurs.	RW	0

Table 13−95. SDRC_DLLA_CTRL (Continued)

Bits	Field Name	Description	Type	Reset
2 (cont'd)	LOADDLL (cont'd)	0x0: Load DLL at 0 put the DLL in lock mode (tracking counter started) 0x1: Load DLL at 1 put the DLL in unlock mode (assert delay value permanently)		
1	DLLPHASE	Nominal digitally controlled delay when DLL is enabled 0x0: 72 degrees (20% of the clock period) 0x1: 90 degrees (25% of the clock cycle)—90 degrees must be used	RW	0
0	ENADLL	DLL Control 0x0: DLL disabled 0x1: DLL enabled	RW	0

Table 13−96. SDRC_DLLA_STATUS

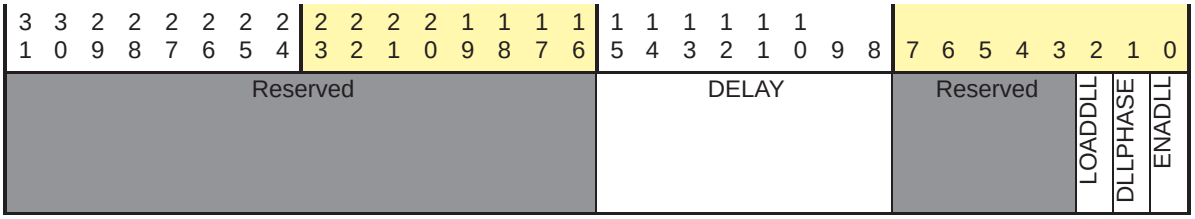
Address Offset	0x0064		
Physical Address	0x6800 9064	**Instance**	SDRC1
Description	This register reflects the current status of the DLL A		
Type	R		
Write Latency	Not relevant		



Bits	Field Name	Description	Type	Reset
31:16	Reserved	Reads return zero	R	0x0000
15:8	DLLCNT	Current DLL counter value for monitoring/debug	R	0x00
7:3	Reserved	Reads return zero	R	0x00
2	LOCK	DLL lock status—reserved for future use 0x0: The DLL is not locked 0x1: The DLL is locked	R	0
1	UDF	DLL counter underflow status 0x0: No underflow 0x1: The DLL counter has underflowed (<0)	R	0
0	OVF	DLL counter overflow status 0x0: No overflow 0x1: The DLL counter has overflowed (>224)	R	0

Table 13−97. SDRC_DLLB_CTRL

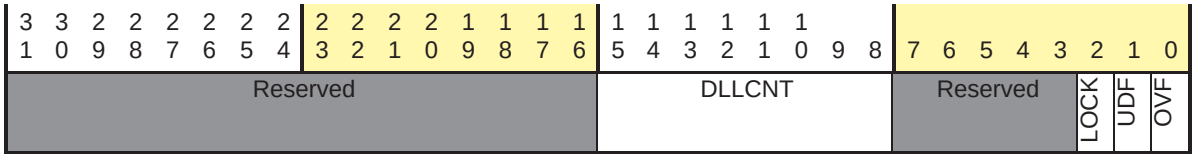
Address Offset	0x0068		
Physical Address	0x6800 9068	**Instance**	SDRC1
Description	This register controls the SDRC DLL B resource, used for fine timing tuning on a double-data-rate interface		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:16	Reserved	Write 0s for future compatibility Reads return 0	RW	0x0000
15:8	DELAY	8-bit delay; valid range 0..224 one step = 22ps 8.5 ps Delay[1:0] must be set to 0x3	RW	0x00
7:3	Reserved	Write 0s for future compatibility. Reads returns 0	RW	0x00
2	LOADDLL	Loads delay field value into the DLL module, to initialize the tracking counter, and put the DLL in lock or unlock mode. Delay field is loaded when a low to high transition occurs. 0x0: Load DLL at 0 put the DLL in lock mode (tracking counter started) 0x1: Load DLL at 1 put the DLL in unlock mode (assert delay value permanently)	RW	0
1	DLLPHASE	Nominal digitally controlled delay when DLL is enabled 0x0: 72 degrees (20% of the clock period) 0x1: 90 degrees (25% of the clock cycle)—90 degrees must be used	RW	0
0	ENADLL	DLL Control 0x0: DLL disabled 0x1: DLL enabled	RW	0

Table 13−98. SDRC_DLLB_STATUS

Address Offset	0x006C
Physical Address	0x6800 906C **Instance** SDRC1
Description	This register reflects the current status of the DLL B
Type	R
Write Latency	Not relevant



Bits	Field Name	Description	Type	Reset
31:16	Reserved	Reads return zero	R	0x0000
15:8	DLLCNT	Current DLL counter value for monitoring / debug	R	0x00
7:3	Reserved	Reads return zero	R	0x00
2	LOCK	DLL lock status—Reserved for future use.	R	0
		0x0: The DLL is not locked		
		0x1: The DLL is locked		
1	UDF	DLL counter underflow status	R	0
		0x0: No underflow		
		0x1: The DLL counter has underflowed (<0)		
0	OVF	DLL counter overflow status	R	0
		0x0: No overflow		
		0x1: The DLL counter has overflowed (>224)		

Table 13−99. SDRC_POWER

Address Offset	0x0070
Physical Address	0x6800 9070 **Instance** SDRC1
Description	The SDRC Power Management register defines the Global Power management policy (shared by CS0/CS1)
Type	RW
Write Latency	Immediate

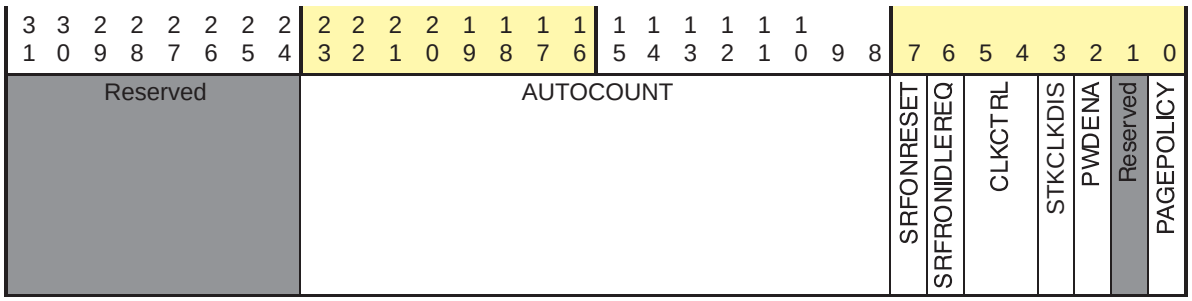
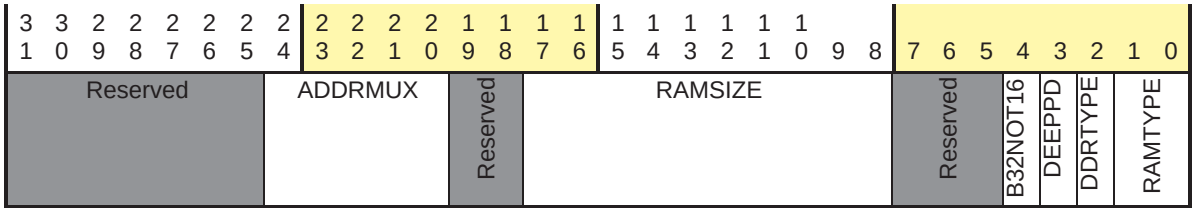


Table 13−99. SDRC_POWER (Continued)

Bits	Field Name	Description	Type	Reset
31:24	Reserved	Write 0s for future compatibility Reads returns 0	RW	0x00
23:8	AUTOCOUNT	16-bit programmable count value used for delayed automatic clock gating and self-refresh entry, assuming CLKCTRL field is not 0.	RW	0x0000
7	SRFRONRESET	Enter self-refresh when a warm reset is applied 0x0: Feature disabled 0x1: Feature enabled	RW	1
6	SRFRONIDLEREQ	Enter self-refresh when on hardware idle request. 0x0: Feature disabled 0x1: Feature enabled	RW	0
5:4	CLKCTRL	Clock control feature defines clock gating and self-refresh. 0x0: No auto clock feature turned on 0x1: Enable internal clock gating on timeout of AUTO_COUNT 0x2: Enable self-refresh on timeout of AUTO_COUNT 0x3: reserved	RW	0x0
3	STKCLKDIS	Disable the clock provided to the stacked memories 0x0: Enable clock 0x1: Disable clock—Logical 0 is applied.	RW	0
2	PWDENA	Activate the power down mode of the target memory, via CKE pin 0x0: Power down mode feature disabled 0x1: Power down mode feature enabled	RW	1
1	Reserved	Write 0s for future compatibility Reads returns 0s	RW	0
0	PAGEPOLICY	Page/segment closure policy with respect to power versus bandwidth trade-off—must be set to 1— 0x1: High power/High Bandwidth Mode (HPHB)	RW	1

Table 13−100. SDRC_MCFG_0

Address Offset	0x0080		
Physical Address	0x6800 9080	**Instance**	SDRC1
Description	This register provides the memory configuration register		
Type	RW		
Write Latency	Immediate		



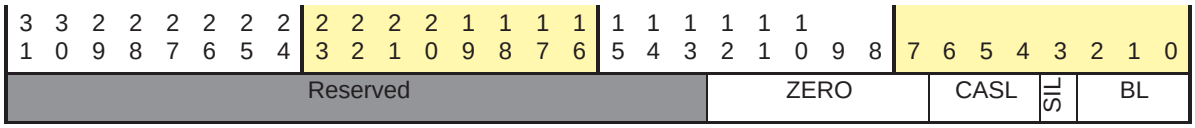
Bits	Field Name	Description	Type	Reset
31:25	Reserved	Write 0s for future compatibility. Reads Return 0s	RW	0x00
24:20	ADDRMUX	Address multiplexing scheme—See add muxing paragraph for details	RW	0x03
		0x0: Not used for the OMAP2423 device		
		0x1: Not used for the OMAP2423 device		
		0x2: Not used for the OMAP2423 device		
		0x3: Not used for the OMAP2423 device		
		0x4: Not used for the OMAP2423 device		
		0x5: Not used for the OMAP2423 device		
		0x6: Not used for the OMAP2423 device		
		0x7: Not used for the OMAP2423 device		
		0x8: Not used for the OMAP2423 device		
		0x9: Not used for the OMAP2423 device		
		0xA: Not used for the OMAP2423 device		
		0xB: Not used for the OMAP2423 device		
		0xC: Not used for the OMAP2423 device		
		0xD: Not used for the OMAP2423 device		
		0xE: Not used for the OMAP2423 device		
		0xF: Not used for the OMAP2423 device		
		0x10: Reserved		
		0x11: Reserved		
		0x12: Reserved		
		0x13: Reserved		

Table 13−100. SDRC_MCFG_0 (Continued)

Bits	Field Name	Description		Type	Reset
42:20 (cont'd)	ADDRMUX (cont'd)	0x14:	Reserved		
		0x15:	Reserved		
		0x16:	Reserved		
		0x17:	Not used for the OMAP2423 device		
		0x18:	Reserved		
		0x19:	Not used for the OMAP2423 device		
		0x1A:	Address mux configuration to be used for the OMAP2423 device		
		0x1B:	Not used for the OMAP2423 device		
		0x1C:	Not used for the OMAP2423 device		
		0x1D:	Undefined		
		0x1E:	Undefined		
		0x1F:	Undefined		
19:18	Reserved	Write 0s for future compatibility. Reads return 0s		RW	0x0
17:8	RAMSIZE	RAM Address space size Number of 2MB-chunks		RW	0x000
7:5	Reserved	Write 0s for future compatibility. Reads return 0s		RW	0x0
4	B32NOT16	Stacked SDRAM Bus width		RW	0
		0x0:	Not used for the OMAP2423 device		
		0x1:	Stacked SDRAM device is x32 bit		
3	DEEPPD	Indicates if the memory supports deep power down mode		RW	0
		0x0:	No Deep power Down mode support		
		0x1:	The memory supports Deep power Down mode		
2	DDRTYPE	DDR memory type (assuming RamType = 01)		RW	0
		0x0:	Mobile DDR		
		0x1:	reserved for future use		
1:0	RAMTYPE	Memory type		RW	0x0
		0x0:	Not used for the OMAP2423 device		
		0x1:	DDR-SDRAM (double data rate)		
		0x2:	reserved		
		0x3:	reserved		

Table 13−101. SDRC_MR_0

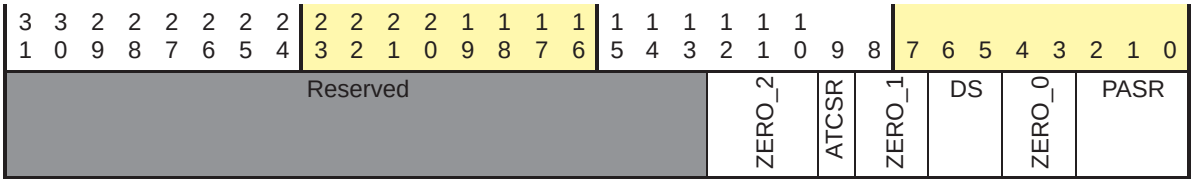
Address Offset	0x0084		
Physical Address	0x6800 9084	**Instance**	SDRC1
Description	This register corresponds to the stacked DDR SDRAM MR register, with the standard bit fields. It is a 13-bit register, all 13 bits are loaded into the memory. The SDRC keeps an internal copy register used internally that is returned when a read access is performed at that address. Load into Memory on OCP write access, using MRS command with BA1,BA0 = 0,0		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:13	Reserved	Write 0s for future compatibility Read returns 0	RW	0x00000
12:7	ZERO	Write 0s, as required by memory specifications Read returns 0	RW	0x00
6:4	CASL	CAS latency as defined in clock periods	RW	0x2
		0x1: Not used for the OMAP2423 device		
		0x2: Not used for the OMAP2423 device		
		0x3: CAS latency = 3		
		0x4: Not used for the OMAP2423 device		
		0x5: Not used for the OMAP2423 device		
3	SIL	Serial or interleaved mode : must be zero	RW	0
		0x0: Serial mode (always used)		
		0x1: interleaved mode (not supported)		
2:0	BL	Memory burst length	RW	0x4
		0x0: Burst length = 1 (not supported)		
		0x1: Burst length = 2 (not supported)		
		0x2: Burst length = 4 (used for the OMAP2423 device)		
		0x3: Burst length = 8 (not supported)		
		0x4: reserved		
		0x5: reserved		
		0x6: reserved		
		0x7: Full page (not supported)		

Table 13−103. SDRC_EMR2_0

Address Offset	0x008C		
Physical Address	0x6800 908C	**Instance**	SDRC1
Description	This register corresponds to the stacked DDR SDRAM EMR register. It is a 13-bit register, all 13 bits are loaded into the memory. The SDRC keeps an internal copy register, used internally, that is returned when a read access is performed at that address. Load into memory on OCP write access, using MRS command with BA1,BA0 = 1,0		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:13	Reserved	Write 0s for future compatibility Read returns 0	RW	0x00000
12:10	ZERO_2	Write 0s, as required by memory specifications Read returns 0	RW	0x0
9	ATCSR	Advanced temperature compensated self-refresh enable bit 0x0: Enabled 0x1: Reserved (ATCSR disabled not supported)	RW	0
8:7	ZERO_1	Write 0s, as required by memory specifications Read returns 0	RW	0x0
6:5	DS	Driver strength 0x0: Normal strength driver 0x1: 1/2 strength driver 0x2: 1/4 strength driver 0x3: 1/8 strength driver	RW	0x0
4:3	ZERO_0	Write 0s, as required by memory specifications Read returns 0	RW	0x0
2:0	PASR	Partial array self-refresh 0x0: All banks. 0x1: 1/2 array 0x2: 1/4 array 0x3: Reserved 0x4: Reserved 0x5: Reserved	RW	0x0

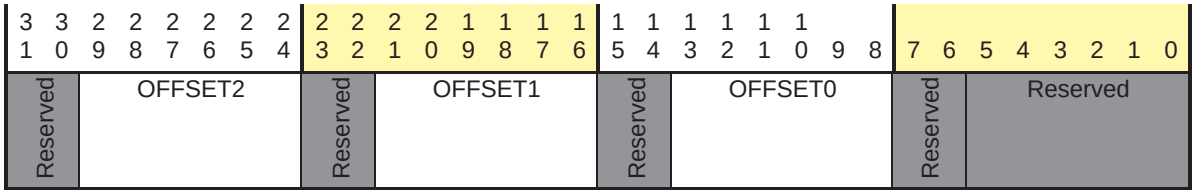
Table 13−103. SDRC_EMR2_0 (Continued)

Bits	Field Name	Description	Type	Reset
2:0 (cont'd)	PASR (cont'd)	0x6: Reserved 0x7: Reserved		

Table 13−104 does not apply to this device.

Table 13−105. SDRC_DCDL1_CTRL

Address Offset	0x094		
Physical Address	0x6800 9094	**Instance**	SDRC1
Description	This register controls all DCDL delay adjustments.		
Type	RW		



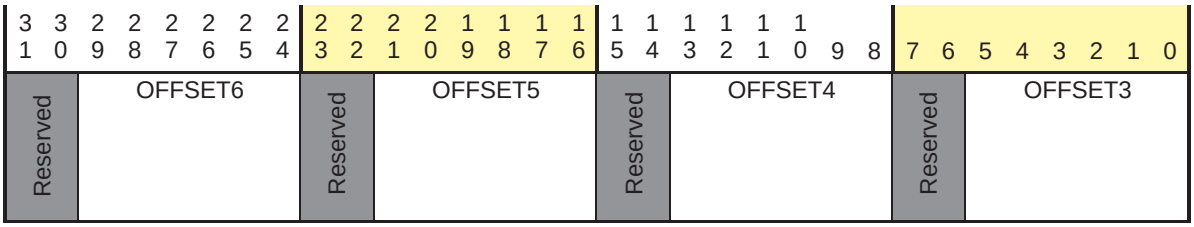
Bits	Field Name	Description	Type	Reset
31:30	Reserved	Write 0s for future compatibility. Reads return 0s	RW	0x0
29:24	OFFSET2	One MSB bit defines when OFFSET2 must be added or subtracted from the DQS2 nominal value for DQS2 adjustment in read mode. 0x0: DQS2/DCDL is subtracted 0x1: DQS2/DCDL is added DQS2/DCDL offset adjustment (step 22 ps) 5 lower bits for unsigned value 0x1F: +31 0x1E: +30 ... 0x01: +1 0x00: 0 0x20: 0 0x21: –1 ... 0x3E: –30 0x3F: –31	RW	0x0
23:22	Reserved	Write 0s for future compatibility. Read returns 0s.	RW	0x0

Table 13−105. SDRC_DCDL1_CTRL (Continued)

Bits	Field Name	Description	Type	Reset
21:16	OFFSET1	Not used for the OMAP2423 device	RW	0x00
15:14	Reserved	Write 0s for future compatibiilty. Reads return 0s	RW	0x0
13:8	OFFSET0	Not used for the OMAP2423 device	RW	0x00
7:6	Reserved	Write 0s for future compatibiilty. Reads return 0s	RW	0x0
5:0	Reserved	Write 0s for future compatibiilty. Reads return 0s	RW	0x00

Table 13−106. SDRC_DCDL2_CTRL

Address Offset	0x098		
Physical Address	0x6800 9098	**Instance**	SDRC1
Description	This register controls all DCDL delay adjustments.		
Type	RW		



Bits	Field Name	Description	Type	Reset
31:30	Reserved	Write 0s for future compatibility. Reads return 0s	RW	0x0
29:24	OFFSET6	One MSB bit defines if OFFSET6 must be added or subtracted from DQS nominal value for DQS adjustment in write mode. 0x0: DQS/DCDL is subtracted 0x1: DQS/DCDL is added Write DCDL offset adjustment (step 22 ps) 5 lower bits for unsigned value 0x1F: +31 0x1E: +30 ... 0x01: +1 0x00: 0 0x20: 0 0x21: –1 ... 0x3E: –30	RW	0x00

Table 13−106. SDRC_DCDL2_CTRL (Continued)

Bits	Field Name	Description	Type	Reset
29:24 (cont'd)	OFFSET6 (cont'd)	0x3F: –31		
23:22	Reserved	Write 0s for future compatibility. Reads return 0s	RW	0x0
21:16	OFFSET5	One MSB bit defines if OFFSET5 must be added or subtracted from DQS5 nominal value for DQS5 adjustment in read mode. 0x0: DQS5/DCDL is subtracted 0x1: DQS5/DCDL is added DQS5/DCDL offset adjustment (step 22 ps) 5 lower bits for unsigned value 0x1F: +31 0x1E: +30 ... 0x01: +1 0x00: 0 0x20: 0 0x21: –1 ... 0x3E: –30 0x3F: –31	RW	0x00
15:14	Reserved	Write 0s for future compatibility. Reads return 0s	RW	0x0
13:8	OFFSET4	One MSB bit defines if OFFSET4 must be added or subtracted from DQS4 nominal value for DQS4 adjustment in read mode. 0x0: DQS4/DCDL is subtracted 0x1: DQS4/DCDL is added DQS4/DCDL offset adjustment (step 22 ps) 5 lower bits for unsigned value 0x1F: +31 0x1E: +30 ... 0x01: +1 0x00: 0	RW	0x00

Table 13−106. SDRC_DCDL2_CTRL (Continued)

Bits	Field Name	Description	Type	Reset
13:8 (cont'd)	OFFSET4 (cont'd)	0x20: 0 0x21: –1 ... 0x3E: –30 0x3F: –31		
7:6	Reserved	Write 0s for future compatibility. Reads return 0s	RW	0x0
5:0	OFFSET3	One MSB bit defines if OFFSET3 must be added or subtracted from DQS3 nominal value for DQS3 adjustment in read mode. 0x0: DQS3/DCDL is subtracted 0x1: DQS3/DCDL is added DQS3/DCDL offset adjustment (step 22 ps) 5 lower bits for unsigned value 0x1F: +31 0x1E: +30 ... 0x01: +1 0x00: 0 0x20: 0 0x21: –1 ... 0x3E: –30 0x3F: –31	RW	0x00

Table 13−107. SDRC_ACTIM_CTRLA_0

Address Offset	0x009C		
Physical Address	0x6800 909C	**Instance**	SDRC1
Description	The ac parameter timing control register sets the ac parameters values in clock cycle unit, in order to best match the memory characteristics		
Type	RW		
Write Latency	Immediate		

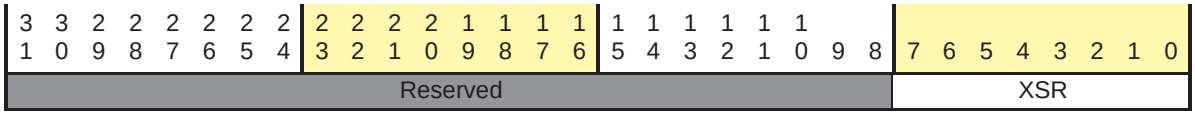
31–29	28–25	24–20	19–16	15–12	11–8	7–4	3–0
TRFC	TRC	TRAS	TRP	TRCD	TRRD	TDPL	Reserved / TDAL

Table 13−107. SDRC_ACTIM_CTRLA_0 (Continued)

Bits	Field Name	Description	Type	Reset
31:27	TRFC	Autorefresh to active	RW	0x00
26:22	TRC	Row Cycle time	RW	0x00
21:18	TRAS	Row active time	RW	0x0
17:15	TRP	Row Precharge time	RW	0x0
14:12	TRCD	Row to column delay time	RW	0x0
11:9	TRRD	Active to active command period.	RW	0x0
8:6	TDPL	Data-in to precharge command (write recovery time tWR)	RW	0x0
5	Reserved	Write 0s for future compatibility Reads return 0	RW	0
4:0	TDAL	Data-in to active command	RW	0x00

Table 13−108. SDRC_ACTIM_CTRLB_0

Address Offset	0x00A0		
Physical Address	0x6800 90A0	**Instance**	SDRC1
Description	The XSR parameter timing control register sets the ac parameters values in clock cycle unit, in order to best match the memory characteristics		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:8	Reserved	Write 0s for future compatibility Reads return zero	RW	0x000000
7:0	XSR	Self-refresh Exit to active period	RW	0x00

Table 13−109. SDRC_RFR_CTRL_0

Address Offset	0x00A4		
Physical Address	0x6800 90A4	**Instance**	SDRC1
Description	SDRAM memory auto-refresh control		
Type	RW		
Write Latency	Immediate		

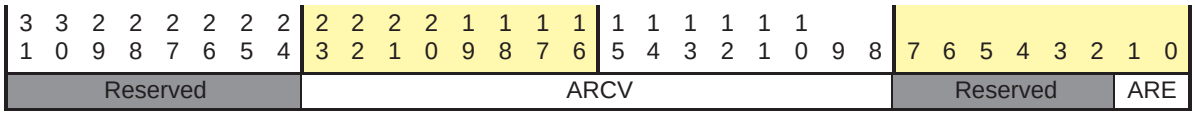


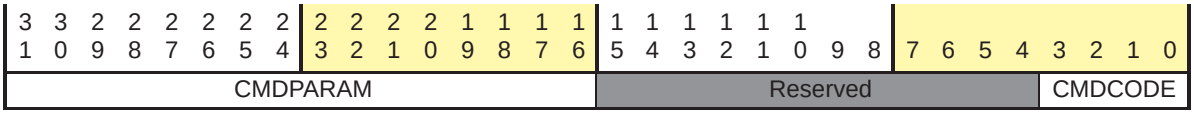
Table 13−109. SDRC_RFR_CTRL_0 (Continued)

Bits	Field Name	Description	Type	Reset
31:24	Reserved	Write 0s for future compatibility. Reads return 0s	RW	0x00
23:8	ARCV	Auto-refresh counter value, to set the refresh period > The formula is: (Refresh Interval†/clock period)−50.	RW	0x0000
7:2	Reserved	Write 0s for future compatibility. Reads return 0s	RW	0x00
1:0	ARE	Auto-refresh enable	RW	0x0
		0x0: Auto-refresh is disabled		
		0x1: Counter is loaded with ARCV: 1 auto-refresh command when auto-refresh counter reaches 0		
		0x2: Counter is loaded with 4xARCV: Burst of 4 auto-refresh commands when auto-refresh counter reaches 0		
		0x3: Counter is loaded with 8xARCV: Burst of 8 auto-refresh commands when auto-refresh counter reaches 0		

† Refresh interval in time units

Table 13−110. SDRC_MANUAL_0

Address Offset	0x00A8		
Physical Address	0x6800 90A8	**Instance**	SDRC1
Description	This register allows sending specific commands to the stacked memory devices under software control. Any write to this register generates the appropriate sequence, based on the command code		
Type	RW		
Write Latency	Immediate		



