MC9S12XEP100 FCS
Full Chip Simulation Implementation Note
Covers MC9S12XE Family

HCS12

CodeWarrior Tools

MC9S12XEP100 FCS

Rev. 1.00

11/2006

freescale.com

freescale semiconductor

1 Introduction

1.1 Overview of MC9S12XE Family

The MC9S12XE-Family of micro controllers is a further development of the S12XD-Family including new features for enhanced system integrity and greater functionality. These new features include a Memory Protection Unit (MPU) and Error Correction Code (ECC) on the Flash memory together with enhanced EEPROM functionality (EEE), an enhanced XGATE, an internally filtered, frequency modulated Phase Locked Loop (IPLL) and an enhanced ATD. The E-Family extends the S12X product range up to 1MB of Flash memory with increased I/O capability in the 208-pin version of the flagship MC9S12XE100. For more details on the MC9S12XE-Family of micro controllers XE Family Data Sheet can be referred.

1.2 MC9S12XE Full Chip Simulation

Full Chip Simulation means not only to simulate the core instruction set but also the on chip I/O devices such as (CRG, PWM, ECT, ...). In the section Simulation Summary the supported I/O devices are listed for each supported derivative of MC9S12XE-Family.

HC(S) 12(X) Debugger Manual can be referred for the details of configuring HCS12 Debugger to use Full Chip Simulation connection. The Full Chip Simulation (FCS) connection runs a complete simulation of all processor peripherals and I/O on the user's Personal Computer. No development board is required. Each derivative has a totally different simulation engine to accurately simulate the memory ranges, I/O, and peripherals for a given derivative.

1.3 Scope

This document details the Full Chip Simulation support for I/O devices for the supported derivatives of MC9S12XE-Family. It details the features that are simulated, not simulated, or simulated with exceptions.

1.4 References

Item	Description
MC9S12XE100.pdf	MC9S12XEP100 Data Sheet, Covers MC9S12XE Family
Debugger_HC12.pdf	HC(S) 12(X) Debugger Manual.

2 Simulation Summary

The following table gives a summary of peripheral options of MC9S12XE Family members. Each derivative for XE family comes with different memory, peripheral and package options. The number of peripherals for each derivative has been mentioned in the table. The modules which have been simulated are hyperlinked and details regarding them can be seen be selecting the hyperlink.

<table>
<tr><th colspan="2">Derivative →</th><th rowspan="2">XEP100</th><th rowspan="2">XEP768</th><th rowspan="2">XEQ512</th><th rowspan="2">XEQ384</th><th rowspan="2">XET256</th><th rowspan="2">XEG128</th></tr>
<tr><th>Peripheral Module</th><th>Module Version</th></tr>
<tr><td>EPIM (port integration module)</td><td>S12XEPIMV1</td><td>1</td><td>1</td><td>1 [6]</td><td>1 [6]</td><td>1 [6]</td><td>1 [7]</td></tr>
<tr><td>MMC (memory mapping control)</td><td>S12XMMCV4 [1][2]</td><td colspan="6">1</td></tr>
<tr><td>EBI (external bus interface)</td><td>S12XEBIV4 [2]</td><td colspan="6">1</td></tr>
<tr><td>DBG (debug module)</td><td>S12XDBGV1 [2]</td><td colspan="6">1</td></tr>
<tr><td>ECRG (clock and reset generator)</td><td>S12XECRGV1</td><td colspan="6">1</td></tr>
<tr><td>ECT (enhanced capture timer)</td><td>S12ECT16B8CV2</td><td colspan="6">1</td></tr>
<tr><td>ATD (analog-to-digital converter)</td><td>ADC12B16CV1</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>1</td></tr>
<tr><td>IIC (inter IC bus)</td><td>IICV3 [3]</td><td>2</td><td>2</td><td>2</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>SCI (serial communications interface)</td><td>S12SCIV5</td><td>8</td><td>8</td><td>6/4/2 [8]</td><td>4/2 [8]</td><td>4/2 [8]</td><td>2</td></tr>
<tr><td>SPI (serial peripheral interface)</td><td>S12SPIV5 [3]</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>2</td></tr>
<tr><td>FTM control registers</td><td>S12XFTMV0 [2]</td><td colspan="6">1</td></tr>
<tr><td>MPU (memory protection unit)</td><td>S12XMPUV1</td><td colspan="6">1</td></tr>
<tr><td>INT (interrupt module)</td><td>S12XINTV2 [4]</td><td colspan="6">1</td></tr>
<tr><td>CAN (controller area network)</td><td>S12MSCANV2 [3]</td><td>5</td><td>5</td><td>4</td><td>4</td><td>3</td><td>2</td></tr>
<tr><td>VREG (voltage regulator)</td><td>S12VREGL3V3V1</td><td colspan="6">1</td></tr>
<tr><td>PWM (pulse width modulator)</td><td>S12PWM8B8CV1</td><td colspan="6">1</td></tr>
<tr><td>PIT (periodic interrupt timer)</td><td>S12PIT24B8CV1</td><td colspan="2">1</td><td colspan="4">-</td></tr>
<tr><td>PIT (periodic interrupt timer)</td><td>S12PIT24B4CV1</td><td colspan="2">-</td><td colspan="3">1</td><td>-</td></tr>
<tr><td>PIT (periodic interrupt timer)</td><td>S12PIT24B2CV1</td><td colspan="5">-</td><td>1</td></tr>
<tr><td>CPU</td><td>S12XCPUV2 [4]</td><td colspan="6">1</td></tr>
<tr><td>XGATE</td><td>S12XGATEV3 [4]</td><td colspan="6">1</td></tr>
<tr><td>TIM (timer module)</td><td>TIM16B8CV2</td><td colspan="2">1</td><td colspan="4">-</td></tr>
<tr><td>BDM (background debug module)</td><td>S12XBDMV2 [2][5]</td><td colspan="6">1</td></tr>
<tr><td>OSC (pierce oscillator)</td><td>S12XOSCLCPV1 [2][5]</td><td colspan="6">1</td></tr>
</table>

1) Any memory is integral part of the core simulator.
2) This module is not simulated.
3) Registers of this module are implemented to allow read and write access but no module specific functionality.
4) This module is fully implemented and part of the core simulator.
5) This Module is not required in context of simulator.
6) Ports F, L and R and associated registers are not available.
7) Ports C, D, F, L and R and associated registers are not available.
8) Number of accessible modules depends on package – the highest numbers of modules are simulated.

3 Peripheral Modules – Simulation Details

3.1 Port Integration Module (S12XEPIMV1)

3.1.1 Implementation of PIM module (S12XEPIMV1)

Major functionalities of GPIO are implemented. GPIO functionalities like reduced output drive, wired-or functions are not implemented. Static routing and dynamic routing (routing based on routing registers) are implemented only for peripherals PWM, TIM, ECT and VREG. Free-running clock outputs and IRQ functionality are not implemented.

3.1.2 Extensions

PIM establishes the interface between the peripheral modules and the I/O pins for all ports. All the port pins are simulated. Routing of signals only for peripherals like PWM, TIM, ECT and VREG are implemented.