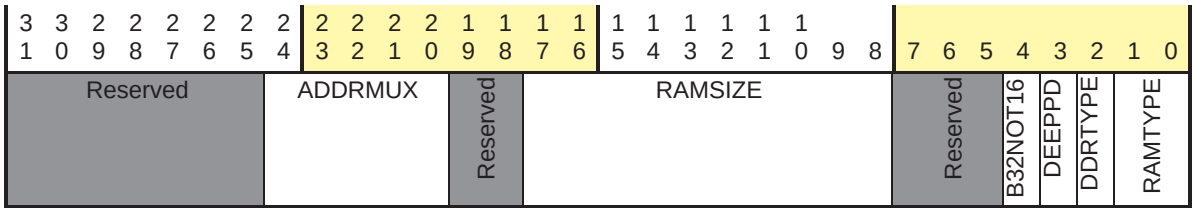
Bits	Field Name	Description	Type	Reset
31:16	CMDPARAM	Manual command parameter, if any	RW	0x0000
15:4	Reserved	Write 0s for future compatibility. Reads returns 0s	RW	0x000
3:0	CMDCODE	Memory command opcode; Other values: Reserved for future implementations	RW	0x0
		0x0: NOP command—No parameter		
		0x1: Precharge All command—No parameter		
		0x2: Autorefresh command—No parameter		
		0x3: Enter deep power down—No parameter		
		0x4: Exit deep power down—No parameter		

Table 13−110. SDRC_MANUAL_0 (Continued)

Bits	Field Name	Description		Type	Reset
3:0 (cont'd)	CMDCODE (cont'd)	0x5:	Enter self-refresh—No parameter		
		0x6:	Exit self-refresh—No parameter		
		0x7:	Set CKE signal high—No parameter		
		0x8:	Set CKE low—No parameter		
		0x9:	reserved		
		0xA:	reserved		
		0xB:	reserved		
		0xC:	reserved		

Table 13−111. SDRC_MCFG_1

Address Offset	0x00B0		
Physical Address	0x6800 90B0	**Instance**	SDRC1
Description	This register provides the memory configuration register		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description		Type	Reset
31:25	Reserved	Write 0s for future compatibility. Reads Return 0s		RW	0x00
24:20	ADDRMUX	Address multiplexing Scheme—See add muxing paragraph for details		RW	0x03
		0x0:	Not used for the OMAP2423 device		
		0x1:	Not used for the OMAP2423 device		
		0x2:	Not used for the OMAP2423 device		
		0x3:	Not used for the OMAP2423 device		
		0x4:	Not used for the OMAP2423 device		
		0x5:	Not used for the OMAP2423 device		
		0x6:	Not used for the OMAP2423 device		
		0x7:	Not used for the OMAP2423 device		
		0x8:	Address mux configuration to be used for the OMAP2423 device		

Table 13−111. SDRC_MCFG_1 (Continued)

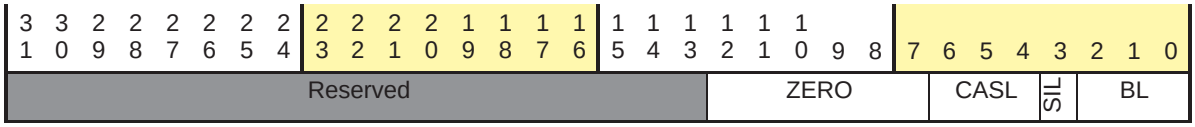
Bits	Field Name	Description		Type	Reset
24:20 (cont'd)	ADDRMUX (cont'd)	0x9:	Reserved		
		0xA:	Reserved		
		0xB:	Reserved		
		0xC:	Reserved		
		0xD:	Reserved		
		0xE:	Reserved		
		0xF:	Not used for the OMAP2423 device		
		0x10:	Not used for the OMAP2423 device		
		0x11:	Not used for the OMAP2423 device		
		0x12:	Not used for the OMAP2423 device		
		0x13:	Not used for the OMAP2423 device		
		0x14:	Not used for the OMAP2423 device		
		0x15:	Not used for the OMAP2423 device		
		0x16:	Reserved		
		0x17:	Not used for the OMAP2423 device		
		0x18:	Reserved		
		0x19:	Not used for the OMAP2423 device		
		0x1A:	Not used for the OMAP2423 device		
		0x1B:	Not used for the OMAP2423 device		
		0x1C:	Not used for the OMAP2423 device		
		0x1D:	Undefined		
		0x1E:	Undefined		
		0x1F:	Undefined		
19:18	Reserved	Write 0s for future compatibility. Reads return 0s		RW	0x0
17:8	RAMSIZE	RAM address space size Number of 2MB-chunks		RW	0x000
7:5	Reserved	Write 0s for future compatibility. Reads return 0s		RW	0x0
4	B32NOT16	Stacked SDRAM Bus width		RW	0
		0x0:	Not used for the OMAP2423 device		
		0x1:	Stacked SDRAM device is x32 bit		

Table 13−111. SDRC_MCFG_1 (Continued)

Bits	Field Name	Description	Type	Reset
3	DEEPPD	Indicates if the memory supports deep power down mode 0x0: No deep power-down mode support 0x1: The memory supports deep power-down mode	RW	0
2	DDRTYPE	DDR memory type (assuming RamType = 01) 0x0: Mobile DDR 0x1: reserved for future use	RW	0
1:0	RAMTYPE	Memory type 0x0: Not used for the OMAP2423 device 0x1: DDR-SDRAM (double data rate) 0x2: reserved 0x3: reserved	RW	0x0

Table 13−112. SDRC_MR_1

Address Offset	0x00B4
Physical Address	0x6800 90B4 **Instance** SDRC1
Description	This register corresponds to the stacked DDR SDRAM MR register, with the standard bit fields. It is a 13-bit register, all 13 bits are loaded into the memory. The SDRC keeps an internal copy register used internally that is returned when a read access is performed at that address. Load into Memory on OCP write access, using MRS command with BA1,BA0 = 0,0
Type	RW
Write Latency	Immediate



Bits	Field Name	Description	Type	Reset
31:13	Reserved	Write 0s for future compatibility. Read returns 0	RW	0x00000
12:7	ZERO	Write 0s, as required by memory specifications Read returns 0	RW	0x00
6:4	CASL	CAS latency as defined in clock periods 0x1: Not used for the OMAP2423 device 0x2: Not used for the OMAP2423 device 0x3: CAS latency = 3	RW	0x2

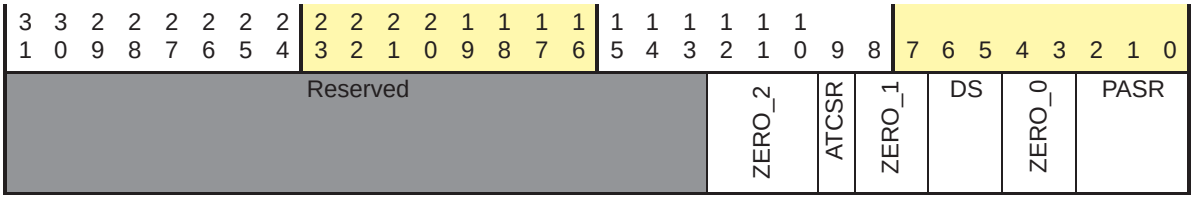
Table 13−112. SDRC_MR_1 (Continued)

Bits	Field Name	Description	Type	Reset
6:4 (cont'd)	CASL (cont'd)	0x4: Not used for the OMAP2423 device		
		0x5: Not used for the OMAP2423 device		
3	SIL	Serial or interleaved mode: must be zero	RW	0
		0x0: Serial mode (always used)		
		0x1: interleaved mode (not supported)		
2:0	BL	Memory burst length	RW	0x4
		0x0: Burst length = 1 (not supported)		
		0x1: Burst length = 2 (not supported)		
		0x2: Burst length = 4 (used for the OMAP2423 device)		
		0x3: Burst length = 8 (not supported)		
		0x4: reserved		
		0x5: reserved		
		0x6: reserved		
		0x7: Full page (not supported)		

Table 13−113 does not apply to this device.

Table 13−114. SDRC_EMR2_1

Address Offset	0x00BC		
Physical Address	0x6800 90BC	**Instance**	SDRC1
Description	This register corresponds to the stacked DDR SDRAM EMR register. It is a 13-bit register, all 13 bits are loaded into the memory. The SDRC keeps an internal copy register, used internally, that is returned when a read access is performed at that address. Load into memory on OCP write access, using MRS command with BA1,BA0 = 1,0		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:13	Reserved	Write 0s for future compatibility Read returns 0	RW	0x00000
12:10	ZERO_2	Write 0s, as required by memory specifications Read returns 0	RW	0x0

Table 13−114. SDRC_EMR2_1 (Continued)

Bits	Field Name	Description	Type	Reset
9	ATCSR	Advanced temperature compensated self-refresh enable bit 0x0: Enabled 0x1: Reserved (ATCSR disabled not supported)	RW	0
8:7	ZERO_1	Write 0s, as required by memory specifications Read returns 0	RW	0x0
6:5	DS	Driver strength 0x0: Normal strength driver 0x1: 1/2 strength driver 0x2: 1/4 strength driver 0x3: 1/8 strength driver	RW	0x0
4:3	ZERO_0	Write 0s, as required by memory specifications Read returns 0	RW	0x0
2:0	PASR	Partial array self-refresh 0x0: All banks. 0x1: 1/2 array 0x2: 1/4 array 0x3: Reserved 0x4: Reserved 0x5: Reserved 0x6: Reserved 0x7: Reserved	RW	0x0

Table 13−115 does not apply to this device.

Table 13−116. SDRC_ACTIM_CTRLA_1

Address Offset	0x00C4
Physical Address	0x6800 90C4 **Instance** SDRC1
Description	The ac parameter timing control register sets the ac parameters values in clock cycle unit, in order to best match the memory characteristics
Type	RW
Write Latency	Immediate

31 30	29 28 27	26 25 24	23 22 21 20	19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
TRFC	TRC	TRAS	TRP	TRCD	TRRD	TDPL	Reserved	TDAL

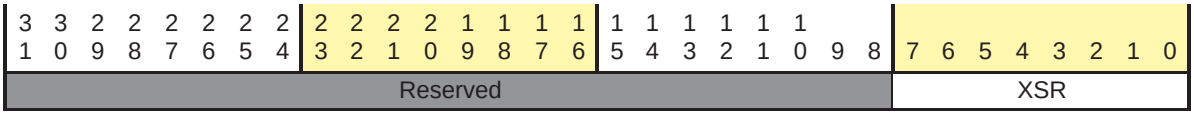
Bits	Field Name	Description	Type	Reset
31:27	TRFC	Autorefresh to active	RW	0x00
26:22	TRC	Row Cycle time	RW	0x00

Table 13−116. SDRC_ACTIM_CTRLA_1 (Continued)

Bits	Field Name	Description	Type	Reset
21:18	TRAS	Row active time	RW	0x0
17:15	TRP	Row Precharge time	RW	0x0
14:12	TRCD	Row to column delay time	RW	0x0
11:9	TRRD	Active to active command period.	RW	0x0
8:6	TDPL	Data-in to precharge command (write recovery time tWR)	RW	0x0
5	Reserved	Write 0s for future compatibility Reads return 0	RW	0
4:0	TDAL	Data-in to active command	RW	0x00

Table 13−117. SDRC_ACTIM_CTRLB_1

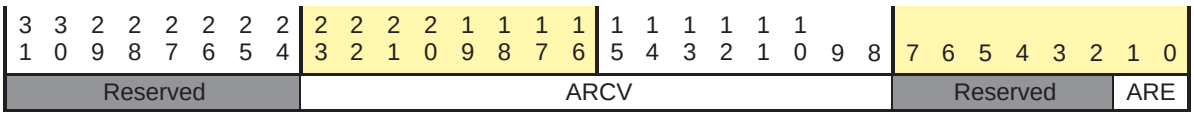
Address Offset	0x00C8		
Physical Address	0x6800 90C8	**Instance**	SDRC1
Description	The XSR parameter timing control register sets the ac parameters values in clock cycle unit, in order to best match the memory characteristics		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:8	Reserved	Write 0s for future compatibility Reads return zero	RW	0x000000
7:0	XSR	Self-refresh Exit to active period	RW	0x00

Table 13−118. SDRC_RFR_CTRL_1

Address Offset	0x00D4		
Physical Address	0x6800 90D4	**Instance**	SDRC1
Description	SDRAM memory auto-refresh control		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:24	Reserved	Write 0s for future compatibility. Reads return 0s.	RW	0x00
23:8	ARCV	Auto-refresh counter value, to set the refresh period > because the formula is: (Refresh Interval† /clock period)−50.	RW	0x0000

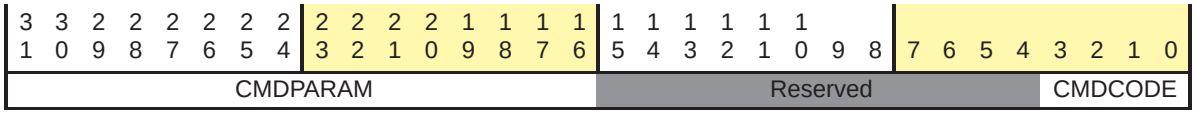
Table 13−118. SDRC_RFR_CTRL_1 (Continued)

Bits	Field Name	Description	Type	Reset
7:2	Reserved	Write 0s for future compatibility. Reads return 0s	RW	0x00
1:0	ARE	Auto-refresh enable	RW	0x0
		0x0: Auto-refresh is disabled		
		0x1: Counter is loaded with ARCV: 1 auto-refresh command when auto-refresh counter reaches 0		
		0x2: Counter is loaded with 4xARCV: Burst of 4 auto-refresh commands when auto-refresh counter reaches 0		
		0x3: Counter is loaded with 8xARCV: Burst of 8 auto-refresh commands when auto-refresh counter reaches 0		

† Refresh interval in time units

Table 13−119. SDRC_MANUAL_1

Address Offset	0x00D8		
Physical Address	0x6800 90D8	**Instance**	SDRC1
Description	This register allows to send specific commands to the stacked memory devices under software control. Any write to this register generates the appropriate sequence, based on the command code		
Type	RW		
Write Latency	Immediate		



Bits	Field Name	Description	Type	Reset
31:16	CMDPARAM	Manual command parameter, if any	RW	0x0000
15:4	Reserved	Write 0s for future compatibility. Reads returns 0s	RW	0x000
3:0	CMDCODE	Memory command opcode; Other values: Reserved for future implementation	RW	0x0
		0x0: NOP command—No parameter		
		0x1: Precharge All command—No parameter		
		0x2: Autorefresh command—No parameter		
		0x3: Enter deep power down—No parameter		
		0x4: Exit deep power down—No parameter		
		0x5: Enter self-refresh—No parameter		
		0x6: Exit self-refresh—No parameter		

Table 13−119. SDRC_MANUAL_1 (Continued)

Bits	Field Name	Description		Type	Reset
3:0 (cont'd)	CMDCODE (cont'd)	0x7:	Set CKE signal high—No parameter		
		0x8:	Set CKE low—No parameter		
		0x9:	Reserved		
		0xA:	Reserved		
		0xB:	Reserved		
		0xC:	Reserved		

12.13 OCM Subsystem Overview

See Section 12.13 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

12.14 OCM Subsystem Integration

See Section 12.14 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

12.15 OCM Subsystem Functional Description

See Section 12.15 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 13

Enhanced Audio Controller

This chapter describes the enhanced audio controller (EAC) device of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 13 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 14

Camera Subsystem

This chapter describes the camera subsystem used as the system interface and functionality to connect image sensor modules to the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 14 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 15

2D/3D Graphics Accelerator

This chapter describes the 2D/3D graphics accelerator (GFX) of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 15 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 16

IVA Subsystem

This chapter describes the image video accelerator (IVA) subsystem of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 16 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 17

Display Subsystem

This chapter describes the display subsystem of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 17 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 18

Timers

This chapter discusses several types of timers used by system software on the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 18 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 19

UART/IrDA/CIR

This chapter discusses the three UART devices of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 19 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 20

I²C

This chapter describes the I²C modules of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 20 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 21

VLYNQ

This chapter discusses the VLYNQ™ serial communications interface, which enables the extension of the internal L3 interconnect in the OMAP2423 multimedia system-in-package (SIP) processor to one or more external physical devices.

This chapter is equivalent to Chapter 21 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 22

Synchronous Serial Interface

This chapter describes the synchronous serial interface (SSI) controller module of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 22 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 23

Multichannel SPI

This chapter describes the two multichannel SPI (McSPI) modules of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 23 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 24

HDQ/1-Wire

This chapter describes the features and functions of the HDQ/1-Wire module on the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 24 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 25

Multichannel Buffered Serial Port

This chapter discusses the multichannel buffered serial port (McBSP) of the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 25 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 26

MMC/SDIO Card Interface

This chapter describes the features and functions of the multimedia card/secure digital I/O (MMC/SDIO) card interface on the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 26 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 27

Memory Stick PRO Host Controller

This chapter describes the Memory Stick PRO™ host controller in the OMAP2420 multimedia device.

This chapter is equivalent to Chapter 27 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 28

Universal Serial Bus

This chapter describes the three USB ports and the several varieties of USB functionality aavailable with the OMAP2423 multimedia system-in-package (SIP) processor.

This chapter is equivalent to Chapter 28 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 29

General-Purpose Interface

This chapter discusses the four general-purpose input/output (GPIO) modules that form the general-purpose interface in the OMAP2423 multimedia system-in-package (SIP) device.

Topic **Page**

29.1 General-Purpose Interface Functional Overview 29-2
29.2 General-Purpose Interface Environment 29-5
29.3 General-Purpose Interface Integration 29-7
29.4 General-Purpose Interface Functional Description 29-17
29.5 General-Purpose Interface Programming Model 29-17
29.6 General-Purpose Interface Registers 29-17

29.1 General-Purpose Interface Functional Overview

The OMAP2423 general-purpose interface is the combination of four general-purpose input/output (GPIO) modules.

Each GPIO module controls 32 dedicated general-purpose pins with input and output capabilities. Therefore, the general-purpose interface provides support for 128 (4 x 32) pins.

These pins can be configured to be used for the following applications:

- Data input (capture)/output (drive)
- Keyboard interface with a debounce (protection circuitry) cell
- Interrupt generation in active mode (module is running synchronously with the interface clock) upon the detection of external events. Detected events are processed by two parallel independent interrupt-generation submodules to support biprocessor operations.
- Wake-up request generation in idle mode (power saving mode; the interface clock is not present) upon the detection of external events

The four GPIO modules, GPIO1 through GPIO4, present a total theoretical capability of 128 GPIO pins at the OMAP2423 device level. These four GPIO modules are regrouped into the *quad-GPIO* module, which connects to a single target agent port on the L4 interconnect.

The module does not include pad control (pull up/down control, open-drain feature), which is part of the chip top-level configuration.).

29.1.1 Global Features

Four operating modes are defined for the GPIO module (see *Power Management* in Section 29.3.1.1):

- Active mode: The module runs synchronously with the provided clock; synchronous interrupts can be generated.
- Idle mode: The interface clock is not present; a wake-up request can be generated.
- Inactive mode: Same as idle mode, but wake-up request cannot be generated.
- Disabled mode: The module has no activity; internal clock paths are gated.

The GPIO modules include the following global features:

- Synchronous interrupt requests in active mode from each channel are processed by two identical interrupt generation submodules to be used independently by the DSP and the MPU subsystems. One of these interrupts is mapped on the DSP subsystem interrupt controller and the other on the MPU subsystem interrupt controller.
- Asynchronous wake-up requests in idle mode from input channels are merged together to issue one wake-up signal.

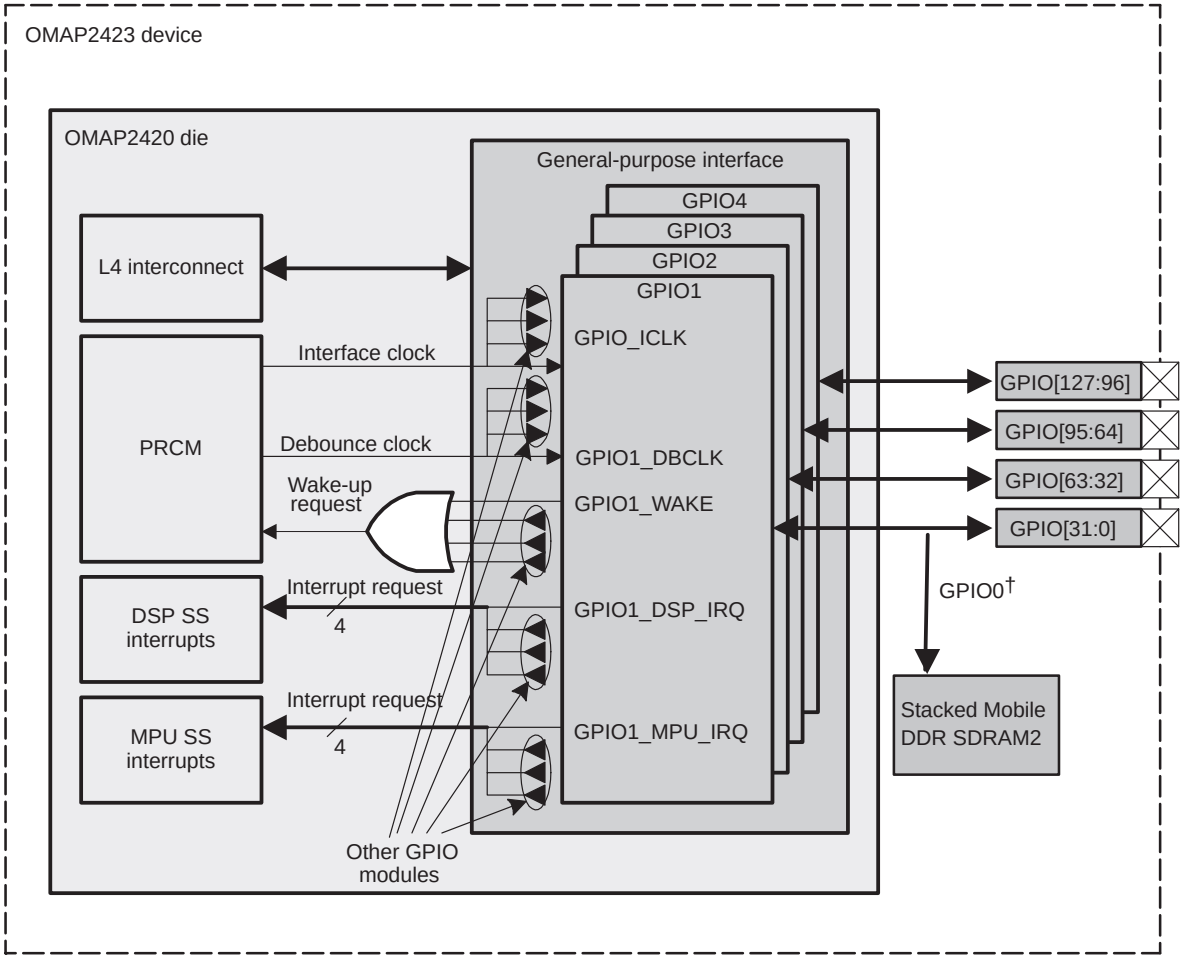
- Data input (capture)/output (drive)

There are 8 interrupt lines (2 interrupt lines per GPIO module instance) in the general-purpose interface.

All wake-up signals (one signal per GPIO module) are internally ORed together in the OMAP2423 device to produce a unique wake-up request to the PRCM module.

Figure 29−1 shows the general-purpose interface.

Figure 29−1. General-Purpose Interface Highlight



† See Section 12.9.3

Each channel in the GPIO modules has the following features:

- The output enable register (GPIO_OE) controls the output capability for each pin.
- The output line level reflects the value written in the data output register (GPIO_DATAOUT) through the L4 interconnect.
- The input line can be fed to the GPIO module through an optional and configurable debounce cell. (The value is global for all ports of one GPIO module, so you can have up to four different debouncing values.)

- The input line value is sampled into the data input register (GPIO_DATAIN) and can be read through the L4 interconnect.
- In active mode, the input line can be used through level and edge detectors to trigger synchronous interrupts. The edge (rising, falling, or both) or the level (logical 0, logical 1, or both) to be used can be configured.
- In idle mode, the input line can be used to activate the asynchronous wake-up request (on edge detection: rising edge, falling edge, or both).

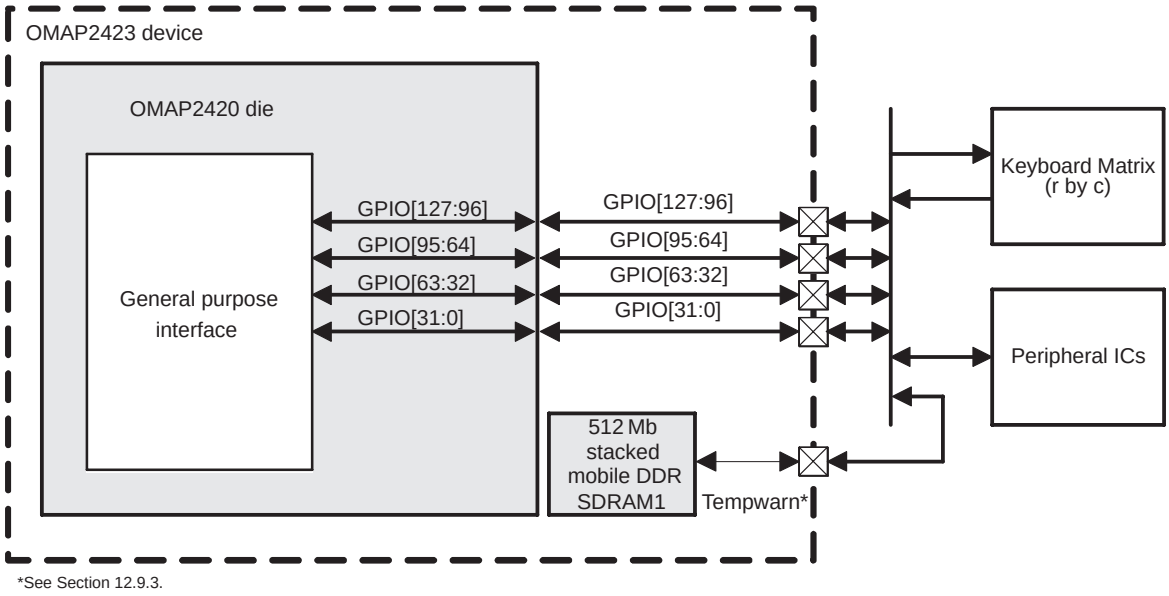
The module provides an alternative to the atomic test and set operations for the data output (GPIO_DATAOUT), interrupt enable (GPIO_IRQENABLE1 and GPIO_IRQENABLE2), and wake-up enable (GPIO_WAKEUPENABLE) registers. For these registers, the modules implement the set and clear protocol register update.

29.2 General-Purpose Interface Environment

The general-purpose interface, the combination of four GPIO modules, is a flexible, user-programmable, general-purpose I/O controller. It implements functions that are not implemented with the dedicated controllers in the OMAP2423 system and that require simple input and/or output software controlled signals. The general-purpose interface allows a variety of custom connections and expands the I/O capabilities of the system to the real world.

Figure 29−2 shows a typical application using the general-purpose interface.

Figure 29−2. General-Purpose Interface Typical Application System Overview



The general-purpose interface physically connects the OMAP2423 device to a keyboard matrix and peripheral ICs.

- GPIO as a keyboard interface

  The general-purpose interface can be used as a keyboard interface. Users can dedicate channels according to the keyboard matrix size (r × c). In the following example (Figure 29−3), row channels are configured as input with the input debounce feature enabled and are driven high with an external pullup; column channels are configured as output and drive a low level.

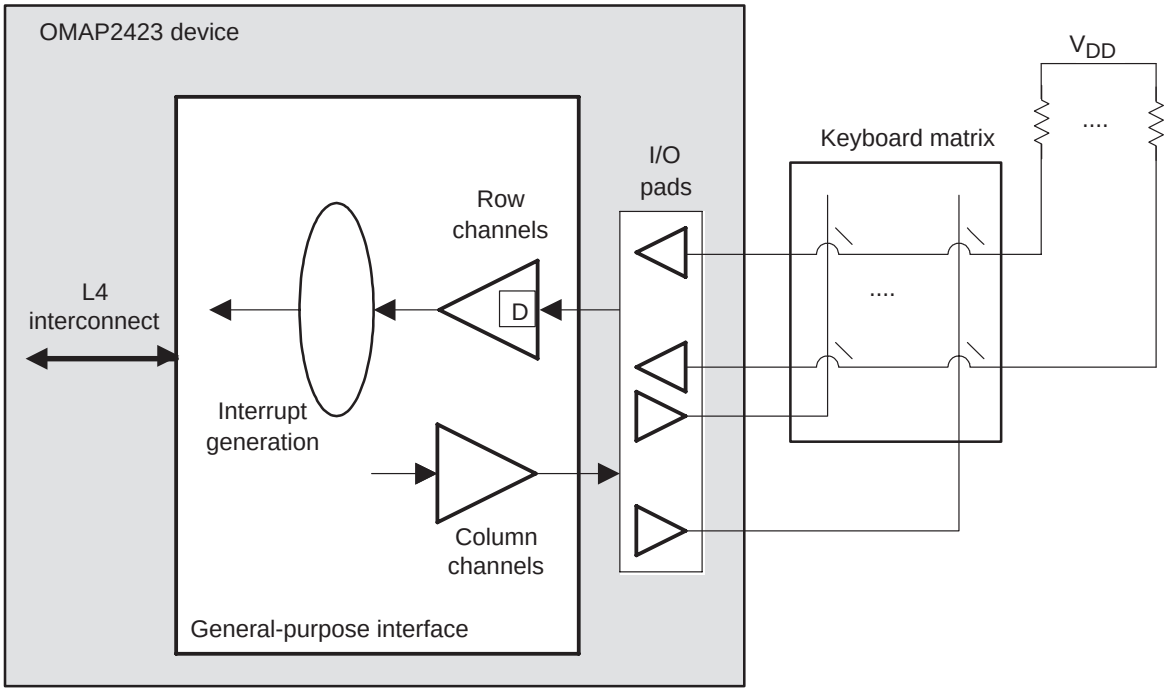
  As soon as any key of the keyboard matrix is pressed, the corresponding row and column lines are shorted together and a low level is driven on the corresponding row channel, generating an interrupt according to the proper configuration (see Section 29.5.3). When the keyboard interrupt is received, the processor (MPU subsystem and/or DSP subsystem in OMAP2423) can disable the keyboard interrupt and scan the column channels to get the key coordinates.

The scanning sequence has as many states as column channels:

- For each step in the sequence, the processor drives one column channel low while the others are set to high.
- The processor reads the values of the row channels and detects which keys in the column are pressed.

At the end of the scanning sequence, the processor establishes which keys are pressed. The keyboard interface can then be reconfigured in the interrupt waiting state.

Figure 29−3. General-Purpose Interface Used as a Keyboard Interface



29.2.1 General-Purpose Interface Functional Interfaces

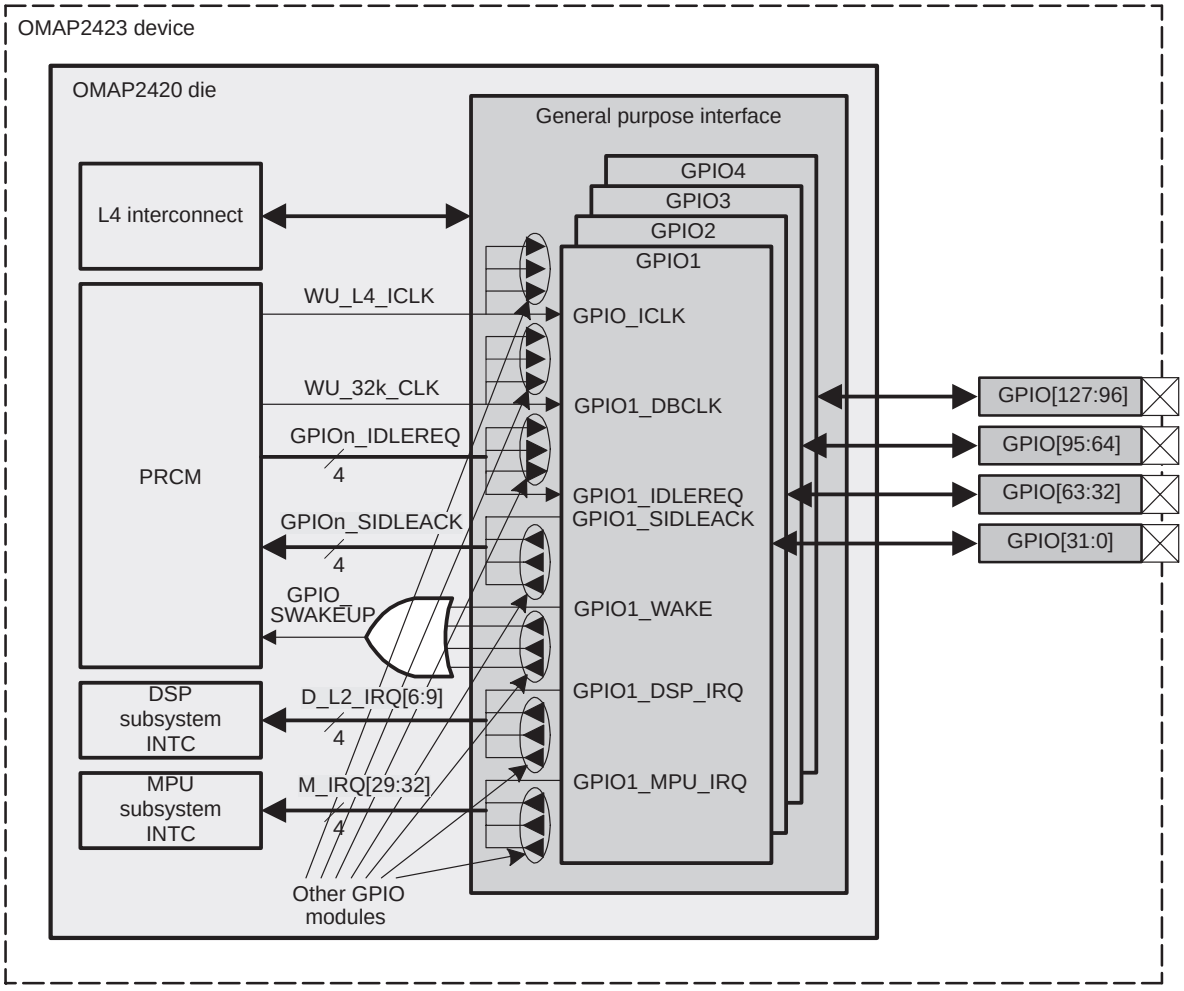
See Section 29.2.1 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

29.3 General-Purpose Interface Integration

29.3.1 Description

Figure 29−5 highlights the general-purpose interface integration in the OMAP2423 device. Figure 29−4

Figure 29−5. General-Purpose Interface Typical Application System Overview



29.3.1.1 Clocking, Reset, and Power Management Scheme

See Section 29.3.1.1 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

29.3.1.2 Hardware Requests

Interrupts Requests

All interrupt sources (the 32 input GPIO channels) are merged to issue two synchronous interrupt requests in each GPIO module. Thus, the general-purpose interface has eight interrupt lines (two interrupt lines per GPIO module instance).

The synchronous interrupt request line 1 is mapped on the MPU interrupt controller. The synchronous interrupt request line 2 is mapped on the DSP interrupt controller.

The synchronous interrupt request lines 1 and 2 are active according to their respective interrupt enable 1 and 2 registers (GPIO_IRQENABLE1 or GPIO_IRQENABLE2).

Table 30−3 lists the interrupt lines that are driven out from the general-purpose interface to the MPU subsystem and DSP subsystem interrupt controllers.

Table 30−3. Interrupts

Name	Mapping	Comments
GPIO1 Module (GPIO[31:0])		
GPIO1_MPU_IRQ	M_IRQ_29	Destination is MPU subsystem interrupt controller.
GPIO1_DSP_IRQ	D_L2_IRQ_6	Destination is DSP subsystem interrupt controller.
GPIO2 Module (GPIO[63:32])		
GPIO2_MPU_IRQ	M_IRQ_30	Destination is MPU subsystem interrupt controller.
GPIO2_DSP_IRQ	D_L2_IRQ_7	Destination is DSP subsystem interrupt controller.
GPIO3 Module (GPIO[95:64])		
GPIO3_MPU_IRQ	M_IRQ_31	Destination is MPU subsystem interrupt controller.
GPIO3_DSP_IRQ	D_L2_IRQ_8	Destination is DSP subsystem interrupt controller.
GPIO4 Module (GPIO[127:96])		
GPIO4_MPU_IRQ	M_IRQ_32	Destination is MPU subsystem interrupt controller.
GPIO4_DSP_IRQ	D_L2_IRQ_9	Destination is DSP subsystem interrupt controller.

Wake-up Generation

The general-purpose interface is attached to the WKUP power domain (see Chapter 5) and can wake up the system.

All wake-up sources (the 32 input GPIO channels) are merged together to issue a single asynchronous wake-up request in each GPIO module following the expected transition(s) (according to the registers programming). Also, all wake-up signals (one signal per GPIO module) are internally ORed together in the OMAP2423 device to produce a unique wake-up request (GPIO_SWAKEUP) to the PRCM module.

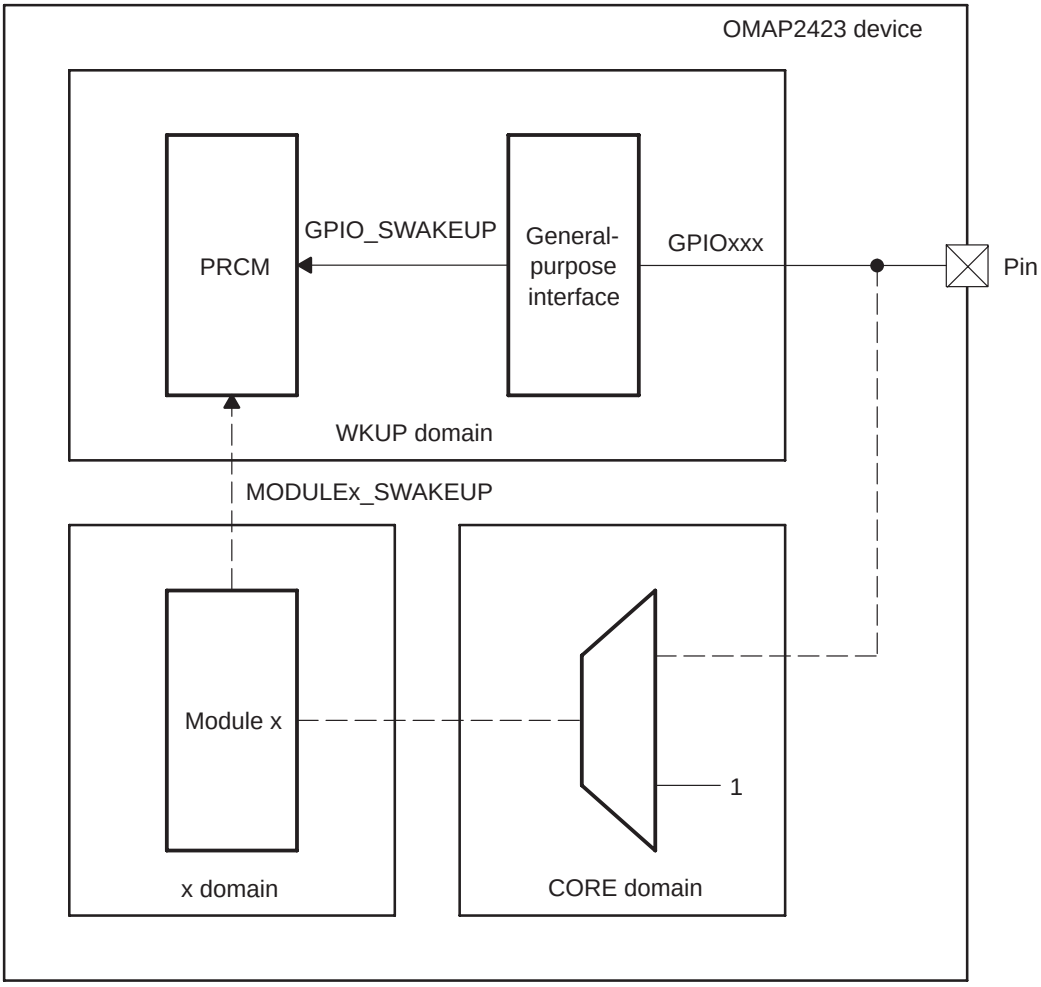
The asynchronous wake-up request line is active according to the wake-up enable register (GPIO_WAKEUPENABLE).

Table 30−4. Wake-up Signal

Name	Mapping	Comments
GPIOn_WAKE (where *n* is 1, 2 ,3 and 4)	GPIO_SWAKEUP	Destination is PRCM module

Figure 29−6 shows a wake-up description in the OMAP2423 device.

Figure 29−6. Wake-up Description



The module *x* can be any OMAP2423 module that does not belong to the WKUP domain (for example, the UART module in the core domain). This module can have a wake-up feature but cannot wake up the system on an external event when its core (*x)* domain is off. The multiplexers used for the pad multiplexing in the OMAP2423 device belong to the core domain (see Chapter 5).

Therefore, when all of the power domains are off (excluding the WKUP power domain, which is continuously active; see Chapter 5), the pin is directly connected to the general-purpose interface. If the GPIO channel muxed on

this pin is configured as a wake-up source, a wake-up request is asserted to the system.

Note:

The wake-up capability works whether the GPIO function of the pin is selected by the pinout configuration or not.

Note:

Some GPIO channels cannot be used as wake-up sources in the OMAP2423 device.

All GPIO channels are muxed with other device pins, but the wake-up feature cannot work if the muxes are inserted between the pin and the GPIO module. One GPIO channel corresponds to several pins.

Three GPIO channels (GPIO.29, GPIO.30, and GPIO.31) have only an output capability in the OMAP2423 device and cannot be used as wake-up sources.

Table 30−5 lists the GPIO channels without the wake-up feature.

The GPIO_IRQENABLE1, GPIO_IRQENABLE2, and GPIO_WAKEUPENABLE register bits for GPIO pins without wake-up capability must be disabled before the core domain enters the RETENTION or OFF mode, then enabled after wake-up, if required. Otherwise, spurious wake-up events are generated by isolation cell between the core and wake-up domains.

Table 30−5. GPIO Channels Without the Wake-up Feature

GPIO Number	Alternate Signals			Comments
GPIO.6	GPMC.A7	GPIO6		
GPIO.7	GPMC.A6	MMC.DAT_DIR0		
GPIO.8	GPMC.A5	UART1.RTS	MMC.CMD_DIR	
GPIO.9	GPMC.A4	UART1.TX		
GPIO.10	GPMC.A3	UART1.RX		
GPIO.11	GPMC.A2	McBSP2.DR		
GPIO.12	GPMC.A1	McBSP2.CLKX		
GPIO.13	GPMC.D15	VLYNQ.CLK		
GPIO.14	GPMC.D14	VLYNQ.RX1		
GPIO.15	GPMC.D13	VLYNQ.RX0		
GPIO.16	GPMC.D12	VLYNQ.TX1		
GPIO.17	GPMC.D11	VLYNQ.TX0		
GPIO.25	GPMC.nCS4	SSI1.FLAG_TX		
GPIO.29	GPMC.nBE0			GPO only, no input

Table 30−5. GPIO Channels Without the Wake-up Feature (Continued)

GPIO Number	Alternate Signals			Comments
GPIO.30	GPMC.nBE1			GPO only, no input
GPIO.31	GPMC.nWP			GPO only, no input
GPIO.38	DDS.D8			
GPIO.52	CAM.D2	SYS.CLKREQ		
GPIO.53	CAM.D1	CAM.D9		
GPIO.54	CAM.D0	CAM.D8		
GPIO.59	SSI1.DAT_TX	MMC.CLKI		

Idle Mode Request/Acknowledge Feature

Two idle control signals allow each GPIO module to enter different idle modes and save power.

Table 30−6 lists the four GPIOn_IDLEREQ signals from the PRCM module and four GPIOn_SIDLEACK signals to the PRCM module (where *n* is 1, 2, 3, and 4).

Table 30−6. Idle Mode Request/Acknowledge Signals

Name	Mapping	Comments
GPIO1 Module (GPIO[31:0])		
GPIO1_IDLEREQ	GPIO1_IDLEREQ	Source is PRCM module.
GPIO1_SIDLEACK	GPIO1_SIDLEACK	Destination is PRCM module.
GPIO2 Module (GPIO[63:32])		
GPIO2_IDLEREQ	GPIO2_IDLEREQ	Source is PRCM module.
GPIO2_SIDLEACK	GPIO2_SIDLEACK	Destination is PRCM module.
GPIO3 Module (GPIO[95:64])		
GPIO3_IDLEREQ	GPIO3_IDLEREQ	Source is PRCM module.
GPIO3_SIDLEACK	GPIO3_SIDLEACK	Destination is PRCM module.
GPIO4 Module (GPIO[127:96])		
GPIO4_IDLEREQ	GPIO4_IDLEREQ	Source is PRCM module.
GPIO4_SIDLEACK	GPIO4_SIDLEACK	Destination is PRCM module.

29.3.1.3 Pin List and Pad Multiplexing With Other Functions

Table 30−7 describes the pin and pad multiplexing options for the general purpose interface.

All the GPIO channels are available on Mode 3 (indicated in table by gray shading) and a few channels are duplicated on the same pin in Mode 0 (white cells show alternate pin functions).

CAUTION

The internal connections between OMAP2420 die and DDR SDRAM stacked memories prohibit using GPIO.1, GPIO.2, GPIO.37, and GPIO.38 as GPIO signals on the customer board. At most, they can be used as debug pins to see specific SDRC signals.

Table 30−7. Pin Multiplexing for the General-Purpose Interface

GPIO	Description	DIR	Ball 2423	Alternate Functions					
Interface				Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.0	General-purpose I/O 0	I/O	AA26						
GPIO.1	General-purpose I/O 1	I/O	W25						
GPIO.2	General-purpose I/O 2	I/O	W26						
GPIO.3	General-purpose I/O 3	I/O	D5	gpmc.a10		sys.ndma-req5			
GPIO.4	General-purpose I/O 4	I/O	AE18	gpmc.a9		sys.ndma-req4			
GPIO.5	General-purpose I/O 5	I/O	J7	gpmc.a8		sys.ndma-req3			
GPIO.6	General-purpose I/O 6	I/O	D6		tv.detect		gpio.6		
			E4	gpmc.a7		sys.ndma-req2			
			D6	gpio.6	tv.detect				
GPIO.7	General-purpose I/O 7	I/O	AF9	gpmc.a6	dss.d23				
			F23	mmc. dat_dir0	ms.dat0_dir				
GPIO.8	General-purpose I/O 8	I/O	AE16	gpmc.a5	dss.d22				
			K20	uart1.rts		dss.d19			
			J24	mmc. cmd_dir					
GPIO.9	General-purpose I/O 9	I/O	W8	gpmc.a4	dss.d21				
			H12	uart1.tx		dss.d20			
GPIO.10	General-purpose I/O 10	I/O	AF10	gpmc.a3	dss.d20				
			T20	uart1.rx		dss.d21			
GPIO.11	General-purpose I/O 11	I/O	W9	gpmc.a2	dss.d19				
			P20	mcbsp2.dr		dss.d22			
GPIO.12	General-purpose I/O 12	I/O	M8	gpmc.a1	dss.d18				
			T24	mcbsp2. clkx		dss.d23			
GPIO.13	General-purpose I/O 13	I/O	H1	gpmc.d15	ssi2.dat_tx				
			W12	vlynq.clk	usb2.se0	sys.ndma-req0			

Table 30−7. Pin Multiplexing for the General-Purpose Interface (Continued)

GPIO	Description	DIR	Ball 2423	Alternate Functions					
Interface				Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.14	General-purpose I/O 14	I/O	H2	gpmc.d14	ssi2.flag_tx				
			W10	vlynq.rx1	usb2.rcv	sys.ndma-req1			
GPIO.15	General-purpose I/O 15	I/O	P2	gpmc.d13	ssi2.rdy_tx				
			AD7	vlynq.rx0	usb2.tllse0				
GPIO.16	General-purpose I/O 16	I/O	U1	gpmc.d12	ssi2.dat_rx				
			W14	vlynq.tx1	usb2.dat	sys.clkout2			
GPIO.17	General-purpose I/O 17	I/O	U2	gpmc.d11	ssi2.flag_rx				
			Y15	vlynq.tx0	usb2.txen				
GPIO.18	General-purpose I/O 18	I/O	T1	gpmc.d10	ssi2.rdy_rx				
GPIO.19	General-purpose I/O 19	I/O	Y1	gpmc.d9	ssi2.wake				
GPIO.20	General-purpose I/O 20	I/O	V1	gpmc.d8					
GPIO.21	General-purpose I/O 21	I/O	P1	gpmc.clk					
GPIO.22	General-purpose I/O 22	I/O	AF14	gpmc.ncs1					
GPIO.23	General-purpose I/O 23	I/O	G4	gpmc.ncs2					
GPIO.24	General-purpose I/O 24	I/O	T8	gpmc.ncs3	gpmc.io_dir				
GPIO.25	General-purpose I/O 25	I/O	H8	gpmc.ncs4					
			AC15	ssi1.flag_tx	uart1.rts	usb1.rcv			
GPIO.26	General-purpose I/O 26	I/O	K3	gpmc.ncs5					
GPIO.27	General-purpose I/O 27	I/O	M7	gpmc.ncs6					
GPIO.28	General-purpose I/O 28	I/O	P3	gpmc.ncs7	gpmc.io_dir				
GPIO.29	General-purpose I/O 29	O	AF12	gpmc.nbe0					
GPIO.30	General-purpose I/O 30	O	U3	gpmc.nbe1					
GPIO.31	General-purpose I/O 31	O	AE15	gpmc.nwp					
GPIO.32	General-purpose I/O 32	I/O	G20	uart1.cts		dss.d18			
GPIO.33	General-purpose I/O 33	I/O	AE20	gpmc.wait1					
GPIO.34	General-purpose I/O 34	I/O	N2	gpmc.wait2					
GPIO.35	General-purpose I/O 35	I/O	T4	gpmc.wait3					
GPIO.36	General-purpose I/O 36	I/O	Y18				gpio.36		sys. boot4
			Y18	gpio.36					sys. boot4
GPIO.37	General-purpose I/O 37	I/O	Y25						
GPIO.38	General-purpose I/O 38	I/O	AE25						
			AD11	dss.d8					
GPIO.39	General-purpose I/O 39	I/O	AD12	dss.d9					

Table 30−7. Pin Multiplexing for the General-Purpose Interface (Continued)

GPIO	Description	DIR	Ball 2423	Alternate Functions					
Interface				Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.40	General-purpose I/O 40	I/O	AC12	dss.d10					
GPIO.41	General-purpose I/O 41	I/O	AE11	dss.d11					
GPIO.42	General-purpose I/O 42	I/O	AE13	dss.d12					
GPIO.43	General-purpose I/O 43	I/O	AD13	dss.d13					
GPIO.44	General-purpose I/O 44	I/O	AC13	dss.d14					
GPIO.45	General-purpose I/O 45	I/O	Y12	dss.d15					
GPIO.46	General-purpose I/O 46	I/O	AD14	dss.d16					
GPIO.47	General-purpose I/O 47	I/O	Y13	dss.d17					
GPIO.48	General-purpose I/O 48	I/O	AE8	dss.acbias		mcbsp2.fsx			
GPIO.49	General-purpose I/O 49	I/O	AD3	cam.d5	hw.dbg7	mcbsp1.clkr			
GPIO.50	General-purpose I/O 50	I/O	V2	cam.d4	hw.dbg6	mcbsp1.fsr			
GPIO.51	General-purpose I/O 51	I/O	AB3	cam.d3	hw.dbg5	mcbsp1.dr			
GPIO.52	General-purpose I/O 52	I/O	U7	cam.d2	hw.dbg4	mcbsp1.clkx			
		I	AD20	sys.clkreq					
GPIO.53	General-purpose I/O 53	I/O	AC7	cam.d9	hw.dbg11				
			Y3	cam.d1	hw.dbg3	sti.din			
GPIO.54	General-purpose I/O 54	I/O	AC6	cam.d8	hw.dbg10				
			Y4	cam.d0	hw.dbg2	sti.dout			
GPIO.55	General-purpose I/O 55	I/O	V4	cam.hs	hw.dbg1	mcbsp1.dx			
GPIO.56	General-purpose I/O 56	I/O	P7	cam.vs	hw.dbg0	mcbsp1.fsx			
GPIO.57	General-purpose I/O 57	I/O	AD6	cam.lclk		mcbsp.clks			
GPIO.58	General-purpose I/O 58	I/O	AE10	vlynq_nla		l4_ext_trig			
GPIO.59	General-purpose I/O 59	I/O	W13	ssi1.dat_tx	uart1.tx	usb1.se0			
			C23	mmc.clki	ms.clki				
GPIO.60	General-purpose I/O 60	I	Y20	sys_nirq					
GPIO.61	General-purpose I/O 61	I/O	AF15	ssi1.rdy_tx	uart1.cts	usb1.txen			
GPIO.62	General-purpose I/O 62	I/O	AD18		uart1.rx	usb1.dat	gpio.62		
			AD18	gpio.62	uart1.rx	usb1.dat			
GPIO.63	General-purpose I/O 63	I/O	W15	ssi1.dat_rx	eac.md_sclk				
GPIO.64	General-purpose I/O 64	I/O	AF11	ssi1.flag_rx	eac.md_din				
GPIO.65	General-purpose I/O 65	I/O	AC16	ssi1.rdy_rx	eac. md_dout				
GPIO.66	General-purpose I/O 66	I/O	AD15	ssi1.wake	eac.md_fs				
GPIO.67	General-purpose I/O 67	I/O	AC24	uart2.cts	usb1.rcv	gpt9. pwm/evt			

Table 30−7. Pin Multiplexing for the General-Purpose Interface (Continued)