The ports and corresponding pins that are available are:

PortA – PIM.PORTAPin0 to PIM.PORTAPin7

PortB – PIM.PORTBPin0 to PIM.PORTBPin7

PortC – PIM.PORTCPin0 to PIM.PORTCPin7

PortD – PIM.PORTDPin0 to PIM.PORTDPin7

PortE – PIM.PORTEPin0 to PIM.PORTEPin7

PortK – PIM.PORTKPin0 to PIM.PORTKPin7

PortT – PIM.PORTTPin0 to PIM.PORTTPin7

PortS – PIM.PORTSPin0 to PIM.PORTSPin7

PortM – PIM.PORTMPin0 to PIM.PORTMPin7

PortP – PIM.PORTPPin0 to PIM.PORTPPin7

PortH – PIM.PORTHPin0 to PIM.PORTHPin7

PortJ – PIM.PORTJPin0 to PIM.PORTJPin7

PortAD0 – PIM.PORTAD0Pin0 to PIM.PORTAD0Pin15

PortAD1 – PIM.PORTAD1Pin0 to PIM.PORTAD1Pin15

PortR – PIM.PORTRPin0 to PIM.PORTRPin7

PortL – PIM.PORTLPin0 to PIM.PORTLPin7

PortF – PIM.PORTFPin0 to PIM.PORTFPin7

3.1.3 Restrictions

Following is the list of the registers which are not fully simulated:
All Reduced Drive Registers (RDRIV, RDRT, RDRS, RDRM, RDRP, RDRH, RDRJ, RDR0AD0, RDR1AD0, RDR0AD1, RDR1AD1, RDRR, RDRL, and RDRF) functionalities are not implemented but read/write to same has been provided.
All Wired-Or Mode Registers (WOMS, WOMM, WOML) functionalities are not implemented but read/write to same has been provided.
Routing registers MODRR, PTLRR, PTFRR are not implemented but read/write to same has been provided.
ECLK Control Register is not implemented and hence free running clock outputs are not provided.
IRQ Control Register is not implemented and hence IRQ is not supported.
Registers at the address at 0x001D, 0x001F, 0x0246, 0x0247, 0x024F, 0x036E, 0x037E are reserved for factory testing of the PIM module and are not available in normal modes.

3.1.4 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 PORTA	R W	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
0x0001 PORTB	R W	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
0x0002 DDRA	R W	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
0x0003 DDRB	R W	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
0x0004 PORTC	R W	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
0x0005 PORTD	R W	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
0x0006 DDRC	R W	DDRC7	DDRC6	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0
0x0007 DDRD	R W	DDRD7	DDRD6	DDRD5	DDRD4	DDRD3	DDRD2	DDRD1	DDRD0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0008	R	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
PORTE	W								
0x0009	R	DDRE7	DDRE6	DDRE5	DDRE4	DDRE3	DDRE2	0	0
DDRE	W								
0x000C	R	PUPKE	BKPUE	0	PUPEE	PUPDE	PUPCE	PUPBE	PUPAE
PUCR	W								
0x000D	R	RDPK	0	0	RDPE	RDPD	RDPC	RDPB	RDPA
RDRIV	W								
0x001C	R	NECLK	NCLKX2	DIV16	EDIV4	EDIV3	EDIV2	EDIV1	EDIV0
ECLKCTL	W								
0x001D	R	0	0	0	0	0	0	0	0
RESERVED	W								
0x001E	R	IRQE	IRQEN	0	0	0	0	0	0
IRQCR	W								
0x001F	R	0	0	0	0	0	0	0	0
RESERVED	W								
0x0032	R	PK7	PK6	PK5	PK4	PK3	PK2	PK1	PK0
PORTK	W								
0x0033	R	DDRK7	DDRK6	DDRK5	DDRK4	DDRK3	DDRK2	DDRK1	DDRK0
DDRK	W								
0x0240	R	PTT7	PTT6	PTT5	PTT4	PTT3	PTT2	PTT1	PTT0
PTT	W								
0x0241	R	PTIT7	PTIT6	PTIT5	PTIT4	PTIT3	PTIT2	PTIT1	PTIT0
PTIT	W								
0x0242	R	DDRT7	DDRT6	DDRT5	DDRT4	DDRT3	DDRT2	DDRT1	DDRT0
DDRT	W								
0x0243	R	RDRT7	RDRT6	RDRT5	RDRT4	RDRT3	RDRT2	RDRT1	RDRT0
RDRT	W								
0x0244	R	PERT7	PERT6	PERT5	PERT4	PERT3	PERT2	PERT1	PERT0
PERT	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0245 PPST	R W	PPST7	PPST6	PPST5	PPST4	PPST3	PPST2	PPST1	PPST0
0x0246 RESERVED	R	0	0	0	0	0	0	0	0
	W								
0x0247 RESERVED	R	0	0	0	0	0	0	0	0
	W								
0x0248 PTS	R W	PTS7	PTS6	PTS5	PTS4	PTS3	PTS2	PTS1	PTS0
0x0249 PTIS	R	PTIS7	PTIS6	PTIS5	PTIS4	PTIS3	PTIS2	PTIS1	PTIS0
	W								
0x024A DDRS	R W	DDRS7	DDRS6	DDRS5	DDRS4	DDRS3	DDRS2	DDRS1	DDRS0
0x024B RDRS	R W	RDRS7	RDRS6	RDRS5	RDRS4	RDRS3	RDRS2	RDRS1	RDRS0
0x024C PERS	R W	PERS7	PERS6	PERS5	PERS4	PERS3	PERS2	PERS1	PERS0
0x024D PPSS	R W	PPSS7	PPSS6	PPSS5	PPSS4	PPSS3	PPSS2	PPSS1	PPSS0
0x024E WOMS	R W	WOMS7	WOMS6	WOMS5	WOMS4	WOMS3	WOMS2	WOMS1	WOMS0
0x024F RESERVED	R	0	0	0	0	0	0	0	0
	W								
0x0250 PTM	R W	PTM7	PTM6	PTM5	PTM4	PTM3	PTM2	PTM1	PTM0
0x0251 PTIM	R	PTIM7	PTIM6	PTIM5	PTIM4	PTIM3	PTIM2	PTIM1	PTIM0
	W								
0x0252 DDRM	R W	DDRM7	DDRM6	DDRM5	DDRM4	DDRM3	DDRM2	DDRM1	DDRM0
0x0253 RDRM	R W	RDRM7	RDRM6	RDRM5	RDRM4	RDRM3	RDRM2	RDRM1	RDRM0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0254 PERM	R W	PERM7	PERM6	PERM5	PERM4	PERM3	PERM2	PERM1	PERM0
0x0255 PPSM	R W	PPSM7	PPSM6	PPSM5	PPSM4	PPSM3	PPSM2	PPSM1	PPSM0
0x0256 WOMM	R W	WOMM7	WOMM6	WOMM5	WOMM4	WOMM3	WOMM2	WOMM1	WOMM0
0x0257 MODRR	R	0	MODRR6	MODRR5	MODRR4	MODRR3	MODRR2	MODRR1	MODRR0
	W								
0x0258 PTP	R W	PTP7	PTP6	PTP5	PTP4	PTP3	PTP2	PTP1	PTP0
0x0259 PTIP	R	PTIP7	PTIP6	PTIP5	PTIP4	PTIP3	PTIP2	PTIP1	PTIP0
	W								
0x025A DDRP	R W	DDRP7	DDRP6	DDRP5	DDRP4	DDRP3	DDRP2	DDRP1	DDRP0
0x025B RDRP	R W	RDRP7	RDRP6	RDRP5	RDRP4	RDRP3	RDRP2	RDRP1	RDRP0
0x025C PERP	R W	PERP7	PERP6	PERP5	PERP4	PERP3	PERP2	PERP1	PERP0
0x025D PPSP	R W	PPSP7	PPSP6	PPSP5	PPSP4	PPSP3	PPSP2	PPSP1	PPSP0
0x025E PIEP	R W	PIEP7	PIEP6	PIEP5	PIEP4	PIEP3	PIEP2	PIEP1	PIEP0
0x025F PIFP	R W	PIFP7	PIFP6	PIFP5	PIFP4	PIFP3	PIFP2	PIFP1	PIFP0
0x0260 PTH	R W	PTH7	PTH6	PTH5	PTH4	PTH3	PTH2	PTH1	PTH0
0x0261 PTIH	R	PTIH7	PTIH6	PTIH5	PTIH4	PTIH3	PTIH2	PTIH1	PTIH0
	W								
0x0262 DDRH	R W	DDRH7	DDRH6	DDRH5	DDRH4	DDRH3	DDRH2	DDRH1	DDRH0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0263 RDRH	R W	RDRH7	RDRH6	RDRH5	RDRH4	RDRH3	RDRH2	RDRH1	RDRH0
0x0264 PERH	R W	PERH7	PERH6	PERH5	PERH4	PERH3	PERH2	PERH1	PERH0
0x0265 PPSH	R W	PPSH7	PPSH6	PPSH5	PPSH4	PPSH3	PPSH2	PPSH1	PPSH0
0x0266 PIEH	R W	PIEH7	PIEH6	PIEH5	PIEH4	PIEH3	PIEH2	PIEH1	PIEH0
0x0267 PIFH	R W	PIFH7	PIFH6	PIFH5	PIFH4	PIFH3	PIFH2	PIFH1	PIFH0
0x0268 PTJ	R W	PTJ7	PTJ6	PTJ5	PTJ4	PTJ3	PTJ2	PTJ1	PTJ0
0x0269 PTIJ	R	PTIJ7	PTIJ6	PTIJ5	PTIJ4	PTIJ3	PTIJ2	PTIJ1	PTIJ0
	W								
0x026A DDRJ	R W	DDRJ7	DDRJ6	DDRJ5	DDRJ4	DDRJ3	DDRJ2	DDRJ1	DDRJ0
0x026B RDRJ	R W	RDRJ7	RDRJ6	RDRJ5	RDRJ4	RDRJ3	RDRJ2	RDRJ1	RDRJ0
0x026C PERJ	R W	PERJ7	PERJ6	PERJ5	PERJ4	PERJ3	PERJ2	PERJ1	PERJ0
0x026D PPSJ	R W	PPSJ7	PPSJ6	PPSJ5	PPSJ4	PPSJ3	PPSJ2	PPSJ1	PPSJ0
0x026E PIEJ	R W	PIEJ7	PIEJ6	PIEJ5	PIEJ4	PIEJ3	PIEJ2	PIEJ1	PIEJ0
0x026F PIFJ	R W	PIFJ7	PIFJ6	PIFJ5	PIFJ4	PIFJ3	PIFJ2	PIFJ1	PIFJ0
0x0270 PT0AD0	R W	PT0AD015	PT0AD014	PT0AD013	PT0AD012	PT0AD011	PT0AD010	PT0AD09	PT0AD08
0x0271 PT1AD0	R W	PT1AD07	PT1AD06	PT1AD05	PT1AD04	PT1AD03	PT1AD02	PT1AD01	PT1AD00