GPIO	Description	DIR	Ball 2423	Alternate Functions					
Interface				Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.68	General-purpose I/O 68	I/O	W20	uart2.rts	usb1.txen	gpt10. pwm/evt			
GPIO.69	General-purpose I/O 69	I/O	AB24	uart2.tx	usb1.se0	gpt11. pwm/evt			
GPIO.70	General-purpose I/O 70	I/O	AD24	uart2.rx	usb1.dat	gpt12. pwm/evt			
GPIO.71	General-purpose I/O 71	I/O	AD4	eac.bt_sclk					
GPIO.72	General-purpose I/O 72	I/O	W7	eac.bt_fs					
GPIO.73	General-purpose I/O 73	I/O	U8	eac.bt_din					
GPIO.74	General-purpose I/O 74	I/O	AD5	eac.bt_dout		sti.clk			
GPIO.75	General-purpose I/O 75	I/O	G19	mmc.dat1	ms.dat1				
GPIO.76	General-purpose I/O 76	I/O	H20	mmc.dat2	ms.dat2	uart2.cts			
GPIO.77	General-purpose I/O 77	I/O	D24	mmc.dat3	ms.dat3	l4.ext_trig			
GPIO.78	General-purpose I/O 78	I/O	D23	mmc. dat_dir1	ms.datu_dir	uart2.rts			
GPIO.79	General-purpose I/O 79	I/O	G23	mmc. dat_dir2	ms.datu_dir	uart2.tx			
GPIO.80	General-purpose I/O 80	I/O	E23	mmc. dat_dir3	ms.datu_dir	uart2.rx			
GPIO.81	General-purpose I/O 81	I/O	Y23	spi1.clk					
GPIO.82	General-purpose I/O 82	I/O	H10	spi1.simo					
GPIO.83	General-purpose I/O 83	I/O	V19	spi1.somi					
GPIO.84	General-purpose I/O 84	I/O	W24	spi1.ncs0					
GPIO.85	General-purpose I/O 85	I/O	W23	spi1.ncs1					
GPIO.86	General-purpose I/O 86	I/O	V23	spi1.ncs2					
GPIO.87	General-purpose I/O 87	I/O	U20	spi1.ncs3					
GPIO.88	General-purpose I/O 88	I/O	V24	spi2.clk					
GPIO.89	General-purpose I/O 89	I/O	U24	spi2.simo	gpt10. pwm/evt				
GPIO.90	General-purpose I/O 90	I/O	V24	spi2.somi	gpt11. pwm/evt				
GPIO.91	General-purpose I/O 91	I/O	AA24	spi2.ncs0	gpt12. pwm/evt				
GPIO.92	General-purpose I/O 92	I/O	Y24	mcbsp1. clkr	ssi2.dat_tx	vlynq.tx1			
GPIO.93	General-purpose I/O 93	I/O	R20	mcbsp1.fsr	ssi2.flag_tx	vlynq.tx0			
GPIO.94	General-purpose I/O 94	I/O	R24	mcbsp1.dx	ssi2.rdy_tx	vlynq.clk			
GPIO.95	General-purpose I/O 95	I/O	U23	mcbsp1.dr	ssi2.dat_rx	vlynq.rx1			

Table 30−7. Pin Multiplexing for the General-Purpose Interface (Continued)

GPIO	Description	DIR	Ball 2423	Alternate Functions					
Interface				Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.96	General-purpose I/O 96	I/O	P23	mcbsp.clks	ssi2.flag_rx	vlynq.rx0			
GPIO.97	General-purpose I/O 97	I/O	R23	mcbsp1.fsx	ssi2.rdy_rx				
GPIO.98	General-purpose I/O 98	I/O	T19	mcbsp1. clkx	ssi2.wake	vlynq.nla			
GPIO.99	General-purpose I/O 99	I/O	L19	i2c2.scl		gpt9. pwm/evt			
GPIO.100	General-purpose I/O 100	I/O	K19	i2c2.sda		spi2.ncs1			
GPIO.101	General-purpose I/O 101	I/O	T23	hdq.sio	usb2.tllse0	sys.altclk			
GPIO.102	General-purpose I/O 102	I/O	K24	uart3.cts/ rctx	uart3.rx/irrx				
GPIO.103	General-purpose I/O 103	I/O	M20	uart3.rts/sd	uart3.tx/irtx				
GPIO.104	General-purpose I/O 104	I/O	G24	uart3.tx/irtx	uart3.rctx				
GPIO.105	General-purpose I/O 105	I/O	H24	uart3.rx/irrx					
GPIO.106	General-purpose I/O 106	I/O	L23	usb0.puen	mcbsp2.dx				
GPIO.107	General-purpose I/O 107	I/O	M23	usb0.vp	mcbsp2.dr				
GPIO.108	General-purpose I/O 108	I/O	N23	usb0.vm	mcbsp2. clkx				
GPIO.109	General-purpose I/O 109	I/O	K23	usb0.rcv	mcbsp2.fsx				
GPIO.110	General-purpose I/O 110	I/O	M24	usb0.txen	uart3.cts/ rctx	uart2.cts			
GPIO.111	General-purpose I/O 111	I/O	L24	usb0.se0	uart3.tx/irtx	uart2.tx			
GPIO.112	General-purpose I/O 112	I/O	J25	usb0.dat	uart3.rx/irrx	uart2.rx			
GPIO.113	General-purpose I/O 113	I/O	AC18	eac.ac_sclk	mcbsp2.clkx				
GPIO.114	General-purpose I/O 114	I/O	AD16	eac.ac_fs	mcbsp2.fsx				
GPIO.115	General-purpose I/O 115	I/O	AD19	eac.ac_din	mcbsp2.dr				
GPIO.116	General-purpose I/O 116	I/O	AF22	eac.ac_dout	mcbsp2.dx				
GPIO.117	General-purpose I/O 117	I/O	Y17	eac.ac_mclk					
GPIO.118	General-purpose I/O 118	I/O	AE22	eac.ac_rst	eac.bt_din	loop_eac. bt_din			
GPIO.119	General-purpose I/O 119	I/O	AF6				gpio. 119		sys. boot0
			AF6	gpio.119					sys. boot0
GPIO.120	General-purpose I/O 120	I/O	AF4				gpio. 120		sys. boot1
			AF4	gpio.120					sys. boot1

Table 30−7. Pin Multiplexing for the General-Purpose Interface (Continued)

GPIO	Description	DIR	Ball 2423	Alternate Functions					
Interface				Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.121	General-purpose I/O 121	I/O	AE6				gpio.121	jtag.emu2	sys.boot2
			AE6	gpio.121				jtag.emu2	sys.boot2
GPIO.122	General-purpose I/O 122	I/O	W3				gpio.122	jtag.emu3	sys.boot3
			W3	gpio.122				jtag.emu3	sys.boot3
GPIO.123	General-purpose I/O 123	I/O	AE19	sys.clkout					
GPIO.124	General-purpose I/O 124	I/O	Y19				gpio.124		sys.boot5
			Y19	gpio.124					sys.boot5
GPIO.125	General-purpose I/O 125	I/O	AE24		sys.jtagsel1	sys.jtagsel2	gpio.125		
			AE24	gpio.125	sys.jtagsel1	sys.jtagsel2			
GPIO.126	General-purpose I/O 126	I/O	AC22						
GPIO.127	General-purpose I/O 127	I/O	N24						

29.3.1.4 General-Purpose Interface Register Summary

See Section 29.3.1.4 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

29.4 General-Purpose Interface Functional Description

See Section 29.4 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

29.5 General-Purpose Interface Programming Model

See Section 29.5 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

29.6 General-Purpose Interface Registers

See Section 29.6 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 30

Pinout Overview

This chapter provides an overview of the pinout for the OMAP2423 multimedia system-in-package (SIP) processor.

Topic **Page**

30.1 Pinout Overview 30-2
30.2 Features 30-2
30.3 Functional Description 30-2
30.4 Pinout Control Integration 30-3
30.5 Pinout Environment 30-27
30.6 Programming Model 30-62
30.7 Pinout Control Module Registers 30-62
30.8 Acronyms, Abbreviations, and Definitions 30-62

30.1 Pinout Overview

See Section 30.1 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

30.2 Features

See Section 30.2 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

30.3 Functional Description

See Section 30.3 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

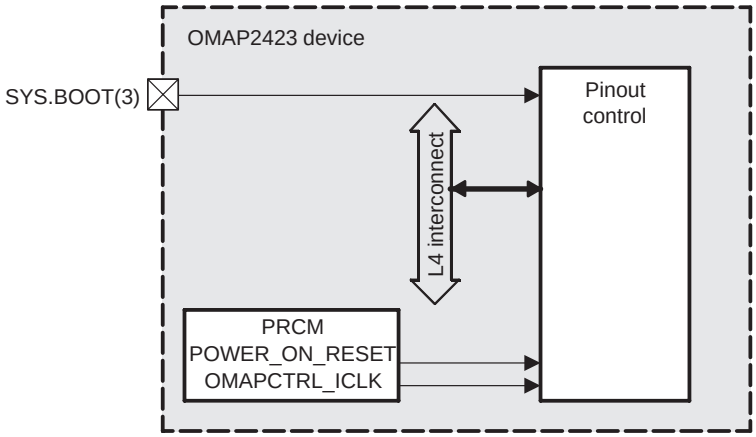
30.4 Pinout Control Integration

30.4.1 Pinout Description

The pinout control module contains a bank of registers used to control I/O pins. Considering no other functionality than register access for this module, only the interface clock is required.

Because pinout control is included in the wake-up domain, the reset is the internal power-on reset. Consequently, the pinout is not altered by a warm-up reset.

Figure 30−5. Pinout Integration



SYS.BOOT(3) is sampled at the release of the reset and changes the reset value of the MUXMODE field for certain pins of the GPMC interface (the default mode, depending on the SYS.BOOT(3) value for these pins). (See Section 30.4.1.2 for the list and behavior of these particular pins.)

30.4.1.1 Clocking, Reset, and Power Management Scheme

See Section 30.4.1.1 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

30.4.1.2 Pin Characteristics

Table 1−1 describes the characteristics of each register pad configuration. Pads are classified alphabetically in ascending order.

- Ball: Ball index of the pin on the package
- I/O type: *I* for pure input, *O* for pure output, and *IO* for I/O pins
- Mux type: *Muxed* when the pin can be multiplexed on several different functions; *dedicated* when the pin is hardwired to one function
- Power: Define the power voltage source for the pin
- Name extension: Suffix of the pinout control register, which contains the pinout-control register field for the pin configuration. To obtain the full register name, concatenate CONTROL_PADCONF_ and Suffix: CONTROL_PADCONF_<Name extension>.

- Offset: Offset address for the 32-bit pinout control register. To obtain the physical address, add 0x4800 0000 to this offset value.

- Byte: The position of the byte pinout-control register field inside the 32-bit pinout control register:

  - 0: Bits 0 to 7
  - 1: Bits 8 to 15
  - 2: Bits 16 to 23
  - 3: Bits 24 to 31

  The addition of the byte and offset give the address of the pinout control register field byte. Gray shading indicates the byte is read-only (for dedicated pins).

- Pin name: The name of both the pin and the pinout control register field. The name refers to the primary function of the pin. Gray shading indicates that the pinout-control register field is unused (for dedicated pins). See Table 1−1.

- Control reset state: The state of the pin at reset:

  - Mode0: The default mode is the primary mode; the function selected in Mode0 is available at the release of the power-on reset.

  - Protect: On release of the power-on reset, the pin is in protected mode to avoid potential electrical contention (high impedance). By default, no functionality is connected to this pin. The pinout-control register field must be programmed by software to enable functionality on the pin.

  - SBoot3: The pin state depends on the SYS_BOOT(3) input pin value sampled on release of the power-on reset. A high SYS_BOOT(3) value indicates an internal boot and protected mode reset state. A low value indicates an external boot and Mode0 reset state.

  Gray shading indicates a dedicated pin; this state is not significant (see Table 31−6).

- Control reset value: The default value for the MUXMODE field in the pinout-control register field (the MUXMODE value defines the mode on the pin).

  Gray shading indicates a dedicated pin; this value is not significant (the value is hardwired and cannot be modified by software).

  - 000: Mode0 is the default mode.

  - 111: Protected mode is the default mode.

  - *** : are reset values determined by the SYS_BOOT(3) input.

    A high SYS_BOOT(3) indicates an internal boot; the bits reset to 111. A low SYS_BOOT(3) value indicates an external boot; the bits reset to 000.

- Pull type: The type of pull cell available internally on the pin:
  - PUPD: Combination of a pullup and pulldown (software programmable)
  - PU: Pullup (enable or disable by software programming)
  - PD: Pulldown (enable or disable by software programming)
  - No: No internal pull available on the pin
- Pull reset state: The state of the pull on the pin:
  - U: Pullup active by default (at release of the power on reset)
  - D: Pulldown active by default (at release of the power on reset)
  - −: No pull active by default

  Gray shading indicates that no internal pull is available for this pin; this state is not significant.
- Pull reset value: The default value for the pull field of the pinout-control register field.

Gray shading indicates that no internal pull is available on the pin; this value is hardwired and cannot be modified by software.

Note:

Gray shading in Table 1−1 indicates a static configuration that cannot be modified by software. Software access to gray-shaded values returns no error but has no effect on the pin.

30.4.1.3 Power Voltage Pins

Pins not used for interfacing with external devices are used for the OMAP2423 supply. Multiple pins are connected to the same voltage. Table 31−4 lists the definition of each voltage.

Table 31−4. Power Pins Definition

Power Pin	Description
VDD	OMAP processor cores and logic operating at full speed
	OMAP processor cores and logic operating in *low-voltage functional mode*
	Low voltage retention (low IDDQ mode)
V_{DDS}, V_{DDS2}	Supply voltage for I/O macros (except GPMC, SDRC)
V_{DDS1}	Supply voltage for I/O to memory interfaces (GPMC, SDRC)
DDR_VDD	Core supply voltage for stacked mobile DDR SRAM memories
DDR_VSS	Core ground for stacked mobile DDR SRAM memories
DDR_VDDQ†	Supply voltage for stacked mobile DDR SRAM output buffers
DDR_VSSQ	Ground for stacked mobile DDR SRAM output buffers
V_{DDA}	Analog supply voltage for video DAC
VDD_PLL	Supply voltage for APLL and DPLLs
VDD_DLL‡	Supply voltage for SDRC DLL core operating at full speed
	Supply voltage for SDRC DLL core operating in *low-voltage functional mode*
V_{SS}	Common ground
V_{SSA}	Analog ground for video DAC

† DDR_VDD must be equal to DDR_VDDQ.
‡ VDD_DLL must operate at the same voltage setting as VDD.

30.4.1.4 OMAP2420 Package

Table 31−5 shows the pin information for the OMAP2420 package.

Table 31−5. OMAP2420 Package

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
AA	vpp	vpp	vss	vlynq_rx0	vdds	vlynq_rx1	vss	vlynq_nla	vdds	vlynq_clk	vddpll	vlynq_tx0	vss	tv_rref	vdds	tv_cvbs	sys_clkreq	tv_vref	vdds	jtag_ntrst	jtag_emu1
Y	vdd	cam_d7	eac_bt_din	cam_d8	gpio_120	dss_hsync	dss_d0	dss_d3	dss_d6	dss_d11	vlynq_tx1	ssi1_dat_rx	ssi1_wake	sys_nrespwron	eac_ac_sclk	sys_nreswarm	sys_32k	sys_xtalin	vdda-dac	sys_nvmode	jtag_emu0
W	vdds	cam_d4	cam_d6	eac_bt_dout	gpio_119	dss_pclk	dss_acbias	dss_d4	dss_d8	dss_d12	dss_d15	ssi1_dat_tx	ssi1_flag_rx	sys_clkout	eac_ac_din	eac_ac_rst	vss	vssa-dac	sys_nirq	uart2_rts	vdd
V	vdd	cam_d1	cam_d2	cam_d5	cam_lclk	cam_d9	dss_vsync	dss_d2	dss_d7	dss_d10	dss_d14	ssi1_flag_tx	ssi1_rdy_rx	eac_ac_mclk	eac_ac_dout	sys_xtalout	gpio_36	gpio_124	uart2_cts	spi1_simo	vdds2
U	vss	cam_vs	cam_xclk	cam_d3														spi1_clk	spi1_cs0	vss	spi1_cs3
T	vdd	vdds	cam_hs	cam_d0														spi1_somi	spi2_clk	vdd	uart1_rx
R	gpmc_nbe1	vss	gpmc_d5	gpmc_d7				eac_bt_sclk	gpio_121	dss_d5	dss_d13	dss_d17	gpio_62	eac_ac_fs				spi1_cs2	spi2_simo	spi2_somi	vss
P	gpmc_wait3	vdd	gpmc_d4	gpmc_d6			gpmc_nbe0	gpio_122	eac_bt_fs	dss_d1	dss_d9	dss_d16	ssi1_rdy_tx	gpio_125	uart2_rx			mcbsp1_dr	mcbsp1_dx	mcbsp1_fsr	mcbsp2_clkx
N	vss	gpmc_ncs3	gpmc_d2	gpmc_d3			gpmc_wait1	gpmc_ncs1						uart2_tx	spi1_cs1			hdq_sio	mcbsp1_clkx	vdd	vdds2
M	gpmc_wait2	vdds1	gpmc_d0	gpmc_d1			gpmc_d9	gpmc_d8						spi2_cs0	mcbsp1_clkr			mcbsp_clks	i2c1_scl	vss	mcbsp2_dr
L	vdd	gpmc_ncs7	gpmc_wait0	gpmc_ncs0			gpmc_d10	gpmc_d12						mcbsp1_fsx	i2c1_sda			uart3_cts_rctx	uart3_rts_sd	uart1_tx	vdd
K	gpmc_ncs6	vss	gpmc_nadv_ale	gpmc_nwe			gpmc_d13	gpmc_d11						uart3_rx_irrx	uart3_tx_irtx			usb0_dat	usb0_txen	usb0_vm	vss

Table 31−5. OMAP2420 Package (Continued)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
J	vdds1	vdd	gpmc_a1	gpmc_clk			gpmc_d15	gpmc_d14						usb0_se0	i2c2_scl			usb0_rcv	usb0_vp	usb0_puen	vss
H	gpmc_ncs5	gpmc_noe	gpmc_a3	gpmc_a2			sdrc_d1	sdrc_d7	sdrc_d14	sdrc_d11	sdrc_d9	sdrc_d17	sdrc_d28	mmc_dat1	mmc_clki			mmc_cmd	i2c2_sda	vdd	uart1_rts
G	vss	gpmc_nwp	gpmc_a4	gpmc_a8				sdrc_a6	sdrc_dqs1	sdrc_d10	sdrc_d8	sdrc_dm1	sdrc_d20	sdrc_d24				mmc_cmd_dir	mmc_clko	vss	vdds2
F	gpmc_ncs4	vdd	gpmc_a6	gpmc_a5														mmc_dat_dir2	mmc_dat_dir0	mmc_dat0	vdd
E	vdds1	gpmc_ncs2	gpmc_a10	gpmc_a7	gpio_6													mmc_dat_dir3	mmc_dat2	mmc_dat_dir1	vss
D	vdds1	vss	gpmc_a9	sdrc_d2	sdrc_dqs0	sdrc_d13	sdrc_a3	sdrc_a1	sdrc_a8	sdrc_ba1	sdrc_a12	sdrc_ncs0	sdrc_cke0	sdrc_ncas	sdrc_d18	sdrc_dqs2	sdrc_dm3	sdrc_d25	mmc_dat3	vdd	uart1_cts
C	vdd	sdrc_d0	sdrc_d5	sdrc_d4	sdrc_d6	sdrc_d15	sdrc_a5	sdrc_a7	sdrc_a10	sdrc_a11	sdrc_ba0	sdrc_ncs1	sdrc_nwe	sdrc_clk	sdrc_d19	sdrc_d23	sdrc_dm2	sdrc_d30	sdrc_d29	sdrc_d31	vss
B	jtag_tck	sdrc_d3	sdrc_a14	sdrc_a13	sdrc_dm0	sdrc_a4	sdrc_a2	vdddll	sdrc_a0	sdrc_a9	vss	sdrc_nras	sdrc_cke1	sdrc_nclk	sdrc_d16	sdrc_d21	sdrc_d22	sdrc_dqs3	sdrc_d26	sdrc_d27	jtag_tms
A	jtag_rtck	vss	vdds1	vss	vss	vdds1	vdds1	vss	vss	vdds1	sdrc_d12	vss	vdds1	vdds1	vss	vss	vdds1	vss	vdds1	jtag_tdi	jtag_tdo
0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**

30.4.1.5 OMAP2423 Package

Table 31−5a shows the pin information for the OMAP2423 package.

Table 31−5a. OMAP2423 Package (Continued)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
AF	vss	ddr_vddq	gpmc_d4	gpio_120	dss_pclk	gpio_119	ddr_vddq	dss_vsync	gpmc_a6	gpmc_a3	ssi1_flag_rx	gpmc_nbe0	ddr_vdd	gpmc_ncs1	ssi1_rdy_tx	ddr_vddq	ddr_vssq	vddpll	gpmc_ncs0	eac_ac_dout	ddr_vdd	eac_ac_dout	DNC †	DNC †	ddr_vdd	vss
AE	ddr_vssq	sys_nreswarm	gpmc_d6	vss	gpmc_d7	gpio_121	ddr_vss	dss_ac-bias	dss_d3	vlynq_nla	dss_d11	gpmc_wait0	dss_d12	ddr_vss	gpmc_nwp	gpmc_a5	ddr_vdd	gpmc_a9	sys_clkout	eac_ac_rst	ddr_vssq	eac_ac_rst	DNC †	gpio_125	DNC †	ddr_vss
AD	ddr_vss	vdds1	cam_d5	eac_bt_sclk	eac_bt_dout	cam_lclk	vlynq_rx0	dss_hsync	dss_d4	dss_d5	dss_d8	dss_d9	dss_d13	dss_d16	ssi1_wake	eac_ac_fs	vss	gpio_62	eac_ac_din	vdda-dac	sys_32k	vdda-dac	tv_cvbs	uart2_rx	ddr_vssq	ddr_vddq
AC	vdd	gpmc_d2	vpp	vpp	vdd	cam_d8	cam_d9	dss_d2	dss_d1	vss	dss_d7	dss_d10	dss_d14	vss	ssi1_flag_tx	ssi1_rdy_rx	ddr_vddq	eac_ac_sclk	sys_xtal-out	jtag_emu1	vss	jtag_emu1	tv_rref	uart2_cts	DNC †	DNC †
AB	gpmc_d3	gpmc_d0	cam_d3	cam_d7																			vssa-dac	uart2_tx	jtag_ntrst	tv_vref
AA	vdds-2	gpmc_d1	vss	cam_d6																			vdd	spi2_ncs0	DNC †	gpio_0
Y	gpmc_d9	ddr_vdd	cam_d1	cam_d0			vdd	ddr_vddq	dss_d0	vss	vdds-4	dss_d15	dss_d17	vss	vlynq_tx0	vdds-5	eac_ac_mclk	gpio_36	gpio_124	sys_nirq			spi1_clk	mcbsp1_clkr	DNC †	DNC †
W	vdd	ddr_vss	gpio_122	cam_xclk			eac_bt_fs	gpmc_a4	gpmc_a2	vlynq_rx1	dss_d6	vlynq_clk	ssi1_dat_tx	vlynq_tx1	ssi1_dat_rx	sys_nvmode	sys_nresp-wron	vss	vss	uart2_rts			spi1_ncs1	spi1_ncs0	DNC †	DNC †
V	gpmc_d8	cam_d4	Temp-Warn	cam_hs			vdd	gpmc_dt											spi1_somi	vdd			spi1_ncs2	spi2_clk	spi2_somi	vdd
U	gpmc_d12	gpmc_d11	gpmc_nbe1	vdd			cam_d2	eac_bt_din											vdd-6	spi1_ncs3			mcbsp1_dr	spi2_simo	ddr_vss	ddr_vdd
0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**

† DNC: Do not connect. Can be used as debug pins for SDRC interface.

Table 31−5a. OMAP2423 Package (Continued)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
T	gpmc_d10	ddr_vdd (ddr_svt)	vdd	gpmc_wait3			vss	gpmc_ncs3											mcbsp1_clkx	uart1_rx			hdq_sio	mcbsp2_clkx	DNC†	DNC†
R	vdd	ddr_vss	vdd	vdd			vdds-1	vss											i2c1_sda	mcbsp1_fsr			mcbsp1_fsx	mcbsp1_dx	DNC†	DNC†
P	gpmc_clk	gpmc_d13	gpmc_ncs7	vss			cam_vs	vss											vdds2	mcbsp2_dr			mcbsp_clks	i2c1_scl	DNC†	vss
N	gpmc_noe	gpmc_wait2	vdds-3	DNC†			DNC†	gpmc_nadv_ale											vss	vss			usb0_vm	jtag_emu0	DNC†	vdd
M	gpmc_nwe	ddr_vss (ddr_hi−z)	vdd	vdd			gpmc_ncs6	gpmc_a1											vdds2	uart3_rts_sd			usb0_vp	usb0_txen	DNC†	DNC†
L	ddr_vdd	DNC†	vdd	vdd			vss	vss											i2c2_scl	vdd			usb0_puen	usb0_se0	DNC†	DNC†
K	ddr_vss	vdds1	gpmc_ncs5	vdds1			vss	vss											i2c2_sda	uart1_rts			usb0_rcv	uart3_cts_rctx	ddr_vss	ddr_vdd
J	vdd	vdd	vdd	vdd			gpmc_a8	vss											vss	vdd			mmc_cmd	mmc_cmd_dir	usb0_dat	vdd
H	gpmc_d15	gpmc_d14	vdd	vdd			vdds1	gpmc_ncs4	vss	spi1_simo	DNC†	uart1_tx	vss	vdd	vdd	vss	mmc_dat0	vss	vdd	mmc_dat2			mmc_clko	uart3_rx_irrx	DNC†	DNC†
G	ddr_vssq	ddr_vddq	vdd	gpmc_ncs2				vdds1	vss	DNC†	DNC†	DNC†	vss	vdd	vdd	vss	vss	vss	mcc_dat1	uart1_cts			mmc_dat_dir2	uart3_tx_irtx	ddr_vssq	ddr_vddq
0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**

† DNC: Do not connect. Can be used as debug pins for SDRC interface.

Table 31−5a. OMAP2423 Package (Continued)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
F	sdrc_d0	sdrc_d1	vdd	vdd																			mmc_dat_dir0	vss	DNC†	DNC†
E	vss	vss	jtag_tck	gpmc_a7																			mmc_dat_dir3	vss	vss	vdd
D	sdrc_d3	sdrc_d5	vss	vss	gpmc_a10	gpio_6	vss	vdds1	vss	vdds1	DNC†	DNC†	vss	vdds1	vdd	vdd	vdds1	vdds2	vss	vss	DNC†	jtag_tms	mmc_dat_dir1	mmc_dat3	DNC†	DNC†
C	DNC†	DNC†	vss	vss	jtag_rtck	vss	vss	DNC†	vss	vdds1	vdds1	vdddll	vss	vdds1	vdd	vdd	DNC†	DNC†	vss	vss	DNC†	jtag_tdi	mmc_clki	jtag_tdo	DNC†	DNC†
B	ddr_vss	sdrc_d2	sdrc_d6	sdrc_d7	vss	ddr_vddq	sdrc_d9	sdrc_d14	vss	DNC†	sdrc_d11	sdrc_d8	vss	DNC†	DNC†	sdrc_nclk(CLKB)	ddr_vssq	vss	DNC†	ddr_vdd	DNC†	vss	DNC†	DNC†	DNC†	ddr_vssq
A	vss	ddr_vdd	sdrc_d4	ddr_vssq	vss	sdrc_d13	sdrc_d15	DNC†	vss	sdrc_d10	sdrc_d12	DNC†	vss	DNC†	sdrc_clk(CLK)	ddr_vss	DNC†	vss	ddr_vddq	DNC	DNC†	vss	DNC†	DNC†	ddr_vddq	vss
0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**

† DNC: Do not connect. Can be used as debug pins for SDRC interface.