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0272 DDR0AD0	R W	DDR0AD015	DDR0AD014	DDR0AD013	DDR0AD012	DDR0AD011	DDR0AD010	DDR0AD09	DDR0AD08
0x0273 DDR1AD0	R W	DDR1AD07	DDR1AD06	DDR1AD05	DDR1AD04	DDR1AD03	DDR1AD02	DDR1AD01	DDR1AD00
0x0274 RDR0AD0	R W	RDR0AD015	RDR0AD014	RDR0AD013	RDR0AD012	RDR0AD011	RDR0AD010	RDR0AD09	RDR0AD08
0x0275 RDR1AD0	R W	RDR1AD07	RDR1AD06	RDR1AD05	RDR1AD04	RDR1AD03	RDR1AD02	RDR1AD01	RDR1AD00
0x0276 PER0AD0	R W	PER0AD015	PER0AD014	PER0AD013	PER0AD012	PER0AD011	PER0AD010	PER0AD09	PER0AD08
0x0277 PER1AD0	R W	PER1AD07	PER1AD06	PER1AD05	PER1AD04	PER1AD03	PER1AD02	PER1AD01	PER1AD00
0x0278 PT0AD1	R W	PT0AD115	PT0AD114	PT0AD113	PT0AD112	PT0AD111	PT0AD110	PT0AD19	PT0AD18
0x0279 PT1AD1	R W	PT1AD17	PT1AD16	PT1AD15	PT1AD14	PT1AD13	PT1AD12	PT1AD11	PT1AD10
0x027A DDR0AD1	R W	DDR0AD115	DDR0AD114	DDR0AD113	DDR0AD112	DDR0AD111	DDR0AD110	DDR0AD19	DDR0AD18
0x027B DDR1AD1	R W	DDR1AD17	DDR1AD16	DDR1AD15	DDR1AD14	DDR1AD13	DDR1AD12	DDR1AD11	DDR1AD10
0x027C RDR0AD1	R W	RDR0AD115	RDR0AD114	RDR0AD113	RDR0AD112	RDR0AD111	RDR0AD110	RDR0AD19	RDR0AD18
0x027D RDR1AD1	R W	RDR1AD17	RDR1AD16	RDR1AD15	RDR1AD14	RDR1AD13	RDR1AD12	RDR1AD11	RDR1AD10
0x027E PER0AD1	R W	PER0AD115	PER0AD114	PER0AD113	PER0AD112	PER0AD111	PER0AD110	PER0AD19	PER0AD18
0x027F PER1AD1	R W	PER1AD17	PER1AD16	PER1AD15	PER1AD14	PER1AD13	PER1AD12	PER1AD11	PER1AD10
0x0368 PTR	R W	PTR7	PTR6	PTR5	PTR4	PTR3	PTR2	PTR1	PTR0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0369	R	PTIR7	PTIR6	PTIR5	PTIR4	PTIR3	PTIR2	PTIR1	PTIR0
PTIR	W								
0x036A	R	DDRR7	DDRR6	DDRR5	DDRR4	DDRR3	DDRR2	DDRR1	DDRR0
DDRR	W								
0x036B	R	RDRR7	RDRR6	RDRR5	RDRR4	RDRR3	RDRR2	RDRR1	RDRR0
RDRR	W								
0x036C	R	PERR7	PERR6	PERR5	PERR4	PERR3	PERR2	PERR1	PERR0
PERR	W								
0x036D	R	PPSR7	PPSR6	PPSR5	PPSR4	PPSR3	PPSR2	PPSR1	PPSR0
PPSR	W								
0x036E	R	0	0	0	0	0	0	0	0
RESERVED	W								
0x036F	R	PTRRR7	PTRRR6	PTRRR5	PTRRR4	PTRRR3	PTRRR2	PTRRR1	PTRRR0
PTRRR	W								
0x0370	R	PTL7	PTL6	PTL5	PTL4	PTL3	PTL2	PTL1	PTL0
PTL	W								
0x0371	R	PTIL7	PTIL6	PTIL5	PTIL4	PTIL3	PTIL2	PTIL1	PTIL0
PTIL	W								
0x0372	R	DDRL7	DDRL6	DDRL5	DDRL4	DDRL3	DDRL2	DDRL1	DDRL0
DDRL	W								
0x0373	R	RDRL7	RDRL6	RDRL5	RDRL4	RDRL3	RDRL2	RDRL1	RDRL0
RDRL	W								
0x0374	R	PERL7	PERL6	PERL5	PERL4	PERL3	PERL2	PERL1	PERL0
PERL	W								
0x0375	R	PPSL7	PPSL6	PPSL5	PPSL4	PPSL3	PPSL2	PPSL1	PPSL0
PPSL	W								
0x0376	R	WOML7	WOML6	WOML5	WOML4	WOML3	WOML2	WOML1	WOML0
WOML	W								
0x0377	R	PTLRR7	PTLRR6	PTLRR5	PTLRR4	0	0	0	0
PTLRR	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0378 PTF	R/W	PTF7	PTF6	PTF5	PTF4	PTF3	PTF2	PTF1	PTF0
0x0379 PTIF	R	PTIF7	PTIF6	PTIF5	PTIF4	PTIF3	PTIF2	PTIF1	PTIF0
	W								
0x037A DDRF	R/W	DDRF7	DDRF6	DDRF5	DDRF4	DDRF3	DDRF2	DDRF1	DDRF0
0x037B RDRF	R/W	RDRF7	RDRF6	RDRF5	RDRF4	RDRF3	RDRF2	RDRF1	RDRF0
0x037C PERF	R/W	PERF7	PERF6	PERF5	PERF4	PERF3	PERF2	PERF1	PERF0
0x037D PPSF	R/W	PPSF7	PPSF6	PPSF5	PPSF4	PPSF3	PPSF2	PPSF1	PPSF0
0x037E RESERVED	R	0	0	0	0	0	0	0	0
	W								
0x037F PTFRR	R	0	0	PTFRR5	PTFRR4	PTFRR3	PTFRR2	PTFRR1	PTFRR0
	W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.2 Memory Mapping Control (S12XMMCV4)

3.2.1 Implementation

The MMC module has not been fully implemented and provides only the basic functionality. The following registers have been implemented in S12XMMC_V4

Global Page Index Register (GPAGE)
Direct Page Register (DIRECT)
Program Page Index Register (PPAGE)
RAM Page Index Register (RPAGE)
EEPROM Page Index Register (EPAGE)

3.2.2 Restrictions

The MMC peripheral does not provide any simulated functionality for the following set of registers neither these registers have been implemented

MMC Control Register (MMCCTL0)

The MMCCTL0 register is used to control external bus functions, like:

Availability of chip selects (available only in Normal Expanded and Emulation expanded mode)
Control of different external stretch mechanism

Mode Register (MODE)

The MODE bits of the MODE register are used to establish the MCU operating mode. The external mode pins MODC, MODB, and MODA determine the operating mode during RESET low (active). The state of the pins is latched into the respective register bits after the RESET signal goes inactive

MMC Control Register (MMCCTL1)

The individual bits in this register control the visibility of the Flash in the memory map for CPU or BDM (not for XGATE). Both local and global memory maps are affected

3.2.3 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x000A MMCCTL0	R W	CS0E7	CS0E6	CS0E5	CS0E4	CS0E3	CS0E2	CS0E1	CS0E0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x000B MODE	R	MODC	MODB	MODA	0	0	0	0	0
	W								
0x0010 GPAGE	R	0	GP06	GP05	GP04	GP03	GP02	GP1	GP0
	W								
0x0011 DIRECT	R	DP15	DP14	DP13	DP12	DP11	DP10	DP9	DP8
	W								
0x0012 Reserved	R	0	0	0	0	0	0	0	0
	W								
0x0013 MMCTL1	R	TGMRAMON	MGROMON	EEEIFRON	PGMIFRON	RAMHM	EROMON	ROMHM	ROMON
	W								
0x0014 Reserved	R	0	0	0	0	0	0	0	0
	W								
0x0015 PPAGE	R	PIX7	PIX6	PIX5	PIX4	PIX3	PIX2	PIX1	PIX0
	W								
0x0016 RPAGE	R	RP7	RP6	RP5	RP4	RP3	RP2	RP1	RP0
	W								
0x0017 EPAGE	R	EP7	EP6	EP5	EP4	EP3	EP2	EP1	EP0
	W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.3 Memory Protection Unit (S12XMPUV1)

3.3.1 Implementation

All the functionality of MPU module is implemented. Bus master 3 support is also provided there but no module is assigned to it yet.

3.3.2 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers or bits not or only partially implemented will be color encoded.

<table>
<tr><th>Register Offset / Name</th><th></th><th>Bit 7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>Bit 0</th></tr>
<tr><td>0x0000</td><td>R</td><td rowspan="2">AEF</td><td>WPF</td><td>NEXF</td><td>0</td><td>0</td><td>0</td><td>0</td><td>SVSF</td></tr>
<tr><td>MPUFLG</td><td>W</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>0x0001</td><td>R</td><td>0</td><td colspan="7">ADDR[22:16]</td></tr>
<tr><td>MPUASTAT0</td><td>W</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>0x0002</td><td>R</td><td colspan="8">ADDR[15:8]</td></tr>
<tr><td>MPUASTAT1</td><td>W</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>0x0003</td><td>R</td><td colspan="8">ADDR[7:0]</td></tr>
<tr><td>MPUASTAT2</td><td>W</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>0x0004</td><td>R</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Reserved</td><td>W</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>0x0005</td><td>R</td><td rowspan="2">SVSEN</td><td>0</td><td>0</td><td>0</td><td>0</td><td rowspan="2" colspan="3">SEL[2:0]</td></tr>
<tr><td>MPUSEL</td><td>W</td><td></td><td></td><td></td><td></td></tr>
<tr><td>0x0006</td><td>R</td><td rowspan="2">MSTR0</td><td rowspan="2">MSTR1</td><td rowspan="2">MSTR2</td><td rowspan="2">MSTR3</td><td rowspan="2" colspan="4">LOW_ADDR[22:19]</td></tr>
<tr><td>MPUDESC0</td><td>W</td></tr>
<tr><td>0x0007</td><td>R</td><td rowspan="2" colspan="8">LOW_ADDR[18:11]</td></tr>
<tr><td>MPUDESC1</td><td>W</td></tr>
<tr><td>0x0008</td><td>R</td><td rowspan="2" colspan="8">LOW_ADDR[10:3]</td></tr>
<tr><td>MPUDESC2</td><td>W</td></tr>
<tr><td>0x0009</td><td>R</td><td rowspan="2">WP</td><td rowspan="2">NEX</td><td>0</td><td>0</td><td rowspan="2" colspan="4">HIGH_ADDR[22:19]</td></tr>
<tr><td>MPUDESC3</td><td>W</td><td></td><td></td></tr>
</table>

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000	R	AEF	WPF	NEXF	0	0	0	0	SVSF
MPUFLG	W								
0x000A	R	HIGH_ADDR[18:11]							
MPUDESC4	W								
0x000B	R	HIGH_ADDR[10:3]							
MPUDESC5	W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

Note: The module addresses 0x0006–0x000B represent a window in the register map through which different descriptor registers are visible.

3.4 Clocks and Reset Generator (S12XECRGV1)

3.4.1 Implementation

The ECRG peripheral module evolved out of CRG module. Primitive CRG has a Synthesizer Divider, and a Reference Divider to generate various PLL output frequencies. S12XECRG has an additional Post Divider, and that makes the “Clock Factor” to be a function of Synthesizer, Reference and Post Dividers.

fPLL = 2 * fOSC * [SYNDIV + 1] / ([REFDIV + 1] * [2 * POSTDIV])

3.4.2 Extensions

The following is the list of external signals/ports/virtual registers:

Clock Factor: The Clock factor is determined by Oscillator frequency and CPU frequency.

In fact, Clock factor is derived from CPU awareness or from peripheral simulation model if available.

3.4.3 Restrictions

The features like Self-Clock Mode and System Reset Generator for hardware failures are not completely-simulated/not simulated. Some other bits which are not simulated/not fully simulated: refer to the table below

3.4.4 Register Details (ECRGV1)

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		**Bit 7**	**6**	**5**	**4**	**3**	**2**	**1**	**Bit 0**
0x0000	R	VCOFRQ[1:0]		SYNDIV[5:0]					
SYNR	W								
0x0001	R	REFFRQ[1:0]		REFDIV[5:0]					
REFDV	W								
0x0002	R	0	0	0	POSTDIV[4:0]				
POSTDIV	W								
0x0003	R	RTIF	PORF	LVRF	LOCKIF	LOCK	ILAF	SCMIF	SCM
CRGFLG	W								
0x0004	R	RTIE	0	0	LOCKIE	0	0	SCMIE	0
CRGINT	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0005	R	PLLSEL	PSTP	XCLKS	0	PLLWAI	0	RTIWAI	COPWAI
CLKSEL	W								
0x0006	R	CME	PLLON	FM1	FM0	FSTWKP	PRE	PCE	SCME
PLLCTL	W								
0x0007	R	RTDEC	RTR6	RTR5	RTR4	RTR3	RTR2	RTR1	RTR0
RTICTL	W								
0x0008	R	WCOP	RSBCK	0	0	0	CR2	CR1	CR0
COPCTL	W			WRTMASK					
0x0009	R	0	0	0	0	0	0	0	0
FORBYP	W								
0x000A	R	0	0	0	0	0	0	0	0
CTCTL	W								
0x000B	R	0	0	0	0	0	0	0	0
ARMCOP	W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.5 Analog-to-Digital Converter (ADC12B16CV1)

3.5.1 Implementation

ADC_12B16C is implemented as an extension of ATD module separately and not over the existing ATD module implementation. Most of the features of this peripheral module have been simulated.

The basic implementation gets the total number of conversions to be done, calculates conversion time and sets up an event for each conversion.

3.5.2 Extensions

PADx

The analog inputs channels are reachable separately through the object pool. They are implemented as Non Memory Mapped ADC virtual registers called PAD0 to PAD15. For the ADC module, PAD0 input corresponds to the AN0 pin of the microcontroller. The following command is used to provide input voltage to analog channel in simulator:

ATDx_SETPAD <CHANNEL> <VOLTAGE AS FLOAT>

3.5.3 Restrictions

The unimplemented hardware features of ADC12B16CV1 are as follow:

Configurable external trigger functionality on any AD channel or any of four additional trigger inputs. The four additional trigger inputs can be chip external or internal.