Table 31−5b. Ball Out OMAP2422/2423 Cross-Reference

Signal Name OMAP2422	Ball 2422	Signal Name OMAP2423	Ball 2423
gpio_6	E5	gpio_6	D6
gpmc_a12	H7	gpmc_a12	F2
gpmc_d14	J8	gpmc_d14	H2
jtag_tck	B1	jtag_tck	E3
gpmc_a11	C2	gpmc_a11	F1
gpmc_a9	D3	gpmc_a9	AE18
gpmc_a7	E4	gpmc_a7	E4
gpmc_a5	F4	gpmc_a5	AE16
gpmc_a10	E3	gpmc_a10	D5
gpmc_ncs2	E2	gpmc_ncs2	G4
gpmc_d15	J7	gpmc_d15	H1
gpmc_a6	F3	gpmc_a6	AF9
gpmc_a8	G4	gpmc_a8	J7
gpmc_ncs4	F1	gpmc_ncs4	H8
gpmc_a4	G3	gpmc_a4	W8
gpmc_nwp	G2	gpmc_nwp	AE15
gpmc_d11	K8	gpmc_d11	U2
gpmc_a2	H4	gpmc_a2	W9
gpmc_noe	H2	gpmc_noe	N1
gpmc_a3	H3	gpmc_a3	AF10
gpmc_ncs5	H1	gpmc_ncs5	K3
gpmc_d13	K7	gpmc_d13	P2
gpmc_clk	J4	gpmc_clk	P1
gpmc_a1	J3	gpmc_a1	M8
gpmc_d12	L8	gpmc_d12	U1
gpmc_nwe	K4	gpmc_nwe	M1
gpmc_nadv_ale	K3	gpmc_nadv_ale	N8
gpmc_ncs6	K1	gpmc_ncs6	M7
gpmc_d10	L7	gpmc_d10	T1
gpmc_ncs0	L4	gpmc_ncs0	AF19
gpmc_wait0	L3	gpmc_wait0	AE12
gpmc_d8	M8	gpmc_d8	V1
gpmc_d9	M7	gpmc_d9	Y1
gpmc_ncs7	L2	gpmc_ncs7	P3

Table 31−5b. Ball Out OMAP2422/2423 Cross Reference (Continued)

Signal Name OMAP2422	Ball 2422	Signal Name OMAP2423	Ball 2423
gpmc_d0	M3	gpmc_d0	AB2
gpmc_wait2	M1	gpmc_wait2	N2
gpmc_d1	M4	gpmc_d1	AA2
gpmc_ncs1	N8	gpmc_ncs1	AF14
gpmc_d2	N3	gpmc_d2	AC2
gpmc_d3	N4	gpmc_d3	AB1
gpmc_ncs3	N2	gpmc_ncs3	T8
gpmc_wait1	N7	gpmc_wait1	AE20
gpmc_d4	P3	gpmc_d4	AF3
gpmc_d6	P4	gpmc_d6	AE3
gpmc_wait3	P1	gpmc_wait3	T4
gpmc_nbe0	P7	gpmc_nbe0	AF12
gpmc_nbe1	R1	gpmc_nbe1	U3
gpmc_d5	R3	gpmc_d5	V8
gpmc_d7	R4	gpmc_d7	AE5
cam_hs	T3	cam_hs	V4
gpio_122	P8	gpio_122	W3
cam_vs	U2	cam_vs	P7
cam_xclk	U3	cam_xclk	W4
cam_d0	T4	cam_d0	Y4
cam_d1	V2	cam_d1	Y3
cam_d2	V3	cam_d2	U7
cam_d3	U4	cam_d3	AB3
cam_d4	W2	cam_d4	V2
cam_d5	V4	cam_d5	AD3
cam_d6	W3	cam_d6	AA4
cam_d7	Y2	cam_d7	AB4
eac_bt_sclk	R8	eac_bt_sclk	AD4
eac_bt_fs	P9	eac_bt_fs	W7
eac_bt_din	Y3	eac_bt_din	U8
eac_bt_dout	W4	eac_bt_dout	AD5
cam_lclk	V5	cam_lclk	AD6

Table 31−5b. Ball Out OMAP2422/2423 Cross Reference (Continued)

Signal Name OMAP2422	Ball 2422	Signal Name OMAP2423	Ball 2423
cam_d8	Y4	cam_d8	AC6
cam_d9	V6	cam_d9	AC7
gpio_119	W5	gpio_119	AF6
gpio_120	Y5	gpio_120	AF4
vlynq_rx0	AA4	vlynq_rx0	AD7
gpio_121	R9	gpio_121	AE6
dss_pclk	W6	dss_pclk	AF5
dss_hsync	Y6	dss_hsync	AD8
dss_vsync	V7	dss_vsync	AF8
dss_acbias	W7	dss_acbias	AE8
dss_d0	Y7	dss_d0	Y9
dss_d1	P10	dss_d1	AC9
vlynq_rx1	AA6	vlynq_rx1	W10
dss_d2	V8	dss_d2	AC8
dss_d3	Y8	dss_d3	AE9
dss_d4	W8	dss_d4	AD9
vlynq_nla	AA8	vlynq_nla	AE10
dss_d5	R10	dss_d5	AD10
dss_d6	Y9	dss_d6	W11
dss_d7	V9	dss_d7	AC11
dss_d8	W9	dss_d8	AD11
dss_d11	Y10	dss_d11	AE11
dss_d9	P11	dss_d9	AD12
dss_d10	V10	dss_d10	AC12
dss_d12	W10	dss_d12	AE13
vlynq_clk	AA10	vlynq_clk	W12
dss_d13	R11	dss_d13	AD13
dss_d14	V11	dss_d14	AC13
dss_d15	W11	dss_d15	Y12
dss_d16	P12	dss_d16	AD14
dss_d17	R12	dss_d17	Y13
vlynq_tx1	Y11	vlynq_tx1	W14
ssi1_dat_tx	W12	ssi1_dat_tx	W13
vlynq_tx0	AA12	vlynq_tx0	Y15

Table 31−5b. Ball Out OMAP2422/2423 Cross Reference (Continued)

Signal Name OMAP2422	Ball 2422	Signal Name OMAP2423	Ball 2423
ssi1_flag_tx	V12	ssi1_flag_tx	AC15
ssi1_rdy_tx	P13	ssi1_rdy_tx	AF15
ssi1_dat_rx	Y12	ssi1_dat_rx	W15
ssi1_flag_rx	W13	ssi1_flag_rx	AF11
ssi1_rdy_rx	V13	ssi1_rdy_rx	AC16
ssi1_wake	Y13	ssi1_wake	AD15
gpio_62	R13	gpio_62	AD18
sys_nrespwron	Y14	sys_nrespwron	W17
sys_clkout	W14	sys_clkout	AE19
eac_ac_mclk	V14	eac_ac_mclk	Y17
eac_ac_fs	R14	eac_ac_fs	AD16
eac_ac_sclk	Y15	eac_ac_sclk	AC18
eac_ac_din	W15	eac_ac_din	AD19
sys_nreswarm	Y16	sys_nreswarm	AE2
eac_ac_dout	V15	eac_ac_dout	AF22
eac_ac_rst	W16	eac_ac_rst	AE22
sys_clkreq	AA17	sys_clkreq	AD20
gpio_125	P14	gpio_125	AE24
sys_32k	Y17	sys_32k	AD21
sys_xtalout	V16	sys_xtalout	AC19
sys_xtalin	Y18	sys_xtalin	AC20
gpio_36	V17	gpio_36	Y18
tv_rref	AA14	tv_rref	AC23
tv_vref	AA18	tv_vref	AB26
tv_cvbs	AA16	tv_cvbs	AD23
vddadac	Y19	vddadac	AD22
gpio_124	V18	gpio_124	Y19
jtag_ntrst	AA20	jtag_ntrst	AB25
sys_nirq	W19	sys_nirq	Y20
sys_nvmode	Y20	sys_nvmode	W16
jtag_emu1	AA21	jtag_emu1	AC22
uart2_rx	P15	uart2_rx	AD24
uart2_tx	N14	uart2_tx	AB24
jtag_emu0	Y21	jtag_emu0	N24

Table 31−5b. Ball Out OMAP2422/2423 Cross Reference (Continued)

Signal Name OMAP2422	Ball 2422	Signal Name OMAP2423	Ball 2423
uart2_rts	W20	uart2_rts	W20
uart2_cts	V19	uart2_cts	AC24
spi1_clk	U18	spi1_clk	Y23
spi1_simo	V20	spi1_simo	H10
spi1_somi	T18	spi1_somi	V19
spi1_ncs0	U19	spi1_ncs0	W24
spi1_ncs1	N15	spi1_ncs1	W23
spi1_ncs3	U21	spi1_ncs3	U20
spi2_clk	T19	spi2_clk	V24
spi1_ncs2	R18	spi1_ncs2	V23
uart1_rx	T21	uart1_rx	T20
spi2_simo	R19	spi2_simo	U24
spi2_somi	R20	spi2_somi	V25
spi2_ncs0	M14	spi2_ncs0	AA24
mcbsp1_dr	P18	mcbsp1_dr	U23
mcbsp1_fsr	P20	mcbsp1_fsr	R20
mcbsp1_dx	P19	mcbsp1_dx	R24
mcbsp2_clkx	P21	mcbsp2_clkx	T24
mcbsp1_clkr	M15	mcbsp1_clkr	Y24
hdq_sio	N18	hdq_sio	T23
mcbsp1_clkx	N19	mcbsp1_clkx	T19
mcbsp1_fsx	L14	mcbsp1_fsx	R23
mcbsp_clks	M18	mcbsp_clks	P23
i2c1_scl	M19	i2c1_scl	P24
mcbsp2_dr	M21	mcbsp2_dr	P20
i2c1_sda	L15	i2c1_sda	R19
uart1_tx	L20	uart1_tx	H12
uart3_cts_rctx	L18	uart3_cts_rctx	K24
uart3_rts_sd	L19	uart3_rts_sd	M20
uart3_rx_irrx	K14	uart3_rx_irrx	H24
uart3_tx_irtx	K15	uart3_tx_irtx	G24
usb0_txen	K19	usb0_txen	M24
usb0_dat	K18	usb0_dat	J25

Table 31−5b. Ball Out OMAP2422/2423 Cross Reference (Continued)

Signal Name OMAP2422	Ball 2422	Signal Name OMAP2423	Ball 2423
usb0_se0	J14	usb0_se0	L24
usb0_vm	K20	usb0_vm	N23
usb0_vp	J19	usb0_vp	M23
usb0_rcv	J18	usb0_rcv	K23
usb0_puen	J20	usb0_puen	L23
i2c2_scl	J15	i2c2_scl	L19
uart1_rts	H21	uart1_rts	K20
i2c2_sda	H19	i2c2_sda	K19
mmc_cmd	H18	mmc_cmd	J23
mmc_clki	H15	mmc_clki	C23
mmc_clko	G19	mmc_clko	H23
mmc_cmd_dir	G18	mmc_cmd_dir	J24
mmc_dat0	F20	mmc_dat0	H17
mmc_dat_dir0	F19	mmc_dat_dir0	F23
mmc_dat1	H14	mmc_dat1	G19
mmc_dat_dir1	E20	mmc_dat_dir1	D23
mmc_dat2	E19	mmc_dat2	H20
uart1_cts	D21	uart1_cts	G20
mmc_dat_dir2	F18	mmc_dat_dir2	G23
mmc_dat3	D19	mmc_dat3	D24
mmc_dat_dir3	E18	mmc_dat_dir3	E23
sdrc_d31(DQ0)		sdrc_d31(DQ0)	
sdrc_d25(DQ1)		sdrc_d25(DQ1)	D25
jtag_tms	B21	jtag_tms	D22
sdrc_d29(DQ2)	C19	sdrc_d29(DQ2)	
sdrc_d27(DQ3)		sdrc_d27(DQ3)	
jtag_tdo	A21	jtag_tdo	C24
sdrc_d24(DQ4)		sdrc_d24(DQ4)	
sdrc_d28(DQ5)		sdrc_d28(DQ5)	
jtag_tdi	A20	jtag_tdi	C22
sdrc_d26(DQ6)		sdrc_d26(DQ6)	
sdrc_d30(DQ7)		sdrc_d30(DQ7)	
sdrc_dm3(DM0)		sdrc_dm3(DM0)	
sdrc_dqs3(DQS0)	B18	sdrc_dqs3(DQS0)	A24

Table 31−5b. Ball Out OMAP2422/2423 Cross Reference (Continued)

Signal Name OMAP2422	Ball 2422	Signal Name OMAP2423	Ball 2423
sdrc_dqs2(DQS1)	D16	sdrc_dqs2(DQS1)	
sdrc_dm2(DM1)		sdrc_dm2(DM1)	
sdrc_d22(DQ8)		sdrc_d22(DQ8)	
sdrc_d20(DQ9)		sdrc_d20(DQ9)	
sdrc_d23(DQ10)	C16	sdrc_d23(DQ10)	B21
sdrc_d21(DQ11)		sdrc_d21(DQ11)	
sdrc_d18(DQ12)		sdrc_d18(DQ12)	
sdrc_d19(DQ13)		sdrc_d19(DQ13)	
sdrc_d16(DQ14)		sdrc_d16(DQ14)	
sdrc_d17(DQ15)		sdrc_d17(DQ15)	
sdrc_ncas(CASB)	D14	sdrc_ncas(CASB)	AA25
sdrc_nclk(CLKB)	B14	sdrc_nclk(CLKB)	B16
sdrc_clk(CLK)	C14	sdrc_clk(CLK)	A15
sdrc_dm1		sdrc_dm1	
sdrc_cke1(CKE2)	B13	sdrc_cke1(CKE2)	AE25
sdrc_cke0(CKE1)	D13	sdrc_cke0(CKE1)	AF24
sdrc_nwe(WEB)	C13	sdrc_nwe(WEB)	AE23
sdrc_nras(RASB)	B12	sdrc_nras(RASB)	AC26
gpmc_a20	H11	gpmc_a20	B7
sdrc_ncs0(CSB1)	D12	sdrc_ncs0(CSB1)	Y26
sdrc_ncs1(CSB2)	C12	sdrc_ncs1(CSB2)	Y25
gpmc_a19	G11	gpmc_a19	B12
sdrc_a12(A12)	D11	sdrc_a12(A12)	W26
sdrc_ba0(BA0)	C11	sdrc_ba0(BA0)	AF23
gpmc_a23	A11	gpmc_a23	A11
gpmc_a22	H10	gpmc_a22	B11
gpmc_a21	G10	gpmc_a21	A10
sdrc_a11(A11)	C10	sdrc_a11(A11)	T26
sdrc_ba1(BA1)	D10	sdrc_ba1(BA1)	AC25
gpmc_a25	H9	gpmc_a25	B8
sdrc_a9(A9)	B10	sdrc_a9(A9)	M26
sdrc_a10(A10)	C9	sdrc_a10(A10)	H26
sdrc_a8(A8)	D9	sdrc_a8(A8)	T25

Table 31−5b. Ball Out OMAP2422/2423 Cross Reference (Continued)

Signal Name OMAP2422	Ball 2422	Signal Name OMAP2423	Ball 2423
sdrc_a0(A0)	B9	sdrc_a0(A0)	H25
sdrc_dqs1		sdrc_dqs1	
sdrc_a7(A7)	C8	sdrc_a7(A7)	R25
vdddll	B8	vdddll	C12
sdrc_a1(A1)	D8	sdrc_a1(A1)	M25
sdrc_a6(A6)	G8	sdrc_a6(A6)	R26
sdrc_a2(A2)	B7	sdrc_a2(A2)	L25
sdrc_a5(A5)	C7	sdrc_a5(A5)	N25
sdrc_a3(A3)	D7	sdrc_a3(A3)	L26
sdrc_a4(A4)	B6	sdrc_a4(A4)	P25
gpmc_a26	C6	gpmc_a26	A7
sdrc_stk_d31(DQ16)		sdrc_stk_d31(DQ16)	
sdrc_stk_d30(DQ17)		sdrc_stk_d30(DQ17)	
gpmc_a18	H8	gpmc_a18	B4
sdrc_stk_d29(DQ18)	D5	sdrc_stk_d29(DQ18)	H11
sdrc_dm0		sdrc_dm0	
sdrc_stk_d28(DQ19)		sdrc_stk_d28(DQ19)	
sdrc_stk_d27(DQ20)		sdrc_stk_d27(DQ20)	
sdrc_stk_d26(DQ21)		sdrc_stk_d26(DQ21)	
gpmc_a17	C5	gpmc_a17	B3
sdrc_stk_d25(DQ22)		sdrc_stk_d25(DQ22)	
gpmc_a24	D6	gpmc_a24	A6
sdrc_stk_d24(DQ23)		sdrc_stk_d24(DQ23)	
sdrc_stk_dm1(DM2)		sdrc_stk_dm1(DM2)	
sdrc_a13(A13)	B4	sdrc_a13(A13)	W25
sdrc_stk_dqs1(DQS2)	H13	sdrc_stk_dqs1(DQS2)	G10
gpmc_a15	C4	gpmc_a15	A3
sdrc_stk_dqs0(DQS3)	H12	sdrc_stk_dqs0(DQS3)	G12
sdrc_dqs0		sdrc_dqs0	
gpio_0	B3	gpio_0	AA26
gpmc_a13	D4	gpmc_a13	B2
sdrc_stk_dm0(DM3)		sdrc_stk_dm0(DM3)	
gpmc_a16	C3	gpmc_a16	D2
sdrc_stk_d23(DQ24)		sdrc_stk_d23(DQ24)	

Table 31−5b. Ball Out OMAP2422/2423 Cross Reference (Continued)

Signal Name OMAP2422	Ball 2422	Signal Name OMAP2423	Ball 2423
sdrc_stk_d22(DQ25)		sdrc_stk_d22(DQ25)	
sdrc_stk_d21(DQ26)	G9	sdrc_stk_d21(DQ26)	G11
sdrc_stk_d20(DQ27)		sdrc_stk_d20(DQ27)	
gpmc_a14	B2	gpmc_a14	D1
sdrc_stk_d19(DQ28)		sdrc_stk_d19(DQ28)	
sdrc_stk_d18(DQ29)		sdrc_stk_d18(DQ29)	
sdrc_stk_d17(DQ30)		sdrc_stk_d17(DQ30)	
sdrc_stk_d16(DQ31)		sdrc_stk_d16(DQ31)	
jtag_rtck	A1	jtag_rtck	C5

30.4.1.6 Multiplexing Summary

Table 31−6 represents the OMAP2423 functional multiplexing. You can use this table to help you select the interface. If conflicts occur, you can check the table line by line. To resolve conflicts, see Section 30.6 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Table 31−6. Multiplexing Summary

Ball 2423	Pin List Specification						
	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6
B2	gpmc_a13						
F2	gpmc_a12						
F1	gpmc_a11						
D5	gpmc_a10		sys_ndmareq5	gpio_3			
AE18	gpmc_a9		sys_ndmareq4	gpio_4			
J7	gpmc_a8		sys_ndmareq3	gpio_5			
E4	gpmc_a7		sys_ndmareq2	gpio_6			
AF9	gpmc_a6	dss_d23		gpio_7			
AE16	gpmc_a5	dss_d22		gpio_8			
W8	gpmc_a4	dss_d21		gpio_9			
AF10	gpmc_a3	dss_d20		gpio_10			
W9	gpmc_a2	dss_d19		gpio_11			
M8	gpmc_a1	dss_d18		gpio_12			
H1	gpmc_d15	ssi2_dat_tx		gpio_13			
H2	gpmc_d14	ssi2_flag_tx		gpio_14			

Table 31−6. Multiplexing Summary (Continued)

Ball 2423	Pin List Specification						
	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6
P2	gpmc_d13	ssi2_rdy_tx		gpio_15			
U1	gpmc_d12	ssi2_dat_rx		gpio_16			
U2	gpmc_d11	ssi2_flag_rx		gpio_17			
T1	gpmc_d10	ssi2_rdy_rx		gpio_18			
Y1	gpmc_d9	ssi2_wake		gpio_19			
V1	gpmc_d8			gpio_20			
AE5	gpmc_d7						
AE3	gpmc_d6						
V8	gpmc_d5						
AF3	gpmc_d4						
AB1	gpmc_d3						
AC2	gpmc_d2						
AA2	gpmc_d1						
AB2	gpmc_d0						
P1	gpmc_clk			gpio_21			
AF19	gpmc_ncs0						
AF14	gpmc_ncs1			gpio_22			
G4	gpmc_ncs2			gpio_23			
T8	gpmc_ncs3	gpmc_io_dir		gpio_24			
H8	gpmc_ncs4			gpio_25			
K3	gpmc_ncs5			gpio_26			
M7	gpmc_ncs6			gpio_27			
P3	gpmc_ncs7	gpmc_io_dir		gpio_28			
N8	gpmc_nadv_ale						
N1	gpmc_noe						
M1	gpmc_nwe						
AF12	gpmc_nbe0			gpio_29			
U3	gpmc_nbe1			gpio_30			
AE15	gpmc_nwp			gpio_31			
AE12	gpmc_wait0						

Table 31−6. Multiplexing Summary (Continued)

Ball 2423	Pin List Specification						
	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6
AE20	gpmc_wait1			gpio_33			
N2	gpmc_wait2			gpio_34			
T4	gpmc_wait3			gpio_35			
Y9	dss_d0						
AC9	dss_d1						
AC8	dss_d2						
AE9	dss_d3						
AD9	dss_d4						
AD10	dss_d5						
W11	dss_d6						
AC11	dss_d7						
AD11	dss_d8			gpio_38			
AD12	dss_d9			gpio_39			
AC12	dss_d10			gpio_40			
AE11	dss_d11			gpio_41			
AE13	dss_d12			gpio_42			
AD13	dss_d13			gpio_43			
AC13	dss_d14			gpio_44			
Y12	dss_d15			gpio_45			
AD14	dss_d16			gpio_46			
Y13	dss_d17			gpio_47			
G20	uart1_cts		dss_d18	gpio_32			
K20	uart1_rts		dss_d19	gpio_8			
H12	uart1_tx		dss_d20	gpio_9			
T20	uart1_rx		dss_d21	gpio_10			
P20	mcbsp2_dr		dss_d22	gpio_11			
T24	mcbsp_clkx		dss_d23	gpio_12			
AF5	dss_pclk						
AF8	dss_vsync						
AD8	dss_hsync						

Table 31−6. Multiplexing Summary (Continued)

Ball 2423	Pin List Specification						
	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6
AE8	dss_acbias		mcbsp2_fsx	gpio_48			
AC7	cam_d9	hw_dbg11		gpio_53			etk_d11
AC6	cam_d8	hw_dbg10		gpio_54			etk_d10
AB4	cam_d7	hw_dbg9					etk_d9
AA4	cam_d6	hw_dbg8					etk_d8
AD3	cam_d5	hw_dbg7	mcbsp1_clkr	gpio_49			etk_d7
V2	cam_d4	hw_dbg6	mcbsp1_fsr	gpio_50			etk_d6
AB3	cam_d3	hw_dbg5	mcbsp1_dr	gpio_51			etk_d5
U7	cam_d2	hw_dbg4†	mcbsp1_clkx	gpio_52			etk_d4
Y3	cam_d1	hw_dbg3†	sti_din	gpio_53			etk_d3
Y4	cam_d0	hw_dbg2†	sti_dout	gpio_54			etk_d2
V4	cam_hs	hw_dbg1†	mcbsp1_dx	gpio_55			etk_d1
P7	cam_vs	hw_dbg0†	mcbsp1_fsx	gpio_56			etk_d0
AD6	cam_lclk		mcbsp_clks	gpio_57			etk_c1
W4	cam_xclk	l4_ext_trig	sti_clk				etk_c2
W13	ssi1_dat_tx	uart1_tx	usb1_se0	gpio_59			
AC15	ssi1_flag_tx	uart1_rts	usb1_rcv	gpio_25			
AF15	ssi1_rdy_tx	uart1_cts	usb1_txen	gpio_61			
AD18	gpio_62	uart1_rx	usb1_dat	gpio_62			
W15	ssi1_dat_rx	eac_md_sclk		gpio_63			
AF11	ssi1_flag_rx	eac_md_din		gpio_64			
AC16	ssi1_rdy_rx	eac_md_dout		gpio_65			
AD15	ssi1_wake	eac_md_fs		gpio_66			
W12	vlynq_clk	usb2_se0	sys_ndmareq0	gpio_13			
W10	vlynq_rx1	usb2_rcv	sys_ndmareq1	gpio_14	cam_d8		
AD7	vlynq_rx0	usb2_tllse0		gpio_15	cam_d7		
W14	vlynq_tx1	usb2_dat	sys_clkout2	gpio_16			
Y15	vlynq_tx0	usb2_txen		gpio_17			
AE10	vlynq_nla		l4_ext_trig	gpio_58	cam_d6		
AC24	uart2_cts	usb1_rcv	gpt9_pwm_evt	gpio_67			

Table 31−6. Multiplexing Summary (Continued)

Ball 2423	Pin List Specification						
	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6
W20	uart2_rts	usb1_txen	gpt10_pwm_evt	gpio_68			
AB24	uart2_tx	usb1_se0	gpt11_pwm_evt	gpio_69			
AD24	uart2_rx	usb1_dat	gpt12_pwm_evt	gpio_70			
AD4	eac_bt_sclk			gpio_71			etk_d11
W7	eac_bt_fs			gpio_72			etk_d10
U8	eac_bt_din			gpio_73			etk_d9
AD5	eac_bt_dout		sti_clk	gpio_74			etk_d8
H23	mmc_clko	ms_clko					
J23	mmc_cmd	ms_bs					
H17	mmc_dat0	ms_dat0					
G19	mmc_dat1	ms_dat1		gpio_75			
H20	mmc_dat2	ms_dat2	uart2_cts	gpio_76			
D24	mmc_dat3	ms_dat3	l4_ext_trig	gpio_77			
F23	mmc_dat_dir0	ms_dat_dir0		gpio_7			
D23	mmc_dat_dir1	ms_dat_dir1	uart2_rts	gpio_78			
G23	mmc_dat_dir2	ms_dat_dir2	uart2_tx	gpio_79			
E23	mmc_dat_dir3	ms_dat_dir3	uart2_rx	gpio_80			
J24	mmc_cmd_dir			gpio_8			
C23	mmc_clki	ms_clki		gpio_59			
Y23	spi1_clk			gpio_81			
H10	spi1_simo			gpio_82			
V19	spi1_somi			gpio_83			
W24	spi1_cs0			gpio_84			
W23	spi1_cs1			gpio_85			
V23	spi1_cs2			gpio_86			
U20	spi1_cs3			gpio_87			
V24	spi2_clk			gpio_88			
U24	spi2_simo	gpt10_pwm_evt		gpio_89			
V25	spi2_somi	gpt11_pwm_evt		gpio_90			
AA24	spi2_cs0	gpt12_pwm_evt		gpio_91			

Table 31−6. Multiplexing Summary (Continued)

Ball 2423	Pin List Specification						
	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6
Y24	mcbsp1_clkr	ssi2_dat_tx	vlynq_tx1	gpio_92			
R20	mcbsp1_fsr	ssi2_flag_tx	vlynq_tx0	gpio_93	spi2_ncs1		
R24	mcbsp1_dx	ssi2_rdy_tx	vlynq_clk	gpio_94			
U23	mcbsp1_dr	ssi2_dat_rx	vlynq_rx1	gpio_95			
P23	mcbsp_clks	ssi2_flag_rx	vlynq_rx0	gpio_96			
R23	mcbsp1_fsx	ssi2_rdy_rx		gpio_97			
T19	mcbsp1_clkx	ssi2_wake	vlynq_la	gpio_98			
P24	i2c1_scl						
R19	i2c1_sda						
L19	i2c2_scl		gpt9_pwm_evt	gpio_99			
K19	i2c2_sda		spi2_cs1	gpio_100			
T23	hdq_sio	usb2_tllse0	sys_altclk	gpio_101			
K24	uart3_cts_rctx	uart3_rx_irrx		gpio_102			
M20	uart3_rts_sd	uart3_tx_irtx		gpio_103			
G24	uart3_tx_irtx	uart3_rctx		gpio_104			
H24	uart3_rx_irrx			gpio_105			
AD23	tv_cvbs						
AB26	tv_vref						
AC23	tv_rref						
L23	usb0_puen	mcbsp2_dx		gpio_106			
M23	usb0_vp	mcbsp2_dr		gpio_107			
N23	usb0_vm	mcbsp2_clkx		gpio_108	uart2_rx		
K23	usb0_rcv	mcbsp2_fsx		gpio_109	uart2_cts		
M24	usb0_txen	uart3_cts_rctx	uart2_cts	gpio_110			
P1	usb0_se0	uart3_tx_irtx	uart2_tx	gpio_111	uart2_rx		
J25	usb0_dat	uart3_rx_irrx	uart2_rx	gpio_112	uart2_tx		
AC18	eac_ac_sclk	mcbsp2_clkx		gpio_113			
AD16	eac_ac_fs	mcbsp2_fsx		gpio_114			
AD19	eac_ac_din	mcbsp2_dr		gpio_115			
AF22	eac_ac_dout	mcbsp2_dx		gpio_116			

Table 31−6. Multiplexing Summary (Continued)

Ball 2423	Pin List Specification						
	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6
Y17	eac_ac_mclk			gpio_117			
AE22	eac_ac_rst	eac_bt_din	loop_eac_bt_din	gpio_118			
W17	sys_nrespwron						
AE2	sys_nreswarm						
Y20	sys_nirq			gpio_60			
W16	sys_nvmode						
AF6	gpio_119		sti_din	gpio_119		sys_boot0	etk_d12
AF4	gpio_120		sti_dout	gpio_120		sys_boot1	etk_d13
AE6	gpio_121			gpio_121	jtag_emu2	sys_boot2	etk_d14
W3	gpio_122			gpio_122	jtag_emu3	sys_boot3	etk_d15
AD21	sys_32k						
AC20	sys_xtalin						
AC19	sys_xtalout						
Y18	gpio_36			gpio_36		sys_boot4	
AD20	sys_clkreq			gpio_52			
AE19	sys_clkout			gpio_123			
D6	gpio_6	tv_detect		gpio_6			
Y19	gpio_124			gpio_124		sys_boot5	
AE24	gpio_125	sys_jtagsel1	sys_jtagsel2	gpio_125			
AC22				gpio_126			
N24				gpio_127			

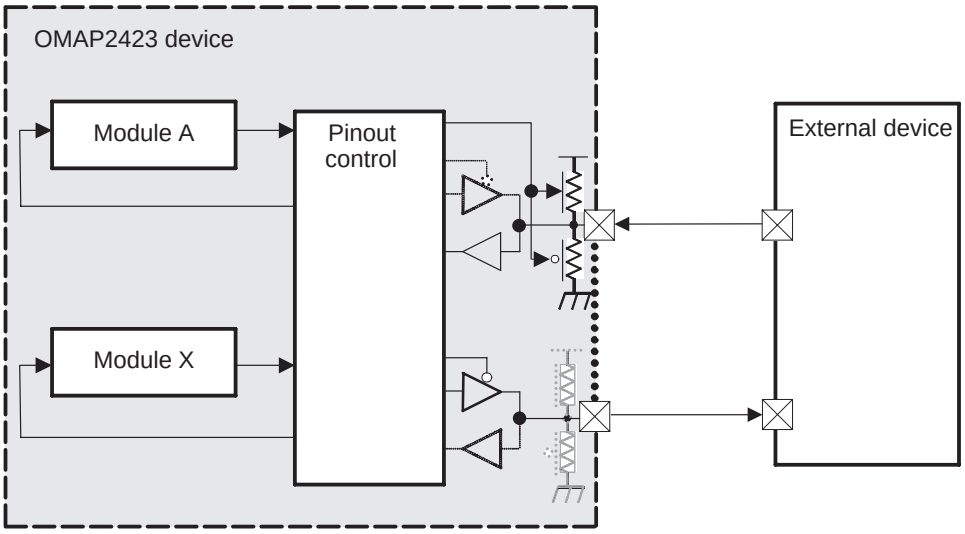
30.4.1.7 L4 Interconnect Interface

See Section 30.4.1.7 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

30.5 Pinout Environment

Pinout control enables the routing of internal module input/output to the chip boundary for interconnection with an external device.

Figure 30−6. Pinout Overview



For each external device interfacing with OMAP2423 processor, it is software's responsibility to ensure proper programming of the pinout control module. For maximum flexibility, each pin can be programmed independently.

30.5.1 Pinout Functional Interface

This chapter identifies all possible pins that can be configured for each module signal are classified by function.

All possible interfaces for each function are summarized in the appropriate table.

Note:

The purpose of this example is to show table organization; all signals and functions do not correspond to real OMAP2423 signals or functions.

Table 31−8. Pinout Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
FUNCn.SIG1	Signal 1 of function n Description	I	V4		funcx.sig2	funcy.sig1	funck.sig5		
FUNCn.SIG2	Signal 2 of function n Description	I	P7	funcx.sig2	funcy.sig4		funck.sig6		
FUNCn.SIG3	Signal 3 of function n Description	IO	W4	funcz.sig1		funcy.sig2			
FUNCn.SIG4	Signal 4 of function n Description	O	AA4		funck.sig2				
			Y3	funcy.sig3					

- This table is related to function n: All interface signals related to function n are listed in the Interface column.
- A maximum of four different signals, named SIG1, SIG2, SIG3, and SIG4, are available for function n.
- A short description of each signal can be found in the Description column. Some outputs are described as open drain. This means the output is not able to drive a high level on the pin but only low level or high impedance. These open drain outputs must use a pullup (internal or external) to ensure correct behavior. The open drain outputs are: HDQ.SIO, I2C1.SCL, I2C1.SDA, I2C2.SCL, I2C2.SDA, SYS.nRESWARM, and VLYNQ.nLA.
- The direction of the signal can be found in column DIR.
  - Notice that this direction is related to the signal, not to the pin: SIG1 is input only, but pin V4 can be an I/O pin. Information related to the pin itself can be found in Section 30.4.1.2.
  - Information related to the direction of the alternates function is not in this table but in the dedicated table for the alternate function. Funcx.sig2 direction would be available in the function x table.
- Ball 2423 provides the pin index that enables to identify the concerned ball on the chip package.
  - When the cell is shaded in black, the pin is not available.
  - Ball 2423 gives pin index to identify where the signal is multiplexed.
  - One signal can be multiplexed on several pins. In the example, SIG4 is available on both pins AA4 and Y3. It is the responsibility of the user to configure SIG4 on one pin only. In case of multiple affectation of a signal, the device is protected from damage by hardware, but only one affectation is functional.
- A signal is effectively associated to a pin only in case the corresponding mode is selected using pinout control register programming. The mode associated to the described function is highlighted in gray in the corresponding cell (pinout control register configuration). In this example:

- The MUXMODE field of the pin control register field that controls pin V4 must be written to 000 to select FUNCn.SIG1.
- Funcn.SIG2 is selected if mode2 is programmed for pin P7.
- Funcn.SIG3 is selected if mode1 is programmed for pin W4.
- Funcn.SIG4 is selected if mode1 is programmed for pin AA4 or if mode3 is programmed for pin Y3.

- Alternate functions inform on the signals that can be multiplexed on the same pin. In this example, if mode1 is programmed for pin V4, then funcx.sig2 is available in spite of FUNCn.SIG1. Alternate function visibility enables the programmer to identify potential resource conflicts on a pin. If several signals that must be used simultaneously in a user application are on the same line of the table, it must be checked that this alternate signals can be multiplexed on other pins.

- Shaded black cells in the table indicate that this mode is not used. In this example, mode4 and mode5 are never used; mode3 is not used on pin W4…

30.5.1.1 Memory Interface

A second level of multiplexing is at the module level for memory interfaces; see Chapter 12 for internal multiplexing management.

Table 31−9. GPMC Interface

GPMC	Description	DIR	Ball 2423	ALTERNATE FUNCTIONS					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPMC.A1	General Purpose Memory Address Bit1	O	M8		dss.d18		gpio.12		
GPMC.A2	General Purpose Memory Address Bit2	O	W9		dss.d19		gpio.11		
GPMC.A3	General Purpose Memory Address Bit3	O	AF10		dss.d20		gpio.10		
GPMC.A4	General Purpose Memory Address Bit4	O	W8		dss.d21		gpio.9		
GPMC.A5	General Purpose Memory Address Bit5	O	AE16		dss.d22		gpio.8		
GPMC.A6	General Purpose Memory Address Bit6	O	AF9		dss.d23		gpio.7		
GPMC.A7	General Purpose Memory Address Bit7	O	E4			sys.ndma-req2	gpio.6		
GPMC.A8	General Purpose Memory Address Bit8	O	J7			sys.ndma-req3	gpio.5		
GPMC.A9	General Purpose Memory Address Bit9	O	AE18			sys.ndma-req4	gpio.4		

Table 31−9. GPMC Interface (Continued)

GPMC	Description	DIR	Ball 2423	ALTERNATE FUNCTIONS					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPMC.A10	General Purpose Memory Address Bit10	O	D5			sys.ndma-req5	gpio.3		
GPMC.A11	General Purpose Memory Address Bit11	IO	F1						
GPMC.A12	General Purpose Memory Address Bit12	IO	F2						
GPMC.A13	General Purpose Memory Address Bit13	IO	B2						
GPMC.A14	General Purpose Memory Address Bit14	IO	D1						
GPMC.A15	General Purpose Memory Address Bit15	IO	A3						
GPMC.A16	General Purpose Memory Address Bit16	IO	D2						
GPMC.A17	General Purpose Memory Address Bit17	IO	B3						
GPMC.A18	General Purpose Memory Address Bit18	IO	B4						
GPMC.A19	General Purpose Memory Address Bit19	IO	B12						

Table 31−9. GPMC Interface (Continued)

GPMC	Description	DIR	Ball 2423	ALTERNATE FUNCTIONS					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPMC.A20	General Purpose Memory Address Bit20	IO	B7						
GPMC.A21	General Purpose Memory Address Bit21	IO	A10						
GPMC.A22	General Purpose Memory Address Bit22	IO	B11						
GPMC.A23	General Purpose Memory Address Bit23	IO	A11						
GPMC.A24	General Purpose Memory Address Bit24	IO	A6						
GPMC.A25	General Purpose Memory Address Bit25	IO	B8						
GPMC.A26	General Purpose Memory Address Bit26	IO	A7						
GPMC.D0	GPMC Data Bit 0	IO	AB2						
GPMC.D1	GPMC Data Bit 1	IO	AA2						
GPMC.D2	GPMC Data Bit 2	IO	AC2						
GPMC.D3	GPMC Data Bit 3	IO	AB1						
GPMC.D4	GPMC Data Bit 4	IO	AF3						
GPMC.D5	GPMC Data Bit 5	IO	V8						
GPMC.D6	GPMC Data Bit 6	IO	AE3						

Table 31−9. GPMC Interface (Continued)

GPMC	Description	DIR	Ball 2423	ALTERNATE FUNCTIONS					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPMC.D7	GPMC Data Bit 7	IO	AE5						
GPMC.D8	GPMC Data Bit 8	IO	V1				gpio.20		
GPMC.D9	GPMC Data Bit 9	IO	Y1		ssi2. wake		gpio.19		
GPMC.D10	GPMC Data Bit 10	IO	T1		ssi2. rdy_rx		gpio.18		
GPMC.D11	GPMC Data Bit 11	IO	U2		ssi2. flag_rx		gpio.17		
GPMC.D12	GPMC Data Bit 12	IO	U1		ssi2. dat_rx		gpio.16		
GPMC.D13	GPMC Data Bit 13	IO	P2		ssi2. rdy_tx		gpio.15		
GPMC.D14	GPMC Data Bit 14	IO	H2		ssi2. flag_tx		gpio.14		
GPMC.D15	GPMC Data Bit 15	IO	H1		ssi2. dat_tx		gpio.13		
GPMC. nCS0	GPMC Chip Select Bit 0	O	AF19						
GPMC. nCS1	GPMC Chip Select Bit 1	O	AF14				gpio.22		
GPMC. nCS2	GPMC Chip Select Bit 2	O	G4				gpio.23		
GPMC. nCS3	GPMC Chip Select Bit 3	O	T8		gpmc. io.dir		gpio.24		
GPMC. nCS4	GPMC Chip Select Bit 4	O	H8				gpio.25		
GPMC. nCS5	GPMC Chip Select Bit 5	O	K3				gpio.26		
GPMC. nCS6	GPMC Chip Select Bit 6	O	M7				gpio.27		
GPMC.nCS7	GPMC Chip Select Bit 7	O	P3		gpmc.io.dir		gpio.28		
GPMC.IO_ DIR	GPMC IO Direction Control	O	T8	gpmc.ncs3			gpio.24		

Table 31−9. GPMC Interface (Continued)

GPMC	Description	DIR	Ball 2423	ALTERNATE FUNCTIONS					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
			P3	gpmc.ncs7			gpio.28		
GPMC.CLK	GPMC Clock	IO	P1				gpio.21		
GPMC.nADV/ALE	Address Valid or Address Latch Enable	O	N8						
GPMC.nOE	Output Enable	O	N1						
GPMC.nWE	Write Enable	O	M1						
GPMC.nBE0	Lower Byte Enable. Also used for Command Latch Enable	O	AF12				gpio.29		
GPMC.nBE1	Upper Byte Enable	O	U3				gpio.30		
GPMC.nWP	Flash Write Protect	O	AE15				gpio.31		
GPMC.WAIT0	External indication of Wait	I	AE12						
GPMC.WAIT1	External indication of Wait	I	AE20				gpio.33		
GPMC.WAIT2	External Indication of Wait	I	N2				gpio.34		
GPMC.WAIT3	External Indication of Wait	I	T4				gpio.35		

Note:

The OMAP2423 SDRC interface is dedicated exclusively to stacked mobile DDR SDRAM memories; no memories can be added. To avoid problems, TI recommends that you not connect these pins. For details, see the stacked DDR controller sections in Chapter 12.

Table 31−10. SDRC Interface

SDRAM	Description	DIR	Ball 2423	ALTERNATE FUNCTIONS					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SDRC.D0	SDRAM Data Bit 0	IO	F1	Muxed with GPMC. A11					
SDRC.D1	SDRAM Data Bit 1	IO	F2	Muxed with GPMC. A12					
SDRC.D2	SDRAM Data Bit 2	IO	B2	Muxed with GPMC. A13					
SDRC.D3	SDRAM Data Bit 3	IO	D1	Muxed with GPMC. A14					
SDRC.D4	SDRAM Data Bit 4	IO	A3	Muxed with GPMC. A15					
SDRC.D5	SDRAM Data Bit 5	IO	D2	Muxed with GPMC. A16					
SDRC.D6	SDRAM Data Bit 6	IO	B3	Muxed with GPMC. A17					
SDRC.D7	SDRAM Data Bit 7	IO	B4	Muxed with GPMC. A18					
SDRC.D8	SDRAM Data Bit 8	IO	B12	Muxed with GPMC. A19					
SDRC.D9	SDRAM Data Bit 9	IO	B7	Muxed with GPMC. A20					
SDRC.D10	SDRAM Data Bit 10	IO	A10	Muxed with GPMC. A21					
SDRC.D11	SDRAM Data Bit 11	IO	B11	Muxed with GPMC.A22					
SDRC.D12	SDRAM Data Bit 12	IO	A11	Muxed with GPMC.A23					
SDRC.D13	SDRAM Data Bit 13	IO	A6	Muxed with GPMC.A24					
SDRC.D14	SDRAM Data Bit 14	IO	B8	Muxed with GPMC.A25					
SDRC.D15	SDRAM Data Bit 15	IO	A7	Muxed with GPMC.A26					
SDRC.D16	SDRAM Data Bit 16	IO							

Table 31−10. SDRC Interface (Continued)

SDRAM	Description	DIR	Ball 2423	ALTERNATE FUNCTIONS					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SDRC.D17	SDRAM Data Bit 17	IO							
SDRC.D18	SDRAM Data Bit 18	IO							
SDRC.D19	SDRAM Data Bit 19	IO							
SDRC.D20	SDRAM Data Bit 20	IO							
SDRC.D21	SDRAM Data Bit 21	IO							
SDRC.D22	SDRAM Data Bit 22	IO							
SDRC.D23	SDRAM Data Bit 23	IO	B21						
SDRC.D24	SDRAM Data Bit 24	IO							
SDRC.D25	SDRAM Data Bit 25	IO							
SDRC.D26	SDRAM Data Bit 26	IO							
SDRC.D27	SDRAM Data Bit 27	IO							
SDRC.D28	SDRAM Data Bit 28	IO							
SDRC.D29	SDRAM Data Bit 29	IO	D26						
SDRC.D30	SDRAM Data Bit 30	IO							
SDRC.D31	SDRAM Data Bit 31	IO							
SDRC. BA0	SDRAM Bank Select 0	O	AF23						
SDRC. BA1	SDRAM Bank Select 1	O	AC25						
SDRC.A0	SDRAM Address Bit 0	O	H25						
SDRC.A1	SDRAM Address Bit 1	O	M25						

Table 31−10. SDRC Interface (Continued)

SDRAM	Description	DIR	Ball 2423	ALTERNATE FUNCTIONS					
Interface				Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SDRC.A2	SDRAM Address Bit 2	O	L25						
SDRC.A3	SDRAM Address Bit 3	O	L26						
SDRC.A4	SDRAM Address Bit 4	O	P25						
SDRC.A5	SDRAM Address Bit 5	O	N25						
SDRC.A6	SDRAM Address Bit 6	O	R26						
SDRC.A7	SDRAM Address Bit 7	O	R25						
SDRC.A8	SDRAM Address Bit 8	O	T25						
SDRC.A9	SDRAM Address Bit 9	O	M26						
SDRC.A10	SDRAM Address Bit 10	O	H26						
SDRC.A11	SDRAM Address Bit 11	O	T26						
SDRC.A12	SDRAM addr Bit 12 or V-Sync Segment Addr 0	O	W26						
SDRC.A13	SDRAM Addr Bit 13	O	W25						
SDRC.A14	SDRAM Addr Bit 14	O	AA26				gpio.0		
SDRC. nCS0	Chip Select 0	O	Y26						
SDRC. nCS1	Chip Select 1	O	Y25						
SDRC. CLK	Clock	IO	A15						
SDRC. nCLK	Clock Invert	O	B16						

Table 31−10. SDRC Interface (Continued)

SDRAM	Description	DIR	Ball 2423	ALTERNATE FUNCTIONS					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SDRC. CKE0	Clock Enable0	O	AF24						
SDRC. CKE1	Clock Enable1	O	AE25						
SDRC. NRAS	SDRAM Row Access	O	AC26						
SDRC. NCAS	SDRAM Column Address Strobe	O	AA25						
SDRC. NWE	SDRAM Write Enable	O	AE23						
SDRC. DM0	Data Mask 0	O							
SDRC. DM1	Data Mask 1	O							
SDRC. DM2	Data Mask 2	O							
SDRC. DM3	Data Mask 3	O							
SDRC. DQS0	Data Strobe 0	IO							
SDRC. DQS1	Data Strobe 1	IO							
SDRC. DQS2	Data Strobe 2	IO	B19						
SDRC. DQS3	Data Strobe 3	IO	A24						

Table 31−11. GP Timer Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPT9.PWM/EVT	PWM or Event for GP Timer 9	IO	AC24	uart2.cts	usb1.rcv		gpio.67		
			L19	i2c2.scl			gpio.99		
GPT10.PWM/EVT	PWM or Event for GP Timer 10	IO	U24	spi2. simo			gpio.89		
			W20	uart2.rts	usb1. txen		gpio.68		
GPT11.PWM/EVT	PWM or Event for GP Timer 11	IO	V25	spi2. somi			gpio.90		
			AB24	uart2.tx	usb1.se0		gpio.69		
GPT12.PWM/EVT	PWM or Event for GP Timer 12	IO	AA24	spi2. ncs0			gpio.91		
			AD24	uart2.rx	usb1.dat		gpio.70		

Table 31−12. UAR/IrDA/CIR Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
UART1.CTS	UART1 Clear To Send	I	G20			dss.d18	gpio.32		
			AF15	ssi1.rdy_tx		usb1.txen	gpio.61		
UART1.RTS	UART1 Request To Send	O	K20			dss.d19	gpio.8		
			AC15	ssi1.flag_ tx		usb1.rcv	gpio.25		
UART1.RX	UART1 Receive Data	I	T20			dss.d21	gpio.10		
			AD18	gpio.62		usb1.dat	gpio.62		
UART1.TX	UART1 Transmit Data	O	H12			dss.d20	gpio.9		
			W13	ssi1.dat.tx		usb1.se0	gpio.59		
UART2.CTS	UART2 Clear To Send	I	AC24		usb1.rcv	gpt9.pwm/ evt	gpio.67		
			H20	mmc.dat2	ms.dat2		gpio.76		

Table 31−12. UAR/IrDA/CIR Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
			M24	usb0.txen	uart3.cts/ rctx		gpio.110		
			K23	usb0.rcv	mcbsp2. fsx		gpio.109	uart2.cts	
UART2.RTS	UART2 Request To Send	O	W20		usb1.txen	gpt10.pwm/ evt	gpio.68		
			D23	mmc.dat_ dir1	ms. datu_dir		gpio.78		
UART2.RX	UART2 Receive Data	I	AD24		usb1.dat	gpt12.pwm/ evt	gpio.70		
			E23	mmc.dat_ dir3	ms. datu_dir		gpio.80		
			J25	usb0.dat	uart3.rx/irrx		gpio.112	uart2.tx	
			N23	usb0.vm	mcbsp2.clkx		gpio.108		
			L24	usb0.se0	uart3.tx/irtx	uart2.tx	gpio.111		
UART2.TX	UART2 Transmit Data	O	AB24		usb1.se0	gpt11.pwm/ evt	gpio.69		
			G23	mmc.dat_ dir2	ms. datu_dir		gpio.79		
			L24	usb0.se0	uart3.tx/irtx		gpio.111	uart2.rx	
			J25	usb0.dat	uart3.rx/irrx	uart2.rx	gpio.112		
UART3.CTS/RCTX	UART3 Clear To Send (input) or Irda Transciever Remote control (output)	IO	K24		uart3.rx/irrx		gpio.102		
			M24	usb0.txen		uart2.cts	gpio.110		
UART3.RCTX	UART3 Clear To Send (input) or IrDA Transciever Remote Control (output)	O	G24	uart3.tx/irtx			gpio.104		
UART3.RTS/SD	UART3 Request To Send or IrDA Tranceiver Shutdown/ Mode Select	O	M20		uart3.tx/irtx		gpio.103		
UART3.RX/IRRX	UART3 Receive Data or IrDA Receive Data	I	H24				gpio.105		

Table 31−12. UAR/IrDA/CIR Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
			K24	uart3.cts/ rctx			gpio.102		
			J25	usb0.dat		uart2.rx	gpio.112	uart2.tx	
UART3.TX/IRTX	UART3 Transmit Data or IrDA Transmit data	O	G24		uart3.rctx		gpio.104		
			M20	uart3.rts/sd			gpio.103		
			L24	usb0.se0		uart2.tx	gpio.111	uart2.rx	

Table 31−13. I2C Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
I2C1.SCL	I2C Master Serial clock. Output is open drain	IO	P24						
I2C1.SDA	I2C Serial Bidir Data. Output is open drain	IO	R19						
I2C2.SCL	I2C Master Serial clock. Output is open drain	IO	L19			gpt9.pwm/ evt	gpio.99		
I2C2.SDA	I2C Serial Bidir Data. Output is open drain	IO	K19			spi2.ncs1	gpio.100		

Table 31−14. VLYNQ ™ Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
VLYNQ.CLK	VLYNQ Clock	IO	W12		usb2.se0	sys.ndma-req0	gpio.13		
			R24	mcbsp1.dx	ssi2.rdy_tx		gpio.94		
VLYNQ.RX0	VLYNQ Receive Channel 0	I	AD7		usb2.tllse0		gpio.15	cam_d7	
			P23	mcbsp.clks	ssi2.flag_ rx		gpio.96		
VLYNQ.RX1	VLYNQ Receive Channel 1	I	W10		usb2.rcv	sys. ndmareq1	gpio.14	cam_d8	
			U23	mcbsp1.dr	ssi2.dat_rx		gpio.95		

Table 31−14. VLYNQ ™ Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
VLYNQ.TX0	VLYNQ tRansceive Channel 0	O	Y15		usb2.txen		gpio.17		
			R20	mcbsp1.fsr	ssi2.flag_ tx		gpio.93	spi2.ncs1	
VLYNQ.TX1	VLYNQ Transceive Channel 1	O	W14		usb2.dat	sys.clkout2	gpio.16		
			Y24	mcbsp1. clkr	ssi2.dat_tx		gpio.92		
VLYNQ.nLA	Link Active − Used for Power Reduction (open drain)	IO	AE10			l4.ext_trig	gpio.58	cam_d6	
			T19	mcbsp1. clkx	ssi2.wake		gpio.98		