Background Debug Freeze Enable — when debugging an application, it is useful in many cases to have the ATD pause when a breakpoint (Freeze Mode) is encountered. The bits [FRZ0:1] I ATDCTL3 register determine how the ATD will respond to a breakpoint.

3.5.4 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000	R		0	0	0	WRAP3	WRAP2	WRAP1	WRAP0
[ATDCTL0]	W								
0x0001	R	ETRIGSEL	SRES1	SRES0	SMP_DIS	ETRIGCH3	ETRIGCH2	ETRIGCH1	ETRIGCH0
[ATDCTL1]	W								
0x0002	R	0	AFFC	ICLKSTP	ETRIGLE	ETRIGP	ETRIGE	ASCIE	ACMPIE
[ATDCTL2]	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0003	R	DJM	S8C	S4C	S2C	S1C	FIFO	FRZ1	FRZ0
[ATDCTL3]	W								
0x0004	R	SMP2	SMP1	SMP0	PRS[4:0]				
[ATDCTL4]	W								
0x0005	R	0	SC	SCAN	MULT	CD	CC	CB	CA
[ATDCTL5]	W								
0x0006	R	SCF	0	ETORF	FIFOR	CC3	CC2	CC1	CC0
[ATDSTAT0]	W								
0x0007	R	0	0	0	0	0	0	0	0
Unimplemented	W								
0x0008	R	CMPE[15:8]							
[ATDCMPEH]	W								
0x0009	R	CMPE[7:0]							
[ATDCMPEL]	W								
0x000A	R	CCF[15:8]							
[ATDSTAT2H]	W								
0x000B	R	CCF[7:0]							
[ATDSTAT2L]	W								
0x000C	R	IEN[15:8]							
[ATDDIENH]	W								
0x000D	R	IEN[7:0]							
[ATDDIENL]	W								
0x000E	R	CMPHT[15:8]							
ATDCMPHTH	W								
0x000F	R	CMPTH[7:0]							
ATDCMPHTL	W								

Left Justified(DJM = 0), ATD Conversion Result Register (ATDDRn)

Register Offset / Name		Bit 7		6		5		4		3		2		1		Bit 0	
0x0010,12, 14,16,18, 1A,1C,1E, 20,22,24, 26,28,2A, 2C,2E	R	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0	0	0	0
ATDDR0 -15	W																

Right Justified (DJM = 1), ATD Conversion Result Register (ATDDRn)

Register Offset / Name		Bit 7		6		5		4		3		2		1		Bit 0	
0x0010,12, 14,16,18, 1A,1C,1E, 20,22,24, 26,28,2A, 2C,2E	R					Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit0
ATDDR0 -15	W																

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.6 Enhanced Capture Timer (S12ECT16B8CV2)

3.6.1 Implementation

The Enhanced Capture Timer (ECT) peripheral module evolved out of the Timer module (TIM) .The enhanced capture timer has 8 input capture, output compare (IC/OC) channels, same as on the HC12 standard timer (timer channels TC0 to TC7).Four IC channels (channels 7–4) are the same as on the standard timer with one capture register each that memorizes the timer value captured by an action on the associated input pin. Four other IC channels (channels 3–0), in addition to the capture register, also have one buffer each called a holding register. Using input control overwrite functionality, the capture register or its holding register cannot be written by an event unless they are empty.

Four 8-bit pulse accumulators are associated with the four buffered IC channels (channels 3–0). Each pulse accumulator has a holding register to memorize their value by an action on its external input. Each pair of pulse accumulators can be used as a 16-bit pulse accumulator.

The 16-bit modulus down-counter can control the transfer of the IC registers and the pulse accumulator's contents to the respective holding registers for a given period, every time the count reaches zero. The modulus down-counter can also be used as a stand-alone time base with periodic interrupt capability.

3.6.2 Extensions

The port associated with the ECT and TIM module is the port T, however its implementation has been done in the PIM module that handles communication with ECT and I/O behavior of the port when the ECT /TIM is disconnected from the pins.

The registers conditioning the connection between ECT and PIM are the TIOS, and the TCTL1-4 registers. The virtual PORTT register and PORTTBit0-7 present in the ECT and TIM are partial images of PTT. They represent the PTT value if the ECT/TIM alone would control the port.

3.6.3 Restrictions

In FREEZE mode, the system behaves as if it would be in stop mode. This is due to the fact that the simulator clock stops in FREEZE mode unlike the hardware.
In the simulator, the holding registers are considered as "empty" if their corresponding IC register is "empty" as well.
The timer prescaler clock is always used as timer counter clock as the simulator handles only prescaler Clock,
The Test Mode which is one of the special mode of operation is not available in simulator
The Pulse Accumulator A (PAA) does not the simulate the clock feedback feature to the timer entry.

3.6.4 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 TIOS	R W	IOS7	IOS6	IOS5	IOS4	IOS3	IOS2	IOS1	IOS0
0x0001	R	0	0	0	0	0	0	0	0
CFORC	W	FOC7	FOC6	FOC5	FOC4	FOC3	FOC2	FOC1	FOC0
0x0002 OC7M	R W	OC7M7	OC7M6	OC7M5	OC7M4	OC7M3	OC7M2	OC7M1	OC7M0
0x0003 OC7D	R W	OC7D7	OC7D6	OC7D5	OC7D4	OC7D3	OC7D2	OC7D1	OC7D0
0x0004 TCNTH	R W	TCNT15	TCNT14	TCNT13	TCNT12	TCNT11	TCNT10	TCNT9	TCNT8
0x0005 TCNTL	R W	TCNT7	TCNT6	TCNT5	TCNT4	TCNT3	TCNT2	TCNT1	TCNT0
0x0006	R	TEN	TSWAI	TSFRZ	TFFCA	PRNT	0	0	0
TSCR1	W								
0x0007 TTOV	R W	TOV7	TOV6	TOV5	TOV4	TOV3	TOV2	TOV1	TOV0
0x0008 TCTL1	R W	OM7	OM7	OM6	OM6	OM5	OL5	OM4	OL4
0x0009 TCTL2	R W	OM3	OL3	OM2	OL2	OM1	OL1	OM0	OL0
0x000A TCTL3	R W	EDG7B	EDG7A	EDG6B	EDG6A	EDG5B	EDG5A	EDG4B	EDG4A
0x000B TCTL4	R W	EDG1B	EDG0A	EDG1B	EDG0A	EDG1B	EDG0A	EDG0B	EDG0A
0x000C TIE	R W	C7I	C6I	C5I	C4I	C3I	C2I	C1I	C0I
0x000D	R	TOI	0	0	0	TCRE	PR2	PR1	PR0
TSCR2	W								
0x000E TFLG1	R W	C7F	C6F	C5F	C4F	C3F	C2F	C1F	C0F

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x000F TFLG2	R	TOF	0	0	0	0	0	0	0
	W								
0x0010 TC0(High)	R W	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0x0011 TC0(Low)	R W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0012 TC1(High)	R W	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0x013 TC1(Low)	R W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0014 TC2(High)	R W	Bit15	Bit14	Bit3	Bit12	Bit11	Bit10	Bit9	Bit8
0x0015 TC2(Low)	R W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0016 TC3(High)	R W	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0x0017 TC3(Low)	R W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0018 TC4(High)	R W	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0x0019 TC4(Low)	R W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x001A TC5(High)	R W	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0x01B TC5(Low)	R W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x001C TC6(High)	R W	Bit15	Bit14	Bit3	Bit12	Bit11	Bit10	Bit9	Bit8
0x001D TC6(Low)	R W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x001E TC7(High)	R W	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0x001F TC7(Low)	R W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0020 PACTL	R	0	PAEN	PAMOD	PEDGE	CLK1	CLK0	PAOVI	PAI
	W								
0x021 PAFLG	R	0	0	0	0	0	0	PAOVF	PAIF
	W								
0x0022 PACN3	R W	PACNT7 (15)	PACNT6 (14)	PACNT5 (13)	PACNT4 (12)	PACNT3 (11)	PACNT2 (10)	PACNT1 (9)	PACNT0 (8)
0x0023 PACN2	R W	PACNT7	PACNT6	PACNT5	PACNT4	PACNT3	PACNT2	PACNT1	PACNT0
0x0024 PACN1	R W	PACNT7 (15)	PACNT6 (14)	PACNT5 (13)	PACNT4 (12)	PACNT3 (11)	PACNT2 (10)	PACNT1 (9)	PACNT0 (8)
0x0025 PACN0	R W	PACNT3	PACNT3	PACNT3	PACNT3	PACNT3	PACNT2	PACNT1	PACNT0
0x0026 MCCTL	R	MCZI	MODMC	RDMCL	0	0	MCEN	MCPR1	MCPR0
	W				ICLAT	FLMC			
0x0027 MCFLG	R	MCZF	0	0	0	POLF3	POLF2	POLF1	POLF0
	W								
0x0028 ICPAR	R	0	0	0	0	PA3EN	PA2EN	PA1EN	PA0EN
	W								
0x0029 DLYCT	R W	DLY7	DLY6	DLY5	DLY4	DLY3	DLY2	DLY1	DLY0
0x002A ICOVW	R W	NOVW7	NOVW6	NOVW5	NOVW4	NOVW3	NOVW2	NOVW1	NOVW0
0x02B ICSYS	R W	SH37	SH26	SH15	SH04	TFMOD	PACMX	BUFEN	LATQ
0x002C OCPD	R W	OCPD7	OCPD6	OCPD5	OCPD4	OCPD3	OCPD2	OCPD1	OCPD0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x002D	R	Timer Test Register							
TIMTST	W								
0x002E	R	PTPS7	PTPS6	PTPS5	PTPS4	PTPS3	PTPS2	PTPS1	PTPS0
PTPSR	W								
0x002F	R	PTMPS7	PTMPS6	PTMPS5	PTMPS4	PTMPS3	PTMPS2	PTMPS1	PTMPS0
PITMCPSR	W								
0x0030	R	0	PBEN	0	0	0	0	PBOVI	0
PBCTL	W								
0x0031	R							PBOVF	
PBFLG	W								
0x032	R	PA3H0	PA3H0	PA3H0	PA3H0	PA3H0	PA3H0	PA3H0	PA3H0
PA3H	W								
0x0033	R	PA2H7	PA2H6	PA2H5	PA2H4	PA2H3	PA2H2	PA2H1	PA2H0
PA2H	W								
0x0034	R	PA1H7	PA1H6	PA1H5	PA1H4	PA1H3	PA1H2	PA1H1	PA1H0
PA1H	W								
0x0035	R	PA0H7	PA0H6	PA0H5	PA0H4	PA0H3	PA0H2	PA0H	PA0H0
PA0H	W								
0x0036	R	MCCNT15	MCCNT14	MCCNT13	MCCNT12	MCCNT11	MCCNT10	MCCNT9	MCCNT8
MCCNT(High)	W								
0x0037	R	MCCNT7	MCCNT6	MCCNT5	MCCNT4	MCCNT3	MCCNT2	MCCNT1	MCCNT0
MCCNT(Low)	W								
0x0038	R	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8
TC0H(High)	W								
0x0039	R	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0
TC0H(Low)	W								
0x03A	R	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8
TC1H(High)	W								
0x003B	R	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0
TC1H(Low)	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x003C	R	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8
TC2H(High)	W								
0x003D	R	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0
TC2H(Low)	W								
0x003E	R	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8
TC3H(High)	W								
0x003F	R	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0
TC3H(Low)	W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.7 Periodic Interrupt Timer (S12PIT24B8CV1)

3.7.1 Implementation

The period interrupt timer (PIT) is an array of 24-bit timers that can be used to trigger peripheral modules or raise periodic interrupts. The implementation provides support for PIT24B8CV1 version of PIT modules i.e. 24 bit eight channel timers .The timers are implemented as modulus down-counters with independent time-out periods that gives time-out interrupts. The time when an interrupt has to be raised is pre-calculated and an event is setup. Each time, the timeout period for an interrupt is changed by writing to registers, we have to recalculate the event time.

3.7.2 Restrictions

The PIT module contains eight hardware trigger signal lines PITTRIG depending on module and version, one for each timer channel, for which simulated functionality has not been implemented.
When entering in FREEZE or WAIT mode, the system behaves like in STOP mode so the simulated functionality for PITSWAI mode is same as that of STOP mode.
When the program is running in Background Debug Mode (BDM) PITFRZ bit is checked .If the PITFRZ bit is set, the PIT module is stalled. The simulated functionality for freeze mode has not been implemented.

3.7.3 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000	R	PITE	PITSWAI	PITFRZ	0	0	0	0	0
PICFLMT	W							PFLMT1	PFLMT0
0x0001	R	0	0	0	0	0	0	0	0
PITFLT	W	PFLT7	PFLT6	PFLT5	PFLT4	PFLT3	PFLT2	PFLT1	PFLT0
0x0002	R	PCE7	PCE6	PCE5	PCE4	PCE3	PCE2	PCE1	PCE0
PITCE	W								
0x0003	R	PMUX7	PMUX6	PMUX5	PMUX4	PMUX3	PMUX2	PMUX1	PMUX0
PITMUX	W								
0x0004	R	PINTE7	PINTE6	PINTE5	PINTE4	PINTE3	PINTE2	PINTE1	PINTE0
PITINTE	W								
0x0005	R	PTF7	PTF6	PTF5	PTF4	PTF3	PTF2	PTF1	PTF0
PITTF	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0006 PITMTLD0	R W	PMTLD7	PMTLD6	PMTLD5	PMTLD4	PMTLD3	PMTLD2	PMTLD1	PMTLD0
0x0007 PITMTLD1	R W	PMTLD7	PMTLD6	PMTLD5	PMTLD4	PMTLD3	PMTLD2	PMTLD1	PMTLD0
0x0008 PITLD0(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x0009 PITLD0(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x000A PITCNT0(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x000B PITCNT0(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x000C PITLD1(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x000D PITLD2(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x000E PITCNT1(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x000F PITCNT1(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x0010 PITLD2(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x0011 PITLD2(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x0012 PITCNT2(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x0013 PITCNT2(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x0014 PITLD3(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0015 PITLD3(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x0016 PITCNT3(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x0017 PITCNT3(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x0018 PITLD4(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x0019 PITLD4(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x001A PITCNT4H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x001B PITCNT4(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x001C PITLD5(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x001D PITLD5(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x001E PITCNT5(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x001F PITCNT5(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x0020 PITLD6(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x0021 PITLD6(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x0022 PITCNT6(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x0023 PITCNT6(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0024 PITLD7(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x0025 PITLD7(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x0026 PITCNT7(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x0027 PITCNT7(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.8 Periodic Interrupt Timer (S12PIT24B4CV1)

3.8.1 Implementation

The period interrupt timer (PIT) is an array of 24-bit timers that can be used to trigger peripheral modules or raise periodic interrupts. The implementation provides support for PIT24B4C version of PIT modules i.e. 24 bit four channels timer. The timers are implemented as modulus down-counters with independent time-out periods that give time-out interrupts. The time when an interrupt has to be raised is pre-calculated and an event is setup. Each time, the timeout period for an interrupt is changed by writing to registers, we have to recalculate the event time

3.8.2 Restrictions

The PIT module contains four hardware trigger signal lines PITTRIG depending on module and version, one for each timer channel, for which simulated functionality has not been implemented.
When entering in FREEZE or WAIT mode, the system behaves like in STOP mode so the simulated functionality for PITSWAI mode is same as that of STOP mode.
When the program is running in Background Debug Mode (BDM) PITFRZ bit is checked .If the PITFRZ bit is set, the PIT module is stalled. The simulated functionality for freeze mode has not been implemented.