Table 31−15. SSI Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SSI1.DAT_TX	Transmit Data	O	W13		uart1.tx	usb1.se0	gpio.59		
SSI1.FLAG_TX	Flag Associated With Transmit Data	O	AC15		uart1.rts	usb1.rcv	gpio.25		
SSI1.RDY_TX	Tranmit Ready Signal	I	AF15		uart1.cts	usb1.txen	gpio.61		
SSI1.DAT_RX	Receive Data	I	W15		eac.md_ sclk		gpio.63		
SSI1.FLAG_RX	Flag Associated With Receive Data	I	AF11		eac.md_ din		gpio.64		
SSI1.RDY_RX	Receive Ready Signal	O	AC16		eac.md_ dout		gpio.65		
SSI1.WAKE	wake Up (optional)	O	AD15		eac.md_ fs		gpio.66		
SSI2.DAT_TX	Transmit Data	O	H1	gpmc.d15			gpio.13		
			Y24	mcbsp1. clkr		vlynq.tx1	gpio.92		
SSI2.FLAG_TX	Flag Associated With Transmit Data	O	H2	gpmc.d14			gpio.14		
			R20	mcbsp1.fsr		vlynq.tx0	gpio.93	spi2.ncs1	

Table 31−15. SSI Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SSI2.RDY_TX	Tranmit Ready Signal	I	P2	gpmc.d13			gpio.15		
			R24	mcbsp1.dx		vlynq.clk	gpio.94		
SSI2.DAT_RX	Receive Data	I	U1	gpmc.d12			gpio.16		
			U23	mcbsp1.dr		vlynq.rx1	gpio.95		
SSI2.FLAG_RX	Flag Associated With Receive Data	I	U2	gpmc.d11			gpio.17		
			P23	mcbsp.clks		vlynq.rx0	gpio.96		
SSI2.RDY_RX	Receive Ready Signal	O	T1	gpmc.d10			gpio.18		
			R23	mcbsp1. fsx			gpio.97		
SSI2.WAKE	wake Up (optional)	O	Y1	gpmc.d9			gpio.19		
			T19	mcbsp1. clkx		vlynq.nla	gpio.98		

Table 31−16. Multi-SPI Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SPI1.CLK	SPI Clock	IO	Y23				gpio.81		
SPI1.SIMO	slave Data In, Master Data Out	IO	H10				gpio.82		
SPI1.SOMI	slave Data Out, Master Data In	IO	V19				gpio.83		
SPI1.nCS0	SPI Enable 0	IO	W24				gpio.84		
SPI1.nCS1	SPI Enable 1	O	W23				gpio.85		
SPI1.nCS2	SPI Enable 2	O	V23				gpio.86		
SPI1.nCS3	SPI Enable 3	O	U20				gpio.87		
SPI2.CLK	SPI Clock	IO	V24				gpio.88		
SPI2.SIMO	slave Data In, Master Data Out	IO	U24		gpt10.pwm/ evt		gpio.89		
SPI2.SOMI	slave Data Out, Master Data In	IO	V25		gpt11.pwm/ evt		gpio.90		
SPI2.nCS0	SPI Enable 0	IO	AA24		gpt12.pwm/ evt		gpio.91		

Table 31−16. Multi-SPI Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SPI2.nCS1	SPI Enable 1	O	K19	i2c2.sda			gpio.100		
			R20	mcbsp1.fsr	ssi2. flag_tx	vlynk.tx0	gpio.93		

Table 31−17. HDQ/1-Wire Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
HDQ.SIO	Bidir HDQ 1-Wire Control and Data interface. Output is open drain	IO	T23		usb2.tllse0	sys.altclk	gpio.101		

Table 31−18. USB Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
USB0.SE0	Single Ended Zero. Used as VM in 4-pin VP_VM mode.	IO	L24		uart3.tx/irtx	uart2.tx	gpio.111	uart2.rx	
USB0.DAT	USB Data. Used as VP in 4−pin VP_VM mode.	IO	J25		uart3.rx/irrx	uart2.rx	gpio.112	uart2.tx	
USB0.TXEN	Transmit Enable	O	M24		uart3.cts/rctx	uart2.cts	gpio.110		
USB0.RCV	Differential Receiver Signal Input (not used in 3 pin mode)	I	K23		mcbsp2. fsx		gpio.109	uart2.cts	
USB0.VP	Vplus Receive Data (not used in 3 or 4 pin configurations)	I	M23		mcbsp2.dr		gpio.107		
USB0.VM	Vminus receive data (not used in 3 or 4 pin configurations)	I	N23		mcbsp2. clkx		gpio.108	uart2.rx	
USB0.PUEN	USB PullUp Enable	O	L23		mcbsp2.dx		gpio.106		
USB1.SE0	Single Ended Zero. Used as VM in 4−pin VP_VM mode, Also used as DP_TLL in 2 or 4 −pin TLL mode.	IO	AB24	uart2.tx		gpt11.pwm/ evt	gpio.69		
			W13	ssi1.dat.tx	uart1.tx		gpio.59		

Table 31−18. USB Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
USB1.DAT	USB Data. Used as VP in 4-Pin VP_VM Mode and as DN_TLL in 2 or 4 pin TLL.	IO	AD24	uart2.rx		gpt12.pwm/ evt	gpio.70		
			AD18	gpio.62	uart1.rx		gpio.62		
USB1.TXEN	Transmit Enable (not used in 2 or 3 pin TLL configurations)	O	W20	uart2.rts		gpt10.pwm/ evt	gpio.68		
			AF15	ssi1.rdy.tx	uart1.cts		gpio.61		
USB1.RCV	Differential Receiver Signal Input (not used in 2 pin configuration)	I	AC24	uart2.cts		gpt9.pwm/ evt	gpio.67		
			AC15	ssi1.flag.tx	uart1.rts		gpio.25		
USB2.SE0	Single Ended Zero. Used as VM in 4-Pin VP_VM Mode, Used as DP_TLL in 2 or 4 pin TLL mode.	IO	W12	vlynq.clk		sys. ndmareq0	gpio.13		

Table 31−18. USB Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
USB2.DAT	USB Data. Used as VP in 4-Pin VP_VM Mode and as DN_TLL in 2 pin TLL.	IO	W14	vlynq.tx1		sys.clkout2	gpio.16		
USB2.TXEN	Transmit Enable (not used in 2 pin TLL configuration),	IO	Y15	vlynq.tx0			gpio.17		
USB2.RCV	Differential Receiver Signal Input. Also used as output for RCV_TLL/VP _TLL in 5-pin TLL mode	I	W10	vlynq.rx1		sys. ndmareq1	gpio.14	cam_d8	
USB2.TLLSE0	VM_TLL Data in 5-Pin TLL Mode (not used in 2, 3, or 4 pin configurations)	I	AD7	vlynq.rx0			gpio.15	cam_d7	

Table 31−19. McBSP Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
McBSP.CLKS	External Clock Input (shared by both McBSPs)	I	P23		ssi2.flag_ rx	vlynq.rx0	gpio.96		
			AD6	cam.lclk			gpio.57		
McBSP1.DR	Received Serial Data	I	U23		ssi2.dat_rx	vlynq.rx1	gpio.95		
			AB3	cam.d3	hw.dbg5		gpio.51		
McBSP1.CLKR	Receive Clock	IO	Y24		ssi2.dat_tx	vlynq.tx1	gpio.92		
			AD3	cam.d5	hw.dbg7		gpio.49		
McBSP1.FSR	Receive Frame Synchronization	IO	R20		ssi2.flag_ tx	vlynq.tx0	gpio.93	spi2.ncs1	
			V2	cam.d4	hw.dbg6		gpio.50		

Table 31−19. McBSP Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
McBSP1.DX	Transmitted Serial Data	O	R24		ssi2.rdy_tx	vlynq.clk	gpio.94		
			V4	cam.hs	hw.dbg1		gpio.55		
McBSP1.CLKX	Transmit Clock	IO	T19		ssi2.wake	vlynq.nla	gpio.98		
			U7	cam.d2	hw.dbg4		gpio.52		
McBSP1.FSX	Transmit Frame Synchronization	IO	R23		ssi2.rdy_rx		gpio.97		
			P7	cam.vs	hw.dbg0		gpio.56		
McBSP2.DR	Received Serial Data	I	P20			dss.d22	gpio.11		
			M23	usb0.vp			gpio.107		
			AD19	eac.ac_din			gpio.115		
McBSP2.DX	TranSmitted Serial Data	O	L23	usb0.puen			gpio.106		
			AF22	eac.ac_ dout			gpio.116		
McBSP2.CLKX	Combined Serial Clock	IO	T24			dss.d23	gpio.12		
			N23	usb0.vm			gpio.108	uart2.rx	
			AC18	eac.ac_ sclk			gpio.113		
McBSP2.FSX	Combined Frame Synchronization	IO	K23	usb0.rcv			gpio.109	uart2.cts	
			AD16	eac.ac_fs			gpio.114		
			AE8	dss.acbias		mcbsp2. fsx	gpio.48		

Table 31−20. EAC Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
EAC.AC_MCLK	Audio Codec Master Clock Input or Output	IO	Y17				gpio.117		
EAC.AC_FS	Audio Codec Interface Frame Sync	IO	AD16		mcbsp2.fsx		gpio.114		
EAC.AC_SCLK	Audio Codec Interface Serial Clock	IO	AC18		mcbsp2.clkx		gpio.113		
EAC.AC_DIN	Audio Codec Interface Serial Data In	I	AD19		mcbsp2.dr		gpio.115		
EAC.AC_DOUT	Audio Codec Interface Serial Data Out	O	AF22		mcbsp2.dx		gpio.116		
EAC.AC_RST	Audio Codec Interface Reset Output (AC97 mode)	O	AE22		eac.bt_din	loop_eac.bt_din	gpio.118		
EAC.MD_SCLK	Modem voice Interface Serial Clock	IO	W15	ssi1.dat_rx			gpio.63		
EAC.MD_FS	Modem Voice Interface Frame Synchro	IO	AD15	ssi1.wake			gpio.66		
EAC.MD_DIN	Modem Voice Interface Serial Data In	I	AF11	ssi1.flag_rx			gpio.64		
EAC.MD_DOUT	Modem Voice Interface Serial Data Out	O	AC16	ssi1.rdy_rx			gpio.65		
EAC.BT_SCLK	Bluetooth Voice Interface Serial Clock	IO	AD4				gpio.71		
EAC.BT_FS	Bluetooth Voice Interface Frame Sync	IO	W7				gpio.72		
EAC.BT_DIN	BT Voice Interface Serial Data In	I	U8				gpio.73		
			AE22	eac.ac_rst		loop_eac.bt_din	gpio.118		
EAC.BT_DOUT	BT Voice Interface Serial Data Out	O	AD5			sti.clk	gpio.74		
LOOP_EAC.BT_DIN	Loop signal between EAC.BT_DIN pin and EAC.AC_RST pin	O	AE22	eac.ac_rst	eac.bt_din		gpio.118		

Special pins (sub-LVDS IO pins) are available on cam.d7, cam.d8, and cam.d9 for differential lines support.

Table 31−21. Camera Interface

Interface				Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
Function n	**Description**	**DIR**	**Ball 2423**	**Alternate Functions**					
CAM.D2	Camera Digital Image Data Bit 2	I	Y3		hw.dbg4	mcbsp1. clkx	gpio.52		
CAM.D3	Camera Digital Image Data Bit 3	I	AB3		hw.dbg5	mcbsp1.dr	gpio.51		
CAM.D4	Camera Digital Image Data Bit 4	I	V2		hw.dbg6	mcbsp1. fsr	gpio.50		
CAM.D5	Camera Digital Image Data Bit 5	I	AD3		hw.dbg7	mcbsp1. clkr	gpio.49		
CAM.D6	Camera Digital Image Data Bit 6	I	AA4		hw.dbg8				
			AE10	vlynq.nla		14.ext_trig	gpio.58		
CAM.D7	Camera Digital Image Data Bit 7	I	AB4		hw.dbg9				
			AD7	vlynq.rx0	usb2.tllse0		gpio.15		
CAM.D8	Camera Digital Image Data Bit 8	I	AC6		hw.dbg10		gpio.54		
			W10	vlynq.rx1	usb2.rcv	sys. ndmareq1	gpio.14		
CAM.D9	Camera Digital Image Data Bit 9	I	AC7		hw.dbg11		gpio.53		
			AF4	gpio.120			gpio.120		
CAM.LCLK	Camera Image Data Latch clock	I	AD6			mcbsp.clks	gpio.57		

Table 31−22. Display Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
DSS.PCLK	LCD Pixel Clock	IO	AF5						
DSS.HSYNC	LCD Horizental Synchronization	IO	AD8						
DSS.VSYNC	LCD Vertical Synchronization	IO	AF8						
DSS.ACBIAS	AC Bias Control (STN) or Pixel Data Enable (TFT) Output; also A0 output (Command/Data selection) in config 1	O	AE8			mcbsp2. fsx	gpio.48		
DSS.D0	LCD Pixel Data Bit 0	IO	Y9						
DSS.D1	LCD Pixel Data Bit 1	IO	AC9						
DSS.D2	LCD Pixel Data Bit 2	IO	AC8						
DSS.D3	LCD Pixel Data Bit 3	IO	AE9						
DSS.D4	LCD Pixel Data Bit 4	IO	AD9						
DSS.D5	LCD Pixel Data Bit 5	IO	AD10						
DSS.D6	LCD Pixel Data Bit 6	IO	W11						
DSS.D7	LCD Pixel Data Bit 7	IO	AC11						
DSS.D8	LCD Pixel Data Bit 8	IO		dss.d8			gpio.38		
DSS.D9	LCD Pixel Data Bit 9	IO		dss.d9			gpio.39		
DSS.D10	LCD Pixel Data Bit 10	IO		dss.d10			gpio.40		
DSS.D11	LCD Pixel Data Bit 11	IO		dss.d11			gpio.41		
DSS.D12	LCD Pixel Data Bit 12	IO		dss.d12			gpio.42		
DSS.D13	LCD Pixel Data Bit 13	IO		dss.d13			gpio.43		
DSS.D14	LCD Pixel Data Bit 14	IO		dss.d14			gpio.44		
DSS.D15	LCD Pixel Data Bit 15	IO		dss.d15			gpio.45		
DSS.D16	LCD Pixel Data Bit 16	IO		dss.d16			gpio.46		

Table 31−22. Display Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
DSS.D17	LCD Pixel Data Bit 17	IO		dss.d17			gpio.47		
DSS.D18	LCD Pixel Data Bit 18	IO	M8	gpmc.a1			gpio.12		
			G20	uart1.cts			gpio.32		
DSS.D19	LCD Pixel Data Bit 19	IO	W9	gpmc.a2			gpio.11		
			K20	uart1.rts			gpio.8		
DSS.D20	LCD Pixel Data Bit 20	O	AF10	gpmc.a3			gpio.10		
			H12	uart1.tx			gpio.9		
DSS.D21	LCD Pixel Data Bit 21	O	W8	gpmc.a4			gpio.9		
			T20	uart1.rx			gpio.10		
DSS.D22	LCD Pixel Data Bit 22	O	AE16	gpmc.a5			gpio.8		
			P20	mcbsp2.dr			gpio.11		
DSS.D23	LCD Pixel Data Bit 23	O	AF9	gpmc.a6			gpio.7		
			T24	mcbsp2. clkx			gpio.12		

Table 31−23. TV Out Interface

TV	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
TV.CVBS	Composite video output (analog)	O	AD23	tv_cvbs					
TV.VREF	Input voltage reference (analog)	I	AB26	tv_vref					
TV.RREF	External resistor reference (analog − 4 kΩ)	I	AC23	tv_rref					
TV.DETECT	Pulse Detection	O	D6	gpio.6	tv.detect		gpio.6		

Table 31−24. MMC/SDIO Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
MMC.CLKO	MMC/SD Output Clock	IO	H23		ms.clko				
MMC.CLKI	MMC/SD Input Clock	I	C23		ms.clki		gpio.59		
MMC.CMD	MMC/SD Command Signal	IO	J23		ms.bs				
MMC.DAT0	MMC/SD Card Data bit 0/SPI Serial Input	IO	H17		ms.dat0				
MMC.DAT1	MMC/SD Card Data Bit 1	IO	G19		ms.dat1		gpio.75		
MMC.DAT2	MMC/SD Card Data Bit 2	IO	H20		ms.dat2	uart2.cts	gpio.76		
MMC.DAT3	MMC/SD Card Data Bit 3	IO	D24		ms.dat3	l4.ext_trig	gpio.77		
MMC.CMD_DIR	MMC/SD Command Direction	O	J24				gpio.8		
MMC.DAT_DIR0	MMC/SD Data Direction Bit 0	O	F23		ms.dat0_ dir		gpio.7		
MMC.DAT_DIR1	MMC/SD Data Direction Bit 1	O	D23		ms. datu_dir	uart2.rts	gpio.78		
MMC.DAT_DIR2	MMC/SD Data Direction Bit 2	O	G23		ms. datu_dir	uart2.tx	gpio.79		
MMC.DAT_DIR3	MMC/SD data direction bit 3	O	E23		ms. datu_dir	uart2.rx	gpio.80		

The internal connections between OMAP2420 die and DDR SDRAM stacked memories prohibit connecting GPIO.1, GPIO.2, GPIO.37, and GPIO.38 on the customer board.

Table 31−25. GPIO Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.0	General Purpose IO 0	IO	AA26						
GPIO.1	General Purpose IO 1	IO	W25						
GPIO.2	General Purpose IO 2	IO	W26						

† Input only. Nothing is multiplexed with SYS.CLKOUT output clock to avoid the risk of perturbation on clock signal.

Table 31−25. GPIO Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.3	General Purpose IO 3	IO	D5	gpmc.a10		sys.ndmareq5			
GPIO.4	General Purpose IO 4	IO	AE18	gpmc.a9		sys.ndmareq4			
GPIO.5	General Purpose IO 5	IO	J7	gpmc.a8		sys.ndmareq3			
GPIO.6	General Purpose IO 6	IO	D6		tv.detect		gpio.6		
			E4	gpmc.a7		sys.ndmareq2			
			D5	gpio.6	tv.detect				
GPIO.7	General Purpose IO 7	IO	AF9	gpmc.a6	dss.d23				
			F23	mmc.dat_dir0	ms.dat0_dir				
GPIO.8	General Purpose IO 8	IO	AE16	gpmc.a5	dss.d22				
			K20	uart1.rts		dss.d19			
			J24	mmc.cmd_dir					
GPIO.9	General Purpose IO 9	IO	W8	gpmc.a4	dss.d21				
			H12	uart1.tx		dss.d20			
GPIO.10	General Purpose IO 10	IO	AF10	gpmc.a3	dss.d20				
			T20	uart1.rx		dss.d21			
GPIO.11	General Purpose IO 11	IO	W9	gpmc.a2	dss.d19				
			P20	mcbsp2.dr		dss.d22			
GPIO.12	General Purpose IO 12	IO	M8	gpmc.a1	dss.d18				
			T24	mcbsp2.clkx		dss.d23			
GPIO.13	General Purpose IO 13	IO	H1	gpmc.d15	ssi2.dat_tx				
			W12	vlynq.clk	usb2.se0	sys.ndmareq0			
GPIO.14	General Purpose IO 14	IO	H2	gpmc.d14	ssi2.flag_tx				
			W10	vlynq.rx1	usb2.rcv	sys.ndmareq1		cam.d8	
GPIO.15	General Purpose IO 15	IO	P2	gpmc.d13	ssi2.rdy_tx				
			AD7	vlynq.rx0	usb2.tllse0			cam.d7	

† Input only. Nothing is multiplexed with SYS.CLKOUT output clock to avoid the risk of perturbation on clock signal.

Table 31−25. GPIO Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.16	General Purpose IO 16	IO	U1	gpmc.d12	ssi2.dat_rx				
			W14	vlynq.tx1	usb2.dat	sys.clkout2			
GPIO.17	General Purpose IO 17	IO	U2	gpmc.d11	ssi2.flag_rx				
			Y15	vlynq.tx0	usb2.txen				
GPIO.18	General Purpose IO 18	IO	T1	gpmc.d10	ssi2.rdy_rx				
GPIO.19	General Purpose IO 19	IO	Y1	gpmc.d9	ssi2.wake				
GPIO.20	General Purpose IO 20	IO	V1	gpmc.d8					
GPIO.21	General Purpose IO 21	IO	P1	gpmc.clk					
GPIO.22	General Purpose IO 22	IO	AF14	gpmc.ncs1					
GPIO.23	General Purpose IO 23	IO	G4	gpmc.ncs2					
GPIO.24	General Purpose IO 24	IO	T8	gpmc.ncs3	gpmc.io_dir				
GPIO.25	General Purpose IO 25	IO	H8	gpmc.ncs4					
			AC15	ssi1.flag_tx	uart1.rts	usb1.rcv			
GPIO.26	General Purpose IO 26	IO	K3	gpmc.ncs5					
GPIO.27	General Purpose IO 27	IO	M7	gpmc.ncs6					
GPIO.28	General Purpose IO 28	IO	P3	gpmc.ncs7	gpmc.io_dir				
GPIO.29	General Purpose IO 29	O	AF12	gpmc.nbe0					
GPIO.30	General Purpose IO 30	O	U3	gpmc.nbe1					
GPIO.31	General Purpose IO 31	O	AE15	gpmc.nwp					
GPIO.32	General Purpose IO 32	IO	G20	uart1.cts		dss.d18			
GPIO.33	General Purpose IO 33	IO	AE20	gpmc.wait1					
GPIO.34	General Purpose IO 34	IO	N2	gpmc.wait2					
GPIO.35	General Purpose IO 35	IO	T4	gpmc.wait3					

† Input only. Nothing is multiplexed with SYS.CLKOUT output clock to avoid the risk of perturbation on clock signal.

Table 31−25. GPIO Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.36	General Purpose IO 36	IO	Y18				gpio.36		sys.boot4
			Y18	gpio.36					sys.boot4
GPIO.37	General Purpose IO 37	IO	Y25						
GPIO.38	General Purpose IO 38	IO	AE25						
			AD11	dss.d8					
GPIO.39	General Purpose IO 39	IO	AD12	dss.d9					
GPIO.40	General Purpose IO 40	IO	AC12	dss.d10					
GPIO.41	General Purpose IO 41	IO	AE11	dss.d11					
GPIO.42	General Purpose IO 42	IO	AE13	dss.d12					
GPIO.43	General Purpose IO 43	IO	AD13	dss.d13					
GPIO.44	General Purpose IO 44	IO	AC13	dss.d14					
GPIO.45	General Purpose IO 45	IO	Y12	dss.d15					
GPIO.46	General Purpose IO 46	IO	AD14	dss.d16					
GPIO.47	General Purpose IO 47	IO	Y13	dss.d17					
GPIO.48	General Purpose IO 48	IO	AE8	dss.acbias		mcbsp2.fsx			
GPIO.49	General Purpose IO 49	IO	AD3	cam.d5	hw.dbg7	mcbsp1.clkr			
GPIO.50	General Purpose IO 50	IO	V2	cam.d4	hw.dbg6	mcbsp1.fsr			
GPIO.51	General Purpose IO 51	IO	AB3	cam.d3	hw.dbg5	mcbsp1.dr			
GPIO.52	General Purpose IO 52	IO	U7	cam.d2	hw.dbg4	mcbsp1.clkx			
		I†	AD20	sys.clkreq					
GPIO.53	General Purpose IO 53	IO	AC7	cam.d9	hw.dbg11				
			Y3	cam.d1	hw.dbg3	sti.din			
GPIO.54	General Purpose IO 54	IO	AC6	cam.d8	hw.dbg10				

† Input only. Nothing is multiplexed with SYS.CLKOUT output clock to avoid the risk of perturbation on clock signal.

Table 31−25. GPIO Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
			Y4	cam.d0	hw.dbg2	sti.dout			
GPIO.55	General Purpose IO 55	IO	V4	cam.hs	hw.dbg1	mcbsp1.dx			
GPIO.56	General Purpose IO 56	IO	P7	cam.vs	hw.dbg0	mcbsp1.fsx			
GPIO.57	General Purpose IO 57	IO	AD6	cam.lclk		mcbsp.clks			
GPIO.58	General Purpose IO 58	IO	AE10	vlynq_nla		l4_ext_trig		cam.d6	
GPIO.59	General Purpose IO 59	IO	W13	ssi1.dat_tx	uart1.tx	usb1.se0			
			C23	mmc.clki	ms.clki				
GPIO.60	General Purpose IO 60	I	Y20	sys_nirq					
GPIO.61	General Purpose IO 61	IO	AF15	ssi1.rdy_tx	uart1.cts	usb1.txen			
GPIO.62	General Purpose IO 62	IO	AD18		uart1.rx	usb1.dat	gpio.62		
			AD18	gpio.62	uart1.rx	usb1.dat			
GPIO.63	General Purpose IO 63	IO	W15	ssi1.dat_rx	eac.md_sclk				
GPIO.64	General Purpose IO 64	IO	AF11	ssi1.flag_rx	eac.md_din				
GPIO.65	General Purpose IO 65	IO	AC16	ssi1.rdy_rx	eac.md_dout				
GPIO.66	General Purpose IO 66	IO	AD15	ssi1.wake	eac.md_fs				
GPIO.67	General Purpose IO 67	IO	AC24	uart2.cts	usb1.rcv	gpt9.pwm/evt			
GPIO.68	General Purpose IO 68	IO	W20	uart2.rts	usb1.txen	gpt10.pwm/evt			
GPIO.69	General Purpose IO 69	IO	AB24	uart2.tx	usb1.se0	gpt11.pwm/evt			
GPIO.70	General Purpose IO 70	IO	AD24	uart2.rx	usb1.dat	gpt12.pwm/evt			
GPIO.71	General Purpose IO 71	IO	AD4	eac.bt_sclk					
GPIO.72	General Purpose IO 72	IO	W7	eac.bt_fs					
GPIO.73	General Purpose IO 73	IO	U8	eac.bt_din					
GPIO.74	General Purpose IO 74	IO	AD5	eac.bt_dout		sti.clk			

† Input only. Nothing is multiplexed with SYS.CLKOUT output clock to avoid the risk of perturbation on clock signal.

Table 31−25. GPIO Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.75	General Purpose IO 75	IO	G19	mmc.dat1	ms.dat1				
GPIO.76	General Purpose IO 76	IO	H20	mmc.dat2	ms.dat2	uart2.cts			
GPIO.77	General Purpose IO 77	IO	D24	mmc.dat3	ms.dat3	l4.ext_trig			
GPIO.78	General Purpose IO 78	IO	D23	mmc.dat_dir1	ms.datu_dir	uart2.rts			
GPIO.79	General Purpose IO 79	IO	G23	mmc.dat_dir2	ms.datu_dir	uart2.tx			
GPIO.80	General Purpose IO 80	IO	E23	mmc.dat_dir3	ms.datu_dir	uart2.rx			
GPIO.81	General Purpose IO 81	IO	Y23	spi1.clk					
GPIO.82	General Purpose IO 82	IO	H10	spi1.simo					
GPIO.83	General Purpose IO 83	IO	V19	spi1.somi					
GPIO.84	General Purpose IO 84	IO	W24	spi1.ncs0					
GPIO.85	General Purpose IO 85	IO	W23	spi1.ncs1					
GPIO.86	General Purpose IO 86	IO	V23	spi1.ncs2					
GPIO.87	General Purpose IO 87	IO	U20	spi1.ncs3					
GPIO.88	General Purpose IO 88	IO	V24	spi2.clk					
GPIO.89	General Purpose IO 89	IO	U24	spi2.simo	gpt10.pwm/ evt				
GPIO.90	General Purpose IO 90	IO	V25	spi2.somi	gpt11.pwm/ evt				
GPIO.91	General Purpose IO 91	IO	AA24	spi2.ncs0	gpt12.pwm/ evt				
GPIO.92	General Purpose IO 92	IO	Y24	mcbsp1.clkr	ssi2.dat_tx	vlynq.tx1			
GPIO.93	General Purpose IO 93	IO	R20	mcbsp1.fsr	ssi2.flag_tx	vlynq.tx0		spi2.ncs1	
GPIO.94	General Purpose IO 94	IO	R24	mcbsp1.dx	ssi2.rdy_tx	vlynq.clk			
GPIO.95	General Purpose IO 95	IO	U23	mcbsp1.dr	ssi2.dat_rx	vlynq.rx1			
GPIO.96	General Purpose IO 96	IO	P23	mcbsp.clks	ssi2.flag_rx	vlynq.rx0			

† Input only. Nothing is multiplexed with SYS.CLKOUT output clock to avoid the risk of perturbation on clock signal.