3.8.3 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000	R	PITE	PITSWAI	PITFRZ	0	0	0	0	0
PICFLMT	W							PFLMT1	PFLMT0
0x0001	R	0	0	0	0	0	0	0	0
PITFLT	W					PFLT3	PFLT2	PFLT1	PFLT0
0x0002	R					PCE3	PCE2	PCE1	PCE0
PITCE	W								
0x0003	R					PMUX3	PMUX2	PMUX1	PMUX0
PITMUX	W								
0x0004	R					PINTE3	PINTE2	PINTE1	PINTE0
PITINTE	W								
0x0005	R					PTF3	PTF2	PTF1	PTF0
PITTF	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0006 PITMTLD0	R W	PMTLD7	PMTLD6	PMTLD5	PMTLD4	PMTLD3	PMTLD2	PMTLD1	PMTLD0
0x0007 PITMTLD1	R W	PMTLD7	PMTLD6	PMTLD5	PMTLD4	PMTLD3	PMTLD2	PMTLD1	PMTLD0
0x0008 PITLD0(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x0009 PITLD0(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x000A PITCNT0(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x000B PITCNT0(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x000C PITLD1(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x000D PITLD2(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x000E PITCNT1(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x000F PITCNT1(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x0010 PITLD2(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x0011 PITLD2(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x0012 PITCNT2(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x0013 PITCNT2(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x0014 PITLD3(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0015 PITLD3(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x0016 PITCNT3(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x0017 PITCNT3(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x0018- 0x0027	R W	Reserved Registers							

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.9 Periodic Interrupt Timer (S12PIT24B2CV1)

3.9.1 Implementation

The period interrupt timer (PIT) is an array of 24-bit timers that can be used to trigger peripheral modules or raise periodic interrupts. The implementation provides support for PIT24B2C version of PIT modules i.e. 24 bit two channels timer. The timers are implemented as modulus down-counters with independent time-out periods that give time-out interrupts. The time when an interrupt has to be raised is pre-calculated and an event is setup. Each time, the timeout period for an interrupt is changed by writing to registers, we have to recalculate the event time

3.9.2 Restrictions

The PIT module contains two hardware trigger signal lines PITTRIG depending on module and version , one for each timer channel ,for which simulated functionality has not been implemented.
When entering in FREEZE or WAIT mode, the system behaves like in STOP mode so the simulated functionality for PITSWAI mode is same as that of STOP mode.
When the program is running in Background Debug Mode (BDM) PITFRZ bit is checked .If the PITFRZ bit is set, the PIT module is stalled. The simulated functionality for freeze mode has not been implemented.

3.9.3 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000	R	PITE	PITSWAI	PITFRZ	0	0	0	0	0
PICFLMT	W							PFLMT1	PFLMT0
0x0001	R	0	0	0	0	0	0	0	0
PITFLT	W							PFLT1	PFLT0
0x0002	R							PCE1	PCE0
PITCE	W								
0x0003	R							PMUX1	PMUX0
PITMUX	W								
0x0004	R							PINTE1	PINTE0
PITINTE	W								
0x0005	R							PTF1	PTF0
PITTF	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0006 PITMTLD0	R W	PMTLD7	PMTLD6	PMTLD5	PMTLD4	PMTLD3	PMTLD2	PMTLD1	PMTLD0
0x0007 PITMTLD1	R W	PMTLD7	PMTLD6	PMTLD5	PMTLD4	PMTLD3	PMTLD2	PMTLD1	PMTLD0
0x0008 PITLD0(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x0009 PITLD0(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x000A PITCNT0(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x000B PITCNT0(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x000C PITLD1(High)	R W	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
0x000D PITLD2(Low)	R W	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
0x000E PITCNT1(H)	R W	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
0x000F PITCNT1(L)	R W	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
0x0010- 0x0027	R W	Reserved Registers							

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.10 Pulse-Width Modulator (S12PWM8B8CV1)

3.10.1 Implementation

It has 8 channels. Each of the channels has a programmable period and duty cycle as well as a dedicated counter. A flexible clock select scheme allows a total of four different clock sources (scaled clock A/B and clock A/B) to be used with the counters. Each of the modulators can create independent continuous waveforms with software-selectable duty rates from 0% to 100%. The PWM outputs can be programmed as left aligned outputs or center aligned outputs. The channels can be eight 8 bit separate or four 16 bit concatenated channels. In emergency shutdown mode, channel 7 acts as input.

3.10.2 Extensions

The port associated with the PWM module is the port P, however its implementation has been done in the PIM module that handles communication with PWM and I/O behavior of the port when the PWM is disconnected from the pins.

The registers conditioning the connection between PWM and PIM is PWME. The virtual PORTP register in the PWM are partial images of PTP. They represent the PTP value if the PWM alone would control the port.

3.10.3 Restrictions

Writing to the bits of PWMSDN register as bitwise instructions, the instruction is executed like it'll read the byte of PWMSDN register and write the bits from the bitwise instructions and write back same values in other bits. If the bit PWM Interrupt Flag (PWMIF) is 1 which reflects change on PWM7IN input, this bit is now cleared by again writing 1 on it. It may lead to incorrect results.

3.10.4 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 PWME	R / W	PWME7	PWME6	PWME5	PWME4	PWME3	PWME2	PWME1	PWME0
0x0001 PWMPOL	R / W	PPOL7	PPOL6	PPOL5	PPOL4	PPOL3	PPOL2	PPOL1	PPOL0
0x0002 PWMCLK	R / W	PCLK7	PCLK6	PCLK5	PCLK4	PCLK3	PCLK2	PCLK1	PCLK0
0x0003 PWMPRCLK	R	0	PCKB2	PCKB1	PCKB0	0	PCKA2	PCKA1	PCKA0
	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0004 PWMCAE	R/W	CAE7	CAE6	CAE5	CAE4	CAE3	CAE2	CAE1	CAE0
0x0005 PWMCTL	R	CON67	CON45	CON23	CON01	PSWAI	PFRZ	0	0
	W								
0x0006 PWMTST	R	0	0	0	0	0	0	0	0
	W								
0x0007 PWMPRSC	R	0	0	0	0	0	0	0	0
	W								
0x0008-09 PWMSCLA-B	R/W								
0x000A PWMSCNTA	R	0	0	0	0	0	0	0	0
	W								
0x000B PWMSCNTB	R	0	0	0	0	0	0	0	0
	W								
0x000C-13 PWMCNT0-7	R	0	0	0	0	0	0	0	0
	W								
0x0014-1B PWMPER0-7	R/W								
0x001C-23 PWMDTY0-7	R/W								
0x0024 PWMSDN	R	PWMIF	PWMIE	0	PWMLVL	0	PWM7IN	PWM7INL	PWM7ENA
	W			PWMRSTRT					

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.11 Serial Communication Interface (S12SCIV5)

3.11.1 Implementation

The basic implementation pushes the data in the buffer and pops up the data from the buffer depending on the baud rate specified. In SCI Module the Baud Rate Generation, Transmission and Receiver functionalities have been implemented. The interrupt TDRE, TC, RDRF, OR, IDLE in SCI status register 1 (SCISR1) are supported in the SCI modules.

3.11.2 Extension

There are two non-memory mapped SCI virtual registers that are used as token interface to receive or transmit 8 or 9 bits data. The register ‘SerialInput’ serves to send characters to the SCI Module. The register ‘SerialOutput’ contain the characters sent from to the SCI Module.

SCI_DEF0_INPUT_NAME (SerialInput)

This is a non memory mapped register and its serves to connect the SCI to the terminal window. The ninth bit is not supported. A read access to SerialInput has no specified meaning. Bit 7..0 data is received from terminal window to SCI

SCI_DEF0_OUTPUT_NAME (SerialOutput)

This not memory mapped register and it serves to connect the SCI to the terminal window. The ninth bit is not supported. A write access to SerialOutput has no specified meaning. Bit 7..0 data is sent from SCI to terminal window.