Table 31−25. GPIO Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.97	General Purpose IO 97	IO	R23	mcbsp1.fsx	ssi2.rdy_rx				
GPIO.98	General Purpose IO 98	IO	T19	mcbsp1.clkx	ssi2.wake	vlynq.nla			
GPIO.99	General Purpose IO 99	IO	L19	i2c2.scl		gpt9.pwm/evt			
GPIO.100	General Purpose IO 100	IO	K19	i2c2.sda		spi2.ncs1			
GPIO.101	General Purpose IO 101	IO	T23	hdq.sio	usb2.tllse0	sys.altclk			
GPIO.102	General Purpose IO 102	IO	K24	uart3.cts/rctx	uart3.rx/irrx				
GPIO.103	General Purpose IO 103	IO	M20	uart3.rts/sd	uart3.tx/irtx				
GPIO.104	General Purpose IO 104	IO	G24	uart3.tx/irtx	uart3.rctx				
GPIO.105	General Purpose IO 105	IO	H24	uart3.rx/irrx					
GPIO.106	General Purpose IO 106	IO	L23	usb0.puen	mcbsp2.dx				
GPIO.107	General Purpose IO 107	IO	M23	usb0.vp	mcbsp2.dr				
GPIO.108	General Purpose IO 108	IO	N23	usb0.vm	mcbsp2.clkx			uart2.rx	
GPIO.109	General Purpose IO 109	IO	K23	usb0.rcv	mcbsp2.fsx			uart2.cts	
GPIO.110	General Purpose IO 110	IO	M24	usb0.txen	uart3.cts/rctx	uart2.cts			
GPIO.111	General Purpose IO 111	IO	L24	usb0.se0	uart3.tx/irtx	uart2.tx		uart2.rx	
GPIO.112	General Purpose IO 112	IO	J25	usb0.dat	uart3.rx/irrx	uart2.rx		uart2.tx	
GPIO.113	General Purpose IO 113	IO	AC18	eac.ac_sclk	mcbsp2.clkx				
GPIO.114	General Purpose IO 114	IO	AD16	eac.ac_fs	mcbsp2.fsx				

† Input only. Nothing is multiplexed with SYS.CLKOUT output clock to avoid the risk of perturbation on clock signal.

Table 31−25. GPIO Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
GPIO.115	General Purpose IO 115	IO	AD19	eac.ac_din	mcbsp2.dr				
GPIO.116	General Purpose IO 116	IO	AF22	eac.ac_dout	mcbsp2.dx				
GPIO.117	General Purpose IO 117	IO	Y17	eac.ac_mclk					
GPIO.118	General Purpose IO 118	IO	AE22	eac.ac_rst	eac.bt_din	loop_eac.bt_din			
GPIO.119	General Purpose IO 119	IO	AF6				gpio.119		sys.boot0
			AF6	gpio.119					sys.boot0
GPIO.120	General Purpose IO 120	IO	AF4				gpio.120	cam.d9	sys.boot1
			AF4	gpio.120				cam.d9	sys.boot1
GPIO.121	General Purpose IO 121	IO	AE6				gpio.121	jtag.emu2	sys.boot2
			AE6	gpio.121				jtag.emu2	sys.boot2
GPIO.122	General Purpose IO 122	IO	W3				gpio.122	jtag.emu3	sys.boot3
			W3	gpio.122				jtag.emu3	sys.boot3
GPIO.123	General Purpose IO 123	I†	AE19	sys.clkout					
GPIO.124	General Purpose IO 124	IO	Y19				gpio.124		sys.boot5
			Y19	gpio.124					sys.boot5
GPIO.125	General Purpose IO 125	IO	AE24		sys. jtagsel1	sys. jtagsel2	gpio.125		
			AE24	gpio.125	sys. jtagsel1	sys. jtagsel2			
GPIO.126	General Purpose IO 126	IO	AC22						
GPIO.127	General Purpose IO 127	IO	N24						

† Input only. Nothing is multiplexed with SYS.CLKOUT output clock to avoid the risk of perturbation on clock signal.

Table 31−26. System Interface

Function n	Description	DIR	Ball 2423	Alternate Functions					
	Interface			Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SYS.32K	32 kHz Clock Input	I	AD21						
SYS.XTALIN	Main Input Clock. Oscillator input or LVCMOS at 19.2, 13, or 12 Mhz.	IO	AC20						
SYS.XTALOUT	Output of Oscillator	O	AC19						
SYS.ALTCLK	Alternate Clock Source Selectable for GPTIMERs (max 54 MHz), USB (48 MHz) or NTSC/PAL (54 MHz).	I	T23	hdq.sio	usb2.tllse0		gpio.101		
SYS.CLKREQ	Request From OMAP24xx Device for system Clock	O	AD20				gpio.52		
SYS.CLKOUT	Configurable Output Clock	O	AE19				gpio.123		
SYS.CLKOUT2	Configurable Output Clock 2	O	W14	vlynq.tx1	usb2.dat		gpio.16		
SYS.BOOT0	Boot Configuration Mode Bit 0	I	AF6	gpio.119			gpio.119		
SYS.BOOT1	Boot Configuration Mode Bit 1	I	AF4	gpio.120			gpio.120	cam.d9	
SYS.BOOT2	Boot Configuration Mode Bit 2	I	AE6	gpio.121			gpio.121	jtag.emu2	
SYS.BOOT3	Boot Configuration Mode Bit 3	I	W3	gpio.122			gpio.122	jtag.emu3	
SYS.BOOT4	Boot Configuration Mode Bit 4	I	Y18	gpio.36			gpio.36		
SYS.BOOT5	Boot Configuration Mode Bit 5	I	Y19	gpio.124			gpio.124		
SYS.nRESPWRON	Power On Reset	I	W17						
SYS.nRESWARM	Warm Boot Reset (open drain output)	IO	AE2						

Table 31−26. System Interface (Continued)

Function n	Description	DIR	Ball 2423	Alternate Functions					
Interface				Mode0	Mode1	Mode2	Mode3	Mode4	Mode5
SYS.nIRQ	Battery Voltage Failure Detection/External FIQ Input	I	Y20				gpio.60		
SYS.nVMODE	Indicates the Voltage Mode	O	W16						
SYS.JTAGSEL1	Enables Alternate Pin Mapping for JTAG Port	I	AE24	gpio.125	sys. jtagsel1	sys. jtagsel2	gpio.125		
SYS.JTAGSEL2	Enables Alternate Pin Mapping for JTAG port	I	AE24	gpio.125	sys. jtagsel1		gpio.125		
SYS.nDMAREQ0	Ext DMA Request	I	W12	vlynk.clk	usb2.se0		gpio.13		
SYS.nDMAREQ1	Ext DMA Request	I	W10	vlynq.rx1	usb2.rcv		gpio.14	cam.d8	
SYS.nDMAREQ2	Ext DMA Request	I	E4	gpmc.a7			gpio.6		
SYS.nDMAREQ3	Ext DMA Request	I	J7	gpmc.a8			gpio.5		
SYS.nDMAREQ4	Ext DMA Request	I	AE18	gpmc.a9			gpio.4		
SYS.nDMAREQ5	Ext DMA Request	I	D5	gpmc.a10			gpio.3		

30.6 Programming Model

See Section 30.6 in the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

30.7 Pinout Control Module Registers

See Section 30.7 in the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

30.8 Acronyms, Abbreviations, and Definitions

See Section 30.8 in the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 31

Security Features

This chapter describes the security features of the OMAP2423 multimedia system-in-package (SIP) processor. All of the security features in this chapter refer to general-purpose devices unless otherwise specified. Contact your TI representative for more information on high-security devices.

This chapter is equivalent to Chapter 31 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 32

Device Booting

This chapter describes the high-level booting concepts on the OMAP2423 multimedia system-in-package (SIP) processor.

For the latest detailed information about booting on the OMAP2423 device, contact your TI representative to obtain the latest version of the *OMAP2410/20 Booting and Flashing ROM Code Functional Specification*.

This chapter is equivalent to Chapter 32 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Chapter 33

Initialization

This chapter provides an overview of the requirements for initializing the OMAP2423 multimedia system-in-package (SIP) device from power-on to the point of operating system (OS) load and applications running. The chapter includes an overview of the overall initialization process, with both hardware- and software-related steps, a general overview of the boot ROM operational requirements, and behavior expectations.

This chapter does not offer a detailed review of the OMAP™ boot ROM functionality. For this information, see Chapter 32.

Topic **Page**

33.1 Introduction **33-2**
33.2 Pre-Initialization Requirements **33-2**
33.3 Power-On Voltage-Clock-Reset Sequence **33-3**
33.4 Device Initialization by ROM Code **33-5**
33.5 RAM Partitioning During Device Initialization **33-5**
33.6 Device Types **33-5**

33.1 Introduction

See Section 33.1 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

33.2 Pre-Initialization Requirements

Successful boot-up with the OMAP2423 device requires specific hardware configuration settings. Clock, reset, and power connections, as well as pads involved in setting the boot memory space for the MPU, must be connected and driven properly for successful device initialization. This section details the specific requirements for the system interface ports.

33.2.1 Power Connections

Table 34−1. OMAP2423 Voltage Levels

Voltage	Description	Operating Levels (Volts)			Absolute Min-Max (Volts)	
		Min	Nom	Max	Min	Max
VDD	OMAP processor cores and logic operating at full speed	1.235	1.3	1.372	0.5	1.7
	OMAP processor cores and logic operating in low voltage functional mode	0.99	1.05	1.11		
VDDS VDDS2	Supply voltage for I/O macros (except GPMC, SDRC)	1.71	1.8	1.89	0.5	2.43
VDDS1	Supply voltage for I/O to memory interfaces (GPMC, SDRC) and to stacked DDR SDRAM DQ buffers	1.71	1.8	1.89	0.5	2.43
DDR_VDD	Core supply voltage for stacked DDR SDRAM memories	1.71	1.8	1.89	−1.0	2.6
DDR_VSS	Core ground for stacked DDR SRAM memories		0			
DDR_VDDQ†	Supply voltage for stacked mobile DDR SRAM output buffers	1.71	1.8	1.89	−1.0	2.6
DDR_VSSQ	Ground for stacked DDR SDRAM output buffers		0			
VDDA	Analog supply voltage for video DAC	1.71	1.8	1.89	0.5	2.43
VDD_PLL	Supply voltage for APLL and DPLLs	1.235	1.3	1.372	0.5	1.7
VDD_DLL‡	Supply voltage for SDRC DLL core operating at full speed	1.235	1.3	1.372	0.5	1.7
	Supply voltage for SDRC DLL core operating in low-voltage functional mode	0.99	1.05	1.11		
VSS	Common ground			0		
VSSA	Analog ground for video DAC			0		

† DDR_VDDQ must be equal to DDR_VDD.
‡ VDD_DLL voltage must match VDD voltage.

33.2.2 System Interface

See Section 33.2.2 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

33.2.3 Clock Configuration

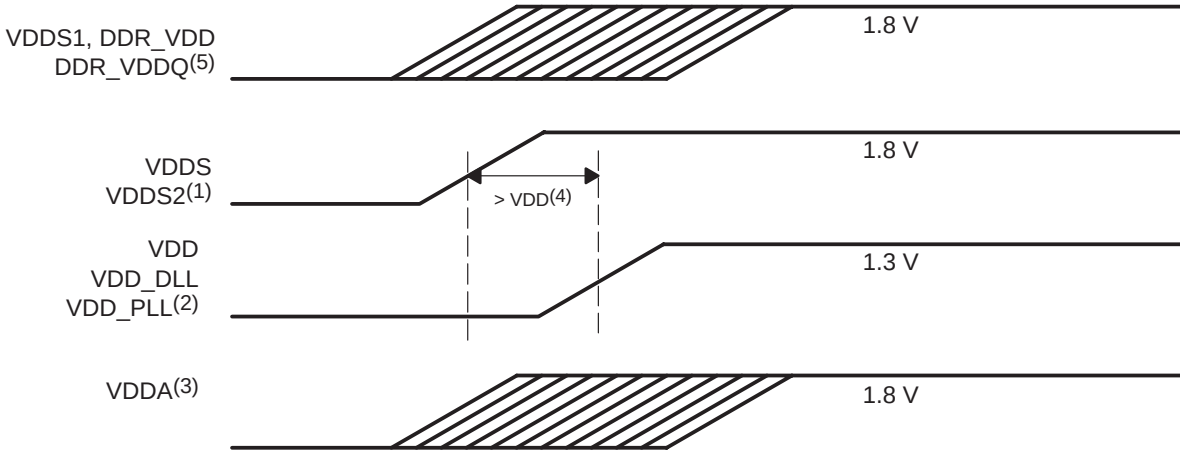
See Section 33.2.3 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

33.3 Power-On Voltage-Clock-Reset Sequence

33.3.1 Power-Up Voltage Sequencing

Figure 33−3 shows the required voltage ramp sequence at power-on for the OMAP2423 device. For power-up requirements, see the *OMAP2423 Data Manual*.

Figure 33−3. Voltages Ramp Sequence at Power-up



(1) VDDS and VDDS2 must maintain a voltage level greater than or equal to VDD.
(2) VDD and VDD_DLL must ramp at the same time (but can have up to 0.6 V difference during the voltage ramp period).
(3) There are no power-on timing or voltage level requirements for VDDA relative to the other supplies. VDDA can ramp either before or after VDD.
(4) The diode between VDDS2/VDD and VDDS/VDD must not be forward biased by letting VDD become greater than VDDS2 and VDDS.
(5) There are no power-on timing requirements for VDDS1, DDR_VDD, and DDR_VDDQ relative to the other supplies. On the other hand, DDR_VDDQ must never ramp high before VDDS1 because SDRC interface data/IOs are not failsafe.

Note:

When power is applied on DDR_VDD and DDR_VDDQ, a pause of 200 µs or longer is required before the two stacked mobile DDR SDRAMs can be accessed.

33.3.2 Power, Clock, Reset Sequence on Power-Up

Table 34−12 and Figure 33−4 show the power-on sequence for the OMAP2423 device, including the clocks, resets and SYS.BOOT signals and their relationship to the core and PLL voltages. These steps are explained in consecutive order.

Table 34−12. Power-on Sequence Event Descriptions

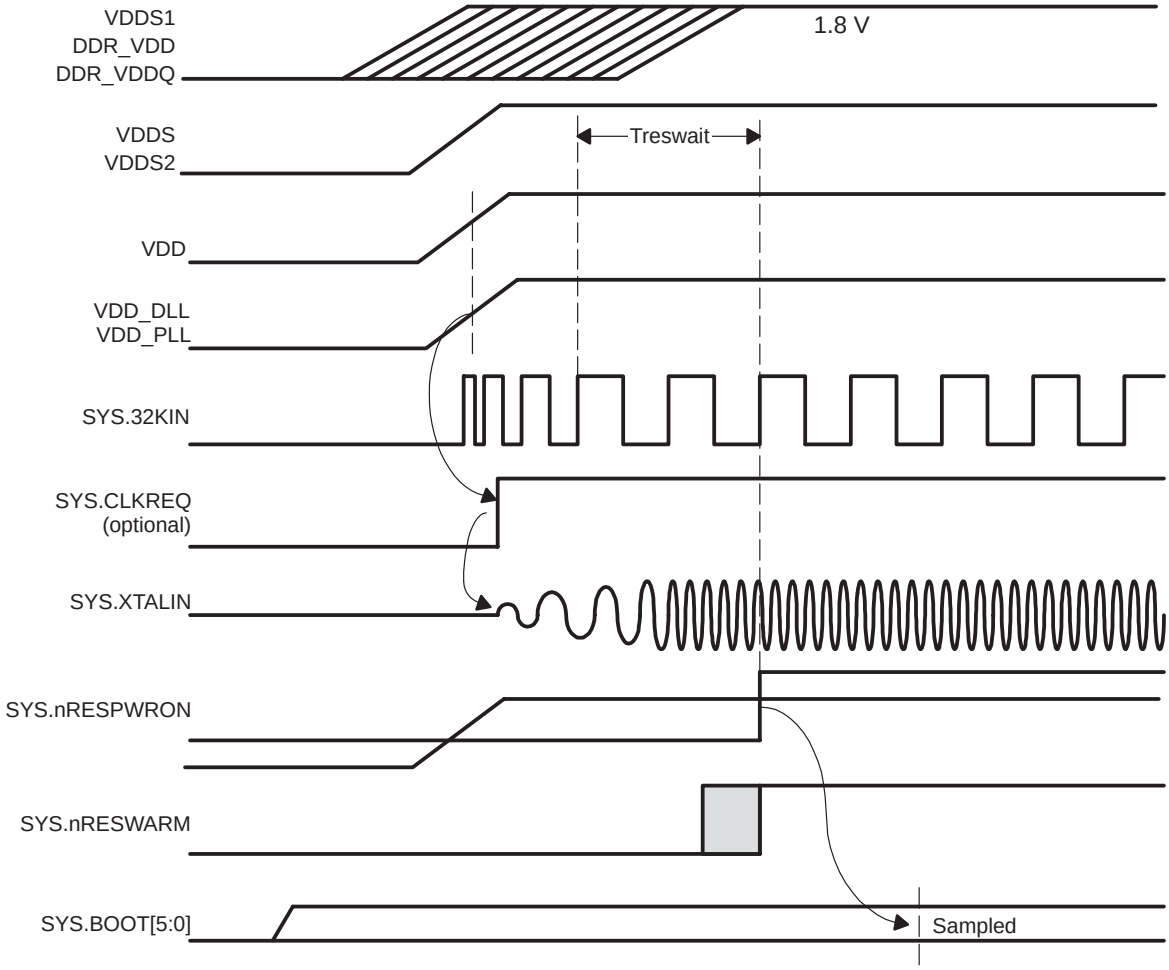
Step	Event Title	Event Description
0	IO supply rails, DDR_VDD, and DDR_VDDQ ramp high†	IO power supply rails, VDDS and VDDS2, are first powered on. VDDS1, DDR_VDD, and DDR_VDDQ are also powered on.
1	Core and PLL supply rails ramp high	Core power supply VDD and VDD_DLL, VDD_PLL power supplies are then powered on.
2	SYS.CLKREQ activated (optional)	Optional SYS.CLKREQ output requests the system clock to be activated by external system.
3	SYS.XTALIN activated	Crystal or input square wave system clock is activated by external system.
4	Power-on reset	For a power-on condition, SYS.nRESPWRON is deasserted (low-to-high transition) after meeting the *Treswait* timing condition. While SYS.nRESPWRON is low, all OMAP2423 IOs are forced to be in their reset state, regardless of the voltage on VDD.
5	SYS.BOOT pins sampled	SYS.BOOT pins are sampled before (SYS.nRESPWRON rising edge + 231 µs). (Device type fuse array must be read and internal memories initialized prior to reading the SYS.BOOT configuration pins.)

† DDR_VDD and DDR_VDDQ must be equal.

WARNING

VDDS1 must ramp high before (worst case, at the same time) as DDR_VDDQ to avoid possible destructive damage.

Figure 33−4. Power-up Sequence



Treswait = Two valid 32k clock cycles or the crystal oscillator start up time (5–8 ms), whichever occurs first.

When the voltages and clocks are ramped and stable, SYS.nRESPWRON is deasserted (high), and the SYS.BOOT pins sampled, the device is ready to enable the boot ROM to run.

33.4 Device Initialization by ROM Code

See Section 33.4 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

33.5 RAM Partitioning During Device Initialization

See Section 33.5 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

33.6 Device Types

See Section 33.6 of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Appendix A

Glossary

This appendix is equivalent to Appendix A of the *OMAP2420 Silicon Revision 2.1.1, 2.2 TRM*, version D (SWPU064D).

Appendix B

512Mb x 32 DDR Mobile RAM Memory Specification

This appendix presents the Elpida 512Mb × 32 DDR mobile RAM memory specification.

Appendix C

256Mb x 32 DDR Mobile RAM Memory Specification

This appendix presents the Elpida 256Mb × 32 DDR mobile RAM memory specification.

Index

C

clock
  clock manager description, configurations, 5-2
  configuration, initialization requirements, 33-3
  distribution, clock manager, 5-2
  generation, 5-2

clock manager
  clock configurations, 5-2
  functional description, 5-2

clock manager description
  clock distribution, 5-2
  clock generation, 5-2

clocking, reset, and power management
  GPMC integration, 12-9
  stacked DDR controller integration, 12-22

D

debug support, OMAP2422, 1-10

description
  GPIO integration, 29-7
  GPMC integration, 12-9
  OCM subsystem integration, 12-108
  OMAP2422, 1-4
  reset manager, 5-9
  stacked DDR controller environment, 12-19
  stacked DDR controller integration, 12-21

device types, initialization, 33-5

DMA controllers, OMAP2422, 1-7

DSP subsystem, OMAP2422, 1-6

DSP subsystem memory space, memory mapping, 2-6

E

emulation support, power reset and clock manager, 5-9
environment
  GPIO, 29-5
  GPMC, 12-4
  power reset and clock manager, 5-2
external memory interfaces, OMAP2422, 1-7

F

features, stacked DDR controller overview, 12-17
functional description
  clock manager, 5-2
  GPIO, 29-17
  GPMC, 12-14
  power management, 5-9
functional interfaces, GPIO environment, 29-7
functional overview, GPIO, 29-2

G

global features, GPIO functional overview, 29-2
global memory space, memory mapping, 2-4
GPIO
  description, 29-7
  environment, 29-5
  functional description, 29-17
  functional interfaces, 29-7
  functional overview, 29-2
  global features, 29-2
  integration, 29-7
  programming model, 29-17
  registers, 29-17
GPMC
  environment, 12-4
  functional description, 12-14
  integration, 12-9
  overview, 12-2
  programming model, 12-14
  registers, 12-14

stacked DDR controller environment, 12-19
stacked DDR controller functional description, 12-30
stacked DDR controller integration, 12-21
stacked DDR controller overview, 12-15
stacked DDR controller programming model, 12-45
stacked DDR controller registers, 12-64
GPMC integration
clocking, reset, and power scheme, 12-9
description, 12-9
GPMC register summary, 12-13
pin list and pad multiplexing, 12-13
GPMC register summary, GPMC integration, 12-13

I

initialization
device types, 33-5
initialization by ROM code, 33-5
power, clock, reset sequence at power−up, 33-3 , 33-4
preinitialization requirements, 33-2
RAM partitioning, 33-5
requirements
clock configuration, 33-3
power connections, 33-2
system interface, 33-3
initialization by ROM code, initialization, 33-5
instance summary, stacked DDR controller registers, 12-64
integration
GPIO, 29-7
GPMC, 12-9
power reset and clock manager, 5-2
interfacing OMAP2420 and stacked DDR SDRAM, stacked DDR controller integration, 12-23
interrupts, power reset and clock manager, 5-9
introduction
initialization, 33-2
memory mapping, 2-2
IVA memory space, memory mapping, 2-6

L

L3, L4 interconnect registers, memory mapping, 2-6

M

memory mapping
DSP subsystem memory space, 2-6
global memory space, 2-4
introduction, 2-2
IVA memory space, 2-6
L3, L4 interconnect registers, 2-6
mobile DDR RAM, stacked DDR controller functional description, 12-43
MPU subsystem, OMAP2422, 1-6
multimedia accelerators, OMAP2422, 1-7

O

OCM subsystem description, OCM_ROM, 12-108
OCM subsystem integration
description, 12-108
overview, 12-108
OCM_ROM, OCM subsystem description, 12-108
OMAP2422
debug support, 1-10
description, 1-4
DMA controllers, 1-7
DSP subsystem, 1-6
external memory interfaces, 1-7
MPU subsystem, 1-6
multimedia accelerators, 1-7
onchip memory, 1-6
overview, 1-2
peripherals, 1-8
power management, 1-8
security, 1-8
onchip memory, OMAP2422, 1-6
overview
GPMC, 12-2
OCM subsystem integration, 12-108
OMAP2422, 1-2
power reset and clock manager, 5-2

P

peripherals, OMAP2422, 1-8
pin list and pad multiplexing, GPMC integration, 12-13
power connections, initialization requirements, 33-2
power management
functional description, 5-9
OMAP2422, 1-8
power reset and clock manager
emulation support, 5-9
environment, 5-2
integration, 5-2
interrupts, 5-9
overview, 5-2
programming model, 5-9
registers, 5-9
sleep and wake−up, 5-9
power, clock, reset sequence, initialization power−on voltage−clock reset, 33-3 , 33-4
preinitialization requirements, initialization, 33-2
programming model
GPIO, 29-17
GPMC, 12-14

power reset and clock manager, 5-9

R

RAM partitioning, initialization, 33-5
register descriptions, stacked DDR controller registers, 12-66
register summary
stacked DDR controller integration, 12-27
stacked DDR controller registers, 12-64
registers
GPIO, 29-17
GPMC, 12-14
power reset and clock manager, 5-9
reset manager, description, 5-9

S

SDRAM controller, stacked DDR controller functional description, 12-35
SDRAM controller programming model, stacked DDR controller programming model, 12-49
SDRAM memory scheduler, stacked DDR controller functional description, 12-30
SDRC register descriptions, stacked DDR controller registers, 12-81
security, OMAP2422, 1-8
sleep and wake−up, power reset and clock manager, 5-9
SMS programming model, stacked DDR controller programming model, 12-45
stacked DDR controller environment
description, 12-19
GPMC, 12-19
stacked DDR controller functional description
GPMC, 12-30
mobile DDR RAM, 12-43
SDRAM controller, 12-35
SDRAM memory scheduler, 12-30
stacked DDR controller integration
clocking, reset, and power management, 12-22
description, 12-21
GPMC, 12-21
interfacing OMAP2420 and stacked DDR SDRAM, 12-23
register summary, 12-27
stacked DDR controller overview
features, 12-17
GPMC, 12-15
stacked DDR controller programming model
SMS programming model, 12-45
GPMC, 12-45
SDRAM controller programming model, 12-49
stacked DDR controller registers
GPMC, 12-64
instance summary, 12-64
register descriptions, 12-66
register summary, 12-64
SDRC register descriptions, 12-81
system interface, initialization requirements, 33-3

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265