3.11.3 Restrictions

In S12SCI_V5, registers SCIASR1 (0x0000), SCIACR1 (0x0001), SCIACR2 (0x00002) become visible only in the hardware and are not simulated. Also the interrupts RXEDGIF, BERRIF, BKDIF related with these registers are not supported.
The Parity check, Noise detection, SCISWAI bit and WAKE bit in SCI control register 1 (SCICR1) has not been implemented in Simulator.
Infra Red encoding and decoding support as well as Infra Red enable bit (IREN) in SCI Baud Register (SCIBDH) has not been implemented in simulator.
Noise Error Flag, Parity Flag and Framing Error Flag bits in SCI status register 2 (SCISR2) has not been implemented in simulator.
The TXDIR, BRK13, TXPOL, RXPOL and AMAP bits in SCI status register 2 (SCISR2) has not been implemented in simulator.
Lin support, Single-Wire mode and LOOP operation has not been implemented in simulator.
Before starting the test for each SCI module, TXD pin of one SCI needs to be connected to RXD pin of second SCI and vice versa using PinConn module. The same test can be run on hardware after connecting TXD and RXD pins of SCI0 and SCI1 modules i.e. only SCI0 can be connected to SCI1 and vice versa

3.11.4 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000*	R	IREN	TNP1	TNP0	SBR12	SBR11	SBR10	SBR9	SBR8
SCIBDH	W								
0x0001*	R	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0
SCIBDL	W								
0x0002*	R	LOOPS	SCISWAI	RSRC	M	WAKE	ILT	PE	PT
SCICR1	W								
0x0000~	R	RXEDGIF	0	0	0	0	BERRV	BERRIF	BKDIF
SCIASR1	W								
0x0001~	R	RXEDGIE	0	0	0	0	0	BERRIE	BKDIE
SCIACR1	W								
0x0002~	R	0	0	0	0	0	BERRMI	BERRM0	BKDFE
SCIACR2	W								
0x0003	R	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
SCICR2	W								
0x0004	R	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
SCISR1	W								
0x0005	R	AMAP	0	0	TXPOL	RXPOL	BRK13	TXDIR	RAF
SCISR2	W								
0x0006	R	R8	T8	0	0	0	0	0	0
SCIDRH	W								
0x0007	R	R7	R6	R5	R4	R3	R2	R1	R0
SCIDRL	W	T7	T6	T5	T4	T3	T2	T1	T0

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

* Those registers are accessible if the AMAP bit in the SCISR2 register is set to zero

~ Those registers are accessible if the AMAP bit in the SCISR2 register is set to one

3.12 Timer Module (TIM16B8CV2)

3.12.1 Implementation

The basic timer consists of a 16-bit, software-programmable counter driven by an enhanced programmable prescaler. The timer can be used for many purposes, including input waveform measurements while simultaneously generating an output waveform. Pulse widths can vary from microseconds to many seconds.

This version of TIM contains 8 complete input capture/output compare(IC/OC) channels and one pulse accumulator. The input capture function is used to detect a selected transition edge and record the time. The output compare function is used for generating output signals or for timer software delays. The 16-bit pulse accumulator is used to operate as a simple event counter or a gated time accumulator. The pulse accumulator shares timer channel 7 when in event mode.

3.12.2 Extensions

The port associated with the ECT and TIM module is the port T, however its implementation has been done in the PIM module that handles communication with ECT and I/O behavior of the port when the ECT /TIM is disconnected from the pins.

The registers conditioning the connection between ECT and PIM are the TIOS, and the TCTL1-4 registers. The virtual PORTT register and PORTTBit0-7 present in the ECT and TIM are partial images of PTT. They represent the PTT value if the ECT/TIM alone would control the port.

3.12.3 Restrictions

In FREEZE mode, the system behaves as if it would be in stop mode. This is due to the fact that the simulator clock stops in FREEZE mode unlike the hardware.
In the simulator, the holding registers are considered as “empty” if their corresponding IC register is “empty” as well.
The timer prescaler clock is always used as timer counter clock as the simulator handles only prescaler Clock,
The Test Mode which is one of the special mode of operation is not available in simulator
The Pulse Accumulator A (PAA) does not the simulate the clock feedback feature to the timer entry.

3.12.4 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 TIOS	R W	IOS7	IOS6	IOS5	IOS4	IOS3	IOS2	IOS1	IOS0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0001 CFORC	R	0	0	0	0	0	0	0	0
	W	FOC7	FOC6	FOC5	FOC4	FOC3	FOC2	FOC1	FOC0
0x0002 OC7M	R/W	OC7M7	OC7M6	OC7M5	OC7M4	OC7M3	OC7M2	OC7M1	OC7M0
0x0003 OC7D	R/W	OC7D7	OC7D6	OC7D5	OC7D4	OC7D3	OC7D2	OC7D1	OC7D0
0x0004 TCNTH	R/W	TCNT15	TCNT14	TCNT13	TCNT12	TCNT11	TCNT10	TCNT9	TCNT8
0x0005 TCNTL	R/W	TCNT7	TCNT6	TCNT5	TCNT4	TCNT3	TCNT2	TCNT1	TCNT0
0x0006 TSCR1	R	TEN	TSWAI	TSFRZ	TFFCA	PRNT	0	0	0
	W								
0x0007 TTOV	R/W	TOV7	TOV6	TOV5	TOV4	TOV3	TOV2	TOV1	TOV0
0x0008 TCTL1	R/W	OM7	OM7	OM6	OM6	OM5	OL5	OM4	OL4
0x0009 TCTL2	R/W	OM3	OL3	OM2	OL2	OM1	OL1	OM0	OL0
0x000A TCTL3	R/W	EDG7B	EDG7A	EDG6B	EDG6A	EDG5B	EDG5A	EDG4B	EDG4A
0x000B TCTL4	R/W	EDG1B	EDG0A	EDG1B	EDG0A	EDG1B	EDG0A	EDG0B	EDG0A
0x000C TIE	R/W	C7I	C6I	C5I	C4I	C3I	C2I	C1I	C0I
0x000D TSCR2	R	TOI	0	0	0	TCRE	PR2	PR1	PR0
	W								
0x000E TFLG1	R/W	C7F	C6F	C5F	C4F	C3F	C2F	C1F	C0F
0x000F TFLG2	R	TOF	0	0	0	0	0	0	0
	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0010 - 0x001F	R W	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
TCxH -TCxL	R W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0020 PACTL	R W	0	PAEN	PAMOD	PEDGE	CLK1	CLK0	PAOVI	PAI
0x021 PAFLG	R W	0	0	0	0	0	0	PAOVF	PAIF
0x0022 PACNTH	R W	PACNT15	PACNT14	PACNT13	PACNT12	PACNT11	PACNT10	PACNT9	PACNT8
0x0023 PACNTL	R W	PACNT7	PACNT6	PACNT5	PACNT4	PACNT3	PACNT2	PACNT1	PACNT0
0x0024 - 2B RESERVED	R W								
0x002C OCPD	R W	OCPD7	OCPD6	OCPD5	OCPD4	OCPD3	OCPD2	OCPD1	OCPD0
0x002D PTPSR	R W	PTPS7	PTPS6	PTPS5	PTPS4	PTPS3	PTPS2	PTPS1	PTPS0
0x002E RESERVED	R W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.13 Voltage Regulator (S12VREGL3V3V1)

3.13.1 Implementation

In the VREG module, only autonomous periodical interrupt (API) functionality is implemented.

When internal RC oscillator is used as clock source, the period can be increased or decreased by 25% using trimming register.

3.13.2 Extensions

The virtual register APIout is implemented which is used to generate a clock or a high pulse at the end of a selected period, depending on the configuration of APIES bit in VREGAPICL register. This signal is routed to a port pin.

3.13.3 Restrictions

None of the Voltage regulators are implemented.

Following modes are not implemented.

Reduced power mode (RPM) (MCU is in stop mode)
Shutdown mode

3.13.4 Register Details

This table gives a detailed view on the registers of this peripheral module. Registers and bits not or only partially implemented will be color encoded.

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x02F0	R	0	0	VSEL	VAE	0		0	
VREGHTCL	W								
0x02F1	R	0	0	0	0	0	LVDS	LVIE	LVIF
VREGCTRL	W								
0x02F2	R	APICLK	0	0	APIES	APIEA	APIFE	APIE	APIF
VREGAPICL	W								
0x02F3	R	APITR5	APITR4	APITR3	APITR2	APITR1	APITR0	0	0
VREGAPITR	W								
0x02F4	R	APIR15	APIR14	APIR13	APIR12	APIR11	APIR10	APIR9	APIR8
VREGAPIRH	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x02F5	R	APIR7	APIR6	APIR5	APIR4	APIR3	APIR2	APIR1	APIR0
VREGAPIRL	W								
0x02F6	R	0	0	0	0	0	0	0	0
RESERVED	W								
0x02F7	R		0	0	0				
VREGHTTR	W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)